

Aluno: Yves Augusto Lima Romero.

Por fazer:

Introdução: 5.
Unidades de Processamento: 4, 13.
CPU: 6, 14.
Unidades de entrada e saída : 5, 6, 13.
Comunicação: 1, 8.
Sistemas Operacionais: 6.

➤Introdução

1. O que diferencia um sistema computacional embarcado de um sistema computacional ‘convencional’?

➤ Num sistema embarcado, o computador é utilizado para efetuar um propósito específico. Diferentemente, nos computadores de propósito geral, o computador é o próprio produto; ao passo que, nos SEMBs, o computador é apenas um instrumento para realizar a finalidade da aplicação, por exemplo, mostrar as horas, fazer a predição de temperatura, etc. Outra diferença, é que o computador dos SEMBs, por realizarem uma tarefa específica, dificilmente vai poder ser usado pelo usuário para um propósito diverso do propósito da aplicação para o qual o SEMB foi projetado.

2. Defina *time-to-market*. Comente a sua importância em sistemas embarcados.

➤ Time to market é o tempo que leva entre a idealização de um sistema embarcado e a sua disponibilização no mercado. É muito importante pois, muitas vezes, o ingresso tardio de um determinado produto no mercado pode comprometer os lucros. Em diversas ocasiões, houve produtos que poderiam ter gerado mais lucro caso tivessem sido lançados no mercado em tempo mais hábil. No entanto, não adianta antecipar a venda do seu produto se isto vier a comprometer a qualidade. É necessário que haja uma ponderação desses fatores.

3. Defina reuso e a sua importância no projeto de sistemas embarcados.

➤ O reuso é o processo pelo qual se pode criar novos softwares ou atualizar softwares já existentes, partindo de componentes e recursos previamente criados. Nos sistemas embarcados, as plataformas fornecem diversas facilidades para o reuso de softwares, devido à semelhança entre as placas suportadas. É um fator de grande relevância num projeto, pois previne os participantes de criar coisas que já estão prontas; além disso, se no decorrer do projeto for

necessário realizar mudanças no hardware, vai ser preciso realizar menos alterações no software a fim de adaptá-lo às circunstâncias do novo hardware.

4. Defina SoC (*System on Chip*). Comente sobre as vantagens de usá-lo no projeto de um sistema embarcado.

➤ Um SoC é um sistema embarcado que contém num só chip a memória, os dispositivos de entrada e saída, e o processador. Diferente de outros computadores, onde esses componentes se encontram espalhados ao longo do dispositivo, nos SoCs, todos ficam posicionados dentro de um mesmo chip. Uma das principais vantagens desse modelo é a sua maior eficiência energética e o custo mais baixo. Em virtude do seu elevado grau de integração e do cabeamento mais curto, ele consome menos energia. Além disso, há o fator da minimização de espaço, e o benefício de amenizar a latência oriunda dos diversos componentes situados na placa mãe, o que atenua os delays resultantes da interferência e agiliza os processos de transmissão de dados.

5. Descreva brevemente a diferença entre Requisitos e Especificação.

➤

6. Sugira uma estratégia para medir o tempo de computação de um programa rodando em um microprocessador.

➤ Basta utilizar duas vezes uma função de medição de tempo dentro do programa, guardando os valores em variáveis distintas, e ao cabo do processo subtrair as duas e imprimir o resultado na tela. Na biblioteca Arduino, por exemplo, existem as funções `millis()` e `micros()`.

7. Sugira uma estratégia para medir a energia gasta por um programa rodando em um microprocessador.

➤ Podemos olhar no datasheet o V_{cc} típico e o I_{cc} , e calcular a potência utilizando a fórmula $V_{cc} * I_{cc}$. Sabendo o tempo de execução do programa, basta multiplicar esta potência por este tempo, e o resultado vai ser dado em joules.

8. Considere os seguintes sistemas embarcados: Um *tablet*, uma impressora e um sistema de navegação automotivo. Faça uma tabela com uma coluna para cada dispositivo e uma linha para cada uma das seguintes métricas de projeto: custo unitário, desempenho, tamanho e potência. Para cada campo da tabela, indique se existe uma forte restrição para aquela métrica.

Métricas	Tablet	Impressora	Sistema de Navegação
Custo Unitário	Restrição média	Restrição baixa	Restrição alta
Desempenho	Restrição média	Restrição média	Restrição alta

Tamanho	Restrição média	Restrição alta	Restrição alta
Potência	Restrição alta	Restrição média	Restrição alta

Unidades de processamento

1. Cite as estratégias utilizadas para implementar o processamento em computadores e destaque as diferenças entre elas.

➤ Hardware dedicado

➤ Hardware genérico

➤ No primeiro, a própria forma do circuito é desenhada para realizar uma tarefa específica, pois já se sabe de antemão que algoritmo ou que tipo de algoritmo será executado. No segundo, não é o caso, o hardware é projetado para rodar o maior número possível de algoritmos, sem se dedicar a algum em particular.

2. Compare um microprocessador, que usa uma estrutura previamente definida e genérica com um circuito dedicado. Apresente pontos positivos e negativos de cada uma das duas estratégias.

➤ O genérico tem mais flexibilidade e facilita as famílias de produtos. O dedicado possui maior performance nas aplicações para as quais foi desenhado.

3. Defina plataforma.

➤ Uma família de arquiteturas que satisfazem simultaneamente a um conjunto de restrições para reuso de hardware e software, segundo a definição de Sangiovanni. Possuem interfaces (APIs) para facilitar a ligação entre a plataforma e os softwares que vão ser desenvolvidos nas placas da arquitetura.

4. Escolha duas plataformas microprocessadas apresentadas em classe e destaque as diferenças e semelhanças entre elas. Indique uma aplicação que seria bem sucedida em uma das plataformas e não seria na outra.

5. O que é um ‘Cross compiler’? Qual a sua importância no desenvolvimento de sistemas embarcados?

➤ É um compilador que, como os demais compiladores, gera executáveis. Mas os executáveis de um compilador cross podem ser executados em plataformas distintas daquela em que o compilador está rodando. É útil porque permite uma separação entre o ambiente de desenvolvimento do ambiente alvo; por exemplo, quando o computador embarcado possui recursos muito limitados, e não seria capaz de rodar um compilador.

7. De que maneira os circuitos reconfiguráveis podem ajudar a diminuir o tempo de projeto de um sistema computacional?

➤ Redução do time to market, devido à prototipação rápida. Basta comprar na loja, e fazer a programação via HML, e já se vai ter um hardware dedicado para determinada aplicação.

- Fácil de corrigir bugs, pois basta reconfigurar o hardware. Em outros casos, dependendo do estágio do projeto onde o bug é descoberto, isso pode acarretar um grande atraso.
- Quando a aplicação não é feita em larga escala, não vale a pena mandar fabricar um chip a fim de desenvolvê-la. Nesse sentido o FPGA ajuda bastante, pois basta configurá-lo para atender às necessidades do projeto.

8. Como um SoC pode ser implementado em um FPGA?

- Os FPGA possuem blocos lógicos que, ao invés de utilizar portas lógicas para produzir as saídas, emulam as saídas dessas portas por meio de memória SRAM, tornando possível copiar o comportamento de qualquer circuito lógico.
- O FPGA é utilizado para emular o SoC. É necessário uma biblioteca para representar o processador e os seus componentes. Basta abrir o software de montagem de processador, onde vai haver o mesmo processador em configurações diferentes, como formas de memórias, dispositivos E/S, etc. Essas partes são combinadas, e a ferramenta gera uma descrição do SoC. Esses são os processadores Soft-Core.
- Há também os híbridos, que possuem uma parte do processador com lógica não configurável, e outra parte idêntica à dos processadores Soft-Core. Mas apresentam potência mais baixa na parte FPGA.

9. Explique por que um processador de propósito geral pode custar mais barato que um processador projetado por você mesmo. Por que, então, você faria seu próprio projeto?

- Porque o processador de propósito geral foi produzido em larga escala, diluindo os custos de projeto entre os consumidores, o que permite a fixação de um preço mais baixo. Já num processador projetado por mim mesmo, eu teria de arcar com os custos do projeto sem ter um retorno compensatório, dependendo exclusivamente do sucesso da minha aplicação no mercado.
- Uma motivação para se fazer seu próprio projeto é, por exemplo, definir a arquitetura e os recursos do processador de forma mais adequada para a sua aplicação, podendo-se pensar num hardware dedicado.

10. Defina processador *soft-core*. Apresente e explique 02 (duas) características (positivas ou negativas) distintas e relevantes dessa estratégia de implementação de sistemas embarcados, comparada com outras de sua escolha.

- São processadores que disponibilizam diversas combinações de memória, dispositivos de E/S, componentes de um processador em geral.

11. Faça uma lista das funcionalidades de um processador DSP.

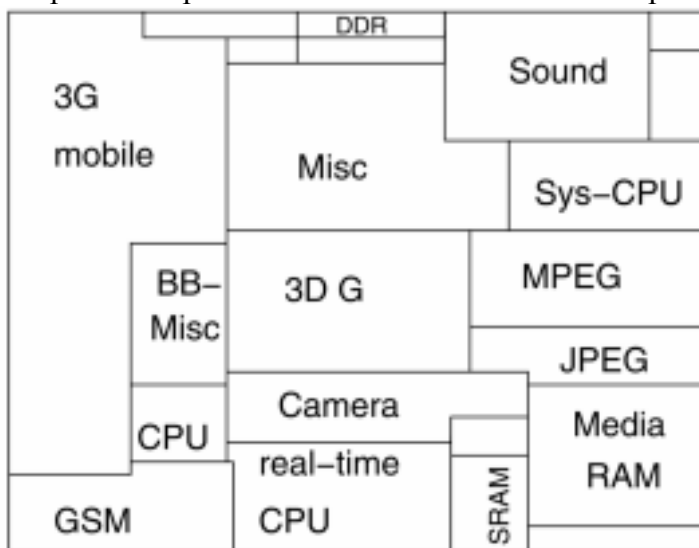
- É um SIMD(Single Instruction Multiple Data), ou seja, cada instrução sua é aplicada sobre diversos conjuntos de dados.
- Tem um acumulador dedicado para multiplicar e acumular os resultados.(isso é o que ajuda a multiplicar matrizes)
- O laços são controlados por um software, de maneira a garantir a correta aplicação das instruções sobre o conjunto de dados.
- Utiliza operadores de ponto fixo ao invés de ponto flutuante.

➤ Possui DMA, que permite acessar os dados de forma independente da unidade de processamento.

12. Defina ASIC e ASSP. Considerando o impacto no custo financeiro do produto final, como estas tecnologias se distinguem uma da outra?

➤ O ASIC é uma hardware especificamente criado para uma companhia e seus produtos. O ASSP é o que se costuma chamar produto de prateleira, é mais universal e pode ser utilizado por diversas companhias e usuários. Ambos são hardwares dedicados.

13. Os processadores multi-core têm sido largamente utilizados, tanto em computadores de propósito geral quanto em computadores embarcados. A figura abaixo mostra um exemplo de arquitetura Multiprocessor systems-on-a-chip (MPSoC), comumente usada em aplicações embarcadas. Apresente as diferenças do modelo MPSoC para os multi-core usados em computadores pessoais e as razões do seu uso em aplicações embarcadas.



CPU - Energia e Desempenho

1. Em um sistema computacional, diferencie Potência de Energia. Como estas grandezas afetam a vida das baterias do sistema?

➤ A potência está associada à corrente e à tensão que são entregues ao sistema, já a energia mede a potência ao longo do tempo. Mas, diferentemente da potência dissipada por um resistor de resistência fixa, a potência requerida por computadores varia bastante.

2. Em um circuito CMOS, explique como as comutações afetam a potência dissipada.

➤ Durante a troca de estado, a corrente e a tensão no transistor se alteram linearmente, consumindo potência nesse meio tempo, sobretudo quando as grandezas atingem um ponto da reta onde o produto $V \times I$ é máximo. Neste ponto, a potência exigida é a mais elevada possível.

3. Apresente e justifique duas estratégias para reduzir a potência dissipada em um circuito fabricado com tecnologia CMOS.

- Reduzir a tensão de alimentação, pois ela influi na potência de maneira quadrática.
- Reduzir a frequência de operação da CPU, apesar disso não amenizar o consumo de energia.
- Impedir que o clock alcance circuitos que estão inativos, e, claro, desligar partes inoperantes da CPU.

4. Explique DVS (*Dynamic Voltage Scaling*) e como esta técnica pode ser usada para aumentar a vida das baterias que alimentam um sistema computacional.

- O DVS gerencia a frequência de clock do sistema conforme a inoperância ou atividade de determinados recursos. Geralmente se tenta operar na menor tensão possível que uma frequência permite. Existe um tempo necessário para se transitar entre os patamares de frequência na escala.

5. Como a profundidade do pipeline afeta a potência de um processador?

- Quanto maior a profundidade, mais processamento será requerido pelo hardware responsável por orquestrar os estágios de pipeline, o que eleva a potência consumida.

6. Como as instruções de desvio encontradas em um programa podem afetar a potência dissipada em um processador pipeline? O que se pode fazer para minimizar esta potência? Apresente várias alternativas e analise os pontos fortes e fracos de cada uma delas.

- O desvio das instruções cria a necessidade de um hardware extra para coordenar esses desvios e garantir a consistência dos dados manipulados pelo programa. (CONTINUA AMANHÃ)

7. Do ponto de vista da potência dissipada, em que um processador EPIC (VLIW) é diferente de um superescalar convencional?

- O VLIW não possui hardware para orquestrar o paralelismo das instruções, pois a garantia é obtida em tempo de compilação.

8. Qual a problema da superescalaridade para aplicações de tempo-real?

- Como o paralelismo precisa ser garantido por um hardware extra, nunca se sabe o quanto rápido vão ser executados os programas. O que é crucial quando se trata de aplicações de tempo real, pois é necessário que haja garantia de que certas tarefas sejam efetuadas em tempo hábil.

9. Por que processadores EPIC/VLIW têm maior aceitação entre aplicações embarcadas do que em computadores desktop?

- Porque o VLIW, por garantir o paralelismo a nível de instrução, presume que o compilador conheça a estrutura interna do processador, o que seria inviável em computadores desktop.

10. Qual o problema do uso de sistemas de memória cache para aplicações de tempo-real?

- A memória cache introduz um certo indeterminismo no sistema, pois nunca se sabe

quando vai ocorrer um cache miss, o que afeta a previsibilidade da aplicação. No entanto, a previsibilidade é essencial para aplicações de tempo real.

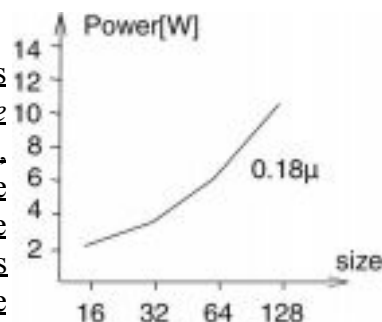
11. Explique a diferença entre um FPGA (*Field-Programmable Gate Array*) e um CPLD (*Complex Programmable Logic Device*).

➤ Os FPGA são voláteis, pois suas configurações são salvas em memória SRAM, o que significa que são perdidas tão logo o processador é desligado da tomada. Já no CPLD, as configurações são conservadas mesmo após o reinício do sistema.

➤ Os CPLDs precisam ser desligados e reconfigurados para alterarem a forma do processador, ao passo que os FPGAs podem sofrer alterações mesmo enquanto estão rodando.

12. O que são instruções tipo SIMD? Como isso pode ajudar a atender restrições de sistemas embarcados?

➤ São instruções que se aplicam sobre diversos conjuntos de dados, como já diz o nome *Single Instruction Multiple Data*. Quando o processador utiliza esse tipo de instrução, as operações são realizadas de forma mais eficiente, o que desafoga os demais componentes do sistema embarcado, e permite que o projetista canalize seus recursos em outras melhorias. O ganho de desempenho na utilização de processadores como o DSP, que se vale de instruções desse tipo, é enorme, pois as operações matemáticas mais complexas que ele executa exigem um esforço computacional muito grande, e seria prejudicial ao sistema se esse esforço fosse depositado nas mãos de um processador comum.



13. A figura a seguir mostra a relação entre a capacidade da memória cache e a potência dissipada nesta memória, quando usada a tecnologia de fabricação 0.18μm. Baseado no gráfico, analise pontos fortes e fracos, bem como restrições do uso de cache em sistemas embarcados.

➤ As memórias cache gastam mais energia e se tornam mais lentas à medida que são maiores. Porém, é possível achar um valor ótimo para a relação entre o tempo de resposta e o tamanho da memória cache, assim como é o caso da energia. Mas isso se torna mais fácil quando já se sabe de antemão que espécie de algoritmo vai rodar no nosso sistema. E nesse ponto já temos uma restrição para os sistemas embarcados, pois acontece diversas vezes de haver variedade de algoritmos, de forma a dificultar a minimização do consumo de energia, por ser necessário encontrar os valores que, na média, resultam num menor gasto.

➤ Outro ponto interessante é que os sistemas embarcados que implementam aplicações de tempo real são prejudicados pelo uso de memória cache, pois vão estar sujeitos ao indeterminismo do *cache miss*, o que torna a aplicação menos previsível e afeta os requisitos de um sistema de tempo real.

14. Considere um sistema de medição da precipitação (pluviômetro) localizado remotamente e capaz de enviar as suas leituras por comunicação sem fio de longa distância. O sistema faz leituras em intervalos de 20 minutos e manda a leitura para um servidor remoto. A alimentação é feita por baterias de 3V. O processo de medição/transmissão dura 10 segundos e durante esse tempo o consumo é de 50mA. Depois disso o sistema entra em repouso e seu consumo é desprezível. Qual a potência que as baterias devem ser capazes de fornecer? Se for usada uma bateria de 2000mAh, em quanto tempo ele precisará ser substituída?

- A potência que deve ser oferecida é $50\text{mA} \times 3\text{V} = 150 \text{ mW}$.
- Uma bateria de 2000mAh consegue manter uma corrente de 50mA durante 40 horas. Como a cada 20 minutos, é injetada uma corrente de 50mA, então em uma hora são consumidos 150mAh: daí, é necessário 13,33 horas para consumir 2000mAh.

Unidades de Entrada e Saída

1. Compare o display de 7 segmentos (LED) com o LCD e o OLED em relação à potência, vida útil e preço. Explique

- O LED é uma solução de muito baixo custo, que gasta pouca energia, é bom para aplicações onde não há necessidade de complexidade nas informações. Mas o LCD gasta menos energia, porque só utiliza campo elétrico, não havendo consumo de corrente para fazer o controle de luz polarizada; o único gasto é a lâmpada na camada de número seis. O OLED consome menos energia porque nele não há necessidade de manter uma lâmpada acesa. Quando a tela está preta, todos os leds são apagados.
- OLED é mais caro que LCD, que é mais caro que LED.

2. Por que o LCD apresenta menor brilho que o OLED?

- Porque não possui luz própria. Toda entonação de cor produzida pelo LCD vai ser resultado de uma combinação entre a intensidade de luz que é barrada ou permitida através da polarização.

3. Explique o funcionamento de um conversor D/A. Análise aspectos como linearidade e resolução.

- São fixadas n entradas, que correspondem ao número total de bits de resolução, ordenadas conforme a significância dos bits. O bit mais significativo fica atrelado à saída do resistor de menor resistência, e à medida que se desce na escala da significância, o valor das resistências é duplicado. Assim, é possível situar o valor recebido pelo conversor numa escala entre 0V e o Vcc, que geralmente é 5V.
- A linearidade do circuito real é bem menor que a do teórico, porque os resistores nem sempre vão atingir a exata relação de proporcionalidade que é requerida para efetuar a conversão.

4. Dê três exemplos de dispositivos atuadores. Cite duas aplicações para cada um deles.

- Cilindros Hidráulicos: pode ser usado em guindastes de caminhão, e empilhadeiras.
- Relés: Usado em automação predial, e controle de circuitos elétricos domiciliares.
- Motor elétrico: cortador de grama e elevadores.

5. Dê três exemplos de sensores. Cite duas aplicações para cada um deles.
6. Proponha três sensores diferentes para detectar chuva em um automóvel. Como um processador poderia ler estes sensores?
7. Compare os conversores A/D tipo aproximação sucessiva e conversão direta em relação a resolução, linearidade e velocidade. Explique.

➤ Velocidade: O Flash é o mais rápido devido ao uso de comparadores, que estabelecem patamares de tensão; a entrega do resultado é quase imediata. Em segundo lugar vem o de aproximação, e depois o integrador. O integrador é mais por usar busca sequencial, ao passo que o outro utiliza um algoritmo de busca binária.

➤ Resolução: O Flash é o mais problemático nesse quesito, pois cada bit de resolução que se incrementa a ele resulta num aumento exponencial do número de comparadores. O integrador precisa duplicar o tamanho da lista de pesquisa ao se acrescentar um bit, e o de aproximação sucessiva só necessita aumentar em 1 o número de etapas da busca binária.

➤ Linearidade:

8. Sugira duas aplicações para cada um dos seguintes tipos de conversor A/D: Integrador, Flash.

➤ O integrador pode ser usado para se medir umidade e temperatura, já que são grandezas que não costumam variar demais em aplicações comuns. Por isso, não havendo pressa para entregar as conversões, pode-se usar o integrador. O Flash deve ser utilizado em aplicações de baixa resolução, devido ao custo envolvido quando se aumenta o número de bits, visto que a cada bit é necessário acrescentar um comparador; ou então aplicações onde a alta frequência de amostragem é uma exigência, como detecção de radares e rádios receptores de banda larga.

9. Dada uma aplicação que requer valores elevados de resolução e linearidade, qual conversor A/D você recomendaria? Por que?

➤ O de aproximação sucessiva. Pois o acréscimo de um bit de resolução só implica em uma etapa a mais no seu algoritmo de busca binária, o que permite um aumento de resolução sem muitos custos adicionais. É fácil conseguir linearidade com ele porque o acréscimo de bits não implica em gastos muito grandes, o que permite maior exatidão na conversão, e consequentemente, os degraus não vão apresentar tanta discrepância em relação uns aos outros.

10. Qual a importância da taxa de amostragem para a fidelidade de um sinal digitalizado?

➤ Se a taxa de amostragem for muito baixa, é provável que diversos pontos do sinal sejam perdidos durante o tempo ocioso do conversor. Isso vai provocar um erro de interpretação do sinal recebido, e o sinal entregue pelo conversor será bem discrepante do original.

11. Explique o erro de quantização. Como ele pode ser reduzido?

➤ O erro de quantização se deve sobretudo à finitude da resolução de um ADC. Quanto menor for sua resolução, mais os valores entregues pela conversão serão

discrepantes dos valores entregues pelos sinal, o que vai provocar mudanças no gráfico. É possível amenizar este erro aumentando a resolução do conversor, ou seja, acrescentando mais bits.

12. Um motor opera a 100 rpm quando submetido a uma tensão de 3,7V. Considere a disponibilidade de um driver PWM que fornece 5V quando em nível alto e 0V quando em nível baixo. Calcule o valor de ciclo de trabalho (razão cíclica ou *Duty cycle*) para que o motor opere a 100 rpm.

➤ $Duty\ cycle = 100\ rpm / 100\ rpm = 1 = 100\%$

13. Por que não podemos usar um conversor D/A para acionar diretamente cargas de alta potência? Sugira uma estratégia para que um sistema digital possa acionar uma carga que opera de forma analógica, como motores CC.

➤

14. Considere um ADC com resolução de 12 bits, tempo de conversão de 12,5 microssegundos e que utiliza o método de aproximação sucessiva. Qual a frequência máxima teórica do sinal que ele poderia converter para digital? Justifique.

➤ O tempo de conversão de um ADC de aproximação sucessiva é calculado pela fórmula $n^\circ\ de\ bits / frequência$. A frequência, portanto, é $12 / 12,5 \times 10^{-6} = 0.96\ MHz$. A frequência do sinal teria de ser no máximo a metade deste valor, devido ao Critério de Nyquist, ou seja, 0.48 MHz.

Comunicação

1. Escolha três padrões de rede usados em aplicações embarcadas e compare-os segundo os requisitos: *Bandwidth*, Robustez, Facilidade de reparos, Privacidade e Tempo-real.
2. Explique a diferença entre comunicação orientada a conexão e *publish-subscribe*. Dê um exemplo de aplicação embarcada para cada uma.

➤ Numa comunicação orientada existe conexão lógica entre a origem e o destino, havendo necessidade de a aplicação destino conhecer o endereço da aplicação que enviou a mensagem. No modelo de publish-subscribe, as mensagens são transmitidas por intermédio de um Broker, que coordena a chegada e o envio dessas mensagens, sem notificar à aplicação de destino de onde elas vieram. Outra característica deste modelo, é que, em caso de uma das aplicações se desligar da comunicação, as outras não precisam ser avisadas, pois tudo é mediado pelo Broker.

3. Descreva o barramento I2C dando os detalhes para os níveis físico, enlace, rede e transporte do modelo OSI.

➤ O foco do sistemas embarcados é nos níveis mais baixos do modelo OSI. O I2C oferece a veiculação do endereço de origem na própria mensagem, e os dois lados partilham da mesma frequência de clock; isto diz respeito à camada física. Já com relação aos dados(data link), eles são transmitidos por meio do pino SDL, utilizando bits de paridade para comprovar a autenticidade das informações recebidas.

4. Considere como aplicação um sistema de elevadores, como mostrado na figura abaixo. Cada andar possui painel de controle de displays. Cada elevador possui um botão por andar e botão de

emergência. Sensores indicam a posição do elevador por todo o prédio, com aumento da resolução na vizinhança dos pontos de parada. Existe um controlador central (monitorea posição dos elevadores e chamados nos andares, despacha elevadores para atender chamados) e um controlador em cada elevador (controle de movimento).

- a) Proponha uma arquitetura de rede.
- b) Selecione uma tecnologia de rede.
- c) Que propriedades seriam necessárias nos níveis: físico, enlace, redes e transporte desta rede?

5. Como o protocolo CSMA-AMP pode contribuir para aumentar o determinismo de uma rede? Comente o efeito sobre os nós que possuem baixa prioridade.

➤ Atribuindo prioridade aos dados de valor mais alto, é garantido que as informações importantes vão chegar o mais rápido possível, e isso eleva o determinismo do sistema.

➤ Os nós de baixa prioridade podem sofrer *starvation*, que consiste na negação, durante longos períodos, para o envio daqueles dados, como se a mensagem estivesse definhando, morrendo de fome.

6. A figura abaixo mostra um cabo USB. Por que os fios de dados são enrolados e os de alimentação não?



➤ Porque os dados são suscetíveis de sofrer a ação de ruídos externos. Por esse motivo, os cabos são enroscados de modo firme, a fim de que o ruído atinge um dos fios seja idêntico que atinge o outro fio, e ao fim da travessia os sinais sejam somados e os ruídos cancelados.

7. Como o destino da mensagem é indicado no padrão I2C? Apresente uma vantagem e uma desvantagem dessa estratégia.

➤ No I2C, o destino da mensagem é enviado na própria mensagem, diferente do SPI, que possui Slave Select. A vantagem do I2C é congrega as vantagens contidas na UART(paridade) e no SPI(bus).

8. Para que servem os drivers RS232 usados com a interface UART de um microcontrolador?

➤ São um protocolo para as comunicações entre o microcontrolador e seus dispositivos periféricos. É usado para definir os parâmetros da comunicação serial, de maneira a auxiliar que as informações enviadas sejam recebidas e corretamente compreendidas pelas máquinas.

9. Explique como a técnica de salto de frequência do padrão Bluetooth contribui para aumentar a sua robustez.

➤ Devido ao salto de frequência, o Bluetooth é capaz de conviver com outros dispositivos que utilizam radiofrequência no seu raio de alcance, por meio de uma troca automática de canais de frequência. Isso torna o padrão mais resistente às interferências do meio, ou seja, mais capaz de sofrer alterações sem ter o funcionamento prejudicado, o que está associado à robustez.

Sistemas operacionais

1. O que é *device driver*? Qual a sua importância no projeto de um sistema embarcado?

➤ É uma camada de abstração visando a independência entre a equipe de hardware e a equipe de desenvolvimento de software. Possui um conjunto de primitivas para acessar o hardware.

2. O que motivaria uma equipe de desenvolvimento a abandonar a estratégia de “laço simples” e partir para um “laço combinado com serviço de interrupção”?

➤ Garantir que certas tarefas possam ser realizadas em tempo hábil, utilizando a interrupção para congelar o laço durante alguns instantes, enquanto o processador se ocupa de um processamento mais urgente.

3. O que motivaria uma equipe de desenvolvimento a abandonar a estratégia de “laço combinado com serviço de interrupção” e partir para o uso de um sistema operacional?

➤ Quando houver dificuldades com relação ao número de segmentos de programa, e consequentemente problemas acerca da divisão desses segmentos entre uma equipe de programadores. Para que a divisão seja funcional, é preciso que os códigos estejam muito intimamente integrados, a fim de não haver conflito entre os códigos, o que impõe diversas dificuldades ao seguimento do projeto. O S.O auxilia no gerenciamento da equipe, e dos segmentos.

4. Qual das três estratégias seria melhor para o caso do elevador acima? Mudaria alguma coisa de fosse um sistema de 8 elevadores?

➤ ????

5. Que propriedades devem ser buscadas em um sistema operacional para uma aplicação embarcada que requer baixa dissipação de potência? Justifique.

➤ Flexibilidade: numa aplicação embarcada dificilmente se vai saber de antemão o que será necessário para cumprir o projeto, por isso é preciso que S.O permita a desabilitação de certas funções, para que o projetista as utilize conforme for conveniente.

➤ Deve haver liberdade para acessar os dispositivos I/O diretamente, bem como se deve evitar mecanismos de proteção desnecessários a uma aplicação embarcada.

6. Escolha três sistemas operacionais usados em aplicações embarcadas e compare-os segundo os requisitos: custo de licenciamento, diversidade de plataformas, suporte, ferramentas de desenvolvimento e complexidade da aplicação.



7. Qual o papel do *middleware* em uma aplicação embarcada?

➤ Os middlewares são mediadores entre a aplicação e o S.O., implementando funções de comunicação que vão se aliar às funções básicas que o sistema operacional já implementa. Trata-se de uma camada de abstração colocada entre as plataformas e a aplicação, provendo diversas facilidades para o desenvolvedor abstrair a distribuição de recursos. Torna-se mais fácil para acessar recursos de uma outra plataforma, a qual pode estar usando outro S.O.

8. Como o sistema operacional pode influenciar a potência dissipada pelo sistema embarcado? Apresente dois efeitos.

➤ Um S.O eficiente vai poder operar numa frequência de clock mais rápida. Um S.O com alta configurabilidade vai ajudar a manter o custo de memória reduzido. Um S.O mais leve e mais eficiente vai ser capaz de trabalhar num processador com menos recursos; o processador sendo mais simples, será mais econômico por consequência. Esses fatores influem na potência.

9. Apresente dois efeitos da escolha do sistema operacional sobre o sistema embarcado que podem influenciar a potência dissipada pelo sistema.

➤ Um S.O eficiente vai poder operar numa frequência de clock mais rápida. Um S.O com alta configurabilidade vai ajudar a manter o custo de memória reduzido. Um S.O mais leve e mais eficiente vai ser capaz de trabalhar num processador com menos recursos; o processador sendo mais simples, será mais econômico por consequência. Esses fatores influem na potência.

Linguagens

1. Considerando as linguagens Assembly, C e Java para desenvolvimento de aplicações embarcadas, identifique os tipos de aplicação mais adequados para cada uma delas. Sugira plataformas de hardware mais adequadas para cada uma e dê um exemplo de aplicação.

➤ O Assembly pode ser usado em device-drivers, que necessitam acessar diretamente o circuito eletrônico da máquina, bem como em aplicações simples de tempo real, nas quais o atraso de uma função pode ser letal; ademais, as bibliotecas, de modo geral, são escritas em Assembly.

➤ A linguagem C suporta melhor a criação de aplicações mais complexas, por ser uma linguagem estruturada, sem perder aquela intimidade original do Assembly com a máquina, pois o C não impõe restrições de acesso ao hardware. Pode ser usada para fabricar sistemas operacionais, a maior parte do Windows, por exemplo, foi programada em C.

➤ Java é utilizado em desenvolvimento web, jogos, e, sobretudo, aplicativos de celular. O programador Java deve habituar-se à disciplina amarrada por esta linguagem, o que é bom para tornar os códigos mais reutilizáveis.

2. Caracterize as linguagens Assembly, C e Java para desenvolvimento de aplicações embarcadas em relação aos aspectos: facilidade de reuso, desempenho, e uso de memória.

➤ Quanto ao reuso, o Java se encontra no topo da lista, devido ao encapsulamento oferecido pela Programação Orientada a Objeto, e porque, ademais, o Java não permite ao programador referir-se à endereços de memória por meio de ponteiros, como é o caso do C, o que torna o código mais legível e conseqüentemente mais fácil de ser reciclado. Mais abaixo vem o C, que é mais reutilizável que o Assembly, por se tratar de uma linguagem estruturada e independente da máquina nativa.

➤ Quanto ao desempenho, a relação das linguagens acompanha a facilidade de acesso ao hardware, com o Assembly no topo, em vista da sua interferência direta no circuito eletrônico; o Assembly faz melhor emprego dos recursos de hardware. Logo a seguir, com menos desempenho, vem o C, acessando a memória através de ponteiros, e sofrendo atrasos em virtude da compilação. E em último lugar o Java, bloqueado pela Máquina Virtual, impedido de acessar ao hardware sem intermediários. O Assembly faz melhor emprego dos recursos de hardware.

➤ Quanto ao uso de memória o Assembly supera a todos, pois o código fica enxuto e as variáveis são representadas de maneira simples; a linguagem C perde desempenho em virtude dos custos adicionais envolvidos na compilação. O Java e o Python, linguagens de alto nível, com elevados graus de abstração, precisam reservar um excedente de memória para as estruturas de controle, o que as torna mais consumidoras.

3. Indique duas estratégias utilizadas por compiladores para diminuir o consumo de energia em aplicações embarcadas.

➤ Os compiladores podem ter funções adicionais destinadas a explorar os recursos que as máquinas podem oferecer, e que ficariam ociosos caso o desenvolvedor não os notasse. Acontece muitas vezes de esses recursos terem sido projetados para aumentar a eficiência energética das aplicações. Um compilador equipado com essas funções vai ser capaz de extrair da máquina o máximo de eficiência energética, reduzindo o tempo de execução e otimizando o acesso à memória.

➤ Outra forma de se abater o consumo de energia é aprimorar o desempenho do compilador na geração do código, o que reduz o tempo total de execução, e influi na energia gasta.

4. Sabe-se que mais de 70% do código das aplicações embarcadas foi escrito em assembly, para atender a requisitos de desempenho e uso de memória. Como os desenvolvedores lidam com esta dificuldade, já que é muito difícil encontrar programadores assembly?

➤ Utilizam bibliotecas feitas Assembly, a fim de que a maior parcela do código já se encontre traduzida e pronta para executar, o que restringe o tempo de compilação apenas à parcela de código redigida na linguagem dominada pelo desenvolvedor.

Software embarcado – análise e otimização

1. Determine o caminho MAIS LONGO para os fragmentos de código abaixo assumindo que todas as operações (atribuições) têm o mesmo custo computacional.

a)

```
for (i=0; i<32; i++)  
    if (a[i] < CONST2)  
        x[i] = a[i] * c[i];
```

O caminho mais longo é o caso de todos os elementos do vetor “a” serem menores que a variável CONST2, pois neste caso a instrução “x[i] = a[i] * c[i]” vai ser executada em todas as iterações do laço.

b)

```
if (a < CONST3) {  
    if (b < CONST4)  
        w = r + s;  
    else {  
        w = r - s;  
        x = s + t;  
    }  
} else {  
    if (c > CONST5) {  
        w = r + t;  
        x = r - t;  
        y = s + u;  
    }  
}
```

2. Para os trechos de código da questão anterior determine o caminho MAIS CURTO assumindo que todas as operações (atribuições) têm o mesmo custo computacional.

3. Dê um exemplo de laço (trecho de código em linguagem C) onde o tempo de computação seja INDEPENDENTE dos dados de entrada.

```
int i;
```

```
for (i = 0; i < 10; i++){  
  
    printf(“i: %d, i ** 2: %d\n”, i, i*i);  
  
}
```

4. Dê um exemplo de laço (trecho de código em linguagem C) onde o tempo de computação seja VARIÁVEL em função dos dados de entrada. Mostre como obter o maior e o menor tempo de computação para o seu código.

```

int tam;

scanf("%d", &tam);

int x[tam];

int i;

for(i = 0; i < tam; i++){

    scanf("%d", &x[i]);

}

printf("[");

for(i = 0; i < tam; i++){

    printf("%d, ", x[i]);

}

printf("]\n");

```

5. Para o código da questão anterior, descreva duas estratégias diferentes para verificar o tempo de execução do seu código em um processador real. Compare as duas estratégias quanto à sua viabilidade apontando pelo menos duas vantagens e duas desvantagens de cada uma em relação à outra.

6. Cite três estratégias para reduzir a energia gasta por uma aplicação embarcada. Classifique a sua lista por ordem de dificuldade de aplicação da técnica.

- (Mais difícil) Otimizar o acesso à memória
- (Dificuldade média) Trocar o algoritmo
- (Simples) Trocar o chip por sua versão Low-Power, ou desligar componentes ociosos do chip atualmente usado.

7. Cite três estratégias para reduzir o tamanho da memória gasta por uma aplicação embarcada. Classifique a sua lista por ordem de dificuldade de aplicação da técnica. Comente sobre o efeito de cada estratégia sobre memória de código e memória de dados.

- (Mais fácil) A forma mais simples é repetir os cálculos, no lugar de guardar seus resultados em variáveis, o que vai reduzir o consumo de memória apesar de prejudicar o tempo de execução
- (Dificuldade média) Dimensionar corretamente as variáveis, não alocando mais espaço que o necessário.

➤ (Dificuldade maior) Reuso de variáveis globais, essa estratégia vai evitar a criação de novas variáveis armazenar o resultado de diferentes operações, mas é um método difícil de se controlar, pois frequentemente essas variáveis vão ser disputadas e alteradas por diversos métodos, o que pode tornar difícil a detecção de possíveis erros.

8. Cite duas estratégias para medir a energia gasta por uma aplicação embarcada. Explique.

➤ Pode-se ligar um resistor Shunt em série com a carga na qual a aplicação está rodando, e medir a corrente elétrica que passa nele, e com isso, sabendo-se o valor da tensão, calcular a potência, uma vez que a corrente no Shunt é a mesma que flui para a máquina. A energia decorre naturalmente ao multiplicar-se a potência pelo tempo de execução, o qual pode ser obtido mediante métodos invasivos ou não invasivos. Um dos métodos invasivos é o timer embutido na biblioteca do processador.

➤ Pode-se medir a corrente elétrica por meio de um método direto, como a ponta de corrente, que consiste num aparelho que vai abraçar o fio que conecta a máquina à fonte, convertendo o campo magnético resultante deste fio no valor da corrente que nele circula. Daí, basta saber a tensão destinada à placa e o tempo de execução da aplicação para se obter a energia consumida.