
Autor do material: Cauan Teixeira Machado, Lucas Feitosa Lima Sátiro, Michael Alves dos Santos

Área: Sistemas Embarcados, Ciência de Dados

Palavras-chave: Embarcados, Arduino, Ciência de Dados

Sistema de Medição de Umidade e Temperatura: Integrando Arduino e Ciência de Dados

Sumário

1. Introdução	3
1.1. Contextualização	3
1.2. Conceitos Fundamentais	3
1.3 Visão Geral do Sistema	3
2. Componentes e Funcionamento	3
2.1. Hardware necessário	3
3. Implementação do Sistema	4
3.1. Código do Arduino	4
3.2. Código Python	4
3.3 Código Python para Análise	6
4. Análise de Dados	6
4.1. Processo de Coleta	6
4.2. Processamento	6
4.3. Resultados Esperados	6

1. Introdução

1.1. Contextualização

Este documento descreve a implementação de um sistema de medição de umidade e temperatura utilizando Arduino e Python. O projeto combina hardware e software para criar um sistema de monitoramento preciso e análise de dados em tempo real.

Aplicações: Monitoramento de reservatórios, Controle de processos industriais, Sistemas de irrigação automatizados.

1.2. Conceitos Fundamentais

a. Sensor Analógico de Temperatura e Umidade

- Princípio de funcionamento baseado na condutividade do vapor de água
- Variação da resistência elétrica conforme temperatura e umidade do ar
- Saída analógica proporcional à temperatura e umidade

b. Conversão Analógico-Digital

- Utilização do ADC do Arduino
- Resolução de 10 bits (valores de 0 a 1023)
- Conversão de tensão em valores digitais

1.3 Visão Geral do Sistema

O sistema opera em três etapas principais:

a. Captação

- Sensor realiza medições umidade e temperatura
- Conversão de nível em sinal elétrico

b. Digitalização

- Arduino processa o sinal analógico
- Conversão em dados digitais

c. Análise

- Python processa os dados recebidos
- Geração de insights e visualizações

2. Componentes e Funcionamento

2.1. Hardware necessário

Componentes Principais:

1. Arduino UNO
2. Sensor de umidade e temperatura

3. Cabos de conexão

4. Recipiente graduado

5. Computador para processamento

3. Implementação do Sistema

3.1. Código do Arduino

```
#include "dht.h" //INCLUSÃO DE BIBLIOTECA

#define pinoDHT11 A2//PINO ANALÓGICO UTILIZADO PELO DHT11

dht DHT; //VARIÁVEL DO TIPO DHT

void setup() {

    Serial.begin(9600); //INICIALIZA A SERIAL

    delay(2000); //INTERVALO DE 2 SEGUNDO ANTES DE INICIAR

}

void loop() {

    DHT.read11(pinoDHT11); //LÊ AS INFORMAÇÕES DO SENSOR

    Serial.print(DHT.humidity, 0); //IMPRIME NA SERIAL O VALOR DE
UMIDADE MEDIDO

    Serial.print(DHT.temperature, 0); //IMPRIME NA SERIAL O VALOR DE
UMIDADE MEDIDO E REMOVE A PARTE DECIMAL

    delay(2000); //INTERVALO DE 2 SEGUNDOS * NÃO DIMINUIR ESSE VALOR

}
```

3.2. Código Python

```
# Importação das bibliotecas utilizadas

import serial as ard

import pandas as pd

import time

# Função que lê o valor relativo ao sensor pelo Serial

def read_data(temperatura, umidade):
```

```
global arduino

dados = arduino.readline().decode('utf-8').strip().split(',')

dados_umidade = dados[0]

dados_temperatura = dados[1]

print(f"Dados recebidos: {dados_umidade},
{dados_temperatura}")

umidade.append(dados_umidade)

temperatura.append(dados_temperatura)


# Conecta ao serial pela porta utilizada pelo arduino

arduino = ard.Serial('COM9', 9600) # Alterar para o relativo ao
seu computador

contador = 1 # contador para controlar o número de leituras

MAX_VALUE = 150 # número máximo de leituras dos dados


# listas que armazenarão os valores lidos, relativos para cada
volume de água

lista_temperatura = []

lista_umidade = []

# Leitura de dados, com o volume de 50mL

while contador <= MAX_VALUE:

    read_data(lista_temperatura, lista_umidade)

    contador += 1


# Cria o dataframe utilizando as listas, para que possamos
analisar esses dados

df = pd.DataFrame({'Temperatura': lista_temperatura, 'Umidade':
lista_umidade})
```

```
# Exporta para csv o arquivo gerado com os dados

df.to_csv('dados_arduino.csv', index=False)

print(df)
```

3.3 Código Python para Análise

Disponível no repositório:

[GitHub | X SEMAC: Ciência de Dados para Sistemas Embarcados](#)

4. Análise de Dados

4.1. Processo de Coleta

Parâmetros de Amostragem:

- 150 amostras
- 2 medições - temperatura e umidade
- Intervalo de 2 segundos entre leituras

4.2. Processamento

Etapas de Processamento:

1. Coleta de dados brutos
2. Filtragem de ruídos
3. Cálculo de médias e desvios
4. Geração de gráficos
5. Análise estatística

4.3. Resultados Esperados

Métricas de Análise:

- Linearidade do sensor
- Precisão das medições
- Faixa de operação
- Resolução efetiva

5. Glossário

Termo	Definição
ADC	Conversor Analógico-Digital
Serial	Protocolo de comunicação Arduino-PC
DataFrame	Estrutura de dados do Pandas
Buffer	Memória temporária para dados
Calibração	Processo de ajuste sensor-medição