SFWR ENG 4J03

Author: Kemal Ahmed

Instructor: Dr. Gowri Krishnasamy

Winter 2016

*Math objects made using* [*MathType*](http://www.dessci.com/en/products/mathtype/)*.*

Table of Contents

[Abbreviations 1](#_Toc442304923)

[Angle Modulation 1](#_Toc442304924)

[Frequency Modulation 1](#_Toc442304925)

[Phase Modulation 1](#_Toc442304926)

[Amplitude Modulation 1](#_Toc442304927)

[Generating SSB-SC 1](#_Toc442304928)

[Frequency Discrimination Method 1](#_Toc442304929)

[Phase Discrimination Method 1](#_Toc442304930)

[Information Theory 1](#_Toc442304931)

[Shannon-Fano 1](#_Toc442304932)

# Abbreviations

(O/P): Output

# Angle Modulation

**Signal to Noise Ratio (SNR)**:



## Frequency Modulation

**Quantizer**:

**Quantization**: truncates, rounds

**Modulation**: the process of varying one or more properties of a periodic waveform, called the carrier signal, with a modulating signal that typically contains information to be transmitted

**Angular Frequency** [ω]: 2πf

**Message Angular Frequency** [ωm]: 2πfm

**Carrier Angular Frequency** [ωc]: 2πfc

**Angle Modulation**: frequency or phase modulation

**Frequency Modulation (FM)**:

**Phase modulation (PM)**:

**Demodulation**:

**Inductance** [L]:

**Capacitance** [C]:

**Message Frequency** [fm]:

**Carrier Frequency** [fc]: 

**Instantaneous Frequency** [fi]: fc + kf m(t)

**Oscillator**: produces a signal that converts the digital message into analog signal

* requires very precise frequency and phase to match the carrier

**Difference Signal**: oscillator frequency – input signal

**Balanced Modulator**: frequency translations

**Bandwidth (BW)**:

**FM Bandwidth** [BT]: 

**Peak Frequency Deviation** [Δfc]: kfAm

**Narrow Band Frequency Modulation (NBFM)**:

**Wide Band Frequency Modulation (WBFM)**:

**Frequency Modulation index** [β]: max frequency deviation / fm

## Power

**Power**:

**Carrier Signal Power** [Pc]:

**Power Spectral Density** [SM(f)]:

**Message Power** [Pm]: PUSB + PLSB

[PUSB]: m2/4

[PLSB]: m2/4

**Total Power** [Ptotal]: 

**Peak Power** [Pp]: 

**Harmonics**: when waves build up…

**Audible frequency range**:

**Audio modulating frequency range**:



fourier: 764

**Frequency-Division Multiplexing (FDM)**:

## Phase Modulation

hi

# Amplitude Modulation

**Amplitude Modulation (AM)**: modulated signal contains two side bands and an unmodulated carrier signal

**Maximum Amplitude** [Am]:

**Message Signal** [m(t)]: m(t) = Am cos(ωmt)

a.k.a. modulating signal, unmodulated signal, data signal, information signal

**Carrier Signal** [c(t)]:

**Modulated Signal** [s(t)]: Output (O/P)

s(t) = m(t) × c(t)

[AAM]: Ac + m(t)

**DSB-AM**: a.k.a. conventional AM



**Amplitude Modulation Index** [m]: 

**AM BW**: max – min = (fc + fm) – (fc – fm) = 2∙fm

**Band Pass Filter (BPF)**:

**Vestigeal Sideband Transmission (VSB)**: one side band along with just a trace of the other side band (a.k.a. **vestige**). This trace is useful for ensuring that important information is not cut off when reading  
  
Filters, such as BPF, tend to remove a little bit of the message. You can avoid this by extending the length of the transmission and including a trace of the opposite SB.

**Message Bandwidth** [W]: maximum message frequency, i.e. message

**SideBand (SB)**:

**Suppressed Carrier (SC)**: don't transmit carrier signal, so less power, but complicated filter  
  
**Double SB-SC (DSB-SC)**: USB & LSB

**Single SB-SC (SSB-SC)**: transmit only one sideband frequency, usually a DSB through a BPF  
  
**Lower SB (LSB)**: (fc – fm)  
  
**Upper SB (USB)**: (fc + fm)

**Envelope Detection**: a demodulation method that converts AM to m(t), using c(t)

## Generating SSB-SC

1. Frequency discrimination method
2. Phase discrimination method

### Frequency Discrimination Method



### Phase Discrimination Method

π/4 phase shift → product modulator → S1(x) → Sum → SSB-SC S(k)

# Current

[IC]:

[IT]:

# Information Theory

**binits**: binary bits

## Shannon-Fano

**Shannon-Fano code**: finds efficiency of code, listed with probabilities in decreasing order

**Ensemble**: source of the messages

1. Split into 2 groups as similar in size as possible without first rearranging. Sometimes it may be more efficient to put a smaller group on top because it is more probable and will require less bits.
2. Allocate 1s to one group and 0s to the other. Either put 0s on all the top groups or 1s in all the top groups
3. Split your groups into smaller groups
4. Continue partitioning until you only have groups of size one.

## Huffman Coding

[N~]: average number bits per message

**Code efficiency** [η]:

