

Guião de Laboratório de Arquitectura de Computadores

Simulação 4.4 – Acesso à memória em 16 bits com índice variável

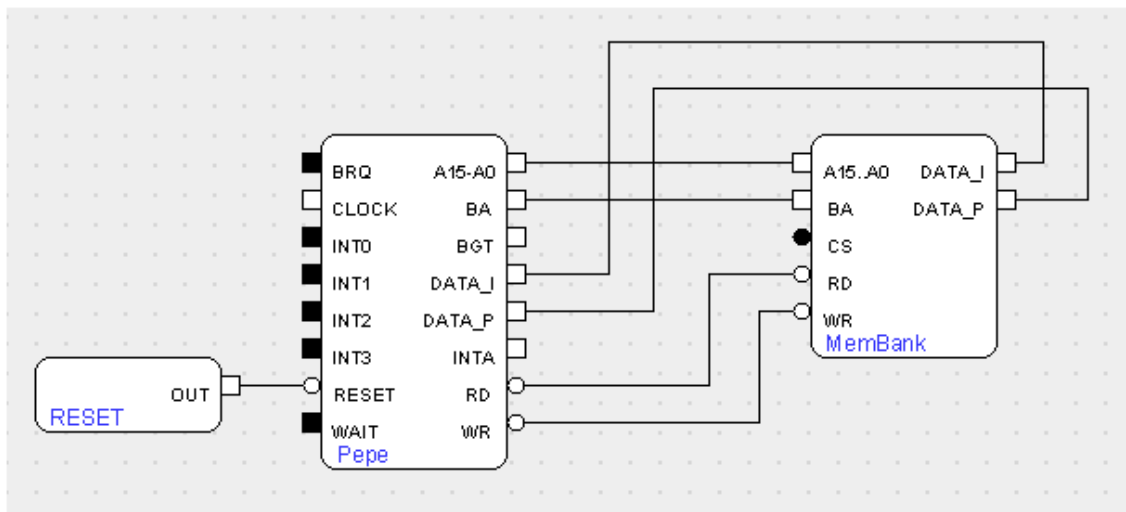
1 – Objectivos

Esta simulação ilustra o acesso à memória em 16 bits com índice variável, tendo por base o Programa 4.3 e o Programa 4.4. Os aspectos cobertos incluem os seguintes:

- Inicialização dos valores das tabelas;
- Funcionamento individual das instruções de acesso à memória em 16 bits;
- Execução passo a passo dos programas.

2 – Circuito

O ficheiro “pepe.cmod” implementa o circuito da Fig. 4.7. A simulação 4.1 contém indicações mais detalhadas sobre a sua utilização no simulador.



3 – Simulação do programa 4.3

Carregue este circuito no simulador e passe para Simulação.

Abra o painel da memória e carregue (com File->Load) o ficheiro “tabela-simples.dat”. Este ficheiro contém a inicialização da memória, no endereço 1000H (é preciso ir até lá no painel da memória), dos 8 valores da tabela que o programa 4.3 irá somar. Os valores usados são 1 a 8, um valor por cada palavra (ocupa 16 bytes). Naturalmente, pode


colocar lá manualmente outros valores ou alterar algum depois de carregar este ficheiro.
Nota – Deve carregar o ficheiro de dados antes do programa.


Abra o painel do PEPE e compile e carregue () o ficheiro “programa4-3.asm”.

Execute as instruções passo a passo e vá vendo os registos, percebendo o que o programa vai fazendo. Note que a base da tabela (R1) nunca muda, e que é o R2 que vai variando a palavra da memória acedida.

Imediatamente antes da execução da instrução MOV R0, [R1+R4], anote os valores de R0, R1, R4 e o conteúdo da memória no endereço indicado por R1+R4. Verifique que após a instrução R0 fica com esse valor.

O resultado final em R3 será 24H (soma de todos os inteiros de 1 a 8).

A alternativa é correr o programa em execução contínua, carregando no botão .

Para parar o programa e ver o valor de R3 (com o programa em execução contínua o processador não actualiza a interface), coloque primeiro um ponto de paragem na instrução “fim:” (clicando lá, o que coloca lá uma barra roxa) ou parando manualmente o programa (botão ).

4 – Simulação do programa 4.4

Carregue este circuito no simulador e passe para Simulação.

Abra o painel da memória e carregue (com File->Load) o ficheiro “tabela-simples.dat”. Este ficheiro contém a inicialização da memória, no endereço 1000H (é preciso ir até lá no painel da memória), dos 8 valores da tabela que o programa 4.3 irá somar. Os valores usados são 1 a 8, um valor por cada palavra (ocupa 16 bytes). Naturalmente, pode colocar lá manualmente outros valores ou alterar algum depois de carregar este ficheiro.
Nota – Deve carregar o ficheiro de dados antes do programa.

Abra o painel do PEPE e compile e carregue () o ficheiro “programa4-3.asm”.

Execute as instruções passo a passo e vá vendo os registos, percebendo o que o programa vai fazendo. Note que a base da tabela (R1) nunca muda, e que é o R2 que vai variando a palavra da memória acedida.

Imediatamente antes da execução da instrução MOV R0, [R1+R2], anote os valores de R0, R1, R2 e o conteúdo da memória no endereço indicado por R1+R2. Verifique que após a instrução R0 fica com esse valor.

No entanto, ao contrário do caso anterior, R2 começa em 0 e vai até 14, andando apenas pelos números pares.

De qualquer modo, o resultado final em R3 será igualmente 24H (soma de todos os inteiros de 1 a 8).

Pode igualmente correr o programa em execução contínua, procedendo como no caso anterior.