Unidade de proteção de memória

Universidade Federal do Ceará - Campus Quixadá

Roberto Cabral
rbcabral@ufc.br

18 de Março de 2021

Arquitetura e Organização de Computadores II

Introdução

- Alguns sistemas embarcados usam operações multitarefas e devem garantir que uma tarefa em execução não interrompa a operação de outras tarefas.
- A proteção dos recursos do sistema e outras tarefas contra acesso indesejado é chamada de **proteção**.
- Existem dois métodos para controlar o acesso aos recursos do sistema, **desprotegidos** e **protegidos**.
- Um sistema desprotegido depende apenas do software para proteger os recursos do sistema.
- Um sistema protegido depende de hardware e software para proteger os recursos do sistema.

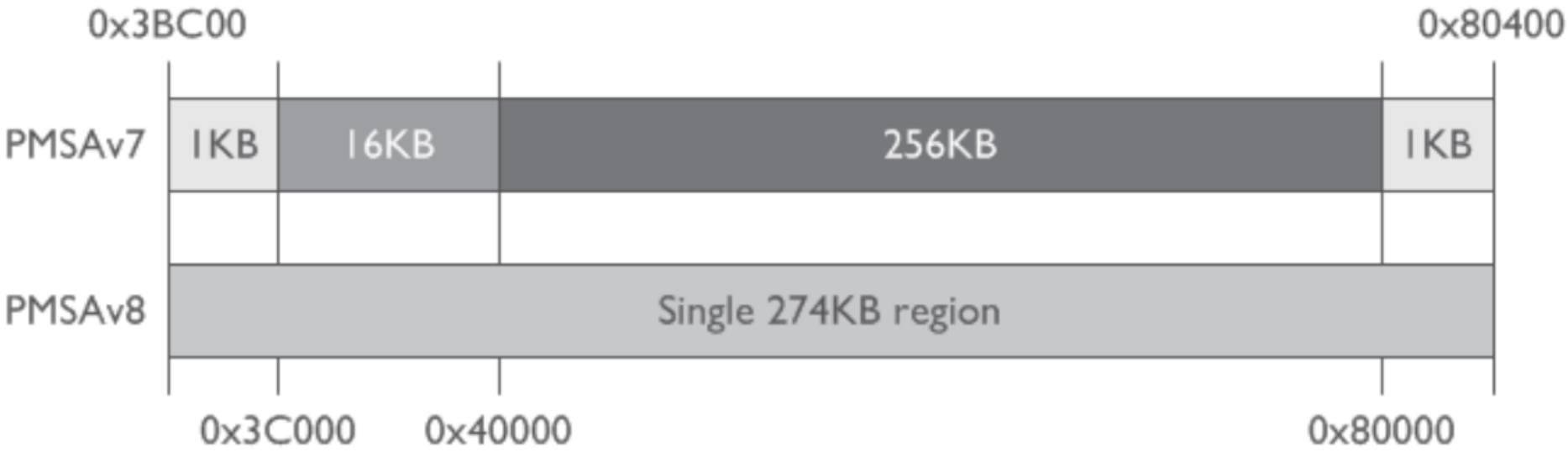
Unidade de Memória Protegida - MPU

- A MPU (Memory Protection Unit) é uma unidade programável que permite que software privilegiado defina permissões de acesso à memória.
- Ela monitora transações que podem acionar uma exceção de falha quando uma violação de acesso é detectada.
- Por exemplo:
  - Evitar que os estouros de pilha em uma tarefa corrompam a memória pertencente a outra tarefa.
  - Definir regiões de memória onde o acesso nunca é permitido por buscas de instruções, evitando que qualquer código malicioso em potencial seja executado a partir dessas regiões.
  - Proteja as regiões de RAM e SRAM contra corrupção acidental, definindo essas regiões como somente leitura.
  - Defina regiões de memória como "compartilháveis" quando vários mestres no sistema tiverem acesso a essa região.

Mudanças da MPU na arquitetura ARMv8-M

- A MPU na arquitetura do ARMv8-M possui um modelo de programação diferente para a MPU nas arquiteturas anteriores ao ARMv8-M.
- Nas arquiteturas ARMv6-M e ARMv7-M a MPU requer que uma região de memória MPU seja alinhada a um endereço que seja múltiplo do tamanho da região e que o tamanho da região seja uma potência de dois.
- Na arquitetura do ARMv8-M, o tamanho de uma região MPU pode ser de qualquer tamanho (por exemplo, 274 KB) com uma granularidade de 32 bytes.

Mudanças da MPU na arquitetura ARMv8-M



Características da MPU

- O MPU ARMv8-M suporta um número configurável de regiões programáveis, com uma implementação típica suportando entre zero e oito regiões por estado de segurança.
  - O menor tamanho que pode ser programado para uma região MPU é de 32 bytes.
  - O tamanho máximo de qualquer região MPU é de 4 GB, mas deve ser múltiplo de 32 bytes.
  - Todas as regiões devem começar em um endereço alinhado de 32 bytes.
  - As regiões têm permissões de acesso de leitura/gravação independentes para código privilegiado e não privilegiado.
  - O atributo eXecure Never (XN) permite a separação de regiões de código e dados.

Tipos de Memória

- Na arquitetura do ARMv8-M, os tipos de memória são divididos em **Memória normal** e **Memória do dispositivo**.
- Se a arquitetura ARMv8-M implementa extensão de segurança, o espaço da memória será particionado nas regiões de memória segura e não segura.

Memória normal

- O tipo de memória normal pode ser usado para regiões MPU usadas para acessar instruções gerais ou memória de dados.
- A memória normal permite que o processador execute algumas otimizações de acesso à memória, como reordenação de acesso.
- A memória normal também permite que a memória seja armazenada em cache e é adequada para armazenar código executável.
- A memória normal pode ter vários atributos que podem ser aplicados a ela.
  - Cacheability
  - Shareability
  - eXecute Never

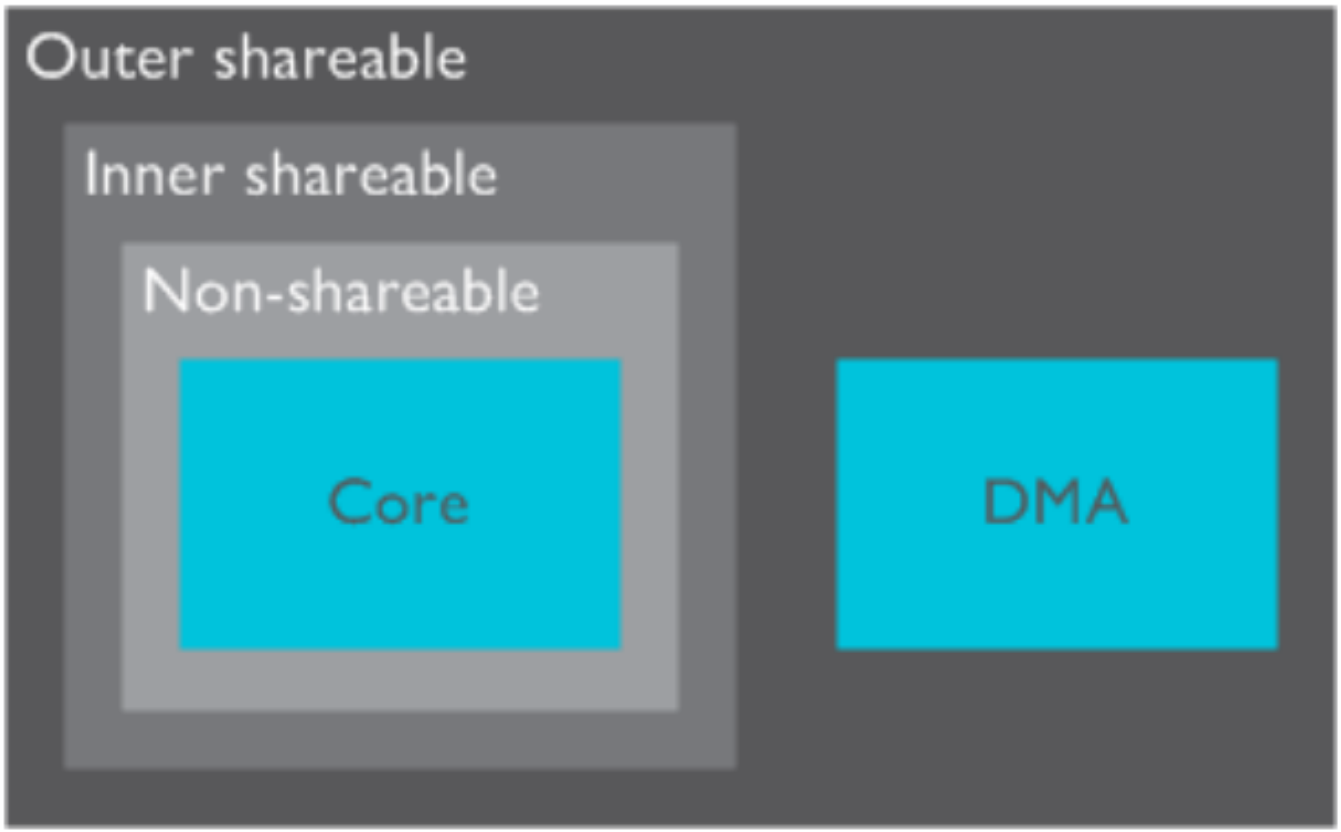
Memória normal - Cacheability

- A memória normal pode ser armazenável em cache ou não.
- Pode ser divida em: *cache policy*, *allocation*, *Transient hint*.
- A arquitetura suporta dois níveis de atributos de cache.
- São os atributos de cache interno e externo.
- A configuração de uma região MPU com um tipo de memória armazenável em cache não significa que os dados devem ser armazenados em cache, mas apenas indicam ao hardware que eles podem ser armazenados em cache.
- Se uma região for definida como software armazenável em cache, será responsável por executar qualquer operação de manutenção de cache necessária.

Memória normal - Shareability

- Muitos sistemas têm vários mestres de barramento, múltiplos processadores ou uma mistura de processadores e outros mestres, como os mecanismos de acesso direto à memória (DMA).
- O atributo *shareability* permite que o software indique ao hardware quais desses dispositivos devem poder ver as atualizações em uma área específica da memória.
- Para gerenciar a *shareability*, a arquitetura do ARM agrupa todos os mestres em um dos três domínios:
  - Memória não compartilhável.
  - Memória interna compartilhável.
  - Memória externa compartilhável.

Memória normal - Shareability



Memória normal - Shareability

- Definir a capacidade de compartilhamento de uma região de memória impõe alguns requisitos funcionais ao hardware, mas não restringe como o hardware implementa essa funcionalidade.
- O requisito *outer shareable* (OSH) é que todos os mestres no domínio compartilhável externo possam ver os efeitos de qualquer atualização de memória:
  - Em um sistema sem caches e apenas no nível da RAM, qualquer mestre pode ver qualquer atualização de memória.
  - Em um sistema com caches, nem todos os mestres podem acessar todos os caches, e o sistema pode empregar coerência de cache no hardware para tornar as atualizações visíveis ou tratar qualquer memória compartilhável como não armazenável em cache, tornando as atualizações visíveis.

Memória não compartilhável

- Representa a memória acessível apenas por um único processador.
- Os acessos à memória nunca precisam ser sincronizados com outros processadores.
- Somente o próprio processador deve ver as informações, embora possam ser tornadas visíveis para outros agentes

Memória interna compartilhável

- Representa um domínio que pode ser compartilhado por vários mestres, mas não necessariamente todos os agentes no sistema.
- Um sistema pode ter vários domínios compartilháveis internos.
- Uma operação que afeta um domínio compartilhável interno não afeta outros domínios compartilháveis internos no sistema.
- Todos os agentes desse domínio podem ver a memória.

Memória externa compartilhável

- É compartilhado por vários agentes e pode consistir em um ou mais domínios compartilháveis internos.
- Uma operação que afeta um domínio compartilhável externo também afeta implicitamente todos os domínios compartilháveis internos dentro dele.
- No entanto, não se comporta como uma operação compartilhável interna.

Unidade de proteção de memória

Universidade Federal do Ceará - Campus Quixadá

Roberto Cabral
rbcabral@ufc.br

18 de Março de 2021

Arquitetura e Organização de Computadores II