

Universidade Federal do Rio Grande do Sul
Escola de Engenharia
Departamento de Sistemas Elétricos de Automação e Energia
ENG10032 Microcontroladores

Projeto

Prof. Walter Fetter Lages

7 de março de 2018

O objetivo do projeto é desenvolver um *shield* para a Galileo Gen2 capaz de acionar pelo menos uma das juntas do robô Quanser 2DSFJE mostrado na Figura 1. Ou seja, o *hardware* deve ser capaz de acionar o motor ($27V \times 3A$) e ler os sensores (2 *encoders* e 2 sensores de fim de curso) de pelo menos uma das juntas do robô.

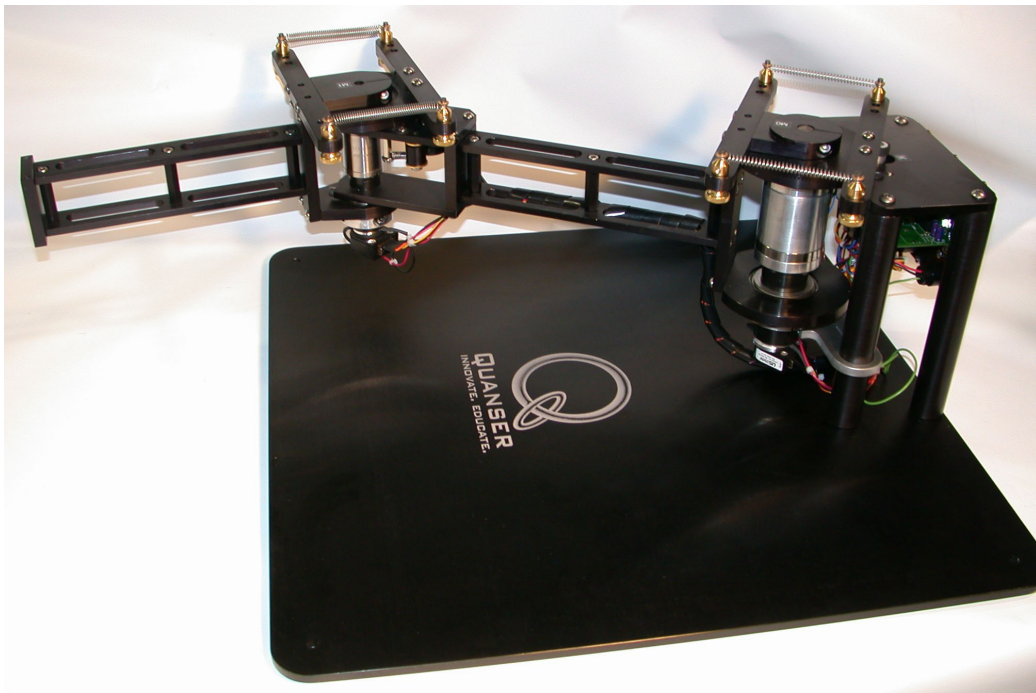


Figura 1: Robô Quanser 2DSFJE.

A montagem final deve consistir em uma placa de circuito impresso para ser encaixada diretamente na Galileo e incluir cabos e conectores para conexão com

o robô e fontes. As fontes de 27 V para alimentação dos motores e de 5 V para alimentação dos *encoders* estão disponíveis no laboratório.

O motor deve ser acionado por PWM (com resolução de pelo menos 8 bits) e os *encoders* devem ser decodificados em quadratura e a contagem, pelo menos parcial, feita em *hardware*, de forma que não haja perda. O acionamento por PWM deve ser tal que um ciclo de trabalho de 0% corresponde ao motor girando em velocidade máxima em um sentido e ciclo de trabalho de 100% corresponde ao motor girando em velocidade máxima no sentido oposto, com uma variação linear entre os dois extremos.

O *software* deverá incluir uma biblioteca, com o(s) respectivo(s) arquivo(s) de cabeçalho, com uma API para ser utilizada pelos desenvolvedores do controlador. Esta API deverá possibilitar o comando do motor em Volts, a leitura dos *encoders* em radianos e a leitura dos sensores de fim de curso. Esta biblioteca deverá executar no sistema Linux como um programa no espaço do usuário, sem privilégios de superusuário. Se necessário poderão existir também módulos do *kernel* para implementar *drivers* ou outras funcionalidades que não possam ter, ou que não seja conveniente que tenham, a sua implementação no espaço do usuário. Deve ser feito um *script* de inicialização para configurar as permissões necessárias e carregar eventuais módulos do *kernel* que sejam necessários.

Para demonstrar o funcionamento da biblioteca deve ser feito um programa no espaço do usuário que implemente um controlador PID. Para tanto, as molas das juntas do robô podem ser travadas, de forma a eliminar a flexibilidade das juntas. Ou seja, a tensão a ser aplicada no motor no instante t é dada por:

$$u(t) = K_p e(t) + K_i \int_0^t e(t) dt + K_d \frac{d}{dt} e(t) \quad (1)$$

onde $u(t)$ é a tensão aplicada no motor, $e(t)$ é o erro dado por $e(t) = r(t) - y(t)$, sendo $r(t)$ a posição desejada para a junta, a ser informada através do terminal, $y(t)$ é a posição angular da junta (que como as molas estão travadas deve ser a mesma em ambos *encoders*), K_p é o ganho proporcional, K_i é o ganho integral e K_d é o ganho derivativo. O ajuste dos ganhos pode ser feito por um método empírico, como Ziegler-Nichols ou método do ganho crítico, entre outros. Vide, por exemplo, [1] para uma descrição destes métodos.

A avaliação do projeto inclui tanto o *hardware* projetado quanto o *software* necessário para demonstração do seu funcionamento e o *firmware* dos dispositivos programáveis utilizados, se for o caso. Em especial, o adequado particionamento do *software* entre espaço do *kernel* e espaço do usuário faz parte da avaliação. Os nodos no `/dev` que forem necessários devem ser criados pelo próprio *driver*, para o que deve ser criado também o arquivo de regras para o `udev`. A tabela 1 detalha o critério de avaliação.

Os trabalhos poderão ser feitos em grupos de até 3 alunos.

Tabela 1: Critério de Avaliação

Item	Esquemático/Código	Funcionamento
Geração do sinal de PWM	0.5	0.5
Ponte H e acionamento dos motores	0.5	0.5
Decodificação em quadratura e contagem	0.5	0.5
Leitura dos sensores de fim de curso	0.5	0.5
Rotina de acionamento em Volts	0.5	0.5
Rotina de leitura dos <i>encoders</i> em radianos	0.5	0.5
Rotina de PID	0.5	0.5
Netlist	0.5	
Gerbers	0.5	
Documentação do <i>software</i>	0.5	
<i>Script</i> de inicialização	0.5	
Qualidade da montagem	0.5	
Estrutura de diretórios/bibliotecas	0.5	

O relatório do projeto deve incluir os esquemáticos do *hardware* do projeto e a sua descrição. Estes diagramas esquemáticos deverão ser feitos em um sistema de CAD apropriado para captura de esquemáticos e deverão estar em folhas A4 ou A3. Os esquemáticos deverão ser tais que um técnico seja capaz de montar o circuito projetado a partir deles. Sugere-se a utilização do gEDA <<http://www.geda-project.org>> para desenvolvimento e documentação do *hardware* e a utilização do Doxygen <<http://www.doxygen.org>> para geração da documentação do *software*.

Uma versão prévia do esquemático deverá ser postada no Moodle até o dia 09/05/2018. Recomenda-se que sejam feitos esforços para que esta versão prévia já contemple todos os requisitos de *hardware* do projeto, de forma que haja tempo hábil para corrigir eventuais problemas.

A postagem do relatório final do projeto, em formato PDF, no sistema Moodle em <<http://moodle.ece.ufrgs.br>>, deverá ser feita até 04/07/2018. Até esta mesma data deverá ser demonstrado o funcionamento do projeto. A demonstração só será apreciada se o correspondente relatório tiver sido submetido pelo grupo através do sistema Moodle. Caso contrário será considerado que o grupo não entregou o trabalho para avaliação.

Juntamente com o relatório final, deverão ser postados no Moodle:

- Esquemáticos em PDF, em folhas A4 ou A3

- Arquivo de netlist gerado pelo CAD
- Arquivos de BOM (*bill of materials*) gerado pelo CAD
- Arquivos Gerber do *layout* do PCB
- Código fonte do *Software* e *firmware* desenvolvido

O manual de referência do robô e outros projetos de *hardware* semelhantes para acionamento de motores e sensores estão disponíveis no Moodle para referência, assim como os *datasheets* de diversos componentes que podem ser úteis para o projeto.

Referências

- [1] A. S. Bazanela and J. M. Gomes da Silva Jr. *Sistemas de Controle: Princípios e Métodos de Projeto*. Editora da UFRGS, Porto Alegre, 2005.