

# Relatório Técnico - Teste com ZephyrOS no QEMU

Gabriel Alves Massuda

Repositório GitHub: <https://github.com/seu-usuario/seu-repositorio>

24 de março de 2025

# Sumário

|          |  |          |
|----------|--|----------|
| <b>1</b> | <b>Introdução</b>                          | <b>2</b> |
| <b>2</b> | <b>Configuração do Ambiente</b>            | <b>2</b> |
| 2.1      | Pré-requisitos . . . . .                   | 2        |
| 2.2      | Passos para Configuração . . . . .         | 2        |
| <b>3</b> | <b>Implementação</b>                       | <b>2</b> |
| 3.1      | Estrutura do Projeto . . . . .             | 3        |
| 3.2      | Compilação e Execução . . . . .            | 3        |
| <b>4</b> | <b>Desafios e Soluções</b>                 | <b>3</b> |
| <b>5</b> | <b>Testes e Resultados</b>                 | <b>3</b> |
| <b>6</b> | <b>Melhorias e Funcionalidades Futuras</b> | <b>4</b> |
| <b>7</b> | <b>Conclusão</b>                           | <b>4</b> |

# 1 Introdução

Este relatório descreve a implementação de um sistema embarcado simulado utilizando o ZephyrOS® no ambiente de virtualização QEMU®, conforme solicitado pelo desafio proposto pela Equipe Raptor. O objetivo foi criar uma solução capaz de exibir a mensagem **"Hello World! This is Raptor!"** em um hardware simulado, utilizando as bibliotecas e APIs do Zephyr.

O projeto foi desenvolvido com foco em clareza, organização e facilidade de execução, garantindo que até mesmo pessoas leigas possam reproduzir os resultados. O código-fonte completo está disponível no repositório GitHub mencionado na capa deste relatório.

## 2 Configuração do Ambiente

### 2.1 Pré-requisitos

Para executar este projeto, é necessário ter instalado:

- Sistema Operacional Windows (versão 10 ou superior recomendada)
- Zephyr SDK (seguindo o Zephyr Getting Started Guide)
- QEMU (instalado automaticamente com o Zephyr SDK)
- Git (para clonar o repositório)

### 2.2 Passos para Configuração

#### 1. Instale o Zephyr SDK:

- Siga o guia oficial de instalação para Windows
- Certifique-se de que as variáveis de ambiente (ZEPHYR\_BASE, ZEPHYR\_TOOLCHAIN\_VARIANT, etc.) estejam configuradas corretamente

#### 2. Clone o repositório:

```
1 git clone https://github.com/seu-usuario/seu-repositorio.git
2 cd seu-repositorio
3
```

#### 3. Execute o projeto como administrador:

- Abra o Prompt de Comando (cmd) ou PowerShell (PS) como administrador

## 3 Implementação

O projeto foi baseado no exemplo `hello_world` do Zephyr, com as seguintes etapas:

### 3.1 Estrutura do Projeto

- Foi criado um diretório chamado `Desafio` para organizar os arquivos
- O código principal (`main.c`):

```
1 #include <zephyr/kernel.h>
2 #include <zephyr/sys/printf.h>
3
4 void main(void) {
5     printf("Hello World! This is Raptor!\n");
6 }
7
```

- Os arquivos `CMakeLists.txt` e `prj.conf` foram configurados conforme a estrutura de build do Zephyr

### 3.2 Compilação e Execução

Para compilar e executar o projeto no QEMU:

```
1 west build -b qemu_x86 ./Desafio
2 west build -t run
```

Após a execução, a mensagem **"Hello World! This is Raptor!"** será exibida no terminal.

## 4 Desafios e Soluções

Durante o desenvolvimento, os seguintes desafios foram encontrados e resolvidos:

- **Configuração das Variáveis de Ambiente:**
  - Problema: As variáveis de ambiente não foram configuradas automaticamente
  - Solução: Configuração manual no Painel de Controle do Windows
- **Execução no Windows:**
  - Problema: Erros de permissão ao executar comandos do Zephyr
  - Solução: Execução do Prompt de Comando ou PowerShell como administrador
- **Compilação e Execução:**
  - Problema: Dificuldade inicial com os comandos do Zephyr
  - Solução: Uso dos comandos `west build` conforme documentação oficial

## 5 Testes e Resultados

O projeto foi testado com sucesso no ambiente QEMU, exibindo a mensagem esperada sem erros. A saída do terminal foi:

```
1 Hello World! This is Raptor!
```

## 6 Melhorias e Funcionalidades Futuras

Caso haja tempo para evoluir o projeto, as seguintes melhorias podem ser implementadas:

- **Simulação de LED Virtual:** Adaptar o projeto para incluir um exemplo semelhante ao `blink`, simulando a interação com um LED virtual
- **Logs de Depuração:** Implementar logs detalhados para facilitar a depuração em cenários mais complexos
- **Exploração de Periféricos:** Utilizar periféricos simulados no QEMU, como UART e GPIO, para expandir as funcionalidades

## 7 Conclusão

O projeto demonstrou a viabilidade de utilizar o ZephyrOS em conjunto com o QEMU para simular sistemas embarcados, atendendo ao desafio proposto. A maior dificuldade foi a configuração inicial do ambiente, mas uma vez resolvida, o desenvolvimento foi eficiente. O código e este relatório estão disponíveis no repositório GitHub para consulta e reprodução por outros desenvolvedores, incluindo aqueles com menos experiência.

## Anexos

- Link para o repositório GitHub: <https://github.com/seu-usuario/seu-repositorio>
- Link para a documentação do Zephyr: <https://docs.zephyrproject.org/latest/>