**Networks, Internet and Hardware**

**Part 1**

**Representation of Data**

**Bit of This and a Bit of That**

---

**What is a Number?**

- We use the Hindu Numeric System
  - positional grouping system
  - each position represents a power of 10
- Binary numbers
  - based on the same system
  - use powers of $2$ rather than 10

---

**Base 10 Number**

The number $1846$ is ...

<table>
<thead>
<tr>
<th>$10^4$</th>
<th>$10^3$</th>
<th>$10^2$</th>
<th>$10^1$</th>
<th>$10^0$</th>
</tr>
</thead>
<tbody>
<tr>
<td>10000</td>
<td>1000</td>
<td>100</td>
<td>10</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>8</td>
<td>4</td>
<td>6</td>
</tr>
</tbody>
</table>

$(1 \times 1000) + (8 \times 100) + (4 \times 10) + (1 \times 6) = 1846$

---

**Binary Number Example**

The number $01101001$ is ...

<table>
<thead>
<tr>
<th>$2^7$</th>
<th>$2^6$</th>
<th>$2^5$</th>
<th>$2^4$</th>
<th>$2^3$</th>
<th>$2^2$</th>
<th>$2^1$</th>
<th>$2^0$</th>
</tr>
</thead>
<tbody>
<tr>
<td>128</td>
<td>64</td>
<td>32</td>
<td>16</td>
<td>8</td>
<td>4</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>

$64 + 32 + 8 + 1 = 105$
**Binary Number Example**

The number **1101 1011** is ...

<table>
<thead>
<tr>
<th>2⁷</th>
<th>2⁶</th>
<th>2⁵</th>
<th>2⁴</th>
<th>2³</th>
<th>2²</th>
<th>2¹</th>
<th>2⁰</th>
</tr>
</thead>
<tbody>
<tr>
<td>128</td>
<td>64</td>
<td>32</td>
<td>16</td>
<td>8</td>
<td>4</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

\[128 + 64 + 16 + 8 + 2 + 1 = 219\]

**Bits & Bytes**

- One binary digit is called a **bit**
  - either 1 or 0
  - shorthand for a bit is **b**
- A group of 8 bits is a **byte**
  - (Europe did not like the name so they call it an **octet**)
  - e.g. **0010 0100**
  - shorthand for a byte is **B**

**Hexadecimal Numbers**

- Writing out long binary numbers is cumbersome and error prone
- As a result, computer scientists often write computer numbers in **hexadecimal**
- Hexadecimal is base-16
  - We only have 0...9 to represent digits
  - So, hexadecimal uses A...F to represent 10...15

The number **1AC** is ...

\[ (1 \times 256) + (10 \times 16) + 12 = 428\]

**Converting Binary to Hex = Easy**

- Since 16 = 2⁴, a single hex character can represent a total of 4 bits
- Byte can be represented with 2 hex digits!
- When looking at raw data, Hex Editors, display bytes as groups of 2 hex characters

**Network Basics**

What is a Network?
What is a Network?

- When computers are connected...
  - able "talk" to each other
  - able exchange data
- They can be connected:
  - using ports and wires
  - radio waves
  - etc....

Networking Advantages

- Share Data
  - transfer files
  - send e-mail
- Share Devices
  - lab printers
  - workstation scanner...
- Work together
  - regardless of time and place
  - e.g. messaging, Skype

Network Disadvantages

- More vulnerable to unauthorized access
  - data theft – the wrong people get data
  - hacking – someone attacks your computer
- More vulnerable to malicious code
  - worms
  - viruses
  - etc....

Networking Disadvantages

Geographic Scope

- Personal Area Network
- Local Area Network
- Neighborhood Area Network
- Metropolitan Area Network
- Wide Area Network

Bandwidth

- **Communication channel**
  - physical path or a frequency
  - e.g. computer cables, cell phone frequency
- **Bandwidth**
  - capacity of a communications channel
  - the higher the value, the more data can be transferred in a given amount of time

Bandwidth

- High-bandwidth systems
  - aka *broadband*
  - excellent for transmitting multimedia
- Low-bandwidth systems
  - aka *narrowband*
  - adequate for simple data
  - e.g. messaging, e-mail, twitter
Network Links

- Wired network
  - data travels between devices over a **cable**
  - best approach for stationary computers
- Wireless network
  - data travels between devices through the **air**
  - increasingly popular – but allows ease droppers

What is a Protocol?

- For computers to communicate, they need to "speak" to each other
- A protocol defines:
  - rules for sending
  - how data packaged
  - when data is transmitted

Protocol Layers

- Application Layer
- Transport Layer
- Network Layer
- Data Link Layer
- Physical Layer

London Bridge is Falling Down ...

- London Bridge
  - commuter bridge constructed in 1831
  - don't confuse it with the London's Tower Bridge
- By 1924 ...
  - London Bridge **was** falling down
  - it was also sinking in the clay
- The British decided build a new bridge

London Bridge is Falling Down ...

- Robert P. McCulloch
  - businessman
- bought London Bridge
- decided to move it to Lake Havasu City, Arizona
- How do you move a bridge?
London Bridge is Moving 'round

- Break the bridge into pieces
  - since the bridge is masonry
  - "bricks" was the logical choice
- Each was labeled carefully
  - the span it was in: 5 total
  - the row of stone it was in
  - its position in the row

- Bricks were put into crates
  - Each crate was *carefully* labeled
  - Crates were transported
    - moved by ship to the East Coast
    - then, trucked to Arizona

London Bridge is Homeward Bound

- The crates
  - stuff arrived at different times
  - and took different routes
- Since it was organized...
  - bridge could be reconstructed
  - and it was!
  - London Bridge is in Arizona!

Lake Havasu, Arizona

- The "Bricks"
- Top-most layer
- Protocol depends on the applications
  - what type of data is being sent
  - what extra information is needed
Transport Layer

- The "Crates"
- The units (groups) of data that pass over a network
  - the "crates" are called packets
  - different networks use different formats
- The Internet uses TCP

Network Layer

- "Crate Mailing Addresses"
- Each computer has a unique address
  - contains a Source and Destination
  - allows packets to be "mailed" to the target
- The Internet uses IP

Data Link Layer

- The "Shipping Department"
- Determines
  - when data will be sent over physical devices
  - which physical devices
- Sends bursts of data called "frames"
- Ethernet (IEEE 802.x) is the dominate format

Physical Layer

- The "Boats and Trucks"
- The physical way data is transferred
- Examples
  - wires
  - fiber optic cables
  - radio and microwave transmissions

Protocol Layer Summary

- Application Layer: Bricks
- Transport Layer: Crates
- Network Layer: Mailing Address
- Data link Layer: Shipping Routes
- Physical Layer: Boats, trucks, etc...

The Internet

- The Information Super Highway
What is the Internet?
- The *Internet* is a massive collection of networks that are linked together
- Computers can anywhere in the World
- Originally designed by the United States

What is the Internet?
- Designed to never go down
- Main routes are the *Internet backbone*
- Networks communicate using the TCP/IP protocols

How the Internet Got Started
- In 1957, the USSR launched *Sputnik* – the first human-made satellite
- At the time, the Cold War divided the world between democracy and communism
- The US feared it was losing the "space race" and

How the Internet Got Started
- In response, the US created the *Advanced Research Projects Agency (ARPA)*
- *ARPANET*
  - connected computers at 4 universities - 1969
  - help scientists communicate
  - share valuable computer resources
- This eventually evolved into the Internet

Who Controls the Internet?
- It is *not* owned, operated, or controlled by any single entity
- It is controlled by:
  - private businesses
  - ... at least in the United States
  - Federal Government only provides some management, but contracts this duty

Network Service Providers
- Manage and own parts of the Internet
- *Network Access Points*
  - connection points between Network Service Providers
  - handles a large amount of traffic
Internet Service Providers

- Internet Service Providers operate network devices that handle the physical aspects of transmitting and receiving data
- Many ISPs
  - connect the user to the Internet
  - handle incoming and outgoing mail
  - maintain web servers for subscriber Web sites

Success of the Internet

- The Internet is incredibly successful
- This success is due to that the Internet only defines the protocol layers it needs to connect separate networks

How does the Internet work?

Internet Protocols

Just TCP and IP...

- To send data, computers...
  - need the data to be transported
  - need to uniquely identify the target so data can moved over the network
- It does not specify an application layer...
  - so, it doesn't care what is being sent
  - only how to get to its destination
Internet Protocols

**Transport Layer: TCP**
- **Transport Control Protocol (TCP)** is used by bundle and transport packets
- Features:
  - requires a connection between the two hosts
  - guarantees that the data will arrive without errors, no missing segments and no duplicate segments
  - sender will not send faster than the receiver can accept
  - practically all data sent on the Internet uses TCP

**Transport Layer: UDP**
- **User Datagram Protocol (UDP)** is an alternative to TCP
- Features:
  - is simple and fast
  - doesn’t require a two-way connection
  - but, not reliable and no guarantee of delivery
- Used when a lost packet is not a problem – e.g. streaming video

**Network Layer: Internet Protocol**
- Internet uses Internet Protocol (IP) to identify computers on the network
- Benefits:
  - easy to implement and extensible
  - public and free
  - allows all the different networks to communicate
- Internet Addresses - **IP Address**
  - format of addresses used by the Internet
  - every device on the Internet has one

**Internet Protocol version 4 (IPv4)**
- Older, but still the most common format
- Each address is 32-bit
  - addresses are denoted as 4 decimal numbers delimited by periods
  - this allows up to 4 billion addresses
  - however, we ran out in the 1990’s!

**Internet Protocol version 6 (IPv6)**
- Overcomes many shortcomings of IPv4
- It expands the IP address to 128-bit!
  - this allows over $3.4 \times 10^{38}$ addresses
  - that’s $430,413,813,021,962,048,220$ addresses (430 Quintillion) for every square inch of the planet Earth!
  - addresses are denoted in hexadecimal
Network Hardware

Data Link & Physical Layer

Network Technology

- Any computing device connected to a network is referred to as a **host**
- This device can be:
  - shared printer
  - network hard drive
  - your computer
  - etc.

Network Interface Controller (NIC)

- Hardware that connects a device is called **network interface controller**
- Connects to network cable or has an wireless antenna/transmitter
- Historically called a network interface **card**:
  - since they were often expansion cards
  - now they are typically built into motherboards

Network Interface Controller (NIC)

- Each controller has a **media access control (MAC)** address
  - not a Macintosh computer!
  - **unique** value that identifies a **specific** card
  - usually **hard-wired** by the card manufacturer
- Works on Physical and Data-Link layer

Network Interface Controller (NIC)

- Think of a MAC the device's **physical coordinates** – longitude and latitude
- Mail truck metaphor:
  - trucks drive on shared streets
  - …but only delivers it to the right location
- Data-link layer sends out data:
  - card, that it is meant for, receives the data
  - the rest just ignore it as **background noise**

Hubs

- Wires become a problem!
- A device called a hub allowed multiple devices to connect together
  - **Physical-layer** ("dumb") repeaters
- Just resends copies of data over multiple lines
Hubs: What They Do

- Bits coming in the up-link go out all other links at same rate
- So, all computers connected to the hub get the same data
- Computers simply "ignore" data that is not for them

Switches

- Datalink-layer device
- Smarter than hubs, take active role…
  - exam incoming packets
  - only send it to the right destination
- Benefits
  - hosts are unaware of presence of them
  - plug-and-play
  - self-learning

Ethernet

- Data-Link Layer protocol
- Commonly used for computers on LANs
- Still in common use today
  - easy to maintain, setup, etc....
  - used by both wired and wireless technology
- Speed as increased over time
  - 10 Mbs at first
  - now up to 100 Mbs to 1000 Mbs

- Data travels...
  - simultaneously broadcasts to all devices
  - first come, first served basis
  - If two workstations send data at the same time, a collision occurs. That data must be resent.
- Current standards
  - IEEE 802.3
  - CSMA/CD
Network Address Translation

What a wonderful and clever idea

Original IPv4 Address Format

- Originally IPv4 was structured into 3 different “classes”
- Each allows different number of owners (“networks”) and hosts (“addresses”)
- Different organizations could get a Class A, B or C block

Original IPv4 Address Format

- The classes:
  - Class A – 254 networks with 16 millions hosts
  - Class B – 16,384 networks with 65,536 hosts
  - Class C – 2 million networks with 255 hosts
- So, for example, a Class B has 65,536 unique address that they can use at their leisure

Original IPv4 Address Format

- Class A includes the large NSPs such as AT&T
- Class B typically contains large business and universities
- Class C is everyone else

IP Addresses Classes

- Class format was ultimately abandoned when new IP Addresses were becoming scarce
- Classless Inter-Domain Routing (CIDR)
  - replaced the class system
  - more flexible and allows “subnets” to be any bit-length
- However, existing licenses are still valid
  - Sac State has a Class B license and has over 65,000 addresses it can assign at its leisure
  - trust me, others are jealous!

IPv4 Address Format
Subnetting

- Often a company/organization will further chop up their address bits into subnets
- This allows routers to efficiently send the packets to another node based on a bit pattern
- Systems use a subnet mask, to divide the host part of the CIDR into smaller parts

And, we ran out again!

- Even using Classless Inter-Domain Routing, IP addresses were becoming hard-to-find
- In fact, we ran out in the 90’s
- Some sort of solution was needed to keep the Internet running before IPv6 would be implemented
- … it was Network Address Translation

Network Address Translation

- How can you have more than one device at home or business use the Internet?
- You could get an IP address for each – but that is expensive!
- And, there simply isn’t enough IPv4 address to make this even possible

Network Address Translation

- NAT (Network Address Translation) is a clever trick where you set up your own private Internet!
- We just need to be able to connect your Internet to the real one

- Allows small business and individual users get Internet access at a low cost
- This is how most “coffee-shop” wireless networks work
This is so cool….

- How it works:
  - computers on the NAT use their own **private** IP addresses
  - only one "real" IP address is needed – it talks to the real Internet
- So…
  - the local network uses one **public** IP address
  - the outside world only sees one!

Maintenance Advantages

- Local network can change **internal** addresses without notifying outside world
- You can change the public address without changing the internal addresses
- Devices inside local network are not explicitly visible by outside world (*a security plus*)

How do they pull this off?

- In each “class”...
  - there is a special **internal use only** address
  - anyone can use these
- These are **not** used on any public computer
  - hence, there is no confusion in the Internal network between the outside world and internal
  - this feature has allowed the Internet stay up long after IPv4 address space was exhausted

IP Addresses Revisited

- Given notion of “NAT”, let’s re-examine IP addresses
- IP Addresses are broken into three classes

<table>
<thead>
<tr>
<th>Main IP Classes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Class A</td>
</tr>
<tr>
<td>Class B</td>
</tr>
<tr>
<td>Class C</td>
</tr>
</tbody>
</table>

IP Addresses Revisited

- Special Internal use only (RFC 1918 Non-routable)
- Also 127.ANY Loopback (127.0.0.1)

<table>
<thead>
<tr>
<th>Internal Use IP Addresses</th>
</tr>
</thead>
<tbody>
<tr>
<td>Class A</td>
</tr>
<tr>
<td>Class B</td>
</tr>
<tr>
<td>Class C</td>
</tr>
</tbody>
</table>

How it Works: Outgoing

- Outgoing packets of data must be modified so the work on the "real" Internet
- Source address (the computer that sent it) is replaced with the shared **NAT IP address**
  - so, Internal address ➞ Shared NAT IP
  - as a result, responses will be sent to the Shared NAT IP
How it Works: Incoming

- Incoming packets of data need to be modified to work on the **internal** Internet
- NAT IP address is replaced by the internal address (of the computer being sent to)
  - So, Shared NAT IP ➔ Internal address
  - Hence, the data reaches the correct internal computer

Example:
(S = Sender, D = Destination)

(S = 130.86.12.66)
(D = 218.76.29.7)

130.86.12.66

218.76.29.7

10.0.0.1
10.0.0.2
10.0.0.3