Architecture And Algorithms For The Implementation Of Digital Wireless Receivers In FPGA And ASIC: ISDB-T And DVB-S2 Cases

UNIVERSITAT
POLITÈCNICA
DE VALÈNCIA

Eduardo Rodrigues de Lima
Departamento de Comunicaciones

Thesis Directors

Vicenç Almenar Terré
Santiago Flores Asenjo

January 2016

Ithaka[1]

As you set out for Ithaka
hope the voyage is a long one,
full of adventure, full of discovery.
Laistrygonians and Cyclops,
angry Poseidon - don't be afraid of them:
you'll never find things like that on your way
as long as you keep your thoughts raised high,
as long as a rare excitement
stirs your spirit and your body.
Laistrygonians and Cyclops,
wild Poseidon - you won't encounter them
unless you bring them along inside your soul,
unless your soul sets them up in front of you.

Hope the voyage is a long one.
May there be many a summer morning when,
with what pleasure, what joy,
you come into harbors seen for the first time;
may you stop at Phoenician trading stations
to buy fine things,
mother of pearl and coral, amber and ebony,
sensual perfume of every kind -
as many sensual perfumes as you can;
and may you visit many Egyptian cities
to gather stores of knowledge from their scholars.

Keep Ithaka always in your mind.
Arriving there is what you are destined for.
But do not hurry the journey at all.
Better if it lasts for years,
so you are old by the time you reach the island,
wealthy with all you have gained on the way,
not expecting Ithaka to make you rich.

Ithaka gave you the marvelous journey.
Without her you would not have set out.
She has nothing left to give you now.

And if you find her poor, Ithaka won't have fooled you.
Wise as you will have become, so full of experience,
you will have understood by then what these Ithakas mean.

Constantine P. Cavafy(1863-1933)

[1]Translated by Edmund Keeley/Philip Sherrard

What is now proved was once only imagined.
William Blake(1757-1827)

in the memory of my father, Fernando
to my mother, Eliete
to my wife, Celia
to my baby, Henrique

Acknowledgements

This has been a very long journey, full of discoveries. At this point, besides I have figured out new techniques, written several papers as well as my thesis, and the *viva voce* defense is approaching, I realized that I have had the opportunity to figure out new cultures, new countries (even nations within countries) and new languages. For sure, the pressure of pursuing a PhD in such a scenario has significantly changed the way I see the things and the World, which became smaller. Those who know me personally can see that. Nevertheless, none of that would make sense without the people I have found along the way. Now, they are part of my personal history. Indeed I have made a lot of good friends and colleagues that at this moment are spread all over the world, settling them own ways to *Ithaka*. During this journey I've meet a lot of characters that inspired me and contributed in some way to help me going further. Some of them helped me in technical aspects, other helped me to support the long time far away from my family and Brazilian friends, during part of this journey that was performed in Europe. But all of them, even without realizing that, helped me to grow as a human been. Having said that, it is time to thank. So, the minimum I must do is to express my deepest gratitude to everyone which was involved in this process.

First of all I would like to say thanks to my thesis directors Prof. Vicenç Almenar Terré for all support, opportunities and patience during all these years, as well as for the excellent reviews on my thesis, and the elegant Professor Santiago Flores Asenjo, especially for opening the UPV doors to me and to introduce me to COST273 Cooperation Group which gave access and closeness to inspiring people like Prof. Luis Correa, Prof. Alister Burr, Prof. Mischa Dohler, Prof. Ernst Bonek, Dr. Reinaldo Valenzuela, Prof. Alan Sibile, Prof. Roberto Verdone, Prof. Luc Vandendorpe, Prof. Emmanuel Van Lil, Prof. Andreas Molish, Dr. Robert Bultitude, Dr. Tricia Willink, and many other bright minds in the telecommunications community.

Despite Professor Javier Valls Coquillat from UPV was not my formal adviser, I consider him as so. As such, I owe him a special thanks for introducing me to the FPGA world and to the wireless communications implementations aspects in FPGA. Thanks for the long conversation during the SPL2010, in São Carlos , Brazil. That

talk gave me the necessary strength and courage to go ahead. I will also always remember the Fideuá you cooked for our research team, at Playa de Xeraco. Definitely, you are a great guy.

I would like to thank the members of my thesis reviewing committee, for have found time in them busy agenda to evaluate my work in a timely manner: Prof. Vicente Baena Lecuyer from the Electronics Engineering Department, of the Universidad de Sevilla - my thesis is strongly related to his research field - and Prof. Rodolfo Jardim Azevedo from the Computing Institute of the Universidade Estadual de Campinas, which introduced me to the Brazil-IP and CI-Brasil Program, so in short, without have met him, this thesis would be quite different.

On my thesis evaluation committee, I first thank the busy Prof. Alberto Gonzalez Salvador for being so kind and accept to be the president of the Jury. He is an example to follow and still keeps inspiring me. When I do some research I always remember the first time I went to his office and I asked his opinion about a theoretical idea on Joint Probability and Mutual Information to be used in Frequency Diversity in OFDM, and he asked me: "does it have any practical application?". That question says a lot about him. Next, Prof. Antonio Garcia Rios from the Electronics and Computers Technology Department at Universidad de Granada, which gently accepted to be the Secretary of my thesis Jury, and who certainly will significantly contribute for improving my work due to his large experience on the implementation of Digital Signal Processing in FPGA and VLSI. For last but not least, I give a special thanks to Prof. Hercules Pereira Neves, from Uppsala Universitet and Unitec Semiconductors. As one of the Brazilians who gave his contributions for past and current attempts, to make Brazil a player in the international semiconductor industry, he has advised me and gave me an unique perspective, as someone who is working in this area for a long time. In this way, he helped me to figure out how I could contribute for the current Brazilian attempt. Thanks also for being part of my reviewing committee.

Prof. Maria José Canet and Dr. Fabian Angarita you are great guys! Thank you for always being there during my staying in Gandia, to give me support, for the friendship, and an especial thanks for the remote support when I came back Brazil. I also thanks José Miguel and Kleber Iguchi for the friendship and for sharing good times in València. Other members of UPV that I'm in debt are Prof. Narcis Cardona, Prof. Trini Sansaloni, Prof. Asunción Perez-Pascual, Prof. Vicente Torres, Prof. Amparo Girona, Prof. José Marin-Roig, Prof. Gema Piñero, Prof. Juan Reig Pascal, Prof. Juan L. Corral, Prof. José F. Alarcón, Prof. Raul Bosch, Prof. Francisco Castells, Prof. José Javier López, Prof. Luis Vergara, Prof. José Capmany, Prof. Vicente Casares, Prof. Luis Guijarro and Prof. Marta Cabedo. I'm also in debt with Prof. Eduardo Boemo and Prof. Gustavo Sutter from Universidad Autônoma de Madrid, as well as Prof. Elias Todorovich from UNICEN, Argentina.

While pursuing my doctoral degree, I had the pleasure to be accepted as international scholar at Katholieke Universiteit Leuven and performed research at IMEC,

in Leuven-Belgium. There I rediscovered the passionate Microelectronics world, that was introduced me by Prof. Nobuo Oki during the Electrical Engineering undergraduate course at UNESP in Ilha Solteira, and I was introduced to Nanoelectronics. During my research period at IMEC, I have also confirmed that top level research can live in peace with practical applications. There, I have met several smart and kind persons. I know that part of you are no longer at IMEC but you will always be my IMEC connection. Thank you all to propitiate such a good time in Belgium: Dr. Liesbet V. der Peere, Dr. Frederik Petre, Dr. David Novo, Carolina Blanch, Cinzia De Marco, Dr. Francesca Barbagini, Dr. Vitto Giannini, Dr. Yann Joncour, Adrian Sarbu ,Steve Stoffels, Daniel Aguado, Salvador Eslava, Enrico Tenaglia, Olalla Varela, Suevia Rodriguez, Concepción Sanz, Dr. Francesca Clemente, Lühn Ole, Thomas Schuster, Hans Cappelle, Roeland Vandebriel, Dr. Mustafa Badaroglu, Dr. Jimmy Nsenga, Dr. Claude Desset, and Eduardo Estraviz. A special thanks to Prof. David Gesbert, from EURECOM, that introduced me to Dr. Frederick Petré that opened the DESICS/IMEC doors for me.

A special thanks to the other two members of my Brazilian connection that I had the pleasure to meet at IMEC: Dr. Augusto Redolfi and Dr. Guilherme Lujan . Together, we laugh a lot during the lunch time, chatting (a lot) often about an infinity of subjects and of course about micro and nanoelectronics and the future of microelectronics in Brazil.

I would like to thanks Prof. Guido Araújo from Computing Institute at UNICAMP and Prof. Edna Barros from LINCS/CETENE/UFPE for introducing me to the Brazil-IP and CI-Brasil Program. I salute you for the effort you have done for the success of microelectronics in Brazil. Thanks to my former colleagues at UNICAMP: Prof. Romis Attux, Tarciano Pegoraro, Prof. Ricardo Suyama. You inspired me to start my doctorate. I also thanks Prof. João Marcos Romano from DECOM/UNICAMP, because it was during his classes that I started thinking seriously on pursuing a PhD.

Many, many thanks to my former colleagues at Idea! Electronic Systems, that worked hard in the DTV01 project, and strongly contributed and some of them still contribute for the attempt to bring Brazil to the international scenario of the semiconductor industry, as a reliable silicon design provider. Together we had good times, we did great things and even given the limited resources, we did our best. You guys must be proud of it: Valdiney Pimenta, Gabriel Teles, Tiago Barros, Pedro Paro, Tomaz Vilela, Mauricio Suguiy, Jefferson Capovilla, Marcelo Guedes, Roberto Barbosa, Roberto Borgognoni, Aleksey Covacevice, Rafael Madeira, Tiago Lyrio, Agknaton Bottenberg, Alexandre Santiago, Davi Castro, Thiago Crespo, Leandro Gontijo, Flavio Silva, Denis Fantinato, Felipe Torrezan, Daniela Corsi and Frederico Gonalves. Thank you also, Samuel Goto and Felipe Renon.

A special thanks to the enthusiastic, energetic and busy José Eduardo Bertuzzo from Instituto de Pesquisas Eldorado for all his support during the ISDB-T and

DVB-S2 receivers projects, and for the personal and professional engagement in the microelectronics projects. Thank Rodrigo Sales to always, when possible, answer my requests for equipments and budget for travels and conferences, as well as for sharing some Friday evening with me, Renato Nascimento and Haroldo Onisto, in get togethers (of course before I get started writing my thesis).

Eu quero agadecer especialmente à minha me Eliete, aos meus irmãos José Carlos, Waldisia, Walmir e Waldir, s minhas cunhadas, aos meus sobrinhos Luciana, Felipe e Fernando, pelo apoio e torcida durante todos estes anos. No decorrer desta tese novos sobrinhos nasceram e fizeram parte desta jornada mais alegre e agitada: minha afilhada Gabrieli, Ana Clara, Alexsander, Daniel, Lucas e Ana Julia.

A very special thank to my team at Instituto de Pesquisas Eldorado that worked on the DVB-S2 project: Augusto Queiroz, Gabriel Silva, Denise Alves, Cesar Chaves, André Sapper, João Tardelli, as well as the others members of my team: Maique Correa, Daniel Urdaneta, Tiago Diadami, Vanessa Brischi, Jssica Oliveira and José Arnaldo. I also thank Prof. Marcelo Fernandes from UFRN for the support during the DVB-S2 project, specially on Adaptive Equalization issues. I also thank to the former DVB-S2 mattes Felipe Erazo, Gerson Brito and Marcos Matias.

My sincere thank to my colleagues at Eldorado that took part in the ISDB-T receiver project (DTV01): Fernando Idalirio, Ricardo Yoshioka, Marcelo Jara (former Idea! and Eldorado), Ciro Carvalho, Hamilton Costa, Jeroen Veremereen, Victor Boni, Fernando Martines, Jackeline Mertes, Tiago Merlo, Pedro Mietto, Murillo Fraguas, Erasmo Chiappetta, Sergio Rueda, Jairo Arguello, Tiago Rubia, Flávia Costa and the former Eldorado colleagues: Rodolfo Quarto, Honório Gomes and Fábio Ribas. My sincere thanks also goes to the Perceptia Devices Inc. colleagues: Andy Growstra and Tim Robins.

Thank you Eduardo Catto to be so nice and accepted to review the English language of part of my thesis, on such a short notice.

I also would like to thank the entities that granted me scholarships during my PhD studies or had funded the projects presented in this thesis in certain way: Instituto de Pesquisas Eldorado, Idea! Electronics Systems, Brazilian Ministry of Science, Technology and Innovation (MCTI), CNPq, CI-Brasil Program, Brasil-IP, FINEP/SIBRATEC and Generalitat Valenciana.

Last but certainly not least, my most heartfelt thank to my loved wife Célia - she knows my best side and tolerate my worst, we share plans, dreams and nightmares - and to the sunshine of our lives, our little Henrique, who brought joy into our lives.

Eduardo Rodrigues de Lima

Campinas, December 2015.

Abstract

The first generation of Terrestrial Digital Television(DTV) has been in service for over a decade. In 2013, several countries have already completed the transition from Analog to Digital TV Broadcasting, most of which in Europe. In South America, after several studies and trials, Brazil adopted the Japanese standard with some innovations. It was the first country to commercially broadcast Digital Television in the region, in 2007. Nevertheless, in 2013 the country still had not completed the transition to Digital.

Japan and Brazil started Digital Terrestrial Television Broadcasting (DTTB) services in December 2003 and December 2007 respectively, using Integrated Services Digital Broadcasting - Terrestrial (ISDB-T), also known as ARIB STD-B31 [1]. ISDB-T was clearly inspired on DVB-T [2], the first generation of European Terrestrial Digital Television Broadcast standard. Nevertheless, it overcomes one of the weaknesses of DVB-T, i.e. the bad performance under impulsive noise. DVB-T and ISDB-T make use of Orthogonal Frequency Division Multiplex (OFDM) as a Transmission Technique. Due to its unique characteristics, OFDM has been chosen by several wireless systems designers and standardization bodies. In order to provide an idea about the importance of the OFDM transmission technique, we can mention some examples of standards that employ it: IEEE802.15.4.g, IEEE802.11a/g/n/ac, Wimax, DVB-T, IEEE 802.20 and the 3GPP Long Term Evolution (LTE) mobile broadband standard downlink.

Before adopting ISDB-T, the Brazilian government made an enormous effort, putting together Universities, R&D Centers, the local Broadcast Industry and Broadcasters in order to establish a Brazilian digital television standard that would explore state-of-art techniques, e.g., Low Density Parity Check (LDPC) code and Space-Time Coding. Nevertheless, after all that effort, due to scale economy and its unique characteristics at that time, the Brazilian government adopted the ISDB-T Physical Layer, with some updates on Video and Audio Coding and Interactivity, as the basis for the Brazilian System of Terrestrial Digital Television (SBTVD). So, the ISDB-T with the aforementioned updates became known as ISDB-Tb. On the Digital Broadcasting Expert Group (DIBEG) website (www.dibeg.org), the reader can find the equivalence

between the Brazilian and Japanese Standards for Terrestrial Digital Television.

In June 2005 the Committee for the Information Technology Area (CATI) of Brazilian Ministry of Science and Technology and Innovation MCTI approved the incorporation of the IC-Brazil Program, in the National Program for Microelectronics (PNM) which is a Priority Program within the Brazilian Policies for Informatics and Automation. The main goals of IC-Brazil are the formal qualification of IC designers, support to the creation of semiconductors companies focused on projects of ICs within Brazil, and the attraction of semiconductors companies focused on the design and development of ICs in Brazil.

The work presented in this thesis originated from the unique momentum created by the combination of the birth of Digital Television in Brazil and the creation of the IC-Brazil Program by the Brazilian government. Without this combination it would not have been possible to make these kind of projects in Brazil. These projects have been a long and costly journey, albeit scientifically and technologically worthy, towards a Brazilian DTV state-of-the-art low complexity Integrated Circuit, with good economy scale perspectives due to the fact that at the beginning of this project ISDB-T standard was not adopted by several countries like DVB-T.

During the development of the ISDB-T receiver proposed in this thesis, it was realized that due to the continental dimensions of Brazil, the DTTB would not be enough to cover the entire country with open DTV signal, specially for the case of remote localizations far from the high urban density regions. A natural choice to overcome this issue is to use satellite broadcasting system, as medium to distribute open DTV signal straight to the households, without the need of having a local DTTB system. At that time, Eldorado Research Institute and Idea! Electronic Systems, foresaw that, in a near future, there would be an open distribution system for high definition DTV over satellite, in Brazil. Based on that, it was decided by Eldorado Research Institute, that would be necessary to create a new ASIC for broadcast satellite reception. At that time DVB-S2 standard was the strongest candidate for that, and this assumption still stands nowadays. Therefore, it was decided to apply to a new round of resources funding from the MCTI - that was granted - in order to start the new project.

This thesis discusses in details the Architecture and Algorithms proposed for the implementation of a low complexity Intermediate Frequency(IF) ISDB-T Receiver on Application Specific Integrated Circuit (ASIC) CMOS. The Architecture proposed here is highly based on the COordinate Rotation Digital Computer (CORDIC) Algorithm, that is a simple and efficient algorithm suitable for VLSI implementations. The receiver copes with the impairments inherent to wireless channels transmission and the receiver crystals. The thesis also discusses the Methodology adopted and presents the implementation results. The receiver performance is presented and compared to those obtained by means of simulations.

Furthermore, the thesis also presents the Architecture and Algorithms for a DVB-

S2 receiver targeting its ASIC implementation. However, unlike the ISDB-T receiver, only preliminary ASIC implementation results are introduced. This was mainly done in order to have an early estimation of die area to prove that the project in ASIC is economically viable, as well as to verify possible bugs in early stage. As in the case of ISDB-T receiver, this receiver is highly based on CORDIC algorithm and it was prototyped in FPGA. The Methodology used for the second receiver is derived from that used for the ISDB-T receiver, with minor additions given the project characteristics.

Resumen

La primera generación de Televisión Digital Terrestre(DTV) ha estado en servicio por más de una década. En 2013, varios países completaron la transición de transmisión analógica a televisión digital, la mayoría de ellas en Europa. En América del Sur, después de varios estudios y ensayos, Brasil adoptó el estándar japonés con algunas innovaciones. Fue el primer país en difundir comercialmente televisión digital en la región, en 2007. Sin embargo, en 2015 el país todavía no haba completado la transición a digital.

Japón y Brasil comenzaron a prestar el servicio de Difusión Televisión Digital Terrestre (DTTB) en diciembre de 2003 y diciembre de 2007 respectivamente, utilizando Radiodifusión Digital de Servicios Integrados (ISDB-T), también conocida como ARIB STD-B31 [1]. ISDB-T es claramente inspirado en DVB-T [2], la primera generación de la norma europea de Televisión Digital Terrestre. Sin embargo, ISDB-T supera una de las debilidades de DVB-T, el mal desempeño bajo ruido impulsivo. DVB-T e ISDB-T hacen uso de la técnica de transmisión Acceso múltiple por división de frecuencias ortogonales (OFDM). Debido a sus características únicas, OFDM ha sido elegido por varios diseñistas de sistemas inalámbricos y los organismos de normalización. Con el fin de dar una idea de la importancia de la técnica de transmisión OFDM, podemos mencionar algunos ejemplos de normas que la emplean: IEEE802.15.4.g, IEEE802.11a/g/n/ac, Wimax, DVB-T, IEEE 802.20 y el estándar de banda ancha móvil 3GPP Long Term Evolution (LTE).

Antes de adoptar el estándar ISDB-T, el gobierno brasileño hizo un enorme esfuerzo, juntando Universidades, Centros de I&D, la Industria de difusión y las emisoras locales con el propósito de establecer un estándar de televisión digital brasileña que explorarse técnicas de punta, por ejemplo, códigos para comprobación de paridad de baja densidad (LDPC) y de Espacio-Tiempo. Sin embargo, después de todo ese esfuerzo, debido a la economía de escala y a sus características únicas en ese momento, el gobierno brasileño adoptó la capa física del ISDB-T, con algunas actualizaciones en la codificación de vídeo y audio e interactividad, como base para el Sistema Brasileño de Televisión Digital Terrestre (SBTVD). Así, el ISDB-T con las actualizaciones antes mencionadas, se hizo conocido como ISDB-Tb. En el sitio

web del Grupo de Expertos de Radiodifusión Digital (DiBEG) (www.dibeg.org), el lector puede encontrar la equivalencia entre las normas brasileñas y japonesas para la Televisión Digital Terrestre.

En junio de 2005, el Comité del Área de Tecnología de la Información (CATI) del Ministerio de Ciencia, Tecnología e Innovación de Brasil - MCTI aprobó la incorporación del Programa CI-Brasil, en el Programa Nacional de Microelectrnica (PNM), el cual es un programa prioritario dentro de las Políticas brasileñas de Informática y Automatización. Los principales objetivos de la CI-Brasil son la formación de diseńistas de CIs, apoyar la creación de empresas de semiconductores enfocadas en proyectos de circuitos integrados dentro de Brasil, y la atracción de empresas de semiconductores interesadas en el diseño y desarrollo de circuitos integrados en Brasil.

El trabajo presentado en esta tesis se originó en el impulso único creado por la combinación del nacimiento de la televisión digital en Brasil y la creación del Programa de CI-Brasil por el gobierno brasileño. Sin esta combinación no hubiera sido posible realizar estes tipo de proyectos en Brasil. Estos proyectos han sido un trayecto largo y costoso, aunque meritorio desde el punto de vista científico y tecnológico, hacia un Circuito Integrado brasileño de punta y de baja complejidad para DTV, con buenas perspectivas de economía de escala debido al hecho que al inicio de este proyecto, el estándar ISDB-T no fue adoptado por varios países como DVB-T.

Durante el desarrollo del receptor ISDB-T propuesto en esta tesis, se observó que debido a las dimensiones continentales de Brasil, la DTTB no sería suficiente para cubrir todo el país con la señal de televisión digital abierta, especialmente para el caso de localizaciones remotas, apartadas de las regiones de alta densidad urbana. Una elección natural para superar este problema es utilizar el sistema de radiodifusión por satélite, como medio para distribuir la señal de televisión digital abierta y de alta definicin directamente a los hogares, sin la necesidad de tener un sistema DTTB local. En ese momento, el Instituto de Investigación Eldorado e Idea! Sistemas Electrónicos, previeron que en un futuro cercano habría un sistema de distribución abierto para DTV de alta definición a través de satélite en Brasil. Con base en eso, el Instituto de Investigación Eldorado decidió que sería necesario crear un nuevo ASIC para la recepción de radiodifusión por satélite. En ese tiempo, el estándar DVB-S2 era el candidato más fuerte para eso, y esta suposición sigue en pie hoy en día. Por lo tanto, se decidió aplicar a una nueva ronda de financiación de recursos del MCTI - que se concedió - con el fin de iniciar el nuevo proyecto con esto propósito.

En esta tesis se analiza en detalle la Arquitectura y algoritmos propuestos para la implementación de un receptor ISDB-T de baja complejidad y frecuencia intermedia (IF) en Application Specific Integrated Circuit (ASIC) CMOS. La arquitectura aquí propuesta se basa fuertemente en el algoritmo del computador digital para rotación de coordenadas (CORDIC), el cual es un algoritmo simple, eficiente y adecuado para implementaciones VLSI. El receptor hace frente a las deficiencias inherentes a las transmisiones por canales inalámbricos y los cristales del receptor. La tesis también

analiza la metodología adoptada y presenta los resultados de la implementación. El desempeño del receptor es presentado y comparado con los obtenidos por medio de simulaciones.

Por otro lado, la tesis también presenta la arquitectura y los algoritmos para un receptor DVB-S2 dirigido a la implementación en ASIC. Sin embargo, a diferencia del receptor ISDB-T, se introducen sólo los resultados preliminares de implementación en ASIC. Esto se hizo principalmente con el fin de tener una estimación temprana del área del die para demostrar que el proyecto en ASIC es económicamente viable, así como para verificar posibles errores en etapa temprana. Como en el caso de receptor ISDB-T, este receptor se basa fuertemente en el algoritmo CORDIC y fue un prototipado en FPGA. La metodología utilizada para el segundo receptor se deriva de la utilizada para el receptor ISDB-T, con adiciones menores, dadas las características del proyecto.

Resum

La primera generació de Televisió Digital Terrestre (TDT) ha estat en servici durant més d'una dècada. En 2013, diversos països ja van completar la transició de la radiodifusió de televisió analògica a la digital, i la majoria van ser a Europa. A Amèrica del Sud, després de diversos estudis i assajos, Brasil va adoptar l'estàndard japonés amb algunes innovacions. Va ser el primer país a difondre comercialment televisió digital a eixa regió, el 2007. No obstant això, en 2015 el país encara no havia completat la transició a la TV digital.

Japó i Brasil van comenar els servicis de Radiodifusió de Televisió Terrestre Digital (DTTB) al desembre de 2003 i al desembre de 2007, respectivament, utilitzant la Radiodifusió Digital amb Servicis Integrats de (ISDB-T), també coneguda com a ARIB STD-B31 [1]. Òbviament, la ISDB-T estava inspirada en la DVB-T [2], la primera generació de la norma europea de Radiodifusió de Televisió Terrestre Digital. No obstant això, va superar una de les debilitats de la DVB-T, és a dir, el baix rendiment amb soroll de commutació. Tant la DVB-T com la ISDB-T fan ús del Múltiple per Divisió Ortogonal de Freqëència (OFDM) com a tècnica de transmissió. A causa de les seues característiques úniques, el OFDM ha estat escollit per diversos dissenyadors de sistemes sense fil i organismes de normalització. Per tal de donar una idea de la importància de la tècnica de transmissió OFDM, podem esmentar alguns exemples de normes que la fan servir: IEEE802.15.4.g, IEEE802.11a/g/n/ac, Wimax, DVB-T, IEEE 802.20 i l'enlla de baixada estàndard de banda ampla móbil 3GPP Long Term Evolution (LTE).

Abans d'adoptar la ISDB-T, el govern brasiler va fer un enorme esfor, en fer col.laborar universitats, centres de R+D, la indústria de radiodifusió i les emissores locals a fi d'establir un estàndard brasiler de televisió digital que explorés tècniques punteres, com ara la Comprovació de Paritat de Baixa Densitat (LDPC) i la codificació espai-temps. No obstant això, després de tot eixe esfor, a causa de l'economia d'escala i les seues característiques úniques d'eixe moment, el govern brasiler va adoptar la Capa Física de la ISDB-T, amb algunes actualitzacions en codificació de vídeo i àudio i interactivitat, com a base per al Sistema Brasiler de Televisió Digital Terrestre (SBTVD). Així, l'ISDB-T amb estes actualitzacions va passar a conéixer-se com a ISDB-Tb. Al

lloc web del Grup d'Experts de Radiodifusió Digital (DiBEG) (www.dibeg.org), el lector pot trobar l'equivalència entre les normes brasileres i japoneses de Televisió Digital Terrestre.

Al juny de 2005, el Comité de l'Àrea de Tecnologia de la Informació (CATI) del Ministeri de Ciència i Tecnologia i Innovació del Brasil (MCTI) va aprovar la incorporació del programa CI Brasil al Programa Nacional de Microelectrnica (PNM), que és un programa prioritari dins de les polítiques brasileres d'informàtica i automatització. Els principals objectius de CI Brasil són la qualificació formal dels dissenyadors de circuits integrats, el suport a la creació d'empreses de semiconductors centrades en projectes de circuits integrats dins del Brasil i l'atracció d'empreses de semiconductors centrades en el disseny i desenvolupament de circuits integrats al Brasil.

El treball presentat en esta tesi es va originar en l'impuls únic creat per la combinació del naixement de la televisió digital al Brasil i la creació del programa Brasil CI pel govern brasiler. Sense esta combinació no hauria estat possible realitzar este tipus de projectes a Brasil. Estos projectes han suposat un viatge llarg i costós, tot i que digne científicament i tecnològica, cap a un circuit integrat punter de baixa complexitat per a la TDT brasilera, amb bones perspectives d'economia d'escala perquè a l'inici d'este projecte l'estàndard ISDB-T no va ser adoptat per diversos països, com el DVB-T.

Durant el desenvolupament del receptor de ISDB-T proposat en esta tesi, va resultar que, a causa de les dimensions continentals de Brasil, la DTTB no seria suficient per cobrir tot el país amb el senyal de TDT oberta, especialment pel que fa a les localitzacions remotes allunyades de les regions d'alta densitat urbana. Una elecció natural per superar este problema és utilitzar el sistema de radiodifusió per satèl.lit com a mitjà per distribuir el senyal de TDT directament a les llars, sense la necessitat de tenir un sistema DTTB local. En este moment, l'Institut de Recerca Eldorado i Idea! Sistemes Electrònics van preveure que, en un futur pròxim, no hi hauria a Brasil un sistema de distribució oberta de TDT d'alta definició a través de satèl.lit. D'acord amb això, l'Institut de Recerca Eldorado va decidir que seria necessari crear un nou ASIC per a la recepció de radiodifusió per satèl.lit. En eixe moment, l'estàndard DVB-S2 era el candidat més fort per aò i esta suposició segueix en peus hui en dia. Per tant, es va decidir demanar una nova ronda de finanament de recursos del MCTI que es va concedir , per tal d'iniciar el nou projecte.

En esta tesi s'analitza en detall l'arquitectura i els algorismes proposats per l'execuci d'un receptor ISDB-T de Freqüència Intermèdia (FI) de baixa complexitat sobre CMOS de Circuit Integrat d'Aplicacions Específiques (ASIC). L'arquitectura ací proposada es basa molt en l'algorisme de l'Ordinador Digital de Rotació de Coordenades (CORDIC), que és un algorisme simple i eficient adequat per implementacions VLSI. El receptor fa front a les deficiències inherents a la transmissió de canals sense fil i els cristalls del receptor. Esta tesi també analitza la metodologia adoptada i presenta els resultats de l'execució. Es presenta el rendiment del receptor i es compara amb els

obtinguts per mitjà de simulacions.

D'altra banda, esta tesi també presenta l'arquitectura i els algorismes d'un receptor de DVB-S2 de cara a la seua implementació en ASIC. No obstant això, a diferència del receptor ISDB-T, només s'introdueixen resultats preliminars d'implementació en ASIC. Això es va fer principalment amb la finalitat de tenir una estimació primerenca de la zona de dau per demostrar que el projecte en ASIC és econòmicament viable, així com per verificar possibles errors en l'etapa primerenca. Com en el cas del receptor ISDB-T, este receptor es basa molt en l'algorisme CORDIC i va ser un prototip de FPGA. La metodologia utilitzada per al segon receptor es deriva de la utilitzada per al receptor ISDB-T, amb addicions de menor importància, ateses les característiques del projecte.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Motivation

This work was motivated by two important events that occurred in Brazil around 2007:

- the beginning of the broadcasting of Digital Television (DTV);
- the consolidation of the National Programs on Microelectronics (PNM).

Together, these events created a favorable scenario for the project of a high complexity Integrated Circuit (IC) for the demodulation of DTV signal.

Each one of these events alone, would not to be enough to motivate this work, specially because the high costs and the lack of tradition in the country in performing this kind of project. On the other hand, if the project is successful, the country will have acquired two important technological expertise:

- the full modeling of a digital receiver targeting Integrated Circuit implementation for real world applications;
- the capability of implementing a high-complexity Integrated Circuit (in terms of number of gates).

In the possession of those two expertise, new and more complex projects can be carried out in the future, which would be drivers for innovations in both fields. In addition, several other related areas could also advance, creating a virtuoso circle. A straight example of that is the DVB-S2 receiver also presented in this thesis.

1.2 From Landell to Digital Television in a Glance

Since 1900, when one of the radio pioneers, a Brazilian catholic priest named Landell de Moura [6], transmitted voice signal through a wireless system, in São Paulo, in the presence of a select audience, a lot has happened in the Brazilian wired and wireless communications scenario. Regarding TV broadcasting, the first experiments took place at Rio de Janeiro in 1929. The first automatic telephones were also installed in the same year [7]. The first commercial radio transmission company came out in 1950, and after 6 years television reached 1 million users. The first microwave link was installed in 1957. The color TV standard PAL-M started broadcasting in 1972. Only nineteen years would separate the first color TV transmission and the beginning of the studies on Digital Television (DTV) in Brazil.

1.2.1 DTV First Efforts and Tests

DTV started being studied in Brazil in 1991, when a committee was established with the aim of creating a policy for the sector. In 1993 the National Broadcast Association (NAB) invited the Brazilian Radio and TV Association (ABERT) to take part of a study group for the development and analysis of Coded Orthogonal Frequency Division Multiplex (COFDM) digital transmission technology, applied to TV channels with 6 MHz bandwidth [8]. In 1995, a group involving ABERT and the Brazilian Radio and TV Society (SET) was created to bring together TV broadcasters, experts, R&D institutions and universities to help broadcasters acquire expertise on the DTV transmission technology. During this process Europe adopted the OFDM-based standard DVB-T, and the U.S. adopted ATSC (8VSB-based technology).

In 1998, the Brazilian Telecommunications Agency (ANATEL) joined the process, promoting a public consultation to conduct experiments on DTV systems. In the same year, ABERT/SET and Makenzie University signed an agreement to create a laboratory for DTV testing, using private funding. Other partners joined this group, e.g. industries, broadcasters and R&D centers. While the tests on DVB-T and ATSC technologies started in 1999,the ISDB-T tests began later, because this standard was still under development. First results of that work were presented in the year 2000 in [9]. In 2003, ABERT/SET and Makenzie University signed a new agreement to verify the evolution of the DVB-T, ATSC and ISDB-T standards is signed. Still in 2003, the government published the Brazilian Digital Television System (SBTVD) implementation guidelines and in 2004 they established that within 1 year the country would set a roadmap for DTV.

1.2.2 The R&D Efforts

In 2004, the Science & Technology Ministry along with the Communications Ministry, by means of the Studies and Projects Funding (FINEP) and the National Telecommunications Fund (FUNTEL), launched an open call to select projects to create consortia on DTV related subjects. In response to that call, 22 consortia were created, bringing together universities, R&D centers and industry. About 100 institutions and 1000 people were involved in these consortia. The aim of this joint effort was to provide results and analysis to support the Government to decide among the existent DTV standard and a new one created/suggested by the consortia. One of the proposal from the university side was to add the latest technology used for multimedia broadcasting and create a new local standard. One example of the outcome of that effort is the research presented in [10] called MI-SBTVD. That work proposes an OFDM-based Physical Layer which contains concatenated Reed-Solomon(RS) and Low Density Parity Check(LDPC) codes. Furthermore, the authors made use of Space-Time-Code (STC) based on Alamouti Scheme [11], to achieve diversity gain and reduce the receiver complexity. Another contribution made by the consortia is the declarative middleware GINGA-NCL [12].

1.2.3 The Decision and Some Numbers

In February 2006, the Telecommunications Research and Development Center (CPqD) released the final report with the recommendations for the SBTVD. In June 2006, after all those efforts reported on previous sections, the Brazilian government made the decision to adopt the ISDB-T standard as basis for the SBTVD, with the adoption of the following innovations:

- H.264 standard for Digital Video Coding;
- GINGA Middleware;
- The use of Wimax technology in the interactivity channel.

ISDB-T in combination with these improvements is the so called ISDB-Tb. In November 2006, the government created the Forum for the Brazilian System of Terrestrial Digital Television (Forum SBTVD). This group, consisting of broadcasters, equipment manufacturers, software companies, R&D Institutes and university representatives, was formed to support the decisions on which innovations would be incorporated in the SBTVD.

The transmission of ISDB-Tb started in São Paulo in December 2007 and in 2012 DTV was spread all over the country. According to the information found in [13], from September 2012:

- Brazil has about 10.800 TV networks;
- The number of cities with DTV signal was about 450;
- 46% of the Brazilian households had access to DTV signal;
- 70% of the inhabitants in the states of Rio de Janeiro and São Paulo, i.e. 40 million people, could access DTV signal;
- 11 countries in Central and South America had adopted ISDB-Tb.

1.2.4 How DTV Influenced the Brazilian Industry of Broadcasting

Since the beginning of the DTV studies in Brazil, local companies realized the potential revolution DTV would cause in the local broadcast and consumer industries. Taking advantage of the momentum most of the Analog TV Transmitter manufacturers started developing skills to create DTV transmitters or make agreements with foreign companies to produce or import DTV equipments. There are 3 Brazilian companies that manufacture RF transmitters for DTV, one of which was recently sold to a well known Japanese company, as well as more companies that designed locally and/or also made international agreements to produce locally a number of equipments: e.g. re-multiplexers, H.264 encoders/decoders, digital modulators (for terrestrial and satellite broadcasting), gap fillers, low and high power RF transmitters, antennas, channel filters, high and low-end set-top-boxes. Some of these Brazilian and foreign companies are opening plants overseas or making agreements with representatives, in order to take advantage of the spread of ISDB-Tb based DTV around South America.

In the transmitter path, most of the Brazilian companies incorporate some locally designed and implemented technology. Nevertheless, regarding TV related goods and receivers for set-top-boxes (STB), TVs and PCs, most of the companies that dominate the market are non-Brazilian. As far as I know, the research presented in this work is the only original attempt of a Brazilian R&D center along with a private Brazilian company to design and create DTV ASIC receivers for terrestrial and satellite broadcasting reception of high definition television signal, with commercial or non-commercial purposes, that has the potential to reach not only Brazil but a large number of countries within South and Central America. Furthermore, DVB-S2 receiver has a global appealing, giving to this specific receiver project and its evolution the possibility to be used worldwide. This could provide a global reach, for a high complexity semiconductor project made in Brazil by Brazilian companies, that never was seen before.

1.3 The National Program on Microelectronics and the Ambitions of the Brazilian Government

Since 1947, when the transistor effect was discovered at Bell Labs, and the development of Planar Process for Integrated Circuits (ICs) fabrication in 1959, at Fairchild, what resulted in the first commercial IC's in 1962, the Semiconductor area had evolved immensely and due to its strong R&D basis, had a very important role in the development of several other areas, from the aerospace industry to mass consumer electronics applications, alongside medical, agricultural and industrial applications. It is a challenging, almost impossible, task to find a single piece of equipment around us at this moment that does not contain an IC. Dominating the entire cycle of IC creation, from conception and design, to packaging is a challenging task and several countries tried to get there. Examples of countries that successfully achieved this were: USA (where the IC was invented), Japan, Germany, France, the United Kingdom, China, Malaysia, the Philippines, Thailand, Indonesia, South Corea, Taiwan and Singapore.

The basis for the establishment of the Brazilian electronic industry started in the 1960s and got stronger in the 1970s with the fast expansion of the consumer goods market. In the beginning of the 1980s, under the protection of the government's Informatics Policies there were about 23 electronic components industries in operation in Brazil. Within that time, several efforts were done by the universities, R&D centers and industry to strengthen the semiconductor area. Nevertheless, between 1980 and 2000 the industry suffered from the stagnation and strong crash occurred in the beginning of the 1990s due to the end the protective policies existent until that time. According to the Brazilian Bank for Social and Economic Development (BNDS) (see [14]), in the end of the 1990s Brazil had a strong commercial deficit due to electronic components, which was in the order of several billions of US dollar. Therefore, it was clear for the government that the creation of a Foundry in Brazil could solve part of that imbalance and also promote the economical development in the area. Nevertheless, in order to put this idea into practice the country had to create the basis for that. The ability to design Integrated Circuits and the capacity to produce and package ICs are paramount for the development of the Microelectronics industry in the country.

The Microelectronics industry is strongly R&D based. Therefore, the country that expects to succeed in this area should have plenty of highly-skilled manpower on this subject and related areas. Therefore, after several studies, in June 2005 the Committee of Information Technology Area (CATI) of Brazilian Ministry of Science and Technology and Innovation MCTI approved the incorporation of IC-Brazil Program, in the National Program for Microelectronics (PNM) which is a Priority Program within the Brazilian Policies for Informatics and Automation. The main goals of IC-Brazil [15] are formal qualification of IC designers, support to the creation of semiconductors companies focused on projects of ICs within Brazil, and the attraction of semiconductors companies focused on project and development of ICs

in Brazil.

As I consider important to show how and why Brazil got to the current PNM, I provide an short historical view of the semiconductor market in Brazil that can be found within the Appendix B of this thesis.

1.4 Thesis Objectives

The main objective of this thesis is to develop two low computational complexity receivers. The first is an Intermediate Frequency (IF) Receiver for ISDB-T signal demodulation, specially for fixed-reception of high definition television. Nevertheless, it is worth to note that the ISDB-T standard inherently allows the reception of moderate bit rates in moderate speeds without any additional signal processing. The second is a DVB-S2 satellite receiver devoted to the reception of open high definition digital television signal. To accomplish this development, subjects in several fields need to be studied and covered, e.g. systems design, wireless communications systems modeling, impairments and wireless channel modeling, baseband modulation and detection techniques, Orthogonal Frequency Division Multiplex(OFDM) access technique, LDPC and BCH coding and decoding, several aspects of FPGA and ASIC design implementation, radio frequency and satellite systems engineering, tuner specifications and system test techniques.

To accomplish the main objective of this thesis, several specific objectives were pursued such as:

- achieve fundamental understanding of the ISDB-T and DVB-S2 standard;
- achieve fundamental understanding of OFDM-based Receivers;
- achieve fundamental satellite systems;
- modeling ISDB-T and DVB-S2 Transmitters;
- modeling wireless channels;
- modeling receiver impairments;
- explore algorithms for synchronization of ISDB-T and DVB-S2 signals;
- proposal micro-architectures for the receiver blocks of both receivers;
- modeling the full ISBD-T and DVB-S2 IF/Zero-IF receivers;
- proposal of suitable full architectures, for the IF/Zero-IF ISDB-T and DVB-S2 receivers to be implemented in FPGA and ASIC;

- system specifications for implementation of the receivers blocks in VHDL;
- provide continuous implementation aspects support to the VHDL designers and backend engineers ;
- FPGA receivers system tests;
- VLSI implementation system tests;
- comparison of system tests results with simulations or target performance defined in the standards;
- proposal of improvements for the second version of DTV VLSI implementation;
- provide DVB-S2 support to work under Inter-Symbols Interference scenario;

1.5 Outline of the Thesis and Survey of the Contributions

In this section, a preview of the thesis chapters is given. Furthermore, a survey of the main contributions is provided. The content of this thesis follows the logical order from the fundamental understanding of wireless communication systems issues (that affect the correct functioning of the receivers) to practical measurements and comparison with simulations results or standard defined target performance. A detailed view on the ISDB-T and DVB-S2 standards is also provided.

Chapter 2 presents the wireless communications basics, in order to show the main issues to handle when implementing wireless digital receivers. First, an overview on the OFDM transmission technique is given, as well as its main advantages and disadvantages. Next, the impairments caused by long and short-term characteristics of the wireless channels is provided. Finally, the main impairments caused by the analog front-end, oscillator crystal and Analog-to-Digital conversion are introduced.

ISDB-T and DVB-S2 standards are full of details - specially the ISDB-T standard - that must be clearly understood, otherwise any misinterpretation can lead to a wrong implementation specification definition, which in turn can drive to a receiver implementation that can work in simulations, but simply would not work in hardware. Any mistake can be highly costly, in terms of time and financial budget, for an ASIC implementation. For that reason, in Chapter 3 an in-deep look into the ISDB-T and DVB-S2 standards are provided in order to show the structure of frames, embedded pilots, header content and format, which will be explored by the synchronization and equalization algorithms.

As the ISDB-T receiver was the first to be conceived and implemented and its successful design would boost the existence of the DVB-S2 receiver project, I devote

Chapter 4 to present details on the methodology used to design such receiver. The same methodology, with small variation, was used during the project of the DVB-S2 receiver.

- The methodology used in the project, the first results on the receiver implementation and the connection of the ISDB-T receiver project with the IC-Brazil Program were presented, as invited work, in [16].
- Seminars and personal tutoring, on ISDB-T and DVB-S2 standards, were also done with the aim to check if the interpretation of the standards that each member of the development team made was correct. This was done to minimize or avoid mistakes that could cause a delay in the project or could lead to a catastrophic error that would make the final design not work properly.

Within Chapter 5 the ISDB-T receiver proposed architecture and algorithms used to overcome the impairments presented in Chapter 2, are shown in details.

- The algorithms proposed in this chapter are result of an in-depth literature survey on OFDM-based wireless digital receivers and implementation aspects of wireless communication systems.
- The proposed architecture is strongly based on the CORDIC algorithm that was used to reduce the implementation complexity and make it possible its reuse in several operations within the receiver.
- One example of usage is within the Channel Estimator, where the division is replaced by a CORDIC-based divider. Nevertheless, to overcome the limitation imposed by CORDIC, i.e. the division result shall be smaller or equal to 2, a mechanism based on left base-2 shifts was created to avoid the overflow. The proposed channel estimator was submitted to patent at the USPTO (United States Patent and Trademark Office) [17].
- The ISDB-T receiver core and the IC architectures were presented in [18].
- The final IC implementation and the BER performance of the implemented architecture in ASIC were submitted to publication in [19].
- Other examples of contributions are the proposal of an architecture for the implementation of the Sampling Clock and Fine Frequency Estimators both using the CORDIC algorithm.
- In order to evaluate the performance of the proposed algorithms and the receiver architecture, a simulation tool was developed. It models the ISDB-T transmitter, wireless channel and receiver impairments, receiver algorithms and other ISDB-T blocks. Some simulation results are presented within this Chapter and BER simulations for AWGN and wireless channels are presented in Chapter 6.

Chapter 6 presents the results of the implementation of the ISDB-T receiver architecture proposed in this thesis and the laboratory equipment used to evaluate the receiver performance and to generate real stimulus for debugging the algorithms and VHDL code. It also shows the FPGA prototyping environment, the implemented ASIC, some preliminary products generated from the implemented ASIC and experimental results used to validate the ASIC, named DTV01.

- The ASIC implementation of the proposed receiver architecture was presented in [18].
- Within this chapter, the performance of the implemented ASIC under AWGN and wireless channels (using channel emulator) is compared to that of the receiver modeled using Matlab. These results along with the ASIC implementation and Methodology were submitted for publication in [19].

Chapter 7 is devoted to introduce the proposed DVB-S2 receiver architecture and algorithms, to handle the real world impairments and that will be implemented in ASIC. Actually, the chapter introduces two architectures, one that is currently prototyped and tested in FPGA, and the a second that is the one which will be implemented in ASIC, for commercial purposes. It is worth to mention that all algorithms and blocks present in the current implemented architecture will be present in the final architecture. The main difference between both is that the final architecture contains Adaptive Equalizer based on Decision Feedback Equalizer (DFE), Soft Demapper for all modulations and support to variable sample rate.

- The algorithms proposed in this chapter are result of a long survey in the literature on DVB-S2 and others receivers devoted to wireless communications such as satellite, cable or cellular systems.
- The architectures proposed in this chapter are strongly based on CORDIC algorithm, that was reused within several receiver blocks.
- The results on the first receiver integration was presented in [20].
- The initial results on the implementation of min-sum based LDCP decoder were presented in [21].
- Syndrome calculator architecture for the BCH decoder was presented in [22]. The full architecture of the BCH decoder and its first VLSI estimates were presented in [23]. This work won the Best Paper Award and was selected to be submitted to [24] in a extended version;
- The proposed hard and soft demappers were presented in [25].
- The physical validations and BER measurements for the FEC decoding subsystem was presented in [26]. In the same work the platform for BER measurements was introduced. This work won the Best Paper Award.

- An research area that is growing in interest is the mobile reception of satellite signals. This means that some satellite receivers will work in environments subject to a high number of scatters. Furthermore, in the case of fixed-reception the proximity to scatters and the cable distribution systems can be cause ISI, that can be significant specially as the DVB-S2 bandwidth increases. For those reasons I propose the use of a Decision Feedback Equalizer to make the reception of DVB-S2 reliable on those scenarios. The Decision Feedback Equalizer proposed is presented in [27], and its use to mitigate the nonlinear effects of the satellite channel is addressed in [28].
- A first full integration of the receiver (i.e the Signal Processing portion integrated with the FEC Decoding subsystem), BER measurements for AWGN, FPGA prototyping results and VLSI estimates were presented in [29].
- An improvement on the Fine Frequency estimator architecture was presented in [30]. The same concept can be extend to the Coarse Frequency estimator .

Within Chapter 8 I present the Methodology used in the DVB-S2 receiver project, FPGA prototyping results, VLSI implementation, BER performance measurement for the entire receiver under AWGN and with real world impairments, and BER performance for the isolate Forward Error Correction (FEC) decoding subsystem for AWGN only.

Finally, Chapter 9 is devoted to the conclusion and future work.

1.6 Summary of my Main Roles and Responsibilities on ISDB-T and DVB-S2 Receivers Projects

Before ending this Chapter, it is worth to mention which were my main duties during the ISDB-T and DVB-S2 receivers projects. In boot cases I was the most senior member of the project teams with respect to wireless communications systems and its implementation aspects in hardware as well as in system aspects of RF. I acted as system architect, algorithm designer, project manager, system test engineer and hardware designer, and verification engineer for a couple of block such as CORDIC and FIR filter. Following a short description of my roles in each of this positions:

- As a senior member my main functions were training members in Digital Signal Processing, implementation aspects of wireless communication systems, RF, system test, as well as guarantee the quality of the design in terms of system performance;
- As system architecture, my main roles were propose the digital receiver architecture and algorithms for implementation in FPGA and VLSI, choosing the

appropriate RF front-ends to be used during the projects, and provide continuous feedback for the project team in the case of architecture and/or algorithm changes;

- Modeling the transmitter and receiver blocks, write specifications or supervise engineers when doing this tasks were my roles when acting as algorithm designer;
- As project manager, my functions were basically guarantee that all design and simulations were done and the results were delivered in the right time, in order to minimize the duration of the project and make it evolve in a coherent order . Other role was to interact with external partners;
- As hardware designer and verification engineer, I was responsible for designing the first versions of the CORDIC as well as FIR filter in VHDL and test them using GHDL/GTKWAVE [31] [32], and in FPGA;
- As system test engineer, I was responsible to set up the laboratory environment, training the project members on RF systems aspects and how to use the laboratory equipments, such as channel emulator. I was also responsible to test the receiver every time when was needed: 1) starting from the first ISDB-T signal capture in FPGA (to be used during the first receiver blocks implementation using Matlab) ; 2) during the endless partial tests to validate the implementation and integration of the blocks in FPGA; 3) until the final integration system tests, before starting the ASIC design flow. My final task acting as system test engineer was the test of the implemented ASIC at Eldorado R&D Center laboratory. Nevertheless, before that, I also gave support to ASIC designers (front-end and backend) on the ISDB-T receiver functioning and interpretation of the RTL and gate level tests using the ISDB-T real signal, captured using our capture platform.

Chapter 2

Wireless Communications Basics

The main goal of any wireless receiver is to cope with all wireless channel and hardware impairments that the receiver is subjected to, in order to deliver the bit stream/or packets with the lowest possible bit error rate[1]. At the same time, the implementation of receivers in VLSI creates limitations like minimum die area and cost, and low power consumption (usually for handheld and portable devices), for instance. In the one hand, the receiver design shall obey the VLSI implementation criteria/limitations. On the other hand, the receiver shall contain the adequate algorithms to overcome the channel and hardware impairments. Therefore, a design targeting practical implementation shall always achieve a trade-off between receiver complexity and performance. With the aim of achieving such a trade-off, it is necessary to have a clear understanding on all the phenomena and impairments that the receiver may be subjected to, in order to propose a suitable architecture containing the adequate algorithms. This chapter is devoted to the explanation of the main OFDM systems characteristics, wireless channel phenomena, and hardware and radio impairments that affect OFDM based and single carrier (such as DVB-S2) wireless communication systems. Nevertheless, it is worth to mention that a wireless receiver will not necessarily be affected significantly by all the phenomena and impairments described within this chapter. Therefore, following the natural order of a receiver design, the researchers, designers, systems architects and practitioners, shall understand the phenomena and then identify if the system under investigation/design is subjected to or significantly affected by those phenomena and impairments. Details on the DVB-S2 single carrier modulation are presented within Chapter 3 .

[1] Off course, some degradation in the receiver performance, due to fixed-point limitations, is allowed.

Figure 2.1 shows a generic OFDM system and the main radio, hardware and channel impairments the system is subjected to. The non-idealities presented in this chapter are generic, i.e. any communication system (based on OFDM or not) is subjected to, in certain level. In this chapter, the phenomena are described at a higher level of abstraction and in a brief manner. Nonetheless, those impairments that affect the proposed designs in this thesis, are revisited within Chapters 5 (for ISDB-T) and 7 (for DVB-S2).

2.1 OFDM - Orthogonal Frequency Division Multiplex

OFDM can be seen as a special case of multicarrier transmission, in which a single data stream is transmitted over a number of lower-rate subcarriers (see Figure 2.5), which reduces the influence of multipath fading as well as the equalizer complexity [33].

OFDM can be considered an access technique or a digital modulation. When it is used to transfer multiple user bit streams, it is considered a multiple access technique, and when it is used to transfer a single user bit stream it can be seen as a digital modulation.

In the case of OFDM, Orthogonal means that the multiple carriers belonging to an OFDM symbol do not cause interference on each other. In Figure 2.2, it is possible to see that the maximum of each subcarrier belonging to an OFDM symbol is always free of the interference caused by the side lobes of other subcarriers. In other words, the surrounding subcarriers always have values equal to zero at any subacarrier frequency.

The widespread way to generate an OFDM symbol is to take advantage of a basic property of the *Discrete Fourier Transform* (DFT). That is the orthogonality of the DFT sinusoids. The inverse of the *Fast Fourier Transform* (FFT), i.e. the IFFT, is the simplest and most efficient way to generate an OFDM symbol. The most commonly used algorithm to compute the FFT and IFFT is the well known *Cooley-Tukey FFT Algorithm*[34].

Given a sampled sequence vector $\mathbf{S}[n]$, where n is the OFDM symbol index, containing baseband modulated symbols, in frequency domain, as presented in Equation 2.1:

$$\mathbf{S}\left[n\right] = \left[s_0\left[n\right] s_1\left[n\right] \cdots s_{N-1}\left[n\right]\right]^T, \tag{2.1}$$

the OFDM symbol in the time domain is obtained by taking the Inverse DFT (i.e. IDFT) of $\mathbf{S}[n]$ as:

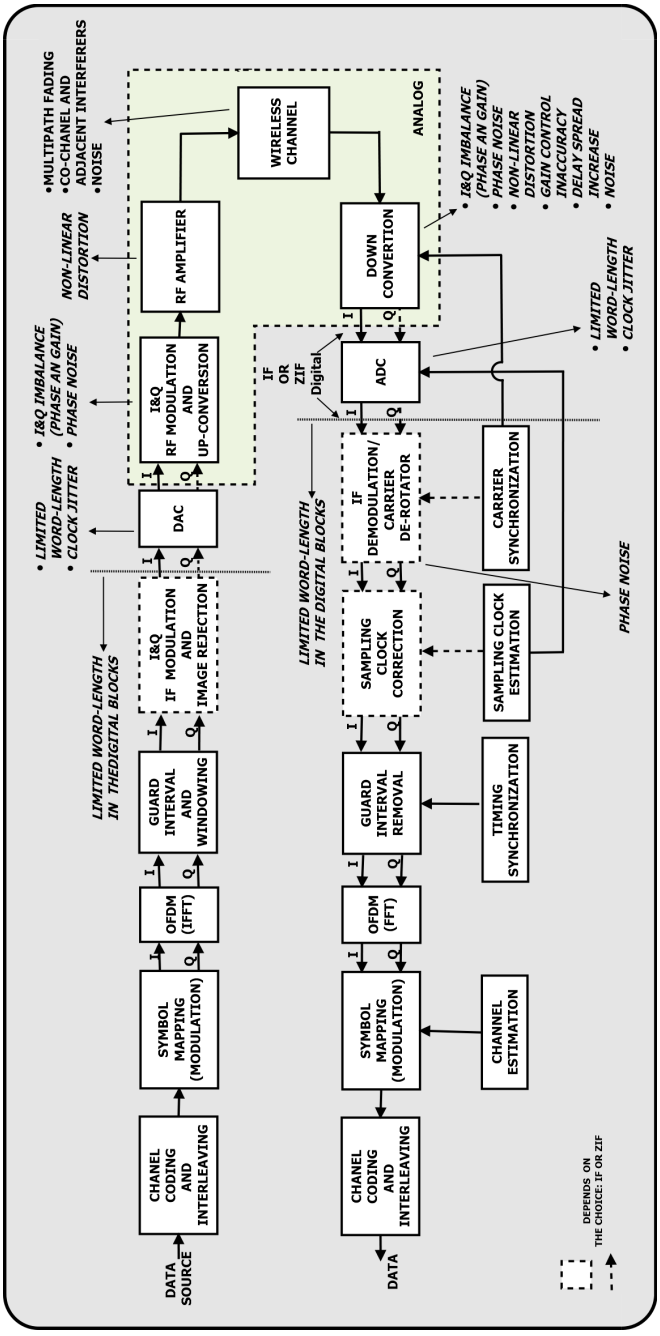


Figure 2.1: Simplified OFDM transceiver and, radio and mixed signal impairments, that affect OFDM based and single carrier wireless communication systems.

$$\mathbf{X}[n] = \mathscr{F}^{-1}\mathbf{S}[n], \tag{2.2}$$

where $\mathscr{F}^{-1}$, denotes the IDFT matrix, which is given by:

$$\mathscr{F}^{-1} = \frac{1}{N}\begin{bmatrix} 1 & 1 & 1 & \cdots & 1 \\ 1 & \exp^{j\frac{2\pi}{N}} & \exp^{j\frac{4\pi}{N}} & \cdots & \exp^{j\frac{2(N-1)\pi}{N}} \\ 1 & \exp^{j\frac{2\pi}{N}\times 2} & \exp^{j\frac{4\pi}{N}\times 2} & \cdots & \exp^{j\frac{2(N-1)\pi}{N}\times 2} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & \exp^{j\frac{2\pi}{N}\times(N-1)} & \exp^{j\frac{4\pi}{N}\times(N-1)} & \cdots & \exp^{j\frac{2(N-1)\pi}{N}\times(N-1)} \end{bmatrix} \tag{2.3}$$

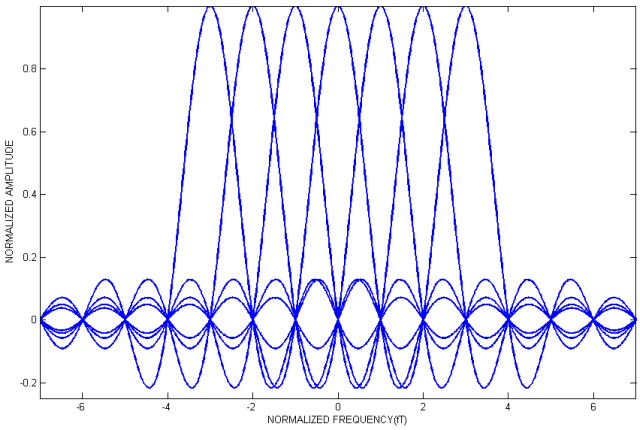


Figure 2.2: Example of an OFDM signal spectrum, showing the orthogonal subcarriers.

Equation 2.2 can be expanded as:

$$\mathbf{X}[n] = \sum_{k=0}^{N-1} \mathbf{X}_k[n], \tag{2.4}$$

where

$$\mathbf{X}_k[n] = \frac{1}{N} s_k[n]\,\mathbf{w}_k, \quad \text{for} \quad k = 0, 1, 2, \cdots, N-1 \tag{2.5}$$

and

$$\mathbf{w}_k = \frac{1}{N}\begin{bmatrix} 1 \\ \exp^{j\frac{2\pi k}{N}\times 1} \\ \exp^{j\frac{2\pi k}{N}\times 2} \\ \vdots \\ \exp^{j\frac{2\pi k}{N}\times (N-1)} \end{bmatrix}. \tag{2.6}$$

This shows that the kth symbol $s_k[n]$ modulates a complex carrier at frequency $f_k = \frac{2\pi k}{N}$ as presented in Figure 2.3.

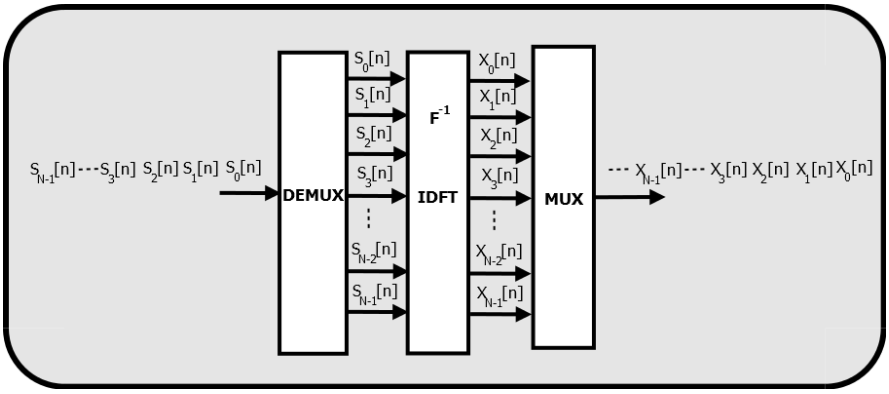


Figure 2.3: OFDM symbol generation using IDFT.

In order to be transmitted, the *OFDM Symbols* $\mathbf{X}[n]$ shall be concatenated as shown in Figure 2.4. The same figure shows a characteristic of OFDM-based transmission systems, that is the *Inter Symbol Interference* (ISI), caused by the echoes from the previous *OFDM Symbol*. The echoes, also called multipaths[1], are the scattering of the transmitted symbols and are caused by the reflection of the signal in the objects present all over the wireless environment. Multipath phenomena causes distortion in the signal and, when the path delays are larger than the symbol sampling period, ISI also occurs and causes loss of orthogonality among OFDM subcarriers. Therefore, complex equalization techniques would be necessary to eliminate ISI and recover the carrier orthogonality.

Due to the ISI, and as a consequence of the need of high-complexity equalization, the concept of *Guard Interval* (GI) is extensively used in OFDM systems, in order

[1]Multipath phenomena is going to be explained in details within Section 2.1.

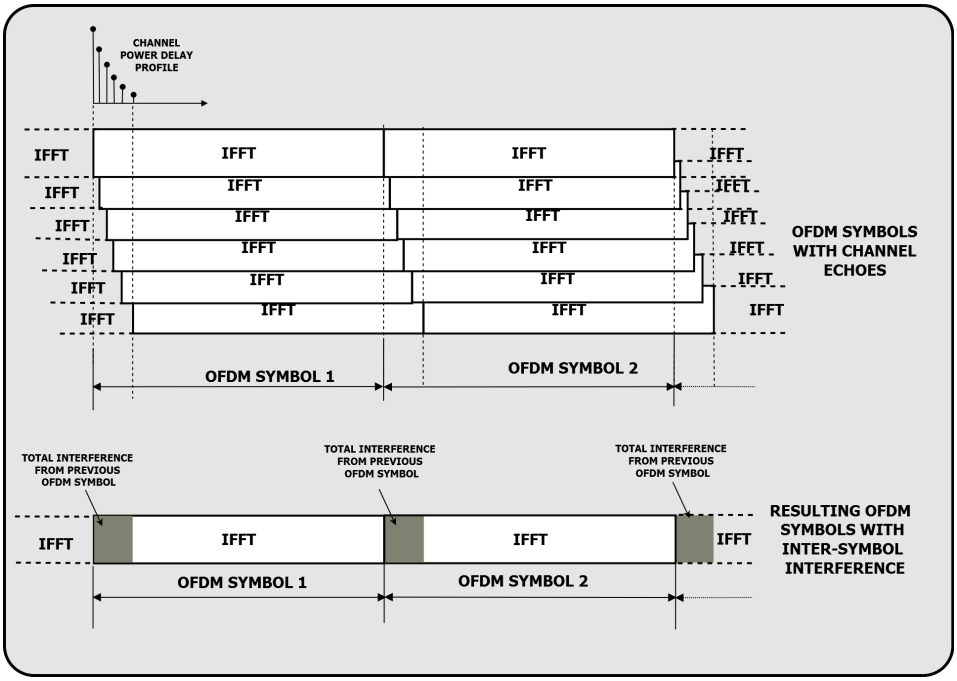


Figure 2.4: OFDM signal and wireless channel delay spread causing ISI.

to make it possible to make use of one-tap equalizer per subcarrier, to eliminate distortions.

Usually, in OFDM systems, the GI protection is implemented by copying the last portion of the OFDM Symbol (after IDFT) and adding it at the beginning of the *OFDM Symbol*, as can be seen in Figure 2.5. This makes it possible to create a zone where the multipaths of the previous symbols will fall inside. At the receiver, after the correct *OFDM Symbol* boundary synchronization, this region (GI) is discarded and DFT is performed. Then, the only remaining distortion can be equalized by a simple on-tap equalizer. It is worth to mention that, before being discarded, the GI can be used for receiver synchronization purposes, as will be shown later within Chapter 5.

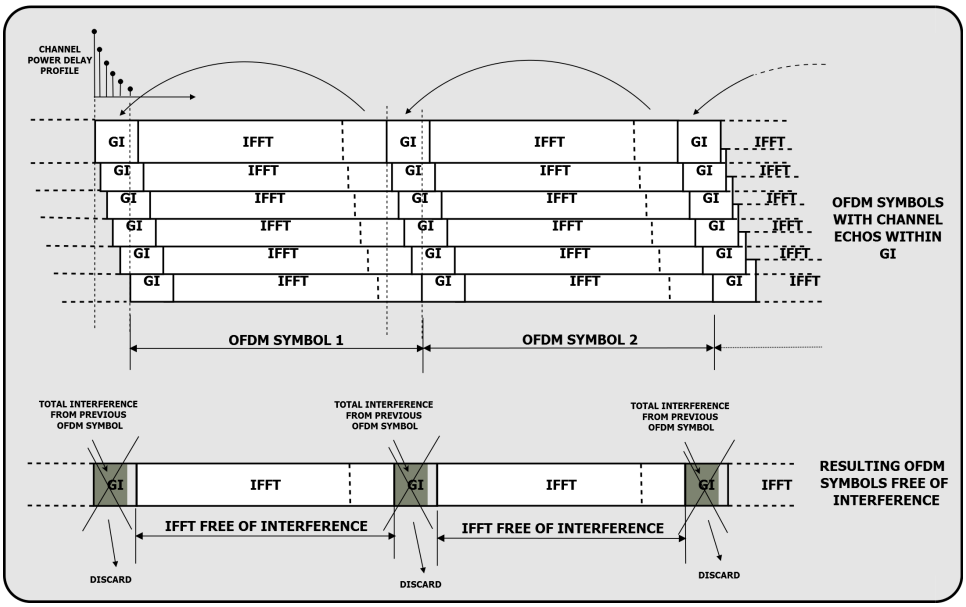


Figure 2.5: OFDM signal with GI and wireless channel delay spread causing no ISI.

2.1.1 OFDM Advantages and Drawbacks

OFDM is widely used in the emerging and future generations of high speed communication systems, due to its unique characteristics. Therefore, it looks like that

the advantages of using it overcomes the disadvantages. An example to highlight the importance of OFDM in the emerging and future communication systems, it is the combination of OFDM with the breakthrough technology for multiple antennas, *Multiple-Input Multiple-Output* (MIMO), the so called MIMO-OFDM [35], to achieve the near-Shannon limit performance for wireless channels, with low receiver complexity. This combination has been adopted for several standards as Wimax, LTE, IEEE 802.11n and IEEE 802.11ac. On the other hand, a well known disadvantage of OFDM is the fact that it is a non-constant envelope modulation and for that reason it has a large Peak-to-Average-Power-Ratio (PAPR), which requires the use of high dynamic range linear power amplifier. This Section is devoted to presenting the main advantages and drawbacks of using OFDM.

2.1.1.1 Advantages of OFDM

Despite the drawbacks of OFDM, it is worth to notice that most of the current standard, for duplex or simplex communications, for broadband radio and cable based accesses, make use of OFDM. A short list of the standards that make use of OFDM is: DVB-T/DVB-H, Wimax, ISDB-T, DVB-T2, IEEE 802.11a/g/n/ac, LTE, DAB, ADSL, VDSL2, DVB-C2, IEEE 1901 and IEEE 802.15.4g. It is also worth to mention that several works related to OFDM over optical fiber can be found in the scientific and technological literature.

2.1.1.1.1 High Spectral Efficiency

According to [36] and references therein, OFDM is a highly efficient modulation scheme which has been shown to approach the information theoretical capacity with water-filling across its subcarriers. Although subcarrier based power loading is less feasible in practice, adaptive coded modulation on OFDM subchannels (each subchannel comprises of a group of subcarriers) has already been adopted into IEEE standards. In [37], (within [38]) the authors show a plot (reproduced from [39]) from which a comparison of OFDM with *Single Carrier* (SC) modulation can be done. The plot shows that for high SNR the normalized transmission rate (bit/sec) of a SC system using *Decision Feedback Equalizer* (DFE) is equal to that of an OFDM system. Nevertheless, for low SNRs the normalized transmission rate of OFDM is much larger than SC with DFE.

2.1.1.1.2 One Tap Equalization

For a given delay spread, the implementation complexity is significantly lower than that of a single-carrier system with an equalizer [40]. Nevertheless, it is worth to mention that this assumption is based on the fact that all paths of the considered wireless channel shall fall into GI. If the GI does not accommodate the multipaths, extra time domain equalization would be necessary. Some degradation due to delay spread larger than GI are acceptable, but after a certain amount it is necessary to adopt a larger GI (which some times is not possible) or to adopt time domain equalization.

2.1.1.1.3 Robust Against ISI

Thanks to GI and the large OFDM symbol sampling time, OFDM is robust against ISI. For a given bit rate, OFDM Symbol sampling time is much larger[1] (especially for high bit rates) than that of a SC system. Nevertheless, as the wireless channel delay spread increases, ISI can occurs. For that reason GI was introduced in OFDM systems. This makes the OFDM-based system robust to ISI caused by large delay spread channels. However, at the cost of bandwidth efficiency reduction.

2.1.1.1.4 Resistance Against Fading and Narrowband Interference

OFDM is robust against frequency selective fading and narrowband interference. Narrowband interference only affects a small percentage of OFDM subcarriers [40] and the effects of multipath fading can be effectively mitigated using simple one-tap equalizers. In addition, the combination of OFDM with *Forward Error Correction*(FEC), the so-called COFDM, improves significantly the performance of OFDM systems over AWGN and in multipath fading scenarios.

2.1.1.1.5 Efficient Implementation Using FFT

The OFDM modem implementation complexity can be reduced using IFFT/FFT. According to [36], when using FFT, the number of operations in each *OFDM Symbol* is in the order of $N\text{Log}N$ and the number of operation of a Single Carrier system, with equalizer, is at least NLe, where Le is the number of taps in the equalizer.

[1] Of course it depends on the parameters used in the OFDM system. But I am considering a well designed/optimized OFDM system.

2.1.1.1.6 Low Sensitivity to Time Synchronization Errors

In opposition to SC systems, OFDM is resilient to timing errors. This characteristic is due to the long OFDM Symbols duration [38].

2.1.1.1.7 SFN and Macrodiversity

OFDM makes *Single Frequency Networks* possible [40], facilitating the implementation of transmitter macrodiversity. This characteristic is specially attractive for broadcasting systems.

2.1.1.2 Disadvantages of OFDM

A OFDM-based system has its unique characteristics and implementation challenges. As mentioned in Section 2.1.1.1, OFDM is less sensitive to timing issues than SC systems. On the other hand, OFDM has high sensitivity to frequency offsets due to the OFDM subcarriers proximity. This characteristic is the opposite of that of a SC system which is less sensitive to frequency offsets. This reinforces the need of thinking about OFDM in a different fashion than the SC system. To identify the major disadvantages of OFDM, its main drawbacks are described in this section.

2.1.1.2.1 Sensitivity to Doppler Shift

For a given bandwidth, to achieve spectral efficiency in OFDM, the number of subcarriers must be as large as possible (it depends on the OFDM design parameters). This makes the subcarriers distance diminish and, consequently increases the sensitivity to frequency errors. In addition, when the OFDM system is subject to doppler (due to relative displacement between transmitter and receiver) *Inter Carrier Interference* (ICI) occurs. ICI also occurs due to frequency errors.

2.1.1.2.2 Sensitive to Frequency Synchronization Errors

OFDM is highly sensitive to frequency synchronization errors. With frequency synchronization errors anything can go wrong [41]. Frequency errors have several

origins, for instance oscillator impairments. One effect of this error is the baseband constellation rotation. Depending on the level of error, and the type of baseband modulation this rotation can be catastrophic [42] within [43][1]. It can drive the receiver to a *Bit Error Rate* (BER) much higher than expected, making the reception of the signal with the due quality impossible .

2.1.1.2.3 Sensitive to Phase Noise

OFDM systems are sensitive to Phase Noise because it can induce ICI. Phase Noise can also cause *Common Phase Error* (CPE), which can be easily dealt with the one-tap equalizer. Nevertheless, the first phenomenon, when significant, needs special signal processing to have its influence reduced or mitigated. Besides, the OFDM system parameters are usually designed in such a way that phase-noise-induced ICI is several dB lower than the operational noise level [44].

2.1.1.2.4 Loss of Efficiency Due to GI

It is quite easy to see that the addition of the GI makes OFDM systems less spectrally efficient. Nevertheless, it allows the OFDM receiver to have less implementation complexity than SC systems. When designing an OFDM system, several aspects must be taken into account, and the size of the GI is one of them. According to [45], due to the use of the GI, the spectral efficiency of the system and the achievable data rate are reduced by a factor $\eta = Ts/(Ts + Tg)$, where Ts is the OFDM symbols duration, and Tg is the GI duration. In general the maximum GI length used is 1/4.

2.1.1.2.5 High PAPR

OFDM signal can be seen as the superposition of low bit rate streams modulated at different frequencies. This superposition is the cause of PAPR [36], in other words, the more the number of subcarriers the more the PAPR. The PAPR tends to reduce the power efficiency of the *Radio Frequency* (RF) amplifiers [40]. Also, high PAPR imposes stringent requirements on the ADs and DAs [36].

[1] e.g. BPSK is less susceptible to frequency errors than 64-QAM

2.2 Wireless Channels

Reliable mobile wireless communication is difficult to achieve due to the characteristics of wireless channels. The mobile radio channel places fundamental limitations on the performance of wireless communications systems [46]. In generic studies, the mobile radio channel is evaluated from statical propagation models and three mutually independent, multiplicative phenomena can be distinguished: Multipath Fading, Shadowing and Large Scale Path Loss [47]. To design a wireless communication system (transmitters, receivers or both) is of paramount importance to know the phenomena that occurs in the wireless channels. As part of this work is focused on the design of an ISDB-T receiver, I assume that the engineers and researchers involved in the definition of that standard, defined the system parameters based on the expected scenarios the system would work on. On the other hand, it is the responsibility of the receiver designer to guarantee that the receiver will work in those expected scenarios. To do so, a strong background in mobile channel characteristics is necessary. First, to model the wireless channels or use the existent channel simulator(s) in a correct fashion and, second, to test the implementation of the design. Otherwise, unexpected errors could occur or be misinterpreted.

The correct definition of the characteristics of wireless channel and radio impairments holds the highest responsibility for a successful or ill-fated design. In general, one could implement a wireless receiver assuming AWGN (i.e. without considering characteristics of wireless channels) reception and using a crystal (or oscillator) with very low *ppm*[1] deviation. This can leads to a successful demo, which will work in a very specific scenario. Nevertheless, when the receiver is subject to real channel characteristics the chances of failure are tremendous.

Figure 2.6 shows the main elements that affect the behavior of the signal propagated from the transmitter to the receiver, in an urban scenario. As can be seen, the transmitted wave reaches the receiver in several directions (angle of arrival). All paths that reach the receiver are attenuated due to the distance (the attenuation also depends on the RF frequency). The paths also suffer reflections and diffractions. Nevertheless, direct waves can reach the receiver also. Nevertheless in an urban environment with high density of buildings, most of the time the receivers are reached by reflected, refracted signals or both. It is easy to realize that in suburban environments the number of reflections is smaller than in urban environments and in rural environments the number of reflectors around the receiver is much smaller and sometimes they simply do not exist.

Propagation models that predict the mean signal strength for an arbitrary separation distance between transmitter and receiver are called *large-scale* propagation models and are useful to estimate the coverage area. Propagation models that char-

[1] Low ppm crystals can occult timing and frequency errors. They can work for packet-based systems, but for broadcast systems, which have continuous transmission, there will be a catastrophic error after certain time.

acterize the rapid fluctuations of the received signal strength over a short distance or short time durations are named *small-scale fading* models [46]. Following, the short and long term propagation models are going to be explained with focus on the characteristics or scenarios that affect the correct ISDB-T receiver functioning. From the point of view of the receiver, the long-term fading (shadowing) affects the dynamic range of the tuner and consequently defines the maximum distance of the transmitter from which the receiver is going to work properly. The *small-scale* fading affects the digital receiver in a more aggressive way. First, in the worst case scenario the deep fading forces the equalizer to work in a large dynamic range, and second because it affects the synchronization algorithms' performance.

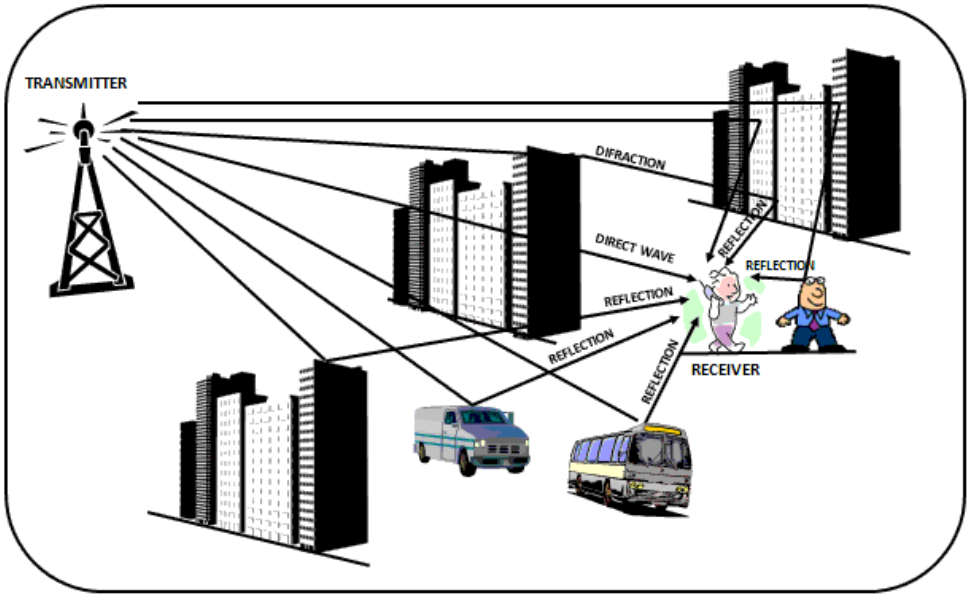


Figure 2.6: Multipath propagation.

2.2.1 Small-Scale Fading

Small-scale fading is characterized by statical parameters used to model the channel in time and frequency within a reduced area of coverage or time in order to allow considering the channel *Wide-Sense Stationary* (WSS)[1]. Multipath creates the small-scale fading effects. According to [46] the following are the most important effects:

[1]The correlation functions of the WSS channel are invariant for a time translation [48]. In other words the fading characteristics do not change over a certain time interval

- Rapid changes in the signal strength over a small distance or time interval;
- Random frequency modulation due to varying Doppler shifts on different multipath signals;
- Time dispersion caused by the multipath delays.

From the observation of Figure 2.6, some of the many factors that can influence the small scale fading characteristics (e.g. number of multipaths, moving scatters, static scatters, the existence of a direct wave, etc) can be observed. The multipaths are created due to reflections and refractions of the transmitted signal. The number of multipaths that reaches the receiver, and the path delays depend on the geographic characteristics of the area the receiver is located. In very dense urban environment (as the one shown in the figure) the number of reflections and refractions are much larger than in suburban area, while rural areas have even less multipaths than suburban areas. Nevertheless, it is worth to mention that each city (or neighborhood) has its own characteristics (density of buildings, hills, dimensions of the streets, average building height, type of material the buildings are made off, distance between buildings etc.), which leads to different multipath *Power Delay Profiles* for different locations. The main effects that causes small-scaling statistics are:

- Multipath propagation;
- Speed of the receiver;
- Speed of the receiver surrounding objects;
- The signal transmission bandwidth.

Small-scale variations can be directly related to the channel impulse response of the mobile radio channel. The impulse response is a wideband channel characterization and contains all the information necessary to simulate or analyze any type of transmission through a channel [46]. The mobile radio channel can be modeled as a linear filter with a time varying impulse response. The multipath received signal is made of a number of attenuated, time-delayed and phase-shifted replicas of the transmitted signal, whose baseband impulse response can be expressed as [49]:

$$h(t,\tau) = \sum_{i=0}^{N-1} \alpha_i(t,\tau) exp\left[j(2\pi f_c \tau_i(t) + \phi_i(t,\tau))\right] \delta(\tau - \tau_i(t)) \tag{2.7}$$

where, $\alpha_i(t,\tau)$ and $\tau_i(t)$ are the i-th multipath amplitudes and *Excess Delay* at time t. The term $2\pi f_c \tau_i(t) + \phi_i(t,\tau)$ is the phase and represents the phase shift due free space propagation, plus others channel phase shifts. In general the total phase shifts

are represented by a single variable $\theta(t,\tau) = 2\pi f_c \tau_i(t) + \phi_i(t,\tau)$, then Equation 2.7 becomes:

$$h(t,\tau) = \sum_{i=0}^{N-1} \alpha_i(t,\tau) exp\left(j\phi_i(t,\tau)\right) \delta(\tau - \tau_i(t)). \tag{2.8}$$

2.2.1.1 Power Delay Profile

Figure 2.7 shows an example of *Power Delay Profile* (PDP) for a given wireless channel. Due to practical reasons the PDP is usually sampled and has a limited number of resolvable paths (i.e. N). Assuming that the wireless channel impulse response is time invariant or at least WSS, the PDP, $P(t)$, is defined as the squared absolute value of the *Channel Impulse Response*:

$$P(t) = \sum_{i=0}^{N-1} \alpha_i^2(t,\tau)\delta(t - \tau_i). \tag{2.9}$$

The PDP represents the relative received power in function of the *Excess Delay* with respect to the first path. PDP are found by averaging instantaneous PDP measurements[50]. From the PDP , the instantaneous channel frequency $H(f)$ response can be computed.

2.2.1.2 Excess Delay

It is the relative delay of the i-th multipath component compared to the first arrival component of the discretized (into equal time delays τ) multipath delay axis. Each equal time delay (indexed by i) segment is called *excess delay bin*. The relative delay is given by τ_i.

2.2.1.2.1 Maximum Excess Delay

The *Maximum Excess Delay* is given by $N\delta\tau$, where N is the total number of equally spaced multipath components and $\delta\tau = \tau_0 - \tau_1$

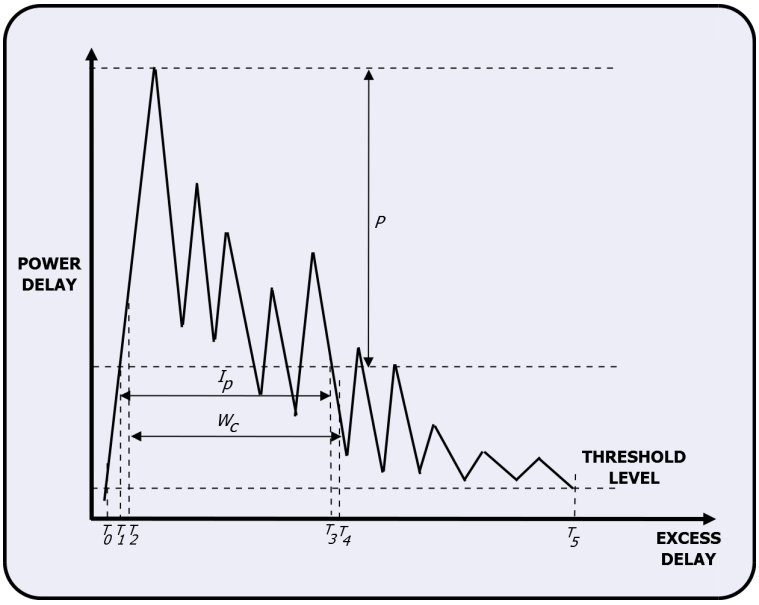


Figure 2.7: Power Delay Profile.

2.2.1.3 Mean Delay

$$\overline{\tau} = \frac{\int_{-\infty}^{+\infty} tP(t)dt}{\int_{-\infty}^{+\infty} P(t)dt} \tag{2.10}$$

Using Equation 2.9, Equation 2.10 becomes:

$$\overline{\tau} = \frac{\sum_{i=0}^{N-1} \tau_k \alpha_k^2}{\sum_{i=0}^{N-1} \alpha_k^2} \tag{2.11}$$

2.2.1.4 RMS Delay Spread

The *RMS Delay Spread is computed as:*

$$\tau_{RMS} = \sqrt{\frac{\int_{-\infty}^{+\infty} (t-\overline{\tau})^2 P(t)dt}{\int_{-\infty}^{+\infty} P(t)dt}} \tag{2.12}$$

Using Equation 2.9, Equation 2.12 becomes:

$$\tau_{RMS} = \sqrt{\frac{\sum_{i=0}^{N-1} (t-\overline{\tau})^2 \alpha_i^2 dt}{\sum_{i=0}^{N-1} \alpha_k^2}} \tag{2.13}$$

τ_{RMS} is used to give a rough indication of the maximum data rate that can be supported by the channel in a reliable manner, when no special measures, as equalization, are taken [50]. Also, according to [50] the rule of thumb presented below is often applied for the length of the channel impulse response:

$$\delta\tau = 4\tau_{RMS} \tag{2.14}$$

2.2.1.5 Coherence Bandwidth

The channel frequency response $H(f)$ autocorrelation is given by:

$$R(\delta f) = \int_{-\infty}^{+\infty} H(f)H^*(f+\delta f)df \tag{2.15}$$

where, H^* is the conjugate frequency response.

Exponential decaying delay profile is a common class of wireless channels[1] used in the literature. For a given received signal with local-mean power, the autocorrelation of this kind of channel can be computed as:

[1]Despite being a commonly used assumption, it is not a rule because there are several examples in the literature of non exponential decaying delay profiles. Very simple examples of this are the SFN channels power delay profile.

$$R(\delta f) = E\left[H(f)H^*(f+\delta f)\right] = \frac{1}{(1+j2\pi\tau_{RMS}\delta f)} \tag{2.16}$$

Where, $E[\ ]$ is the expectation operator.

According to [51], Equation 2.16 can be written as:

$$R(\delta f) = E\left[H(f)\ H^*(f+\delta f)\right] = \frac{1}{\left(1+(j2\pi\tau_{RMS}\delta f)^2\right)} \tag{2.17}$$

The *Coherence Bandwidth* is a measured for statistical averages over which the channel characteristics are uncorrelated. In other words, it is the band (3 dB) over which the frequency response is uncorrelated. For exponential decaying channel delay profile the *Coherence Bandwidth* is given by:

$$B_{coh} = \frac{1}{2\pi\tau_{RMS}} \tag{2.18}$$

2.2.1.6 Doppler Shift

Given the remote *Radio Frequency* source name **S** shown in Figure 2.8. Consider now a moving vehicle dislocating from position **X** to **Y** at a constant speed v. **X** and **Y** are so distant from **S** that it is possible to assume the angle θ to be the same for both positions. The distance between **X** and **Y** is $\delta l = d\cos\theta = v\delta t\cos\theta$.

Given that λ is the *Radio Frequency* source wavelength, the phase change due to the δl displacement is given by:

$$\delta\phi = \frac{2\pi\delta l}{\lambda} = \frac{2\pi v\delta l}{\lambda}\cos\theta \tag{2.19}$$

and therefore, the frequency change, i.e. *Doppler Shift*, is given by:

$$f_d = \frac{1}{2\pi}\frac{\delta\phi}{\delta t} = \frac{v}{\lambda}\cos\theta \tag{2.20}$$

The paths that arrive at the receiver are, in general, multiple. Each of the multiple paths experiences the *Doppler Shift* (that can be negative or positive) in an independent manner. The example in Figure 2.8 is illustrative and just show one path and a

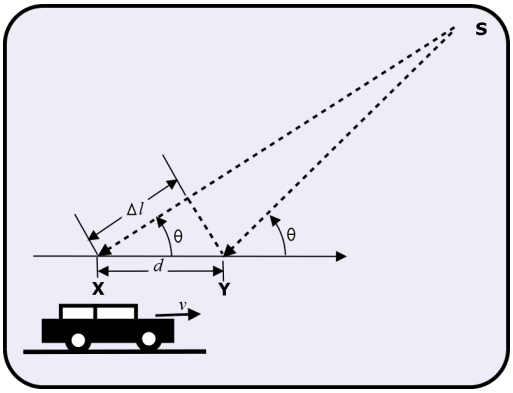


Figure 2.8: Doppler effect.

positive *Doppler Shift*, i.e. the receiver perceives the source frequency as being higher than transmitted. Contrarily, if the receiver perceives the signal with lower frequency, the *Doppler Shift* is said to be negative, and the receiver is moving away from the RF source.

2.2.1.7 Doppler Spread

Doppler Spread and *Coherence Time* are parameters which describe the time-varying nature of the channel in an small region. This channel variation occurs due to relative motion between transmitter and receivers and by the movement of objects within the channel [46]. Each path of the multipath channel can experiences a different *Doppler Shift*, which corresponds to different rates of changes in phase. *Doppler Spread* is a measure of the spectral broadening, caused by time changes of the mobile channel. According to [46], it is defined as the range of frequencies over the receiver *Doppler Spectrum* is essentially zero.

2.2.1.8 Coherence Time

According to [46] *Coherence Time* is the statistical measure of the time duration over which the channel impulse response is essentially invariant. *Coherence Time* is inversely proportional to the *Doppler Spread*.

Given two channel responses at different times, $h(t)$ and $h(t+\delta t)$, the time corre-

lation between the two channel responses is:

$$R(\delta t) = \int_{-\infty}^{+\infty} h(t)h^*(t+\delta t)dt \tag{2.21}$$

As in the case of *Coherence Bandwidth*, the *Coherence Time* T_c is defined as the time for which the correlation decreases 3 dB. For an exponentially decaying delay profile the and given *Doppler* frequency the *Coherence Time* is given by:

$$B_{coh} \approx \frac{1}{f_d}. \tag{2.22}$$

2.2.1.9 Types of Small-Scaling Fadings

Figure 2.9 shows the types of Small-Scaling Fadings. The type of fading experienced by the signal depends on signal parameters such as symbol period and bandwidth, and and channel parameters as *RMS Delay Spread* and *Doppler Spread*. Within Figure 2.9 it is possible to identify the type of fading experienced by he signal based on those parameters.

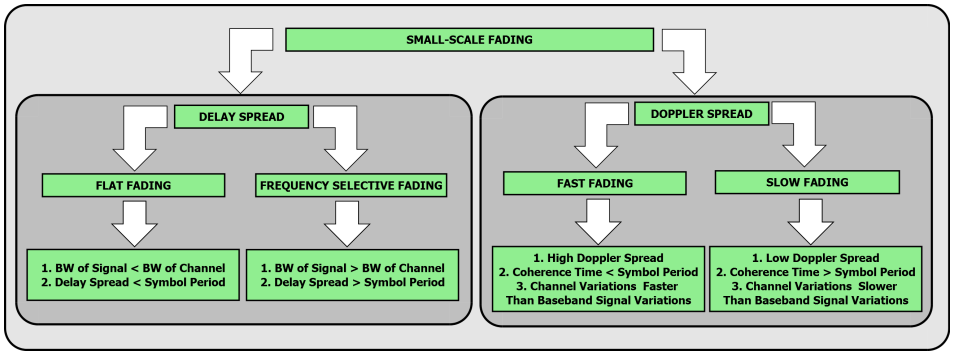


Figure 2.9: Classification of small-scaling fading channels.

2.2.1.9.1 Fadings Due to Multipath Time Delay Spread

Multipath delay spread cause time and frequency selectivity.

2.2.1.9.1.1 Flat Fading

This is the most common type of fading described in the literature [46]. In this kind of fading the wireless channel presents a constant gain and linear phase over a channel bandwidth that is larger than the signal bandwidth. Nevertheless, the strength of the received signal changes over time, due to multipath. Flat fading channels are also know as narrowband[1] channels. Typical values for deep fading within Flat Fading channels are in the order of 20 to 30 dB. A consequence of this is the need of special error correction protection. The statistical distribution of the amplitude of this type of channel can be Rayleigh, Rice or Nakagami.

2.2.1.9.1.2 Frequency Selective Fading

In this kind of channel the gain and phase are constant over a time that is smaller than the symbol period. This causes the frequency selectivity of the channel. This type of channel causes Inter-Symbol Interference. The term frequency selective comes from the fact that the frequency response of the channel is not flat, i.e. some frequency components have gains lower than others. This type of channels are known as wideband channels, due to the fact that the signal bandwidth is larger than the channel bandwidth. According to [46] a rule a thumb for this channel is: if it has a period 10 times larger than the symbol period, i.e. $T_s \leq 10\sigma_t$, the channel can be considered frequency selective. The effects of frequency selective fading channels are worst than flat fading channels, in terms of BER.

2.2.1.9.2 Fadings Due to Doppler Spread

Doppler Spread causes frequency dispersion and time selective fading. Recall that *Doppler Spread* is caused by the relative movements of receiver and transmitter and/or scatters. Depending on the channel impulse response changing rate, compared to baseband symbol variations, the channel can be classified as either *Fast Fading Channel* or *Slow Fading Channel*.

[1] This name comes from the fact that the channel bandwidth is narrower than the signal bandwidth.

2.2.1.9.2.1 Fast Varying Fading

When the *Coherence Time* of the channel is smaller than the baseband symbol period the channel is said to be *Fast Fading*. That is to say, the channel impulse response rapidly varies within the baseband symbol period. A consequence of the fast fading channel Doppler spreading is the frequency dispersion. This type of channel is also known as *Time Selective Channel*.

2.2.1.9.2.2 Slow Varying Fading

In opposition to the *Fast Fading Channel*, when the *Coherence Time* is larger than the baseband symbol period, the channel is said to be *Slow Fading*. In other words, the channel impulse response varies slowly compared to baseband symbol variations. The frequency spread due to slow fading is much smaller than the baseband symbol bandwidth.

2.2.1.10 Large-Scale Fading

Large-scale fading represents the average signal power attenuation or path loss due to motion over a large areas [52]. It can be considered a spatial average over a small-scale (small-scaling fading) fluctuations of the signal, as can be seen in Figure 2.10. It is evaluated averaging the signal over 10 to 30 wavelengths. Figure 2.11 shows the small-scale fading that is superimposed to large scale path loss. According to [46], the following are the three basic mechanisms that impact large-scale fading[1]:

- **Reflection**: happens when RF signals reaches a surface with dimensions larger than the RF wavelength(λ);
- **Diffraction**: occurs when the propagation path is obstructed by a large building causing a secondary wave behind the obstructing building;
- **Scattering**: occurs when the RF wave reaches a surface with dimension in the order of or less than λ, causing the energy of the incident wave to be scattered in all directions;

[1] Small-Scale Fading statistics are also affected by these 3 propagation mechanisms.

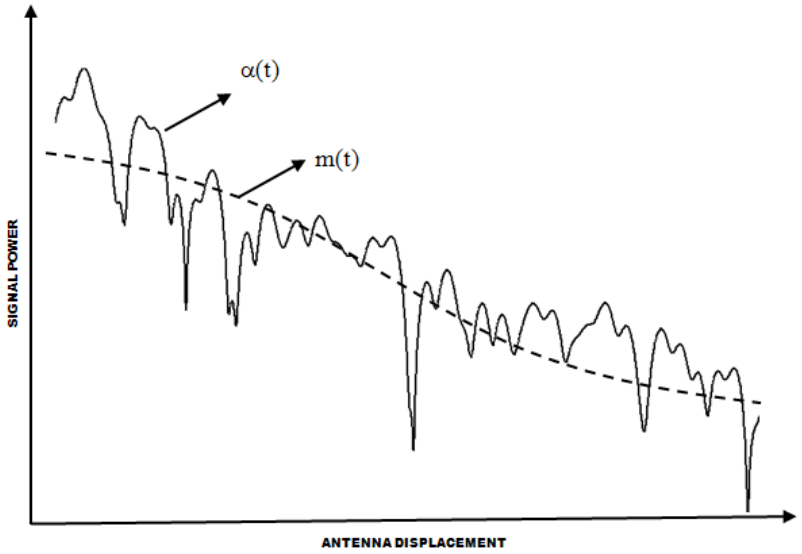


Figure 2.10: Small-scale fading superimposed on large-scale fading.

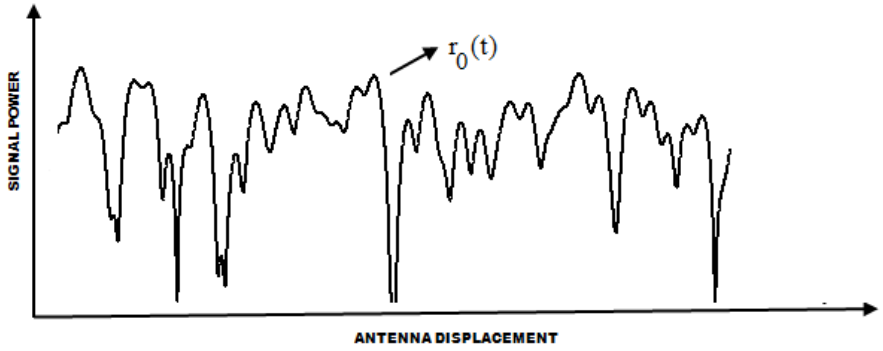


Figure 2.11: Small-scale fading regarding an small displacement (average constant power).

2.2.1.10.1 Shadowing

Large-scale fading represents the average signal power attenuation or the path loss due to motion over large areas [52]. It is affected by terrain contours and geography between the transmitter and receiver. The statistics of large-scale fading are used to estimate the path loss as a function of distance. Path losses due to shadowing are often described in terms of mean path loss (n-th power loss) and have Lognormal distribution variations over mean [52].

2.2.1.10.1.1 Log-Normal Distribution

Given a received signal r, the *Log-Normal Distribution* is computed as:

$$p(r) = \frac{1}{r\sigma\sqrt{2\pi}} e^{-\frac{(ln(r)-m_r)^2}{2\sigma^2}} \tag{2.23}$$

2.2.1.10.2 Path Loss

There are several Path Loss models. Nevertheless, by far, Okumura-Hata model is the most well-known. Okumura [53] made some of the earlier comprehensive path-loss measurements for a wide range antenna height and coverage range. Hata [54] converted the Okumura measurements into parametric formulas [52]. The Equation 2.24, 2.25 and 2.26 are the famous Okumura-Hata[1] propagation models for *Typical Urban*, *Typical Suburban* and *Rural* areas [55]:

2.2.1.10.2.1 Typical Urban

$$L_{50} = 69.55 + 26.16\log f_c + (44.9 + 6.55\log h_b)\log d - 13.82\log h_b - a(h_m) \quad \text{dB} \tag{2.24}$$

[1]Despite these equations have been derived by Hata, they were based on Okumura's work. Sometimes the are called Hata models, but they are known as Okumura-Hata models.

where,

$a(h_m)$ is the correction factor for mobile antenna height, given by:

$$a(h_m) = 8.29[\log(1.54h_m)]^2 - 1.1 \qquad f_c \leq 200 \quad MHz \quad \text{(for large cities)}$$

$$a(h_m) = 3.2[\log(11.75h_m)]^2 - 4.97 \qquad f_c \geq 400 \quad MHz \quad \text{(for large cities)}$$

$$a(h_m) = [1.1\log(f_c) - 0.7]h_m - [1.56\log(f_c) - 0.8] \quad \text{(for small and medium cities).}$$

2.2.1.10.2.2 Typical Suburban

$$L_{50} = L_{50}(\text{urban}) - 2\left[\left(\log\left(\frac{f_c}{28}\right)^2\right) - 5.4\right] \quad \text{dB} \tag{2.25}$$

2.2.1.10.2.3 Rural

$$L_{50} = L_{50}(\text{urban}) - 4.78\left(\log f_c\right)^2 + 18.33\log f_c - 40.94 \quad \text{dB} \tag{2.26}$$

where:

$$\begin{aligned}
f_c &= \text{carrier frequency (MHZ);} \\
d &= \text{distance between transmitter and receiver(km);} \\
h_b &= \text{trasmitter(base station) station antenna(height);} \\
h_m &= \text{mobile antenna height;} \\
150 &\leq f_c \leq 1500 \text{ MHz;} \\
30 &\leq h_b \leq 200 \text{ m;} \\
1 &\leq h_m \leq 10 \text{ m;} \\
1 &\leq d \leq 20 \text{ km.}
\end{aligned}$$

2.2.1.10.3 Small Scaling Fading

As shown in Figure 2.11, the RF signal presents average constant power over small displacements. In the vicinity of the transmitter antenna there exist less scatter and the signal has a strong Line-of-Sight component. In this case the constant average power signal has Rice distribution. As the receiver moves away from the transmitter antenna, the distribution becomes Nakagami. As the receiver moves even further, the power can be characterized by the Rayleigh distribution. There are other distributions that can characterize the signal like e.g. those presented in [56] and [57].

2.2.1.10.3.1 Rice Distribution

$$p(r) = \frac{r}{\sigma^2} e^{-\left(\frac{r^2+A^2}{2\sigma^2}\right)} I_0\left(\frac{Ar}{\sigma^2}\right) \quad \text{for} \quad A \geq 0 \quad \text{and,} \quad r \geq 0 \tag{2.27}$$

Given that, A is the peak amplitude of the dominant signal, $I_0(...)$ is the modified Bessel Function of the first kind and zero order. $\frac{r^2}{2}$ is the instantaneous power and σ is the standard variation of the local power. The *Rician* distribution is often described by the parameter K, known as *Rice Factor*, given that:

$$K = 10Log\frac{A^2}{2\sigma^2} \quad dB \tag{2.28}$$

2.2.1.10.3.2 Rayleigh Distribution

$$p(r) = \frac{r}{\sigma^2} e- \left(\frac{r^2}{2\sigma^2}\right) \quad \text{for} \quad 0 \leq r \leq \infty \tag{2.29}$$

Where, σ is the *rms* value of the received signal and σ^2 the local average power of the receiver signal(before envelope detection).

2.2.1.10.3.3 Nakagami Distribution

$$p(r) = \frac{2}{\Gamma(m)} \left(\frac{m}{\Omega}\right)^m r^{2m-1} e^{\left(-\frac{m}{\Omega} r^2\right)} \quad \text{for} \quad M \geq \frac{1}{2} \tag{2.30}$$

Where

$$\Omega = \frac{\sum_{i=1}^{N} r_i^2}{N}$$

$$m = \frac{4.4}{\sqrt{u_2}} + \frac{17.4}{1.29 u_2}$$

$$u = 20log(r_i), \qquad u_1 = \frac{\sum_{i=1}^{N} u_i}{N}, \qquad u_2 = \frac{1}{N} \sum_{i=1}^{N} (u_i - u_1)^2$$

and

$\Gamma(m)$ is the gamma function.

2.2.1.10.3.4 Other Distributions

This section presents two distribution examples that appear in the literature to represent Small-Scaling fading statistics, namely $n-\mu$ and $\alpha-\mu$, presented in [56] and [57], respectively. In [56], the author claims that $n-\mu$ is a general fading distribution, that includes the one-sided Gaussian, the Rayleigh, and the Nakagami-m distributions as special cases. According to the author, Rice and Log-normal distributions may also be well-approximated by the $n-\mu$ Distribution. For more details on that distribution, refer to [56]. Within [57] the authors also present a distribution that can model Small-Scale fading, claiming the advantage of representing several distribution using a single distribution.

2.2.1.11 Reception Scenarios for Wireless Channels

As described earlier, in this chapter, the signal propagated between a transmitter and a receiver undergoes several phenomena caused by the geography, geometry and relative speed of the channel elements. Another element that causes variations in the received signal are the receiver's localization regarding the constructions, i.e. indoor or outdoor, or whether the receiver is inside or outside a vehicle. For this reason I present in Figure 2.12, a summary of the possible scenarios a receiver can face.

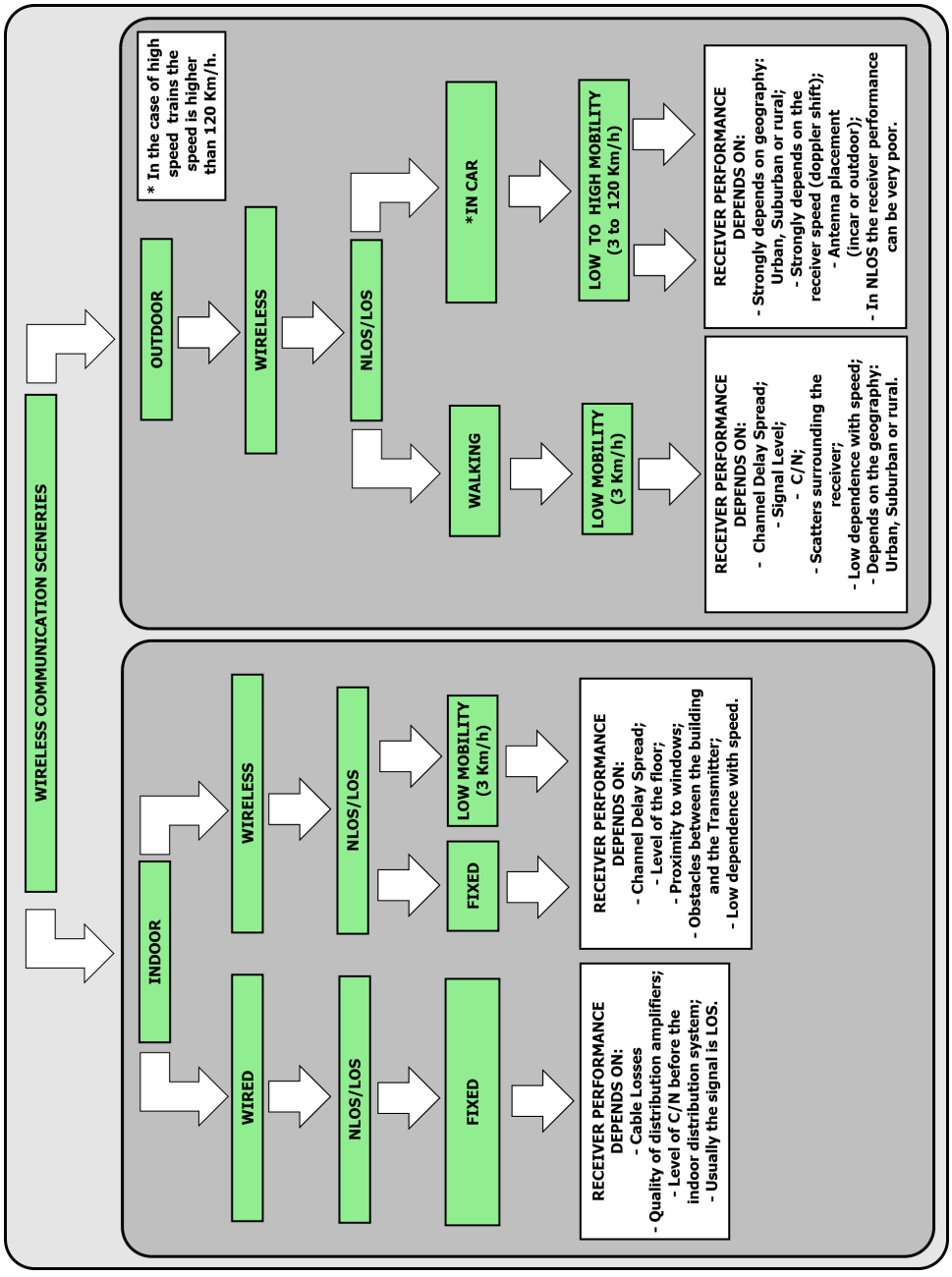


Figure 2.12: Reception scenarios for wireless channels.

2.2.2 Main RF Analog Front-End Impairments

Anything that causes signal distortion and quality degradation of the transmitted information in a communication system can be considered an impairment. Wireless channel fading, AWGN noise, non-linearities of amplifiers, frequency errors, sampling timing errors, I&Q imbalance, phase noise and DC offset are examples of impairments of communication systems. Figure 2.1 shows the sources of impairments in a wireless communication systems. As can be observed, the impairments can be located at the transmitter, the channel and the receiver. They can be radio impairments, hardware impairments and channel impairments. Figure 2.1 also shows that a given impairment can have different sources, e.g. frequency errors can be generated at transmitter and receiver oscillators.

A direct consequence of the impairment is the increase in the BER, which is a well known figure-of-merit in digital communications systems. Depending on the level of the impairment, the degradation can be so high, that the system cannot synchronize. Following the modeling of the communication system, impairment modeling is the most important task, specially when designing receivers. If an important[1] impairment is neglected during the modeling, and the system is implemented in hardware (e.g. in VLSI) without considering it, the consequences can be catastrophic. This can drive the receiver to a irreducible BER floor, or maybe the receiver will not work in some wireless radio scenarios[2]. Therefore the correct definition of impairments and their simulation is of paramount importance when designing wireless communication systems targeting real world applications.

Contemporary high data rate wireless systems (e.g. ISDB-T) are very sensitive to the RF Analog Front-End impairments, which are difficult to avoid in low-cost low-power transceivers integrated in deep sub-micrometer CMOS technology [58]. Due to that fact, a new trend in communications systems is to accept some RF Analog Front-End remaining impairments [59], that need to be compensated in digital domain using advanced signal processing techniques[60]. It is necessary to highlight at this point that not necessarily all impairments presented in this section will affect significantly the digital receiver performance due to the fact that some RF Analog Front-End have better performance than others. Therefore, it is important for digital designers to have a clear idea, or well-defined specifications, of the type of impairments the chosen commercial RF Front-Ends generate, in order to specify good digital receiver architectures to overcome those impairments, avoiding non-necessary signal processing, and consequently saving area and power. Within this section the main radio and hardware impairments and their origins are presented.

[1] Certain impairments are so small that do not need to be considered when simulating the system performance, e.g. a phase noise which is quite below the system noise.

[2] A simple example of this is not to simulate receiver behavior under speed conditions for a hardware that aims to work over a broad range of speeds.

2.2.2.1 Radio Conversion Architectures Impairments

Despite the main goal of this thesis being to propose architectures for a Digital Receivers[1], it is necessary to pay special attention to RF-to-IF or Baseband conversion, which shall be made by a silicon tuner, called *RF* or *Analog Front-End.* First, because as can be seen in Figure 2.1 the *Analog Front-End* is responsible for several impairments the digital receiver shall overcome, and second because it is assumed that the *Digital Receiver* shall work with different commercial silicon tuners. Therefore, the performance of this receiver can be strongly affected by the architecture characteristics of those elements (see Table 2.1). Even if the receiver does not have dedicated algorithms to eliminate some silicon tuners-generated impairments, the practitioners that will integrate and test the *Digital Receiver* shall understand the impairments generated by the tuners architecture in order to choose the one that best fits the *Digital Receiver* characteristics in order to minimize the BER. Table 2.1, reproduced from [61], presents a good overview of the main types of conversion and architectures adopted in analog front-ends and tuners, as well as the advantages and drawbacks when adopting each of the architectures. If the System Architecture has not sure about to adopt an specific tuner or has plans to change it during the project (either due to technical or commercial reasons), the knowledge of the drawbacks and advantages of all possible radio conversion architectures is necessary to define an architecture for the digital receiver that support the major RF front-end impairments, in the case of tuner changing. Of course some of these impairments can be simulated in order to check their influences in the *Digital Receiver* performance. For detailed discussions on the RF front-end regarding architectures, topologies and advanced architectures for Software-Defined-Radio dedicated conversion refer to [62],[63], [61] and [64].

Table 2.1 shows the several drawbacks and impairments that the main analog front-end architectures present. Nevertheless, in general, *Digital Receivers* only use signal processing to tackle, I/Q mismatch, DC offset, frequency and phase errors and adjacent interferences. When adjacent channel interference is present and it is not very close to the main signal, it is possible to eliminate it or significantly reduce it using simple FIR bandpass (for IF signal) or lowpass filtering (for Zero-IF signal).

2.2.2.2 A Second Look at RF Analog Front-End and Wireless Chanel Impairments

Within this chapter, and up to this moment, I presented two main sources of impairments that need to be handled by any generic wireless digital receiver. The first is the wireless channel and the second is the RF Analog Front-End Architecture. Within this section, I describe other sources of impairments and review those presented be-

[1] The proposed receivers work in IF or Zero-IF modes.

Table 2.1: Main RF Front-End Receiver Architectures.

ARCHITECTURE	ADVANTAGES	MAJOR DRAWBACKS
SUPER-HETERODYNE	- Selectivity	- Image Frequency and I/Q Mismatch
	- Sensitivity	- High quality discrete components
	- Immune to DC offset	- Perfect LNA load (50 Ω load)
		- Complexity
		- Noise Figure
		- Nonlinear behavior in components
ZERO-IF	- Simplicity	- Strong DC problems
	- IC Integration	- I/Q Mismatch
		- Even/Odd Distortion
		- Flicker Noise
LOW-IF	- No DC problems	- I/Q Mismatch
	- Simplicity	- Image Frequency
	- Less high quality discrete components	- Requires high performance ADC
BAND-PASS SAMPLING	- Flexibility	- Susceptible to clock aperture jitter
	- Signal Manipulation	- Noise Figure degradation
	- Low cost silicon area	- Aperture Distortion
	- Minimize DC and RF issues in digital domain	- Power Consumption
SIMPLE DETECTOR	- Simplicity	- Huge degradation with interferes
	- Low Cost	- Low Selectivity and Sensitivity
		- Some DC issues
HARTLEY	- Good IRR	- I/Q mismatch
	- Less Discrete Components	- Shift-by-90^o block and adder
	- Reduce load problems	- Variation of R and C in RC-CR Network
		- Increase the number of components
WEAVER	- Similar to Hartley	- Huge number of mixers
	- Avoid RC-CR Network	- I/Q mismatch
		- Dependent VCO
		- Strong adjacent channel interferes
		- Increased number of components

forehand, showing the sources and explaining in details the possible[1] consequences to the receiver when living with such impairments. As can be seen in Figure 2.1, the impairments can be generated at the transmitter, receiver and wireless channels. Besides some impairments (e.g. frequency errors), have different sources. In the case of frequency errors, the sources are the oscillators, located at the transmitter and receiver, and the speed of the device (i.e. Doppler-shift). At the transmitter and receiver, this error can be generated in the digital and analog domains, depending on the transmitter and receiver architectures. For instance, a transmitter that has a digital IF output and an analog up-conversion, has digital and analog generated frequency errors[2]. The same happens at the receiver, i.e. an IF digital receiver has oscillator errors at the analog RF-to-IF conversion and at digital IF-to-Baseband conversion. In the case of direct conversion, i.e. baseband-to-RF and RF-to-baseband at the transmitter and receiver respectively, the only source of frequency errors in hardware are the Analog oscillators at both ends. As mentioned previously, not necessarily all impairments described in this section affect the ISDB-T digital receiver proposed in this work, and part of them can be simply ignored. This is going to be shown within Chapter 5. Last but not least, it is important to recall that some of the impairments detailed in this section are generated at the transmitter, but regardless the origin those impairments affect the system performance, which is usually[3] measured at the receiver, for instance by means of the BER.

2.2.2.2.1 Non-Linearity

Power Amplifiers working on the non-linear region above the 1 dB compression point will start generating intermodulation (IMD) or harmonics. When using single tones the high order harmonics (i.e. 2nd, 3rd, etc...) are out of the signal band and can be eliminated using filtering. Nevertheless, as the band of the desired signal increases, the non-linearities causes IMD (specially due to second order harmonics [58]), which appears as interference within the band of the signal. This interference can not be eliminated from the received signal. Several methods are proposed in the literature to avoid the transmitted signal driving the power amplifier to the non-linear region and by consequence to avoid such IMD and out of band interferences(refer to [65] in [66]). The proposed methods can be used to minimize the effects of the distortion

[1] I say "'possible"', because some receivers are more susceptible to certain impairments than others.

[2] Assuming that the clock source for the digital and analog domains is the same, i.e. both has the same crystal, an obvious reason for the errors is the crystal frequency error caused by the *ppm* variation. A second and not obvious reason affects the digital domain, that is the finite precision of the NCO accumulator. Even in the case of both domains have two independent clock sources, the generators of reference frequency always have *ppm* variation.

[3] Some impairments like non-linear distortion originated at transmitter power amplifier sometimes are handled by pre-distortion techniques at the transmitter, i.e. they are measured and compensated at the transmitter end.

in amplifiers or to allow the use of other less efficient amplifiers. Non-linearities can cause significant performance degradation. Therefore system designers must define clearly the IP2 and IP3 amplifiers Intercept Points in order to minimize its effects. On the other hand, for the digital-oriented design, it is mandatory to choose the adequate RF Front-End to minimize non-linearities effects in the receiver performance.

2.2.2.2.2 Power Amplifier Distortion

Power Amplifiers (PAs) are used to provide gain to a given signal to be transmitted over a channel. It is well known that wireless channels introduces high losses in the transmitted signal. Therefore, the goal of power amplification together with the antenna system is to provide enough power to the transmitted signal in order to deliver the signal to the receiver over the largest possible distance. The gain that the combination of power amplifier and antenna can provide to the transmitted signal is limited mainly by local regulations and costs. It is worth to mention that power amplifiers devoted to DTV broadcast can provide output power in the order of several kW. On the other hand, a simple bluetooth power amplifier provides power in the order of few mW. PAs, like any other amplifier, exhibit nonlinear behavior and saturating gain at large input levels. Two consequences of amplifying a signal that drives the amplifier to non-linear region (up to the 1 dB compression point) are the degradation of the Error Vector Magnitude (EVM)[1] metric and the spectral regrowth [67] [68], i.e. increasing the spectrum in the neighborhood of the wanted signal, causing interference in the adjacent channels.

OFDM signal has high PAPR, when compared to constant envelop modulations. In order to avoid the Power Amplifier (PA) to be driven into nonlinear region, due to OFDM PAPR, it must be a linear amplifier (Class A) with the largest possible dynamic rage, i.e the PA must have the largest possible *Input Back-Off* (IBO)[2]. Nevertheless, PA of Class A have high cost when compared to other classes of amplifiers. This can make its use prohibitive. Power amplifier distortion can be modeled by AM/AM (amplitude modulation/amplitude modulation), which models amplitude distortion, and AM/PM (amplitude modulation/phase modulation) curves, which models the phase shift [58].

2.2.2.2.3 Phase Noise

Phase Noise is a random perturbation in the sinusoidal wave phase. It is usually

[1] In short, EVM is the receive constellation error regarding the original constellation.
[2] IBO is the ratio between the 1 dB compression point input power and the signal average power.

described in the frequency domain as the noise spectrum centered at the oscillation frequency [69]. On the other hand it can be described in the time domain by measuring the oscillator jitter, i.e. how precise the sinusoid periodicity is. *Phase Noise* occurs due to the fact that a pure sinusoid (i.e. a pure impulsive spectra) can not be generated by oscillators. It occurs in the transmitter and the receiver, in analog and digital domains. It can occur in the digital receiver/transmitter because, depending on the chosen architectures, sinusoidal waves must be generated at the digital domain to create an IF oscillator.

In the RF Analog Front-End, when *Local Oscillator* (LO) is used in the receiver down-converting mixer, the resulting baseband signal spectrum will be the passband signal spectrum convolved with the LO spectrum with non-zero bandwidth [69]. Similar convolution occurs in the transmitter analog oscillator and also at the digital oscillators mixers that convert the signal to IF (at the transmitter) or baseband (at the receiver).

Phase Noise causes two effects in the complex signal: rotation and noise-like blurring [33]. The first is known as *Common Phase Error* (CPE) and the second *Inter-Carrier Interference* (ICI). Nevertheless, the level of (that can be seen as a kind of noise) caused by *Phase Noise* can be far below the main signal not causing any significant degradation in the system performance, especially for systems that work in high SNR. In the case of OFDM with GI the CPE can be easily handled by the one-tap equalizer.

2.2.2.2.4 Carrier Frequency Offset

Frequency errors can occur at any sinusoidal oscillator in wireless systems, but it effectively appears in the receiver after down conversion of the signal from RF to baseband (as a matter of simplicity, I also consider in the RF to baseband conversion the conversion from IF to baseband, if used) process. *Carrier Frequency Offset* (CFO) can also occur due to Doppler shift. To generate sinusoidal oscillations, commercial crystals are used as reference. As a general rule, due to scale issues, crystals used in consumer electronics devices need to be cheap. A consequence of that is the high p.p.m presented by those crystals. Therefore a significant part of the frequency mismatches is caused by the error in the reference crystal. Nevertheless, there are other significant sources of frequency errors like in satellite receivers, which have Low Noise Block (LNB) down converter with very poor oscillator performance, in certain cases the oscillator error can be in the order of several hundreds of *ppm*. On the other hand DTV Set-Top-Boxes (STBs) in general use crystals with less than 100 *ppm*. The main consequences of CFO in OFDM systems are the occurrence of an integer[1] offset from the ideal frequency and a fractional error (smaller than subcarrier spacing) which

[1]The term integer is related to an integer offset in relation to the sucabarrier spacing.

causes *Inter Carrier Interference* (ICI) and phase rotation.

2.2.2.2.5 Sampling Clock Offset

Due to the adoption of high p.p.m. crystals, the *Sampling Frequency* (also known as *Sampling Clock*) used at the receiver and transmitter presents an offset regarding the desired ideal *Sampling Frequency*, known as *Sampling Clock Offset* (SCO). The *Sampling Clock* is used at DAC (transmitters) and ADC (receivers). In general, the high-end transmitters as those used to broadcast DTV signals, use very low p.p.m crystals, which cause a very low offset in the DAC SCO. Nevertheless, for consumer-oriented transceivers as those used in WLAN and cell phones, DAC and ADC[1] present high SCO due to the use of low-cost-high-ppm crystals. The SCO can be positive or negative, and a first obvious consequence of this is the signal band increase and decrease, this phenomena causes ICI in OFDM-based systems. Another less obvious consequence is a linear phase rotation that increases with the symbol index. This happens due to the sample drift in relation to the ideal sampling point. Another effect of the sample drift is the variation in the ideal sample used as reference to discard the GI just before FFT. In other words, if the sampling drift is towards the GI the ideal FFT sampling point is affected and after a certain amount of time it will make the equalization harder. If this drift is towards the next OFDM symbol, it can causes *Inter Symbol Interference* (ISI).

2.2.2.2.6 Timing Synchronization Errors

In short, *Timing Synchronization Error* occurs due to the fact that after initialization, the receiver starts sampling the received signal at a random position in relation to the ideal timing. The ideal initial sampling point (ideal timing) is the first sample of any OFDM symbol. Recalling that an OFDM Symbol is made of GI + IFFT and its first sample is the first sample of the GI, it is easy to realize that this reference point is used (among other things) to compute the right GI samples to be discarded. As in the case of small drift caused by the SCO, the *Timing Synchronization Error* can be either towards the GI or next OFDM Symbol. The consequence is the same as for sampling drifts: ISI (towards next OFDM Symbol) or phase rotation when the timing error is towards GI. The later causes phase rotation and depending on the number of samples it can simply render adequate equalization impossible.

[1] In the case of Set-Top-Boxes, for terrestrial DTV, only ADC are used.

2.2.2.2.7 I&Q Imbalance or I&Q Mismatch

Most of the RF Analog Front-End architectures have I&Q mismatch (see Table 2.1). I&Q mismatch is generated during the mixing process used to up-convert and down-convert the baseband signal, in the transmitter and receiver respectively. To fully employ the bandwidth of the signal it is preferable to use quadrature mixers, that operates the signal in complex domain [58]. By using quadrature mixers it is possible to distinguish between positive and negative frequencies of a baseband signal centered around DC frequency. Nevertheless, real quadrature mixers implementation are impaired in gain and phase, which causes rotation and amplitude distortion. For Zero-IF and Low-IF architectures the effects of I&Q mismatches are different. In the case of a transmitter using Zero-IF architecture, it causes baseband co-channel interference, and in the Low-IF architectures case, it generates out-of-band emissions which cause adjacent interferences. At the receiver end the effects are similar. Nevertheless, for Low-IF architecture the I&Q mismatch results in an aliasing of an adjacent channel into the channel bandwidth[58], which limits the receiver SNR.

2.2.2.2.8 DC Offset

DC offset is an RF Analog Front-End dependent impairment. Table 2.1 shows that, among the main RF Analog Front-Ends the Zero-IF architecture is the only one which presents DC offset. A process known as self-mixing, i.e. the mixing of the LO with itself, causes the appearance of DC offset at the output signal. This occurs due to the finite isolation typical of silicon-based ICs between the LO and RF ports of a mixer [70]. A simple way to solve this issue at RF Analog Front-End would be the adoption of high pass filtering. Nevertheless, several signals have significant content at DC frequency, e.g. DTV uses DC carrier to carry valid data, and would be affected by this approach. Three other methods to fix it at RF Analog Front-End are presented in [70] and references therein:

- Sub or super-harmonic mixing with an LO at some integer multiple or fraction of the RF;
- DC-free coding at the transmitter: by removing the signal content at DC;
- Offset calibration techniques using digital sample-and-hold feedback, offset estimation with corrective feedback at baseband, and servo loops.

Even tough, if the system architect choose the RF front-end with the architecture most immune to DC offset, some residual DC offset can still occur and need to be handled at the baseband digital receiver. In the case of significant DC-offset, in order

to preserve the system's performance, the adoption of an ADC with higher number of bits (when compared to that number for a signal without DC offset) is necessary.

2.2.2.2.9 Additive Noise

Electronic and thermal noises are inherent in wireless channels and analog circuits in the Analog RF Front-End. Flicker noise and shot noise are also major sources of noises in electronic devices [69]. Despite being independent phenomena, they are modeled as being *Additive White Gaussian Noise* (AWGN). The receiver noise floor and the limitations in the possible transmitted power creates a limit in the achievable receiver SNR. As a general rule, digital wireless systems are designed to work in specific SNR for a target BER. But, of course, they can work on higher SNR with better performance than the target. The system design is done in such a way that below that SNR the system BER is not acceptable (the system can work but its performance will be poor). White noise is a kind of noise with flat spectrum over $-\infty$ to $+\infty$. Nevertheless, this would lead to an infinite power. In practice, any noise with a flat spectrum in the band of interest is called white [71]. In short, AWGN cannot be removed from the signal, and FEC is necessary to minimize its effects.

There is another kind of noise that significantly affects the OFDM-based systems, it is the *Impulsive Noise*. This kind of noise can be modeled as *Laplacian Noise* [72]. A receiver suffering *Impulsive Noise* interference can have its performance seriously degraded. *Impulsive Noise* consists of short bursts of noise with high power spectral density and random occurrence [73]. It is caused by electromagnetic sources like distribution power lines and microwave ovens, computational platforms such as notebook due to clocks, buses activity and processors for instance. There are other sources of Impulsive noise, but I am going to limit the examples to these. In OFDM systems, specific signal processing is necessary to minimize the effects of impulsive noise, like the one proposed in [73] and other techniques there mentioned. Impulsive Noise effects can be also mitigated or reduced by using time interleaving as in ISDB-T. Nevertheless, the cost of such a solution in terms of area, due to use of large memory, can be prohibitive.

2.2.2.2.9.1 Noise Figure and Sensitivity

According to Friis [74], *Noise Figure* (NF) is defined as the ratio between the SNR at the input and the SNR at output of a device. Analog RF Front-End of receivers are usually made of several cascaded blocks (e.g. several amplification stages). The overall NF of such a Front-End can be computed by the Friis Equation:

$$F_{RX} = F_1 + \frac{F_2 - 1}{G_1} + \frac{F_3 - 1}{G_1 G_2} \cdots + \frac{F_N - 1}{G_1 G_2 \cdots G_{N-1}} \tag{2.31}$$

In 2.31, F_1 and G_1 are the stage 1 NF and gain respectively, and F_N and G_N are the NF and gain for the *Nth* stage. As can be observed from that equation, the NF of stage 1 shall be as low as possible. This justifies the use of the so called Low Noise Amplifiers (or Very Low Noise Amplifiers) as input stage of the RF Analog Front-End of any receiver. A consequence of NF is the limitation of the receiver sensitivity, that is the minimum input power that leads the receiver to a reliable reception for a given target SNR. The sensitivity of a given receiver that has a target SNR of SNR_{targ} is given by [58]:

$$Sens = F_{RX} \times kTB \times SNR_{targ}. \tag{2.32}$$

Where k is the *Boltzman Constant*, T is the temperature in *Kelvin*, B is the band of interest and F_{RX} is the overall RF Analog Front-End NF. The same equation expressed in dBm is:

$$Sens_{dBm} = 10Log_{10}(kT) + 10Log_{10}(F_{RX}) + 10Log_{10}(B) + 10Log_{10}(SNR_{targ}). \tag{2.33}$$

2.2.2.2.10 Multipath

Multipath effects were widely described within section 2.2.

2.2.2.2.11 Quantization Noise and Limited Bit Length

Quantization errors occur at first due to DA/AD conversion. In addition, within the digital end, at both transmitters and receivers, several adjustment in the bit length are done, in order to reduce the design area and power consumption as well. Quantization causes an error which is often treated as noise. According to [75], this assumption is usually reasonable, but care is required: 1) its distribution is uniform, not Gaussian and 2) in the frequency domain, it can become very colored if the signal frequency and sampling frequency are related.

In general, the way to evaluate the effects of the limited bit length is through the BER measurements at the receiver. Another way to evaluate the quantization effects is by observing its effect in system EVM.

2.2.2.2.12 Quantization Clipping (Saturation)

It is well known that OFDM-based systems have large PAPR when compared to constant envelope modulations. In addition, the attenuation of transmitted wireless signal make the receiver *Automatic Gain Control* (AGC) vary the amplitude of the signal at ADC input. The incorrect choice of the ADC (i.e choosing an ADC than does not support the entire dynamic range of the OFDM signal) and an imperfect AGC can make the signal be clipped at ADC. The bad election of the DAC also produces signal clipping . Clipping is a non-linear effect and can produce severe harmonics and spurs, that post processing cannot reverse [75]. At the transmitter, the correct choice of DAC can minimize clipping effects. On the other hand, to reduce the clipping at the receiver it is necessary to guarantee a correct AGC behavior and adequate ADC number of bits.

Chapter 3

ISDB-T and DVB-S2 Standards

Most of the algorithms, proposed in this thesis for the ISDB-T and DVB-S2 receivers rely on the frame and pilot structure of the respective standard. Therefore, in order to better understand the proposed architectures and algorithms a good understanding on both standards is the necessary. Part of the information described within this Chapter, related to frame and pilot structure of both standards, could be placed along with the algorithms descriptions in Chapters 5 and 7. Nevertheless, this would make those chapters very long and boring to be read, which could cause misunderstandings and/or misinterpretations. The reader can, optionally, skip this Chapter and try to read Chapters 5 and 7 without the foundations presented here. Nevertheless, I suggest a first read of this Chapter and later, during the reading of those chapters, the reader can always use this Chapter as reference. On the other hand, standards are always full of details which demand plenty of attention, to avoid misinterpretation of the specifications. Furthermore, it is not good practice to skip details when doing the kind of projects presented in this thesis. Misinterpretation can be catastrophic when designing a ASIC and can cause the need of a re-design. First-design-success (especially for designs with large gate count such as those presented in this thesis) is almost an obligation for the small and early stage Design Houses, and emerging R&D Centers such Idea! Electronic Systems and Eldorado R&D Center, as they work with limited budget and very high results expectations.

3.1 ISDB-T Standard Basics

3.1.1 ISDB-T a First Look

The ISDB-T baseband transmitter physical layer standard [1] is based on OFDM transmission technology. It is inspired in the European standard DVB-T. Figure 3.1 presents the baseband transmitter according to the standard and Table 3.1 shows its major system parameters.

The input of ISDB-T transmitter consists of one or more MPEG-TS streams, that are re-multiplexed to form a single stream to pass through the RS encoding process. Depending on the services, the stream can be split from 1 to 3 layers namely A, B and C. Each one of these layers contains a number of segments (each segment is one OFDM block), the minimum number of segments per layer is 1.

The total of segments that an ISDB-T channel transports is 13. Each layer has its own convolutional code rate. The signals within the layers pass through a bit energy dispersal, followed by a byte-interleaver. After convolutional encoding, the signals within each layer are bit-interleaved and mapped to D-QSPK, QPSK, 16-QAM or 64 QAM. Next, the layers are combined into 1 single signal, which is time-interleaved to render the coding process more robust, especially for mobile reception conditions. Subsequently, the signal is frequency-interleaved. Then, the frame structure of an OFDM symbol is produced by inserting pilots, Auxiliary Channel (AC), and control information (Transmission and Multiplex Configuration Control - TMCC), among the data subcarriers according to ISDB-T frame specification. Finally, IFFT is performed, followed by the Guard Interval (GI) insertion.

The basic IFFT sampling rate is 512/63 (8.12698....) MHz, and the IFFT length can be 2048, 4096 and 8192, also known as Mode 1, 2 and 3 respectively. Mode 1 is less susceptible to the Doppler effects than modes 2 and 3, due to fact that it has the largest distance between subcarriers. Doppler effect induces ICI (Inter Carrier Interference) that increases the BER significantly. On the other hand, Mode 3 is less susceptible to the effects of large delay spreads, than Modes 1 and 2. This makes Mode 3 become the choice for a large number of broadcasters that targets fixed-reception. It is worth to mention that, to deal with delay spreads larger than GI, adaptive time equalization is required in order to reduce the delay spread of the channel. Besides, to overcome the ICI, extra signal processing on time or frequency domain are paramount. Adopting these solutions can cause a significant increase in the receiver complexity, and consequently in the ASIC area and cost.

ISDB-T has a special configuration called 1-SEG (one segment). This consists of a single segment transmitted at the center of the baseband channel (layer A), targeting handheld and mobile devices with low screen resolution and high mobility (e.g. vehicles and trains). This single segment is surrounded by the remaining 12

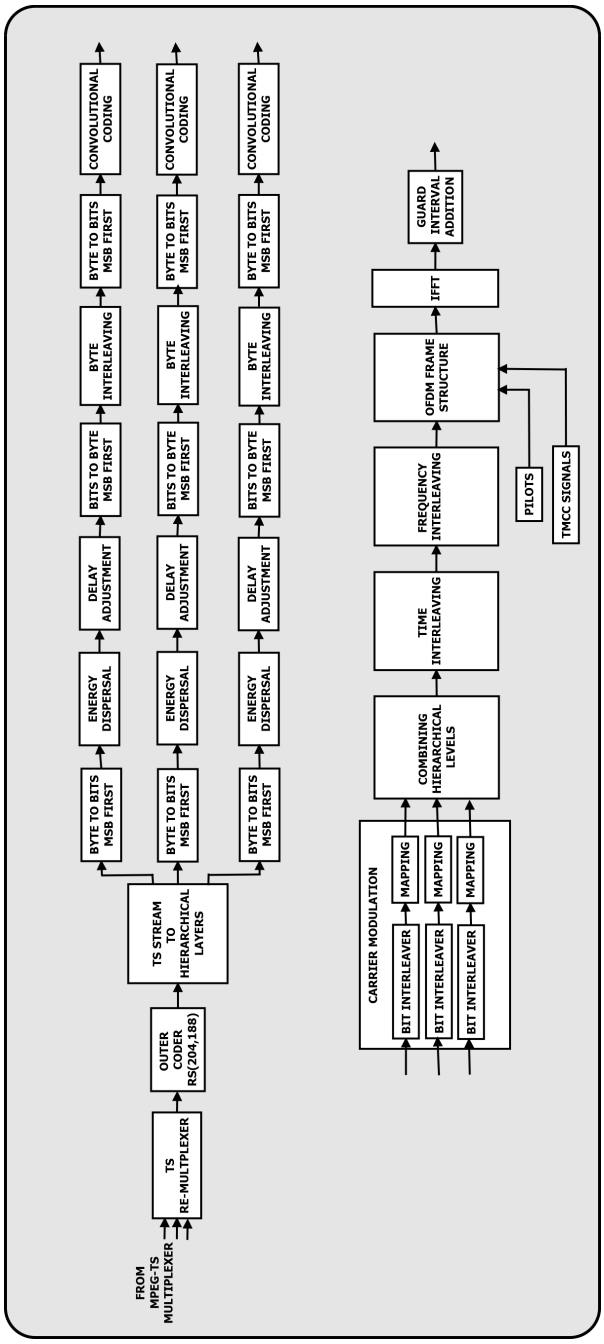


Figure 3.1: The ISDB-T transmitter, according to the standard [1].

Table 3.1: Main ISDB-T Transmission Signal Parameters.

MODE	1	2	3
Number of OFDM Segments		13	
Bandwidth	5.575 (kHz)	5.573 (kHz)	5.572 (kHz)
Spacing Between Carriers Frequencies	3.968 (kHz)	1.984 (kHz)	0.992 (kHz)
Symbols per Frame		204	
Effective Symbol Length	252 us	504 us	1008 us
	63 us (1/4)	126 us (1/4)	252 us (1/4)
Guard	31.5 us (1/8)	63 us (1/8)	126 us (1/8)
Length	15.75 us (1/16)	31.5 us(1/16)	63 us (1/16)
	7.875 us (1/32)	15.75 us (1/32)	31.5 us (1/32)
	64.26 ms (1/4)	128.52 ms (1/4)	257.04 ms (1/4)
Frame	57.834 ms (1/8)	115.668 ms (1/8)	231.336 ms (1/8)
Length	54.621 ms (1/16)	109.242 ms (1/16)	218.484 ms (1/16)
	53.0145 ms (1/32)	106.029 ms (1/32)	212.058 ms (1/32)
Inner Code	Convolutional Code (1/2, 2/3, 3/4 , 5/6, 7/8)		
Outer Code	Reed-Solomon(204,188)		
IFFT Length	2048	4096	8192
IFFT Sampling Frequency	512/63 MHz = 8,1269841 MHz		
Time Interleaving Length	0, 2, 8, 16	0, 2, 4, 8	0, 1, 2, 4

segments that can belong to a second layer (i.e. B) or to other two layers (B and C), each one with a number of segments. The total of segments to be transmitted in the ISDB-T channel is always 13. Figure 3.2 shows an example of hierarchic baseband configuration for the ISDB-T transmitter, targeting 3 services: layer A targets low screen resolution devices with moderate to high mobility, layer B targets handheld devices with larger resolution with moderate mobility and layer C targets fixed-reception with high definition decoding capabilities.

The minimum bit rate possible to be transmitted by an ISDB-T layer is 280.85 kbps (l layer with 1 segment, Code Rate 1/2, QPSK or DQPSK and GI = 1/4) and the maximum is 23.234 Mbps (1 layer with 13 segments, Code Rate 7/8, 64-QAM and GI=1/32). For more details on the physical layer parameters and configuration we refer the reader to the next section or to [1] and [76].

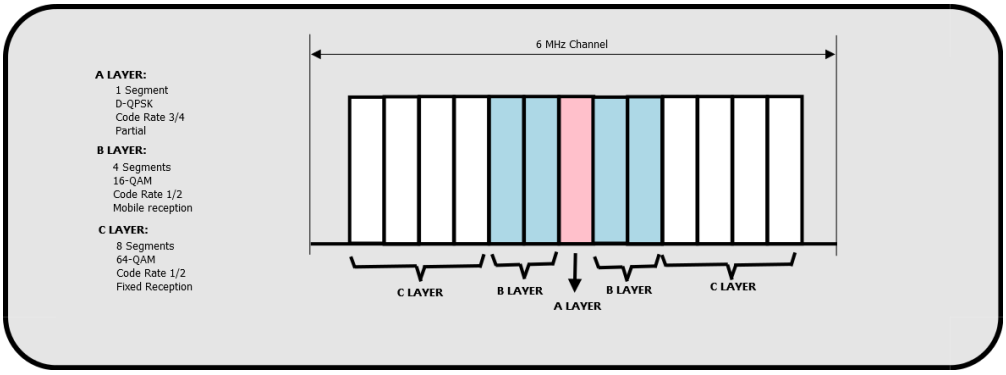


Figure 3.2: ISDBT hierarchy example

3.1.2 ISDB-T a Detailed View

Within the previous Section a first look at that ISDB-T system was given. Nevertheless, as this work is related to the implementation of a receiver for the ISDB-T standard, in this Section a detailed view on the blocks of the ISDB-T transmitter [1], shown in Figure 3.1 is provided.

3.1.2.1 Transport Stream Re-multiplexer (Remux)

The ISDB-T *Re-multiplexer* can be either an independent equipment or incorporated in the ISDB-T transmitter. It does multiplex the MPEG transport streams from independent MPEG encoders (e.g. 3 encoders, each one targeting one layer - A, B

or C) to form a single stream of 204 bytes packets. It also inserts transmitter configuration, control, synchronization - for Single Frequency Network (SFN) networks - information packets, and null packets. It is worth to note that null packets generated at the Remux are not transmitted over the air.

3.1.2.2 Reed Solomon Encoder

As in the case of the *DVB-T* standard, *ISDB-T* makes use of a RS(204,188,t=8) encoder, which is a shortened version of RS(255,239,t=8) code, with the same *Primitive Polynomial* (see Equations 3.2 and B.1), and *Field Generator Polynomial* (see Equations 3.1 and B.2). The shortened version is obtained by adding 51-byte 00HEX at the beginning of the input packets that will feed in the RS(255,239,t=8), and removing theses bytes after the encoding.

$$p\left(x\right) = x^8 + x^4 + x^3 + x^2 + 1 \tag{3.1}$$

$$g\left(x\right) = \left(x + \lambda^0\right)\left(x + \lambda^1\right)\left(x + \lambda^2\right)\cdots\left(x + \lambda^{15}\right) \tag{3.2}$$

The packets at *RS Encoder* input are 188 bytes long, and the output is made of 204 bytes-long protected data packets,i.e. 187 data bytes plus 1 synchronization byte ($47HEX$) and 16 parity bytes.

3.1.2.3 Hierarchical Layer Divider

The function of the *Hierarchical Layer Divider* is to split the incoming bytes among the adequate number of layers. The number of layers will depend on the operational parameters choice, but the maximum is 3 (named A, B and C).

At the input of *RS Encoder*, the parity portion (i.e. the final 16 bytes) of data packets contains several pieces of information, including the layer to which the packet belongs to. This information is replaced by the parity bytes at the RS Encoder output, after the encoding process. Nevertheless, the layer information is saved and used to support the layer divider to perform the correct distribution of the packets among the right layers.

According to the standard [1], the first byte of each OFDM-Frame is the first byte of the data payload instead of the synchronization byte (i.e 47_{HEX}) of the first MPEG packet belonging to that OFDM-Frame. In other words, the first synchronization byte is dropped off side, and is not used.

3.1.2.4 Bytes to Bit

In order to be scrambled by a bit-based *Energy Dispersal* block, the bytes from the *Hierarchical Layer Divider* are converted into a stream of bits. Each layer has its own *Bytes to Bit* converter.

3.1.2.5 Energy Dispersal

Energy Dispersal block uses a PRBS, that has the *Polynomial Generator* defined in Equation 3.3, to scramble the incoming bit stream. This is performed in a layer basis. The main goal of the scrambling operation is to avoid long runs of zeros or ones.

$$g\left(x\right) = x^{15} + x^{14} + 1 \tag{3.3}$$

The initialization of $g(x)$ registers is [1 0 0 1 0 1 0 1 0 0 0 0 0 0 0].

3.1.2.6 Delay Adjustment

Delay Adjustment is a feature that together with the byte interleaving aims at providing the same transmission and reception delays for all hierarchical layers.

According to the modulation, code rate and operation mode, an appropriate delay (see Table 3.2) is chosen for each layer in such a way that all delays summed with [1]byte interleaver and byte deinterleaver delays, are always 1 frame length (i.e 204 OFDM symbols, as will be seen in the section OFDM-Frame).

3.1.2.7 Bits to Byte

To be processed by the *Byte Interleaver*, each bit stream belonging to a different hierarchical layer needs to be converted back to byte format.

3.1.2.8 Byte Interleaving

The next step in the ISDB-T transmitter flow, is the *Byte Interleaving*, which is achieved by means of a *Convolutional Byte Interleaver*, according to Figure 3.3. The MPEG synchronization byte shall pass trough the zero delay branch (path zero) of the convolutional interleaver.

[1] *Byte Interleaver* and *Byte Deinterleaver* cause 11 TSPs delays,each one

Table 3.2: Delay Adjustments According to [1].

CARRIER	CONVOLUTIONAL	DELAY ADJUTMENT: NUMBER OF TSPs		
MODULATION	CODE	MODE 1	MODE 2	MODE 3
	1/2	12 × N-11	24 × N-11	48 × N-11
	2/3	16 × N-11	32 × N-11	64 × N-11
QPSK/DQPSK	3/4	18 × N-11	36 × N-11	72 × N-11
	5/6	20 × N-11	40 × N-11	80 × N-11
	7/8	21 × N-11	42 × N-11	84 × N-11
	1/2	24 × N-11	48 × N-11	96 × N-11
	2/3	32 × N-11	64 × N-11	128 × N-11
16QAM	3/4	36 × N-11	72 × N-11	144 × N-11
	5/6	40 × N-11	80 × N-11	160 × N-11
	7/8	42 × N-11	84 × N-11	168 × N-11
	1/2	36 × N-11	72 × N-11	144 × N-11
	2/3	48 × N-11	96 × N-11	192 × N-11
16QAM	3/4	54 × N-11	108 × N-11	216 × N-11
	5/6	60 × N-11	120 × N-11	240 × N-11
	7/8	63 × N-11	126 × N-11	252 × N-11

N is the number of segments used by the hierarchical layer.

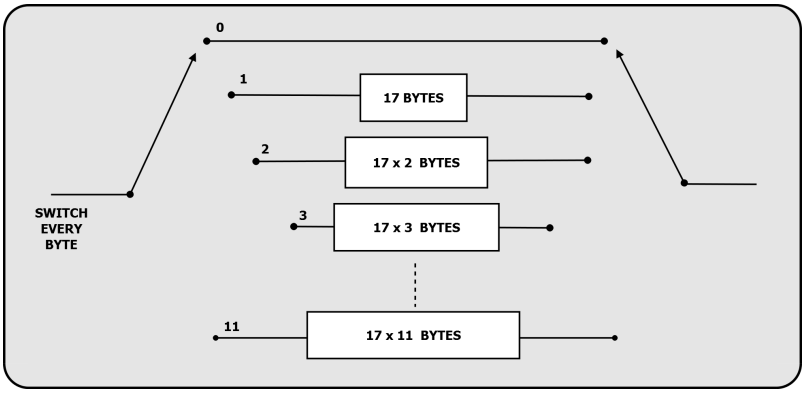


Figure 3.3: Byte interleaving.

3.1.2.9 Byte to Bits

Again, a *Byte to Bits* conversion is done, but at this time to generate a bit stream to feed the *Convolutional Interleaver*. This done for all the *Hierarchical Layer*

3.1.2.10 Convolutional Coding

The *Inner Coder* used in ISDB-T is a 1/2 rate punctured *Convolutional Encoder* with *Constraint Length* $k = 7$ and generating polynomials $G1 = 171_{OCT}$ and $G2 = 133_{OCT}$. Figure 3.4 shows the *Convolutional Encoder*.

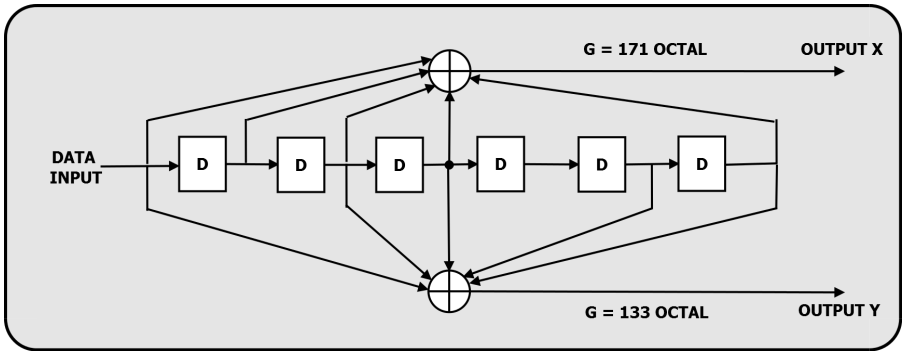


Figure 3.4: ISDB-T Convolutional Encoder.

After being encoded by the 1/2 rate encoder shown in Figure 3.4, the bit streams X and Y are punctured according to Table 3.3, to achieve the code rates 2/3, 3/4, 5/6 and 7/8. The puncture pattern is reset at the beginning of the OFDM-Frame.

Table 3.3: Convolutional Coding Transmission Sequence.

CODE RATE	PUNCTURING PATH	TRANSMISSION SEQUENCE
1/2	X: 1 Y: 1	X1, Y1
2/3	X: 1 0 Y: 1 1	X1, Y1, Y2
3/4	X: 1 0 1 Y: 1 1 0	X1, Y1, Y2, X3
5/6	X: 1 0 1 0 1 Y: 1 1 0 1 0	X1, Y1, Y2, X3, Y4, X5
7/8	X: 1 0 0 0 1 0 1 Y: 1 1 1 1 0 1 0	X1, Y1, Y2, Y3, Y4, X5, Y6, X7

3.1.2.11 Carrier Modulation

Carrier Modulation process is performed in three steps, as can be seen in Figure 3.5. Initially, a *Delay Adjustment* is performed, followed by the *Bit Interleaving* and *Mapping*. ISDB-T has four types of modulation to be used in the data payload: *DQPSK*, *QPSK*, *16QAM* and *64QAM*. Nevertheless, in general, the commercial Integrate Circuit implementations do not use DQPSK, modulation. It is worth to mention at this point that bit zero $'0'$ is mapped to 1 and bit one $'1'$ is mapped to -1 before being modulated[1].

With the aim of having a total of 2 OFDM symbols delay, for any modulation or operation mode the *Delay Adjustment* is performed, according to Table 3.4. The 2 OFDM symbols delays is the sum of that delay adjustment plus the 120 carriers delay at the mapper and demapper.

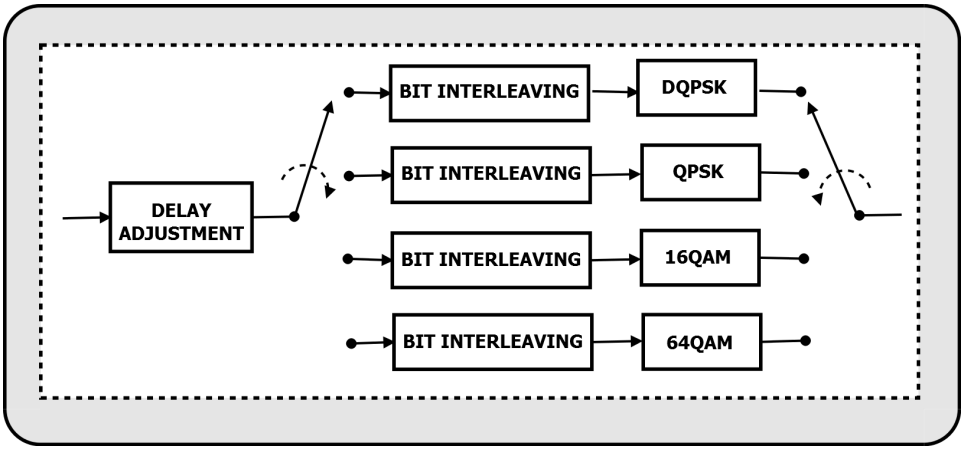


Figure 3.5: Carrier modulation configuration.

Table 3.4: Carrier Modulator Delay Adjustments Values.

CARRIER	DELAY ADJUTMENT: NUMBER OF BITS		
MODULATION	MODE 1	MODE 2	MODE 3
DQPSK/QPSK	384 × N-240	768 × N-240	1536 × N-240
16QAM	768 × N-480	1536 × N-480	3072 × N-480
64QAM	1152 × N-720	2304 × N-720	4608 × N-720

N is the number of segments used by the hierarchical layer.

[1] See the constellation mapping to complex symbols if Figures 3.13, 3.11, 3.9 and 3.7.

3.1.2.11.1 DQPSK

$\pi/4 - shift\ DQPSK$ modulation works as shown in Figure 3.6. The serial bit stream is split into 2 streams. The top branch is a zero-delay branch, and the bottom branch, passes through a 120-bits delay line. Next, the two bits are used to compute the phase shift based in the Table 3.5 mapping. The computed phase is used to rotate the previous DPQSK symbol, according to the Equation 3.4.

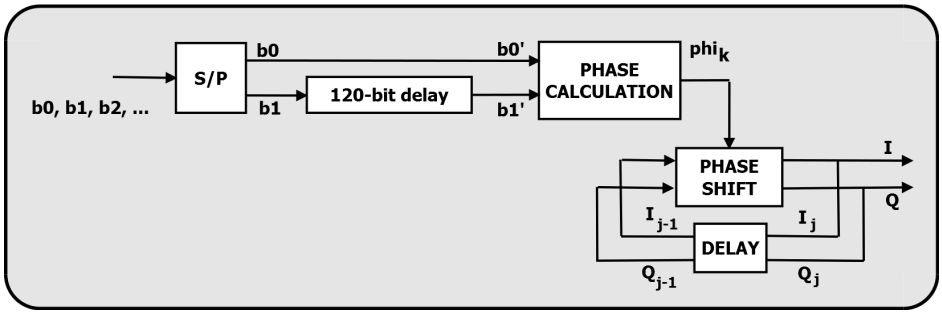


Figure 3.6: DQPSK modulation and bit interleaving.

$$\begin{pmatrix} I_j \\ Q_j \end{pmatrix} = \begin{pmatrix} cos\theta_j & -sin\theta_j \\ sin\theta_j & cos\theta_j \end{pmatrix} \begin{pmatrix} I_{j-1} \\ Q_{j-1} \end{pmatrix} \tag{3.4}$$

Table 3.5: Phase as Function of Input Bits.

INPUT	OUTPUT
b0′ b1′	θ_j
0 0	$\pi/4$
0 1	$-\pi/4$
1 0	$3\pi/4$
1 1	$-3\pi/4$

3.1.2.11.2 QPSK

According to Figure 3.8, the input bit stream is split into 2 sub-streams. The sub stream in the bottom branch passes through a 120-bit delay line and the top branch has zero delay. Next, the bits are mapped according to Figure 3.9.

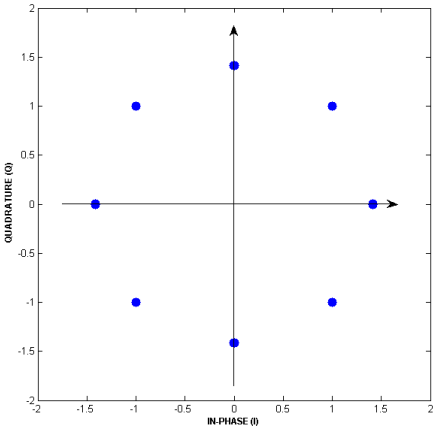


Figure 3.7: DQPSK modulation constellation.

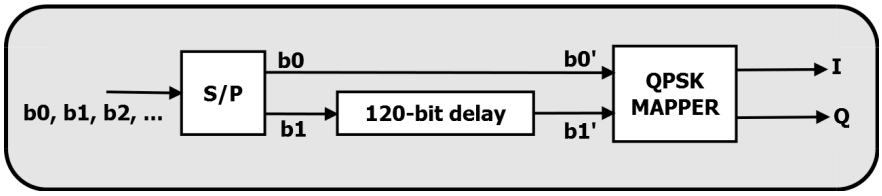


Figure 3.8: QPSK modulation and bit interleaving.

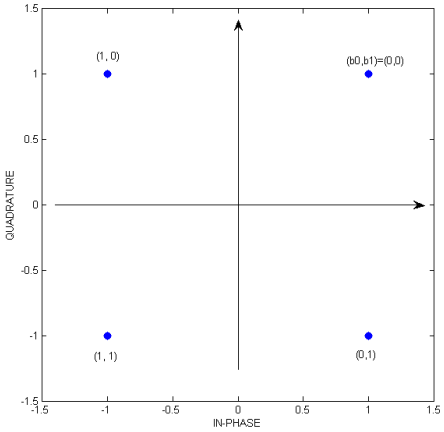


Figure 3.9: QPSK modulation constellation.

3.1.2.11.3 16-QAM

16-QAM modulation and interleaving diagram is shown in Figure 3.10. The resulting constellation is presented in Figure 3.11. The input bit stream is split into 4 branches, the first one is placed in the top and has zero delay, It corresponds to bit b0. The following from top down correspond to bits b1, b2 and b3 respectively. Each one of these three branches has a different delay, as can be seen in Figure 3.10.

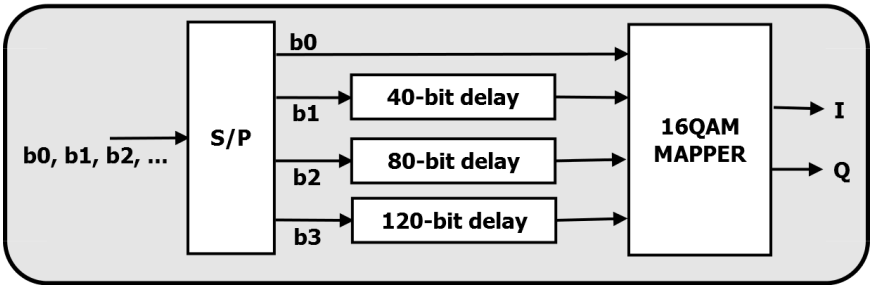


Figure 3.10: 16-QAM modulation and bit interleaving.

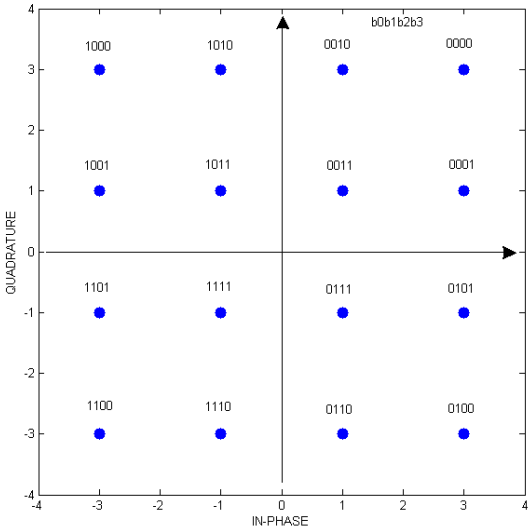


Figure 3.11: 16-QAM modulation constellation.

3.1.2.11.4 64-QAM

The input bit stream is split into 6 branches, the first one is placed in the top and has zero delay, and corresponds to bit b0. The following, from top down, correspond to bits b1, b2, b3, b4 and b5, respectively. Each one of these three branches has a different delay, as can be seen if Figure 3.10. The 64-QAM modulation and interleaving diagram is shown in Figure 3.12. The resulting constellation is presented in Figure 3.13.

3.1.2.11.5 NORMALIZATION FACTOR

To achieve an unitary average power, ISDB-T standard [1] defines the normalization factors shown in Table 3.6. After mapping the input bits into complex symbols $Z = I + jQ$, these symbols are normalized accordingly.

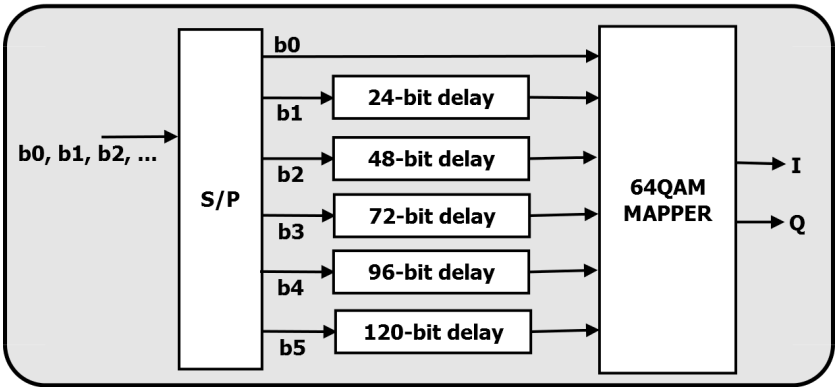


Figure 3.12: 64-QAM modulation and bit interleaving.

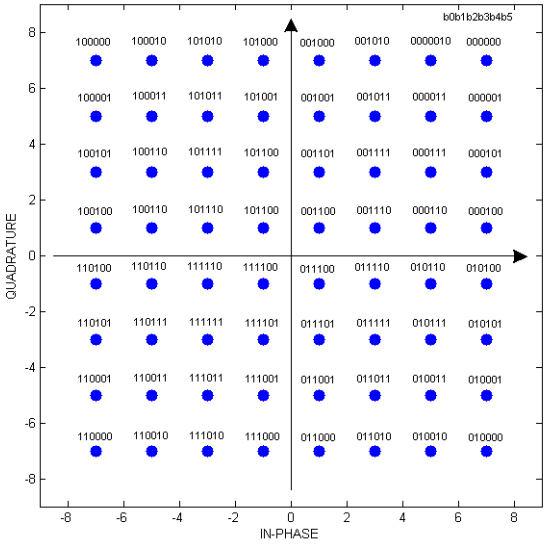


Figure 3.13: 64-QAM modulation constellation.

Table 3.6: Normalization Factor.

CARRIER MODULATION	NORMALIZATION FACTOR
$\pi/4$-shift DQPSK	$Z/\sqrt{2}$
QPSK	$Z/\sqrt{2}$
16-QAM	$Z/\sqrt{10}$
64-QAM	$Z/\sqrt{42}$

3.1.2.11.6 Data Segment

The stream of complex symbols is grouped in *Data Segments*. Later, between these symbols, pilots and control information will be added to form an *OFDM-Frame*. The number of complex symbols per *Data Segment* is 96, 192 and 384 for Modes 1, 2 and 3 respectively. Each complex symbol mapped in *DQSPK*, *QPSK*, *16-QAM* and *64-QAM* can be represented by $S_{i,j,k}$, where i is the carrier index within a segment, k is the segment number and j is the number of the OFDM symbols that the segment belongs to. Figure 3.14 shows the representation of 1 segment configuration for Modes 1, 2 and 3. It is worth recalling that a single OFDM symbol has 13 segments.

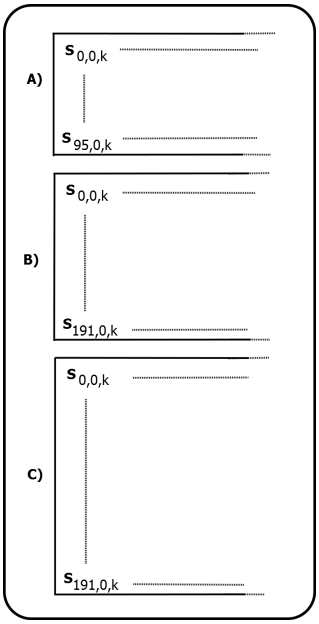


Figure 3.14: A single *Data Segment* representation for Modes 1 (A), 2 (B) and 3 (C).

3.1.2.12 Hierarchical Layer Combining

At this point, the TS stream or streams (in the case of more than 1 layer), belonging to different layers are already channel coded and mapped into complex symbols. Those independent data layers are grouped to form the *Data Segment* structure and subsequently are read at IFFT Sample Rate (i.e. 512/63 MHz), from carrier number 0 of *Data Segment* 1 to carrier number N_{c-1} of *Data Segment* number 13. This procedure is shown in Figure 3.15. In the figure N_{s1}, N_{s2} and N_{s3} are the number of *Data Segment*, per *Hierarchical Layer*. The sum $N_{s1} + N_{s2} + N_{s3}$ is always 13.

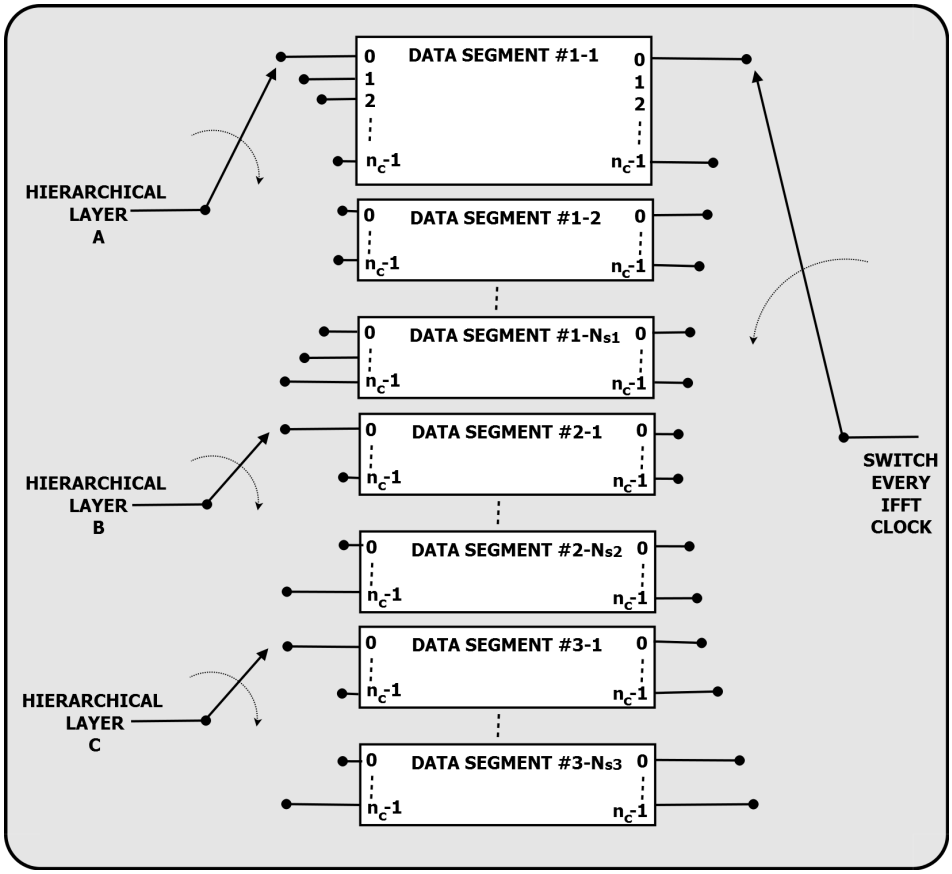


Figure 3.15: Hierarchical layer combiner.

3.1.2.13 Time Interleaving

The aim of the *Time Interleaving* block is to achieve robustness against wireless channel fading. It also allows performance improvement under spike noise scenarios. Each independent *Hierarchical Layer*, i.e. A, B or C, is made of *Data Segments* that transports data using the same modulation. Nevertheless, two or more layers can make use of the same modulation.

Time Interleaving is done in a *Data Segment* basis, as shown in Figures 3.16 and 3.17, using the parameter I shown in Table 3.7. The parameter is independently specified for each layer, because each layer can target a different type of reception equipment. For instance, 1-SEG (layer A) targets high mobility terminals as cell phones and other low screen resolution handheld devices, so a larger interleaver length can be used in this case. On the other, hand a hypothetical second layer (named B) with 12 segments targeting fixed-reception Set-Top-Boxes (STB), can have a interleaver length smaller than that used for 1-SEG layer.

Table 3.7: Time Interleaver Parameters.

MODE 1			MODE 2			MODE 3		
I LENGTH	DELAYED SYMBOLS	DELAYED FRAMES (TX and RX)	I LENGTH	DELAYED SYMBOLS	DELAYED FRAMES (TX and RX)	I LENGTH	DELAYED SYMBOLS	DELAYED FRAMES (TX and RX)
0	0	0	0	0	0	0	0	0
4	28	2	2	14	1	1	109	1
8	56	4	4	28	2	2	14	1
16	112	8	8	56	4	4	28	2

Delay in Symbols.

Figures 3.18 and 3.19 show two examples of *Time Interleaver* behavior for two configurations: Mode 1, I=8 and Mode 3, I=6. As can be seen in the figures, as I increases the number of symbols delay in the *OFDM symbol* direction increases. The value of m_i is computed according to Equation 3.5. Another issue related to time interleaving is the need for adding an appropriate extra complex symbols delay, in order to achieve a transmission and reception delay multiple of an *OFDM Frame* length[1].

$$m_i = (i \times 5) mod 96 \tag{3.5}$$

As can be seen from Equation 3.5 and Figures 3.18 and 3.19, as the value of I increases, the interleaver depth increases. So, it is worth to note at this moment that the price paid for the time interleaving is a large area occupied by the intereleaver memory within the ASIC receiver. At the transmitter this is not an issue because that

[1] 1 OFDM Frame is 204 OFDM symbols long.

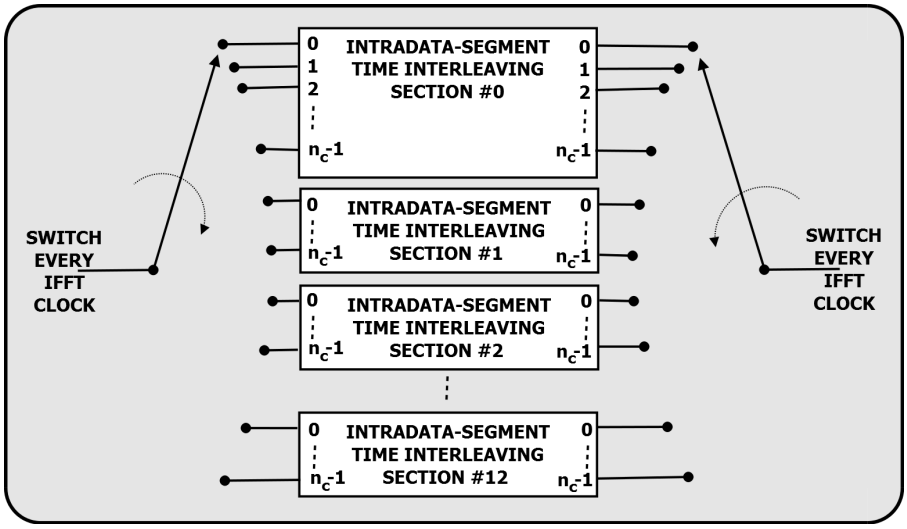


Figure 3.16: Time interleaving section.

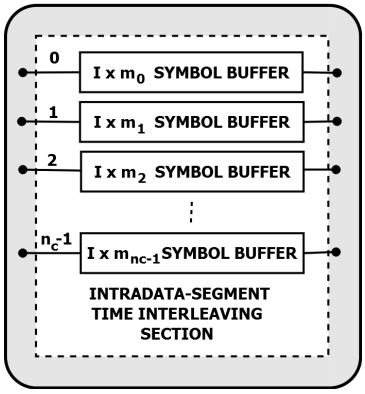


Figure 3.17: Intra segment time interleaving section.

kind of equipment has much larger dimensions than the receivers and in general they use a memory chip separated from the baseband (or IF) modulator implemented in FPGA or ASIC. In the case of the ASIC receivers, the memory will play a significant role in the cost of the ASIC production. The area occupied by the receiver *De-interleaver* memory will be shown in Chapter 6.

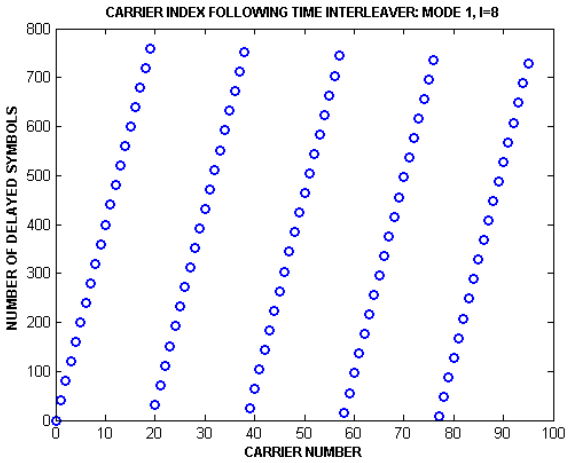


Figure 3.18: Time Interleaver carrier arrangement for Mode 1, I=8.

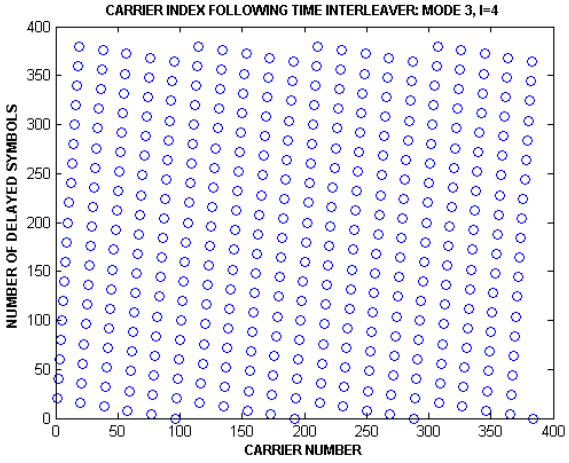


Figure 3.19: Time Interleaver carrier arrangement for Mode 3, I=4.

3.1.2.14 Frequency Interleaving

Frequency Interleaver treats differently *Data Segments* belonging to the following groups:

- *Data Segment* belonging to the *Partial Reception* segment, also known as 1-SEG;
- *Data Segment(s)* containing differential modulation, i.e. DQPSK;
- *Data Segment(s)* transporting coherent modulation, i.e. QPSK, 16-QAM and 64-QAM.

As can be seen in Figure 3.20, the *Data Segment(s)* belonging to those groups listed above goes through a different processing branch.

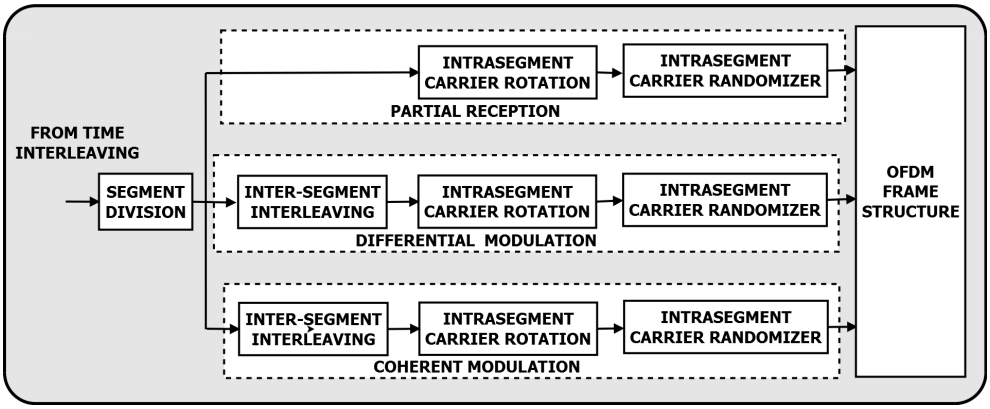


Figure 3.20: Frequency interleaving scheme.

The hierarchical layers A, B and C are defined according to the number of *Data Segments* that each one transports, in such a way that layer A is the one with the smallest number of *Data Segments* and C is the one with the largest number. According to [1], during the segment division, *Data Segments* from 0 to 12 are assigned sequentially to the partial-reception portion, differential modulation and coherent modulation. *Data Segments* of the same hierarchical level must be arranged successively. As can be seen in Figure 3.20, *Inter-Segment Interleaving* shall be conducted

on segments belonging to the same kind of modulation, i.e. coherent or differential modulation group. Inter-segment shall be conducted on *Data Segment* belonging to different *Hierarchical Layer* as long as they transport the same modulation group.

3.1.2.14.1 Intersegment Time Interleaving

The carrier symbols, within the Data Segments, must be intersegment interleaved according to the mapping shown in Figures 3.21, 3.23 and 3.25. In those figures, $S_{i,j,k}$, represents the carrier symbols and n the number of the *Data Segments*. As previously mentioned, the *Intersegment Interleaving* shall be performed on each of the synchronous modulation and differential modulation. Figures 3.22, 3.24 and 3.26, show the relation between the index of the carriers at the input of the *Intersegment Interleaver* block and the index at its output, for Modes 1, 2 and 3, for a hypothetical layer B with 12 *Data Segments*.

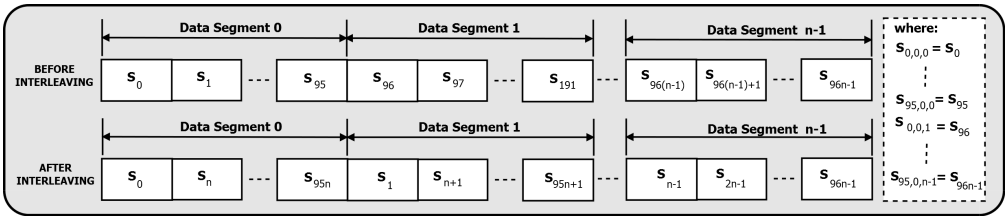


Figure 3.21: Inter-segment Interleaver Mode 1.

3.1.2.14.2 Intrasegment Interleaver

As can be seen in Figure 3.20, *Intrasegment Interleaver* is conducted in two steps: the first is the *Intrasegment Carrier Rotation* and the second is *Intrasegment Carrier Randomizer*. Following, a short description of these sub-blocks is given.

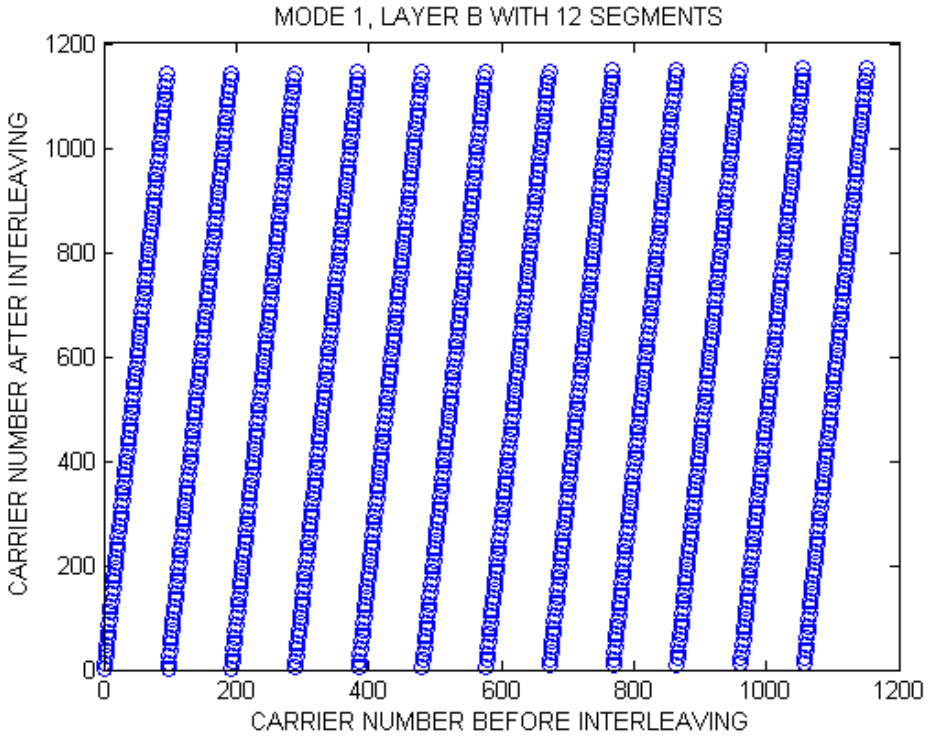


Figure 3.22: Example of Inter-segment Interleaver Mode 1, layer B with 12 Data Segments.

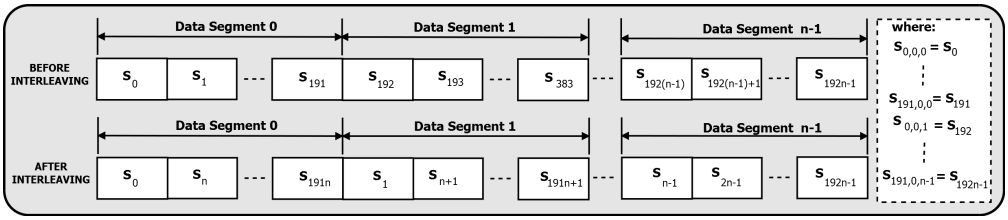


Figure 3.23: Inter-segment Interleaver Mode 2.

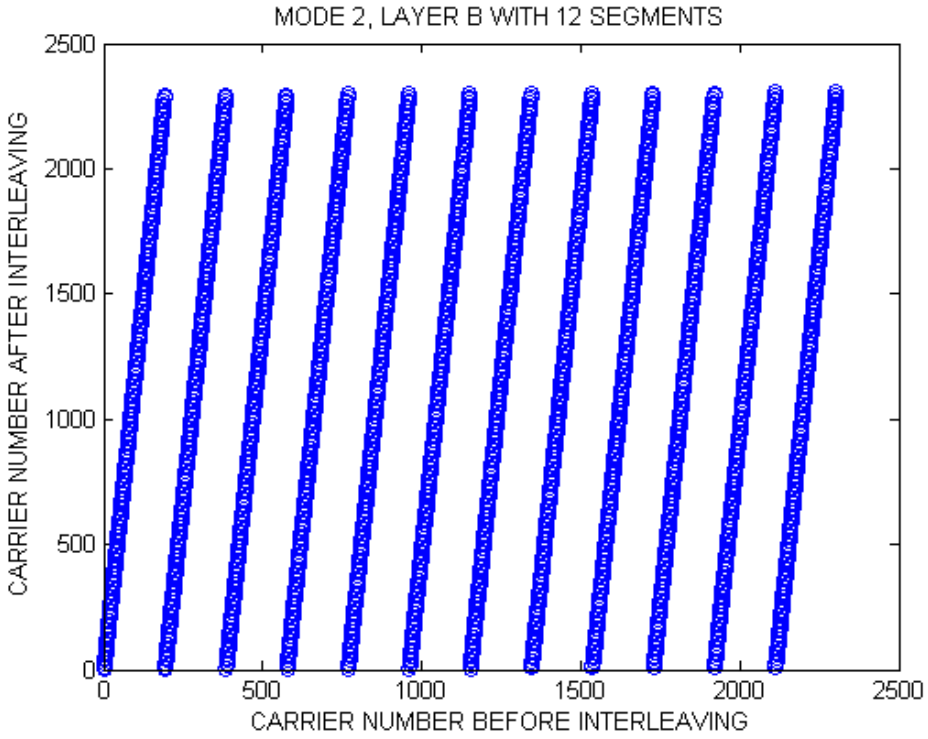


Figure 3.24: Example of Inter-segment Interleaver Mode 2, layer B with 12 Data Segments.

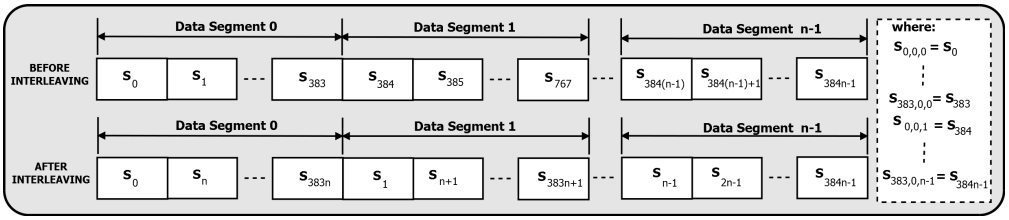


Figure 3.25: Inter-segment Interleaver Mode 3.

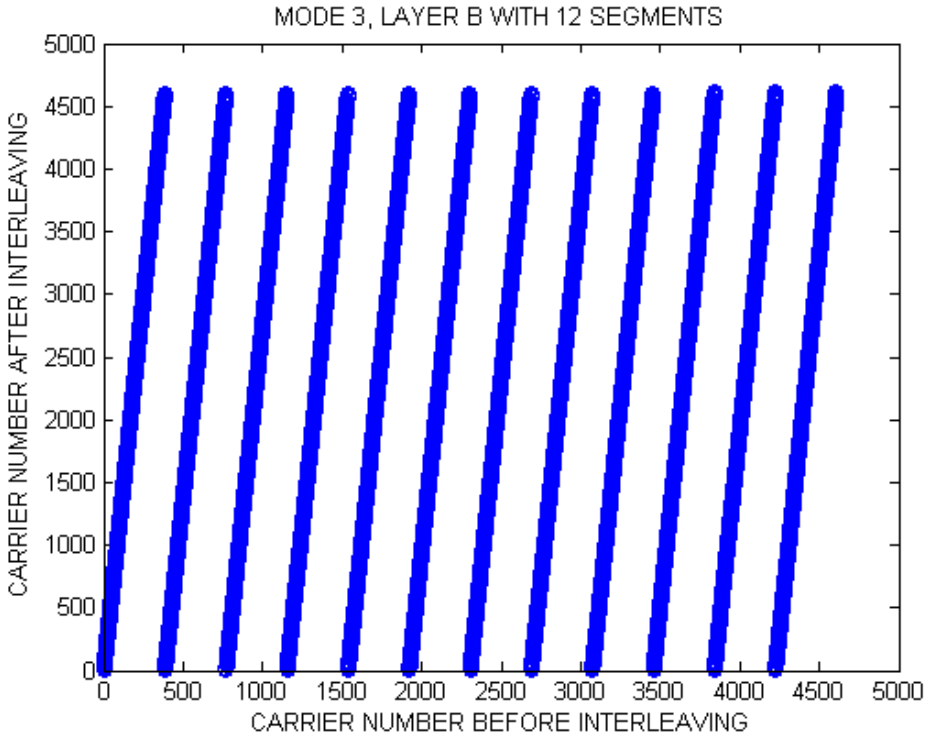


Figure 3.26: Example of Inter-segment Interleaver Mode 3, layer B with 12 Data Segments.

3.1.2.14.2.1 Intrasegment Carrier Rotation

The carrier rotation is carried out according to Figures 3.27, 3.29 and 3.29. The resulting carrier rotation for Modes 1, 2 and 3 are presented in Figures 3.28, 3.30 and 3.32. Those figures show the relation between the carrier's indexes at the input and output of the *Intrasegment Carrier Rotation* block.

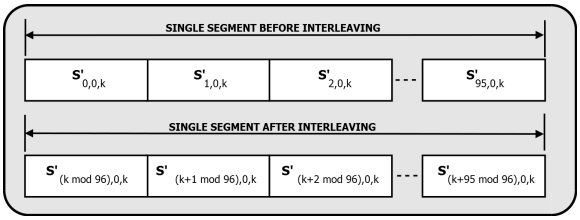


Figure 3.27: Intra-segment Interleaver, Mode 1.

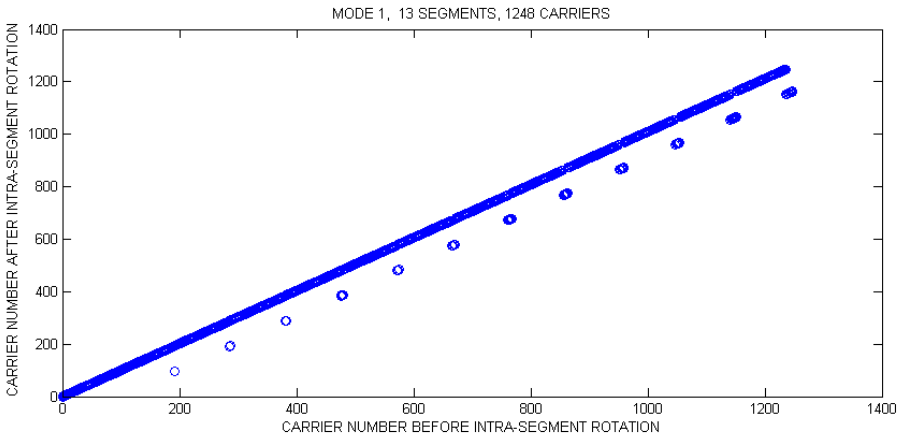


Figure 3.28: Intra-segment carrier rotation for Mode 1.

3.1.2.14.3 Intrasegment Carrier Randomizer

Intrasegment Carrier Randomizer is based on several mapping tables presented in section 3.11.2.2 of the ISDB-T standard [1]. Table 3.8 is the reproduction of the

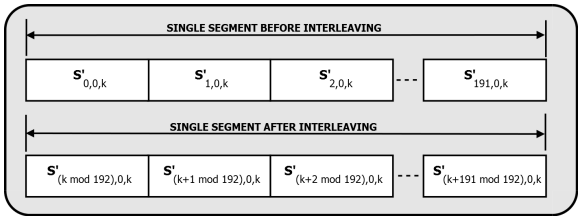


Figure 3.29: Intra-segment interleaver for Mode 2.

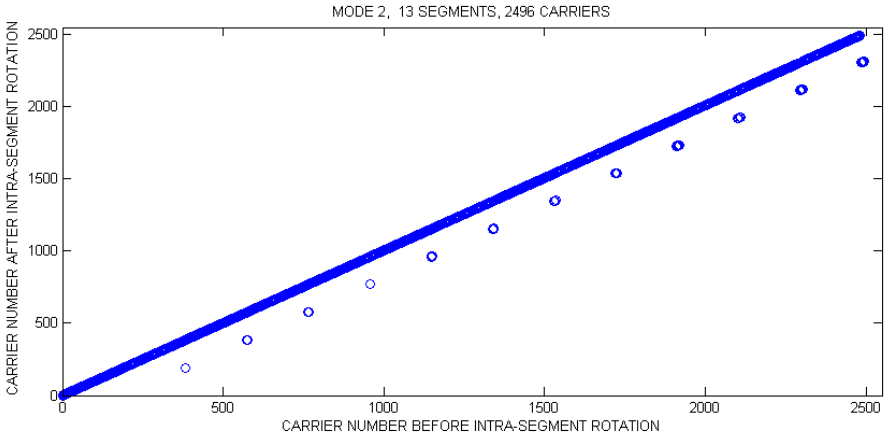


Figure 3.30: Intra-segment carrier rotation for Mode 2.

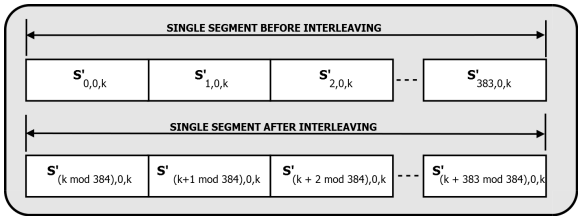


Figure 3.31: Intra-segment Interleaver for Mode 3.

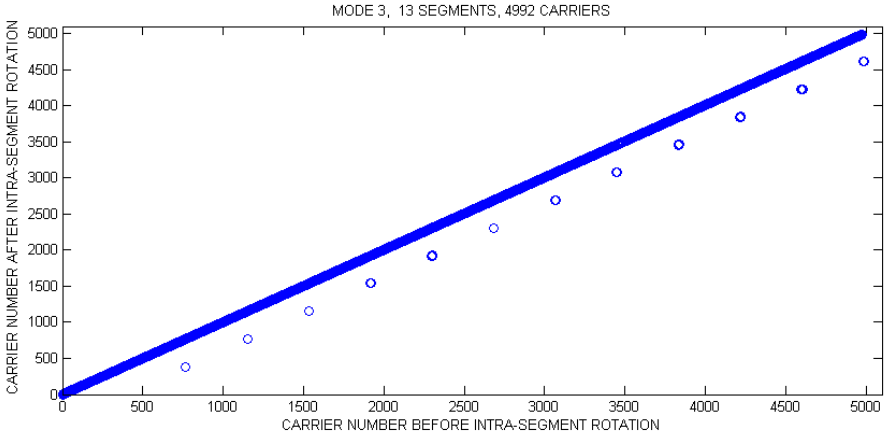


Figure 3.32: Intra-segment carrier rotation for Mode 3.

mapping table for Mode 1. It is straightforward to figure out that the size of the tables for Modes 2 and 3 are respectively the double and four times the size of Table 3.8. As a matter of size and due to the fact that those tables do not bring any new information beyond the fixed random mapping defined by the standard, I do not reproduce those tables here.

According to the standard, the goal of carrier rotation and randomizations is to eliminate periodicity in the carrier arrangement, making it possible to prevent burst errors in a specific segment, due to the matching of the carrier arrangement period and the frequency selective fading after *Intersegment Interleaving*.

As a matter of illustration, Figures 3.33 and 3.34 show the mapping resulting intrasegment random mapping before and after the *Time Interleaver*, for *Segment* number, zero, Mode 1.

BEFORE	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23
AFTER	80	93	63	92	94	55	17	81	6	51	9	85	89	65	52	15	73	66	46	71	12	70	18	13
BEFORE	24	25	25	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47
AFTER	95	34	1	38	78	59	91	64	0	28	11	4	45	35	16	7	48	22	23	77	56	19	8	36
BEFORE	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64	65	66	67	68	69	70	71
AFTER	83	40	14	79	27	57	44	37	30	68	47	88	75	41	90	10	33	32	62	50	58	82	53	24
BEFORE	72	73	74	75	76	77	78	79	80	81	82	83	84	85	86	87	88	89	90	91	92	93	94	95
AFTER	83	40	14	79	27	57	44	37	30	68	47	88	75	41	90	10	33	32	62	50	58	82	53	24

Table 3.8: Intrasegment Carrier Randomizing in Mode 1.

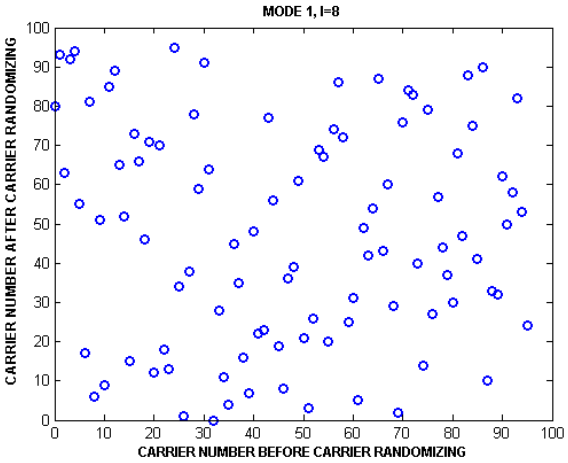


Figure 3.33: Carrier arrangement for Carrier Randomizer without Time Interleaver: Mode 1, I=8, Segment 0.

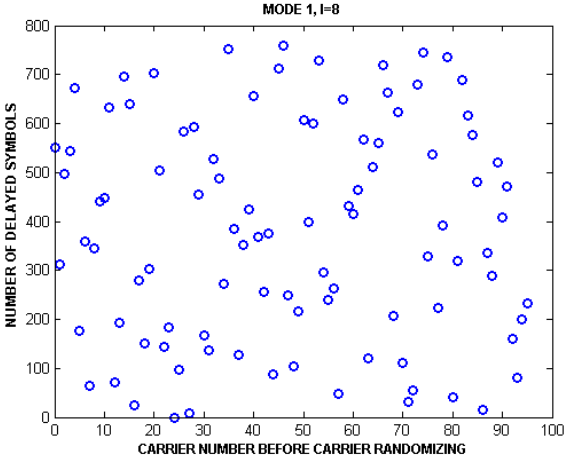


Figure 3.34: Carrier arrangement for Carrier Randomizer after Time Interleaver: Mode 1, I=8, Segment 0.

3.1.2.15 OFDM Frame Structure

According to the ISDB-T transmitter block diagram, shown in Figure 3.1, all necessary processing for channel coding has been done at this point. The next step to be performed at the transmitter side is to build the *OFDM Frame* Structure.

OFDM Frame structure is made of 13 *Data Segments* containing modulated coded and interleaved symbols, *Scattered Pilot* (SP), TMCC information, *Auxiliary Channel* (AC) information and *Continual Pilots*(CP). There are two possible types of *Data Segments* in what concerns modulation. The first, transports *Differential Modulation* symbols, and the second *Synchronous Modulation*. Nevertheless, it is worth to recall that most commercial ISDB-T receivers do not implement demodulation of *Differential Modulation*.

Basically, the OFDM Segment is built, by embedding TMCC, AC, CP (for DQPSK) and SP(for Synchronous Modulation) in between the data subcarriers. This mapping follows several random arrangements predefined in the section 3.12 of ISDB-T standard.

3.1.2.15.1 OFDM Segment for Differential Modulation

Figure 3.35 shows an schematic view of an *OFDM Segment* for differential modulation. It contains embedded TMCC, CP and *Auxiliary Channels* AC1 and AC2. In Figure 3.35, $S_{i,j}$ represents the data symbols within the Segment after the interleaving section. The figure shows the scheme for Mode 1. The total number of subcarriers after embedding pilot, control information and, AC1 and AC2 is 107. Table 3.9 shows the number of TMCC, CP, AC1 and AC2 subcarriers for each *Mode*. The number of carriers per segment before OFDM Segment arrangements for Modes 1,2 and 3 are respectively 96, 182 and 384. After the arrangement, the number of carriers is 108, 216 and 432.

3.1.2.15.2 OFDM Segment for Coherent Modulation

As can be observed when comparing Figure 3.35 and Table 3.9 to Figure 3.36 and Table 3.10, the *OFDM Segment* structures for coherent and differential modulations are significantly different, especially regarding the number of TMCCs and the SP necessary for channel estimation and equalization, and synchronization on synchronous modulation case. As in the case of differential modulation, the number of carriers per segment before *OFDM Segment* arrangement for Modes 1,2 and 3 are respectively 96,

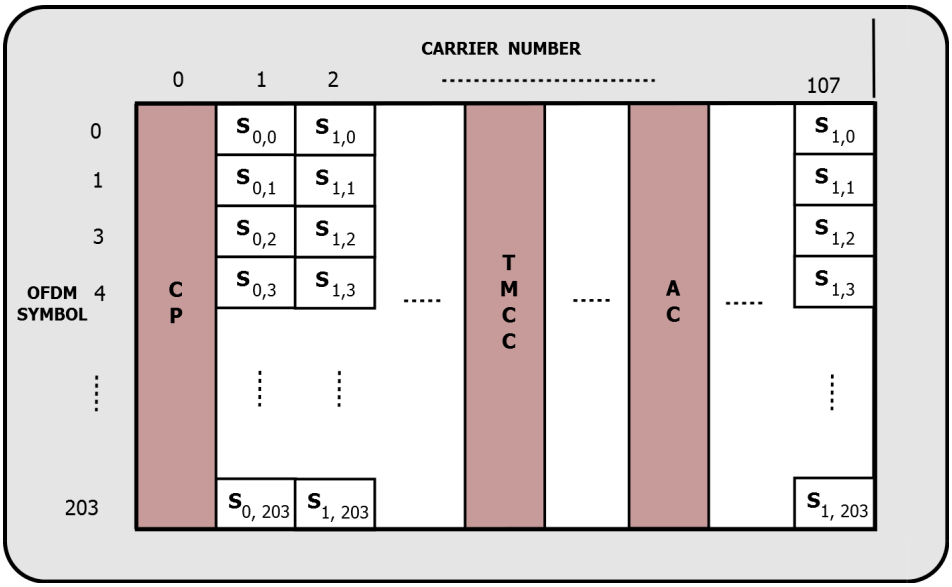


Figure 3.35: Configuration of a OFDM Segment for differential modulation.

Table 3.9: Number of TMCC, AC1, AC2 and CP For Each Mode: Differential Modulation.

MODE 1				MODE 2				MODE 3			
1 SINGLE SEGMENT											
CP	TMCC	AC1	AC2	CP	TMCC	AC1	AC2	CP	TMCC	AC1	AC2
1	5	2	4	1	10	4	9	1	20	8	19
N SEGMENTS											
CP	TMCC	AC1	AC2	CP	TMCC	AC1	AC2	CP	TMCC	AC1	AC2
N*1	N*5	N*2	N*4	N*1	N*10	N*4	N*9	N*1	N*20	N*8	N*19

N is the number of *Segments*.

182 and 384. After the arrangement the number of carriers is 108, 216 and 432.

OFDM Frame

Table 3.10: Number of TMCC and AC1 For Each Mode: Synchronous Modulation.

MODE 1		MODE 2		MODE 3	
SINGLE SEGMENT					
TMCC	AC1	TMCC	AC1	TMCC	AC1
1	2	2	4	4	8
N SEGMENTS					
TMCC	AC1	TMCC	AC1	TMCC	AC1
N*1	N*2	N*2	N*4	N*4	N*8

N is the number of *Segments*.

3.1.2.16 Pilot and Control Signals

I will now describe briefly the characteristics of SP, CP, TMCC and AC signals mentioned in the previous sub-section. Later, in Section 3.1.2.17, the main TMCC parameters will be described in details.

3.1.2.16.1 Scattered Pilot (SP)

ISDB-T standard SP is generated using the *Polynomial Generator* presented below:

$$x^{11} + x^9 + 1.$$

This polynomial generates a BPSK PRBS, and can be implemented using a *Linear Feedback Shift Register* (LFSR) as shown in Figure 3.37. The output scrambling sequence is W_i, where i corresponds to the carrier index within an *OFDM Symbol*. The initialization of the LFSR, for all modes and *Segment* numbers are shown in Table 3.11.

Before being embedded within the *OFDM Segments*, the BPSK bits W_i are mapped to the complex values shown in Table 3.12.

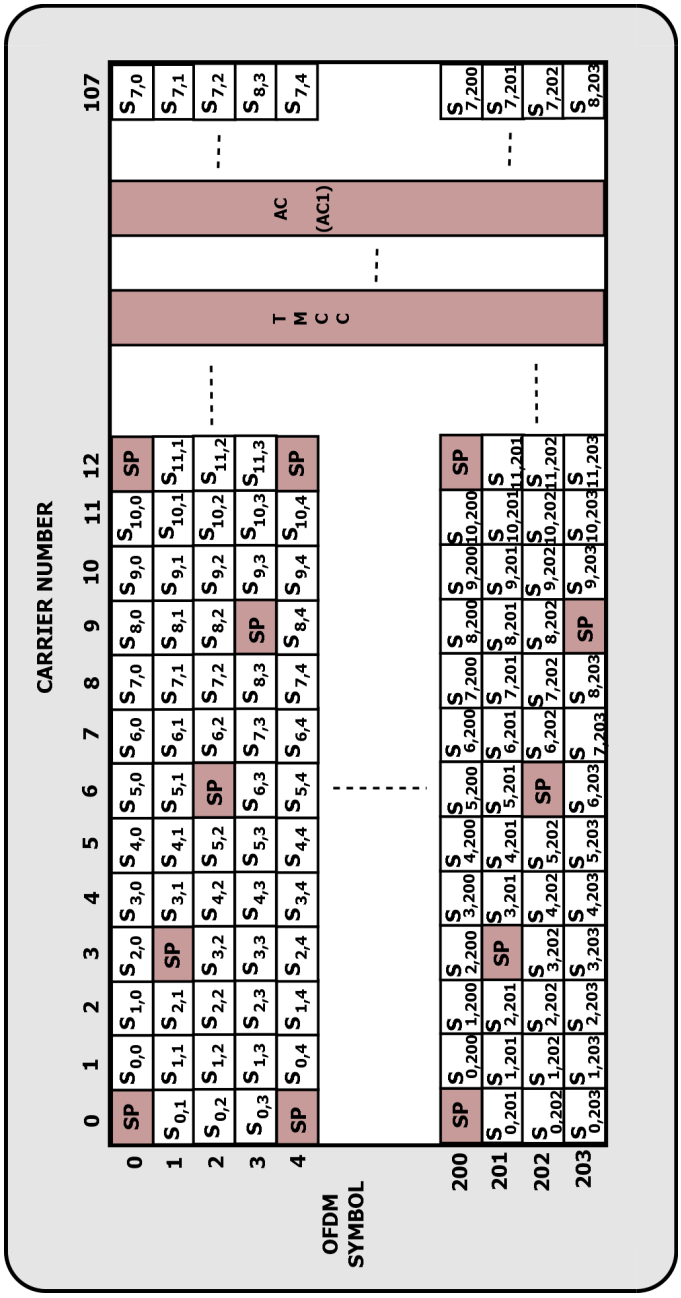


Figure 3.36: Configuration of a OFDM Segment for synchronous modulation.

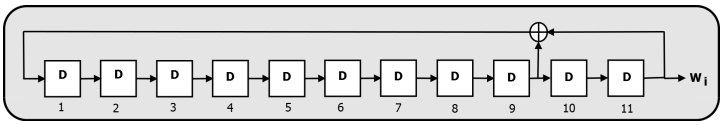


Figure 3.37: Scattered Pilot PRBS based on LFSR.

Table 3.11: PRBS Initialization.

SEGMENT	MODE 1	MODE 2	MODE 3
11	1 1 1 1 1 1 1 1 1 1 1 1	1 1 1 1 1 1 1 1 1 1 1 1	1 1 1 1 1 1 1 1 1 1 1 1
9	1 1 0 1 1 0 0 1 1 1 1	0 1 1 0 1 0 1 1 1 1 0	1 1 0 1 1 1 0 0 1 0 1
7	0 1 1 0 1 0 1 1 1 1 0	1 1 0 1 1 1 0 0 1 0 1	1 0 0 1 0 1 0 0 0 0 0
5	0 1 0 0 0 1 0 1 1 1 0	1 1 0 0 1 0 0 0 0 1 0	0 1 1 1 0 0 0 1 0 0 1
3	1 1 0 1 1 1 0 0 1 0 1	1 0 0 1 0 1 0 0 0 0 0	0 0 1 0 0 0 1 1 0 0 1
1	0 0 1 0 1 1 1 1 0 1 0	0 0 0 1 0 1 1 0 0 0 0	1 1 1 0 0 1 1 0 1 1 0
0	1 1 0 0 1 0 0 0 0 1 0	0 1 1 1 0 0 0 1 0 0 1	0 0 1 0 0 0 0 1 0 1 1
2	0 0 0 1 0 0 0 0 1 0 0	0 0 0 0 0 1 0 0 1 0 0	1 1 1 0 0 1 1 1 1 0 1
4	1 0 0 1 0 1 0 0 0 0 0	0 0 1 0 0 0 0 1 1 0 0 1	0 1 1 0 1 0 1 0 0 1 1
6	1 1 1 1 0 1 1 0 0 0 0	0 1 1 0 0 1 1 1 1 0 0 1	1 0 1 1 1 0 1 0 0 1 0
8	0 0 0 0 1 0 1 1 0 0 0	1 1 1 0 0 1 1 0 1 1 0	0 1 1 0 0 0 1 0 0 1 0
10	1 0 1 0 0 1 0 0 1 1 1	0 0 1 0 1 0 1 0 0 0 1	1 1 1 1 0 1 0 0 1 0 1
12	0 1 1 1 0 0 0 1 0 0 1	0 0 1 0 0 0 0 0 1 0 1 1	0 0 0 1 0 0 1 1 1 0 0

Table 3.12: W_i Signal Modulation Amplitude.

W_i	MODULATING SIGNAL (I, Q)
1	$(-4/3, 0)$
0	$(+4/3, 0)$

3.1.2.16.2 Continual Pilot (CP)

As with SP, CP is a BPSK signal modulated by the values shown in Table 3.12. The continuous value that the modulated signal assumes depends on the subcarrier index. For instance, we can assume a hypothetical OFDM transmitter configured to work using coherent modulation within all 13 *Segments*, in *Mode 1*, i.e., the number of carriers within a single *OFDM Segment* is 108 and 1045 within an entire *OFDM Symbol*. $1045 = 13 \times 108 + 1$, where the last element corresponds to CP carrier addition. The PRBS reset, is done at the first carrier, of the first *OFDM Symbol* of an *OFDM Frame*. Then, after 1405 clock cycles, the subcarrier index is 1404. So, the value that CP assumes is the value the PRBS output W_i has at this point. This value is going to be used for all *OFDM Symbols*, i.e., continuously over the entire broadcasting time. Equivalently, one can assume that there is a reset at each *OFDM Segment* and the registers shown in Figure 3.37 are loaded with the values presented in Table 3.11. Then, at the beginning of *OFDM Segment* number 13 there is a reset, and the registers of PRBS are loaded with the adequate values. After 109 system clocks, the output W_i assumes the same value as the carrier 1044 in the previous explanation.

3.1.2.16.3 TMCC

TMCC information is transmitted using DBPSK. The parameters that TMCC transports are presented in details in Section 3.1.2.17. The reference for differential modulation B_0, presented in Table 3.14, is determined according to PRBS W_i output, which is related to the carrier index i, of the carrier which transports the TMCC. Refer to the standard, i.e. [1], to see the exactly fixed random carrier location for the TMCCs for each ISDB-T configuration. Bits B_0 to B_{203} are the values previous to differential modulation. B'_0 to B'_{203} are the bits to be transmitted after differential modulation. They are obtained as follows:

$$B'_0 = W_i$$
$$B'_k = B'_{k-1} \oplus B_k$$

where, k varies from 1 to 203. After differential modulation, B' bits are amplitude modulated according to the values presented in Table 3.12.

3.1.2.16.4 AC

AC is used to transport additional ISDB-T information. The values AC transport are differentially modulated using DBPSK. The reference value for the modulation is the first carrier value assumed by the PRBS at the beginning of an *OFDM Frame*. The AC bit values are amplitude modulated according to Table 3.12. ISDB-T has two possible ACs, called AC1 and AC2. It is worth to mention that in this work the AC values are discarded at the receiver. For this reason, no more details on those channels are provided here.

3.1.2.16.5 Transmission Spectrum

ISDB-T *Data Segments* from 0 to 12, with the embedded SPs, ACs, TMCCs and CP are called *OFDM Segments*. After the addition of these elements the *OFDM Segments* must be rearranged in order to be transmitted adequately over-the-air. Figure 3.38 shows the *Segment* order before and after the arrangement. The amplitude of the CP shown in that figure is modulated by the values presented in Table 3.13.

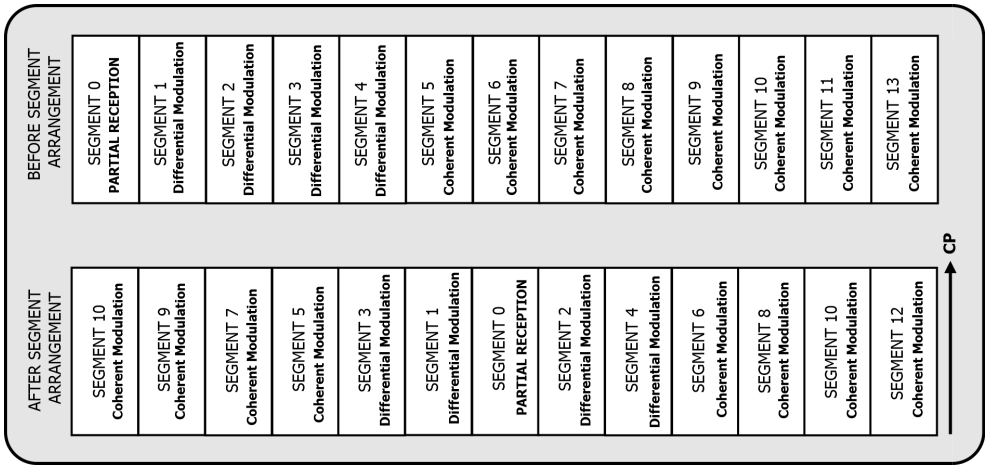


Figure 3.38: ISDB-T OFDM Frame Spectrum.

Segment 0 must be transmitted at *OFDM Segment* number 0. This segment must always be placed at the central portion of the band. For hierarchical transmissions, differentially modulated *OFDM Segments* must be alternately placed around *OFDM Segment* 0. The coherently modulated *OFDM Segments* must be alternately placed

Table 3.13: Continual Pilot Carrier Modulation Signal Amplitude.

MODE	MODULATING SIGNAL FOR (I, Q)MODE 3
1	$-4/3, 0)$
2	$(+4/3, 0)$
3	$(+4/3, 0)$

Continual Pilots, as the name states, is a continual stream of complex symbol $I + jQ$, across *OFDM Symbols*.

around the segments that transports *Differential Modulation*.

3.1.2.16.6 IFFT and GI

After the 13 Segments have been arranged, in the alternate segments configuration, carriers must be reordered and stuffed with zeros in the central part before the IFFT is applied. The reorder is to adequately shape the spectrum after IFFT, and stuffing it with zeros is necessary to complete the number of subcarriers in such a way that the number of subcarriers belonging the 13 segments plus the most right CP, and the stuffed zeros is equal to 2048, 4096 and 8192 subcarriers for Modes 1, 2 and 3 respectively. Subsequently to IFFT be applied to the 2048, 4096 or 8192 subcarriers, a Guard Interval is inserted, as can be shown in Figure 3.39. Now the signal can be up-converted and power amplified to be transmitted over the air.

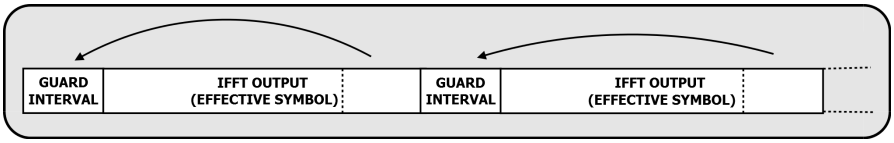


Figure 3.39: IFFT and GI insertion.

3.1.2.17 TMCC Fields

TMCC carries important information for the receiver, because it is based on the information transported within it that the receiver can be correctly configured with the ISDB-T transmitter parameters, like *Modulation*, *Code Rate*, number of *Hierarchical Layers*, *Time Interleaver* depth and so on. In this section I shortly describe the TMCC main parameters shown in Table 3.14.

Table 3.14: TMCC Bit Assignment.

B_0	Reference For Differential Modulation
B_1 to B_{16}	Synchronization word w0 = 0011010111101110, w1=1100101000010001
B_{17} to B_{19}	Segment Type: Synchronous (000) or differential (111)
B_{20} to B_{121}	TMCC Information
B_{121} to B_{203}	Parity Bits

TMCC is made of 204 bit differentially modulated, using the reference bit B_0,as explained in Section 3.1.2.16.3.

3.1.2.17.1 Synchronization Word

It is made of 16 bits and takes values W_0 or W_1. $W_1 = \overline{W_0}$, i.e. W_1 elements are obtained by inverting the bits of W_0. It is used to provide TMCC and *OFDM Frame* synchronization at the receiver side. The *Synchronization Word* is inverted at each *OFDM Frame* to prevent false synchronization lock.

3.1.2.17.2 Segment Type Identification

A 3-bit word to identify the type of modulation carried at each segment. 000 and 111 means differential or synchronous modulation respectively.

3.1.2.17.3 TMCC Information

It consists of a block with 102 bits. This portion of the TMCC supports the receiver to demodulate and decode several pieces of information: *System Identification*, *Transmission Parameter Switching*, *Emergency Alarm Broadcast Flag*, *Phase-Shift Correction for Terrestrial Audio Broadcasting* and, Current Information and Next Information for the *Hierarchical Layers*.

Table 3.15: TMCC Information.

B_{20} to B_{21}	System Identification	
B_{22} to B_{25}	Transmission Parameter Switching	
B_{26}	Emergency Alarm Broadcast Flag	
B_{27}		Partial Reception Flag
B_{28} to B_{40}	Current Information	Transmission Parameters Layer A
B_{41} to B_{53}		Transmission Parameters Layer B
B_{54} to B_{66}		Transmission Parameters Layer C
B_{67}		Partial Reception Flag
B_{68} to B_{80}	Next Information	Transmission Parameters Layer A
B_{81} to B_{93}		Transmission Parameters Layer B
B_{94} to B_{106}		Transmission Parameters Layer C
B_{107} to B_{109}	Phase-Shift Correction for Terrestrial Audio Broadcasting	
B_{110} to B_{121}	Reserved	

3.1.2.17.3.1 System Identification

Identifies if the current transmission system is either based on [1] for TV Broadcasting or ISDB-T_{SB} (sound broadcasting).

3.1.2.17.3.2 Transmission Parameter Switching

Indicates after how many *OFDM Frames* the transmission parameters will change and the values presented in the field *Next Information* shall be used by the receiver.

3.1.2.17.3.3 Emergency Alarm Broadcast Flag

As the name states, it is used to alert in case of emergency situations, e.g. natural disasters like tsunamis, hurricanes, earthquakes, etc.

3.1.2.17.3.4 Current and Next Information

These parameters are associated to *Transmission Parameter Switching* parameter, and indicate the change or not of the following parameters for each *Hierarchical Layer*: *Carrier Modulation Scheme*, *Convolutional Code Rate*, *Interleaveing Length*, *Number of Segments*. It also indicates if the system supports *Partial Reception* or not.

3.1.2.17.3.5 Phase-Shift Correction for Terrestrial Audio Broadcasting

It is used for audio broadcasting, which is not the aim of this work. This flag is ignored by the receiver proposed in this work.

3.1.2.17.3.6 Parity Bits

TMCC is error protected by means of a shortened code (184,102) of the difference cyclic code (273,191). The following is the *Generating Polynomial* used in this code:

$$g(x) = x^{82} + x^{77} + x^{76} + x^{71} + x^{67} + x^{66} + x^{56} + x^{52} + ...$$
$$...x^{48} + x^{40} + x^{36} + x^{34} + x^{24} + x^{22} + x^{18} + x^{10} + x^4 + 1.$$

This code allows the detection of TMCC over C/N ratios lower than those specified in the standard for decoding ISDB-T modulations.

3.2 DVB-S2 Standard Basics

DVB-S2 is the state-of-art ETSI standard for satellite broadcasting [77] [78]. It was developed by the Digital Video Broadcasting (DVB) project and it is the evolution of

DVB-S. It takes advantage of Low Density Parity Check Code (LDPC) concatenated with Bose-Chaudhuri-Hocquenghem (BCH) code, to achieve near Shannon limit performance. The key characteristics, that allows quasi-error free operation near (0.6 to 1.2 dB) Shannon limit are: 1) very large LDPC code block length (64800 bits for Normal Frame and 16200 bits for Short Frame); 2) large number of iterations on LDPC (around 50); and 3) the concatenation with BCH .

According to [77] and [79], DVB-S2 has been designed for applications such as: broadcast for High Definition Television (HDTV), iterative services for consumer applications, Digital TV distribution and news gathering, distribution of signal to terrestrial transmitters, and others. It achieves about 30% of capacity gain over DVB-S, under the same transmission conditions.

The standard has been specified around three key concepts: best transmission performance, total flexibility and reasonable receiver complexity. It supports Variable Coding Modulation (VCM), functionality that allows, 3/4, 4/5, 5/6, 8/9 and 9/10) to be used on a frame by frame different modulation (QPSK, 8-PSK, 16-QAM and 32-QAM) and error protection levels (1/4, 1/3, 2/5 ,1/2 , 3/5, 2/3 basis . Figure 3.40 shows the blocks of a DVB-S2 system.

DVB-S2 was specified to cope with any existing satellite transponder characteristics, with a large variety of spectrum efficiencies and Carrier-to-Noise ratio (C/N) requirements (it covers from -2.4 to 16 dB). In order to preserve compatibility with DVB-S, DVB-S2 has an optional compatibility mode. Moreover it may implement Adaptive Coding and Modulation (ACM), when used for interactive services. Due to all the previously mentioned advantages of DVB-S2, the implementation of DVB-S2 receivers is a challenging area of research.

3.2.1 DVB-S2 a Detailed View

Following, the blocks presented in Figure 3.40 are described.

3.2.1.1 Mode Adaptation

Mode Adaption subsystem is responsible for Input Interfacing, Input Stream Synchronization, Null-Packets deletion, CRC-8 encoding, input stream merging and input stream slicing. In addition, this section is responsible for adding baseband signaling (BB Signaling) named BBHEADER, used for Mode Adaptation purposes. The output of this section is the BBHEADER followed by the DATAFIELD.

3.2.1.1.1 Input Interface

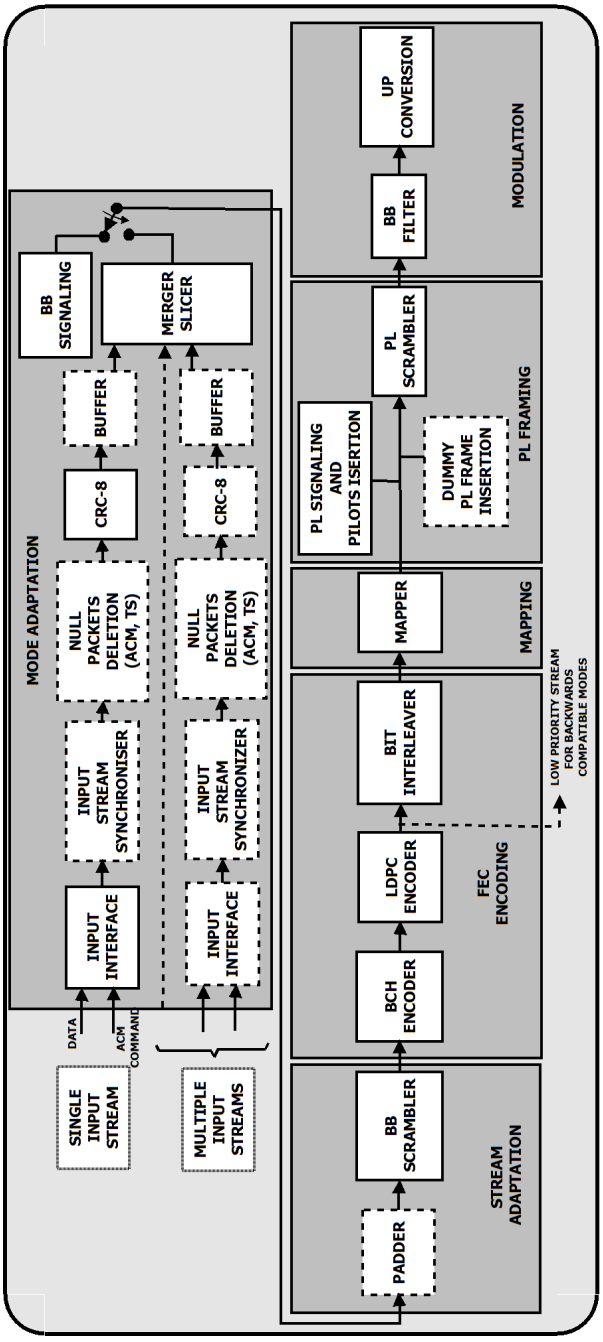


Figure 3.40: DVB-S2 transmitter blocks. The dashed blocks are non-mandatory for single stream services.

Input Interface subsystem supports four types of interface: MPEG Transport Stream (TS), Generic Stream, Adaptive Coding and Modulation (ACM) command, and Mode Adaptation. The first bit at the input will always be the MSB. The MPEG-TS is support User Packets (UP) with constant length. The UP length (UPL) is $188x8$ bits, the first byte being 47_{HEX}. The maximum UPL for the Generic Stream is 64 bit. ACM Command transports DVB-S2 parameters that shall be adopted by the transmitter. The only mandatory interface, for the case of video broadcasting, is the MPEG Transport Stream (TS).

3.2.1.1.2 Input Stream Synchronizer

As the focus of the implemented receiver is single TS, this block is not mandatory[1] for such a case (see [78]).

3.2.1.1.3 Null Packets Deletion

Null packets deletion is not mandatory for the broadcasting of single TS with no ACM. It consists basically of removing null packets which has Packet IDentifier (PID)= 8191_D. Later at the receiver they can be re-inserted in the correct position using the mechanisms described in the Annex D of [78],

3.2.1.1.4 CRC-8 Encoder

The useful part of the UP, with UPL not equal to 0, excluding the sync-byte shall be processed by the systematic 8-bit CRC encoder, which has the following generator polynomial and for each UP is initialized with zeros:

$$g(x) = x^8 + x^7 + x^6 + x^4 + x^2 + 1,$$

Given the input sequence $u(X)$, the CRC output is computed as:

$$CRC = remainder[x^8 u(x) : g(x)],$$

The CRC-8, computed over the UPL-8 bits of the UP, shall replace the sync byte of the next UP. The sync byte of the UP shall be placed in the SYNC field of the

[1] Despite several blocks are non-mandatory, if they are implemented they must must be compliant with the standard [78].

BBHEADER.

3.2.1.1.5 Merger/Slicer

The function of the Merger/Slicer for a DVB-S2 transmitter devoted to a single MPEG-TS stream is to create the DATAFIELD, which is composed of Data Field Length (DFL) bits such as:

$$K_{bch} - (10 \times 8) \geq DFL \leq 0,$$

where K_{bch} is the number of uncoded bits of the BCH encoder (see section 3.2.1.3.1).

As can be seen in Figure 3.41, the DATAFIELD can contain slices of MPEG-TS packets. The value of the field SYNCD within the BBHEADER is used to recompose the original sliced packet.

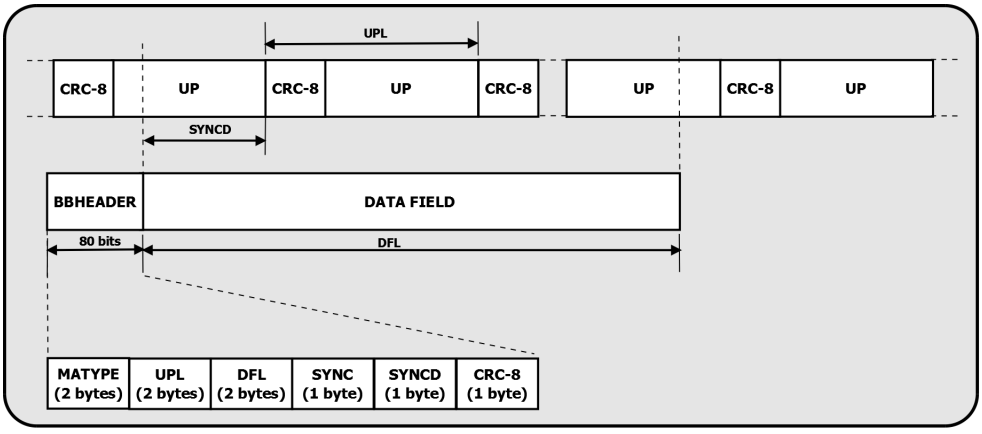


Figure 3.41: Output of Mode Adaptation Subsystem.

3.2.1.1.6 Base Band Header (BBHEADER)

The fixed length BBHEADER is shown in Figure 3.41. The MATYPE field identifies the input stream format: TS, General Stream, Single or Multiple Streams, Constant Coding and Modulation (CCM) or Adaptive Coding Modulation (ACM) and Adaptive Coding Modulation (ACM), Input Stream Synchronization Indicator,

Null Packet Deletion state (on/off), Roll-off Factor. BBHEADER also carriers UPL, DFL, SYNC, SYNCD and its own CRC-8. The CRC-8 of BBHEADER is computed over 72 bits, using the CRC-8 presented in Section 3.2.1.1.4

3.2.1.2 Stream Adaptation

The Stream Adaption subsystem functions are padding, to complete K_{bch} bits within a Base Band Frame (BBFRAME) , DUMMY PLFRAME[1],and scrambling. Nevertheless, in the Broadcast Service case, both padding and DUMMY PLFRAME are not necessary. Therefore, in that case, the only function of Stream Adaptation Subsystem is the scrambling.

3.2.1.2.1 Base Band (BB) Scrambling

BB Scrambling is performed to randomize the BBFRAME to avoid long runs of bits zeros and ones. The scrambling is performed over K_{bch} bits. The Polynomial Generator, whose initialization sequence is 100101010000000,of BB Scrambling is:

$$g(x) = x^{15} + x^{14} + 1.$$

The BBFRAME shall be *xored* with the scrambler output.

3.2.1.3 FEC Encoding Subsystem

FEC Encoding Subsystem performs Bose-Chaudhuri-Hocquenghem (BCH) encoding (inner code), LDPC encoding (outer code) and Bit Interleaving. The scrambled K_{bch} bits of each BBFRAME are the input of FEC Subsystem and FECFRAME is the output as can be seen in Figure 3.42. As shown in the that Figure, the BBFRAMEs, which contains K_{bch} bits, are first processed by the BCH encoder that produces $N_{bch} - K_{bch}$ CRC bits, namely BCHFEC. Following, the BBFRAME +BCHFEC are processed by the LDPC encoder and that produces $n_{ldpc} - k_{ldpc}$ CRC bits, namely LDPCFEC. The output of FECFRAME are $n_{ldpc}bits$. FECFRAME length can be 64800 bits, *normal* frame and 16200 for and *short* frame.

3.2.1.3.1 BCH Encoder

[1]DUMMY PLFRAME is a PLFRAME filled with zeros.

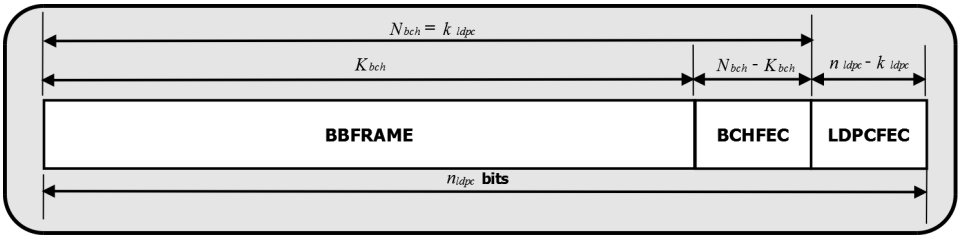


Figure 3.42: Data format before interleaving.

The BCH (N_{bch}, K_{bch}) encoder used in DVB-S2 has t-correcting capability according to Tables 3.16 and 3.17 for *normal* and *short* frames respectively. The number of uncoded bits (K_{bch}) at the input of BCH encoder as well as the BCH coded bits (N_{bch}), are also shown at those Tables. The BCH polynomials for *normal* and *short* frames are shown in Tables 3.18 and 3.19 respectively.

To obtain t-error correcting capability, for *normal* and *short* frames, the first t polynomials of Tables 3.16 and 3.17 shall be multiplied. Using this resulting polynomial it is straightforward to encode the bit block, by means of a LFSR.

Table 3.16: Coding Parameters for normal FECFRAME: $n_{ldpc} = 64800$.

LDPC CODE RATE	BCH UNCODED BLOCK K_{bch}	BCH CODED BLOCKS N_{bch} LDPC UNCODED BLOCK k_{ldpc}	BCH t-error correction	LDPC CODED BLOCK n_{ldpc}
1/4	16008	16200	12	64800
1/3	21408	21600	12	64800
2/5	25728	25920	12	64800
1/2	32208	32400	12	64800
3/5	38688	38880	12	64800
2/3	43040	43200	10	64800
3/4	48408	48600	12	64800
4/5	51648	51840	12	64800
5/6	53840	54000	10	64800
8/9	57472	57600	8	64800
9/10	58192	58320	8	64800

3.2.1.3.2 Low Density Parity Check (LDPC) Encoder

The output of BCH encoder, i.e k_{ldpc} bits, shall be systematically encoded by the LDPC encoder, using the parameters shown in Tables 3.16 and 3.17. The output of the LDPC encoder (n_{ldpc} bits) is 64800 or 16200 bits long for *normal* and *short* frames respectively. For Broadcasting Services the mandatory frame length, i.e. n_{ldpc} is 64800 bits.

During the DVB-S2 LDPC encoding procedure a Constant q is used. It is Code

Table 3.17: Coding Parameters for short FECFRAME: $n_{ldpc} = 16200$. The Capacity of t-error Corrections is 12 for all Code Rates.

LDPC CODE RATE	BCH UNCODED BLOCK K_{bch}	BCH CODED BLOCKS N_{bch} LDPC UNCODED BLOCK k_{ldpc}	EFFECTIVE LDPC RATE $K_{ldpc}/16200$	LDPC CODED BLOCK n_{ldpc}
1/4	3072	3240	1/5	16200
1/3	5232	5400	1/3	16200
2/5	6312	6480	2/5	16200
1/2	7032	7200	4/9	16200
3/5	9552	9720	3/5	16200
2/3	10632	10800	2/3	16200
3/4	11712	11880	11/15	16200
4/5	12432	12600	7/9	16200
5/6	13152	13320	37/45	16200
8/9	14232	14400	8/9	16200
9/10	NA	NA	NA	NA

Table 3.18: BCH Polynomials for *normal* Frames.

$g_1(x)$	$1 + x^2 + x^3 + x^5 + x^{16}$
$g_2(x)$	$1 + x + x^4 + x^5 + x^6 + x^8 + x^{16}$
$g_3(x)$	$1 + x^2 + x^3 + x^4 + x^5 + x^7 + x^8 + x^9 + x^{10} + x^{11} + x^{16}$
$g_4(x)$	$1 + x^2 + x^4 + x^6 + x^9 + x^{11} + x^{12} + x^{14} + x^{16}$
$g_5(x)$	$1 + x + x^2 + x^3 + x^5 + x^8 + x^9 + x^{10} + x^{11} + x^{12} + x^{16}$
$g_6(x)$	$1 + x^2 + x^4 + x^5 + x^7 + x^8 + x^9 + x^{10} + x^{12} + x^{13} + x^{14} + x^{15} + x^{16}$
$g_7(x)$	$1 + x^2 + x^5 + x^6 + x^8 + x^9 + x^{10} + x^{11} + x^{13} + x^{15} + x^{16}$
$g_8(x)$	$1 + x + x^2 + x^5 + x^6 + x^8 + x^9 + x^{12} + x^{13} + x^{14} + x^{16}$
$g_9(x)$	$1 + x^5 + x^7 + x^9 + x^{10} + x^{11} + x^{16}$
$g_10(x)$	$1 + x + x^2 + x^5 + x^7 + x^8 + x^{10} + x^{12} + x^{13} + x^{14} + x^{16}$
$g_11(x)$	$1 + x^2 + x^3 + x^5 + x^9 + x^{11} + x^{12} + x^{13} + x^{16}$
$g_12(x)$	$1 + x + x^5 + x^6 + x^7 + x^9 + x^{11} + x^{12} + x^{16}$

Table 3.19: BCH Polynomials for *short* Frames.

$g_1(x)$	$1 + x + x^3 + x^5 + x^{14}$
$g_2(x)$	$1 + x^6 + x^8 + x^{11} + x^{14}$
$g_3(x)$	$1 + x + x^2 + x^6 + x^9 + x^{10} + x^{14}$
$g_4(x)$	$1 + x^4 + x^7 + x^8 + x^{10} + x^{12} + x^{14}$
$g_5(x)$	$1 + x^2 + x^4 + x^6 + x^8 + x^9 + x^{11} + x^{13} + x^{14}$
$g_6(x)$	$1 + x^3 + x^7 + x^8 + x^9 + x^{13} + x^{14}$
$g_7(x)$	$1 + x^2 + x^5 + x^6 + x^7 + x^{10} + x^{11} + x^{13} + x^{14}$
$g_8(x)$	$1 + x^5 + x^8 + x^9 + x^{10} + x^{11} + x^{14}$
$g_9(x)$	$1 + x + x^2 + x^3 + x^9 + x^{10} + x^{14}$
$g_10(x)$	$1 + x^3 + x^6 + x^9 + x^{11} + x^{12} + x^{14}$
$g_11(x)$	$1 + x^4 + x^{11} + x^{12} + x^{14}$
$g_12(x)$	$1 + x + x^2 + x^3 + x^5 + x^6 + x^7 + x^8 + x^{10} + x^{13} + x^{14}$

Rate dependent and defined as follows:

$$q = \frac{(n_{ldpc} - k_{ldpc})}{360} = \frac{n_{ldpc}}{360}\left(1 - \frac{k_{ldpc}}{n_{ldpc}}\right),$$

where, $\frac{k_{ldpc}}{nldpc}$ is the LDPC Code Rate and $(n_{ldpc} - k_{ldpc})$ is the parity check length. Tables 3.20 and 3.21 show the values of q for *normal* and *short* frames.

Table 3.20: values of q for *normal* frames.

Code Rate	q
1/4	135
1/3	120
2/5	108
1/2	90
3/5	72
2/3	60
3/4	45
4/5	36
5/6	30
8/9	20
9/10	18

Table 3.21: values of q for *short* frames.

Code Rate	q
1/4	135
1/3	120
2/5	108
1/2	90
3/5	72
2/3	60
3/4	45
4/5	36
5/6	30
8/9	20

Next I will show, as an example, how to encode the DVB-S2 *normal* frame for LDPC Code Rate of 1/2 ($q = 90$). The procedure can be easily extended to the other Code Rates for *normal* as well as *short* frames.

Given, the information block $i_0, i_1, i_2 \ldots i_{k_{ldpc}-1}$ containing k_{ldpc} bits, the encoder shall create $n_{ldpc} - k_{ldpc}$ parity bits, forming a vector $p_0, p_1, p_2 \ldots p_{n_{ldpc}-k_{ldpc}-1}$, following the procedure described below [78]:

1. Initialize $p_0, p_1, p_2 \ldots p_{n_{ldpc}-k_{ldpc}-1} = 0$;

2. Accumulate the first bit i_0 at parity address specified at first row of Table B.4 in GF(2) as follows:

$$\begin{aligned} p_{54} &= p_{54} \oplus i_0 \\ p_{9318} &= p_{9318} \oplus i_0 \\ p_{14392} &= p_{14392} \oplus i_0 \\ p_{27561} &= p_{27561} \oplus i_0 \\ p_{26909} &= p_{26909} \oplus i_0 \\ p_{10219} &= p_{10219} \oplus i_0 \\ p_{2534} &= p_{2534} \oplus i_0 \\ p_{8597} &= p_{8597} \oplus i_0 \end{aligned}$$

3. For the next 359 information bits i_m,$m = 1, 2, 3 \ldots, 359$, i.e. $i_1, i_2, i_3, \ldots, i_{359}$ accumulate i_m at parity bit address:

$$\{x + m \bmod 360 \times q\} \bmod (n_{ldpc} - k_{ldpc}).$$

   x corresponds to the address of the parity bit accumulator of the first bit i_0, i.e. $x = 54, 9318, 14392, \ldots, 8597$.

   The accumulation for the information bit i_1 is:

$$\begin{aligned} p_{144} &= p_{144} \oplus i_1 \\ p_{9408} &= p_{9408} \oplus i_1 \\ p_{14482} &= p_{14482} \oplus i_1 \\ p_{27651} &= p_{27651} \oplus i_1 \\ p_{26999} &= p_{26999} \oplus i_1 \\ p_{10309} &= p_{10309} \oplus i_1 \\ p_{2624} &= p_{2624} \oplus i_1 \\ p_{8687} &= p_{8687} \oplus i_1 \end{aligned}$$

4. For the information bit i_{360} the addresses of parity bit accumulator are those in the second row of Table B.4;

5. For the next 359 information bits i.e. i_m,$m = 361, 362, 363 \ldots, 719$, accumulate i_m at parity bit address:

$$\{x + m \bmod 360 \times q\} \bmod (n_{ldpc} - k_{ldpc}).$$

   x corresponds to the addresses of the parity bit accumulator of the information bit i_{360}, which are the values in the second row of Table B.4.

6. For every new group of 360 information bits, a new row of Table B.4 is used to find the parity bit accumulators addresses.

7. After computing all parity bit accumulators, the final parity bits are computed as:

$$p_i = p_i \oplus p_{i-1} \quad i = 1, 2, 3, \ldots, n_{ldpc} - k_{ldpc} - 1.$$

p_0 is equals to its original value.

The Tables containing the addresses of the bit accumulators,for all valid code rates, and $n_{ldpc} = 64800$ and $n_{ldpc} = 16200$ are presented within Annex B of [78].

3.2.1.3.3 Bit Interleaver

DVB-S2 standard only applies bit interleaving on 8-PSK, 16-APSK and 32-APSK modulations. The interleaving is applied in a block fashion. As can be seen in Figure 3.43 the LDPC encoded data, i.e. FECFRAME, is serially written in the columns and read out in a row-wise fashion.The MSB of BBHEADER is read first for all cases except for 8-PSK with Code Rate 3/5, where MSB of BBHEADER is read out thirdly. The number of columns and rows varies according to modulation and N_{ldpc} and are shown in Table 3.22

Table 3.22: Bit Interleaver Parameters.

Modulation	Rows for $N_{ldpc} = 64800$	Rows for $N_{ldpc} = 16200$	Columns
8-PSK	21600	5400	3
16-ASPK	16200	4050	4
32-APSK	12960	3240	5

3.2.1.4 Bit Mapping

The interleaved FECFRAME shall be modulated using QPSK, 8PSK, 16APSK or 32APSK, to form a XFECFRAME with length $64800/\eta_{MOD}$ or $16200/\eta_{MOD}$, where $\eta_{MOD} = 2, 3, 4 or 5$ is the number of bits per modulation symbols. QPSK and 8PSK have radius equals to 1. Despite the fact that 16APSK ($4 + 12$ ASPK) and 32APSK

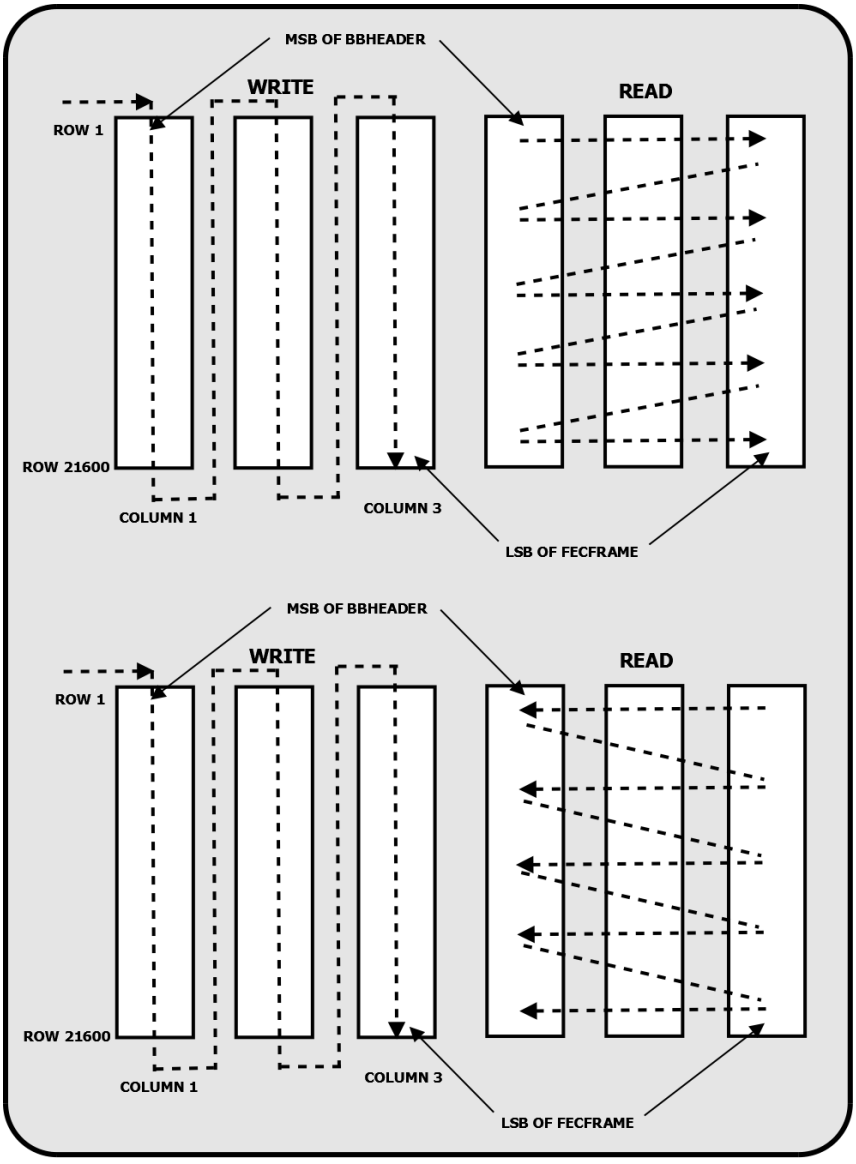


Figure 3.43: Example of bit interleaving write and reading schemes for 8-PSK and *Normal* Frame: (top) all rates except 3/5, (bottom) Code Rate 3/5.

(4 + 12 + 15 APSK) targets professional application, they can also be used for broadcasting. They have variable constellation radius and are intended for linear and quasi-linear channels. Nevertheless, they can be used in non-linear channels if pre-distortion is applied in the up-link.

3.2.1.4.1 16APSK

16APSK (4 + 12 ASPK) is composed as shown in Figure 3.44, MSB first. It is made of 2 concentric radius R_1 and R_2 , where the ratio $\gamma = R_2/R_1$, shall be according to Table 3.23. According to [80], two are the values admitted for the constellation amplitude without pre-distortion:

- $4[R_1]^2 + 12[R_2]^2 = 16$, the average signal energy is 1;
- $R_2 = 1$.

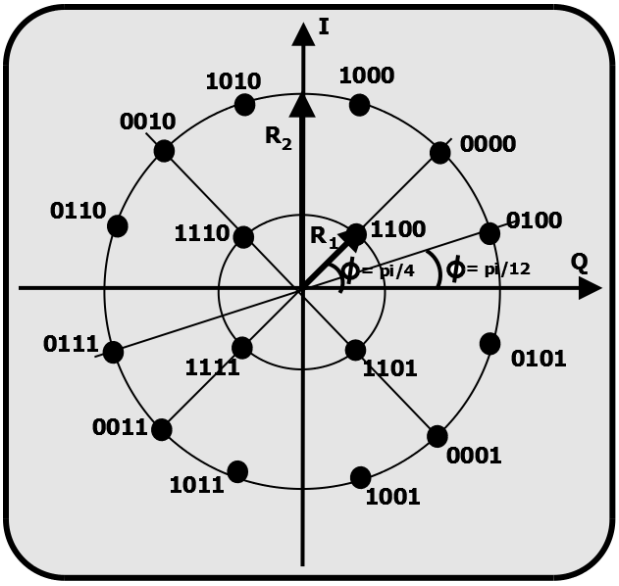


Figure 3.44: 16APSK constellation.

3.2.1.4.2 32APSK

Table 3.23: 16APSK Optimum Constellation Radius Ratio γ, for Linear Channel.

Code Rate	Modulation/Coding Spectral Efficiency	γ
2/3	2,66	3,15
3/4	2,99	2,85
4/5	3,19	2,75
5/6	3,32	2,70
8/9	3,55	2,60
9/10	3,59	2,57

32APSK $(4 + 12 + 15$ APSK) is composed as shown in Figure 3.45, MSB first. It is made of 3 concentric radius R_1, R_2 and R_3, where the ratios $\gamma_1 = R_2/R_1$ and $\gamma_2 = R_3/R_1$, shall be according to Table 3.24. According to [80], two are the values admitted for the constellation amplitude without pre-distortion:

- $4[R_1]^2 + 3[R_2]^2 + 4[R_3]^2 = 8$, the average signal energy is 1;
- $R_3 = 1$.

Table 3.24: 32APSK Optimum Constellation Radius Ratios γ_1 and γ_2, for Linear Channel.

Code Rate	Modulation/Coding Spectral Efficiency	γ_1	γ_2
2/3	3,74	2,84	5,27
3/4	3,99	2,72	4,87
4/5	4,15	2,64	4,64
5/6	4,43	2,54	4,33
8/9	4,49	2,53	3,30

3.2.1.5 Physical Layer (PL) Framing

PL Framing subsystem is responsible for generating the Physical Layer Frame (PLFRAME). PLFRAME structure is shown in Figure 3.46. There are four tasks performed within this subsystem:

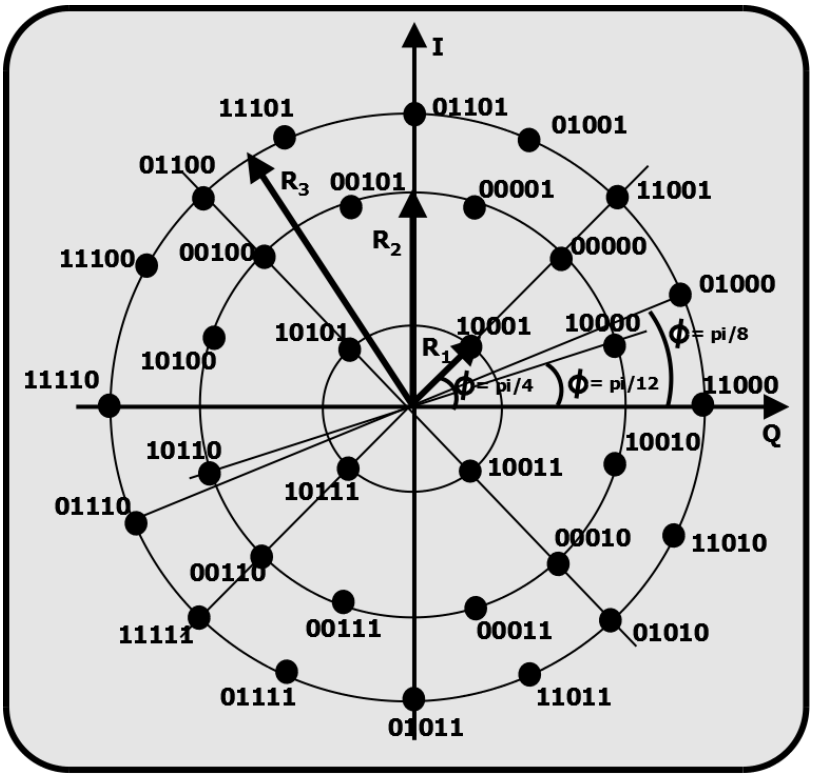


Figure 3.45: 32APSK constellation.

1. Dummy PL Frame generation, when there is no XFECFRAME at the subsystem input to be processed and transmitted;
2. Slicing of the XFECFRAME into an integer number S of constant length Slots with length M = 90 symbols. S is defined according to Table 3.25;
3. Physical Layer Header (PLHEADER) generation and insertion, to configure the receiver;
4. Pilot Block insertion (in case pilots are required), every 16 Slots, for synchronization and equalization purposes. Each pilot block contains P = 36 pilot symbols;
5. Randomization of the I&Q modulated symbol using a Physical Layer Scrambler (PL SCRAMBLER).

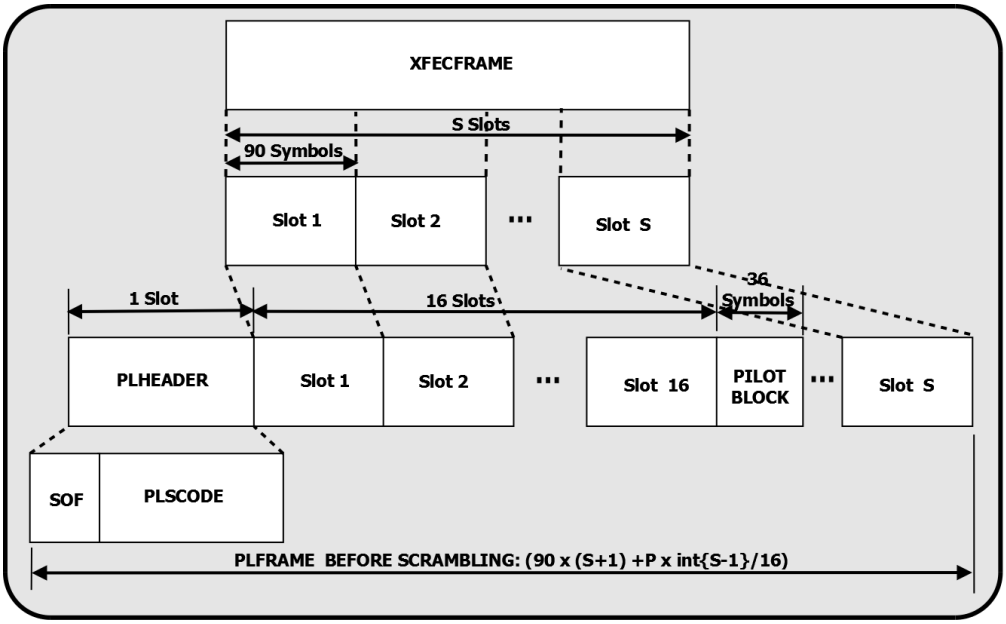


Figure 3.46: PLFRAME contents. int{.} is the Integer Function.

3.2.1.5.1 Dummy PLFRAME Insertion

A Dummy PLFRAME is made of a PLHEADER and 36 Slots of un-modulated carriers with $I = Q = 1/\sqrt{2}$. This block is not relevant to broadcasting systems with single transport stream.

Table 3.25: Value of S According to the Modulation and Frame Length.

η_{MOD}	$n_{ldpc} = 64800$ Normal Frame		$n_{ldpc} = 16200$ Short Frame	
	S	η % no-pilot	S	η % no-pilot
2	360	99,72	90	98,90
3	240	99,59	60	98,36
4	180	99,45	45	97,83
5	144	99,31	36	97,30

3.2.1.5.2 PL Signaling

The 90-symbols PLHEADER is made of the following fields:

- 26 symbols identifying the Start Of Frame (SOF). The sequence is $18D2E82_{HEX}$ (01100011010010111010000010) MSB first;
- 64 symbols containing the encoded Physical Layer Signaling (PLS), named PLSCODE, that is encoded by a non-systematic binary code of length 64 and dimension 7 with minimum distance $d_{min} = 32$. The PLS contains 7 bits, and is made of two fields MODCOD and TYPE, described Sections 3.2.1.5.2.2 and 3.2.1.5.2.3.

The 90 symbols shall be modulated using $\pi/2$BPSK using the following rule:

$$I_{2i-1} = Q_{2i-1} = \left(1/\sqrt{(2)}\right)\left(1 - 2\mathrm{y}_{2i-1}\right)$$
$$I_{2i} = -Q_{2i} = -\left(1/\sqrt{(2)}\right)\left(1 - 2\mathrm{y}_{2i}\right)$$

Where, y_i for $i = 1 \ldots 90$ is the sequence representing the encoded PLHEHEADER.

3.2.1.5.2.1 Start-Of-Frame (SOF) Field

The SOF sequence is $18D2E82_{HEX}$ (01100011010010111010000010) MSB first;

3.2.1.5.2.2 MODCOD Field

MODOCD is a 5 bit sequence which describes the Modulation and the XFECFRAME Code Rate according to Table 3.26.

Table 3.26: MODCOD Values, According to Baseband Modulation and Code Rate.

Mode	MODCOD	Mode	MODCOD	Mode	MODCOD	Mode	MODCOD
QPSK 1/2	1_D	QPSK 5/6	9_D	8PSK 9/10	17_D	32APSK 4/5	25_D
QPSK 1/3	2_D	QPSK 8/9	10_D	16APSK 2/3	18_D	32APSK 5/6	26_D
QPSK 2/5	3_D	QPSK 9/10	11_D	16APSK 3/4	19_D	32APSK 8/9	27_D
QPSK 1/2	4_D	8PSK 3/5	12_D	16APSK 4/5	20_D	32APSK 9/10	28_D
QPSK 3/5	5_D	8PSK 2/3	13_D	16APSK 5/6	21_D	Reserved	29_D
QPSK 2/3	6_D	8PSK 3/4	14_D	16APSK 8/9	22_D	Reserved	30_D
QPSK 3/4	7_D	8PSK 5/6	15_D	16APSK 9/10	23_D	Reserved	31_D
QPSK 4/5	8_D	8PSK 8/9	16_D	32APSK 3/4	24_D	DUMMY PLFRAME	32_D

3.2.1.5.2.3 TYPE Field

The TYPE field contains 2 bits to identify the FECFRAME length, i.e. 64800 or 16200, and the presence of pilots or not.

3.2.1.5.2.4 PLS Code

The PLS encoding is performed as depicted in Figure 3.47. 6 out of the 7-bits PLS are encoded by the generator matrix shown in Figure 3.48. The MSB of PLS, i.e. bit number 1, is multiplied by row 1, the second MSB by row 2 and so on, until bit 6 which is the MSB of TYPE Field. The encoding construction guarantees that each odd bit is either equal to the previous or the opposite. Before being transmitted, the 64-bit of the output of the PLS encoder, i.e. $y_1, y_1 \oplus b_7, y_2, y_2 \oplus b_7, \ldots, y_3 2, y_3 2 \oplus b_7$, are scrambled by the following sequence:

$$0111000110011101100000111100100101010011010000100010110111111010.$$

3.2.1.5.3 Pilots Insertion

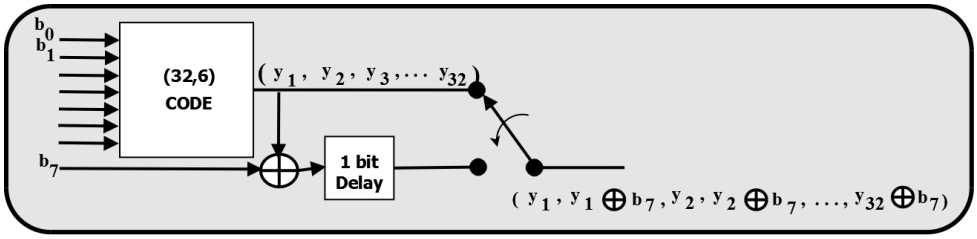


Figure 3.47: PLS encoding.

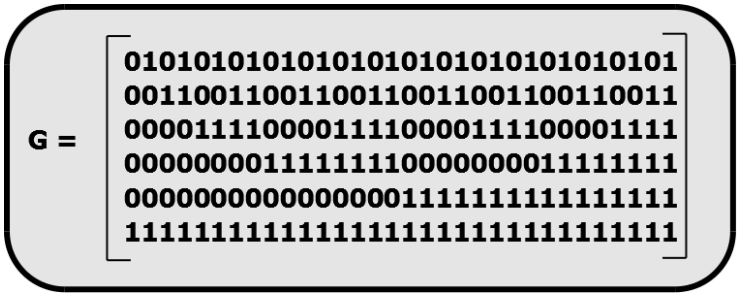


Figure 3.48: Generator matrix for the MODCOD Linear Block Encoder.

As depicted in Figure 3.46, when a pilot is present, the PLFRAME contains a number of 36-pilot grouping, embedded among the data slots. The 36 pilots are made of un-modulated I&Q symbols, with I=Q=1/$sqrt(2)$.

3.2.1.5.4 Physical Layer Scrambling

After pilot insertion, the PLFRAME I&Q symbols must be scrambled, for energy dispersal purposes, by a complex randomizer named PL Scrambler. The PL Scrambler shall be reset at the end of PLHEADER (and not applied to it) as shown in Figure 3.49.

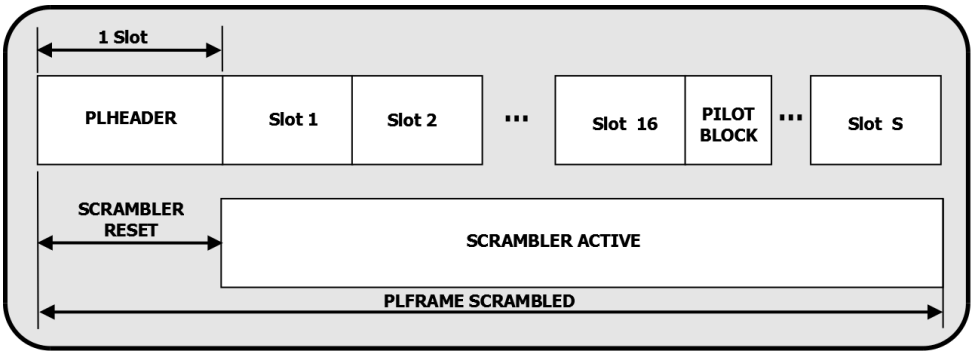


Figure 3.49: PLFRAME scrambling.

The PL Scrambler is constructed according to Figure 3.50. In figure, the LFSR on the top is initialized with $x(0) = 1$ and the remaining 17 registers with zeros. The LFSR in the bottom is initialized with 18 ones. The binary Gold sequence $z_n(i)$, for $n = 0, 1, 2, \ldots, 2^{18} - 2$ and $i = 0, 1, 2, \ldots, 2^{18} - 2$, is converted into a integer sequence $\mathrm{R}_n(i)$, with values 0,1,2 or 3. Finally, the complex PL Scrambler sequence, containing $\mathrm{C_I}(i) + j\mathrm{C_Q}(i)$ complex values is obtained as:

$$\mathrm{C_I}(i) + j\mathrm{C_Q}(i) = \exp(j\mathrm{R}_n(i)\pi/2)$$

Where $i = 0, 1, 2, \ldots, 66419$. Recall that $\mathrm{R}_n(i)$ = 0,1,2 or 3.

3.2.1.6 Base Band Shaping

Following Figure 3.40, the last block before up-conversion is the baseband shaping, which aims at shaping the transmitted spectrum, by means of a Square Root Rised

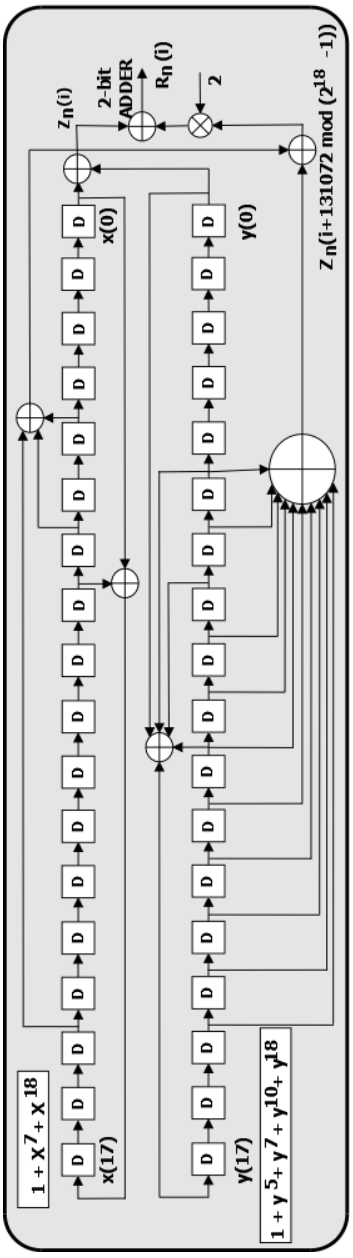


Figure 3.50: PL Scrambler for n= 0, which is the value for broadcasting services.

Cosine filter. The roll-off factors, α, defined in the standard [78] for that filter are 0.20, 0.25 and 0.35. The spectral mask regarding frequency normalized to system the Sample Rate is presented in the ANNEX A of [78].

3.2.2 DVB-S2X: the Future of DVB-S2

Despite its advantages regarding DVB-S and have been specified about 10 years ago, DVB-S2 is still not vastly used worldwide. A search on DVB-S2 usage in [81] and [82]clearly shows that. Furthermore, most of the satellite links using DVB-S2 makes use of 8-PSK modulation. Therefore there is still room for the use of DVB-S2 standard and its current configuration options. Even though, in 2014, DVB-S2X, which is an amendment to DVB-S2, was released by ETSI [83] as ETSI EN 302 307-2.

According to [83], DVB-S2X main amendments to DVB-S2 are:

- Extended operational range to cover emerging markets such as mobile applications;
- Support of C/N down to -10 dB for mobile applications;
- Roll-off options of 5% and 10%;
- New constellation options for linear and non-linear channels;
- A finer gradation and extension of number of modulation and coding modes;
- Additional scrambling options for critical co-channel interference situations;
- Channel bonding of up to 3 channels;
- Super-frame option.

Further details on this amendment can be found in the draft version of the standard, [84], published by the DVB Project.

Chapter 4

The ISDB-T Receiver Design Methodology

Aiming achieving an small chip area, which would lead to a low cost chip, and given the funding limitations, the main requirements for the design of the receiver proposed in this work were defined as:

- Low complexity receiver with low gate count (that would lead to an small chip area) ;
- Design targets are fixed-reception and low mobility devices.

With those two requirements in hands, the next task was to define a methodology to be followed in such a way that all project members could have an overall view of the project evolution. An advantage of defining the methodology at an early stage of the project is to make it possible to create a logical order to execute the project tasks, *i.e.* model designing, RTL coding, buying the IPs that would not be designed in the project framework, buying equipments for testing and prototyping, etc.

4.1 Overview of the Methodology

The methodology adopted in this project has as its main goal the implementation of an ASIC, and use the FPGA as a prototyping step previous to ASIC design flow. It is strongly dependent on the receiver architecture definition and the implementation of the receiver reference model and algorithm exploration. Without those steps and

the several refinements loops, this methodology would be reduced to a conventional ASIC design flow.

This chapter provides an overview of the receiver *Design Flow*, from the algorithm definition to the VLSI tests results. In addition, a short description of each design step is provided. An overall view of the *Design Flow* can be seen in the flow diagram presented in Figure 4.1.

The initial task in this project was to design an ISDB-T transmitter model using Matlab, followed by the impairments (sampling, border, frequency) modeling. At that phase, it was possible to accomplish an initial exploration on algorithms and receiver architecture. Correct impairments generation is of paramount importance to design a receiver with a realistic behavior. The transmitter and impairments generation models were used to generate stimulus for the receiver blocks design using Matlab and VHDL. Once the blocks were designed using independent platform VHDL coding and verified using the stimuli from Matlab/Octave, they were synthesized and physically tested using Altera's FPGA. After the physical verification of each receiver block they were progressively integrated and verified mainly using the free tools GHDL [31] and GTKWave [32], followed by a synthesis targeting FPGA and a physical validation, again in FPGA. This procedure was followed until the entire receiver was completely integrated and ready for System Tests. Once extensive System Tests were done in the FPGA prototype and the performance of the receiver was close to those values expected in the standard (considering implementation losses), the receiver design implementation could go to the ASIC design flow phase. A simplified view of this process can be seen in Figure 4.1. Several other tasks were also performed such as DFT chain insertion and Memory Built In Self Test (M-BIST). During ASIC design flow phase several optimizations were performed, e.g. timing, area and power reduction. Corrections on VHDL codes were also done in order to achieve ASIC flow requirements. Every time a block was optimized/corrected a new physical test on FPGA was done to validate the entire receiver behavior. Once the ASIC requirements were met, physical IC was implemented using Global Foundry 65 nm CMOS. Upon receiving encapsulated physical IC, several tests were done in order to characterize the electrical and System Level behavior.

4.2 Receiver Design Steps

For a clear understanding of the Methodology used to design the receiver, this section provides a short description of each step presented in Figure 4.1, which delivers a simplified view of the entire receiver designer process.

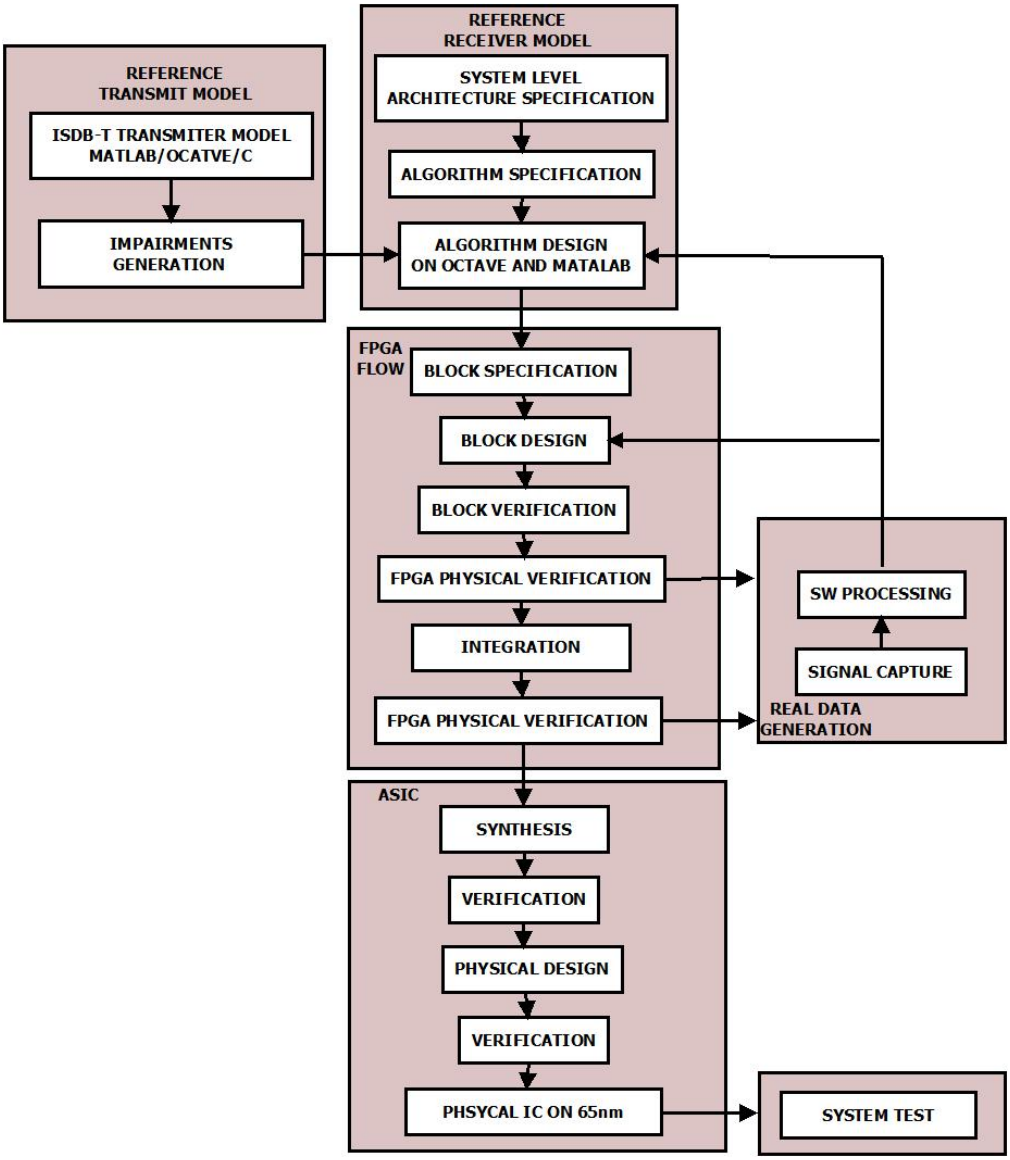


Figure 4.1: A simplified view of the ISDB-T receiver design flow.

4.2.1 Transmitter Model and Impairments Generation

In order to model the receiver architecture, a transmitter model compliant to the ISDB-T standard was built using Matlab. In addition channel models and radio impairments were also modeled.

4.2.1.1 ISDB-T Transmitter Model

This transmitter model contains all ISDB-T standard blocks plus up-conversion. The ISDB-T blocks are shown in Figure 3.1 and the explanation about the function of each block is presented in Section 3.1.2. It can work in baseband, IF, or RF mode. The ISDB-T compliant signal generated from the output of each transmitter block or at the output of the complete transmitter were used as reference for:

1. Generating system reference performance over wireless channels or AWGN;
2. Receiver blocks and algorithms modeling;
3. Golden inputs for the VHDL design;
4. Golden inputs for physical tests in FPGA of each individual receiver block ;
5. Golden inputs for physical tests during receiver tests in FPGA;
6. Golden inputs used during the Verification process in the ASIC design flow.

Before being used, the signals generated from the transmitter model were quantized and converted to bit stream when convenient.

4.2.1.2 Impairments Generation

Impairments modeling is of paramount importance when modeling and designing a proof-of-concept or commercial receiver, because if the impairments are not taken into account during the algorithm and architecture definition the implemented receiver (in FPGA and/or ASIC) could not work or present performance at non acceptable levels, when subject to real world impairments. This subject was widely addressed in Chapter 2. Impairments that can be generated includes wireless channels effects, AWGN, Carrier Frequency Error, Sampling Clock Error, Phase Noise and DC Offset.

4.2.2 Receiver Architecture and Model

Despite being the motivation of this thesis, it is worth highlighting that it would not be possible to define and model the receiver architecture without a deep understanding of the ISDB-T transmitter model and the radio impairments that could affect the ISDB-T receiver.

The receiver architecture definition and modeling were split in two main parts: the inner receiver (herein called *Signal Processing*) and the outer receiver (herein called *Data Processing*). *Signal Processing* and *Data Processing* are made of several blocks that sometimes could be modeled and implemented in an independent manner, *i.e.* blocks can be modeled and designed without paying special attention to the surrounding blocks behavior. In other words, the golden input and output from the transmitter model suffice to support the block modeling/implementation. In this section I describe some important tasks I have done or taken part into, before I proposed and defined the final receiver architecture and start the receiver modeling.

4.2.2.1 System Level Architecture Specification

The focus of this thesis is to propose a suitable receiver architecture that:

1. Obeys the initial requirements (fixed-reception and low mobility, and low complexity);
2. Overcomes the channel, radio and other hardware impairments to have acceptable performance when compared to other commercial solutions;
3. Could evolve to use advanced algorithms and blocks (e.g. for mobility) without significant changes;
4. Is suitable to be implemented in FPGA and VLSI.

Before defining the final receiver architecture and modeling it, several theoretical themes were subject of studies, some of which were presented within previous chapters. In parallel to the theoretical studies, some practical exploration was also done in order to prove the feasibility of the project, e.g., I implemented a CORDIC algorithm for DDS generation and also a simple FIR filter to be used as Low Pass Filter, both in VHDL. These elements were used to create a simple IF to Zero conversion block, that was synthesized in FPGA (Stratix II -EP2S180 Kit, from Altera). The input signal was provided by an ISDB-T IF generator. The signal coming from the generator was injected in the ADC at the FPGA kit and processed by the DDS plus Low Pass blocks. The outputs of these blocks were captured and used as real input for the Matlab simulations, in which initial exploration of synchronization and equalization

algorithms was done without the need of building the ISDB-T transmitter model. These were very simple exploration tasks, if compared to the final architecture or the final receiver implementation.

Besides the theoretical studies and initial implementation explorations, several other activities were essential to define the ultimate architecture to be prototyped in FPGA and implemented in ASIC. It is worth to recall that I was the system architect of the receiver and took part of them. Following a short list containing some of the activities is presented:

- Study of commercial ISDB-T receivers for benchmarking;
- Search in the technical and scientific literature for ISDB-T receiver implementations and algorithms for OFDM receivers;
- Definition of commercial tuner platforms to be used in the test environment;
- Search and definition of tuner ICs to be candidates for future commercial applications using the implemented ISDB-T digital receivers;
- Study of the characteristics of the commercial tuners, in order to define the main impairments generated by those tuners;
- Definition of the FPGA integration and prototype platform ;
- Specification of the peripherals to be used in the integration/prototype platform to send the MPEG-TS out the FPGA and to capture signal from tuner via ADC;
- Definition of the laboratory setup to test the prototype;
- Creation of test scenarios.

4.2.2.2 Algorithm and Block Design In Octave and Matlab

Once defined the impairments to be tackled and the overall receiver architecture, the next step was to define the candidate algorithms and implement them in Matlab or Octave (Matlab compliant code). This phase was strongly based on literature review and algorithm exploration. The first partition to be explored was the *Inner Receiver*, followed by the *Outer Receiver*. The architecture of the algorithms designed in Matlab are as close as possible to that being implemented in VHDL, in order to provide the VHDL designers with a code that would serve as an algorithm specification. Nevertheless, it is worth to mention that the outputs of this step were two: the Matlab code and a high level algorithm specification document. With these in hands and due to the closeness I worked with the VHDL designers, we could speed up the implementation in VHDL, by exchanging information and solving issues in a fast way.

In order to explore the algorithms for synchronization and to tackle other impairments, I built a simple ISDB-T transmitter model containing the Frame Structure of ISDB-T, i.e. a transmitter containing random data (QPK, 16-QAM and 64-QAM) and the pilot structure complaint with ISDB-T standard. With this simple model, it was possible to explore and define the algorithms for synchronization and equalization. Later, this simple transmitter evolved to encompass all ISDB-T standard blocks, which made possible to implement the *Outer Receiver* blocks.

At the end of this phase we had a complete ISDB-T transmitter and the entire ISDB-T Digital Receiver model working. In addition several high level documents containing the blocks and algorithms specification were delivered to the designers, in order to design the blocks in VHDL. All documents and codes were stored into an SVN [85] data base.

With the complete model containing transmitter and receiver, it was possible to explore auxiliary blocks and algorithms, as SNR estimators, and to generate golden inputs and outputs to be used during the implementation in VHDL and for physical validation in FPGA.

4.2.3 Digital Front-End Design and FPGA Prototyping

In this step the explored algorithms and system architecture were mapped into synthesizable VHDL code, verified and physically tested in FPGA, but initially they were specified by means of VHDL Implementation specification documents. The inputs for this step came from the blocks/algorithms modeled in Matlab and the high level document specification, describing the blocks and the algorithms.

4.2.3.1 Block Specification

It consists of a document created by the block designers, that is used as reference for the digital design and for the verification engineer to design the verification tests. It contains a detailed explanation about the block and sub-blocks functioning, the input and output interfaces definition, detailed explanation about each input/output interface and the dependencies regarding other blocks.

4.2.3.2 Block Design

The project design flow was meant to arrive at this point with the maximum of information in terms of documents and models, to make the block design in VHDL fast and accurate. In other words, it was either expected that the re-design of the blocks would not occur, or would be minimized, after this step. It basically consists

of designing independent platform VHDL blocks, using the input documents (system specification and specification for VHDL implementation) and respecting the code style adopted for the project. Two teams were assembled to implement the blocks. The first group was responsible for the Inner Receiver, and the second for the Outer receiver.

4.2.3.3 Block Verification

The blocks designed in VHDL had to be tested in SW before going to the next step which was the physical design (synthesis and tests in FPGA). This means the blocks were first tested using VHDL simulator and scripts in order to guarantee the correct functioning and that the block was synthesizable. An extensive battery of tests were conducted to guarantee the coverage of the code and the functioning. Each block was tested independently and after having a certain number of blocks verified the integration of the receiver started. As each new block was ready it could be added to the integrated receiver, that was tested in the same way of the block, that is, using golden inputs and outputs.

4.2.3.4 FPGA Physical Verification of Blocks

After SW verification of the VHDL blocks, they could be synthesized and validated in FPGA, using golden input and outputs generated from the transmitter and/or receiver models, or from the real data[1]. In short this process, called *Physical Validation* consists of an FPGA platform containing a NIOS[2] processor, two VHDL blocks called Driver and Monitor, the block under test and a SW running into the NIOS processor. The golden input, generated from files (or real data capture, in those cases mentioned in *footnote 1*) and stored into NIOS memory, or from real time generation within NIOS (in the case of simple blocks, e.g. a simple PRBS), feeds the Driver, that adapts the Avalon protocol to the protocol used by the block. The Driver injects the data in the block under test, then the block send its output to the Monitor, that adapts the block output protocol to the Avalon protocol, then NIOS processor can read the block output via Monitor and compare it to the expected output. This process was done for each receiver block.

As mentioned in the previous section, the receiver started to be integrated in SW after the integrated blocks passed the SW-based test. After the individual validation of the SW-integrated blocks in HW, and the validation of the integration in SW, the integrated blocks could be physically validated in HW in conjunct (i.e integration). This process is very dynamic and aims accelerating the integration process while minimizing the risk of integration not working if it is done at once at the end of

[1] Real data was mostly used during the integration and debugging phase.
[2] Embedded processor from Altera.

the individual validation of each block in SW and HW. This step is one of the most important in the receiver design and involves system architect/algorithm designer, designers and verification engineers.

4.2.3.5 Receiver Integration

There were two types of integration in the receiver project. The SW integration that is related to the integration of VHDL blocks in SW and its verification was done by means of simulations using GHDL/Gtkwave, or other non-open source tools and scripts, and the HW integration that we call Physical Test, that was done in the VHDL synthesized in FPGA. The integration could be partial or total. In the case of partial integration, as each block was tested in SW and HW, and after having a number of blocks that could be logically connected, they could be integrated and form a kind of sub-partition of the receiver.

In this section, the 'entire receiver integration' means when all sub partitions previously integrated in HW and/or SW, and the other remaining blocks were put together, obeying a logical sequence to form the entire receiver. This task was done for the partial integration case, *i.e.* it was done in SW and then in HW. The golden inputs were generated from the transmitter model or from real data capture. The gold outputs were generated from the reference receiver architecture. At the end of the integration process, the ISDB-T receiver was complete and could validate in FPGA through system tests.

4.2.3.6 FPGA Physical Verification of the Receiver

Once the full receiver was integrated and the test environment was setup, several system tests were done in order to prove the correct receiver functioning. They consists of configuration tests and performance tests. The test environment will be shown in Chapter 6. The configuration tests consisted of accessing (read/write) the receiver parameters and signals via register bank using USB or I2C interface. Examples of parameters are: Mode, Modulation, Code Rate. Those parameters can be forced (via write command) to a specific value, for test and debugging purposes. Example of signal that can be read, is the signal power after ADC conversion. The register bank can also be used to bypass blocks and force signals to be sent to the USB output. During this phase blocks debug occurred and new features were tested as well.

After debugging, receiver improvements (as bit length exploration and the addition of Soft Demapping, that was initially implemented in a hard decision fashion, for instance), validation of the receiver behavior and comparisons with the ISDB-T BER performance specification for AWGN, the ASIC design flow started.

4.2.3.7 Real data generation and capturing

The receiver design was strongly supported by real data generation and capturing. In order to do so, an FPGA platform was created. This tool was widely used during the project for several tasks, among which I can mention:

- Capture data from ISDB-T generator and Tuner output, in IF or Zero-IF to:
  - Test and validate Matlab receiver model;
  - Feed VHDL simulator for validation or debugging;
  - Feed FPGA physical test platform with real data;
  - Validate transmitter models;
- Capture data from the receiver prototype blocks, within the FPGA, to refine or debug algorithms in Matlab and/or VHDL;
- To support validation of blocks in HW.
- Injection of real and simulated (from transmitter and/or receiver models) data into FPGA receiver during the HW integration.

4.2.4 ASIC Design Flow

In order to accomplish the SoC implementation on silicon, the state-of-the-art ASIC design strategies were applied, following a top-down design flow. As can be seen in the previous steps, a system level architecture was described by using mathematical modeling and simulations on Matlab and/or Octave. Next, the models were mapped to a HDL. In order to validate the code, logic synthesis and rapid prototyping in FPGA devices were initially targeted, and an ISDB-T receiver prototype was built to validate complete HDL implementation of the receiver. A SoC/ASIC-focused logic and physical synthesis flow was executed afterward, in order to drive the design to the final SoC integration and physical implementation in silicon. All IP macros (on-chip memories, ADC, USB modules, IO interfaces) and the synthesized logic were integrated on the chip die. Logic and Physical Synthesis, timing closure, pre/post layout Functional Verification processes were executed, including automatic low-power insertion and DFT (Design-For-Testing) techniques, in order to guarantee the final quality-of-silicon. In the following section the main steps of the ASIC design are described.

4.2.4.1 Synthesis

In this stage the same HDL code implemented and tested using FPGA rapid prototyping is re-used for elaboration and circuit synthesis now focusing at the final silicon

technology (65 nm Global Foundries). In this step called front-end phase, the HDL code files are compiled (syntax analysis), elaborated, consistency and dependency checked, pre-optimized and mapped to the target technology, further optimization steps are executed in sequence, aiming at minimizing timing critical paths, power[1] and die area. DFT scan chain is also inserted in this step.

4.2.4.2 Gate level post-synthesis verification

This step consists of verification testbench elaboration, HDL code simulation, gate-level simulations post-synthesis (without and with timing). This verification is done in order to guarantee that the mapping to technology did not affect the SoC behavior and performance, compared to HDL simulations and FPGA prototyping. One of the main issues in this kind of verification is the long time needed to run the tests, especially for long tests as those that include the receiver synchronization (time, frequency, frame and MPEG) and equalization.

4.2.4.3 Physical design

This activity is focused at the SoC physical implementation on silicon. The main tasks involved are die size definition and Floorplanning, Placement, Clock Tree Synthesis, Pads insertion, Spare Cells[2], Routing and Area/Power/Timing Optimization. This phase is generally referred to as P&R (Place and Route) and includes a high number of iteration/optimization steps, targeting timing closure, in which the final work frequency should be accomplished. The file produced at the output of the physical design is the GDSII (GDS stands for Graphic Data System) file. Before the fabrication of the IC, the GDSII file is checked to verify LVS/DRC, antenna effects and metal density.

4.2.4.4 Gate level post-physical design verification

This step is similar to the Gate Level Verification Post-Synthesis. Nevertheless, it takes into account timing constraints, net delays and padding delays, in the simulations. In addition, the timing is checked if requirements for hard IPs are met. If the simulation succeeds the GDSII file generated can be sent to the foundry. Otherwise the Synthesis and/or Physical Design shall be done once more.

[1] Clock and Power Gating.

[2] Spare cells are added to achieve adequate cell density. These spare cells can be used to reduce the number of metal layers needed to be modified in case of ECO. It allows corrections with minimum mask modifications.

4.2.4.5 Physical IC in 65 nm

Before the fabrication of the IC, the GDSII file is checked to verify LVDS/DRC and other effects again. This is done by the IC designers with the support of the foundry designers. Only after this checking is that can the IC be fabricated, reducing the risks of wrong manufacturing. Finally, the IC is packaged; in this project, by a third part specialized company.

4.2.5 System Test and IC Characterization

In this step the fabricated IC is characterized electrically and several system tests are performed in order to confirm that the IC is working properly. Tests under AWGN and using wireless channels emulator were performed. In addition, several simple configuration tests were also carried out. In order to do so a test board was designed and the same SW used to test and configure the FPGA prototype was adapted to test the IC.

Chapter 5

The Proposed ISDB-T Receiver Architecture and Algorithms

5.1 Introduction

One of the aims of this work was to design a low complexity receiver to be used for High Definition Television fixed-reception as well as in low mobility devices such as laptops. Nevertheless, it is worth recalling that due to the intrinsic characteristics of the ISDB-T standard, the receiver can also work on speeds of about 200 km/h and higher, when the transmission system adopts Mode 1, QPSK, Code Rate 1/2 and Time Interleaver 2, for instance, without any advanced signal processing.

In the two previous chapters, ISDB-T standard and the main impairments a wireless receiver shall overcome, when working towards a practical implementation, were presented. Without the deep understanding of the standard and those impairments, any practical implementation compliant to ISDB-T standard would not be possible.

Along this Chapter, the algorithms selected to tackle those impairments and their architectures are going to be described in details. In addition some impairments are going to be revisited in the context of an OFDM-based receiver.

The chapter starts with the presentation of the main architecture of the receiver as well as the architecture of the implemented IC. In the IC architecture, the ISDB-T receiver is split in two blocks named Signal Processing - which encompasses synchronization blocks, FFT and equalizer - and and Data Processing, which contains the

blocks of the standard as well as the RS and Viterbi decoders. As a matter of implementation strategy, the Frame Synchronizer and Soft Demapper are within the Data Processing block.

One of the approaches used to reduce the receiver complexity (specially with regards to the Signal Processing blocks) was the adoption of the CORDIC algorithm, which is an algorithm for evaluating various elementary functions using a shift-and-add approach. So, before the presentation of the receiver blocks in detail, a review on CORDIC is provided. Following CORDIC overview, the impairments the receiver is subject to and the respective strategy to tackle them are presented in details.

In general, the first impairments to be handled are those related to timing. So, the selected algorithms for time synchronization are presented first. Nevertheless, in OFDM-based systems the timing error estimation is divided in coarse and fine OFDM symbol boundary error estimations, and residual sampling clock offset estimation. Initially, I present the selected algorithms for coarse boundary detection (which jointly detect the coarse frequency error) and fine boundary detection. The same algorithm used to estimate the coarse error is used to blindly detect the Mode and GI length. The remaining timing and frequency errors are handled by specific blocks that will be introduced later.

In OFDM systems, after coarse frequency error estimation and correction, the remaining frequency error is split in two components: an integer part (normalized to the subcarrier distance in Hertz) and a fractional part (i.e. any amount of error smaller than the subcarrier distance). So, the next blocks presented in this chapter are those which estimate those errors. It is worth to mention that the algorithm which estimates the integer frequency error also detects the pilot type (i.e. which of the four types of pilots that ISDB-T standard makes use of).

The next errors to be estimated are the fractional frequency error and remaining sampling clock error. The selected algorithm, that jointly detected those errors, makes use of the ISBD-T pilot structure. Following the strategy used to correct the remaining sampling clock error is presented.

The remaining of the chapter presents the equalization algorithm, the Soft Demapper, the Frame Synchronizer and some examples of simulations results. The chapter ends with a conclusion section.

5.2 An Architecture for an IF Digital Receiver for ISDB-T Signal Reception

The most demanding task in the ISDB-T receiver project was to design and validate a reference model of the receiver using a high level abstraction language. It was from that model that the receiver blocks were specified to be mapped into an independent platform HDL. In this work we have used Matlab and Octave to generate a full Matlab compliant code. After building the receiver blocks, simulations were carried out in order to simulate the behavior of the complete receiver under real environment conditions such as timing and frequency impairments, wireless channel, degradation due to C/N variations. The proposed receiver architecture (ISDB-T core) is presented in Figure 5.1.

The proposed receiver architecture supports low-IF or Zero-IF input signal. Following a 10 bits analog-to-digital conversion, power estimation is provided for AGC purposes. DC (Direct Current) level removal can be applied or bypassed depending on the chosen tuner architecture and performance regarding this impairment. Nevertheless, it is worth to mention that high DC level at tuner output causes a reduction in the useful dynamic range of the ADC. In the case of IF reception, down conversion is performed by using a CORDIC-based DDS, followed by low pass filtering. On the other hand if Zero-IF is the case, only low pass filtering is performed in order to reduce adjacent channel interference. Initially, after receiver power-up and reset, the CORDIC-based de-rotator is bypassed. Later, after coarse boundary detection and coarse frequency estimation in time domain, the de-rotator is fed with the coarse frequency estimation and starts working. Next, after frequency domain timing and frequency estimation (based on CORDIC algorithm), fine frequency correction is performed using the same de-rotator, followed by a Farrow-based timing resampling that adjusts the fine timing error and apply a constant decimation factor, due to oversampling at the initial stages. Initially, the timing resampling unit is bypassed and only coarse OFDM symbol boundary and frequency error are estimated and corrected. At this point it is possible to apply blind time-domain estimation for the GI and operation mode (i.e mode 1, 2 or 3), as well as fine boundary estimation and correction.

Once the signal is synchronized in time and frequency domain, GI removal is performed , followed by FFT (2048, 4096 or 8192 points). Fine boundary is continuously performed, in order to keep the FFT symbol border correct and avoid inter-symbol interference. This task, as well as the coarse boundary estimation/removal are specially critical for SFN or SFN-like (caused by repeaters and gap fillers) wireless channels. After FFT is carried out, pilot extraction is done. These pilots are used for estimation of Integer Frequency Error, channel frequency response, SNR(C/N), fine frequency error and fine time error as well. The channel estimation is based in a division performed by the CORDIC working on Linear Vectoring Mode. The CORDIC-based channel estimator also provides CSI (Channel State Information) for Soft Viterbi Decoder. Subsequent to equalization, frame synchronization based on TMCC infor-

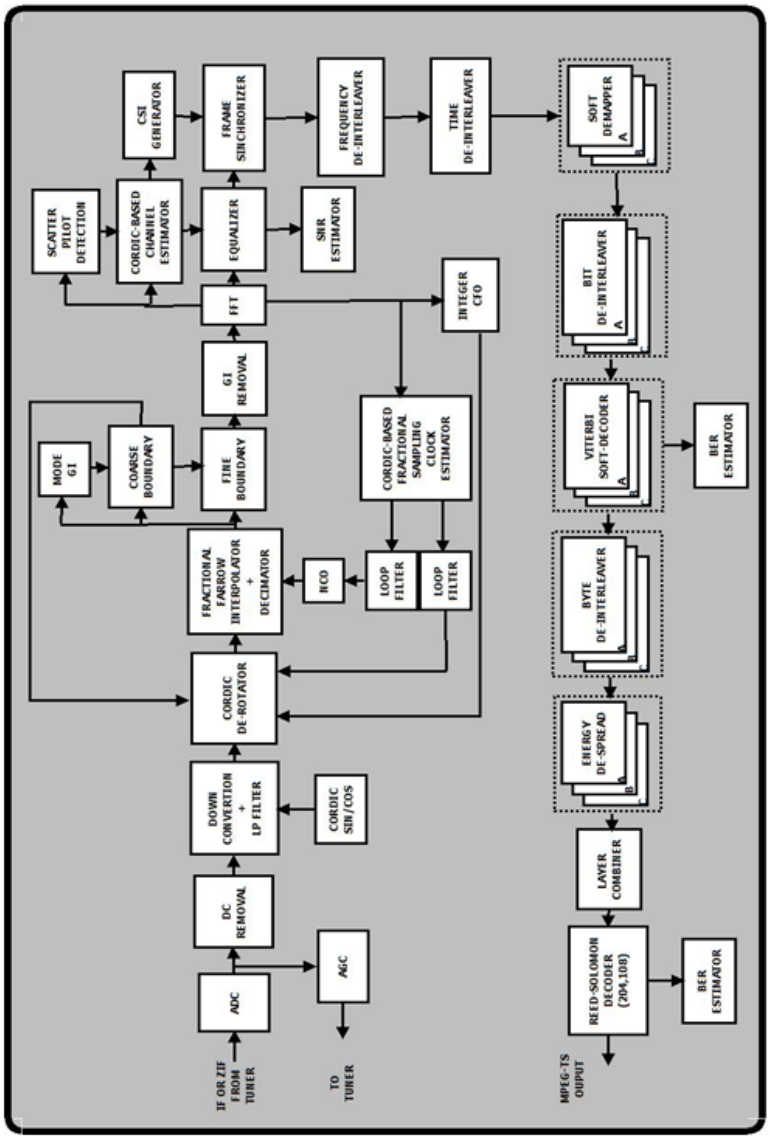


Figure 5.1: Proposed receiver architecture.

mation is performed, so the following control and configuration parameters can be retrieved (from TMCC) in order to configure the remaining blocks of the receiver pipeline: baseband modulation, time interleaving length, number of layers, number of segments per layer and code rate. From this point on, the remaining blocks are functionally the inverse of the ISDB- T transmitter blocks, presented in Figure 5.1. Nevertheless, Soft Demapper, Soft Viterbi Decoder and Reed Solomon Decoder are much more complex than baseband mapping and encoders respectively. Two other blocks are implemented in order to support the evaluation of the receiver performance: Viterbi and Reed Solomon BER estimator.

At Reed Solomon decoder output the MPEG-TS packets are available for audio and video decoding. In this work we proposed two solutions to deliver the MPEG packets to MPEG-TS decoder: 1) MPEG-TS output; and 2) delivery of MPEG-TS over USB Interface. It is worth to mention that ADC, and USB analog PHY were supplied by IP providers. Figure 5.2 shows the proposed architecture for the ISDB-T receiver to be implemented in ASIC and prototyped in FPGA.

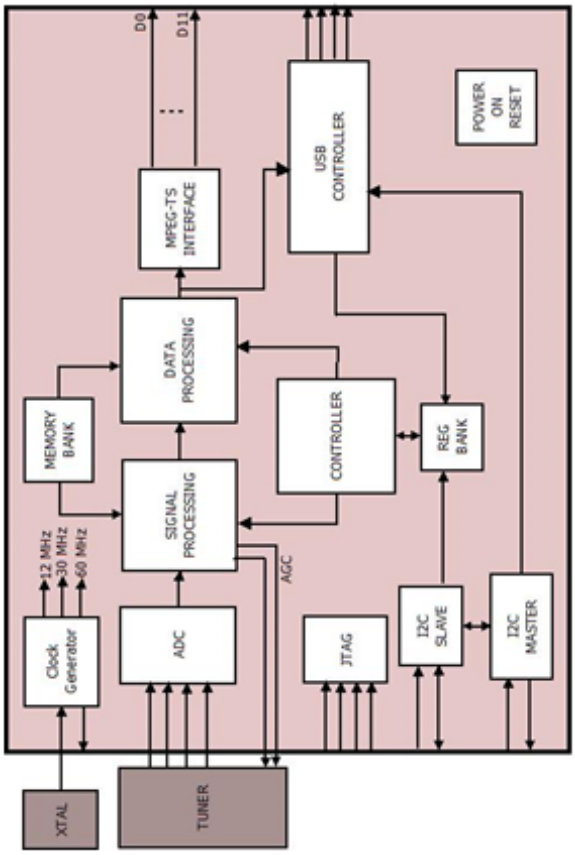


Figure 5.2: IC architecture.

For sake of clearness and to make the design management easier, the ISDB-T core receiver was split in two functional partitions, namely Signal Processing and Data Processing. The latter encompasses from Frame Synchronization to Reed Solomon Decoding. The blocks, presented in Figure 5.1, preceding Frame Synchronization belongs to Signal Processing partition.

As previously mentioned, the receiver model was used as reference for the system level specification of the receiver blocks to be implemented in HDL. The simulation results of the receiver performance under impairments conditions, quantization, as well as for different wireless channels and AWGN, were used to guarantee that the chosen algorithms and architecture would achieve the AWGN performance determined by the standard [1], as well as performance over wireless channels Brazil-A, Brazil-B and ETSI channel [2].

5.3 The Digital Receiver Algorithms in Details

5.3.1 CORDIC Algorithm

The architecture proposed in this thesis is strongly based on the CORDIC algorithm, introduced by Volder in [86]. Several applications of CORDIC can be found in [87] and [88]. It is an iterative arithmetic computing algorithm capable of evaluating various elementary functions using a unified shift-and-add approach [87]. Due to its flexibility and low resource usage, CORDIC is an alternative architecture for the implementation of several algorithms in DSP and VLSI. Due to its unified approach CORDIC is especially suitable for VLSI array processors that make use of elementary trigonometric functions and multiplication-accumulation (MAC). In this thesis, CORDIC was used to implement DDS (Digital Discrete Synthesis), angle rotation, angle computation, module and division.

5.3.1.1 The CORDIC Algorithm

Consider the rectangular vector (X_{in}, Y_{in}) and a given rotation angle θ. The equation that represents the rotation of that input vector by the angle θ to obtain a new vector (X_{out}, Y_{out}) is:

$$\begin{bmatrix} X_{out} \\ Y_{out} \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \cdot \begin{bmatrix} X_{in} \\ Y_{in} \end{bmatrix} \tag{5.1}$$

The operation shown in Equation 5.1 is observed in Figure 5.3 and can be rewritten as:

$$\begin{bmatrix} X_{out} \\ Y_{out} \end{bmatrix} = \cos \begin{bmatrix} 1 & -\tan\theta \\ \tan\theta & 1 \end{bmatrix} \cdot \begin{bmatrix} X_{in} \\ Y_{in} \end{bmatrix}. \tag{5.2}$$

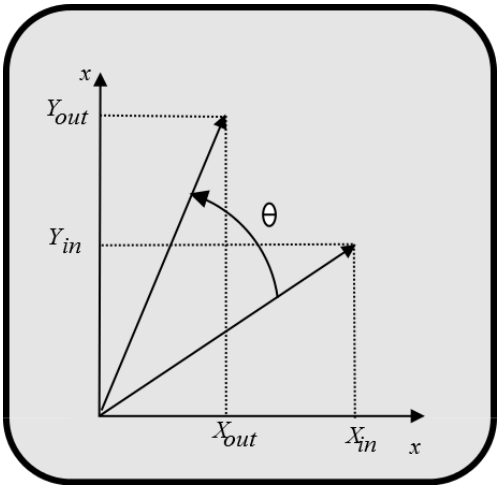


Figure 5.3: Rotation of a vector.

The basic concept of CORDIC is vector rotation by means of decomposing the desired rotation angle into the weighted sum of a set of n predefined elementary rotation angles [87]. The rotation for each of the n angles can be accomplished by means of simple shift-and-add operations. The desired rotation angle is represented as:

$$\theta = \sum_{i=0}^{n-1} \mu_i \alpha_m(i). \tag{5.3}$$

The i_{th} elementary rotation angle, $\alpha_m(i)$is defined as:

$$\alpha_m(i) = \frac{1}{\sqrt{m}} \tan^{-1}\left(\sqrt{m} 2^{-s(m,i)}\right). \tag{5.4}$$

Where,

$$\begin{aligned} \alpha_m(i) &= -2^{s(m,1)} && m = 0 \\ \alpha_m(i) &= \tan^{-1} 2^{-s(m,i)} && m = 1 \\ \alpha_m(i) &= \tan^{-1} 2^{-s(-m,i)} && m = -1 \end{aligned} \quad . \tag{5.5}$$

$m = 1,\ -1$ and 0, corresponds to the rotation operation in circular coordinates, hyperbolic coordinate system, and linear coordinate system, respectively. μ_i is a

sequence of $\pm 1s$. The term $s(m,i)$ corresponds to the integer shift sequence. Given x_0, y_0 and z_0 and using Equation 5.1, the CORDIC algorithm is described as:

For $i = 0$ *to* $n-1$

$$\begin{bmatrix} x_{i+i} \\ y_{i+1} \end{bmatrix} = \begin{bmatrix} 1 & -m\mu_i 2^{-s(m,i)} \\ \mu_i 2^{-s(m,i)} & 1 \end{bmatrix} . \begin{bmatrix} x_i \\ y_i \end{bmatrix} \tag{5.6}$$

And

$$z_{i+1} = z_i - \mu_i \alpha_i \tag{5.7}$$

End i Loop.

The pseudo code shown in Equation 5.7, represents the pseudo rotations of CORDIC algorithm. In order to recover the amplitude information, a scaling is need after the nth iteration. The scaling factor is given by[1]:

$$\begin{bmatrix} x_{out} \\ y_{out} \end{bmatrix} = \frac{1}{K_m(n)} . \begin{bmatrix} x_n \\ y_n \end{bmatrix} . \tag{5.8}$$

Where,

$$K_m = \prod_{i=0}^{n-1} \sqrt{1 + m\mu_i^2 2^{-2s(m,i)}} \tag{5.9}$$

After some iterations the factor $\frac{1}{K_m}$ becomes constant and equals to 0.6073. A single iteration of the generalized CORDIC algorithm, can be written as:

$$\begin{array}{lcl} x_{i+1} & & x_i - md_i 2^{-i} y_i \\ y_{i+1} & = & y_i + d_i 2^{-i} x_i \\ z_{i+1} & & z_i - d_i \alpha_i \end{array} \tag{5.10}$$

Table 5.1 shows a summary with the final result (i.e. for $i = n$) of the generalized CORDIC algorithm.

[1] Scaling operation is only needed for $m = \pm 1$ cases

Table 5.1: Unified CORDIC Functions.

m	COORDINATES	ROTATION MODE	VECTORING MODE
1	$arctan(2^{-i})$ (Circular)	$x_n = K(x_0 \cos z_0 - y_0 \sin z_0)$ $y_n = K(y_0 \cos z_0 + x_0 \sin z_0)$ $z_n = 0$	$K\sqrt{(x_0^2 + y_0^2)}$ 0 $z_0 - \arctan\left(\frac{y_0}{z_0}\right)$
0	2^{-i} (Linear)	$x_n = x_0$ $y_n = y_0 + x_0 z_0$ $z_n = 0$	x_0 0 $z_0 - \frac{y_0}{z_0}$
-1	$arctanh(2^{-i})$ (Hyperbolic)	$x_n = K_h(x_0 \cosh z_0 + y_0 \sinh z_0)$ $y_n = K_h(y_0 \cosh z_0 + x_0 \sinh z_0)$ $z_n = 0$	$K_h\sqrt{(x_0^2 - y_0^2)}$ 0 $z_0 + arctanh\left(\frac{y_0}{z_0}\right)$

5.3.2 Time Synchronization

There are three causes for the wrong OFDM symbol boundary positioning at the receiver. The first is the random initialization of the OFDM receiver that causes the OFDM-symbol to be sampled at a position with a positive or negative offset regarding the ideal OFDM symbol boundary. The second is the wrong estimation of the boundary. The third is the drift from ideal sampling point due to clock errors, caused by the crystal frequency deviation (i.e. crystal *ppm*). The greater the *ppm* the faster will be the drift, and depending on it value, it is possible to cause a integer number of sample shifts between two or more OFDM symbols, requiring the computation of a new boundary position. Usually, the *ppm* of the crystals used in receivers devoted to broadcast signal reception is low and causes only fractional sampling errors (*i.e.* does not cause integer sample drifts between two consecutive OFDM symbols), which will be covered in Section 5.3.4. It is worth noticing that there are OFDM-based systems that use packet-oriented burst transmission and can work with high *ppm* crystals. These systems have training sequences at the beginning of the burst transmission and are synchronized in a burst basis. The time difference between two bursts is such that it does not cause border deviation in those OFDM symbols that transport the data burst.

The symbol boundary error is given by integer values, and a straight consequence of this error is the adoption of a wrong FFT window. The effects of that wrong choice are described below.

5.3.2.1 Effects of Timing Offset

The symbol boundary error can be either towards GI (early border) or OFDM symbol (late border), as can be seen in Figure 5.4. For the same error absolute value, the first case causes less degradation in the received signal than the second, whose effects are more severe. Figures 5.5 to 5.8 show the qualitative effects of early and late OFDM symbol border detection, for Mode 3 and GI = 1/4, without noise. The inspection of those figures show that indeed the latter case causes more degradation in the signal than the former. In addition, Figure 5.9 shows the effects of boundary error in the Bit Error Rate.

As a matter of simplicity, lets first consider the case in which the received signal is only corrupted by AWGN. When the OFDM symbol boundary is early, a phase rotation occurs, which can be handled by the equalizer. Nevertheless, in the case of a wireless channel with a given delay spread, after a number of early samples the current symbol will start suffering from Inter Symbol Interference (ISI)[1] in addition to the

[1]It is worth mentioning that the FIR filters used in the receiver, e.g. for low pass filtering, can be seen as a channel with a given delay and can increase the delay spread of the wireless channel and possibly cause ISI in the early-case of symbol boundary error.

phase rotation [44]. As mentioned previously, phase rotation can be handled by the equalizer but not the ISI. Figures 5.5 to 5.7 show those effects. In the case of a late border, Inter Carrier Interference (ICI) will occur due to the loss of the orthogonality between the OFDM subcarriers, as well ISI, because part of the following symbol will be demodulated as belonging to the current symbol [44]. Figure 5.8 shows the effect for 20 samples late and Figure 5.9 shows the effect of boundary error in the BER for Mode 1 with GI=1/4.

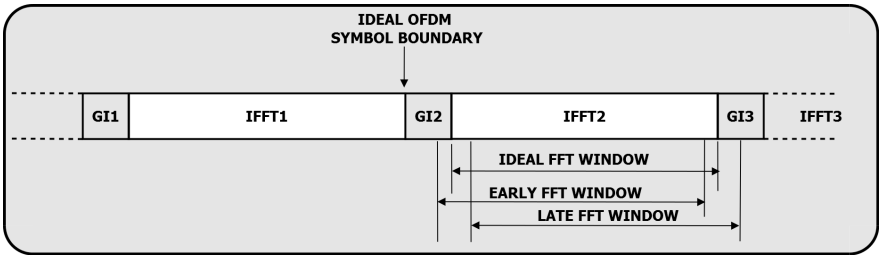


Figure 5.4: OFDM Symbol boundary error.

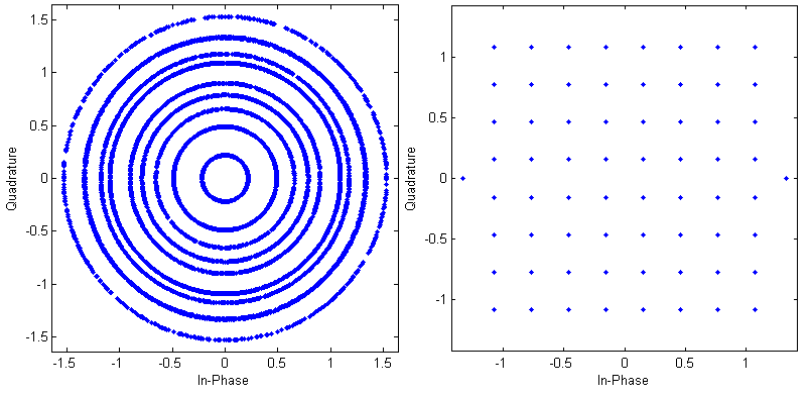


Figure 5.5: Timing error effects due to OFDM Symbol boundary error of 20 early samples, with (right) and without (left) frequency-domain equalization - Mode 3, GI 1/4.

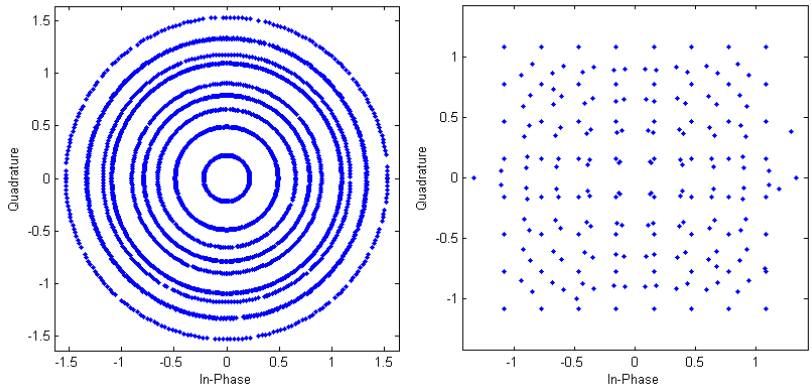


Figure 5.6: Timing error effects due to OFDM Symbol boundary error of 800 early samples, with (right) and without (left) frequency-domain equalization - Mode 3, GI 1/4.

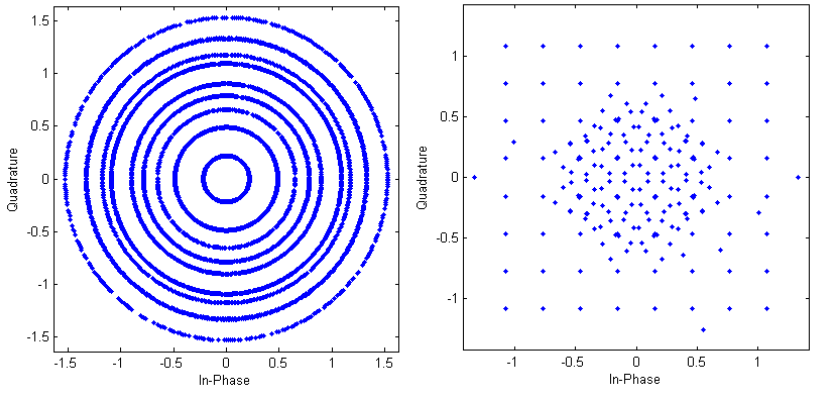


Figure 5.7: Timing error effects due to OFDM Symbol boundary error of 1200 early samples, with (right) and without (left) frequency-domain equalization - Mode 3, GI 1/4.

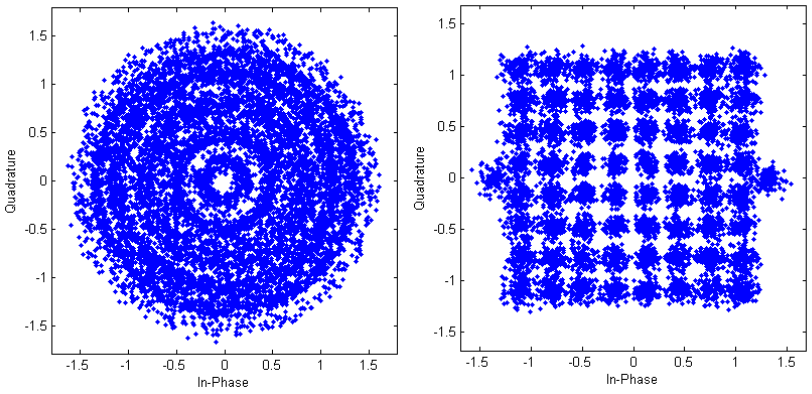


Figure 5.8: Timing error effects due to OFDM Symbol boundary error of 20 late samples, with (right) and without (left) frequency-domain equalization - Mode 3, GI 1/4.

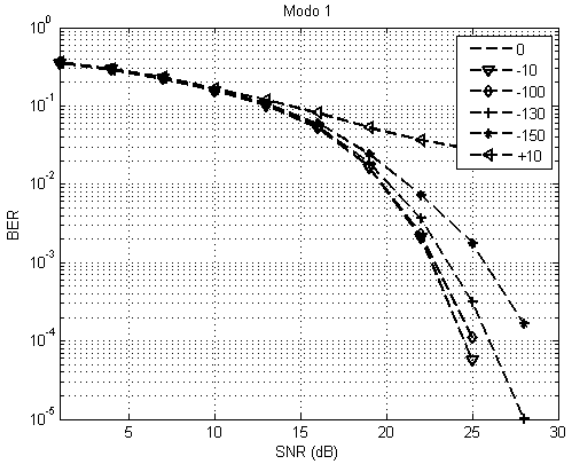


Figure 5.9: Effects of boundary error in the BER for Mode 1.

5.3.2.2 Symbol Boundary Detection, Coarse Fractional Carrier Frequency Offset Estimation and, Mode and GI Detection Algorithm

In this work, the algorithm used as basis to estimate the symbol boundary error, coarse fractional Carrier Frequency Offset (CFO), mode and GI length, is the Delay and Correlate (DC) algorithm described in [69], [89] and [90]. Nevertheless, I used the normalized version of the DC, named NDC. The advantage of using the normalization is that it is possible to define a clear threshold above which the parameters are considered be valid, *i.e.* the correct boundary is found and the received OFDM symbols samples can be synchronized. The price paid for the normalization is the use of a division operation, that adds complexity in the implementation. Nevertheless, in this work, that operation is performed using the CORDIC algorithm working in the Vectoring Mode with Linear Coordinates (see Table 5.1).

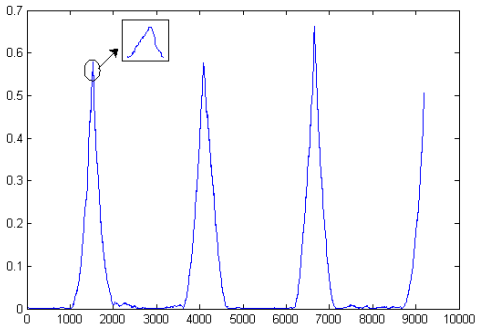


Figure 5.10: Example of estimation of coarse boundary - Mode 1 GI 14 ETSI channel 5dB.

5.3.2.2.1 Detection of Coarse and Fine Symbol Boundaries, Mode and GI

Figure 5.11 shows a simplified block diagram of the main algorithm, that works as follows. First, the input buffer is filled in with $2 \times (N + Ng)$ baseband samples corrupted by frequency, timing and boundary errors, where N and N_g are the FFT and GI length. Mode and GI are chosen and the two secondary buffers are filled in with N_g samples, from the input buffer, starting from sample $n = 0$. Mode and GI are chosen starting from Mode 3 and GI=1/4, correlation and peak search[1] is performed.

[1]In order to smooth the correlation curve, a configurable length moving average is applied and the peak position is shifted left (towards the previous OFDM symbol) in order to guarantee the border falls in the GI, instead falling into the following OFDM symbol. This adjustment is configurable.

If the boundary is not found, the secondary buffers are filled with a new set of N_g samples from the input buffer, but for $n = 1$ and so on. This is repeated until the entire set of samples of input buffer is used. Nevertheless, if the border is not found yet, the Mode is kept and a new valid GI[1] value is chosen. When all valid GI values are used and the border is still not found, the Mode is changed to 2 and the process is repeated, starting from input buffer sample $n = 0$. If the boundary is still not found, Mode 1 is selected, followed by the selection of different GI values, until the border is found. Nevertheless, if after all this search over the valid set of Mode and GI the boundary is still not found, the receiver will reset, fill the input buffer with new set of data and repeat the search process as described previously until the symbol boundary is found. After a certain number of attempts, the border failure flag is raised, meaning that the receiver is under severe channel conditions or the SNR is below the threshold the NDC is capable of working properly. If at any time the boundary is detected, *i.e.* the normalized peak is above the threshold, consequently GI and Mode are also found. Therefore, the boundary symbol can be corrected and coarse fractional CFO can be computed. At each reset a counter is initialized, and for each rising edge of the sampling clock this counter is incremented. After the boundary is found, the counter can be used to compute the offset between the computed boundary and that relative to the free running received samples. A detailed view of the NDC algorithm is found in Figure 5.12. NDC is a very robust algorithm, especially for AWGN. Figure 5.10 shows an example of symbol boundary computation using the NDC algorithm for ISDB-T working in Mode 1 GI =1/4, with 5 dB of SNR in AWGN channel.

[1]The ISDB-T parameters can be found in Sections 3.1.2 and 3.1.1.

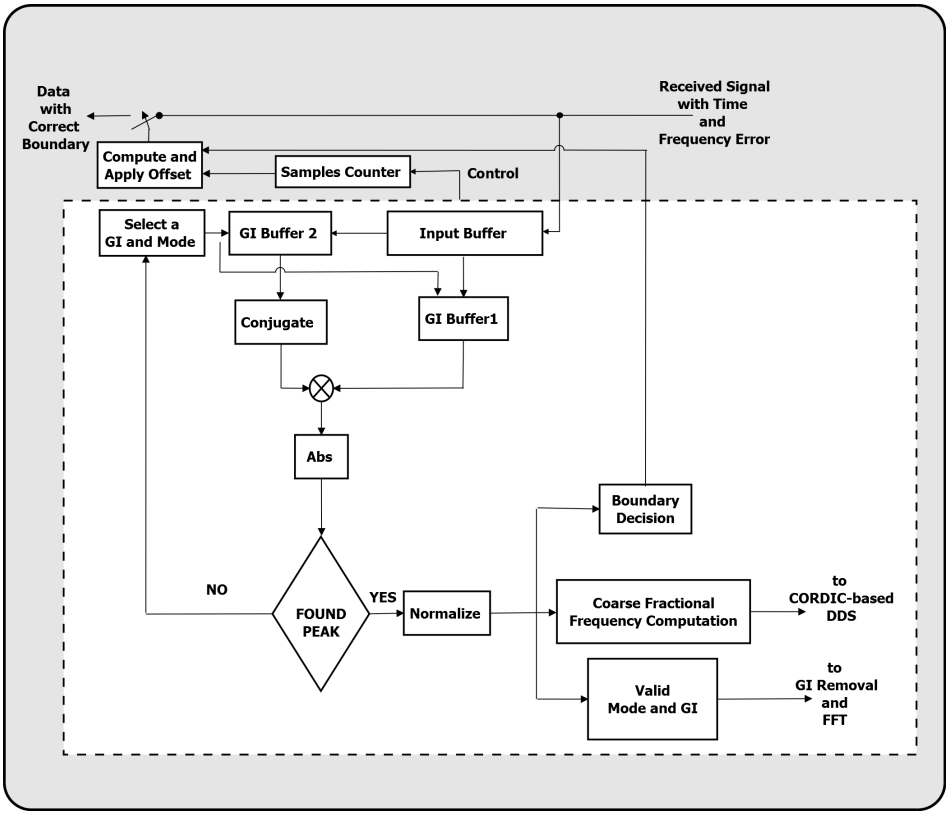


Figure 5.11: Simplified block diagram of the algorithm for boundary detection, coarse fractional CFO estimation and, Mode and GI detection .

The Fine Boundary Detection Algorithm works in a similar manner as the Coarse Boundary Detection Algorithm. Nevertheless, the search for the peak is done over a smaller search buffer, around the estimated coarse boundary. However, in this case the secondary buffer is filled with real-time captured samples, for each new incoming OFDM symbol. The length of the search windows is configurable and depends on the detected Mode and GI. The search for the fine border is done for each OFDM symbol. Fine Boundary Detection Algorithm, which is a simplified version of the Coarse Boundary Detection Algorithm, also makes use of the NDC algorithm. Recall that the algorithm used for coarse estimation, jointly compute the Coarse Fractional CFO and blindly detects Mode and GI.

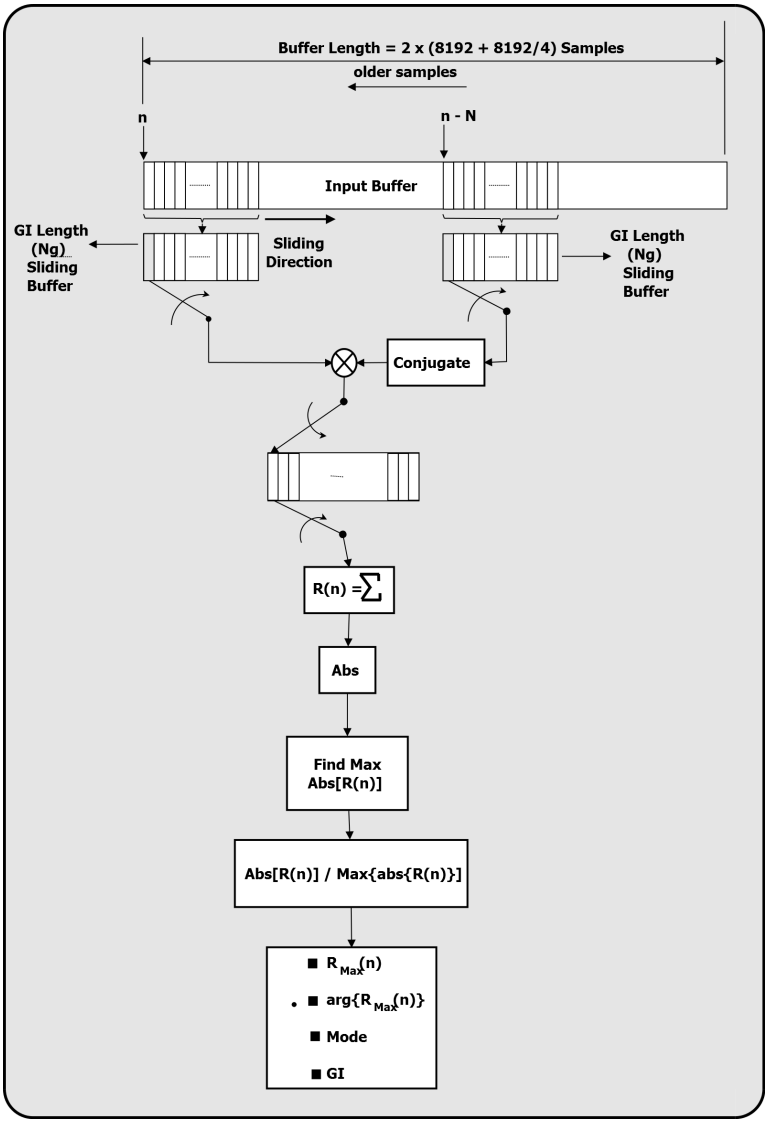


Figure 5.12: Schematic of the computation of $R(n)$ and normalized maximum of $R(n)$, showing the sliding GI buffers.

5.3.2.2.2 Coarse Fractional CFO Estimation

Given the baseband complex time-domain received signal $r(k)$, the DC algorithm

is obtained by evaluating the following autocorrelation equation [90] [91]:

$$R(n) = \sum_{k=0}^{k+N_g-1} r(n+k)r(n+k+N)^* \tag{5.11}$$

Where, $r()^*$ is the conjugate of $r()$.

Given the baseband received signal $r(n)$, with a normalized CFO regarding the subcarrier space, namely Δf_{CFO}, the $r(n+N)$ sample can be computed using equation 5.12, in which N is the FFT length. Δf_{CFO} is divided in 2 components, the first is the integer part of CFO, Δf_I, that is multiple of the subcarrier spacing $F_{sc} = 1/Tu$, and the second,Δf_F, is the fractional part. So, it can be computed as $\Delta f_{CFO} = \Delta f_I + \Delta f_F$.

$$r(n+N) = r(n)\exp^{j\frac{2\pi\Delta f_{CFO}}{N}xN} = r(n)\exp^{j2\pi\Delta f_{CFO}}. \tag{5.12}$$

As $\exp^{j2\pi\Delta f_{CFO}}$ is a constant, when applying the DC Algorithm Equation, *i.e.* Equation 5.11, to the samples that result in the maximum correlation, *i.e.* for $n = n_{max}$, $R(n_{max})$ can be computed as:

$$R(n_{max}) = R(n_{max})\exp^{j2\pi\Delta f_{CFO}}. \tag{5.13}$$

Therefore, Δf_{CFO} can be computed evaluating the argument of $R(n)$ at the maximum correlation point, *i.e.* $R(n_{max})$. Nevertheless, as the phase of the error present in Equation 5.12 can only be evaluated in $[-\pi, \pi]$, the DC (and NDC) algorithm can only compute the fractional part of Δf_{CFO}. If the received signal has also an integer frequency offset, the frequency domain algorithm presented in Section 5.3.3.2 can compute it. In this work I applied the NDC algorithm to compute the Coarse Fractional CFO, as shown in Equation 5.14. The remaining fractional CFO (RCFO) is computed and corrected as will be shown in Section 5.3.4.

$$Coarse\ Fractional\ CFO = \frac{1}{2\pi}\arctan\left(\frac{\text{Imag}\{max[R(n)]\}}{\text{Real}\{max[R(n)]\}}\right) \tag{5.14}$$

The Coarse Fractional CFO value, in Hertz, is computed by multiplying the subcarrier distance, which is Mode-dependent[1], by the *Coarse Fractional CFO*. Then, the estimated frequency is applied to the CORDIC-based de-rotator[2] as shown in Figure 5.1.

[1]See ISDB-T parameters in Sections 2.1.4 and 2.1.3.
[2]With CORDIC algorithm working in Rotation Mode with Circular Coordinates

5.3.3 Fine and Integer Carrier Frequency Offset

The CFO is divided in two parts: an integer part, named Integer CFO (ICFO), and the fractional part as mentioned in Section 5.3.2.2.2. On the other hand, the fractional error is also divided in two parts, the Coarse Fractional CFO and the Remaining Fractional CFO, named RFCO. The Coarse Fractional CFO is jointly estimated with the coarse OFDM symbol boundary and is not able to handle all the fractional frequency error. So, the remaining part of the fractional error needs to be corrected, in order to preserve the subcarrier orthogonality. RCFO is also known as fine frequency offset. In this work, both ICFO and RCFO are computed in the frequency domain using the embedded Scattered Pilots (SP), whose structure is shown in Figure 5.13. SP's are generated using the rules presented in Section 3.1.2.16.1 and reference [1].

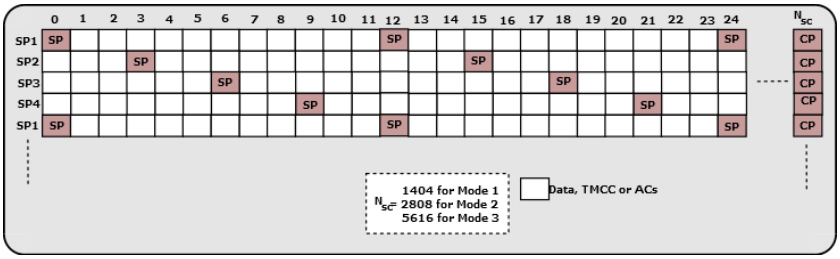


Figure 5.13: Pilot types and pilot structure for ISDB-T frame using synchronous modulation.

5.3.3.1 The Effects of ICFO and RCFO

ICFO causes a shift in the subcarrier index, what renders the signal recovery impossible after FFT. The sign of the ICFO can be either positive or negative with relation to the central OFDM subcarrier. The subcarrier index shift due to ICFO allows computing the error (offset) in the Frequency Domain.

RCFO has three effects: attenuation in the magnitude of the subcarrier, phase rotation (constant) and Inter-Carrier Interference (ICI) [92] [69]. Figures 5.14 to 5.17 presents simulations showing the effects of RCFO in the symbol constellation for equalized and non-equalized OFDM symbols, for Modes 1 and 3, without RCFO correction.

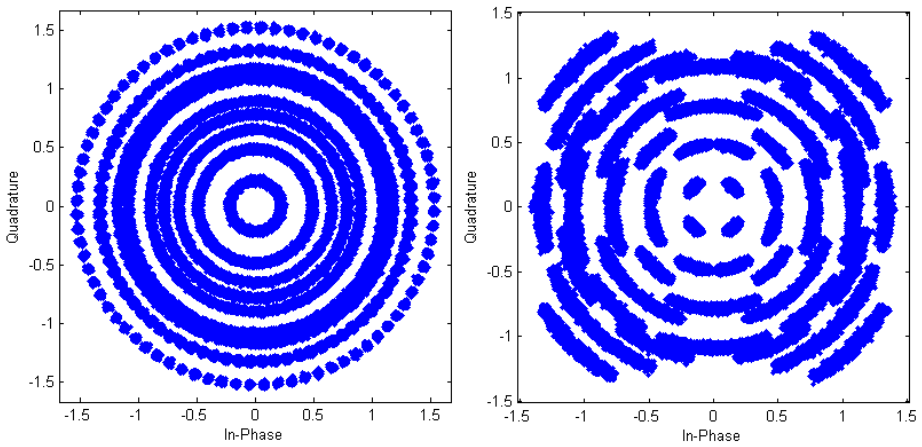


Figure 5.14: Effect of 10 Hz RCFO, for Mode 3, with 64-QAM - equalized (right) and non-equalized (left) OFDM symbols, without noise.

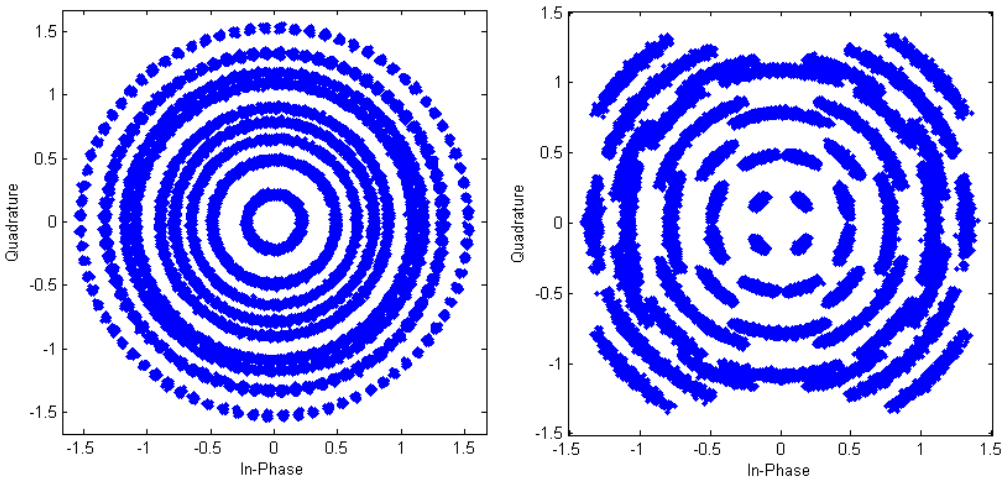


Figure 5.15: Effect of 10 Hz RCFO, for Mode 1, with 64-QAM - equalized (right) and non-equalized (left) OFDM symbols, without noise.

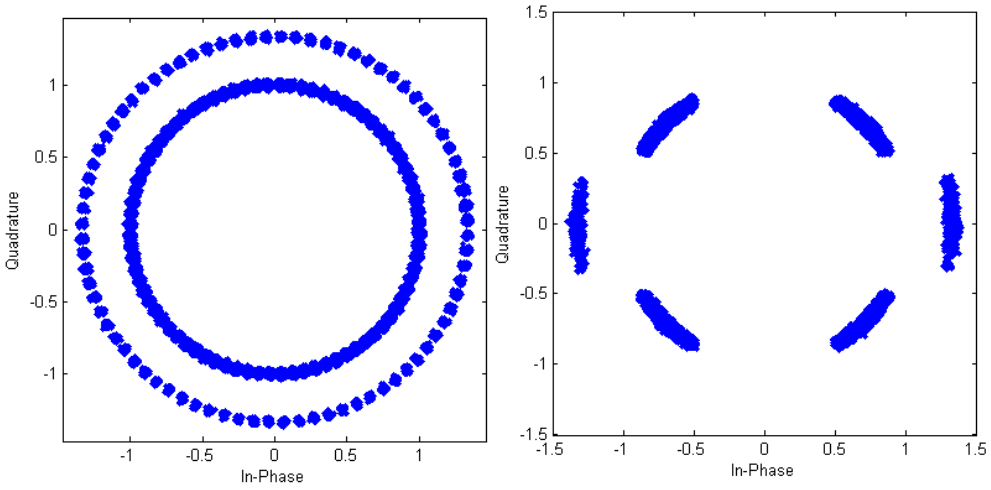


Figure 5.16: Effect of 10 Hz RCFO, for Mode 1, with QPSK - equalized (right) and non-equalized (left) OFDM symbols, without noise.

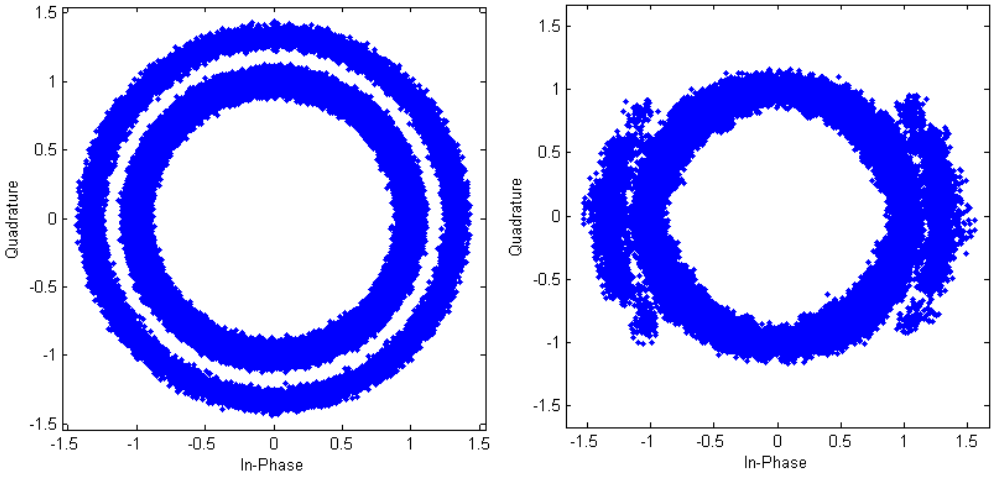


Figure 5.17: Effect of 30 Hz RCFO, for Mode 1, with QPSK - equalized (right) and non-equalized (left) OFDM symbols, without noise.

5.3.3.2 ICFO Estimation

Since ICFO causes shift in the subcarrier index, and the fact that there are SPs embedded among the OFDM Symbols subcarriers, it is a natural choice to select a Frequency Domain, *i.e.* post-FFT, algorithm to compute the frequency error. In order to reduce the complexity of ICFO estimation algorithm, the search for the integer error is performed over a frequency band, called *Search Window* (see Figure 5.18), which is configurable, and strongly depends on the receiver crystal *ppm* choice, *i.e* the larger the crystal *ppm*, the larger the ICFO. ICFO estimation/correction is preceded by the pre-FFT Coarse Fractional CFO estimation/correction.

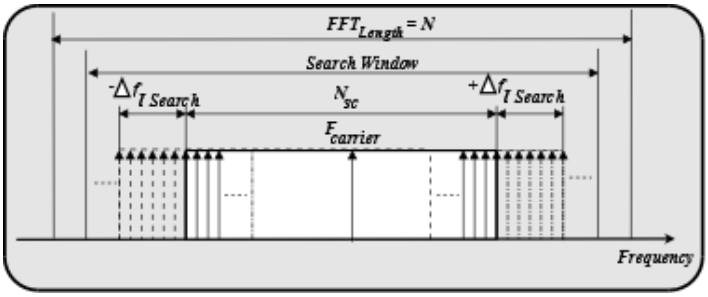


Figure 5.18: Integer frequency shift.

Figure 5.19 presents a schematic showing the principle of the post-FFT ICFO estimation algorithm, *i.e.* the correlation between a locally generated SP with the SPs extracted from the incoming OFDM Symbol. Nevertheless, an issue arises at this point: the unknown *Pilot Type* of the current OFDM Symbol that is used to feed the four correlation buffers shown in Figure 5.20. As shown in Figure 5.13, *Pilot Type* can be SP1, SP2, SP3 or SP4 depending on the starting point of the first SP within the OFDM Symbol. Therefore, it is necessary to figure out, first, what the *Pilot Type* is to then compute the ICFO. Figure 5.20 shows the schematic of the joint *Pilot Type* detection and ICFO estimation algorithm. The algorithm is based on cross-correlation between a locally generated SP and the Pilots captured from the incoming OFDM Symbols. The Algorithm works as follows:

1. N points FFT is performed in an OFDM Symbol, whose Mode and GI were detected and its boundary synchronized. N is 2048, 4096 or 8192;
2. A *Search Window* (see Figure 5.18) for the ICFO estimation is defined as *Search Window* $= N_{sc} + 2 \times \Delta f_{I\ Search}$ subcarriers, where N_{sc} is the number of subarriers containing data, pilots and TMCCs, and $\Delta f_{I\ Search}$ is the positive or negative interval to perform the ICFO search, *i.e.* the ICFO (δf_I) shall be smaller or equal to $\Delta f_{I\ Search}$;

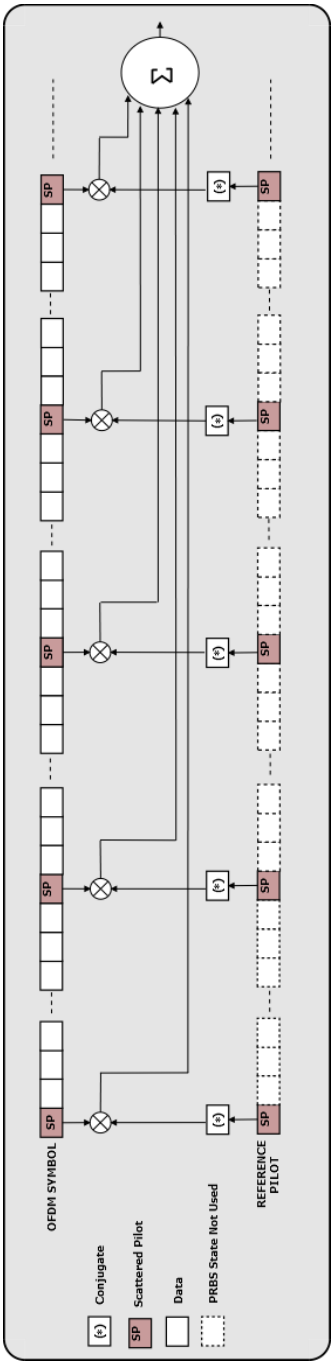


Figure 5.19: Principle of the estimation of integer frequency error based on cross-correlation.

3. Next, *Search Window* samples around the central subcarrier, *i.e.* DC, are selected and used to feed the four correlation buffers shown in Figure 5.20;

4. Starting from the first symbol in Frequency Domain, it picks out every 12th element until it obtains the total of N_{pilots} frequency domain symbols, as shown in Figure 5.20, where N_{pilots} is the number of Pilots per OFDM Symbol, which is Mode dependent;

5. Correlate the N_{pilots} symbols, that were picked out from the FFT output samples, with N_{pilots} of SP1, SP2, SP3 and SP4, as shown in Figures 5.19 and 5.20;

6. One of the four correlators which presents the highest correlation above a predefined threshold (that is configurable) is considered to be the right *Pilot Type*;

7. If the correlation peak is not found, the algorithm moves one carrier index ahead and repeats *items* 4 to 6, until it finds the *Pilot Type* and the ICFO;

8. In case the entire *Search Window* is used and the correlation peak is not found, a new OFDM Symbols is used to feed the correlation buffer and the algorithm goes to step 3 until it finds the correlation peak;

9. If after several (configurable) attempts the *Pilot Type* and ICFO are not found the receiver will reset and start the synchronization process from the beginning, *i.e.* Mode, GI and Boundary detection;

10. If even after all these attempts, the *Pilot Type* and ICFO are not detected, the receiver is supposed to be in a low level coverage area or suffering strong interference for instance. Therefore, it is necessary to either move the receiver or redirect the antenna.

Once the ICFO is found, the value is added to the CORDIC-based IF generator, shown in Figure 5.1, and the estimator is turned off. In parallel the *Pilot Type* is sent to the Frequency Domain Equalizer and to RCFO/SCO estimators. It is worth mentioning that while the joint ICFO and *Pilot Type* estimator is working the number of OFDM Symbols that are passing through the receiver pipeline is being counted. By doing so, it is possible to define the *Pilot Type* of the next OFDM Symbol that is going to the Equalizer and RCFO/SCO estimators.

5.3.4 Joint RCFO and SCO Estimator

The blind coarse CFO estimator is not able to handle all fractional CFO. Therefore, a second stage of estimations is necessary, in order to lead the steady state error to acceptable levels. As described in Section 5.3.3.1, fractional CFO (coarse CFO plus RCFO) causes a constant phase shift over all OFDM Symbol subcarriers. On the other

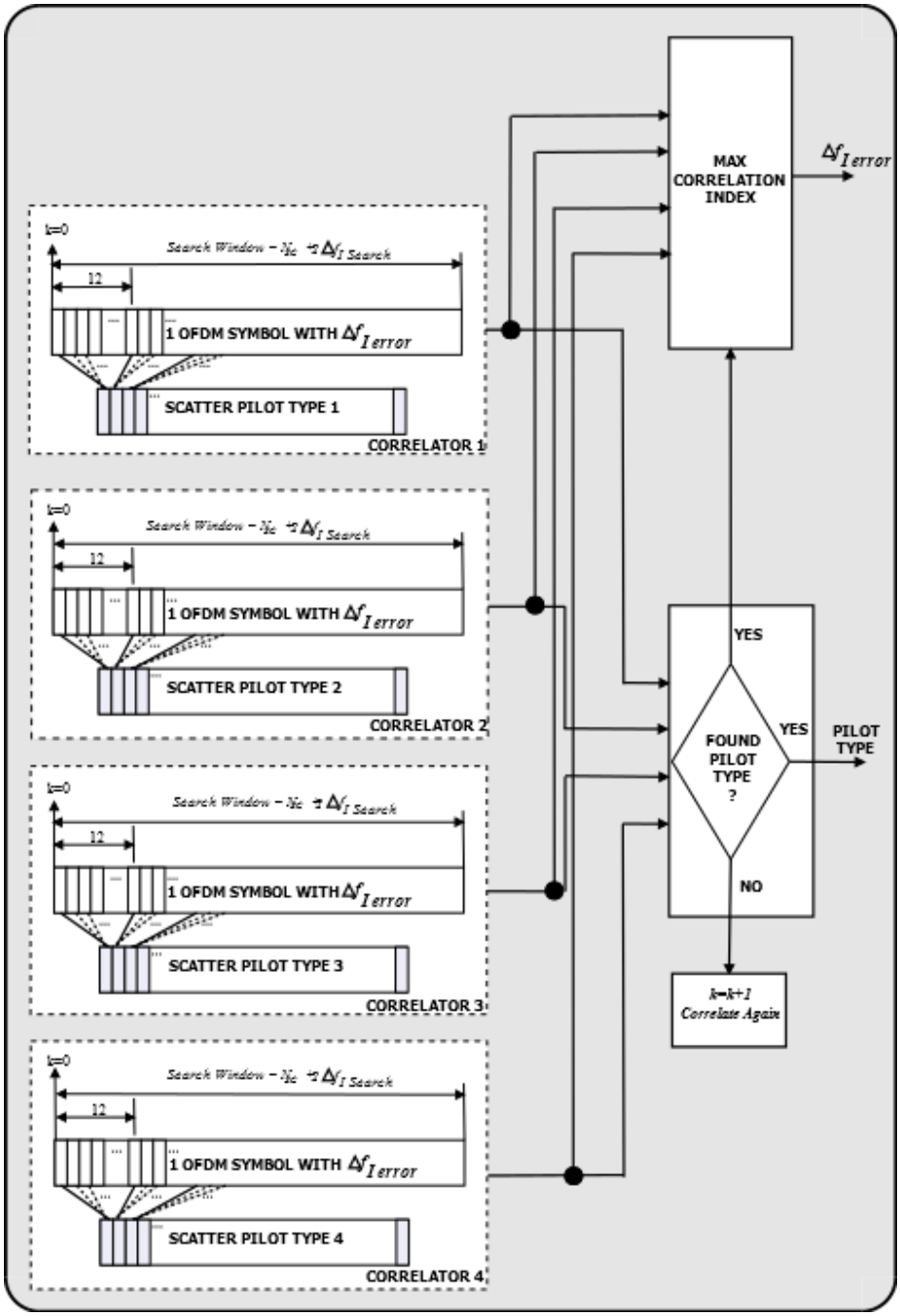


Figure 5.20: Estimation of integer frequency error and Scatter Pilot type detection.

hand, SCO generates a phase shift, that increases proportionally to the subcarrier index [92] [44]. SCO is an increasing positive or negative drift of the sample regarding the ideal sampling time, due to the clock *ppm* (see Figure 5.21). The phase rotations effect due to RCFO and SCO can be seen in Figures 5.22 to 5.25. Phase rotation caused by RCFO and SCO are uncorrelated, and can be computed independently. In this work I propose the use of the data-aided post-FFT joint RCFO/SCO estimator presented in [92] and [44]. Figure 5.26 shows the block diagram of the algorithm.

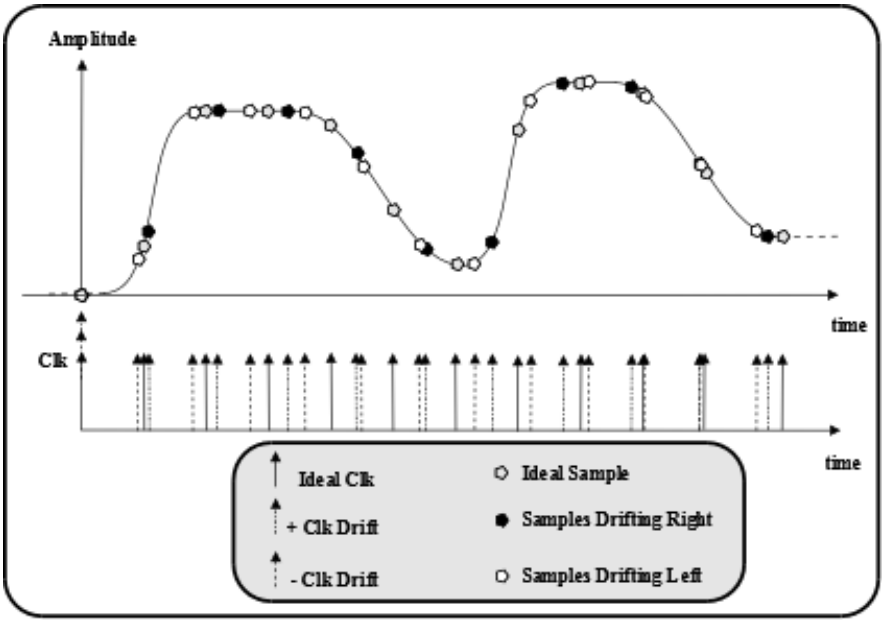


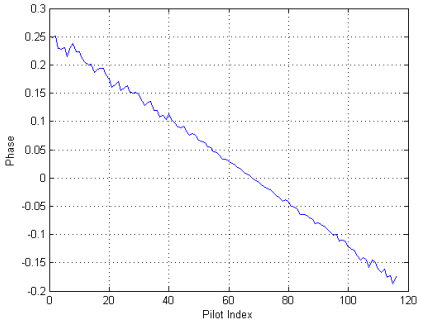
Figure 5.21: Ideal samples and sampling time, and samples drifting right or left, regarding the ideal samples, due to the positive and negative clock drift, respectively.

5.3.4.1 Estimation of RCFO and SCO

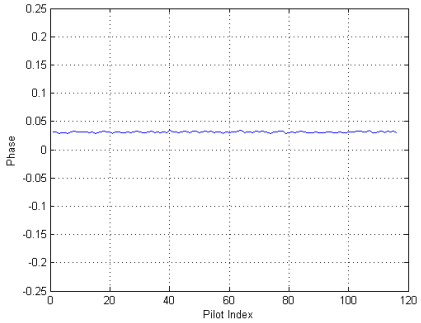
The received Pilot subcarriers of the OFDM Symbols can be represented using the following equation [92] [89]:

$$R_{l,k} = H_k P_{l,k} e^{j2\pi k t_\Delta l \frac{T_s}{T_u}} \tag{5.15}$$

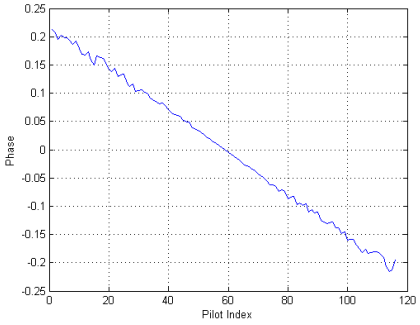
Where, t_Δ is the normalized timing offset, given by $t_\Delta = \frac{T'-T}{T}$, $P_{l,k}$ is the Pilot symbol, k is the subcarrier index, l is the OFDM Symbol index, and $H(k)$ is the



(a) SCO with 10 ppm and 1 Hz of CFO.

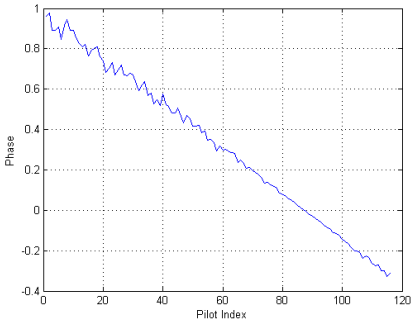


(b) Without SCO and with 1 Hz of CFO.

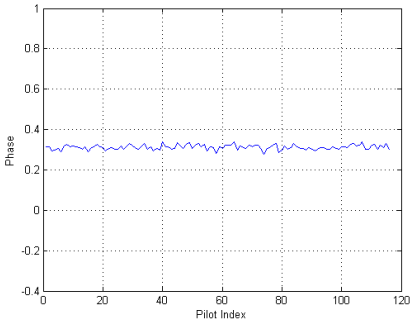


(c) SCO with 10 ppm, without CFO.

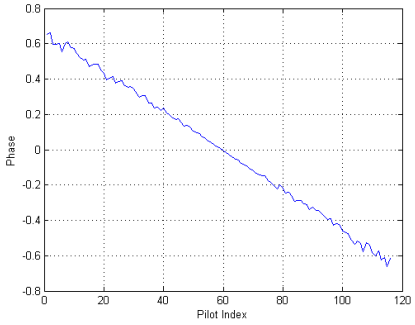
Figure 5.22: Phase rotation of pilots, caused by SCO and CFO for Mode 1



(a) SCO with 30 ppm and 10 Hz of CFO.

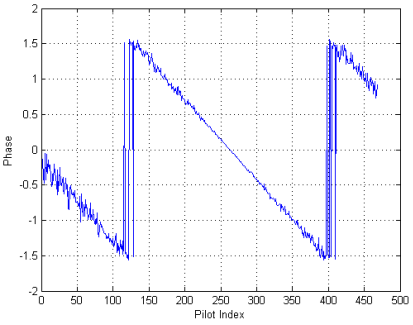


(b) Without SCO and with 10 Hz of CFO.

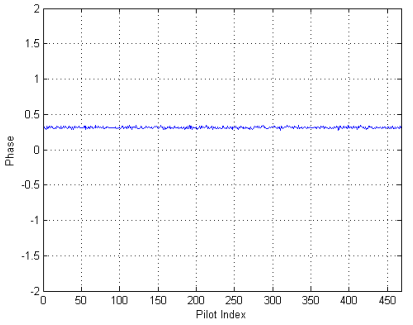


(c) SCO with 30 ppm, without CFO.

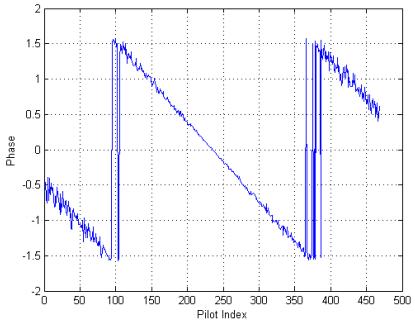
Figure 5.23: Phase rotation of pilots, caused by SCO and CFO for Mode 1



(a) SCO with 30 ppm and 10 Hz of CFO.

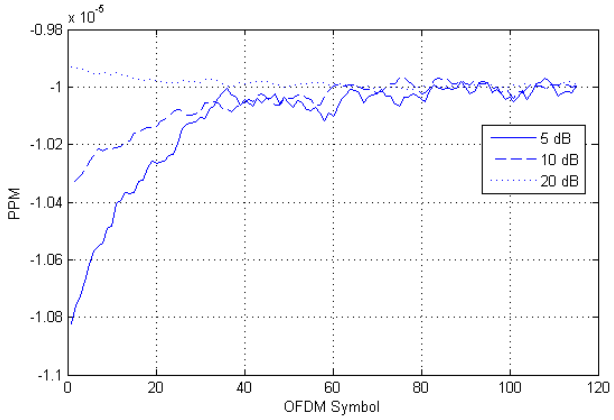


(b) Without SCO and with 10 Hz of CFO.

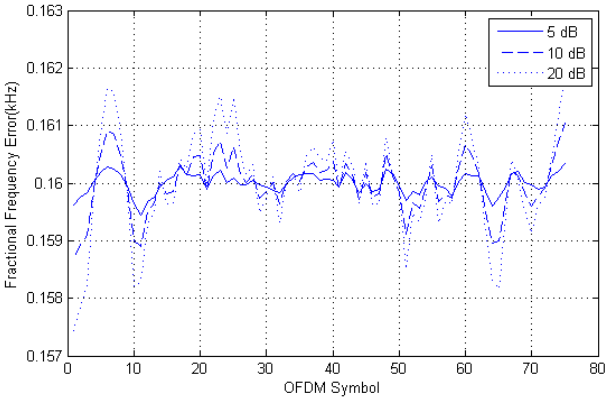


(c) SCO with 30 ppm, without CFO.

Figure 5.24: Phase rotation of pilots, caused by SCO and CFO for Mode 3.



(a) Estimation of clock error (in ppm): 10 ppm, Mode 3 for 5, 10 and 20 dB.



(b) Estimation of fractional frequency error: 160 Hz, Mode for 3 5,10 and 20 dB.

Figure 5.25: Estimation of clock error (in ppm) and fractional frequency error.

one-tap complex channel, which for simplicity's sake, is assumed to be constant over 4 consecutive OFDM Symbols.

To compute the rotation between 2 OFDM symbols separated by $l = 4$, which in ISDB-T systems is the distance between two Pilots of the same *Pilot Type*, it is necessary to multiply the two OFDM Symbols in a carrier-by-carrier basis as shown in Equation 5.16 [1].

$$\begin{aligned} Z_{l,k} &= R_{l,k} R^*_{l-4,k} \\ &= H_k P_{l,k} e^{j2\pi k t_\Delta l \frac{T_s}{T_u}} \left(H_k P_{l,k} e^{j2\pi k t_\Delta (l-4) \frac{T_s}{T_u}} \right)^* \\ &= {H_k}^2 \, {P_{l,k}}^2 \, e^{j2\pi k t_\Delta l \frac{T_s}{T_u}} e^{-j2\pi k t_\Delta (l-4) \frac{T_s}{T_u}} \\ &= {H_k}^2 \, {P_{l,k}}^2 \, e^{j2\pi k t_\Delta 4l \frac{T_s}{T_u}} \end{aligned} \tag{5.16}$$

With Equation 5.16 in hands the joint RCFO/SCO algorithm modified for ISDB-T systems, shown in Figure 5.26 works as follows:

1. The SPs of the same type, which are separated by 4 OFDM Symbols are extracted and used to fill two separate buffers. The elements of the bottom buffer are conjugated;
2. The elements of both buffers are multiplied, in an element-by-element basis;
3. The resulting multiplication feeds two secondary buffers with two sets of data, named C_1 and C_2 [92]. C_1 is the one with subcarrier index is greater than $K/2$ and C_2 is filled with subcarriers whose index smaller than K_2 and greater than 0. K is the Number of useful SP subcarriers, which is a Mode-dependent parameter;
4. Next, the accumulated phase over the two sets is are computed according to:

$$\phi_{1,l} = \arctan \left[\sum_{k \in C_1} Z_{l,k} \right] \tag{5.17}$$

$$\phi_{2,l} = \arctan \left[\sum_{k \in C_2} Z_{l,k} \right] \tag{5.18}$$

In this work, the angle is computed using CORDIC algorithm working in *Vectoring Mode* with *Circular Coordinates* (see Table 5.1).

[1] In Equation 5.16, I slightly modified the equation shown in [92], changing the term $(l-1)$ by $(l-4)$, due to the different Pilot structure of ISDB-T regarding DVB-T.

5. Finally, RCFO and SCO are computed as follows:

$$RCFO = \frac{1}{2\pi\,(1+Ng/N)}\frac{1}{2}(\phi_{1,l}+\phi_{2,l}) \tag{5.19}$$

$$SCO = \frac{1}{2\pi\,(1+Ng/N)}\frac{1}{4K/2}(\phi_{1,l}-\phi_{2,l}) \tag{5.20}$$

Figures 5.22 to 5.24(c) show the plots of the phase computation over the Pilot subcarriers of $Z_{l,k}$ (see Equation 5.16), for several ISDB-T parameters, SCO and RCFO. Figure 5.25, present the SCO and RCFO estimation using the joint RCFO/SCO algorithm for different values of SNR.

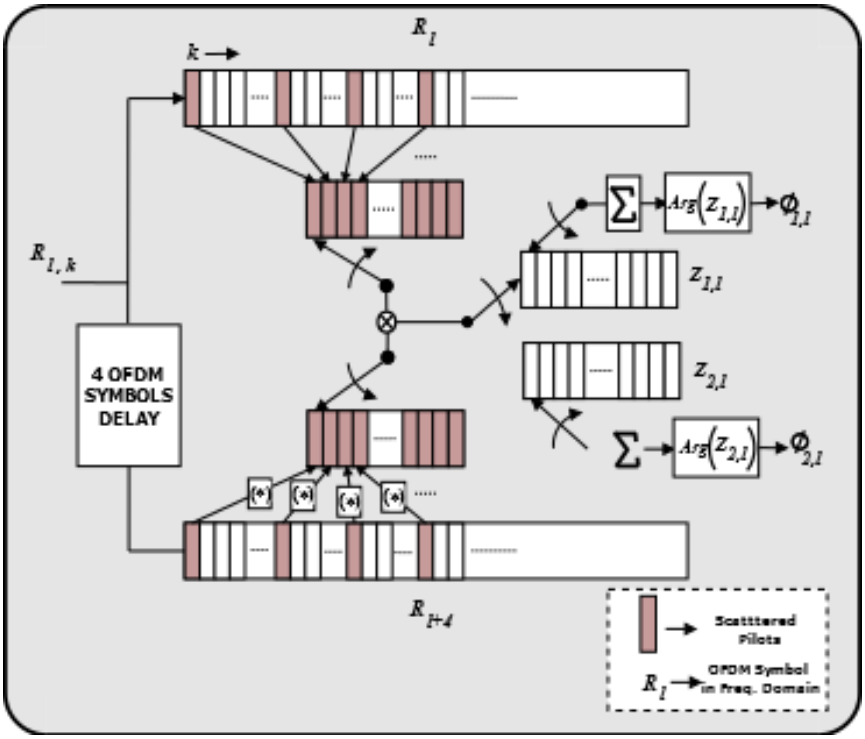


Figure 5.26: Block diagram of the joint SCO/RFCO detection algorithm.

5.3.4.2 Correction of RCFO and SCO

The RCFO and SCO values, computed using the algorithm presented in Figure 5.26 and described in the previous Section, are filtered by a second order loop filter that

have the following transfer function $H(z) = \mu_1 + \frac{\mu_2}{1-Z^{-1}}$, in order to reduce the high frequency noise. μ_1 and μ_2 are used to control the convergence speed and the closed loop bandwidth. The filtered RCFO and SCO, feed a CORDIC de-rotator and a Resampler respectively. The Resampler is made of a Fractional Interpolator based on Farrow structure , using Lagrange Polynomial interpolation [93] [94], and a Decimator as shown in Figure 5.1. The fractional interpolator compensates two errors. The first, is the SCO caused by the crystal *ppm*, and the second is a constant error generated by the clock excess regarding 3.69140625 times the basic ISDB-T sampling clock, which is $512/63\ MHz$. The clock excess occurs due to the difference between 30 MHz, which is the USB clock, and $512/63\ MHz$ that is the output clock of the block containing the Farrow Interpolator and the Decimator.

Lagrange interpolation is probably the easiest way to design an FIR filter to approximate a fractional delay [95]. The Lagrange coefficients are obtained from Equation 5.21. Table 5.2 shows how to compute the 1st to 4th order Lagrange coefficients for 5 coefficient filters, considering the first sample index is equal to 0 and the last to 4. In the work presented in this thesis, I used the third order Lagrange interpolation with the interpolants computed as shown in Figure 5.27. The commonly used Farrow coefficients for third order fractional Lagrange interpolators, are presented in Figure 5.28. They are computed as shown in Equations 5.22 to 5.25, using Equation 5.21, for $n = -1\ to\ 2$ and $k = 0 : N-1$.

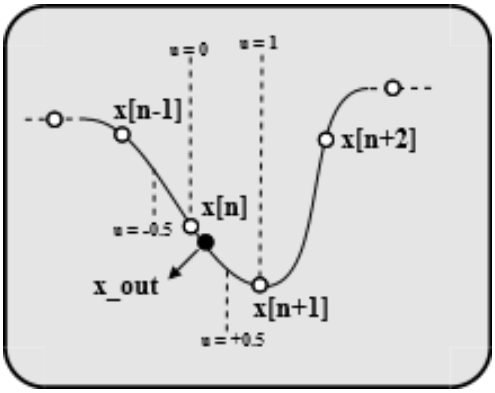


Figure 5.27: Interporlant generation using cubic Farrow Interpolator, given the factional delay μ.

$$h(n) = \prod_{k=0}^{N} \frac{D-k}{n-k} \quad \text{for} \quad n = 0, 1, 2, ...N \quad \text{and} \quad k \neq n. \tag{5.21}$$

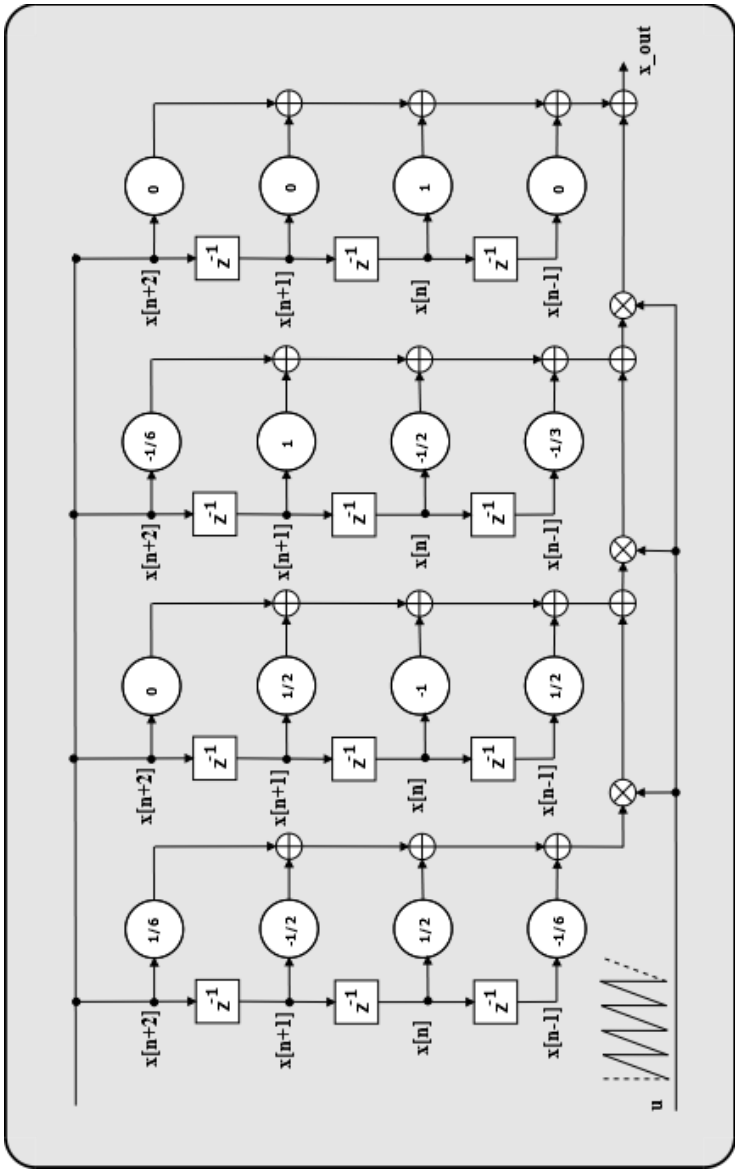


Figure 5.28: Farrow structure for cubic Lagrange interpolation.

H

Table 5.2: Lagrange Coefficients for 1st to 4th Order.

	h(0)	h(1)	h(2)	h(3)	h(4)
N=1	1-D	D			
N=2	(D-1)(D-2)/2	-D(D-2)	D(D-1)/2		
N=3	-(D-1)(D-2)(D-3)/6	D(D-2)(D-3)/2	-D(D-1)(D-3)/2	D(D-1)(D-2)/6	
N=4	(D-1)(D-2)(D-3)(D-4)/24	D(D-2)(D-3)(D-4)/6	D(D-1)(D-3)(D-4)/4	-D(D-1)(D-2)(D-4)/6	D(D-1)(D-2)(D-3)/24

$$
\begin{aligned}
h(n=2) &= \frac{(D-(-1))}{(2-(-1))}\frac{(D-0)}{(2-0)}\frac{(D-1)}{(2-1)} \\
&= \frac{(D+1)}{3}\frac{(D)}{2}\frac{(D-1)}{1} \\
&= \frac{D^3}{6} - \frac{D}{6}
\end{aligned} \tag{5.22}
$$

$$
\begin{aligned}
h(n=1) &= \frac{(D-(-1))}{(1-(-1))}\frac{(D-0)}{(1-0)}\frac{(D-2)}{(1-2)} \\
&= \frac{(D+1)}{2}\frac{(D-0)}{1}\frac{(D-2)}{-1} \\
&= -\frac{D^3}{2} + \frac{D^2}{2} + \frac{D}{1}
\end{aligned} \tag{5.23}
$$

$$
\begin{aligned}
h(n=0) &= \frac{(D-(-1))}{(0-(-1))}\frac{(D-1)}{(0-1)}\frac{(D-2)}{(0-2)} \\
&= \frac{(D+1)}{2}\frac{(D-1)}{2}\frac{(D-2)}{2} \\
&= \frac{D^3}{2} - \frac{D^2}{1} + \frac{D^2}{2} + 1
\end{aligned} \tag{5.24}
$$

$$
\begin{aligned}
h(n=-1) &= \frac{(D-0)}{(-1-0)}\frac{(D-1)}{(-1-1)}\frac{(D-2)}{(-1-2)} \\
&= \frac{(D)}{-1}\frac{(D-1)}{-2}\frac{(D-2)}{-1-2} \\
&= -\frac{D^3}{6} + \frac{D^2}{2} - \frac{D}{3}
\end{aligned} \tag{5.25}
$$

Using Equation 5.21, and following the example presented in Figure 5.28, it is straightforward to extend the Farrow interpolator for other orders[1].

The SCO correction mechanism, shown in Figure 5.29, works as follows:

- The estimated SCO, t_Δ, is filtered by a second order loop filter;
- The filtered t_Δ, which is a small positive or negative number is added to 1. This value feeds the Accumulator, that in the literature is called NCO due to the oscillating behavior of its output which varies between 0 and 1. Figure 5.25(b) shows the output of the loop filter for the estimation of +10 *ppm*;

[1] Matlab has embedded functions that can be used to compute the Farrow Filter coefficients for several orders.

- Four samples feed the Fractional interpolator internal registers;
- The output of the Accumulator is summed with its input. The result value can be smaller than 1, greater than 1 or greater that 2;
- If the resulting value is greater than 2, 2 is subtracted and the μ value is obtained. The samples in the Fractional Interpolator are shifted by two and two new samples feed the last registers. At the same time a valid corrected sample is obtained at the output of the interpolator;
- If the resulting value is greater than 1 and smaller than 2, 1 is subtracted and the μ value is obtained. The samples in the Fractional Interpolator are shifted by one and a new sample feeds the last register. At the same time a valid corrected sample is obtained at the output of the interpolator;
- Finally, if the resulting value is smaller than 1, it is the μ that is used by the Fractional Interpolator. The samples in the Fractional Interpolator registers are kept. At the same time a valid corrected sample is obtained at the output of the interpolator.

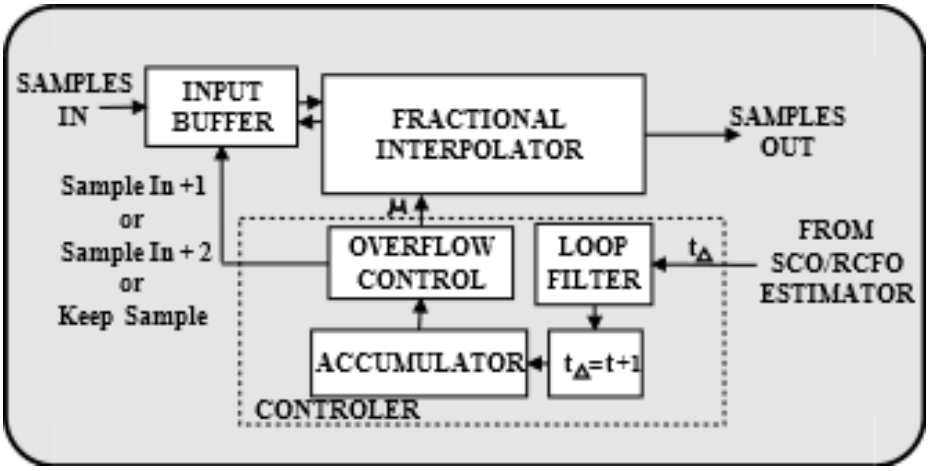


Figure 5.29: SCO correction mechanism.

5.3.4.2.1 Farrow Interpolation Advantages and Disadvantages

The main advantage of Farrow interpolation is that the filter coefficients are fixed. The only parameters that vary are the fractional interval, i.e. μ, and the control

signal that makes the sample shifts by 1, 2 or 0. Among the known disadvantages (see [96]), I can mention, the unchangeable frequency response. But it is possible to handle that behavior increasing the Sample Rate[1] at the input of Farrow Interpolator, and that was the case in this work. In addition, the out of band attenuation can not be improved easily by increasing the interpolation degree. Therefore, the gain in the system performance is negligible [96]. That is the reason because I adopted the third-order interpolation.

5.3.5 Frequency Domain Equalizer

Channel Estimation is performed to recover the phase and amplitude of the subcarriers that were rotated and distorted by the wireless channel, imperfect synchronization and filters. The overall Channel Estimation mechanism is based on the Channel Frequency Response (CFR) estimation using the SPs of four OFDM Symbols, followed by interpolation by an integer factor of 4, in the frequency dimension, as can be seen in Figures 5.30 and 5.31. It is assumed that the channel response does not vary within the four OFDM Symbols, which is granted for fixed-reception receivers with antennas on the roof of the house. For low mobility and fixed-reception devices, such as notebooks, this may not be assumed, at least for a short period, due to the moving of local scatters in the receiver vicinity. Figures 5.33, 5.34, 5.35 and 5.36 show examples of CFR for static channels, obtained by averaging 100 channel realizations, for ISDB-T working on Mode 3, with C/N equal to 25 dB, for ETSI, Brazil-A, Brazil-B and Brazil-C channels respectively. The PDPs of ETSI channel and, Brazil channels A, B and C are found in [2] and [5], respectively.

5.3.5.1 The Channel Estimator and the Equalization

In this work I propose the use of a CORDIC-based Least-Squares (Zero-Forcing) Channel Estimator [97] [98], followed by a simple complex multiplication (1 tap equalization) to equalize the TMCCs, ACs and Data subcarriers. The estimation/equalization scheme is shown in Figure 5.32 and works as follows:

- Starting from an OFDM Symbol containing embedded Pilot Type SP1, 4 post-FFT OFDM Symbols are stored, per time, in the input buffer;
- Starting from the first pilot of the first OFDM Symbol, which is placed at subcarrier with index $k = 0$, i.e. $SP1_e(k = 0)$, until the last pilot of the fourth

[1] The architecture proposed in this thesis works with 3.69140625 times the basic ISDB-T Sample Rate (512/63 MHz). That number was shown to be enough.

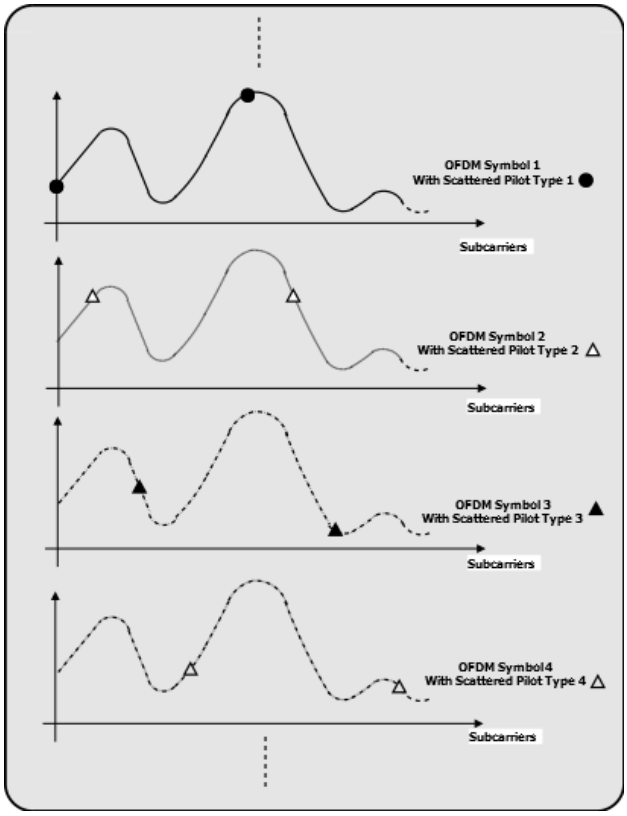


Figure 5.30: Frequency response for four consecutive OFDM symbols.

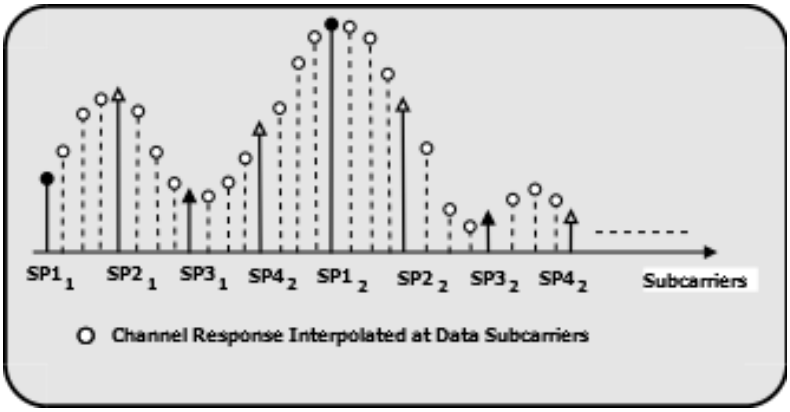


Figure 5.31: Channel interpolation using Farrow structure and third order Lagrange interpolation filters.

OFDM, the corrupted embedded pilots are extracted[1]. For each extracted scattered pilot, its squared absolute value is computed and compared to $2 \times 1.3333...$, in order to guarantee that the CORDIC division is always going to be smaller than 2;

- If the squared absolute values of the extracted pilots, i.e. $|SPX_e(k)|^2$ (where X can be 1, 2, 3 or 4) are greater than $2 \times 1.3333...$, $|SPX_e(k)|^2$ is shifted left[2] until that condition is achieved. For each shift the variable **exp** (see Figure 5.32) is incremented. This variable is used to shift left the squared absolute values of the distorted pilots and to recovery the scale (left shift again) after the CORDIC-based division;

- If, at the first check, the value of $|SPX_e(k)|^2$ is smaller or equals to $2 \times 1.3333...$ the value of the variable **exp** (see Figure 5.32) is 0, *i.e.* no shift is performed;

- After the division the results are multiplied by the conjugate of $SPX_e(k)$, buffered and interpolated by 4, in order to compute the CFR in the TMCCs, ACs and Data subcarriers;

- Finally, the resulting estimation is used to Equalize the corrupted data subcarriers;

- Four new post-FFT OFDM Symbols are stored in the input buffer, **exp** is set to 0 and the entire process is repeated.

[1]The distorted pilots, that originally are $\pm 1.33333...$, now are complex numbers.
[2]The operator $<<$ performs the shift to left.

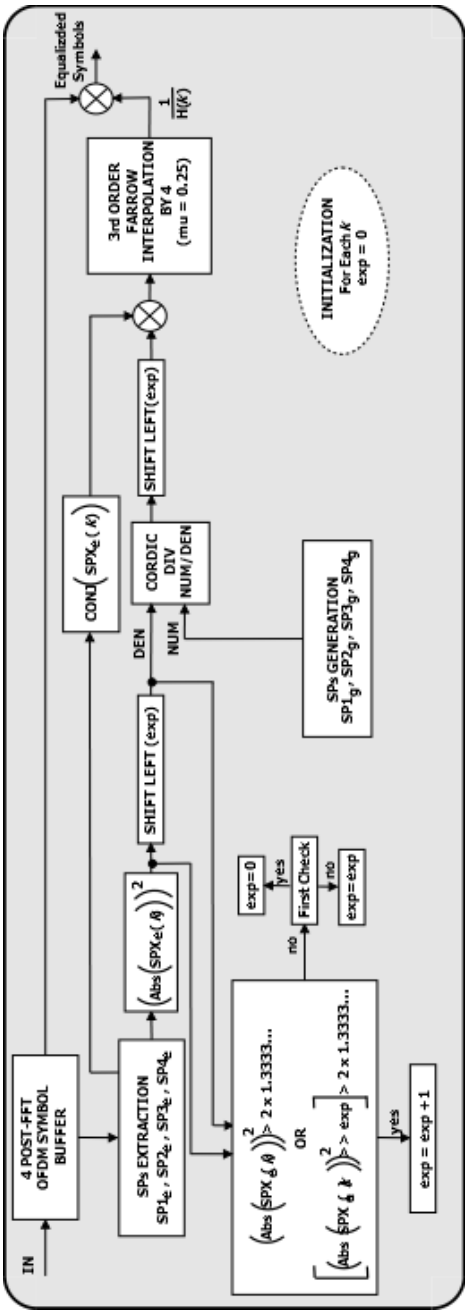


Figure 5.32: The CORDIC-based channel estimation scheme.

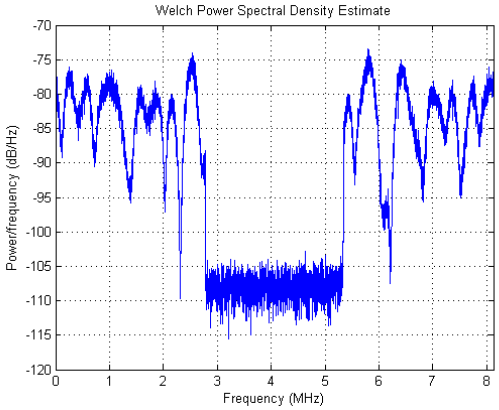


Figure 5.33: Frequency response of the average of 100 realizations of *ETSI* channels.

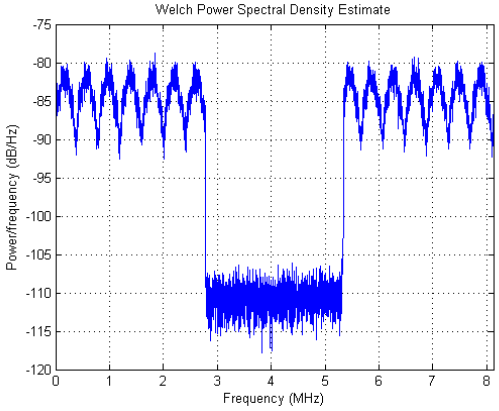


Figure 5.34: Frequency response of the average of 100 realizations of *Brazil A* channels.

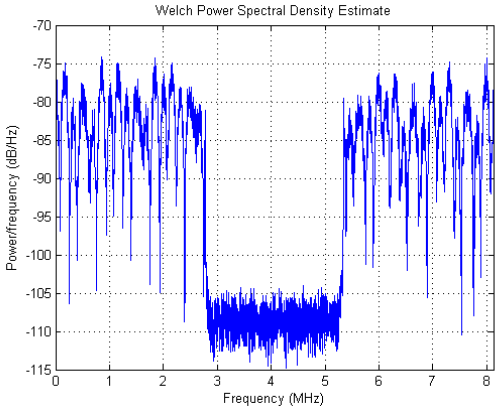


Figure 5.35: Frequency response of the average of 100 realizations of *Brazil B* channels.

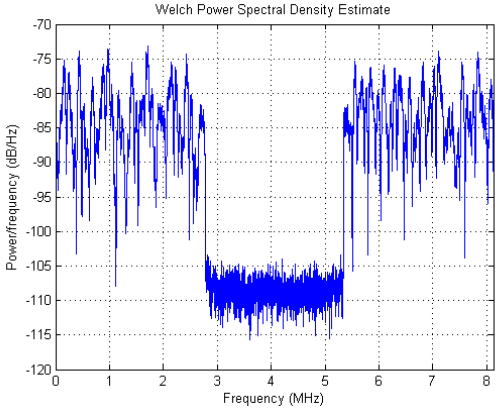


Figure 5.36: Frequency response of the average of 100 realizations of *Brazil C* channels

5.3.6 Soft-Demapper With CSI

Viterbi Decoding algorithm [99] [100] is a method widely used to decode convolutional codes [3]. It is an efficient implementation of the Maximum Likelihood (ML) decoding, because it avoids the need to consider every possible transmitted coded sequence, avoiding extensive search as in the ML decoding case. It can use hard decision or soft decision decoding strategies. In the hard decision case the algorithm searches, trough the trellis, for the path that contains the minimum number of different bits regarding the hard-decided received sequence, and detects the code sequence which minimizes the Hamming distance. Soft Viterbi decoder makes decision using soft information generated by a soft demappping strategy, where the demapper creates a vector containing soft information instead of hard bits based on hard decision. When Channel State Information (CSI) is available, it can be used to improve the performance of soft decision Viterbi algorithm. This is called soft CSI decision decoding [3].

The Frequency Domain equalizer, especially the ZF kind, significantly amplifies the noise at those subcarriers that suffers from high attenuation. Viterbi decoder path metric can be severely affected by this effect. When using CSI this effect is attenuated because of the magnitude of the channel is taken into account, e.g. a hard bit "'0"'(or "'1"'), that belongs to a severely attenuated subcarrier can become a weak soft bit "'0"' (or a weak "'1"'), when CSI is applied.

In this thesis, I propose the use of the soft demapping strategy presented in [101], with 5 soft-bits per modulation bit. The simplified strategy proposed therein, named Simplified Log-Likelihood Ratio(LLR), does not make use of noise estimation, as used to compute the conventional LLR. Afterward, each one of the soft-bits are weighted by the CSI, which is the inverse of of the channel estimation, as shown in Figure 5.37. The proposed demapper can either work in hard or soft mode. The same stands for the Viterbi decoder.

Following [3], and applying the nomenclature used therein, the portion of the ISDB-T receiver related to soft demapping and Viterbi Decoding can be represented as shown in Figure 5.37; where l is the OFDM Symbol index. 204 OFDM Symbols form an OFDM Frame[1]. k is the OFDM subcarrier index and m is the bit index belonging to the baseband modulation symbol made of M bits.

[1]The Viterbi Decoder is reset at the beginning of each OFDM Frame.

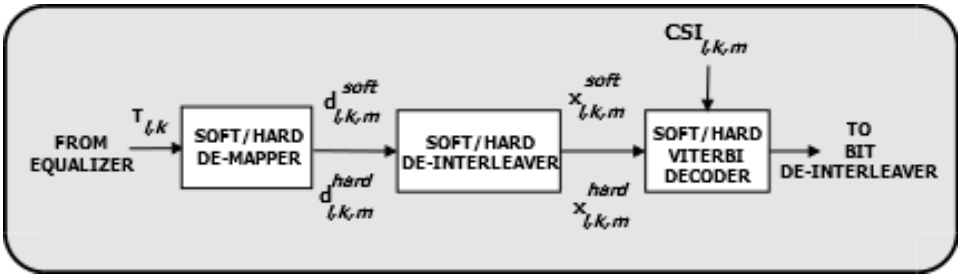


Figure 5.37: Soft-Demapper and Soft Viterbi Decoder interconnection, as shown in [3].

If the Soft Demapper plus soft CSI Viterbi decoding were implemented as shown in Figure 5.37, the length of the Time Deinterleaver bus would be the number of soft bits from the demapper, *i.e.* 5, plus the the number of bits of the CSI, that was also chosen to be 5. Forcing the Time Deinterleaver bus length to 10 bit. This would make the ASIC memory area almost double its size and the area of the chip, which is limited by the Time Deinterleaver memory area[1], rendering the ASIC economically unfeasible. The simple solution was to implement the CSI scaling before the Viterbi Decoder, so the number of bits of the Time Deinterleaver became only 5 bits. The chosen architecture of the Soft Demapper with CSI became the one displayed in Figure 5.38, which is based on the normalized CSI regarding the maximum $CSI_{l,k}$, or in other words, the minimum $H_{l,k}$.

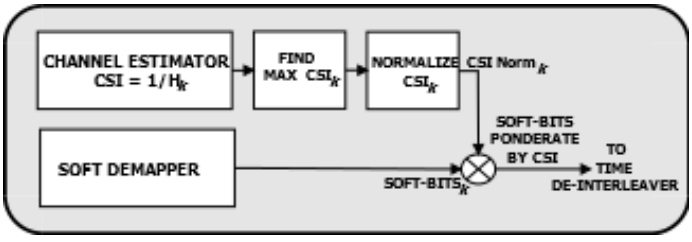


Figure 5.38: Soft Demapper with bits weighted by the normalized Channel State Information (CSI).

[1]This is shown in Chapter 6

5.3.6.1 The Soft Bits Computation

According to the equations presented within [101], the soft bits for QPSK, 16-QAM and 64-QAM modulations can be obtained using Equations 5.26 to 5.31. The mentioned equations can be applied to the in-phase ($y_I[i]$), as presented below, and to the quadrature ($y_Q[i]$) components of the equalized symbols, resulting in the soft bits $D_{I,k}$ and $D_{Q,k}$.

5.3.6.1.1 QPSK

$$D_{I,1} = y_I[i], \qquad |y_I[i] \leq 2|. \tag{5.26}$$

5.3.6.1.2 16-QAM

$$D_{I,1} = \begin{cases} y_I[i], & |y_I[i]| \leq 2 \\ 2(y_I[i]-1), & y_I[i] > 2 \\ 2(y_I[i]+1), & y_I[i] < -2 \end{cases} \tag{5.27}$$

$$D_{I,2} = -|y_I[i]| + 2. \tag{5.28}$$

5.3.6.1.3 64-QAM

$$D_{I,1} = \begin{cases} y_I[i], & |y_I[i]| \leq 2 \\ 2(y_I[i]-1), & 2 < y_I[i] \leq 4 \\ 3(y_I[i]-2), & 4 < y_I[i] \leq 6 \\ 4(y_I[i]-3, & y_I[i] > 6 \\ 2(y_I[i]+1), & -4 \leq y_I[i] < -2 \\ 3(y_I[i]+2), & -6 \leq y_I[i] < -4 \\ 4(y_I[i]+3, & y[i] < -6 \end{cases} \tag{5.29}$$

$$D_{I,2} = \begin{cases} 2(-|y_I[i]|+3), & |y_I[i]| \leq 2 \\ 4-|y_I[i]|, & 2 < |y_I[i]| \leq 6 \\ 2(-|y_I[i]|+5), & |y_I[i]| > 6 \end{cases} \tag{5.30}$$

$$D_{I,3} = \begin{cases} |y_I[i]| - 2, & |y_I[i]| \leq 4 \\ -|y_I[i]| + 6, & |y_I[i]| > 4 \end{cases} . \tag{5.31}$$

Nevertheless, it was observed that the above equations could be simplified to those presented below, without significant losses in the BER performance. As in the case of Equations 5.26 to 5.31, the soft bits for in-phase and quadrature components can be computed using the same equations.

5.3.6.1.4 QPSK

$$D_{I,1} = y_I[i]. \tag{5.32}$$

5.3.6.1.5 16-QAM

$$D_{I,1} = y_I[i] \tag{5.33}$$

$$D_{I,2} = -|y_I[i]| + 2. \tag{5.34}$$

5.3.6.1.6 64-QAM

$$D_{I,1} = y_I[i] \tag{5.35}$$

$$D_{I,2} = -|y_I[i]| + 4 \tag{5.36}$$

$$D_{I,3} = -||y_I[i]| - 4| + 2. \tag{5.37}$$

5.3.7 Joint TMCC Detection and Frame Synchronization

In order to detect the ISDB-T transmission parameters and allow their use to automatically configure the receiver blocks with the correct parameters (e.g. baseband modulation type, code rate and emergency flag, etc), TMCC detection shall be accomplished. ISDB-T frame configuration forces the existence of a significant number of TMCC as shown in Sections 3.1.2.17 and 3.1.2.15. This permits the detection of TMCC parameters under very low SNR scenario. Techniques, such as Maximal Ration Combing (MRC), could be used to even improve the TMCC detection performance. Nevertheless, as the main goal of this work is to implement a low complexity receiver that targets HDTV signals, and the number of OFDM Segments used for such scenarios always allows a large number of TMCCs, the detection scheme I propose is a simple average over a configurable number of TMCCs. The joint OFDM Frame synchronization and TMCC parameters detection scheme are shown in Figure 5.39, and works as follows:

1. The use of coherent modulation on all subcarriers is assumed. Therefore, the index of the subcarriers that transport a TMCC is known beforehand and can be stored in a table at the receiver;

2. Starting from the first OFDM Segment until the last segment, the TMCC symbols $TMCC_{l,k}$, are accumulated in an OFDM Symbol basis, and averaged using a configurable parameter M. l is the OFDM Symbol index, k is the subcarrier index that contains a valid TMCC symbol and M is the number of TMCCs used to obtain the average. The result of this computation is a vector containing the $TMCC_{l,1}$ samples;

3. The first element of the vector, $TMCC_{1,1}$ is considered to be the reference bit for differential decoding;

4. After generating the $TMCC_{l,1}$ elements and performing the differential decoding, the next task is to find the synchronization words **W0** and **W1** (see Section 3.1.2.17) in order to detect the OFDM Frame boundary;

5. If **W0** word is found and 204 symbols later the a **W1** word is also found or vice-versa, the frame border is found and OFDM Frame synchronization is achieved. Otherwise the reference bit is shifted by 1 and a new search for **W0** and **W1** is performed, until the search is successful;

6. A sequence **W0**, **W1**, **W0**, ... or **W1**, **W0**, **W1**, ... shall be found, otherwise Frame Synchronization has failed and the overall search shall restart on step 2;

7. After a successful Frame Synchronization the next step is to check the parity of the TMCC. If the parity is 1 the TMCC is considered a valid TMCC, otherwise the overall synchronization shall restart on step 2;

8. After a successful Frame Synchronization and if the check of the parity is OK, the next task is to extract the TMCC parameters and make them available to configure the receiver;

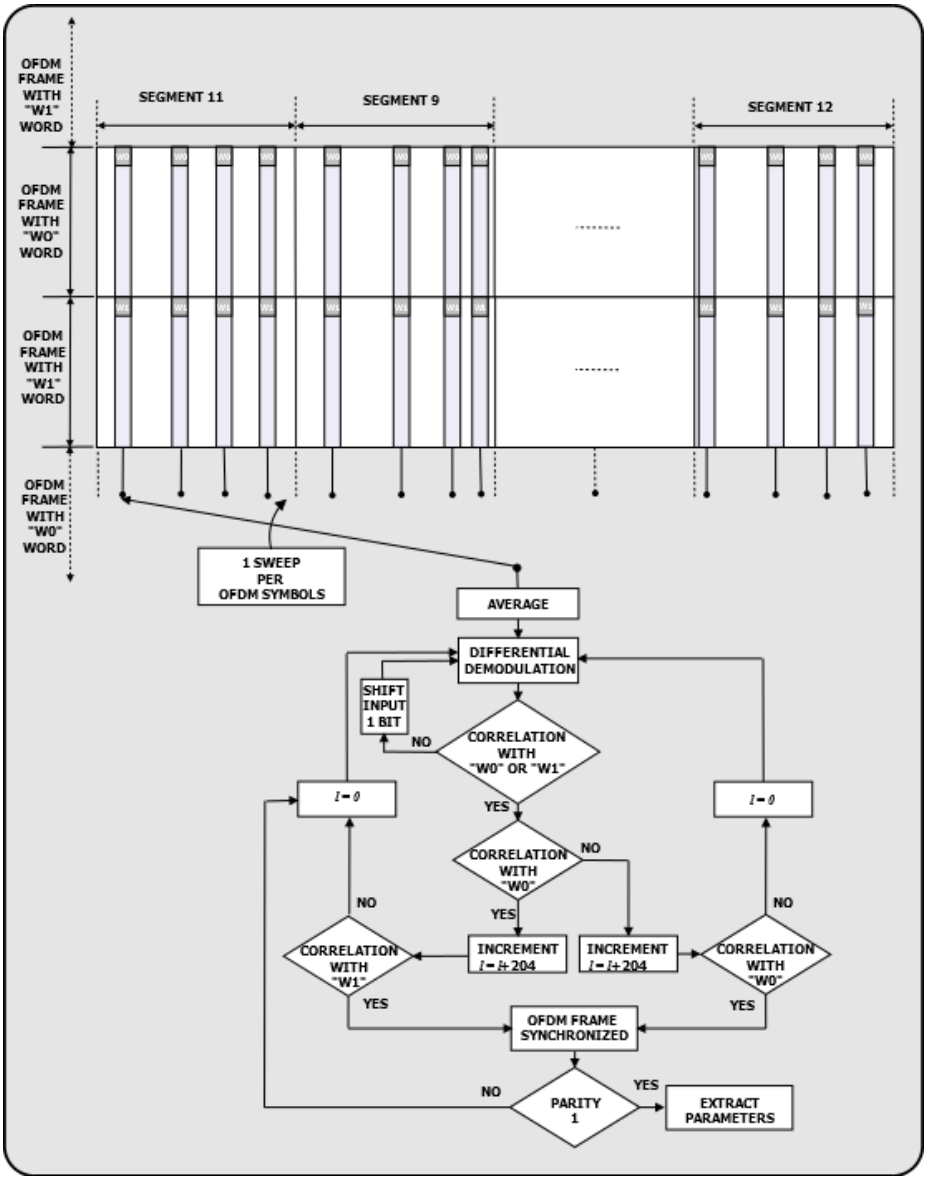


Figure 5.39: TMCC detection.

5.3.8 SNR Estimator

The SNR estimator proposed in this work is very straightforward. It is based on the Frequency-Time SNR estimator shown in [102] and references therein. Nevertheless, the version presented in this thesis makes use of the estimation of the noise in the ACs and TMCCs subcarriers during a configurable number of OFDM Symbols. The number of OFDM Symbols multiplied by the number of TMCCs and ACCs is given by the parameter M. The SNR is obtained by dividing, using the CORDIC algorithm, the power of the ACs and TMCCs by the power of the noise estimated on each subcarriers within an OFDM Symbol. This is done over a number of OFDM Symbols, followed by the averaging. A simplified vision of the process is given in Figure 5.40.

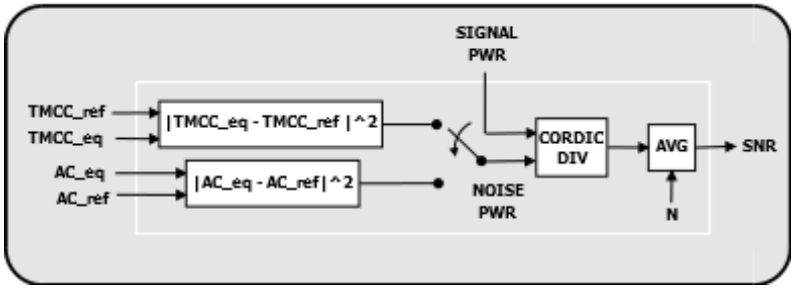


Figure 5.40: CORDIC-based SNR estimator, using AC pilots and TMCC symbols.

5.3.9 Soft Time Deinterleaver

The Time Deinterleaver used in ISDB-T is the most memory consuming block within the receiver. Due to the use of Soft-Demapping with CSI the number of bits within the Time Deinterleaver grows significantly as the number of soft-bits with CSI increases. As displayed in Chapter 6 the memory dominates the area of the ASIC. Most of the memory presented therein is due to the Time Deinterleaving process. Therefore the definition of the number of soft-bits affects the area cost and consequently the cost of the ASIC.

5.3.10 BER Estimator

5.3.10.1 Post Viterbi Decoder Bit Error Rate (BER)

In the FPGA and ASIC hardware implementations, the measurement of post Viterbi Decoder BER is performed using the number of bits corrected by the Reed-Solomon Decoder, for a configurable number of MPEG-TS Packets. The number of corrected bits is accumulated over a configurable number of TS Packets and the BER is obtained by dividing the number of accumulated corrected bits by the number of bits decoded by the Reed-Solomon Decoder. The approximate Post Viterbi Decoder BER is measurable at each one of the layers A, B and C, using the following Equation:

$$BER_{VIT\ OUT} = \frac{Number\ of\ Corrected\ Bits \times 8}{Total\ Number\ of\ Packets \times 204 \times 8} \tag{5.38}$$

Where,

- *Number of Corrected Bits*, is the number of bits corrected by RS decoder within a measurement cycle;
- *Total Number of Packets*, is the configurable number of MPEG-TS packets used to measure the BER.

Assuming that all packets used for the BER measurement have are uncorrectable, i.e. there are packets with a number of bytes with error equal to 9, the equations will achieve a BER saturation level.

5.3.10.2 Post Reed-Solomon BER

Assuming that the occurrence of more than 9 bytes errors within a TS packet is low, the post RS Decoder BER can be measured using as reference the number of uncorrectable bytes after RS Decoder, *i.e.* 9. Therefore, when the TS packet can not be correct it means that it has 9 byte errors. So, approximate BER at RS decoder can be computed using the following Equation:

$$BER_{RS\ OUT} = \frac{Number\ of\ Packets\ With\ Errors \times 9 \times 8}{Total\ Number\ of\ Packets \times 204 \times 8} \tag{5.39}$$

Where,

- *Number of Packets With Errors*, is the number of MPEG-TS packets with more than 9 bytes with errors;
- *Total Number of Packets*, is the configurable number of MPEG-TS packets used to measure the BER.

As in the case of post Viterbi decoder BER, the post RS decoder BER can be measured in each of ISDB-T layers.

5.4 Simulation Results

In this section I present several performance curves for the model receiver with different ISDB-T configuration parameters. Figures 5.41 to 5.48 show the effects of the wireless channels, named Brazil-A and Brazil-B, whose PDP are shown in Table 5.3 and are described in [5]. The performance is evaluated after Viterbi and RS Decoders. Figure 5.49 shows an example of performance improvement achieved when using Time Interleaving, under Impulsive Noise. Figures 5.50 and 5.51 present the effects of the ADC quantization in the post Viterbi and post RS decoders BER, for AWGN and Brazil-A channel respectively. Finally, Figure 5.52 presents the convergence of SCO, RCFO and Fine Boundary algorithm under AWGN, with C/N = 25 dB.

Table 5.3: Power Delay Profiles of Brazil A and Brazil B Channels, According to ITU Guideline for Digital Terrestrial Television Evaluation [5].

TAP	BRAZIL A		BRAZIL B	
	DELAY (us)	Attenuation (dB)	DELAY (us)	Attenuation (dB)
1	0	0	0	0
2	0.15	-13.8	0.30	-12.0
3	2.22	-16.2	3.50	-4.0
4	3.05	-14.9	4.40	-7.0
5	5.86	-13.6	9.50	-15.0
6	5.93	-16.4	12.7	-22.0

5.4.1 Performance Over Brazil-A and Brazil-B Wireless Channels

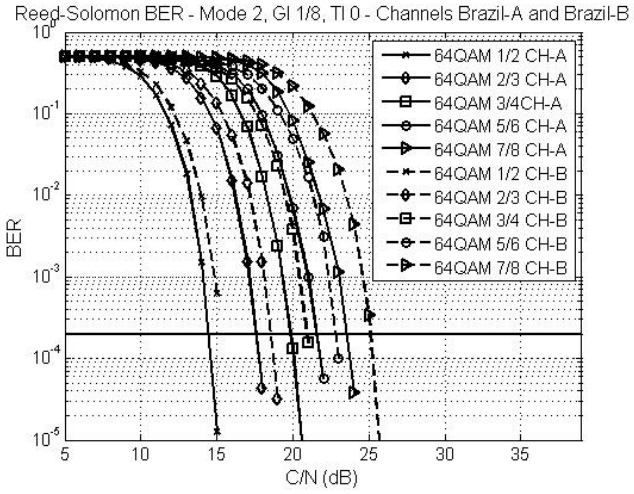


Figure 5.41: Model receiver performance after RS decoder, over Brazil-A and Brazil-B channels, for Mode 2, GI 1/8, Time Interleave 0, 64 QAM.

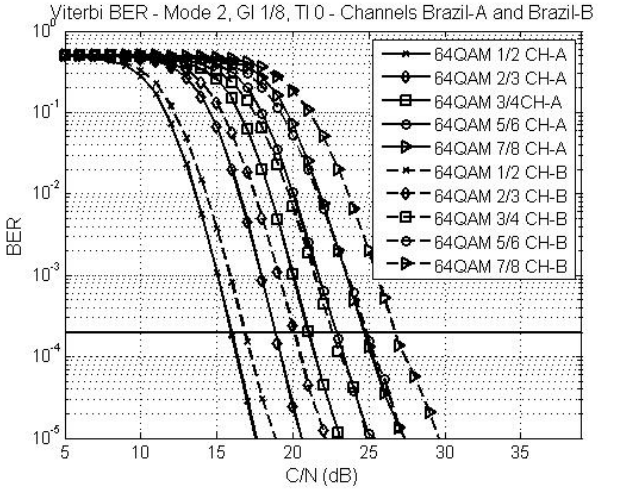


Figure 5.42: Model receiver performance after Viterbi Decoder, over Brazil-A and Brazil-B channels, for Mode 2, GI 1/8, Time Interleave 0, 64 QAM.

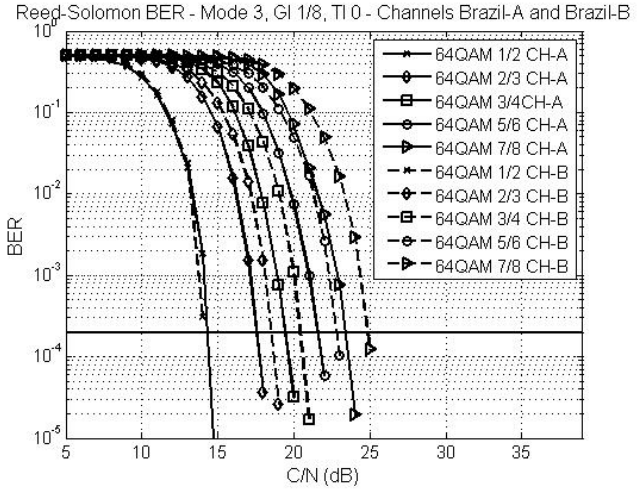


Figure 5.43: Model receiver performance after RS Decoder, over Brazil-A and Brazil-B channels, for Mode 3, GI 1/8, Time Interleave 0, 64 QAM.

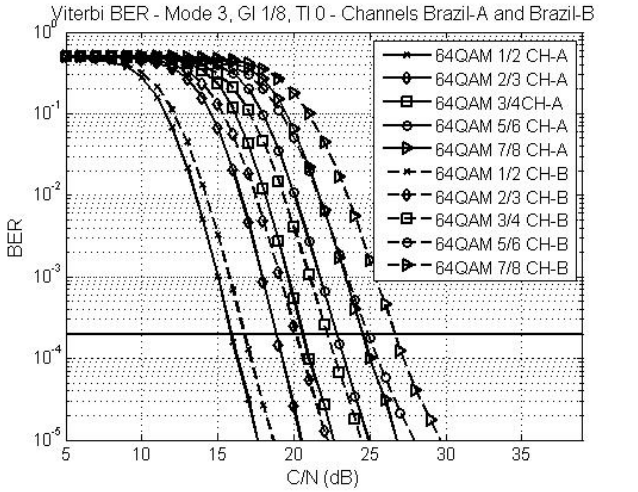


Figure 5.44: Model receiver performance after Viterbi Decoder, over Brazil-A and Brazil-B channels, for Mode 3, GI 1/8, Time Interleave 0, 64 QAM.

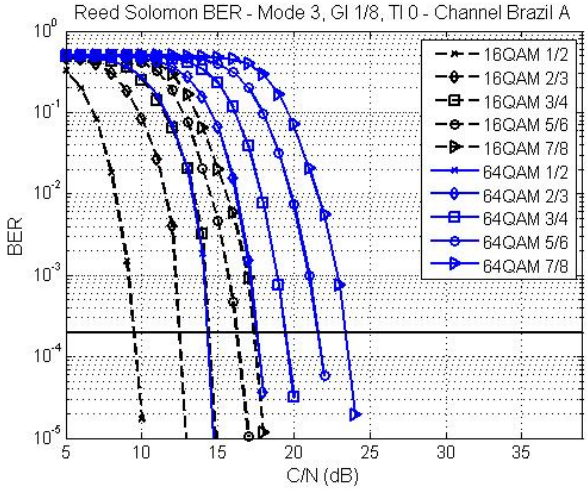


Figure 5.45: Model receiver performance after RS Decoder, over Brazil-A channel, for Mode 3, GI 1/8, Time Interleave 0, 64-QAM and 16-QAM.

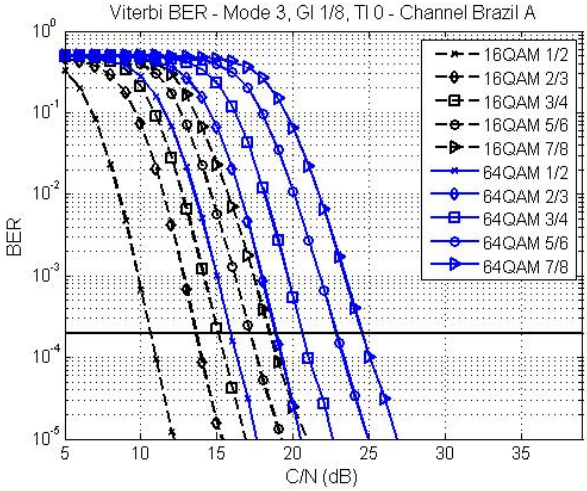


Figure 5.46: Model receiver performance after Viterbi Decoder, over Brazil-A channel, for Mode 3, GI 1/8, Time Interleave 0, 64-QAM and 16-QAM.

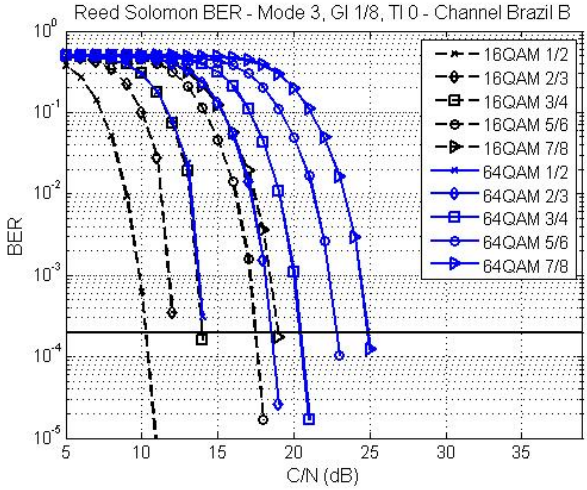


Figure 5.47: Model receiver performance after RS Decoder, over Brazil-B channel, for Mode 3, GI 1/8, Time Interleave 0, 64-QAM and 16-QAM.

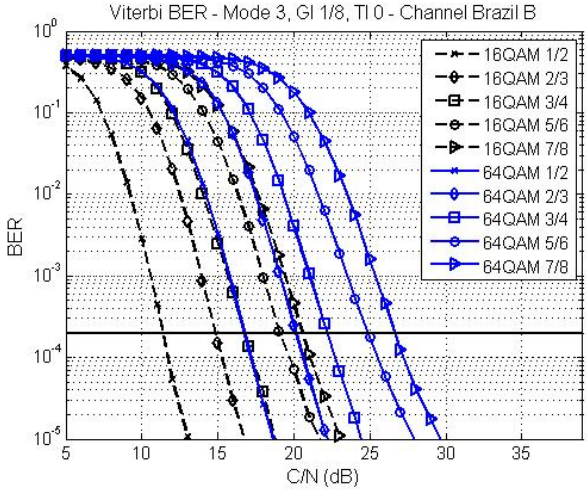


Figure 5.48: Model receiver performance after Viterbi Decoder, over Brazil-B Channel, for Mode 3, GI 1/8, Time Interleave 0, 64-QAM and 16-QAM.

5.4.2 Performance Under Impulsive Noise

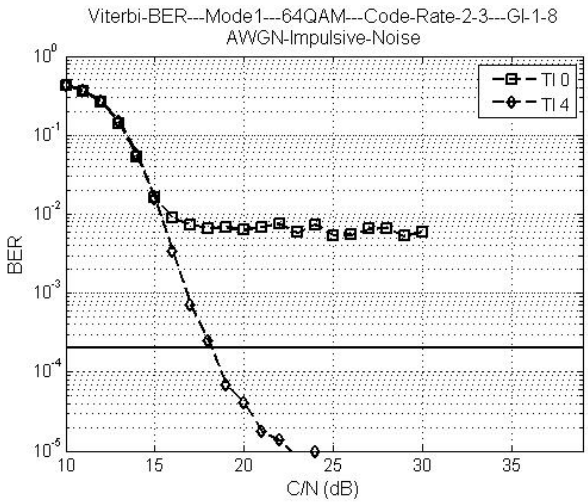


Figure 5.49: Model receiver performance, after RS Decoder, under Impulsive Noise, for Mode 1, GI 1/8, Time Interleaver 0 and 4, 64-QAM, Code Rate 2/3.

5.4.3 ADC Quantization Effects in the Performance

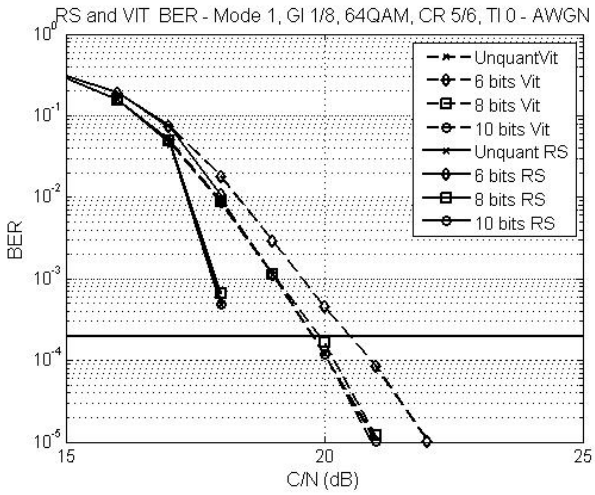


Figure 5.50: ADC quantization effects in the BER after RS and Viterbi Decoders, under AWGN, for Mode 1, GI 1/8, Time Interleave 0, 64-QAM, Code Rate 5/6.

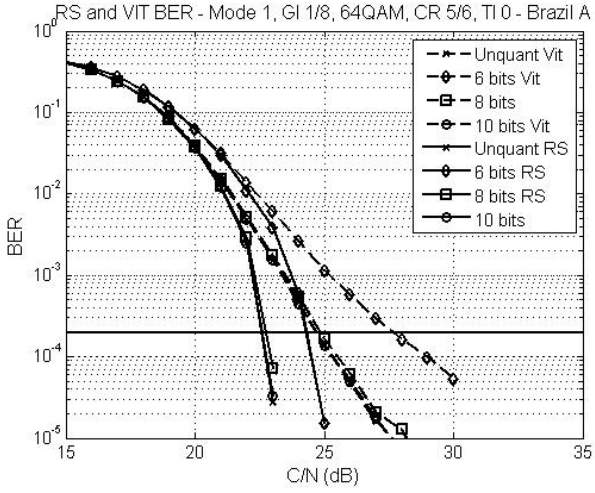


Figure 5.51: ADC quantization effects in the BER after RS and Viterbi Decoders, over Brazil-A channel, for Mode 1, GI 1/8, Time Interleave 0, 64-QAM, Code Rate 5/6.

5.4.4 SCO, RCFO and Fine Boundary Convergence

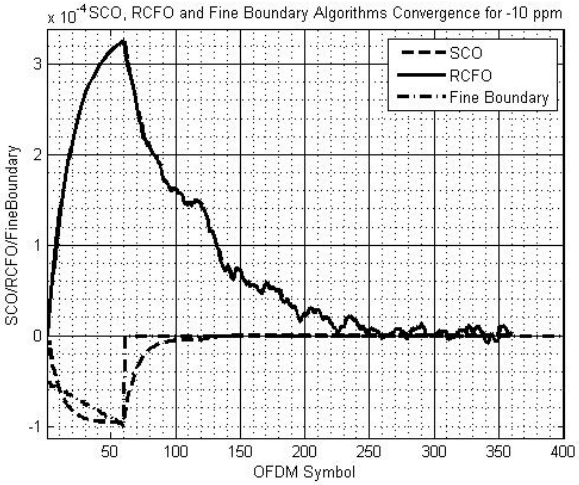


Figure 5.52: SCO, RCFO, fine boundary searcher, under AWGN with C/N = 25 dB.

5.5 Conclusions

In this chapter, the receiver architecture and selected algorithms for the implementation, in ASIC, of a receiver compliant with the ISDB-T and ISDB-Tb, were presented. The aim of the study presented in this chapter was to select and evaluate the algorithms candidates to be implemented in hardware (specially in silicon), which would make possible the receiver to achieve an acceptable BER performance in the presence of the several real world impairments an OFDM-based receiver is subject to. Furthermore, architectures were proposed for the implementation of those algorithms in hardware. An efficient way to implement the mathematical operations used in Digital Signal Processing, such as divisions and trigonometric functions, in hardware is using the CORDIC algorithm. For that reason, the CORDIC algorithm was adopted in several of the implementation architecture proposed in this chapter.

The algorithms were evaluated using Matlab and Octave and after initial exploration, which aimed to validate the selected algorithms in terms of performance, they were integrated to form the complete receiver model. Several simulations results were presented to show that the selected algorithms are suitable for implementation in terms of performance. Overall BER vs. C/N simulations, for AWGN and wireless channel, were also performed to check the convergence and performance of the synchronization and equalization algorithms when working together.

An important task, evaluated during the selection of the algorithms and definition of their architecture, was to check the performance with a limited number of bits. In the case of the ISDB-T receiver proposed in this thesis, the number of bits used in certain blocks pay a significant role in the final area of the ASIC. So, an adequate tradeoff between bit length and area should be observed to make the final ASIC commercially viable. This is specially taken into account in those ASIC targeting consumer electronics. Since the beginning of the ISDB-T receiver project it was clear that the main area of the IC would be occupied by the Time Deinterleaver memory, that in other hand would be limited by the number soft bits of the Soft Demapper with CSI. The adequate definition of the number of bits of that block was an important task to define the area the implemented IC would occupy.

The main contributions presented in this chapter were reported in [16] in the form of an invited presentation and in [18] in the form of a conference paper. Some of the blocks presented in this Chapter have been subjected to patent application. One example of that is the US Patent Application regarding the implementation of the engine for channel estimation and equalization [17]. Other candidate to patent application is the architecture of the remaining carrier frequency offset and sampling clock offset blocks. Furthermore, the final results on the implementation of the algorithms described in this Chapter as well as BER performance of the implemented ASIC (named DTV01) were submitted to a journal paper [19].

In the next chapter , the implementation results of the architecture and algorithms presented in this chapter will be shown. The prove of the design in silicon is a must, to show that the algorithms and architecture leads to acceptable receiver performance and ASIC area/cost.

Chapter 6

ISDB-T Receiver: VLSI Implementation, FPGA Prototype and Test Results

6.1 Introduction

In the two previous chapters, the methodology adopted in the ISDB-T receiver project as well as the receiver architecture and its main algorithms were presented in detail. In this Chapter, the main implementation results, the FPGA prototyping and IC test environments are going to be presented.

The major task when designing an IC for consumer electronics is to reduce the production costs of the final die. Usually, this is achieved by reducing the area of the IC or increasing the number of wafers to be produced, but in general the number of wafers to be produced only increases if the IC is a success. Nevertheless, it is not so simple to reduce the area of an IC, because several tradeoffs (associated to constraints like timing, power, BER performance, throughput, level of pipeline, among others), shall be evaluated. However, it is worth to highlight that the area of an IC not necessarily shall shrink only in order to reduce the production costs of the IC. It can be a requirement of a device (e.g. computer, cell phone, tablet, etc) architecture, that sometimes has limited area for the IC in the printed circuit board (PCB). So, small footprint could be of paramount importance for the success of the IC. Of course, the IC footprint not only depends on die area but also on the packaging type. Two examples of packing containing the ISDB-T receiver die are shown in Sections 6.2.4 and 6.2.5. The first is low budget ceramic package used to test the implemented

IC. The second is BGA package for a Multi-Chip-Module (MCM), which contains the ISDB-T receiver die, a tuner and a PCI Express transceiver plus Analog TV demodulator. It is worth to mention at this point that to attract investors for the IC and its evolution, as well as buyers, a silicon proven is mandatory, independently if the IC is part of a research or a commercial IC.

In Section 6.2.1 the estimated power dissipation of the ISDB-T receiver IC is presented. Power dissipation is the major constraint for nowadays omnipresent portable devices. Nevertheless, it is not a big issue if the implemented IC does not target a portable device. Despite power dissipation was taken into account (and the estimated values are quite low) when designing the ISDB-T receiver and mapping it to ASIC, that was not the major constraint of the project.

The major constraint during the ISDB-T IC receiver design and implementation was the area, because this could reduce the price of the final die and by consequence it could increase the number of wafers to be produced, decreasing the cost per die even more. On the other hand the major metric for any wireless receiver is the BER vs. SNR target performance, for AWGN and wireless channels, which is limited by the performance of the selected algorithms. For those reasons, the die area and the performance of the implemented ISDB-T receiver, for AWGN and wireless channels, are presented in Sections 6.2 and 6.5 respectively.

During the receiver architecture and algorithm exploration phase, it was realized that the block who would dominate the area in the IC was the memory devoted to the time deinterleaver, whose size is defined by the deinterleaver depth and the number of bits of the Soft Demapper with CSI, that was an essential block to make the receiver performs in a acceptable manner under multipath fading scenarios. This is shown in Section 6.2.3.

The main goal of this chapter is to present the BER performance of the ISDB-T receiver implemented in ASIC (which is shown in Section 6.5) in order to prove that the chosen architecture and algorithms are suitable for the target application.

In short, in this Chapter I shortly describe the environment used to test the FPGA prototype and the ISDB-T Receiver ASIC implementation . In addition I present some test results for the ASIC implementation named DTV01. Furthermore I show two preliminary products originated from the DTV01: 1) a Mini-PCI demo board containing a DTV01, a Tuner and a PCI bridge Integrated Circuits; 2) a Multi-Chip-Module(MCM), which is made of a DTV01, a Tuner and a PCI Bridge dies.

In order to validate the proposed receiver architecture prototyped in FPGA and implemented in an ASIC, several laboratory setups were used. Also, several test boards were implemented and a software tool compatible with Linux and Windows was developed. With this tool, it was possible to configure the DTV01 prototyped in FPGA and the ASIC, as well as access the internal register bank of the FPGA prototype and ASIC, to access the receiver parameters to support debugging and

validate the receiver functionalities and performance.

6.2 VLSI Implementation in 65 nm CMOS

Figure 6.1 presents the DTV01 chip encapsulated with a low-cost ceramic package, and connected to the test board via a connector ZeroInsertion Force (ZIF). In zoom, it is possible to identify both the ADC and the analog portion of the USB, on the left and right sides of the DTV01 respectively. Figure 6.2 shows the resulting Floorplan obtained using the Calibre tool. In addition to the analog portion of the USB and the ADC, the Figure shows the seaofgates with all the DTV01 Digital Logic, and the Memory Blocks (black blocks). As can be seen, the memory blocks dominate the area of the DTV01.

6.2.1 Physical Synthesis

Results reported by the Synthesis tool (Logic and Physical) using some limited Place&Route (P&R) capabilities are reported in this section.

The information in Table 8.2 was reported by the CAD/EDA Automatic Synthesis tool at the end of the latest timing optimization steps, using some features of Physical Synthesis. In this mode, the Synthesis (front-end) and P&R (back-end) software interact and P&R tool is invoked to execute a quick P&R procedure, with the objective of generating a synthesized netlist, intended to be better co-related with the final silicon implementation, in terms of timing. Both the automatic floorplan generated by the tool and the manual floorplan were deployed.

Table 6.1: Final Synthesis Report, Generated by the EDA/CAD Logic Synthesis Tool.

	Instances Number (Devices Count)	Area (μm^2)	Internal Power (nW)	Leakage Power (nW)
Auto-Floorplan	483821	13066916.663	84169037.008	1733173.315
Manual-Floorplan	533172	13218313.143	107262766.515	1747496.56

6.2.2 Physical Design

The result of the Physical Design flow is presented by the layout shown in Figure 6.2, where a die with dimensions of 4.3x4.4 mm was created. All the main functionalities (Data Processing and Signal Processing) were placed in the core region of the die, spread and distributed on the area of sea-of-gates, the placement criteria followed a

timing-driven process by the EDA/CAD P&R tool, the floorplan below shown in that Figure aimed at decreasing congestion and generating a routable design.

In Figure 6.2, the core region in green & blue represents the location of the sea-of-gates, where all the synthesized logic cells are placed (chip_top is the top hierarchy cell), routed and afterward optimized by the Physical Design process. The blocks surrounding the sea-of-gates (polygons represented by white boundaries and black color background) correspond to the hard IP macros: memory blocks (ROMs and RAMs), ADC hard IP (analog and mixed-signal), located on the south-west corner of the sea-of-gates, and surrounded by a guard-ring (in blue). The USB interface hard IP is located on the south-east corner (light-blue), near the IO pad-ring, in both cases for a more effective interfacing and connectivity with the IO pads.

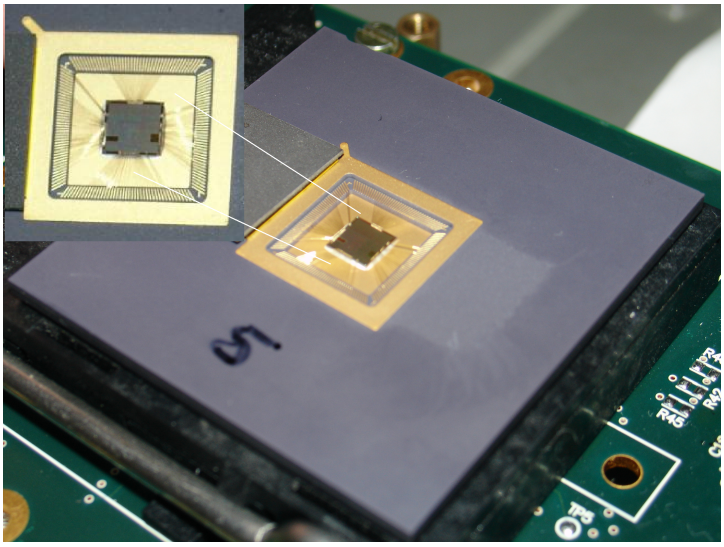


Figure 6.1: DTV01 encapsulated with low cost ceramic package and die zoom.

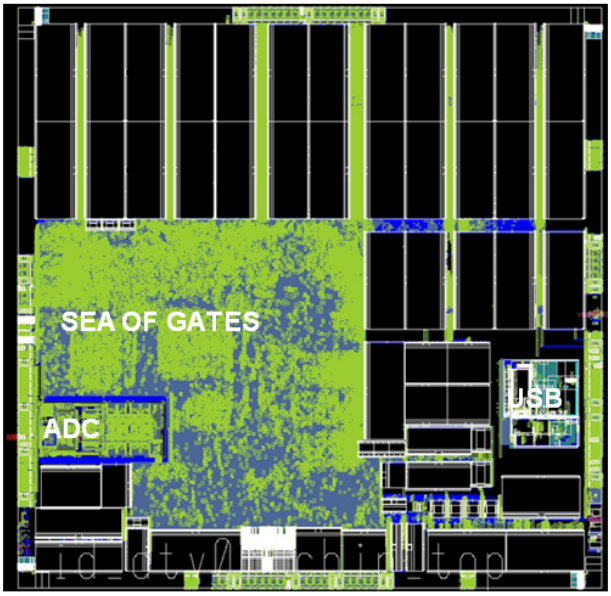


Figure 6.2: Final layout of ISDB-T receiver(DTV01), from Calibre tool (chip_top): main logic, ADC and USB PHY areas are identified. The black boxes are memories (RAMs and ROMs).

6.2.3 Memory Area

Table 6.2 presents the DTV01 memory usage. The inspection of the results shows in that Table demonstrate that memories occupy more than 50% of the area, and the Deinterleaver is the receiver block with the highest memory usage. It occupies about 38 % of the area.

Table 6.2: Memory Usage and Occupied Area.

ENTITY	NUMBER OF BITS	AREA (m^2)
Total	13331072	9.158332031
Deinterleaver	9830400	6.59
FFT	1094400	0.68
Others	2406272	1.89

6.2.4 DTV01 Demo Board

In order to evaluate and characterize the implemented ASIC, the demo board presented in Figure 6.3 was developed. The demo board is connected to the FPGA Development Kit STRATIX IV GX, from Altera, via an HSMC connector. The FPGA is used to write and read the receiver parameters via Register Bank, using a program running into the embedded processor NIOS II. The ASI output is used to transport the MPEG-TS packets to an external MPEG Decoder, that can be a dedicated hardware or an AT40USB/AT40XR2USB from Alitronika, which is used to record the data and send it to a PC where the TS stream can be decoded by a program such as VLC and Elecard. In Section 6.4, Figure 6.10 presents a block diagram with the details of one of the test environments used in the project.

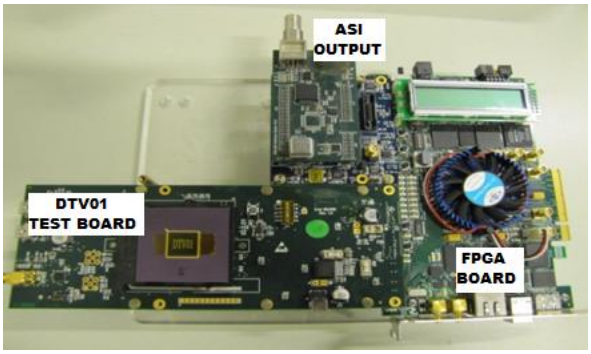


Figure 6.3: DTV01 demo board with MPEG-TS over ASI output, with Stratix IV kit as control board.

6.2.5 Mini PCI Based DTV01 Demo Board and Multi-Chip-Module(MCM)

One of the goals of the IC Brazil Program is to create cooperation environment between companies and the DHs in order to create products with the ICs projected by the DHs. In this way I present two examples of products developed at Eldorado Research Institute, using the DTV01, that were ordered by a client company. The first is a demo board with mini PCI connection capability, shown in Figure 6.4, in a zoom view, and in Figure 6.5, connected to a mini PCI slot of a PC. That demo board was used by the client company to verify the performance and functionalities of the DTV01, using two software tools. The first is used to control the IC (the tool screen can be seen in Figure 6.5) and the second is a Player that decodes and presents the video in a Windows based PC (there is also a version of these tools for Linux). Due to success of the first product, the partner company ordered an MCM to Eldorado

Research Institute. The main goals of the MCM, which is shown in Figure 6.6, are to reduce the Bill-Of-Material (BOM) of the mini PCI board, and to fit it in the reduced area available in the new mother board, projected by the customer. The MCM was first tested in a demo board, shown in Figure 6.7, and after that step the MCM was integrated in the new mother board.

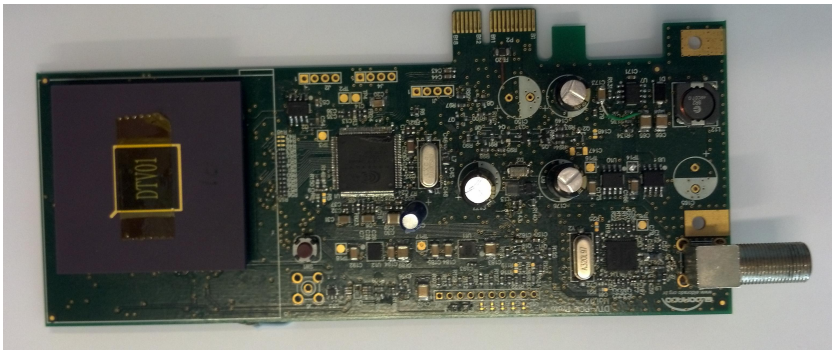


Figure 6.4: DTV01 mini PCI demo board

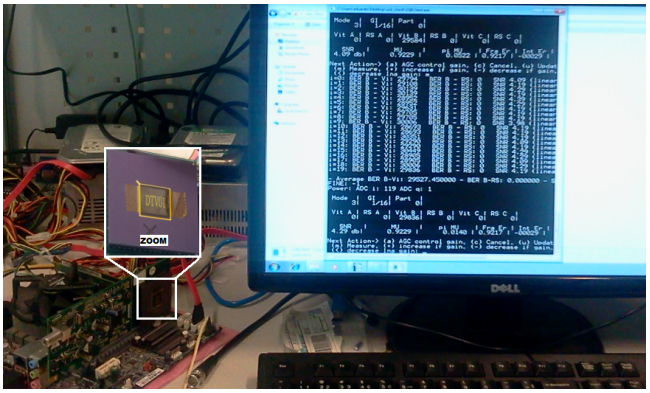


Figure 6.5: DTV01 mini PCI demo board connected to a PC and a snapshot of the control tool screen.

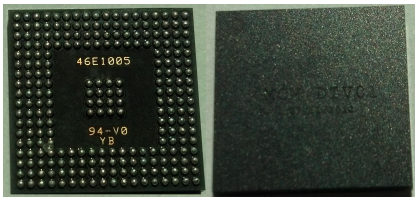


Figure 6.6: DTV01 - MCM bottom and top views.

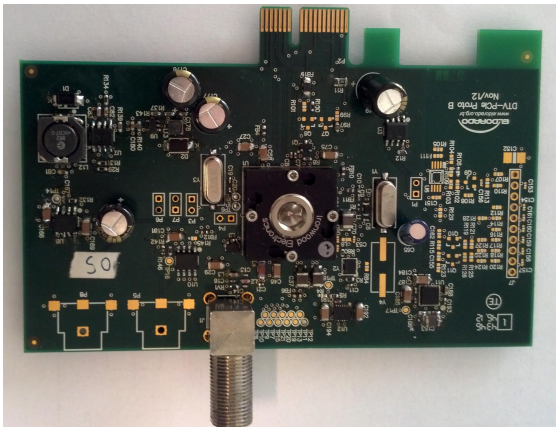


Figure 6.7: DTV01 - MCM test board

6.3 FPGA Prototype

The cost of MPW runs and EDA/CAD tools for IC design, sometimes make the access to them prohibitive to small design houses and startups in early stages, in order to make a first prototype in ASIC. In this way FPGA prototyping is a very attractive technology for concept proving, before a larger investment is done to move the design into ASIC.

In this thesis, I will not discuss the merit of using FPGA as a final platform for certain applications. But rather, due to the reasons mentioned in the previous paragraph, I state that it was the only choice to follow, to prove that the receiver architecture was working properly in hardware, and would be safe to move the design to an ASIC.

In this Section I shortly describe the FPGA prototyping phase as well as the used

resources to prove the receiver architecture concept. Some steps followed in this phase were already described in Chapter 4, so the focus here lies in describing the FPGA prototyping platform, peripherals boards and tools.

As described in Chapter 4, one of the phases after HDL design is the physical validation of the hardware in FPGA.Each one of the receiver blocks implemented in VHDL were tested using one or more of the following FPGA prototyping development kits: Nios II Development Kit Stratix II EPS260, DSP Development Kit Stratix II EP2S180 and Stratix IV GX. In the initial phase of the project, the EP2S60 was used, and as the project evolved, the design was migrated to EP2S180 where the entire receiver was first integrated. Finally, we used the Stratix IV GX. It was not possible to validate all Time Deinterleaver lengths in the FPGA boards, due to the necessary size of memory. So, the final version of the Time Deinterleaver was simulated using the memory controller and an ARM models of the ASIC memory.

Figures 6.8 and 6.9 show the two FPGA integration platforms used in this project. The EPS2S60 is not shown here because it was used only to perform physical tests of the blocks. The integration platforms consist of a tuner provided by a partner, an FPGA design kit and an output board. In the first kit (Figure 6.8) the output board is an ASI daughter card. In the second platform the USB was already integrated at the digital receiver, so an external analog USB PHY was used to control and configure the receiver and to capture information such as control parameters and TS packets. It is also possible to transfer the samples, captured by the ADC or from certain points of the receiver, to the PC that is controlling the USB. This data can be used as input for the refinements of the algorithms/blocks in the Matlab or as input for the HDL simulator, for instance. The tools for controlling the tuner were provided by the tuner providers. The access to the receiver in the first platform was done via *in system memory content editor*. Some of the debug information such as BER, SCO and IFOA could be accessed using a external Display, that is not shown in Figure 6.8. In the second board the information from the register bank could be accessed using the USB, so a SW tool was built by Idea! in order to do so. The debug information could also be accessed using the Display, that is shown in Figure 6.9.

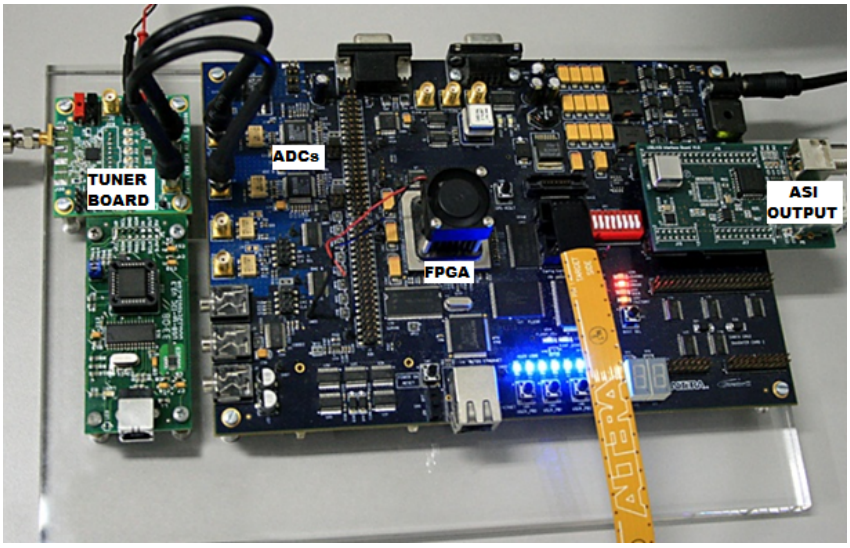


Figure 6.8: FPGA integration platform 1 : EPS2S180, tuner number 1.

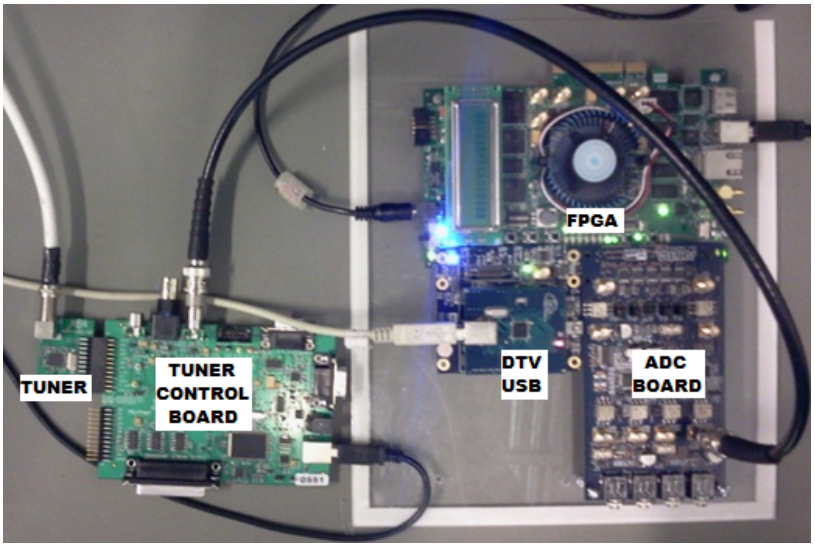


Figure 6.9: FPGA integration platform 2: Stratix IV GX - tuner number 2.

6.4 Test Environment

In order to validate the performance of the implemented receiver in FPGA and ASIC, in real world scenarios, two laboratory environments were assembled. One at Idea! Electronic Systems and the second at Eldorado Research Institute. Some equipments were shared (e.g. Channel Emulator), sometimes they were used at Idea! and sometimes at Eldorado. The overall block diagram of the Laboratory at Idea! is shown in Figure 6.10 and part of the equipments can be seen in Figure 6.11. Most of the time, the laboratory at Idea! was used to test and debug the FPGA prototypes.

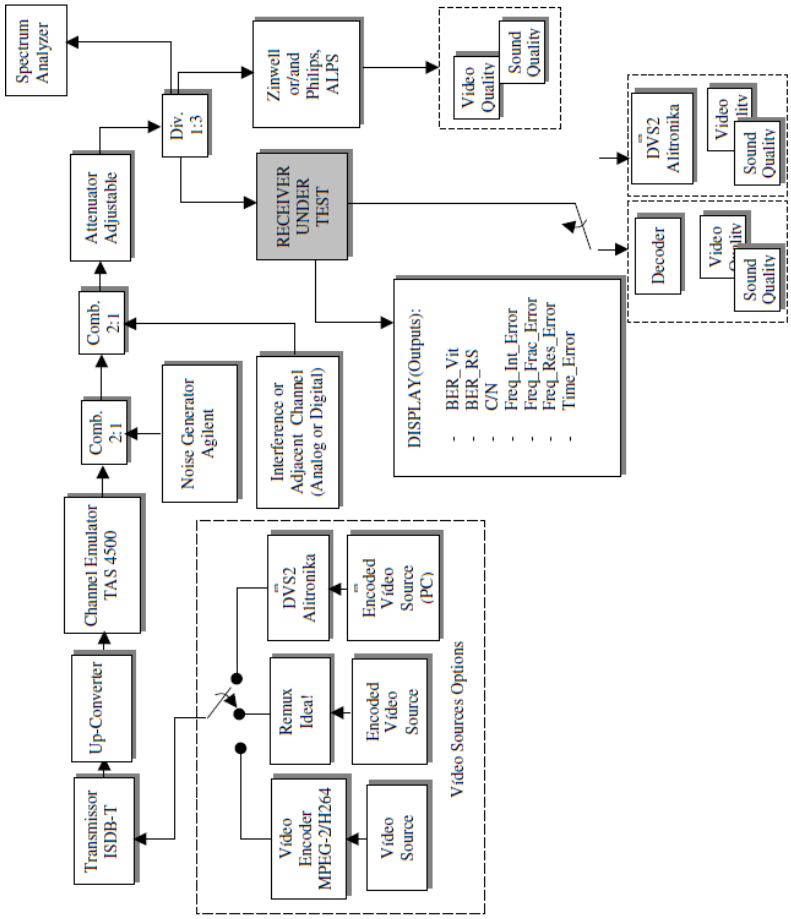


Figure 6.10: Laboratory setup used to test the FPGA prototype at Idea! Electronic Systems.

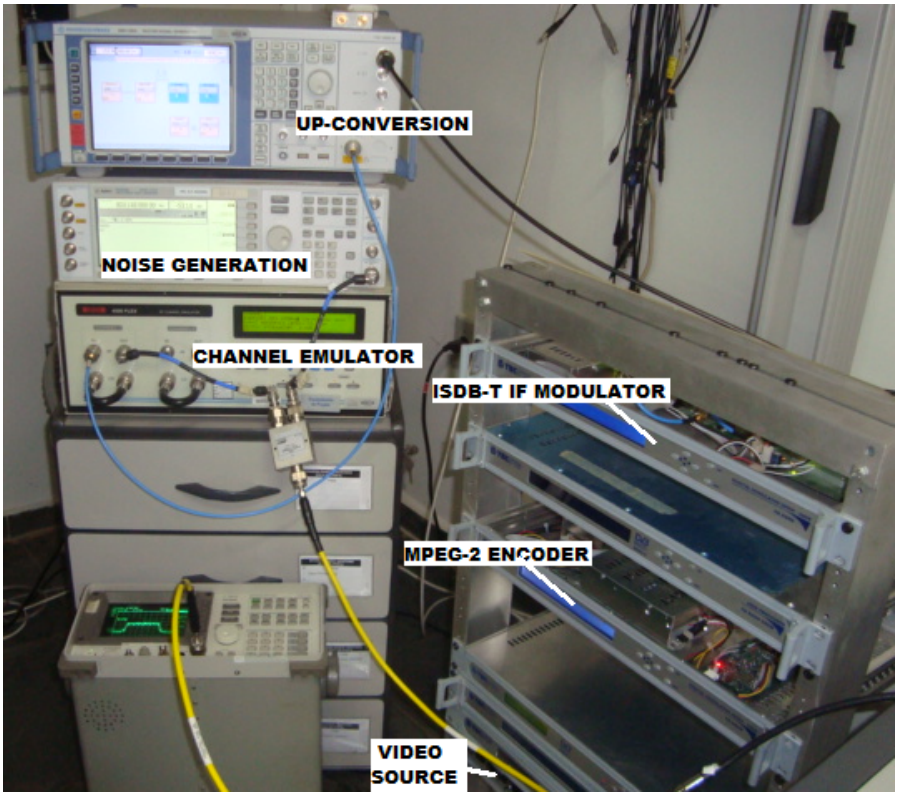


Figure 6.11: Laboratory set-up used at Idea! Electronic Systems to perform tests during the receiver integration and for the final FPGA prototype.

Figure 6.12 shows the main equipments used in the Laboratorylocated at Eldorado.This laboratory was mainly used to test the DTV01 (ASIC bringup, adjust of parameters and performance evaluation) and perform demonstrations. Some test results using this Laboratory are going to be shown in Section 6.5. All equipments in that laboratory are mandatory. Nevertheless, the main equipment is the 6-taps Channel Emulator from Spirent (former name), showed in zoom in Figure 6.13. With this equipment it is possible to reproduce most of the wireless channel characteristics without the need of making costly and logistic-complex drive tests.

It is worth mentioning that in both laboratories, commercial DTV signal from a number of DTV broadcasters placed in the region of Campinas-SP could be received and used to test the implemented receivers (FPGA and ASIC) at any time.

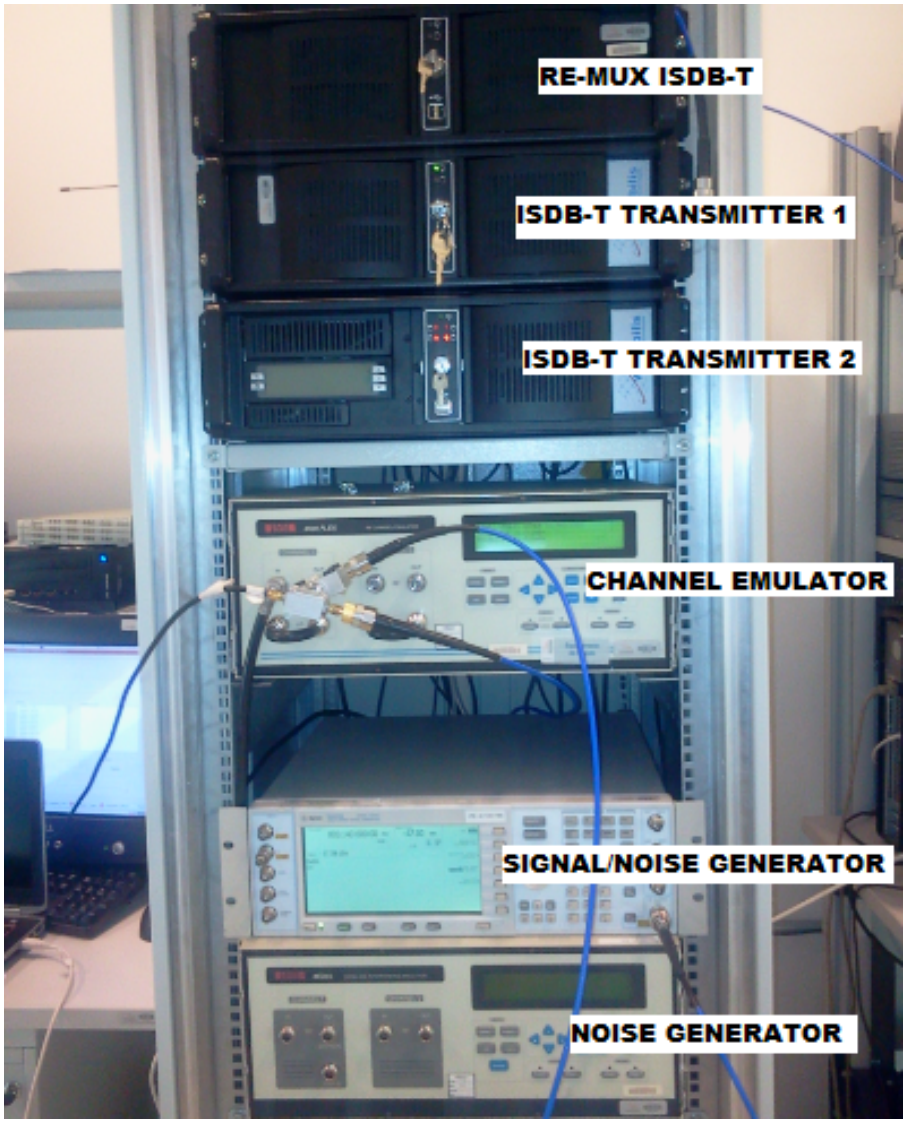


Figure 6.12: Some laboratory equipments used to test DTV01 and FPGA prototype.

Figure 6.13: Channel emulator generating a snapshot of ETSI static channel.

6.5 Experimental Results

Herein I present some results of the tests performed using the DTV01, to prove the correct functioning of the ASIC. The tests were done at Eldorado Research Institute Digital Television Laboratory. Performance evaluation over Brazil-A and Brazil-B PDPs and AWGN were performed. The results for AWGN were compared to the target values presented on the ISDB-T standard. The performance over Channel-A and Channel-B were compared to simulations results. The results of those tests are presented in Tables 6.4 and 6.3. I also present results for the ICFO and the C/N estimators, in Table 6.5 and Figure 6.14 respectively.

As can be seen, in Table 6.3, the implementation losses, regarding the target CN defined in the standard [1] for AWGN, is less or equal to 1 dB. Table 6.4 shows that, in most cases of the tests over Brazil-A and Brazil-B channels the implementation losses regarding the simulated values with perfect synchronization, which are shown in Figures 5.48 to 5.41, is smaller than 1.2 dB. Just in the case of Brazil-B PDP with Code-Rate 3/4 the error is about 2.5 dB at Quasi-Error-Free point. The SNR estimator(see Figure 6.14) presents a good approximation for 7.5 to 26 dB and the ICFO works as expected as shown in Table 6.5.

Table 6.3: Performance of the 65nm ASIC ISDB-T Receiver Layer B, for 2x10-4 at Viterbi Decoder Output: AWGN, Mode 3, GI 1/8, 2 Layers - Layer A 1 SEG, Layer B 12 Segments.

MODULATION, CODE RATE	64-QAM, CR 1/2	64-QAM, CR 2/3	64QAM, CR 3/4
C/N TARGET (dB)	16.5	18.7	20.1
C/N MEASURED (dB)	15.6	19.0	20.2
IMPLEMENTATION ERROR (dB)	-0.9	+ 0.3	+0.1
MODULATION, CODE RATE	16-QAM , CR 1/2	16-QAM , CR 2/3	16-QAM,CR 3/4
C/N TARGET (dB)	11.5	13.5	14.1
C/N MEASURED (dB)	10.3	12.8	13.9
IMPLEMENTATION ERROR (dB)	-1	-0.7	-0.2

Table 6.4: Performance of the 65 nm ASIC ISDB-T Receiver Layer B, for 2x10-4 (Error Free at Reed Solomon Decoder Output) and 1x10-5 at Viterbi Decoder Output: Measured over Channel Brazil-A and Brazil-B, Using the Mode 3, GI 1/8, 2 Layers - Layer A =¿1 SEG, Layer B =¿ 12 Segments, TI=0.

MODULATION, CODE RATE	64-QAM , CR 1/2		64-QAM , CR 2/3		64QAM, CR 3/4	
CHANNEL	BRAZIL A		BRAZIL A		BRAZIL A	
BER - VITERBI OUT	2x10-4	2x10-5	2x10-4	2x10-5	2x10-4	2x10-5
C/N TARGET (dB)	16	17.5	18.5	20.8	20.8	22.6
C/N MEASURED (dB)	15.7	17.6	19.3	21.3	21.3	23.5
IMPLEMENTATION ERROR (dB)	-0.3	+0.1	+0.8	+0.5	+0.5	+0.9
MODULATION, CODE RATE	64-QAM , CR 1/2		64-QAM , CR 2/3		64QAM, CR 3/4	
CHANNEL	BRAZIL B		BRAZIL B		BRAZIL B	
BER - VITERBI OUT	2x10-4	2x10-5	2x10-4	2x10-5	2x10-4	2x10-5
C/N TARGET (dB)	16.9	18.5	21	22.3	22.8	24.6
C/N MEASURED (dB)	17.4	19.1	20.9	23.4	24	27.1
IMPLEMENTATION ERROR (dB)	+0.5	+0.6	-0.1	-1.1	+1.2	+ 2.5

Table 6.5: Integer Frequency Error Measure: Mode1

Frequency (MHz)	Measured Integer Frequency Error
213.26	-127
213.259	-126
213.258	-125
213.257	-124
213.23	-97
213.225	-92
213.22	-87
213.21	-77
213.2	-67
213.19	-56
213.18	-46
213.17	-36
213.16	-26
213.155	-21
213.15	-16
213.149	-15
213.148	-14
213.147	-13
213.146	-12
213.145	-11
213.144	-10
213.143	-9
213.142	-8
213.141	-7
213.14	-6
213.135	-1
213.13	4
213.12	14
213.11	24
213.1	34
213.09	44
213.08	54
213.07	64
213.06	74
213.05	84
213.04	94
213.011	124
213.01	125
213.009	126
213.008	127

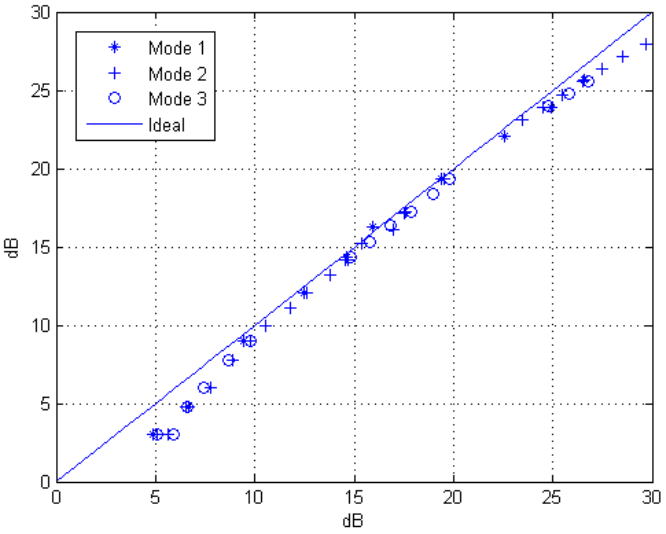


Figure 6.14: SNR measured × ideal.

6.6 Conclusions

As it was shown along this chapter and within the two previous ones, the design of a digital IF/baseband wireless receiver targeting ASIC has several steps. In general, despite obvious importance of the receiver modeling and algorithm definition, if any error occurs in between the algorithms study phase and the front-end design (in HDL) phase, the costs to fix those errors are not so high, when compared to those related to ASIC design flow, specially regarding the EDA tools license costs.

When the IC design follows to the ASIC design flow, the rent of the tools license dominates the costs of the project and usually the prices make it prohibitive this phase to take too long. So, sometimes in order to reduce the time expended in this phase, the gate level simulations are not so extensive as they should be, and the approach adopted by some ASIC designers and verification engineers is to reduce the coverage of the gate level verification (performing long gate level simulations only for the critical states - such as the initial receiver synchronization - or critical blocks) as well as do not correct some DRC violations, that are assumed to be not so critical. This forces the checking of the effects of those missing steps to be done after a MPW-based silicon implementation. This is done because sometimes, the cost of a MPW fabrication is lower than the cost of the EDA tools license for a long period of gate level simulation and debugging. Another important reason to prove an IP in silicon, via MPW, is

because, sometimes, the timing models do not come up with a realistic behavior of the gates, e.g. due to the use of not very accurate timing constraints during/after Synthesis, Place or Route steps or due to limited simulation time (specially at gate level). The timing models are used as constraint for certain tradeoffs studies (e.g. timing vs. area). The lack of maturity of a foundry process is also a known reason for performing silicon proves of IPs, through MPW. In short, the prove of an IP in silicon not only proves the IC was designed properly at front-end and algorithm levels, but also proves the maturity of the adopted ASIC design flow, the maturity of the designers[1] and the maturity of the foundry process as well.

In this chapter, the results on the implementation in silicon of the selected algorithms and proposed architecture, presented in Chapter 5, were introduced. As in the case of the system level performance evaluation presented in the previous chapter, the receiver implemented in ASIC also achieved acceptable performance and presented low deviation from the expected values for simulations over AWGN and Brazil-A and Brazil-B channels. Although some work shall be done in order to improve receiver performance for SFN-like channels, it can be concluded that the architecture proposed works properly over AWGN and wireless channels with exponentially-decaying power delay profiles, and can handle all impairments an OFDM receiver, which targets real world applications, shall overcome.

The main contributions presented in this chapter are the presentation of the results on the implementation of the algorithms described in Chapter5, which implementation in ASIC were partially reported in [17] and submitted to a journal paper [19]. Some of the architectures used to implemented the algorithms described in the previous chapter are going to be submitted for patent application at USPTO, as for instance [17], which is related to the channel equalizer.

[1] Experienced designers are able to skip certain steps or performing it with some relaxation and even so the final design will work properly.

Chapter 7

The Proposed DVB-S2 Receiver Architecture and Algorithms

7.1 Introduction

The DVB-S2 receiver project foundations were settled during the project of the ISDB-T receiver presented in Chapters 4 5 and 6. A direct consequence is that the methodology adopted for this receiver is the same used during the ISDB-T receiver project, with some minor differences that will be presented later in Chapter 8. It is worth recalling that during the ISDB-T receiver project it was realized - due to continental dimensions of Brazil - the potential for a hybrid IC, capable of receiving open HDTV over satellite and terrestrial. Nevertheless, despite the first goal of the DVB-S2 receiver presented in this thesis is the reception of open high definition satellite TV, it can be used in Set-Top-Boxes for payed TV distributed via satellite or high-end devoted receivers, like those used for news gathering or signal distribution for digital terrestrial transmitters placed in remote locations. Furthermore, due to the flexibility of DVB-S2 standard parameters (e.g variable bandwidth, different code rates, two length of frames, different flavors of modulation, etc) the standard can be used for other applications. It only depends on the creativity of the final user and the flexibility of the implemented transmitted and receiver. One simple example of that is to use the receiver in remote locations for real-time railway and highway signaling. Other example is the use of DVB-S2 receiver to drive actuators in energy distribution networks (such as Smart Grids). Along with a transmitter the DVB-S2 receiver can be used in remote wireless sensor/actuator networks.

The three previous Chapters of this thesis were devoted to the presentation of the ISDB-T receiver implementation methodology, its algorithms and architecture, as well as the implementation results. This chapter is devoted to the presentation of a proposed architecture for a DVB-S2 Receiver, as well as selected algorithms, to be implemented in Hardware, to overcome the real-world impairments that a satellite receiver is subject to.

This chapter starts with the introduction of two DVB-S2 receiver architectures in Section 7.2. The first is called full receiver architecture and contains all blocks the final DVB-S2 receiver shall contain at the end of the project, with exception of baud rate estimation block that is missing in the block diagram. The second is a fully functional[1] partial architecture, named current implemented architecture, that was set to prove the feasibility of the DVB-S2 receiver as well as to explore algorithms and attract investors for the project. Following the presentation of the two architectures, the chosen algorithms algorithms for the current and final architectures are exposed in detail in Section 7.3. In addition a platform created to evaluate the FEC decoding subsystem is introduced.

7.2 Architecture for an IF Digital Receiver for DVB-S2 Signal Reception

The full receiver architecture proposed in this thesis is depicted in Figure 7.1. Nevertheless, some of the chosen blocks are still under study and/or development, or are not integrated in the receiver yet. Examples of those blocks are Adaptive Equalizer and extended Soft-Demapper. The Adaptive Equalizer needs to be physically verified in FPGA and integrated in the receiver. The extended Soft-Demapper shall supports all DVB-S2 modulations (but at this moment only works for QPSK, 8-PSK and 16-APSK) also needs to be physically verified and integrated in the receiver. The Soft-Demapper, currently implemented in hardware, only supports QPSK and 8PSK. The current implemented DVB-S2 receiver architecture is shown in Figure 7.2. As can be seen by the inspection of Figures 7.1 and 7.2, with exception of the SNR estimator, all blocks depicted in Figure 7.2 are presented in Figure 7.1.

7.2.1 The Current Implemented Architecture

The proposed receiver architecture, presented in Figure 7.1, aims to be be implemented in ASIC. Nevertheless, as part of the adopted methodology, a prototype phase in FPGA is carried out. As in the case of the ISDB-T receiver, I propose the use of CORDIC algorithm for several signal processing operations, within the blocks of the

[1] i.e. it can demodulate a DVB-S2 compliant signal and deliver MPEG-TS packets at its output.

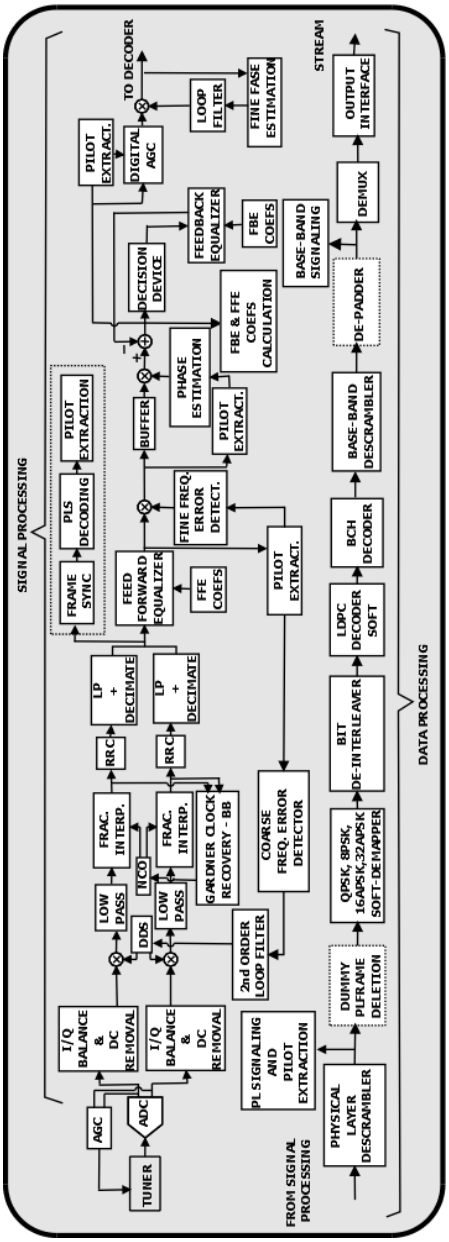


Figure 7.1: The original proposed DVB-S2 receiver architecture.

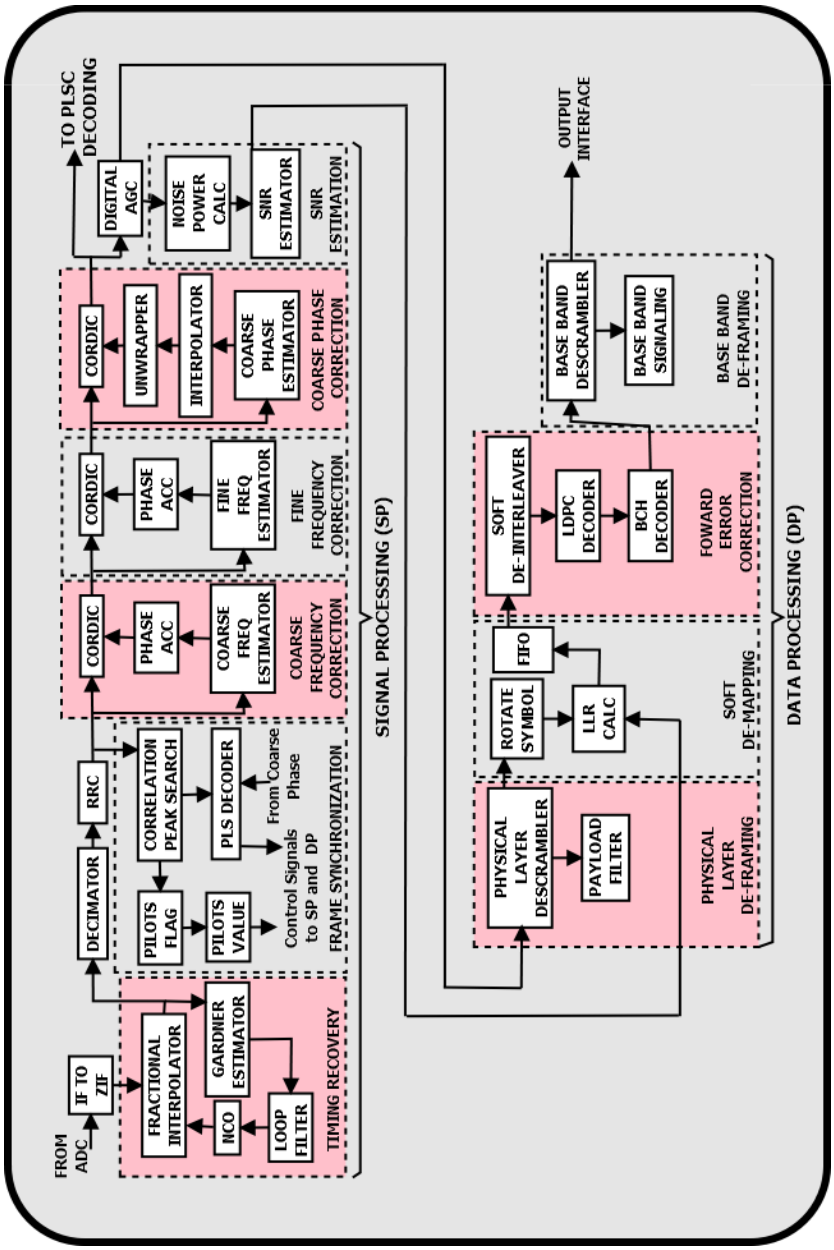


Figure 7.2: The current integrated DVB-S2 receiver architecture.

DVB-S2 receiver architecture. Following, an overview on the current implemented architecture is provided. The differences between the current and the final architecture are also provided, at a glance. Before going into the explanation of the architecture, it is worth mentioning that as in the case of the ISDB-T receiver, the DVB-S2 receiver architecture proposed in this thesis was split in two partitions namely Signal Processing and Data Processing.

7.2.1.1 Signal Processing

The architecture presented in Figure 7.2 supports Low-IF and Zero-IF input signal. Following a 10 bits analog-to-digital conversion, the signal containing all impairments feeds the Signal Processing blocks. In the case of IF reception, down conversion is performed by using a CORDIC-based [87] DDS, followed by low pass filtering. On the other hand if Zero-IF is the case, only low pass filtering is performed in order to reduce adjacent channel interference.

The Timing Recovery loop performs symbol sampling error estimation and correction. The estimator uses Gardner's Algorithm [103] while the correction is obtained by Fractional Interpolation based on Farrow Structure with fixed taps [104]. A 2nd order Loop Filter is used to smooth the error estimation, as the NCO accumulates the error and preserves the relation between the input and output samples of the loop. This is achieved by either discarding or retaining the samples in the Fractional Interpolator. The output samples are then filtered by a Root Raised Cosine (RRC) and decimated to Symbol Rate.

Subsequently, Frame Synchronization is performed using the header autocorrelation and the Peak Search Algorithm [105; 106] to detect the frame boundary and the SOF. The PL header is the first part of the PL Frame and contains information regarding the modulation, code rate and frame type. These parameters, protected by a Reed-Muller (RM) code, along with the SOF flag, are used for the Physical Layer Signaling (PLS), being decoded by a Fast algorithm for Walsh Hadamard Transform (FHT) [107]. Before decoding, they are soft-demapped, descrambled and averaged in a two soft-bit fashion. This process is possible due to the bit repetition established in the DVB-S2 standard [77], allowing the PLS-Decoder to work properly under a lower Signal-to-Noise-Ratio (SNR). The parameters are sent to other units through Pilot Flags, for frequency, phase and amplitude correction.

Thereafter, Frequency Correction is performed in two steps, i.e. Coarse and Fine. In order to estimate the Coarse Frequency error, the proposed receiver uses the scheme presented in [108], which is based on autocorrelation and the algorithm derived by Kay in [109]. The Fine Frequency estimation is performed using the Maximum Likelihood algorithm proposed in [106]. Both estimations are done over a configurable number of frames and provide a single pilot-based estimation. The frequency estimations are applied to their respective Phase Accumulators and CORDIC de-rotator [87].

Afterward, the baseband constellation only presents phase and amplitude mismatches, which are respectively corrected using Coarse Phase Corrector and Automatic Gain Control units. The Coarse Phase Correction uses correlation and pilots to estimate the error as shown in [108]. The phase estimation is evaluated by an unwrapper to eliminate phase ambiguity, then interpolation between the phase difference of two neighbor pilots is done and applied to the CORDIC de-rotator.

It is worth noticing that the use of CORDIC algorithm for diverse signal processing operations, other than DDS, reduced the complexity of the DVB-S2 receptor.

7.2.1.2 Data Processing

At a glance, Data Processing blocks convert received Physical Layer (PL) Frames [77] into Transport Stream (TS). For convenience, the first three design units (Physical Layer Descrambler, Payload Filter and Soft-Demapper) are placed in this partition, even working with symbols and not bits.

First of all, the Data Processing partition receives the signal coming from Signal Processing partition in the form of PL Frames. They are descrambled by the Physical Layer Descrambler to recover the original transmitted DVB-S2 sequence of frames, previously scrambled for energy dispersal purposes. The resulting PL Frames have pilots and header symbols removed by the Payload Filter, outputting only the payload data. Following the sequence, an 8-PSK Soft-Demapper converts the symbols into soft-bits, which are equivalent to the Log-likelihood Ratios (LLRs) representing the probabilities of the bits being '0' or '1'. These probabilities are obtained using the estimated SNR, provided by the SNR Estimator block through the power noise calculated between the known pilots and the received signals. The demapped soft-bits are buffered by the Demapper FIFO, whose size is determined by the De-Interleaver throughput.

Next comes the Forward Error Correction (FEC) sub-system, composed by De-interleaver and, LDPC and BCH decoders. Although the De-interleaver is included in the FEC sub-system, it does not correct any errors in the frame, it rearranges the data in the original sequence, previously interleaved in a non-contiguous way, to increase performance in error-correction coding. The structure of the frame provided by the De-interleaver is illustrated in Figure 7.3. The length of each frame segment depends on the adopted code-rate and frame type, as established in [77].

The error correction obtained by the combination of LDPC and BCH decoders: the former executes a "'coarse"'correction, eliminating most of the errors, while the latter is responsible for a fine correction, by switching up to 12 erroneous bits.

The LDPC Decoder implements the Minimum-Sum algorithm, which is a simplification of the Belief-Propagation algorithm based on LLRs [110]. For each of the 21

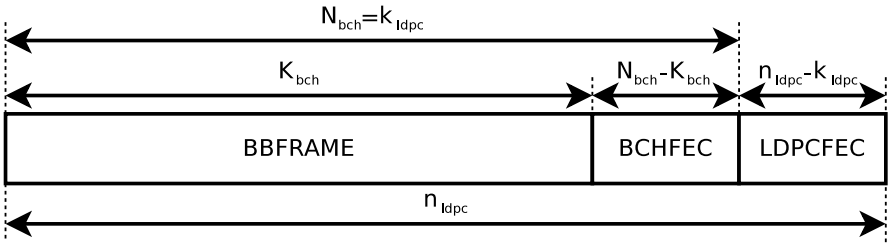


Figure 7.3: DVB-S2 FECFRAME

frame configurations, a different LDPC coding scheme is applied. The architecture of the implemented LDPC decoder is described in [21]. Based on the received soft-bits, the decoder runs a given number of iteration (normally between 30 and 50) and finally delivers to the BCH decoder the payload BBFRAME+BCHFEC and discards the parity bits (LDPCFEC).

Consequently, the BCH-Decoder uses the BCHFEC bits for recovering the original BBFRAME, correcting up to 8, 10 or 12 errors according to the code-rate and frame type. This is accomplished by serially executing three sub-blocks, the syndrome calculator [22], a key equation solver as the Berlekamp-Massey algorithm [111] and a polynomial roots finder like the Chien Search algorithm [112]. Afterward, the BBFRAME is forwarded without any FEC parity bits.

At this point, PL Frames are converted into Baseband (BB) Frames and descrambled again, but this time by the BB-Descrambler, retrieving the original frames scrambled at the transmitter. Next, in the BB Frames the header bits are removed and bits related to the Cyclic Redundancy Check (CRC) are replaced by synchronization bits [77]. This process is performed by the BB-Signaling block and generates the TS, which contains the user packets. The Data Processing flow ends with the BB-Signaling. Nevertheless, in order to feed external MPEG-TS decoding hardware equipped with Asynchronous Serial Interface (ASI), a Bit-to-ASI conversion is done.

7.2.2 The Final Architecture

Due to large time and high costs involving design and redesign of a receiver targeting its implementation in ASIC, the receiver architecture shall be as complete as possible to achieve a fully functional receiver in the case of a first-pass silicon success. Recalling that when designing a receiver targeting silicon implementation, in general, the tradeoff between the silicon area (die area) and receiver complexity (related to the blocks and algorithms to be implemented) is taken into account. In short, in order to achieve better performance (BER vs. SNR), the number of receiver functionalities shall augment. So, the complexity increases, as well as the silicon area. That said,

the aim of the proposed full architecture is to handle the main impairments a satellite receiver (working with such variety of parameters is allowed by the DVB-S2 standard) is subject to, while keeping low implementation complexity.

In this Section I describe shortly the main differences between the current implemented and the final architecture presented in Figures 7.2 and 7.1, respectively. The first main difference is the I&Q and DC level removal block that is present in the final architecture. The presence of these two impairments is dependent on the chosen tuner performance and architecture. So, this functional block can be applied or bypassed depending on the selected tuner. It is worth to recall that high DC level at tuner output causes a reduction in the useful dynamic range of the ADC.

A block that is not explicitly depicted in the final architecture is the SNR estimator, for which estimations are needed to be used by the Soft-Demapper. Another usage of this estimation is to access the SNR the receiver is subject to, for engineering debug purposes or to deliver to the user the quality of the receiver signal.

As explained in Chapter 3, the DVB-S2 standard is very flexible and allows different uses for the satellite link. It has a variety of parameters and a variable bandwidth that are used according to the broadcaster's choice. As the bandwidth of the transmitted signal increases the receiver is more likely to suffer the bad effects of multipath propagation, e.g. caused by atmosphere scintillation and scatter around the antenna of the satellite receiver. These scatters can be significant in the case of terrestrial mobile reception (*e.g.* cars, trains and pedestrian). In order to overcome the effects of multipath in DVB-S2 signal, an Adaptive Equalizer based on Decision Feedback Equalization, which architecture makes use of the DVB-S2 frame structure, is used.

Another block that is not present in the current-implemented architecture is the Fine Phase Estimation/Correction. Despite being a simple block, it was not implemented due to the fact that the phase noise present in the current-implemented receiver is not significant. Nevertheless this block is going to be implemented in the final architecture.

The Soft-Demapper present in the current architecture only works with QPSK and 8PSK modulations. The final one will cover all modulation.

The AGC block computes the ADC output power. By using the computed value and configurable thresholds, it can compute RF Gain up/down and IF or Base Band Gain up/down. These ups and downs control are implemented by two independent configurable Pulse Width Modulation (PWM) outputs. Of course depending on the tuner choice the AGC is not necessary, because some tuners, *e.g.* as that used in the ISDB-T receiver, have its own embedded AGC.

Last but not least, I would like to mention that there is a block which is not present in both architectures, but it was found to be significant for adequate autonomous functioning of the receiver, that is the block responsible for the automatic Sample

Rate estimation. This block is not covered in this thesis.

7.3 The Digital Receiver Algorithms in Details

This section details the main algorithms selected and implemented in the current architecture as well as those that will be integrated soon, such as Advanced Soft Demapper and Adaptive Equalizer.

7.3.1 Timing Recovery

Symbol timing recovery is a critical operation in any digital receiver because the receiver is not aware of the precise time of arrival of the samples, so correct detection and decoding of the symbols are not possible. The main goal of the Timing Recovery block is to estimate, as precisely as possible, the instants of start and finish of the received symbols.

Timing recovery consists of a timing measurement (the estimation of the timing error) and a timing correction (the correction of the estimated error). The timing correction can be divided into two groups depending on the kind of adopted sampling: synchronous or asynchronous. In the first group the sampling depends on the sent signal and in the second one it is independent. The asynchronous sampling is the only one which allows all-digital implementation, which is the choice in this thesis.

Timing recovery structure can be feed-forward or feedback. Figure 7.4 shows a feedback structure, that is adopted in the proposed receiver architecture. The input signal, $x(t)$, is a band limited signal and it is sampled by the Analog to Digital Converter (A/D) according to the sampling clock in T_s (in which aliasing cannot occur), generating the signal $x(mT_s)$. The Interpolator, which is responsible for the timing recovery, generates and outputs interpolants $y(kT_i)$ for interpolating the signal $x(mT_s)$ in interpolation interval T_i. The Timing Error Detector (TED), which is responsible for the timing measurement, detects a timing error from the interpolants. The Loop Filter removes a noise component of the detected timing error. The Controller extracts information from the timing error to control the operation of the Interpolator.

The proposed architecture for the Time Recovery block uses 2nd order Farrow Interpolator [93] [103] [104], 2nd order Loop Filter [113] [106] and the timing error $e(n)$ is estimated by means of the TED proposed by Gardner [114], the so-called Gardner Algorithm, which the basic equation is 7.1. The same equation, for complex baseband samples, is 7.2.

$$e(n) = y(n + 1/2)[y(n) - y(n + 1)] \tag{7.1}$$

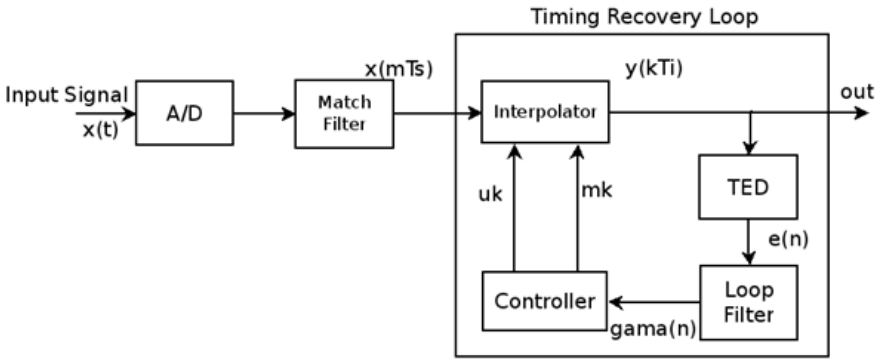


Figure 7.4: Timing Recovery with feedback structure.

$$e(n) = yI(n+1/2)[yI(n) - yI(n+1)] + yQ(n+1/2)[yQ(n) - yQ(n+1)] \quad (7.2)$$

Where, n is the samples index, $yI(.)$ is the in-phase component of the complex received symbol $y(.)$ and $yQ(.)$ is the in-quadrature component of the complex received symbol $y(.)$.

7.3.2 Frame Synchronization and PLS Decoding

Frame Synchronization is the necessary pre-requisite to correct data decoding, and needs to work with Es/No as low as -2.3 dB (see Es/No targets in [77]). Having that in mind a smart design is needed to avoid this step from becoming the system bottleneck. The typical approach is frame epoch estimation based on a pilot-aided strategy, where a known preamble is inserted in the beginning of each frame, identified as Unique Word (UW) or a Start-Of-Frame (SOF). At the receiver end, this known information is correlated with the corrupted received symbols, then Frame Synchronization is declared if correlation peak is found. Finally the PLS decoding can be performed. A simplified view of Frame Synchronization and PLS Decoding is depicted in Figure 7.5.

Figure 7.6 shows a simplified view of the DVB-S2 frame structure, that can be found in details within Section 3.2 and in [77].

The main characteristics of DVB-S2 frame structure used in the Frame Synchronization and PLS Decoding are:

- SOF with 26 known symbols;

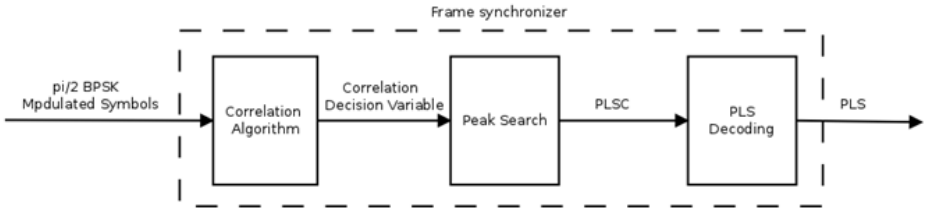


Figure 7.5: Frame synchronization and PLS decoding block diagram.

- PLS with 64 symbols, containing Coding and Modulation format and Frame Length;
- 36 pilots symbols inserted at each of the 16 slots (recall that each slot is 90 symbols long);
- The worst case total frame length (in modulated symbols) is 33282 for QPSK symbols, i.e. 26(SOF)+64(PLS)+64800(bits)/2 + 792 Pilots.

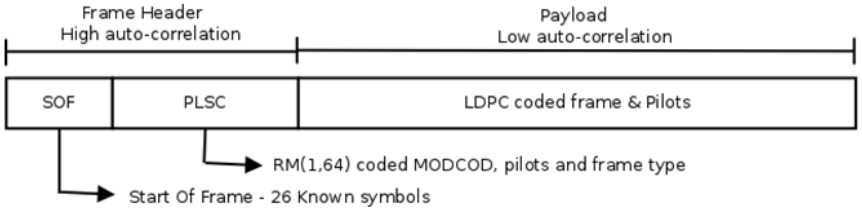


Figure 7.6: Simplified view of the DVB-S2 frame structure.

7.3.2.1 Frame Synchronization and PLS Decoding

Frame Synchronization consists of two basic steps:

1. **Acquisition** - provides coarse epoch estimation;
2. **Peak Serch** - provides fine estimation.

7.3.2.1.1 Acquisition

In short, the acquisition consists in computing the correlation within a predefined window. In this thesis the chosen algorithm for acquisition is the Differential Correlation (DC), proposed in [115], with the optimization presented in [116]. The DC with optimization is depicted in Figure 7.7. Initially, to estimate the correlation factor Λ, the signal is differentially decoded. Next, the data goes through a 89 wide shift register. Then, in parallel, two correlations are computed as follows: 1) the 25 first bits are correlated with the 25 known symbols of the SOF, that are pre-stored in 25 taps; 2) 32 out of 64 symbols are correlated with a known sequence generated from the PLS Code scrambler presented in Section 3.2.1.5.2.4, that is pre-stored in 32 taps. Finally, the sample index of the maximum values between these two correlations is chosen. The DC algorithm also provided the presence/absence of the embedded pilots. The sequence for the second correlation is:

$$-j\ j\ j\ -j\ -j\ -j\ j\ -j\ -j\ \ j\ j\ j\ j\ -j\ -j\ -j\ -j\ j\ j\ -j\ j\ j\ -j\ j\ -j\ j\ -j\ j\ j\ -j\ -j.$$

The DC equation to compute the correlation factor Λ is:

$$\Lambda = \max\left(\left|\sum_{j=1}^{L_{PLSC}/2} R_{2j-1}R_{2j}^{*}T_j + \sum_{i=1}^{L_{SOF}-1} r_i r_{2i+i}^{*} t_i\right|,\ \left|\sum_{j=1}^{L_{PLSC}/2} R_{2j-1}R_{2j}^{*}T_j - \sum_{i=1}^{L_{SOF}-1} r_i r_{2i+i}^{*} t_i\right|\right) \quad (7.3)$$

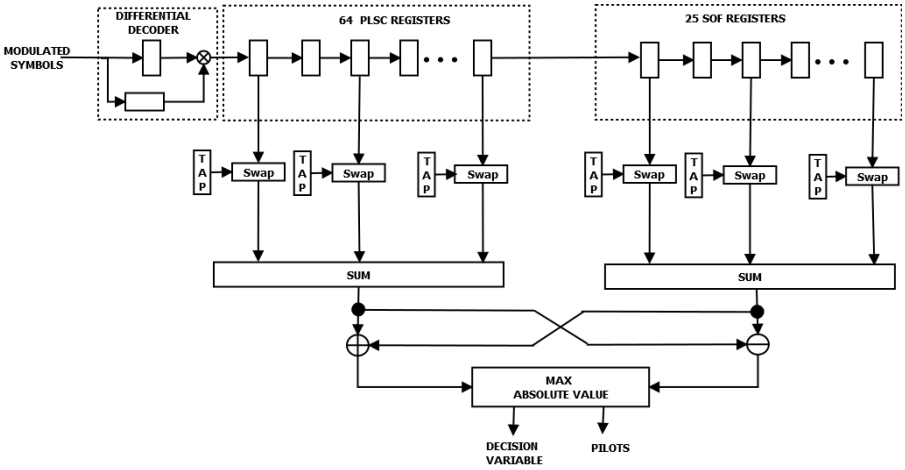


Figure 7.7: Differential detection - using SOF and PLSC.

The optimization proposed in [115] is performed within the blocks named *Swap*, in Figure 7.7. In the original DC algorithm this block is a multiplier. Nevertheless, as the values of T_j and t_i only assume the values $\pm j$, the complex multiplication can be replaced as follows:

$$(a + b * j)(\pm j) = \mp b \pm a * j. \tag{7.4}$$

7.3.2.1.2 Peak Search

The received data, continuously feeds the DC block, which computes the correlation and detects the presence/absence of embedded pilots. The presence/absence of pilots (defined by LSB of Type Field - see Section 3.2.1.5.2.3) defines the PLS sequence to be transmitted. As can be seen in Figure 3.47 the LSB of TYPE field of PLS is the bit 7 at the input of the PLS Encoder. That said, it is easy to realize that depending on bit 7 value, the PLS encoded is made of pairs of repeated symbols or pairs with the symbols followed by the respective negated values, i.e. $(y_1, y_1, y_2, y_2 ..... y_{32}, y_{32})$ or $(y_1, \overline{y_1}, y_2, \overline{y_2} ..... y_{32}, \overline{y_{32}})$. It is important to notice that this feature allows the DC algorithm to overcome the performance of the higher complexity DGPDI algorithm proposed in [117] (see also [118]) as shown in Figure 7.10, and as will be explained later, within Section 7.3.2.2.

Following the DC computation, the next step is to find the correlation peak. There are two ways of doing that search. The first method consists only in searching the correlation peak within one window. This window must have the size of the biggest modulated frame which is QPSK with pilots (33282 symbols). After detecting the highest correlation peak, the next step is to decode the PLS Code (PLSC) to find the location of the next SOF and then try to detect a correlation peak again in the expected position. This approach is threshold-based, *i.e.* synchronization will be declared if the peak is higher than a predefined threshold. The second approach is a non-threshold based approach [115] [116] [118] [119], where more than one correlation peak is used. The method adopted in this thesis is the double correlation approach proposed in [116]. Nevertheless, the method was extended for 16APSK and it been extended to 32APSK modulation.

The double peak search approach acquires the two maximum peaks within one search window. Then, it repeats the double peak search in the next search window. If the locations of the peaks, over two subsequent windows, are the same, the synchronization is declared. Otherwise the algorithm keeps looking for the double peak location. As the receiver does not know which is the transmitted modulation, the scheme performs the double peak search, for the four modulations schemes DVB-S2 allows, in parallel as shown in Figure 7.8. A decision engine is responsible for declaring the synchronization. The search windows are defined in Table 7.1. Figure 7.9 shows an example of the output of the four double peak search engines.

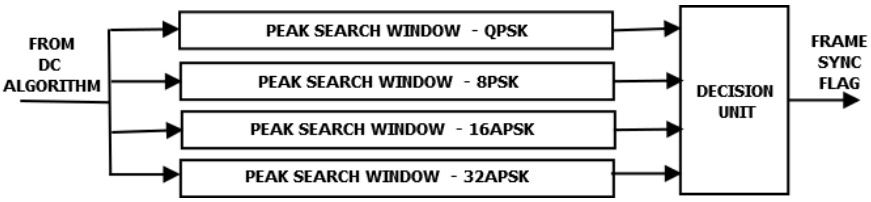


Figure 7.8: Block diagram of the double peak search scheme.

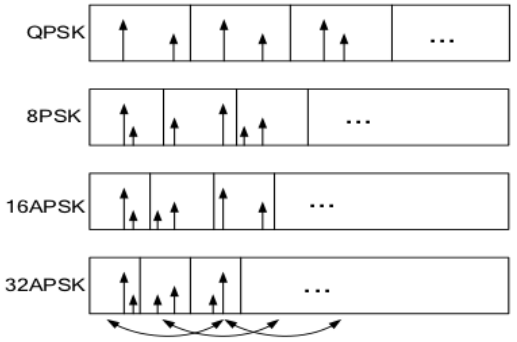


Figure 7.9: Double peak search windows.

Table 7.1: DVB-S2 Frame Sizes.

FRAME	PILOT	MODULATION	FRAME SIZE	SYMBOLS	SLOTS	PILOTS	WINDOW SIZE
NORMAL	1	2	64800	32400	360	22	33282
	0	2	64800	32400	360	0	32490
SHORT	1	2	16200	8100	90	5	8370
	0	2	16200	8100	90	0	8190
NORMAL	1	3	64800	21600	240	15	22230
	0	3	64800	21600	240	0	21690
SHORT	1	3	16200	5400	60	3	5598
	0	3	16200	5400	60	0	5490
NORMAL	1	4	64800	16200	180	11	16686
	0	4	64800	16200	180	0	16290
SHORT	1	4	16200	4050	45	2	4212
	0	4	16200	4050	45	0	4140
NORMAL	1	5	64800	12960	144	9	13374
	0	5	64800	12960	144	0	13050
SHORT	1	5	16200	3240	36	2	3402
	0	5	16200	3240	36	0	3330

Modulation values 2, 3, 4 and 5 correspond to QPSK, 8PSK, 16APSK and 32APSK respectively.

7.3.2.2 PLS Decoding

After Frame Synchronization using DC algorithm and double peak search strategy, the next step is to decode the PLSC information. For this, I propose the use of Soft RM decoding based on Fast Hadamard Transform Algorithm. Nevertheless, as mentioned in Section 7.3.2.1.2, the DC algorithm allows the detection of the presence/absence of embedded pilots. With this information in hand it is possible to know beforehand if the symbols of PLSC are either repeated or inverted. By knowing that, I propose the average of the two symbols (taking into account inversion of the second symbols, when due) and the soft demapping of the averaged symbols. This will cause a gain in the PLSC decoding scheme BER performance, regarding the DGPDI with soft demapping, as shown in Figure 7.10. DGPDI does not allow the detection of the presence/absence of a pilot. Therefore, symbol combining is not possible. Furthermore, there is no need to detect the remaining PLSC parameters before frequency and phase corrections, because those parameters will only be used in the Data Processing partition (see Figure 7.2). This is another reason for not using the DGPDI. Furthermore, DC with PLSC combining is less complex than DGPDI. One disadvantage of DC + PLC combining is that for very low Es/No, DC+PLSC takes a little longer to synchronize. But as can be seen in Table 7.2 this disadvantage is for Es/No values lower than those required for DVB-S2.

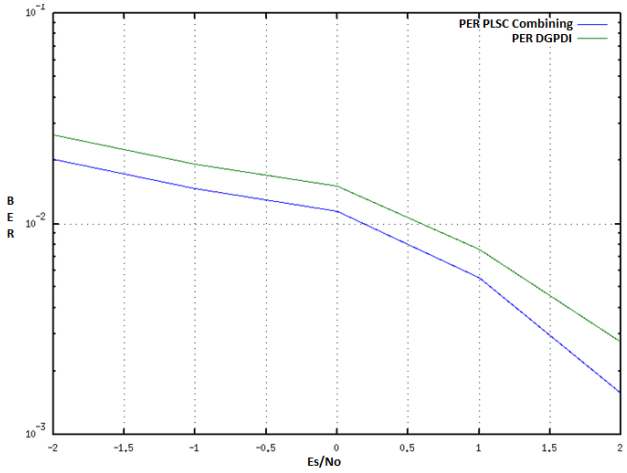


Figure 7.10: Bit error rate for PLSC combining and DGPI techniques, measured over the bits of the PLSCs subsequent to the first successful PLSC decoding.

The RM decoder accepts hard or soft bits but to execute the decoding algorithm first we have to descramble the encoded information. The descrambling can be done using an XOR operation, like in hard decision but in all soft bits as shown in Table 7.3.

Table 7.2: DC+PLSC × DGPDI Synchronization Time.

Es/No	AVERAGE PACKETS TO SYNC/100 Packets	
	DGPDI	PLSC COMBINING
-5	0.180	0.250
-4.5	0.150	0.180
-4	0.100	0.120
-3.5	0.080	0.090
-3	0.070	0.075
-2.5	0.050	0.050
-2	0.038	0.038
-1.5	0.035	0.030
-1	0.030	0.028

Table 7.3: Hard Decision × Sof Decision For 3 bits Quantization.

HARD DECISION DESCRAMBLER	$0 \times 1 = 1$	$0 \times 0 = 0$
	$1 \times 1 = 0$	$1 \times 0 = 1$
SOFT DESCISION DESCRAMBLER	$0 \times 7 = 7$	$0 \times 0 = 0$
	$1 \times 7 = 6$	$1 \times 0 = 1$
	$2 \times 7 = 5$	$2 \times 0 = 2$
	$3 \times 7 = 4$	$3 \times 0 = 3$
	$4 \times 7 = 3$	$4 \times 0 = 4$
	$5 \times 7 = 2$	$5 \times 0 = 5$
	$6 \times 7 = 1$	$6 \times 0 = 6$
	$7 \times 7 = 0$	$7 \times 0 = 7$

In mathematical terms, decoding a Reed-Muller(RM) code means multiplying the received codeword with a Hadamard matrix in order to perform the Hadamard Transform. The principle of a RM code is that the correlation between any valid codeword and any other is zero, while the autocorrelation of a codeword is 2^k (for k data bits). In DVB-S2 the Pilot information (bit 7 of PL before RM encoding) as described in 3.2.1.5.2.4 can invert the codeword sequence. The impact in the RM decoding is that when the highest absolute value decoded is positive, embedded pilots are present, and if it is negative the pilots are absent. For DVB-S2 the encoded data is 64 bits wide, and simple multiplication will cost 64^3 multiplications. So another strategy is to use a Fast Hadamard Transform (FHT) Algorithm, that gets the same results using only 80 multiplications. Figure 7.11 shows the FHT Algorithm . This algorithm accepts hard and soft bits decoding. The performance comparison for these two approaches is shown in Figure 7.12. Regarding the legend of that Figure: 1)"'Majority Decoding"' ; 2) "'PLSC Combining"' is related to the average of the two subsequent symbols of PLSC followed by Soft Demapping and Soft FHT Decoding; 3) "'FHT Soft"' is related to Soft Decoding without PLSC Combining; 4)"'FHT Hard"' is related to Decoding using hard decision bits; 5)"'$pi/2$"' is the reference performance for that modulation without coding. It is worth recalling that the adopted technique for the current and final architecture is the "'PLSC Combining"'.

```
Algorithm: FHT
Input: data – RM encoded information
Output: x – vector resultant from the multiplication data X hadamard(64)

N = pow2(floor(log2(length(data))));
x = data(1:N);
k1=N; k2=1; k3=N/2;
for i1=1:log2(N)
   L1=1;
   for i2=1:k2
        for i3=1:k3
           i=i3+L1-1; j=i+k3;
           temp1= x(i); temp2 = x(j);
           x(i) = temp1 + temp2;
           x(j) = temp1 - temp2;
        end
           L1=L1+k1;
   end
        k1 = k1/2;
        k2 = k2*2;
        k3 = k3/2;
end
```

Figure 7.11: The Fast Hadamard Transform algorithm.

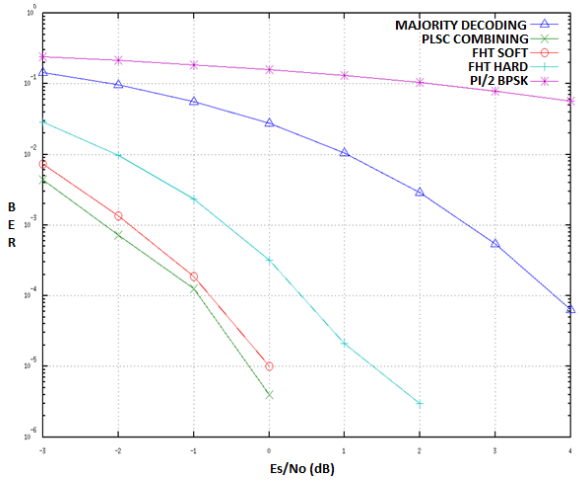


Figure 7.12: BER for PLS Decoding Techniques.

7.3.2.3 The Proposed Implementation Architecture

The block diagram of the architecture implemented in hardware is depicted in Figure 7.13.

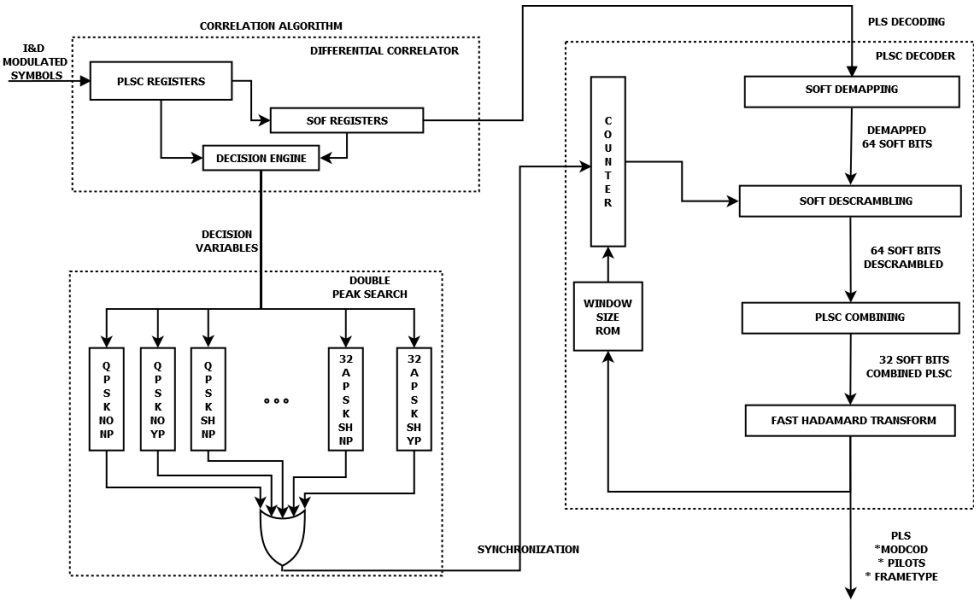


Figure 7.13: The proposed architecture for implementation.

7.3.3 Coarse and Fine Frequency Estimation (CFE and FFE) and Correction

The synchronization algorithms developed for DVB-S2 shall allow the receivers to work with existing outdoor equipment (Low Noise Block-downconverter - LNB - for DVB-S) and with low-cost local oscillators for mass-commercial production, even in very low signal to noise ratio (SNR) environments, such as those around -2.35 dB. These pieces of equipment lead to a large initial frequency offset of around $\pm 5MHz$, which represents 20% of a $25Mbaud$ symbol rate [106]. To achieve the Quasi Error Free (QEF) performance, specified by [77], in such a scenario, the FFE has to provide a normalized residual frequency offset smaller than 5.2 x 10^{-5}. In order to achieve this residue, the conventional approach in DVB-S2 systems concatenates CFE and FFE. The CFE can be based on algorithms such as those presented in [108] and [109], and the FFE on the algorithm presented in [106], which is a modified version of L&R algorithm proposed by [120]. The CFE and FFE algorithms mentioned previously,

make use of the DVB-S2 frame structure, already presented in this thesis. However, clarity, it is reproduced in Figure 7.16.

In this phase of the project, I propose the use of an architecture for coarse and fine frequency estimation and correction to support DVB-S2 working on pilot mode, *i.e.* with embedded pilots. Nevertheless the CFE is computed using the SOF field, and the FFE uses the embedded pilots. In spite of that, it is possible to extend the FFE to be used in non-pilot mode, by recreating the 90 encoded symbols of the PLSC Header after PLSC decoding and Frame detection, and using them as pilots to perform the estimations, as shown in [121]. The entire PLSC Header is necessary to enhance the performance of FFE and can be used to enhance the performance of CFE, that currently only uses 26 symbols. The price to be payed, regarding hardware implementation, is that RM encoding and PLSC modulation shall be done at the receiver side.

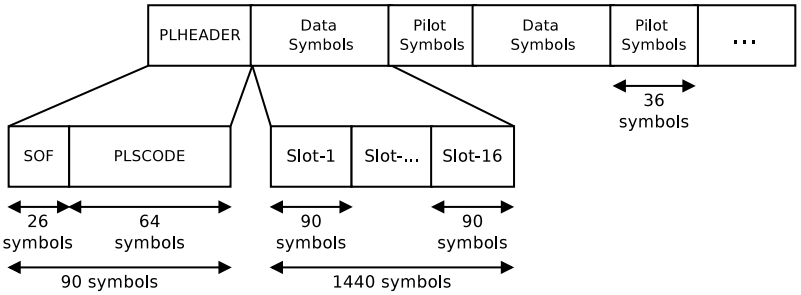


Figure 7.14: DVB-S2 frame strucuture used by the Fine Frequency Estimator.

7.3.3.1 Coarse Frequency Estimation and Correction

In this thesis, I propose the use of the algorithm for carrier frequency offset estimation presented in [108], as the technique for CFE. The initial architecture was based on a direct implementation of the algorithm. Nevertheless, as in the case of the FFE (see Section 7.3.3.2), it was found an unnecessary resource usage to implement the correlations, $R(m)$, in parallel, because the CFE only uses the 26 pilots of SOF to compute the estimation. This new approach will save a significant amount of resources (see Chapter 8). The algorithm for carrier frequency offset estimation proposed in [108], shown in Figure 7.15 works as follows:

1. Compute the autocorrelations $R(m)$ as:

$$R(m) = \frac{1}{N-m} \sum_{n=m}^{N-1} r_n r_{n-m}^* \tag{7.5}$$

Where: m is the index of the pilot of the SOF field and $m = 1, \cdots, L$ ($L \leq N-1$); N is the window size , that in this case is the length of the SOF, that is 26; $n = 0, \cdots, N-1$; and L is the observation window, that shall be at maximum half of N;

2. Compute the argument of $R(m)$, form $m = 1, \cdots, L$, using a CORDIC:

$$\theta(m) = \arg[R(m)] \tag{7.6}$$

3. Compute the phase difference Δ_m:

$$\Delta_m = \begin{cases} (\theta(1), & m = 0 \\ \theta(m+1) - |\theta(m))\mathrm{mod}(2\pi), & 1 < m < L \end{cases} . \tag{7.7}$$

4. Compute the Weight Function w_m^*:

$$w_m^* = \frac{1((2L+1)^2 - (2m+1)^2)}{((2L+1)^2 - 1)(2L+1)}, \quad m = 0, \cdots, L-1 \tag{7.8}$$

5. Finally, the frequency error, f, can be computed as:

$$f = \frac{1}{2\pi T} \sum_{m=0}^{L-1} \omega_m^* \Delta_m \tag{7.9}$$

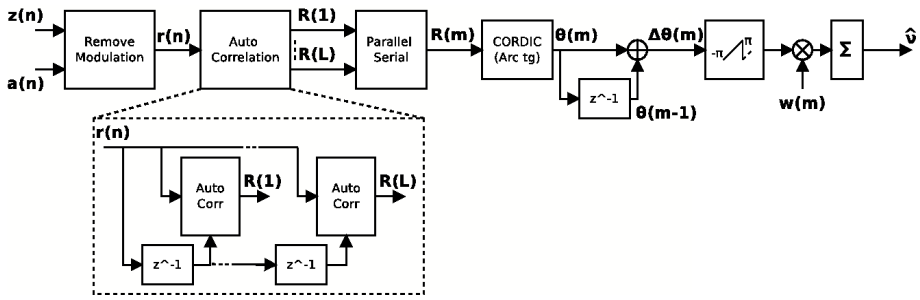


Figure 7.15: Coarse Frequency Estimation Correction.

7.3.3.2 Fine Frequency Estimation and Correction

For FFE, the proposed algorithm is a modified version of L&R [120] presented in [106]. Nevertheless, the conventional implementation of the L&R algorithm presents

a considerable hardware complexity [120], *i.e*, large amount of multipliers and accumulators. Even the variations of this algorithm, that aim to reduce the frequency residue, keep a hardware complexity similar to the original. Several approaches are proposed to reduce the hardware complexity, such as those presented in [122], [123]. In this thesis I propose a new serial architecture to implement the modified L&R algorithm [106] that meets the normalized residual frequency offset required by DVB-S2, with a considerable reduction of hardware complexity. The proposed FFE implements the pilot-based feedforward algorithm presented in [106] derived from [120], taking advantage of the DVB-S2 pilot structure (see Figure 7.16) to achieve efficient frequency synchronization. Equation 7.10 is used to obtain the frequency estimation based on L consecutive pilot fields,

$$\hat{\nu}T = \frac{1}{\pi(N+1)} arg\{\sum_{l=1}^{L}\sum_{m=1}^{N} R_l(m)\}, \tag{7.10}$$

where $R_l(m)$, $m \in \{1, ..., N\}$, are the N points of the autocorrelation computed over the L_p pilot symbols of the l_{th} pilot symbol field using Equation 7.11.

$$R_l(m) = \frac{1}{L_p - m} \sum_{k=m}^{L_p-1} r^{(\mathrm{p})}(k) r^{(\mathrm{p})*}(k-m). \tag{7.11}$$

Finally, $r^{(p)}(k)$ is computed as in Equation 7.12, where $z(k)$ is the k_{th} received pilot symbol of the p_{th} pilot symbol field, and $c(k)$ is the k_{th} pilot symbol used as reference to the data-aided algorithm in question.

$$r^{(\mathrm{p})}(k) = z^{(\mathrm{p})}(k) c^{*}(k) \tag{7.12}$$

In short, the frequency estimation of this algorithm is achieved by accumulating the correlations between the received pilot symbols and the known training sequence. In [106], to improve the performance, the N point autocorrelation is averaged over L consecutive pilot fields before computing the argument.

To perform the fine frequency estimation, the FFE must compute the N points of the $R_l(m)$. Such operation demands an amount of hardware proportional to the N parameter, requiring a considerable number of complex multipliers. Figure 7.17 illustrates the conventional architecture presented in [120] to implement this portion of the estimator, and corresponds to the blocks within the dashed box in Figure **??**.

Concerning how the estimation is done in Figure 7.17, first of all, the N-length shift register is driven with the complex-conjugated pilot symbol sequence. At each clock step, the content of the register is shifted, multiplied by the weights $1/(L_p - m), m \in \{1, 2, ..., M\}$, and added up. The output of the sum is multiplied by the input signal $r(k)$ and the result is accumulated, calculating: $\sum_{m=1}^{N} R_l(m)$.

The process of multiplying the shifted content by the weights and adding them can be considered as the one done by an FIR filter. In this case, the structure of the filter

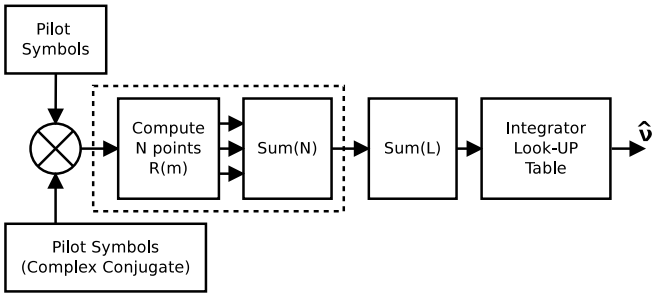


Figure 7.16: Conventional Fine Frequency Estimator.

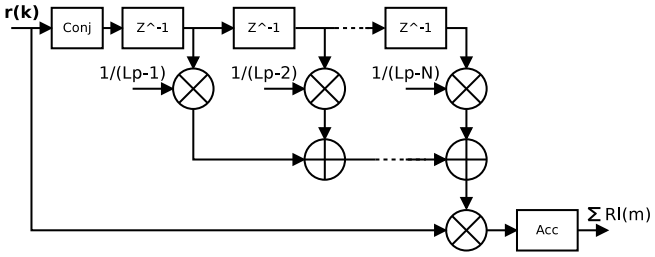


Figure 7.17: Partial architecture of the conventional Fine Frequency Estimator.

would contain N multipliers and $N-1$ adders. Each of the multipliers and adders of this portion of the structure are implemented as two decoupled real-input components. Furthermore, the additional multiplier is a true complex multiplier composed of four real multipliers and two real adders.

7.3.3.2.1 The Proposed Architecture

The proposed architecture is illustrated in Figure 7.18. The blocks that compose the architecture are grouped into modules according to their specific functions: $R_l(m)$, Sum, Angle Step and Iteration Controller.

The $R_l(m)$ is formed by two buffers that store the input and known pilot symbols separately. It also contains a block for calculating the symbols conjugate (Conj.), an Accumulator and a Shifter. Without the optimization, the Shifter block performs multiplications between each of the weights represented by $1/(L_p - m)$ and the correlation result. However, in the proposed approach, the multiplications are replaced by shifts in the accumulated result. The $1/(L_p - m)$ factor, which weight the $R(m)$ function, is approximated to the inverse of the nearest power of two. And that multiplication can be implemented by an amount of shifts to the right equivalent to the correspondent power. Moreover, as will be shown later, the approximation of the weights does not affect the fine frequency estimation.

The Sum module performs both summations in Equation 7.10, $\sum_{l=1}^{L}$ and $\sum_{m=1}^{N}$. The Angle Step module, containing the Average and Argument blocks, provides the final result of the FFE, *i.e.* the necessary angle step to correct the fine frequency error. The argument is computed using the CORDIC algorithm in vectoring mode and circular coordinates.

The Iteration Controller block contains three counters, that generate the indexes k, m and L, and interacts with the other blocks to make possible the reuse of both, the correlators and the corresponding hardware structure. Therefore, the autocorrelation becomes a loop controlled by a state machine. A simplified diagram of this state machine is depicted in Figure7.19.

The limitation of the proposed architecture lies on the number of cycles required to calculate $\sum_{m=1}^{N} R_l(m)$. Also, the autocorrelation of all pilot symbols, of a pilot field, demands less cycles to be performed on each of the N terms of the summation. This variation of cycles occurs according to $L_p - m$, $m \in \{1, ..., N\}$. In the case of DVB-S2, $L_p = 36$ and $N = 18$. Consequently, the minimum number of cycles, required by the FFE, between two pilot fields must be 477 cycles. This number of cycles is less than the 1440 available cycles until the arrival of the next pilot symbol.

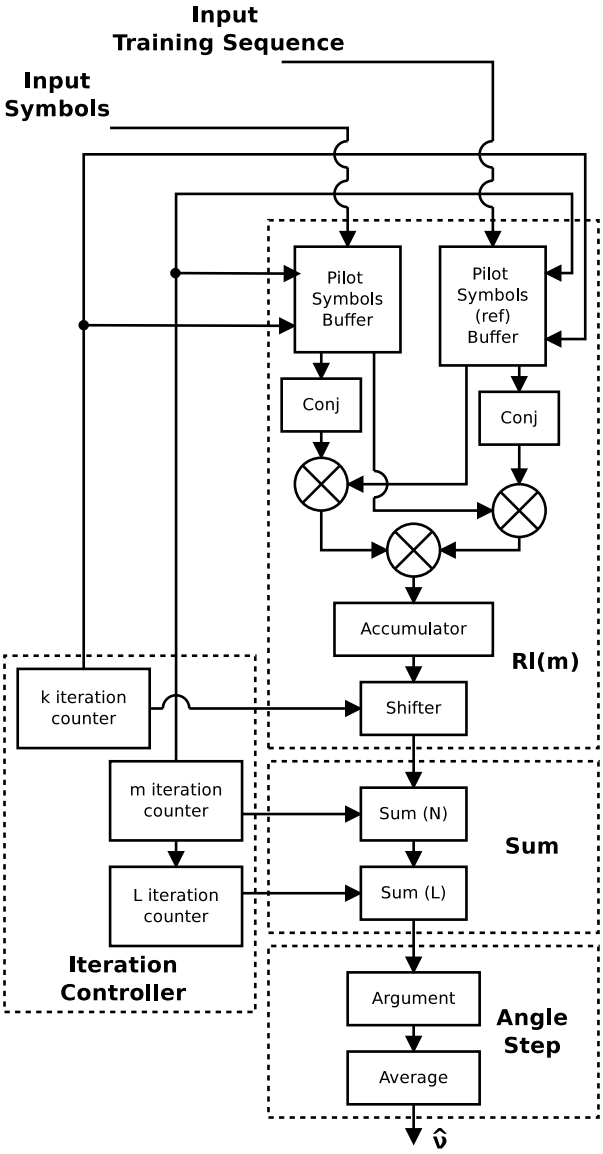


Figure 7.18: Proposed Fine Frequency Estimator.

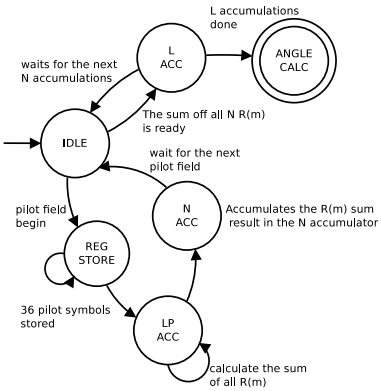


Figure 7.19: Diagram of the state machine that controls the correlator.

7.3.3.2.2 Simulations

The L and N parameters, which define the upper limits of both summations, are defined by simulation as mentioned in [106]. Where $L = 1000$ and $N = 18$ guarantee that the residual frequency error is always lower than the maximum allowed (*i.e.* 5.3×10^{-5}). The models were quantized in order to reflect, in simulations, the behavior of the implemented hardware. Also, the simulations were performed assuming perfect symbol timing synchronization.

The operation range of the estimator for positive frequencies can be seen in Figure 7.20, the negative part of the plot showed same behavior. The total operation range goes from -5% to 5% of the input frequency, that is $\pm 1.25 MHz$ at $25 Mbaud$. However, in DVB-S2 systems, the maximum residual offset to be corrected by this algorithm is assumed to be lower then $\pm 4e^{-3}$.

Figures 7.21 and 7.22 present the performance of the FFE under the variation of Es/No, maintaining a constant coarse frequency residual, and under the variation of this residual, keeping the Es/No constant. The mentioned residual is the frequency error of the FFE input signal.

The performance of the FFE is degraded as the input frequency error increases, as can be verified in Figure 7.21. This plot shows that, approximately at 1.2% of the input frequency, the output residual error rises to the order of 10^{-4}. During the entire simulation, Es/No is kept in $-2dB$.

Figure 7.22 shows the result of a simulation where the frequency error was fixed in 4×10^{-3} , which is the expected maximum error provided by a coarse frequency algorithm such as [106]. The fine frequency residual is lower then the target of $5.3 \times$

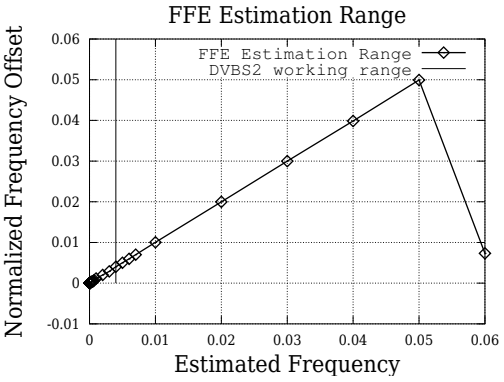


Figure 7.20: FFE estimation range.

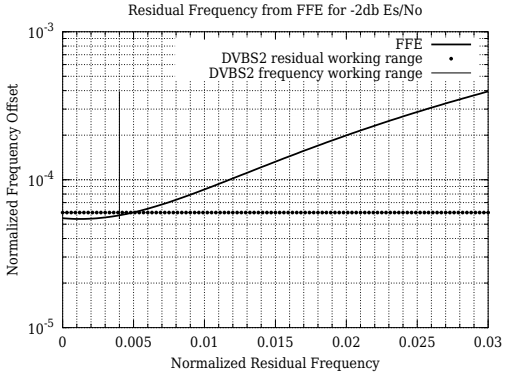


Figure 7.21: Normalized fine frequency estimation error for the operation range.

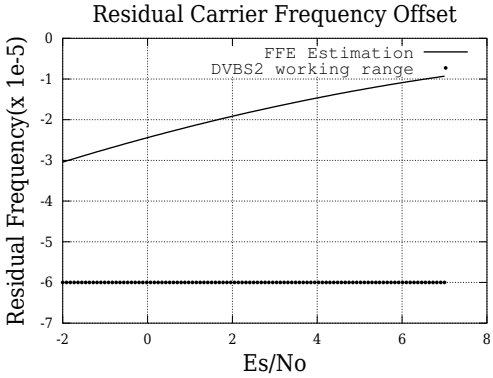


Figure 7.22: Residual frequency offset after the FFE.

10^{-5} and smaller than 1×10^{-5} for Es/No greater than $7dB$. As shown in the results the serial implementation of the estimator does not affect the performance.

7.3.4 Coarse Phase Estimation and Correction

When processing the received signal, after timing and frequency correction, a phase error is present and needs to be corrected. This shall be done after the receiver is synchronized and the pilots are identified. This block estimates the coarse phase error present in the received signal, using the frame pilots. The estimation is performed through correlation of the transmitted known pilots with the received pilots corrupted by the channel, followed by a phase error computation. After the phase computation, phase de-rotation is applied in all the subsequent symbols, by means of a CORDIC, until the next pilots field, where a new estimation is performed. It is worth mentioning that in this phase of the project a fine phase correction is not performed. Nevertheless, in the future, after the integration of the Adaptive Equalizer (see Section 7.3.7) , the coarse and fine phase errors shall be corrected by that block.

The scheme adopted in this thesis is based on the approach presented in [106]. Nevertheless, interpolation is not performed. Furthermore, in this thesis, the phase computation and correction are performed using the CORDIC algorithm, while in that work the authors make use of look-up table. Figure 7.23 shown the detailed proposed architecture.

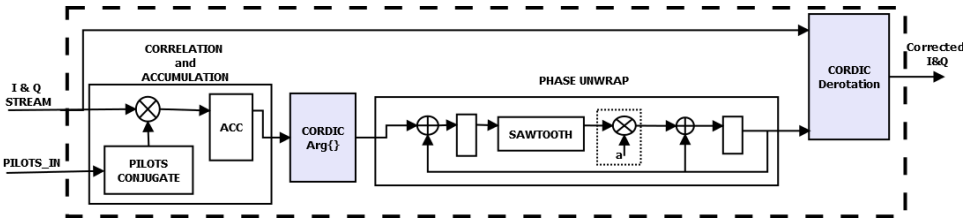


Figure 7.23: Detailed architecture for the coarse phase estimation and correction.

7.3.5 Digital Automatic Gain Control (AGC)

Even after applying the RF, IF or BB AGCs, which are performed in the RF front-end (i.e. tuner) with support of the digital receiver, the signal converted by the ADC can present values above/below of those expected by the digital receiver, i.e. the constellation radius/radii (depending on the constellation) is/are smaller or larger the is/are expected to be. This mismatch affects the right definition of the demapper's decision regions, the computation of the SNR and by consequence the soft information will not be computed correctly. Consequently, the BER after demapping and, of course, after the FEC Decoding subsystem can increase significantly, depending on the mismatch level. Therefore, in order to recover the signal amplitude a Digital AGC shall be used. This block is placed after the phase correction. It is worth to mention at this point that when using the Adaptive Equalizer (see Section 7.3.7) this block may not be necessary.

The approach used in the receiver proposed in this thesis is the one presented in [106], which is a variation of the complex decision-directed vector tracker (DD-VT) [124]. Nevertheless, a modification is proposed in this thesis, that is the rotation of all pilots to the first quadrant, before the computation of the correlation and gain. This makes the storage of the embedded pilots unnecessary. The method proposed in [124] uses the pilot symbols exploited for carrier phase estimation also for AGC thus adopting the data-aided version of the vector-tracker AGC (DA-VT AGC) [106].

The AGC only works when the analyzed current symbols are pilots, then it calculates the correlation between the ideal pilots and the received pilots to estimate the amplitude difference and then outputs the corrected symbol. The gain calculation is accumulative and takes into consideration every pilot field in the transmission. The block diagram of the AGC is shown in Figure 7.24. The flag PILOTS_IN, is active, i.e. it is equals to 1, when the current symbols is a regular pilot.

Using the pilot fields, the AGC estimates how far are the received symbols from the ideal pilots. Next, using the calculated gain coefficient, it applies the same scale in the payload until the next pilot. Then, it updates the gain coefficient. Equation 7.14 shows how the calculations are performed. Observe that in Equation 7.14 the

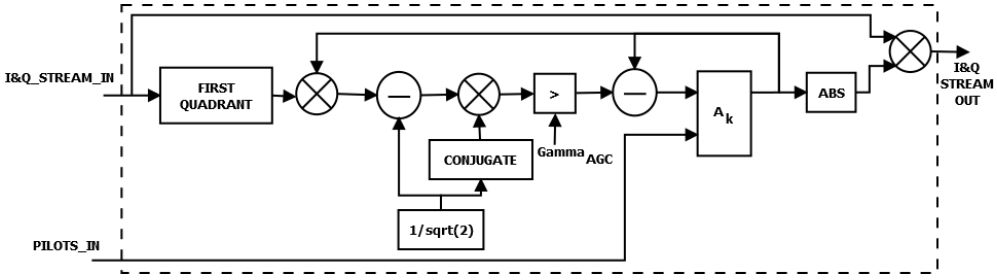


Figure 7.24: Architecture of the digital AGC.

coefficient update is only active when analyzing pilot symbols.

Given the received symbols, $z(k)$ - where k is the sample index - after timing and frequency synchronization, and coarse phase correction, the symbols with gain compensation after AGC, $s(k)$, are given by Equation 7.13.

$$s(k) = A_k z(k), \qquad A_k = |\alpha_k| \tag{7.13}$$

The gain α_k is obtained as it follows:

$$\alpha_{k+1} = \begin{cases} \alpha_k - \gamma_{AGC}[\alpha_k z^p(k) - c^p(k)]c^p(k)^*, & \text{for } k \in \text{pilot symbols} \\ \alpha_k, & \text{otherwise} \end{cases}. \tag{7.14}$$

Where, γ_{AGC} is the AGC adaptation step.

7.3.6 Physical Layer Descrambler - PL Descrambler

Before presenting the architecture for the PL Descrambler, it is worth to recall that, as it was introduced in Section 3.2.1.5.4, the parameter n, which defines the mask of the the PL Scrambler, is equal to zero for broadcasting services. Despite the main application of the proposed receiver is reception of broadcasting services, in order to extend the range of applications of the receiver, I propose a PLD Descrambler architecture that supports multiple values of n, by means of two configurable masks, namely M_X and M_Y, the first for the polynomial X and the late for Y. The values to be used by the masks, that by default are those related to $n = 0$, can be read via register bank from an external application (e.g. via I2C or SPI communication ports)

or can be stored into the Processor, used to configure the receiver [1].

The architecture of the proposed PL Descrambler is shown in Figure 7.25

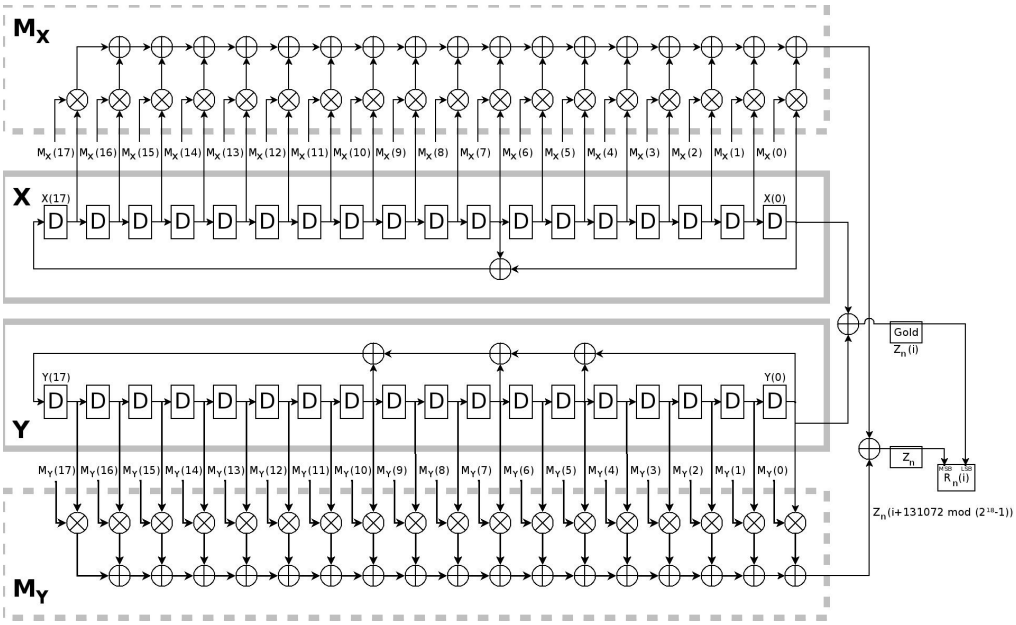


Figure 7.25: Architecture of the PL Descrambler Sequence Generator (R_n), with configurable masks.

Figure 7.26 shows, in dashed line, the MSB (the signal bit) of each I+jQ pair, that are the complex symbols, which have to be descrambled. The MSBs (i.e the sign) of the I and Q components are the part of I&Q that suffers modification during the scrambling/descrambling process. The rest of the data bits (N-1:0) of each vector represents the magnitude of the symbol (continuous line) and are preserved. The implementation here described takes into account that the symbol representation is implemented with some binary signed representation like, as for example, signal-and-magnitude or 2's complement.

7.3.7 Adaptive Equalizer

This section presents a proposal of an Adaptive Equalizer based on DFE, and its architecture for implementation in FPGA and ASIC, to make DVB-S2 transmission reliable under ISI (specially for wideband transmissions) and to make the mobile re-

[1] Commercial receivers, usually have a Processor used for configuration purposes.

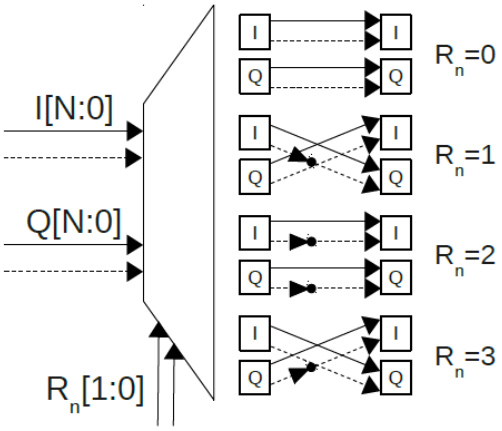


Figure 7.26: Descrambling.

ception over satellite links for QPSK modulation possible. The Equalizer also presents the capability to reduce the nonlinear distortion effects (this is going to be shown in Section 7.3.7.2.2), which are typically found in the satellite link and are caused by the satellite power amplifiers. This mainly affects high order modulations. The implementation is very flexible because it can make use of CMA and DD algorithms as well as the DVB-S2 header and embedded pilots as training sequences. Its adaptation is performed using the LMS algorithm.

A significant feature of the DVB-S2 standard is to support high-order modulations, such as 16-APSK and 32-APSK. These modulation schemes, although specifically designed for nonlinear channels, are particularly sensitive to the characteristics of the satellite transponders. Computer simulation studies, based on the use of satellite transponder models, demonstrated that there are significant opportunities to further enhance the performance by predistortion of the transmitted signal and/or Intersymbol Interference (ISI) suppression technique in the receiver [77]. Most of the time, the channels and, consequently, the transmission systems transfer functions are unknown. Also, the channel impulse response may vary with time. The result of this is that the equalizer cannot be fixed or previously designed. So, the most preferred scheme is to exploit adaptive equalizers. An adaptive equalizer is an equalization filter that automatically adapts to time-varying properties of the communication channel. In other words, it is a filter that self-adjusts its transfer function according to an optimizing algorithm.

In this thesis I propose the use of a Decision Feedback Equalizer (DFE) which exploits the DVB-S2 frame structure and makes use of Constant Modulus Algorithm (CMA) [125] and Decision-Directed (DD) algorithms, as well as the DVB-S2 header and embedded pilots as training sequences. This adaptive equalizer will be integrated

in the receiver proposed in [20] to make the communication link reliable in the presence of ISI.

Initial studies, presented in Section 7.3.7.2.2, has suggested the potential use of DFE to minimize the effects on nonlinear distortion. Especially when using Pilot-Based combined with and DD training. Instead of CMA, as used in this work, the results presented in [126] suggests that Generalized Multilevel Constant Module Algorithm can be used.

The DFE decision device is based on the well known CORDIC algorithm [127] and the adaptation is obtained by means of the Widrow-Hoff Least Mean Squares (LMS) algorithm [128] [129].

Generally speaking the satellite wireless channel impairments can be categorized as either linear (dispersive) or nonlinear. The main consequence of dispersion is the occurrence of ISI. In DVB-S2, the channel bandwidth can vary from a couple of kHz to tens of MHz. Therefore, as the bandwidth of the signal increases, the signal is more susceptible to ISI. In satellite systems, ISI mostly occurs due to scattering caused by vegetation foliage and buildings in the receiver neighborhood [130]. Nevertheless, ISI can also occur in the cable distribution system for satellite signal distribution from a single antenna to several users at different floors within a building. In this case, ISI is caused by the impedance mismatch at the user outlets and distribution taps that generate multiple short echoes [131]. On the other hand, the nonlinear impairments occur due to the characteristics of the satellite power amplifiers, specially the old satellites devoted to analog transmission and re-used for digital systems. Refer to [106] for the characteristics of power amplifier nonlinearities, defined by the DVB-S2 committee.

DFE is the most common nonlinear equalizer. One of its advantages over linear equalizers is that it does not amplify the noise while reducing the ISI that is caused by the spread of the symbols impulse response over a number of symbol samples, as illustrated by Figure 7.27.

The echoes before the correct sampling are called precursors and the ones after are named poscursors. In this work, the DFE is also used for compensating nonlinear distortions, because it is considered to be a powerful technique to cope with nonlinear satellite channel distortion (see [122] and references therein). A common DFE diagram is shown in Figure 7.28.

7.3.7.1 Proposed Equalizer Architecture

The proposed architecture incorporates three main features: (1) a resource-sharing *FIR-LMS Core* implementation, (2) an automatic error tracking control and (3) the

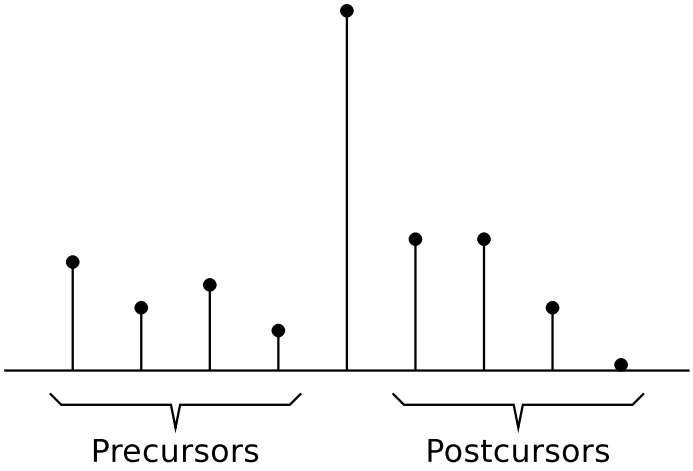


Figure 7.27: An impulse response that causes ISI.

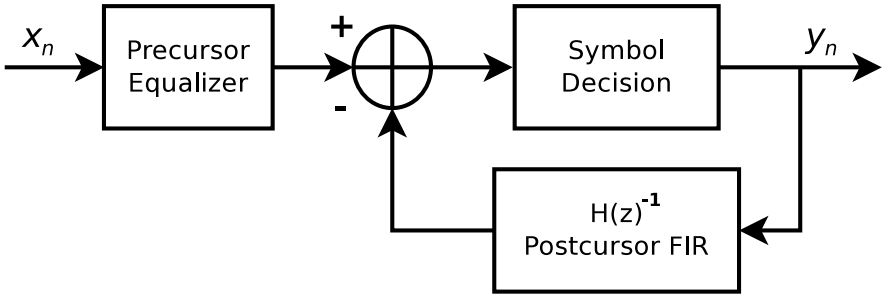


Figure 7.28: Common DFE diagram

utilization of the *CORDIC* for performing arithmetic/trigonometric operations within sub-blocks.

Moreover, it is worth noting that in this work the DFE design considers only area and latency constraints. The design still has space for power consumption reduction, especially when targeting handheld devices. Figure 7.29 illustrates the proposed architecture of the DFE containing its main sub-blocks. Both, the $\boldsymbol{x}_n$ input and the $\boldsymbol{y}_n$ output, have a 16-bit width.

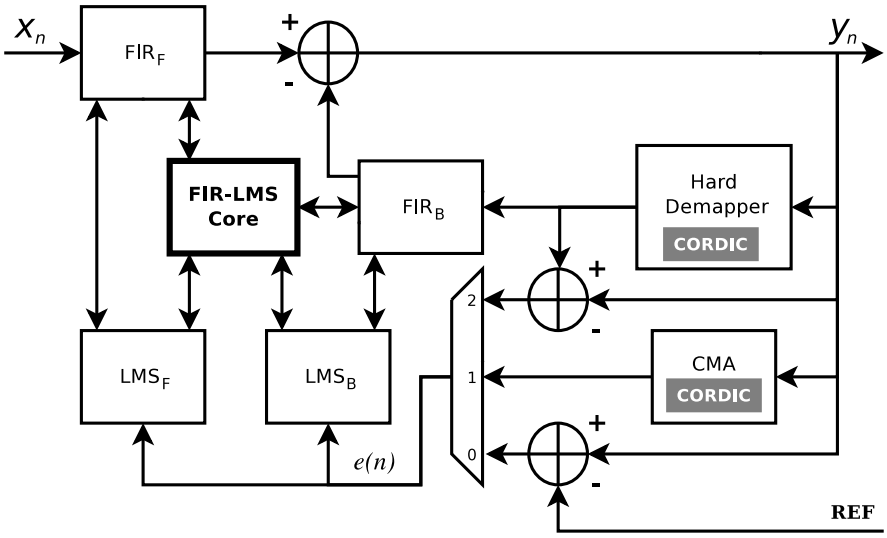


Figure 7.29: Proposed architecture for the bound rate adaptive equalize.

The sub-blocks shown in Figure 7.29 and the error tracking are explained in the following subsections.

7.3.7.1.1 FIR-LMS Core

In the digital signal processing area, one of the main algorithms used for adaptive filtering is the LMS. Its low computational complexity, relatively easy implementation and efficiency in a wide range of applications, makes it attractive for hardware implementation.

The FIR filter calculates the output signal y_n according to Equation 7.15, and the LMS updates the filter coefficients according to Equation 7.16, where $\boldsymbol{x}_n$ is the filter input vector, which contains 16 complex elements, $\boldsymbol{w}_n$ is the filter coefficients vector, which also contains 16 complex elements, μ is the step size of the LMS, and e_n is the computed complex error.

$$y_n = \boldsymbol{x}_n^{\intercal} \cdot \boldsymbol{w}_n \tag{7.15}$$

$$\boldsymbol{w}_{n+1} = \boldsymbol{w}_n + \mu \cdot e_n \cdot \boldsymbol{x}_n \tag{7.16}$$

Since the FIR and the LMS sub-blocks use the same arithmetic operators (i.e. complex adders and complex multipliers), and the DFE incorporates two FIR Filters and two LMS instantiations (one for forward and another for backward filtering), a resource sharing architecture was adopted. The chosen step-sizes for the Forward and Backward Filters, for the simulations presented in this Section, were $\mu_F = 0.01$ and $\mu_B = 0.001$, respectively. Those parameters are configurable, in order to be adjusted in a adequate manner after integration in the DVB-S2 receiver and for other operation scenarios (e.g. for different power delay profiles and signal levels).

7.3.7.1.2 Error Tracking

As shown in Figure 7.30, a DVB-S2 frame contains 90 symbols that correspond to the *PLHEADER*, and additionally to the data symbols (grouped by slots) it may also contain embedded *pilot* symbols [77].

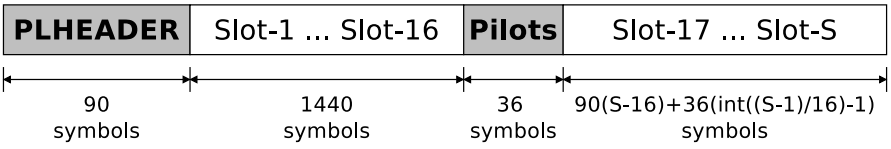


Figure 7.30: DVB-S2 physical frame with pilots

A typical operation procedure of the proposed equalizer is: it starts in training mode using the *PLHEADER* gathering information about the channel, and afterward, it switches to the CMA blind mode as shown in [132]. Next, it goes to Decision-Directed (DD) mode or uses the embedded *pilot* symbols, if they exist. CMA is used before DD due to the fact that any DD scheme is prone to misconvergence, which causes the equalizer locks to a rotate constellation state [133] [134].

When in training mode, the equalizer calculates error e_n either in a supervised or a blind fashion. The former takes place when receiving *PLHEADER* or embedded *pilot* symbols, since there is a known sequence sent by the transmitter that can be reproduced within the receiver to serve as reference. The blind equalization is executed while receiving data symbols, thus no reference is received for improving the filters weights computation, in this case, the CMA algorithm is the right choice. On the other hand, when in the DD mode, the symbols outputted by the *Hard Demapper* (HD) are fed to the backward filter, and also used for calculating e_n.

Switching between the two modes is determined by the central TAP trigger value, which is reached after a number iterations, indicating that the gains of FIR_F and FIR_B were correctly updated. Table 7.4 details the appropriate e_n signal according to the current modulation and central TAP value.

Table 7.4: Error tracking functionality

Modulation	**Central TAP**	$e(n)$
QPSK & 8-PSK	-	REF - $y(n)$
	< 0.4	CMA
	≥ 0.4	HD_{out} - $y(n)$
16-APSK & 32-APSK	-	REF - $y(n)$
	< 0.9	CMA
	≥ 0.9	HD_{out} - $y(n)$

7.3.7.1.3 CMA

The Constant Modulus Algorithm (CMA) is useful when no training signal is available, and works best for constant modulus modulations such as PSK. However, if the CMA has no additional side information, it can introduce phase ambiguity. For example, the CMA might find weights that produce a perfect QPSK constellation but might introduce a phase rotation of 90, 180, or 270 degrees. Alternatively, differential modulation can be used to avoid phase ambiguity [106].

As expressed by Equation 7.17, a CMA error computation requires a low computational complexity. Moreover, the *CORDIC* is used to perform the magnitude operation. The *HD* and the CMA execute simultaneously on the same stage of the Finite State Machine (FSM), which implies the instantiation of dedicated *CORDIC* sub-blocks for each one.

$$e_{CMA} = y_n(R - |y_n|^2) \tag{7.17}$$

Where y_n is the equalized data (see Figure 7.29) and R is the Godard radius. For QPSK and 8-PSK, $R = 1$, while for modulations of 16-APSK and 32-APSK we use an average constellation radius that depends on the γ parameter, described in [77], which on its turn depends on the used code rate.

7.3.7.1.4 Decision Device

The decision device used in the Adaptive Equalizer is a Hard Demapper that will be introduced in Section 7.3.9. Nevertheless, instead hard bits the decision device delivers hard symbols.

7.3.7.2 Simulation Results

7.3.7.2.1 Inter Symbol Interference - ISI

In this Section I present simulation results for the model and the HDL simulations for four DVB-S2 receiver configurations. Table 7.5 shows the configuration used in simulation results.

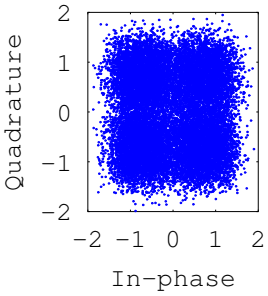
Table 7.5: Simulation configuration parameters.

Modulation	MODCOD	$E_s/No(dB)$	Channel[1]
QPSK 1/4	1_D	$(-2.35 + 12.00) = 9.65$	[1 .4]
8-PSK 3/5	12_D	$(5.50 + 7.15) = 12.65$	[1 .4]
16-APSK 2/3	18_D	$(8.97 + 6.00) = 14.97$	[1 .4]
32-APSK 3/4	24_D	$(12.73 + 6.00) = 18.73$	[1 .4]

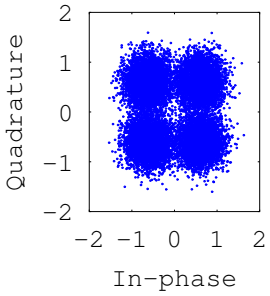
[1] The multipath channel used for the initial evaluation has only two sample-spaced taps. In 7.3.7.2.2 we evaluate the equalizer using a channel with longer PDP.

Figure 7.31, 7.32, 7.33, and 7.34 show the simulation results of the DFE. Each figure has five plots that match with (a) GNU Octave Model input, (b) GNU Octave Model output, (c) HDL output, (d) Instantaneous Error and (e) Central TAP convergence & Canceled ISI. The HDL simulations of the DFE use the same input as the golden model, but quantized with 16-bit in Q3.13 (meaning thirteen bits for the fractional part, two for the integer part e one for the sign) in fixed-point format. The first MODCOD of each modulation was used in order to test the DFE with the worst theoretical condition - the lowest Signal to Noise (SNR) - among all other MODCODs of the same modulation, as listed in Table 7.5.

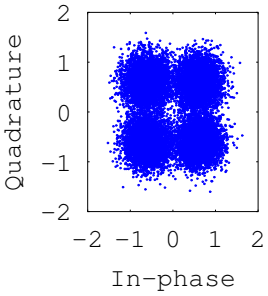
Moreover, as can be seen in Figures 7.31(c), 7.32(c), 7.33(c) and 7.34(c), the output of the golden model and of the HDL code have the same behavior. This validated the HDL implementation, and consequently the passing to the FPGA prototyping phase. Item (d) of those figures shows the computed error in dB for the golden model for each modulation. The subfigures (e) show the central TAP convergence and the canceled ISI. The results presented in subfigures (d) and (e) are synchronized in time, for each modulation. It can be realized from the observation of those figures that the number of samples required for convergence are from approximately 4000 samples (for QPSK) to around 12000 samples (for 32-APSK). This is less than the length of one *Short Frame*, and can be considered a adequate time that does not lead the user to have a annoying experience during the synchronization. Just to give numeric examples: for a 25 MHz bandwidth, 12000 samples take 480 μs; for a 300 kHz bandwidth, 12000 samples takes 40 ms.



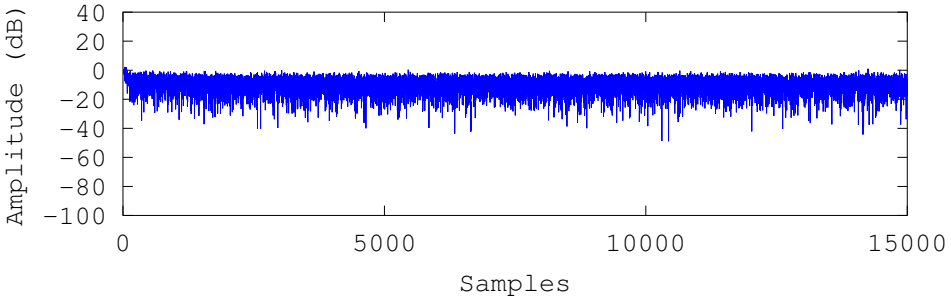
(a) Model input.



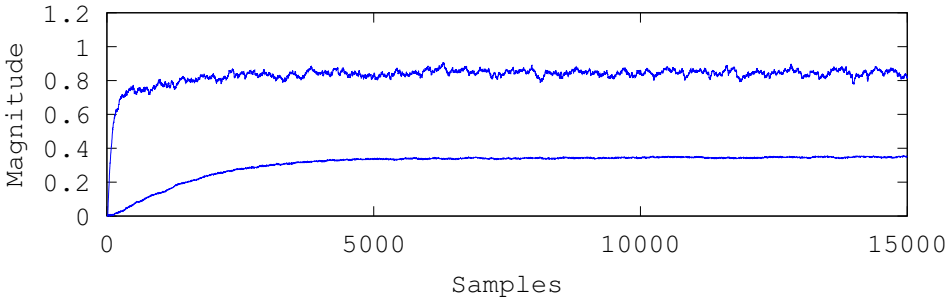
(b) Model output.



(c) HDL output.

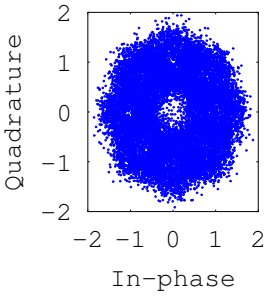


(d) Instantaneous error.

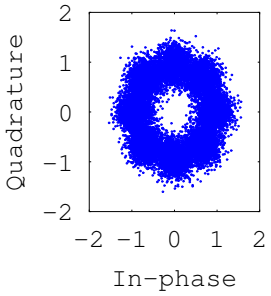


(e) Central tap & cancelled ISI.

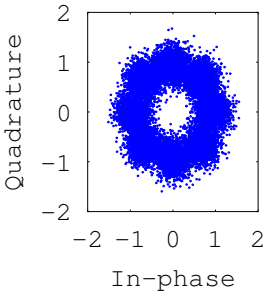
Figure 7.31: QPSK.



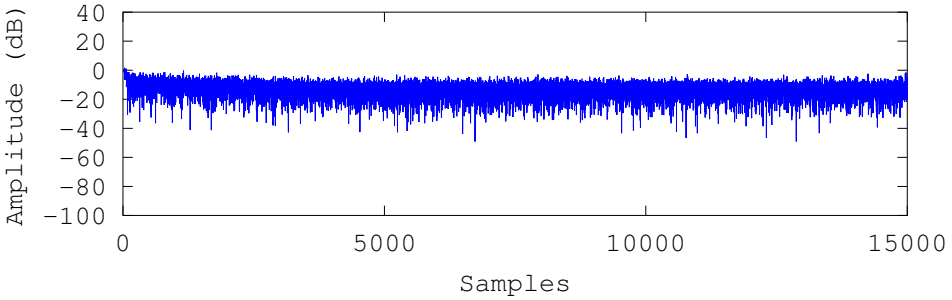
(a) Model input.



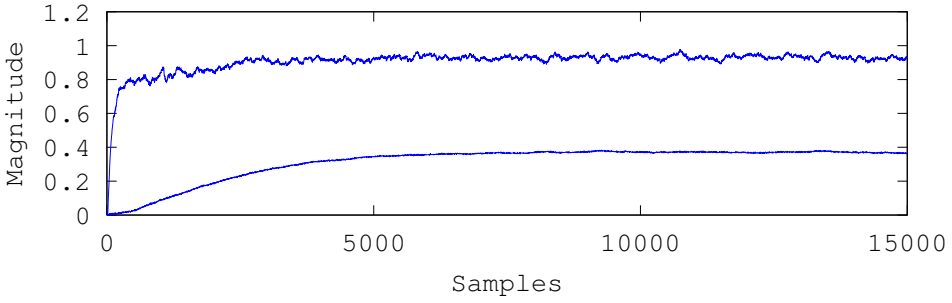
(b) Model output.



(c) HDL output.

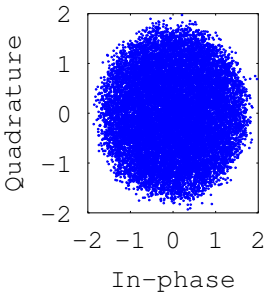


(d) Instantaneous error.

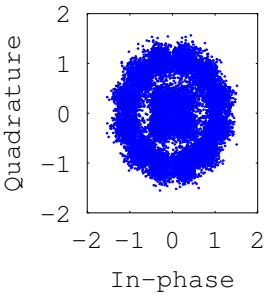


(e) Central tap & cancelled ISI.

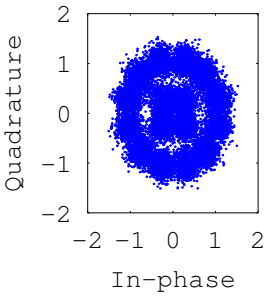
Figure 7.32: 8-PSK.



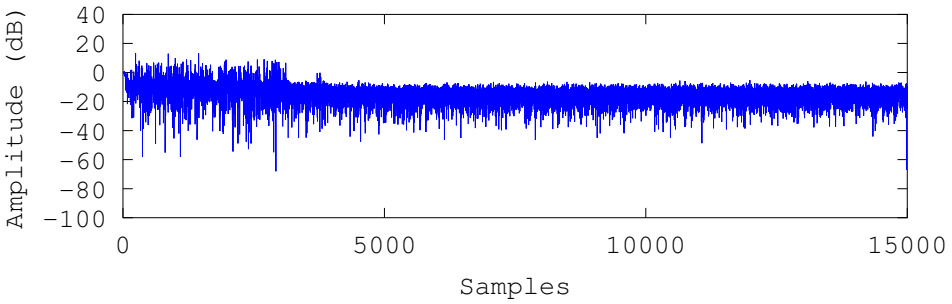
(a) Model input.



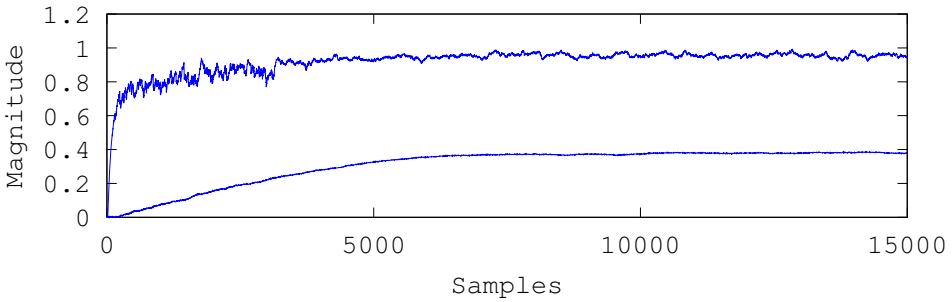
(b) Model output.



(c) HDL output.

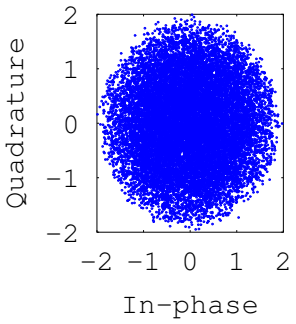


(d) Instantaneous Error.

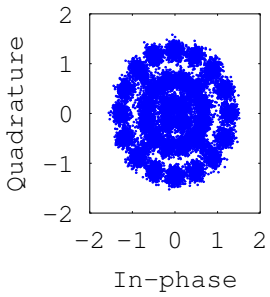


(e) Central tap & cancelled ISI

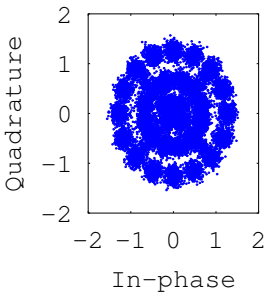
Figure 7.33: 16-APSK



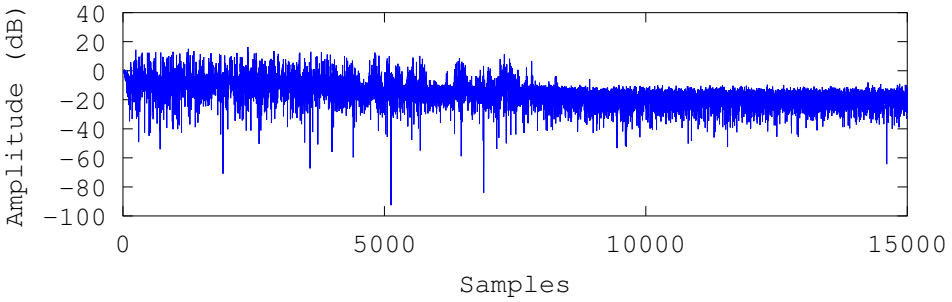
(a) Model input.



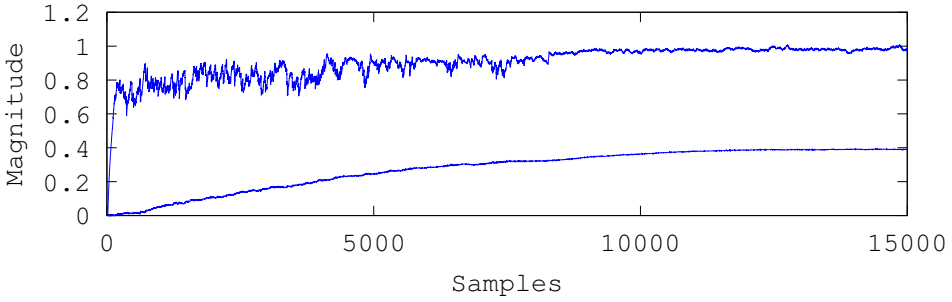
(b) Model output.



(c) HDL output.



(d) Instantaneous error.



(e) Central tap & cancelled ISI.

Figure 7.34: 32-APSK.

7.3.7.2.2 TWTA Nonlinear Effect

In satellite systems, the effects caused by the nonlinearity of the TWTA and

filter characteristics of an input de-multiplexer (IMUX) and an output multiplexer (OMUX) - which cause linear distortion [106] - are very critical to the system performance [106] [126] and shall be tackled. A traditional way to deal with those effects is by means of pre-distortion techniques. Nevertheless, in [126], the authors have shown the potential of using a nonlinear Adaptive Equalizer, such as the DFE, to improve the receiver performance in the presence of those impairments. In this section, the performance enhancements the proposed Adaptive Equalizer can bring to a DVB-S2 receiver in the presence of the nonlinear distortion, caused by the TWTA, are evaluate by means of simulations. In addition, the BER performance improvements for the combination of Land-Mobile-Satelite-Channesl (LMSC) and TWTA are also evaluated. It is worth to mention that the proposed equalizer can be easily extended to cope with DVB-S2X [84]. Despite further simulations shall be done, e.g. to adjust the number of taps of the FB and FF filters, its is clear the advantage of using such approach.

In this section we present several BER vs. SNR curve, obtained through simulations using Octave and discuss the results. The IBO and OBO parameters used to obtain the simulations results shown in Figures 7.35 to 7.38 are presented in Table 7.6. The channel used to obtain the simulations results presented in Figures 7.37 and 7.38 is shown in Table 7.7. The Saleh model's IBO and OBO parameters, for moderate and severe nonlinear distortion presented in Despite *low* nonlinear distortion was not used in the work presented in this thesis, it is depicted in Table 7.6 for a didactic sake.

Table 7.6: Input Back Off (IBO) and Output Back Off (OBO) parameters for the Saleh Model.

LOW	MODERATE	SEVERE
IBO		
-21.5957	1.40433	7.40433
OBO		
32.91183	9.91183	3.91183

Table 7.7: Taps of the LMSC used for simulations.

LMSC TAPS
0.4577+0.0282i
0.1537+0.2213i
0.3578+0.4211i
0.2904+0.0980i
0.1966-0.2084i
-0.1530-0.2537i

To obtain the curves presented in this section, 3 parameters were varied in a combined fashion: 1) the use or not of CMA; 2) the use of moderate or severe nonlinear distortion and 3) the use or not of ISI (in the case of LMSC). During the tests where CMA is not used, if the central tap target value is not reached, the gains update process is frozen until the DFE goes into supervised training mode, either by using PLHEADER or embedded pilots.

It can be realized from the inspection of the BER curves presented in Figures 7.35 to 7.38 that, with exception of QPSK modulation and for moderate nonlinearity (see Figure 7.35), all results present significant improvements in the performance when adopting the proposed DFE, in both case, i.e. for TWTA and TWTA combined with ISI. The case of QPSK with low nonlinearity is still under investigation to discover the reason of the bad performance. Nevertheless, this does not disqualify the good results presented for the other cases.

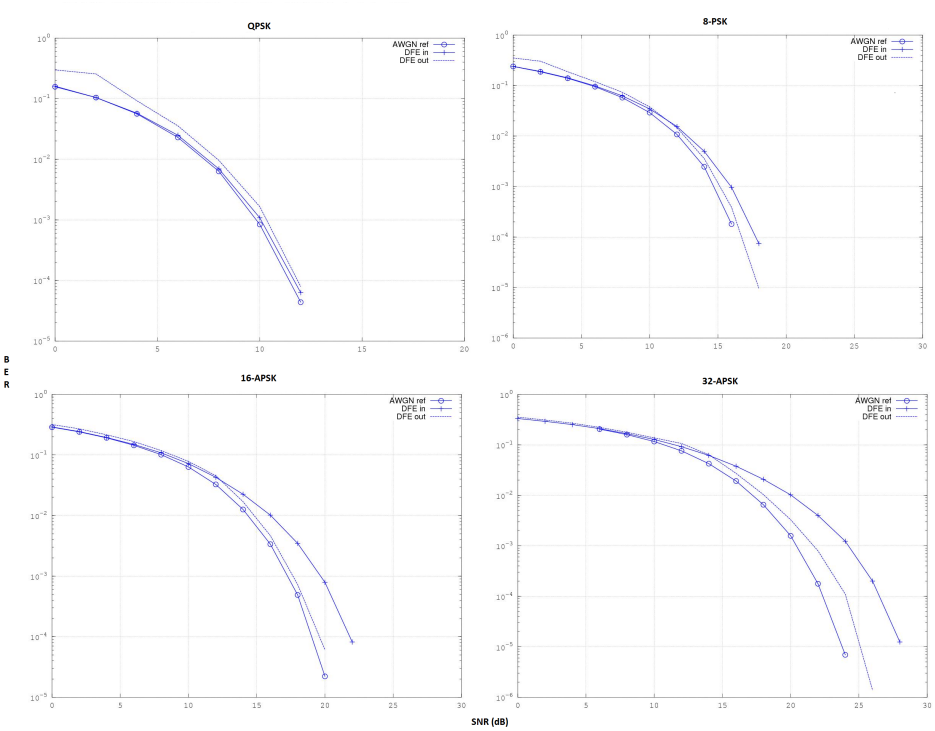


Figure 7.35: BER performance of DVB-S2 for all modulations with Moderate nonlinearity and CMA off: Saleh Model, IBO: 1.40433 and OBO: 9.91183.

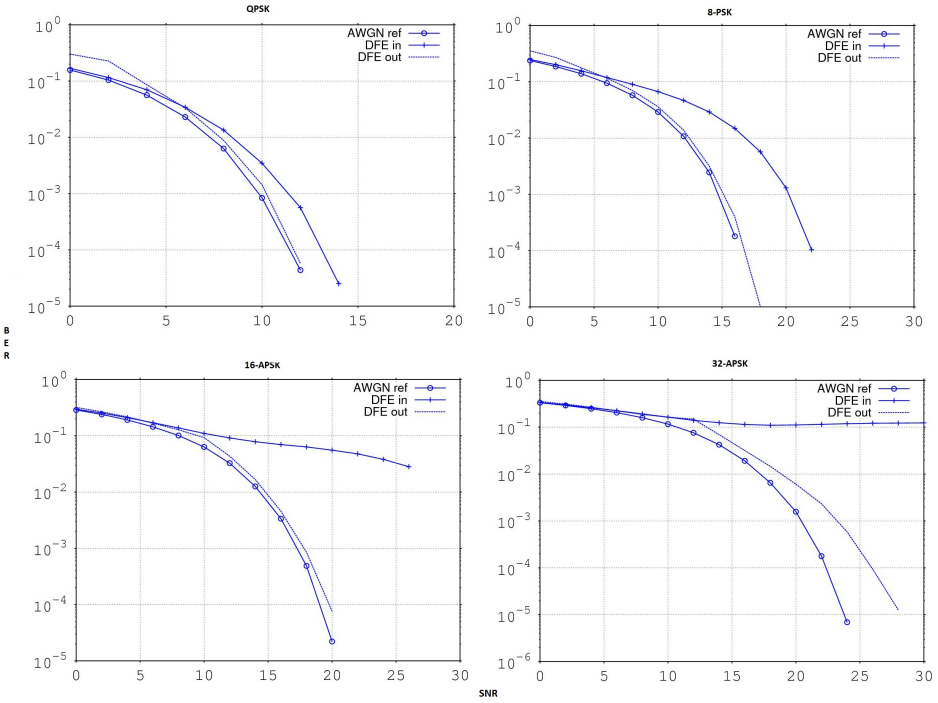


Figure 7.36: BER performance of DVB-S2 for all modulations with Severe nonlinearity and CMA off: Saleh Model, IBO: 7.40433 and OBO: 3.91183.

7.3.8 SNR Estimator

SNR estimation is mainly used within the DVB-S2 receiver for two main purposes. The first is to generate the soft information by the Soft Demapper and the second is for debugging. With the estimated value in hands, it is possible to access the SNR the receiver is facing. A third application for the SNR estimator, in this thesis, is to measure the SNR and generate the soft information for a debugging platform created to measure the BER performance of the implemented FEC subsystem in FPGA. Further details on that platform can be found within Section 7.3.10.4.

The architecture of the proposed SNR Estimator is depicted in Figure 7.39. As can be seen there, the estimator is based on the CORDIC algorithm, which is used to compute the module and the linear Signal-to-Noise-Ratio by means of a division. It is worth mentioning that as in the case of the Coarse Phase Estimator, the input complex symbols are rotated to the first quadrant in order to avoid the need for storing the reference embedded pilots.

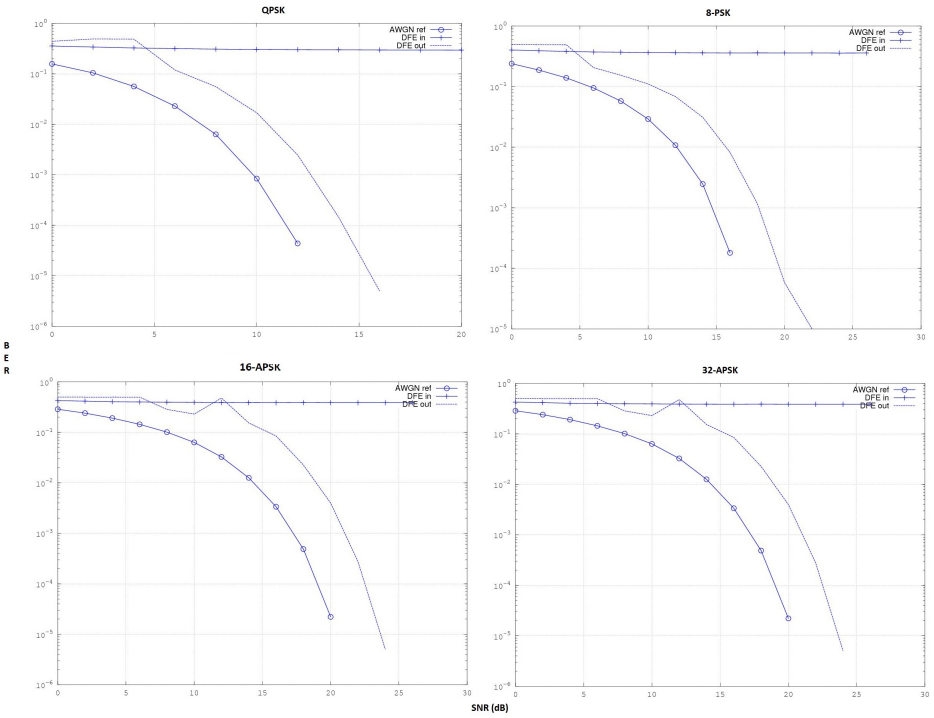


Figure 7.37: BER performance of DVB-S2 for all modulations with Moderate non-linearity plus ISI and CMA on: Saleh Model, IBO: 1.40433 and OBO: 9.91183.

The estimator works as follows:

1. The corrupted received complex pilots symbols - that originally assume the values $\pm\sqrt{2}/2 \pm j \times \sqrt{2}/2$ - are rotated to the first quadrant;
2. The rotated I and Q components of the received signal are subtracted from the first-quadrant-only pilot, $\sqrt{2}/2 + j \times \sqrt{2}/2$ (that has power equal to 1), in order to obtain the complex noise. Then, the noise power is computed and averaged over a configurable interval to avoid abrupt variations of the resulting estimated noise power;
3. To calculate the inversion of the noise power *(i.e.* the SNR), the CORDIC is used. Nevertheless, it has a limitation regarding its output range. To overcome this limitation the numerator is scaled, by means of an appropriate number of shifts to right, so that the result of the inversion can be within the valid range;
4. Finally, the output of the CORDIC is shifted by the same number of shifts, but this time to the left, resulting in the inverse of the noise variance;

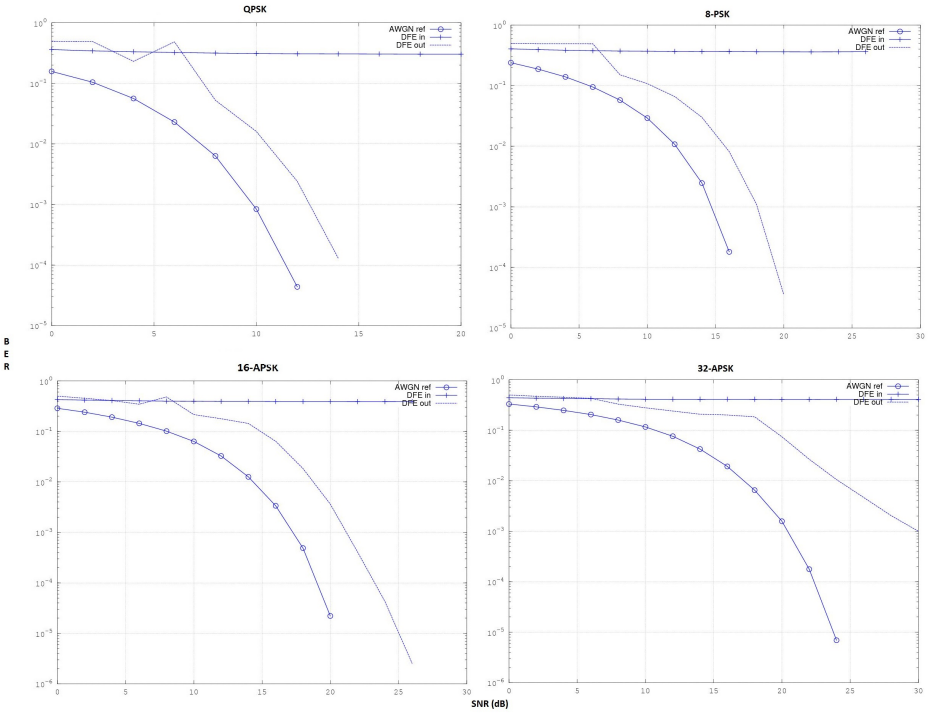


Figure 7.38: BER performance of DVB-S2 for all modulations with Severe nonlinearity plus ISI and CMA on: Saleh Model, IBO: 1.40433 and OBO: 9.91183.

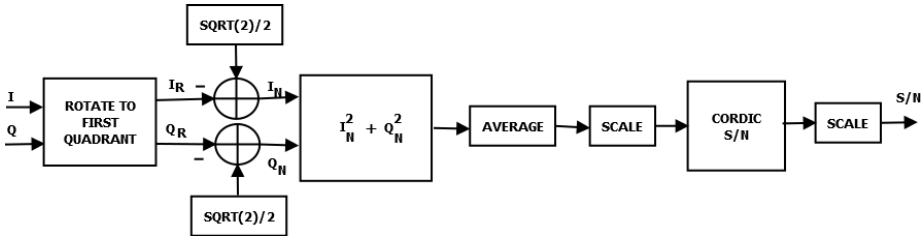


Figure 7.39: Architecture of the SNR and Soft Information Estimator.

7.3.9 Soft and Hard-Decision Demappers

In this section I describe the proposed Hard-Decision and Soft Demappers. The Soft Demapper (SD) is essential for the DVB-S2 receiver to achieve the target performance defined in [77]. On the other hand, the proposed Hard-Decision Demapper (HD) is

used to obtain the hard symbols used in the Decision-Directed mode of the Adaptive Equalizer, as well as in the SNR estimator. Another possible application of HD is for fine frequency estimation[1].

7.3.9.1 Hard-Decision Demapper

The main objective of the HD is to convert complex baseband symbols corrupted by noise and the channel into bit streams. It adopts different methods for each modulation. Nevertheless, all the methods basically consist in checking whether the received complex symbol lies within a specific decision region around a given constellation point. Figure 7.40 presents three examples of those regions, for 8PSK, 16APSK and 32APSK. For simplicity's sake, the DVB-S2 baseband modulation constellations, presented in [77] and Section 3.2, are not reproduced here.

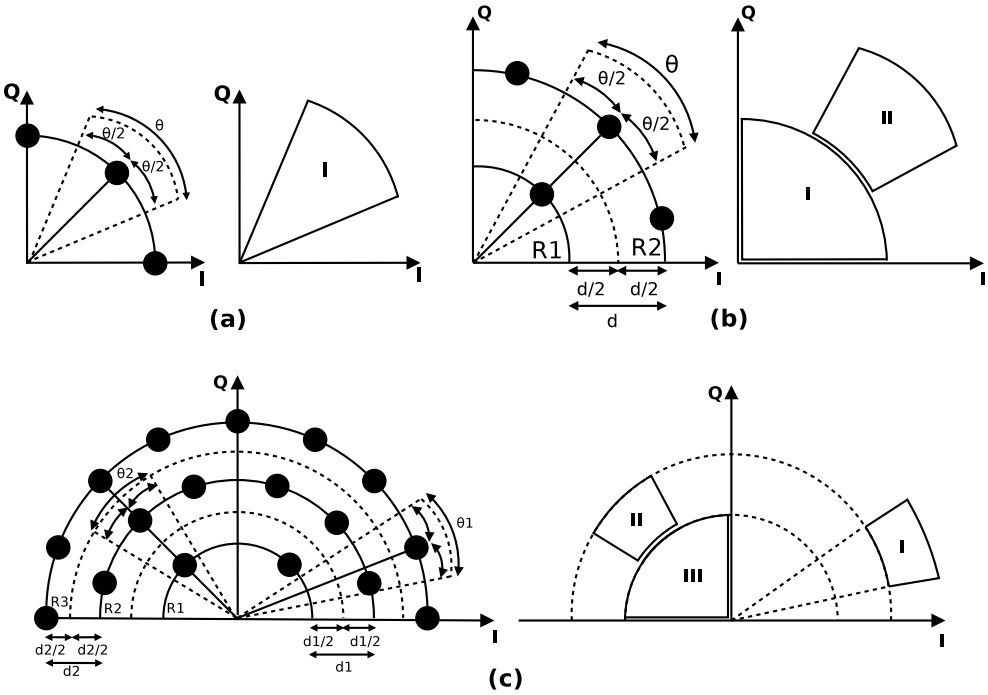


Figure 7.40: Decision regions of (a) 8PSK, (b) 16APSK and (c) 32APSK.

For QPSK modulation, the signs (positive or negative) of in-phase (I) and quadrature (Q) components of the received complex signals are analyzed, obtaining their respective quadrants, so, hard-bits or hard-symbols can be obtained.

[1]This was not implemented in this thesis.

Given three adjacent symbols, the decision regions of 8PSK are defined as the region delimited by two bisectors of each adjacent pair of symbols, as can be seen in Figure 7.40(a).

In the case of 16APSK, the decision regions are defined for two cases. The first takes into account the bisectors of two pairs of adjacent symbols placed in the outer constellation radius, R2, as well as an imaginary circle equally distant of R2 and the inner constellation circle, R1. An example of that decision region is area II in Figure 7.40(b). For the second case, the decision regions are defined for the received symbols that have module smaller than the imaginary circle between R1 and R2. In this case, the hard-decisions are made by checking in which quadrant the received symbol lies in. Area I in Figure 7.40(b) is an example of that.

Based on the previous explanations, the decision regions can be directly derived for the 32APSK. Nevertheless, it is worth mentioning that this modulation has three radii in its constellation, as can be seen in Figure 7.40(c).

7.3.9.2 Soft Demapper (SD)

Given a received symbol corrupted by the channel, which transports one or more bits belonging to a specific modulation (e.g. QPSK, 8PSK, 16-APSK, etc), for each of the bit belonging to a constellation point, it is possible to compute the soft information using a Soft Demapper. On the contrary of hard decision operation, the soft information provides a measure o uncertainty on the bit under evaluation. After computation of the soft information for all bits belonging to a given received symbol, the soft information is quantized with "'N"' bits , e.g. for QPSK symbols, which transports 2 bits per symbol, hard decision provides 2 hard bits and soft demapping provides $N \times 2$ bits. The soft-decision is essential for the DVB-S2 Forward Error Correction (FEC) system to achieve quasi-error-free performance near Shannon-limit.

A widespread soft-decision computation method is the so called Log-Likelihood Rate (LLR). However, this method uses exponential and logarithmic operations that increase hardware complexity to obtain soft-bits. To reduce the complexity, various SD methods have been proposed, being MAX (see [135]) the most used method. In the algorithm for 8PSK Soft-Demapper, some modifications in the original method were necessary. The QPSK algorithm was also modified to be used in the 8PSK algorithm. Even without using exponential and logarithm operations, its complexity is considerably large regarding the one adopted in the work presented in this thesis.

The implementation proposed in this thesis is based on different reference methods for each modulation, which were derived from MAX method seeking a lower hardware complexity. QPSK SD uses the method presented in [136]. 8PSK SD is a slightly modified version of the solution proposed in [136]. 16APSK SD is based on the decision regions presented in [137], but with other LLR equations derived in the work

presented in this thesis. Until this moment, 32APSK SD has not been implemented.

Following [136] notation, the soft information for QPSK and 8PSK can be computed using I&Q component values, the estimated noise variance (σ^2), and a set of constants, as shown in Equation 7.18. For higher order modulations, LLRs corresponding to the most significant bits are obtained by using other methods.

$$L(b) = (K_1 I + K_2 Q)/\sigma^2 \tag{7.18}$$

For b_0 and b_1 of QPSK modulation, the constant pair (K_2, K_1) are respectively set as ($\sqrt{2}$, 0) and (0, $\sqrt{2}$).

8PSK SD case provides three soft-bits, b_0, b_1 and b_2. For b_0 and b_1, the input symbols are rotated by $-\pi/8$, generating I_r and Q_r. They are used in Equation 7.18 with the same constants of QPSK SD. For b_2, I and Q are used in the Equation 7.18, and constants are selected according to Table 7.8.

Table 7.8: Coefficients for Equation 7.18 for 8-PSK

Decision Region	K_1	K_2
$I \geq 0, Q \geq 0$	0.707	-0.293
$I < 0, Q \geq 0$	-0.293	-0.707
$I < 0, Q < 0$	-0.707	0.293
$I \geq 0, Q < 0$	0.293	0.707

Finally, for 16APSK modulation, the third and the fourth soft-bits are calculated according to the equations presented in Table 7.9. The complex symbol components, depending on θ (I_θ and Q_θ), are the remapped symbol components, i.e. $|I|$ and $|Q|$, rotated by $\theta_{ref} - \theta$. Where $|\cdot|$, is the module operation. It is mentioning that the input symbols are remapped to the first quadrant to take advantage of the symmetry, and all the reference values are defined by the boundaries of the decision regions in this quadrant.

7.3.9.3 Proposed Architecture

7.3.9.3.1 Hard Demapper

The HD is implemented as a set of comparisons depending on the decision regions, which are defined by the modulation type. Figure 7.41 shows the HD architecture.

Table 7.9: Equations for bits 2 and 3 of 16-APSK

Decision region	Equation
$\boldsymbol{R < R_{ref}, \theta \geq \theta_{ref}}$	$(R - R_{ref})/\sigma^2$
$\boldsymbol{R < R_{ref}, \theta < \theta_{ref}}$	$-\max(\lvert\,\lvert I\rvert - I_{ref}\,\rvert, \lvert\,\lvert Q\rvert - Q_{ref}\,\rvert)/\sigma^2$
$\boldsymbol{R \geq R_{ref}, \theta < \theta_{ref}}$	$-\max(\lvert\,\lvert I\rvert - I_{\theta}\,\rvert, \lvert\,\lvert Q\rvert - Q_{\theta}\,\rvert)/\sigma^2$
$\boldsymbol{R \geq R_{ref}, \theta \geq \theta_{ref}}$	$\min(R - R_{ref}, \max(\lvert\,\lvert I\rvert - I_{\theta}\,\rvert, \lvert\,\lvert Q\rvert - Q_{\theta}\,\rvert))/\sigma^2$

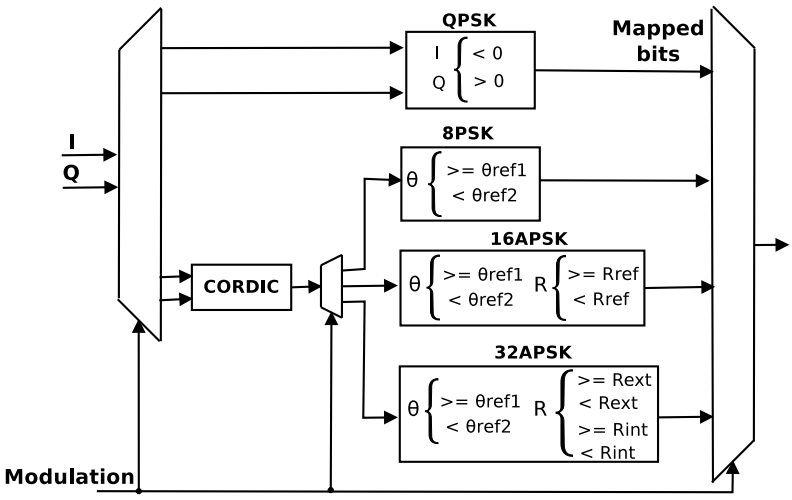


Figure 7.41: Hard Demapper architecture.

When working in QPSK mode, the signs of the I&Q components define the symbol quadrant. Each quadrant is related to a specific pair of hard-bits. The analysis, in 8PSK modulation, is done with the angle θ calculated by the CORDIC. The computed angle is compared to a group of reference angle intervals, related to each constellation symbols, so the three hard-bits can be defined. In the 16APSK mode, besides the angle, the radius, R, of the symbol, is also computed by the CORDIC and compared to stored reference values. With this information in hands it is possible to define the correct hard-bits. Similar to 16APSK, the 32APSK HD analyzes radius and the angle calculated by the CORDIC. The difference is the existence of two groups of reference angles, θ_1 and θ_2, based on internal and external radius, R_1 and R_2.

The CORDIC is instantiated in circular coordinate and vectoring mode operation.

7.3.9.3.2 Soft Demapper

The SD instantiates a sub-block for each mode to obtain the LLRs according to the formulas and methods presented in Section 7.3.9.2. The structure called LLR Calc (LLR Calculator), formed by a sum and three multipliers implements the Equation 7.18.

The microarchitecture of the QPSK SD is illustrated in Figure 7.42(a) which also shows that each LLR Calc adopts only one K, due to simplifications in Equation 7.18. In this way, one multiplier and the sum are discarded. The value of the remaining constant K is $\sqrt{2}$, as already mentioned in Section 7.3.9.2.

Figure 7.42(b) presents the microarchitecture of the 8PSK SD. This structure brings two new elements: the Rotate Symbol and a LUT. The LUT stores K_1 and K_2, both applied to compute the LLR of b_2, and the Rotate Symbol performs a rotation in order to reuse the QPSK microarchitecture for b_0 ad b_1 . K_1 and K_2 values are in Table 7.8.

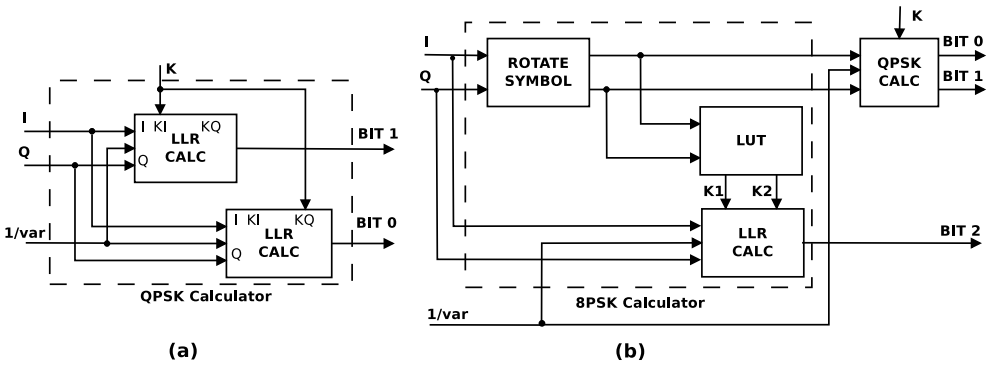


Figure 7.42: Microarchitecture of (a) QPSK and (b) 8PSK Soft Demappers.

The soft-bits, b_1 and b_0, of the 16APSK SD, are obtained using QPSK microarchitecture. The other soft-bits, *i.e.* b_2 and b_3, are calculated using the 16APSK Bit Calc (16APSK Bit Calculator) block, displayed in Figure 7.43. The Absolute Symbol block provides the remapped symbol mentioned in Section 7.3.9.2. As in Section 7.3.9.3.1, the CORDIC provides θ and R, which are used to define the decision regions where the received symbol lays in. Next, it is chosen between (I_{ref}, Q_{ref}) and (I_θ, Q_θ) components, which pair is going to be used to compute the chessboard distance (also know as Chebyshev distance) from the remapped symbol. I_θ and Q_θ are provided by the Symbol Adjustment. Also, after definition of the decision region, the 16APSK LLR Calc (16APSK LLR Calculator) performs one of the Table 7.9 operations. The difference between the structures used to compute b_2 and b_3 lies in the adopted reference

values, that change due to symmetry. The Absolute Symbol, the Symbol Adjustment and the CORDIC were reused in order to reduce hardware complexity.

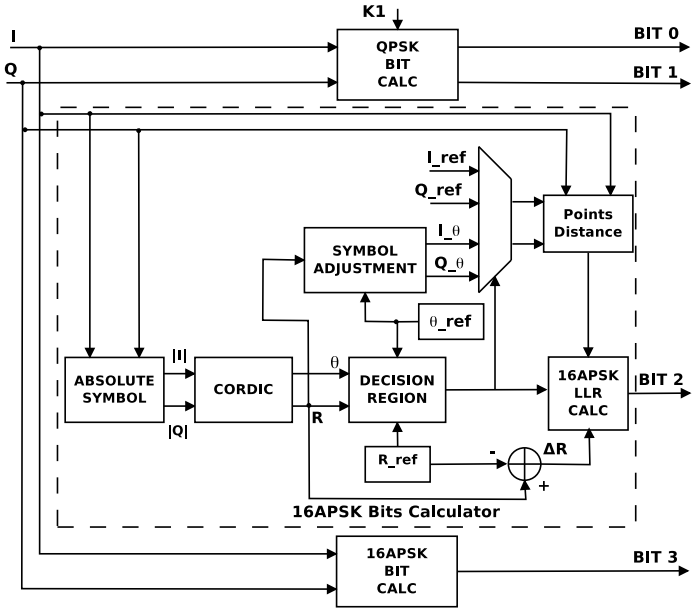


Figure 7.43: Microarchitecture of 16PSK Soft Demapper.

7.3.10 FEC Subsystem and BER Measurement Platform

The main function of the FEC decoding subsystem in the DVB-S2 receiver is restoring the useful bits that were encoded, transmitted and corrupted by the satellite channel. This is achieved by three concatenated blocks, 1) Deinterleaver, 2) LDPC decoder and 3) BCH decoder, which are shown in the Figure 7.44.

In this section I describe the architecture of those three elements, the algorithms used for LDPC and BCH decoding, as well as the proposed platform used to evaluate the BER performance of the entire FEC subsystem without the influences of impairments such as timing and frequency errors.

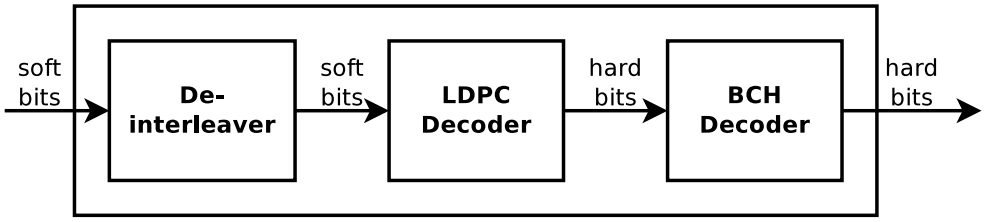


Figure 7.44: FEC decoding subsystem block diagram.

Although the output of the FEC subsystem in the transmitter are hard-bits, *i.e.* bits that assume '0' or '1' values, the input to the corresponding subsystem in the receiver are soft-bits, i.e. integer number representing probabilities that are associated to the bits inside the frame. Those probabilities represent how likely a given bit is '0' or '1' and that kind of information is mandatory for the LDPC Decoder.

The decoding process is executed in a frame-by-frame basis. The received symbols are soft-demapped and the outputted soft-bits are rearranged by the Deinterleaver according to the method presented in [77] and Section 3.2.1.3.3. The rearranged frame follows the format presented in Figure 7.45. The BBFRAME section contains the useful bits while the LDPCFEC and the BCHFEC sections contain the parity bits, that were added by the encoders before transmission.

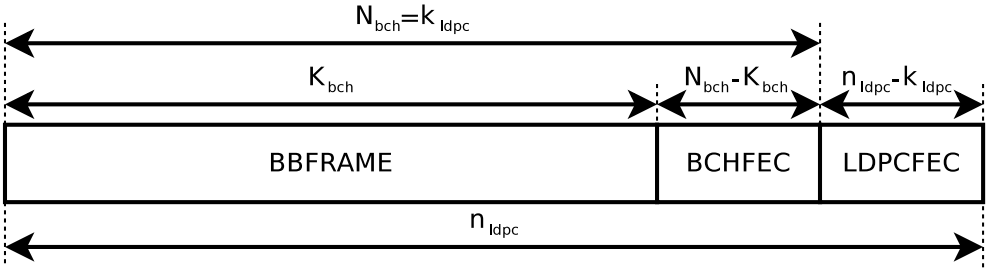


Figure 7.45: DVB-S2 FECFRAME.

The LDPC decoder receives the rearranged FECFRAME, corrects most of the erroneous bits and discards the LDPCFEC. Next, the BCH decoder corrects the remaining errors (up to 12 according to the frame type and code rate) and removes the BCHFEC. Finally, the decoded useful data, BBFRAME, is forwarded to the following modules of the DVB-S2 receiver.

The Deinterleaver, the LDPC and BCH decoders will be explained in the following subsections.

7.3.10.1 Deinterleaver

The interleaving scheme is frequently used in digital communication to improve the performance of FEC coding. Its main objective is spreading the errors that often occur in burst, so that those errors present a more uniform distribution in the decoder input [138], as illustrated in Figure 7.46.

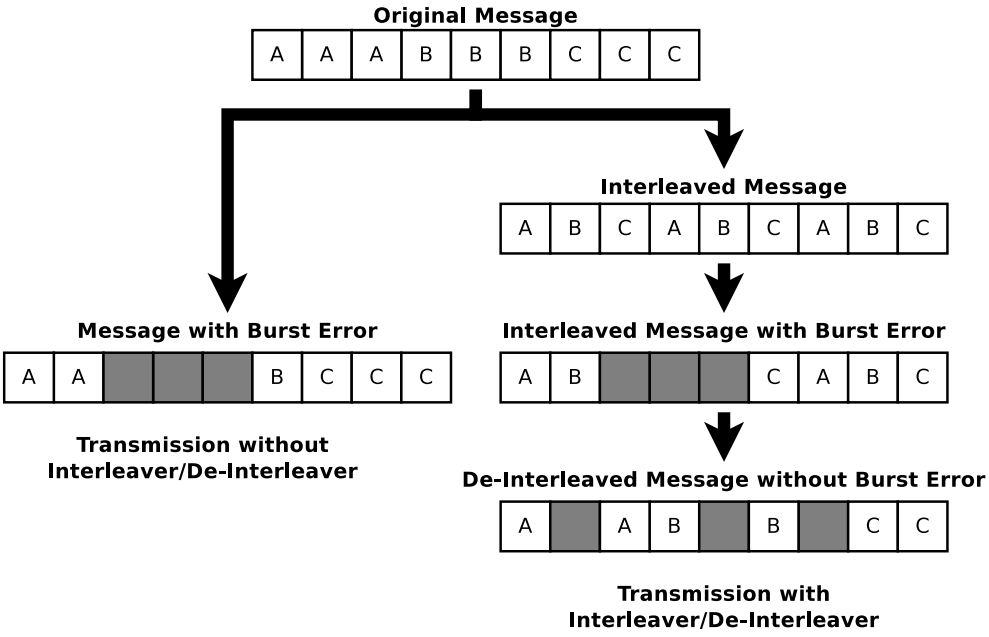


Figure 7.46: Deinterleaver error spreading.

The Deinterleaver block rearranges the data that has been arranged in a non-contiguous way into its original sequence. In order to achieve this, the input data must be written and read in a specific way, so the storage based on a structure of c columns and r rows is made. In the deinterleaving process, each received bit (or the associated soft-bit) is written row-wise and read out column-wise, reversing the interleaving process, as shown in Figure 7.47.

The number of rows and columns of the adopted structure depends on the DVB-S2 modulation format. The whole process is not applied to the QPSK modulation due to the robustness of this modulation. Further details on DVB-S2 interleaving can be found in [77] and Section 3.2.1.3.3.

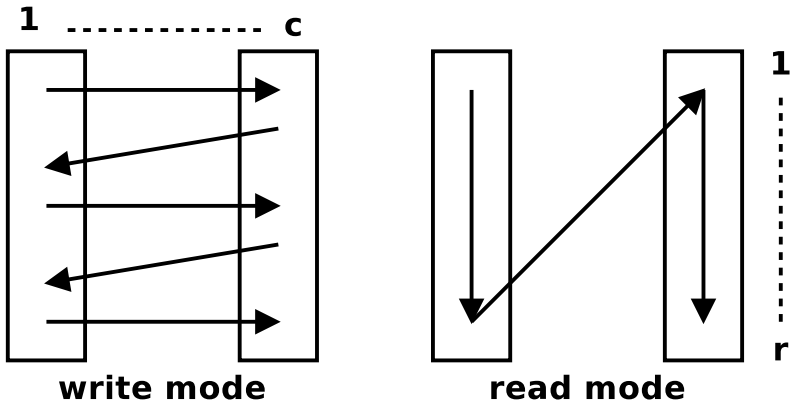


Figure 7.47: Deinterleaving Scheme.

7.3.10.2 LDPC Decoder

The LDPC Decoder is responsible for decoding a sequence of bits inside a frame, based on the soft-bits that carry channel information. Those soft-bits are the quantized version of the Log Likelihood Ratios $LLR(x_n) = log\frac{P(x_n=0|y_n)}{P(x_n=1|y_n)}$, which express how likely a given bit is '0' or '1', so that the more positive the LLR is, more likely the bit is '0', and in the same way, the more negative the LLR is, more likely the bit is '1'.

For decoding LDPC codes, the *Minimum-Sum* algorithm, a simplification of the *Belief-Propagation (BP)* based on LLRs [139], was implemented with a *layered decoding* [140] approach. The basic idea of this iterative algorithm is exchanging messages between *Variable Nodes (VNs)*, corresponding to the received bits/symbols, and *Check Nodes (CNs)*, corresponding to the parity-check equations, in order to calculate new probabilities.

In layered decoding, a VN is represented by its *Soft-output (SO)* value, that is nothing else than the LLR, which is frequently updated during the iterations, and one iteration is split into sub-iterations, one sub-iteration for each layer made of one or several CNs. Assuming that L_n is the initial LLR of bit n, Q_n is the LLR (or SO) of bit n considering all the messages exchanged so far, Q_{nm} is the message sent by VN n to CN m, R_{mn} is the message sent by CN m to VN n, and $N(m) \setminus n$ is the set of VN's connected to CN m excluding n, the steps of the layered decoding algorithm are:

1. Initialization: $Q_n = L_n$, $R_{mn} = 0$
2. VN message calculation: $Q_{nm} = Q_n - R_{mn}^{old}$
3. CN update: $R_{mn}^{new} = f(Q_{n'm}),\ for\ n' \in N(m) \setminus n$

4. SO update: $Q_n^{new} = Q_{nm} + R_{mn}^{new}$
5. Decision: if all the VNs satisfy the check equations, a codeword was detected (so finalize algorithm); if the maximum number of iterations was reached, the decoding failed (so finalize algorithm); otherwise go to step 2 for the next iteration.

Using the Minimum-Sum derivations, the function $f(.)$, corresponding to the CN update step, can be expressed in two ways, with α and β being configurable parameters:

- Offset Min-Sum:

$$R_{mn} \approx \left(\prod_{n'} sign(Q_{n'm})\right) \max\left(\min_{n'}|Q_{n'm}| - \beta, 0\right)$$

- Normalized Min-Sum:

$$R_{mn} \approx \left(\prod_{n'} sign(Q_{n'm})\right) \left(\min_{n'}|Q_{n'm}|\right) / \alpha.$$

For connecting the VNs to the CNs, a parity-check matrix is defined, and the same matrix must be used by both encoder and decoder. For each one of the 21 frame configurations (considering the combination of 11 code rates and 2 frame lengths), the DVB-S2 standard defines a different coding scheme, which makes the LDPC feature the most complex part of a DVB-S2 FEC subsystem [141]. Although their contents are different, the parity-check matrices follow a well defined structure based on an address table and a code rate dependent constant, but they can be rearranged so that they are composed of shifted Identity sub-matrices, which makes the parallelization of the decoding algorithm easier. Given that the parity-check matrix is defined as $H = [A \mid B]$, with A connecting the *parity-check equations* to the *information bits*, and B connecting the equations to the *parity bits*, the rearranged H matrices in DVB-S2 are exemplified by Figure 7.48.

Each shifted Identity sub-matrix connects a *Check Node Group (CNG)* to a *Variable Node Group (VNG)*, and each group is composed of $P = 360$ nodes. The B side of H presents the same format for all frame lengths and code rates, while the A side varies for each configuration, as well as the number of CNGs (q) and information VNGs (i). The parallelism of a hardware-implemented decoder can be achieved by processing all the CNs of a given CNG at the same time. One BP *iteration* is split into q *sub-iterations*, one for each CNG (layer), and a sub-iteration consists in updating all the VNs connected to the CNs inside current CNG.

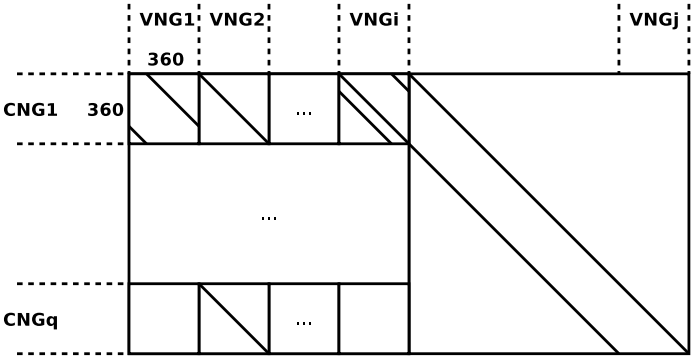


Figure 7.48: Structure of the rearranged H matrix.

The structure of the standard matrices allows a parallelism of 360, i.e. 360 logic elements executing the message update calculations at the same time, without interfering in the results of each other. The LDPC decoder implemented in this work supports two levels of parallelism, 180 and 360, which can be chosen according to the hardware resources available for the project, concerning both logic and memory. Figure 7.49 presents the LDPC Decoder diagram, with emphasis on the memory banks and on the logic elements that parallelize the decoding algorithm.

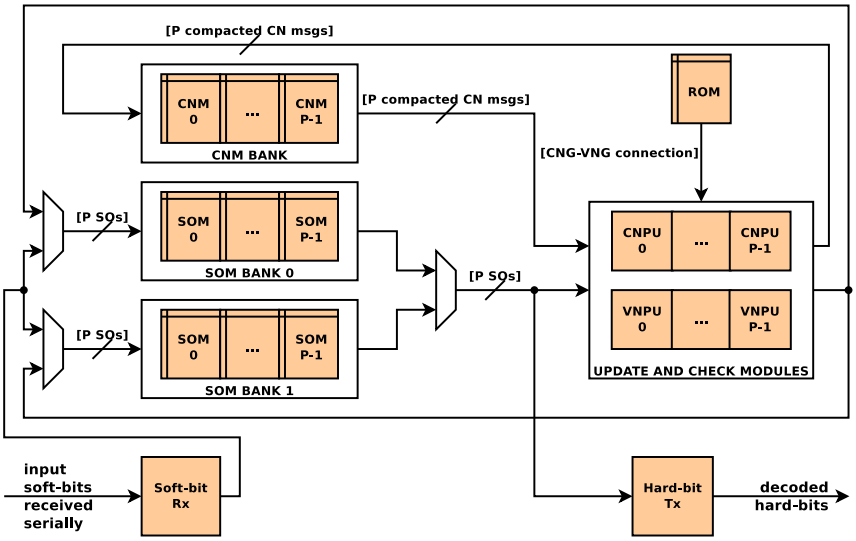


Figure 7.49: LDPC Decoder data flow.

The update steps of layered decoding are executed by the *Check Node* and *Variable Node Processing Units* (*CNPUs* and *VNPUs*), one CNPU-VNPU pair for each CN inside the CNG that is being updated. In addition to those processing units, there are

other logic elements responsible for receiving the initial LLRs, shifting the messages in each sub-iteration according to the H matrix configuration, controlling memory access, checking the parity at the end of each iteration and transmitting the decoded bits. For the parity-check matrices, one *ROM* block is used, and for the BP messages, three RAM banks, two for the SOs (*SOM0* and *SOM1*) and one for the CN messages (*CNM*). The two SO Memory banks are alternately used to receive the initial LLRs of next frame and to update the SOs of current one, so that while a given frame is being decoded by the update and check modules, the LLRs of the next one can be buffered by the *Soft-bit Rx* sub-block. More details about the architecture of the implemented LDPC Decoder can be found in [21].

7.3.10.3 BCH Decoder

The BCH-Decoder uses the BCHFEC bits for recovering the original BBFRAME, with a t error correction capacity of up to 8, 10 or 12 errors according to the code-rate and frame type. This is accomplished by following Algorithm 1.

Algorithm 1 BCH decodification

Input: R: FECFRAME without LDPCFEC bits.
Output: C: Recovered BBFRAME .
1: C = BBFRAME part of R.
2: Calculate syndromes S.
3: **if** $S \neq 0$ **then**
4: Determine the Error Locator Polynomial σ.
5: Find the roots of σ.
6: Swap the bits of C that correspond to the Galois inverse of the root.
7: **end if**
8: **return** C

Based on Algorithm 1, the BCH decoder was designed as illustrated in Figure 7.50, where the *BCH Control Unit* contains a finite state machine to orchestrate each step of the decoding process and a *Memory Buffer* stores the FECFRAME until the *Polynomial Roots Finder* is ready to signalize which of the bits of the frame have been swapped.

As mentioned in [142], the calculation of syndromes (S) can be done considering the received frame as a polynomial $R(x)$, and substituting the first $2t$ α elements of the Galois fields established for normal and short FECFRAMEs in [77] (i.e. GF(2^{16}) and GF(2^{14}), respectively), as expressed in $S_i = R(\alpha^i)$, *where* $i \in \{1, \cdots, 2t\}$. For the *Syndromes Calculator*, the design detailed in [22] was implemented. The *Key Equation Solver* is commonly based on the Berlekamp-Massey (BM) algorithm, and the Simplified inversion-free BM (SiBM) (Algorithm 2) was adopted, with the corrections presented in [111]. This variation is optimized for binary codes and avoids

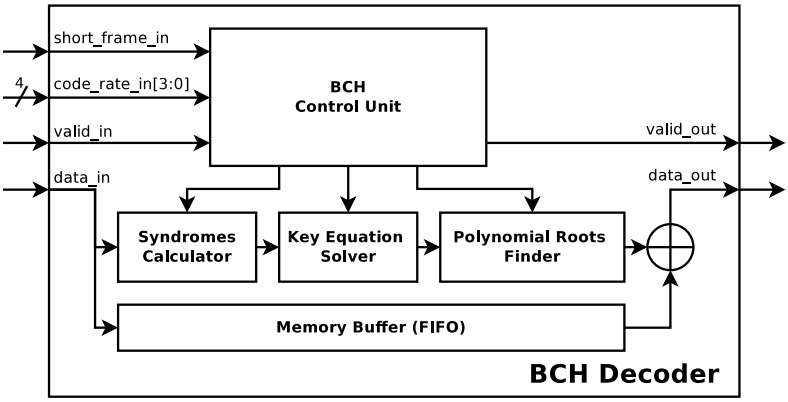


Figure 7.50: Internal architecture of the proposed BCH decoder.

the complexity of GF invertions [143]. The *Polynomial Roots Finder* is based on the Chien Search algorithm, but since the DVB-S2 uses 21 shortened BCH codes, the variant from [112] was used as illustrated in Figure 7.51, where $\beta = 2^{K_{bch}} - N_{bch} - 1$.

Algorithm 2 Simplified inverse-free Berlekamp-Massey

Input: t: Error correction capacity.
S: $2t$ Syndromes calculated for the received FECFRAME.
Output: σ: Error Locator Polynomial.

$\kappa(0) = 0, \gamma(0) = 1$
$\delta_i(0) = \theta_i(0) = S_i \quad \forall\ i \in \{0, 1, 2, ..., 2t-2\}$
$\delta_{2t-1}(0) = \theta_{2t-1}(0) = 0,\ \delta_{2t}(0) = \theta_{2t}(0) = 1$
for $r \leftarrow 0$ **to** $t-1$ **do**
  $\delta_i(r+1) = \gamma(r)\delta_{i+2}(r) + \delta_0(r)\theta_{i+1}(r)$
    $\forall\ i \in \{0, 1, 2, ..., 2t\}$
    where $\delta_{2t+1}(r) = \delta_{2t+2}(r) = \theta_{2t+1}(r) = 0$
  if $\delta_0(r) \neq 0$ **and** $\kappa(r) \geq 0$ **then**
    $\theta_i(r+1) = \delta_{i+1}(r)$
      $\forall\ i \in \{0, 1, 2, ..., 2t\} \setminus \{2t-2-2r,\ 2t-3-2r\}$
    $\gamma(r+1) = \delta_0(r)$
    $\kappa(r+1) = -\kappa(r)$
  else
    $\theta_i(r+1) = \theta_i(r)$
      $\forall\ i \in \{0, 1, 2, ..., 2t\} \setminus \{2t-2-2r,\ 2t-3-2r\}$
    $\gamma(r+1) = \gamma(r)$
    $\kappa(r+1) = \kappa(r) + 1$
  end if
  $\theta_i(r+1) = 0 \quad \forall\ i \in \{2t-2-2r,\ 2t-3-2r\}$
end for
$\sigma_i = \delta_i(t)\ \forall\ i \in \{0, 1, 2, ..., t\}$
return σ

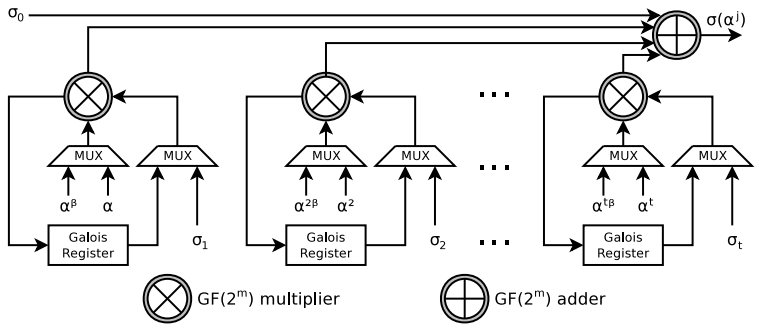


Figure 7.51: Shortened Chien Search.

7.3.10.4 Platform for FEC Subsystem BER Measurement

In order to measure the BER performance of the implemented FEC Subsystem and perform parameters adjustments of the LDPC Decoder (operations that would take a very long time using simulations), a platform for BER Measurement and parameters adjustment is proposed in this thesis. The architecture of that platform is shown in Figure 7.52.

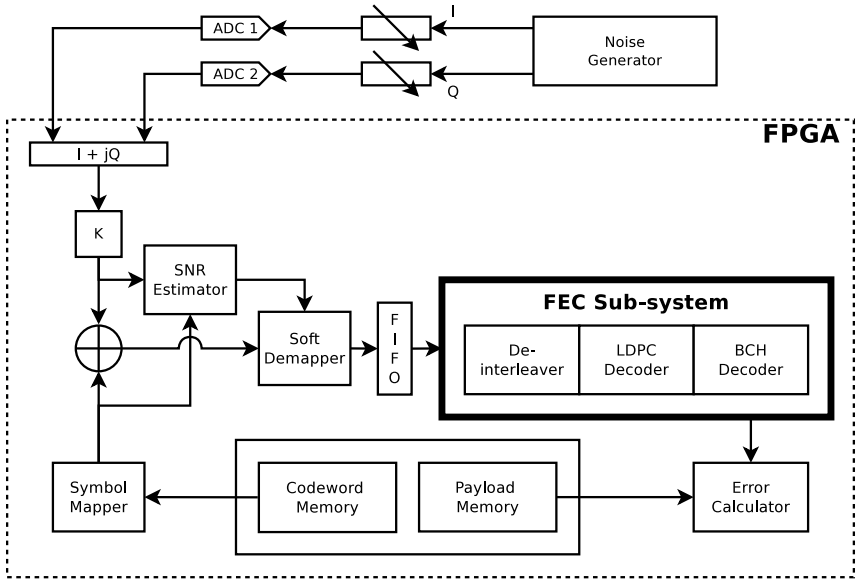


Figure 7.52: BER performance evaluation platform for FEC Subsystem.

The stimulus and output references come from a database that was generated by

a DVB-S2-compliant transmitter model. This database contains the data (in bit or symbol format) processed by each step of the transmission flow.

Inside the FPGA, there are two memory blocks: the Codeword Memory, which contains the constellation points that must be converted into symbols by the Symbol Mapper; and the Payload Memory, which contains the payload (the same mapped bits excluding the parity bits) that must be used as reference for checking the FEC output. The data stored in each memory correspond respectively to the interleaved FECFRAME and the BBFRAME (see Figure 7.45). For further details on DVB-S2 frame structure refer to Section 3.2.

The reason for using the constellation points instead of the mapped symbols, that could also be obtained from the offline database, was that the chosen configuration (Memory+Mapper) would save memory resource, since the symbol representation (in-phase and quadrature components) requires much more bits than a concatenation of bits that varies from 2 (QPSK modulation) to 5 (32APSK).

The error source for the Bit Error Rate (BER) measurement is an external Noise Generator, which provides a complex baseband noise through two separate channels, one for each symbol component. Those components are analog-to-digital converted and then passed to the system as the noise, which is incorporated to the signal by the adder. The target Signal-to-Noise Ratios (SNRs) are adjusted by attenuating or amplifying the amplitude of the generated noise through a combination of external attenuators and internal multiplier.

The SNR values are calculated by the SNR Estimator block. As it can be seen in Figure 7.53, the SNR Estimator has the Power Signal sub-block, which is instantiated twice. The instances are responsible for providing the signal and noise powers to calculate $P_x = |x|^2 = \sqrt{I_x^2 + Q_x^2}^2 = I_x^2 + Q_x^2$. This process is repeated a specific amount of times, accumulating the results and averaging them, in order to avoid abrupt variations and make the final result more reliable.

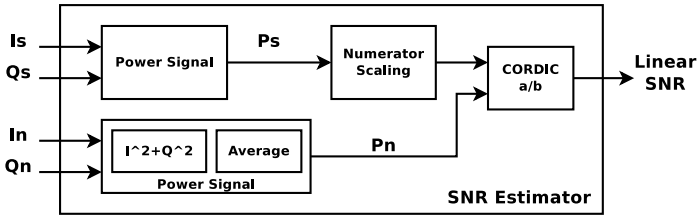


Figure 7.53: Architecture of the SNR Estimator used to measure the BER of the FEC Subsystem.

The last step of the estimation process is obtaining the relation between both averaged powers $SNR = P_s/P_n$. For this calculation, a CORDIC IP is used, and to

avoid underflow/overflow problems, the numerator is scaled and suited so that the output of the CORDIC can fit within the expected range.

Based on the received symbols and on the estimated linear SNR, the Soft-demapper calculates the soft-bits and then the FEC sub-system processes the data in a frame-by-frame basis. As the Deinterleaver must receive a complete frame before starting the deinterleaving process, a FIFO is instantiated between this block and the Soft Demapper in order to buffer the frame. Afterward, the FEC output is compared with the data read from the Payload Memory by the Error Calculator to calculate the Bit Error Rate (BER).

The Error Calculator is composed by two parts: the *Data stream synchronization* and the *Bit comparison*, as presented in Figure 7.54.

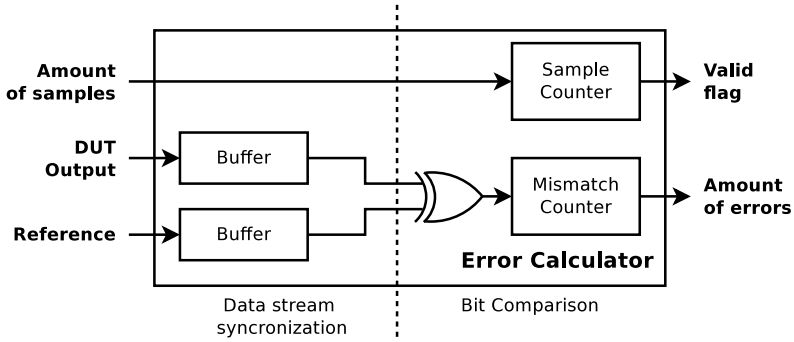


Figure 7.54: Error Calculator internal architecture.

The first part stores the FEC output and the Reference output separately into two buffers and aligns them in order to avoid the comparison of non-correspondent bits. The second part compares the two streams and counts the mismatches found within the specified *Amount of samples*. When this value is reached, the *Sample Counter* signalizes that the output of the *Mismatch Counter* is valid. Afterward, the BER is calculated by dividing the *Amount of errors* by the *Amount of samples*.

7.4 Conclusions

In order to select the adequate algorithms and architectures for the implementation of a digital wireless receiver aiming practical applications, one shall thoroughly explore the system and implementation aspects related to the algorithms the receiver under consideration will make use. Furthermore, for the case of a receiver that shall be compliant to a transmission standard such as the ISDB-T (presented in previous chapters) and DVB-S2, one shall be aware of the frame and pilot structure determined

by the standards. By doing so, it is possible to make the adequate use of those elements in the algorithms used by the receiver.

In this chapter, two solutions for implementation of a DVB-S2 receiver were proposed. The first is a simplified version of the second one and contains the basic algorithms to prove the feasibility of the project. It was also used to explore the selected algorithms. The second solution contains all blocks an DVB-S2 receiver shall contain, including an adaptive equalizer for the case of the reception under ISI conditions, such as in the case of LMSC, as well to reduce the non-linearity effects. Furthermore, that receiver is going to be capable of demodulating all DVB-S2 modulations, working with both frame length and code rates, and blind estimate the baud rate, which is an essential feature for the DVB-S2 receiver working in an autonomous fashion.

The main contributions presented in this chapter were the study and selection of algorithms to be implemented in a pilot-based DVB-S2 receiver and the proposal of a platform for the validation and debug of the FEC decoding subsystem. The methodology used was a thorough literature search, algorithm selection , in some cases the algorithms were adapted (as for instance in the case of the frame synchronizer, soft-demapper and DFE), in other cases they were evaluated and adopted using their original structure or well known approach proposed in the literature (e.g. Gardner clock recovery with farrow interpolation to estimate and compensate sampling clock error). Other important contributions, derived from the work presented in this chapter, were the proposal of implementation architectures for the selected algorithms, aiming low complexity implementation. This was mainly done by the adoption of CORDIC algorithm for trigonometric and some DSP operations such as module and division computation.

The contributions of this chapter were reported in the form of conference papers in [20], [21], [22], [23], [25], [26], [27] and [29], [30], [28] and two were accepted as journal papers: the Adaptive Equalizer [144] and the BCH decoder [24]. Some blocks, such as the DFE, the optimized fine frequency estimator, the improved frame synhcronizer and the BCH decoder will be subjected to USPTO patent application.

The next chapter will present: the methodology used in the DVB-S2 receiver project, the results on the prototyping of the current implemented architecture in FPGA, the BER rate performance for that implementation and its early ASIC synthesis results. Furthermore, the Adaptive Equalizer prototyping results in FPGA, Advanced Soft Demapper and Hard Demapper implementation results in ASIC and FPGA, and the FEC Decoding subsystem evaluations using the evaluation platform will be also presented.

Chapter 8

DVB-S2 Receiver: The Methodology, VLSI Implementation, FPGA Prototype and Test Results

8.1 Introduction

Concerning literature about DVB-S2 receivers, the authors usually only present details on system level aspects of algorithms as well as their simulated and theoretical performances, disregarding the implementations aspects. On the other hand, those who implement them in hardware (specially for ASIC), in general, do not details their architectures and implementation results, which make difficult comparisons.

With regard to commercial ICs, for DVB-S2 reception, the simple specifications provided by the vendors, usually, are not very clear and/or detailed enough to show the real performance of the IC or to provide some clues about the algorithms that have been used to implement the receiver. That kind of information is usually provided under NDA (non-disclosure agreement) and only for companies that plan to use the IC commercially. So, it is natural that potential competitors are no allowed to have access to that kind of information.

Some algorithms that have very good theoretical and/or simulated performance, sometimes are not good candidates to be implemented in hardware, due to the high resources usage, or simply because they can not be directly used in applications

different from that they were originally designed for [1]. In short, despite there are number of ICs on the market to perform DVB-S2 reception, in general one do not know which are the algorithms used by them. One obvious reason for that is the protection of the Intellectual Property.

In the previous chapter I described the main algorithms and architectures for a DVB-S2 receiver overcome real world impairments. The architectures presented there are made of a selection of algorithms that were extensively searched in the literature and carefully studied, in order provide acceptable performance and low complexity after hardware implementation. As previously mentioned, there are two solutions, one that is the final goal of the DVB-S2 receiver project presented in this thesis and a second one that was used to validate the implementation of certain algorithms in hardware, proving the feasibility of the project and attracting partners.

In order to prove that the selected algorithms and architecture, which were presented in Chapter 7, are suitable to be implemented in hardware, they shall be validated. So, in this Chapter I describe the Methodology[2] used to implement the receiver and its blocks, as well as some prototyping results in FPGA and VLSI preliminary results, which have been used to explore the ASIC design flow and to estimate the resource usage of the DVB-S2 receiver in ASIC.

Section 8.2, is devoted to present the methodology used in the DVB-S2 receiver project. Section 8.3.1, presents the *current implemented architecture* prototyping results in FPGA, BER measurements for different code rates and qualitative evaluation of the 8-PSK modulation in different parts of the receiver. In Section 8.3.2, early VLSI design exploration results are presented, also for the *current implemented architecture*. All those results as well as the methodology were reported in [20] and [29].

Section 8.4 presents the evaluation of the BER performance for the FEC Decoding Subsystem (which is made of the LDPC and BHC Decoders and Time Deinterleaver) solely, i.e. without the Signal Processing algorithms, for several code rates. In order to do so, a hardware platform was proposed and implemented. The platform was also used to adjust LDPC parameters. This was done to avoid the long simulation time needed to evaluate LDPC decoders at very low BER performance. The first implementation results on the LDPC Decoder were reported in [21]. The first results on the BCH Decoder implementation were presented in [22], later the entire BCH decoder implementation was presented in [23]. Finally, the overall performance of the FEC Decoding Subsystems , using the proposed platform, was reported in [26].

The remaining Sections, i.e. 8.5, 8.6 and 8.7, are devoted to present the prototyping results of the Advanced Soft Demapper and Hard Demapper, Adaptive Equalizer

[1] e.g. algorithms that use pilot, as training symbols, present different convergence and performance if the pilot structure used is different of the original one.

[2] The methodology adopted to develop the DVB-S2 receiver is based on that used for the ISDB-T receiver.

and the optimized implementation of the Fine Frequency Estimator, respectively. The results on those blocks are report in [25], [27] [144] and [30], respectively. Those blocks are still not integrated in the DVB-S2 receiver and are part of the final receiver architecture.

8.2 The Methodology

The methodology adopted for the study and implementation of the DVB-S2 receiver proposed in this thesis is derived from the one adopted for the ISDB-T receiver implementation. Nevertheless, there are some differences that are going to be described along this section. One example of that was the need to implement a basic SDR-based DVB-S2 transmitter in order to generate real stimulus to validate the receiver prototype in FPGA and to be used as input to debug the models and HDL blocks. This was done due to lack of funding at the beginning of the project. Nevertheless, the current laboratory setup contains state-of-the-art professional equipments used to validate the proposed receiver architecture.

The definition of the Design Flow to be followed was the first task of the project. A simplified block diagram of the entire process is shown in Figure 8.1. One of the major tasks when developing digital receivers is the conception and implementation of a model that can be used for several tasks along the project, e.g. refinement of algorithms and stimulus generation for RTL verification. In general, the very first model to be created is the transmission path, and that was the case in this work. Initially, for convenience, a DVB-S2 reference model provided by the System Vue tool from Agilent was used. Nevertheless, in order to be more flexible a transmitter model was created using Matlab and Octave. This model was used to create a DVB-S2 compliant transmission hardware, which was implemented using GNU Radio Companion and USRPN210 SDR platform. The transmitter hardware was validated using a commercial receiver.

The reference transmitter model can generate baseband stimulus for algorithm modeling, RTL design and verification, and physical tests on FPGA. In addition, DVB-S2 compliant stimulus can be used to perform system test to confirm receiver performance, in FPGA. In the next phase of the project, it will be used to generate stimulus for the complete ASIC design flow. It is worth recalling that this work only presents the basic ASIC estimates for 65nm technology, based on the HDL code that was successfully prototyped in FPGA. This is used to provide initial estimates (e.g. area and power) as well as to find possible bugs, masked by FPGA tool specific optimizations, beforehand.

The design of receiver blocks was done in the following manner:

1. First of all, a receiver architecture was defined, containing all the sub-blocks of

a digital Intermediate Frequency (IF) receiver and DVB-S2 specific blocks;

2. Once defined the architecture, algorithms and functionality specifications were done, followed by the models implementation using a high level language (Matlab or Octave compliant with Matlab Code);

3. After models validation, block specifications for RTL design were written, followed by the platform independent design, using VHDL, and a simple verification made by the designer of the block;

4. Aiming independence between design and verification, whenever possible, the full block verification was done by a specific verification Engineer;

5. Once verified the design, synthesis targeting Altera's FPGA (Stratix IV) -refer to [145] [146]- was performed, followed by physical verification, i.e. injection of real stimulus into the FPGA synthesized block, capture of block answer and comparison to expected results;

6. After validation through physical verification, integration of the blocks was carried out gradually, always followed by new physical tests, until the completed receiver was integrated;

7. System tests were done in order to validate the functionalities of DVB-S2 receiver;

8. A simplified ASIC design flow was also done for early design issues detection. As can be seen in Figure 8.1, real signal capture can be performed at any time to be used for refinements of the models, RTL design or throughput improvements.

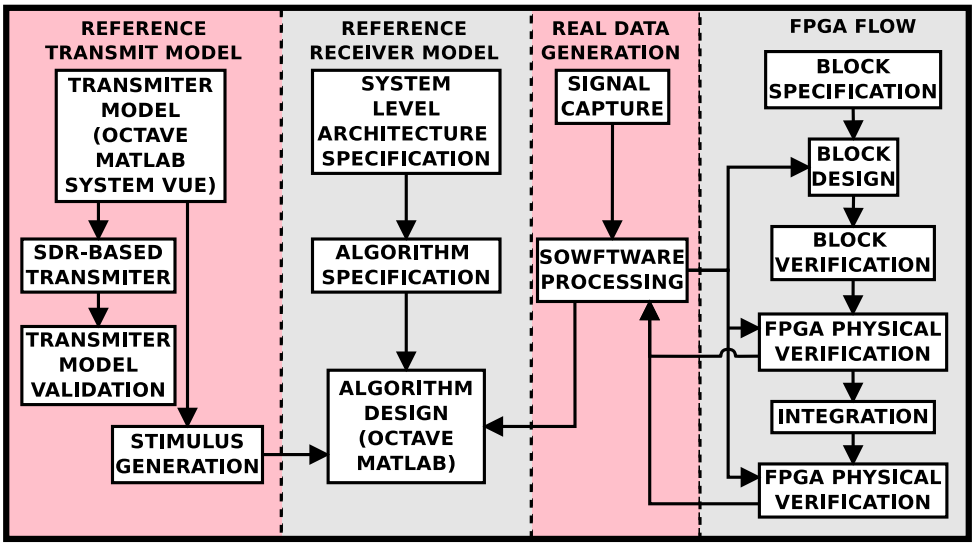


Figure 8.1: Simplified design flow chart.

8.3 Prototype in FPGA and VLSI Implementation in 65nm CMOS

In this section examples of implementation results are presented in order to show that the proposed current architecture and chosen algorithms work properly in FPGA and it is possible to proceed to the ASIC design flow. Preliminary ASIC design exploration are also introduced. The results, in FPGA, show that the resource usage of the modules are moderate and the ASIC will have an acceptable die area. Results for the blocks that were synthesized in FPGA but not physically validated yet, i.e the Adaptive Equalizer and the complete Soft Demapper, that at the moment is working for QPSK,8-PSK and 16-APSK, are also presented.

8.3.1 Full Receiver FPGA Prototyping

FPGA prototyping is done in order to guarantee the correct functioning of the implemented design. The results presented in this section regards the currently implemented receiver architecture, introduced in the previous chapter and shown in Figure 7.2. A sample of the implementation results is given in this section: signals acquired from some key points of the synthesized blocks are compared to Matlab models and RTL simulation results. A summary of FPGA resources usage is presented. In addi-

tion, an overall BER performance for code rates 3/5, 5/6 and 9/10 is given. BER for other code rates, for the isolate FEC Decoding subsystem free of timing, frequency and phase impairments are shown in Section 8.4.

In Figure 8.2, the left subfigures show the baseband constellation after resampling, for SNR values of 16 (bottom) and 21(top) dB. The subfigures on the right show the same constellations after frequency synchronization (coarse and fine), phase and amplitude correction.

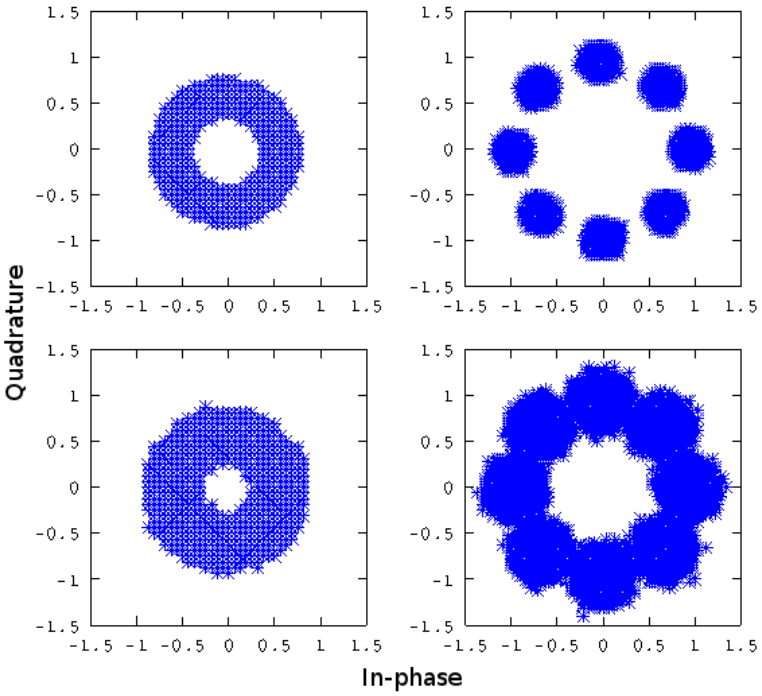


Figure 8.2: Constellations captured from the FPGA prototype for SNRs of 16 dB (bottom) and 21 (top) dB.

FPGA implementation results are summarized in Table 8.1. It gives a brief summary of the logic usage by design units, split in registers, ALUT's, memory bits and DSP resources. Refer to [145] for details on those resources.

In order to validate the hardware implementation, the BER was measured for the following receiver configuration: Bandwidth =1.5625 MHz, 8-PSK, Code Rates 3/5, 5/6 and 9/10, Normal Frame with pilots. These measurements are shown in Figure 8.3. The BER curves are shifted from the expected quasi-error free target SNR, due to the residual errors from the synchronization algorithms (frequency and phase) which are still being characterized and optimized for lower SNR values, the FEC

Table 8.1: DVB-S2 FPGA Prototyping Results For Altera's Stratix IV.

	DESIGN UNITS	ALUTS	REGISTERS	MEMORY BITS	DSP ELEMENTS (18-BIT)	DSP ELEMENTS (18x18)
DATA PROCESSING	SOFT DEMAPPER QPSK/8-PSK	879	218	0	6	3
	SNR ESTIMATOR	1487	1301	45	2	1
	BASE BAND SIGNALING	311	126	0	0	0
	BASEBAND DESCRAMBLER	13	18	0	0	0
	BIT TO ASI	25	21	0	0	0
	DEINTERLEAVER	504	212	864000	0	0
	DEMAPPER FIFO	96	37	518400	0	0
	PAYLOAD FILTER	7	35	0	0	0
	PILOT FLAGS	34	26	0	0	0
	LDPC DECODER	53607	59692	4240736	0	0
	BCH DECODER	4928	926	58192	0	0
	PL DESCRAMBLER	129	113	0	0	0
SIGNAL PROCESSING	L. PASS FILTER I	7935	5328	611	0	0
	L. PASS FILTER Q	7935	5328	0	0	0
	NCO + CORDIC	2158	2396	0	0	0
	DECIMATOR	20	37	0	0	0
	RCC I	3387	2252	0	0	0
	RCC Q	3413	2265	352	0	0
	TIMING RESAMPLING	325	455	132	12	7
	FINE FREQUENCY[1]	1882	1385	0	22	4
	AGC	295	205	102	18	10
	COARSE PHASE[2]	2663	2179	1728	18	9
	FRAME SYNC	7127	9862	3275	4	4
	COARSE FREQUENCY	5895	3709	64	48	48
	PILOT FLAGS	34	26	0	0	0
	PILOT VALUES	69	87	0	0	0

[1]Fine Frequency: Also uses one 12x12 and four 36x36 DSP elements.
[2]Coarse Phase: Also uses one 36x36 DSP element.

(LDPC+BCH) was tested separately and it is compliant with the ETSI standard, as will be shown in Section 8.4.

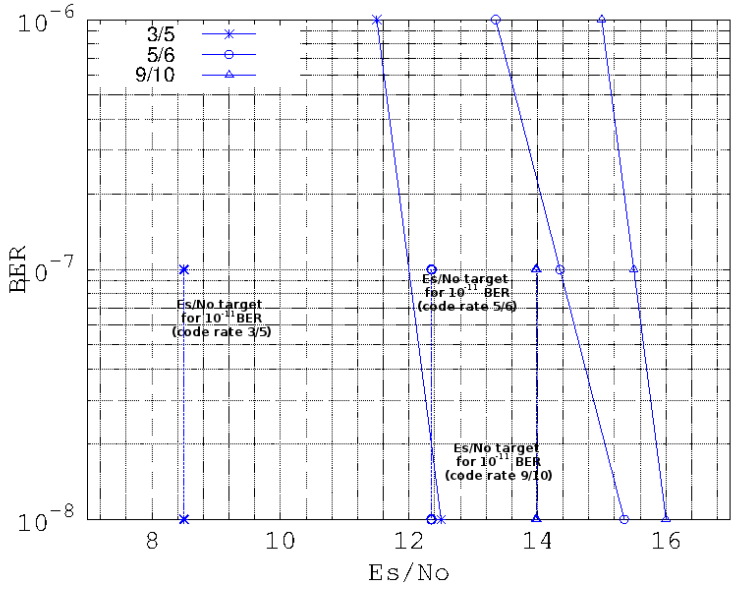


Figure 8.3: Normal Frame BER for Code Rates 3/5, 5/6 and 9/10: FPGA prototype.

8.3.2 Receiver VLSI Physical Synthesis

Synthesis, floorplanning and placement with basic constraints were performed in order to estimate the die area, and to identify possible design improvements and aspects which were not analyzed in the FPGA project. It is worth mentioning that detailed timing analysis was not performed since it is not the focus at this phase of the project.

Figure 8.4 shows the amoeba view of the IPs placement. The Signal Processing partition is allocated in the upper part of the die, and the Data Processing partition is spread along the remaining area. The memory IPs used by the FEC blocks, i.e. Deinterleaver, LDPC and BCH decoders, are allocated in a clockwise order from the upper right corner to the upper left, following their respective blocks. The LDPC decoder logic and its memories represent about 80% of the die area. That percentage is justified by the amount of processing units required to achieve the target level of parallelism for LDPC decoding.

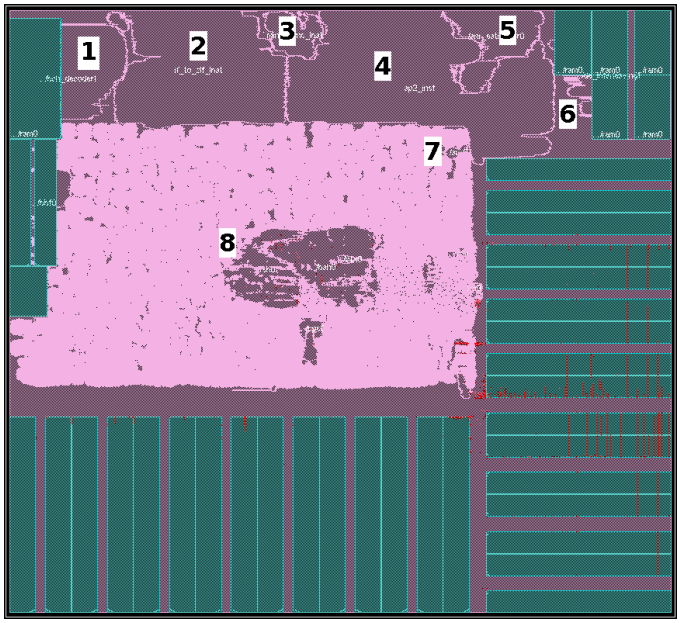


Figure 8.4: Amoeba View of the die: 1) BCH Decoder, 2) IF to ZIF, 3)Time synchronization and Decimator, 4) PLSC decoding, Frame synchronization, Coarse and Fine Frequency, Coarse Phase and AGC, 5) SNR Estimator, 6) Deinterleaver, 7) Glue logic, 8) LDPC logic. The not numbered blocks are the memories. The memories in the top-right corner belongs to the Deinterleaver. The Memories in the top-left corner belong to the BCH Decoder. The remaining memories, belong to the LDPC Decoder.

Table 8.2: Synthesis Report Results, Generated by the EDA/CAD Logic Synthesis Tool.

	Instances Number (Devices Count)	Area (μm^2)	Internal Power (nW)	Leakage Power (nW)
Auto-Floorplan	483821	13066916.663	84169037.008	1733173.315
Manual-Floorplan	533172	13218313.143	107262766.515	1747496.56

8.4 FEC FRAME Decoder: FPGA Prototype

In order to verify the functionality of the implemented FEC subsystem as well as to measure the performance of the FEC subsystem without the influences of residual time, frequency and phase impairments (i.e. after estimation and correction), a verification platform was designed. That platform was already extensively explained in Section 7.3.10.4, and its architecture is depicted in Figure **??**.

In this section I present the FPGA prototyping resource usage for the entire platform, as well as the measured BER performance as a function of the AWGN.

Table 8.3: FEC FRAME FPGA prototyping results for Altera's Stratix IV

Entity	Comb ALUTs	Registers	Memory bits	DSP elements 18-bit / 18x18 / 36x36
Main platform elements				
Error Counter	208	178	0	0 / 0 / 0
Mapper	89	30	0	0 / 0 / 0
Stimulus RAM	55	38	64800	0 / 0 / 0
Reference RAM	56	36	58192	0 / 0 / 0
SNR Estimator	2329	1967	0	4 / 2 / 0
Monitor RAM	40	79	60	0 / 0 / 0
Config RAM	39	100	80	0 / 0 / 0
IP's (part of the receiver)				
8-PSK Soft-demapper	813	194	0	6 / 3 / 0
FIFO for De-interleaver	87	71	327680	0 / 0 / 0
De-interleaver	22	8	324000	0 / 0 / 0
LDPC Decoder	48513	53677	3860576	0 / 0 / 0
BCH Decoder	9153	1250	58192	0 / 0 / 0
Others	1372	3084	104608	12 / 2 / 2
Total	62776	60712	4798188	22 / 7 / 2

The DVB-S2 standard [77] defines the target E_s/N_0 for Quasi-Error-Free (QEF) performance in all the modulation/code-rate modes. For converting the variable E_s/N_0, which considers single sided noise power spectral density, into the real SNR, which is based on double-sided noise, Equation 8.1 was used.

$$SNR(dB) = Es/No(dB) + 3 \tag{8.1}$$

Four code rates were tested: 3/4, 5/6, 8/9, 9/10, all of them for 8PSK and normal-length frames. The combination of external attenuators and internal scaling was

arranged so that the required SNRs could be properly set. Table 8.4 presents the ideal E_s/N_0 for the tested modes.

Table 8.4: Quasi-Error-Free performance requirements

Mode	Ideal E_s/N_0
8PSK 3/4	7.91
8PSK 5/6	9.35
8PSK 8/9	10.69
8PSK 9/10	10.98

For each of the four code rates, the test was executed by accumulating errors in 10^{10} useful bits (discarding the parity bits). The modulation was 8PSK, and the LDPC decoding algorithm was the Normalized Minimum-Sum with $\alpha = 2$. The BER result for all code rates at the target SNR for QEF performance was zero.

Once it was verified that the implemented FEC sub-system delivers the expected performance at the ideal E_s/N_0, more SNR points were simulated for drawing the BER curves of the code rates. The results are presented in Figure 8.5.

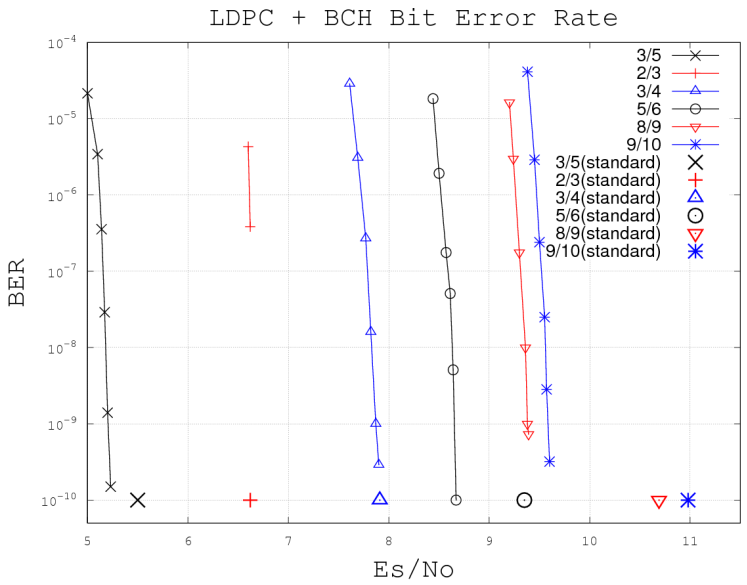


Figure 8.5: BER performance - FEC only.

8.5 Soft and Hard Demappers: FPGA Prototype and VLSI Physical Synthesis

This section shows the FPGA prototyping and ASIC synthesis results for the HD and SD.The HD and SD were implemented using the architecture defined in Section 7.3.9.3. The validated VHDL codes were prototyped in FPGA (Stratix IV GX). The prototyping results are presented in Table 8.9. According to the obtained results, the maximum frequencies that the SD and HD can operate on are 111.54 MHz and 110.07 MHz, respectively.

Table 8.5: FPGA Prototyping Results for Altera's Stratix IV.

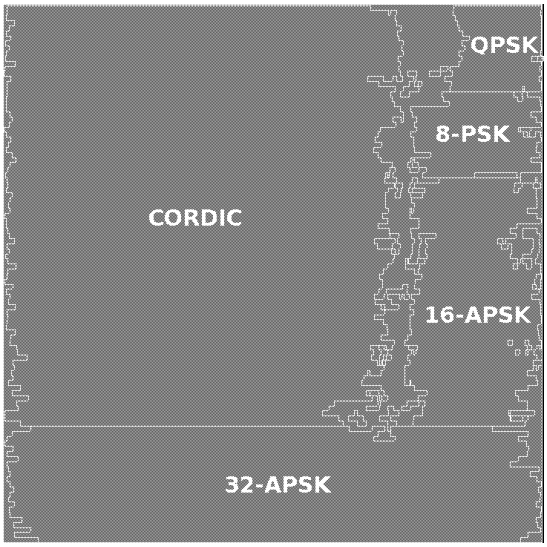
Module	Combinational ALUTs HD/SD	Memory ALUTs HD/SD	Registers HD/SD	Memory Bits HD/SD	DSP (18-bits) HD/SD
QPSK	44/283	0/0	34/74	0/0	0/4
8PSK	68/768	0/0	35/308	0/0	0/10
16-APSK	106/2028	0/2	49/1 657	0/1088	0/8
32-APSK	179/-	0/-	60/-	0/-	0/-
CORDIC	1 031/927*	16/0	834/811*	0/0	0/0
Total	1 905/3 224	16/2	1 327/2 039	0/1088	0/22

The HD and the SD were logically synthesized with the Encounter RTL Compiler, [147], using a 65 *nm* HVT library from Global Foundries working with supply voltage of 1.08 V at 62.5 *MHz*. The backend was only carried on with the base clock frequency, since it was the one relevant in this phase of the project. Table 8.6 shows the cells usage for each block of HD and SD, including glue logic. It is important to note that the CORDIC is an internal block in 16-APSK SD microarchitecture, so its cells are included in the final amount of SD cells (*).

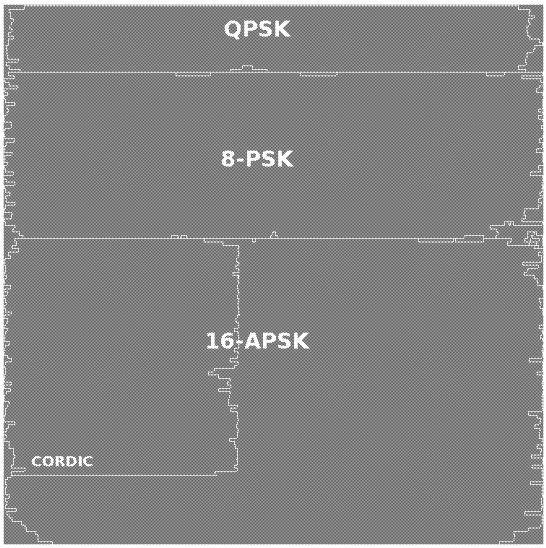
Table 8.6: Logical synthesis results for $65nm$ CMOS

	Hard Demapper		Soft Demapper	
Module	**Cells**	**Area (μm^2)**	**Cells**	**Area (μm^2)**
QPSK	120	708	2 630	10 628
8-PSK	271	796	6 806	30 371
16-APSK	1 031	2 391	15 065	91 537
32-APSK	2 225	5 105	—	—
CORDIC	4 282	28 084	4 282*	28 080*
Total	9 410	45 112	24 698	132 919

The results of physical synthesis is shown in Table 8.7. The back-end *amoeba* view of both blocks is in Figure 8.6, where the Figure 8.6(a) presents the HD and the Figure 8.6(b) the SD.



(a) Hard Demapper.



(b) Soft Demapper.

Figure 8.6: Back-end amoeba view.

Table 8.7: Physical synthesis results for $65nm$ CMOS

Demapper	Gates	Dimensions (μm)	Area (μm^2)	Density (%)	Power (mW)
Hard	22 944	170 × 170	28 900	76.215	0.7102
Soft	61 857	290 × 290	84 100	70.610	4.007

8.6 Adaptive Equalizer: FPGA Prototype Results

]

The Adaptive Equalizer architecture proposed in this thesis and introduced in Section 7.3.7, was not physically tested in FPGA yet. Nevertheless, it was prototyped in order to estimate the resource usage. This synthesis was used to check the amount of logic, registers and digital signal processing elements used by each sub-block. Table 8.8 presents the main results for the proposed architecture presented in Figure 7.29.

Table 8.8: Adaptive Equalizer: FPGA Prototyping Results for Altera's Stratix IV FPGA.

SUB-BLOCK	ALUTS	REGISTERS	DSP
FIRF	0	512	0
FIRB	0	512	0
LMSF	0	640	0
LMSB	0	640	0
FIR-LMS Core	4633	0	64
HD	3866	2831	0
CMA	1020	858	4
TOTAL	9519	5993	68

As it can be seen in Table 8.9, the block that consumed the greatest amount of logic elements was the FIR-LMS-Core, even though it was implemented using the technique of resource sharing, which resulted in the use of 64 DSP elements and 4633 ALUTs for glue logic. The second block that consumed greater amount of logical resources was the HD. The HD makes use of a pipelined CORDIC, which increases considerably its maximum operation frequency up to 230 MHz.

8.7 Optimized Fine Frequency Estimator: FPGA Prototype Results

The FFE architecture, described in Section 7.3.3.2, was implemented and coded using the VHDL hardware description language. In order to verify properly this implementation, a golden model was implemented using the GNU Octave high-level interpreted

language [148]. The results obtained by the golden model were adopted as reference and compared to those using the implementation through GHDL, for RTL simulations.

The FFE design was synthesized and prototyped on FPGA (Stratix IV GX FPGA development kit from Altera [146]) for an initial validation and preliminary results showing that this design can run at $150MHz$. Further on, the FFE was integrated in the DVB-S2 receiver presented in section 8.3.1. The resource usage for the proposed architecture, shown in Figure 7.18 and the parallel implementation, shown in Figure 7.16, are listed in Table 8.9.

Table 8.9: FPGA prototyping results for Altera's Stratix IV

Implementation	**Combinational ALUTs**	**Registers**	**18-bit Multipliers**
Conventional L&R	22 890	17 057	160
Proposed architecture	3 559	3 851	13
Reduction (%)	84	77	91

The proposed architecture reduces significantly the number of logic used in the design, regarding a brute force full correlation implementation of L&R (named Conventional L&R in Table 8.9) . Additionally the number of multipliers was reduced in 91%, achieving a total Stratix IV utilization reduction of 78%.

8.8 Conclusions

In this chapter, and overview on the main FPGA prototyping and early VSLI design flow exploration results were provided, for the current implemented architecture of the DVB-S2 receiver proposed in the project presented in this thesis, as well as for the Advanced Soft-Demapping, which is not integrated yet on the DVB-S2 receiver. FPGA prototyping results for the Adaptive Equalizer as well as for the optimized Fine Frequency Estimator were also presented. In addition, FEC Frame decoder evaluation using a platform specially designed for that purpose was also provided.

The results show that the proposed DVB-S2 receiver architecture when implemented in hardware works with acceptable BER performance, although still needs improvements. The early VLSI design exploration shows that the area of the DVB-S2 receiver IC presents a size similar to that presented by the DTV01 receiver, which was shown to be an acceptable area for an ASIC implementation targeting consumer electronics receivers. The area of commercial ICs, usually are not provided by vendors, which make difficult to compare our designs with other similar designs. In the literature searching, it was not found any DVB-S2 receiver ASIC design to compare

with our design. Nevertheless, just to provide a simple source of comparison, in the DVB-S2 user guideline (ETSI TR 102 376 V1.1.1) they provide a reference number for the LDPC decoder (which is the block that dominates the receiver area). There, it is informed that the area for a LDPC decoder (it is not mentioned which was the algorithm used and the number of Soft Bits) which they claim "'to make it easy to trade-off throughput and complexity"': "'the maximum decoder complexity was set to correspond to 14 mm^2 of silicon using a $0,13$ μ m technology"'. Our initial ASIC design exploration presents about 13 mm^2 using 65 nm technology and contains all DVB-S2 except those already mentioned (which are not present in the current implemented architecture).

The FEC Decoding subsystem performance evaluated using the test platform presents acceptable performance, with some minor degradation and sometimes it outperforms the standard target values.

The other blocks (i.e. Adaptive Equalizer, Advanced Soft Demapper and optimized Fine Frequency Estimator), that are still under evaluation before been integrated in the receiver, work properly for simulation (in Matlab and HDL), nevertheless they still need to be physically tested in hardware. The Soft Demapper still need to be extended for 32-APSK modulation.

Although the final DVB-S2 receiver is not totally finished and optimizations and improvements can be done in the current implemented algorithms, and there are some blocks that need to be integrated, physically tested and/or implemented (such as the Baud Rate Estimator), it is possible to conclude that the proposed algorithms and architecture are suitable for implementation in hardware and present good performance.

Despite the work that shall be done and were already mentioned previously, there are other future works that can be done to extend the funcionalities of the DVB-S2 receiver, such as: testing the current synchronization algorithms for the case of LMSC and make the possible improvements needed to support those scenarios, and extend the receiver to support DVB-S2X parameters and frame structure. Two very clear examples about which shall be done to support DVB-S2X are: the extension of the Soft-Demapper for the new modulations contemplate by that amendment and the extension of the FEC Decoding subsystem for the new decoding capabilities.

Chapter 9

Conclusions and Future Work

This thesis presents architectures and algorithms for the prototyping in FPGA and implementation in ASIC, of an ISDB-T and DVB-S2 receivers. The aim of the thesis was to propose an Integrated Circuits that would take advantage of the momentum created by the birth of Digital Television in Brazil and the efforts of the Brazilian government to create a strong national ecosystem in Microelectronics.

The main conclusion of this thesis is that the proposed architectures and algorithms are suitable for silicon. This was shown through simulations, BER rate evaluation, prototyping in FPGA and implementation in ASIC or via initial ASIC mapping exploration, which was the case for the proposed DVB-S2 receiver.

The cost of design and producing an IC sometimes make it prohibitive for IC Design Houses in the early stages, or in the initial rounds of funding raising, to access state-of-art silicon technology. The main causes for that are the high costs to rent the ASIC design tools and for the production masks of the IC. So, an unsuccessful design could means the end of the Design House. To reduce the risks of an unsuccessful ASIC design, the prototyping of the design in FPGA was a step adopted in the design flow of the projects presented in this thesis. So, with a functional hardware in FPGA it was possible to attract private and government funds, for tools renting, MPW as well as for the final production mask.

The thesis presents the blossoming of Digital Television in Brazil, the evolution of the local Microelectronics industry and the recent government efforts to create an environment that made it possible to create a project aiming at to support the local industry to conceive, design and test high complexity ICs in Brazil. Thanks to the core goal of this thesis that was to propose a feasible architecture for an ISDB-T receiver to be implemented in ASIC, other goals were possible to be achieved during the project development, like:

- Training of a large number of Digital IC Designers in concepts regarding all implementation and test aspects of wireless communications systems and Digital Signal Processing;
- Foundation for close relationship between local industry and Design Houses (such as Idea! Electronic Systems and the Hardware Design Department at Eldorado) to implement products with the ICs conceived and designed in Brazil;
- Creation of a strong relation between the local DH and the international chain of service and IP providers and foundries.

The goals were in concordance with those of IC-Brazil Program of the Brazilian Ministry of Science, Technology and Innovation. Furthermore, during the ISDB-T project, one of the goals of the thesis was achieved (create the basis for the development of new receivers projects), and a new IC project started, i.e. the DVB-S2 receiver project which is also part of this thesis. Therefore, the projects developed in this thesis were responsible not only for generating an IC for DTV reception, but rather to help the IC-Brazil goals become reality. This thesis was also the basis for the project *Development of a Multi-Standard Demodulator ISDB-TDVB-S2*, whose results are presented in the publication list shown in Appendix A. The multi-standard project is under development and part of the work was presented in this thesis.

With the demonstration in silicon of the proposed ISDB-T receiver architecture, it was possible to prove that Eldorado Research Institute and Idea! Electronic Systems consolidated all the needed competencies from the conception and algorithm design to the bring-up of a complex IC.

In the introductory Chapter 1, the motivations for this thesis are presented. The chapter contains the Evolution of Digital Television in Brazil and its influence in the local industry. The National Program of Microelectronics foundations and goals are also presented therein. In addition, the thesis objective are presented. Furthermore, the Introduction presents the thesis outline and a survey of the thesis contributions.

The understanding of the fundamentals for the design of the ISDB-T and DVB-S2 digital receivers digital are presented in Chapter 2. They are three: the characteristics of OFDM systems, the wireless channels and the impairments that a digital wireless receiver shall overcome to work properly.

To propose a suitable architecture for any receiver based on a specific standard, the fundamental understanding of the related standard is strongly necessary. So, in Chapter 3 I present part of the study and the main characteristics and parameters of ISDB-T and DVB-S2 standards.

Chapter 4 presents a summary of the the methodology used in the project of the ISDB-T receiver. That methodology, was proven to be valid and, with minor modifications and additions, was used later in the DVB-S2 receiver project.

Chapter 5 is devoted to the presentation of the algorithms and the CORDIC-based architecture used in the ISDB-T receiver. It also shows simulation results for some algorithms and BER curves for the proposed architecture.

Within Chapter 6 the ASIC implementation results as well as the laboratory environments for both FPGA prototype and VLSI implementation are shown. Results on the evaluation of the implemented ASIC are also presented. Some hardware used to test the IC and pre-products (e.g. the MCM) based on the ISDB-T receiver IC are presented as well.

Chapter 7 is devoted to the proposed architectures and algorithms for the CORDIC-based DVB-S2 receiver, and Chapter 8 shows the used methodology, the laboratory setups and several results on the implementation of the DVB-S2 proposed architecture and algorithms. Furthermore, BER performance results are presented for the FPGA prototyping.

The next subsections shows the main conclusion with respect the each one of the mains subjects covered in this thesis, as well as future works directions.

9.1 Fundamental Understanding on the Wireless Radio Impairments

A fundamental understanding on the wireless channel and radio impairments was achieved by means of a through literature review and simulations of those impairments. Without fundamental understanding on radio impairments and its correct modeling, it is not possible to select and/or modify algorithms, propose implementation architectures and implement algorithms aiming a wireless receiver that must be robust and work properly in all real world environments. The main radio and wireless channel impairments were presented in Chapter 2.

It is worth to mention that when designing digital wireless receivers, the impairments observed by the receiver are strongly influenced by the performance of the RF front-end (i.e. tuner), which most of the time has its own clock source. Furthermore, depending on its architecture, the tuner can deliver a signal to the IF/Baseband digital demodulator with low/high level of certain impairments. An example of that is the phase noise, which presented low level for all tuners used during the ISDB-T receiver project. A consequence of it is that it was not selected and implemented any algorithm to overcome this impairment in the DTV receiver. Despite the negative effects of the phase noise in the satellite link, until the moment it was not considered any special technique to improve the performance of the proposed receiver for that specific impairment, that was not considered the most harmful to the receiver.

Despite all care taken during the definition on the ISDB-T receiver algorithms

and impairments simulations, the effort on finding low complexity algorithms (as the one used for channel estimation) made the selected algorithm for channel estimation/interpolation only work well for channel with exponential-decaying power delay profile and for SFN-like channels with low delay spread. This means that the effects the SFN channel in the selected algorithm for channel estimation was not evaluated properly, and the result of that is that enhancements on the channel estimator shall be done in order to extend the functionality of the receiver for those scenarios.

9.2 Proposal of a ISDB-T Receiver Architecture and Algorithms Exploration

One of the goals of this thesis was to define a receiver architecture and algorithms for the implementation of a ISDB-T receiver with focus in commercial applications, in the context of a governmental program aiming at training IC designers. Therefore, despite the training environment the project evolved, at the end of the project the receiver should present good BER performance for AWGN and wireless channels, as well as achieve low implementation complexity, that in short can be measured by the die area of the implemented ASIC and comparing it with other(s) commercial IC die area.

Once understood the main impairments the receiver should overcome, the next steps were to select the algorithms used to tackle those impairments and define a suitable architecture for hardware implementation. It is worth to recall that further than the algorithms, the ISDB-T receiver should be made of several auxiliary blocks/functions, like BER measurement capability, SNR estimators and access to a register bank to read and writing internal receiver parameters.

Due to the limited interest on ISDB-T standard (compared to the DVB-T standard that at that time was widely used world wide), because it was originally adopted only in Japan, it was not so straight finding literature references containing algorithms and architectures that could be implemented or used in a straightforward manner. As a consequence, this led to a lack of similar design for comparison. So, the approach adopted was to search among DVB-T and other OFDM-based receivers, the candidate algorithms (such as those devoted WLAN and/or Wimax) to be adopted and/or modified to be used in the ISDB-T receivers.

It was shown, by means of the simulations presented in Chapter 5 and later in the implementation results presented in Chapter 6, that the selected algorithms and proposed implementation architectures for the ISDB-T receiver can overcome most of the real world impairments with acceptable level of degradation. Although they present acceptable performance in most of the scenarios that a real ISDB-T receiver would be subject to (e.g. different crystal p.p.m. values, wireless channels with exponential-decay power profile), the proposed receiver did not work well under SFN-

like channels with delays greater than $4\mu s$. It was found that this occurs because the Equalizer does not provide a good performance for the channels which frequency response present equidistant deep spectral nulls (which is the case for SFN). One possible solution for that is to perform channel interpolation in time domain. In spite of that, it is possible to conclude that if the wireless channels has exponential-decaying power delay profiles, which is going to happen in most of the reception scenarios, the receiver will work properly and the performance of the channel estimator and coarse/fine boundary estimators and other blocks are going to be appropriate.

9.3 ISDB-T Receiver: VLSI Implementation , FPGA Prototype and Test Results

In general, despite the issues with SFN channels, it is possible to conclude that the selected algorithms when mapped to hardware provide a strong robustness to the receiver and make it work properly, overcoming the main radio and wireless channel impairments for most of the real world scenarios. The test results presented for the ASIC implementation show that the receiver was tested under frequency error, sampling clock error, timing error, AWGN and multipath wireless channels.

Regarding the complexity, the selected algorithms and architecture, when mapped to ASIC present an acceptable area for an ISDB-T receiver ASIC devoted to consumer electronics. Nevertheless, it is worth to highlight that, even after an extensive literature review, it was not found any other ASIC implementation to compare our design, despite the ISDB-T receiver ASIC named SMS3230, which specifications can be found in the commercial website of Ciano. So, by comparing the designed receiver area with that presented by Ciano's ISDB-T commercial IC receiver SMS3230, which they claim to have $4 \times 4\ mm$ and used similar 65 nm process, we can conclude that the achieved die area is similar to a commercial receiver and by consequence is commercially viable. Those are not strong arguments but that was the only source of comparison found. Another possible manner to compare the complexity, in a rough manner, is by checking the number of soft bits used the ISDB-T receiver vendors. Once this number is known, it is possible to extrapolate the size of the deinterleaver memory. As the area of the IC is dominated by the memories, specially the deinterleaver memory, then a rough estimation of the competitor area would be possible.

9.4 Proposal of a DVB-S2 Receiver Architecture and Algorithms Exploration

The choice of the algorithms to be used by a DVB-S2 receiver has a tremendous impact in the flexibility and performance of the receiver when operating in real world

environment and in different applications. This can be realized by the evaluation of the implemented solutions presented in Chapters 7 and 8.

The main contributions on DVB-S2 receiver architecture and algorithm exploration were the study and selection of algorithms to be implemented in a pilot-based DVB-S2 receiver and the proposal of a platform for the validation and debug of the FEC decoding subsystem. The methodology used was a thorough literature search, algorithm selection , in some cases the algorithms were adapted (as for instance in the case of the frame synchronizer, soft-demapper and DFE), in other cases they were evaluated and adopted using their original structure or well known approach proposed in the literature (e.g. Gardner clock recovery with farrow interpolation to estimate and compensate sampling clock error). Other important contributions, derived from the work presented in Chapter 7, were the proposal of implementation architectures for the selected algorithms, aiming low complexity implementation. This was mainly done by the adoption of CORDIC algorithm for trigonometric and some DSP operations such as module and division computation.

In Chapter 7 two solutions for implementation of a DVB-S2 receiver were proposed. The first is a simplified version of the second one and contains the basic algorithms to prove the feasibility of the project. It was also used to explore the selected algorithms. The second solution contains all blocks an DVB-S2 receiver shall contain, including an adaptive equalizer for the case of signal reception in the presence of ISI (such as in the case of LMSC) as well as to reduce the non linear effects of the TWTA. For that, in this thesis, a DFE which makes use of the DVB-S2 frame structure was proposed. This work was presented in [27] [28] and [144].

The contributions presented in this thesis, on DVB-S2 receiver architecture and algorithm exploration, were reported in the form of conference papers in [20], [21], [22], [23], [25], [26], [27] and [29], [30], [28] and two were accepted as journal papers: the Adaptive Equalizer [144] and the BCH decoder [24]. Some blocks, such as the DFE, the optimized fine frequency estimator, the improved frame synchronizer and the BCH decoder will be subjected to USPTO patent application.

9.5 DVB-S2 Receiver: FPGA Prototype, Early VLSI Design Exploration and Test Results

The results presented in Chapter 8 shown that the proposed DVB-S2 receiver solution when implemented in hardware works with acceptable BER performance degradation, although still needs improvements. The early VLSI design exploration shows that the die-area of the DVB-S2 receiver IC presents a size similar to that presented by the DTV01 receiver, which was shown to be an acceptable area for an ASIC implementation targeting consumer electronics receivers.

The die-area of commercial ICs, are not usually provided by vendors, which make difficult to compare our design with other similar designs. In the literature review, it was not found any DVB-S2 receiver ASIC design to compare with ours. Nevertheless, just to provide a second source of comparison, in the DVB-S2 user guideline [78] (ETSI TR 102 376 V1.1.1) and [149], the authors provide reference numbers for an LDPC decoder (which is the block that dominates the receiver area) devoted to DVB-S2. The guideline [78] and [149] provides us the die-area for such LDPC[1]. The authors state that *"the maximum decoder complexity was set to correspond to* $14\ mm^2$ *of silicon using a* $0,13\ \mu\ m$ *technology"* using a *"recursive"* decoding technique. Nevertheless, the authors do not mention which was the number bits used for the LLRs. The initial ASIC design exploration of our DVB-S2 receiver presents a total area of about $13\ mm^2$ using 65 nm technology and contains all DVB-S2 receiver blocks except those already mentioned in Chapters 7 and 8. Therefore, the number presented above reinforce that our design has an acceptable die-area for commercial applications.

Although the final DVB-S2 receiver is not totally finished and optimizations and improvements can be done in the current implemented algorithms, and there are some secondary blocks that need to be integrated (such as the DFE), physically tested and/or implemented, it is possible to conclude that the proposed algorithms and architecture are suitable for implementation in ASIC and present good performance. Furthermore, it can be conclude that the die-are of the receiver is acceptable, for commercial applications.

9.6 Creating and Proving the Competences for ASIC Implementation in Brazil

In short words, there two ways to start a ASIC project regarding front-end design: buying or doing the design. Buying IPs (specially the silicon proven) sometimes is very expensive and in the case of certain blocks (e.g. synchronization blocks) there are no front-end vendors for the desired block/system. Therefore, in order to augment the profits and appropriating of advanced knowledge, it is reasonable to affirm that one shall adopt the second approach. To do so, people with system level and front-end design competences on the area of interest need joining to the ASIC design process. So, a complete design team shall be made of system level and algorithm designers, front-end design team and the ASIC mapping team. This thesis contributed for training designers in the aforementioned areas.

It is was shown throughout the all results presented along this thesis that the main competences to specify, model, design and perform the mapping of a digital

[1] According to [78] and [149], the LDPC was the winner of a selection process which, based on computation simulation, compared seven proposals of FEC.

wireless receiver to ASIC technology, as well as evaluating the implemented ASIC, were achieved and consolidated during both presented projects. In the first project, i.e. the ISDB-T receiver, the technical basis and the methodology for ASIC design of digital wireless receivers were settled. During the second project, i.e. the DVB-S2 receiver, despite it is not fully implemented in ASIC yet, the basis developed during the first project were exercised and applied in a successful manner, to a different kind of receiver. In this way, this thesis has contributed for the Brazilian government plans to create and consolidate the competences for the development of high complexity Integrated Circuits in Brazil, and contributed a little to put Brazil in the international semiconductor industry scenario.

9.7 Future Works Directions

As it was shown along this thesis, several challenges were already identified to improve the proposed ISDB-T and DVB-S2 receivers, in order to make them achieve higher flexibility, support advanced configurations and work in environments that commercial applications are expected to deal with. Despite they were observed during the projects and some directions on how to tackle them were already discussed or defined, they are not the subject of this thesis. In order to reinforce the importance of addressing those open or under development issues, they are recalled hereafter.

Regarding the ISDB-T receiver, the main future works that are under study or development are: study of equalization and OFDM symbols boundary estimation algorithms to make the ISDB-T receiver work in SFN channels and extension of the full segment ISDB-T receiver to work in high mobility specially 16-QAM and 64-QAM modulations and Modes 2 and 3.

Regarding the DVB-S2 receiver, the main topics that are under development or study are: the selection and implementation of a baud rate estimator and its integration, study of the behavior of the current algorithms in NLOS channels, extension of the receiver to work in pilotless mode and extend the soft demapper to support 32-APSK. Furthermore, the integration of both receivers shall be addressed, in order to implement a hybrid ASIC for ISDB-T/DVB-S2 reception.

Despite the scope of this thesis being limited to provide solutions for implementation of ISDB-T and DVB-S2 receivers for Digital Television reception, the two main intrinsic subjects covered by the thesis, i.e. OFDM systems and satellite communications, along the main subject, i.e. Digital Television, are interesting areas of research. So, during the several studies performed along the two projects presented in this thesis, some general lines of research regarding those subjects were identified. Some of them are listed below by means of bullets indexes:

- DVB-S2X is a recently released amendment to increase the capabilities of DVB-

S2 standard. So, there are only a couple of solutions for DVB-S2X in the market (e.g. provided by Ericsson and Newtec), that still do not contain all functionalities contemplated by the DVB-S2X amendment. Therefore, there is still room for new players to provide DVB-S2 transmitters and receivers solutions. In the context of the DVB-S2 receiver project presented in this thesis, the algorithms used in the current DVB-S2 receiver shall be extended to work with DVB-S2X, and/or new algorithms shall be evaluated and proposed. Examples of work that shall be done are: 1) the evaluation of the algorithms in the claimed -10 dB of SNR that DVB-S2X is expected to work (the main algorithm to be evaluated is the PLS decoder); 2) the higher order modulations proposed by that standard shall have the adequate Soft Demappers; 3) The FEC Decoding subsystem of the current DVB-S2 receiver shall be extended to support the new fine grained FEC codes.

- Although Baud Rate estimation is a subject that has been explored for a while, it was not very well explored in this work. Furthermore, it is a subject that has gained interest recently due to its importance for the so called Cognitive Radio.

- Bringing mobility to HD-DTV with single antenna, specially for ISDB-T receivers is a subject that still has room for investigation. First, because there is no solution in the market for that, second that bringing high mobility for OFDM-based system is a well known area of interest for the current and emerging wireless communications systems. One of the main issues on this subject is bringing mobility to those systems while keeping receiver complexity at acceptable levels.

- Recently, the reception of satellite signals under mobility and in NLOS scenarios has gained interest. One of the subjects that has been studied is the so called Land to Satellite Mobile Channel. Several subjects are related to this theme: antennas, equalization, SISO and MIMO wireless channels modeling, physical layer improvements and others.

- A subject that has been researched recently is the use of OFDM for satellite communications. So, the proposal of new physical layers based on that transmission technique, e.g. adopting MIMO-OFDM approach can be a line of research aiming at to explore the NLOS scenarios. Due to its high PAPR, OFDM can drive the satellite amplifiers to the non-linear regions. So, the effects of non-linearities in the physical layers based on OFDM, as well as proposals on how to mitigate this effect in the context of satellite communications is a interesting area. The reception of OFDM signal, distributed via satellite, in high mobility is other possible area of research.

- An interesting area of research in satellite communications is the use the of DFE to reduce the non-linearity effects. It was shown in the literature the potential of using such approach [126], but further studies shall be done to increase the BER vs. SNR gains, specially for multilevel modulations such as 16-APSK, 32-APSK and those specified in DVB-S2X amendment.

- Due to crescent interest in land to mobile satellite channels, bringing mobility for DVB-S2 and DVB-S2X receivers in those channels is also an area where contributions can be done.
- The use of BCH decoders in conjunction with binary LDPC has become a very powerful FEC combination, that is currently used in several communications systems. Nevertheless, the use of non binary LDPC (NBLDPC) is growing in interest. So, several areas of research around this subject can be mentioned: the design of codes, reduction of the complexity of the algorithms used to decode NBLDPC, LLR computation for NBLDPC, the combination of MIMO with NBLDPC it is also an interesting area of research.

Appendix A

List of Publications

Patent

1. Eduardo R. de Lima, "Method and Apparatus for Channel Estimation and Equalization". US Patent 62115418 filled in 2015.

Journal Paper

1. Eduardo R. de Lima et al, "A 65nm CMOS ISDB-T VLSI Receiver Design: Architecture, Algorithms and Implementation", IEEE Transactions in Broadcasting *(submitted)*
2. Cesar G. Chaves, Eduardo R. de Lima and Jacqueline G. Mertes, *"A Synthesizable BCH Decoder for DVB-S2 Satellite Communications"*, (selected from WCAS2014 to be submitted to the Journal of Integrated Circuits and Systems (JICS)), SBMicro/SBC.
3. Andre N. Sapper, Eduardo R. de Lima, Cesar G. Chaves, Gabriel S. da Silva, Augusto F. R. Queiroz,Marcelo A. C. Fernandes, "Um Equalizador Adaptativo para Compensar os Efeitos da ISI em Sistemas de Comunicação DVB-S2", IEEE Latin American Transactions (to appear).

Peer Reviewed Conference Papers

1. Eduardo R. de Lima, Andre N. Sapper and Marcelo A. C. Fernandes, "An Adaptive Equalizer to Reduce the Nonlinear Distortion Effects in Satellite Broadcasting Systems," XXXIII Simpósio Brasileiro de Telecomunicações, Juiz de Fora, Brazil, September, 2015.

2. Eduardo R. de Lima, Augusto F. R. Queiroz, Denise C. Alves, Gabriel S. da Silva, Cesar G. Chaves, Jacqueline G. Mertes, Thiago M. Marson, "A Detailed DVB-S2 Receiver Implementation: FPGA Prototyping and Preliminary ASIC Resource Estimation", 6th LATINCOM - IEEE Latin-American Conference on Communications, Cartagena de Indias, Colombia, November, 2014.

3. Eduardo R. de Lima, Augusto F. R. Queiroz, Gabriel S. da Silva, Felipe A. M. Erazo, and Jose E. Bertuzzo, "Design and FPGA prototyping of a DVB-S2 receiver: Towards a VLSI implementation in CMOS", In proceeding of: 3rd Workshop on Circuits and Systems Design (WCAS 2013), Curitiba-PR-Brazil, September, 2013.

4. Gabriel S. da Silva, Augusto F. R. Queiroz, Eduardo R. de Lima and Cesar G. Chaves, "A Novel Fine Frequency Estimation Serial Architecture applied in Satellite Communications", ISCAS 2015 - International Symposium On Circuits and Systems, Lisboa, Portugal, May, 2015.

5. Andre N. Sapper, Eduardo R. de Lima, Cesar G. Chaves, Gabriel S. da Silva, Augusto F. R. Queiroz,Marcelo A. C. Fernandes, "An Adaptive Equalizer for Reliable Transmissions in DVB-S2 Satellite Communications Under ISI", 6th LATINCOM - IEEE Latin-American Conference on Communications, Cartagena de Indias, Colombia, November, 2014.

6. Denise C. Alves, Cesar G. Chaves, Eduardo R. de Lima, Gabriel S. da Silva and Augusto F. R. Queiroz, "FPGA Implementation of a FEC Decoding Subsystem for a DVB-S2 Receiver", IX Southern Programmable Logic Conference -SPL, Buenos Aires, Argentina, November, 2014. *(Best Paper Award)*

7. João A. B. R. Tardelli, Eduardo R. de Lima, Gabriel S. da Silva, Gerson Brito, Denise C. Alves, Cesar G. Chaves and Jacqueline G. Mertes, "Implementation of CORDIC-based Soft and Hard Demappers for a DVB-S2 Receiver", 4th Workshop on Circuits and Systems Design -WCAS, Aracaju, Sergipe, Brazil, September, 2014.

8. Cesar G. Chaves, Eduardo R. de Lima and Jacqueline G. Mertes, "A Synthesizable BCH Decoder for DVB-S2 Satellite Communications", 4th Workshop on Circuits and Systems Design -WCAS, Aracaju, Sergipe, Brazil, September, 2014. *(Best Paper Award)*

9. Denise C. Alves, Eduardo R. de Lima, and Jose E. Bertuzzo, "A pipelined semi-parallel LDPC Decoder architecture for DVB-S2", In proceeding of: 3rd

Workshop on Circuits and Systems Design (WCAS 2013), Curitiba, Paraná, Brazil, September, 2013.

10. Cesar G. Chaves, Eduardo R. de Lima, Jacqueline G. Mertes, and Jose E. Bertuzzo, "A Synthesizable Serial-in Syndrome Calculator for DVBS2 BCH Decoding", In proceeding of: 3rd Workshop on Circuits and Systems Design (WCAS 2013), Curitiba, Paraná, Brazil, September, 2013.

11. Eduardo R. de Lima et al, "ISDB-T receiver architecture and VLSI implementation in 65 nm CMOS, for Fixed-Reception high definition Digital Television",In Proceedings of 2012 IEEE International Conference on Consumer Electronics - Berlin (ICCE-Berlin), Berlin, September, 2012.

Invited

1. Eduardo R. de Lima, "Desenvolvimento de um Chipset Nacional para Recepção de Sinais ISDB-T: Lições Aprendidas", Congresso SET2010 - São Paulo, August, 2010.

Other Peer Reviewed Works

1. Eduardo R. de Lima, Denise C. Alves, Gabriel S. da Silva, Cesar G. C. Arroyave, Daniel Urdaneta, Tiago D. Perez, Augusto F. R. Queiroz, "Architecture design and implementation of key components of an OFDM transceiver for IEEE802.15.4g", IEEE International Symposium on Circuits and Systems - ISCAS 2016, Montreal, May, 2016 (accepted).

2. J. A. Bianco Filho, Eduardo R. de Lima, Cesar G. C. Arroyave, Luis G.P. Meloni, "Algorithm and Digital Circuit for Blind Frequency and Band Estimation in Spans with Multiple Signals", IEEE Computer Society Annual Symposium on VLSI - ISVLSI2016, Pittsburgh,July, 2016 (submitted).

3. Augusto F. R. Queiroz, Gabriel S. da Silva, Cesar G. A. Chaves, Tiago D. Perez, Daniel G. Urdaneta, Denise C. Alves, Maique C. Garcia, Eduardo R. de Lima, "Demo: FPGA Implementation of an IEEE802.15.4g MR-OFDM Baseband Modem for Smart Utility Networks", IEEE 4th Global Conference on Consumer Electronics (GCCE 2015), Osaka, October, 2015.

4. Augusto F. R. Queiroz, Jessica A. J. Oliveira, Eduardo R. de Lima, "A MR-FSK Transceiver Compliant to IEEE802.15.4g for Smart Metering Utility Applications: FPGA Implementation and ASIC Resource Estimation", 7th IEEE Latin-American Conference on Communications, Arequipa, Peru, 4-6 November, 2015.

5. Vanessa B. Olivatto, Eduardo R. de Lima, Renato R. Lopes, "Simplified LLR calculation for DVB-S2 LDPC decoder",2015 IEEE International Conference on Communication, Networks and Satellite (COMNETSAT 2015), Bandung , Indonesia, December 2015.

6. Denise C. Alves, Eduardo R. de Lima, "A Frame Synchronizer for IEEE 802.15.4-g MR-OFDM PHY: Algorithm Proposal and Hardware Implementation", 7th IEEE Latin-American Conference on Communications, Arequipa, Peru, November, 2015.

7. Daniel G. Urdaneta, Eduardo R. de Lima, Gabriel S. da Silva, Cesar G. A. Chaves, Jaqueline G.Mertes, Luis G. P. Meloni, "FPGA Implementation and ASIC Resource Estimation of an FFT/IFFT for an MR-OFDM Transceiver Complaint with IEEE802.15.4g", 5th Workshop on Circuits and Systems Design (WCAS 2015), Salvador, September, 2015.

8. Flavia C. da Costa, Eduardo R. de Lima, Ricardo T. Yoshioka, Jose .E. Bertuzzo, Jim Koeppe, "Impedance measurement of dipole antenna for EPC Global compliant RFID tag", In Proceedings of Microwave & Optoelectronics Conference (IMOC), 2013 SBMO/IEEE MTT-S International, Pages 1-5, Rio de Janeiro, Brazil, August, 2013.

9. Flavia C. da Costa, Ricardo. T. Yoshioka, Eduardo R. de Lima and F. I. L. Leite, "T-Matching Variation Effect of RFID Tag Antenna for 915 MHz",IEEE Brasil RFID 2014 Conference, São Paulo, SP, Brazil, September, 2014.

Appendix B

Digital Terrestrial Television Around The World

B.1 Digital Terrestrial Television Around The World

Digital Television is a pervasive technology. Most of the countries worldwide have already started DTV broadcasting or defined the standard they are going to adopt. A significant number of countries have already switched off the analog transmitters and another significant group of countries is planning to do so in the next years. Nevertheless, a small number of countries has not decided yet which standard they will adopt. This can be seen in the report presented in [150] where the author presents the status of DTV and compares the worldwide digital terrestrial broadcasting systems ATSC,DVB-T, and ISDB-T.

ISDB-T standard was already introduced in Chapter 3. Even so, due to the seminal importance of DVB-T and ATSC systems, an overview of the standards related to them is given, in this section. In addition, the most important standard for mobile TV (Media-FLO, DVB-H, ISDB-T(1-SEG) and CMMB) and the second generation of Digital Television(DVB-T2 and T-DMB) are also presented. Furthermore, with the eyes in the future, the concept of Dynamic Broadcast [151], that claims to save Radio Frequency Spectrum, and the Future of Broadcast Television Initiative (FOBTV) are introduced [152].

B.1.1 First Generation of Digital Television

In addition to ISDB-T and among the variety of DTV standards, capable of transmitting High Definition Digital Television, it is possible to point out other two of them: ATSC and DVB-T. In this section, they are described in the order which the respective standard was released. Independent of the particular characteristics of each one, it is worth mentioning that they play a very important role in the revolution that DTV is causing in the Television Broadcasting ecosystem.

B.1.1.1 Digital Video Broadcasting - Terrestrial (DVB-T)

DVB-T is based on COFDM and supports channels with 6, 7 and 8 MHz bandwidth. The system transmits compressed audio, video and other data in an MPEG-TS stream. The first edition of the standard was released by ETSI in 1997 as ETS 300744.

According to [2], a DVB-T system is defined as the functional block of equipment performing adaptation of the baseband TV signal from the output of the MPEG-2 transport multiplexer, to terrestrial channels characteristics. Figure B.1 presents a simplified view of this process.

B.1.1.1.1 Mux Adaptation and PRBS Scrambling

The first operation performed at the input of DVB-T modulator is to adapt the transport multiplex stream, originated in an MPEG-2 transport multiplexer, and to randomize it. The input stream is first organized in 188 bytes (including the sync word byte 47_{HEX}) fixed length packets, with the MSB of each byte transmitted first. Following, a randomization is performed using a PRBS with the period of 1503 bytes (*i.e.* 8 packets long minus 1 byte). This occurs because the first output bit of the *PRBS* shall be applied to the first bit of the first byte following the first *SYNC* byte with a 8 packets long frame. The first *SYNC* byte of a sequence of 8 shall be bit-wise inverted. Due to *MPEG-2* synchronization issues, the *PRBS* shall not be applied to the following 7 *SYNC* bytes, but it shall continue running during these bytes. The main goal of the randomizer is to guarantee adequate binary transitions, in other words to cause the number of bits zeros and ones to be the same (or approximately the same).

B.1.1.1.2 Reed Solomon Encoding

The next step, is to perform the Reed-Solomon(RS) encoding using an RS(204,188,t=8)

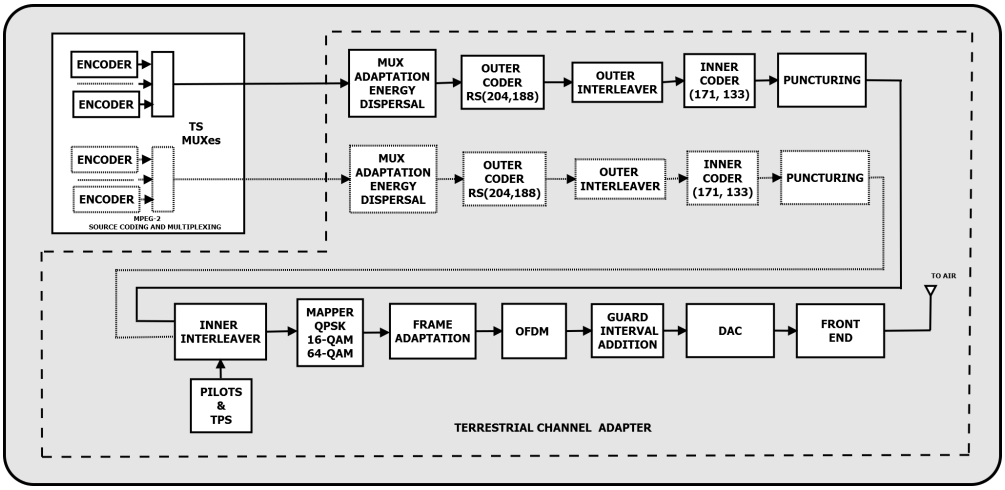


Figure B.1: The DVB-T Transmitter, According to the Standard [2]

encoder, which is a shortened version of RS(255,239,t=8) code. The encoding is performed in a packet basis. At the input of RS encoder the packet is 188 bytes long and at its output is 204 error-protected bytes long. The *Code Generator Polynomial* and the *Field Generator Polynomial* used in the RS encoder are shown in Equations B.1 and B.2 respectively:

$$g(x) = \left(x + \lambda^0\right)\left(x + \lambda^1\right)\left(x + \lambda^2\right)\cdots\left(x + \lambda^{15}\right) \tag{B.1}$$

$$p(x) = x^8 + x^4 + x^3 + x^2 + 1 \tag{B.2}$$

where, $\lambda \equiv 02_{HEX}$.

B.1.1.1.3 Byte-Wise Interleaving

As can be seen in Figure B.1 the next step in the transmit path is the Interleaver block. It comprises a byte-wise convolutional Interleaver with depth $I = 12$. This means that the incoming bytes will be split one by one among 12 branches (from 0 to 11), each one made of *FIFO* shift registers. The first (*i.e.* 0) has length equal to 0, the second 17 bytes, the third 17×2 bytes, the fourth 17×3 bytes and so on, until the last one that has 17×11 bytes. The Deinterleaver can be directly derived. Its first line of *FIFO* shift registers has length equal to 17×11 bytes and the last (i.e.

line 11) 0 *bytes*. *SYNC* byte (i.e. 47_{HEX}) can be inverted or not and shall be always routed in the branch 0.

B.1.1.1.4 Convolutional Encoding

After interleaving, the sequence of bits shall goe through the convolutional encoding process. The *Inner Coder* is based on a 1/2 rate *Convolutional Enconder* with 64 states. To achieve the other code rates determined by the standard, i.e. 2/3, 3/4 5/6 and 7/8, the output of the encoder shall be punctured. The generator polynomials of the *Inner Coder* are $G_1 = 171_{OCT}$ and $G_2 = 133_{OCT}$ for X and Y respectively.

B.1.1.1.5 Bit and Symbol Interleaving

Following the convolutional encoding the bit stream goes through the *Inner Interleaver* for $2K$ and $8K$ modes. There are optional in-depth symbol interleavers for $2K$ and $4K$ that are not going to be described here. Refer to Annex F of [2] for more details about those interleavers.

The inner interleaver is made of a bit-wise interleaver followed by a symbol interleaver. The interleaver input, which as can be seen in Figure B.1 and according to [2], consists of up to 2 bit streams[1]. In the case of non-hierarchical mode the single stream is demultiplexed into v sub-streams: $v = 2$ for *QPSK*, $v = 4$ for *16-QAM*, and $v = 6$ for *64-QAM*. In the case of 2 bit streams, they are demultiplexed in a slightly different manner, as follows:

Non-hierarchical Mode ($QPSK$, $16-QAM$ and $64-QAM$:

$$xdi = b[di(mod)v](div)(v/2) + 2[di(mod)(v/2)], di(div)v$$

Hierarchical Mode ($16-QAM$ and $64-QAM$ only:

[1] One for the high priority stream and the other for the low priority, in the case of DVB-T working in the hierarchical mode.

$$x^{'}di = bdi(mod)2, di(div)2$$

$$x^{''}di = b[di(mod)(v-2)](div)((v-2)/2) + 2[di(mod)((v-2)/2)] + 2, di(div)(v-2).$$

Where:

xdi	is the demultiplexer input stream for non-hierarchical mode;
$x^{'}di$	is the demultiplexer high priority input stream, for hierarchical mode;
$x^{''}di$	is the demultiplexer low priority input stream, for hierarchical mode;
di	is the input bit number;
$b_{e,do}$	is the demultiplexer output;
e	is the demultiplexed bit stream number ($0 \leq e < v$);
do	is the bit number of a given stream at the output of the demultiplexer;
mod	is the integer modulo operator;
div	is the integer division operator.

Each of the sub-stream (2 to 6) is processed by a different bit interleaver. There are up to 6 bit interleavers, labeled $I0$, $I1$, $I2$, $I3$, $I4$ and $I5$. $I0$ and $I1$ for *QPSK*, $I0$, $I3$ for *16-QAM* and $I0$ to $I5$ for *64-QAM*. Bit interleaver is only applied to useful data and the bit interleaver block size is 126 bits.

The input of each bit interleaver is defined by:

$$B(e) = (b_{e,0}, b_{e,1}, b_{e,2}, \cdots, b_{e,125})$$

where e ranges from 0 to 1.

The output vector of the bit interleaver, $A(e) = (a_{e,0}, a_{e,1}, a_{e,2}, \cdots, a_{e,125})$ is defined by:

$$a_{e,w} = b_{e,H_e(w)}$$

where:

$$w = 0, 1, 2, 3, \cdots, 125$$

and, $H_e(w)$ is a permutation function which is different for each bit interleaver, defined as:

$$I0: \quad H_0(w) = w$$

$$I1: \quad H_1(w) = (w + 63) mod 126$$

$$I2: \quad H_2(w) = (w + 105) mod 126$$

$$I3: \quad H_3(w) = (w + 42) mod 126$$

$$I4: \quad H_4(w) = (w + 21) mod 126$$

$$I5: \quad H_5(w) = (w + 84) mod 126$$

The v output streams of the bit interleaver are grouped(selecting one bit from each layer) in a bus, to form one digital symbol y' with $I0$ being the most significant bit:

$$y'_w = (a_{0,w}, a_{1,w}, a_{v-1,w})$$

As previously mentioned, the interleaver is made of a bit interleaver followed by a symbol interleaver. The goal of the symbol interleaver is to map the v bit words, sequentially read from the bit interleaver, onto the 1512 carriers for *2K mode* and 6048 carriers for *8K mode*. The symbol interleaver length is 1512 and 6048 for *2K mode* and *8K mode* respectively. For *2K mode* the interleaver input is:

$$Y' = (y'_0, y'_1, y'_2, \cdots, y'_{1511}).$$

In the *8K mode* case, the interleaver input is:

$$Y' = (y'_0, y'_1, y'_2, \cdots, y'_{6047}).$$

The interleaved output vector $Y = (y_0, y_1, y_2, \cdots, y_{N_{max}-1})$ is defined as:

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

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

where $N_{max} = 1512$ for *2K mode* and $N_{max} = 6048$ for *8K mode*. $H(q)$ is a permutation function defined according to the following:

1) An $(N_r - 1)$ bit binary word R'_i is defined, with $N_r = log_2 M_{max}$, where $M_{max} = 2048$ for *2K mode* and $M_{max} = 8192$ for *8K mode*, where R'_i takes the values:

$i = 0, 1$: $\quad R'_i[N_r - 2, N_r - 3, \cdots, 1, 0] = 0, 0, \cdots, 0, 0$

$i = 2$: $\quad R'_i[N_r - 2, N_r - 3, \cdots, 1, 0] = 0, 0, \cdots, 0, 1$

$2 < i < M_{max}$: $\quad R'_i[N_r - 3, N_r - 4, \cdots, 1, 0] = R'_{i-1}[N_r - 2, N_r - 3, \cdots, 2, 1]$

For the case of $2 < i < M_{max}$:

$$R'_i[9] = R'_{i-1}[0] \oplus R'_{i-1}[3], \text{ for } \textit{2K mode}.$$

$$R'_i[11] = R'_{i-1}[0] \oplus R'_{i-1}[1] \oplus R'_{i-1}[4] \oplus R'_{i-1}[6], \text{ for } \textit{4K mode}.$$

2) A vector R_i is derived from the vector R'_i by the permutations given Tables B.1 and B.2.

The permutation function $H(q)$ is defined by the following algorithm:

Table B.1: *2K mode* Bit Permutation.

R'_i bit positions	9	8	7	6	5	4	3	2	1	0
R_i bit positions	0	7	5	1	8	2	6	9	3	4

Table B.2: *8K mode* Bit Permutation.

R'_i bit positions	11	10	9	8	7	6	5	4	3	2	1	0
R_i bit positions	5	11	3	0	10	8	6	9	2	4	1	7

$$
\begin{gathered}
q = 0; \\
\text{for} \quad (i = 0; i < M_{max}; i = i + 1) \\
\{H(q) = (i mod 2).2^{N_r - 1} + \sum_{j=0}^{N_r - 2} R_i(j).2^j \\
\text{if} \quad (H(q) < N_{max}) q = q + 1; \}
\end{gathered}
\tag{B.3}
$$

As for y', y is made up of v bits:

$$y_{q'} = (y_{0,q'}, y_{1,q'}, \cdots, y_{v-1,q'})$$

where q' is the symbol number at output of symbol interleaver.

B.1.1.1.6 Constellation Mapping

DVB-T data subcarriers within an OFDM frame are modulated using, *QPSK*, *16-QAM*, *64-QAM*, non-uniform *16-QAM* or non-uniform *64-QAM*. They are mapped in a Gray mapping fashion. Nevertheless, a coefficient α forces a radial displacement of the original Gray constellation. The constellation that uses $\alpha \neq 1$ are called non-uniform. This stands for *16-QAM* and *64-QAM*.

The factor $\alpha = 2$ adds an offset of +1 to the positive values and -1 to the negative values of both axes Real and Imaginary. $\alpha = 4$ adds an offset of +3 and -3. $\alpha = 1$ does not cause any change in the original Gray constellations. The values of the complex constellations points, that have the form z =(n+jm), are presented below. In

addition, in order to achieve an average power equals to 1, the symbols, after being mapped, are scaled by a c factor also presented below:

QPSK

- n $\in \{-1, 1\}$, m $\in \{-1, 1\}$
- Bit ordering:
- $y_{0,q'}, y_{1,q'}$
- Real: $y_{0,q'}$
- Imaginary: $y_{1,q'}$
- $c = z/\sqrt{2}$

16-QAM - Non-hierarchical and Hierarchical With $\alpha = 1$

- n $\in \{-3, -1, 1, 3\}$, m $\in \{-3, -1, 1, 3\}$
- Bit ordering:
- $y_{0,q'}, y_{1,q'}, y_{2,q'}, y_{3,q'}$
- Real: $y_{0,q'}, y_{2,q'}$
- Imaginary: $y_{1,q'}, y_{3,q'}$
- $c = z/\sqrt{10}$

16-QAM - Non-uniform With $\alpha = 2$

- n $\in \{-4, -2, 2, 4\}$, m $\in \{-4, -2, 2, 4\}$
- Bit ordering: $y_{0,q'}, y_{1,q'}, y_{2,q'}, y_{3,q'}$
- Real: $y_{0,q'}, y_{2,q'}$
- Imaginary: $y_{1,q'}, y_{3,q'}$
- $c = z/\sqrt{20}$

16-QAM - Non-uniform With $\alpha = 4$

- n $\in \{-6, -4, 4, 6\}$, m $\in \{-6, -4, 4, 6\}$
- Bit ordering: $y_{0,q'}, y_{1,q'}, y_{2,q'}, y_{3,q'}$
- Real: $y_{0,q'}, y_{2,q'}$
- Imaginary: $y_{1,q'}, y_{3,q'}$
- $c = z/\sqrt{52}$

64-QAM - Non-hierarchical and Hierarchical With $\alpha = 1$

- n $\in \{-7, -5, -3, -1, 1, 3, 5, 7\}$, m $\in \{-7, -5, -3, -1, 1, 3, 5, 7\}$
- Bit ordering: $y_{0,q'}, y_{1,q'}, y_{2,q'}, y_{3,q'}, y_{4,q'}, y_{5,q'}$
- Real: $y_{0,q'}, y_{2,q'}, y_{4,q'}$
- Imaginary: $y_{1,q'}, y_{3,q'}, y_{5,q'}$
- $c = z/\sqrt{42}$

64-QAM - Non-uniform With $\alpha = 2$

- n $\in \{-8, -6, -4, -2, 2, 4, 6, 8\}$, m $\in \{-8, -6, -4, -2, 2, 4, 6, 8\}$
- Bit ordering: $y_{0,q'}, y_{1,q'}, y_{2,q'}, y_{3,q'}, y_{4,q'}, y_{5,q'}$
- Real: $y_{0,q'}, y_{2,q'}, y_{4,q'}$
- Imaginary: $y_{1,q'}, y_{3,q'}, y_{5,q'}$
- $c = z/\sqrt{60}$

64-QAM - Non-uniform With $\alpha = 4$

- n $\in \{-10, -8, -6, -4, 4, 6, 8, 10\}$, m $\in \{-10, -8, -6, -4, 4, 6, 8, 10\}$
- Bit ordering: $y_{0,q'}, y_{1,q'}, y_{2,q'}, y_{3,q'}, y_{4,q'}, y_{5,q'}$
- Real: $y_{0,q'}, y_{2,q'}, y_{4,q'}$
- Imaginary: $y_{1,q'}, y_{3,q'}, y_{5,q'}$
- $c = z/\sqrt{108}$

B.1.1.1.7 OFDM Frame Structure

DVB-T has 2 transmission modes called 2K and 8K. There is another mode, defined in [2] called 4K that is used only for DVB-H and will not be covered in this section. The standard,[2], define transmission parameters for channels with 8, 7, 6 and 5 MHz. Tables B.3, B.4, B.5 and B.6, present the main DVB-T parameters for 8, 7, 6 and 5 MHz channels respectively.

The frame rules are the same for any channel bandwidth. What differs for each case is the elementary IFFT sampling rate, which causes differences in: carrier space, symbol length, guard interval length and useful bit rate, as can be seen in Tables B.3, B.4, B.5 and B.6.

An OFDM symbol transports data, modulates in *QPSK*, *16-QAM* or *64-QAM*, scattered pilots, continual pilot carriers (that are boosted by a factor of 16/9) and *Transmission Parameter Signaling* (TPS). The main use of pilots are frame, time and frequency synchronization, as well as channel estimation and transmission mode identification. Figure B.2 shows a simplified view of the DVB-T Frame Structure. The values of scattered or continual pilot are derived from the PRBS generator that has the following polynomial :

$$x^{11} + x^2 + 1.$$

PRBS initialization word is [11111111111]. Scattered pilots are boosted by the factor of 4/3 and assume values +1.3333... for bit 0, and -1.3333...for bit 1. Continual pilot is boosted by the factor of 16/9. *8K mode* and *2K mode* have 45 and 177 continual pilot subcarriers, and 68 and 17 TPS subcarriers, respectively. TPS carries the following information: modulation, α value, hierarchy information, guard interval length, inner code rate, transmission mode, frame number in a super-frame, cell identification. TPS blocks is 68 bits long and carries the following payload: 1 initialization bit, 16 synchronization bits, 37 information bits (the last six bits of this field are set to zero) and 14 redundancy bits BCH code $(67, 53, t = 2)$. It is a shortened version of BCH $(127, 113, t = 2)$ code that has the following polynomial generator:

$$h(x) = x^{14} + x^9 + x^6 + x^5 + x^4 + x^2 + x + 1.$$

TPS blocks are *Differential Binary Phase Shift Keying* (DBPSK).

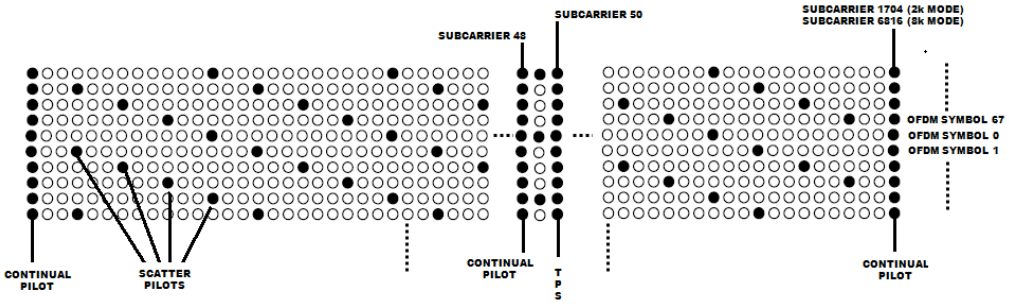


Figure B.2: Simplified View of DVB-T Frame Structure According to [2].

Table B.3: Main DVB-T Parameters for 2K and 8K Modes for 8 MHz Channels.

MODE	8K	2K
Bandwidth	7.61(MHz)	
Spacing Between Carriers Frequencies	1.116(kHz)	4.464(kHz)
Number of Carriers	6817	1705
OFDM Symbols per Frame	68	
OFDM Symbols per Super-Frame	4x68	
	224 us (1/4)	56 us(1/4)
Guard	112 us (1/8)	28 us (1/8)
Length	56 (1/16)	14 us(1/16)
	28 (1/32)	7 us (1/32)
Effective Symbol Length	896 us	224 us
	76.16 ms (1/4)	19.04 ms (1/4)
Frame	68.54 ms (1/8)	17.14 ms (1/8)
Length	64.74 ms (1/16)	16.18 ms (1/16)
	62.83 (1/32)	15.71 ms (1/32)
Inner Code	Convolutional Code (1/2, 2/3, 3/4 , 5/6, 7/8)	
Outer Code	Reed-Solomon(204,188)	
IFFT Length	8192	2048
IFFT Sampling Frequency	64/7 MHz = 9.142857... MHz	
Bit Rate Non-Hierarchical	4.98 to 31.67 Mbps	
Bit Rate Hierarchical QPSK/16-QAM	4.98 to 10.56 Mbps	
Bit Rate Hierarchical QPSK/64-QAM	4.98 to 21.11 Mbps	

Table B.4: Main DVB-T Parameters for 2K and 8K Modes for 7 MHz Channels.

MODE	8K	2K
Bandwidth	6.66(MHz)	
Spacing Between Carriers Frequencies	0.976563(kHz)	3.90625(kHz)
Number of Carriers	6817	1705
OFDM Symbols per Frame	68	
OFDM Symbols per Super-Frame	4x68	
	256 us (1/4)	64 us(1/4)
Guard	128 us (1/8)	32 us (1/8)
Length	64 (1/16)	16 us(1/16)
	32 (1/32)	8 us (1/32)
Effective Symbol Length	1024 us	256 us
	87.04 ms (1/4)	21.76 ms (1/4)
Frame	78.34 ms (1/8)	19.58 ms (1/8)
Length	73.98 ms (1/16)	18.50 ms (1/16)
	71.81 (1/32)	17.95 ms (1/32)
Inner Code	Convolutional Code (1/2, 2/3, 3/4 , 5/6, 7/8)	
Outer Code	Reed-Solomon(204,188)	
IFFT Length	8192	2048
IFFT Sampling Frequency	7/1 MHz = 7 MHz	
Bit Rate Non-Hierarchical	4.354 to 27.710 Mbps	

Table B.5: Main DVB-T Parameters for 2K and 8K Modes for 6 MHz Channels.

MODE	8K	2K
Bandwidth	5.71(MHz)	
Spacing Between Carriers Frequencies	0.837054(kHz)	3.348214(kHz)
Number of Carriers	6817	1705
OFDM Symbols per Frame	68	
OFDM Symbols per Super-Frame	4x68	
	298.667 us (1/4)	74.667 us(1/4)
Guard	149.333 us (1/8)	37.333 us (1/8)
Length	74.667 (1/16)	18.667 us(1/16)
	37.333 (1/32)	9.333 us (1/32)
Effective Symbol Length	1194.667 us	298.6667 us
	101.54 ms (1/4)	25.38 ms (1/4)
Frame	91.39 ms (1/8)	22.85 ms (1/8)
Length	86.31 ms (1/16)	21.58 ms (1/16)
	83.78 (1/32)	20.94 ms (1/32)
Inner Code	Convolutional Code (1/2, 2/3, 3/4 , 5/6, 7/8)	
Outer Code	Reed-Solomon(204,188)	
IFFT Length	8192	2048
IFFT Sampling Frequency	48/7 MHz = 6.8571428... MHz	
Bit Rate Non-Hierarchical	3.732 to 23.751 Mbps	

Table B.6: Main DVB-T Parameters for 2K and 8K Modes for 5 MHz Channels.

MODE	8K	2K
Bandwidth	4.75(MHz)	
Spacing Between Carriers Frequencies	697.545(kHz)	2790.179(kHz)
Number of Carriers	6817	1705
OFDM Symbols per Frame	68	
OFDM Symbols per Super-Frame	4x68	
	358.40 us (1/4)	448.00 us(1/4)
Guard	179.20 us (1/8)	403.2 us (1/8)
Length	89.60 (1/16)	1380.80 us(1/16)
	44.80 (1/32)	369.60 us (1/32)
Effective Symbol Length	1433.60 us	358.40 us
	121.86 ms (1/4)	30.46 ms (1/4)
Frame	109.67 ms (1/8)	27.42 ms (1/8)
Length	103.58 ms (1/16)	25.89 ms (1/16)
	100.53 (1/32)	25.13 ms (1/32)
Inner Code	Convolutional Code (1/2, 2/3, 3/4 , 5/6, 7/8)	
Outer Code	Reed-Solomon(204,188)	
IFFT Length	8192	2048
IFFT Sampling Frequency	40/7 MHz = 5.71428571... MHz	
Bit Rate Non-Hierarchical	3.110 to 19.793 Mbps	

B.1.1.2 ATSC - Advanced Television Systems Committee Standards

The ATSC standard related to the physical layer of the transmitter for *Terrestrial Digital Television Broadcasting* is [4]. This standard is basically used in the USA and in a small number of countries around the world. The terrestrial broadcast mode, is known as 8-VSB. This mode is known worldwide as ATSC. That standard defines the RF and transmission system characteristics for 6 MHz bandwidth channels. ATSC group was formed in 1982. Nevertheless the first version of the standard for *Terrestrial Digital Television Broadcasting* was published in 1995 by the Digital *HDTV Grand Alliance*, also known as *Grand Alliance*. According to [153], that name was suggested by the FCC Advisory Committee, with the aim of putting together the four digital HDTV proponents for the evolution of NTSC.

ASTC delivers MPEG-2-TS at approximately $T_r = 2 \times \left(\frac{188}{208}\right)\left(\frac{312}{313}\right)\left(\frac{684}{286}\right) =\approx 19.39\cdots \approx 19.39$ Mbps, in a 6 MHz channel. Compared to other standards like DVB-T and ISDB-T, the Physical layer of ATSC standard for DTV transmission is much simpler to be understood. This can be observed, by inspecting the functional block diagram shown in Figure B.3.

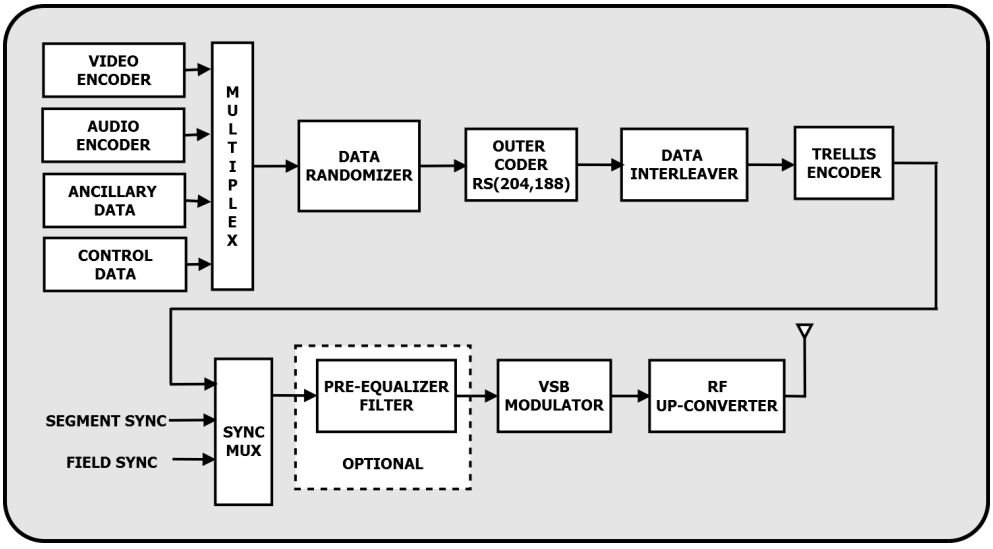


Figure B.3: Functional Block of Terrestrial Broadcast Mode (8-VSB) [4]

.

B.1.1.2.1 Transmission Data Organization

Figure B.4 shows the data organization for ATSC transmission. Two *Data Fields* form a *Data Frame*. Each *Data Field* contains 313 *Data Segments*. The first *Data Segment* o each *Data Field* is a unique synchronization signal, called *Data Field Sync*. The remaining 312 *Data Segments* carry the MPEG2-TS payload with RS correction. Each *Data Segment* carries 4 MPEG2-TS payload without the MPEG2-TS SYNC word (187 bytes), plus the RS FEC overhead (20 bytes), plus a four-bits word equivalent to the MPEG2-TS SYNC word, given a total of 832 symbols/*Data Segment*.

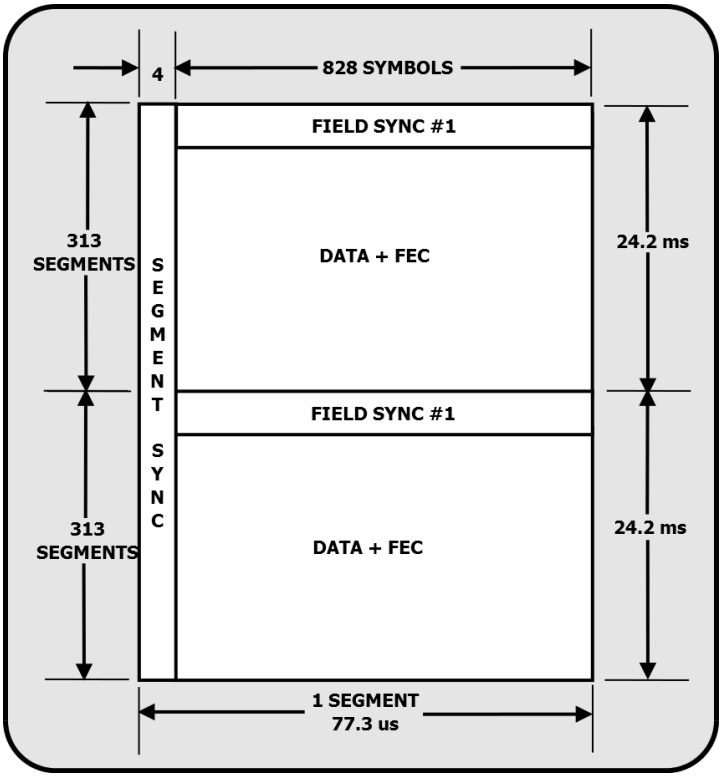


Figure B.4: VSB Data Frame Without Extra SYNC.

B.1.1.2.2 Data Randomizing

The data payload, from the multiplexer, and transport blocks are randomized by a PRBS with the following generator polynomial:

$$g(x) = x^{16} + x^{13} + x11 + x^7 + x^6 + x^3 + x + 1$$

that is initially loaded with the word [1111000110000000] at the beginning of the *Data Field*, during the *Data Segment Sync* interval prior to the first *Data Segment*. Figure B.5 shows how to generate the randomizer byte from the 16-bit PRBS . Each 8 input bits is randomized by the bytes $D^0D^1D^2D^3D^3D^4D^6D^7$ MSB first.

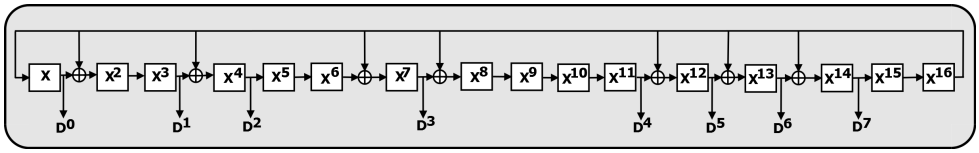


Figure B.5: Randomizer Polynomial.

B.1.1.2.3 Reed-Solomon Enconding

The output bytes of PRBS-based randomizer are the input of *RS Encoder*. The RS code for VSB transmission is $RS(207, 187, t = 10)$. Where 207 is the RS block size, and 187 is the data block size, that leads to 20 parity bits for error correction. The *Data Segment* is made of 207 bytes. The parity generator polynomial and the primitive field generator polynomial are shown in Equations B.4 and B.5 respectively.

$$g(x) = \left(x + \lambda^0\right)\left(x + \lambda^1\right)\left(x + \lambda^2\right)\cdots\left(x + \lambda^{19}\right) \tag{B.4}$$

$$p(x) = x^8 + x^4 + x^3 + x^2 + 1 \tag{B.5}$$

B.1.1.2.4 Convolutional Byte Interleaving

The interleaver used in ATSC has a depth of one-sixth of a*Data Field* i.e. $24.2/6ms \approx 4ms$. It is a 52-*data segment* convolutional byte interleaver. Only the 207 byte of the *Data Segment*, *i.e. Data Segment Sync* excluding the SYNC byte, shall be convolutionally interleaved. An illustration of the convolutional interleaver is shown in Figure B.6. The interleaver is synchronized with the first data byte of the *Data Field*, *i.e.* the first byte has zero-byte delay.

B.1.1.2.5 Trellis Coding and Intrasegment Symbol Interleaving

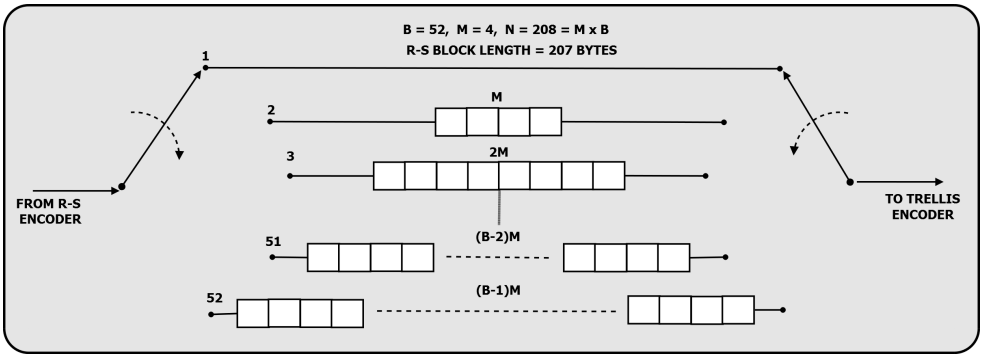


Figure B.6: Convolutional Byte Interleaver.

The *Trellis Coding* is made of twelve independent *Trellis Encoder*. Each byte outputted from the *Convolutional Byte Interleaver* feeds a single *Trellis Encoder*, that can be seen in Figure B.7 [1]. The output of the 12 encoders form a single 8-VSB baseband symbol stream, as can be seen in Figure B.8. Following, a brief description of the entire process shown in that figure, is given.

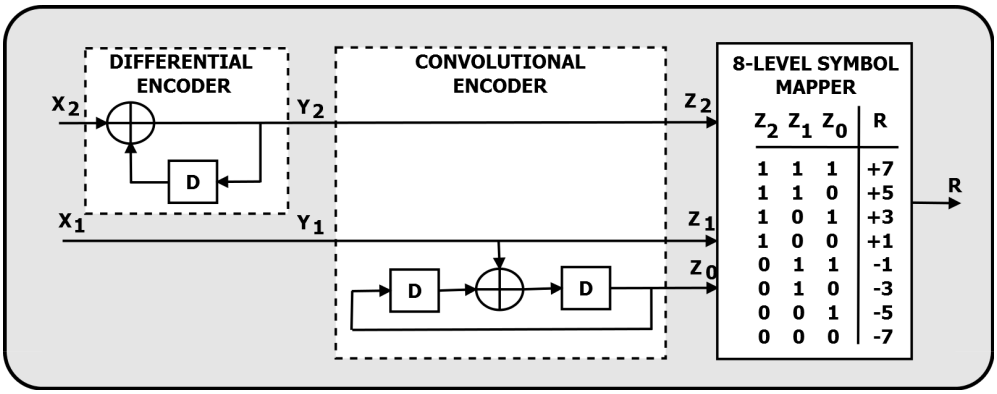


Figure B.7: Single Trellis Encoder. D = 12 Symbols Delay.

A sequence of bytes belonging to different MPEG2-TS (this is caused by the byte interleaver) packets, feeds in the order of arrival, a different branch (from 1 to 12). Initially, each one of theses bytes is stored in a 1 byte buffer. When the 13th byte arrives at 1-byte buffer belonging to the 1st branch, all twelve bytes buffered in the previous 1-byte buffers are transferred at once for a second column of 12 parallel 1-byte buffers. In each one of the branches, the bytes are split into four lines with 2 bits

[1] The dashed blocks, in Figure B.7, clocks 12 times faster than the mapper block

each line. Next, the bits are trellis encoded. Each 2-bits mathematically generates 3 bits at the encoder output. This can be seen in Figure B.7. Each 3-bits generate one 8-VSB baseband symbol. For this reason, this encoder is said to be a 2/3 rate coder. Finally, all baseband symbols are read one-by-one to form a single stream of 8-VSB baseband symbols.

The entire *Trellis Coding* and *Intrasegment Symbol Interleaving* shown in Figure B.8 can be interpreted in terms of a hypothetical input clock named F_{clock} as follows. Lets assume that the arriving bytes at input of the 12-output demultiplexer are read at a frequency of F_{clock} in Hz. The output of the first and second columns of 1-byte buffers are read at $F_{clock}/12$. One output of each one of the byte-to-bit pairs blocks is read at $4F_{clock}$. Trellis encoder blocks outputs are read at $4F_{clock}$. Nevertheless, each single *Trellis Encoder* has two clock domains internally. As can be seen in Figure B.7. The input clock is $4F_{clock}$, the dashed blocks works at $48F_{clock}$ and the mapper input/output works at $4F_{clock}$. Which means that the output clock of the *Trellis Coder Interleaver* is 4 times its input clock.

The input demultiplexer shown in Figure B.8 sends the first byte of the first *Data Segment* of each *Data Field* to trellis encoder number 0. Successive input bytes of a given *Data Segment* shall be sent to successive *Trellis Encoders*. The data output from the input demultiplexer follows normal ordering from encoder 0 through 11 (and repeating) for the first *Data Segment* of the frame.

The input demultiplexer shall advance by four positions on each segment boundary, due to the *Data Segment Sync* interval. As a result, the first byte of the second *Data Segment* is sent to trellis encoder number 4. The states of the trellis encoders shall not advance during the *Data Segment Sync* interval.

The output multiplexer shown in Figure B.8 select the output of trellis encoder number 0 for the first symbol of the first *Data Segment* in a *Data Field*. The output multiplexer shall advance one position per output symbol until a total of 828 symbols (corresponding to one *Data Segment*) has been output. The output multiplexer shall advance by four positions on each segment boundary,corresponding to the *Data Segment Sync* interval.

Due to the advance of the input demultiplexer during the four bit corresponding to the *Data Segment Sync* the first byte of the second segment is read by the 1-byte buffer of the branch number 4 instead 0. Due to the same fact, the order of the trellis encoder reading changes as well. The first to be read is the trellis encoder number 4, instead of 1. It is easy to extrapolate what is going to happen for the next segments until the end of the frame. Nevertheless, it is worth highlighting that *Trellis Encoder* states do not change during the advance of the output selector during the *Data Segment Sync*. Also, the 4 symbols advance during the *Data Segment Sync* will cause the occurrence of a three-segment pattern that repeats 104 times throughout the 312 *Data Segments* of each Data Field.

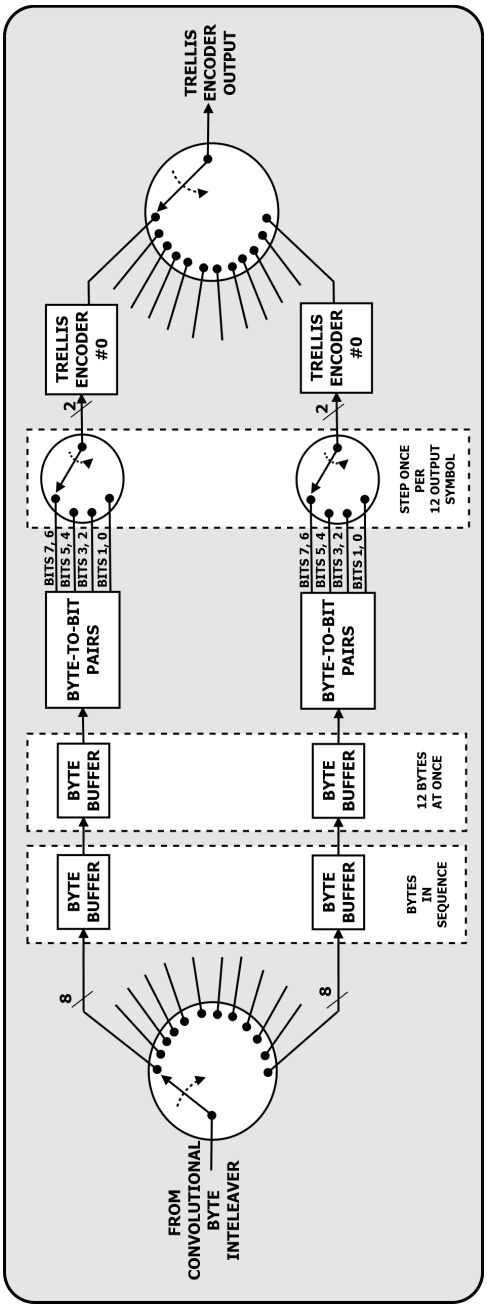


Figure B.8: Trellis Code Interleaver.

B.1.1.2.6 Data Segment Sync

ASTC 8-VSB synchronization comprises *Data Segment* synchronization and *Data Field* synchronization.

After trellis encoding, the data passes through the *Sync Mux* (see Figure B.3, that inserts the *Data Segment Sync* and *Data Field Sync*. *Data Segment Sync* is inserted into the 8-level data stream. The four-symbol sequence consists of the pattern [+5, -5 -5 +5]. Recall that *Data Segment Sync* pattern symbols are not RS or trellis encoded, nor are they interleaved.

B.1.1.2.7 Data Field Sync

The first *Data Segment* of each *Data Field* is the *Data Field Sync* which has the structure shown in Figure B.9.

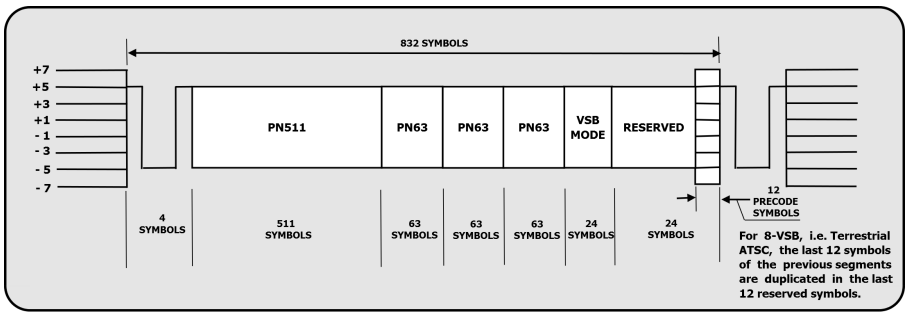


Figure B.9: VSB Data Field Sync.

PN511 Sequence is generated from the *Polynomial Generator* represented in Equation B.6. The pre-load value is [0 1 0 0 0 0 0 0 0]. At output, the binary value 1 is mapped to +5 and the bit 0 to -5.

$$p(x) = x^9 + x^7 + x^6 + x^4 + x^3 + x + 1 \tag{B.6}$$

Each *PN63 Sequence* is generated from the *Polynomial Generator* represented in Equation B.7. The pre-load value is [1 0 0 1 1 1]. At output, the binary value 1 is mapped to +5 and the bit 0 to -5.

$$p(x) = x^6 + x + 1 \tag{B.7}$$

The sequence of 24 *VSB Mode* symbols, in the case of Terrestrial ATSC, i.e. 8-VSB trellis encoded, is [0000 1010 0101 1111 0101 1010]. As in the case of PN sequences 511 and 63, the binary value 1 is mapped to +5 and the bit 0 to -5. There are 7 others bit patterns for *VSB Mode*. Nevertheless, as they do not refer to Terrestrial ATSC, they are not going to be shown here. Refer to [4], section 5.3.2.1, for more information on that subject.

B.1.1.2.8 VSB Modulation

After bit-to-symbol mapping, a pilot is added at the suppressed carrier frequency i.e. at 309 kHz. This is done by means of the addition of a DC offset of 1.25 to every 8-level baseband symbol. Next, the signal is suppressed carrier modulated to an Intermediate Frequency and then upconverted to RF. Next, the signal is amplified and transmitted. Figure B.10 show the 8-VSB spectrum of ATSC at an IF of 20 MHz.

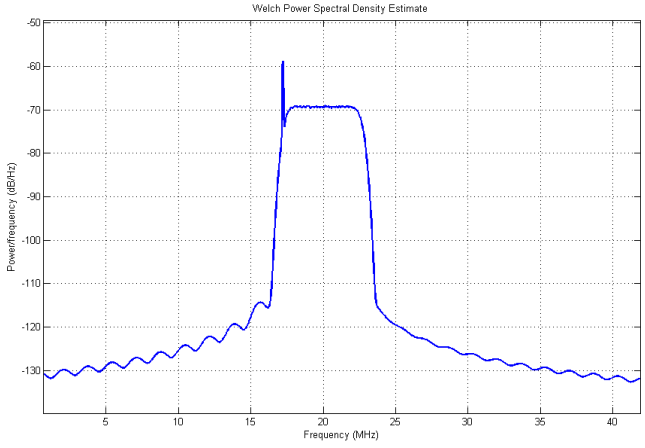


Figure B.10: Matlab Generated ATSC 8-VSB Signal Spectrum: 20 MHz IF, Sampling Frequency = 80 MHz.

B.2 Dedicated Mobile Digital Television Standards

With the advent of Terrestrial Digital Television, the broadcast community realized the potential usage of Digital Television into mobile and portable devices as cell phones, handheld devices and notebooks. Currently, the main standards in the market targeting this type of devices are DVB-H, T-DMB, CMBB and 1-SEG. Nevertheless, it is worth noting that an expected event has occurred in the roadmap of Digital Television targeting small resolution screen devices: the birth of analog TV targeting low-end mobile devices. This was possible due to the fact that there is still a large market for this, especially because several countries around the world still have analog transmitters broadcasting and will keep them for a while. Several fabless companies implemented chips for this market, examples of which are Telegent System (the pioneer that was acquired by Spreadtrum in the 2011), MediaTek, RDA Microelectronics, Newport Media. Nevertheless it is quite clear that this a market niche and will shortly shrink until down to a predictable end. In the following sections, the main standards for Digital Mobile TV are depicted.

B.2.1 DVB Transmission System for Handheld Devices (DVB-H)

DVB-H is an amendment for DVB-T standard targeting power-limited mobile devices, i.e. handhelds. According to [154], the following are the additions made to DVB-T standard for DVB-H target devices:

- in-depth symbol interleaver for 2K (2048 points FFT) and 4K (4096 points FFT) operation modes to improve robustness in mobile and impulsive noise environment;
- 4K mode, to achieve balance between SFN cell size and mobility(medium size cell and high speed);
- special DVB-H signaling in the TPS-bits to enhance and speed up service discovery;
- addition of time-slicing to reduce the average power consumption of the terminal and seamless frequency handoff;
- *Forward Error Correction for Multiprotocol Encapsulated Data* (MPE-FEC) to improve C/N and doppler performance, and to improve tolerance to impulsive noise.

Figure B.11 shows a simplified view of a DVB-T system with DVB-H capabilities. In that figure is possible to see the DVB-H additions to the DVB-T standard:

4K mode, *MPE-FEC*, *MPE*, *TIME SLICING* and *DVB-H TPS* signaling. In the following sections I describe these elements.

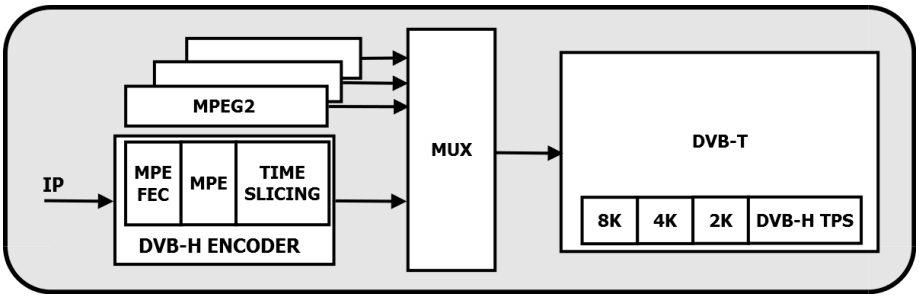


Figure B.11: DVB-H System View.

B.2.1.1 Time-Slicing

Time-slicing consists of sending the data in bursts in high instantaneous bit-rates. There is a parameter *delta-t*, that is transmitted within the burst to indicate when the next burst is going to be transmitted. Time-slicing allows the receiver to be active only a fraction of the time while receiving the requested data. During the time DVB-H receiver is not receiving a burst, it can monitor other cells. This allows the receiver to perform handover. Time-slicing is always used in DVB-H.

B.2.1.2 MPE-FEC

Multiprotocol Encapsulation Forward Error Correction is obtained through the addition of FEC at MPE layer. Parity information, which is calculated from the datagram, is sent in a separate MPE-FEC section. According to the DVB-H amendment MPE-FEC is not mandatory.

B.2.1.3 4K mode and in-depth interleavers

According to the DVB-H amendment, [154], the aim of the 4K mode is to improve network planning flexibility by trading off mobility and SFN size. To further improve robustness of the DVB-T 2K and 4K modes in a mobile environment and impulse noise reception conditions, an in-depth symbol interleaver is also standardized. The additional 4K transmission mode is a scaled set of the parameters defined for the 2K and 8K transmission modes.

B.2.1.4 DVB-H signaling

DVB-H signaling aims to render the receiver access to the signaling more robust. Therefore, it speeds up the service discovery, allowing the demodulator to TPS-lock at low C/N. This provides a faster way to access signaling than demodulating and decoding the Service Information (SI) or the MPE-section header. DVB-H TPS indicates the presence of *Time-Slicing* and *MPE-FEC*.

B.2.2 ISDB-T(1-SEG)

1-SEG information (audio, video and interactivity data) is transmitted within the *OFDM Symbols* that transports ISDB-T segments targeting terrestrial fixed-reception and very low mobility receivers. It allows the reception of low bit rate video, targeting low screen resolution devices, as cell phones, in high speed, without any extra signal processing.

1-SEG stands for 1 OFDM Segment of ISDB-T. It is the central segment out of the 13 that form the ISDB-T *OFDM Symbol*. It can use in-depth time interleaving. One advantage of 1-SEG, when compare it to ISDB-T for fixed-reception, is that the size of the FFT at the receiver can be much smaller than the one used to recover the information in the remaining 12 OFDM segments devoted to high resolution TV devices. Usually, when targeting high speed devices, the carriers modulation for 1-SEG is *QPSK*. A typical configuration, for 1-SEG, used by the broadcasters is: Mode 3, *QPSK*, *Code Rate* 1/3, GI 1/4 and *Time Interleaving* depth 4. The other possible parameter for 1-SEG can be found in Table 3.1. Despite the fact that broadcasters, in general, use that configuration, the best performance under Doppler conditions is achieved using *Mode 1*, which has the highest distance between subcarriers. Nevertheless, in general, the broadcasters of ISDB-T focus on fixed-reception high definition devices and use Mode 3 that allows better performance in wireless channel with large *Delay Spread*, as in the case of SFNs and regions covered by the main transmitter and repeaters at the same time.

B.2.3 Terrestrial Digital Multimedia Broadcasting (T-DMB)

T-DMB is a Korean standard based on the European project Eureka-147 DAB(Digital Audio Broadcasting) system. This standard incorporates H.264 video encoding and more robust protection code based on RS(204, 188, t = 8) Reed-Solomon encoder plus a *Convolutional Interleaver*. It targets mobile, portable and fixed devices. T-DMB is intended to be used not only in Korea, but in Europe, Pacific Asia and other regions.

T-DMB handles audio, video and data services. It works with 1.536 MHZ bandwidth DAB channels and can be used in VHF, UHF and L Band. As the DAB stan-

dard, T-DMB uses OFDM as transmission technique, carrying DQPSK modulated subcarriers.

B.2.4 Chinese Mobile Multimedia Broadcasting - CMMB

This mobile television and multimedia standard was developed for the Chinese market and is based on the *Satellite and Terrestrial Interactive Multiservice Infrastructure* (STiMi) chinese technology. CMBB is intended to be used in low screen resolution devices as smartphones, GPS and PDAs.

CMBB uses a robust protection code made of an RS encoder concatenated with LDPC, and two interleavers and H.264 video compression . It is based on OFDM (4K points FFT for 8 MHZ channels and 1K point FFT for 2 MHz channels) and carries BPSK, QPSK and 16QAM modulated subcarriers. It supports SFN (Single Frequency Networks). CMBB can be transmitted over satellite S-Band or terrestrial UHF. In both cases it uses OFDM as transmission technique.

B.3 Second Generation Digital Television and a Little About the Future of DTV

Currently, among the first generation standards for *DTV* devoted to terrestrial broadcasting, the only which presented a concrete evolution for the Physical Layer was DVB-T, the so called *DVB-T2*. Nevertheless, there is a group that considers *SBTVD* as an evolution of ISDB-T. However, as mentioned in the Introduction of this thesis, the *Physical Layer* of both *SBTVD* and *ISDB-T* are the same. Of course, the SBTVD made some contributions that improved the ISDB-T system, e.g. the Middleware GINGA and the adoption of H.264. So, in this section I describe the DVB-T2 features and discuss some efforts towards the future of DTV around the world.

B.3.1 DVB-T2

DVB-T2 is the ETSI standard proposed as the evolution of DVB-S2. DVB forum, claims DVB-T2 to be the world's most advanced Digital Terrestrial TV (DTT) System [155]. According to [156], the commercial requirements of DVB-T2 system are:

- Reuse of both existing domestic receiving antenna installations and existing transmitter infrastructures This requirement eliminated the possibility of using MIMO techniques as Spatial Multiplexing;
- A minimum of 30 percent throughput increase over DVB-T, working within the same planning constraints and conditions as DVB-T;
- Improved SFN performance compared with DVB-T;
- Service-specific robustness, e.g. to offer, within a single 8 MHz channel, some services for rooftop reception and other services for portable reception;
- Flexibility in bandwidth and frequency allocation within the UHF and VHF bands, but with additional features and/or modes to support higher frequency bands;
- A mechanism to reduce the peak-to-average-power ratio of the transmitted signal in order to reduce transmission costs.

According to [156] and [157], some of the DVB-T2 ETSI standard, [158], features are:

- FFT lengths of 1K, 2K, 4K, 8K, 16K and 32K;
- Guard Intervals of 1/128, 1/32, 1/16, 19/256, 1/8, 19/128 and 1/4;
- The same FEC used in the DVB-S2 system [77] (BCH code concatenated to LDPC) adding the 256-quadrature amplitude modulation (QAM) constellation, to take full advantage of the efficiency of the error-correction technique, and introducing a concept called rotated constellation, which can significantly improve the system performance in frequency selective terrestrial channels;
- Code rates 1/2, 3/5, 2/3,3/4, 4/5 and 5/6 with block lengths 64800 and 16200;
- Channel bandwidth of 1.7, 5, 6, 7 8 and 10 MHz;
- An extended-carrier mode to allow optimum use of the channel bandwidth together with the higher FFT sizes. When this option is used (supported for 8K, 16K and 32K FFT) the carrier spacing is the same as when the normal carrier is used, but additional carriers are added at both ends of the spectrum;
- Two separate mechanisms for reducing the *Peak-to-Average-Power Ratio* (PAPR) of the transmitted signal;
- The application of transmit diversity (MISO) to increase performance especially in single frequency networks. This is an optional mechanism, based on the well known Alamouti scheme [11];

- A very flexible frame structure that contains a special (short) identification symbol, which can be used for rapid channel scanning and signal acquisition, and which also signals some basic frame-structure parameters;
- Rotated constellations, which provide a form of modulation diversity, to assist in the reception of higher-code-rate signals in demanding transmission channels;
- Special techniques to reduce the PAPR of the transmitted signal;
- A method of transporting individual data services in separate logical channels, called *Physical Layer Pipes* (PLPs);
- Time interleaver of at least 70 ms for high-data-rate services;
- An option for extending the transmitted signal by including provision for *Future-Extension Frames* (FEFs), which are unspecified portions of the signal that first-generation receivers will know to ignore, but which could provide a compatible route for later upgrades.

B.3.2 The Future of Terrestrial DTV

Despite the efforts and contributions of the research community, industry, and many groups, as *DIBEG*, *DVB*, *SBTVD forum*, standardization bureaus as ARIB and ETSI, and many others, it is observed nowadays a myriad of standards, targeting *DTV* broadcasting, popping up around the world. One bad result of this almost endless number of standards is the MediaFLO technology (from Qualcomm), that after almost 10 years had their services discontinued in the *USA*.

There is an initiative named *FOBTV* [152], that is aimed at developing technologies for next-generation terrestrial broadcasting systems and making recommendations to standardization organizations around the world. According to [152], the goals of *FOBTV* association are the following:

- Develop future ecosystem models for terrestrial broadcasting taking into account business, regulatory and technical environments;
- Develop requirements for next generation terrestrial broadcast systems;
- Foster collaboration of *DTV* development, laboratories;
- Recommend major technologies to be used as the basis for new standards;
- Request standardization of selected technologies by appropriate standards development organizations;

The association was founded in 2011 and has representatives of different areas such as, broadcasters, manufacturers, network operators, standardization organizations, R&D centers and others, in more than 20 countries. Some of the members that founded *FOBTV* are *DVB*, *ATSC*, *NHK*, *IEEE*, *ETSI*, *ARIB*, *SET* and *NAB*.

Appendix C

Semiconductor Area in Brazil

C.1 An Historical View of Semiconductor Area in Brazil

According to [159] and references therein, since 1963 Brazilian universities have researched on the semiconductor area and, in 1966, industrial activities on semiconductor started at Philco, with the assembling of a plant to produce transistors and diodes in São Paulo. After that landmark several R&D and industrial activities started to appear. From 1968 to the mid -80's five laboratories were assembled in five different universities with emphasis on the research on silicon-based microelectronics. On the other hand, other laboratories intended to research semiconductor devices (e.g. laser) not based on silicon were also assembled and are still active nowadays. Furthermore, several Research Institutes not belonging to universities also joined the first efforts on semiconductors in Brazil. Despite all the efforts to assemble laboratories, creating groups and training researchers and other classes of workers with strong expertise in the semiconductors area, within Universities and R&D centers, some of the activities were not completed or discontinued. One example of that extinction is the cease of optical and microelectronics activities at the Brazilian lead center in research in telecommunications, CPqD.

In the industrial branch, following Philco's initial activities, the plant was transferred, in the 1970's to another state, in the form of a joint venture between Philco and RCA. In that new plant they started the production of linear bipolar Integrated Circuits (ICs). But, in 1984 the plant was closed and sold to another group that decided to stop the diffusion of semiconductor components in 1996, limiting the operations to assembling and packaging. Which were discontinued in 2000 [159].

In 1974 a new company called Transit was introduced to the semiconductor scene, with focus on diodes and transistor production using technology transferred from the Microelectronics Laboratory of the Technical School of University of São Paulo (LME-EPUSP). The production started in 1976, and in 1978, Transit started using SGS-Ates(from Italy) technology. The company did not survive and closed their doors in the end of the 1970's. Two other international companies (Icotron and Semicron) also joined the scene to produce power diodes and thyristors. Another company (AEGIS) was created in 1982 to produce the same components. Other examples of companies that had facilities in Brazil, but closed the business operations in the middle of the 1990's are: Texas Instruments, Fairchild, Philips, Sanyo and Rhom. Other Brazilian companies that are worth mentioning are ASA Microelectronics and ASGA Microelectronics. The first was dedicated to assemble and package diode LEDs and close the business in the end middle of the 1990's. The second, started business in the beginning of the 1990's. The last Brazilian company to be mentioned is Heliodinâmica, that was created in 1984 to produce silicon dowels and fotocells [159].

The basis for the establishment of the Brazilian electronic industry started in the 1960's and got stronger in the 1970's with the fast expansion of the consumer goods market. In the beginning of the 1980's, under the protection of the government Informatics Policies, there were about 23 electronic component industries in operation in Brazil. Within that time, several efforts were done by the Academy, R&D centers and industry to strengthen the semiconductor area. Nevertheless, in between 1980 and 2000 the industry suffered from the stagnation and a strong crash occurred in in the beginning of the 1990's due to the end the protective policies existent until that time. According to the Brazilian Bank for Social and Economic Development (BNDS) (refer to [14]), in the end of the 1990's Brazil had a strong commercial deficit due to electronic components, which was in the order of several billions of US dollar. Therefore, it was clear for the government that the creation of a Foundry in Brazil could solve part of that imbalance and also promote the economical development in the area. Nevertheless, in order to put this idea into practice the country had to create the basis for that. The ability to conceive and design Integrated Circuits and the capacity to produce and package IC's are paramount for the development of the Microelectronics industry in the country.

C.1.1 The National Program on Microelectronics - PNM

According to [14] and references therein, in 2001 there were 58 professors holding PhD degree working in ICs project and CAD tools, in Brazil. 55 of which in 14 public institutions, and the remaining 3 from a single private institution. The estimated number of graduate students produced by these institutions was 48 MSCs/year and 17 PhDs/years. In 2002, a more in-depth study showed that there were 108 Research groups on microelectronics related themes (including IC projects and CAD tools), distributed among 38 universities and research centers, working on 411 different research

topics. The total of researchers and technicians working directly in these groups was 457. Brazil's population in 2002 was of about 170 million spread over 8.515.767 square kilometers and the PNM started in that same year.

The microelectronics industry is strongly R&D based. Therefore, the country that expects to succeed in this area should have plenty of highly skilled manpower on this subject and related areas. In order to exemplify this challenge, it is worth mentioning the Taiwanese microelectronics industry that started in 1980 and, in 1995, there were 13 semiconductor companies with fabrication facilities. According to [160], in 1995, among the 28 private universities of Taiwan the number of professors with specialization in microelectronics was 300. There were 2.000 full time graduate students on electrical engineering doing research in microelectronics. In 1983, Taiwan produced only 7 PhDs, but in 1994 it increased to 136. In the same period, the number of MSEE graduate grew from 254 to 1024, BSEE's grew from 2290 in 1983 to 3426 in 1995, and about 20.000 EE technicians were formed. In 1994, Taiwan's population was 20 million spread over 36000 square kilometers [160].

As previously mentioned, the microelectronics was born in Brazil in the 1960's and had a strong development throughout the 1970's. Within that period, several efforts were made by the Universitites, R&D centers and industry to strengthen the semiconductor and microelectronics area, and it is possible to assert that until the beginning of the 1980's, the local research community and some private companies were up-to-date in several subfields of the semiconductors area. Nevertheless, between 1980 and 2000 the industry suffered from the stagnation and was almost totally dismantled. Several workers migrated to other areas, some left the country and most of the effort to generate high skilled manpower was lost. For this reason, new efforts should be made by the country to face the enormous gap in the technological and financial (due the huge amount of money necessary to install foundries) fronts and to form manpower. So the question that was rose was: is it worth doing this?

After several studies funded by the government, a new plan based on the triad Design Houses, Foundries and Packaging/Testing, was announced. This plan is called the National Program on Microelectronics - PNM. The institutions responsible to manage the plan are MCTI (Brazilian Ministry of Science, Technology and Innovation (former MCT), FINEP, BNDES, MDCI (Ministry of Commerce and Industry), and academic and business representatives.

In December 2002 MCTI released a document (see [14]) containing a draft on what should be the basis for the PNM. The document does not contain a conclusion, but delivered several recommendations on how to overcome the challenges and have the benefit of the opportunities of introducing Brazil in the world chain of Design Houses, Foundries and Package/Testing. Following I describe briefly the opportunities mentioned in [**?**].

C.1.1.1 Design Houses

Clearly, the Design House (DH) segment turned out to be the most attractive for the Brazilian PNM due to the low cost of creating a DH, when compared to Foundries and Packaging/Test facilities. Nevertheless, due to the lack of local manpower, a solid plan for training Digital, Analog and Mixed-Signal IC Designers was implemented. This effort is known as the IC-Brazil Program [15]. Despite the regular training provided by local universities, two IC Design training centers were created under the supervision of the IC-Brazil Program with funds of MCTI. These two centers receives student from Brazil and South America Countries. All the students on this program receive grants offered by the MCTI. There are three phases: formal training and oriented project within the training centers and a third phase offered by a DH working in an industrial project. Only the best students remain in the third phase of the training program. Some DHs got fund from specific open calls for IC design projects. In this way they could hire the trainees after the third phase. According to the IC-Brazil website, in the beginning of 2013 the Program had 22 DHs that were able to receive students for the third phase of the training. In the end of 2014 a new training center, named CT3, was created in São Paulo city.

Since 2006 an expressive number of IC designers have finished the training period and a significant number of them are working on the existent design houses or following Masters or PHDs programs at local universities. Part of those trainees decided to go overseas for professional activities or graduate programs on Microelectronics.

C.1.1.2 Foundries

The Foundry segment is the long term PNM goal. This segment is of paramount importance for a country that want to have citizens and companies that posses skills in the entire cycle of semiconductors production. However a huge amount of time and money are necessary to assemble a facility and stabilize the production process. Nevertheless, using government funds and equipment donated by Freescale (former Motorola Semiconductors) a foundry (focused on prototyping and low volume production) was created in 2000 and inaugurated in 2008. This is a government company called CEITEC and in the beginning of 2013 it is still not running on its full expected capacity. Later, in 2012 after years of negotiation the creation a company called SIX Semiconductors was announced. A partnership between IBM, BNDES, BDMG (Bank for the Development of Minas Gerais State), a Brazilian private holding (EBX), and a private fund to build a foundry, to cover local needs but with the eyes in the worldwide market. The infrastructure of the foundry is in an advanced stage but the business has already started. The beginning of the operation is foreseen for September 2015. In the end of 2014 the EBX was substituted by the in the venture by the Argentine company Unitec Blue. The current name was also changed to Unitec Semicondutores.

C.1.1.3 Package and Test

According to [14], due to the huge local market, Package and Test would be the segment that would lead to the highest volume of capital gain in short term, especially for the Memory area for computers and cell phones. Nevertheless, in the beginning of 2013 the only company that had operations in Brazil in this area was SMART Modular that opened a plant with that purpose in 2006. The remaining operations of the Brazilian company Itaucom in this area were discontinued in 2004 according to [15].

C.1.2 What PNM Has Achieved So Far

The outcomes until May 2013 are: one government foundry installed, one private foundry under construction, one fully operational private memory package company and 22 active design houses , running projects on Analog, Digital, Mixed-Signal and RF. Regarding IC designs results, the reports are not publicly available and it is not possible to show the ICs designed by the DHs. Nevertheless, the receiver described in this work is considered to be the largest and most complex IC ever designed in Brazil.

Bibliography

[1] ARIB Standard STD B-31. Transmission System For Digital Terrestrial Television Broadcasting, 2005. v, ix, xiii, xxvii, xxxv, 54, 55, 57, 58, 60, 66, 73, 78, 87, 91, 132, 146, 202

[2] ETSI Standard - EN 300 744. Digital Video Broadcasting (DVB); Framing Structure, Channel Coding and Modulation For Digital Terrestrial Television, 2004. v, ix, xiii, xxxiii, 132, 165, 302, 303, 304, 311, 312

[3] M.R.G. Butler, S. Armour, P.N. Fletcher, A.R. Nix and D.R. Bull. Viterbi Decoding Strategies for 5 GHz Wireless LAN Systems. Proceedings of 54th IEEE Vehicular Technology Conference. Vol.1, pp. 77 - 81, October 2001,. xxx, 171, 172

[4] ATSC Standard ASTC A/53 Part 1. ATSC Digital Television Standard, Part1 - Digital Television System, 2011. xxxiii, 316, 323

[5] ITU-R. Guidelines and Techniques for the Evaluation of Digital Terrestrial Television Broadcasting Systems. REPORT ITU-R BT.2035, 2003. xxxvi, 165, 179

[6] M. Salazar-Palma, A. Garcia-Lamprez, T.K.Sarkar and D.L. Segupta. The Father of Radio: A Brief Chronology of the Origin and Development of Wireless Communications. IEEE Antennas and Propagation Magazine, Vol. 53, No. 6, December 2011. 2

[7] M. S. Alencar, W. T. A. Lopes and F. Madeiro. History of Television in Brazil. Proceedings of Second IEEE Region 8 Conference on History of Telecommunications(HISTELCON), 2010. 2

[8] V. Donzelli. Digital TV in Brazil History and Prominent Role of SET. SET Magazine, Special Edition, March 2010. 2

[9] Mackenzie/ABERT/SET. Relatório Inicial do Projeto de Testes em Sistemas de Televiso Digital. `http://www.mackenzie.br/ee_atividades_hdtv_testes_lab.html`, 2000. 2

[10] L.L. Mendes, J. M. C. Brito, F. A. Cardoso, C. A.F.Rocha and D. S. Arantes. Projeto, Simulação e Implementação de um Sistema de Televisão Digital. Revista Científica Periódica - Telecomunicações,Vol. 09, No. 01, pp 21-30, November 2006. 3

[11] S. Alamouti. A simple transmit diversity technique for wireless communications. IEEE Journal on Selected Areas in Communications,(Volume:16 , Issue: 8), October 1998. 3, 328

[12] L.F.G. Soares, M.F. Moreno, C.S.S. Neto, M.F. Moreno. Ginga-NCL: Declarative Middleware for Multimedia IPTV Services. pages 74–81. ieee, June 2010. 3

[13] . Forum SBTVD. http://www.http://www.dtv.org.br/. 3

[14] A.A. de Araújo, P.B. Tigre, S. Bampi, S.F. Alves and M.W. de Almeida. Programa Nacional de Microeletrônica - Contribuições para a Formulação de um Plano Estruturado de Ações. Technical report, 2002. 5, 332, 333, 335

[15] CI-Brasil Program. Technical report. 5, 334, 335

[16] E. R. de Lima. Desenvolvimento de um Chipset Nacional para Recepção de Sinais ISDB-T: Lições Aprendidas (invited). In *Congresso SET 2010*, São Paulo, Brazil, 2010. SET. 8, 187

[17] E.R.de Lima. Method and Apparatus for Channel Estimation and Equalization (US Patent O62115418 - submitted, 2015. 8, 187, 206

[18] E.R. de Lima, T. Barros, G.S. Teles, V. Pimenta, M.A.J. Perez. J.P.S. Vermeeren and J.E. Bertuzzo. ISDB-T receiver architecture and VLSI implementation in 65 nm CMOS, for Fixed-Reception high definition Digital Television. In *Proceedings of 2012 IEEE International Conference on Consumer Electronics - Berlin (ICCE-Berlin)*. IEEE, 2012. 8, 9, 187

[19] E.R. de Lima et al. A 65nm CMOS ISDB-T VLSI Receiver Desing: Architecture, Algorithms and Implementation (submitted). In *IEEE Transactions in Broadcasting*. IEEE, 2015. 8, 9, 187, 206

[20] E.R. de Lima, A.F.R. Queiroz, G.S. da Silva, F.A.M. Erazo and J.E. Bertuzzo. Design and FPGA prototyping of a DVB-S2 receiver: Towards a VLSI implementation in CMOS. In *3rd Workshop on Circuits and Systems Design (WCAS 2013)*, Curitiba, Brazil, September 2013. 9, 238, 269, 272, 292

[21] D. C. Alves, E. R. de Lima, and J. E. Bertuzzo. A pipelined semi-parallel LDPC Decoder architecture for DVB-S2. In *3rd Workshop on Circuits and Systems Design (WCAS 2013)*, Curitiba, Brazil, Sep. 2013. 9, 213, 264, 269, 272, 292

[22] C. G. Chaves, E. R. de Lima, J. G. Mertes, and J. E. Bertuzzo. A Synthesizable Serial-in Syndrome Calculator for DVB-S2 BCH Decoding. In *3rd Workshop on Circuits and Systems Design (WCAS 2013)*, Curitiba, Brazil, September 2013. 9, 213, 264, 269, 272, 292

[23] C. G. Chaves, E. R. de Lima and J. G. Mertes. A Synthesizable BCH Decoder for DVB-S2 Satellite Communications (Best Paper Award). In *Proceedings of 4th Workshop on Circuits and Systems Design - WCAS*, Aracaju, Brazil, September 2014. 9, 269, 272, 292

[24] C. G. Chaves, E. R. de Lima and Jacqueline G. Mertes. A High Throughput BCH Decoder for DVB-S2. In *JICS - Journal of Integrated Circuits and Systems*. SBMicro/SBCCI, submitted 2015. 9, 269, 292

[25] J.A.B.R. Tardelli, E.R. de Lima, G.S. da Silva, G. Brito, D. C. Alves, C. G. Chaves and J. G. Mertes. Implementation of CORDIC-based Soft and Hard Demappers for a DVB-S2 Receiver. In *Proceedings of 4th Workshop on Circuits and Systems Design - WCAS*, Aracaju, Brazil, September 2014. 9, 269, 273, 292

[26] D. C. Alves, C. G. Chaves, E. R. de Lima, G. S. da Silva and A. F. R. Queiroz. FPGA Implementation of a FEC Decoding Subsystem for a DVB-S2 Receiver (Best Paper Award). In *Proceedings IX Southern Programmable Logic Conference -SPL*, Buenos Aires, Argentina, November 2014. 9, 269, 272, 292

[27] A. N. Sapper, E. R. de Lima, C. G. Chaves, G. S. da Silva, A. F. R. Queiroz and M. A. C. Fernandes. An Adaptive Equalizer for Reliable Transmissions in DVB-S2 Satellite Communications Under ISI. In *Proceedings of 6th LATINCOM - IEEE Latin-American Conference on Communications*, Cartagena de Indias, Colombia, November 2014. 10, 269, 273, 292

[28] E. R. de Lima, A.N. Sapper and M. A. C. Fernandes. An Adaptive Equalizer to Reduce the Nonlinear Distortion Effects in Satellite Broadcasting Systems. In *XXXIII Simpósio Brasileiro de Telecomunicaçẽs*, Juiz de Fora, Brazil, September 2015. 10, 269, 292

[29] E. R. de Lima, A. F. R. Queiroz, D. C. Alves and G. S. da Silva, C.G. Chaves, J.G. Mertes and T. M. Marson. A Detailed DVB-S2 Receiver Implementation: FPGA Prototyping and Preliminary ASIC Resource Estimation. In *Proceedings of 6th LATINCOM - IEEE Latin-American Conference on Communications*, Cartagena de Indias, Colombia, November 2014. 10, 269, 272, 292

[30] G. S. da Silva, A. F. R. Queiroz, E. R. de Lima and C. G. Chaves. A Novel Fine Frequency Estimation Serial Architecture applied in Satellite Communications. In *Proceedings ISCAS 2015 - IEEE International Symposium On Circuits and Systems*, Lisboa, Portugal, May 2015. 10, 269, 273, 292

[31] GHDL-G Hardware Description Language. `http://ghdl.free.fr/ghdl/index.html`. 11, 116

[32] GTKWave. http://6004.csail.mit.edu/6.371/cadtools/gtkwave/wave.html. 11, 116

[33] A. Prasad. Simulation of Physical Layer Impairments on Communication System. Master's thesis, San Diego State University, December 2012. 14, 46

[34] J. W. Cooley and J.W. Tukey. An Algorithm For the Machine Calculation of Complex Fourier Series, 1965. 14

[35] E.R. Lima, S.J. Flores, V. Almenar and A. Girona. Performance enhancements in OFDM-WLAN systems using MIMO access techniques. 1st International Symposium on Wireless Communication Systems, September 2004. 20

[36] H. Liu and G. Li, editors. *OFDM-Based Broadband Wireless Networks - Design and Optimization*. Willey-Interscience, 2005. 20, 21, 23

[37] Y. Li and L. Stuber. *Orthogonal Frequency Division Multiplexing for Wireless Communications*, chapter Basic Concepts. In Y.Li and Stuber [38], 2006. 20

[38] Y.Li and L. Stuber, editors. *Orthogonal Frequency Division Multiplexing for Wireless Communications*. Springer, 2006. 20, 22, 340

[39] T.J. Willink and P.H. Wittke. Optimization and Performance Evaluation of Multicarrier Transmission. IEEE Trans. Inform. Theory, (vol. 43, Issue 2, March 1997. 20

[40] R. Prasad. OFDM for wireless communications systems, 2002. 21, 22, 23

[41] H. Sari, M.I. Rahman, S. Kumar and R. Prasad. Transmission Techniques for Digital Terrestrial TV Broadcasting. IEEE Comm. Mag., (vol. 33, no. 2), February 1995. 22

[42] S. Kumar M. Marchetti, M.I Rahman and R. Prasad. *New Directions in Wireless Communications Research*, chapter OFDM: Principles and Challenges. In Tarokh [43], 2009. 23

[43] V. Tarokh, editor. *New Directions in Wireless Communications Research*. Springer Publishing Company, 2009. 23, 340

[44] M. Speth, S.A Fechtel, G. Fock and H. Meyr. Optimum receiver design for wireless broad-band systems using OFDM - Part I. IEEE Trans.Commun, (vol. 47), November 1999. 23, 138, 153

[45] V. D. Nguyen, H-P Kuchenbecker, H. Haas, K. Kyamakya and G. Gelle. Channel Impulse Response Length and Noise Variance Estimation for OFDM Systems with Adaptive Guard Interval. Journal on Wireless Communications and Networking (Volume 2007), January 2007. 23

[46] T.S. Rappaport. Wireless Communications: Principles and Practice, 1996. 24, 25, 26, 31, 33, 34

[47] J. P. Linnartz. Wireless Communication, 2006. 24

[48] A. Papoulis. Probability, Random Variables and Stochastic Processes, 3th edition, 1991. 25

[49] G. Turin, F.D. Clapp, T.L. Johnston, S.B Fine and D. Lavry. A Statistical Model of Urban Multipth Propagation. IEEE Trans. on Vehicular Technology (Vol. VT-21), February 1972. 26

[50] M. Engels. Wireless OFDM Systems: How to Make Them Work?, 2002. 27, 29

[51] W.C. Jakes. Microwave Mobile Communications, 1993. 30

[52] B. Sklar. Digital Communications: Fundamentals and Applications (2nd edition), 2001. 34, 36

[53] Y. Okumura et al. Field Strength and its Variability in VHF and UHF Land Mobile Radio Service. Review of the Elec. Comm. Lab. Vol. 16 numbers 9 and 10, 1968. 36

[54] M. Hata. Empirical Formula for Propagation Loss in Land Mobile Radio Service. Vehicular Technology, Vol. VT-29, no 3, 1980. 36

[55] V.K. Garg. Wireless Network Evolution: 2G to 3G, 2002. 36

[56] M. D. Yacoub. The $n - \mu$: General Fading Distribution. on Proceedings of Vehicular Technology Conference, 2000. IEEE-VTS Fall VTC 2000 Vehicular Technology Conference, 2000. 38, 39

[57] G. Fraidenraich. and M.D. Yacoub. The $\lambda - \mu$: General Fading Distribution . on Proceedings of the 2003 SBMO/IEEE MTT-S International(Volume:1), 2003. 38, 39

[58] L. Smaini. RF Analog Impairments Modeling for Communication Systems Simulation: Application to OFDM-based Transceivers, 2012. 41, 44, 45, 48, 50

[59] G. Fettweis, M. Lohning, D. Petrovic, M. Windisch, P. Zillmann and W. Rave. Dirt RF: A new paradigm. International Journal of Wireless Information Networks, Vol. 14, No. 2,, June 2007. 41

[60] F. Horlin and A. Bourdoux. Digital Compensation for Analog Front-Ends: A New Approach to Wireless Transceiver Design, 2008. 41

[61] H. P. Cruz, H. Gomes and N. Carvalho. Receiver Front-End Architectures - Analysis and Evaluation. http://cdn.intechopen.com/pdfs/9961/InTech-Receiver_front_end_architectures_analysis_and_evaluation.pdf, 2010. 42

[62] J.C. Rudell, J.-J. Ou, T.B. Cho, G. Chien, F. Brianti, J.A. Weldon and P.R. Gray. A 1.9-GHz Wide-Band IF Double Conversion CMOS Receiver for Cordless Telephone Applications. IEEE Journal of Solid-State Circuits, Vol. 32, No. 12, December 1997. 42

[63] M. Renfors. Lecture Notes: Receiver Architectures - Part 2 Increasing the role of DSP in receiver front-ends, 2008. 42

[64] J. Craninckx. CMOS Software-Defined Radio Transceiver: Analog Design in Digital Technology. IEEE Communications Magazine, Vol. 50, Issue 4, April 2012. 42

[65] A. Burr, E. Jaffrot, E. R. De Lima and W. Tech. *Mobile Broadband Multimedia Networks: Techniques, Models and Tools for 4G*, chapter Chapter 2: Transmission Techniques. In Correa [66], 2006. 44

[66] L. M. Correa, editor. *Mobile Broadband Multimedia Networks: Techniques, Models and Tools for 4G*. Elsevier, 2005. 44, 342

[67] C. Liang, J. Jong, W.E. Stark and J.R. East. Non linear amplifier effects in communications systems. IEEE Transactions on Microwave Theory and Techniques, Vol. 47, No. 8,, August 1999. 45

[68] G.T. Zhou and J.S. Kenney,J.S. Predicting spectral regrowth of nonlinear power amplifiers. IEEE Transactions on Communications, Vol. 50, Issue 5,, May 2002. 45

[69] T.-D. Chiueh and P.-Y Tsai. OFDM Baseband Receiver Design for Wireless Communications, 2007. 46, 49, 141, 146

[70] R. Svitek and S. Raman. DC Offsets in Direct-Conversion Receivers: Characterization and Implications, September 2005. 48

[71] B. Razavi. Design of Analog CMOS Integrated Circuits, 2001. 49

[72] D.G. Haldeman R.J. Marks, G. L. Wise and J.L. Whited. Detection In Laplace Noise, November 1978. 49

[73] M. Nassar J. Lin and B.L. Evans. Non-Parametric Impulsive Noise Mitigation in OFDM Systems Using Sparse Bayesian Learning, 2011. 49

[74] H.T. Friss. Noise Figures of Radio Receivers. Proceedings of the I.R.E., July 1994. 49

[75] Analog Devices. Converter Requirements for RX/TX: Techniques and Specifications, 1998. 50, 51

[76] ABNT Standard. ABNT NBR 15601 - Digital terrestrial television Transmission system. http://www.abnt.org.br/imagens/Normalizacao_TV_Digital/ABNTNBR15601_2007Vc_2008.pdf, 2009. 57

[77] ETSI Standard ETSI EN 302 307 V1.2.1 . Digital Video Broadcasting (DVB); Second Generation Framing Structure, Channel Coding and Modulation Systems for Broadcasting, Interactive Services, News Gathering and Other Broadband Satellite Applications (DVB-S2). ETSI, August 2009. 92, 93, 211, 212, 213, 216, 224, 237, 241, 242, 252, 253, 259, 260, 264, 280, 328

[78] ETSI TR 102 376, v. 1.1.1. Digital Video Broadcasting (DVB); User guidelines for the second generation system for Broadcasting, Interactive Services, News Gathering and other broadband satellite applications (DVB-S2). ETSI, February 2005. 92, 95, 100, 102, 113, 293

[79] A. Morello and V. Mignone. DVB-S2 ready for lift off. EBU Technical Review, October 2004. 93

[80] ETSI EN 302 307, V1.3.1. Digital Video Broadcasting (DVB); User guidelines for the second generation system for Broadcasting, Interactive Services, News Gathering and other broadband satellite applications (DVB-S2). ETSI, March 2013. 104, 105

[81] Portal EDS. http://www.portaleds.com/. 113

[82] Tracksat. http://www.tracksat.com/. 113

[83] DVB Fact Sheet - May 2014. http://www.dvb.org/resources/public/factsheets/dvb-s2x_factsheet.pdf month=May, year=2014. 113

[84] DVB. Digital Video Broadcasting (DVB); Second generation framing structure, channel coding and modulation systems for Broadcasting, Interactive Services, News Gathering and other broadband satellite applications - Part II: S2-Extensions (DVB-S2X) . https://www.dvb.org/resources/public/standards/a83-2_dvb-s2x_den302307-2.pdf, March 2014. 113, 248

[85] SVN. http://en.wikipedia.org/wiki/Apache_Subversion. 121

[86] J. E. Volder. he CORDIC Trigonometric Computing Technique. RE Trans. Elect. Comp., vol. EC-8, No. 3, 1959. 133

[87] Y. Hu. CORDIC-based VLSI Architectures for Digital Signal Processing. IEEE Sig. Proc. Mag., 1992. 133, 134, 211

[88] J. Valls, T. Sansaloni, A. Perez-Pascual, V. Torres and V. Almenar. The use of CORDIC in software defined radios: a tutorial . pages 46–50, September 2006. 133

[89] J. Terry and J. Heiskala. OFDM Wireless LANs: A Theoretical and Practical Guide, 2002. 141, 153

[90] T. Keller and L. Hanzo. Orthogonal Frequency Division Multiplex Synchronisation Techniques for Wireless Local Area Networks. in Proc. IEEE Int. Symp. on Personal, Indoor, and Mobile Radio Communications, Taipei, Taiwan, pp. 963-967, Oct. 1996,. 141, 145

[91] L. Hanzo and T. Keller. OFDM and MC-CDMA for Broadband Multi-User Communications, WLANs and Broadcasting, 2003. 145

[92] M. Speth, S. Fechtel, G. Fock and H. Meyr. Optimum receiver design for OFDM-based broadband transmission Part II: A case study. IEEE Transactions on Communications, Vol. 49, No 4, April 2001. 146, 153, 158

[93] C.W. Farrow. A Continuously Variable Digital Delay Element. In IEEE International Symposium on Circuits and Systems pp. 2641-2645, June 1988. 160, 215

[94] G.S. Liu, C.H. Wei. Programmable Fractional Sample Delay Filter with Lagrange Interpolation. Electronic Letters, Vol. 26, No.19, pp. 1608-1610, September 1990,. 160

[95] T.I. Laakso, V. Valimaki, M. Karjalainen and U.K. Laine. Splitting: Tools for Fractional Delay Filter Design. IEEE Signal Processing Magazine, pp. 30-60, Jan. 1996. 160

[96] D. Babic, J. Vesma, T. Saramaki, V. Lehtinen and M. Renfors. Polynomial-Based Interpolation Filters for DSP Applications Design, Implementation and Applications. http://www.cs.tut.fi/kurssit/TLT-5806/Interpol.pdf, 2008. 165

[97] J.-J. Van de Beek, O. Edfors, M. Sandell, S.K. Wilson and P.O. Borjesson. On Channel Estimation in OFDM Systems. Proceeding of IEEE Vehicular Technology Conf., vol. 2, pp.815 - 819 1995, 1995. 165

[98] O. Edfors. Low-complexity algorithms in digital receivers. PhD Thesis, 1996. 165

[99] A. Viterbi. Convolutional Codes and Their Performance in Communication Systems. IEEE Transactions on Communications, Vol. 19, No. 5, pp. 751-772, October, 1971. 171

[100] G. Forney. The Viterbi Algorithm. Proceedings of, Vol. 61, No. 3, pp. 268-278, March, 1973. 171

[101] F. Tosato and P. Bisaglia. Simplified Soft-Output Demapper for Binary Interleaved COFDM with Application to Hiperlan/2. Proceeding of IEEE International Conference on Communications - ICC. Vol. 2, pp. 664 - 668, April- May 2002. 171, 173

[102] H. Arslan and S. Reddy. Noise Power and SNR Estimation for OFDM Based-Wireless Communication Systems. Proceedings of IASTED Int. Conf. on Wireless and Optical Communications, 2003. 177

[103] F.M. Gardner. Interpolation in digital modems. I: Fundamentals. *IEEE Transactions on communications*, 41(3):501–507, 1993. 211, 215

[104] F.M. Gardner L. Erup and R.A. Harris. Interpolation in digital modems. II. Implementation and performance. *Communications, IEEE Transactions on*, 41(6):998–1008, 1993. 211, 215

[105] Y. Jiang F.-W.Sun and L.-N. Lee. Frame synchronization and pilot structure for second generation DVB via satellites. *International journal of satellite communications and networking*, 22(3):319–339, 2004. 211

[106] R. Gaudenzi E. Casini and A. Ginesi. DVB-S2 modem algorithms design and performance over typical satellite channels. *International Journal of Satellite Communications and Networking*, 22(3):281–318, 2004. 211, 215, 224, 226, 227, 231, 233, 234, 238, 242, 248

[107] A. Houghton. *Error coding for engineers*. Springer, 2001. 211

[108] Y. Jiang, F.B. Verahrami, W. Ting, R.L. Richmond and J.S. Baras. Data-aided ML parameter estimators for PSK burst modems and their systolic VLSI implementations. In *Global Telecommunications Conference, 1999. GLOBECOM'99*, volume 4, pages 2177–2182. IEEE, 1999. 211, 212, 224, 225

[109] S. Kay. A fast and accurate single frequency estimator. *Acoustics, Speech and Signal Processing, IEEE Transactions on*, 37(12):1987–1990, 1989. 211, 224

[110] J. Chen, A. Dholakia, E. Eleftheriou, M.P. Fossorier and X.-Y. Hu. Reduced-complexity decoding of LDPC codes. *Communications, IEEE Transactions on*, 53(8):1288–1299, 2005. 212

[111] M. Xie M. Yin and B. Yi. Optimized algorithms for binary bch codes. In *Circuits and Systems (ISCAS), 2013 IEEE International Symposium on*, pages 1552–1555. IEEE, 2013. 213, 264

[112] M. Gomes, G. Falcão, V. Silva, V. Ferreira, A. Sengo, L. Silva, N. Marques and M. Falcão. Scalable and parallel codec architectures for the DVB-S2 FEC system. In *Circuits and Systems, 2008. APCCAS 2008. IEEE Asia Pacific Conference on*, pages 1506–1509. IEEE, 2008. 213, 265

[113] M. Moeneclaey H. Meyr and S.A. Fechtel, editors. *Digital Communication Receivers: Synchronization, Equalization and Signal Processing*. John Wiley & Sons, Inc., 1998. 215

[114] F.M. Gardner. A BPSK/QPSK Timing-Error Detector for Sampled Receivers. *IEEE Transactions on Communications*, 34(5):423–429, 1986. 215

[115] Y. Jiang F.W. Sun and L.N. Lee. Frame Synchronization and Pilot Structure for Second Generation DVB via Satellites. *I. J. S. Communication and Network, Vol 22*, pages 319–339, June 2004. 218, 219

[116] L. Qing, Z. Xiaoyang, W. Chuan, Z. Yulong, D. Yunsong and J. Han. Optimal frame synchronization for DVB-S2. Proceedings of IEEE International Symposium on Circuits and Systems - ISCAS 2008., May 2008. 218, 219

[117] P. Kim, G.E. Corazza, R. Pedone, M. Villanti, D. Chang and D. Oh. Enhanced Frame Synchronization for DVB-S2 System Under a Large Frequency Offset. Proceedings of IEEE Wireless Communications and Networking Conference (WCNC) 2007., March 2007. 219

[118] Y. Zhang, X. Chen, W. Fan, J. Han, and X. Zeng. Robust and Reliable Frame Synchronization Method for DVB-S2 System. Proceedings of Wireless Telecommunications Symposium (WTS), April 2010. 219

[119] C. Wang R. Xue and X. Li. A Multiple Correlation Peak Value Detecting Method for Frame Synchronization of DVB-S2. Proceedings of 12th IEEE International Conference on Communication Technology (ICCT), November 2010. 219

[120] M. Luise and R. Reggiannini. Carrier Frequency Recovery in All-Digital Modems for Burst-Mode Transmissions. *Communications, IEEE Transactions on*, 43(2/3/4):1169–1178, 1995. 224, 226, 227

[121] S. Jung and D.-G. Oh. Frequency Estimation for Non-pilot Mode of DVB-S2 System. In *Proceedings of the World Congress on Engineering and Computer Science 2012*. LNCS, 2012. 225

[122] D.-I. Chang P. Kim, J. Han and G.-G. Oh. Efficient Channel Equalization Technique for DVB-S2 Standard. In *Advanced satellite multimedia systems conference (asma) and the 11th signal processing for space communications workshop (spsc), 2010 5th*, pages 208–212. IEEE, 2010. 227, 238

[123] J.W. Park, H.J. Yun, M.H. Sunwoo, P. Kim and D. Chang. Efficient Coarse Frequency Synchronizer Using Serial Correlator for DVB-S2. In *International Symposium on Circuits and Systems (ISCAS 2008)*, Washington, USA, May 2008. 227

[124] R. De Gaudenzi and M. Luise. Design and analysis of an all-digital demodulator for trellis coded 16-QAM transmission over a nonlinear satellite channel. *IEEE Transactions on Communications*, 43(2/3/4 Part I):659–688, 1995. 234

[125] D. Godard. Self-Recovering Equalization and Carrier Tracking in Two-Dimensional Data Communication Systems. *Communications, IEEE Transactions on*, 28(11):1867–1875, 1980. 237

[126] Y. Suzuki, A. Hashimoto, M. Kojima, H. Sujikai, S. Tanaka, T. Kimura and K. Shogen. A Study of Adaptive Equalizer for APSK in the Advanced Satellite Broadcasting System. In *Proceedings of the 28th IEEE conference on Global telecommunication GLOBECOM'09*, pages 2862–2867. IEEE, 2009. 238, 248, 295

[127] J.E. Volder. The CORDIC Trigonometric Computing Technique. *Electronic Computers, IRE Transactions on*, (3):330–334, 1959. 238

[128] B. Widrow and M. E. Hoff. Adaptive switching circuits. *WESCON Conv. Rec.*, part 4:96–140, 1960. 238

[129] S.O. Haykin, editor. *Adaptive Filter Theory*. Prentice Hall, 2002. 238

[130] E. Nemer. Physical Layer Impairments in DVB-S2 Receivers. In *Proceedings of Second IEEE Consumer Communications and Networking Conference*, pages 487 – 492. IEEE, 2005. 238

[131] A. Morello M. Cominetti, V. Mignone and M. Visintin. The European System for Digital Multi-Programme Television by Satellite. *Broadcasting, IEEE Transactions on*, 41(2):49–62, 1995. 238

[132] G. Malik and A.S. Sappal. Adaptive Equalization Algorithms: An Overview. *International Journal of Advanced Computer Science and Applications*, 2(3):62–67, 2011. 241

[133] A. Bastug, A. Ancora, and D.T.M. Slock. Adaptive Equalization at HSDPA Symbols Level. In *The 2nd International Workshop on Multiple Access Communications (MACOM09)*, Dresden, June 2009. 241

[134] A. Margetts and P. Schniter. Adaptive chip-rate equalization of downlink multirate wideband CDMA. *IEEE Trans. on Signal Processing*, 53(36), June 2005. 241

[135] C. Luschi S. Allpress and S. Felix. Exact and approximated expressions of the log-likelihood ratio for 16-QAM signals. In *38th Asilomar Conference on Signal, System and Computer*, 2004. 254

[136] J.W. Park, C.D. Ryu, M.H. Sunwoo, P.S. Kim and D. Chang. Simplified soft-decision demapping algorithm for DVB-S2. In *2009 International SoC Design Conference (ISOCC)*, pages 444–447, 2009. 254, 255

[137] K. Cho, K. Hyun, D. Yoon, S. Park and S. Cho. An approximated soft decoding algorithm of 16-APSK signal for DVB-S2. In *Consumer Electronics, 2007. ICCE 2007. Digest of Technical Papers. International Conference on*, pages 1–2. IEEE, 2007. 254

[138] Y.Q. Shi, X. Zhang, M. Xi, Z.-C. Ni and N. Ansari. Interleaving for Combating Bursts of Errors. *IEEE Circuits and Systems Magazine*, 4(1):29–42, 2004. 260

[139] C. Jinghu, A. Dholakia, E. Eleftheriou, M.P.C. Fossorier and H. Xiao-Yu. Reduced-Complexity Decoding of LDPC Codes. *IEEE Transactions on Communications*, 53(8):1288–1299, 2005. 261

[140] L. Conde-Canencia C. Marchand, J.-B. Dore and E. Boutillon. Conflict resolution for pipelined layered LDPC decoders. In *Signal Processing Systems, 2009. SiPS 2009. IEEE Workshop on*, pages 220–225, 2009. 261

[141] S. Muller, M. Schreger, M. Kabutz, M. Alles, F. Kienle and N. Wehn. A Novel LDPC Decoder for DVB-S2 IP. In *Design, Automation Test in Europe Conference Exhibition, 2009. DATE'09.*, pages 1308–1313. 262

[142] S. Lin and D.J. Costello. *Error Control Coding: Fundamentals and Applications.* Pearson-Prentice Hall, 2004. Page 167. 264

[143] H.-C Chang and C Bernard C. B. Shung. New Serial Architecture for the Berlekamp-Massey Algorithm. *Communications, IEEE Transactions on*, 47(4):481–483, 1999. 265

[144] A. N. Sapper, E.R. de Lima, C. G. Chaves, G. S. da Silva, A.F.R. Queiroz and M. A. C. Fernandes. An Adaptive Equalizer for Reliable Transmissions in DVB-S2 Satellite Communications Under ISI(acepted). In *IEEE Latin America Transactions*, to appear 2015. 269, 273, 292

[145] Altera Corporation. Stratix IV GX FPGA Development Board Reference Manual, August 2012. http://www.altera.com/literature/manual/rm_sivgx_fpga_dev_board.pdf[Accessed 25th Aug 2014]. 274, 276

[146] Altera Corporation. Stratix IV GX FPGA Development Kit User Guide, August 2010. http://www.altera.com/literature/ug/ug_sivgx_fpga_dev_kit.pdf[Accessed 25th Aug 2014]. 274, 285

[147] Cadence Design Systems. Encounter® RTL Compiler Datasheet, 2012. http://www.cadence.com/rl/Resources/datasheets/encounter_rtlcompiler.pdf [Accessed 25th Aug 2014]. 282

[148] John W. Eaton. About GNU Octave. http://www.gnu.org/software/octave/ [Accessed 25th Aug 2014]. 285

[149] A. Morello and V. Mignone. DVB-S2: The Second Generation Standard for Satellite Broad-band Services. Proceedings of the IEEE, Vol. 94, No. 1, January 2006. 293

[150] R. Trappe. DTV Status compares the worldwide digital broadcasting systems ATSC, DTMB, DVB-T/DVB-T2, and ISDB-T. http://en.dtvstatus.net/, 2013. 301

[151] U. Reimers. Dynamic Broadcasting - en Route to Convergence of Broadcast and Mobile. http://icin.biz/files/icin2011keynotes/ICIN_2011_Keynote_Reimers.pdf, 2011. 301

[152] FOBTV Initiative. Future of Broadcast Terrestrial Television. http://www.nercdtv.org/fobtv2012/index.html, 2013. 301, 329

[153] R. Hopkins. Tutorial on the Grand Alliance HDTV System. http://rhopkins.us/drbob.tv/documents/FCC_Tutorial_1994.pdf, 1994. 316

[154] ETSI EN 302 304 v1.1.1. Digital Video Broadcasting(DVB); Transmission System for Handheld Terminals (DVB-H), November 2004. 324, 325

[155] Produced by DVB Project Office. DVB Fact Sheet - August 2013. http://www.dvb.org/resources/public/factsheets/dvb-t2_factsheet.pdf, 2013. 327

[156] L. Vangelista, N. Benvenuto, S. Tomasin, C. Nokes, J. Stott, A. Filippi, M. Vlot, V. Mignone and A. Morello. Key Technologies for Next-Generation Terrestrial Digital Television Standard DVB-T2. IEEE Communications Magazine, October 2009. 327, 328

[157] L. Ardito and C. Sattler. Mobile Broadcast Bearer Technologies:a Comparison Update. http://www.oipf.tv/docs/BMCO, February 2009. 328

[158] ETSI Standard ETSI EN 302 755 V1.3.1 . Digital Video Broadcasting (DVB); Frame structure channel coding and modulation for a second generation digital terrestrial television broadcasting system (DVB-T2). ETSI. 328

[159] J.W. Swart. Evolução de Microeletrônica a Micro-Sistemas. Technical report. 331, 332

[160] J.B. Kuo. *Microelectronics R & D Blossoms in Taiwan.* ieee, March 1995. 333