

Atividade Prática

Contador com Display Multiplexado

1 Itens abordados

Para completar esta atividade prática, você precisa conhecer os seguintes tópicos:

- Periféricos:
 - *Ports* de I/O.
 - Interrupções Externas.
 - Temporizadores em modo de operação normal e CTC.
- Dispositivos:
 - *Display* de sete segmentos multiplexado.

2 Critérios Avaliativos

Esta atividade deve ser postada em um arquivo compactado **em formato ZIP** no SIGAA impreterivelmente até às ____:____:____ do dia ____/____/____. Recomenda-se a postagem da atividade o quanto antes, de modo a evitar problemas associados à indisponibilidade de acesso à Internet.

O arquivo zip deve conter:

- a) O arquivo *workspace* e o diretório *.vscode* do Visual Studio Code.
- b) Todos os códigos-fonte (.c, .cpp, .h, .hpp, .s, .inc, .asm) necessários para a compilação com sucesso do código.
- c) O arquivo de simulação do Proteus, compatível com a versão 8.6 SP2 build 23525.

A nota do atividade prática é baseada no funcionamento e na qualidade da solução apresentada. São critérios indispensáveis:

- Funcionamento da solução;
 - Cumprimento dos critérios estabelecidos.
- Organização do código;
 - Documentação (comentários necessários no código-fonte);
 - Endentação coerente;
 - Formatação adequada.

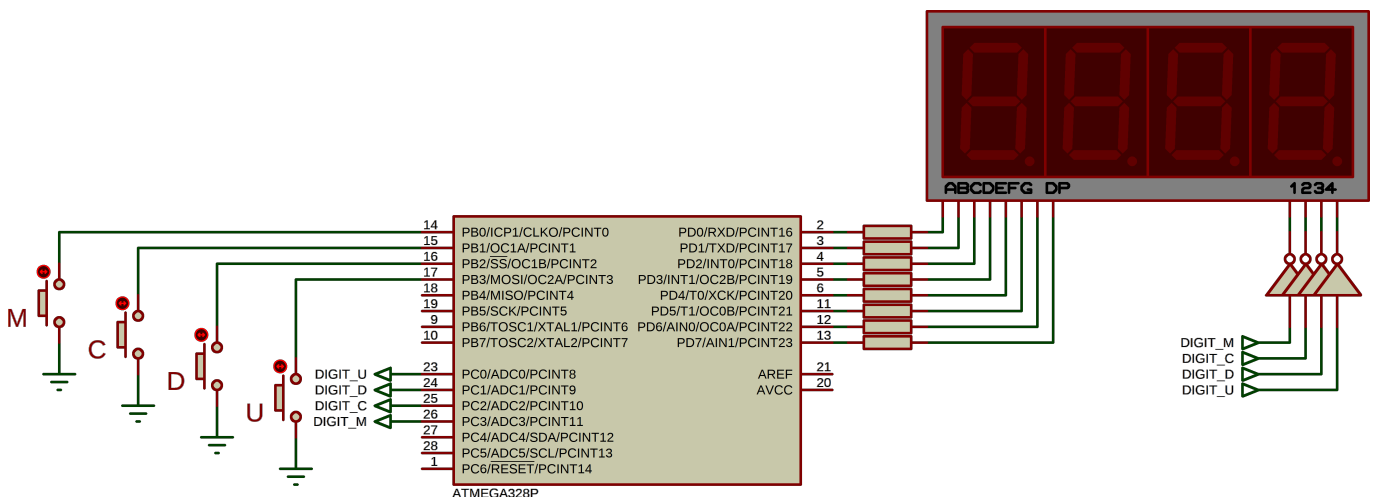
- Otimização;
 - Uso apropriado dos periféricos;
 - Uso racional de memórias RAM, EEPROM e Flash.

3 Objetivo

Desenvolver um contador crescente de quatro dígitos em um *display* de sete segmentos multiplexado, utilizando interrupções externas e um temporizador.

4 Hardware

A seguir, é apresentado o diagrama esquemático do arquivo de simulação e a lista de componentes e suas configurações:



- Microcontrolador;
 - Device: ATMEGA328P;
 - Library: AVR2;
 - Part Reference: *oculto*;
 - Part Value: *oculto*;
 - Encapsulamento: SPDIL28;
 - Fuse CLKDIV8: (1) Unprogrammed;
 - Fuse CLKSEL: (1111) External crystal 8.0–MHz;
 - Clock Frequency: 16 MHz;
- Botão;
 - Device: BUTTON;
 - Library: ACTIVE;
 - Part Reference: *deixar em branco*;
 - Part Value: *deixar em branco*;
 - Off resistance: 100M;
 - On resistance: 100m;
 - Switching time: 1m;
- Display;

- Device: 7SEG-MPX4-CA
- Library: DISPLAY;
- Part Reference: *em branco*;
- Part Value: *em branco*;
- Minimum Trigger Time: 1ms;
- Porta Inversora;
 - Device: POT-HG
 - Library: ACTIVE;
 - Part Reference: *oculto*;
 - Part Value: *oculto*;
 - Data to Q Delay (Low-High): (Default);
 - Data to Q Delay (High-Low): (Default);
 - Deglitch Time FOr Q Output: (Default);
- Resistores;
 - Device: RES;
 - Library: DEVICE;
 - Part Reference: *oculto*;
 - Resistance: 1k5 *oculto*;
 - Model Type: DIGITAL;
 - PCB Package: RES40;

5 Requisitos Obrigatórios

- 1) O código deve ser programado para o microcontrolador ATmega328P.
- 2) O código deve ser programado em C/C++.
- 3) O valor da contagem deve ser armazenado em uma variável **uint16_t value** e deve iniciar em 1802.
- 4) O valor de **value** deve ser apresentado no *display* multiplexado.
- 5) O sistema possui quatro botões de contato rápidos, denominados **M**, **C**, **D** e **U**.
- 6) A cada pulso no botão de contato rápido **U**, o valor de **value** é incrementado em uma unidade.
- 7) A cada pulso no botão de contato rápido **D**, o valor de **value** é incrementado em uma dezena.
- 8) A cada pulso no botão de contato rápido **C**, o valor de **value** é incrementado em uma centena.
- 9) A cada pulso no botão de contato rápido **M**, o valor de **value** é incrementado em um milhar.
- 10) Os botões devem ser tratados via interrupção, de modo a permitir que o valor da contagem continue a ser apresentado corretamente no *display*, mesmo que os botões estejam pressionados.
- 11) O usuário pode pressionar mais de um botão ao mesmo tempo, porém, a ação de cada botão deve ser executada apenas quando aquele botão é pressionado.
- 12) O valor de **value** vai de 0000 a 9999, se o usuário mandar incrementar o valor além deste limite, o valor deve estourar ciclicamente, por exemplo, $9945 + 100 = 0045$.