

Apêndice C - Uma linguagem de máquina simples

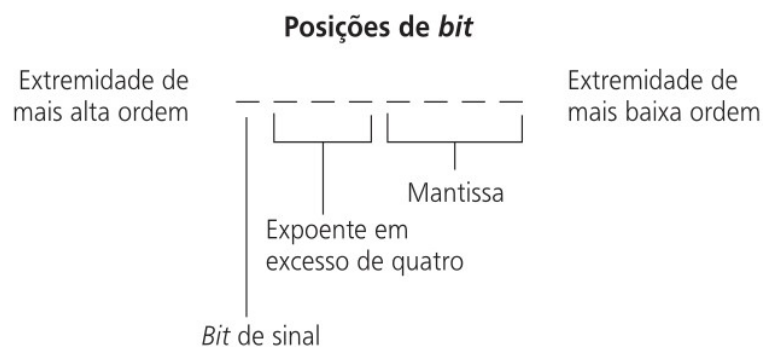
Neste apêndice, apresentamos uma linguagem de máquina simples, porém representativa. Iniciamos explicando a arquitetura da máquina.

Arquitetura da máquina

A máquina tem 16 registradores de propósito geral, numerados de 0 a F (em hexadecimal). Cada registrador tem um byte de comprimento (oito bits). Para indicar os registradores nas instruções, cada registrador é associado univocamente a um padrão de quatro bits, que representa o número do registrador.

Assim, o registrador 0 é identificado por 0000 (hexadecimal 0) e o 4, por 0100 (hexadecimal 4). Existem 256 células na memória principal da máquina. A cada uma é atribuído um único endereço, que consiste em um número inteiro na faixa de 0 a 255. Um endereço pode ser, portanto, representado por um padrão de oito bits, que varia de 00000000 até 11111111 (ou um valor hexadecimal, no intervalo de 00 a FF).

Os valores de ponto flutuante são armazenados no formato mostrado a seguir:



Linguagem de máquina

Cada instrução de máquina tem dois bytes de comprimento. Os primeiros quatro bits constituem o código de operação; os últimos 12 bits compõem o campo de operando. A tabela seguinte contém a lista das instruções, em notação hexadecimal, juntamente com uma rápida descrição de cada uma. As letras R, S e T são usadas nesses campos no lugar de dígitos hexadecimais para identificar um registrador, que varia conforme cada aplicação específica da instrução. As letras X e Y são usadas no lugar de dígitos hexadecimais nos campos variáveis que não representam um registrador.

Código de operação	Operando	Descrição
1	RXY	LOAD (carrega) o registrador R com o padrão de bits XY. Exemplo: 20A3 coloca o valor A3 no registrador 0.
2	RXY	LOAD (carrega) o registrador R com o padrão de bits XY. Exemplo: 20A3 coloca o valor A3 no registrador 0.

3	RXY	<p>STORE (armazena) o padrão de bits encontrado no registrador R na posição de memória de endereço XY.</p> <p>Exemplo: 35B1 armazena o conteúdo do registrador 5 na posição de memória de endereço B1.</p>
4	0RS	<p>MOVE (copia) o padrão de bits encontrado no registrador R para o registrador S.</p> <p>Exemplo: 40A4 copia o conteúdo do registrador A no registrador 4.</p>
5	RST	<p>ADD (soma) os padrões de bits dos registradores S e T, em complemento de dois, e coloca o resultado no registrador R.</p> <p>Exemplo: 5726 soma os valores binários dos registradores 2 e 6 e coloca esse resultado no registrador 7.</p>
6	RST	<p>ADD (soma) os padrões de bits dos registradores S e T em notação de ponto flutuante e coloca o resultado em ponto flutuante no registrador R.</p> <p>Exemplo: 634E soma os valores dos registradores 4 e E na notação em ponto flutuante e coloca o resultado calculado no registrador 3.</p>
7	RST	<p>OR (ou) executa a operação lógica OR sobre os padrões de bits dos registradores S e T e coloca o resultado no registrador R.</p> <p>Exemplo: 7CB4 coloca no registrador C o resultado da operação OR com os conteúdos dos registradores B e 4.</p>
8	RST	<p>EXCLUSIVE OR (ou exclusivo) executa a operação de OU EXCLUSIVO sobre os padrões de bits dos registradores S e T e coloca o resultado no registrador.</p> <p>Exemplo: 95F3 coloca no registrador 5 o resultado da operação EXCLUSIVE OR entre os conteúdos dos registradores F e 3.</p>
9	RST	<p>EXCLUSIVE OR (ou exclusivo) executa a operação de OU EXCLUSIVO sobre os padrões de bits dos registradores S e T e coloca o resultado no registrador.</p> <p>Exemplo: 95F3 coloca no registrador 5 o resultado da operação EXCLUSIVE OR entre os conteúdos dos registradores F e 3.</p>
A	R0X	<p>ROTATE (gira) o padrão de bits do registrador R, de X bits para a direita. Sempre coloca o bit que está na extremidade de mais baixa ordem na de mais alta ordem.</p> <p>Exemplo: A403 gira em 3 bits para a direita o conteúdo do registrador 4, de forma circular.</p>
B	RXY	<p>JUMP (salta) para a instrução localizada na posição de memória de endereço XY se o padrão de bits do registrador R coincidir com o padrão de bits do registrador 0. Caso contrário, prossegue na sequência normal de execução. (O salto é implementado copiando o valor de XY no contador de instruções durante a fase de execução.)</p> <p>Exemplo: B43C primeiro compara o conteúdo do registrador 4 com o do registrador 0. Se os dois forem iguais, o padrão 3C será colocado no contador de instruções, de modo que a próxima instrução a ser executada será a localizada naquele endereço de memória. Caso contrário, a execução do programa continuará em sua sequência normal.</p>
C	000	<p>HALT (para) a execução.</p> <p>Exemplo: C000 pára a execução do programa.</p>