



UFRR

**PODER EXECUTIVO
MINISTÉRIO DA EDUCAÇÃO
UNIVERSIDADE FEDERAL DE RORAIMA
DEPARTAMENTO DE CIÊNCIA DA COMPUTAÇÃO**

ARQUITETURA E ORGANIZAÇÃO DE COMPUTADORES

RELATÓRIO DO PROJETO: PROCESSADOR KRAKEN

ALUNOS:

DÉBORA BIANCA TAVEIRA DE MOURA – 2016011555

EWELLY FABIANE CUNHA DE SOUSA – 2016011439

**Janeiro de 2018
Boa Vista/Roraima**



UFRR

**PODER EXECUTIVO
MINISTÉRIO DA EDUCAÇÃO
UNIVERSIDADE FEDERAL DE RORAIMA
DEPARTAMENTO DE CIÊNCIA DA COMPUTAÇÃO**

ARQUITETURA E ORGANIZAÇÃO DE COMPUTADORES

RELATÓRIO DO PROJETO: PROCESSADOR KRAKEN

**Janeiro de 2018
Boa Vista/Roraima**

Resumo

Este relatório aborda o projeto e a implementação do processador de 8 bits **Kraken**, da disciplina de Arquitetura e organização de computadores, desenvolvido pelas alunas Débora Bianca e Ewelly Fabiane. Neste documento estarão descritos os componentes do processador e seu respectivo datapath, o formato de suas instruções, baseado no aprendizado da disciplina.

O nome do processador faz analogia ao **Kraken** da mitologia nórdica, sendo 8 de seus tentáculos referência aos 8 bits do processador. Os seus tentáculos têm a intenção de capturar alimento, assim como nosso processador captura suas oito instruções, mergulhado em um mar de binários.

Conteúdo

1	Especificação.....	7
1.1	Plataforma de desenvolvimento.....	7
1.2	Conjunto de instruções.....	7
1.3	Descrição do Hardware.....	8
1.3.1	ALU ou ULA.....	8
1.3.2	Banco de Registradores.....	8
1.3.3	PC.....	9
1.3.4	Controle.....	9
1.3.5	Memória de dados.....	9
1.3.6	Memória de Instruções.....	9
1.3.7	Somador.....	9
1.3.8	Controle da ULA.....	9
1.3.9	Multiplexador.....	9
1.3.10	Extensor de sinal 2-8.....	10
1.3.11	Extensor de sinal 2-4.....	10
1.4	Datapath.....	11
2	Simulações e Testes.....	12
3	Considerações finais.....	12

Lista de Figuras

FIGURA 1 – DATAPATH IDEAL.....	11
--------------------------------	----

Lista de Tabelas

TABELA 1 – TIPO R.....7

TABELA 2 - TIPO I.....7

TABELA 3 -TIPO J.....7

TABELA 3 -OPCODE.....8

1 Especificação

Nesta seção é apresentado o conjunto de itens para o desenvolvimento do processador **Kraken**, bem como a descrição detalhada de cada etapa da construção do processador e a definição de seus componentes.

1.1 Plataforma de desenvolvimento

Para a implementação do processador **Kraken** foi utilizado a IDE Quartus Prime 17.1 Lite, e a linguagem utilizada foi VHDL.

1.2 Conjunto de instruções

O processador Kraken possui 2 registradores: S0, S1. Assim como três formatos de instruções de 8 bits cada, Instruções do **tipo i (transferência)**, **r (Aritmético)**, **j (Lógico)**, seguem algumas considerações sobre as estruturas contidas nas instruções:

- **Opcode:** a operação básica a ser executada pelo processador, tradicionalmente chamado de código de operação;
- **rs:** o registrador contendo o primeiro operando fonte;
- **rt:** o registrador contendo o segundo operando fonte e adicionalmente para alguns tipos de instruções (ex. instruções do tipo I) é o registrador de destino;

TIPO DE INSTRUÇÃO:

- ◆ **Formato do tipo R (Aritmético):** Esse formato aborda instruções de adição, subtração,

Tipo R	Reg1	Reg2
Opcode 4 bits	rs 2 bits	rt 2 bits

- ◆ **Formato do tipo I (Transferência):** Este formato aborda instruções de Load, Store e instruções baseadas em operações aritméticas.

Tipo I	Reg1	Reg2
Opcode 4 bits	rs 2 bits	rt 2 bits

- ◆ **Formato do tipo J (Jump):** Este formato aborda instruções de desvio incondicional, como por exemplo o Jump.

Tipo J	Reg2
Opcode 4 bits	rt 4 bits

Visão geral das instruções do Processador Kraken:

O número de bits do campo **Opcode** das instruções é igual a quatro, sendo assim obtemos um total ($\text{Bit}(0 \text{ e } 1)^{\text{Numero Total de Bits do Opcode}} \therefore 2^X = X$) de 16 **Opcodes (0-3)**, entretanto, apenas 8 são distribuídos entre as instruções, assim como é apresentado na Tabela a seguir.

Tabela 1 – Tabela que mostra a lista de Opcodes utilizadas pelo processador Kraken.

Opcode	Nome	Formato	Breve Descrição	Exemplo
0000	add	R	Soma	add \$s1, \$s1,\$s2
0001	sub	R	Subtração	sub \$s1, \$s1, \$s2
0010	mult	R	Multiplicação	mult \$s1,\$s2
0011	lw	I	Load	lw \$s1,100
0100	sw	I	Store	sw \$s1,100
0101	j	J	Jump	j 100
0110	beq	I	Beq	beq \$s1, 100
0111	and	R	And	and \$s1, \$s1,\$s2

*Obs.: A instrução de beq receberá em seu registrador o valor lógico da instrução **and**, sendo ele decisivo para a realização do desvio, já que nosso processador possui apenas dois registradores a comparação de dois registradores e o salto na mesma instrução era inviável, logo a instrução foi dividida em duas.*

1.3 Descrição do Hardware

Nesta seção são descritos os componentes do hardware que compõem o processador **Kraken**, incluindo uma descrição de suas funcionalidades, valores de entrada e saída.

1.3.1 ALU ou ULA

O componente ALU (Unidade Lógica Aritmética) tem como principal objetivo efetuar as principais operações aritméticas, dentre elas: soma, subtração, divisão (considerando apenas resultados inteiros) e multiplicação. Adicionalmente o QALU efetua operações de comparação de valor como maior ou igual, menor ou igual, somente maior, menor ou igual. O componente ALU recebe como entrada três valores: **A** – dado de 8bits para operação; **B** - dado de 8bits para operação e **OP** – identificador da operação que será realizada de 4bits. O ALU também possui três saídas: **zero** – identificador de resultado (2bit) para comparações (1 se verdade e 0 caso contrário); e **result** – saída com o resultado das operações aritméticas.

1.3.2 Banco de Registradores

São registradores que recebem instruções (rs, rt e rd) ou dados.

1.3.3 PC (Program Counter)

Armazena o endereço da próxima instrução a ser executada somando 4 bits ao endereço. PC transfere o endereço somado para o MAR.

1.3.4 Controle

Manipula o caminho dos dados de acordo com as informações obtidas do campo op. Controla os componentes que emita sinais de controles.

1.3.5 Memória de dados

Usado nas operações lw e sw, onde chega os dados vindo da ALU resultante da soma de rs com o endereço de 8 bits. A escrita e lida é manipulada pelos sinais de controle EscMem e LerMem.

1.3.6 Memória de Instruções

Contem as instruções de 8 bits que serão encaminhadas aos demais componentes da MIPS: registradores, ALU.

1.3.7 Somador

Somente efetua somas.

1.3.8 Controle da ULA

Indica que tipo de operação será feita pela ALU. Recebe 2 bits de sinal de controle da ALUop. Nos casos do tipo R 10 bits indica a operação do campo func (add, sub, and), nos casos lw e sw recebe sinal 00 (add), nos casos de beq 01 (sub).

1.3.9 Multiplexador 2x1

Determina que tipo de dado ira passar por ele, ele desvia o dado para um caminho, é controlado pelos sinais de controle que recebe sinal 0 ou 1.

1.3.10 Extensor de sinal (2 para 8 bits)

Transforma o campo de endereço 2 bits para de 8 bits nos casos do tipo I (lw, sw e beq).

1.3.11 Extensor de sinal (2 para 4)

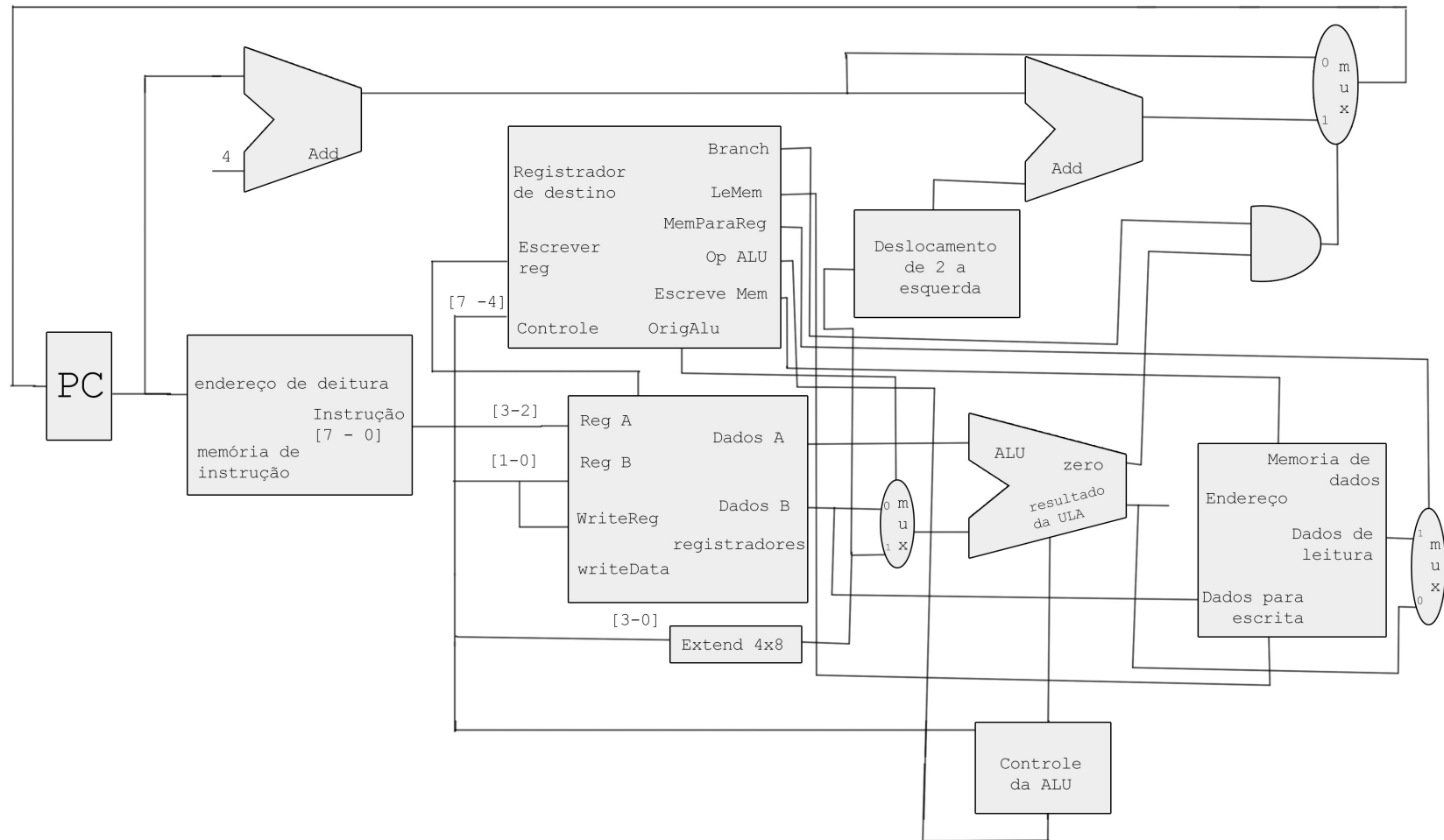
Transforma o campo de endereço 2 bits para de 4 bits nos casos do tipo I (lw, sw e beq).

1.4 Datapath

Um caminho de dados (também escrito como datapath) é um conjunto de unidades funcionais que realizam operações de processamento de dados . Datapaths, juntamente com uma unidade de controle, compõem a unidade de processamento central (CPU) de um sistema de computador. Um caminho de dados maior também pode ser criado juntando mais de um conjunto usando multiplexadores.

Atualmente, os caminhos de dados só podem ser configurados uma vez. Os pesquisadores estão tentando encontrar formas de imprimir caminhos de dados nos tecidos e torná-los reconfiguráveis. Esta ação permitiria que eles fossem configurados em tempo de execução, proporcionando uma eficiência aprimorada e economias de energia.

1.4.1 Datapath Idealizado



2 Simulações e Testes

As simulações e testes não foram realizados pois seu datapath não foi codificado. Seu teste seria realizado com calculo de um número fatorial e com a verificação de um número par ou impar, assim, utilizando todas as instruções solicitadas na criação do processador.

3 Considerações finais

Este trabalho apresentou o projeto do processador de 8 bits denominado de Kraken, que apesar de não ter sido implementado, testado e finalizado, ele contribuiu fortemente para o entendimento da disciplina de Arquitetura e Organização de Computadores e a formação acadêmica da dupla desenvolvedora, nos ensinando a base de um processador de modo geral, seu funcionamento, suas necessidades, além de demonstrar possíveis situações que podem ocorrer durante o desenvolvimento ou estudo de um processador.