

Arquitetura HIL para teste de sistemas embarcados como *vehicle interface* de veículos autônomos baseados no Autoware

Projeto – Etapa 2

Gabriel Toffanetto França da Rocha

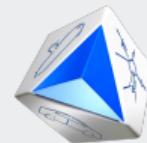
g289320@dac.unicamp.br

Professor Dr. Rodrigo Moreira Bacurau

IM420X – Projeto de Sistemas Embarcados de Tempo Real

Faculdade de Engenharia Mecânica
Universidade Estadual de Campinas

12 de novembro de 2024



Agenda

1 Introdução

2 Sistema embarcado

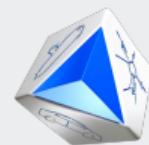
3 Testes dos módulos

4 Problemas encontrados

5 Cronograma



Introdução



Proposta

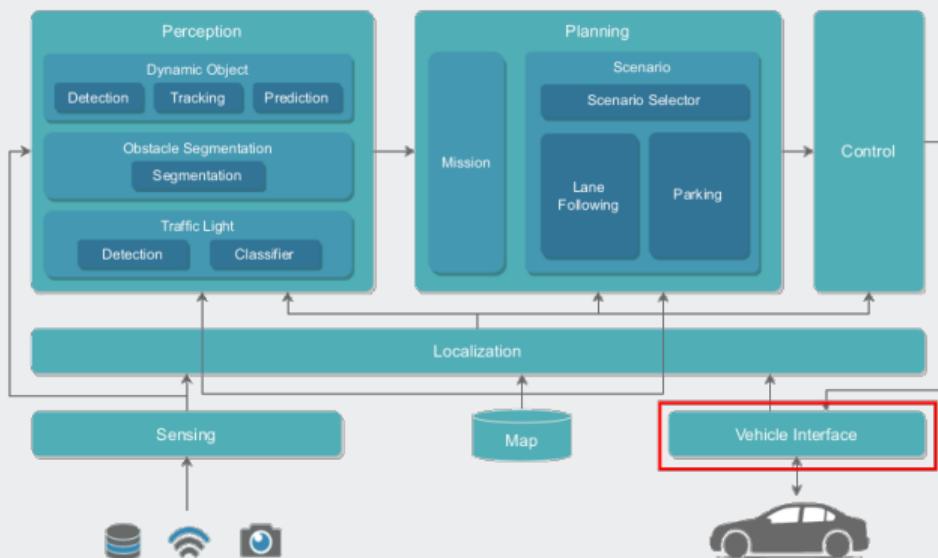


Figura 1: Escopo do projeto na arquitetura Autoware.

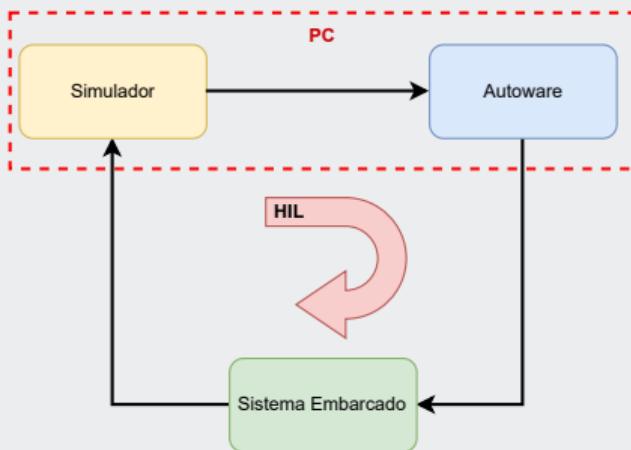


Figura 2: Arquitetura de teste do *hardware*.



Sistema embarcado

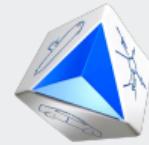


Diagrama de blocos

- Sobrecarga do micro-ros para controle e tráfego dos dados do simulador;
- Aproximação que leva à *overhead* comparado com a arquitetura real.

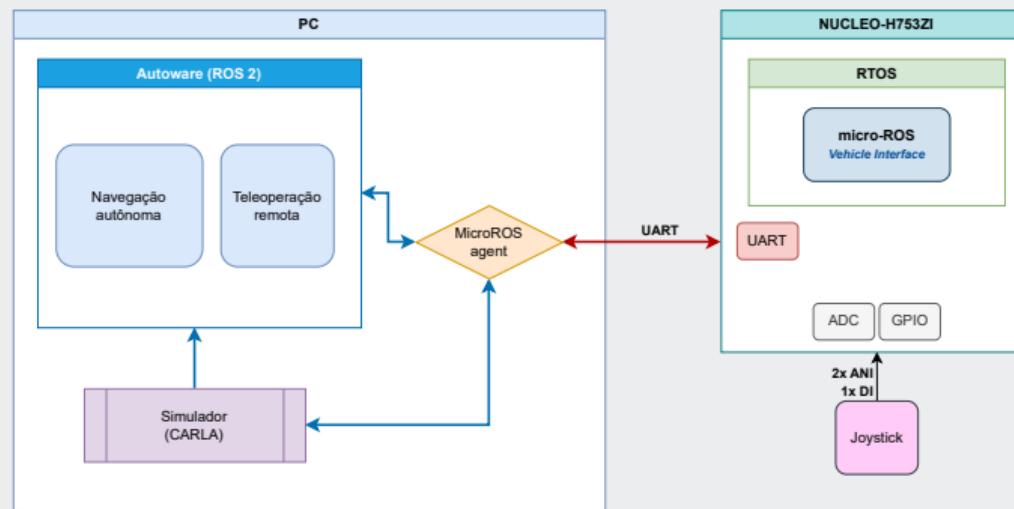


Figura 3: Diagrama de blocos atualizado.

Diagrama de blocos

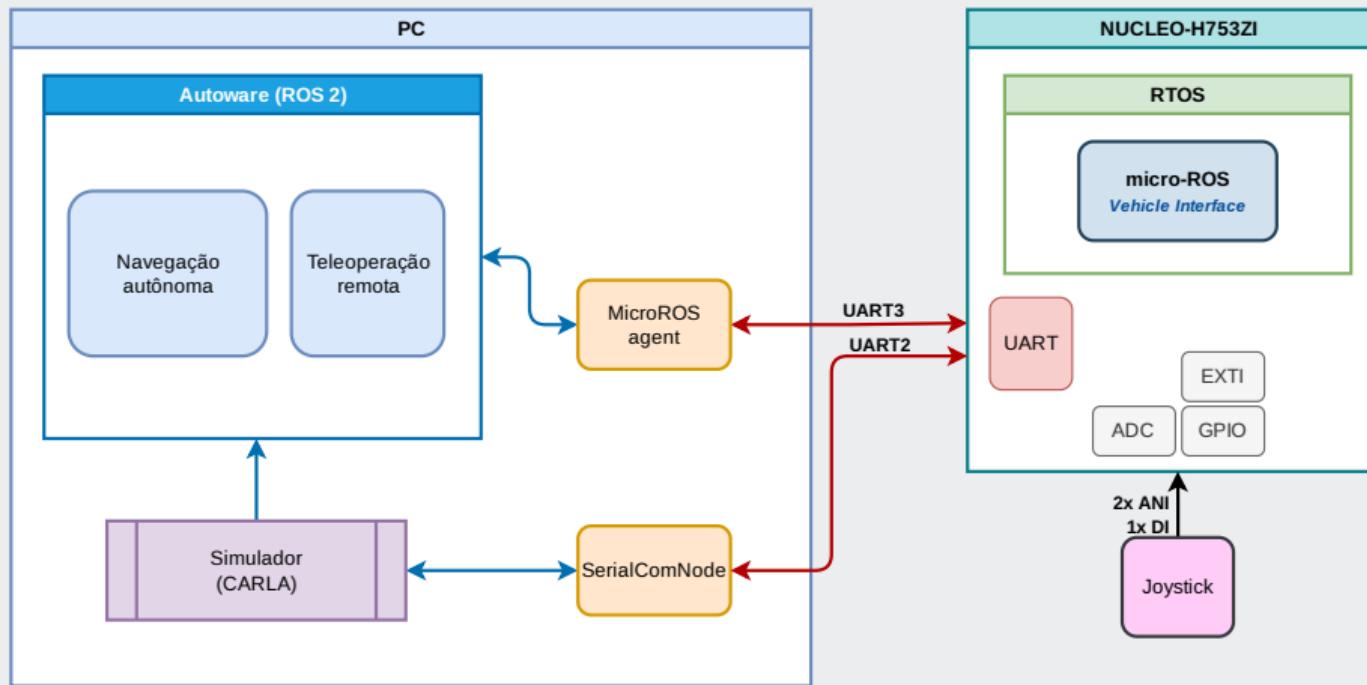


Figura 4: Diagrama de blocos atualizado.



Diagrama de tarefas

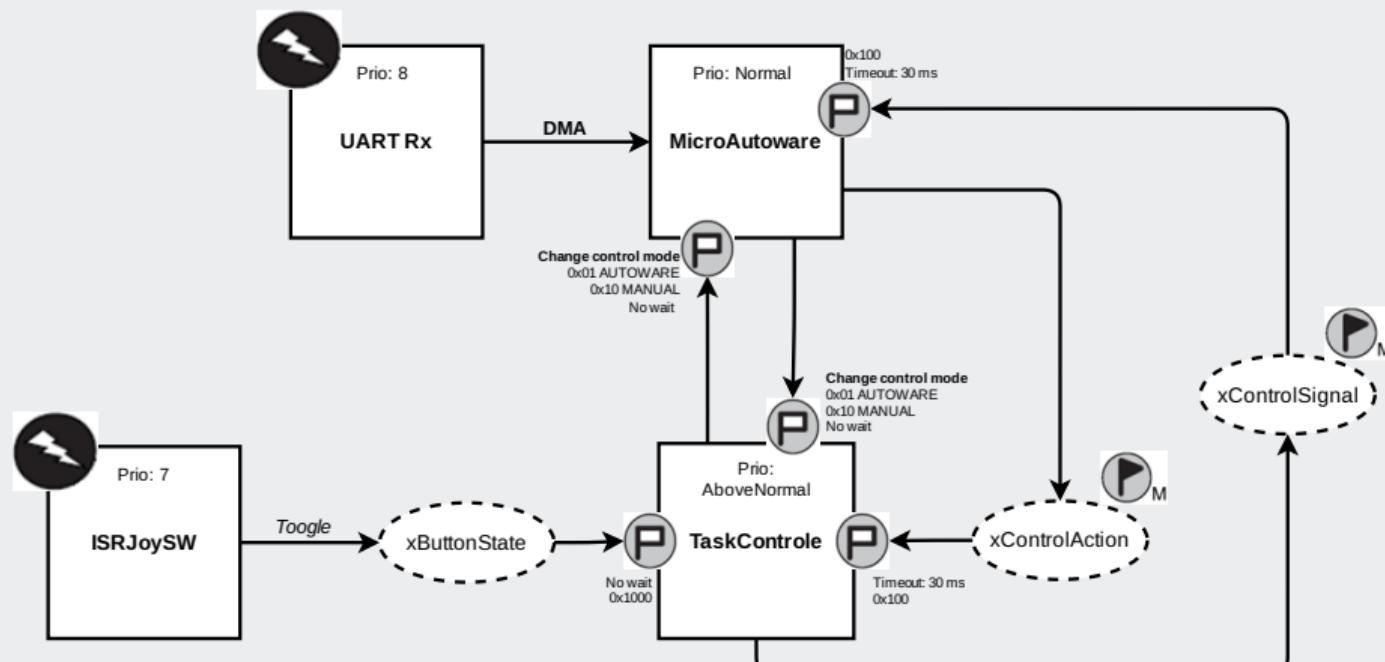


Figura 5: Diagrama de tarefas antigo.

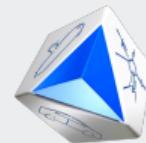


Diagrama de tarefas

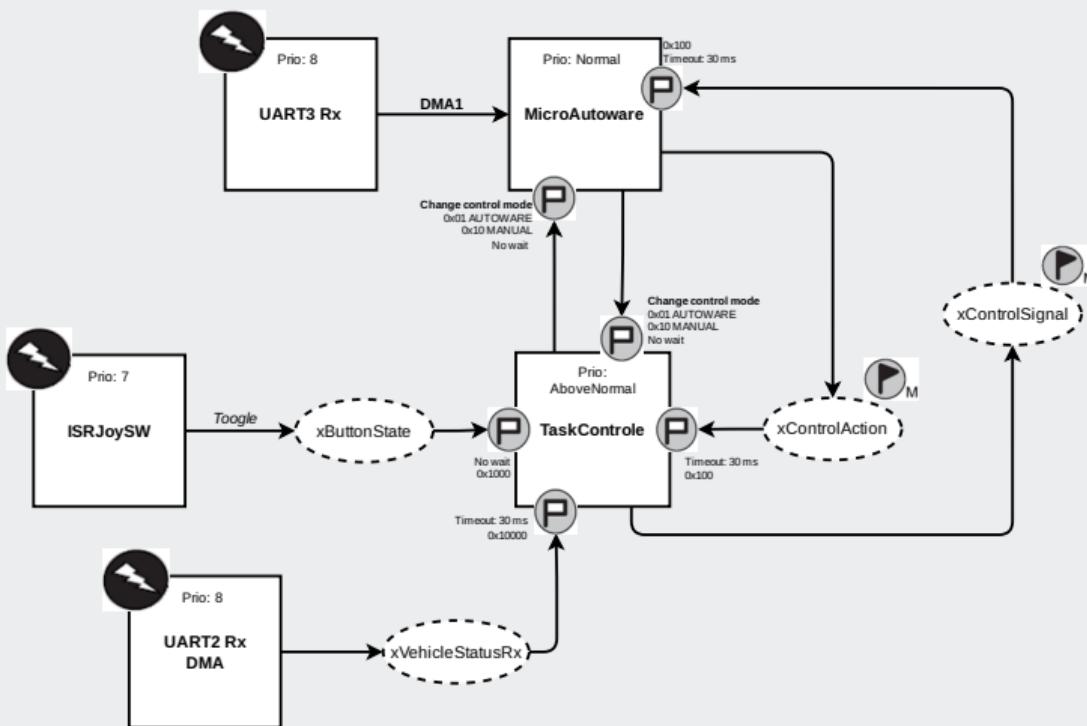
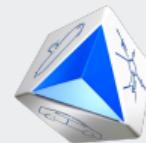


Figura 6: Diagrama de tarefas atualizado.



Comunicação com o simulador

Sistema embarcado → CARLA

- float fTrottle;
- float fBrake;
- float fSteeringAngle;
- unsigned char ucHandBrake;
- unsigned char ucReverse;
- unsigned char ucManualGearShift;
- unsigned char ucGear;
- unsigned char ucControlMode;

CARLA → Sistema embarcado

- float fLongSpeed;
- float fLatSpeed;
- float fHeadingRate;
- unsigned char ucGear;



Máquinas de estados de comunicação

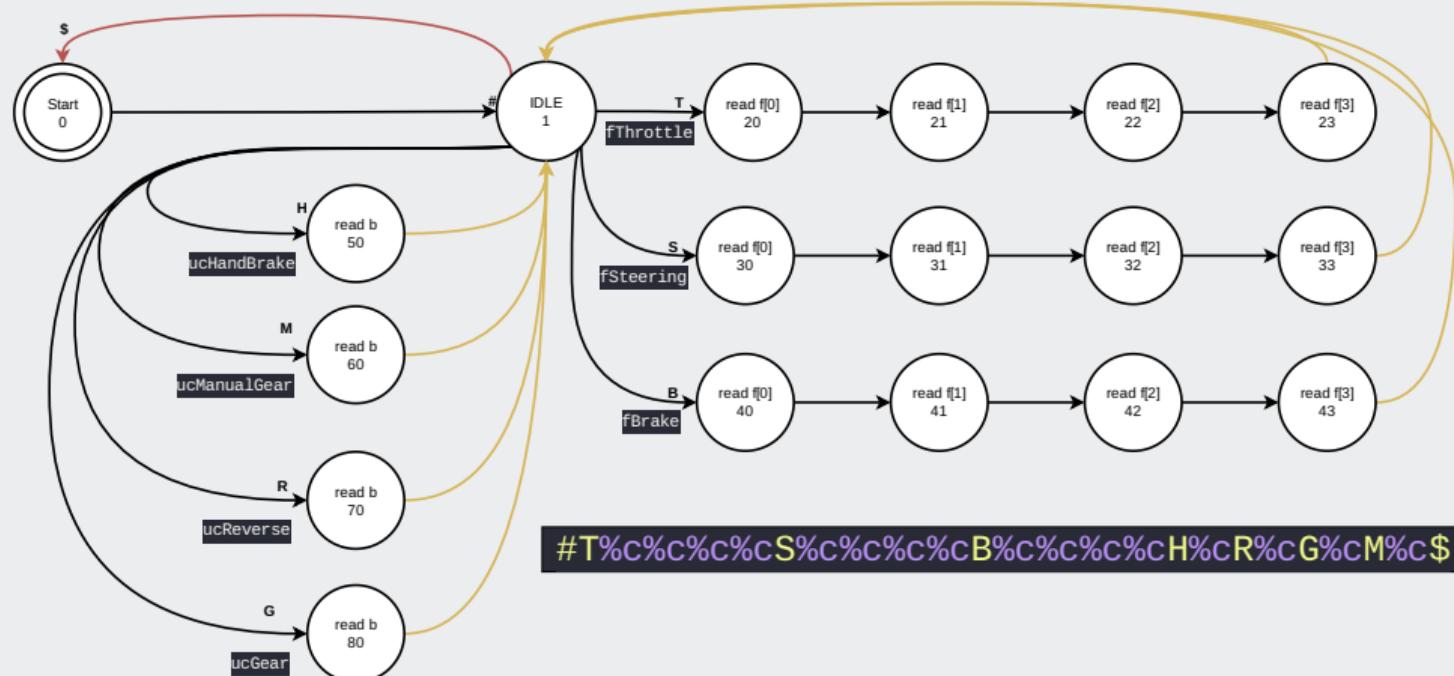
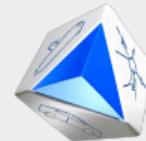


Figura 7: Maquina de estados da comunicação sistema embarcado → CARLA.



Máquinas de estados de comunicação

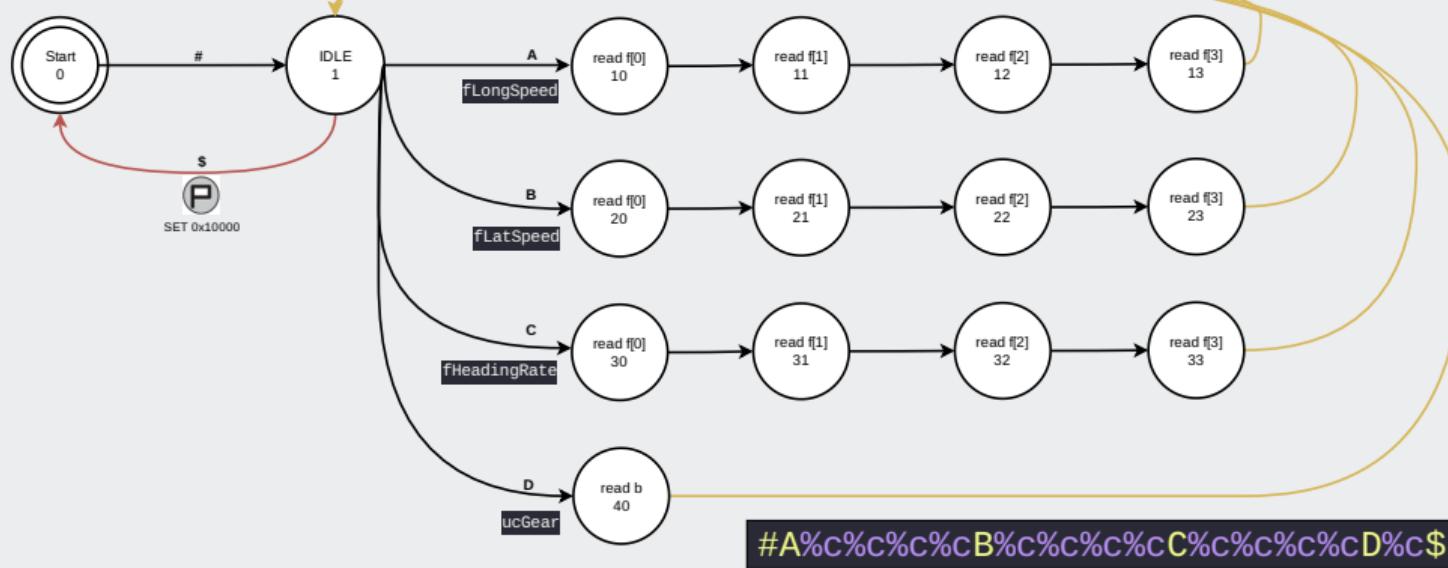
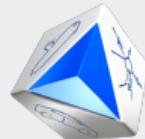
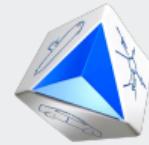


Figura 8: Maquina de estados da comunicação CARLA → sistema embarcado .



Testes dos módulos



Montagem do *hardware*

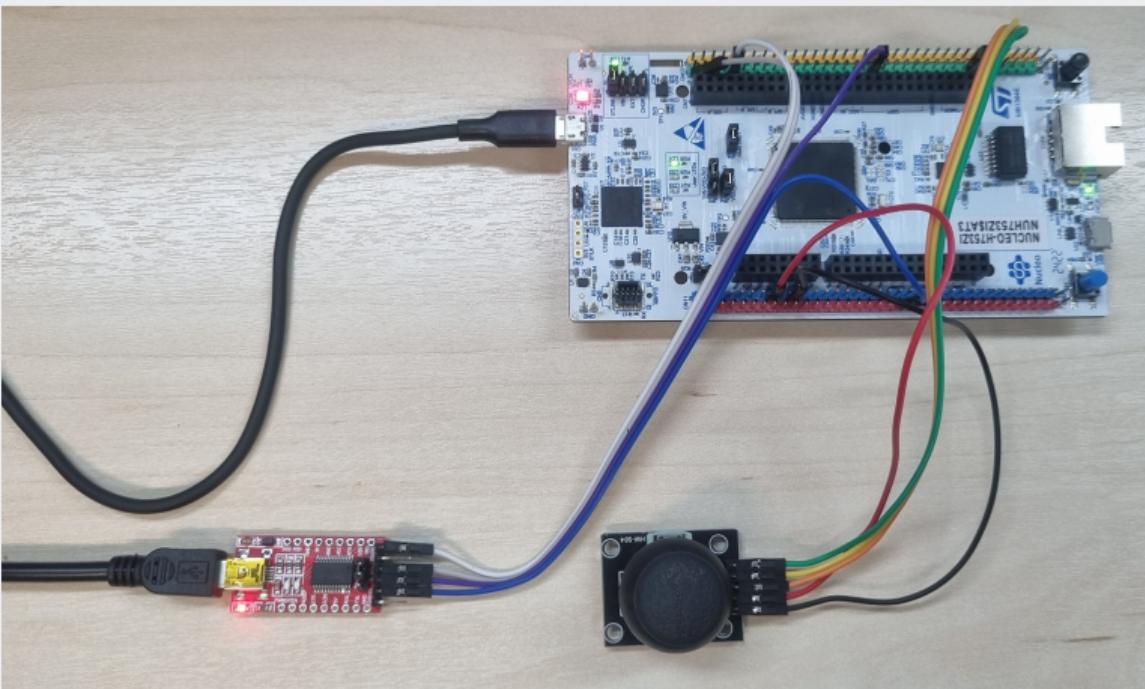
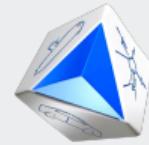
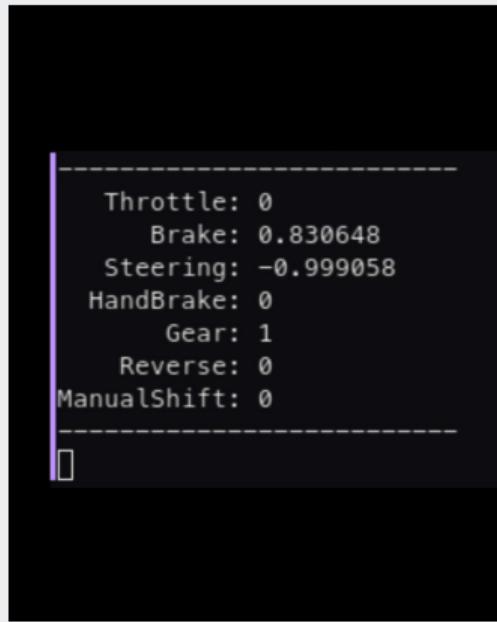


Figura 9: Montagem física dos componentes utilizados.



Leitura do Joystick + Comunicação serial com o CarlaSerialBridge



```
-----  
    Throttle: 0  
    Brake: 0.830648  
    Steering: -0.999058  
    HandBrake: 0  
    Gear: 1  
    Reverse: 0  
    ManualShift: 0  
-----
```

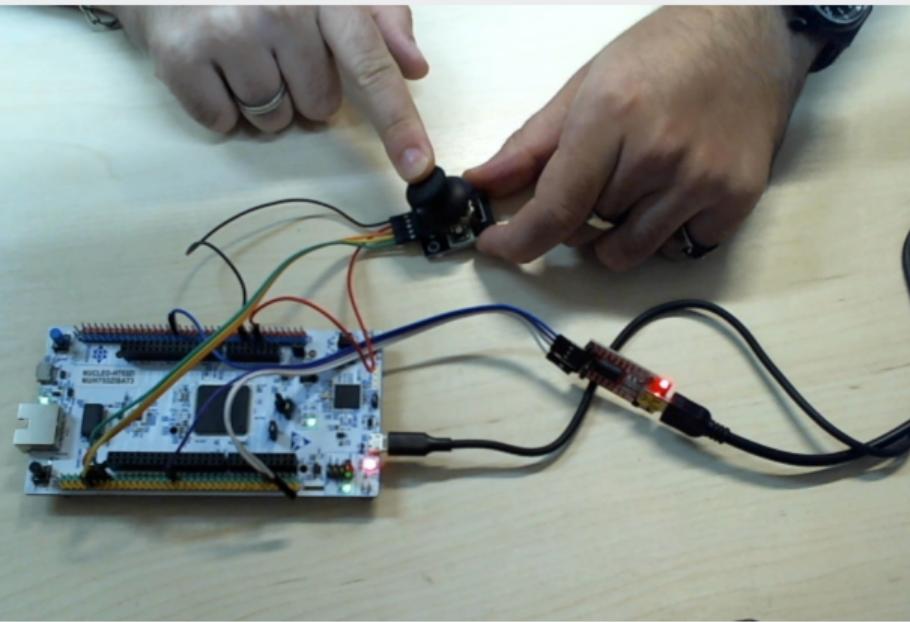
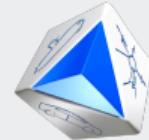


Figura 10: Leitura do Joystick pelo *node* CarlaSerialBridge.



Interrupção JoySW

```
Brake: 0
Steering: -0.000577279
HandBrake: 0
Gear: 1
Reverse: 0
ManualShift: 0
-----
4 Error reading from serial p
4 Error reading from serial p
[]
```

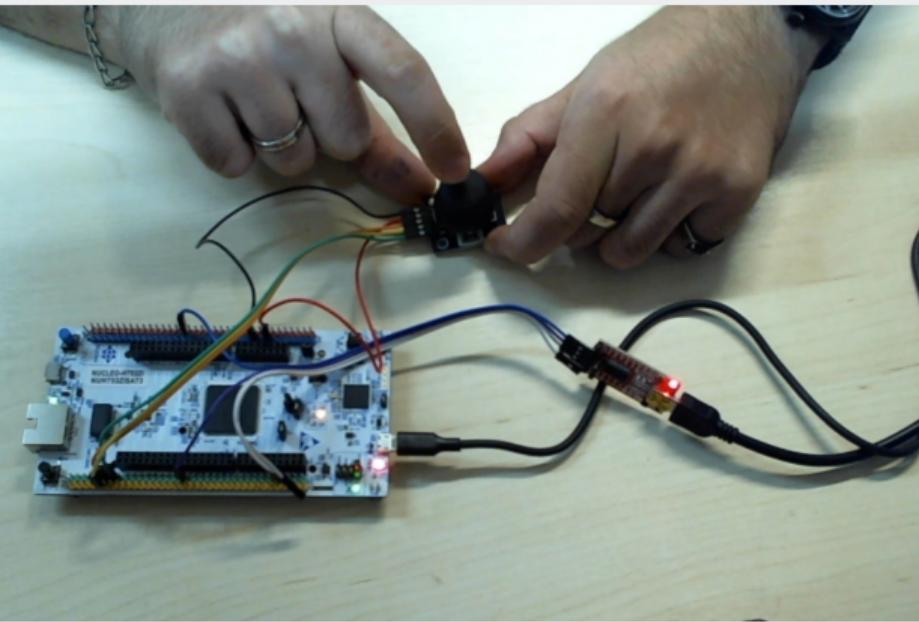
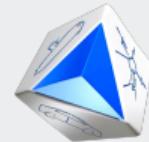


Figura 11: Troca de modo de condução pela interrupção EXTI JoySW.



Modo de operação manual



Figura 12: Validação da comunicação serial com o simulador e controle manual.

micro-ROS

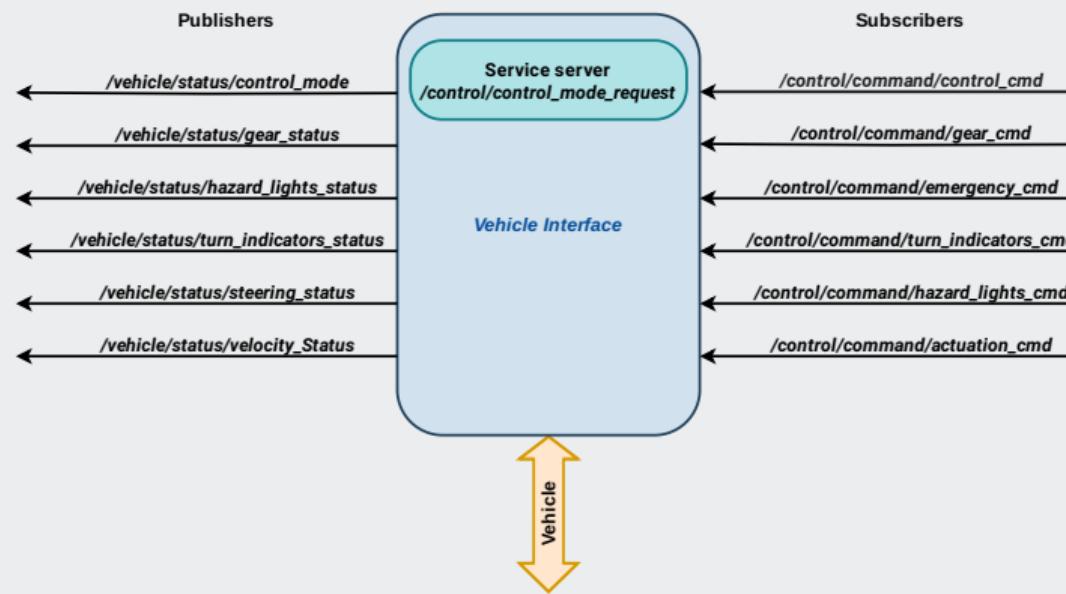
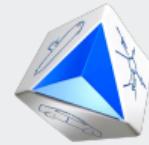


Figura 13: Diagrama de *nodes* e *topics* para a *vehicle interface* construída.



Problemas encontrados



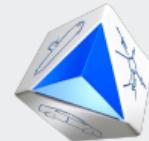
Problemas encontrados

Problemas diagnosticados

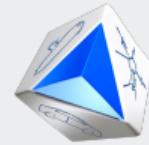
- Perda de ThreadFlags;
- Bounce no botão JoySW;

Problemas à serem verificados

- Escolha dos *timeouts*;



Cronograma

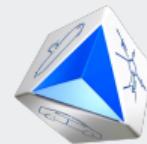


Cronograma

| Atividade/Semana | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
|------------------------------------------------|---|---|---|---|---|---|---|---|---|
| Proposta do projeto | | ■ | | | | | | | |
| Projeto de <i>hardware e software</i> | | | ■ | | | | | | |
| Integração do STM com o micro-ROS | | | ■ | | | | | | |
| Integração do micro-ROS com o Autoware | | | | ■ | ■ | | | | |
| Implementação das tarefas do sistema embarcado | | | | | ■ | ■ | ■ | | |
| Construção do ambiente de testes | | | | | ■ | ■ | ■ | | |
| Realização dos testes | | | | | | ■ | ■ | ■ | |
| Escrita do relatório | | ■ | ■ | ■ | ■ | ■ | ■ | ■ | |

Tabela 1: Cronograma de atividades.

- Semana 2: Apresentação Etapa 1
- Semana 4: Apresentação Etapa 2
- Semana 7: Apresentação Etapa 3
- Semana 9: Apresentação Final



Obrigado!

Dúvidas?

