

Spectrum-Aware Orthogonal Frequency Division Multiplexing

Adolfo L. Recio

Dissertation submitted to the Faculty of the
Virginia Polytechnic Institute and State University
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy
in
Electrical Engineering

Dr. Peter M. Athanas, Chair

Dr. Cameron D. Patterson

Dr. Thomas L. Martin

Dr. R. Michael Buehrer

Dr. Eric de Sturler

December 15, 2010

Bradley Department of Electrical and Computer Engineering
Blacksburg, Virginia

Keywords: Configurable Computing, Dynamic Spectrum Access, OFDM, Filter Bank
Multi-carrier, Spectrum Awareness, Cognitive Radio, Overlay Systems.

Copyright 2010, Adolfo L. Recio

Spectrum-Aware Orthogonal Frequency Division Multiplexing

Adolfo L. Recio

(ABSTRACT)

Reconfigurable computing architectures are well suited for the dynamic data flow processing requirements of software-defined radio. The software radio concept has quickly evolved to include spectrum sensing, awareness, and cognitive algorithms for machine learning resulting in the cognitive radio model.

This work explores the application of reconfigurable hardware to the physical layer of cognitive radios using non-contiguous multi-carrier radio techniques. The practical tasks of spectrum sensing, frame detection, synchronization, channel estimation, and mutual interference mitigation are challenges in the communications and the computing fields that are addressed to optimally utilize the capacity of opportunistically allocated spectrum bands.

FPGA implementations of parameterizable OFDM and filter bank multi-carrier (FBMC) radio prototypes with spectrum awareness and non-contiguous sub-carrier allocation were completed and tested over-the-air. Sub-carrier sparseness assumptions were validated under practical implementation and performance considerations. A novel algorithm for frame detection and synchronization with mutual interference rejection applicable to the FBMC case was proposed and tested.

The author would like to thank the Harris Corporation, Government Communications Division, for supporting this research.

Dedication

To my wife and children, for being a wonderful family and for supporting me at all times and enduring the difficulties of having a student husband and father.

Acknowledgments

To Dr. Peter Athanas, thank you for guiding me through the research process and for being there always to help me when I required your support.

To Neiyer Correal, for suggesting that I should go back to school and get a doctorate.

To Dave and Angela Cassady, who provided unconditional support for our family plans.

To Ricardo Uribe, for extending his helping hand when we most needed it.

To Tony Frangieh, Prabhaav Bhardwaj, Jacob Couch, Ali Sohangpurwala, Charles Irick, Jorge Suris, and many more great fellow students who became great friends during these years at Virginia Tech.

To my parents and sisters for believing in my capability to reach ambitious goals.

Acronyms

AGC	Automatic Gain Control
ALU	Arithmetic Logic Unit
AMC	Adaptive Modulation and Coding
AP	Alternating Projection
ASIC	Application Specific Integrated Circuit
AWGN	Additive White Gaussian Noise
BS	Base Station
BPSK	Binary Phase Shift Keying
CAS	Carrier Allocation Scheme
SINR	Signal to Interference and Noise Ratio
CDMA	Code Division Multiple Access
CSI	Channel State Information
CQI	Channel Quality Indicator
DCM	Digital Clock Manager
DS-CDMA	Direct Sequence Code Division Multiple Access
DSP	Digital Signal Processor
DVB-T	Digital Video Broadcasting - Terrestrial
ECA	Elemental Computing Architecture
EVM	Error Vector Magnitude

FBMC	Filter bank multi-carrier
FDD	Frequency Division Duplex
FDMA	Frequency Division Multiple Access
FER	Frame Error Rate
FFT	Fast Fourier Transform
FIR	Finite Impulse Response
FIFO	First In First Out
FPGA	Field Programmable Gate Array
GPP	General Purpose Processor
HDL	Hardware Description Language
ICI	Intercarrier Interference
IEEE	Institute of Electrical and Electronics Engineers
IF	Intermediate Frequency
IFFT	Inverse Fast Fourier Transform
ISI	Intersymbol Interference
LO	Local Oscillator
LTE	Long Term Evolution
LUT	Look Up Table
MAC	Media Access Control Layer
MC-CDMA	Multi-carrier Code Division Multiple Access
MIMO	Multiple Input Multiple Output
MLE	Maximum Likelihood Estimation
MMSE	Minimum Mean Squared Error
OFDM	Orthogonal Frequency Division Multiplexing
OFDMA	Orthogonal Frequency Division Multiple Access
OS	Operating System
OSI	Open Systems Interconnection
PAPR	Peak to Average Power Ratio

PCIe	Peripheral Component Interconnect Express
PHY	Physical Layer
PSK	Phase Shift Keying
PU	Primary User
QAM	Quadrature Amplitude Modulation
RF	Radio Frequency
RSSI	Received Signal Strength Indication
SAGE	Space Alternating Generalized Expectation-Maximization
SDR	Software Defined Radio
SC-FDMA	Single Carrier - Frequency Division Multiple Access
SNR	Signal to Noise Ratio
SOC	System On Chip
SS	Subscriber Station
SU	Secondary User
TDMA	Time Division Multiple Access
TDD	Time Division Duplex
UWB	Ultra-Wide Band
WRAN	Wireless Regional Area Network
XML	eXtensible Markup Language

Contents

1	Introduction	1
1.1	Motivation	1
1.2	Advantages of FPGAs for Radio Processing	5
1.3	Research Problems	6
1.4	Research Questions	7
1.4.1	Primary Research Questions	7
1.4.2	Secondary Research Questions	8
1.5	Research Contributions	8
1.6	Organization of the Document	9
2	Multi-carrier Overlay Systems	10
2.1	Classification of Dynamic Spectrum Access	10
2.2	OFDM-based Overlay Systems	14
2.2.1	The Concept of OFDM	14
2.2.2	Clasification of Sub-carrier Allocations	15
2.2.3	OFDM Overlay	16

2.2.4	Challenges	17
2.2.5	The Mutual Interference Problem	18
2.2.6	Techniques for Mutual Interference Mitigation	20
2.2.7	A Multiuser Detection Approach for Interference Mitigation	21
2.3	Filter Bank Multi-carrier Overlay Systems	22
2.3.1	Filter Bank Theory	24
2.3.2	Introduction to FBMC Methods	24
2.3.3	FBMC for Cognitive Radio	26
2.3.4	Synchronization and Channel Estimation	27
2.3.5	Structures for Efficient Implementation	28
2.4	Conclusions	28
3	Configurable Computing for Radio Applications	30
3.1	FPGA Software Defined Radio	30
3.1.1	Layered Radio Architecture	31
3.1.2	Rapid Radio	32
3.1.3	Agile Hardware for Radio Development	33
3.1.4	Hardware Acceleration of GNU Radio	33
3.1.5	The Berkeley BEE2 Hardware Platform	34
3.1.6	The Winlab Cognitive Radio Hardware Platform	35
3.2	Multi-carrier Radio on FPGA	36
3.2.1	Techniques for Implementation of Multi-carrier Radios	36

3.2.2	Hardware Reductions for Multi-carrier Radio	37
3.2.3	Heterogeneous Architecture Partitions	40
3.3	FPGA Architecture and Programming Model	42
3.4	Conclusions	42
4	Introduction to Cognitive Radio	44
4.1	The Ideal Cognitive Radio	44
4.2	Spectrum Sensing	46
4.2.1	Spectrum Sensing Strategies	47
4.2.2	Distributed Spectrum Sensing	50
4.2.3	An Opportunity for the Application of Machine Learning	52
4.3	Cognitive Networks	52
4.4	Multi-carrier Cognitive Radio	53
4.5	Conclusion	55
5	Spectrum Awareness	56
5.1	Modeling of the primary user spectral shape	57
5.1.1	Description of the Problem	58
5.1.2	An Algorithm for Signal Feature Extraction	60
5.1.3	Advantages over Power Sensing Methods	69
5.1.4	Over-the-air Experiments	71
5.2	Automatic Modulation Classification	74

5.2.1	Previous Work on Automatic Modulation Classification	75
5.2.2	Proposed Approach	77
5.2.3	Parameter Estimation and Signal Conditioning	77
5.2.4	Synchronization and Amplitude Normalization	78
5.2.5	Classification Features	80
5.2.6	Bayesian Network Classifier	85
5.2.7	Results	86
5.3	Conclusions	92
6	FMT with Mutual Interference Rejection	93
6.1	Legacy Frame Detection and Timing Offset Estimation Algorithms	94
6.1.1	Frame Detection and Coarse Timing Acquisition	95
6.1.2	Carrier Frequency Offset Acquisition	96
6.1.3	Fine Timing Acquisition	97
6.1.4	Channel estimation	97
6.1.5	Timing and Frequency Tracking	98
6.1.6	Summary	99
6.2	Filtered Multitone	99
6.2.1	Derivation of the FMT Transmitter	100
6.2.2	Derivation of the FMT Receiver	103
6.3	CAZAC Sequences	106

6.4	An Algorithm for Frame Detection and Timing Offset Estimation with Interference Rejection	111
6.4.1	Efficient Calculation of the Cyclic Cross-Correlation Function	112
6.4.2	Sub-carrier Excision	117
6.4.3	Frame Detection Hypothesis Test	118
6.4.4	SNR Estimation	118
6.4.5	Frame Design	119
6.4.6	Channel Estimation	120
6.4.7	Algorithm Summary	120
6.5	Results	123
6.6	Conclusion	124
7	Hardware Architecture	126
7.1	Hardware Architecture of the OFDM Radio	126
7.1.1	Allocation vector	127
7.1.2	Frame Format	127
7.1.3	The Harris System-in-Package	130
7.1.4	Transmitter Architecture	131
7.1.5	Receiver Architecture	140
7.2	Hardware Architecture of the FMT Radio	158
7.2.1	FMT Transmitter	158
7.2.2	FMT Receiver	161

7.3	Results	172
7.3.1	Measured Spectrum	172
7.3.2	FPGA Resource Utilization	173
7.4	Conclusion	176
8	Conclusions	178
8.1	Future Work	181
8.2	Publications	185
8.3	Planned Publications	186
A	Prototype Reference Manual	205
A.1	LabView Graphical User Interface	206
A.2	Primary User Generation	208
A.3	Project File Organization	209
A.4	RF Front End	209
A.5	Lego Robot	209
A.6	Instructions to Run the Demonstration	210

List of Figures

2.1	Taxonomy of dynamic spectrum access (after [1]).	12
2.2	Hierarchical Access Methods	13
2.3	An OFDM Modulator	14
2.4	The three cases of OFDMA sub-carrier allocation	15
2.5	Sensing, allocation, and transmission	17
2.6	Detection of primary user activity and recovery	18
2.7	FBMC simplification	23
2.8	Response of an example prototype filter	23
3.1	Pruning of nodes of the FFT algorithm (after [2]).	38
4.1	The cognitive cycle	46
4.2	White space in a managed band.	48
4.3	Energy detection spectral sensing.	48
4.4	Energy detection spectral sensing with filter banks.	49
4.5	Spectral sensing with matched filtering.	49
4.6	Spectral sensing based on cyclostationary features.	50

4.7	Spectral sensing based on partial matched filtering (after [3]).	50
4.8	Impairments in Spectrum Sensing	51
4.9	Diversity in Spectrum Sensing (after [1]).	52
5.1	Tiers of PU modeling	57
5.2	Raised-cosine function $y(f, \beta)$	63
5.3	Normalized Jacobian matrix for $f_c = 250$ kHz, $f_s = 200$ kHz, $\alpha = 0.55$, ($C + N$)/ $N = 3$ dB	66
5.4	Example fitted carrier under parameters of Table 5.1 and a ($C + N$)/ N of 3 dB.	68
5.5	Estimates of spectrum utilization	69
5.6	Threshold line for PU signal detection	70
5.7	Down conversion and down sampling	72
5.8	Normalized standard deviation of $\hat{f}_c$ relative to f_s	73
5.9	Normalized standard deviation of $\hat{f}_s$	74
5.10	Normalized standard deviation of roll-off estimate $\hat{\alpha}$	75
5.11	AMC system block diagram	78
5.12	Signal parameter estimation and pre-conditioning	79
5.13	Symbol synchronizer architecture	79
5.14	Amplitude profiles of a 16-QAM constellation with 18 dB SNR	82
5.15	Differential phase profile under QPSK with 12 dB SNR	83
5.16	Two-dimensional PDF under QPSK with 12 dB SNR	84
6.1	IEEE 802.11a/g Frame.	94

6.2	Set of signals for frame detection and CFO estimation	96
6.3	OFDM receiver architecture	97
6.4	Time-domain tracking receiver architecture	98
6.5	Frequency-domain tracking receiver architecture	99
6.6	FMT transmitter	101
6.7	Modulated filter bank FMT transmitter	103
6.8	FMT receiver based on band pass filters, downconversion, and downsampling	104
6.9	Efficient implementation of an FMT receiver	106
6.10	Robustness of CAZAC sequences to carrier frequency offset	109
6.11	Detail of sinc envelope with a fractional delay of $\theta = 0.7$	111
6.12	Random sampling time offset at the output of the receive-side filter bank . .	113
6.13	FMT preamble generation with upsampling rate of 2	114
6.14	Block processing of the cross-correlation operation	115
6.15	Block cross-correlation with sub-carrier excision	117
6.16	Frame structure with two isolated preambles	120
6.17	ROC with timing offset $\theta = N/4$	124
6.18	ROC with timing offset $\theta = N/4$ and $\rho = 32$	125
7.1	Frame format	128
7.2	Amplitude of the short preamble in the frequency domain	129
7.3	Block diagram for generation of the short preamble	129
7.4	Signal field format	130

7.5	The Harris SiP	131
7.6	Transmitter architecture	132
7.7	Clock sources of the TX1 FPGA	133
7.8	OFDM transmitter outer state machine	134
7.9	Data mapping state machine	135
7.10	Transmitter state machine verification	138
7.11	Filter requirements for interpolation by three	139
7.12	Three stage CIC filter response	140
7.13	Order 95 interpolating FIR filter response	141
7.14	Module dependency of the OFDM transmitter implementation	142
7.15	Receiver architecture	142
7.16	Conceptual down conversion filter	143
7.17	Efficient down conversion filter	143
7.18	Clock sources of the RX1 FPGA on the Harris SiP 1	144
7.19	Clock sources of the RX1 FPGA on the Harris SiP 2	145
7.20	Implementation of the autocorrelation function	146
7.21	Frame detection verification	147
7.22	Implementation of the adder tree	150
7.23	Fine timing estimation	151
7.24	Phase tracking subsystem	152
7.25	IEEE 802.11 16-QAM constellation (from [4])	155

7.26	Module hierarchy for the OFDM receiver on SiP 1	156
7.27	Module hierarchy for the OFDM receiver on SiP 2	157
7.28	FMT transmitter processing with over sampling rate of two	158
7.29	FMT transmitter outer state machine	159
7.30	Frequency domain response of the transmit side prototype filter	160
7.31	Impulse response of the transmit side prototype filter	160
7.32	Module hierarchy for the FMT transmitter	161
7.33	Block diagram of the FMT Receiver	162
7.34	Block processing core	162
7.35	Dataflow for spectrum sensing	166
7.36	Dataflow for frame detection, timing estimation, and timing alignment . . .	167
7.37	Dataflow for channel estimation	168
7.38	Dataflow for demodulation	169
7.39	Module hierarchy for the FMT receiver on SiP 1	170
7.40	Module hierarchy for the FMT receiver on SiP 2	171
7.41	OFDM spectrum with two sets for zeroed sub-carriers	172
7.42	FMT spectrum with two sets for zeroed sub-carriers	173
7.43	Equivalent rejection filter	174
7.44	Utilization of DSP resources	175
8.1	Frame format for 2×2 MIMO	183
A.1	Overview of the demonstration setup	206

A.2	Set up of the demonstration	206
A.3	The LabView GUI	208
A.4	Options of the LabView GUI	211
A.5	The PU transmitter	212
A.6	The Avnet Virtex-5 VSX95T board with DAC daughter board	212
A.7	Directory Organization of the project	213
A.8	Up converter	213
A.9	Down converter	214
A.10	The Harris SiP board	214
A.11	A receive antenna and filter	215
A.12	Lego robot with deflection shield	215

List of Tables

1.1	Benchmarks for a 1024 point FFT calculation, 16 bit precision.	6
5.1	Parameters used for simulation and over-the-air experiments	72
5.2	Set of metrics for a BPSK signal	87
5.3	Set of metrics for a QPSK signal	87
5.4	Set of metrics for an 8-PSK signal	88
5.5	Set of metrics for a $\pi/4$ -DQPSK signal	88
5.6	Set of metrics for an 8-QAM signal	89
5.7	Set of metrics for a 16-QAM signal	89
5.8	Set of metrics for a Star-16 signal	90
5.9	Set of metrics for a 32-QAM signal	90
5.10	Set of metrics for a 64-QAM signal	91
6.1	Specifications of the IEEE 802.11a/g waveform	95
6.2	Set of parameters for simulation of frame detection	123
7.1	Allocation vector coding	127

7.2	CAD bus address map for TX1 FPGA	134
7.3	Bandpass sampling frequencies	142
7.4	CAD bus address map for RX1 FPGA	146
7.5	Sign bit mapping	149
7.6	FFT core configuration options	165
7.7	Latency of the timing alignment logic (clock cycles)	166
7.8	Resource utilization of the FMT receiver	175
7.9	Resource utilization of the OFDM acquisition stage	176
7.10	Resource comparison of the transmitters	176
A.1	Specifications of the primary user signal	209

List of Algorithms

1	Trust-region size parameter update	65
2	Algorithm for frame detection and timing offset correction	121
3	Algorithm for disambiguation and channel estimation	122

Chapter 1

Introduction

1.1 Motivation

The formation of wireless communications networks capable of high speed wide band transmission, low cost, and high quality of service is a top priority of the telecommunications industry. New ideas have been proposed to increase the utilization efficiency and availability of the spectrum while generating revenue for service providers and equipment manufacturers. A promising vein to achieve these goals is the study of techniques for secondary or opportunistic networks. These networks exploit the spectrum bands not used by licensed or primary users in the same way that carpooling is used as a strategy to reduce cost, congestion, and pollution in transportation.

Licensed spectrum users have permission to use a certain bandwidth, but they cannot necessarily exploit this resource at all times or locations. This is an inefficient model, similar to a car running with empty seats. If a group of people with similar transportation requirements can share a ride, using a car in a cooperative manner makes more sense economically and environmentally. The main complication of this idea is how to efficiently schedule the trips. Once the time, origin, and destination are agreed, the car owner and his passengers can

choose one of many strategies: planning in advance how to share the expenses of the trip, taking turns for driving each others car, or ad hoc riding for free. In many cities, having passengers enables the car driver to use the high occupancy vehicle (HOV) lane and reach his destination faster than if he did not carry any [5].

Passengers are usually required to adapt to the conditions of the driver. Free-riding passengers can keep records to better understand the travel patterns of car owners and of other free riders. Analysis of data such as days and times of the week with heavy traffic congestion and the times when a preferred driver passes every day, can be used to define strategies and actions based on other's habits and own personal preferences.

Primary radio spectrum users pay expensive licenses for the use of spectrum bands; they can benefit from sharing spectrum with secondary users by charging them a lease fee. They may even be mandated by the regulation to share the allocated spectrum bands or time-slots. Secondary users need to adapt rapidly to the conditions of the available spectrum, and need to migrate to a different radio band when the primary users in the target band are making full use of their resources. This adaptability requirements suggest the use of cognitive radio techniques [6].

Orthogonal frequency division multiplexing (OFDM) features multiple sub-carriers located at a set of orthogonal frequencies acting as parallel low-speed channels that convey a high rate of information when combined. OFDM-based systems are in use by several standards for fixed and mobile networks, such as the American IEEE 802.11 (WiFi), IEEE 802.16 (WiMax), the European Digital Video Broadcast-Terrestrial (DVB-T), and Long Term Evolution of mobile communications (LTE). It is also one of the options proposed for the IEEE 802.22 TV band overlay system [7] that intends to develop an unlicensed Wireless Regional Area Network (WRAN) technology exploiting unused TV bands.

The benefits of OFDM are its parallel-friendly hardware implementation, bandwidth efficiency, ability to combat frequency selective fading, and control over the parameters of each sub-carrier. These characteristics make OFDM a suitable technique for broadband commu-

nications.

The individual sub-carrier control feature of OFDM can be further exploited by adding spectrum sensing to detect and avoid interfering with a primary user. Interference avoidance is accomplished by transmitting null carriers at the primary frequency bands and dynamically responding to changes.

Several authors claim that the patterns observed in a specific sub-carrier allocation may be leveraged for hardware optimizations [8, 9, 10, 2, 11]. For example, the operations of the IFFT used to perform modulation may be optimized for the specific subset of carriers assigned at a given time.

While a wealth of literature is devoted to the theory and simulation aspects of adaptive OFDM [12], non-contiguous OFDM [13], and overlay OFDM/OFDMA systems [14, 15, 16], little work has focused on leveraging the capabilities of reconfigurable computing architectures for their implementation. Veilleux et al. [17] and Guffey et al. [18] propose implementations of adaptive OFDM on reconfigurable hardware; however, they do not consider spectrum sensing and interference mitigation in a secondary user setting. In addition to OFDM adaptation, Wang et al. [19] propose a protocol for run-time hardware reconfiguration in response to the communications channel conditions.

In spite of the advantage provided by individual sub-carrier control, the frequency-domain sidelobes of an OFDM signal produce high mutual interference between a primary user and the overlay application. This phenomenon limits the information capacity of an OFDM-based overlay system. Several techniques have been proposed to mitigate this problem in the OFDM realm. Some of these ideas are canceling sub-carriers, time domain windowing, and guard sub-carriers. These methods do not address the problem of finite symbol time duration that produces frequency domain spreading of the sub-carrier energy.

Pulse shapes spanning the duration of several symbols is a natural way to enhance the spectral containment of each sub-carrier; however, using shaping filters prevents the utilization of a cyclic prefix and its associated simplifications. Polyphase decompositions allow the

creation of modulated filter banks (channelizers). The addition of a single filter bank to the existing FFT/IFFT operations of OFDM permits the generation of signals with high spectral containment at a low computational cost [20].

During the development of this project, filter bank multi-carrier (FBMC) methods became a focal point. Their spectral efficiency [21] and individual sub-carrier control make them a more appropriate solution for secondary user applications.¹ Recent publications discuss the efficient implementation of FBMC systems, among them Vangelista et al. [22] and Moret et al. [23]. Neither researcher analyzes the secondary user application or provides details on their implementation on reconfigurable hardware.

The research presented in this dissertation explores an efficient hardware implementation of interference rejection at frame detection and synchronization acquisition time in opportunistic spectrum systems based on filter bank methods.

New directions of research are possible considering the high computational power of FPGAs such as the development of cognitive wide band radios. Multiple bands can be managed simultaneously from the same radio, increasing the data speed and expanding the set of frequencies that can be allocated for a transmission. Attention must be paid to the development of modular scalable components and strategies for managing large bandwidths.

Another area of research is the enhancement of system dynamics, defined as the agility in sensing the environment and adapting to changes. In spatially distributed systems, adaptation decisions must be taken considering the network interactions with the primary user and between peer radios.

¹A compilation of works on FBMC methods can be found at the PHYDYAS project site <http://www.ict-phydyas.org>

1.2 Advantages of FPGAs for Radio Processing

Data flow applications such as radio processing perform faster in configurable hardware than in sequential processors, as illustrated below. In hardware the signal samples are flowing through the computing structure, so data or instruction fetch from memory is not required.

Another benefit of dedicated hardware is its flexibility to trade resources for speed, for example, an N element inner product can be performed in a fully parallel fashion if N multipliers are assigned. It can also be split into Q sequential multiplications, requiring N/Q shared multipliers. These trade offs are evident in Table 1.1² where a Virtex-6 VLX-75T FPGA [24] is compared to a Texas Instruments TMS320C6000 DSP [25]. While the sequential execution of the DSP leaves no degrees of freedom, FPGA implementations allow the user to exchange resources for speed by allocating more hardware for parallel processing.

Configurable hardware possesses a clear advantage in computational performance over DSP. Similar comparisons can be made in terms of power consumption [26]; however, the process of implementing an FPGA design requires more time and effort than programming a sequential processor for several reasons:

- Hardware description languages are low-level. The designer must be careful about details such as registering and timing closure.
- The parallel nature of FPGAs requires a detailed design of the algorithms to achieve the desired balance between area and performance.
- FPGAs have fine-grained architectures, thus the designer must choose the specific resources he wants to assign for a given task.

The following is a summary of aspects to consider when choosing between reconfigurable hardware and sequential processors, such as GPP and DSP, for radio implementations:

²FPGA implementation options are: realtime mode, optimized for speed, scaled, convergent rounding, natural ordering, block RAM. Streaming is not performing cyclic prefix appending.

Table 1.1: Benchmarks for a 1024 point FFT calculation, 16 bit precision.

Hardware	Implementation	DSP Slices	Clock Speed	Cycles	Latency
Xilinx XC6VLX75T	Radix-2 Lite	3	395 MHz	12476	31.58 μs
	Radix-2	6	395 MHz	7357	18.63 μs
	Streaming	40	395 MHz	3202	8.11 μs
TI TMS320C6000	Radix-2	1	200 MHz	20815	104 μs
	Radix-4	1	200 MHz	13228	66 μs

Processing power. Reconfigurable hardware has a higher computational processing power and flexibility in allocation of resources than sequential processors.

Rapid reconfiguration. The capability of agile reconfiguration in hardware for radio applications is proven by previous projects developed at the Configurable Computing Laboratory [27], and can be applied to enable a quick response to decisions made at the cognitive layer of a radio.

Prototyping capability. Reconfigurable hardware provides a viable mechanism for rapid prototyping and demonstration of concepts when compared to ASICs.

Cost. Large FPGAs are expensive and difficult to program compared to sequential processors.

1.3 Research Problems

This project addresses the following problems:

Interferer signal modeling. This is an important aspect of spectrum awareness that can be accomplished at several levels of abstraction. The lowest level is power thresholding, and the highest level is full parameter estimation and automatic modulation

recognition. This topic is treated in Chapter 5.

Interference rejection. Frame detection, synchronization acquisition, and channel estimation are usually taken for granted. Typical algorithms to develop these tasks are based on time-domain analysis that does not include strategies for the rejection of wide band interference. An improved algorithm for the accomplishment of this tasks is proposed and analyzed in Chapter 6.

Common operators. FPGA partial reconfiguration requires more time to be performed when compared to the intervals available among the components of a high-speed data frame. For this reason, one of the focuses of research on area reduction and application specific architecture is the reuse of parameterizable static modules. The hardware reuse architecture proposed for the multi-carrier overlay radio is presented in Chapter 7.

1.4 Research Questions

The following questions were addressed in the duration of the project. They can be classified as primary and secondary research focuses:

1.4.1 Primary Research Questions

- What is the best model of the primary user signals for the purpose of minimizing interference?
- How should the signal be designed and what are the receiver algorithms for a secondary user radio system with built-in mutual interference rejection applicable at frame detection, symbol and frequency synchronization acquisition, and demodulation times?
- What are the best hardware configurations to support the developed algorithms considering the trade off between latency and component reuse?

1.4.2 Secondary Research Questions

- What are the methods to obtain a flexible and scalable architecture in terms of number and allocation of the sub-carriers?
- What are the most computationally inexpensive algorithms for link adaptation in a secondary user scenario?
- What is the optimal task partition between reconfigurable computing hardware and a host computer?

1.5 Research Contributions

The realization of the project produced the following contributions:

- The application of a non-linear least squares optimization algorithm for the estimation of the parameters of a single-carrier primary user/interferer based in a model of the spectral shape obtained by the unmodified periodogram. A clear contribution is the use of the spectral shape as a means for obtaining the frequency offset, symbol frequency, roll-off factor, carrier power density, and noise power density without prior knowledge of the linear modulation type.
- An algorithm for automatic modulation classification based on the set of parameters obtained from the analysis of the power spectral density. The algorithm is blind in the sense that it performs matched filtering and synchronization using the values obtained from the parameter estimation step. A central contribution of this work is the calculation of the hypothesized probability density functions. The Hellinger distance is proposed as a metric from these functions to a set of histograms obtained from the synchronized samples of the input signal, and used to approximate a conditional probability metric that is fed to a Bayesian network.

- The parameterization of hardware for multi-carrier transmitters and receivers based in an allocation vector that is obtained from the spectrum awareness algorithms. Parameterization is a fast method for modifying the behavior of a hardware module to fit the conditions for generating or demodulating a specific signal type.
- A new physical-layer protocol to support filter bank multi-carrier (FBMC) signals with non-contiguous sub-carrier allocation and interference rejection. The protocol is based on separated preamble symbols to avoid the inter-symbol interference (ISI) effects produce when the receiver is out of timing synchronization.
- A mathematical description of a novel algorithm and its corresponding hardware architecture for the processing of spectrum sensing, frame detection, timing offset estimation, channel estimation, and demodulation tasks on FBMC signals with component reuse.
- OFDM and FBMC prototypes based on FPGA that are used to evaluate over-the-air the concepts that resulted from the theoretical work.

1.6 Organization of the Document

This dissertation is organized as follows: the related work and literature review is presented in Chapters 2, 3, and 4. The contributed algorithms and their performance analysis are shown in Chapters 5 and 6. Chapter 7 explains the hardware architecture. The conclusions of the project are summarized in Chapter 8. The source code organization and the demonstration setup are presented in Appendix A.

Chapter 2

Multi-carrier Overlay Systems

This chapter presents an introduction to OFDM and FBMC as methods for the implementation of multi-carrier overlay systems and a summary of past work on this topic. OFDM is usually presented as the ideal method for secondary user applications; however, the mutual interference problem limits the applicability of this approach. A related family of modulations known as filter bank multi-carrier (FBMC) presented in recent papers is an effective alternative to overcome the main drawbacks of OFDM thanks to their inherent spectrum containment characteristics. A general classification of dynamic spectrum access is presented in Section 2.1 to introduce the discussion.

2.1 Classification of Dynamic Spectrum Access

Zhao et al. present an overview of the strategies for dynamic spectrum access [1], summarized in Figure 2.1. Spectrum access is classified into the following categories:

Dynamic exclusive use. In this model, spectrum rights are exclusive; the conditions of the license are defined when it is granted. The following variants over the current licensing scheme are proposed to add flexibility:

- Spectrum property rights. The licensee has the freedom to trade spectrum and to choose the technology to use in the frequency band. The main disadvantage of this market-inspired scheme is be the lack of government regulation over the spectrum.
- Dynamic spectrum allocation. This scheme relies on the temporal and spatial statistics of different type of services, to assign the spectrum usage in a dynamic fashion, with instantaneous assignments in response to the user demand of resources. This scheme would be more efficient than the current spectrum licensing policy, but it would still suffer from inefficiencies caused by unused spaces.

Open sharing. This model proposes the free use of spectrum, in the same way today's IEEE802.11 networks operate. The management of the resources is left to coexistence policies implemented either in the network elements or in a centralized spectrum management processor.

Hierarchical access. This model proposes the existence of licensed primary users (PU) with the highest priority for the use of the spectrum, and unlicensed secondary users (SU) that opportunistically use the remaining interference margin or spectral white spaces. Two types of hierarchical access derive from these approaches:

- Underlay systems. In this spectrum sharing scheme, opportunistic users take advantage of the the interference margin allowed by the primary user and use transmission modes that spread the signal over a wide band with low-power spectrum density. This is a "soft" sharing scheme, because the users are operating on the same frequency band and its capacity limit is defined by the level of mutual interference among systems (interference temperature). This method has the advantage of not requiring spectrum sensing. However, quality sensing is still required to guarantee that the communication links provide the specified quality of service.

- Overlay systems. In this scheme, secondary users rely on *white spaces* either in the time or the frequency domain, taking advantage of the time and frequency locations where primary users are not making use of the resources assigned to them (this is the origin of the term *opportunistic*). Its main disadvantage is the fact that the resources are either available or not, so the radios must constantly scan over a band of frequency large enough to increase the probability of finding a white space. It also suffers from the hidden terminal problem, occurring when the power of a far transmitter does not reach the level required to be sensed, risking interference on a near receiver. To work around this problem in the case of TV white-space, the current regulation mandates the maintenance of a geo-location database to avoid interference to incumbent users [28].

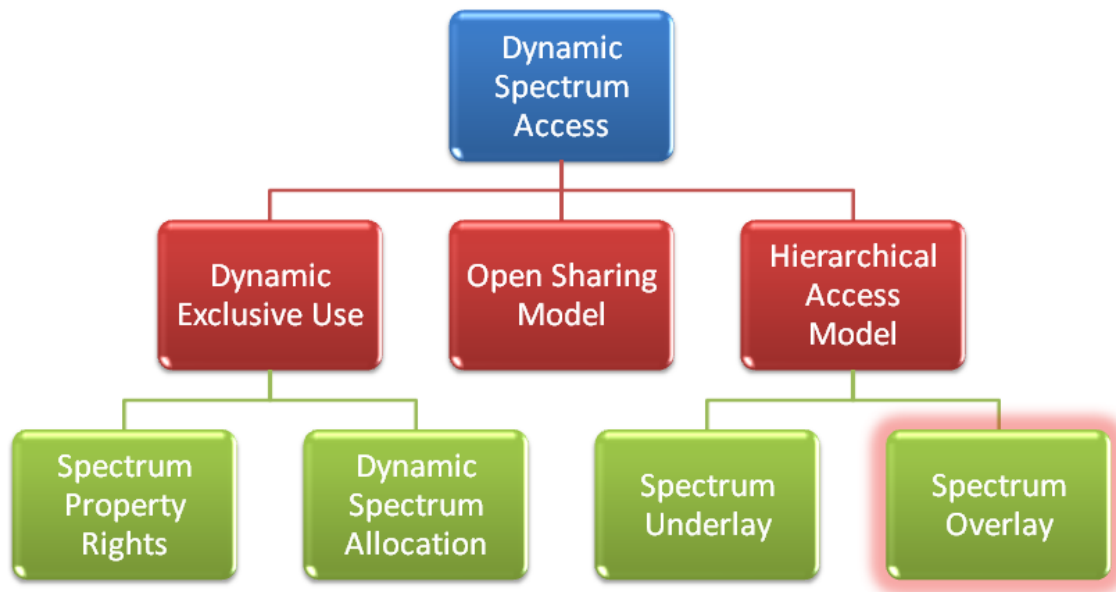


Figure 2.1: Taxonomy of dynamic spectrum access (after [1]).

A diagram showing the spectral behavior of hierarchical access systems is presented in Figure 2.2. The capacity of OFDM overlay systems is limited by their high side lobes, that produce mutual interference between the primary user (PU) and the secondary user (SU).

Mutual interference can be avoided in the $\text{PU} \rightarrow \text{SU}$ direction with expensive band-etching filters, discussed in Section 7.3. A good solution in the $\text{SU} \rightarrow \text{PU}$ direction is the use of pulse shaping, leading to the concept of filter bank multi-carrier methods, explained in detail in Section 2.3.

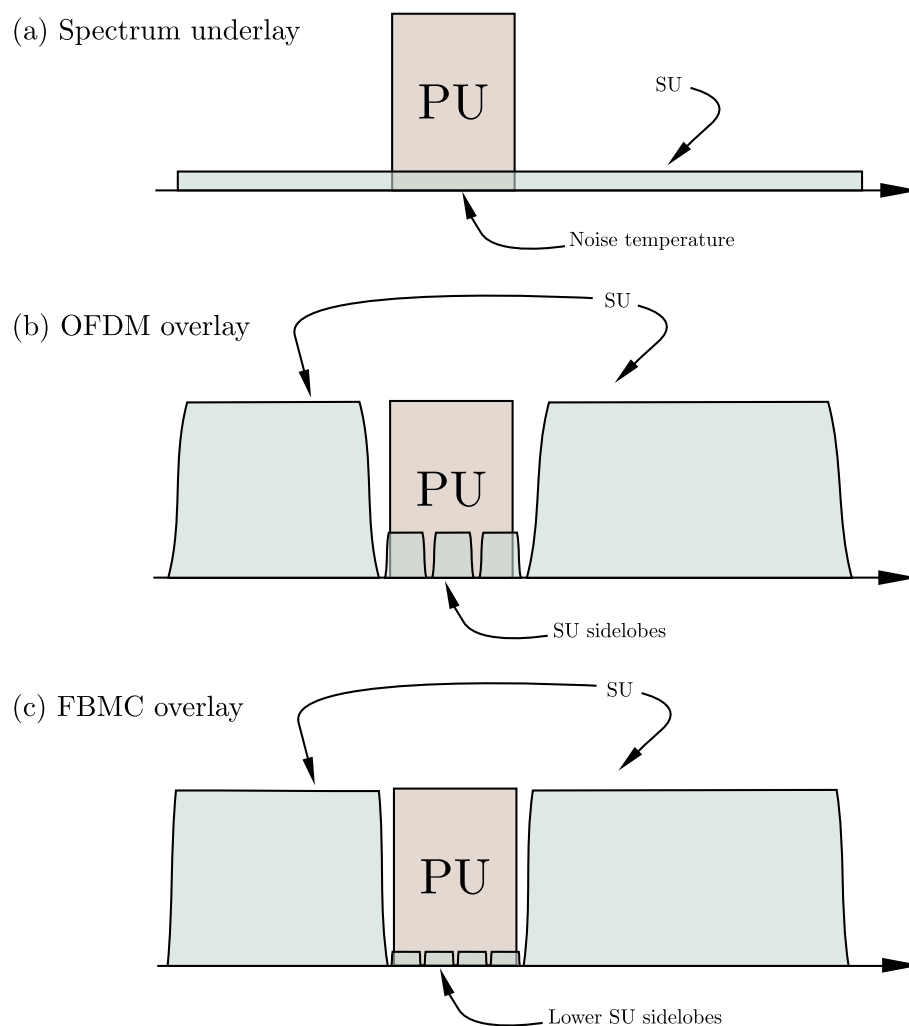


Figure 2.2: Hierarchical Access Methods

2.2 OFDM-based Overlay Systems

A summary of the state of the art in OFDM opportunistic access applications, and a summary of open problems are presented in the following subsections.

2.2.1 The Concept of OFDM

In OFDM, a serial data stream is mapped to symbols of any linear constellation, such as BPSK, 8-PSK, or 16-QAM. The resulting symbol stream is converted in parallel and fed to an IFFT operation, which is a computationally efficient way to modulate the parallel symbols into an orthogonal set of sub-carriers. The result of the IFFT is converted back to serial and a cyclic prefix is added at the beginning of each block, forming an OFDM symbol. This transmitter structure, presented in Figure 2.3 offers several degrees of support for adaptability: At mapping time, each sub-carrier may be assigned a different constellation type. For example, a bandwidth efficient constellation may be assigned to high SNR subchannels, while a low efficiency constellation may be assigned to the low SNR subchannels. This operation is called bit loading. Similarly, power loading is possible if the symbol amplitudes are configured according to the channel gain at each frequency.

The basic concept of OFDM overlay is to assign a zero amplitude to the subchannels that are in the same frequency as the primary user.

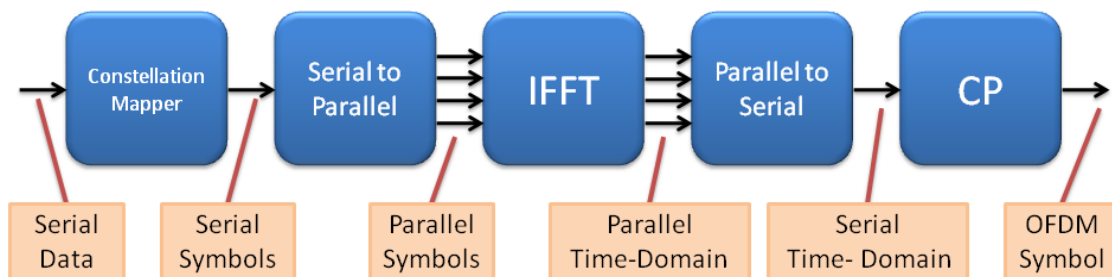


Figure 2.3: An OFDM Modulator

2.2.2 Clasification of Sub-carrier Allocations

Extensive work has been performed on sub-carrier allocation problems for OFDMA systems. In OFDMA, there are three classes of carrier assignment schemes (CAS) that are presented here given their applicability to the non-contiguous OFDM case [29].

- Localized (sub-band).
- Interleaved (uniformly spaced, comb).
- Generalized.

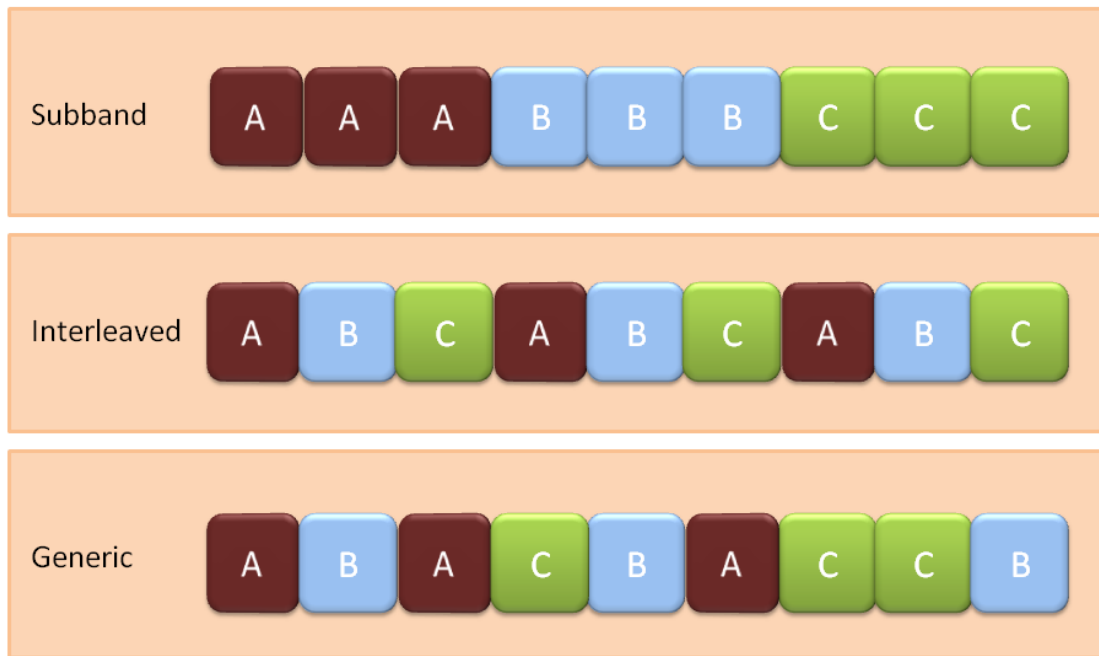


Figure 2.4: The three cases of OFDMA sub-carrier allocation

Each of the sub-carrier allocation schemes presented in Figure 2.4 for the case of three users (A, B, and C) has advantages and disadvantages.

The simplest scheme is the sub-band allocation, where the sub-carriers forming a sub-channel are located in a contiguous block in frequency; it has the advantage of simplifying channel

identification and synchronization tasks. It suffers from a low frequency diversity and a low capability for relocation, as the sub-channel must be reallocated as a block.

The interleaved scheme offers excellent frequency diversity; however, a periodic distribution in spectrum makes it unusable for exploiting secondary spectrum, as each user is potentially accessing the whole spectrum band.

The generic allocation scheme is the most flexible in terms of usage as an overlay system; its drawback being the computationally demanding procedures required to perform channel estimation and synchronization tasks. Hybrid schemes can also be considered, offering a coarser granularity in the reconfigurable blocks, by clustering sets of contiguous sub-carriers and implementing a block sub-band or a block-generalized scheme.

2.2.3 OFDM Overlay

An example of the operation of an OFDM overlay system is presented in Figure 2.5. In this scenario, Primary User 1 (PU1) and Primary User 2 (PU2) are active at the moment of sensing. The OFDM overlay assigns frequency sub-bands to data, DC, pilot, and cancellation sub-carriers. Transmission occurs until Primary User 3 (PU3) is detected, as presented in Figure 2.6. The sub-carriers that are interfering with PU3 must be turned off to avoid interfering with the primary user communications. Interleaving across sub-carriers and error correction codes can be used to recover the lost bits and maintain integrity, assuming that the interferer state information can be used to delete the affected symbols. Next, system recovery could be performed by re-allocating the sub-carriers or by migrating to a different band if the available number of sub-carriers is insufficient to sustain the information rate required by the system.

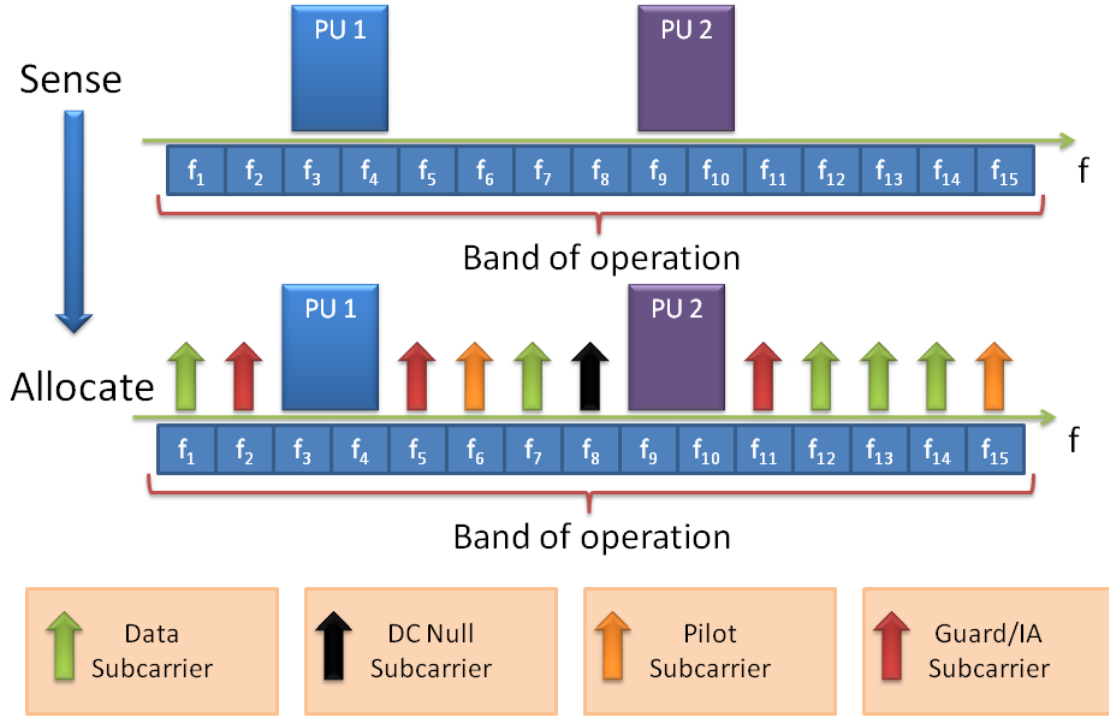


Figure 2.5: Sensing, allocation, and transmission

2.2.4 Challenges

The ideal case presented above is useful to establish a long term research objective; however, several practical considerations hinder this overlay network strategy, among them:

- The frame detection, fine timing, and frequency acquisition operations are strongly hindered by the primary users when performed in the time domain. Today's OFDM protocols assume that these operations are performed in the time domain.
- The primary users, which are non-orthogonal to the OFDM sub-carriers, generate interference to the secondary user due to the spectral leakage of the DFT. Interference occurs also from the secondary user to the primary user. This impairment is known as *mutual interference*.
- The arbitrary sub-carrier assignment required to avoid the interferers requires novel

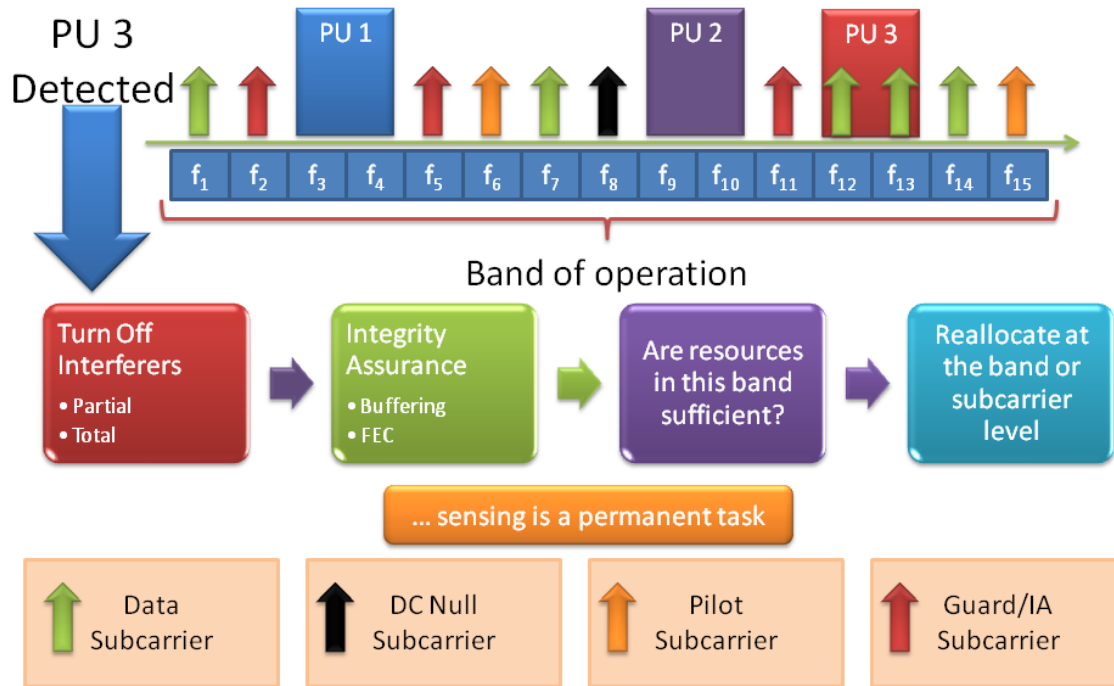


Figure 2.6: Detection of primary user activity and recovery

synchronization and channel estimation techniques.

- Spectrum awareness depends on the probabilities of detection and false alarm that are a function of the location of the sensors. It can be enhanced by cooperative sensing, at the cost of the overhead of sensing information transmitted among secondary nodes.
- A control channel is required to convey the sensing and sub-carrier allocation information. However, secondary users by definition don't have a bandwidth availability guarantee to hold this channel.

2.2.5 The Mutual Interference Problem

Non-contiguous reconfigurable OFDM is a candidate technique to implement frequency agile secondary use of spectrum. To make it applicable, practical considerations must include the mutual interference caused between the primary and the secondary users due mainly to the

finite time windowing of OFDM and to the lack of orthogonality of the primary user with respect to the OFDM sub-carriers.

Expressions for the mutual interference are derived by Weiss et al. and are extended below [30]. If we assume an OFDM modulated signal implemented with a rectangular time-domain pulse shape, we find a baseband representation of the power spectral density expressed as:

$$S_n(f) = A_n^2 T_s \text{sinc}^2[(f - f_n)T_s] \quad (2.1)$$

where $\text{sinc}(x) = \sin(\pi x)/(\pi x)$, T_s is the symbol duration, A_n is the baseband amplitude of the n th sub-carrier, and f_n is the frequency of the n th sub-carrier, assuming a zero guard interval.

If we use $\mathcal{N}$ to denote the set of active sub-carriers in the OFDM signal, and assume that a primary user is located in the frequency band $\mathcal{B}$, the interference caused to the primary user can be calculated as:

$$I_{SU \rightarrow PU} = \sum_{n \in \mathcal{N}} \int_{\mathcal{B}} S_n(f) df \quad (2.2)$$

To calculate the interference caused from the primary user to the secondary user, we must realize that in the absence of previous processing, the FFT is performed over a rectangular windowed time domain signal in order to obtain the frequency-domain symbols. The windowed version of the time domain signal $r_w(k)$ is represented as:

$$r_w(k) = r(k)w(k) \quad (2.3)$$

where $r(k)$ is the time-domain representation of the primary user, and $w(k)$ is a rectangular window of length N samples. The time domain product maps to a cyclic convolution in the frequency domain. Therefore, the interference from the primary user to the n th sub-carrier

of the secondary user can be calculated as:

$$I_{PU \rightarrow SU}^{(n)} = \frac{1}{2\pi N} \int_{-\pi}^{\pi} S_R(\theta_n) \left[\frac{\sin(\theta_n - \psi)N/2}{\sin(\theta_n - \psi)/2} \right]^2 d\psi \quad (2.4)$$

where $S_R(\omega)$ represents the primary user power spectral density, and θ_n is the frequency of the n th sub-carrier in the discrete-time periodic frequency domain.

2.2.6 Techniques for Mutual Interference Mitigation

The use of time-domain windowing at the transmitter is a proven technique to reduce the spectral leakage in OFDM, and can be used to mitigate $SU \rightarrow PU$ interference. Redfern analyzes the use of time-domain windowing functions at the receiver to reduce interference in the direction $PU \rightarrow SU$ [31].

Weiss et al. suggest the use of guard sub-carriers at the boundaries between the primary and the secondary users [30]. Brandes et al. augment this concept by the use of canceling sub-carriers designed using singular value decomposition methods that combine destructively with the spectral side lobes of the OFDM signal [32], achieving a side lobe suppression of more than 20 dB using two canceling sub-carriers at each border of the active set of sub-carriers. Further, Cosovic et al. propose applying weighting factors to the active sub-carriers [33] achieving side lobe reductions of 10 dB without adding additional canceling signals.

In subsequent publications, Cosovic et al. propose a mapping of the transmitted symbol sequence to a set of sequences and choosing a sequence out of the set with the lowest side lobe power [34, 35]. The method is named *multiple-choice sequences* (MCS). To provide a side lobe power reduction of 10 dB requires an overhead of 14%.

Senst et al. analyze the joint problem of side lobe reduction and PAPR mitigation, proposing the use of *reserved tones* located at the edges of the band [36]. The amplitude and phase of the reserved tones is calculated according to a joint minimization function.

Another technique for side lobe reduction, simulated in the environment of non-contiguous OFDM conditions is presented in the work [37], and consists in the dual application of constellation expansion and canceling sub-carriers. Constellation expansion consists in mapping low order constellations to subsets of points in higher-order constellations, and choosing the mappings that offer the lowest side lobes. The authors claim side lobe reductions of about 16 dB in a 32 sub-carrier OFDM signal.

A common characteristic in the methods presented up to this point is the reduction in efficiency caused by the transmission of additional data (canceling sub-carriers and multiple-choice sequences) or amplitude scaling (weighted sub-carriers). Bansal et al. make the point that minimization of interference to the primary user should be the primary objective of bit and power loading, relating two topics before treated separately and arriving to the conclusion that interference in the direction $SU \rightarrow PU$ can be reduced by allocating less power to the carriers close to the PU; the author demonstrated by simulation that properly allocated power can have better effects than just leaving blank sub-carriers at the boundaries between PU and SU [38].

2.2.7 A Multiuser Detection Approach for Interference Mitigation

The techniques proposed in the literature for interference mitigation assume that the signal characteristics of the primary user are unknown; however, if the interfering primary user employs a known modulation method, an alternative for interference mitigation in the direction $PU \rightarrow SU$ is to perform multiuser detection on the set of received signals, including the PU and SU. Two variants are proposed: The first is to implement a full receiver for the PU signal with the purpose of subtracting it from the received stream; the second is to use the output of the FFT bins located at the PU band to estimate and cancel the effect of the PU over the SU sub-carriers. These techniques are worthy of further research, especially in the case when the PU power is much higher than that of the SU.

2.3 Filter Bank Multi-carrier Overlay Systems

. The mutual interference problem produced by the high side lobes of OFDM increases the noise temperature of the primary user signal and reduces the effective capacity of the secondary users in shared spectrum applications. Filter bank multi-carrier methods can be considered an evolution of OFDM by the addition of generalized symbol shaping filters. This section presents a summary of references discussing the basic theory of filter bank methods and related work in their application to cognitive radio PHY.

A simplified diagram of a FBMC transmitter is presented in Figure 2.7. The signals $\bar{s}_k(n)$ for $k = 0..N - 1$ represent zero-padded complex symbol signals that feed a set of N identical shaping filters $h(n)$. Each of the shaped waveforms is modulated by the orthogonal sub-carriers $c_k(n) = \exp(j2\pi nk/N)$. The sum of the set of modulated signals produces the FDM signal. The set padded the signals $\bar{s}_k(n)$ are defined as:

$$\bar{s}_k(n) = \begin{cases} s_k(m) & \text{for } n = Nm \\ 0 & \text{otherwise} \end{cases} \quad (2.5)$$

where m is the temporal symbol index, n is the sampling index at the modulator output, and N is the FFT/IFFT size.

A null sub-carrier corresponds to a frequency bin with $s_k(n) = 0$. In a polyphase implementation, the non-zero padded symbol values $s_k(m)$ are fed to the IFFT operator, as in OFDM. The prototype filter $h(n)$ is decomposed in its polyphase components. Each branch of the polyphase decomposition FB is applied to the output of the IFFT operator, and the resulting signal is converted from parallel to serial (P/S). The mathematical justification of the equivalence and the analysis of the computational efficiency of this configuration when implemented on an FPGA is presented in subsequent chapters.

A prototype filter other than a rectangle function (see Figure 2.8) enhances the spectral side lobes by several decibels, adjustable according to the design specifications. In the case of

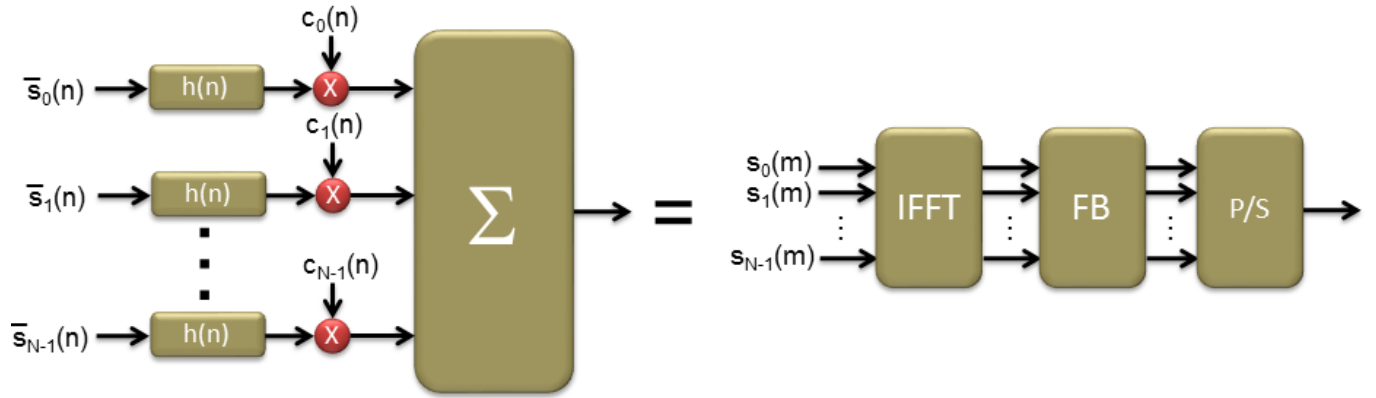


Figure 2.7: FBMC simplification

spectrally overlapped sub-carriers, the prototype filter response must be carefully designed to keep the orthogonality conditions. The figure shows an example root-raised cosine filter with a roll-off factor of 0.9 truncated to a duration of $4 * N$ samples (it spans four symbol periods) and weighted by a Kaiser window with parameter $\beta = 3.45$.

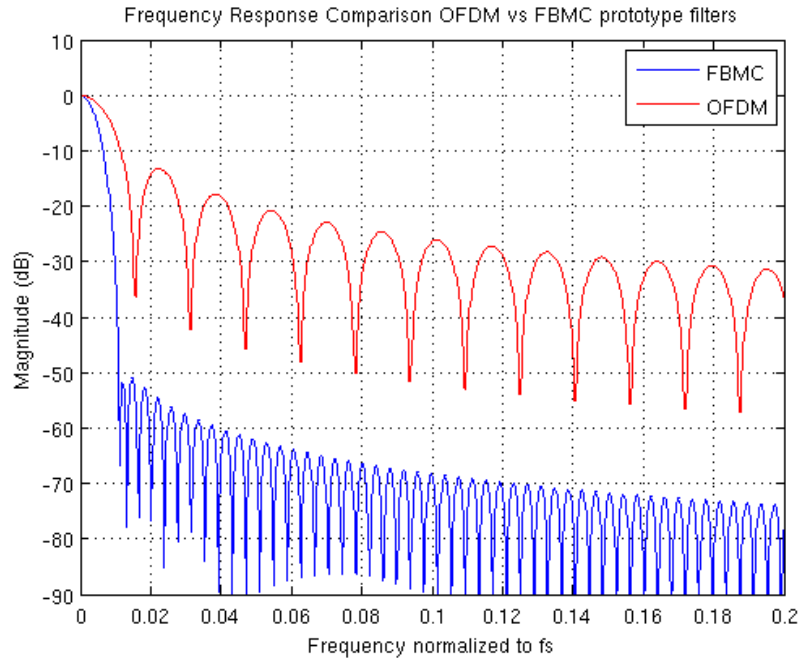


Figure 2.8: Response of an example prototype filter

2.3.1 Filter Bank Theory

An introduction to the theory of modulated filter banks for the implementation of critically and non-critically sampled FBMC systems is presented by Harris in [39] under the denominations of “polyphase channelizers” or “transmultiplexers.” The modulated filter bank converts a frequency-division multiplexed signal into a time-division multiplexed one. Serpentine shifts required on the filter bank state vector to obtain the sampling rate conversions required for arbitrary values of the prototype filter roll-off factor.

2.3.2 Introduction to FBMC Methods

In his master’s thesis, Matteo sets a general foundation for multi-carrier modulations, starting from discrete multi-tone with cyclic prefix (DMT with CP, commonly known as OFDM), and setting an evolutionary path through its variants such as Windowed OFDM, OFDM-OQAM, and Shaped OFDM [40]. The evolution from the traditional OFDM concept is produced by the addition of pulse shaping filters with an impulse response different from a square of duration equal to one symbol period. The author claims that these methods possess a lower sensitivity to residual carrier frequency offset (CFO). A review of each of these methods is presented next.

OFDM. OFDM can be generalized as a filter bank multi-carrier method with a rectangular filter shape of duration equal to that of a symbol. OFDM permits the use of a cyclic prefix to absorb intersymbol interference and the application of 1-tap equalizers per sub-channel by its implicit implementation of circular convolution. The use of a square filtering function saves the computation effort of a shaping filter, one of the reasons for the simplicity of OFDM.

The main drawbacks of OFDM are listed as loss of spectral efficiency by the insertion of the cyclic prefix, sensitivity to fast time variations of the channel and to carrier frequency offset, and high side lobes due to the use of a square prototype filter.

Windowed OFDM. The windowing operation can be analyzed as the insertion of a polyphase filter with duration of one symbol and non-constant impulse response. Common windowing functions such as the Hamming, Hanning, or Blackman windows can be used as prototype filters. Care must be taken when designing the filter response to maintain the orthogonality conditions.

Shaped OFDM and Filtered Multitone (FMT). Shaped OFDM uses prototype filters spanning more than one symbol time and does not require (or support) the use of a cyclic prefix. The author claims that the implementation of the required polyphase filter bank would significantly increase the system complexity; however, the results of this dissertation show that the contribution of the filter bank to the system complexity is marginal.

Critically sampled filtered multitone systems require the use of frequency-contained prototype filters to achieve a sub-carrier spacing equal to the symbol rate. Non-critically sampled systems have a relaxed prototype filter specification at the expense of spectral efficiency.

OFDM-OQAM. Offset QAM methods are applied to satisfy orthogonality conditions among sub-carriers, allowing frequency-domain overlapping of contiguous sub-bands. In this method, a careful design of the prototype filter response and phase rotation of successive sub-carriers are required to achieve isolation between sub-bands. The author analyzes the effect of defective time and frequency synchronization at the receiver.

The technical report presented by Du and Signell presents a comparison between OFDM and OFDM/OQAM systems with emphasis on the spectral efficiency of OFDM/OQAM systems and in the design of prototype filters with good time-frequency localization (TFL) [41]. The authors compare the rectangular, half cosine, Isotropic Orthogonal Transfer Algorithm (IOTA), and Extended Gaussian Function (EGF) prototype filters in terms of the ambiguity and the interference functions.

2.3.3 FBMC for Cognitive Radio

Amini et al. summarize the advantages of FBMC modulations for the implementation of cognitive radios [20]. The filtered multi-tone (FMT) and cosine modulated multi-tone (CMT) are compared to OFDM in terms of spectral efficiency and spectrum sensing capabilities. The lower spectral efficiency of FMT with respect to OFDM/OQAM is presented as a weakness that is compensated by its straightforward implementation. Both methods offer excellent spectral containment, a requisite for low mutual interference, as well as precise spectrum sensing by detecting only power located within the filter frequency response for each sub-carrier.

A tutorial paper by Farhang-Boroujeny and Kempter presents a review of filter bank multi-carrier methods that are suitable for the physical layer of cognitive radio [42] such as filtered multi-tone (FMT), cosine-modulated multitone (CMT), and staggered multitone (SMT or OFDM/OQAM). The authors emphasize the low side lobe amplitude achievable with FBMC, and its lower *interference temperature* with respect to OFDM. The multitaper method based on prolate filters is also proposed for spectrum sensing and compared to filter bank methods. An important advantage of the latter is that a modulated filter bank is already a component of the receiver and can be reused for spectrum sensing tasks.

In 2009, Zhang et al. discussed the benefits of OFDM/OQAM with respect to OFDM and windowed OFDM in terms of its spectral efficiency and of the interference level caused to the PU [43, 44]. For Zhang the concept of cognitive radio is equivalent to opportunistic spectrum usage. In these papers, the model of a modulated filter bank for OFDM/OQAM is presented as a departure point to obtain mutual interference metrics for IOTA and PHY-DYAS prototype filters. The authors find the data throughput of the studied modulation types using constant interference values and obtain experimental results supporting the claim that OFDM/OQAM has a larger spectral efficiency than OFDM and windowed OFDM for secondary user applications.

In a further paper, Zhang et al. calculate the spectral efficiency of OFDM/OQAM in an

uplink cellular environment, considering factors such as path loss, Rayleigh fading, and inter-cell interference (ICI) induced by timing offset. Simulations are based on the sequential processing of sub-carrier assignment and power allocation. The maximization of a metric called *Averaged Capacity* (AC) is proposed and evaluated under different ICI conditions.

2.3.4 Synchronization and Channel Estimation

Fusco et al. analyze the problem of data-aided joint symbol timing and carrier frequency offset estimation for FBMC systems [45]. They propose a solution based on periodic training sequences using a least squares approach applied to training signals in OFDM/OQAM and FMT. This strategy relies on an exhaustive search over their objective function to find a maximum, which may be difficult to implement in real-time.

Pilot-based synchronization along with channel estimation and equalization are discussed by Stitz et al. [46]. They propose a joint algorithm for the estimation of carrier frequency offset (CFO), fractional time delay (FTD), and channel response based on iterative interference cancellation. The authors manifest the requirement of using the filter bank properties for interference suppression before attempting to obtain synchronization in the frequency domain (instead of the traditional time-domain approach), but assume that coarse CFO and FTD estimates were already obtained in the time domain, and focus on frequency-domain approaches only to refine those estimates.

Two channel estimation methods for OFDM/OQAM based in preambles are proposed by Lele et al. [47]. Both methods are enhancements of the so-called *pseudo perfect estimation*, consisting of the estimation of the channel response by the sub-carrier by sub-carrier ratio of the transmitted and the received signals. The first method is called *interference approximation method* (IAM) and is based in the cancellation of the interference contribution of neighboring symbols. The second method is based on *pairs of real pilots* (POP) inserted at consecutive time positions on the same sub-carrier. Both methods require a previous estimate of timing, which is not considered in this work. Du and Signell elaborate on the

IAM method, providing a theoretical framework for preamble design [48].

2.3.5 Structures for Efficient Implementation

Tonello and Pecile discuss topologies for the implementation of analysis and synthesis filter banks for non-contiguous sub-carrier allocations in FMT [49], and remark the support for asynchronous users as a key advantage of FMT over OFDMA, as well as its inherent capability for spectrum shaping by nulling sub-carriers. Both of these advantages arise from the good spectral containment of FBMC methods. The authors obtain metrics for implementation complexity per user in several sub-carrier allocation scenarios.

Moret and Tonello present a comparison among three different FMT realizations and analyze the conditions of the prototype filter for perfect reconstruction and orthogonality. They evaluate the performance of the prototype filter under fading channels, finding a higher robustness of the IOTA filter with respect to the traditional root-raised-cosine filter [23].

The FPGA implementation of the filter banks for cognitive radio applications of FBMC is analyzed by Fahmy and Doyle [50]. They discuss the advantages of modulated filter banks with respect to OFDM for spectrum sensing, and present a hardware architecture for filter banks using heterogeneous resources found in FPGAs with sharing of the sub-band filter resources based on their low sampling rate. The filter state information is saved on RAM to enable the sharing of the multiply-and-add hardware resources.

2.4 Conclusions

An overview on the relevant literature on FBMC methods for general wireless and for secondary user applications was presented in this chapter. The main advantages of FBMC over OFDM were presented, mainly spectral containment and efficiency.

Common variants of FBMC such as FMT and OFDM/OQAM offer a trade-off between

implementation complexity and spectral efficiency, and their practical aspects such as implementation architectures, synchronization, detection, and frame detection have been recently analyzed in the literature.

Chapter 3

Configurable Computing for Radio Applications

FPGAs and ASICs offer unrivaled processing capabilities for data flow applications. They can be used for rapid prototyping, permitting the quick testing of designs, or as reconfigurable computers, providing the flexibility of software. These characteristics make these devices ideal for the implementation of software radio applications, but currently limited by price, power, and productivity. Extensive research has been performed in the definition of architectures and protocols for radio applications over reconfigurable hardware. This chapter summarizes relevant efforts related to FPGA-oriented software radio architectures and multi-carrier radio developments on FPGA platforms.

3.1 FPGA Software Defined Radio

Thanks to their extensive hardware resources, parallel operation, and field programmability, FPGAs are appropriate hardware platforms for the implementation of high-end software defined radios. This section presents a review of previous works directed towards the defi-

nition of architectures for flexible and modular FPGA-based radios and their interfaces to mainstream SDR architectures.

3.1.1 Layered Radio Architecture

A *layered radio architecture* for reconfigurable platforms is proposed by Srikanteswara et al. [51]. The advantages of run-time reconfiguration and over-the-air updates are presented. According to this work, the main considerations for the design of a reconfigurable architecture for radio are the following:

- Resource allocation
- Flexibility
- Reusability of hardware
- Data flow properties
- Scalability
- Replication

The architecture formalizes the concept of *hardware paging* in the same way that software uses memory paging. A common bus is used for data and for programming information. All signal processing is stream-based, which is claimed to simplify the interface between modules and to permit module replacement and insertion. The functions of the radio are separated into layers; top to down: soft radio interface (SRI), configuration, and processing. An application-layer software application running on a sequential processor sits on top of the layer stack.

The difficulties of implementing feedback connections existing in communication transceivers is presented; the proposed solution is to swap the direction of the data buses. This approach

may lead to performance penalties in time-tight synchronization tasks. The architecture provides flexibility in the building of radios at the expense of an overhead by packetizing the data.

3.1.2 Rapid Radio

An example of a spectrum-aware reconfigurable radio architecture on FPGA is the Rapid Radio project developed at Virginia Tech. It is based on a template-based parameterizable receiver [52, 53, 54]. In the listening mode, the system observes a telecommunications carrier selected by a user and performs parameter estimation based in the spectral shape of the signal. Departing from the estimated parameters, it passes the signal through an automatic modulation classification (AMC) stage. The classification algorithms work with no previous knowledge of carrier or symbol synchronization.

The host computer running the classification algorithm produces an XML description file of a suitable receiver which is used to generate tailored hardware by binding library modules. The modules are placed and routed using vendor tools to generate a receiver for the signal of interest. The framework includes a knowledge base encoded in CLIPS [55]. It contains rules used to choose the appropriate set of templates from the module library. A user can steer the classification process, and has access to graphical depictions of the metrics used for classification. If none of the possible modulation types fits the signal under analysis, the user can define a new modulation type using XML syntax, and plug it into the set of hypothesized modulations. Similarly, the system publishes the likelihood of each hypothesis, and lets the user decide among competing hypothesis before triggering the generation of a radio receiver.

3.1.3 Agile Hardware for Radio Development

Suris et al. propose a system for rapid assembly of modules in a run-time reconfigurable region called (the *sandbox*) by implementing run-time routing and bitstream assembly algorithms [27]. In a demonstration graphical interface, precompiled modules can be dragged and dropped into the sandbox. Connections between adjacent modules are routed automatically.

The original system was augmented by building a library of modules required to implement low and high speed wireless transmitters and receivers. Over-the-air reconfiguration capabilities were added, so a radio at one end of the communication can command the radio at the other end to change its configuration. If a side control channel is available, commands can be send without competing with the user data.

This project provides an agile development flow for the reconfiguration of FPGA radios. Given the right set of primitive modules in the library, a controller can quickly reconfigure the hardware in response to changing environmental conditions and standards.

3.1.4 Hardware Acceleration of GNU Radio

The GNU Radio project [56] provides blocks that can be connected and executed in a general purpose processor (GPP). While the low level DSP functions are written in C++, the module interconnection and parameterization is done using Python scripts.

GNU Radio is targeted to run on a standard PC with an interface to a Universal Software Radio Peripheral (USRP) RF front end. Although the USRP is based on an FPGA, this device is only used to perform upconversion and upsampling in the transmit path as well as downconversion and downsampling in the receive direction. All the baseband processing functions are performed on the PC, which is connected to the USRP via a USB cable (in the version 1 of the USRP), or via a gigabit Ethernet connection (in the USRP-2). The USRP is not specific to GNU Radio, and other development and simulation environments, as Simulink, also support it.

In his masters thesis, Irick realizes the shared library-based approach between GNU Radio and the Agile Hardware framework, and investigates the offloading of the signal processing functions from the PC to a high performance FPGA, which communicates directly to the USRP, demonstrating that the flexibility of GNU Radio can be applied to FPGA radios using the modular approach of Agile Hardware [57].

3.1.5 The Berkeley BEE2 Hardware Platform

Researchers at the Berkeley Wireless Research Center developed a platform for emulation and prototyping of high performance computing systems based on FPGA called the Berkeley Emulation Engine 2 (BEE2). It is a high-performance general purpose platform composed of five Virtex-II Pro 70 FPGAs, 20 GB of high-speed memory, and high-speed interconnection links [58].

Chang et al. present a review of applications for the BEE2 as a general-purpose hardware platform and emphasizes the need to use hardware description languages to maximize computational efficiency [59]. This paper presents benchmarks on sample applications implemented on four different hardware technologies. Some of the example applications include:

- Emulation and design of wireless communication systems.
- Digital signal processing.
- Scientific computing.
- Acceleration of CAD tools.

Mishra et al. presented the application of the BEE2 platform as a cognitive radio testbed with emphasis in the exploration of the spectrum sensing problem [60]. The metrics for the detection of a primary user are defined in this work as:

- Time to detection of a primary user.

- False negative and false alarm probabilities.
- The time required to clear the spectrum once a primary user is detected.

A multi-radio setup is proposed using the capability of BEE2 to connect to up to 18 RF front ends. No experimental results are presented in this work.

Tkachenko et al. use the BEE2 to develop and test spectrum sensing strategies [61]. An energy detector strategy is implemented using a periodogram that averages M observations obtained from the magnitude squared of a size $N = 256$ FFT performed at a sampling rate of 64 MHz, resulting in frequency bins separated by 250 kHz. A second spectrum sensing strategy based in cyclostationary features is also implemented by the calculation of the spectral correlation function (SCF). An experiment on collaborative spectrum sensing was performed using combined measurements obtained at several locations in a laboratory. The experimental cooperative gains measured are documented as part of the results.

Mellers et al. present an overview of spectrum sensing and radio telescope applications using the BEE2 [62]. These works use Xilinx System Generator as the preferred development tool.

3.1.6 The Winlab Cognitive Radio Hardware Platform

The WINLAB Network Centric Cognitive radio hardware platform (WiNC2R) is an effort of the Wireless Information Network Laboratory at Rutgers University [63]. It is composed of a set of RF, modem, networking, and data modules. The modem modules are formed each by an RF front end and a SX-series Virtex-4 FPGA used for physical layer baseband processing tasks; the networking modules are implemented using FX-series Virtex-4 FPGAs. The data modules are implemented with 32-bit RISC processors.

The WiNC2R platform is expected to support the following features:

- Multi-band operation, fast spectrum scanning, and frequency agility.

- Support for high-data-speed software defined radios.
- Spectrum policy processing for dynamic spectrum sharing algorithms.
- Different MAC algorithms.
- Support for multiple PHY/MAC instances.
- High throughput networking operations including ad hoc association and multi-hop routing.

3.2 Multi-carrier Radio on FPGA

Multi-carrier techniques such as OFDM and FBMC are becoming mainstream when high data rates and spectral efficiency is required. The flexibility provided by individual sub-carrier control, their built-in spectrum sensing capabilities, and the spectral containment achievable in the case FBMC make these techniques a match for the requirements of white space radios.

A pertinent area of research in FPGA multi-carrier radios is hardware optimization based in the regularity and parallelism of this communication scheme. Another main area of research gravitates around appropriate hardware architectures for the efficient partition of functions among heterogeneous hardware resources.

3.2.1 Techniques for Implementation of Multi-carrier Radios

A full OFDM transceiver for IEEE 802.11a with high-level descriptions of the modulator, demodulator, frame detector, and fine timing estimation was presented in 2003 by Dick and Harris [64]. The *System Generator* tool from Xilinx is used to implement the physical layer data flow processing. Harris et al. presented the theory of polyphase filter banks for generalized channel separation and sampling rate in multi-carrier systems [65]. In 2008, they

analyzed the topic of PAPR reduction using square-root Nyquist filters. The complementary code modulation (CCM) technique for PAPR reduction was presented in [66].

3.2.2 Hardware Reductions for Multi-carrier Radio

Hardware reconfiguration is an attractive strategy to provide flexible re-use of the FPGA resources; however, in the current development stage of this class of devices, parameterization is a more practical method. The choice of reconfiguration vs. parameterization is a common question that appears in hardware reductions. This section presents a review of techniques to make an efficient use of resources in configurable computing hardware for the implementation of non-contiguous multi-carrier transceivers based in two prominent characteristics: FFT sparseness and non-concurrency of tasks.

FFT sparseness

The computational efficiency of the Fast Fourier Transform is derived from the regularity of the Discrete Fourier Transform when divided in simpler computations (butterflies). Each butterfly requires a single multiplication with a twiddle factor, additions, and data reorganization at each stage of the algorithm. In the case of a multi-carrier transmitter with zeroed sub-carriers, further simplifications may be achieved by pruning the butterfly graph paths corresponding the zero input values. Analogously, in a receiver only a subset of the FFT outputs is required. The corresponding paths can also be pruned. The concept is illustrated in Figure 3.1.

The computational savings obtained by pruning have been analyzed on patterned sub-carrier allocations. Shousheng et al. propose the evaluation of the comb spectrum required for an interleaved carrier allocation scheme [10]. It is optimized by performing a regularized pruning operation. The proposed algorithm requires $O(N + M \log M)$ multiplications, where N is the FFT size and M is the number of required sub-carriers. This work analyzes the behavior

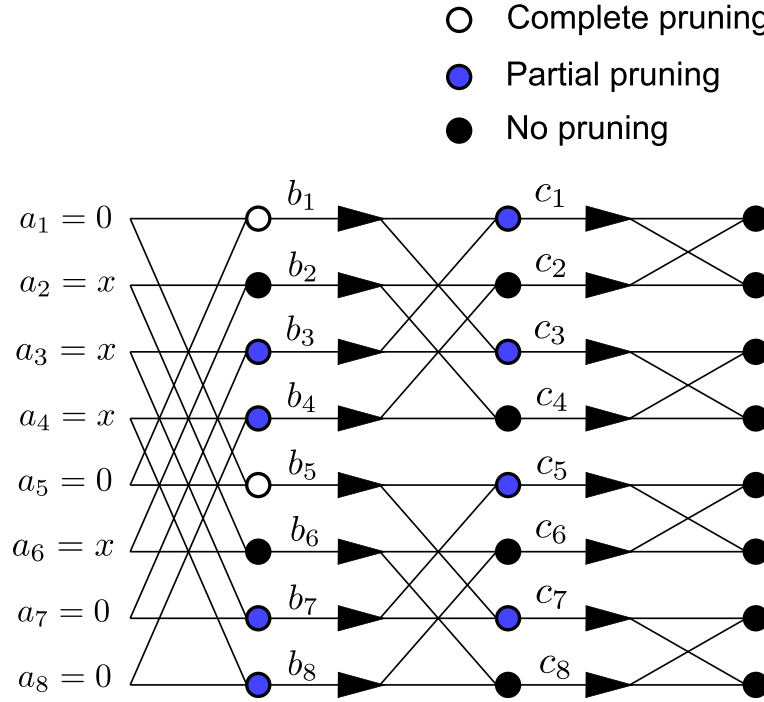


Figure 3.1: Pruning of nodes of the FFT algorithm (after [2]).

of a static implementation of the pruned FFT, without consideration of system dynamics.

An FFT pruning technique is also presented by Rajbanshi et al. [2]. The algorithm can be applied to arbitrary zero-input patterns. Despite the reduction in multiply and add operations, the burden of the control operations is increased. Computational savings are only achieved for a high level of sparseness. The authors claim that the algorithm performs better than a conventional FFT when the number of deactivated sub-carriers is large (over 60%), for an FFT size of $N = 1024$. Further, the algorithm does not consider the case when the FFT algorithm is executed on parallel hardware.

Two low-complexity FFT schemes were presented and compared by Kim et al. [9]. One of the methods was suited for subband and generalized carrier allocation schemes (CAS), and uses a transform decomposition in blocks of three FFT butterflies that are assembled according to the particular sub-carrier allocation. The second method, called the transposed transform decomposition, gives good results for interleaved CAS. Computational savings are

only obtained if the number of active sub-carriers is below $N/4$. Following this work, a hardware implementation of the transform decomposition method is described by Qiwei et al. [8].

It is not possible to take advantage of the sub-carrier allocation sparseness if the FFT core is re-used for spectrum sensing functions; as all frequency samples at the output of the FFT module are required to obtain observations of the spectrum across the band of interest.

In conclusion, it is not practical to simplify the FFT module at the receiver side in a spectrum sensing radio with FFT re-use. Simplifications done at the transmitter side in the OFDM and FBMC cases can produce demonstrated computational savings only under high sub-carrier sparseness conditions.

Non-concurrent tasks

FFT simplification has a limited applicability in practical FPGA radios. A potential source of area savings is the re-use of hardware for non-concurrent tasks. A set of mappings of the functions for OFDM over shared hardware on reconfigurable machines was presented by Pionteck et al., leveraging the common characteristics of the wireless LAN standards IEEE 802.11a and HiperLAN/2 and using the fact that not all the processes are active at the different parts of a burst [67]. Pionteck et al. proposed three mappings for non-concurrent tasks:

Mapping A. Comprises the synchronization task, based in the calculation of the autocorrelation function and a CORDIC module for carrier frequency offset (CFO) correction.

Mapping B. Formed by the equalizer, channel estimation, and soft-demodulator.

Mapping C. Consists of a timing offset estimator based on the slope of a MMSE-fitted first order polynomial over the phase measured at the pilot sub-carriers, as well as the equalizer and a channel estimation tracking algorithm.

The reuse of hardware related to the FFT and error correction modules was not considered in this work.

Hardware parameterization techniques are presented by Alaus et al. as a means to enhance the software/hardware sharing balance [68]. Parameterization can be implemented using *Common Function* and *Common Operator* techniques. Common Function refers to the sharing of a function among different standards, where all the components with similar functionality are put together in the same parameterizable module. This permits the construction of multi-standard radios that accept a set of parameters to activate a given standard. A finer search for common patterns reveals similar hardware structures among dissimilar functionalities. This approach is called Common Operator.

Examples of the Common Function approach are the design of a common constellation mapper for a wide set of modulation types [68] and the reuse strategies proposed by Saponara et al. by finding common FFT operator characteristics among the DVB-T/H, xDSL, IEEE802.11a/n, and UWB standards [69]. Re-use is enabled by a configurable macrocell, parameterizable in terms of the IFFT/FFT size and word length. The macrocell is based on a cascade of radix-4 IFFT/FFT modules.

Examples of the Common Operator approach found in the literature are the parameterization of the FFT operator for filtering tasks, to implement Reed-Solomon codes [70], and to perform the Viterbi algorithm [71].

3.2.3 Heterogeneous Architecture Partitions

Helmschmidt et al. exposed the need for a mix of sequential processors and reconfigurable hardware for the implementation of OFDM transceivers [72]. The modem and coding functions are split in different computing machines given the high computational load of each task. For the modem function, they reached hardware optimizations based on the fact that several tasks don't overlap in time and can therefore benefit from hardware multiplexing. As

an example, preamble detection is time multiplexed with the demodulation module. Computations were partitioned among a microcontroller, DSP, FPGA, and a reconfigurable array processor.

The subject of OFDM task partition was also presented by Ebeling et al., where the configurable hardware was used for synchronization acquisition while the demodulation task was not active (and vice versa) [73]. Different optimizations were proposed under a custom coarse grained architecture called RaPiD.

Mennenga et al. proposed a layered architecture and the use of a Configuration Management Module (CMM) to coordinate the reconfiguration tasks; however, no implementation details were provided [74].

Zhang et al. discussed the implementation of spectrum sensing for the discovery of licensed user activity in a cognitive radio [75]. Different spectrum sensing architectures were proposed: scan-based, real-time FFT based, and scanning FFT based. This work proposed the adaptability of the number of sub-carriers, cyclic prefix, and bit loading per carrier, and blank carriers, defined as carriers with zero bit loading. A heterogeneous computing architecture was proposed to develop the transceiver. The hardware architecture is composed of tiles with elements as GPP, domain specific configurable hardware, DSP, FPGA, and ASIC. The authors proposed partitioning the OFDM transceiver into communicating functional processes, but further details were not presented.

Nikdash et al. analyzed the overhead caused by the fine granularity of FPGAs and proposed a heterogeneous architecture named Reconfigurable Heterogeneous System (HERS) [76]. A reconfigurable modem engine was optimized to time-multiplex the functions of filtering, resampling, scrambling, FFT and correlation calculations. A coding engine implemented various channel coding functions, as Viterbi decoding, Turbo coding, Reed-Solomon and LDPC. In accordance to the observations of Helmschmidt et al. [72], Nikdash remarks that one reconfigurable machine is not enough to manage the functions of modem and coding. The HERS architecture uses sequential and data flow elements (DSP and Fine Grain Blocks)

together to form a processing element (PE). It also proposed a hierarchical interconnection between PEs. The architecture can support the IEEE 802.11 and DVB-T standards.

3.3 FPGA Architecture and Programming Model

FPGAs are historically characterized for a fine-grained granularity, which allows a high level of parallelism, and have evolved to a heterogeneous architecture. Contemporary FPGAs as the Virtex-6 contain a mix of LUT, DSP slices, block RAM, and built-in communications interfaces for connection to other devices. Fine granularity is obtained at the expense of having most of the device area used for wires and switching devices. Device configuration is not as granular: the minimum configuration unit is a frame. Further, reconfigurations must be accomplished by reloading the configuration RAM.

Traditional FPGA programming is accomplished at low level using Hardware Description Languages such as VHDL or Verilog. Graphical tools such as Core Generator and Simulink based System Generator are of great help for the designer when the use of canned modules is possible. The designer is responsible for the appropriate connection of the modules and for guaranteeing the fulfillment of timing constraints. Therefore, the FPGA programming flow requires a mix of low and high level digital design skills, making the task a specialized one.

3.4 Conclusions

This chapter presented background sections on the application of FPGAs for Software Defined Radio, FPGA specific optimizations for OFDM transceivers, and a summary on FPGA architectures. The state of the art presented in the chapter shows that FPGAs are ready to become mainstream for the implementation of high bandwidth wireless systems for base stations, and that many optimizations and architectural enhancements are still possible,

particularly for multi-standard radios.

At the present stage of FPGA technology, partial reconfiguration techniques depend on long bitstream download times that exclude the possibility of intra-frame read-modify-write of the hardware. For illustration, consider a contemporary FPGA device such as the Virtex-5 LX330. This device has a full bitstream size of 9.5 MB. Assuming a write speed of 100 MHz and a bus width of 32 MB, writing a partial bitstream to reconfigure 10% of the device requires 23.75 ms. In the flavor of FMT used in this dissertation, the duration of a symbol is only $6\mu s$. A speed-up of 4000 times would be required to achieve intra-frame reconfigurations even at modest data rates. Partial reconfiguration of FPGA-based transceivers is still useful for multi-standard radios that allow a longer time to switch the radio operation mode.

This chapter presented hardware parameterization techniques known as common function and common operator as an alternative to run-time reconfiguration. They are an appropriate set of methods for run-time hardware reuse thanks to their capability for quickly switching the parameters or the function of a module. These methods require a specific module design to achieve re-parameterization, precluding the use of canned module libraries.

Chapter 4

Introduction to Cognitive Radio

This chapter presents an overview and fundamental issues on the topic of cognitive radio. The ideal cognitive radio as originally defined by Mitola [6] is presented, as well as related work in the fields of cognitive networks and multi-carrier cognitive radio.

4.1 The Ideal Cognitive Radio

Software defined radios (SDR) are implemented using signal processing functions in software, as opposed to radios based on fixed hardware components. SDRs are inherently flexible and potentially inexpensive, as they can be implemented in generic hardware. Software implementation facilitates the adaptation of physical level functions according to specifications dictated by upper-layer functions, for example the connection to a database containing past performance data, or containing geographically referenced information about local spectrum usage. Mitola described the evolution of software defined radio to cognitive radio by the evolution through the following capabilities [6]:

Awareness. A wireless terminal that correlates different sensing categories, for example RF spectrum occupation with location, is RF-location aware. Awareness can be used to

predict or reinforce the information from the environment.

Adaptation. Adaptations can happen once a terminal is aware of the environment. Continuing the example of RF-location awareness, spectrum occupation can be associated to location. In this case, RF-location awareness could help the wireless terminal prioritize white space search in those bands that historically offer the best performance for a given location.

Cognition. A cognitive radio should be able to learn from the environment and produce adaptation rules based on its experience under a very general set of objectives, such as offering the best quality of service or the lowest communication cost at a baseline service quality.

The ideal cognitive radio described by Mitola performs the functions of the seven-stage cognitive cycle depicted graphically in Figure 4.1:

1. **Sensing.** The capability to sense not only the spectrum, but location, temperature, and any other environmental parameter that provides insight into the requirements for best communication of the wireless user.
2. **Perception.** The capability to judge the information collected by the sensors.
3. **Orienting.** Assessing the perceived situation and determining if it is familiar. Looking for the set of operation parameters required under the perceived conditions.
4. **Planning.** Being capable of deliberately constructing a list of alternative actions that might be performed in the future under a set of potential scenarios.
5. **Making decisions.** Choosing the best action to perform.
6. **Taking action.** Producing changes on the environment, such as transmitting on a specific frequency band, or asking the user or other machines for additional information or commands.

7. Learning autonomously. Learning from experience at executing the capabilities in the cognitive loop.

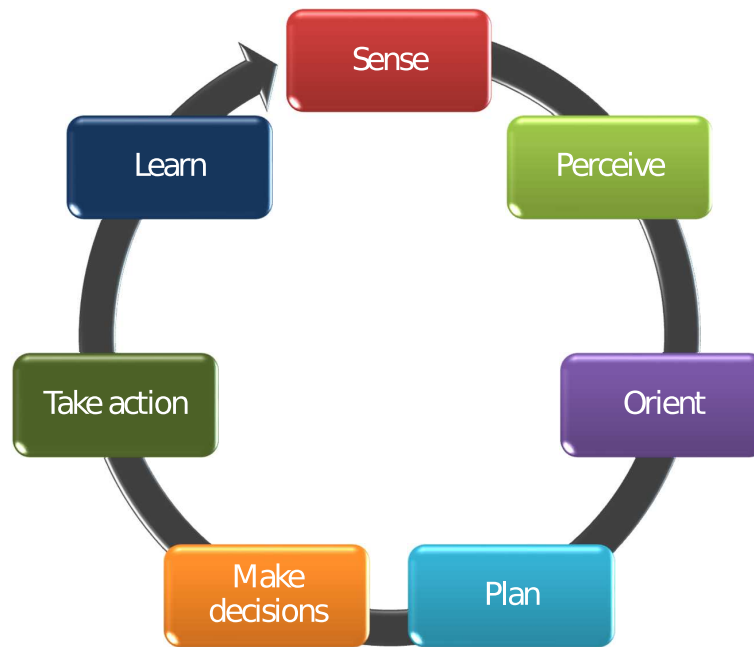


Figure 4.1: The cognitive cycle

This is a quite detailed model, conveying a complete cognitive model applied to living beings. Simpler models as the OODA loop (Observe, Orient, Decide, and Act) can also be applied to cognitive radios.

4.2 Spectrum Sensing

Sensing has the objective of permitting a single element or a set of elements to be aware of its environment. In multi-carrier overlay systems sensing has two main instances: Previous to carrier allocation, sensing allows the network elements to identify the permissible sub-carrier frequencies. During data transfer, sensing permits the detection of primary user activity: the bandwidth used by a leaving PU may be added to the available set of frequencies.

Similarly, an entering PU will interfere with the current communication and demands not to be interfered; therefore, this band segment must be subtracted from the available set of frequencies.

4.2.1 Spectrum Sensing Strategies

There are several approaches to spectrum sensing, based on varying levels of knowledge about the signals present in the observed band. The following is a brief explanation of different techniques available. The general objective of spectrum sensing is to identify the “white space” in an observed band, defined as the spectrum that remains after detecting the bands occupied by the primary users, as presented in Figure 4.2. Energy detection methods rely on the measurement of the power spectral density, which does not require knowledge of the characteristics of the primary user signals. Better results can be obtained if the characteristics of the potential primary users are known, as presented in the following summary.

Energy detection or radiometry

This strategy is based in the analysis of the amplitude of the bins of a FFT performed over an observation window [77]. This method is simple but presents difficulties for the election of decision thresholds in a changing environment. It has the potential advantage of re-using the FFT operator already employed in multi-carrier receivers [78]. A diagram of this sensing scheme is presented in Figure 4.3. Sensing is performed by calculating the FFT over an observation window of length N samples over the complex baseband signal denoted by $\bar{r}(\ell)$. A thresholding operation is performed over the magnitude squared of the frequency domain representation of the signal. The result is the decision vector $\bar{d}(\ell)$.

The FFT operator produces leakage of the primary user power over neighboring bands when orthogonality conditions are not fulfilled. A solution to this problem is the use of filter banks

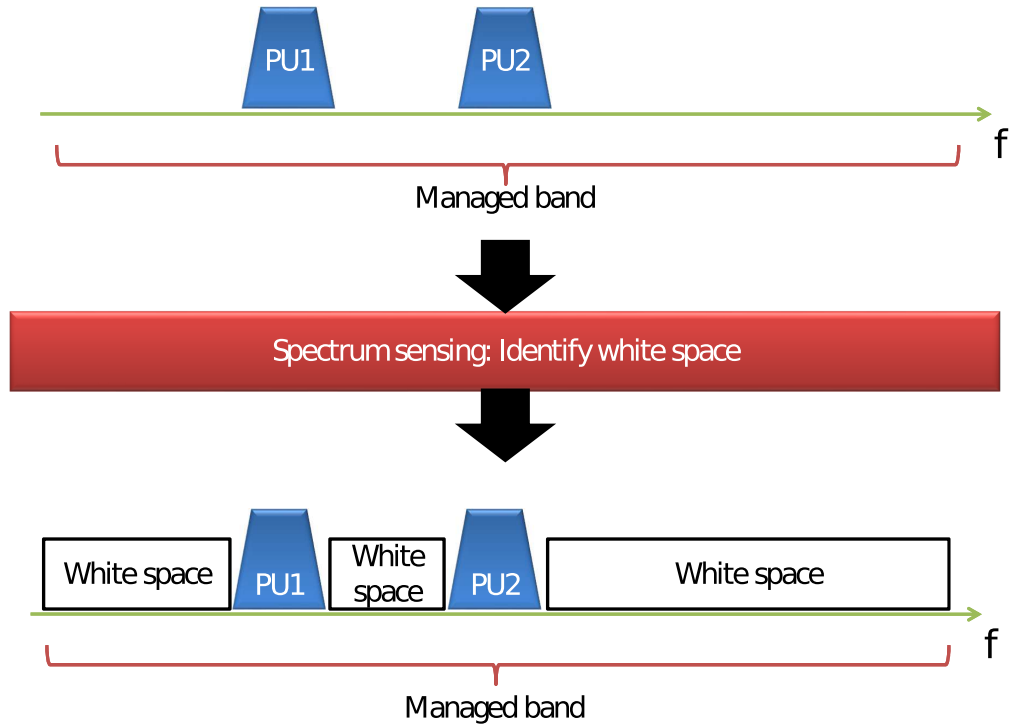


Figure 4.2: White space in a managed band.

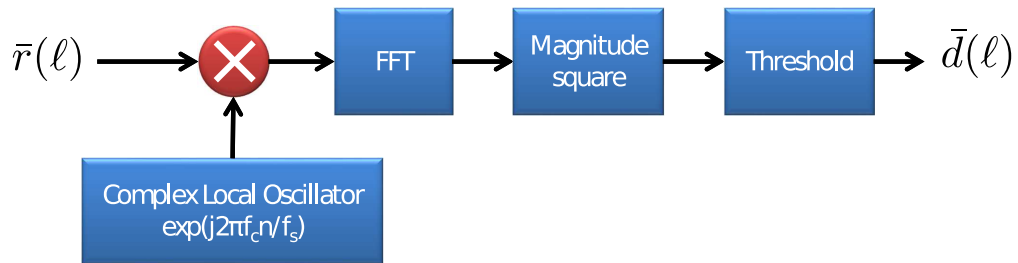


Figure 4.3: Energy detection spectral sensing.

with better spectral containment derived from the frequency response of the prototype filter presented in Figure 2.8. An enhanced version of the energy detector with high spectral containment is presented in Figure 4.4.

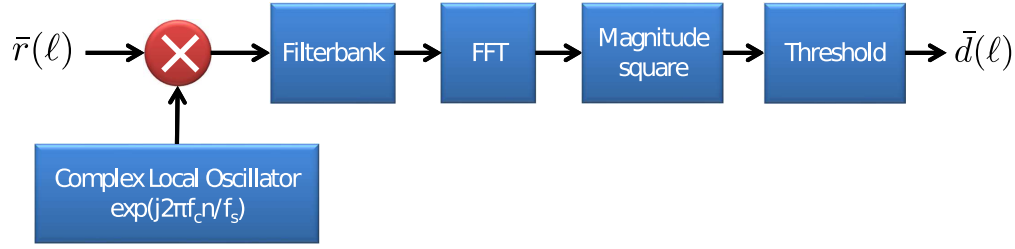


Figure 4.4: Energy detection spectral sensing with filter banks.

Matched filter detection

This method requires the implementation of a receiver for each of the potentially expected PU signals. To match-filter a signal the receiver must contain the appropriate receiver filter and demodulator structure. Several parameters must be known *a priori*, for example the center frequency f_c , symbol rate f_s , modulation type, and the roll-off factor α . This rules out environments where the specifications of the interferer signals are unknown [77]. Figure 4.5 presents a block diagram of a matched filter receiver. A matched filter receiver is required for each possible primary user signal.

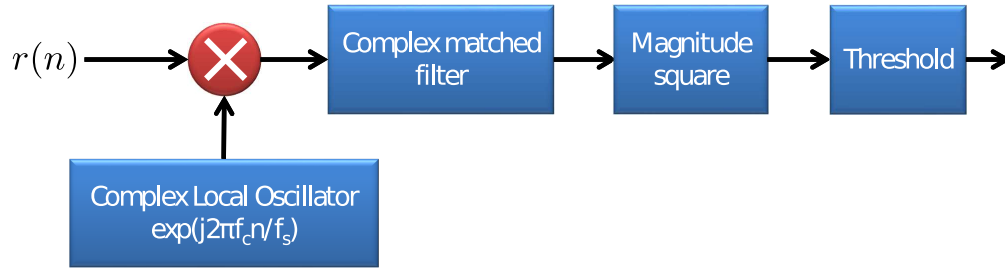


Figure 4.5: Spectral sensing with matched filtering.

Detection of cyclostationary features

The calculation of cyclostationary features requires a high computational power, and presumably, synchronization to the PU signal [79]. Further, the training of a Hidden Markov Model (HMM) pattern matching is required for classification. Its main advantage is the ability to detect signals with very low SNR. The main cyclostationarity-based metrics proposed

by Kim et al. are the spectral correlation function (SCF), spectral coherence (SC), and cycle frequency domain profile (CDP). Figure 4.6 shows a schematic of a signal detector based on cyclostationary features.

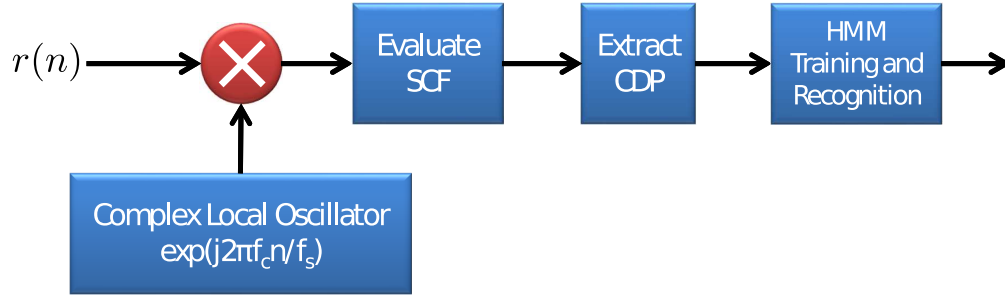


Figure 4.6: Spectral sensing based on cyclostationary features.

Energy detection with partial matched filtering

Spectrum characterization using this strategy is based in the estimation of the bandwidth and center frequency of the primary users. The estimated parameters are used to search a database of known waveform types. This method, presented in Figure 4.7, is named “partial match-filtering algorithm” by Yücek and Arslan [3].

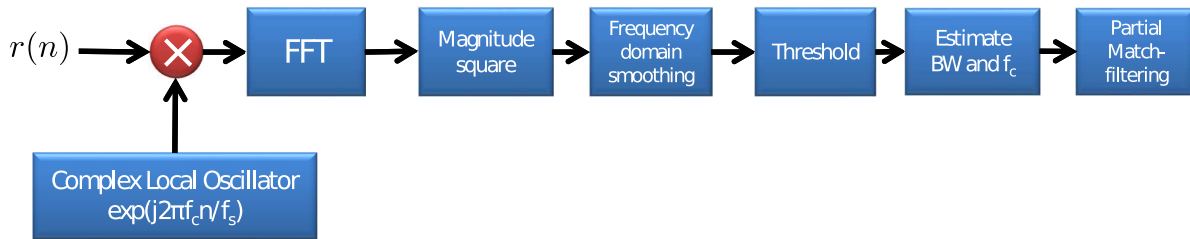


Figure 4.7: Spectral sensing based on partial matched filtering (after [3]).

4.2.2 Distributed Spectrum Sensing

Accurate spectrum sensing measurements requires direct observations of the primary user that are not always achieved because of effects such as shadowing and frequency selective

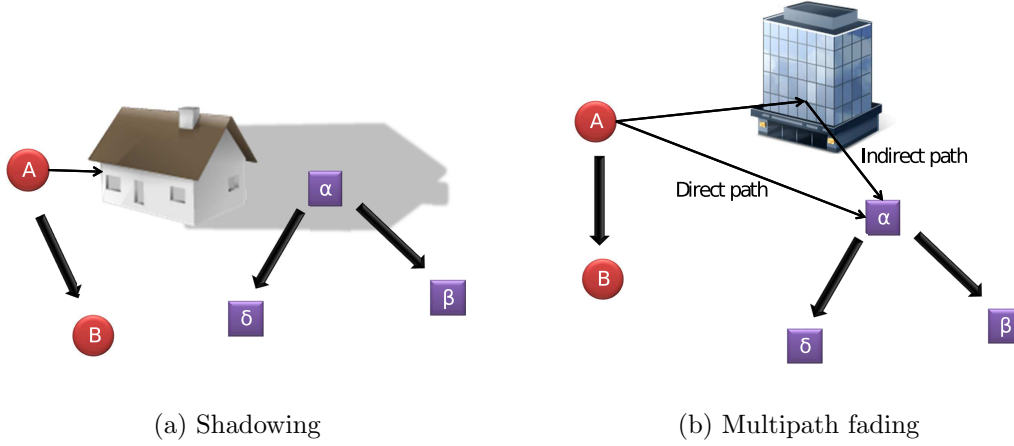


Figure 4.8: Impairments in Spectrum Sensing

fading.

Shadowing occurs when an obstacle is present between the primary user and the sensor. As observed in Figure 4.8a, secondary user α does not receive power from primary transmitter A . If α decided to transmit without further information, it would interfere secondary receiver B .

Frequency selective fading is produced by multipath fading. In this situation, some frequencies add destructively, causing the erroneous perception that they are available for secondary usage. In Figure 4.8b, the superposition of the direct and the indirect paths from A to α produces frequencies with destructive interference that may be sensed as white space.

Shadowing and frequency selective fading restrict the quality of the observations obtained from a single user. The use of multiple sensors distributed in space permits diversity gains by combining their measurements. Figure 4.9 presents a scenario where a secondary transmitter α cannot detect the signal of a primary transmitter A due to signal attenuation, interfering B when transmitting to β , and causing β to receive a corrupted version of the intended signal. Sensor network concepts can be used in this scenario to perform distributed sensing by all the elements in the network, which enhances the detection and the false alarm probabilities,

achieving a multiuser diversity gain [80, 81, 1]. A protocol for the exchange of spectrum sensing information is proposed by Weiss [82].

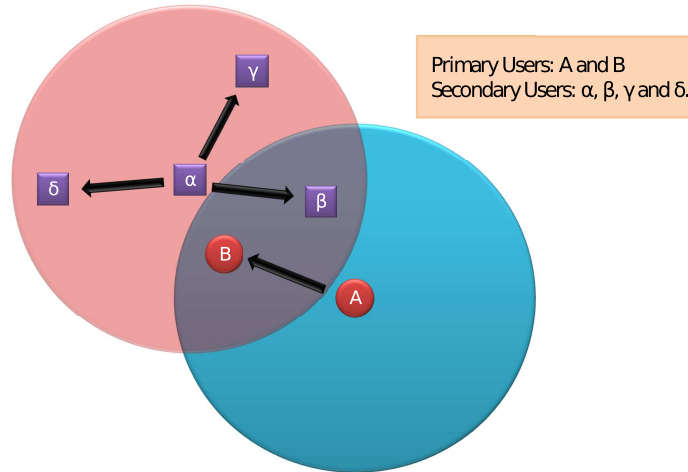


Figure 4.9: Diversity in Spectrum Sensing (after [1]).

4.2.3 An Opportunity for the Application of Machine Learning

Spectrum sensing is one of the components of cognitive radio that provides an opportunity for machine learning to be used. If the sensing measurements are related to the time of day, location and the user's traffic pattern, substantial gains in terms of detection performance can be achieved by modeling and machine learning. Examples are presented by Renk and Xiaoyu [83, 84].

4.3 Cognitive Networks

As Neel exposed, network decisions taken independently by a set of cognitive terminals can lead to catastrophic scenarios [85]. In his work, the interactions between cognitive radios were modeled and analyzed using techniques such as optimization and game theory. He provides a set of algorithms related to power control, frequency selection, interference avoidance, and network formation.

Thomas analyzed the interactions among network elements from a game theory perspective, and proposed the modeling of a network cognitive process consisting of software agents that have both autonomous and cooperative traits, and identified three characteristics of the elements that influence the performance of the network: their selfishness, their degree of ignorance, and their amount of control over the network [86]. He developed a model to find the equilibrium conditions obtained under different structures and individual element characteristics.

In his PhD dissertation, Rondeau presented an overview of artificial intelligence methods that can be applied to cognitive radio [87]. This work focused on the evaluation of candidate techniques such as neural networks, hidden Markov models, fuzzy logic, evolutionary algorithms, and case-based reasoning for the implementation of a cognitive engine to control the sensing, learning, and multi-objective optimization tasks required for the adaptation of the radio to its environment.

The optimized radio configuration was described in an XML file, which is passed to an implementation engine. This work claimed that OFDM is a promising waveform for the implementation of the physical layer of a radio thanks to abundant parameters that can adapt to the environment as cyclic prefix length, sub-carrier separation, modulation per sub-carrier, number of data sub-carriers, and channel coding; however, it did not consider the adaptation of the sub-carrier allocation scheme in a multi-carrier overlay system scenario.

4.4 Multi-carrier Cognitive Radio

Newman et al. analyze the performance of genetic algorithms (GA) for single and multiple-carrier cognitive radios [88]. This work proposes a set of fitness functions and analyze their behavior under a set of controllable transmission parameters. The controllable parameters in this work are: transmit power, modulation type, and modulation index. Environment awareness is provided by sensing bit-error-rate, signal-to-noise ratio and noise power. The

fitness objectives are the minimization of bit-error-rate, maximization of throughput, and minimization of power consumption. The application of a GA to the case of opportunistic spectrum usage is not considered in this work.

In 2009, Shaat and Bader discuss the use of multi-carrier cognitive radios in opportunistic applications [89]. An iterative power allocation algorithm is proposed to maximize the capacity of the system under total power and interference constraints. Such *power-interference* (PI) algorithm is evaluated under OFDM modulation with single and multiple-user scenarios.

Shaat and Bader continue in 2010 the discussion about the problem of resource allocation in multi-carrier-based cognitive radio networks, this time considering OFDM and FBMC methods for transmission in the down link direction [90, 91]. They derive the PI suboptimal algorithm for resource allocation, and claim that it achieves a performance close to optimal in the FBMC case. In conventional multi-carrier radios, the water filling power allocation is known to maximize the channel capacity. In spectrum overlay-based cognitive radios, the interactions between the Primary User (PU) and Secondary User (SU) must be considered, namely, spectrum band usage and mutual interference.

The mutual interference per sub-carrier is analyzed with respect to the set of frequencies occupied by the PU, as well as the multiple channels involved ($SU \leftrightarrow SU$ and $PU \leftrightarrow SU$). The resulting algorithm assigns power even to sub-carriers located in the PU bands if the interference $SU \rightarrow PU$ constraints are satisfied. The authors claim that the proposed algorithm outperforms previously published results.

It is important to note that even if the word *cognitive* is used in the works by Shaat and Bader, there are no autonomous learning capabilities integrated to their algorithms. What these works present instead is a set of resource allocation algorithms for combinatorial optimization of the sub-carrier power using the channel capacity as the objective function.

4.5 Conclusion

Extensive research work is currently performed in cognitive radios and cognitive networks; however, only a few approaches to the topic of multi-carrier cognitive radio are found in the literature.

The subject of system dynamics of cognitive radio is not covered in the works reviewed. The algorithms chosen for sensing, sharing of spectral measurements, radio configuration, and machine learning must respond rapidly to the changing environment to assure quality of service and minimum disruption to the primary users. In this frame set, multi-carrier radios may have a deep impact, as the single parameter of adaptation relative to the primary user usage is the allocation vector, defined as the set of sub-carrier locations to activate at a given time.

Chapter 5

Spectrum Awareness

Spectrum awareness is the process of gathering information about the occupation of the target frequency bands. A summary of related work on spectrum sensing techniques and their shortcomings was presented in Chapter 4. This chapter presents two techniques for estimation of the primary parameters that provide enhanced levels of detail about the primary user signal characteristics after a detection has occurred. The relationship between PU detection, PU modeling, and automatic modulation classification in terms of the information extracted at each level of processing is presented in Figure 5.1.

Modeling of the primary user spectral shape. This technique is proposed to provide information about the center frequency, symbol rate, roll-off factor and signal-to-noise ratio of a primary user, augmenting the information provided by energy detection when there is not enough information about the expected signal to perform matched filtering detection. It uses spectral power density measurements to perform curve fitting of a theoretical carrier shape according to a hypothesized modulation family.

In the case of linear modulations, a signal can be parameterized by its power, signal-to-noise ratio, carrier frequency, symbol rate, and roll-off factor. These parameters are used to fit a modeling function to the sensed spectral shape using a non-linear

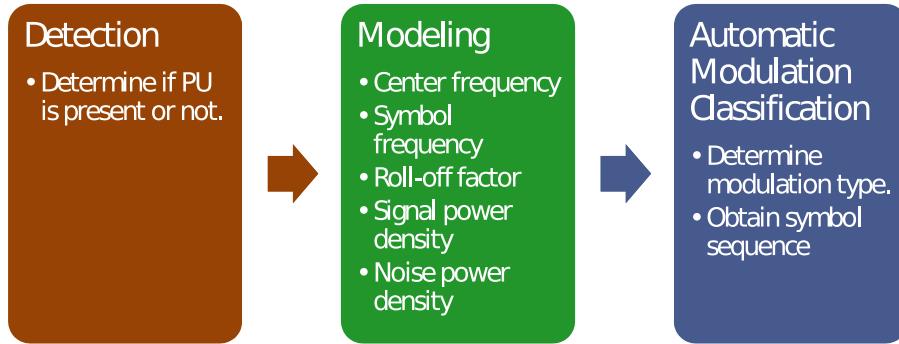


Figure 5.1: Tiers of PU modeling

optimization algorithm. A description of the method was published in [92]. The method was applied in the Rapid Radio project as a first step in the synthesis of a radio receiver for a signal with unknown parameters and modulation type [54]. An in-depth description and performance results of the algorithm are presented in Section 5.1.

Automatic modulation classification for active interference cancellation. A secondary receiver may be designed to closely track the primary users. The PU signals can be demodulated to extract information such as frame length, and to collect information about temporal occupation. Further, active cancellation of interference can be achieved by local generation of the PU signal using the developments of the Rapid Radio Project [54]. While this method requires a high computational power, modern FPGAs have extensive hardware resources that can be used for the implementation of several receivers in the same chip. This level of complexity might be justified in high-end applications. A detailed explanation of the algorithm and the metrics developed for automatic modulation classification are presented in Section 5.2.

5.1 Modeling of the primary user spectral shape

This section presents a method for estimating a set of modulation parameters of a signal in the class of the digital linear modulations, using the periodogram and non-linear optimization

methods. Low bias estimates of the carrier frequency, symbol frequency, and roll-off factor can be obtained under flat fading channels in low SNR environments. The proposed technique was validated with over-the-air laboratory experiments in the 2.05 GHz band. The variance of the estimates is calculated as a function of the $(C + N)/N$ ratio. An estimate of the signal-to-noise ratio can be obtained from the set of estimated parameters. The spectral shape of the PU signal is not distorted under flat fading conditions; therefore, the following sections assume that the channel shape is affected by a constant fading amplitude in the bandwidth of the PU.

The estimates of the carrier frequency (f_c), symbol rate (f_s), roll-off factor (α), and carrier plus noise to noise ratio $((C + N)/N)$ can accurately describe the front end of a radio receiver suited for the signal of interest. Analysis in the frequency domain using the Discrete Fourier Transform seems as an appropriate method to obtain the aforementioned parameters; however, it is known that the DFT is not a consistent spectral estimate [93]. A technique for variance reduction of the DFT spectral estimate without producing distortion to the underlying shape is temporal averaging of multiple DFT observations, known as the periodogram.

Section 5.1.1 describes the problem of parameter extraction from the spectral representation of a modulated signal. Section 5.1.2 describes the estimation technique proposed and its theoretical limits, while Section 5.1.3 presents the advantages of performing spectral sensing using the proposed method based in the calculation of the sensitivity of threshold-based power sensing methods to variations in the PU signal and noise power. The performance of the method under simulation and over-air experiments are presented in Section 5.1.4.

5.1.1 Description of the Problem

The complex baseband signal $\tilde{s}(t)$ at the receiver can be modeled as:

$$\tilde{s}(t) = \gamma(t) \left(\sum_{m=-\infty}^{\infty} a_m g(t - mT) \right) e^{j2\pi f_c t} + w(t) \quad (5.1)$$

where T is the symbol time, $\gamma(t)$ is the time-varying flat fading channel response, a_m are non-correlated complex random symbols with zero mean, $g(t)$ is a root raised cosine pulse shaping filter with roll-off factor α , f_c is the residual carrier frequency error, and $w(t)$ represents additive white Gaussian noise (AWGN) with variance σ_w^2 .

The squared magnitude of the Discrete Fourier Transform (DFT) is chosen as a power spectrum density (PSD) estimator over a time window NT , where N represents the number of samples of the DFT:

$$\hat{S}_\ell(k) = \left| \sum_{n=0}^{N-1} \tilde{s}(t_0 + \ell NT_s + nT_s) e^{-j2\pi kn/N} \right|^2 \quad (5.2)$$

where t_0 is the initial time of an observed window, T_s is the inverse of the sampling frequency F_s , and ℓ is the index of observation windows of length N samples each. Rectangular windowing is used to minimize the bias induced by frequency domain smoothing.

If the time window NT_s is commensurate to the channel coherence time T_c , the fading term can be considered as a realization of a random variable during the ℓ^{th} observation window. This condition also relies on the flat fading nature assumed for the channel.

$$\gamma(t) = \gamma_\ell \quad \text{for} \quad t \in (t_0 + \ell NT_s, t_0 + (\ell + 1)NT_s) \quad (5.3)$$

The expected value of $\hat{S}_\ell(k)$ is therefore:

$$E[\hat{S}_\ell(k)] = E\left[|\gamma_\ell|^2 N \frac{|G(k)|^2}{T} + W(k)\right] \quad (5.4)$$

where $G(k)$ is the DFT of the pulse shaping function $g(nT_s)$ and $E[W(k)] = N\sigma_w^2$. Considering the independence between the channel fading and the transmitted signal this expression can be simplified as:

$$E[\hat{S}_\ell(k)] = E[|\gamma_\ell|^2] N \frac{|G(k)|^2}{T} + N\sigma_w^2 \quad (5.5)$$

The ratio $R = \frac{F_s}{B}$ between the sampling frequency and the signal bandwidth $B = (1 + \alpha)f_s$, where $f_s = 1/T$ is the symbol rate, must provide an acceptable frequency resolution. In a complex baseband model, an N point FFT contains N/R samples describing the carrier of interest. If $R \approx N$, the number of frequency samples used to describe the signal of interest may be as small as 1, prohibiting the use of the spectral shape to extract the features of the signal.

The squared magnitude of the DFT is an inconsistent spectral estimate for a general random process: it is non-central χ^2 distributed with an expected value equal to the true power spectrum, and its variance is equal to the square of the expected value [94],

$$E \left[\hat{S}_\ell(k) \right] = S(k) \quad (5.6)$$

$$\text{Var} \left[\hat{S}_\ell(k) \right] = S^2(k) \quad (5.7)$$

An algorithm for parameter extraction from the PSD estimate $\hat{S}_\ell(k)$ must deal with its high variance and with the heterogeneity of the variance across the observed frequency band.

5.1.2 An Algorithm for Signal Feature Extraction

This section describes the set of operations performed to obtain the estimation of the parameters of a linearly modulated signal based on its spectral shape. They assume that high spectral resolution, obtained by down conversion and down sampling, as well as the calculation of the periodogram for estimation of the power spectral density.

The primary user signal is modeled as a raised cosine power spectral shape, that has a non-linear dependency on its parameters; hence, a non-linear optimization method is proposed to reach a solution using the Levenberg-Marquardt iterative method [95]. A numerical regularization is required to avoid ill-conditioning of the problem. The variance of the

estimates can be estimated using a multivariate Gaussian approximation.

Down conversion and down sampling

To enhance the resolution of the observation, down conversion and down sampling are required before the calculation of the PSD estimate.

Estimation of the power spectral density

The unbiased periodogram is calculated as

$$P(k) = \frac{1}{NL} \sum_{\ell=0}^{L-1} \hat{S}_{\ell}(k) \quad (5.8)$$

The variance of the periodogram is known to be

$$\text{Var}[P(k)] = P^2(k)/L \quad (5.9)$$

According to Equation 5.4, the expected value of the periodogram is:

$$E[P(k)] = E \left[\frac{1}{L} \sum_{\ell=0}^{L-1} \left(|\gamma_{\ell}|^2 \frac{|G(k)|^2}{T} + \frac{W(k)}{N} \right) \right] \quad (5.10)$$

$$E[P(k)] = E[|\gamma_{\ell}|^2] \frac{|G(k)|^2}{T} + \sigma_w^2 \quad (5.11)$$

Equation 5.11 shows that the fading term affects the power of the signal component without modifying its shape. Under this condition, no bias is introduced to the estimates.

Estimation over a raised cosine spectral shape

The analytic model of the raised cosine spectrum is fitted to the unmodified periodogram $P(k)$ of the received signal. The carrier frequency (f_c), symbol rate (f_s), roll-off factor (α), the level of the top of the carrier A_{up} , and the noise level A_{dn} are the variables required to describe the model. The ratio A_{up}/A_{dn} is an estimator of the $(C + N)/N$ ratio that can be used to estimate the symbol energy to noise spectral density ratio, E_s/N_0 .

The raised cosine power spectral response $y(f, \beta)$ is defined as a piece-wise function of frequency with parameters $\beta = \{f_c, f_s, \alpha, A_{up}, A_{dn}\}$, according to Equation (5.12). It is presented in Figure 5.2,

$$y(f, \beta) = \begin{cases} A_{dn} & \text{for } f \in [0, f_1) \\ A_{dn} + \mu \left[1 - \cos \left(\frac{f-f_1}{\alpha f_s} \pi \right) \right] & \text{for } f \in [f_1, f_2) \\ A_{up} & \text{for } f \in [f_2, f_3) \\ A_{dn} + \mu \left[1 + \cos \left(\frac{f-f_3}{\alpha f_s} \pi \right) \right] & \text{for } f \in [f_3, f_4) \\ A_{dn} & \text{for } f \in [f_4, \infty) \end{cases} \quad (5.12)$$

where $\mu = (A_{up} - A_{dn})/2$, and

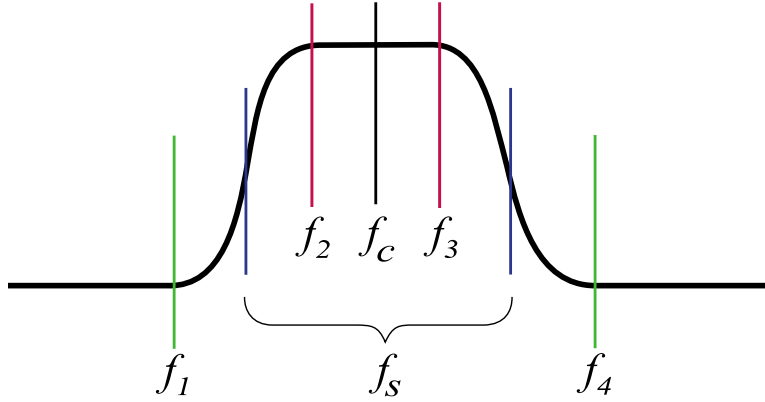
$$f_1 = f_c - \frac{1}{2}(1 + \alpha)f_s$$

$$f_2 = f_c - \frac{1}{2}(1 - \alpha)f_s$$

$$f_3 = f_c + \frac{1}{2}(1 - \alpha)f_s$$

$$f_4 = f_c + \frac{1}{2}(1 + \alpha)f_s$$

The model can be fitted at the frequency samples $f(k), k \in \{0 \cdots N - 1\}$ of the unmodified periodogram $P(k)$. This is a classic non-linear optimization problem expressed as:

Figure 5.2: Raised-cosine function $y(f, \beta)$

$$\hat{\beta} = \arg \min_{\beta} \left\{ \frac{1}{2} \|y(f(k), \beta) - P(k)\|_Q^2 \right\}, \quad k \in \{0 \cdots N-1\} \quad (5.13)$$

where N is the FFT size, y and P are arranged as column vectors, and Q is a diagonal weighting matrix with elements calculated as the inverse of the variance $V(k)$ of the periodogram at the true coefficient set.

The estimate of the positive definite matrix Q at the n th iteration $Q_n = \{q_{ij}^{(n)}\}$ is defined as:

$$q_{ij}^{(n)} = \begin{cases} 1/V^{(n)}(k) & \text{for } i = j = k + 1 \\ 0 & \text{for } i \neq j \end{cases} \quad (5.14)$$

As this coefficient set is unknown, the variance can be estimated using Equation 5.9, the periodogram variance at the n th iteration is estimated as:

$$V^{(n)}(k) = [y(f(k), \hat{\beta}_n)]^2 / L \quad (5.15)$$

where $\hat{\beta}_n$ is the set of coefficients estimated at iteration n .

Non-linear optimization

Non-linear optimization problems are typically solved with Newton-type iterative methods that require the calculation of a local linear model formed with the Jacobian and the Hessian matrices. The analytic calculation of the Jacobian is obtained from the partial derivatives of the analytic description of the raised cosine spectrum.

Following the local model of the cost function can potentially cause unbounded iteration steps. Consequently, a trust-region strategy is appropriate in this situation. Several trust-region methods are available. The Levenberg-Marquardt algorithm is a common choice due to its simplicity [95]. The general form of the Levenberg-Marquardt iteration for the generalized least squares problem is:

$$s_n = - \left[J(\hat{\beta}_n)^T Q_n J(\hat{\beta}_n) + \lambda_n \mathbf{I} \right]^{-1} J(\hat{\beta}_n)^T Q_n r_n \quad (5.16)$$

$$\hat{\beta}_{n+1} = \hat{\beta}_n + s_n \quad (5.17)$$

where the frequency term was dropped for simplicity, $J(\hat{\beta}_n)$ denotes the Jacobian evaluated at $\hat{\beta}_n$, and s_n denotes the update to the parameter set, and r_n represents the residual $r_n = y(f(k), \hat{\beta}_n) - P(k)$.

Observe that the true matrix Q is not known during the iterative procedure. A technique to solve the problem is to replace Q for its local approximation Q_n , calculated from the set of values $\hat{\beta}_n$.

A value of the trust-region size parameter $\lambda_n \geq 0$ is guaranteed to exist, such that $\|s_n\| = \Delta_n$, where Δ_n denotes the radius of the trust region. Instead of calculating the optimal value of Δ_n , a simplified update to the value of λ_n can be implemented as follows:

Further details of the method are presented in [96] and [97].

Algorithm 1 Trust-region size parameter update

if $|r_n| \leq |r_{n-1}|$ **then**

$$\lambda_{n+1} \leftarrow \frac{1}{2}\lambda_n$$

else

$$\lambda_{n+1} \leftarrow 2\lambda_n$$

end if

Calculation of the Jacobian

The elements of the Jacobian matrix J are the partial derivatives $j_{kl} = \partial y(f(k), \beta) / \partial \beta_l$. The expression of the Jacobian for general values of the frequency f is presented in Equations (5.18) to (5.22).

$$\frac{\partial y(f, \beta)}{\partial f_c} = \begin{cases} 0 & \text{for } f \in [0, f_1) \\ \mu \left[\frac{-1}{\alpha f_s} \pi \sin \left(\frac{f-f_1}{\alpha f_s} \pi \right) \right] & \text{for } f \in [f_1, f_2) \\ 0 & \text{for } f \in [f_2, f_3) \\ \mu \left[\frac{1}{\alpha f_s} \pi \sin \left(\frac{f-f_3}{\alpha f_s} \pi \right) \right] & \text{for } f \in [f_3, f_4) \\ 0 & \text{for } f \in [f_4, \infty) \end{cases} \quad (5.18)$$

$$\frac{\partial y(f, \beta)}{\partial f_s} = \begin{cases} 0 & \text{for } f \in [0, f_1) \\ \mu \left[\frac{-(f-f_c)}{\alpha f_s^2} \pi \sin \left(\frac{f-f_1}{\alpha f_s} \pi \right) \right] & \text{for } f \in [f_1, f_2) \\ 0 & \text{for } f \in [f_2, f_3) \\ \mu \left[\frac{(f-f_c)}{\alpha f_s^2} \pi \sin \left(\frac{f-f_3}{\alpha f_s} \pi \right) \right] & \text{for } f \in [f_3, f_4) \\ 0 & \text{for } f \in [f_4, \infty) \end{cases} \quad (5.19)$$

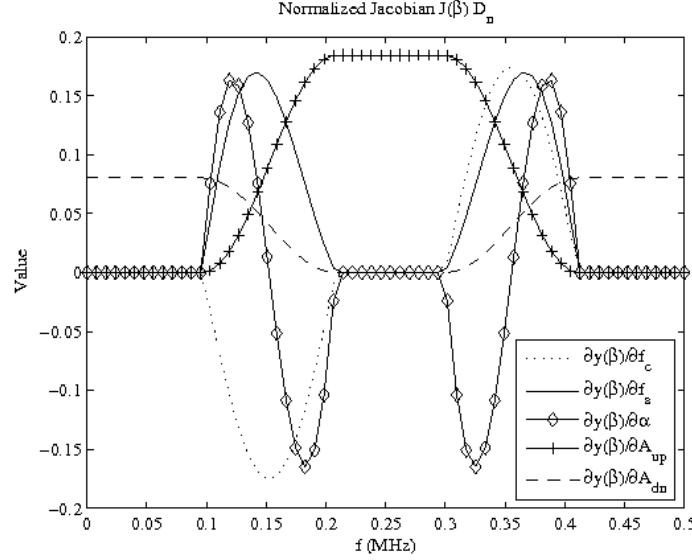


Figure 5.3: Normalized Jacobian matrix for $f_c = 250$ kHz, $f_s = 200$ kHz, $\alpha = 0.55$, $(C + N)/N = 3$ dB

$$\frac{\partial y(f, \beta)}{\partial \alpha} = \begin{cases} 0 & \text{for } f \in [0, f_1) \\ \mu \left[\frac{-(f - f_c + f_s/2)}{\alpha^2 f_s} \pi \sin \left(\frac{f - f_1}{\alpha f_s} \pi \right) \right] & \text{for } f \in [f_1, f_2) \\ 0 & \text{for } f \in [f_2, f_3) \\ \mu \left[\frac{(f - f_c - f_s/2)}{\alpha^2 f_s} \pi \sin \left(\frac{f - f_3}{\alpha f_s} \pi \right) \right] & \text{for } f \in [f_3, f_4) \\ 0 & \text{for } f \in [f_4, \infty) \end{cases} \quad (5.20)$$

$$\frac{\partial y(f, \beta)}{\partial A_{up}} = \frac{y(f, \beta) - A_{dn}}{A_{up} - A_{dn}} \quad (5.21)$$

$$\frac{\partial y(f, \beta)}{\partial A_{dn}} = 1 - \frac{\partial y(f, \beta)}{\partial A_{up}} \quad (5.22)$$

Figure 5.3 presents an example of the coefficients of a Jacobian matrix. The matrix inversion complexity in this implementation is determined by the number of parameters to estimate and not by the number of frequency samples.

Start-up parameter values

A set of parameter values need to be used to initialize the algorithm. A simple way to obtain the initial parameters is to use the minimum value of the observed window to initialize the parameter A_{dn} and the maximum to initialize A_{up} . The center frequency f_c startup value is given by the location of the maximum value. A small value is used as a starting value the symbol rate f_s (a percentage of the observed band). Finally, almost any value of the roll-off factor α can provide a starting point for the iteration.

Numerical regularization

Scaling is required to avoid ill-conditioning of the matrix inversions. A scaling matrix D_n based in the diagonals of the matrix $B_n = J(\hat{\beta}_n)^T Q_n J(\hat{\beta}_n)$ can be defined as:

$$d_{ij} = \begin{cases} 1/\sqrt{b_{ij}} & \text{for } i = j \\ 0 & \text{for } i \neq j \end{cases} \quad (5.23)$$

where d_{ij} are the elements of the scaling matrix D_n and b_{ij} are the elements of the matrix B_n at the n^{th} step of the iteration.

To apply the normalization the Jacobian is post-multiplied by the scaling matrix before every update calculation: $\tilde{J}(\hat{\beta}_n) = J(\hat{\beta}_n)D_n$, where the symbol $\tilde{J}$ indicates a scaled Jacobian matrix. After a normalized step is obtained, the update to the parameter vector $\hat{\beta}_n$ is de-normalized as $s_n = D_n \tilde{s}_n$.

Problem-specific constraints are introduced into the iteration to guarantee consistency of the parameter set. For instance, the noise floor estimate A_{dn} cannot be above the carrier top estimate A_{up} . Every iteration update is inspected for consistency before accepting an update to the parameter set. An example curve fitting is presented in Figure 5.4.

A further regularization that can be introduced to avoid diverging results is to check the singular values of the normalized Jacobian matrix. Very small singular values that may pro-

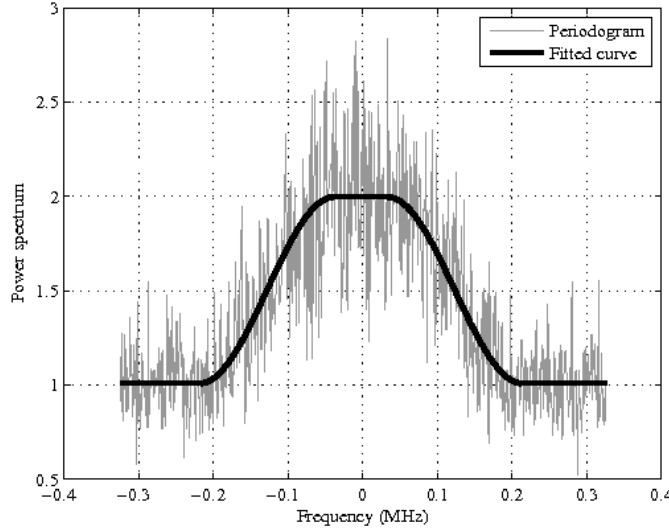


Figure 5.4: Example fitted carrier under parameters of Table 5.1 and a $(C + N)/N$ of 3 dB.

duce large update steps can be replaced by values close to unity. This type of regularization is similar to the effect of the parameter λ_n the Levenberg-Marquardt algorithm, with the difference that it can be applied specifically to the directions suffering from low singular values.

Theoretical standard deviation

The evaluation of the function $y(\hat{\beta}_n)$ on the current estimates can be used to estimate the covariance matrix of the estimator. The probability distribution of the estimate $\hat{\beta}_n$ is approximated by a Gaussian:

$$\hat{\beta}_n - \beta^* \sim \mathcal{N}\left(0, \sigma^2 \left[J^T(\hat{\beta}_n) Q_n J^T(\hat{\beta}_n) \right]^{-1}\right) \quad (5.24)$$

where β^* is the true parameter set, and $\sigma^2 = 1$ because the diagonals of the weighting matrix Q_n are chosen as the inverses of the variance of the periodogram estimator. The diagonal of the covariance matrix is used to determine the variance of the estimates.

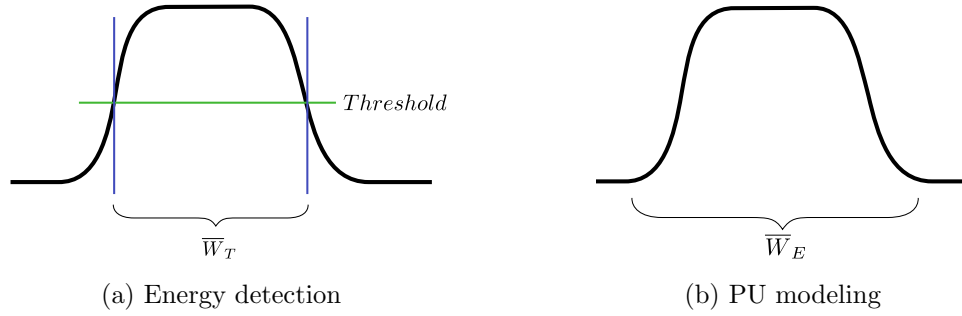


Figure 5.5: Estimates of spectrum utilization

5.1.3 Advantages over Power Sensing Methods

The proposed method for spectrum sensing offers two advantages: first, carrier model fitting in the frequency domain merges the information from all the frequency bins, correcting erroneous decisions that would arise from non-correlated point-to-point comparisons to a threshold value. Second, it desensitizes the primary user bandwidth estimation from variables as signal and noise power thanks to the decoupling of the variables.

The bandwidth occupation estimate $\overline{W}_E$ given by the parameter estimation technique is:

$$\overline{W}_E = (1 + \alpha)f_s \quad (5.25)$$

this result is not sensitive to changes in the signal or noise power, assuming ideal estimation of the true values. A comparison between spectrum utilization estimation by energy detection with thresholding and by primary user modeling is presented in Figure 5.5.

In Figure 5.6 the PU spectral occupation is estimated by thresholding. This method requires an estimate of the A_{up} and A_{dn} values to set a proper threshold value. Assume for the moment that the decision threshold is set at $(A_{up} + A_{dn})/2$. In this case the expected value of the measured occupied bandwidth $\overline{W}_T$ equals the symbol rate f_s . The length of the frequency domain tails is determined by the value of the roll-off factor α that could be used for adjusting the estimate, but is unknown in power sensing methods.

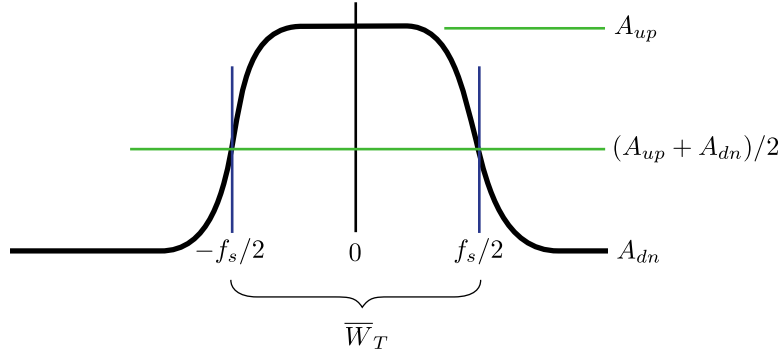


Figure 5.6: Threshold line for PU signal detection

To calculate the sensitivity of $\bar{W}_T$ to the noise power spectral density while abstracting away the local effects of noise, we can calculate its partial derivative with respect to A_{dn} using the chain rule:

$$\frac{\partial \bar{W}_T}{\partial A_{dn}} = \frac{\partial \bar{W}_T}{\partial y(f, \beta)} \frac{\partial y(f, \beta)}{\partial A_{dn}} \quad (5.26)$$

The first derivative is found by realizing that the estimated bandwidth $\bar{W}_T$ corresponds to the length of the projection of the intersections of the threshold line and the raised-cosine function on the frequency axis. Therefore,

$$\frac{\partial \bar{W}_T}{\partial y(f, \beta)} = \left| \frac{\partial y(f, \beta)}{\partial f} \right|_{f=-f_s/2} + \left| \frac{\partial y(f, \beta)}{\partial f} \right|_{f=f_s/2} \quad (5.27)$$

by symmetry, and taking evaluating at the negative frequency intersection to avoid the absolute value operator, we obtain:

$$\frac{\partial \bar{W}_T}{\partial y(f, \beta)} = 2 \frac{\partial y(f, \beta)}{\partial f} \bigg|_{f=-f_s/2} \quad (5.28)$$

therefore,

$$\frac{\partial \bar{W}_T}{\partial y(f, \beta)} = \frac{\alpha f_s}{\pi} \frac{1}{A_{up} - A_{dn}} \quad (5.29)$$

The partial derivative $\partial y(f, \beta)/\partial A_{dn}$ was calculated previously in Equation 5.22. Evaluating at $f = -f_s/2$ we obtain:

$$\left. \frac{\partial y(f, \beta)}{\partial A_{dn}} \right|_{f=-f_s/2} = \frac{1}{2} \quad (5.30)$$

the sensitivity of the bandwidth estimate with respect to the noise floor can be expressed as:

$$\frac{\partial \overline{W}_T}{\partial A_{dn}} = \frac{\alpha f_s}{2\pi} \frac{1}{A_{up} - A_{dn}} \quad (5.31)$$

We obtain an intuitive result: the estimated bandwidth sensitivity is directly proportional to the roll-off factor α and to the symbol rate f_s , and inversely proportional to the difference between the top-of-signal power spectral density and the noise floor. A similar approach can be used to find the sensitivity of the threshold-driven power sensing technique with respect to the PU signal power.

5.1.4 Over-the-air Experiments

Experiments of the method were conducted over-the-air using the Harris SiP FPGA system for signal generation and acquisition, and an RF front end at 2.05 GHz.

The test signals were generated with a symbol rate of 250 kHz and a roll-off factor of 0.75. While digital up conversion is required to achieve the 70 MHz IF signal required by the RF front end, bandpass aliased sampling was implemented with a frequency of 8 MHz, mapping the 70 MHz IF to an aliased band of 2 MHz in the digital circuitry. Down sampling to 1 MHz was performed to enhance the spectral resolution of the PSD estimate. A variable attenuator at the transmitter side was used to change the signal level and hence the signal to noise ratio. Each experiment was repeated 40 times to obtain an estimate of the variance of the method. A summary of the values used for the experiments is shown in Table 5.1.

In the proposed implementation, down conversion to the 70 MHz intermediate frequency

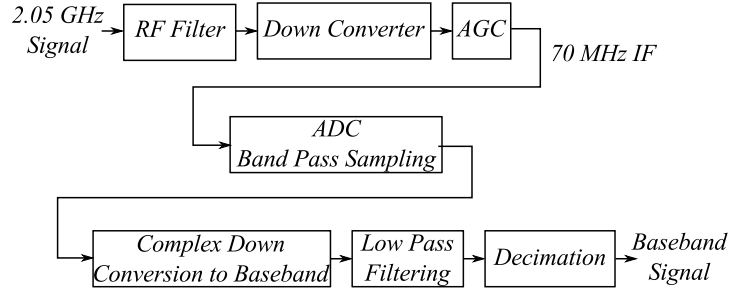


Figure 5.7: Down conversion and down sampling

Table 5.1: Parameters used for simulation and over-the-air experiments

Parameter	Value
F_s	1 MHz
f_c	0 Hz
f_s	250 kHz
α	0.75
N_W	2^{10}
L	32
Number of Repetitions	40

band is performed using an analog RF front end. Band pass sampling is performed to obtain the benefits of down sampling while relaxing the requirements on the analog to digital converter. Filtering of the signal of interest is performed digitally before a second down sampling stage. Figure 5.7 illustrates the process. The conditions of the test and simulations are presented in Table 5.1.

Standard deviation

The standard deviation figures present an asymptotic behavior in performance, converging to a non-zero floor. This floor is due to the lack of consistency of the DFT that causes a non-zero variance of the PSD estimate even under noise free conditions.

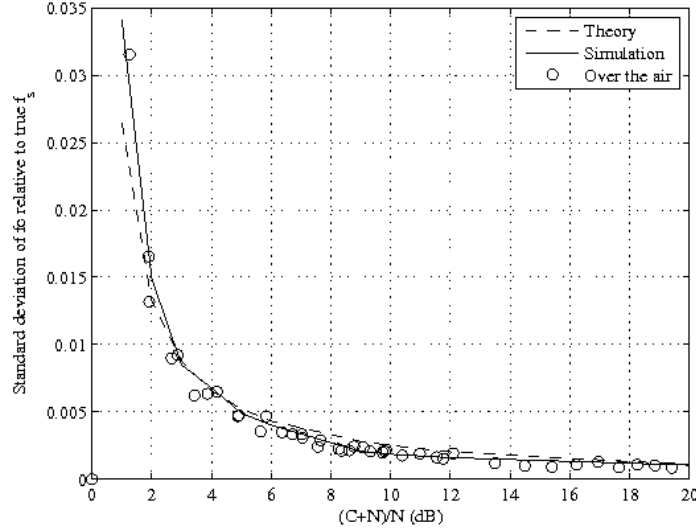
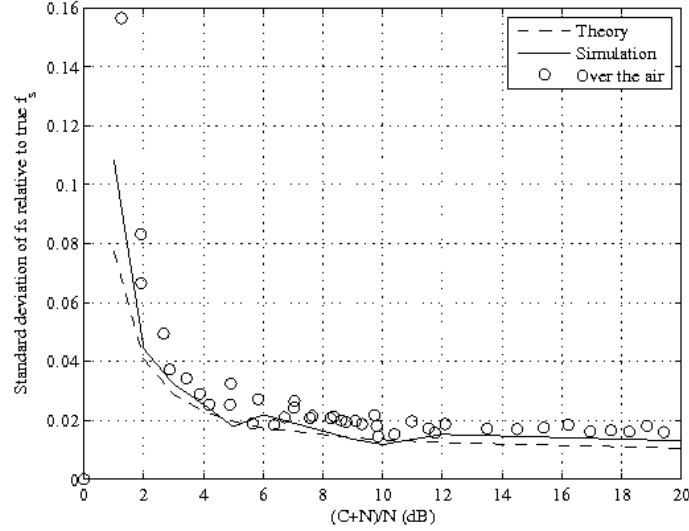


Figure 5.8: Normalized standard deviation of $\hat{f}_c$ relative to f_s

The standard deviation of the estimate of the carrier frequency $\hat{f}_c$ is normalized to the symbol frequency because in baseband the carrier frequency is expected to be zero, and the effect of carrier frequency error in terms of phase shift per symbol depends on its relation to the true symbol rate f_s . This result is presented in Figure 5.8. The theory, simulation, and experimental results match closely, indicating that the theoretical performance is a good approximation of the system.

Figure 5.9 and 5.10 show the standard deviation of the symbol rate estimate $\hat{f}_s$ and roll-off factor α normalized to their respective true values. Their standard deviation is higher than what was obtained for the carrier frequency offset estimate. The most likely reason for the lower performance is the high coupling between f_s and the roll-off factor α , demonstrated by the element $\sigma_{2,3}$ of the normalized covariance matrix defined in Equation 5.32. The evaluation of $\tilde{\Sigma}$ for the set of true parameters is presented in Equation 5.33.

$$\tilde{\Sigma} = \left[(J(\beta^*)D^*)^T Q (J(\beta^*)D^*) \right]^{-1} \quad (5.32)$$

Figure 5.9: Normalized standard deviation of $\hat{f}_s$

$$\tilde{\Sigma} = \begin{bmatrix} 1.00 & 0.00 & 0.00 & 0.00 & 0.00 \\ 0.00 & 6.11 & -4.33 & -3.20 & -0.44 \\ 0.00 & -4.33 & 4.72 & 2.55 & -0.74 \\ 0.00 & -3.20 & 2.55 & 2.77 & -0.16 \\ 0.00 & -0.44 & 0.74 & -0.16 & 1.76 \end{bmatrix} \quad (5.33)$$

5.2 Automatic Modulation Classification

Automatic Modulation Classification (AMC) can be used to demodulate an otherwise unknown signal and perform active cancellation of PU interference. This approach is resource-consuming, and requires a system for the generation of a radio receiver for each PU. Such a system is the topic of the dissertation of Suris [54]. This Section presents the AMC system developed for the Rapid Radio project, and its application for interference cancellation in a secondary user scenario.

While there are many proposed AMC techniques, few consider the problem of lack of syn-

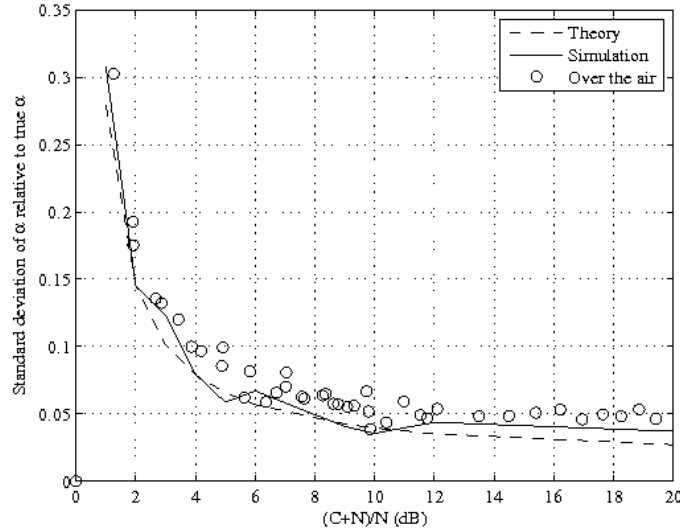


Figure 5.10: Normalized standard deviation of roll-off estimate $\hat{\alpha}$

chronization at the symbol and the carrier levels, as well as the capability to classify an arbitrary set of modulation types. The focus of the presented investigation is on a AMC system for the class of linear modulations, as well as its interactions with the synchronization stages and with the general flow of a prototyping system.

This section is organized as follows: Section 5.2.1 presents an overview of AMC techniques. The approach chosen for the modulation classification stage is explained in Section 5.2.2. The parameter estimation and signal conditioning stage is presented in Section 5.2.3. The classification features are explained in Section 5.2.5, while the Bayesian network classifier is discussed in Section 5.2.6. The results are presented in Section 5.2.7.

5.2.1 Previous Work on Automatic Modulation Classification

The literature about Automatic Modulation Classification (AMC) covers several techniques directed toward classifying a signal using diverse features or performing likelihood tests. Many of these techniques rely on assumptions about the stage of processing of the signal (i.e.:

a baseband representation is available, or sampling is synchronous to the symbol epochs), or about the modulation being restricted to a limited set. Dobre et al. compiled a comprehensive summary of the literature in AMC [98]. The main two families of algorithms are likelihood-based (LB) and feature-based (FB).

Likelihood-based algorithms perform hypotheses testing directly over a baseband representation of the signal. Simplifications of the exact likelihood function lead to the generalized likelihood ratio test (GLRT), the average likelihood ratio test (ALRT), and the hybrid likelihood ratio test (HLRT). Panagiotou presents detailed explanations of each of these methods [99]. Polydoros and Kim propose the quasi-log-likelihood ratio qLLR to approximate the likelihood function of BPSK/QPSK signals, assuming that the carrier frequency and symbol timing are known [100]. Tadaion presents a suboptimal implementation of a HLRT that provides independence of the noise parameters for the M-PSK class of modulations [101]. Wen and Mendel found asymptotic results on the performance of maximum likelihood classifiers in the I-Q domain for linear modulations [102]. A drawback of this family of methods is their capability to differentiate only among a particular set of constellations, and they are susceptible to nesting when a set of constellations generate the same value of the test statistic.

Most of the mentioned authors use the quasi-baseband signal obtained after down conversion, a strategy known as the signal space approach. Using the signal space has the advantage that it is readily available as an intermediate signal in a receiver, but assumes a previous symbol timing recovery.

Analyses of the constellation in the signal space using pattern recognition techniques such as clustering are proposed by Mobasseri and Zhijin et al. [103, 104]. Clustering introduces undesired additional degrees of freedom to the problem because linear digital modulation constellations are highly regular and have predetermined shapes. Modulation classification techniques based on fuzzy logic [105] and artificial neural networks [106, 87, 107] are also proposed in the literature.

Feature-based algorithms include the wide family of cyclostationarity based classifiers [108], as those based in spectral coherence and spectral correlation [79, 109, 110, 106]; cyclic cumulants [111, 112, 113]; and higher order statistics [114]. Other features explored in the literature are: moment matrices [115], the standard deviation of the normalized-centered instantaneous amplitude and kurtosis [116], and the Hellinger distance [117].

“Holistic” approaches that acknowledge the requirement to gather the basic parameters of the signal and consider the interactions between modulation recognition and synchronization are presented in [118, 119, 120, 121, 122].

5.2.2 Proposed Approach

The technique implemented for AMC is based in the analysis of four features: amplitude histogram, differential phase histogram, I-Q plane histogram, and symbol transition matrix. Symbol synchronization is a condition to obtain the amplitude and differential phase profiles. Symbol and carrier synchronization are required to obtain the I-Q plane histogram and the symbol transition matrix. The generation of a baseline probability density function (PDF) for histogram matching with each hypothesized constellation type requires accurate constellation descriptions. SNR estimates are also needed for the construction of the baseline PDFs. A Bayesian network is used for the integration of the metrics and for the calculation of total scores used to reach a classification result. A general block diagram of the automatic signal classification system is presented in Figure 5.11.

5.2.3 Parameter Estimation and Signal Conditioning

The first step for the classification of a signal is the identification of the frequency band where the carrier of interest resides, the implementation of a downconverter, and the coarse estimation of the parameters of the signal, including carrier frequency offset (CFO), symbol rate, roll-off factor, and signal to noise ratio, as described in Section 5.1. The complex

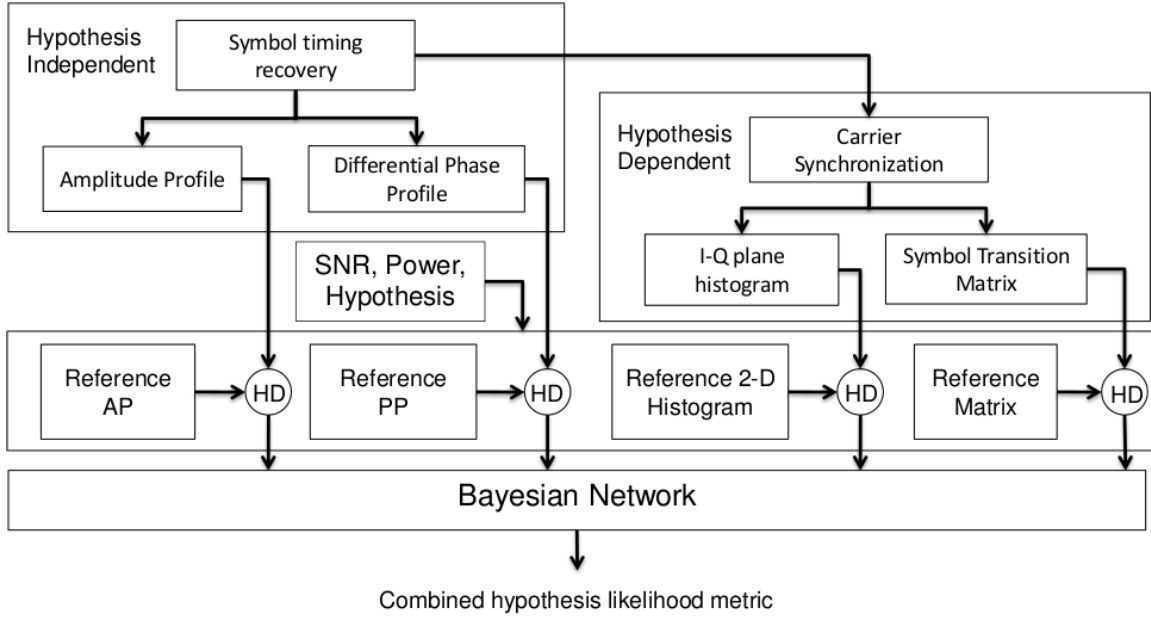


Figure 5.11: AMC system block diagram

baseband signal $x(n)$ is conditioned according to the system presented Figure 5.12 by using the estimates of the symbol rate and the roll-off factor to produce a matched filter followed by an arbitrary frequency re-sampler to obtain a filtered signal $y(m)$ with a nominal sampling frequency of four samples per symbol.

5.2.4 Synchronization and Amplitude Normalization

Symbol timing recovery is an adaptation of the Godard synchronizer [123] at baseband sampled at four samples per symbol. It uses a non-linearity to generate a spectral line at the symbol rate, aided by a half-symbol delay in one of its branches, as proposed by Simon [124]. This arrangement is followed by a PLL to filter out the phase noise. Figure 5.13 presents a detailed block diagram of the synchronizer.

This symbol synchronization approach is constellation agnostic, and can be performed without previous assumptions about the constellation shape.

The value of the signal $s(k)$ at the optimum sampling instant is found using a Lagrange

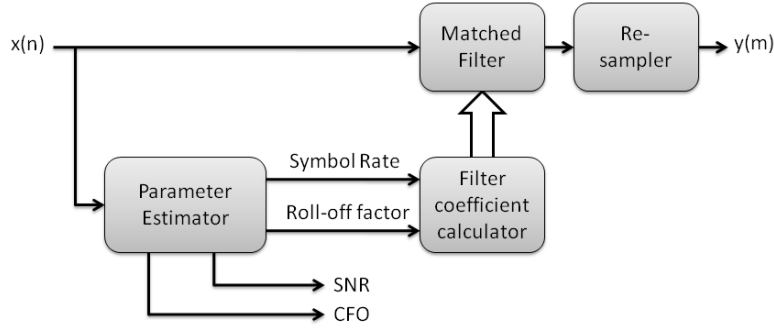


Figure 5.12: Signal parameter estimation and pre-conditioning

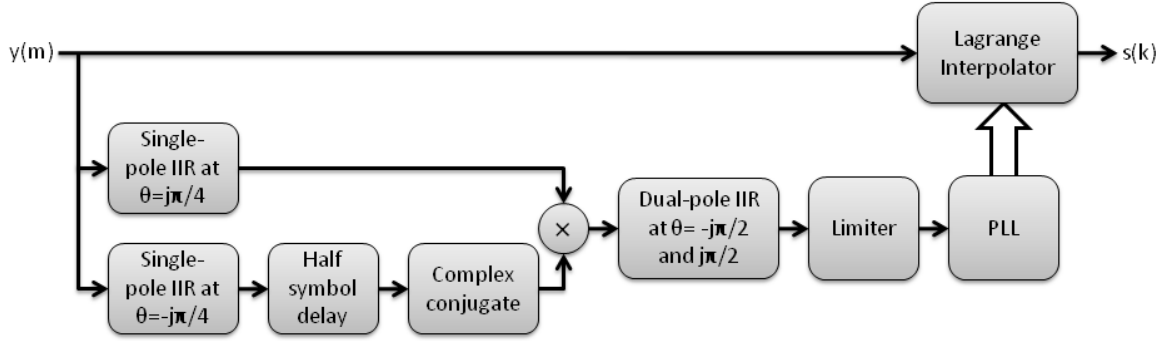


Figure 5.13: Symbol synchronizer architecture

polynomial interpolator controlled by the PLL time base.

Carrier frequency offset synchronization ambiguity

Two of the modulation classification features presented below, the differential phase profile and the amplitude profile, can be applied to the signal without carrier frequency offset (CFO) synchronization; therefore a low-tier classification can be performed using the symbol synchronizer and these features. A high-tier classification system requires the implementation of a CFO synchronizer, allowing the generation of two additional features: the two-dimensional I-Q profile and the symbol differential matrix.

The CFO synchronizer must be fed the parameters of the hypothesized constellation to obtain hard decisions used to measure the phase error, and therefore is not constellation agnostic.

This approach produces constellation nesting. For instance, an 8-PSK CFO synchronizer applied to a QPSK signal may produce what looks like a 8-PSK constellation, by introducing an additional $\pi/4$ phase rotation between consecutive symbols. Similarly, a $\pi/4$ -DQPSK modulated signal fed to a 8-PSK hypothesis will produce a valid I-Q profile. The differential matrix comes handy at resolving the rotation ambiguities introduced by a CFO synchronizer running under different hypotheses.

5.2.5 Classification Features

Amplitude profile test

If the channel noise is assumed to be AWGN, the theoretical amplitude probability density function at the proper sampling instants can be described as a mixture of Ricean-distributed random variables calculated according to (5.34). An example of the amplitude profile function is presented in Figure 5.14.

$$f_{ap|c}(x) = \sum_{i=0}^{N-1} p_i f(x, \nu_i) \quad (5.34)$$

where the value ν_i represents the magnitude of the i_{th} element of the constellation under test, N is the number of elements in the constellation, and p_i is the probability of a constellation element. All of these values are obtained from the XML constellation description assuming equiprobable symbols. The function $f(x, \nu)$ is the PDF of the Ricean distribution given by (5.35).

$$f(x, \nu) = \frac{x}{\sigma^2} \exp\left(-\frac{x^2 + \nu^2}{2\sigma^2}\right) I_0\left(\frac{x\nu}{\sigma^2}\right) \quad (5.35)$$

where $I_0(x)$ is a Bessel function of the first kind.

To obtain the proper constellation points, the amplitude of the signal is normalized according to (5.36), in order to achieve unit power. The parameter σ^2 can then be calculated as

according to (5.37).

$$\tilde{s}(k) = \frac{s(k)}{\sqrt{P^{\frac{\gamma-1}{\gamma}}}} \quad (5.36)$$

where $P = E[s^2(k)]$ is the mean square value of the received symbol sequence s_i assuming $E[s(k)] = 0$, and γ is the measured carrier plus noise to noise ratio obtained at the parameter estimator.

$$\sigma^2 = \frac{1}{2} \frac{1}{\gamma - 1} \quad (5.37)$$

A histogram of the received signal amplitudes is obtained with L bins of size δ , chosen according to Scott's rule [125]. This histogram is scaled to form the empirical PDF of the received data set. A similarity metric between the hypothesized PDF and the empirical PDF is obtained using the Hellinger distance (5.38), presented by Huo and Donoho [117]. The factor $\delta/2$ is added to produce results in the interval $[0, 1]$.

$$d_H(f_1, f_2) = \frac{\delta}{2} \sum_{i=1}^L \left(\sqrt{f_1(x_i)} - \sqrt{f_2(x_i)} \right)^2 \quad (5.38)$$

Differential phase profile test

The amplitude profile test is useful to classify amplitude-modulated signals. However, it does not permit a classification among phase-modulated signals, such as BPSK, QPSK, 8-PSK, and $\pi/4$ -DQPSK. The differential profile is obtained by forming the histogram of the phase difference between consecutive symbols. Contrary to the amplitude profile, it is not independent of the CFO: the obtained profile will be phase-shifted by an amount of $2\pi\Delta_f T_{\text{symp}}$, where Δ_f is the CFO, and T_{symp} is a symbol duration. The hypothesized phase profile is obtained as a mixture distribution from the XML constellation description and the

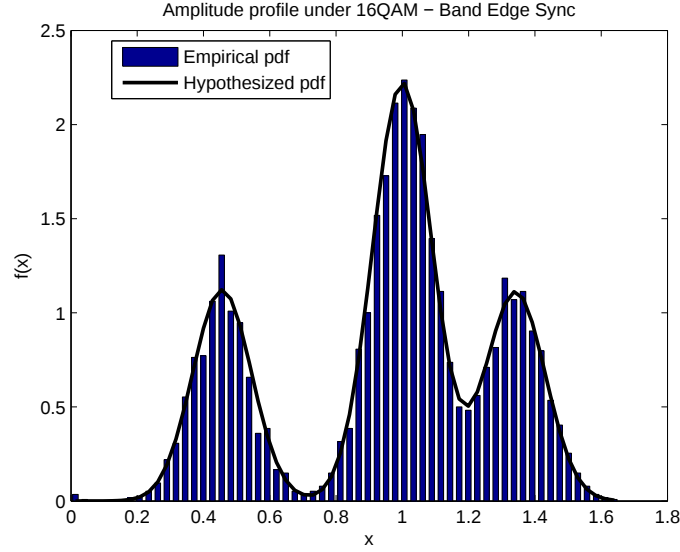


Figure 5.14: Amplitude profiles of a 16-QAM constellation with 18 dB SNR

estimated SNR value using the expression for the PDF of the angle between two vectors contaminated by AWGN [126], presented in (5.39).

$$f(\psi) = \frac{1}{2\pi} \int_0^{\pi/2} \exp(-\rho(1 - \cos \psi \cos \theta)) (1 + \rho + \rho \cos \psi \cos \theta) \cos \theta \, d\theta \quad (5.39)$$

where $\rho = 1/\sigma^2$, and $\psi = \theta(n) - \theta(n-1)$ represents the phase difference between consecutive symbols.

The integral is solved numerically for each angle under evaluation. In order to overcome the offset introduced by the CFO, an exhaustive search is performed over all the phase shifts allowed by the discrete histogram bins, looking for the smallest Hellinger distance. The minimum distance is used as the metric for this classification feature. An example differential phase profile is presented in Figure 5.15. The top part of the figure presents the results of the minimum search, while the bottom part presents the matched PDFs.

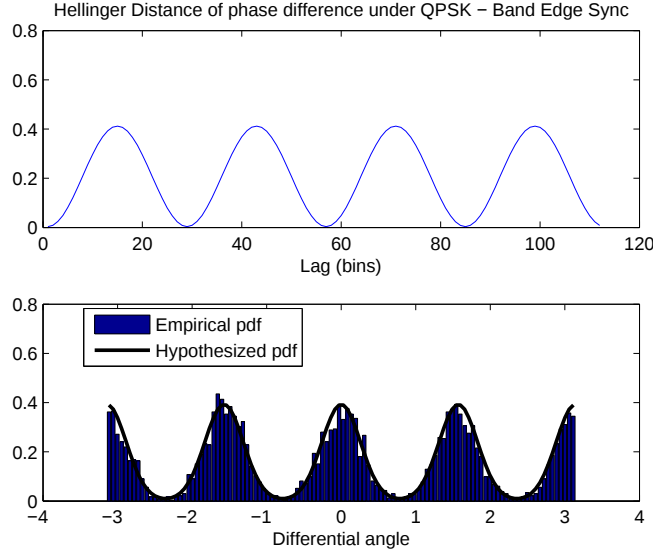


Figure 5.15: Differential phase profile under QPSK with 12 dB SNR

Two-dimensional PDF test

The two-dimensional PDF is one of the features that can be obtained after the CFO tracking loop. As the CFO tracking system requires knowledge of the hypothesized constellation, this measurement is sometimes meaningless when used under incorrect hypotheses, being usually out-of-lock. However, under the correct hypothesis, the tracking loop will be locked, producing a small distance metric when the empirical and the hypothesized PDFs are compared.

The hypothesized PDFs are built as two-dimensional Gaussian mixtures, with centers given by the constellation points, and variance σ^2 obtained from the SNR estimate. An example for the case of QPSK modulation is presented in Figure 5.16.

Transition matrix test

The transition matrix test is added to solve constellation nesting problems presented in 8-PSK vs. $\pi/4$ -DQPSK or QPSK vs. $\pi/4$ -DQPSK. When a signal is demodulated under certain hypothesis, a transition matrix is calculated. This feature provides insight into the

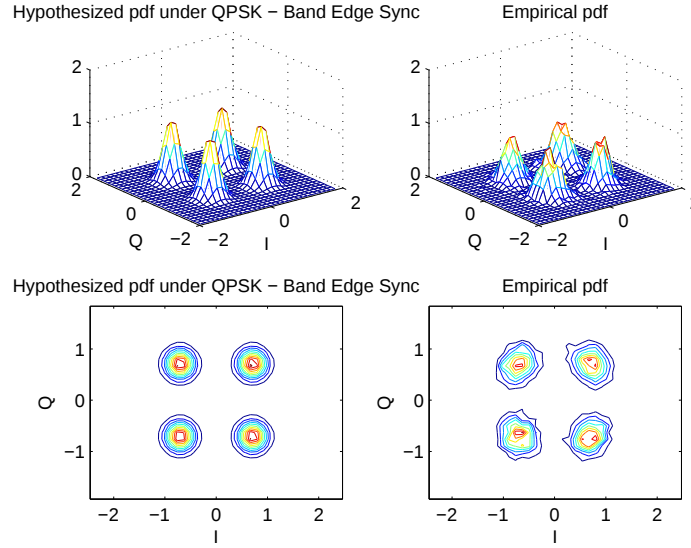


Figure 5.16: Two-dimensional PDF under QPSK with 12 dB SNR

differential structure of the modulation scheme, allowing, for example, the classification of a $\pi/4$ -DQPSK signal, that would obtain the same metric as 8-PSK under the two-dimensional PDF test, and the same metric as QPSK under the differential phase test. An example transition matrix for $\pi/4$ -DQPSK is presented in Equation 5.40.

$$\begin{bmatrix}
 0 & 1/4 & 0 & 1/4 & 0 & 1/4 & 0 & 1/4 \\
 1/4 & 0 & 1/4 & 0 & 1/4 & 0 & 1/4 & 0 \\
 0 & 1/4 & 0 & 1/4 & 0 & 1/4 & 0 & 1/4 \\
 1/4 & 0 & 1/4 & 0 & 1/4 & 0 & 1/4 & 0 \\
 0 & 1/4 & 0 & 1/4 & 0 & 1/4 & 0 & 1/4 \\
 1/4 & 0 & 1/4 & 0 & 1/4 & 0 & 1/4 & 0 \\
 0 & 1/4 & 0 & 1/4 & 0 & 1/4 & 0 & 1/4 \\
 1/4 & 0 & 1/4 & 0 & 1/4 & 0 & 1/4 & 0
 \end{bmatrix} \quad (5.40)$$

5.2.6 Bayesian Network Classifier

To support the “plug-in” capabilities of the signal classifier, a Bayesian network was chosen to combine the metrics of the classifier. The Bayesian approach offers several advantages, among them:

1. Different sets of prior probabilities can be chosen according to the environment and previous experience.
2. There is no need to revise the decision tree if a new constellation type is added to the set under consideration.
3. The classifications are soft: instead of a yes/no answer, the posterior probabilities can be sorted and used as a likelihood measurement.

The conditional probabilities of an empirical PDF given the hypothesis h_i are assigned according to (5.41).

$$\begin{aligned}
 P(ap|h_i) &= 1 - d_H(f_{ap}, f_{ap|c_i}) && \text{Amplitude profile} \\
 P(pp|h_i) &= 1 - d_H(f_{pp}, f_{pp|c_i}) && \text{Differential phase profile} \\
 P(sd|h_i) &= 1 - d_H(f_{sd}, f_{sd|c_i}) && \text{2-D distribution} \\
 P(tm|h_i) &= 1 - d_H(f_{tm}, f_{tm|c_i}) && \text{Transition matrix}
 \end{aligned} \tag{5.41}$$

where the set of functions f_x represent the empirical distribution for the feature x , and the functions $f_{x|c_i}$ represent the theoretical PDF of each feature given the hypothesized constellation c_i .

The posterior probabilities are calculated according to Bayes rule, as presented in (5.42).

$$P(h_i|ap, pp, sd, tm) = \frac{P(ap, pp, sd, tm|h_i)P(h_i)}{\sum_{i=0}^{N-1} [P(ap, pp, sd, tm|h_i)P(h_i)]} \tag{5.42}$$

where the set of prior probabilities $P(h_i)$ can be chosen according to an experience-based criteria. For simplicity, Conditional independence is assumed for the set of conditional probabilities. Therefore:

$$P(ap, pp, sd, tm|h_i) = P(ap|h_i)P(pp|h_i)P(sd|h_i)P(tm|h_i) \quad (5.43)$$

5.2.7 Results

The results of the AMC system implemented are affected doubly when the signal to noise ratio is low. A low SNR affects the synchronization stages, specially for highly dense constellations, as 64-QAM. Therefore, the rate of correct classification is much lower to the one obtained under ideal synchronization conditions.

Tables 5.2 to 5.10 present the results of the individual PDF distance metrics. Equal priors have been used to initialize the Bayesian network. Its output is shown in the “Posteriors” column. An SNR of 18 dB and 4096 symbols are used to obtain this result. An important aspect that requires further development is the fact that the transition matrix is only helpful in the case of a QPSK constellation being masked as $\pi/4$ -DQPSK. In other cases, as in 64-QAM, presented in Table 5.10, it even produces a classification error that could have been otherwise avoided by using only the remaining three features that are clear winners against the set of false hypotheses.

Table 5.2: Set of metrics for a BPSK signal

Hypothesis	$P(ap h_i)$	$P(pp h_i)$	$P(sd h_i)$	$P(tm h_i)$	Posteriors
BPSK	99.40%	99.59%	97.68%	100.00%	66.06%
QPSK	99.40%	69.26%	68.97%	50.76%	16.47%
8PSK	99.40%	48.71%	49.27%	28.71%	4.68%
16QAM	80.78%	46.72%	62.35%	28.57%	4.59%
64QAM	75.84%	44.09%	61.02%	27.49%	3.83%
32QAM	70.43%	43.62%	49.27%	30.09%	3.11%
PI4DQPSK	99.40%	69.15%	49.27%	2.72%	0.63%
8QAM	42.02%	48.71%	9.00%	28.46%	0.36%
STAR16	29.72%	48.71%	11.18%	24.92%	0.28%

Table 5.3: Set of metrics for a QPSK signal

Hypothesis	$P(ap h_i)$	$P(pp h_i)$	$P(sd h_i)$	$P(tm h_i)$	Posteriors
QPSK	99.62%	99.18%	96.95%	99.89%	48.24%
BPSK	99.62%	70.09%	45.20%	99.99%	15.91%
8PSK	99.62%	70.92%	68.94%	55.70%	13.68%
16QAM	78.71%	68.06%	68.43%	41.22%	7.62%
64QAM	74.07%	64.24%	62.20%	37.29%	5.56%
32QAM	68.72%	63.60%	54.19%	45.06%	5.38%
PI4DQPSK	99.62%	99.06%	68.94%	4.73%	1.62%
8QAM	39.92%	70.92%	14.48%	66.84%	1.38%
STAR16	26.28%	70.92%	13.42%	48.62%	0.61%

Table 5.4: Set of metrics for an 8-PSK signal

Hypothesis	$P(ap h_i)$	$P(pp h_i)$	$P(sd h_i)$	$P(tm h_i)$	Posteriors
8PSK	99.60%	98.75%	95.71%	99.79%	33.22%
PI4DQPSK	99.60%	69.87%	95.71%	71.15%	16.76%
QPSK	99.60%	69.89%	54.53%	99.93%	13.42%
16QAM	82.61%	85.68%	64.11%	62.11%	9.97%
32QAM	71.45%	86.64%	53.52%	68.06%	7.98%
64QAM	76.41%	86.66%	62.75%	48.93%	7.19%
BPSK	99.60%	49.10%	34.43%	99.98%	5.95%
8QAM	43.99%	98.75%	23.82%	99.84%	3.65%
STAR16	33.32%	98.75%	17.73%	89.85%	1.85%

Table 5.5: Set of metrics for a $\pi/4$ -DQPSK signal

Hypothesis	$P(ap h_i)$	$P(pp h_i)$	$P(sd h_i)$	$P(tm h_i)$	Posteriors
PI4DQPSK	99.45%	99.05%	95.10%	99.40%	42.68%
8PSK	99.45%	71.66%	95.10%	75.51%	23.46%
16QAM	84.39%	70.44%	62.15%	52.18%	8.84%
32QAM	73.17%	66.37%	53.47%	55.94%	6.66%
BPSK	99.45%	70.10%	19.65%	100.00%	6.28%
64QAM	77.88%	67.04%	63.23%	38.98%	5.90%
8QAM	45.69%	71.66%	24.04%	77.84%	2.81%
QPSK	99.45%	98.91%	4.28%	99.97%	1.93%
STAR16	35.92%	71.66%	18.56%	66.44%	1.45%

Table 5.6: Set of metrics for an 8-QAM signal

Hypothesis	$P(ap h_i)$	$P(pp h_i)$	$P(sd h_i)$	$P(tm h_i)$	Posteriors
8QAM	99.35%	98.11%	94.25%	99.83%	35.34%
STAR16	89.97%	98.11%	82.42%	77.25%	21.66%
32QAM	88.66%	92.86%	73.32%	67.19%	15.63%
64QAM	84.96%	92.83%	70.71%	55.10%	11.84%
16QAM	70.23%	91.63%	35.94%	84.86%	7.56%
8PSK	35.44%	98.11%	28.65%	99.85%	3.83%
PI4DQPSK	35.44%	69.26%	28.65%	70.63%	1.91%
QPSK	35.44%	69.22%	16.44%	99.96%	1.55%
BPSK	35.44%	48.63%	9.81%	99.98%	0.65%

Table 5.7: Set of metrics for a 16-QAM signal

Hypothesis	$P(ap h_i)$	$P(pp h_i)$	$P(sd h_i)$	$P(tm h_i)$	Posteriors
16QAM	99.64%	98.85%	91.40%	99.27%	27.45%
32QAM	88.73%	98.72%	70.41%	92.97%	17.61%
64QAM	92.26%	98.64%	77.38%	69.73%	15.08%
STAR16	75.57%	86.16%	52.56%	97.45%	10.24%
8PSK	73.09%	86.16%	53.01%	99.63%	10.21%
8QAM	70.19%	86.16%	34.60%	99.11%	6.37%
PI4DQPSK	73.09%	67.12%	53.01%	67.18%	5.36%
QPSK	73.09%	67.13%	32.63%	99.91%	4.91%
BPSK	73.09%	46.85%	26.20%	100.00%	2.76%

Table 5.8: Set of metrics for a Star-16 signal

Hypothesis	$P(ap h_i)$	$P(pp h_i)$	$P(sd h_i)$	$P(tm h_i)$	Posteriors
STAR16	99.56%	96.64%	91.77%	99.28%	37.24%
8QAM	90.59%	96.64%	54.41%	96.10%	19.44%
32QAM	79.15%	95.05%	62.42%	84.08%	16.77%
16QAM	74.47%	93.60%	43.87%	93.75%	12.18%
64QAM	78.92%	95.07%	63.49%	58.66%	11.87%
8PSK	20.56%	96.64%	14.15%	99.87%	1.19%
PI4DQPSK	20.56%	68.86%	14.15%	70.20%	0.60%
QPSK	20.56%	68.90%	8.29%	99.97%	0.50%
BPSK	20.56%	47.90%	4.92%	100.00%	0.21%

Table 5.9: Set of metrics for a 32-QAM signal

Hypothesis	$P(ap h_i)$	$P(pp h_i)$	$P(sd h_i)$	$P(tm h_i)$	Posteriors
32QAM	99.74%	99.00%	88.08%	96.53%	26.07%
64QAM	97.43%	99.01%	84.55%	74.84%	18.96%
16QAM	88.14%	97.24%	59.87%	97.99%	15.62%
STAR16	77.51%	86.76%	55.80%	99.00%	11.54%
8QAM	86.47%	86.76%	45.61%	98.33%	10.45%
8PSK	63.89%	86.76%	44.62%	99.83%	7.67%
PI4DQPSK	63.89%	62.48%	44.62%	71.01%	3.93%
QPSK	63.89%	62.48%	31.45%	99.97%	3.90%
BPSK	63.89%	43.80%	21.49%	100.00%	1.87%

Table 5.10: Set of metrics for a 64-QAM signal

Hypothesis	$P(ap h_i)$	$P(pp h_i)$	$P(sd h_i)$	$P(tm h_i)$	Posteriors
32QAM	96.22%	99.25%	77.83%	96.16%	22.36%
64QAM	99.79%	99.27%	87.63%	76.73%	20.84%
16QAM	90.92%	97.59%	60.12%	97.87%	16.33%
STAR16	75.65%	87.39%	53.87%	99.11%	11.04%
8QAM	81.24%	87.39%	42.46%	98.66%	9.30%
8PSK	67.40%	87.39%	47.69%	99.79%	8.77%
QPSK	67.40%	63.69%	34.33%	99.97%	4.61%
PI4DQPSK	67.40%	63.69%	47.69%	69.27%	4.44%
BPSK	67.40%	44.50%	24.65%	99.97%	2.31%

5.3 Conclusions

This chapter demonstrated the feasibility of spectrum sensing based on modeling of the spectral shape of primary user signals. Modeling the spectral shape of the PU signal is proposed as a robust alternative to thresholding in power sensing strategies. The model is obtained by hypothesizing a modulation type and estimating fundamental modulation parameters. The adopted strategy relies on the power spectral estimation obtained using the unmodified periodogram and the use of a trust-region optimization method. The mean and standard deviation obtained by simulation experiments and over-the-air closely match the theory.

The PU model permits the determination of clear boundaries on the spectrum occupation regardless of variations of the signal strength (as opposed to power threshold algorithms) or the noise level. It also can be used to provide a total signal power estimate that can be used for triangulation in environments with multiple network elements.

Potential enhancements to the proposed method include the tracking of a variable number of carriers and the analysis of intermittent signal, commonly found in mobile communications. The parameters of the model can be used as start-up values for an automatic modulation classification algorithm.

The realization of a practical AMC system must acknowledge the need for synchronization. Four features were tested and compared, showing their individual susceptibility to nesting effects that can be compensated by appropriate merging of the individual feature metrics obtained using the Hellinger distance as a hypothesis likelihood metric. A Bayesian network is proposed for the calculation of a set of posterior probabilities.

Chapter 6

FMT with Mutual Interference Rejection

This chapter describes an algorithm for the mitigation of mutual interference in a spectrum overlay system based on a particular filter bank multi-carrier method, FMT. While the potential of FBMC for interference rejection is well known, few previous works deal with the specific problems of frame detection, timing synchronization, and channel estimation in the presence of a wide-band interferer.

The following sections present a detailed explanation of an algorithm that leverages the frequency response of the input filter bank. Section 6.1 presents a review of the legacy frame detection and timing offset estimation algorithms, currently in use in standards like IEEE 802.11 and IEEE 802.16. The filtered multitone modulation scheme is described in Section 6.2, while Section 6.3 shows the properties of CAZAC sequences and the subset of this family possessing the CFO immunity property desired for the proposed algorithm.

The proposed algorithm for frame detection and timing offset estimation is presented in Section 6.4. Channel estimation is obtained as a by-product of the main algorithm. Results are presented in Section 6.5. Finally, the conclusions of the chapter are presented in Section 6.6.

6.1 Legacy Frame Detection and Timing Offset Estimation Algorithms

This section presents an overview of the most common techniques for frame detection and timing offset estimation in legacy OFDM systems. The problem of synchronization of an OFDM signal can be divided into several tasks.

1. Frame detection and coarse timing estimation
2. Carrier frequency offset (CFO) estimation
3. Fine timing acquisition
4. Timing and frequency tracking

These particular sequence of tasks is supported by existing preamble structures, as the one defined in standard IEEE 802.11-2007 [4], shown in Figure 6.1. Even if this strategy represents a cost in terms of bandwidth efficiency, preambles save a great amount of computation and permit a quick acquisition; as a result their use is widespread in recent wireless standards such as WiFi [4], WiMax [127], and LTE [128].

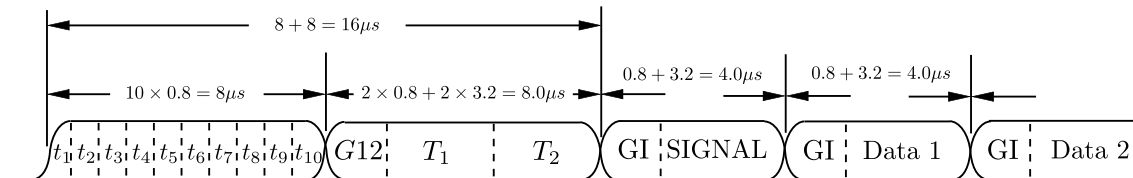


Figure 6.1: IEEE 802.11a/g Frame.

The primitive parameters of the IEEE 802.11a/g waveform are used in the following discussion. They are summarized in Table 6.1.

Table 6.1: Specifications of the IEEE 802.11a/g waveform

Specification	Value
FFT Size (N)	64
Sampling rate	20 MHz
Symbol duration	$4 \mu s$
Cyclic prefix duration	$0.8 \mu s$
Modulation types	BPSK, QPSK, 16-QAM, 64-QAM

6.1.1 Frame Detection and Coarse Timing Acquisition

In the frame structure presented in Figure 6.1, the short preamble is composed of ten identical short symbols, each formed by 16 samples. The repetitive structure can be detected by an autocorrelation operation. A common time-lag for the autocorrelation operation is one short symbol. This operation is expressed in Equation 6.1.

$$R_{rr}(k) = \sum_{i=n-N+1}^n r(i)r(i-k+1)^* \quad \text{with } k = N/4 \quad (6.1)$$

where $r(n)$ represents the receiver's complex baseband signal.

Frame detection occurs by comparison of the autocorrelation magnitude with the estimate of the power, according to Equation 6.2.

$$\|R_{rr}(n)\| > \gamma P(n) + \epsilon \quad (6.2)$$

The parameter γ is used to compensate the increase on signal power due to noise. The parameter ϵ is used to avoid false alarms. The power estimate is found according to:

$$P(n) = \sum_{i=n-N+1}^n r(i)r(i)^* \quad (6.3)$$

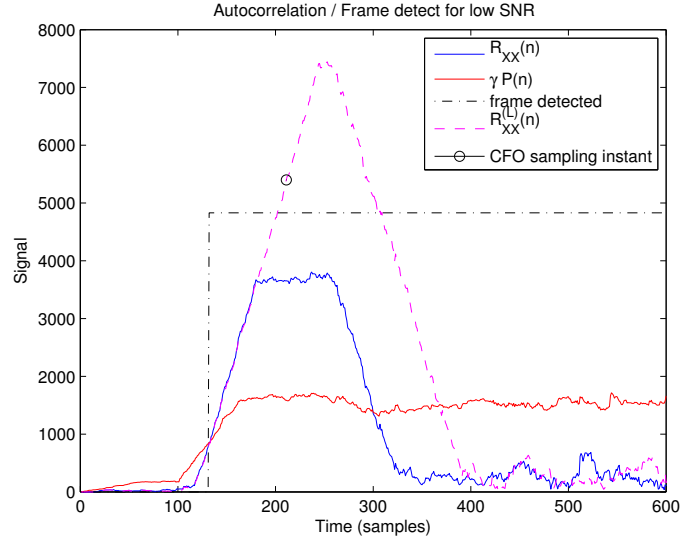


Figure 6.2: Set of signals for frame detection and CFO estimation

Figure 6.2 presents the signals involved in the detection of the frame generated by simulation in MATLAB. Other algorithms use the characteristic “plateau” to obtain frame detection. The instant of frame detection is used as the coarse timing estimate. The location of this estimate relative to the frame is dependent on the signal-to-noise ratio; consequently, a fine timing estimate is later obtained using the properties of the long preamble. Observe that the autocorrelation signals are calculated sample-by-sample in the time domain.

6.1.2 Carrier Frequency Offset Acquisition

Carrier frequency offset (CFO) can be obtained as a by-product of the autocorrelation function $R_{rr}(k)$. The phase of the autocorrelation function corresponds to the angle accumulated during a time equal to the autocorrelation lag. A longer integration time can be used to obtain the autocorrelation function $R_{rr}^{(L)}(k)$ defined in Equation 6.4 to enhance the CFO estimation error.

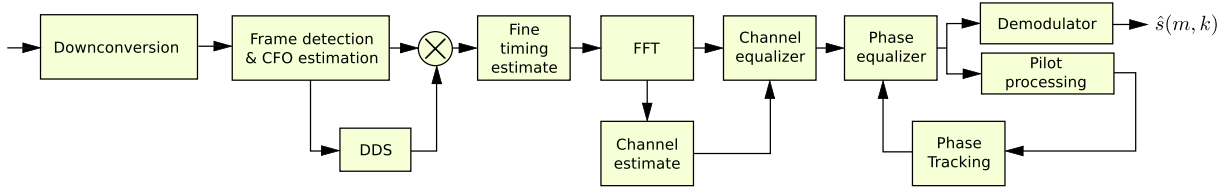


Figure 6.3: OFDM receiver architecture

$$R_{rr}^{(L)}(k) = \sum_{i=n-2N+1}^n r(i)r(i-k+1)^* \quad \text{with } k = N/4 \quad (6.4)$$

6.1.3 Fine Timing Acquisition

The autocorrelation function of the received signal does not provide an exact timing reference. Fine timing estimation is based on the calculation of the cross-correlation function of a long preamble symbol and a locally stored copy of it. This cross-correlation function is meant to provide a peak when the input signal is aligned to the local reference; however, the phase rotation caused by the CFO would null the integration result. To avoid phase rotations during the dwell time of the cross-correlation integration, the CFO must be previously corrected. The block diagram of Figure 6.3 presents a OFDM receiver with time-domain acquisition that illustrates this concept.

6.1.4 Channel estimation

The long preamble is composed of two OFDM symbols plus a cyclic prefix with 50% of a symbol length. As mentioned before, the first long preamble symbol is used to perform the fine timing acquisition. The second preamble is destined for channel estimation. In IEEE 802.11, the long preambles are formed by a frequency domain pseudo-random binary sequence (mapped to +1 and -1). Therefore, when the DFT is applied to the second preamble, a simple sign correction is enough to find the channel estimate. In the OFDM receiver of Figure 6.3, the channel estimate is stored in memory for the duration of the current frame,

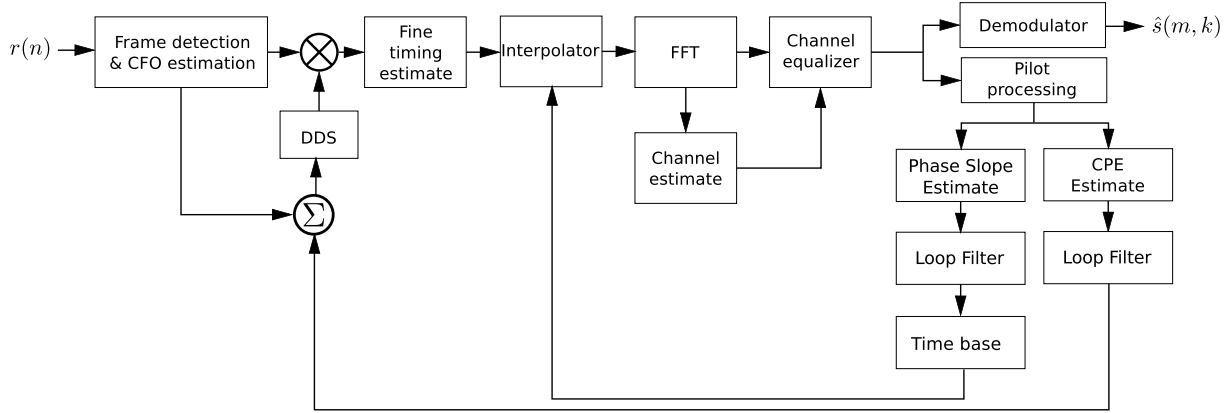


Figure 6.4: Time-domain tracking receiver architecture

corresponding to the assumption of little or no variations of the wireless channel over a frame time. It is important to note that any error in the fine timing estimate is absorbed by the channel estimate in the form of a linear phase term.

6.1.5 Timing and Frequency Tracking

Timing and frequency tracking loops can be grouped in the general categories of time-domain or frequency-domain. Synchronization tracking loops can be driven either by decision feedback or by pilot sub-carriers.

Time-domain control loops, presented in Figure 6.4, permit the direct control of the CFO frequency and sampling time; yet, the control loops have a large latency because they must traverse the FFT operator. Sampling time correction also has phase errors due to the frequency response of sample interpolators.

Frequency domain control loops, presented in Figure 6.5, have the advantage of a low latency in the control loop. They permit a more responsive and stable control loop. However, they are dependent on a fixed observation window and CFO correcting factor. In the case of the observation window, if the sampling frequency error is high, the observation window may fall in the next or previous symbol, causing a high inter symbol interference (ISI).

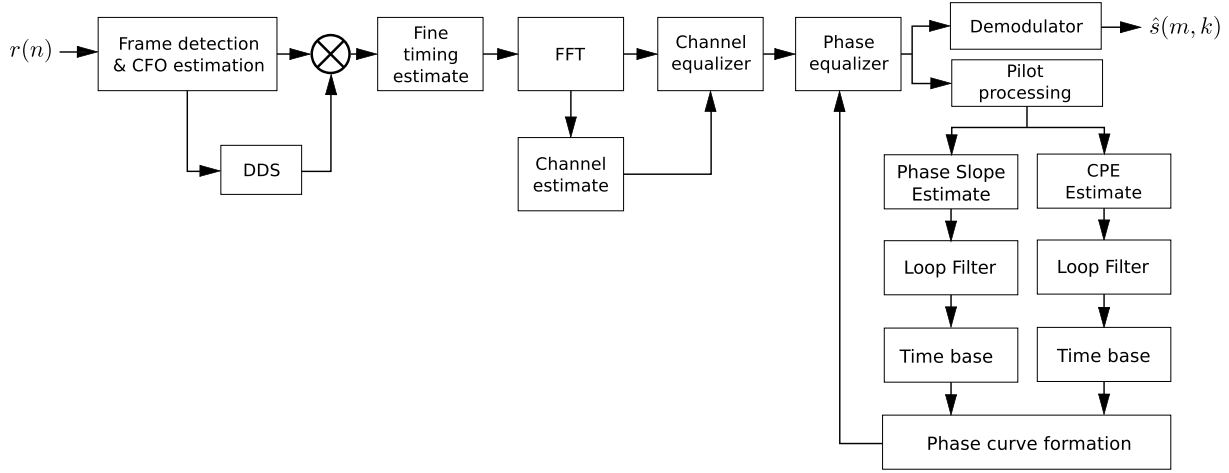


Figure 6.5: Frequency-domain tracking receiver architecture

6.1.6 Summary

The purpose of this section was to show how the acquisition algorithms commonly used for OFDM signals are based on time-domain processing: the autocorrelation function is used for frame detection, coarse timing estimation, and carrier frequency offset. The cross-correlation function, also based on time-domain sampling, is used for fine timing acquisition. While producing a satisfactory performance in non-interfered scenarios, these set of schemes would require expensive primary user rejection input filters to operate in presence of wide band interference. The tracking functions can be implemented either in the time or in the frequency domain, with complexity, stability, and range trade-offs.

6.2 Filtered Multitone

Despite its lower efficiency with respect to other FBMC methods, filtered multitone has a set of advantages that makes it worthy of consideration:

- The spectral efficiency is a function of the roll-off factor α . A signal with a roll-off factor of 0 has a spectral efficiency of 1 symbol/s per Hz at the price of a long symbol

duration. The roll-off factor can be set to any value between 0 and 1, trading spectral efficiency for symbol duration. An FMT system with a roll-off factor $\alpha = 1$ has an efficiency of 0.5 symbol/s per Hz.

- Orthogonality of the sub-carriers is achieved by their separation in the frequency domain. A minimum sub-carrier separation of $f_s(1 + \alpha)$ guarantees separation between the sub-carriers, without resorting to prototype filter constraints to guarantee orthogonality.
- FMT is typically non-critically sampled ($\alpha \neq 0$). An excess over-sampling rate is actually an advantage for frame detection tasks, as presented in subsequent sections.
- In OFDM/OQAM, a more efficient modulation type, the modulated data at a specific time and frequency is either real or imaginary. This precludes the use of generic training sequences.

6.2.1 Derivation of the FMT Transmitter

A generic FMT transmitter is presented in Figure 6.6. The maximum number of sub-carriers is N . The sub-carrier separation (and the sampling rate) is $f_s I / N$, where I is an integer interpolation rate fulfilling the condition $I \geq (1 + \alpha)N$, and f_s is the symbol rate. A system with an interpolation rate different to the number of sub-carriers ($I \neq N$) is referred to as non-critically sampled.

The pulse-shaping filter $g(n)$ must satisfy the Nyquist sampling conditions for ISI avoidance. A common choice is the root-raised cosine spectral shape, with roll-off factor α . The discrete-time index m is increased every time a new set of complex symbols $s^{(k)}(m)$ is transmitted. The up sampler $\uparrow I$ zero-pads the symbol sequence before feeding it to the pulse shaping filter. The set of shaped symbols is modulated to N equally-spaced frequency locations in the unit circle $\theta_k = 2\pi/N$ by the complex multipliers. Finally, the set of modulated signals is added to produce the FDM signal $x(n)$.

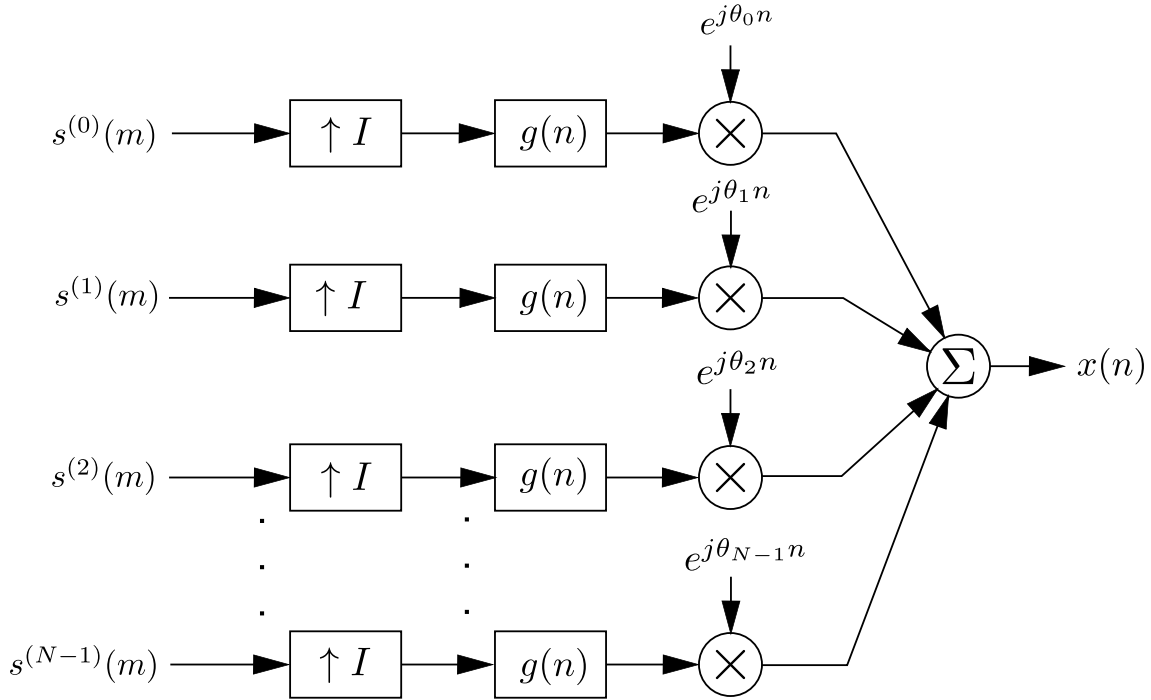


Figure 6.6: FMT transmitter

The implementation of this base modulator can be greatly optimized by deriving expressions in terms of polyphase filter banks and DFT operators. To reach these simplifications, we must first obtain an expression for $x(n)$:

$$x(n) = \sum_{k=0}^K \sum_{m=-\infty}^{\infty} s^{(k)}(m)g(n - mI)e^{j\theta_k n} \quad (6.5)$$

Regroup the elements of the equation:

$$x(n) = \sum_{m=-\infty}^{\infty} \sum_{k=0}^{N-1} e^{j\theta_k n} s^{(k)}(m)g(n - mI) \quad (6.6)$$

The inner summation corresponds to N -times the inverse DFT of the input symbol set.

Define the set of samples $a^{(i)}(m)$ as:

$$a^{(i)}(m) = \sum_{k=0}^{N-1} e^{j\theta_k i} s^{(k)}(m) = N \times \text{IFFT}\{s^{(k)}(m)\} \quad (6.7)$$

where the IFFT operator denotes a computationally efficient implementation of the inverse DFT of size N .

Partition the time index n in terms of multiples of the IFFT size: $n = \ell N + i$. Therefore,

$$x(\ell N + i) = \sum_{m=-\infty}^{\infty} a^{(i)}(m) g(\ell N + i - mI) \quad (6.8)$$

Following the notation presented in [129], perform a change of variable:

$$m = \left\lfloor \frac{\ell N}{I} \right\rfloor - q \quad (6.9)$$

the signal $x(\ell N + i)$ can then be expressed as:

$$\begin{aligned} x(\ell N + i) &= \sum_{q=-\infty}^{\infty} a^{(i)} \left(\left\lfloor \frac{\ell N}{I} \right\rfloor - q \right) g \left(\ell N + i - \left\lfloor \frac{\ell N}{I} \right\rfloor I + qI \right) \\ &= \sum_{q=-\infty}^{\infty} a^{(i)} \left(\left\lfloor \frac{\ell N}{I} \right\rfloor - q \right) g((\ell N)_I + i + qI) \end{aligned} \quad (6.10)$$

where $(\ell N)_I = \ell N - \left\lfloor \frac{\ell N}{I} \right\rfloor I = \ell N \bmod I$.

The prototype filter $g(n)$ can be expressed as a time-variant polyphase decomposition with period equal to $\text{lcm}(N, I)$, where lcm represents the least common multiple of N and I :

$$g^{(i)}(\ell, q) = g((\ell N)_I + i + qI) \quad (6.11)$$

Therefore, the output signal can be expressed as:

$$x(n) = x(\ell N + i) = \sum_{q=-\infty}^{\infty} a^{(i)} \left(\left\lfloor \frac{\ell N}{I} \right\rfloor - q \right) g^{(i)}(\ell, q) \quad (6.12)$$

This equation results in the efficient realization of a synthesis DFT filter bank as presented in Figure 6.7.

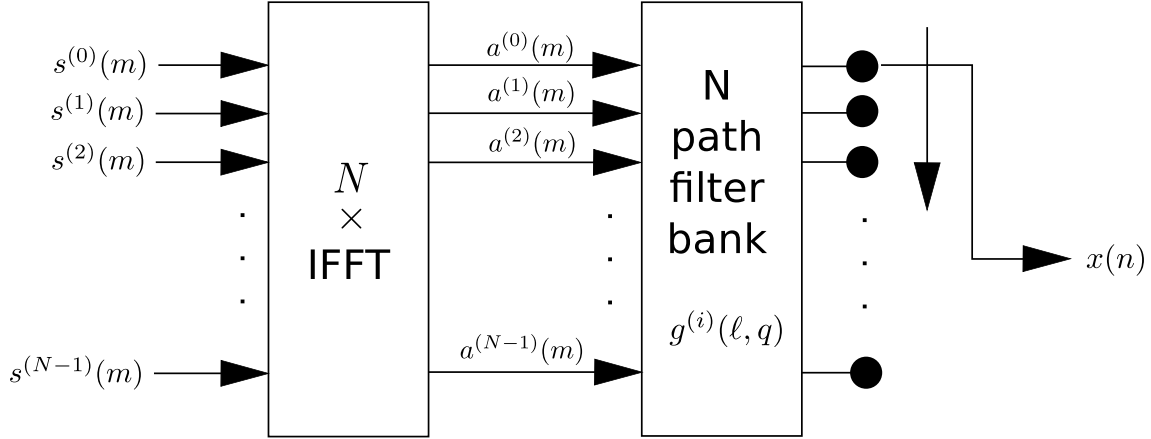


Figure 6.7: Modulated filter bank FMT transmitter

6.2.2 Derivation of the FMT Receiver

The optimum FMT receiver is found using the simplifications provided by the fact that the output is downsampled by a factor D that should be equal to the interpolation factor I in the transmitter side.

Consider the block diagram presented in Figure 6.8, where the filter $h_k(n)$ is a passband filter based on the baseband prototype $h(n)$ by the relation $h_k(n) = h(n)e^{j\theta_k n}$. The filter $h(n)$ is matched to the transmitter filter $g(n)$. The received signal $r(n)$ is obtained after the signal $x(n)$ is affected by the channel response and noise. The recovered set of subchannel signals after downconversion is denoted by $y(m, k)$, where m corresponds to the symbol index in the time domain, and k is the frequency bin index.

The output of the down converters is sampled at the times $n = mD$. Therefore, the set of signals $y^{(k)}(m)$ can be expressed as:

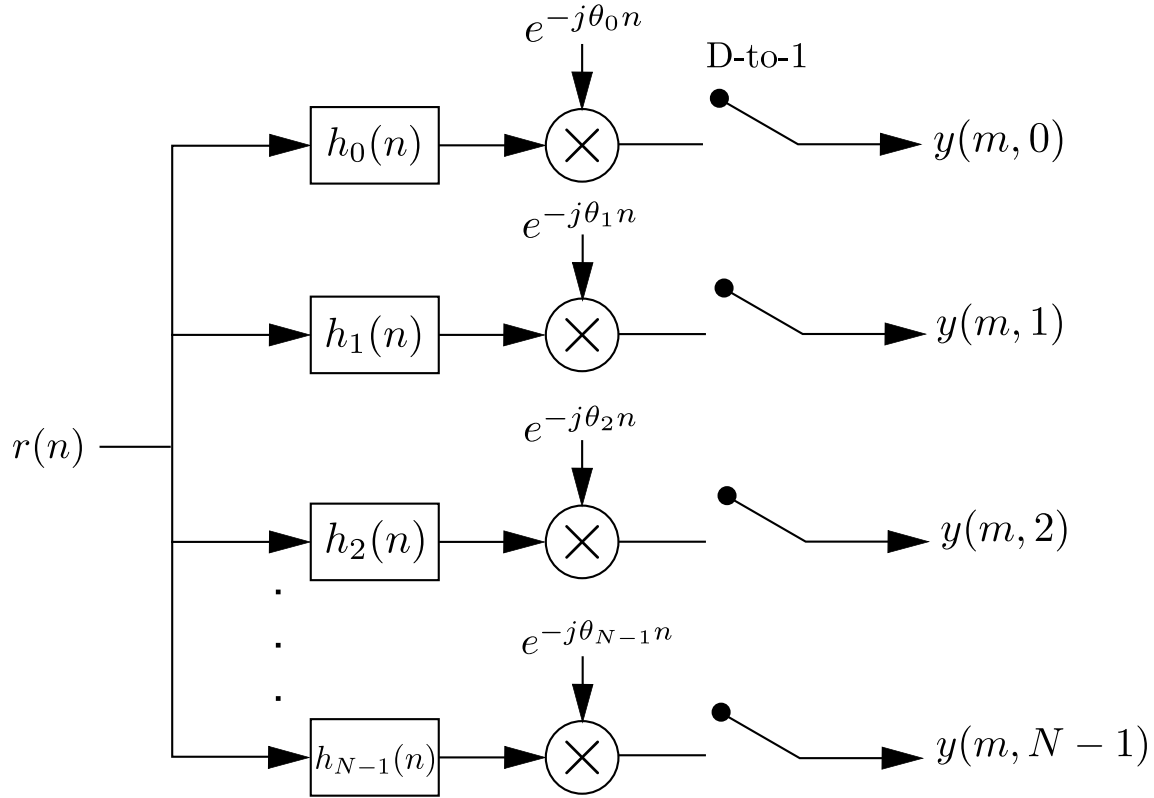


Figure 6.8: FMT receiver based on band pass filters, downconversion, and downsampling

$$y^{(k)}(m) = \sum_{n=-\infty}^{\infty} r(n)h_k(mD - n)e^{-j\theta_k mD} \quad (6.13)$$

Expanding the band pass filters $h_k(n)$ in terms of the base band prototype, we obtain:

$$y^{(k)}(m) = \sum_{n=-\infty}^{\infty} r(n)h(mD - n)e^{j\theta_k mD}e^{-j\theta_k n}e^{-j\theta_k mD} \quad (6.14)$$

After cancellation of exponential terms, and replacing $n = \ell N + i$ the equation can be written as:

$$y^{(k)}(m) = \sum_{\ell=-\infty}^{\infty} \sum_{i=0}^{N-1} r(\ell N + i)h(mD - \ell N + i)e^{-j\theta_k(\ell N + i)} \quad (6.15)$$

Noting that $e^{-j\theta_k \ell N} \equiv 1$, and rearranging the summations, we obtain:

$$y^{(k)}(m) = \sum_{i=0}^{N-1} \left[\sum_{\ell=-\infty}^{\infty} r(\ell N + i) h(mD - \ell N + i) \right] e^{-j\theta_k i} \quad (6.16)$$

the set of output values $y(m, k)$ for $k = 0, \dots, N-1$ equals the DFT of the output of the time-variant filter bank described by the inner summation. To provide an insight on the polyphase decomposition, let's perform a change of variable:

$$\ell = \left\lfloor \frac{mD}{N} \right\rfloor - q \quad (6.17)$$

the result of the inner summation can therefore be expressed as:

$$\begin{aligned} b^{(i)}(m) &= \sum_{q=-\infty}^{\infty} r \left(i - qN + \left\lfloor \frac{mD}{N} \right\rfloor N \right) h((mD)_N + qN - i) \\ &= \sum_{q=-\infty}^{\infty} r \left(i - qN + \left\lfloor \frac{mD}{N} \right\rfloor N \right) h^{(i)}(m, q) \end{aligned} \quad (6.18)$$

the output of the demodulator is

$$y^{(k)}(m) = \sum_{i=0}^{N-1} b^{(i)}(m) e^{-j\theta_k i} = \text{FFT}\{b(m)^{(i)}\} \quad (6.19)$$

it can be expressed in block form as:

$$\bar{y}(m) = \text{FFT}\{\bar{b}(m)\} \quad (6.20)$$

This representation leads to the efficient implementation shown in Figure 6.9.

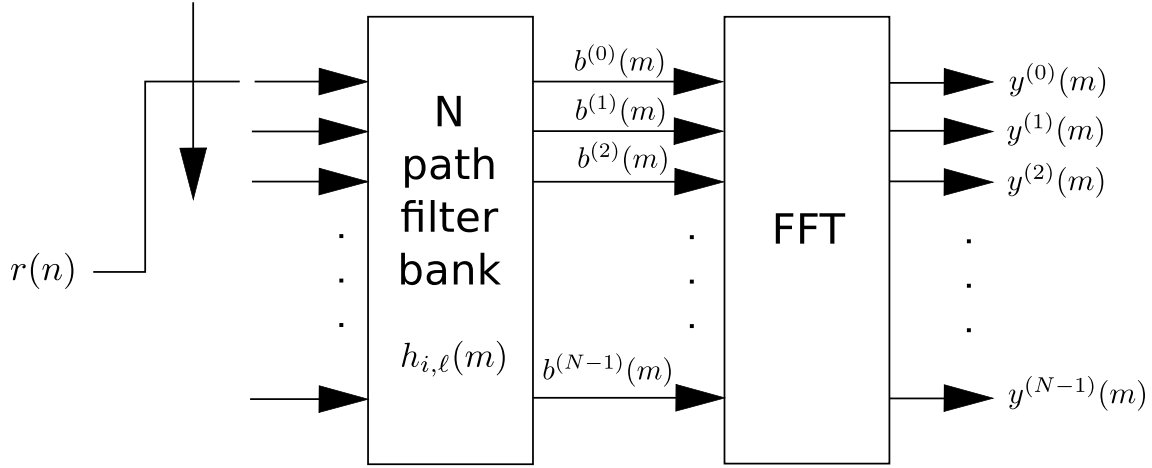


Figure 6.9: Efficient implementation of an FMT receiver

6.3 CAZAC Sequences

Constant Amplitude Zero Autocorrelation (CAZAC) sequences are also known as Zadoff-Chu sequences, defined as:

$$C_r(k) = e^{-j\pi r k^2 / N} \quad (6.21)$$

for N even, $k \in [0..N-1]$, and r coprime to N . The minus sign is added for consistency with the frequency-domain description of the sequences, to be clarified below. Odd N CAZAC sequences are not considered at this time.

The following are some relevant properties of the CAZAC sequences:

1. The cyclic autocorrelation equals zero for lags different to zero [131].

$$R_C(m) = \sum_{k=0}^{N-1} C_r(k) C_r^*(k+m) = \begin{cases} N & \text{for } m = 0 \pmod{N} \\ 0 & \text{for } m \neq 0 \pmod{N} \end{cases} \quad (6.22)$$

where the operator $*$ denotes the complex conjugate.

2. The DFT of a CAZAC sequence is also a CAZAC sequence [132, 133].

Demonstration

Define the Zadoff-Chu sequence $c_r(n) = e^{j\pi r n^2/N}$. Its DFT $C_r(k)$ is therefore:

$$C_r(k) = \sum_{n=0}^{N-1} e^{j\pi r n^2/N} e^{j2\pi k n/N} = \sum_{n=0}^{N-1} e^{j\pi(r n^2 + 2kn)/N} \quad (6.23)$$

Factor the term $e^{-j\pi r' n^2/N}$, where r' is the modular multiplicative inverse of r , defined as:

$$rr' = 1 \pmod{N} \quad (6.24)$$

the result is:

$$\begin{aligned} C_r(k) &= e^{-j\pi r' n^2/N} \sum_{n=0}^{N-1} e^{j\frac{\pi}{N} r(n^2 + 2knr' + r'^2)} \\ &= e^{-j\pi r' n^2/N} \sum_{n=0}^{N-1} e^{j\frac{\pi}{N} r(n+kr')^2} \\ &= c_{r'}^*(k) \sum_{n=0}^{N-1} c_r(n + kr') \\ &= c_{r'}^*(k) C_r(0) \end{aligned} \quad (6.25)$$

the last step is possible because the term kr' is an integer and the summation happens over a full period of the sequence $c_r(n)$.

Hence, the result of the DFT of $c_r(n)$ is the complex conjugate of its dual sequence $c_{r'}(k)$ phase-shifted by $C_r(0)$, also a CAZAC sequence.

3. CAZAC sequences have constant amplitude and consequently, a low peak-to-average ratio (PAPR).
4. Modulation property (or frequency-time offset ambiguity): When $r = N - 1$, a cyclic time shift is equivalent to modulating the sequence [134].

Demonstration

Take a cyclic shift θ of the sequence $c_r(n)$:

$$\begin{aligned}
 c_r(n - \theta) &= e^{j\frac{\pi r}{N}(n+\theta)^2} \\
 &= e^{j\frac{\pi r}{N}n^2} e^{j\frac{2\pi r}{N}n\theta} e^{j\frac{\pi r}{N}\theta^2} \\
 &= c_r(n) e^{j\frac{2\pi r}{N}n\theta} e^{j\frac{\pi r}{N}\theta^2} \\
 &= c_r(n) e^{j\frac{2\pi(N-1)}{N}n\theta} e^{j\frac{\pi r}{N}\theta^2} \\
 &= c_r(n) e^{j\frac{2\pi\theta}{N}n} e^{j\frac{\pi r}{N}\theta^2}
 \end{aligned} \tag{6.26}$$

We observe that the original sequence $c_r(n)$ is modulated by a complex exponential with frequency $\omega = 2\pi\theta/N$, with an initial phase term $e^{j\frac{\pi r}{N}\theta^2}$.

The cyclic autocorrelation properties of some sequences is used for sequence synchronization in direct-sequence spread spectrum systems as well as in OFDM systems to perform timing synchronization acquisition. In the case of CAZAC sequences, the values of the cyclic autocorrelation for non-zero time lags are identically zero, reducing the probability of false detection.

The modulation property permits the cross-correlation function calculated between a received preamble and a memory-stored version of the sequence to produce a detection peak under various carrier frequency offsets. A frequency offset is mapped to a time-domain shift, that can be compensated with equalization techniques in the receiver.

The constant amplitude property together with the DFT property of CAZAC sequences has the following advantage: it can produce a training sequence with constant amplitude in all the sub-carriers, and a constant amplitude in the time-domain, providing minimum PAPR. This allows the transmitter to maximize its output power, increasing the probability of frame detection.

The use of CAZAC sequences is not mandatory to implement the algorithm; however,

other preamble sequences are not optimal in the PAPR sense, and can only provide a cross-correlation peak under a very low CFO.

A comparison of the cross-correlation performance of CAZAC vs. pseudo-random binary sequences (PRBS) based on the peak of the cross correlation function for varying CFO values is presented in Figure 6.10. The periodic behavior of the curve is a result of the frequency-time offset ambiguity. For CFOs that are integers of the sub-carrier separation, a new peak is produced, with an offset of one sample in the time domain.

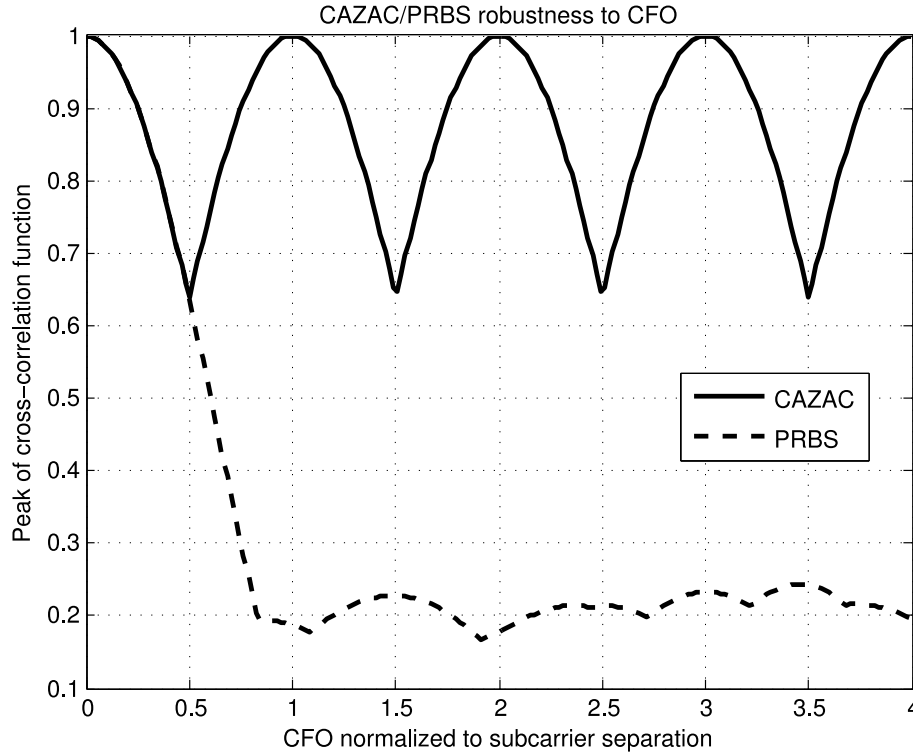


Figure 6.10: Robustness of CAZAC sequences to carrier frequency offset

5. Fractional time-offsets.

Fractional time offsets produce spectral leakage effects, and due to the frequency-time offset ambiguity, are equivalent to carrier frequency offset. Assuming low CFO conditions, fractional time offset would be the main contributor to spectral leakage. To model the effect of a fractional offset over the cyclic autocorrelation we resort to

the correlation property presented in Equation 6.30.

If a CAZAC sequence in the time domain is considered:

$$c_r(n) = e^{j\pi r n^2 / N} \quad (6.27)$$

Using the ambiguity property and the cross-correlation theorem, the cross-correlation of a CAZAC sequence shifted by the fractional amount θ with a non-shifted reference sequence can be calculated as follows, using the result from Equation 6.26:

$$\begin{aligned} R_{cc}(k) &= \sum_{n=0}^{N-1} c_r(n+k+\theta) c_r^*(n) \\ &= \sum_{n=0}^{N-1} c_r(n+k) c_r^*(n) e^{j\frac{2\pi\theta}{N}n} e^{j\frac{\pi r}{N}\theta^2} \\ &= \sum_{n=0}^{N-1} c_r(n) c_r^*(n) e^{j\frac{2\pi k}{N}n} e^{j\frac{\pi r}{N}k^2} e^{j\frac{2\pi\theta}{N}n} e^{j\frac{\pi r}{N}\theta^2} \\ &= e^{j\frac{\pi r}{N}(k^2+\theta^2)} \sum_{n=0}^{N-1} e^{j\frac{2\pi\theta}{N}n} e^{j\frac{2\pi k}{N}n} \\ &= N e^{j\frac{\pi r}{N}(k^2+\theta^2)} \text{IFFT} \left\{ e^{j\frac{2\pi\theta}{N}n} \right\} \end{aligned} \quad (6.28)$$

The magnitude of the cross-correlation function $R_{cc}(k)$ for fractional values of θ can be found as:

$$|R_{cc}(k)| = \left| \frac{\sin(\pi(k-\theta))}{N \sin(\pi(k-\theta)/N)} \right| \quad (6.29)$$

Figure 6.11 presents a detail of the sinc-type envelope for a fractional delay of 0.7. The side lobes are a consequence of time misalignment, but do not convey interference power from a potential primary user.

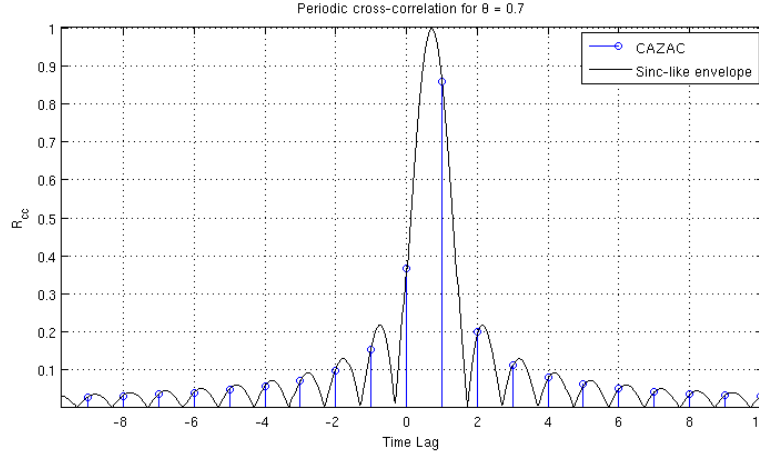


Figure 6.11: Detail of sinc envelope with a fractional delay of $\theta = 0.7$

6.4 An Algorithm for Frame Detection and Timing Offset Estimation with Interference Rejection

Tian et al. proposed a frequency domain method for time synchronization of OFDM signals in presence of narrowband interference. The authors employ CAZAC sequences and excision of the interfered frequency bins [134]. The cross-correlation operation is performed in the frequency domain using accumulation of time-lags, a computationally inefficient implementation. Frame detection is not considered.

The algorithm proposed in this section performs the functions of interference rejection, frame detection, fine timing estimation, and channel estimation using block processing in the frequency domain. The FFT/IFFT processing core can be re-used if latency constraints are satisfied. An appropriate frame structure is also proposed. While the following description is directly applicable to the FMT case, OFDM can be seen as a particular case with a rectangular symbol shaping function of length one symbol.

6.4.1 Efficient Calculation of the Cyclic Cross-Correlation Function

The cross-correlation property of the inverse Fourier transform states that:

$$\bar{x} \circledast \bar{y} = \text{IFFT} \left\{ \overline{X^* Y} \right\} \quad (6.30)$$

where the operator $\circledast$ represents the cyclic cross-correlation of length N between the two time-domain vectors $\bar{x}$ and $\bar{y}$, and $\overline{X}$ and $\overline{Y}$ correspond their FFT of size N , respectively. The operator $*$ represents the complex conjugate.

In a typical OFDM receiver, the cross-correlation function used for fine timing estimation is calculated between an observation vector of duration N that slides one sample at a time, and a local copy of the preamble sequence. This operation requires N complex products and $N - 1$ complex additions per sample; that is N^2 complex products and $N(N - 1)$ complex additions to obtain the values of the cross-correlation function for N consecutive time lags. This computationally expensive process is usually simplified by using the signs of the input signals instead of their full-precision values [135]. Using only the signs introduces quantization errors to the cross-correlation algorithm.

Processing N time lags over a block of N samples using the IFFT reduces the computational complexity to $\mathbf{O}(N \log(N))$.

The calculation in the frequency-domain of the cyclic cross-correlation for block data is clear for the non-shaped OFDM case; however, it is not intuitive in the case of FMT. In this case, the implementation details depend on the upsampling rate. The following example is valid for an upsampling rate of two, but can be expanded to other upsampling rates in the interval $[1, 2]$.

The signal shown in Figure 6.12 represents the ideal sampling instants at the output of the receive-side filter bank for an isolated symbol. In a receiver without information about

the timing at the transmitter, the sampling instant is a uniformly distributed real random variable in the interval $[0, N)$. For $\theta \neq 0$, the signal to noise ratio is sub-optimal. This imposes a penalty on the probability of frame detection P_D .

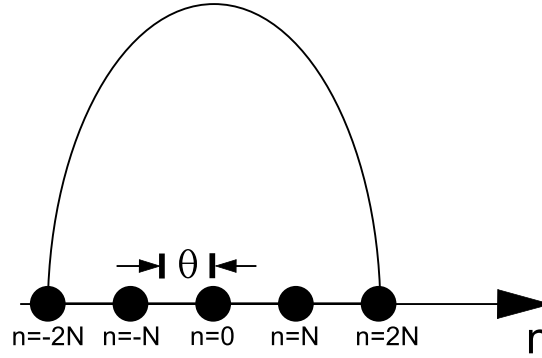


Figure 6.12: Random sampling time offset at the output of the receive-side filter bank

An isolated single-symbol preamble containing the frequency-domain CAZAC sequence $C_{N-1}(k)$ as the input signal is transmitted using the system presented in Figure 6.13 . The complex baseband time-domain output $x(n)$ can therefore be represented as:

$$x(n) = c_{N-1}(n)g(n) \quad (6.31)$$

where $c_{N-1}(n)$ is the $n \bmod N$ element of the time-domain CAZAC sequence obtained by $c_{N-1}(n) = \text{IFFT}\{C_{N-1}(k)\}$ and $g(n)$ is the baseband prototype of the root-raised cosine time-domain filter response.

To better describe the block operations, we will represent the receiver input signal $r(n)$ as: $r^{(i)}(\ell)$, where $\ell = \lfloor \frac{n}{N} \rfloor$ and $i = n - m$. Hence, $r^{(i)}(\ell)$ represents the i^{th} element of the ℓ^{th} observation vector. Alternatively, we can denote the ℓ^{th} observation vector as $\bar{r}(\ell)$.

Assume an AWGN channel with circular noise $w(n)$ and a receiver time offset of θ samples, where $\theta \in [0, N)$. Therefore, the signal arriving at the receiver can be modeled as $r(n) = x(n - \theta) + w(n)$, expressed in vector notation as:

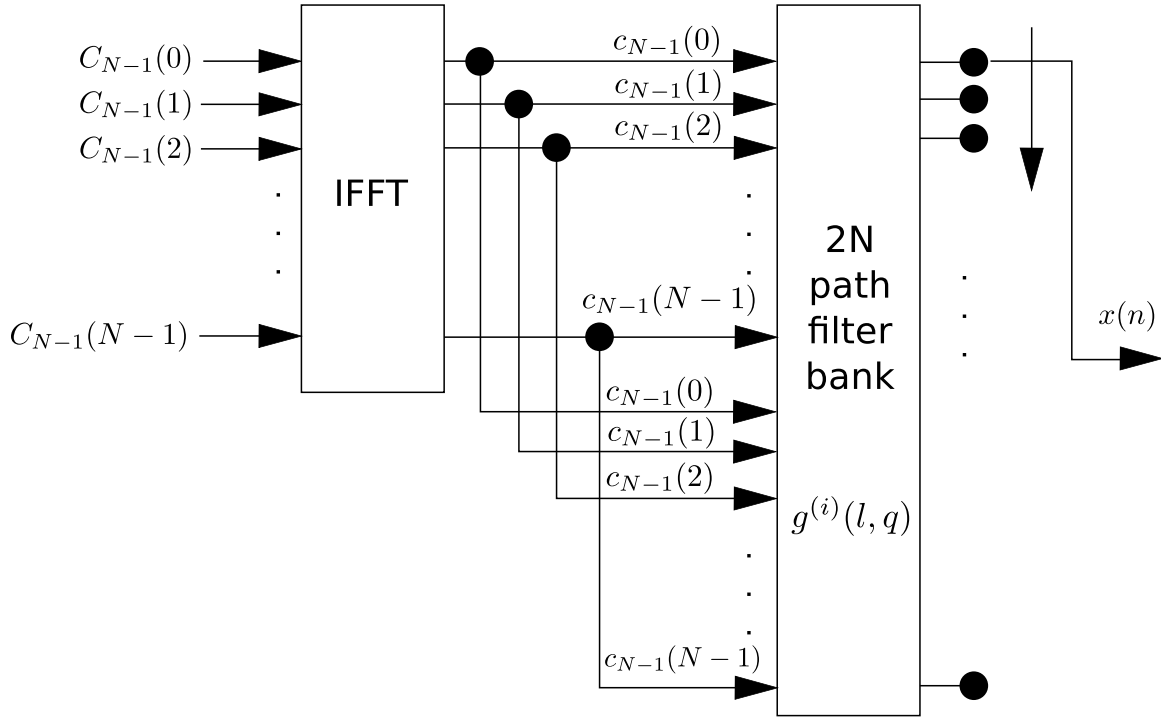


Figure 6.13: FMT preamble generation with upsampling rate of 2

$$\begin{aligned}
 \bar{r}(\ell) &= \bar{x}_\theta(\ell) + \bar{w}(\ell) \\
 &= \bar{c}_{N-1, \theta(\ell)} \circ \bar{g}_\theta(m) + \bar{w}(\ell)
 \end{aligned} \tag{6.32}$$

where $\bar{x}_\theta(\ell)$ represents an N -sample window with a timing offset of θ samples, and the operator $\circ$ represents the Schur or element-wise product.

During the frame detection stage, the cyclic time shift and downsampling by D in the receive filter bank is suspended to enable obtaining a result at the output of the filter bank every N samples instead of D . This oversampling at the receive side is a key strategy to enhance the probability of frame detection. After passing through the static receive-side filter bank, the signal $b^{(i)}(\ell)$ can be modeled as:

$$b^{(i)}(\ell) = r^{(i)}(\ell) \star h^{(i)}(\ell) \tag{6.33}$$

where $h^{(i)}(\ell)$ represents the impulse response of the ℓ^{th} branch of the receiver's matched filter bank, and the symbol $\star$ denotes the convolution operator. We can express Equation 6.33 in terms of a point-wise vector convolution:

$$\bar{b}(\ell) = \bar{r}(\ell) \star \bar{h}(\ell) \quad (6.34)$$

The block calculation of the cross-correlation function is presented in Figure 6.14. The benefit of performing the product in the frequency-domain will become apparent when describing the interference rejection function.

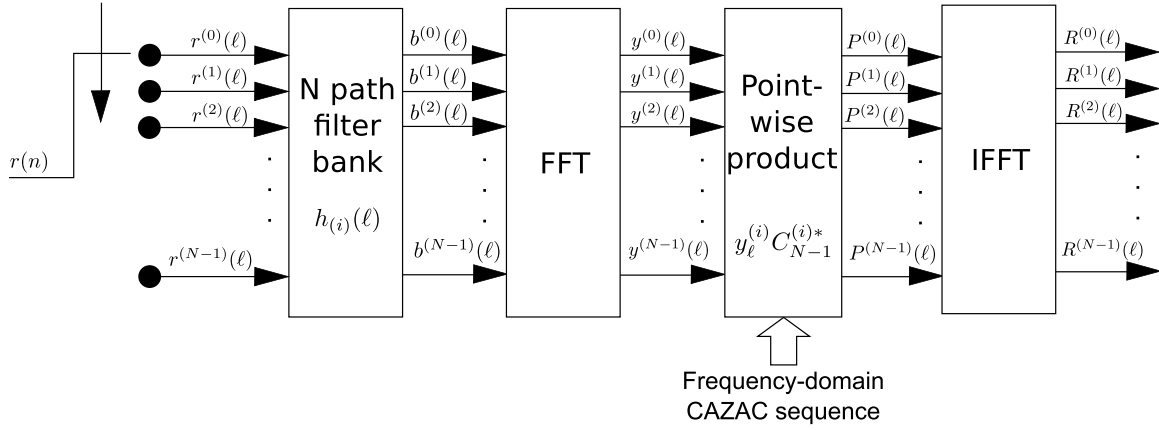


Figure 6.14: Block processing of the cross-correlation operation

At the m^{th} FFT block output, the cyclic cross-correlation vector containing the time lags from 0 to $N - 1$ is calculated as:

$$\bar{R}(\ell) = \text{IFFT} \{ \bar{C}_{N-1}^* \bar{y}(\ell) \} \quad (6.35)$$

Putting together Equations (6.31) and (6.33), and considering the cross-correlation identity (6.30), we obtain:

$$\bar{R}(\ell) = \bar{c}_{N-1} \otimes [\bar{r}(\ell) \star \bar{h}(\ell)] \quad (6.36)$$

If we exchange the order of the cross-correlation and the convolution operators, we obtain:

$$\overline{R}(\ell) = [\bar{c}_{N-1} \circledast \bar{r}(\ell)] \star \bar{h}(\ell) \quad (6.37)$$

Now we replace $\bar{x}(\ell)$ with its constituent signals:

$$\overline{R}(\ell) = [\bar{c}_{N-1} \circledast (\bar{c}_{N-1} \circ \bar{g}(\ell) + \bar{w}(\ell))] \star \bar{h}(\ell) \quad (6.38)$$

the signal and noise terms can now be separated:

$$\overline{R}(\ell) = \bar{c}_{N-1} \circledast (\bar{c}_{N-1, \theta} \circ \bar{g}_\theta(\ell)) \star \bar{h}(\ell) + \bar{w}(\ell) \star \bar{h}(\ell) \quad (6.39)$$

The term $\Psi(m) = \bar{c}_{N-1} \circledast (\bar{c}_{N-1, \theta} \circ \bar{g}_\theta(m))$ presents a loss of orthogonality due to the term $\bar{g}_\theta(m)$. In the case of zero-autocorrelation sequences, it can be shown that:

$$\Psi(\ell) = \bar{c}_{N-1} \circledast (\bar{c}_{N-1, \theta} \circ \bar{g}_\theta(\ell)) = \delta_\theta \star_N \text{IFFT} \{g(\ell)\} \quad (6.40)$$

where the operator $\star_N$ represents the circular convolution of size N , and δ_θ is the Kronecker delta function circularly shifted by θ elements that can be extended to the non-integer case.

We can therefore re-express Equation 6.39 as:

$$\overline{R}(\ell) = \Psi(\ell) \star \bar{h}(\ell) + \bar{w}(\ell) \star \bar{h}(\ell) \quad (6.41)$$

The convolution with the receive-side filter $\bar{h}(\ell)$ can be seen as a matched-filter operation in the case of perfect alignment ($\theta = 0$). In other cases it can be considered as a smoothing filtering operation over consecutive cross-correlation vectors.

6.4.2 Sub-carrier Excision

In the previous subsection it was demonstrated how the IFFT operator permits the calculation of the cross-correlation between the reference and the received training sequence, as long as the training symbol is transmitted in isolation. There is an orthogonality penalty caused by the time-domain shape of the transmit-side filter that depends on the section of the filter captured at the observation window.

An advantage of the translation of the observation window to the frequency domain required to perform the operation described in Equation 6.30 is that the weight of the frequency components can be manipulated by the point-to-point multiplication of the output of the FFT by a weight vector $\bar{V}$, either to compensate for poor channel conditions at a set of sub-carriers, or to cancel the effect of an interferer or primary user in the case of an overlay system. Sub-carrier excision is defined as the process of zeroing a set of sub-carriers as presented in Figure 6.15. Sub-carrier excision has the advantage of canceling interference directly in the frequency domain, saving the need for expensive time-domain interference rejection filters, inheriting the characteristics of the receive-side prototype filter.

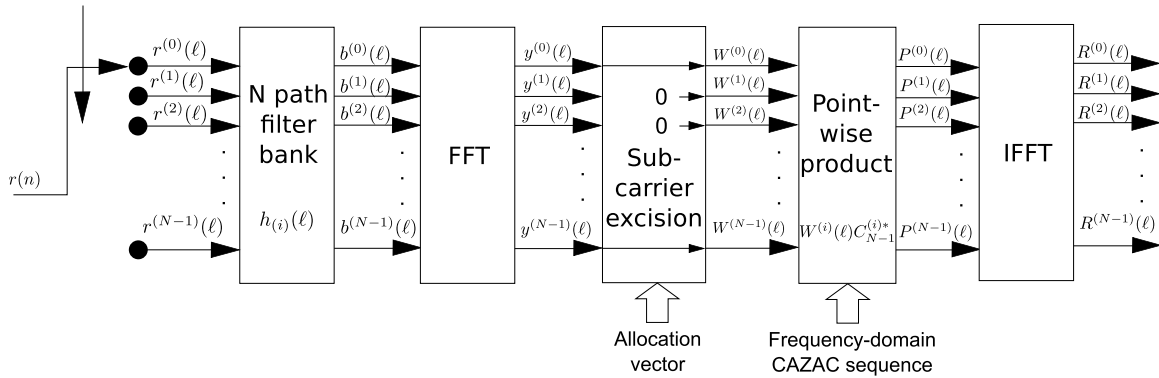


Figure 6.15: Block cross-correlation with sub-carrier excision

Sub-carrier excision, or equivalently, the product with the weight vector $\bar{V}$ manifests in the auto-correlation domain as a loss of orthogonality of the time-domain sequences corresponding to the excised CAZAC frequencies in the time domain. The number of excised

sub-carriers accounts for a loss in the amplitude of the peak, while their distribution produces different patterns of smearing in the cross-correlation function.

6.4.3 Frame Detection Hypothesis Test

Frame detection is based in the detection of a peak in the circular cross-correlation function. A by-product of the peak detection is that its location is a direct estimate of the timing offset θ .

The timing offset is estimated by:

$$\hat{\theta}_\ell = \arg \max_p \{|R(\ell, p)|\} \quad (6.42)$$

where $R(\ell, p)$ represents the p^{th} lag of the cyclic cross-correlation vector $\overline{R}(\ell)$ and the operator $|\cdot|$ is the magnitude of a complex scalar.

The hypothesis test is performed as:

$$H_\ell = |R(\ell, \hat{\theta}_\ell)|^2 > \gamma \frac{\rho}{N} \sum_{p=0}^{N-1} |R(\ell, p)|^2 + \epsilon \quad (6.43)$$

where the parameter γ is a decision threshold chosen according to the desired probability of detection P_D , ρ is the number of active sub-carriers, and ϵ is a small positive number required to avoid stray detections.

6.4.4 SNR Estimation

An estimate of the signal-to-noise ratio (SNR) can be obtained if a constant value is assumed in the time domain transmit filter bank, as in the case of OFDM signals, as well as a flat channel response. Appropriate adjustments can be made in the case of a non-constant filter response, as in the case of FBMC.

Assumming an AWGN channel, the ratio γ of the expected values of the peak and sum of the cyclic cross-correlation is defined as follows:

$$\gamma = \frac{\mathbf{E} \{|R(\ell, \theta_m)|^2\}}{\mathbf{E} \left\{ \sum_{p=0}^{N-1} |R(\ell, p)|^2 \right\}} = \frac{1}{N} \frac{\rho + \sigma^2/\rho^2}{1 + \sigma^2} \quad (6.44)$$

where σ^2 represents the noise variance. A signal representation with unit amplitude in the frequency domain is assumed.

The SNR can then be found as: $SNR = 1/\sigma^2$. Therefore, the SNR can be estimated as follows:

$$SNR = \frac{1 - N\rho^2\gamma}{\rho^2(N\gamma - \rho)} \quad (6.45)$$

6.4.5 Frame Design

Even if a frame can be detected and the timing offset can be estimated, the oversampling feature of FMT will result in an ambiguity of half a symbol. Therefore, a second isolated preamble symbol is required. The timing offset estimate obtained after the first preamble symbol is used to align the input buffer. Following the alignment, a peak search over the second preamble symbol permits to find the optimum symbol sampling instant. An advantage of this approach is that the value of the signal at the peak can be used to calculate a memoryless channel estimate that can be used to feed a trivial one-tap equalizer, or can serve as an initialization value for an equalizer that may be required by the fact that a FBMC signal does not feature a cyclic prefix.

The frame structure required for the case of a filter duration of two symbols is presented in Figure 6.16. An implementation with a smaller a smaller roll-off factor may require a longer blank time between training symbols.

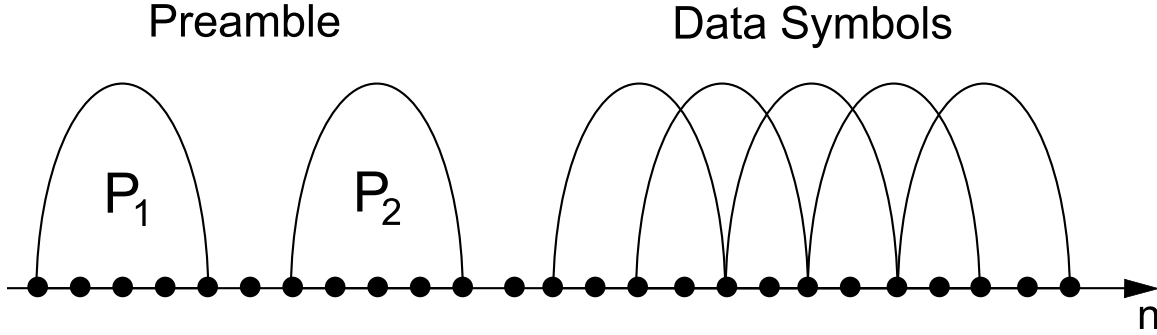


Figure 6.16: Frame structure with two isolated preambles

6.4.6 Channel Estimation

As presented in Section 6.4.5, a channel estimate can be obtained when the peak of the second preamble symbol is detected. If the channel response is assumed to be flat over the bandwidth of a sub-carrier, a single tap channel estimate is enough to equalize the input signal. The FFT output at the k th sub-carrier is then:

$$y^{(k)}(\ell) = C_r^{(k)} H^{(k)} + W^{(k)}(\ell) \quad (6.46)$$

where $H^{(k)}$ is the channel response and $W^{(k)}(\ell)$ is the noise contribution. Considering the unit-amplitude of the CAZAC sequence, the channel estimate $\hat{H}^{(k)}$ can be found as:

$$\hat{H}^{(k)} = C_r^{(k)*} y^{(k)}(\ell) \quad (6.47)$$

this operation is already performed as part of the peak detection algorithm and all that is required is to store this partial result at peak detection time.

6.4.7 Algorithm Summary

The algorithm for frame detection and timing estimation, is summarized in Algorithm 2. A similar algorithm is applied to the second preamble to disambiguate the sampling window and obtain the channel estimate, as presented in Algorithm 3.

Algorithm 2 Algorithm for frame detection and timing offset correction

```

 $\ell \leftarrow 0$ 
 $peak \leftarrow 0$ 
 $Preamble1Det \leftarrow 0$ 
while  $Preamble1Det \neq 1$  do
   $\bar{y}(\ell) \leftarrow \text{FFT} \{ \bar{b}(\ell) \}$ 
   $\bar{W}(\ell) \leftarrow \bar{V} \bar{C}_{N-1}^* \bar{y}(\ell)$  { $\bar{V}$  performs interference rejection}
   $\bar{R}(\ell) \leftarrow \text{IFFT} \{ \bar{W}(\ell) \}$  {Calculation of the cyclic cross-correlation}
   $\hat{\theta}_\ell \leftarrow \arg \max_p \{ |R(\ell, p)| \}$ 
   $H(\ell) \leftarrow |R(\ell, \hat{\theta}_\ell)|^2 > \gamma \frac{\rho}{N} \sum_{p=0}^{N-1} |R(\ell, p)|^2 + \epsilon$  {Hypothesis test}
  if  $peak < |R(\ell, \hat{\theta}_\ell)|^2$  then
     $peak \leftarrow |R(\ell, \hat{\theta}_\ell)|^2$ 
  end if
  if  $\left( |R(\ell, \hat{\theta}_\ell)|^2 < peak \right)$  and  $H(\ell - 1) = 1$  then
     $Preamble1Det \leftarrow 1$  {Peak detected while hypothesis tested positive}
    Align input buffer by  $\hat{\theta}_{\ell-1}$ 
  end if
   $\ell \leftarrow \ell + 1$ 
end while

```

Algorithm 3 Algorithm for disambiguation and channel estimation

```

 $peak \leftarrow 0$ 
 $Preamble2Det \leftarrow 0$  {Wait for second preamble}
while  $Preamble2Det \neq 1$  do
   $\bar{y}(\ell) \leftarrow \text{FFT} \{ \bar{b}(\ell) \}$ 
   $\bar{W}(\ell) \leftarrow \bar{V} \bar{C}_{N-1}^* \bar{y}(\ell)$  { $\bar{V}$  performs interference rejection}
   $\bar{R}(\ell) \leftarrow \text{IFFT} \{ \bar{W}(\ell) \}$  {Calculation of the cyclic cross-correlation}
   $\hat{\theta}_\ell \leftarrow \arg \max_p \{ |R(\ell, p)| \}$  { $\hat{\theta}_\ell$  expected to equal 1}
   $H(\ell) \leftarrow |R(\ell, \hat{\theta}_\ell)|^2 > \gamma \frac{\rho}{N} \sum_{p=0}^{N-1} |R(\ell, p)|^2 + \epsilon$  {Hypothesis test}
  if  $peak < |R(\ell, \hat{\theta}_\ell)|^2$  then
     $peak \leftarrow |R(\ell, \hat{\theta}_\ell)|^2$ 
  end if
  if  $(|R(\ell, \hat{\theta}_\ell)|^2 < peak)$  and  $H(\ell - 1) = 1$  then
     $Preamble2Det \leftarrow 1$  {Peak detected for disambiguation}
     $\hat{H} \leftarrow \bar{W}(\ell - 1)$  {Channel estimate is a by-product of the algorithm}
  end if
   $\ell \leftarrow \ell + 1$ 
end while

```

Table 6.2: Set of parameters for simulation of frame detection

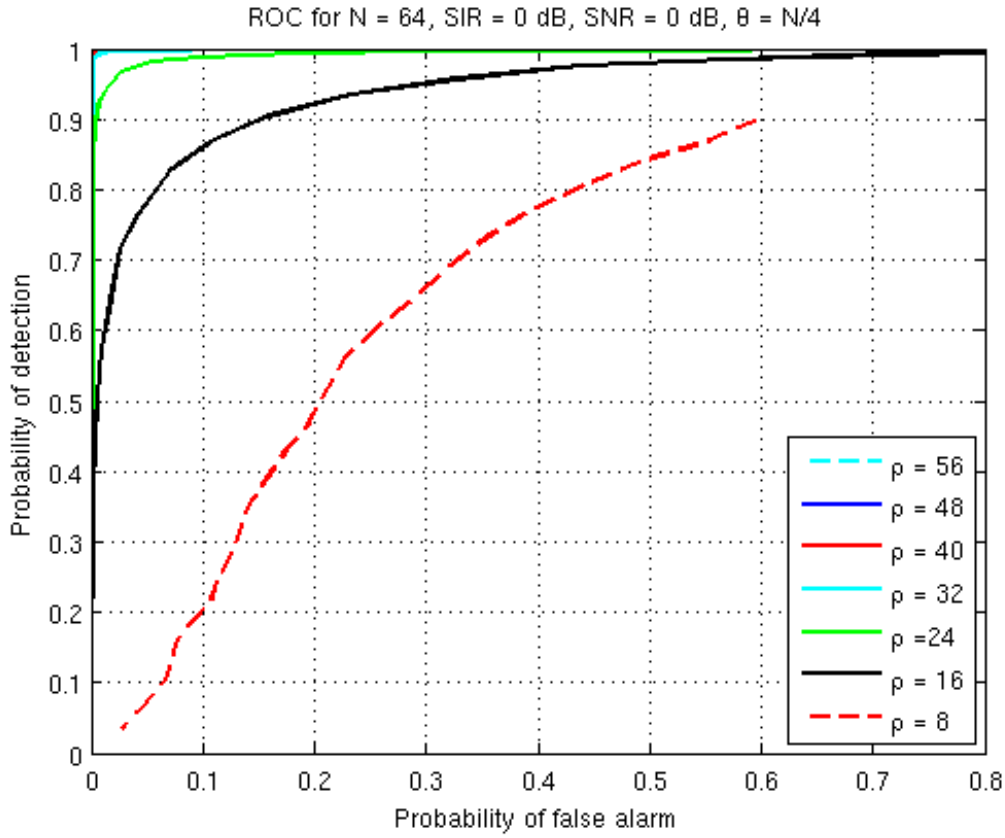
Specification	Value
FFT Size (N)	64
Sampling rate	20 MHz
Signal to interference ratio	0 dB
Signal to noise ratio	0 dB
Integer timing offset	$N/4$
Oversampling rate	2
Symbol shaping filter	Root-raised cosine with $\alpha = 0.99$
Number of iterations	10,000

6.5 Results

The receiver operating characteristic (ROC) for various values of the number of active sub-carriers ρ is obtained by simulation of the overall system, under the set of parameters presented in Table 6.2. Figure 6.17 present the ROC for a fractional timing offset of $\theta = N/4$. This is the worst case in terms of the time mismatch of the matched filter receiver.

These results demonstrate the ability of FBMC methods to deal with primary user interference at frame detection time. The performance of the algorithm degrades rapidly for sub-carrier occupations of less than 50%. Consequently the sparseness conditions to obtain computational savings in the IFFT operator at the transmit and receive side are not fulfilled. Further, the IFFT-based cross correlation requires the use of the full-size FFT. This makes core re-use a better approach for hardware optimization on the receive side.

The receiver operating characteristic for a number of active sub-carriers $\rho = 32$ and different SNR values is presented in Figure 6.18. Degradations are noticeable for values of SNR lower than 0 dB.

Figure 6.17: ROC with timing offset $\theta = N/4$

6.6 Conclusion

This chapter presented the properties of CAZAC sequences and showed that their frequency-time offset ambiguity can be used as an advantage to get a cross-correlation peak in the presence of carrier frequency offset. The mapping of frequency error to timing must be compensated by the succeeding equalizing and tracking states.

CAZAC sequences can deliver a cross-correlation peak for CFOs that are multiples of the sub-carrier separation, but cannot resolve integer CFO without mapping it to non-existent time offsets. This situation requires the following preamble symbols to permit the measurement of integer CFOs. The mixed use of preambles can overcome the limitations of CAZAC sequences.

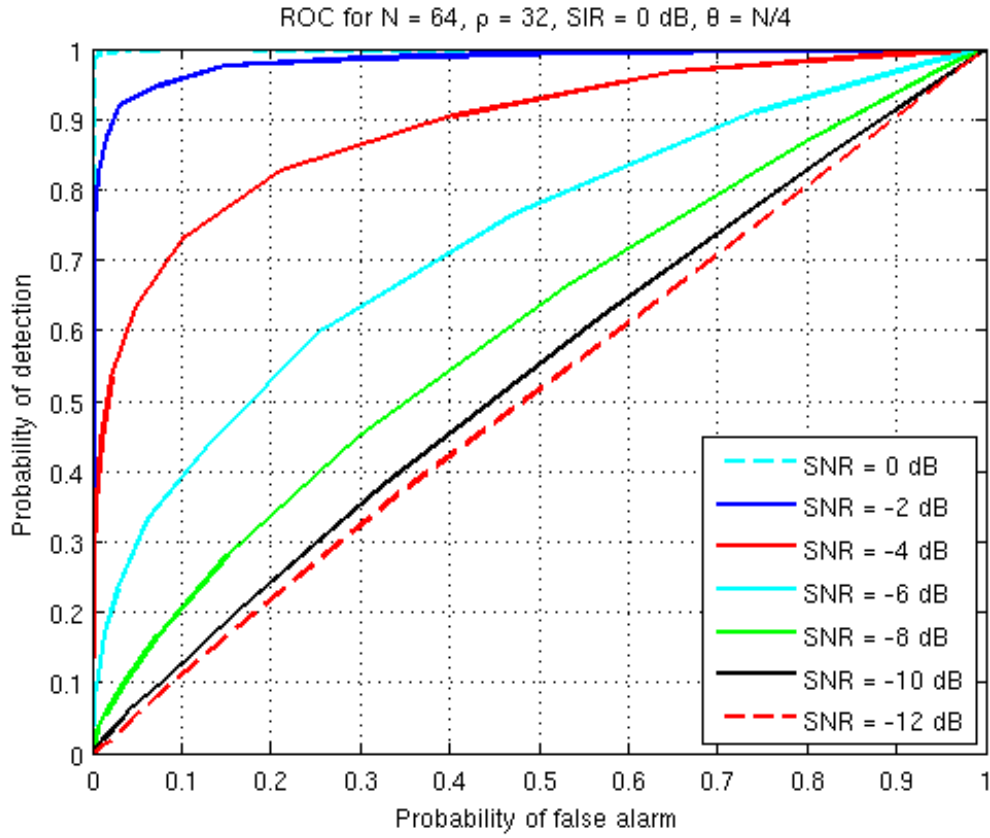


Figure 6.18: ROC with timing offset $\theta = N/4$ and $\rho = 32$

Interference rejection using frequency bin excision comes at the cost of the smearing of the cross-correlation peak. Even if the SNR is not affected by the excision, because noise at the excised sub-carrier bins is canceled, the lower amplitude of the peak reduces the effective peak-to-average ratio of the cross-correlation vector.

Chapter 7

Hardware Architecture

This chapter describes the hardware design and architecture adopted for two of the prototypes developed as part of the project. The first one is an OFDM radio inspired in the physical layer of the IEEE 802.11a/g standard. Forward error correction (FEC) was not implemented for this project. With the proper parameterization this development can be used as part of a standard-compliant transceiver. The second prototype is a filtered multitone (FMT) transceiver that shares the equalization, demodulation, and synchronization modules developed for the OFDM radio. It includes a hardware-efficient architecture for performing the functions of spectrum sensing, frame detection, timing offset estimation, channel estimation, and demodulation.

7.1 Hardware Architecture of the OFDM Radio

The OFDM radio was developed to support arbitrary allocation vectors. This flexibility allows the radio to be parameterized to implement the IEEE 802.11a/g standard. Although the description of the signal field in the frame is hard-coded, it can be updated to make the radio fully compliant. This radio is useful as a test bed to evaluate the mutual interference

Table 7.1: Allocation vector coding

Sub-carrier type	Code (base 2)
Null	00
Data	01
Positive Pilot	10
Negative Pilot	11

produced between a cyclic-prefix OFDM radio and a primary user. The autocorrelation functions used in frame detection are inexpensive to calculate, as they are only evaluated at a time-lag of sixteen samples (the length of a short preamble symbol). The cross-correlation function required to find fine timing is expensive to calculate, and required the implementation of a simplified hardware design based on the quantization of the input complex signals to only one bit, and the use of an adder tree. Common DSP functions such as FFT, FIR filtering, and CORDIC were implemented using Xilinx Core Generator.

7.1.1 Allocation vector

It is assumed that the allocation vector reached both transmitter and receiver via a secondary channel, or via a higher-order protocol. The values of the allocation vector for each sub-carrier are presented in Table 7.1.

Positive and negative pilots are included for compatibility with the IEEE 802.11 PHY pilot allocations. The size of the allocation vector is $2N_{FFT}$ bits.

7.1.2 Frame Format

The IEEE 802.11 standard frame format is presented in 6.1. A modified version was adopted for the developed radios, and is presented in Figure 7.1. The short and long preambles are

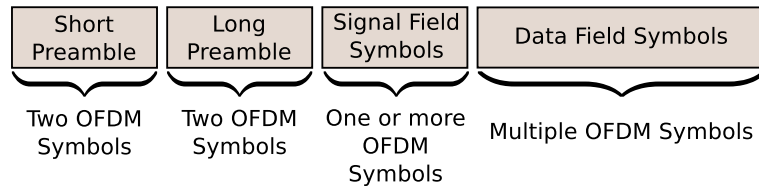


Figure 7.1: Frame format

based in the standard, but are qualified according to the allocation vector $\bar{V}$.

The signal field is always modulated in BPSK and contains information about the data rate, frame size, and modulation type. A standard OFDM symbol has 48 data sub-carriers; therefore, that is the maximum size of the signal field. If an arbitrary allocation vector is used, the number of symbols required to transmit the signal field is unknown. For this reason it has a length of one or more symbols.

Preambles

The preambles of the OFDM transmitter are chosen as a subset of the IEEE 802.11 standard preamble qualified with the allocation vector $\bar{V}$. Only the frequency bins corresponding to non-zero elements of the allocation vector are transmitted. In the standard, the short preamble is formed by two OFDM symbols in which only the sub-carriers with indexes multiple of four are active, producing a waveform with a period of one quarter of the useful symbol time. This is a limitation to achieving arbitrary allocation vectors, as the location of the active sub-carriers in the short preamble reduces the flexibility of the allocation vectors, as illustrated in Figure 7.2.

Two OFDM symbols plus their respective cyclic prefix (CP) contain ten periods of this waveform. Masking some of the sub-carriers with the allocation vector does not modify the periodicity of the waveform, making the masked preamble suitable for frame detection as long as two conditions are met: First, the higher layer protocol must guarantee that the allocation vector does not null out all the non-zero sub-carriers of the short preamble.

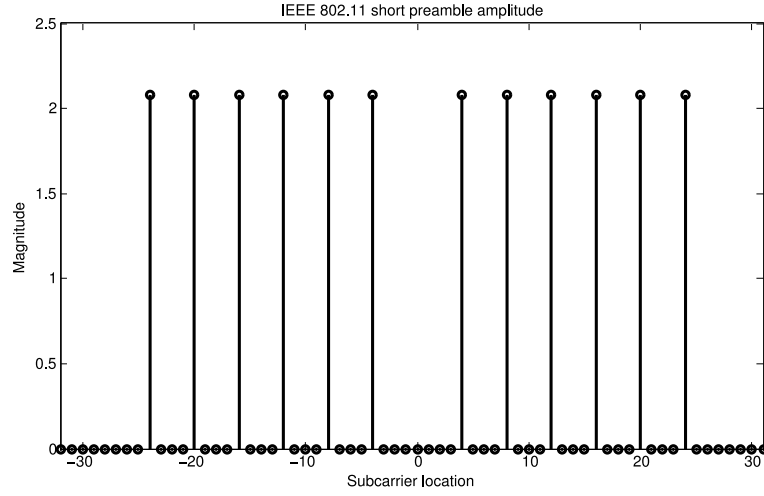


Figure 7.2: Amplitude of the short preamble in the frequency domain

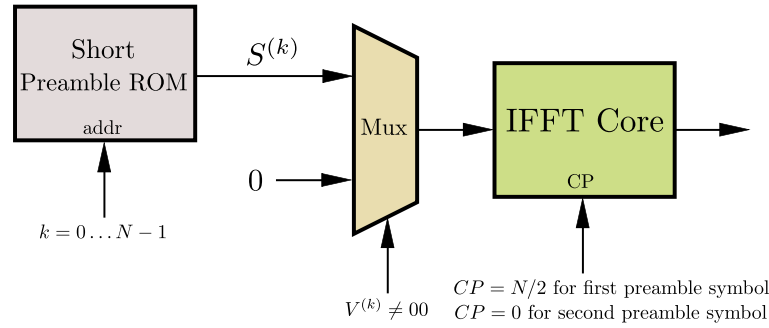


Figure 7.3: Block diagram for generation of the short preamble

Second, the preamble power can be scaled to account for the power loss caused by the masked sub-carriers. The short preamble is used in the receive-side for frame detection and for carrier frequency offset (CFO) estimation. A diagram of the generation of the short preamble is presented in Figure 7.3

The long preamble consists of a pseudo-random sequence in the frequency domain that has periodicity of one OFDM symbol in the time domain. The long preamble cyclic prefix is generated in a different way: the first long preamble symbol is prefixed by a 50% repetition, while the second long preamble symbol does not have a cyclic prefix. In this way, the set of two long preamble symbols has the duration of two standard symbols with 25% CP each. The advantage of this arrangement will become apparent in the design of the receiver. The

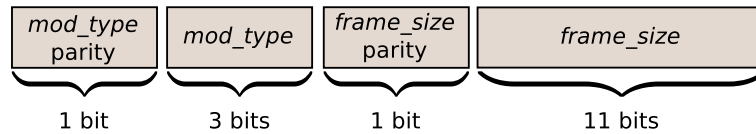


Figure 7.4: Signal field format

generation of the masked long preamble is performed in the same way as the short preamble. The long preamble is used for fine timing and for channel response estimation.

Signal field

In standard OFDM the first symbol, known as the signal field, is used to transmit the modulation type, forward error correction (FEC) rate, and length of the frame. When using arbitrary allocation vectors, the transmission of the signal field may require more than one OFDM symbol, which requires the transmitter logic to count the number of bits that are being sent at each OFDM symbol to determine when the signal field has been transmitted, and pad the remainder of the last signal field symbol with pseudo-random generated tail bits to avoid a high peak to average power ratio (PAPR). The format of the signal field is presented in Figure 7.4. The *mod_type* field indicates the modulation type by the number of bits per constellation coordinate, i.e.: 1 for BPSK, 2 for QPSK, and 4 for 16-QAM. This field is protected by a parity bit, as well as the *frame_size* field, which carries the size of the frame in sixteen-bit words. No FEC is implemented at this time. The signal field is non-scrambled and is BPSK modulated.

7.1.3 The Harris System-in-Package

The platform chosen for the hardware implementation is the Harris system-in-package composed of four Virtex-4 FPGAs for signal processing operations, a switch FPGA, a Digital to Analog Converter (DAC), Analog to Digital Converter (ADC) and a TI daVinci SiP running Linux in its ARM processor. The system contains inter-FPGA connections, and 16-bit wide

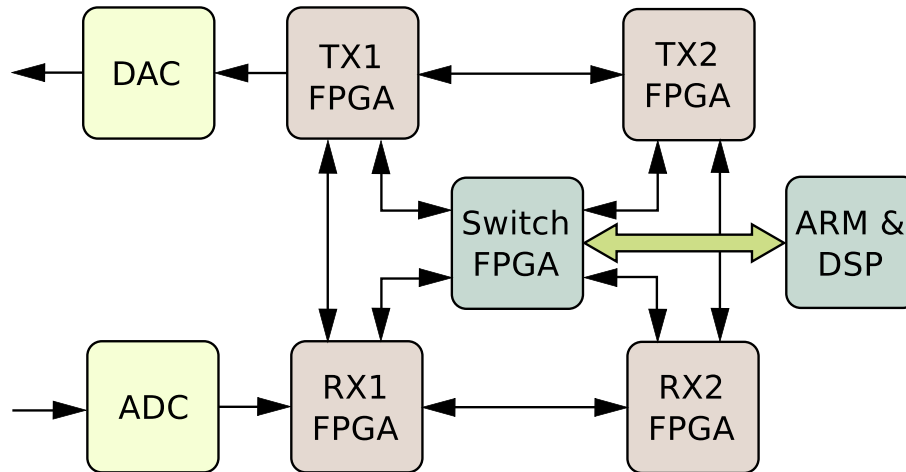


Figure 7.5: The Harris SiP

CAD bus (control, address, and data bus) connections to the ARM processor via the switch FPGA, that maps the four Virtex-4 FPGAs to memory addresses on the ARM. Figure 7.5 presents a simplified diagram of the Harris SiP.

7.1.4 Transmitter Architecture

The transmitter architecture is based on the FFT/IFFT core from Xilinx Core Generator, and is presented in Figure 7.6. The cyclic prefix feature available in the newer versions of the tools is used for frame-to-frame configuration of the cyclic prefix length. The frame FIFO receives the frames sent by the MAC layer, and a counter of frames in the FIFO is used to signal the frame state machine that there are new frames to transmit. The allocation vector is fed to the state machine to manage the formation of the OFDM symbols in the frequency domain, which are then passed to the IFFT core. The burst output of the IFFT core is passed to a FIFO and re-timed to form in-phase and quadrature streams at a sampling rate of 20 MHz.

The I and Q streams are up sampled by a factor of three before being passed to the DAC at a sampling rate of 60 MHz. The DAC up samples the signal by a factor of four, and

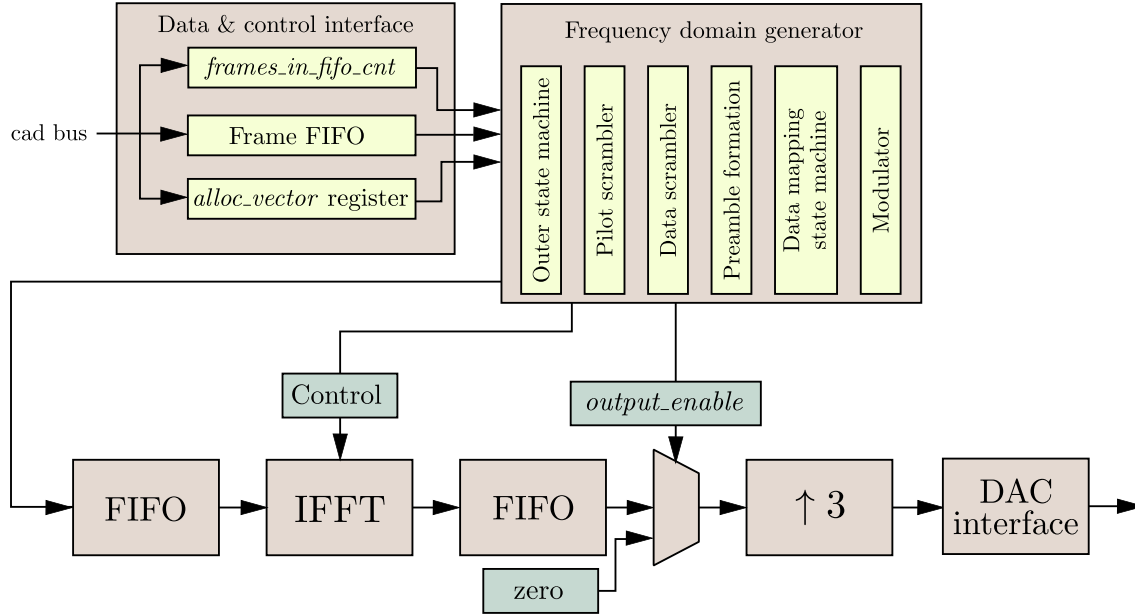


Figure 7.6: Transmitter architecture

generates an output modulated at a 60 MHz intermediate frequency (IF). This IF signal is up converted to the RF band of 2.05 GHz and transmitted over-the-air using off-the-shelf components.

Clock generation and domains

Figure 7.7 shows the clock generation network and clock domains in the TX1 FPGA. The CAD bus (the CPU-to-FPGA interface) provides a clock at 48 MHz that is used for interfacing data to and from the switch FPGA. The Digital-to-Analog Converter (DAC) is set to receive a clock from an external generator. Quadrature modulation is only available when the DAC clock is equal to four times the data clock. Therefore, these values are set to 240 MHz and 60 MHz respectively. This arrangement produces an intermediate frequency signal at 60 MHz. In the quadrature modulation mode, the DAC output I outputs $I(t) \cos(\omega_c t) \pm Q(t) \sin(\omega_c t)$ while the signal at the DAC output Q corresponds to $I(t) \cos(\omega_c t) \mp Q(t) \sin(\omega_c t)$. The sign of the quadrature component is configurable via a SPI

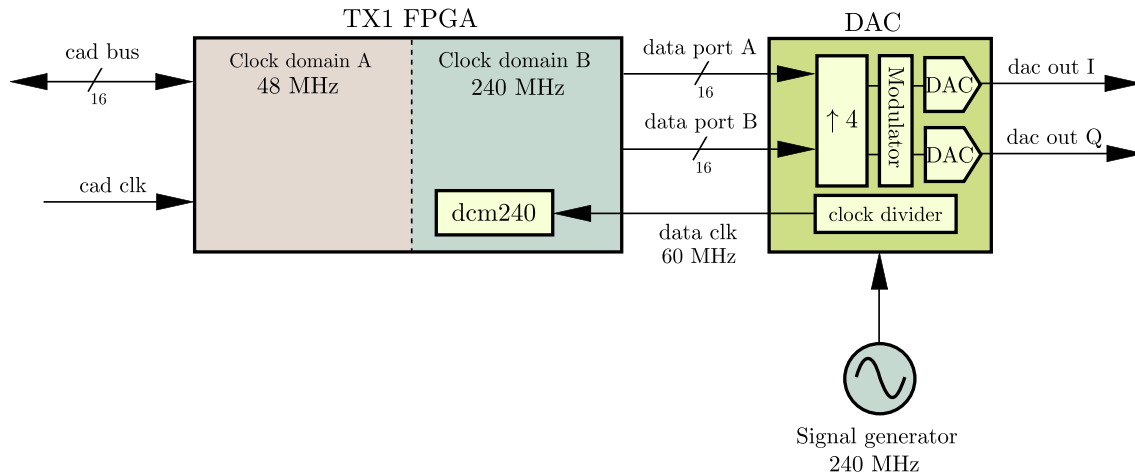


Figure 7.7: Clock sources of the TX1 FPGA

interface.

A digital clock manager (DCM) synthesizes back a clock of 240 MHz from the 60 MHz signal coming from the DAC. This clock is used for the signal processing operations of the transmitter. The next section provides details on the implementation of the clock domain crossing from clock domain A to clock domain B.

Data and control interface

There are 16 memory addresses available in the TX1 FPGA, that are used for the purposes detailed in Table 7.2. A frame FIFO was implemented using a simple FIFO and control logic to buffer entire frames and set a control signal to the subsequent framer logic whenever there are one or more full frames in the FIFO ready for transmission.

The allocation vector is also passed through the CAD bus to the internal logic.

Transmitter outer state machine

Whenever *frames_in_fifo* is greater than one, the state machine transitions to the *TX_SHORT1* state, where a short preamble symbol with a cyclic prefix of 50% is transmitted. *TX_SHORT2*

Table 7.2: CAD bus address map for TX1 FPGA

Address	Write mapping	Read mapping
0x00	Start to write allocation vector	Start to read allocation vector
0x01	Write allocation vector	Read allocation vector
0x02	End write allocation vector	End read allocation vector
0x03	Data Write	FIFO write count
0x04	End Data Write	Not used

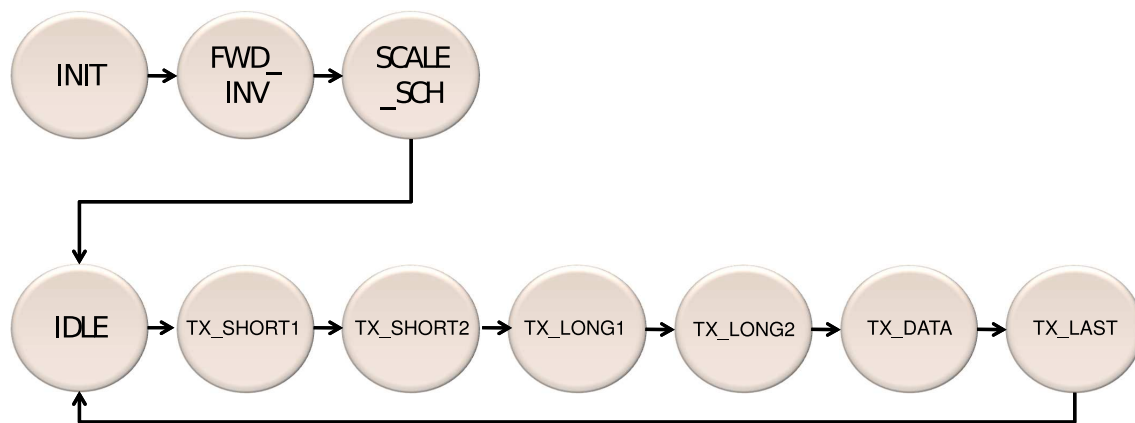


Figure 7.8: OFDM transmitter outer state machine

forms a second short symbol with CP of 0%. Similarly, *TX_LONG1* and *TX_LONG2* perform the transmission of the long preamble symbols with CP of 50% and 0% respectively, fulfilling in this way the requirements of the IEEE 802.11 standard. The FFT core permits the configuration of the CP length at the same time that the *ifft_start* signal is applied. The frequency domain descriptions of the short and long preambles are stored in ROM and fetched by the FFT core.

After the preamble is transmitted, the signal field is transmitted using the *TX_DATA* state. A proprietary signal field, shown in Figure 7.4, is proposed for the design; however, it can be easily modified to match the standard. The data flow of the signal field is the same as in the *TX_DATA* field, with three exceptions:

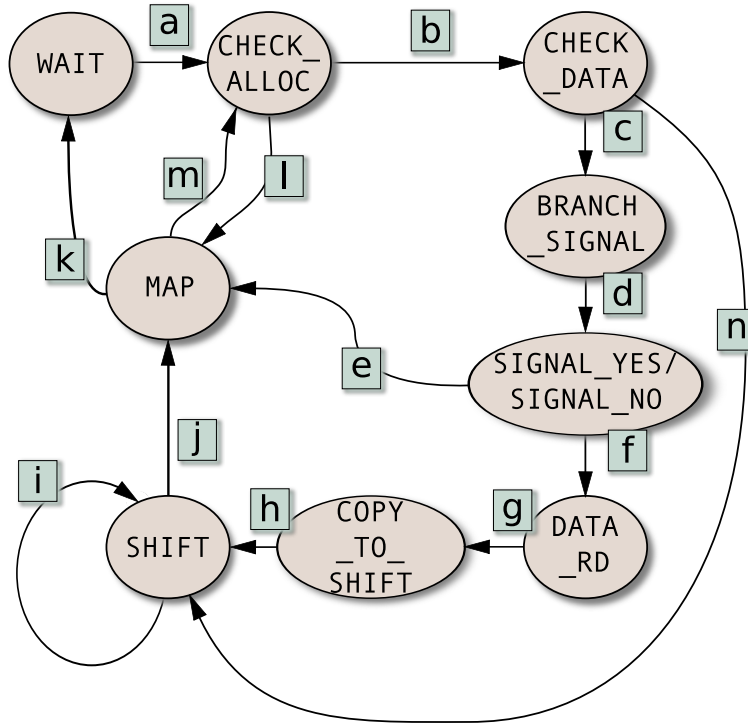


Figure 7.9: Data mapping state machine

- The signal field is always transmitted in BPSK.
- Any remaining sub-carriers in the current symbol are zero-filled.
- The signal field is not scrambled.

Data mapping state machine

The transmitter states are driven by the allocation vector, the modulation type, and the frame size. The number of bits per OFDM symbol is not constant, and a given input word may get split between two consecutive symbols. Control logic is required to keep track of the number of bits that is being copied to the OFDM symbol, so a new read from the frame FIFO occurs when an input word boundary is crossed. These requirements are fulfilled by the state machine presented in Figure 7.9.

Every time a data sub-carrier is allocated, as determined at the *CHECK_ALLOC* state, the

state machine checks if the modulation shift register is empty at the *CHECK_DATA* state. If so, it reads a new word from the FIFO at the *DATA_RD* state. Then, the bits are scrambled and shifted into the *mod_reg* register in the amount required by the modulation type. The modulation type field controls the number of counts performed at the *SHIFT* state. Once the *mod_reg* is formed, the *MAP* state passes it to the modulation mapper, which maps it to a constellation coordinate in the I-Q plane according to the modulation type selected.

If the allocated sub-carrier is of the pilot or null types, the state machine jumps directly to the *MAP* state where flags are set to make the modulation mapper generate a BPSK-mapped pilot or a zero value whenever a null sub-carrier is transmitted.

The *BRANCH_SIGNAL* state was introduced to control the symbol padding at the end of both the signal and the data fields.

At the *MAP* state, the allocation address is checked for the ending of an OFDM symbol, and the symbol count is checked against the *frame_size*. The state machine returns to the *WAIT* state until the flag to form a new symbol is set.

The state machine can be clearly understood by its transition conditions, which are presented next:

- a. A new symbol is to be formed.
- b. The allocation vector entry indicates that the current sub-carrier contains data.
- c. The shift register is determined to be empty.
- d. Determine if the current data belongs to the *SIGNAL* field or to the *DATA* field. These fields have different zero padding conditions after their specific lengths are reached, and is evaluated in the states *SIGNAL_YES* and *SIGNAL_NO*. These states are drawn together for simplicity.
- e. If pseudo-random sequence filling is required, jump to *MAP* and set the filling flag.
- f. If no filling needs to be performed and the shift register is empty, read new data from the

FIFO.

- g. After one cycle wait for the FIFO output, copy the data to the shift register.
- h. At the shift state, scramble the data and shift it into the modulation register *mod_reg*.
- i. Stay at *SHIFT* as long as required: 1 clock cycle for BPSK, 2 clock cycles for QPSK, 4 four cycles for 16-QAM.
- j. After shifting, jump to *MAP*. This state triggers the modulation mapping module to perform the mapping of the data, pilots, and null sub-carriers to their corresponding constellation points.
- k. After *MAP*, if the allocation address equals N_{FFT} , the OFDM symbol is formed. Transition to *WAIT*.
- l. When the sub-carrier allocation is null or pilot, go straight to the *MAP* state.
- m. After *MAP*, if the allocation address is less than N_{FFT} , jump to *CHECK_ALLOC* on the next sub-carrier.
- n. At *CHECK_DATA*, if the shift vector is not empty, no FIFO reads or symbol filling are required. Jump directly to *SHIFT*.

State machine verification

A ChipScope snapshot of operation of the data mapping state machine is presented in Figure 7.10 during the transmission of a QPSK modulated symbol, illustrating the allocation vector analysis process and modulation register forming. In the figure, follow the *alloc_addr* and the *sc_alloc* signals:

- At *alloc_addr* = 0, the DC sub-carrier (null) is transmitted (*sc_alloc* = 0). The *ena* signal passes the results of this stage to the modulation mapper module *mod_map*.
- Addresses (sub-carriers) 1 to 6 allocate data sub-carriers *sc_alloc* = 1. At address 1,

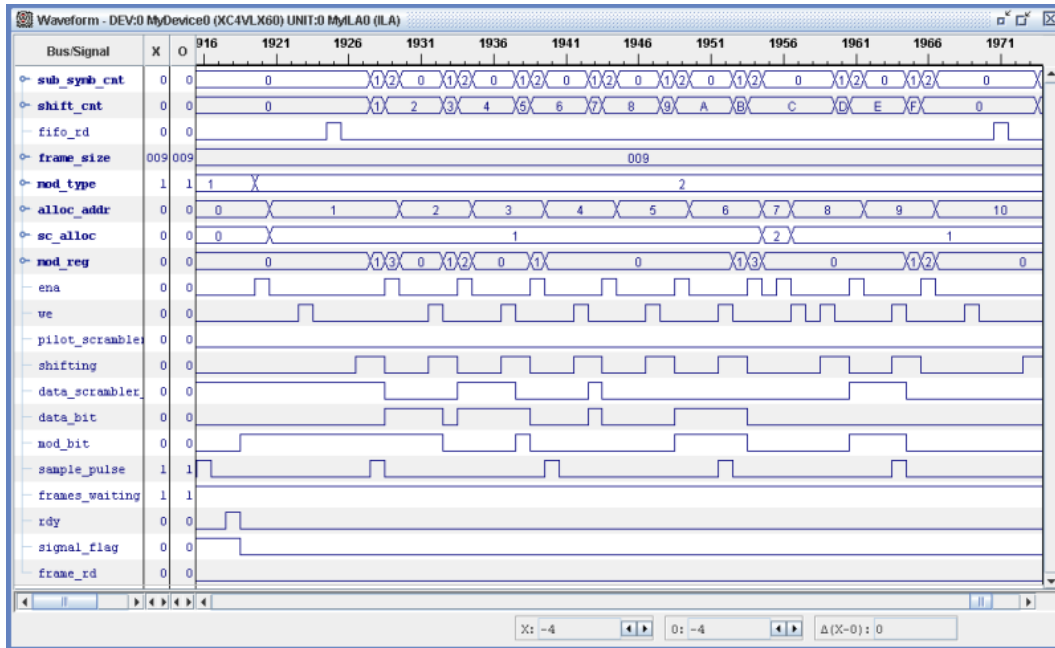


Figure 7.10: Transmitter state machine verification

the FIFO is read. Then the shifting signal stays active for two clock cycles to shift two scrambled bits into *mod_reg*. The result is passed to modulation mapping at *ena*.

- Address 7 corresponds to a positive pilot (a value of 2). There is no shifting and the *pilot_scrambler* bit is passed directly to the modulation mapper.
- On the top, *sub_symb_cnt* is controlling the shifting signal (counting up to two for QPSK), and *shift_cnt* is controlling the feeding of the shift register with new data from the CAD bus FIFO.
- The symbol shown is the first one after the *SIGNAL* field. The transition of *mod_type* from BPSK (one) to QPSK (two) can be also be observed.

Transmit interpolation filter

A common approach for the interpolation stage that bridges the sampling rate of the base band processing stage with that of the DAC is the use of a cascade integrator comb (CIC) fil-

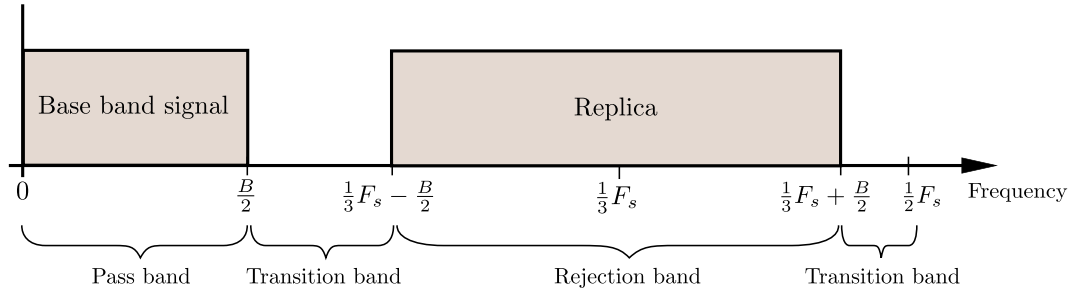


Figure 7.11: Filter requirements for interpolation by three

ter. This type of filter implements a simple moving average operation (boxcar) that does not require the use of multipliers, and is used extensively, for example, in the FPGA processing stage of the popular USRP and USRP-2 [136].

In the case of an IEEE 802.11 compliant PHY, the maximum number of active sub-carriers is limited to 52 (accounting for 48 data and 4 pilot sub-carriers) the output bandwidth for the signal can be calculated as $B = 20 \times (52 + 2)/64 = 16.875$ MHz, where the two additional values account for the DC sub-carrier and for the bandwidth of the edge sub-carriers, assuming that most of the energy is contained in their main lobe.

In order to successfully implement up sampling, the spectral replicas produced when the signal is zero padded must be rejected, as presented in Figure 7.11. For illustration purposes, a CIC filter designed with three stages for an up sampling rate of 3 is shown in Figure 7.12. Clearly, the CIC filter does not qualify for interpolation of a wide band signal.

To fulfill the requirements of the interpolation filter, a FIR filter of order 95 is designed using the Parks-McClellan optimal equiripple algorithm for an attenuation of 90 dB in the rejection band. The resulting frequency-domain response is presented in Figure 7.13.

Modular structure

The organization and dependencies of the modules used in the transmitter are presented in Figure 7.14.

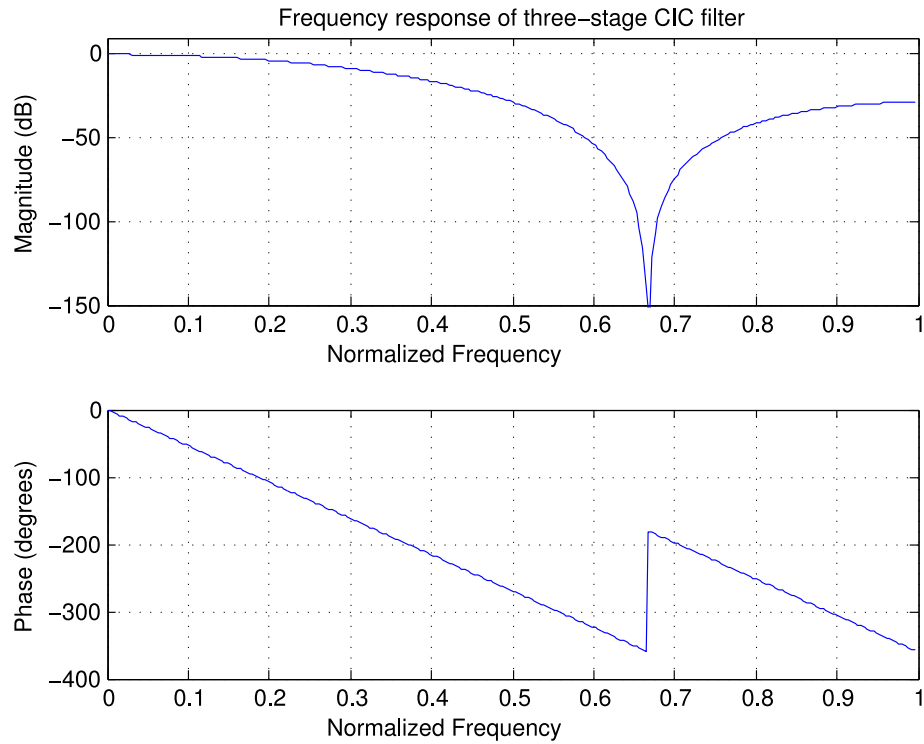


Figure 7.12: Three stage CIC filter response

7.1.5 Receiver Architecture

The receiver architecture is presented in Figure 7.15. The receiver uses the preambles for frame detection and for estimation of the CFO, fine timing, and channel response. It is assumed that the channel response remains constant for the duration of a frame. The equalization and phase tracking functions are accomplished entirely in the frequency domain to reduce the loop latency.

Sampling and filtering front-end

The first stage, previous to the baseband processing presented in Figure 7.15, is formed by the ADC, a quarter sampling frequency down-converter, and a low-pass decimating filter required to obtain the complex base band representation of the signal.

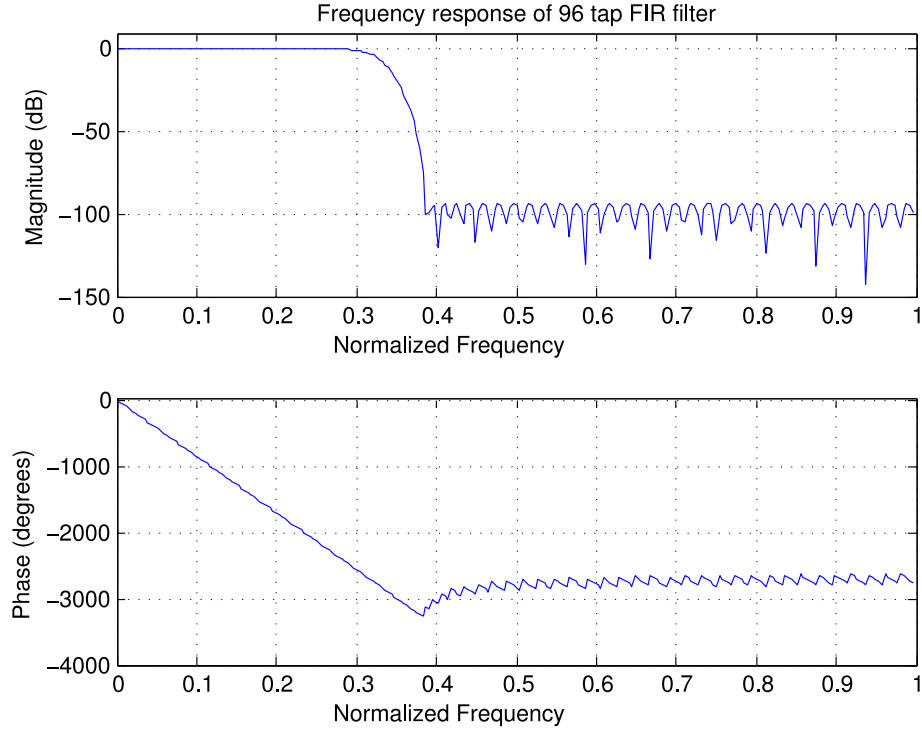


Figure 7.13: Order 95 interpolating FIR filter response

The ADC sampling frequency is selected to produce an aliased copy of the received carrier at one quarter of the sampling frequency [65]. To calculate the aliased frequencies, the following equation must be fulfilled for integer values of k , under the constraint that the signal bandwidth B plus the transition band B_T must be less than half the sampling frequency F_s , as presented in Equation 7.1.

$$f_c = \left(k \pm \frac{1}{4}\right) F_s \quad \text{subject to} \quad F_s/2 > (B + B_T) \quad (7.1)$$

where f_c is the carrier frequency.

Under these conditions, the set of available sampling frequencies for $f_c = 60$ MHz is presented in Table 7.3.

Considering that the maximum sampling frequency of the ADC is 125 MHz, and that the processing sampling rate for IEEE 802.11 is 20 MHz, the best choice for the sampling fre-

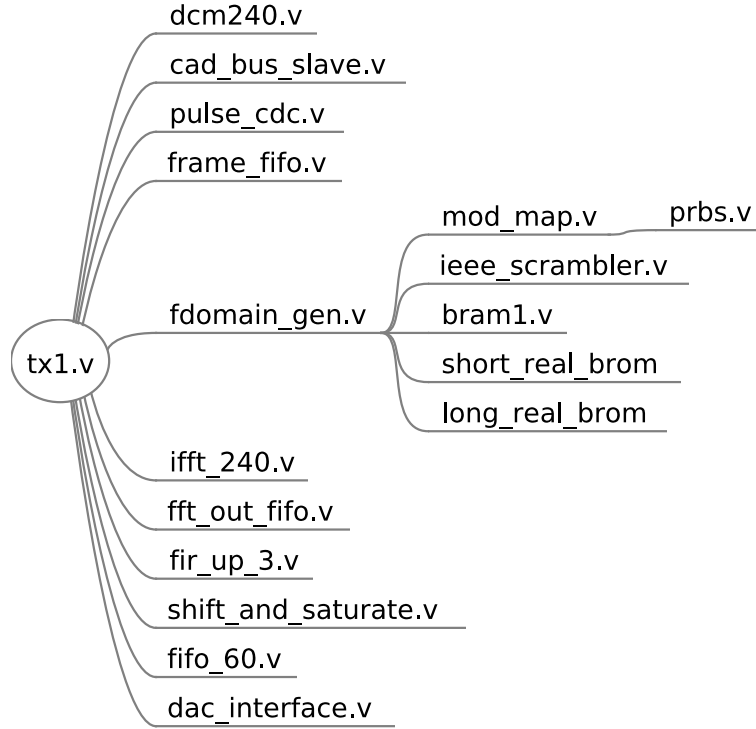


Figure 7.14: Module dependency of the OFDM transmitter implementation

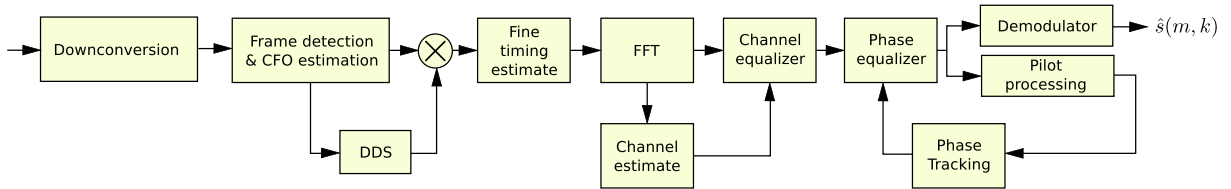


Figure 7.15: Receiver architecture

Table 7.3: Bandpass sampling frequencies

k	Positive alias	Negative alias
0	240 MHz	240 MHz
1	48 MHz	80 MHz

quency is 80 MHz. At this frequency, the complex down converter is simplified by the fact that the multiplication with the cosine ($\{1, 0, -1, 0\}$) and the sine ($\{0, 1, 0, -1\}$) signals consists only of a nulling operation and sign inversions. Further reductions can be obtained by

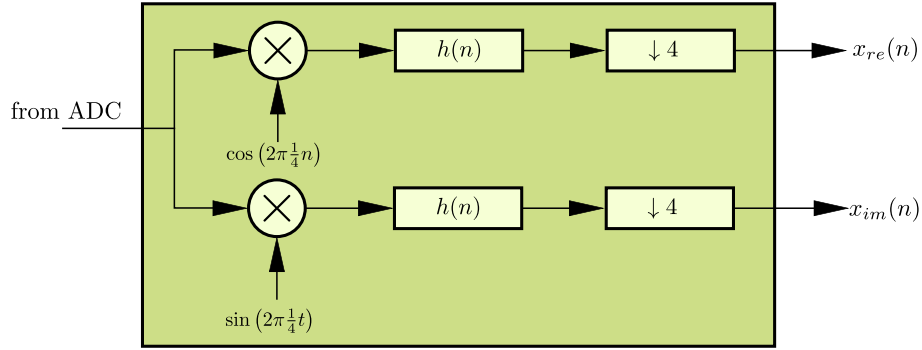


Figure 7.16: Conceptual down conversion filter

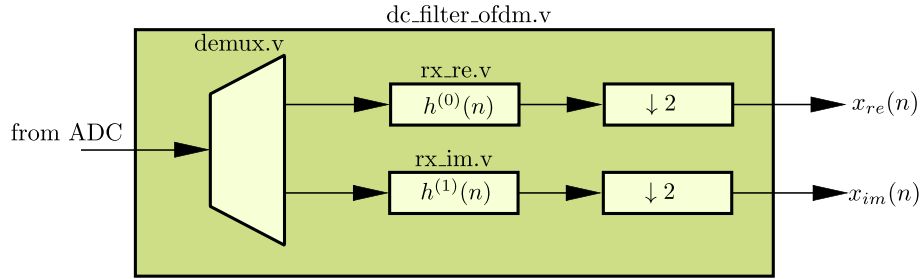


Figure 7.17: Efficient down conversion filter

considering the zero factors that can be accounted for as a down sampling by a factor of two, and by pushing the signs into the polyphase decomposition of the low-pass filter $h(n)$, as presented in Equations 7.2 and 7.3.

$$h^{(0)}(n) = h(2n)(-1)^n \quad (7.2)$$

$$h^{(1)}(n) = h(2n+1)(-1)^n \quad (7.3)$$

The conceptual implementation of the down conversion stage is presented in Figure 7.16. The resulting hardware implementation of the aliasing digital down converter is shown in Figure 7.17.

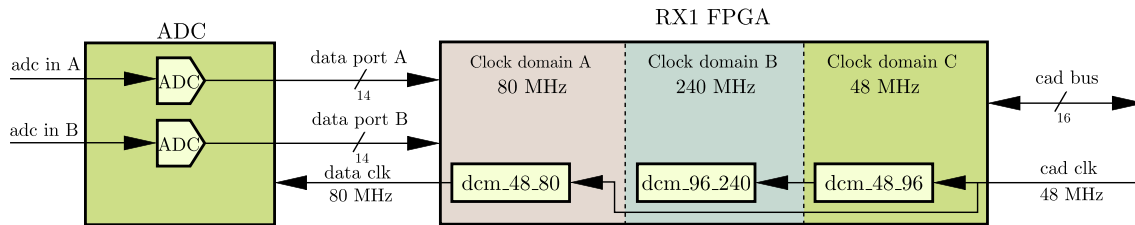


Figure 7.18: Clock sources of the RX1 FPGA on the Harris SiP 1

Clock generation and domains

The clock generation network at the receivers is different depending on the model of the Harris SiP. In the older model, the clocks are derived from the *cad clk* signal, that runs at 48 MHz. Three digital clock managers are required to obtain an 80 MHz sampling clock, and a 240 MHz processing clock. Three clock domains are required, as shown in Figure 7.18.

In the newer model, an external reference is required to clock the Analog-to-Digital converter through a clock manager device. The clock generator was set at 480 MHz to provide compatibility with the sampling of 96 MHz required by an existing radio (by setting the divider to a factor of five instead of six). The clock domain diagram is presented in Figure 7.19.

Data and control interface

The address map for the data and control interface designed for the FMT transmitter is presented in Table 7.4. The spectrum sensing interface was fully developed for the FMT radio, and can be easily extended for the OFDM case. Addresses 0x00, 0x01, and 0x02 are used for the allocation vector write protocol. It can be read back for verification purposes. Address 0x03 reads the dropped frames counter that is increased if the receiver receives a new frame from the air and the FIFO does not have enough space for it. Address 0x04 is used to read the received data from the FIFO, while address 0x05 signals the frame FIFO that the frame was fully read. The number of frames in FIFO can be fetched reading from address 0x06. Reading from address 0x07 resets the receive FIFO. Addresses 0x08 and 0x09 hold

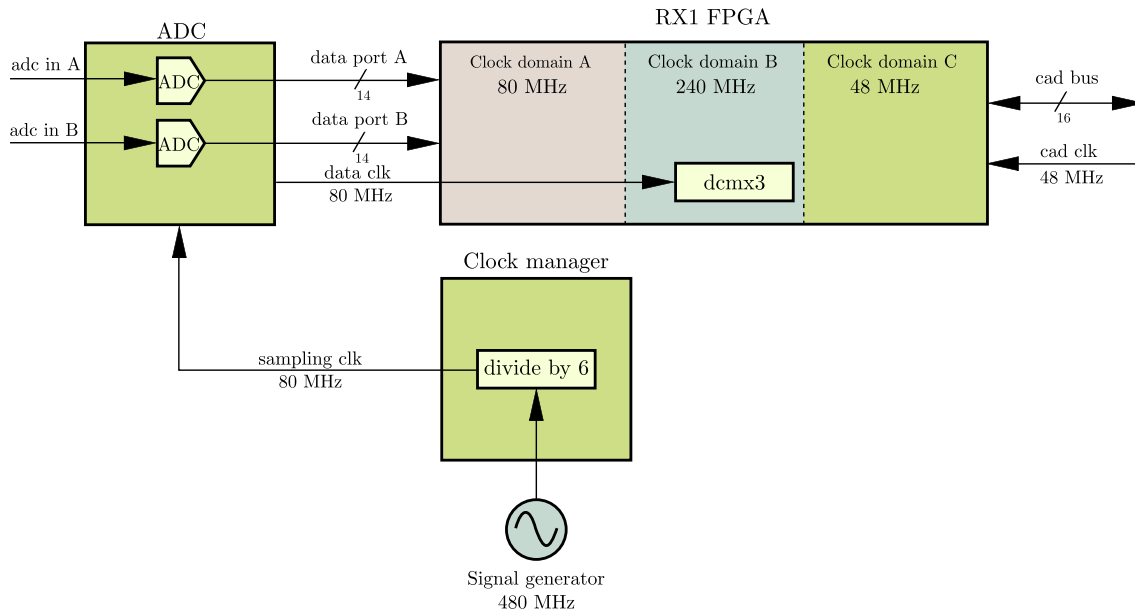


Figure 7.19: Clock sources of the RX1 FPGA on the Harris SiP 2

the parity error counters for the modulation type and frame size fields respectively. Besides the data FIFO, two additional FIFOs are included in the receiver: the monitor FIFO (at address 0x0B) permits the verification of the measured channel response, and the I-Q values of the FFT output for constellation plotting purposes, while the sensing FIFO (0x0D) holds the measured power spectrum.

Frame detection and CFO estimation

The period of the short preamble is preserved even when a subset of the full short preamble sub-carriers is transmitted. This allows the use of the autocorrelation magnitude estimate for frame detection. Rejection of the interference caused by the primary user is not accounted for at this stage of the project development. The autocorrelation function of the complex baseband signal is calculated using the topology of Figure 7.20.

The test for detection of a frame depends on the running estimation of power, that uses the topology presented in Figure 7.20 except for the 16 sample delay. In the actual implemen-

Table 7.4: CAD bus address map for RX1 FPGA

Address	Write mapping	Read mapping
0x00	Allocation vector start write	Allocation vector start read
0x01	Allocation vector write	Allocation vector read
0x02	Allocation vector end write	Allocation vector end read
0x03	Not used	Dropped frames counter read
0x04	Not used	Data Read
0x05	Not used	End read
0x06	Not used	Frames in FIFO
0x07	Not used	Reset FIFO
0x08	Not used	Modulation type parity error counter
0x09	Not used	Frame Size parity error counter
0x0A	Not used	Monitor FIFO check
0x0B	Not used	Monitor FIFO data
0x0C	Not used	Sensing FIFO check
0x0D	Not used	Sensing FIFO data

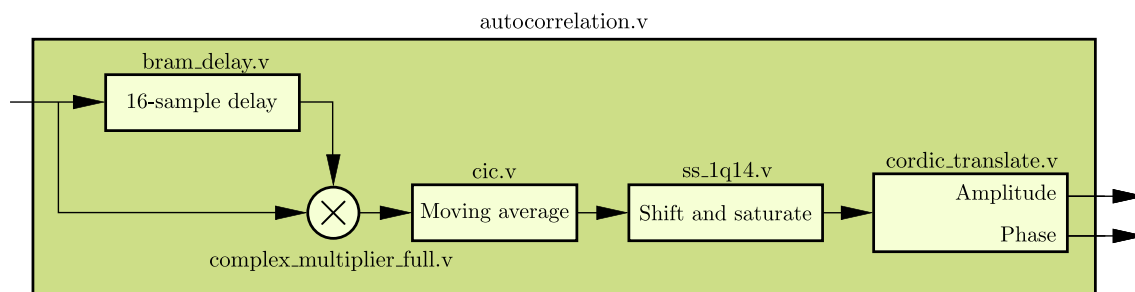


Figure 7.20: Implementation of the autocorrelation function

tation, a set of successive positive tests is required to filter out autocorrelation peaks arising from the autocorrelation of random noise.

The *cordic_translate* module is re-used for CFO estimation from a second autocorrelation

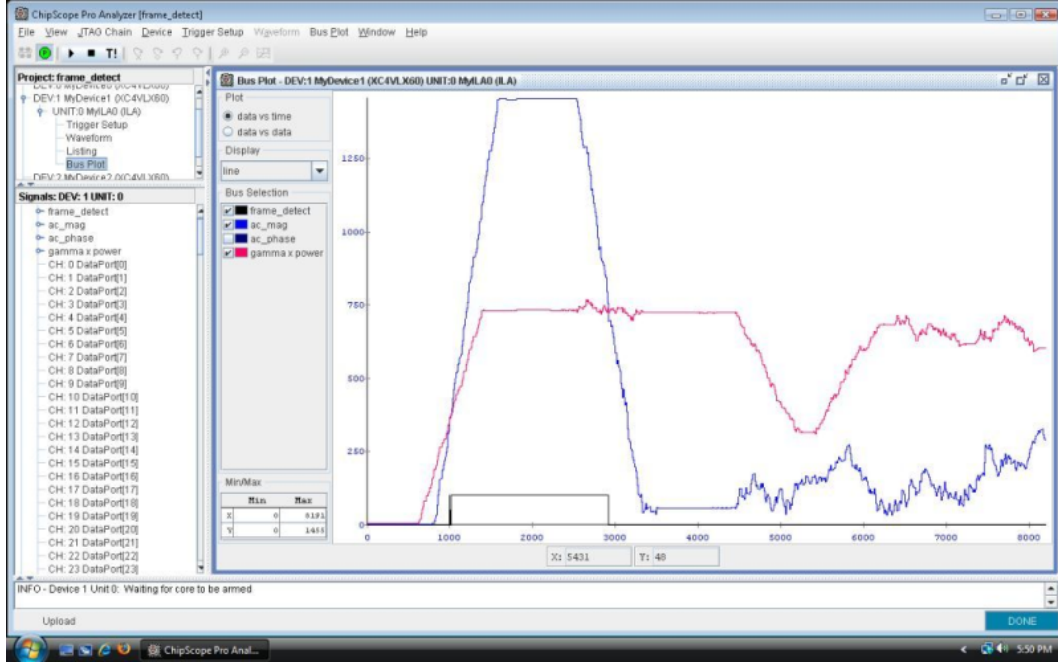


Figure 7.21: Frame detection verification

function calculated over a longer integration time of eight short symbols ($2N_{FFT}$), as proposed by Schwoerer [135]. The longer integration time is achieved by a different parameterization of the *cic* module in the autocorrelation diagram. The CFO estimate is obtained by reading the phase output of the cross-correlation module at the established time. Figure 7.21 presents a ChipScope capture of the signals involved in frame detection.

Derotation

The long-autocorrelation phase $\hat{\psi}$ is obtained using a time lag of $N_{FFT}/4$, which is a power of two. Therefore, the CFO estimate $\hat{\nu}$ can be easily obtained using a shift to the right, as presented in (7.4).

$$\hat{\nu} = \hat{\psi} \gg \log_2(N_{FFT}/4) \quad (7.4)$$

where the $>>$ operator represents a sign-extended shift to the right. A phase accumulator followed by a CORDIC rotation module forms the digital direct synthesizer (DDS) that produces a complex exponential. The incoming signal is derotated by multiplication with the complex conjugate of the DDS output.

Fine timing estimation

The fine timing estimate is obtained using the approach presented by Schwoerer [135]. The magnitude of the cross-correlation function between the derotated signal and a previously stored copy of a half-swapped long preamble symbol peaks when the signals are aligned.

To perform the cross-correlation function, N_{FFT} complex products and $N_{FFT} - 1$ complex additions are required at each input sampling instant. A simplified complex product is obtained by multiplying only the sign bits of the complex input signals, and adding the results. An adder tree over the resulting N_{FFT} real and imaginary multiplication results is implemented to perform the time domain integration efficiently, as presented in Figure 7.22. The *cordic_translate* module is used to obtain the magnitude of the resulting complex values. The peak of the magnitude of the cross-correlation function is used to synchronize the FFT observation window.

Denote the i th product in the cross-correlation calculation as $x_i y_i^*$, where $x_i = a_i + j b_i$ and $y_i = c_i + j d_i$. The complex product is calculated as:

$$q_i = \frac{(a_i c_i + b_i d_i) + j(b_i c_i - a_i d_i)}{2} \quad (7.5)$$

where the denominator represents the scaling of the simplified complex addition and subtraction. If the two's complement input signals are quantized to one bit, the sign bit can be used to represent the signal value, as presented in Table 7.5.

Using the sign bit representation, the one bit products can be calculated using the XOR function:

Table 7.5: Sign bit mapping

Sign bit	Quantized value
0	1
1	-1

$$a_i c_i = a_i \oplus c_i \quad (7.6)$$

$$b_i d_i = b_i \oplus d_i$$

$$b_i c_i = b_i \oplus c_i$$

$$a_i d_i = a_i \oplus d_i$$

where the $\oplus$ symbol denotes the logic XOR.

The result of the addition and subtraction operations can be obtained according to:

$$\begin{aligned} \frac{a_i c_i + b_i d_i}{2} &= \{a_i c_i \wedge b_i d_i, \neg(a_i c_i \oplus b_i d_i)\} \\ \frac{b_i c_i + a_i d_i}{2} &= \{b_i c_i \wedge \neg a_i d_i, b_i c_i \oplus a_i d_i\} \end{aligned} \quad (7.7)$$

where α, β represents a two's complement number formed by the sign bit α and the value β .

Two adder trees are required, one for the real values and other for the imaginary. A sketch of the adder three structure for adding 64 values is presented in Figure 7.22.

Figure 7.23 presents a ChipScope capture of the sliding cross-correlation function calculated during the long preamble symbol. Observe the peak generated and the behavior of the counter *wait_cnt* used to measure time offset.

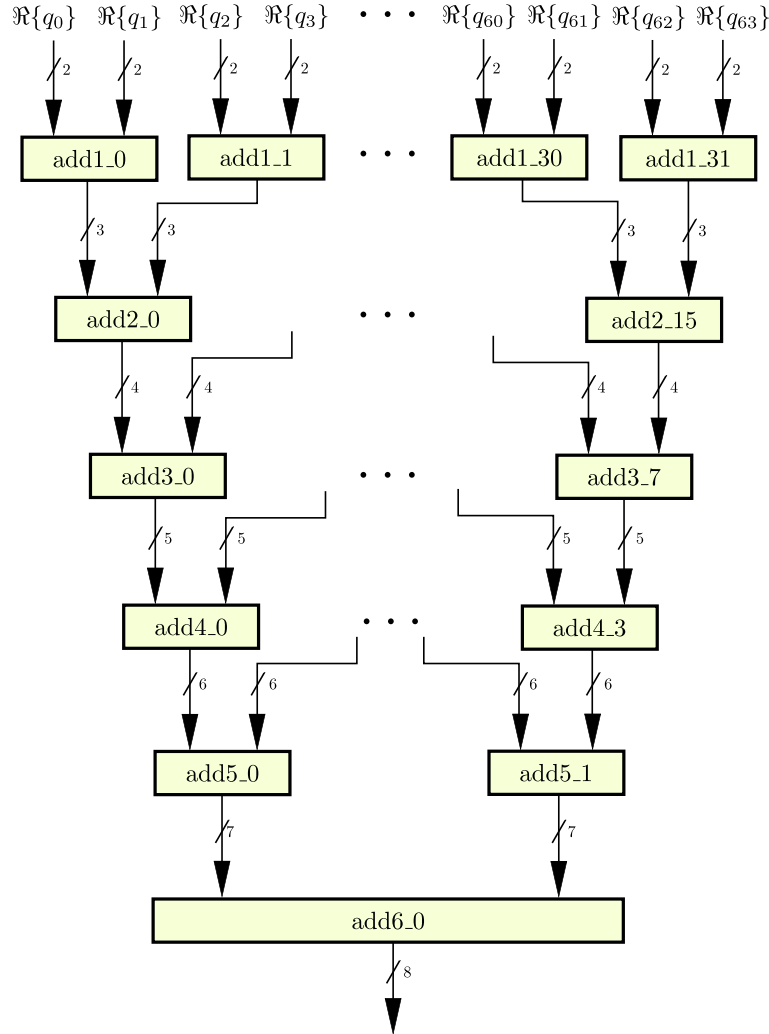


Figure 7.22: Implementation of the adder tree

Channel estimation and equalization

The long preamble is also used for channel estimation. An FFT is calculated over the second long preamble symbol, which employs all the active sub-carriers and can be considered a full pilot symbol. The resulting set of values in the frequency domain is sign-corrected according to the signs of the transmitted sub-carriers, giving the channel estimate. The estimate is stored in RAM and retrieved at every arriving OFDM symbol, assuming that the channel is stationary during the duration of a frame. Matched filtering is implemented using a complex

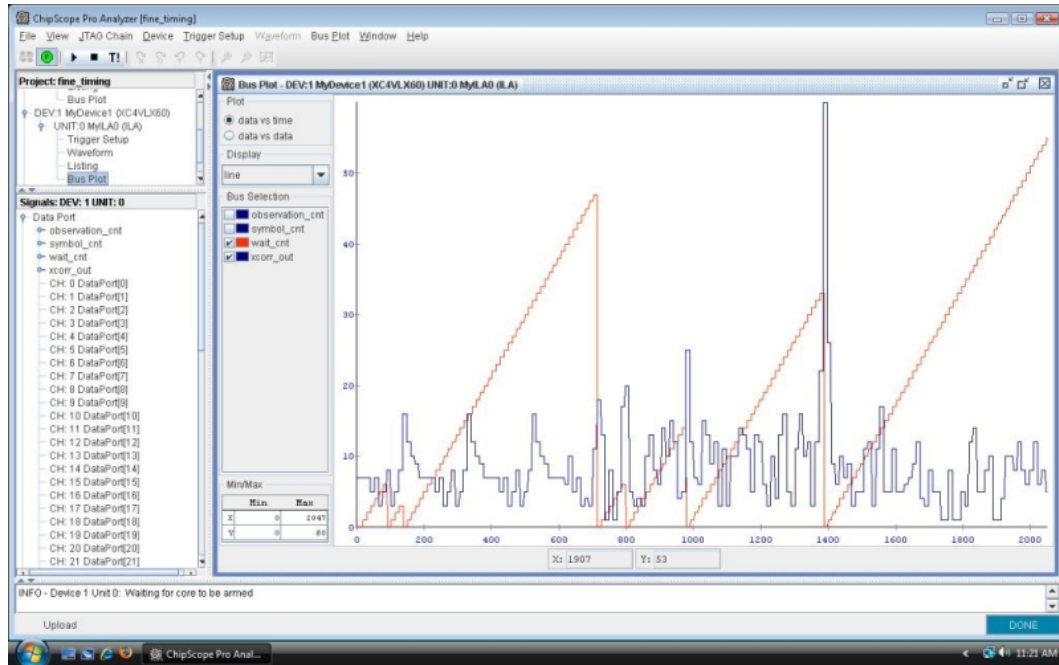


Figure 7.23: Fine timing estimation

multiplier as a phase equalizer for the input samples.

Synchronization Tracking

This section presents a method for synchronization tracking applicable to arbitrary pilot sub-carrier allocations that are parameterized by the allocation vector.

Cyclic prefix OFDM possesses the interesting property that the OFDM window sampling instant error can be absorbed into the channel estimate as long as the FFT observation window does not contain inter-symbol interference (ISI) by crossing the next symbol boundary or by receiving interference from the previous symbol due to the duration of the channel impulse response. Under these conditions, over the frame duration, two sources of phase errors affect the receiver: sampling frequency error, and CFO residual. The sampling frequency error accumulates as a shift in the time-domain observation window, which maps to a linear phase term in the frequency domain. The CFO residual, even if small enough to not produce

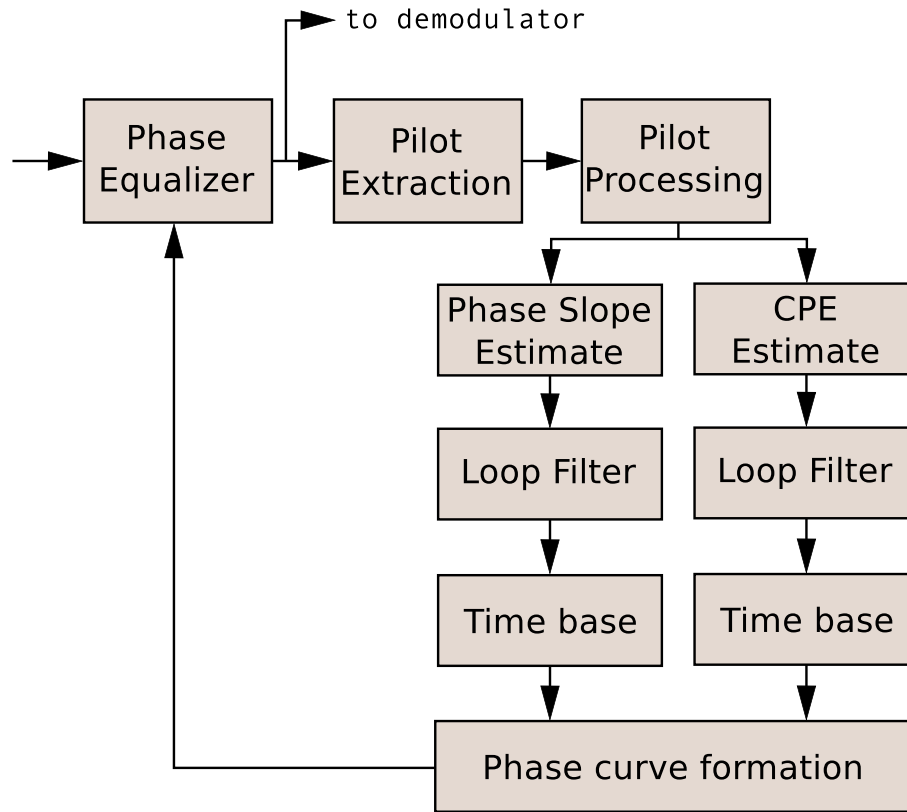


Figure 7.24: Phase tracking subsystem

inter-carrier interference (ICI), produces an accumulating common phase error (CPE) for all the sub-carriers.

Two sources of phase errors affect the receiver: sampling frequency error, and CFO residual. The sampling frequency error accumulates as a shift in the time-domain observation window that maps to a linear phase term in the frequency domain. The CFO residual, even if small enough to not produce inter-carrier interference (ICI), produces an accumulating common phase error (CPE) for all the sub-carriers. Two control loops are required to perform the tracking of the sub-carrier phases, as presented in Figure 7.24. The error signals that feed the control loops are obtained from the measured phase of the pilot sub-carriers.

Support for arbitrary allocation vectors requires the capability to extrapolate an estimate of the phase response over the active sub-carriers, using phase measurements from a set of

arbitrary locations of pilot sub-carriers. To achieve this goal, a least-squares estimate of the CPE and phase slope is calculated for every symbol. The system of equations presented in Equation 7.8 is formed using M pilot location indexes and their corresponding measured phase values. The pilot locations and signs is extracted from the allocation vector. The cognitive engine is responsible for the correct allocation of pilot sub-carriers.

$$\begin{bmatrix} 1 & p_1 \\ 1 & p_2 \\ \vdots & \vdots \\ 1 & p_M \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} \theta_1 \\ \theta_2 \\ \vdots \\ \theta_M \end{bmatrix} \quad (7.8)$$

where p_i is the sub-carrier index of the i^{th} pilot that takes values in the interval $[-N_{FFT}/2 : N_{FFT}/2 - 1]$, θ_i is the measured phase at the sub-carrier p_i , x_1 is the CPE estimate, and x_2 is the phase slope estimate. This equation of the form $Ax = b$ is approximated using least squares by solving the system of equations $A^T Ax = A^T b$, where the operator T represents the transpose of a matrix. The matrix $A^T A$ is formed as presented in Equation 7.9, while the vector $A^T b$ is shown in Equation 7.10.

$$A^T A = \begin{bmatrix} M & \sum_{i=1}^M p_i \\ \sum_{i=1}^M p_i & \sum_{i=1}^M p_i^2 \end{bmatrix} \quad (7.9)$$

$$A^T b = \begin{bmatrix} \sum_{i=1}^M \theta_i \\ \sum_{i=1}^M p_i \theta_i \end{bmatrix} \quad (7.10)$$

The solution of the 2×2 system of equations is obtained by Gaussian elimination. The LS estimates of CPE and phase slope are used to feed the proportional-integral loop filters that in turn feed time bases formed by accumulators. The output of the accumulators is used to form the phase curve, converted to a rectangular representation of the phase correction for each sub-carrier. The input samples are rotated by the phase correction, producing the phase equalization. Gaussian elimination requires one division per estimate.

Demodulator

The matched filter applied at the channel equalizer has the undesirable effect of scaling the original amplitude by the magnitude of the channel squared. This is not inconvenient for BPSK and QPSK demodulation, as the information is contained in the sign of the in-phase and quadrature components. In the 16-QAM case, the amplitude scaling term is compensated by performing the mapping of the constellation back to bits using the properties of the standard constellation shown in Figure 7.25. If the received word is represented as $b_0b_1b_2b_3$, then:

$$\begin{aligned}
 b_0 &= \Re(h^*x) > 0 \\
 b_1 &= \|\Im(h^*x)\| - \frac{2}{\sqrt{10}}h^*h < 0 \\
 b_2 &= \Im(h^*x) > 0 \\
 b_3 &= \|\Re(h^*x)\| - \frac{2}{\sqrt{10}}h^*h < 0
 \end{aligned} \tag{7.11}$$

where h and x represent the channel response and the FFT output at a sub-carrier, respectively and the operators $\Re$ and $\Im$ represent the real and imaginary parts. The term $\frac{2}{\sqrt{10}}h^*h$ represents the amplitude scaling of a quadrant centroid, located in the coordinate $(2, 2)$ of the ideal 16-QAM constellation.

Modular structure

The module organization and file dependencies for the OFDM receiver implemented on the Harris SiP 1 is presented in Figure 7.26. The corresponding graph for the receiver implemented on the Harris SiP 2 is presented in Figure 7.27.

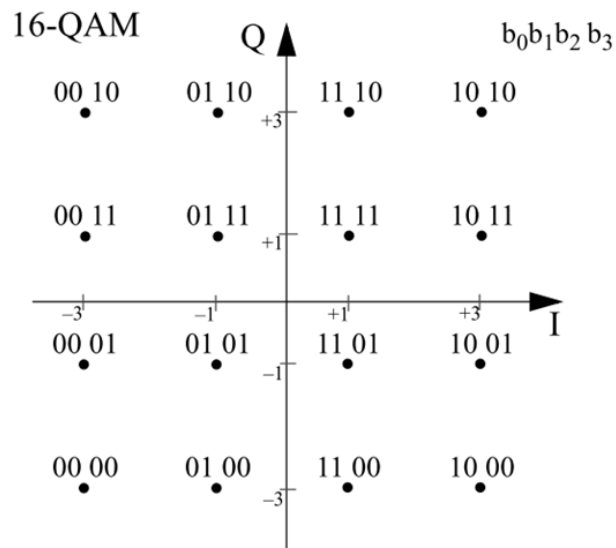


Figure 7.25: IEEE 802.11 16-QAM constellation (from [4])

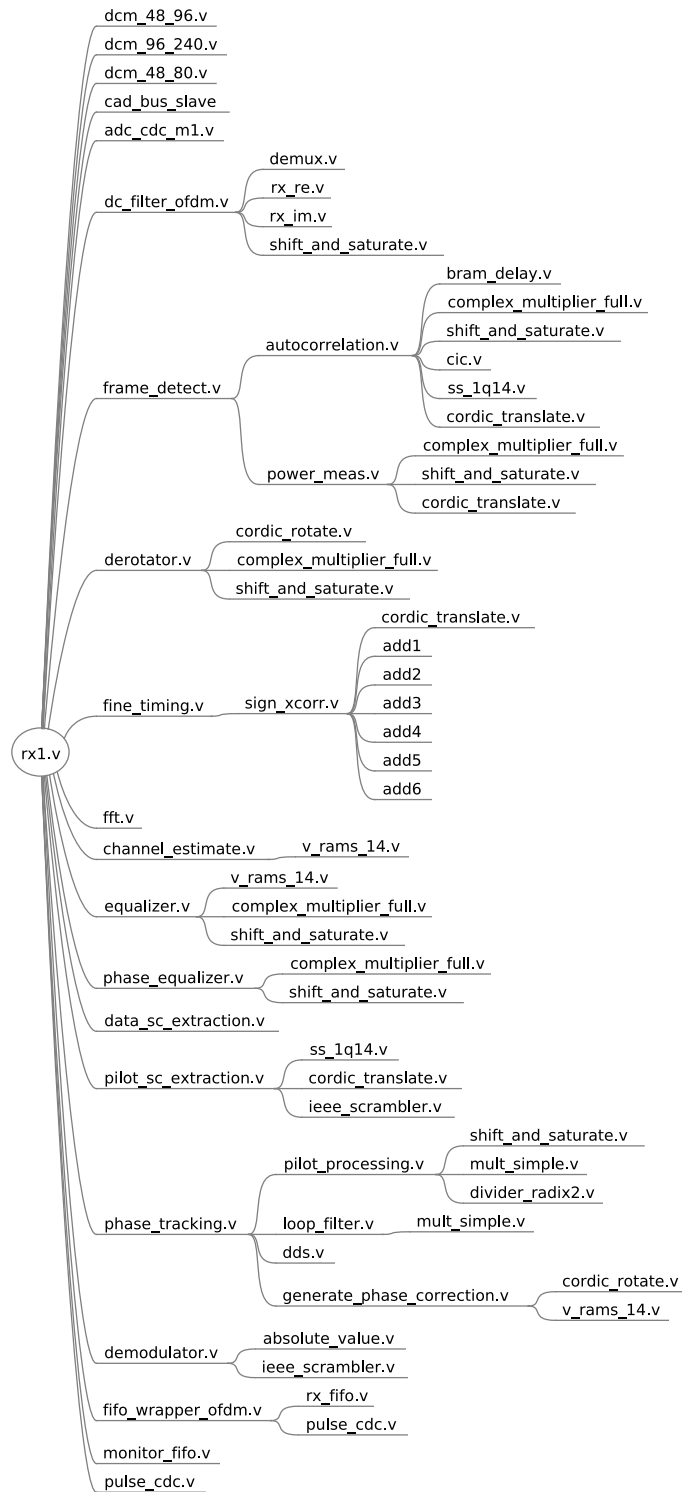


Figure 7.26: Module hierarchy for the OFDM receiver on SiP 1

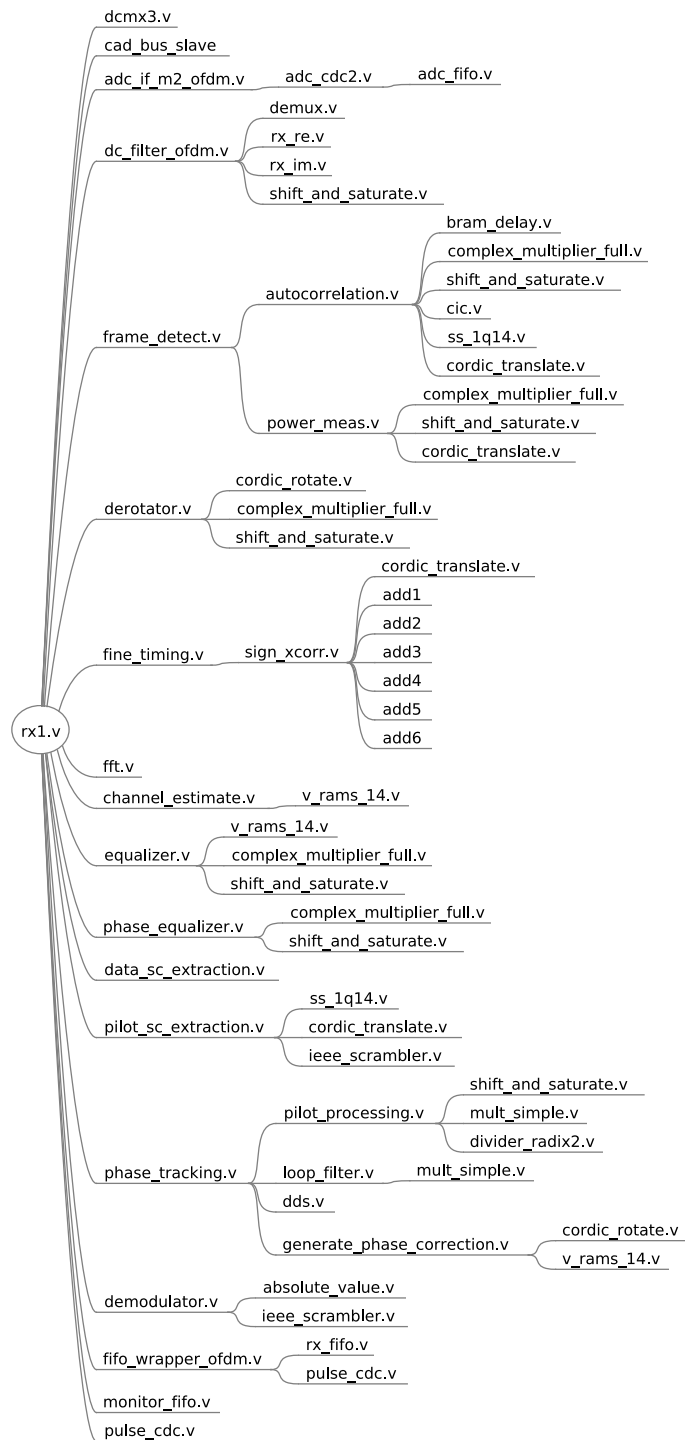


Figure 7.27: Module hierarchy for the OFDM receiver on SiP 2

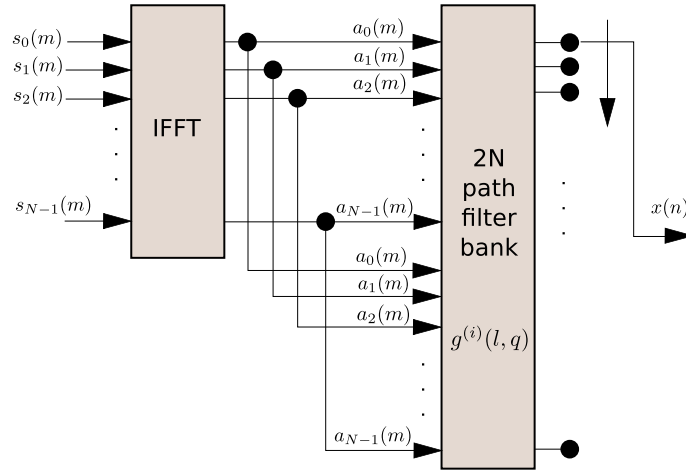


Figure 7.28: FMT transmitter processing with over sampling rate of two

7.2 Hardware Architecture of the FMT Radio

A filter bank multi-carrier radio implementation with interference rejection was prototyped using the OFDM radio as a departing point. On the transmit side the only differences are the inclusion of the transmit filter bank, a new outer state machine to accommodate the new frame structure, and ROM containing the CAZAC sequence frequency domain description. On the receive side, an innovative implementation of the frame detection, fine timing offset estimation, and channel estimation functions is developed using the algorithm of Chapter 6.

7.2.1 FMT Transmitter

A way to implement a FBMC FMT transmitter with an up sampling factor of two is to feed two copies of the IFFT output to a filter bank composed of a $2N$ path polyphase decomposition of the baseband prototype filter. The only architectural difference between the OFDM transmitter presented in Section 7.1.4 and the FMT transmitter is the addition of logic for the replication of the IFFT output as well as the output filter bank. The filter bank interconnection is presented in Figure 7.28.

A new outer state machine is required to accommodate the new frame design with spacing

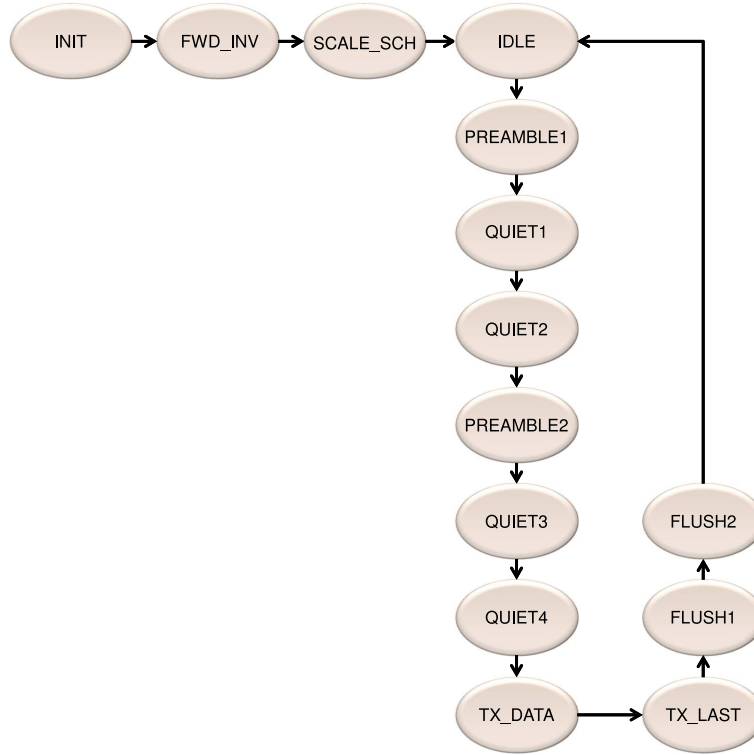


Figure 7.29: FMT transmitter outer state machine

between the isolated preambles. The ROMs containing the IEEE 802.11 short and long preamble tables is replaced by the $\bar{C}_{N-1}$ CAZAC sequence with $N = 64$.

The outer state machine, presented in Figure 7.29, is modified to include the quiet times between the first and second preamble, and between the second preamble and the data field. Flush states are added at the end of the transmission to cover the case when two consecutive frames are transmitted.

Transmission Filter Bank

The filter bank is designed to produce a root of raised cosine (RRCOS) prototype filter response with a roll-off factor $\alpha = 0.99$. The filter design is subject to the constraint of having reduced time-domain side lobes to avoid spurious frame detections, and a time-duration of two symbols to avoid interference between preambles according to the frame

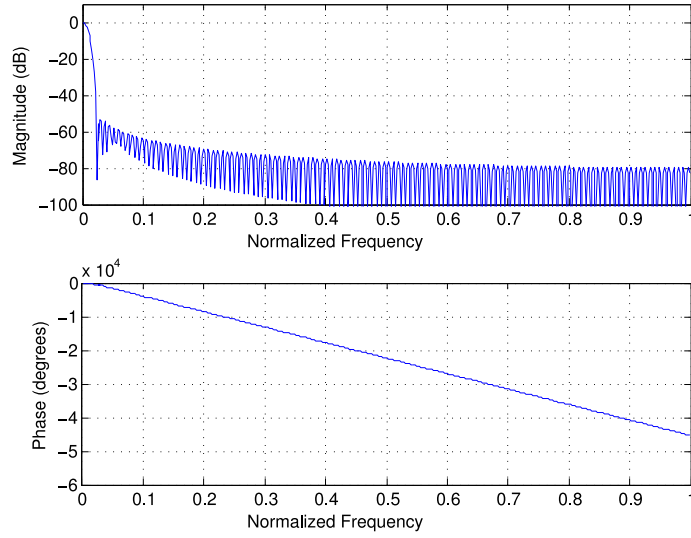


Figure 7.30: Frequency domain response of the transmit side prototype filter

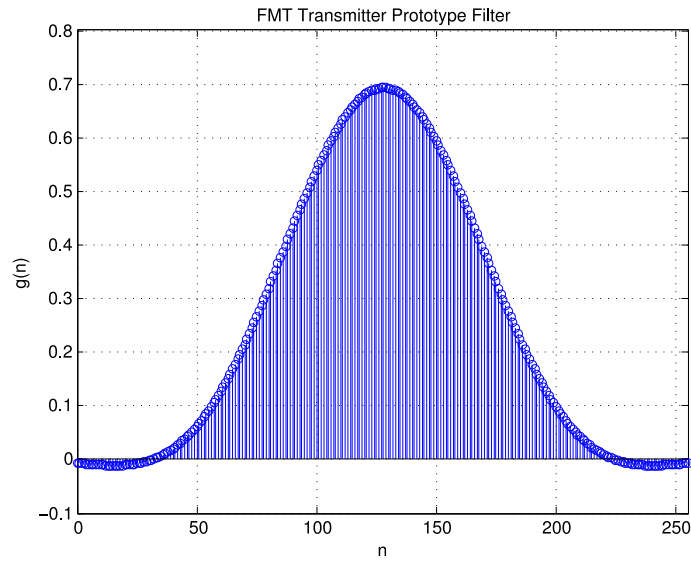


Figure 7.31: Impulse response of the transmit side prototype filter

design presented in Chapter 6. Despite these constraints, a side lobe attenuation of more than 50 dB is obtained, as shown in Figure 7.30. The time domain response of the prototype filter is presented in Figure 7.31. For an oversampling rate of two, each symbol contains 128 samples.

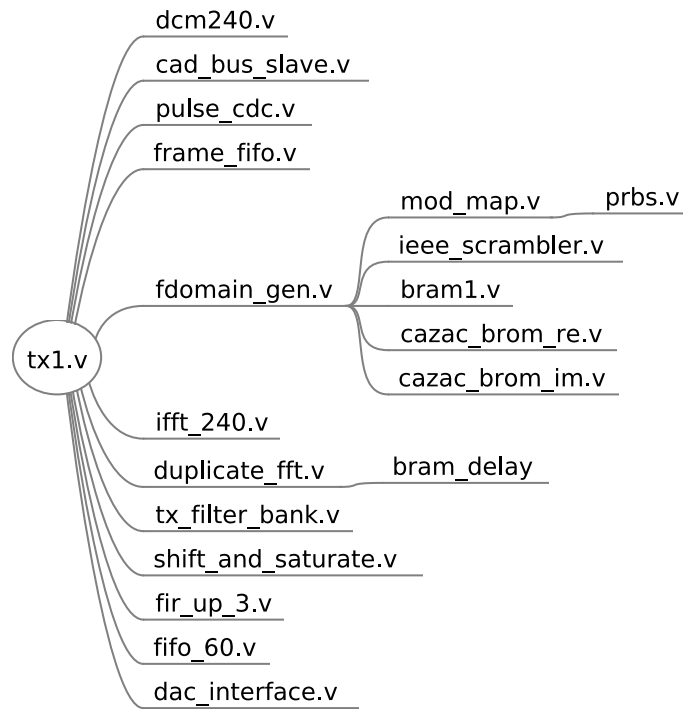


Figure 7.32: Module hierarchy for the FMT transmitter

Modular structure

The module structure and file dependencies of the FMT transmitter are presented in Figure 7.32.

7.2.2 FMT Receiver

An FMT receiver with primary user interference rejection was developed on an FPGA based on the algorithm presented in Section 6.4. The frame detection and fine timing synchronization sections of the base OFDM receiver are replaced by the FMT block processing core. The FFT operator is also included in the processing core due to its application in the frame detection and timing offset estimation tasks. An illustrative diagram is presented in Figure 7.33.

The block diagram of the processing core for frame detection, timing estimation, channel

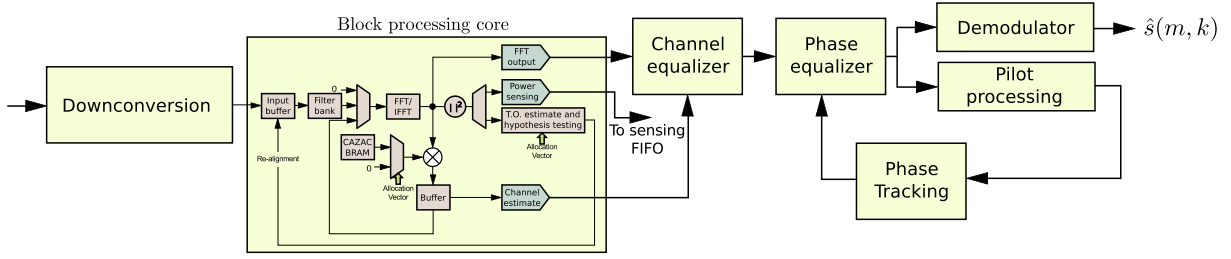


Figure 7.33: Block diagram of the FMT Receiver

estimation, and demodulation is presented in Figure 7.34. This architecture is could be implemented in dedicated hardware (ASIC) or in configurable hardware, of which FPGAs are a particular case. A detailed explanation of each of the components is presented below. A finite state machine (not shown) is required to coordinate the operation of the processing core.

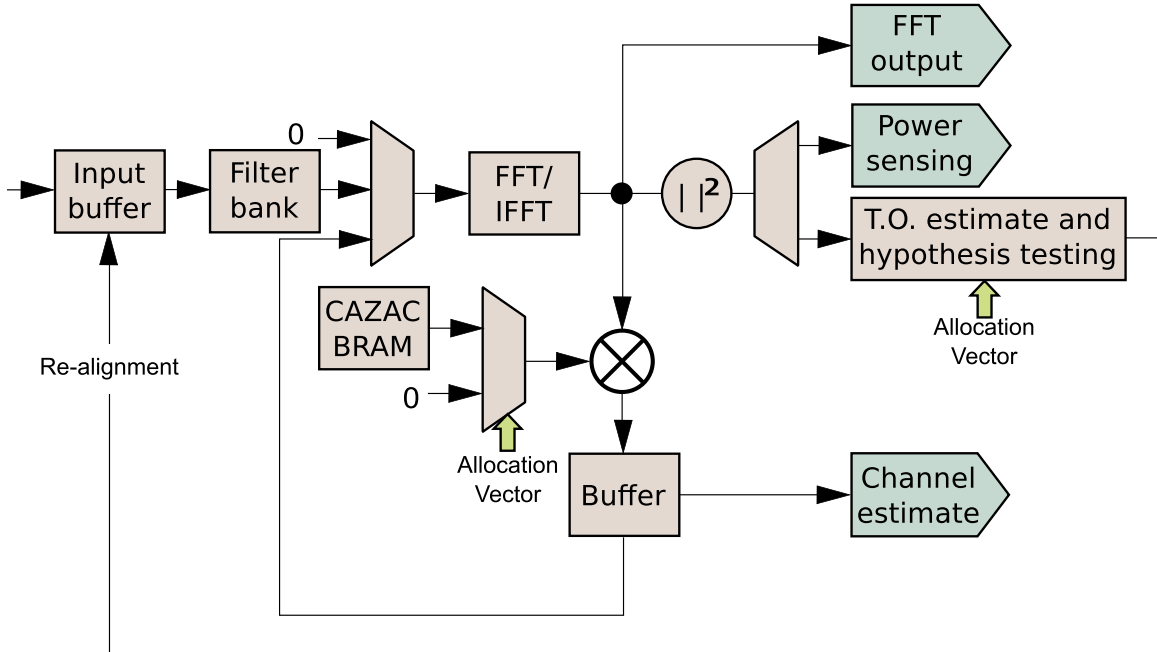


Figure 7.34: Block processing core

1. **Input buffer.** The input buffer permits input signal delay control by pointer manipulation, with the purpose of acquiring fine timing of the input symbols. An alternative

to the use of an input buffer is the direct control of the filter bank state registers. Therefore, the inclusion of an input buffer is optional and depends on the features of the filter bank implemented.

2. **Filter bank.** The filter bank implements the polyphase decomposition of the baseband prototype of the receiver-side matched filter. As such, it must interface to the input buffer, as well as with the FFT/IFFT core.
3. **Pre-FFT Multiplexer.** This multiplexer is used to select the input signal to the FFT/IFFT core: zero when not in use, the filter bank input when in FFT mode, or the feedback buffer (Item 8) output when performing the IFFT operation.
4. **FFT/IFFT.** The FFT/IFFT block can be considered the main processing operation in the core. Several processing options can be chosen according to the design requirements, as fixed or floating point operations, and radix-2 or radix-4 operations, as long as the processing latency is small enough to process the two passes required in the frame detection and timing estimation mode in less than one observation window time.
5. **CAZAC Block RAM.** A memory to hold the complex values of the locally-stored frequency-domain CAZAC sequence.
6. **Excision multiplexer.** The sub-carrier excision operation is performed by zeroing the frequency locations marked as null by the allocation vector. The interfered frequency bins are cancelled as the zeroes are fed to the complex multiplier (Item 7).
7. **Complex multiplier.** The complex multiplier performs the product between the FFT output and the complex conjugate of the excision multiplexer output.
8. **Latency compensation and channel estimate buffer.** This buffer is required for two tasks: the first one is to hold the results of the complex multiplication, which is pipelined with the FFT operation, until the FFT core finishes the output data transfer. This is required because of the type of FFT core chosen is non-streaming. The second

task is to keep the output value of the complex multiplexer output during the half-symbol deambiguation peak search. The set of values obtained at the peak are output later as the channel response estimates.

9. **Magnitude squared.** This module computes the magnitude squared of the input data. This module is included in two of the datapaths: for power sensing, when the FFT output is fed to it, as well as for obtaining the magnitude squared of the cyclic cross-correlation function output from the IFFT core.
10. **Power sensing/hypothesis test demultiplexer.** This demultiplexer routes the output of the magnitude squared module (Item 9) to either the output, to be used as a sub-carrier power sensing measurement, or as the magnitude of the hypothesis test and timing offset estimator module.
11. **Timing offset estimate and hypothesis testing module.** This module performs the functions explained in Section 6.4.3. If the result of the hypothesis test is positive, then the timing offset estimate is fed to the input buffer to perform the re-alignment operation.

Latency Considerations

Latency is an important constraint in the proposed processing core. In the OFDM receiver, based on time domain calculations, the FFT is used *after* frame detection and timing are achieved. In the frequency-domain block-based approach, a closed loop is formed, where the FFT and IFFT operations are involved in the calculation of the frame detection statistic and in the timing offset estimation. The latter is used to re-align the input buffer. To minimize the duration of quiet symbols in the frame structure, a latency budget equal to the duration of one FFT observation window was set. Given that the sampling rate is 20 MHz, under a processing clock of 240 MHz the maximum loop latency is 768 clock cycles ($N \times 240/20$).

A Radix-2 Lite, Burst I/O FFT core was used to implement the OFDM transmitter and

Table 7.6: FFT core configuration options

Core type	Latency	Multipliers	Block RAM
Radix-2 Lite, Burst I/O	539	2	2
Radix-2, Burst I/O	422	3	3
Radix-4 Burst I/O	241	9	7
Pipelined, Streaming I/O	266	8	1

receiver, as well as the FMT transmitter. To optimize the multipliers in the OFDM receiver implementation, the samples at the output of the FFT core are buffered and re-timed at one sample every four clock cycles, allowing the implementation of a low area complex multiplier; further, feeding the data back into the IFFT operator requires a new set of buffering operations.

To satisfy the latency constraints in the FMT receiver, a low latency complex multiplier was designed at the cost of additional multipliers, eliminating the need of re-timing buffers. The FFT operator was re-designed for low latency, choosing the Radix-4, burst I/O configuration among the options presented in Table 7.6.

Table 7.7 presents the total loop latency before and after the modifications required to fulfill the latency budget. Some of the quantities are approximated.

Data flow for spectrum sensing

The data flow for spectrum sensing is presented in Figure 7.35. During the sensing stage, the magnitude squared of the FFT results is output. Observe that the application of spectrum sensing is concurrent with the frame detection and timing estimation application. Spectral containment is obtained as a result of the filter bank operation, with several degrees of isolation between subbands according to the baseband prototype filter design.

Table 7.7: Latency of the timing alignment logic (clock cycles)

Operation	Re-using from OFDM radio	New design
Filter bank	N	N
FFT	539	241
Re-timing buffer	3N	Eliminated
Complex product with C_{N-1}^*	10	6
Re-timing buffer	3N	Eliminated
IFFT	539 (Radix-2 Lite)	241 (Radix-4)
Re-timing buffer	3N	Eliminated
Magnitude Squared	10	6
Peak detector	N+6	N+6
Accumulated Latency	1808	628

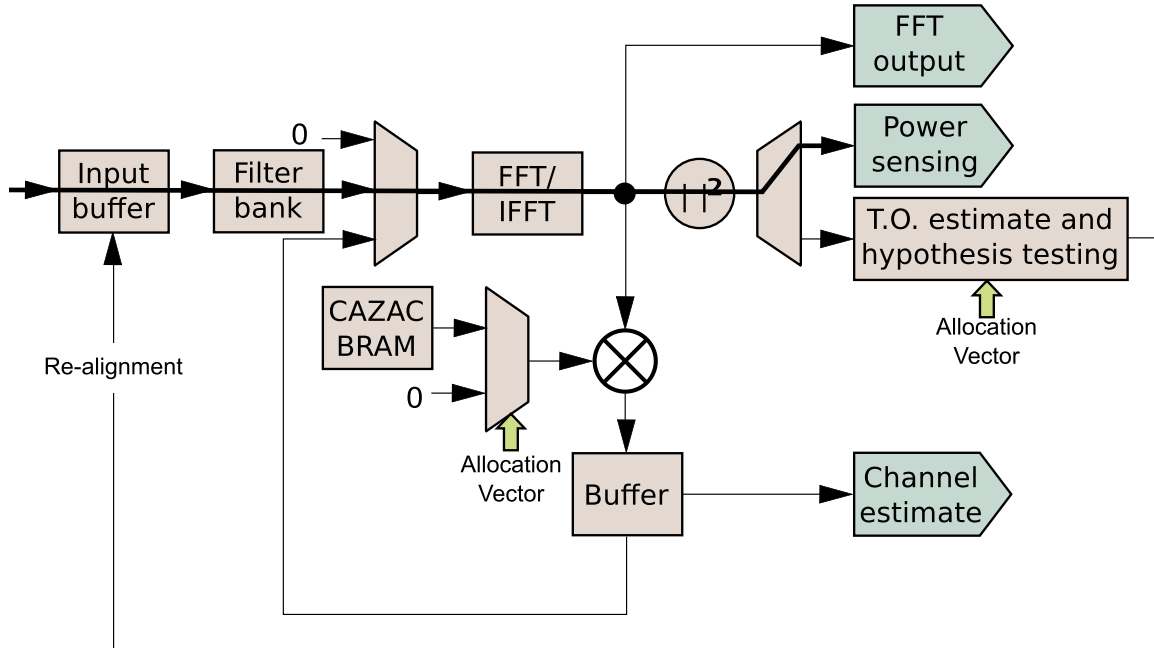


Figure 7.35: Dataflow for spectrum sensing

Data flow for frame detection, timing estimation, and timing alignment

Figure 7.36 presents the dataflow for the frame detection, timing estimation, and timing alignment operation. As mentioned before, the re-use of the FFT/IFFT core imposes strict latency conditions.

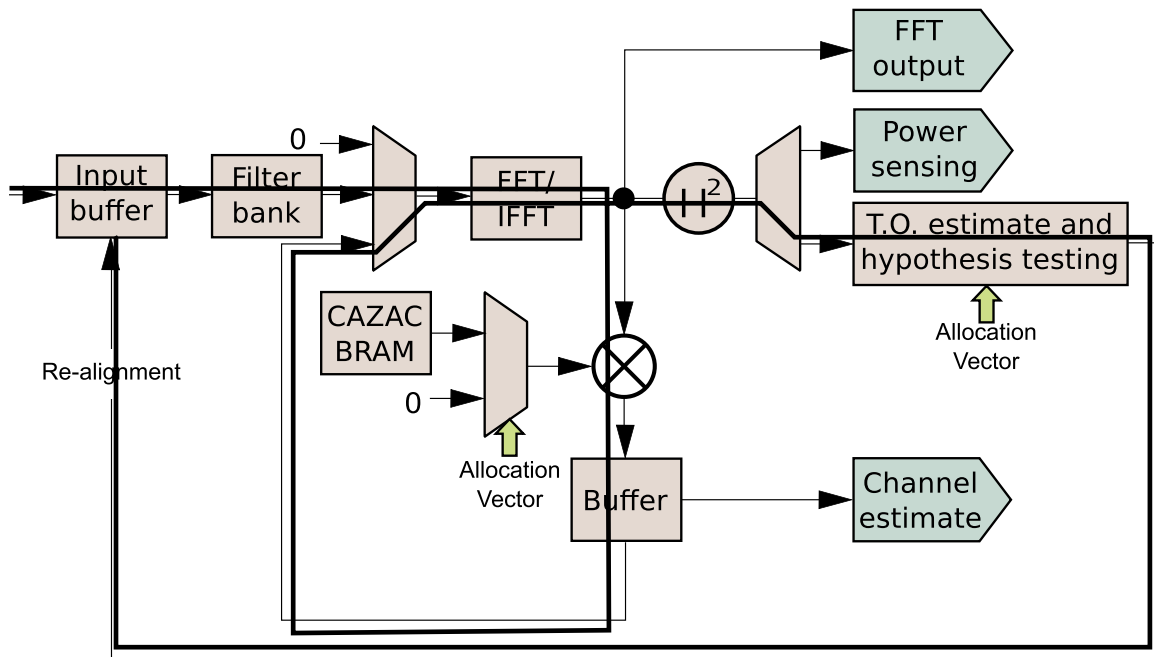


Figure 7.36: Dataflow for frame detection, timing estimation, and timing alignment

Data flow for channel estimation

A channel estimate is obtained after a peak search is performed over the half-symbol aligned second preamble P_2 . The peak location also indicates the right sampling time for the following data symbols, as a half-symbol ambiguity exists for the case of an oversampling rate of two. The dataflow for this application is presented in Figure 7.37.

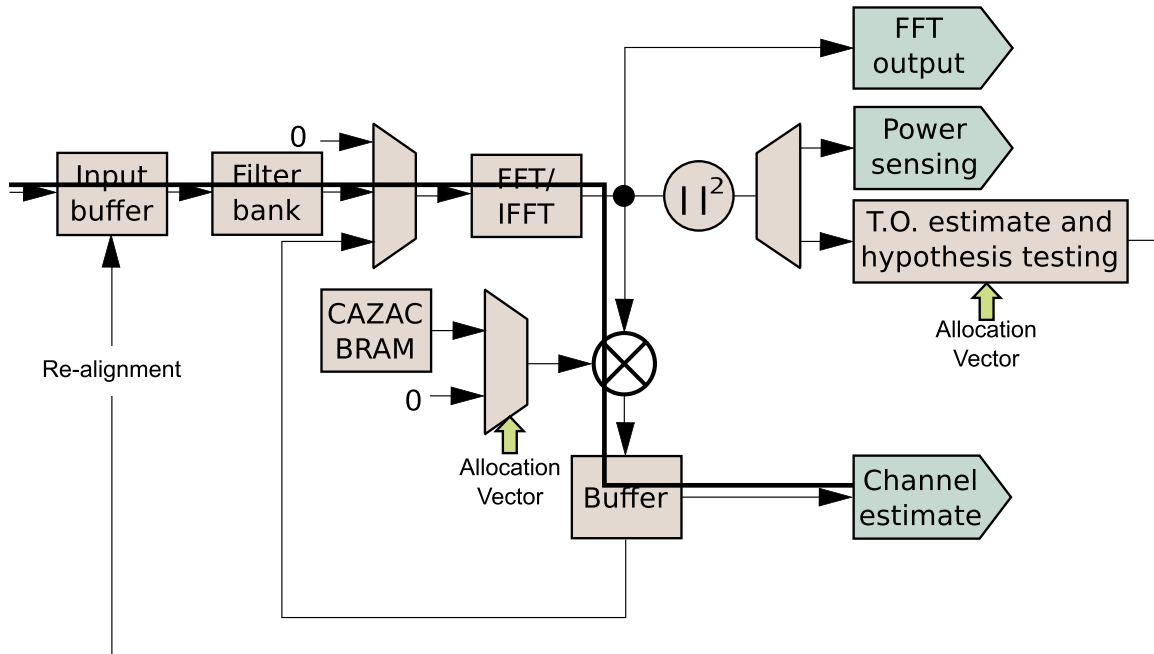


Figure 7.37: Dataflow for channel estimation

Data flow for demodulation

Signal demodulation, the simpler operation performed by the module, can only be performed after the tasks of spectral sensing, frame detection, timing estimation, and channel estimation have been performed. It consists of simply outputting the FFT result of the correctly aligned modulated filter bank. The dataflow for this application is presented in Figure 7.38.

Modular structure

The modular structure and dependencies for the FMT receivers implemented on the Harris SiP 1 and on the Harris Sip 2 are presented in Figures 7.39 and 7.40 respectively.

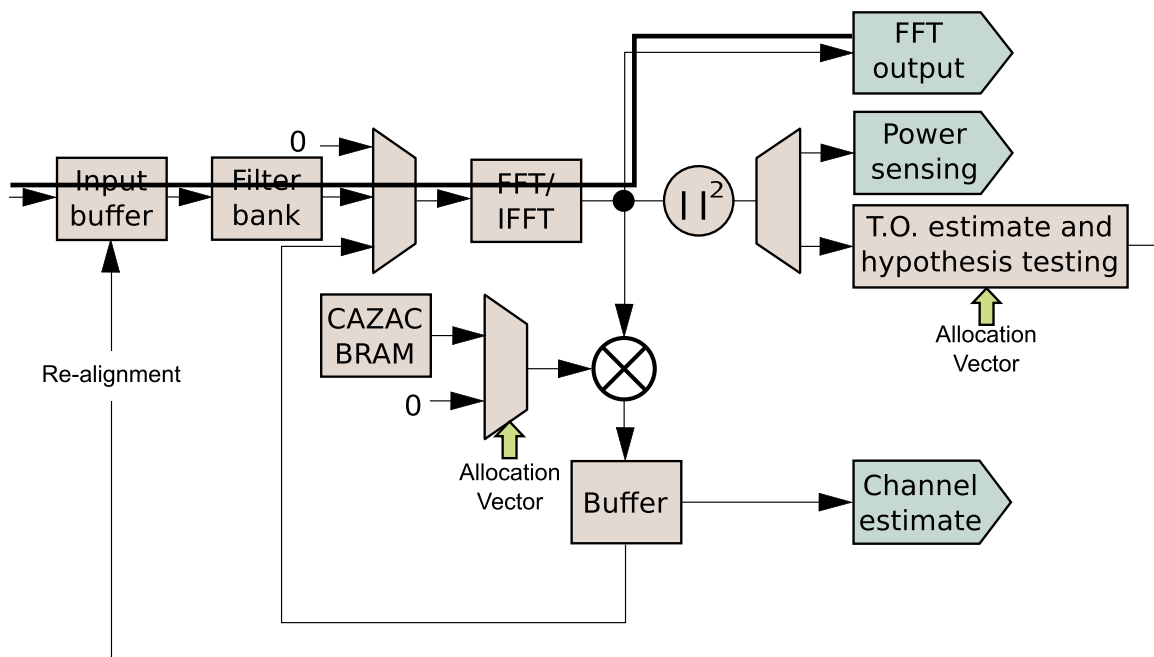


Figure 7.38: Dataflow for demodulation

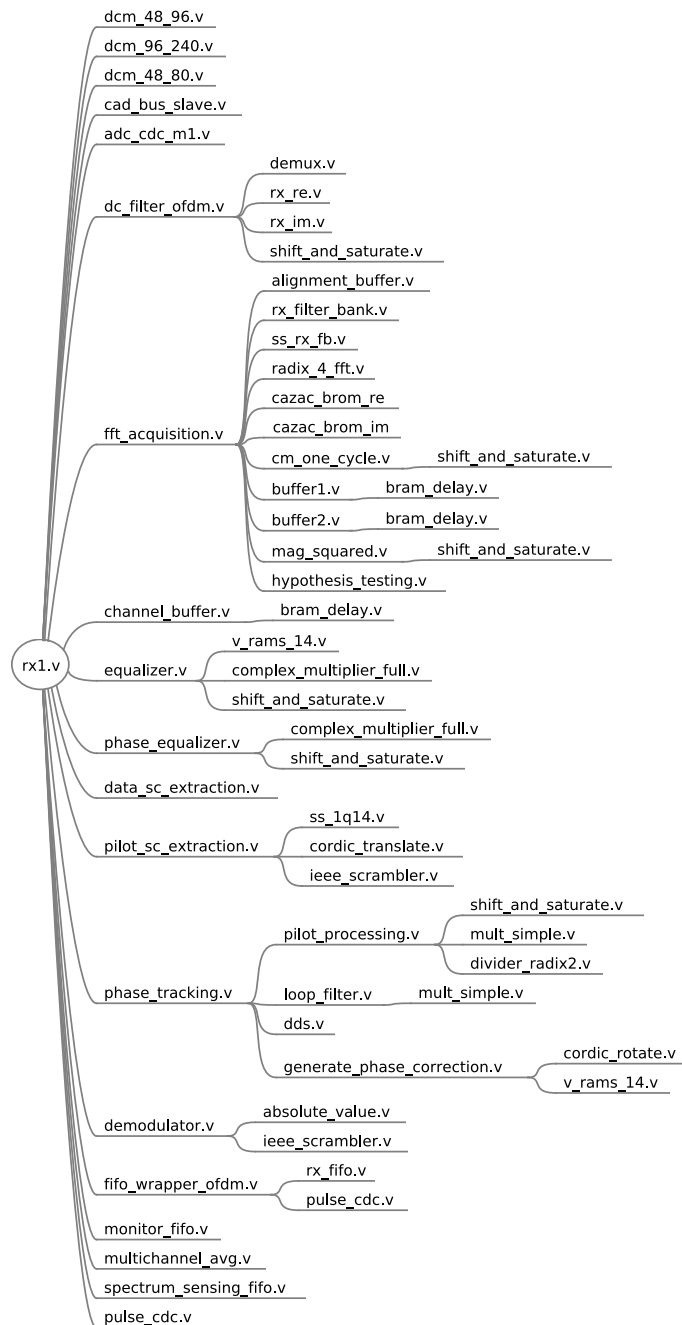


Figure 7.39: Module hierarchy for the FMT receiver on SiP 1

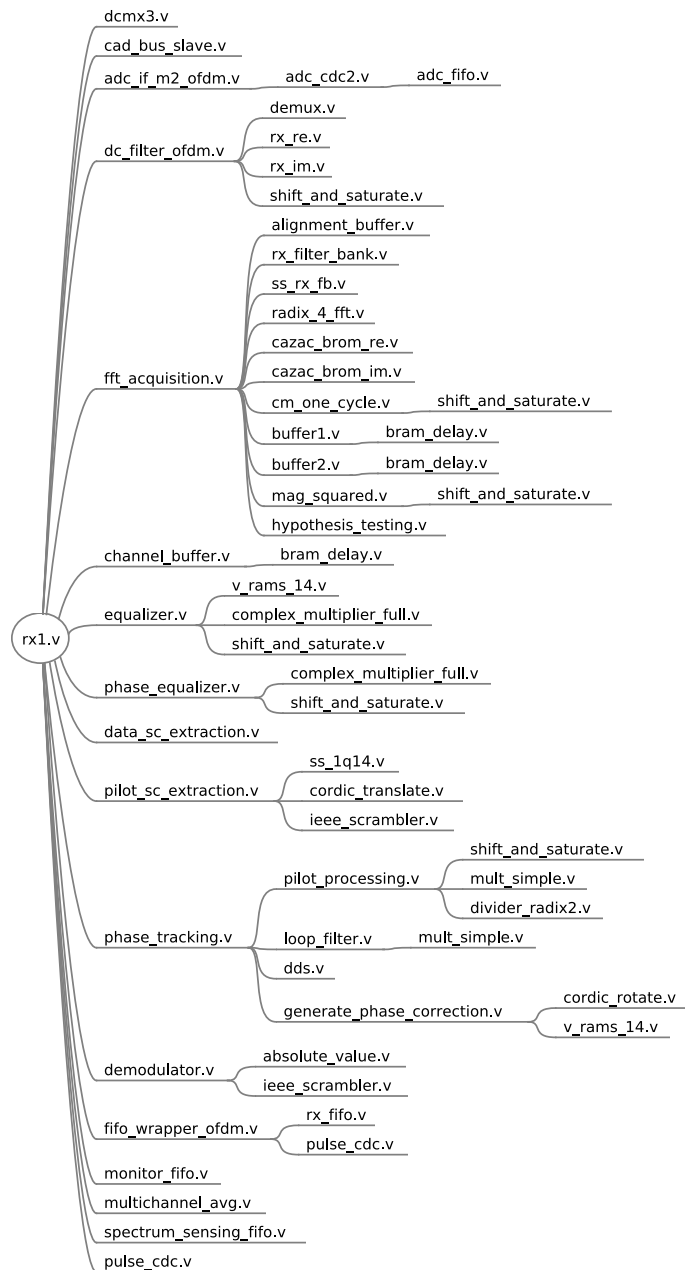


Figure 7.40: Module hierarchy for the FMT receiver on SiP 2

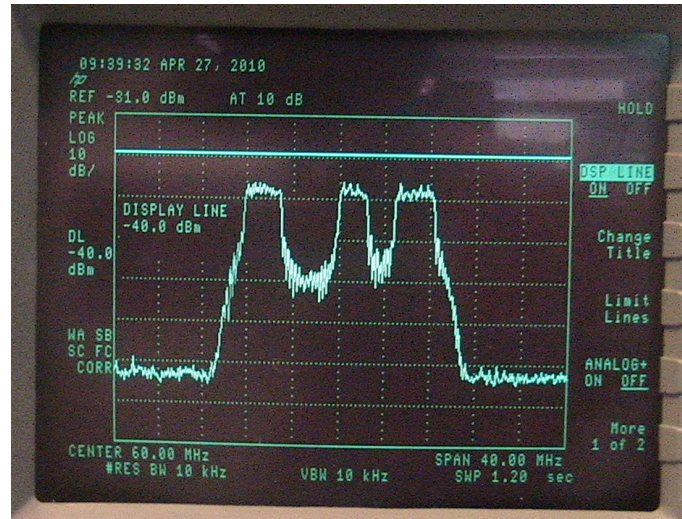


Figure 7.41: OFDM spectrum with two sets for zeroed sub-carriers

7.3 Results

This section presents results related to the spectral shape of the OFDM and FMT radios, as well as the resource utilization of their different blocks. The prototypes constructed are operational for transmission and reception of data over-the-air.

7.3.1 Measured Spectrum

The spectral shape measured at the received down converter using a fixed OFDM transmitter with control of the allocation vector is presented in Figure 7.41. Observe that the spectral rejection in the null band is less than 20 dB. The same measurement was taken for a FMT transmitter with the same set of nulled sub-carriers. The rejection obtained is superior to 50 dB, as shown in Figure 7.42.

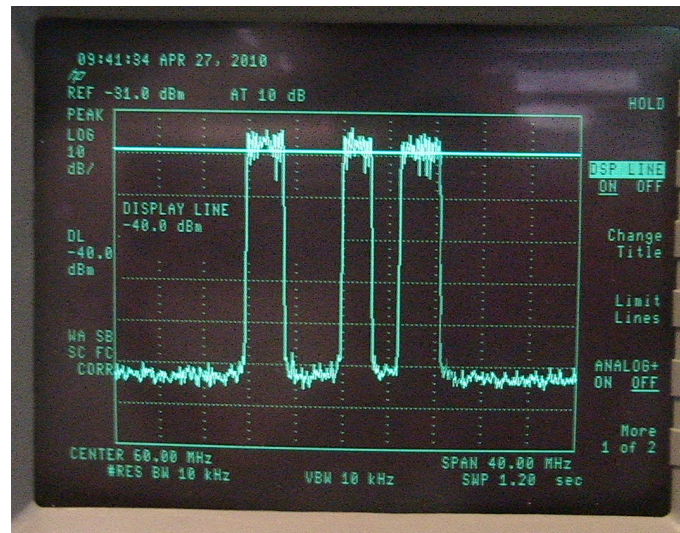


Figure 7.42: FMT spectrum with two sets for zeroed sub-carriers

7.3.2 FPGA Resource Utilization

The resource utilization of the filter bank multi-carrier by modules instantiated in the top level is presented in Table 7.8. As mentioned above, the difference between the OFDM and the FMT receivers is the acquisition module (encoded in the *fft_acquisition.v* file). The last row of Table 7.9 shows the added utilization of the modules that develop the equivalent function in the OFDM receiver.

A key feature of the modulated filter bank is its efficiency in terms of physical resources. For comparison purposes, an FIR filter with a rejection of 45 dB in the interfered band with transition bands equal to one-half a sub-carrier bandwidth was designed to mimic the equivalent frequency response of the excised filter bank. The filter is designed to reject sub-carriers 24 to 34 out of 64, and requires an order of 800. Its frequency response is presented in Figure 7.43.

Such a filter has the additional disadvantage of requiring dynamic coefficient reloading according to the allocation vector. This means that a new set of coefficients must be calculated according to each new allocation vector, and downloaded to block RAM resources in the fil-

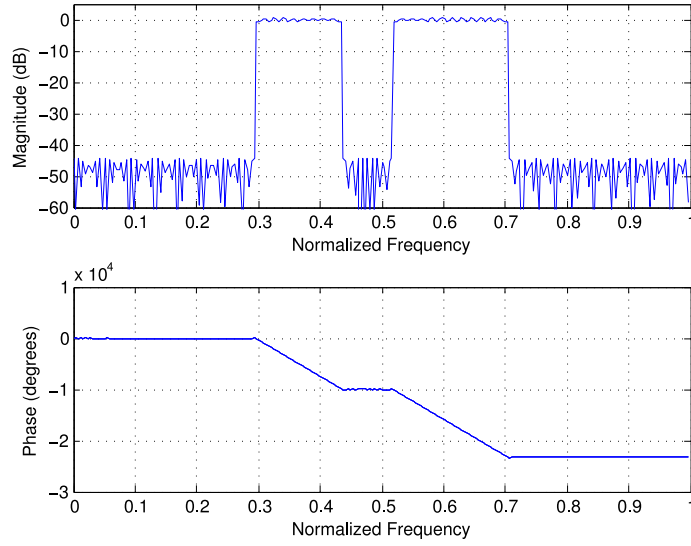


Figure 7.43: Equivalent rejection filter

ter. Further, by filtering out the side lobes of the OFDM signal, the filter produces distortion to the waveforms of the individual sub-carriers.

The filtering operation is performed at the sampling frequency of 80 MHz, immediately at the ADC output, to support the non-symmetric characteristic of a generalized response. Consider that this filter is only effective on the receiver side, and does not contribute to mitigate the $SU \rightarrow PU$ interference.

A comparison of the hardware requirements in terms of DSP-48 blocks at the frame detection stages of the non-filtered OFDM and FBMC receivers, as well the hypothetical interference rejection filter is presented in Figure 7.44. Note that an OFDM receiver with an interference rejection filter would require the addition of their respective DSP resources. This result shows the impressive benefit of filter bank methods in terms of implementation cost to achieve the interference rejection capability.

The main difference between the OFDM and the FMT transmitter is the addition of the filter bank on the latter. The filter bank represents two additional multipliers and other logic as presented in Table 7.10.

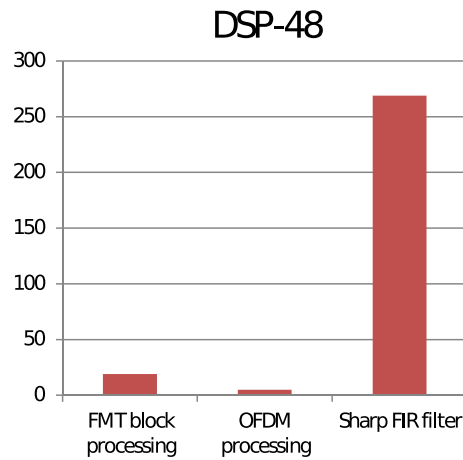


Figure 7.44: Utilization of DSP resources

Table 7.8: Resource utilization of the FMT receiver

Module	Total LUTs	BRAM	DSP-48	Comments
dc_filter_ofdm.v	376	0	8	Down conversion
fft_acquisition.v	2990	17	19	Block acquisition
channel_buffer.v	9	1	0	Stores channel response
equalizer.v	136	2	2	Channel matched filtering
phase_equalizer.v	108	0	2	Phase rotation
data_sc_extraction.v	163	0	0	
pilot_sc_extraction.v	1156	0	0	
phase_tracking.v	2161	1	6	Dual control loop
demodulator.v	275	0	1	
fifo_wrapper.v	288	15	0	Data interface to ARM
monitor_fifo.v	362	60	0	I,Q, and channel for display
multichannel_avg.v	1532	1	6	Average spectrum sensing
spectrum_sensing_fifo.v	60	1	0	Power density measurement
Total	9616	98	44	

Table 7.9: Resource utilization of the OFDM acquisition stage

Module	Total LUTs	BRAM	DSP-48	Comments
frame_detect.v	2566	6	2	Includes CFO measurement
derotator.v	1178	0	1	complex product
fine_timing.v	1826	0	0	Optimized cross-correlation
fft.v	454	3	2	Radix-2 Lite Burst I/O
Total OFDM acquisition	6024	9	5	

Table 7.10: Resource comparison of the transmitters

Radio	Total LUTs	BRAM	DSP-48
OFDM transmitter	3726	18	12
FMT transmitter	4722	21	14

7.4 Conclusion

This chapter presented a detailed discussion of the hardware design of a functional OFDM physical layer that was tested over-the-air. This radio was implemented with a built-in capability to be parameterized by an allocation vector created externally, that controls the sub-carriers activated/deactivated and assigned as pilots. The OFDM radio relies on the IEEE 802.11 preambles. The short preamble symbols impose limitations on the set of sub-carriers that can be de-activated, due to its inherent sparseness. This limitation is solved in the FMT radio design with the use of CAZAC sequences.

On the receive side of the OFDM radio, the allocation vector manages the calculation of the phase measurement, allowing the use of an arbitrary number and location of pilots, thanks to the properties of the $A^T b$ and $A^T A$ matrices that are formed by accumulation and whose size does not depend on the number of pilots. The allocation vector also manages the data and pilot extraction module. However, primary user rejection is not implemented on this

radio due to the expensive input filters required, and their inability to match an arbitrary allocation pattern.

A novel architecture for block processing of the frame detection functions in a FMT receiver is presented. This structure permits the calculation of the hypothesis test variable, as well as the rejection of the primary user interference using the allocation vector to signal the sub-carriers that must be excised. The structure permits the calculation of the cross-correlation function by re-use of the FFT core, subject to latency constraints that increase the area of the operators. This area increase is small compared to the resources that it would take to implement a dynamic input filter with the same rejection and frequency agility capabilities.

A prototype setup was implemented to demonstrate the capabilities of the radio in presence of a wide band interferer. Further test setups are recommended to evaluate the systems in terms of error vector magnitude (EVM), frame detection rate, false frame detection rate, frame error rate, and bit error rate. Another measurement to consider is the resistance of the fixed-point implementation of the FFT to high power primary users.

Chapter 8

Conclusions

The advent of cognitive radio and secondary spectrum applications was expedited by the first regulations that establish the use of TV white space. This creates the need for a new set of technologies and services that make an efficient use of the spectrum, thus relieving the so-called *spectrum scarcity*. Several authors have pointed to the use of multi-carrier techniques as a way to achieve transceivers that are aware of the spectrum utilization and respond dynamically with sub-carrier allocations fitted to the white spaces left by primary users.

A modeling technique for the primary user spectral shape was proposed in Chapter 5 under the assumption of single carrier linearly modulated primary users. In a further development, the parameters extracted from the spectral shape of the signal are used as initial estimates of an Automatic Modulation Classification system, and when augmented with a radio receiver synthesis system provides the capability to extract data from the primary user.

While OFDM seems like an appropriate technique for the development of secondary spectrum applications, its high side lobes and lack of spectral containment induce mutual interference and a low efficiency when dealing with one or more occupied spaces in the spectrum. Guard or canceling sub-carriers are required to share the spectrum resources without drastically

affecting the performance of the communications of both primary and secondary users.

Thanks to their tight spectral containment, FBMC methods outperform OFDM for secondary user applications in terms of their low mutual interference and computational expense, and high efficiency when compared to external rejection filters. The sharp spectrum etching capabilities of FBMC are demonstrated by spectrum analyzer measurements even for a small prototype filter length.

A block processing algorithm for the implementation of frame detection and fine timing estimation on filtered multi-tone, a variant of FBMC, was developed and tested on an FPGA platform, demonstrating its capabilities to reject interference by excision of the affected frequency bins. The results show a gradual degradation in performance as more sub-carriers are nulled. This degradation is too high under sparse allocations, thus ruling out the hypothesis of performing optimized FFT/IFFT operations by pruning algorithms.

The proposed algorithm is scalable and benefits from a larger FFT size if the dwell time is increased. The use of CAZAC sequences makes the system immune to frequency offset errors at the price of mapping the CFO to a timing offset. Under this case, the frames are still detected and further enhancements could be introduced to correct the induced timing offset.

A frame design (air interface protocol) to support the frequency domain block processing was proposed and evaluated. To avoid ambiguities produced by inter-symbol interference (ISI) at the frame detection time when the system is out of timing alignment, isolated preamble symbols were used. The time isolation of the preamble symbols imposes restrictions on the length of the prototype filter. Even with these restrictions, a modest length of two symbols can produce side lobes as low as -60 dB relative to the main lobe. Therefore, the constrained design retains the main strength of FBMC methods, namely their spectral containment.

In the current demonstration setup, explained in detail in Appendix A, the tasks are partitioned into three tiers, shown in Figure A.1. The sensing and radio physical layer presented in Chapter 7 is implemented on two FPGAs in the Harris SiP. The ARM processor in the

SiPs runs processes to extract the sensing information from the FPGAs, and passes it via TCP sockets to a server computer where a LabView manager application is in charge of displaying the channel and the constellation diagrams. The determination of the PU occupation is performed by comparison of the modulated filter bank output magnitude with a threshold set by the user via a graphical interface. The manager application can be augmented to include the PU modeling algorithm presented in Chapter 5. Using the graphical interface, the user can choose to use template allocation vectors or let the tools perform the sub-carrier allocation based on the spectrum awareness algorithm.

The parameter estimation and automatic modulation classification algorithms for a primary user were successfully implemented and applied to the Rapid Radio project. One of the assumptions of the parameter estimation algorithm is the use of a periodogram based on a non-smoothed FFT operator with a high spectral resolution that may not be available in a multi-carrier receiver. This requirement is an opportunity to investigate efficient ways to reuse the available processing resources in a receiver for enhancing the spectral resolution of the sensing operation. Another powerful concept for cognitive radios is the demodulation of the primary user signal as a means to obtain embedded scheduling information among others.

The parameterization algorithm developed for the hardware in terms of an allocation vector formed by $2N$ bits permits a high speed of response to the new spectral allocation, enhancing the system dynamics. The re-parameterization is performed by writing the allocation vector to a register that is read each time a symbol in the frame is processed.

In the present prototype, sensing information and allocation vectors are transmitted using the existing wired network. The sensing and allocation information distribution across the network elements remains a central problem in secondary spectrum applications.

The mathematical description of the algorithm for spectrum sensing, frame detection, timing offset estimation, channel estimation, and demodulation tasks, that can be mapped to a hardware architecture was presented, as well as the limitations of the algorithm. An in-

interesting result is that the cross correlation performed at the output of the filter bank is equivalent to passing a sequence of cross-correlation coefficients through the transmit and receive filters, thus achieving an enhancement in the detection performance of the system.

The computational efficiency of the filter bank for the implementation of the FBMC modulation was demonstrated by actual implementation. The latest version of Xilinx CoreGen provides filter banks that connect seamlessly to the FFT cores, but are limited to critically sampled implementations. Non-critically sampled systems require the addition of buffering elements in the case of oversampling by integer rates. Implementations of non-integer oversampling rates would require control of the indexing operations inside the filter, not provided by today's version of the tools.

8.1 Future Work

This section presents a list of topics revealed in the development of this project that deserve additional research.

Hardware scalability

The current design was developed for an FFT size of 64 and a sampling rate of 20 MHz, following the baseline OFDM design based on the IEEE 802.11a/g standard. While many of the modules can be parameterized to operate with a higher sub-carrier count, others have to be generated from scratch, such as the FIR filter banks, FFT/IFFT, and the CAZAC sequence ROMs. In the case of the block processing module, latency is a critical parameter that must be verified before attempting the assembly of its constituent modules.

A hardware generation flow with parameterized generation of the modules required for the implementation of the multi-carrier transceiver would be a relevant project, considering the success of library-based radios.

Analysis of the dynamic range requirements of the FFT/IFFT operator

The power received from the PU signal can be several orders of magnitude larger than the secondary signal intended for a receiver. The performance of the PU interference rejection feature depends on the linearity of the FFT/IFFT implementation. If saturation occurs at one or more of the stages of the FFT, power from the PU will leak to other sub-carriers producing interference and destroying the intended signal.

A high dynamic range is also required for the calculation of the cyclic cross correlation function used to detect the incoming frame. The purpose of the cross correlation calculation is to obtain a distinctive peak. The IFFT operator is required to hold the maximum expected peak value.

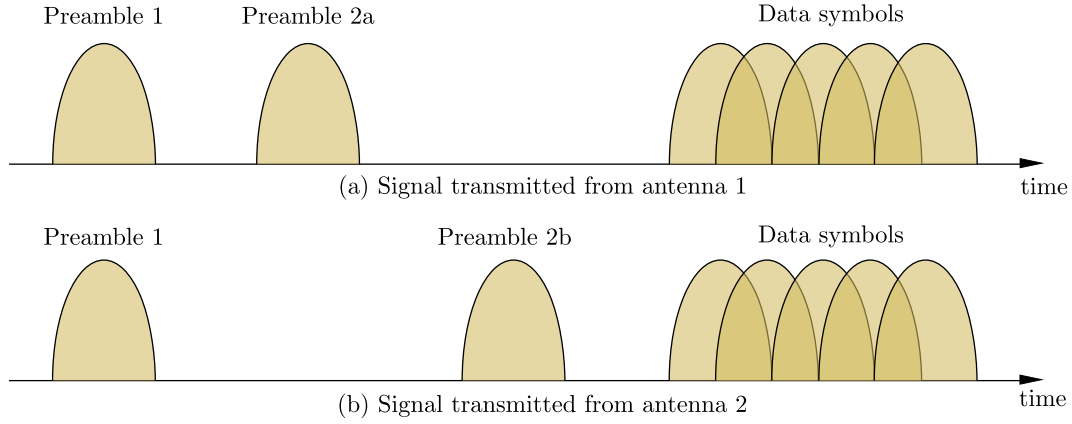
The current implementation of the FFT operator uses 16-bit fixed-point FFT/IFFT operators with scaling schedules found by simulation to minimize saturation events. A floating point implementation would probably be better suited for dynamic range purposes, but it would require a larger area or latency.

Multiband OFDM/FBMC ultra-wide band communications

Ultra-wide band communications schemes have been proposed using multiple sets of multi-carrier signals at non-overlapping bands. The spectrum sensing and arbitrary allocation ideas can be applied to such communication schemes to reduce the amount of mutual interference to and from PUs.

Extension to MIMO

One way to estimate Multiple-Input Multiple-Output (MIMO) channels is the addition of extra “second preamble” symbols, while keeping the first preamble as a common reference for frame detection. For example, the 2×2 case can be implemented by a preamble structure

Figure 8.1: Frame format for 2×2 MIMO

as shown in Figure 8.1. In that case, the channel from Tx antenna 1 to Rx Antenna 1 and Rx antenna 2 can be estimated using preamble 2a, while the same idea can be applied to the second Tx antenna using preamble 2b. Sub-optimal channel models could be used to combine the first preamble symbol to enhance frame detection.

The use of orthogonal preambles can avoid the insertion of extra preambles, with modifications to correct the CFO before performing the cross-correlation operation.

Extension to other oversampling ratios

As mentioned before, the current filter bank realization in Core Generator only supports integer over sampling ratios. Consequently, developing FMT transceivers with arbitrary oversampling ratios would require the construction of the filter banks from scratch. The filter banks are just the tip of the iceberg. The algorithm itself must be augmented to support fractional symbol observation windows and obtain an estimate of the shift for the receive filter bank coefficient set that matches the transmitter side.

OFDM-OQAM

OFDM-OQAM has double the spectral efficiency of FMT, while keeping the same spectrum containment characteristics. One trade-off of this efficiency is that it requires the double of FFT/IFFT operations to generate an offset QAM symbol. With the appropriate frame structure and training sequences, the proposed algorithm and hardware architecture could be modified to also hold OFDM-OQAM.

Extending the dwell time

As presented in Chapter 6, the performance of the algorithm degrades as the number of active sub-carriers is reduced. A way to overcome this limitation is to extend the dwell time at the receiver. One way to achieve this is to increase the number of preambles at the expense of frame efficiency. A detailed analysis is required to determine if this is a viable approach. A positive by-product of this strategy might be the ability to estimate carrier frequency offset.

CFO estimation

As mentioned before, CFO estimation may be gained when having several preamble symbols. The autocorrelation between consecutive training symbols may be performed in the frequency domain, while enabling interference rejection. This strategy may be useful to reach a CFO estimate, similar to the way it is done in traditional OFDM. In this scenario, the use of CAZAC sequences needs to be re-evaluated, since their time-frequency ambiguity may be a disadvantage.

Analysis of the algorithm under multi-path channels

Multi-path channels spread the power of the received signal over a non-zero time duration. The net effect of multi-path in terms of the frame detection algorithm is the generation of several peaks, as many as the relevant paths in the signal propagation environment. An optimum algorithm for frame detection should be able to constructively add the peaking signals to enhance the frame detection statistics.

Hardware implementation of primary user model

In this dissertation the modeling of the primary user signal, presented in Chapter 5 was implemented in a external computer and was not implemented in the delivered prototype, that uses a threshold-based energy sensing strategy. Several steps conducive to the hardware implementation of the PU modeling algorithms would be required to provide more information to the spectrum awareness module.

8.2 Publications

The following publications were performed as part of the research presented in this dissertation.

1. Recio, A. and Athanas, P., “Physical Layer for Spectrum-Aware Reconfigurable OFDM on an FPGA,” in *IEEE Euromicro Conference on Digital System Design. Euromicro DSD*. Lille, France 2010.
2. Suris, J.A., Recio, A. and Athanas, P., “On the Implementation of a Quasi-Generic Synchronization Architecture for Linear Digital Modulations,” in *Journal of Signal Processing Systems*. Springer, 2010

3. Suris, J.A., Recio, A. and Athanas, P., “RapidRadio: A Domain-Specific Productivity Enhancing Framework,” in *International Journal of Reconfigurable Computing*, 2011. [Accepted]
4. Recio, A., Suris, J.A. and Athanas, P., “Automatic Modulation Classification for Rapid Radio Development,” in *IEEE Rapid System Prototyping Conference. RSP*. Fairfax, VA 2010.
5. Recio, A., Suris, J.A. and Athanas, P., “Blind Signal Parameter Estimation for the Rapid Radio Framework,” in *IEEE Military Communications Conference, MILCOM*. Boston MA, 2009.
6. Suris, J.A., Recio, A. and Athanas, P., “Enhancing the Productivity of Radio Designers with RapidRadio,” in *IEEE International Conference on Reconfigurable Computing and FPGAs. ReConFig*. Cancun, Mexico, 2009. **This work received the best paper award.**
7. Suris, J.A., Recio, A. and Athanas, P., “Rapid Radio: Human-assisted Signal Classification and Receiver implementation,” in *Software Defined Radio Forum Technical Conference*. Washington D.C., 2008.
8. Suris, J., Recio, A., and Athanas, P. “Rapid Radio: A Framework for Human-Assisted Signal Classification and Receiver Implementation,” in *Proceedings of the 2008 IEEE National Aerospace and Electronics Conference, NAECON 2008*, Fairborn, Ohio.

8.3 Planned Publications

A list of publications that can be derived from the contents of this dissertation is presented next.

1. “Frame Detection and Timing Estimation Using Block Processing for Cognitive Radio.” This paper will discuss the detailed algorithm presented in Chapter 6, and extended it through more analysis of the performance of the algorithm under multipath channel conditions.
2. “A Hardware Architecture for FMT Frame Detection in Cognitive Radio Applications.” This paper will augment the Euromicro DSD paper by considering the FBMC application to non-contiguous sub-carrier allocations in presence of interference, using the hardware description presented in Chapter 7.

Bibliography

- [1] Q. Zhao and B. M. Sadler, “A Survey of Dynamic Spectrum Access,” *Signal Processing Magazine, IEEE*, vol. 24, no. 3, pp. 79–89, 2007.
- [2] R. Rajbanshi, A. M. Wyglinski, and G. J. Minden, “An Efficient Implementation of NC-OFDM Transceivers for Cognitive Radios,” in *Cognitive Radio Oriented Wireless Networks and Communications, 2006. 1st International Conference on*, June 2006, pp. 1–5.
- [3] T. Yucek and H. Arslan, “Spectrum Characterization for Opportunistic Cognitive Radio Systems,” in *Military Communications Conference, 2006. MILCOM 2006*, H. Arslan, Ed., 2006, pp. 1–6.
- [4] “IEEE Standard 802 Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications. (Includes IEEE Std 802.11, 1999 Edition; IEEE Std 802.11a.-1999; IEEE Std 802.11b.-1999; IEEE Std 802.11b.-1999/Cor 1-2001; and IEEE Std 802.11d.-2001),” pp. 1–721, 2005.
- [5] “Slugging and Slug Lines Information for Washington D.C.” [Online]. Available: <http://www.slug-lines.com>
- [6] J. Mitola, *Cognitive Radio Architecture : The Engineering Foundations of Radio XML*. Hoboken, N.J.: Wiley-Interscience, 2006.

- [7] C. Cordeiro, K. Challapali, D. Birru, and N. Sai Shankar, "IEEE 802.22: the First Worldwide Wireless Standard Based on Cognitive Radios," in *New Frontiers in Dynamic Spectrum Access Networks, 2005. DySPAN 2005. 2005 First IEEE International Symposium on*, 2005, pp. 328–337.
- [8] Q. Zhang, A. B. J. Kokkeler, and G. J. M. Smit, "An Efficient FFT for OFDM Based Cognitive Radio on a Reconfigurable Architecture," in *2007 IEEE International Conference on Communications*. Piscataway, NJ, USA: IEEE, 2008, pp. 6522–6.
- [9] Y. Kim and J. Kim, "Low Complexity FFT Schemes for Multicarrier Demodulation in OFDMA Systems," *IEICE Transactions on Communications*, vol. E90B, no. 11, pp. 3290–3293, 2007.
- [10] H. Shousheng and M. Torkelson, "Computing Partial DFT for Comb Spectrum Evaluation," *Signal Processing Letters, IEEE*, vol. 3, no. 6, pp. 173–175, 1996.
- [11] S. Pagadarai, R. Rajbanshi, G. Minden, and A. Wyglinski, "Agile Transmission Techniques," in *Cognitive Radio Transmission and Networks*, A. Wyglinski, M. Nekovee, and T. Hou, Eds. Elsevier, 2010, ch. 6.
- [12] M. O. Pun, M. Morelli, and C. J. Kuo, *Multi-Carrier Techniques for Broadband Wireless Communications*. London: Imperial College Press, 2007.
- [13] R. Rajbanshi, A. Wyglinski, and G. Minden, "OFDM-Based Cognitive Radios for Dynamic Spectrum Access Networks," in *Cognitive Wireless Communication Networks*, 2007, pp. 165–188.
- [14] K. Nolan, P. Doyle, L. Mackenzie, and D. O'Mahony, "Interference Avoidance Dynamic Adaptive OFDM using a Reconfigurable Radio Platform," in *Proceedings of the 7th IFIP International Conference on Mobile and Wireless Communication Networks (MWCN)*, Marrakech, Morocco, 2005.

- [15] U. Berthold and F. K. Jondral, "Guidelines for Designing OFDM Overlay Systems," in *New Frontiers in Dynamic Spectrum Access Networks, 2005. DySPAN 2005. 2005 First IEEE International Symposium on*, 2005, pp. 626–629.
- [16] I. Budiarjo, M. Lakshmanan, and H. Nikookar, "Cognitive Radio Dynamic Access Techniques," *Wireless Personal Communications*, vol. 45, no. 3, pp. 293–324, 2008.
- [17] J. Veilleux, P. Fortier, and S. Roy, "An FPGA Implementation of an OFDM Adaptive Modulation System," in *IEEE-NEWCAS Conference, 2005. The 3rd International*, June 2005, pp. 353–356.
- [18] J. Guffey, A. Wyglinski, and G. Minden, "Agile Radio Implementation of OFDM Physical Layer for Dynamic Spectrum Access Research," in *Global Telecommunications Conference, 2007. GLOBECOM '07. IEEE*, Pagadarai, Nov. 2007, pp. 4051–4055.
- [19] Q. Wang, L. Zhuo, V. K. Prasanna, and J. Leon, "A Multi-Mode Reconfigurable OFDM System on FPGA," in *Military and Aerospace Programmable Logic Devices (MAPLD) Conference*. Annapolis, MD: NASA Electronic Parts and Packaging (NEPP) Program, September 2008.
- [20] P. Amini, R. Kempter, R. Chen, L. Lin, and B. Farhang-Boroujeny, "Filter Bank Multitone: A Physical Layer Candidate for Cognitive Radios," in *2005 Software Defined Radio Technical Conference*, 2005.
- [21] H. Zhang, D. Le Ruyet, D. Roviras, Y. Medjahdi, and H. Sun, "Spectral Efficiency Comparison of OFDM/FBMC for Uplink Cognitive Radio Networks," *EURASIP Journal on Advances in Signal Processing*, vol. 2010, pp. 1–15, 2010.
- [22] L. Vangelista and N. Laurenti, "Efficient Implementations and Alternative Architectures for OFDM-OQAM Systems," *Communications, IEEE Transactions on*, vol. 49, no. 4, pp. 664–675, 2001.

- [23] N. Moret and A. Tonello, “Design of Orthogonal Filtered Multitone Modulation Systems and Comparison among Efficient Realizations,” *EURASIP Journal on Advances in Signal Processing*, vol. 2010, 2010.
- [24] *Logicore IP Fast Fourier Transform v8.0 Product Specification - DS808*, Xilinx, September 2010.
- [25] *TMS320C62X Assembly Benchmarks at Texas Instruments*, Texas Instruments, 2008. [Online]. Available: <http://www.ti.com/sc/docs/products/dsp/c6000/62bench.htm>
- [26] N. S. Bhatia, “A Physical Layer Implementation of Reconfigurable Radio,” Master’s thesis, Virginia Polytechnic Institute and State University, Blacksburg, VA, 2004.
- [27] J. Suris, M. Shelburne, C. Patterson, P. Athanas, J. Bowen, T. Dunham, and J. Rice, “Untethered On-The-Fly Radio Assembly With Wires-On-Demand,” in *Proceedings of the 2008 National Aerospace and Electronics Conference, NAECON 2008*, Fairborn, OH, 2008.
- [28] Federal Communications Commission, “Second Memorandum Opinion and Order - FCC 10-174,” Sep. 2010. [Online]. Available: http://hraunfoss.fcc.gov/edocs_public/attachmatch/FCC-10-174A1.pdf
- [29] M. Morelli, C. C. J. Kuo, and M. O. Pun, “Synchronization Techniques for Orthogonal Frequency Division Multiple Access (OFDMA): A Tutorial Review,” *Proceedings of the IEEE*, vol. 95, no. 7, pp. 1394–1427, 2007.
- [30] T. Weiss, J. Hillenbrand, A. Krohn, and F. K. Jondral, “Mutual Interference in OFDM-Based Spectrum Pooling Systems,” in *59th IEEE Vehicular Technology Conference*, Milan, ITALY, 2004, pp. 1873–1877.
- [31] A. J. Redfern, “Receiver window design for multicarrier communication systems,” *Selected Areas in Communications, IEEE Journal on*, vol. 20, no. 5, pp. 1029–1036, 2002.

- [32] S. Brandes, I. Cosovic, and M. Schnell, "Sidelobe Suppression in OFDM Systems by Insertion of Cancellation Carriers," in *Vehicular Technology Conference, 2005. VTC-2005-Fall. 2005 IEEE 62nd*, vol. 1, Sept. 2005, pp. 152–156.
- [33] I. Cosovic, S. Brandes, and M. Schnell, "Subcarrier Weighting: A Method for Sidelobe Suppression in OFDM Systems," *Communications Letters, IEEE*, vol. 10, no. 6, pp. 444–446, 2006.
- [34] I. Cosovic and V. Janardhanam, "Sidelobe Suppression in OFDM Systems," in *Multi-Carrier Spread-Spectrum*, 2006, pp. 473–482.
- [35] T. M. Ivan Cosovic, "Suppression of Sidelobes in OFDM Systems by Multiple-Choice Sequences," *European Transactions on Telecommunications*, vol. 17, no. 6, pp. 623–630, 2006.
- [36] M. Senst, M. Jordan, M. Dorpinghaus, M. Farber, G. Ascheid, and H. Meyr, "Joint Reduction of Peak-to-Average Power Ratio and Out-of-Band Power in OFDM Systems," in *Global Telecommunications Conference, 2007. GLOBECOM '07. IEEE*, Nov. 2007, pp. 3812–3816.
- [37] S. Pagadarai, A. M. Wyglinski, and R. Rajbanshi, "A Novel Sidelobe Suppression Technique for OFDM-Based Cognitive Radio Transmission," in *New Frontiers in Dynamic Spectrum Access Networks, 2008. DySPAN 2008. 3rd IEEE Symposium on*, 2008, pp. 1–7.
- [38] G. Bansal, M. Hossain, and V. Bhargava, "Link Adaptation in OFDM-Based Cognitive Radio Systems," in *Cognitive Wireless Communication Networks*, 2007, pp. 189–211.
- [39] f. harris, *Multirate Signal Processing for Communication Systems*. Upper Saddle River, N.J.: Prentice Hall PTR, 2004, vol. xiii, 478 p.

- [40] T. Matteo, “Windowed/Shaped OFDM and OFDM-OQAM: Alternative Multicarrier Modulations for Wireless Applications,” Master’s thesis, Università degli studi di Padova, Padova, Italy, 2005.
- [41] D. Jinfeng and S. Signell, “Classic OFDM Systems and Pulse Shaping OFDM/OQAM Systems,” KTH - Royal Institute of Technology, Stockholm, Sweden, Tech. Rep., Feb. 2007.
- [42] B. Farhang-Boroujeny and R. Kempter, “Multicarrier Communication Techniques for Spectrum Sensing and Communication in Cognitive Radios,” *Communications Magazine, IEEE*, vol. 46, no. 4, pp. 80–85, Apr. 2008.
- [43] H. Zhang, D. Le Ruyet, and M. Terre, “Spectral Efficiency Comparison Between OFDM/OQAM and OFDM-Based CR Networks,” *Wireless Communications and Mobile Computing*, vol. 9, no. 11, pp. 1487–1501, 2009.
- [44] —, “Spectral Efficiency Analysis in OFDM and OFDM/OQAM Based Cognitive Radio Networks,” in *Vehicular Technology Conference, 2009. VTC Spring 2009. IEEE 69th*, 2009, pp. 1–5.
- [45] T. Fusco, A. Petrella, and M. Tanda, “Data-Aided Symbol Timing and CFO Synchronization for Filter Bank Multicarrier Systems,” *Wireless Communications, IEEE Transactions on*, vol. 8, no. 5, pp. 2705–2715, 2009.
- [46] T. H. Stitz, T. Ihalainen, A. Viholainen, and M. Renfors, “Pilot-Based Synchronization and Equalization in Filter Bank Multicarrier Communications,” *EURASIP Journal on Advances in Signal Processing*, vol. 2010, pp. 1–19, 2010.
- [47] C. Lele, J. P. Javaudin, R. Legouable, A. Skrzypczak, and P. Siohan, “Channel Estimation Methods for Preamble-Based OFDM/OQAM Modulations,” in *Power Line Communications and Its Applications, 2007. ISPLC ’07. IEEE International Symposium on*, Paris, FRANCE, 2007, pp. 741–750.

- [48] J. Du and S. Signell, “Novel Preamble-Based Channel Estimation for OFDM/OQAM Systems,” *2009 IEEE International Conference on Communications*, pp. 1–6, Jun. 2009.
- [49] A. Tonello and F. Pecile, “A Filtered Multitone Modulation Modem for Multiuser Power Line Communications with an Efficient Implementation,” in *Power Line Communications and Its Applications, 2007. ISPLC '07. IEEE International Symposium on*, 2007, pp. pp.155–160.
- [50] S. Fahmy and L. Doyle, *Reconfigurable Polyphase Filter Bank Architecture for Spectrum Sensing*, ser. Lecture Notes in Computer Science. Berlin: Springer, 2010, vol. 5992, pp. 343–350.
- [51] S. Srikanteswara, J. H. Reed, P. Athanas, and R. Boyle, “A soft radio architecture for reconfigurable platforms,” *IEEE Communications Magazine*, vol. 38, no. 2, pp. 140–147, 2000.
- [52] J. Suris, A. Recio, and P. Athanas, “Rapid Radio: Human-Assisted Signal Classification and Receiver Implementation,” in *Software Defined Radio Forum Technical Conference*, Washington D.C., 2008.
- [53] —, “Enhancing the Productivity of Radio Designers with RapidRadio,” in *International Conference on Reconfigurable Computing and FPGAs, ReConFig 2009.*, Dec. 2009, pp. 350–355.
- [54] J. Suris, “Rapid Radio: Analysis-Based Receiver Deployment,” Ph.D. dissertation, Virginia Polytechnic Institute and State University, Blacksburg, VA, August 2009.
- [55] “CLIPS: A Tool for Building Expert Systems.” [Online]. Available: <http://clipsrules.sourceforge.net>
- [56] “GNU Radio.” [Online]. Available: <http://gnuradio.org>

- [57] C. Irick, “Enhancing GNU Radio for Hardware Accelerated Radio Design,” Master’s thesis, Virginia Polytechnic Institute and State University, Blacksburg, VA, June 2010.
- [58] Berkeley Wireless Research Center, “BEE2 - Berkeley Emulation Engine 2,” Feb. 2008. [Online]. Available: <http://bee2.eecs.berkeley.edu>
- [59] C. Chang, J. Wawrzyniek, and R. Brodersen, “BEE2: a High-End Reconfigurable Computing System,” *Design Test of Computers, IEEE*, vol. 22, no. 2, pp. 114 – 125, mar. 2005.
- [60] S. Mishra, D. Cabric, C. Chang, D. Willkomm, B. van Schewick, S. Wolisz, and B. Brodersen, in *A Real Time Cognitive Radio Testbed for Physical and Link Layer Experiments*, Nov. 2005, pp. 562 –567.
- [61] A. Tkachenko, D. Cabric, and R. Brodersen, “Cognitive Radio Experiments using Reconfigurable BEE2,” in *Signals, Systems and Computers, 2006. ACSSC ’06. Fortieth Asilomar Conference on*, oct. 2006, pp. 2041 –2045.
- [62] S. Mellers, B. Richards, H.-H. So, S. Mishra, K. Camera, P. Subrahmanyam, and R. Brodersen, “Radio Testbeds Using BEE2,” in *Signals, Systems and Computers, 2007. ACSSC 2007. Conference Record of the Forty-First Asilomar Conference on*, Nov. 2007, pp. 1991 –1995.
- [63] Z. Miljanic, I. Seskar, K. Le, and D. Raychaudhuri, “The WINLAB Network Centric Cognitive Radio Hardware Platform WiNC2R,” *Mobile Networks and Applications*, vol. 13, pp. 533–541, 2008.
- [64] C. Dick and f. harris, “FPGA Implementation of an OFDM PHY,” in *Signals, Systems and Computers, 2003. Conference Record of the Thirty-Seventh Asilomar Conference on*, vol. 1, 2003, pp. 905–909.

- [65] f. harris, C. Dick, and M. Rice, "Digital Receivers and Transmitters Using Polyphase Filter Banks for Wireless Communications," *Microwave Theory and Techniques*, vol. 51, no. 4, pp. 1395–1412, 2003.
- [66] f. harris and C. Dick, "Non Iterative Techniques for Low Peak-to-Average Power Ratio OFDM Signals," in *2009 International Waveform Diversity and Design Conference*. IEEE, Feb. 2009, pp. 296–301.
- [67] T. Pionteck, L. D. Kabulepa, and M. Glesner, "Exploring the Capabilities of Reconfigurable Hardware for OFDM-Based WLANs," in *12th International Conference on Very Large Scale Integration of System on Chip (VLSI-SoC 2003)*, M. Glesner, R. Reis, L. Indrusiak, V. Mooney, and H. Eveking, Eds., Darmstadt, GERMANY, 2003, pp. 149–164.
- [68] L. Alaus, J. Palicot, C. Roland, Y. Lout, and D. Noguet, "Promising Technique of Parameterization For Reconfigurable Radio, the Common Operators Technique: Fundamentals and Examples," *Journal of Signal Processing Systems*, pp. 1–13, 2009.
- [69] S. Saponara, N. L'Insalata, F. Tosti, L. Fanucci, and P. Terreni, "Analysis of FPGA Solutions for Baseband Processing in Multi Carrier Communication Standards for Consumer Applications," in *26th IEEE International Conference on Consumer Electronics*, Las Vegas, NV, 2008, pp. 77–78.
- [70] A. Al Ghouwayel, Y. Louet, and J. Palicot, "A Reconfigurable Architecture for the FFT Operator in a Software Radio Context," in *IEEE International Symposium on Circuits and Systems*, Kos Isl, GREECE, 2006, pp. 181–184.
- [71] M. Naoues, L. Alaus, and D. Noguet, "A Common Operator for FFT and Viterbi Algorithms," in *13th Euromicro Conference on Digital System Design, Architectures, Methods and Tools. Euromicro DSD.*, Lille, France, 2010, pp. 309–313.

- [72] J. Helmschmidt, E. Schuler, P. Rao, S. Rossi, S. di Matteo, and R. Bonitz, "Reconfigurable Signal Processing in Wireless Terminals," in *Design, Automation and Test in Europe Conference and Exhibition, 2003*, 2003, pp. 244–249 suppl.
- [73] C. Ebeling, C. Fisher, X. Guanbin, S. Manyuan, and L. Hui, "Implementing an OFDM Receiver on the RaPiD Reconfigurable Architecture," *IEEE Transactions on Computers*, vol. 53, no. 11, pp. 1436–48, 2004.
- [74] B. Mennenga, R. Guo, and G. Fettweis, "A component based reconfigurable baseband architecture," in *16th IST Mobile and Wireless Communications*, Budapest, HUNGARY, 2007, pp. 909–913.
- [75] Q. Zhang, G. J. M. Smit, L. T. Smit, A. Kokkeler, R. W. Hoeksema, and M. Heskamp, "A reconfigurable platform for cognitive radio," in *Mobile Technology, Applications and Systems, 2005 2nd International Conference on*, 2005, pp. 5 pp.–5.
- [76] A. Niktash, H. T. Parizi, and N. Bagherzadeh, "Application of a Heterogeneous Reconfigurable Architecture to OFDM Wireless Systems," in *Circuits and Systems, 2007. ISCAS 2007. IEEE International Symposium on*, 2007, pp. 2586–2589.
- [77] D. Cabric, S. M. Mishra, and R. W. Brodersen, "Implementation Issues in Spectrum Sensing for Cognitive Radios," in *38th Asilomar Conference on Signals, Systems and Computers*, M. B. Matthews, Ed., Pacific Grove, CA, 2004, pp. 772–776.
- [78] M. P. Wylie-Green, "Dynamic Spectrum Sensing by Multiband OFDM radio for Interference Mitigation," in *New Frontiers in Dynamic Spectrum Access Networks, 2005. DySPAN 2005. 2005 First IEEE International Symposium on*, 2005, pp. 619–625.
- [79] K. Kim, I. A. Akbar, K. K. Bae, J.-S. Um, C. M. Spooner, and J. H. Reed, "Cyclostationary Approaches to Signal Detection and Classification in Cognitive Radio," in *New Frontiers in Dynamic Spectrum Access Networks, 2007. DySPAN 2007. 2nd IEEE International Symposium on*, 2007, pp. 212–215.

- [80] U. Berthold and F. K. Jondral, "Distributed Detection in OFDM Based Ad Hoc Overlay Systems," in *67th IEEE Vehicular Technology Conference*, Marina Bay, SINGAPORE, 2008, pp. 1666–1670.
- [81] L. Yi, Y. Rong, and X. Shengli, "Optimal Cooperative Sensing Scheme Under Time-Varying Channel for Cognitive Radio Networks," in *New Frontiers in Dynamic Spectrum Access Networks, 2008. DySPAN 2008. 3rd IEEE Symposium on*, 2008, pp. 1–6.
- [82] T. Weiss, J. Hillenbrand, A. Krohn, and F. Jondral, "Efficient Signaling of Spectral Resources in Spectrum Pooling Systems," in *Proc. 10th Symposium on Communications and Vehicular Technology (SCVT)*, 2003.
- [83] T. Renk, C. Kloeck, and F. K. Jondral, "A Cognitive Approach to the Detection of Spectrum Holes in Wireless Networks," in *4th IEEE Consumer Communications and Networking Conference*, Las Vegas, NV, 2007, pp. 1118–1122.
- [84] W. Xiaoyu, H. Pin-Han, and A. Wong, "Towards Efficient Spectrum Sensing for Cognitive Radio through Knowledge-Based Reasoning," in *New Frontiers in Dynamic Spectrum Access Networks, 2008. DySPAN 2008. 3rd IEEE Symposium on*, 2008, pp. 1–8.
- [85] J. O. Neel, "Analysis and Design of Cognitive Radio Networks and Distributed Radio Resource Management Algorithms," Ph.D. dissertation, Virginia Polytechnic Institute and State University, Blacksburg, VA, 2006.
- [86] R. W. Thomas, "Cognitive Networks," Ph.D. dissertation, Virginia Polytechnic Institute and State University, Blacksburg, VA, 2007.
- [87] T. W. Rondeau, "Application of Artificial Intelligence to Wireless Communications," Ph.D. dissertation, Virginia Polytechnic Institute and State University, Blacksburg, VA, 2007.
- [88] T. R. Newman, B. A. Barker, A. M. Wyglinski, A. Agah, J. B. Evans, and G. J. Minden, "Cognitive Engine Implementation for Wireless Multicarrier Transceivers,"

- Wireless Communications and Mobile Computing*, vol. 7, no. 9, pp. 1129–1142, Nov. 2007.
- [89] M. Shaat and F. Bader, *Power Allocation with Interference Constraint in Multicarrier Based Cognitive Radio Systems*, ser. Lecture Notes in Electrical Engineering. Dordrecht: Springer, 2009, vol. 41, pp. 259–268.
- [90] —, “Computationally Efficient Power Allocation Algorithm in Multicarrier-Based Cognitive Radio Networks: OFDM and FBMC Systems,” *EURASIP Journal on Advances in Signal Processing*, 2010.
- [91] —, “Fair and Efficient Resource Allocation Algorithm for Uplink Multicarrier Based Cognitive Networks,” in *Proc. of IEEE Personal, Indoor and Mobile Radio Communications Symposium (PIMRC)*, Istanbul, Turkey, Sep. 2010.
- [92] A. Recio, J. Suris, and P. Athanas, “Signal Parameter Estimation for Linear Digital Modulations over Flat Fading Channels,” *IEEE Military Communications Conference - MILCOM 2009*, 2009.
- [93] R. A. Maronna, *Robust Statistics: Theory and Methods*. Chichester, England :: J. Wiley, 2006, vol. XX.
- [94] R. Shiavi, *Introduction to Applied Statistical Signal Analysis : Guide to Biomedical and Electrical Engineering Applications*, 3rd ed. Amsterdam ; Boston :: Elsevier Academic Press, 2007, vol. xxii, 402 p. :.
- [95] J. Moré, “The Levenberg-Marquardt Algorithm: Implementation and Theory,” in *Numerical Analysis*, 1978, pp. 105–116.
- [96] J. Nocedal and S. J. Wright, *Numerical Optimization*, 2nd ed., ser. Springer series in operations research. New York: Springer, 2006.
- [97] G. A. F. Seber and C. J. Wild, *Nonlinear Regression*. New York :: John Wiley and Sons, 2003.

- [98] O. A. Dobre, A. Abdi, Y. Bar-Ness, and W. Su, "Survey of Automatic Modulation Classification Techniques: Classical Approaches and New Trends," *Communications, IET*, vol. 1, no. 2, pp. 137–156, 2007.
- [99] P. Panagiotou, A. Anastasopoulos, and A. Polydoros, "Likelihood Ratio Tests for Modulation Classification," in *MILCOM 2000. 21st Century Military Communications Conference Proceedings*, A. Anastasopoulos, Ed., vol. 2, 2000, pp. 670–674 vol.2.
- [100] A. Polydoros and K. Kim, "On the Detection and Classification of Quadrature Digital Modulations in Broad-band Noise," *IEEE Transactions on Communications*, vol. 38, no. 8, pp. 1199–1211, 1990.
- [101] A. A. Tadaion, M. Derakhtian, S. Gazor, and M. Aref, "Likelihood Ratio Tests for PSK Modulation Classification in Unknown Noise Environment," in *Electrical and Computer Engineering, 2005. Canadian Conference on*, M. Derakhtian, Ed., 2005, pp. 151–154.
- [102] W. Wen and J. M. Mendel, "Maximum-likelihood Classification for Digital Amplitude-phase Modulations," *IEEE Transactions on Communications*, vol. 48, no. 2, pp. 189–193, 2000.
- [103] B. G. Mobasser, "Digital Modulation Classification using Constellation Shape," *Signal Processing*, vol. 80, no. 2, pp. 251–277, 2000.
- [104] Z. Zhijin, G. Qi, and X. Chunyun, "M-ary Digital Modulation Signals Identification Based on Fuzzy C-mean Clustering and CMHD," in *Proceedings of the Sixth IASTED International Multi-Conference on Wireless and Optical Communications*. Banff, Alta., Canada: ACTA Press, 2006.
- [105] W. Wen and J. M. Mendel, "A Fuzzy Logic Method for Modulation Classification in Nonideal Environments," *IEEE Transactions on Fuzzy Systems*, vol. 7, no. 3, pp. 333–344, 1999.

- [106] A. Fehske, J. Gaeddert, and J. H. Reed, "A New Approach to Signal Classification using Spectral Correlation and Neural Networks," in *New Frontiers in Dynamic Spectrum Access Networks, 2005. DySPAN 2005. 2005 First IEEE International Symposium on*, 2005, pp. 144–150.
- [107] Z. Yaqin, R. Guanghai, and Z. Zhi, "Modulation Recognition of SDR Receivers Based on WNN," in *Vehicular Technology Conference, 2006. VTC 2006-Spring. IEEE 63rd*, R. Guanghai, Ed., vol. 5, 2006, pp. 2140–2143.
- [108] W. A. Gardner and C. M. Spooner, "Cyclic Spectral Analysis for Signal Detection and Modulation Recognition," in *Military Communications Conference, 1988. MILCOM 88, Conference record. '21st Century Military Communications - What's Possible?'*. 1988 IEEE, C. M. Spooner, Ed., 1988, pp. 419–424 vol.2.
- [109] W. A. Gardner, "Signal Interception: A Unifying Theoretical Framework for Feature Detection," *IEEE Transactions on Communications*, vol. 36, no. 8, pp. 897–906, 1988.
- [110] C. M. Spooner and W. A. Gardner, "Robust Feature Detection for Signal Interception," *IEEE Transactions on Communications*, vol. 42, no. 5, pp. 2165–2173, 1994.
- [111] O. A. Dobre, Y. Bar-Ness, and S. Wei, "Robust QAM Modulation Classification Algorithm using Cyclic Cumulants," in *Wireless Communications and Networking Conference, 2004. WCNC. 2004 IEEE*, vol. 2, 2004, pp. 745–748 Vol.2.
- [112] G. Han, J. Li, and C. Chen, "A Robust MDPSK Modulation Classifier Based on Cumulants," *Advanced Information Networking and Applications, 2003. AINA 2003. 17th International Conference on*, pp. 265–268, 27-29 March 2003.
- [113] P. Marchand, J. L. Lacoume, and C. Le Martret, "Multiple Hypothesis Modulation Classification Based on Cyclic Cumulants of Different Orders," in *Acoustics, Speech and Signal Processing, 1998. Proceedings of the 1998 IEEE International Conference on*, J. L. Lacoume, Ed., vol. 4, 1998, pp. 2157–2160 vol.4.

- [114] J. M. Mendel, “Tutorial on Higher-Order Statistics (Spectra) in Signal Processing and System Theory: Theoretical Results and Some Applications,” *Proceedings of the IEEE*, vol. 79, no. 3, pp. 278–305, 1991.
- [115] H. Hadinejad-Mahram and I. Hero, A. O., “Robust QAM Modulation Classification via Moment Matrices,” in *Personal, Indoor and Mobile Radio Communications, 2000. PIMRC 2000. The 11th IEEE International Symposium on*, vol. 1, 2000, pp. 133–137 vol.1.
- [116] A. K. Nandi and E. E. Azzouz, “Algorithms for Automatic Modulation Recognition of Communication Signals,” *IEEE Transactions on Communications*, vol. 46, no. 4, pp. 431–436, 1998.
- [117] X. Huo and D. Donoho, “A Simple and Robust Modulation Classification Method Via Counting,” *Acoustics, Speech and Signal Processing, 1998. Proceedings of the 1998 IEEE International Conference on*, vol. 6, pp. 3289–3292 vol.6, 12-15 May 1998.
- [118] F. F. Liedtke, “Computer Simulation of an Automatic Classification Procedure for Digitally Modulated Communication Signals with Unknown Parameters,” *Signal Processing*, vol. 6, pp. 311–323, 1984.
- [119] H. Miranda, P. Pinto, and S. Silva, “A Self-Reconfigurable Receiver Architecture for Software Radio Systems,” *Radio and Wireless Conference, 2003. RAWCON '03. Proceedings*, pp. 241–244, 10-13 Aug. 2003.
- [120] B. Le, T. Rondeau, D. Maldonado, D. Scaperoth, and C. Bostian, “Signal Recognition for Cognitive Radios,” in *Software Defined Radio Forum Technical Conference*, 2006.
- [121] D. Grimaldi, S. Rapuano, and L. De Vito, “An Automatic Digital Modulation Classifier for Measurement on Telecommunication Networks,” *IEEE Transactions on Instrumentation and Measurement*, vol. 56, no. 5, pp. 1711–1720, 2007.

- [122] H. Ishii, S. Kawamura, T. Suzuki, and M. A. Kuroda, “An Adaptive Receiver Based on Software Defined Radio Techniques,” in *Personal, Indoor and Mobile Radio Communications, 2001 12th IEEE International Symposium on*, S. Kawamura, Ed., vol. 2, 2001, pp. G-120–G-124 vol.2.
- [123] D. Godard, “Passband Timing Recovery in an All-Digital Modem Receiver,” *IEEE Transactions on Communications*, vol. 26, no. 5, pp. 517–523, 1978.
- [124] M. Simon, “Noncoherent Symbol Synchronization Techniques,” *Military Communications Conference, 2005. MILCOM 2005. IEEE*, vol. 5, pp. 3321–3327, Oct. 2005.
- [125] D. W. Scott, “On Optimal and Data-Based Histograms,” *Biometrika*, vol. 66, no. 3, pp. 605–610, 1979.
- [126] R. Pawula, “Distribution of the Phase Angle Between Two Vectors Perturbed by Gaussian Noise II,” *Vehicular Technology, IEEE Transactions on*, vol. 50, no. 2, pp. 576–583, Mar 2001.
- [127] “IEEE Standard for Local and Metropolitan Area Networks Part 16: Air Interface for Fixed Broadband Wireless Access Systems IEEE Std 802.16-2004 (Revision of IEEE Std 802.16-2001),” pp. 1–857, 2004.
- [128] “3GPP TS 36.211 V8.4.0. Technical Specification,” Sept. 2008, 3rd Generation Partnership Project; Technical Specification Group Radio Access Network; Evolved Universal Terrestrial Radio Access (E-UTRA); Physical Channels and Modulation (Release 8).
- [129] J. G. Proakis and D. K. Manolakis, *Digital Signal Processing: Principles, Algorithms and Applications*, 3rd ed. Prentice Hall, 1995.
- [130] P. Fan and M. Darnell, *Sequence Design for Communications Applications*. Taunton, England: Research Studies Press, 1996.
- [131] D. Chu, “Polyphase Codes with Good Periodic Correlation Properties (Corresp.),” *Information Theory, IEEE Transactions on*, vol. 18, no. 4, pp. 531–532, 1972.

- [132] “Optimized Architecture for Computing Zadoff-Chu Sequences with Application to LTE,” in *IEEE Global Telecommunications Conference, 2009. GLOBECOM 2009.*, 2009, pp. 1–6.
- [133] S. Beyme and C. Leung, “Efficient computation of DFT of Zadoff-Chu sequences,” *Electronics Letters*, vol. 45, no. 9, p. 461, 2009.
- [134] Y. Tian, X. Lei, Y. Xiao, and S. Li, “Time Synchronization for OFDM Systems with Narrowband Interference,” in *Rough Sets and Knowledge Technology, Proceedings*, no. 6060, 2009, pp. 491–496.
- [135] L. Schwoerer, “VLSI Suitable Synchronization Algorithms and Architecture for IEEE 802.11a Physical Layer,” in *Circuits and Systems, 2002. ISCAS 2002. IEEE International Symposium on*, vol. 5, 2002, pp. 721–724.
- [136] GNU Radio, “USRP Documentation,” Nov. 2010. [Online]. Available: <http://gnuradio.org/redmine/wiki/gnuradio/UsrpFAQ>

Appendix A

Prototype Reference Manual

An operational test and demonstration setup was build using the configurable computing laboratory resources. An overview of the main hardware and software entities is presented in Figure A.1. Three tiers are identified: the upper tier is the GUI where the user commands the bitstreams that get loaded into the FPGAs, controls the assignment of the allocation vector, and modifies the characteristics of the test PU. In the second tier we find the servers running on the ARM processor of the Harris SiP. These are in charge of reading and writing information from and to the CAD bus interface of the FPGAs and routing this information to the LabView GUI via TCP sockets. The lower tier is formed by the FPGAs, that embody the transmitter and receiver physical layer logic. An Avnet Virtex-5 VSX95T board is used to generate a test PU signal. The details are presented below.

Figure A.2 shows the setup for the demonstration. The Harris SiPs appear in the upper left and right. The RF front-end components are shown in the lower left and right. A Lego robot with remotely-controllable translation and rotation of a deflector shield is present to produce impairments to the wireless channel.

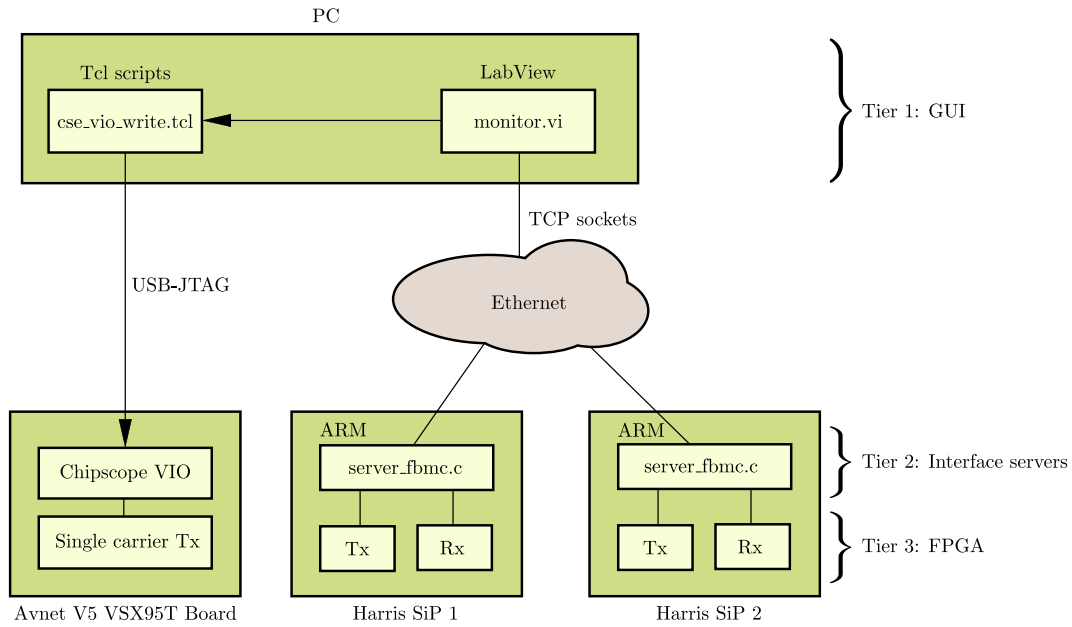


Figure A.1: Overview of the demonstration setup

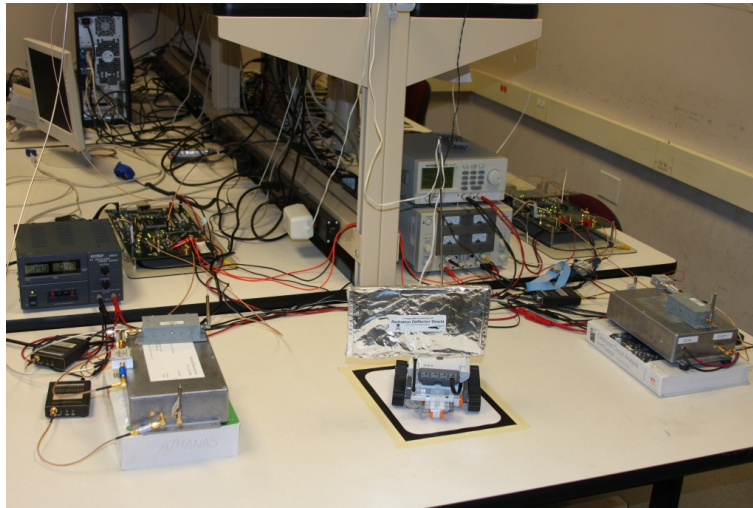


Figure A.2: Set up of the demonstration

A.1 LabView Graphical User Interface

A screen shot of the LabView GUI to control the flow of the demonstration is presented in Figure A.3. There are displays to observe the received signal characteristics at the receivers of both SiPs. The graphical information presented consists of: FFT output, channel magnitude

response, received constellation, and sensing output. The difference between the FFT output and the sensing output is that the former shows the I and Q channels out of the FFT only for the symbols that form a frame. the latter presents the energy measurement during the sensing cycle. The channel response is obtained directly from the channel estimate calculated at the peak of the second preamble symbol. Finally, the received constellation is a scaled scatter plot formed by dividing the equalized symbols by the magnitude squared of the channel response, in order to obtain a normalized amplitude. This operation is performed at the LabView vi.

Additionally, the selection of the decision threshold is left to the user, who uses the knobs to choose its value at each receiver. A logic vector is generated with the occupation vector at each receiver. This information is ORed to generate an occupation vector, presented in the plot at the lower right. The “Load Auto SC” button is provided to start the generation of the allocation vector assignment algorithm. The ORed occupation is passed to each of the SiPs, where the server assigns the pilot locations.

Figure A.4 presents the options available at each list control of the GUI. The “Tx Bitstreams” can be set separately for each SiP. The “Basic FMT Tx” produces a permanent transmission of a pseudo-random sequence, that can be controlled by the allocation vector and used to observe the spectrum of the signal. The “Full FMT Tx” bitstreams produces the operating transmit radio, that transmits a burst only when a new frame is ready to be transmitted. The “Dual 6M Tx” and the “Single 250k Tx” are legacy single carrier radio transmitters that were used to test the signal before the definitive PU generator was implemented.

A set of predefined allocation vectors can be loaded separately to the Tx and Rx FPGAs. These allocation vectors are encoded in text files that can be edited to produce distinct spectral shapes that may be required for tests where the automatic allocation vector feature is not required or desired.

The modulation type of the next frame to be transmitted is set under the option “Modulation type”. The user can choose among BPSK, QPSK, or 16-QAM.

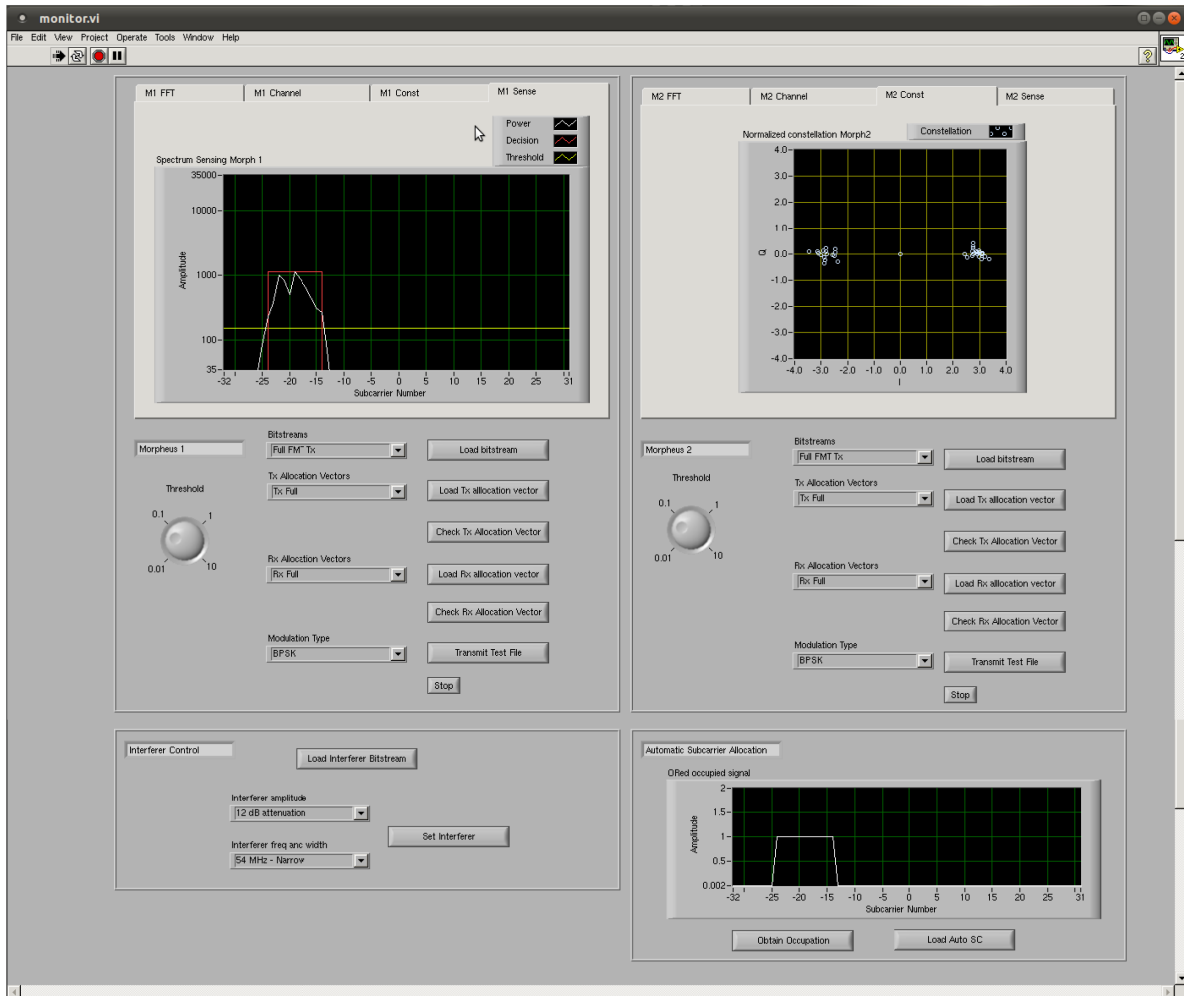


Figure A.3: The LabView GUI

The characteristics of the PU or interferer, such as amplitude, frequency, and bandwidth are also set from this interface.

A.2 Primary User Generation

A test radio design was implemented in the Avnet Virtex-5 VSX95T board. Table A.1 presents the specifications of the PU generator.

The block diagram of the PU transmitter is presented in Figure A.5,

Table A.1: Specifications of the primary user signal

Parameter	Value
Narrow signal symbol rate	2.4576 MHz
Wide signal symbol rate	4.9152 MHz
Shaping filter type	Root-raised cosine
Modulation type	BPSK
Roll-off factor	0.75
DAC sampling rate	983.04 MHz
Set of IF frequencies	54, 57, 60, 63, and 66 MHz
Set of attenuation values	0, 6, 12, 18, 24, 30, 36 dB

A.3 Project File Organization

Figure A.7 shows the organization of the directory tree organization of the project.

A.4 RF Front End

Figures A.8 and A.9 present, respectively, the RF front end for the radio transceivers based on the Harris SiP, shown in Figure A.10. A close up of one of the antennas is presented in Figure A.11.

A.5 Lego Robot

A Lego Mindstorms 2.0 robotics set was used to implement a robot that moves between the radios and rotates a deflector plate, causing variations to the channel response. The deflector can completely disrupt communications if it is put directly in front of an antenna. The robot features a color sensor that prevents it from falling from the table, and a Bluetooth wireless

connection to a PC running LabView with the Lego NXT communications plug-in. This arrangement permits the users to control the robot using a web service. A picture of the robot is presented in Figure A.12.

A.6 Instructions to Run the Demonstration

The following is a list of steps required to run the spectrum-aware reconfigurable OFDM demonstration:

1. Open terminal consoles on the Harris SiP 1 and the Harris SiP 2 using *ssh*.
2. Execute *server_fbmc* on each of them.
3. Open the LabView virtual instrument *monitor.vi*.
4. Run the LabView vi. The monitor and the sensing sockets must report on both SiPs.
5. By default, null allocation vectors are loaded to the receivers. The sensing operation occurs always that no frames are transmitted.
6. Set the PU emulation, if desired. Use the threshold knobs to set the primary user detection thresholds.
7. Use the controls to write the pre-set allocation vectors or let the system assign and write the allocation vectors assigned based on the observations of the spectrum.
8. Make sure that the allocation vectors agree between the transmitters and the receivers. Push the “Transmit file” button to transfer a file over the air.
9. At any time, the shape of the spectrum can be verified loading the “Basic FMT Tx” bitstreams and writing the desired allocation vectors. Make sure that the allocation vector written to the receivers is set to null to avoid stray frame detections. The resulting spectrum can be observed in the “Sense” tabs of the display pane.

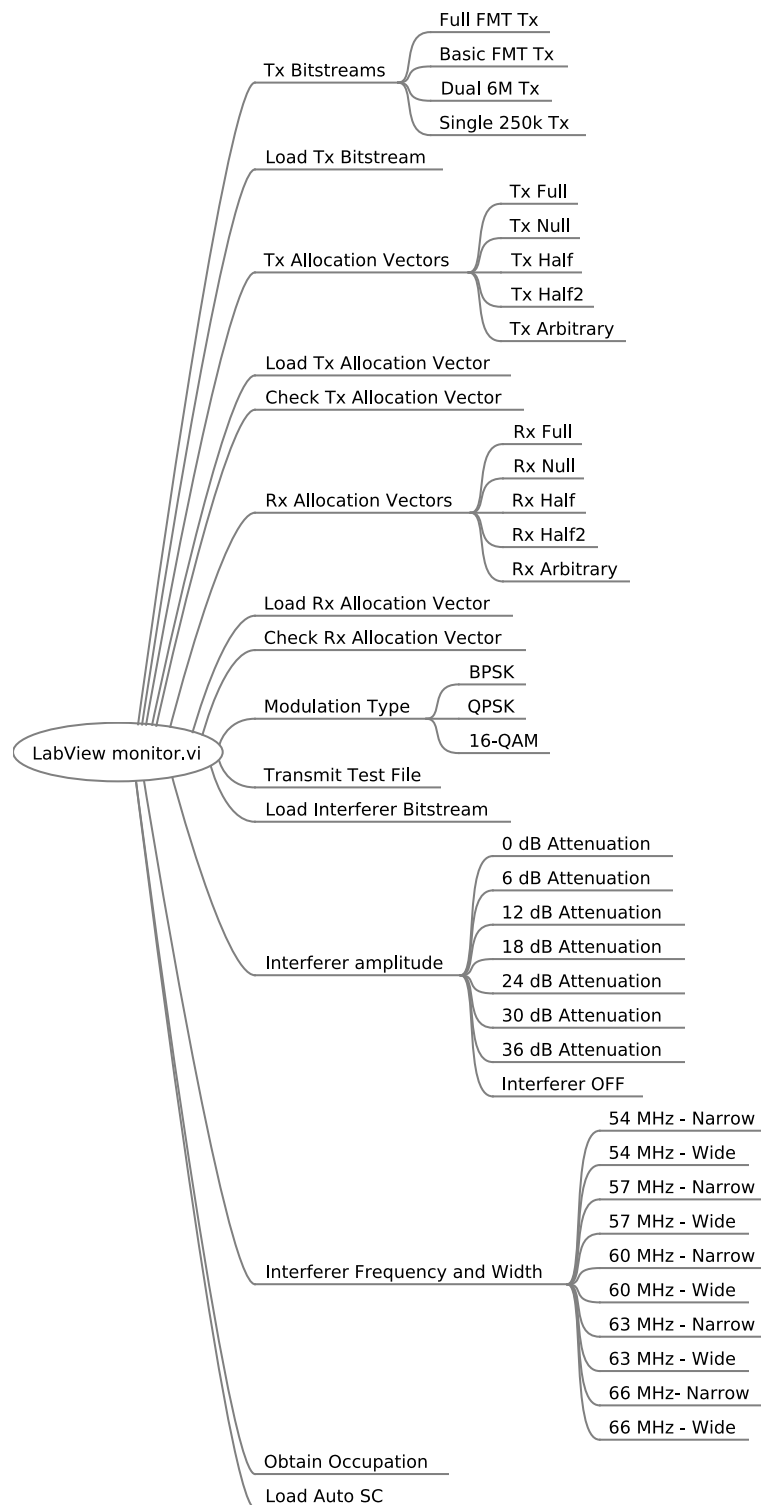


Figure A.4: Options of the LabView GUI

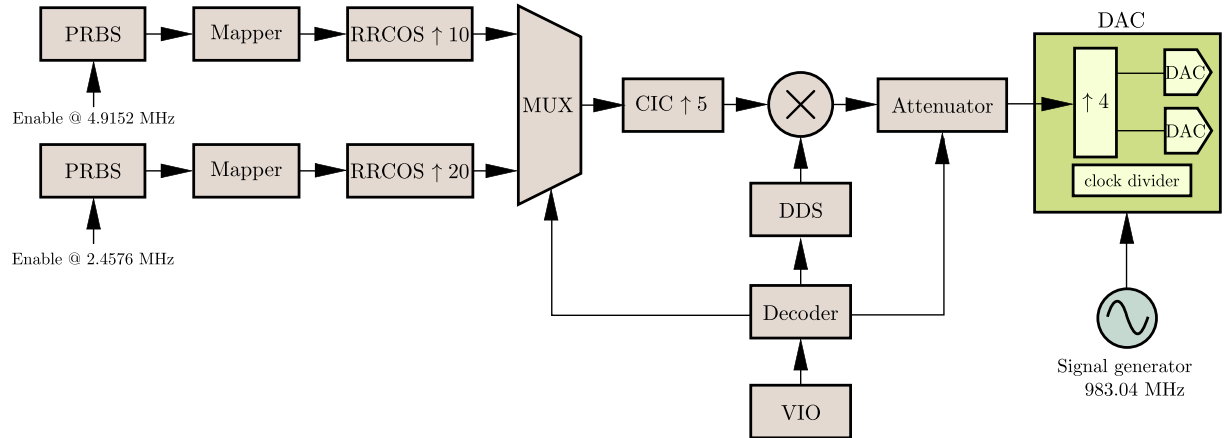


Figure A.5: The PU transmitter

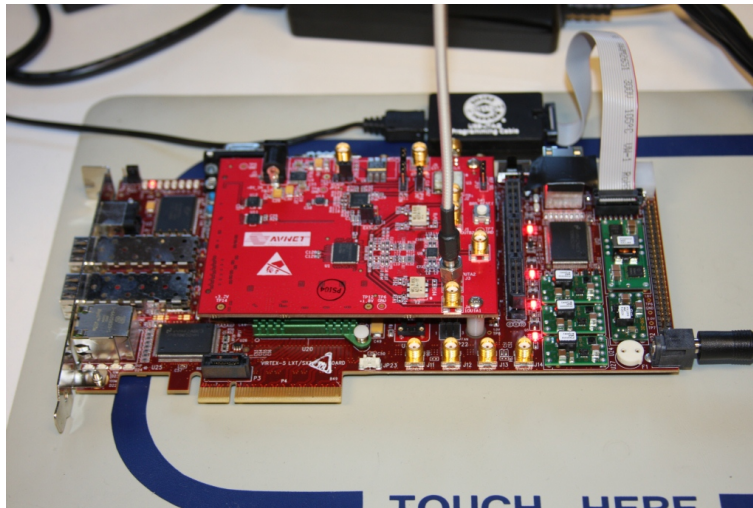


Figure A.6: The Avnet Virtex-5 VSX95T board with DAC daughter board

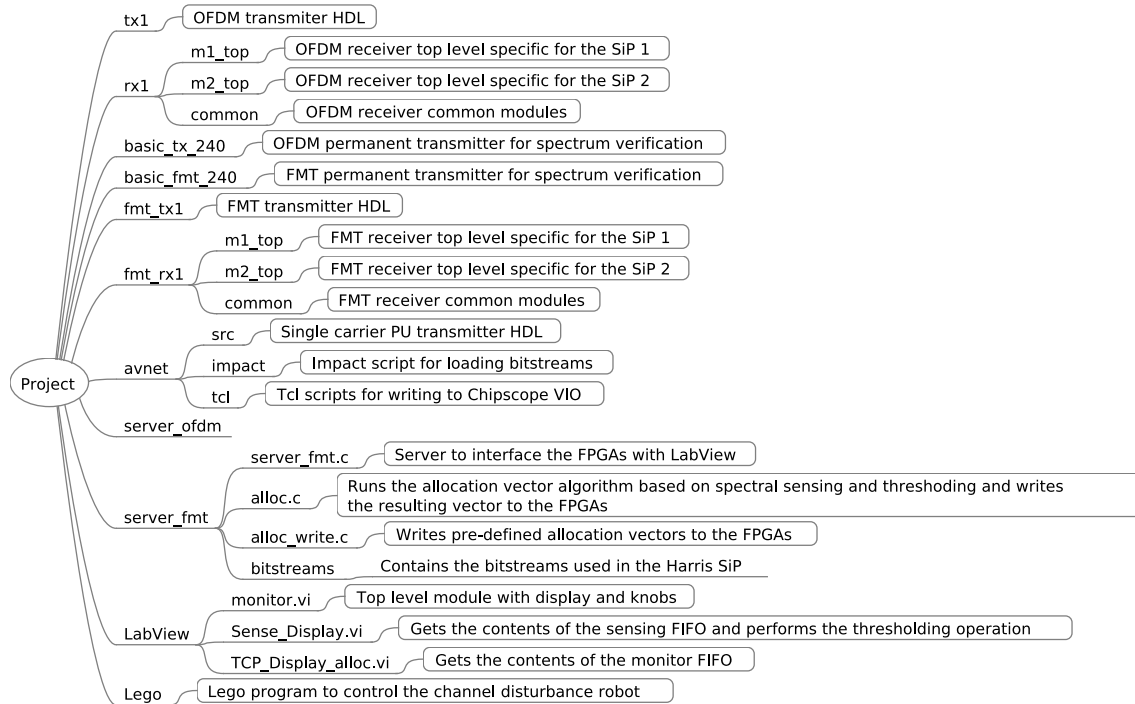


Figure A.7: Directory Organization of the project

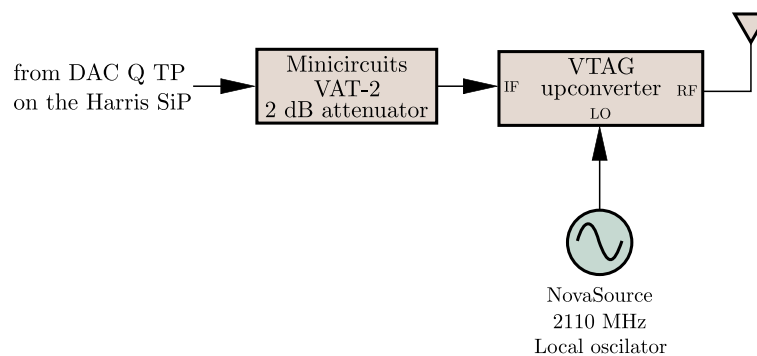


Figure A.8: Up converter

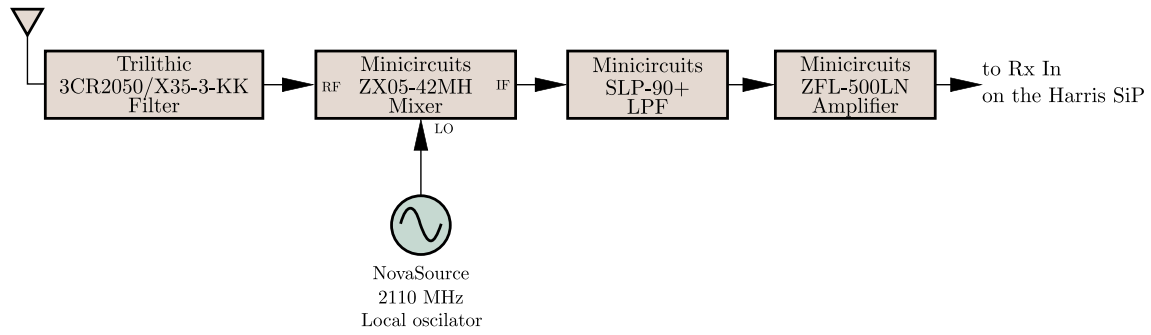


Figure A.9: Down converter

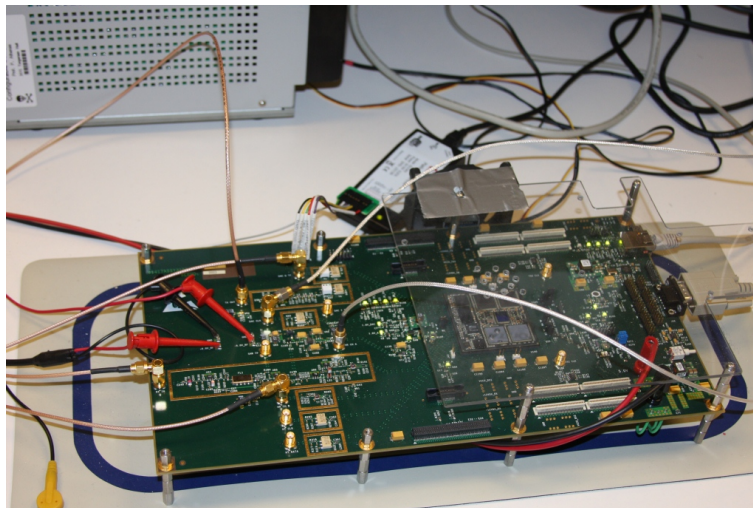


Figure A.10: The Harris SiP board

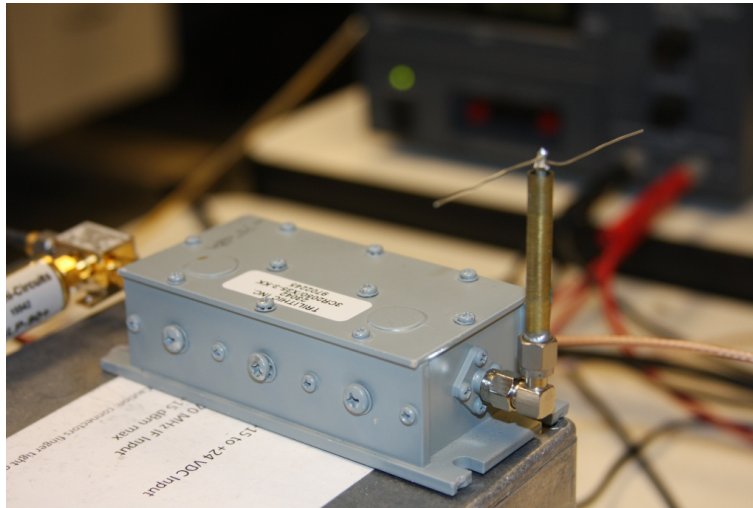


Figure A.11: A receive antenna and filter

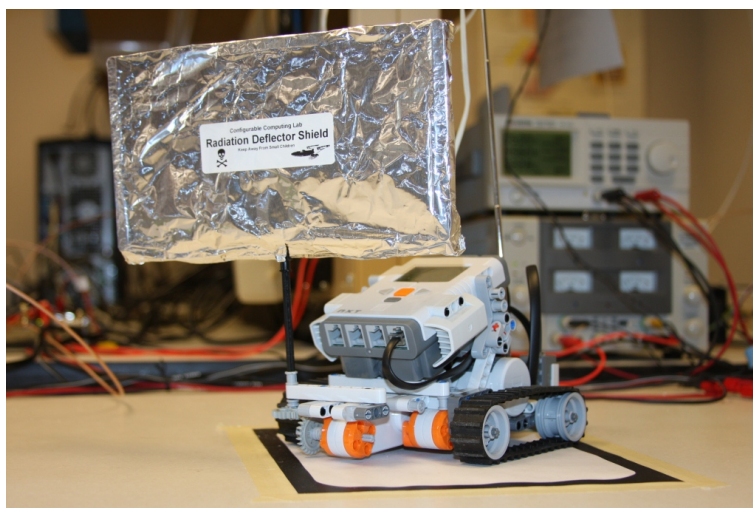


Figure A.12: Lego robot with deflection shield