

Documentação do Projeto de Telemetria CAN com ESP32

Eduardo Schvinn

April 22, 2025

1 Introdução

Este projeto tem como objetivo transmitir dados de sensores (aceleração e GPS) via protocolo CAN utilizando ESP32. A proposta visa facilitar a coleta de dados em tempo real para aplicações como monitoramento veicular em protótipos Baja SAE.

2 Componentes Utilizados

- 2 x ESP32 (ESP-IDF)
- 1 x Módulo MPU6050 (Acelerômetro e Giroscópio)
- 1 x Módulo GPS NEO-M8N
- 2 x Módulo CAN com CI SN65HVD230
- Cabos, protoboards e fonte de alimentação

3 Funcionamento Geral

- O ESP transmissor coleta dados do MPU6050 e do GPS.
- As informações são organizadas em pacotes CAN com identificadores `0x123` (aceleração) e `0x124` (GPS).
- O ESP receptor interpreta os pacotes e exibe os dados no terminal.

4 Estrutura dos Frames CAN Transmitidos

O sistema transmite dois frames distintos na rede CAN, cada um com identificador e estrutura de payload definidos:

Frame 1: Dados de Aceleração (ID 0x123)

- 6 bytes no total.
- Cada componente da aceleração (X, Y, Z) codificado em dois bytes (`int16_t`).
- Formato big-endian:

[ax_H, ax_L, ay_H, ay_L, az_H, az_L]

- Exemplo:

– ax = -512 0xFE00, ay = 256 0x0100, az = 16383 0x3FFF

– Payload: FE 00 01 00 3F FF

Frame 2: Coordenadas GPS (ID 0x124)

- 8 bytes no total.
- Latitude e longitude como float (IEEE 754), 4 bytes cada.
- Formato little-endian:

[latitude (4B), longitude (4B)]

- Exemplo:

– Latitude = -30.0457, Longitude = -51.2309

– Representação: 9A 99 F0 C1 0B 8D 4E C2

Resumo dos Frames

Frame	ID	Bytes (Payload)	Conteúdo
Frame 1	0x123	6	Aceleração (X, Y, Z)
Frame 2	0x124	8	Latitude (4B) + Longitude (4B)

5 Formato dos Dados Transmitidos

A separação em dois frames evita a perda de precisão das coordenadas GPS.

Frame 1 (ID 0x123):

- Bytes 0-1: aceleração em X (`int16`)
- Bytes 2-3: aceleração em Y (`int16`)
- Bytes 4-5: aceleração em Z (`int16`)

Frame 2 (ID 0x124):

- Bytes 0-3: latitude (`float`)
- Bytes 4-7: longitude (`float`)

6 Conversão de Dados GPS no Receptor

A transmissão das coordenadas GPS segue o padrão IEEE 754 para `float` (4 bytes), utilizando a ordem *little-endian*.

No Transmissor

Os valores são copiados diretamente para o buffer da mensagem:

```
memcpy(&msg2.data[0], &latitude, sizeof(float));  
memcpy(&msg2.data[4], &longitude, sizeof(float));
```

No Receptor

O processo reverso reconstitui os valores:

```
float latitude, longitude;  
memcpy(&latitude, &msg.data[0], sizeof(float));  
memcpy(&longitude, &msg.data[4], sizeof(float));
```

Considerações

- A ordem dos bytes deve ser idêntica em ambos os dispositivos.
- O uso de `memcpy` garante fidelidade dos dados sem conversões.
- O formato binário permite transmitir floats com eficiência.

7 Fluxo do Sistema

1. Inicialização de UART (GPS), I2C (MPU6050) e CAN.
2. Leitura dos sensores.
3. Transmissão a cada 500ms.
4. Receptor interpreta e exibe os dados no terminal.