

# Documentação Técnica - Controlador Python

## Protocolo de Comunicação - ESP32

[Protocolo de Comunicação - ESP32]

O ESP32 utiliza um protocolo de comunicação serial baseado em pacotes com delimitadores fixos para interagir com o software controlador em Python. As mensagens seguem a seguinte estrutura binária:

Formato da Mensagem:

[STX][ID\_CHIP][COMANDO\_ID][DADOS...][ETX]

- STX (Start of Text): 0x02
- ID\_CHIP:
  - MASTER\_DATA (0x08): Enviado pelo controlador Python
  - SLAVE\_DATA (0x07): Reservado para resposta do ESP32
- COMANDO\_ID: Byte que representa o tipo de comando ou requisição
- DADOS: Conteúdo do comando, variando entre 0, 1 ou múltiplos bytes
- ETX (End of Text): 0x03

Exemplo de mensagem:

[0x02, 0x08, 0x06, 0x01, 0xF4, 0x03] -> Define setpoint de temperatura para 50.0 grausC

Tamanhos de dados:

- 1 byte: comandos simples (ligar/desligar)
- 2 bytes: valores maiores, como setpoints
- 6 bytes: comandos RGB

Lista de COMANDO\_IDs:

- 0x01 - Ajustar Velocidade (1 byte)
- 0x05 - Ajustar Inclinação (1 byte)
- 0x04 - Setpoint de Pressão (2 bytes)
- 0x06 - Setpoint de Temperatura (2 bytes)

## Documentação Técnica - Controlador Python

- 0x09 - Requisição de Temperatura
- 0x0A - Requisição de Pressão
- 0x0B - Resposta de Temperatura Atual (2 bytes)
- 0x0C - Resposta de Pressão Atual (2 bytes)
- 0x0D - Resposta de Velocidade Atual (2 bytes)
- 0x0E - Resposta de Inclinação Atual (2 bytes)
- 0x0F - Alternância de Modo (1 byte, 0=Manual, 1=Automático)
- 0x10 - Controle do Aquecedor (1 byte)
- 0x11 - Controle da Bomba (1 byte)
- 0x12 - Controle da Lâmpada (1 byte)
- 0x13 - RGB LEDs (6 bytes: Rext, Gext, Bext, Rint, Gint, Bint)
- 0x15 - Liga/Desliga LEDs (1 byte)
- 0x16 - Controle Neon (1 byte)
- 0x17 - Ativar Aromatizador (1 byte: sempre 1)

### Tratamento:

- A função `process_PC_data()` interpreta os dados recebidos conforme o ID do comando.
- Verificação de integridade é feita por delimitadores STX/ETX e checagem do chip ID.

### Resposta:

- Para comandos como ID\_TEMP\_REQ (0x09), o ESP32 responde com a estrutura:

[0x02, 0x08, 0x0B, byte1, byte2, 0x03]

onde os bytes representam temperatura \* 10, convertida para int.

### Controle de LEDs:

- O ID 0x13 usa 6 bytes para definir RGB externo e interno.
- O ID 0x15 com valor 0 ou 1 ativa/desativa os LEDs com cor armazenada.

### Robustez:

- Buffer máximo: 128 bytes
- Proteção contra overflows e mensagens inválidas
- Erros geram envio de NAK (0x15)

## **Documentação Técnica - Controlador Python**

Todos os comandos são tratados em tempo real com resposta imediata, aproveitando o loop principal do ESP32 e interrupções de hardware.