

Guião de Laboratório de Arquitectura de Computadores

Simulação 4.19 – Deslocamentos lineares lógicos

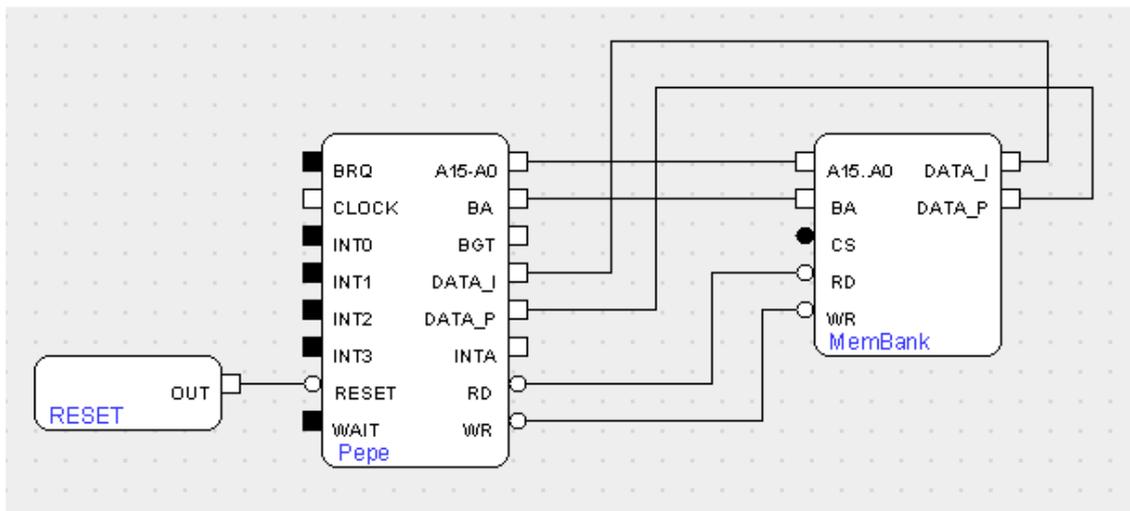
1 – Objectivos

Esta simulação permite analisar o funcionamento do Programa 4.21. Os aspectos cobertos incluem os seguintes:

- Verificação do funcionamento da instrução SHR;
- Verificação da evolução dos registos relevantes, iteração a iteração.

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 4.21

Carregue este circuito no simulador e passe para Simulação.

Abra o painel do PEPE e compile e carregue (📁) o ficheiro “programa4-21.asm”.

Execute as instruções passo a passo e vá vendo os registos, percebendo o que o programa vai fazendo.

Imediatamente antes da execução da instrução SHR R1, 1, anote o valor do registo R1 e do bit de estado C. Verifique que, após a execução desta instrução, R1 teve os seus bits deslocados de uma posição para a direita (entrando um 0 do lado esquerdo) e que o bit C ficou com o valor que o bit de menor peso de R1 tinha.

Verifique também o funcionamento da instrução ADDC R2, R3, que soma 1 condicionalmente (dependendo do valor de C) a R2 (uma vez que R3 é zero). Na prática, isto permite contar os bits a 1 que vão saindo de R1 em cada iteração.

Como resultado final, R1 fica todo a zero (depois de saírem todos os bits, a partir do valor inicial 6AC5H), e R2 fica com 8, o número de bits a 1 encontrados.

The screenshot shows a debugger window titled "User Interface" with the following assembly code and register values:

Address	Label	Instruction	PC	R0	R1	R2	R3	R4	R5	R6	R7	R8	R9	RL	SP	SSP	RE	BTE	TEMP
0000	início	MOVL R1,c5	0012	0000	0000	0008	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000
0002		MOVH R1,6a																	
0004		MOVL R2,0																	
0006		MOVL R3,0																	
0008	maisUm	ADD R1,0																	
000A		JZ fim																	
000C		SHR R1,1																	
000E		ADDC R2,R3																	
0010		JMP maisUm																	
0012	fim	JMP fim																	

TD	TV	B	A	V	C	N	Z
<input type="radio"/>	<input checked="" type="radio"/>						
R	NP	DE	IE3	IE2	IE1	IE0	IE
<input type="radio"/>							

RCN	0000	RCCD	0000
RCCI	0000	RCMV	0000
RTP	0000	RPID	0000
A6	0000	A7	0000
A8	0000	A9	0000
A10	0000	A11	0000
A12	0000	A13	0000
A14	0000	A15	0000

0008	maisUm
0012	fim
0000	início

Pode também correr o programa em execução contínua, carregando no botão .

Para parar o programa, coloque primeiro um ponto de paragem na instrução “fim:” (clicando lá, o que coloca lá uma barra roxa) ou páre manualmente o programa (botão ).

Pode também colocar pontos de paragem (fazendo clique nas instruções onde quiser parar).