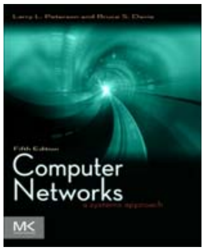
Computer Networks: A Systems Approach, 5e

Larry L. Peterson and Bruce S. Davie



Chapter 2

Getting Connected

Problems

- In Chapter 1 we saw networks consists of links interconnecting nodes. How to connect two nodes together?
- We also introduced the concept of “cloud” abstractions to represent a network without revealing its internal complexities. How to connect a host to a cloud?

Chapter Outline

Chapter 2

- Perspectives on Connecting nodes
- Encoding
- Framing
- Error Detection
- Reliable Transmission
- Ethernet and Multiple Access Networks
- Wireless Networks

MK MORGAN KAUFMANN 3

Chapter Goal

Chapter 2

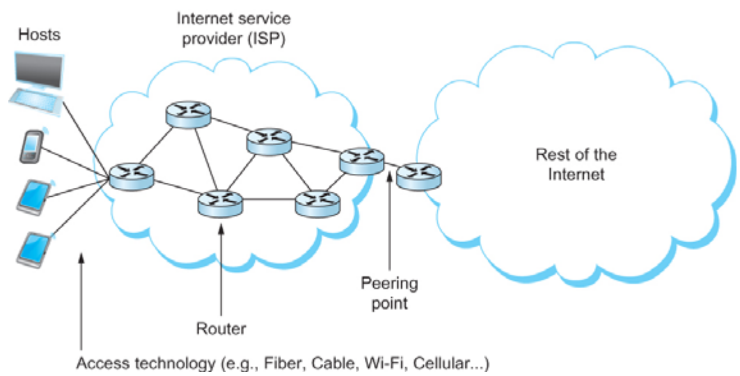
- Exploring different communication medium over which we can send data
- Understanding the issue of encoding bits onto transmission medium so that they can be understood by the receiving end
- Discussing the matter of delineating the sequence of bits transmitted over the link into complete messages that can be delivered to the end node
- Discussing different technique to detect transmission errors and take the appropriate action

MK MORGAN KAUFMANN 4

Chapter Goal (contd.)

- Discussing the issue of making the links reliable in spite of transmission problems
- Introducing Media Access Control Problem
- Introducing Carrier Sense Multiple Access (CSMA) networks
- Introducing Wireless Networks with different available technologies and protocol

Perspectives on Connecting



An end-user's view of the Internet

Link Capacity and Shannon-Hartley Theorem

- Gives the upper bound to the capacity (C) of a link in terms of bits per second (bps) as a function of signal-to-noise ratio of the link measured in decibels (dB).
- $C = B*\log_2(1+S/N)$
  - Where B = 3300 – 300 = 3000Hz is the bandwidth
  - S is the signal power over the bandwidth
  - N the average noise over the bandwidth.
  - The signal to noise ratio (S/N) is measured in decibels is related to $dB = 10 \times \log_{10}(S/N)$. If the result 30dB then S/N = 1000.
  - Now $C = 3000 \times \log_2(1001)$ = 30kbps (29.902kbps)
  - How can we get 56kbps?

Links

- All practical links rely on some sort of electromagnetic radiation propagating through a medium or, in some cases, through free space
- One way to characterize links, then, is by the medium they use
  - Typically copper wire in some form (as in Digital Subscriber Line (DSL) and coaxial cable),
  - Optical fiber (as in both commercial fiber-to-the home services and many long-distance links in the Internet's backbone), or
  - Air/free space (for wireless links)

Links

Chapter 2

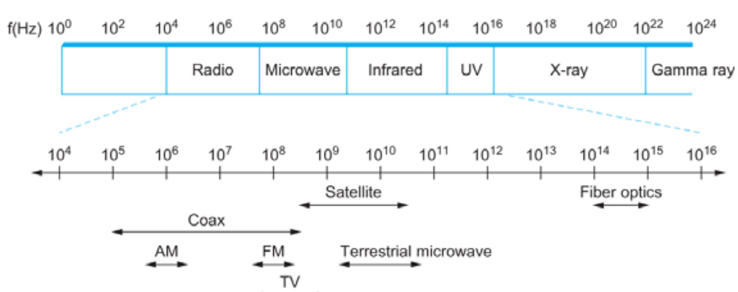
- Another important link characteristic is the *frequency*
  - Measured in hertz, with which the electromagnetic waves oscillate
- Distance between the adjacent pair of maxima or minima of a wave measured in meters is called *wavelength*
  - Speed of light divided by frequency gives the wavelength.
  - Frequency on a copper cable range from 300Hz to 3300Hz; Wavelength for 300Hz wave through copper is

  (speed of light on copper) / frequency = $2/3*(3 \times 10^8)/300 = 667 \times 10^3$ meters.
- Placing binary data on a signal is called *encoding*.
- Modulation involves modifying the signals in terms of their frequency, amplitude, and phase.

9

Links

Chapter 2



Electromagnetic spectrum

10

Links

Chapter 2

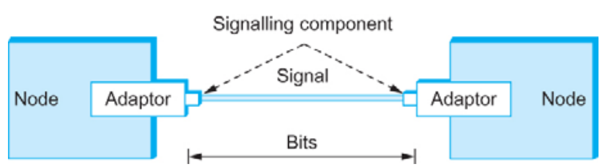
Service	Bandwidth (typical)
Dial-up	28–56 kbps
ISDN	64–128 kbps
DSL	128 kbps–100 Mbps
CATV (cable TV)	1–40 Mbps
FTTH (fibre to the home)	50 Mbps–1 Gbps

Common services available to connect your home

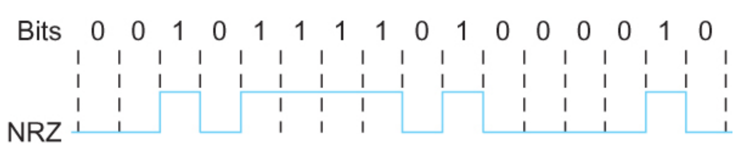
11

Encoding

Chapter 2



Signals travel between signaling components; bits flow between adaptors



NRZ encoding of a bit stream

12

Encoding

Chapter 2

- Problem with NRZ - Baseline wander
  - The receiver keeps an average of the signals it has seen so far
  - Uses the average to distinguish between low and high signal
  - When a signal is significantly lower than the average, it is 0, else it is 1
  - Too many consecutive 0's and 1's cause this average to change, making it difficult to detect

Encoding

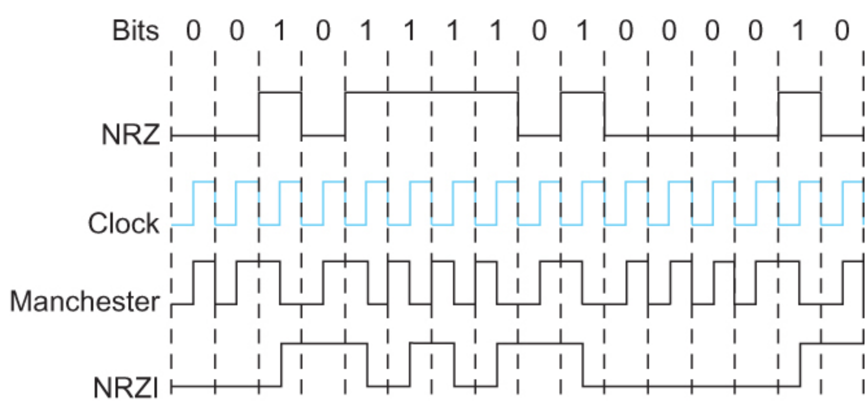
Chapter 2

- Problem with NRZ - Clock recovery
  - Frequent transition from high to low or vice versa are necessary to enable clock recovery
  - Both the sending and decoding process is driven by a clock
  - Every clock cycle, the sender transmits a bit and the receiver recovers a bit
  - The sender and receiver have to be precisely synchronized

Encoding

- NRZI - Non Return to Zero Inverted
  - Sender makes a transition from the current signal to encode 1 and stay at the current signal to encode 0
  - Solves for consecutive 1's
  - Does not solve problem with consecutive 0's
  - Transition occurs on rising clock edge
  - Bit rate is the same as the baud rate

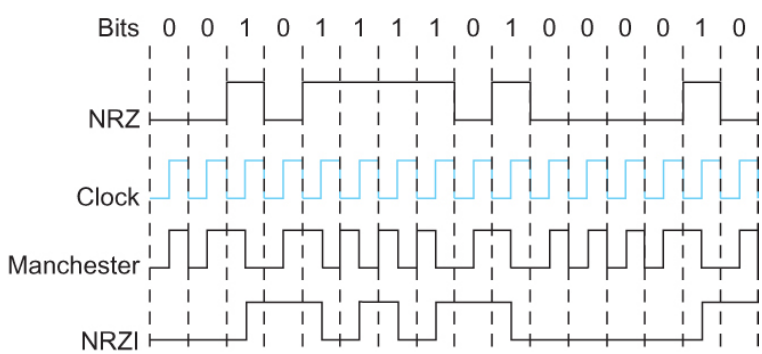
NRZI Waveform



Encoding

- Manchester encoding
  - Merging the clock with signal by transmitting Ex-OR of the NRZ encoded data and the clock
  - Clock is an internal signal that alternates from low to high, a low/high pair is considered as one clock cycle
  - In Manchester encoding
    - 0: low→ high transition
    - 1: high→ low transition

Encoding



Different encoding strategies

Encoding

Chapter 2

- Problem with Manchester encoding
  - Doubles the rate at which the signal transitions are made on the link
    - Which means the receiver has half of the time to detect each pulse of the signal
  - The rate at which the signal changes is called the link's baud rate
  - In Manchester the bit rate is half the baud rate (50% efficiency)

MK MORGAN KAUFMANN 19

Encoding

Chapter 2

- 4B/5B encoding
  - Insert extra bits into bit stream so as to break up the long sequence of 0's and 1's
  - Every 4-bits of actual data are encoded in a 5- bit code that is transmitted to the receiver
  - 5-bit codes are selected in such a way that each one has no more than one leading 0(zero) and no more than two trailing 0's.
  - No pair of 5-bit codes results in more than three consecutive 0's
  - Transmitted using NRZI
  - 80% efficient

MK MORGAN KAUFMANN 20

Encoding

- 4B/5B encoding

Data	*Transmit*	*Other 5B codes*
0000 →	11110	16 left (16 used for data)
0001 →	01001	11111 – when the line is idle
0010 →	10100	00000 – when the line is dead
..		00100 – to mean halt
..		
1111 →	11101	13 left : 7 invalid, 6 for various control signals

Encoding

- 4B/5B encoding

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