

MIT WORLD PEACE UNIVERSITY

Computer Networks  
Second Year B.Tech Semester 3  
Academic Year 2022-23

---

---

MODULE 1 - CLASS NOTES

---

---

NOTES

Prepared By

P34. Krishnaraj Thadesar

Batch A2

August 29, 2022

## Contents

<b>1</b>	<b>Conversion Techniques</b>	<b>2</b>
1.1	Analog to Analog Conversion . . . . .	2
1.1.1	Amplitude Modulation . . . . .	2
1.1.2	Frequency Modulation . . . . .	2
1.1.3	Phase Modulation . . . . .	2
1.2	Digital to Analog Conversion . . . . .	3
1.3	Analog to Digital Conversion . . . . .	3
1.4	Digital to Digital Conversion . . . . .	3
1.4.1	Line Coding . . . . .	3
1.4.2	Unipolar . . . . .	4
1.4.3	Polar . . . . .	4
1.4.4	BiPolar . . . . .	4
1.4.5	Block Coding . . . . .	5
1.4.6	Scrambling . . . . .	5
1.4.7	Analog to Digital Conversion . . . . .	5
<b>2</b>	<b>Introduction to Medium Access Control</b>	<b>5</b>
<b>3</b>	<b>Network Architecture</b>	<b>6</b>
3.1	Client Server Model . . . . .	6
3.2	Peer to Peer Network . . . . .	6
3.3	Distributed Network . . . . .	6
3.4	Software Define Network . . . . .	6
<b>4</b>	<b>Protocol Hierarchies: Layers, Protocols and interfaces</b>	<b>6</b>
4.1	Layers . . . . .	6
4.2	Reference Models . . . . .	6
<b>5</b>	<b>IP Addresses</b>	<b>7</b>
5.1	Classes of IP Addresses . . . . .	7
5.2	Network Address . . . . .	7

## 1 Conversion Techniques

Analog to Analog or digital to Analog is modulation. If you convert Anything to digital then it is called encoding

### 1.1 Analog to Analog Conversion

#### 1.1.1 Amplitude Modulation

We change the amplitude, and keep the frequency and phase constant. The constant frequency is called the carrier frequency.

1. There is a modulation signal, which is the base analog signal, and there is a carrier frequency. We then combine them, and that produces the final amplitude modulated signal.
2. There is more noise in these signals.
3. Bandwidth:  $B_{am} = 2B$

Advantages are:

1. Easier to implement
2. Components are cheap
3. Can be demulated using a circuit consisting of a very few components

Disadvantages:

1. Poor Performance
2. Inefficient

#### 1.1.2 Frequency Modulation

It is the process by which the frequency of a carrier signal which changes with respect to the modulation signal. We just keep the amplitude and the phase constant, and we only change the frequency.

1. The logic is simple. If the input amplitude of the analog signal is more, then we will increase the frequency in that region. And vice versa. The bandwidth is also calculated in the same manner.
2. The transmitted signal also has a fixed value. So like when we transmit something, that audio in itself has some value of volume that you can output. Coz to increase the volume, you need to give more audio data. That volume we know is measured in Decibels (db).
3. The total bandwidth required for FM transmission, is  $B_m = 10B$ .

#### 1.1.3 Phase Modulation

Again over here the frequency and amplitude is constant. If the amplitude is high for the base audio signal, then you will shift the signal accordingly. It is costly, and takes more effort. The Bandwidth is again 10 times the bandwidth of the base signal.

### 1.2 Digital to Analog Conversion

Modulator obviously converts digital to analog. Coz we know that conversion to analog is called modulation. So the thing that does that is the modulator, and then on the receiver side, you will need a demodulator.

There are different techniques.

1. Amplitude Shift Keying : Coz we are converting from digital to analog, its rather simple. Coz you just have 2 values. It would be high or 0. So you transmit some wave when there is 1, and if its 0 then dont.
2. Frequency Shift Keying: Again this is also something similar. you have certain high frequency to transmit the 1, and a low frequency level to represent 0. So we have 2 base carrier waves. So the final wave has the combination of the 2 frequencies. This reduces noise.
3. Phase Shift Keying: To represent 1, you have a certain phase. And to represent 0, you change the phase by 180 degrees. That now generates a final output wave with varing phase. Now we dont require 2 frequencies. And that reduces noise.

### 1.3 Analog to Digital Conversion

### 1.4 Digital to Digital Conversion

This is encoding

1. Line Encoding
2. Block Encoding
3. Scrambling

#### 1.4.1 Line Coding

Converting a string of 1s and 0s into a sequency of signals that denote the 1s and 0s. So you would just need an encoder and a decoder. This way you could send a string of digital data to someone.

Refer to the Text here.

Example: A signal is carrying data in which one element is encoded as one signal element.  $r = 1$ . If the bit rate is 100 kbps. what is the average value of the baud rate if  $c$  is between 0 and 1. Assume that the average value of  $c$  is 0.5

Line Coding Schemes

1. Unipolar - NRZ
2. Polar - NRZ, RZ and biphase
3. Bipolar - AMI and pseudoterary
4. Multilevel - 2b/1Z, 8B/T etc.
5. Multi Transition - MLT 3

*Data Rate : It is the number of Bits set per second. Now to just transmit 1 bit, you might need 1 signal or 2 signals, or more. The value of a variable  $r$  = is data element / signal elements*

*Baud Rate: It is related to the signal. You could use 3 signal elements to send like 4 data elements, or 4 bits. So then value of  $r$  would be  $4/3$ . To Calculate  $S = c.N.(1/r)$*

*NRZ: non return to zero. Whenever the base signal has a change from 1 to 0, we will create a new signal, that doesn't return it to 0, we just flip the voltage. So this makes it such that the signal never has 0, and that makes it easier to key.*

### 1.4.2 Unipolar

Signal levels are on one side of the time axis either above or below NRZ scheme is an example of this code. The signal level does not return to zero during symbol transmission. It has no synchronization or error detection. It is simple but costly in power consumption.

### 1.4.3 Polar

Non return to Zero - Level - Positive voltage for one symbol and negative for the other.

NRZ - Inversion - The change or lack of change in polarity determines the value of a symbol. Eg. A 1 symbol inverts the polarity a "0" does not. Basically we only change when the bit value is actually changing. This means unless you change, the computer will assume that you have the same value. This prevents any confusion as to how many values are there.

The voltages for Polar are on both sides of the time axis. Polar NRZ scheme can be implemented with two voltages. Eg. +V for 1 and -V for 0

Return to Zero (RZ)

1. This scheme uses three voltage values. +, 0 and -
2. You basically return to zero after each transmission.

### 1.4.4 BiPolar

1. So voltage level for one symbol is at 0 and the other alternates between + and -.
2. Bipolar alternate Mark Inversion AMI - The 0 symbol is represented by zero voltage and the 1 symbol alternates between +V and -V
3. Pseudoternary is the reverse of AMI. Here 0 is represented by alternate voltage and 1 is represented by 0.
4. AMI is used in USA
5. Pseudoternary is used in Europe.

### 1.4.5 Block Coding

A technique of sending data in a set of sequence.

Block coding is done in Steps. You are basically wanting to convert a bunch of bits into a bunch of blocks. The base block may be of some value  $m$ , and you might convert it to another block of size  $n$ , which may be greater or lesser than  $m$ . This is  $mB$  to  $nB$  substitution.

### 1.4.6 Scrambling

1. Earlier coding schemes are not suitable for long distance because of DC components.
2. Scrambling is a technique to avoid long sequence of 0s used to create a sequence of bits that has the required bandwidth for transmission. Self clocking, no low frequencies, no wide bandwidth.
3. It is implemented at the same time as encoding the bit stream is created on the fly.
4. It is implemented at the same time as encoding, the bit stream is created on the fly.
5. It replaces unfriendly runs of bits with a violation code that is easy to recognize.
6. The reason we are doing this is mainly just because we know that for long distances DC signals don't travel well. So when you send long strings of data, like say 8 0's, then some may get lost somewhere. To avoid that, we scramble the data, by violating it. When you have 8 zeroes, the sender realizes that, and after 3 zeroes, it flips the data bits as + - 0 - +. So the receiver realizes that there were 8 zeroes. The value of the violated signal is different from the signal for 1 or 0.

### 1.4.7 Analog to Digital Conversion

- (a) A digital signal is superior to analog signal. It is more robust to noise and can easily be recovered, corrected and amplified.
- (b) For this reason it is better to change an analog signal to digital data.

## 2 Introduction to Medium Access Control

### Type of Networks

- Personal Area Network : 1 m
- LAN upto 1 km
- MAN upto 10km
- WAN upto 1000km : Mostly with Fiber and Serial Links.
- More than that would be the Internet.

**802.11** Laptops with recent technology have this tech in it. A wireless router is connected to the wired network.

**Switched Ethernet** Ethernet Switch is connected to the rest of the network.

**Ad Hoc Network** Any temporary network where the devices are connected for sometime is called the ad hoc network. Bluetooth

### 3 Network Architecture

#### 3.1 Client Server Model

This is like the usual browser and html thing. You send a request, and you get a reply over the network.

#### 3.2 Peer to Peer Network

No centralized data source. Its the ultimate form of democracy on the internet. Any node can initiate a connection.

#### 3.3 Distributed Network

This is a critical thing, and most important commercial networks fall into this. Its simple, there are several servers, who are connected to other computers of their own. These servers are then connected with each other.

#### 3.4 Software Define Network

This is a new type coz there is an application layer, a control layer, and a forwarding layer. A single network dude cant handle thousands of applications, and so there is a software that controls them. Think of apps. The main difference here is that the network is controlled by the software.

### 4 Protocol Hierarchies: Layers, Protocols and interfaces

#### 4.1 Layers

- The concept here is that data communication takes place in layers. There are several layer Interfaces.
- Layer 1 for example would be the Communication between NIC and Wireless card of your system with the Cellular Network Tower. Layer 2 would be between the Servers of your application. etc.

#### 4.2 Reference Models

- OSI Model: Open system Interconnection Model.  
Imagine you have opened several tabs on your Browser on your Lap. You are connected to the Google servers via serveral other computers. Your First base physical layer would be the fact that you have a router connected via 2.4Ghz wifi. Your second data link layer, would be your NIC Card that actually sends the bits of data to be sent over the network. But the packet of data that is sent has to be received at the destination. That is what happens at the network Layer. Between you and the Google network, there are several other computers, that receive these packets till their network layer. A packet leaves your computer, with your IP address, and bounces between these computers, where it goes through several checks, and

finally reaches the Google Server. The Browser application is yet another layer, which manages what information to send. Each browser maintains several sessions in each of those tabs. That is yet another layer. Even above the Browser application layer is not the last one, there is yet another presentation layer, between the browser itself that manages presents the final form of data to you.

In this example, we see 7 layers, present everywhere on every computer. 2 of them are physical layers, the rest 5 need to be in your OS.

**Addresses** The Network layer uses your IP Address. the Data Layer is where the MAC Address is defined. The Physical layer doesn't have an address. Data sent from a computer is sent in packets. We know that. But these packets are abstracted in the network layer. At the data link layer, you call them frames, which then make up packets. Frames that have headers that contain the Source and Destination MAC Address. Network layer has packets arranged in the same manner. At the transport layer, we use Port addresses. These addresses are associated with the processes running on your machine. Like what happens when you run a Jupyter Notebook file.

IP Addresses are not known, and we have to send a request to the DNS which then gives us the exact IP address.

## 5 IP Addresses

### 5.1 Classes of IP Addresses

There are multiple networks, and they all have to be connected to each other some how.

a usual IP address would be 192.168.1.1 from Class C

There are multiple classes for using IPs

1. Class A: it has |Network|Host|Host|Host|

There are 4 integers, so 32 bits, A has certain starting bits that define what the range of the first bit is gonna be like A would have 0 in the first bit fixed, so naturally 7 would be 1. Rest of the bits are also 1.

So we have range like 127.255.255.255

2. Class B: It has |Network|Network|Host|Host|

128.255.255.255 to 191.255.255.255

3. Class C: It has |Network|Host|Host|Host|

192.255.255.255 to 223.255.255.255

127.0.0.0 is reserved for loopback, so you can ping yourself. anything.255.255.255 is reserved for broadcast address, and so that would be if you wanna send something to the entire class.

### 5.2 Network Address

It has all 0's in the Host place, and so it refers to a range. So then the class is A, coz its between 0 and 1. The block has a netid of 17, and the range is from 17.0.0.1 to 17.255.255.254 eg. Given the network address 17.0.0.0,

eg. 132.21.0.0 - is again for the network, as there are 0's in the host

Class would be B for this.



Range would be from 132.21.0.1 to 132.21.255.254  
Block netid is 132.21

eg. 220.34.76.0  
This is class C obviously.

### Default Subnet Mask

When are the network bits are 1, it becomes a subnet mask. You need this so that the computer has to know what class you are talking about. If you are using like class A, then you have an IP, but to differentiate between the network and host bits, you have to tell the computer that its gonna be 255.0.0.0, so they know what the hosts are.

The number of bits used by Class a would be 8 for network, and 24 for host. for b it is 16 bits for network, and 16 for host. The number of machines would then  $2^{value}$  Now these are your public addresses. So they have to be unique. But there are a lot of machines that you can use here. Say you just wanna use private IPs for your lan, then you can have that too, and it can be and has to be repeated.

So like every private lan address might have 192.168.0.1. But your ip on the web will be different.

### 5.3 Borrowing Subnet bits

if the number of bits you b