

# Guião de Laboratório de Arquitectura de Computadores

## Simulação 4.8 – Acesso à memória em 8 e 16 bits

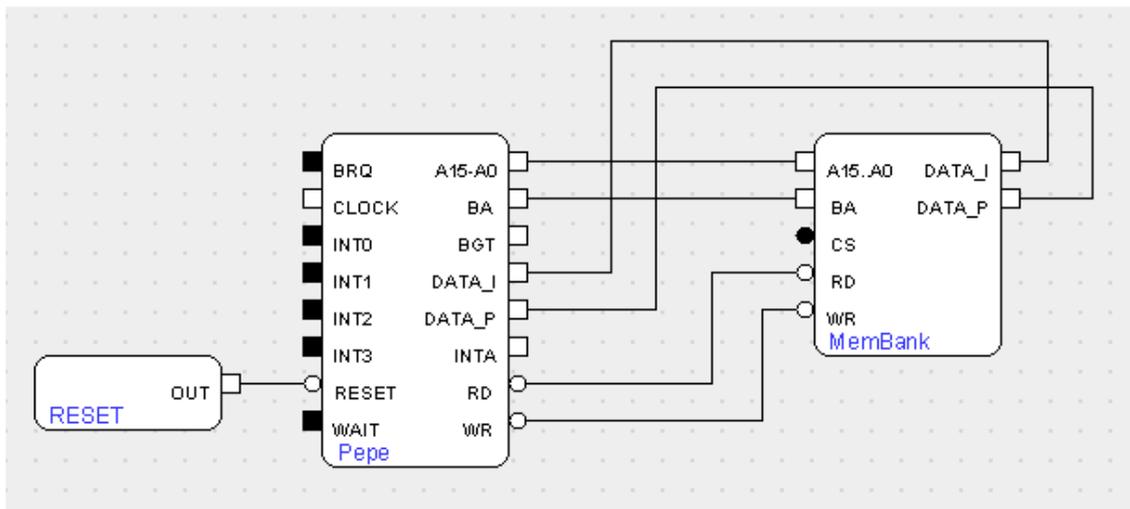
### 1 – Objectivos

Esta simulação ilustra a mistura entre acessos à memória em 8 bits e em 16 bits, tendo por base o Programa 4.8. Os aspectos cobertos incluem os seguintes:

- Inicialização da cadeia de caracteres;
- Execução passo a passo do programa;
- Verificação da evolução dos registos relevantes.

### 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 4.8

Carregue este circuito no simulador e passe para Simulação.

Abra o painel da memória e carregue (com File->Load) o ficheiro “computadores.dat”. Este ficheiro contém a string “computadores” a partir do endereço 1000H (é preciso ir até lá no painel da memória), tal como indicado na Fig. 4.11. Nota – Deve carregar o ficheiro de dados antes do programa.

Abra o painel do PEPE e compile e carregue (📁) o ficheiro “programa4-8.asm”.

Execute as instruções passo a passo e vá vendo os registos, percebendo o que o programa vai fazendo. Note que há dois apontadores para as strings, origem e destino,

em que o primeiro varia de 1 em 1 unidade porque cada byte é tratado à vez, enquanto o segundo varia de 2 em 2 unidades porque cada carácter é escrito numa palavra sozinho (em vez de dois caracteres por cada palavra).

Imediatamente antes da execução da instrução `MOVB R0, [R1]`, anote os valores de R0 e R1 e o conteúdo da memória no endereço (byte) indicado por R1. Verifique que, após a instrução, o byte de menor peso R0 fica com esse valor e o byte de maior peso fica a 0.

Imediatamente antes da execução da instrução `MOV [R2], R0`, anote os valores de R0 e R2 e o conteúdo da memória no endereço (palavra) indicado por R2. Verifique que, após a instrução, a palavra (2 bytes) de memória endereçada pelo R2 fica com 00H seguido do valor do byte de menor peso de R0.

Ao longo da execução do programa, pode ver-se o resultado na própria memória, a partir do endereço 2400H, tal como indicado na Fig. 4.13. As letras maiúsculas vão aparecendo no lado direito do painel da memória, entremeadas com bytes a 00H (devido aos acessos em 16 bits).

A alternativa é correr o programa em execução contínua, carregando no botão , mas só verá os resultados na memória quando parar o programa (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 páre manualmente o programa (botão ).