University of Glasgow | School of Computing Science

Networks & Operating Systems Essentials

Dr Angelos Marnerides

<*angelos.marnerides@glasgow.ac.uk*>

School of Computing Science

Based on slides © 2017 Colin Perkins

PHYSICAL LAYER (L1)

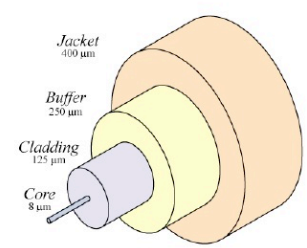
The Physical Layer

- 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?
  - What is the capacity of the channel?
- Physical characteristics of cable or optical fibre:
  - Size and shape of the plugs
  - Maximum cable/fibre length
  - Type of cable (e.g., electrical voltage, current, modulation)
  - Type of fibre (e.g., single- or multi-mode, optical clarity, colour, power output, and modulation of the laser)
- Main focus: how to transmit a sequence of bits (0/1 values) over an analogue channel, subject to noise/clock skew/hw limitations/etc.
- Interface to L2: sequence of bits
  - Hides away complexity of encoding/decoding

Example Wired Media

- Unshielded Twisted Pair (UTP)
  - 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
    - 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

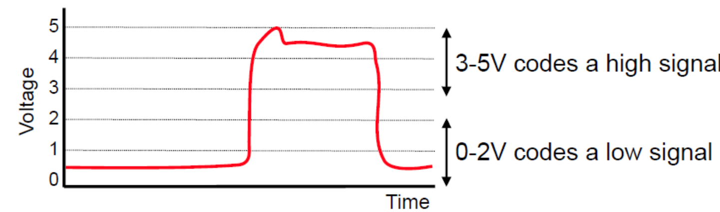


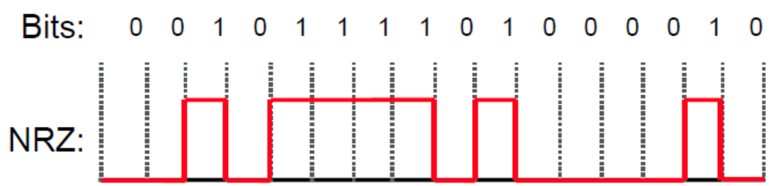
Wired Data Transmission

- Signal usually directly encoded onto the channel
- 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

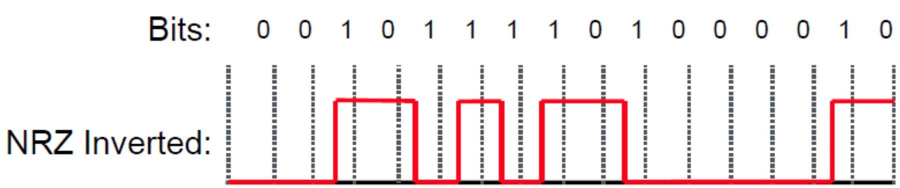
- Encode the signal as change in voltage applied to cable, or change in brightness of laser in optical fibre
- Example:





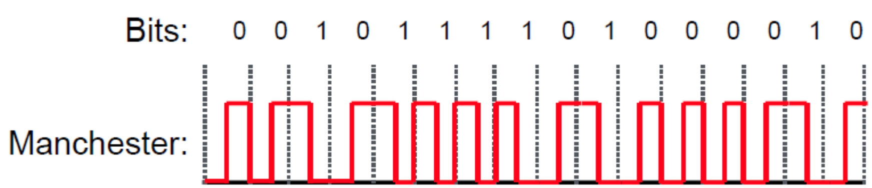
Encodes a 1 as a high signal, a 0 as a low signal

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

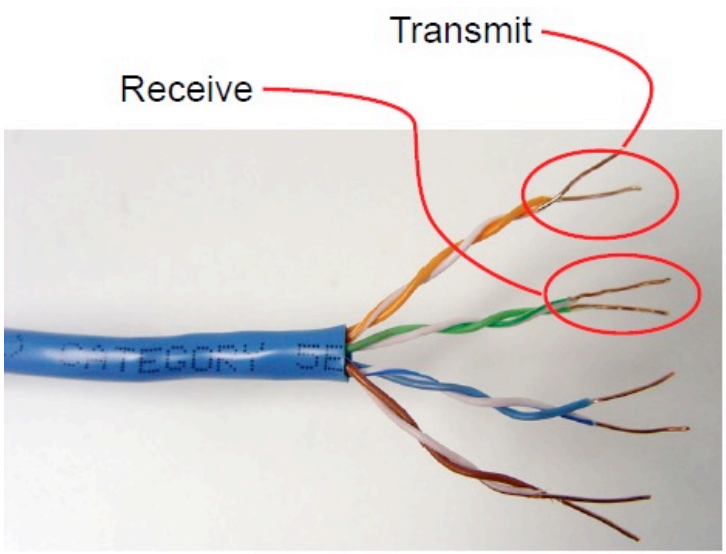


Encode a 1 as high-low transition, a 0 as a low-high transition

Doubles the bandwidth needed, but avoids problems with NRZ encoding

Example: Ethernet

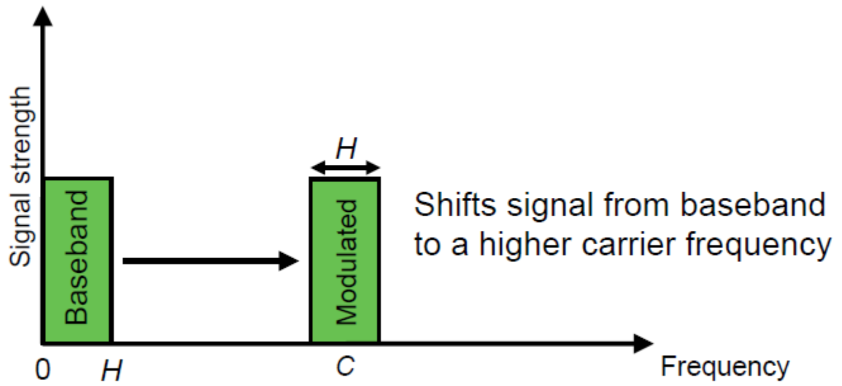
- Baseband data with Manchester coding at 10 Mbps, or 4B/5B coding at 100 Mbps



4 twisted pairs per cable
3 twists per inch
24 gauge (~0.5mm) copper
100m maximum cable length

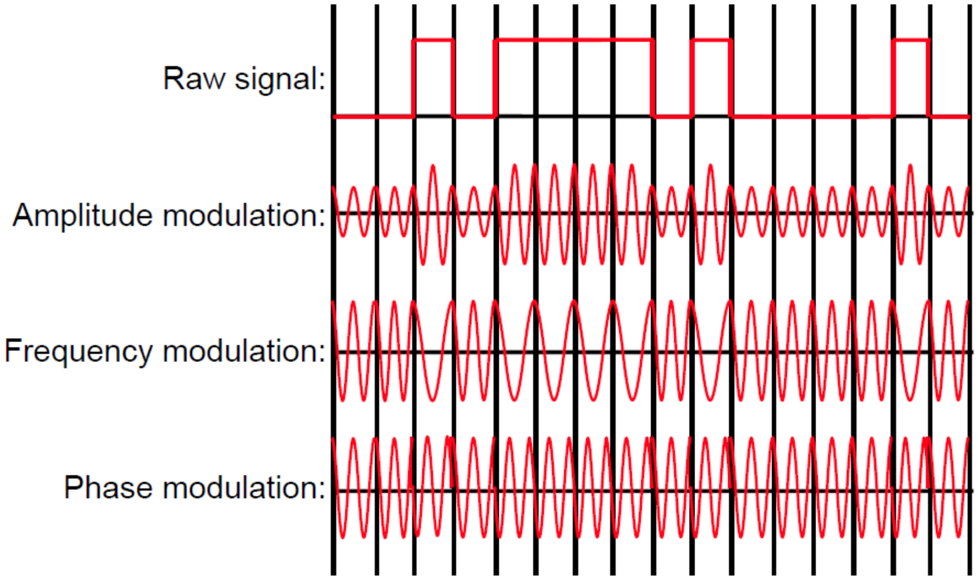
Carrier Modulation

- 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

Carrier Modulation



Symbols per second

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

Hedy Lamarr (1914-2000)

Bandwidth, Channel Capacity, Noise

- 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 (Nyquist's) 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 (Hz)
      - V = number of discrete values per symbol
    - Assumption: perfect, noise-free, channel
- Real world channels are subject to noise that corrupts the signal
  - Electrical interference, cosmic radiation, thermal noise, …
- Can measure channel's signal power, *S*, and noise floor, *N*, and compute its signal-to-noise ratio (*S/N* or *SNR*)
  - Typically quoted in decibels dB = $10 \log_{10} S/N$
  - Example: ADSL modems report *S/N* ~30 dB for good quality phone lines => signal power 1000x greater than noise
- Maximum transmission rate of a channel grows ~logarithmically to the SNR
  - $R_{max} = H \log_2(1 + S/N)$ -- *Shannon's Theorem*
- Bandwidth and SNR are fundamental limits: might be reached with careful engineering, but cannot be exceeded

Based on slides © 2017 Colin Perkins

DATA LINK LAYER (L2)

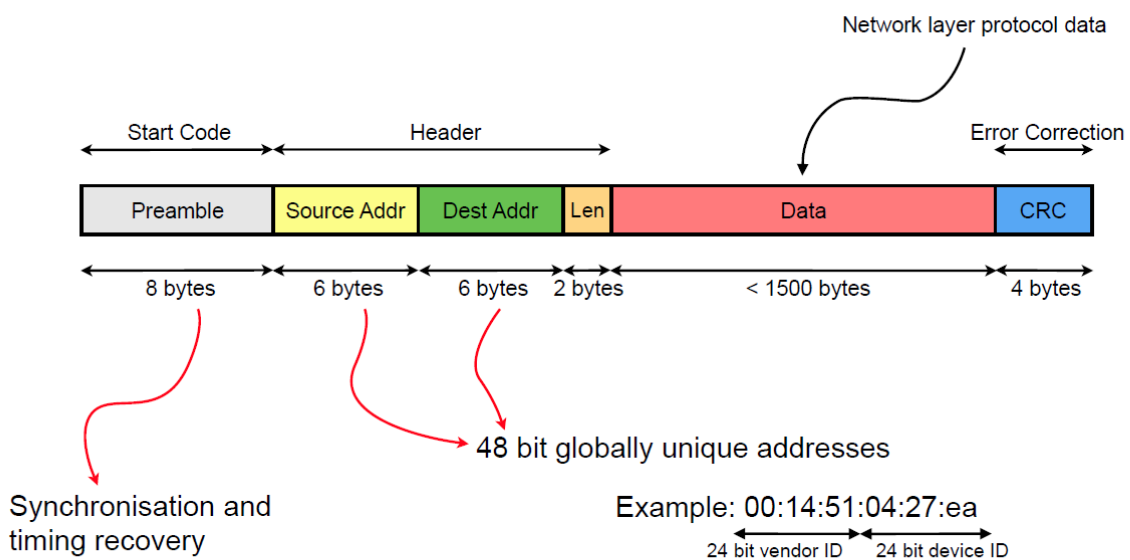
Purpose of Data Link Layer

- 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
- Interface with L1: raw bit stream
- Interface with L3: structured communication (addressing, packets)

Basic Services

- 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, 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 & synchronization
  - 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

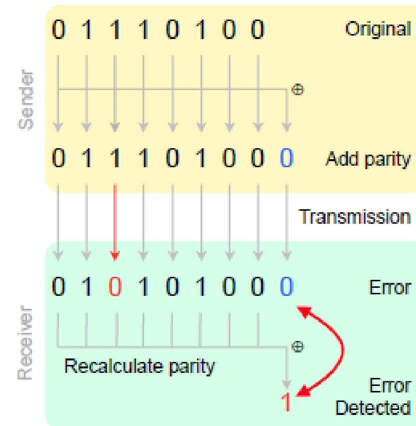


Error Detection & Correction

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



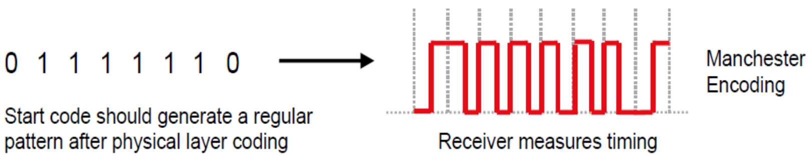

```c
#include <stdint.h>
// Internet checksum algorithm.
// Assumes data padded to a 16-bit
// boundary.
uint16_t
internet_cksum(uint16_t *buf, int buflen)
{
   uint32_t sum = 0;
   while (buflen--) {
      sum += *(buf++);
      if (sum & 0xffff0000) {
         // Carry occurred, wrap around
         sum &= 0x0000ffff;
         sum++;
      }
   }
   return ~(sum & 0x0000ffff);
}
```

Error Detection & Correction

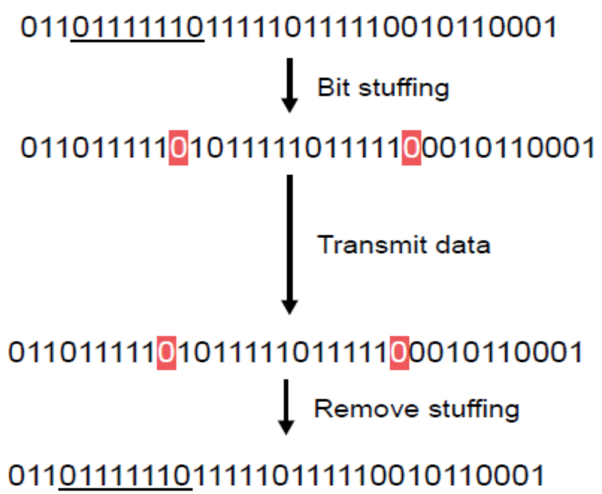
- More powerful error detecting codes exist
  - Cyclic redundancy code (CRC)
  - More complex → fewer undetected errors
- Error detecting codes can be extended to also correct errors
  - Transmit error correcting code as additional data within each frame
  - Allows receiver to correct (some) errors without contacting sender
  - Example: Hamming Code
    - Allows the receiver to detect and correct all possible errors that corrupt only a single bit, and some errors affecting multiple bits
- Other error correcting codes exist
  - Trade-off complexity and amount of data added, for the ability to correct multi-bit errors
- Error correcting codes not the only means of repair
  - Can also request retransmission on error detection

Synchronisation

- 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

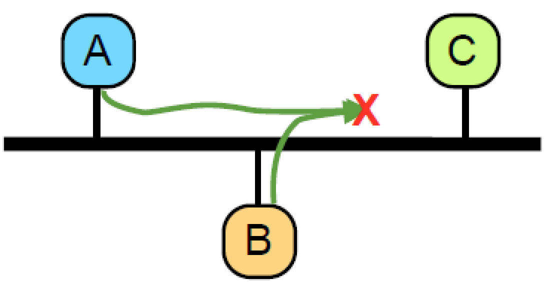


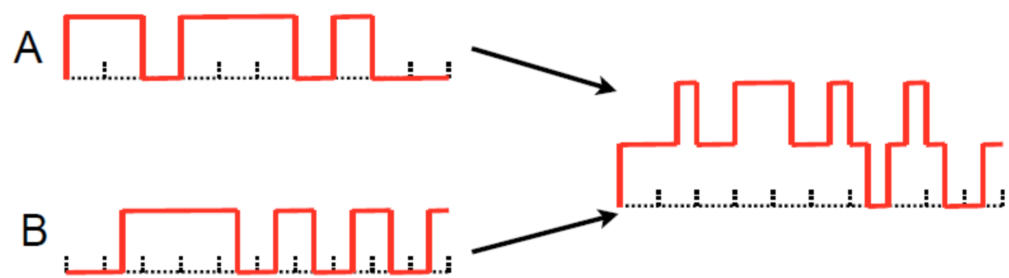
- What if start code appears in data?
  - *Bit stuffing* can give a transparent channel
  - Sender inserts a 0 bit after sending any five consecutive 1 bits – unless sending start code
  - If receiver sees five consecutive 1 bits, look at sixth bit:
    - If 0, has been stuffed, so remove
    - If 1,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



Media Access Control

- How to arbitrate access to the link?
- Links may be point-to-point or multi-access
  - 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
  - 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





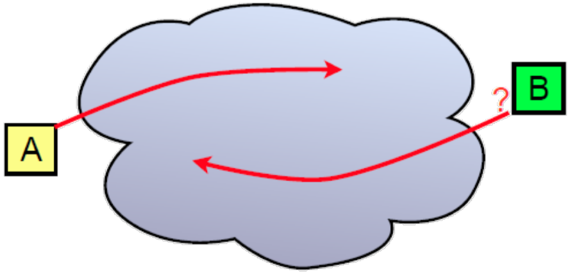
Contention-Based MAC

- 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

Contention-Based MAC

- 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 (CSMA)
  - When propagation delay low, listen before sending
    - If another transmission is active: backoff 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

Contention-Based MAC

- High propagation delay → increased collision rate; what then?
- 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

- 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 → 2x

Reading Material

- Peterson, Davies "Computer Networks: a systems approach" 5th Edition. Chapter 2

- Tanenbaum, Wetherall, "Computer Networks" 5th Edition. Chapters 2 and 3

Coming up next...

Source: https://xkcd.com/742/