第五讲物理内存管理：非连续内存分配

第 6 节 RISC-V 页机制

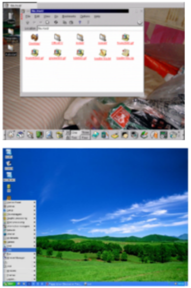
向勇、陈渝

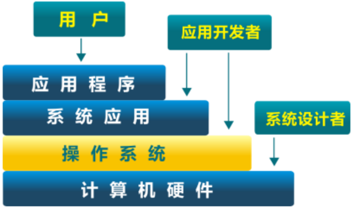
清华大学计算机系

xyong,yuchen@tsinghua.edu.cn

2020 年 2 月 28 日

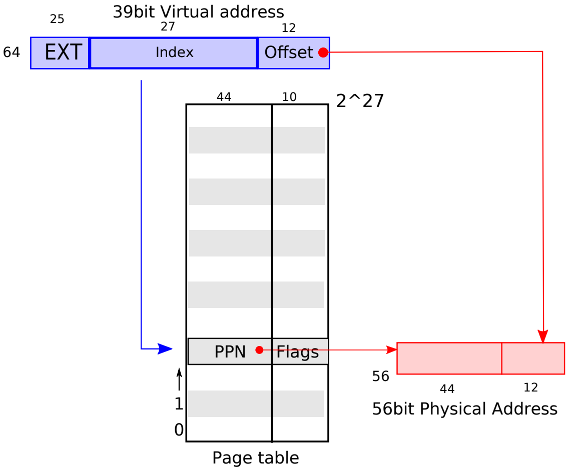
回顾





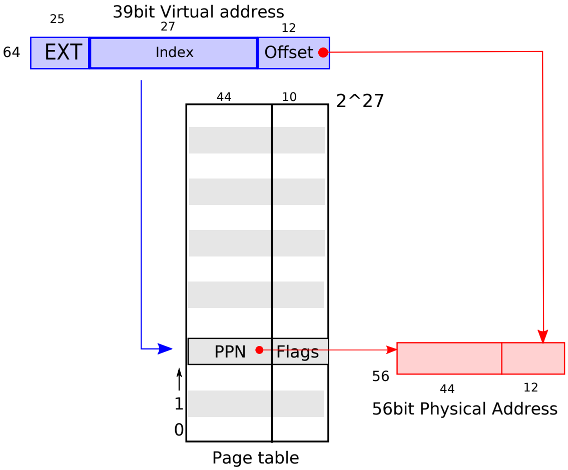
- 通过页表来实现隔离与共享
  - 运行的应用程序之间的隔离与共享
  - 应用与内核之间的隔离与共享
  - 便于非连续内存管理

RISC-V Supervisor 特权模式



- RISC-V 对页表的硬件支持
  - 页表基址

RISC-V Supervisor 特权模式

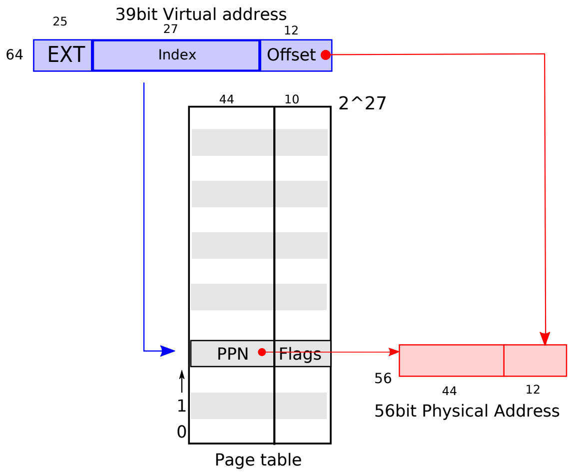


- RISC-V 对页表的硬件支持
  - 页表基址：satp
  - Supervisor Address Translation and Protection (satp) Register

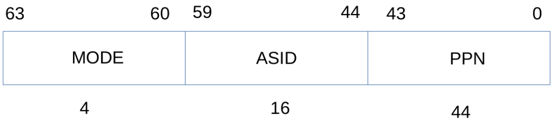
31	30	22	21	0
MODE	ASID		PPN	
1	9		22	

RV32 Supervisor address translation and protection register satp

RISC-V Supervisor 特权模式

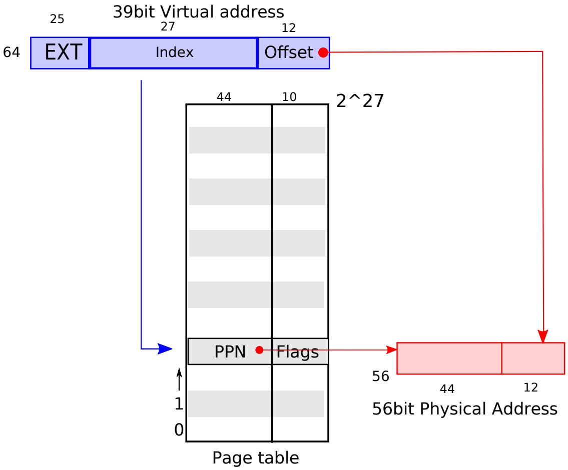


- RISC-V 对页表的硬件支持
  - 页表基址：satp
  - Supervisor Address Translation and Protection (satp) Register



RV64 Supervisor address translation and protection register satp

RISC-V Supervisor 特权模式

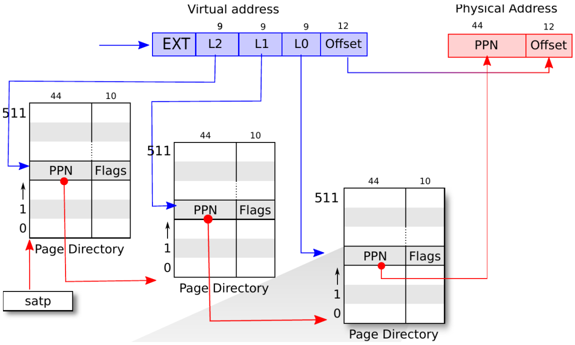


- RISC-V 对页表的硬件支持
  - 页表基址：satp
  - Supervisor Address Translation and Protection (satp) Register

RV32		
Value	Name	Description
0	Bare	No translation or protection.
1	Sv32	Page-based 32-bit virtual addressing.

RV64		
Value	Name	Description
0	Bare	No translation or protection.
1–7	—	*Reserved*
8	Sv39	Page-based 39-bit virtual addressing.
9	Sv48	Page-based 48-bit virtual addressing.
10	*Sv57*	*Reserved for page-based 57-bit virtual addressing.*
11	*Sv64*	*Reserved for page-based 64-bit virtual addressing.*
12–15	—	*Reserved*

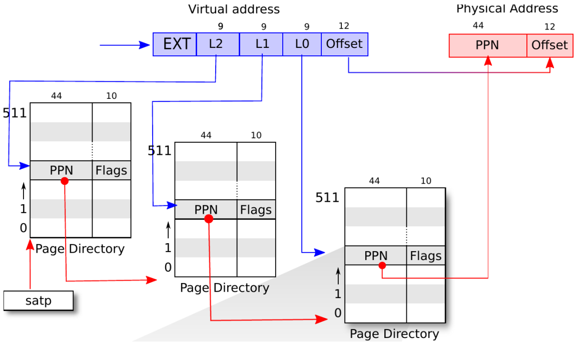
RISC-V Supervisor 特权模式



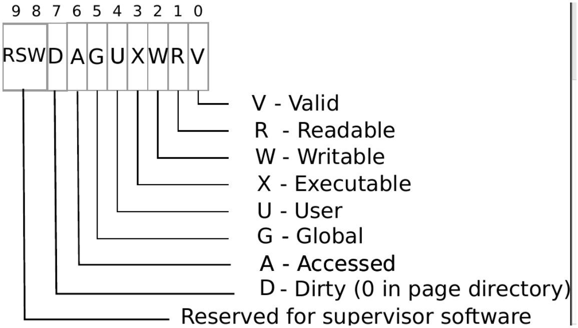
- RISC-V 对页表的硬件支持
  - 地址保护
  - 页表项（page table entry）

X	W	R	Meaning
0	0	0	Pointer to next level of page table.
0	0	1	Read-only page.
0	1	0	*Reserved for future use.*
0	1	1	Read-write page.
1	0	0	Execute-only page.
1	0	1	Read-execute page.
1	1	0	*Reserved for future use.*
1	1	1	Read-write-execute page.

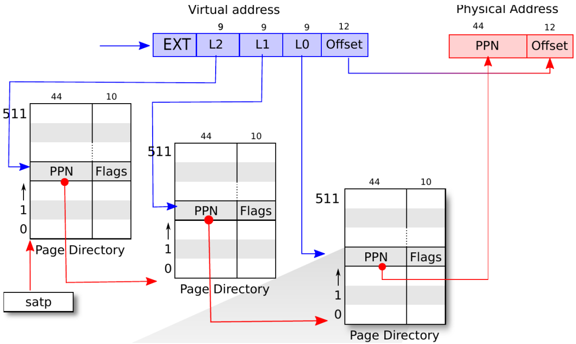
RISC-V Supervisor 特权模式



- RISC-V 对页表的硬件支持
  - 地址保护
  - 页表项（page table entry）



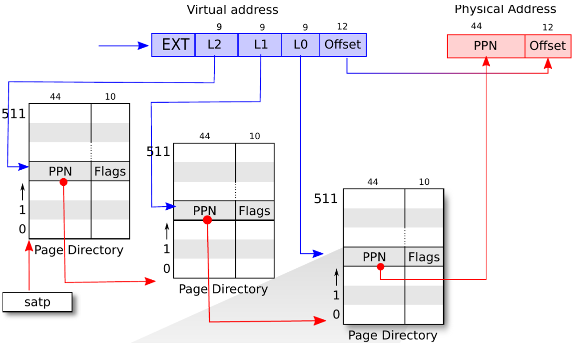
RISC-V Supervisor 特权模式



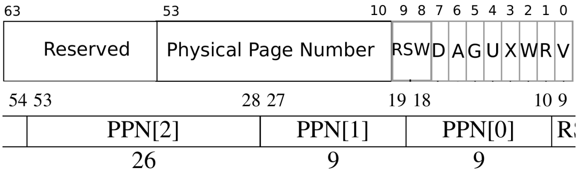
- RISC-V 对页表的硬件支持
  - 地址保护
  - 页表项（page table entry）

63	53 10	9 8	7	6	5	4	3	2	1	0
Reserved	Physical Page Number	RSW	D	A	G	U	X	W	R	V

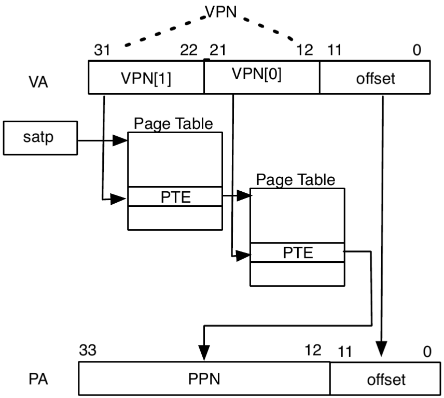
RISC-V Supervisor 特权模式



- RISC-V 对页表的硬件支持
  - 地址保护
  - 页表项（page table entry）

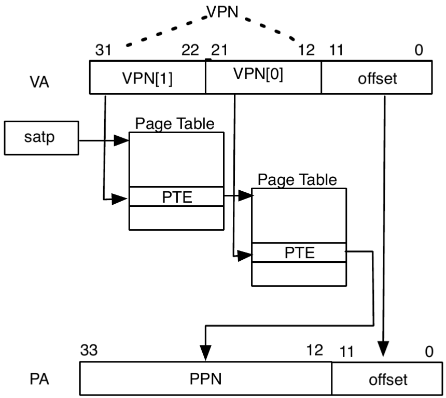


RISC-V 地址转换



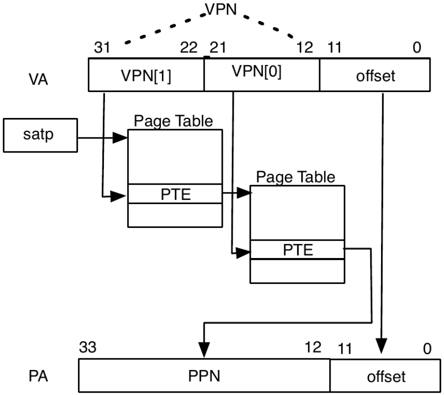
- RV32
  - 当在 satp 寄存器中启用了分页时，虚拟地址映射启动。
  - 1. satp.PPN 给出一级页表基址,VA[31:22] 给出一级页号,CPU 会读取位于地址 (satp. PPN $\times$ 4096 $+$ VA[31: 22] $\times$ 4) 的页表项。

RISC-V 地址转换



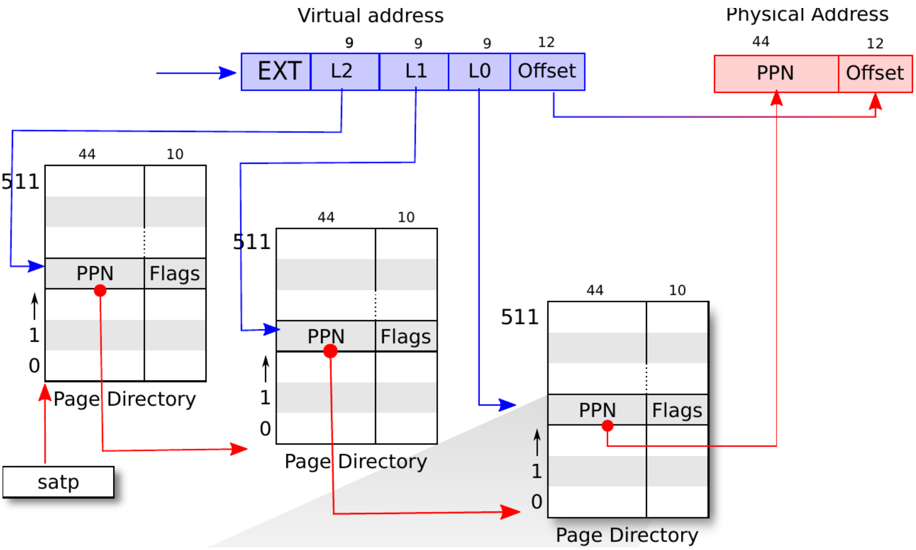
- RV32
  - 当在 satp 寄存器中启用了分页时，虚拟地址映射启动。
  - 1. satp.PPN 给出一级页表基址,VA[31:22] 给出一级页号,CPU 会读取位于地址 (satp.PPN $\times$ 4096 + VA[31: 22] $\times$ 4) 的页表项。
  - 2. 该 PTE 包含二级页表的基址,VA[21:12] 给出二级页号,CPU 读取位于地址 (PTE.PPN $\times$ 4096 + VA[21: 12] $\times$ 4) 的叶节点页表项。

RISC-V 地址转换



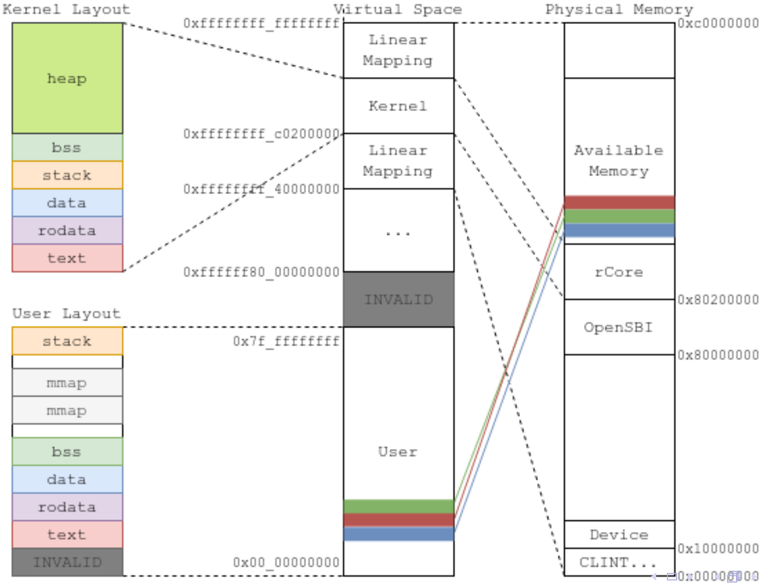
- RV32
  - 当在 satp 寄存器中启用了分页时，虚拟地址映射启动。
  - 1. satp.PPN 给出一级页表基址,VA[31:22] 给出一级页号,CPU 会读取位于地址 (satp. PPN × 4096 + VA[31: 22] × 4) 的页表项。
  - 2. 该 PTE 包含二级页表的基址,VA[21:12] 给出二级页号,CPU 读取位于地址 (PTE. PPN × 4096 + VA[21: 12] × 4) 的叶节点页表项。
  - 3. 叶节点页表项的 PPN 字段和页内偏移 (原始虚址的最低 12 个有效位) 组成了最终结果：物理地址就是 (LeafPTE. PPN × 4096 + VA[11: 0])

OS 配置页表的流程



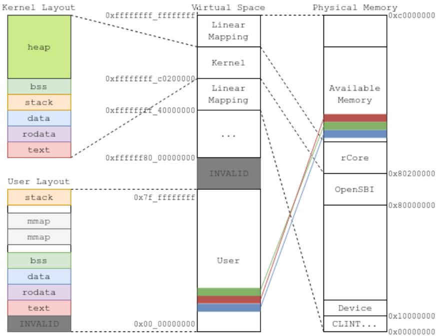
Virtual address
9
9
9
12
EXT
L2
L1
L0
Offset
Physical Address
44
12
PPN
Offset
44
10
511
PPN
Flags
1
0
Page Directory
satp
44
10
511
PPN
Flags
1
0
Page Directory
44
10
511
PPN
Flags
1
0
Page Directory

OS 配置页表的流程



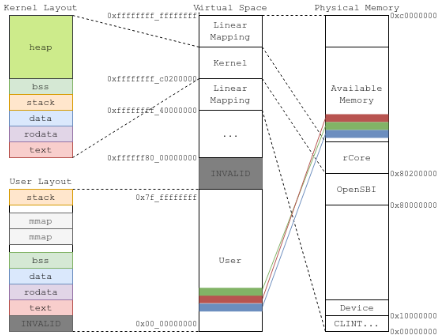
Kernel Layout
heap
bss
stack
data
rodata
text
User Layout
stack
mmap
mmap
bss
data
rodata
text
INVALID
Virtual Space
0xffffffff_ffffffff
Linear Mapping
Kernel
0xffffffff_c0200000
Linear Mapping
0xffffffff_40000000
...
0xffffff80_00000000
INVALID
0x7f_ffffffff
User
0x00_00000000
Physical Memory
0xc0000000
Available Memory
rCore
0x80200000
OpenSBI
0x80000000
Device
0x10000000
CLINT...
0x00000000

OS 配置页表的流程



- 为页表分配物理内存
- 确定映射的物理空间与虚拟空间
- 创建页表
- 设置 sapt，使能页表

OS 配置页表的流程



- 为页表分配物理内存
- 确定映射的物理空间与虚拟空间
- 创建页表
- 设置 sapt，使能页表