Physical Layer and Data Link Control

Guevara Noubir
CSG150, lecture 2

Outline

- Physical Layer: virtual pipe
  - Brief introduction to this very large topic
- Data Link Control (DLC) layer:
  - Error detection
  - Retransmission strategies (ARQ)
  - Framing

Physical Layer: Modems

- Function:
  - Maps bits (from DLC) onto waveforms sent over the channel $s(t)$
  - s(t) is sent, and $r(t)$ is received (distorted, delayed, attenuated)
  - How to recover $s(t)$ from $r(t)$
- We focus on digital communication for point-to-point channels

Basic Encoding Schemes

- NRZ (Non Return to Zero)
- NRZI (Non return to Zero Inverted)
- Manchester coding
- *Not to be confused with error-control codes*

Physical Transmission and Channel Effect

- Created by the system designer and by the channel
  - Focus on linear time-invariant filtering:
    - If $s(t)$ -> $r(t)$, then $s(t-\tau)$ -> $r(t-\tau)$
    - If $s(t)$ -> $r(t)$, then for any real number α: $\alpha s(t)$ -> $\alpha r(t)$
    - If $s(t)_1$ -> $r(t)_1$, and $s(t)_2$ -> $r(t)_2$, then $s(t)_1 + s(t)_2$ -> $r(t)_1 + r(t)_2$
- Effect of increasing the bitrate: increased distortion=> *inter-symbol interference* (*ISI*).

Channel Impulse Response

- Let $h(t)$ be the channel output corresponding to an infinitesimally narrow pulse of unit area at time 0
  - $h(t)$ is called the channel impulse response
  - $\delta s(\tau)$ -> $\delta s(\tau)h(t-\tau)$
  - Because of the channel linearity: $r(t) = \int_{-\infty}^{+\infty} s(\tau)h(t-\tau)\,\mathrm{d}\tau$ *convolution integral*
  - The channel behavior is completely characterized by the impulse response
- Remarks:
  - $h(t) = 0$ for $t<0$.
  - The larger the non-zero duration of $h(t)$ the more ISI

Frequency Response

- Let's assume that $s(t)$ is a complex function:
  - E.g., $s(\tau) = e^{j2\pi f\tau} = \cos(2\pi f\tau) + j\sin(2\pi f\tau)$ then,

    $r(t) = H(f)e^{j2\pi ft}$ where $H(f) = \int_{-\infty}^{+\infty} h(\tau)e^{-j2\pi f\tau}\,\mathrm{d}\tau$

    $H(f)$ is called the frequency response of the channel

    The response of the channel to a real sinusoid input at frequency f is a sinusoid output at the same frequency:
    - Scaling factor: $|H(f)|$
    - Phase shift: $\angle H(f)$

Fourier Transforms

- Any function $s(t)$ can be represented as a superposition of complex sinusoids of weight: $S(f)$
  - Fourier transform $S(f) = \int_{-\infty}^{+\infty} s(t)e^{-j2\pi ft}\,\mathrm{d}t$
  - Inverse Fourier transform $s(t) = \int_{-\infty}^{+\infty} S(f)e^{j2\pi ft}\,\mathrm{d}f$
- Since the channel is linear: $r(t) = \int_{-\infty}^{+\infty} H(f)S(f)e^{j2\pi ft}\,\mathrm{d}f$
  - Then: $R(f) = H(f)S(f)$
  - Convolution in time domain = multiplication in frequency domain

Sampling Theorem

- Theorem: If a waveform is low-pass limited ($S(f) = 0$ for $|f|>W$) then $s(t)$ is completely determined by its values each $1/2W$ seconds:

$$s(t) = \sum_{i=-\infty}^{+\infty} s\left(\frac{i}{2W}\right) \frac{\sin[2\pi W(t - i/(2W))]}{2\pi W(t - i/(2W))}$$

- Conclusion: incoming digital data can be mapped into sample values spaced by $1/2W$ seconds. The resulting waveform is low-pass limited and can go through any ideal low-pass filter (W) unmodified. The received waveform can be used to recover the original data.
- How many bits can we transmit per Hertz?

Band-pass Channels

- Low-pass channels assume that: $|H(f)|$ nonzero only for a frequency band around $f = 0$.
- Most physical channels are band-pass: $|H(f)|$ nonzero for $f_1 < f < f_2$; $H(0) = 0$. (no dc component).
- Two solutions:
  - Direct coding into signals with no-dc component (Manchester encoding)
  - Modulation

Modulation

- Simplest modulation: Amplitude Modulation (AM)
  - NRZ signal is multiplied by a carrier sinusoidal signal (frequency f_0)

$$s(t)\cos(2\pi f_0 t)$$

  - Signal recovery is achieved by multiplying the received signal again by the carrier frequency

$$r(t) = s(t)\cos^2(2\pi f_0 t)$$

$$r(t) = \frac{s(t)}{2} + \frac{s(t)\cos(4\pi f_0 t)}{2}$$

Higher Modulation Schemes

- $\cos(2\pi ft)$ and $\sin(2\pi ft)$ are “orthogonal” then:
  - Two signals/bits of data can be transmitted simultaneously (QAM)
  - Phase shift keying: QPSK, 8-PSK, 16-PSK, 64-PSK
  - Frequency Shift Keying (FSK)

Shannon's Theorem

- The capacity of a channel (maximum achievable data rate in bps) is given by:

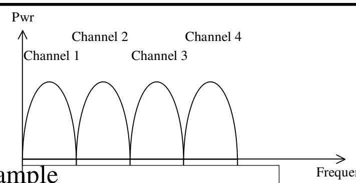
$$C = W \log_2 (1 + \frac{S}{N_0 W})$$

  - W is the available bandwidth, S is the signal power (seen by the receiver), N_0 is the noise power per Hertz
- Signal-to-noise ratio is usually expressed in dB: $10\log_{10}(S/(N_0W))$
- Example: S/N_0W = 30 dB => How many bps per Hertz?

Multiplexing Schemes

- Frequency Division Multiple
- Time Division
- Code Division
- Space Division

Frequency Division Multiple Access (FDMA)



- Example
  - FM radio set receiver: a single broadcasting station for each frequency channel
- Concept
  - assign different frequency bands to different users
  - no sharing of a frequency band between several senders
  - user separation using band-pass filters
  - continuous flow
  - two-way: two frequency bands or Time Division Duplex (TDD)

Time Division Multiple Access (TDMA)

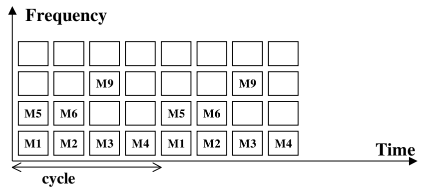
Timeslot 1	**Timeslot 2**	**Timeslot 3**	**Timeslot 4**

Concept
- use the same frequency over non-overlapping periods of time

FDMA/TDMA: Comparing delays

- Transmission delay in a FDMA system
  - DelayFD = T (transmission time in FDMA)
- Transmission delay in a TDMA system
  - DelayTD = T/M + Average-waiting-time
  - Average-waiting-time = (T/2)*(1-1/M)
  - DelayTD = DelayFD –(T/2)*(1-1/M)

Combining FDMA and TDMA



Code Division Multiple Access (CDMA)

- Concept
  - use the same frequency over overlapping periods of time with <u>different codes</u>
  - codes generate signals with "good-correlation" properties
  - signals from another user appear as "noise"
  - signals are spread over a wideband using pseudo-noise sequences
- Techniques: Spread Spectrum
  - Direct Sequence Spread Spectrum
    - IEEE802.11 (SS no CDMA), IS-95, CDMA2000, WCDMA
  - Frequency Hopping Spread Spectrum (slow and fast)

CDMA

- Advantages
  - frequency diversity:
    - resistance to jamming, selective fading, multi-path fading
  - easy frequency planning
  - soft-handover (macro-diversity)
  - better performance when load is low
- Drawbacks
  - requires efficient synchronization:
    - easy on down-link, difficult on uplink
  - all users signals must reach the base-station with the same power: near-far problem
    - power-control: accurate (1dB)
  - codes allocation

SDMA and PDMA

- Space Division Multiple Access (resource reuse)
  - a frequency/time slot/code can be used by two different users but not at the same location
  - examples: distant cells, satellite spot beams
- Polarization Division Multiple Access
  - different antenna polarization are used
  - example: two-orthogonal antenna polarization in satellite communication

Digital Channels

- Several channels (wired) are designed to carry digital data directly (no need for a modem)
- When a digital repeater is used these channels provide an improved performance over channels carrying analog data.
  - The reason is that the noise is removed at each repeater in a digital channel, while it is amplified with an analog repeater.

Error Detection

- Assume that we when know the beginning/end of the frames. The number of data bits in the frame is *K* bits.
- How can we detect if one/several bits changed duration their transmission?
- Since all the frames can potentially be received then we have to add some redundancy bits (*L*) to detect errors (checksum).

Parity Checks

- Single parity checks:
  - For every string of data bits append a single bit: parity bit
  - If the number of 1's in the string is even then the parity = 0; otherwise 1.
  - E.g., ASCII characters of 7 bits + 1 parity bit.
  - Number of 1's in an encoded string is always even.
  - This encoding allows to detect all single errors and no-two errors, etc.
  - Not sufficiently reliable specially when errors occur in bursts

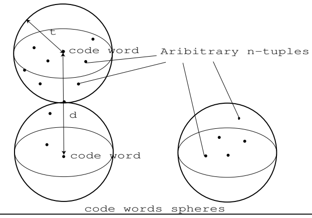
Horizontal-Vertical Parity Checks

- Data is arranged into a two-dimensional array
  - A single parity bit is appended to each row and each column

1 1	1	0	0	0	1
0 1	0	1	1	0	1
1 0	1	0	1	1	0
1 0	0	0	0	1	0
1 1	1	1	1	0	1
0 1	1	0	1	0	1

Hamming Distance for Block Codes

- The Hamming distance between two codewords is the number of places where they differ
- The Hamming distance of a Block code is the minimum distance between two code words



Example of Binary Block Code (7, 4)

- Any two different code words are different on at least three different coordinates. This code has Hamming distance 3.

Message block	Code word
(0 0 0 0)	(0 0 0 0 0 0 0)
(1 0 0 0)	(1 1 0 1 0 0 0)
(0 1 0 0)	(0 1 10 1 0 0)
(1 1 0 0)	(1 0 1 1 1 0 0)
(0 0 1 0)	(1 1 1 0 0 1 0)
(1 0 1 0)	(0 0 1 1 0 1 0)
(0 1 1 0)	(1 0 0 0 1 1 0)
(1 1 1 0)	(0 1 0 1 1 1 0)
(0 0 0 1)	(1 0 1 0 0 0 1)
(1 0 0 1)	(0 1 1 1 0 0 1)
(0 1 0 1)	(1 1 0 0 1 0 1)
(1 1 0 1)	(0 0 0 1 1 0 1)
(0 0 1 1)	(0 1 0 0 0 1 1)
(1 0 1 1)	(1 0 0 1 0 1 1)
(0 1 1 1)	(0 0 1 0 1 1 1)
(1 1 1 1)	(1 1 1 1 1 1 1)

- Notice that the last 4 bits of the code word are the same as the message
  - This is a systematic coding
  - The other 3 bits are redundancy bits

Cyclic Redundancy Checks (CRC)

- Data bits: $s_{K-1}, s_{K-2}, \ldots, s_1, s_0$
- Polynomial representation:

$$S(D) = s_{K-1}D^{K-1} + s_{K-2}D^{K-2} + \ldots + s_1 D + s_0$$

- The CRC is also viewed as polynomial:

$$C(D) = c_{L-1}D^{L-1} + c_{L-2}D^{L-2} + \ldots + c_1 D + c_0$$

- The transmitted frame can be represented as:

$$x(D) = s(D)D^L + c(D)$$

$$x(D) = s_{K-1}D^{L+K-1} + \ldots + s_0 D^L + c_{L-1}D^{L-1} + c_{L-2}D^{L-2} + \ldots + c_1 D + c_0$$

Generating the CRC

- The CRC is the remainder of dividing the information polynomial $S(D)$ by a generator polynomial $g(D)$.

$$c(D) = \text{Remainder}[\frac{s(D)D^L}{g(D)}]$$

- $g(D) = D^L + g_{L-1}D^{L-1} + \ldots + g_1 D + 1$

- Example: divide D^5+D^3 by D^3+D^2+1

Hardware Generation of the CRC

- Binary divisions can be efficiently implemented using Linear Feed-Back Shift Registers (LFSR)

Error Detection Capability of CRC

- All single bit errors are detected
- All errors of burst length less than L+1 are detected
- Primitive polynomials allow to detect all double-errors when the frame length is less than 2^L-1

- Choice of the generator polynomial:
  - Product of a primitive polynomial by (D+1)
  - CRC-16: $D^{16} + D^{15} + D^2 + 1$
  - CRC CCITT: $D^{16} + D^{12} + D^5 + 1$

Non-Binary Codes

- The data is a sequence of symbols of several bits
- A symbol is in error if any of its bits has an error
- Advantages:
  - Easier to implement in software (e.g., TCP checksum)
  - Handle bursts of errors

Automatic Repeat reQuest (ARQ)

- Assumption:
  - The DLC knows the start/end of the frame
  - The DLC is able to detect frames with errors (e.g., using a CRC)
  - Frames that are not lost arrive in their transmission order
- ARQ retransmission strategies:
  - Stop-and-Wait ARQ
  - Go Back *n* ARQ
  - Selective Repeat ARQ

Stop-and-Wait ARQ Protocol

- Protocol for A is sending packets to B
- A simple but bogus strategy:
  - The first packet is sent in the first frame
    - If the packet is correctly received by B, B sends an Acknowledgement (Ack)
    - If the frame has errors, B sends a negative ACK (Nak)
    - If the frame, Ack/Nak is lost, or a Nak is received, then A retransmits the packet.
  - When the packet is acknowledged A proceeds to the next packet
- Malfunctions can occur due to variable transmission delays or loss of Ack.
- Conclusion: it is necessary to use frame numbering.

Stop-and-Wait ARQ Protocol

- Algorithm at node *A*:
  1. Set the integer variable *SN* to 0
  2. Accept/wait for a packet from higher layer and assign *SN* to this packet
  3. Transmit the SN^{th} packet in a frame (sequence number=*SN*)
  4. If an error-free frame is received from *B* with $RN>SN$, increase *SN* to *RN* goto 2, otherwise [within some finite delay] goto 3.
- Algorithm at node *B*:
  1. Set *RN* to 0; loop on 2 and 3
  2. On receipt of an error-free frame with $SN=RN$ release packet to higher layer and increment *RN*
  3. At arbitrary times (bounded delay) after receiving any error-free data from *A*, transmit a frame to *A* containing *RN* in the request number field

Correctness of Stop-and-Wait

- Safety property: never produces an incorrect result
- Liveness property: no-deadlock

Sequence Number Optimization for Stop-and-Wait ARQ

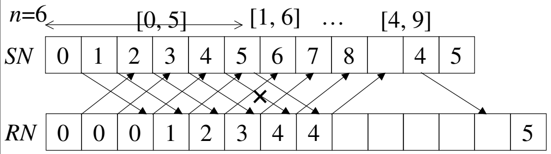
- The sequence number *SN* cannot increase without bound:
  - Loss of bandwidth, and requires a variable size *SN* field
- In the Stop-and-Wait protocol the uncertainty is always between:
  - *B* side: SN received are either *RN*(*t*) or *RN*(*t*)-1 (*i*+1, or *i*)
  - *A* side: *RN* received are either *SN*(*t*) or *SN*(*t*)+1 (i, or i+1)
- Sequence number is sent modulo 2: also called the alternating bit protocol

Limitation of S&W

- When the sender is waiting for the acknowledgment the channel is not used!
- Loss of bandwidth specially for channels with high round-trip-delay
  - E.g., satellite channels, channels with low-computation power nodes (buffering), etc.

Go-Back *n* ARQ Protocol

- Informal description:
  - Several successive packets are allowed to be sent: up to *n*.
  - The receiver accepts packets only in the correct order.
  - A request *RN* acknowledges all packets with *SN*<*R* and requests the transmission of packet *RN*



Go-Back *n* Algorithm

- Transmitting side: *A*
  1. Set SN_{min} and SN_{max} to 0.
  2. Do steps 3, 4, 5 repeatedly in any order.
  3. If $SN_{max} < SN_{min} + n$, and if a packet is available from the higher layer, accept it, assign number SN_{max} to it and increment SN_{max}.
  4. If an error-free frame is received from *B* containing $RN > SN_{min}$, increase SN_{min} to *RN*
  5. If $SN_{min} < SN_{max}$ (+no frame being transmitted) choose *SN*: $SN_{min} <= SN < SN_{max}$ and transmit the SN^{th} packet (generally on timout select $SN=SN_{min}$)
- Receiving side: *B*
  1. Set *RN* to 0 and loop over steps 2 and 3
  2. Whenever an error-free frame is received containing a sequence number equal to *RN*, release the packet to the higher layer and increment *RN*
  3. At arbitrary times, within a bounded delay after receiving a frame from *A*, transmit a frame to *A* containing *RN* in the request number field

Sequence Number Modulus in GBN

- If we assume that frames arrive in the their transmission order, then the GBN can use sequence number, and request numbers modulo $m > n$.
- *Sketch of proof* on *A* side (similar on *B* side):
  - Let t_1 bet the time when a frame is generated by *A*, and t_2 the time it is received by *B*. The *SN* of this frame verifies:
  - $SN_{min}(t_1) <= SN <= SN_{min}(t_1) + n - 1$, and
  - $SN_{min}(t_1) <= RN(t_2) <= SN_{min}(t_1) + n$,
  - $| RN(t_2) - SN| <= n < m$,
  - Thus RN = SN (mod *m*) *iff SN* = *RN* (there is no ambiguity)

Selective Repeat ARQ Protocol

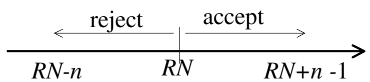
- Go-Back-N is better than Stop-and-Wait
  - Channel usage is improved
- However, when a packet is lost, then all sub-sequent packets have to re-transmitted. This is because the receiver accepts packets only in correct order
- Idea behind Selective Repeat (SR) ARQ:
  - The receiver accepts all error-free frames in the range of *RN* to *RN*+*n*-1. Even if they are out-of-order.
  - The receiver sends *RN* and also sends the *SN* of other correctly received frames (with *SN*>*RN*).

GBN vs. SR

- Use GBN:
  - Some channels are subject to errors in bursts. There is a high probability that when a frame is lost, the following frames will also be lost.
  - Some communication nodes are limited in memory and processing. They cannot store packets that are out-of-order, or they cannot re-order them.
- Otherwise, use SR.

Sequence Numbers in SR

- If we assume that frames arrive in the their transmission order, then the SR can use sequence number, and request numbers modulo $m => 2n$.
- *Sketch of proof* (*A* side):
  - $RN(t_2) - n <= SN <= RN(t_2) + n - 1$



Two-way communication

- SN and RN can be piggy-backed in the exchanged frames
- To avoid blocking one-direction when then other one is idle, it is necessary to permit frames containing only request. These frames are activated by a timer.

Error Recovery at Transport Layer

- Difference between a link layer, and transport layer communications:
  - In transport layer: numbering of session packets. In link layer numbering of all link packets
  - Packets may arrive out-of-order in transport layer
  - Timers setup is much more critical at the transport layer because of congestion problems
- ARQ protocols:
  - Same basic protocols as in the link layer
  - If packets can arrive out-of-order: numbering packets modulus a number is no more possible

Error Recovery at Transport Layer

- Solutions to the order problem:
  - Require that the network delivers packets in order (e.g., fixed routing). However, this is generally not realistic since links may break and routing path will have to change.
  - Use a very large modulus.
  - Use a mechanisms that destroys packets after a limited time, and use a modulus large enough so that a wraparound does not happen during the lifetime of a packet.

Framing

- The physical layer provide a mean to transmit a sequence of bits.
- How can one determine the beginning/end of a frame?
- Solutions:
  - Character-based framing (use special control characters)
  - Bit-oriented framing with flags
  - Length counts

Character-Based Framing

- We can use a character code such the ASCII (American Standard Code for Information Interchange): sequence of 7 bits (usually with a parity bit).

SYN	SYN	STX	Header	Packet	ETX	CRC	SYN	SYN

  - SYN: Synchronous idle, STX: Start of text, ETX: End of text
- Problem 1: if control characters appear within the header, or CRC.
  - These are known location, one can skip control characters in these fields
- Problem 2: if CTRL characters appear in the packet.
  - Use a Data Link Escape (DLE) character before STX (for start frame), and before ETX (for end frame), but not when it appears within the packet.
  - If DLE appears within the packet replace it with DLE DLE.
    - Example: DLE STX => start of frame, DLE ETX => end of frame,
    - DLE DLE STX => binary sequence DLE STX

Character-based Framing (Cont'd)

- Disadvantages:
  - Excessive overhead: DLE STX and DLE ETX for each frame, for each DLE within the packet and additional character has to be inserted (potentially doubling the size of the frame)
  - The frame has to contain an integral number of characters
- Potential errors:
  - Error in the DLE ETX => undetected end of frame
  - Error leading to the appearance of a DLE ETX

Bit-Oriented Framing: Flags

- Use a flag to indicate the end of a frame. The appearance of the flag within the frame is masked through a technique called bit stuffing (e.g., High-level DLC).
- A bit-oriented frame can have any length.
- Example:
  - Flag: 01111110 (= 01^60) (other flags could be used).
  - Rule of bit stuffing: insert a 0 after any sequence of five 1's
- Expected number of insertions = $(E(K)-j+3)2^{-j}$.
  - K is frame length, bit j-1 counts twice.
  - Optimal $j = \lfloor \log_2 E\{K\} \rfloor$

Length Fields

- SYN + Length field in the frame header
  - E.g., DECNET
- Can an encoding method require a smaller expected number of bits? Answer Shannon's source coding theorem of information theory.
- The minimum expected number of bits is at least the entropy of the distribution:

$$H = \sum_K P(K) \log_2 \frac{1}{P(K)}$$

- Idea: map more likely values of K into shorter bit strings
- Example the unary-binary encoding ($j = \lfloor \log_2 E\{K\} \rfloor$):
  - $K = i2^j + r$ is encoded as 0^i1r

Framing with Errors

- Lost flag (the receiver will stop at the next flag). Detection through CRC (2^{-L}).
- Error within the frame to change a bit into a flag (e.g., 01^60). Detection through CRC.
- Error in the length field. Detection through CRC or header CRC.

- Let γ be the expected number of transmitted frames from *A* to *B* per successfully accepted packet at *B*
- *Let* β be the expected number of transmitted frames between the transmission of a frame and reception of feedback about that frame
- Let *p* be the probability that a frame arriving at *B* contains errors
- *A* is always busy transmitting frames and *n* is large enough (*A* never goes back in absence of feedback)
- Then: $\gamma = 1 + p(\beta + \gamma)$

Frequency Division Multiple Access (FDMA)

- Advantages: simple receivers
  - longer symbol duration: no-need for equalization
    - low inter-symbol interference
    - *e.g.*, 50kb/s QPSK ⇒ 40 µs >> 1-10µs delay spread
- Drawbacks
  - frequency guard bands, costly tight band-filters
  - long fading duration: need slow frequency hopping

Time Division Multiple Access (TDMA)

- Advantages
  - simple filters (window)
  - possibility to transmit and receive over the same frequency channel
- Drawbacks
  - users must be synchronized
  - guard times
  - short symbol duration: need for equalization, training sequences...
    - high inter-symbol interference
    - e.g., 50Kbps, QPSK, 8 users:
      - 5 µs symbol duration
      - delay spread: 1µs (cordless), upto 20µs for cellular