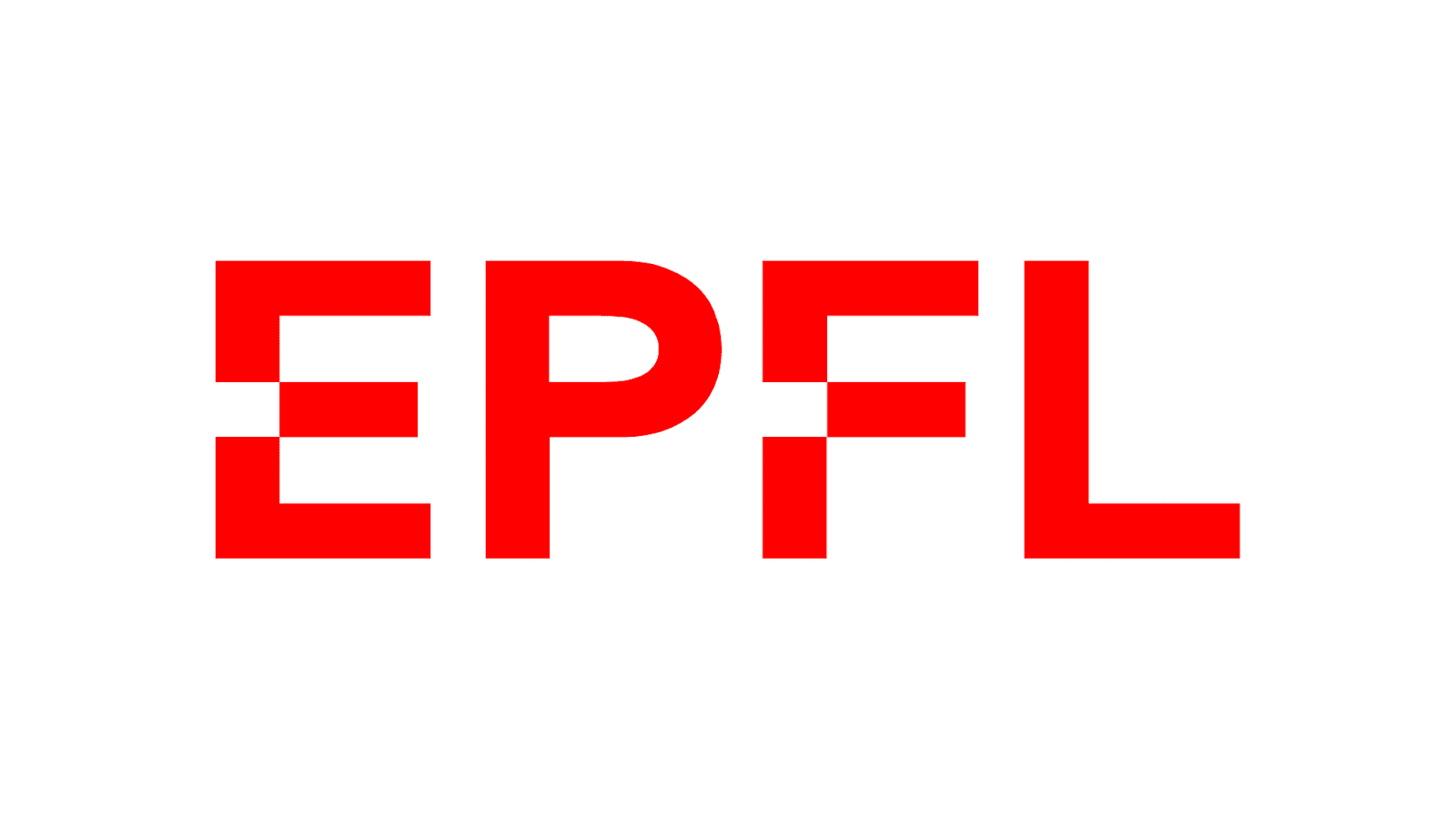
**École Polytechnique Fédérale de Lausanne**



Wireless receivers: algorithms and architectures

**OFDM Project**

Professor:                                                                                                                   Candidates:

Andreas Burg                                                                                                 Francessco Poluzzi

Melani Cvetkovska

Contents

[Introduction 3](#_Toc186199191)

[Transmitting the signal 5](#_Toc186199192)

# Introduction

Insingle-carrier modulation, all data is transmitted using one carrier frequency, making it simple but highly sensitive to channel impairments like multipath fading and interference.

This was not a reliable modulation for modern technologies where we seek reliability and security. So that’s why we avert to multi-carrier modulation techniques. The idea is to divide the data across multiple carrier frequencies and send them with N parallel streams, improving resilience to impairments.

Orthogonal Frequency Division Multiplexing (OFDM) is a cornerstone modulation technique in modern digital communications, celebrated for its capacity to efficiently utilize bandwidth while mitigating intersymbol interference (ISI) and combating the challenges of multipath fading. As communication systems demand greater data rates and robustness, OFDM has emerged as an optimal solution, striking a balance between spectral efficiency and system reliability.

OFDM divides the available channel bandwidth into numerous subcarriers, each transmitting data independently. These subcarriers, spaced at precise intervals based on the symbol rate, maintain orthogonality in the frequency domain. This orthogonality ensures that signals transmitted over one subcarrier do not interfere with those on adjacent subcarriers, thereby upholding the principles of interference-free transmission while enabling simultaneous data streams.

One of the most compelling advantages of OFDM lies in its simplified channel equalization. The narrowband nature of each subcarrier ensures that the channel response remains nearly constant within its scope, reducing the complexity of combating multipath effects. This trait, combined with OFDM's inherent resilience to noise and interference, renders it particularly effective in challenging environments with significant distortion or reflection.

This report delves into the theoretical foundation of OFDM, tracing its evolution from single-carrier modulation to its current form as a multicarrier paradigm. Through an exploration of system design and implementation, the interplay between critical design parameters and performance metrics such as Bit Error Rate (BER) and channel spectrum are analysed. The discussion also includes various tracking methods to address channel variability, ensuring a comprehensive understanding of the principles and practicalities of OFDM communication systems.

A diagram of different colors

Description automatically generated

Figure 1 Single vs. Multi carrier comparison

The signal after the modulation will have the following format:

Where X(k) is the data stream transmitted on the k-th subcarrier, T is the duration of the OFDM signal, is the frequency of the k-th subcarrier, where is the subcarrier spacing with .

With a sampling interval , we will get to the sampled version of the OFDM signal, given by:

The last equation shows the expression of the Inverse Discrete Fourier Transform (IDFT), showing that IFFT modulation can be used.

# Transmitting the signal

We have some raw bits that need to be transmitted. For this to be done we need a clear plan of the steps we need to take to process the bits into a signal suitable for transmission through the physical channel. For that we built the following block diagram and followed it up with a suitable Matlab code.

A diagram of a computer program

Description automatically generated with medium confidence

Figure 2 Block diagram of transmitter

## Mapping

The first step that need to be taken is finding a way to map them into symbols.

In digital communication systems, information is represented in bits (binary data). However, transmission over a physical channel is achieved using modulated waveforms, not raw binary sequences. Mapping bits into symbols is an essential step to bridge the digital world (bits) with the analog domain (waveforms). Symbols represent groups of bits as distinct values or points in a constellation diagram, each corresponding to a unique waveform. In this project Gray QPSK mapping will be used. QPSK is a digital modulation scheme where two bits are mapped onto a single symbol, resulting in four unique symbols. Each symbol corresponds to a phase shift in the carrier signal, typically 0, 90, 180 or 270 degrees. Each constellation point is equidistant from the origin to represent equal energy, and the angle of each point corresponds to a specific phase of the carrier. The Gray part comes into play so that we ensure the symbols differ by 1 bit from one another.

## Forming the TX symbols

The next step is to combine the bits we are transmitting with the training symbols sequence. The training symbols are predefined sequences of bits known to both the transmitter and receiver. They play a crucial role in channel estimation and equalization, enabling the system to mitigate the distortion effects caused by the real channel.  
Two different ways of generating the training symbols were used:   
**block-based** and **comb-based** training methods.

### Block-based training

In the block-based approach, training symbols are transmitted periodically in an OFDM system, with each training symbol spanning across all the subcarriers in the system. These periodic training symbols enable the receiver to perform channel estimation at regular intervals, allowing it to adapt to slowly varying channel conditions effectively.

* Training placement – training sequence T, repeated periodically every K symbols:

, K – block interval, – data symbols

* Channel estimation – the received training symbols are related to the transmitted training symbols by the channel resposnse and the noise:
* Data equalization – for every subsequent data block we equalize with the channel response:

### Comb-based training

On the other hand, the comb-based technique distributes training symbols across the subcarriers within each symbol, with a chess-like structure. By embedding training information within every transmission, the comb-based method supports real-time channel estimation for scenarios with rapid channel variations. The subcarriers are divided into pilot subcarriers, which will be used for channel estimation, and data subcarriers, for the payload.

At a specific subcarrier Pm, where m indexes the pilot subcarriers we have the following:

Since is know, can be estimated as:

To estimate the channel response for non-pilot subcarriers we need to use interpolation techniques. In this case, linear interpolation is used, as following:

While block-based training conserves bandwidth by using fewer training symbols, comb-based training enhances the system's ability to track fast changes in the channel, offering a trade-off between spectral efficiency and the ability to combat fading and distortion.

A comparison of different types of symbols

Description automatically generated

Figure 3 Block and Comb training methods

## Serial to Parallel Conversion

Next logical step is converting the serial signal to parallel N streams to mimic the nature of OFDM. A function is used that reshapes the data into 2D matrix, where the number of rows represents the number of subcarriers, and the columns represent the data for the OFDM symbols distributed across the subcarriers. Each subcarrier’s data is in a separate row, giving the parallel look.

## Inverse Fast Fourier Transform

This operation is used to transform the data from frequency to the time domain, so that the signal can be physically transmitted through the channel. It’s done by combining the modulated subcarriers into a single composite time-domain signal, while ensuring the orthogonality of the subcarriers, to minimize the interference between them.

## Adding Cyclic Prefix

The cyclic prefix (CP) is added to each OFDM symbol by copying the last ​ samples of the symbol and appending them to its beginning. This ensures that the transmitted signal experiences circular convolution with the channel, enabling efficient equalization in the frequency domain using FFT. The CP also acts as a guard interval to combat Inter-Symbol Interference (ISI) caused by multipath fading, preventing overlapping symbols from interfering with one another.

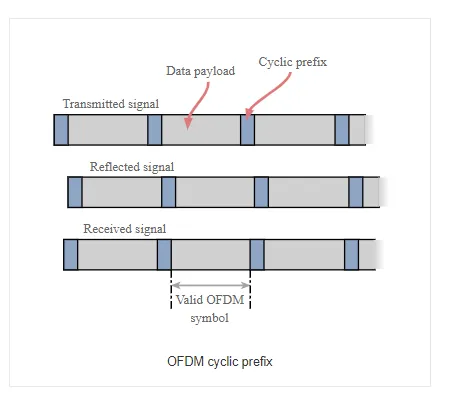


Figure 4 Representation of Cyclic Prefix positioning

Mathematically, for an OFDM symbol x[n] of length N, the transmitted symbol with CP is:

Important note to add is that the cyclic prefix length ​ must be grater than the maximum channel delay spread to fully mitigate the ISI. While adding the CP reduces spectral efficiency, it is crucial for preserving subcarrier orthogonality and ensuring reliable communication in multipath environments.

## Parallel to serial

We need to get the signal to the original serial form because it needs to pass through the actual physical channel which supports serial data.

## Preamble

On the other hand, there are some processing steps done on the preamble as well, which is essential for synchronization. The preamble is generated like a random like bit sequence. It needs to be mapped into symbols, so in this case BPSK mapping is used. Each bit is represented by one of 2 phases: 0 or π. Bit value of 0 is mapped into +1, and a bit value of 1 is mapped into -1. Then the preamble is up sampled to match the transmission rate of the transmission signal and for better precision to be achieved when syncing. At the end the preamble is pulse shaped with the RRC filter so that the bandwidth and the ISI are limited.

## Normalizing signal energy

After inserting the preamble at the start of the signal, the energy of the signal is normalized. Since QPSK mapping is used, normalizing the energy does not cause issues with ISI, as the constellation points remain well-separated. However, with higher-order modulation schemes like 16-QAM or 32-QAM, normalization can compress the constellation points, potentially increasing susceptibility to noise and ISI.

## Up conversion

At the very end before transmitting the signal though the channel up conversion is done, where the baseband signal is shifted to higher frequency, in particular the carrier frequency .

A screen shot of a sound wave

Description automatically generated

Figure 5 Signal that was transmitted

A screen shot of a graph

Description automatically generated

Figure 6 Spectrum of the transmitted signal

# Receiving the signal

After the signal had passed through the physical channel, on the receiver side some processing steps need to be done. The following block diagram represent the steps taken in this project:

A diagram of a flowchart

Description automatically generated

Figure 7 Block diagram of receiver process flow

Firstly, the received bits are now affected by the channel and its impairments, like fading, and noise.

## Down conversion

To process the received signal, down-conversion is performed to shift the signal from the carrier frequency back to the baseband. At the receiver, the incoming radio frequency (RF) signal is mixed with a local oscillator (LO) signal at the carrier frequency ​, effectively removing the high-frequency carrier. This operation isolates the original baseband signal, which contains the transmitted data.

Mathematically, if the received RF signal is:

When mixed with the LO signal it will produce the following:

The first term represents the baseband signal, while the second one is the high frequency part that needs to be removed, which is done in the next step.

## Low Pass Filtering

Since the channel in OFDM modulation acts like a filter, LP Filter will be enough to filter out the high frequencies. This means that the signal will be cut just above the bandwidth, to avoid removing important signal information. In this project we cut off at 30dB, meaning the signal will be attenuated to of the original amplitude. With this we ensure that we remove the noise and successfully isolate the baseband signal for further processing.

## Frame synchronization

To determine where the actual signal starts, frame synchronization needs to be performed. The start of the data is detected by cross correlating the received signal with the known preamble. The matched filter output, combined with a detection threshold, identifies the peak correlation point, which corresponds to the beginning of the data frame. The synchronization process ensures accurate alignment of the receiver with the transmitted signal, compensating for timing offsets introduced by the channel.

## Cyclic Prefix Removal

The next logical step is to remove the CP to extract the payload and training symbols. This ensures that the remaining data corresponds exactly to one OFDM symbol, ready for further operations like FFT and demodulation, without contamination from ISI.

## Serial to Parallel Conversion

Serial-to-parallel conversion is necessary before performing the FFT in an OFDM system because the FFT operates on parallel data streams corresponding to individual subcarriers. The received signal is a serial stream, but each OFDM symbol consists of multiple subcarrier signals transmitted simultaneously. By converting the serial stream into parallel data, each subcarrier's time-domain samples are correctly aligned, enabling the FFT to efficiently transform the data into the frequency domain for demodulation.

## Fast Fourier Transform

FFT makes sure the signal streams are correctly converted to the frequency domain from the time domain. FFT decomposes the dime domain discrete signal with length N into frequency components producing the following discrete signal:

This operation reveals the amplitude and phase of each frequency component kkk in the signal, corresponding to the subcarriers in OFDM.

What is done essentially is that FFT maps the received signal onto its respective subcarriers, effectively demultiplexing the data. Since OFDM modulates data across orthogonal subcarriers, the FFT ensures that the subcarriers remain independent, and their data symbols are efficiently extracted for demodulation and decoding.

## Phase tracking

In this project two scenarios were looked at. One where no phase tracking was one, and one where the Viterbi-Viterbi algorithm was used.   
The Viterbi-Viterbi algorithm is used for phase tracking and equalization of received OFDM symbols, ensuring accurate detection despite phase distortions caused by the channel. Initially, a known training symbol is used to estimate the phase shift introduced by the channel. This phase shift is calculated by comparing the received signal with the training symbol. As the receiver processes subsequent symbols, the algorithm tracks changes in the phase by estimating the current phase shift and comparing it to the previous one. A set of possible phase shifts is used to find the closest match, minimizing errors caused by phase drift. The phase estimate is then smoothed to ensure stability, combining the current and previous estimates. Finally, the phase correction is applied to the received symbols, compensating for any phase misalignments, and the equalized symbols are computed. This approach effectively mitigates phase-related distortions and enhances the accuracy of symbol detection, particularly in the presence of phase noise or Doppler shifts.

## Channel equalization

Two different channel equalization methods are implemented, based on the type of training used: one using a conventional training symbol for each OFDM symbol and the other using a comb-type training pattern. In the first method, the channel frequency response is estimated by dividing the received training symbols by the known reference training symbol. This estimate is then used to equalize the received data symbols by compensating for the channel effects, allowing for improved symbol detection.

The second method, called comb-type training, inserts known training symbols at specific intervals across the carriers. For each OFDM symbol, the training positions are identified, and the channel is estimated for these positions by dividing the received symbols by the reference training symbols. The channel estimate is then used to equalize all subsequent symbols for those carriers, allowing for better recovery of the payload data. The comb-type training method is particularly useful in scenarios where fewer training symbols are available but can still provide reliable channel estimates for the carriers with training symbols, helping to reduce the impact of channel distortions on the received signal.

## Parallel to Serial conversion

After the channel equalization, the next step involves converting the received data from parallel to serial form. This is achieved by the serial-to-parallel conversion function, which reorganizes the serial data into multiple parallel streams, corresponding to the individual subcarriers used in the OFDM system. In this process, the data is reshaped so that each column represents the samples for a specific symbol and subcarrier. The reshaped matrix has dimensions based on the number of subcarriers and the number of symbols, allowing for the alignment of the data for further processing.

## Normalizing signal energy

The signal energy is normalized to ensure consistent power levels across the transmitted symbols, preventing any potential distortion or imbalance during the signal processing stages.

## Demapping

The demapping process converts the received symbols back into bits by checking the real and imaginary parts of each symbol. For each symbol, the real part determines the first bit, and the imaginary part determines the second bit. The result is a binary vector, with each element representing the corresponding bit value from the symbol. This step is crucial for recovering the transmitted data from the modulated symbols.

After all this processing of the signal is done, the receiver side has the RX bits which can be shown via various plots and analysing of the channel and flow can be done.

A screen shot of a graph

Description automatically generated

Figure 8 Spectrum of received signal

A screen shot of a sound wave

Description automatically generated

Figure 9 Signal that was received'