Universidade de Aveiro Departamento de
2017 Eletrónica, Telecomunicações e Informática

João Alexandre Domingues Milheiro

Deteção e Caracterização de Sinais DVB-T para Implementação do Modelo LSA em Portugal

Universidade de Aveiro Departamento de
2017 Eletrónica, Telecomunicações e Informática

João Alexandre Domingues Milheiro

Deteção e Caracterização de Sinais DVB-T para Implementação do Modelo LSA em Portugal

Detection and Characterization of DVB-T Signals for LSA Model Implementation in Portugal

Dissertação apresentada à Universidade de Aveiro para cumprimento dos requisitos necessários à obtenção do grau de Mestre em Engenharia de Eletrónica e Telecomunicações, realizada sob a orientação científica do Professor Doutor Nuno Miguel Gonçalves Borges de Carvalho, professor catedrático do Departamento de Eletrónica, Telecomunicações e Informática da Universidade de Aveiro.

Dissertation presented to the University of Aveiro for the fulfilment of the necessary requisites to obtain the degree of Master in Electronics and Telecommunications Engineering, developed under the scientific guidance of PhD Professor Nuno Miguel Gonçalves Borges de Carvalho, Professor in Department of Electronics, Telecommunications and Informatics of the University of Aveiro.

Apoio financeiro da
ANACOM
Autoridade Nacional de Comunicações

“The starting point of all achievements is desire”

o júri / the jury

Presidente / President	**Professor Doutor Pedro Miguel Ribeiro Lavrador** Professor Auxiliar do Departamento de Eletrónica, Telecomunicações e Informática da Universidade de Aveiro
Vogais / Examiners comittee	**Doutor José Pedro Mateiro Matias Borrego** Engenheiro de Sistemas na ANACOM, Autoridade Nacional de Comunicações (arguente)
	Professor Doutor Nuno Miguel Gonçalves Borges de Carvalho Professor Catedrático do Departamento de Eletrónica, Telecomunicações e Informática da Universidade de Aveiro (orientador)

Agradecimentos / Acknowledgements

Um agradecimento muito especial aos meus pais pelo apoio, dedicação e conselhos que me deram ao longo dos cinco anos de universidade.

Agradeço à Sofia Correia por ter estado ao meu lado nos bons e maus momentos e pela confiança que me deu para continuar a lutar pelos meus sonhos.

Um grande obrigado ao meu orientador, Prof. Nuno Borges de Carvalho, pelas oportunidades e motivação que me deu ao longo deste ano e porque sem ele esta dissertação não seria possível.

Obrigado ao Ricardo Loureiro e Gonçalo Duarte Rosa pelo companheirismo e amizade única que se criou, foram sem dúvida uma amizade que me fez chegar onde estou agora e agradeço-lhes por isso.

Um agradecimento aos alunos de doutoramento Ricardo Correia, Daniel Belo, Luis Almeida e Marina Jordão pela ajuda que deram e pelas dúvidas tiradas.

Um ultimo agradecimento aos técnicos Paulo Gonçalves e Nuno Silva pelo apoio prestado no desenvolvimento desta dissertação.

Palavras-chave DVB-T, LSA, Partilha de espetro, Deteção de sinais RF

Resumo

Com a necessidade de uma maior eficiência no uso do espectro radioelétrico devido às exigências dos consumidores, as entidades reguladoras tentam arranjar soluções para tornar essa utilização de espetro mais eficaz mantendo sempre a qualidade de serviço e confiabilidade do mesmo. Tendo isto em conta a RSPG propôs o conceito de modelo *Licensed Shared Access* (LSA) que faz uso de três dimensões para realizar partilha de espetro: tempo, frequência e geolocalização.

O objetivo desta dissertação é apresentar um sistema a ser incorporado no modelo LSA em Portugal, cujo propósito é localizar, detetar e caracterizar o sinal do incumbente de modo a verificar se este tem condições para transmitir o seu sinal ou não.

Keywords DVB-T, LSA, Share Spectrum, Signal Sensing

Abstract

With the necessity of more efficient use of the spectrum due to the demand from the consumers, the spectrum regulatory entities must find ways to make better use of the available spectrum and create techniques that allow good QoS and the same reliability as before. Taking this into account, RSPG proposed the concept of the Licensed Shared Access (LSA) model which makes use of three dimensions to perform the frequency sharing: time, frequency and geolocation.

The objective of this dissertation is to present a system that can incorporate the LSA Model in Portugal and its purpose is to locate, detect and characterize the incumbents' signal in order to verify if there are good conditions for the incumbents' signal to be transmitted.

Contents

List of Figures

List of Tables

List of Acronyms

CEPT : *European Conference of Postal and Telecommunications Administrations*
CP : *Cyclic Prefix*
CR : *Cognitive Radio*
ECC : *Electronic Communications Committee*
ETSI : *European Telecommunications Standards Institute*
FFT : *Fast Fourier Transform*
GI : *Guard Interval*
GPRS : *General Packet Radio Service*
GSM : *Global System for Mobile Communications*
ICI : *Inter-Carrier Interference*
IFFT : *Inverse Fast Fourier Transform*
ISI : *Inter-Symbol Interference*
LSA : *Licensed Shared Access*
MFCN : *Mobile/Fixed Communications Networks*
MNC : *Maximum Normalized Correlation*
NRA : *National Regulatory Authority*
NTFA : *National Tables of Frequency Allocation*
OFDM : *Orthogonal frequency-division multiplexing*
PCB : *printed circuit board*
QAM : *Quadrature Amplitude Modulation*
QoS : *Quality of Service*
QPSK : *Quadrature phase shift keying*
SAB : *Services Ancillary to Broadcasting*
SAP : *Services Ancillary to Programme-making*
TPS : *Transmission Parameter Signaling*

Chapter 1

1 Introduction

1.1 Context & Motivation

1.1.1 Radio Spectrum Management

Nowadays, the demand of frequency spectrum bands by the consumers lead to a greater and better management of the spectrum. Every National Regulatory Authority (NRA) has the responsibility to ensure the QoS and the proper frequencies bands for each user. National Tables of Frequency Allocation (NFTA) are used worldwide to ensure that there are transmission rules for most applications and for every frequency. These applications, normally, are attributed only to one frequency chosen by the NRA, which will need certain authorizations to be used. The Figure 1.1 shows how NRA manages the radio spectrum with the use of NTFA and different kind of authorizations for each consumer/user.

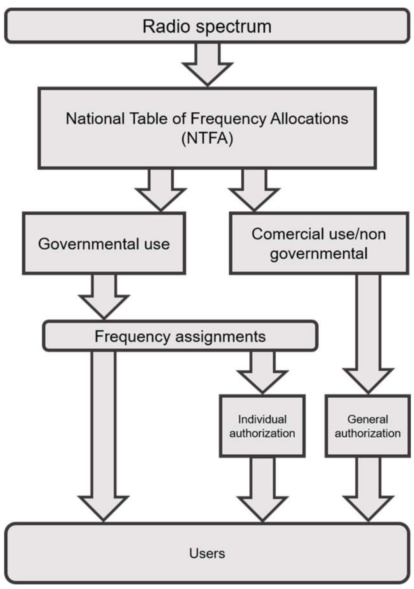


Figure 1.1 - Radio spectrum management

However, consumers are becoming more and more dependable on mobile broadband services and require low cost, high quality and faster technology to their services. Due to this demand, the amount of bandwidth used by these services is increasing rapidly and the actual method of spectrum management could not be enough to ensure a proper QoS for the consumers [1].

With this problem in mind, the RSPG came with the concept of spectrum sharing which basically consist in a *"common usage of the same spectrum resource by more than one user"* [1]. Despite the promising future for this concept, this is a very delicate matter in the sense that both entities sharing the spectrum must not be affected by this deal, and both must get the QoS requested and with no harmful interference to the signal.

1.1.2 LSA & Spectrum Management

In 2013 the *RSPG opinion on Licensed Shared Access* was approved and it defines LSA as:

> "A regulatory approach aiming to facilitate the introduction of radiocommunication systems operated by limited number of licensees under an individual licensing regime in a frequency band already assigned or expected to be assigned to one or more incumbent users. Under the Licensed Shared Access (LSA) approach, the additional users are authorized to use the spectrum (or part of the spectrum) in accordance with sharing rules included in their rights of use of spectrum, thereby allowing all the authorized users, including incumbents, to provide a certain Quality of Service". [2]

The LSA is a complementary tool for spectrum management that makes use of three dimensions to perform the frequency sharing: time, frequency and area. The frequency band approved for the LSA trials was the 2.3 – 2.4 GHz and the mobile/fixed communications networks (MFCN) are the services approved to be used as second users of that frequency band. The LSA model will give the incumbent(s) and licensee(s) a predictable QoS and protection of the signal against harmful interference. Figure 1.2 shows a scheme of how LSA's shared framework can be attributed to the incumbent and allow the other licensees to access the same spectrum resource. This shared framework is the set of sharing rules and/or sharing conditions, and it is NRA's responsibility to set these conditions. They can be established in consideration to future needs of an incumbent (marked in blue), as shown in Figure 1.2. Since the green area will not be used by the incumbent, it can be allocated to a licensee. This way, neither the incumbent, nor the licensee, will be affected from each other signals and both can utilize the same spectrum resource.

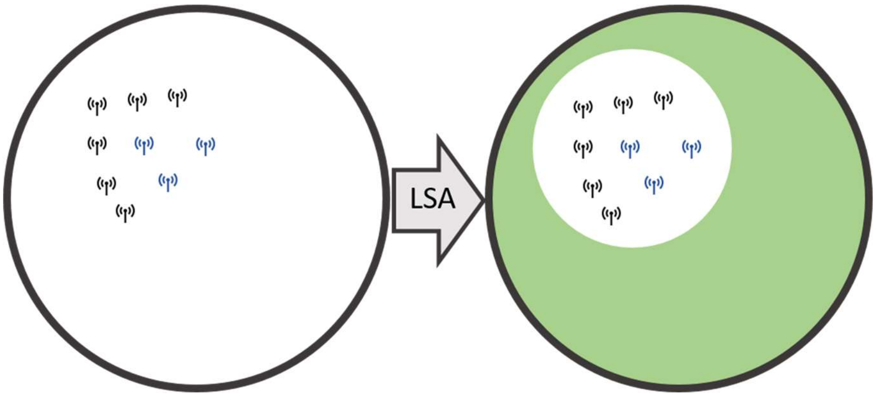


Figure 1.2 - Geographical influence of the LSA model

1.1.3 LSA Functional Blocks

According to CEPT [1], there will be at least 2 main components that will compose the LSA model: the **LSA repository** which will contain real time information about the spectrum usage, the availability of each spectrum resources and the conditions that are associated to them. This repository is controlled by the NRA or the incumbent, and will manage the spectrum allocation and conditions for the incumbents/licensees for each spectrum resources; The **LSA controller** will manage the access of the licensee to the designated spectrum resource, making use of the sharing rules and conditions, available in the repository, to make the decisions. In Figure 1.3 is represented the functional blocks for the LSA model according to ECC.

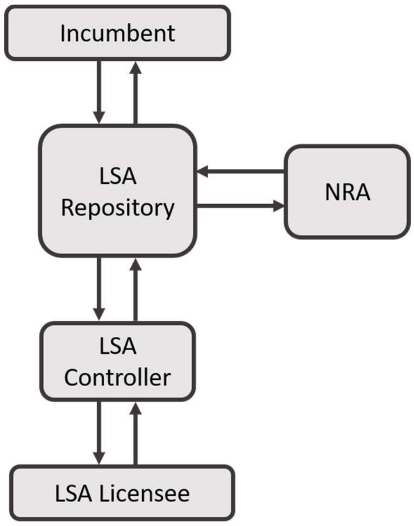


Figure 1.3 - Diagram of the main functional blocks of the LSA model

1.1.4 LSA Model in Portugal

The NRA in Portugal, ANACOM, is now starting to study the possibility to implement the LSA model in the country. The proposal presented by ANACOM is shown in Figure 1.4, and since in Portugal the frequency band of 2.3 and 2.4 GHz is allocated to Services Ancillary to Programme-making (SAP) or Services Ancillary to Broadcasting (SAB) applications [3], the television broadcasters were established as the incumbents.

The goal is to attach a LSA warner to every camera from the major national television broadcasters and anytime they are up to transmit their signal, the warner gathers spectrum information and its geolocation. The warner must be able to detect any signal being emitted between the frequency band of 2,3 and 2,4 GHz and verify if the incumbent has all the conditions to transmit his signal. This information is then sent to the LSA repository which will communicate with the several LSA controllers, to allow (or not) the licenses to transmit in that frequency and in that area. All this operation is controlled and supervised by ANACOM and every rule to the licensees and incumbents are also defined by them.

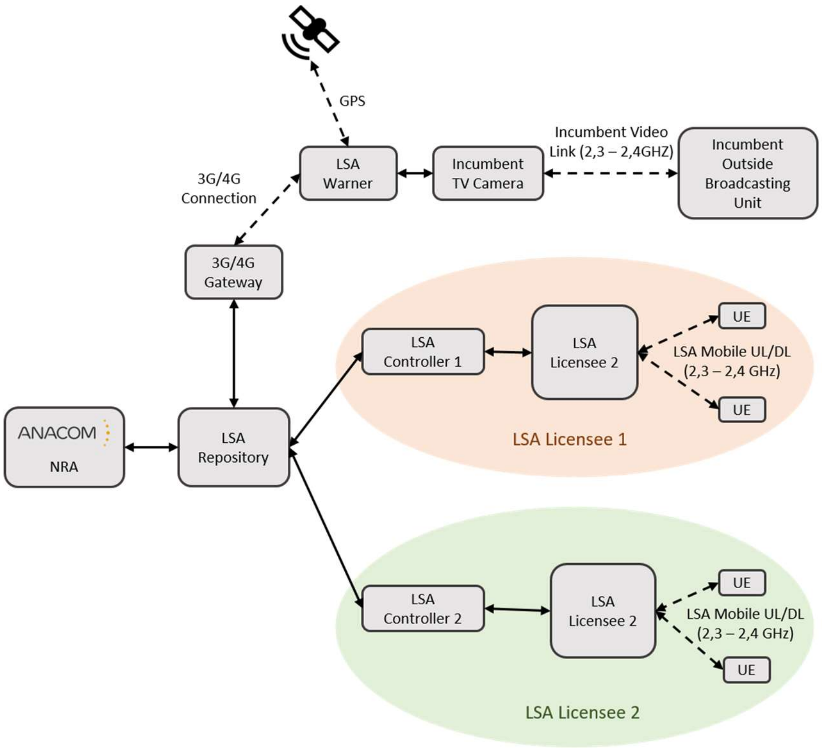


Figure 1.4 - ANACOM solution for the LSA model in Portugal

1.2 Dissertation Objectives

The goal of this dissertation is to develop the LSA warner which will incorporate the LSA model proposed by ANACOM. To achieve this goal, the development of four main blocks for the LSA warner is proposed:

- **Signal Energy Detector:** which consists in a RF receiver able to detect any signal with at least -80 dBm in the frequency bands from 2.3 and 2.4 GHz;
- **Signal Characterizer:** which consists in DVB-T demodulator implemented in a FPGA. The purpose of this demodulator is to check if a signal, when detected, is really a DVB-T signal and verify if the signal condition is acceptable to the transmission of the incumbent's signal. However, the demodulator will first be simulated to verify the liability of this method and, only then, implemented in the FPGA;
- **GPS & GSM/GPRS module**: this part has two purposes, the first is to get position and time from a GPS module and the second to transmit all the information gathered from the systems above through a GSM/GPRS transmitter and send it to an online database.

- **Server/Website**: Here, a simple website is developed to display all the information sent by the warner, it is also necessary to develop an interface to read the warner's data.

1.3 Dissertation Outlines

This document if composed by five chapters, including this introductory chapter:

- **Chapter 2 – DVB-T protocol:** It explains the proprieties of the DVB-T protocol and shows a basic DVB-T transceiver.
- **Chapter 3 – DVB-T Signal Detection**: In this chapter, the basic concept of RF signals detections is explained and some work that has been done in this area is shown.
- **Chapter 4 – LSA Warner Proposal:** Firstly, an overview of the system to develop and all the modules necessary to fulfill the goal is explained. The following subchapters are used to explain, in detail, what was made and to show all the results obtained in each warner's module.
- **Chapter 5 – Conclusions and Future Work:** The last chapter presents the conclusions to all the work done and shows the future work related to this project.

1.4 Original Contributions

This dissertation provided the opportunity to participate in the URSI Portugal 2017, 11º Congress of URSI's Portuguese Committee held in Lisbon on 24th of November. For this congress, a paper was submitted: J. Milheiro, N. B. Carvalho, *DVB-T signal sensing for the LSA model in Portugal.* This paper was accepted and presented at the congress and can be seen at Appendix A.

Chapter 2

2 DVB-T Protocol

This chapter is based on the European Standard *ETSI EN 300 744* published by ETSI, about the DVB-T structure, channel coding and modulation [4]. The Digital Video Broadcasting – Terrestrial is the European standard for transmission of digital terrestrial television signal and allows different transmission modes, code rates and modulations, which give this protocol a great versatility and adaptation to the necessities of the service requirements. This is a protocol based on OFDM system with several transmissions modes, the "2k mode", the "8k mode" and the "4k mode"; it also allows different levels of QAM modulations, code rates and the possibility of hierarchical or non-hierarchical channel coding and modulation. The DVB-T signals can be transmitted under three types of channel, with different bandwidth (8, 7 and 6 MHz), which will conditionate the OFDM symbol duration.

In Figure 2.1, the functional blocks of a DVB-T transmitter are presented. This transmitter is composed by the following stages:

- Transport multiplex adaptation and randomization for energy dispersal:
    - The information is first randomized, to ensure proper binary transitions, and then organized in packets with fixed length and a synchronization word at the beginning of each packets;
- Outer/Inner Coder and Interleaver:
    - DVB-T protocol implies the use of two layers of coding and interleaving. Both serve the purpose of making the information more unlikely to be affected by errors during transmission;
- Mapping and modulation:
    - Mapping and modulations will translate the binary information, processed by the previous blocks, to phases and amplitudes according to the

modulation type desired, here the pilots are also inserted in the OFDM symbol;

- OFDM transmission:
    - After the modulation is completed, the information needs to be rearranged properly to be applied to the IFFT block, which will convert the information from frequency domain to time domain for subsequent transmission by the front-end.

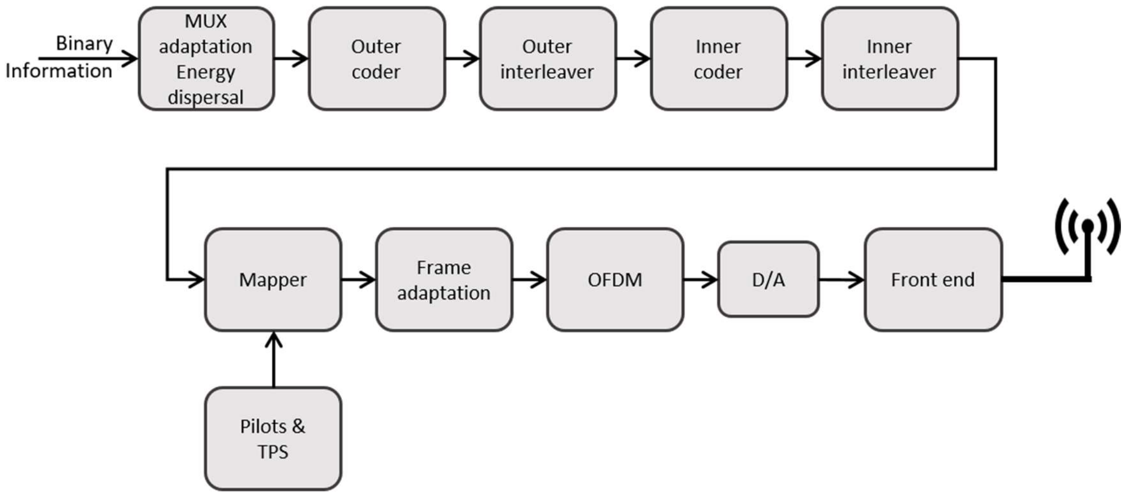


Figure 2.1 - DVB-T transmitter block diagram of the standard.

2.1 Transport Multiplex Adaptation and Randomization for Energy Dispersal

In this step, the received binary data is rearranged into packets for the MPEG-2 transport multiplex (MUX). Each packet contains 187 bytes of information plus a byte for a synchronization word (47_{hex}). Every seven packets, this word is bit-wise inverted ($B8_{hex}$), as shown in Figure 2.2. Also, in this stage, the packets need to be randomized and, for that, the PRBS generator is used, which uses an initial sequence (100101010000000) to generate the random bits of the binary data in the packets according to the following generator: $1 + x^{14} + x^{15}$, the PRBS cannot affect the synchronization words and for that an enable flag is used which is deactivated every time the eight bits of the synchronization word are being processed, this way the word is not affected by the PRBS generator. Figure 2.3 shows how the PRBS output is obtained.

Figure 2.2 - Binary packets with the MPEG-2 transport multiplexer

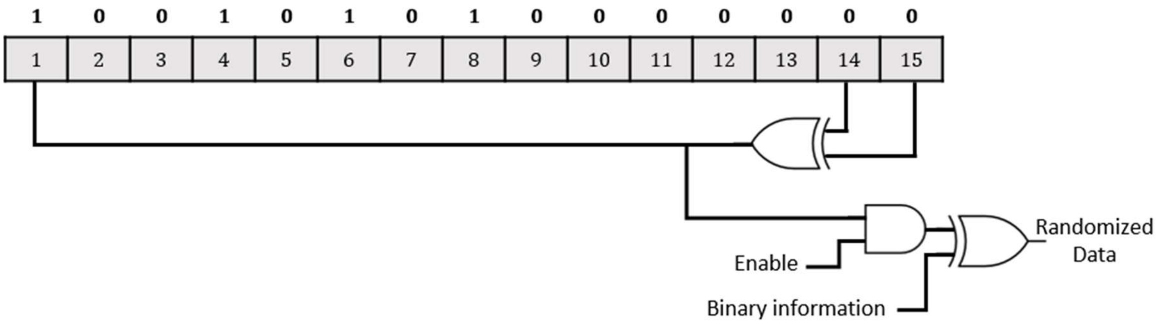


Figure 2.3 - PRBS generator for randomization of the packets information

2.2 Outer Coding

In this stage, a Reed-Solomon coding is performed to the packets presented in Figure 2.2, which generates error protected packets with length of 204 bytes, which the first byte is the synchronization word. The following 187 bytes are the randomized data and the remaining 16 bytes are parity bytes resulted from the Reed-Solomon coding, as presented in Figure 2.4. This coder uses the code generator polynomial $g(x) = (x + \lambda^0)(x + \lambda^1)(x + \lambda^2) \dots (x + \lambda^{15})$, where $\lambda = 02_{HEX}$, and the field generator polynomial $p(x) = x^8 + x^4 + x^3 + x^2 + 1$. Usually, the shortened Reed-Solomon method is chosen for implementation, which can be achieved by adding 51 bytes of zeros before the information and apply the $RS(255,239, t = 8)$ encoder. The null bytes inserted can then be discarded and a Reed-Solomon coding with N=204 bytes is obtained.

Sync 1 *or* SyncN 1 byte	Binary information 187 bytes	Parity bytes 16 byte

Figure 2.4 - Reed-Solomon coded packets

2.3 Outer Interleaver

The outer interleaver consists in a Forney interleaver [5] and it is used to spread errors throughout the packets for a more effective outer coding. This kind of interleaver consists in delaying the bytes of the packets and spread them into various packets, a diagram of the interleaver and deinterleaver is shown in Figure 2.5, where is also represented the delays for each branch.

Figure 2.5 - Diagram of the interleaver and deinterleaver

For this protocol, an interleaver with the following dimensions is used:

- I=12 branches;
- FIFOs with size of M=17;
- Packets to protect with the size of N=204 bytes;

2.4 Inner Coding

The inner coding consists in a selectable range of punctured convolutional codes, based on a mother convolutional code with a code rate of 1/2 presented in Figure 2.6. The input data is the RS packets generated before, which are inserted in series at the “Data Input” entry, the outputs X and Y are then converted to serial. Using the different puncturing patterns to achieve the code rates of 2/3, 3/4, 5/6 and 7/8 presented in Table 2.1.

Code Rates	Puncturing pattern	Transmitted sequence (serial)
1/2	X: 1 Y: 1	$X_1\ Y_1$
2/3	X: 1 0 Y: 1 1	$X_1\ Y_1\ Y_2$
3/4	X: 1 0 1 Y: 1 1 0	$X_1\ Y_1\ Y_2\ X_3$
5/6	X: 1 0 1 0 1 Y: 1 1 0 1 0	$X_1\ Y_1\ Y_2\ X_3\ Y_4\ X_5$
7/8	X: 1 0 0 0 1 0 1 Y: 1 1 1 1 0 1 0	$X_1\ Y_1\ Y_2\ X_3\ Y_4\ X_5\ Y_6\ X_7$

Table 2.1 - Puncturing patterns to obtain the different code rates

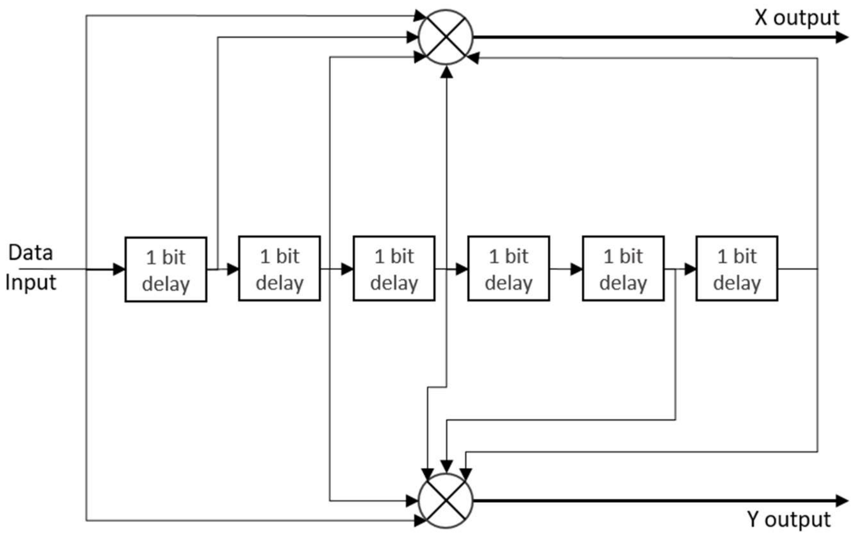


Figure 2.6 - Mother convolutional code of rate 1/2

2.5 Inner Interleaver

The inner interleaver can have two input streams depending if the signal that is being emitted in the hierarchical mode or not. However, in both cases, the interleaver is composed by a demultiplexer stage, a bit interleaver stage and, finally, a symbol interleaver stage. The architecture of the interleaver will depend on the modulation chosen and the type of transmission of the signal (hierarchical or non-hierarchical). Starting with the demultiplexer, in both of hierarchical and non-hierarchical transmissions the objective of this component is to convert the serial input bits into parallel, however, in the case of the hierarchical mode, two demultiplexers are needed, one regarding the high priority stream and other for the low priority stream. Also depending on the modulation picked for the signal, the number of the outputs of the demultiplexer will change. All the possibilities of demultiplexers and their outputs are shown in Table 2.2, where:

- $x_{\mathrm{n,i}}$ → demultiplexer input (series);
- $y_{n,o}$ → demultiplexer output (parallel);
- n → demultiplexer number;
- i → demultiplexer input;
- o → demultiplexer output.

Transmission Mode	Modulation	Nº of deMUX	Nº of outputs		Demultiplexing mapping
QPSK		1	2		x_{00} → y_{00}
					x_{01} → y_{01}
Non-Hierarchical	16-QAM		4		x_{00} → y_{00}
					x_{01} → y_{01}
					x_{02} → y_{02}
					x_{03} → y_{03}
	64-QAM		6		x_{00} → y_{00}
					x_{01} → y_{01}
					x_{02} → y_{02}
					x_{03} → y_{03}
					x_{04} → y_{04}
					x_{05} → y_{05}
Hierarchical	16-QAM	2	HP	2	x_{00} → y_{00}
					x_{01} → y_{01}
			LP	2	x_{10} → y_{10}
					x_{11} → y_{11}
	64-QAM		HP	2	x_{00} → y_{00}
					x_{01} → y_{01}
			LP	4	x_{10} → y_{10}
					x_{11} → y_{11}
					x_{12} → y_{12}
					x_{13} → y_{13}

Table 2.2 - Demultiplexer possibilities for each transmission mode and modulation

Every sub-stream from the demultiplexer outputs are then processed by separated bit interleavers. For the QPSK modulation, it is necessary to have two bit interleavers, for the 16-QAM modulation, four and, for the 64-QAM, six. Each interleaver has the same size of 126 bits and its input bit vector is defined by:

$B(e) = b_{e,0}, b_{e,1}, \dots, b_{e,125}$, where **e** is the number of interleavers, counting from zero.

The interleaved output vector $A(e) = a_{e,0}, a_{e,1}, \dots, a_{e,125}$ is defined by:

$$a_{e,0} = b_{e,H_e(w)} \text{ , where w= 0,1, 2, … ,125}$$

$H_e(w)$ is the permutation function and for every interleaver there is one function defined as:

Interleaver 0	$\boldsymbol{H_0(w) = w}$
Interleaver 1	$\boldsymbol{H_1(w) = (w + 63)\ mod\ 126}$
Interleaver 2	$\boldsymbol{H_2(w) = (w + 105)\ mod\ 126}$
Interleaver 3	$\boldsymbol{H_3(w) = (w + 42)\ mod\ 126}$
Interleaver 4	$\boldsymbol{H_4(w) = (w + 21)\ mod\ 126}$
Interleaver 5	$\boldsymbol{H_5(w) = (w + 84)\ mod\ 126}$

Table 2.3 - Interleavers permutation functions

The outputs of all bit interleavers are then grouped to form the digital data symbols, where the number of bits of the data symbols corresponds to the number of bit interleavers used. The next stage is the symbol interleaver that has the purpose of mapping the bit words into a vector with length of 1512 ($N_{max} = 12\ groups * 126\ data\ symbols$) or 6048 ($N_{max} = 48\ groups * 126\ data\ symbols$), when the transmission mode used is the 2k or 8k mode, respectively. The output vector of this interleaver is defined as $Y' = (y'_0, y'_1, y'_2, \dots, y'_{1512(6048)})$ where:

$$y_{H(q)} = y'_q \text{ , } for\ even\ symbolf\ for\ q = 0, \dots, N_{max} - 1$$

$$y_q = y'_{H(q)} \text{ , } for\ odd\ symbols\ for\ q = 0, \dots, N_{max} - 1$$

2.1

The function $H(q)$ can be obtained by the following pseudo-code below:

```
q = 0;
for(i = 0; i < M_max; i++) {
    H(q) = (i mod 2) * 2^(N_r-1) + Σ_{j=0}^{N_r-2} R_i(j). 2^j;
    if (H(q) < N_max) {
        q = q + 1; }
}
```

2.2

Where $N^r = \log_2(M_{max})$ and the $R_i(j)$ vector is obtained through another vector denominated $R_i'(j)$ which is generated with the rules presented in 2.3.

$$
\begin{array}{ll}
i = 0\,,1: & R_i'[N_r-2, N_r-3, \dots, 1, 0] = 0, 0, \dots, 0, 0 \\
i = 2: & R_i'[N_r-2, N_r-3, \dots, 1, 0] = 0, 0, \dots, 0, 1 \\
2 < i < M_{max}: & R_i'[N_r-3, N_r-4, \dots, 1, 0] = R_{i-1}'[N_r-3, N_r-3, \dots, 2, 1] \\
 & \quad in\ the\ 2k\ mode: R_{i-1}'[9] = R_{i-1}'[0] \oplus R_{i-1}'[3] \\
 & \quad in\ the\ 8k\ mode: R_{i-1}'[11] = R_{i-1}'[0] \oplus R_{i-1}'[1] \oplus R_{i-1}'[4] \oplus R_{i-1}'[6]
\end{array}
\tag{2.3}
$$

$R_i(j)$ vector is now obtained by the permutation of the $R_i'(j)$ following the rules presented in Table 2.4.

2k mode	R_i' bit position	9	8	7	6	5	4	3	2	1	0		
	R_i bit position	0	7	5	1	8	2	6	9	3	4		
8k mode	R_i' bit position	11	10	9	8	7	6	5	4	3	2	1	0
	R_i bit position	5	11	3	0	10	8	6	9	2	4	1	7

Table 2.4 - Bit permutation to obtain the $R_i'(j)$ vector

2.6 Mapping and Modulation

This protocol uses the OFDM transmission and the carriers used for information can be modulated in QPSK, 16-QAM, 64-QAM, non-uniform 16-QAM or non-uniform 64-QAM constellations. Since this protocol allows hierarchical and non-hierarchical transmission, the proportions of the constellation scheme used to modulate the information depends on that transmission mode. When the signal is emitted in the hierarchical mode, a parameter α can be set to 1, 2 or 4, which originates different constellation diagrams displayed in Figure 2.7 to Figure 2.11. This parameter represents the minimum distance separating two high priority constellation points, divided by the minimum distance separating two constellations points. In the non-hierarchical mode, the same constellation diagram is used, as the case with $\alpha = 1$. The Table 2.5 shows the exact values for the constellation points defined as $z \in \{n + jm\}$.

Constellation	α	$n = m\,\epsilon$
QPSK	-	$\{-1, 1\}$
16-QAM	1	$\{3, -1, 1, 3\}$
	2	$\{-4, -2, 2, 4\}$
	4	$\{6, -4, 4, 6\}$
64-QAM	1	$\{-7, -5, -3, -1, 1, 3, 5, 7\}$
	2	$\{-8, -6, -4, -2, 2, 4, 6, 8\}$
	4	$\{-10, -8, -6, -4, 4, 6, 8, 10\}$

Table 2.5 - Constellation points values for every constellation scheme

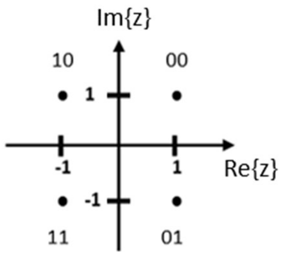


Figure 2.7 - QPSK constellation scheme

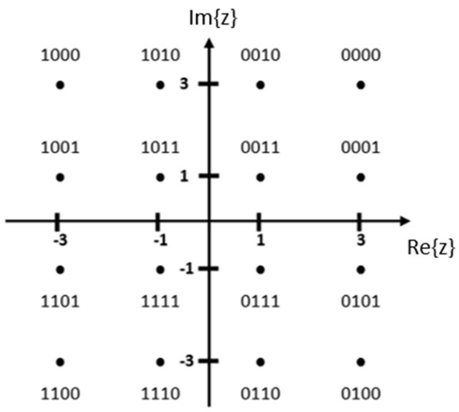


Figure 2.8 - 16-QAM α =1 constellation scheme

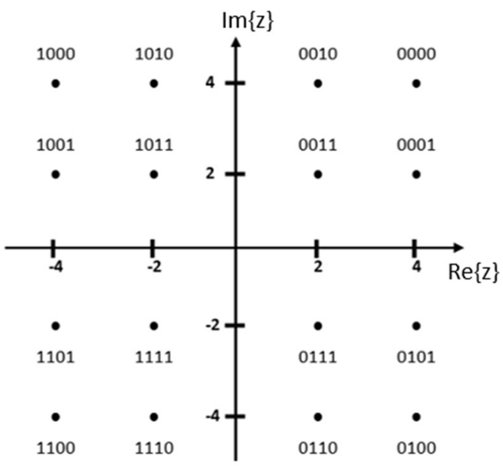


Figure 2.9 - 16-QAM α =2 constellation scheme

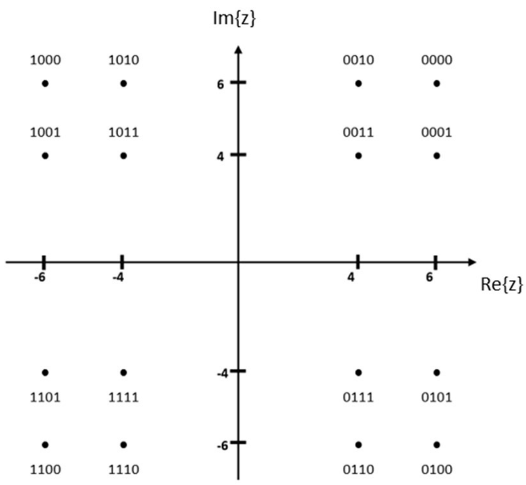


Figure 2.10 - 16-QAM α =4 constellation scheme

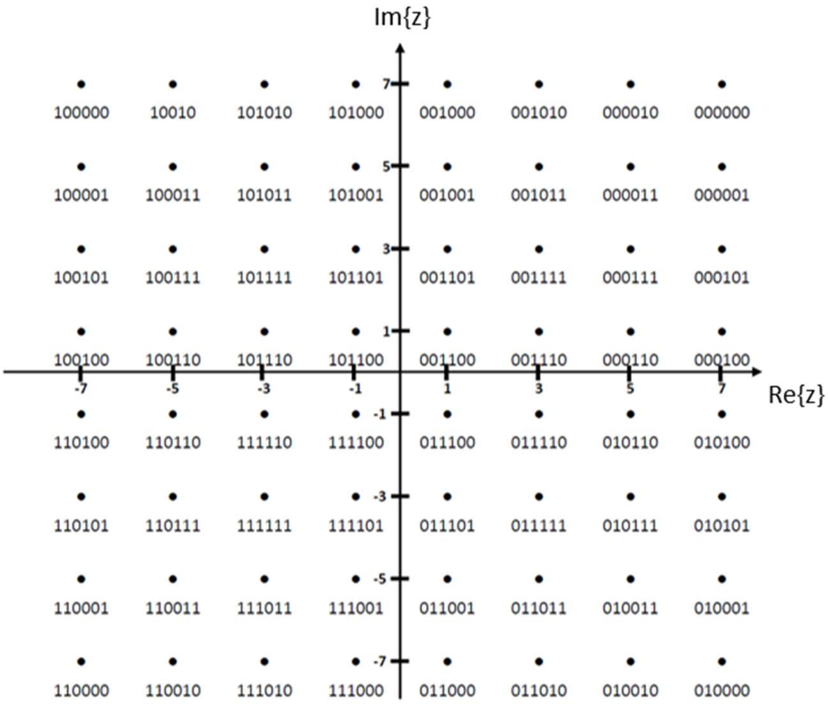


Figure 2.11 - 64-QAM α =1 constellation scheme

The constellations values must be normalized to be properly transmitted, and for that, every constellation point must be multiplied by a normalization factor that yields $E[c \cdot c^*] = 1$, where the c correspond to the normalized modulation values. In Table 2.6 the values of that factor, for each constellation scheme and α, are shown.

Modulation Scheme		Normalization Factor
QPSK		$c = \frac{z}{\sqrt{2}}$
16-QAM	**α = 1**	$c = \frac{z}{\sqrt{10}}$
	α = 2	$c = \frac{z}{\sqrt{20}}$
	α = 4	$c = \frac{z}{\sqrt{52}}$
64-QAM	**α = 1**	$c = \frac{z}{\sqrt{42}}$
	α = 2	$c = \frac{z}{\sqrt{60}}$
	α = 4	$c = \frac{z}{\sqrt{108}}$

Table 2.6 - Normalization factor for constellation schemes

2.7 OFDM Frame Structure

DVB-T protocol has two transmission modes, the 2k and the 8k modes and it is organized in frames. A set of four frames constitute one super-frame. Each frame has 68 OFDM symbol and each symbol as a total number of carriers of 2048 or 8192 for the 2k or 8k, respectively. In case of the 2k mode, from the 2048 existing carriers only 1705 are used for information, either in the form of data or in the form of pilots. The others stay unused and work as guard bands for inter-symbol interferences. In the case of the 8k mode, from the 8192 existing carriers only 6817 are used to contain the signal. A Guard Interval (GI), also called Cyclic Prefix (CP), is added before the signal and it consists in a repetition of the OFDM Symbol's last carriers which can have the size of 1/4, 1/8, 1/16 or 1/32 of the symbol duration.

Every symbol as a constant number of carriers assigned to specific functions, for instance, in the 2k(8k) mode there are always 1512(6048) available carriers for data and the remaining 193(769) are attributed to signal pilots and the TPS.

The pilot carriers containing reference information, are modulated at the "boosted" power level, which is greater than the power of the data carriers modulation. There are two kinds of pilots, the continual pilots, which position is constant in every symbol, and the scattered pilots whose position varies from symbol to symbol, however, these positions are the same for every super-frame. Both pilots are modulated according to a PRBS generator (shown in Figure 2.12), with the initial sequence of "11111111111". It generates a bit (w_k), for every k carrier in the symbol and determines its modulation according to equation 2.4, where m is the frame index, k is the carrier index and l is the symbol index.

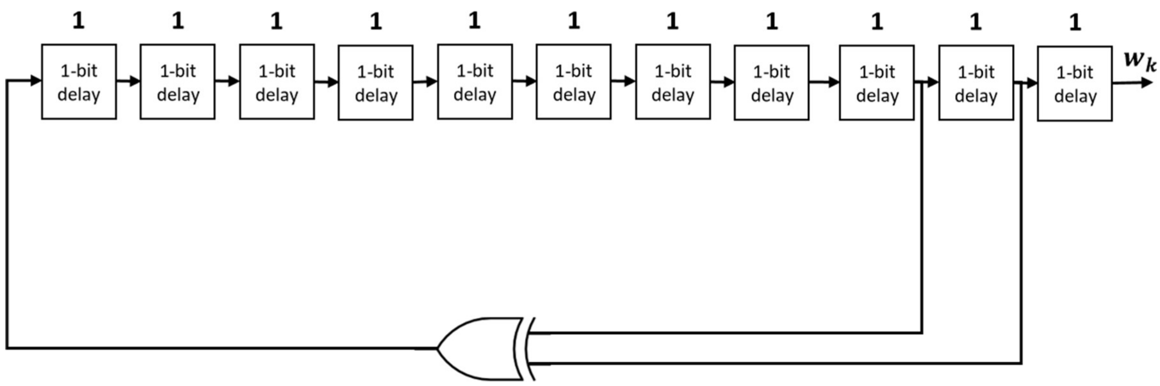


Figure 2.12 - PRBS Generator

The scattered pilot location is calculated by the following equation:

$$K = K_{min} + 3 \cdot (l\ mod\ 4) + 12p$$

2.4

Where:

- K_{min} → minimal index for the symbol carrier
- K_{max} → maximum index for the symbol carrier
- $p \rightarrow\ p \geq 0$ and is an integer that is incremented while k doesn't exceed the valid range $[K_{min}, K_{max}]$

The continual pilots have a fixed position through the OFDM symbol, and the carrier indeces are the ones in Table 2.7.

Index number k for "2k mode"						
0	48	54	87	141	156	192
201	255	2799	282	333	432	450
483	525	531	618	636	714	759
765	780	804	873	888	918	939
942	969	984	1050	1101	1107	1110
1137	1140	1146	1206	1269	1323	1377
1491	1683	1704				

Table 2.7 – Example of the continual pilots' carrier indices for 2k transmission mode

The Transmission Parameter Signaling (TPS) are pilots used to give the receiverinformation about the signal. Every symbol has 17 TPS carriers for the "2k mode" and 68 carriers for the "8k mode" and contains the same differentially encoded information bit, meaning that the TPS is defined over 68 symbols, since the TPS block is composed by 68 bits. Like the continual carriers, the TPS also have fixed positions in the OFDM symbol, these positions are shown in Table 2.8.

Index number k for "2k mode"				
34	50	209	346	413
569	595	688	790	901
1073	1219	1262	1286	1469
1594	1687			

Table 2.8 - Example of the TPS pilot carrier indeces for 2k transmission mode

The TPS block is composed by 68 bits, where the first is a initialization bit, the following 16 are synchronization bits, the next 37 bits contain information about the signal, the last 14 are redundancy bits for error protection. From the 37 information bits, 31 are used and the remaining are set to zero. The 68 bits that compose the TPS block are mapped with the following organization:

- **Initialization -** S_0: This bit is generated through the PRBS generator explained above;
- **Synchronization** - $S_1 - S_{16}$: Contains a 16 bits synchronization word which will change depending on the super-frame number. For the first and third super-frame the synchronization word is $S_1 - S_{16} = 0011010111101110$ and for the second and fourth super frame the word is $S_1 - S_{16} = 1100101000010001$;
- **TPS length indicator -** $S_{17} - S_{22}$: Contains the information about the number of bits used by the TPS block. Depending if the cell identifier is used or not the TPS length Indicator will count with that bits or not;
- **Frame number** - S_{23}, S_{24}: Contains the current frame number in the super-frame;
- **Constellation -** S_{25}, S_{26}: Contains the constellation scheme used which can be QPSK, 16-QAM and 64-QAM, respectively;
- **Hierarchy information -** S_{27}, S_{28}, S_{29}: Contains information regarding the interleaver format and the values of α used in the constellation scheme;
- **Codes rate -** $S_{30} - S_{35}$: Contains the code rates used in the transmission, since the first three bits are used to the high priority stream and the last three correspond to the low priority stream;
- **Guard Interval -** S_{36}, S_{37}: Contains the size of the Guard interval used in the OFDM symbols.
- **Transmission mode -** S_{38}, S_{39}: Contains the transmission mode used ("2k mode", "8k mode" or "4k mode");
- **Cell identifier -** $S_{40} - S_{47}$: Contains the identification of the cell from which the signal comes from. If not used, all bits must be set to zero;
- **Error protection of TPS -** $S_1 - S_{53}$: All the 53 bits after the initialization bits are extended with 14 parity bits from the BCH (67,53, t=2) shortened code, which is derived from the original systematic BCH (127, 113, t=2) code.

The TPS is modulated in DBPSK, and for every symbol conveys the same message. The PRBS generator output is also used for this modulation following the conditions bellow:

$$if\ s_1 = 0, \rightarrow\ Re\{c_{m,l,k}\} = Re\{c_{m,l-1,k}\}\,;Im\{c_{m,l,k}\} = 0 \qquad \qquad Re\{c_{m,l,k}\} = 2\cdot\left(\frac{1}{2} - w_k\right)$$

$$where$$

$$if\ s_1 = 1, \rightarrow\ Re\{c_{m,l,k}\} = -Re\{c_{m,l-1,k}\}\,;Im\{c_{m,l,k}\} = 0 \qquad \qquad Im\{c_{m,l,k}\} = 0$$

2.5

Figure 2.13 shows an example of what a OFDM frame looks like after the insertion of all the pilots and the guard interval. The blue area corresponds to the Guard Interval which is a copy of the last carriers of the symbol; the gray area is the unused carriers that are set to zero; the green squares represent the positions of the scattered pilots; the red squares represent the continual pilots; and the orange ones are the TPS pilots. The indexes shown in the figure are relative to the 2k mode transmission only.

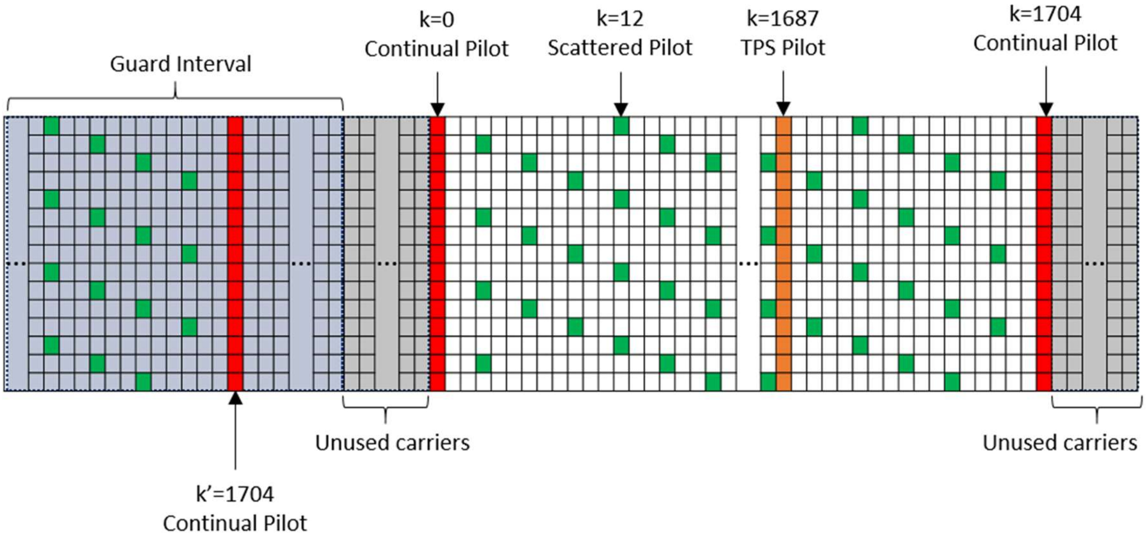


Figure 2.13 - OFDM symbol representation

Chapter 3

3 DVB-T Signal Detection & Synchronization

3.1 Introduction to Signal Sensing

Spectrum sensing plays a huge role in all the dynamic spectrum access technologies, since all of them have the function to search for unused spectrum resources and allow, or not its utilization by different users. One of the most relevant technology is the Cognitive Radio (CR), which uses opportunistic access to allow spectrum users to use the same frequencies [6]. In this technology, the detection speed and precision is a key factor to ensure the QoS and fidelity of the system to the user.

In the LSA model, the spectrum sensing concept is one of the cores of this model. It works around a repository with the statuses of a spectrum resource, which are gathered by spectrum sensing techniques. Although detection speed isn't necessarily a main characteristic for the spectrum sensors, in the LSA model, precision and liability must be achieved to ensure a good reading on the spectrum for a proper decision on which user can transmit its signal.

Several spectrum sensing techniques are used to gather information about the spectrum and each one has advantages and disadvantages depending on the objective and characteristics of the detector. The three most common types of detection are the energy detection, the matched filtered detection and the cyclostationary signatures detection. The energy detection method is used when the detector knows very little about the signal to be received. This kind of detection measures the energy of the received signal over a time interval, and comparing that energy to a defined threshold, it can declare if there is, or not, a signal. Although this method can only give information about the energy of the signal, it is a relatively simple method to implement and it's widely used for spectrum sensing [7][8]. The matched filtered detection method gives the best results in terms of detections, however the detector needs to know all the information about the transmitted signal [8][9]. This method works with the correlation of the received signal with, a known part of the signal, normally, the pilots since they are a well known part of the signal [10]. In the cyclostationary signature detection a known sequence is inserted in the signal periodically, and it is this periodicity that allows the detector to verify the presence of the signal. Using the correlation with the received signal and the signature, the detector is able to detect the signal being emitted [11]. This method implies a small modification to the transmitter, since it is necessary to add the signature to the signal and although it is relatively easy to implement, it requires a large amount of calculations by the receiverto be able to detect the signatures [8].

3.2 DVB-T Detection Techniques

Several techniques are used to detect DVB-T signals, they are based on the sensing techniques explained above and on the characteristics of the DVB-T protocol, like the continual/scattered pilots and the cyclic prefix present in every symbol of the signal.

One of the most common techniques used for signal sensing is the auto-correlation of the received signal [12], in the DVB-T signal. This method works due to the presence of the cyclic prefix (CP) in each symbol. Since the CP is a copy of the end of each OFDM symbol, it acts as a cyclostationary signature of itself and allows the auto-correlation to achieve maximums at the beginning of each symbol. The auto-correlation with the guard interval of the signal is obtained with equation 3.1, where $c[n]$ is auto-correlation signal, $r[n]$ is the received signal, N_{FFT} is the length of the FFT used, L_G is the length of the CP used and n correspond to the n^{th} OFDM symbol. The auto-correlation signal, $c[n]$, reachs its maximum when $r^*[n+i]$ and $r[n+i+N_{FFT}]$ correspond to the positions of the symbols guard interval and the end of the same symbol (represented in the Figure 3.1).

$$c[n] = \sum_{i=0}^{L_G-1} r^*[n+i] \cdot r[n+i+N_{FFT}] \tag{3.1}$$

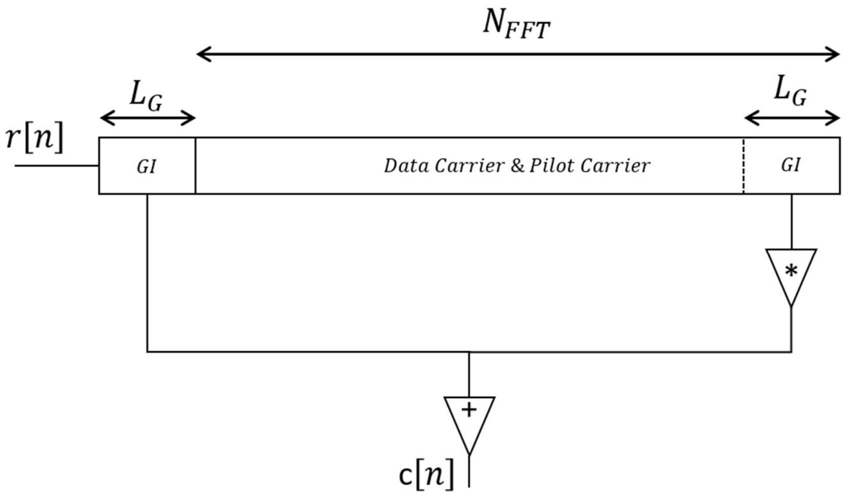


Figure 3.1 - DVB-T signals auto-correlation with the guard interval

Another method widely used to detect signals is the cross-correlation with the protocol pilots, in the case of the DVB-T protocol it is possible to correlate the continual and/or the scattered pilots. The principle is the same as the auto-correlation explained before but this time, instead of correlating the signal with a delayed part of it, it correlates a vector with the protocols pilots[13]. For this technique to work both the signal and the pilot vector must be in the same domain, either in the time domain or in the frequency domain. Equation 3.2 represents the cross-correlation with the received signal and the continual pilots in the time domain. The sum is maximum when the pilots' position of both the received signal and the pilots vector are aligned within the symbol.

$$c[n] = \sum_{i=0}^{N_{FFT}-1} r^*[n+i] \cdot P[i] \ , \qquad i \ \in Pilots'position \tag{3.2}$$

A major problem with this method is the frequency offset of the received signal, the detector performance decreases and it is necessary to compensate that offset to obtain better results.

The cross-correlation and the auto-correlation are the main methods to identify the DVB-T signals, and many of methods used for this purpose are based on these two methods and, normally, some variations, that improve the system behavior in certain conditions, are added. For instance, a variation of the auto-correlation method presented in [14], uses an adapted algorithm of the auto-correlation in order to attenuate the effect of the multipath in the detection results, as shown in Figure 3.2. This technique allows to detection of the signal even with the presence of multipath effect and although the presence of multipath is still visible after the correlation, with the proper threshold it's possible to detect the signal.

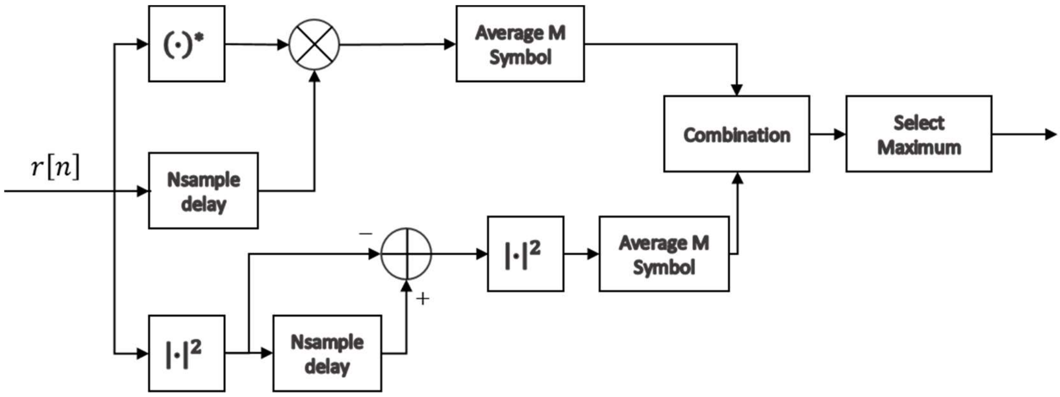


Figure 3.2 - Jiang Fang proposal for CP correlation method able to reduce influence of multipath [14].

3.3 DVB-T Synchronization

Every OFDM signal suffers from imperfections when transmitted, they are affected by the different types of transmissions channels, which can cause time, carrier frequency, phase and sampling frequency offsets to the signal when it is received at the receiver. All these problems can be solved or diminished by a good synchronizer at the receiver. Its goal is to compensate all those offsets and allow the receiver to read and decode the signal properly with as less error as possible. Time offsets cause inter-carrier interference (ICI) and inter-symbol interference (ISI). Both occur because the OFDM symbols are misplaced in the FFT window, since the carrier position is not the correct one the receiver will get a value for a specific carrier that is not the supposed value. Carrier Frequency offsets are mainly caused by the difference from the transmitter and the receiver local oscillators. In this case the signal spectrum is shifted accordingly the offset frequency and the FFT window will get the carriers values shifted, causing ICI and phase offsets. The sampling frequency offset occurs because of the offset between the clocks of the transmitter DAC and receiver ADC. This causes the ADC to sample the signal at different times from the DAC as shown in Figure 3.3.

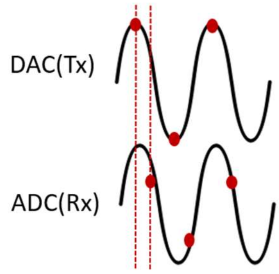


Figure 3.3 - Illustration of sampling frequency offset

In the case of the DVB-T protocol, the synchronization techniques used are somehow similar to the techniques used to detect this kind of signals. Both rely on cross and auto-correlation to identify the beginning of the OFDM symbols and the protocol pilots to identify the carriers position in the OFDM symbol.

A DVB-T receiver synchronizer can be composed by two stages of synchronization, a Pre-FFT synchronizer and a Post-FFT synchronizer [15]. The time offset is corrected in the Pre-FFT stage where the auto-correlation of the received signal is used as explained in equation 3.1. Since this correlation results in peaks at the beginning of each symbol, it is possible to adjust the carrier positions, so that, the first carrier detected corresponds to the first carrier of the OFDM symbol. This way, a coarse time synchronization of the signal is achieved. The frequency synchronization is firstly done in the Pre-FFT stage also using the auto-correlation with the guard intervals to detect the beginning of the symbol. In order to improve the symbol detection and the results of the offset estimation a metric can be used, which optimizes the results of the auto-correlation

The frequency offset is roughly estimated in the Pre-FFT stage. After the estimation of the frequency offset, a Post-FFT synchronization can be done, where the cross-correlation of the signal with the pilots of the DVB-T protocol is used, as explained with the equation 3.2. This way, the shifted carriers are positioned at the right position eliminating the frequency offset of the signal.

Chapter 4

4 LSA Warner Proposal

The diagram shown in Figure 4.1 illustrate the major blocks of the LSA Warner to be developed. The main goal of the system is to sense the presence of any signals at the frequencies band of 2,3 and 2,4 GHz, verify if the incumbent's DVB-T signal, for that frequency, has the proper conditions to be transmitted and send that information, alongside with the system's geolocation, to an online database.

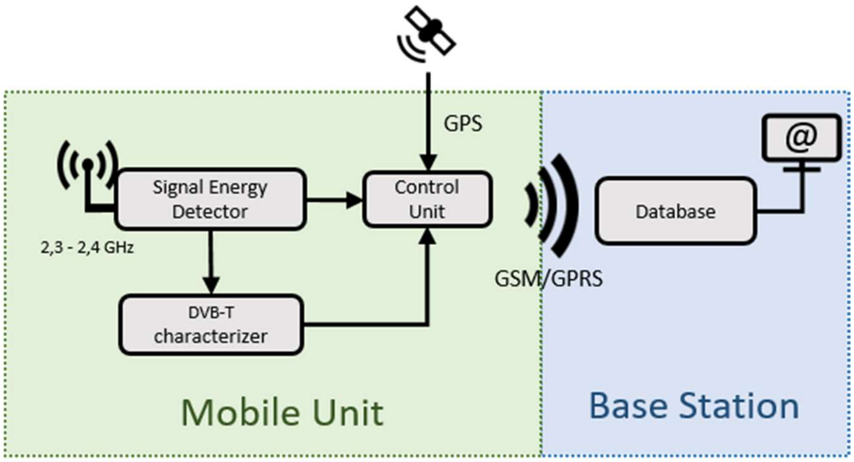


Figure 4.1 - Proposal for detection of DVB-T signals

The signal energy detector consists in a RF circuit that can convert the received signal energy into a DC tension. This circuit must detect signals with, at least, -80 dBm and, ideally, only in the frequency bands desired (2,3-2,4 GHz). The DVB-T characterizer has the objective of demodulating the received signal, verify if it is an DVB-T signal and compare the signal constellation with the expected constellation to ensure the signal quality.

The energy level of the received signal, constellation condition values and the geolocation of the system, is then gathered by a control unit to be sent, through GSM/GPRS connection, to an online database for further presentation in the website.

4.1 Signal Energy Detection

The first module developed was the signal energy detection and several stages were needed to achieve the requirements presented above:

- **Amplifier Stage:** Essential to allow the RF detector to detect signals in the order of the -80 dBm. Was assembled an RF amplifier with enough gain to achieve the pretended goal, it was then tested to verify the gain of the circuit;
- **Filter Stage:** This stage limits the frequency bandwidth. A band pass filter was designed, simulated and implemented for the 2.3 – 2.4 GHz band. After implementation, it was measured and tested;
- **Limiter Stage:** This stage works as a protection stage for the RF detector. Since the detector is sensible to large signals, it was necessary to protect it from those signals;
- **RF Detector Stage:** A multistage was assembled logarithmic amplifier integrated circuit able to convert RF signals into DC voltage. It was first tested with signals within the frequency band to verify the DC range that the convertor can produce.

After testing and verifying the performance of all components individually, a circuit with all the components together was created and then tested.

4.1.1 Amplifier Stage

A search was made to find an amplifier with high gain and very small dimensions, the MNA-6A+ from the Mini – Circuits® was a good match for the requirements of this systems. The amplifier requires only one voltage supply (2.8v-5v), it is well matched for 50 Ω and it is a very straightforward IC to mount, all those characteristics were decisive to select this amplifier to incorporate the system.

The first step was to test individually the amplifier and to do that, a PCB design was made according to the amplifier's datasheet [16]. Figure 4.2 is the layout for the amplifier testbed, where R1 correspond to a 33 Ω resistance and C1 to a 100pF, a 10nF and a 2.2μF capacitors. To measure the amplifier, a network analyzer was used to obtains the S parameters shown in Figure 4.3.

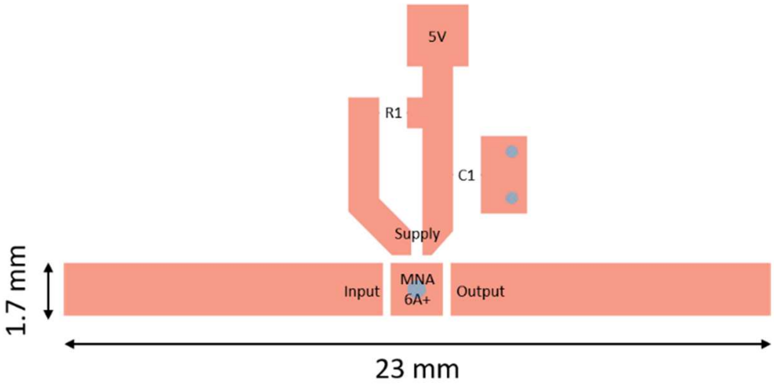


Figure 4.2 - PCB for a single MNA-6A+ amplifier

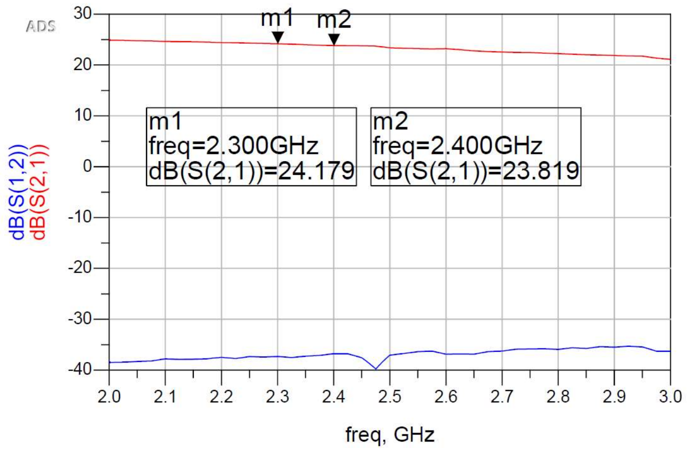


Figure 4.3 - Single MNA-6A+ amplifier results

Even if the obtained gain for a single amplifier was a good one, to detect at least -80dBm, a bigger gain was necessary. Two MNA 6A+ in series were then tested, in theory, to duplicate the gain. This new testbed was made like the previous one, but this time with two slots for two amplifiers as shown in Figure 4.4.

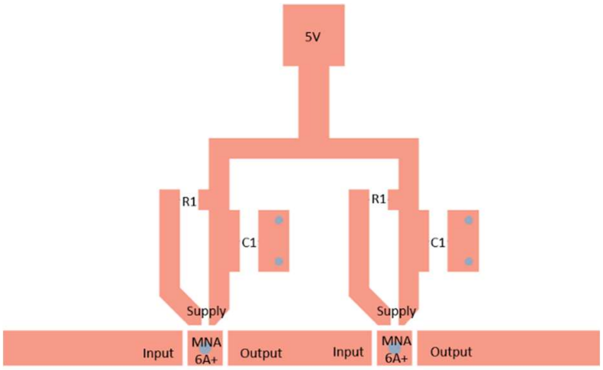


Figure 4.4 - PCB for two MNA-6A+ amplifiers in series

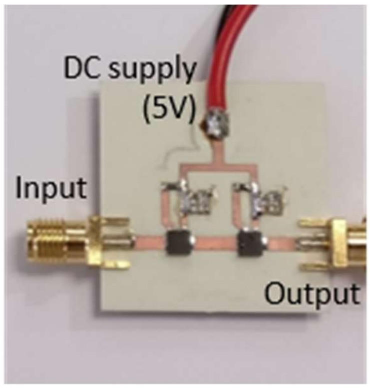


Figure 4.5 - Test board for the amplifier stage

The results obtained for the two amplifiers in series are shown in Figure 4.6. These results were obtained by inserting a tone at the amplifier input and measure the output with a spectrum analyzer. Gains between the 45 and 46.5 dB were obtained for the frequency band of 2.3 and 2.4 GHz; an amplifier stage with these gains is enough to allow the energy detector to detect -80dBm signals.

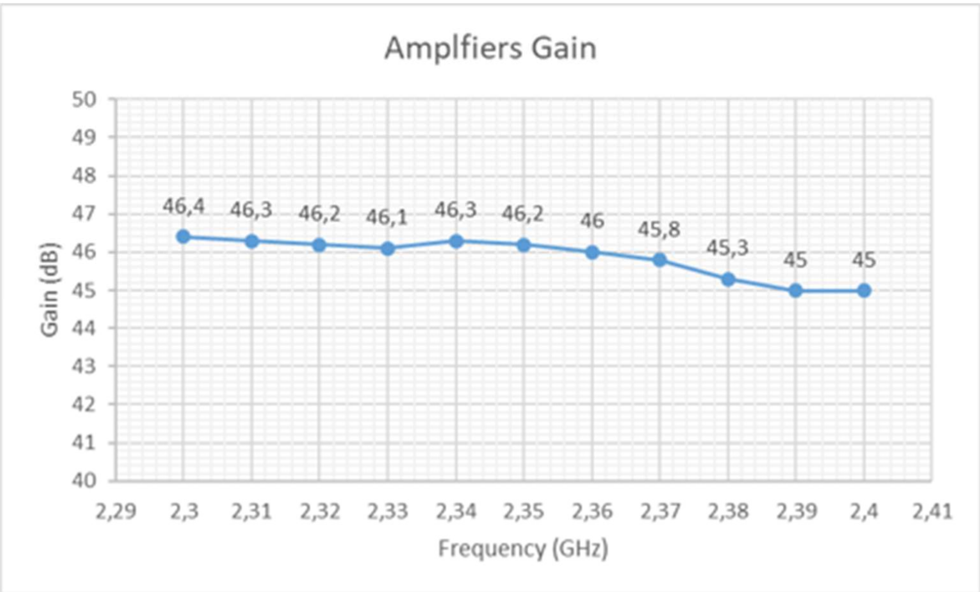


Figure 4.6 - Results for the two MNA-6A+ amplifiers in series

4.1.2 Filter Stage

A RF filter is used to control the frequency response of an RF circuit and is widely used in wireless communication systems. Several frequency responses are used such as high-pass, low-pass, band-pass and band-stop. For this solution, a band-pass filter is the desired response for the system, the wanted frequencies are the ones between the 2.3 and 2.4 GHz and any other frequency outside this band should be attenuated.

To implement the band-pass filter a microstrip coupled line filter was a good option since it allows narrower frequency bands [17]. This kind of filter makes use of the electromagnetic fields interaction of the proximal lines to control the frequency response of the circuit; the interaction with two proximal lines can be represented as capacitors between lines and modifying the space and position of the lines allows us to get the proper frequency response to create band-pass filter.

With the help of $ADS^{®}$ simulator and after some optimization, a band-pass filter with a proper frequency response was achieved and it is shown in Figure 4.7, which is the layout of the filter. Figure 4.8 shows the printed filter with SMA ports to set up the measurements. The filter has 19 mm width and 93.2 mm length and the rest of the filter dimensions are described in Table 4.1.

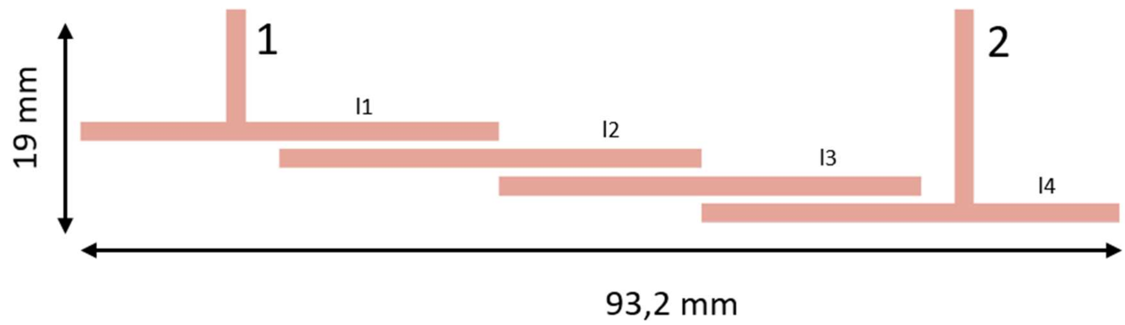


Figure 4.7 - PCB layout to the BPF designed

Parameter		Size	Parameter		Size	Parameter		Size
Length	l1	37.4 mm	Width	l1	1.64 mm	Gap	l1-l2	0,7 mm
	l2	37,9 mm		l2	1.64 mm		l2-l3	0,8 mm
	l3	37,9 mm		l3	1.64 mm		l3-l4	0,7 mm
	l4	37.4 mm		l4	1.64 mm			

Table 4.1 - Designed Filter dimensions

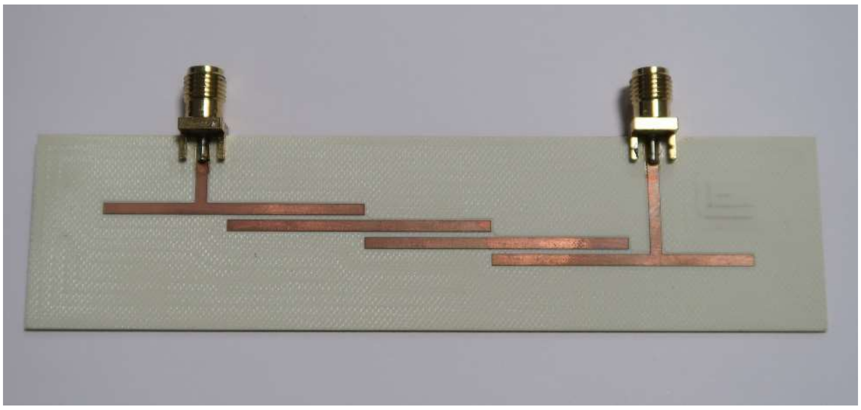


Figure 4.8 - Test board for the designed filter

The simulated frequency response for this filter is compared to the measured frequency with a virtual network analyzer shown in Figure 4.9 and Figure 4.10; the S parameters S11 and S12 for the designed filter are compared and since the filter is symmetric, the S22 and S21 are equal to S11 and S12, respectively.

From the measurements made to the filters, it's possible to see in the frequency band of 2.3 to 2.4 GHz, an attenuation only between 1.76 and 2 dB, which are acceptable values to use this filter for the LSA warner system.

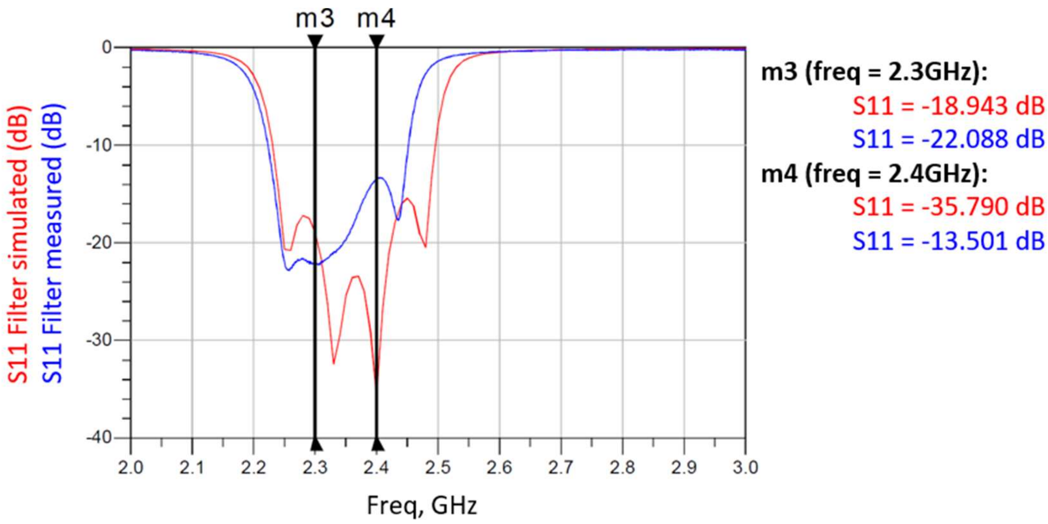


Figure 4.9 - Simulated and measured S11 parameter

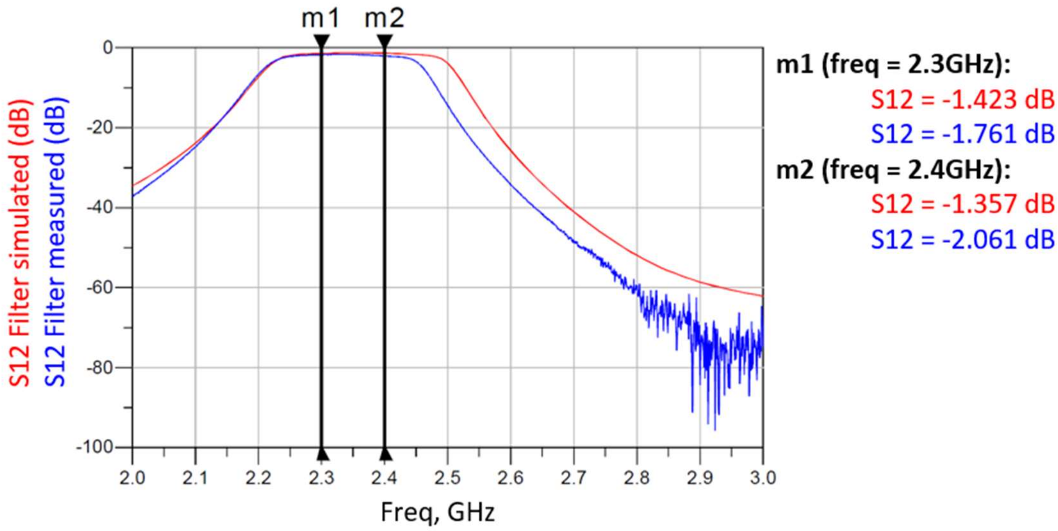


Figure 4.10 - Simulated and measured S12 parameter

4.1.3 Limiter Stage

The RF limiters are used to protect power-sensitive components from high power signals, and in this case, to protect the energy detector at the end of the circuit. This limiter is placed after the amplifier and filter stage to prevent possible high-power signals to destroy the detector.

The maximum power that the detector can receive at the frequency of 2.5GHz is -5dBm with the absolute maximum input power of 19dBm [18], a limiter with the capability to limit signals with high power (at least 20dBm) to -5dBm was ideal, however, it is very hard to get this kind of limiter and the ones found were very expensive, so the RLM -23 -1WL+ from Mini – Circuits® was chosen to implement in our system. This limiter can handle signals up to 30 dBm and limit them to 0 dBm, and even if it is above the range of the RF detector, it won't reach the detector's absolute maximum value. A testbed was made to check the limiter capabilities and both the PCB and the circuit are shown in Figure 4.11 and Figure 4.12.

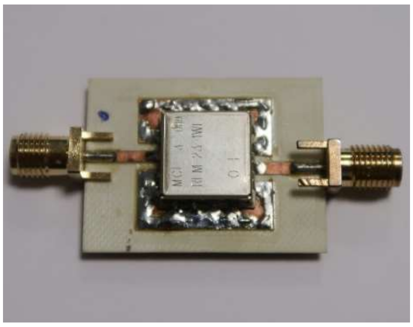


Figure 4.11 - Limiter test board

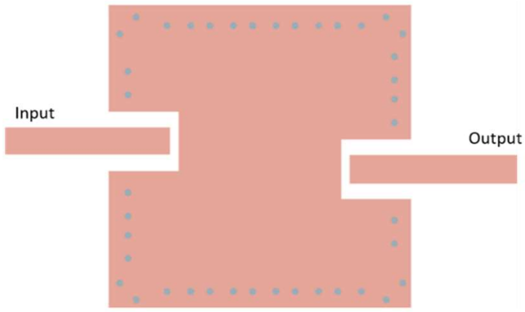


Figure 4.12 - Limiter PCB test board

Like the test made for the amplifier stage, the limiter was tested with a tone as input and measured the output signal with a spectrum analyzer; the obtained results presented in Figure 4.13 show that with an input signal with 18dBm, the output signal doesn't exceed the -1dBm. Since the amplifier output power at 1dB compression is around 20dBm, the detector won't receive signals with more than -1 dBm and it will be protected to high power signals.

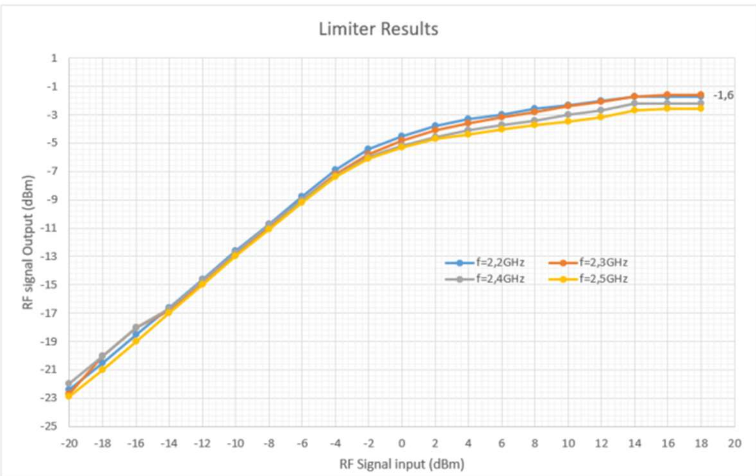


Figure 4.13 - Limiter input vs output signal attenuation

4.1.4 RF Detector Stage

The final stage is the RF detector, our goal is to convert the RF signal energy to a corresponding DC voltage, which can be read by the microcontroller at the control unit for further use. Our concern with this stage was the capability to capture signals at low power levels (around -80dBm), even with the amplifier stage which allows even further detection. The detector itself needed to be able to capture low power levels and the lower the better.

For that, it was picked the MAX2015 from Maxim Integrated®, which is a multistage logarithmic amplifier designed to convert RF signals into DC voltage, that can detect signals with -40dBm at the frequency of 2GHz. This IC was selected not only for the price, but also for the capability to detect much lower power signals for that frequency and that size. Like the other stages, a testbed for this IC was created as shown in Figure 4.14 and Figure 4.15.

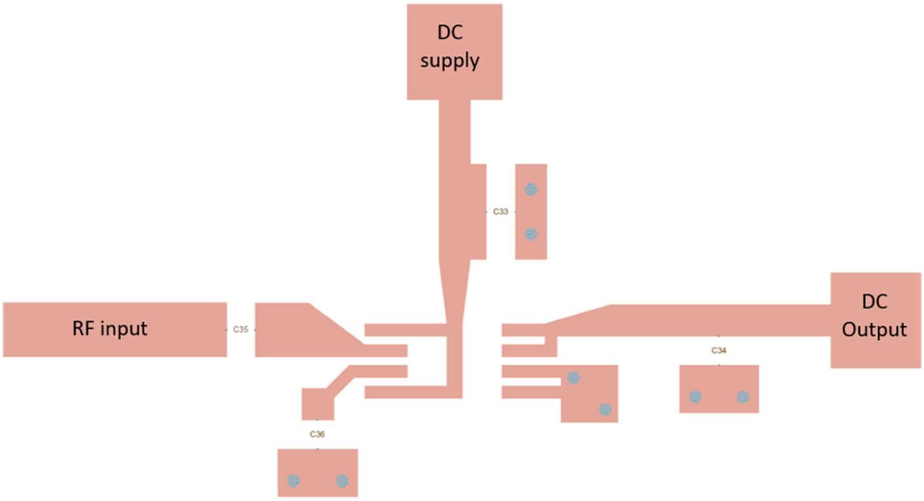


Figure 4.14 - PCB layout for the MAX2015

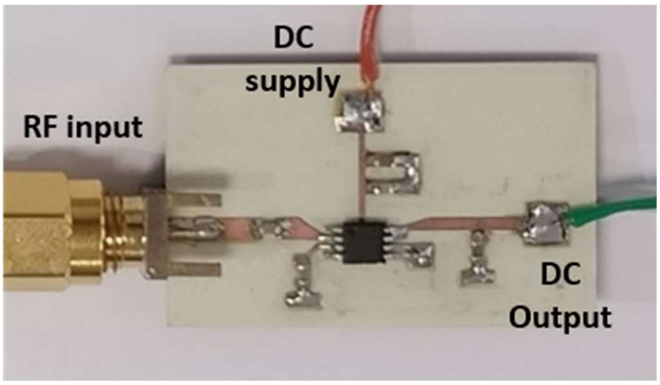


Figure 4.15 - PCB test board for the MAX2015

The obtained results in Figure 4.16 show that when turned on, the MAX2015 has a threshold of 0.6V that was decided to be the "No Signal" threshold, and so with an input tone of -40dBm the DC voltage slightly increased meaning that the detector is able to capture signals with -40dBm by itself, and that's is why the amplifier stage has a gain around 40dB to reach the -80dBm signals.

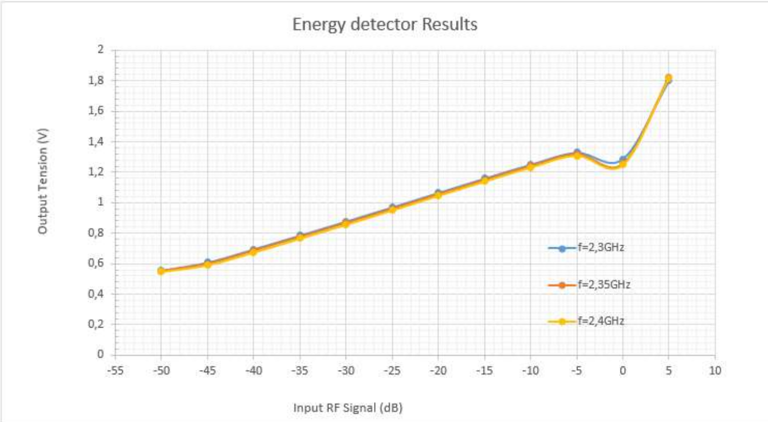


Figure 4.16 - Results for the MAX2015

4.1.5 Complete Circuit

After verifying the proper functionality of all the components, it was assembled a circuit with all of them and tested. The PCB design is shown in Figure 4.17 and after mounting the components, the board (presented in Figure 4.18) was tested to verify that all the components were properly working. The voltage supply for all the components is the same, 5V, and so to ensure a constant voltage, a power supply was added just to prevent any variation to the components' supply voltage. The results obtained are presented in Figure 4.19, and the circuit has "No Signal" threshold around 0.65V and to the frequency band of 2.3-2.4 GHz the circuit can detect signals up to -80dBm. It is observable the limiter working properly when the signal's power is increased, the filter's frequency response is also noticeable since for frequency outside the band the DC level obtain is much lower from the in-band frequencies.

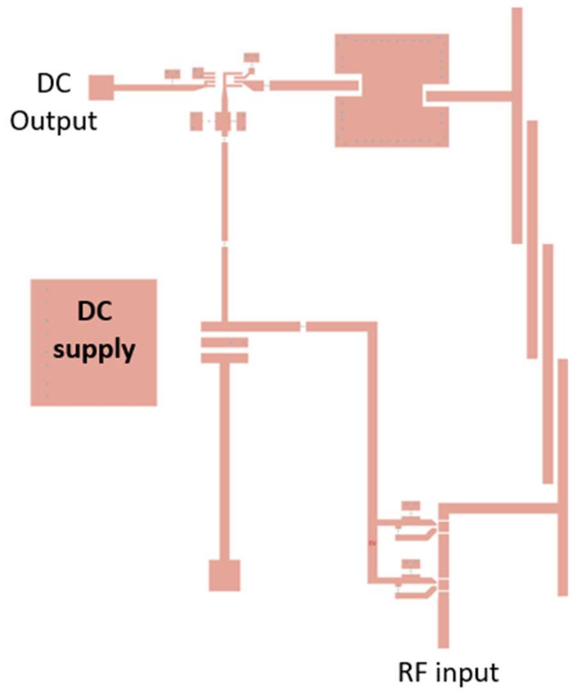


Figure 4.17 - PCB Layout to for the LSA Warner

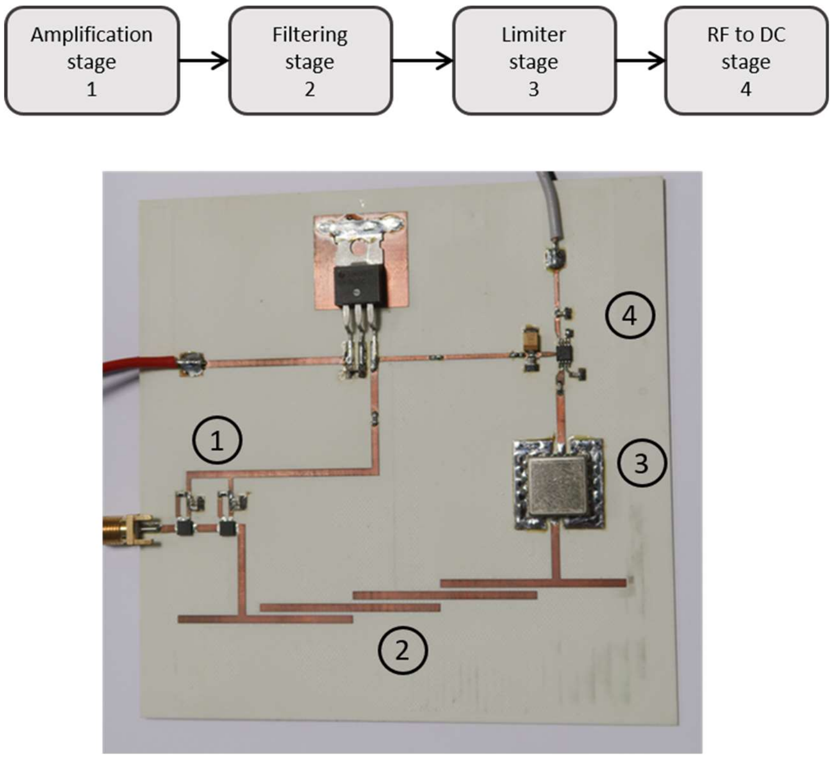


Figure 4.18 - Prototype board for the LSA Warner

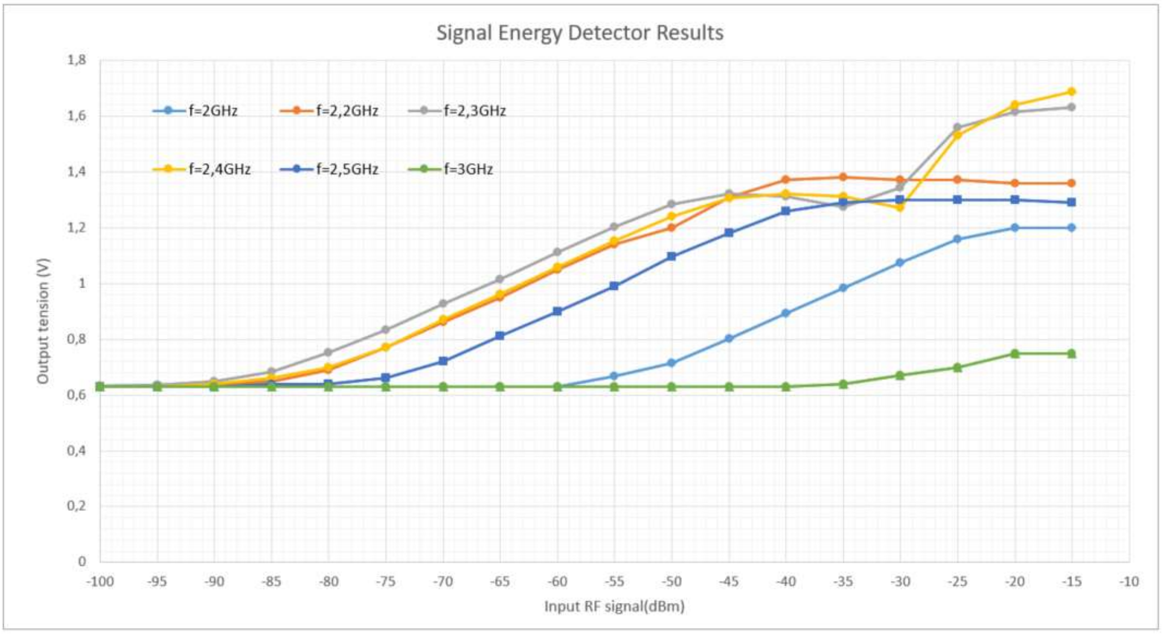


Figure 4.19 - Results to the LSA Warner prototype

4.2 Control Unit & GPS Module

After the development of the signal energy detector, the control unit for the LSA warner was implemented. This control unit has the function to gather all the information from the modules attached to it, treat it and send it to the database. An Arduino board was picked to work as the microcontroller for the unit, because it is very versatile and has compatibility with the several components that will be attached to it, such as the GPS module and the GSM/GPRS shield used for the 3G/4G connection to the database.

To receive the GPS signals, a simple module was acquired to read the GPS signals and give coordinates and data/time of the module to the user. Due to the simplicity of the GPS Module, it is only necessary to connect it to the Arduino board. From the Arduino side it only needs to read the information gathered from the module outputs and treat it accordingly the protocol presented at the datasheet of the module [19]. Below is a pseudo-code of the interaction between the Arduino and GPS module.

```
include GPS module Library

while (1) {
   if (GPS signal is available) {
      return get_locationCoordinated ();
      return get_date ();
   }
   else return invalidGPS_string;
   delay(9000ms);
}
```

Another function of the control unit is to read the DC voltage at the energy detector output and the information from the DVB-T characterizer, and for that the arduino needs to read that information and save it. A pseudo-code about this functionality is shown below. For a better approximation of the energy detector output, a median of the value is done to ensure a good approximation.

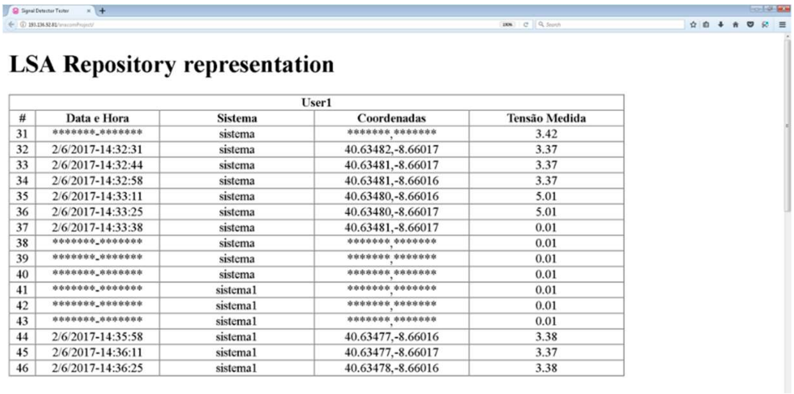
```
#set analogic port1 to input
#set analogic port2 to input

while (1) {
   read_DVB-t_Characterizer ();
   for (i=0; i<50; i++) {
       tensionIn = read_AnalogInput;
       medTensionIn = (medTensionIn + tensionIn)/2;
   }
   Return medTensionIn;
   delay(9000ms);
}
```

Lastly, the Control unit must gather all the information read and send it to the database, and for that a GPS/GPRS Arduino shield with a SIM card was used. The shield needs the information about the server database to establish a connection between both, and send PhP “GET” commands to the server’s console which will write the information into the database. A pseudo code below shows the steps to access the server with GSM/GPRS Arduino shield.

```
      #include GSM module library
      Setup {
      While(notConnected)
            try to Connect SIM card to the operator;
            delay(1000ms);
      }
      while (1) {
         if (database's server is available & client is connected)
   {
            Print GET commands into database's server console with
      information gathered;
         }
         delay(9000ms);
      }
```

To check the functionality of the control unit, a very simple MySQL database alongside with PhP scripts were created. Their function is to check if there is any information to receive, read it and store it in the server. This database is composed by five columns and for every reading a new line for each column is added. The information stored in the database server consists in: an ID, which is attributed to every reading and is unique; the date/time read by the GPS module; the name of the system where the information came from; the location of that system, also gathered by the GPS module; and the DC Voltage which is the output of the energy detector.

To better see the information stored in the server, a simple website was developed, to display the information present in the database and update with the new readings, in Figure 4.20 is displayed a print-screen of the website.



LSA Repository representation

User1				
#	Data e Hora	Sistema	Coordenadas	Tensão Medida
31	*******_*******	sistema	*******,*******	3.42
32	2/6/2017-14:32:31	sistema	40.63482,-8.66017	3.37
33	2/6/2017-14:32:44	sistema	40.63481,-8.66017	3.37
34	2/6/2017-14:32:58	sistema	40.63481,-8.66016	3.37
35	2/6/2017-14:33:11	sistema	40.63480,-8.66016	5.01
36	2/6/2017-14:33:25	sistema	40.63480,-8.66017	5.01
37	2/6/2017-14:33:38	sistema	40.63481,-8.66017	0.01
38	*******_*******	sistema	*******,*******	0.01
39	*******_*******	sistema	*******,*******	0.01
40	*******_*******	sistema	*******,*******	0.01
41	*******_*******	sistema1	*******,*******	0.01
42	*******_*******	sistema1	*******,*******	0.01
43	*******_*******	sistema1	*******,*******	0.01
44	2/6/2017-14:35:58	sistema1	40.63477,-8.66016	3.38
45	2/6/2017-14:36:11	sistema1	40.63477,-8.66017	3.37
46	2/6/2017-14:36:25	sistema1	40.63478,-8.66016	3.38

Figure 4.20 - Website to display the database contents

4.3 DVB-T Characterizer

As explained before, the DVB-T characterizer has the goal to receive a DVB-T signal and demodulate it in order to evaluate the constellation and verify if the signal is being transmitted with the proper conditions. For this module was simulated, a DVB-T transmitter, a simple transmission channel, which induces noise, frequency offset and time offset to the transmitted signal, and a receiver able to compensate for those errors and demodulate the signal and display the constellation.

The transmitter in this simulation was created with the possibility to generate a DVB-T signal with all the parameters supported by the standard (except for the transmission mode "4k mode"). At the beginning of the simulation the user is asked to choose what parameters he want to use in the creation of the DVB-T signal, to allow the generation of the OFDM parameters, holding all the information about the OFDM symbol such as FFT size, guard interval size, number of data carriers and pilot carriers, and carrier duration. After those processes are done, random binary information is generated in order to always achieve four complete frames, making a super frame.

```
%% Choose initial parameters for the transmited data
    transmitionInitParam = struct;
%Choose transmition mode: 2kmode and 8kmode
    transmitionModeOptions = {'2kmode' , '8kmode'};
    transmitionInitParam.Mode = transmitionModeOptions(1);

%Choose channel spacing: 8MHz, 7MHz and 6MHz
    channelSpacingOtions = [8 7 6];
    transmitionInitParam.channelSpacing = channelSpacingOtions(1);
```

```
%Choose modulation type: qpsk, 16qam or 64qam
    modTypeOptions = {'qpsk';'16qam';'64qam'};
    transmitionInitParam.modType = modTypeOptions(2);

%Choose modulation type alfa parameters: alfa=1, alfa=2, alfa=4
% alfa = 2 and 4 only aplies to 16qam and 64qam modulation
    modTypeOptions = [0 1 2 4];
    transmitionInitParam.modType_alfa = modTypeOptions(1);

%Choose code rate: 1/2 , 2/3 , 3/4 , 5/6 or 7/8
    codeRateOptions = [1/2 2/3 3/4 5/6 7/8];
    transmitionInitParam.codeRate = codeRateOptions(1);

%Choose guard interval
    guardIntervalOptions = [1/32 1/16 1/8 1/4];
    guardInterval = guardIntervalOptions(1);

%choose Interleaver Format
    InterleaverFormatOptions = {'inDepth' ; 'native'};
    transmitionInitParam.interleaverFormat =
InterleaverFormatOptions(1);
```

All the random binary information is then processed with the DVB-T transmission chain, presented in Chapter 2, and converted to time domain using the inverse FFT with the proper size to the transmission mode selected. From Figure 4.21 to Figure 4.25 the results of the DVB-T signal, before the IFFT, for some combinations of the parameters available in the protocol are shown, the number of binary samples generated for the DVB-T signal composition are also displayed.

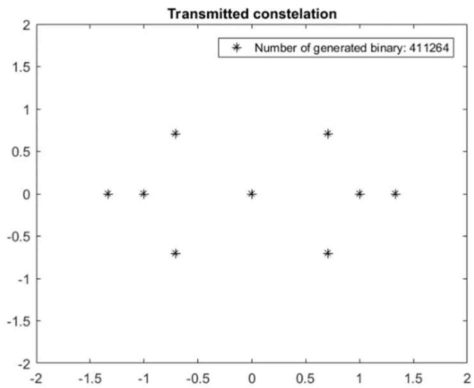


Figure 4.21 - Generated signal for a QPSK in 2k mode with a 1/2 code rate

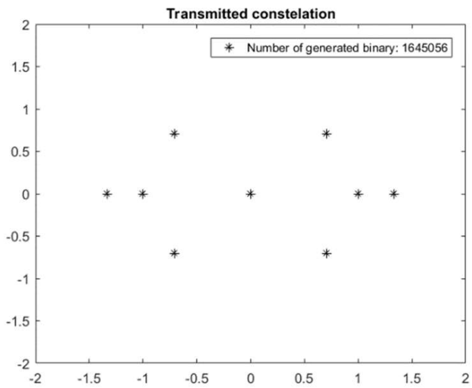


Figure 4.22 - Generated signal for a QPSK in 8k mode with a 1/2 code rate

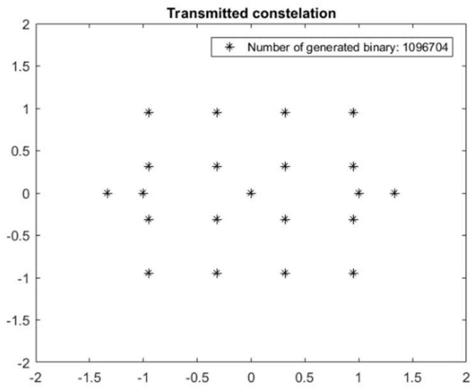


Figure 4.23 - Generated signal for a 16-QAM in 2k mode with a 2/3 code rate

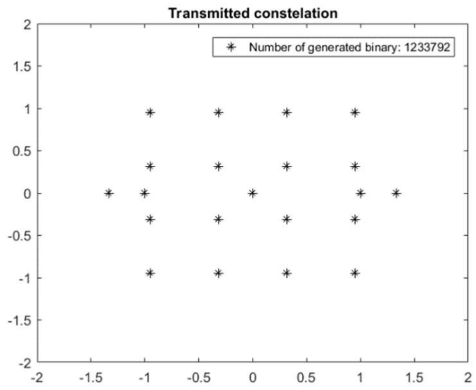


Figure 4.24 - Generated signal for a 16-QAM in 2k mode with a 3/4 code rate

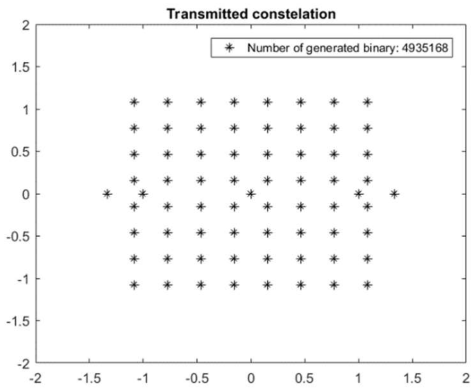


Figure 4.25 - Generated signal for a 16-QAM in 2k mode with a 2/3 code rate

For the channels simulation, a simple channel was developed, able to impose some errors to the signal, such as noise, time offset and frequency offset. To add white Gaussian noise to the signal a *MATLAB* function, *awgn()*, was used, which add to a complex signal, noise with a given SNR. Time offset was made by copying a random part of the signal to the beginning, in this case it was important to not copy samples from the guard interval. This way it acts as random information added at the beginning of the signal. The frequency offset was made by multiplying the signal with complex exponential, as shown in the following equation:

$$signal_{offset} = signal_{received} * e^{j2\pi f_{offset}t} \tag{4.1}$$

The f_{offset} is the wanted frequency offset in hertz and t is a vector, which is generated according with the elementary period of each carrier and the total number of the signal samples.

```
%add noise to the signal
snr = 35;   %dB

Tx_OFDM_noise = awgn (Tx_OFDM, snr,'measured');

%add time offset to the signal
samplesOffset = 300;    %samples

Tx_OFDM_timeOffset = [ Tx_OFDM(100:100+samplesOffset) Tx_OFDM];

%add frequency offset to the signal
freqOffset = 50;         %Hz
T = (7/64)*10^-6;        %Elementary period of the carriers
t=T:T:T*2112*272;        %time vector for the specified Carrier time.

Tx_OFDM_offset = Tx_OFDM.*exp(1i*2*pi*freqOffset*t);
```

Besides of the demodulation, the receiver is able to synchronize the received signal for a proper demodulation. The synchronizer was developed based on the structure of a synchronizer explained in chapter 3, which has a pre-FFT synchronizer and a post-FFT synchronizer. The core of the pre-FFT synchronizer is based on the auto-correlation of the guard intervals, as explained before, however, for better results a metric was used to optimize the correlation results. This method makes the symbol detection more efficient and therefore a better synchronization. The metric used is the Maximum Normalized Correlation (MNC) because of its results presented in [15] and its low complexity. The main characteristics of this metric is that it considers the power of the input signal and estimates the frequency offset with an optimized signal of the correlated signal. The following expression explain how the offset is calculated:

$$\varepsilon_{MNC} = \arg\left(\max_{\mathrm{n}}(m[n])\right) = \arg\left(\max_{\mathrm{n}}\left(\frac{c[n]}{p[n]}\right)\right) \quad 4.2$$

The ε_{MNC} is the frequency offset estimated for each of the maximums found in the relation of $\frac{c[n]}{p[n]}$, where $p[n]$ is the power of the signal which can be calculated by:

$$p[n] = \sum_{i=0}^{N_{pow}-1} |r[n+i]|^2$$

4.3

With the auto-correlation of the signal, a maximum is obtained at every first sample of an OFDM symbol with the guard interval, as can been seen in Figure 4.26 to Figure 4.28. Using the MNC metric the signal is significantly optimized and maximums are obtained with greater amplitude and allowing the use of a more liable threshold for the symbol detection.

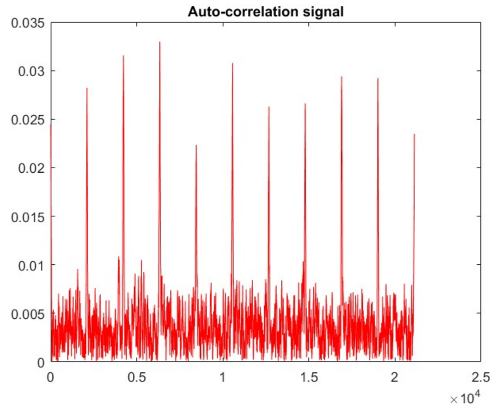


Figure 4.26 - Auto-correlation of the received signal

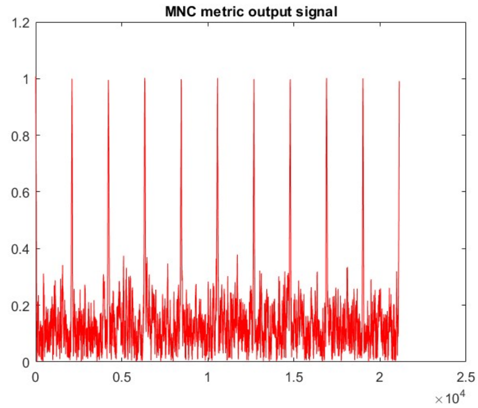


Figure 4.27 - Result of the MNC metric applied to the auto-correlation signal without the time offset

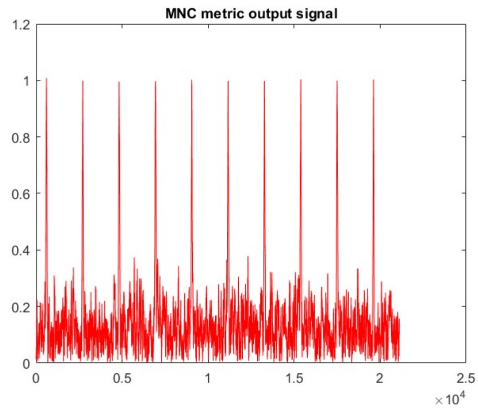


Figure 4.28 - Result of the MNC metric applied to the auto-correlation signal with time offset

Since, the size of the symbol and the guard interval of that signal are known, it is possible to readjust the carriers positions to the desired ones and remove the guard interval from the symbol. Also, with the complex value of the maximums, the frequency offset for each symbol is estimated, individually, using the equation 4.2. This estimated offset is then applied to the signal through the complex exponential with the following expression:

$$signal_{offset} = signal_{received} * e^{j2\pi\cdot(l(FFT_{size}+GI_{size})+k)\cdot\varepsilon_{MNC}} \quad 4.4$$

In the expression above, $k \in [0\ , 2048]$ (for the 2k transmission mode) corresponds to the number of l^{th} symbol. FFT_{size} and GI_{size} are the number of carriers used in the symbol and in the GI size, respectively. This compensation works until the offset reaches 2π radians deviation from the original constellation, after that, this method cannot distinguish the proper rotation angle of the constellation. After the pre-FFT synchronization, the signal passes through the FFT block to convert the signal into the frequency domain. The results obtained with the synchronizer are displayed after the FFT module because they are more noticeable in the frequency domain than in the time domain. From Figure 4.29 to Figure 4.38 the received signal after the FFT block is displayed, with and without the synchronizer influence and the behavior of the synchronizer is displayed for different frequency offsets ($f_{offset} = 1, 50, 1000, 2500, 4500\ Hz$).

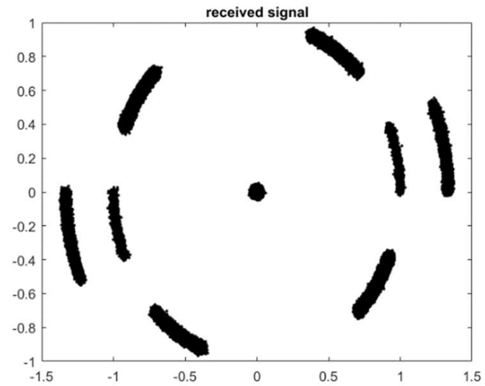


Figure 4.29 - Received Signal with an offset of 1 Hz

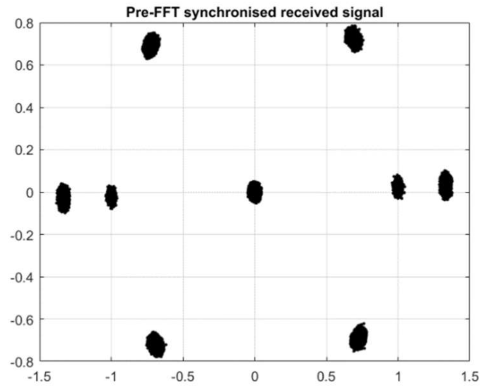


Figure 4.30 - Pre-FFT synchronizer output for an offset of 1 Hz

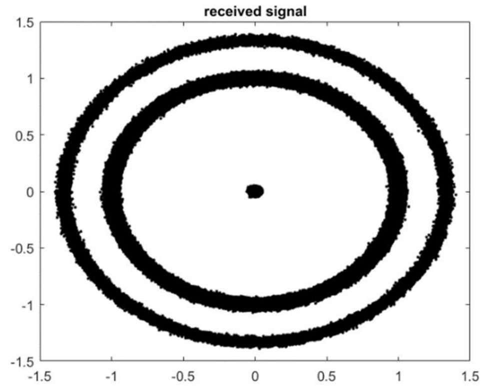


Figure 4.31 - Received Signal with an offset of 50 Hz

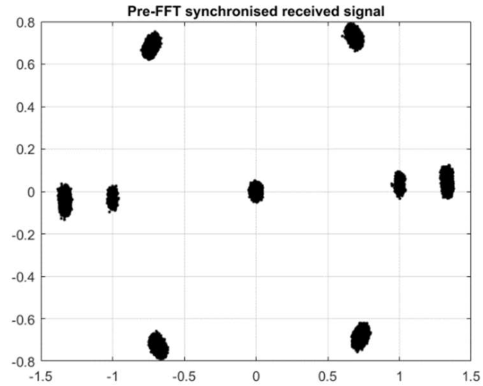


Figure 4.32 - Pre-FFT synchronizer output for an offset of 50

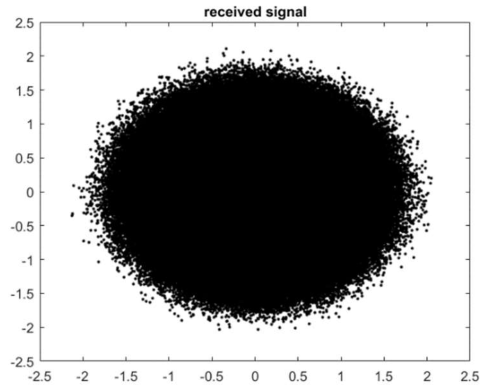


Figure 4.33 - Received Signal with an offset of 1 kHz

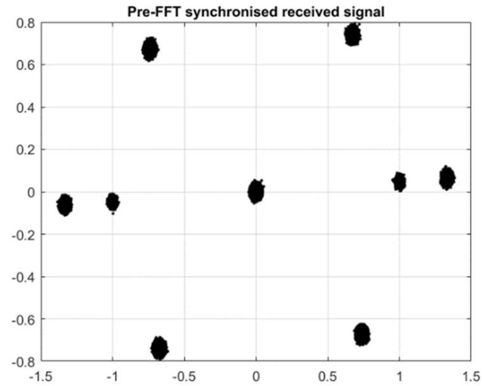


Figure 4.34 - Pre-FFT synchronizer output for an offset of 1 kHz

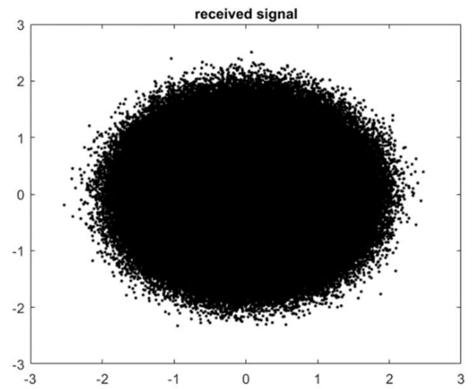


Figure 4.35 - Received Signal with an offset of 2.5 kHz

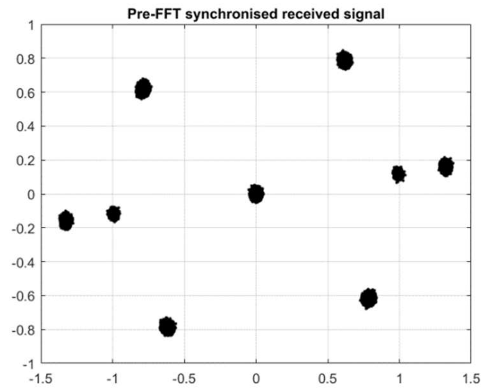


Figure 4.36 - Pre-FFT synchronizer output for an offset of 2.5 kHz

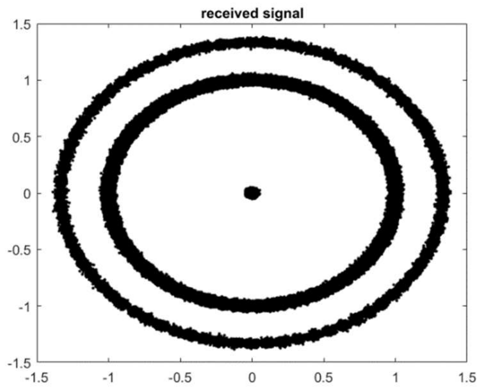


Figure 4.37 - Received Signal with an offset of 4.5 kHz

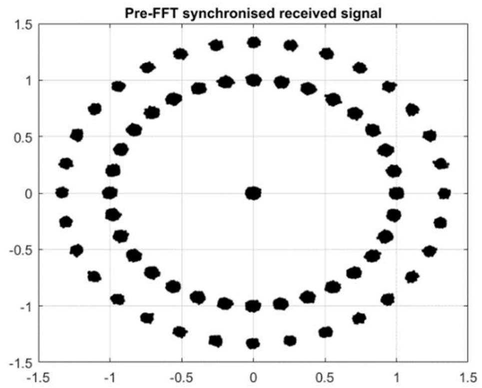


Figure 4.38 - Pre-FFT synchronizer output for an offset of 4.5 kHz

After the offset of 4.5 kHz this synchronizer can't properly adjust the signal because the angle of rotation of the constellation exceeds the 2π radians and the method isn't capable of estimating the angles above that. To compensate this error, the signal goes to the post-FFT synchronizer, which identifies the beginning and the end of the symbol, in the frequency domain, to adjust the positions of the carriers to the right position. The developed method consists in the detection of the unused carriers from the OFDM symbol. Since the symbol is already in the FFT windows, it is only necessary to adjust the carriers' positions to the right positions. Firstly, the signal is concatenated with a copy of itself. This way, the detection of the unused carrier is easier in case of the symbol being completely off position. The symbol is then detected by searching for two unused carriers spaced by 1705 or 6817 positions depending on the transmission mode, and although it may not change visually, the signal constellation will align the carriers to the right position allowing the next stage to compensate for the phase offset of the signal.

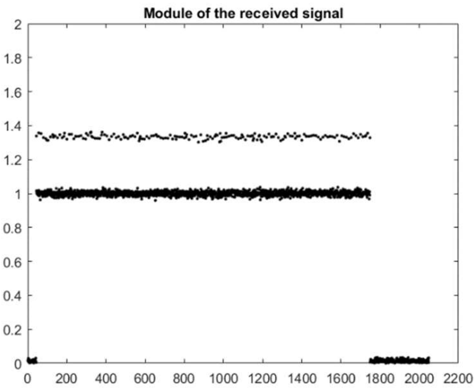


Figure 4.39 - Module of the OFDM symbol after an insufficient pre-FFT synchronization

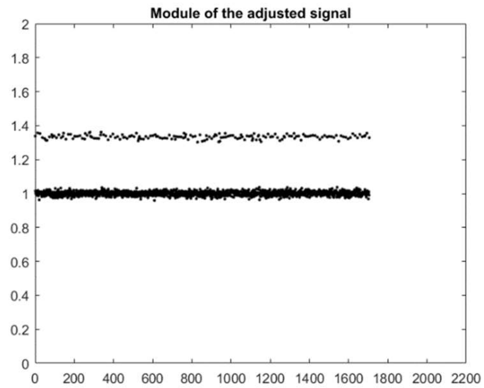


Figure 4.40 - Result of the post-FFT adjustment of the OFDM symbol

Since the carriers are now aligned, it is possible to compensate for any phase offset the signal has. This is made comparing the continual pilots. Since these pilots have a unique modulation for each carrier of each symbol, a comparison of the modulation of the carriers is made in the positions of the continual pilots with the ideal constellation. If the received signal has an offset compared to the ideal constellation, a rotation equal to the angle between both the signal and ideal constellation is applied to adjust the signal phase to a proper one. In the figures below, the results of the IQ compensator developed for the same frequency offset tested in the pre-FFT synchronizer are presented.

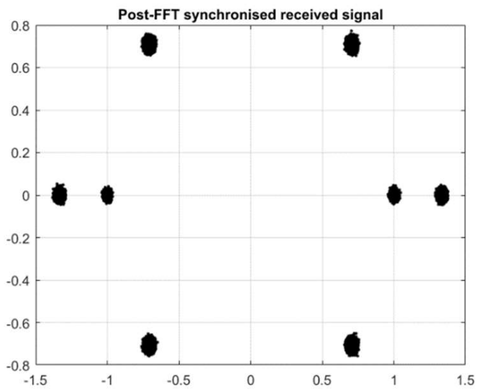


Figure 4.41 - IQ compensation for the 1 Hz offset

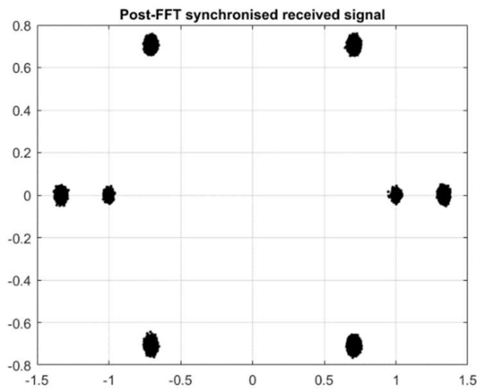


Figure 4.42 - IQ compensation for the 50 Hz offset

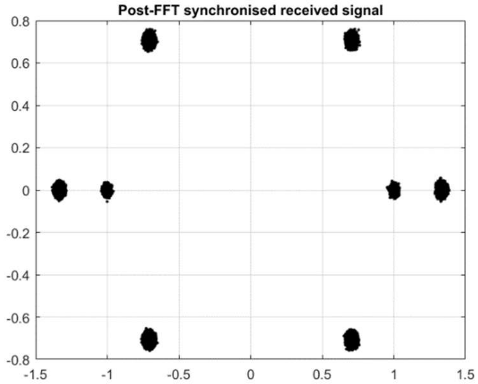


Figure 4.43 - IQ compensation for the 1 kHz offset

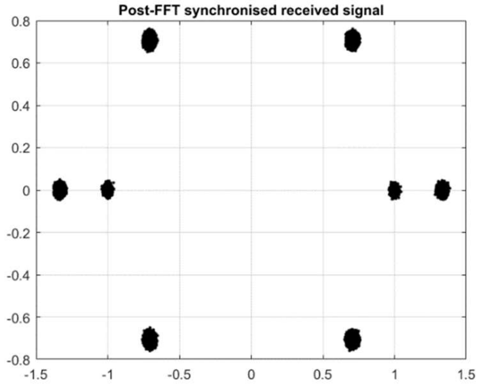


Figure 4.44 - IQ compensation for the 2 kHz offset

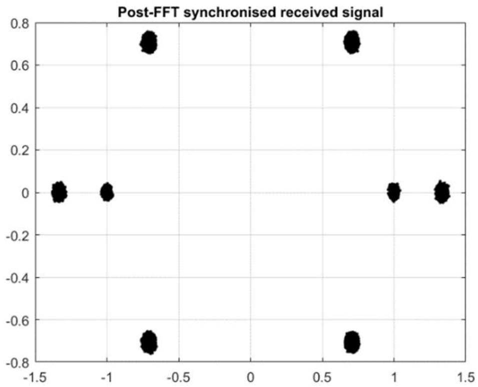


Figure 4.45 - IQ compensation for the 4 kHz offset

As demonstrated in the results of the IQ compensator, it can adjust the phase offset of the signal if the carriers are not properly aligned with the right position. With the demodulated and synchronized signal, it is now possible to verify the quality of the transmitted DVB-T. If there was any interference in the signal the constellation would not be readable or it would have unwanted modulations and, with that information, the LSA controller can act on a solution to rectify that situation.

Chapter 5

5 Conclusions and Future Work

The main objective of this dissertation was to develop a radio system prototype for the LSA Warner to implement the LSA model in Portugal. This warner should be able to detect the presence of any signal in the frequency band of 2.3 and 2.4 GHz and verify the condition of the incumbent's DVB-T signal and verify if it has the right conditions to be transmitted.

The proposed warner is divided in three modules, the Energy Detector, the DVB-T Characterizer and the Control Unit:

- The Energy Detector, this module works as expected since it can detect signals down to -80 dBm in the desired frequency band, all the stages developed for this module work and simulations done for each component correspond to the practical results obtain.
- In the DVB-T Characterizer module, the initial objective was to implement a DVB-T demodulator in a FPGA, however, this module was only simulated using MATLAB, in the simulation, a DVB-T transceiver was made, which works as expected and allows the user to choose the DVB-T parameters for the generated signal; a simple transmission channel able to induce different kind of errors to the signal was made; and lastly the receiver, which is able to synchronize signal, until a certain frequency offset error and, when inserted a time offset to the signal, the errors from the correlation signal do not allow a proper alignment of the signal. Besides this malfunction when added time offset, the receiver works as expected and allows the visualization of the constellation after the DVB-T signals passes thought a transmission channel;
- The Control Unit purposed gathers information from the Energy Detector. The DVB-T Characterizer and GPS module to send it to an online database

through GSM/GPRS connection. The developed control unit is able to read the information from the Energy Detector and GPS module and successfully send it to the database, allowing the visualization of the results from all the previous modules.

5.1 Future Work

As mentioned before, the next step for this project could be to improve the receiver module and implement this DVB-T Characterizer physically and do field tests to ensure the proper functionality to the module; also, an interface needs to be created to allow the Control Unit to read the information from the FPGA and sent it to the database.

Although the DVB-T detector can be very useful to check the incumbent's signal quality, another method can be used to verify if the frequency band is clear to the incumbents. This method consists in sensing the licensees' signals. Another version for the LSA Warner can be made by sensing LTE-TDD signals, which is the protocol adopted for the secondary user. In this case, it only has to check the presence of this signal and not the signal characteristic or interference.

References

[1] CEPT, "Licensed Shared Access," *ECC Rep. 205*, no. February, 2014.

[2] RSPG, "RSPG Opinion on Licensed Shared Access," *Eur. Comm. Radio Spectr. Policy Gr.*, 2013.

[3] ANACOM, "Quadro Nacional de Atribuição de Frequências," 2015.

[4] ETSI, "Digital Video Broadcasting (DVB); Framing Structure, Channel Coding and Modulation for Cable Systems," *En 300 429 V1.2.1*, 1998.

[5] G. D. Forney, "Burst-Correcting Codes for the Classic Bursty Channel," *IEEE Trans. Commun. Technol.*, vol. 19, no. 5, pp. 772–781, 1971.

[6] Keysight Technologies, "Cognitive Radio Algorithm Development and Testing," *White paper*, 2014.

[7] H. Urkowitz, "Energy Detection of Unknown Deterministic Signals," *Proceeding IEEE*, vol. 55, no. 4, pp. 523–531, 1967.

[8] M. Abdulsattar and Z. Hussein, "Energy Detection Technique for Spectrum Sensing in Cognitive Radio: A Survey," *Int. J. Comput. Networks Commun.*, vol. 4, no. 5, pp. 223–242, 2012.

[9] F. Salahdine, H. El Ghazi, N. Kaabouch, and W. F. Fihri, "Matched Filter Detection with Dynamic Threshold for Cognitive Radio Networks," *2015 Int. Conf. Wirel. Networks Mob. Commun.*, 2015.

[10] A. Sahai, N. Hoven, and R. Tandra, "Some Fundamental Limits on Cognitive Radio," *Allert. Conf. Control. Commun. Comput.*, pp. 1662–1671, 2004.

[11] P. D. Sutton, K. E. Nolan, and L. E. Doyle, "Cyclostationary Signatures in Practical Cognitive Radio Applications," *IEEE J. Sel. Areas Commun.*, vol. 26, no. 1, pp. 13–24, 2008.

[12] P. Šrámek, J. Svobodová, R. Maršálek, and A. Prokeš, "Using Cyclic Prefix Correlation for DVB-T Signals Detection," *2010 Conf. Proc. - ICECom 20th Int. Conf. Appl. Electromagn. Commun.*, 2010.

[13] V. Gaddam and M. Ghosh, "Robust Sensing of DVB-T Signals," *2010 IEEE Symp. New Front. Dyn. Spectrum, DySPAN 2010*, 2010.

[14] J. Fang, N. Zheng, H. Tan, X. P. Zhu, and B. Huang, "A Hybrid CP Detection Algorithm for DVB-T," *Proc. - 2010 3rd IEEE Int. Conf. Broadband Netw. Multimed. Technol. IC-BNMT2010*, pp. 639–643, 2010.

[15] C. El Hajjar, "Synchronization Algorithms for OFDM Systems (IEEE802. 11a, DVB-T)," 2007.

[16] M. Circuits, "Monolithic Amplifier - MNA-6A+."

[17] D. Pozar, *Microwave Engineering Fourth Edition.* 2005.

[18] Maxim, “MAX2015 - 0.1GHz to 3GHz, 75dB Logarithmic Detector/Controller.”

[19] V. KEL ELECTRONICS, “VK2828U7G5LF V1.0.”

Appendices

Appendix A

Article for URSI Portugal 2017, 11º Congress of URSI's Portuguese Committee

DVB-T signal sensing for the LSA model in Portugal

João Milheiro, Nuno Borges de Carvalho
Instituto de Telecomunicações, Departamento de Eletrónica, Telecomunicações e Informática
Universidade de Aveiro, Portugal

***Abstract*—With the necessity of more efficient use of the spectrum due to the demand from the consumers, the spectrum regulatory entities must find ways to make better use of the available spetrum and create tecniques that allow QoS and the same reliability as before. Taking this into account, CEPT came with the concept of the LSA model which makes use of three dimensions to perform the frequency sharing: time, frequency and area.**

The objective with this paper is to present a system that can incorporate the LSA Model in Portugal and its purpose is to locate, detect and characterize the incumbents' signal in order to verify if there is good conditions for the incumbents' signal to be transmitted.

Keywords*—*DVB-T, LSA, Share Spectrum, Signal Sensing,

I. Introduction

Nowadays the necessity of frequency bands by the consumers lead to a greater and better management of the spectrum, it is every National Regulatory Authoriy (NRA) responsibility to ensure the QoS and the proper frequency bands for each user. National Tables of Frequency Allocation (NTFA) are used worldwide to ensure there is no overlapping in the use of frequencies, they are allocated to only one entity and the NRA chooses what frequency to allocate and which will need certain authorizations to be used.

However, consumers are becoming more and more dependable on mobile broadband services and require low cost, high quality and faster technology for their services. Due to this demand, the amount of bandwidth used by these services is increasing rapidly and the actual method of spectrum management may not be enough to ensure a proper QoS for the consumers [1].

In response to this demand, the Radio Spectrum Policy Group (RSPG) came with the concept of the LSA model and in 2013 their opinion was approved allowing the spectrum regulatory organizations to start trials through Europe[2].

A. LSA Model

The License Shares Access (LSA) was difined by RSPG as "A regulatory approach aiming to facilitate the introduction of radiocommunication systems operated by limited number of licensees under an individual licensing regime in a frequency band already assigned or expected to be assigned to one or more incumbent users. Under the Licensed Shared Access (LSA) approach, the additional users are authorized to use the spectrum (or part of the spectrum) in accordance with sharing rules included in their rights of use of spectrum, thereby allowing all the authorized users, including incumbents, to provide a certain Quality of Service"[2].

The LSA is a complementary tool for spectrum management that makes use of three dimensions to perform the frequency sharing: time, frequency and area. Because of this the LSA will give to incumbent(s) and licensees a predictable QoS and easily protect the signal from harmful interference. The figure 1 shows a scheme of how LSA's shared framework can be attributed to the incumbent and allow the other licensees to access the same spectrum resource. This shared framework is a set of sharing rules and/or sharing conditions, and it is NRA responsibility to set these conditions, which can be set in consideration to future needs of an incumbent (marked in blue), as shown in figure 1. Since the green area will not be used by the incumbent, it can be allocated to a licensee. This way neither the incumbent nor the licensee will be affected from each other signals and both can utilize the same spectrum resource.

According to CEPT [1] the LSA model main functional blocks are composed by:

- **Incumbent**: The primary user of the spectrum resource;
- **Licensee**: The secondary user of the same spectrum resource;
- **LSA Repository**: A database which will gather all information about the spectrum usage and the geolocation of the incumbents. This repository is controlled by the NRA and is the central component of the LSA model;
- **LSA controller**: Makes all the calculation to allow or not, the licencees base stations to transmit in that frequency band. This calculations take into account all the information from the incumbent;

It was also aproved by CEPT that the frequency band to be used in the LSA models was the 2,3 - 2,4 GHz and it will be used for mobile/fixed communication networks (MFCN), including broadband wireless systens (BWS), while maintaining the current incumbent use[3].

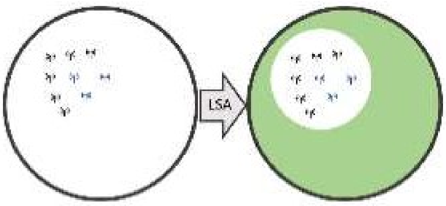


Fig. 1: Geographical influence of the LSA model

B. *LSA Model in Portugal*

The NRA in Portugal, ANACOM, is now starting to study the possibility to implement the LSA model in the country, and presented the solution shown in figure 2. Since in Portugal, the 2.3 and 2.4 GHz frequency band is being used to Ancillary to Programme-making (SAP) and Ancillary to Broadcasting (SAB) applications, the goal is to allocate a LSA warner to every camera from the national television broadcasters and anytime they are up to transmit their signal, the warner gathers spectrum information and its geolocation and send it to the LSA repository. This LSA warner must to be able to sense the presence of DVB-T and/or LTE-TDD signals, either to check if the signal from incumbents has a good MER, or if there is any LTE-TDD signal in that area that could interfere with the transmitted signal from the incumbents. Also the communication between the LSA repository and the LSA warner can be made by a 3G/4G connection.

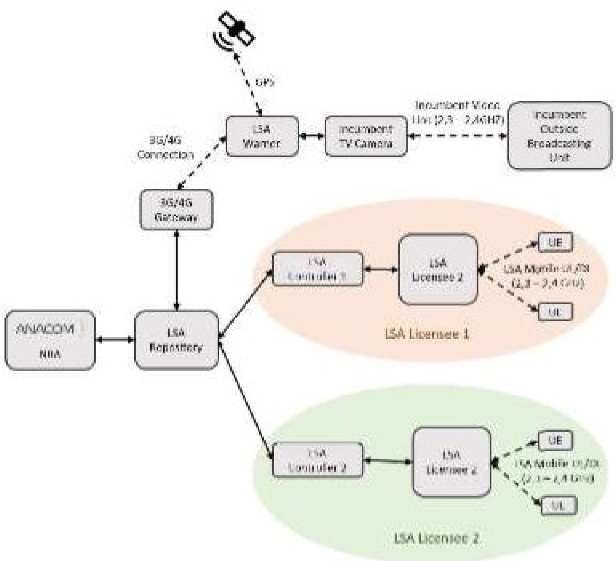


Fig. 2: LSA model concept

II. LSA Warner - DVB-T Signal Sensing

A. *Sensor Concept*

The system under development is composed by three distinct parts, the signal energy sensing circuits, the GPS & GSM/GPRS and finally the DVB-T Signal characterizer. The whole system as the purpose to work as a portable detector for signals emitted in the frequency band of 2.3 and 2.4 GHz and an analyzer for the incumbent's signal, which is DVB-T or a variant of it.

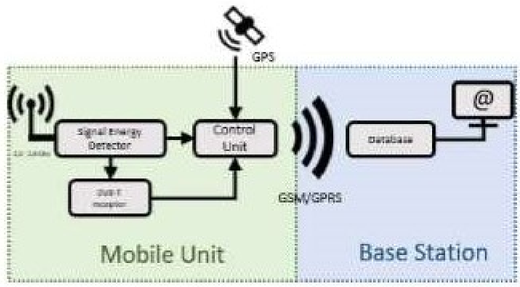


Fig. 3: Main functional blocks for the DVB-T sensor

The purpose of the signal energy detector is to sense any signal being transmitted in the frequency band of 2.3 and 2.4 GHz. It allows the conversion of the received signal power into a DC voltage corresponding and is able to detect signals with at least -80dBm. Even if this sensor doesn't give any information about the received signal besides intensity, it allows the system to detect if there is something being emitted in that band.

The DVB-T signal characterizer will be directly connected to the energy sensor, since it can be activated only when it detects any signal. This system is a DVB-T receptor implemented in a FPGA with the purpose to calculate MER level of the received signal and evaluate if the incumbents has the necessary conditions to transmit their signal.

Lastly the GPS&GSM/GPRS works as the control unit of the warner, it receives through a GPS module the position and time of the warner and gathers the information retrieved from the energy detector and the DVB-T signal characterizer and sends it all to an online database which is representing the LSA repository. This connection with the database is done with a GSM/GPRS module which uses 3G/4G connection to access the database server.

B. *Signal Energy Detector - Implementation and results*

This detector is based on a RF power detector integrated circuit, however the multistage logarithmic amplifier used is able to detect any signal up to 3GHz and it is only capable to detect signal with -50 dBm for the frequency band required, some components needed to be added the integrated circuit for it to be able to reach the desired requirements.

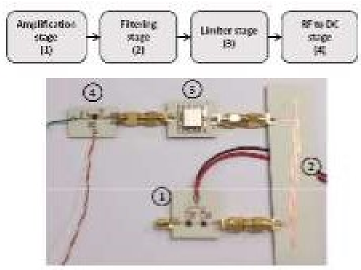


Fig. 4: Diagram and Circuit developed to the Signal Energy Detector

The full circuit diagram is present in figure 4, it is composed by a stage of amplification with a gain of approximately 40dB, this stage is necessary to be able to reach the -80 dBm signals. Followed by a filtering stage composed by a band-pass filter for the 2.3 and 2.4 GHz frequency band. The next stage is composed by a limiter, this works as a protection to the RF power detector since it is sensible to stronger signals. Lastly is the RF power detector itself, which will convert the received RF signal into a corresponding DC voltage.

The results of the whole circuit are shown bellow and it's possible to see that when inserted a -85 dBm signal the detector is already detecting it.

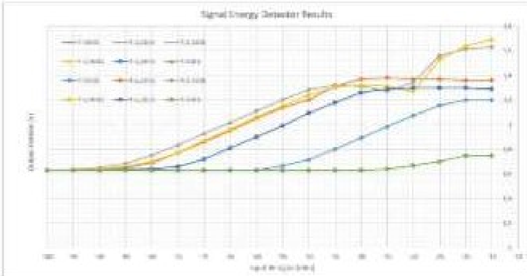
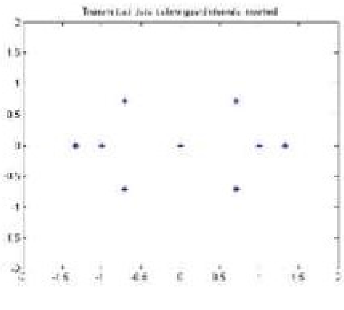
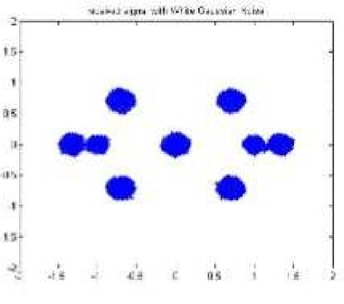


Fig. 5: Obtained results when injected RF signals into the circuit



(a)



(b)

Fig. 6: Transmitted (a) and Received (b) DVB-T Signal through a software simulation

C. DVB-T Signal Characterizer - Implementation and results

The goal with this module is to evaluate if the incumbents have the proper condition to transmit their signal, and for that it will be calculated the MER of the received signal and verify the quality of the incumbent's signal. This module is still under development and only a DVB-T transmitter and receptor has been simulated in software. An example can been seen in figure 6 where a DVB-T signal was transmitted using a QPSK modulation.

At this moment the synchronization methods for the DVB-T protocol to a proper reception of the signal are being studied and simulated such as correlation with the guard interval, as shown in figure 7 initial position of each symbol can be adjusted accordingly to the correlation signal.

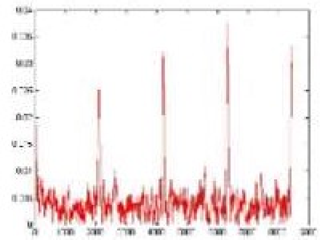


Fig. 7: Result for the guard interval correlation with the signal

D. GPS & GSM/GPRS & database - Implementation and results

The control unit is controlled by a microprocessor that reads the signal from the energy detector, the DVB-T demodulator and the time/position of the system from a GPS module. All this information is then treated to be sent to a database. For this unit we picked an Arduino microcontroller for its versatility and compatibility with the several components needed, such as the GPS module and the GSM/GPRS shield used to the 3G/4G connection to the database. In order to test the control unit a very simple database was created along with a website to display the information sent by the control unit. In Figure 8 is shown an example of what that website looks like and mainly, the information that arrived to the database.

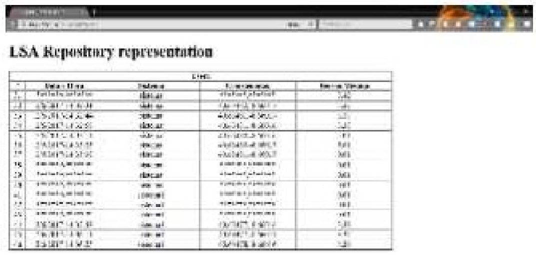


Fig. 8: Information sent by the control unit to the database

III. Conclusion & Future Work

In conclusion, with the work done so far only it is possible to the LSA warner to detect any kind of signal present in the frequency band of 2.3 and 2.4 GHz and also to address that information to an online database, alongside with its location. The DVB-T characterizer even in a early stage will be a great upgrade to the current sensor. After finishing the simulations tests to the DVB-T receptor, it will be translated to FPGA language an implemented in the Picozed SDR and start the field tests.

References

[1] CEPT. Licensed shared access. *ECC Report 205*, (February), 2014.

[2] RSPG. RSPG Opinion on Licensed Shared Access RSPG Opinion on Licensed Shared Access. *European Commission, Radio Spectrum Policy Group*, 2013.

[3] CEPT. Harmonised technical and regulatory conditions for the use of the band 2300-2400 MHz for Mobile/Fixed Communications Networks (MFCN), 2014.

Appendix B

Code used for the interface of the control Unit and for the communication with the database

```
#include <AltSoftSerial.h>
#include <TinyGPS++.h>
#include <GSM.h>

// PIN Number
#define PINNUMBER "0000"
// APN data
#define GPRS_APN       "net2.vodafone.pt" // replace your GPRS APN
#define GPRS_LOGIN     "vodafone"    // replace with your GPRS login
#define GPRS_PASSWORD  "vodafone" // replace with your GPRS password

#define sistemaID "sistema1"

//*****************************VARIAVEIS*********************************/

//******GPS*******//
/* EN (amarelo)   -> 11
   GND (preto)    -> gnd
   RxPin (azul)   -> 48
   TxPin (verde)  -> 46
   VCC (vermelho) -> 5V
   PPS (branco)   -> ... */

static const int ENPin = 7;
static const uint32_t GPSBaud = 9600;

static const int pwrSensorPin = A5;


int numSat, error;
float latDeg, lngDeg, altMeters;
TinyGPSPlus gps;
TinyGPSDate data;
TinyGPSTime horas;
AltSoftSerial altSerial;

//****GPRS*****//
GSMClient client;
GPRS gprs;
GSM gsmAccess;

// server, path & port
//char server[] = "dbtester.000webhostapp.com";
char server[] = "193.136.92.81";
char path[] = "/anacomProject/addData2DB.php";
int port = 80; // port 80 is the default for HTTP

//************//

String temp_data;
String temp_hora;
String temp_dataHora;
String temp_coords_lat;
String temp_coords_long;
String temp_coords;
float temp_pwrMeasurded;

int sensorPin = A0;

//*************************Configurações*****************************//
void setup() {

  //Serial configs//
  Serial.begin(9600);
  while (!Serial);

  //;

  pinMode(ENPin, OUTPUT);
  digitalWrite(ENPin, HIGH);

  Serial.println("Starting Arduino web client.");
  // connection state
  boolean notConnected = true;

  // After starting the modem with GSM.begin()
  // attach the shield to the GPRS network with the APN, login and password
  while (notConnected) {
    if ((gsmAccess.begin(PINNUMBER) == GSM_READY) &
        (gprs.attachGPRS(GPRS_APN, GPRS_LOGIN, GPRS_PASSWORD) == GPRS_READY)) {
      notConnected = false;
      Serial.println("attached!");
    } else {
      Serial.println("Not connected");
      delay(1000);
    }
  }

  Serial.println("connecting...");
  Serial.println();

}

void loop() {

  getVarsFromPwrMetter();
  altSerial.begin(GPSBaud);
  getVarsFromGPS();
  altSerial.end();
  String dataHora = "dataHora=" + temp_dataHora;
  String coords = "coordenadas=" + temp_coords;
  String pwrMeasurded = "potenciaMedida_db=" + String(temp_pwrMeasurded);
  String sistemas = String("sistema=" + String(sistemaID));

  if (client.connect(server, port)) {
    client.print("GET ");
    client.print(path);
    client.print("?");
    client.print(dataHora);
    client.print("&");
    client.print(coords);
    client.print("&");
    client.print(pwrMeasurded);
    client.print("&");
    client.print(sistemas);
    client.print("&");
    client.println(" HTTP/1.1");
    client.print("Host: ");
    client.println(server);
    client.println("Connection: close");
    client.println();
    client.stop();
    Serial.println(F("sucess"));
    Serial.println();
  }
  else Serial.println(F("connection failed"));

  Serial.println();
  delay(9000);
}
void getVarsFromPwrMetter() {

  int sensorValue = 0;  // variable to store the value coming from the sensor
  int sensorValuemed = 0;

  for (int i = 0; i < 50; i++) {
    sensorValue = analogRead(sensorPin);
    sensorValuemed = (sensorValuemed + sensorValue) / 2;
  }

  temp_pwrMeasurded = sensorValuemed * (5.0 / 1023.0) + 0.01;
  Serial.print("PwrMeasurded: ");
```

```
  Serial.println(temp_pwrMeasurded);
}
void getVarsFromGPS() {
  Serial.print("Num de Satelites: ");
  Serial.println(getNumSatelite(gps));
  smartDelay(0);

  temp_coords_lat = getLatitude(gps);
  //  Serial.print("Latiude: ");
  //  Serial.println(temp_coords_lat);
  smartDelay(0);

  temp_coords_long = getLongitude(gps);
  //  Serial.print("Longitude: ");
  //  Serial.println(temp_coords_long);
  smartDelay(0);

  temp_coords = temp_coords_lat + "," + temp_coords_long;
  Serial.print("Coordenadas ");
  Serial.println(temp_coords);

  temp_data = getDate(gps);
  //  Serial.print("data: ");
  //  Serial.println(temp_data);
  smartDelay(0);

  temp_hora = getTime(gps);
  //  Serial.print("tempo: ");
  //  Serial.println(temp_hora);

  temp_dataHora = temp_data + "-" + temp_hora;
  Serial.print("Data/Horas: ");
  Serial.println(temp_dataHora);

  smartDelay(1000);

  if (millis() > 5000 && gps.charsProcessed() < 10)
    Serial.println(F("No GPS data received: check wiring"));

}
//***************************************************************//
#define invalidGPS 0
#define str_invalidGPS "*******"
#define precision 5

String getNumSatelite(TinyGPSPlus gps){
  if (gps.satellites.isValid()){
    return String(gps.satellites.value());
  }
  else{
    return str_invalidGPS;
  }
}

String getLatitude(TinyGPSPlus gps){
  if (gps.location.isValid()){
    return String(gps.location.lat(),precision);
  }
  else{
    return str_invalidGPS;
  }
}

String getLongitude(TinyGPSPlus gps){
  if (gps.location.isValid()){
    return String(gps.location.lng(),precision);
  }
  else{
    return str_invalidGPS;
  }
}

String getDate(TinyGPSPlus gps){
  if (gps.date.isValid()){
    return String(gps.date.day()) + "/" + String(gps.date.month()) + "/" +
    String(gps.date.year());
  }
  else{
    return str_invalidGPS;
  }
}

String getTime(TinyGPSPlus gps){
  String Time;
  if (gps.time.isValid()){
    return String(gps.time.hour()) + ":" + String(gps.time.minute()) + ":" +
    String(gps.time.second());
  }
  else{
    return str_invalidGPS;
  }
}

static void smartDelay(unsigned long ms){
  unsigned long start = millis();
  do
  {
    while (altSerial.available())
      gps.encode(altSerial.read());
  } while (millis() - start < ms);
}
```

Appendix C

Code used for the MySQL database, HTML and PHP for the website

File: index.php

```html
<!DOCTYPE html>

<!--<script src="javaScript.js"></script>-->
<script src="https://code.jquery.com/jquery-1.10.2.js"></script>
<script> var auto_refresh = setInterval((function ()
{$("#dynamicTable").load("tables.php"); }), 1000); </script>


<html>
   <head>
      <title>Signal Detector Tester</title>
   </head>
<body>

   <h1>LSA Repository representation</h1>

    <div id="dynamicTable" style ="width:auto; height:1000px">
      <?php include("tables.php") ?>
    </div>



</body>
</html>
```

File: tables.php

```php
        <table border='1' cellspacing='0' cellpadding='0'>

        <tr>
            <th colspan='5'>User1</th>
        </tr>
        <tr>
            <td align='center' width='auto'><b> # </b></td>
            <td align='center' width='160px'><b> Data e Hora </b></td>
            <td align='center' width='200px'><b> Sistema </b></td>
            <td align='center' width='200px'><b> Coordenadas </b></td>
            <td align='center' width='200px'><b> Tensão
            Medida </b></td>
        </tr>

        <?php
            include("dBconnection.php");

            $link=Connection();

            $result= $link->query("SELECT * FROM `system_1` ");

            if($result!==FALSE){
                while($row = $result->fetch_array(MYSQLI_ASSOC)) {
                    printf("

                                <tr>
                                <td align='center'>   %s  </td>
                                <td align='center'>   %s  </td>
                                <td align='center'>   %s   </td>
                                <td align='center'>   %s  </td>
                                <td align='center'>   %s  
                                </td></tr>",
                            $row["id"],$row["dataHora"],
                            $row["sistema"],$row["coordenadas"],
                            $row["potenciaMedida_db"]);

                }
                mysqli_free_result($result);
                mysqli_close($link);
            }
        ?>

        </table>
```

File: dBconnection.php

```
<?php
    include("dBconnection.php");

    $link=Connection();

        $VAR_dataHora=$_GET["dataHora"];
        $VAR_coordenadas=$_GET["coordenadas"];
        $VAR_potenciaMedida_db=$_GET["potenciaMedida_db"];
        $VAR_sistema=$_GET["sistema"];

        $query = "INSERT INTO system_1 (dataHora , sistema, coordenadas,
        potenciaMedida_db)
        VALUES
        ('$VAR_dataHora','$VAR_sistema','$VAR_coordenadas','$VAR_potenciaMedida_
        db')";

        $link -> query($query);


    mysqli_close($link);
    header("Location: index.php");
?>
```

File: addData2DB.php

```
<?php
 function Connection(){
  $server="193.136.92.81";
  $user="tester";
  $pass="tester";
  $db="arduinodb_anacomproject";

  $connection = new mysqli($server, $user, $pass, $db);

  if ($connection -> connect_error) {
      die('MySQL ERROR: ' . $mysqli->connect_errno .')'.$mysqli->connect_error);
  }

  return $connection;
 }
?>
```