University of Glasgow

School of Computing Science

The Physical and Data Link Layers

Networked Systems (H)

Lecture 2

Lecture Outline

- The physical layer
  - Wired links
  - Wireless links
  - Channel capacity
- The data link layer
  - Addressing
  - Framing
  - Error detection and correction
  - Media access control

The Physical Layer

The Physical Layer

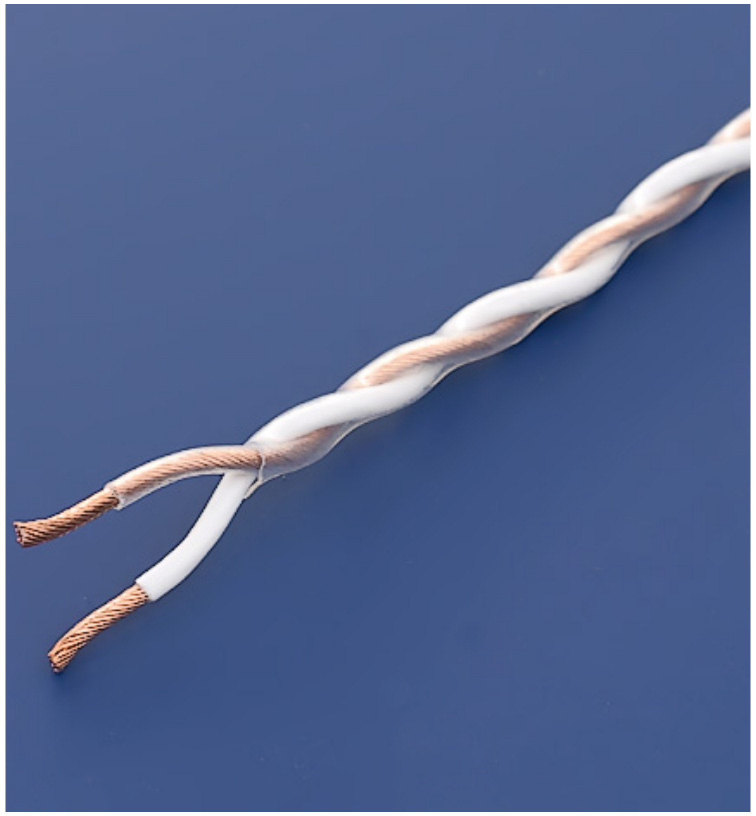
- The physical layer is concerned with transmission of raw data bits
  - What type of cable or wireless link do you use?
  - How to encode bits onto that channel?
    - Baseband encoding
    - Carrier modulation
  - What is the capacity of the channel?

Wired Links

- Physical characteristics of cable or optical fibre:
  - Size and shape of the plugs
  - Maximum cable/fibre length
  - Type of cable: electrical voltage, current, modulation
  - Type of fibre: single- or multi-mode, optical clarity, colour, power output, and modulation of the laser

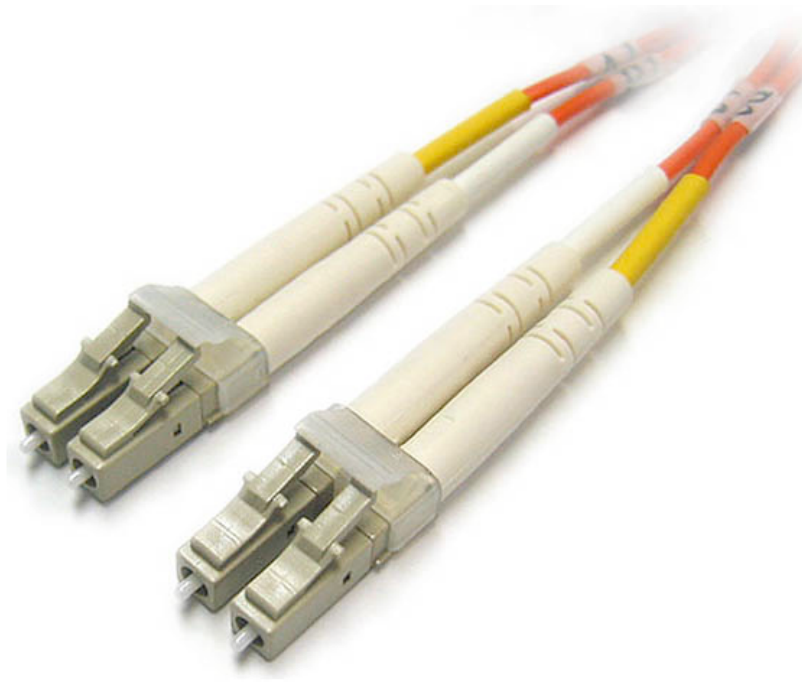
Unshielded Twisted Pair

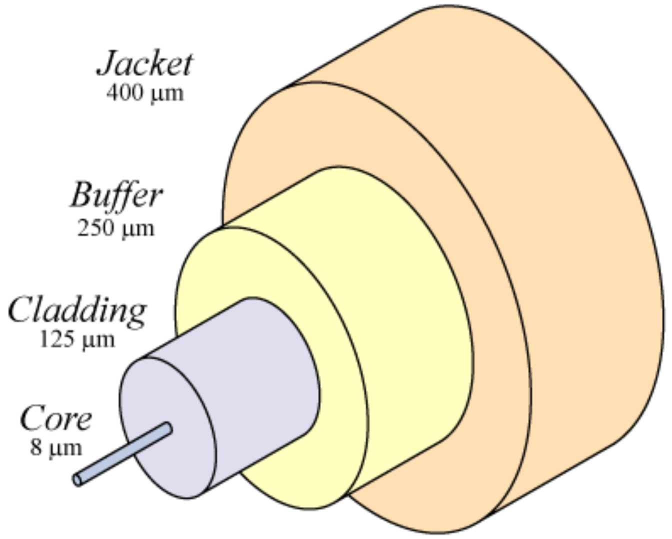
- Electrical cable using two wires twisted together
  - Each pair is unidirectional: signal and ground
  - Twists reduce interference and noise pickup: more twists → less noise
  - Cable lengths of several miles possible at low data rates; ~100 metres at high rates
  - Susceptibility to noise increases with cable length
  - Extremely widely deployed:
    - Ethernet cables
    - Telephone lines



Optical Fibre

- Glass core and cladding, contained in plastic jacket for protection
  - Somewhat fragile: glass can crack if bent sharply
  - Unidirectional data: transmission laser at one end; photodetector at the other
    - Laser light trapped in fibre by total internal reflection
  - Very low noise, since electromagnetic interference does not affect light
  - Very high capacity: 10s of Gbps over 100s of miles
  - Very cheap to manufacture
  - Requires relatively expensive lasers to operate

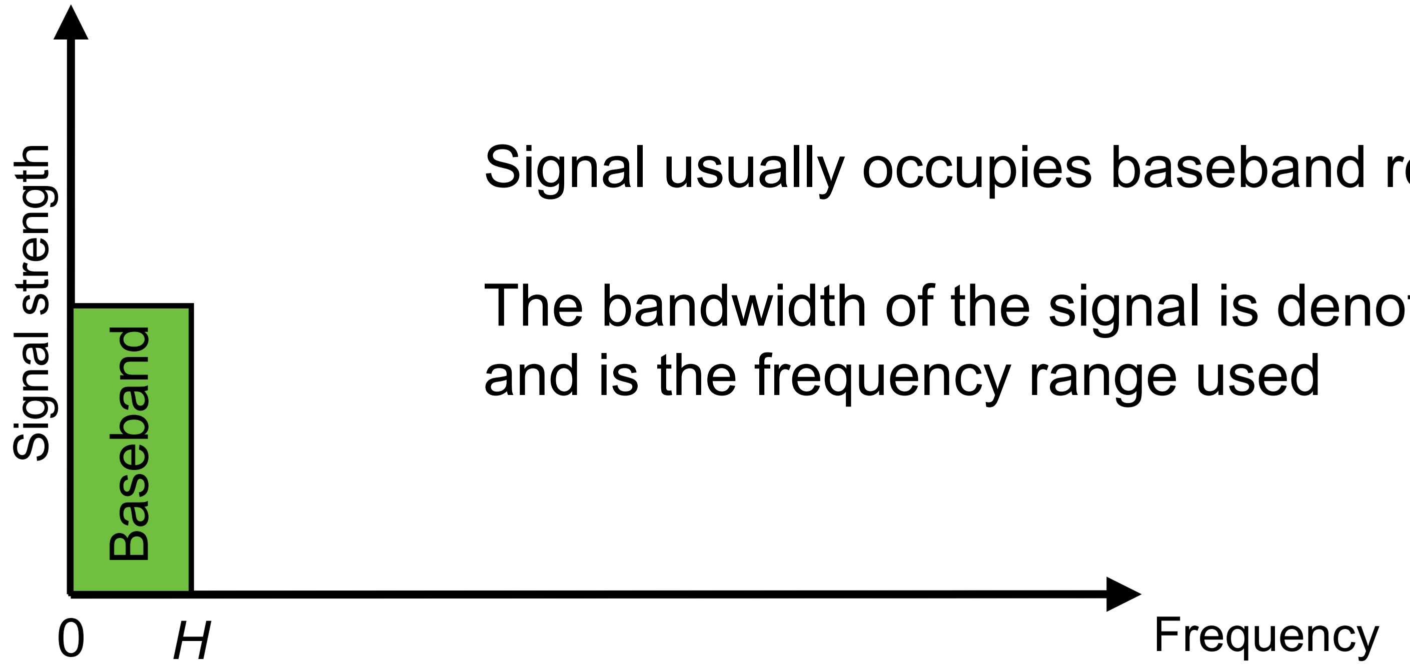




Source: Wikipedia/Bob Mellish/CC BY-SA 3.0

Wired Data Transmission

- Signal usually directly encoded onto the channel



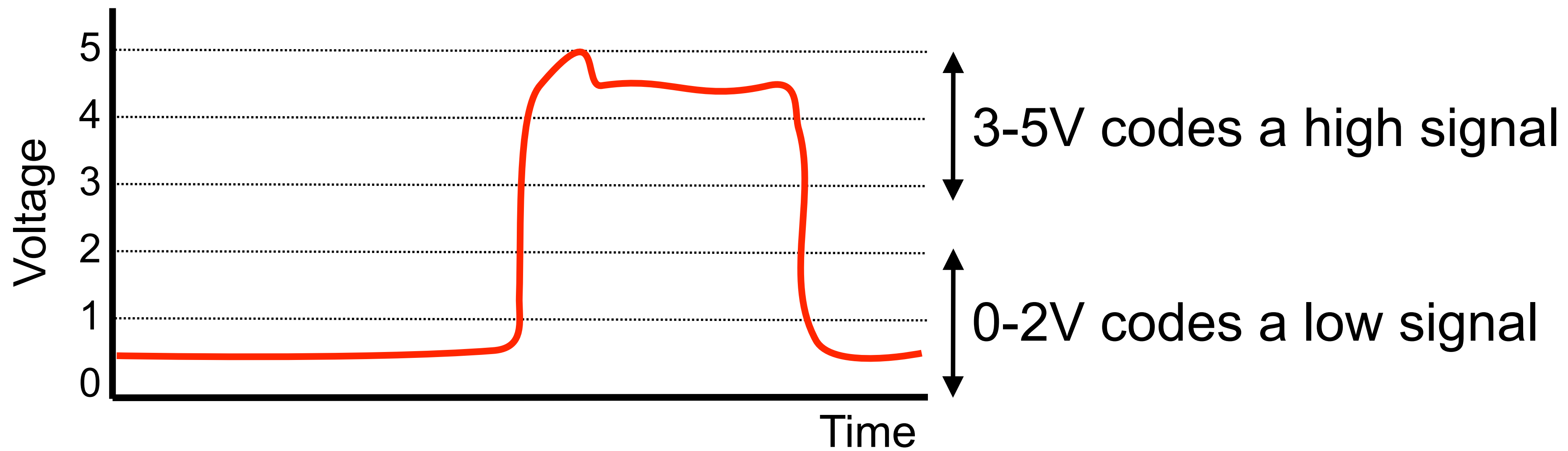
Signal usually occupies baseband region

The bandwidth of the signal is denoted H, and is the frequency range used

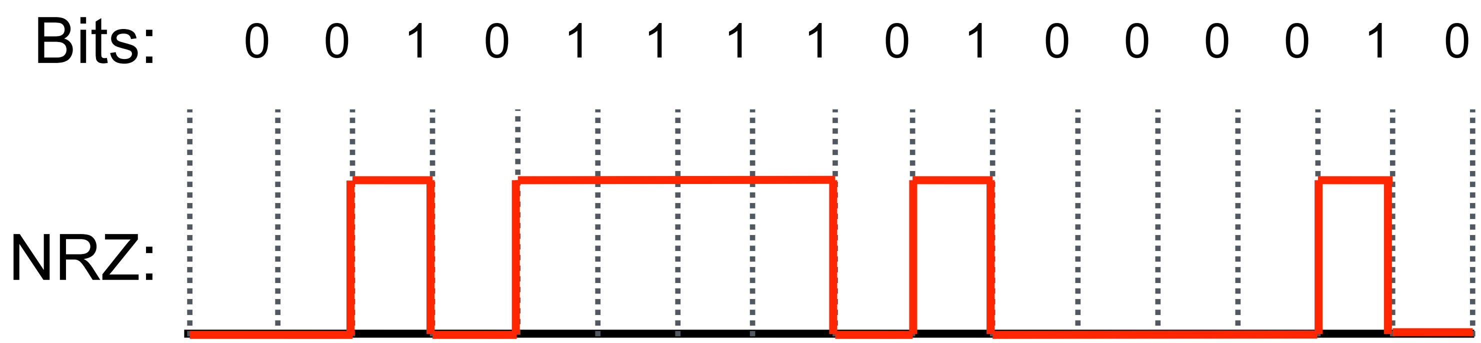
- Encoding performed by varying the voltage in an electrical cable, or the intensity of light in an optical fibre
- Many encoding schemes exist: NRZ, NRZI, Manchester, 4B/5B, etc.

Baseband Data Encoding

- Encode the signal as change in voltage applied to cable, or change in brightness of laser in optical fibre
- Example:

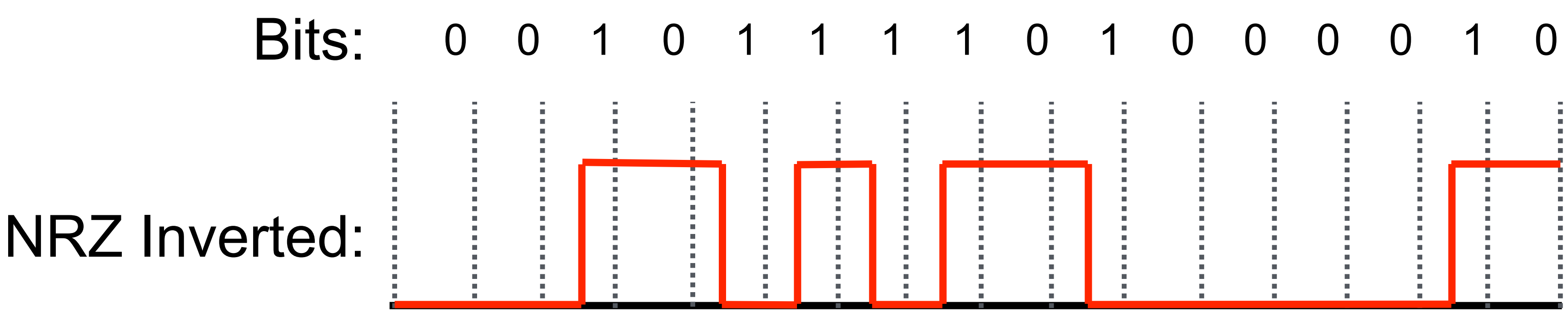


Baseband Data Encoding



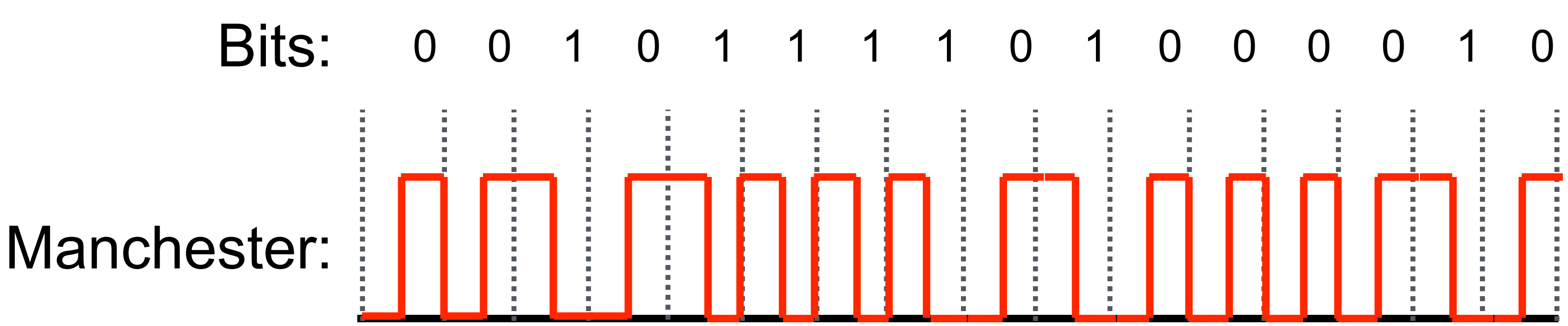
Encodes a 1 as a high signal, a 0 as a low signal

Runs of the same value → clock skew and baseline wander



Encode a 1 as a change in signal value, a 0 as a constant signal

Solves problem with consecutive 1s, not runs of consecutive 0s



Encode a 1 as high-low transition, a 0 as a low-high transition

Doubles the bandwidth needed, but avoids problems with NRZ encoding

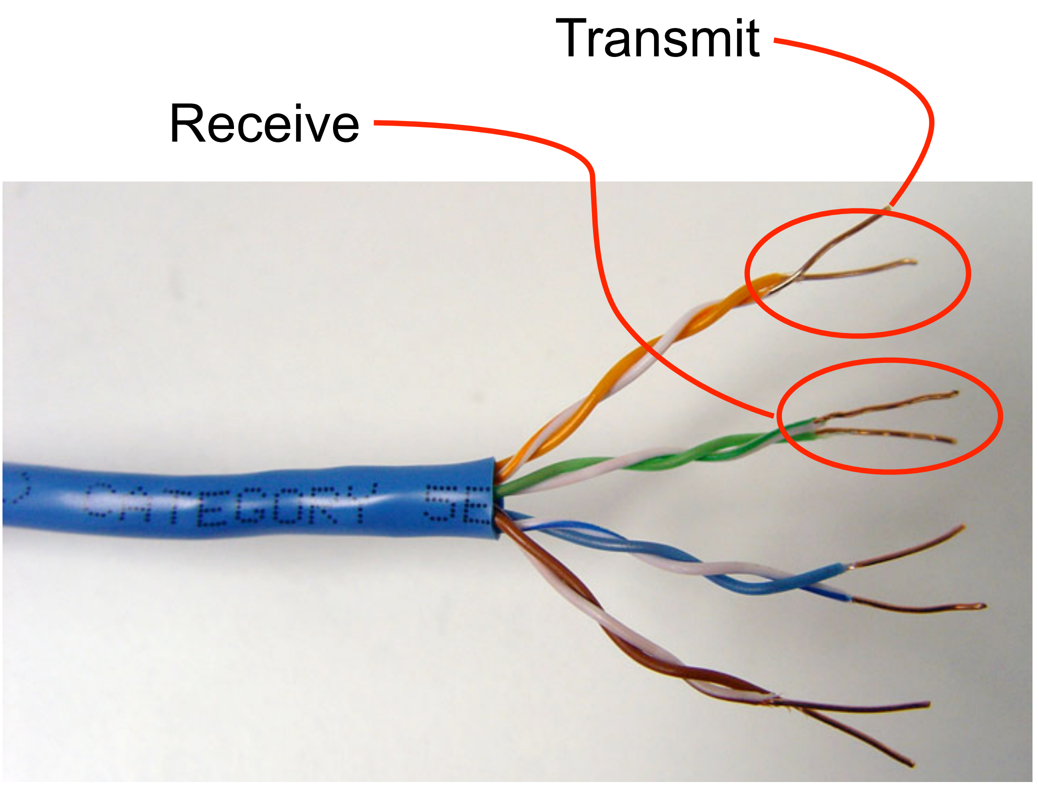
4B/5B Encoding

4-Bit Data Symbol	5-Bit Encoding
0000	11110
0001	01001
0010	10100
0011	10101
0100	01010
0101	01011
0110	01110
0111	01111
1000	10010
1001	10011
1010	10110
1011	10111
1100	11010
1101	11011
1110	11100
1111	11101

- Manchester encoding inefficient – only 50% of link capacity used
- Alternative – insert extra bits to break up sequences of same bit
  - Each 4 bit data symbol is changed to a 5 bit code for transmission; reversed at receiver
  - Transmit 5 bit codes using NRZI encoding
  - 80% of link capacity used for data

Example: Ethernet

- Baseband data with Manchester coding at 10 Mbps, or 4B/5B coding at 100 Mbps



4 twisted pairs per cable

3 twists per inch

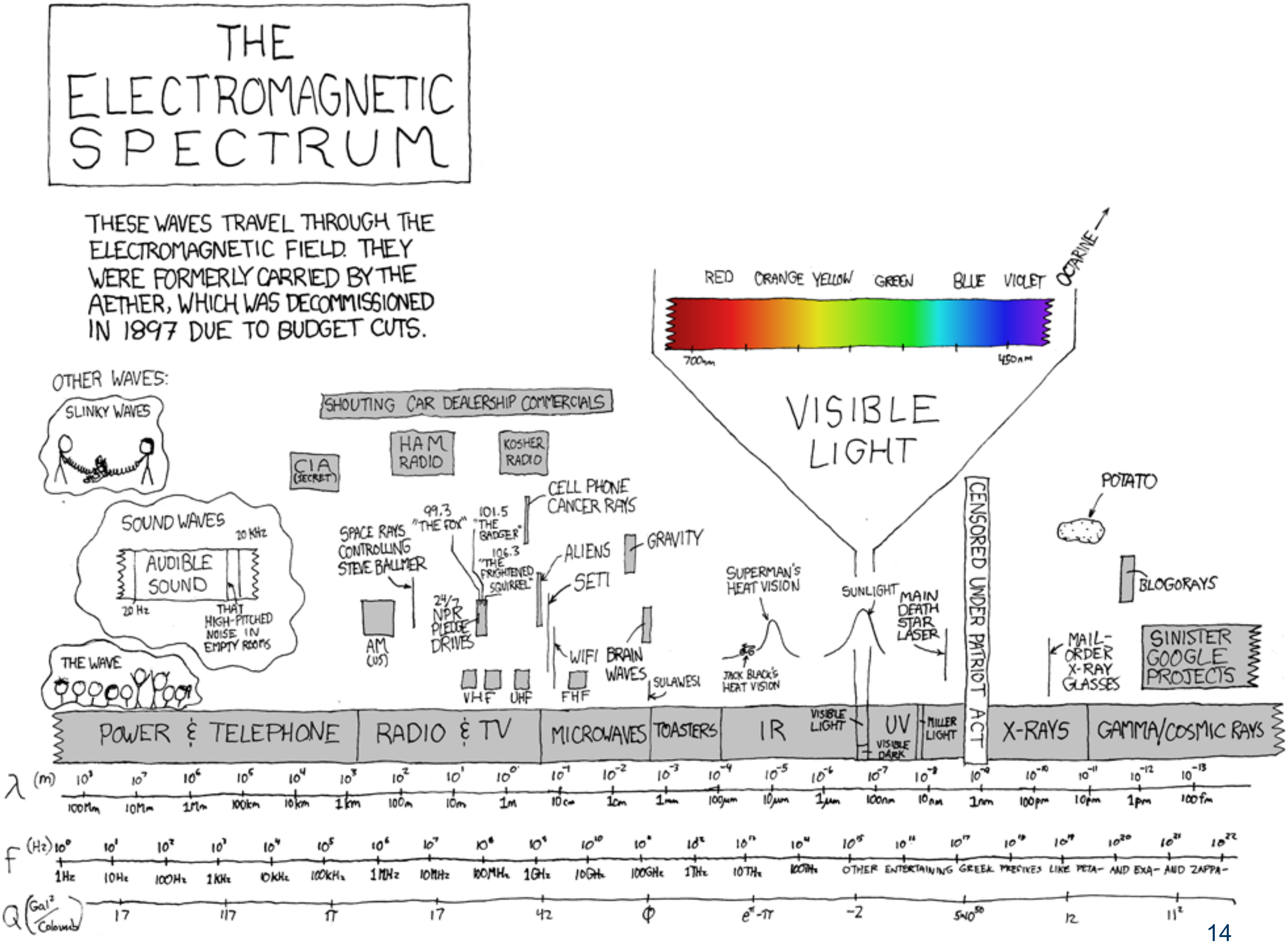
24 gauge (~0.5mm) copper

100m maximum cable length

Wireless Links

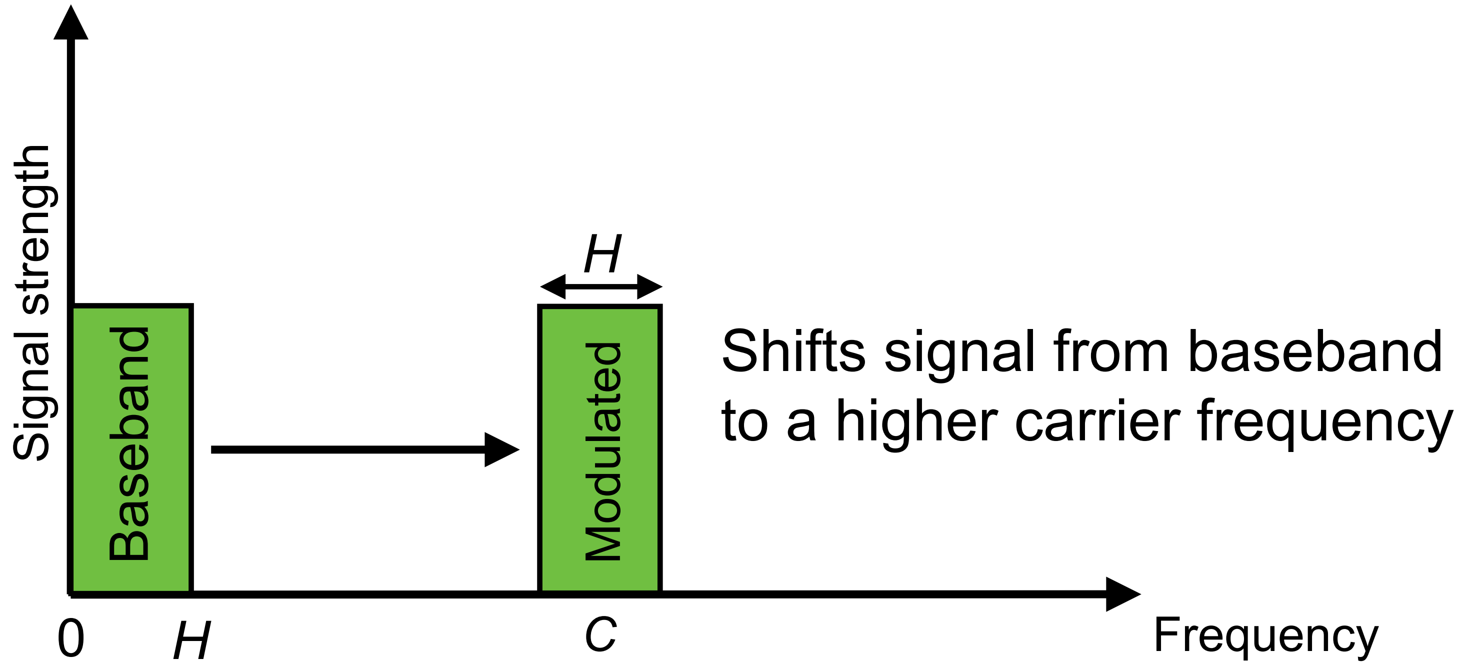
- Wireless links use carrier modulation, rather than baseband transmission
- Performance affected by:
  - Carrier frequency
  - Transmission power
  - Modulation scheme
  - Type of antenna, etc.

Source: http://xkcd.com/273/



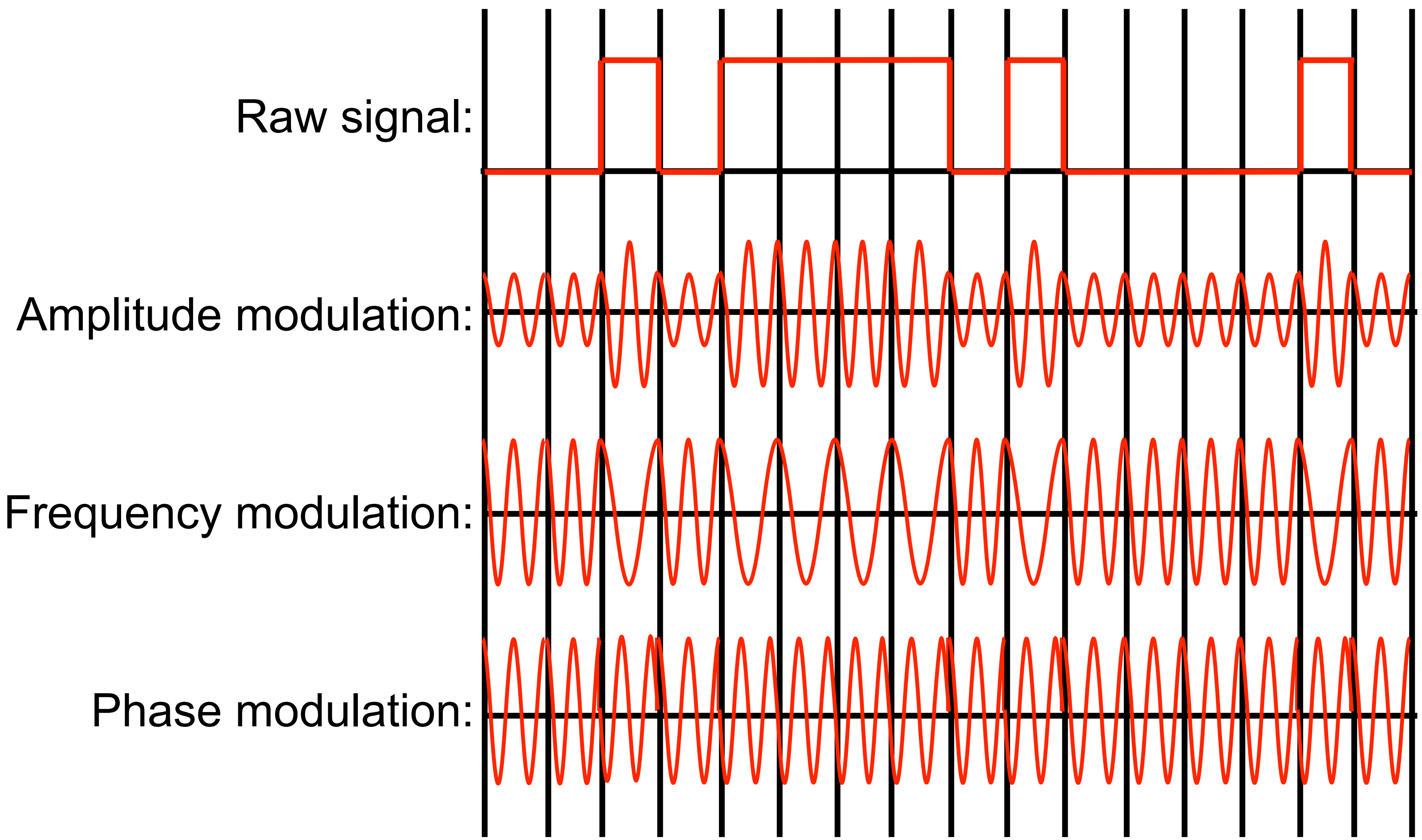
Carrier Modulation

- Carrier wave applied to channel at frequency, C
- Signal modulated onto the carrier



- Allows multiple signals on a single channel
  - Provided carriers spaced greater than bandwidth, *H*, of the signal
  - Usually applied to wireless links, but can be used on wired links – this is how ADSL and voice telephones share a phone line

Amplitude, Frequency, Phase Modulation



Complex Modulations

- More complex modulation schemes allow more than one bit to be sent per baud
  - Use multiple levels of the modulated component
    - Example: gigabit Ethernet uses amplitude modulation with five levels, rather than binary signalling
  - Combine modulation schemes
    - Vary both phase and amplitude → quadrature amplitude modulation
    - Example: 9600bps modems use 12 phase shift values at two different amplitudes
- Extremely complex combinations regularly used

Spread Spectrum Communication

- Single frequency channels prone to interference
  - Mitigate by repeatedly changing carrier frequency, many times per second: noise unlikely to affect all frequencies
  - Use a pseudo-random sequence to choose which carrier frequency is used for each time slot
  - Seed of pseudo-random number generator is shared secret between sender and receiver, ensuring security

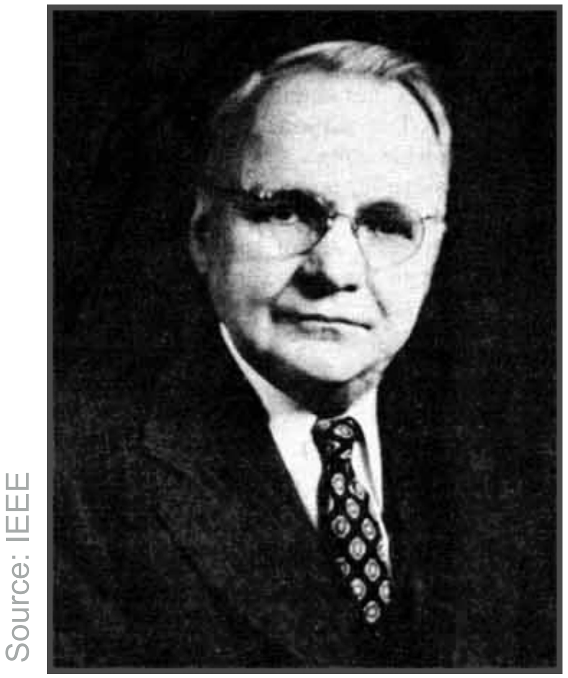
Source: (Wikipedia/Public Domain)

Hedy Lamarr (1914-2000)

  - Example: 802.11b Wi-Fi uses spread spectrum using several frequencies centred ~2.4 GHz with phase modulation

Bandwidth and Channel Capacity

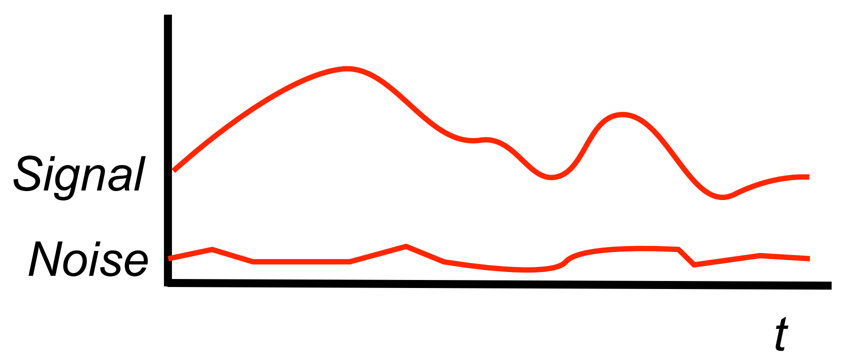
- The bandwidth of a channel determines the frequency range it can transport
  - Fundamental limitations based on physical properties of the channel, design of the end points, etc.
- What about digital signals?
  - Sampling theorem: to accurately digitise an analogue signal, need $2H$ samples per second
  - Maximum transmission rate of a digital signal depends on channel bandwidth: $R_{max} = 2H \log_2 V$
    - R_{max} = maximum transmission rate of channel (bits per second)
    - H = bandwidth
    - V = number of discrete values per symbol
  - Assumption: perfect, noise-free, channel



Harry Nyquist (1889-1976)

Noise

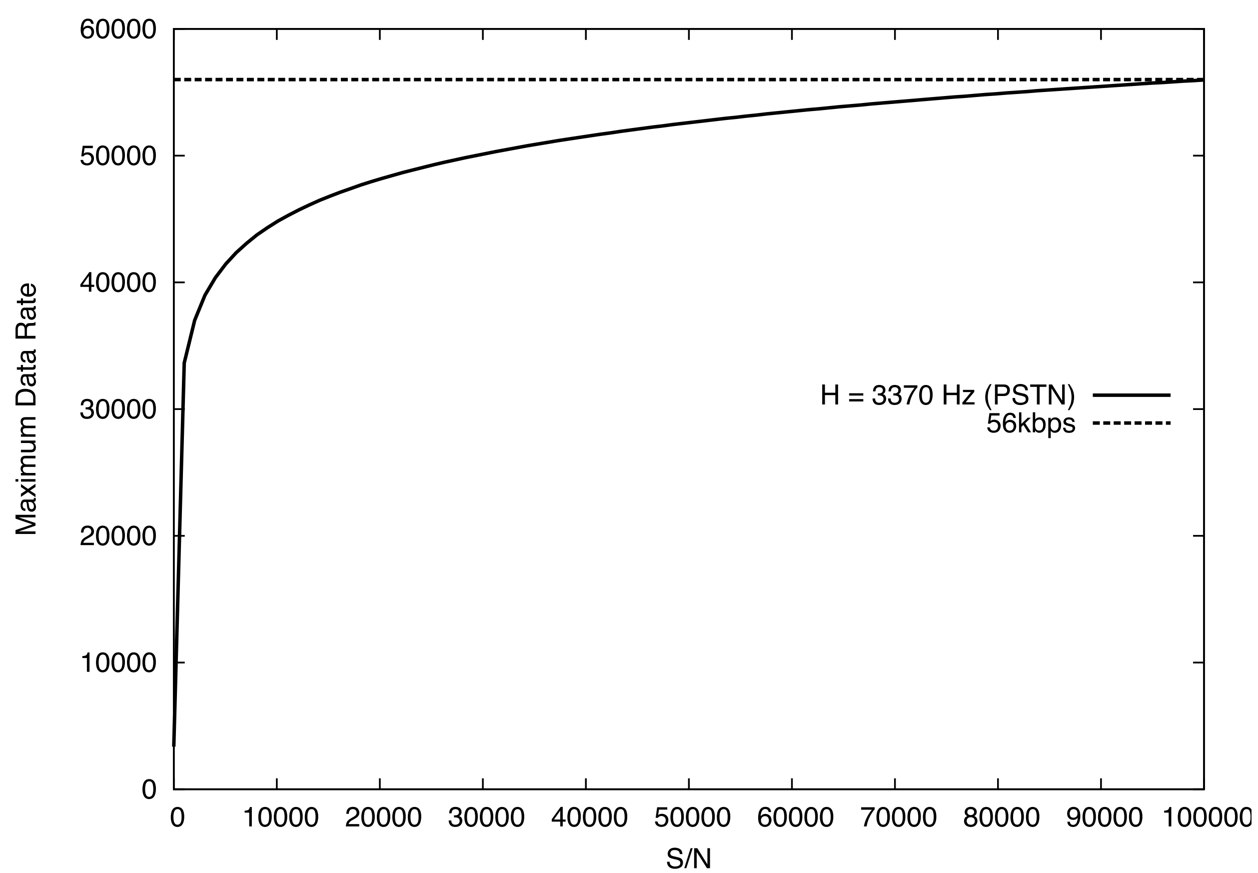
- Real world channels are subject to noise that corrupts the signal
  - Electrical interference
  - Cosmic radiation
  - Thermal noise
- Can measure signal power, *S*, and noise floor, *N*, in a channel
  - Gives signal-to-noise ratio: S/N
    - Typically quoted in decibels (dB), not directly
    - Signal-to-noise ratio in dB = 10 log10 S/N
    - Example: ADSL modems report S/N ~30 for good quality phone lines: signal power 1000x greater than noise

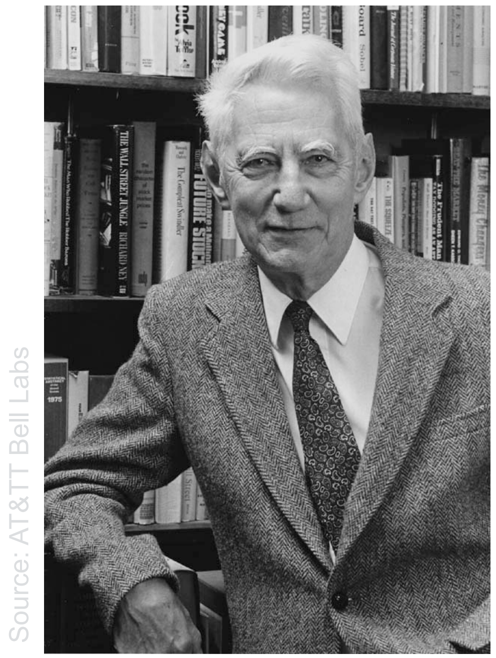


S/N	dB
2	3
10	10
100	20
1000	30

Capacity of a Noisy Channel

- Capacity of noisy channel depends on type of noise
  - Uniform or bursty; affecting all or some frequencies
  - Simplest to model is Gaussian noise: uniform noise that impacts all frequencies equally
  - Maximum transmission rate of a channel subject to Gaussian noise: $R_{max} = H \log_2(1 + S/N)$
- Example: dial-up modem bandwidth limitation





Claude Shannon (1916-2001)

Implications

- Physical characteristics of channel limit amount of information that can be transferred
  - Bandwidth
  - Signal to noise ratio
- These are fundamental limits: might be reached with careful engineering, but cannot be exceeded

The Data Link Layer

Purpose of Data Link Layer

- Arbitrate access to the physical layer
  - Identify devices – addressing
  - Structure and frame the raw bitstream
  - Detect and correct bit errors
  - Control access to the channel – media access control
- Turn the raw bit stream into a structured communications channel

Addressing

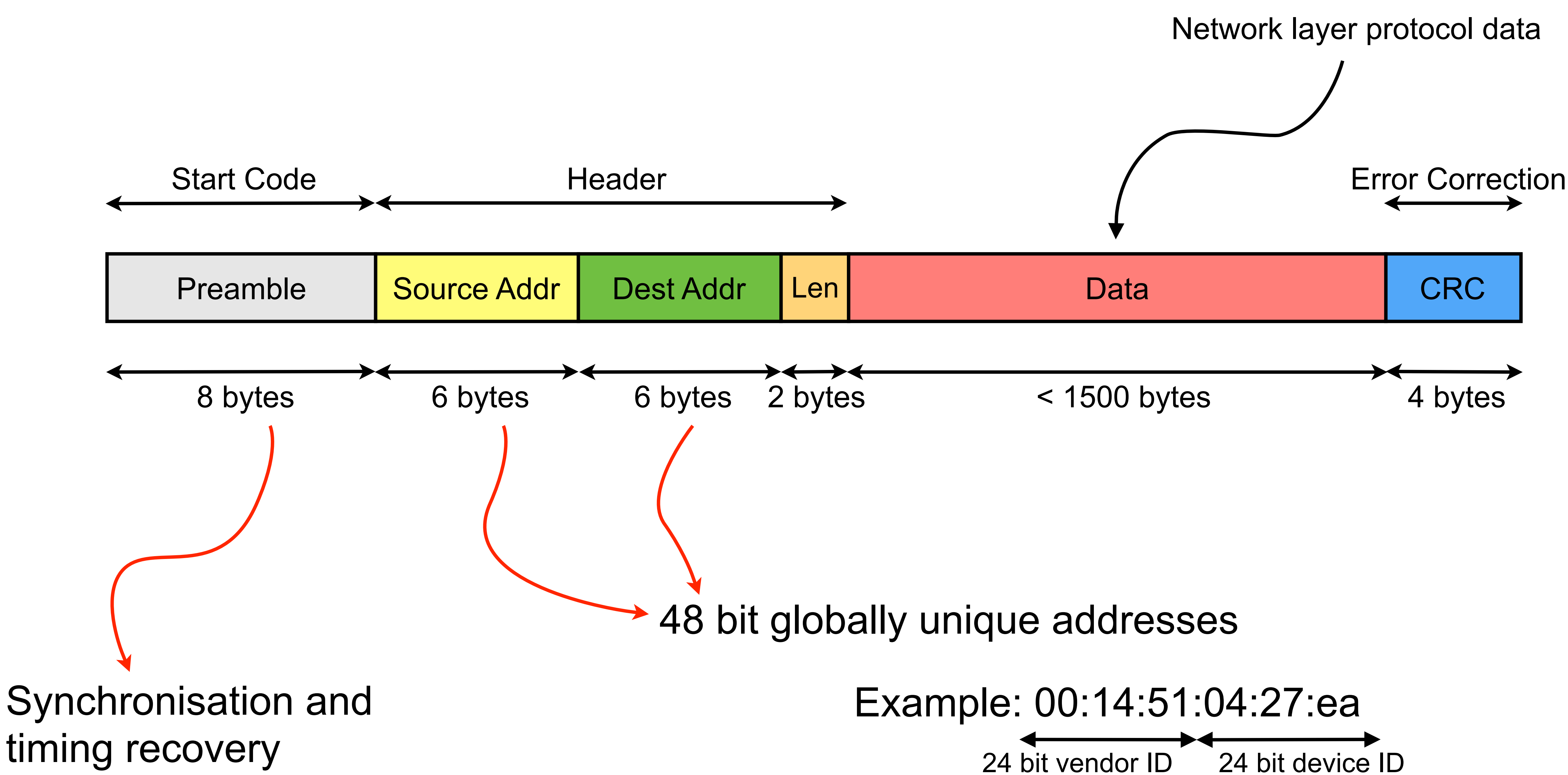
- Physical links can be point-to-point or multi-access
  - Wireless links are common example of multi-access, but several hosts can also be connected to a single cable to form multi-access wired link
  - Multi-access links require host addresses, to identify senders and receivers

- Host addresses may be link-local or global scope
  - Sufficient to be unique only amongst devices connected to a link
    - But needs coordination between devices to assign addresses
  - Many data link layer protocols use globally unique addresses
    - Examples: Ethernet and IEEE 802.11 Wi-Fi
    - Simpler to implement if devices can move, since don't need to change address when connected to a different link – privacy concerns

Framing and Synchronisation

- Physical layer provides unreliable raw bit stream
  - Bits might be corrupted
  - Timing can be disrupted
- Data link layer must correct these problems
  - Break the raw bit stream into frames
  - Transmit and repair individual frames
  - Limit scope of any transmission errors

Example: Ethernet

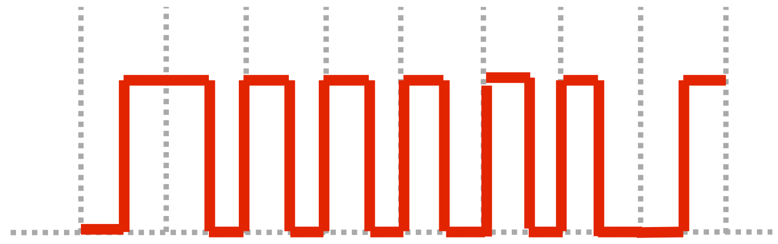


Synchronisation (1)

- How to detect the start of a message?
  - Leave gaps between frames
    - Problem – physical layer typically doesn't guarantee timing (clock skew, etc.)
  - Precede each frame with a length field
    - What if that length is corrupted? How to find next frame?
  - Add a special *start code* to beginning of frame
    - A unique bit pattern that only occurs at the start of each frame
    - Enables synchronisation after error – wait for next start code, begin reading frame headers

0 1 1 1 1 1 1 0

Start code should generate a regular pattern after physical layer coding

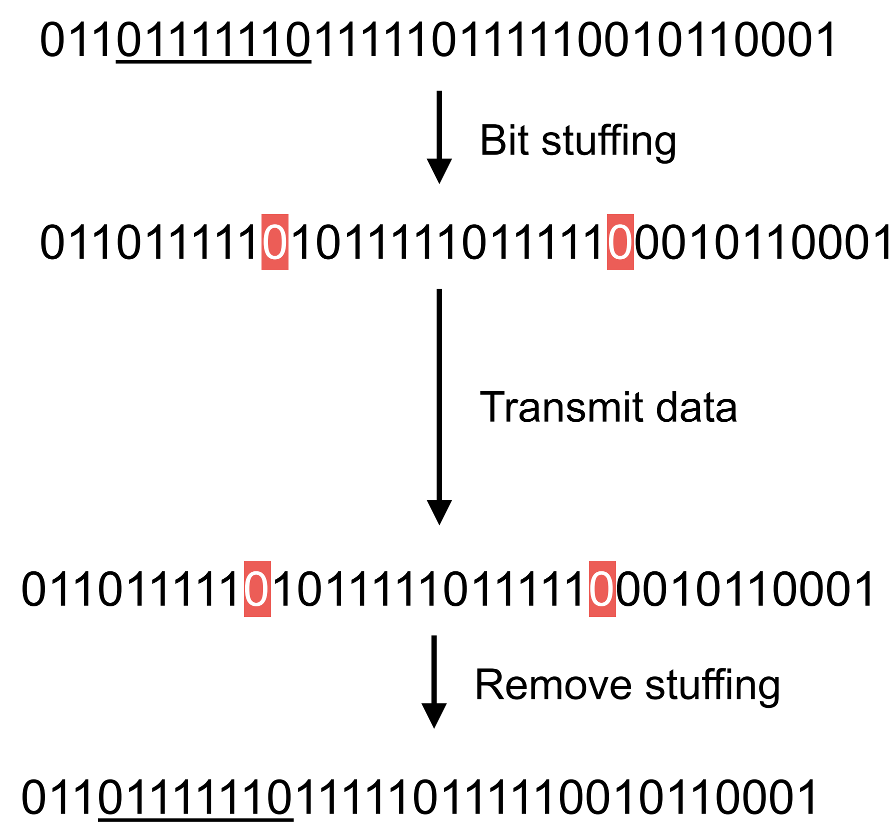


Manchester Encoding

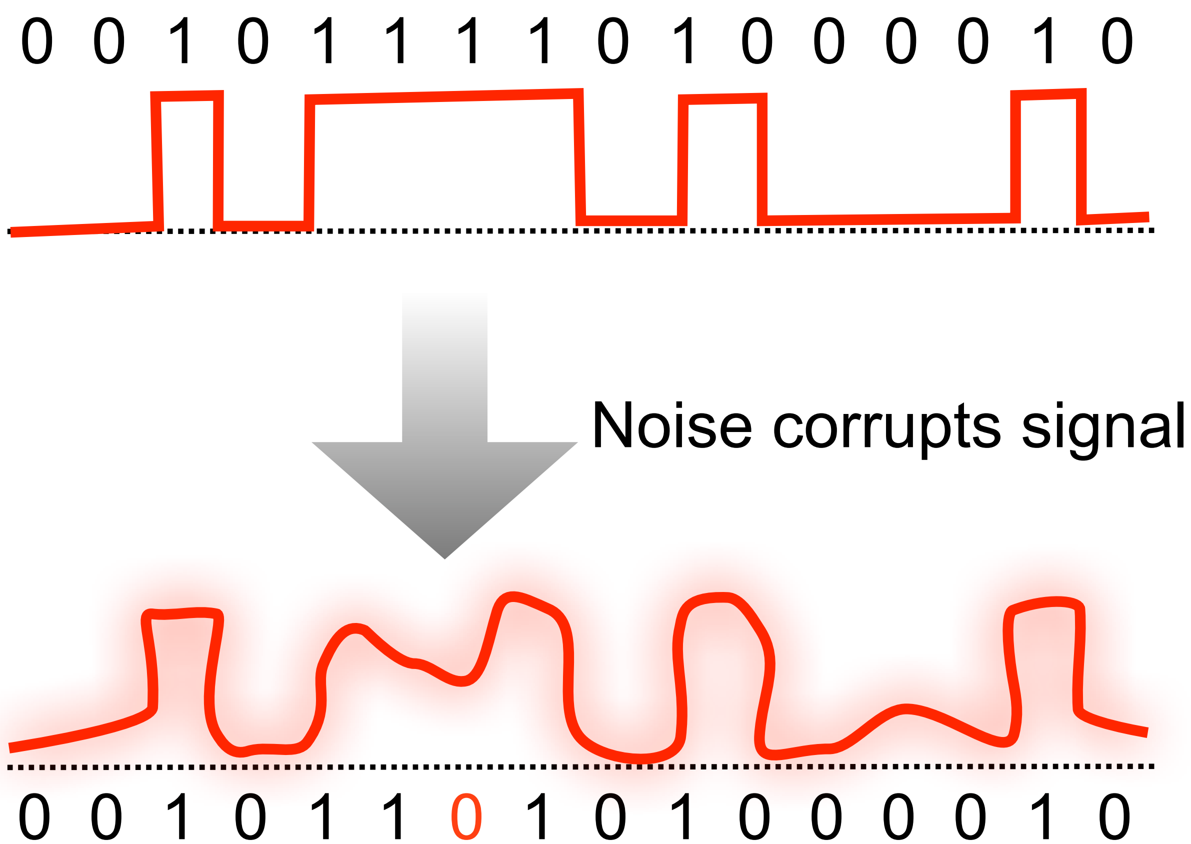
Receiver measures timing

Synchronisation (2)

- What if start code appears in data? *Bit stuffing* can give a transparent channel
- Sender inserts a 0 bit after sending any five consecutive 1 bits – unless sending start code
- If receiver sees five consecutive 1 bits, look at sixth bit:
  - If 0, has been stuffed, so remove
  - If 1, look at seventh bit:
    - If 0, start code
    - If 1, corrupt frame
- A binary-level escape code



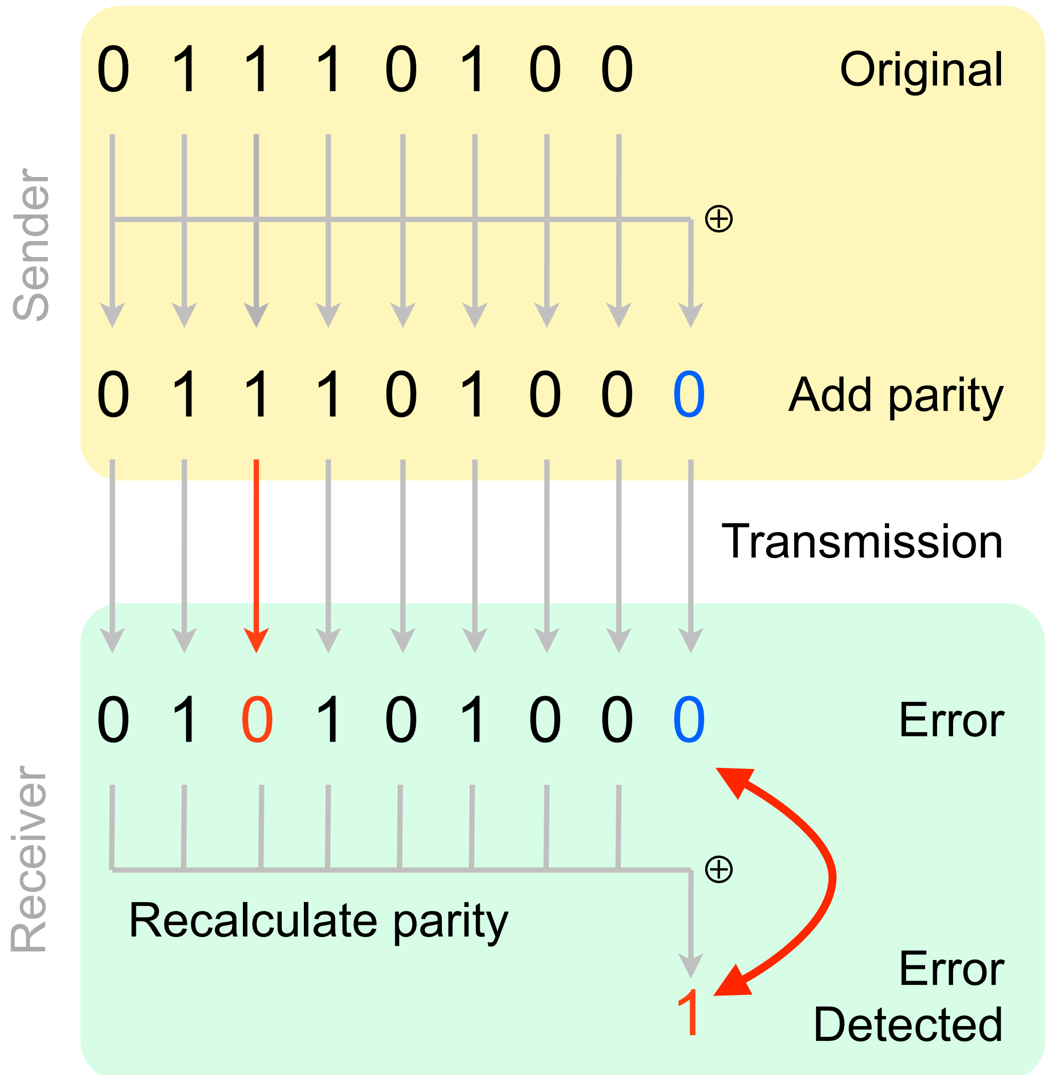
Error Detection



- Noise and interference at physical layer can cause bit errors
  - Rare in wired links, common in wireless systems
- Add *error detecting code* to each packet

Example: Parity Codes

- Simplest error detecting code
- Calculate *parity* of the data
  - How many 1 bits are in the data?
  - An odd number → parity 1
  - An even number → parity 0
  - Parity bit is the XOR ("⊕") of data bits
- Transmit parity with the data, check at receiver
  - Detects all single bit errors



Example: The Internet Checksum

```
#include <stdint.h>

// Internet checksum algorithm. Assumes
// data is padded to a 16-bit boundary.
uint16_t
internet_cksum(uint16_t *buf, int buflen)
{
    uint32_t sum = 0;

    while (buflen--) {
        sum += *(buf++);
        if (sum & 0xffff0000) {
            // Carry occurred, wrap around
            sum &= 0x0000ffff;
            sum++;
        }
    }
    return ~(sum & 0x0000ffff);
}
```

- Sum data values, send *checksum* in each frame
    - Internet protocol uses a 16 bit ones complement checksum
- Receiver recalculates checksum, a mismatch → bit error occurred
- More effective than parity codes – can detect some multiple bit errors

Other Error Detecting Codes

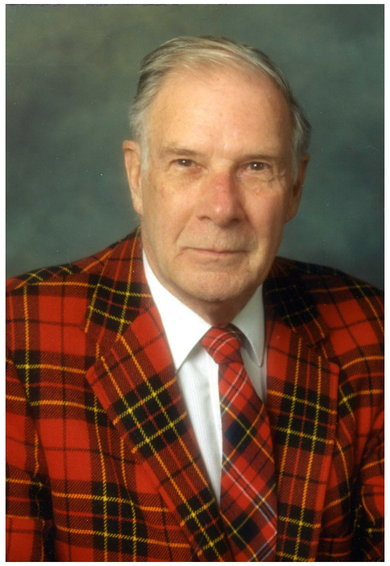
- Parity codes and checksums relatively weak
  - Simple to implement
  - Undetected errors reasonably likely
- More powerful error detecting codes exist
  - Cyclic redundancy code (CRC)
  - More complex → fewer undetected errors
  - (see recommended reading for details)

Error Correction

- Extend error detecting codes to correct errors
  - Transmit error correcting code as additional data within each frame
  - Allows receiver to correct (some) errors without contacting sender

Error Correcting Codes: Hamming Code

- Invented by Richard Hamming in the 1940s to improve reliability of computer systems – later applied to communications links



Richard Hamming

- Simple error correcting code
- At the sender:
  - Send *n* data bits and *k* check bits each word
  - Check bits are sent as bits 1, 2, 4, 8, 16, …
  - Each check bit codes parity for some data bits:
    - $b_1 = b_3 \oplus b_5 \oplus b_7 \oplus b_9 \oplus b_{11} \ldots$
    - $b_2 = b_3 \oplus b_6 \oplus b_7 \oplus b_{10} \oplus b_{11} \oplus b_{14} \oplus b_{15} \ldots$
    - $b_4 = b_5 \oplus b_6 \oplus b_7 \oplus b_{12} \oplus b_{13} \oplus b_{14} \oplus b_{15} \ldots$
    - i.e., starting at check bit *i*, check *i* bits, skip *i* bits, repeat

Character	ASCII	Hamming Code
H	1001000	00110010000
a	1100001	10111001001
m	1101101	11101010101
m	1101101	11101010101
i	1101001	01101011001
n	1101110	01101010110
g	1100111	11111001111
	0100000	10011000000
c	1100011	11111000011
o	1101111	00101011111
d	1100100	11111001100
e	1100101	00111000101

Error Correcting Codes: Hamming Code

- At the receiver:
  - set *counter* = 0
  - recalculate check bits, *k* = 1, 2, 4, 8, … in turn {
   if check bit *k* is incorrect {
       counter += *k*
    }
}
  - if (*counter* == 0) {
    no errors
} else {
    bit *counter* is incorrect
}
- Allows the receiver to detect *and correct* all possible errors that corrupt only a single bit, and some errors affecting multiple bits

Character	ASCII	Hamming Code
H	1001000	00110010000
a	1100001	10111001001
m	1101101	11101010101
m	1101101	11101010101
i	1101001	01101011001
n	1101110	01101010110
g	1100111	11111001111
	0100000	10011000000
c	1100011	11111000011
o	1101111	00101011111
d	1100100	11111001100
e	1100101	00111000101

Error Correcting Codes

- Other error correcting codes exist – tradeoff complexity and amount of data added, for the ability to correct multi-bit errors

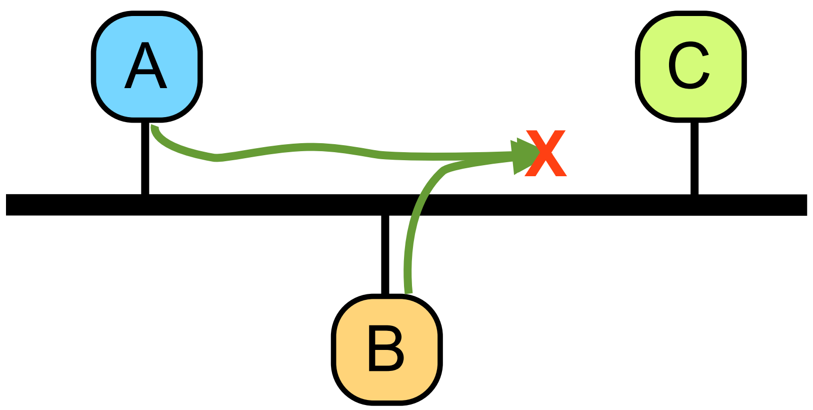
- Can also detect errors and request retransmission – error correcting codes not the only means of repair

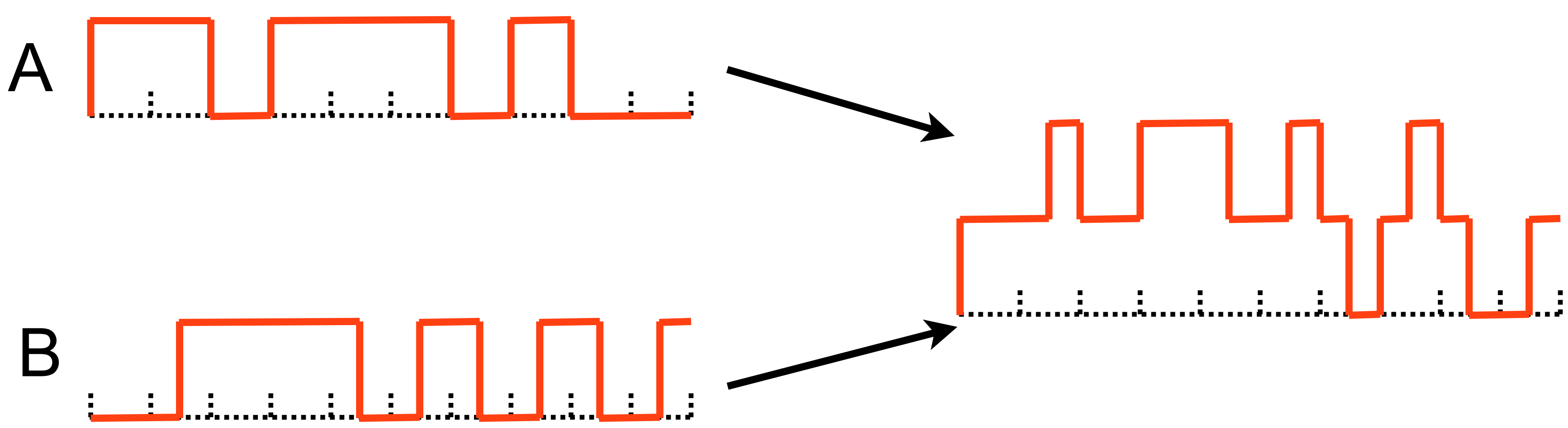
Media Access Control

- Links may be point-to-point or multi-access
- How to arbitrate access to the link?
  - Point-to-point links typically two unidirectional links
    - Separate physical cables for each direction
    - Need framing in each direction, but there is no contention for the link
    - ARQ with stop-and-wait or sliding-window for flow control
  - Multi-access links typically share a bidirectional link
    - A single physical cable – nodes contend for access to the link
    - A single radio frequency

Link Contention

- Two hosts transmit simultaneously → Collision
- Signals overlap: only garbage received



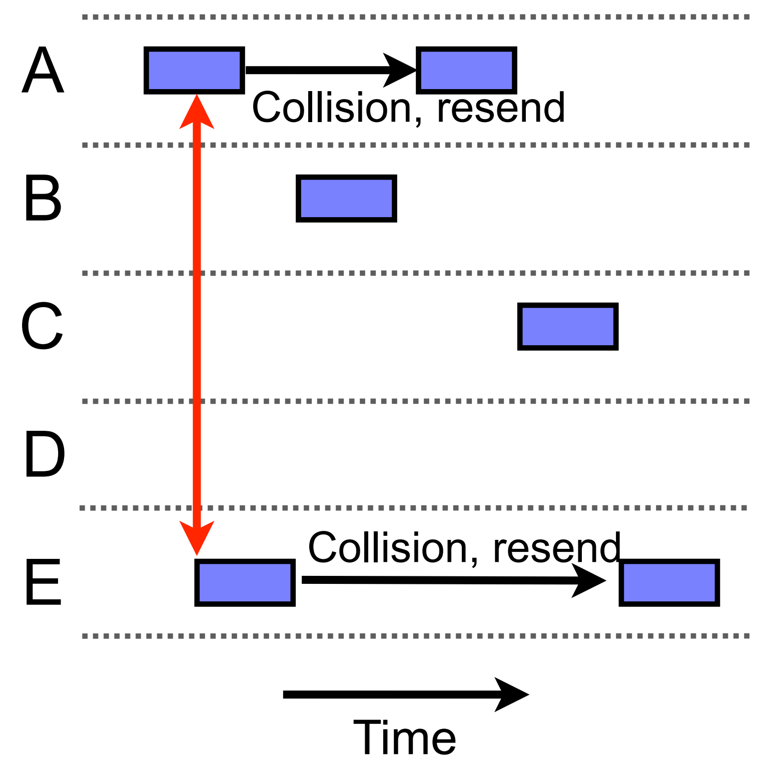


Contention-based MAC

- Multiple hosts share channel in a way that can lead to collisions: system is *contention-based*
- Two-stage access to channel:
    - Detect that a collision is occurring/will occur
        - By listening to the channel while/before sending
    - Send if no collision, or back-off and/or retransmit data according to some algorithm to avoid/resolve collision
        - Back-off delay randomised and increasing to prevent repeated collisions
        - Can be arranged to give priority to certain hosts/users/traffic classes
- Probabilistic, variable latency, access to channel

The ALOHA Network

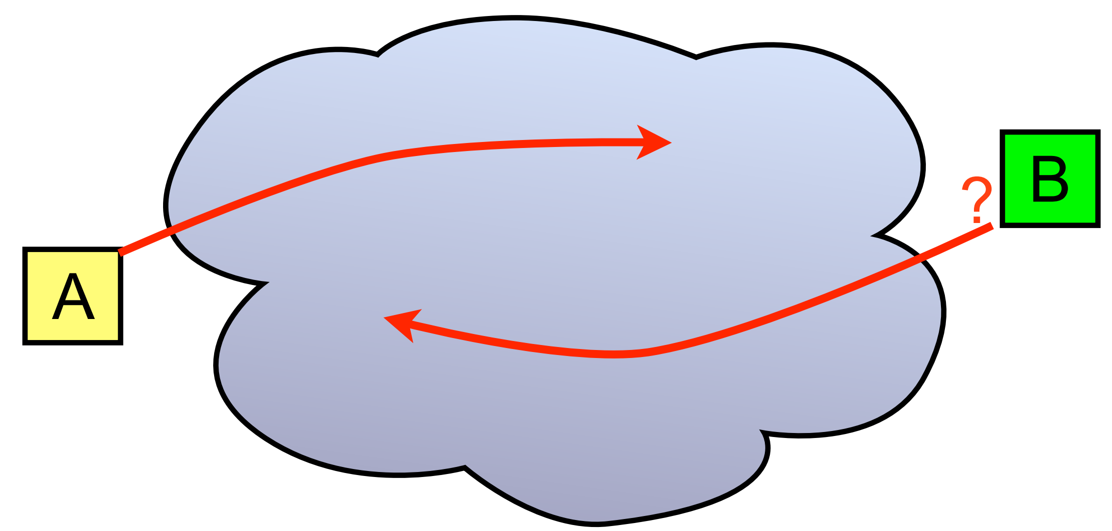
- Wireless network developed at the University of Hawaii (1970)
  - The first wireless packet switched network
- Simplest contention-based MAC
  - Try to transmit whenever data is available
  - If a collision occurs, wait random amount of time then retransmit; repeat until successful
- Simple, but poor performance
  - Low channel utilisation; long delays



Carrier Sense Multiple Access

- When propagation delay low, listen before sending
  - If another transmission is active: back-off as if collision occurred, without sending anything
  - If link is idle, send data immediately
- Improves utilisation
  - Active transmissions not disrupted by collisions
  - Only the new sender backs-off if the channel is active

Why does propagation delay matter?



A starts transmitting

B listens, hears no traffic (message from A hasn't reached it yet)

B starts transmitting

Collision occurs, as messages overlap in transit; smaller propagation delay → less likely to occur

CSMA/CD

- High propagation delay → increased collision rate
- CSMA updated with collision detection (CSMA/CD)
  - Listen to channel before, *and while*, transmitting data
  - If collision occurs, immediately stop sending, back-off, and retransmit
    - Collision still corrupts both packets
    - But, time channel is blocked due to collisions is reduced – better performance than plain CSMA
- Examples: Ethernet, 802.11 Wi-Fi

CSMA/CD: How to Back-Off?

- CSMA/CD uses back-off to avoid/resolve collisions
- How long is the back-off interval?
  - Should be random – to avoid deterministic repeated collisions
  - Should increase with the number of collisions that affect a transmission – repeated collisions signal congestion; reduce transmission rate allows the network to recover
  - Good strategy:
    - Initial back-off interval x seconds ± 50%
    - Each repeated collision before success, $x \rightarrow 2x$

Summary

- Data link layer arbitrates access to the physical layer
  - Identifies devices
  - Structures and frames the raw bitstream
  - Detects and corrects bit errors
  - Controls access to the channel
- Turn the raw bit stream into a structured communications channel

- Combination of physical and data link layers allows transmission of structured frames of data across a single physical link