

Sistemas Elétricos e Eletrónicos de Veículos

Aula Laboratorial nº 6

Objetivos:

- Projeto de circuitos elétricos e *software* para o controlo de sistemas eletromecânicos em veículos automóveis;
- Estudo do funcionamento e afinação de controladores digitais do tipo PI;
- Implementação prática dos pontos anteriores.

Lista de material fornecido:

- Kit de desenvolvimento Curiosity (PIC18F46K80);
- Arduino UNO para comunicação série;
- Componentes elétricos: resistência de 330Ω , resistência de $4,7k\Omega$, transístor MOSFET IRF530 (com dissipador de calor) e CI de isolamento ótico TLP521-1;
- Kit com sistema eletromecânico (Anexo A);
- 2x cabos banana-crocodilo.

- 2.

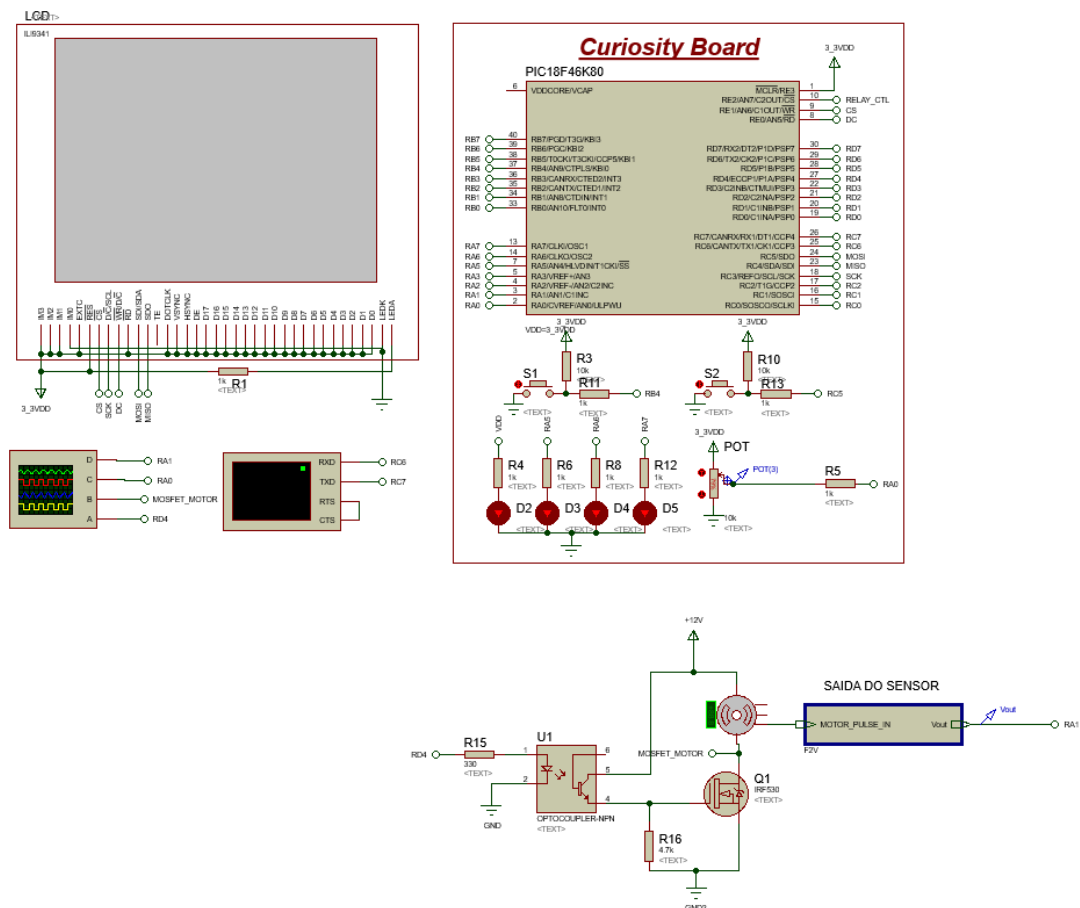


Figura 1 – Circuito para o controlo de um atuador eletromecânico.

Sem efetuar qualquer simulação (em Proteus), responda às seguintes questões:

- Identifique as saídas e entradas do microcontrolador utilizadas. Qual a respetiva função?
- Que periféricos do microcontrolador são utilizados no programa disponibilizado? Quais os respetivos objetivos/funcionalidades?
- O circuito/programa funciona em malha aberta ou malha fechada? Justifique!
- O que acontece se o microcontrolador recebe um *byte* no porto RX?
- Apresente o fluxograma do programa implementado.

3. Compile o código disponibilizado e teste o funcionamento do projeto em simulação (software Proteus).

Atenção: dadas várias limitações (velocidade processamento da simulação, modelação do sistema eletromecânico, etc..) não é possível simular o controlo do atuador eletromecânico em tempo real. Deve, no entanto, explorar a simulação de todas as funcionalidades implementadas em código e validar o funcionamento, ou pelo menos a reação esperada do mesmo.

--- 2ª Parte ---

4. **Implementação prática.** A implementação prática do controlo do atuador eletromecânico será realizada por fases. Só deve avançar para a fase seguinte, caso tenha concluído e compreendido totalmente os resultados obtidos das fases anteriores. Caso não cumpra as instruções neste enunciado, pode danificar irreversivelmente os componentes e sistemas utilizados. Não serão disponibilizados novos sistemas nesse caso.
5. **1ª Fase – Sensores e comunicação.** Ligue a placa de desenvolvimento aos terminais série da placa arduino (PIC18 **TX** -> Arduino UNO **TX**; PIC18 **RX** -> Arduino UNO **RX**) e o sensor do sistema eletromecânico (Anexo A).

Não monte o circuito isolador ótico, o transistor MOSFET nem a parte de potência do atuador eletromecânico. Não necessita de montar o interruptor e potenciômetro RV1, já disponíveis na placa de desenvolvimento.

Abra o monitor série da aplicação *ARDUINO IDE*, de forma a poder comunicar com a placa através do PC (terminal), utilizando a seguinte configuração: Placa "Arduino UNO", Porta X; Bits por segundo / baudrate = 9600; Controlo de fluxo = Nenhum.

Será útil alterar a opção "Sem fim de linha".

Verifique:

- a) A correta leitura dos sensores (atuador e potenciômetro de referência) no terminal (atue o atuador eletromecânico à mão). Não se preocupe caso a gama de funcionamento do sensor não vá dos 0% aos 99%;
- b) A configuração de parâmetros via terminal;
- c) A variação do fator-de-ciclo (DC) em função do Erro e do estado do interruptor.

6. **2ª Fase – Controlo proporcional.** Desligue a placa de desenvolvimento. Complete as ligações da eletrónica de controlo do sistema eletromecânico (Anexo A) de acordo com a Figura 1 (circuito isolador ótico, transistor MOSFET e resistências R15 e R16).

Atenção: a tensão de alimentação **VCC12V** e massa **GND2** da fonte de alimentação do atuador não ligam nem a **VCC5V / +V** nem à massa (**GND**) da placa de desenvolvimento.

Alimente a placa de desenvolvimento, ligue a fonte de alimentação de 12V e teste a resposta do atuador eletromecânico ao potenciômetro de referência.

Cuidado: Não permita que o atuador eletromecânico fique em esforço na posição máxima (consumo de corrente elevado) durante muito tempo, deve reduzir a posição de referência (potenciômetro POT).

Verifique:

- a) A resposta do atuador eletromecânico em função do valor de KP (incremente em passos pequenos o valor de KP via terminal);
- b) Qual o valor de KP para o qual a resposta do atuador é criticamente estável (KP crítico)? (Caso não atinja a estabilidade crítica do atuador, retire no código o fator 1/10 na aplicação da constante proporcional);
- c) A capacidade do controlador proporcional em eliminar o erro entre o valor de referência e o valor dado pelo sensor;
- d) De modo geral, a relação do valor de KP com o erro entre o valor de referência e o valor real (dado pelo sensor do atuador eletromecânico).

--- 3ª Parte ---

7. Implementação de um controlador PI (proporcional e integral). Altere o código dado de modo a implementar um controlador PI, consulte a documentação disponível na página Moodle da UC [5 e 6]. As alterações ao código também devem permitir:

- Reprogramar o valor de KI (constante do controlador integral) pelo porto série (de modo idêntico a KP);
- A visualização do somatório dos erros via porto série (junto às restantes variáveis, fica: SetPoint, Sensor, Erro, SomaErros e DC).

Outras sugestões:

- Limite o valor do somatório dos erros (integral do erro) em $[0, X]$, comece por utilizar $X=100$;
- Faça um *reset* ao somatório dos erros sempre o controlador estiver desativo.

8. Teste e verifique a funcionalidade das funções implementadas em *software* Proteus.

9. Não avance para esta questão caso não tenha o controlador PI, com as funcionalidades pedidas, a funcionar totalmente em simulação.

Teste a implementação prática do controlador PI desenvolvido. Comece por utilizar valores baixos para as constantes, proporcional e integral, do controlador.

Atenção: caso o atuador eletromecânico se fixe na posição máxima, com elevado consumo de corrente, desligue imediatamente a fonte de alimentação do mesmo (12V).

Verifique:

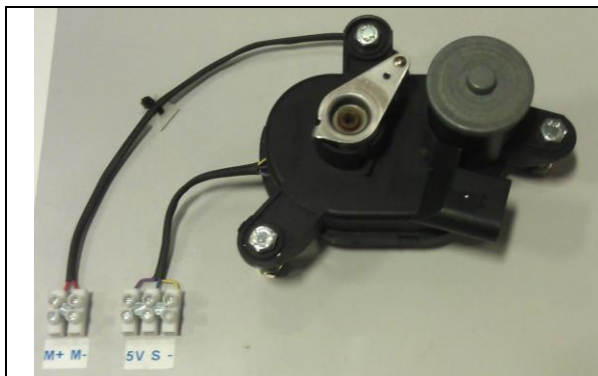
- A resposta do atuador eletromecânico em função do valor de KI (utilize um valor para KP igual a $0.4 \times K_P$ crítico);
- Qual o valor de KI para o qual a resposta do atuador é instável?
- A possibilidade de sintonizar o controlador PI de acordo com as metodologias apresentadas em "*Digital PID Controllers*" [5];
- A capacidade do controlador PI em eliminar o erro entre o valor de referência e o valor dado pelo sensor.

--- Anexo A ---

Sistemas eletromecânicos a utilizar na implementação prática da aula nº 8.

Ligações comuns:

Atuador (motor DC)	
M+ ou +V	12V
M- ou V-	Dreno MOSFET
Sensor	
5V	+V (placa de desenvolvimento)
S	Saída do sensor (RA1)
-	GND (placa de desenvolvimento)



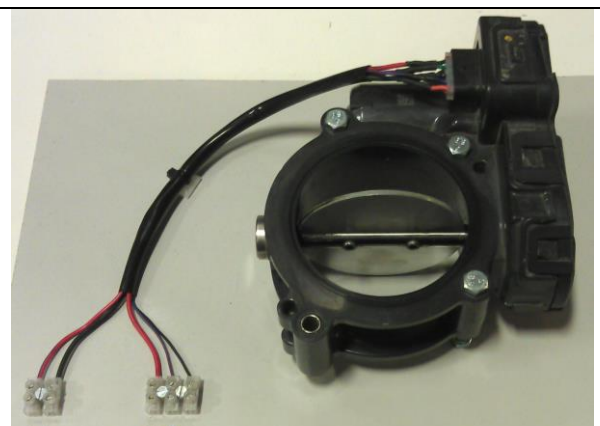
Processo A



Processo B



Processo C – utilizar a saída S2



Processo D