



Universidade de Brasília

Instituto de Ciências Exatas
Departamento de Ciência da Computação

RISC-V: Ambiente de montagem e simulação

Matheus Y. Matsumoto

Monografia apresentada como requisito parcial
para conclusão do Curso de Engenharia da Computação

Orientador
Prof. Dr. Ricardo Pezzuol Jacobi

Brasília
2017



Universidade de Brasília

Instituto de Ciências Exatas
Departamento de Ciência da Computação

RISC-V: Ambiente de montagem e simulação

Matheus Y. Matsumoto

Monografia apresentada como requisito parcial
para conclusão do Curso de Engenharia da Computação

Prof. Dr. Ricardo Pezzuol Jacobi (Orientador)
CIC/UnB

Prof. Dr. Donald Knuth Dr. Leslie Lamport
Stanford University Microsoft Research

Prof. Dr. Ricardo Pezzuol Jacobi
Coordenador do Curso de Engenharia da Computação

Brasília, 26 de março de 2017

Dedicatória

Na *dedicatória* o autor presta homenagem a alguma pessoa (ou grupo de pessoas) que têm significado especial na vida pessoal ou profissional. Por exemplo (e citando o poeta):
Eu dedico essa música a primeira garota que tá sentada ali na fila. Brigado!

Agradecimentos

Nos *agradecimentos*, o autor se dirige a pessoas ou instituições que contribuíram para elaboração do trabalho apresentado. Por exemplo: *Agradeço aos gigantes cujos ombros me permitiram enxergar mais longe. E a Google e Wikipédia.*

Resumo

O *resumo* é um texto inaugural para quem quer conhecer o trabalho, deve conter uma breve descrição de todo o trabalho (apenas um parágrafo). Portanto, só deve ser escrito após o texto estar pronto. Não é uma coletânea de frases recortadas do trabalho, mas uma apresentação concisa dos pontos relevantes, de modo que o leitor tenha uma ideia completa do que lhe espera. Uma sugestão é que seja composto por quatro pontos: 1) o que está sendo proposto, 2) qual o mérito da proposta, 3) como a proposta foi avaliada/validada, 4) quais as possibilidades para trabalhos futuros. É seguido de (geralmente) três palavras-chave que devem indicar claramente a que se refere o seu trabalho. Por exemplo: *Este trabalho apresenta informações úteis a produção de trabalhos científicos para descrever e exemplificar como utilizar a classe L^AT_EX do Departamento de Ciência da Computação da Universidade de Brasília para gerar documentos. A classe UnB-CIC define um padrão de formato para textos do CIC, facilitando a geração de textos e permitindo que os autores foquem apenas no conteúdo. O formato foi aprovado pelos professores do Departamento e utilizado para gerar este documento. Melhorias futuras incluem manutenção contínua da classe e aprimoramento do texto explicativo.*

Palavras-chave: risc, LaTeX, metodologia científica, trabalho de conclusão de curso

Abstract

O *abstract* é o resumo feito na língua Inglesa. Embora o conteúdo apresentado deva ser o mesmo, este texto não deve ser a tradução literal de cada palavra ou frase do resumo, muito menos feito em um tradutor automático. É uma língua diferente e o texto deveria ser escrito de acordo com suas nuances (aproveite para ler [http://dx.doi.org/10.6061/2Fclinics%2F2014\(03\)01](http://dx.doi.org/10.6061/2Fclinics%2F2014(03)01)). Por exemplo: *This work presents useful information on how to create a scientific text to describe and provide examples of how to use the Computer Science Department's L^AT_EX class. The UnB-CIC class defines a standard format for texts, simplifying the process of generating CIC documents and enabling authors to focus only on content. The standard was approved by the Department's professors and used to create this document. Future work includes continued support for the class and improvements on the explanatory text.*

Keywords: LaTeX, scientific method, thesis

Sumário

1	Introdução	1
1.1	RISC-V	1
1.1.1	Objetivos	1
1.1.2	História	2
1.1.3	RISC-V Foundation	2
1.1.4	Open Source	3
1.2	Objetivos e motivação deste projeto	3
1.3	Processadores MIPS, ARM e RISC-V	4
2	Fundamentação teórica	5
2.1	Arquitetura RISC-V: ISA	5
2.1.1	Diferenças	5
2.1.2	Características	5
2.1.3	Como funciona	5
2.1.4	Instruções	5
2.2	Montador	6
2.2.1	Conceito	6
2.2.2	Algoritmo de duas passagens	6
2.3	Aplicações web	6
2.3.1	Arquitetura	6
2.3.2	Vantagens e desvantagens	6
2.4	Escalabilidade (novos modulos)	6
3	Ambiente Proposto	7
3.1	Ambiente proposto	7
3.2	Arquitetura de software	7
3.3	Interface web	7
3.4	Simulador	7
3.5	Características	7

3.6 Montador	7
3.7 Novos modulos	7
4 Resultados e Avaliação do Sistema	9
4.1 entradas do sistema	9
4.2 saidas do sistema	9
4.3 simulação	9
5 Conclusões	10
5.1 objetivos atingidos	10
5.2 pontos positivos e negativos	10
5.3 dificuldades	10
5.4 melhorias	10
5.5 trabalhos futuros	10
Referências	11

Capítulo 1

Introdução

1.1 RISC-V

RISC-V é uma arquitetura de conjunto de instruções aberta, criada na Universidade da Califórnia, em Berkeley. Originalmente foi pensada para ser utilizada na pesquisa e ensino da área de arquitetura de computadores, mas está se tornando um padrão de arquitetura aberta para a indústria. [1]

Sua arquitetura obedece aos padrões RISC (Reduced Instruction Set Computing), tendo instruções simples e completas. Foi projetada para ser rápida, ocupar pouco espaço físico, ter baixo consumo de energia, ser extensível, e compatível com entre suas versões.

Seu nome é pronunciado na língua inglesa como "*risc five*". O motivo de ser "*five*" é devido ao fato de que é o quinto maior projeto de uma ISA RISC desenvolvida na UC Berkeley. As primeiras foram RISC-I, RISC-II, SOAR, e SPUR. O numeral romano "V" de RISC-V também funciona com significado de "*variations*" e "*vectors*".

1.1.1 Objetivos

Seus projetistas sempre são perguntados o motivo ao qual eles quiseram desenvolver uma nova ISA. Alguns dos motivos para o qual usar uma ISA comercial são a existência de suporte de um ecossistema de software, tanto ferramentas de desenvolvimento, portabilidade e ferramentas educacionais, outros benefícios seriam a grande quantidade de documentação, tutoriais e exemplos para o desenvolvimento.

Porém estas vantagens são pequenas na prática, e listam várias desvantagens ao utilizar ISAs comerciais,

- ISAs comerciais são proprietárias

- ISAs comerciais são populares somente em alguns nichos do mercado
- ISAs comerciais vem e vão
- ISAs populares são complexas
- ISAs comerciais dependem de outros fatores para trazer aplicações
- ISAs comerciais populares não são projetadas para extensibilidade
- Uma ISA comercial modificada é uma nova ISA

A posição dos projetistas é que, em um sistema computacional, a ISA talvez seja a interface mais importante, e não existe razão pra que esta seja proprietária.

- eficiente energetica
- compatibilidade
- simples
- escalavel
- modular

1.1.2 História

A ISA RISC-V foi originalmente desenvolvida na Universidade da Califórnia, Berkeley, na Divisão de Ciência da computação, no departamento de Engenharia Elétrica e Ciência da Computação. Baseada na experiência com projetos passados de seus projetistas, a definição da ISA foi iniciada no verão de 2010.

Os primeiros processadores RISC-V fabricados foram escritos em Verilog e manufaturados em tecnologia de pré-produção de 28 nm FD-SOI (*Fully Depleted Silicon On Insulator*) da companhia STMicroelectronics com o nome *Raven-1*

1.1.3 RISC-V Foundation

A *RISC-V Foundation* é uma organização sem fins lucrativos, criada para direcionar futuro desenvolvimento e incentivar a utilização da ISA RISC-V. [2]

O presidente do conselho atualmente é Krste Asanovic, professor do departamento de Engenharia elétrica e ciência da computação na Univerisdade da Califórnia em Berkeley. Também co-fundado da empresa SiFive Inc., a qual incentiva do uso comercial de processadores RISC-V.

E o vice-presidente é o professor David Patterson, muito conhecido pelo livro *Computer Architecture: A Quantitative Approach*, que escreveu juntamente com John Hennessy, e suas pesquisas relacionadas a RISC, RAID, e Redes de estações de trabalho.

Outros membros incluem:

- Zvonimir Bandic, pesquisador e diretor da Western Digital Corporation.
- Charlie Hauck, CEO da Bluespec Inc.
- Frans Sijstermans, vice presidente de engenharia da NVIDIA.
- Ted Speers, chefe de arquitetura de produtos e planejamento do grupo SoC da Microsemi.
- Rob Oshana, gerente de desenvolvimento de software e negócios em segurança na NXP Semiconductors.

e também, Sue Leininger, gerente de comunidade e Rick O'Connor, director executivo. [3]

1.1.4 Open Source

O fato da arquitetura ser Open Source traz inúmeras vantagens, principalmente na parte de distribuição e compartilhamento. vantagens

vulnerabilidades

the spectre and meltdown vulnerabilities

licença

1.2 Objetivos e motivação deste projeto

RISC-V tem ganhado muito espaço tanto na academia quanto na indústria, começar um estudo desta arquitetura na universidade de Brasília e prover ferramentas para facilitar o início destes estudos.

ao iniciar o projeto havia poucas ferramentas e as ferramentas que haviam eram de difícil instalação configuração.

este projeto é um primeiro estudo da arquitetura em questão, tem o intuito de ser de fácil utilização por isso foi feito para ser utilizada na web, e ser extensível e fácil legibilidade, por isso a escolha da linguagem python, por ser mais acessível,

para casos onde se deseja performance, podem conectar módulos através de extensões para python como cython [4]

importancia do riscv

- academica
- industrial

1.3 Processadores MIPS, ARM e RISC-V

- historia entre eles - comparacoes

Capítulo 2

Fundamentação teórica

2.1 Arquitetura RISC-V: ISA

2.1.1 Diferenças

- comparações com ARM
- comparações com MIPS

2.1.2 Características

2.1.3 Como funciona

2.1.4 Instruções

Resumo de instruções

Tipos de instruções

Formatos de instruções

2.2 Montador

2.2.1 Conceito

2.2.2 Algoritmo de duas passagens

2.3 Aplicações web

2.3.1 Arquitetura

- API
- FrontEnd
- BackEnd

2.3.2 Vantagens e desvantagens

2.4 Escalabilidade (novos modulos)

Capítulo 3

Ambiente Proposto

3.1 Ambiente proposto

3.2 Arquitetura de software

3.3 Interface web

- codemirror

3.4 Simulador

-

3.5 Características

entradas e saídas

artefatos

3.6 Montador

3.7 Novos módulos

- systemC

- precisão de ciclo

- avaliação de energia

Capítulo 4

Resultados e Avaliação do Sistema

exemplos de codigos

fibonacci

primos

4.1 entradas do sistema

exemplos de codigos

imagens

4.2 saidas do sistema

dados de saida

tela de simulação

4.3 simulação

regitradores

memory map

console

montador

Capítulo 5

Conclusões

5.1 objetivos atingidos

5.2 pontos positivos e negativos

5.3 dificuldades

5.4 melhorias

- arquitetura 64 bits
- sistema de login/logout
- capacidade de salvar, importar códigos
- contador de instruções
- bitmap mapping

5.5 trabalhos futuros

Referências

- [1] Waterman, Editors Andrew e RISC V Foundation Krste Asanovic: *The risc-v instruction set manual, volume i: User-level isa, document version 2.2*. <https://riscv.org/specifications/>, May 2017. 1
- [2] Foundation, RISC V: *Sobre a risc-v foundation*. <https://riscv.org/risc-v-foundation/>, 2018. Accessed: 2018-06-20. 2
- [3] Foundation, RISC V: *Membros do conselho risc-v foundation*. <https://riscv.org/leadership/>, 2018. Accessed: 2018-06-20. 3
- [4] *Cython c-extensions for python*. <http://cython.org/>. Accessed: 2018-06-20. 3