# CS 61C: Great Ideas in Computer Architecture (Machine Structures)

*Lecture 24*
*More I/O: DMA, Disks, Networking*

Instructors: Krste Asanovic & Vladimir Stojanovic
**Guest Lecturer: Sagar Karandikar**
http://inst.eecs.berkeley.edu/~cs61c/



http://research.microsoft.com/apps/pubs/default.aspx?id=212001

# CS61C / CS150 in the News

**Microsoft "Catapult", ISCA 2014**

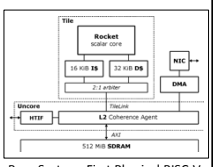

FPGAs are "programmable" hardware used by computer architects and digital circuit designers, lie somewhere between CPUs and custom chips (ASICs).

"Microsoft published a paper at ISCA about using FPGAs in datacenters for page ranking processing for Bing. In a test deployment, MS reported up to 95% more throughput for only 10% more power. The added TCO was less than 30%. Microsoft used Altera Stratix V FPGAs in a PCIe form-factor with 8GB of DDR3 RAM on each board. The FPGAs were connected using a 10Gb SAS network." - AnandTech

ASPIRE

## Hardware Acceleration of Key-Value Stores

- Datacenter apps, path through CPU/kernel/app ≈ 86% of request latency
- Goal: Serve popular Key-Value Store GET requests without CPU
- Soft-managed cache attached to NIC, RoCC CPU interface
- Benchmarking on FPGA:
  - –RISC-V Rocket @ 50 MHz
  - –NIC from TEMAC/PCS-PMA + 1 Gb SFP
  - –Hardware KV-Store Accelerator
  - –Traffic Manager, DMA Engine
- •Written in Chisel

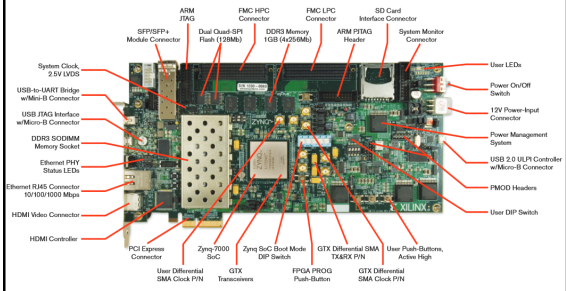



Base System: First Physical RISC-V System with Networking Support

*Sagar Karandikar*
Howard Mao
Albert Ou
Yunsup Lee
Krste Asanovic

ASPIRE

## Hardware Acceleration of Key-Value Stores




–Traffic Manager, DMA Engine
•Written in Chisel

Yunsup Lee
Krste Asanovic

# Review: I/O

- Programmed I/O:
  - – CPU execs lw/sw instructions for all data movement to/from devices
  - – CPU spends time doing 3 things:
    - Getting data from device to main mem.
    - Using data to compute
    - Sending data back to main mem.
- Polling
- Interrupts



# Working with real devices

- ~~Programmed I/O:~~ **DMA**
  - – ~~CPU execs lw/sw instructions for all data movement to/from devices~~
  - – CPU spends time doing ~~3 things~~:
    - ~~Getting data from device to main mem.~~
    - Using data to compute
    - ~~Sending data back to main mem.~~
- Polling
- Interrupts

## Agenda

- **Direct Memory Access (DMA)**
- Disks
- Networking

7

## What's wrong with Programmed I/O?

- CPU has sole control over main memory
- Not ideal since…
  - CPU has to execute all transfers, could be doing other work
  - Device speeds don't align well with CPU speeds
  - Energy cost of using beefy general-purpose CPU where simpler hardware would suffice

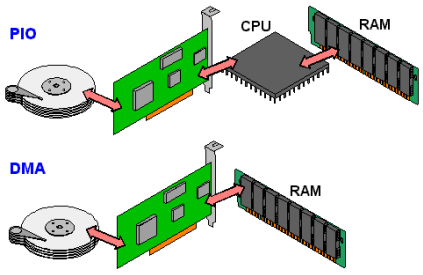
8

## Direct Memory Access (DMA)

- Allows other devices to directly read/write from main memory
- New Hardware: the DMA Engine
- DMA engine contains registers written by CPU:
  - Memory address to place data
  - # of bytes
  - I/O port #, direction of transfer
  - unit of transfer, amount to transfer per burst

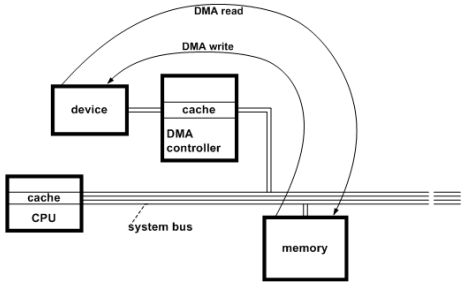
9

## PIO vs. DMA




10

## System with DMA




11

## DMA: Incoming Data

- Receive interrupt from device
- CPU takes interrupt, begins transfer
  - Instructs DMA engine/device to place data @ certain address
- Device/DMA engine handle the transfer
  - CPU is free to execute other things
- Upon completion, Device/DMA engine interrupt the CPU again

12

## DMA: Outgoing Data

- CPU decides to initiate transfer, confirms that external device is ready
- CPU begins transfer
  - Instructs DMA engine/device that data is available @ certain address
- Device/DMA engine handle the transfer
  - CPU is free to execute other things
- Device/DMA engine interrupt the CPU again to signal completion

13

## DMA: Some new problems

- Where in the memory hierarchy do we plug in the DMA engine? Two extremes:
  - Between L1 and CPU:
    - Pro: Free coherency
    - Con: Trash the CPU's working set with transferred data
  - Between Last-level cache and main mem:
    - Pro: Don't mess with caches
    - Con: Need to explicitly manage coherency

14

## DMA: Some new problems

- How do we arbitrate between CPU and DMA Engine/Device access to memory? Three options:
  - Burst Mode
    - Start transfer of data block, CPU cannot access mem in the meantime
  - Cycle Stealing Mode
    - DMA engine transfers a byte, releases control, then repeats - interleaves processor/DMA engine accesses
  - Transparent Mode
    - DMA transfer only occurs when CPU is not using the system bus

15

## Administrivia

- HKN Course Surveys on Tuesday
- Midterm 2 scores up:
  - Regrade request deadline is 23:59:59 on Sunday April 26th
- Proj 4-1 due date extended to Wed, April 29
- Proj 4-2 due Sunday, May 3
  - Run your Proj 4-1 code on a ~12 node EC2 cluster
- HW6 (Virtual Memory) due May 3

16

## iClicker: How's Project 4-1 going?

A) Haven't started yet
B) I've read the spec/sample code
C) I've written some code
D) I'm nearly done
E) I'm finished

17

## Agenda

- Direct Memory Access (DMA)
- **Disks**
- Networking

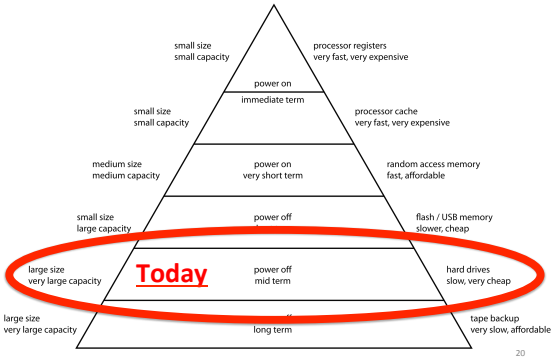
18

## Review - 6 Great Ideas in Computer Architecture

1. Layers of Representation/Interpretation
2. Moore's Law
3. **Principle of Locality/Memory Hierarchy**
4. Parallelism
5. Performance Measurement & Improvement
6. Dependability via Redundancy

19

## Computer Memory Hierarchy



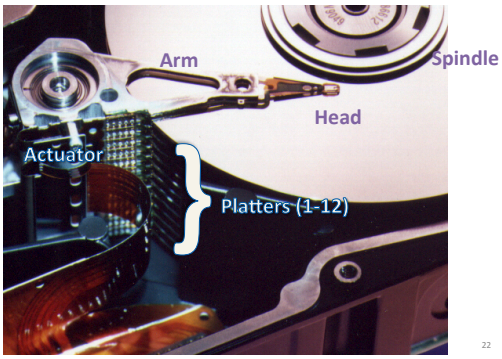

20

## Magnetic Disk – common I/O device

- A kind of computer memory
  - Information stored by magnetizing ferrite material on surface of rotating disk
    - similar to tape recorder except digital rather than analog data
- A type of non-volatile storage
  - retains its value without applying power to disk.
- Two Types of Magnetic Disk
  - Floppy disks – slower, less dense, removable.
  - Hard Disk Drives (HDD) – faster, more dense, non-removable.
- Purpose in computer systems (Hard Drive):
  - Long-term, inexpensive storage for files
  - "Backup" for main-memory. Large, inexpensive, slow level in the memory hierarchy (virtual memory)

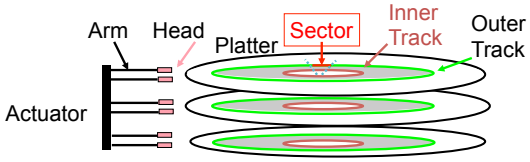
21

## Photo of Disk Head, Arm, Actuator




22

## Disk Device Terminology

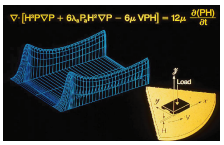



- Several platters, information recorded magnetically on both surfaces (usually)
- **Bits recorded in tracks, which in turn divided into sectors (e.g., 512 Bytes)**
- **Actuator moves head (end of arm) over track ("seek"), wait for sector rotate under head, then read or write**

23

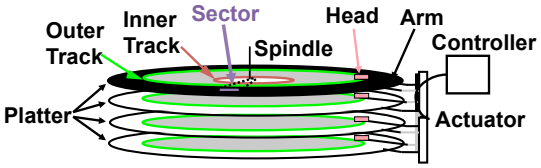
## Hard Drives are Sealed. Why?

- The closer the head to the disk, the smaller the "spot size" and thus the denser the recording.
  - Measured in Gbit/in^2
  - ~900 Gbit/in^2 is state of the art
  - Started out at 2 Kbit/in^2
  - ~450,000,000x improvement in ~60 years
- Disks are sealed to keep the dust out.
  - Heads are designed to "fly" at around 5-20nm above the surface of the disk.
  - 99.999% of the head/arm weight is supported by the air bearing force (air cushion) developed between the disk and the head.



24

## Disk Device Performance (1/2)




- **Disk Access Time = Seek Time + Rotation Time + Transfer Time + Controller Overhead**
  - Seek Time = time to position the head assembly at the proper cylinder
  - Rotation Time = time for the disk to rotate to the point where the first sectors of the block to access reach the head
  - Transfer Time = time taken by the sectors of the block and any gaps between them to rotate past the head

25

## Disk Device Performance (2/2)

- Average values to plug into the formula:
- Rotation Time: Average distance of sector from head?
  - 1/2 time of a rotation
    - 7200 Revolutions Per Minute ⇒ 120 Rev/sec
    - 1 revolution = 1/120 sec ⇒ 8.33 milliseconds
    - 1/2 rotation (revolution) ⇒ 4.17 ms
- Seek time: Average no. tracks to move arm?
  - Number of tracks/3 (see CS186 for the math)
  - Then, Seek time = number of tracks moved * time to move across one track

26

## But wait!

- Performance estimates are different in practice
- Lots of "magic" going on behind the scenes in disks. One example:
  - Many disks have on-disk caches, which are completely hidden from the outside world
  - Previous formula completely replaced with on-disk cache access time

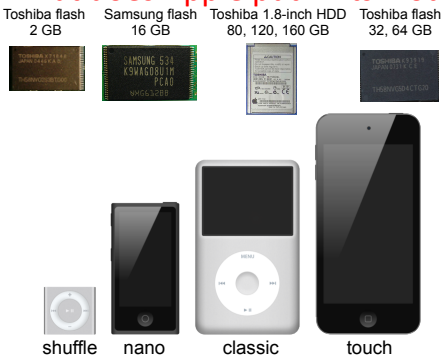
27

## Where does Flash memory come in?

- ~5-10 years ago: Microdrives and Flash memory (e.g., CompactFlash) went head-to-head
  - Both non-volatile (no power, data ok)
  - Flash benefits: durable & lower power (no moving parts, need to spin µdrives up/down)
  - Disk cost = fixed cost of motor + arm mechanics, but actual magnetic media cost very low
  - Flash cost = most cost/bit of flash chips
  - Over time, cost/bit of flash came down, became cost competitive

28

## What does Apple put in its iPods?




29

## What does Apple put in its iPods?


30

## Flash Memory / SSDs

- How does Flash memory work?
  - NMOS transistor with an additional conductor between gate and source/drain which "traps" electrons. The presence/absence is a 1 or 0
- Requires complex management algorithms to avoid wearing out cells
  - Early SSDs had severe reliability issues due to poorly implemented firmware
  - More in CS162

31

## iClicker Question

- We have the following disk:
  - 15000 Cylinders, 1 ms to cross 1000 Cylinders
  - 15000 RPM = 4 ms per rotation
  - Want to copy 1 MB, transfer rate of 1000 MB/s
  - 1 ms controller processing time
- What is the access time using our model?

**Disk Access Time = Seek Time + Rotation Time + Transfer Time + Controller Processing Time**

| A | B | C | D | E |
|---|---|---|---|---|
| 10.5 ms | 9 ms | 8.5 ms | 11.4 ms | 12 ms |

32

## Clicker Question

- We have the following disk:
  - 15000 Cylinders, 1 ms to cross 1000 Cylinders
  - 15000 RPM = 4 ms per rotation
  - Want to copy 1 MB, transfer rate of 1000 MB/s
  - 1 ms controller processing time
- What is the access time?

Seek = # cylinders/3 * time = 15000/3 * 1ms/1000 cylinders = 5ms

Rotation = time for ½ rotation = 4 ms / 2 = 2 ms

Transfer = Size / transfer rate = 1 MB / (1000 MB/s) = 1 ms

Controller = 1 ms

Total = 5 + 2 + 1 + 1 = 9 ms

33

## Agenda

- Direct Memory Access (DMA)
- Disks
- **Networking**

34

## Networks: Talking to the Outside World

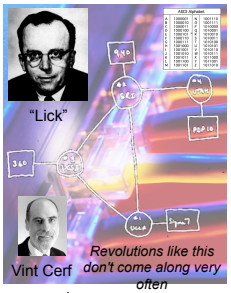
- Originally sharing I/O devices between computers
  - E.g., printers
- Then communicating between computers
  - E.g., file transfer protocol
- Then communicating between people
  - E.g., e-mail
- Then communicating between networks of computers
  - E.g., file sharing, www, …

4/24/15 35

www.computerhistory.org/internet_history

## The Internet (1962)

- History
  - JCR Licklider, as head of ARPA, writes on "intergalactic network"
  - 1963 : ASCII becomes first universal computer standard
  - 1969 : Defense Advanced Research Projects Agency (DARPA) deploys 4 "nodes" @ UCLA, SRI, Utah, & UCSB
  - 1973 Robert Kahn & Vint Cerf invent TCP, now part of the Internet Protocol Suite
- Internet growth rates
  - Exponential since start!




www.greatachievements.org/?id=3736

4/24/15 en.wikipedia.org/wiki/Internet_Protocol_Suite 36

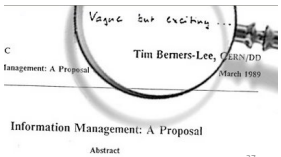
en.wikipedia.org/wiki/History_of_the_World_Wide_Web

## The World Wide Web (1989)

- "System of interlinked hypertext documents on the Internet"
- History
  - 1945: Vannevar Bush describes hypertext system called "memex" in article
  - 1989: Sir Tim Berners-Lee proposes, gets system up '90
  - ~2000 Dot-com entrepreneurs rushed in, 2001 bubble burst
- Wayback Machine
  - Snapshots of web over time
- Today : Access anywhere!

Tim Berners-Lee

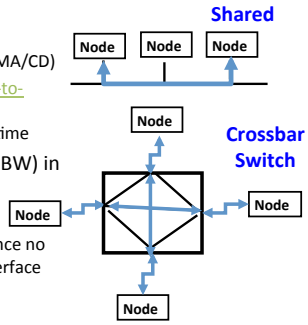
World's First web server in 1990






## Shared vs. Switch-Based Networks

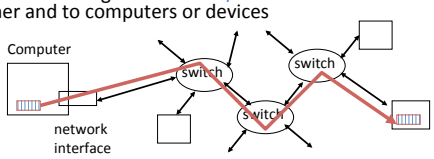
- Shared vs. Switched:
  - Shared: 1 at a time (CSMA/CD)
  - Switched: pairs ("point-to-point" connections) communicate at same time
- Aggregate bandwidth (BW) in switched network is many times shared:
  - point-to-point faster since no arbitration, simpler interface






## What makes networks work?

- links connecting switches and/or routers to each other and to computers or devices



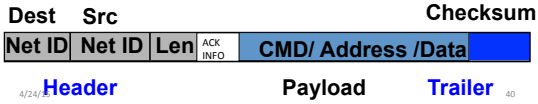

- **ability to name the components and to route packets of information - messages - from a source to a destination**

- **Layering, redundancy, protocols, and encapsulation as means of abstraction (61C big idea)**



## Software Protocol to Send and Receive

- SW Send steps
  1: Application copies data to OS buffer
  2: OS calculates checksum, starts timer
  3: OS sends data to network interface HW and says start
- SW Receive steps
  3: OS copies data from network interface HW to OS buffer
  2: OS calculates checksum, if OK, send ACK; if not, delete message (sender resends when timer expires)
  1: If OK, OS copies data to user address space, & signals application to continue






## Networks are like Ogres

https://www.youtube.com/watch?v=_bMcXVe8zIs



## Protocol for Networks of Networks?

What does it take to send packets across the globe?

- Bits on wire
- Packets on wire
- Delivery packets within a single physical network
- Deliver packets across multiple networks
- Ensure the destination received the data
- Do something with the data

## Protocol for Networks of Networks?

- **Abstraction to cope with complexity of communication**
- **Networks are like ~~ogres~~ onions**
  - **Hierarchy of layers:**
    - **Application (chat client, game, etc.)**
    - **Transport (TCP, UDP)**
    - **Network (IP)**
    - **Data Link Layer (ethernet)**
    - **Physical Link (copper, wireless, etc.)**

Networks are like onions.

They stink?

Yes. No!

Oh, they make you cry.

No!... Layers. Onions have layers. Networks have layers.



## Protocol Family Concept

- Key to protocol families is that communication occurs logically at the same level of the protocol, called peer-to-peer…

  …but is implemented via services at the next lower level

- Encapsulation: carry higher level information within lower level "envelope"



## Inspiration…

- CEO A writes letter to CEO B
  - Folds letter and hands it to assistant
  - Puts letter in envelope with CEO B's full name
  - Takes to FedEx
- FedEx Office
  - Puts letter in larger envelope
  - Puts name and street address on FedEx envelope
  - Puts package on FedEx delivery truck
- FedEx delivers to other company

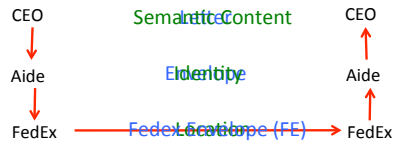
***Dear John,***

***Your days are numbered.***

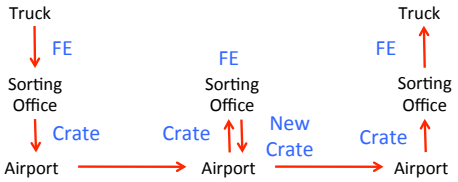
***–Pat***

## The Path of the Letter

"Peers" on each side understand the same things
No one else needs to
Lowest level has most packaging

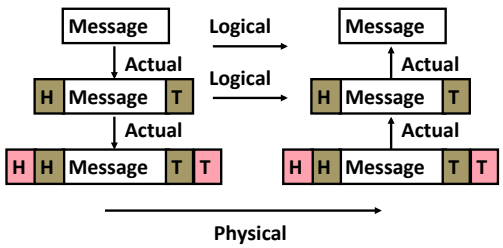



## The Path Through FedEx




Deepest Packaging (Envelope+FE+Crate)
at the Lowest Level of Transport

## Protocol Family Concept



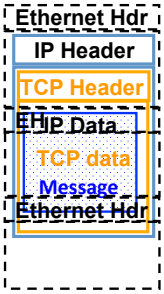
## Protocol for Network of Networks

- Transmission Control Protocol/Internet Protocol (TCP/IP)
  (TCP :: a Transport Layer)
  - This protocol family is the basis of the Internet, a WAN protocol
  - IP makes best effort to deliver
    - Packets can be lost, corrupted
  - TCP guarantees delivery
  - TCP/IP so popular it is used even when communicating locally: even across homogeneous LAN



## TCP/IP packet, Ethernet packet, protocols

- Application sends message
- **TCP breaks into 64KiB segments, adds 20B header**
- **IP adds 20B header, sends to network**
- **If Ethernet, broken into 1500B packets with headers, trailers**






## "And in conclusion…"

- I/O gives computers their 5 senses
- I/O speed range is 100-million to one
- Polling vs. Interrupts
- DMA to avoid wasting CPU time on data transfers
- Disks for persistent storage, replaced by flash
- Networks: computer-to-computer I/O
  - Protocol suites allow networking of heterogeneous components. Abstraction!!!