



MONITORAMENTO DE ESTUFA



IDEIA INICIAL



- Monitorar em **tempo real** as condições ambientais da estufa.
- Garantir **estabilidade térmica** para favorecer o crescimento das plantas.
- Registrar o estado da porta da estufa para **segurança e controle** operacional.
- Desenvolver o sistema utilizando o **STM32F103C8T6**.
- Criar uma **placa de circuito impresso** e um **modelo** para o produto real.

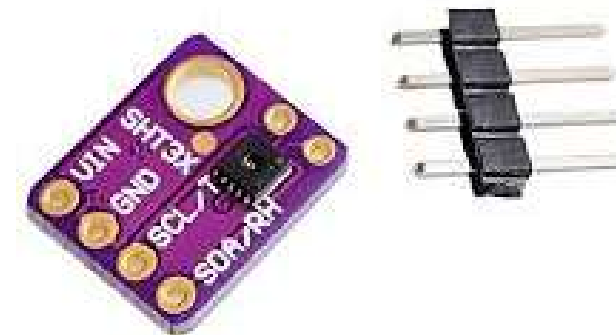
COMPONENTES PRINCIPAIS



Sensor Capacitivo
de Umidade do
Solo (CSMS)



Sensor de
Temperatura do Solo
(DS18B20)

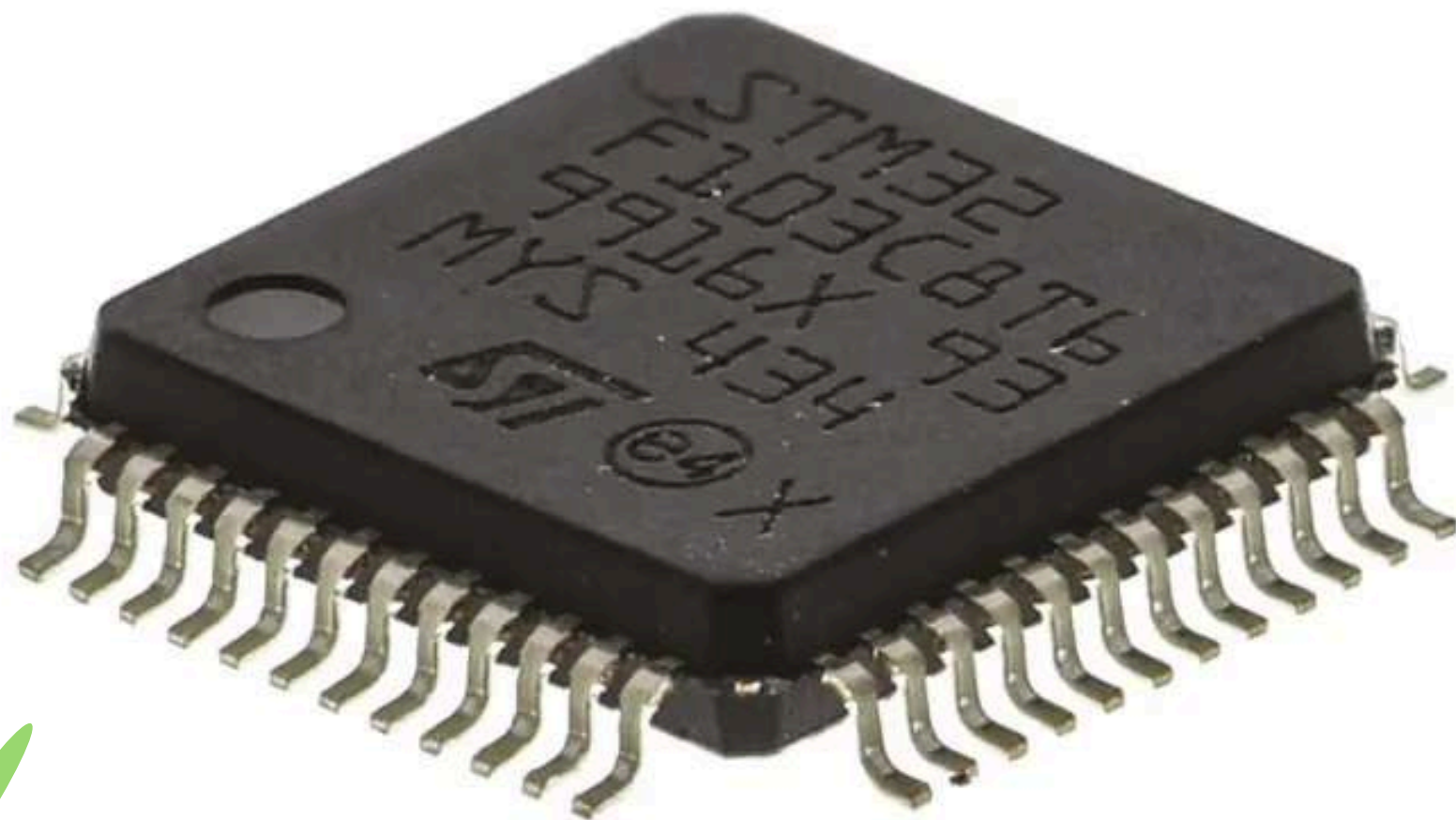


Sensor de
Temperatura e
Umidade do Ar
(SHT31)



Sensor Magnético de
Proximidade
(Reed Switch)

COMPONENTES PRINCIPAIS



Microcontrolador **STM32F103C8T6**

- Processamento **rápido**, ideal para leitura de sensores.
- Múltiplos periféricos para **comunicação com sensores** (I2C, GPIO, OneWire e USART).
- Compatibilidade com **STM32CubeIDE**.

CÓDIGO



```
float temperature = 0.0f;
SHT31_Data_t sht31_data;
float soil_humidity_percent = 0.0f;
uint8_t reed_switch_state = 0;

extern TIM_HandleTypeDef htim3;
extern UART_HandleTypeDef huart1;
extern I2C_HandleTypeDef hi2c1;
extern ADC_HandleTypeDef hadc1;

#define TEMPERATURA_LIMITE 28.0f
```

- Declaração das **variáveis**.
- Função extern: permite usar os **periféricos** configurados no **CubeMX**.
- Definição o **limiar de temperatura**.

Aqui, já dentro do **loop while**:

- **Leitura** dos dados dos **sensores**.
- **Ciclo contínuo** de monitoramento.

```
temperature = DS18B20_ReadTemp();

SHT31_Read_Data(&sht31_data);

soil_humidity_percent = CSMS_Read_Percent();
```

CÓDIGO



```
if (HAL_GPIO_ReadPin(GPIOB, GPIO_PIN_1) == GPIO_PIN_RESET)
{
    reed_switch_state = 1;
    HAL_GPIO_WritePin(GPIOB, GPIO_PIN_10, GPIO_PIN_SET);
}
else
{
    reed_switch_state = 0;
    HAL_GPIO_WritePin(GPIOB, GPIO_PIN_10, GPIO_PIN_RESET);
}
```

- Leitura do estado do **Reed Switch** no pino **PB1**.
- Controle do **LED** no pino **PB10**.



CÓDIGO



```
if (temperature > -127.0f) {  
    char tx_buffer[200];  
    int len = sprintf(tx_buffer,  
        "DS18B20 T(Solo): %.2f C\r\n"  
        "CSMS H(Solo): %.2f %%\r\n"  
        "SHT31 T(Ar): %.2f C\r\n"  
        "SHT31 H(Ar): %.2f %%\r\n"  
        "REED: %s\r\n\r\n",  
        temperature,  
        soil_humidity_percent,  
        sht31_data.temperature,  
        sht31_data.humidity,  
        (reed_switch_state == 1) ? "PORTA ABERTA" : "PORTA FECHADA"  
    );  
    HAL_UART_Transmit(&huart1, (uint8_t*)tx_buffer, len, 100);  
}
```

- Verificação de **leitura válida** do **DS18B20**.
- **String formatada** com todas as medições.
- String é enviada pela **UART1** para um computador em **tempo real**.

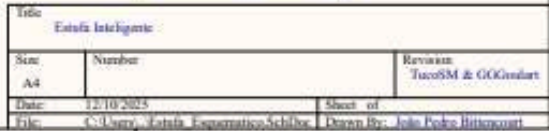
CÓDIGO



```
if (temperature > -127.0f)
{
    if (temperature >= TEMPERATURA_LIMITE)
    {
        HAL_GPIO_WritePin(GPIOB, GPIO_PIN_12, GPIO_PIN_RESET);
    } else {
        HAL_GPIO_WritePin(GPIOB, GPIO_PIN_12, GPIO_PIN_SET);
    }
} else {
    HAL_GPIO_WritePin(GPIOB, GPIO_PIN_12, GPIO_PIN_SET);
}

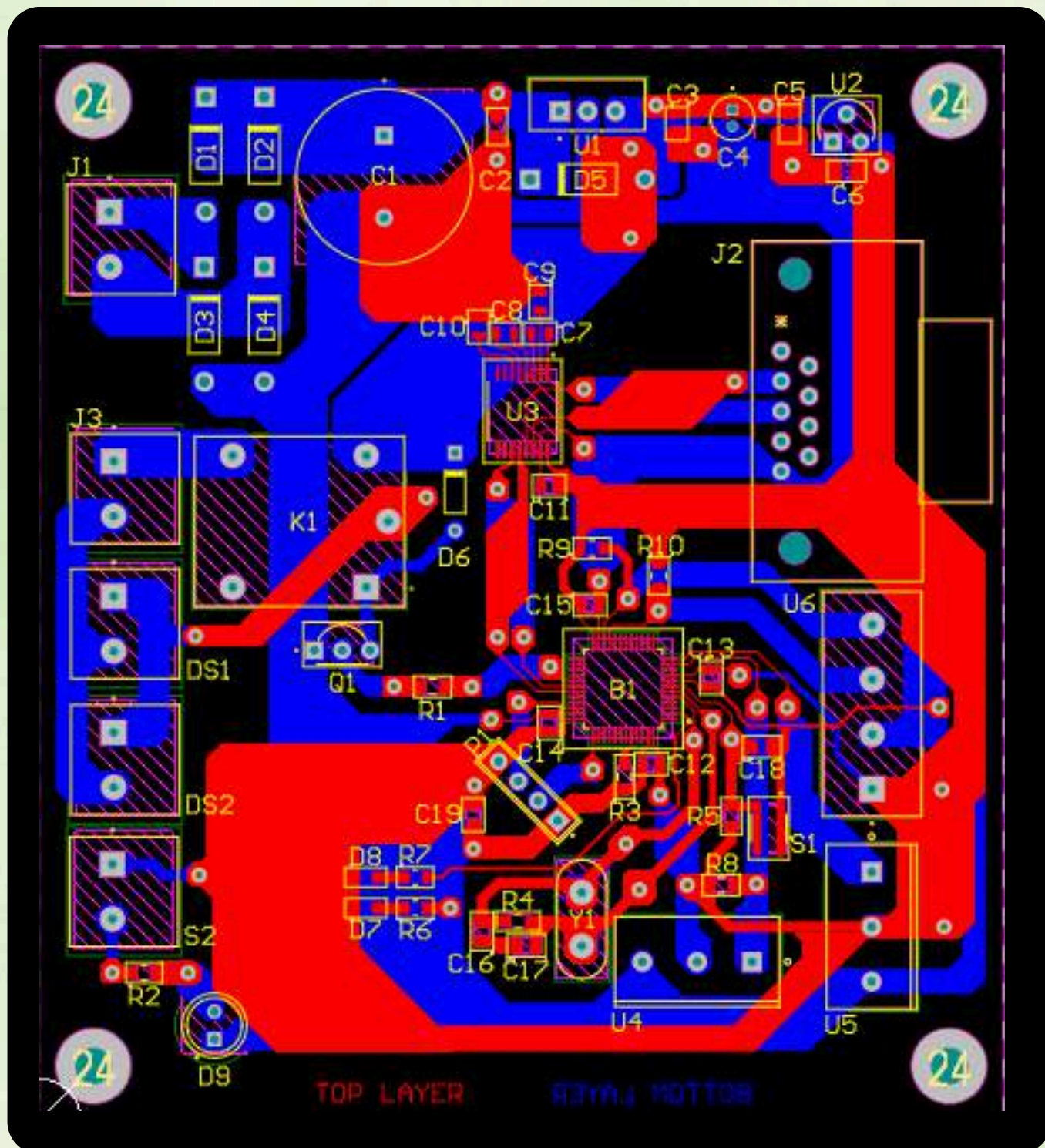
HAL_Delay(2000);
```

- O **controle da lâmpada** é feito com base na **temperatura**.
- Se está **abaixo do limite**, o sistema **liga a lâmpada** para aquecer a estufa.
- O **HAL_Delay(2000)** define o intervalo entre cada **ciclo de leitura** e controle.



- O esquemático **integra todos os blocos do sistema**: microcontrolador, sensores, fonte, comunicação e acionamento de carga.
- A **fonte AC/DC** utiliza ponte retificadora, capacitor de filtragem e reguladores de tensão para fornecer a **tensão e corrente** necessária para o sistema.

PCB




Roteamento da PCB:

- PCB em **duas camadas** e em setores separando sinais sensíveis, lógica e regiões de potência.
- **trilhas reforçadas** nas regiões de corrente mais alta (5V e 127V).
- Distâncias de segurança entre trilhas de AC e trilhas de lógica para **evitar interferência** e riscos.
- Conectores DIN organizam a ligação de sensores e atuadores de **forma modular**.



Visualização 3D:

- A visualização 3D permite verificar **posicionamento de conectores**, altura de componentes, colisões e **acessibilidade**.
 - Terminal Blocks organizam a **conexão dos sensores** (DS1, DS2, S1, etc.).
 - O microcontrolador, cristal, reguladores e componentes SMD ficam **agrupados de forma compacta**, reduzindo a interferência e caminho das trilhas
- 

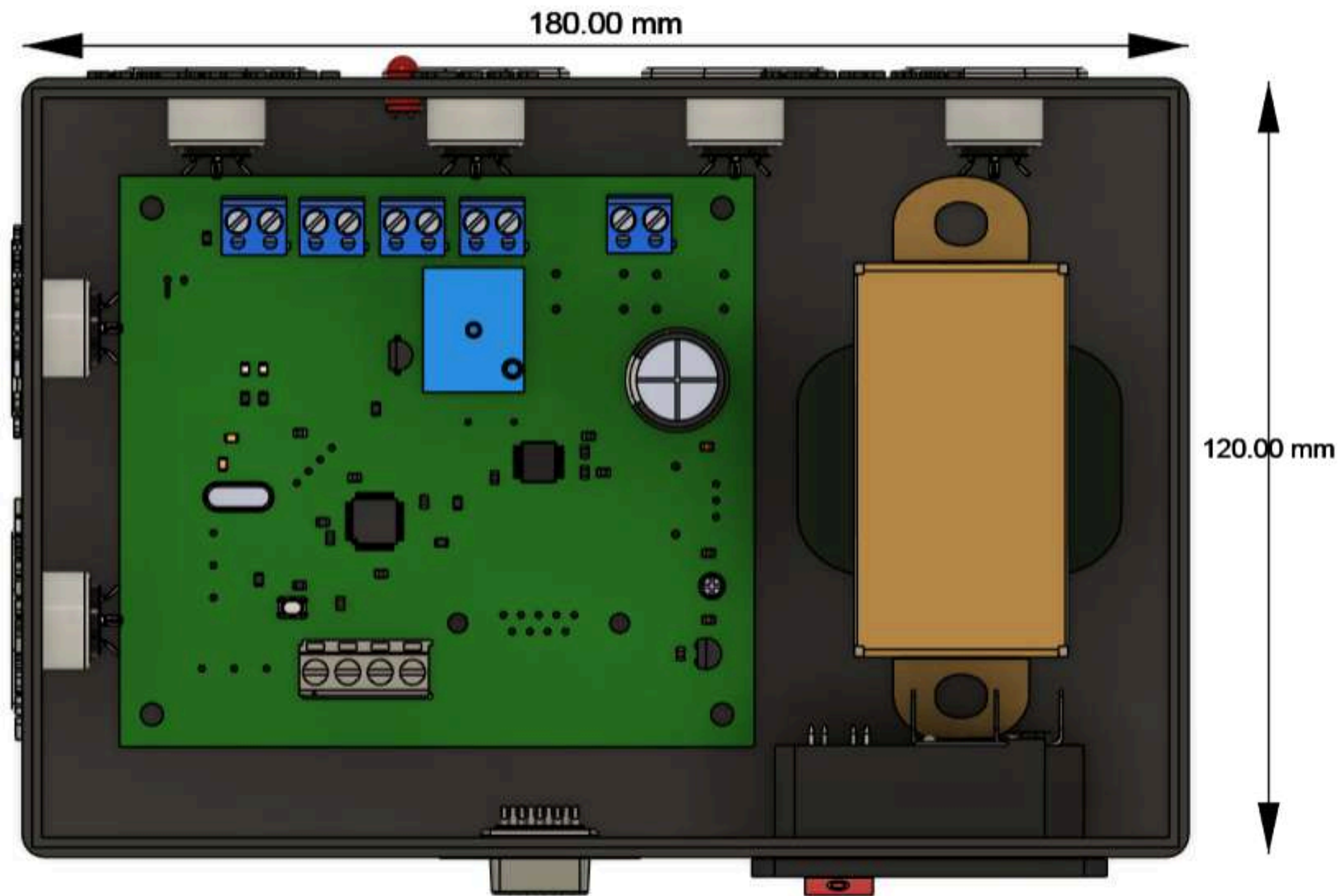
MODELO 3D



- Modelado no **Autodesk Fusion 360**, permitindo visualizar a disposição interna e externa da caixa.
- Utiliza **conectores DIN de 3 pinos** para ligar sensores e atuador.
- Inclui conector DB9 dedicado à **transmissão de dados** para um computador.
- Possui entrada para tomada com chave liga/desliga integrada, garantindo **segurança e facilidade** de alimentação.



MODELO 3D



- Design **compacto e organizado**, facilitando manutenção e identificação dos módulos.
- Módulos de terminais organizados na borda da placa **facilitam a conexão** de sensores.
- Transformador interno fornece **isolamento e alimentação** adequada para todo o sistema.

MODELO 3D



- Sensores dedicados de temperatura e umidade, garantindo **medições precisas e confiáveis**.
- Conectores DIN de 3 pinos para **fácil conexão** com a unidade principal.
- **Design compacto** projetado para ser enterrado no **solo**.



CONCLUSÃO



- Foi montado um **protótipo funcional** do kit para **monitoramento de estufas**.
- Na protoboard todos os **sensores** propostos foram **validados**.
- **PCB** final **projetada e fabricada**.
- Aprendizados:
 - **Roteamento** de PCB.
 - Firmware em **STM32**.
 - Integração e **controle de sensores**.
 - Comunicação serial.

OBRIGADO!



