



Universidade Federal de Ouro Preto
Instituto de Ciências Exatas e Biológicas
Departamento de Computação
Arquitetura de Computadores
Ana Luiza Fernandes Moraes - 15.1.4244

- 1- O Playstation, Nintendo e iPhone são exemplos de aparelhos que utilizam a arquitetura RISC.
- 2- O projeto RISC-V é um ISA *open-source* desenvolvido por profissionais, professores e voluntários no ramo da Arquitetura de Computação com base em uma série de outros projetos acadêmicos de design de computadores. O ISA RISC-V é um Conjunto de Instruções para Computador de Propósito Geral moderno e de alta qualidade. RISC-V é um projeto cujas instruções são de fácil implementação comparado a outras alternativas existentes no mercado.
- 3- Não. A meta do projeto RISC-V é criar um conjunto de instruções 'universal', que é livre e aberto para todos os usuários, provendo tudo que é necessário para suportar perfeitamente qualquer projeto comercial. Apesar de ser um projeto novo, possui grande aceitação na indústria de semicondutores.
- 4- As extensões do RISC-V são M, A, F, D, Q e C. Multiplicação M: adiciona 4 instruções de multiplicação, duas de divisão e duas de manipulação de restos; Sincronização A: adiciona 11 instruções de sincronização visando consistência e atomicidade da operação; F: precisão simples; D: Precisão dupla; Q: quádrupla precisão; C: codifica as instruções inteiras para salvar espaço e com isso reduzir o tamanho do *footprint*.
- 5- Sim. A extensibilidade do RISC-V é a parte essencial da sua universalidade. Por isso, pode-se adicionar módulos e ter uma arquitetura que atende a todos os projetos de hardware.
- 6- O RISC-V possui 32 registradores internos e 32 registradores opcionais para ponto flutuante. Os registradores opcionais de ponto flutuante são incluídos junto com o pacote de extensão de operações de ponto flutuante.
- 7- O RISC-V foi desenvolvido com foco na praticidade das implementações, com características que aumentam a velocidade computacional enquanto

reduz seu custo e energia. Dentre as otimizações podemos citar a colocada dos bits mais significantes numa posição fixa e a disposição de bits para reduzir o número de multiplexadores no chip em sua implementação.

- 8- Não. O RISC-V não possui *carry* de operações aritméticas complicadas como a multiplicação e a divisão.
- 9- A extensão C codifica as instruções inteiras para salvar espaço e com isso reduzir o tamanho do *footprint*.
- 10- O RV32I possui o seguinte formato de operações: ADDI, SLTI[U], ANDI, ORI, XORI; já o RV64I e a Extensão C possuem o ANDIW;
- 11- O RISC-V não possui códigos condicionais, nem mesmo bit de *carry*.
- 12- Deixar os registradores rd, rs1 e rs2 na mesma posição em todos os formatos é um método utilizado para facilitação da decodificação. Neste método os imediatos estão sempre no final da instrução.
- 13- A instrução *jump* desvia para o procedimento. A instrução *jump and link* desvia para o procedimento e salva o endereço de retorno.
- 14- A ferramenta *Rocket-Chip* permite gerar diferentes configurações de RISC-V para diferentes ambientes SoC e possui cinco estágios.
- 15- As configurações geradas pela ferramenta são especificadas por meio de parâmetros em linguagem *Chisel* que podem ser alterados livremente e com isso, pode-se selecionar o que se quer gerar como chip em nosso projeto almejado.