

Questão 1 - Dê exemplos de aparelhos que utilizam ISA RISC.

A ARM, desenvolvedora de microprocessadores, detém 75% do mercado de processadores RISC de 32-bit, incluindo a grande maioria dos smartphones atuais. MIPS, que na década de 90 detinha 1/3 do mercado, e é encontrado em roteadores da Cisco, dispositivos com Windows CE, e videogames como Nintendo 64 e Playstation.¹

Questão 2 - Explique o que é o projeto RISC-V.

O RISC-V é um *ISA (Instruction Set Architecture) open-source* criado com o objetivo de criar um conjunto de instruções universal e livre, que seja adequado para quase todo tipo de implementação, suportando desde projetos embarcados de pequeno porte, até servidores ou celulares. Parte essencial de sua universalidade, é a possibilidade de se adicionar extensões ao RISC-V.

Questão 3 - RISC-V é utilizado só para uso acadêmico?

Não. Por ser *open-source*, não é necessário a compra de uma licença para o uso do RISC-V, e ele já é utilizado em indústrias de grande porte.

Questão 4 - Quais são as extensões oficiais do RISC-V ditas em aula e qual o propósito de cada uma?

A extensão de Multiplicação (M) adiciona quatro instruções de multiplicação, duas de divisão e duas de manipulação de restos. Sua base é indicada pela base do ISA. Preza pela simplicidade ao, por exemplo, não causar interrupções ao se deparar com divisões por zero.

A extensão de Sincronização (A) adiciona onze instruções de sincronização, divididas em dois grupos: *Load Reserved* e *Store Conditional* para operações atômicas na memória. Visam a consistência e atomicidade da operação.

A extensão de Ponto Flutuante é separada em três extensões diferentes: Precisão Simples (F), Precisão Dupla (D) e Precisão Quádrupla (Q). A extensão F é pré-requisito para a extensão D, que é pré-requisito para a extensão Q. Possui um registrador de 5 bits para exceções, sendo que estas não geram interrupções e devem ser verificadas por meio de consultas.

A extensão de Compressão de Tamanho de Código (C) não acrescenta nenhuma outra função, porém codifica instruções inteiras, compactadas em 16 bit, a fim de salvar espaço, reduzindo o tamanho do *footprint*.

¹ Fonte: http://web.uvic.ca/~epurcell/group_web_page.html

Questão 5 - É possível adicionar minha própria extensão, cujo nome é extensãoUFOP ao projeto?

Sim. RISC-V é um ISA *open-source* criado para suportar extensões e arquiteturas customizadas.²

Questão 6 - Quantos registradores o RISC-V possui e quais são eles? Esse número pode ser alterado?

O RISC-V possui 32 registradores inteiros e 32 registradores opcionais para ponto flutuante, incluído junto com o pacote de extensão de operações de ponto flutuante. Também existe uma variante de pequeno porte do RISC-V com apenas 16 registradores inteiros. O número de registradores não pode ser alterado.

Questão 7 - Cite duas otimizações realizadas no projeto do RISC-V.

O projeto RISC-v foi desenvolvido com foco na praticidade das implementações, com características que aumentam a velocidade computacional ao mesmo tempo em que reduz o custo e energia. Para isso, foram feitas otimizações tais como colocar os bits mais significantes em uma posição fixa, e a disposição de bits a fim de reduzir o número de multiplexadores no chip em sua implementação.

Questão 8 - RISC-V possui operações de multiplicação com acesso direto à memória principal?

Não. Por ser uma máquina de *load-store*, apenas essas duas instruções possuem acesso à memória principal, e todas as outras operações lógico-aritméticas ocorrem apenas entre registradores.

Questão 9 - Qual é o propósito de utilizar a extensão C?

Reduzir o tamanho do código binário, energia e custo para pequenos computadores, visando sistemas embarcados.

Questão 10 - Existe alguma diferença entre os formatos de operações do RV32I, RV64I, e a Extensão C?

Não. Os formatos de operações permanecem idênticos, apesar da diferença de espaço de endereço e tamanho de instruções.

² Fonte: <http://ieeexplore.ieee.org/document/7760808/>

Questão 11 - RISC-V possui o condicional *if*? Como ele realiza suas instruções de condição?

O RISC-V intencionalmente não possui códigos condicionais ou bit de carry, a fim de simplificar o desenvolvimento do CPU minimizando interações entre instruções. Instruções de condição são feitas através de operandos de comparação dentro dos *jumps* condicionais.

Questão 12 - Observando as posições dos registradores na tabela de formato de instruções, por que eles estão situados na mesma posição em todas os formatos?

A fim de facilitar a decodificação, *rd*, *rs1* e *rs2* estão sempre na mesma posição, e imediatos estão sempre no final da instrução.

Questão 13 - Qual a diferença das instruções *jump* e *jump and link*?

O *jump and link* é similar ao *jump*, mas com um endereço de retorno.

Questão 14 - O que é o Rocket-Chip e quantos estágios ele possui?

Rocket-chip é um gerador parametrizado SoC escrito em linguagem Chisel. Possui um pipeline de 5 estágios em ordem.³

Questão 15 - Explique sucintamente como Rocket-Chip é utilizado.

O Rocket-Chip é disponibilizado principalmente para desenvolvedores de sistemas embarcados, pessoas que necessitam de mais que apenas utilizar o produto – que precisam desenvolver e modificar este. Assim, para cada tipo de requisito, deve-se procurar o respectivo parâmetro de configuração.³

³ Fonte: <http://www.decom.ufop.br/imobilis/rocket-chip-soc-generator-chisel/>