

Um processador (CPU) pode apresentar **menor** \_\_\_\_\_ como desvantagem, se comparado a um circuito dedicado, mas tem **maior** \_\_\_\_\_ como vantagem.

**desempenho; flexibilidade**

**eficiência energética; flexibilidade**

Um processador (CPU) pode apresentar **maior** \_\_\_\_\_ como vantagem, se comparado a um circuito dedicado, mas tem **menor** \_\_\_\_\_ como desvantagem.

**flexibilidade; desempenho**

**flexibilidade; eficiência energética**

Um processador (CPU) apresenta maior \_\_\_\_\_ e menor \_\_\_\_\_ como **vantagens**, se comparado com um circuito dedicado.

**flexibilidade; custo por unidade**

**flexibilidade; custo**

**flexibilidade; preço por unidade**

**flexibilidade; preço**

Um processador (CPU) apresenta menor \_\_\_\_\_ e maior \_\_\_\_\_ como **vantagens**, se comparado com um circuito dedicado.

**custo por unidade; flexibilidade**

**custo; flexibilidade**

**preço por unidade; flexibilidade**

**preço; flexibilidade**

Um processador (CPU) apresenta menor \_\_\_\_\_ e **menor** \_\_\_\_\_ como **desvantagens**, se comparado com um circuito dedicado.

**eficiência energética, desempenho**

O DSP utiliza instruções do tipo SIMD. Isso permite diminuir o tempo de computação e a aumentar a eficiência energética.

**TRUE**

O \_\_\_\_\_ é um exemplo de processador soft-core.

**MicroBlaze**

**NIOS II**

**NIOS 2**

**OpenRISC**

**Leon3**

Um processador soft-core (implementado em FPGA) apresenta maior desempenho (velocidade) que um processador "de prateleira" equivalente.

**FALSE**

Um processador soft-core (implementado em FPGA) apresenta menor eficiência energética que um processador "de prateleira" equivalente.

**TRUE**

Um processador "de prateleira" apresenta menor desempenho (velocidade) que um processador soft-core (implementado em FPGA) equivalente.

**FALSE**

Um processador "de prateleira" apresenta maior eficiência energética que um processador soft-core (implementado em FPGA) equivalente.

**TRUE**

O FPGA e o ASSP são tecnologias disponíveis para se implementar uma aplicação embarcada. O ASSP oferece maior eficiência energética.

**TRUE**

O ASSP e o FPGA são equivalentes em sua capacidade de produzir hardware específico. Entretanto, o FPGA pode oferecer maior desempenho (velocidade) e eficiência energética.

**FALSE**

O ASIC e o FPGA são equivalentes em sua capacidade de produzir hardware específico. Entretanto, o ASIC pode oferecer maior desempenho (velocidade) e eficiência energética.

**TRUE**

O FPGA e o ASIC são equivalentes em sua capacidade de produzir hardware específico. Entretanto, o FPGA pode oferecer maior flexibilidade e menor time-to-market.

**TRUE**

O FPGA e o ASIC são equivalentes em sua capacidade de produzir hardware específico. Entretanto, o ASIC apresenta menor time-to-market.

**FALSE**

O ASIC e o FPGA são equivalentes em sua capacidade de produzir hardware específico. Entretanto, o FPGA pode oferecer maior flexibilidade e menor tempo de projeto.

**TRUE**

Sistemas híbridos são SoC que oferecem \_\_\_\_\_ como parte de sua lógica interna.

**um FPGA**

**um módulo reconfigurável**

**um reconfigurável**

SoC que oferecem \_\_\_\_\_ como parte de sua lógica interna são chamados sistemas híbridos.

**um FPGA**

**um módulo reconfigurável**

**um reconfigurável**

Quando é necessário prolongar os cabos de conexão entre computadores ligados por uma interface UART, uma das estratégias é aumentar os níveis de tensão. Os \_\_\_\_\_ podem ser usados para isso.

**drivers RS-232**

Os \_\_\_\_\_ usados com a interface UART de um microcontrolador permitem elevar as tensões aplicadas no cabo de conexão.

**drivers RS-232**

Para aumentar o alcance da interface UART de um microcontrolador se elevam as tensões aplicadas no cabo de conexão. Isso pode ser feito pelo(s) \_\_\_\_\_.

**drivers RS-232**

Um dos motivos para a Linguagem C ser predominante no desenvolvimento de aplicações embarcadas é a disponibilidade de compiladores para todas as plataformas.

**TRUE**

A disponibilidade de compiladores para todas as plataformas é um dos motivos para a Linguagem C ser predominante no desenvolvimento de aplicações embarcadas.

**TRUE**

A linguagem C tem acesso mais facilitado ao hardware. Isso torna a execução do programa mais rápido, comparado com linguagens orientadas a objeto.

**FALSE**

\_\_\_\_\_ foi a linguagem de programação usada nas primeiras aplicações embarcadas. Uma das razões para isso era \_\_\_\_\_.

**Assembly, a falta de compiladores**

**Assembly, a simplicidade das aplicações**

**Assembly, a baixa complexidade das aplicações**

As aplicações embarcadas antigamente eram escritas em linguagem \_\_\_\_\_.  
\_\_\_\_\_ era uma das razões para isso.

**Assembly, a falta de compiladores**

**Assembly, a baixa complexidade das aplicações**

As primeiras aplicações embarcadas eram escritas em linguagem \_\_\_\_\_. Uma das razões para isso era \_\_\_\_\_.

**Assembly, a falta de compiladores**

**Assembly, a baixa complexidade das aplicações**

As primeiras aplicações embarcadas eram escritas em linguagem \_\_\_\_\_.  
\_\_\_\_\_ e \_\_\_\_\_ eram as razões para isso.

**Assembly, A falta de compiladores, a baixa complexidade das aplicações**

**Assembly, A baixa complexidade das aplicações, a falta de compiladores**

**Assembly, A simplicidade das aplicações, a falta de compiladores**

No padrão SPI o endereço de destino da mensagem vai na própria mensagem.

**FALSE**

Na comunicação síncrona, transmissor e receptor usam o mesmo sinal de clock. É o caso do SPI.

**TRUE**

O padrão I2C utiliza apenas um fio para tráfego de dados enquanto o SPI utiliza dois.

**TRUE**

SPI e I2C adotam comunicação síncrona, transmissor e receptor usam o mesmo sinal de clock.

**TRUE**

Os padrões I2C e SPI permitem operar a uma taxa de transmissão maior que a UART.

**TRUE**

Nos barramentos de comunicação, como o I2C, os mesmos sinais conectam todos os dispositivos. Já no padrão UART, a conexão é ponto a ponto.

**TRUE**

Os processadores VLIW dissipam menor potência que os superescalares convencionais. Isso é consequência de seu menor desempenho.

**FALSE**

Os processadores VLIW apresentam menor desempenho que os superescalares convencionais, por isso dissipam menor potência.

**FALSE**

Os processadores VLIW possuem menor número de unidades funcionais que os superescalares convencionais, diminuindo a potência dissipada.

**FALSE**

Com os processadores VLIW (EPIC) a detecção de paralelismo é movida do processador para o compilador. Isso evita gastos com silício e energia em tempo de execução, aumentando a eficiência energética em relação aos superescalares normais.

**TRUE**

Um motor opera a 500 rpm quando submetido a uma tensão de 5,0V. Considere a disponibilidade de um driver PWM que fornece 5V quando em nível alto e 0V quando em nível baixo. O valor de ciclo de trabalho (razão cíclica ou Duty cycle) para que o motor opere a 300 rpm é \_\_\_\_\_.

**0,6 (porcentagem)**

Um motor opera a 600 rpm quando submetido a uma tensão de 5,0V. Considere a disponibilidade de um driver PWM que fornece 5V quando em nível alto e 0V quando em nível baixo. O valor de ciclo de trabalho (razão cíclica ou Duty cycle) para que o motor opere a 450 rpm é \_\_\_\_\_.

**0,75 (porcentagem)**

Um motor opera a 1000 rpm quando submetido a uma tensão de 5,0V. Considere a disponibilidade de um driver PWM que fornece 5V quando em nível alto e 0V quando em nível baixo. O valor de ciclo de trabalho (razão cíclica ou Duty cycle) para que o motor opere a 450 rpm é \_\_\_\_\_.

**0,45 (porcentagem)**

O termo duty cycle (ciclo de trabalho), na modulação conhecida como PWM, descreve a proporção de tempo ligado em relação a um período de tempo.

**TRUE**

O uso de linguagens de mais alto nível de abstração em aplicações embarcadas vem sendo imposto pelo aumento da complexidade dessas aplicações.

**TRUE**

O aumento da complexidade das aplicações embarcadas levou ao uso de linguagens de mais alto nível de abstração.

**TRUE**

O aumento da complexidade das aplicações embarcadas levou à necessidade de combinar em um projeto diferentes linguagens, como Assembly e Python.

**FALSE**

O critério de Nyquist diz que a frequência de amostragem para digitalização de um sinal deve ser maior ou igual ao dobro da maior frequência contida no sinal.

**TRUE**

O critério de Nyquist diz que a frequência de amostragem para digitalização de um sinal deve ser menor ou igual à metade da maior frequência contida no sinal.

**FALSE**

Segundo o critério de Nyquist, a maior frequência contida num sinal não deve ultrapassar a metade da frequência de amostragem para digitalização do sinal.

**TRUE**

Segundo o critério de Nyquist, a frequência de amostragem para digitalização de um sinal deve ser maior ou igual à maior frequência contida no sinal.

**FALSE**

O(s) \_\_\_\_\_ apresentam menor flexibilidade e maior tempo de prototipação como desvantagens em relação aos Reconfiguráveis.

**ASIC**

Os Reconfiguráveis oferecem como vantagens sobre o(s) \_\_\_\_\_ maior flexibilidade e menor tempo de prototipação.

**ASIC**

Uma estratégia para diminuir o tempo de computação de uma aplicação é a introdução de cache no sistema. A desvantagem desse método é o aumento da potência.

**FALSE**

A introdução de cache no sistema é uma estratégia para diminuir o tempo de computação de uma aplicação. A desvantagem desse método é o aumento da potência.

**FALSE**

A introdução de cache no sistema é uma estratégia para diminuir o tempo de computação de uma aplicação. Adicionalmente, esse método ajuda a diminuir a potência.

**TRUE**

Uma estratégia para diminuir o tempo de computação de uma aplicação é a introdução de cache no sistema. Adicionalmente, esse método ajuda a diminuir a potência.

**TRUE**

Um Cross compiler é um compilador sendo executado em uma plataforma e gerando código executável em outra plataforma.

**TRUE**

Um compilador sendo executado em uma plataforma e gerando código executável em outra plataforma é chamado Cross compiler.

**TRUE**

Uma estratégia para diminuir o tempo de computação de uma aplicação é substituir o algoritmo. A desvantagem desse método é o possível aumento da energia.

**FALSE**

Substituir o algoritmo é uma estratégia para diminuir o tempo de computação de uma aplicação. Adicionalmente, esse método pode diminuir a energia.

**TRUE**

Uma estratégia para diminuir o tempo de computação de uma aplicação é aumentar a frequência de operação do processador. A desvantagem desse método é o possível aumento da energia.

**FALSE**

Aumentar a frequência de operação do processador é uma estratégia para diminuir o tempo de computação de uma aplicação. A desvantagem desse método é o possível aumento da energia.

**FALSE**

Ao se aplicar uma estratégia que reduza a potência de um sistema digital haverá um benefício extra de também reduzir a sua energia.

**FALSE**

Para implementar uma função trigonométrica um programador C deve usar uma biblioteca (math.h) ao invés dele mesmo escrever em C o código que calcula a função (Usando série de Taylor, por exemplo). No contexto de aplicações embarcadas isso se faz porque \_\_\_\_\_, oferecendo **menor tempo de computação**.

**a lib foi escrita originalmente em assembly**

**a biblioteca foi escrita originalmente em assembly**

Para implementar uma função trigonométrica um programador C deve usar uma biblioteca (math.h) ao invés dele mesmo escrever em C o código que calcula a função (Usando série de Taylor, por exemplo). No contexto de aplicações embarcadas isso se faz porque \_\_\_\_\_, o que implica em **menor tempo de desenvolvimento**.

**o código da biblioteca já está pronto**

**o código da lib já está pronto**

Nem sempre uma estratégia que reduz a potência de um sistema digital leva a uma redução na sua energia.

**TRUE**

Uma redução de energia de um sistema digital necessariamente está associada a uma redução na potência.

**FALSE**

Nem sempre uma estratégia que reduz a potência de um sistema digital leva a uma redução na sua energia. Já uma redução de energia necessariamente está associada a uma redução na potência.

**FALSE**

O DSP é um exemplo de arquitetura (ISA) personalizada pra um certo domínio de aplicação. Com isso se pode diminuir o tempo de computação sem ter que recorrer a altas frequências de clock.

**TRUE**

Um exemplo de arquitetura (ISA) personalizada pra um certo domínio de aplicação é o DSP. Essa estratégia permite uma menor potência dissipada pelo dispositivo podendo manter o tempo de computação de um processador de propósito geral.

**TRUE**

Diminuir a memória de um SoC tem um benefício duplo, no preço e na potência dissipada.

**TRUE**

Processador, Memórias e dispositivos de E/S são os elementos que compõem um SoC (System on Chip).

**TRUE**



Um SoC (System on Chip) é composto de Processador, Memórias e dispositivos de E/S.

**TRUE**

UART adota comunicação síncrona, transmissor e receptor usam o mesmo sinal de clock.

**FALSE**

Na comunicação síncrona, transmissor e receptor usam o mesmo sinal de clock. É o caso da UART.

**FALSE**

Estudos indicam que cerca de 75% do código de aplicações embarcadas em microcontroladores foi escrito em Assembly. Isso ocorre porque a equipe do projeto usou \_\_\_\_\_.

**bibliotecas escritas em assembly**

Dado um ADC com resolução de 10 bits, frequência de clock de 200 kHz e que utiliza o método de integração simples, o tempo de uma conversão é de \_\_\_\_\_.

**$(2^{10})/200\text{Khz} = 5,12\text{ms}$  ( $2^{\text{Bits/Frequencia.clk}}$ )**

Dado um ADC com resolução de 10 bits, frequência de clock de 200 kHz e que utiliza o método de aproximação sucessiva, o tempo de uma conversão é de \_\_\_\_\_ us.

**$10/200\text{Khz} = 50$  (bits/frequencia.clk)**

Dado um ADC com resolução de 12 bits, frequência de clock de 100 kHz e que utiliza o método de aproximação sucessiva, o tempo de uma conversão é de \_\_\_\_\_ us.

**$12/100\text{Khz} = 120$  (bits/frequencia.clk)**

A técnica de salto de frequência do padrão Bluetooth contribui para diminuir a sua potência de operação.

**FALSE**

A técnica de salto de frequência do padrão Bluetooth contribui para aumentar a sua taxa de transmissão.

**FALSE**

A técnica de salto de frequência do padrão Bluetooth contribui para aumentar a sua robustez.

**TRUE**

O padrão Bluetooth opera a uma taxa de comunicação menor que o Wi-fi, mas o que o torna atraente é a menor potência dissipada.

**TRUE**

Um ASIC implementa uma lógica dedicada (específica) para uma aplicação. Essa estratégia permite uma menor potência dissipada pelo dispositivo podendo manter o tempo de computação de um processador de propósito geral.

**TRUE**

Compiladores não são muito eficientes ao traduzir código a partir da linguagem de alto nível. A solução para isso é os projetos de aplicações embarcadas contarem com algum desenvolvedor em Assembly.

**FALSE**

Um ASIP possui instruções especializadas para atender a um certo domínio de aplicações. Isso aumenta o seu time-to-market, comparado com um microprocessador, já que requer programadores especializados.

**FALSE**

As funções (em software) que usam os recursos de um ASIP (Application-Specific Instruction set Processor) são escritas originalmente em assembly.

**TRUE**

Medimos o tempo de execução de um trecho de código usando o osciloscópio para observar os sinais Tx e Rx.

**FALSE**

A taxa (ou frequência) de um conversor analógico-digital se refere ao número de digitalizações feitas por unidade de tempo.

**TRUE**

\_\_\_\_\_ é o tempo necessário desde a concepção de um projeto até o seu lançamento no mercado.

**Time-to-market**

O tempo necessário desde a concepção de um projeto até o seu lançamento no mercado é chamado \_\_\_\_\_.

**Time-to-market**

Aumentar a memória de um SoC tem um prejuízo duplo, no preço e na potência dissipada.

**TRUE**

Considere um sistema que é ativado em intervalos regulares de 20 minutos. O processo (computação/comunicação) dura 10 segundos e durante esse tempo o consumo é de 50mA. Depois disso o sistema entra em repouso e seu consumo é desprezível.

A alimentação é feita por baterias de 3V. As baterias devem ser capazes de fornecer uma potência de \_\_\_\_ mW. Se for usada uma bateria de 2000mAh, ele precisará ser substituída ou recarregada em \_\_\_\_ horas. \*

$$50 \text{ mA} \times 3 \text{ V} = 150 \text{ mW}; 4800$$

$$3 \text{ v} \times 2000 \text{ mAh} = 6000 \text{ mWh} = 6000 \times 60 \text{ (min)} \times 60 \text{ (sec)} = 21600000 \text{ mWs}$$

$$150 \text{ mW} \times 10 \text{ (sec)} \times 3 \text{ (x por h)} = 4500$$

$$21.600.000 \text{ mWs} / 4500 = 4800$$

$$21600000$$

$$60 \times 30 \times 3 = 6400$$

$$21600000 / 6400$$

$$3 \text{ v}$$

$$20 \text{ minutos}$$

$$30 \text{ segundos}$$

$$20 \text{ mA}$$

**FORMS????????????????**

A comunicação por par trançado é mais robusta que a tradicional (sinalgnd). Isso se deve ao uso de circuitos eletrônicos mais rápidos pra implementar a conexão com o meio físico de comunicação.

**FALSE**

Dado um ADC com resolução de 12 bits, frequência de clock de 100 kHz e que utiliza o método de aproximação sucessiva, o tempo de uma conversão é de 120 us.

**TRUE**

Processadores com conjunto de instruções comprimidas (como o Thumb do ARM) permitem otimizar o tempo de computação pela diminuição do tempo de acesso à memória.

**FALSE**

É dada uma aplicação embarcada em uma plataforma que consome 400mA com uma tensão de 3,3V. Uma bateria de 3,6V deve ser capaz de fornecer no mínimo \_\_\_\_\_ W para alimentar esse sistema.

**1,32**

Considere uma aplicação que é ativada em períodos regulares (T). Ela roda por um tempo (t1) e desliga o sistema no restante do tempo. Mantendo o algoritmo e a tensão do processador e diminuindo a frequência de clock, a Potência máxima do sistema vai \_\_\_\_\_ e a Energia vai \_\_\_\_\_; desde que o novo tempo de computação (t2) seja menor do que \_\_\_\_\_.

**diminuir; se manter; T**

\_\_\_\_\_ é um barramento de comunicação, em que os mesmos fios conectam todos os dispositivos (nós). Já no padrão \_\_\_\_\_, a conexão é ponto a ponto

**I2C; UART**

Tanto o \_\_\_\_\_ quanto o \_\_\_\_\_ são chips projetados para uma aplicação específica. A diferença entre eles está no(a) \_\_\_\_\_.

**ASIC; ASSP; mercado alvo**

Um motor opera a 500 rpm quando submetido a uma tensão de 5,0V. Considere a disponibilidade de um driver PWM que fornece 5V quando em nível alto e 0V quando em nível baixo. \_\_\_\_\_ é o valor do ciclo de trabalho (razão cíclica ou Duty cycle) para que o motor opere a 450 rpm.

**0,9**

Os processadores de 8 bits apresentam menor \_\_\_\_\_ e menor \_\_\_\_\_ como vantagens sobre os de 32 bits.

**preço; potência**

A \_\_\_\_\_ de um Conversor Digital-Analógico está ligada ao número de bits que ele utiliza.  
**resolução**

O modelo de programação “laço combinado com serviço de interrupção” é superior ao modelo \_\_\_\_\_ para aplicações embarcadas com tarefas que sejam sensíveis a atrasos.

**laço simples**

Um System on Chip é composto de \_\_\_\_\_, \_\_\_\_\_ e dispositivos de E/S.

Processador; Memórias

Defina time-to-market no contexto de sistemas embarcados.

**Tempo necessário desde a idealização de uma produto até a sua chega ao mercado.  
Fundamental na competição entre os produtos de eletrônica de consumo, principalmente.**