

Guião de Laboratório

de

Arquitectura de Computadores (2ª edição)

Simulação 7.6 – Multiprogramação

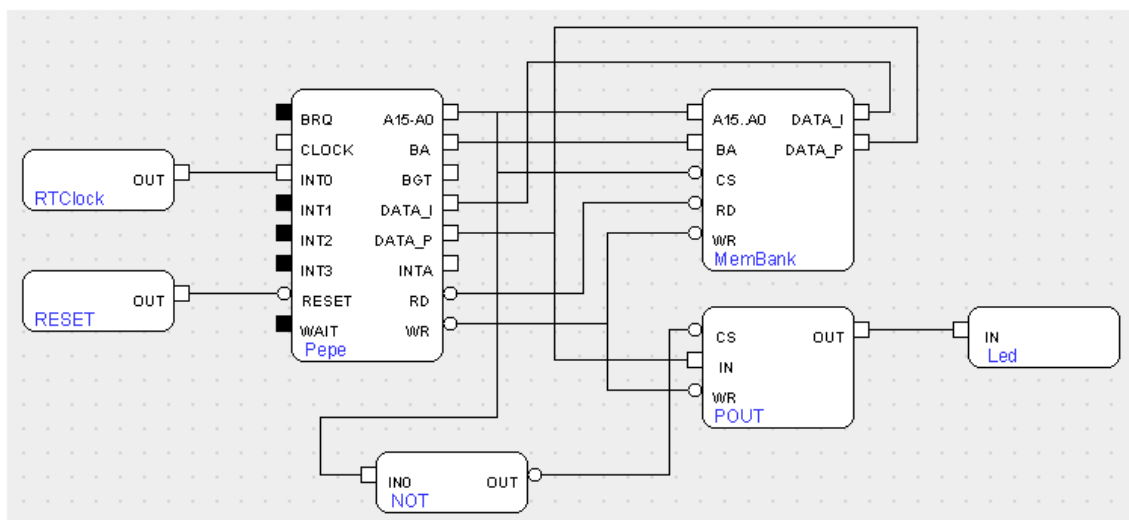
1 – Objectivos

Esta simulação toma como base o Programa 7.2 e exemplifica o funcionamento da multiprogramação no PEPE. Em particular, são focados os seguintes aspectos:

- Criação de processos;
- Verificação do funcionamento da multiprogramação e de que um processo não bloqueia outro, apesar de incluir ciclos infinitos, nem interfere com os valores dos registos dos outros processos, apesar de utilizar os mesmos registos;
- Distinção entre tempo de processo e tempo real (aumento do tempo real de processamento com o aumento do número dos processos);
- Funcionamento do mecanismo do despacho;
- Efeito do aumento exagerado da fatia de tempo de cada processo;
- Marcação das temporizações por interrupção (temporizador) em vez de ser por contagem de iterações (variante ao Programa 7.2).


2 – Circuito


O ficheiro “multiprogramacao.cmod” implementa o circuito da figura seguinte, que consiste apenas no PEPE, no banco de memória, um periférico ligado a oito leds e um relógio de tempo real, ligado à interrupção 0.



3 – Simulação do programa 7.2

Carregue este circuito no simulador e passe para Simulação.

Abra o painel do PEPE e compile e carregue () o ficheiro “programa7-2.asm”, que contém o respectivo programa em assembly.

Abra o painel dos leds e execute o programa em contínuo (botão  no painel do PEPE).

Verifique que apenas o led 0 pisca (processo 0). Se necessário, ajuste o valor da constante ESPERA (definida com EQU no início do programa) para que o led pisque a um ritmo de umas 5 a 10 vezes por segundo (mais ou menos. Rápido mas não demasiado, mas de forma que as piscadelas sejam perceptíveis). Este valor depende da rapidez do computador.

Abra o painel do relógio de tempo real (ligado à interrupção 0) e carregue em “One-shot”, o que gera apenas um ciclo de relógio.

Verifique que agora são os leds 2 e 3 que piscam alternadamente, e o led 0 ficou fixo (a 0 ou 1, depende de como o led estava na altura em que carregou no botão). O que fez foi executar a rotina Despacho uma única vez, ou seja, comutou de processo.

Vá carregando em “One-shot”, no relógio, e veja que o processo em execução muda de cada vez que carrega. Mas só um processo corre de cada vez.

Coloque agora 50 na janela “Period” do relógio e carregue em “Start”. Verifique que agora tanto o led 0 pisca como os leds 2 e 3 alternam (ou seja, os dois processos estão a executar), mas a sensivelmente metade da velocidade (os leds piscam mais lentamente do que só com um processo em execução).

O valor 50 indica que o período do ciclo de relógio é de 50 milissegundos. Isto é, em cada 50 milissegundos o processo muda. Esta mudança é tão rápida que aos nossos olhos parece que os dois processos estão a ser executados simultaneamente. Esta técnica é conhecida por multiprogramação.

No entanto, durante 50 milissegundos o PEPE só está a executar um processo, ou por outras palavras, cada processo só é executado 50 milissegundos em cada 100, o que justifica a redução da velocidade a que os leds piscam. Quantos mais processos houver, mas lento cada processo individual fica, pois o processador demora mais tempo a dar a volta e a executar novamente um dado processo.

Este facto ilustra a diferença entre tempo real e o tempo que cada processo “vê”, cuja escala é, em média N vezes mais lenta que o tempo real, sendo N o número de processos em execução.

Note-se que a mudança entre processos é feita por meio de uma interrupção, o que quer dizer que, mesmo que um dado processo tenha um ciclo de espera, como é o caso deste programa, não bloqueia a mudança.

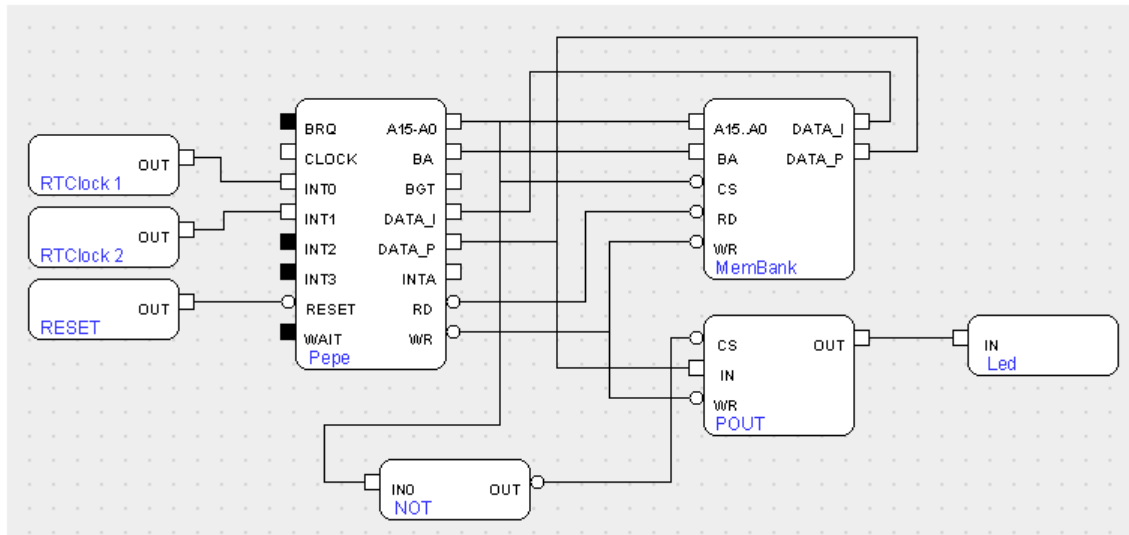
A rotina de interrupção não faz mais do que guardar todos os registos do processo (o seu contexto) numa lista, ir buscar o contexto de outro processo (incluindo o seu PC), carregando os registos, e continuar a execução no valor PC em que esse processo tinha sido interrompido.

Páre o relógio (carregando em “Stop”) e coloque 3000 no período, fazendo novamente Start. Note que agora um processo corre continuamente durante 3 segundos, depois corre o outro durante 3 segundos, voltando depois ao primeiro. A alternância de

processos continua, mas a percepção de execução simultânea já não existe. O ritmo de mudança de processos tem de ser suficientemente rápida, à escala humana, para essa percepção existir.

4 – Simulação de uma variante ao programa 7.2

Carregue agora o circuito do ficheiro “variante-multiprogramacao.cmod”, cuja diferença face ao anterior reside no segundo relógio de tempo real ligado ao pino de interrupção INT1 do PEPE.



O objectivo é manter as temporizações do processo Pisca (que faz piscar o led 0) por meio deste relógio, em vez de ser por meio de um ciclo de espera.

Compile e carregue (📁) o ficheiro “variante-programa7-2.asm” no PEPE.

Este programa distingue-se do anterior por ter uma variável em memória (“evento”), colocada a 1 sempre que a interrupção 1 ocorre e continuamente testada pelo processo Pisca. Sempre que esta variável passa a 1 (devido à rotina de interrupção Tempo), este processo troca o valor do led e coloca a variável a 0.

Execute o programa, tendo o cuidado de fazer Start nos dois relógios. Altere o valor do RTClock2 e veja que o led 0 pisca de forma bem controlada, enquanto os leds 2 e 3 alternam de forma ainda controlada pelo ciclo de espera. Não se esqueça que o período de RTClock2 é metade do período do led 0 (o led muda em cada período completo do relógio).