

Relatório — Implementação da FFT na Raspberry Pi Pico 2

EQUIPE: André Alves, Arthur Roberto, Maryana Moraes e Pedro Nóbrega

1.1 Montagem do Hardware

O circuito de medição de corrente foi construído utilizando o sensor de corrente hw-670. O sinal de saída do sensor foi adaptado para leitura pelo ADC da Raspberry Pi Pico 2 através de um circuito de condicionamento composto por:

- Divisor de tensão resistivo com dois resistores de 100 kΩ ligados entre 3,3 V e GND, fornecendo uma referência DC de 1,65 V para centralizar o sinal AC no meio da faixa do ADC.
- Saída do sensor conectado ao nó de referência de 1,65 V.
- O ponto de medição (nó de referência de 1,65 V) foi conectado ao pino A0 / GP26 da Pico 2.

Esse circuito garante que o sinal senoidal oriundo do sensor fique dentro da faixa de 0–3,3 V, centrado em 1,65 V, e possa ser mostrado com segurança e fidelidade.

1.2 Implementação da FFT na Raspberry Pi Pico 2

A transformada rápida de Fourier foi implementada usando a biblioteca arduino FFT, que permite análise espectral em tempo real diretamente no microcontrolador.

Trechos-chave do código:

- **Configuração da FFT:**

```
constexpr uint16_t N_AMOSTRA = 1024;  
constexpr double FREQ_AMOS = 8000.0; // 8 kHz  
ArduinoFFT FFT(vReal, vImag, N_AMOSTRA, FREQ_AMOS);
```
- **Amostragem do sinal com temporização precisa:**

```
for (uint16_t i = 0; i < N_AMOSTRA; ++i) {  
    while ((micros() - t0) < i * samplePeriodUs) { }  
    double sample = adcRawToVoltage(analogRead(PIN_LOAD));  
    vReal[i] = sample;  
    vImag[i] = 0.0;  
    sum += sample;  
}
```
- **Pré-processamento (remoção da componente DC):**

```
double dc = sum / N_AMOSTRA;  
for (uint16_t i = 0; i < N_AMOSTRA; ++i) vReal[i] -= dc;
```
- **Execução da FFT e envio do espectro via UART:**

```
FFT.windowing(FFT_WIN_TYP_HANN, FFT_FORWARD);  
FFT.compute(FFT_FORWARD);  
FFT.complexToMagnitude();  
for (uint16_t i = 1; i < (N_AMOSTRA / 2); ++i) {  
    double freq = i * freqResolution;  
    double mag = vReal[i];  
    Serial.print(freq, 2);  
    Serial.print(',');  
    Serial.println(mag, 6); }
```

O código realiza a análise espectral a cada **500 ms**, enviando os resultados por UART delimitados por marcadores `---FFT_START---` e `---FFT_END---`.

1.3 Testes Realizados

Testes foram feitos com diferentes cargas simples e seguras como: lâmpada de LED, ventilador de mesa, carregador de celular, carregador de notebook, liquidificador, ferro de solda, etc.

Com o sensor posicionado em um dos fios da rede de cada carga, o sinal foi observado em um script em python e sua frequência fundamental (~60 Hz) foi corretamente detectada como o pico principal na FFT. As harmônicas também apareceram de forma coerente com o tipo de carga.

1.4 Resultados da Detecção

Embora a FFT não tenha sido usada diretamente para detecção ON/OFF, ela forneceu uma representação precisa da presença de corrente e de padrões de frequência. Como alternativa, um algoritmo simples de detecção por valor RMS médio (baseado na energia da FFT ou no valor de pico) poderia ser usado para identificar a presença de carga.

1.5 Desafios Enfrentados

Ruído na linha de medição: O sinal apresentou pequenas variações mesmo sem carga, devido à sensibilidade do ADC e à captação de ruído da rede elétrica.

Precisão do ADC: O ADC da Raspberry Pi Pico 2 possui resolução de 12 bits, o que é suficiente, mas exige atenção à referência (V_{REF}) e ruído na linha de alimentação.

Taxa de amostragem constante: A geração de amostras com `micros()` funcionou bem, mas requer cuidados para manter a precisão da temporização em loops longos.

Capacidade da FFT: FFT de 1024 pontos com 8 kHz ocupa memória RAM significativa, mas a Pico 2 conseguiu processar dentro do tempo real com margem confortável.