Privileged ISAs

Reminder: OS as resource manager

OS manages how processes access resources

- CPU time
- Physical memory
- Disk
- Network
- I/O

What help does the OS need from the hardware (CPU)?

Privileged ISAs

Did you know we were only working with Volume 1 of the RISC-V spec?

Volume 2 describes the RISC-V privileged architecture

Allows CPU to be executing in one of three modes (U, S, M)

(x86 and arm have similar notion of privilege modes/levels)

CSRs (control status registers) are SPRs that track and control what the CPU does in each mode

Some registers only readable/writable in some modes

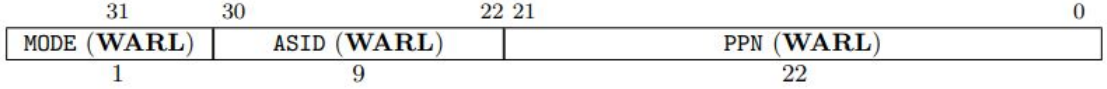


Figure 4.14: Supervisor address translation and protection register `satp` when SXLEN=32.

Atomic CSR instructions (from v1)

CSR Instructions

We define the full set of CSR instructions here, although in the standard user-level base ISA, only a handful of read-only counter CSRs are accessible.

31 20	19 15	14 12	11 7	6 0
csr	rs1	funct3	rd	opcode
12	5	3	5	7
source/dest	source	CSRRW	dest	SYSTEM
source/dest	source	CSRRS	dest	SYSTEM
source/dest	source	CSRRC	dest	SYSTEM
source/dest	uimm[4:0]	CSRRWI	dest	SYSTEM
source/dest	uimm[4:0]	CSRRSI	dest	SYSTEM
source/dest	uimm[4:0]	CSRRCI	dest	SYSTEM

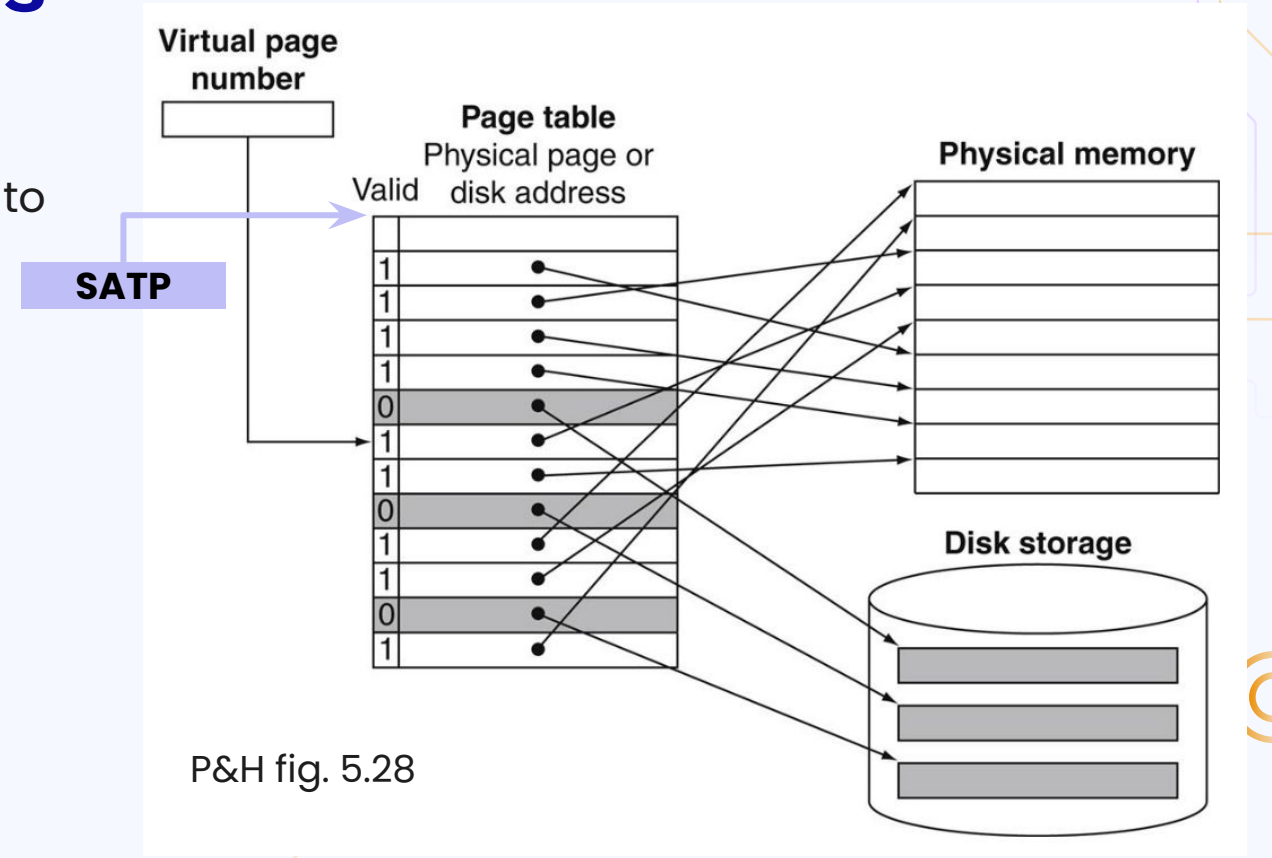
Page tables

One for each process

Map virtual addresses to physical addresses

Not a cache – **why?**

Where does the page table live?



P&H fig. 5.28

Virtual memory in RISC-V 32bit

32 bit virtual address space (4KB pages) → 20 bit virtual page numbers (VPNs)

22-bit physical page number (PPN)

Page tables are the size of a page

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.

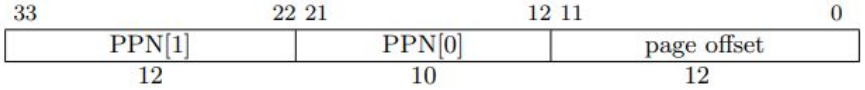


Figure 4.17: Sv32 physical address.

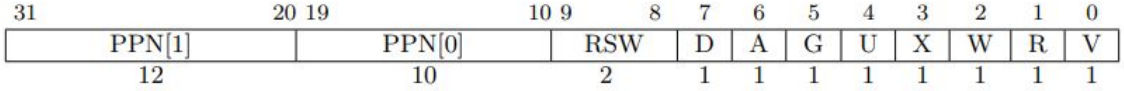
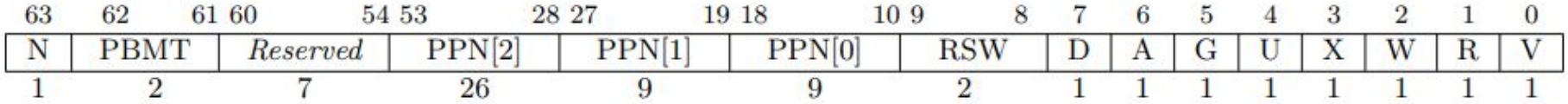


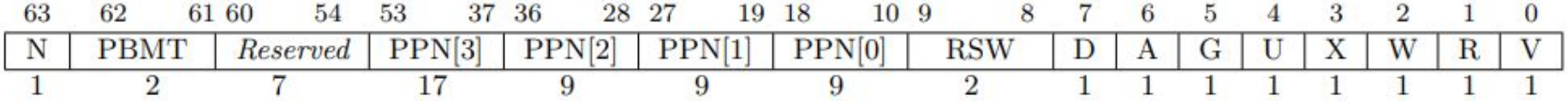
Figure 4.18: Sv32 page table entry.

Virtual memory in RISC-V 64bit



63	62 61	60 54	53 28	27 19	18 10	9 8	7	6	5	4	3	2	1	0
N	PBMT	*Reserved*	PPN[2]	PPN[1]	PPN[0]	RSW	D	A	G	U	X	W	R	V
1	2	7	26	9	9	2	1	1	1	1	1	1	1	1

Figure 4.21: Sv39 page table entry.



63	62 61	60 54	53 37	36 28	27 19	18 10	9 8	7	6	5	4	3	2	1	0
N	PBMT	*Reserved*	PPN[3]	PPN[2]	PPN[1]	PPN[0]	RSW	D	A	G	U	X	W	R	V
1	2	7	17	9	9	9	2	1	1	1	1	1	1	1	1

Figure 4.24: Sv48 page table entry.

? ? ?

What should happen if our virtual address space is so big that the page table can’t efficiently fit in main memory?

Multi-level page table

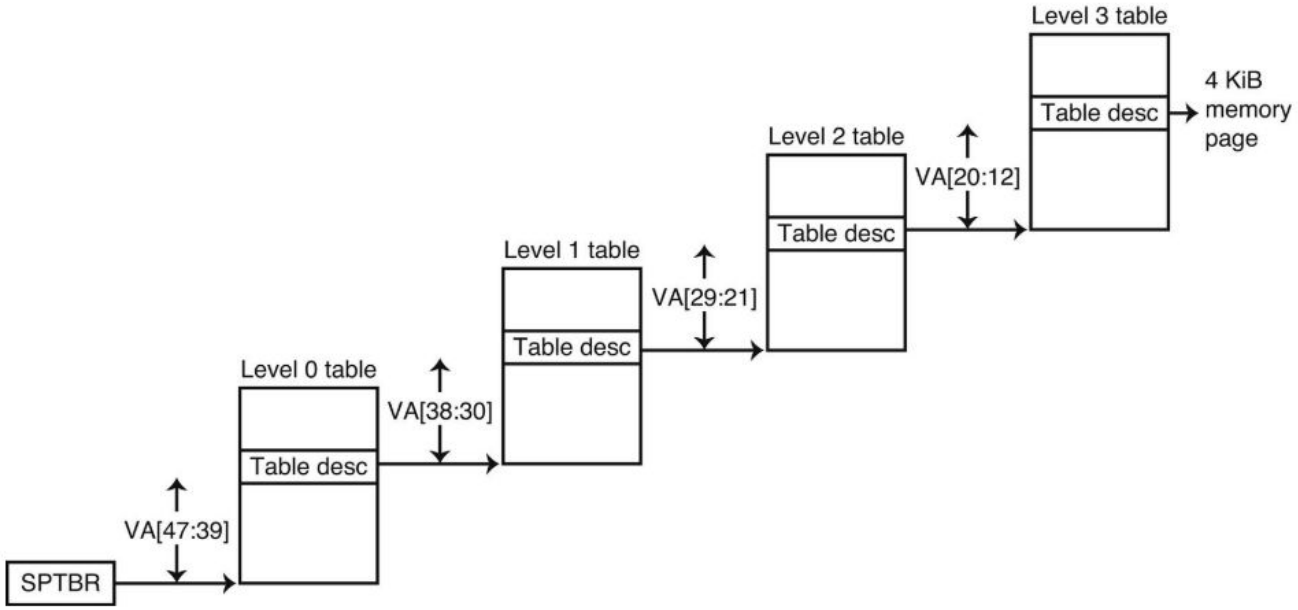


FIGURE 5.29 RISC-V uses four levels of tables to translate a 48-bit virtual address into a 40-bit physical address.