



UNIVERSIDADE FEDERAL DE MINAS GERAIS
ESCOLA DE ENGENHARIA

PROJETO DE SISTEMAS EMBUTIDOS

Lista de Exercícios 02

Aluno:

Giovanni Martins de Sá Júnior

Matrícula:

2017001850

15 de dezembro de 2023

1. Com relação a temporizadores/contadores, defina:

- **Watchdog Timer:** é um dispositivo temporizador que é usado para detectar falhas em sistemas embarcados. Ele precisa ser regularmente resetado pelo software do sistema. Se o software falhar ou ficar bloqueado, e o Watchdog Timer não for resetado dentro de um intervalo pré-definido, ele geralmente gera um reset no sistema para tentar recuperar de um estado potencialmente instável.
- **Input Capture:** é uma funcionalidade encontrada em alguns temporizadores. Ele permite que você capture o valor do contador no momento em que uma entrada externa específica ocorre, como uma borda de subida ou descida de um sinal. Isso é frequentemente usado para medir o tempo decorrido entre eventos externos.
- **Output Compare:** é outra funcionalidade associada a alguns temporizadores. Ele permite que se configure um valor de comparação e associe uma ação a ser realizada quando o valor do temporizador atingir esse ponto. Isso é frequentemente usado para gerar pulsos de largura de pulso variável, gerar sinais PWM (Modulação por Largura de Pulso) ou para controlar o tempo de execução de certas operações.
- **Real-time Clock:** é um relógio que fornece a hora atual em tempo real, geralmente incluindo informações como dia, mês, ano e às vezes informações adicionais, como temperatura. RTCs são frequentemente usados em sistemas embarcados ou dispositivos autônomos que precisam manter um registro preciso do tempo mesmo quando estão desligados.

2. Com relação a conversão de sinais analógicos, descreva a função dos seguintes termos:

- **Condicionamento de sinais:** refere-se ao processo de preparação e manipulação de sinais analógicos antes de serem convertidos para o domínio digital. Isso pode incluir amplificação, filtragem, isolamento e outras operações para garantir que o sinal de entrada atenda aos requisitos de precisão e faixa dinâmica do sistema de conversão analógico para digital (ADC).
- **Sample-and-hold:** é uma etapa crucial em sistemas de conversão analógico-para-digital. Ele envolve a amostragem (captura) do valor de um sinal analógico em intervalos regulares e a retenção desse valor constante durante o processo de conversão. Isso é necessário para garantir que o valor analógico seja convertido com precisão sem ser afetado por mudanças durante o processo de conversão.
- **Erro de Quantização:** refere-se à diferença entre o valor real de um sinal analógico e o valor quantizado correspondente após a conversão analógico-para-digital (ADC). Como os valores analógicos são representados por números discretos em um sistema digital, sempre haverá algum erro associado à representação discreta. O erro de quantização é uma

fonte de ruído e pode afetar a precisão do sinal digitalizado.

- **Aliasing:** ocorre quando um sinal analógico é amostrado a uma taxa inadequada, resultando em uma interpretação errada do sinal digitalizado. Se a taxa de amostragem não for suficientemente alta em relação à frequência do sinal de entrada, podem ocorrer artefatos indesejados. O Teorema de Nyquist-Shannon estabelece que a taxa de amostragem deve ser pelo menos o dobro da frequência máxima do sinal para evitar o aliasing.

3. Descreva os seguintes protocolos de comunicação:

- **SPI:** é um protocolo de comunicação síncrona serial usado para a transferência de dados entre um mestre (geralmente um microcontrolador ou processador) e vários dispositivos periféricos (escravos). Ele utiliza quatro linhas principais:
 - MISO (Master In Slave Out): Transmissão de dados do escravo para o mestre.
 - MOSI (Master Out Slave In): Transmissão de dados do mestre para o escravo.
 - SCLK (Serial Clock): Sinal de clock compartilhado entre o mestre e os escravos.
 - SS/CS (Slave Select/Chip Select): Linha usada para selecionar um dispositivo específico.
- **I2C:** é um protocolo de comunicação síncrona serial bidirecional usado para a comunicação entre vários dispositivos em uma placa. Ele usa duas linhas:
 - SDA (Serial Data): Transmissão de dados bidirecional.
 - SCL (Serial Clock): Sinal de clock compartilhado entre os dispositivos.
- **UART/USART:** são protocolos assíncronos usados para comunicação serial entre dispositivos. Eles usam duas linhas:
 - TX (Transmitter): Transmissão de dados do dispositivo de origem.
 - RX (Receiver): Recebimento de dados no dispositivo de destino.

4. Descreva os seguintes protocolos de comunicação:

- **volatile short int:** A palavra-chave volatile é usada para indicar ao compilador que a variável pode ser alterada a qualquer momento, sem qualquer ação realizada pelo código que está próximo. Isso evita que o compilador otimize operações relacionadas à variável, garantindo que leituras e gravações na variável sejam realizadas conforme especificado no código. O tipo short int indica um tipo de dado inteiro curto (com menos bits do que um inteiro padrão).

- **unsigned char:** é um tipo de dado em C/C++ que representa um caractere sem sinal (ou seja, apenas valores positivos ou zero). A faixa de valores para um unsigned char geralmente varia de 0 a 255.
- **#pragma vector _Vector_Timer-A:** #pragma é uma diretiva de pré-processador que fornece informações adicionais ao compilador. No contexto de microcontroladores e sistemas embarcados, pragma vector é frequentemente usado para definir vetores de interrupção. O trecho Vector_Timer-A sugere que esse pragma está associado a um vetor de interrupção específico relacionado ao Timer A.
- **#define UM 1:** #define é uma diretiva de pré-processador que substitui todas as ocorrências de um identificador por um valor ou expressão. Neste caso, UM está sendo definido como 1, então todas as instâncias de UM no código serão substituídas por 1.
- **const int UM = 1:** Isso declara uma constante inteira chamada UM com o valor de 1. Ao contrário da diretiva define, a declaração const cria uma variável de apenas leitura que ocupa espaço de armazenamento. O uso de const indica que o valor não deve ser alterado durante a execução do programa.

5. Defina as seguintes metodologias de projeto:

- **top-down:** é uma abordagem de design e desenvolvimento que começa com uma visão global do sistema e, em seguida, divide esse sistema em componentes menores e mais detalhados. Cada componente é projetado individualmente, e o processo é repetido até que os componentes se tornem pequenos o suficiente para serem facilmente implementados. Essa abordagem geralmente envolve a criação de um design abstrato antes de detalhar os componentes.
- **bottom-up:** é o oposto da top-down. Começa com a implementação dos componentes mais simples e, em seguida, esses componentes são agrupados para formar sistemas mais complexos. À medida que os componentes são combinados, o sistema geral é construído. Essa abordagem muitas vezes destaca a reutilização de código e a integração gradual de funcionalidades.
- **waterfall:** é um modelo linear e sequencial de desenvolvimento de software. Ele consiste em fases distintas, como requisitos, design, implementação, testes, instalação e manutenção, com cada fase dependendo da conclusão da fase anterior. Uma vez que uma fase é concluída, não há retorno a fases anteriores. Isso implica que as mudanças nos requisitos são difíceis de serem acomodadas após o início do processo.
- **ágil:** é uma abordagem iterativa e incremental para o desenvolvimento de software, enfocando a flexibilidade e a colaboração contínua. Ela se baseia em valores e princípios definidos no Manifesto Ágil. As equipes ágeis dividem o desenvolvimento em ciclos curtos chamados iterações,

onde entregam incrementos de software funcionais. A metodologia ágil é adaptativa, permitindo mudanças frequentes nos requisitos e respondendo rapidamente ao feedback do cliente.