

Guião de Laboratório de Arquitectura de Computadores

Simulação 4.9 – Soma e excesso: série de Fibonacci

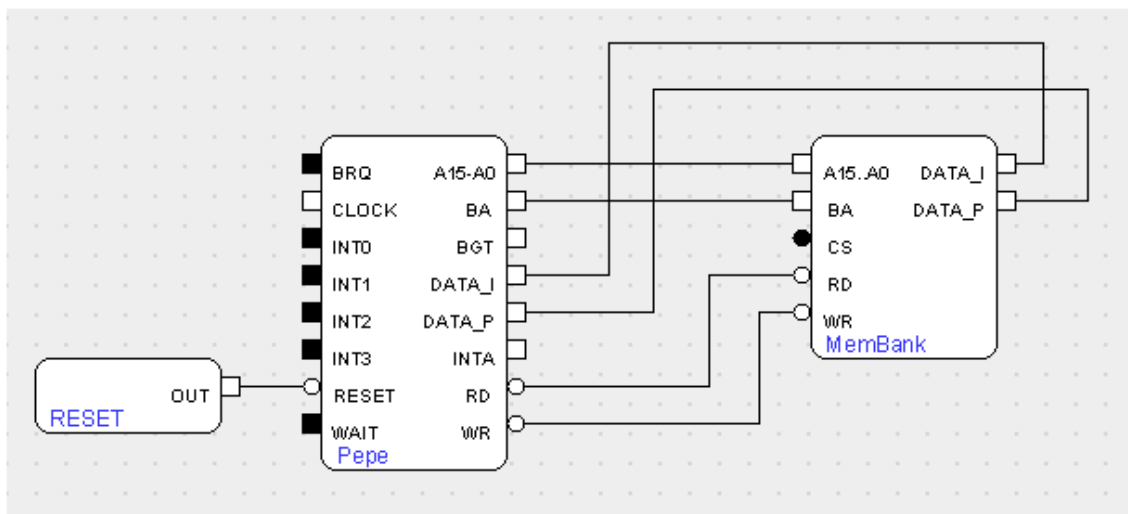
1 – Objectivos

Esta simulação ilustra o funcionamento da adição e da situação de excesso, tendo por base o Programa 4.9. Os aspectos cobertos incluem os seguintes:

- Execução passo a passo e com pontos de paragem do programa;
- Verificação da evolução dos registos relevantes e da memória, iteração a iteração;
- Verificação da condição de excesso.

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 4.9

Carregue este circuito no simulador e passe para Simulação.

Abra o painel do PEPE e compile e carregue (📁) o ficheiro “programa4-9.asm”.



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

Abra o painel da memória e vá vendo, a partir do endereço 1000H, os valores da série de Fibonacci,

frame0::C:\Users\José Delgado\Documents\simulacoes\Cap4\simulação 4.9\progr...

User Interface

0000	início	MOVL R3,0	PC	001A	R0	0000
0002		MOVH R3,10	R1	6FF1	R2	B520
0004		MOVL R2,1	R3	1000	R4	002E
0006		MOVL R1,1	R5	0000	R6	0000
0008		MOV [R3],R2	R7	0000	R8	0000
000A		MOV [R3 + 2],R2	R9	0000	R10	0000
000C		MOVL R4,4	RL	0000	SP	0000
000E	maisUm	ADD R2,R1	SSP	0000	RE	000A
0010		JV fim	BTE	0000	TEMP	0000
0012		MOV [R3 + R4],R2	TD	<input type="radio"/>	TV	<input type="radio"/>
0014		ADD R4,2	B	<input type="radio"/>	A	<input type="radio"/>
0016		SWAP R1,R2	V	<input checked="" type="radio"/>	C	<input type="radio"/>
0018		JMP maisUm	N	<input checked="" type="radio"/>	Z	<input type="radio"/>
001A	fim	JMP fim	R	<input type="radio"/>	NP	<input type="radio"/>
			DE	<input type="radio"/>	IE3	<input type="radio"/>
			IE2	<input type="radio"/>	IE1	<input type="radio"/>
			IE0	<input type="radio"/>	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
			001A	fim		
			000E	maisUm		
			0000	início		

Pode também 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 ).