

Projeto ISA e Microarquitetura

Denilson

21 de novembro de 2024

Sumário

1	Definição da ISA	2
1.1	Formatos de Instrução	2
1.2	Conjunto de Instruções	2
1.3	Conjunto de Registradores	2
2	Implementação da Microarquitetura	2
3	Código Assembly	3

1 Definição da ISA

A ISA (Instruction Set Architecture) definida neste trabalho é baseada em um conjunto de instruções de 24 bits, organizadas em três formatos principais: R, I e J. O objetivo é criar uma arquitetura eficiente e flexível, capaz de realizar operações aritméticas, lógicas, de memória e de controle.

1.1 Formatos de Instrução

Os formatos de instrução foram definidos como segue:

Formato	Tamanho	Descrição
R	24 bits	Operações aritméticas e lógicas entre registradores.
I	24 bits	Operações com valores imediatos ou acesso à memória.
J	24 bits	Instruções de salto e controle de fluxo.

1.2 Conjunto de Instruções

As instruções suportadas pela ISA incluem:

Nome	Formato	Descrição
ADD	R	Soma dois registradores.
SUB	R	Subtrai dois registradores.
ADDI	I	Soma um valor imediato ao registrador.
LW	I	Carrega uma palavra da memória para o registrador.
SW	I	Armazena uma palavra do registrador na memória.
BEQ	I	Salta se dois registradores forem iguais.
J	J	Salto incondicional.

1.3 Conjunto de Registradores

A ISA possui 16 registradores de 32 bits, com as seguintes funções:

Nome	Descrição
\$0	Constante zero (valor fixo 0).
\$4-\$7	Registradores de uso geral.
\$13	Registrador de pilha (SP).
\$14	Contador de programa (PC).

2 Implementação da Microarquitetura

A microarquitetura foi implementada no Logisim Evolution, com os seguintes componentes principais:

Componente	Descrição
PC	Contador de programa de 24 bits.
Banco de Registradores	16 registradores de 32 bits.
ULA	Unidade Lógica e Aritmética para operações R e I.
Memória de Instruções	Armazena instruções de 24 bits.
Memória de Dados	Armazena palavras de 32 bits alinhadas.

O funcionamento da microarquitetura segue os passos clássicos de busca, decodificação e execução de instruções.

3 Código Assembly

O seguinte código assembly foi desenvolvido para testar as instruções da ISA:

```
# Soma dois valores e armazena na memória
ADDI $4, $0, 10      # Carrega o valor 10 no registrador $4
ADDI $5, $0, 20      # Carrega o valor 20 no registrador $5
ADD $6, $4, $5        # Soma $4 e $5, armazena o resultado em $6
SW $6, 0($13)         # Salva o resultado na memória
```

A tradução para linguagem de máquina segue o formato definido:

Assembly	Binário
ADDI \$4, \$0, 10	000001 00000 00100 000000001010
ADD \$6, \$4, \$5	000000 00100 00101 00110 000 000
SW \$6, 0(\$13)	000010 00110 01101 000000000000