

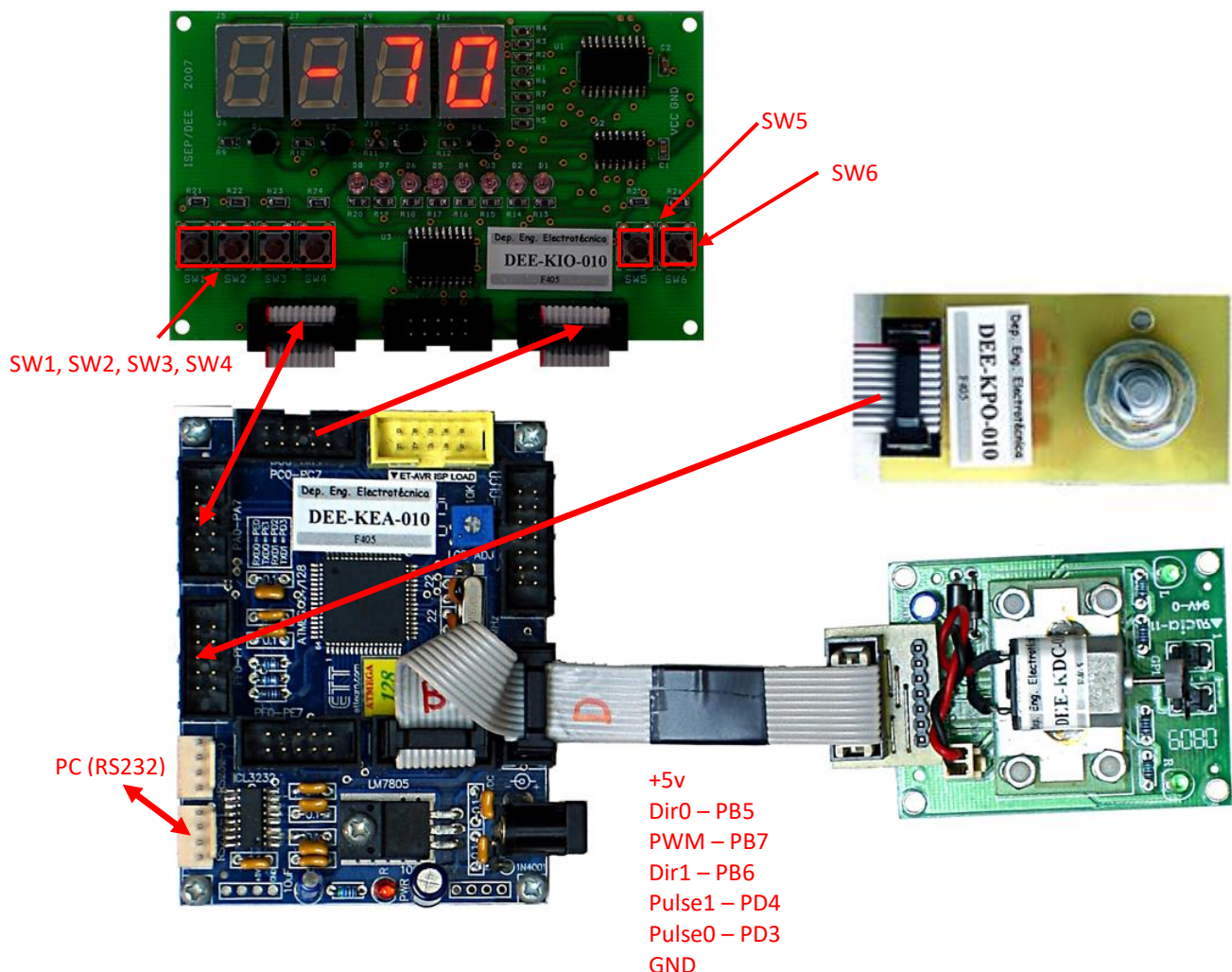
Trabalho 3

Objectivo: Programação em linguagem C, controlo de velocidade de um motor DC em malha aberta, comunicação série assíncrona e conversor A/D.

Necessário: Conhecimento linguagem C, conhecimentos de temporizadores contadores, interrupções, USART, conversor A/D e motor passo a passo (MPP).

Funcionamento 1: pretende-se controlar a velocidade de um motor DC utilizando PWM. Os interruptores **SW1**, **SW2**, **SW3**, **SW4** permitem definir as velocidades de rotação do motor com os valores 25%, 50%, 70%, 90% da velocidade nominal, respectivamente. Sempre que for accionado um dos interruptores o valor percentual da velocidade de rotação do motor deverá ser apresentado no *display*. **SW5** inverte o sentido de rotação e **SW6** deve parar o motor.

Hardware a utilizar:



Detalhes do funcionamento:

O valor percentual da velocidade de rotação do motor deverá ser apresentado no *display*. No caso da rotação do motor ser no sentido anti-horário deve ser activado o sinal (-) no *display* 2. Para inverter o sentido de rotação do motor deve-se, previamente, parar o motor durante **500 ms**.

Sugestão: Utilizar o **TC0**, programado em modo **CTC**, para obter uma temporização base de 5ms

Utilizar o **TC2**, programado em modo **PWM (Fast PWM ou Phase Correct PWM)**, para gerar o sinal de PWM na saída **OC2 (PB7)**. A frequência deste sinal de PWM deve ser de, aproximadamente, **500 Hz**. O sentido de rotação será definido através das saídas **Dir0 (PB5)** e **Dir1 (PB6)**.

Implementação do software:

Implementar o software em linguagem C

Funcionamento 2:

Utilizando a **USART1** para uma comunicação série assíncrona (RS232), **4800 bps**, **8 bits** de dados e **1 stop bit**, pretende-se fazer o controlo da velocidade do motor através do PC. Os caracteres a enviar para o ATmega 128 são os seguintes:

Carácter	Descrição
"P"	Motor parado
"1"	25% da velocidade nominal (sentido definido por SW5)
"2"	50% da velocidade nominal (sentido definido por SW5)
"3"	70% da velocidade nominal (sentido definido por SW5)
"4"	90% da velocidade nominal (sentido definido por SW5)
"I"	Inverter sentido de rotação
"C"	Pedido de envio do <i>duty cycle</i> actual do motor (em percentagem)

Funcionamento 3:

Pretende-se alterar o funcionamento 1 por forma a controlar a velocidade do motor através do valor da tensão aos terminais de um potenciómetro, de tal maneira que a variação de **0 V** a **Vcc** provoca um incremento da velocidade de 0 RPM até à velocidade nominal (0 V deve corresponder a 0% da velocidade nominal e Vcc a 99% da velocidade nominal).

Utilizando a **USART1** para uma comunicação série assíncrona (RS232), pretende-se adicionar dois novos comandos para seleccionar o modo de funcionamento do sistema. O carácter "**A**" selecciona o modo de funcionamento analógico (controlado pelo potenciómetro) e o carácter "**D**"

seleciona o modo de funcionamento digital (controlado pelo PC). O sistema deve começar com o modo de funcionamento digital seleccionado.

Aquisição de dados analógicos:

A conversão analógica/digital da tensão produzida pelo potenciómetro deve ser obtida a partir da média de **4** leituras consecutivas do conversor A/D, com uma precisão de **10 bits**.

A rotina de aquisição de dados deve ser implementada em Assembly.

Implementação do software:

Utilizar **linguagem C + linguagem Assembly** (rotina de aquisição de dados)

Funcionamento 4:

Pretende-se controlar a posição de um motor passo-a-passo (20 passos / rotação ou 40 meios passos / rotação) através de comandos enviados do PC. O carácter “**R**” faz uma rotação para a direita e o carácter “**L**” faz uma rotação para a esquerda. O veio do motor pode deslocar-se de -180° a +180°. O carácter “**S**” deve permitir a definição da posição **0°** do motor. O carácter “**Z**” deve rodar o motor para a posição 0° definida com o carácter “**S**”. O tempo entre 2 passos consecutivos do motor deve ser aproximadamente 25 ms.

Os caracteres a enviar para o ATmega 128 são os seguintes:

Carater	Descrição
“R”	Rotação 1 passo para a direita
“L”	Rotação 1 passo para a esquerda
“Z”	Rodar para a posição 0°
“S”	Definição da posição 0°

Tabelas de acionamento do motor passo-a-passo:

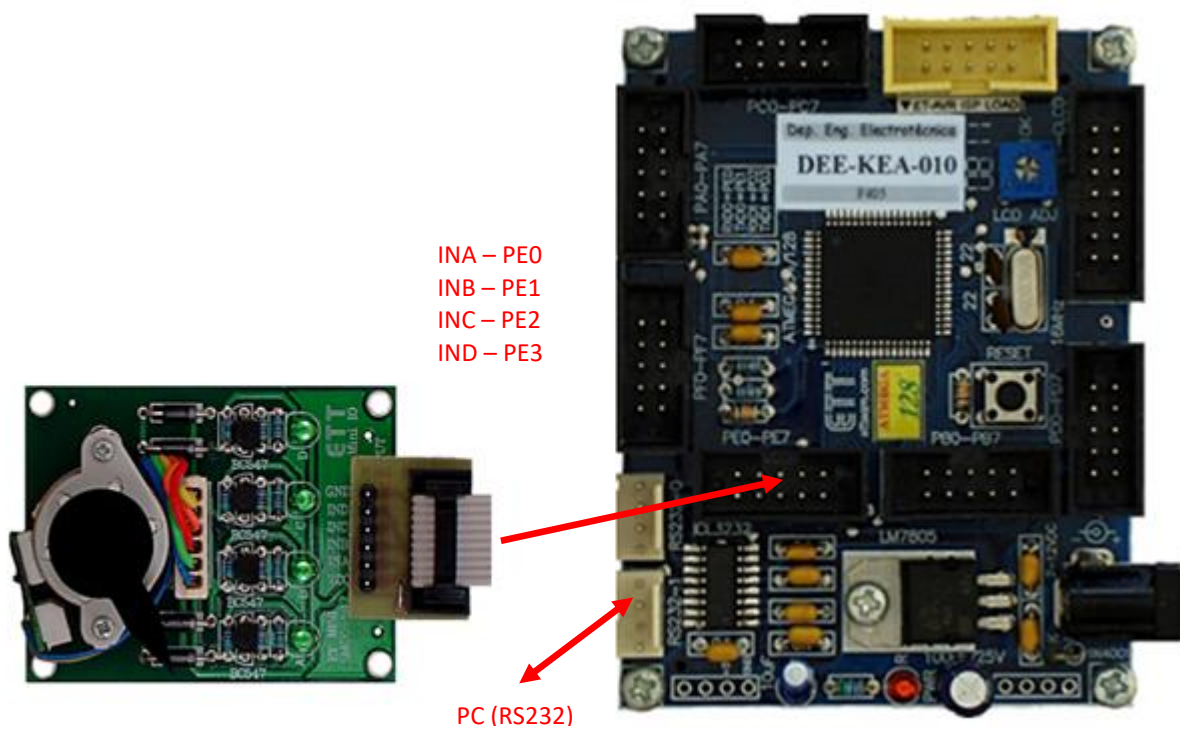
- Acionamento com passo completo (*full step*)

IND	INC	INB	INA
1	0	0	1
1	1	0	0
0	1	1	0
0	0	1	1

- Acionamento com meio-passo (*half step*)

IND	INC	INB	INA
0	0	0	1
1	0	0	1
1	0	0	0
1	1	0	0
0	1	0	0
0	1	1	0
0	0	1	0
0	0	1	1

Hardware a utilizar:



Implementação do software:

Implementar o software em **linguagem C**