

Prova de Microcontroladores para automação – Monitoria

1) Defina as diferenças básicas entre um microcontrolador e um microprocessador:

Um uP (microprocessador) é um chip integrado empregado para a realização de cálculos e tomadas de decisão do sistema em questão, podendo abranger sistemas genéricos e não especializados, sendo capaz de processar grandes quantidades de dados ao mesmo tempo que tem a capacidade de gerenciá-los. Um uP geralmente precisa de periféricos para funcionar, como por exemplo, memórias de dados e de programas (RAM e ROM), componentes de gerenciamento de I/Os, osciladores de sincronismo, entre outras, possuindo um barramento para comunicá-los.

Ao contrário, um uC (microcontrolador) já um chip integrado usado para sistemas que tenham funções especializadas, específicas, voltado para o controle de I/Os com rotinas de comandos específicas para esse controle. Normalmente um uC possui internamente, todos dispositivos que precisa para funcionar, estando tudo encapsulado no mesmo chip, como memórias de programa e dados, osciladores, registradores, unidades de processamento, gerenciadores de I/Os e conversores e outros.

No entanto, a destinação de um uC e um uP é diferente, por exemplo, um uP pode rodar um SO sobre ele, enquanto um uC não, tornando suas funções diferentes. Um uC geralmente é empregado para se fazer o controle de algum sistema bastante específico, como por exemplo, gerenciar o estado de alguma lâmpada de sinalização, enquanto o uP faz toda a administração dos dados do sistema, cálculos de multimídia e afins. Não é estranho se ver um uC e um uP trabalhando juntos, um faz a aquisição ou atuação de dados I/O enquanto o outro faz o gerenciamento e processamento desses dados.

Um exemplo de uP é um Raspberry pi 4, que é capaz de rodar um sistema operacional e muitas funções multimídia, além de programas de camadas de alto nível.

Um exemplo de uC é um Arduino Uno, que é facilmente programável e muito barato, usado para projetos que necessitam fazer o monitoramento de I/Os como por exemplo o controle de lâmpadas de uma casa.

2) Explique individualmente as principais partes de um microcontrolador

- a) CPU: Unidade de processamento Central, a responsável por todo processamento dentro do uC, ela que define as rotinas de processo e acessos às memórias de todo o processo. Todos os dados processados no uC passam pela CPU, por isso é associada ao cérebro do processo.
- b) ULA: Unidade Lógica Aritmética, são unidades destinadas exclusivamente para o cálculo dos dados enviados a ela pela CPU. Todo e qualquer tipo de cálculo aritmético que os dados são submetidos via código de programação, são realizados pela ULA. Por exemplo, a conversão de um sinal analógico 10 bits em um sinal 8 bits através da função map().

- c) Memórias: As memórias são locais onde os diferentes tipos de dados são armazenados, podendo ser voláteis ou não voláteis.

Memórias voláteis (RAM): São memórias de acesso rápido, porém que não possuem a capacidade de retenção dos dados caso o sistema deixe de ser alimentado(desligue), por isso são usadas somente para as rotinas de processamento, como por exemplo, armazenar dados para cálculos e/ou dados auxiliares do programa.

Memórias não voláteis (ROM): Geralmente EEPROM ou Flash, são as responsáveis por armazenar rotinas de programação ou dados importantes do sistema. Todo código escrito, fica salvo nesse tipo de memória, pois nelas, mesmo que o uC desligue, as rotinas ainda estarão escritas ali para serem lidas quando o sistema voltar.

- d) Barramento: Um barramento dentro de um uC é um caminho pelo qual os dados serão transmitidos internamente e são conexões físicas. Como citado antes, se o CPU decide que um determinado dado seja manipulado ou armazenado na memória, ele irá enviar esse dado até seu destino através de um barramento, que fará essa comunicação.

Em geral existem 3 tipos de barramentos, os de dados, endereço e de operação. Para um dado ser transmitido, ele precisa de um rótulo de destino (endereço), qual a sua finalidade (Operação) e o seu valor em si (dado).

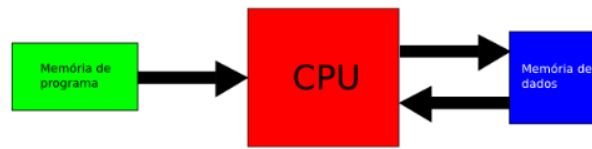
- e) Registradores: Os registradores são memórias especiais do uC, que armazenam informações importantes para o funcionamento dele. São memórias não voláteis e geralmente possuem um nome associado a suas funções. Um exemplo de registrador pode ser o registrador de entradas e saídas de uma GPIO. Nesse registrador, está definido se uma GPIO vai ser usada como entrada ou saída, no caso do Arduino esse registrador pode ser chamado pelo comando DDRx, um registrador de 8 bits onde cada bit representa uma GPIO específica (0=INPUT, 1=OUTPUT).

- 3) Quais as diferenças entre as arquiteturas Harvard e von-Neuman, inclusive descrevendo-as através do diagrama simplificado apresentado em aula.

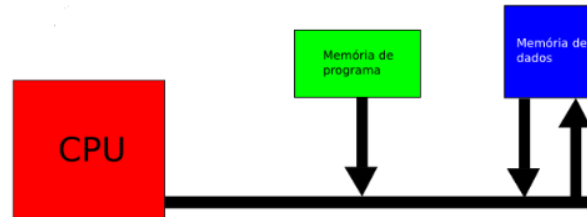
Von-Neuman: Não há distinção entre o espaço de instrução e o espaço de dados, ambas trafegam pelo mesmo barramento e pertencem ao mesmo espaço de endereçamento.

Harvard: Possui distinção entre o espaço de programa e de dados, possuindo dois barramentos distintos. Essa conta com a vantagem de permitir a execução de alguma instrução ao mesmo tempo em que busca um dado na memória.

Arquitetura Harvard



Arquitetura Van Neuman



4) Diferencie memória dinâmica de memória estática:

Ambas são memórias voláteis, no entanto, as memórias dinâmicas necessitam frequentemente serem renovadas, como se possuíssem uma bateria que descarregasse e o seu valor se perdesse se não fosse reescrito(carregado), enquanto a estática, possui seus dados salvos sem a necessidade de renovação armazenando seus estados enquanto estiverem recebendo alimentação.

5) Explique o que são interrupções e cite 3 das principais interrupções em um microcontrolador.

Uma interrupção, é qualquer sinal que interrompa o fluxo de instruções que o uC estivesse executando e desviasse esse fluxo para alguma sub-rotina que após realizada, retorna ao fluxo normal do uC no exato ponto que parou. Os três exemplos de interrupções padrões do uC que são:

- 1) Interrupção devido ao overflow do Contador.
- 2) Interrupção por variação de tensão nos pinos.
- 3) Interrupção por fim de escrita.

Algumas interrupções já são padrões do uC, podendo ser chamadas com uma função ISR(Nome_da_interrupção), enquanto outras podem ser criadas através das GPIOs com o `pinAttachInterrupt`.