

MINISTÉRIO DA EDUCAÇÃO
UNIVERSIDADE FEDERAL DO PARANÁ

PATRICK THIERRY LORUSSO EL OMAIRI

**DESENVOLVIMENTO DE UMA SOLUÇÃO INTERNET DAS COISAS
PARA MEDIDA REMOTA DE TEMPERATURA, UMIDADE DO AR E
DO SOLO COM BAIXO CUSTO E BAIXO CONSUMO DE ENERGIA**

CURITIBA

2018

PATRICK THIERRY LORUSSO EL OMAIRI

**DESENVOLVIMENTO DE UMA SOLUÇÃO INTERNET DAS COISAS
PARA MEDIDA REMOTA DE TEMPERATURA, UMIDADE DO AR E
DO SOLO COM BAIXO CUSTO E BAIXO CONSUMO DE ENERGIA**

Trabalho de conclusão de curso
apresentado como requisito parcial
para a obtenção do grau de engenheiro
eletricista, no programa de graduação
em engenharia elétrica da Universidade
Federal do Paraná.

Orientador: Prof. Dr. Márlcio José do Couto
Bonfim

CURITIBA

2018

AGRADECIMENTOS

Agradeço o Professor Málio José do Couto Bonfim pela orientação durante todo esse trabalho que foi crucial para o sucesso do mesmo. Gostaria de agradecer também toda a minha família, minha namorada e meus amigos que tanto me apoiaram em todos os momentos.

RESUMO

Considerando a grande extensão territorial do Brasil e suas vastas lavouras, é imprescindível o uso de tecnologia na agricultura para aumento da produtividade. Dispositivos *IoT* e o avanço da agricultura 4.0 podem auxiliar imensamente nesta tarefa. Neste contexto, esse trabalho propõe o desenvolvimento de uma solução de medição remota da temperatura, umidade relativa do ar e umidade do solo, com transmissão por uma rede *IoT* LoRa de baixo consumo, com a intenção de obter uma solução com uma bateria de longa duração e baixo custo, com o uso de um microcontrolador STM32 e os sensores integrados na placa. O escopo deste trabalho vai desde o desenvolvimento teórico desta solução até um protótipo para ser testado em ambientes controlados. Este projeto será *open source* e *open hardware*, para que fazendeiros e pesquisadores da área possam utilizá-lo livremente. O protótipo desenvolvido tem uma autonomia de aproximadamente um ano e quatro meses com duas pilhas AA alcalinas, com um preço de 25 reais cada sensor. O protótipo mostra que é possível o desenvolvimento de um dispositivo como foi proposto.

Palavras Chave Agricultura 4.0, IoT, umidade do solo, LoRa.

ABSTRACT

Considering Brazil has a vast territory and a big part of it is being used for agriculture, it is crucial to use technology to augment productivity. IoT devices and agriculture 4.0 can help immensely in this task. In this context, this project seeks to develop a solution for the remote monitoring of temperature, soil moisture, and air humidity, using a LoRa network for transmission and an STM32L051 microcontroller aiming for a long battery life and low cost of the device. This project goes from the theory until the prototype, and it is open source and open hardware so that people can use this project in the future freely. The prototype has a battery life of about one year and four months, costing US\$ 6.65 a piece. The prototype shows it's possible to develop a product based on what was proposed.

Key-words: Agriculture 4.0, IoT, soil moisture, LoRa

LISTA DE FIGURAS

2.1	Curva característica de um resistor NTC. Fonte: O Autor (2018)	11
2.2	Disposição dos condutores no sensor capacitivo. Fonte: DEAN et al. (2012)	13
2.3	Visão transversal dos condutores no sensor capacitivo. Fonte: GOSWAMI, MONTAZER e SARMA (2018)	13
2.4	Divisor de tensão com um sensor. Fonte: O Autor (2018)	14
4.1	Esquemático do circuito de condicionamento feito no KiCAD. Fonte: O Autor (2018)	17
4.2	Simulação da resposta do sensor aproximado com uma função de primeiro e terceiro grau e erros associados, a potência dissipada no NTC e o aumento da temperatura devido a essa potência dissipada. Fonte: O Autor (2018)	18
4.3	Fluxograma do funcionamento do sensor NTC. Fonte: O Autor(2018)	19
4.4	Simulação de campo elétrico para sensor com $w = 1,5mm$ e $d = 0,3mm$ com ar em volta do sensor. Fonte: O Autor (2018)	20
4.5	Layout do sensor no software KiCAD. Fonte: O Autor (2018)	21
4.6	Sensor feito a partir do desenho feito no KiCAD. Fonte: O Autor (2018)	22
4.7	Circuito para aquisição do sinal do sensor de umidade do solo. Fonte: O Autor (2018)	23
4.8	Circuito para simulação do sinal do sensor de umidade do solo. Fonte: O Autor (2018)	24
4.9	Resultado da simulação do sinal do sensor de umidade do solo. Fonte: O Autor (2018)	24
4.10	Fluxograma de funcionamento do sensor de umidade do solo. Fonte: O Autor (2018)	26
4.11	Circuito de condicionamento para o HR202. Fonte: O Autor(2018)	27
4.12	Ajuste de curva para a resposta do sensor HR202. Fonte: O Autor(2018)	27
4.13	Simulações do sensor HR202. Fonte: O Autor(2018)	29
4.14	Simulação do sensor HR202 no LTSPice. Fonte: O Autor(2018)	29
4.15	Fluxograma do funcionamento do HR202. Fonte: O Autor (2018)	31
4.16	Esquemático do módulo LoRa. Fonte: O Autor(2018)	32
4.17	Mostra os <i>spreading factors</i> e o a quantidade de símbolo por bit de cada um e a sensibilidade necessária no receptor de cada um. Fonte: SEMTECH (2015)	33
4.18	Relaciona a largura de banda e a velocidade de transmissão. Fonte SEMTECH (2015).	33
4.19	Conteúdo do pacote de transmissão LoRa. Fonte: SEMTECH (2015).	34
4.20	Antena do módulo LoRa. Fonte: O Autor (2018)	35
4.21	Diagrama da interação entre os módulos. Fonte: O Autor (2018)	38
4.22	Layout da PCB. Fonte: O Autor (2018)	39
4.23	Diagrama de conexões do <i>gateway</i> . Fonte: O Autor (2018)	39
5.1	Versões de hardware anteriores do projeto. Fonte: O Autor (2018)	42
5.2	Ultima versão da placa, v1.2. Fonte: O Autor(2018)	42

5.3	Tensão de saída na aquisição da temperatura. Fonte: O Autor (2018) .	43
5.4	Tensão de saída do sensor de umidade do solo, teste com o capacitor. Fonte: O Autor (2018)	44
5.5	Demonstração do aumento da grandeza medida quando é colocado mais água na terra onde se encontra o sensor. Fonte: O Autor (2018) .	44
5.6	Regressão linear das medidas obtidas para obtenção da equação que relaciona capacitância do sensor com umidade do solo. Fonte: O Autor (2018)	46
5.7	Tensão de saída na aquisição da umidade relativa do ar. Fonte: O Autor (2018)	47
5.8	Tensão de saída quando o efeito da capacitância é mais forte. Fonte: O Autor (2018).	47
5.9	Evolução da umidade relativa do ar dentro do recipiente com uma solução de sal saturada, mostra que o sensor está calibrado. Fonte: O Autor(2018)	48
5.10	Valores coletados pelo sensor e enviados para o <i>gateway</i> via <i>LoRa</i> e armazenados. Fonte: O Autor (2018).	49
5.11	Interface <i>phpLiteAdmin</i> para o banco de dados <i>SQLite</i> . Fonte: O Autor (2018)	50

LISTA DE SIGLAS

IoT	Internet das coisas
MCU	Microcontrolador
RTC	Relógio tempo real
GPIO	Pino de entrada ou saída
PCB	Placa de circuito impresso
ADC	Conversor analógico-digital
TDR	Reflectometria no domínio do tempo

SUMÁRIO

RESUMO	5
ABSTRACT	6
1 INTRODUÇÃO	9
1.1 Problematização	9
1.2 Objetivos	9
1.2.1 Objetivo Geral	9
1.2.2 Objetivos Específicos	9
1.3 Justificativa	10
2 FUNDAMENTAÇÃO TEÓRICA	11
2.1 Sensor de temperatura	11
2.2 Sensor de umidade relativa do ar	11
2.3 Sensor capacitivo de umidade do solo	12
2.4 Método da inflexão para linearização de sensores	13
2.5 Comunicação - LoRa	14
3 METODOLOGIA	15
4 DESENVOLVIMENTO	16
4.1 Sensor de temperatura	16
4.1.1 Concepção	16
4.1.2 Simulação	16
4.1.3 Código e algoritmos	16
4.2 Sensor de umidade do solo capacitivo	17
4.2.1 Concepção	17
4.2.2 Determinação dos parâmetros do sensor	20
4.2.3 Simulação	23
4.2.4 Código e algoritmos	23
4.3 Sensor de umidade relativa do ar	25
4.3.1 Concepção do sensor	25
4.3.2 Simulação do circuito	28
4.3.3 Código e algoritmos	28
4.4 Modulo de comunicação LoRa	32
4.4.1 Circuito	32
4.4.2 Corpo de dados	34
4.4.3 Antena	34
4.4.4 Implementação no código	34
4.5 Bateria	35
4.5.1 Bateria de íon-lítio	35
4.5.2 Pilha AA Alcalina	35
4.6 Microcontrolador	36
4.6.1 Timers	36

4.6.2	ADC	36
4.6.3	Comparador	36
4.6.4	Interfaceamento	36
4.6.5	Osciladores	37
4.6.6	RTC	37
4.6.7	Modos de baixo consumo	37
4.6.8	Botões, LEDs e SWD	37
4.7	Firmware	38
4.8	Desenvolvimento da PCB	38
4.9	Gateway LoRa	38
5	RESULTADOS	41
5.1	Placa de circuito impresso	41
5.1.1	Versões de hardware da PCB	41
5.2	Sensor de temperatura	41
5.2.1	Calibração	41
5.3	Sensor de umidade do solo	44
5.3.1	Calibração	44
5.4	Sensor de umidade relativa do ar	46
5.4.1	Calibração	46
5.5	Comunicação LoRa	48
5.6	Consumo de energia	48
5.7	Gateway	49
5.8	Preço estimado	49
6	CONCLUSÃO	51
	BIBLIOGRAFIA	52
A	ESQUEMÁTICO COMPLETO DO CIRCUITO	53

CAPÍTULO 1

INTRODUÇÃO

1.1 Problematização

Com a crescente demanda para uma produção agrícola mais eficiente, devido ao crescimento populacional em escala mundial e nacional, existe uma demanda muito grande para métodos mais eficientes e ecológicos de agricultura.

Uma das maneiras de deixar o plantio mais eficiente, é medindo grandezas no campo, como por exemplo a temperatura, a umidade do solo e do ar. O acompanhamento dessas grandezas em tempo real pode dar informações essenciais para o fazendeiro. Por exemplo, tendo informações sobre a umidade do solo, será possível saber quando será o momento ideal para irrigar os campos.

Com vários sensores capazes de medir tais grandezas espalhados pela fazenda, o fazendeiro seria capaz de obter informações mais específicas sobre determinadas partes do campo, como a distribuição da temperatura, da umidade do solo e relativa do ar, e derivados destes.

Este projeto visa desenvolver um sensor capaz de medir a temperatura, a umidade relativa do ar e a umidade do solo, capaz de enviar essas informações por uma rede *IoT*, mais especificamente, *LoRa*, tendo um consumo baixo e consequentemente uma grande autonomia.

1.2 Objetivos

1.2.1 Objetivo Geral

Desenvolver um dispositivo eletrônico capaz de medir temperatura, umidade relativa do ar e do solo, e enviar os dados por uma rede *IoT LoRa*. Todos os sensores integrados junto do microcontrolador numa só placa de baixo custo e baixo consumo de energia. O sensor de umidade do solo será capacitivo para que ele tenha uma boa performance e será integrado dentro da placa de circuito impresso.

1.2.2 Objetivos Específicos

- Desenvolvimento do hardware.
- Simulação do hardware no LTSpice ou em *scripts* Python.
- Desenvolvimento da uma placa de circuito impresso.
- Teste do hardware.
- Desenvolvimento do software embarcado para. microcontrolador STM32
- Implementação da comunicação via rede LoRa.

1.3 Justificativa

As exportações brasileiras ligadas ao agronegócio chegam a 41,5% das vendas externas totais no país, onde somente a soja é o principal produto exportado com 42,5% das exportações nacionais do agronegócio (FIESP, 2018).

Além disso, a produção de carnes, outro setor muito importante para a exportação nacional, depende diretamente da agricultura uma vez que as rações dos animais são feitas com produtos provenientes da agricultura.

Sendo então a agricultura uma área essencial para o desenvolvimento econômico e sustentável do Brasil, inovações nessa área tem um potencial enorme para deixar mais eficiente o crescimento das plantas, diminuindo gastos com água, por exemplo.

Um dispositivo eletrônico capaz de disponibilizar essas informações do campo para o fazendeiro pode ser essencial para o desenvolvimento do negócio, uma vez que o fazendeiro poderá monitorar seu plantio. E com o surgimento das redes *IoT*, se torna mais viável colocar vários dispositivos desses, de baixo custo, espalhados pelo campo para conseguir o maior número de dados possível.

CAPÍTULO 2

FUNDAMENTAÇÃO TEÓRICA

2.1 Sensor de temperatura

O sensor de temperatura escolhido (NTC) é um sensor do tipo termistor, onde sua resistência diminui exponencialmente conforme a temperatura aumenta, como mostrado na equação 2.1 e na figura 2.1. Termistores NTC são muito utilizados graças a seu custo reduzido e sua praticidade, mesmo não sendo lineares e não tendo uma faixa de temperatura grande quanto outros sensores de temperatura.

$$R_{NTC} = R_0 \cdot e^{B \cdot \left(\frac{1}{T} - \frac{1}{T_{ref}} \right)} \quad (2.1)$$

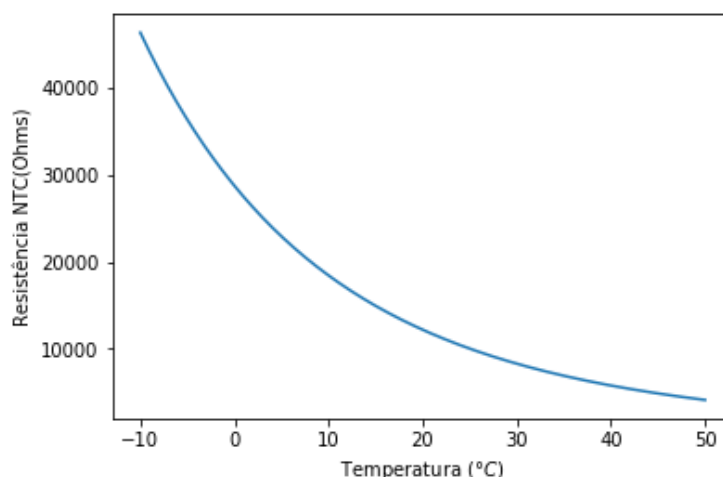


Figura 2.1: Curva característica de um resistor NTC. Fonte: O Autor (2018)

Mesmo com seu comportamento não linear, é possível linearizar o sensor com um resistor de linearização, fazendo um divisor de tensão entre o sensor e este resistor. O valor deste resistor pode ser calculado pelo método de inflexão, como é explicado na seção 2.4.

2.2 Sensor de umidade relativa do ar

O sensor de umidade do ar escolhido, o HR202, é um sensor de umidade relativa do ar, ele contém uma película orgânica que muda a resistência de acordo com a umidade relativa do ar. Para este sensor é necessário que seja aplicada uma tensão alternada sobre ele para que ele não perca suas propriedades, pois a película dele é sensível e pode mudar suas propriedades caso seja aplicada uma tensão contínua sobre o sensor.

Este sensor também é exponencial, mas não constam no seu *datasheet* os parâmetros dessa equação, somente uma tabela onde se podem aproximar os valores.

2.3 Sensor capacitivo de umidade do solo

Para medir a umidade do solo, uma das maneiras é fabricar um capacitor onde o dielétrico seja o solo. Como a água tem uma constante dielétrica muito maior que os outros elementos, é possível estimar a quantidade de água na amostra com base na capacitância.

Sensores capacitivos para medir umidade são eficientes pois a permissividade da água é muito maior que a do ar, vácuo ou solo, DEAN et al. (2012) cita algumas aplicações onde sensores capacitivos deste tipo são adotados:

- Umidade do solo
- Umidade em grãos
- Umidade em papel
- Detecção de chuva
- Sensor de proximidade
- Botões capacitivos
- Sensores biomédicos

Segundo DEAN et al. (2012), outras maneiras de medir a umidade do solo são:

- Propagação de ondas sísmicas
- Sonda de nêutrons
- Sonda de raios gama
- Reflectometria no domínio do tempo (TDR)

A grande desvantagem de todas estas outras maneiras é a complexidade e o custo, se pode fazer um sensor capacitivo numa placa de circuito impresso, normalmente esses outros métodos necessitam de circuitos em uma frequência muito maior, principalmente o mais usado entre eles, o TDR. A vantagem do TDR é que ele é mais preciso.

Ao contrário de um capacitor de placas paralelas, um sensor capacitivo deste tipo é feito de maneira em que os condutores fiquem no mesmo plano, como na figura 2.2, um corte transversal da estrutura pode ser visto na figura 2.3.

A largura de isolamento, do condutor e de repetições afetam diretamente o desempenho do sensor. Quanto mais repetições maior será a capacitância, quanto menor a isolamento maior será a capacitância, e quanto maior a largura do condutor maior será a penetração do campo elétrico no solo (MIZUGUCHI et al., 2015).

Várias configurações diferentes, com variações na largura do condutor, da isolamento e do uso da segunda camada são propostas, GOSWAMI, MONTAZER e SARMA (2018) faz uma comparação entre várias configurações propostas por outros artigos e demonstra vantagens do uso das duas camadas para aumentar a sensibilidade do sensor.

Este sensor deve ser calibrado para cada tipo de solo (GOSWAMI; MONTAZER; SARMA, 2018).

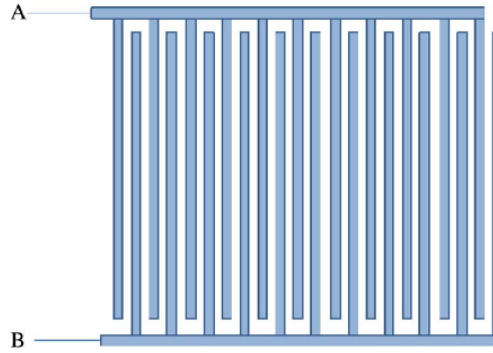


Figura 2.2: Disposição dos condutores no sensor capacitivo. Fonte: DEAN et al. (2012)

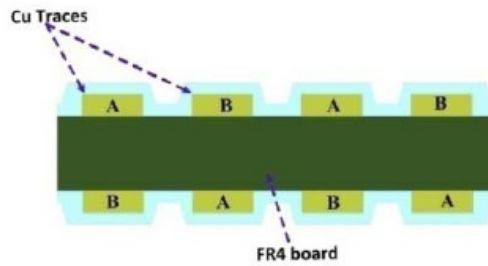


Figura 2.3: Visão transversal dos condutores no sensor capacitivo. Fonte: GOSWAMI, MONTAZER e SARMA (2018)

2.4 Método da inflexão para linearização de sensores

O ponto de inflexão é o ponto onde a segunda derivada é nula, logo, é o ponto onde a curva muda de côncava para convexa ou vice versa. Neste ponto, se uma reta for traçada neste ponto com a inclinação (ou seja, a primeira derivada) deste ponto, será obtida uma função de linearização que terá erros muito baixos, principalmente perto do ponto.

Sensores como o NTC e o sensor de umidade relativa do ar tem características exponenciais, que podem ser representadas por uma equação como a 2.2, logo é essencial poder linearizá-las, onde X é o valor da grandeza medida, e B é uma constante.

$$R_{sensor}(X) = R_0 \cdot e^{(B/X - B/X_0)} \quad (2.2)$$

Quando é feito o circuito de aquisição de um sensor exponencial como o NTC ou um sensor de umidade relativa do ar, é usada uma configuração de divisor de tensão como esta da figura 2.4, e a tensão de saída pode ser descrita pela equação 2.3.

$$V_o = V_{cc} \cdot \frac{R_s}{R_s + R_{sensor}} \quad (2.3)$$

Como esta equação muda o comportamento da saída, ela muda também o local do ponto de inflexão em função do resistor R_s . Substituindo 2.2 em 2.3, derivando duas vezes e isolando R_s obtemos a equação 2.4, onde X_c é o valor da grandeza medida no meio da faixa de interesse, e B é uma constante dada no *datasheet*.

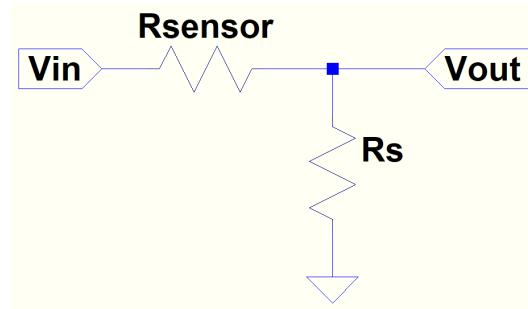


Figura 2.4: Divisor de tensão com um sensor. Fonte: O Autor (2018)

$$R_s = R_{sensor}(X_c) \cdot \frac{B - 2 \cdot X_c}{B + 2 \cdot X_c} \quad (2.4)$$

Onde R_{sensor} é a resistência do sensor no meio da faixa de interesse. Com esta equação é possível mudar o ponto de inflexão para o meio da faixa de interesse, assim diminuindo o erro nos circuitos de medição. Além disso o divisor de tensão ajuda a linearizar o sensor exponencial, como pode ser visto na figura 2.1 a resistência de um NTC, e na figura 4.2 este mesmo sensor onde foi aplicado o método da inflexão para a linearização.

2.5 Comunicação - LoRa

A rede LoRa é uma rede de baixo consumo, alto alcance e baixa taxa de transmissão. É uma rede muito utilizada por dispositivos *IoT*. LoRa utiliza um sistema de modulação com *chirps*, uma senoide que muda de frequência em função do tempo.

É possível que o alcance de um *gateway* possa chegar até 10 km, dependendo do local, da potência transmitida e do tipo de antena. Existem varias frequências de operação possíveis, dependendo da legislação local. Algumas frequências podem ser 433 MHz (EU), 868 MHz (EU) e 915 MHz (EUA).

CAPÍTULO 3

METODOLOGIA

A metodologia a ser seguida para desenvolver o projeto é a seguinte:

Começar por uma revisão bibliográfica, para ver o que já foi feito. O foco nessa parte é procurar artigos que implementaram, principalmente, a medida de umidade no solo, pois é o mais importante para a planta.

Desenvolvimento do hardware, onde será feita a escolha dos componentes, dos circuitos de condicionamento dos sensores, do microcontrolador e do módulo de transmissão LoRa.

O sensor de temperatura é simulado no python, para escolher o resistor ideal em série e comparar alguns diferentes.

O sensor de umidade do ar também é simulado no python, para traçar a tensão de saída e escolher os resistores ideais. Além de ser simulado no LTSpice para tentar modelar os componentes capacitivos deste sensor.

Para o sensor de umidade do solo, os parâmetros de largura e isolamento, são simulados no software FEMM. O sensor também é simulado no LTSpice para ver sua resposta ao circuito externo.

Uma vez a simulação feita, é testado o hardware em testes de bancada. Com o hardware funcionando, é feita a placa de circuito impresso no software KiCAD.

Com a PCB feita, é desenvolvido o software embarcado para STM32. O *firmware* será escrito em C, na IDE Atollic TrueSTUDIO. O software STM32CUBEMX também será utilizado para geração do código base.

Finalmente, será desenvolvida a comunicação via rede LoRa, com a construção de um *gateway* LoRa e com o código para esse *gateway*, onde é definida a *payload*, e os parâmetros de rede.

CAPÍTULO 4

DESENVOLVIMENTO

Todos os arquivos do projeto estão disponíveis no *GitHub* pelo link <https://github.com/patricktlo/lotSoilSensor>.

4.1 Sensor de temperatura

O sensor de temperatura escolhido foi o NTC, pois ele atende às demandas do projeto de baixo consumo, baixo custo e funciona independente da tensão de alimentação. Este último item é crucial pois como existe uma queda de tensão na bateria com sua descarga, é necessário um sensor que trabalhe independente da tensão.

4.1.1 Concepção

O circuito de condicionamento consiste em um divisor de tensão, sendo um sensor não linear, como visto na seção 2.1, necessita de um circuito de condicionamento para ser linearizado. Este circuito consiste simplesmente de um divisor de tensão, como mostrado na figura 4.1

O resistor de linearização foi escolhido usando o método da inflexão, como explicado na seção 2.4. O valor resultante foi $R = 7k\Omega$.

4.1.2 Simulação

Foram simulados, no python, a resposta do sensor real e aproximada por uma função de grau um e três, erros para esses dois casos também foram computados. Além disso foi simulada a potência dissipada no NTC e o aumento de temperatura no sensor devido a essa potência dissipada, os resultados estão na figura 4.2. O erro com a função de primeiro grau é menor (mais centrado) no caso onde $R = 10k\Omega$, logo este valor foi escolhido. O aquecimento do sensor é irrisório e pode ser ignorado.

A partir destas simulações, se pode pegar os coeficientes da reta para se obter a temperatura, neste caso temos a equação 4.1, que relaciona saída do ADC e temperatura, onde $a = 0,00966$ é a inclinação da reta e $b = 0,25$ é a interseção da reta em y .

$$T = \frac{ADC/4095 - b}{a} \quad (4.1)$$

4.1.3 Código e algoritmos

A energia provida para a ponte resistiva vem de um pino *GPIO*, para poder economizar energia e desligar quando medições não estarem sendo feitas.

A obtenção da temperatura a partir das medidas do divisor de tensão é feito com a equação 4.1 para uma aproximação de primeiro grau, ou 4.2 para uma

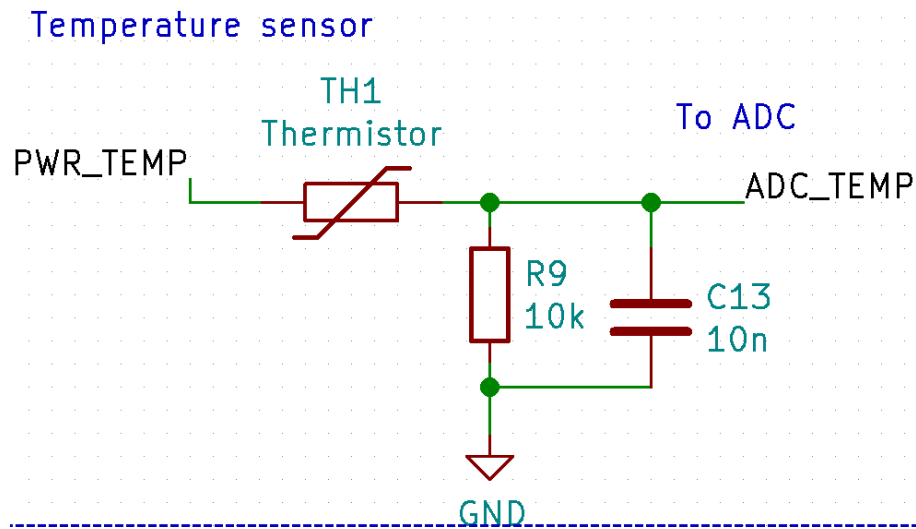


Figura 4.1: Esquemático do circuito de condicionamento feito no KiCAD. Fonte: O Autor (2018)

aproximação de terceiro grau, na segunda, onde seus coeficientes são $[a_3, a_2, a_1, a_0] = [174.324, -220.791, 193.122, -38.177]$. Porém isso depende de cada sensor e deve ser feito após a calibração do mesmo.

$$T = a_3 \cdot V_{out}^3 + a_2 \cdot V_{out}^2 + a_1 \cdot V_{out} + a_0 \quad (4.2)$$

O fluxograma da figura 4.3 representa o funcionamento do código. O programa começa configurando o ADC, ligando o ADC e ligando a alimentação do NTC, então são feitas 16 amostras e a média entre elas (esta parte é feita por hardware), o NTC e o ADC são desligados, e o valor é retornado.

Como a informação de temperatura não é de interesse do sensor e sim do *gateway*, que mostra a informação ao usuário, o sensor vai transmitir a informação do ADC bruta ao *gateway* e este irá fazer os cálculos para a temperatura, que é aplicar a equação 4.1 ou 4.2 à informação recebida.

4.2 Sensor de umidade do solo capacitivo

4.2.1 Concepção

Para medir a umidade do solo, foi escolhido um sensor capacitivo, que muda a capacitância de acordo com a quantidade de água no solo. Para isso, foi feito o layout de um sensor capacitivo em uma PCB, onde o dielétrico é o solo. Como a água tem uma permissividade elétrica em torno de 80 vezes maior do que a do solo, a capacitância aumenta com a quantidade de água no solo.

Mais informações sobre o design estão presentes na seção 2.3.

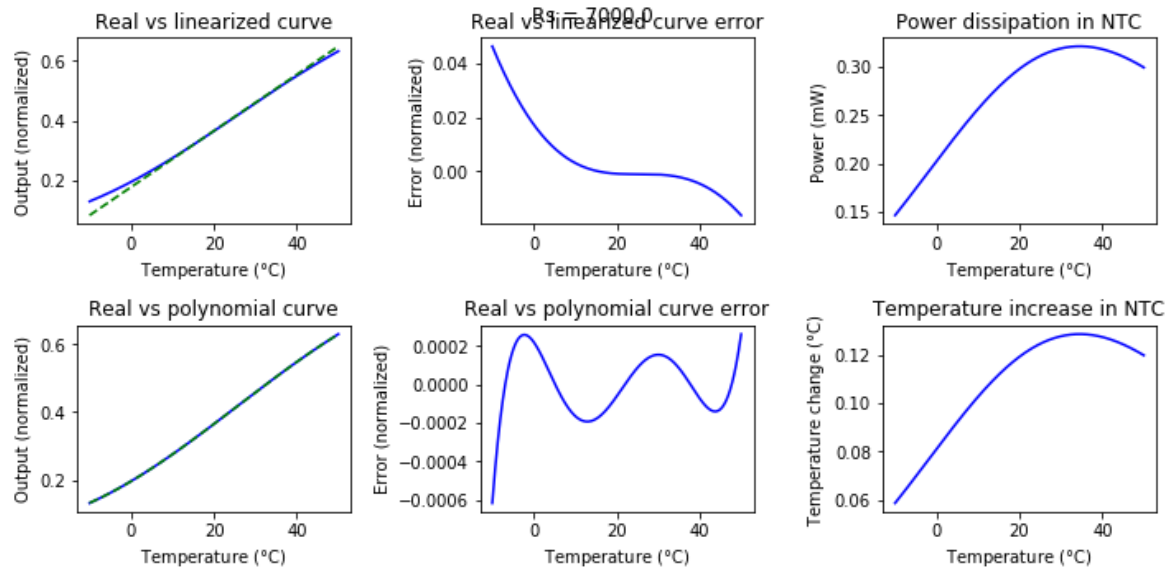
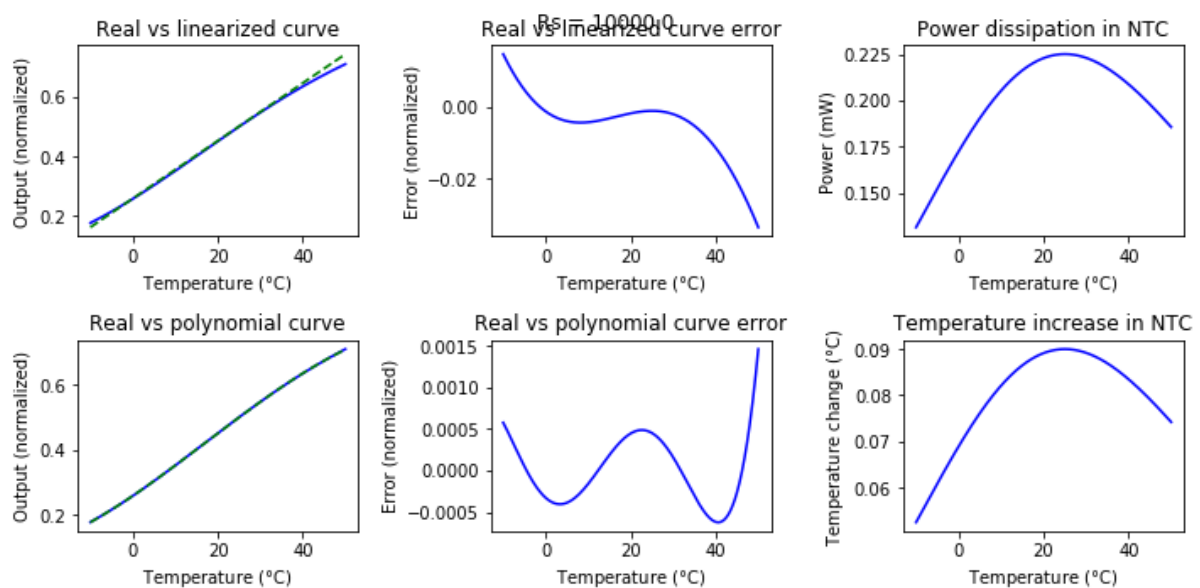
(a) Simulação com $R = 7k\Omega$ (b) Simulação com $R = 10k\Omega$

Figura 4.2: Simulação da resposta do sensor aproximado com uma função de primeiro e terceiro grau e erros associados, a potência dissipada no NTC e o aumento da temperatura devido a essa potência dissipada. Fonte: O Autor (2018)

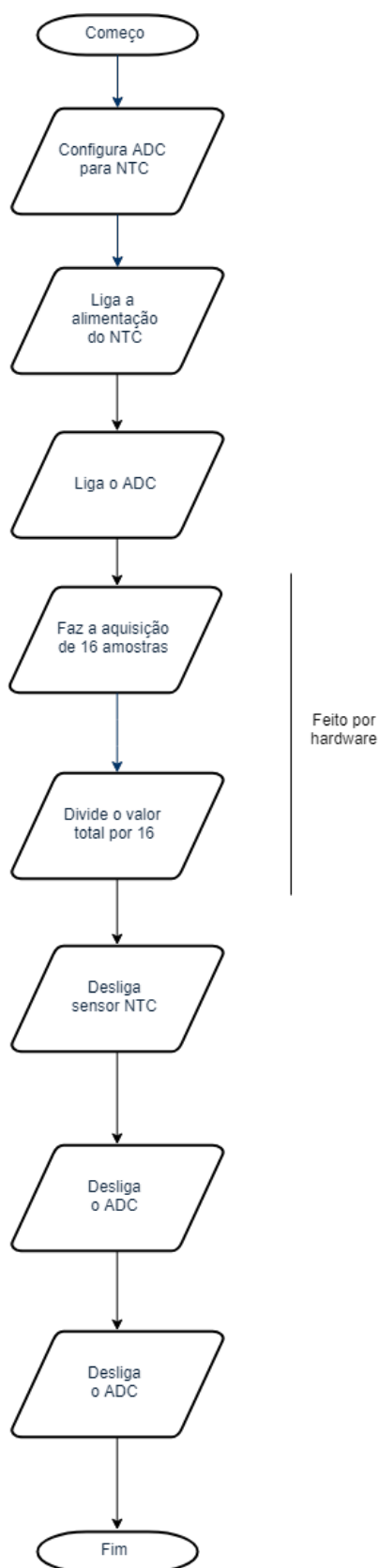


Figura 4.3: Fluxograma do funcionamento do sensor NTC. Fonte: O Autor(2018)

4.2.2 Determinação dos parâmetros do sensor

Para escolher os parâmetros de largura do condutor e do isolador (w, d , respectivamente) será usado o software FEMM para simular o efeito desses parâmetros na capacitância e na penetração do campo elétrico.

As simulações foram feitas considerando uma PCB de $1,6\text{mm}$ de espessura de fenolite, com cobre de espessura de $36\mu\text{m}$.

Os parâmetros escolhidos no final foram os da figura 4.4, com $d = 0,3\text{mm}$ pois é um valor factível para uma PCB caseira, e $w = 1,5\text{mm}$ pois oferece uma boa penetração do campo elétrico enquanto possibilita a repetição de vários conjuntos como esse para aumentar a sensibilidade do sensor.

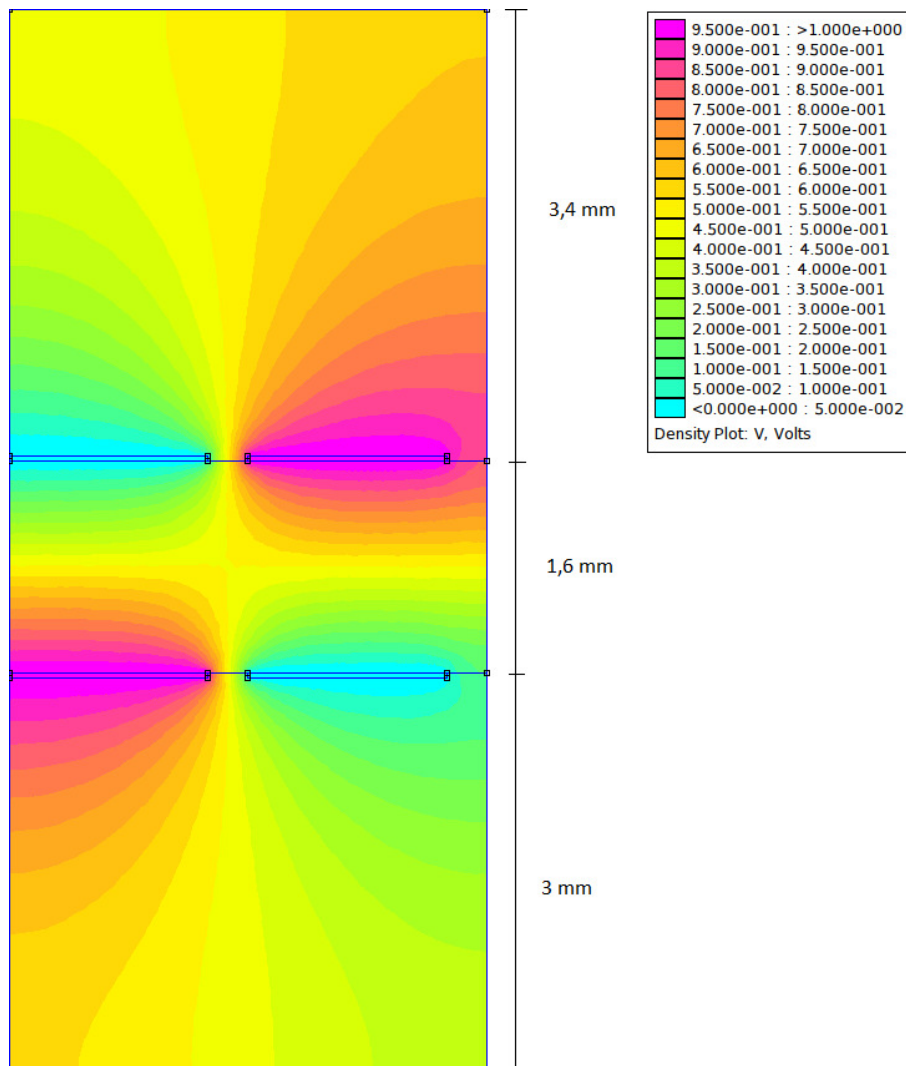


Figura 4.4: Simulação de campo elétrico para sensor com $w = 1,5\text{mm}$ e $d = 0,3\text{mm}$ com ar em volta do sensor. Fonte: O Autor (2018)

Na tabela 4.1 se encontram diversas configurações diferentes com os valores de capacitância para solo totalmente seco, solo parcialmente molhado e água para uma PCB de 47mm .

Segundo a simulação, a capacitância varia entre 57pF até 740pF , segunda a tabela 4.1, na coluna onde $w = 1,5\text{mm}$ e $d = 0,3\text{mm}$.

Tabela 4.1: Valores de capacitância para diversas configurações testadas

Dimensões	w (mm)	0.50	0.50	2.00	1.50	1.50
	d (mm)	0.50	0.30	0.90	0.50	0.30
C (pf)/par	Seco	2.80	3.17	5.42	4.84	5.23
	Terra 30%	15.89	18.92	20.65	22.26	25.77
	Molhado	42.37	50.74	51.20	57.51	67.34
Numero de pares		23.00	29.00	8.00	11.00	13.00
Ctotal (pF)	Seco	64.40	72.91	157.18	38.72	57.53
	Terra 30%	365.47	435.16	598.85	178.08	283.47
	Molhado	974.51	1167.02	1484.80	460.08	740.74

Desenvolvimento do sensor

O sensor desenvolvido foi desenhado como um componente no software KiCAD, o resultado está representado na figura 4.5.

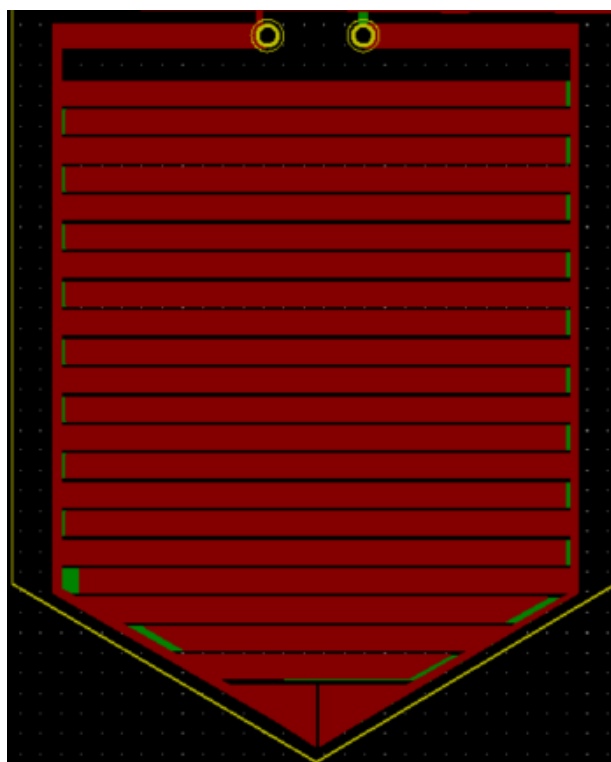


Figura 4.5: Layout do sensor no software KiCAD. Fonte: O Autor (2018)

O sensor foi produzido com o método fotossensível, o resultado pode ser visto na figura 4.6. Na placa final o sensor pode ser acoplado ao resto do circuito ou separado, neste caso ele está separado mas na figura 5.2 a placa está com o sensor acoplado.

Circuito de condicionamento

Para medir a capacitância do sensor, será usado um simples circuito RC, com um comparador interno ao microcontrolador que irá mandar um sinal quando o valor de

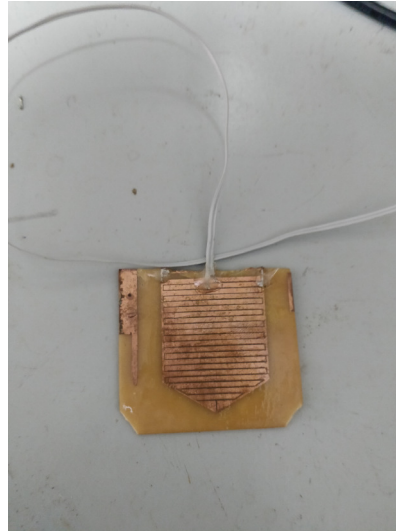


Figura 4.6: Sensor feito a partir do desenho feito no KiCAD. Fonte: O Autor (2018)

saída do RC for maior que um valor definido por uma ponte resistiva na entrada negativa do comparador.

O circuito de medição é apresentado na figura 4.7.

A escolha do resistor em serie, da frequência da onda e da ponte resistiva se baseiam nos valores simulados de capacitância.

Consideraremos como valor mínimo e máximo da capacitância do sensor $50pF$ e $2nF$ respectivamente, para ter uma margem de segurança. Considerando que o *timer* do *MCU* possua um valor de contador de 16 bits, e considerando que o circuito se carrega em $5 \cdot R \cdot C_s$, o valor máximo, em contagens do *timer* pode ser representado pela equação 4.3. Valores adequados para a resistência e a frequência para obter a maior resolução seriam $R = 220k\Omega$ e $f = 8MHz$, que são valores usuais e fáceis de obter.

$$5 \cdot R \cdot C_{max} \cdot f < 2^{16} \quad (4.3)$$

Além disso o valor de *auto reload* do *timer*, isto é, o valor que o *timer* irá recomençar a contagem, será de $60k$, que obedece a inequação 4.3, e corresponde à uma onda quadrada de período $T = 15ms$.

Para a ponte resistiva que irá delimitar o valor na entrada negativa do comparador, é conveniente escolher um valor onde tanto a carga quanto descarga do circuito RC tenham a mesma constante de tempo, usando a equação da carga e descarga de um circuito RC, nas equações 4.4 e 4.5, e igualando as duas se pode obter $t = RC \cdot (\ln(2) - \ln(1)) = 0,693 \cdot RC$, e para esse t a tensão normalizada pode ser obtida a partir de qualquer uma das duas equações e corresponde à $V(t)/V_0 = 0,5$, logo uma ponte resistiva com dois resistores iguais é desejada, os valores escolhidos são $22k\Omega$.

$$V(t) = V_0 \cdot (1 - e^{t/RC}) \quad (4.4)$$

$$V(t) = V_0 \cdot e^{t/RC} \quad (4.5)$$

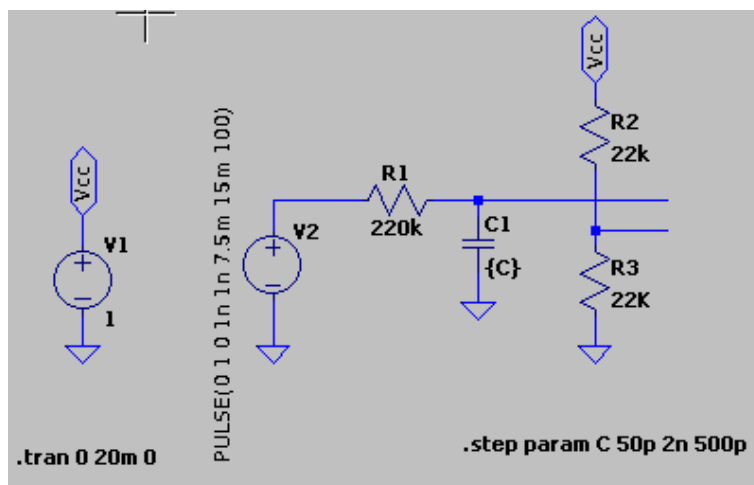


Figura 4.8: Circuito para simulação do sinal do sensor de umidade do solo. Fonte: O Autor (2018)

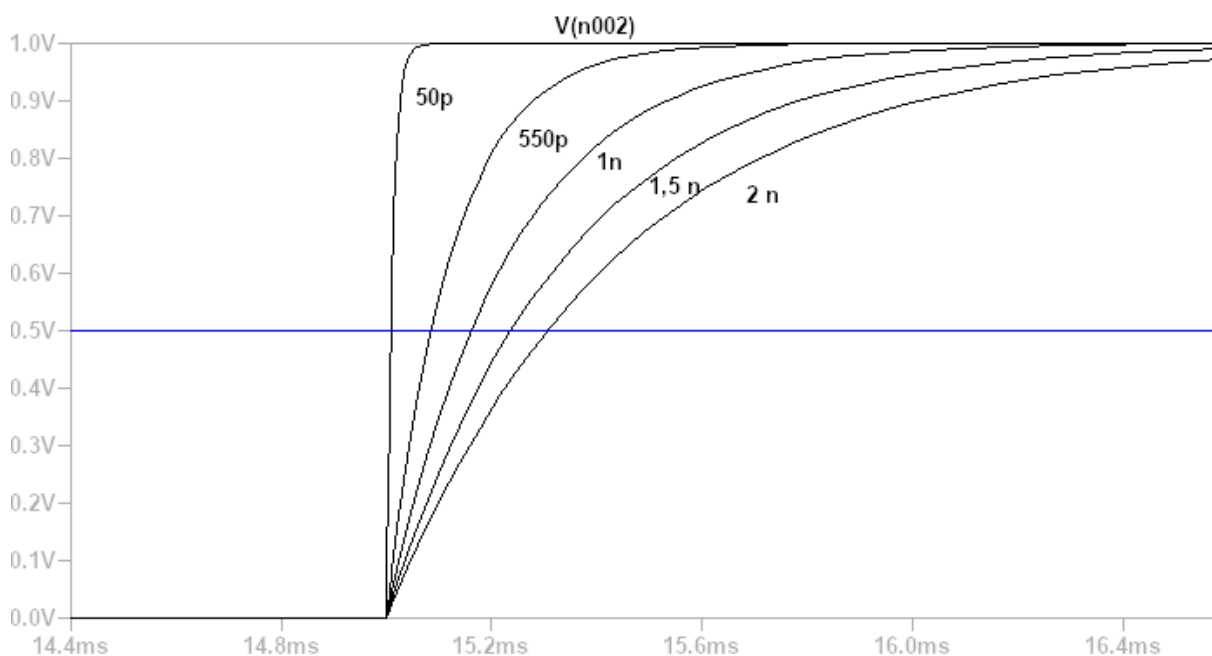


Figura 4.9: Resultado da simulação do sinal do sensor de umidade do solo. Fonte: O Autor (2018)

O funcionamento pode ser resumido no fluxograma da figura 4.10. O programa começa ligando a ponte resistiva (do comparador), o comparador e a onda quadrada (e consequentemente seu timer), então o programa checa em loop se o estado do comparador mudou, quando for o caso, é anotado o valor do timer (que será usado futuramente para cálculo da capacitância). Isto é repetido oito vezes para pegar várias medidas, e então é desligado todos os componentes, é feita a média dos oito valores e o valor é retornado.

4.3 Sensor de umidade relativa do ar

O sensor de umidade do ar escolhido é o HR202, pois este atende às necessidades do projeto de custo, precisão e não depende da tensão da bateria, já que é um circuito resistivo.

O sensor contém condutores imersos em uma camada de uma substância que muda a resistência com a umidade relativa do ar.

4.3.1 Concepção do sensor

Uma particularidade deste sensor é a necessidade da aplicação de uma tensão alternada sobre ele, para não mudar as propriedades do sensor.

Para fazer isto, o sensor foi ligado como na figura 4.11, ligando o sensor entre uma saída com uma onda quadrada e uma ponte resistiva que polariza a saída em $V_{cc}/2$.

Este sensor possui uma característica exponencial, porém, diferente do NTC o *datasheet* não contém nenhuma equação, só uma tabela de valores que variam com a temperatura. Além disto este sensor possui uma característica capacitiva, seu modelo elétrico é um resistor com um capacitor em paralelo (NENOV; IVANOV, 2007). A frequência escolhida para a onda quadrada é de $3,9Hz$, com um período de $256ms$, para evitar que os efeitos capacitivos interfiram nas medidas.

Para linearizar o circuito é utilizado o método da inflexão, a temperatura é fixada em $25^{\circ}C$, e usada a função *curve_fit* da biblioteca *scipy* para aproximar os dados da tabela para uma função de forma da equação 4.8, resultando na adequação mostrada na figura 4.12.

$$R_{HR202} = R_0 \cdot \exp\left(\frac{\beta}{RH} - \frac{\beta}{RH_0}\right) \quad (4.8)$$

A equação da resistência equivalente visto do ADC, do esquemático representado na figura 4.11, é representada na equação 4.9, onde R_s são as resistências que compõem a ponte resistiva da saída ($R9$ e $R10$ da figura 4.11) e R_{rh} é o valor do sensor.

$$R_{eq} = \frac{R_s \cdot R_{rh}}{R_s + 2 \cdot R_{rh}} \quad (4.9)$$

O método da inflexão, como explicado na seção 2.4, é aplicado substituindo R_{rh} na equação 4.9 pela equação 4.8, derivando duas vezes, igualando a zero e isolando R_s na equação, onde se chega no valor $R_s = 39k\Omega$, foi escolhido então $R_s = 47k\Omega$ por ser um valor mais comum.

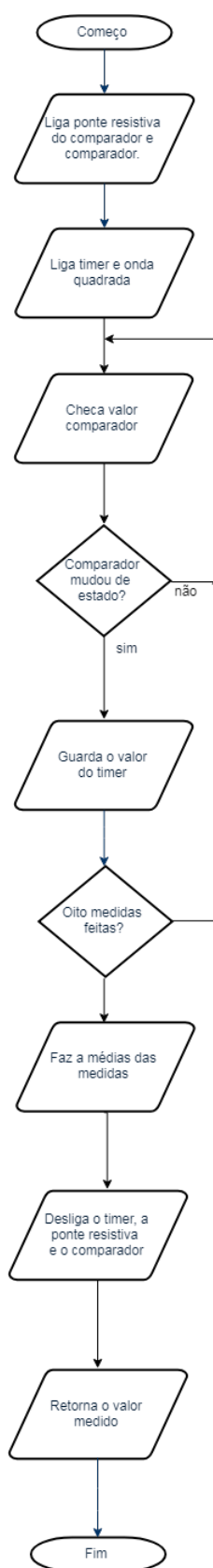


Figura 4.10: Fluxograma de funcionamento do sensor de umidade do solo. Fonte: O Autor (2018)

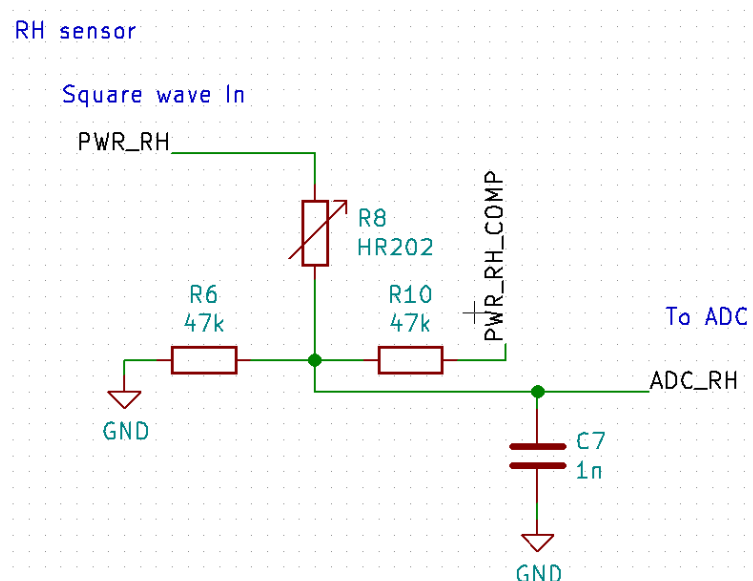


Figura 4.11: Circuito de condicionamento para o HR202. Fonte: O Autor(2018)

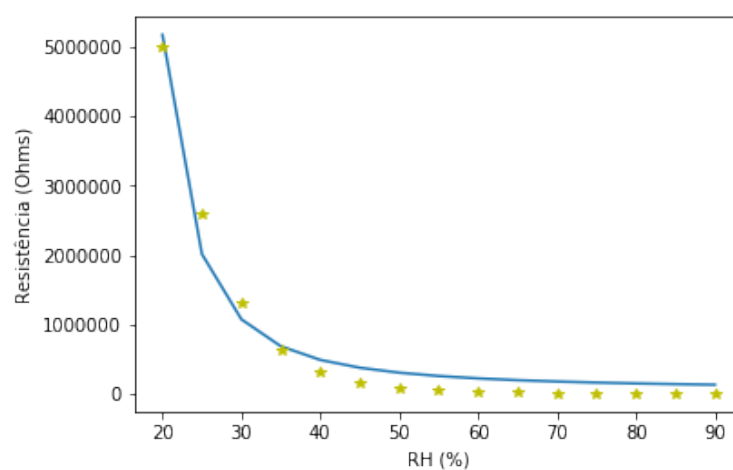


Figura 4.12: Ajuste de curva para a resposta do sensor HR202. Fonte: O Autor(2018)

O circuito final fica então como na figura 4.11, com a adição de um capacitor em paralelo com a entrada do ADC que possui a função de filtro passa-baixas e também para suprir cargas mais rapidamente ao capacitor de amostragem interno ao conversor ADC.

Para calcular a capacitância máxima deste capacitor, é necessário calcular a resistência equivalente máxima dos dois resistores de linearização e do sensor, que é dada pela equação 4.9 quando R_{rh} está no seu valor máximo, segundo seu *datasheet* o valor máximo é próximo de $13M\Omega$, logo podemos considerar que é um circuito aberto e temos como resistência equivalente o paralelo entre as duas resistências R_s do divisor de tensão, logo, $R_{pmax} = R_s/2$, e neste caso $R_{pmax} = 23,5k\Omega$.

Dado que a frequência é de $3,9Hz$, o período em que o sinal fica em cada sinal lógico é $128ms$, para calcular o capacitor, consideramos que é necessário $100ms$ para que a tensão fique estável e resta $28ms$ para a aquisição do ADC. Com isso o valor da capacitância é limitada pela equação 4.10.

$$10 \cdot R \cdot C < T_{ADC}/2 \quad (4.10)$$

Inserindo os valores na equação 4.10, temos que $C < 59nF$, o valor escolhido é $C = 4,7nF$, para manter um valor alto mas que não interfira muito.

4.3.2 Simulação do circuito

Este circuito foi simulado em um *script* python e no LTSpice.

No *script* python foi simulada a tensão de saída do circuito e a sensibilidade do circuito, os resultados podem ser vistos, para diversos valores de R_s na figura 4.13.

No software LTSpice para simular a resposta do circuito à uma onda quadrada. Na figura 4.14 consta a resposta do circuito simulado, com uma frequência de $3,9Hz$. É interessante notar o efeito capacitivo do sensor quando ocorre a mudança de estado da onda quadrada, principalmente para os altos valores de resistência (baixos valores de umidade relativa do ar).

4.3.3 Código e algoritmos

Para filtrar o sinal, será feita a amostragem em dois momentos, quando a onda quadrada estiver em V_{cc} o valor do ADC será armazenado em adc_{high} e quando estiver em $0V$ será armazenado em adc_{low} . Em cada valor obtido existirá também uma sobre-amostragem, feita por hardware, com 16 amostras. Além desta sobre-amostragem, é feita uma outra utilizando 4 ondas em tempos diferentes, duas positivas e duas negativas.

A aquisição das amostras será feita $100ms$ após a mudança de estado da onda quadrada, para não tomar em conta os efeitos produzidos pela capacitância do sensor. Para garantir que o ADC e o *timer* estejam defasados corretamente, é usado um outro *timer*, deslocado em $100ms$ do *timer* do da onda quadrada, que ativa o ADC nos momentos certos.

O funcionamento do código é resumido no fluxograma representado na figura 4.15. O programa começa configurando o ADC, ligando o divisor resistivo do circuito, ligando o ADC, iniciando os timers (tanto da onda quadrada quanto o do ADC), e então a cada interrupção do ADC é guardado o seu valor, e ao final de quatro são feitas duas

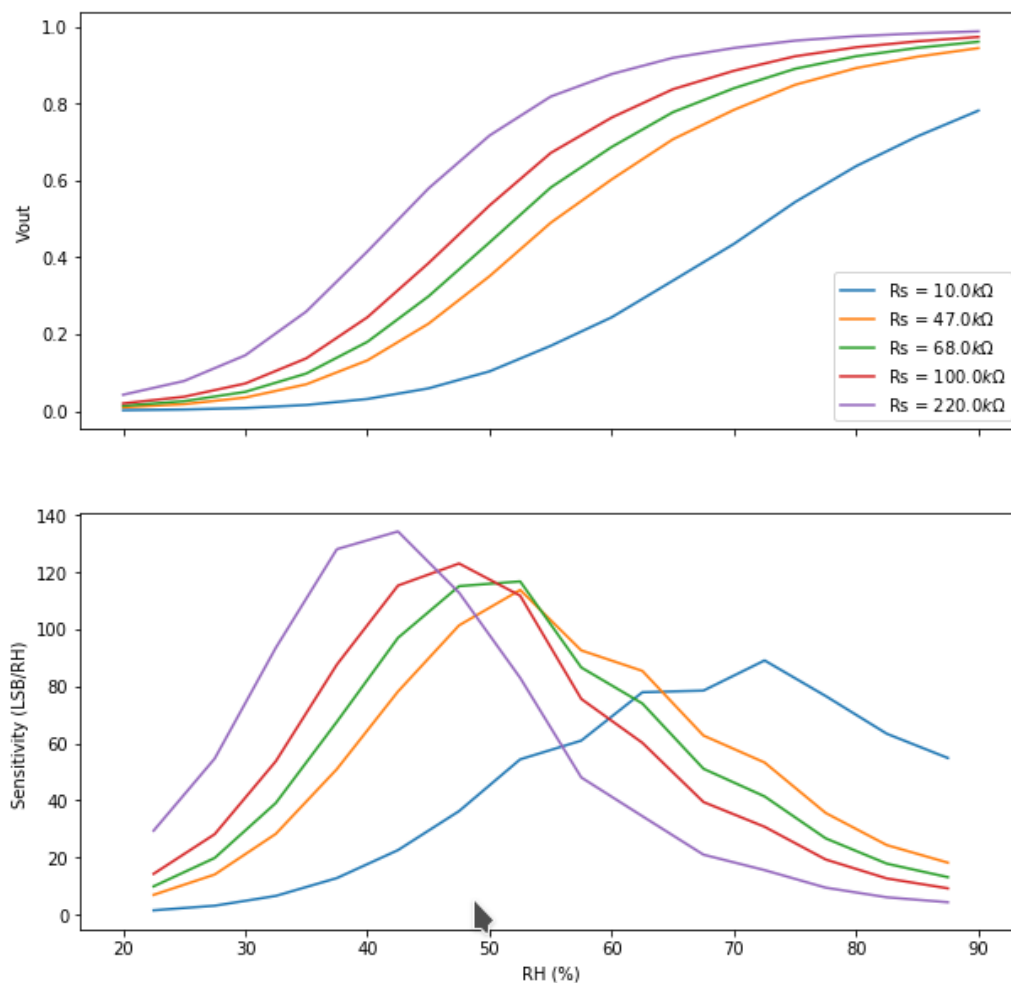


Figura 4.13: Simulações do sensor HR202. Fonte: O Autor(2018)

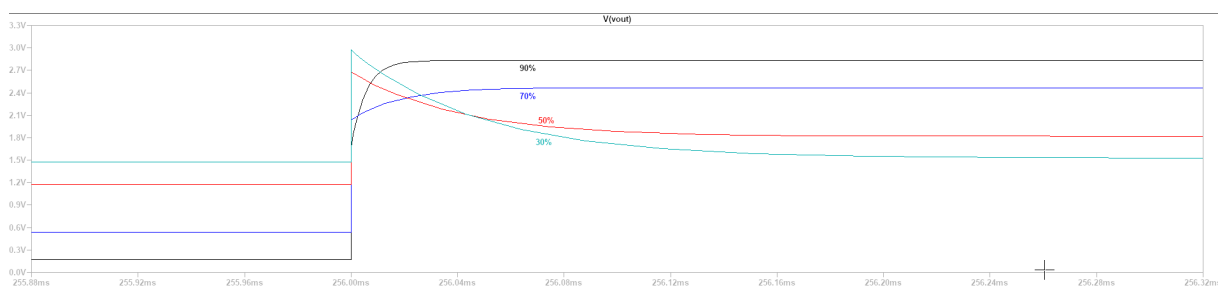


Figura 4.14: Simulação do sensor HR202 no LTSPice. Fonte: O Autor(2018)

média entre elas (uma para quando a onda quadrada está em estado lógico 1, e uma para estado lógico 0), e todos os componentes são desligados.

Dado que o tratamento do sinal é complexo, a placa manda a informação bruta para o *gateway*, que analisa esta informação bruta e converte em umidade do ar relativa.

Decodificação dos valores no gateway

Com as aquisições recebidas, é subtraído destes valores a metade da resolução do ADC (equações 4.11 e 4.12), e então é feita a média entre os dois na equação 4.13, para se obter uma medida que toma em conta os dois casos. Esta parte é feita no *gateway*.

$$M_1 = \frac{adc_{high} - 2^{12-1}}{2^{12} - 1} \quad (4.11)$$

$$M_2 = \frac{adc_{low} - 2^{12-1}}{2^{12} - 1} \quad (4.12)$$

$$V_m/V_{cc} = M1 - M2 \quad (4.13)$$

A equação que relaciona esta saída V_m aos valores dos resistores é dada pela equação 4.14.

$$V_m = V_{out}|_{V_{sq}=1} - V_{out}|_{V_{sq}=0} \quad (4.14)$$

Aplicando análise de circuitos, desenvolvendo a equação 4.14 e isolando R_{RH} , chegamos a equação

$$R_{RH} = R_S \cdot \frac{1 - V_m}{2V_m} \quad (4.15)$$

Uma vez com o valor de R_{RH} , é necessário interpolar a tabela 4.2 para obter o resultado correto.

Tabela 4.2: Valores de umidade do ar para diferentes resistências e temperaturas.
Fonte: ELECROW ()

	0°C	5°C	10°C	15°C	20°C	25°C	30°C	35°C	40°C	45°C	50°C	55°C	60°C
20%RH				10M	6.7M	5.0M	3.9M	3.0M	2.4M	1.75M	1.45M	1.15M	970K
25%RH		10M	7.0M	5.0M	3.4M	2.6M	1.9M	1.5M	1.1M	880K	700K	560K	450K
30%RH	6.4M	4.6M	3.2M	2.3M	1.75M	1.3M	970K	740K	570K	420K	340K	270K	215K
35%RH	2.9M	2.1M	1.5M	1.1M	850K	630K	460K	380K	280K	210K	170K	130K	150K
40%RH	1.4M	1.0M	750K	540K	420K	310K	235K	190K	140K	110K	88K	70K	57K
45%RH	700K	500K	380K	280K	210K	160K	125K	100K	78K	64K	50K	41K	34K
50%RH	370K	26K	200K	150K	115K	87K	69K	56K	45K	38K	31K	25K	21K
55%RH	190K	140K	110K	84K	64K	49K	39K	33K	27K	24K	19.5K	17K	14K
60%RH	105K	80K	62K	50K	39K	31K	25K	20K	17.5K	15K	13K	11K	9.4K
65%RH	62K	48K	37K	30K	24K	19.5K	16K	13K	11.5K	10K	8.6K	7.6K	6.8K
70%RH	38K	30K	24K	19K	15.5K	13K	10.5K	9.0K	8.0K	7.0K	6.0K	5.4K	4.8K
75%RH	23K	18K	15K	12K	10K	8.4K	7.2K	6.2K	5.6K	4.9K	4.2K	3.8K	3.4K
80%RH	15.5K	12.0K	10.0K	8.0K	7.0K	5.7K	5.0K	4.3K	3.9K	3.4K	3.0K	2.7K	2.5K
85%RH	10.5K	8.2K	6.8K	5.5K	4.8K	4.0K	3.5K	3.1K	2.8K	2.4K	2.1K	1.9K	1.8K
90%RH	7.1K	5.3K	4.7K	4.0K	3.3K	2.8K	2.5K	2.2K	2.0K	1.8K	1.55K	1.4K	1.3K

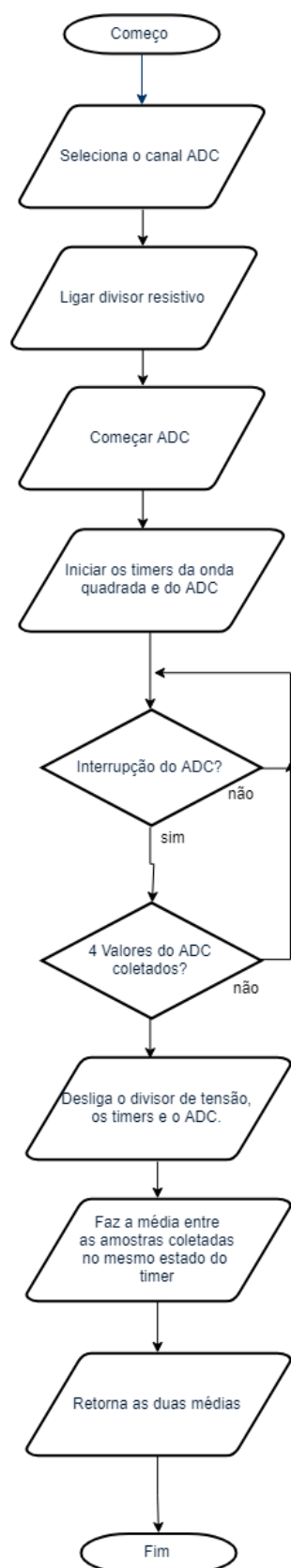


Figura 4.15: Fluxograma do funcionamento do HR202. Fonte: O Autor (2018)

A interpolação foi feita no python com a função *interp1d* da biblioteca *scipy*, a temperatura considerada no cálculo é arredondada para os $5^{\circ}C$ mais próximos. Esta função faz a interpolação linear dos valores.

4.4 Modulo de comunicação LoRa

O módulo escolhido é o RA-01, que tem como chip o SX1278 e transmite numa frequência de $433MHz$.

4.4.1 Circuito

O módulo LoRa RA-01 foi conectado com SPI, como mostra na figura 4.16

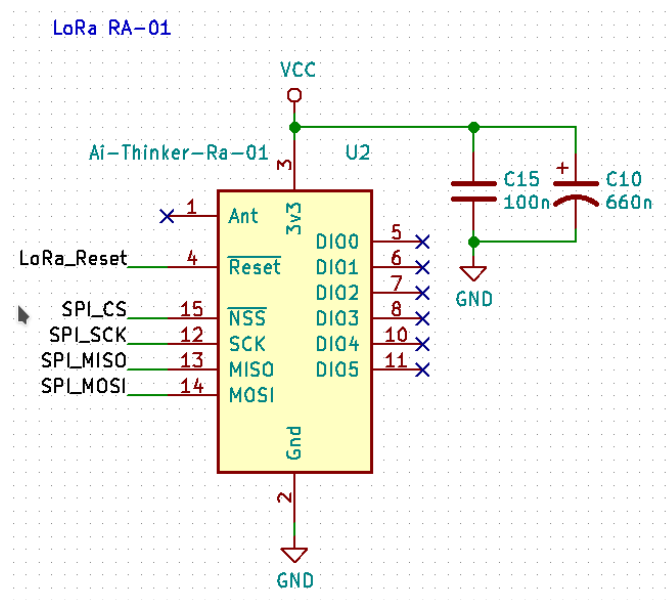


Figura 4.16: Esquemático do módulo LoRa. Fonte: O Autor(2018)

Existem alguns parâmetros que podem ser mudados com o intuito de ajustar a transmissão para atender as necessidades do projeto.

Spreading Factor

O *spreading factor* é um parâmetro que dita quantos símbolos são mandados para cada bit de informação. Quanto mais símbolos, mais sensibilidade ao ruído terá o receptor, menor será a velocidade da transmissão e maior será o consumo de energia, como mostra a figura 4.17.

Como neste projeto não é necessário uma alta taxa de transmissão e se preza alcance sobre isso, logo o *spreading factor* escolhido é 12. Mesmo com um consumo de energia mais alto, ele continua dentro do aceitável como será discutido na seção 5.6.

<i>SpreadingFactor</i> (RegModulationCfg)	Spreading Factor (Chips / symbol)	LoRa Demodulator SNR
6	64	-5 dB
7	128	-7.5 dB
8	256	-10 dB
9	512	-12.5 dB
10	1024	-15 dB
11	2048	-17.5 dB
12	4096	-20 dB

Figura 4.17: Mostra os *spreading factors* e o a quantidade de símbolo por bit de cada um e a sensibilidade necessária no receptor de cada um. Fonte: SEMTECH (2015)

Bandwidth (kHz)	Spreading Factor	Coding rate	Nominal Rb (bps)
7.8	12	4/5	18
10.4	12	4/5	24
15.6	12	4/5	37
20.8	12	4/5	49
31.2	12	4/5	73
41.7	12	4/5	98
62.5	12	4/5	146
125	12	4/5	293
250	12	4/5	586
500	12	4/5	1172

Figura 4.18: Relaciona a largura de banda e a velocidade de transmissão. Fonte SEMTECH (2015).

Largura de banda

A largura de banda influencia na velocidade de transmissão, como pode ser visto na figura 4.18.

Para diminuir o tempo de transmissão, será utilizada uma largura de banda de 125kHz , uma largura de banda mais alta diminui também as chances de assincronia entre o receptor e o transmissor, causado por erros no oscilador.

Header

Existem dois tipos de *headers* disponíveis, o implícito e o explícito. No explícito o tamanho da *payload*, e a presença de um CRC são especificados, no implícito, para diminuir a informação para transmitir, o tamanho da *payload* e a presença de um CRC são definidos no transmissor e receptor. Como se trata de um protótipo e as informações podem mudar, por ora foi escolhido um *header* explícito, num produto final seria ideal utilizar um *header* implícito.

4.4.2 Corpo de dados

O corpo de dados (*payload*) é composto da informação dos três sensores, estas informações estão na tabela 4.3. Estes são só os dados da transmissão, na figura 4.19 se encontram todos os dados que serão transmitidos no pacote. Serão transmitidos 10 bytes de dados.

Tabela 4.3: Tamanho dos dados de cada sensor. Fonte: O Autor (2018)

Sensor	Tamanho do dado (bits)
NTC	12
MS	16
HR202	24
Número da mensagem	14
Número do sensor	10
Total	76

Desta maneira, o sistema suportaria 1024 sensores, e cada um teria um número de mensagem único por aproximadamente dois anos.

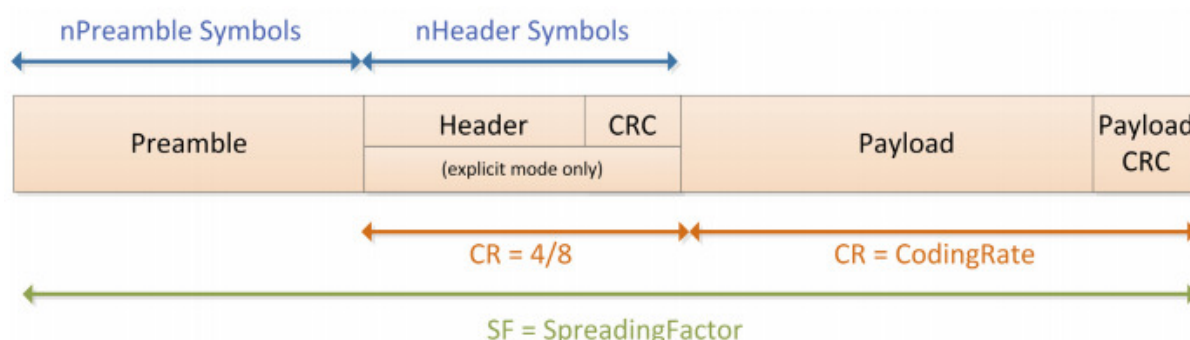


Figura 4.19: Conteúdo do pacote de transmissão LoRa. Fonte: SEMTECH (2015).

4.4.3 Antena

As antenas utilizadas, tanto para o *gateway* quando para o sensor, são as fornecidas junto com o módulo, como pode ser visto na figura 4.20. Principalmente para o *gateway*, seria interessante mudar de antena para melhorar o alcance. O tipo da antena a ser usada depende muito do uso específico e do local onde será instalada a antena dentro da fazenda. Se todos os campos estiverem só de um lado da casa, uma antena diretiva como a Yagi pode ser escolhida, se não uma antena dipolo de meia onda poderia ser usada também.

Se for necessário um alcance muito grande, poderá ser usado um amplificador também.

4.4.4 Implementação no código

A comunicação com o módulo é feito por SPI, o comando é feito escrevendo em registros dentro do módulo as configurações ou dados a serem enviados. Os *drivers*



Figura 4.20: Antena do módulo LoRa. Fonte: O Autor (2018)

para o módulo SX1278 se baseiam num código por Wojciech Dowski, disponível em <https://github.com/wdomski/SX1278>. O código foi modificado para se adaptar ao hardware em questão e foram feitas mudanças em algumas funções com a finalidade de diminuir o circuito externo do módulo.

4.5 Bateria

A bateria é uma parte essencial para o funcionamento do projeto, se a bateria perde muita tensão com a perda de carga, ela pode não fornecer tensão suficiente para alimentar os componentes, o microcontrolador necessita $1,8V$ para funcionar, o *datasheet* do chip LoRa SX1278 menciona o mesmo valor como valor mínimo, os três sensores não são sensíveis a mudanças de tensão e funcionam desde que o ADC do microcontrolador esteja funcionando. Logo é necessária uma bateria que forneça a maior parte da sua carga antes de ficar com tensão inferior à $1,8V$.

As principais baterias levadas em consideração foram:

4.5.1 Bateria de íon-lítio

Muito usada para os mais diversos aparelhos eletrônicos - desde celulares até baterias de carros elétricos - por ter uma grande capacidade, por poderem suprir muita corrente e por serem recarregáveis. Porém uma célula de uma bateria como a 18650 pode começar com uma tensão de até $4V$, segundo o *datasheet* EEMB (2010), que passa do limite tolerado pelo microcontrolador escolhido (STM32L051) de $3,6V$. Além disso para recarga este tipo de bateria demanda um circuito específico e complexo. A bateria de polímero de lítio possui uma desvantagem parecida com a de íon-lítio, com uma tensão máxima passando de $4,2V$ ultrapassando a tensão máxima permitida pelo do microcontrolador.

4.5.2 Pilha AA Alcalina

Uma bateria alcalina tem uma tensão nominal de $1,5V$, variando entre $1,5V$ e $0,8V$. Se duas estiverem em série, a tensão fica entre $3V$ e $1,6V$, onde uma grande parte fica dentro da faixa de funcionamento do microcontrolador, sem precisar de circuito adicional para regular a tensão. A pilha alcalina foi escolhida, devido a sua carga

maior e uma auto-descarga menor comparada com uma bateria AA de zinco-carbono tradicional.

4.6 Microcontrolador

O microcontrolador STM32L051 foi escolhido pois a arquitetura ARM Cortex-M possui alta capacidade de processamento, associada a um baixo custo e baixo consumo de energia (linha M0+).

Como este projeto visa fazer um protótipo diretamente em uma PCB, o microcontrolador foi soldado diretamente na placa desenvolvida na seção 4.8.

O microcontrolador possui inúmeros periféricos e funções essenciais para este projeto, sendo que os módulos utilizados neste projetos serão descritos na sequência.

O esquemático do circuito, incluindo as conexões do microcontrolador, está disponível no anexo A.

4.6.1 Timers

O STM32L051 contém 9 *timers*, este periférico é usado em dois sensores, o sensor de umidade do solo e do ar.

Sensor de umidade do solo

Este sensor usa o *timer 2*, em modo *output compare* para gerar uma onda quadrada de período $15ms$.

Sensor de umidade relativa do ar

Usa dois *timers*, o 21 em modo *output compare* para gerar uma onda quadrada de período de $256ms$, e o 22 que é idêntico ao 21 mas está deslocado de $100ms$ para acionar o ADC.

4.6.2 ADC

Usado pelos três sensores, o *ADC* do STM32L051 tem resolução de 12 bits, frequência de amostragem de até 1.14 Msps e 16 canais, com um mínimo de 1,65 V.

Outra característica deste periférico é que ele pode fazer sobre-amostragem por hardware, diminuindo a carga do *MCU*.

4.6.3 Comparador

O periférico do comparador é usado para o sensor de umidade do solo, quando o sinal do circuito RC for maior que um valor fixo dado por uma ponte resistiva, o comparador vai ter como saída 1, no caso contrario será zero.

4.6.4 Interfaceamento

O interfaceamento do chip *LoRa SX1278* é feito por meio de comunicação SPI, o microcontrolador conta com um periférico SPI.

4.6.5 Osciladores

O circuito conta com dois osciladores, um de alta velocidade com 16 MHz, e outro para o RTC de 32,768 kHz. Depois de todos os testes será avaliado se estes são necessários, pois se o oscilador interno for preciso o suficiente, uma boa parte da PCB ficaria livre e seria possível fazê-lo menor.

4.6.6 RTC

O *MCU* contém um *real time clock* integrado. Ele aceita como *clock* de entrada um oscilador interno de 37kHz, ou um oscilador externo de baixa frequência, normalmente de 32,768kHz. O *RTC* é muito importante pois é ele que tira o *MCU* do modo de baixo consumo.

É programado um alarme para acordar o *MCU* a cada hora.

4.6.7 Modos de baixo consumo

Existem cinco modos de baixo consumo no STM321051: *LPRun*, *Sleep*, *LPSleep*, *Stop* e *Standby*.

A tabela 4.4 resume os cinco modos de baixo consumo. O modo que consome menos energia é o *Standby*, seguido pelo *Stop*. No modo *Stop* a *SRAM* e o estado dos *GPIOs* são mantidos, o *core* é pausado, a execução do programa se reinicia após uma interrupção vinda do *RTC*.

No modo *Standby*, a *SRAM* é desenergizada e os *GPIOs* entram em alta impedância para economizar energia. Como a *SRAM* não guarda os conteúdos, todos os valores guardados nela são perdidos,

Tabela 4.4: Tabela resumindo os modos de baixo consumo de energia, de melhor (1) até pior (5). Fonte: Mielke (2017)

	LPRun	Sleep	LPSleep	Stop	Standby
Performance	1	2	3	4	5
Power Consumption	4	5	3	2	1
Wake-up Sources	1	2	3	4	5
Wake-up Time	2	1	4	3	5

O modo escolhido para este projeto é o *Stop*, que tem um consumo muito baixo de energia e guarda os conteúdos da *SRAM*.

4.6.8 Botões, LEDs e SWD

Foi incluído um botão de *reset*, para facilmente resetar o microcontrolador caso seja necessário. Dois LEDs foram incluídos na PCB, um é ligado diretamente na alimentação, para mostrar que a alimentação está correta (este será futuramente retirado para diminuir o consumo de energia) e outro está ligado em um *GPIO* para uso geral, como por exemplo para mostrar quando o dispositivo está transmitindo.

Foi feita uma conexão SWD para que o microcontrolador possa ser programado com um dispositivo *ST Link*.

4.7 Firmware

A figura 4.21 exemplifica a interação entre os diferentes módulos do *firmware*. A informação entra pelos três sensores, é tratada pelo agendador de tarefas (*scheduler*) e então encaminhada para o módulo *LoRa*.

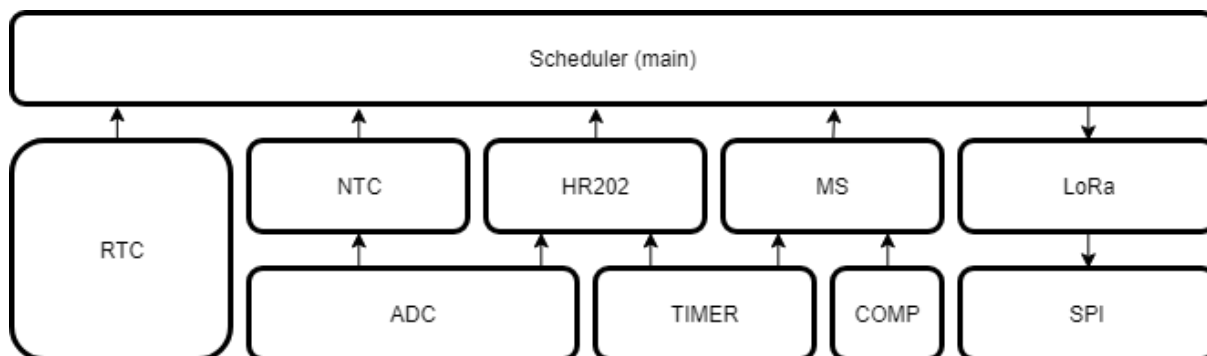


Figura 4.21: Diagrama da interação entre os módulos. Fonte: O Autor (2018)

O código para cada módulo ou sensor foi explicado na própria seção dedicada a cada sensor, todos os códigos foram escritos de forma modular, para manter boas praticas de programação e para evitar *bugs*.

A função *main* do *firmware* consiste em inicializar todos os periféricos, coletar as informações dos três sensores, enviar as informações via *LoRa*, e entrar em modo de baixo consumo até que o *RTC* acorde o *MCU* para recomençar este processo.

4.8 Desenvolvimento da PCB

O desenvolvimento da PCB foi feito no software KiCAD. O esquemático geral do circuito está presente no anexo A.

Para o roteamento da PCB foi tomado muito cuidado para que esta possa ser feita em casa, com métodos caseiros, para poder fazer um protótipo rapidamente. A placa tem duas camadas, mesmo sendo possível fazer em uma, pois o sensor de umidade do solo tem uma sensibilidade bidirecional se ele se estende nas duas camadas, além de aumentar a sensibilidade.

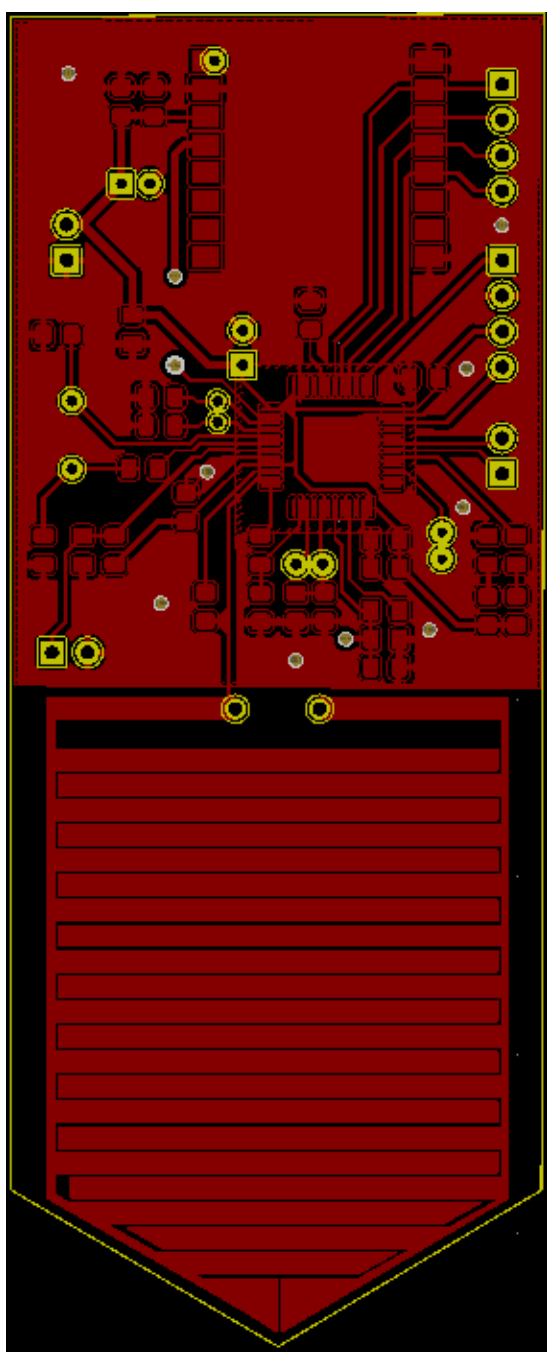
O roteamento foi feito tentando preservar ao máximo o plano de massa para melhorar o desempenho da placa. O layout da placa, pode ser visto na figura 4.22

4.9 Gateway LoRa

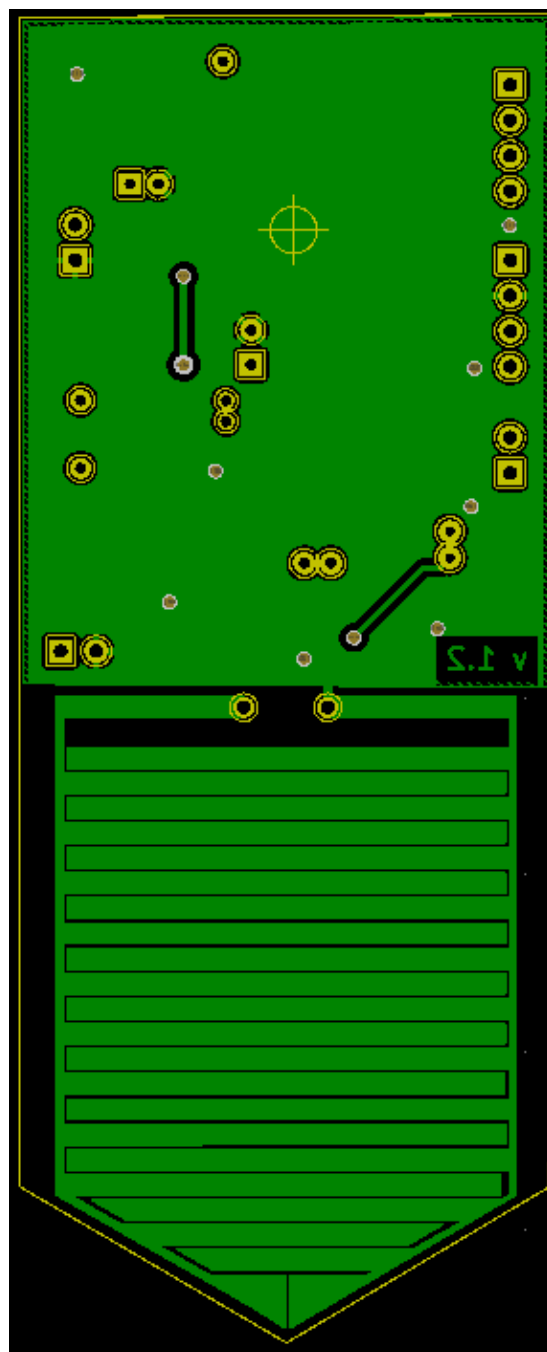
O *gateway LoRa* consiste em um módulo LoRa SX1278, ligado ao microcontrolador STM32F103 via *SPI*, e este ligado à uma *SBC Raspberry Pi* via *UART* para processamento, armazenamento e divulgação dos dados na rede local.

O diagrama da figura 4.23 representa estas ligações.

Uma vez o dado adquirido e processado (como explicado nas seções dedicadas a cada sensor), os dados são armazenados num banco de dados *SQLite*. As colunas deste banco de dados são: o número da mensagem geral (entre todos os

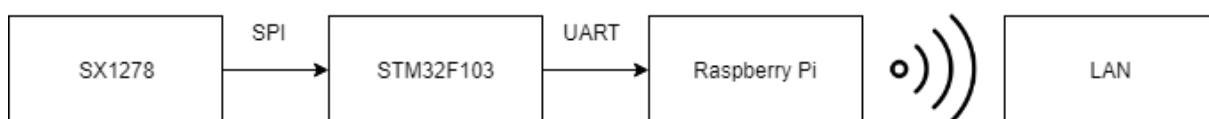


(a) Camada superior da PCB.



(b) Camada inferior da PCB.

Figura 4.22: Layout da PCB. Fonte: O Autor (2018)

Figura 4.23: Diagrama de conexões do *gateway*. Fonte: O Autor (2018)

dispositivos), a data e hora de recebimento, a temperatura, a umidade do ar e do solo, o número da mensagem deste dispositivo, e o identificador do dispositivo.

CAPÍTULO 5

RESULTADOS

5.1 Placa de circuito impresso

A PCB foi desenvolvida com o método fotossensível, pois este é mais preciso que o por transferência térmica, já que a placa contém um chip com espaçamento de $0,5mm$, com trilhas e espaçamento de $0,25mm$. Este método consiste em aplicar um filme sobre a placa de cobre, cobrir a placa de cobre com o fotolito onde esta impresso o layout do circuito, logo é usada uma lâmpada ultravioleta para revelar as partes que devem ficar com cobre. Após a revelação dos dois lados, é inserida a placa em uma solução para retirar a parte que não foi revelada, e então imersa em percloroeto de ferro para retirar o cobre, e finalmente em uma outra solução para retirar a tinta revelada.

5.1.1 Versões de hardware da PCB

Foram feitas algumas versões de hardware diferentes, as versões foram evoluindo de acordo com dificuldades para fazer a placa ou por causa de erros na placa. A figura 5.1 mostra estas placas.

A ultima versão é a versão v1.2 (figura 5.2).

5.2 Sensor de temperatura

A placa faz a aquisição dos sinais e os envia em dados brutos, ou seja, a leitura direta do ADC (depois de feita a sobre-amostragem).

A aquisição é feita rapidamente, é energizado o sensor, se faz a aquisição pelo ADC (com sobre-amostragem por hardware), e então tudo é desligado, como se pode ver na figura 5.3.

No *gateway* for escolhida a equação 4.2, polinomial de terceira ordem para relacionar a tensão de saída com a temperatura.

A figura 5.10 mostra o resultado da medida de temperatura no banco de dados do *gateway*.

5.2.1 Calibração

A calibração do sensor de temperatura foi feita adquirindo medidas de resistência do sensor em várias temperaturas conhecidas, as medidas foram feitas em um recipiente com água e gelo, na temperatura ambiente e em um recipiente com água em ebulição, com a ajuda de um termômetro calibrado. Os resultados estão presentes na tabela 5.1. Utilizando a equação 5.1 e os dados coletados, é calculada a constante B para cada par de dados, e finalmente a constante escolhida para calibração é a média entre os três valores de B calculados.



Figura 5.1: Versões de hardware anteriores do projeto. Fonte: O Autor (2018)



Figura 5.2: Última versão da placa, v1.2. Fonte: O Autor(2018)

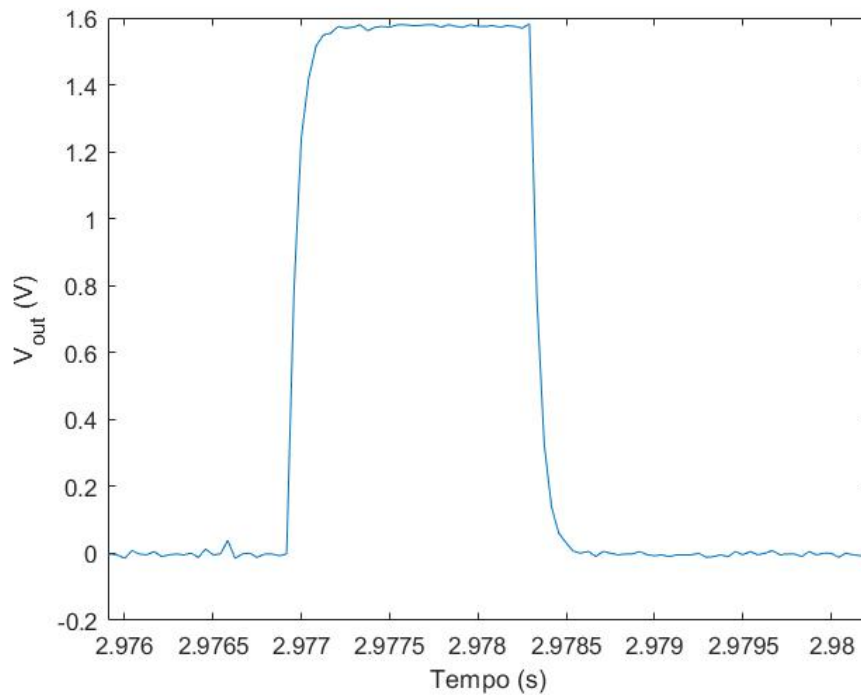


Figura 5.3: Tensão de saída na aquisição da temperatura. Fonte: O Autor (2018)

Tabela 5.1: Informações sobre a calibração do sensor NTC. Fonte: O Autor(2018)

T (C)	T (K)	R (Ohms)	B (K)
0	273.15	31570	3931.76320918964
19	292.15	12380	4073.91862681217
82.1	355.25	1040	4033.91475238639
		Média	4013.19886279607

$$B = \frac{\ln(R_{T1}/R_{T2})}{1/T_1 - 1/T_2} \quad (5.1)$$

Usando esta calibração e a equação 4.2 para fazer a transição entre a tensão de saída e a temperatura, obtemos os erros apresentados na tabela 5.2, o erro é alto para altas temperaturas, mas como estas não serão usadas, isto não é um problema.

Tabela 5.2: Erro do sensor de temperatura nas temperaturas de calibração. Fonte: O Autor (2018)

T (C)	Erro (C)
0	1,65
19	0,52
82.1	-5,47

5.3 Sensor de umidade do solo

Na figura 5.4 se encontra a tensão de saída no pino de entrada, onde está o circuito RC, com um capacitor de $470pF$, e o resistor de $240k\Omega$. O sensor faz oito medidas no total, quando o sinal muda de estado.

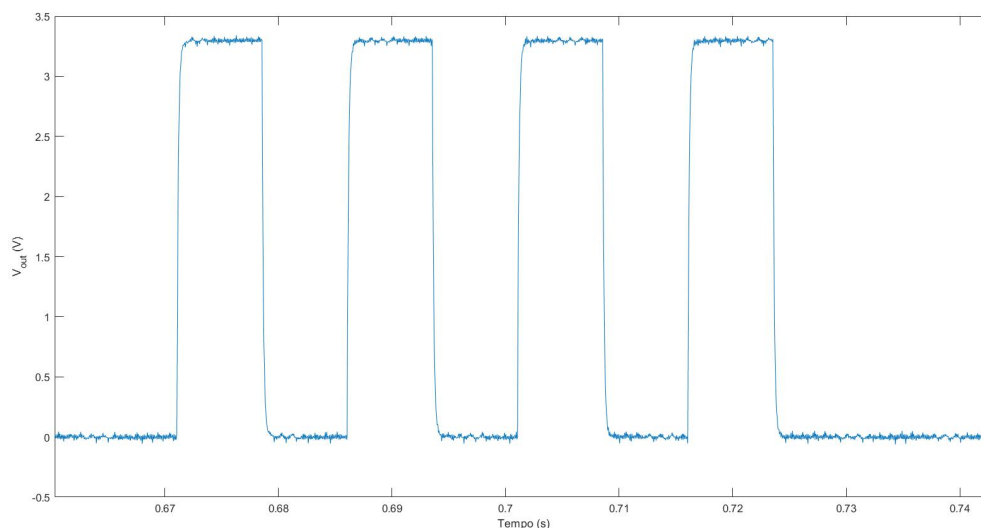


Figura 5.4: Tensão de saída do sensor de umidade do solo, teste com o capacitor. Fonte: O Autor (2018)

A figura 5.5 mostra a o resultado da medição de umidade do solo. O sensor está inserido dentro do solo, na primeira e na segunda linha, se pode ver que a umidade do solo (quinta coluna) está baixa, então um pouco de água é adicionada a terra perto do sensor e a umidade aumenta significativamente (terceira linha), e depois de um tempo a água se espalha pela terra, diminuindo um pouco a umidade do solo perto do sensor (quarta linha).

id	currentdate	temperature	air_humidity	soil_humidity	msg_nb	devid
272	2018-11-26 14:51:19	22.2460368416555	48.571972098922	3.70737845405529	141	1
273	2018-11-26 14:52:21	22.5182692271649	45.902533640441	3.70737845405529	142	1
274	2018-11-26 14:53:23	21.8260650306011	48.245065789473	18.3597499630832	143	1
275	2018-11-26 14:54:25	22.2213078623169	48.450261780104	11.9728187924811	128	1

Figura 5.5: Demonstração do aumento da grandeza medida quando é colocado mais água na terra onde se encontra o sensor. Fonte: O Autor (2018)

5.3.1 Calibração

Para confirmar que o circuito proposto mede corretamente a capacitância, foram usados capacitores com valores conhecidos para testar a performance do sistema. A tabela 5.3 contém dados dos vários capacitores testados, o valor nominal deles e o valor encontrado pelo algoritmo.

Se o valor obtido for subtraído dos valores a vazio, quando nada está conectado (ou seja, a capacitância parasita), se obtém valores muito próximos dos

Tabela 5.3: Resultado do teste do sensor de umidade do solo com capacitores conhecidos. O valor "count" se refere ao valor obtido pelo sensor subtraído do sensor em aberto. Fonte: O Autor (2018)

Capacitor	Count	C (F)
Aberto	0	0
2,2p	2	1,50E-12
5p	5	3,76E-12
15p	20	1,50E-11
220p	293	2,20E-10
470p	629	4,73E-10
1 n	1342	1,01E-09
10 n	12945	9,73E-09

valores nominais de capacitância, e dentro da tolerância dos capacitores, como visto na tabela 5.3.

Para a calibração do sensor em si, é usado o método gravimétrico, que consiste em três etapas:

1. Pesar as amostras do solo.
2. Colocar amostras em uma estufa por 24-48h em uma temperatura entre 95 C e 105 C.
3. Pesar as amostras novamente.

Como as amostras ficaram na estufa por muito tempo, a água evapora e a diferença entre as massas medidas era em água, e então é possível calcular a umidade do solo com a equação 5.2, onde M_m é a massa da amostra molhada, M_s é a massa da amostra seca, e M_r é a massa do recipiente.

$$MS = \frac{M_m - M_s}{M_s - M_r} \quad (5.2)$$

Este método foi seguido, com três amostras (T1, T2 e T3), para calibrar o sensor, o resultado está na tabela 5.4.

Tabela 5.4: Medidas para a calibração do sensor de umidade do solo via método gravimétrico. Fonte: O Autor(2018)

Amostra	T1	T2	T3	Água
Mr (g)	4	5	6	-
Mm (g)	24	29	29	-
Ms (g)	18	20	20	-
Cont	935	1248	1578	2313
C (pF)	702.56	937.75	1185.71	1737.99
MS (%)	42.86	60	64.29	101.7*

À partir destas medidas, é possível utilizar fazer uma regressão linear para obter a equação que relaciona a capacitância com a umidade do solo. A figure 5.6

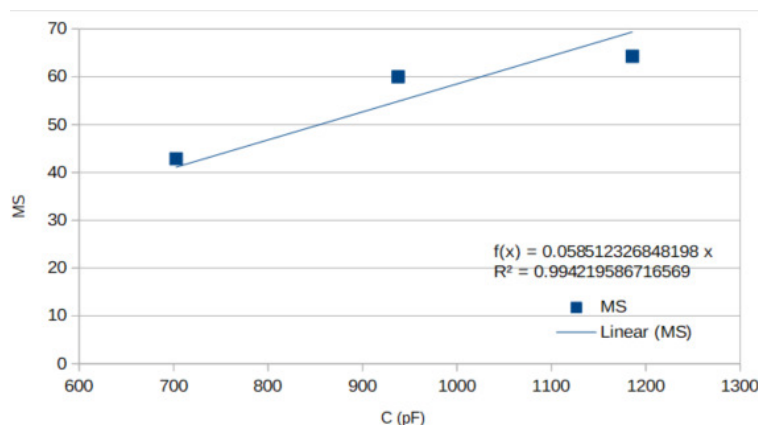


Figura 5.6: Regressão linear das medidas obtidas para obtenção da equação que relaciona capacitância do sensor com umidade do solo. Fonte: O Autor (2018)

mostra este cálculo feito no Libreoffice Calc, de onde se pode tirar a equação (5.3) que relaciona a capacitância com a umidade do solo, na equação, C é medido em pF , esta equação apresenta um coeficiente de determinação (R^2) de 0,994.

$$MS = 0,0585 \cdot C \quad (5.3)$$

Na tabela 5.4 também foi feito um teste com água, e foi calculada a umidade do solo a partir da medida da capacitância. Como é esperado, o valor é muito próximo de 100%.

Lembrando que estes valores só servem para o tipo de solo testado, para outro tipo de solo seria necessário fazer uma recalibração.

5.4 Sensor de umidade relativa do ar

A placa faz a aquisição dos sinais e os envia em dados brutos, ou seja, diretamente a leitura do ADC (depois de feita a sobre-amostragem).

A figura 5.7 mostra a tensão de saída quando o sensor é acionado. Como foi mencionado na parte 4.3, o ADC tem que ser sincronizado com o *timer* que comanda a onda quadrada, na figura 5.7 é possível ver quando o ADC termina sua conversão. Desta maneira as medidas do ADC não são afetadas pelas componentes capacitivas do sensor.

Existe um pequeno *overshoot* devido a capacitância, porém é um valor aceitável, quando a resistência ou capacitância do sensor aumentarem, este *overshoot* e o tempo que demora para ficar estável podem mudar, como pode ser visto em uma outra aquisição feita (figura 5.8).

A figura 5.10 mostra o resultado da medida de umidade relativa do ar, onde a medida já foi processada e armazenada num banco de dados no *gateway*.

5.4.1 Calibração

Para que o sensor seja calibrado, é necessária a obtenção de um valor conhecido de umidade do ar. Para isto, dentro de um recipiente fechado foi colocada uma solução

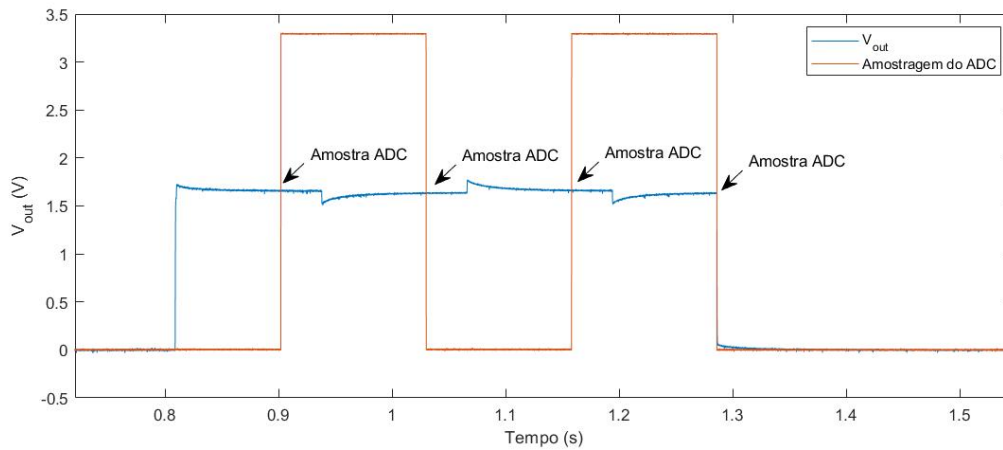


Figura 5.7: Tensão de saída na aquisição da umidade relativa do ar. Fonte: O Autor (2018)

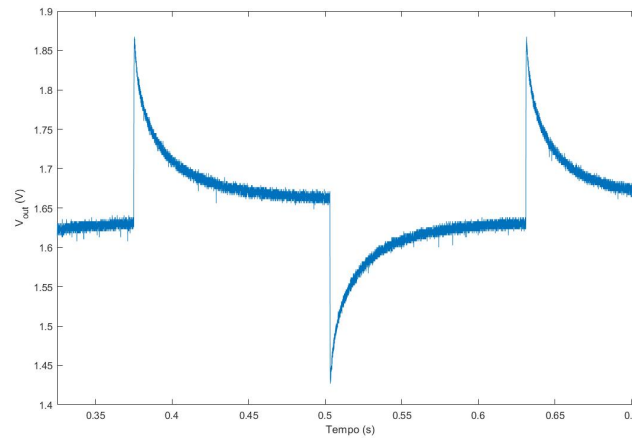


Figura 5.8: Tensão de saída quando o efeito da capacitância é mais forte. Fonte: O Autor (2018).

saturada de sal (sal com algumas gotas de água), que fazem com que a umidade relativa do ar com o tempo seja 75%. Além disso foi usado um outro sensor de umidade do ar (este já calibrado), para obter mais pontos de comparação.

Em um primeiro momento, o sensor de umidade do ar do dispositivo não alegou uma umidade do ar de 75% dentro do recipiente, porém o sensor de referência mede exatamente o valor esperado, logo este está calibrado.

Para compensar o sensor do dispositivo, foram retirados do recipiente os dois sensores e anotados novamente as medidas feitas, estas medidas e as medidas do recipiente estão presentes na tabela 5.5, e com estes dados da tabela foi feita uma equação para relacionar o valor alegado pelo sensor do dispositivo com o valor real, que está presente na equação 5.4.

$$RH_{real} = 2RH_{disp} - 18,12 \quad (5.4)$$

Após a calibração, foi feita uma nova tentativa para medir a umidade relativa do ar quando o dispositivo é colocado dentro do recipiente com a solução de sal saturada.

Tabela 5.5: Valores para calibração do sensor de umidade relativa do ar. Fonte: O Autor(2018)

	Ambiente	Recipiente
Valor medido	42.2	46.715
Valor real	66	75

O resultado está na figura 5.9, que mostra a evolução da variável em função do tempo até ela se estabilizar perto de 75%, o que mostra que o sensor foi calibrado.

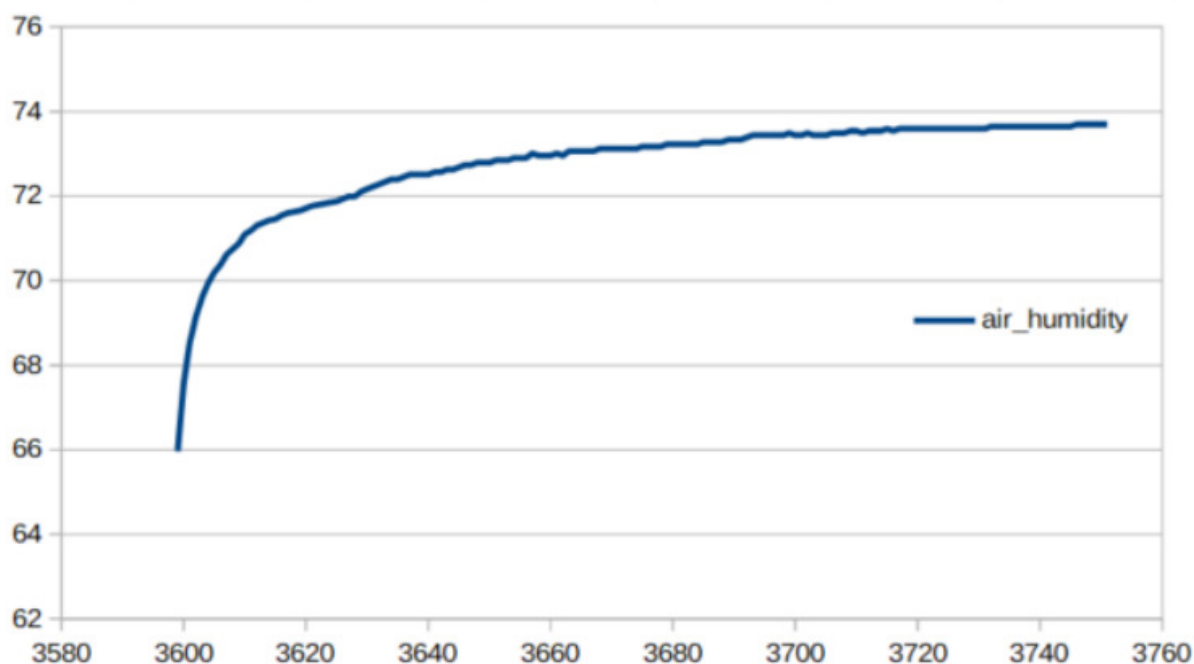


Figura 5.9: Evolução da umidade relativa do ar dentro do recipiente com uma solução de sal saturada, mostra que o sensor está calibrado. Fonte: O Autor(2018)

5.5 Comunicação LoRa

Foi testada o alcance do sistema, o teste foi feito dentro da cidade com muitos prédios e desníveis, logo não representa muito bem a situação do campo, porém mesmo nessas condições adversas o receptor conseguiu receber mensagens em até 1,4km de distância.

O dispositivo para testes manda uma mensagem por minuto, para ajudar na validação e calibração dos resultados, o dispositivo final manda uma mensagem por hora.

5.6 Consumo de energia

A tabela 5.6 mostra os estados mais importantes, seus consumos respectivos e o tempo que o dispositivo fica em cada estado.

Considerando duas bateria de $2000mAh$ cada uma, uma mensagem por hora, e a tabela 5.6, o dispositivo funcionaria sem trocar a bateria por aproximadamente 1 ano e 4 meses (equação 5.5). Neste cálculo é desconsiderada a energia necessária para adquirir as amostras, pois é um valor irrisório comparado com os outros.

Tabela 5.6: Corrente utilizada pelo dispositivo em diferentes modos. Fonte: O Autor (2018)

Estado	Corrente (mA)	Tempo
Transmitindo	70	900 ms
Stop	0,320	1 h

$$d = \frac{4000mAh}{0,320mA \cdot 1 + 70mA \cdot 0,9/3600} = 1,35anos \quad (5.5)$$

5.7 Gateway

Os dados são adquiridos e armazenados num banco de dados (*SQLite*) dentro de um computador de placa única (*Raspberry Pi*). É possível de obter os dados armazenados no banco de dados com comandos *SQL*. A figura 5.10 mostra um exemplo dos dados coletados, onde todos os sensores estão presentes.

```
sqlite> SELECT * FROM SENSORVALS WHERE id>300;
```

id	currentdate	temperature	air_humidity	soil_humidity	msg_nb	devid
301	2018-11-26 15:21:19	24.0358429477047	46.2042394898441	8.59150229039733	202	1
302	2018-11-26 15:22:21	23.7361537773432	45.7349794643516	8.9672041239628	203	1
303	2018-11-26 15:23:23	24.1108605676673	45.8610915107424	10.0943096246574	204	1
304	2018-11-26 15:24:25	23.2877187121289	45.2942526976353	12.7242224596121	205	1
305	2018-11-26 15:25:27	24.0608445182495	45.9231448763251	14.2270297938703	206	1
306	2018-11-26 15:26:29	23.5117757593728	46.1846121087373	14.9784334610013	207	1

Figura 5.10: Valores coletados pelo sensor e enviados para o *gateway* via *LoRa* e armazenados. Fonte: O Autor (2018).

Uma interface web para o banco de dados *SQLite* foi instalada no *gateway*, esta interface se chama *phpLiteAdmin* (<https://www.phpliteadmin.org/>), e ela possibilita a fácil manipulação e visualização dos dados do servidor em qualquer computador da rede local, uma vez que também foi instalado um servidor HTTP (Apache) no *gateway*.

Um exemplo desta interface pode ser visto na figura 5.11

5.8 Preço estimado

O preço foi calculado desconsiderando os custos de resistores e capacitores, na tabela 5.7 se encontram os componentes principais considerados e os seus preços respectivos para baixas quantidades (dez unidades ou menos), o cálculo foi feito com $1USD = 3,74BRL$.

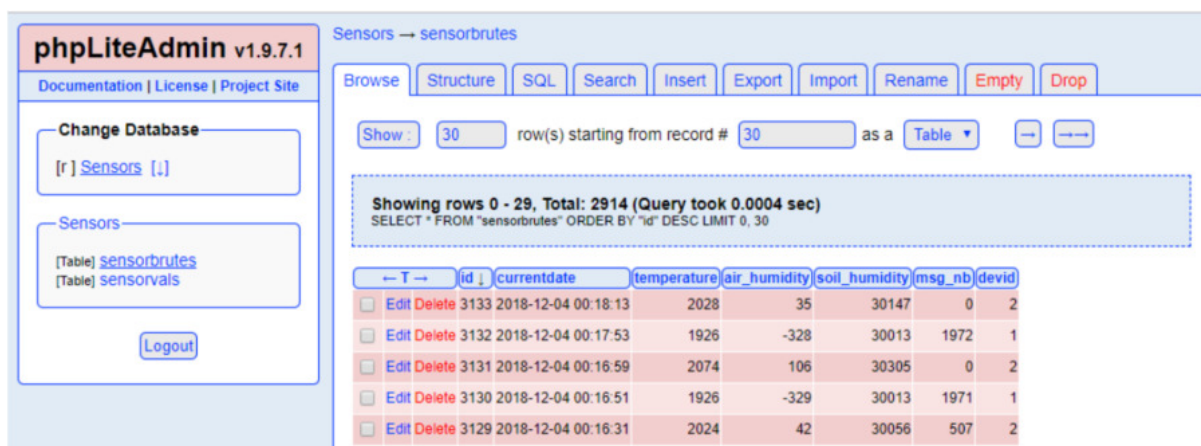


Figura 5.11: Interface *phpLiteAdmin* para o banco de dados *SQLite*. Fonte: O Autor (2018)

Tabela 5.7: Tabela contendo os preços aproximados para a placa.

Componente	Preço (USD)
NTC	0,10
HR202	0,52
MCU	2,71
LoRa	2,52
PCB	0,80
Total (USD)	6,65
Total (BRL)	24,90

CAPÍTULO 6

CONCLUSÃO

Este projeto oferece uma solução de baixo custo e grande alcance e autonomia para a medição de temperatura, umidade do solo e do ar, para ser aplicado nas plantações. O Brasil possui uma grande parte de seu território dedicada a plantações, é essencial a aplicação da tecnologia nesses lugares para o aumento de produtividade. O projeto é *open source* e *open hardware*, e livre para que qualquer pessoa possa utilizá-lo, os arquivos do projeto estão disponíveis em <https://github.com/patricktlo/lotSoilSensor>.

O projeto atinge seu objetivo geral, desenvolvendo um protótipo para mostrar que é possível realizar um produto com o intuito de medir a temperatura, a umidade do solo e do ar, com transmissão por rede Lora, baixo custo do dispositivo e grande autonomia. O projeto foi desenvolvido com o intuito de otimizar esses itens. Foram escolhidos como microcontrolador um STM32L051, como sensor de temperatura um NTC, como sensor de umidade relativa do ar um HR202, como chip para a rede LoRa um SX1278, como bateria duas pilhas AA alcalinas e foi desenvolvido o sensor de umidade do solo na PCB, que não interage com o solo pois faz suas medidas com a capacitância do sensor.

O custo total de cada dispositivo é de aproximadamente 25 reais, e sua autonomia em bateria de aproximadamente um ano e quatro meses.

Além do protótipo do dispositivo foi feito um *gateway* que recebe as informações brutas do sensor, processa, armazena e divulga essas informações por meio de uma base de dados *SQLite*. Esta estrutura é feita com um *Raspberry Pi*, além de um microcontrolador *STM32F103* para interagir com o chip para a rede LoRa, SX1278. O banco de dados tem a capacidade de distinguir cada aparelho e a data em que foi enviada a mensagem.

Sugestões para melhoria do projeto incluem a mudança da antena do *gateway* para uma antena mais eficiente, a adição de sensores úteis para a agricultura como sensor de pH e de nutrientes, escrever um software para que se possa visualizar em um mapa os valores dos diversos sensores, facilitando a visualização e a análise de dados para as pessoas. Para melhorar a precisão do sensor de umidade do ar é possível fazer uma interpolação por splines, tanto da temperatura quando da resistência, e para melhorar a precisão do sensor de temperatura, utilizar a equação de Steinhart-Hart para o NTC ao invés da equação exponencial, e para melhorar ainda mais a vida útil do dispositivo, diminuir o consumo do *MCU* em modo de economia de energia, usando por exemplo o modo *standby*.

BIBLIOGRAFIA

DEAN, R. N. et al. A capacitive fringing field sensor design for moisture measurement based on printed circuit board technology. *IEEE Transactions on Instrumentation and Measurement*, v. 61, n. 4, p. 1105–1112, April 2012. ISSN 0018-9456.

EEMB. *Lithium-ion Battery DATA SHEET*. [S.l.], 1 2010. Rev. 1.

ELECROW. *HR202 Humidity Sensor*. [S.l.].

FIESP, D. *Balança Comercial Brasileira do Agronegócio - Agosto 2018*. 2018. <<http://www.fiesp.com.br/indices-pesquisas-e-publicacoes/balanca-comercial/attachment/file-20180918140623-bca2018/>>. Acesso em 24/09/2018.

GOSWAMI, M. P.; MONTAZER, B.; SARMA, U. Design and characterization of a fringing field capacitive soil moisture sensor. *IEEE Transactions on Instrumentation and Measurement*, p. 1–10, 2018. ISSN 0018-9456.

MIELKE, M. *Low-Power Modes on the STM32L0 Series*. 9 2017. <https://www.digikey.com/eewiki/display/microcontroller/Low-Power+Modes+on+the+STM32L0+Series>.

MIZUGUCHI, J. et al. Fringing field capacitive sensor for measuring soil water content: Design, manufacture, and testing. *IEEE Transactions on Instrumentation and Measurement*, v. 64, n. 1, p. 212–220, Jan 2015. ISSN 0018-9456.

NENOV, T.; IVANOV, S. Linearization of characteristics of relative humidity sensor and compensation of temperature impact. *Sensors and Materials*, v. 19, n. 3, p. 95–106, 2007.

SEMTECH. *SX1276/77/78/79 - 137 MHz to 1020 MHz Low Power Long Range Transceiver*. [S.l.], 3 2015. Rev. 4.

APÊNDICE A ESQUEMÁTICO COMPLETO DO CIRCUITO

