

# COM-405: Mobile Networks

## Lecture 2.0: Modulation & OFDM Haitham Hassanieh





Channel (Single Path):

- Adds Noise
- Delays the Signal
- Attenuates the Signal
- Rotates the Phase of the Signal

$$h \propto \frac{\lambda}{d} e^{j2\pi d/\lambda}$$

Multipath Channel:

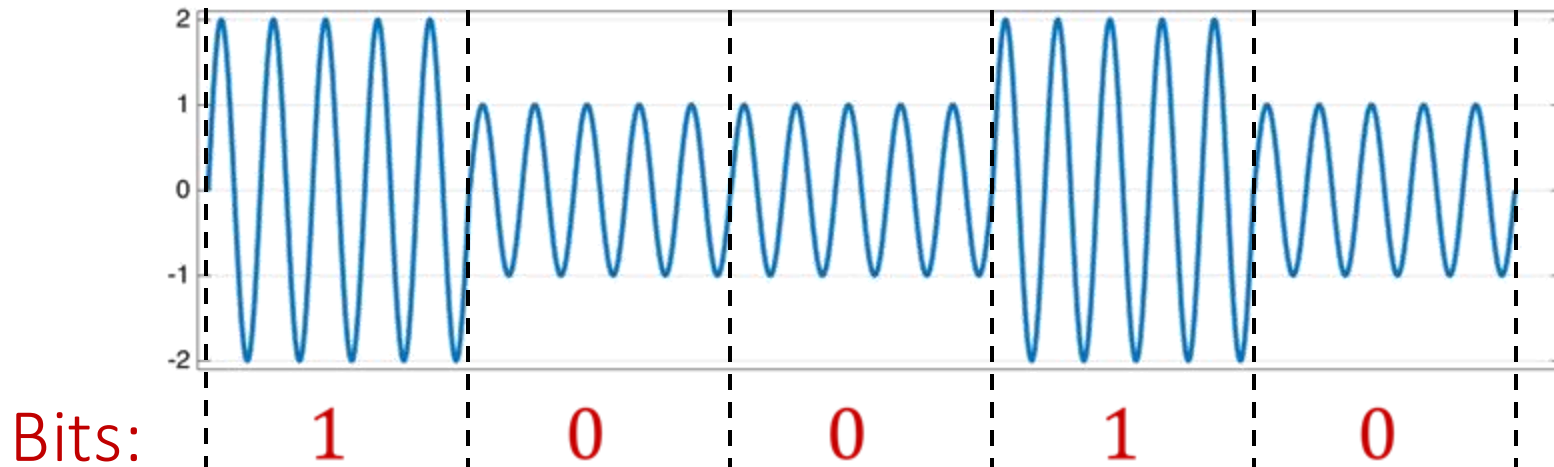
- ISI: Inter-Symbol-Interference
- Frequency Selective Fading
- Difficult to estimate and correct for

# Modulation

Based on how the bits are encoded

ASK

Amplitude  
Shift Keying



# Modulation

Based on how the bits are encoded

ASK



Amplitude  
Shift Keying



1 0 0 1 0

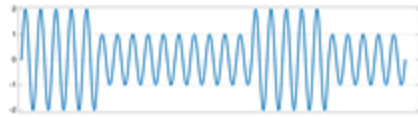
# Modulation

Based on how the bits are encoded

ASK



Amplitude  
Shift Keying

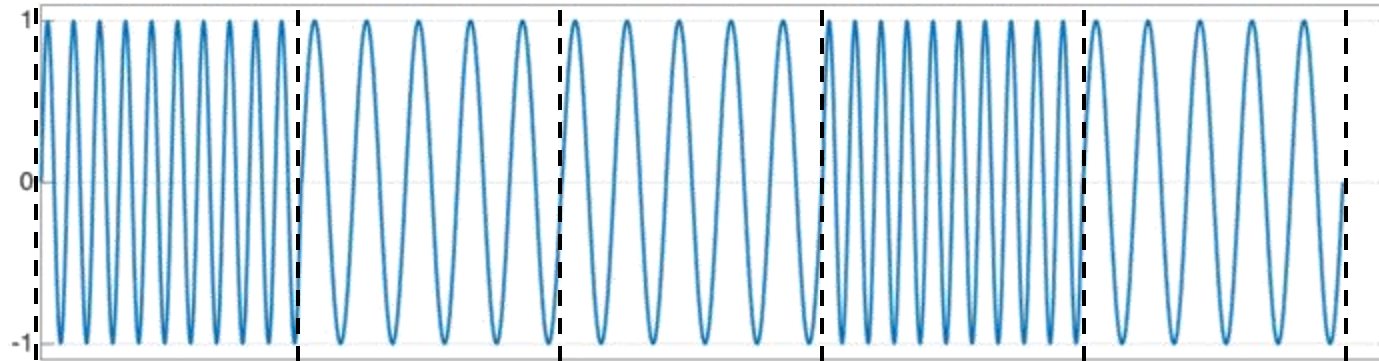


1 0 0 1 0

FSK



Frequency  
Shift Keying



Bits:

1

0

0

1

0

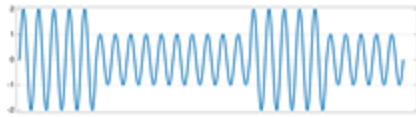
# Modulation

Based on how the bits are encoded

ASK



Amplitude  
Shift Keying

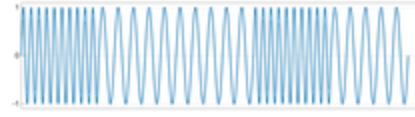


1 0 0 1 0

FSK



Frequency  
Shift Keying



1 0 0 1 0

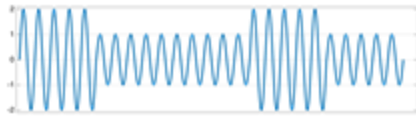
# Modulation

Based on how the bits are encoded

ASK



Amplitude  
Shift Keying

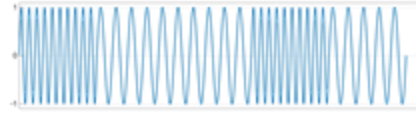


1 0 0 1 0

FSK



Frequency  
Shift Keying

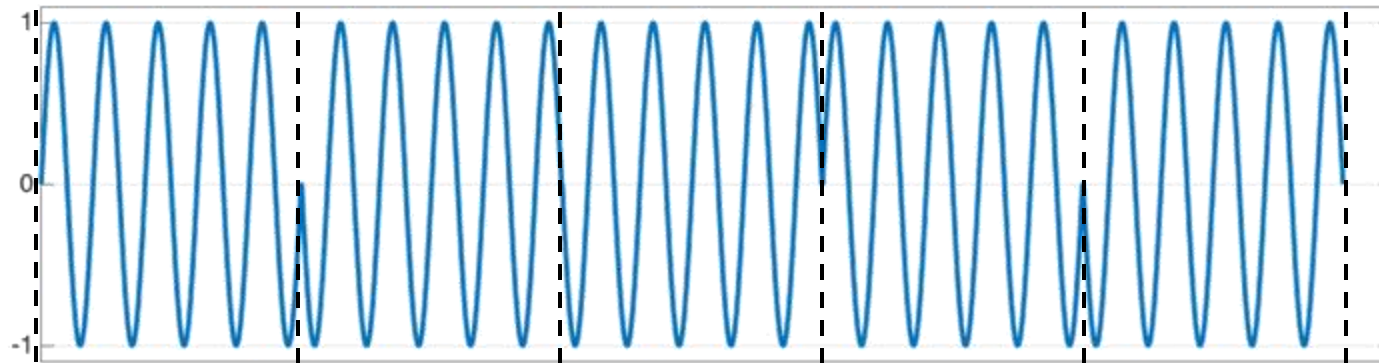


1 0 0 1 0

PSK



Phase  
Shift Keying



Bits:

1

0

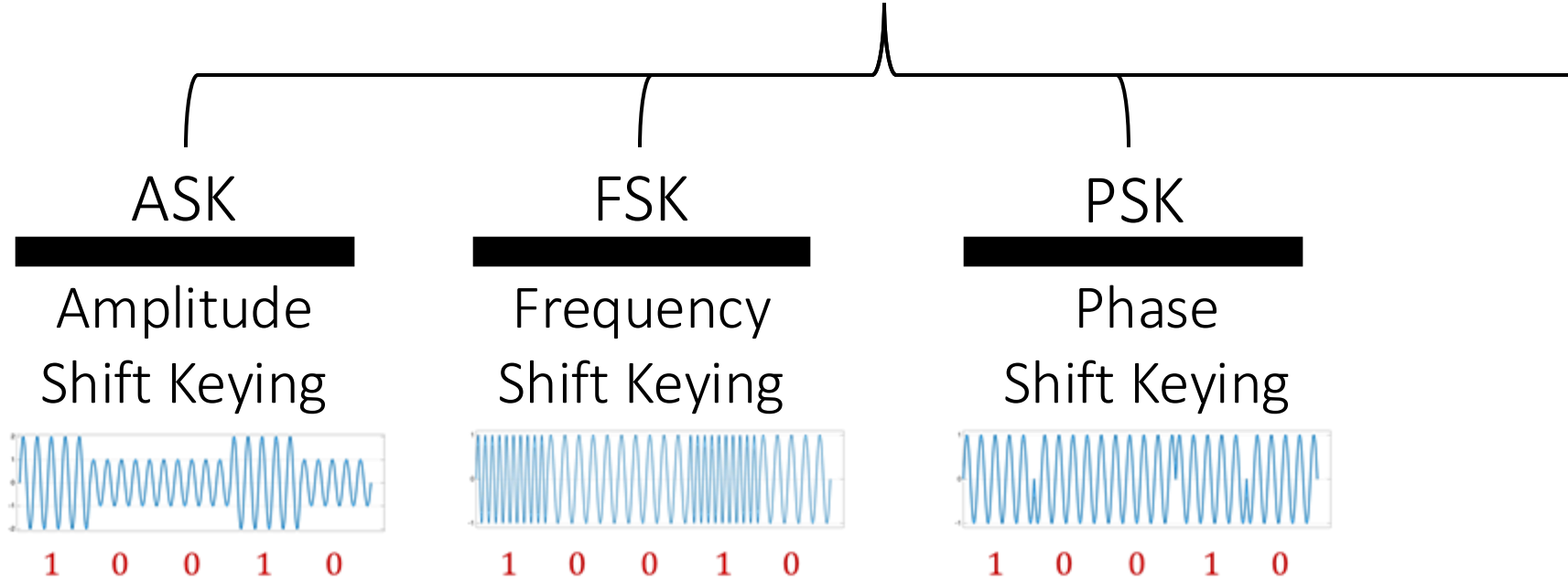
0

1

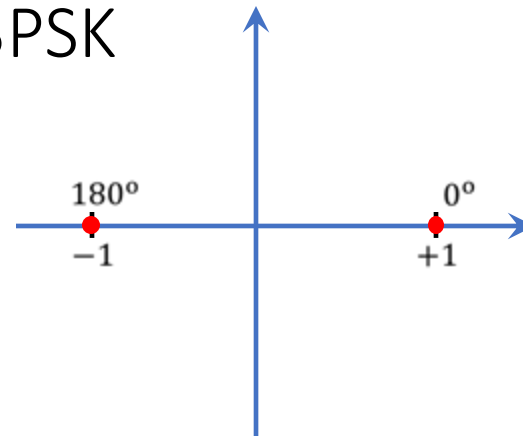
0

# Modulation

Based on how the bits are encoded

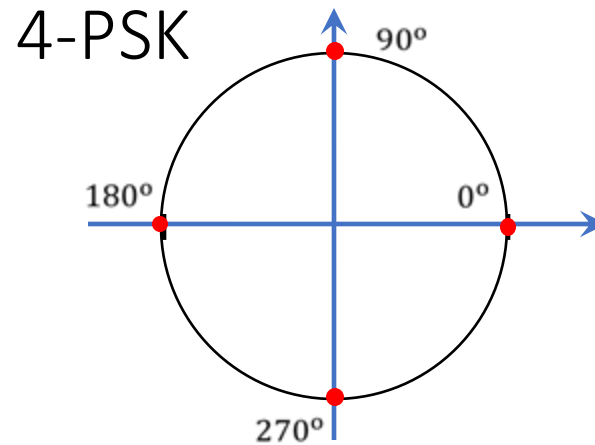
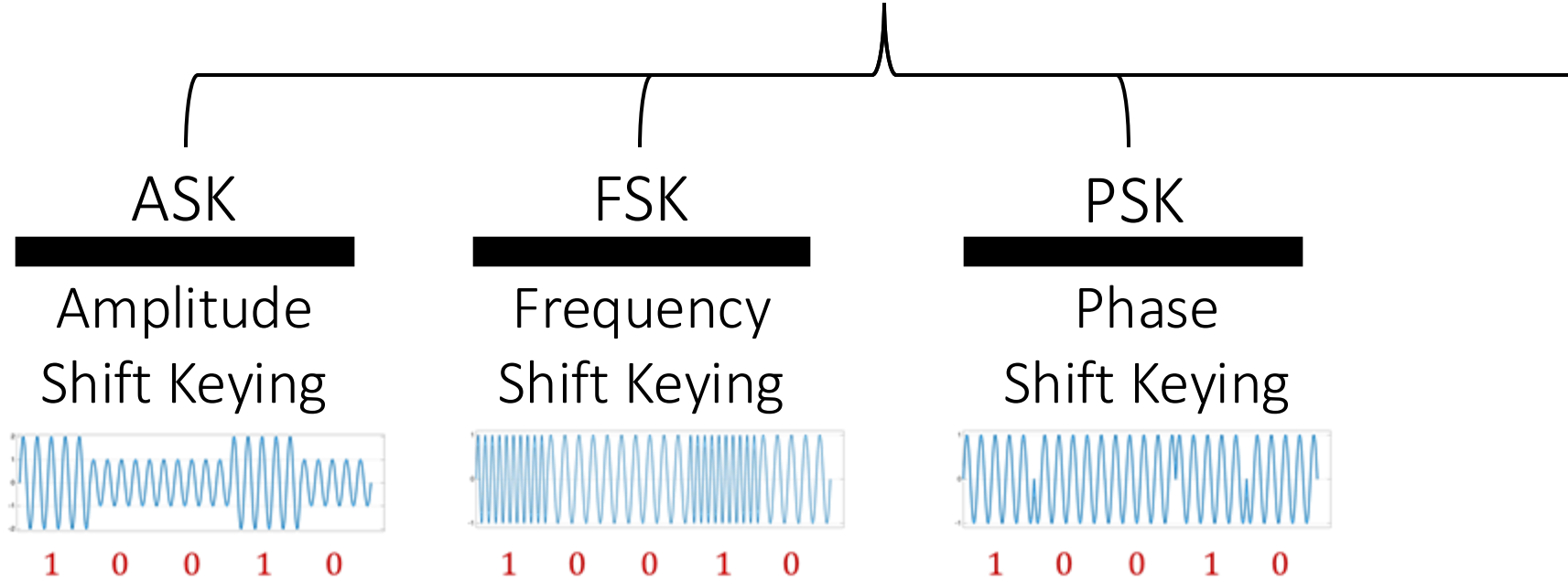


BPSK



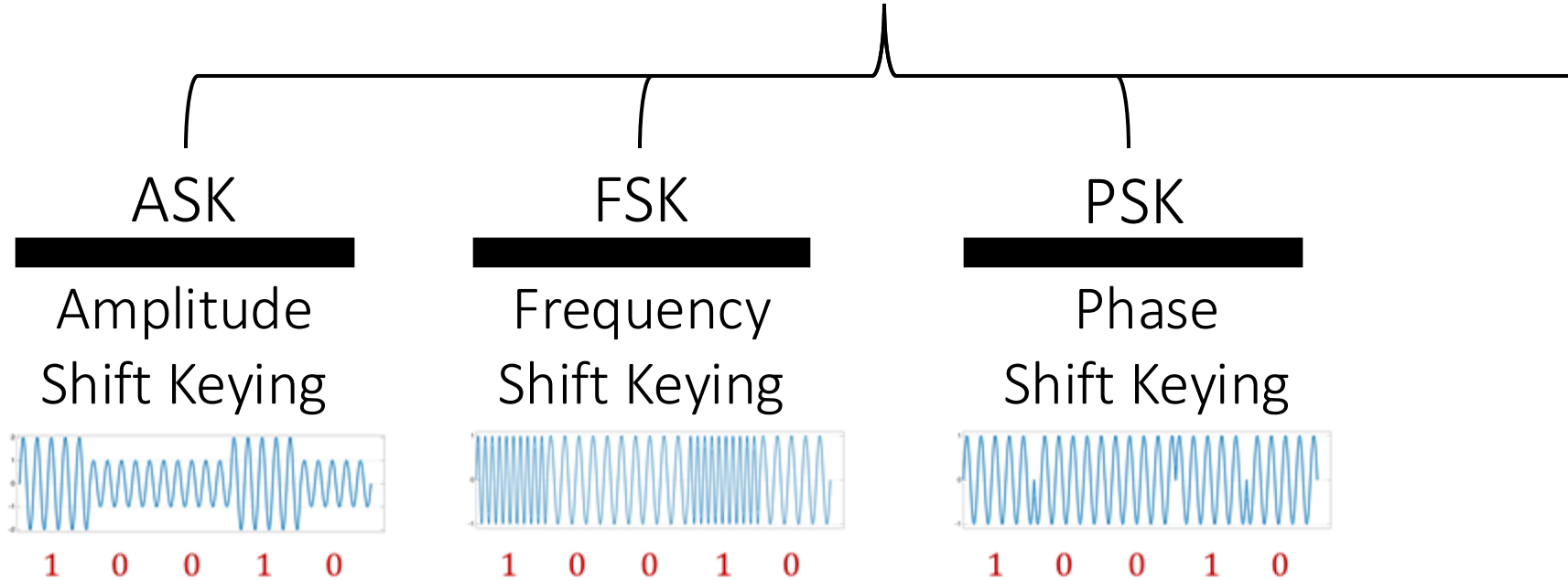
# Modulation

Based on how the bits are encoded

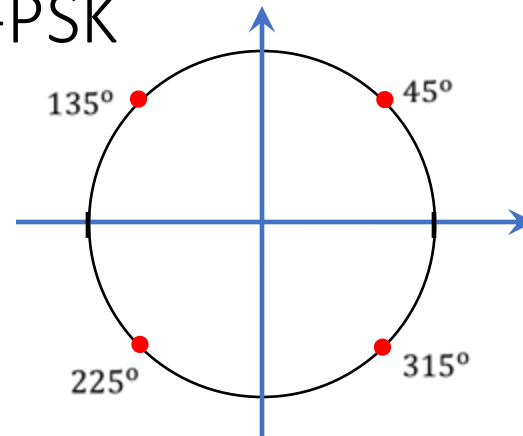


# Modulation

Based on how the bits are encoded



4-PSK



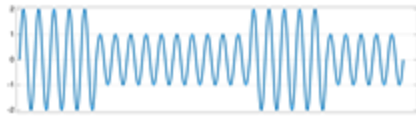
# Modulation

Based on how the bits are encoded

ASK



Amplitude  
Shift Keying

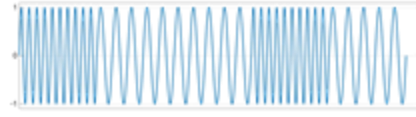


1 0 0 1 0

FSK



Frequency  
Shift Keying

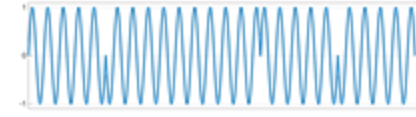


1 0 0 1 0

PSK

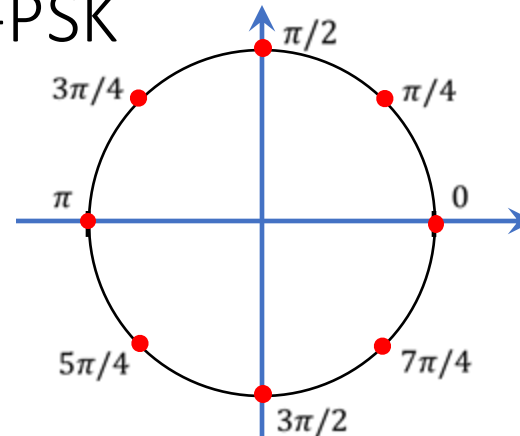


Phase  
Shift Keying



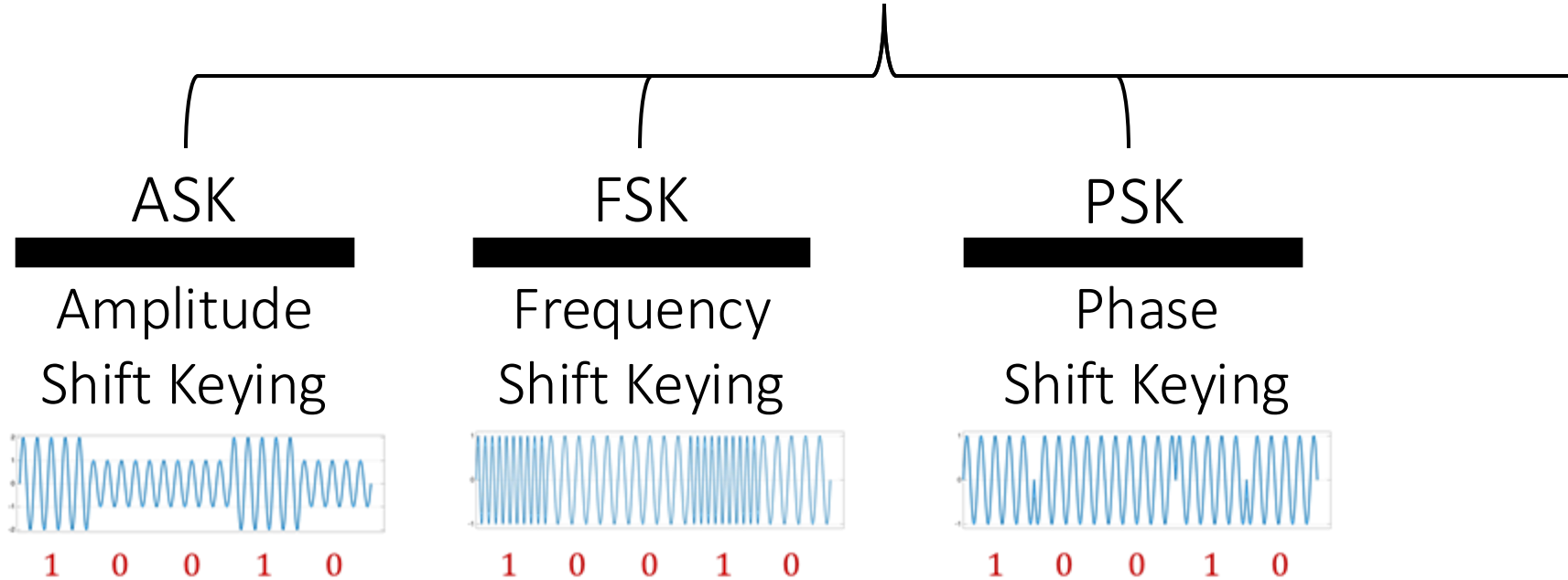
1 0 0 1 0

8-PSK



# Modulation

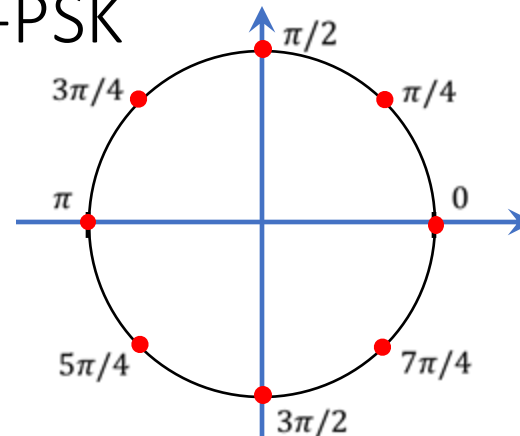
Based on how the bits are encoded



M-PSK

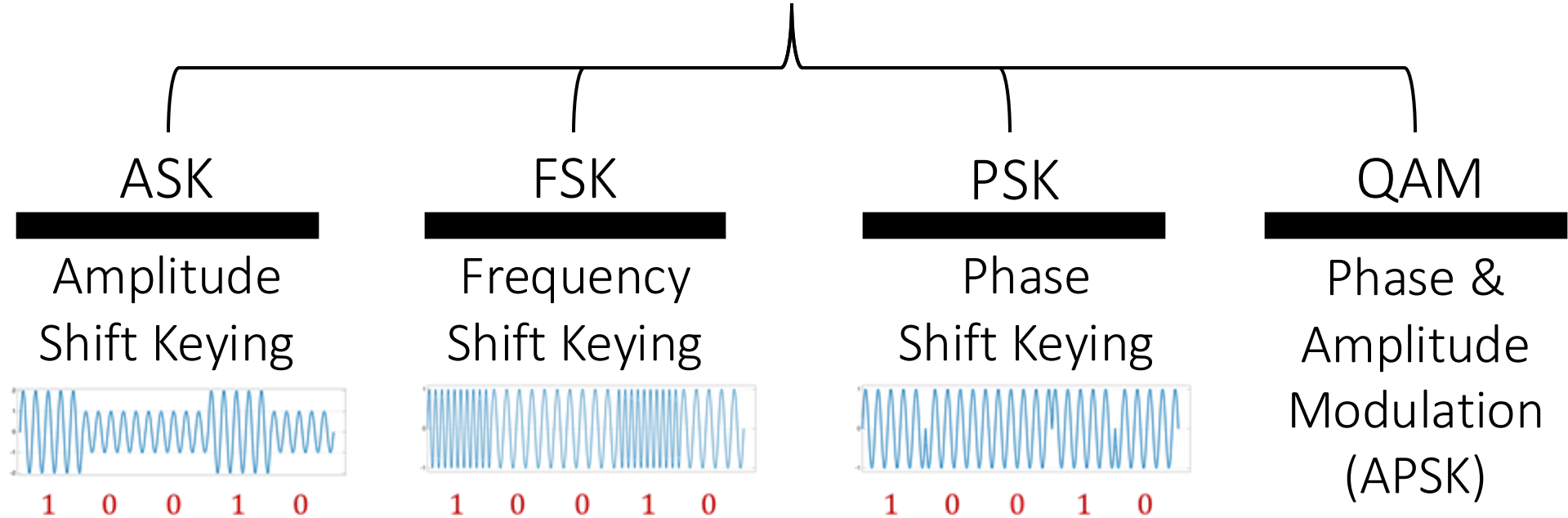
Phase is integer multiple of:  $\frac{2\pi}{M}$

Bits per symbol:  $\log_2 M$



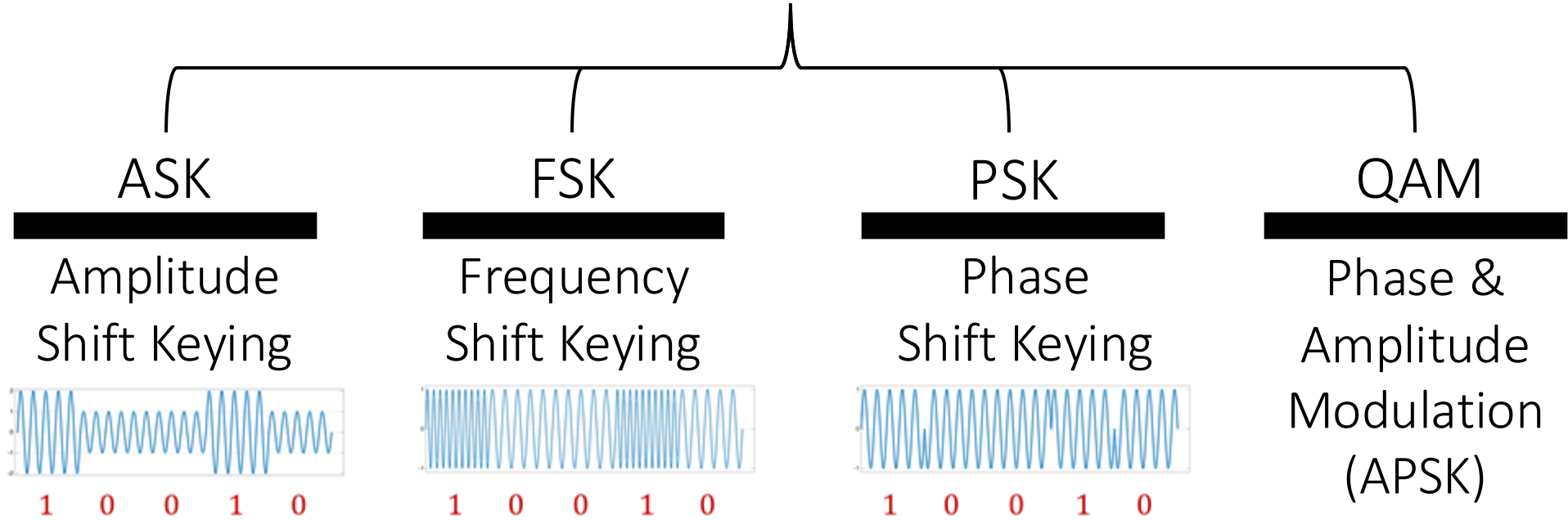
# Modulation

Based on how the bits are encoded

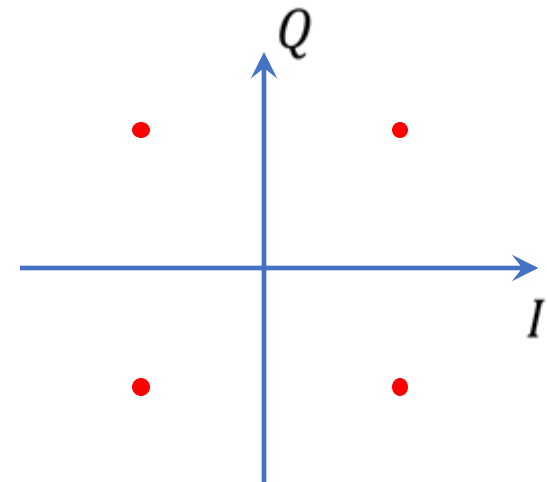


# Modulation

Based on how the bits are encoded

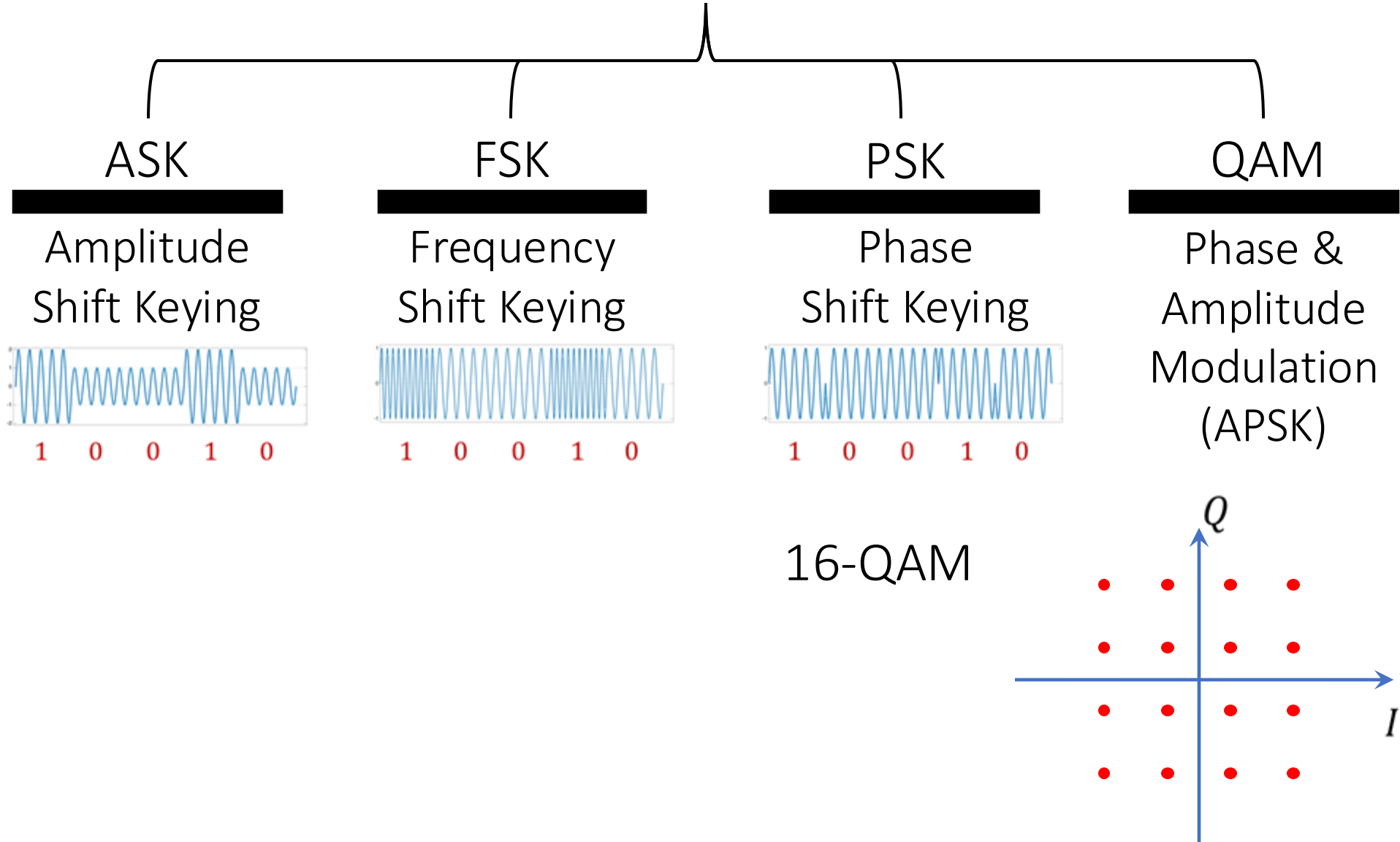


4-QAM



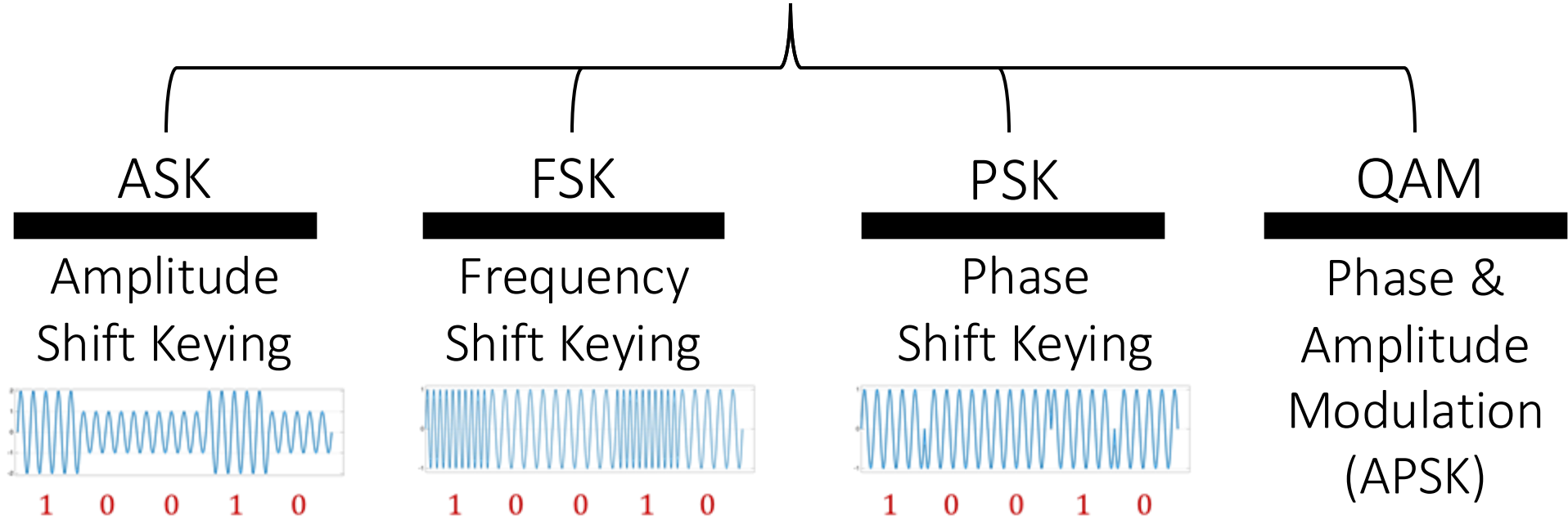
# Modulation

Based on how the bits are encoded

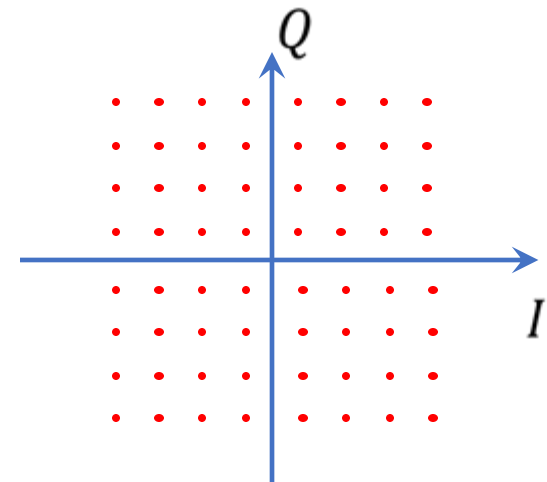


# Modulation

Based on how the bits are encoded

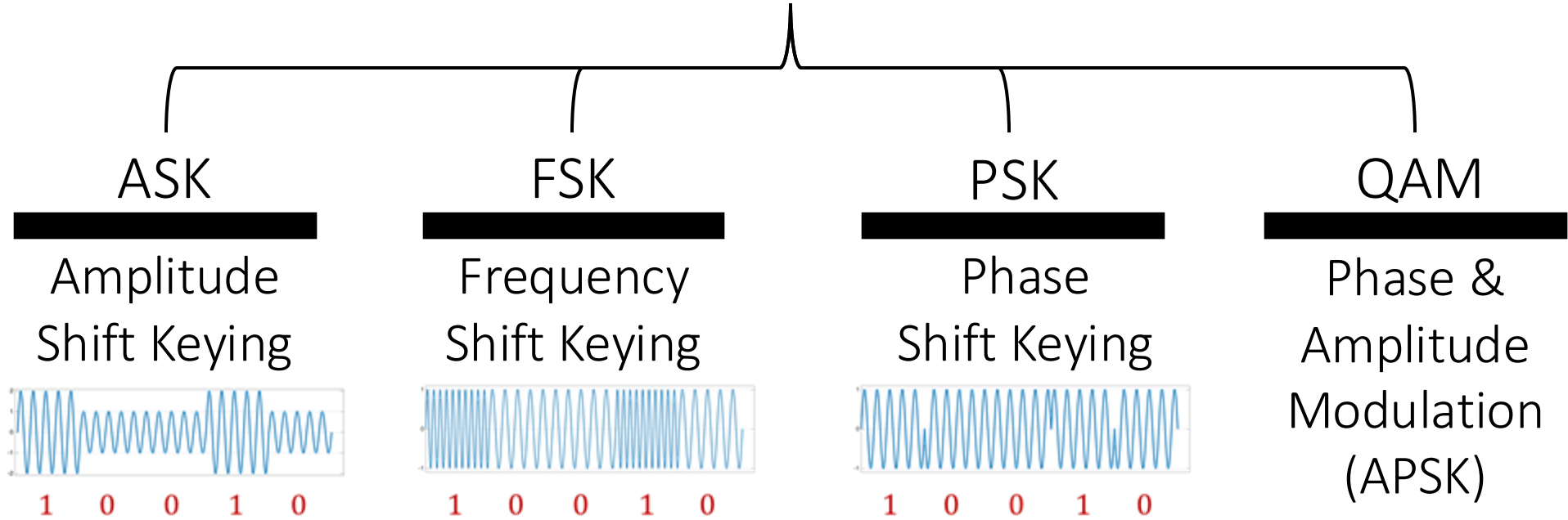


64-QAM

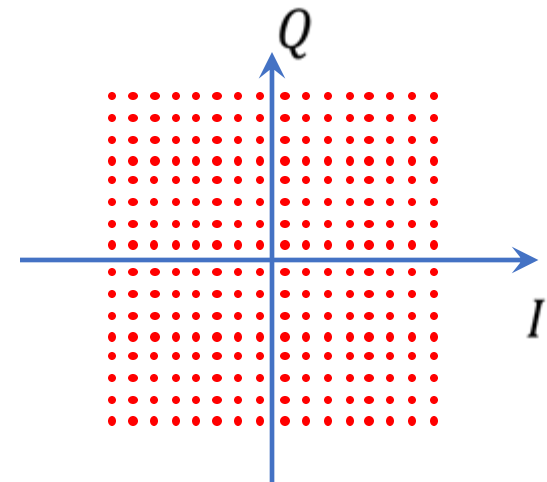


# Modulation

Based on how the bits are encoded

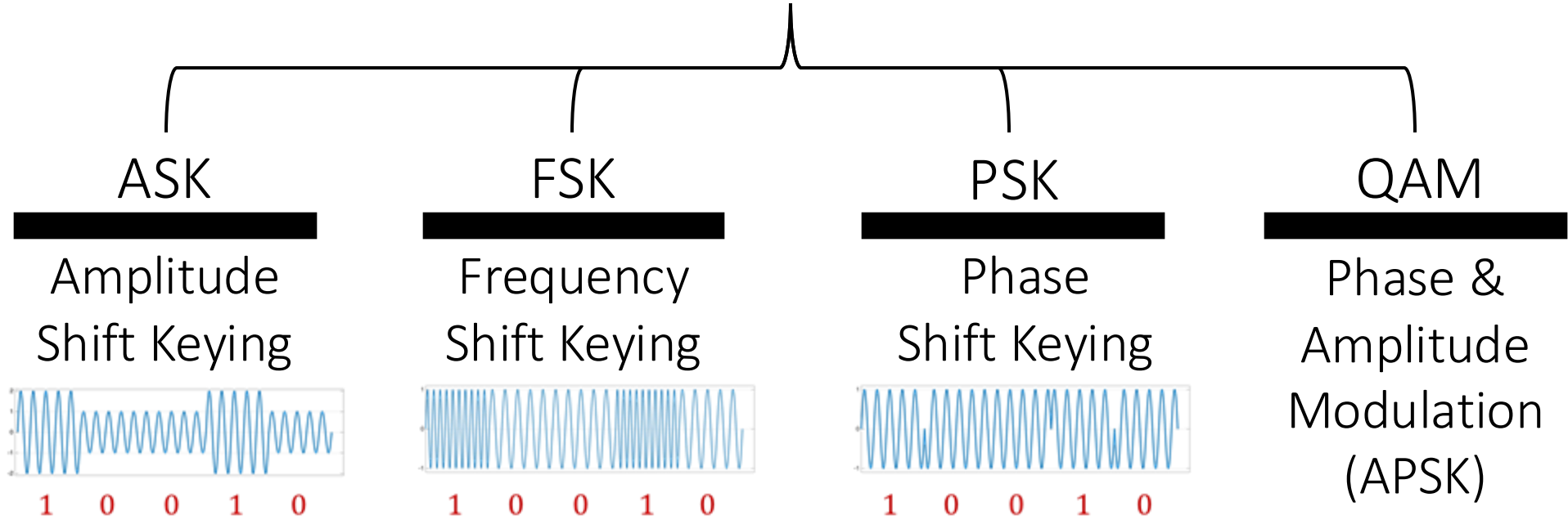


256-QAM



# Modulation

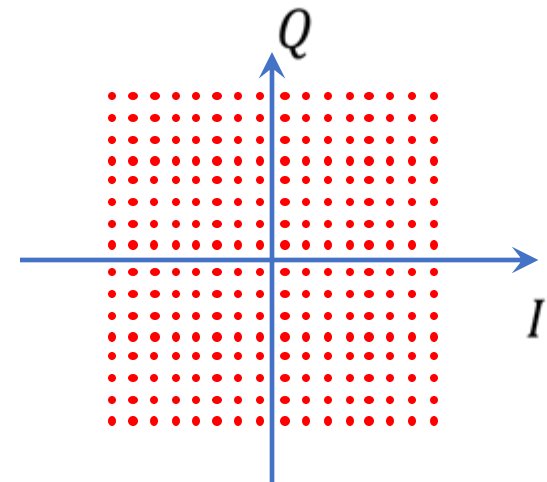
Based on how the bits are encoded



M-QAM

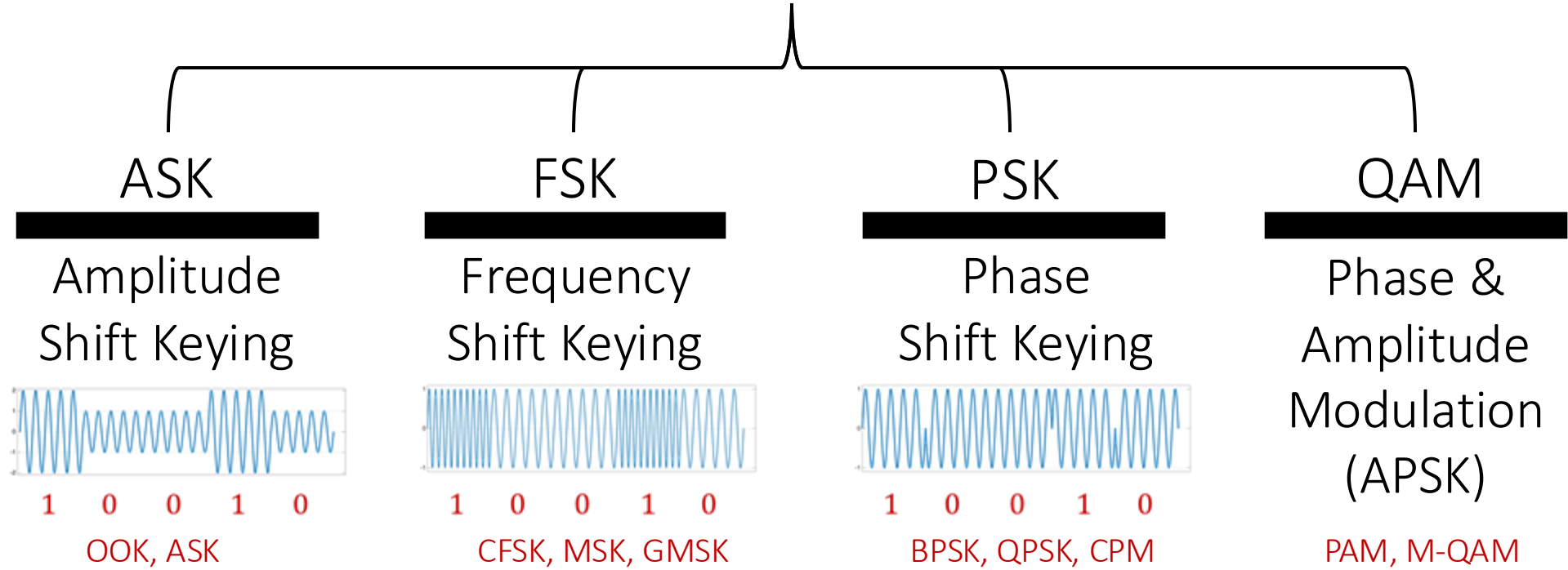
$$\text{IQ grid: } \sqrt{M} \times \sqrt{M}$$

$$\text{Bits per symbol: } \log_2 M$$



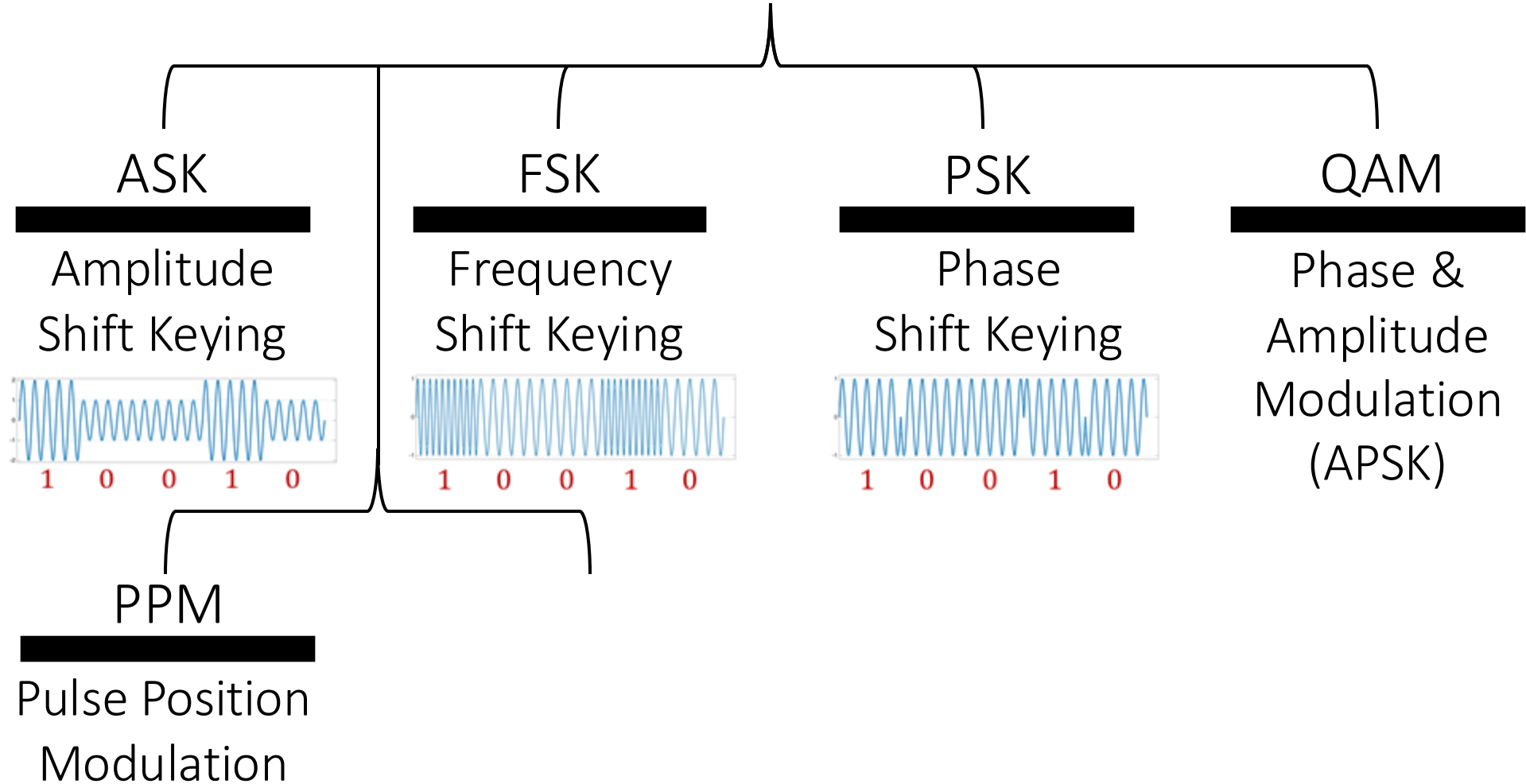
# Modulation

Based on how the bits are encoded



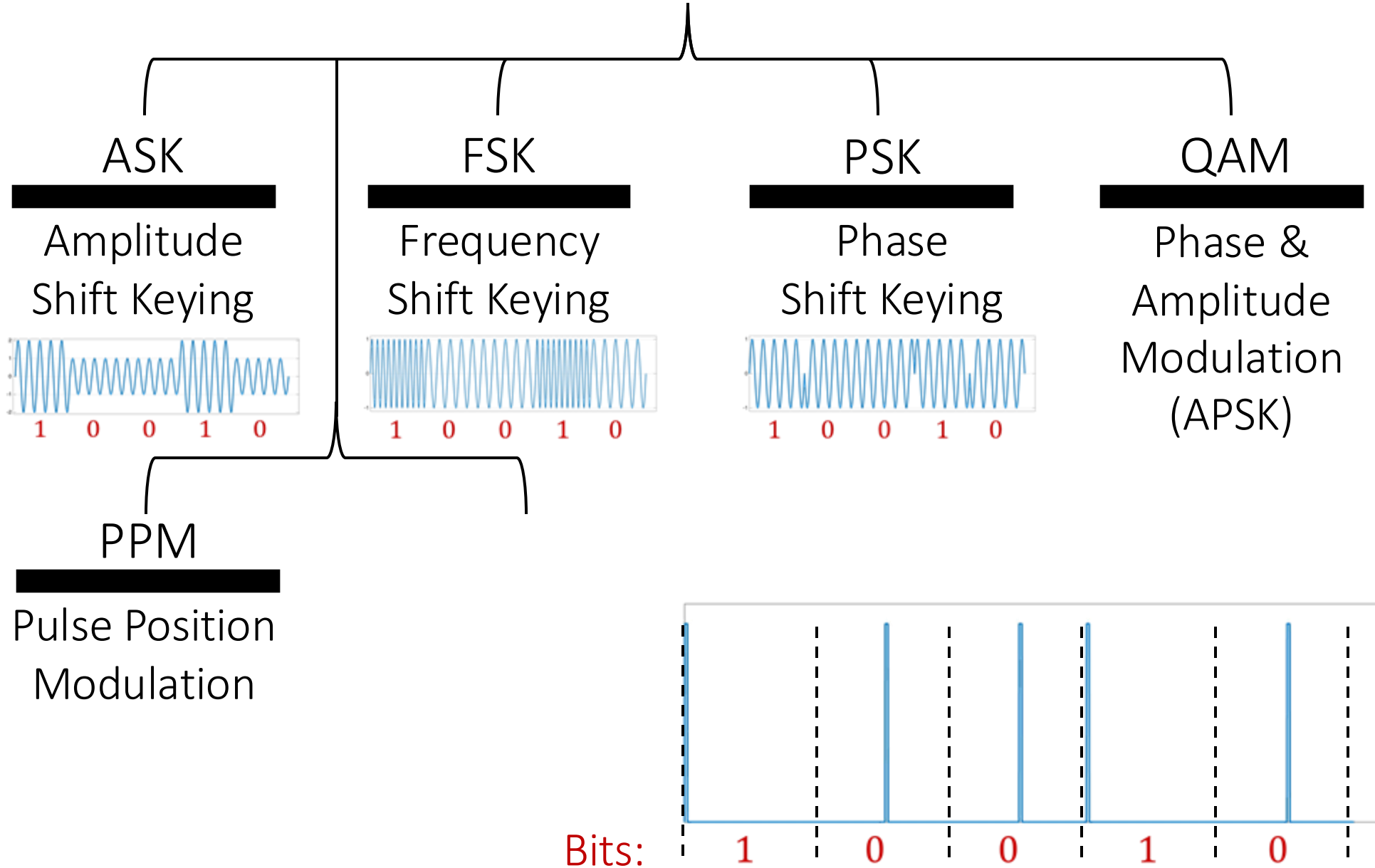
# Modulation

Based on how the bits are encoded



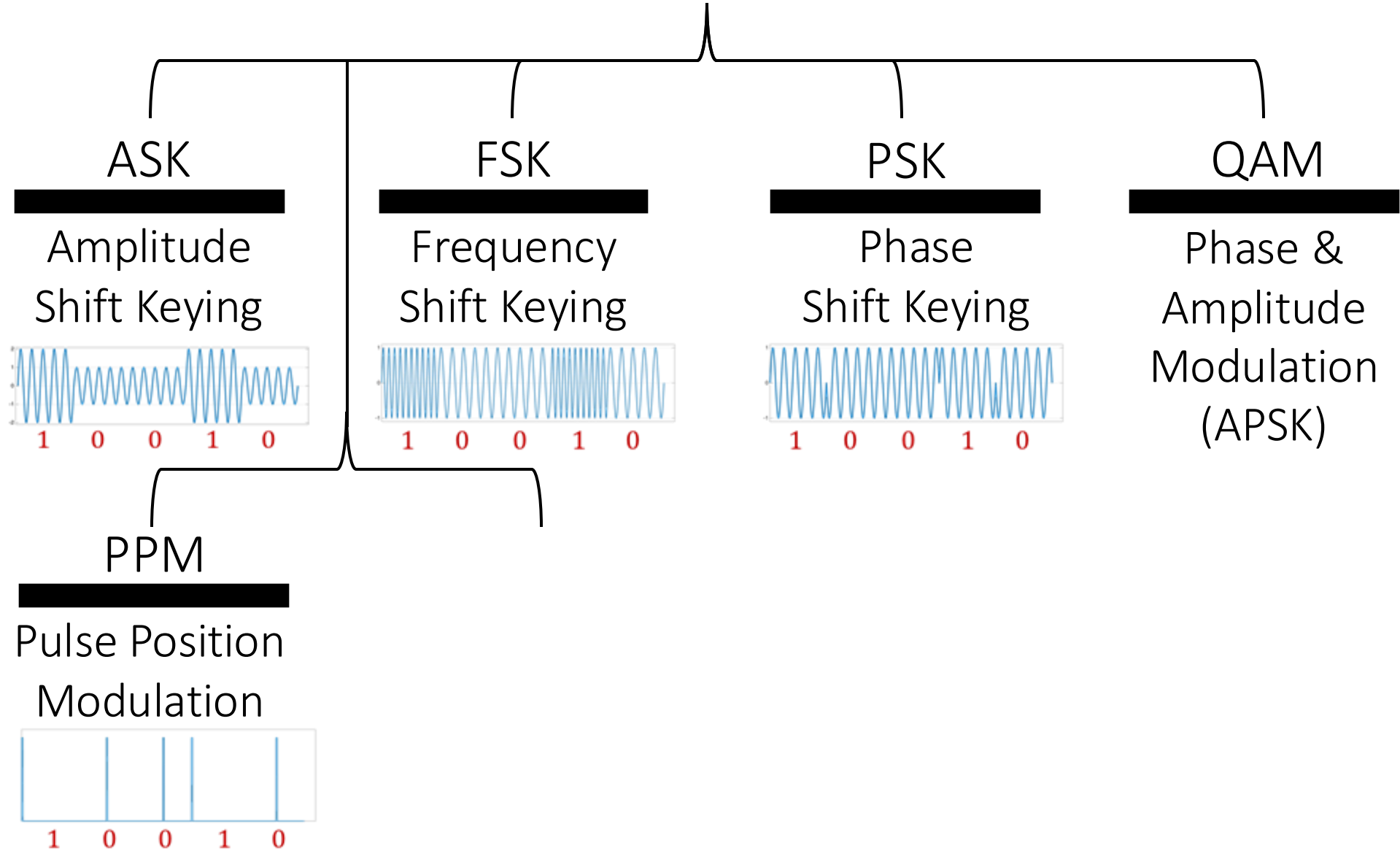
# Modulation

Based on how the bits are encoded



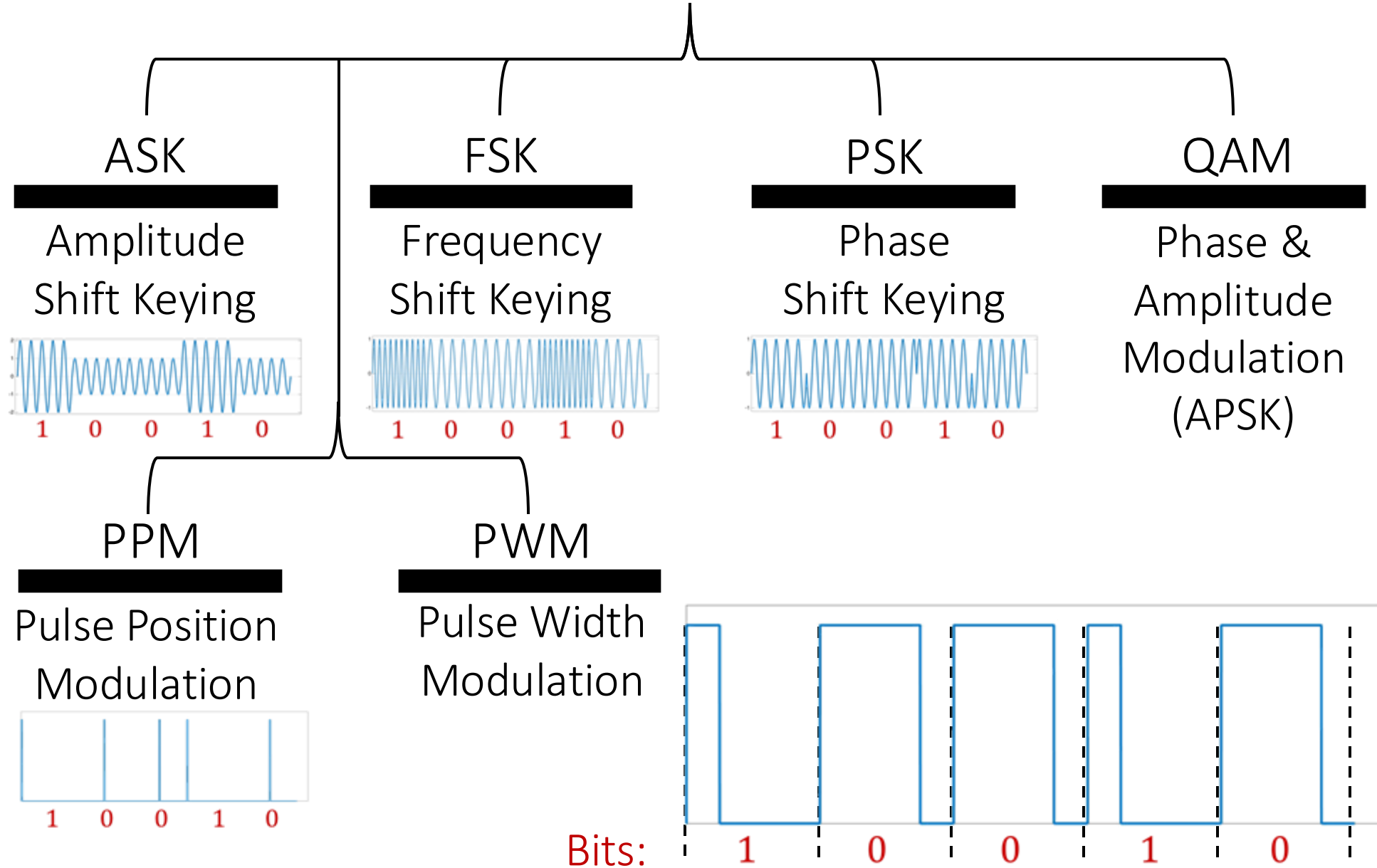
# Modulation

Based on how the bits are encoded



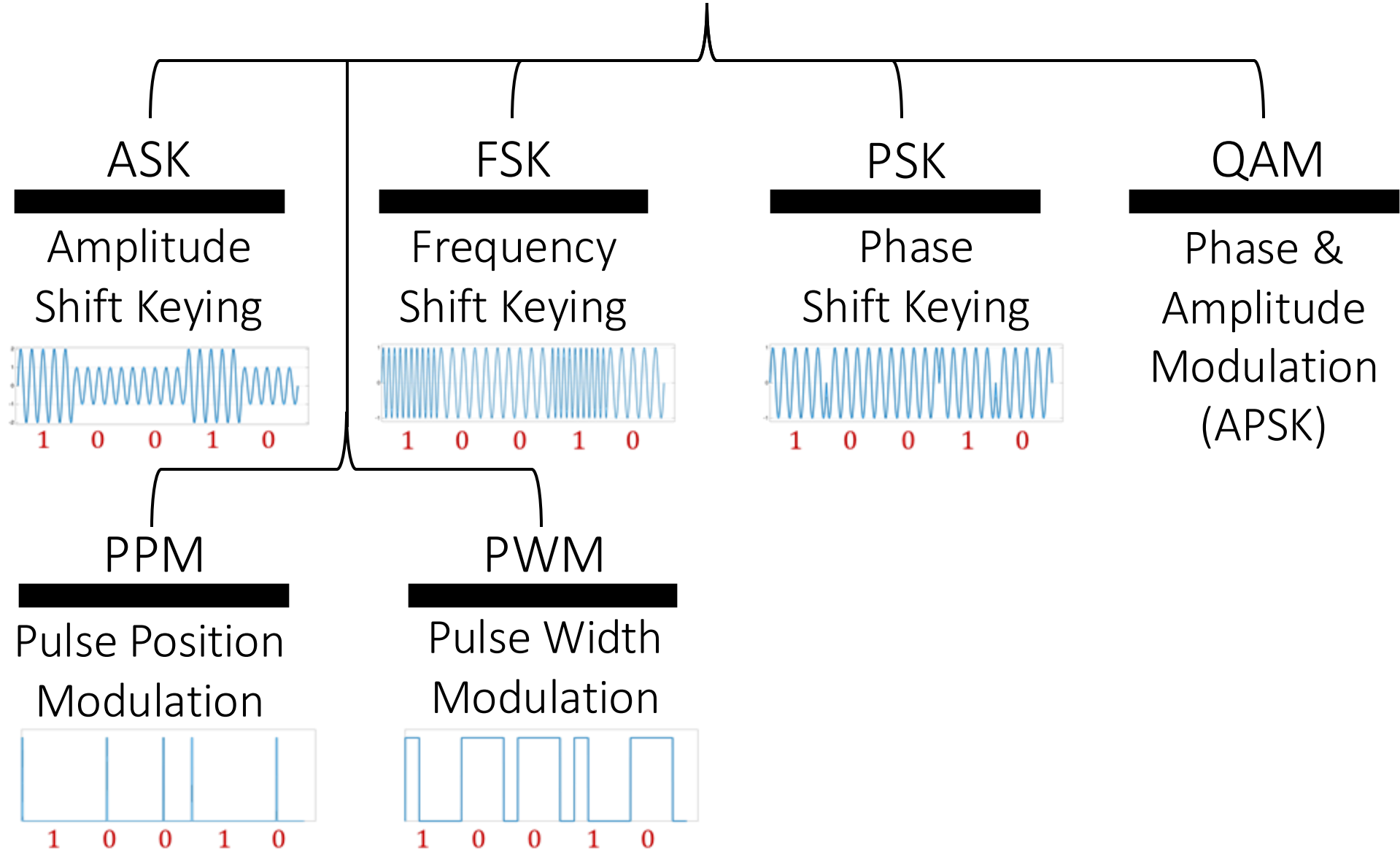
# Modulation

Based on how the bits are encoded



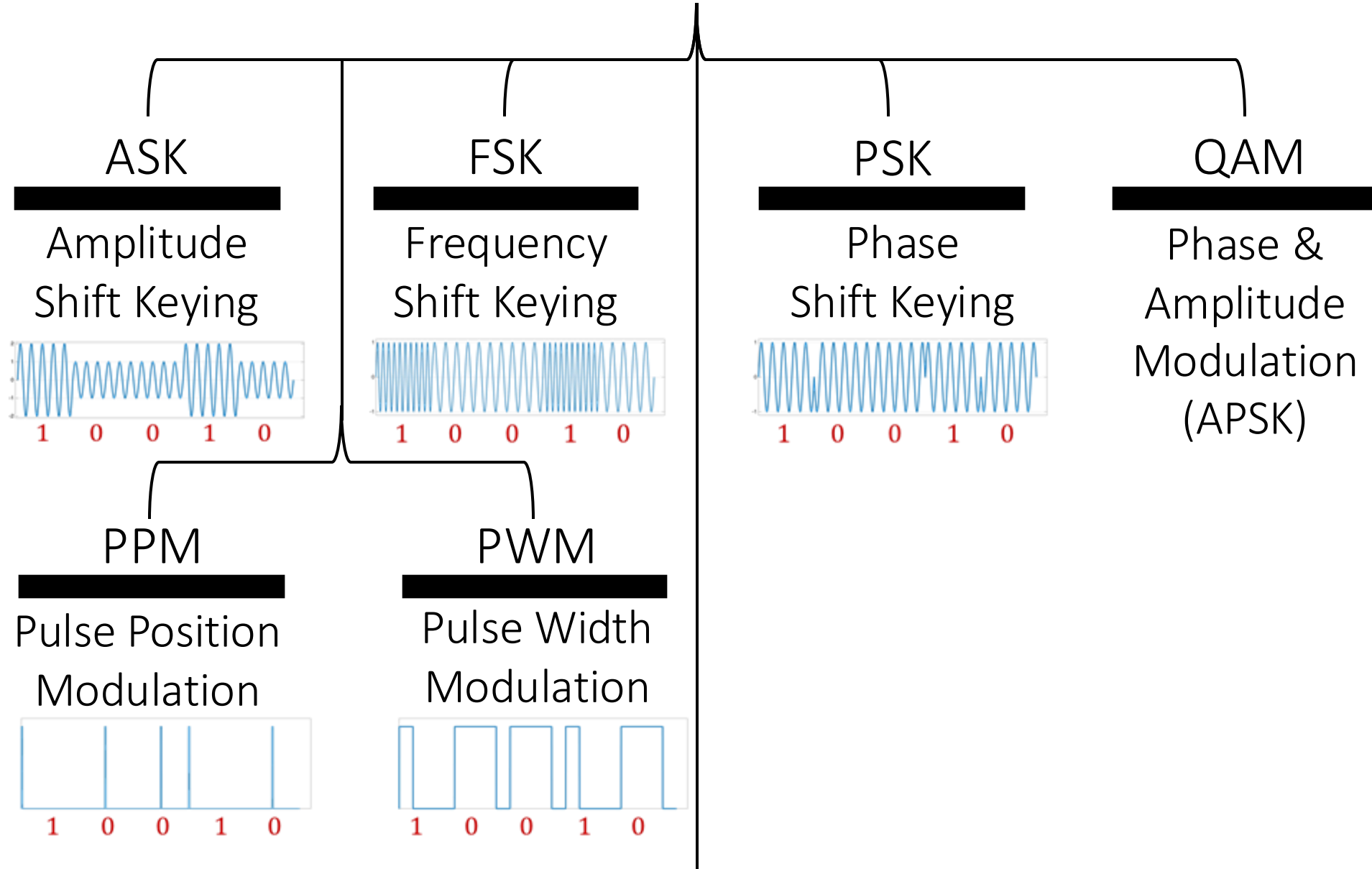
# Modulation

Based on how the bits are encoded



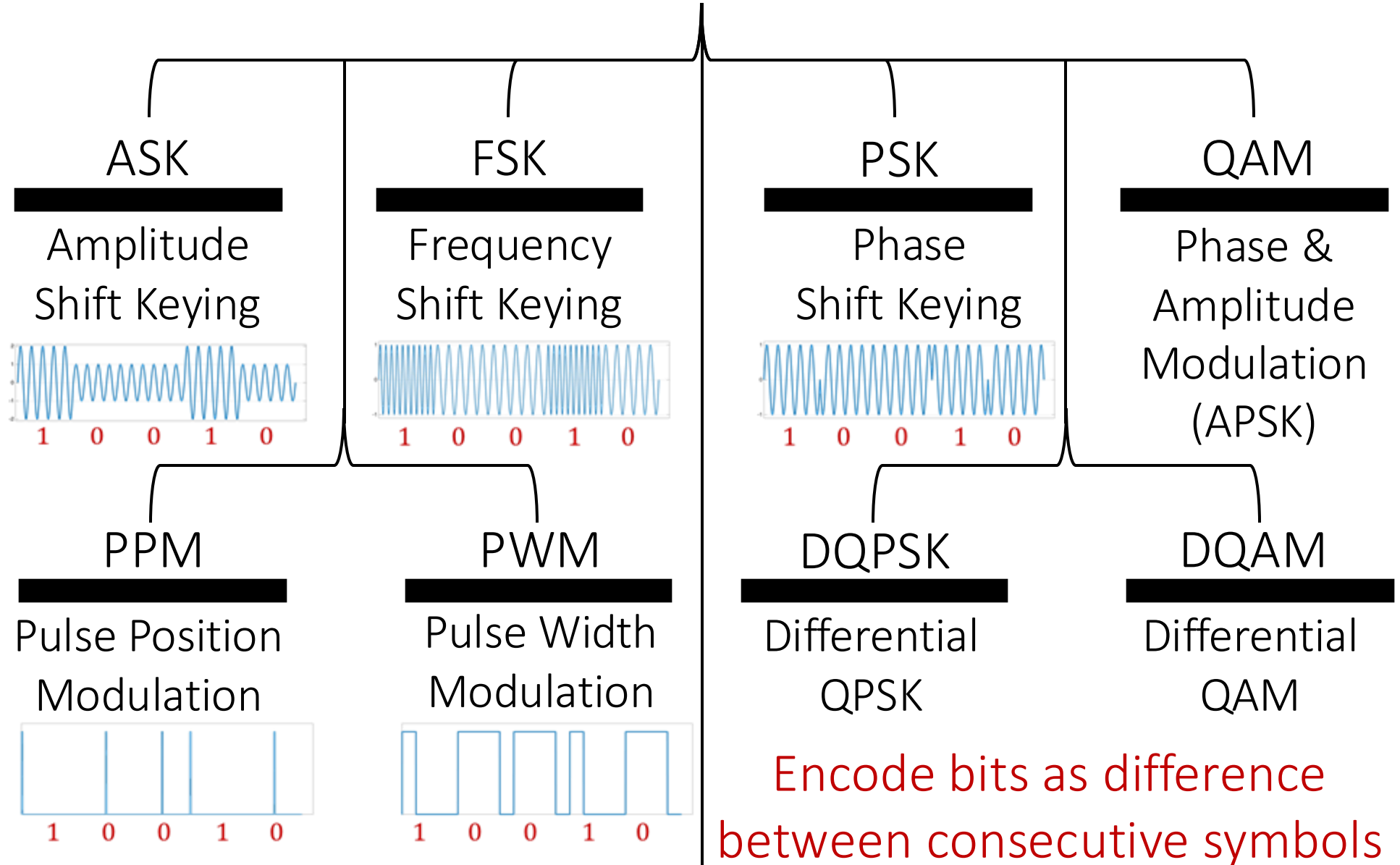
# Modulation

Based on how the bits are encoded



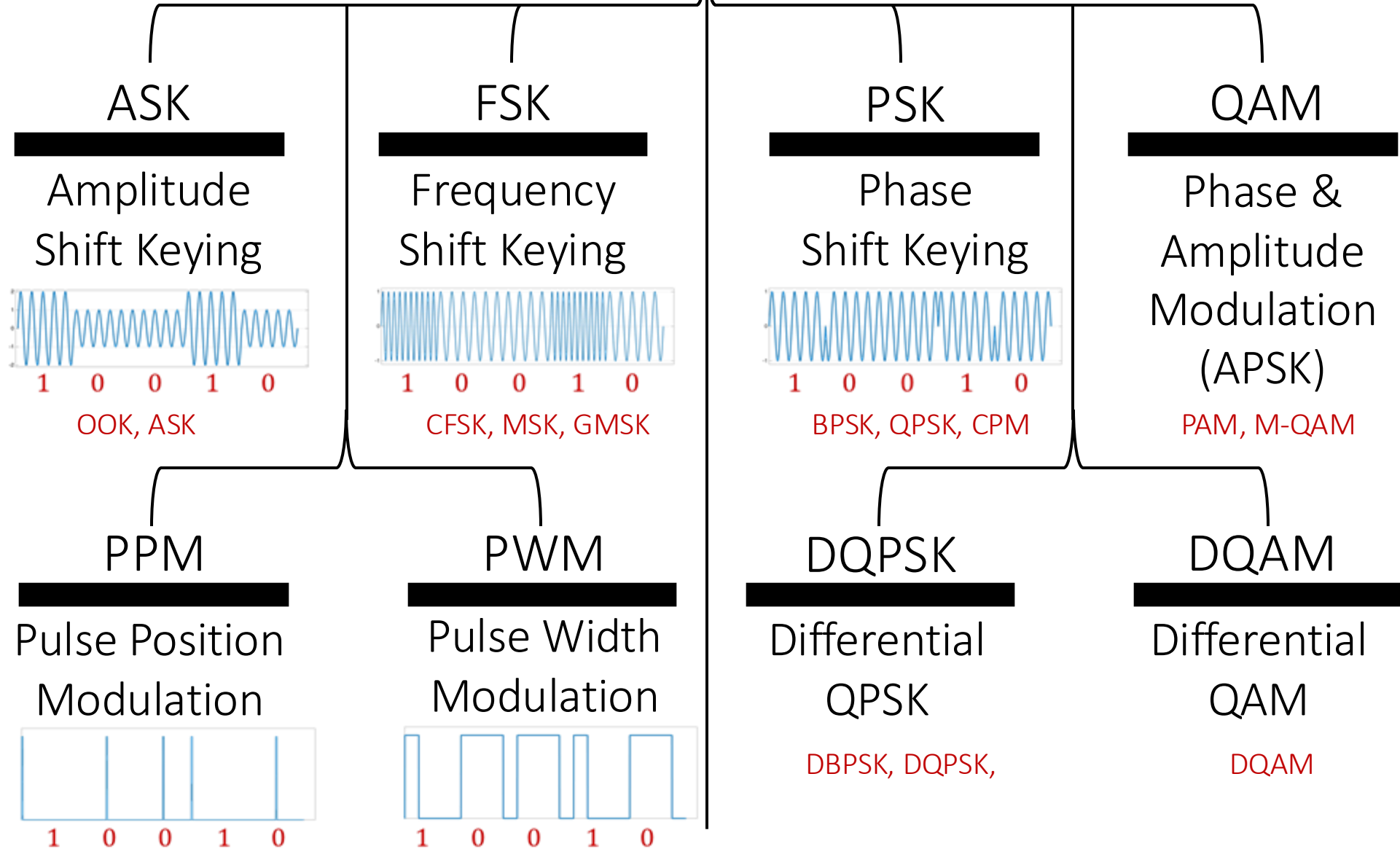
# Modulation

Based on how the bits are encoded



# Modulation

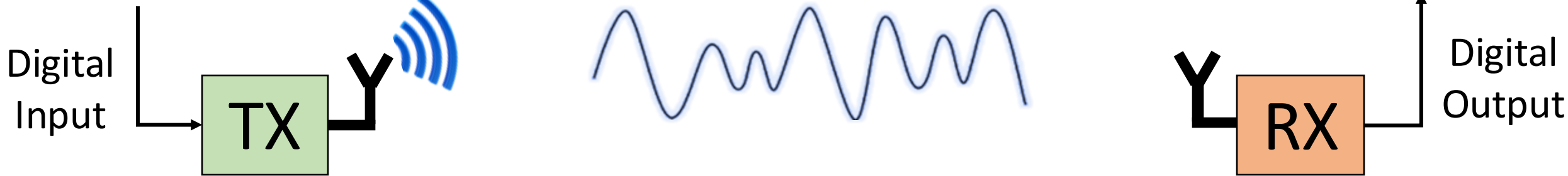
Based on how the bits are encoded



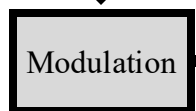
# Wireless Communication

1011010110011001

1011010110011001



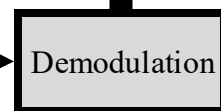
1011010110



$x(t)$



$y(t)$



1011010110

1011010110  $\rightarrow$   $s[n]$   $\rightarrow$   $s(t)$   $\rightarrow$   $s(t) \cos 2\pi f_c t$   $\rightarrow$   $h(t) * s(t) \cos 2\pi f_c t \times \cos 2\pi f_c t$   $\rightarrow$   $h(t) * s(t)$   $\rightarrow$   $h[n] * s[n]$

$s[n]$

1011010110

# Wireless Communication

1011010110011001

1011010110011001

Digital Input

TX

RX

Digital Output

1011010110

Modulation

$x(t)$

Channel

$y(t)$

Demodulation

1011010110

1011010110  $\rightarrow$   $s[n]$   $\rightarrow$   $s(t)$   $\rightarrow$   $s(t)e^{-j2\pi fct}$   $\rightarrow$   $h(t) * s(t)e^{-j2\pi fct} \times e^{+j2\pi fct}$   $\rightarrow$   $h(t) * s(t)$   $\rightarrow$   $h[n] * s[n]$

$s[n]$

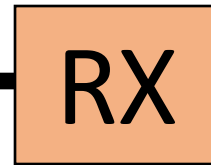
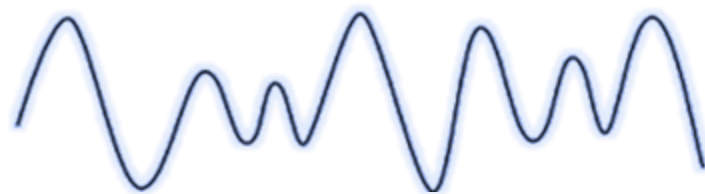
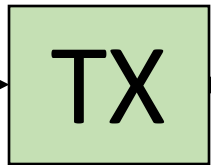
1011010110

# Wireless Communication

1011010110011001

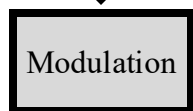
1011010110011001

Digital Input



Digital Output

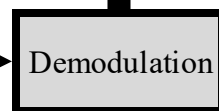
1011010110



$x(t)$



$y(t)$



1011010110

1011010110

$\rightarrow s[n] \rightarrow$

$s[n]e^{-j2\pi fct}$

$\rightarrow$

$h[n] * s[n]e^{-j2\pi fct} \times e^{+j2\pi fct}$

$\rightarrow h[n] * s[n]$

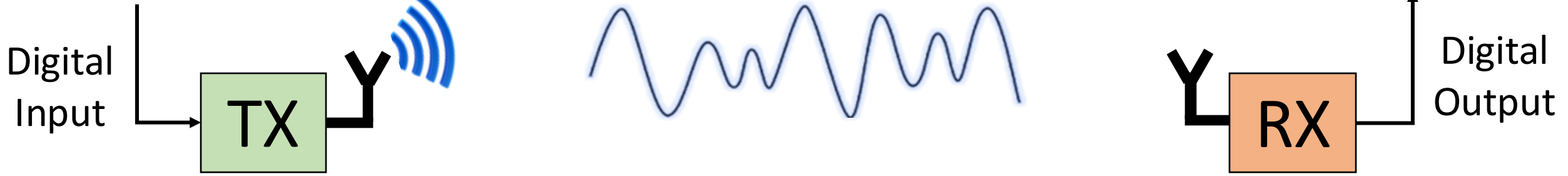
$s[n]$

1011010110

# Wireless Communication

1011010110011001

1011010110011001



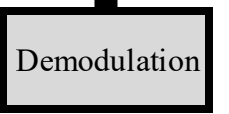
1011010110



$x(t)$



$y(t)$



1011010110

$$1011010110 \rightarrow s[n] \rightarrow s[n]e^{-j2\pi f_c t} \rightarrow h[n] * s[n]e^{-j2\pi f_c t} \times e^{+j2\pi f_c t} \rightarrow h[n] * s[n] \rightarrow s[n] \rightarrow 1011010110$$

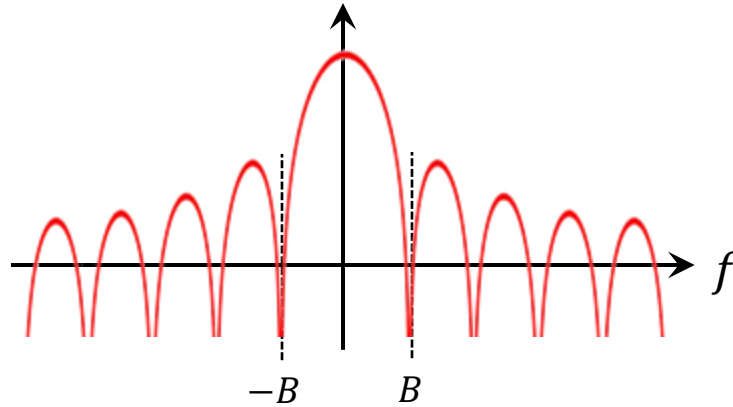
## Single Carrier Modulation

Symbols modulated on a single carrier frequency:  $s[n]e^{-j2\pi f_c t}$

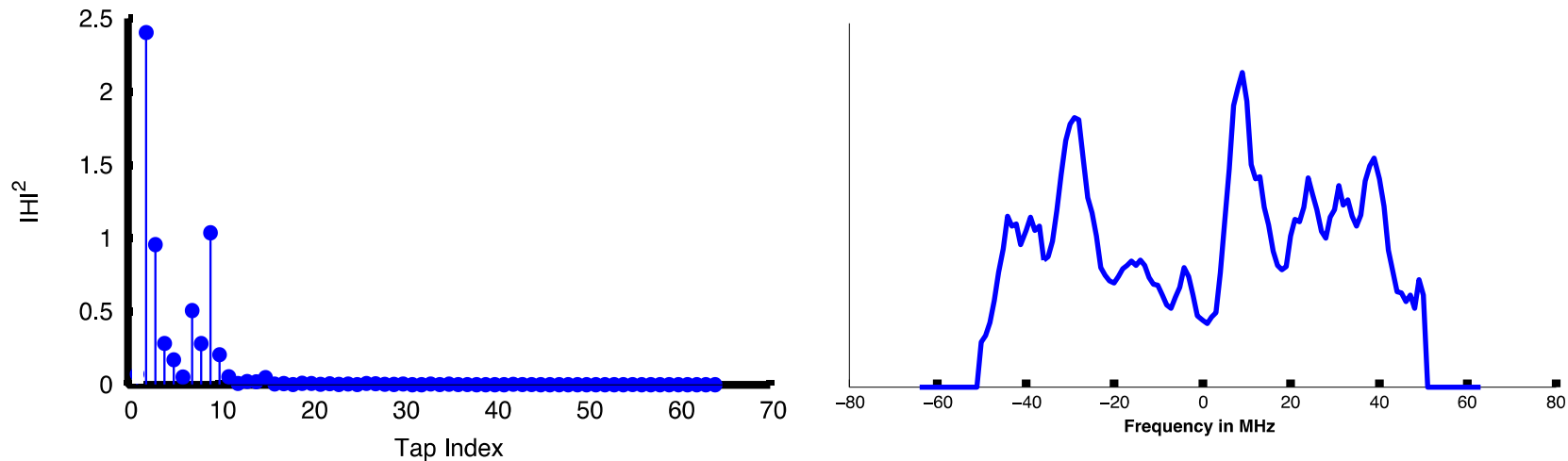
# Single Carrier Modulation

Symbols modulated on a single carrier frequency

- Low Spectral Efficiency: leakage



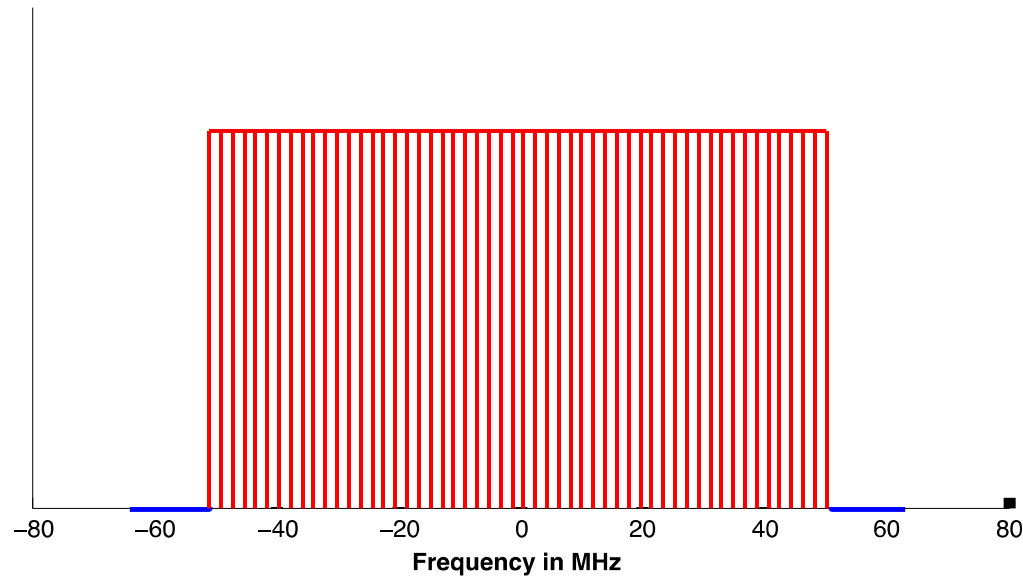
- ISI: Inter-Symbol-Interference limits performance



# Multi-Carrier Modulation

Symbols modulated on multiple Sub-carrier frequencies

- Divide spectrum into many narrow bands



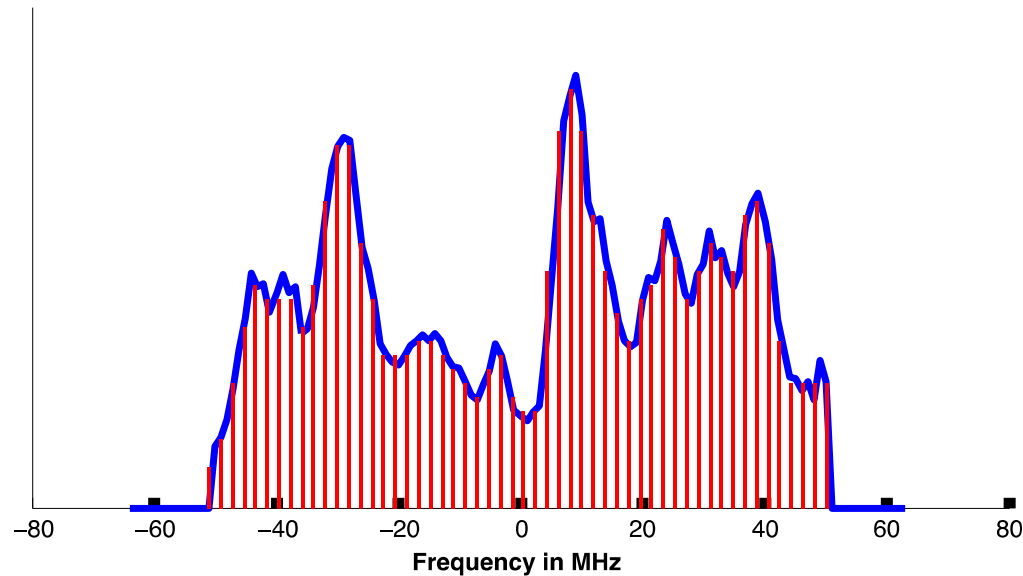
$$x(t) = \sum_i s_i[n] e^{-j2\pi f_i t}$$

- Transmit symbols on different carriers in narrow bands
- Channel is Flat → No need to worry about ISI

# Multi-Carrier Modulation

Symbols modulated on multiple Sub-carrier frequencies

- Divide spectrum into many narrow bands



$$x(t) = \sum_i s_i[n] e^{-j2\pi f_i t}$$

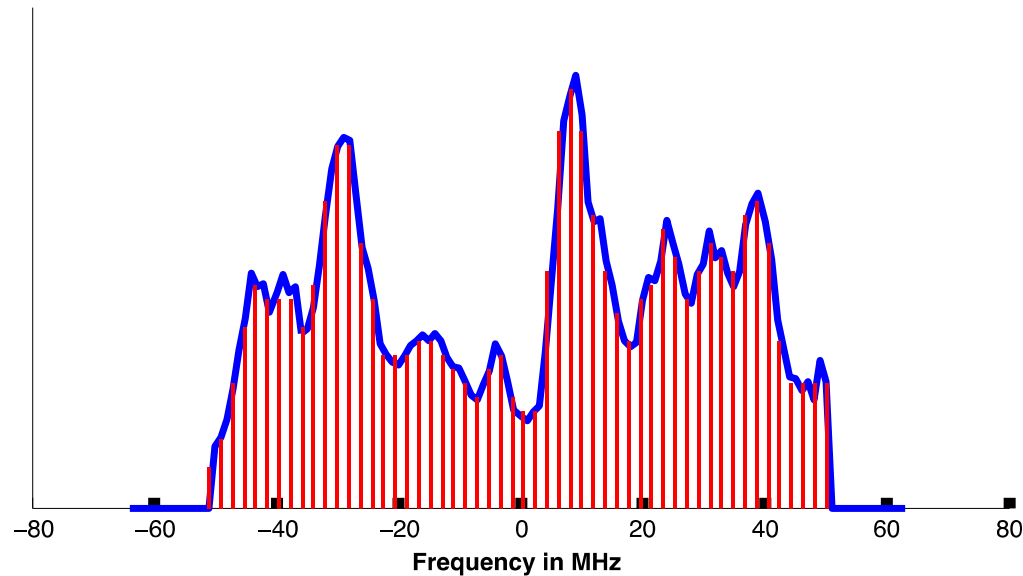
$$y(t) = \sum_i h_i s_i[n] e^{-j2\pi f_i t}$$

- Transmit symbols on different carriers in narrow bands
- Channel is Flat → No need to worry about ISI

# Multi-Carrier Modulation

Symbols modulated on multiple Sub-carrier frequencies

- Divide spectrum into many narrow bands



$$x(t) = \sum_i s_i[n] e^{-j2\pi f_i t}$$

$$y(t) = \sum_i h_i s_i[n] e^{-j2\pi f_i t}$$

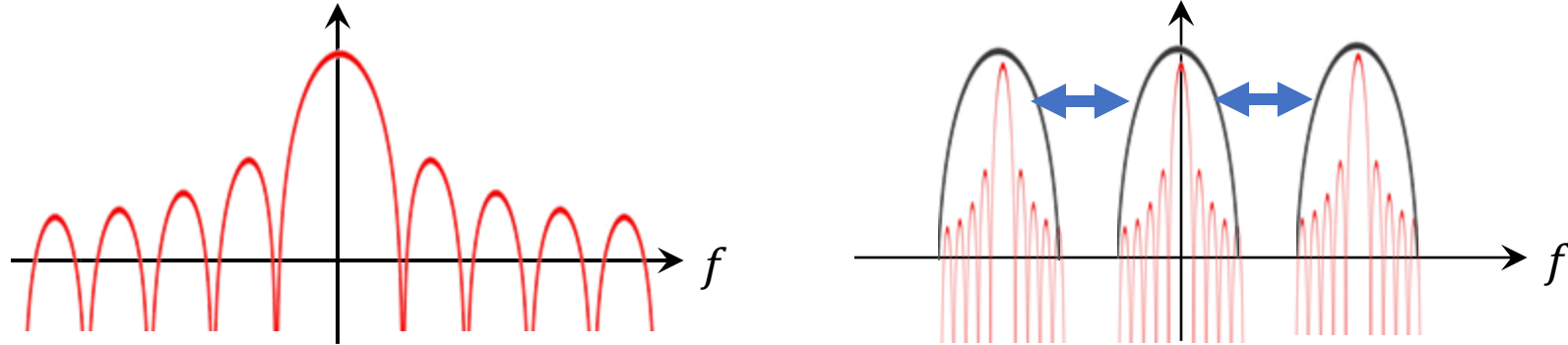
- Transmit symbols on different carriers in narrow bands
- Channel is Flat → No need to worry about ISI

**Not That Simple!**

# Multi-Carrier Modulation

Symbols modulated on multiple Sub-carrier frequencies

- Divide spectrum into many narrow bands



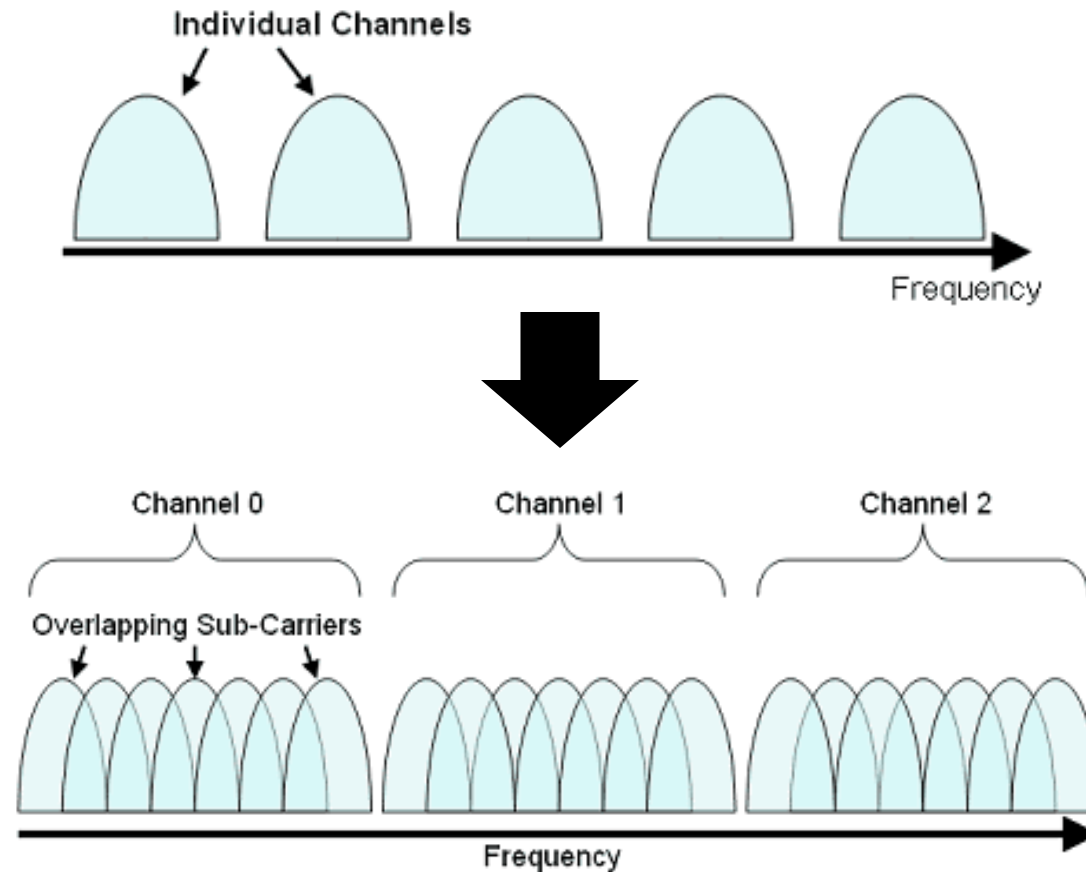
- Significant Leakage between adjacent subcarriers
- Need Guard Bands → Very inefficient!

**Solution: Make the Sub-Carriers Orthogonal**

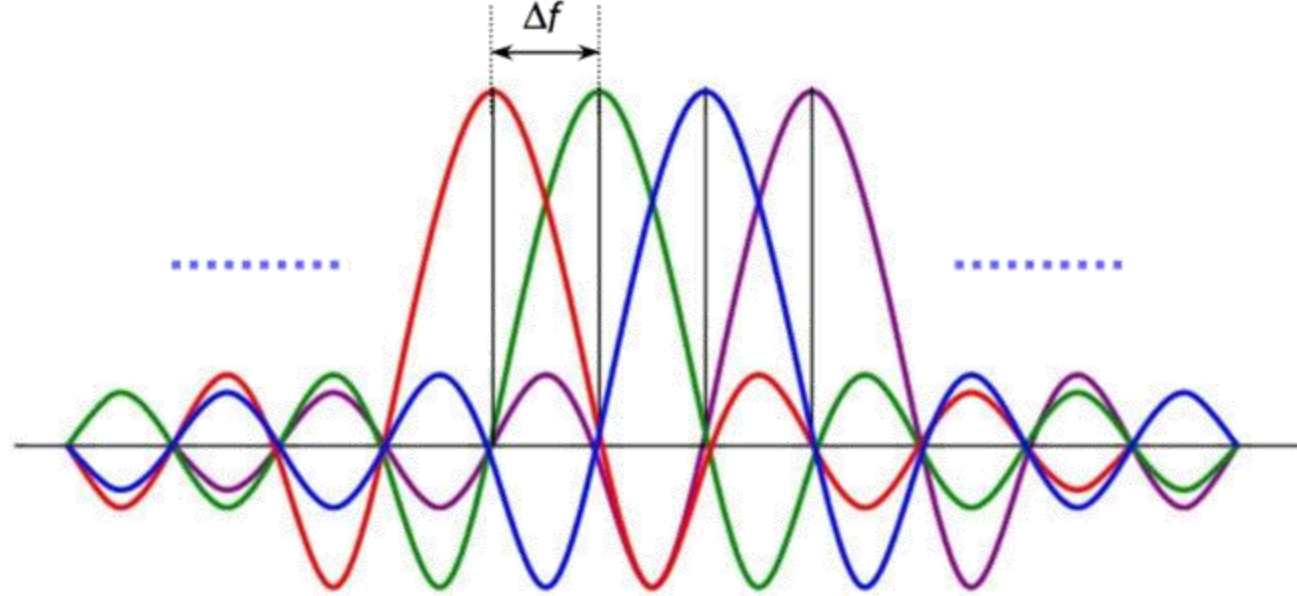
# Multi-Carrier Modulation

Symbols modulated on multiple Sub-carrier frequencies

Make the Sub-Carriers Orthogonal



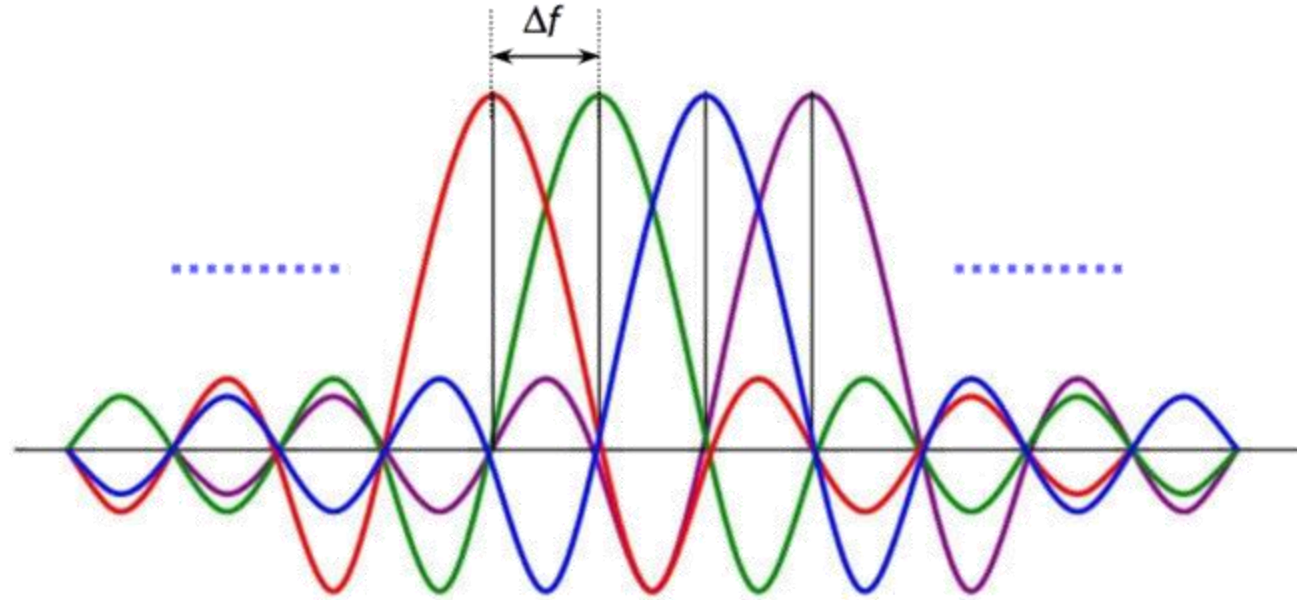
# OFDM: Orthogonal Frequency Division Multiplexing



- Subcarriers are orthogonal: At the sub-carrier frequency, the sampled value has zero leakage from other subcarriers.
- Subcarrier separation can be very small, for  $N$  subcarriers and bandwidth  $B$ :

$$\Delta f = \frac{B}{N}$$

# OFDM: Orthogonal Frequency Division Multiplexing



- Subcarriers are orthogonal: At the sub-carrier frequency, the sampled value has zero leakage from other subcarriers.
- Subcarrier separation can be very small, for  $N$  subcarriers and bandwidth  $B$ :

$$\Delta f = \frac{B}{N}$$

## How to Achieve This?

# OFDM: Orthogonal Frequency Division Multiplexing

## Use DFT: Discrete Fourier Transform

$$\text{N-Point DFT: } X(f_i) = \frac{1}{N} \sum_{t=0}^{N-1} x(t) e^{-j \frac{2\pi f_i t}{N}}$$

$$\text{N-Point IDFT: } x(t) = \sum_{f_i=0}^{N-1} X(f_i) e^{j \frac{2\pi f_i t}{N}}$$

## Send symbols in Frequency Domain

$X(f_i) = s[n] \rightarrow$  Compute and transmit  $x(t)$  using IDFT

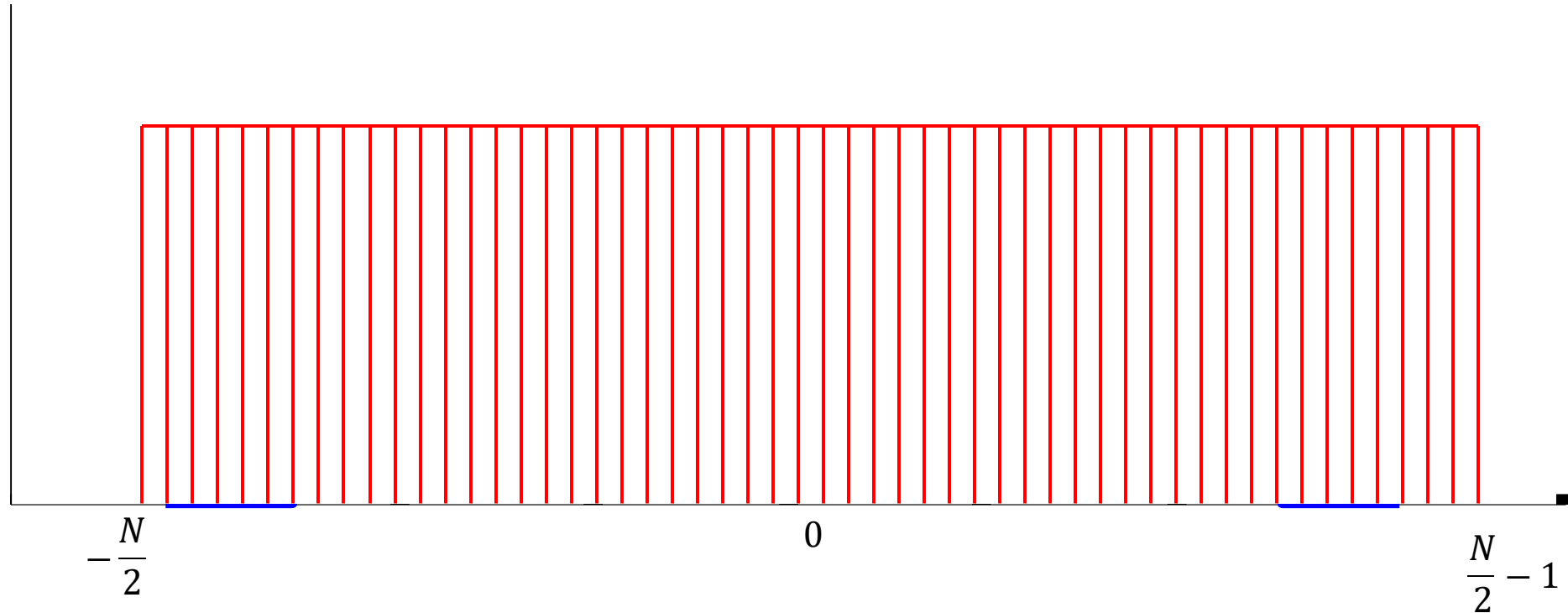
# OFDM: Orthogonal Frequency Division Multiplexing

## Send symbols in Frequency Domain

$X(f_i) = s[n] \rightarrow$  Compute and transmit  $x(t)$  using IDFT

- $N$  subcarrier  $\rightarrow$  IDFT of length  $N$
- Symbols  $s[n]$  can come from any modulation: BPSK, QPSK, QAM...
- $x(t)$  is complex  $\rightarrow$  need  $I$  &  $Q$   $\rightarrow$  No point using PAM or ASK ...
- OFDM Symbol:  $N$  samples of  $x(t)$  generated from the same modulated symbols using IDFT.
- OFDM Symbol Time:  $T = N/B$  where  $B$  is the bandwidth.
- OFDM Frequency Bin Width:  $\Delta f = 1/T = B/N$

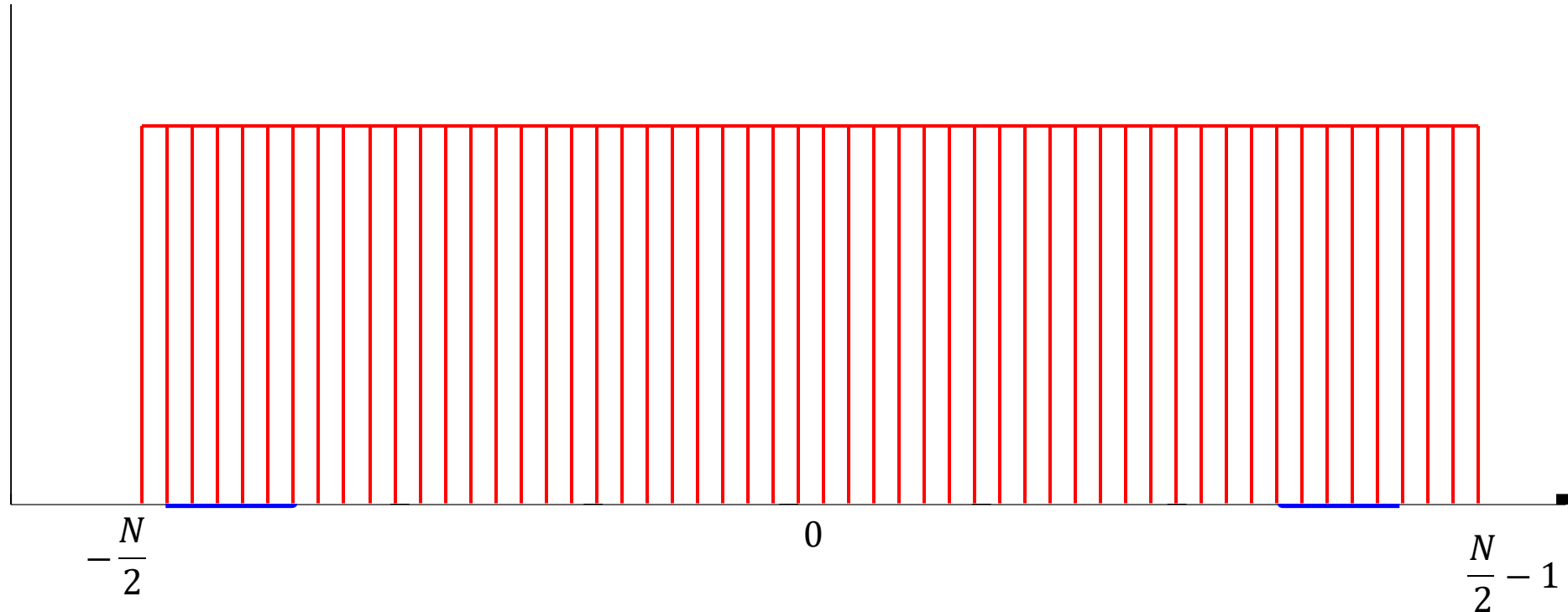
# OFDM Symbol in Frequency Domain



- FFT can be represented 0 to  $N - 1$  or  $N/2$  to  $N/2 - 1$ .
- OFDM Symbol created in digital baseband  $\rightarrow$  0 bin corresponds to DC

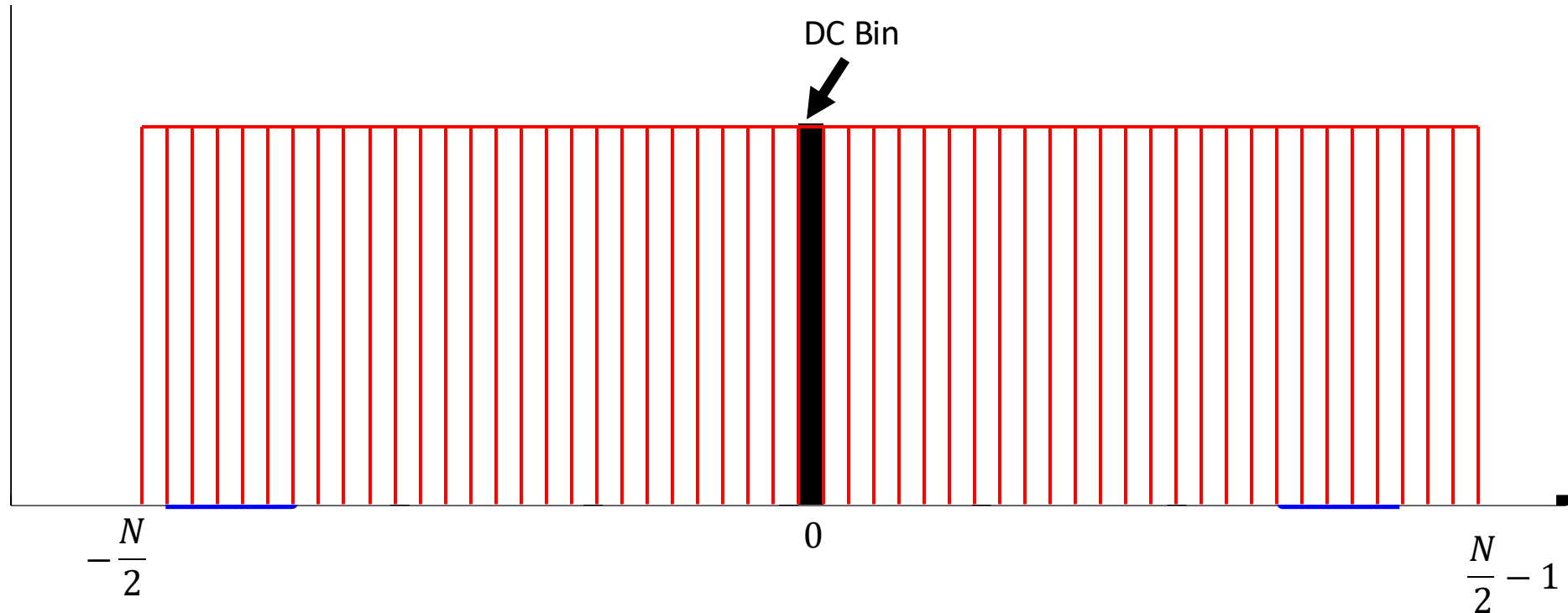
$$X(0) = \frac{1}{N} \sum_{t=0}^{N-1} x(t) e^{-j \frac{2\pi 0 t}{N}} = \frac{1}{N} \sum_{t=0}^{N-1} x(t) = DC$$

# OFDM Symbol in Frequency Domain



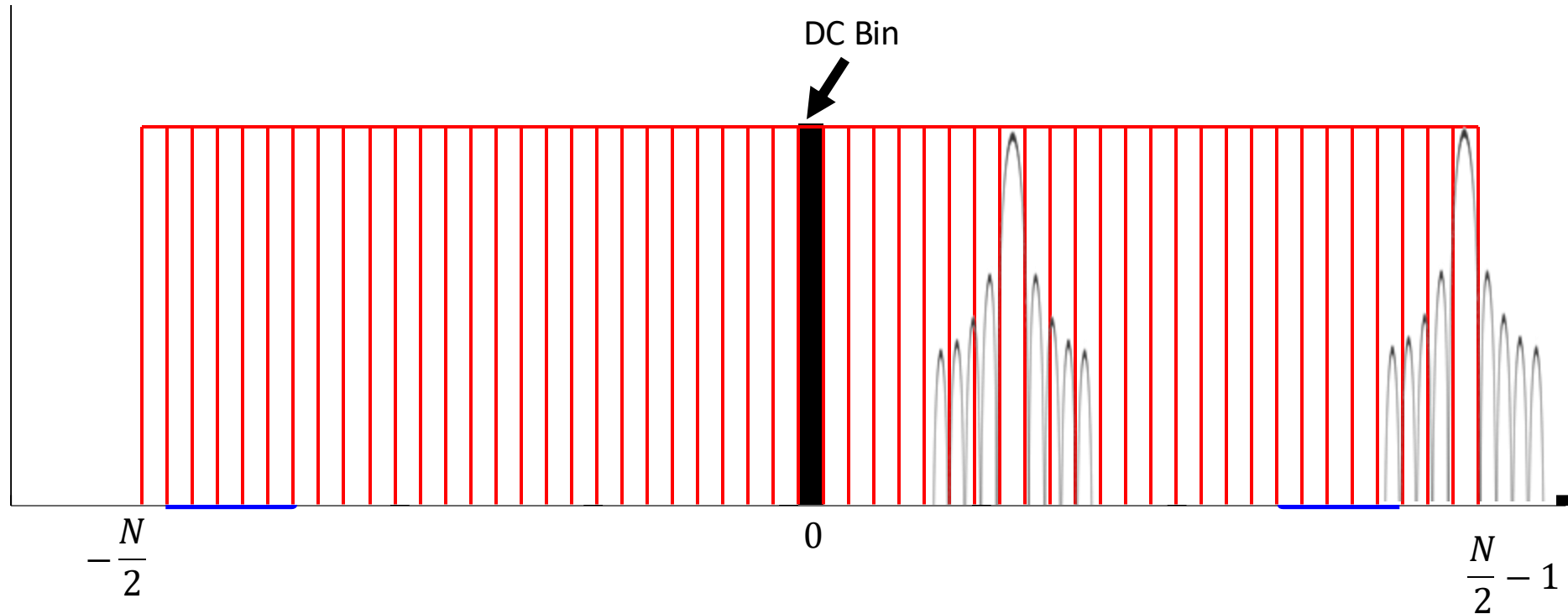
- FFT can be represented  $0$  to  $N - 1$  or  $N/2$  to  $N/2 - 1$ .
- OFDM Symbol created in digital baseband  $\rightarrow$   $0$  bin corresponds to DC
- DC of the circuits corrupts bits sent on the  $0$  bin  $\rightarrow$  Do not use  $0$  bin

# OFDM Symbol in Frequency Domain



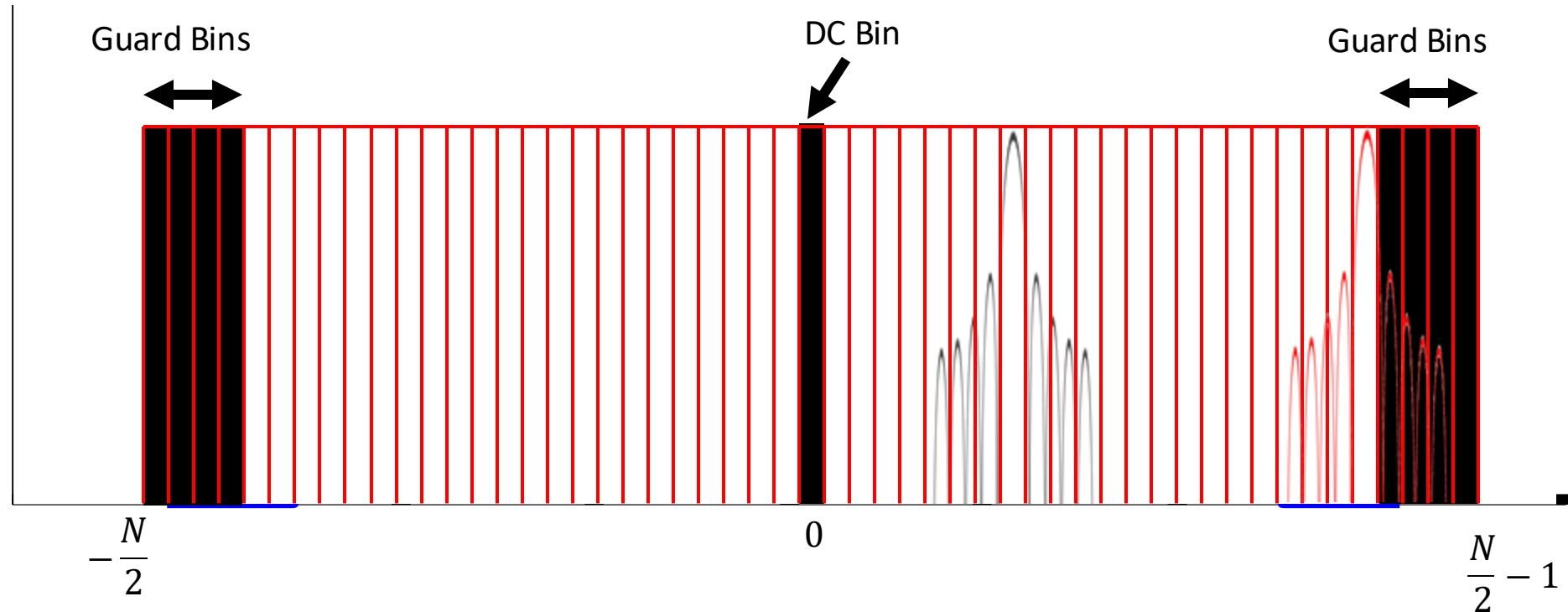
- FFT can be represented  $0$  to  $N - 1$  or  $N/2$  to  $N/2 - 1$ .
- OFDM Symbol created in digital baseband  $\rightarrow$   $0$  bin corresponds to DC
- DC of the circuits corrupts bits sent on the  $0$  bin  $\rightarrow$  Do not use  $0$  bin

# OFDM Symbol in Frequency Domain



- Subcarriers orthogonal to each other but not to near by channels.
- Need Guard Bins at sides of the channel → Transmit nothing there

# OFDM Symbol in Frequency Domain



- Subcarriers orthogonal to each other but not to near by channels.
- Need Guard Bins at sides of the channel → Transmit nothing there
- Reduce Number of Guard band from  $N$  to 2 → Very Spectrally Efficient

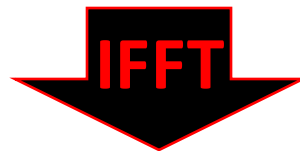
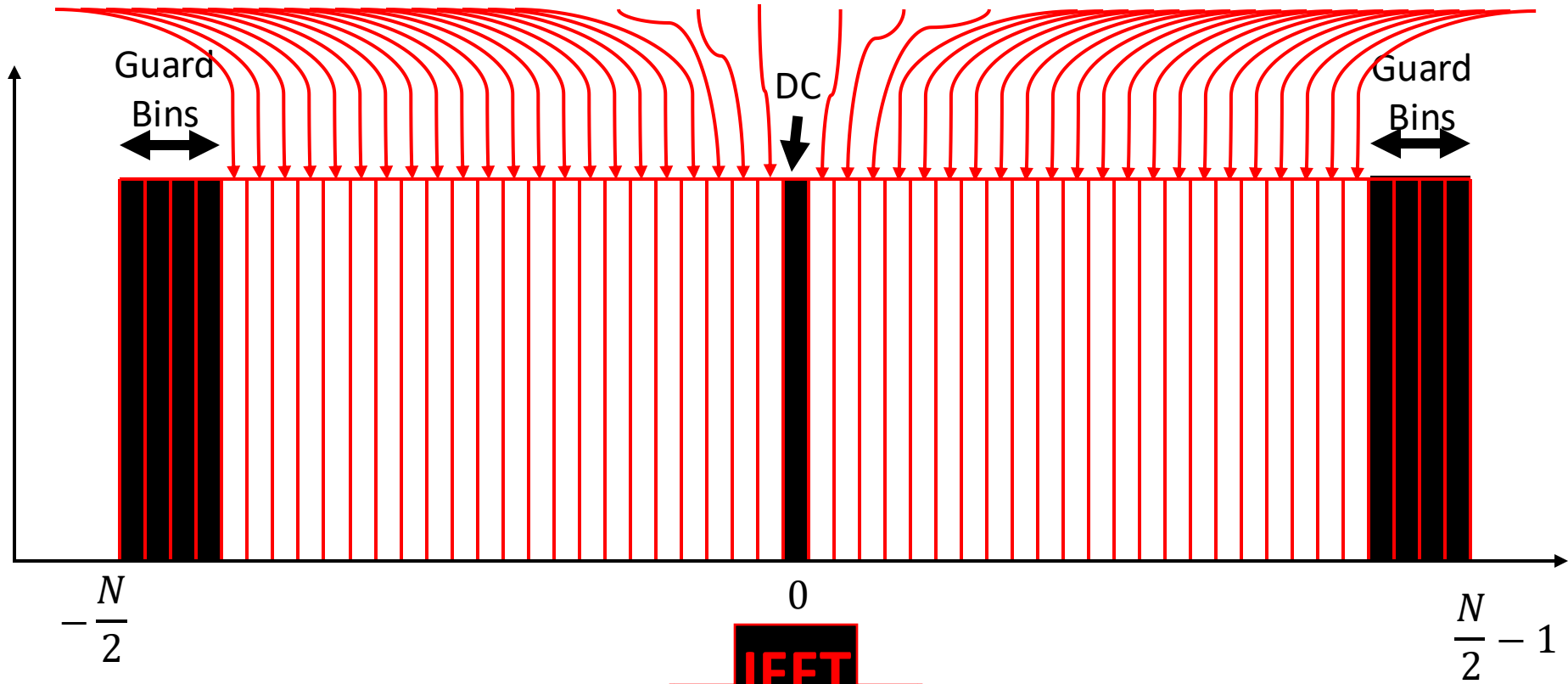
# OFDM Symbol

Bits:

1 0 1 0 1 0 0 0 1 1 0 1 1 0 0

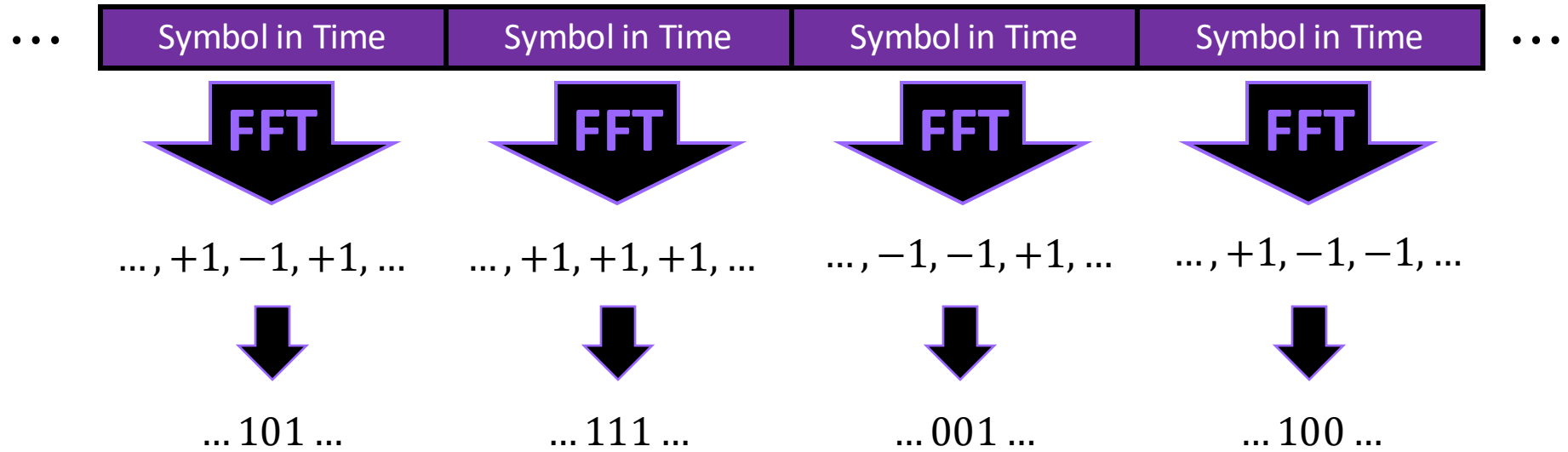


..., +1, -1, +1, -1, +1, -1, -1, -1, +1, +1, -1, +1, +1, -1, -1, ...



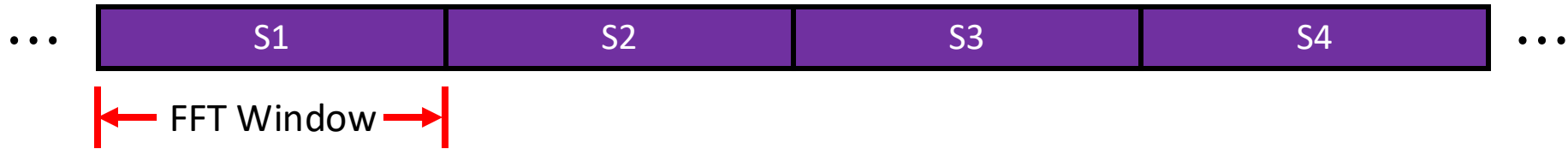
Symbol in Time

# OFDM Symbol



**Not That Simple**

# OFDM Symbol

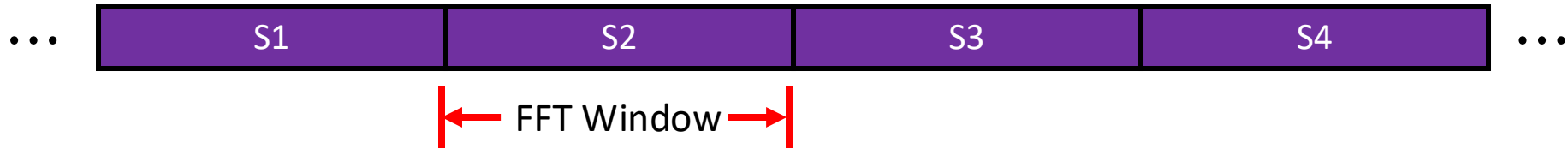


..., +1, -1, +1, ...



... 101 ...

# OFDM Symbol



..., +1, -1, +1, ...



... 101 ...

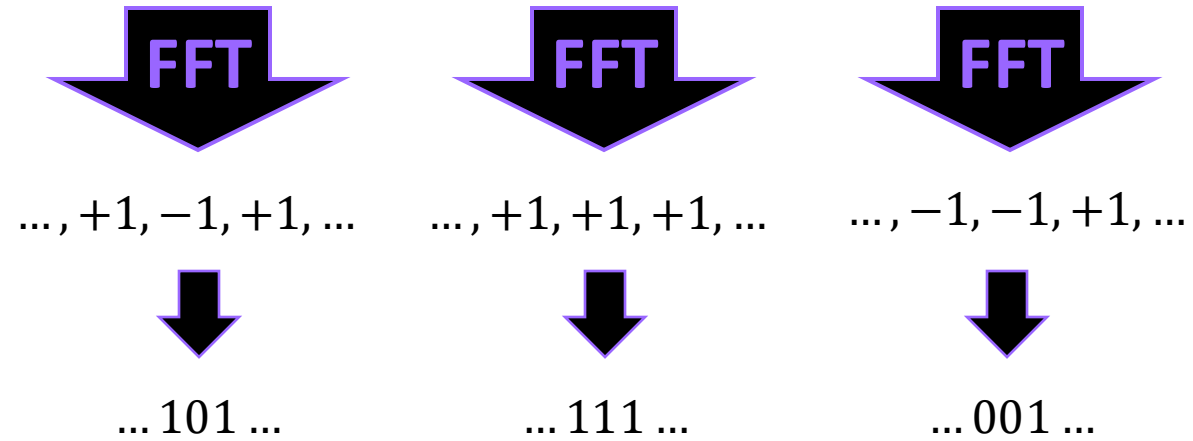
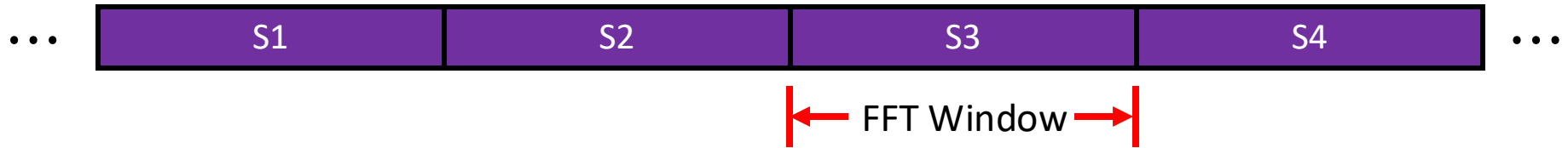


..., +1, +1, +1, ...

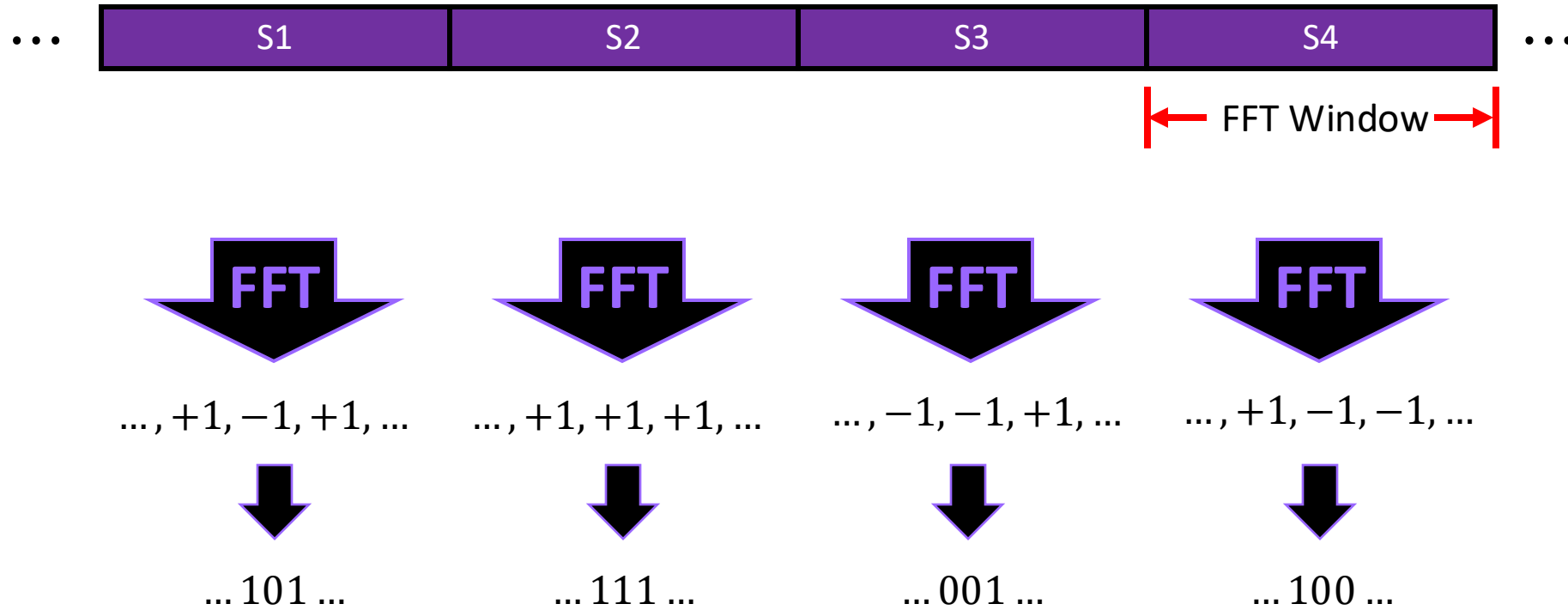


... 111 ...

# OFDM Symbol



# OFDM Symbol



Assumes FFT window is perfectly aligned with symbol boundaries

# OFDM Symbol



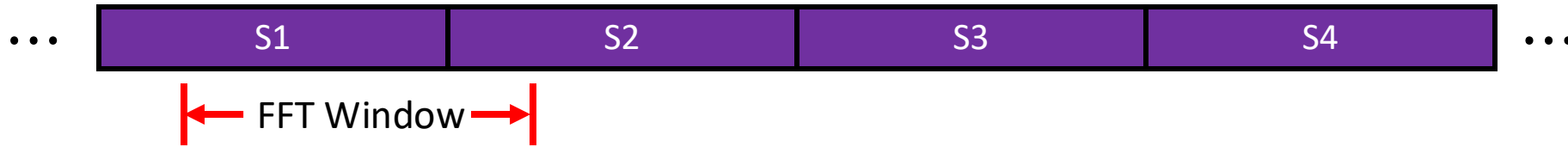
...,  $+0.5 + 1i$ ,  $-0.7 + 0.3i$ , ...

**✗ Cannot decode!**

FFT window is misaligned with symbol

Subcarriers are no longer orthogonal.

# OFDM Cyclic Prefix

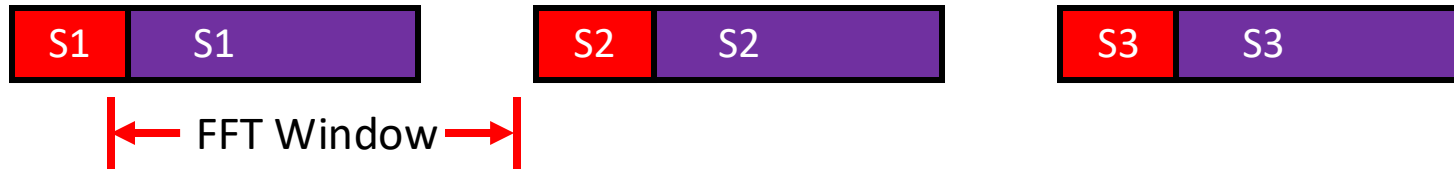


- DFT (FFT) assumes time samples are periodic of period  $N$
- Circular Shift before taking FFT:

$$x[t] \rightarrow X[f]$$

$$x[t - \tau \bmod N] \rightarrow X[f]e^{-j\frac{2\pi f\tau}{N}}$$

# OFDM Cyclic Prefix



- DFT (FFT) assumes time samples are periodic of period  $N$
- Circular Shift before taking FFT:

$$x[t] \rightarrow X[f]$$

$$x[t - \tau \bmod N] \rightarrow X[f]e^{-j\frac{2\pi f\tau}{N}}$$

# OFDM Cyclic Prefix

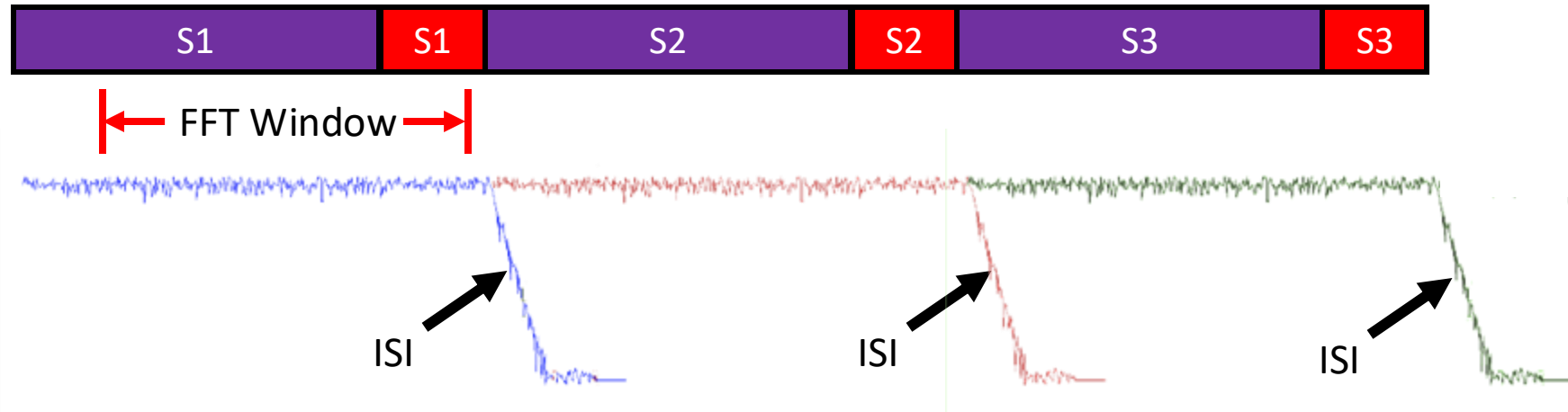


- Even if FFT window is misaligned, CP ensures that all samples come from the same symbol → Orthogonality is preserved!
- Cyclic Prefix can be created by:
  - Take first few samples and append them to end of symbol.
  - Take last few samples and prefix them to beginning of symbol.
- Simple Phase Shift → Can be corrected by lumping with channel  $H[f]$

# OFDM Cyclic Prefix

Cyclic Prefix:

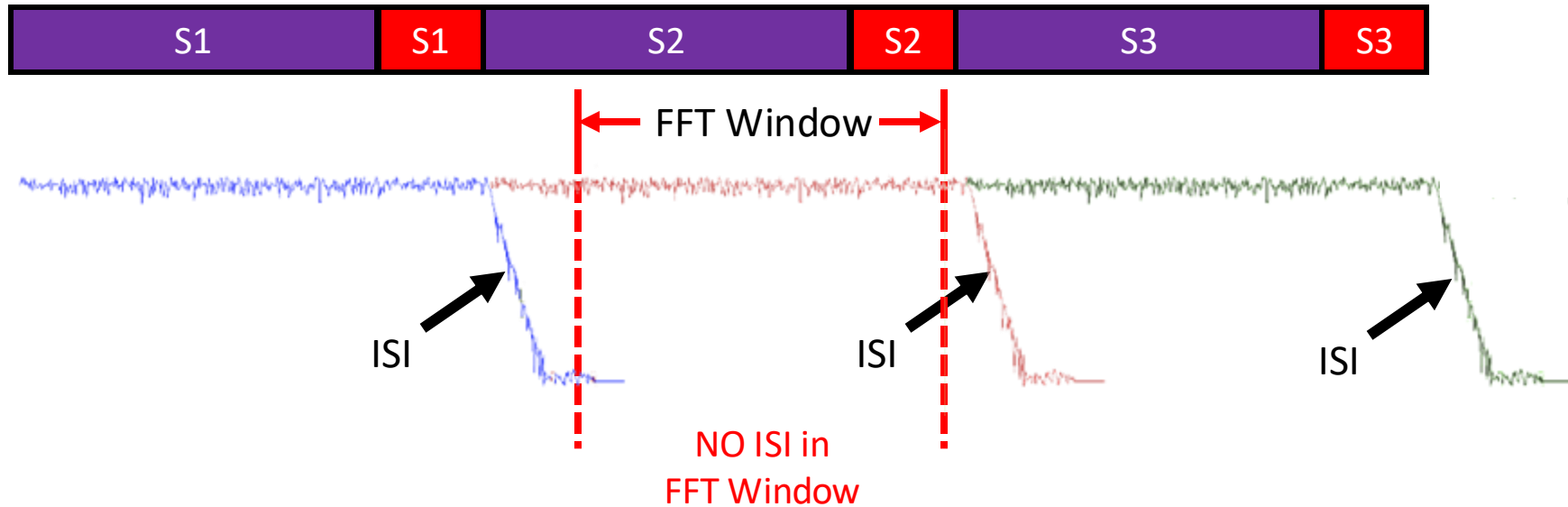
- Preserves orthogonality by allowing some misalignment in FFT Window
- Deals with Inter-Symbol-Interference



# OFDM Cyclic Prefix

Cyclic Prefix:

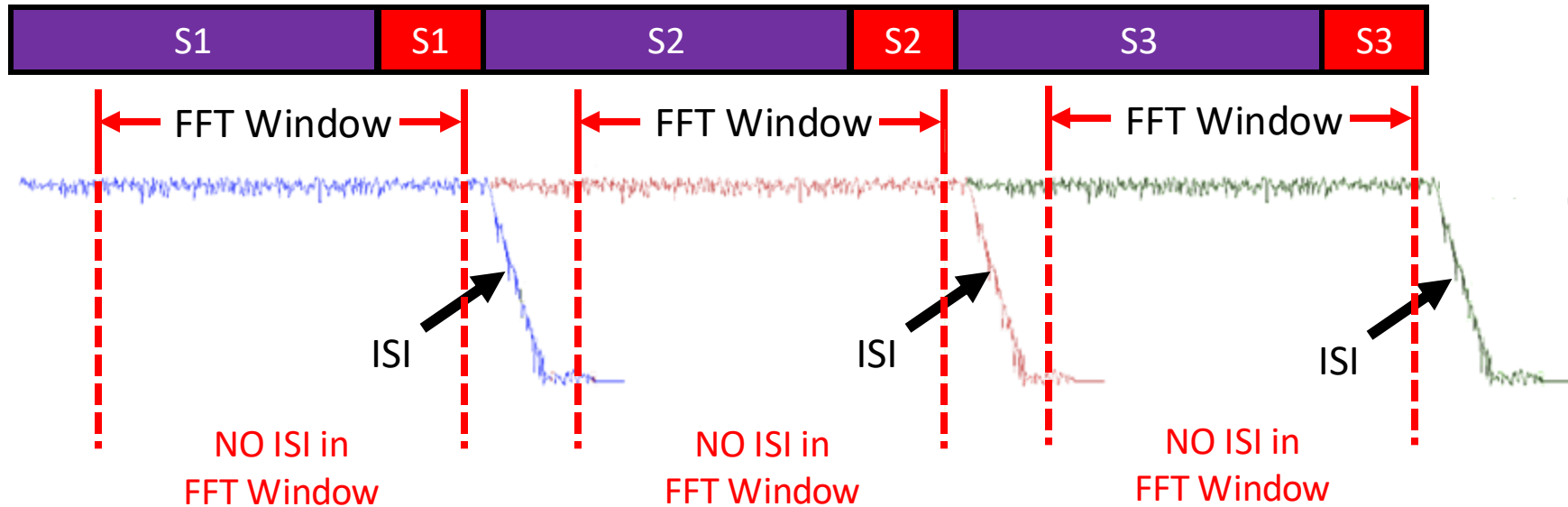
- Preserves orthogonality by allowing some misalignment in FFT Window
- Deals with Inter-Symbol-Interference



# OFDM Cyclic Prefix

Cyclic Prefix:

- Preserves orthogonality by allowing some misalignment in FFT Window
- Deals with Inter-Symbol-Interference



# OFDM Cyclic Prefix

Cyclic Prefix:

- + Preserves orthogonality by allowing some misalignment in FFT Window
- + Deals with Inter-Symbol-Interference
- Overhead: Send  $CP + N$  samples for every  $N$  samples

$$\text{Overhead} = \frac{CP}{CP + N}$$

e. g. WiFi 802.11n:  $N = 64$ ,  $CP = 16 \rightarrow \text{Overhead} = 20\%$

e. g. LTE:  $N = 1024$ ,  $CP = 72 \rightarrow \text{Overhead} = 6.5\%$

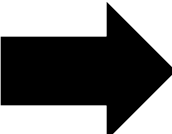
# OFDM Cyclic Prefix



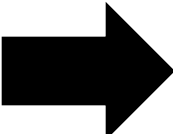
- Cyclic prefix is a not a bullet proof solution.
- Can still end up misaligned!
- Need a way to ensure we detect the beginning of the packet correctly.
- If we do, CP will ensure that even if we are not accurate, we can still decode.

# OFDM Transmitter

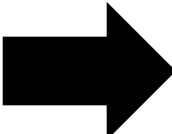
101101011  
Data Bits



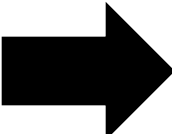
Modulation



IFFT



Add Cyclic  
Prefix



Transmit  
Signals



Repeat until all data bits are transmitted

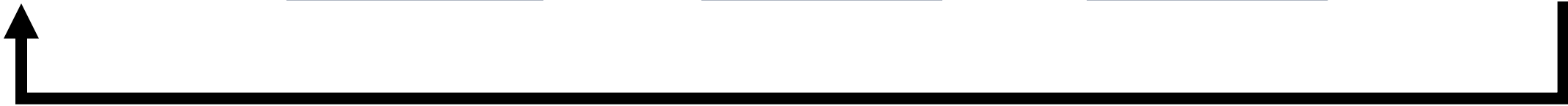
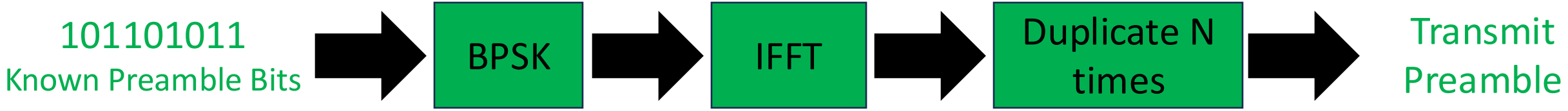
# OFDM Packet Detection

- Detect Beginning of packet to make sure we are within the CP
- Send Training Sequence: Preamble Symbols
- Preamble Symbols: Known Symbol Repeated at the beginning of packet



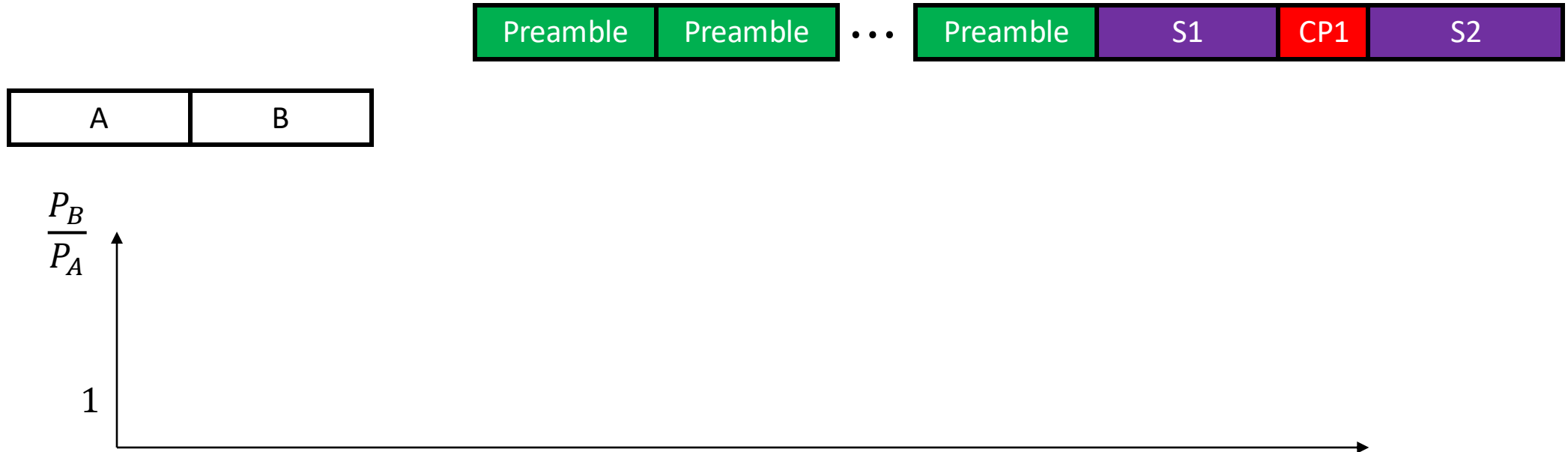
- No need for CP with preamble symbols

# OFDM Transmitter



Repeat until all data bits are transmitted

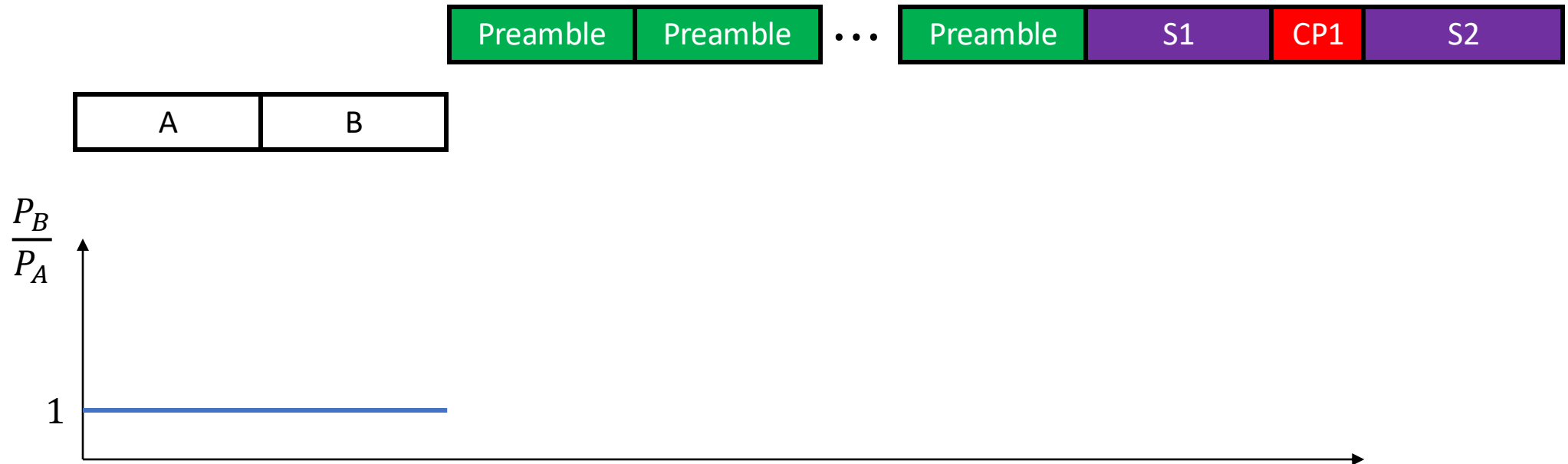
# OFDM Packet Detection: Sliding Window



- Two windows of  $L$  ( $2N$ ) samples each.

- Compute: 
$$\frac{P_B}{P_A} = \frac{\sum_{k=t+L}^{t+2L} |y[k]|^2}{\sum_{k=t}^{t+L} |y[k]|^2}$$

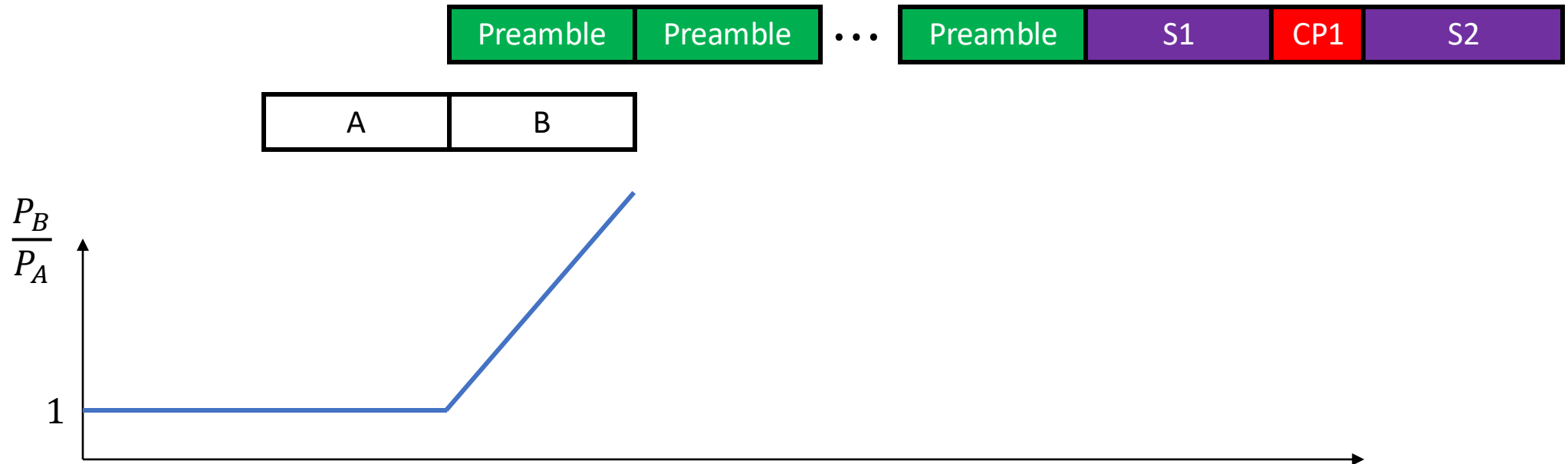
# OFDM Packet Detection: Sliding Window



- Two windows of  $L$  ( $2N$ ) samples each.

- Compute: 
$$\frac{P_B}{P_A} = \frac{\sum_{k=t+L}^{t+2L} |y[k]|^2}{\sum_{k=t}^{t+L} |y[k]|^2}$$

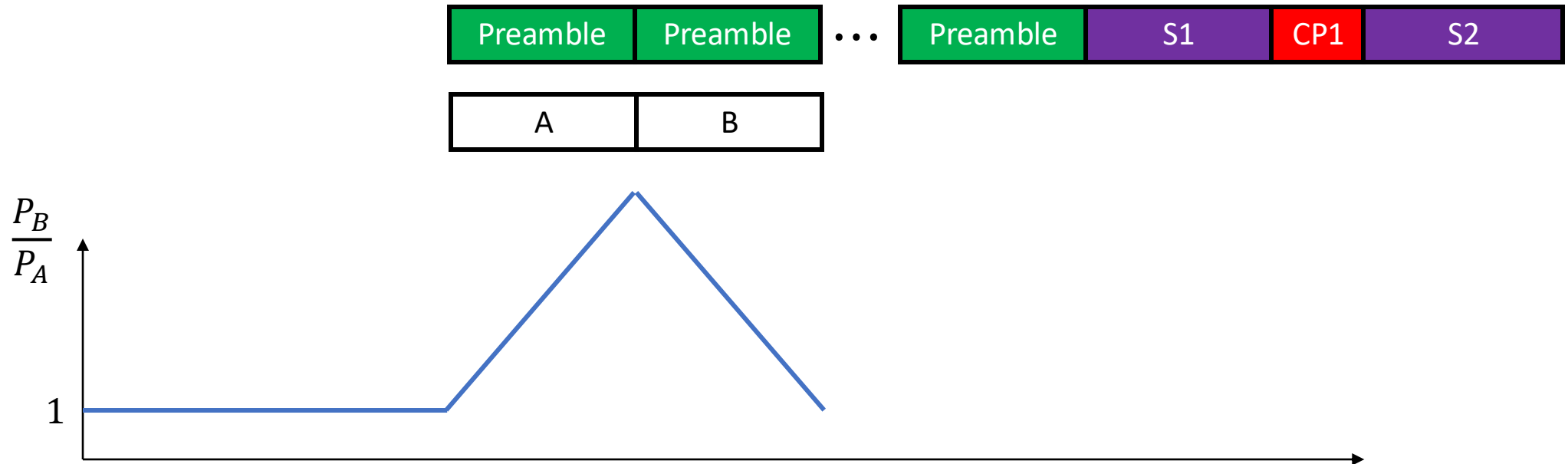
# OFDM Packet Detection: Sliding Window



- Two windows of  $L$  ( $2N$ ) samples each.

- Compute: 
$$\frac{P_B}{P_A} = \frac{\sum_{k=t+L}^{t+2L} |y[k]|^2}{\sum_{k=t}^{t+L} |y[k]|^2}$$

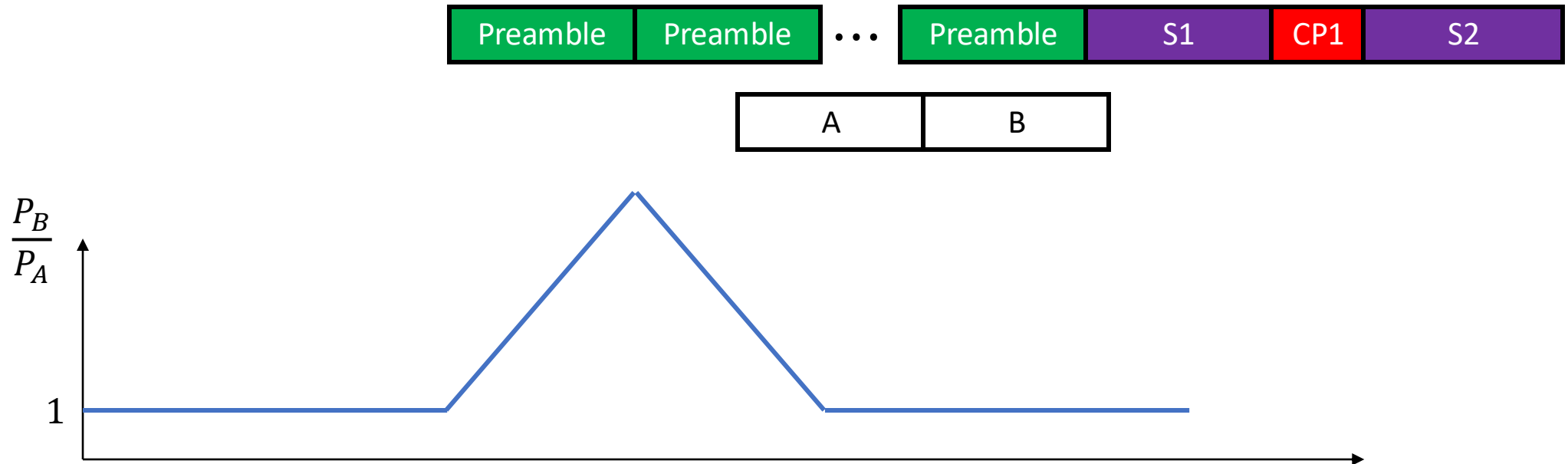
# OFDM Packet Detection: Sliding Window



- Two windows of  $L$  ( $2N$ ) samples each.

- Compute: 
$$\frac{P_B}{P_A} = \frac{\sum_{k=t+L}^{t+2L} |y[k]|^2}{\sum_{k=t}^{t+L} |y[k]|^2}$$

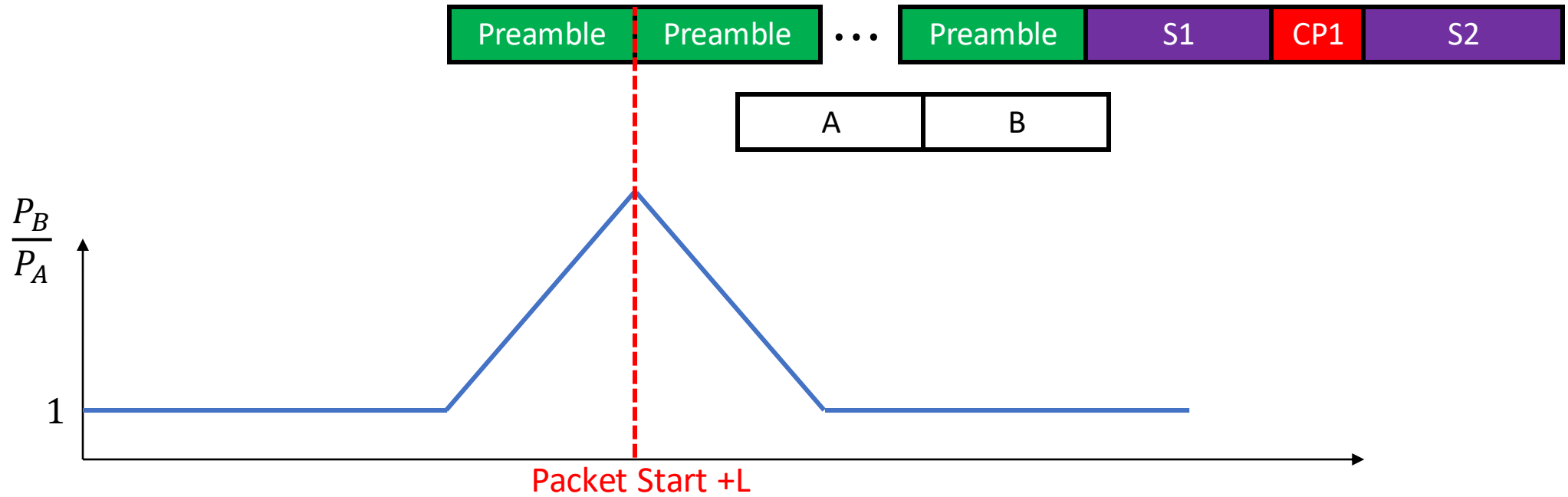
# OFDM Packet Detection: Sliding Window



- Two windows of  $L$  ( $2N$ ) samples each.

- Compute: 
$$\frac{P_B}{P_A} = \frac{\sum_{k=t+L}^{t+2L} |y[k]|^2}{\sum_{k=t}^{t+L} |y[k]|^2}$$

# OFDM Packet Detection: Sliding Window



- Two windows of  $L$  ( $2N$ ) samples each.

- Compute: 
$$\frac{P_B}{P_A} = \frac{\sum_{k=t+L}^{t+2L} |y[k]|^2}{\sum_{k=t}^{t+L} |y[k]|^2}$$

# Carrier Frequency Offset



$$\begin{aligned}
 x(t) &\longrightarrow x(t) \times e^{-j2\pi f_c t} \longrightarrow h(t) * x(t) e^{-j2\pi f_c t} \longrightarrow h(t) * x(t) e^{-j2\pi f_c t} \times e^{j2\pi f_c t} \\
 &\longrightarrow h(t) * x(t) \\
 &\longrightarrow y(t) = h(t) * x(t) + v(t)
 \end{aligned}$$

Assumes TX & RX perfectly synched

# Carrier Frequency Offset



$$x(t) \xrightarrow{\quad} x(t) \times e^{-j2\pi f_c t} \xrightarrow{\quad} h(t) * x(t) e^{-j2\pi f_c t} \xrightarrow{\quad} h(t) * x(t) e^{-j2\pi f_c t} \times e^{j2\pi f'_c t}$$

TX & RX are not synched

$$\xrightarrow{\quad} h(t) * x(t) e^{-j2\pi \Delta f_c t}$$

CFO:  $\Delta f_c = f_c - f'_c$

$$\xrightarrow{\quad} y(t) = h(t) * x(t) e^{-j2\pi \Delta f_c t} + v(t)$$

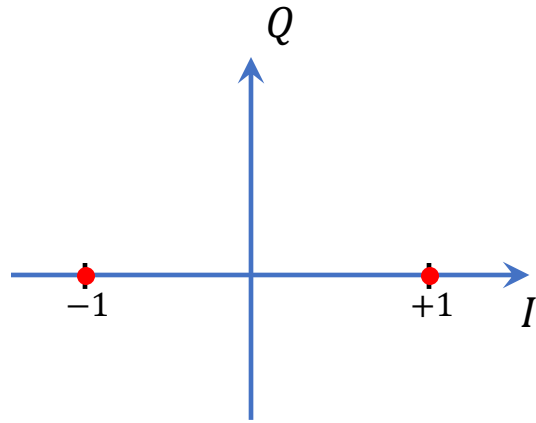
**Phase changes with time!**

# Carrier Frequency Offset

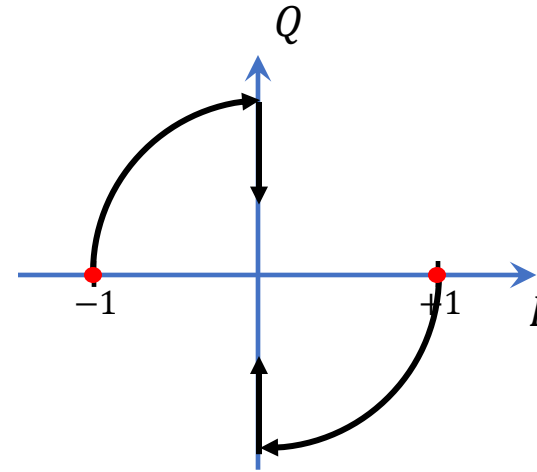
Consider BPSK Modulation.

0  $\rightarrow$  -1

1  $\rightarrow$  +1



$x(t)$



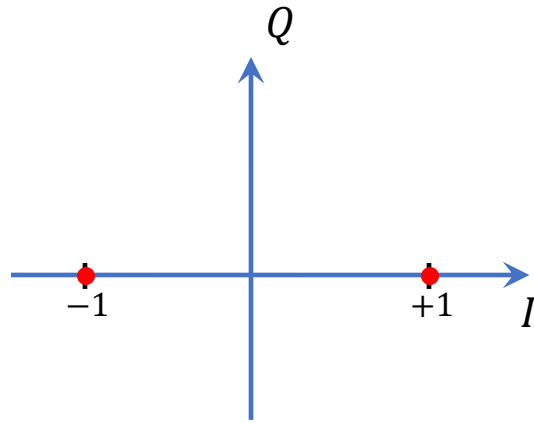
$h x(t - \tau) e^{-j2\pi\Delta f_c t} + v(t)$

# Carrier Frequency Offset

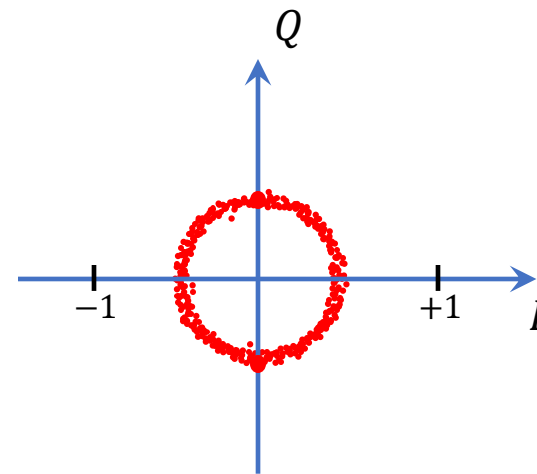
Consider BPSK Modulation.

0  $\rightarrow$  -1

1  $\rightarrow$  +1



$x(t)$

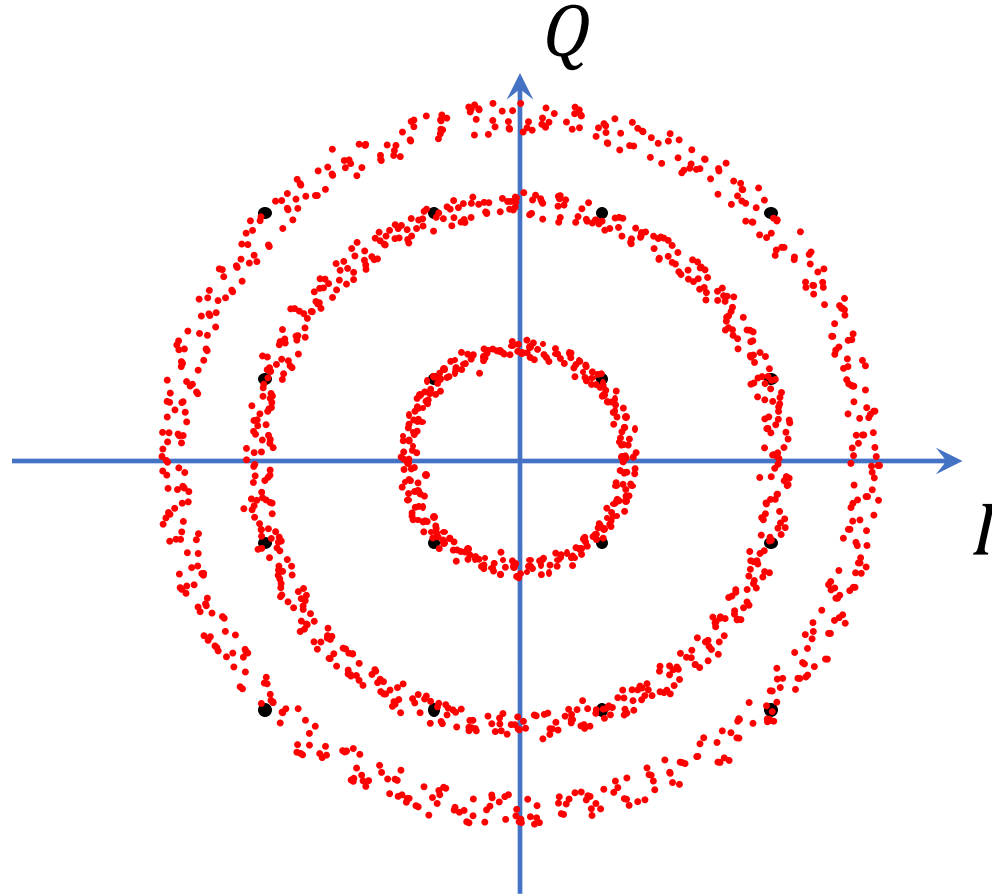


$h x(t - \tau) e^{-j2\pi\Delta f_c t} + v(t)$

**Impossible to Decode!**

# Carrier Frequency Offset

Consider 16 QAM Modulation



**Need to estimate and correct CFO to decode!**

# OFDM CFO Estimation & Correction



- Use Preamble to estimate CFO

$$y_1(t) = x(t)e^{-j2\pi\Delta f_c t}$$

Sample at  $t = nT_s$

Symbol has  $N$  samples  $\rightarrow T = NT_s$

$$y_2(t) = x(t)e^{-j2\pi\Delta f_c(t+T)}$$

# OFDM CFO Estimation & Correction



- Use Preamble to estimate CFO

$$y_1[n] = x[n]e^{-j2\pi\Delta f_c n T_s}$$

Sample at  $t = nT_s$

Symbol has  $N$  samples  $\rightarrow T = NT_s$

$$y_2[n] = x[n]e^{-j2\pi\Delta f_c (nT_s + NT_s)}$$

- Compute:  $A = \sum_{n=1}^N y_1[n]y_2^*[n] = \sum_{n=1}^N x[n]x^*[n]e^{j2\pi\Delta f_c NT_s}$

$$= e^{j2\pi\Delta f_c NT_s} \sum_{n=1}^N |x[n]|^2 \quad \rightarrow \quad \Delta f_c = \frac{\angle A}{2\pi NT_s}$$

# OFDM CFO Estimation & Correction



- Use Preamble to estimate CFO

$$y_1[n] = x[n]e^{-j2\pi\Delta f_c nT_s}$$

$$y_2[n] = x[n]e^{-j2\pi\Delta f_c (nT_s + NT_s)}$$

- Compute:  $A = \sum_{n=1}^N y_1[n]y_2^*[n] \rightarrow \Delta f_c = \frac{\angle A}{2\pi NT_s}$

- Correct CFO:  $y[n] \times e^{j2\pi\Delta f_c nT_s}$

# OFDM Channel Estimation



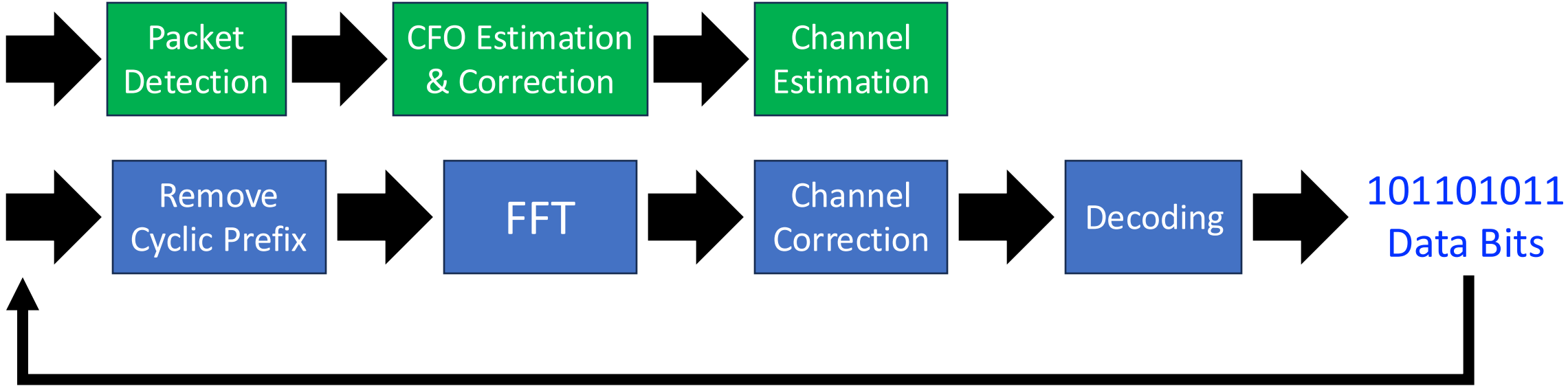
- Use Preamble to estimate the channel

$$y(t) = h(t) * x(t) \leftrightarrow Y(f) = H(f) X(f)$$

- Send  $X(f)$  :  $-1, +1, -1, -1, -1, +1, \dots$
- Receive:  $-H(1), H(2), -H(3), -H(4), -H(5), H(6), \dots$
- Estimate:  $\tilde{H}(f) = \frac{Y(f)}{X(f)}$
- Use two preambles to average noise:  $\tilde{H}(f) = \frac{Y_1(f) + Y_2(f)}{2 X(f)}$

# OFDM Receiver

Received  
Signal



Repeat until all OFDM Symbols are decoded

# Phase Tracking

So Far: Estimated and Corrected For Coarse Value of CFO

- Residual CFO:

$$y(t) = h(t) x(t) e^{-j2\pi\Delta f_c t} + v(t)$$

$$\Delta f_c = df_c + \delta f_c$$

Coarse CFO

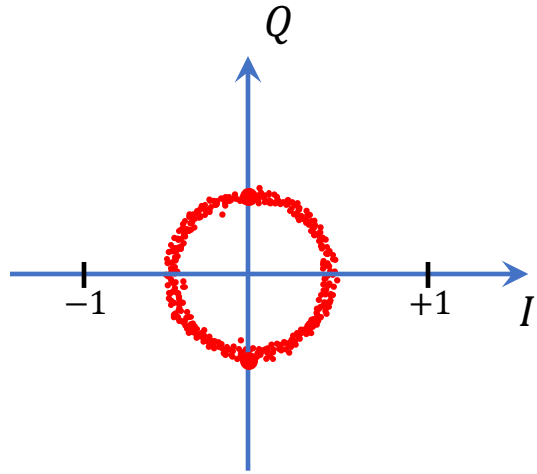
Residual CFO

We estimated and corrected for coarse CFO!

Even small residual can accumulate over time  
to create large phase:  $e^{-j2\pi\delta f_c t}$

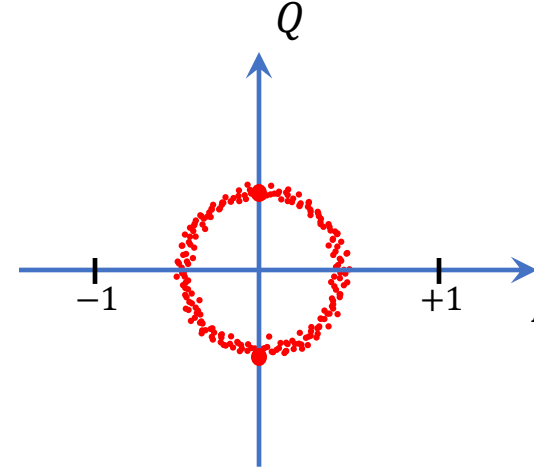
Need to track the phase

# Phase Tracking



$$h x(t - \tau) e^{-j2\pi\Delta f_c t} + v(t)$$

Without Course CFO Correction



$$h x(t - \tau) e^{-j2\pi\delta f_c t} + v(t)$$

Without Residual CFO Correction

# Phase Tracking

- Residual CFO (Carrier Frequency Offset)
- Residual SFO (Sampling Frequency Offset)

# Phase Tracking

- Residual CFO (Carrier Frequency Offset)
- Residual SFO (Sampling Frequency Offset)

When we sample the signal there is a residual sampling offset:  $n\delta T_s$

$$y[n] = x[n + n\delta T_s]e^{-j2\pi\delta f_c n T_s}$$

$$= \sum_{f=0}^{N-1} X[f]e^{j\frac{2\pi f(n+n\delta T_s)}{N}} e^{-j2\pi\delta f_c n T_s}$$

$$Y[f] = X[f]e^{j\frac{2\pi f n \delta T_s}{N} - 2\pi\delta f_c n T_s}$$

# Phase Tracking

- Residual CFO (Carrier Frequency Offset)
- Residual SFO (Sampling Frequency Offset)

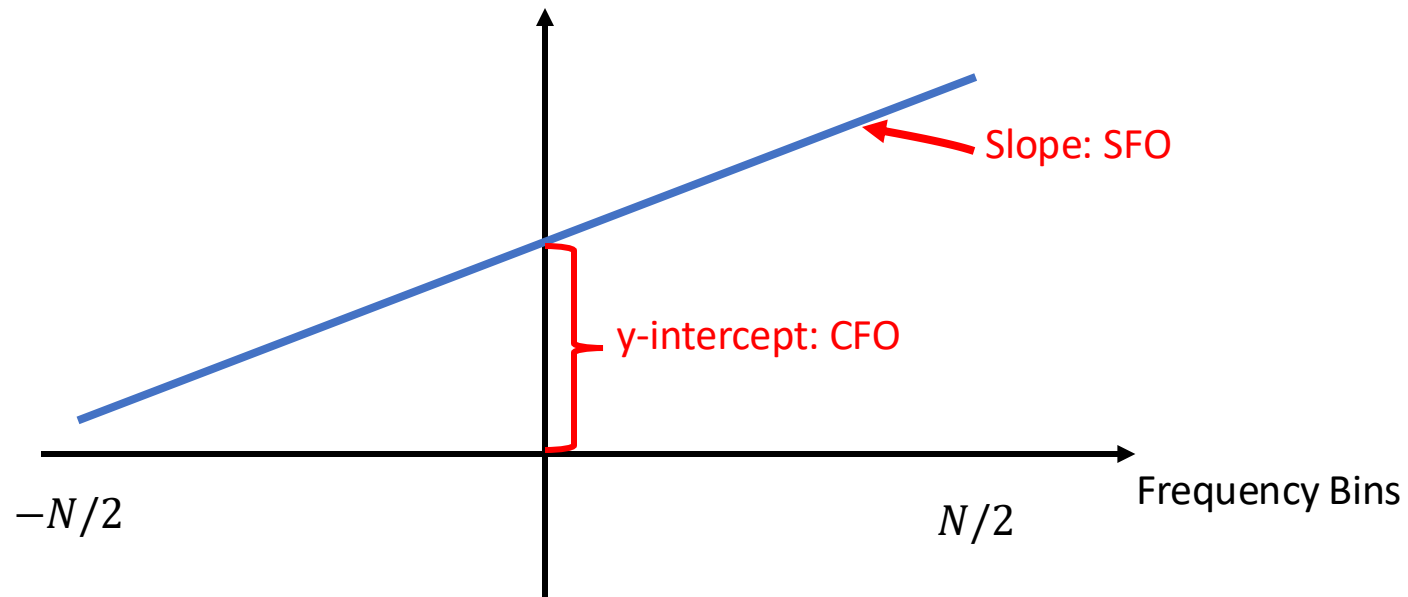
$$Y_1[f] = X_1[f] e^{j \frac{2\pi f n \delta T_s}{N} - 2\pi \delta f_c n T_s}$$

$$Y_2[f] = X_2[f] e^{j \frac{2\pi f (n+N+CP) \delta T_s}{N} - 2\pi \delta f_c (n+N+CP) T_s}$$

Phase accumulation:  $\Delta\phi = 2\pi f \frac{(N + CP) \delta T_s}{N} - 2\pi \delta f_c (N + CP) T_s$

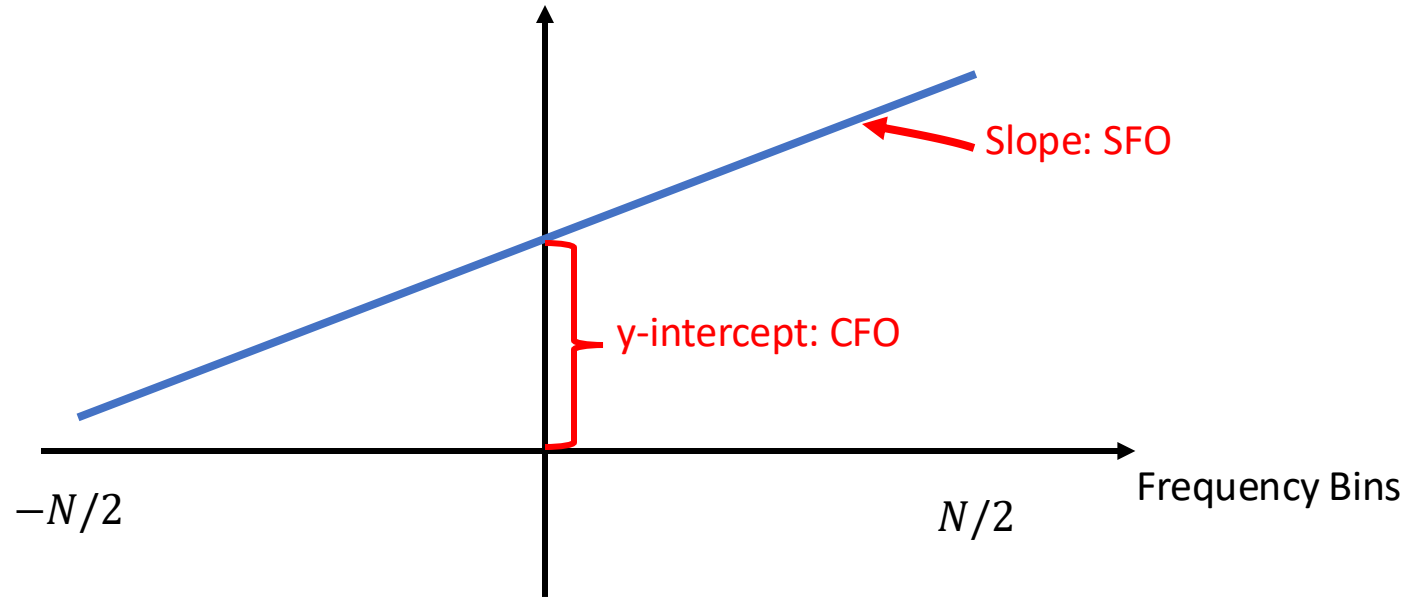
# Phase Tracking

- Residual CFO (Carrier Frequency Offset)
- Residual SFO (Sampling Frequency Offset)
- Phase accumulation:  $\Delta\phi = 2\pi f \frac{(N + CP)\delta T_s}{N} - 2\pi\delta f_c(N + CP)T_s$



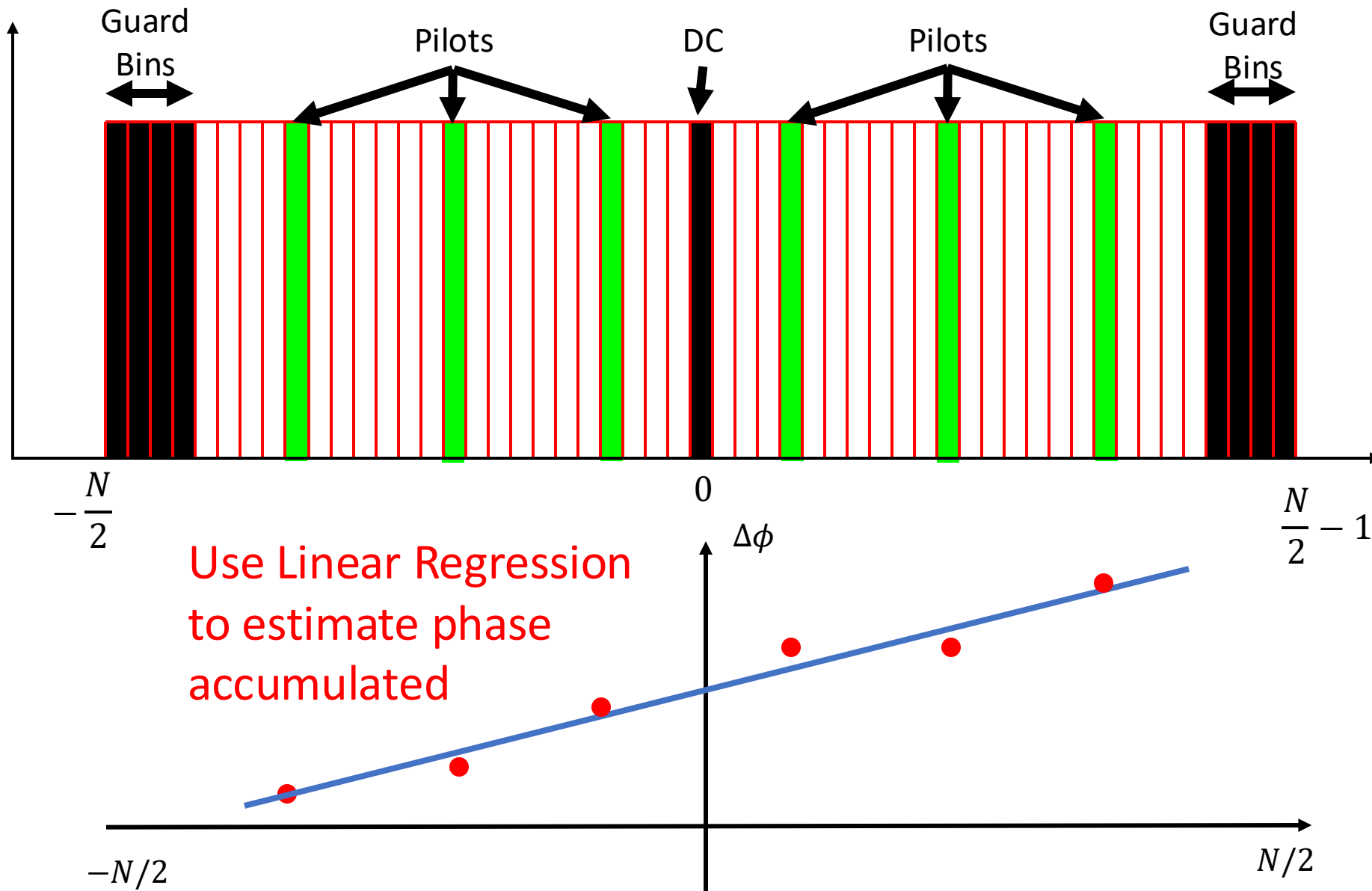
# OFDM Phase Tracking

- Phase accumulation:  $\Delta\phi = 2\pi f \frac{(N + CP)\delta T_s}{N} - 2\pi\delta f_c(N + CP)T_s$

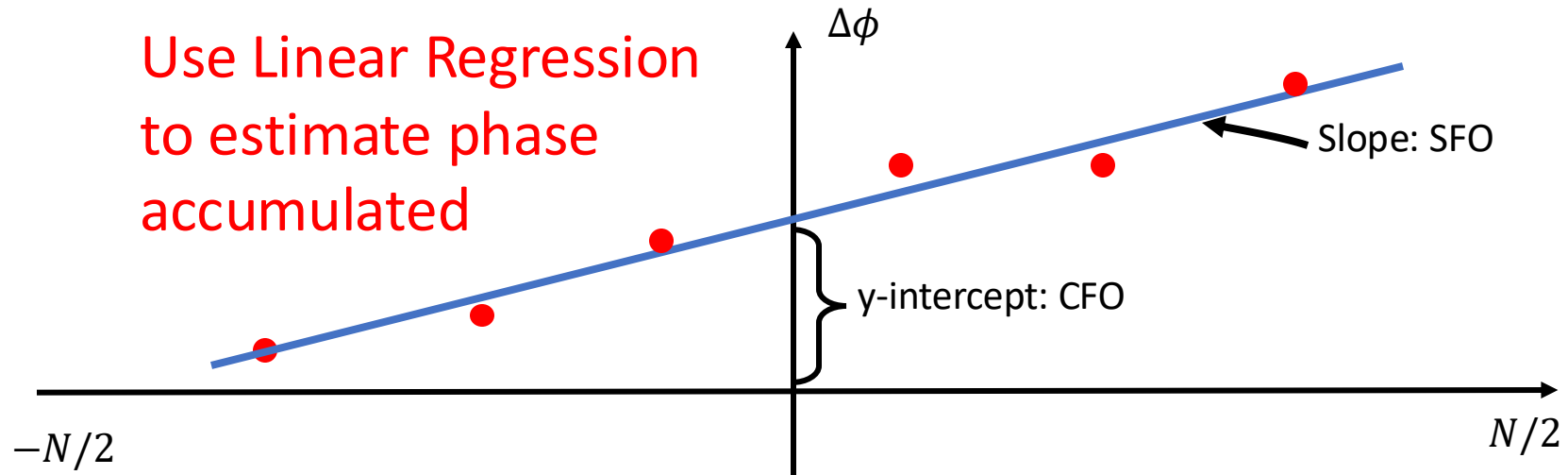


- Sufficient to estimate slope & y-intercept to know the phase accumulated for all subcarriers.
- Use only few subcarriers as pilots & send known bits in them.

# OFDM Phase Tracking



# OFDM Phase Tracking



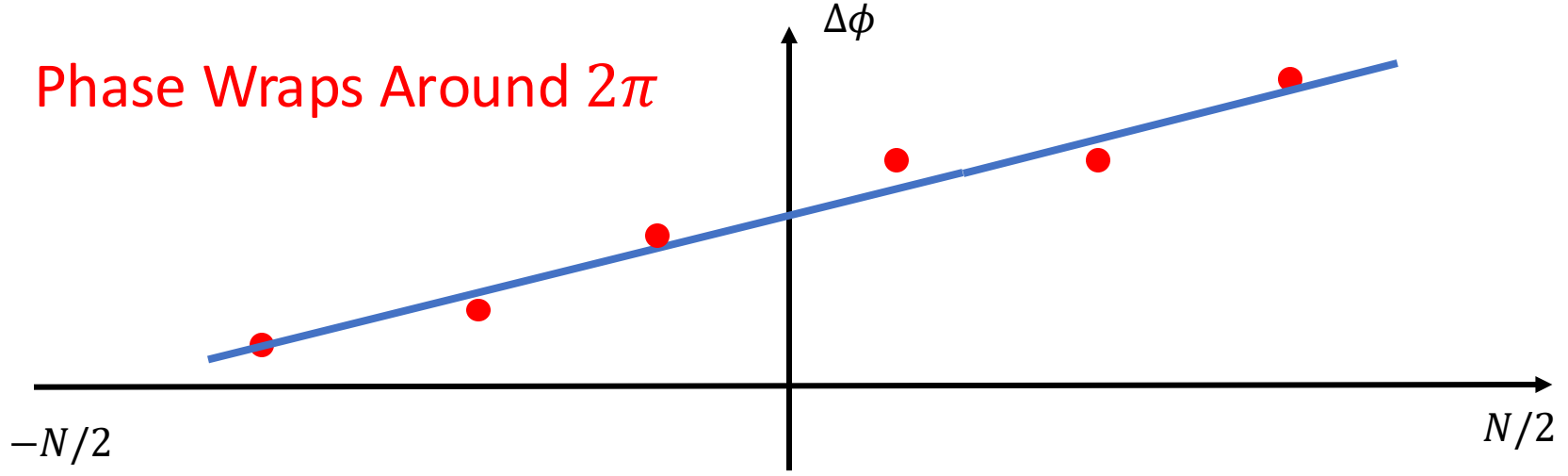
$$\text{Slope} = 2\pi \frac{(N + CP)\delta T_s}{N}$$

$$Y - \text{intercept} = -2\pi\delta f_c(N + CP)T_s$$

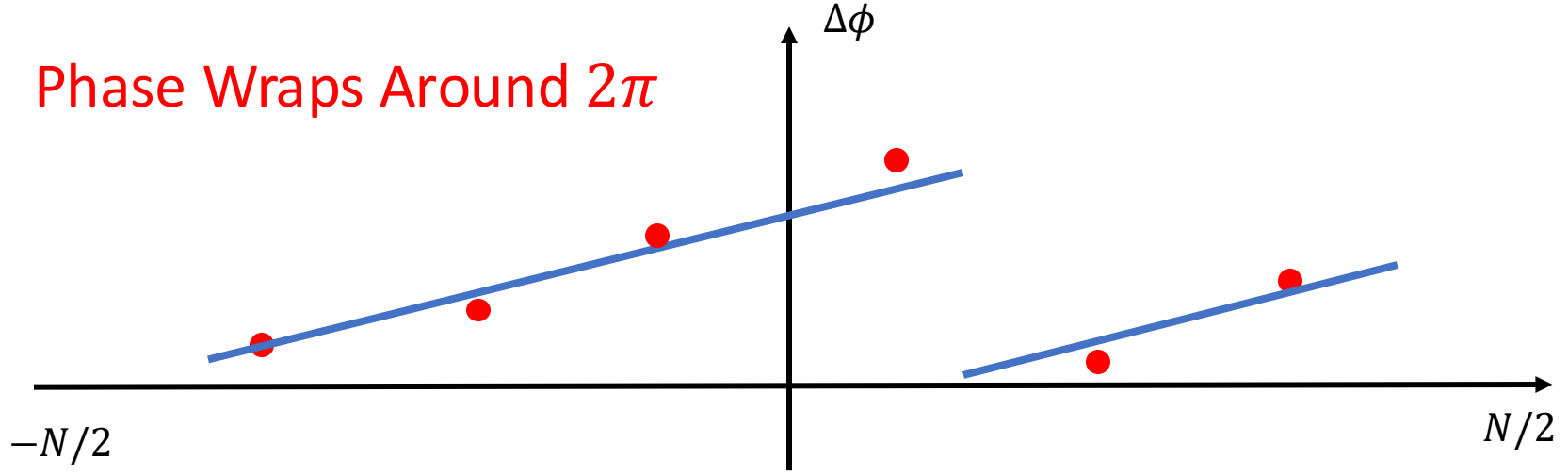
$$\Delta\phi = 2\pi f \frac{(N + CP)\delta T_s}{N} - 2\pi\delta f_c(N + CP)T_s$$

$$\tilde{H}_{new}(f) = \tilde{H}_{old}(f) \times e^{j\Delta\phi}$$

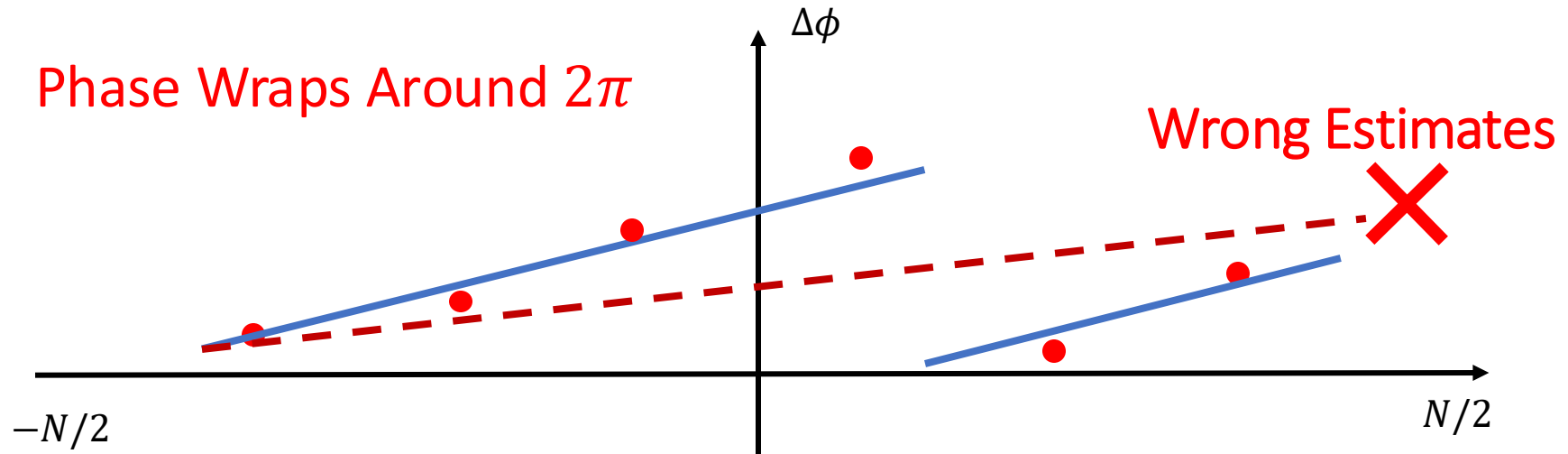
# Be Careful!



# Be Careful!



# Be Careful!

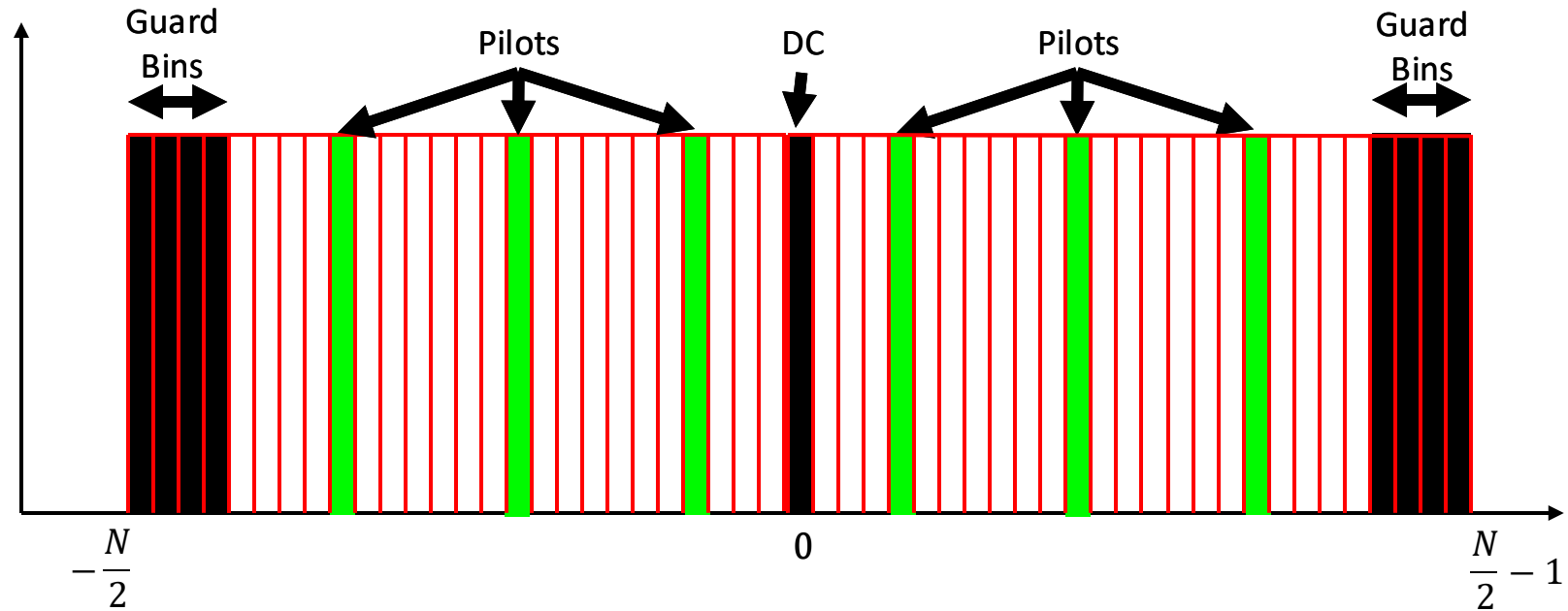


Need to unwrap phase mod  $2\pi$  first!

When estimating slope and y-intercept, equation assumes subcarriers index  $f$  is between  $-N/2$  and  $N/2 - 1$

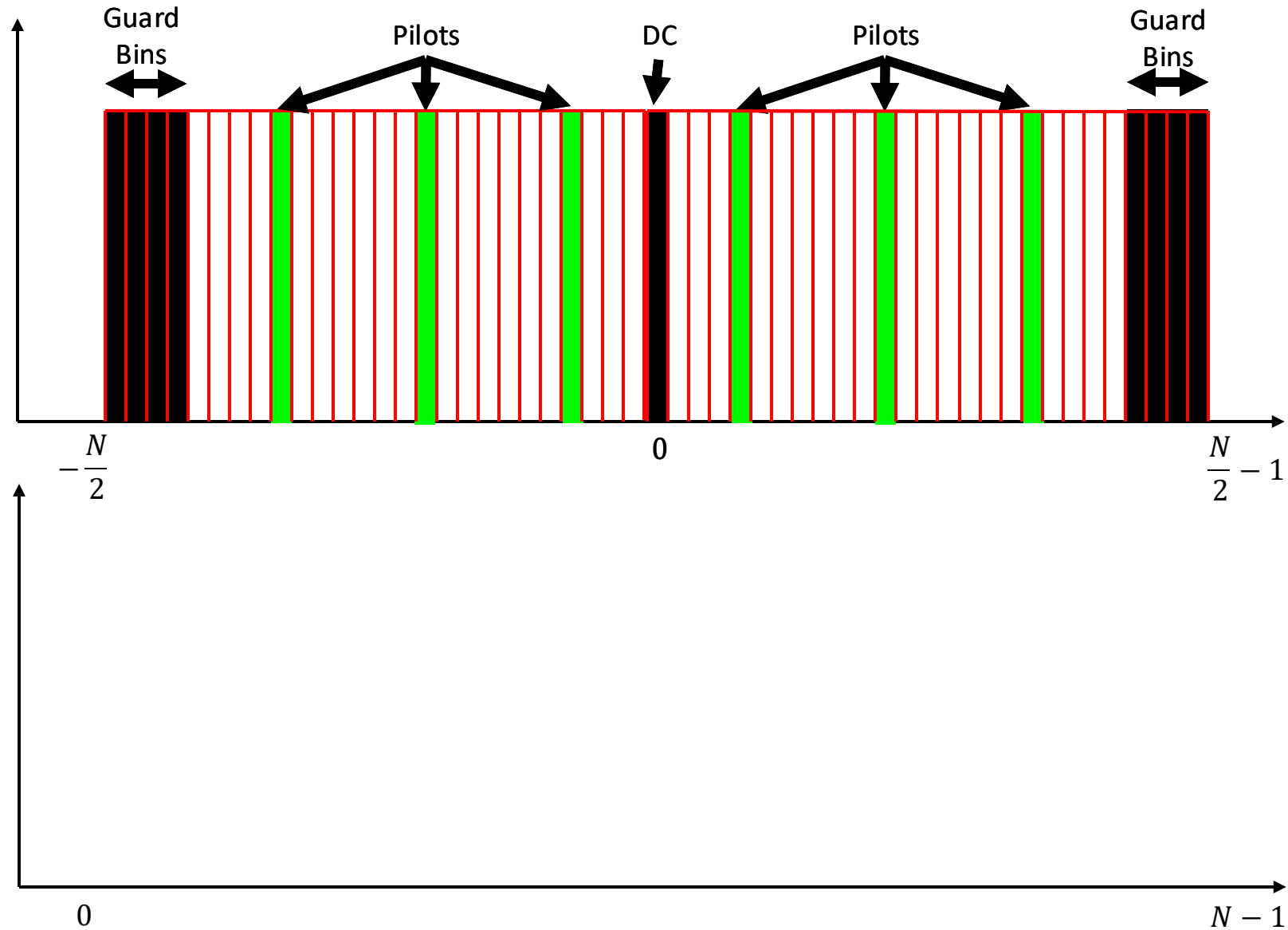
# Be Careful!

When estimating slope and y-intercept, equation assumes subcarriers index  $f$  is between  $-N/2$  and  $N/2 - 1$



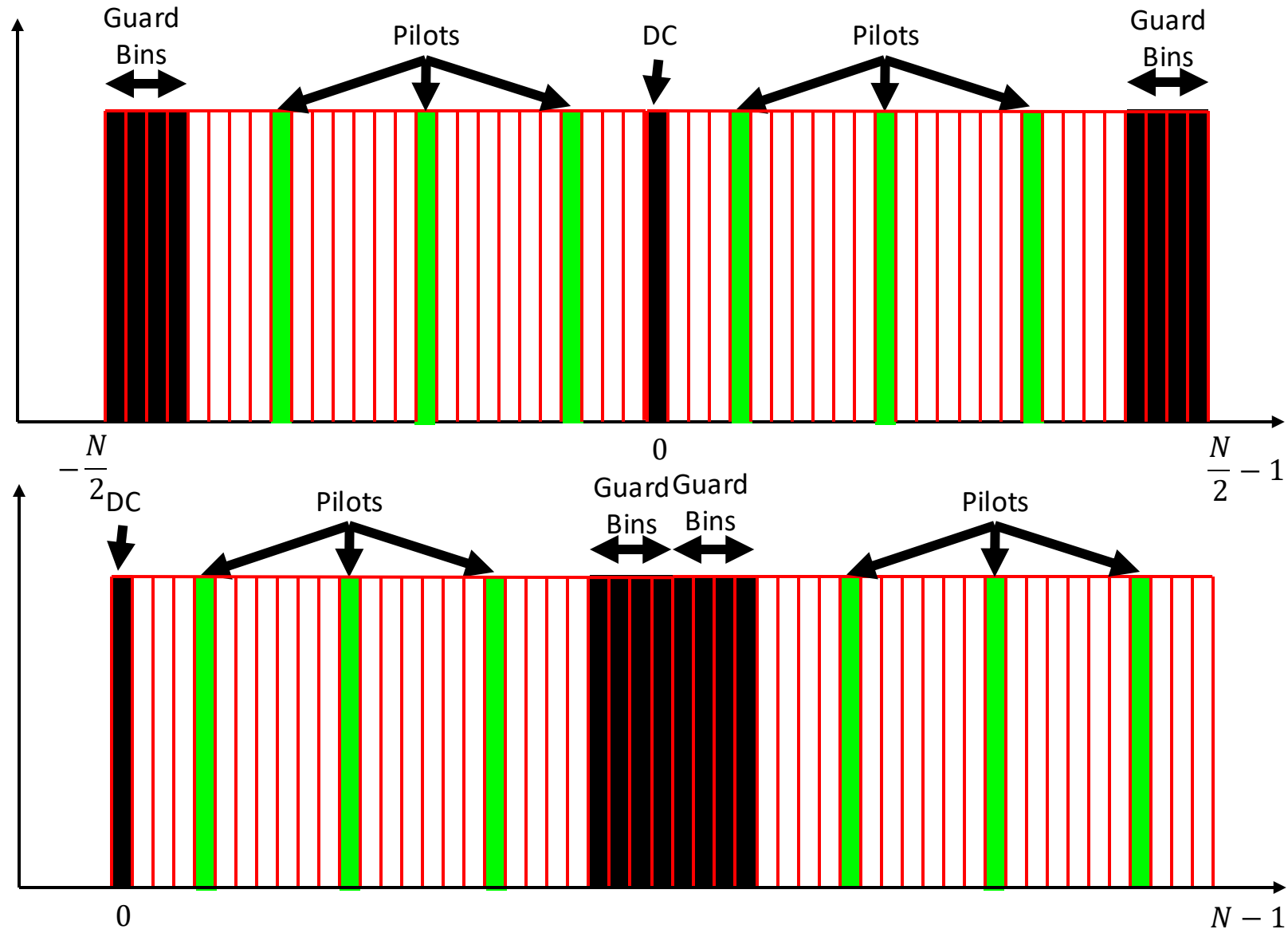
# Be Careful!

When estimating slope and y-intercept, equation assumes subcarriers index  $f$  is between  $-N/2$  and  $N/2 - 1$



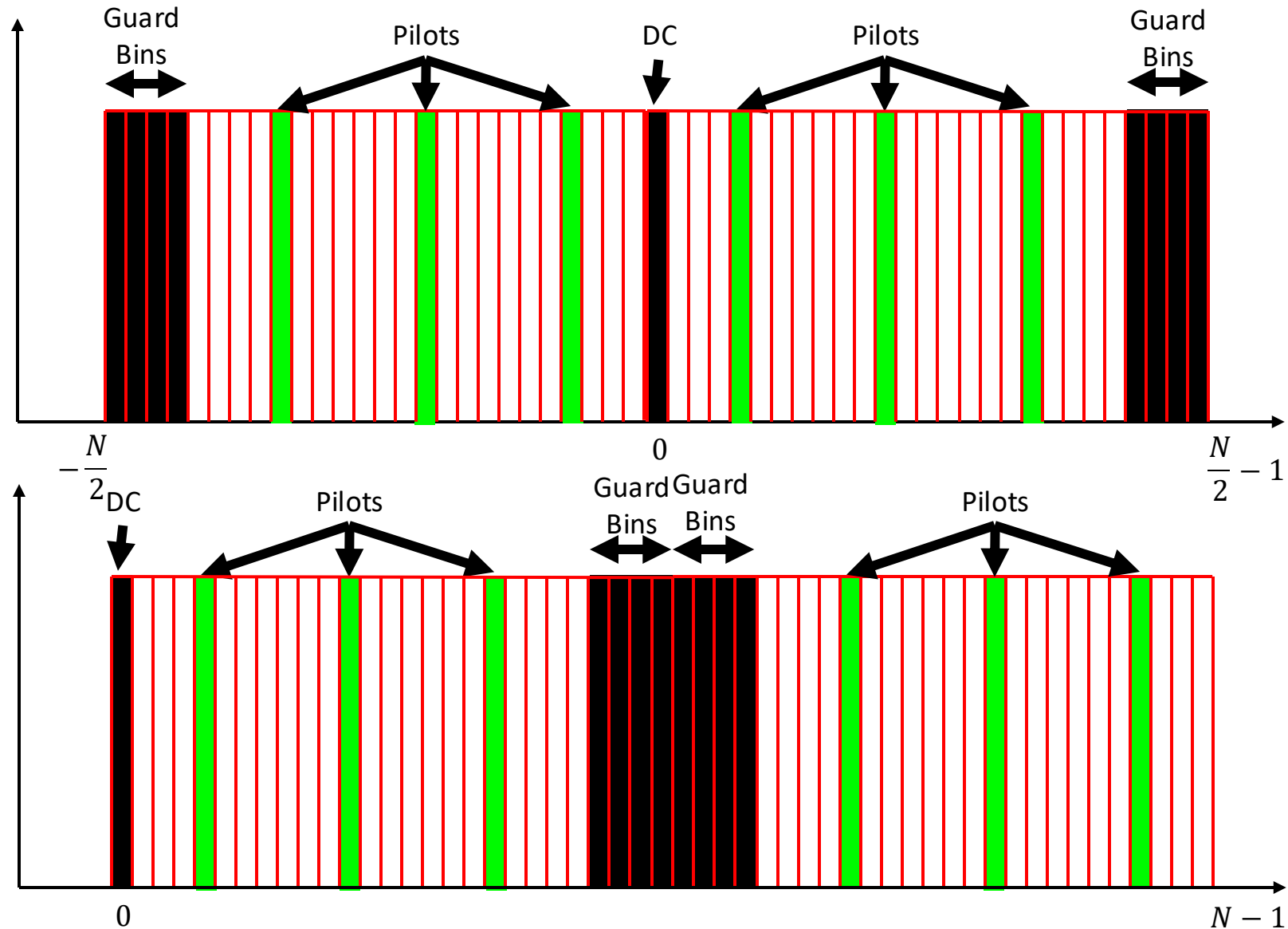
# Be Careful!

When estimating slope and y-intercept, equation assumes subcarriers index  $f$  is between  $-N/2$  and  $N/2 - 1$

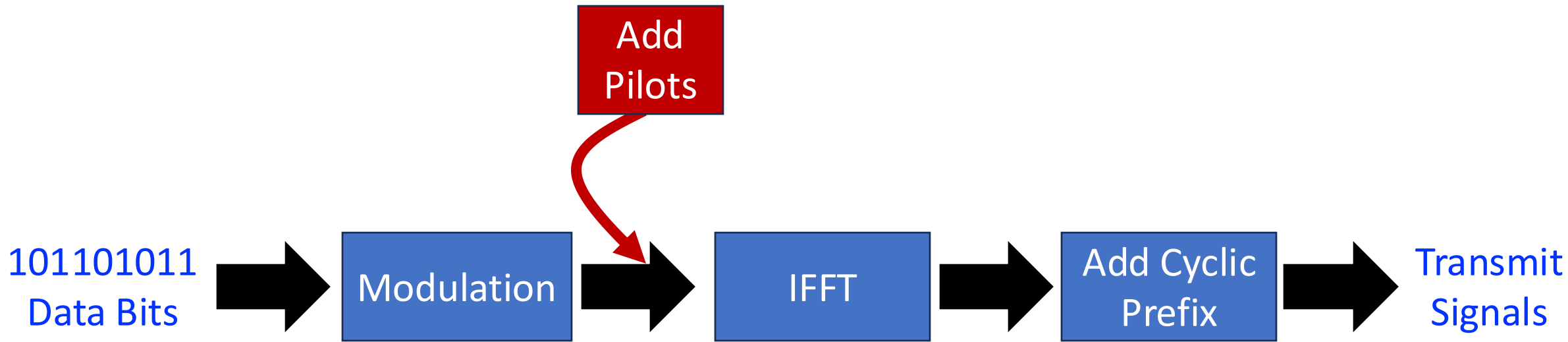
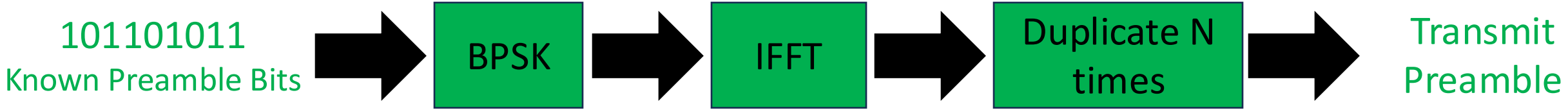


# Be Careful!

When estimating slope and y-intercept, equation assumes subcarriers index  $f$  is between  $-N/2$  and  $N/2 - 1$



# OFDM Transmitter



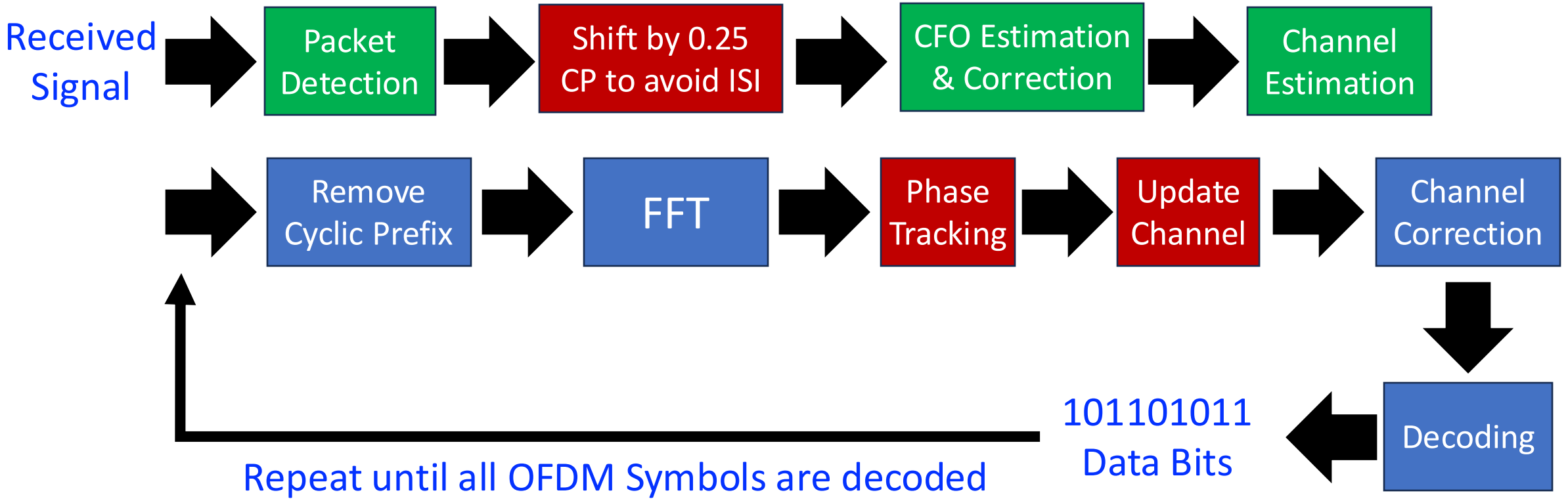
Repeat until all data bits are transmitted

# OFDM: Putting it Together

## At TX:

- Create preamble symbol from training sequence (Uses BPSK)
- Repeat preamble symbol:
  - 4 times for packet detection
  - 2 times for CFO estimation
  - 2 times for channel estimation
  - Add CP for the last preamble
- Create data symbol from:
  - Data bits (Uses BPSK, QPSK, M-QAM)
  - Pilot bits (Uses BPSK)
- Add cyclic prefix to data symbols.

# OFDM Receiver



# OFDM: Putting it Together

## At RX:

- Detect beginning of packet.
- Jump  $\approx 0.25 CP$  samples into symbol to avoid ISI
- Estimate & correct for CFO.
- Estimate the channel.
- For each subsequent data symbol:
  - Remove CP
  - Take FFT of Size N
  - Use linear regression to estimate residual CFO and SFO
  - Estimate accumulated phase  $\Delta\phi(f)$  for each frequency bin
  - Add  $\Delta\phi(f)$  to channel estimate  $\tilde{H}(f)$
  - Correct for channel by dividing with  $\tilde{H}(f)$
  - Decode Bits