

SISTEMA DE AQUISIÇÃO DE DADOS E CONTROLE EM REDE DE ÁREA LOCAL (LAN) SEM FIO PARA SISTEMAS AUTOMATIZADOS.

Kaio Cesar Silva Santos 1, kaiocesar20092001@gmail.com, Universidade Federal do Norte do Tocantins¹, Nilo Mauricio Sotomayor Choque 2, nmsch@mail.uft.edu.br, Universidade Federal do Norte do Tocantins

CIÊNCIAS EXATAS E DA TERRA

RESUMO

Este projeto teve como objetivo criar um sistema de aquisição e controle de dados remoto, com foco na automação de sistemas de irrigação, com potencial para aplicações em diversas áreas. A metodologia incluiu a construção de um programa LabVIEW para receber, processar dados, além da criação de módulos Arduino com comunicação via XBee. O módulo envia dados físicos (como umidade e temperatura) para o programa LabVIEW, que permite o monitoramento em tempo real e o controle remoto. O programa foi projetado para ser facilmente configurável para diferentes aplicações. Os resultados demonstram que o sistema é eficaz na automação da irrigação, além de sua versatilidade o torna aplicável em diversos cenários de automação. Este sistema se destaca pela viabilidade de soluções de aquisição e controle de dados remotos para aumentar a eficiência em sistemas automatizados, promovendo a sustentabilidade e a economia de recursos.

Palavras-chave: Automação; irrigação, aquisição de dados remota.

INTRODUÇÃO

Uma parte importante para o desenvolvimento humano é o cultivo, é por tal método que conseguimos maior parte dos alimentos do nosso dia a dia, com o avanço da tecnologia foram criadas formas de otimizar a tarefa, como a irrigação, um método usado para entregar água e nutrientes para as plantas, sendo um sistema de extrema importância para o automatizar e melhorar a produção de alimentos (Ingale; Kasat. 2012).

Uma ferramenta bastante utilizada para esse e tantos outros processos é a plataforma Arduino, consiste em uma placa eletrônica que permite a construção de projetos, como acender leds, construir robôs, também, sistemas de irrigação (Admiral, 2022). Sendo uma ferramenta de hardware livre e de baixo custo, o que a torna extremamente versátil, podendo adicionar outras ferramentas, como sensores de umidade, temperatura, ou módulos de comunicação por rádio frequência.

Um desses meios de comunicação é o módulo Xbee, no qual se comunica de forma remota utilizando o protocolo zigbee, com radiofrequências entre 868MHz, 915MHz e 2,4GHz, com uma taxa máxima de 250kb/s. Uma parte vantajosa desse módulo é que ele não consome tanta energia, já que passa a maior parte de seu tempo em modo de economia de energia, podendo durar anos funcionando sem precisar trocar sua bateria.

Com base nesses conceitos, o objetivo deste trabalho foi criar um sistema de irrigação automatizado, no qual seja controlado remotamente por um programa criado em NI-Labview que nos mostra os dados de temperatura, umidade durante um certo período, e que consiga controlar o mesmo.

1. METODOLOGIA

O projeto foi desenvolvido em duas etapas, O conhecimento das plataformas e produção do projeto. A ideia inicialmente foi entender como usar os programas e equipamentos, tratando primeiramente projetos básicos com Arduino, dentro da própria IDE, ou avulsos na internet, com a intenção de ter afinidade com a plataforma. Em paralelo, foi trabalhado sobre a plataforma labview (um tipo de linguagem de programação baseado em diagrama de blocos) alguns conceitos básicos, como as estruturas de programação e suas ferramentas. Em seguida, com um conhecimento básico sobre labview e Arduino, foi desempenhado estudos sobre o protocolo zigbee e os módulos Xbees, entendendo como os dados são passados de um Router para o Coordinator.

Na segunda etapa ocorreu a prática, na qual corresponde a criar um programa em labview; programar as placas arduino; configurar os módulos Xbee e fazer a montagem. Para a montagem foi utilizado 2 placas arduino; 3 modulos Xbee; 2 shields; 2 sensores de umidade; cabos para conectar os sensores ao arduino; uma fonte de alimentação; e uma protoboard, um recipiente com uma bomba de água de baixa tensão.

2. RESULTADOS E DISCUSSÃO.

Durante a produção houve alguns problemas com os módulos Xbee, inicialmente foram usados 2 módulos para teste, porém um deles não estava funcionando e outro não era possível colocar em comunicação em API, foram disponibilizados mais dois módulos, que também não se comunicavam, recentemente, foi obtido mais 3, nos quais comunicavam entre si, porém o programa em Labview não conseguia identificar os dados, esse problema foi solucionado modificando o programa no arduino, pois o mesmo não estava mandando o frame em um posição que o Labview reconheça.

Figura 1: Programa em Arduino

```
/*CODE FOR SENDING API TRANSMIT REQUEST FROM A ROUTER XBEE TO A MESH COORDINATOR XBEE*/
/*THE FRAME SEND DIGITAL HUMIDITY SENSOR DATA A CADA 5 SEGUNDOS.*/
/*THE HUMIDITY SENSOR IS CONNECTED TO THE A0 ANALOG INPUT PORT OF ARDUINO UNO PCB*/
/*CODE TESTED AND FINALIZED IN 14TH AUGUST 2023. AUTHOR: N. M. SOTOMAYOR, RAIO CESAR SILVA SANTOS*/

#include <XBee.h>

const int sensorPin = A0; // Pino onde o sensor está conectado
int sensorValue; // Variável para armazenar os dados do sensor

// Create an XBee object
XBee xbee = XBee();

// Create a frame data buffer
uint8_t payload[6];

// Create an XBee address object for the coordinator
XBeeAddress64 coordinatorAddress = XBeeAddress64(0x0013A20041AED4E3);

// Create a TX request object
ZBTxRequest txRequest = ZBTxRequest(coordinatorAddress, payload, sizeof(payload));

void setup() {
  // start serial communication
  Serial.begin(9600);

  // start Xbee module
  xbee.begin(Serial);
}

void loop() {
  // read sensor value
  sensorValue = analogRead(sensorPin);

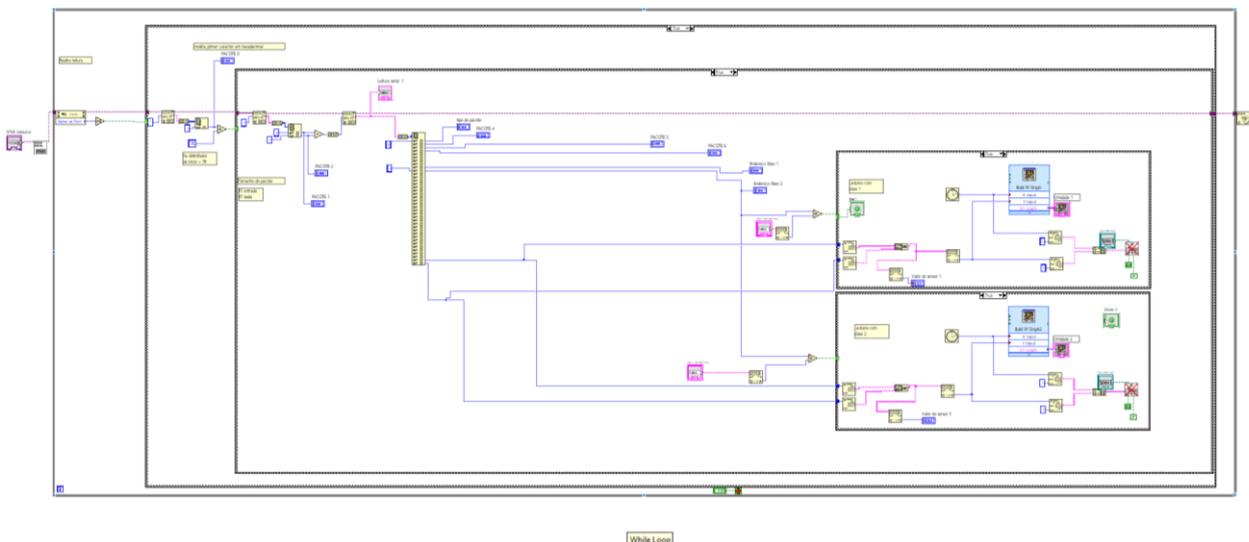
  // update payload with sensor value
  payload[0] = 0x7E;
  payload[1] = (uint8_t)(sensorValue >> 8);
  payload[2] = (uint8_t)(sensorValue & 0xFF);
  uint8_t checksum = 0xFF - (payload[1] + payload[2]);
  payload[3] = 0x00;
  payload[4] = 0x02;
  payload[5] = checksum;

  //send data for coordinator
  xbee.send(txRequest);
  // wait a 5 secs
  delay(5000);
}
```

A figura 1 mostra como ficou o programa para o arduino. O que esse programa faz basicamente e reconhecer o valor de um sensor na porta A0 e montar um frame em uma array e manda do router para o coordinator

No programa em Labview foram feitas algumas alterações, no qual foi adicionado um Case Structure entre os dados do sensor, tendo como indicador o endereço do Router que o usuário estiver usando, também foi reconhecido onde os dados ficam na array, para que eles sejam separados por endereço de cada Xbee, e assim plotando em um gráfico ou salvando em um arquivo, conforme mostra a figura 2. Foi colocado tratamento de dados para 2 Xbees pois foram usados apenas dois routers, pode-se modificar para mais módulos Xbee conforme a necessidade de sua aplicação, porém como os dados na qual o usuário queria pode ser escolhido, talvez não seja necessário aumentar a quantidade de ícones (os gráficos por exemplo).

Figura 2: Diagrama de blocos do programa em Labview.



Fonte: Autoria própria.

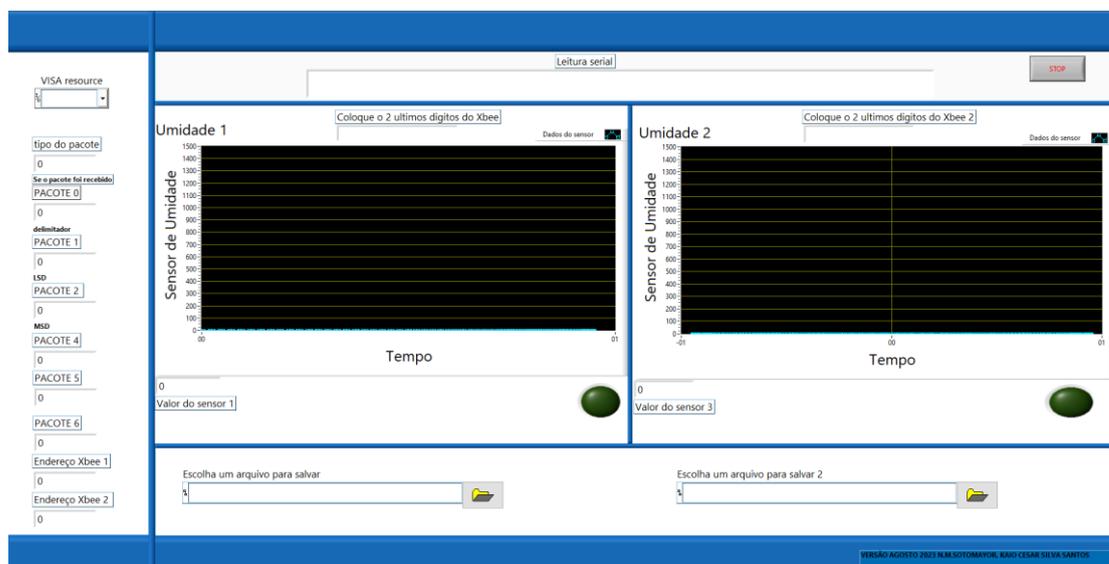
Também foi feita uma interface de usuário, conforme a figura 3. Ela mostra para o usuário qual porta escolher do coordinator, alguns pacotes em hexadecimal, e os últimos bytes do endereço hexadecimal do Xbee que está sendo lido. Na parte superior podemos ver a leitura serial que mostra todo o frame que o Labview está reconhecendo, e um botão para parar. Na área central do programa o usuário poderá ver os dados de alguns Xbee se colocar o ultimo byte do endereço, em seguida ele poderá ver como os dados vão se comportar ao longo do tempo, e ver o valor que o sensor abaixo do gráfico, foi colocado um booleano para indicar se o programa está lendo o Xbee. Por fim na parte inferior do programa o usuário pode selecionar onde salvar os dados que o recebe.

A figura 2 mostra um exemplo de montagem de um prototipo, basicamente e colocado uma fonte de alimentação no arduino, e conectados o sensor na porta A0, GND e 5V, e colocado a bomba. Em cima do arduino foi colocado um shield e o Xbee. Para receber os dados no computador foi utilizado outro Xbee com uma entrada USB.

Também foi feita uma interface de usuário, conforme a figura 3. Ela mostra para o usuário qual porta escolher do coordinator, alguns pacotes em hexadecimal, e os últimos bytes do endereço hexadecimal do Xbee que está sendo lido. Na parte superior podemos ver a leitura serial que mostra todo o frame que o Labview está reconhecendo, e um botão para parar. Na área central do programa o usuário poderá ver os dados de alguns Xbee se colocar o ultimo byte do endereço, em seguida ele poderá ver como os dados vão se comportar ao longo do tempo, e ver o valor que o sensor abaixo do gráfico, foi colocado um booleano para indicar se o programa está lendo o Xbee. Por fim na parte inferior do programa o usuário pode selecionar onde salvar os dados que o recebe.

A figura 2 mostra um exemplo de montagem de um prototipo, basicamente e colocado uma fonte de alimentação no arduino, e conectados o sensor na porta A0, GND e 5V, e colocado a bomba. Em cima do arduino foi colocado um shield e o Xbee. Para receber os dados no computador foi utilizado outro Xbee com uma entrada USB.

Figura 3: interface do usuário.



Fonte: Autoria propria.

3. CONCLUSÕES

O sistema de irrigação é fundamental para o cultivo e produção de alimento do ser humano, pois além de melhorar a produção, ainda reduz custos e desperdícios. Com o avanço da tecnologia tais sistemas ficaram cada vez mais automáticos, com várias formas de otimizar tal mecanismo, dentre elas está o uso da plataforma Arduino. O Arduino é um microcontrolador de hardware livre, tendo uma imensa aplicabilidade, dentre eles, o uso de sensores de umidade e temperatura, e portabilidade para transferência de dados por ondas eletromagnéticas. Diante disto, este trabalho relata a construção de um sistema de irrigação automático que recebe e controla dados físicos de forma remota, usando o protocolo Zigbee, em um software na linguagem de blocos (NI-Labview).

O projeto foi desenvolvido em duas partes, uma parte teórica e outra parte mais prática, em um primeiro momento foi feito estudos sobre a plataforma Arduino, Labview e Xbee. Em um segundo

momento foi feita parte prática, desenvolvendo o software de programação em Labview, e configurando os módulos Xbee, e por fim montado todo sistema de irrigação.

O sistema consiste em receber dados de um sensor de umidade e enviar para um modulo Xbee conectado a um computador com o programa criado em Labview, esse programa pegar o frame recebido e trata esses dados plotando ou enviando para um arquivo. Este arranjo funciona perfeitamente, vale destacar que pode ser colocado um sensor além do sensor de umidade, como um de temperatura ou de luz. Uma coisa que pode-se fazer e criar um executável para o programa em labview para que possa ser instalado em qualquer computador sem que precise do programa NI-Labview, também pode-se explorar tal projeto para automatizar outras áreas de serviço, como em incubadoras, ou até coisas mais simples, como na abertura de portas, poder ligar ou desligar outros dispositivos remotamente.

4. FINANCIAMENTOS

O trabalho contou com suporte do Instituto Nacional de Eletrônica Orgânica INEO, do Conselho Nacional de Desenvolvimento Científico e Tecnológico CNPq e da Pró-Reitoria de Pesquisa e Pós-Graduação.

5. REFERÊNCIAS

ADMIRAL, TIAGO DESTÉFFANI. **Aula experimental remota:** determinação do coeficiente de restituição utilizando Arduino, Revista de Enseñanza de la Física, vol 34. Pag 93-100. 2022.

INGALE, H. T.; KASAT, N. N. Automated Irrigation System. International Journal of Engineering Research and Development, v. 4, n. 11, p. 51-54, Nov. 2012. ISSN: 2278-067X.

SHAHIN FARAHANI. Zigbee Wireless Networks and Transceivers. NewnesEd. USA (2008)