REAL TIME FPGA IMPLEMENTATION OF AN AUTOMATIC MODULATION CLASSIFIER FOR ELECTRONIC WARFARE APPLICATIONS

J. Grajal, O. Yeste-Ojeda, M.A. Sanchez, M. Garrido and M. Lopez-Vallejo

ETSI Telecomunicacion, UPM
Dept. SSR and DIE
{jesus,omar,mgarrido}@gmr.ssr.upm.es and {masanchez,marisa}@die.upm.es

ABSTRACT

This paper presents an automatic modulation classifier for electronic warfare applications. It is a pattern recognition modulation classifier based on statistical features of the phase and instantaneous frequency. This classifier runs in a real time operation mode with sampling rates in excess of 1 Gsample/s. The hardware platform for this application is a Field Programmable Gate Array (FPGA). This AMC is subsidiary of a digital channelised receiver also implemented in the same platform.

1. INTRODUCTION

Automatic modulation classification (AMC) is the automatic identification of the modulation format of transmitted signals by observing the received data samples, which are corrupted by the noise and fading channels. It is an intermediate operation between the signal detection and the data demodulation. AMC plays an important role in civilian and military applications such as software-defined radio, cognitive radio, intelligent modem, dynamic spectrum management, interference identification, electronic surveillance and electronic warfare [1, 2].

In general, the *decision-theoretic* methods and the *pattern recognition* solutions are two groups of typical AMC approaches [2]. Decision-theoretic approach is based on the likelihood function [2, 3], where the modulation classification can be deemed as a multiple hypothesis test. The decision-theoretic classifiers are optimal, but the corresponding close-form solutions either are unavailable or involve a numerical search of high computational complexity. This approach is not robust with respect to the model mismatch in the presence of phase or frequency offsets, residual channel effects.

On the other hand, in pattern recognition approaches [2, 4] the modulation classifier is composed of two subsystems: a feature extraction subsystem, which extracts the key features from the received signal, and a pattern recogniser, which processes those features and determines the modulation type of the transmitted signal according to a pre-designed decision rule. The most adopted features are higher-order statistics [2, 4, 5].

In contrast to the decision-theoretic methods, pattern recognition methods may be non-optimal but are simple to implement in software, even though, their hardware implementation still poses a challenge. Following this approach, we have designed a pattern recognition modulation classifier based on statistical features of the phase and instantaneous frequency of detected radar pulses followed by a hierarchical decision tree. This classifier is intended for the classification of radar signals in electronic warfare applications. It has been implemented in a Virtex4 LX100 FPGA. This FPGA has been filled up to a 16%, what allows implementing additional signal processing modules in the remaining area.

Partly supported by TEC2008-02148 and TEC2009-08589.

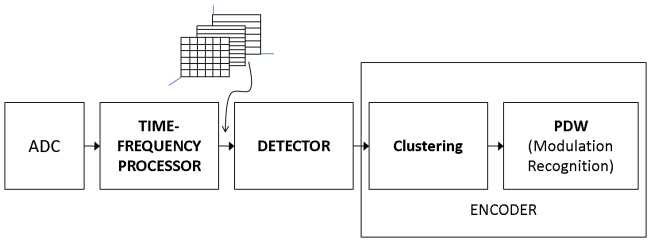

Figure 1: Digital channelised receiver for multiple signal detection and automatic modulation classification.

2. DIGITAL CHANNELISED RECEIVER WITH AMC

Modern electronic interception systems must perform the tasks of detection, classification, and identification in a difficult environment consisting of noise, interference, and multiple non-stationary signals [6]. Moreover, some waveforms are intentionally designed to reduce the probability of interception (low probability of interception, LPI, signals). This environment demands advanced signal processing algorithms running on digital receivers, which have been attracting considerable attention over the past years [5, 6]. The detector is based on a digital channelised receiver, which uses time frequency analysis before detection and encoding (Fig. 1). Time-frequency analysis [5, 7] allows simultaneous description of a signal in time and frequency, so that the temporal evolution of the signal spectrum can be analysed. It has become essential for non-stationary signal analysis; therefore, it is an appealing tool for the interception of many radar and communication signals. Our digital channelised receiver works on the time-frequency representation and builds feature vectors containing the information relative to the time and frequency where each detection occurred. The encoder clusters all the feature vectors belonging to the same signal and estimates the pulse descriptor word (PDW).

Next (see Figure 1) we identify the main features of the proposed channelised receiver. The sampling frequency of the analog-to-digital converter (ADC) is 1.3 GSamples/s with 10 bits. The time-frequency processor is based on a 128-point Fast Fourier Transform (FFT) with a Chebyshev window with a dynamic range larger than 60 dB. The overlapping of the successive FFTs is M=64. Equivalently, this

Table 1: Signal categories.

NM	No modulation
AM	Amplitude modulation
FM	Nonlinear frequency modulation
LFMa	Ascending linear frequency modulation
LFMd	Descending linear frequency modulation
BPSK	Binary phase modulation (Barker code, for example)
PM	Polyphase modulation
BPSK-AM	BPSK with amplitude modulation
PM-AM	Polyphase and amplitude modulation simultaneously

is the decimation factor after filtering with a 128-point FFT. The detector determines the sample blocks where the pulse leading and trailing edges occur. A Neyman-Pearson test is used: The amplitudes of the channels of the FFTs are compared to a threshold, which fixes the false alarm probability of the receiver. Once a pulse has been identified with its Time of Arrival (TOA), pulse width (PW), and pulse amplitude (PA), its samples are further processed to extract the features required for the AMC.

Within the encoder, clustering is needed because the signal can hop among the channels of the receiver (frequency hopping signals or chirp signals). Finally, the AMC classifies the detected signals according to their modulation, and delivers this information to the encoder to create the PDW.

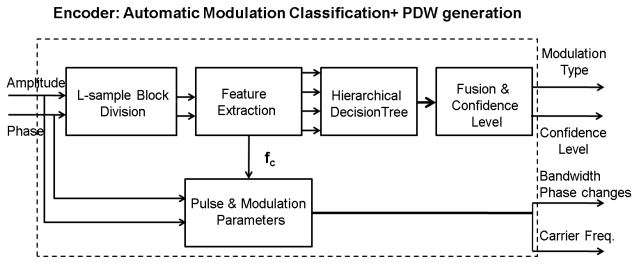

Figure 2: Block diagram of the encoder.

2.1 Automatic modulation classifier & PDW generator

Figure 2 shows the block diagram of the encoder for the generation of the PDW. Its main subsystems are the automatic modulation classifier following the pattern recognition approach [1, 4, 5] and a module for the estimation of some parameters of the pulse: carrier frequency (f_c), modulation bandwidth for frequency modulations (BW), and number of phase transitions for phase modulations. The AMC distinguishes among the nine categories shown in Table 1. These categories have been selected because they represent real signals in radar and electronic warefare applications [8].

The receiver works on a block-by-block basis, namely the input signal -amplitude and phase of the complex signal coming out from the pulse detector- is divided into blocks separately analysed by the AMC. The block-by-block operation is motivated by the concurrent implementation into FPGAs.

2.2 Features

The required features for our AMC are calculated for each sample block. These features are compared with different

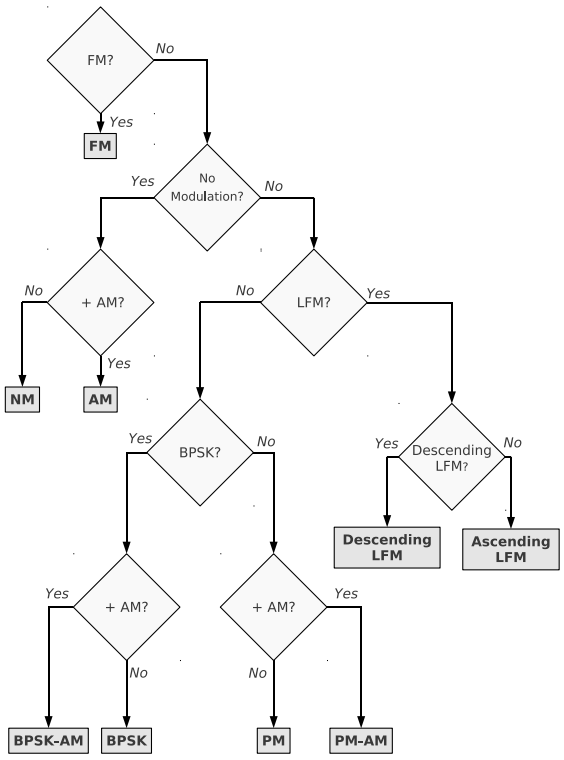

Figure 3: Hierarchical decision tree.

thresholds following the hierarchical decision tree depicted in Figure 3. All the features except γ_{AM}, which is used to detect amplitude modulation, are derived from the phase and instantaneous frequency, $f_i(n)$, obtained from the phase information supplied by the FFT. The AMC proceeds following the next steps.

Step 1. The frequency linear model is obtained by least squares for the median-filtered instantaneous frequency data. The length of the median filter is P=5. The model error, γ_{FM} helps to separate non-linear FM:

$$\gamma_{FM} = \frac{1}{L}\sum_{n=1}^{L} |f_i(n) - \hat{a}n - \hat{b}|^2 \tag{1}$$

where $\hat{a}$ and $\hat{b}$ are the coefficients of the linear regression.

Step 2. Separation of non-modulated signals. We obtained a linear model for the unwrapped input phase once the effect of the carrier frequency is subtracted.

$$\gamma_{NM} = \frac{1}{L}\sum_{n=1}^{L} |\phi_u(n) - \hat{c}n - \hat{d}|^2$$
$$\phi_u(n) = unwrap(\phi(n) - 2\pi\hat{f}_c Mn)$$

where $\hat{c}$ and $\hat{d}$ are the coefficients of the linear regression, and M is the decimation factor. The test is:

$$\gamma_{NM} \underset{\text{NM}}{\gtrless} Th_{NM} \tag{2}$$

Step 3. Separation of phase-modulated signals and LFM signals. The feature is the absolute value of the slope obtained in the linear model of Step 1, $\gamma_{LFM} = |\hat{a}|$, and the comparison is defined by:

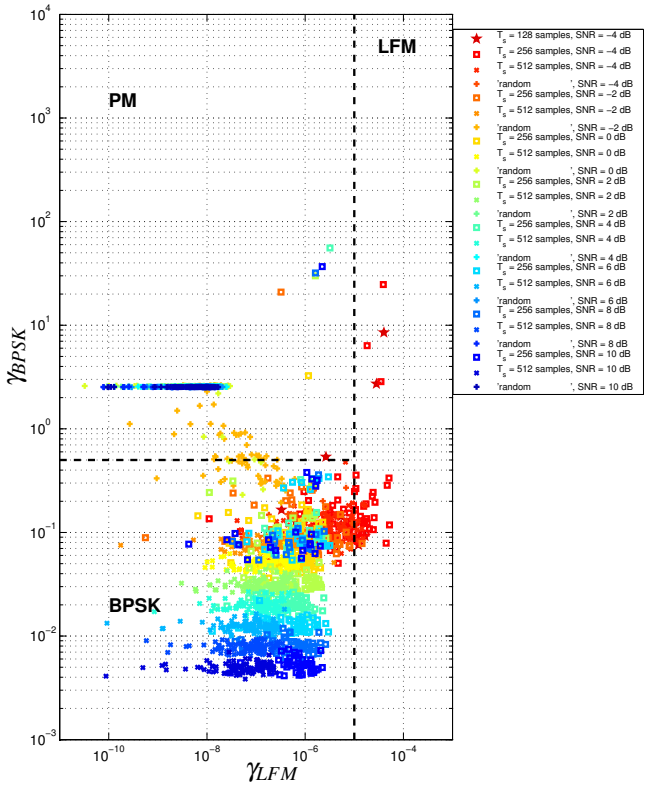

Figure 4: Statistics γ_{LFM} and γ_{BPSK}. Monte Carlo simulations of BPSK signals with 100 trials per SNR. The signal identified with label "random" corresponds to a recorded signal with BPSK modulation and 2 Mb/s.

$$\gamma_{LFM} \underset{}{\overset{\text{LFM}}{\gtrless}} Th_{LFM} \tag{3}$$

Step 4. Once the LFM signals have been grouped, the sign of the slope $\hat{a}$ is used for declaring ascending or descending LFM signals. This classifier is unable to classify correctly slow LFM signals. These signals are classified as non-modulated signals.

Step 5. Separation of BPSK signals from other phase modulated signals. If the phase of a BPSK-modulated signal is multiplied by 2, the changes in the phase will be 2π, and they will be eliminated after unwrapping.

$$\gamma_{BPSK} = \frac{1}{L}\sum_{n=1}^{L} |\phi_{u2}(n) - \hat{e}n - \hat{f}|^2$$
$$\phi_{u2}(n) = unwrap(2[\phi(n) - 2\pi\hat{f}_c Mn])$$

where $\hat{e}$ and $\hat{f}$ are the coefficients of the linear regression, and M is the decimation factor. The test is:

$$\gamma_{BPSK} \underset{\text{BPSK}}{\overset{PM}{\gtrless}} Th_{BPSK} \tag{4}$$

Step 6. The amplitude modulation of the signal is detected by calculating the variance of the amplitude normalised by the squared mean amplitude (γ_{AM}). As can be seen in figure 3, a signal can be simultaneously AM and PSK.

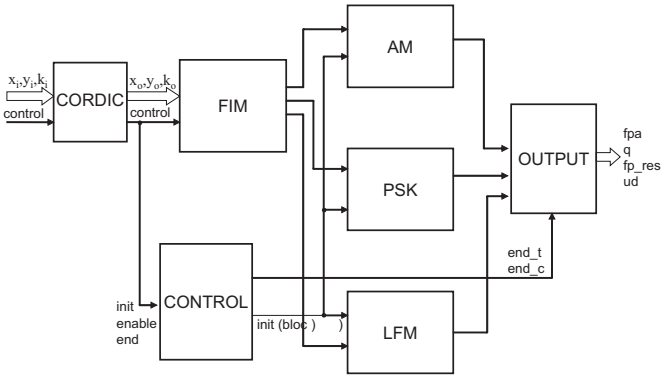

Figure 5: Hardware architecture.

2.3 Threshold calculation

The different thresholds, Th_i, are calculated by means of Monte Carlo simulations for a collection of representative signals and a broad range of Signal to Noise Ratio, SNR. Additionally, real signals recorded with the implemented digital channelised receiver were available. These signals were used both to validate the threshold selection and the AMC. Figure 4 shows the procedure for the selection of the thresholds: 100 Monte Carlo trials for each signal and SNR are analysed. The thresholds are selected to maximise the correct classification of the whole set of relevant signals for the application. Signals with low SNR are misclassified.

3. HARDWARE IMPLEMENTATION

The hardware architecture of the AMC is depicted in figure 5. The implementation is fully pipelined, what allows continuous processing just after the first sample arrives. The architecture provides real time capability for the digital channelised receiver described in section 2.

As can be seen in figure 5, the sample flow arrives to a CORDIC module that transforms cartesian coordinates to amplitude and phase required by FIM module to obtain the instantaneous frequency. This module also carries out the median filtering. Afterwards, the key features for the modulation classification can be computed in the AM, PSK and LFM modules. These modules include computationally intensive calculations such as linear regressions or FIR filtering. The output of these modules is compared against the thresholds kept in a configurable register file included in the control module. This control module sequences the block-by-block operation of the AMC. Finally, the ouput module performs the hierarchy decision tree and integrates the whole sequence results from the block partial results. Figure 6 shows the correspondece between the algorithm described in section 2 and the resulting hardware modules (surrounded by a dashed line).

Next, after the most decisive implementation issues are discussed, the main hardware modules will be described in more detail.

3.1 Implementation details

When implementing the AMC system two hard constraints were considered.

- First, to minimise the use of FPGA up to 25% of resources to allow room for other important subsystems:

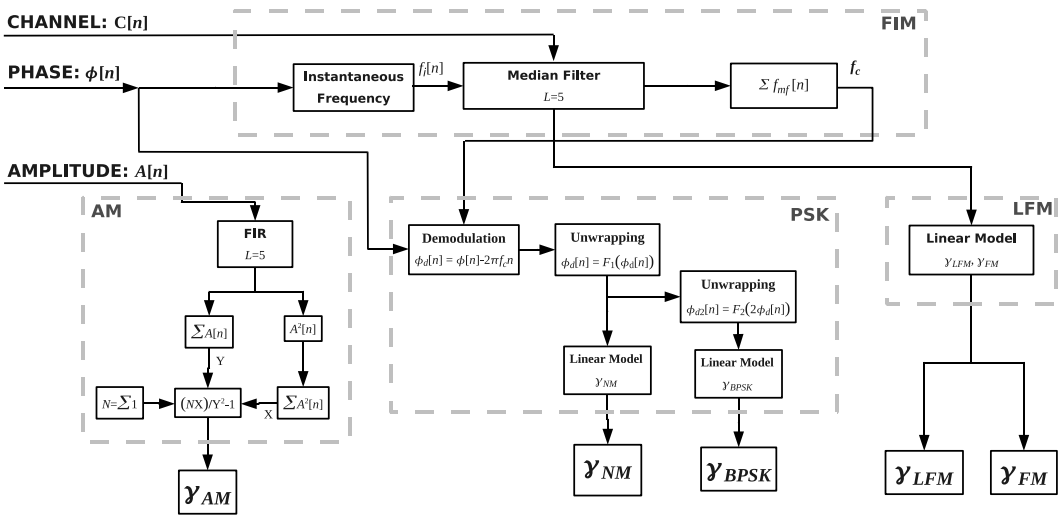

Figure 6: Identification of algorithmic modules with hardware architecture modules.

detection and communication infrastructure.

- Second, the module had to provide real time processing. As mentioned in section 2 we deal with an ADC of 1.3 Gsamples/s and the AMC module can receive a 20 Msample/s rate after the 128-FFT filtering and a 50 % overlapping.
- Finally, there is an additional latency constraint to provide the classification once the last sample arrives.

Concerning these hard constraints a key design decision was the use of block-by-block operation, what avoids any overflow even if the input sequence is arbitrarily long. Furthermore, bitwidths are always within reasonable bounds. Based on the hardware implementation we have chopped the sequence in 256 sample blocks.

To save area, before comparing threshold values against the resulting feature values (see section 2.2) we have multiplied every threshold by the resulting feature divisor to avoid the use of expensive dividers.

Next, the main characteristics of the main hardware modules that appear in figure 5 are described briefly while a special section is devoted to the linear regression module that has been designed for LFM and PSK modules.

3.1.1 CORDIC

Given that the input to this AMC system comes from an FFT we need transforming cartesian coordinates to amplitude and phase. CORDIC algorithm is the best design choice in terms of area and performance since it only occupies a small fraction of the final AMC area (see section 4 for more details).

The impact of choosing CORDIC in the final architecture is determinant. Other alternatives would require computations based on square amplitude values with the corresponding increase in bitwidths or the implementation of a square root operator. Furthermore, the calculation of the phase would require a divisor and *arctangent* operators [9].

3.1.2 FIM

Once we have amplitude and phase, the FIM module is in charge of performing fundamental operations required by the computation of AM, LFM and PSK features. More in detail, this module computes the instantaneous frequency and reduces the spikes in the instantaneous frequency through median filtering ($L = 5$).

Other important functions carried out by this module are:

- The synchronization of the input sequences all through the feature computation modules. This is accomplished by means of FIFOs.
- The computation of the number of phase jumps.
- Bandwidth computation from the instantaneous frequency after median filtering.

3.1.3 AM, LFM and PSK Modules

For the feature calculation, AM module computation is based on the use of the sequence of amplitudes while LFM and PSK computations are based on phase and frequecy information.

The AM module compares the variance of the amplitude against the threshold multiplied by squared mean amplitude, given that the variance depends on the signal power. This comparison avoids a divisor.

The core of LFM and PSK modules is the implementation of a linear regression procedure of the input data providing the values for the slope and the approximation error. The implementation of this module is based on reordering the mathematical expression what simplifies the divisions since additions are transformed into powers of N.

In the regression we calculate terms of order $O(N^6)$ what results in a significant increase in the bitwidth of intermediate variables. Thus, the number of samples is bounded to $N = 256$, which is the block size.

3.1.4 Control and Output Modules

Control is an FSM-based module in charge of splitting the sequence in blocks and controlling the sequencing of all modules. This module includes a RAM memory storing the different thresholds that can be reconfigured through an address port.

The Output module implements the decision tree based on the features provided by previous modules for each block.

Table 2: Hardware performance and resource use.

Frequency (MHz)	100,3
Latency (cicles)	110
Area (Slices, %)	7918 out of 49K, 16%
Area (DSP48S, %)	23 out of 96, 24%

Table 3: Area distribution among the different modules.

Module	Area (%)
CORDIC	6.5
FIM	5.5
AM	14
LFM	28.5
PSK	42.5
Control	1
Output	2

The decision tree has been implemented by means of a lookup-table. A single output is provided for the whole sequence upon individual block results.

4. EXPERIMENTAL RESULTS

The AMC has been implemented in a Xilinx Virtex4 LX100 FPGA. The most significant results appear in Table 2. As can be seen, the final implementation has fully exploited the use of built-in multipliers while leaving more free space in slices. More in detail, in Table 3 the percentage occupied by each sub-module is enumerated. As can be seen, the largest modules are those that include the linear regression blocks, consuming approximately 71% of the total area. It is remarkable that the CORDIC sub-module only requires a 6.5% of the final area.

The VHDL code has been highly parameterised to ease the adaptation of the implementation to different cases: bitwidth, constants, statistics, threshold and number of stages.

The implementation has been exhaustively validated at different levels (behavioral, post synthesis, post place and route and in-board) and with different benchmarks (different signals and with a broad range of SNRs). The hardware detection results have been also contrasted against the high level simulation models implemented in Matlab. As can be seen in Table 2, the FPGA has been filled up to 16% and the resulting frequency is 100 MHz (100 Msamples/s), over the real time requirements of the system. When comparing with the initial constraints (see section 3.1), the final area is below the target (25 %) and we have achieved a five time faster implementation (20 Msamples/s input rate was required).

Our AMC performance has been evaluated for a broad set of signal and SNR margins with excellent results (see Table 4). Sensitivity is defined as the SNR at the input of the time-frequency processor (see Figure 1). The main limitation of this classifier is related to the fact that the signals at the input of the AMC are decimated by the digital channelised receiver. Therefore, slow phase-modulated signals are classified as frequency modulated signals. A possible solution for this limitation could be the use of more complex decision regions or the selection of more sophisticated statistics.

It is difficult to compare with other related works because the FFT-based channelised receiver reduces the input noise and decimates the input signal to the AMC.

Table 4: AMC performance evaluation.

Signal	**Description**	**Sensit.** (dB)
FSK	$\Delta f = 5MHz$, $T_s \geq 128$ samples	-3
FSK	$\Delta f \geq 10MHz$, $T_s \geq 64$ samples	-5
CW	-	-5
LFM	$5-350MHz$, $5-20\mu s$	-5
2-4-8PSK	$T_s \geq 256$ samples	-2
Huffman 5,13,15	$T_s \geq 512$ samples	0
Frank	$T_s \geq 512$ samples	0
P4	$T_s \geq 512$ samples	-2

5. CONCLUSIONS

We have designed and implemented an automatic modulation classifier, which complements the detection task performed by a digital channelised receiver. Both detection and modulation classification tasks are carried out in real time due to an optimised implementation of this EW receiver in a FPGA platform.

The FPGA implementation of this AMC has been accomplished fulfilling hard area and performance constraints. It is a fully pipelined architecture that works in a block-by-block basis and includes efficient parallel computation for all defined modulation features. Extensive simulations and prototype tests validate the designed AMC and its corresponding hardware architecture.

Further research is in progress to analyse new features which could improve the separation of the signals, and could be used for a larger variety of signals.

REFERENCES

[1] EE Azzouz and AK Nandi. Automatic modulation recognition–I. *Journal of the Franklin Institute*, 334(2):241–273, 1997.

[2] O.A. Dobre, A. Abdi, Y. Bar-Ness, and W. Su. Survey of automatic modulation classification techniques: classical approaches and new trends. *Communications, IET*, 1(2):137–156, 2007.

[3] B. Ramkumar. Automatic modulation classification for cognitive radios using cyclic feature detection. *Circuits and Systems Magazine, IEEE*, 9(2):27–45, 2009.

[4] J. Lundén and V. Koivunen. Automatic radar waveform recognition. *IEEE Journal of Selected Topics in Signal Processing*, 1(1):124–136, 2007.

[5] G. Lopez-Risueño, J. Grajal, and A. Sanz-Osorio. Digital channelized receiver based on time-frequency analysis for signal interception. *Aerospace and Electronic Systems, IEEE Transactions on*, 41(3):879–898, 2005.

[6] J.Tsui. *"Digital Techniques for Wideband Receivers"*. Artech House, 2001.

[7] S. Qian and D. Chen. Joint time-frequency analysis. *Signal Processing Magazine, IEEE*, 16(2):52–67, 2002.

[8] N. Levanon, E. Mozeson, and Inc ebrary. *Radar signals*. Wiley Online Library, 2004.

[9] R. Gutiérrez and J. Valls. Implementation on FPGA of a LUT-Based atan (Y/X) Operator Suitable for Synchronization Algorithms. In *Field Programmable Logic and Applications, 2007. FPL 2007. International Conference on*, pages 472–475. IEEE, 2007.