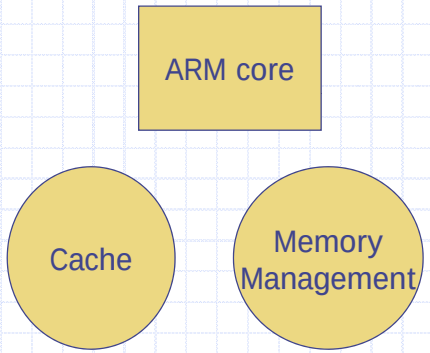
ARM Memory Protection Units

ARM Processor Organization



MPU & MMU

- MPU == hardware protection over software-designated regions
- MMU == hardware protection + virtual memory support
- E.g., ARM920T vs. ARM940T
  MMU MPU

Why Protection Necessary?

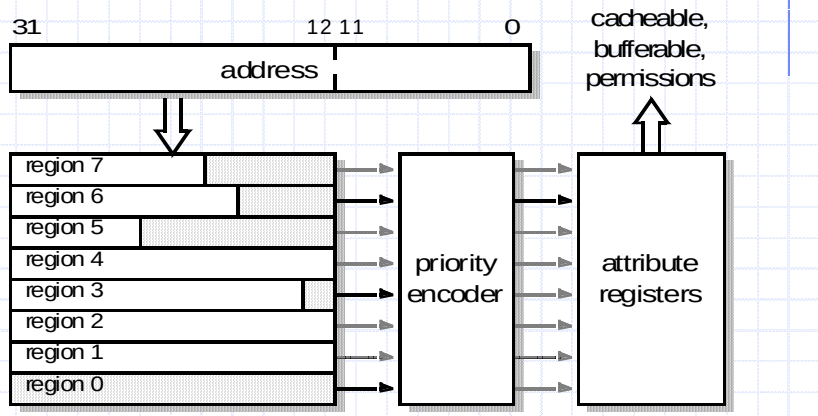
Protected Regions

- A basic management unit of system protection
- 8 regions: region 0 – region 7
- Attributes:
  - Starting address
  - Length (4KB ~ 4GB, power of two size only)
  - Access rights, cache/WB policies, cache write policy
    - Read-write, read-only, no access
    - Based on the current processor mode, [privileged] [user]
    - Memory access violations → abort exceptions

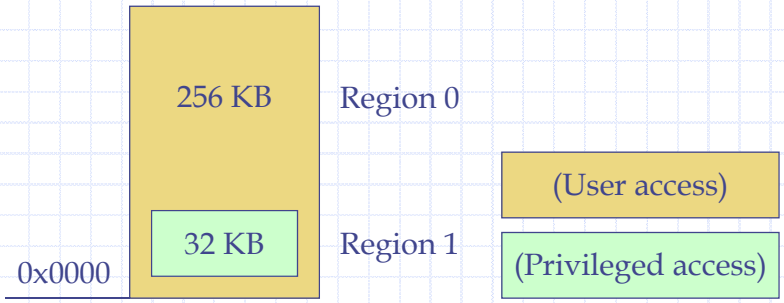
Region Rules

- Can overlap other regions
- Each region has its own priority
  - The higher the region number, the higher its priority
  - For overlapped areas, the attributes of the highest priority are applied
- Starting address must be aligned to its size
- Accessing an area outside of a defined region results in an abort.

MPU Organization



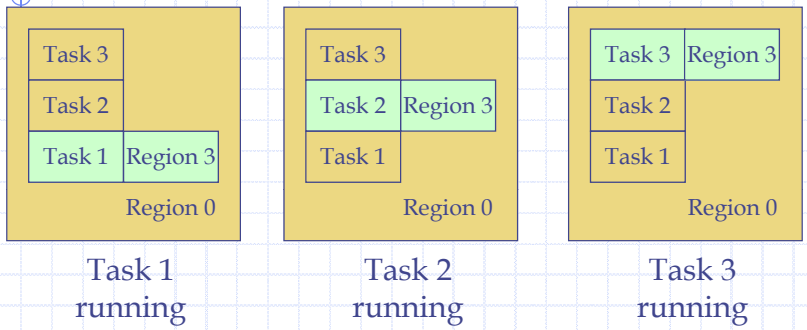
Ex: Overlapping Regions



Background Regions

- A low-priority region used to assign the common attributes to a large memory area.
- Other higher regions can overrule when necessary.

Ex: Background Regions



Task 1 running

Task 2 running

Task 3 running

Region 0: Background Region
Region 3: Active Task

(User access)

(Privileged access)

MPU Initialization Steps

1. Define regions. (CP15:c6)
2. Set access permission for each region. (CP15:c5)
3. Set the cache/WB attributes for each region. (CP15:c2 (cache) c3 (WB))
4. Enable caches/MPU. (CP15:c1)

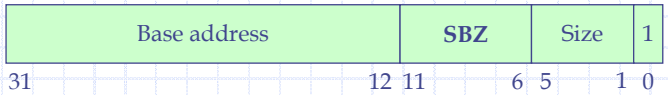
- Standard AP

Supervisor	User
No Access	No Access
R/W	No Access
R/W	Read Only
R/W	R/W

- Inst $: Not Cached/Cached
- Data $: Not Cached & Not Buffered,
  Not Cached & Buffered
  Cached (WT)
  Cached (WB)

Ex: Region size and location

- One secondary register for each region:
  - CP15:c6:c0:0 for region 0



Size of region: $2^{(Size+1)}$

SBZ == Should Be Zero