EE-442 | **Final Project**
OFDM for Acoustic Transmission

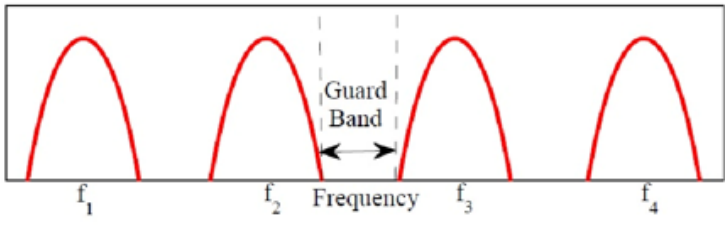
Francesco Cecchetti, Gloria Dal Santo

December 2020

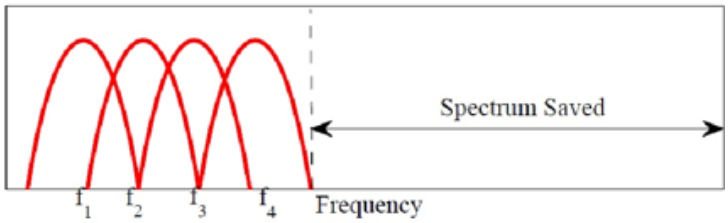
1 Introduction

Frequency Division Multplexing (FDM) is a multi-carrier modulation scheme which consist in sending multiple symbols at the same time dividing the bandwidth into sub-channels and transmitting multiple low-rate signals, also called "carriers", in parallel by carrying each one of them on a separate carrier frequency. As shown in Figure 1, in FDM a frequency "guard-band" is required between two different carriers to avoid ICI (Inter Carrier Interference). The problem in this case is that the bandwidth is not completely used to transmit data thus leading to a not-very-efficient spectral utilisation.

A better approach is OFDM (Orthogonal Frequency Division Multiplexing), which is a FDM transmission system with orthogonal carriers. In order for the carriers to be orthogonal, their spacing in frequency must be a multiple of the symbol rate $1/T_s$, where T_s is the symbol duration. This condition allows the system to fulfil the Nyquist criterion, i.e. that at the central frequency of each carrier (Nyquist points), there is no overlapping between different carriers. Thanks to that, the carriers can be positioned very close to each other in frequency, without having ICI problems and thus leading to a very efficient spectral utilisation. Furthermore, in OFDM and more generally in multi-carrier modulation schemes, the sub-carriers period is large - equivalently speaking they are narrow-banded - thus making both channel estimation and equalisation straightforward as it will be discussed in the next sections.



(a) Conventional FDM.



(b) Orthogonal FDM.

Figure 1: Frequency Division Multiplexing

2 Transmitter

This section discusses the different components of the OFDM transmitter, whose structure is outlined by the following block diagram:

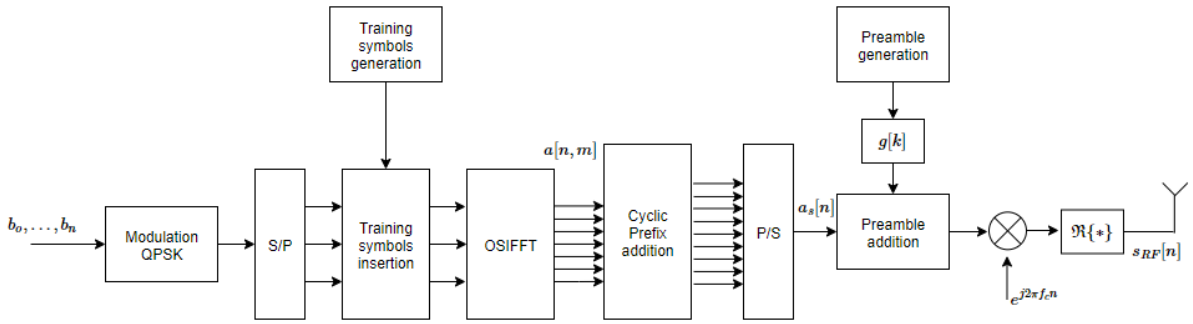


Figure 2: Transmitter Block diagram

2.1 Modulation

First, the uniformly-distributed randomly-generated bit sequence $b_0, ..., b_n$ is translated into a sequence of symbols (also called "constellation points"). The following two types of modulation are considered in this implementation:

- BPSK (Binary phase-shift keying): two phases separated by 180°. Modulation at 1bit/symbol
- QPSK (Quadrature phase-shift keying): four points equally spaced around a circle. Modulation at two bits per symbol.

BPSK modulation was used for the preamble and for the training sequence whereas the QPSK was applied to the data. In order to be transmitted with the same energy, the symbols a_i are normalized as follows:

$$\bar{a}_i = \frac{a_i}{\sqrt{2}} \text{ for QPSK, and } \bar{a}_i = a_i \text{ for BPSK} \tag{1}$$

After the modulation process, the serial stream is converted to N parallel streams, corresponding to the N sub-carriers of the OFDM symbol.

2.2 Training symbol generation and insertion

Key in OFDM systems is to add a training sequence - i.e. a sequence known at both transmitter and receiver sides - to allow the receiver to perform an estimation of the channel response and the initial phase offset. A pseudo-random BPSK-modulated sequence has been generated as training sequence and added to the data stream according to two different types of training: Block-type and Comb-type. A deeper discussion about the reasons that bring to the choice of a pseudo-random sequence and about the two types of training is presented in section 3.4.

2.3 OFDM Modulation: Inverse Discrete Fourier Transform

At this point, the OFDM symbol in the time domain is shaped via Inverse Discrete Fourier Transform: the input symbols correspond to frequency amplitudes coefficients and by using the IDFT, which is implemented very efficiently by the Inverse Fast Fourier Transform (IFFT) algorithm, the time-domain OFDM symbol is created. The reason why a simple IDFT is sufficient is linked to the fact that in time domain each symbol approaches a square shape, while in frequency domain each sub-carrier is sinc-shaped. In this implementation it has been used a function that performs at the same time the IFFT to shape the time-domain signal and the upsampling process. The reason why the signal must be upsampled lies on the sampling frequency of the sound card at the receiver side, which can not be chosen freely, thus leading to an unavoidable oversampling in the system.

2.4 Cyclic prefix addition

Another paramount aspect is the need for a guard interval in between the OFDM symbols. This guard interval is added by prepending the last part of the symbol at the very beginning. This process goes under the name of Cyclic prefix addition (see Figure 3). The reason why this sequence is required is the following. By writing the received sequence $y[n]$ as the convolution between the sent signal $x[n]$ and the channel $h[t]$ is possible to notice that if L is the length of the multi-path channel, only the first $L-1$ samples of the OFDM symbol are affected by the previous one (Inter Symbol Interference). It it thus clear that by adding the cyclic prefix with a length of $N_G >= L$, only the first N_G samples of the symbol (corresponding to the cyclic prefix) will be affected by the previous one whereas the actual symbol (for $n > N_G$) is not affected by the previous one (no ISI). Section 5 shows the trade-offs that have to be considered when choosing the cyclic prefix length.

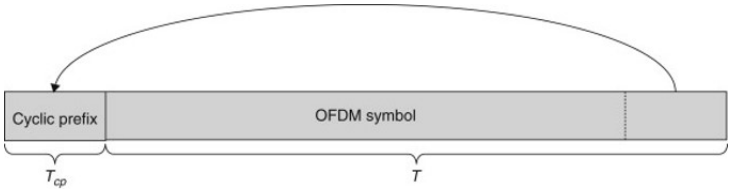


Figure 3: Cyclic prefix addition

2.5 Preamble generation and addition

The parallel stream is converted back to a serial stream and at this point a preamble - known at both receiver and transmitter - is prepended to the sequence to allow the receiver to detect the beginning of the data frame within the received signal. The way the receiver detects the preamble (the detection algorithm) is based on a correlation filter (cfr Section 3.2). In order to have a very clear peak in the correlator's output, the preamble should be a pseudo-random sequence. This sequence is created implementing a Linear Feedback Shift Register (LFSR) and then BPSK modulated. Since the preamble is sent according to a single-carrier modulation scheme, before prepending it to the signal, the generated sequence is filtered by a pulse-shaped filter. The latter process consists in in defining a prototype waveform ("pulse shape") $g[n]$ and then convolving it with the preamble $p_0[n]$:

$$p[n] = p_0[n] \circledast g[n]$$

Since the pulses overlap in time, the choice of the filter $g[n]$ is made both to have an impulse response that converges rapidly to zero and to fulfil the Nyquist criterion:

$$g[kT] = 0 \;\; \forall k \neq 0$$

This condition assures that whenever the transmitted preamble is sampled by the receiver at the time instances $t = nT$, only the n-th pulse contributes to the n-th sample and thus there is no ISI (inter symbol interference). However, the receiver does not sample the signal exactly at multiples of the symbol period T and for this reason at the sample points the pulses are interfering with each other. Thus, the reason why a filter whose impulse response converges rapidly to zero is required.
The chosen filter $g[n]$ is the root-raised-cosine filter (RRC) whose impulse response is:

$$g_{RRC}[n] = \frac{\frac{4\alpha}{\pi}\cos\left(\frac{\pi(1+\alpha)n}{T}\right) + (1-\alpha)\text{sinc}\left(\frac{\pi(1-\alpha)n}{T}\right)}{1-\left(\frac{4\alpha n}{T}\right)^2}$$

At the receiver side the preamble is filtered by a filter matched to the root-raised-cosine filter $g_{RRC}[n]$:

$$g_{MF}[n] = g^*_{RRC}[-n]$$

and it is easy to notice that since the $g_{RRC}[n]$ is symmetrical and real-valued, $g_{MF}[n]$ corresponds exactly to the $g_{RRC}[n]$. The total effective filter is then the convolution between the two root-raised-cosine filters,

which is usually called RC (Raised Cosine) and it can be shown that the Raised-Cosine filter fulfils the Nyquist criterion.

At this point, the transmitted signal - which consist in the pulse-shaped preamble and the OFDM-modulated symbols - is normalized by dividing the preamble and OFDM symbols by the square-root of their respective average power.

2.6 Up-conversion and transmission

In order to be transmitted through the acoustic channel, the sequence must be transformed into a real-valued sequence and its spectrum must be shifted around a carrier frequency f_c. These two steps are simply implemented as follows:

$$s_{RF}[n] = \Re\{s[n]e^{j2\pi f_c n}\}$$

At this point the signal is converted into time domain by the Digital to Analog Converter (DAC) of the transmitter and sent through the channel.

3 Receiver

The receiver's components are schematised in the following block diagram:

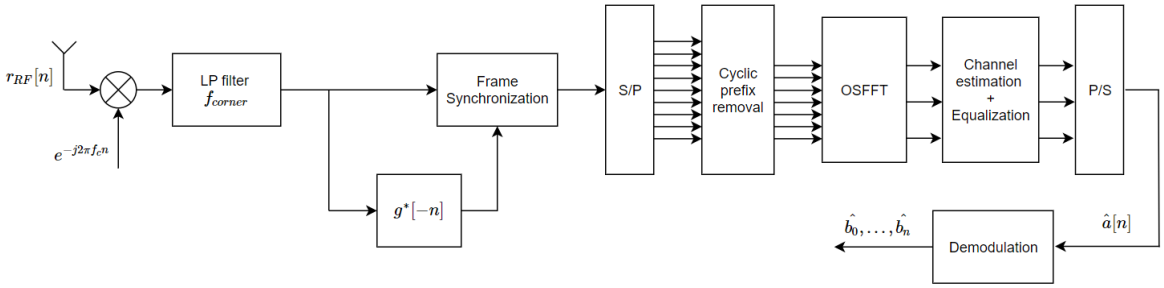


Figure 4: Receiver Block diagram

3.1 Down-conversion and low-pass filtering

The continuous-time signal is sampled by the receiver: $r_{RF}[n] = r_{RF}(nT_s)$, where $f_s = 1/T_s$ is the sampling frequency, and then down-converted to the base-band by simply multiplying it by the complex phasor $e^{-j2\pi f_c n}$.
After the down conversion, the useful base-band signal is extracted by means of a IIR low pass filter with corner frequency $f_{corner} = 1.05 \cdot BW_{BB}$.

3.2 Frame synchronization

At this point the receiver must detect the beginning of the data frame, by means of the preamble prepended to the symbols sequence. The detection is implemented using a correlator, *i.e.* a filter matched to the complex-conjugate, time-reversed preamble. The pseudo-random structure of the preamble allows the receiver to see a clear peak in the correlator's output and thus to detect the beginning of the data. By virtue of the generalised likelihood test, the beginning of the frame is the index for which, the magnitude of the correlator's output, normalized to the energy of the signal's portion inside the correlator, is bigger than a threshold value. The threshold value must be carefully chosen considering that an excessively low value brings to "false alarms", *i.e.* the receiver starts receiving the sequence at a point which does not correspond to the actual beginning of data; conversely, if the threshold is too high, a "detection miss" (the receiver misses the beginning of the data frame) would occur. After analysing the trade-offs between the two scenarios, a threshold value of 30 has been set.

3.3 Cyclic prefix removal and DTFT

Once the beginning of the data has been detected, the serial stream is converted back into a parallel stream to allow the simultaneous conversion into frequency domain via IDTFT. Before this conversion, the cyclic prefix is removed. The conversion into frequency domain works exactly in the opposite way as the one outlined in section 2.3. Again the function OSFFT, along with the frequency domain output, performs a down-sampling in the system.

3.4 Channel estimation and equalization

The effects of the channel on the transmitted signal are in the form of a phase and magnitude variations that must be estimated and corrected in the channel equalization step. Channel estimation is performed by sending a OFDM training symbol which is known also by the receiver so that the Maximum-Likelihood per-tone channel estimator can be found as follows:

$$\hat{H}_l = \arg\max_{H_l} |Y_l - H_l T_l|^2 = \frac{Y_l}{T_l}$$

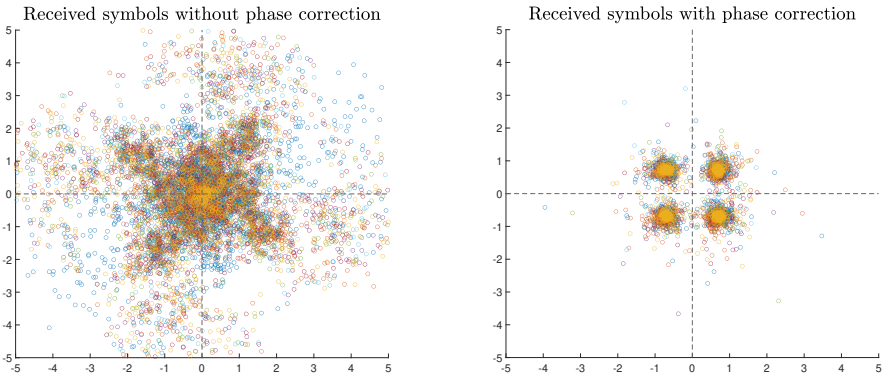
where H_l is a complex number that represent the transfer function of the channel at sub-carrier l, Y_l is the received sub-carrier symbol and T_l is the sub-carrier coefficient of the training symbol.
Since T_l is known, once the receiver detects the training sequence it can directly compute $\hat{H}_l$ and correct Y_l. Then, assuming that the channel is block fading, all the following received symbols can be corrected with the same estimate $\hat{H}_l$. However, this method is most likely to fail in most of the cases since it does not track any changes in the channel until another frames is sent, thus it keeps correcting the symbols with an outdated estimate. To overcome this issue, three different implementations for continuous training can be carried out:

- **Block-training**: the training symbol is sent every n data symbols, where n must be set according to the channel conditions. In this way the channel estimate $\hat{H}_l$ can be updated more frequently;
- **Block-training with Viterbi-Viterbi**: the training symbol is sent every n data symbols and Viterbi-Viterbi algorithm is applied on a sub-carrier level to keep track of the phase changes that might occur while waiting for the next training symbol to arrive. This method is more robust under fast changing channels and helps to reduce the amount of data overhead. It should be noticed that Viterbi-Viterbi algorithm is efficient only for phase offset that can be modelled as a slow varying signal $\theta[n] = \theta[n-1] + \Delta\theta[n]$ where $\Delta\theta[n] \sim \mathcal{N}(0, \sigma^2_{\Delta\theta})$, thus phase offsets that arise from carrier frequency mismatches might be too large to be tracked. This last remark applies also to the other two channel estimation methods;
- **Comb-training**: the entire training symbol is sent on multiple symbol periods. In each time instant, only a part ($n < N$) of the sub-carriers contain the training coefficients , while the other ones keep carrying the actual data. In this way, data and training symbols are continuously transmitted. If $n \geq L$, channel estimation operation can be performed at each arriving symbol, by employing a proper $N \times n$ Fourier matrix. However, in the current system the already implemented FFT function is used and the training symbol is sent in a checkerboard pattern so that channel estimation can be fully computed every two symbol periods.

Channel equalization is then achieved as follows:

$$\hat{a}_l = \frac{Y_l}{|\hat{H}_l|} e^{-j \arg\{H_l\}} \tag{2}$$

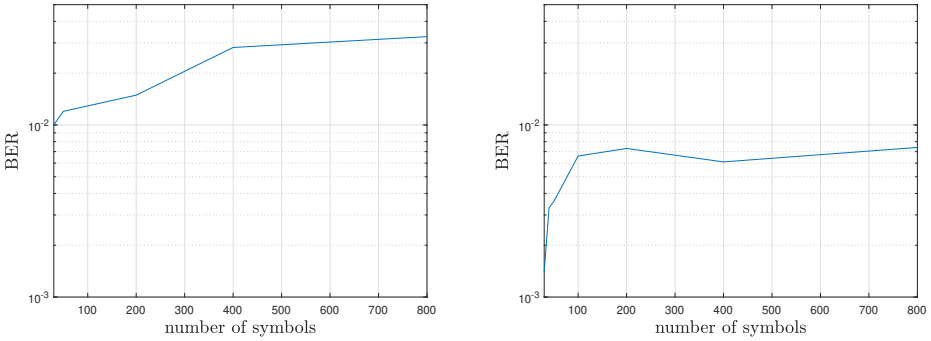
Figure 5 illustrates the effect of the channel equalization on the phase offset when simple Block-Training is applied on a data-stream of 20 OFDM symbols and a training interval of 10 symbols.
Continuous phase tracking reduces the BER in most of the cases, however for a small number of data symbols one training OFDM symbol sent at the beginning of the stream can be sufficient. Figure 6 shows the BER plots for increasing number of data symbols and with $N = 1024$. The first picture shows how the BER increases when no phase-tracking is performed at all. If Viterbi-Viterbi is applied, the BER is



(a) Received symbols (b) Received symbols with phase correction

Figure 5: Phase correction

in general lower and after an initial increase, it becomes almost constant. In order to set a maximum number of symbols that can be transmitted without having to track the phase, a suitable threshold value of the BER must be defined. This threshold value of course depends on the application, for example, for a BER threshold of 10^{-2}, phase tracking can be avoided from transmissions of less than 30 symbols.



(a) Without phase tracking (b) With phase tracking

Figure 6: BER dependence on phase tracking

3.4.1 Training sequence

The choice of the training symbol must fulfill the following considerations in terms of estimation error: Being the received symbol affected by the channel noise, Y_l can be expressed as

$$Y_l = H_l T_l + W_l$$

where W_l is the noise.
Thus the channel estimated noise is

$$\begin{aligned} \mathbb{E}\{|\hat{H}_l - H_l|^2\} &= \mathbb{E}\{|\frac{Y_l}{T_l} - H_l|^2\} \\ &= \mathbb{E}\{|H_l + \frac{W_l}{T_l} - H_l|^2\} \\ &= \mathbb{E}\{|\frac{W_l}{T_l}|^2\} = |T_l^{-1}|^2 \sigma_W^2 \end{aligned} \tag{3}$$

where σ_W is the noise variance.
From eq.(3) it can be deduced that a larger training coefficients' magnitude would result in a smaller

error. However, the magnitude of the training coefficients must comply to the power constraint of the system, that is $\frac{1}{N}\sum_l |T_l|^2 = 1$. Furthermore, a second constraint arises when minimizing the average estimation error $\frac{1}{N}\sum_l |\sigma_W T_l^{-1}|^2$. This is achieved by imposing the same $|T_l|$ for all sub-carriers.

As for the phase of the training symbol, it must take the form of a pseudo random sequence. This is because if the phases combine constructively the system would exhibit an high PAPR (Peak-to-average Power ratio). In this scenario, the transmitter's power amplifier would struggle to generate such a high power signal.

To fulfill all the above conditions, an uniformly-distributed, BPSK-modulated random sequence was chosen as training symbol.

3.5 Demapping

The last step consists in the parallel to series conversion followed by the demapping operation. Once the payload data is corrected according to eq.(2), it is decoded from QPSK to binary according to a hard decision between the possible symbols. To do so, the quadrants of the complex plane are considered as decision regions and the classification of the symbols is made by analyzing the signs of their imaginary and real parts.

4 Acoustic channel

From the estimated channel transfer function it is possible to derive the impulse response of the channel. The impulse response depicted in Figure 7 was derived from two block-training sequences sent in $N = 1024$ sub-carries with $f_{spacing} = 5$ Hz. It can be seen that the main peak has nonzero amplitude on its left side, due to the several filtering operations that were applied along the system chain. The audio system has indeed an unavoidable passive low-pass filter which, together with the low-pass filter applied just after down-conversion, is convoluted with the transmitted signal causing a short time delay that could be compensated through a slight mis-synchronization. Even if this delay contributes to the delay spread, it was decided to not consider it and thus to set the location of the main peak as the zero reference. With this consideration the obtained delay spread of the channel is $t_{ds} \approx 0.7$ ms, which was computed on the 97% decrease in amplitude from the mean peak. With the system sampling frequency f_s, t_s corresponds to a channel length $L = 32$ taps.

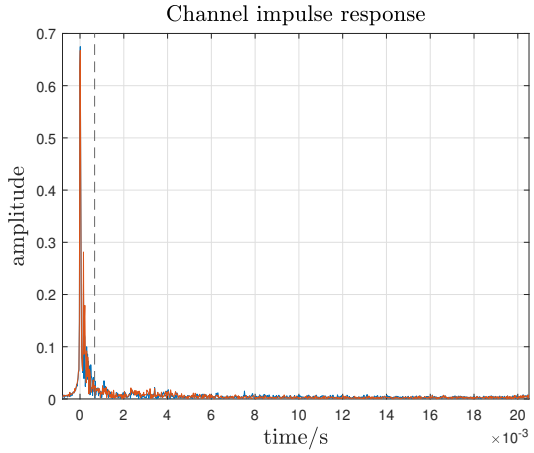


Figure 7: Channel impulse response

The magnitude and phase variation of the channel over the signal bandwidth is shown in Figure 8 where the magnitude is normalized by its maximum value. The magnitude and phase are affected by the system low-pass filters already mentioned, hence the magnitude takes its maximum value around the carrier frequency and decays for lower and higher frequencies, with a maximum variation of ≈ 45 dB (an

isolated case presents at 9009 Hz were the magnitude reaches −66 dB). Besides the effect of the filters, at higher frequencies the channel exhibits larger fluctuations.

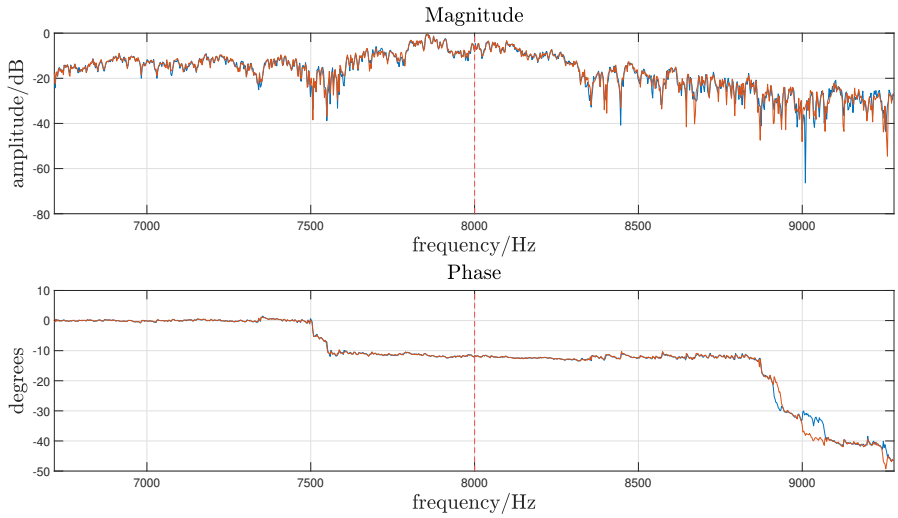


Figure 8: Magnitude and phase of the channel

Computing the ratio between the received symbols and the transmitted ones over a data stream of 200 symbols, the magnitude and phase time variations depicted in Figure 9 were obtained. The plots show the behaviour for the lower, middle and higher frequency sub-carriers: 6.17 kHz (blue), 8 KHz (red), 9.28 kHz (yellow). It can be immediately seen that at the higher frequency the channel undergoes greater magnitude and phase fluctuations over time while for the mid-low frequencies they are rather constant. The behaviour is in general different among the sub-carriers.

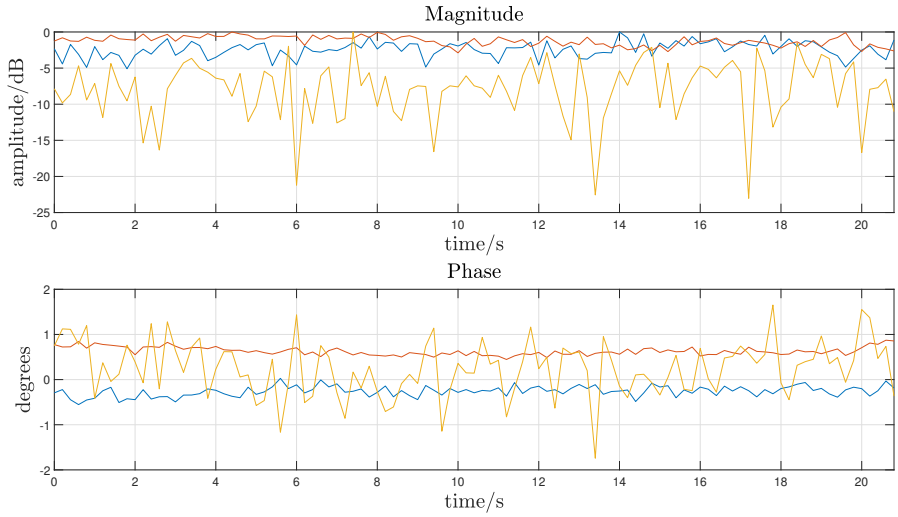


Figure 9: Magnitude and phase time variations of the channel

5 Efficiency and Design Trade-offs

The efficiency of the system can be computed as follows:

$$\eta = \frac{N}{N_G + N}$$

where N is the number of sub-carriers used to send the data and N_G the number of sub-carrier used for the transmission of the cyclic prefix.
Increasing N or decreasing N_G would result in an higher efficiency, however, both methods have drawbacks and a trade-off between the two should be considered.

5.1 Cyclic Prefix length

Table 1 and Figure 10a summarize the results obtained in terms of BER and efficiency for $N = 1024$ and different Cyclic Prefix lengths N_G. If $\eta = 1$, the BER reaches its maximum since the system cannot prevent ISI without any CP symbol. For increasing N_G the BER decreases, in particular at $N_G = L$, where $L = 32$ is the channel length derived in section 4, the BER_{32} is equal to 0.02 and the efficiency is $\eta = 96.97\%$. It can be observed that for $N_G = 16$ the BER_{16} is greater than BER_{32} by only 0.0004, hence a reasonable choice could be setting $N_G = 16$ to preserve a higher efficiency. For $N_G \geq 512$ the BER stops decreasing and it becomes nearly constant. Indeed, even if ISI is avoided, the effect of the channel noise cannot be reduced with longer Cyclic Prefix symbols.

Efficiency and BER vs CP length		
N_G	BER	η
0	0.0721	1.0000
8	0.0450	0.9922
16	0.0204	0.9846
32	0.0200	0.9697
64	0.0186	0.9412
128	0.0172	0.8889
256	0.0133	0.8000
512	0.0112	0.6667
640	0.0102	0.6154
768	0.0098	0.5714
896	0.0102	0.5333

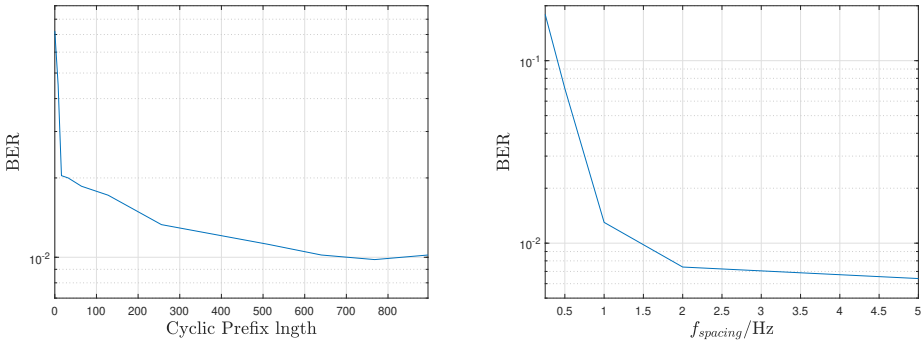
Table 1: Effect of Cyclic Prefix length on the efficiency and BER

5.2 Number of sub-carriers

Table 2 and Figure 10b summarize the results obtained in terms of BER and efficiency for $N_G = 512$, $BW_{BB} = 2050$ Hz and different total number of sub-carriers N. In order to transmit more sub-carriers within the same bandwidth, it is necessary to decrease their spacing in frequency. This makes the symbols more sensitive to frequency offsets and thus Inter Carrier Interference problems might occur. In the figure, indeed, it can be seen that the BER decreases for larger spacing between the sub-carriers.

Efficiency and BER vs CP length			
$f_{spacing}$	N	BER	η
0.25	16384	0.1793	0.9697
0.5	8192	0.0704	0.9412
1	4096	0.0130	0.8889
2	2048	0.0074	0.8000
4	1024	0.0064	0.6667

Table 2: Effect of $f_{spacing}$ on the efficiency and BER



(a) BER vs Cyclic Prefix length N_G

(b) BER vs $f_{spacing}$

Figure 10: BER dependence on N_G and $f_{spacing}$

6 Conclusions

In this work, the performance of a OFDM system in terms of BER was tested, analyzing the trade-offs between cyclic prefix length, number of sub-carriers and type of training to perform a proficient channel estimation and equalization. A particular attention has been given to the study of the channel's impulse response.

By observing the OFDM transceiver's main blocks and the results, it is straightforward to see how this scheme makes a very efficient use of the available spectrum, allowing high data rates without inter-symbol interference problems, and how this system is computationally efficient thanks to the use of the FFT algorithm to shape the OFDM symbols. It thus results clear how powerful this type of modulation technique is and why it became so popular for wide-band digital communication.