Summary documentation of hardware implementation issues

The following are some of the implementation problems encountered when implementing the Svnapot extension on NutShell. We have extended the extension for the general situation, hoping to avoid the pitfalls for the future

1.Compatibility problem between Napot PTE and 4KB PTE

1.1 problem definition

The 64KB base page specified by the Svnapot extension is extended on top of a 4KB base page, that is, it is a special case of a 4KB base page.
The same is true for implementation.
NutShell's DTLB was initially a group connected structure, in which 4KB, 2MB and 1GB of PTE were stored. No matter 64KB of PTE was added, there was a potential problem, that is, in the case of group connected, it was necessary to first find the corresponding group according to the virtual address, and then determine the hit way in this group.
The problem with finding a corresponding group of virtual addresses is that it usually cuts off a part of the virtual address as an index to select a specific group. This will cause different virtual addresses belonging to the same 2MB or 1GB page to find different groups due to different indexes, when they should find the same group.
In this case, although the TLB cached the PTE, it could not hit because it was indexed to different groups, resulting in two identical caches in the TLB, and the potential problem of multiple hits and inconsistencies exists.

1.2 solutions

- The easiest way to do this is to change the group concatenation to fully concatenation, removing the index step and treating 4KB, 2MB, 1GB and 64KB PTEs the same.
- Because the number of fully connected items can not be too many, eventually you have to go back to group connected.
  In general, today's high performance processors use a group connected structure, with different ways of organization.
  For example, some implementations specially set a group connected structure for 4KB PTE, and similarly set a structure for 2MB and 1GB respectively. Different indexing strategies are used to judge hits. The three PTEs are stored separately and different indexing strategies are used to avoid multiple hits and inconsistencies.
  After implementing the Svnapot extension, a 64KB PTE was added, which needs to be compatible with a 4KB PTE, with the following two strategies:
  - The 64KB is stored together with the 4KB PTE in a group connected structure, but an indexing strategy is used to ensure that no conflicts occur.
  - The 64KB PTE is stored separately in a group connected structure, but this structure is wasted if the software does not support the Svnapot extension.

2.Hardware management A/D bit problems

2.1 problem definition

NutShell adopts the strategy of hardware backfilling A/D bits, that is, if the TLB hits and finds that the A/D bits are not set correctly, the hardware will automatically place the A/D position of the PTE in memory, without the involvement of software.
This strategy is always correct in the case of single core, but in the case of multi-core it may need to be combined with the zsa extension to ensure correctness.
For the sake of area, NutShell does not store a complete PTE(**or even a complete PTE.ppn**) in TLB, because it takes into account that there are some reserved bits in PTE that are unnecessary to store. Although there is no complete PTE stored in TLB, a complete PTE must be assembled when A/D bits need to be backfilled. This PTE can only be A/D bit different from the in-memory PTE.
NutShell combines PTE by default with all the reserved bits being 0, but in the case of Svnapot expansion, not all the reserved bits are 0. If we follow the previous pattern, it is possible to change an Svnapot extended PTE in memory to a normal PTE(changing N bits to 0) due to A/D bit backfilling.
The next time you get this PTE from memory, it's not going to be right.

2.2 solutions

- When backfilling the A/D bits, check if the TLB cache PTE is a Svnapot extended PTE. If so, keep the N bits in the reserved bits as 1. If not, keep the reserved bits as 0 as before.

3.Summary

- No matter what kind of TLB structure to implement the Svnapot extension needs to pay attention to the group connected problem due to the index policy.
- Since most TLBs may not store the full PTE, if the hardware is managing the A/D bits, the implementation of Svnapot extension needs to be careful to backfill the Svnapot PTE with N bits being 1.