

UNIVERSIDADE FEDERAL DE OURO PRETO

Lista: RISC-V – 24 de janeiro de 2017

Orientação: Rodolfo Labiapari - rodolfolabiapari@decom.ufop.br

Arquitetura de Computadores - BCC 236

Aluna: Sântia de Sena Coelho 15.2.5909

Questão 1 Dê exemplos de aparelhos que utilizam ISA RISC.

Projetos embarcados de pequeno porte até servidores, celulares high-end e usada em processadores PowerPC (da Apple, Motorola e IBM) e SPARC (SUN), DEC Alpha, SPARC, MIPS, e PowerPC.

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

É um conjunto limpo, modular, com inteiros bases em 32, 64 e 128 bits e várias opções de extensão de instruções como ponto-flutuante, multiplicadores, etc.

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.

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

RISC-V é disponibilizado livremente para todos usarem bem como quiserem incluindo fazer design, fabricar e vender os chips e software RISC-V.

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

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.

Ponto Flutuante:

- Precisão simples(F), Precisão Dupla(D), e quádrupla precisão (Q).

Extensão: Compressão de tamanho de código: extensão C não adiciona nenhuma outra função, mas, ao invés disso, **codifica** as instruções inteiras para **salvar espaço** e com isso reduzir o tamanho do *footprint*. É disponível para bases inteiras, bem como *load* e *store* para pontos flutuantes e cria-se então instruções comprimidas em 16 bit. Basicamente, cada função compactada é mapeada diretamente à instrução real e possui algumas restrições para a compressão sobre o formato dos operandos.

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

Sim, a maneira mais simples de criar uma nova extensão é implementar a interface RoCC.

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

RISC-V possui 32 registradores inteiros, e 32 registradores opcionais para ponto flutuante. Também existe uma variante RISC-V com 16 registradores inteiro e sua memória possui endereçamento de 1 byte.

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

- 1) Colocar os bits mais significantes numa posição fixa;
- 2) Disposição de bits para reduzir o número de multiplexadores no CPU;

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

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

O Propósito é reduzir o tamanho do código binário. Ela codifica as instruções inteiras para **salvar espaço** e com isso reduzir o tamanho do *footprint*.

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

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

Não possui códigos condicionais. Construíram 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?

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

jump: desvio condicional para um endereço de memória apontado por um label.

jump and link: salta para o endereço especificado, salvando o endereço da próxima instrução em \$ra.

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

É um gerador open-source de sistemas com cores em ordem que implementam variações do RISC-V ISA. É um projeto altamente “parameterizável” para gerar vários tipos de designs possíveis a abranger a maior quantidade de projetos possível.

O projeto baseia-se no Rocket, versão de 2014 onde possui 5 estágios em ordem baseando na implementação do RV64G.

- **RoCCIO:** Interface de comunicação entre Rocket *Core* e o Acelerador;
- **HTIFIO:** Leitura e Escrita no CSR;
- **TileLinkIO:** Comunicação Interna entre blocos;
- **MemIO:** Interface simples de memória AXI-like;
- **HostIO:** Interface central para HTIF.

Questão 15 Explique sucintamente como Rocket-Chip é utilizado.

Referências

<http://www.diegomacedo.com.br/arquitetura-de-processadores-risc-e-cisc/>

<http://www.decom.ufop.br/imobilis/o-risc-v/>

http://www.cpdee.ufmg.br/~frank/lectures/SPP/SPP-aula04-Conjunto_de_Instrucoes_MIPS_3.pdf

<http://www.decom.ufop.br/imobilis/soc-risc-v-e-energia/>

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