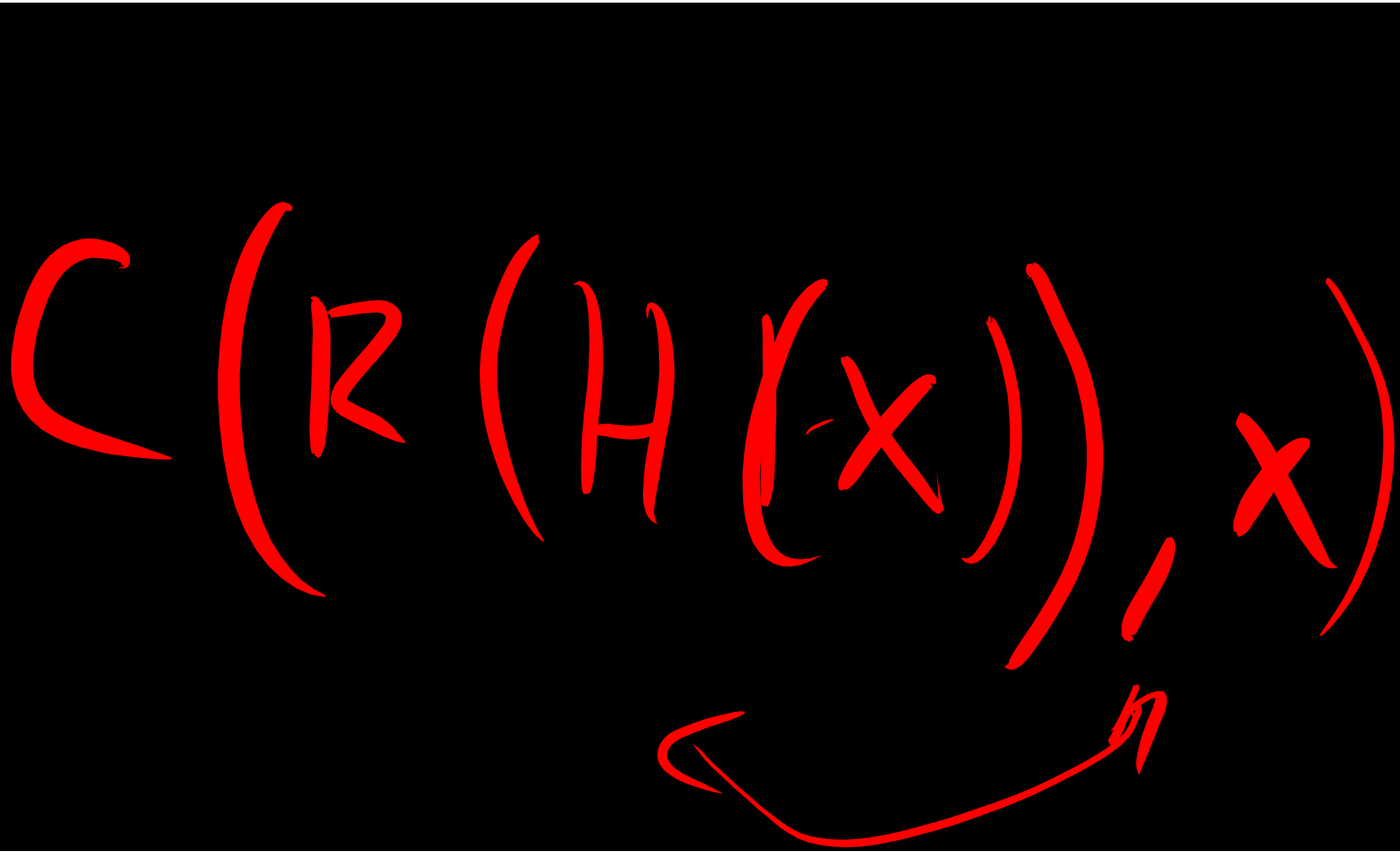
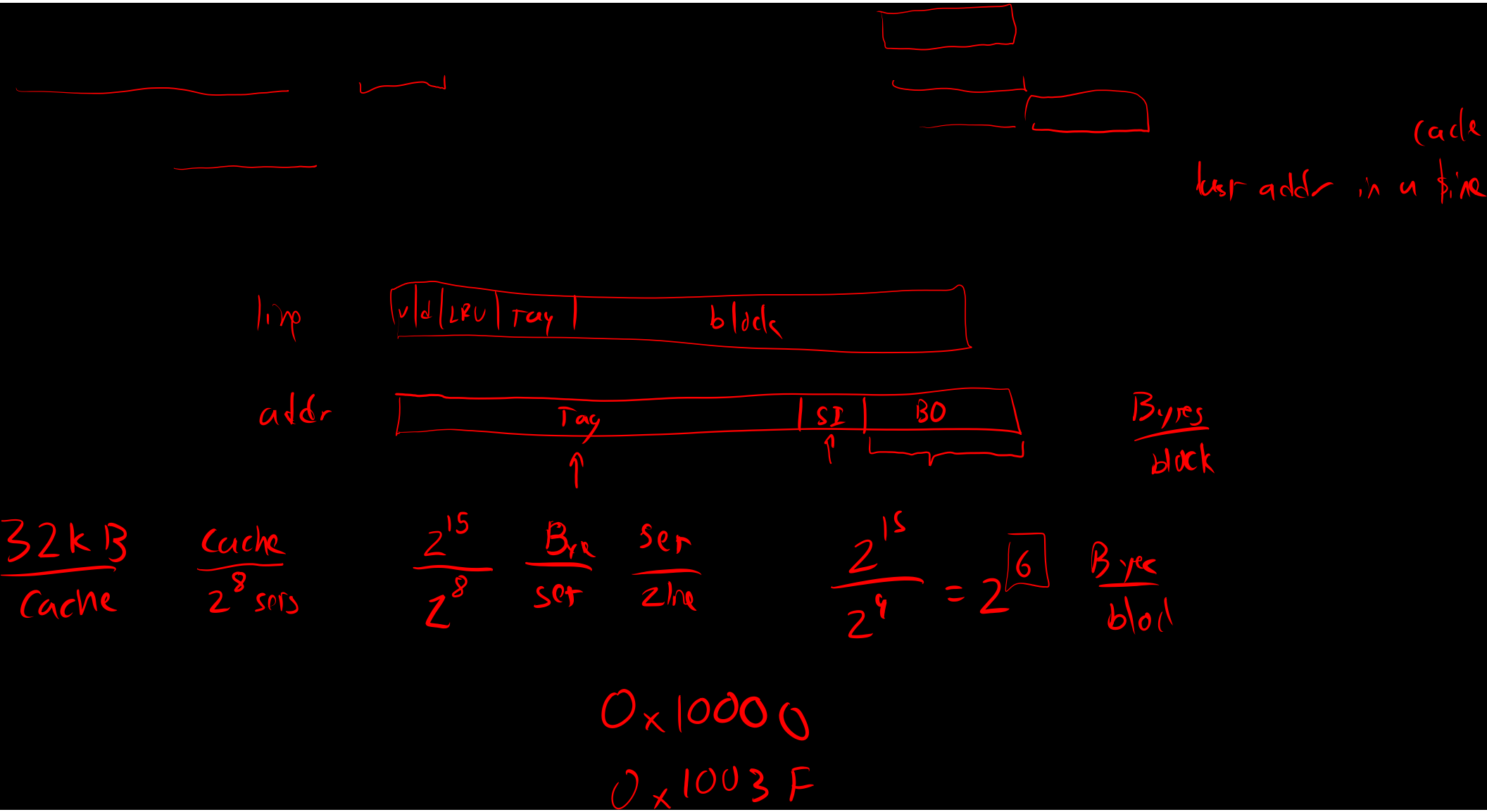
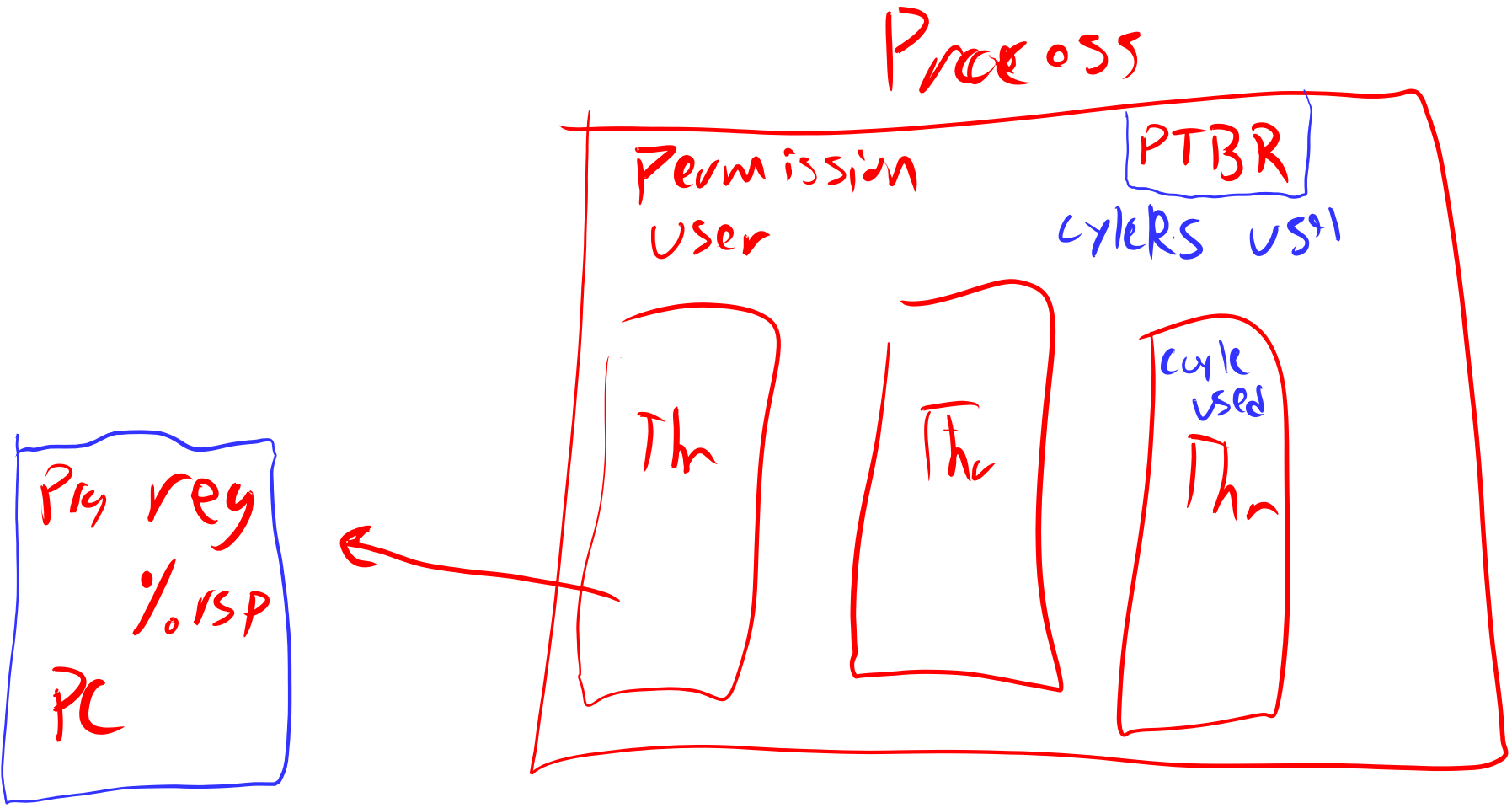

C(R(H(x)), x)


cache
last addr in a line
line
vld
LRU
Tag
block
addr
Tag
SI
BO
Bytes
block
32kB
Cache
Cache
2^8 sets
2^{15}
2^8
Byte
set
set
line
$\frac{2^{15}}{2^9} = 2^6$
Byte
block
0x10000
0x1003F

Process

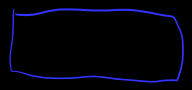
Thread



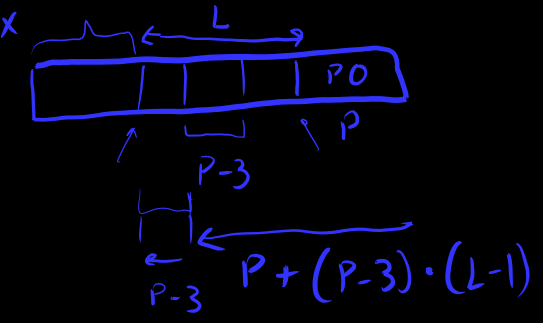
$P_{page} = 2^P$ bytes

PTE = 8 bytes

$\frac{PTE}{P_{page}} = \frac{2^P}{2^3} = 2^{P-3}$



$\lg(8) = 3$



$\sim((\sim 0) << (P-3))$

10000
01111

shift

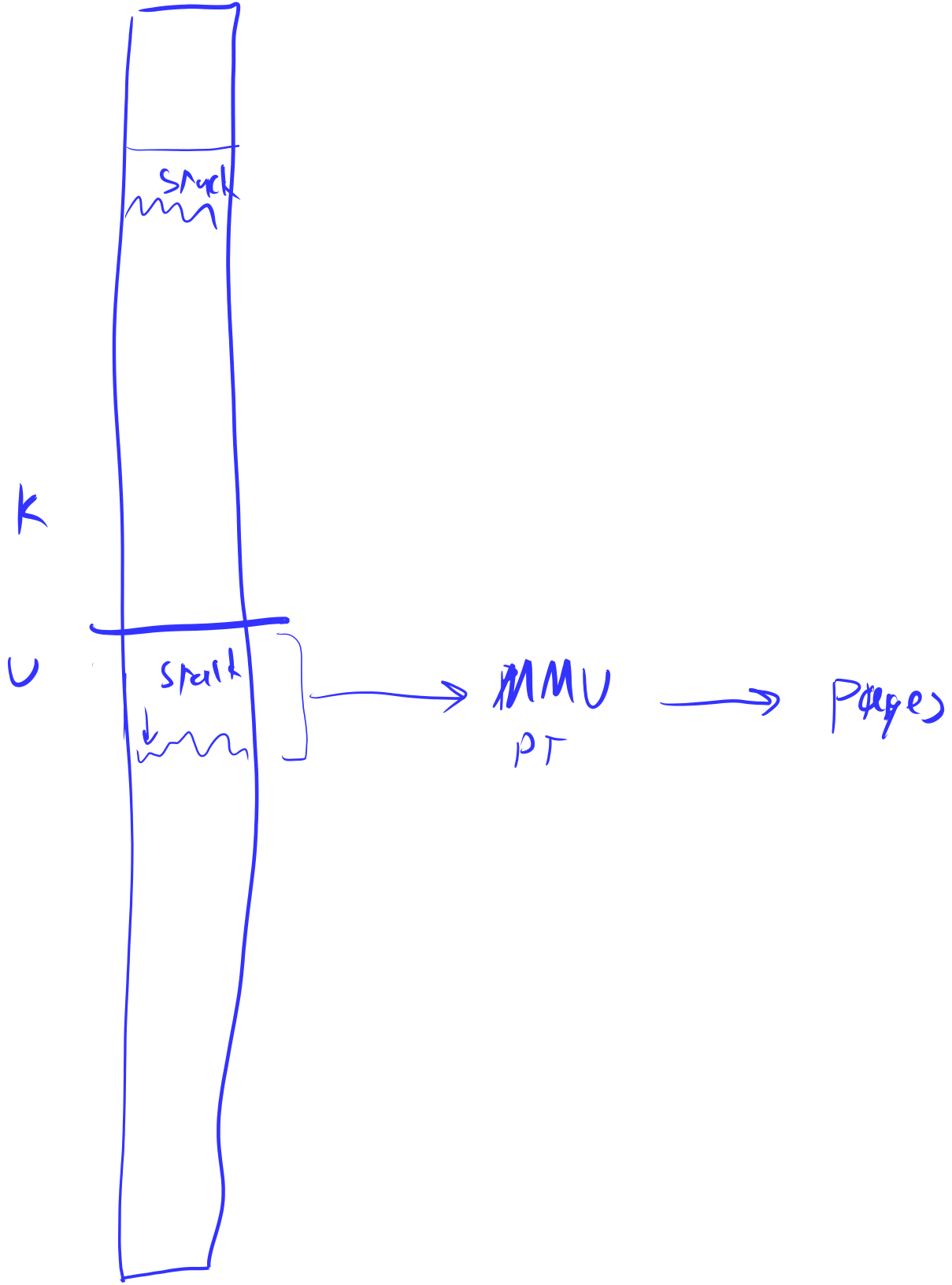
mask

$(x >> (P + (P-3) * (L-1))) \& ((1 << (P-3)) - 1)$


TLB
cach
VA → PA
PA
invalid

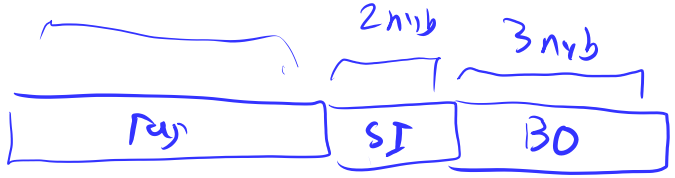
Segfaults

- check segments
  - allowed? → map a physical page
    edit PT
    return - re-run code that faulted
  - not allowed → generate a signal
    - Check for sig handler
      - if ∃, call it
      - otherwise, abort process


K
U
MMU
PT

cache Misses

- cold — first access
- conflict — (Direct-mapped) — 2 addr w/ same idx use close together
- Capacity — (fully-associative) — we've used more lines since last access than our cache holds



0x100000 — cold
0x200000 — cold
0x100000 — conflict

Amdahl's law

$$\text{work} = \text{serial} + \frac{\text{parallizable}}{\text{Threads}} + \text{overhead}$$

Parallizing