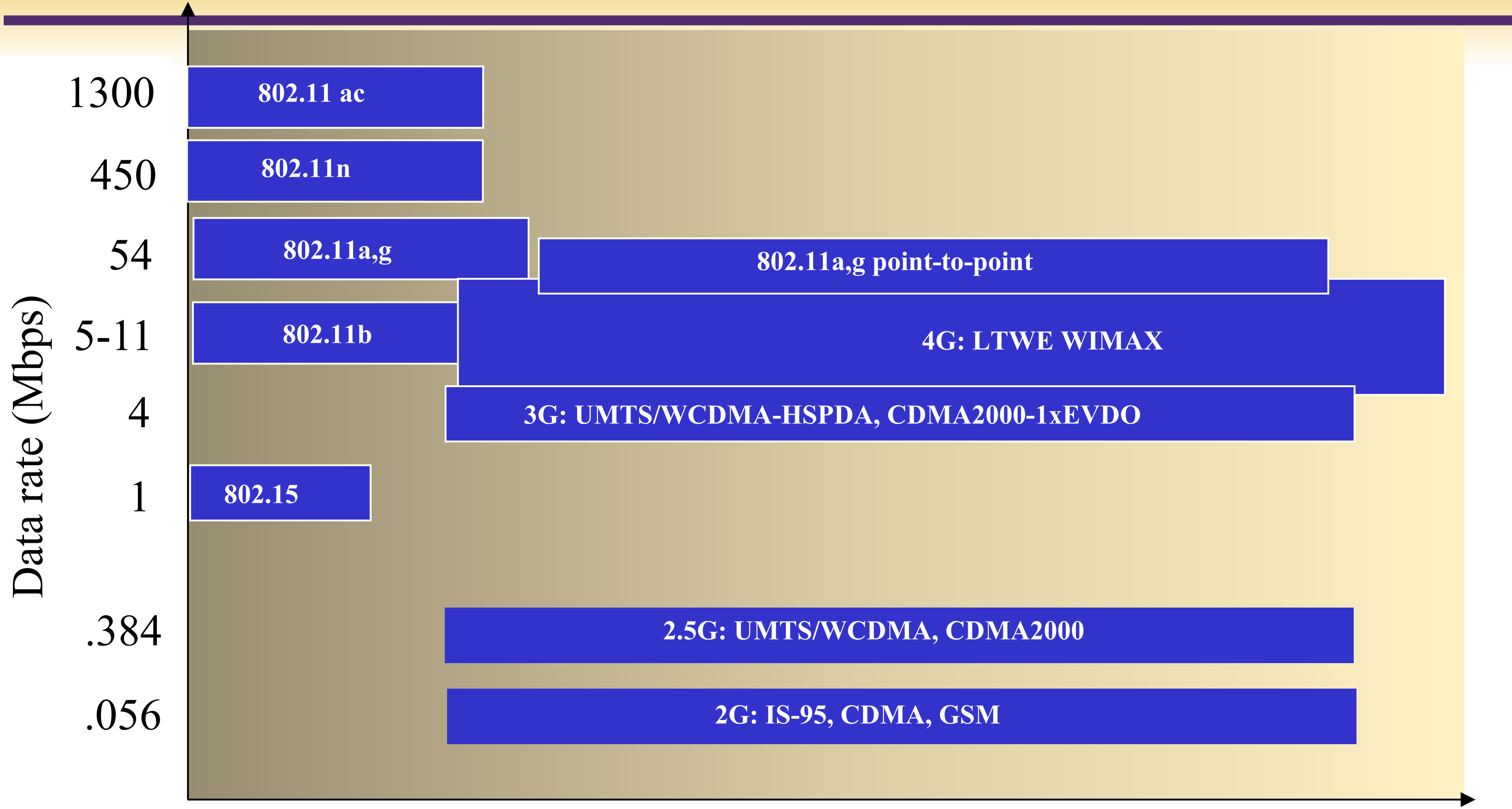
Modern Wireless Networks

Wireless LANs

UNIVERSITY AT ALBANY
State University of New York

IECE 574– Spring 2021
Prof. Dola Saha

Wireless Standards / Protocols



Data rate (Mbps)

1300

450

54

5-11

4

1

.384

.056

802.11 ac

802.11n

802.11a,g

802.11a,g point-to-point

802.11b

4G: LTWE WIMAX

3G: UMTS/WCDMA-HSPDA, CDMA2000-1xEVDO

802.15

2.5G: UMTS/WCDMA, CDMA2000

2G: IS-95, CDMA, GSM

Indoor
10-30m

Outdoor
50-200m

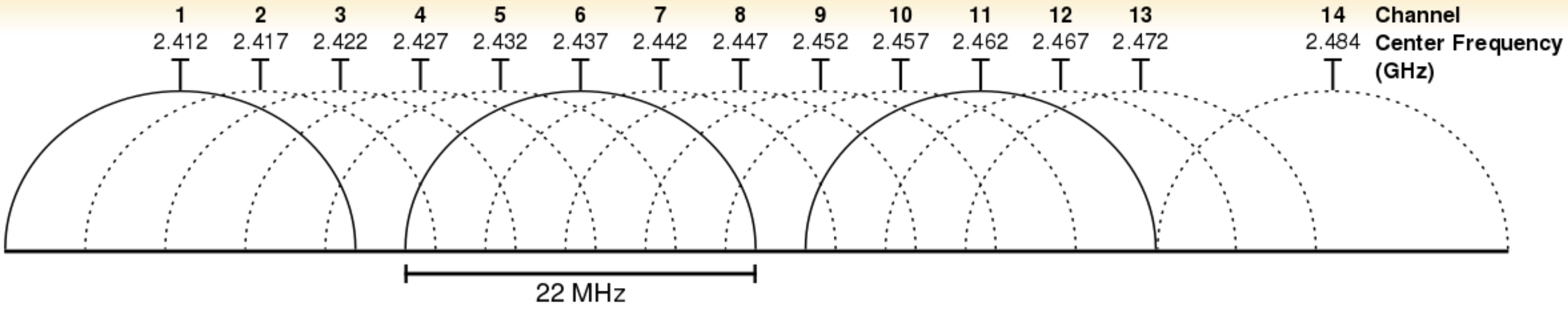
Mid-range outdoor
200m – 4 Km

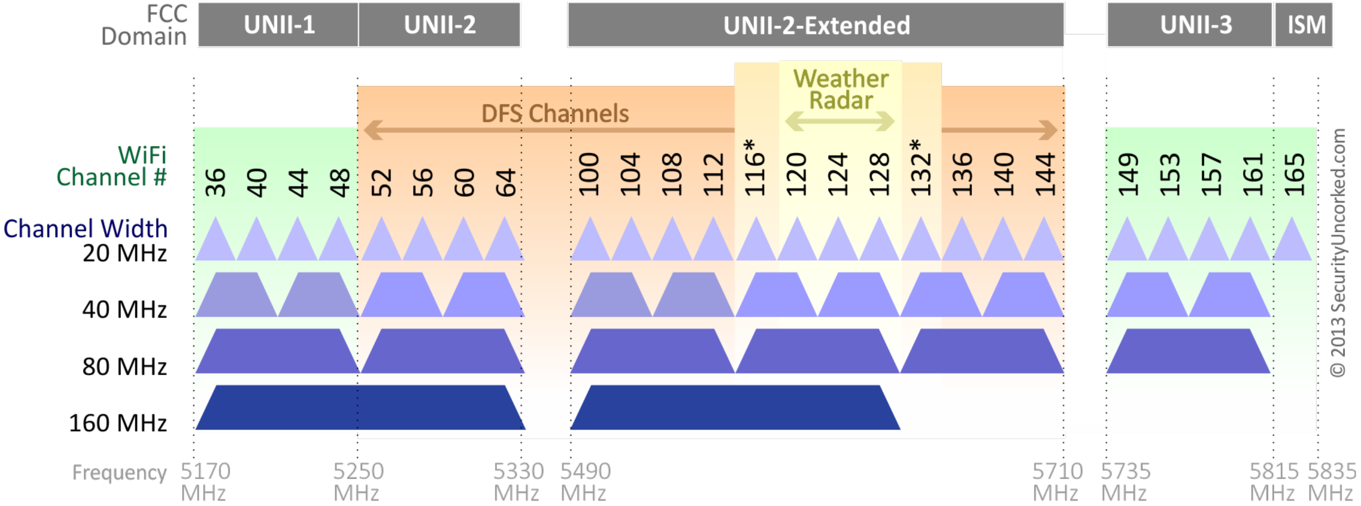
Long-range outdoor
5Km – 20 Km

Wireless Network Taxonomy

	Single Hop	Multiple Hop
Infrastructure	host connects to base station (WiFi, WiMAX, cellular), which connects to larger Internet	host may have to relay through several wireless nodes to connect to larger Internet: *mesh network*
Non-infrastructure	no base station, no connection to larger Internet (Bluetooth, ad hoc nets)	no base station, no connection to larger Internet. May have to relay to reach other a given wireless node MANET, VANET

Channels (Frequency)

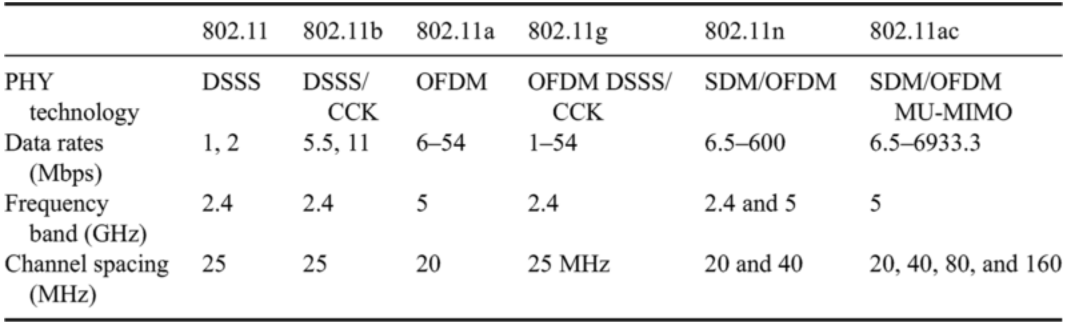




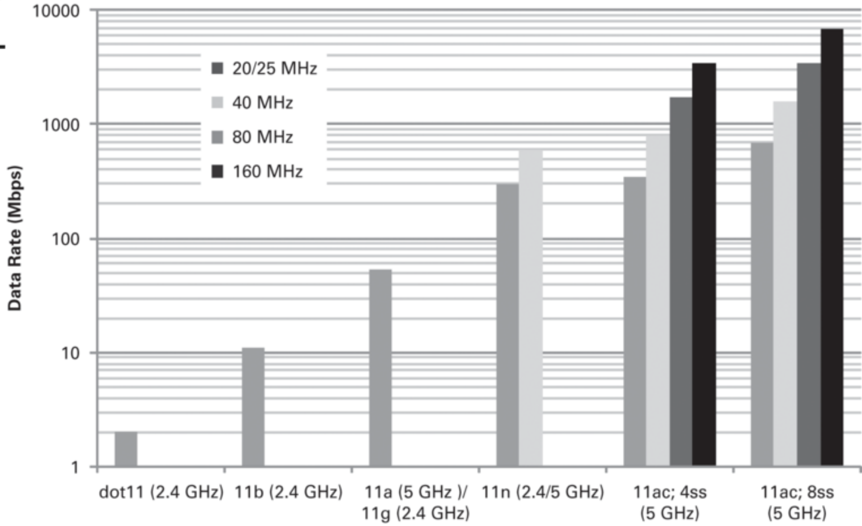
IEEE 802.11 Wireless LANs

➢ MAC and PHY Standard

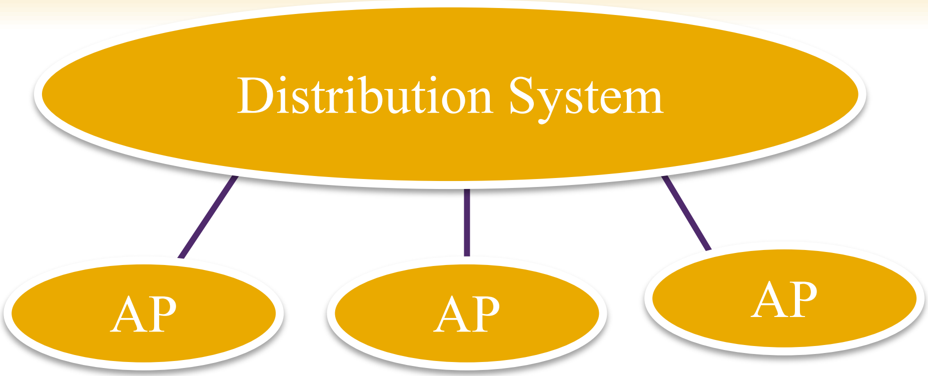
http://www.ieee802.org/11/

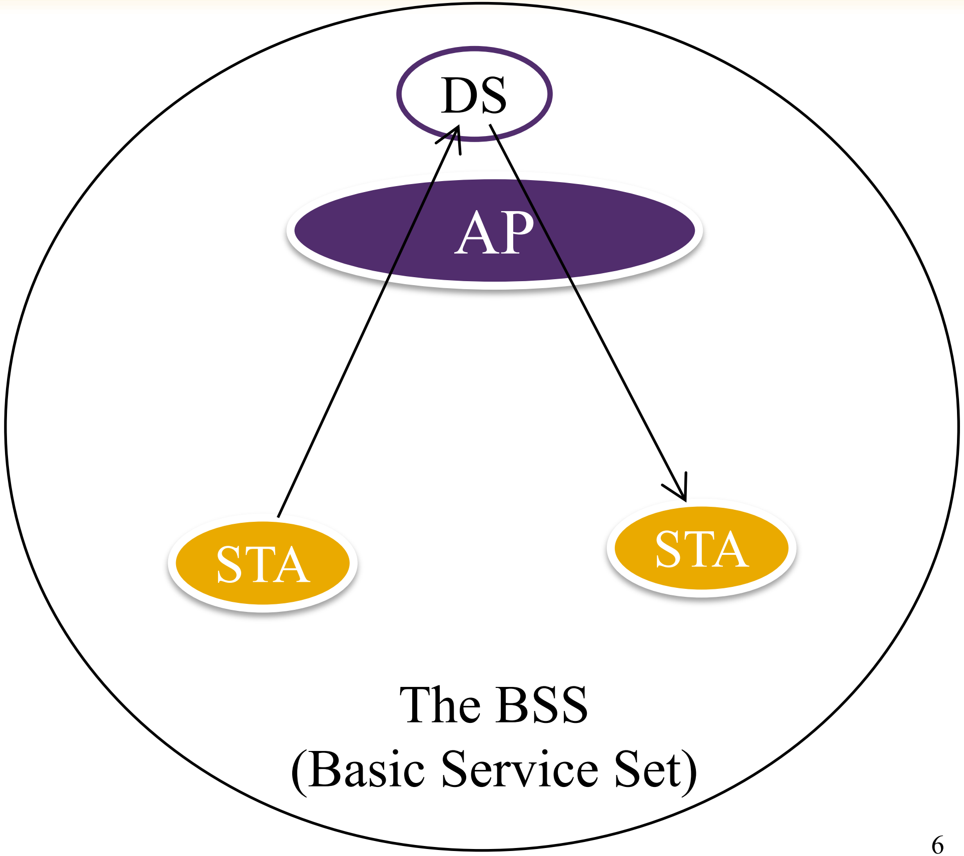


	802.11	802.11b	802.11a	802.11g	802.11n	802.11ac
PHY technology	DSSS	DSSS/ CCK	OFDM	OFDM DSSS/ CCK	SDM/OFDM	SDM/OFDM MU-MIMO
Data rates (Mbps)	1, 2	5.5, 11	6–54	1–54	6.5–600	6.5–6933.3
Frequency band (GHz)	2.4	2.4	5	2.4	2.4 and 5	5
Channel spacing (MHz)	25	25	20	25 MHz	20 and 40	20, 40, 80, and 160



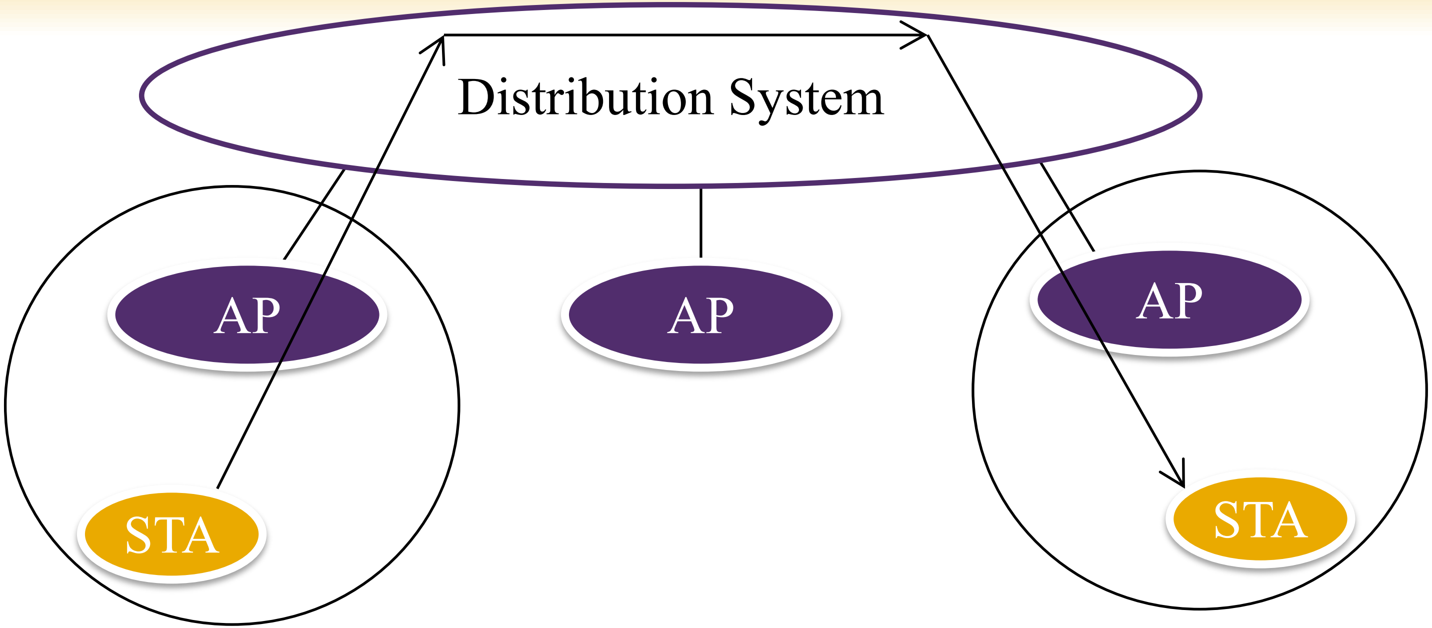
Network Architecture





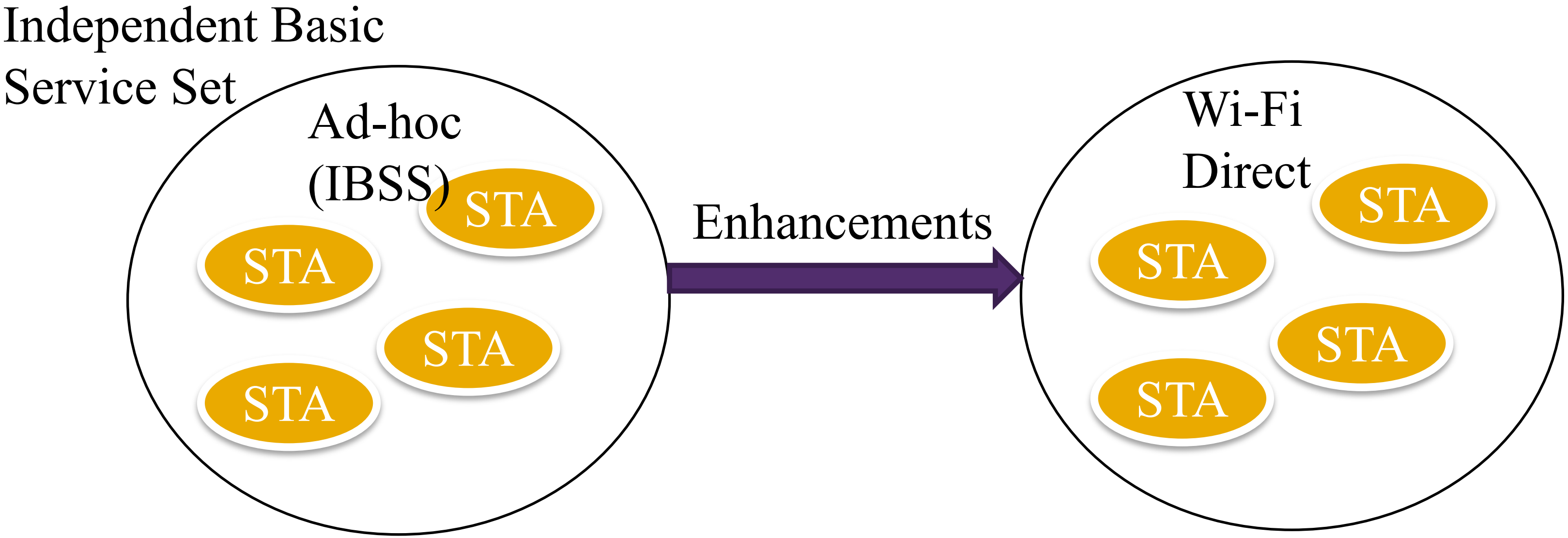
The BSS
(Basic Service Set)

Network Architecture



- A set of one or more interconnected basic service sets (BSSs) and integrated local area networks (LANs) that appears as a single BSS to the logical link control layer at any station associated with one of those BSSs.

Infrastructure-less

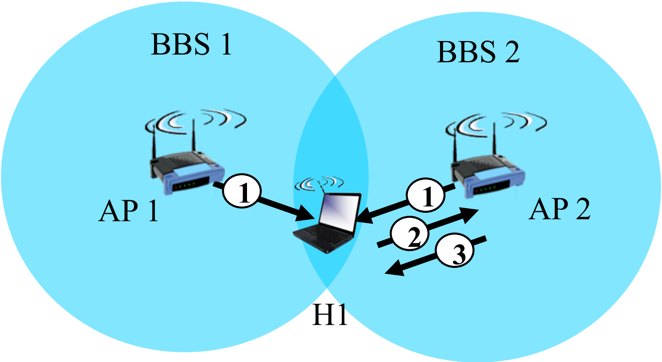

Independent Basic Service Set

Ad-hoc (IBSS)

Enhancements

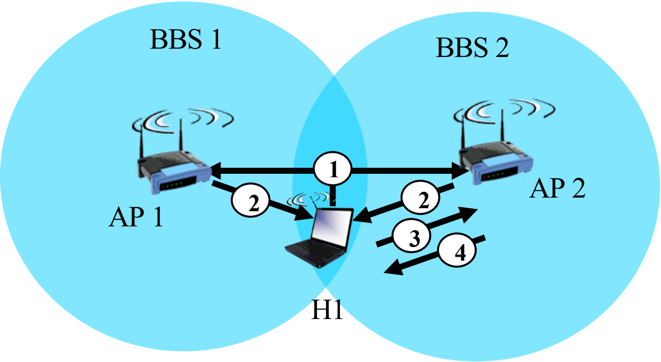
Wi-Fi Direct

802.11: passive/active scanning



passive scanning:

(1) beacon frames sent from APs
(2) association Request frame sent: H1 to selected AP
(3) association Response frame sent from selected AP to H1

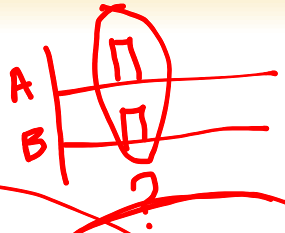


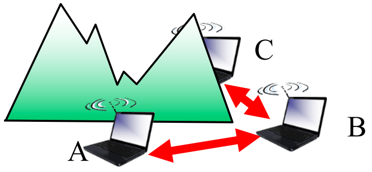
active scanning:

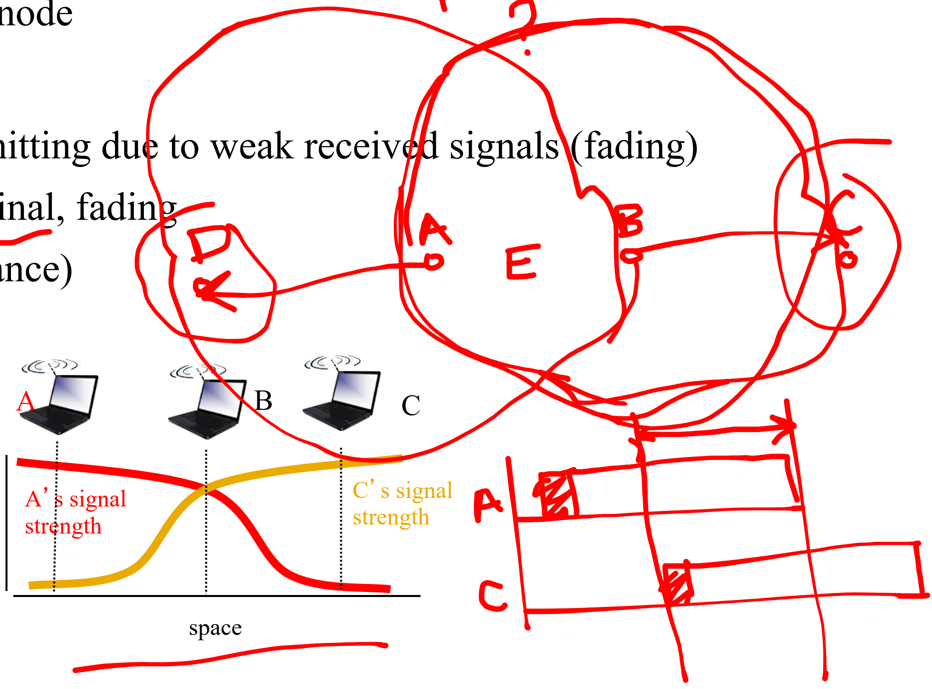
(1) Probe Request frame broadcast from H1
(2) Probe Response frames sent from APs
(3) Association Request frame sent: H1 to selected AP
(4) Association Response frame sent from selected AP to H1

IEEE 802.11: multiple access

- avoid collisions: 2+ nodes transmitting at same time
- 802.11: CSMA - sense before transmitting
  - don’t collide with ongoing transmission by other node
- 802.11: *no* collision detection!
  - difficult to receive (sense collisions) when transmitting due to weak received signals (fading)
  - can’t sense all collisions in any case: hidden terminal, fading
  - goal: *avoid collisions:* CSMA/C(ollision)A(voidance)







IEEE 802.11 MAC Protocol: CSMA/CA

802.11 sender

1 if sense channel idle for **DIFS** then

transmit entire frame (no CD)

2 if sense channel busy then

start random backoff time

timer counts down while channel idle

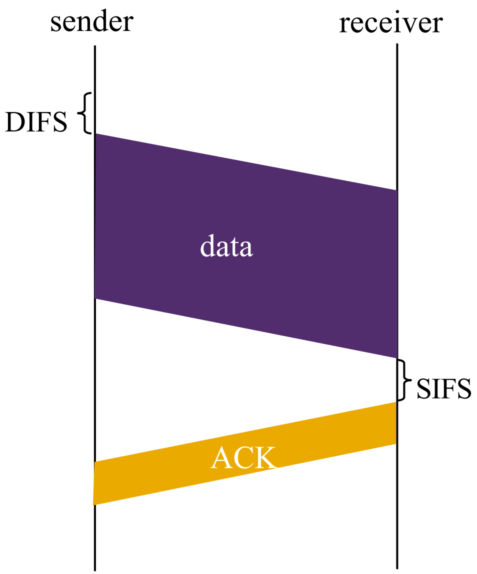
transmit when timer expires

if no ACK, increase random backoff interval, repeat 2

802.11 receiver

- if frame received OK

return ACK after **SIFS** (ACK needed due to hidden terminal problem)



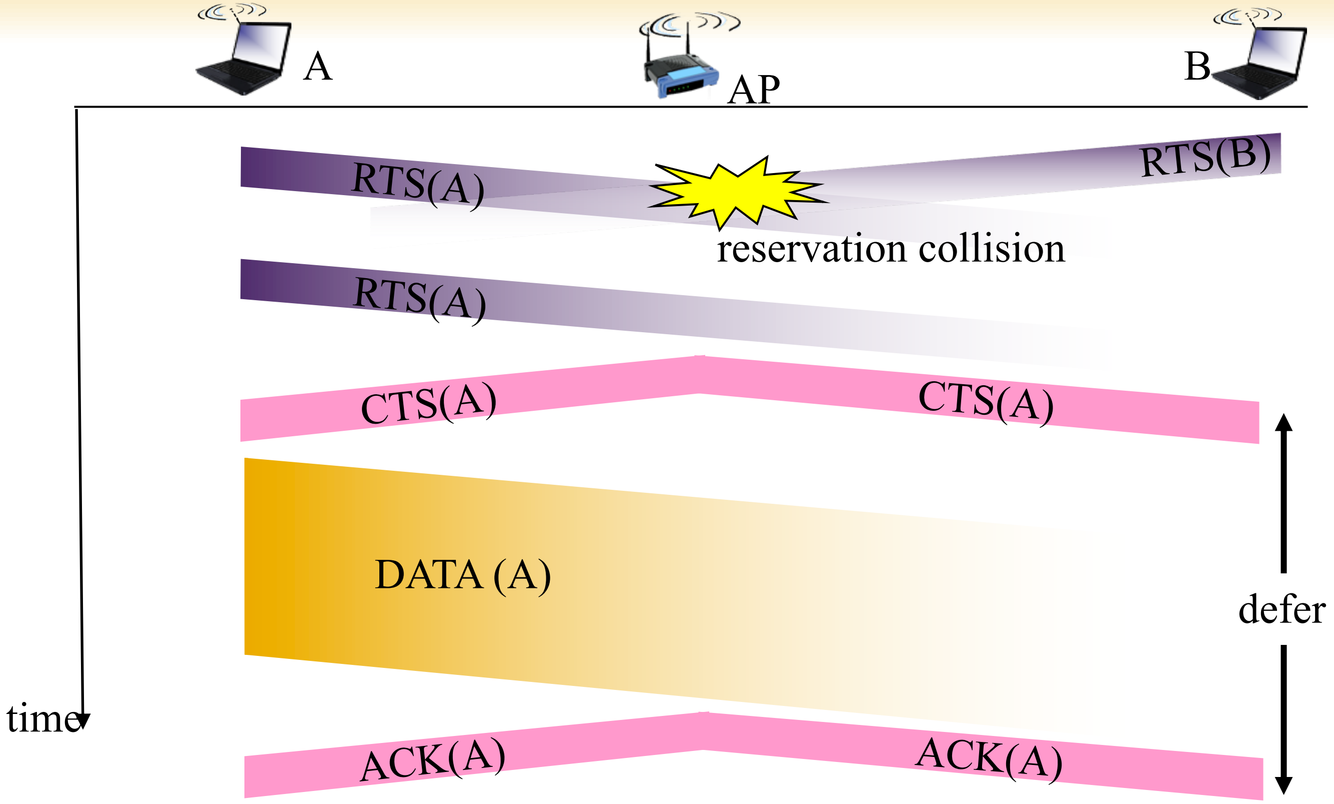
Avoiding collisions (more)

idea: allow sender to "reserve" channel rather than random access of data frames: avoid collisions of long data frames

- sender first transmits *small* request-to-send (RTS) packets to BS using CSMA
  - RTSs may still collide with each other (but they're short)
- BS broadcasts clear-to-send CTS in response to RTS
- CTS heard by all nodes
  - sender transmits data frame
  - other stations defer transmissions

avoid data frame collisions completely using small reservation packets!

Collision Avoidance: RTS-CTS exchange



Hidden Terminal & Exposed Terminal

802.11 frame: addressing

2	2	6	6	6	2	6	0 - 2312	4
frame control	duration	address 1	address 2	address 3	seq control	address 4	payload	CRC

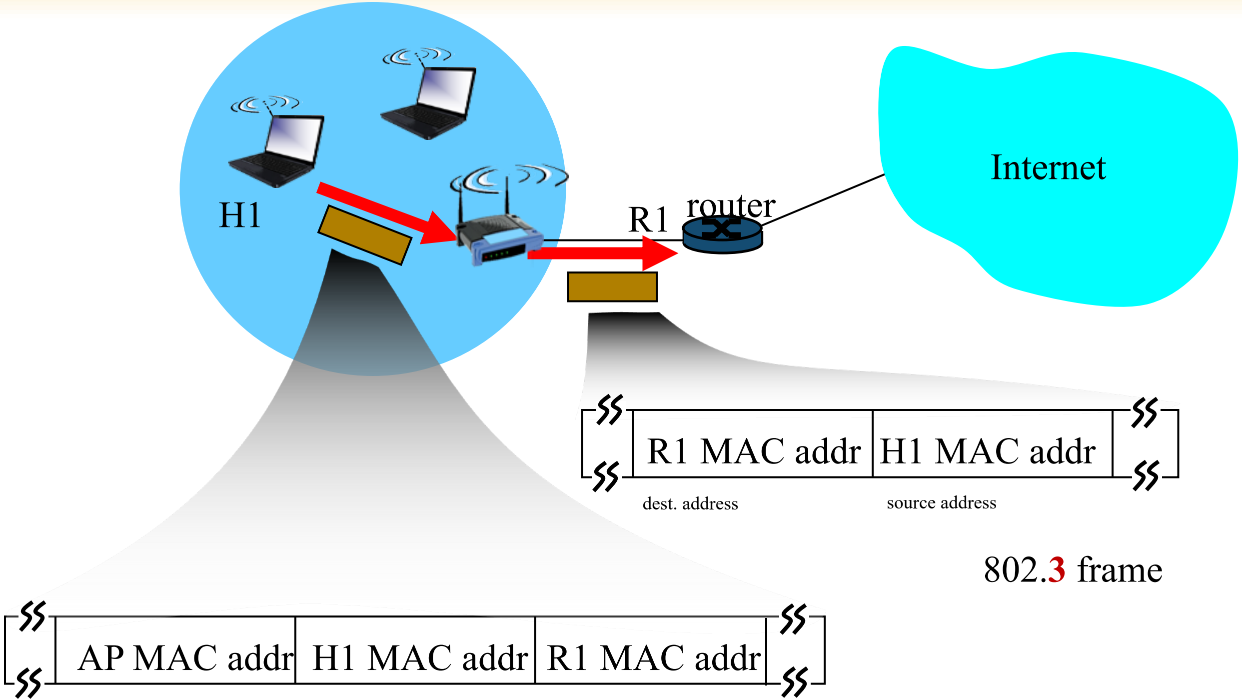
Address 1: MAC address of wireless host or AP to receive this frame

Address 2: MAC address of wireless host or AP transmitting this frame

Address 3: MAC address of router interface to which AP is attached

Address 4: used only in ad hoc mode

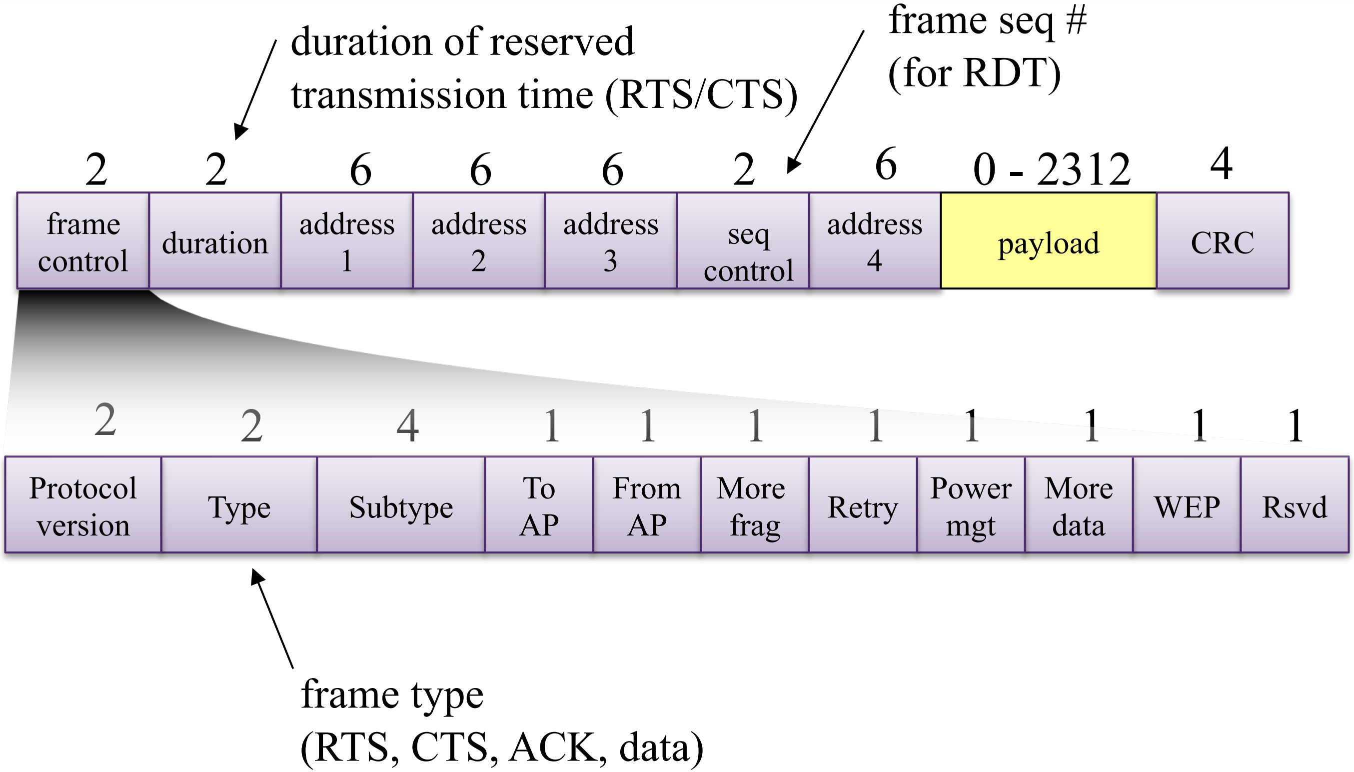
802.11 frame: addressing



802.3 frame

802.11 frame

802.11 frame



duration of reserved transmission time (RTS/CTS)

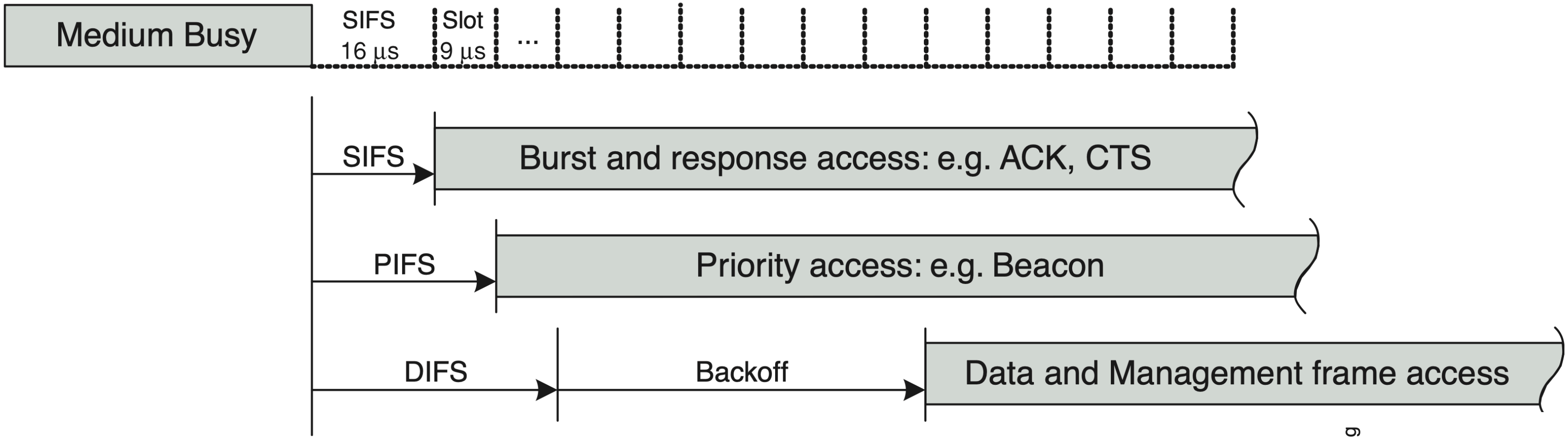
frame seq # (for RDT)

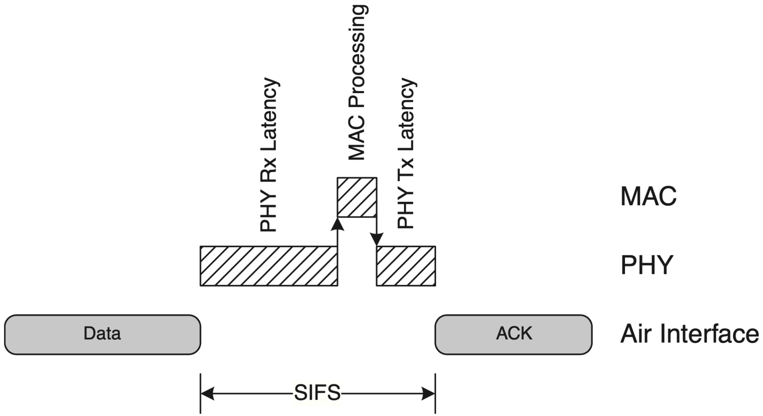
2	2	6	6	6	2	6	0 - 2312	4
frame control	duration	address 1	address 2	address 3	seq control	address 4	payload	CRC

2	2	4	1	1	1	1	1	1	1	1
Protocol version	Type	Subtype	To AP	From AP	More frag	Retry	Power mgt	More data	WEP	Rsvd

frame type (RTS, CTS, ACK, data)

Inter-frame spacing





Slot Time

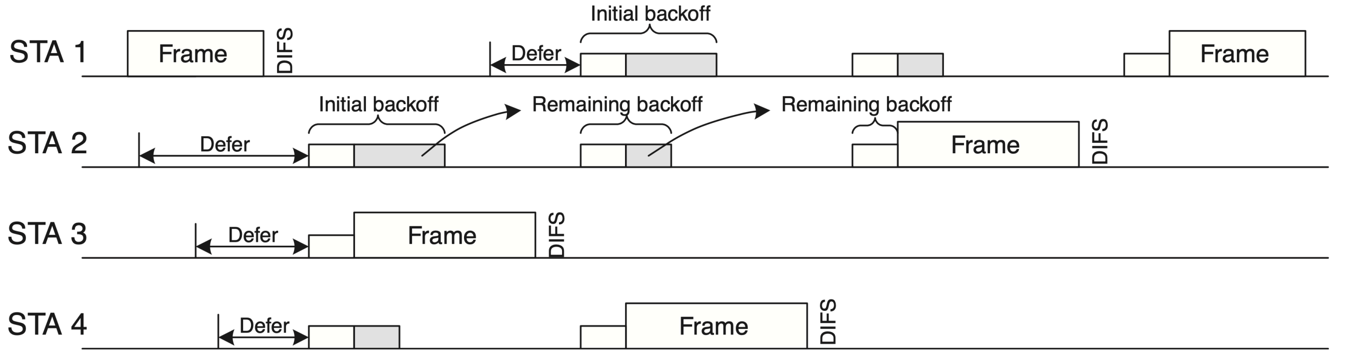
- The slot time for a particular PHY is defined by the aSlotTime parameter.
- For the 802.11a, 802.11g, and 802.11n PHYs the value is 9 µs
- PIFS

$$\text{PIFS} = \text{aSIFSTime} + \text{aSlotTime}$$

- DIFS

$$\text{DIFS} = \text{aSIFSTime} + 2 \times \text{aSlotTime}$$

Random Backoff



- ➢ When the medium transitions from busy to idle, multiple stations may be ready to send data.
- ➢ To minimize collisions, stations select a random backoff count and defer for that number of slot times.
- ➢ The random backoff count is selected as a *pseudo-random integer* drawn from a uniform distribution over the interval [0, CW], where CW, an integer value, is the contention window.

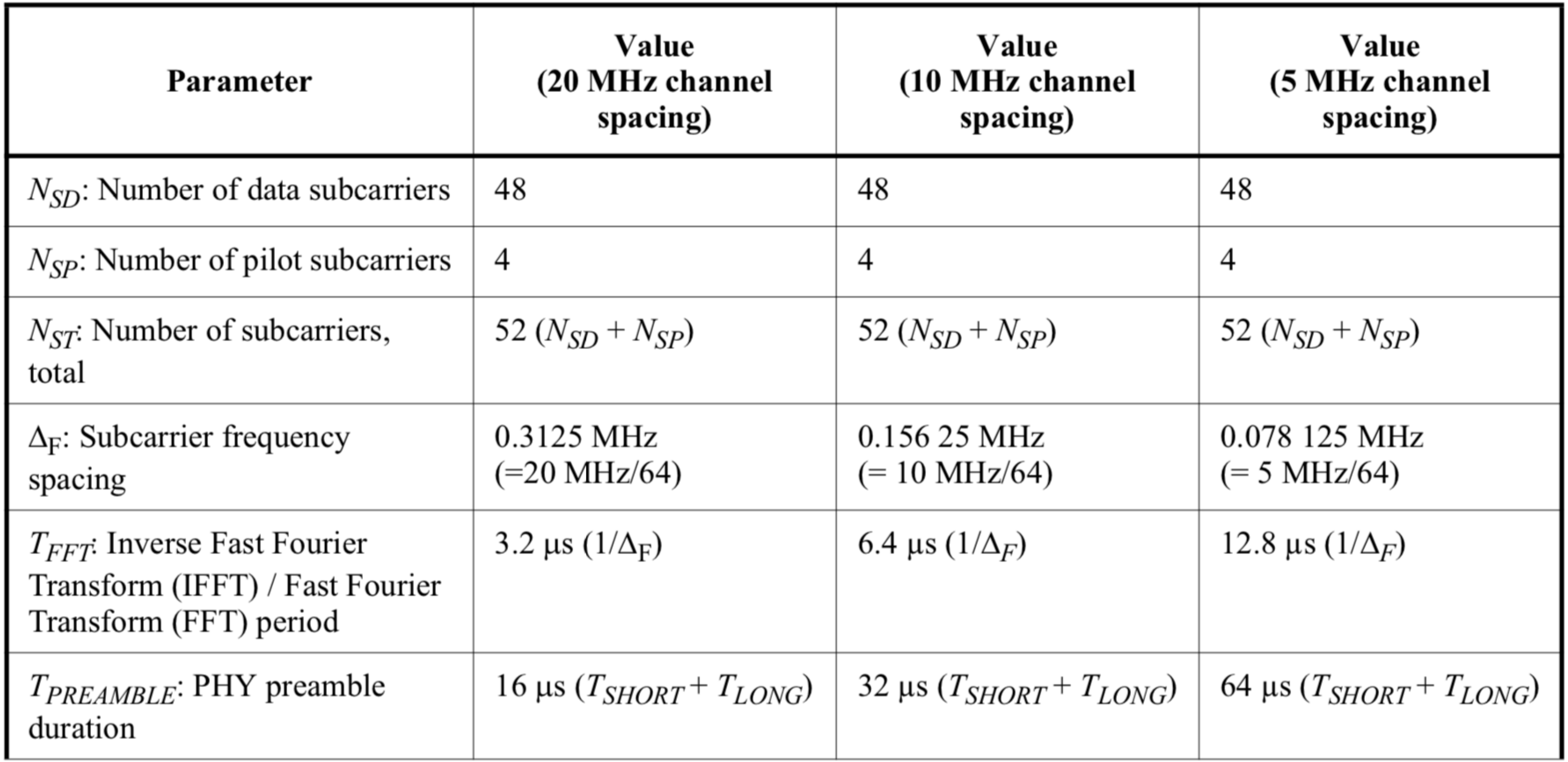
Contention Window

- The contention window (CW) parameter takes the initial value CWmin
- CW doubles on each unsuccessful packet transmit
- If the CW reaches CWmax it remains at that value until it is reset.
- CW is reset to CWmin after every successful packet transmit.

PHY

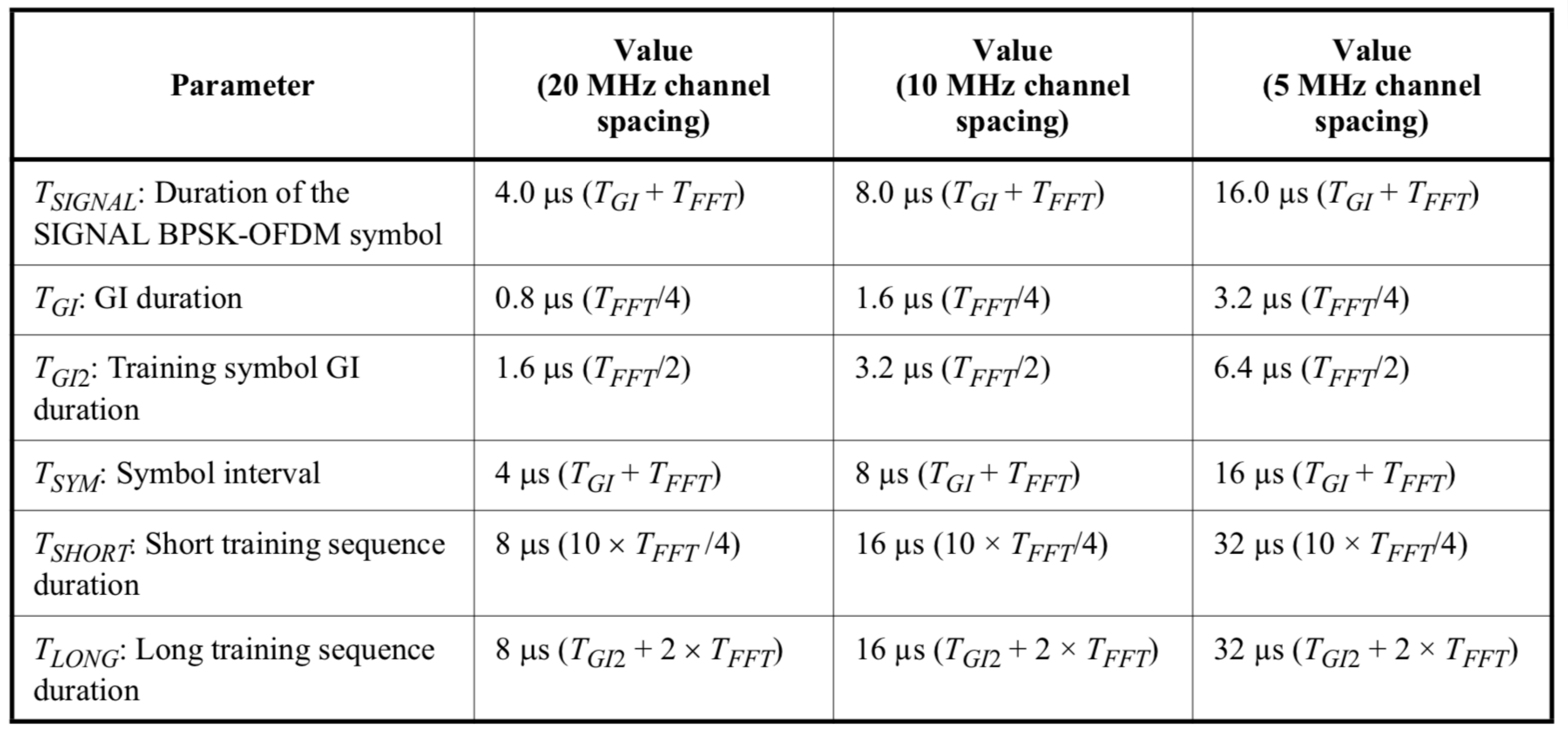
- DSSS (Chapter 15 of standard)
- High rate DSSS (Chapter 16)
- OFDM PHY (Chapter 17)
- Extended Rate PHY (Chapter 18)
- High Throughput PHY (Chapter 19)
- Directional multi-gigabit (DMG) PHY (Chapter 20)
- Very high throughput (VHT) PHY (Chapter 21)
- Television very high throughput (TVHT) PHY (Chapter 22)

OFDM-PHY Timing



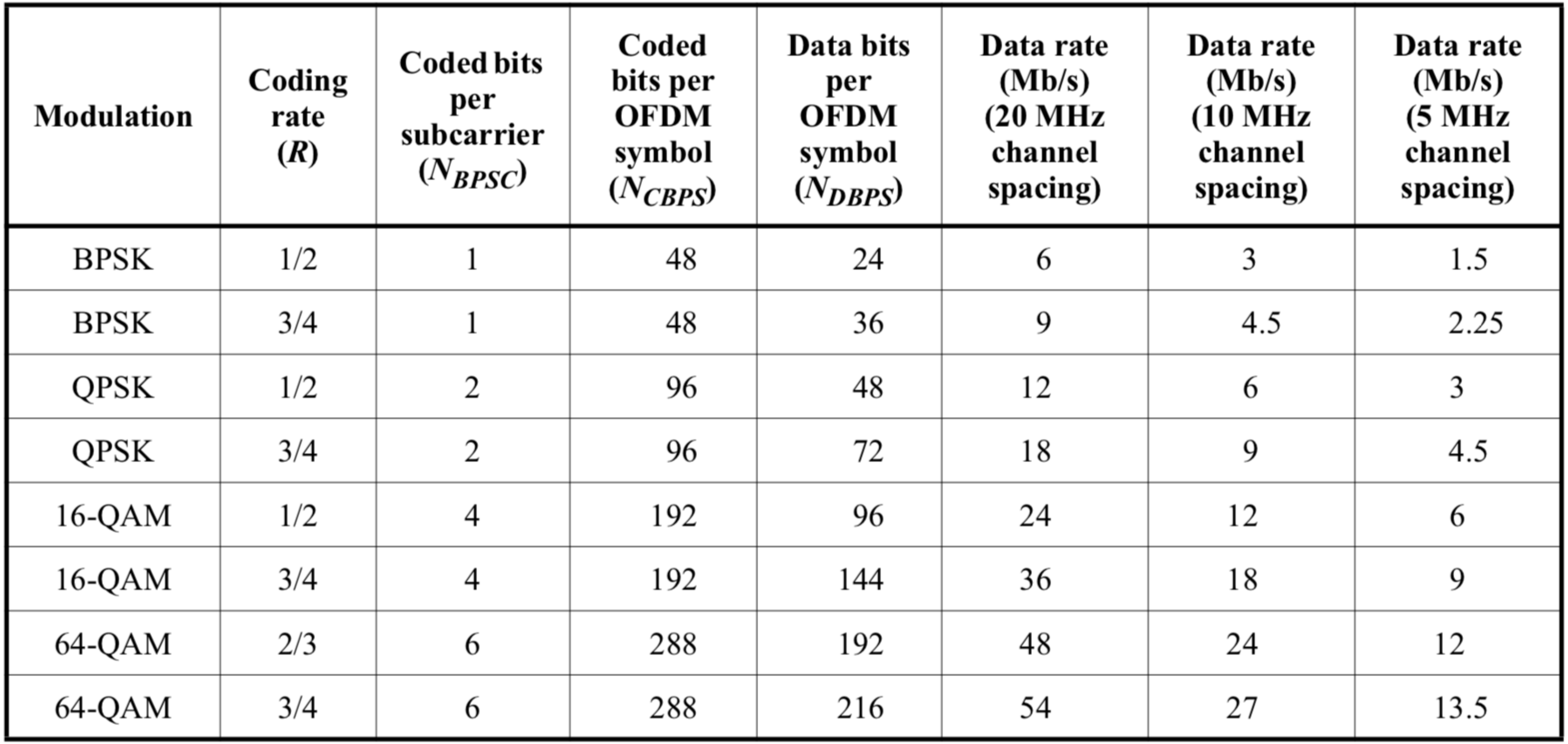
Parameter	**Value (20 MHz channel spacing)**	**Value (10 MHz channel spacing)**	**Value (5 MHz channel spacing)**
N_{SD}: Number of data subcarriers	48	48	48
N_{SP}: Number of pilot subcarriers	4	4	4
N_{ST}: Number of subcarriers, total	52 ($N_{SD} + N_{SP}$)	52 ($N_{SD} + N_{SP}$)	52 ($N_{SD} + N_{SP}$)
Δ_F: Subcarrier frequency spacing	0.3125 MHz (=20 MHz/64)	0.156 25 MHz (= 10 MHz/64)	0.078 125 MHz (= 5 MHz/64)
T_{FFT}: Inverse Fast Fourier Transform (IFFT) / Fast Fourier Transform (FFT) period	3.2 μs ($1/\Delta_F$)	6.4 μs ($1/\Delta_F$)	12.8 μs ($1/\Delta_F$)
$T_{PREAMBLE}$: PHY preamble duration	16 μs ($T_{SHORT} + T_{LONG}$)	32 μs ($T_{SHORT} + T_{LONG}$)	64 μs ($T_{SHORT} + T_{LONG}$)

OFDM-PHY Timing



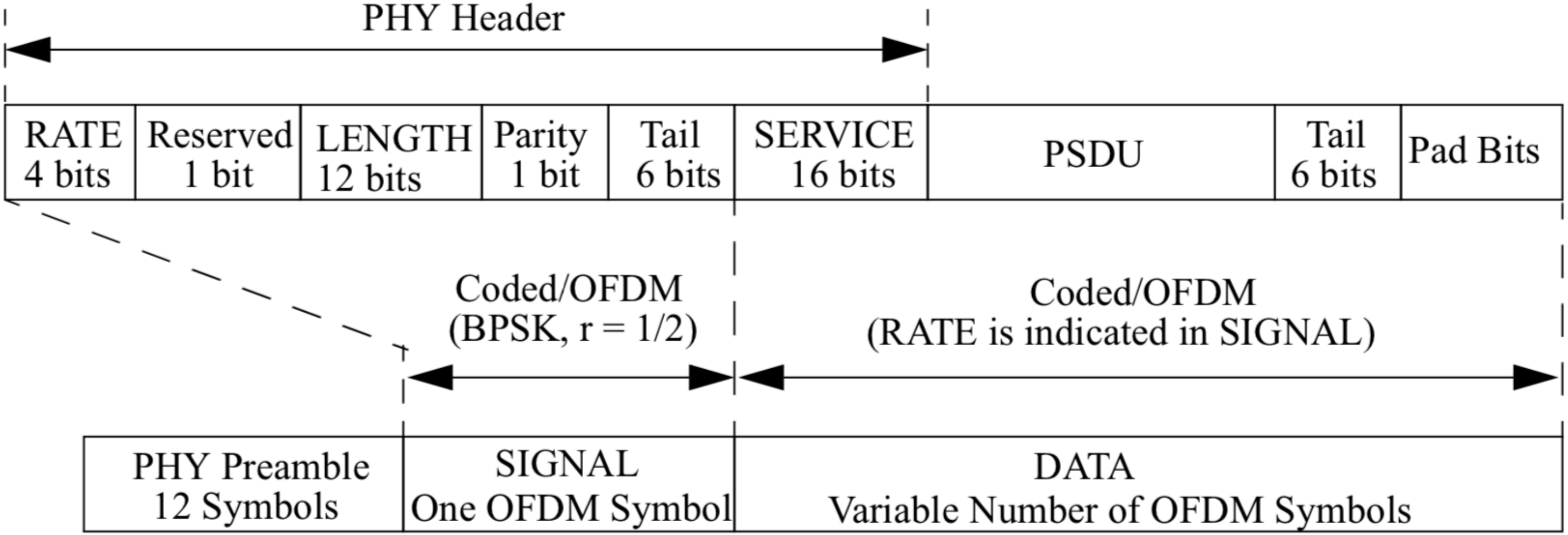
Parameter	Value (20 MHz channel spacing)	Value (10 MHz channel spacing)	Value (5 MHz channel spacing)
T_{SIGNAL}: Duration of the SIGNAL BPSK-OFDM symbol	4.0 µs ($T_{GI} + T_{FFT}$)	8.0 µs ($T_{GI} + T_{FFT}$)	16.0 µs ($T_{GI} + T_{FFT}$)
T_{GI}: GI duration	0.8 µs ($T_{FFT}/4$)	1.6 µs ($T_{FFT}/4$)	3.2 µs ($T_{FFT}/4$)
T_{GI2}: Training symbol GI duration	1.6 µs ($T_{FFT}/2$)	3.2 µs ($T_{FFT}/2$)	6.4 µs ($T_{FFT}/2$)
T_{SYM}: Symbol interval	4 µs ($T_{GI} + T_{FFT}$)	8 µs ($T_{GI} + T_{FFT}$)	16 µs ($T_{GI} + T_{FFT}$)
T_{SHORT}: Short training sequence duration	8 µs ($10 \times T_{FFT}/4$)	16 µs ($10 \times T_{FFT}/4$)	32 µs ($10 \times T_{FFT}/4$)
T_{LONG}: Long training sequence duration	8 µs ($T_{GI2} + 2 \times T_{FFT}$)	16 µs ($T_{GI2} + 2 \times T_{FFT}$)	32 µs ($T_{GI2} + 2 \times T_{FFT}$)

Modulation & Coding



Modulation	Coding rate (R)	Coded bits per subcarrier (N_{BPSC})	Coded bits per OFDM symbol (N_{CBPS})	Data bits per OFDM symbol (N_{DBPS})	Data rate (Mb/s) (20 MHz channel spacing)	Data rate (Mb/s) (10 MHz channel spacing)	Data rate (Mb/s) (5 MHz channel spacing)
BPSK	1/2	1	48	24	6	3	1.5
BPSK	3/4	1	48	36	9	4.5	2.25
QPSK	1/2	2	96	48	12	6	3
QPSK	3/4	2	96	72	18	9	4.5
16-QAM	1/2	4	192	96	24	12	6
16-QAM	3/4	4	192	144	36	18	9
64-QAM	2/3	6	288	192	48	24	12
64-QAM	3/4	6	288	216	54	27	13.5

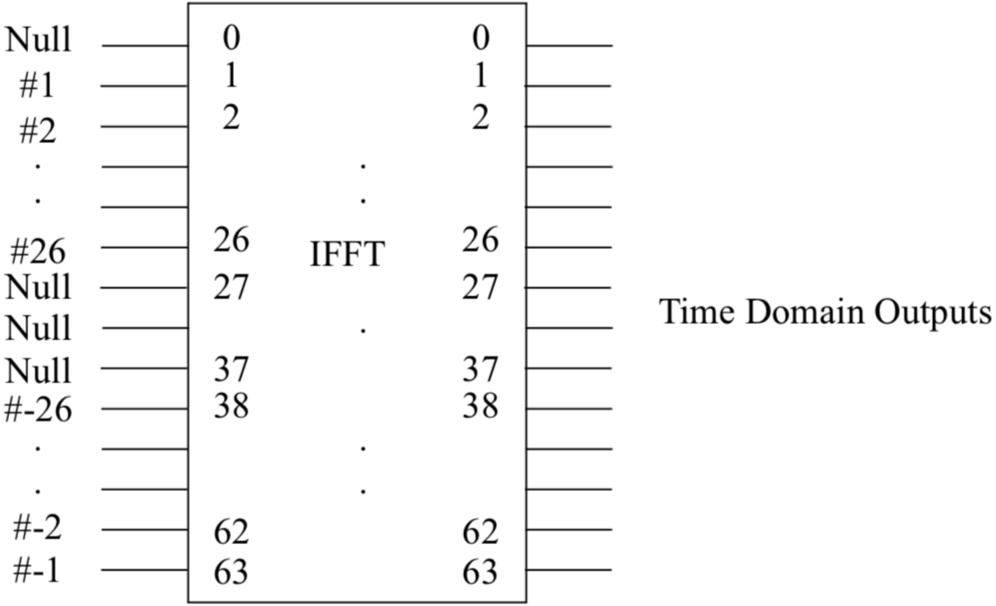
OFDM PHY



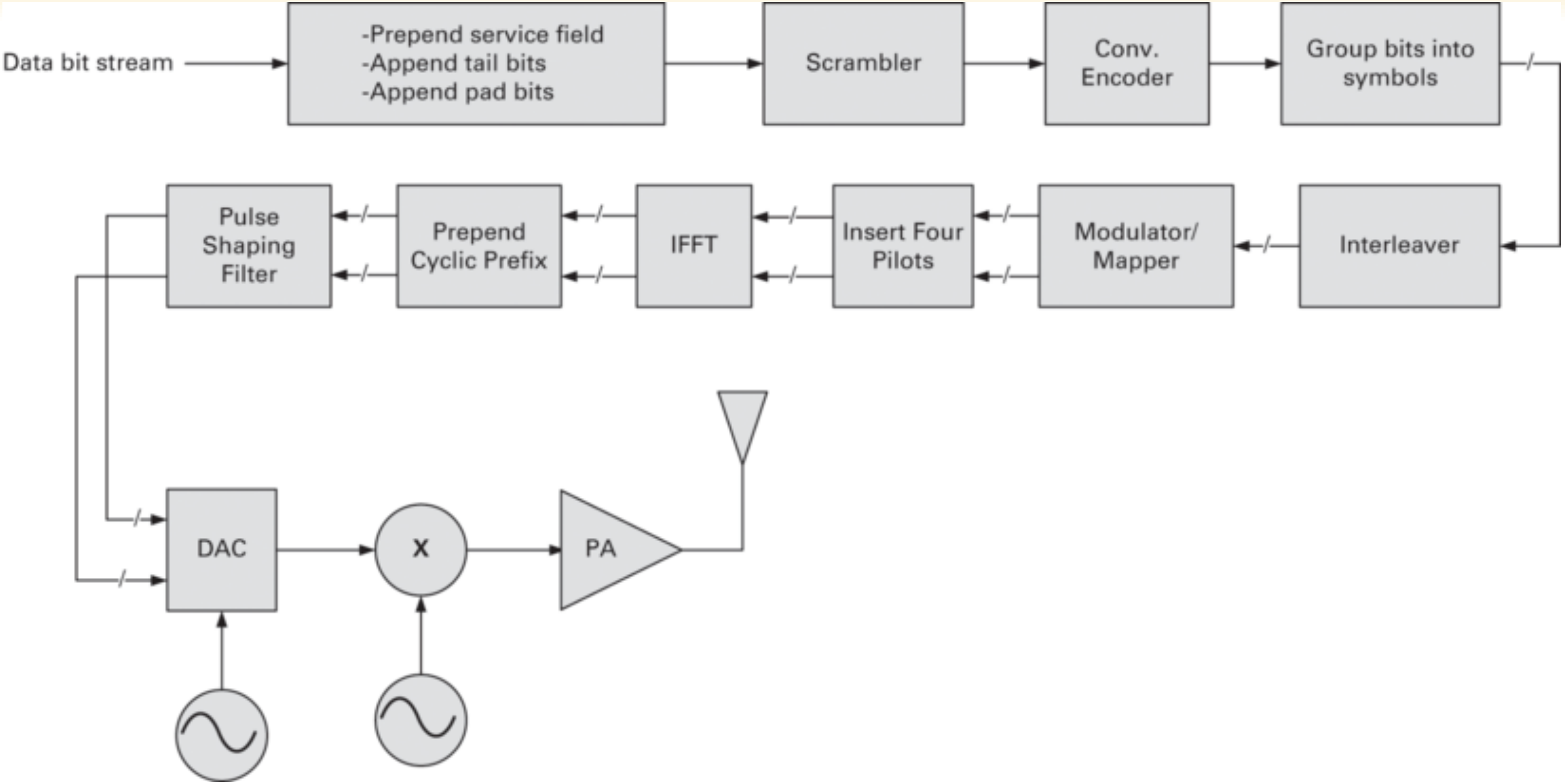
UNIVERSITY AT ALBANY
State University of New York

OFDM Parameters

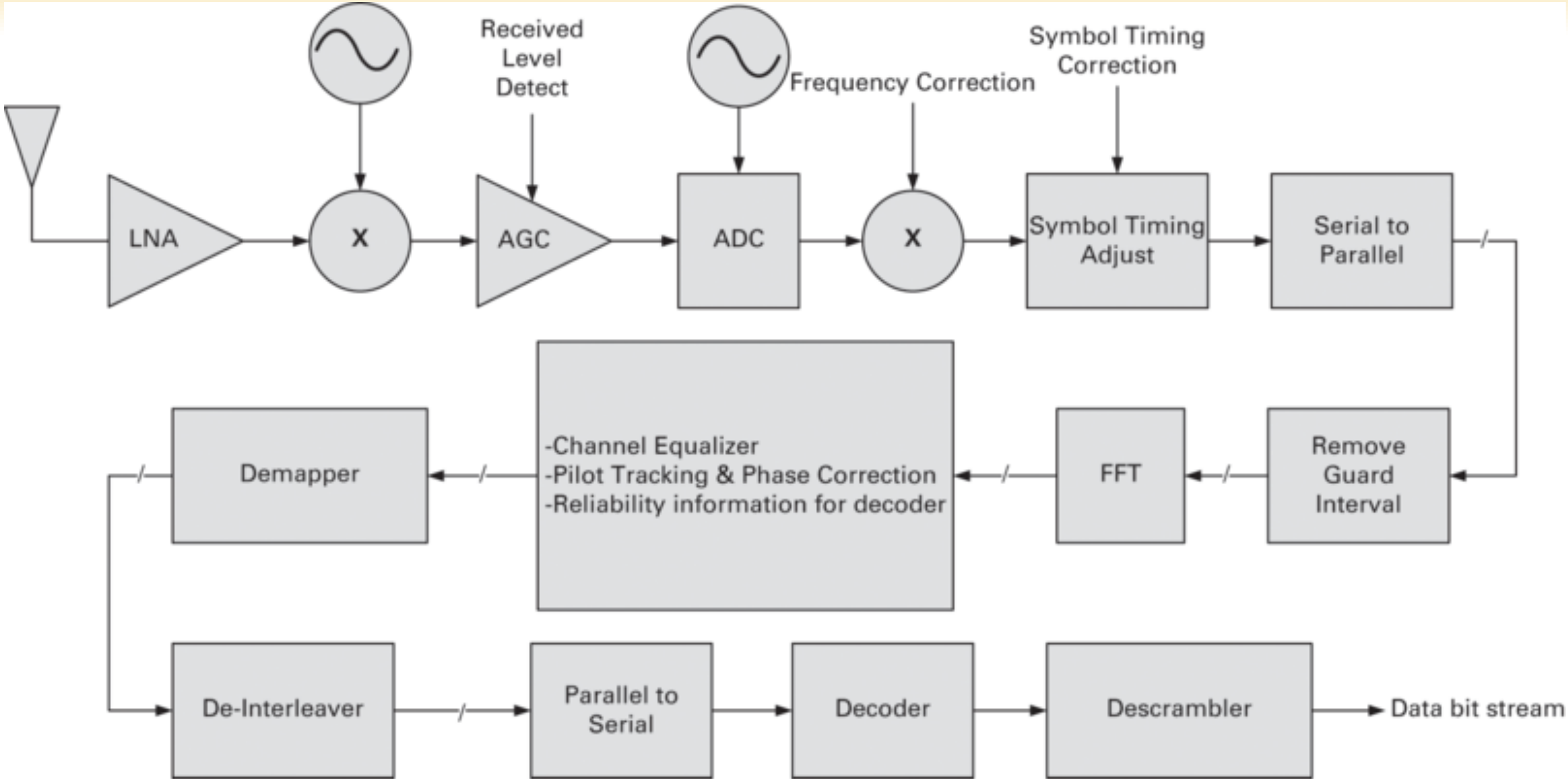
- 64-pt FFT (64 subcarriers)
- FFT convention
- 48 data subcarriers



Transmitter Block for Data



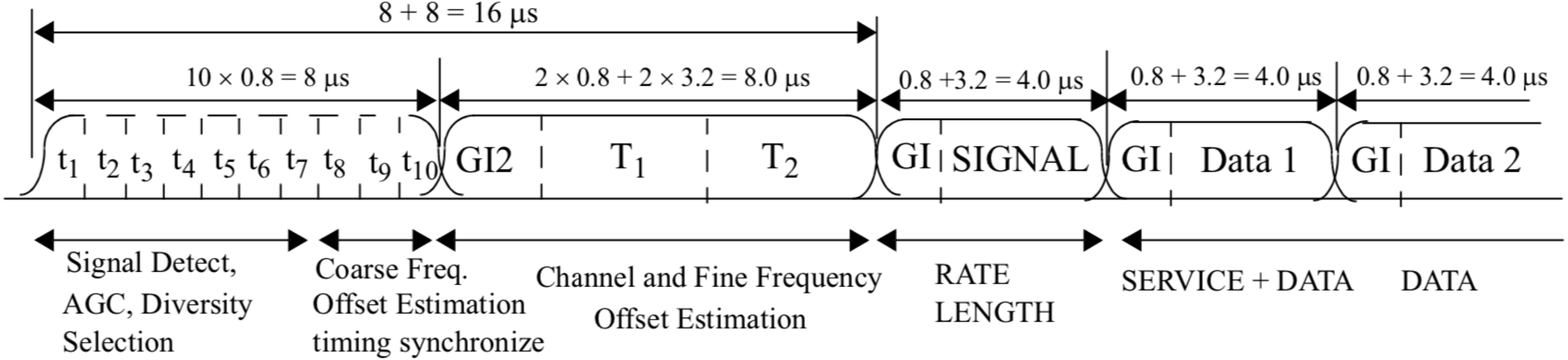
Receiver Block for Data

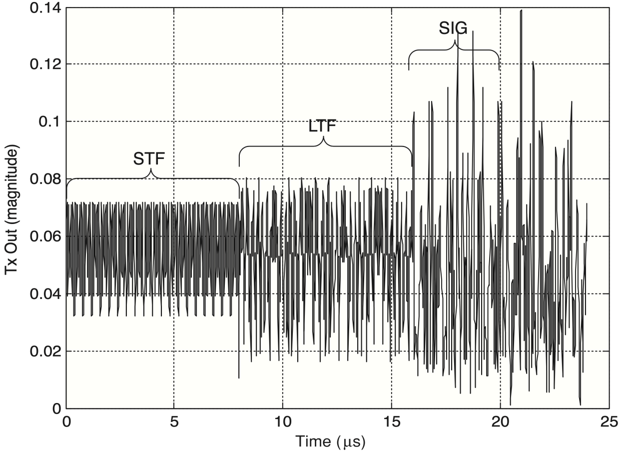


Steps for encoding PPDU

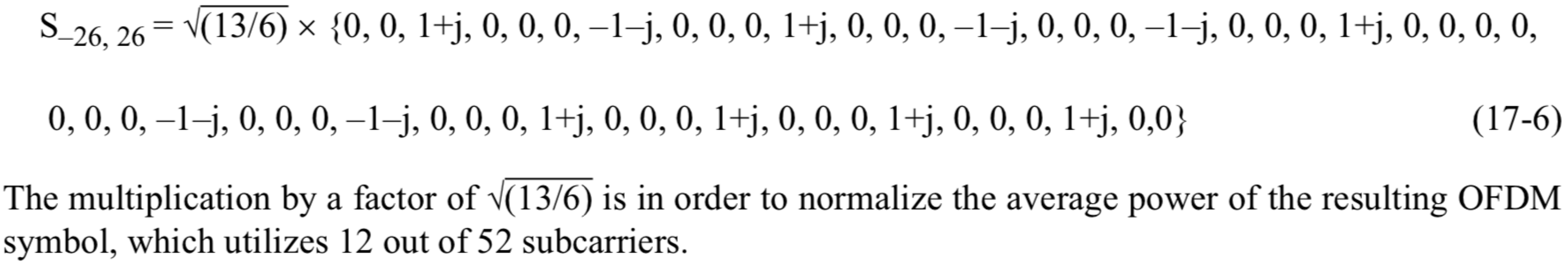
- **17.3.2.2 Overview of the PPDU encoding process**
  - Step 1 - Produce the PHY Preamble field, composed of 10 repetitions of a “short training sequence” (used for AGC convergence, diversity selection, timing acquisition, and coarse frequency acquisition in the receiver) and two repetitions of a “long training sequence” (used for channel estimation and fine frequency acquisition in the receiver), preceded by a guard interval (GI)

Preamble


$8 + 8 = 16\ \mu s$
$10 \times 0.8 = 8\ \mu s$
$2 \times 0.8 + 2 \times 3.2 = 8.0\ \mu s$
$0.8 + 3.2 = 4.0\ \mu s$
$0.8 + 3.2 = 4.0\ \mu s$
$0.8 + 3.2 = 4.0\ \mu s$
t_1 t_2 t_3 t_4 t_5 t_6 t_7 t_8 t_9 t_{10}
GI2
T_1
T_2
GI
SIGNAL
GI
Data 1
GI
Data 2
Signal Detect, AGC, Diversity Selection
Coarse Freq. Offset Estimation timing synchronize
Channel and Fine Frequency Offset Estimation
RATE LENGTH
SERVICE + DATA
DATA


SIG
LTF
STF
Tx Out (magnitude)
Time (µs)
0
0.02
0.04
0.06
0.08
0.1
0.12
0.14
0
5
10
15
20
25

Short Preamble



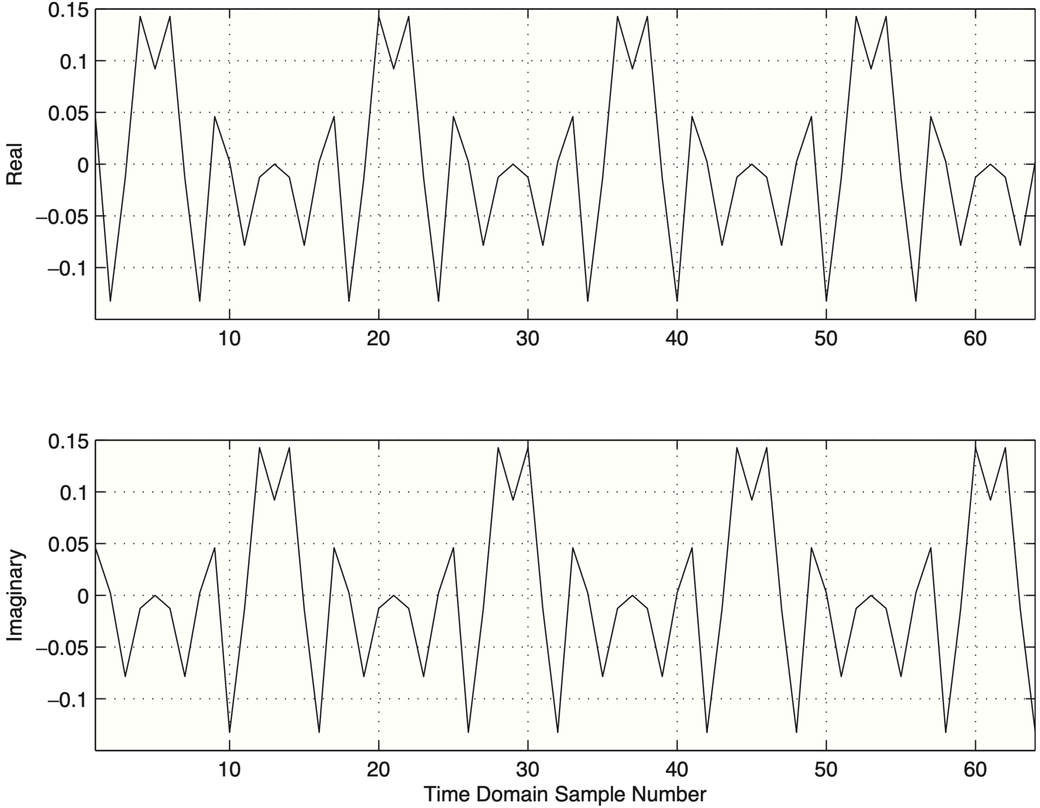
$$S_{-26,26} = \sqrt{(13/6)} \times \{0, 0, 1+j, 0, 0, 0, -1-j, 0, 0, 0, 1+j, 0, 0, 0, -1-j, 0, 0, 0, -1-j, 0, 0, 0, 1+j, 0, 0, 0, 0,$$

$$0, 0, 0, -1-j, 0, 0, 0, -1-j, 0, 0, 0, 1+j, 0, 0, 0, 1+j, 0, 0, 0, 1+j, 0, 0, 0, 1+j, 0,0\} \qquad (17\text{-}6)$$

The multiplication by a factor of $\sqrt{(13/6)}$ is in order to normalize the average power of the resulting OFDM symbol, which utilizes 12 out of 52 subcarriers.

- The sequence was chosen to have good correlation properties and a low peak-to- average power so that its properties are preserved even after clipping or compression by an overloaded analog front end

Short Training Symbol



Long Preamble

$$L_{-26,26} = \{1, 1, -1, -1, 1, 1, -1, 1, -1, 1, 1, 1, 1, 1, 1, -1, -1, 1, 1, -1, 1, -1, 1, 1, 1, 1, 0,$$

$$1, -1, -1, 1, 1, -1, 1, -1, 1, -1, -1, -1, -1, -1, 1, 1, -1, -1, 1, -1, 1, -1, 1, 1, 1, 1\}$$

A long OFDM training symbol shall be generated according to the following equation:

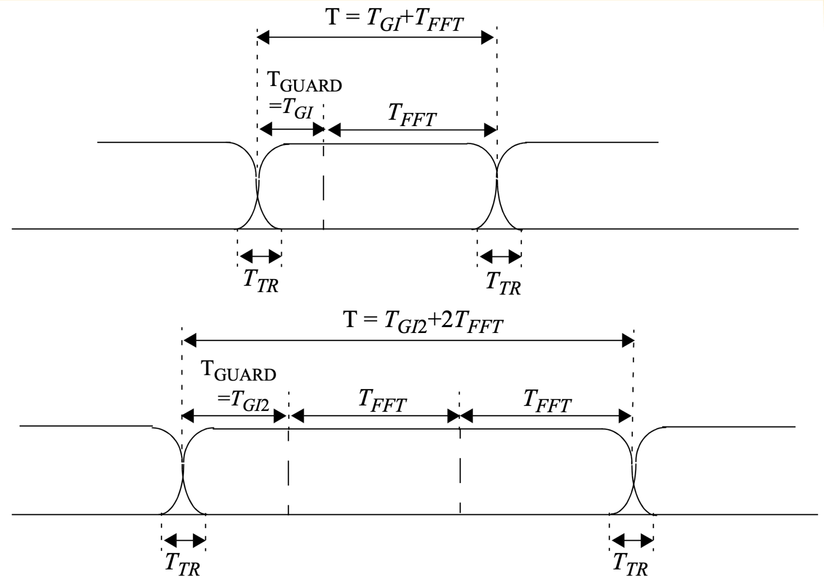
$$r_{LONG}(t) = w_{TLONG}(t) \sum_{k=-N_{ST}/2}^{N_{ST}/2} L_k \exp(j2\pi k\Delta_F(t - T_{G12})) \qquad (17\text{-}9)$$

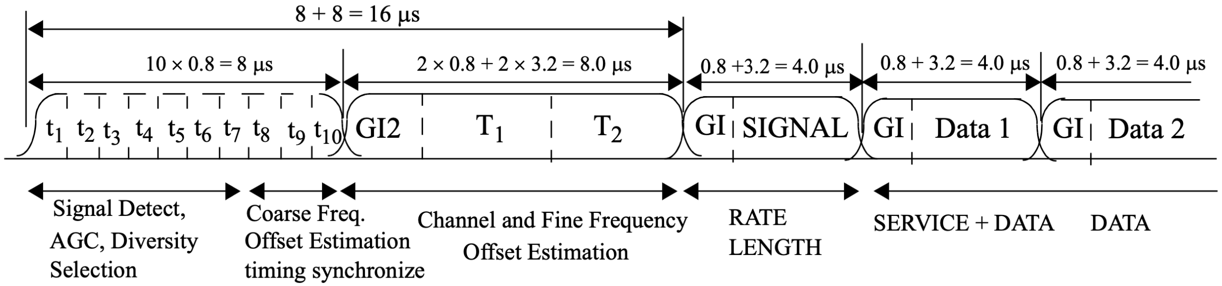
where

$T_{G\,12} = 1.6\ \mu s$

Two periods of the long sequence are transmitted for improved channel estimation accuracy, yielding $T_{LONG} = 1.6 + 2 \times 3.2 = 8\ \mu s$.

Guard


$T = T_{GI} + T_{FFT}$
$T_{GUARD} = T_{GI}$
T_{FFT}
T_{TR}
T_{TR}
$T = T_{GI2} + 2T_{FFT}$
$T_{GUARD} = T_{GI2}$
T_{FFT}
T_{FFT}
T_{TR}
T_{TR}

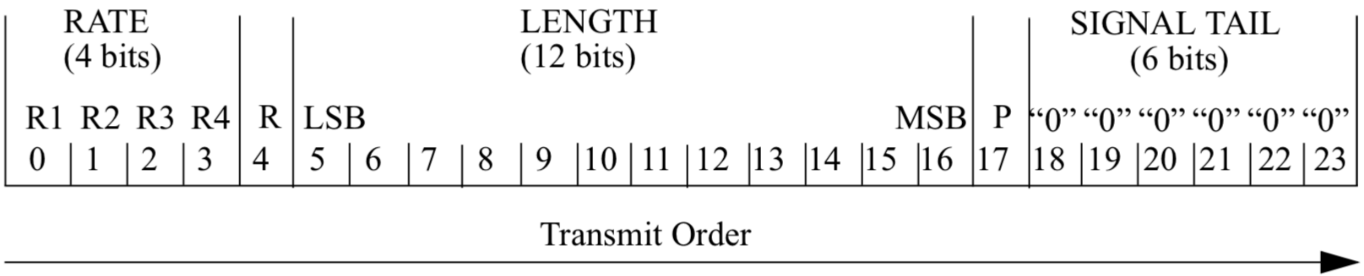

8 + 8 = 16 μs
10 × 0.8 = 8 μs
2 × 0.8 + 2 × 3.2 = 8.0 μs
0.8 +3.2 = 4.0 μs
0.8 + 3.2 = 4.0 μs
0.8 + 3.2 = 4.0 μs
t_1 t_2 t_3 t_4 t_5 t_6 t_7 t_8 t_9 t_{10}
GI2
T_1
T_2
GI
SIGNAL
GI
Data 1
GI
Data 2
Signal Detect, AGC, Diversity Selection
Coarse Freq. Offset Estimation timing synchronize
Channel and Fine Frequency Offset Estimation
RATE LENGTH
SERVICE + DATA
DATA

Steps for encoding PPDU

17.3.2.2 Overview of the PPDU encoding process

- Step 2 - Produce the PHY header field from the RATE, LENGTH, and SERVICE fields of the TXVECTOR by filling the appropriate bit fields. The RATE and LENGTH fields of the PHY header are encoded by a convolutional code at a rate of R = 1/2, and are subsequently mapped onto a single BPSK encoded OFDM symbol, denoted as the SIGNAL symbol. In order to facilitate a reliable and timely detection of the RATE and LENGTH fields, 6 zero tail bits are inserted into the PHY header. The encoding of the SIGNAL field into an OFDM symbol follows the same steps for convolutional encoding, interleaving, BPSK modulation, pilot insertion, Fourier transform, and prepending a GI as described subsequently for data transmission with BPSK-OFDM modulated at coding rate 1/2. *The contents of the SIGNAL field are not scrambled.*

SIGNAL Field



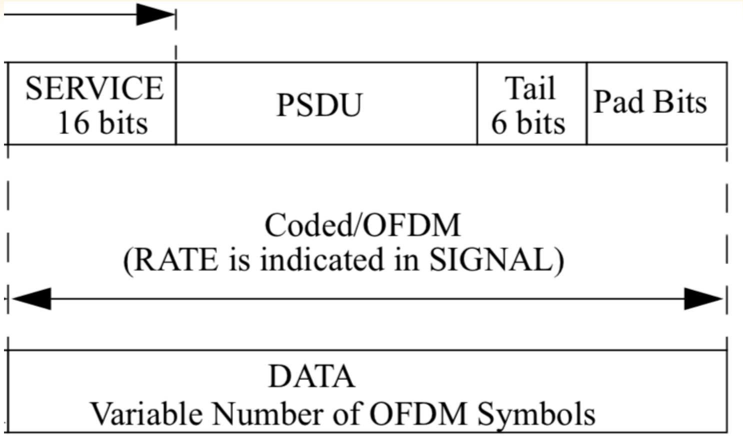
RATE (4 bits)					LENGTH (12 bits)													SIGNAL TAIL (6 bits)					
R1	R2	R3	R4	R	LSB											MSB	P	"0"	"0"	"0"	"0"	"0"	"0"
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23

Transmit Order

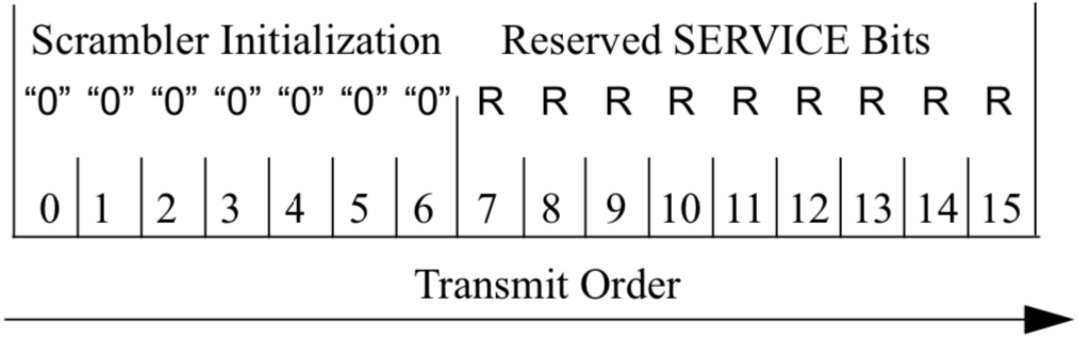
- Bit 4 is reserved. It shall be set to 0 on transmit and ignored on receive.
- Bit 17 shall be a positive parity (even parity) bit for bits 0–16.
- The bits 18–23 constitute the SIGNAL TAIL field, and all 6 bits shall be set to 0.

R1–R4	**Rate (Mb/s) (20 MHz channel spacing)**	**Rate (Mb/s) (10 MHz channel spacing)**	**Rate (Mb/s) (5 MHz channel spacing)**
1101	6	3	1.5
1111	9	4.5	2.25
0101	12	6	3
0111	18	9	4.5
1001	24	12	6
1011	36	18	9
0001	48	24	12
0011	54	27	13.5

SERVICE Field



- The DATA field contains the SERVICE field, the PSDU, the TAIL bits, and the PAD bits, if needed



Scrambler Initialization							Reserved SERVICE Bits								
“0”	“0”	“0”	“0”	“0”	“0”	“0”	R	R	R	R	R	R	R	R	R
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

Transmit Order

R: Reserved

PPDU Tail Bits

- The PPDU TAIL field shall be six bits of 0, which are required to return the convolutional encoder to the zero state.
- This procedure *improves the error probability of the convolutional decoder*, which relies on future bits when decoding and which may not be available past the end of the message.
- The PPDU TAIL field shall be produced by replacing six scrambled zero bits following the message end with six nonscrambled zero bits.

Pad Bits

- The number of bits in the DATA field shall be a multiple of N_{CBPS}, the number of coded bits in an OFDM symbol (48, 96, 192, or 288 bits).
- To achieve that, the length of the message is extended so that it becomes a multiple of N_{DBPS}, the number of data bits per OFDM symbol.
- 6 bits are appended to the message, in order to accommodate the TAIL bits

$$N_{SYM} = \left\lceil \frac{16 + 8 \times \text{LENGTH} + 6}{N_{DBPS}} \right\rceil$$

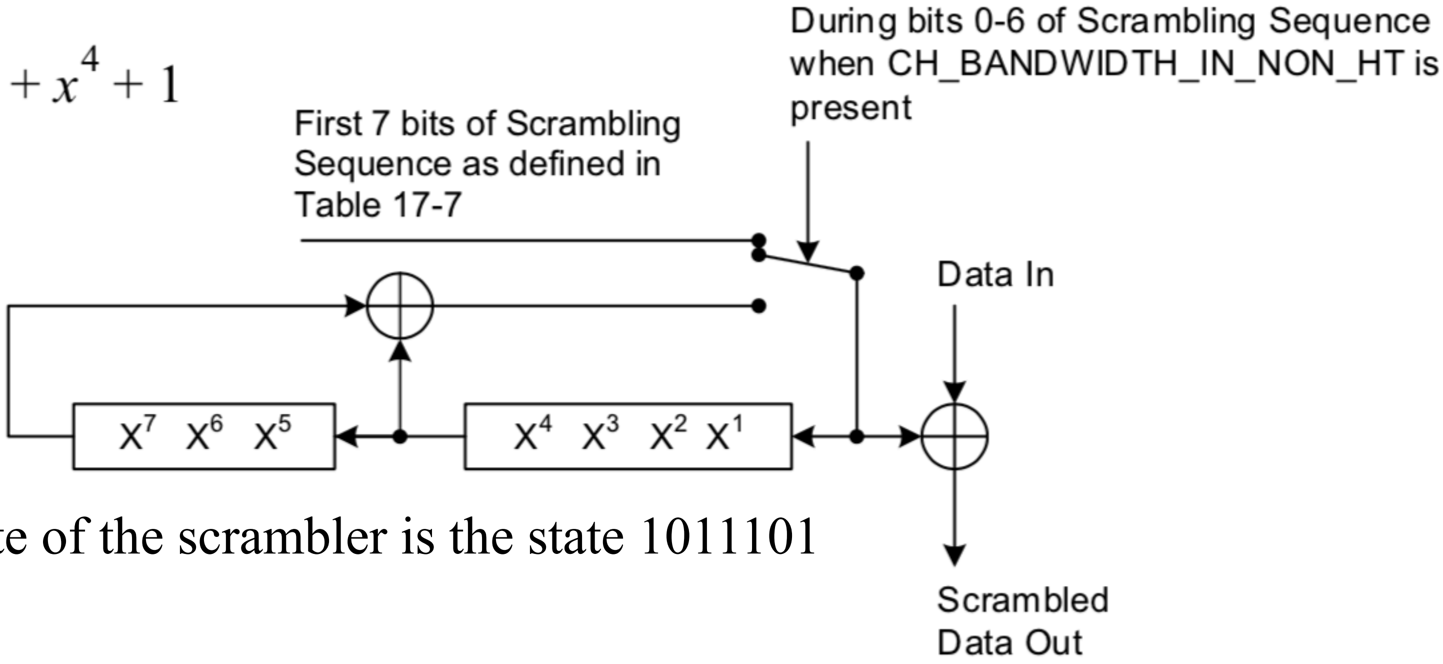
$$N_{DATA} = N_{SYM} \times N_{DBPS}$$

$$N_{PAD} = N_{DATA} - (16 + 8 \times \text{LENGTH} + 6)$$

Scrambler & Descrambler

- The scrambling step at the transmitter side is to avoid long consecutive sequences of 0s or 1s

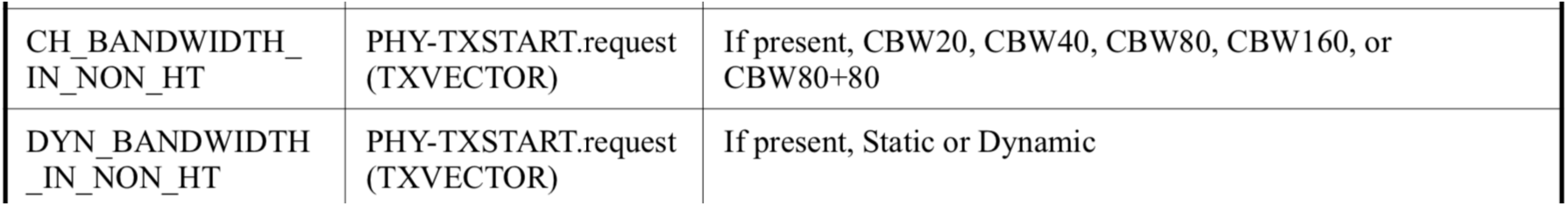
$$S(x) = x^7 + x^4 + 1$$



The initial state of the scrambler is the state 1011101

Scrambler

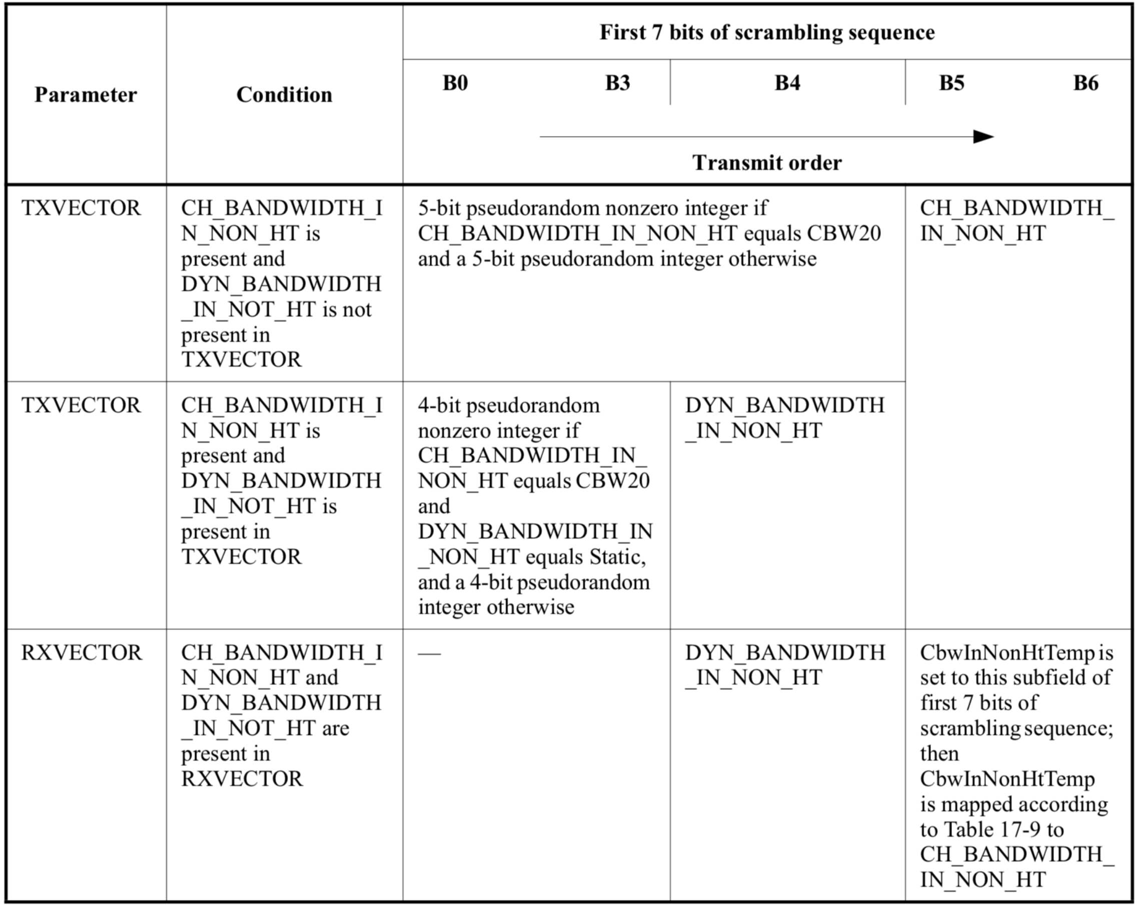
- The same scrambler is used to scramble transmit data and to descramble receive data.
- Scrambler depends on TXVECTOR
- If the TXVECTOR parameter CH_BANDWIDTH_IN_NON_HT is not present, when transmitting, the initial state of the scrambler shall be set to a pseudorandom nonzero state.



CH_BANDWIDTH_IN_NON_HT	PHY-TXSTART.request (TXVECTOR)	If present, CBW20, CBW40, CBW80, CBW160, or CBW80+80
DYN_BANDWIDTH_IN_NON_HT	PHY-TXSTART.request (TXVECTOR)	If present, Static or Dynamic

Scrambler Initialization

- The seven LSBs of the SERVICE field shall be set to all 0s prior to scrambling to enable estimation of the initial state of the scrambler in the receiver.



Parameter	Condition	First 7 bits of scrambling sequence		
		B0 B3	B4	B5 B6
		Transmit order		
TXVECTOR	CH_BANDWIDTH_IN_NON_HT is present and DYN_BANDWIDTH_IN_NOT_HT is not present in TXVECTOR	5-bit pseudorandom nonzero integer if CH_BANDWIDTH_IN_NON_HT equals CBW20 and a 5-bit pseudorandom integer otherwise		CH_BANDWIDTH_IN_NON_HT
TXVECTOR	CH_BANDWIDTH_IN_NON_HT is present and DYN_BANDWIDTH_IN_NOT_HT is present in TXVECTOR	4-bit pseudorandom nonzero integer if CH_BANDWIDTH_IN_NON_HT equals CBW20 and DYN_BANDWIDTH_IN_NON_HT equals Static, and a 4-bit pseudorandom integer otherwise	DYN_BANDWIDTH_IN_NON_HT	
RXVECTOR	CH_BANDWIDTH_IN_NON_HT and DYN_BANDWIDTH_IN_NOT_HT are present in RXVECTOR	—	DYN_BANDWIDTH_IN_NON_HT	CbwInNonHtTemp is set to this subfield of first 7 bits of scrambling sequence; then CbwInNonHtTemp is mapped according to Table 17-9 to CH_BANDWIDTH_IN_NON_HT

Convolutional Encoder

- The DATA field, composed of SERVICE, PSDU, tail, and pad parts, shall be coded with a convolutional encoder of coding rate R = 1/2, 2/3, or 3/4, corresponding to the TXVECTOR parameter RATE.

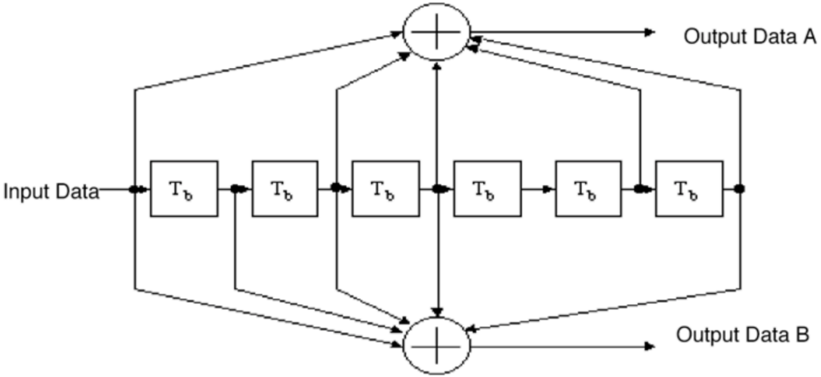


Figure 17-8—Convolutional encoder (k = 7)

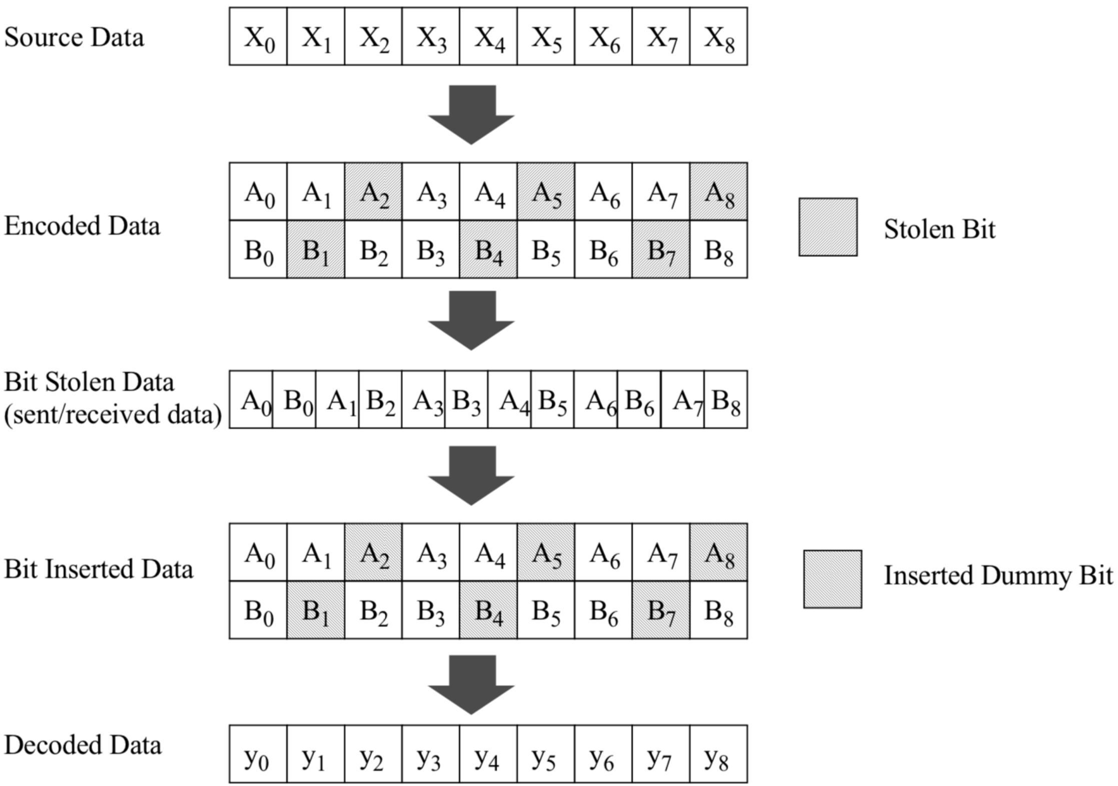
The convolutional encoder shall use the industry-standard generator polynomials, $g_0 = 133_8$ and $g_1 = 171_8$, of rate R = 1/2

MATLAB: poly2trellis(7, [133 171]);

https://www.mathworks.com/help/comm/ref/poly2trellis.html

Puncturing (to yield higher data rate)

Punctured Coding (r = 3/4)



Source Data

X_0	X_1	X_2	X_3	X_4	X_5	X_6	X_7	X_8

Encoded Data

A_0	A_1	A_2	A_3	A_4	A_5	A_6	A_7	A_8
B_0	B_1	B_2	B_3	B_4	B_5	B_6	B_7	B_8

▨ Stolen Bit

Bit Stolen Data
(sent/received data)

A_0	B_0	A_1	B_2	A_3	B_3	A_4	B_5	A_6	B_6	A_7	B_8

Bit Inserted Data

A_0	A_1	A_2	A_3	A_4	A_5	A_6	A_7	A_8
B_0	B_1	B_2	B_3	B_4	B_5	B_6	B_7	B_8

▨ Inserted Dummy Bit

Decoded Data

y_0	y_1	y_2	y_3	y_4	y_5	y_6	y_7	y_8

Data Interleaving

- To avoid burst error
- block interleaver with a block size corresponding to the number of bits in a single OFDM symbol, N_{CBPS}
- two-step permutation

The first permutation is defined by the rule

$$i = (N_{CBPS}/16) \times (k \bmod 16) + \lfloor k/16 \rfloor \quad k = 0,1,\ldots,N_{CBPS}-1$$

adjacent coded bits to be mapped onto nonadjacent subcarriers

The second permutation is defined by the rule

$$j = s \times \left\lfloor \frac{i}{s} \right\rfloor + \left(i + N_{CBPS} - \left\lfloor \frac{16 \times i}{N_{CBPS}} \right\rfloor \right) \bmod s \quad i = 0,1,\ldots\ N_{CBPS}-1$$

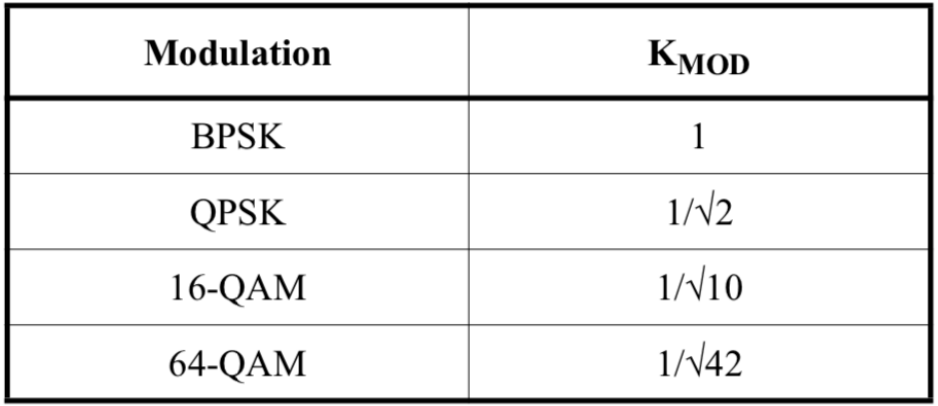
The value of s is determined by the number of coded bits per subcarrier,

$$s = \max(N_{BPSC}/2,1)$$

adjacent coded bits to be mapped alternately onto less and more significant bits of the constellation and, thereby, long runs of low reliability (LSB) bits are avoided

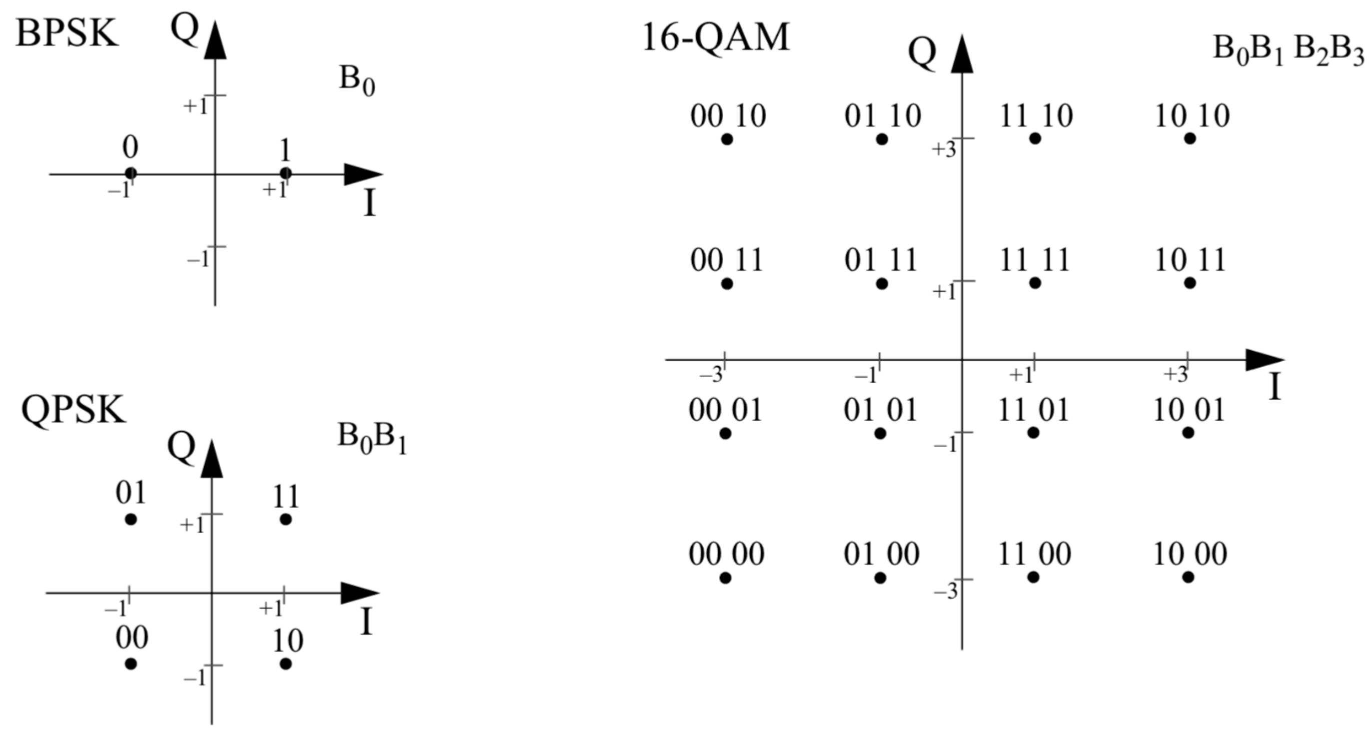
Subcarrier Modulation Mapping

$$d = (I + jQ) \times K_{MOD}$$

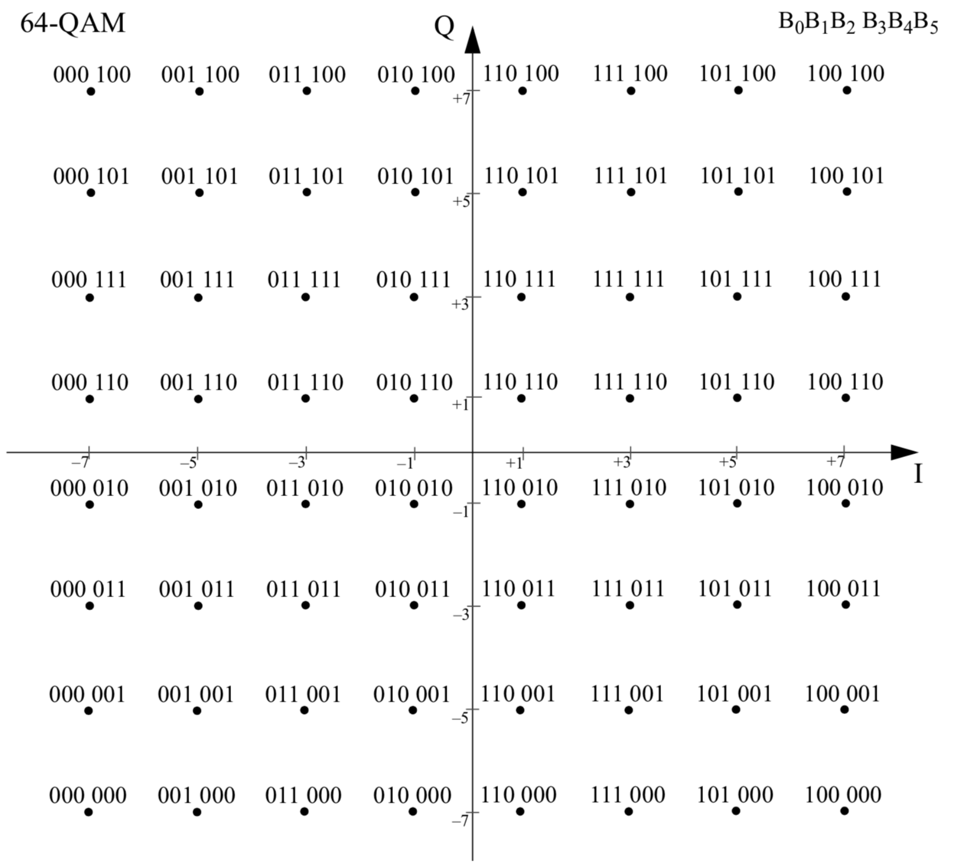


Modulation	K_{MOD}
BPSK	1
QPSK	$1/\sqrt{2}$
16-QAM	$1/\sqrt{10}$
64-QAM	$1/\sqrt{42}$

Gray Coded Mapping



Gray Coded Mapping



	−7	−5	−3	−1	+1	+3	+5	+7
+7	000 100	001 100	011 100	010 100	110 100	111 100	101 100	100 100
+5	000 101	001 101	011 101	010 101	110 101	111 101	101 101	100 101
+3	000 111	001 111	011 111	010 111	110 111	111 111	101 111	100 111
+1	000 110	001 110	011 110	010 110	110 110	111 110	101 110	100 110
−1	000 010	001 010	011 010	010 010	110 010	111 010	101 010	100 010
−3	000 011	001 011	011 011	010 011	110 011	111 011	101 011	100 011
−5	000 001	001 001	011 001	010 001	110 001	111 001	101 001	100 001
−7	000 000	001 000	011 000	010 000	110 000	111 000	101 000	100 000

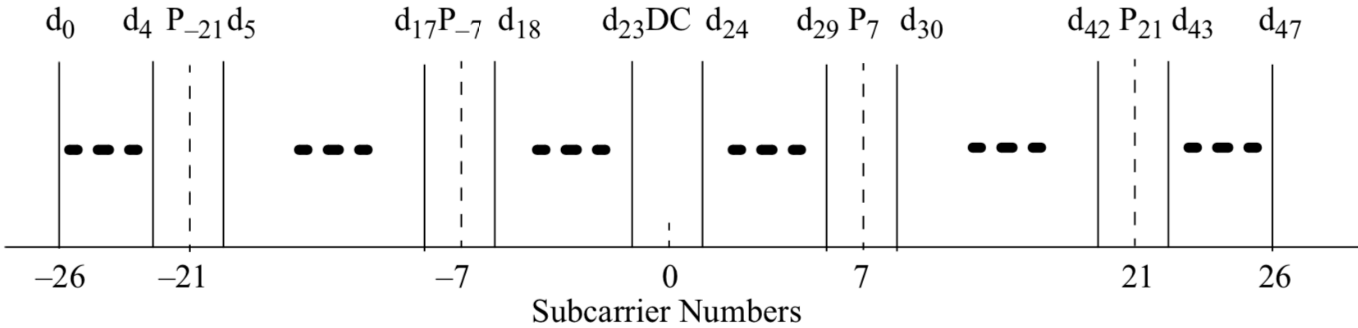
Pilot Subcarriers

- Pilot signals are in –21, –7, 7, 21
- The pilots shall be BPSK modulated by a pseudo-binary sequence to prevent the generation of spectral lines.

$$P_{-26,26} = \{0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0,$$

$$0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, -1, 0, 0, 0, 0, 0\}$$

To avoid difficulties in D/A and A/D converter offsets and carrier feedthrough in the RF system, the subcarrier falling at DC (0th subcarrier) is not used.



Polarity of Pilot Subcarrier

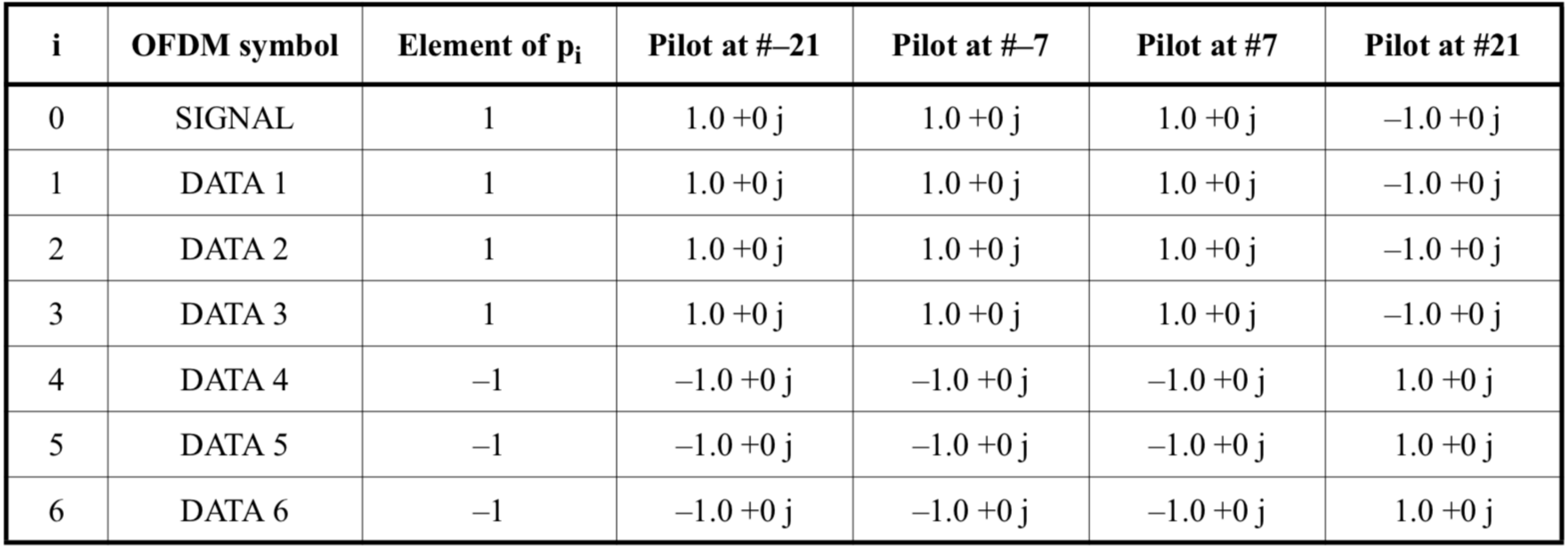
- Controlled by the sequence, p_n
- The sequence p_n is generated by the scrambler

$$S(x) = x^7 + x^4 + 1$$

- Scrambler initial state all 1s

$$\begin{aligned} p_{0..126v} = \{&1,1,1,1,\ -1,-1,-1,1,\ -1,-1,-1,-1,\ 1,1,-1,1,\ -1,-1,1,1,\ -1,1,1,-1,\ 1,1,1,1,\ 1,1,-1,1,\\ &1,1,-1,1,\ 1,-1,-1,1,\ 1,1,-1,1,\ -1,-1,-1,1,\ -1,1,-1,-1,\ 1,-1,-1,1,\ 1,1,1,1,\ -1,-1,1,1,\\ &-1,-1,1,-1,\ 1,-1,1,1,\ -1,-1,-1,1,\ 1,-1,-1,-1,\ -1,1,-1,-1,\ 1,-1,1,1,\ 1,1,-1,1,\ -1,1,-1,1,\\ &-1,-1,-1,-1,\ -1,1,-1,1,\ 1,-1,1,-1,\ 1,1,1,-1,\ -1,1,-1,-1,\ -1,1,1,1,\ -1,-1,-1,-1,\ -1,-1,-1\} \end{aligned}$$

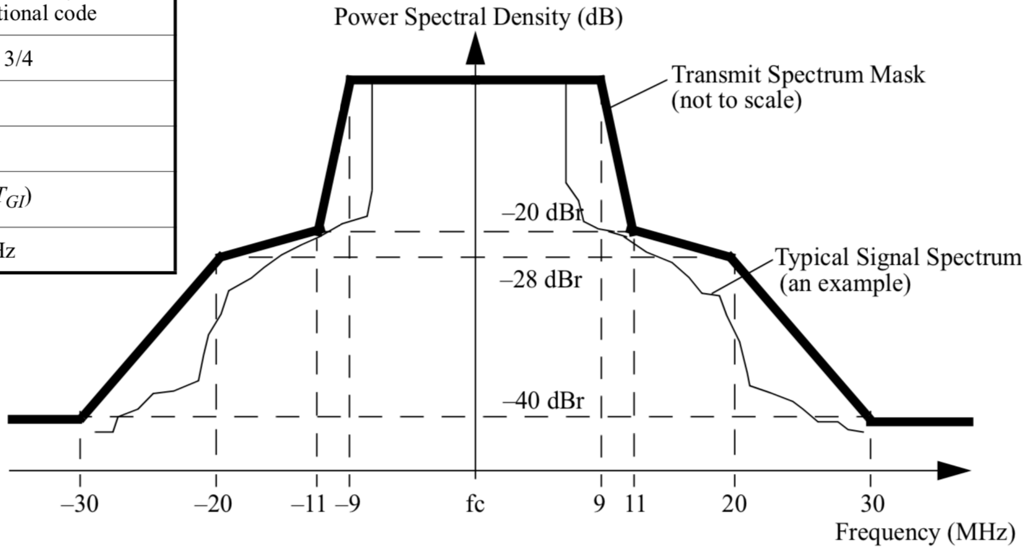
Pilot Generation



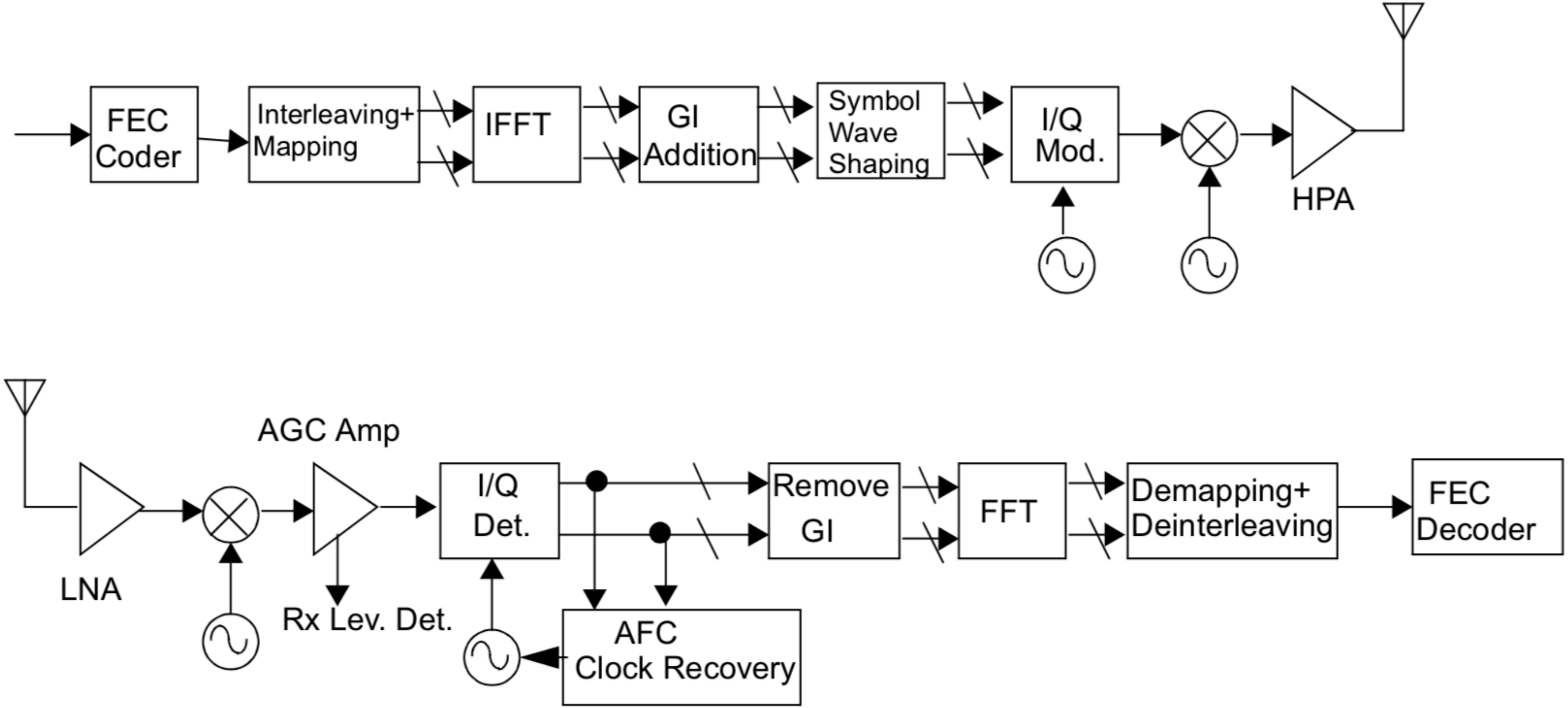
i	OFDM symbol	Element of p_i	Pilot at #–21	Pilot at #–7	Pilot at #7	Pilot at #21
0	SIGNAL	1	1.0 +0 j	1.0 +0 j	1.0 +0 j	–1.0 +0 j
1	DATA 1	1	1.0 +0 j	1.0 +0 j	1.0 +0 j	–1.0 +0 j
2	DATA 2	1	1.0 +0 j	1.0 +0 j	1.0 +0 j	–1.0 +0 j
3	DATA 3	1	1.0 +0 j	1.0 +0 j	1.0 +0 j	–1.0 +0 j
4	DATA 4	–1	–1.0 +0 j	–1.0 +0 j	–1.0 +0 j	1.0 +0 j
5	DATA 5	–1	–1.0 +0 j	–1.0 +0 j	–1.0 +0 j	1.0 +0 j
6	DATA 6	–1	–1.0 +0 j	–1.0 +0 j	–1.0 +0 j	1.0 +0 j

Transmit Spectral Mask

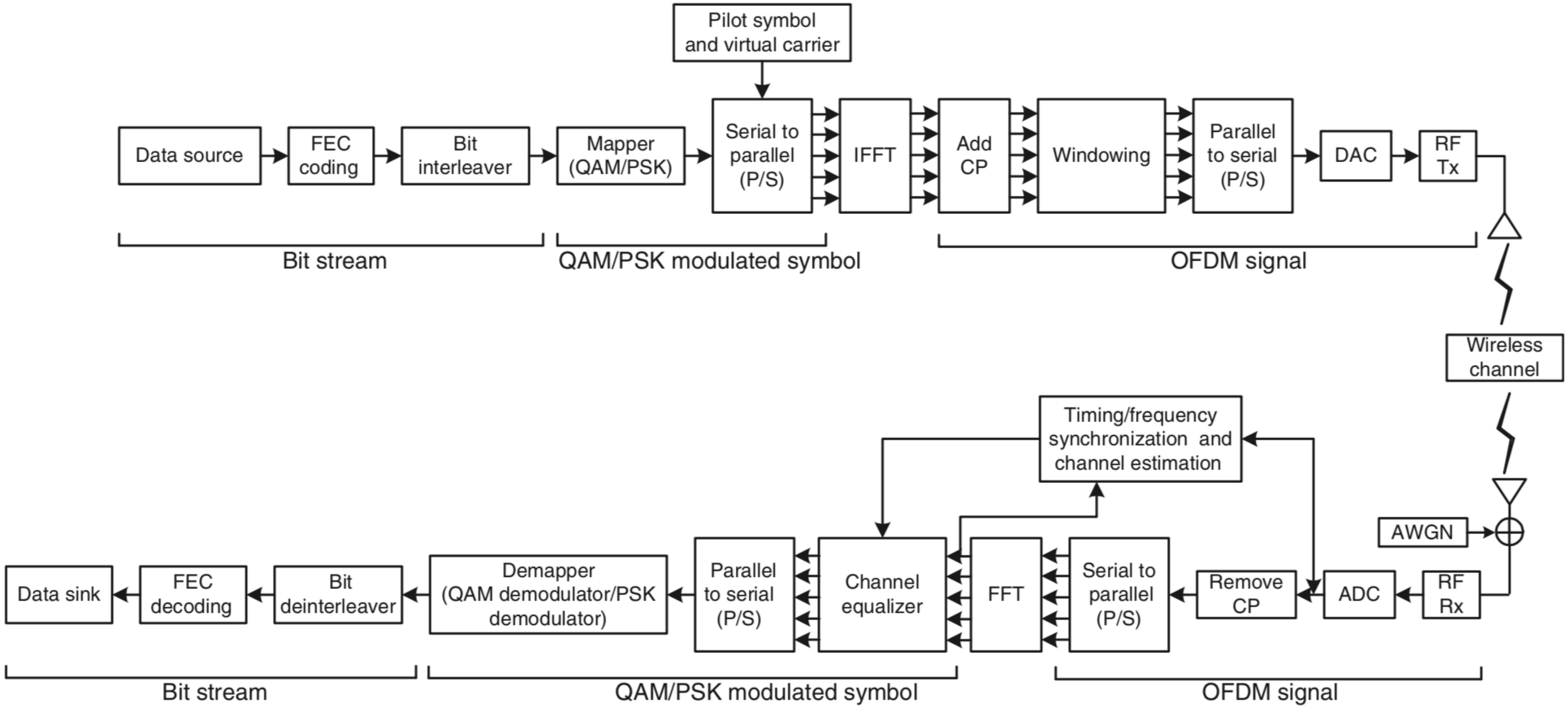
Information data rate	**6, 9, 12, 18, 24, 36, 48, and 54 Mb/s (6, 12, and 24 Mb/s are mandatory) (20 MHz channel spacing)**	**3, 4.5, 6, 9, 12, 18, 24, and 27 Mb/s (3, 6, and 12 Mb/s are mandatory) (10 MHz channel spacing)**	**1.5, 2.25, 3, 4.5, 6, 9, 12, and 13.5 Mb/s (1.5, 3, and 6 Mb/s are mandatory) (5 MHz channel spacing)**
Modulation	BPSK OFDM QPSK OFDM 16-QAM OFDM 64-QAM OFDM	BPSK OFDM QPSK OFDM 16-QAM OFDM 64-QAM OFDM	BPSK OFDM QPSK OFDM 16-QAM OFDM 64-QAM OFDM
Error correcting code	K = 7 (64 states) convolutional code	K = 7 (64 states) convolutional code	K = 7 (64 states) convolutional code
Coding rate	1/2, 2/3, 3/4	1/2, 2/3, 3/4	1/2, 2/3, 3/4
Number of subcarriers	52	52	52
OFDM symbol duration	4.0 µs	8.0 µs	16.0 µs
GI	0.8 µs[a] (T_{GI})	1.6 µs (T_{GI})	3.2 µs (T_{GI})
Occupied bandwidth	16.6 MHz	8.3 MHz	4.15 MHz



Tx and Rx Blocks (Standard)



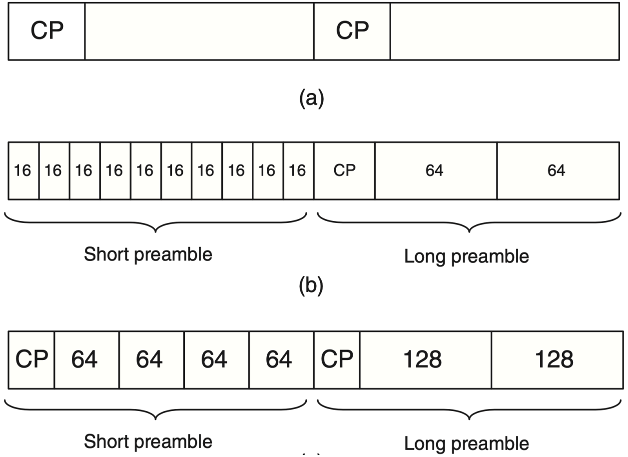
Tx/Rx Blocks



Yong Soo Cho, Jaekwon Kim, Won Young Yang, Chung G. Kang, MIMO-OFDM Wireless Communications with MATLAB

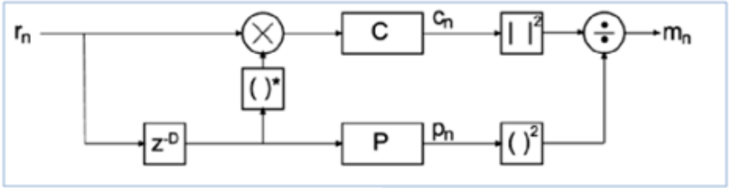
Packet Detection

- Take advantage of the repetition in OFDM signals



- Use Correlator and Maximum Searcher

Start of Symbol Detection



$$c_n = \sum_{l=1}^{N_R} \sum_{K=0}^{D-1} r_{n+k,l} r^*_{n+k+D,l} \qquad p_n = \sum_{l=1}^{N_R} \sum_{k=0}^{D-1} |r_{n+k+D,l}|^2$$

- Correlation with Long Preamble
  - Cross-correlation (Received, Stored)
  - Auto-correlation (Received, Delayed Received)
- Window *C* autocorrelates between the received signal and the delayed version, c_n
- Window *P* calculates the energy received in the autocorrelation window, p_n.
- The decision statistics, m_n, normalize the autocorrelation by p_n so that the decision statistic is not dependent on the absolute received power level.
- The recommended default value of 0.5 for threshold favors false detections over missed detections considering a range of SNRs and various antenna configurations.

$$m_n = \frac{|c_n|^2}{(p_n)^2}$$

Fine Symbol Timing Detection

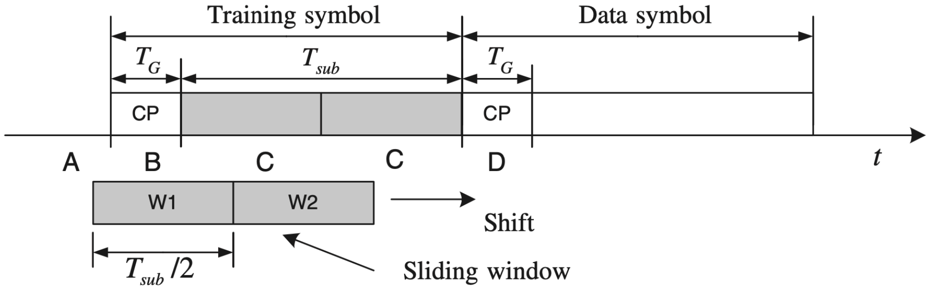
- Instead of correlating the noisy received waveform with a delayed version of the noisy received waveform, the receiver can correlate the received noisy signal with the 'clean' preamble waveform using a matched filter
- Optimal Timing using Cross-correlation:

$$\Phi_{zp}(m) = \sum_{q=0}^{Q-1} z_{m+q} \cdot p_q^*$$

Q = Length of preamble
z = Received samples
p = Preamble samples

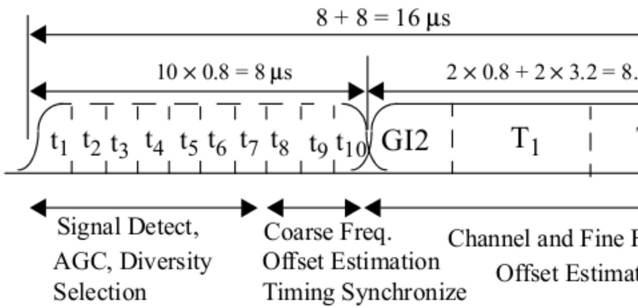
- Start of signal is where the maximum magnitude occurs

$$\hat{m}_{\text{MAX}} = \arg \max_{m} |\Phi_{zp}(m)|$$



Coarse Frequency Offset Correction from SP

- Carrier Frequency Offset:
  - Device impairments introduce difference between the carrier frequency of the receiver and that of the transmitter.
  - When this happens, the received baseband signal, instead of being centered at DC (0MHz), will be centered at a frequency $f_\Delta = f_{Tx} - f_{Rx}$
  - Received signal in baseband (ignoring noise) = $y(t) = x(t)e^{j2\pi f_\Delta t}$
- Short Preamble:
  - Periodic with $\delta t = 0.8\mu s$ = 16 samples
  - $y(t - \delta t) = x(t)e^{j2\pi f_\Delta (t-\delta t)}$
  - At receiver, both $y(t)$ and $y(t - \delta t)$ are known



Coarse Frequency Offset Correction from SP

- $y(t-\delta t)y^*(t) = x(t)e^{j2\pi f_\Delta (t-\delta t)}x^*\,(t)e^{-j2\pi f_\Delta t} = |x(t)|^2 e^{j2\pi f_\Delta (-\delta t)}$
- Considering only the angle, we get $-2\pi f_\Delta \delta t$
- Hence, $CFO(f_\Delta) = -\frac{\angle y(t-\delta t)y^*(t)}{2\pi\delta t}$
- We can use the same technique for Fine Frequency Offset Correction using Long Preamble

Cho, Yong Soo and Kim, Jaekwon and Yang, Won Young and Kang, Chung G., "MIMO-OFDM Wireless Communications with MATLAB," in *Wiley Publishing*.

https://openofdm.readthedocs.io/en/latest/overview.html

Pilot based Phase Noise Correction

- Channel gain (H) is the mean of two LTSs.

$$H[i] = \frac{1}{2}(LTS_1[i] + LTS_2[i]) \times L[i], i \in [-26, 26]$$

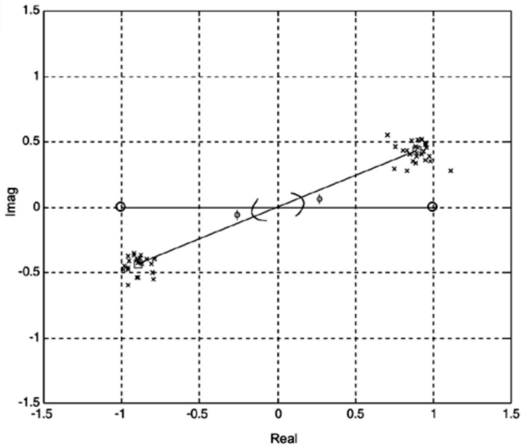
$L[i]$ is the sign of the LTS sequence

- FFT Output *X[i]* is normalized as $Y[i] = \frac{X[i]}{H[i]}, i \in [-26, 26]$
- Residual Phase Offset in symbol *n*

$$\theta_n = \angle(\sum_{i \in \{-21,-7,7,21\}} \overline{X^{(n)}}[i] \times P^{(n)}[i] \times H[i])$$

- Combine residual phase offset & channel gain

$$Y^{(n)}[i] = \frac{X^{(n)}[i]}{H[i]} e^{j\theta_n}$$

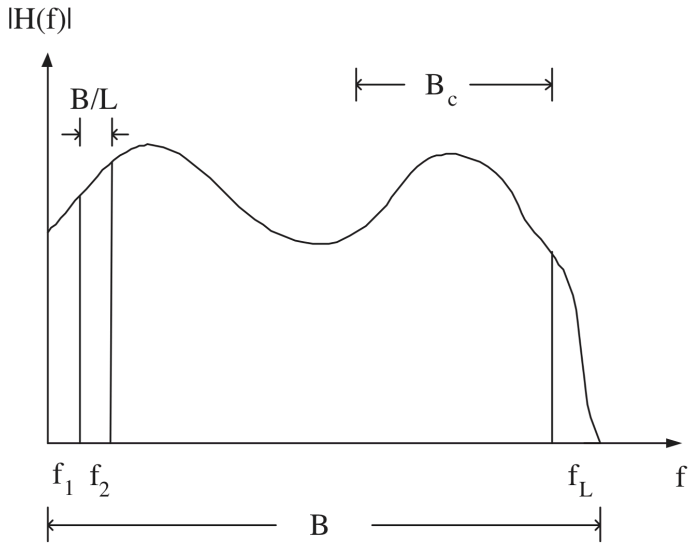


A. van Zelst and T. C. W. Schenk, "Implementation of a MIMO OFDM-based wireless LAN system," in *IEEE Transactions on Signal Processing*, vol. 52, no. 2, pp. 483-494, Feb. 2004.

UNIVERSITY AT ALBANY
State University of New York

Channel Estimation

- OFDM ooperates in frequency selective fading channel
- Divides frequency-selective-faded signal band into a large number of narrow-band flat-fading subchannels
- Objective: obtain precise channel estimation for equalizing each subcarrie
- Requires reference signals
  - Preamble
  - Pilot



Fading across subcarriers

- transmit power on subcarrier i is P_i
- fading on that subcarrier is α_i
- received SNR in subcarrier i is $\gamma_i = P_i \alpha_i^2 / (N_0 B)$
- where N_0 is the noise power and B is the bandwidth

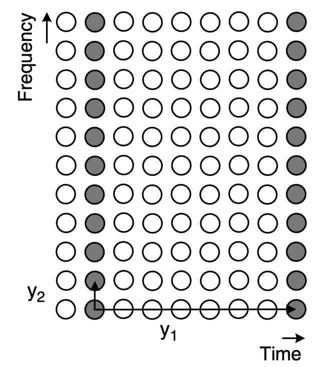
- Received SNR depends on α_i
- α_i varies with time in wireless channels

Frequency Equalization

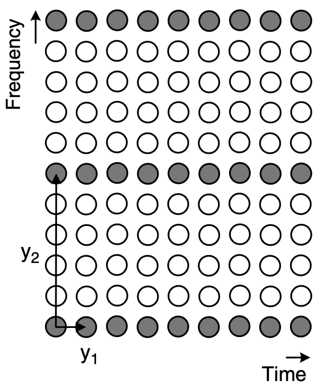
- The fading α_i is inverted in the receiver
- Received signal is multiplied by $1/\alpha_i$
- Received signal power $\frac{P_i {\alpha_i}^2}{{\alpha_i}^2} = P_i$
- Pros: removes the impact of fading
- Cons: it enhances the noise (incoming noise gets multiplied by $1/\alpha_i$)

Pilot Signals

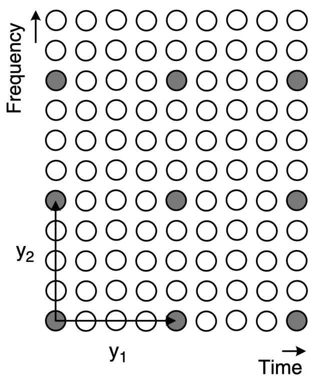
Generic, not specific to 802.11



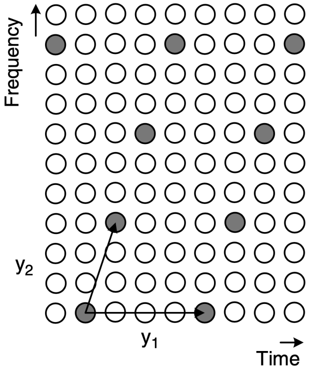
block type



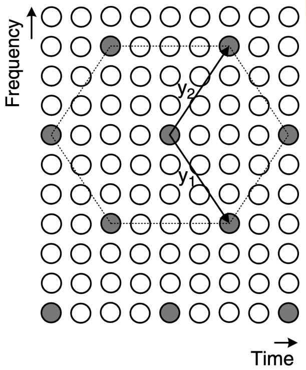
comb type



rectangular grid



parallelogram-shaped grid



hexagonal grid

- Block Type (suitable for slow fading):
  - the interval between two consecutive pilot symbols must be significantly shorter than the channel coherence time.
- Comb Type (suitable for fast fading):
  - spacing of pilot subcarriers must be less than coherence bandwidth of the channel

Channel Estimation (Training based)

- Since all subcarriers are orthogonal (i.e., ICI-free), the training symbols for N subcarriers can be represented by **X**

$$\mathbf{X} = \begin{bmatrix} X[0] & 0 & \cdots & 0 \\ 0 & X[1] & & \vdots \\ \vdots & & \ddots & 0 \\ 0 & \cdots & 0 & X[N-1] \end{bmatrix}$$

- Given that the channel gain is *H[k]* for each subcarrier *k*, the received training signal *Y[k]* can be represented as

$$\mathbf{Y} \triangleq \begin{bmatrix} Y[0] \\ Y[1] \\ \vdots \\ Y[N-1] \end{bmatrix} = \begin{bmatrix} X[0] & 0 & \cdots & 0 \\ 0 & X[1] & & \vdots \\ \vdots & & \ddots & 0 \\ 0 & \cdots & 0 & X[N-1] \end{bmatrix} \begin{bmatrix} H[0] \\ H[1] \\ \vdots \\ H[N-1] \end{bmatrix} + \begin{bmatrix} Z[0] \\ Z[1] \\ \vdots \\ Z[N-1] \end{bmatrix}$$

$$= \mathbf{XH} + \mathbf{Z}$$

  - **H** is the channel vector
  - **Z** is the noise vector

Least-Square (LS) channel estimation

- Goal is to minimize the cost function

$$\begin{aligned} J(\hat{\mathbf{H}}) &= \left\|\mathbf{Y}-\mathbf{X}\hat{\mathbf{H}}\right\|^2 \\ &= (\mathbf{Y}-\mathbf{X}\hat{\mathbf{H}})^H(\mathbf{Y}-\mathbf{X}\hat{\mathbf{H}}) \\ &= \mathbf{Y}^H\mathbf{Y}-\mathbf{Y}^H\mathbf{X}\hat{\mathbf{H}}-\hat{\mathbf{H}}^H\mathbf{X}^H\mathbf{Y}+\hat{\mathbf{H}}^H\mathbf{X}^H\mathbf{X}\hat{\mathbf{H}} \end{aligned}$$

- By setting the derivative of the function with respect to $\widehat{H}$ to zero,

$$\frac{\partial J(\hat{\mathbf{H}})}{\partial \hat{\mathbf{H}}} = -2\left(\mathbf{X}^H\mathbf{Y}\right)^* + 2\left(\mathbf{X}^H\mathbf{X}\hat{\mathbf{H}}\right)^* = 0$$

$$\mathbf{X}^H\mathbf{X}\hat{\mathbf{H}} = \mathbf{X}^H\mathbf{Y}$$

$$\hat{\mathbf{H}}_{LS} = (\mathbf{X}^H\mathbf{X})^{-1}\mathbf{X}^H\mathbf{Y} = \mathbf{X}^{-1}\mathbf{Y}$$

$$\hat{H}_{LS}[k] = \frac{Y[k]}{X[k]}, \quad k = 0, 1, 2, \cdots, N-1$$

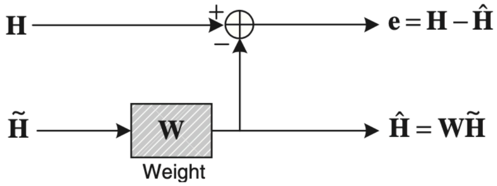
X is assumed to be diagonal due to the ICI-free condition

MSE is inversely proportional to the SNR

MMSE Channel Estimation

Minimize $J(\hat{\mathbf{H}}) = E\{||\mathbf{e}||^2\} = E\{||\mathbf{H} - \hat{\mathbf{H}}||^2\}$

LS Estimate



$$\mathbf{W} = \mathbf{R}_{\mathbf{H}\tilde{\mathbf{H}}}\mathbf{R}_{\tilde{\mathbf{H}}\tilde{\mathbf{H}}}^{-1}$$

$$\hat{\mathbf{H}} = \mathbf{W}\tilde{\mathbf{H}} = \mathbf{R}_{\mathbf{H}\tilde{\mathbf{H}}}\mathbf{R}_{\tilde{\mathbf{H}}\tilde{\mathbf{H}}}^{-1}\tilde{\mathbf{H}}$$

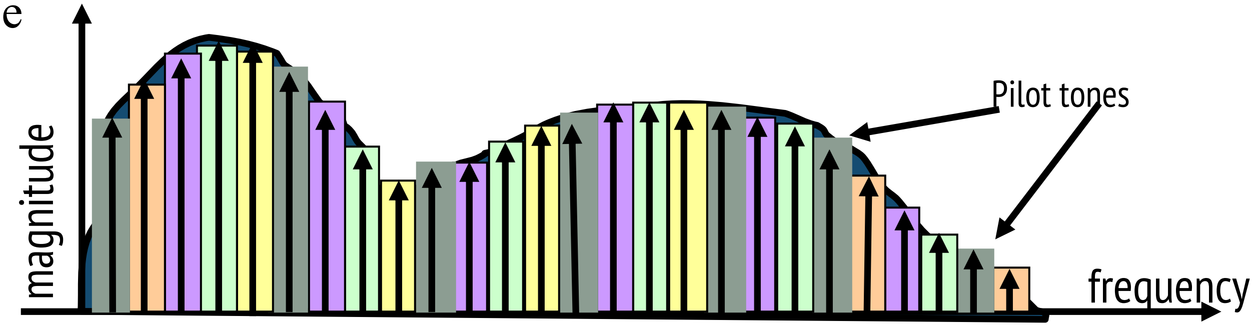
$$= \mathbf{R}_{\mathbf{H}\tilde{\mathbf{H}}}\left(\mathbf{R}_{\mathbf{H}\mathbf{H}} + \frac{\sigma_z^2}{\sigma_x^2}\mathbf{I}\right)^{-1}\tilde{\mathbf{H}}$$

$\mathbf{R}_{\tilde{\mathbf{H}}\tilde{\mathbf{H}}}$ is the autocorrelation matrix of $\tilde{\mathbf{H}}$

$\mathbf{R}_{\mathbf{H}\tilde{\mathbf{H}}}$: Cross-correlation matrix between true channel vector (**H**) and temporary channel estimate vector in the frequency domain

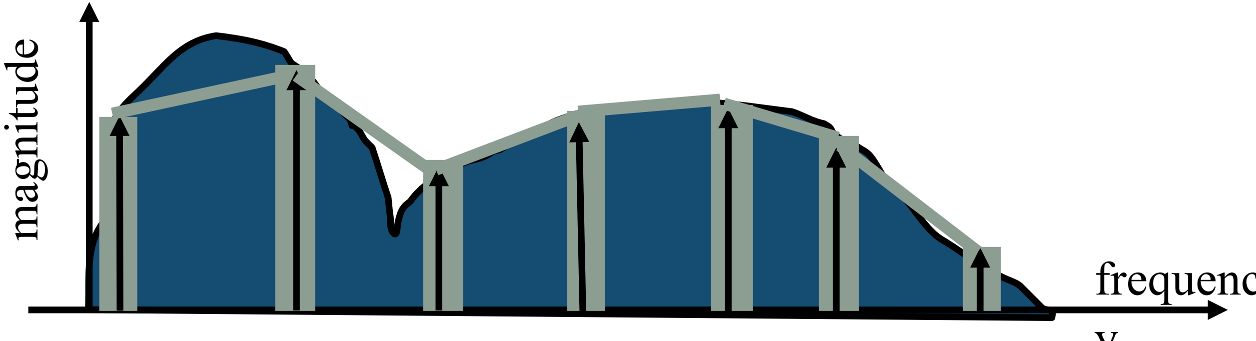
Ideal Channel Estimation

- Wireless channels change frequently ~ 10 ms
- Require frequent channel estimation
- Many systems use pilot tones – known symbols
  - Given s_k, for $k = k_1, k_2, k_3, \ldots$ solve $x_k = \sum_{l=0}^{L} h_l e^{-j2\pi k l/N} s_k$ for h_l
  - Find $H_k = \sum_{l=0}^{L} h_l e^{-j2\pi k l / N}$ (significant computation)
- More pilot tones
  - Better noise resilience
  - Lower throughput



Channel Estimation Via Interpolation

- More efficient approach is interpolation
- Algorithm
  - For each pilot k_i find $H_{ki} = x_{ki} / s_{ki}$
  - Interpolate unknown values using interpolation filter
  - $H_m = a_{m,1} H_{k1} + a_{m,2} H_{k2} + \dots$
- Comments
  - Longer interpolation filter: more computation, timing sensitivity
  - Typical 1dB loss in performance in practical implementation

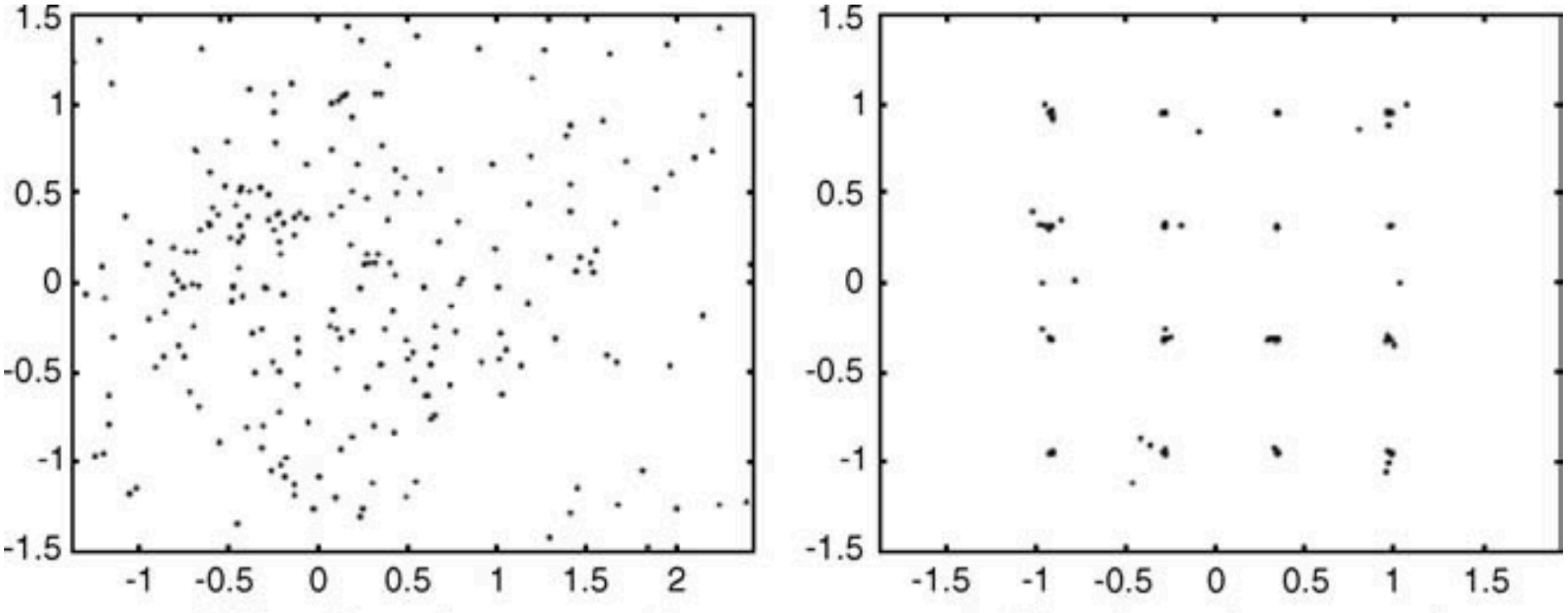


Channel Estimation for WLAN

$$\hat{H}_{LS}[k] = \frac{Y[k]}{X[k]}, \quad k = 0, 1, 2, \cdots, N-1$$

- Long Preamble: contains N=52 subcarriers
- Pilot: contains only 4 (-21, -7, 7, 21)
- Use above equation for 4 pilots
- Then intrapolate/extrapolate rest of the subcarriers
- We can assume channel is fairly constant over duration of one packet in low mobility WLAN
- Often channel estimation is done over LP, pilots are used for phase tracking

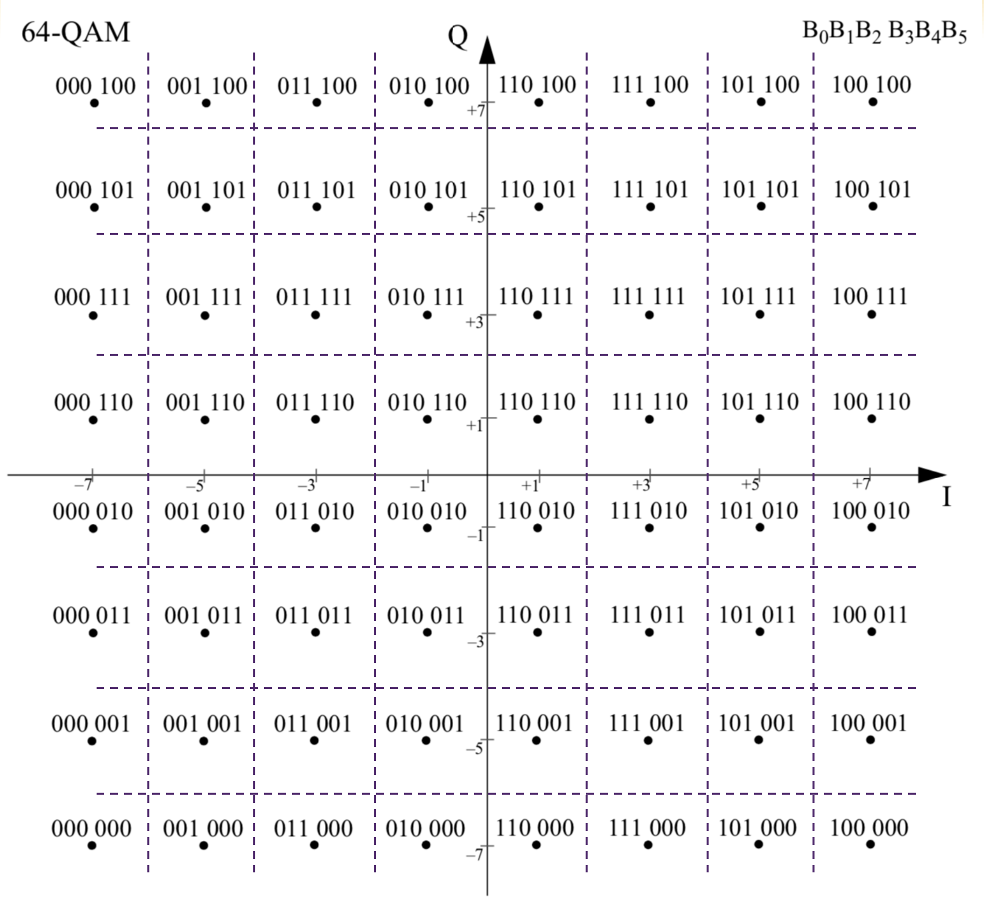
Channel Estimation and Equalization



(a) Before channel compensation (b) After channel compensation

Demodulation

- Find the nearest Euclidian distance to a transmitted constellation
- Use Thresholds



	−7	−5	−3	−1	+1	+3	+5	+7
+7	000 100	001 100	011 100	010 100	110 100	111 100	101 100	100 100
+5	000 101	001 101	011 101	010 101	110 101	111 101	101 101	100 101
+3	000 111	001 111	011 111	010 111	110 111	111 111	101 111	100 111
+1	000 110	001 110	011 110	010 110	110 110	111 110	101 110	100 110
−1	000 010	001 010	011 010	010 010	110 010	111 010	101 010	100 010
−3	000 011	001 011	011 011	010 011	110 011	111 011	101 011	100 011
−5	000 001	001 001	011 001	010 001	110 001	111 001	101 001	100 001
−7	000 000	001 000	011 000	010 000	110 000	111 000	101 000	100 000

Deinterleave

- Inverse of Interleave
- First Permutation

$$i = s \times \lfloor j/s \rfloor + \left(j + \left\lfloor \frac{16 \times j}{N_{CBPS}} \right\rfloor\right) \bmod s \quad j = 0,1,\ldots N_{CBPS} - 1$$

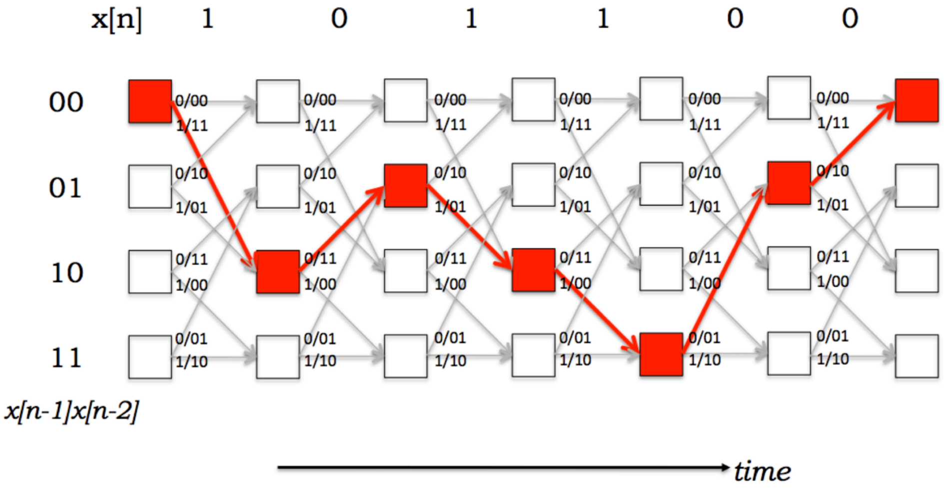
- Second Permutation

$$k = 16 \times i - (N_{CBPS} - 1) \times \left\lfloor \frac{16 \times i}{N_{CBPS}} \right\rfloor, \ i = 0,1,\ldots N_{CBPS} - 1$$

$$s = \max(N_{BPSC}/2,1)$$

Viterbi Decoding

- The Viterbi algorithm can be used for decoding the space-time trellis-coded systems.
- Soft Decision Decoder – samples before digitizing
- Hard Decision Decoder – bits received



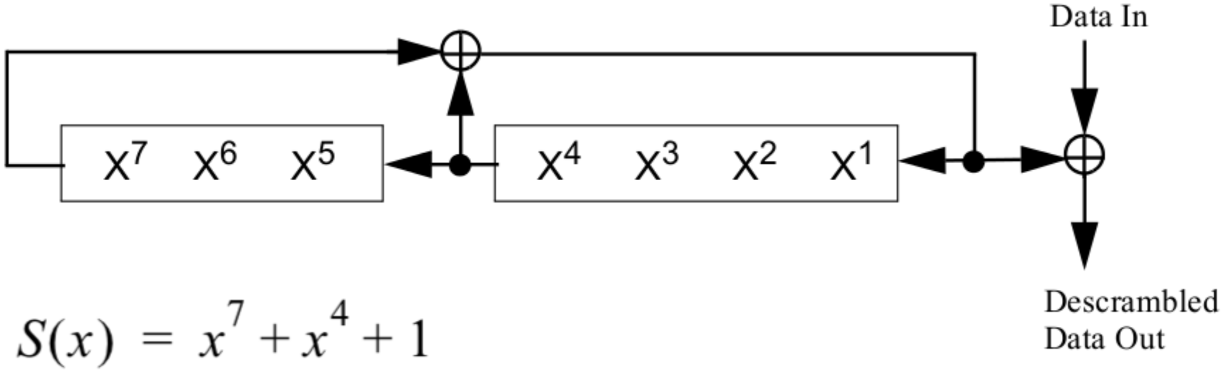
MATLAB Function: vitdec()

https://www.mathworks.com/help/comm/ref/vitdec.html

http://web.mit.edu/6.02/www/f2010/handouts/lectures/L9.pdf

Descrambling

- Inverse of scrambling procedure
- the first 7 un-scrambled bits (B0 to B6) are all zeros

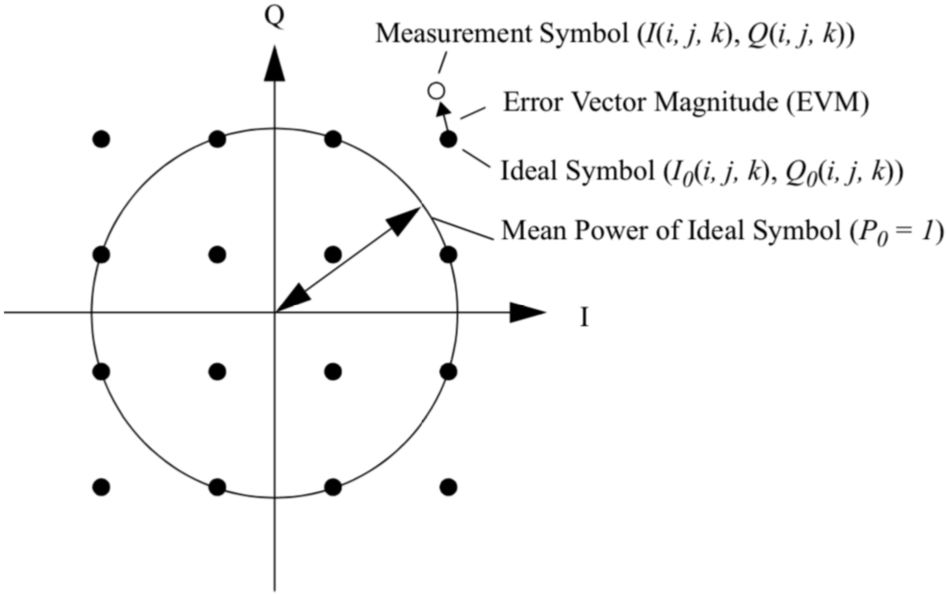


$$S(x) = x^7 + x^4 + 1$$

The initial state of the scrambler is the state 1011101

Constellation Error

- Error Vector Magnitude (EVM)

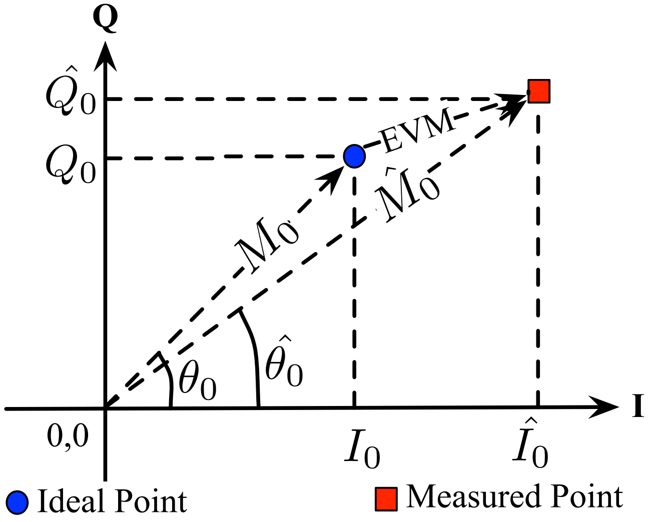


Measurement Symbol ($I(i, j, k), Q(i, j, k)$)

Error Vector Magnitude (EVM)

Ideal Symbol ($I_0(i, j, k), Q_0(i, j, k)$)

Mean Power of Ideal Symbol ($P_0 = 1$)



● Ideal Point
🟥 Measured Point

$\{I_0, Q_0, M_0, \theta_0\}$ = Ideal I, Q, Magnitude, Phase

$\{\hat{I}_0, \hat{Q}_0, \hat{M}_0, \hat{\theta}_0\}$ = Measured I, Q, Magnitude, Phase

$$EVM = \sqrt{\left(I_0 - \hat{I}_0\right)^2 + \left(Q_0 - \hat{Q}_0\right)^2}$$

I_{Disp} = Dispersion in I = $I_0 - \hat{I}_0$

Q_{Disp} = Dispersion in Q = $Q_0 - \hat{Q}_0$

M_{Disp} = Dispersion in Magnitude = $M_0 - \hat{M}_0$

RMS Average of all Errors in a Packet

$$Error_{RMS} = \frac{\sum_{i=1}^{N_f} \sqrt{\frac{\sum_{j=1}^{L_P}\left[\sum_{k=1}^{52}\{(I(i,j,k)-I_0(i,j,k))^2+(Q(i,j,k)-Q_0(i,j,k))^2\}\right]}{52L_P \times P_0}}}{N_f} \qquad (17\text{-}28)$$

where

L_P is the length of the packet;

N_f is the number of frames for the measurement;

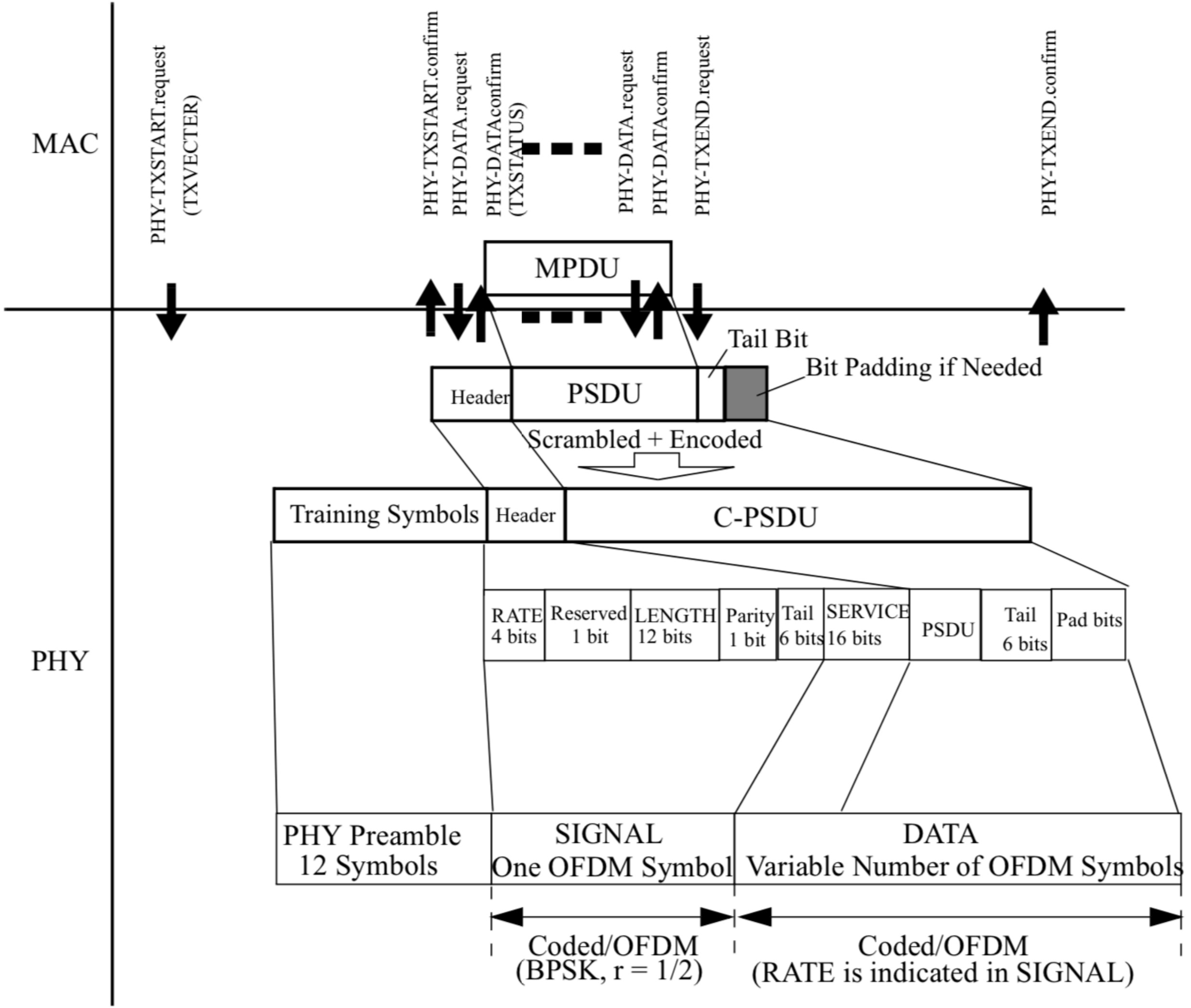
$(I_0(i,j,k), Q_0(i,j,k))$denotes the ideal symbol point of the i^{th} frame, j^{th} OFDM symbol of the frame, k^{th} subcarrier of the OFDM symbol in the complex plane;

$(I(i,j,k), Q(i,j,k))$denotes the observed point of the i^{th} frame, j^{th} OFDM symbol of the frame, k^{th} subcarrier of the OFDM symbol in the complex plane (see Figure 17-16);

P_0 is the average power of the constellation.

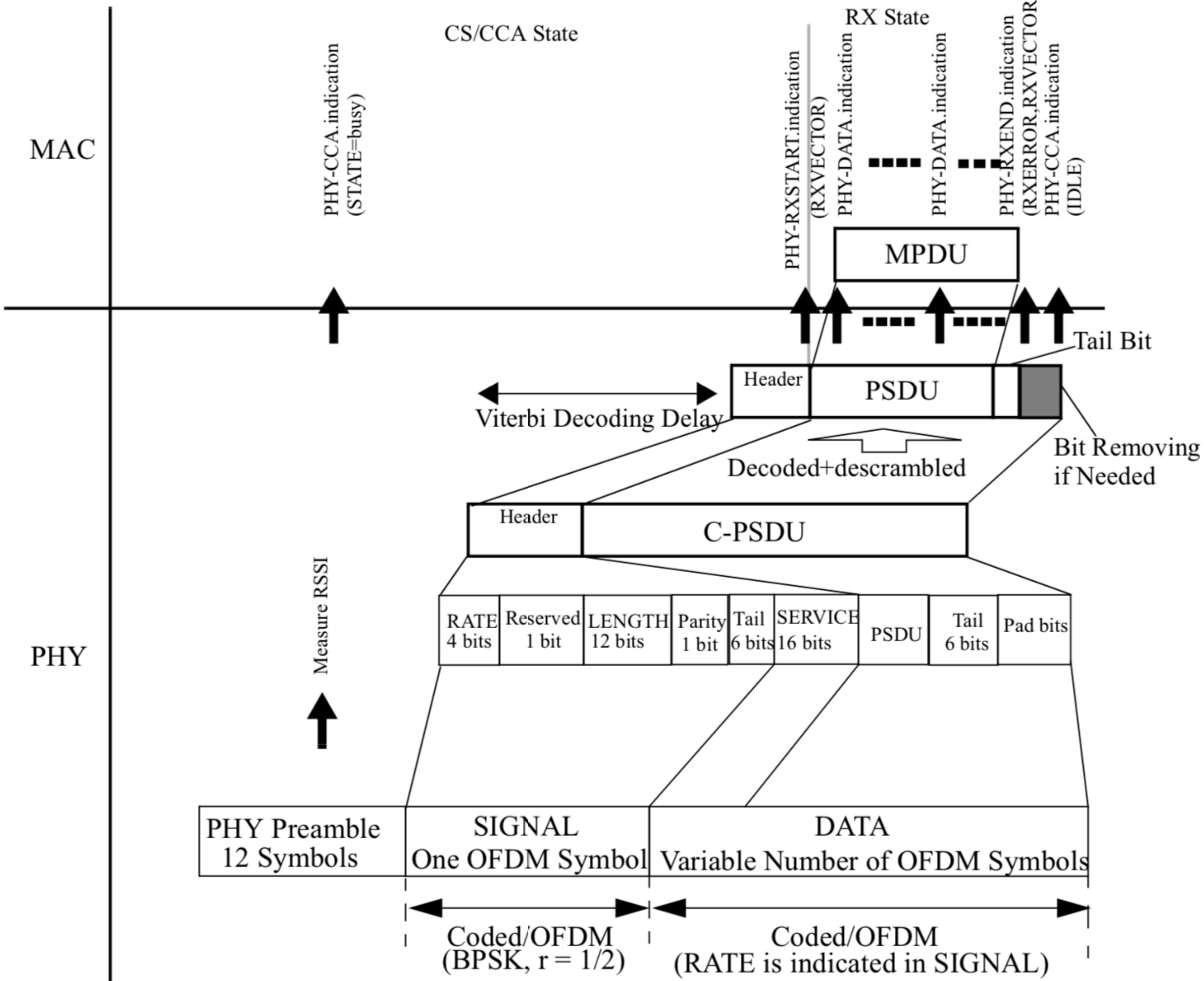
MAC/PHY

- Tx


MAC
PHY
PHY-TXSTART.request
(TXVECTER)
PHY-TXSTART.confirm
PHY-DATA.request
PHY-DATAconfirm
(TXSTATUS)
PHY-DATA.request
PHY-DATAconfirm
PHY-TXEND.request
PHY-TXEND.confirm
MPDU
Tail Bit
Bit Padding if Needed
Header
PSDU
Scrambled + Encoded
Training Symbols
Header
C-PSDU
RATE 4 bits
Reserved 1 bit
LENGTH 12 bits
Parity 1 bit
Tail 6 bits
SERVICE 16 bits
PSDU
Tail 6 bits
Pad bits
PHY Preamble 12 Symbols
SIGNAL One OFDM Symbol
DATA Variable Number of OFDM Symbols
Coded/OFDM (BPSK, r = 1/2)
Coded/OFDM (RATE is indicated in SIGNAL)

MAC/PHY

- Rx


MAC
PHY
CS/CCA State
RX State
PHY-CCA.indication
(STATE=busy)
PHY-RXSTART.indication
(RXVECTOR)
PHY-DATA.indication
PHY-DATA.indication
PHY-RXEND.indication
(RXERROR,RXVECTOR)
PHY-CCA.indication
(IDLE)
MPDU
Tail Bit
Header
PSDU
Viterbi Decoding Delay
Decoded+descrambled
Bit Removing
if Needed
Header
C-PSDU
Measure RSSI
RATE
4 bits
Reserved
1 bit
LENGTH
12 bits
Parity
1 bit
Tail
6 bits
SERVICE
16 bits
PSDU
Tail
6 bits
Pad bits
PHY Preamble
12 Symbols
SIGNAL
One OFDM Symbol
DATA
Variable Number of OFDM Symbols
Coded/OFDM
(BPSK, r = 1/2)
Coded/OFDM
(RATE is indicated in SIGNAL)