ch8. Intro to Paging>

1)

Memory protection
- ARM Cortex-M: Memory Protection Unit (MPU)
- x86
  - segmentation
  - paging.

1.1) Segmentation.

regs
- fetching instrs: `cs` (code segment)
- stack operations (pop/push): `ss` (stack segment)
- other: `ds` (data segment) or `es` (extra segment)
- additional: `fs`, `gs`.

- protected mode:

(regs) idx
segment descriptors → GDT/LDT.
(offset + segment size + access permission)

- cons: fragmentation (external)

bootloader: sets the access permissions for each page

1.2) Paging.

one-to-one
pages → frames
(virt) (phy)

→ Our kernel runs on paging.
(内核中使用 virt addr)
(除 VGA)
virt 0xb8000 → phy 0xb8000
(identity mapped)

- cons: hidden fragmentation (internal frag.)

- Page Table.
  - page + frame + flags.
  - currently active table = `CR3` (on x86)
  - translation cache (page → frame) = TLB

p.s. page fault = `CR2`

- Multilevel PT on x86

offset 1 2 3 4
  - 4-level page table ([0:11], [12:20], [21:29], [30:38], [39:47])
  - a page of 4KiB size (512 entries, each 8 bytes) idx.
  - [48:63] = bit-47 (sign-extension). (P.S. 5-level: [48:57])
    Ice lake Intel CPU

- PTE (8 bytes)

(格式见原文)
52 bits phy addr
- [12:51]: physical address. (4096 bits aligned → [0:11] = 0)
- flags
- available.