

UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ  
DEPARTAMENTO ACADÊMICO DE ELETRÔNICA  
CURSO DE ENGENHARIA ELETRÔNICA

ANGELICA CRISTINA MANFRIM

**PROJETO E IMPLEMENTAÇÃO DE TRANSDUTORES PARA UM SISTEMA DE  
MONITORAMENTO AGROMETEOROLÓGICO**

TRABALHO DE CONCLUSÃO DE CURSO

CAMPO MOURÃO

2018

ANGELICA CRISTINA MANFRIM

**PROJETO E IMPLEMENTAÇÃO DE TRANSDUTORES PARA UM SISTEMA DE  
MONITORAMENTO AGROMETEOROLÓGICO**

Trabalho de Conclusão de Curso do curso Superior de Engenharia Eletrônica do Departamento Acadêmico de Eletrônica - DAELN - da Universidade Tecnológica Federal do Paraná (UTFPR) do Campus Campo Mourão, como requisito para obtenção do título de Bacharel em Engenharia Eletrônica.

Orientador: Prof. Dr. Gilson Junior Schiavon

CAMPO MOURÃO

2018

---

**TERMO DE APROVAÇÃO**  
**DO TRABALHO DE CONCLUSÃO DE CURSO INTITULADO**

Projeto e implementação de transdutores para um sistema de  
monitoramento agrometeorológico

por

Angélica Cristina Manfrim

Trabalho de Conclusão de Curso apresentado no dia 19 de Novembro de 2018 ao Curso Superior de Engenharia Eletrônica da Universidade Tecnológica Federal do Paraná, Campus Campo Mourão. O Candidato foi arguido pela Banca Examinadora composta pelos professores abaixo assinados. Após deliberação, a Banca Examinadora considerou o trabalho aprovado.

---

Prof. Roberto Ribeiro Neli  
(UTFPR)

---

Prof. Osmar Tormena Junior  
(UTFPR)

---

Prof. Gilson Junior Schiavon  
(UTFPR)  
Orientador

## **AGRADECIMENTOS**

Gostaria de agradecer primeiramente a Deus, por ter sido um refúgio nos momentos difíceis ao longo do caminho.

Agradeço a minha família, por todo apoio emocional e financeiro, em especial meu pai Antonio, minha mãe Floripes, meu irmão Junior, e meus avós: Antenor, Josias, Hilda e Lourdes. Ao meu amigo e amado Marcelo, que sempre esteve ao meu lado nos momentos bons e ruins. E a minha gata Luna, e minhas cachorras Kaira, Choinha e Paraguaio, que sempre me motivaram a voltar para casa.

Agradeço ao meu orientador Gilson por todo conhecimento adquirido em todos esses anos de trabalho, por sempre confiar e acreditar em mim. Agradeço ao professor Reginaldo por sempre me encorajar a não desistir e chegar até este momento. Agradeço também aos professores Aline, Eduardo, Flavio, Gabriel, Heber, Lucas, Leonardo, Marcio, Osmar e Roberto, que de alguma forma nesses longos anos fizeram a diferença na minha vida acadêmica. Agradeço também ao técnico do almoxarifado Reginaldo por ter sido sempre solícito comigo.

Agradeço aos meus amigos Alisson, Andressa, Ariel, Caio, Dionni, Kawe, Guilherme, Gustavo Chicha, Gustavo Melo, Lucas, Marcia, Rafael, Richard, Rodrigo, Vitor, Victor, que dividiram comigo de alguma forma essa empreitada.

Agradeço a bolsa recebida no Edital 3/2018 - PROGRAD/PROREC: “Edital de Apoio à Execução de Trabalhos de Conclusão de Curso – TCC”, com recursos da PROGRAD e da PROREC.

*“Eu não pertencço a esse mundo...  
As coisas são como são...  
Algo me separa das outras pessoas...  
Para onde quer que eu vire...  
Há algo bloqueando minha fuga.”  
(Tradução livre)*

*13 Beaches (DEL REY, Lana. 2017)*

## RESUMO

A modernização da agricultura é um ramo crescente da tecnologia. Como as condições ambientais influenciam diretamente no crescimento e produção de uma planta, saber as variáveis de temperatura, umidade, pluviosidade e irradiação solar pode auxiliar na tomada de decisões: quando plantar, adubar, colher e afins. Este trabalho apresenta o projeto de uma Estação Agrometeorológica, para realizar o monitoramento das condições em um ambiente agrícola, com os sensores de umidade do ar e do solo, temperatura do ar e do solo, temperatura e tensão da bateria, e velocidade do ar. Os circuitos foram testados em *protoboard* e desenvolvidos em placa de circuito impresso. Os dados são apresentados em um *display* LCD e salvos em um cartão SD por meio do *shield logger*. A placa microcontrolada utilizada foi a do Aduino UNO e o código foi desenvolvido em sua linguagem própria de programação. Para a proteção dos sensores, placa e bateria, foi construída a parte física da estação. O projeto monitorou por algumas horas os sensores, no qual foi possível então plotar os gráficos com as variações medidas, e realizar algumas análises sobre os resultados obtidos. Notou-se ser um projeto viável e implementável, necessitando de algumas melhorias na fabricação da placa e alteração de sensores utilizados.

## **ABSTRACT**

Modernization of agriculture is a growing branch of technology. The environmental conditions directly influence the growth and production of a plantation, knowing the variables of temperature, humidity, rainfall and solar irradiation can help in decision making: when planting, fertilizing, harvesting and others. This work presents the project of an Agrometeorological Station, to perform the monitoring of the conditions in an agricultural environment, with the humidity sensors of air and soil, air and soil temperature, temperature and voltage of the battery, and air velocity. The circuits were tested in protoboard and developed in circuit board. The data is displayed on an LCD display and saved on an SD card via the shield logger. The microcontroller board used was the Aduino UNO and the code was developed in its own programming language. For the protection of the sensors, board and battery, the physical part of the station was built. The project monitored the sensors for a few hours, in which it was then possible to plot the graphs with the measured variations, and to perform some analysis on the results obtained. It was noted to be a feasible and implementable project, requiring some improvements in the manufacturing of the board and alteration of the sensors used.

## LISTA DE FIGURAS

Figura 1 – Estação agrometeorológica.....	12
Figura 2 – Fluxograma do projeto .....	14
Figura 3 – Funcionamento de um transdutor.....	16
Figura 4 – Curva de corrente quiescente por temperatura do LM35 .....	17
Figura 5 – Circuito simples de utilização do LM35 .....	18
Figura 6 – Curva de capacitância por umidade .....	18
Figura 7 – Circuito para utilização do sensor .....	19
Figura 8 – Protocolo de comunicação do DHT11 .....	20
Figura 9 – Dados da comunicação do DHT11.....	20
Figura 10 – SHT20 .....	21
Figura 11 – Construção eletrônica do anemômetro .....	22
Figura 12 – Curva de relação entre a corrente reversa e a irradiação solar .....	22
Figura 13 – Circuito amplificador de transresistencia.....	23
Figura 14 – Construção da bateria de chumbo-ácido.....	24
Figura 15 – Arduino UNO .....	25
Figura 16 – Barramento de dados do protocolo I2C.....	26
Figura 17 – Diagrama de blocos do <i>display</i> LCD 16x2 .....	27
Figura 18 – Circuito de ligação do DHT11 .....	29
Figura 19 – Sinal de dados do DHT11 .....	29
Figura 20 – Circuito de ligação do SHT20.....	30
Figura 21 – Sinais SDA e SCL do sensor SHT20 .....	30
Figura 22 – Circuito de funcionamento do LM35.....	31
Figura 23 – Circuito de monitoramento da bateria .....	32
Figura 24 – Circuito de utilização do anemômetro .....	32
Figura 25 – <i>Logger Shield</i> .....	33
Figura 26 – Frases apresentadas no <i>display</i> LCD .....	35
Figura 27 – Instalação física do projeto.....	36
Figura 28 – Temperatura do solo .....	37
Figura 29 – Umidade do solo .....	38
Figura 30 – Temperatura do ar.....	38
Figura 31 – Umidade do ar.....	39
Figura 32 – Temperatura da bateria.....	39
Figura 33 – Velocidade do ar .....	40
Figura 34 – Tensão da bateria .....	41

## LISTA DE QUADROS

Quadro 1 – Pinos do <i>display</i> LCD .....	26
Quadro 2 – Conexões LCD 16X2.....	34

## LISTA DE ABREVIATURAS, SIGLAS E ACRÔNIMOS

$\Omega$	Ohm
A	Ampere
$^{\circ}\text{C}$	Graus Celsius
cm	Centímetros
D	Diodo
F	Faraday
GND	<i>Ground/Terra</i>
h	Hora
I	<i>Current/Corrente</i>
I/O	<i>Input/Output</i>
LCD	<i>Liquid Crystal Display</i>
LED	<i>Light Emitting Diode</i>
MC	Microcontrolador
m	Metros
R	Resistência
RH	Umidade Relativa
s	Segundos
V	Volt/Tensão
Vcc	Alimentação
Vdd	Alimentação
Vss	<i>Ground/Terra</i>

## SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO</b> .....	<b>11</b>
1.1	TEMA.....	11
1.1.1	Delimitação do tema.....	11
1.2	OBJETIVOS .....	12
1.2.1	Objetivo geral .....	12
1.2.2	Objetivos específicos.....	12
1.3	JUSTIFICATIVA.....	13
1.4	PROCEDIMENTOS METODOLÓGICOS .....	13
<b>2</b>	<b>FUNDAMENTAÇÃO TEÓRICA</b> .....	<b>15</b>
2.1	ESTAÇÃO AGROMETEOROLÓGICA.....	15
2.2	SENSORES E TRANSDUTORES .....	16
2.2.1	Sensores de temperatura .....	16
2.2.2	Sensor de umidade.....	17
2.2.3	Sensores eletrônicos .....	17
2.2.3.1	LM35.....	17
2.2.3.2	HS-1101 .....	17
2.2.3.3	DHT11 .....	17
2.2.3.4	SHT20 .....	20
2.2.3.5	Sensor de velocidade do ar .....	21
2.2.3.6	BPW34 .....	22
2.3	BATERIA ESTACIONARIA .....	23
2.4	ARDUINO .....	24
2.5	PROTOCOLO DE COMUNICAÇÃO I2C.....	25
2.6	DISPLAY LCD .....	26
<b>3</b>	<b>PROJETO DA ESTAÇÃO AGROMETEOROLOGICA</b> .....	<b>28</b>
3.1	DESENVOLVIMENTO DO PROJETO .....	28
3.1.1	Código .....	28
3.1.2	Sensor DHT11 .....	28
3.1.3	Sensor SHT20 .....	29
3.1.4	LM35 como sensor de temperatura da bateria .....	30
3.1.5	Monitoramento da tensão na bateria .....	31
3.1.6	Anemômetro .....	32
3.1.7	<i>Datalogger</i> .....	33
3.1.8	<i>Display lcd</i> .....	34
3.1.9	Sensores BPW34 e HS-1101 .....	35
3.1.9	Esquemático e placa de circuito impresso.....	35
3.2	CONSTRUÇÃO FÍSICA.....	36
<b>4</b>	<b>RESULTADOS E DISCUSSÕES</b> .....	<b>37</b>
4.1	COLETA DE DADOS .....	28
4.1.1	Sensores de umidade e temperatura.....	37
4.1.2	Anemômetro .....	40
4.1.2	Tensão da Bateria .....	41
<b>5</b>	<b>CONSIDERAÇÕES FINAIS</b> .....	<b>43</b>
	REFERÊNCIAS.....	44
	APÊNDICE 1 – FLUXOGRAMA DO CODIGO .....	48
	APÊNDICE 2 – CÓDIGO DE ARDUINO DA ESTAÇÃO AGROMETEOROLOGICA .....	49
	APÊNDICE 3 – ESQUEMÁTICO DO LOGGER SHIELD.....	57
	APÊNDICE 4 – ESQUEMÁTICO DO CIRCUITO .....	59

<b>APÊNDICE 5 – BOARD DO CIRCUITO .....</b>	<b>60</b>
<b>APÊNDICE 6 – PARTE DO BANCO DE DADOS.....</b>	<b>61</b>

# 1 INTRODUÇÃO

## 1.1 TEMA

Este trabalho apresenta a proposta de desenvolvimento de uma estação agrometeorológica, para monitoramento das condições ambientais de uma área agrícola.

### 1.1.1 DELIMITAÇÃO DO TEMA

O agronegócio está em constante expansão no país. Nos últimos 20 anos, a área plantada com grãos cresceu 37 % e a produção mais de 176 %. A busca por novas tecnologias que possam ser implantadas no ramo agrícola é constante, sendo o desenvolvimento tecnológico relacionado ao agronegócio de suma importância para o mesmo. (ECO AGRO, 2014).

A área da ciência que estuda os fenômenos físicos da atmosfera é a meteorologia, sendo uma de suas divisões a agrometeorologia ou meteorologia agrícola, na qual se estuda a influência das condições climáticas no meio rural. Estas condições geram impactos diretamente na produtividade de uma plantação: pragas, doenças, alterações no crescimento das plantas e afins são alguns destes. Apesar de ser uma ferramenta essencial, a agrometeorologia ainda é pouco aplicada no meio rural. (SENTELHAS, 1998).

As condições meteorológicas influenciam em todas as etapas de um cultivo agrícola: o preparo da terra; a semeadura; colheita; e armazenamento da produção. Os impactos de uma mudança climática como, por exemplo, chuva em excesso ou escassez, geram prejuízos financeiros e sociais, fatores estes que ocasionam perdas de grandes áreas plantadas ou até mesmo áreas inteiras, estes impactos podem ser minimizados pelo uso de tecnologias. (SENTELHAS, 1998).

A análise do solo e do ar, por exemplo, pode maximizar a produção a partir de dados adquiridos em sistemas de aferição e monitoramento, e, assim, gerando possíveis decisões a fim de escolher o melhor momento para plantar: conforme a umidade do solo; quando adubar conforme as condições do ar; dentre outros. (SENTELHAS, 1998).

A modernização da meteorologia agrícola se dá pela implantação de estações agrometeorológicas, em suma, possuem sensores de temperatura e de umidade, do ar e do solo, além de piranômetro (sensor de irradiação solar), sensor de velocidade do ar e de pluviosidade. (AGRO SMART, 2015). A Figura 1 apresenta um exemplo de uma estação agrometeorológica.

**FIGURA 1 – ESTAÇÃO AGROMETEOROLÓGICA.**



Fonte: AGRO SMART, 2015.

## 1.2 OBJETIVOS

### 1.2.1 OBJETIVO GERAL

O objetivo geral deste trabalho é o desenvolvimento de uma estação agrometeorológica para análise dos dados coletados pelos sensores, a qual será instalada em uma propriedade na cidade de Cianorte-PR.

### 1.2.2 OBJETIVOS ESPECÍFICOS

- Desenvolver os circuitos eletrônicos para os sensores utilizados em uma estação agrometeorológica;
- Desenvolver o código do microcontrolador para adquirir os dados, arquivá-los no cartão SD e apresentá-los no *display* LCD;

- Desenvolver a placa de circuito impresso com *display* LCD (*Liquid Crystal Display, display* de cristal líquido), reguladores de tensão e conectores para os sensores, encaixe para o *logger shield* e o Arduino UNO;
- Construir a estação agrometeorológica para proteção e acomodação dos circuitos eletrônicos;
- Instalação da estação agrometeorológica em uma propriedade;
- Verificar o funcionamento e eficiência proporcionada pelos sensores e o sistema;
- Analisar os resultados obtidos.

### 1.3 JUSTIFICATIVA

A estação pode ser instalada em uma área agrícola, sendo analisadas as condições ambientais, dando ao usuário uma gama de informações que podem influenciar na tomada de decisões, como em que momento plantar, em que momento adubar, realizar o controle de pragas e afins.

O desenvolvimento e construção da estação agrometeorológica proverá uma ferramenta para o estudo e ensino de sensores aplicados a atividade agrícola e poderá ser utilizada como uma ferramenta para algumas disciplinas do curso, como sensores industriais, sistemas embarcados, metrologia e microcontroladores.

Em conjunto com o microcontrolador foi utilizado um *display* LCD, demonstrando os dados dos sensores em tempo real de forma simplificada e prática, além de um *datalogger* que registra os dados em um arquivo, possibilitando uma análise posterior mais minuciosa e detalhada dos dados obtidos.

### 1.4 PROCEDIMENTOS METODOLÓGICOS

Foram realizados os testes dos circuitos em *protoboard*, e o mesmo foi fabricado em uma placa de circuito impresso.

Para que o usuário tenha acesso aos dados em tempo real, coletados dos sensores, foi utilizado um LCD de 16x2. Para o *datalogger*, utilizou-se um Arduino UNO e um *logger shield*, salvando os dados em um cartão SD em formato .CSV

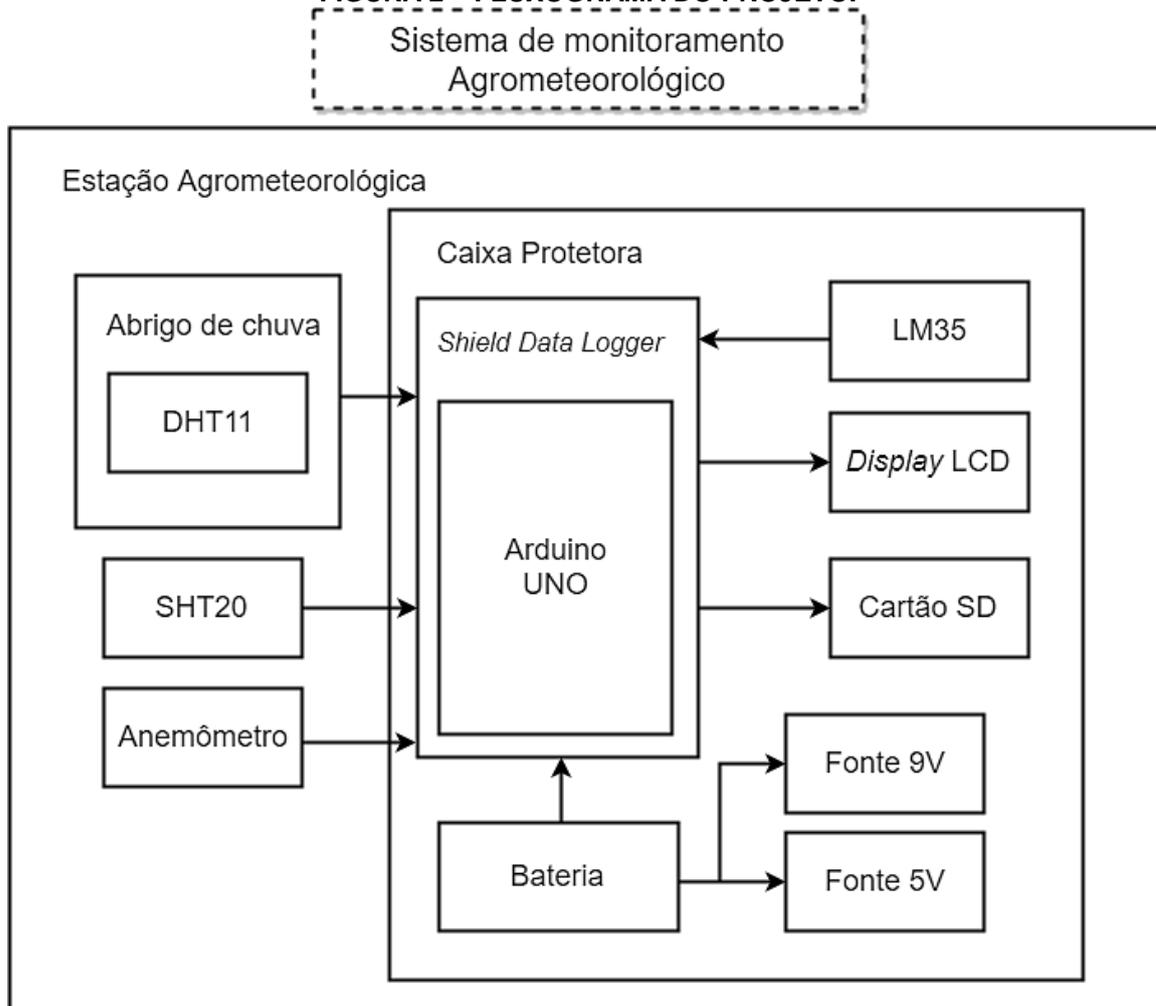
(*comma-separated values, valores separados por vírgula*), possibilitando ao usuário manipulá-los posteriormente em um *software* adequado.

Como o sistema pode ser instalado em um local aberto, foi construído um aparato para facilitar a instalação e, principalmente, proteger e acomodar todo o sistema eletrônico do meio externo. Todo o aparato é constituído por tripé para suporte e instalação, caixa protetora para bateria e placa de circuito, e o abrigo de chuva para o sensor de umidade e temperatura do ar, e o anemômetro anexado ao suporte.

Os dados foram coletados durante algumas horas. Foram plotados os gráficos e realizados suas análises.

A Figura 2 apresenta um fluxograma simplificado do projeto, omitindo circuitos intermediários.

**FIGURA 2 – FLUXOGRAMA DO PROJETO.**



Fonte: Autoria própria.

## 2 FUNDAMENTAÇÃO TEÓRICA

### 2.1 ESTAÇÃO AGROMETEOROLÓGICA

Uma estação meteorológica mede em intervalos programados de tempo informações como: temperatura do ar e solo, umidade do ar e do solo, precipitação, velocidade do ar e radiação solar. A coleta desses dados é feita a partir de sensores eletrônicos e para que a estação funcione fora da rede elétrica é necessário um sistema com bateria e placa solar. (INMET, 2011).

Segundo Sentelhas (1998), a principal diferença entre uma estação meteorológica e uma estação agrometeorológica, são os sensores, enquanto na meteorológica temos o sensor de pressão atmosférica, na agrometeorológica temos os sensores de umidade e temperatura do solo e irradiação solar. Os principais benefícios para um produtor agrícola pelo uso de uma estação agrometeorológica são:

- Manejo do solo: a umidade do solo influencia diretamente na atividade de arado e gradeação do solo, em caso de excesso de umidade pode ocorrer a compactação do solo e em caso de pouca umidade a erosão do mesmo;
- Semeadura: a germinação das sementes está diretamente ligada à umidade presente no solo;
- Irrigação: em caso de áreas com irrigação automática, saber a umidade do solo é imprescindível para um manejo adequado da quantidade de água;
- Pragas e doenças se proliferam em condições específicas, o que implica na necessidade de aplicação de defensivos agrícolas, dos quais não devem ser aplicados em dias de chuva para não contaminar o solo e mananciais de água, além de desperdiçar os agrotóxicos;
- A colheita é influenciada pela umidade do solo, já que pode impossibilitar o maquinário de atuar;
- É possível fazer uma previsão do rendimento da plantação conforme a temperatura e umidade do ambiente;
- Analisar os dados para prever riscos de incêndio na plantação.

## 2.2 SENSORES E TRANSDUTORES

Sensor é o nome dado ao dispositivo sensível a mudanças de energia do ambiente, essas mudanças podem ser luminosas, térmicas, cinéticas, e afins. Podemos relacioná-las com variáveis que queremos aferir como temperatura, luz, posição entre outros. (THOMAZINI, 2008).

Transdutor é um dispositivo completo, composto, pelo sensor, amplificador e outro circuito necessário para tornar o sinal do sensor em um sinal de corrente ou tensão, para que ele possa ser interpretado por um microcontrolador, por exemplo. A Figura 3 demonstra por meio de diagrama de blocos o sistema de funcionamento de um transdutor. (SEIDEL, 2011).



Fonte: SEIDEL, 2012.

### 2.2.1 SENSORES DE TEMPERATURA

Sensores de temperatura são dispositivos que medem a temperatura a partir de suas próprias propriedades físicas, como: dilatação térmica, campo eletromagnético, radiação térmica, resistência térmica e afins. (CITISYSTEMS, 2018).

Existem diversos componentes eletrônicos disponíveis no mercado que atuam como sensores de temperatura. São diodos, transistores e circuitos integrados, baseados em sua construção de silício e a influência da mudança de temperatura nas características do mesmo. Eles possuem a facilidade de instalação em placas de circuito impresso e uma boa precisão para utilizações em temperaturas ambiente. (THOMAZINI, 2008).

## 2.2.2 SENSOR DE UMIDADE

A umidade é a quantidade de vapor de água no ar ou quantidade de água líquida em uma amostra de solo. A umidade relativa do ar é uma relação entre quantidade de vapor de água no ar conforme a temperatura e a quantidade máxima de vapor d'água que o ar pode conter nessa temperatura. (THOMAZINI, 2008).

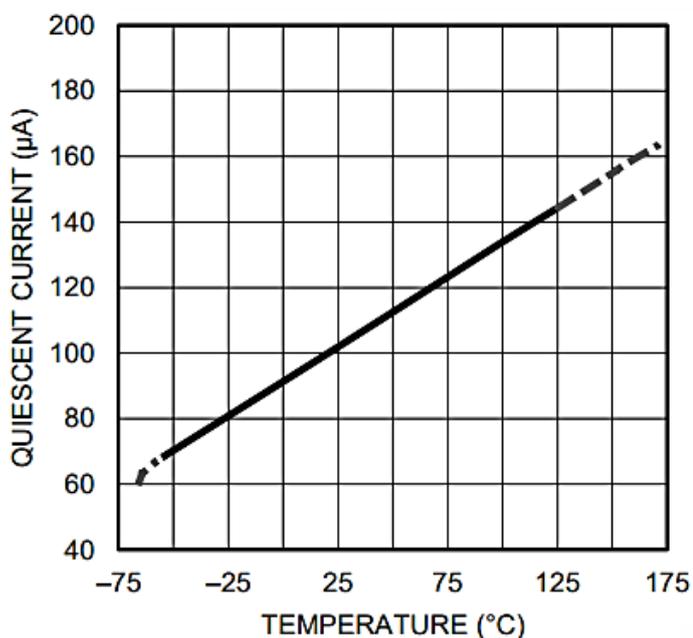
O princípio básico da medição de umidade, é baseado no efeito capacitivo ou da condutividade elétrica. Os sensores enviam um sinal proporcional à quantia de água dependendo do material a ser aferido. (THOMAZINI, 2008).

## 2.2.3 SENSORES ELETRÔNICOS

### 2.2.3.1 LM35

Um dos sensores eletrônicos mais populares é o circuito integrado LM35, com encapsulamento TO-92, 3 pinos. Sua corrente quiescente é proporcional à temperatura, que pode variar de  $-55^{\circ}\text{C}$  até  $150^{\circ}\text{C}$ , com uma precisão de  $0,5^{\circ}\text{C}$ , cada grau célsius representa 10 mV da sua tensão de saída. As Figuras 4 e 5 apresentam respectivamente a reta referente à corrente quiescente conforme a temperatura ambiente e um exemplo de circuito de utilização do dispositivo. (TEXAS, 2017).

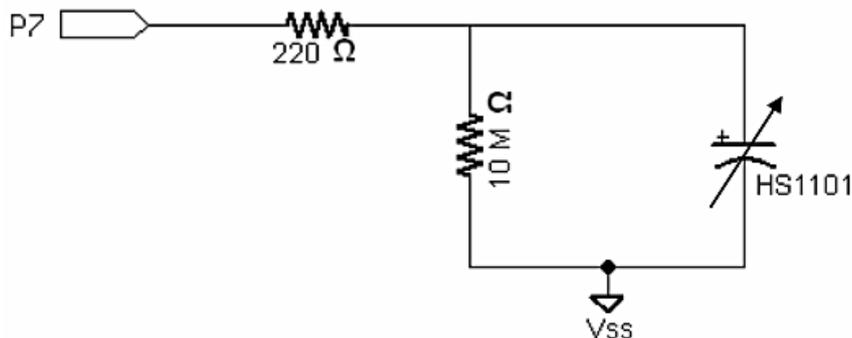
FIGURA 4 – CURVA DE CORRENTE QUIESCENTE POR TEMPERATURA DO LM35.





A partir do tempo de carga do capacitor, é possível se relacionar a medida da capacitância com a umidade relativa do ar, utilizando o sensor e um resistor como no circuito da figura 7 a seguir:

FIGURA 7 – CIRCUITO PARA UTILIZAÇÃO DO SENSOR:



Fonte: PARALLAX, 2009.

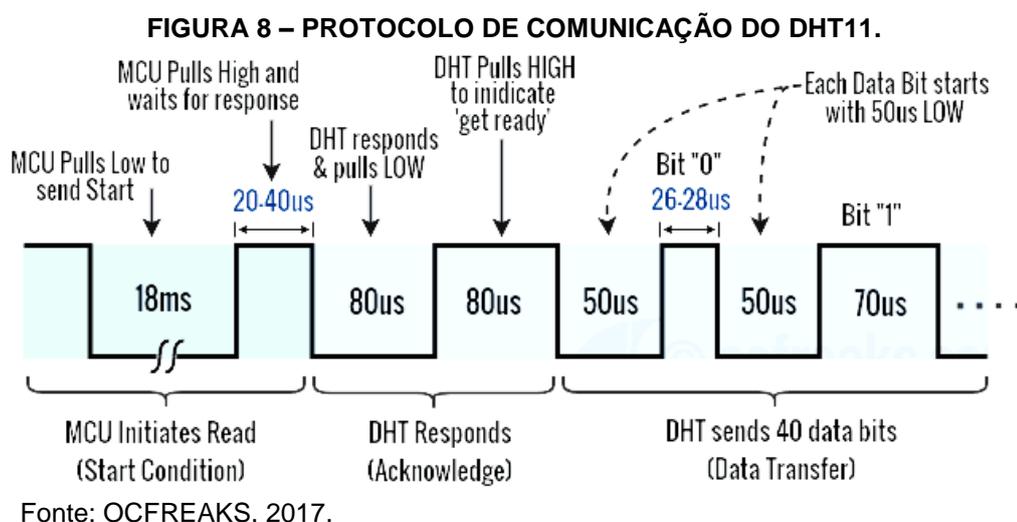
### 2.2.3.3 DHT11

DHT11 é um sensor de umidade e temperatura, com um único barramento de comunicação, possuindo um protocolo customizado próprio. Sua faixa dinâmica de medição vai de 20% a 90% de umidade relativa do ar e de 0°C a 50°C. Sua acurácia de temperatura é de  $\pm 2^{\circ}\text{C}$  e de umidade é de  $\pm 5\%$ . (MOUSER, 2017).

Opera entre 3 e 5,5 V. Possui uma resolução de 8 bits, e um período de tempo para cada amostra de, aproximadamente, 1 segundo. O sensor utiliza um protocolo de *master-slave* (mestre-escravo), sendo o microcontrolador o mestre, que inicia a comunicação e faz a leitura dos dados, e o sensor é o escravo, transmitindo os dados quando solicitado. (OCFREAKS, 2017).

O protocolo do sensor funciona da seguinte forma: se a linha está inativa, o microcontrolador a mantém em *LOW* por 18 ms. Depois o microcontrolador manda um pulso *HIGH* de 20 ms a 40 ms, esse sinal indica ao sensor o início da comunicação. O sensor então mantém o barramento em *LOW* por 80  $\mu\text{s}$ , o que indica que está preparado para enviar os dados ao microcontrolador, em seguida ele envia 40 bits de dados. (OCFREAKS, 2017).

Para identificar como sendo bit 0, o sensor envia um pulso *HIGH* de 26 à 28  $\mu\text{s}$  e para o bit 1 um pulso de 70  $\mu\text{s}$ . Para finalizar a comunicação, o mesmo mantém o barramento em *HIGH*. (OCFREAKS, 2017).



Os 40 bits de dados são divididos em 5 *bytes*: o primeiro indica a umidade relativa e o segundo os décimos da umidade relativa, que são sempre 0. O terceiro indica a temperatura e o quarto os décimos da temperatura, que são sempre 0. O quinto indica uma soma de verificação dos dados (*Checksum*). (OCFREAKS, 2017).



#### 2.2.3.4 SHT20

SHT20 é um sensor de umidade e temperatura, possuindo uma versão a prova d'água, encapsulada com um material respirável, porém impermeável, permitindo que as moléculas de água penetrem enquanto bloqueia a infiltração de gotículas de água, seu modelo é o KLDZ88. Funciona tanto com uma alimentação de 3,3 V e de 5 V. Possui uma acurácia de  $\pm 3\%$  para umidade relativa e  $\pm 0,3\%$  para a temperatura. Possuindo uma faixa dinâmica que vai de 0% a 100% de umidade relativa e de temperatura de  $-40^{\circ}\text{C}$  a  $125^{\circ}\text{C}$ . (USINAINFO, 2018). A Figura 10 a seguir apresenta o sensor e seu encapsulamento:

FIGURA 10 – SHT20.



Fonte: A autoria própria

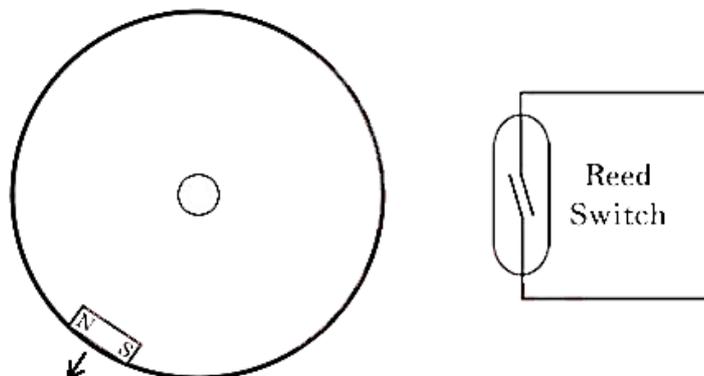
Possui quatro conectores: VDD (Alimentação), GND (Terra), SDA (*Serial Data*) e SCL (*Serial Clock*). O pino SCL é responsável por sincronizar a comunicação do sensor com o microcontrolador, e o pino SDA é responsável por transferir os dados entre o microcontrolador e o sensor. Sua comunicação é realizada por meio do protocolo I2C e sua resolução é de 12 bits os dados de umidade e 14 bits para os de temperatura. (USINAINFO, 2018).

#### 2.2.3.5 SENSOR DE VELOCIDADE DO AR

Anemômetro é um instrumento capaz de medir a velocidade do ar, muito utilizado em estações meteorológicas. O mais utilizado é o modelo de copo, no qual os copos são anexados as extremidades dos braços do dispositivo e o vento empurra os copos fazendo os braços rodarem em seu eixo central. (PORTALSAOFRANCISCO, 2018).

Sua construção eletrônica é simples, constituído por um ímã acoplado ao seu eixo principal e um *Reed Switch*, que é uma chave acionada por campo magnético. Cada vez que o ímã passa por ele, é gerado um pulso, que pode ser contado pelo microcontrolador. (BLOGUSINAINOFO, 2017). Sua construção básica é apresentada na Figura 11 a seguir:

FIGURA 11 – CONSTRUÇÃO ELETRÔNICA DO ANEMÔMETRO.



Ímã preso ao eixo

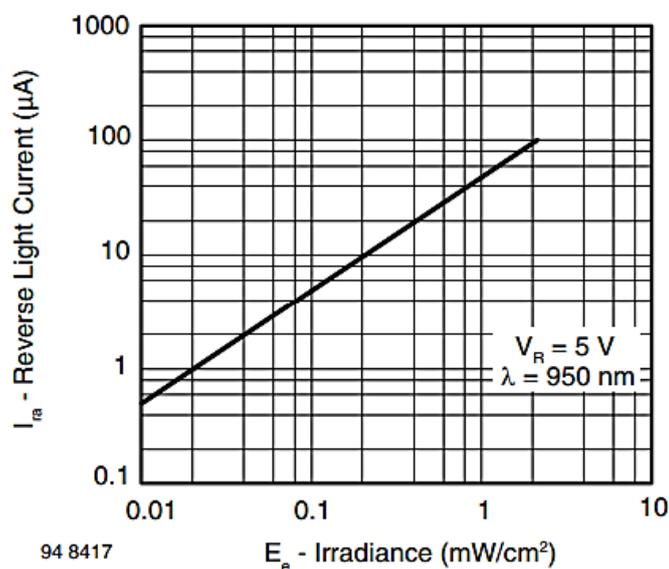
Fonte: Adaptado de BLOGUSINAINFO, 2017.

#### 2.2.3.6 BPW34

O sensor BPW34 mede a irradiação solar, sendo um fotodiodo, possui sensibilidade a comprimentos de onda entre 430 nm e 1000 nm. Quando reversamente polarizado o fotodiodo não conduz corrente elétrica na ausência de luz, apenas com sua incidência, essa corrente gerada se chama corrente reversa. (VISHAY SEMICONDUTOR, 2017).

A curva característica da corrente reversa proporcional à irradiação solar está na Figura 12.

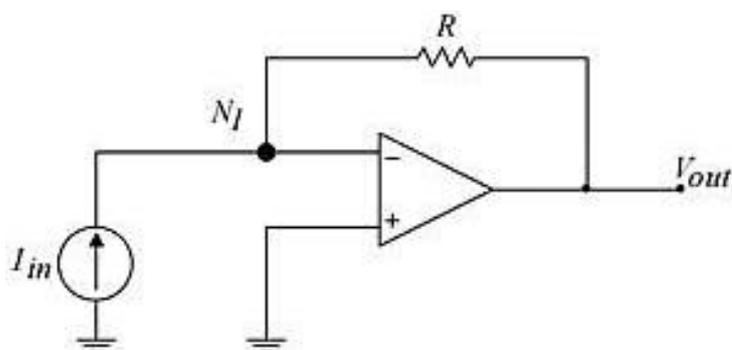
FIGURA 12 – CURVA DE RELAÇÃO ENTRE A CORRENTE REVERSA E A IRRADIAÇÃO SOLAR:



Fonte: VISHAY SEMICONDUTOR, 2017.

O sensor BPW34 possui uma relação entre a sua corrente reversa e a irradiação solar. Utilizando o amplificador operacional em modo de transresistencia, é possível converter essa corrente em tensão, com valores que podem ser calculados, este circuito está na Figura 13. (ANALOGICTIPS, 2016).

**FIGURA 13 – CIRCUITO AMPLIFICADOR DE TRANSRESISTENCIA:**



Fonte: ANALOGICTIPS, 2016.

A equação da tensão se saída relacionada à entrada é:

$$V_{out} = -R * I_{in} \quad (1)$$

### 2.3 BATERIA ESTACIONARIA

As baterias estacionarias são utilizadas em sistemas que necessitam de uma duração maior da bateria, semelhantes as baterias de carro, porém, uma capacidade de descarga maior. O tipo mais comum é o de chumbo-ácido, muito utilizado em som automotivo, *no-breaks*, centrais telefônicas, alarmes, energia solar e eólica, carro elétrico, entre outros. Em sistemas solares são utilizadas para armazenar a energia das placas solares e manter o funcionamento do sistema quando não há sol. (REDIMAX, 2017).

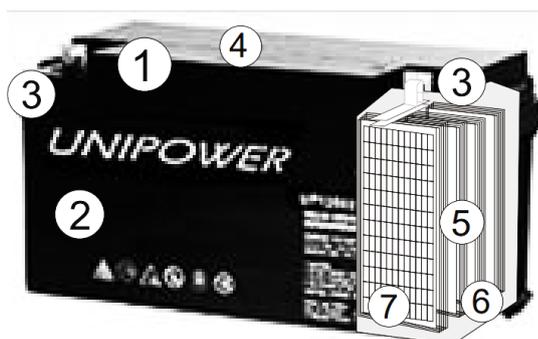
As baterias de chumbo-ácido possuem células, dependendo da fabricante, com uma média de 2 V por célula. Logo, uma bateria de 12 V, então, possui seis células. As baterias seladas dispensam manutenção e tem uma longa duração que pode chegar a 10 anos. A vida útil está diretamente relacionada à carga acoplada e o carregamento correto da bateria. (ROOBOCORE, 2018)

Segundo ROBOCORE, 2018, a bateria é composta pelos seguintes elementos:

1. Tampa
2. Vaso
3. Terminal
4. Válvulas de Segurança
5. Placa Negativa
6. Separador
7. Placa Positiva

Que podem ser observados na Figura 14 a seguir:

**FIGURA 14 – CONSTRUÇÃO DA BATERIA DE CHUMBO-ÀCIDO.**



Fonte: ROBOCORE, 2018.

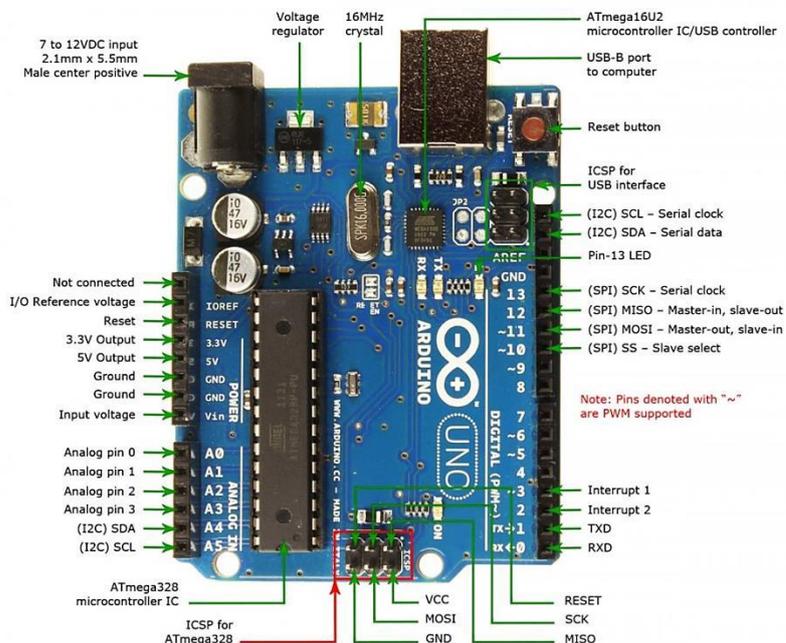
## 2.4 ARDUINO

O Arduino é uma plataforma de prototipagem eletrônica *open-source*, tanto seu *hardware* quanto seu *software* são simples e de fácil utilização. Sua IDE (*Integrated Development Environment*, em português Ambiente de Desenvolvimento Integrado), possibilita a criação do código com uma linguagem semelhante ao do C/C++, compilando o mesmo para a linguagem do *hardware*. (VIDADESILICIO, 2017).

O microcontrolador presente na versão UNO do Arduino é o *ATMEGA 328P-PU*, de 8 bits da família AVR, possui 14 pinos de entrada e saída (I/O) digitais e 6 pinos de entrada analógicas. Sua alimentação pode ser realizada pelo conector de

fonte externa, pela conexão USB ou pelo pino Vin, essas informações podem ser observadas na Figura 15. (EMBARCADOS, 2013).

FIGURA 15 – ARDUINO UNO.



Fonte: TEACHMEMICRO, 2017.

## 2.5 PROTOCOLO DE COMUNICAÇÃO I2C

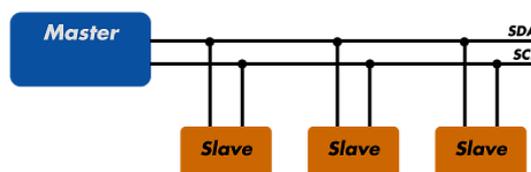
Protocolos de comunicação servem para padronizar o envio de dados entre dispositivos, sendo necessário um canal, um emissor e um receptor. Os dados podem ser enviados em pacotes, com identificação e valores em seus bits. (CRIARWEB, 2006).

O protocolo I2C significa “*Inter-Integrated Circuit*” e funciona utilizando o modelo *master-slave* (mestre e escravo): o *master* coordena as comunicações, enviando e consultando as informações dos *slaves*, os *slaves* aguardam o contato dos *masters* e lhe enviam dados solicitados. É possível mais de um *master* e mais de 100 *slaves* em uma única conexão. (ARDUINOBR, 2014).

Utilizando barramentos (*bus*), todos os dispositivos estão ligados aos mesmos fios, o Arduino possui as portas SDA (*Serial Data*) e SCL (*Serial Clock*) que são próprias para a comunicação I2C. O barramento SDA é bidirecional, ou seja, envia e recebe dados, e o barramento SCL, tem por função realizar a temporização

entre os dispositivos, a Figura 16 apresenta o barramento de dados. (ARDUINOBR, 2014).

**FIGURA 16– BARRAMENTO DE DADOS DO PROTOCOLO I2C.**



Fonte: ARDUINOBR, 2014.

Em relação à comunicação entre dispositivos, cada um possui um endereço de identificação. Como o endereço é composto por 7 bits, será possível fazer a multiplexação de até 127 dispositivos diferentes. Além disso, há a existência de um oitavo bit para dizer se a operação realizada pela I2C será de leitura ou escrita. O tratamento de particularidades de leitura/escrita (R/W), pode ser feito através da análise do *datasheet* do dispositivo. (ARDUINOBR, 2014).

## 2.6 DISPLAY LCD

*Displays* LCD são constituídos por células, um *display* de 16x2 tem 32 células, por exemplo, cada célula possui uma matriz de 8x5 *pixels*.

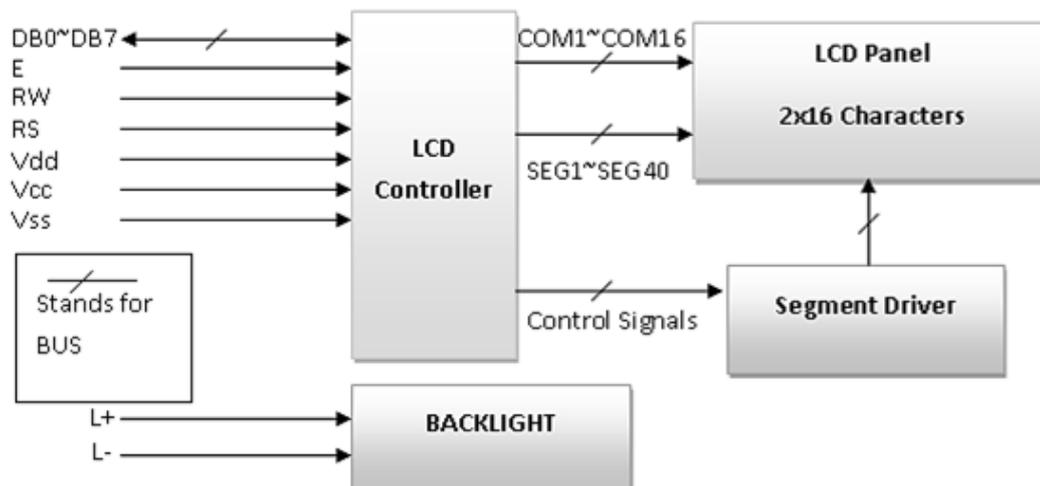
**QUADRO 1 – PINOS DO DISPLAY LCD.**

Pinos do display LCD		
Pino	Símbolo	Função
1	Vss	GND
2	Vcc	5V
3	Vdd	Ajuste de contraste
4	RS	Seleção de registro: 0 → Registo de instruções; 1 → Registo de dados.
5	R/W	Sinal leitura/escrita: 1→Leitura; 0→ Escrita.
6	E	Enable; Borda de descida
7	DB0	Barramento de dados bidirecional, a transferência de dados é realizada uma vez, através de DB0 a DB7, no caso de comprimento de dados de interface de 8 bits; e duas vezes, através de DB4 para DB7, no caso de comprimento de dados de interface é de 4 bits. Quatro bits <i>HIGH</i> primeiro, em seguida, quatro bits <i>LOW</i> .
8	DB1	
9	DB2	
10	DB3	
11	DB4	
12	DB5	
13	DB6	
14	DB7	
15	LED-(K)	<i>Backlight</i> LED terminal catodo
16	LED+(A)	<i>Backlight</i> LED terminal anodo

Fonte: Adaptado de CIRCUITSTODAY, 2017.

O módulo de *display* LCD 16x2 possui um bloco próprio de controle, painel LCD, *driver* dos segmentos e o *backlight*. Este diagrama esta na Figura 17.

FIGURA 17 – DIAGRAMA DE BLOCOS DO DISPLAY LCD 16X2:



Fonte: CIRCUITSTODAY, 2017.

### 3 PROJETO DA ESTAÇÃO AGROMETEOROLOGICA

#### 3.1 DESENVOLVIMENTO DO PROJETO

O desenvolvimento do projeto foi praticamente todo experimental, os sensores, *display*, *datalogger* e afins, foram montados e testados em *protoboard*, somente após o funcionamento adequado, foi realizado o projeto da placa de circuito impresso.

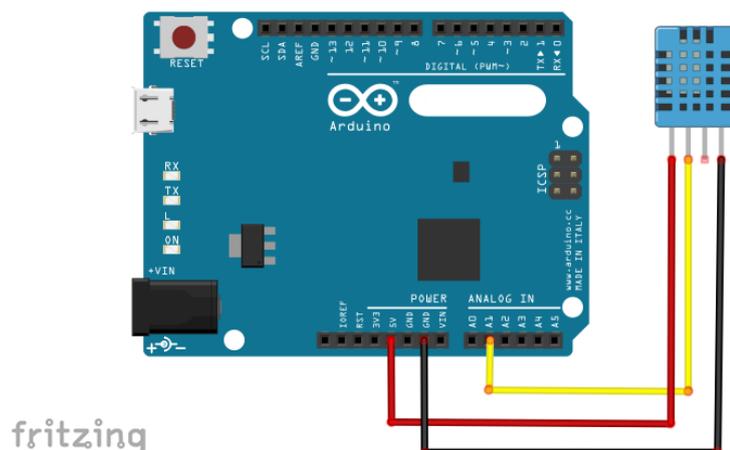
##### 3.1.1 CÓDIGO

O código foi desenvolvido com base na linguagem suportada pelo Arduino, sendo necessárias algumas bibliotecas para o funcionamento adequado dos componentes do circuito, optou-se pela divisão em funções para um melhor entendimento do mesmo, o fluxograma do código está no Apêndice 1 e o código no Apêndice 2.

##### 3.1.2 SENSOR DHT11

Para o sensor DHT11 foi utilizado à biblioteca *dht.h*. O sensor mede a umidade e temperatura do ar, o código de funcionamento esta na função “*dht11ar*” no Apêndice 2. Para colocar o sensor no abrigo de umidade foi utilizado cabo flexível de três vias, com 1 metro de comprimento, sem comprometer a medição. O pino de dados do sensor foi ligado a entrada analógica 1 (A1) do Arduino.

FIGURA 18 – CIRCUITO DE LIGAÇÃO DO DHT11.

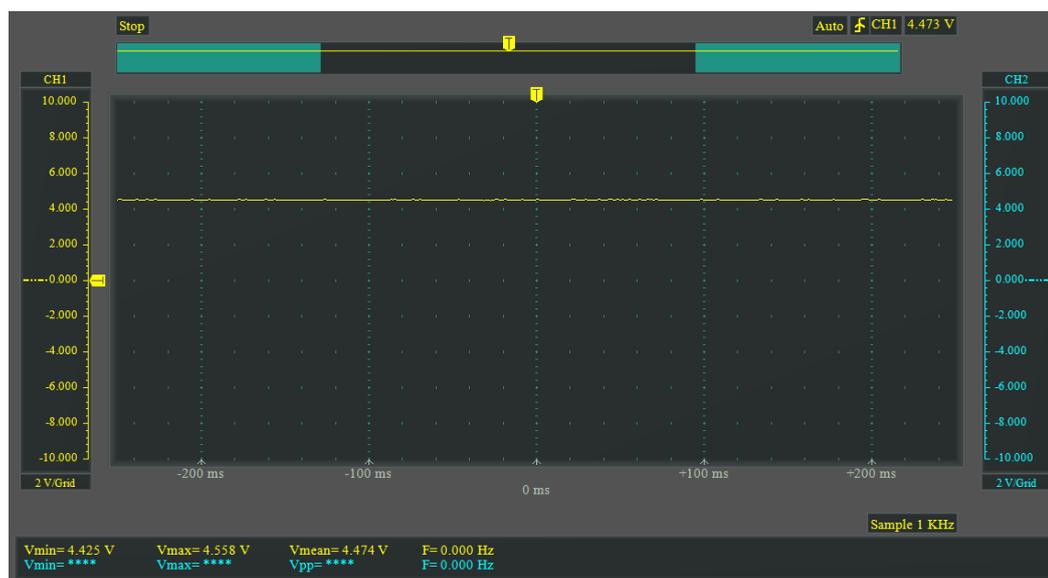


fritzing

Fonte: Autoria própria.

O sinal de saída do DHT11 utilizado foi observado com o auxílio de um osciloscópio USB e está mostrado na Figura 19 a seguir.

FIGURA 19 – SINAL DE DADOS DO DHT11.



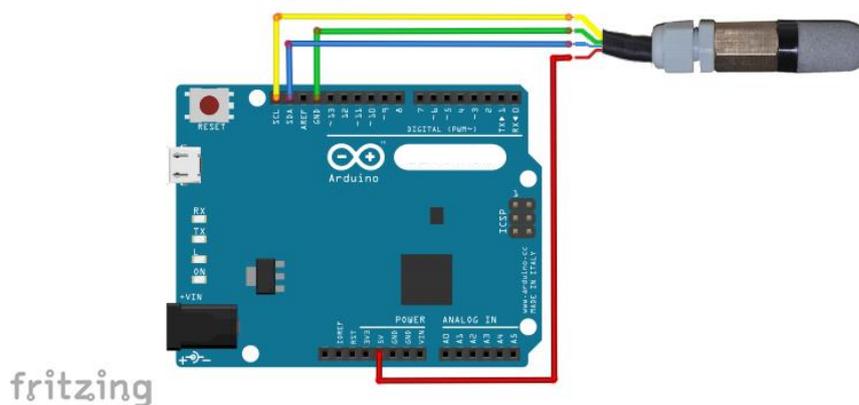
Fonte: Autoria própria.

### 3.1.3 SENSOR SHT20

Para o sensor SHT20 foi utilizado a biblioteca: *DFRobot\_SHT20.h*. O sensor mede a umidade e temperatura do solo e foi enterrado à 15 centímetros no solo, o código de funcionamento está na função “sht20solo” presente no Apêndice 2. Para alongar o cabo do sensor foi utilizado cabo flexível de quatro vias, com 1,5 metros

de comprimento, sem comprometer a medição. Os pinos SDA e SCL foram ligados às respectivas portas do Arduino, como na Figura 20.

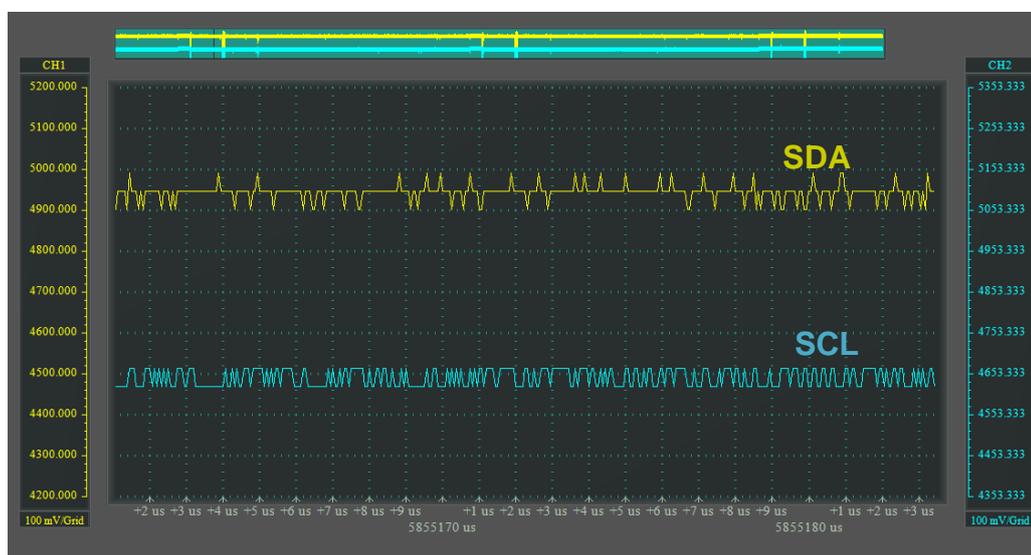
**FIGURA 20 – CIRCUITO DE LIGAÇÃO DO SHT20.**



Fonte: Autoria própria.

A partir de um osciloscópio USB, foi possível observar os sinais SDA e SCL de dados do SHT20. A Figura 21 a seguir apresenta os sinais.

**FIGURA 21 – SINAIS SDA E SCL DO SENSOR SHT20.**



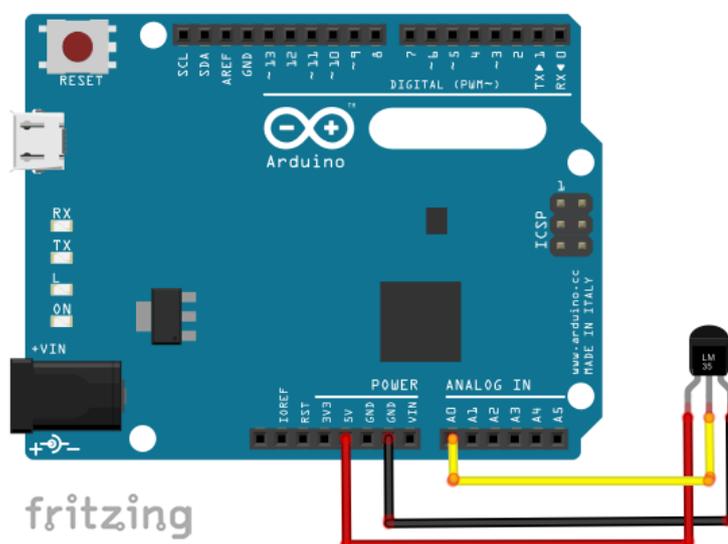
Fonte: Autoria própria.

### 3.1.4 LM35 COMO SENSOR DE TEMPERATURA DA BATERIA

Para o monitoramento da temperatura da bateria foi utilizado o sensor LM35, colocado na placa de circuito impresso próximo a bateria, poderia também ter sido colocado utilizando cabos, anexando-o diretamente a bateria.

O pino 2 do sensor foi ligado entrada analógica 0 (A0) do Arduino. O código de funcionamento está na função “temperaturabateria” no Apêndice 2, e o circuito de funcionamento está na Figura 22.

**FIGURA 22 – CIRCUITO DE FUNCIONAMENTO DO LM35.**

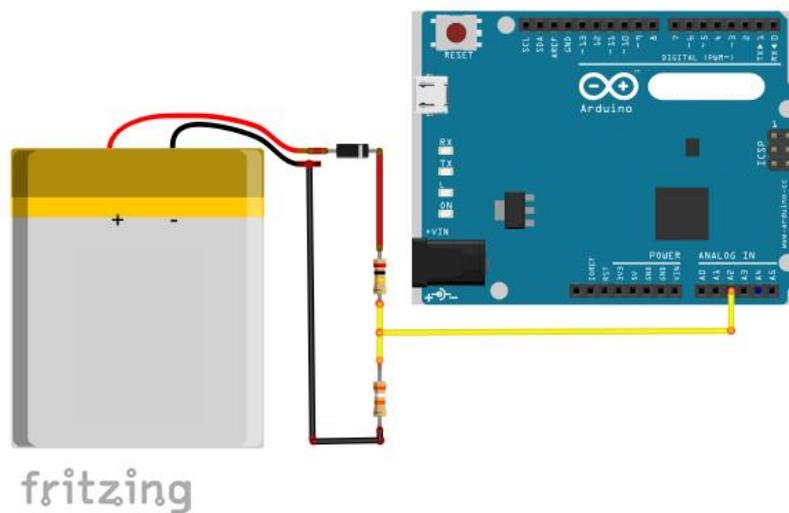


Fonte: Autoria própria.

### 3.1.5 MONITORAMENTO DA TENSÃO NA BATERIA

O monitoramento da bateria foi realizado utilizando um divisor de tensão resistivo e conectando-o a entrada analógica 2 (A2) do Arduino. O circuito é mostrado na Figura 23.

**FIGURA 23 – CIRCUITO DE MONITORAMENTO DA BATERIA.**



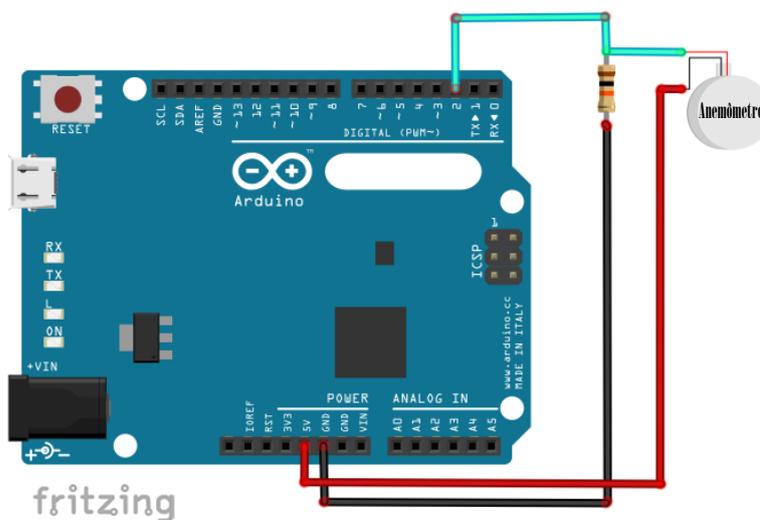
Fonte: Autoria própria.

O código que faz essa leitura está na função “tensaobateria” do Apêndice 2.

### 3.1.6 ANEMÔMETRO

Para o anemômetro foi utilizado o seguinte circuito da Figura 24.

**FIGURA 24 - CIRCUITO DE UTILIZAÇÃO DO ANEMÔMETRO.**



Fonte: Autoria própria.

O código do Arduino consiste em um contador de pulsos, que relaciona as rotações com o tempo, o código foi adaptado do disponibilizado pelo comerciante: USINAINFO (2018).

O código implementado está no Apêndice 2 e é composto pelas funções: “vento”, “windvelocity”, “RPMcalc”, “SpeedWind”, “addcount”.

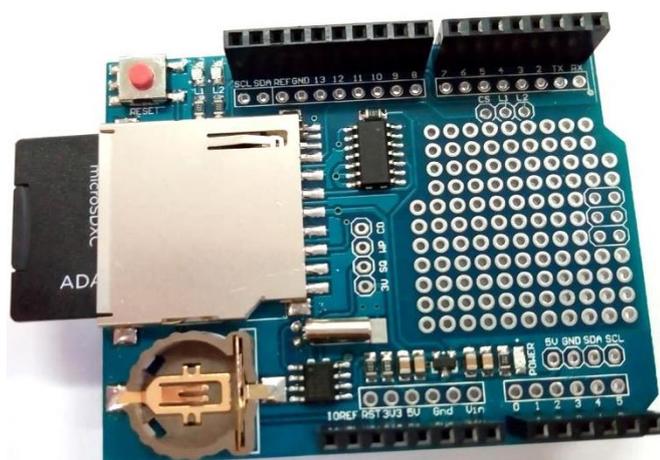
### 3.1.7 DATALOGGER

Para salvar os dados no cartão SD foi utilizado o *Logger Shield*, apresentado na figura 25, voltado para o uso em Arduino, sendo o mesmo compatível com vários modelos de Arduino, inclusive o UNO que foi utilizado. O *shield* está conectado aos pinos digitais 10 (D10), 11 (D11), 12 (D12) e 13 (D13), sendo o pino 10 de dados a serem salvos no cartão SD, e as analógicas 5 (A5) e 6 (A6). O esquemático do *shield logger* está no Apêndice 3.

Para utilização do mesmo foram utilizadas as seguintes bibliotecas:

- *SPI.h*: Serial Peripheral Interface
- *SD.h*: cartão SD
- *Wire.h*: protocolo I2C
- *RTClib.h*: RTC (Real Time Clock)

**FIGURA 25 – LOGGER SHIELD.**



Fonte: Autoria própria.

### 3.1.8 DISPLAY LCD

Para que usuário tenha acesso aos dados dos sensores, foi utilizado um *display* LCD 16x2. Foram realizadas as seguintes ligações conforme mostrado no Quadro 2.

**QUADRO 2 – CONEXÕES LCD 16X2.**

Conexões LCD 16x2		
Pino	Função	Ligação
1	Vss	GND
2	Vdd	VDD
3	V0	Resistor 1 K $\Omega$
4	RS	D9
5	RW	GND
6	E	D8
7	D0	Não conectado
8	D1	Não conectado
9	D2	Não conectado
10	D3	Não conectado
11	D4	D5
12	D5	D4
13	D6	D3
14	D7	D6
15	A	Chave (ligada ao VDD)
16	K	GND

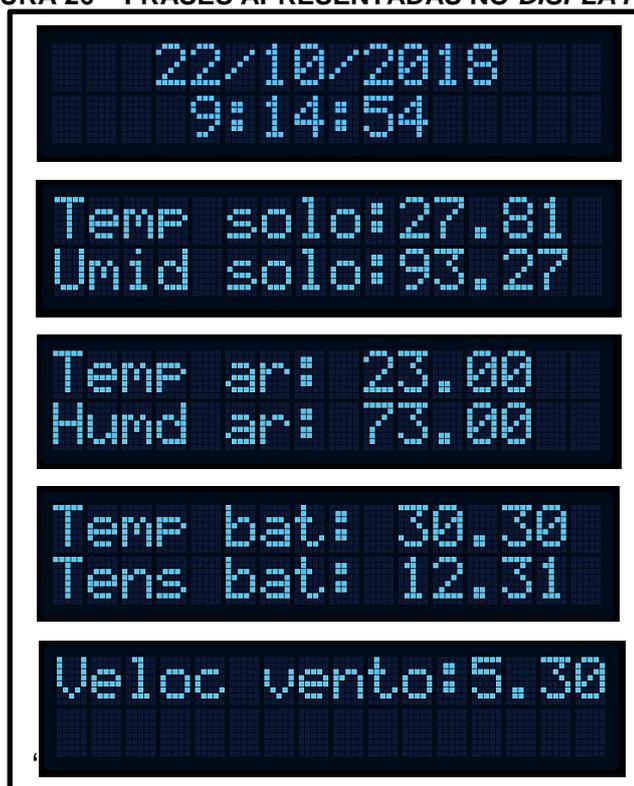
Fonte: Autoria própria.

Para utilizar o *display*, foi necessária a utilização da biblioteca: *LiquidCrystal.h*. Os códigos estão no Apêndice 2. Para mostrar os dados no *display* utilizam-se os seguintes comandos:

- *lcd.clear()*: limpa o *display*;
- *lcd.setCursor(valorx, valory)*: seta o início da apresentação dos dados;
- *lcd.print("Frase fixa:")*: frases identificando o sensor apresentado;
- *lcd.print(valorsensor)*: apresenta os dados do sensor no *display*.

As frases programadas no código estão na Figura 26 a seguir. Sendo as temperaturas em graus celsius, as umidades em porcentagem de umidade relativa, a tensão da bateria em volts e a velocidade do vento em quilômetros por hora.

FIGURA 26 – FRASES APRESENTADAS NO DISPLAY LCD.



Fonte: Aatoria própria.

### 3.1.9 SENSORES BPW34 E HS-1101

Devido à falta de equipamentos de medição da irradiação solar e umidade do ar, optou-se por não colocar os circuitos desenvolvidos para estes sensores no projeto final da estação. Seria necessário utilizar um higrômetro para medir a umidade e um piranômetro para medir a irradiação solar, possibilitando fazer a calibração dos sensores adequadamente.

### 3.1.10 ESQUEMÁTICO E PLACA DE CIRCUITO IMPRESSO

Para desenvolver o esquemático foi utilizado o *software* Eagle da Autodesk®, o *software* possui uma versão completa gratuita para estudantes. O esquemático foi desenvolvido com base em técnicas e teorias de circuitos eletrônicos.

O esquemático completo está no Apêndice 4, para as ligações entre os componentes foram utilizados *labels*. Foram adicionados alguns elementos como chaves liga/desliga, LED de indicação de funcionamento, fontes de 9 V e 5 V com os reguladores da família 78XX.

A ausência de um painel solar adequado impossibilitou projetar o sistema fotovoltaico para o carregamento da bateria, o equipamento possui um custo muito elevado, tendo sido impossível adquirir o mesmo para a instalação na estação.

A *board* foi criada utilizando o mesmo *software*, sendo desenvolvida uma placa de 15x13 cm de tamanho e com face única (*single layer*).

### 3.2 CONSTRUÇÃO FÍSICA

Para proteção do sensor de umidade foi utilizado um abrigo que permite a passagem de ar, porém sem deixar a chuva atingir o sensor. O anemômetro adquirido é próprio para instalações externas, sendo blindado contra a entrada de água. Para proteção dos circuitos e da bateria, foi utilizada uma caixa protetora, também blindada contra a entrada de água.

Os equipamentos foram acoplados em um cano de alumínio, que era um suporte de antena, reaproveitando esse material. O próprio suporte possui uma extremidade afiada, facilitando à instalação no solo e, conseqüentemente, em plantações, necessitando apenas cavar um buraco para instalar o cano. Sua altura ficou reduzida devido ao tamanho do cano, que poderia ser mais alto. O sensor SHT20 necessita apenas ser enterrado de 10 a 20 cm no solo. A Figura 27 apresenta a instalação do projeto.

**FIGURA 27 – INSTALAÇÃO FÍSICA DO PROJETO.**



Fonte: Autoria própria.

## 4 RESULTADOS E DISCUSSÕES

### 4.1 COLETA DE DADOS

Os dados foram coletados entre os dias 16 e 17 de outubro de 2018, na cidade de Cianorte, situada no interior do estado do Paraná, entre 10h07 às 3h30 respectivamente, com cada medição sendo realizada em um período de 18 segundos. Durante essa faixa de tempo foram coletados 3442 registros de dados de cada sensor.

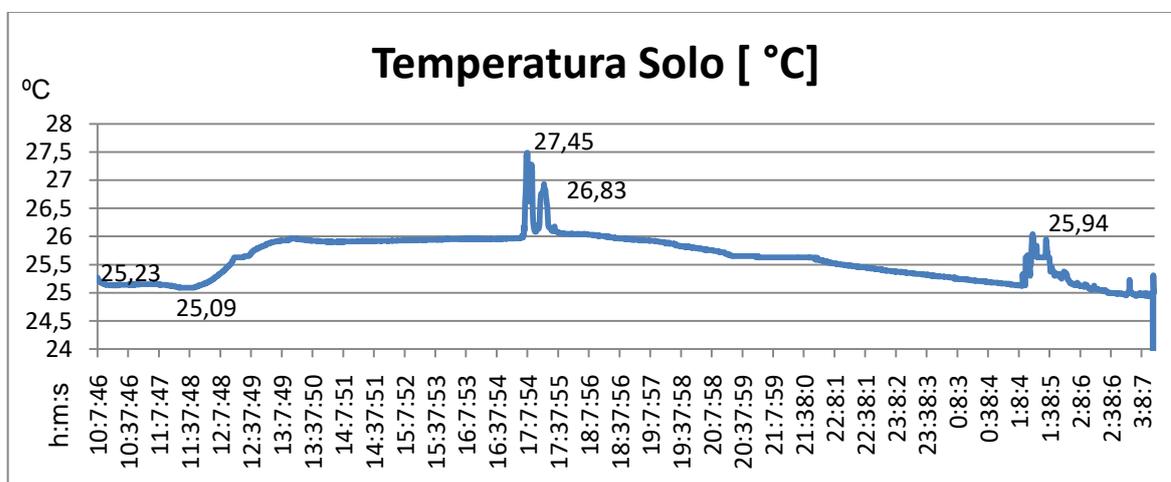
Durante os dias de funcionamento, as condições climáticas foram instáveis, intercalando momentos de chuva e sol, que se refletiu nas medições dos sensores, principalmente de temperatura e umidade.

O arquivo .CSV (*comma-separated values*) criado pelo *datalogger shield* possui apenas 234 kB, muito abaixo dos 8 GB de capacidade disponível do cartão SD. Parte dos dados do arquivo está no Apêndice 6. Para a manipulação dos dados no *software* utilizado para plotar gráficos, foi necessária a substituição do “.” por “,” nos valores dos sensores.

#### 4.1.1 SENSORES DE UMIDADE E TEMPERATURA

Os dados dos sensores de temperatura e umidade estão mostrados nas Figuras 28 à 34.

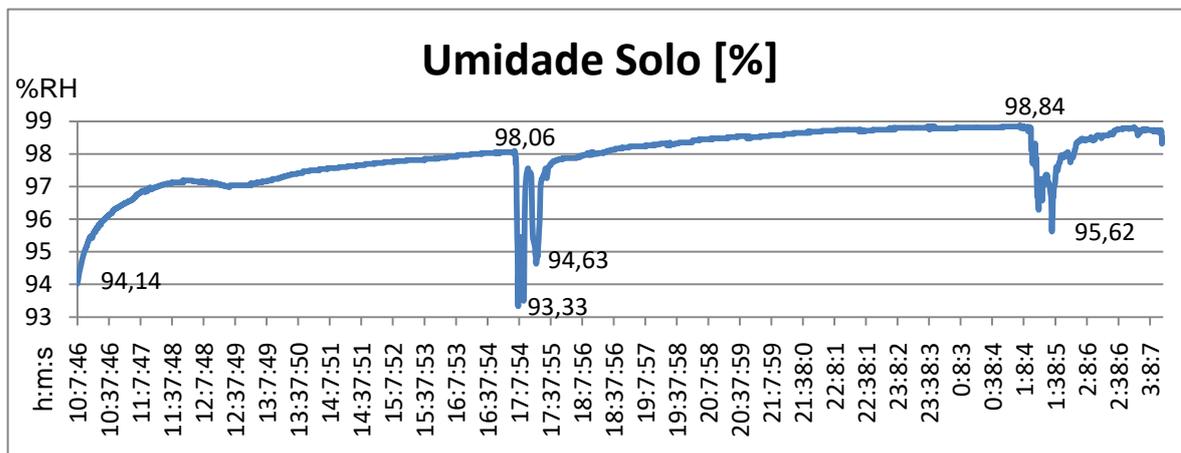
FIGURA 28 – TEMPERATURA DO SOLO.



Fonte: Autoria própria.

A temperatura do solo não apresentou grandes mudanças, sendo a mínima 24 °C e a máxima 27,45 °C, o sensor SHT20 operou de maneira bem linear e possui uma boa precisão.

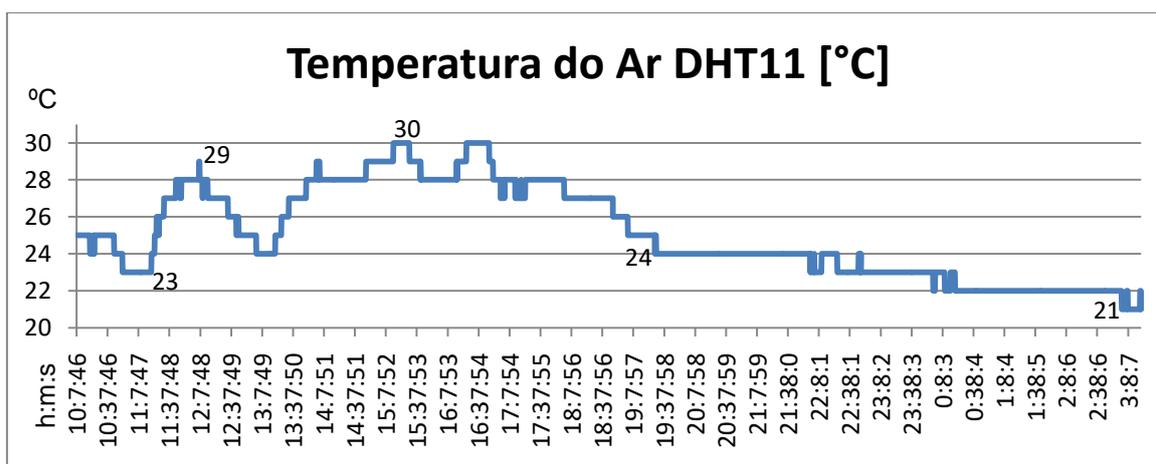
FIGURA 29 – UMIDADE DO SOLO.



Fonte: Autoria própria.

A umidade teve uma variação de 93,33 a 98,84 %RH, pode-se comparar a figura 28 e 29, nota-se que quando a temperatura aumentou, a umidade diminuiu, sendo um comportamento coerente. O sensor também operou de maneira linear e precisa medindo a umidade.

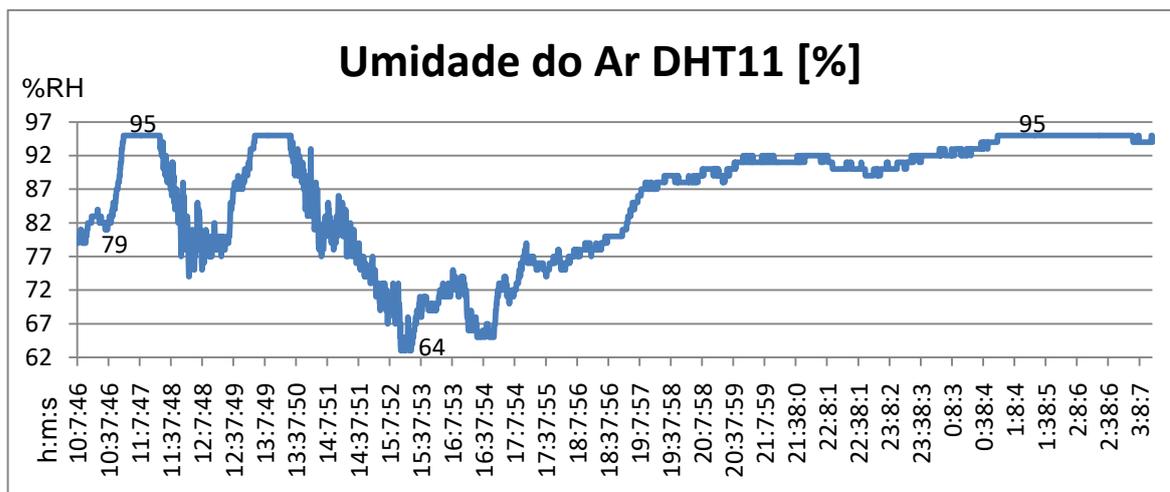
FIGURA 30 – TEMPERATURA DO AR.



Fonte: Autoria própria.

A temperatura do ar variou entre 21 e 30 °C. O sensor DHT11 não possui uma boa linearidade e precisão decimal.

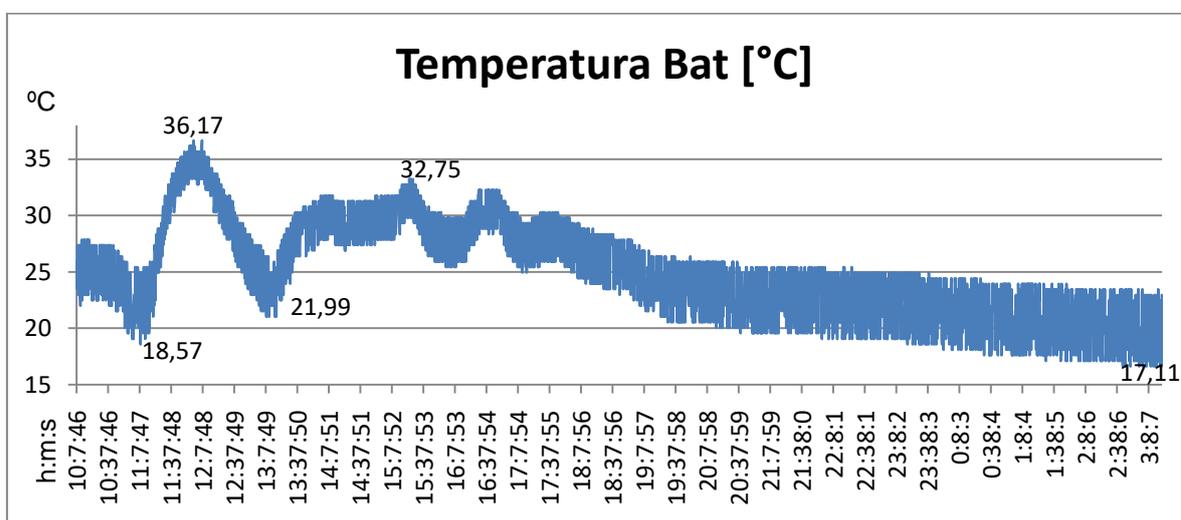
FIGURA 31 – UMIDADE DO AR.



Fonte: Autoria própria.

A umidade do ar variou entre 64 e 95 %RH. Nota-se também em comparação entre as figuras 30 e 31 o comportamento semelhante, em que conforme a temperatura aumenta a umidade diminui. O gráfico dos dados ficou mais ruidoso que da temperatura.

FIGURA 32 – TEMPERATURA DA BATERIA.



Fonte: Autoria própria.

A temperatura da bateria variou entre 17,11 e 36,17 °C, o sensor LM35 possui uma boa precisão e linearidade, porém o gráfico se mostrou ruidoso, devido ao grande número de amostras com variações entre si.

Mesmo com cada sensor instalado em um local diferente, foi possível analisar algumas características dos mesmos. Os sensores SHT20 e LM35 possuem um comportamento mais linear e uma precisão muito maior que do sensor DHT11. Devido às condições climáticas do dia, ocorreram grandes variações entre as medições.

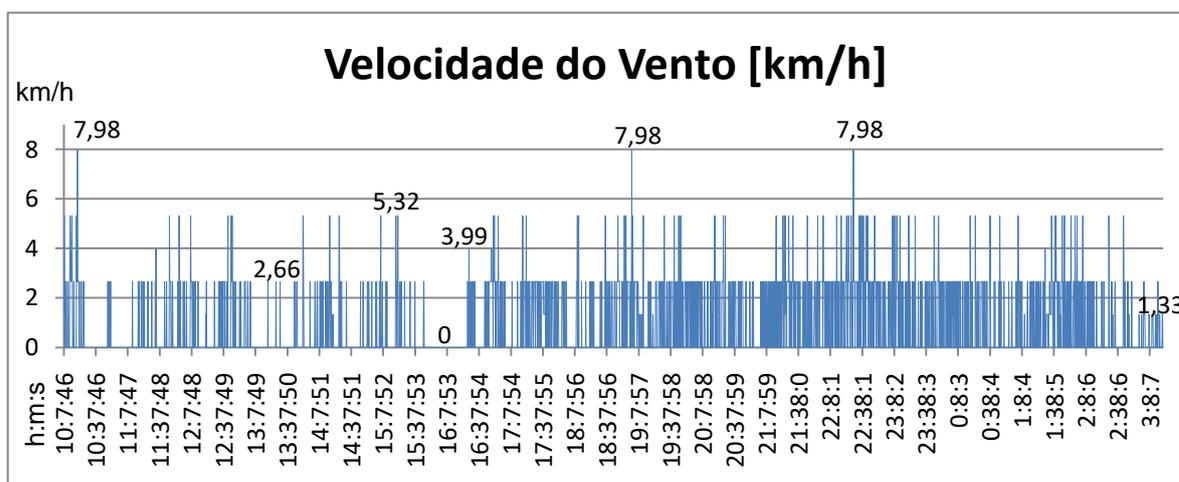
Analisando individualmente os sensores, fica evidente a superioridade em qualidade das medições do sensor SHT20 em comparação com os demais, seguido do LM35 e, por último o DHT11. Apesar da precisão o sensor LM35 apresentou um ruído em suas medições, que poderia ser resolvido realizando uma media das amostras no *software*. Devido à baixa precisão do sensor DHT11, chega-se a conclusão de uma substituição futura deste sensor por um mais adequado ao projeto, como por exemplo, o DHT22.

Segundo o manual do fabricante, a bateria estacionária pode ser utilizada em ambientes de  $-15^{\circ}\text{C}$  a  $50^{\circ}\text{C}$ , sendo a maior temperatura atingida no dia  $35^{\circ}\text{C}$  e a menor  $17,11^{\circ}\text{C}$ . Seria possível projetar um sistema de segurança para a bateria, que desativasse a mesma, caso a temperatura atingisse o limiar de segurança.

#### 4.1.2 ANEMÔMETRO

A Figura 33 apresenta os dados coletados da velocidade do ar durante as horas de medição.

FIGURA 33 – VELOCIDADE DO AR.



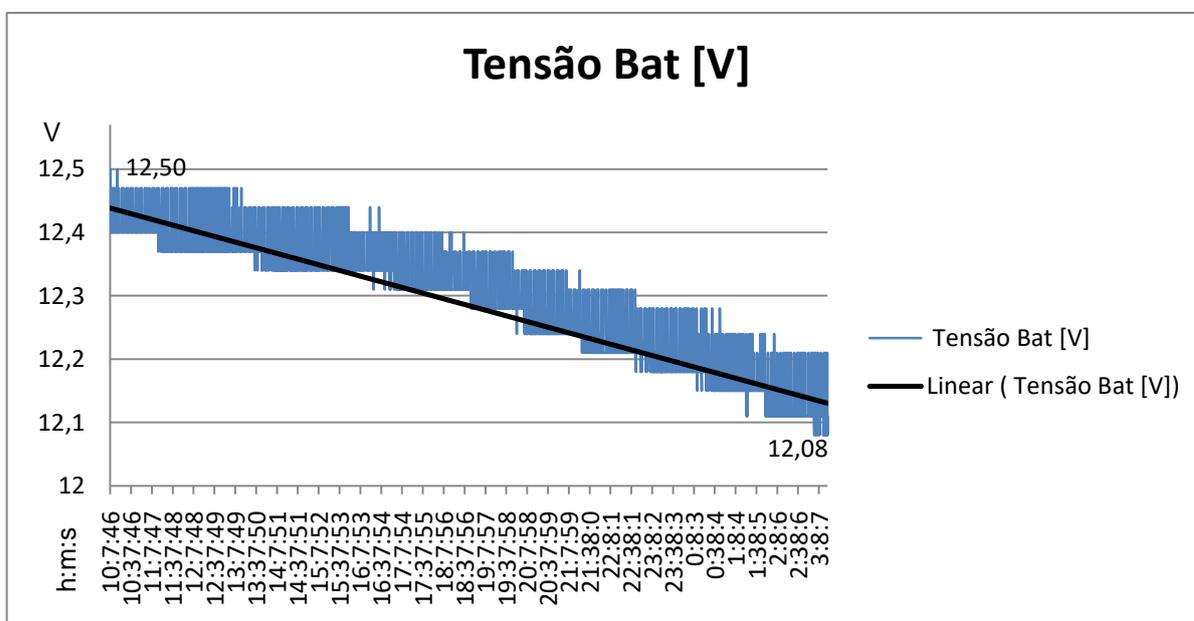
Fonte: Autoria própria.

Durante o funcionamento do projeto, não ocorreram rajadas fortes de ar no local, sendo o pico de 7,98 km/h e chegando a momentos sem nenhum movimento do ar. O sensor apresenta um comportamento pouco linear e apesar de apresentar as casas decimais, o sensor não é muito preciso.

#### 4.1.3 TENSÃO DA BATERIA

A Figura 34 mostra o gráfico do sensoriamento de tensão da bateria utilizada.

FIGURA 34 – TENSÃO DA BATERIA.



Fonte: Autoria própria.

O gráfico se mostrou com muito ruído apesar do comportamento linear, esse problema poderia ser resolvido calculando a média dos valores aferidos.

A princípio, a bateria fornecia uma tensão de 12,5 V, como está no Apêndice 6, esse valor também foi aferido com um multímetro. Já ao final da obtenção dos dados, esta apresentava uma tensão de 12,08 V. Segundo o fabricante, a bateria é considerada descarregada ao atingir 10,5 V.

Levando em consideração que em aproximadamente 17 horas a bateria teve uma queda de 3,36% em sua tensão, e para ser considerada descarregada ela precisa ter uma queda de 16%, realizando uma simples regra de três, estima-se uma

autonomia de aproximadamente 81 horas da bateria, o que daria um pouco mais de três dias até o limite segundo o fabricante.

O regulador 7809 utilizado necessita de uma tensão de entrada de pelo menos 11 V para funcionar adequadamente, ou seja, quando a bateria descarregasse o Arduino também pararia de funcionar.

Fica evidente, então, a necessidade da implementação de um sistema fotovoltaico para recarregar a bateria, de modo que a demanda, aliada a autonomia requerida pelo projeto, seja atendida. Pode-se também ligar o circuito diretamente a uma fonte de tensão contínua, de 12 V ou mais, porém essa solução é pouco eficaz quando se leva em conta o tipo de local que o projeto pode ser instalado, plantações em propriedades rurais, que costumam serem locais remotos.

## 5 CONSIDERAÇÕES FINAIS

Os resultados obtidos atestaram o funcionamento do projeto, que monitorou os dados dos sensores, armazenou esses dados no cartão SD e os apresentou no *display* LCD. A falta de equipamentos de medição e calibração, e de um painel solar adequado acabou impactando nos circuitos a serem desenvolvidos. Os sensores operaram de maneira esperada, e foi possível analisar de maneira experimental a sua precisão.

O Arduino, o *logger shield* e a bateria mantiveram seu perfeito funcionamento. Nota-se então a necessidade de uma melhoria nos processos de fabricação da placa ou optar por uma industrial.

Devido ao grande volume de dados, que acabam atrapalhando na hora da análise, seria interessante obter os dados em um intervalo de tempo maior, diminuindo as amostras, como na maioria das estações automáticas.

Para trabalhos futuros poderiam ser realizados alguns aprimoramentos, como o desenvolvimento dos sensores de irradiação e pluviosidade com escoamento automático, pode-se modificar o microcontrolador para um mais robusto, e projetar um sistema de comunicação sem fio, para a transmissão dos dados em tempo real.

## REFERÊNCIAS

AGRO SMART. **Estação meteorológica: como funciona e sua importância na agricultura**, 2014. Disponível em: <<https://agrosmart.com.br/blog/meteorologia/estacao-meteorologica-funciona-importancia-agricultura/>>. Acesso em: 01 Maio 2018.

ANALOGICTIPS. **Current to voltage conversion: the role of transimpedance amplifiers**, 2016. Disponível em: <<https://www.analogictips.com/faq-converting-current-to-voltage-the-role-of-transimpedance-amplifiers/>>. Acesso em: 01 outubro 2018.

ARDUINOBR. **I2C – Protocolo de Comunicação**, 2014. Disponível em: <<http://www.arduino.br/arduino/i2c-protocolo-de-comunicacao/>>. Acesso em: 01 outubro 2018.

BLOGUSINAINFO. **ANEMÔMETRO ARDUINO – UM SENSOR DE VENTO PARA ESTAÇÃO METEOROLÓGICA**, 2017. Disponível em: <<http://blog.usinainfo.com.br/anemometro-arduino-um-sensor-de-vento-para-estacao-meteorologica/>>. Acesso em: 01 Maio 2018.

CITISYSTEMS. **Sensor de Temperatura: Encontre o Tipo Ideal para sua Aplicação**, 2018. Disponível em: <<https://www.citisystems.com.br/sensor-de-temperatura/>>. Acesso em: 28 Outubro 2018.

CRIARWEB. **Protocolos de comunicações**, 2006. Disponível em: <<http://www.criarweb.com/artigos/533.php/>>. Acesso em: 28 Outubro 2018.

CIRCUITSTODAY. **A Note on Character LCD Displays**, 2006. Disponível em: <<http://www.circuitstoday.com/a-note-on-character-lcd-displays/>>. Acesso em: 28 Outubro 2018.

ECO AGRO. **O Agronegócio no Brasil**, 2014. Disponível em:  
<<http://www.ecoagro.agr.br/agronegocio-brasil/>>. Acesso em: 01 Maio 2018.

EMBARCADOS. **Arduino UNO**, 2013. Disponível em:  
<<https://www.embarcados.com.br/arduino-uno/>>. Acesso em: 20 Outubro 2018.

INMET, Instituto Nacional de Meteorologia. **Rede de Estações Meteorológicas Automáticas do INMET**, 2011. Disponível em:  
<[http://www.inmet.gov.br/portal/css/content/topo\\_iframe/pdf/Nota\\_Tecnica-Rede\\_estacoes\\_INMET.pdf](http://www.inmet.gov.br/portal/css/content/topo_iframe/pdf/Nota_Tecnica-Rede_estacoes_INMET.pdf)>. Acesso em: 15 Maio 2018.

HUMIREL. **Datasheet: HS 1100 / HS 1101 RELATIVE HUMIDITY SENSOR**, 2002. Disponível em: <<https://www.parallax.com/sites/default/files/downloads/27920-Humidity-Sensor-Datasheet.pdf>>. Acesso em: 20 Maio 2018.

MOUSER. **DHT11 Humidity & Temperature Sensor**, 2017. Disponível em:  
<<https://www.mouser.com/ds/2/758/DHT11-Technical-Data-Sheet-Translated-Version-1143054.pdf>>. Acesso em: 28 Outubro 2018.

OCFREAKS. **Basics of Interfacing DHT11/DHT22 Humidity and Temperature Sensor with MCU**, 2017. Disponível em: <<http://www.ocfreaks.com/basics-interfacing-dht11-dht22-humidity-temperature-sensor-mcu/>>. Acesso em: 28 Outubro 2018.

PARALLAX. **HS1101 Relative Humidity Sensor**, 2009. Disponível em:  
<<https://www.parallax.com/sites/default/files/downloads/27920-Humidity-Sensor-Documentation-S1101-v1.0.pdf?fbclid=IwAR08BMqY9PYXpM51-kBYw820EUhdN71-JWhM96KLwNLWxzEp5T0MTSNsGMc>>. Acesso em: 28 Outubro 2018.

PORTALSAOFRANCISCO. **Anemômetro**, 2018. Disponível em:  
<<https://www.portalsaofrancisco.com.br/fisica/anemometro>>. Acesso em: 28 Outubro 2018.

REDIMAX. **O que é uma bateria estacionária?**, 2017. Disponível em: <<https://redimax.com.br/dicas/o-que-e-uma-bateria-estacionaria/>>. Acesso em: 28 Outubro 2018.

ROBOCORE. **Manual Técnico Bateria Chumbo-Ácida Selada Regulada por Válvula**, 2018. Disponível em: <<https://www.robocore.net/upload/ManualTecnicoBateriaUnipower.pdf/>>. Acesso em: 28 Outubro 2018.

SEIDEL, Álysson Raniere. **Instrumentação Aplicada**, 2011. Disponível em: <[http://estudio01.proj.ufsm.br/cadernos\\_automacao/setima\\_etapa/instrumentacao\\_aplicada\\_2012.pdf](http://estudio01.proj.ufsm.br/cadernos_automacao/setima_etapa/instrumentacao_aplicada_2012.pdf)>. Acesso em: 11 Maio 2018.

SENTELHAS, P. C. ; PEREIRA, A. R. ; ANGELOCCI, L. R. . **Meteorologia Agrícola**. 1998 (APOSTILA DFM/ESALQ/USP, 136p.

TEACHMEMICRO. **What is Arduino?**, 2017. Disponível em: <<https://www.teachmemicro.com/arduino-tutorials/what-is-arduino/>>. Acesso em: 20 Outubro 2018.

TEXAS, instruments. **Datasheet: LM35 Precision Centigrade Temperature Sensors**, 2017. Disponível em: <<http://www.ti.com/lit/ds/symlink/lm35.pdf>>. Acesso em: 20 Maio 2018.

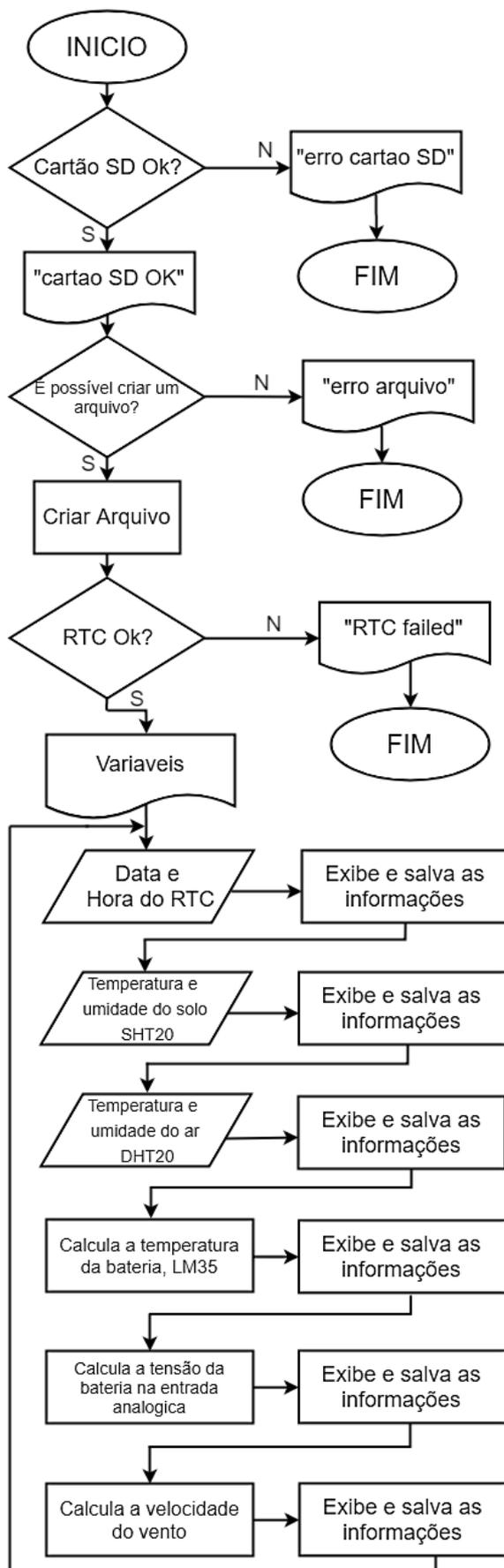
THOMAZINI, Daniel; ALBUQUERQUE, Pedro U. B. de. **Sensores industriais: fundamentos e aplicações**. 8. ed. São Paulo: Érica. 2011. 224 p ISBN 9788536500713.

USINAINFO. **Sensor de Umidade e Temperatura SHT20 com Sonda À Prova d'Água - KLDZ88**, 2018. Disponível em: <<https://www.usinainfo.com.br/sensor-de-temperatura-arduino/sensor-de-umidade-e-temperatura-sht20-com-sonda-a-prova-d-agua-kldz88-5233.html>>. Acesso em: 20 Maio 2018.

VIDADESILICIO. **O que é Arduino e como funciona?**, 2017. Disponível em: <<https://portal.vidadesilicio.com.br/o-que-e-arduino-e-como-funciona/>>. Acesso em: 20 Outubro 2018.

VISHAY SEMICONDUCTOR. **Datasheet: BPW34, BPW34S Silicon PIN Photodiode**, 2017. Disponível em: <<https://www.vishay.com/docs/81521/bpw34.pdf>>. Acesso em: 20 Maio 2018.

## APÊNDICE 1 – FLUXOGRAMA DO CODIGO



## APÊNDICE 2 – CÓDIGO DE ARDUINO DA ESTAÇÃO AGROMETEOROLOGICA

```

//Biblioteca SPI (Serial Peripheral Interface)
#include <SPI.h>
//Biblioteca SD (Cartao SD)
#include <SD.h>
//Biblioteca Wire (I2C)
#include <Wire.h>
//Biblioteca RTCLib (RTC)
#include "RTCLib.h"
//Biblioteca DFRobot_SHT20 (SHT20)
#include "DFRobot_SHT20.h"
//Biblioteca dht (DHT11)
#include <dht.h>
//Biblioteca LiquidCrystal (LCD)
#include <LiquidCrystal.h>

//Pino DATA do Sensor ligado na porta Analogica A1
#define dht_dpin A1
//Inicializa o sensor DHT11
dht DHT;

//SHT20 ligado aos pinos SCL e SDA
DFRobot_SHT20 sht20;

//Pinos do LCD
LiquidCrystal lcd(9, 8, 5, 4, 3, 6);

// LM35 ligado a analogica A0
const int LM35 = A0;
// Variavel de temperatura da bateria
float tempbat;

const int TB = A2;
// Variavel de temperatura da bateria
float tensbat;

int a = 0;

//define pino do sensor de vento
# define Hall sensor 2
// Numero pi
const float pi = 3.14159265;
// Tempo de medida(miliseconds)
int period = 5000;
// Tempo entre as amostras (miliseconds)
int delaytime = 2000;
// Raio do anemometro(mm)
int radius = 147;
//Variaveis

```

```

// Numero de amostra
unsigned int Sample = 0; // Sample number
// Contador magnetico do sensor
unsigned int counter = 0;
// Rotações por minuto
unsigned int RPM = 0;
// Wind speed (m/s) e (km/h)
float speedwind = 0;
float windspeed = 0;

// A simple data logger for the Arduino analog pins
// mills between entries (reduce to take more/faster data)
#define LOG_INTERVAL 1000
// mills between calls to flush() - to write data to the card
#define SYNC_INTERVAL 1000
// time of last sync()
uint32_t syncTime = 0;
// echo data to serial port

// Wait for serial input in setup()
#define WAIT_TO_START 0

// RTC (Real Time Clock)
RTC_DS1307 RTC;

// Define digital 10 como saída para o cartão SD
const int chipSelect = 10;

File logfile;

void error(char *str)
{
  Serial.print("error: ");
  Serial.println(str);
  lcd.setCursor(1, 0);
  lcd.println("erro cartao SD "); //Erro no cartão SD
  while(1);
}

void setup(void)
{
  Serial.begin(9600);
  Serial.println();
  sht20.initsHT20(); // Init SHT20 Sensor
  delay(100);
  sht20.checkSHT20();

  pinMode(2, INPUT); //Pino 2 digital, sensor de vento
  digitalWrite(2, HIGH); //internall pull-up active
  lcd.begin(16, 2); //inicia o LCD e define como um display de 16x2

```

```

#if WAIT_TO_START
  Serial.println("Type any character to start");
  while (!Serial.available());
#endif //WAIT_TO_START

// initialize the SD card
Serial.print("Initializing SD card...");
// make sure that the default chip select pin is set to
// output, even if you don't use it:
pinMode(10, OUTPUT);

// see if the card is present and can be initialized:
if (!SD.begin(chipSelect)) {
  error("Card failed, or not present");
  lcd.println("Erro cartao SD"); //cartão SD
}
Serial.println("card initialized.");
lcd.println("cartao SD Ok   "); //cartão SD
delay(2000);

// create a new file
char filename[] = "LOGGER00.CSV (COMMA-SEPARATED VALUES)";
for (uint8_t i = 0; i < 100; i++) {
  filename[6] = i/10 + '0';
  filename[7] = i%10 + '0';
  if (!SD.exists(filename)) {
    // only open a new file if it doesn't exist
    logfile = SD.open(filename, FILE_WRITE);
    break; // leave the loop!
  }
}

if (!logfile) {
  error("couldnt create file");
  lcd.println("erro arquivo  ");
}

Serial.print("Logging to: ");
Serial.println(filename);

// connect to RTC
Wire.begin();
if (!RTC.begin()) {
  logfile.println("RTC failed");

  Serial.println("RTC failed");
}

logfile.println("Data;Hora;Temperatura Solo [C];Umidade Solo [%];Temperatura do Ar DHT11
[C]; Umidade do Ar DHT11 [%]; Temperatura Bat [C]; Tensão Bat [V]; Velocidade do Vento
[Km/h]");

```

```

    Serial.println("Data;Hora;Temperatura Solo [C];Umidade Solo [%];Temperatura do Ar DHT11 [C];
Umidade do Ar DHT11 [%]; Temperatura Bat [C]; Tensão Bat [V]; Velocidade do Vento [Km/h]");

}

void loop() {
    // put your main code here, to run repeatedly:
    horario();
    sht20solo();
    dht11lar();
    temperaturabateria();
    tensaobateria();
    vento();

    logfile.println();
    Serial.println();
    // Now we write data to disk! Don't sync too often - requires 2048 bytes of I/O to SD card
    // which uses a bunch of power and takes time
    if ((millis() - syncTime) < SYNC_INTERVAL) return;
    syncTime = millis();
    logfile.flush();
}

void horario(){
    DateTime now;
    delay((LOG_INTERVAL -1) - (millis() % LOG_INTERVAL));
    now = RTC.now();
    float aux = now.second();

    logfile.print(now.day(), DEC);
    logfile.print("/");
    logfile.print(now.month(), DEC);
    logfile.print("/");
    logfile.print(now.year(), DEC);
    logfile.print(" ");

    lcd.clear();
    lcd.setCursor(3, 0);
    lcd.print(now.day(), DEC);
    lcd.print("/");
    lcd.print(now.month(), DEC);
    lcd.print("/");
    lcd.print(now.year(), DEC);

    Serial.print(now.day(), DEC);
    Serial.print("/");
    Serial.print(now.month(), DEC);
    Serial.print("/");
    Serial.print(now.year(), DEC);
}

```

```

Serial.print(" ");

logfile.print(now.hour(), DEC);
logfile.print(":");
logfile.print(now.minute(), DEC);
logfile.print(":");
logfile.print(now.second(), DEC);

lcd.setCursor(4, 1);
lcd.print(now.hour(), DEC);
lcd.print(":");
lcd.print(now.minute(), DEC);
lcd.print(":");
lcd.print(now.second(), DEC);

Serial.print(now.hour(), DEC);
Serial.print(":");
Serial.print(now.minute(), DEC);
Serial.print(":");
Serial.print(now.second(), DEC);
delay(2000);
}

void sht20solo(){
  float humd = sht20.readHumidity();           // Read Humidity
  float temp = sht20.readTemperature();       // Read Temperature

  logfile.print(" ");
  logfile.print(temp);

  lcd.clear();
  lcd.setCursor(0,0);
  lcd.println("Temp solo:");
  lcd.setCursor(10,0);
  lcd.print(temp);

  Serial.print(" ");
  Serial.print(temp);

  logfile.print(" ");
  logfile.print(humd);

  lcd.setCursor(0,1);
  lcd.println("Umid solo:");
  lcd.setCursor(10,1);
  lcd.print(humd);

  Serial.print(" ");
  Serial.print(humd);
  delay(2000);
}

```

```

void dht11ar(){
  DHT.read11(dht_dp1n); //Lê as informações do sensor
  float humiddht11 = DHT.humidity;
  float tempdht11 = DHT.temperature;

  logfile.print("; ");
  logfile.print(tempdht11);

  lcd.clear();
  lcd.setCursor(0,0);
  lcd.println("Temp ar:");
  lcd.setCursor(8,0);
  lcd.print(tempdht11);

  Serial.print("; ");
  Serial.print(tempdht11);

  logfile.print("; ");
  logfile.print(humiddht11);

  lcd.setCursor(0, 1);
  lcd.println("Umid ar:");
  lcd.setCursor(8,1);
  lcd.print(humiddht11);

  Serial.print("; ");
  Serial.print(humiddht11);
  delay(2000);
}

void temperaturabateria(){
  tempbat =(float(analogRead(LM35)) *5/(1023))/0.01;

  logfile.print("; ");
  logfile.print(tempbat);
  lcd.clear();
  lcd.setCursor(0,0);
  lcd.println("Temp bat:");
  lcd.setCursor(9,0);
  lcd.print(tempbat);

  delay(2000);

  Serial.print("; ");
  Serial.print(tempbat);
}

void tensaobateria(){
  tensbat =(float(analogRead(TB)) *5/(1023))/(0.01*15.13);
  logfile.print("; ");
}

```

```

logfile.print(tensbat);

lcd.setCursor(0,1);
lcd.println("Tens bat:");
lcd.setCursor(9,1);
lcd.print(tensbat);

delay(2000);

Serial.print("; ");
Serial.print(tensbat);
}

void vento(){
  Sample++;
  windvelocity();
  RPMcalc();
  SpeedWind();

  logfile.print("; ");
  logfile.print(speedwind);

  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.println("Veloc vento:");
  lcd.setCursor(12,0);
  lcd.print(speedwind);

  Serial.print("; ");
  Serial.print(speedwind);
  delay(2000);
}

void windvelocity(){
  speedwind = 0;
  windspeed = 0;
  counter = 0;
  attachInterrupt(0, addcount, RISING);
  unsigned long millis();
  long startTime = millis();
  while(millis() < startTime + period) {
  }
}

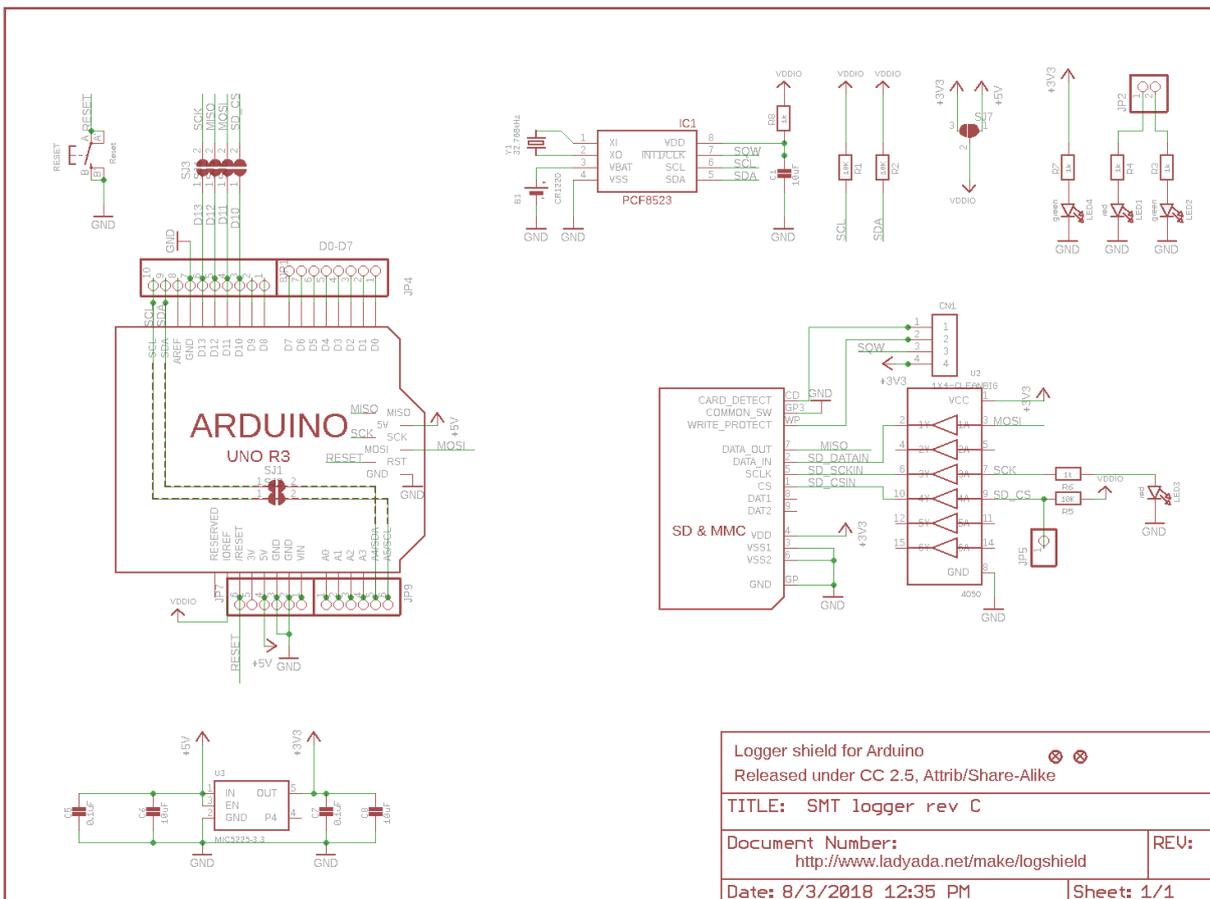
void RPMcalc(){
  RPM=((counter)*60)/(period/1000); // Calculate revolutions per minute (RPM)
}

void SpeedWind(){
  speedwind = (((4 * pi * radius * RPM)/60) / 1000)*3.6; // Calculate wind speed on km/h
}

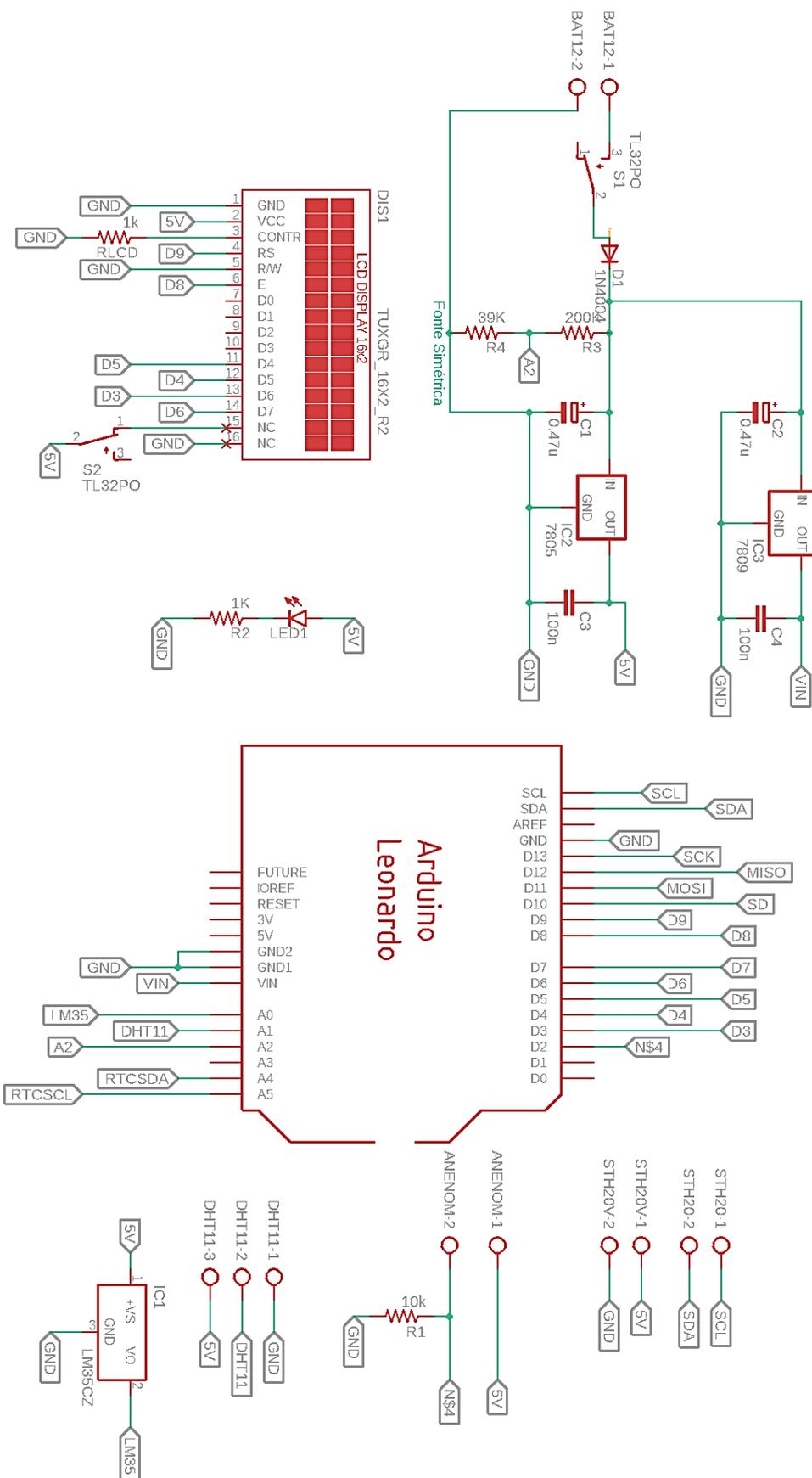
```

```
void addcount() {  
    counter++;  
}
```

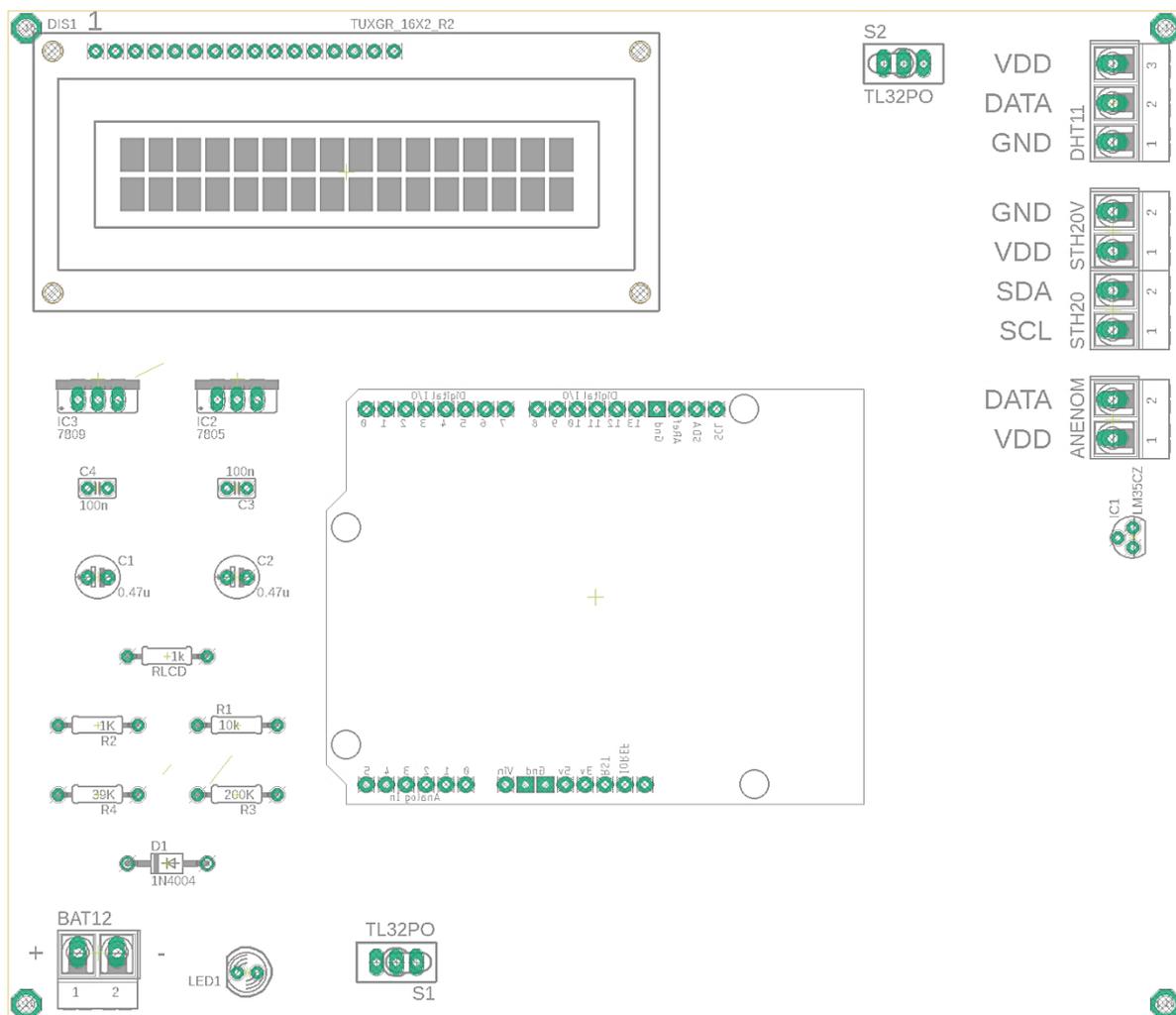
### APÊNDICE 3 – Esquemático do *Logger Shield*

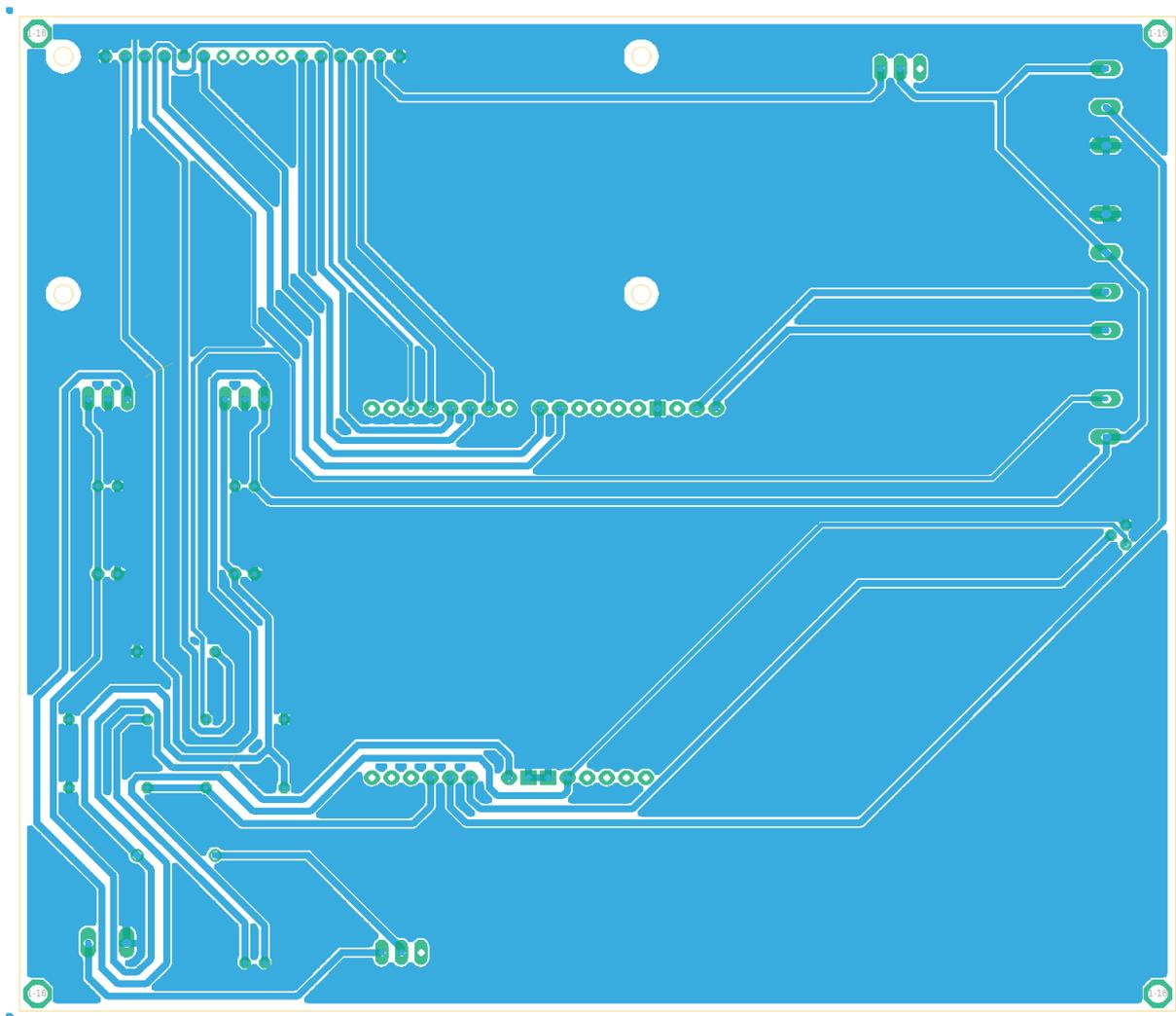


### APÊNDICE 4 – Esquemático do circuito



## APÊNDICE 5 – Board do circuito





## APÊNDICE 6 – Parte do Banco de dados

Data	Hora	Temperatura Solo [C]	Umidade Solo [%]	Temperatura do Ar DHT11 [C]	Umidade do Ar DHT11 [%]	Temperatura Bat [C]	Tensão Bat [V]	Velocidade do Vento [Km/h]
16/10/2018	10:7:46	25.27	94.02	25.00	80.00	25.90	12.50	1.33
16/10/2018	10:8:4	25.24	94.07	25.00	79.00	25.90	12.40	5.32
16/10/2018	10:8:22	25.24	94.14	25.00	79.00	23.46	12.40	2.66
16/10/2018	10:8:40	25.23	94.19	25.00	79.00	27.37	12.40	0.00
16/10/2018	10:8:58	25.22	94.28	25.00	80.00	24.44	12.40	2.66
16/10/2018	10:9:16	25.21	94.33	25.00	80.00	23.95	12.47	2.66
16/10/2018	10:9:34	25.20	94.39	25.00	80.00	22.97	12.40	2.66
16/10/2018	10:9:52	25.20	94.42	25.00	80.00	27.37	12.40	0.00
16/10/2018	10:10:10	25.19	94.48	25.00	80.00	27.37	12.40	0.00
16/10/2018	10:10:28	25.19	94.50	25.00	80.00	27.37	12.40	0.00
16/10/2018	10:10:46	25.18	94.56	25.00	81.00	23.46	12.40	0.00
16/10/2018	10:11:4	25.18	94.59	25.00	80.00	22.97	12.40	2.66
16/10/2018	10:11:22	25.18	94.63	25.00	81.00	21.99	12.47	2.66
16/10/2018	10:11:40	25.18	94.69	25.00	80.00	26.39	12.40	2.66
16/10/2018	10:11:58	25.17	94.75	25.00	80.00	27.37	12.40	0.00
16/10/2018	10:12:16	25.17	94.78	25.00	80.00	27.37	12.40	0.00
16/10/2018	10:12:34	25.17	94.81	25.00	79.00	27.86	12.40	0.00
16/10/2018	10:12:52	25.16	94.85	25.00	79.00	22.48	12.40	0.00
16/10/2018	10:13:10	25.16	94.88	25.00	79.00	27.37	12.40	2.66
16/10/2018	10:13:28	25.16	94.91	25.00	79.00	26.39	12.47	5.32
16/10/2018	10:13:46	25.15	94.94	25.00	79.00	23.46	12.40	3.99
16/10/2018	10:14:4	25.15	94.98	25.00	79.00	22.97	12.40	2.66
16/10/2018	10:14:22	25.15	95.01	25.00	79.00	27.86	12.40	2.66
16/10/2018	10:14:40	25.15	95.04	25.00	79.00	22.97	12.40	2.66
16/10/2018	10:14:58	25.15	95.07	25.00	80.00	26.88	12.40	5.32
16/10/2018	10:15:16	25.15	95.10	25.00	79.00	27.86	12.40	5.32
16/10/2018	10:15:34	25.14	95.10	25.00	79.00	27.86	12.40	2.66
16/10/2018	10:15:52	25.15	95.13	25.00	80.00	27.86	12.47	0.00
16/10/2018	10:16:10	25.14	95.16	25.00	80.00	25.90	12.40	0.00
16/10/2018	10:16:28	25.14	95.19	25.00	80.00	22.97	12.40	0.00
16/10/2018	10:16:46	25.14	95.19	25.00	81.00	22.97	12.40	0.00
16/10/2018	10:17:4	25.14	95.26	25.00	81.00	27.37	12.40	2.66
16/10/2018	10:17:22	25.14	95.29	25.00	81.00	22.97	12.40	0.00
16/10/2018	10:17:40	25.14	95.32	25.00	81.00	27.86	12.40	0.00
16/10/2018	10:17:58	25.14	95.32	25.00	82.00	25.42	12.50	0.00
16/10/2018	10:18:16	25.14	95.35	25.00	82.00	27.86	12.40	0.00
16/10/2018	10:18:34	25.14	95.38	25.00	82.00	23.95	12.40	0.00
16/10/2018	10:18:52	25.14	95.41	25.00	82.00	22.97	12.40	0.00
16/10/2018	10:19:10	25.14	95.44	25.00	82.00	27.86	12.40	0.00

16/10/2018	10:19:28	25.14	95.44	25.00	82.00	27.86	12.40	5.32
16/10/2018	10:19:46	25.13	95.47	25.00	82.00	23.46	12.40	2.66
16/10/2018	10:20:4	25.14	95.47	25.00	82.00	26.88	12.47	2.66
16/10/2018	10:20:22	25.14	95.47	25.00	82.00	27.37	12.40	7.98
16/10/2018	10:20:40	25.14	95.42	24.00	82.00	24.44	12.40	2.66
16/10/2018	10:20:58	25.14	95.47	24.00	82.00	24.44	12.40	2.66
16/10/2018	10:21:16	25.14	95.42	24.00	82.00	24.44	12.40	2.66
16/10/2018	10:21:34	25.14	95.42	24.00	83.00	27.37	12.40	2.66
16/10/2018	10:21:52	25.14	95.50	24.00	83.00	23.95	12.40	0.00
16/10/2018	10:22:10	25.13	95.50	24.00	83.00	22.48	12.40	2.66
16/10/2018	10:22:28	25.13	95.53	24.00	83.00	27.37	12.40	0.00
16/10/2018	10:22:46	25.13	95.56	24.00	83.00	26.39	12.40	0.00
16/10/2018	10:23:4	25.14	95.56	24.00	83.00	24.93	12.40	0.00
16/10/2018	10:23:22	25.14	95.59	24.00	83.00	22.48	12.40	0.00

17/10/2018	3:13:13	24.99	98.70	21.00	94.00	22.97	12.11	1.33
17/10/2018	3:13:31	24.98	98.70	21.00	94.00	20.04	12.11	1.33
17/10/2018	3:13:49	24.98	98.70	21.00	94.00	21.51	12.11	1.33
17/10/2018	3:14:7	24.94	98.68	21.00	94.00	16.62	12.11	0.00
17/10/2018	3:14:25	24.95	98.68	21.00	94.00	20.04	12.08	0.00
17/10/2018	3:14:43	24.94	98.68	21.00	94.00	22.97	12.11	1.33
17/10/2018	3:15:1	24.95	98.68	21.00	94.00	22.48	12.08	1.33
17/10/2018	3:15:19	24.97	98.73	21.00	94.00	16.62	12.21	0.00
17/10/2018	3:15:37	24.95	98.73	21.00	94.00	19.06	12.08	2.66
17/10/2018	3:15:55	24.97	98.73	21.00	94.00	18.57	12.11	2.66
17/10/2018	3:16:13	24.97	98.73	21.00	94.00	18.57	12.11	1.33
17/10/2018	3:16:31	24.97	98.70	21.00	94.00	20.04	12.11	1.33
17/10/2018	3:16:49	24.99	98.70	21.00	94.00	17.11	12.11	0.00
17/10/2018	3:17:7	24.99	98.64	21.00	94.00	23.46	12.08	0.00
17/10/2018	3:17:25	24.95	98.64	21.00	94.00	17.11	12.21	1.33
17/10/2018	3:17:43	24.93	98.68	21.00	94.00	20.53	12.11	0.00
17/10/2018	3:18:1	24.92	98.71	21.00	94.00	22.48	12.08	0.00
17/10/2018	3:18:19	24.97	98.73	21.00	94.00	20.04	12.11	0.00
17/10/2018	3:18:37	24.95	98.73	21.00	94.00	17.11	12.11	0.00
17/10/2018	3:18:55	25.00	98.70	21.00	94.00	16.62	12.08	0.00
17/10/2018	3:19:13	14.69	98.69	21.00	94.00	16.62	12.08	0.00
17/10/2018	3:19:49	25.07	98.39	22.00	95.00	21.51	12.21	1.33
17/10/2018	3:20:7	25.05	98.35	22.00	95.00	16.62	12.08	1.33
17/10/2018	3:20:25	25.01	98.48	21.00	94.00	22.97	12.11	0.00