



Laboratório de Arquitetura e Organização de Computadores II

Protocolo Snooping

Nomes: Beatriz Souza da Silva

Marcos Junio Xavier

Professora: Daniela Cascini

Introdução

O protocolo Snooping é um protocolo usado para manter coerência de dados entre memórias cache, garantindo persistência e integridade dos dados. O diagrama de estados abaixo (Figura 1) representa a máquina de estados que codifica esse protocolo:

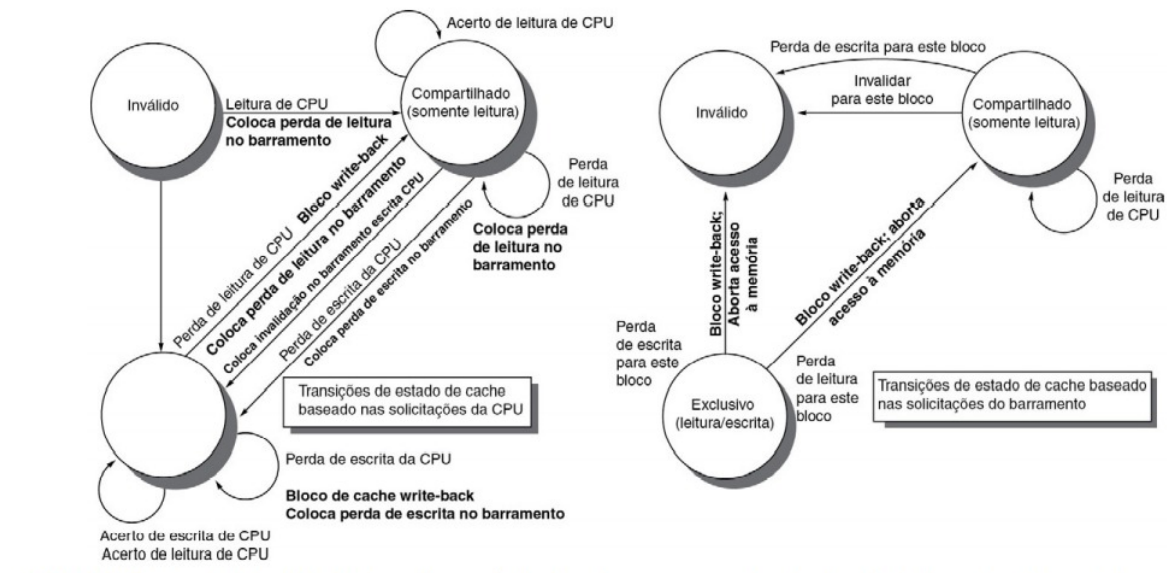


Figura 1: Máquinas de estado do protocolo Snooping

A primeira máquina de estados indica a máquina emissora, que emitirá os sinais quando algum dado for modificado na memória. A segunda mostra a máquina receptora, responsável por receber os sinais enviados pela emissora e realizar as operações necessárias para garantir a consistência de dados.

Em ambas as máquinas temos 3 estados: Inválido, compartilhado e exclusivo. O estado inválido indica quando algum dado inválido está salvo na posição de memória, ou seja, não pode ser lido. O estado compartilhado indica que os dados podem ser compartilhados entre duas ou mais caches. O estado exclusivo indica modificação dos dados, ou seja, tal valor em posição de memória é exclusivo da memória cache que a contém, ou seja, pode ser acessado somente por ela.

Simulações

As simulações do projeto consistem em verificar as transições da máquina de estados receptora/emissora. Os parâmetros utilizados para teste podem ser encontrados na Figura 2 abaixo.

```
parameter exclusive = 2'b10;
parameter shared   = 2'b01;
parameter invalid  = 2'b00;

parameter read     = 1'b0;
parameter write    = 1'b1;

parameter read_miss = 2'b11;
parameter write_miss = 2'b01;
parameter invalidate = 2'b10;

parameter hit      = 1'b1;
parameter miss     = 1'b0;
```

Figura 2: parâmetros utilizados na máquina de estados

Para simplificação dos testes, os estados da máquina de estados foram configurados como entrada. A análise da corretude dos resultados foi bastante prejudicada pelo alto numero de combinações de entrada e saída de uma máquina de estados sequencial. Para isso, criamos uma variável para simplificar a visão dos dados, chamada estado_prox_emissor, que indica para qual o próximo estado que determinada configuração fará essa máquina ir.

A variável `emissor_bus` indica a mensagem a ser escrita no barramento de dados, ou seja, a mensagem enviada quando uma leitura, escrita ou invalidação dos dados acontece. A variável `estado_wb_emissor` indica se ocorrerá write back ou não dos dados. A variável `op` indica uma leitura ou escrita, enquanto `op_estado` indica falha ou sucesso na operação.

As seções abaixo serão divididas para mostrar as simulações da máquina emissora e receptora.

- Máquina emissora**

Com o estado inválido, ao acontecer uma operação de escrita (`op=1`) com falha (`estado_op=0`), a mensagem emitida no bus é Write miss (01) e o próximo estado será Exclusivo/Modificado (10). Em uma operação de leitura (`op=0`) com falha (`estado_op=0`), o próximo estado será Compartilhado (01) e a mensagem do bus será Read Miss (11). Em uma operação de escrita com acerto (`estado_op=1`), a mensagem emitida é Write miss (01) e o próximo estado será Exclusivo/Modificado (10). Em uma operação de leitura com acerto (`estado_op=1`), a mensagem emitida é Read miss (11) e o próximo estado será Compartilhado (01). Em nenhum caso terá write-back ou acesso abortado.

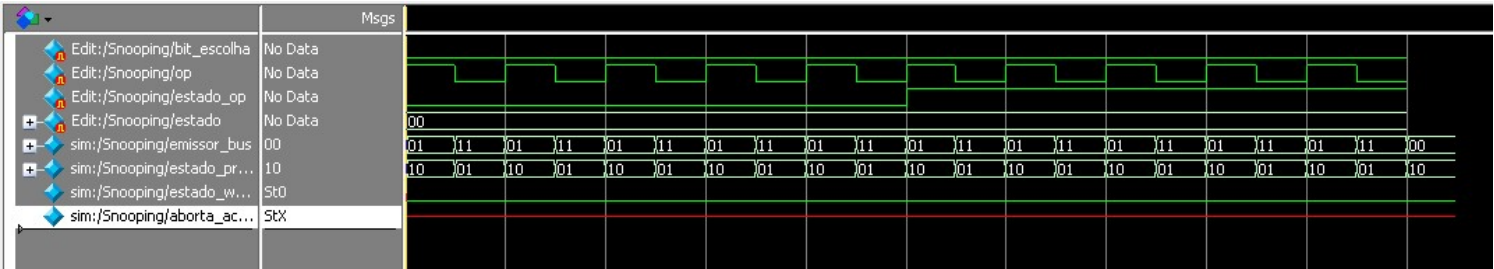


Figura 2: Estado Inválido

No estado compartilhado, quando ocorre uma operação de escrita (`op=1`) com falha (`estado_op=0`), a mensagem emitida é Write miss (01) e o próximo estado será Modificado (10). Quando ocorre uma operação de leitura (`op=0`) com falha (`estado_op=0`), a mensagem emitida é Read Miss (11) e o próximo estado será Compartilhado (01). Em uma operação de escrita com acerto (`estado_op=1`), a mensagem emitida é Invalidate (10) e o próximo estado será Modificado (10). Em uma operação de leitura com acerto (`estado_op=1`), não terá mensagem emitida (00) e o próximo estado será Compartilhado (01). Em nenhum caso terá write-back ou estado abortado.

Para melhor explicar a próxima simulação, os estados Compartilhado e Exclusivo serão simulados simultaneamente. Lembrando que a variável receptor_bus é uma variável de entrada, que guarda a mensagem que a máquina receptora está lendo.

No estado Compartilhado (01), quando não ocorre nenhuma mensagem a máquina mantém o mesmo estado. O mesmo ocorre no estado Exclusivo (10). Uma operação de Write Miss (01), se a máquina estiver no estado Compartilhado, o próximo estado será Inválido (00), e o mesmo ocorre caso a máquina estiver no estado Exclusivo mas com write-back e acesso abortado. Se a mensagem no bus for Invalidate (10), caso a máquina estiver no estado Compartilhado ela irá para o estado Inválido (00), mas caso estiver no estado Exclusivo ela se mantém em Exclusivo. Caso a mensagem for Write miss (11) e a máquina estiver no estado Compartilhado, ela continuará em Compartilhado, mas caso estiver no estado Exclusivo, irá para o estado Compartilhado e dará write-back e acesso abortado.

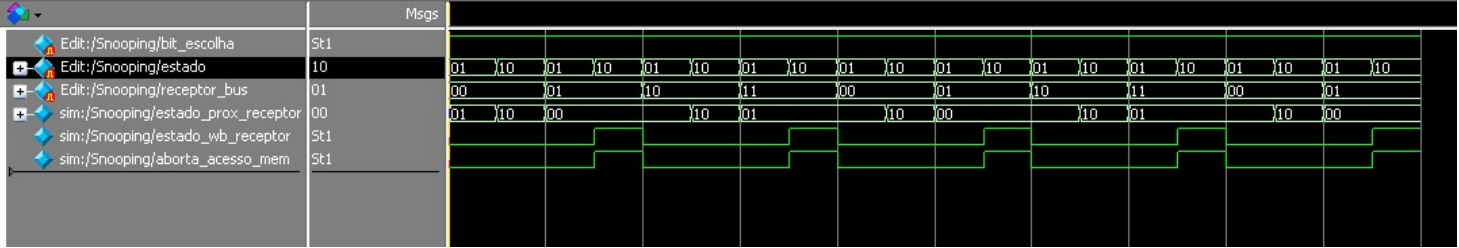


Figura 6: Estados Exclusivo e Compartilhado

Todas as simulações feitas neste relatório estão disponíveis no diretório do projeto para serem acessadas.