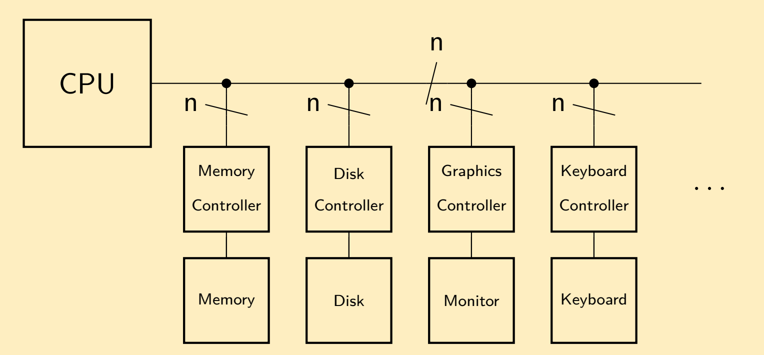
Computer System

January 29, 2025

Classical Computer System

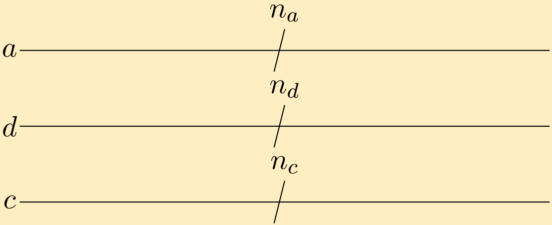


- This is how computer systems **used** to look
- **We** can still think they look like this
- Things get hairy when we program at the kernel level
- (or worse, the firmware level)

Sub Buses

Generaly speaking, the bus lines can be viewed as 3 busses:

- Address bus
- Data bus
- Control bus



Memory Mapped I/O

- Each controller has range of addresses it responds to
  - The memory controller has a huge range
  - The graphics controller might have a large range
  - The other controllers have a rather modest range
- The address ranges of different controllers are **disjoint**
- Initiator of a read/write operation has no idea who it refers to

This is quite the standard nowadays (2024)

I/O Space

In the olden times

- A control line differentiated between memory and I/O
- There were special machine instructions to access I/O controllers
  - in
  - out
- Still the I/O address ranges of different controllers were disjoint

x86 is both ancient and new

- Ancient controllers use the I/O address space
- (Not spotted in nature for eons)
- Less ancient ones are memory mapped

Disjointness, How?

Olden times

- Hardwired into the controller
- Human controlled
  - Jumpers
  - Dip switches

Nowadays

- Firmware
- Might be also OS

Address Space Splitting

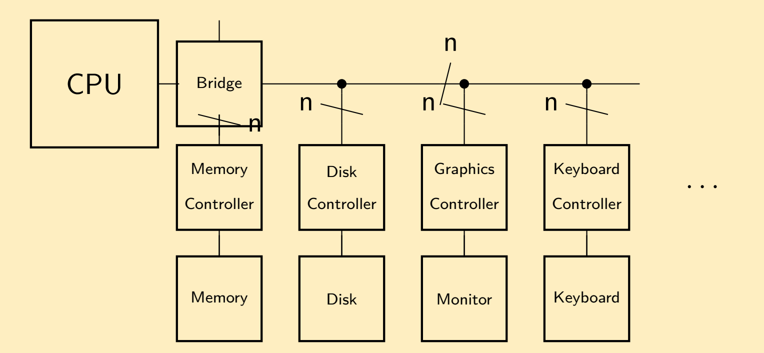
Issues with the Classical View

- Memory has lots of addresses: Wide address bus
- Controllers have modest number of registers: Narrow address bus
- Memory is faster than controllers: Fast bus
- Controllers do not need such a fast bus

Solution: Split the bus

(this has its price)

(the first) Bus splitting



Observe

- The Bridge itself is a controller!
- In the olden times everything was haredwired
- Nowadays everything is software (firmware or OS) controlled

Instead of one **bus** we have

- Memory bus: Wide address and maybe data, fast
- I/O bus: Narrow address and maybe data, slower

The Bridge has hard work to do:

- Transcation from the processor need to be forwareded to one of the busses
- Transactions from the I/O bus need to be forwarded to memory
- Addresses need to be convrted
- (Maybe buffering for optimization)

The Bridge should know the address architecture!

This is a new thing which gets worse along the years

Kind of an History

- The bus is proprietery
- Each of the components is big iron
- The CPU goes on chip
- The CPU die contains more and more controllers
- Standard busses appear: USB, sAta, DDR
- The die contains several processors
- The propriety busses are on die