UNIVERSITAT POLITÈCNICA DE CATALUNYA

DATA TRANSMISSION THROUGH RADIO OVER FIBER LINKS

Degree Thesis
Submitted to the Faculty of the
Escola Tècnica d'Enginyeria de Telecomunicació de Barcelona
Universitat Politècnica de Catalunya
by

Edison Gabriel Zaragosin Bautista

In partial fulfillment of the requirements for the degree of Bachelor's Degree in Telecommunications Technologies and Services Engineering

Supervisor: Maria Concepción Santos Blanco

Barcelona, January 2023

Abstract

Radio-over-Fiber (RoF) systems applications have increased lately due to the fast wireless connection need. That's the reason why RoF market size is continuously growing according to *Business Research* study [1] .

In this thesis a RoF communication system is explained and analyzed by software and a radiofrequency (RF) system by experimentation. First one is carried out through MATLAB simulation while the second is carried out assembling the mentioned system. To do so, SMW200A vector signal generator and N9010B vector signal analyzer are used, finally bit error rate and error vector magnitude experimentally obtained are compared with simulation results.

Resum

L'aplicació dels sistemes RoF s'ha incrementat ultimament a causa de la necessitat d'una connexió sense fils més ràpida. Aquesta és la raó per la qual el mercat dels sistemes RoF està en creixement continu d'acord amb un estudi de *Business Research* [1].

En aquest projecte s'explica i s'analitza un sistema de comunicacions radio sobre fibra via simulació i un sistema de radiofreqüència de manera experimental. El primer es duu a terme utilitzant MATLAB, mentre el segon es duu a terme construint el sistema esmentat. Per fer-ho s'utilitza un generador de senyals vectorial SMW200A i un analitzador de senyals vectorial N9010B, finalment es compara la taxa d'error de bit i el vector magnitud derror obtinguts de forma experimental amb els resultats de la simulació.

Resumen

La aplicación de los sistemas RoF se ha incrementado ultimamente debido a la necesidad de conseguir una conexión inalambrica más rápida. Esta es la razón por la que el mercado de los sistemas RoF está en continuo crecimiento de acuerdo a un estudio de *Business Research*[1].

En este proyecto se explica y se analiza un sistema de comunicaciones RoF via software y un sistema RF de forma experimental. El primero se lleva a cabo utilizando MATLAB, mientras el segundo se lleva a cabo construyendo un dicho sistema, para hacerlo se utiliza un generador de señales vectorial SMW200A y un analizador de señales vectorial N9010B, finalmente se compara la tasa de error de bit y el vector magnitud de error obtenidos de forma experimental con los resultados de la simulación.

Dedication

My thesis is dedicated to my family but specially to my mother for always supporting me through the whole career.

I'd also like to dedicate my thesis to Maria Jover for changing my life in a positive way, thank you for being a reason to continue every day. I really thank God for allowing me meet someone like you. I'll always have the best wishes for you and I'd always choose you as you already know.

Acknowledgments

First of all, I'd like to thank God for permitting me complete this career and keeping me humble to face the most difficult challenges.

Secondly, I'd like to thank Maria Santos for trusting in me to carry out this thesis, I also thank her for supporting me through the whole process with theoretical and experimental sessions. My thanks also go to Antoni Barlabé for helping me in the experimental part of the thesis.

Finally, I'd like to thank to my family for keeping me motivated when things went wrong, without their help it would has not been possible.

Revision history and approval record

Revision	**Date**	**Purpose**
0	22/09/2022	Document creation
1	21/10/2022	Document revision
2	22/01/2023	Document revision

DOCUMENT DISTRIBUTION LIST

Name	**e-mail**
Edison Gabriel Zaragosin Bautista	
Maria Santos Blanco	
Antoni Barlabé	

Written by:	
Date	22/09/2022
Name	Edison Gabriel Zaragosin Bautista
Position	Project Author

Reviewed and approved by:	
Date	22/01/2023
Name	Maria Santos Blanco
Position	Project Supervisor

Index

List of Figures

List of Tables

List of Acronyms

- PRBS: Pseudorandom binary sequence
- Nbits: Bits number
- BR: Bit rate
- Tb: Bit time
- Ts: Symbol time
- QAM: Quadrature amplitude modulation
- BER: Bit error rate
- EVM: Error vector magnitude
- AWGN: Additive white Gaussian noise
- SR: Sample rate
- B: Bandwidth
- RF: Radiofrequency
- RoF: Radio over fiber
- MZM: Mach Zehnder modulator
- BpS: Bits per symbol
- M: Modulation index
- ISI: InterSymbol interference
- Nsa: Samples number
- Nsasy: Samples per symbol number
- Nsy: Symbols number
- FIR: Finite impulse response
- RC: Root cosine
- RRC: Root raised cosine
- NRZ: Non return to zero
- RZ: Return to zero
- FFT: Fast Fourier Transform
- PSD: Power spectrum density
- β: Roll-off factor
- η_B: Bandwidth efficiency

- VSG: Vector signal generator
- VSA: Vector signal analyzer
- fc: Clock frequency
- IVI: Interchangeable virtual instrument
- VISA: Virtual instrument software architecture
- REF: Reference signal
- PP: Push pull
- V: Electrical signal
- V_b: Bias voltage
- ER: Extinction ratio
- E_{in}: Input optical electric field
- P_{in}: Input power
- E_{out}: Output optical electric field
- P_{out}: Output power
- V_π: Normalization voltage
- θ_B: Normalized bias voltage
- m: Optical modulation index
- P_C: Carrier power
- P_{SB}: Side-band power
- D: Chromatic dispersion parameter
- L: Optical fiber length
- $\beta(\omega)$: Constant propagation
- λ: wavelength
- c: Speed of light
- I_{PD}: Photocurrent
- σ_{th}: Thermal noise standard deviation
- σ_{sh}: Shot noise standard deviation

1 Introduction

The topic of the present work is the RoF system operation mode. Its functioning depends on system structure, it can be said that, in general terms, it is formed by RF system and an optical channel, both can be implemented in a minor scale, i.e. using accessible software and equipment, as presented in this project.
The implementation of a RoF system aims computing the bit error rate (BER) and error vector magnitude (EVM) to evaluate the system performance. To achieve the objective the following requirements need to be fulfilled:

- Provide an analysis of the RoF system features, specifying its main components with their characteristics and limitations.
- Develop a simulation program that allows to verify and to quantify the impact of every parameter in the system performance.
- Define an experimental set up which may be used to asses the quality of RoF system.

Last requirements can be accomplished by restudying some subjects explained along career, self-study and with the support of the thesis supervisor.

1.1 Project Workplan

At project simulation part, a template code has been used for BER and EVM computations, this template is based on a previous paper [8]. Regarding experimental part, instrument vendors code has also been used as a template to send/receive signal.
The project workplan has suffered slight changes caused by equipment limitations and lack of information sources during experimental part. The Gantt diagram at figure 1 shows workplan tasks and the project total progress, where green labels represent completed tasks and blue label is the future work.

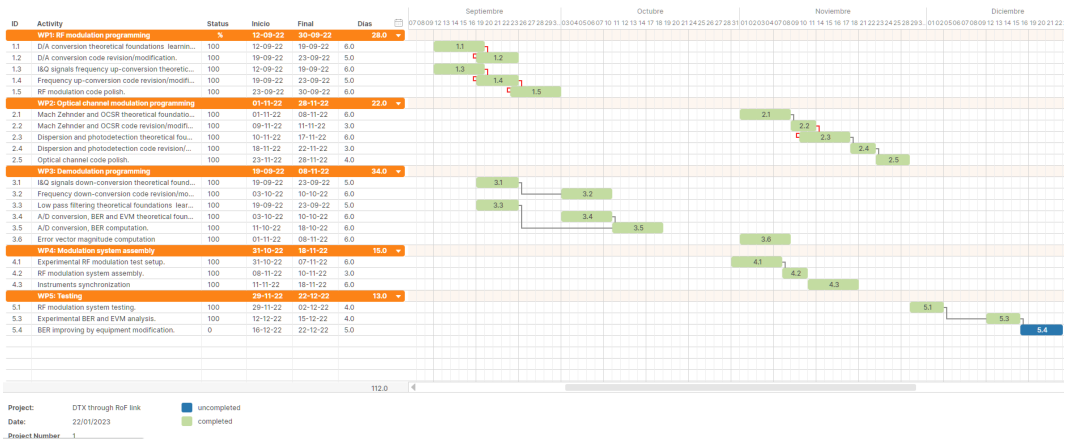


Figure 1: Gantt diagram

1.2 Thesis chapters guide

To facilitate user reading, a little summary for every thesis chapter is presented.

- **RF communications system:** In this chapter the operation mode and implementation through code of each system element are explained.
- **Radio-over-fiber communications system:** In this chapter the operation mode and implementation through code of optical components added to RF system to turn it into RoF are explained.
- **Experimental part:** In this chapter, the RF system assembly is explained in detail.

The remaining chapters talk about the project budget, conclusions and future work.

2 State of art

Radio-waves are a kind of electromagnetic radiation, its spectrum range goes from 3 Khz to 300 GHz [2]. This type of waves are used to provide multiple services such as message sending/receiving, radio services, etc. They are also important in many applications such as military tasks or even act as a complement for other study fields [3]. Radio-waves are the main tool of the radio frequency communication systems and they are widely used in these days. However, in recent years internet traffic has increased, it is translated in a huge amount of users but RF systems were not ready for it, so that they evolved into RoF systems, this evolution allows higher carrying capacity due to fiber networks great bandwidth [4].
A radio-over-fiber communication system is in charge of modulating the light through a radio signal and then transmitting it over optical fiber [5]. This way of transmission gives some advantages over RF systems transmission, like reduction of loss at transmission and sensitivity reduction to noise [6].
In order to provide the latest advances in this topic, a list of the recent studies below is presented. A study from *International Journal of Engineering Science and Technology* suggests that the operation mode of a RoF system can be optimized by changing the position of the biasing point from the typical quadrature biasing but it is not the only way to do optimize it, it can also be done by replacing the Mach Zehnder Modulator for an electro-absorption modulator. Last modulator can turn a non-linear modulated laser response into linear, it makes worthy the using of RoF systems in long distance communications [7].
Finally, it is worth mentioning that there is another kind of RoF systems called *intermediate frequency-over-fiber* comms systems, they modulate light using the intermediate frequency of a radio signal, that frequency used to be lower than 10 [GHz] [5].

3 RF communications system

3.1 Introduction

The main purpose of the chapter is to explain how an RF comms system works and determine how BER and EVM are obtained through MATLAB code, detailing functionality of each block and how it is built, theoretically and mathematically. Specifically, the RF system is going to be analyzed from bits generation passing through digital-analog converters, filters, modulation/demodulation processes and finally to bits recovering.

3.2 RF comms system description

As initial idea, an RF communications system can be seen as a simple block where bits are sent and then are recovered, but inside it there are many blocks that make possible the communication process.

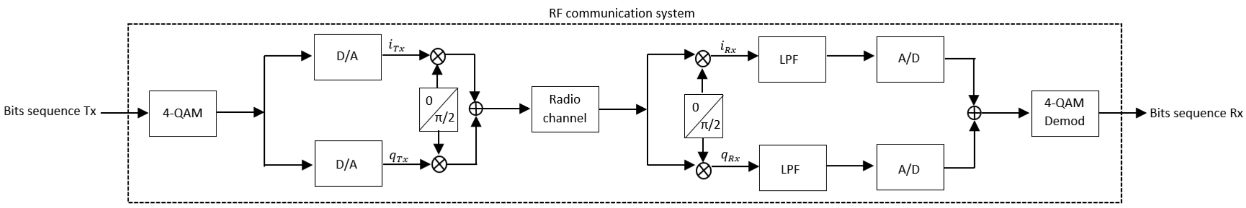


Figure 2: RF comms system blocks diagram

The blocks diagram shown at figure (2) begins with a pseudo-random binary sequence (transmitted bits), it acts as a message, then these bits become symbols through an M-QAM modulation. Once the modulation is completed, those symbols pass to digital/analog converters (DAC) where the inter-symbol interference (ISI) is minimized, thanks to its internal filter, and a base-band analog signal is obtained, then this signal is multiplied by a carrier signal with an specific frequency (radiofrequency), as a result a pass-band signal is obtained (frequency up-conversion), then it is transmitted through the radio channel to receiver side. At receiver, the pass-band signal is multiplied again by the same carrier signal as before in order to recover the transmitted base-band analog signal. The pass-band signal in frequency domain shows some undesired harmonics generated by the previous multiplication, that's the reason why a low-pass filter is needed, the filter is applied to isolate the base-band signal frequency component from the harmonics (frequency down-conversion). The resulting signal passes through the analog/digital converter (ADC) thus obtaining the symbols. The next step is to demodulate the symbols to get the set of received bits. Finally, BER and EVM are computed, they are essential when evaluating the radio channel performance [9].

3.3 RF comms system initial parameters

While implementing RF system in code, some parameters have specific values chosen by the user, these values depend on the project final goal and how it is going to be carried out.

First of them is the amount of bits needed ($Nbits$), it depends on the BER desired, the higher amount of bits is, the more bit errors appear for the same BER value. However, bit errors shouldn't be reduced to zero because it's needed to get a result statistically relevant, that's the reason why there are a minimum number of bits to be considered. Regarding communications channel, it is considered acceptable when BER is around 10e-3, to obtain the last value, the transmitted bits number should be around 10e4 so that BER is computed with at least 10 erroneous bits [11].
The next parameter to be considered is the bit rate (BR), it is in charge of controlling the temporary distance of the bits with each other, it also has an additional feature, the higher bit rate is, the higher the bandwidth is $\left(B = \frac{BR}{BpS}\right)$. Bandwidth also depends on modulation type, in this case, 4-QAM has been established, so bits per symbol is equal to 2 ($BpS = log_2 Mod_{index}$). The symbols number (N_{sy}) is affected by the previous parameters due to is defined as the relation between $Nbits$ and BpS $\left(N_{sy} = \frac{N_{bits}}{BpS}\right)$, hence, it is directly proportional to N_{bits} and inversely proportional to BpS. Radio-frequency value (f_{RF}) is established by the radio application, it is directly involved in the sample rate (SR) choosing through Nyquist-Shannon sampling theorem [10].

$$SR > 2f_{max} \tag{1}$$

In the equation (1), f_{max} is the maximum frequency reached by the signal centered at radio-frequency (f_{RF}). It is known that the maximum frequency is defined as the following way:

$$f_{max} = f_{RF} + \frac{B_{signal}}{2} \tag{2}$$

$$B_{signal} \simeq \frac{1}{T_{sy}} = \frac{1}{BpS \cdot T_b} = \frac{BR}{BpS} \tag{3}$$

$$f_{max} = f_{RF} + \frac{BR}{2BpS} \tag{4}$$

Finally, Nyquist-Shannon inequality is as follows:

$$SR > 2\left(f_{RF} + \frac{BR}{BpS}\right) \tag{5}$$

The theorem fulfillment allows to reproduce faithfully the sampled signal during the whole simulation process.

3.4 IQ signal generation and DAC parameters

As mentioned at 3.2, an RF communications system has many stages formed by many blocks, the scheme at figure 3 shows the first stage to be analyzed.

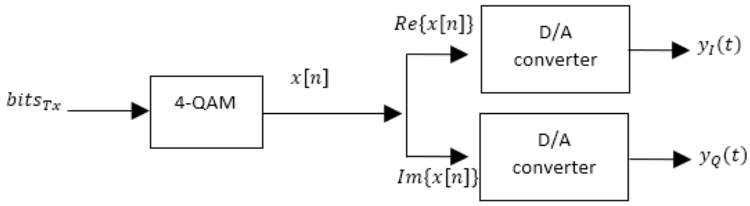


Figure 3: RF comms system first stage

The IQ signal is generated by modulating the transmitted bits sequence, 4-QAM is the modulation chosen as previously mentioned. At the coding, bit sequence modulation can not be computed directly, it requires some previous vector processing. The first step is about cutting bit sequence to get an integer number of symbols, i.e. limit the bit sequence vector to its last highest integer number, then there is a bits/decimal conversion, it is made turning the *cut bits sequence* into a matrix, then pairing elements from each row and building integers, `reshape()` and `bi2de()` functions are in charge of that. Finally, the resulting vector is modulated with 4-QAM, it is made with `qammmod()` function. This process is illustrated in figure 4.

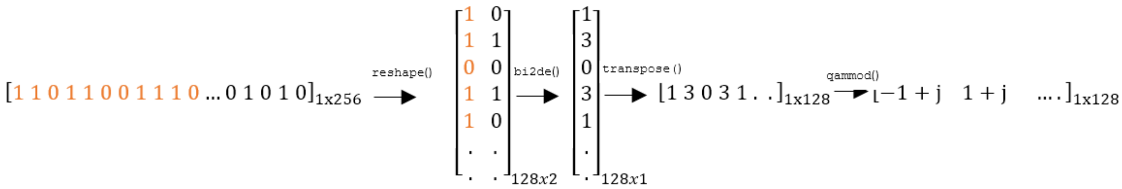


Figure 4: IQ signal generation process

The modulated signal corresponds to symbols vector, symbols are represented through complex numbers, so they can be split in two components, in phase ($\Re$) and in quadrature ($\Im$), both components pass through DAC.

Regarding DAC parameters, at the coding, the most relevant is the number of samples per symbol (N_{sasy}) due to the time resolution of the simulation, i.e. DAC Bandwidth, is determined through it.

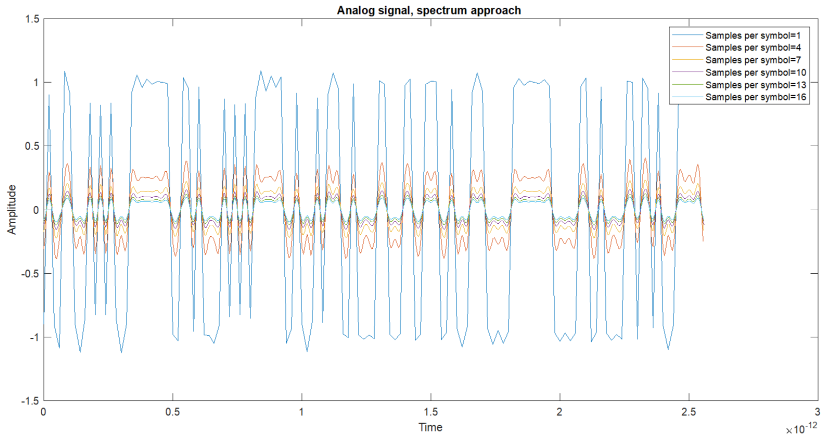


Figure 5: Analog signal resolution variation for many N_{sasy} values

As it can be seen at figure 5, for the same bits sequence, the resolution of the resulting analog signal gets worse when samples per symbol are lower, i.e. signal peaks look like rhombus or with some distortion, instead of peaks. The explanation is given in the sample time formula, the higher N_{sasy} is, the lower sample time (T_{sa}) is, hence, more samples are used and better signal resolution is obtained.

$$T_{sa} = \frac{1}{SR} \tag{6}$$

$$T_{sa} = \frac{TW}{N_{sa}} \tag{7}$$

$$T_{sa} = \frac{N_{bits}}{N_{sasy} N_{sy} BR} \tag{8}$$

$$T_{sa} = \frac{BpS}{N_{sasy} BR} \tag{9}$$

The ISI is produced inside DAC, i.e. when symbols signal goes through DAC, so that a filter is needed in order to minimize it as mentioned at 3.2, in this case, the most appropriate filter is the one called raised cosine (RC).

3.5 RC filter

In the previous section, the RC filter is introduced, this section explains its structure, main features, how it works and why it is chosen above other filters.

The RC filter satisfies *Nyquist criterion for zero ISI* and is usually implemented as finite impulse response (FIR) filter. The Nyquist criterion for Zero ISI can be easily understood when plotting symbols in time domain.

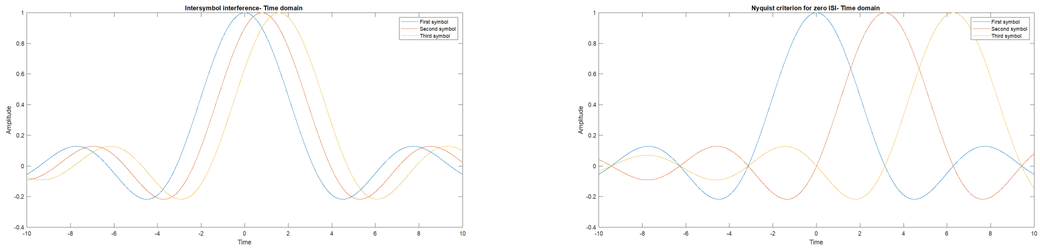


Figure 6: Nyquist criterion for zero ISI

In the figure 6 it is shown that there is no ISI when *Nyquist criterion* is applied. Each peak represents the symbol ideal sampling location, just at the top of the peak, it can be noticed that the adjacent impulse responses have zero crosses at this point.
It's worth to say that rectangular pulses with abrupt transitions, avoid ISI easily but they have infinite bandwidth in frequency, so they are not feasible, on the other hand, *sinc* pulses or similar shapes like RC are a better option due to the finite bandwidth consumption [12].

Regarding FIR filters, these are filters whose main feature is having impulse responses with a finite number of terms [13]. This kind of filters have many interesting properties, some of them are the following ones: feedback is not required, give a lot of stability and allow linear designs [15].

3.5.1 RRC filter: implementation and math parameters

As previously mentioned, the main goal is getting a RC filter in order to take profit of its features, to do so, two root raised cosine filters (RRC) must be implemented in transmission and reception in order to combine both frequency responses and finally getting the desired RC frequency response [16].

$$H_{RC}(f) = H_{RRC_{Tx}}(f) \cdot H_{RRC_{Rx}}(f) = \sqrt{H_{RRC}(f)} \cdot \sqrt{H_{RRC}(f)} \tag{10}$$

It's also relevant to explain the parameters of the RRC filter in order to understand its equation in time and frequency domains. Roll-off factor (β) and symbol time (T_s) are the only parameters to characterize these filters. The first one, is in charge of measuring the bandwidth excess of the filter, a smaller rolloff factor produces narrower bandwidth, on the other hand, a smaller symbol time increases it, they are both set by the user [16].

$$B = \frac{1+\beta}{2T_s} \tag{11}$$

The RRC filter, mathematically, can be defined through impulse and frequency responses, they are represented by the following equations and with their respective figures [16][8].

$$h_{RRC}(t) = \frac{2\beta}{\pi\sqrt{T_s}} \frac{cos\left(\frac{(1+\beta)\pi t}{T_s}\right) + \frac{T_s}{4\beta t} sin\left(\frac{(1-\beta)\pi t}{T_s}\right)}{1 - \left(\frac{4\beta t}{T_s}\right)^2} \tag{12}$$

$$H_{RRC}(f) = \sqrt{\frac{1}{2}\left[1 + cos\left(\pi\frac{T_s}{\beta}\left(|f| - \frac{1-\beta}{2T_s}\right)\right)\right]} \tag{13}$$

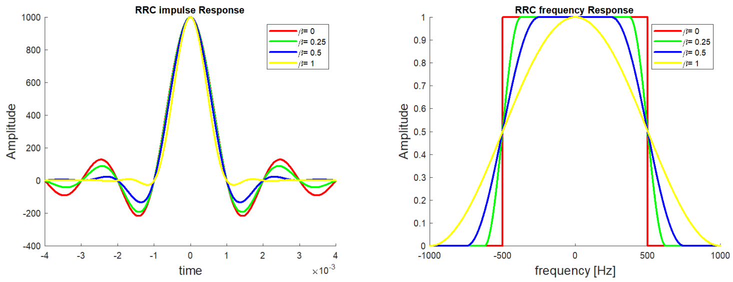


Figure 7: RRC filter impulse and frequency responses

According to figure 7, it is seen how roll-off factor changes the pulse shape in time and frequency domains, in time, it can be seen that zeros appear every symbol time multiple [17].

3.6 DAC from frequency domain

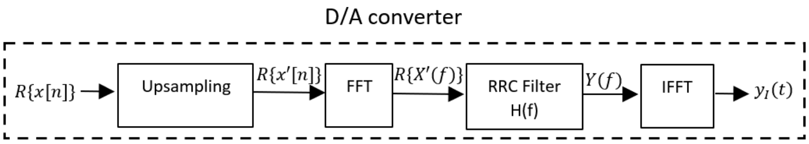


Figure 8: DAC from frequency domain- Blocks scheme

As shown in figure 8 the first step to carry out this conversion, from frequency domain, is making an up-sampling to modulated signal ($x[n]$) by using `upsample()` function in order to increase the sample rate, by doing so, the resulting signal will have a better resolution, resulting signal is called $x'[n]$. The next step is about computing the Fast Fourier Transform (FFT) by using `fft()` function, thus getting a resulting signal in frequency domain ($X'(f)$), next, a RRC filter frequency response is defined through the equation (13). Once the filter is ready, it is multiplied by the resulting signal FFT thus obtaining $Y(f)$. As final step, just remains to acquire the analog signal, to do so it's just a matter to compute the Inverse Fourier Transform of the filtered signal (IFFT[$Y(f)$]) using `ifft()` function. The result of this conversion is shown at figure 9 as an example.

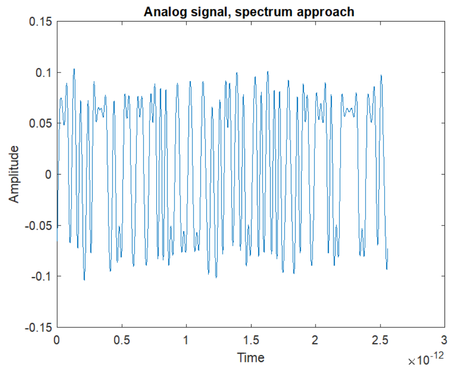


Figure 9: Analog signal from Spectrum Approach- Nbits=$2{\cdot}10^{18}$, Nsasy=64

3.7 DAC from time domain

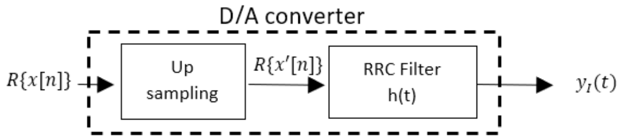


Figure 10: DAC from time domain- Blocks scheme

This conversion type can be carried out by convolution, using `conv()` function, or directly using `upfirdn()` function.
For the first case, it's just a matter to upsample the input signal and then convolve it with the RRC impulse response. The second case follows almost the same process, the main difference is that signal is upsampled at the same time as filter is applied, `upfirdn()` function is in charge of it. RRC filter, in time domain, can be created with `rcosdesign()` function or using its time domain equation previously defined at (12). If `rcosdesign()` function is used, many parameters need to be taken into account, these are the following ones: roll-off factor, filter span, samples per symbol, and pulse shape. The most important parameter is the filter span and it'll be analyzed in the next section, but there is also a peculiarity related to pulse shape parameter. According MATLAB help this function has two modes: normal and square root, however, when the fourth parameter is not specified `rcosdesign()` assumes that the pulse shaping is set as a square root.
Finally, a DAC comparison is presented in the following figure.

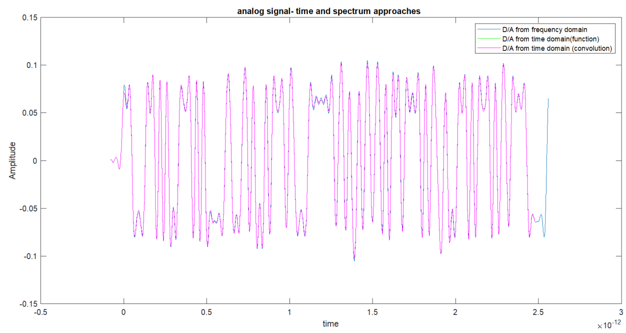


Figure 11: Resulting analog signal, frequency and time approaches

In the figure 11, slight variations between the analog conversions are appreciated, those variations are caused by RRC filter span absence during conversion from frequency domain, a more detailed explanation is given in the next section.

3.8 RRC span presence/absence impact on time and frequency approaches

It's important to remark the differences between the three DAC types because none of them gives exactly the same result. The first case to be analyzed are DAC from time domain, it can not be noticed that their shapes don't exactly match at the same points, despite of that, it can be ensured through `isequal()` function that there are "micro-gaps", i.e. phase shifts, between them. If `upfirdn()` function is carefully analyzed, those micro-gaps appear due to its using, for example two filtered signals but one of them was upsampled and then convolved while the other one is obtained using given function will differ in length and numerically. Those signals will not be completely equal to each other, there will be a minimum error between them, but that's not a significant value and they can be considered as if they were equal, last affirmation is shown at figure 12.

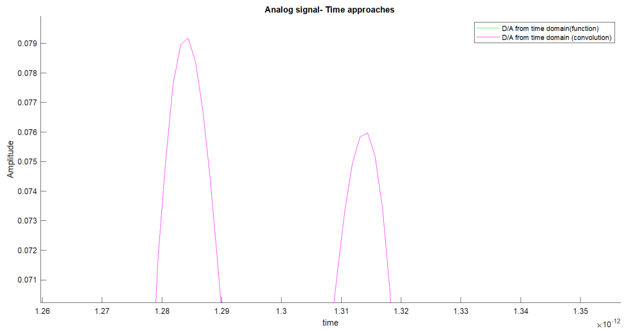


Figure 12: Filter span impact on DAC from time domain

As remarked in section 3.7, the analog signal obtained from frequency domain also does not match with the others. This case is quite different because it is carried out by Fourier transforms. In this case, the RRC filter comes from its frequency equation instead of `rcosdesign()` function, so filter length (span) is not involved. After several tests it has been determined that the bigger filter length is, the more similar three analog signals are. It means that the frequency approach is considering an ideal RRC filter, i.e. infinite filter length, as shown in figure 13. In conclusion it is not going to be useful in practice to get realistic results, however, some comms software like VPI photonics uses it.

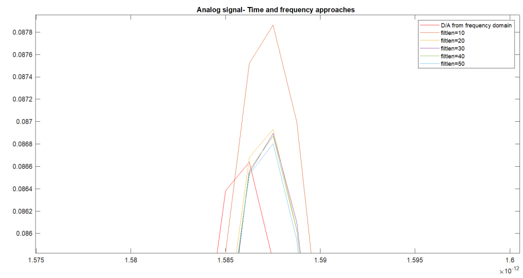


Figure 13: Filter span increasing impact in analog signal

3.9 RRC span variation impact on: BER and DAC

BER is one of the most important parameters when a set of bits are recovered at the end of the communication system.
RRC span, from now on called $filtlen$, represents the number of symbols the filter is made of, it has a lot of influence when BER computing, but so does N_{sasy}. In the first case, for a fixed samples per symbol value ($N_{sasy} = 64$), it can be noticed that the lower $filtlen$ is, the more BER increases as shown at table 1.

RRC filter span ($filtlen$)	BER
8	0.0018
4	0.0028
2	0.012

Table 1: RRC span impact on BER - RoF comms system

The reason why BER improves, i.e. decreases, when $filtlen$ increases is that a higher value of $filtlen$ means a better approximation to ideal response, as concluded at section 3.8.
On the other hand, when N_{sasy} decreases, BER increases much more than the previous case, reaching values close to 0.5 which is even worse when bits recovering. This is because the DAC resolution is controlled by this parameter as explained at section 3.4.
It may be useful to know how many coefficients are going to be needed, they can be computed by the following equation:

$$Filter_{\text{coeff}} = filtlen \cdot N_{sasy} + 1 \tag{14}$$

The filter order is defined by the first sum term in (14), so that it is directly proportional to $filtlen$. It will be demonstrated how much impact has the filter order on digital to analog converters [18].

$$Filter_{\text{order}} = filtlen \cdot N_{sasy} \tag{15}$$

Having a higher filter order has some advantages like the following ones: greater attenuation between bandpass and stopband, narrower transition band and flatter bandpass. There are also some disadvantages like delays, filter instability and an increasing in computation time, experimentally those are also reasons why when $filtlen$ decreases, filter response gets worse, and the result is not the expected one [19].

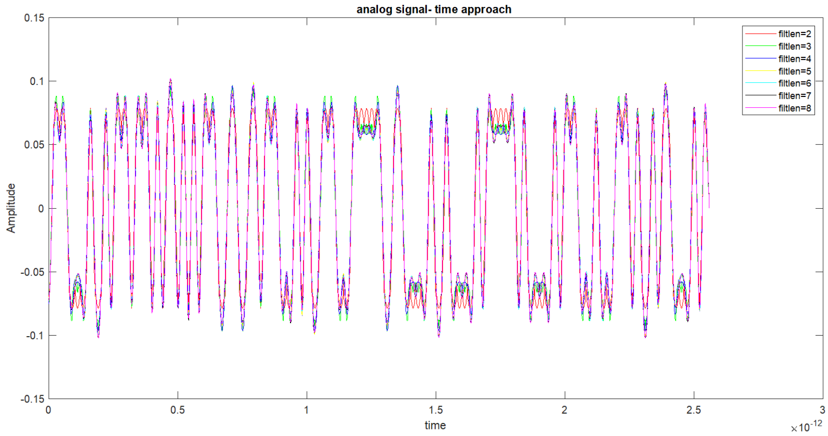


Figure 14: Filter length impact (span) on D/A conversion

As it can be seen in the figure 18, the analog signal is better filtered as filter order increases, i.e. *filtlen* or N_{sasy} increases. This phenomenon can be clearly appreciated for any input signal, the higher filter order is, the better filtered signal is, as shown in figure 15.

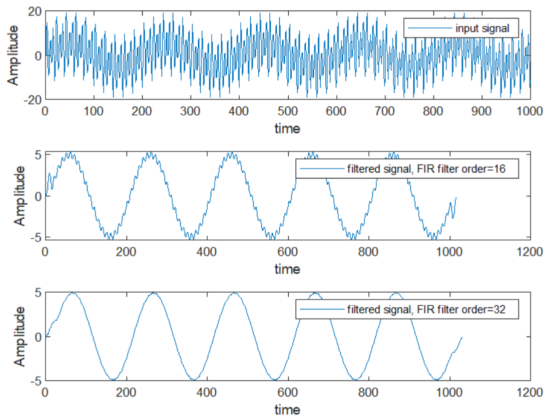


Figure 15: Filter order impact for arbitrary low pass filtered signal

3.10 Frequency up - conversion

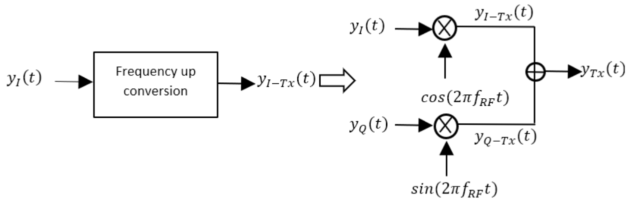


Figure 16: Frequency up-conversion block

Once an analog signal from DAC is obtained, it can be split in two signals *in phase* corresponds to real part called $y_I(t)$, and *in quadrature* corresponds to imaginary part called $y_Q(t)$, both enter to the *frequency up* block, inside it, the first one is going to be multiplied by cosine and the other one by sine, both sinusoidal waves are set to the desired radio-frequency. Once the last process is finished, resulting signals from each branch are added up into one signal called $y_{Tx}(t)$.
The reason why the sinusoidal waves are used is to take profit of their orthogonality property that allows to recover original signal, that recovering is carried out when demodulating signal.

$$y_{Tx}(t) = y_I(t)cos(2\pi f_{RF}t) + y_Q(t)sin(2\pi f_{RF}t) \tag{16}$$

$$Y_{Tx}(f) = \frac{1}{2}Y_I(f - f_{RF}) + \frac{1}{2}Y_I(f + f_{RF}) + \frac{1}{2j}Y_Q(f - f_{RF}) - \frac{1}{2j}Y_Q(f + f_{RF}) \tag{17}$$

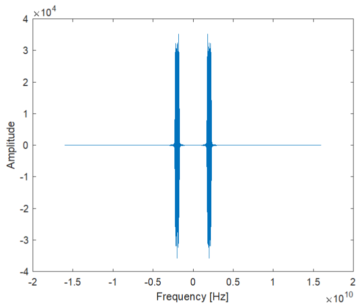


Figure 17: Modulated signal spectrum- Real components

As shown at figure 17, the base-band signal has turned into a pass-band signal as demonstrated at the equation (17).

3.11 Frequency down - conversion

The resulting pass-band signal is transmitted through a radio channel, experimentally, it is implemented with a coaxial wire.

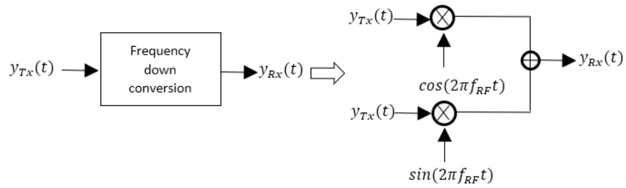


Figure 18: Frequency down-conversion block

To demodulate the received signal, $y_{Tx}(t)$, sinusoidal waves are needed again in order to recover the original signal as explained in section 3.10. Mathematically, it is demonstrated as follows:

$$y_{RxI}(t) = [y_I(t)cos(2\pi f_{RF}t) + y_Q(t)sin(2\pi f_{RF}t)]cos(2\pi f_{RF}t) \tag{18}$$

$$y_{RxI}(t) = y_I(t)cos^2(2\pi f_{RF}t) + y_Q(t)sin(2\pi f_{RF}t)cos(2\pi f_{RF}t) \tag{19}$$

$$y_{RxI}(t) = y_I(t)cos^2(2\pi f_{RF}t) + \frac{1}{2}y_Q(t)sin(4\pi f_{RF}t) \tag{20}$$

$$y_{RxI}(t) = \frac{1}{2}y_I(t) + \frac{1}{2}cos(4\pi f_{RF}t)y_I(t) + \frac{1}{2}y_Q(t)sin(4\pi f_{RF}t) \tag{21}$$

$$y_{RxQ}(t) = [y_I(t)cos(2\pi f_{RF}t) + y_Q(t)sin(2\pi f_{RF}t)]sin(2\pi f_{RF}t) \tag{22}$$

$$y_{RxQ}(t) = \frac{1}{2}y_Q(t) - \frac{1}{2}cos(4\pi f_{RF}t)y_Q(t) + \frac{1}{2}y_I(t)sin(4\pi f_{RF}t) \tag{23}$$

$$y_{Rx}(t) = y_{RxI}(t) + y_{RxQ}(t) \tag{24}$$

$$y_{Rx}(t) = \frac{1}{2}y_I(t) + \frac{1}{2}y_Q(t) + \frac{1}{2}cos(4\pi f_{RF}t)[y_I(t) - y_Q(t)] + \frac{1}{2}sin(4\pi f_{RF}t)[y_I(t) + y_Q(t)] \tag{25}$$

$$|Y_{Rx}(f)| = \frac{1}{2}[Y_I(f) + Y_Q(f)] + \frac{1}{4}[Y_I(f-2f_{RF}) - Y_Q(f-2f_{RF})] + \frac{1}{4}[Y_I(f+2f_{RF}) - Y_Q(f+2f_{RF})] \tag{26}$$

The equation (26) shows that more spectral components have been generated in this process, they appear at RF frequency multiples with a lower amplitude than the main ones as shown at figure 19.

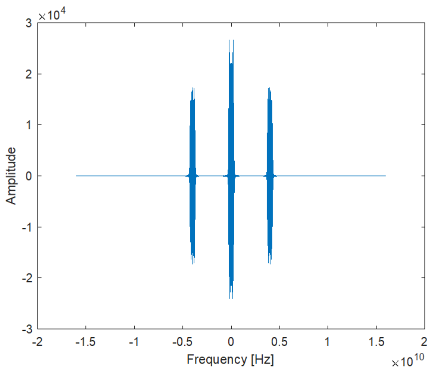
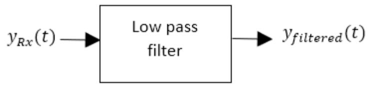


Figure 19: Demodulated signal spectrum- Real components

3.12 Low pass filter effect on demodulated signal



Some harmonics appear and the signal amplitude gets smaller when demodulating the received signal, as demonstrated at (26). To get rid of them, a low pass filter is needed. Mathematically, signal is filtered as follows:

$$LPF(f) = 2 \cdot rect(f) \tag{27}$$

$$Y_{filtered}(f) = |Y_{Rx}(f)| \cdot LPF(f) \tag{28}$$

$$Y_{filtered}(f) = Y_I(f) + Y_Q(f) \tag{29}$$

$$y_{filtered}(t) = y_I(t) + y_Q(t) \tag{30}$$

Where:

$$rect(f) = \begin{cases} 0 & |f| > \frac{B}{2} \\ \frac{1}{2} & |f| = \frac{B}{2} \\ 1 & |f| < \frac{B}{2} \end{cases} \tag{31}$$

The whole low-pass filtering process is represented graphically in the figure 20.

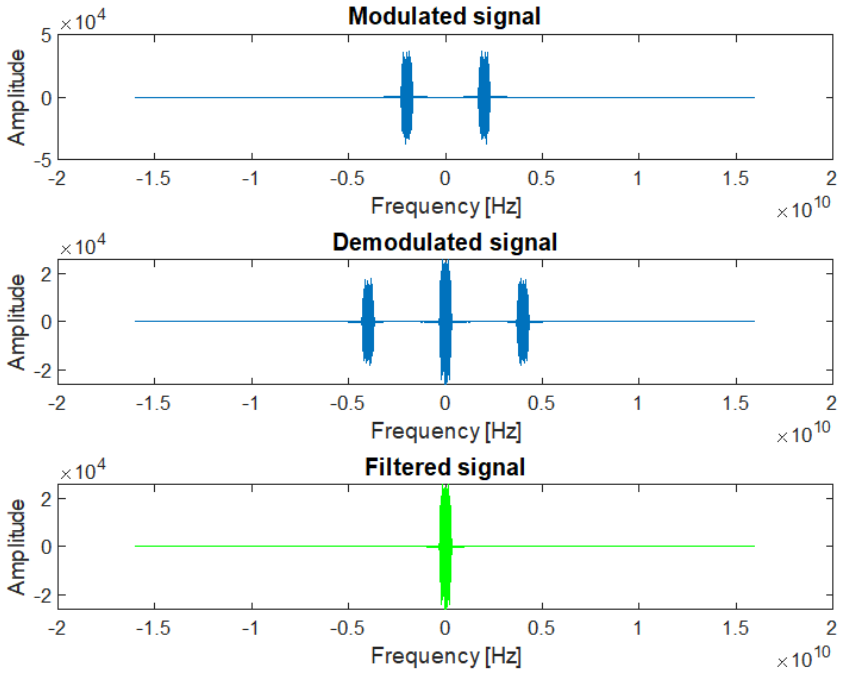


Figure 20: Filtering signal process

3.13 Analog to digital conversion

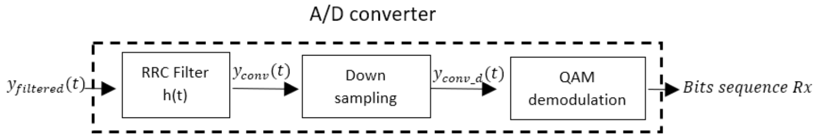


Figure 21: Analog to digital conversion block

This conversion follows the same steps as D/A conversion but in the opposite way. In this case, the first step is passing filtered signal through the second RRC filter, thus obtaining the desired RC response explained at 3.5.1, the next step is down-sampling the signal by using `downsample()` function, in order to undo the up-sampling made at DAC. Finally, the analog signal goes through QAM demodulation block, inside it, symbols are recovered, then they are converted to bits by using the process described at 3.4 but in the opposite direction as shown at figure 22.

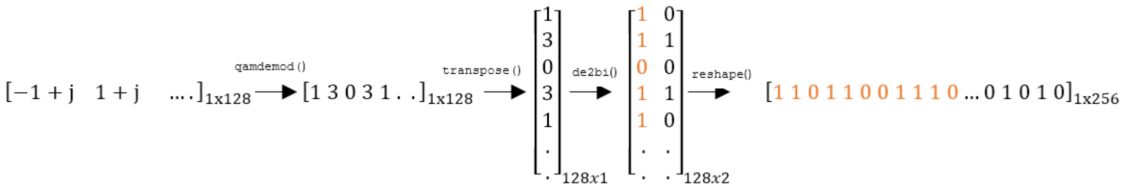


Figure 22: Symbols to bits conversion

3.14 Bit error rate

At the end, the bit error rate is the most relevant parameter to be analyzed in a telecommunication system. BER is defined as the probability of receiving X erroneous bits in a set of N bits, and that's how it's computed in the simulation. Specifically, what has been done in simulation is subtracting the received bits to the transmitted bits in order to identify in which positions bits are different, then those positions are counted, i.e. number of erroneous bits is obtained, finally it's just a matter of dividing it by the total number of bits.

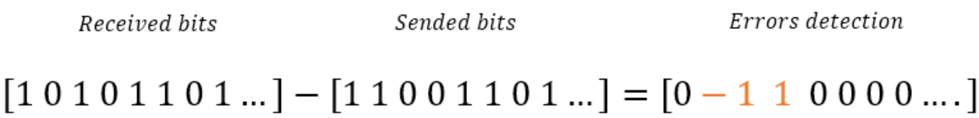


Figure 23: Erroneous bits detection

Mathematically, BER depends on modulation order (M) and signal to noise ratio (SNR) as presented in the following equation [20]:

$$BER = \frac{2\left(1 - \frac{1}{\sqrt{M}}\right) \operatorname{erfc}\left(\sqrt{\frac{3\,\mathrm{SNR}}{2(M-1)}}\right)}{log_2 M} \tag{32}$$

For 4-QAM, last equation is as follows:

$$BER = \frac{\operatorname{erfc}\left(\sqrt{\frac{\text{SNR}}{2(M-1)}}\right)}{2} \tag{33}$$

3.15 Error vector magnitude

This parameter is taken into account when the quality of modulation is evaluated. EVM is defined as the magnitude of a vector located on the complex plane, in this case IQ components plane, in other words, EVM is the result of subtracting received signal vector to reference signal vector [21]. Experimentally, it is used to be computed by vector signal analyzer (VSA) but it also can be computed manually by using vectors properties [20].

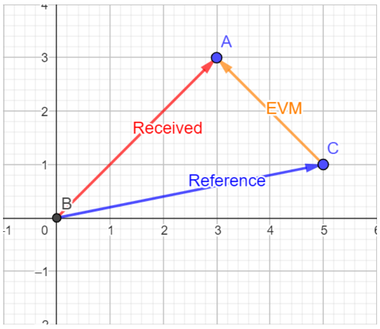


Figure 24: Error vector magnitude definition

$$\text{EVM} = \sqrt{I_{\text{EVM}}^2 + Q_{\text{EVM}}^2} \tag{34}$$

Where:

$$I_{\text{EVM}} = I_{\text{Reference}} - I_{\text{Received}} \tag{35}$$

$$Q_{\text{EVM}} = Q_{\text{Reference}} - Q_{\text{Received}} \tag{36}$$

In MATLAB, it can be implemented easily just using (34) where sent signal acts as a reference signal.

4 Radio-over-fiber comms system

Once RF communications system has been fully explained, the next step will be the adding of the optical components in order to build a RoF comms system. Those all components are part of the optical channel block and they are listed below: optical source, i.e. laser, Mach-Zehnder modulator, photo-diode and an optical fiber. See figure 25.

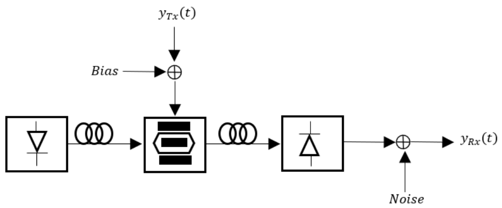


Figure 25: Optical components implementation in RF system

Each optical component listed above is explained in this chapter, from theoretical and programming point of views in order to understand their operation mode and their implementation in the comms system through MATLAB.
RoF scheme at figure 25 can also be represented by blocks as shown at figure 26 .

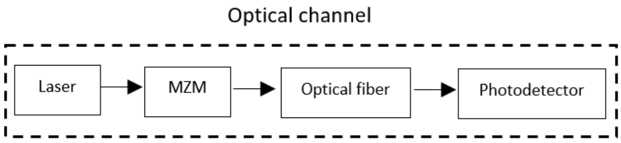


Figure 26: Optical channel blocks diagram

4.1 Laser

The first block to be analyzed is the laser block, obviously, it is the light source of the RoF system, its emitted light is the one which is going to be modulated by an electrical signal (V). Laser signal is equivalent to an optical electric field (E_{in}), in code terms, it's represented jointly to the square root of the Mach Zehnder modulator insertion loss ($\sqrt{L_{MZM}}$) as a constant value because it only acts as the amplitude of the MZM output expression.

4.2 MZM

Mach Zehnder modulator is in charge of optically modulating the RF signal [22]. Mathematically, the operation mode is based in applying the MZM transfer function ($h_{MZM}(t)$) to the field (E_{in}). Theoretically what happens is that the input field is divided in two parts, one for each modulator branch, then an electrical signal is applied in both branches acting as a phase shifter, i.e. it modulates input field in phase.

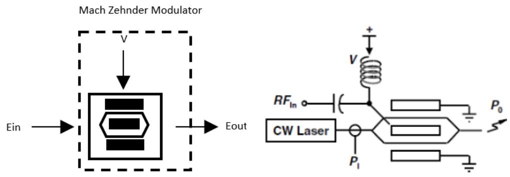


Figure 27: Mach Zehnder modulator block and circuit diagram

MZM has three different configurations: Push Pull (PP), Dual-Electrode (DE) and Dual-Parallel (DP), the first configuration is most common and is the chosen one, its main feature is that electrical signal is the opposite at each branch ($V_1 = V$ and $V_2 = -V$).

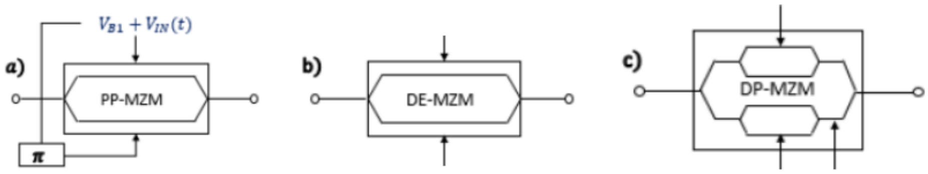


Figure 28: Mach Zehnder modulator configurations [23]

The electrical signal is composed of the RF signal and a bias voltage (V_B), last one is in charge of determining power transfer function working point, this point is used to be located in the middle section of the transfer function, i.e. where function is linear, this point is called *quadrature point*.

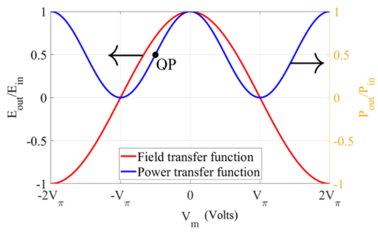


Figure 29: QP: quadrature point [8]

At MZM output both phase-shifted signals are combined into one, this phenomenon causes that the phase modulation turns into an intensity modulation [24]. In PP configuration the output field equation is as follows:

$$E_{out} = \frac{E_{in}}{2\sqrt{L_{MZM}}} \left(e^{j\frac{\pi}{2V_\pi}V} + e^{-j\frac{\pi}{2V_\pi}V} \right) \tag{37}$$

or

$$E_{out} = \frac{E_{in}}{\sqrt{L_{MZM}}} cos\left(\frac{\pi}{2V_\pi}V\right) \tag{38}$$

Where V_π is the voltage difference between the first maximum and first zero of the field transfer function.
Power can also be defined knowing its proportionality relationship with the field, as shown below:

$$P_{out} \sim |E_{out}|^2 \tag{39}$$

$$P_{out} \sim \left|\frac{E_{in}}{\sqrt{L_{MZM}}} cos\left(\frac{\pi}{2V_\pi}V\right)\right|^2 \tag{40}$$

$$P_{out} \sim \frac{P_{in}}{L_{MZM}} cos^2\left(\frac{\pi}{2V_\pi}V\right) \tag{41}$$

$$P_{out} \sim \frac{P_{in}}{2L_{MZM}} \left(1 + cos\left(\frac{\pi}{V_\pi}V\right)\right) \tag{42}$$

The equations (38) and (42) are considering infinite extinction ratio (ER), i.e. an ideal condition, this condition is not recommended when implemented. As ER is defined by the quotient between maximum and minimum power, an infinite ER means that the maximum power is infinite too and this is not true in any case. A Gamma (γ) parameter has been defined in order to take into account a finite ER and its equation is as follows:

$$\gamma = \frac{10^{\frac{ER}{20}} - 1}{10^{\frac{ER}{20}} + 1} \qquad \textit{General case} \tag{43}$$

$$\gamma_\infty = \lim_{ER\to\infty} \frac{10^{\frac{ER}{20}} - 1}{10^{\frac{ER}{20}} + 1} = 1 \quad \textit{Ideal case} \tag{44}$$

The output field equation (38) taking into account a finite ER is as follows:

$$E_{out} = \frac{E_{in}}{2\sqrt{L_{MZM}}} \left(e^{j\frac{\pi}{2V_\pi}V} + \gamma e^{-j\frac{\pi}{2V_\pi}V}\right) \tag{45}$$

Finally it's worth to remark that the higher extinction ratio is, the lower bit error rate is [25]. The table 2 shows how BER tends to zero as ER is increasing.

Extinction Ratio [dB]	Bit Error Rate
3	0.3036
10	0.0150
20	0.0027
40	0.0018
50	0.0017

Table 2: Extinction Ratio vs Bit Error Rate

BER values of table 2 are obtained by setting optical variables in MATLAB with the following values: θ_B=$\frac{3\pi}{4}$, m=0.15. Those values are considered as the default values of the optical parameters through the whole simulation.

4.2.1 Normalized Bias Voltage & Optical Modulation Index

This section is about rewriting the equation (38) through a normalized bias voltage (θ_B) and an optical modulation index (m) in order to get the adequate equation to be implemented in the code.
The first step is defining the electrical signal as described at 4.2.

$$V = V_b + V_{RF}cos(w_{RF}t) \tag{46}$$

As second step, the equation (46) is replaced inside the equation (38).

$$E_{out} = \frac{E_{in}}{\sqrt{L_{MZM}}}cos\left(\frac{\pi}{2V_\pi}\left(V_b + V_{RF}cos(w_{RF}t)\right)\right) \tag{47}$$

The third step is about defining the equations of θ_B and m and replacing them in (47).

$$\theta_B = \pi\frac{V_b}{V_\pi} \qquad m = \pi\frac{V_{RF}}{V_\pi} \tag{48}$$

$$E_{out} = \frac{E_{in}}{\sqrt{L_{MZM}}}cos\left(\frac{\pi}{2V_\pi}\left(\theta_B\frac{V_\pi}{\pi} + m\frac{V_\pi}{\pi}cos(w_{RF}t)\right)\right) \tag{49}$$

$$E_{out} = \frac{E_{in}}{\sqrt{L_{MZM}}}cos\left(\frac{1}{2}\left(\theta_B + m\cdot cos(w_{RF}t)\right)\right) \tag{50}$$

The equation (50) can also be expressed as exponential function.

$$E_{out} = \frac{E_{in}}{2\sqrt{L_{MZM}}}\left(e^{j\frac{\theta_B+m\cdot cos(\omega_{RF}t)}{2}} + e^{-j\frac{\theta_B+m\cdot cos(\omega_{RF}t)}{2}}\right) \tag{51}$$

And considering a finite ER, the equation (51) is as follows:

$$E_{out} = \frac{E_{in}}{2\sqrt{L_{MZM}}}\left(e^{j\frac{\theta_B+m\cdot cos(\omega_{RF}t)}{2}} + \gamma e^{-j\frac{\theta_B+m\cdot cos(\omega_{RF}t)}{2}}\right) \tag{52}$$

The last equation, (52), is the way the optical field is implemented in the code, this way makes easy to define the field due to only 4 parameters are needed to be set.
If desired, the same criteria can be applied to P_{out}:

$$P_{out} = \frac{P_{in}}{2L_{MZM}}\left(1 + cos\left(\frac{\pi}{V_\pi}(V_b + V_{RF}cos(w_{RF}t))\right)\right) \tag{53}$$

$$P_{out} = \frac{P_{in}}{2L_{MZM}}\left(1 + cos\left(\frac{\pi}{V_\pi}\left(\theta_B\frac{V_\pi}{\pi} + m\frac{V_\pi}{\pi}cos(w_{RF}t)\right)\right)\right) \tag{54}$$

$$P_{out} = \frac{P_{in}}{2L_{MZM}}(1 + cos(\theta_B + \theta_{RF})) \tag{55}$$

4.3 Optical Carrier to Signal Ratio

The OCSR is the relation between carrier spectral component power (P_C) and the side-bands spectral components power (P_{SB}).

$$OCSR = \frac{P_C}{P_{SB}} \tag{56}$$

OCSR is helpful when getting the detected RF power (P_{RF}) through total optical power (P_T), the whole demonstration is shown below:

$$P_T = P_C + P_{SB} \tag{57}$$

$$P_T = P_C \left(1 + \frac{1}{OCSR}\right) \tag{58}$$

$$P_{RF} = P_C \cdot P_{SB} \tag{59}$$

$$P_{RF} = P_T^2 \frac{OCSR}{(1 + OCSR)^2} \tag{60}$$

The equation (60) shows that the detected RF power for a given total optical power is maximized for OCSR=1.

4.3.1 OCSR MATLAB Computation

To begin with, θ_B, m and ER are defined as initial values, those values must be chosen according the BER desired, for example a minimum BER is obtained when m and ER increase. The θ_B is chosen according the amount of OCSR needed and taking into account its *linear* zone, the closer to null it is, the lower OCSR is.
The equation (52) is applied as mentioned at 4.2.1, then a FFT is needed to get the optical spectrum and through it the carrier spectral component power and side-bands spectral components power are computed.

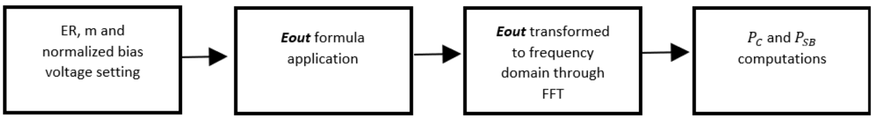


Figure 30: OCSR MATLAB computation block diagram

4.4 Optical fiber

4.4.1 Chromatic Dispersion

When working with optical fibers the dispersion effect must be considered. There are two kinds of dispersion, modal and chromatic dispersion, however the one implemented in RoF system is the chromatic.

Chromatic Dispersion occurs when multiple wavelengths, traveling by fiber, arrive in different times at destination, in time domain that's translated in wave spreading as shown at figure (31).

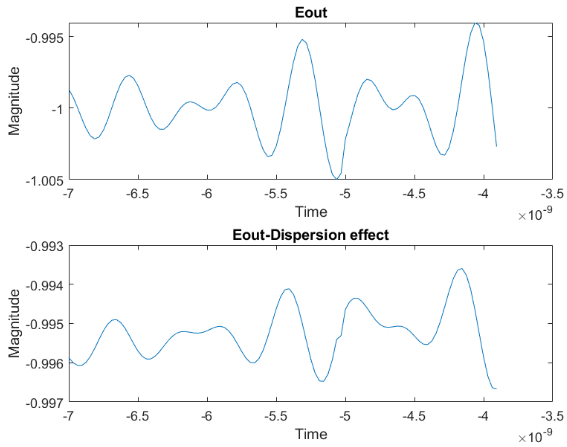


Figure 31: Wave spreading

Mathematically chromatic dispersion can be modeled by the following transfer function:

$$H(w) = e^{j\pi DL\lambda_0^2 \frac{f^2}{c}} \tag{61}$$

Where D is the dispersion parameter, L is the fiber length, λ_0 is the optical window wavelength, f is the carrier frequency and c is the speed of light.
To implement the chromatic dispersion in MATLAB just (61) is required so that all above mentioned parameters need to be defined with a desired values, obviously BER will suffer variations depending on chosen values.

4.5 Photodetector

Photodetector is a photodiode that turns optical signal into an electrical signal, it's implemented after optical fiber as shown at (26).
Despite of existing various types of photodiodes, the one used is the PIN photodiode. Its structure has three zones: P-type zone, active zone and N-type zone. Regarding its operation mode, it is based in absorption, it means that the photon energy causes the electron-hole pairs, then through an electric field applied the current is generated and that's called photocurrent (I_{PD}).

$$I_{PD} = R_d P_{in} \simeq |E|^2 \tag{62}$$

Where P_{in} is the photodector input power and R_d is the responsivity, last one depends on many constants and is measured in $\left[\frac{A}{W}\right]$ as shown below:

$$R_d = \eta \frac{q}{hf_1} \tag{63}$$

Where η is the quantum efficiency, h is the plank constant, q is the electron charge and f_1 is the optical window frequency.
Photodetector is affected by various types of noises, in this case shot noise and thermal noise are considered. The first one appears as a consequence of discrete nature of the chargers [26], while the second is generated through the electron thermal agitation inside a semiconductor [27]. These noises are characterized by their variances and their formulas are shown below:

$$\sigma_{th} = \sqrt{\frac{2k_B T}{R_L} BW_{PD}} \tag{64}$$

$$\sigma_{sh} = \sqrt{q|I_{PD}|BW_{PD}} \tag{65}$$

Where k_B is the Boltzmann constant, BW_{PD} is photodiode bandwidth, T is the standard environment temperature and R_L is the photodiode load resistor. From last set of parameters the most important is the bandwidth due to its shot noise variance proportionality, so that a higher bandwidth will increase shot noise variance and so does BER.
The MATLAB implementation just requires a photodiode current and the computations of noises variances, resulting signal goes directly to frequency down-converter block from RF comms system as shown in figure 33.

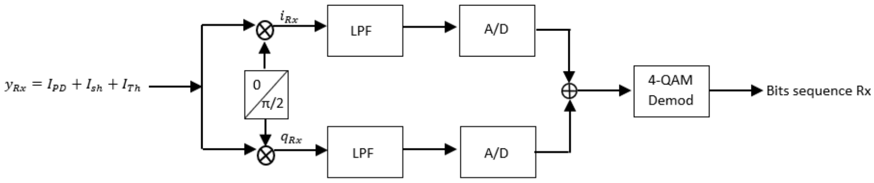


Figure 32: Optical channel ending scheme

5 Experimental Part

5.1 Introduction

To carry out the experimental part SMW200A Rhode&Schwarz vector signal generator (*VSG*) and N9010b Keysight vector signal analyzer (*VSA*) were used. First one is able to perform signal filtering and modulation while second one can perform signal demodulation and compute many IQ signal parameters. Software tools as MATLAB and 89600VSA are also required in this stage, both are helpful when generating baseband signal and then analyzing how signal is received in time and frequency domains, specifically, first task is perfomed by MATLAB and second task by 89600VSA.
It must be taken into account that instruments need drivers in order to be compatible with computers and, as explained in the following sections, drivers must be carefully selected according to the instrument brand.
Some additional software tools appeared when trying to use instrument drivers, one of them is the application programmable interface (*API*) called Virtual instrument software architecture (*VISA*). This API is in charge of supplying a programming interface to control instruments through different types of connections [28].
Finally, it's remarkable to say that, the following equipment and computer configurations were tested on Windows 11 laptop, intel core i7 12th generation.

5.2 Experimental equipment

In this section is given a general description of the instruments used, explaining their main functions and features. This section also contains a full explanation about used drivers and how instruments were configured in order to avoid computer or connectivity crashes/incompatibilities. The section content can be summarized as a blocks diagram shown in figure 33.

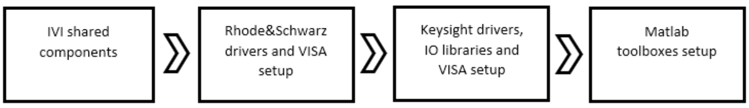


Figure 33: Blocks diagram of the experimental equipment setup

5.2.1 IVI drivers

When using drivers from different brands their interaction is fundamental, to ensure that, there are other drivers acting as a "bridge" between them, these ones are called interchangeable virtual instrument (*IVI*) [29]. In this case, they were essential to carry out the project, because the whole system was formed by instruments from different vendors. Just remains to say that they can be downloaded from IVI foundation web, their installation is quite simple and the files used are called: `IviSharedComponents_300.exe` and `IviNetSharedComponents_200.exe`.

5.2.2 VSG description

Vector signal generator allows signal generation using a specific modulation, it also has many other features like noise addition or filtering, however, modulation skill is what makes it different in comparison to traditional signal generator [30].
As mentioned at section 5.1, the VSG used is the one called SMW200A from Rhode&Schwarz, it's never been used to carry out a radio-over-fiber communication system at university yet, that's why project is innovative.
SMW200A has many features that made it unique, some of them are the following ones: wide frequency range (up to 67 [GHz]), radio-frequency modulation bandwidth up to 2 [GHz], real time signals displaying and many more [31].
An illustration of the VSG used is shown in the figure 34.

Figure 34: Vector signal generator- SMW200A

5.2.3 VSG setup

First of all, vector signal generator needs "general" drivers to be remotely controlled. One driver is called `IVI.NETx64RsRfSigGen` and second one is called `VXIplug&playx64RsRfSigGen`, they can be found on Rhode&Schwarz (R&S) official website and its installation is quite simple.
As mentioned at (5.1), VISA software is essential to use instrument drivers, in this case, the used VISA comes from National Instruments (NI) vendor instead of R&S because it accepts the same standard commands for programmable instruments ($SCPI$) as R&S VISA and it's compatible with Keysight VISA at the same time. The used version in this project is called 2022-Q3 and can be found on National Instruments official website, once installed, do not try to install R&S VISA because it causes some incompatibilities. These ones are translated in "death screen" or "windows blue screen" forcing user to reset the computer in the worst case, so be careful with these installations. It's also highly recommended to not install any other driver in parallel to NI VISA in order to avoid warning messages.
As a recommendation, on R&S website there is one MATLAB toolkit, it provides some scripts with examples of how to make the first remote connection, how to change VSG parameters remotely and many more functions. One of provided example scripts is used as a basecode in this project, it's called `Convert_Mat2Wv.m`. In the next sections its usage is explained.

5.2.4 VSA description

Vector signal analyzer is able to measure radio-frequency signal magnitude and phase, this allows error vector magnitude computation. Signal analyzer is also helpful in signal demodulation as mentioned at (5.1).[32][33]
As in VSG case, Keysight N9010b has many advantages above signal analyzers from other vendors, its differential features are the following ones: wide frequency range (10[Hz] to 44[GHz]), analysis bandwidth up to 40[MHz], good phase-noise relation and many more [34].
Keysight provides a simplified block scheme of its VSA, it is presented in figure 35.

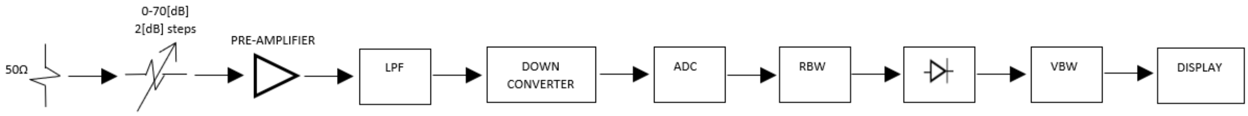


Figure 35: Vector signal analyzer- Blocks scheme [35]

Many VSA scheme blocks match with blocks explained at simulation part and that part in turn is based in telecommunications theory, here it's seen how theory leads to real applications.
An illustration of the VSA used is shown in the figure 36.

Figure 36: Vector signal analyzer- N9010b

5.2.5 VSA setup

The vector signal analyzer can be configured almost instantly just installing `Instrument cont -rol Toolbox support package for Keysight (Agilent) IO Libraries and VISA Interface`, but an alternative way to do is presented in this section.
The first element to be installed is Keysight VISA, it can be installed by separately or implicitly with software `IOLibrariesSuite`. This software is highly recommended because it allows to see which instruments are connected to the network (even those which are not from Keysight), see their VISA and IP directions, all this information will be helpful when MATLAB interacts with instrument. Last component to be installed is the `instrument driver for MATLAB`. Each software element previously listed can be obtained from Keysight and MATLAB websites respectively.
Keysight also provides more drivers that work with MATLAB, they are used for many applications like radio tests, LTE, WCDMA, WLAN, etc.

5.2.6 MATLAB setup

The MATLAB to instruments interaction is achieved using some toolboxes, they allow that instrument drivers and SCPI work in MATLAB programming environment. In this project the toolboxes installed are the following ones: Instrument control Toolbox, Data acquisition toolbox, Instrument control Toolbox support package for Keysight (Agilent) IO Libraries and VISA Interface, Communications Toolbox, DSP System Toolbox, Signal processing Toolbox, Statistics, Machine Learning Toolbox and LTE Toolbox, all of them are needed when experimental part and simulation part are carried out.
It is worth to remark that all drivers from both instruments have `.mdd` extension. That knowledge may be helpful when they require to be moved manually to MATLAB drivers folder. That folder is usually located at: `C:/ProgramFiles/MATLAB/R2022a/toolbox/instrument/instrument/drivers`. Not all the drivers must be installed, some of them just require to be moved to the folder, in case of installation, there is a way to check whether it was successful or not, to do so, `midedit` command must be typed on command window, then click on `file/import` and a little tab shows all the drivers installed.

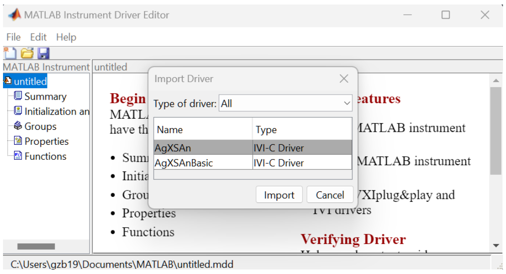


Figure 37: Installed drivers to be used in MATLAB

5.3 RF system assembly

Radio-frequency system is formed by three equipment connected to each other through LAN, specifically to university network called TSC-EEF. Regarding RF channel, it is represented by one RF cable, it goes from vector signal generator RF output to vector signal analyzer RF input. There are two more connections, one is used for signal stabilization, it goes from VSG output trigger to VSA input trigger and second one is in charge of clock synchronization and it goes from VSG REF OUT to VSA REF IN.

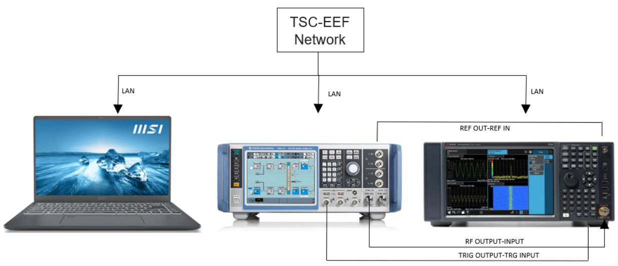


Figure 38: RF system assembly scheme

5.3.1 MATLAB to VSG connection - Code description

This connection allows the information sending, it's carried out using a R & S code as a template, `Convert_Mat2Wv.m`, this code has a set of functions developed by R & S and some SCPI commands.
The goal of this section is to give an overall explanation of the code and its internal functions. From now on this code is renamed as *Sending code* in order to give a better explanation of it. *Sending code* is built by three main functions: `connect()`, `query()` and `generateWave()`, all of them are explained below. `Connect()` is in charge of pairing MATLAB to VSG, this connection can be carried out through VISA or TCP/IP, in this case second option is the chosen one, despite of having NI VISA previously installed, the reason of this choice is that an IP address is easier and more familiar to the user than a long VISA address. The next function to be explained is `query()`, it allows SCPI usage. Through all these commands the instrument input parameters like center frequency, signal level and many more can be remotely modified. Finally, `generateWave()` creates a waveform file and sends it to the instrument.
Once explained main involved functions it's turn to get in detail of how IQ signal is read in the *Sending code*. IQ signal comes from data generated at simulation part, this data consists in: In phase vector, quadrature vector and clock rate, last parameter is not given but it can be computed as follows:

$$CLK_{rate}[\text{MHz}] = \frac{f_{RF} \cdot O_s}{10^6} \tag{66}$$

Where f_{RF} means radio frequency, O_s is the oversampling factor, this factor has a typical value of twelve to have decent view of the waveform at reception .
Both vectors from simulation part, I & Q, and clock rate computed must be stored in `.mat` file and then be loaded by *Sending code*.

5.3.2 VSG to VSA connections

Instruments have three connections in common, they are carried out by RF cable, BNC cable for triggering and another BNC cable for reference signal. RF cable mainly produces attenuation and noise addition to the sent signal, the attenuation increases as RF does, that fact must be considered when choosing the cable. Regarding noise, it mainly depends on cable quality but it's not going to be that important because it can be filtered enough at receiver, i.e. vector signal analyzer [36].

Figure 39: Radio-frequency cable [37]

The next cable connects trigger ports of both instruments, this connection allows signal stabilization, the whole process is explained below.
A trigger signal is sent through the cable, once arrived to receiver it's in charge of marking

the beginning of the received signal to be able of displaying always the same signal on VSA screen, i.e. static signal. It's worth to remark that SMW200A has two output trigger signals, one of them comes from *USER-1* port and the other from *T/M/C-1* port. In this case, second output must not be used due to it acts as a local clock signal for the particular baseband signal while first one contains the right marked trigger signal, according to the signal generator data sheet.

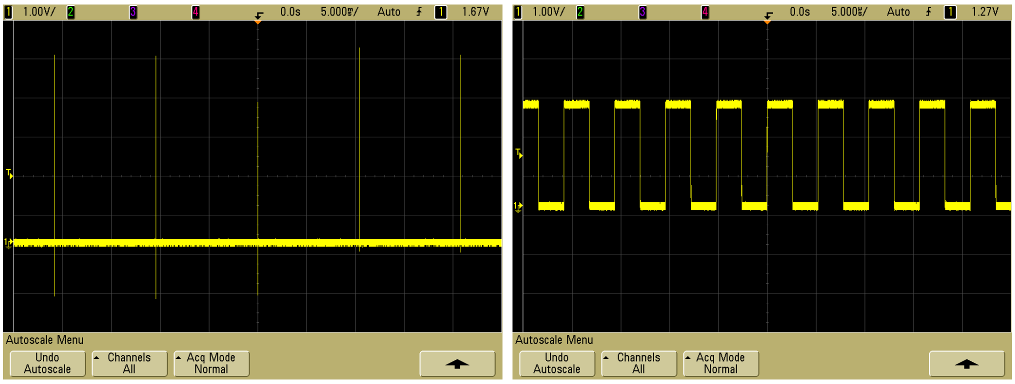


Figure 40: Trigger signals: *USER-1* and *T/M/C-1*

The remaining connection is between reference signal ports of both instruments. This signal is used to synchronize signal generator and signal analyzer thus avoiding signal phase shifts. SMW200A provides a reference signal of 10 [MHz] and N9010B also accepts a reference signal with the same frequency according instrument data sheet, that's why instruments are able to synchronize with each other without problems.

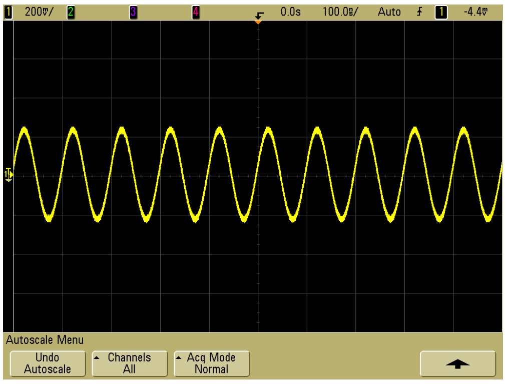


Figure 41: SMW200A Reference signal

All signals shown in this section were captured using an oscilloscope, specifically the MSO6104A from Agilent Technologies.

5.3.3 89600VSA to computer connection

89600VSA is a powerful software, it is able to compute a lot of measurements for multiples technologies, it also allows signal displaying in many ways.

The software setup is explained in this section in order to get the desired results like EVM, I & Q signals and constellation of the received symbols.
First of all, connection between 89600VSA to signal analyzer is required in order to control it remotely, to do so, the software was configured in the following way: Utilities- Hardware- Discovered instruments- Configuration- plus button- right arrow button- OK button. Once signal analyzer is recognized by the software, it just remains to select it as shown in the following figure:

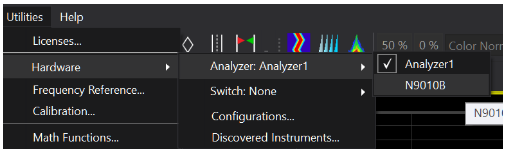


Figure 42: Vector signal analyzer remote connection

The next step is configuring the software to show demodulation results, to do so, a measurement is required, the following figure summarizes this process:

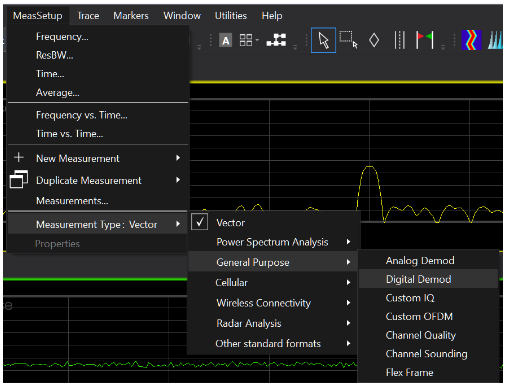


Figure 43: Measurement creation with 89600VSA

Next steps are customizing filter parameters, choosing number of symbols displayed and selecting desired demodulation. In this case, the filter selected is the RRC and the demodulation chosen is QPSK because it is equivalent to the 4-QAM used in simulation part, the remaining parameters can be tuned according user needs.

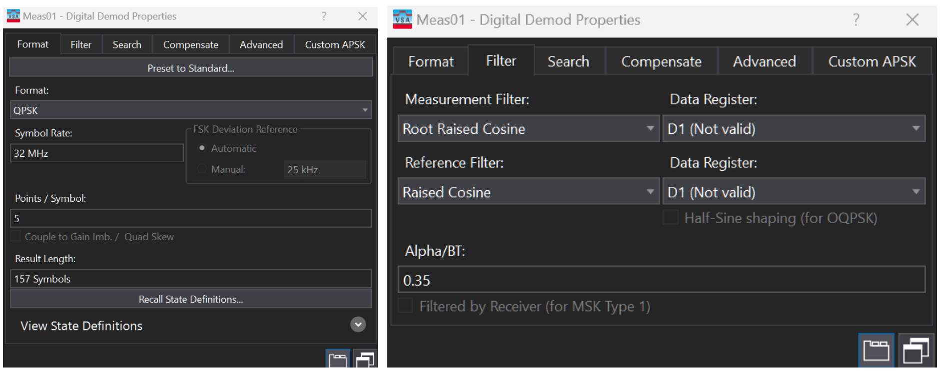


Figure 44: Demodulation properties tuning

The final step is setting a 2x2 view and adjust each graphic to the desired view.

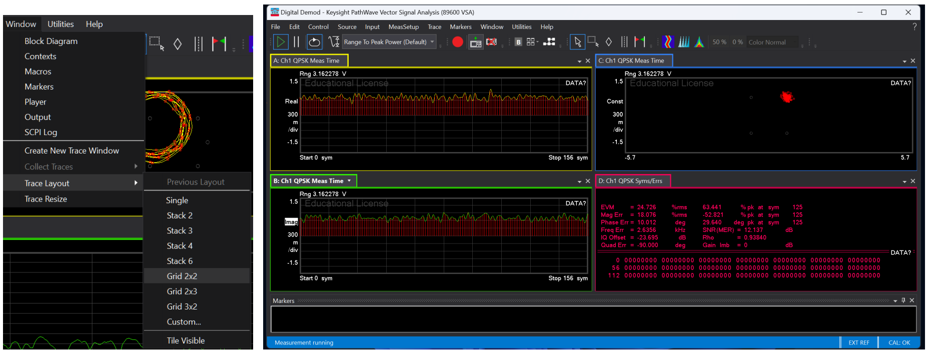


Figure 45: Demodulation results

Finally, it's important to say that when vector signal analyzer is connected to 89600VSA, it just can be controlled through this software.

5.3.4 MATLAB to VSA connection- Code description

The aim of this section is giving an overview of the code needed to send the signal from VSA to MATLAB.
From now on this code is renamed as *Receiving code* in order to give a better explanation of it. To begin with, the main difference with *Sending code* is that it does not use SCPI, however it requires that a driver called *AgilentSADriver.mdd* to be moved to MATLAB drivers folder before running. *Receiving code* also uses a TCP/IP protocol instead Keysight VISA despite of having it previously installed. In this case to establish the connection the instrument IP address and one TCP port are needed, last one is typically the dedicated port 5025. Once both parameters are already defined, the first step is creating the vector signal analyzer as an object in MATLAB environment, next step is assigning the IP and port to the created instrument to finally configure it in *Basic Mode* and *Continuously acquiring*. The remaining part of the code is in charge of plotting obtained signal, generating `.mat` file to export data and closing TCP/IP connection.

5.4 Experiment description

In this section, the RF comms system implementation is explained from system assembling going through signal sending until signal recovering.

5.4.1 RF system assembling

RF system must be assembled as described at 5.3, once completed it the whole system should look like the following figure:

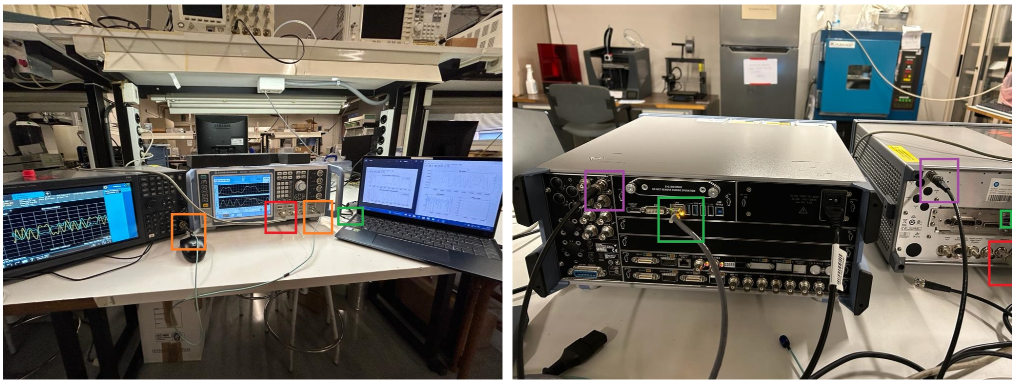


Figure 46: LAN connection-Green, RF connection-Orange, Trigger connection-Red, REF connection-Purple

5.4.2 IQ signal sending

Before starting, it's highly recommended that each `.m` file is located in the same folder and this folder is included in MATLAB path.
To begin with, the code explained at simulation part generates an IQ signal (I,Q vectors) and computes clock frequency (fc), once run it, these three variables must be stored in a `.mat` file. The next steps are carried out at *Sending Code*, here is just a matter of loading `.mat` file previously created and adding the IP of signal generator, last one can be found on VSG main screen or being consulted at *Connection expert* software.

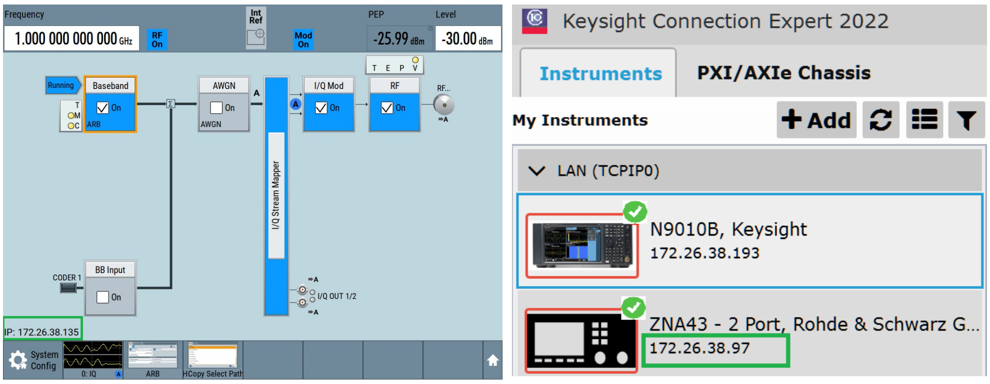


Figure 47: Ways to find VSG IP address

Once signal is sent to VSG it can be displayed in time and frequency domain.

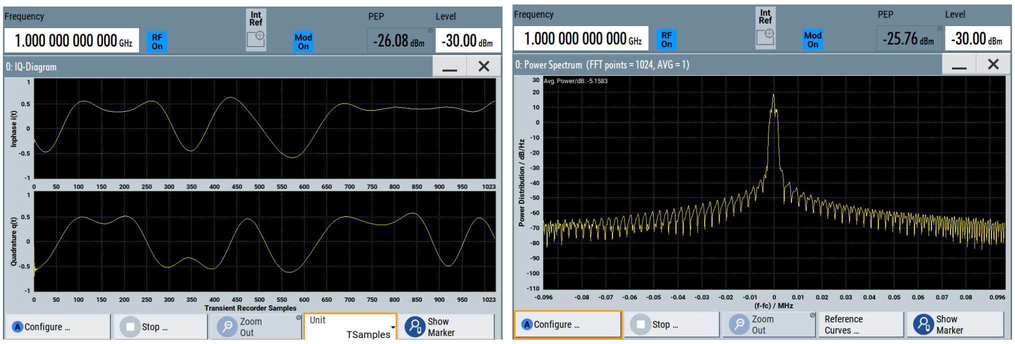


Figure 48: IQ Signal sent to VSG

5.4.3 Signal synchronizing

Signal triggering is not part of the synchronization process but it's needed to view received signal in a static way. Before starting with this process it is worth mentioning that the signal must be periodic to achieve triggering.
The equipment needs one more configuration to trigger the signal. In the case of vector signal generator, it can trigger signal by changing marker signal mode to *Restart*, this mode allows that the rising edge of the marker signal is generated at the signal generation start and at each subsequent signal restart time [38].

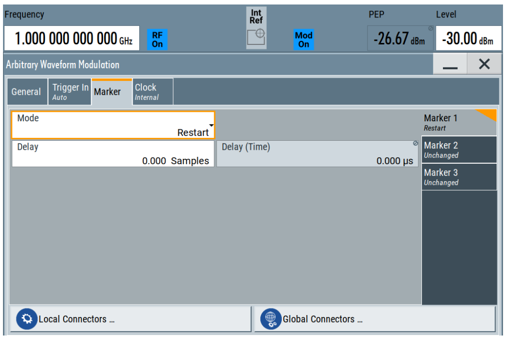


Figure 49: Marker signal settings at transmitter

At vector signal analyzer, trigger source must be selected in trigger settings.
Signal synchronization it is needed to view signals occur at the same time. Synchronization is achieved immediately when connecting instruments REF ports with each other as shown at 5.3.

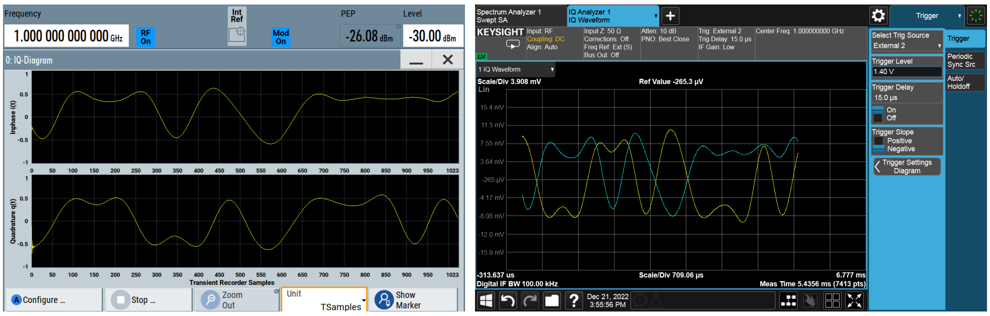


Figure 50: Synchronized I & Q signals- Time domain

Spectrum is not affected in any case, about it just remains to say that received signal experiments a great attenuation at receiver as expected.

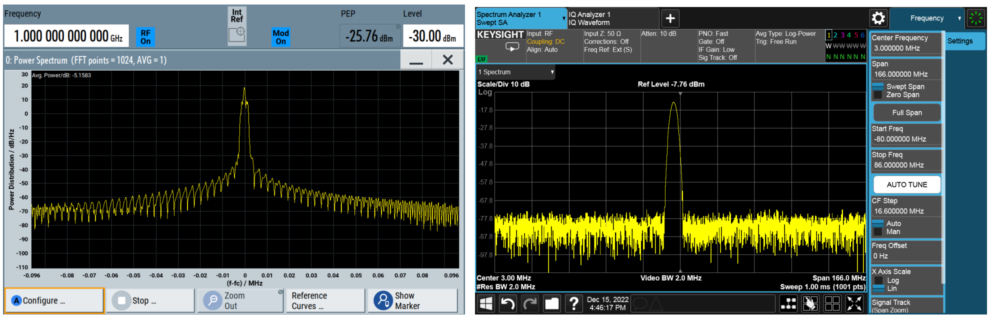


Figure 51: Synchronized I & Q signals- Frequency domain

5.4.4 IQ signal recovering and BER & EVM computations

Once synchronization is achieved, with the help of *Receiving code*, the signal is recovered from vector signal analyzer. Recovered signal suffers some distortion during the whole process that's why in phase and quadrature signals don't look exactly the same.

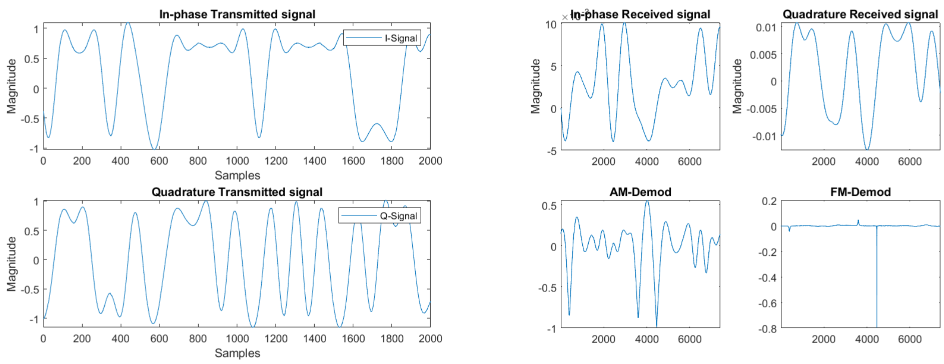


Figure 52: Transmitted IQ signal vs Received IQ signal

At simulation part it is shown that BER must be zero in ideal conditions but experimentally it's not that true. Some tests were made in order to get as minimum BER as possible, obtained results are presented in the following table:

Attempt	**BER**
1º	0.0622
2º	0.0621
3º	0.0627
4º	0.0619
5º	0.0624

Table 3: BER computation

EVM is computed as explained at 3.15 resulting in 22.82 % , that value is closer to the 24.72 % computed by 89600VSA at figure 45.

6 Budget

This project is equivalent to three career subjects, i.e. 18 ECTS, each credit means 25 work hours it gives a total of 450 hours. Regarding professors hours, they're computed taking into account the number of sessions held (10 sessions with director and 4 sessions with an assistant professor), each session lasted 2 hours.

Position	**Hours**	**Hour cost**	**Total cost**
Junior Engineer	450	9 €	4050 €
UPC Professor	20	30 €	600 €
UPC Professor	8	30 €	240 €
		Total	4890 €

Table 4: Staff cost table

The following table shows the indirect costs of the project, specifically, equipment costs (VSG and VSA) are computed as if they were rented for 4 months due to the incapacity to find their real prices and useful life. Regarding software costs, they are assigned according the information found in their official websites students section.

Concept	**Cost**
N9010B (VSA)	2164 €/4 months
SMW200A (VSG)	15448 €/4 months
MATLAB	69 €
89600VSA	4016 €
Total	21697 €

Table 5: Equipment cost table

In conclusion, taking into account both computed costs, the total cost of the project is 26587 €

7 Conclusions

In this project an analysis of Radio over Fiber systems for communications has been presented, providing a detailed description of its main components and how to implement them in a programming environment.
During programming part there had been some aspects to be taken into account as explained in the first two chapters. For the RF system, bit error rate, number of bits, bit rate and sample rate values choice is fundamental to achieve a correct behavior of the RF system. The first three determine how the signal is transmitted and the quality of it by establishing the BER. The fourth parameter controls that the signal can be generated and then recovered without losing any sample. In this part, both time and frequency domain approaches have also been studied and analyzed. The frequency domain DAC conversion is typical of fiber communications software such as VPI-Photonics, while the time domain approach is found in similar communication system simulations examples made with MATLAB. We have verified that the frequency domain conversion is equivalent to an ideal DAC filter with a long number of taps. The conclusion is that in order to be able to simulate degradations coming from a limited number of taps in the DAC filter, the time domain approach is preferred. Nevertheless, both approaches give very similar results for a typical number of taps around 8. For them to be completely equal a minimum number of 50 taps is required.
For the RoF system, through several tests, it has been seen that as extinction ratio increases the BER is reduced, for example, when ER=40 and the rest of optical parameters are set in default values BER=0.0018, it means that BER is getting close to 0 as ER increases, thus verifying the correct implementation of the system. In this part with the same default values for optical parameters, it has also been verified that the RF power is maximum when OCSR is equal to 1.
At the experimental part we have been able to establish a connection, at the RF level, between a Keysight VSA and a Rohde-Schwarz VSG using MATLAB. The connection is carried out through LAN and the communications protocol used was TCP/IP, that allowed fast and reliable communications between the three equipment. Regarding the signal sent to VSG from MATLAB, it was generated using a PRBS18 and a radiofrequency of 2[GHz], this signal was represented in a vector of 8 million values. The instrument synchronization has been achieved by delimiting the original sequence of bits, due to the trigger limitations found in VSA, then marking the beginning of the cut sent signal through VSG settings in order to be recognized by VSA trigger. The final step is about using the reference signal of VSG to avoid signal phase-shifts between both instruments thus achieving the synchronization. Regarding BER, the one measured experimentally has been 0.06, this value is closer to 0, it can be considered as a very good result due to the BER obtained at simulation part, when just the RF system is used, is equal to 0.

8 Future work

In order to give continuity to the project some tasks are presented below.

- **89600VSA demodulation improving:** Despite of information and assistance proportioned by Keysight, constellation is still missing 3 symbols in 4-QAM demodulation, another problem is that IQ signal can't be triggered with recommended software settings so that these two problems are left to be solved.
- **RF transmission improving:** An attenuator can be added between RF ports connection, it'll help to get a better impedance matching by lowering standard wave ratio [39].
- **Optical channel addition:** It can be added to RF system in order to test and compare experimental to simulation results.
- **BER and EVM improving:** They can be improved by ignoring DC subcarrier on signal analyzer, optimizing I/Q modulator on SMW200A or moving carrier leakage out of band [40].

References

[1] Business Research. 2020. Radio over Fiber market size growth. https://www.businessresearchinsights.com/market-reports/radio-over-fiber-market-100128

[2] Mouser Electronics. ND. RF Wireless Technology. https://www.mouser.in/applications/rf-wireless-technology/

[3] Britannica. ND. Radio waves. https://www.britannica.com/science/electromagnetic-radiation/Radio-waves

[4] Muñoz-Ortiz Rafael Oswaldo, Cely-Mancipe Manuel Arturo, Puerto-Leguizamón Gustavo Adolfo, Suárez-Fajardo Carlos Arturo. 2015. Signal Generation for Radio Over Fibre Systems Based on Optical Combination. https://www.scielo.org.mx/scielo.php?script=sci_arttext&pid=S1405-77432015000400010

[5] FOSCO. ND. WHAT IS RADIO OVER FIBER. https://www.fiberoptics4sale.com/blogs/archive-posts/95043078-what-is-radio-over-fiber

[6] Wikipedia. ND. Radio over fiber. https://en.wikipedia.org/wiki/Radio_over_fiber

[7] SAURABH R. PRASAD/ DR A. A. GHATOL/MRS A. V. PATIL. 2010. RADIO OVER FIBER TECHNOLOGY USING ELECTRO-ABSORPTION MODULATION. https://www.researchgate.net/publication/50361768_RADIO_OVER_FIBER_TECHNOLOGY_USING_ELECTRO-ABSORPTION_MODULATION

[8] Sara Cabanas, Carla Macias, MonteCarlo BER Simulation of Radio Over Fiber Communication Links.

[9] Electronicsnotes. ND. What is Bit Error Rate: BER tutorial. https://www.electronics-notes.com/articles/radio/bit-error-rate-ber/what-is-ber-definition-tutorial.php

[10] Wikipedia. ND. Nyquist-Shannon Sampling theorem. https://en.wikipedia.org/wikiNyquistE28093Shannon_sampling_theorem

[11] CloudRF. ND. Modelling the Bit Error Rate (BER). https://cloudrf.com/modelling-the-bit-error-rate-ber/

[12] Mathuranathan Viswanathan,Wireless Communication Systems in Matlab Second Edition, 2020.

[13] Wikipedia. ND. Finite Impulse Response. https://es.wikipedia.org/wiki/FIR_(Finite_Impulse_Response)

[14] Arturo Aibar, Design and Optimization of Radio-over-Fiber Links, 2021.

[15] Robert Oshana. 2012. FIR Filter. https://www.sciencedirect.com/topics/engineering/fir-filter

[16] Erkin Cubukcu. 2012. Root Raised Cosine (RRC) Filters and Pulse Shaping in Communication Systems. https://ntrs.nasa.gov/api/citations/20120008631/downloads/20120008631.pdf

[17] ResearchGate. 2015. How one could determine the Rolloff Factor for a given modulation type?. https://www.researchgate.net/post/How_one_could_determine_the_Rolloff_Factor_for_a_given_modulation_type

[18] Ken Gentile. 2007. Digital Pulse-Shaping Filter Basics. https://www.analog.com/media/en/technical-documentation/application-notes/AN-922.pdf

[19] Quora. 2017. What is the significance of order of a filter? https://www.quora.com/What-is-the-significance-of-order-of-a-filter

[20] Rosa Maria Roldán Giraldo, Quadrature amplitude modulated microwave signal transmission over a radio-over-fiber link using semiconductor optical amplifiers, 2008.

[21] ElectronicDesign. ND. Understanding Error Vector Magnitude. https://www.electronicdesign.com/resources/engineering-essentials/article/21799720/understanding-error-vector-magnitude

[22] Albert Valverde Cabezas, Radio-over-Fiber Data Transmission ,2021.

[23] Basics RoF pptx, Maria Santos Blanco.

[24] SYNOPSYS. ND. Mach-Zehnder Modulator. https://www.synopsys.com/glossary/what-is-a-mach-zehnder-modulator.html

[25] Embrionix. ND. Extinction Ratio Simplified. https://www.embrionix.com/resource/12G-SDI-Extinction-ratio-simplified-st297-st2082-compliant-sfp

[26] Electronicsnotes. ND. What is Shot Noise. https://www.electronics-notes.com/articles/basic_concepts/electronic-rf-noise/shot-noise-what-is.php

[27] NF Corporation. 2020. Thermal Noise. https://www.nfcorp.co.jp/english/techinfo/technicalvocabulary/thermal_noise.html

[28] NI. ND. Información General sobre NI-VISA. https://www.ni.com/es-es/support/documentation/supplemental/06/ni-visa-overview.html

[29] IVI Foundation. ND. Shared Components. https://www.ivifoundation.org/shared_components/

[30] Tektronix. 2021. Types of signal generators. https://www.tek.com/en/blog/types-of-signal-generators

[31] RohdeSchwarz. ND. RSSMW200A vector signal generator. https://www.rohde-schwarz.com/es/productos/test-y-medida/generadores-de-senales-vectoriales/rs-smw200a_63493-38656.html

[32] Wikipedia. ND. Vector Signal Analyzer. https://en.wikipedia.org/wiki/Vector_signal_analyzer

[33] Sanko. ND. Expert Provider for Wireless Communication Test Equipment. https://www.sanko.my/

[34] Keysight. ND. N9010B EXA Signal Analyzer, 10 Hz to 44 GHz. https://www.keysight.com/us/en/product/N9010B/exa-signal-analyzer-multi-%touch-10-hz-44-ghz.html

[35] TRS-RenTelco. 2019. How to Improve N9010B EXA X-Series Signal Analyzer Sensitivity — TRS-RenTelco. https://www.youtube.com/watch?v=vWdp325wPxs&ab_channel=TRS-RenTelco

[36] Broadband library. ND. Coaxial Cable Attenuation. https://broadbandlibrary.com/coaxial-cable-attenuation/

[37] Amazon. Connector Extension. https://www.amazon.es/WayinTop-Conector-Extensi

[38] RohdeSchwarz. ND. RSSMW200A Vector Signal Generator User Manual. https://scdn.rohde-schwarz.com/ur/pws/dl_downloads/pdm/cl_manuals/user_manual/1175_6632_01/SMW200A_UserManual_en_34.pdf

[39] Wikipedia. ND. Attenuator. https://en.wikipedia.org/wiki/Attenuator_(electronics)

[40] KF7JO. 2022. Handling Carrier Leakage with SMW200A or FSW. https://www.youtube.com/watch?v=5tYUFkEaoow&list=LL&index=77&ab_channel=KF7JO