

27-missing-notes

Missing Pieces; Love Your Motherboard!

Agenda

- 0. Re-Orienting
 - 1. Privilege
 - 2. Traps, Interrupts, Exceptions
 - 3. MMU
 - 4. Putting it All Together
 - 5. Love your motherboard!
 - 6. Real-World Buses
-

Reading: 8.1; 9.1-9.2; pp 567--570; Ch1

[slides \(https://ssl.cs.dartmouth.edu/~sws/cs51-s15/27-missing/slides.pdf\)](https://ssl.cs.dartmouth.edu/~sws/cs51-s15/27-missing/slides.pdf)

0. Re-Orienting

The computer scientist, all the way down to the laws of physics

How the CS thinks about computation might match the ISA/uarch mapping, until the industry started "dirty tricks" to make things faster (at the cost of increased mess)

Today:

- a few "missing pieces"---not in the Y86---which are relevant to a computer scientist's view of what the machine is doing
 - then jumping from the "simple" view of the Y86 architecture to what's on contemporary motherboards
-

1. Privilege

The simplest version: storing one bit in the processor (e.g., in a flip-flop).

- 0 == "privileged" (aka "kernel"). The CPU is operating in special privileged mode right now.
- 1 == "unprivileged" (aka "user").

Then:

- requiring, in hardware that the CPU be in privileged mode in order to do special, important operations
- establishing very controlled ways in which the CPU can change from user mode to kernel mode, and back.

But real systems often have more complex structures.

2. Traps/Interrupts/Exceptions

(8.1 in the book)

Wait for CS58!

Distinctions between the three

3. Memory Management

(9.1 and 9.2 in the book)

Flexibility includes:

- read and write protections (and others?)
- having multiple address spaces present in physical RAM at the same time
- (and others... wait for CS58!)

4. Putting it All Together

Dance between userland processes, the OS, traps/etc, privilege, MMU
