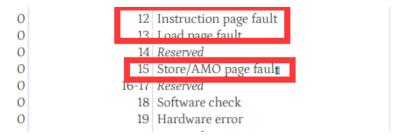
## CH4

- 1. 请列举 SV39 页表页表项的组成,描述其中的标志位有何作用?
  - [63:54]为保留项, [53:10]为44位物理页号, 最低的8位[7:0]为标志位。
    - V(Valid): 仅当位 V 为 1 时, 页表项才是合法的;
    - R(Read)/W(Write)/X(eXecute):分别控制索引到这个页表项的对应虚拟页面是否允许读/写/执行;
    - U(User): 控制索引到这个页表项的对应虚拟页面是否在 CPU 处于 U 特权级的情况下是 否被允许访问;
    - A(Accessed): 处理器记录自从页表项上的这一位被清零之后,页表项的对应虚拟页面是 否被访问过;
    - D(Dirty): 处理器记录自从页表项上的这一位被清零之后,页表项的对应虚拟页面是否 被修改过。
- 2. 请问哪些异常可能是缺页导致的? 发生缺页时, 描述相关重要寄存器的值



## CSR寄存器:

- scause: 中断/异常发生时, CSR 寄存器 scause 中会记录其信息, Interrupt 位记录是中断还是异常, Exception Code 记录中断/异常的种类。
- sstatus: 记录处理器当前状态,其中 SPP 段记录当前特权等级。
- stvec: 记录处理 trap 的入口地址,现有两种模式 Direct 和 Vectored。
- sscratch: 其中的值是指向hart相关的S态上下文的指针,比如内核栈的指针。
- sepc: trap 发生时会将当前指令的下一条指令地址写入其中,用于 trap 处理完成后返回。
- stval: trap 发生进入S态时会将异常信息写入,用于帮助处理 trap,其中会保存导致缺页异常的虚拟地址。
- 3. 处理 10G 连续的内存页面,对应的 SV39 页表大致占用多少内存(估算数量级即可)?

## 20M左右

4. 此时页面失效如何表现在页表项(PTE)上?

V(Valid): 位 V为0

5. 在单页表情况下,如何更换页表? 更新寄存器,刷新tlb

6. 单页表情况下,如何控制用户态无法访问内核页面? 位 U为0

7. 单页表有何优势?仅在进程切换时更换页表, TLB..

8. 双页表实现下,何时需要更换页表?

进程切换,用户态内核态切换

单页表: 进程切换