

Sistema de Monitoramento de Plantações com Zigbee e Labview

Santos, M. B.¹ e Cavarsan, F. A.²

¹Marcelo Bispo dos Santos, Engenharia Elétrica com Habilitação em Telecomunicações, santos.marcelob@gmail.com,

²Fábio Aparecido Cavarsan, Mestre em Engenharia Elétrica, UNISAL, fabio.cavarsan@unisal.br

Resumo – A tecnologia tem ajudado muitas pessoas e empresas em seus respectivos ramos de atuação, agricultura e horticultura também estão sendo beneficiados pelos avanços tecnológicos recentes. Este trabalho tem como objetivo o desenvolvimento de um sistema de monitoramento de plantações, para otimização do cultivo agrícola, será possível analisar os dados coletados por meio de um sistema de comunicação sem fio, os quais serão enviados a um software para geração de gráficos e análises futuras, visando assim uma melhoria contínua no processo produtivo e na qualidade da plantação onde o sistema estiver sendo utilizado. O envio dos dados de monitoramento para o software é feito por meio da utilização de módulos Xbee, os quais utilizam a tecnologia Mesh (malha) e Arduino para gerenciar a comunicação entre os módulos e o software.

Palavras-chave: Wireless mesh, redes mesh, monitoramento, redes, mesh.

Abstract – Technology has been helped many people and companies on its respective fields of activity, agriculture and horticulture are also being affected by recent technology advances. This project's objective is to develop a plantation monitoring system for agricultural cultivation optimization, it will be possible to analyze the collected data through a wireless communication system, which will be sent to a software for graphics generation and future analysis, thereby aiming a continuous improvement of the productive process and in the quality of the plantation where the system is being used. The transmission of monitoring data to the software is done with utilization of Xbee modules, which use Mesh technology and Arduino to manage the communication between modules and software.

Keywords: Wireless mesh, mesh networks, monitoring, networks, mesh.

I. INTRODUÇÃO

Com milhares de anos de existência da agricultura, o tempo, a experiência e a necessidade de uma melhor produtividade provocaram mudanças importantes no setor, como a utilização de estufas agrícolas.

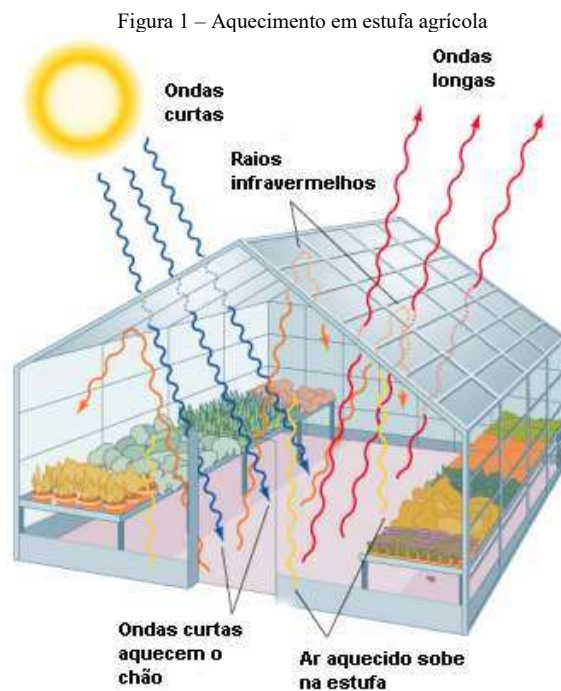
Estas estruturas são utilizadas para reter o calor do sol, protegendo as plantações contra ameaças externas como vento forte e chuvas e mantendo a temperatura

interna controlada de acordo com a entrada de radiação solar, isto acelera os processos de fotossíntese e produção de oxigênio, o que possibilita a melhor frutificação, floração, maior produtividade e qualidade, além de uma maior velocidade no crescimento. Sua utilização se torna cada vez mais comum pois viabiliza a adequação às variações climáticas, evitando seus danos.

As estufas têm como principal função a proteção das plantas e a manutenção de melhores condições para o desenvolvimento do cultivo, é composta de materiais transparentes como vidro ou plástico, que permitem a entrada de radiação solar, que emite radiação infravermelha e aquece o solo (REVISTA AGROPECUARIA, 2019).

Ao atravessar as paredes transparentes de vidro ou plástico, a luz solar (energia radiante) é absorvida por diversos corpos, posteriormente essa energia é emitida na forma de raios infravermelhos que não atravessam o vidro ou plástico, já que estes são materiais opacos para raios infravermelhos, e assim o ambiente interno mantém-se aquecido (STENSMANN, 2002).

A figura 1 demonstra o funcionamento do aquecimento em uma estufa agrícola.



Para um bom desenvolvimento das plantações nas estufas, é necessário o controle de entrada de luz, nível

de umidade na plantação, nível de temperatura, raios UV e todos os outros fatores determinantes para que ela tenha um desenvolvimento de qualidade e com rapidez.

Todos estes controles geralmente são feitos manualmente ou automaticamente, porém não possuem uma maneira de analisar se há mais a ser desenvolvido na qualidade da produção, como por exemplo se a umidade está sendo sempre mantida no nível adequado e se é possível melhorar o processo gerando assim economia de tempo e ganhos financeiros para o produtor (NEGREIRA, 2014).

Este trabalho tem por objetivo possibilitar uma análise aprofundada sobre as plantações, melhorias e aprimoramentos necessários, por meio de um sistema de monitoramento, o qual utilizará a tecnologia Zigbee com *Wireless Mesh* (malha sem fio).

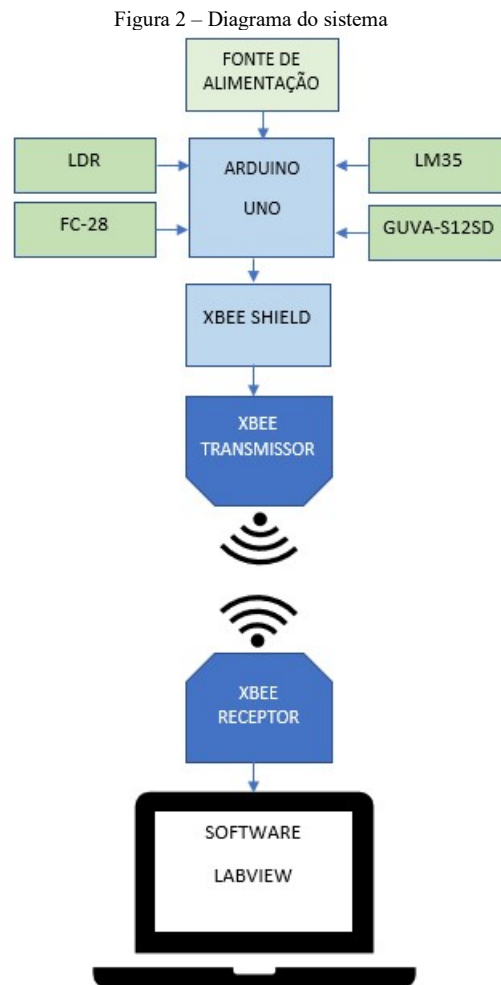
O módulo Xbee transmissor estará conectado a alguns sensores que farão a coleta dos dados (umidade, luminosidade, temperatura e raios UV) em tempo real e encaminharão estas informações ao módulo Xbee receptor, o qual será o coordenador da rede e trabalhará em conjunto com o Arduino, e então o sistema de monitoramento fará a análise dos dados e apresentação ao utilizador.

Desta forma será possível a geração de relatórios e gráficos, para que o produtor possa fazer a análise dos dados e definir planos de ação para o seu negócio, economizando assim recursos como água, energia e tempo, executando melhorias e consequentemente aumentando a qualidade da plantação.

A utilização da tecnologia sem fio em malha também será muito benéfica ao produtor, uma vez que, por não haver utilização de fios, a instalação e manutenção de todo o sistema poderá ser feita com facilidade e a qualquer momento em que se faça necessário modificar a estrutura física do local.

O objetivo final é desenvolver um sistema capaz de atender as necessidades de monitoramento de plantações, de uma maneira completa e simples, facilitando sua utilização e sua implantação em variados tipos de plantações, seja ela de nível pequeno, médio ou grande.

A figura 2 demonstra o diagrama do sistema, onde os dados são coletados nos sensores por meio do Arduino e transmitidos do Xbee transmissor para o Xbee receptor por meio da tecnologia Zigbee, e então apresentado graficamente por meio da aplicação Labview.



Fonte: Acervo do autor

II. REFERENCIAL TEÓRICO

A. Zigbee

O Zigbee é um padrão de comunicação wireless, classificado como WPAN (*Wireless Personal Area Network* - rede pessoal sem fio) e definido pela *Zigbee Alliance*, que é uma associação de empresas de diferentes segmentos, e apesar de ainda não ser muito popular no Brasil, já está sendo utilizado em muitos países. Utiliza a definição 802.15.4 do IEEE para implementar as camadas MAC (*Medium Access Control*) e PHY (*Physical Layer*) e opera em bandas de frequências livres, 2,4 GHz.

Possui muitas qualidades, como o fato de ser altamente confiável, rentável, com baixo consumo de energia e baixa taxa de transmissão, ser altamente seguro e possuir um padrão global. Além disso, sua infraestrutura é mais simples.

Seu alto nível de confiabilidade é pautado em alguns pontos:

- 1) IEEE 802.15.4 com O-QPSK e DSSS

Esta é uma especificação muito robusta e moderna, que usa o O-QPSK (*Offset-Quadrature Phase-Shift Keying*) e o DSSS

(*Direct Sequence Spread Spectrum*), que são uma combinação de tecnologias para prover uma excelente performance em ambientes de baixa relação sinal-ruído.

2) CSMA-CA

O Zigbee utiliza o método de transmissão CSMA/CA (*Carrier Sense Multiple Access / Collision Avoidance*) – Acesso múltiplo com verificação de portadora com prevenção de colisão. Neste método, antes de transmitir o pacote, o dispositivo “ouve” ao canal por um período de tempo para determinar se o canal está livre ou sendo ocupado pela transmissão de outro dispositivo, caso o canal esteja livre então o dispositivo inicia a transmissão, porém se estiver ocupado ele aguardará um período de tempo e reiniciará o processo, isso se repetirá até que o dispositivo consiga efetuar a transmissão de todos os pacotes.

3) CRCs

Em cada pacote é usado a verificação cíclica de redundância CRC (*Cyclic Redundancy Check*) de 16 bits, que é um método de detecção de erros para garantir que os dados sejam entregues corretamente.

4) Reconhecimento a cada etapa

Cada pacote é enviado por até três vezes (para um total de quatro transmissões), caso ele não chegue ao destino após a quarta transmissão o Zigbee informa ao nó transmissor que há um problema.

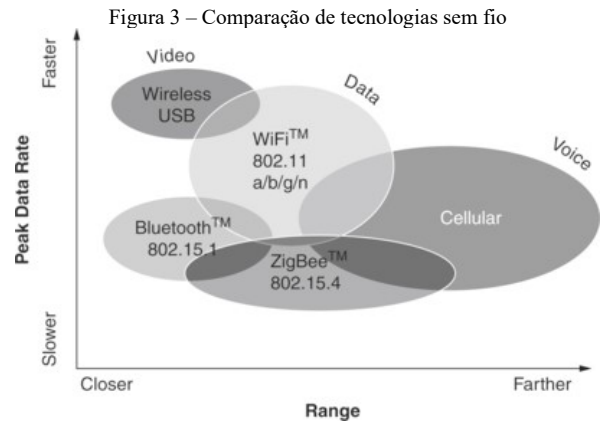
5) Rede em malha

A rede em malha dá confiabilidade para o Zigbee pois possibilita um alcance estendido por meio de redes *ad-hoc* e *multi-hop*, além de descoberta automática de rotas e autocorreção.

6) Reconhecimento de ponta a ponta

Os dados do primeiro nó podem alcançar qualquer outro nó da rede independentemente da distância por meio da topologia de rede em malha, desde que haja módulos suficientes (GISLASON, 2008).

A figura 3 demonstra uma comparação entre as tecnologias sem fio.

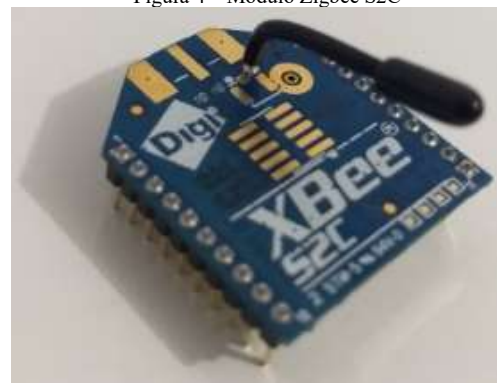


Fonte: GISLASON, 2008

Xbee é o nome dado ao módulo de comunicação da empresa Digi, que utiliza o protocolo Zigbee em sua configuração. O modelo utilizado para este trabalho é o Xbee S2C.

A figura 4 mostra o módulo Zigbee S2C.

Figura 4 – Módulo Zigbee S2C



Fonte: Acervo do autor

a. Tipos de dispositivos

Em uma rede Zigbee há dois tipos de dispositivos, definidos pelo padrão IEEE:

- 1) *Full Function Device* (FFD) – Podem funcionar em qualquer um dos modos de operação (coordenador, roteador ou dispositivo final) e são capazes de se comunicar tanto com os FFD quanto com os RFD. Estes dispositivos têm uma construção mais complexa.
- 2) *Reduced Function Device* (RFD) – São capazes de se comunicar somente com os dispositivos FFD, portanto poderão atuar somente como *end-devices* na rede. Estes são dispositivos mais simples e com menor custo, de forma que seu consumo de energia é reduzido (CARNEVAL et al., 2010).

b. Funções lógicas

Cada dispositivo pode ter três funções lógicas de acordo com sua posição na rede e o tipo de dispositivo (FFD ou RFD):

- 1) Coordenador: Responsável pela distribuição dos endereços, inicialização e manutenção da rede, reconhecimento dos nós, entre outras funções, podendo servir também como ligação entre outras redes Zigbee. Deve ser um dispositivo FFD.
- 2) Roteador: Possui todas as características de um nó qualquer da rede, mas com a função extra de também funcionar como roteador intermediário entre nós, sem precisar do Coordenador. Por intermédio dele uma rede Zigbee pode ser expandida e assim ter mais alcance. Deve ser um dispositivo FFD.
- 3) *End-device* (dispositivo final): É o nó que consome menos energia em toda a rede pois na maioria das vezes fica em espera, este é o dispositivo onde os atuadores e sensores serão hospedados. Pode ser um dispositivo FFD ou RFD (CARNEVAL et al., 2010).

c. Modos de operação

Para a transmissão de dados pela interface serial, os módulos Xbee podem utilizar dois modos de operação, transparente ou API (*Application Programming Interface*, ou interface de programação de aplicação). Em sistemas maiores também é possível utilizar os dois modos de operação na rede, de acordo com a necessidade. A seguir o detalhamento de cada modo (DIGI, 2019).

1) Modo transparente

Neste modo a informação é transmitida pelos módulos exatamente como é recebida. Todos os dados recebidos pela serial são transmitidos pelo ar, quando o outro módulo Xbee recebe os dados, os envia pela interface serial da mesma forma que foram recebidos. Este é um modo de utilização mais simples principalmente onde há somente dois módulos se comunicando.

Algumas das vantagens desse modo de operação é que por ter uma configuração fácil, é uma maneira simples de utilizar o Xbee, é fácil para o suporte, já que o que é enviado é exatamente o que é recebido, e também funciona bem quando existe uma comunicação de duas vias entre módulos Xbee.

Algumas das desvantagens são que não é possível ler a configuração de dispositivos na rede remotamente, é necessário atualizar a configuração antes de definir um novo destino e transmitir dados, não é possível saber de qual módulo na rede o dado foi recebido, já que o endereço do transmissor não está incluso, os dados recebidos não incluem os detalhes da transmissão ou as razões para o sucesso ou falha no envio e não oferece os recursos avançados do modo API, como atualizações de firmware e diagnósticos avançados de rede (DIGI, 2019).

A figura 5 demonstra o funcionamento do modo transparente durante o envio de dados:

Figura 5 – Funcionamento do modo transparente



Fonte: DIGI, 2019

Para este trabalho, o modo de operação escolhido foi o modo transparente, pela sua facilidade e simplicidade, e devido a comunicação ser feita somente entre dois módulos Xbee.

Para que os módulos possam se comunicar pelo modo transparente, é necessário definir o endereço do receptor na configuração do coordenador e vice-versa.

Para isso há dois parâmetros a serem definidos: DH (*Destination Address High*) e DL (*Destination Address Low*). Inserimos então o valor SH (*Serial Number High*) do coordenador no campo DH do roteador, e o valor SL (*Serial Number Low*) do coordenador no campo DL do roteador, essa configuração deve ser feita nos dois módulos inserindo os dados do módulo ao qual ele enviará ou receberá dados. Isso fará com que os módulos se comuniquem de forma transparente (DIGI, 2019).

A figura 6 demonstra a configuração efetuada em ambos os módulos por meio do software de configuração XCTU.

Figura 6 – Configuração modo transparente

XBEE TRANSMISSOR - ROTEADOR		XBEE RECEPTOR - COORDENADOR	
SH Serial Number High	13A200	SH Serial Number High	13A200
SL Serial Number Low	418D81C4	SL Serial Number Low	418D8A92
DH Destination Address High	13A200	DH Destination Address High	13A200
DL Destination Address Low	418D8A92	DL Destination Address Low	418D81C4

Fonte: Acervo do autor

2) Modo API

No modo API os dados são transmitidos pela interface serial em pacotes organizados e em determinada ordem, por meio de uma interface estruturada. Desta forma é possível estabelecer comunicações complexas entre os dispositivos sem a necessidade de definir um protocolo próprio.

É possível configurar o módulo Xbee sem a necessidade de entrar no modo de comando, como é o caso do modo de operação, já que no modo API há quadros diferentes para cada objetivo, como comunicação e configuração. Como os dados de destino são inclusos na estrutura dos quadros, é possível utilizar o modo API para transmitir para vários

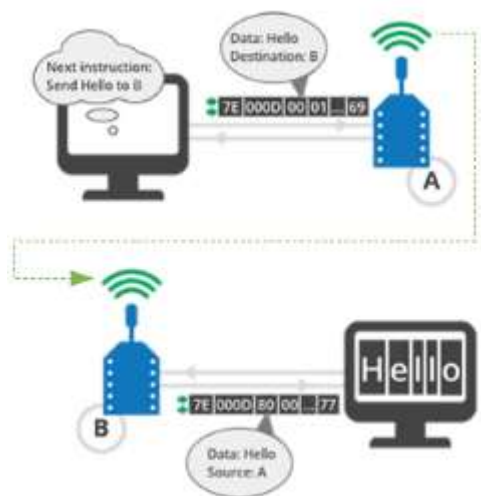
dispositivos, além disso o quadro API inclui a origem da mensagem, facilitando a identificação de onde os dados estão chegando.

Algumas das vantagens do modo API são a possibilidade de configuração local e remota de dispositivos na rede, envio de dados para um ou vários outros dispositivos, identificação da origem de cada pacote recebido assim como status de falha ou sucesso no envio de cada pacote, possibilidade de obter a força do sinal de cada pacote recebido, diagnóstico e gerenciamento avançado na rede, além da possibilidade de executar funções avançadas como atualização remota do firmware e etc.

Contudo, algumas das desvantagens do modo API são que sua interface é mais complexa e os dados são estruturados em pacotes em um formato específico, é mais difícil para suportar porque a transmissão é estruturada em pacotes que precisam ser analisados no recebimento e criados na transmissão, e também porque os dados enviados e recebidos não são iguais já que os pacotes recebidos contêm informações adicionais e dados de controle (DIGI, 2019).

A figura 7 demonstra o funcionamento do modo API durante o envio de dados:

Figura 7 – Funcionamento do modo API



Fonte: DIGI, 2019

d. Topologias

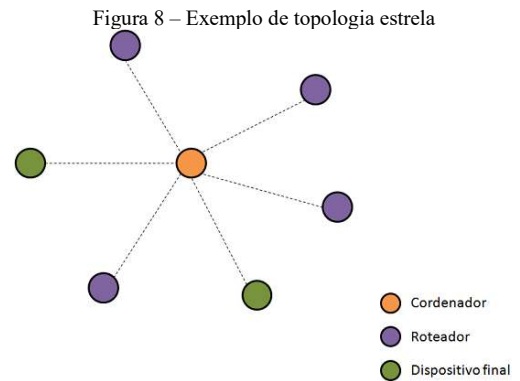
Há algumas maneiras de configurar a rede, e conforme a necessidade específica de cada aplicação é possível escolher a melhor topologia a ser empregada.

1) Estrela

Na topologia em estrela, há um único coordenador, chamado de coordenador PAN (*Personal Area Network*). Quando um FFD é ativado, ele estabelece sua rede e se torna o

coordenador PAN. Cada rede funciona com um identificador PAN, que deve ser diferente dos identificadores utilizados por outras redes que possam estar dentro da região de influência das ondas de rádio, desta forma cada uma das redes opera individualmente (CARNEVAL et al., 2010).

A figura 8 demonstra um exemplo de topologia estrela.



Fonte: CARNEVAL et al., 2010

2) Malha

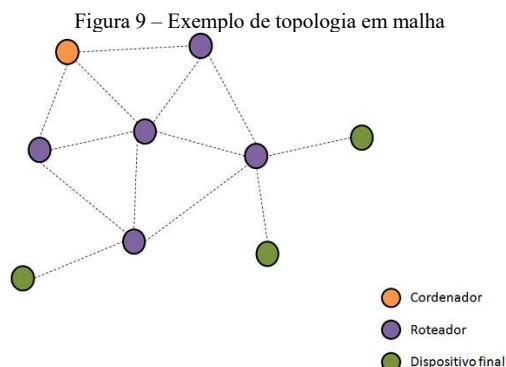
A topologia em malha (conhecida também como *mesh* ou *peer-to-peer*) também possui apenas um coordenador PAN. Todos os dispositivos podem se comunicar desde que estejam dentro da área de alcance. Pode ser considerada uma rede *ad hoc* e tem capacidade de se auto organizar e auto estruturar (*self-organizing e self-healing*).

Cada nó funciona como um repetidor, fazendo com que a rede cresça conforme novos nós forem adicionados a rede, e a comunicação seja efetuada utilizando vários caminhos diferentes.

Uma grande vantagem desta topologia é que caso algum dos nós se desconecte por algum motivo, como por exemplo um defeito, a comunicação chegará até o destino normalmente utilizando um outro caminho dentro da rede, além disso, ela possui uma fácil implantação.

Uma desvantagem é que ela possui um alto custo, porém os valores tendem a diminuir conforme a utilização desta tecnologia for aumentando (CARNEVAL et al., 2010).

A figura 9 demonstra um exemplo de topologia em malha.



Fonte: CARNEVAL et al., 2010

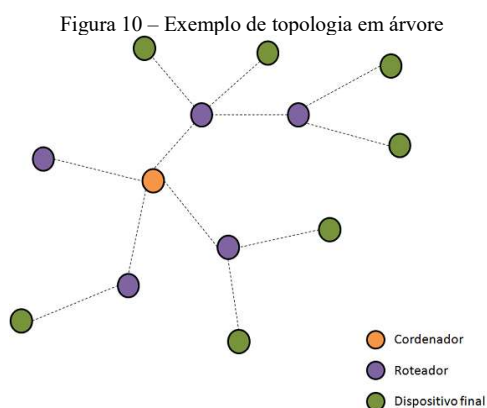
3) Árvore

A topologia em árvore pode ser considerada como um aglomerado de redes com topologia malha, ligados por um coordenador PAN que é responsável pela rede.

O coordenador formará o primeiro cluster e será o coordenador do aglomerado (CLH – *cluster Head*), e definirá um identificador para este cluster (CID – *Cluster Identifier*) através de um identificador PAN ocioso.

O dispositivo faz um *broadcast* do *beacon frame* e anuncia a existência da rede. Após receber o *beacon frame* algum dispositivo solicita a entrada na rede, no CLH. Caso o coordenador PAN permita a entrada do novo dispositivo, ele é considerado um dispositivo filho em sua listagem. Os dispositivos que forem adicionados irão se estabelecer e da mesma forma enviarão *beacon frames* para que novos dispositivos possam se juntar a rede. O coordenador PAN pode instruir um dispositivo a se tornar o CLH de um novo aglomerado. Sua principal vantagem é aumentar a área de cobertura, apesar de que os atrasos na entrega de pacotes também ocorrerão (CARNEVAL et al., 2010).

A figura 10 demonstra um exemplo de topologia em árvore.



Fonte: CARNEVAL et al., 2010

III. PROJETO DESENVOLVIDO

A. Arduino

Segundo a definição do site oficial do Arduino:

“Arduino é uma plataforma *open-source* de prototipagem eletrônica com hardware e software flexíveis e fáceis de usar, destinado a artistas, designers, *hobbistas* e qualquer pessoa interessada em criar objetos ou ambientes interativos” (ARDUINO, 2019)

O Arduino é uma plataforma formada por Hardware (placa Arduino), que podemos conectar a diversos componentes de acordo com o objetivo de determinado projeto, e Software (IDE Arduino), que é a interface de programação do Arduino.

É uma plataforma que tem muita facilidade de utilização, podendo ser utilizada até mesmo por pessoas com poucos conhecimentos da área e aprendida rapidamente (MOTA, 2017).

A figura 11 mostra os detalhes da placa Arduino Uno.



Fonte: MOTA, 2017

O microcontrolador é o dispositivo programável que executa o código gravado na placa. O modelo de microcontrolador utilizado pelo Arduino Uno é o ATmega328 (MOTA, 2017).

Os sensores de luminosidade, umidade, temperatura e raios UV estão conectados ao módulo Xbee transmissor por meio de uma placa Arduino. Esta faz a leitura, tratamento e envio dos dados lidos por meio dos sensores e os imprime na serial da seguinte forma:

LUMINOSIDADE UMIDADE TEMPERATURA
LEITURA-UV INDICE-UV

A figura 12 mostra parte da programação e a leitura em tempo real dos sensores.

Figura 12 – Programação Arduino – Xbee transmissor

```
// PROGRAMAÇÃO DO ARDUINO - XBEE TRANSMISSOR

char buffer[20];
int luminosidade = 0;
int umidade = 0;
const int LM35 = A2;
int temp = 0;
int uv = 0;
int index_uv = 0;

void setup() {
  Serial.begin(9600);
  pinMode(A0, INPUT);
  pinMode(A1, INPUT);
  pinMode(A2, INPUT);
  pinMode(A3, INPUT);
}

void loop() {
  //LEITURA DOS DADOS
  delay(1000);
  luminosidade = analogRead(A0) / 10.24; //LEITURA
  luminosidade = 100 - luminosidade;

  umidade = analogRead(A1) / 10.24; //LEITURA
  umidade = 100 - umidade;

  temp = analogRead(LM35); //LEITURA TEMPERATURA
  temp = int(float(((temp*5)/1023.0/0.010)));
}
```

Fonte: Acervo do autor

B. XCTU

O XCTU (*Configuration & Test Utility Software*) é uma aplicação gratuita criada pela DIGI para gerenciamento de diferentes módulos, incluindo os Xbee por meio de uma interface gráfica.

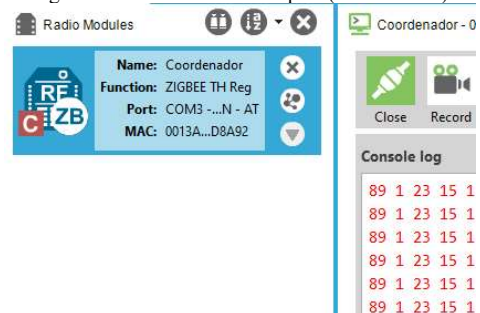
Possui ferramentas inclusas que facilitam a configuração e o teste dos módulos e é multiplataforma, ou seja, compatível com os sistemas operacionais mais utilizados, incluindo Windows, Linux e Mac OS.

Localiza módulos locais e remotos conectados ao PC, independentemente das configurações ou conexões, gerencia e configura vários módulos, incluindo módulos remotos que estejam se comunicando com o XCTU pelo ar.

Permite o uso de console de comando API e AT para comunicação com os módulos, além de possibilitar a gravação das sessões da console e carregá-las a qualquer momento. Permite visualizar a topologia das redes mostrando todos os nós e conexões graficamente ou em uma tabela. Utiliza ferramentas embutidas para executar operações desde recuperação de módulos a efetuar testes de alcance e atualiza a aplicação automaticamente assim como a biblioteca do firmware sem efetuar download de arquivos extras (DIGI, 2020).

A figura 13 mostra a leitura no Xbee receptor (coordenador) dos dados enviados pelo Xbee transmissor (roteador).

Figura 13 – Leitura Xbee receptor (coordenador)



Fonte: Acervo do autor

C. Labview

Labview (*Laboratory Virtual Instrument Engineering Workbench*) é uma aplicação de programação gráfica, desenvolvida pela *National Instruments* em 1986, utiliza a linguagem de programação gráfica “G”, que diferentemente das linguagens de programação tradicionais como C, C++ ou Java, onde a programação é via texto, utiliza a interface gráfica, o que faz com que a programação seja efetuada de forma mais simples e rápida.

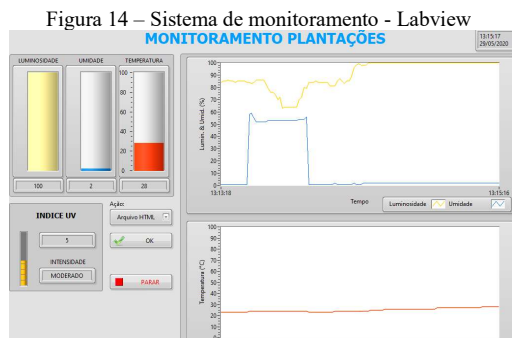
Sua aplicação principal é na realização de medições e automação e os programas desenvolvidos nele são chamados de instrumentos virtuais (IVs) e são compostos pelo painel frontal, com a interface, e pelo diagrama de blocos, onde fica o código gráfico do programa (KRING et al., 2006).

O Labview possui fácil integração com outros dispositivos e aplicações pois tem a possibilidade de utilização de *Add-ons*, que são módulos, *toolkits*, bibliotecas, *frameworks* de aplicação e outros produtos, que possibilitam a economia de tempo e esforço no desenvolvimento de aplicações (NATIONAL INSTRUMENTS, 2020).

O Labview foi a aplicação utilizada para obtenção e apresentação de informações neste projeto.

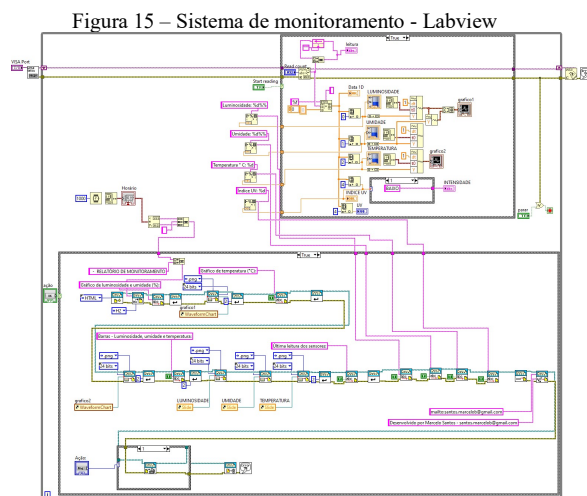
Foi efetuada a comunicação entre o Labview e o módulo Xbee receptor, onde, ao receber informações do módulo Xbee transmissor, irá compartilhá-las com o Labview e este apresentará os dados graficamente para o utilizador, onde será possível fazer uma análise detalhada com relação aos níveis de luminosidade, umidade, temperatura, e índice UV, além da possibilidade de gerar e salvar um relatório das medições em HTML ou imprimi-lo.

A figura 14 mostra a tela inicial do aplicativo criado em Labview para gerenciamento das informações.



Fonte: Acervo do autor

A figura 15 mostra o diagrama de blocos do Labview com a programação para o tratamento dos dados coletados.



Fonte: Acervo do autor

D. Sensor de luminosidade

A luminosidade foi monitorada com o sensor LDR (*Light Dependent Resistor*), também conhecido como foto resistor, é um componente sensível a luz, cuja resistência varia conforme a incidência da luz.

É feito de qualquer material semiconductor com uma alta resistência, um dos mais populares é o sulfeto de cádmio (CdS). Conforme é exposto a luz, os fótons são absorvidos e a energia é transferida para os elétrons, o que faz com que ele possa conduzir eletricidade, diminuindo então a resistência do semiconductor. Este processo é progressivo, quanto mais luz no semiconductor, mais elétrons são liberados para conduzir eletricidade e a resistência cai ainda mais (ELETROINFORMATICA, 2020).

E. Sensor de umidade

Os sensores de umidade possuem diversas aplicações, principalmente na agricultura,

onde a irrigação é um fator extremamente importante. O monitoramento da quantidade de umidade no solo e sua utilização em sistemas automatizados de irrigação ajuda a economizar recursos como água e o trabalho humano, por isso os sensores de umidade fazem a automação mais fácil.

O sensor de umidade FC-28 faz a medição do conteúdo volumétrico de água no solo por meio de duas pontas de prova e dá o nível de umidade na saída, possui saída digital e analógica podendo ser utilizado em qualquer um destes dois modos.

As duas pontas de prova permitem que a corrente passe pelo solo e então obtém o valor da resistência para medir o valor da umidade.

Quanto mais água, maior eletricidade o solo irá conduzir e haverá menor resistência, assim o nível de umidade será maior. Por outro lado, o solo conduz menor eletricidade quando está seco e, portanto, haverá uma menor resistência, assim o nível de umidade será menor.

Quando o valor analógico é medido, o sensor retorna um valor de 0 a 1023, como demonstraremos estes valores em porcentagem, a programação foi feita no Arduino para que a leitura final seja apresentada de 0% a 100%, do nível de umidade menor até o nível maior, respectivamente (JOSON, 2020).

F. Sensor de temperatura

Para a medição da temperatura foi utilizado o sensor LM35.

De acordo com *Texas Instruments* (2017), o sensor LM35 é um sensor de precisão e apresenta uma saída de tensão linear relativa a temperatura em que ele se encontrar no momento em que for alimentado por uma tensão e em sua saída um sinal variável de 10mV para cada grau celsius de temperatura e tem um alcance de -55°C a 150°C.

Possui uma baixa impedância de saída, saída linear e calibração inerente e precisa, o que facilita a interface para leitura e controle do circuito (TEXAS INSTRUMENTS, 2017).

G. Sensor de raios UV

O sensor GUVA-S12SD é um sensor UV (Ultravioleta) que foi desenvolvido para ter ótimos resultados ao ser aplicado juntamente com sistemas microcontroladores, como o Arduino. É capaz de detectar o nível de radiação solar UV por meio do chip GUVA-S12SD, sendo muito funcional na utilização em projetos onde é necessário o

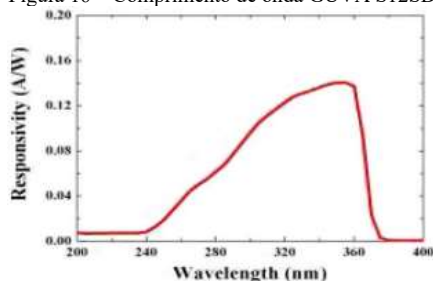
monitoramento de radiação ultravioleta liberada pela luz solar ou de lâmpadas UV.

Trabalha com tensões de 2,5V a 5V, oferecendo sinais analógicos ao microcontrolador e variando sua saída de acordo com o nível de raios UV detectados, sendo ideal para o monitoramento de radiação em praias, piscinas ou em estufas, onde a plantação é exposta aos raios UV controladamente.

Possui 3 pinos (VCC, SIG e GND), faz a leitura precisa do nível UV, possui uma resposta rápida e tem uma fácil conexão. Seu comprimento de onda UV é de 240 a 370 nm e a temperatura de trabalho é de -20 ~ 85°C (USINAINFO, 2020).

A figura 16 demonstra o gráfico da curva de resposta ao comprimento de onda.

Figura 16 – Comprimento de onda GUVA S12SD



Fonte: ROITHNER LASERTECHNIK, 2011.

A figura 17 demonstra a tabela de valores analógicos e seus respectivos índices ultravioleta.

Figura 17 – Leitura analógica e índice UV

UV Index	0	1	2	3	4	5
Vout(mV)	<50	227	318	408	503	606
Analog Value	<10	46	65	83	103	124
UV Index	6	7	8	9	10	11 ⁺
Vout(mV)	696	795	881	976	1079	1170+
Analog Value	142	162	180	200	221	240

Fonte: ARDUINO LEARNING, 2020

IV. ANÁLISE DOS RESULTADOS

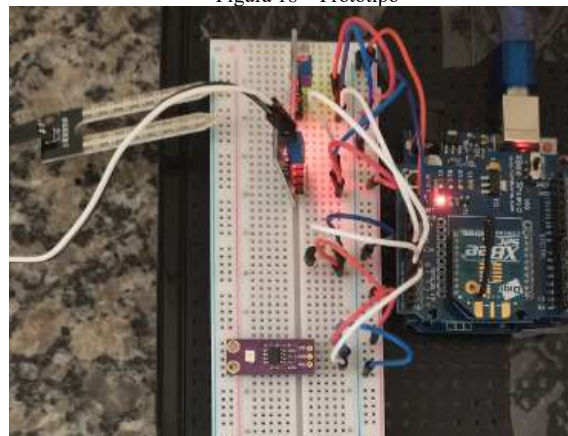
Como resultado deste projeto, esperava-se um bom funcionamento da comunicação via Zigbee em diferentes ambientes e situações, bem como a correta leitura, envio e tratamento dos dados dos sensores, e sua apresentação na aplicação Labview.

Foram efetuados testes na comunicação Zigbee em diferentes distâncias e ambientes, internos e externos, e

foi verificado um bom rendimento no envio e recepção dos dados por meio dos sensores.

Os sensores de luminosidade (LDR), umidade (FC-28), temperatura (LM35) e UV (GUVA-S12SD) foram montados no protoboard e suas conexões foram efetuadas conforme demonstrado na figura 18.

Figura 18 – Protótipo



Fonte: Acervo do autor

Para comprovação do funcionamento e eficácia do protótipo, foram efetuadas algumas análises, como a leitura de todos os sensores durante 3 dias no período das 6hrs às 22hrs ao ar livre, de modo a demonstrar seu funcionamento em um ambiente real. O resultado das leituras é apresentado na tabela 1.

Tabela 1 – Leitura dos sensores por 3 dias

DIA	HORÁRIO	LUMINOSIDADE (%)	UMIDADE (%)	TEMPERATURA (°C)	VALOR ANALÓGICO UV	ÍNDICE UV
25/ mai	06:00	41	2	17	16	1
	08:00	98	1	18	18	1
	10:00	100	2	20	68	3
	12:00	100	1	24	148	7
	14:00	100	1	23	82	3
	16:00	98	1	21	16	1
	18:00	45	1	20	15	1
	20:00	44	1	17	15	1
	22:00	42	1	17	15	1
26/ mai	06:00	43	3	21	14	1
	08:00	99	1	20	18	1
	10:00	100	1	22	95	4
	12:00	100	2	23	156	7
	14:00	100	6	18	118	5
	16:00	99	6	21	17	1
	18:00	45	1	13	15	1
	20:00	44	2	13	14	1
	22:00	37	1	17	15	1
27/ mai	06:00	36	1	14	22	1
	08:00	100	1	16	15	1
	10:00	100	3	29	100	4
	12:00	100	1	30	158	7
	14:00	99	1	34	89	4
	16:00	97	1	24	19	1
	18:00	40	1	21	16	1
	20:00	43	1	20	15	1
	22:00	38	1	20	15	1

Fonte: Acervo do autor

Após as leituras, foi feita uma média dos 3 dias, que é apresentada na tabela 2.

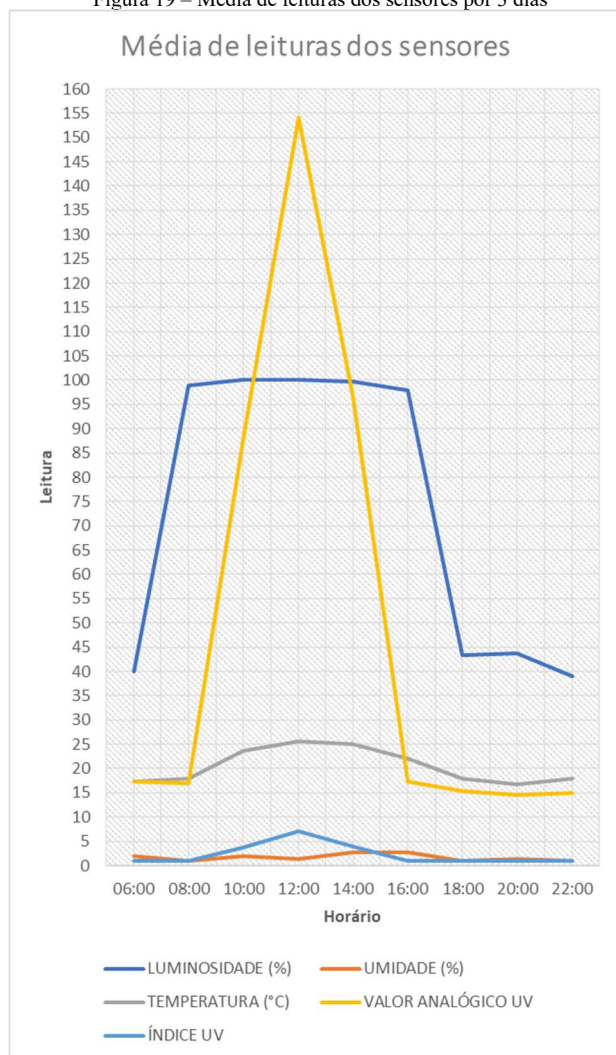
Tabela 2 – Média da leitura dos sensores por 3 dias

MÉDIA DE LEITURA DOS SENSORES - 3 DIAS					
HORÁRIO	LUMINOSIDADE (%)	UMIDADE (%)	TEMPERATURA (°C)	VALOR ANALÓGICO UV	ÍNDICE UV
06:00	40	2	17	17	1
08:00	99	1	18	17	1
10:00	100	2	24	88	4
12:00	100	1	26	154	7
14:00	100	3	25	96	4
16:00	98	3	22	17	1
18:00	43	1	18	15	1
20:00	44	2	15	15	1
22:00	40	1	17	15	1

Fonte: Acervo do autor

A figura 19 demonstra um gráfico das médias das leituras dos sensores na análise apresentada na tabela 2.

Figura 19 – Média de leituras dos sensores por 3 dias



Fonte: Acervo do autor

O sensor de umidade não tem mudanças significativas pois nesta análise ele foi utilizado ao ar livre, e sua função original é ser utilizado diretamente em contato com a superfície.

Para uma melhor demonstração do sensor de umidade foram efetuadas análises colocando-o em contato com pequenos algodões úmidos com diferenças de 1 em 1 ml de água, os resultados são apresentados na tabela 3.

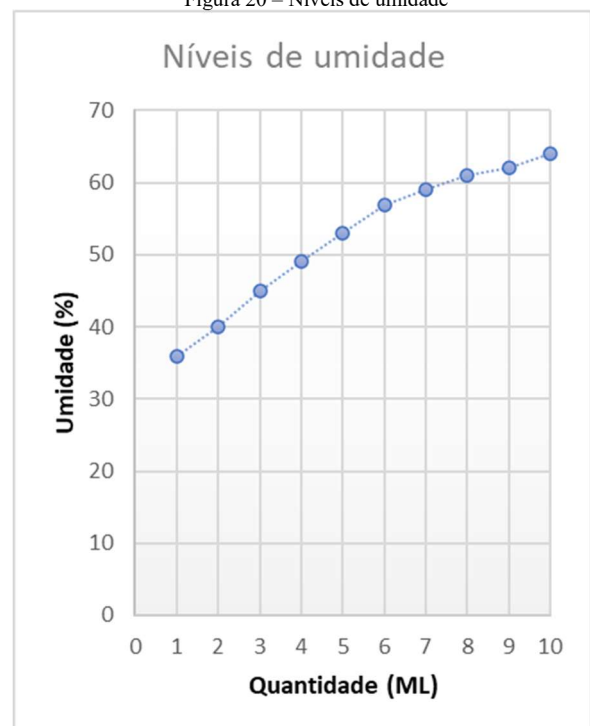
Tabela 3 – Análise de níveis de umidade

ANÁLISE DE NÍVEIS DE UMIDADE	
QUANTIDADE (ML)	UMIDADE (%)
1	36
2	40
3	45
4	49
5	53
6	57
7	59
8	61
9	62
10	64

Fonte: Acervo do autor

A figura 20 demonstra um gráfico com o resultado da análise dos níveis de umidade apresentados na tabela 3.

Figura 20 – Níveis de umidade



Fonte: Acervo do autor

V. CONCLUSÕES

Foi desenvolvido um protótipo com todas as ferramentas descritas e efetuados testes, garantindo a eficiência e segurança da utilização da tecnologia Zigbee no setor agrícola, demonstrando assim os inúmeros benefícios com relação a implantação de

novas tecnologias neste setor, que é extremamente importante para a economia.

A utilização do Arduino, assim como da aplicação XCTU para os módulos Xbee, garantiu uma configuração confiável, que pode ser utilizada na implantação do sistema em ambientes de variados tamanhos.

O Labview apresentou os dados coletados ao usuário final de maneira consistente e dinâmica, facilitando a análise dos mesmos e possibilitando melhorias futuras com base nas informações coletadas.

O sistema foi testado em diferentes situações com o intuito de simular um ambiente real de trabalho, provando assim ser um sistema eficiente, de qualidade e preciso na análise e demonstração dos dados.

REFERÊNCIAS

- ARDUINO LEARNING. **Arduino and GUVa-S12SD UV Sensor**. Arduino Learning, 2020. Disponível em: <<http://arduinolearning.com/code/arduino-guva-s12sd-uv-sensor.php>>. Acesso em 20 mai. de 2020.
- CARNEVAL, V.; COUTINHO, I.; LIMA, M.; VASQUES, B. **Zigbee**. Universidade Federal do Rio de Janeiro, Rio de Janeiro, 2010. Disponível em: https://www.gta.ufrj.br/grad/10_1/zigbee/. Acesso em: 22 set. de 2019.
- DIGI. **XBee®/XBee-PRO S2C Zigbee®**. Disponível em: <<https://www.digi.com/resources/documentation/digidocs/pdfs/90002002.pdf>>. DIGI, 2018 Acesso em: 11 mai. de 2019.
- DIGI. **XBee® Zigbee® Mesh Kit**. Disponível em: <<https://www.digi.com/resources/documentation/digidocs/pdfs/90001942-13.pdf>>. DIGI, 2019. Acesso em: 15 abr. de 2020.
- DIGI. **XCTU User Guide**. Disponível em: <https://www.digi.com/resources/documentation/digidocs/90001458-13/default.htm#concept/c_90001458-13_start.htm%3FTocPath%3D____1>. DIGI, 2020. Acesso em: 15 mai. de 2020.
- ELECTRONICS NOTES. **Light Dependent Resistor LDR: Photoresistor**. Disponível em: <https://www.electronics-notes.com/articles/electronic_components/resistors/light-dependent-resistor-ldr.php>. Acesso em 13 mai. de 2020.
- FRUTH, M. **Formal Methods for the Analysis of Wireless Network Protocols**, University of Oxford, Oxford, 2011. Disponível em: <<https://www.prismmodelchecker.org/papers/matthias-fruth-phd.pdf>>. Acesso em: 13 mar. de 2020.
- FRUTH, M. **IEEE 802.15.4 CSMA-CA Protocol (ZigBee)**, University of Oxford, Oxford, 2011. Disponível em: <<https://www.prismmodelchecker.org/casestudies/zigbee.php>>. Acesso em: 13 mar. de 2020.
- FRUTH, M. **Probabilistic Model Checking of Contention Resolution in the IEEE 802.15.4 Low-Rate Wireless Personal Area Network Protocol**, University of Birmingham, Edgbaston, 2011. Disponível em: <<https://www.prismmodelchecker.org/papers/isola06.pdf>>. Acesso em: 13 mar. de 2020.
- GISLASON, D. **Zigbee Wireless Networking**. 1. ed. Elsevier, 2008.
- HAREENDRAN, T. K. **GUVA-S12SD UV Sensor Module & Circuit**. **Electro Schematics**, 2020. Disponível em: <<https://www.electroschematics.com/guva-s12sd-uv-sensor-module-circuit/>>. Acesso em 15 mai. de 2020.
- JOSON, J. **Arduino and Soil Moisture Sensor - Interfacing Tutorial**. **Circuits Today**, 2020. Disponível em: <<http://www.circuitstoday.com/arduino-soil-moisture-sensor>>. Acesso em 16 mai. de 2020.
- KRING, J.; TRAVIS, J. **Introduction to Graphical Programming with LabVIEW**. Informit, out. 2006. Disponível em: <<https://www.informit.com/articles/article.aspx?p=662895&seqNum=3>>. Acesso em 12 mai. de 2020.
- MOTA, A. **O Que É Arduino e Como Funciona?** Vida de Silício, São Paulo, mai. 2017. Disponível em: <<https://portal.vidadesilicio.com.br/o-que-e-arduino-e-como-funciona/>>. Acesso em: 13 mar. de 2020.
- NATIONAL INSTRUMENTS. **Sistemas de teste e medição automatizados**. NI, 2020. Disponível em: <<https://www.ni.com/pt-br.html>>. Acesso em: 25 abr. de 2020.
- NEGREIRA. **Você sabe o que são as Estufas Agrícolas?** Negreira, São Paulo, fev. 2014. Disponível em: <<https://www.negreira.com.br/blog/voce-sabe-o-que-sao-estufas-agricolas/>>. Acesso em: 13 mai. de 2019.
- REVISTA AGROPECUÁRIA. **Estufas agrícolas: Quais as vantagens da sua utilização?** Disponível em: <<http://www.revistaagropecuaria.com.br/2019/10/18/estufas-agricolas-quais-as-vantagens-da-sua-utilizacao/>>. Revista Agropecuária, 2019. Acesso em: 13 mai. de 2020.
- ROITHNER LASERTECHNIK. **GUVA-S12SD Datasheet**. Vienna, fev. 2011. Disponível em <<http://www.roithner-laser.com/datasheets/pd/uv/guva-s12sd.pdf>>. Acesso em: 15 mai. de 2020.
- STENSMANN, B. **Propagação do Calor**. Instituto de Física, UFRGS, ago. 2002. Disponível em: <http://www.if.ufrgs.br/mpef/mef008/mef008_02/Berenice/aula3.html>. Acesso em 12 mai. de 2020.
- TELECO. **Zigbee**. Teleco. Disponível em: <<https://www.teleco.com.br/tutoriais/tutorialzigbee/default.asp>>. Acesso em: 14 mai. de 2019.
- TEXAS INSTRUMENTS, **LM35 Precision Centigrade Temperature Sensors**. Dallas, dez. 2017. Disponível em: <<http://www.ti.com/lit/ds/symlink/lm35.pdf>>. Acesso em 10 mai. 2020.
- TITUS, J. A. **The Hands-on XBEE Lab Manual**. 1. ed. Massachusetts: Elsevier, 2012.
- USINAINFO. **Sensor UV Ultravioleta para Arduino - Guva-S12SD**, Santo Ângelo, 2020. Disponível em: <

<https://www.usinainfo.com.br/sensor-uv-ultravioleta/sensor-uv-ultravioleta-para-arduino-guvas12sd-4614.html/>>. Acesso em 15 mai. de 2020.

ZIGBEE ALLIANCE. **ZigBee Specification**. Disponível em: <<https://zigbeealliance.org/wp-content/uploads/2019/11/docs-05-3474-21-0csg-zigbee-specification.pdf>>. Acesso em 29 abr. de 2020.