COMP2221 Networks

David Head

University of Leeds

Lecture 19

Previous lectures

In the last three lectures we have looked at the layers below the Application layer:

- **Transport** layer, TCP and UDP, connection management and congestion control.
- In the last two lectures we have looked at the **Network layer** (also known as the IP or Internet layer):
  - How data packets are **forwarded** at routers.
  - The various **routing algorithms** to try and send packets efficiently from source to destination.

Today's lecture

In today's lecture we will look at the bottom two layers:

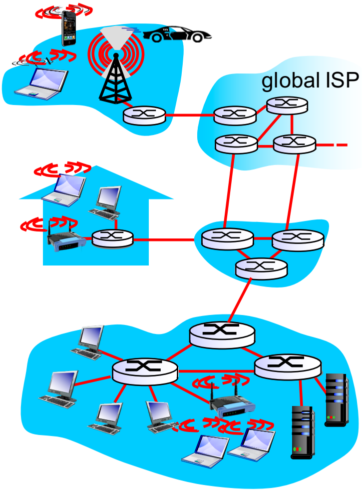
- The **Link layer**, also known as the data link layer.
- **Ethernet** protocols and **MAC addresses**.
- How multiple devices can share the same channel.
- The bottom-most **Physical layer** (very briefly).

This is the last lecture of new material for this module.

Link layer: Terminology

- **Node**: Host or router.
- **Link**: Communication channel that connects **adjacent** nodes.
- Links can be **wired** or **wireless**.
- Also have **LANs** = Local Area Networks.
- Data packets are called **frames**.

The **Link layer** is responsible for transferring data from one node to a **physically adjacent** node.



Link layer analogy

Packet/frame transferred by different link protocols over different links.

- *e.g.* ethernet on first link, frame relays on intermediate links, 802.11 (WiFi) on last link.

Each link protocol provides different services.

- *e.g.* May or may not provide reliable data transfer.

Transportation analogy: Trip from Leeds to Lausanne.

- Taxi from Leeds to Airport (LBA).
- Plane from LBA to Geneva.
- Train from Geneva to Lausanne.

Traveller = packet/frame.
Transport segment = communication **link**.
Transportation mode = **link layer** protocol.
Travel agent = routing algorithm.

Link layer services (1)

Framing and link access:

- Encapsulate the datagram/packet into a **frame**, adding a header and possibly a trailer/footer.
- **Channel access** if the medium is **shared**.
- MAC addresses used in frame headers — **different from IP addresses**.

Reliable delivery between adjacent nodes (possibly)

- Similar strategy to TCP.
- Seldom used on **wired links** because of the low error rate.
- More important for **wireless links** with high error rates.

Link layer services (2)

Flow control:

- Pacing between adjacent sending and receiving nodes.

Error detection:

- Errors caused by signal attenuation and/or noise.
- Receivers detect presence of errors; signals sender to retransmit or drops frame.

Error correction:

- Receiver identifies **and corrects** bit error(s) using a **checksum**, without requiring retransmission.

Half-duplex and full-duplex:

- Half-duplex means nodes at both ends of a link can transmit, but not at the same time.

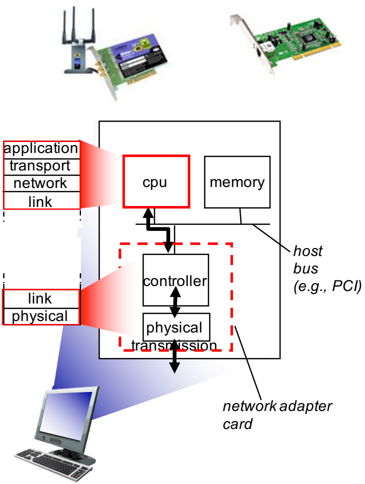
Where is the Link layer implemented?

Link layer implemented in each host.

- **Adaptor** (NIC = Network Interface Card) or **chip**.
- *e.g.* Ethernet card, 802.11 card, Ethernet chipset.

Attaches to the host's system buses.

Combination of hardware, software and firmware.



Multiple access links

Two types of link:

Point-to-point:

- *e.g.* dial-up access, point-to-point link between Ethernet switch and a host.

Broadcast:

- *e.g.* Ethernet, 802.11 Wireless LAN ('Wi-Fi').

shared wire (e.g., cabled Ethernet)

shared Radio Frequency (e.g., 802.11 WiFi)

shared Radio Frequency (satellite)

humans at a cocktail party (shared air, acoustical)

Multiple access protocols

Single, **shared** access channel.

- Two or more simultaneous transmissions may interfere.
- **Collision** if node receives two or more signals at once.

Need a **multiple access protocol**:

- Distributed algorithm that determines how nodes share the channel.
  - *i.e.* when a node can transmit.
- Coordination needed for better channel sharing and communication.

Types of Multiple Access Protocol

Channel partitioning:

- Divide channel into 'pieces' (time slots; frequency; code).
- Allocate pieces for node for **exclusive** use.
- *e.g.* TDMA = Time Division Multiple Access.

Random access:

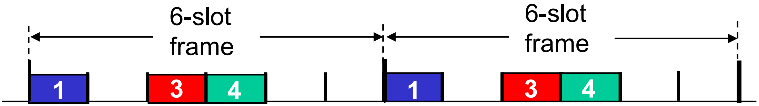
- Randomise send times to minimise chances of collision.
- If collision detected, **recover** (*i.e.* transmit).
- *e.g.* CSMA = Carrier Sends Multiple Access, and variations.

'Taking turns':

- Nodes take turns, but nodes with more to send can take longer turns (coordinate using *e.g.* tokens).

TDMA = Time Division Multiple Access

- Form of **channel partitioning** in which each node gets a fixed length **time slot**.
- Example for a 6-node LAN.



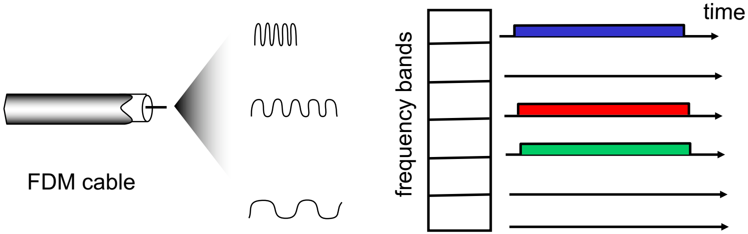
Problem: Unused slots go idle

Not an efficient use of available bandwidth.

In addition, will normally have to **wait** to **start** communicating.

FDMA = Frequency Division Multiple Access

- Alternative form of **channel partitioning**, this time dividing into **frequency bands**.
- Each node assigned **one band**.
- 6-node LAN example:



Problem: Each node still has limited bandwidth

Similar advantages and disadvantages as TDMA.

Random access protocol: Slotted ALOHA

Basic idea

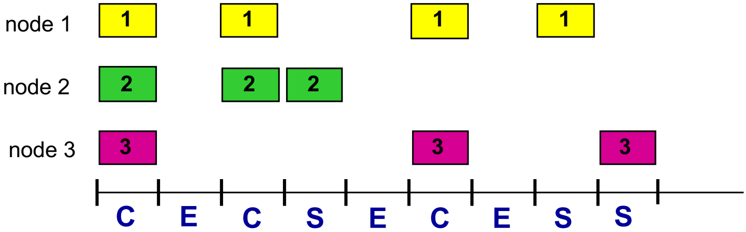
Start sending immediately. If hardware detects a **collision**, resend **after a random time interval**. Repeat if necessary.

For the next slide, have assumed:

- All frames equal size.
- Time divided into **slots**.
- Can only start transmission at the **start** of a slot.
- All nodes can detect **collisions**.
- When there is a collision, re-send in each subsequent slot with probability p, until successful.
- $0 < p < 1$.

Slotted ALOHA: Example

3 nodes start transmitting in the same slot. A possible outcome is:



[Key: **C**=Slot with a collision, **E**=Empty slot, **S**=Successful transfer]

- If only one node is communicating, it uses **full bandwidth**.
- But **collisions** result in **wasted slots**.
- Frames eventually sent because re-transmission is **random**.

Slotted ALOHA: Efficiency

Suppose N nodes are **all** trying to transmit **many** frames:

- Each transmits in a slot with probability p.
- Probability of success is $p \times (1-p)^{N-1}$, *i.e.* p that one node transmits, and $1-p$ for each remaining node to **not** transmit.
- Probability that **any** node has success is this times N, *i.e.*

$$Np(1-p)^{N-1} \quad .$$

- Can show maximum efficiency realised for $p^* = 1/N$.
- Can show this maximum efficiency is $1/e \approx 37\%$ for large N.

Other protocols can achieve higher efficiencies.

MAC Addresses

Whatever the protocol, each node must have a unique **MAC** (Media Access Control) address.

Consider the 32/128-bit IP address:

- Network layer address, used for forwarding.

The MAC (or LAN, physical, Ethernet) address:

- Used 'locally' to get frame from one interface to another.
- Both interfaces in the same network (in IP sense).
- 48-bit MAC addresses burned into NIC ROM.
- *e.g.* `1A-2F-BB-76-09-AD`.

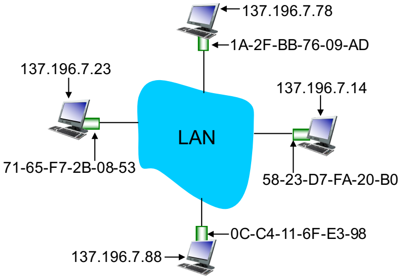
MAC Addresses

- MAC address allocation administered by IEEE.
- Manufacturer buys a range of MAC addresses — **unique**.
- Analogy:
  - MAC address like a **NI/social security number**.
  - IP address is more like a **postal address**.
- Flat (*not* hierarchical), for portability ...
  - Move device from one LAN to another.
- ...unlike IP's hierarchical addressing.
  - IP address depends on subnetwork to which the node is attached.

ARP: Address Resolution Protocol

Question:

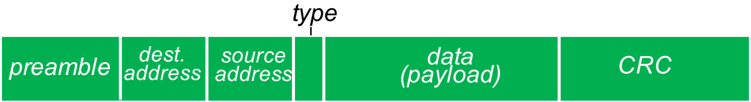
How to determine an interface's MAC address knowing its IP address?



ARP tables:

- Each IP node (host, router) has an ARP table.
- Contains IP/MAC address mappings for nodes.
- Updated dynamically.

Ethernet frame structure



Header and trailer fields:

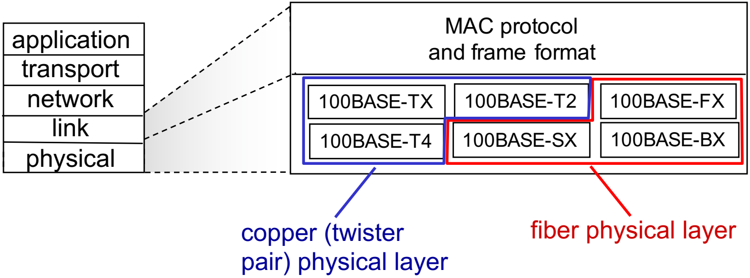
- **Preamble** has fixed bit pattern, for synchronisation.
- **Destination** and **source** MAC addresses.
- **Type** usually IP, but can be another higher-level protocol.
- **CRC** = Cyclic Redundancy Check for detecting errors.

Ethernet is **connectionless** and **unreliable**.

Ethernet standards

There are **many** different Ethernet standards.

- Common MAC protocol and frame format.
- Different speeds: 2 Mbps, 10 Mbps, 100 Mbps, 1 Gbps, 10 Gbps.
- Different Physical layer media: fibre, cable.



Physical layer

The final layer is the **Physical layer**, which is responsible for moving individual bits between nodes.

- Different media (*e.g.* twisted-pair copper, fibre optics) have different characteristics (frequency, signal decay).
- Therefore also have different Ethernet protocols.

Wireless communication has high error rates and multiple access.

- Also has many different protocols.
- *i.e.* IEEE 802.11 (‘WiFi’), IEEE 802.15.1 (‘Bluetooth’), IEEE 802.15.4 (‘Zigbee’; ‘Internet of things’).

Overview and next lecture

Today we have looked at the bottom two levels:

- The **Link** layer, responsible for transferring data between adjacent nodes.
- Unique **MAC** address for each network interface.
- **Many different protocols**, as they only need to be adhered to locally.
- The **Physical** layer, responsible for moving individual bits.

The next lecture will be the final one, where we will summarise what we have learned.