

4ª parte do projeto - definição do ambiente de execução

PCS2056 - Linguagens e Compiladores

Prof. Ricardo Rocha

Data: 10/11/2016

Felipe de Paiva Miranda 7630486
Thiago Ryu Niwa Murakami 7626689

1. Introdução

O ambiente de execução é composto pela simulação de um processador muito simples. Este simulador apresenta um conjunto de elementos de armazenamento e dados, são eles: memória principal, acumulador e registradores auxiliares.

Este processador contém somente 9 instruções que são extremamente simples. Por exemplo, este processador não contém uma unidade de ponto flutuante. Entretanto, este contém as funções matemáticas básicas, sendo possível construir programas como o cálculo de fibonacci e fatorial.

2. Instruções da linguagem de saída

Operação	Mnemônico	Operando	Descrição
Jump	JP	Endereço/Rótulo de desvio	Desvio incondicional
Jump if Zero	JZ	Endereço/Rótulo de desvio	Desvio se valor no acumulador é zero
Jump if Negative	JN	Endereço/Rótulo de desvio	Desvio se valor no acumulador é negativo
Load Value	LV	Constante de 12 bits	Deposita uma constante no acumulador
Add	+	Endereço/Rótulo do operando	Soma o conteúdo do acumulador com o operando
Subtract	-	Endereço/Rótulo do subtraendo	Subtração do conteúdo do acumulador com o subtraendo
Multiply	*	Endereço/Rótulo do multiplicador	Multiplicação do conteúdo do acumulador com o multiplicador
Divide	/	Endereço/Rótulo do divisor	Divisão do conteúdo do acumulador com o divisor
Load	LD	Endereço/Rótulo do dado	Copia valor contido no endereço de memória para acumulador
Move to Memory	MM	Endereço/Rótulo de destino do dado	Copia valor do acumulador para a memória

Subroutine Call	SC	Endereço/Rótulo do subprograma	Desvio para subprograma
Return from Subroutine	RS	Endereço/Rótulo de retorno	Retorno de subprograma
Halt Machine	HM	Endereço/Rótulo do desvio	Parada
Get Data	GD	Dispositivo de E/S	Entrada
Put Data	PD	Dispositivo de E/S	Saída
Operating System	OS	Constante	Chamada de Supervisor

3. Pseudoinstruções da linguagem de saída

Pseudoinstrução	Descrição
@	Recebe um operando numérico, define o endereço da instrução seguinte, uma origem absoluta para o código a ser gerado
K	Define área preenchida por uma constante, o operando numérico tem o valor da constante de 2 bytes (em hexadecimal)
\$	Define um bloco de memória com número especificado de bytes, o operando numérico define o tamanho da área a ser reservada (em bytes)
#	Define o fim do texto fonte
&	Define uma origem relocável para o código a ser gerado, o operando é o endereço em que o próximo código se localizará (relativo à origem do código corrente)
>	Define endereço simbólico de entrada (Entry Point)
<	Define um endereço simbólico que referencia um entry-point externo

4. Características gerais

O ambiente de execução é a MVN disponibilizada, que simula o Modelo de Von Neumann como um processador simples composto por: Memória de 4096 posições e endereços de 12 bits, Acumulador e Registradores Auxiliares.

Na memória principal, são armazenadas as instruções dos programas e os seus dados. O acumulador é um registrador especial utilizado para operações aritméticas e lógicas, é utilizado também nas operações de desvio condicional.

Os registradores auxiliares são utilizados em operações intermediárias e estão descritos na tabela a seguir:

Registrador Auxiliar	Descrição
Registrador de Dados da Memória (MDR)	Utilizado para tráfego de dados entre a memória e outros elementos da MVN
Registrador de Endereço de Memória (MAR)	Contém a origem ou destino dos dados que se encontram no MDR
Registrador de Endereço de Instrução (IC)	Armazena a próxima instrução a ser executada pela máquina
Registrador de Instrução (IR)	Representa a instrução em execução, é composto de duas parcelas: o código de operação (OP) e o operando da instrução (OI).

As variáveis são acessadas diretamente pela memória. Utiliza-se complemento de 2 para determinação de sinal do dado.

Nas chamadas de subrotina, deve-se guardar o endereço de IC (que será a instrução de retorno da subrotina). Após armazenar o endereço do IC, coloca-se nesse mesmo registrador o endereço de execução da subrotina. Assim que a subrotina é executada, para retornar, basta retornar o endereço armazenado ao IC.

Referências

1. Neto J. J. Introdução à Compilação. 1987.