Communication Systems Laboratory
Lab 3: OFDM, Frame Synchronization & Equalization

Report Due: 11:59 pm, Oct. 14, 2025

The goals of this lab are (a) to implement OFDM waveform generation and processing, and (b) to implement Wi-Fi Short Training Sequence (STS) and Long Training Sequence (LTS) for frame synchronization and equalization.

For the submission, please use the MATLAB live script (.mlx).

[Part 1] OFDM signal generation and reception (38.5 points)

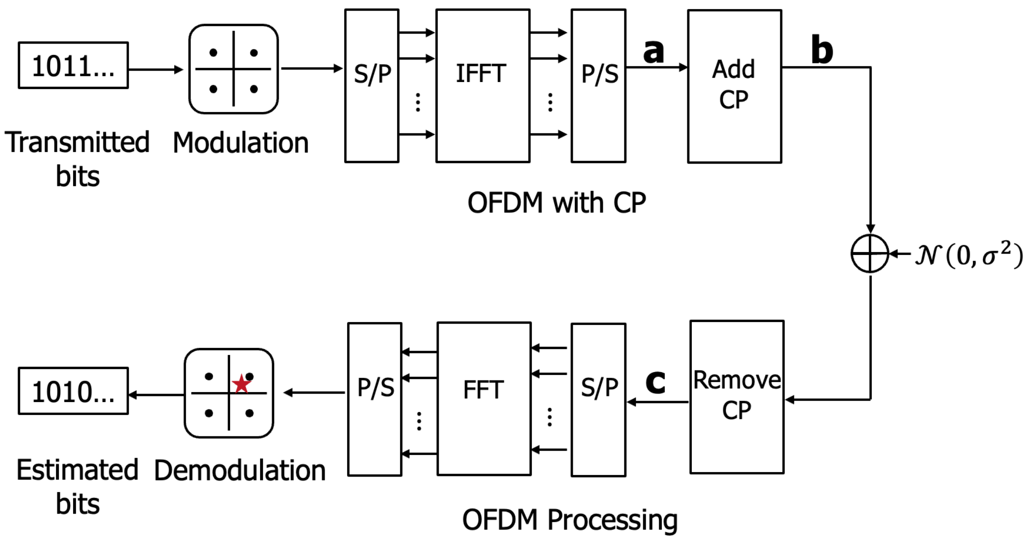

Figure 1. OFDM waveform generation and processing.

In Part 1, you will generate the Wi-Fi OFDM-based waveform using the parameters defined in the Wi-Fi standard. The parameters are as follows:

- FFT size = 64
- Size of cyclic prefix = 16
- Sampling rate = 20 MHz
- Use data tones and pilot tones as defined in 802.11a/g (please refer to the course slide)
- Use BPSK for the pilot tones

Simulation Procedure:

- Generate random bits for the pilot tones and modulate them with BPSK.
- Separately, generate random bits for data tones and modulate them with 16-QAM.
- Generate OFDM symbols and then add the cyclic prefix.
- Add Gaussian noise for a target SNR. Notice that this simulation is represented using the complex envelope, and thus the added noise should be of complex normal distribution.
- Process OFDM symbols by first removing the cyclic prefix and then performing FFT.
- Demodulate the received symbols.

- Compare the received bits to the transmitted bits to obtain the bit error rate (only the data tones).

Waveform generation for one OFDM symbol

(1) Plot the magnitude of the frequency spectrum for the OFDM symbol with the 0 subcarrier in the center and the correct frequency unit in MHz. Also, use a different color for the pilot tones.
(2) Following (1), what is the subcarrier spacing, i.e., the frequency difference between two adjacent subcarriers?
(3) Perform IFFT to obtain the OFDM symbol in the time domain, i.e., signal (**a**). Please ensure you correctly shift the frequency before performing IFFT (you should not perform IFFT when the 0 subcarrier is in the middle). Please plot both the real and imaginary parts of the OFDM time domain signal with the correct time unit in microseconds.
(4) Add the cyclic prefix to signal (**a**) to obtain signal (**b**). Plot signal (**b**) in the time domain, for both the real and imaginary parts, with a clear legend and correct time unit. Also, mark the part of the cyclic prefix in the plot.
(5) Following (4), what is the duration of this OFDM symbol with the cyclic prefix? What is the duration of the OFDM symbol without the cyclic prefix?
(6) Assume the channel is noiseless. Remove the cyclic prefix at the receiver to obtain signal (**c**). Plot the real part of (**c**) and (**a**) on the same plot to show the cyclic prefix is correctly removed. As before, add the legend and mark the axis with the correct time unit.
(7) Perform FFT to signal (**c**) to obtain the received symbols of each subcarrier. In the same figure, plot the magnitude of the frequency spectrum of both the received and the transmitted OFDM symbol to show they are identical. As before, have the 0 subcarrier in the center, add the legend, and mark the axis with the correct frequency unit.

Simulate OFDM transmissions for different SNRs

(8) Generate 1000 different OFDM symbols. For the time domain waveform, what is the peak power, and what is the average power? As a result, what is the Peak-to-Average Power Ratio (PAPR)?
(9) Transmit and receive 1000 different OFDM symbols with SNR = 15dB. Plot the constellation diagram for the received symbols.
(10) Following (8), calculate the SNR using the 1000 OFDM symbols and confirm that it is close to 15 dB.
(11) Transmit and receive 1000 different OFDM symbols with SNR = {0dB, 5dB, 10dB, 15dB} and calculate the BER for the data tones (i.e., excluding pilot tones). Plot the BER vs SNR. Briefly describe the figure.

[Part 2] STS & LTS for frame synchronization and equalization (59.5 points)

In this lab, we will first generate STS and LTS as defined in Wi-Fi. Next, we will use them to synchronize the frame (i.e., detect the beginning of the frame) and equalize the received signal. Please use the parameters specified in Part 1 if not otherwise stated.

STS & LTS generation and frame synchronization

(1) Generate the short training sequence (STS). Please refer to $S_{-26,26}$ in the course slide for STS generation. Plot the STS (10 repeated symbols) in the time domain with the x-axis in μS.
(2) Generate the long training sequence (LTS). Please refer to $L_{-26,26}$ in the course slide for LTS generation. Plot the LTS (2 long training symbols + cyclic prefix as in the course slide) in the time domain with the x-axis in μS.
(3) Create a frame starting with STS, LTS, followed by 5 OFDM symbols (using the same procedure in part 1 with 16-QAM for the data tones, and BPSK for the pilot tones). Normalize the signal if needed so that the whole frame has consistent power. Plot the whole frame in the time domain with the x-axis in μS. Mark STS, LTS, and the data portion in the frame.
(4) Create your own matched filter based procedure that detects the **short training symbol** to correctly determine the start of the OFDM data symbols. Explain your synchronization algorithm.
(5) Following (4), to show the correctness of your proposed procedure, let's test your algorithm on a zero-padded noiseless frame. Specifically, add 30 zeros before and after the created frame (that is, the frame starts with 30 zeros, followed by STS, LTS, OFDM data, and then 30 zeros). Next, apply your proposed algorithm in (4). Plot the match filter result and briefly explain the figure. Does your algorithm correctly identify the beginning of the frame and, thus, the beginning of each OFDM symbol?
(6) Following (4), process the 5 OFDM symbols based on the matched filter result and show that no error occurs.
(7) Following (4), let's test your algorithm under a noisy channel. As in (5), please start with the zero-padded frame (30 zeros, followed by STS, LTS, OFDM data, and then 30 zeros). Next, add white Gaussian noise with 0 dB SNR. Apply your proposed algorithm in (4). Plot the match filter result and briefly explain the figure. Does your algorithm correctly identify the beginning of the frame?
(8) Now, create another matched filter based procedure that detects the **long training symbol**. Please explain your synchronization algorithm.
(9) Similar to (7), but now use your proposed algorithm in (8). Plot the match filter result and briefly explain the figure. Does your algorithm correctly identify the beginning of the frame?
(10) Compare the results in (7) and (9). What do you observe and why?

Channel estimation and equalization for a multipath channel

In the previous simulation steps, the channel is assumed to simple LoS. Here, we further consider a channel with two paths:

- A LoS component: signal attenuation by a factor of 0.4, delay = $0\mu S$.
- An NLoS component: signal attenuation by a factor of 0.3, delay = $0.5\mu S$.

For simplicity, the channel is assumed to be noiseless. For this simulation, use the frame you obtain in (3), which contains STS, LTS, and 5 OFDM symbols.

(11) Using the time domain samples of the frame, simulate the LoS component and the NLoS component, and sum them to obtain the received signal. Please plot the time domain of the LoS component, NLoS component, and the summed signal in one figure, with the x-axis in μS and a clear legend.
(12) We further consider a receiver that is not perfectly time-synchronized, so that it does not start

sampling at $t = 0\mu$S. Instead, the receiver starts sampling at $t = 0.2\mu S$ so that it misses the first few samples. Using the summed signal obtained in (11), obtain the received samples with imperfect time synchronization. Please plot the first 32 samples of the time domain signal for the imperfect synchronization, compared to the case of perfect synchronization (start sampling at $t = 0\mu$S), to show that the imperfect time synchronization is correctly implemented, with the x-axis in μS and a clear legend. Please briefly describe the figure to explain how you confirm that the imperfect synchronization is correctly implemented.

(13) Apply your synchronization algorithm in (8) to the received signal with imperfect time synchronization obtained in (12). Find the samples corresponding to STS, LTS, and each OFDM symbol. Please plot the time domain signal of the frame and mark each part (STS, LTS, OFDM data symbols) in the figure, with the x-axis in μS.

(14) Following (13), perform channel estimation using the LTS samples you identified in (13). Remember to account for the cyclic prefix, and also note that the time domain signal should be first converted to the frequency domain for channel estimation. Plot the estimated channel across all subcarriers (magnitude and phase). As before, please have the 0 subcarrier in the middle and label the x-axis with the correct frequency unit.

(15) Describe the estimated channel obtained in (14). What do you observe and why?

(16) Use the estimated channel obtained in (14) to equalize the received OFDM symbols. Remember that the equalization happens in the frequency domain, so you need to first convert the time domain signals to the frequency domain. Please plot the constellation before and after the equalization for the data tones to show the effectiveness of the equalization.

(17) Demodulate the equalized constellation obtained in (16), and show that it can be correctly demodulated without any error.

Grading:

- 3.5 points for each problem. 98 points in total for 28 questions.
- 2 points for submitting the assignment.