

PS 2025 - Especificação Projeto de Embarcados

No contexto de competições de robótica autônoma, como a RoboCup, os robôs devem ser capazes de executar tarefas complexas de maneira autônoma e em tempo real. Para que isso seja possível, os sistemas embarcados (*firmware*) desempenham um papel fundamental no controle dos robôs. O *firmware* gerencia a comunicação entre os componentes internos do robô, como sensores, atuadores e microcontroladores, além de coordenar tarefas de controle e tomada de decisão.

No RobôCIn, utilizamos microcontroladores embarcados para processar dados de sensores como **tensão de bateria**, **sensores infravermelhos (IR)**, **carga de capacitores**, **encoders** e sensores inerciais, como **giroscópios** e **acelerômetros**. Essas informações são analisadas em tempo real para ajustar a movimentação do robô, monitorar o status dos componentes e tomar decisões rápidas para a execução de ações. Além disso, o *firmware* gerencia a comunicação entre as funcionalidades de baixo nível dos robôs e o software de inteligência artificial, permitindo a coordenação de tarefas em tempo real.

1. Descrição do problema

Uma funcionalidade essencial para a realização de tarefas por robôs móveis de forma autônoma é a estimativa de sua trajetória, conceito também conhecido como **Odometria**. Tal informação pode ser adquirida realizando medidas sequenciais por sensores presentes no robô e, para isto, diferentes tipos de sensores podem ser utilizados, como Light Detection and Ranging (LiDAR), câmeras, encoders, sensores inerciais (giroscópios e acelerômetros), ou até combinações entre as diferentes abordagens.

Na categoria **Small Size League (SSL)** da RoboCup, os robôs competem em partidas de futebol com robôs que têm o tamanho aproximado de uma bola de futebol. Esses robôs são projetados para realizar uma série de ações autônomas, incluindo movimentação e chute, em um ambiente extremamente dinâmico, exigindo uma capacidade de realizar movimentos precisos em alta velocidade.

Neste contexto, o conceito de odometria inercial, isto é, a odometria obtida através de dados de sensores inerciais, pode ser utilizado para calcular as movimentações realizadas pelo robô, permitindo controlar suas movimentações de forma precisa e rápida. Para isto, podemos utilizar um giroscópio para medir as rotações realizadas pelo robô, por exemplo.

2. Projeto da Seletiva

O principal desafio deste projeto é desenvolver uma solução para determinar o deslocamento angular e a velocidade angular de um robô móvel terrestre utilizando medidas de um giroscópio embarcado no robô.

O giroscópio estará presente na placa do robô, porém será necessário implementar a comunicação com este sensor para receber seus dados. Além disso, as leituras fornecidas pelo giroscópio podem conter ruídos ou serem enviesadas por características intrínsecas do sensor, sendo necessário calibrar tais comportamentos.

Além da leitura e filtragem dos dados, é necessário também converter os valores lidos para a escala de radianos (no caso das variações angulares) e radianos por segundo (no caso das velocidades angulares), para obter medidas que condizem com os comandos de movimentação enviados para o robô. Esta conversão também é uma característica particular de cada modelo de sensor inercial.

Portanto, o grande desafio será implementar a comunicação com este sensor e implementar um método para filtrar e converter os dados recebidos a fim de torná-los medidas utilizáveis pelo firmware do robô para calcular sua movimentação e, conseqüentemente, controlá-lo precisamente.

Objetivo:

O candidato deve desenvolver uma solução para ler os dados fornecidos pelo sensor inercial presente no robô e transformá-los em medidas precisas de velocidade angular (em rad/s) e movimentação angular (em rad). Para isto, é importante também que a implementação colete os dados dos sensores em uma frequência fixa.

O sensor utilizado pela equipe atualmente é [MPU-6050](#), o candidato poderá utilizar o mesmo sensor para desenvolver sua solução ou utilizar outro sensor de sua escolha. O microcontrolador utilizado pela equipe é o [STM32F767ZI](#) (a implantação da solução não precisa estar restrita ao uso do microcontrolador em questão).

Será disponibilizado pela equipe um *log* de testes para que os candidatos possam testar a precisão da sua implementação. O arquivo terá dados de leitura do sensor [MPU-6050](#) coletados em um robô da categoria *Small Size League*.

Restrições:

1. O candidato deve criar um repositório para adicionar a solução desenvolvida;
2. A implementação da solução deve ser feita em C++.

Formato da Entrega Parcial:

1. Um documento PDF de uma página que descreva seu progresso até a entrega parcial, podendo indicar não só o que foi planejado, mas também os assuntos que foram estudados e as abordagens definidas;
2. Arquivos de projeto já desenvolvidos, mesmo que de forma incompleta.

Na entrega parcial, tente responder às seguintes perguntas:

- Você está tendo dificuldades para a realização desse projeto? Se sim, quais?
- Quais abordagens foram estudadas para tratar os problemas mencionados no projeto?
- Quais são os materiais utilizados como referência para o desenvolvimento do projeto?

Formato da Entrega Final:

1. Uma apresentação em *slides* dos estudos, dos resultados obtidos e dificuldades enfrentadas durante o projeto, que será apresentada para a equipe em até 15 minutos;
2. Link para o repositório no GitHub com suas implementações

Na sua apresentação, tente responder às seguintes perguntas:

- Qual protocolo de comunicação foi utilizado para fazer leituras do sensor e por quê? Quais as principais vantagens e desvantagens deste protocolo?
- Quais dados são fornecidos pelo sensor?
- Dentre os dados fornecidos pelo sensor, quais foram utilizados para calcular as movimentações angulares do robô?
- Após a leitura do sensor, quais operações precisaram ser realizadas para transformar os dados adquiridos em medidas de velocidade angular e movimentação angular?
- Como foram calibrados os parâmetros utilizados nestas operações?
- Como foi verificada a precisão do comportamento implementado?
- Qual a frequência de atualização das medições realizadas? Como foi definido este valor para a frequência? Como a solução foi implementada para coletar dados nesta frequência?
- Como você implementaria sua solução para integrar o código de um robô com múltiplas funcionalidades?

- Quais foram os principais desafios ao desenvolver e implementar esta solução e como você superou esses desafios durante o desenvolvimento do projeto?

3. Links Úteis

Ferramentas de desenvolvimento

- <https://code.visualstudio.com/>
- <https://docs.platformio.org/en/latest/frameworks/mbed.html>

Documentação da equipe

- <https://github.com/robocin/ssl-firmware/wiki>

Documentação do sensor:

- <https://invensense.tdk.com/wp-content/uploads/2015/02/MPU-6000-Datasheet1.pdf>

Alguma dúvida? Entra em contato com a gente! 😊