IS1211/IS2111

Computer Networks

Dr. Chamath Keppitiyagama

University of Colombo School of Computing

Logarithms

$$\begin{aligned} 100 &= 10^2 \\ log(100) &= 2 \end{aligned}$$

Logarithms

$$\begin{aligned} 100 &= 10^2 \\ log_{10}(100) &= 2 \end{aligned}$$

Logarithms

$$
\begin{aligned}
0.01 &= \frac{1}{100} \\
0.01 &= \frac{1}{10^2} \\
0.01 &= 10^{-2} \\
log_{10}(0.01) &= -2
\end{aligned}
$$

Logarithms

$$\begin{aligned} 32 &= 2^5 \\ log_2(32) &= 5 \end{aligned}$$

Think . . .

Assume that you can count 1 million numbers per second.

How long will it take to count from 1 to

$$2^{64}?$$

Logarithms

$$
\begin{aligned}
2 &= 10^{0.3} \\
\frac{1}{2} &= 2^{-1} \\
2^{-1} &= 10^{-0.3} \\
log_{10}(\frac{1}{2}) &= -0.3
\end{aligned}
$$

Logarithms

$$\begin{aligned}\frac{Power_{out}}{Power_{in}} &= \frac{1}{2}\\ log_{10}(\frac{Power_{out}}{Power_{in}}) &= log_{10}(\frac{1}{2})\\ &= -0.3Bell\end{aligned}$$

$$-0.3Bell = -3dB$$

Bandwidth

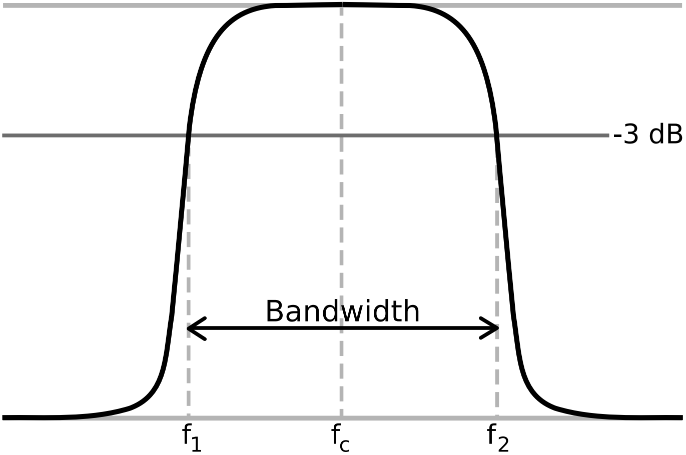


Image credit: Wikipedia

$$10log(\frac{1}{2}) \approx -3$$

Bandwidth and the Bitrate
How fast can we send data over a channel?

Nyquist's Theorem

$$R = 2H \ log_2 L$$

- R = data rate (bits/sec)
- H = bandwidth of the channle (Hz)
- L = number of signal levels

Bandwidth and the Bitrate
How fast can we send data over a channel?

Nyquist's Theorem

$$R = 2H\ log_2 L$$

- R = data rate (bits/sec)
- H = bandwidth of the channle (Hz)
- L = number of signal levels

What is the maximum bitrate possible in a noiseless channel if the bandwidth is $1000Hz$?

Not so fast !!

Shannon's Law

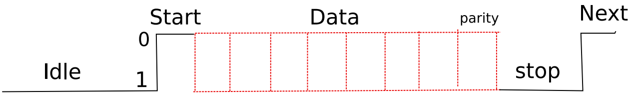
$$R = H \; log_2(1 + \frac{S}{N})$$

- S = signal level
- N = noise level

The problem

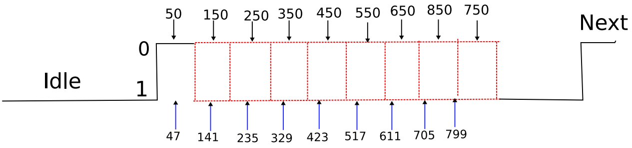
- A sender transmits data at a rate of 1Mbps - one million bits per second.
  - one bit every $\frac{1}{10^6}$ seconds $\rightarrow$ one bit every 1μ seconds
  - The sender has a clock.
- The receiver tries to sample the medium at the center of every bit and should sample the line once every $1\mu s$.
- The receiver has its own clock.
- Assume that the receivers clock is 1% faster.
  - If the first sample is taken right at the center of a bit time then the second sample will be $0.01\mu s$ off from the center.
  - After 50 more samples the sampling would be more than $0.5\mu s$ off from the center !!

Asynchronous Transmission



- Don't send long uninterrupted sequence of bits.
- Send one character at a time.
- At the beginning of each character the receiver gets another chance to synchronize the clock.

Errors

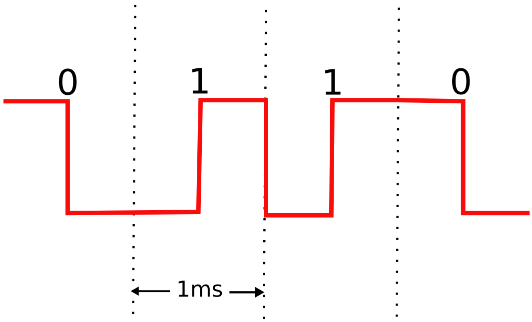


- Sender sends a bit every $100\mu s$.
- The receiver is 6% faster and samples the line every $94\mu s$.
- Two errors
  - Last sampled bit is incorrectly received.
  - If the bit 7 is 1 and bit 8 is 0 then bit 8 could be taken as a starting bit.
    - *Framing Error*

Synchronous Transmission

- Transmit a block of data as a stream of bits without a start or stop bits.
- Keep the clocks synchronized.
  1. Use a seperate set of lines between the sender and receiver. The sender sends the clock pulse to the receiver over these lines.
     - Works for short distances.
     - Clock pulse is another piece of data. We again have another synchronization problem.
  2. Embed the clocking information in the data signal.
     - Manchester encoding.

Manchester Encoding



- There is a transition at the middle of each bit period.
- What is the baud rate?
- What is the bit rate?

Error Detection

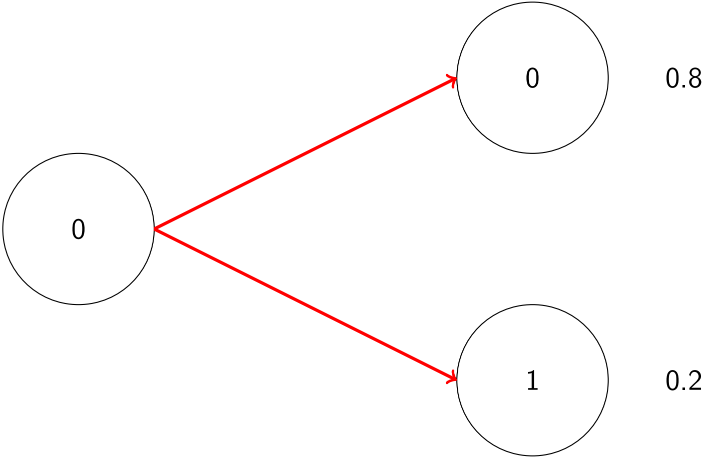

0
0

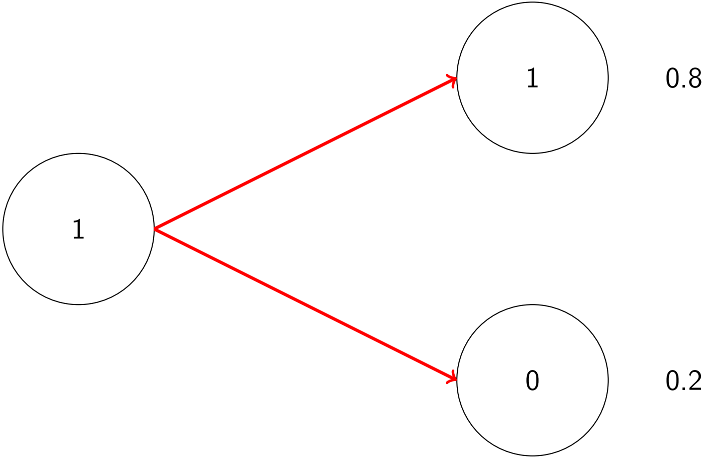

0
1


1
1


1
0


1,0
1,0 ??


0
0
0.8
1
0.2

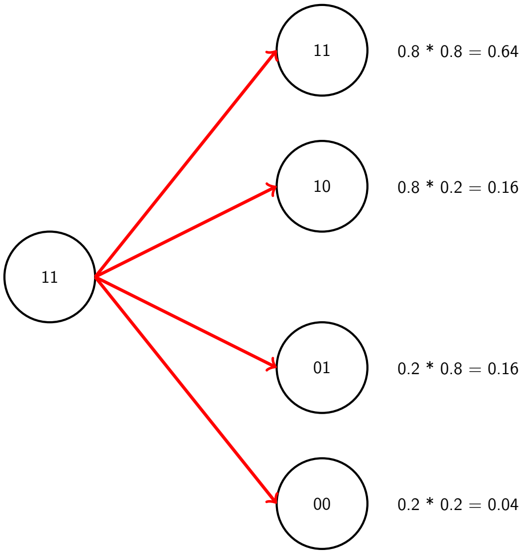

1
1
0.8
0
0.2

Probability of accepting a wrong bit $= 0.2$

Error Detecting Code

$$0 \Rightarrow 00$$

$$1 \Rightarrow 11$$


11
11
10
01
00
0.8 * 0.8 = 0.64
0.8 * 0.2 = 0.16
0.2 * 0.8 = 0.16
0.2 * 0.2 = 0.04

Probability of wrong decoding $= 0.04$

Parity

1	1	1	0	0	1	0	1	·

Parity

1	1	1	0	0	1	0	1	1

Parity

1	0	1	0	0	1	0	1	·

Parity

1	0	1	0	0	1	0	1	0