

SEL0629

Aplicação de Microprocessadores I

Prática 4 PWM e Sensor de Velocidade

Prof. Marcelo Andrade da Costa Vieira



PWM e Sensor de Velocidade

- Objetivos:
 - Aprendizado do microcontrolador PIC18F45k22
 - Uso do Compilador - Mikro C Pro for PIC
 - Linguagem C
 - Estudo das bibliotecas do Mikro C
 - Uso do PWM do PIC e display de LCD

PWM e Sensor de Velocidade

- Prática:
 - Usar 5 botões para variar o ciclo de trabalho (*duty cycle*) do PWM e girar a ventoinha do kit em 5 velocidades diferentes.
 - Calcular a velocidade de giro da ventoinha (RPM) para cada um dos casos.
 - Mostrar o valor do duty cycle e RPM no display de LCD

Módulo CCP



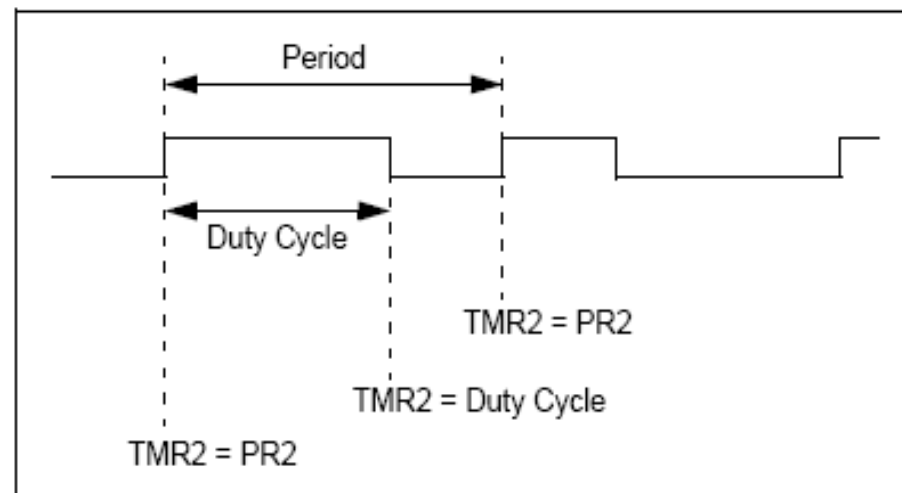
Módulo CCP

- CCP = Capture, Compare, PWM;
- Periférico presente em alguns microcontroladores PIC
- Modo Capture: contagem de tempo entre dois eventos ocorridos no pino do PIC (borda de descida ou subida)
- Modo Compare: contagem de tempo entre dois eventos ocorridos no pino do PIC e comparação com um valor pré determinado
- Modo PWM: geração de um pulso PWM no pino do PIC
- Pode gerar interrupção
- Utiliza os temporizadores do PIC para geração da base de tempo:

CCP/ECCP Mode	Timer Resource
Capture Compare PWM	Timer1 or Timer3 Timer1 or Timer3 Timer2

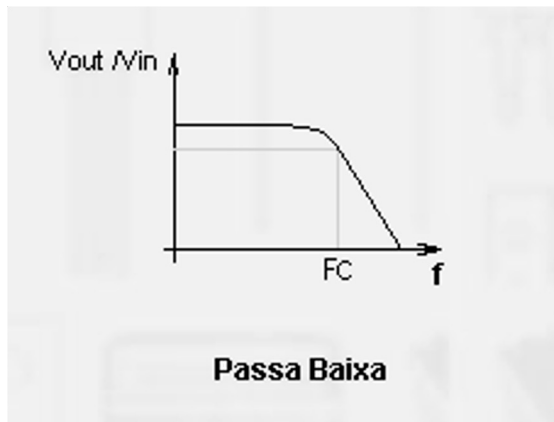
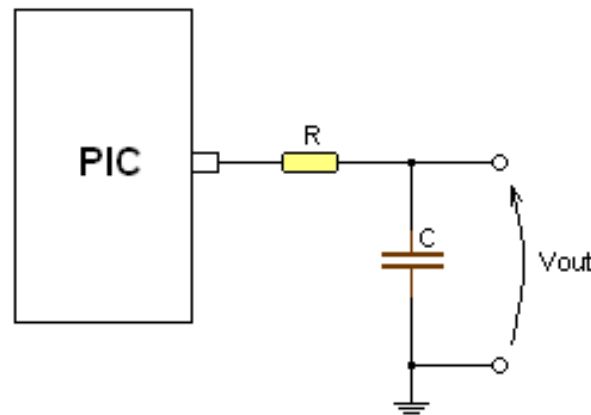
PWM

- *Pulse Width Modulation* = Modulação por largura de pulso
- Onda de frequência constante mas com largura de pulso variável (ciclo de trabalho ou *duty cycle*)
- Obtenção de uma tensão analógica a partir de um sinal digital (conversor D/A)



PWM

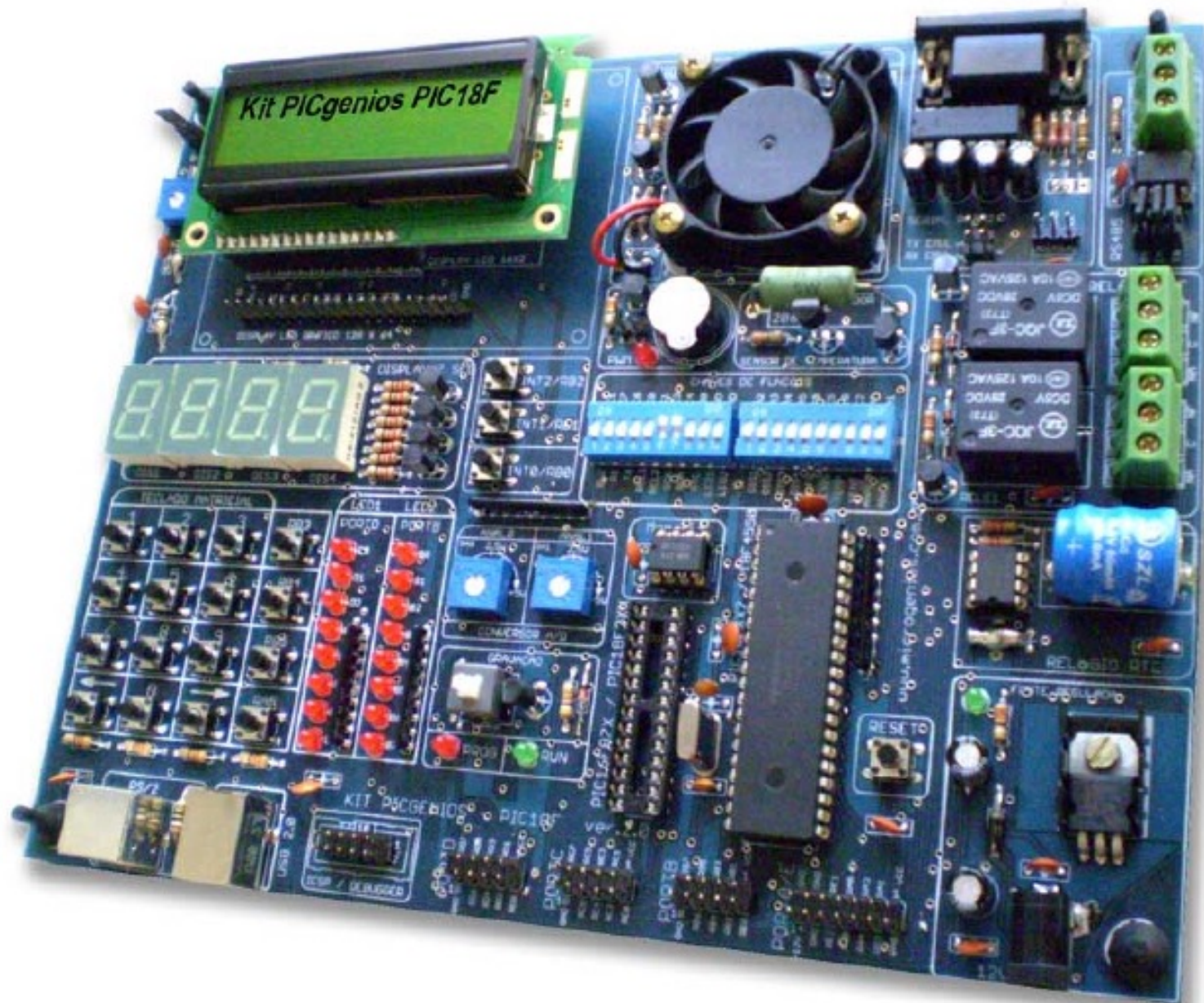
- Uso um filtro passa baixa com frequência de corte menor do que a frequência do PWM



$$f_c = \frac{1}{2\pi RC}$$

Kit Pic Genios





Periféricos

Ventoinha para simulação de controle de rotação via PWM. Podemos controlar a velocidade de giro da ventoinha através do canal CCP1 do PIC.

Sensor infravermelho (emissor e receptor) para leitura da velocidade de rotação da ventoinha. Através dos pulsos gerados por este sensor, podemos ler os pulsos através do canal de contagem RC0 do PIC.

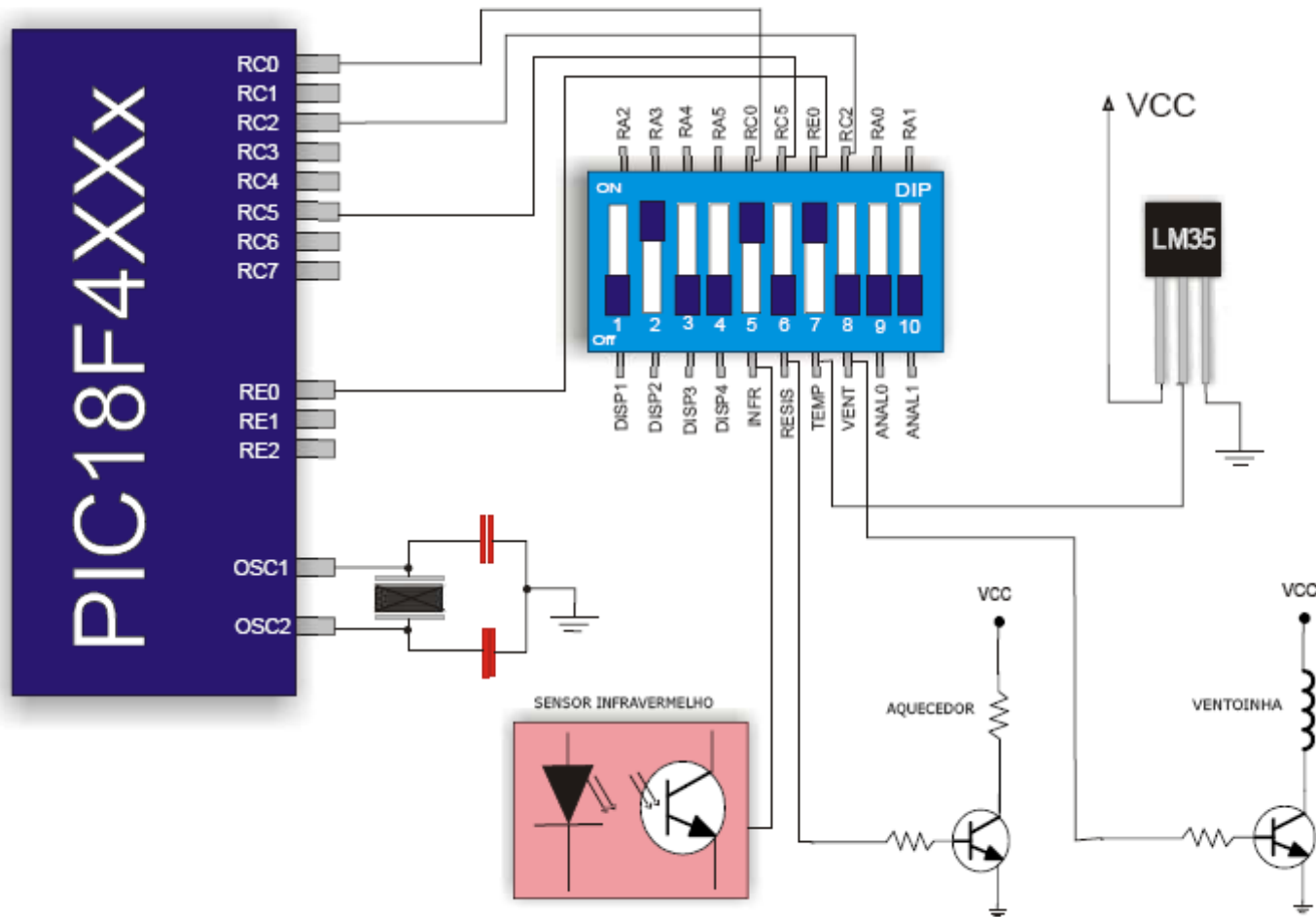
No kit temos um **buzzer** para acionamento de alarme sonoros. Para acionar o buzzer basta enviar nível lógico 1 no pino RC1, mas antes devemos habilitá-lo através da chave seletora..



A ventoinha tem 7 pás!

Resistência de aquecimento. Podemos ligar ou desligar a resistência com o intuito de variar a temperatura ao redor do sensor Lm35, desta forma podemos desenvolver diversas aplicações de práticas utilizando o aquecedor e o sensor de temperatura.

Sensor de temperatura Lm35 ligado ao canal A/D AN2 do PIC. Através deste sensor podemos ler a temperatura ambiente ou a temperatura da resistência.



BUZZER



Descrição das pinagens de ligação dos periféricos

Pino	Descrição
RE0	Sensor de temperatura LM35
RC2	Ventoinha (cooler)
RC5	Resistência de aquecimento
RC1	Buzzer - via Jumper
RC0	Sensor infravermelho (tacometro)


Parte 1:

PWM

- Programar em linguagem C e estudar as bibliotecas disponíveis no Mikro C Pro.
- Utilizar 5 botões do kit Easy Pic para controlar a velocidade da ventoinha do Kit Pic Genius via PWM:
 1. 0%
 2. 25%
 3. 50%
 4. 75%
 5. 100%
- Mostrar o valor do duty cycle escolhido no display LCD

Parte 2:

Sensor Infravermelho

- Programar em linguagem C e estudar as bibliotecas disponíveis no Mikro C.
 - Utilizar os botões do kit para ajustar o ciclo de trabalho (duty cycle) do PWM e girar a ventoinha (parte 1).
 - Utilizar o **sensor infravermelho** para calcular a velocidade de giro da ventoinha **em RPM** (rotações por minuto).
 - Utilizar os temporizadores do PIC e interrupção.
 - Mostrar o valor lido no display de LCD
- 

Prática 4

- Grupo de no máximo 2 pessoas.
- A prática deve ser feita no Micro C Pro for PIC
 - Programa em C
 - Compilar e Simular (debugger)
- Gravar o circuito no Kit EasyPIC v.7 e ligar o Kit Pic Genius
 - Habilitar as chaves necessárias (Display, Botões);
 - Carregar o programa no PIC;
 - Simular: fazer o display LCD mostrar o valor do PWM ligado na ventoinha;
 - Validar a prática: Usar o osciloscópio para medir o valor real da velocidade da ventoinha e comparar com o valor mostrado no LCD (calcular o erro entre as medidas).

Relatório Prática 4

- Colocar introdução, objetivos, resultados, conclusões, etc.
– *consultar as normas para confecção de relatórios no site da disciplina.*
- Colocar o esquemático do hardware utilizado (Proteus, Orcad ou similar).
- Colocar trechos do programa desenvolvido (comentado).
- Colocar os resultados da validação e tabela com os valores e erros.

Envio do Relatório

- Além do relatório em PDF, enviar também os arquivos do projeto gerados pelo compilador (código em C);
- Enviar os arquivos (*.zip) pelo site do e-Disciplinas até a data definida pelo professor (27/06/2023 – 23:59h).
- Mostrar o circuito funcionando para o professor na aula imediatamente posterior ao envio do relatório.