6.033 Lecture 9
Networking II

Americo De Filippo

June 8, 2023

References

[1] Saltzer, Jerome H. and M. Frans Kaashoek. Principles of Computer System Design: An Introduction (2009): **Section: 7.2**

Review of last time about best effort, which means that there are some proprety that won't be garateed.

1 Layering

As we have seen previusly a standard way of dealing with complexity is with modularity, and the modularity used into network system are layers and protocols (which is just a set of rules for make two device communicate in order to deal with complexity). We are gonna see speficiation of this layer in the next lecture starting with the end-to-end layer that will be in charge of the resolution of our best effort's problems.

2 Encapsulation

We are gonna achieve the layering via encapsulation layer by layer, inserting or removing his header/trailer for his information. Another important charateristic is the independecy of each layer, every layer don't know anything about the others layers and in way we are gonna achieve completely indipendecy by the modules (strong modularity).

3 Link Layer Issues

The link layer needs to provide: manager bits transmission via Digital -¿ Analog -¿ Digital conversion, it does framing which is basically how the software on the layer decide whether a new frame has began or ended. Another of them is Channel Access which is how someone that want to send a message used the

wire or the layer without intefering with another one which is transmetting [1]. The last thing that is done in this layer is error detection/correction, the idea is that when transmitting the message on a wire the message could be corrupted or there could be some problem with the transmission.

3.1 D to A to D coversion

Suppose we have a sender that want to send some kind of binary bits flow, that could be rappresented by a quadratic curve, we decide when sending a new bits every clock signal. But when the reciver reciev the data it will have not a perfect smooth signal and for this reason we want to decode the original message. This will be done knowing the frequency of the message [2] But what could happen is our clock signal could be with the same frequency but they are out of phase.

How to fix the out of phase problem The solution is called PLL (phase lock loop) so we basically need to know how much we need to shift our signal. A simple way of implementing this is called **oversampling 8x**, with this tequinique what we should see if we are perfetctly alligned in time the same number(8) of 0 or 1 for due original bits of the signal. And when this does not happen we are gonna to shift to the side of the one with less number of continuosily 1 or 0 [3].

How to fix long sequence of the same bit If I have a long sequence of the same bit is that i cannot know how much should i shift my signal. The solution to this is called Manchester Encoding: the idea is that we are trasmitting a zero for a transition to a low to a high and a 1 from a high to low. In this way i wont have a flat signal [4].

[1] read paper on ethernet
[2] here we know that there are a lot arguments like shannon's etc...
[3] see the slides (slides/L9.pdf) for more clarity
[4] review this on yt