

VLSI Architectures and Hardware Implementation of Ultra Low-Latency and Area-Efficient Pietra-Ricci Index Detector for Spectrum Sensing

Elivander Judas Tadeu Pereira, Dayan Adionel Guimarães, and Rahul Shrestha, *Senior Member, IEEE*

Abstract—The Pietra-Ricci index detector (PRiDe) has been recently proposed as one of the simplest techniques for centralized, data-fusion cooperative spectrum sensing, attaining robustness against time-varying signal and noise levels, constant false alarm rate, and high detection power. In this paper, we propose the design and implementation of the PRiDe detector, targeting field programmable gate array (FPGA) and application-specific integrated circuit (ASIC) solutions. Novel approaches are proposed for computing the PRiDe's test statistic, including the absolute value of complex quantities, the complex multiplier-accumulator, and the spectrum occupancy decision. The absolute value operation, which is critical to the PRiDe test statistic computational cost, applies the coordinate rotation digital computer (CORDIC) algorithm as a low latency and resource-efficient option. Register transfer level (RTL) and Monte Carlo simulations show that the resulting ultra-low latency PRiDe detector architectures attain no performance loss with respect to floating-point simulations. One of the two proposed ASIC design versions of the PRiDe sensor occupies 34.9% lower area compared to the most area-efficient sensor reported in literature, whereas the other one is $5.7\times$ faster than the fastest state-of-the-art sensor. In a nutshell, the proposed detector architecture delivers the highest area and power efficiencies, considering the scaled values of area-time product (ATP) and power-delay product (PDP) metrics, in comparison to implementations reported to date.

Keywords—Cognitive radio, coordinate rotation digital computer, field programmable gate array, application-specific integrated circuit, Pietra-Ricci index detector, spectrum sensing.

I. INTRODUCTION

THE radio frequency (RF) spectrum scarcity in the ultra-high frequency (UHF), L, S, and C bands, which comprise frequencies from a few hundred MHz to around 8 GHz, is the major bottleneck for the deployment of new wireless communication technologies [1]. This range of frequencies hosts many wireless communication services (e.g., TV, mobile communications, aeronautics communications, and military applications), causing spectrum congestion.

Elivander J. T. Pereira and Dayan A. Guimarães are with the National Institute of Telecommunications (*Instituto Nacional de Telecomunicações*, Inatel), Santa Rita do Sapucaí, MG, Brazil. Phone:+55(35)34719227, e-mail: elivander@mtel.inatel.br, dayan@inatel.br. Rahul Shrestha is with the School of Computing and Electrical Engineering, IIT Mandi, Mandi 175005, India. e-mail: rahul_shrestha@iitmandi.ac.in.

This work was partially supported by RNP, with resources from MCTIC, Grant 01245.020548/2021-07, under the 'Brazil 6G Project' of the Radio-communication Reference Center of Inatel; by Huawei, Grant PA6001BRA23032110257684, under the project 'Advanced Academic Education in Telecommunications Networks and Systems'; by EMBRAPA-Inatel Competence Center on 5G and 6G Networks, with resources from the PPI IoT/Manufatura 4.0 from MCTI, under the project 'Proof of Concept of an IoT Spectrum Sensor', Grant 052/2023; and by CNPq, Grant 302589/2021-0.

While mobile communication services demand progressively more spectrum to support their growth of users and applications, most broadcasting services are changing to streaming through the internet and replacing traditional TV and radio systems. Nonetheless, the fixed RF spectrum allocation policy used by regulators led to an inefficient utilization of this resource [2].

An alternative to mitigate the RF spectrum shortage problem is the dynamic spectrum access (DSA) [3], in which a secondary user (SU) can share the spectrum with a primary user (PU), which is the licensed user of a given band in the fixed allocation policy. The DSA solutions are related to the concept of cognitive radio (CR) [4]. A CR is a device that knows the radio environment and uses this knowledge to change its operating parameters (e.g., frequency, modulation, coding and transmission power), yet avoiding interference to the PU's transmissions.

A key enabling technique for the deployment of CR technology is spectrum sensing, which applies signal processing operations to detect the occupancy of a spectrum band by PUs, allowing the CR-enabled SU to opportunistically access this band when it is vacant. Stand-alone spectrum sensing refers to the technique that is performed independently by each SU. It carries simple implementation but suffers from performance degradation under shadowing, multipath fading and hidden terminals. To achieve better performance, cooperative spectrum sensing (CSS) is the preferred solution. It applies multiple SUs that jointly perform spectrum sensing, subsequently forwarding the sensing information to a fusion center (FC), where the information is processed to allow for a global decision on the spectrum occupancy state. The CSS outperforms the stand-alone spectrum sensing due to the spatial diversity attained by multiple SUs in different locations.

Spectrum sensing is a binary hypothesis test in which $\mathcal{H}_1$ denotes the presence of a PU signal in the sensed band, whereas $\mathcal{H}_0$ denotes the absence of a PU signal. The test is made by comparing a test statistic T , which is formed directly or indirectly from the received signal samples, with a decision threshold λ . The decision is in favor of $\mathcal{H}_1$ if $T > \lambda$; otherwise, the decision favors $\mathcal{H}_0$. The spectrum sensing performance is often assessed by means of the probability of detection, P_d , and the probability of false alarm, P_{fa} . The former is the probability of deciding that the PU signal is present in the sensed band, given that it is indeed present. The latter is the probability of declaring that the PU signal is present, when it is in fact absent. The performance is mostly impacted by the

test statistic employed. A multitude of test statistics has been reported in the literature, as exemplified subsequently.

A. Related works

Semi-blind detectors require information about the PU signal characteristics, the noise variance or both to determine the test statistic. Detectors that do not depend on knowing neither the PU signal nor the noise variance are commonly referred to as blind. Among the large number of test statistics reported in the literature, examples of semi-blind detectors are the energy detector (ED) [5] and the maximum eigenvalue detector (MED), also known as Roy's largest root test (RLRT) [6]. Examples of blind detectors are the eigenvalue-based generalized likelihood ratio test (GLRT) [7], the maximum-minimum eigenvalue detector (MMED) [8], the arithmetic to geometric mean (AGM) detector [9], the Hadamard ratio (HR) detector [10], the volume-based detectors (VD) [11], the Gerschgorin radii and centers ratio (GRCR) detector [12], the Gini index detector (GID) [13], and the PRIDe [14].

The ED test statistic has the lowest implementation complexity among all detectors. However, in practice, its overall complexity unveils to be high due to the fact that the noise variance needs to be estimated [15]. The test statistics of the MED, MMED, AGM and GLRT are based on the eigenvalues of the sample covariance matrix (SCM) of the received signal, resulting in high implementation complexity due to the need of eigenvalue estimation. The HR detector applies the determinant of the SCM, which also incurs a high computation cost. On the other hand, the test statistics of the GID, GRCR and PRIDe operate directly on the elements of the SCM, resulting in relatively lower implementation complexities. The PRIDe deserves special attention, not only for being an state-of-the-art detector, but also due to its attractive attributes of blindness, robustness against time-varying signal and noise levels, constant false alarm rate (CFAR), and high detection performance.

As far as the hardware implementation of blind detectors for cooperative spectrum sensing is concerned, a few initiatives can be found in the literature. In [16] and [17], data-fusion based cooperative spectrum sensors have been implemented under the GLRT paradigm. In [18], the sensor has been implemented with an MED/MME reconfigurable architecture. These implementations have similar complexities, as all of them are eigenvalue based designs. However, the computations of the eigenvalues are different among them: [16] and [18] use the iterative power method algorithm, whereas [17] applies the iterative Cholesky decomposition. More algorithms for the eigenvalue computation problem can be found in [19] and references therein, but no matter the algorithm used, these extra steps add more complexity and latency compared to a detector that depends only on the SCM computation. The implementation reported in [20] addresses the GRCR test statistic, whose complexity is similar to the PRIDe. In this case, the test statistic does not rely on eigenvalues, but its complexity is increased due to the computation of the magnitude of the SCM entries. So far, the PRIDe has not been considered for hardware implementation.

B. Contributions and organization of the article

The main contribution of the work reported in this article is the design and implementation of an ultra-low latency PRIDe sensor architecture, targeting FPGA and ASIC solutions. The proposed architecture is suitable for ordinary spectrum sensing, but is especially suitable for those applications that demand fast sensing, for example to scan a wide frequency band in a short time, to fasten the sliding-window approach [21] for detecting pulse radar signals, or simply to reduce the overall sensing time aiming at increasing the secondary network data throughput.

This work also details the complete development flow, from the conception and evaluation of candidate architectures and definition of the best architectures for composing each module, to the design, synthesis and simulation of the hardware for FPGA and ASIC solutions for the PRIDe sensor.

Other contributions related to the design and signal processing operations of the PRIDe test statistic are:

- the analysis of fixed-point operation and word length in the spectrum sensing performance;
- analysis of different hardware architectures to construct resource-efficient designs of multiply-accumulate units;
- a CORDIC-based magnitude computation unit with resource-efficient design and reduced latency;
- a divider-free implementation avoiding division to compute the SCM values and the test statistic.

The remainder of the paper is organized as follows: Section II introduces the centralized data-fusion cooperative spectrum sensing model, presents the PRIDe test statistic, and addresses spectrum sensing performance comparisons among state-of-the-art detectors. Section III discusses the conventional and proposed hardware architectures for the PRIDe sensor implementation. The results obtained through simulations and synthesis of the proposed PRIDe sensor are addressed in Section IV. The conclusions are reported in Section V.

II. CSS MODEL, PRIDE TEST STATISTIC, AND CSS PERFORMANCE COMPARISONS

A. CSS basic model and PRIDe test statistic

The basic model for centralized CSS with data fusion model comprises m SUs, each collecting N samples of the PU signal during each sensing interval. At the FC, the samples collected by all SUs form the matrix $\mathbf{Y} \in \mathbb{C}^{m \times N}$, which is given by

$$\mathbf{Y} = \mathbf{h}\mathbf{x}^T + \mathbf{V}, \quad (1)$$

where the vector $\mathbf{x} \in \mathbb{C}^{N \times 1}$ contains the samples associated to the PU signal, which are zero-mean complex Gaussian random variables whose variance is determined according to the average signal-to-noise ratio (SNR) across the SUs. The channel vector $\mathbf{h} \in \mathbb{C}^{m \times 1}$ is formed by elements h_i , $i = 1, 2, \dots, m$, that represent the channel gains between the PU transmitter and the i -th SU. These gains are constant during the sensing interval and independent and identically distributed (i.i.d.) over the sensing rounds. The matrix $\mathbf{V} \in \mathbb{C}^{m \times N}$ in (1) is formed by i.i.d. Gaussian noise samples with zero mean and SNR-dependent noise variance.

After the matrix $\mathbf{Y}$ defined in (1) is formed at the FC, the SCM of order m is computed as

$$\mathbf{R} = \frac{1}{N} \mathbf{Y} \mathbf{Y}^\dagger, \quad (2)$$

where $\dagger$ denotes the complex conjugate and transpose. Under the hypothesis $\mathcal{H}_0$, it follows that $\mathbf{Y} = \mathbf{V}$. Under $\mathcal{H}_1$, $\mathbf{Y} = \mathbf{h} \mathbf{x}^T + \mathbf{V}$.

Let r_i denote the i -th element of the vector formed by stacking all columns of $\mathbf{R}$, and let

$$\bar{r} = \frac{1}{m^2} \sum_{i=1}^{m^2} r_i, \quad (3)$$

The PRIDe test statistic defined in [14] is

$$T_{\text{PRIDe}} = \frac{\sum_{i=1}^{m^2} |r_i|}{\sum_{i=1}^{m^2} |r_i - \bar{r}|}. \quad (4)$$

The computation of the SCM at the FC becomes an intensive task as the number of SUs (m) performing CSS increases. This is owed to the fact that the numbers of combinational and sequential hardware resources in FPGA and ASIC platforms that deliver parallel processing scale non-linearly with m . Employing a conventional serial processing architecture may solve the hardware resource limitation, but may result in a large latency that can be prohibitive to the spectrum sensing task.

B. Spectrum sensing performance

The basic model described by (1) is enhanced in [14], taking into account typical sensing channel characteristics found in the real world, namely: the combination of Rician fading and thermal noise, the temporal variation of received signal and noise powers across the spectrum sensors, and the time-varying condition of the line-of-sight between the primary transmitter and mobile sensors. Under such an enhanced model, several performance results are presented and discussed in [14], highlighting the superiority of the PRIDe over state-of-the-art detectors in a variety of circumstances.

In this subsection, to complement the results given in [14] and to support the choice of the PRIDe for hardware implementation, its performance is contrasted with the performances attained by the blind detectors listed in Section I-A, namely: GID, HR, AGM, VD number 1 (VD1), MMED, eigenvalue-based GLRT, and GRCR. Their test statistics are given in Table I, where $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_m$ are the eigenvalues of $\mathbf{R}$,

$\det(\mathbf{R})$ is the determinant of $\mathbf{R}$, $r_{i,j}$ is the element in the i -th row and j -th column of $\mathbf{R}$, and $\mathbf{E} = \text{diag}(\mathbf{d})$, where $\text{diag}(\mathbf{d})$ is the diagonal matrix whose main diagonal corresponds to the vector $\mathbf{d} = [d_1, d_2, \dots, d_m]$, with $d_i = \|\mathbf{R}(i, :)\|_2$ and $\|\cdot\|_2$ denoting the Euclidean norm.

In the enhanced model described in [14], whose details are omitted here for conciseness, fractions $0 \leq \rho_N < 1$ and $0 \leq \rho_S < 1$ are defined to set the uniformly-distributed variations of the noise and received signal powers about their averages, respectively. The Rice factor of the channels between the PU transmitter and the SU's receivers is a Gaussian random variable whose mean, μ_K , and standard deviation, σ_K , depend on the morphology of the environment (urban, rural or suburban), according to [22].

The performance results shown hereafter give the probability of detection, P_d , as a function of the most relevant CSS system parameters (ρ_N , ρ_S , SNR, N , m and μ_K), for a CFAR $P_{fa} = 0.1$ [23], and channel parameters characterizing a suburban area [22], for which $\mu_K = 2.63$ dB and $\sigma_K = 3.82$ dB. Each point on all curves has been determined from 50000 Monte Carlo computer simulation runs, which corresponds to the same amount of spectrum sensing rounds, using Matlab. Unless otherwise stated, the default parameter values are $m = 4$ SUs, $N = 100$ samples, SNR = -11 dB, fractions of signal and noise power variations: $\rho_S = 0.9$ and $\rho_N = 0.5\rho_S$, and mean and standard deviation of the Rice factor: $\mu_K = 2.63$ dB and $\sigma_K = 3.82$ dB. In some cases, a few parameters have been set to keep $P_d \approx 0.9$ around the mid-value of the CSS parameter varied, for the best detector in each case. In this way, it can be easily seen the influence, on P_d , of parameters below and above their mid-values.

Fig. 1(a) gives P_d versus the fraction ρ_S that governs the received signal power variations, for $N = 400$ samples. Fig. 1(b) shows P_d versus the SNR across the SUs, for $N = 300$ samples. Fig. 1(c) gives P_d versus the number of samples, N . Fig. 1(d) depicts the influence of the number of SUs, m , on P_d , for SNR = -9.5 dB. Finally, Fig. 1(e) shows P_d versus the mean Rice factor μ_K , for $N = 280$ samples.

The variation patterns of P_d in all graphs in Fig. 1 are consistent with the patterns reported in [14]. Moreover, the graphs show the superiority of the PRIDe for a variety of system parameters; its superiority in other circumstances and for other system parameters can be inferred from the large amount of results reported in [14].

Special attention must be directed to Fig. 1(a), from where it can be seen that the PRIDe is robust against variations of the received signal and noise levels at the SUs, the same happening with the detectors GID, HR, VD1 and GRCR. The detectors GLRT (eigenvalue-based), MMED and AGM are not robust. Fig. 1(e) also deserves attention, since it shows how the level of a dominant multipath signal component affects the performance of the detectors. The PRIDe once again attains attractive performances, for moderate-to-high Rice factors, and the GID becomes quite attractive in strong line-of-sight condition.

Hence, the choice of the PRIDe for hardware implementation is well-supported by its attractive performance and low computation complexity.

TABLE I: Competing test statistics

$T_{\text{GID}} = \frac{\sum_{i=1}^{m^2} r_i }{\sum_{i=1}^{m^2} \sum_{j=1}^{m^2} r_i - r_j }$	$T_{\text{AGM}} = \frac{\frac{1}{m} \sum_{i=1}^m \lambda_i}{(\prod_{i=1}^m \lambda_i)^{1/m}}$
$T_{\text{HR}} = \frac{\det(\mathbf{R})}{\prod_{i=1}^m r_{i,i}}$	$T_{\text{GLRT}} = \frac{\lambda_1}{\sum_{i=1}^m \lambda_i}$
$T_{\text{VD1}} = \log [\det(\mathbf{E}^{-1} \mathbf{R})]$	$T_{\text{MMED}} = \frac{\lambda_1}{\lambda_m}$
$T_{\text{GRCR}} = \frac{\sum_{i=1}^m \sum_{j=1, j \neq i}^m r_{i,j} }{\sum_{i=1}^m r_{i,i}}$	

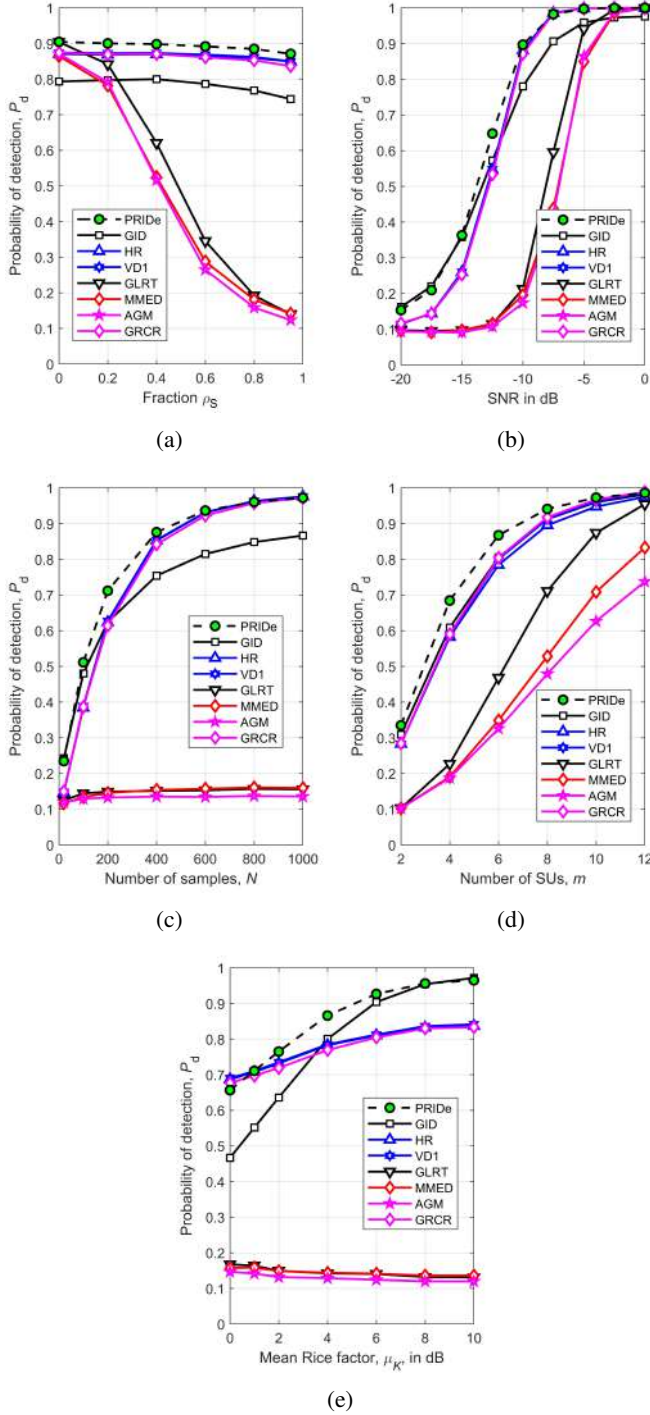


Fig. 1: P_d versus CSS system parameters. Unless otherwise mentioned, $m = 4$, $N = 100$, $\text{SNR} = -11$ dB, $\rho_S = 0.9$, $\rho_N = 0.45$, $\mu_K = 2.63$ dB and $\sigma_K = 3.82$ dB. (a) P_d versus ρ_S for $N = 400$ instead of $N = 100$; (b) P_d versus SNR for $N = 300$ instead of $N = 100$; (c) P_d versus N ; (d) P_d versus m for $\text{SNR} = -9.5$ dB instead of $\text{SNR} = -11$ dB; (e) P_d versus μ_K for $N = 280$ instead of $N = 100$.

III. PROPOSED HARDWARE ARCHITECTURES

This section presents the proposed hardware architectures of the PRIDe sensor device. They are composed of a main generic module for the sample covariance matrix computation, a test

statistic computation module, a decision-making module, and a module for magnitude computation based on the CORDIC algorithm.

A. Sample covariance matrix computation module

At the FC, the n -th sample received from the i -th SU, for $n = 1, \dots, N$ and $i = 1, \dots, m$, corresponds to the element $y_{i,n}$ of the matrix $\mathbf{Y}$ given in (1), which is processed via (2) to yield the SCM $\mathbf{R}$. The element in the i -th row and k -th column of $\mathbf{R}$, for $i, k = 1, \dots, m$, is given by

$$r_{i,k} = \frac{1}{N} \sum_{n=1}^N y_{i,n} y_{k,n}^*, \quad (5)$$

which clearly involves the multiplication of possibly complex-valued quantities. For notation simplicity, let c denote the result of the multiplication between a complex number a and a complex conjugate number b^* . The real and imaginary parts of c are, respectively,

$$\Re(c) = \Re(a)\Re(b) + \Im(a)\Im(b), \quad (6)$$

$$\Im(c) = \Im(a)\Re(b) - \Re(a)\Im(b). \quad (7)$$

This is the natural approach to perform a complex conjugate multiplication in hardware level. It requires 4 real-valued multipliers and 2 real-valued adders (additions and subtractions are treated hereafter just as additions). Consider now the alternative approach

$$\Re(c) = \Re(a)[\Re(b) - \Im(b)] + \Im(b)[\Re(a) + \Im(a)], \quad (8)$$

$$\Im(c) = \Re(a)[\Re(b) - \Im(b)] + \Re(b)[\Im(a) - \Re(a)], \quad (9)$$

in which the number of multipliers has been reduced from 4 to 3 with respect to (6) and (7), owed to the fact that the term $\Re(a)[\Re(b) - \Im(b)]$ is common to (8) and (9). Nonetheless, in (8) and (9) the number of adders has been increased from 2 to 5 with respect to (6) and (7) [24], [25].

It is a well-known fact that a multiplier represents a high resource-consuming unit, and the reduction in the number of multipliers is, then, of paramount relevance to the optimization of hardware designs. To compute an entry $r_{i,k}$ of $\mathbf{R}$ as given in (5), the approach given in (6) and (7), or the one given in (8) and (9) is typically adopted. Here, three multiply-accumulate (MAC) hardware architectures can be devised as options to build the SCM computation module, as shown in Fig. 2¹. Architecture I, which is depicted in Fig. 2a, is the direct implementation of (6) and (7), without simplifications. Architecture II, which is shown in Fig. 2b, is based on (8) and (9), making use of associations to reduce the number of multipliers. Architecture III, which is shown in Fig. 2c, is proposed here as an alternative. It is based on the same equations applied in Architecture I, but reusing the multipliers in two different clock cycles. This architecture halves the

¹In Fig. 2 and similar ones presented throughout the text, the number of bits (word length) used to represent a given input value or result is placed close to the corresponding line carrying the quantity, and is denoted by the number of bits followed by the lowercase letter 'b'.

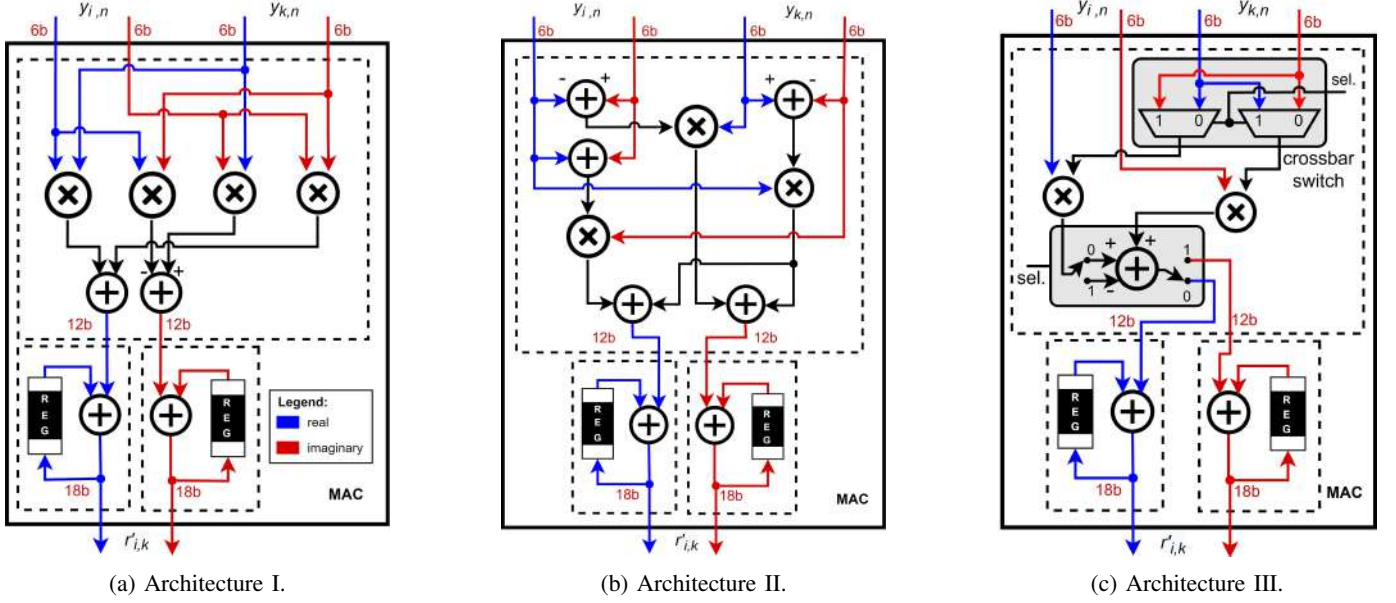


Fig. 2: MAC unit architectures.

number of real multipliers in comparison with Architecture I, with the help of a crossbar switch [26]. The crossbar switch, whose micro-architecture is indicated inside Fig. 2c, allows the inputs to be redirected to the desired outputs according to a selection signal (sel.). Hence, the four multiplications in (6) and (7) can be calculated using only two multipliers. Finally, the selective add-subtract block performs the addition or subtraction operations according to its selector input, routing the computed value to the real or imaginary accumulator. The selection signal from architecture III, presented in Fig. 2c, is an external port of the MAC unit and it is internally shared by the crossbar switch and the add-subtract block.

The lower part of a MAC architecture computes the cumulative sum of multiplication results, processed in the upper part. This yields an element $r'_{i,k}$ of the SCM, that is not divided by N in this step. The division by N is made after a combination of MAC results, as explained in the sequel.

Likewise [20], the SCM computation module considered herein is an array of MAC units whose outputs are multiplexed through a divider. However, similarly to many detectors, the PRIDE test statistic given in (4) is not sensitive to scale factors. Hence, the division by N is purely from a mathematical rigor, having no impact on the spectrum sensing performance. Nonetheless, this division reduces the number of bits (word length) used to represent the result. Moreover, working with binary words, when the number of samples is a power of two the divider becomes a simple right-shift operation, meaning that an arithmetic divider unit is not needed. Thus, following this reasoning, here we propose a simple bit shift to perform the division by the highest power of two below N . For example, if $N = 50$ samples, the division is made as a shift of $\lfloor \log_2 50 \rfloor = 5$ bits to the right, and a bit resizing to reduce the word length from 18 bits to 13 bits. If the number of samples is a power of two, the values of $r_{i,k}$ will be equal to the ones produced by (2); otherwise they will be scaled by

a value in the range $(1, 2)$. Fig. 3 illustrates the architecture of the proposed optimized SCM computation module for the PRIDE test statistic.

As shown in [20], taking into account that $\mathbf{R}$ is Hermitian, $m(m-1)/2$ MACs are needed to calculate the upper or the lower off-diagonal elements of $\mathbf{R}$, and m MAC units are needed to calculate the diagonal elements, resulting in $m(m+1)/2$ MAC units to construct the SCM computation module.

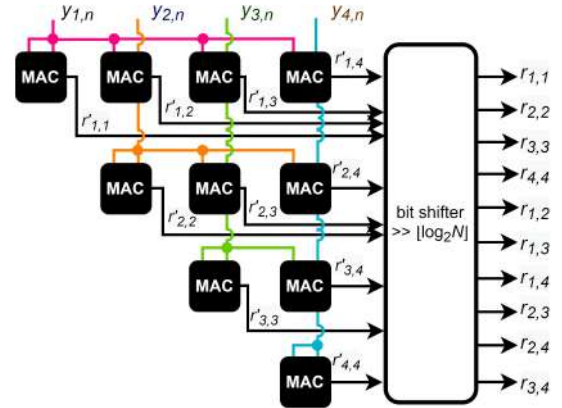


Fig. 3: Optimized SCM computation module.

B. Test statistic computation module

Two units have been designed to compose the test statistic computation (TSC) module: the mean value (MV) unit and the differential absolute value (DAV) unit, whose architectures are presented in Fig. 4. The MV unit, which is illustrated in Fig. 4a, calculates $\bar{r}$ defined in (3), exploring the fact that the upper and lower off-diagonal elements of the SCM are complex conjugates of each other. Hence, their sum is twice the sum of the real parts of one of them, yielding

$$\bar{r} = \frac{1}{m^2} \left(\sum_{i=1}^m r_{i,i} + 2 \sum_{i=1}^m \sum_{j=i+1}^m \Re\{r_{i,j}\} \right). \quad (10)$$

The MV shares the same simplification of the SCM, avoiding dividers when the number of SUs are a power of two. In this case, it is possible to use only a simple bit shifting operation. This property is explored by the authors for $m = 4$.

The DAV unit, which is depicted in Fig. 4b, has been implemented to calculate the numerator and the denominator of the PRIDE test statistic (4). In the numerator, the sum of all absolute values has been simplified, since complex conjugate numbers have the same magnitude. In the denominator, which it is computed by summing the magnitudes of the difference between the SCM elements and their mean, it has been applied the fact that the difference between a complex quantity and a real number has the same magnitude of the difference between its complex-conjugate and the number. Using these simplifications, the PRIDE test statistic can be rewritten as

$$T_{\text{PRIDE}} = \frac{\sum_{i=1}^m |r_{i,i}| + 2 \sum_{i=1}^m \sum_{j=i+1}^m |r_{i,j}|}{\sum_{i=1}^m |r_{i,i} - \bar{r}| + 2 \sum_{i=1}^m \sum_{j=i+1}^m |r_{i,j} - \bar{r}|}. \quad (11)$$

The DAV unit operates in two phases. During the first phase it processes the numerator of (11), while giving the MV unit the time to compute the mean value. In the second phase, the counters inside the internal control unit shown in Fig. 4b switch the multiplexed inputs of the absolute value (abs) operations and configure them to calculate the denominator. The demultiplexer (demux) output is controlled to store the numerator and the denominator in different registers at the proper time.

Aiming at reducing latency, the DAV unit has been implemented to compute all absolute value operations in the numerator and the denominator of (11) in parallel.

The TSC module architecture has been designed under two approaches in regard to the absolute value calculations, as shown in Fig. 5. In the conventional approach, the absolute value of an input is calculated by taking the square root of its real and imaginary parts squared. Two real multipliers are needed to calculate the values squared, subsequently applying a square root algorithm, as shown in Fig. 5a. The non-restoring square root algorithm of [27] has been used to implement the square root operation in this work.

To optimize the hardware architecture of the TSC module, an alternative to perform the absolute value calculation is the use of the CORDIC algorithm, as shown in Fig. 5b. The algorithm was originally proposed in [28] as a digital computer for airborne applications, but finds other uses in several situations [29]. The CORDIC algorithm is designed to perform angular rotations on vectors, iteratively. These rotations are made through a set of trigonometric equations, which can be simplified into a multiplication by the tangent of the desired rotation angle. In base 2, for the n -th iteration, it follows that $\tan \theta = 2^{-n}$, meaning that a multiplication by $\tan \theta$ is a simple shift of the binary word. The algorithm restricts

rotations to angles that meet this condition, so its construction depends only on simple hardware structures such as adders, registers and shifters.

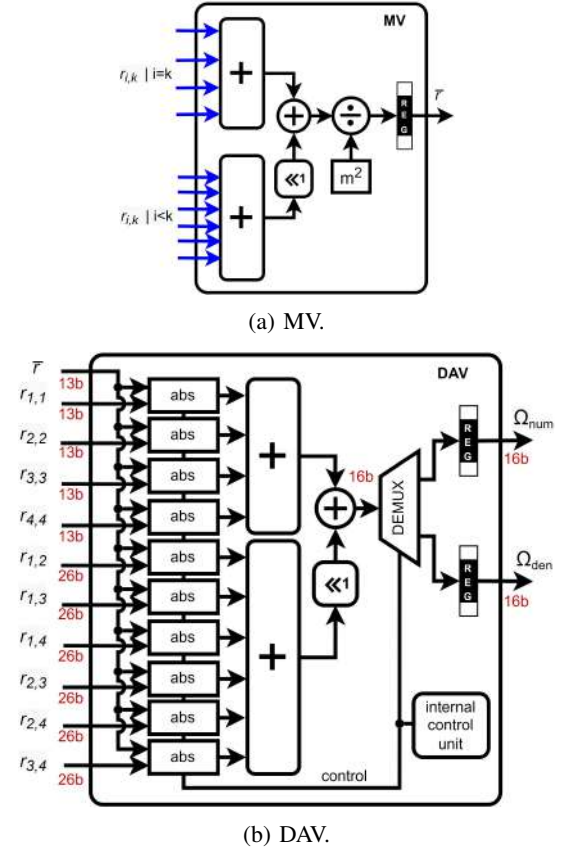


Fig. 4: Architectures of the MV and DAV units.

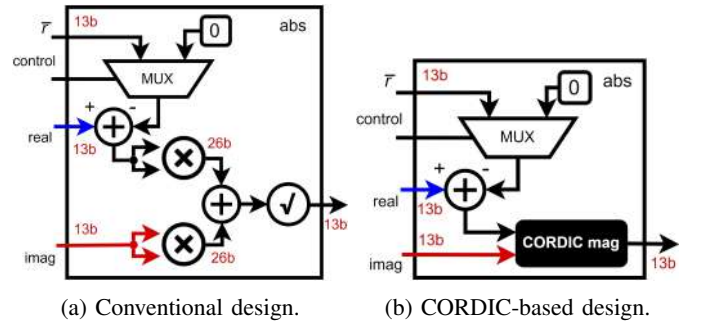


Fig. 5: Approaches used to perform the absolute value (abs) computation inside the DAV unit.

A CORDIC module normally has three inputs: x for the real part of the complex input value, y for the imaginary part and z for the desired rotation angle, and the algorithm has two modes of operation: rotation and vectorization. Only the vectorization mode is used herein. In this mode, the CORDIC algorithm seeks to minimize the residual value stored in the imaginary part register, so that the vector is rotated to the abscissa axis, that is, after n iterations $y \rightarrow 0$. As the aimed result is only the magnitude, all hardware features related to the rotation angle path can be removed, allowing a simplified construction

of the CORDIC module for magnitude calculation (denoted as CORDIC mag in Fig. 5b), which is depicted in Fig. 6. In the serial design, an input multiplexer ensures that after the first iteration, only the feedback values will continue to be processed by the algorithm. A configurable adder is controlled via a signal fed back from the outputs. In this case, the value of the control signal is equal to an exclusive OR between x and y signals, corresponding to their most significant bits (MSBs). The configurable adders are fed by registers and n -bit shifters. The construction of the CORDIC algorithm in this work applies $n = 4$ iterations and 13 bits word length for the real and imaginary inputs, as justified in the subsequent section of this paper.

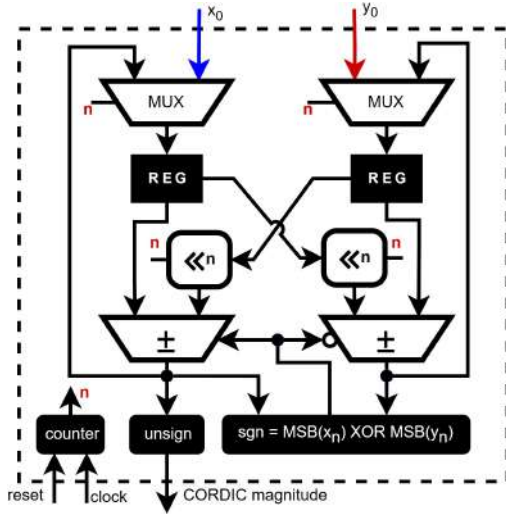


Fig. 6: Proposed serial CORDIC mag architecture.

Once the numerator and denominator of (11) are calculated, a divider unit would be needed to yield the PRIDe test statistic value. However, such a division operation should be avoided in hardware implementations due to its complexity and resource usage, although restoring algorithms allow its implementation more efficiently when the arithmetic operation is unavoidable. In this article, a divider unit has been initially implemented using the restoring algorithm based on [30] to calculate the test statistic value. However, it is possible to circumvent the division operation if only the spectrum occupation decision is the necessary output. This latter approach has been adopted by the final PRIDe sensor architecture, as described in the sequel.

C. Decision-making module

The decision-making (DM) module is simply a comparator. If the test statistic T_{PRIDe} is greater than or equal to the threshold γ , a decision indicator receives bit 1 (decision in favor of $\mathcal{H}_1$); otherwise it receives bit 0 (decision in favor of $\mathcal{H}_0$). After the decision is made, a 'done' signal is output to indicate that the state of the decision indicator is available in the 'decision' output.

Denoting the numerator of the test statistic as Ω_{num} , and the denominator as Ω_{den} , the PRIDe hypothesis test is

$$T_{PRIDe} = \frac{\Omega_{num}}{\Omega_{den}} \leq \gamma. \quad (12)$$

As already mentioned, if the value of T_{PRIDe} has no use and only the decision matters, the hypothesis test can be made by simply moving Ω_{den} to the threshold side, yielding

$$\Omega_{num} \leq \gamma \Omega_{den}, \quad (13)$$

in which the division present in (12) is exchanged by a multiplier operating on Ω_{den} and γ , yielding a simpler hardware implementation alternative.

The DM module is fed by the threshold value γ , which can be directly connected to an input pin of the DM, or can be multiplexed along with the input samples to reduce the number of pins of an FPGA or ASIC package. This latter approach is particularly useful if it is desired that the decision threshold can be changed on-the-fly, due to changes in some system parameter. If this is not the case, feeding the DM module with the threshold value via a dedicated pin is a more reasonable choice. In other words, the SCM computation module is idle during the TSC module operation, having its clock signal deactivated, which gives the TSC module a time equal to the TSC module latency to read the threshold by reusing a pin used as input of the SU samples in a denser package ASIC design. In this work, which targets both FPGA and ASIC designs, the multiplexed pin option was not applied.

D. Complete PRIDe spectrum sensor architecture

The final PRIDe sensor architecture has been implemented by interconnecting the modules previously described, as shown in Fig. 7. Some designs are made with the SCM computation module built using different MAC and TSC module architectures and different absolute value computation approaches. The results of these combinations are presented in the next section.

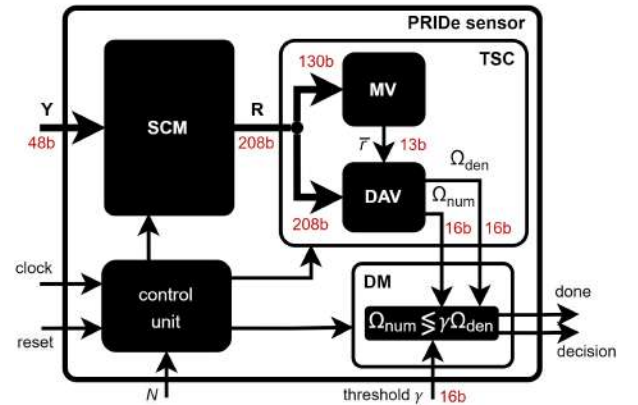


Fig. 7: Architecture of the Pietra-Ricci index detector.

The following design parameters have been adopted: the input samples are 6-bit signed fixed-point numbers for the real and imaginary parts. Hence, assuming $m = 4$ SUs, the sample matrix $\mathbf{Y}$ is fed to the sensor block by means of a 48-bit bus. The matrix $\mathbf{R}$ contains 10 values computed with 26 bits. The absolute value computations within the DAV unit are designed to operate with 13 bits for the real and the imaginary parts, the same setting being adopted by the MV unit. After summing up the absolute values, the numerator and the denominator

TABLE II: Comparative synthesis report for SCM and TSC modules.

	SCM-I	SCM-II	SCM-III	TSC-conventional	TSC-CORDIC
Slice LUTs	1831	1753	1331	5467	2404
Slice Registers	720	288	298	1244	512
Slices	640	558	452	1826	710
Latency (clock cycles)	N	N	$2N$	30	14

outputs are 16 bits wide, which is also the word length of the decision threshold γ . A controller unit processes clock and reset signals for all blocks.

IV. FINAL ARCHITECTURE DESIGN AND ASSESSMENT

This section describes the final architecture design and the assessment of its hardware characteristics. The design and architecture synthesis are addressed firstly, followed by the time and performance analysis of the developed PRIDe detector. Comparisons and discussions about the results and state-of-the-art outcomes are presented subsequently.

A. Design and synthesis

The resource consumption by the proposed architectures were analyzed through the synthesis and implementation report obtained from the FPGA development platform. The SCM computation module architecture is directly impacted by the MAC unit design and the TSC module is directly impacted by the DAV unit employed.

Table II presents the synthesis report of the proposed SCM and TSC modules. SCM-I refers to the architecture implemented with the MAC Architecture I (Fig. 2a), SCM-II refers to the MAC Architecture II, and SCM-III refers to the MAC Architecture III. The synthesis and implementation of the very high speed integrated circuit (VHSIC) hardware description language (VHDL) code have been carried out by using the Xilinx Vivado software, aiming the Xilinx FPGA chipset Zynq[®]-7000 SoC (XC7Z030FBG676-3).

It can be seen from Table II that the SCM-II incurs lesser logic utilization in terms of slice look-up tables (LUTs) compared with SCM-I, which is the simplest implementation in terms of complex multiplications, achieving significant advantage over the SCM-I in terms of register utilization. In the case of SCM-III, the logic utilization in terms of LUTs is $\approx 27\%$ smaller than SCM-I, but the module latency in number of clock cycles is twice the SCM-I latency. Comparing SCM-II and SCM-III, there is no clear winner. If resource usage (area efficiency) is more relevant than latency (time efficiency), SCM-III outperforms SCM-II. If the number of samples N is small, the higher area efficiency of SCM-III favors its choice. When N is too large, the higher time efficiency of SCM-II favors its choice.

The synthesis report of the TSC module presented in Table II unveils that the CORDIC algorithm attains resource utilization of 55%, 58% and 61% lesser in terms of LUTs, registers and slices, respectively, relative to the conventional implementation of the absolute value computation using the non-restoring square root algorithm of [27]. The latency attained with the CORDIC algorithm is 53% smaller than

the conventional implementation. Although both the CORDIC and the square root algorithms are iterative, the number of iterations of the square root depends on the word length associated to the value being processed [27], while the CORDIC uses a fixed preset number of iterations. For this reason, the CORDIC-based solution has been chosen here to implement the TSC module of the proposed PRIDe spectrum sensor.

In this work, FPGA implementations of two hardware architectures of the PRIDe spectrum sensor have been carried out. These suggested architectures are referred to as PRIDe V1 and PRIDe V2. Here, PRIDe V1 has been designed by incorporating SCM-II (that uses MAC-II architecture) and TSC-conventional modules, along with controller and DM module, as presented in Fig. 7. On the other hand, PRIDe V2 incorporates SCM-III (that uses MAC-III architecture) and TSC-CORDIC modules. Further, FPGA implementation results of both these spectrum sensors are listed in Table III.

From Table III, it can be concluded that the PRIDe V2 uses less hardware resources than the PRIDe V1, at the cost of a higher latency, which is the sum of latencies of all modules of our spectrum sensor.

TABLE III: Synthesis report of the PRIDe sensor. BUFG refers to global clock buffers, and bonded IOB refers to the input/output pins required by the module.

	PRIDe V1	PRIDe V2
Slice LUTs	4506	4086
Slice Registers	809	820
Slices	1360	1247
Bonded IOB	68	
BUFG	2	
Latency (clock cycles)	$N + 15$	$2N + 15$

Table IV summarizes the hardware complexities of the developed modules in terms of the basic units like MAC and CORDIC-based absolute value computation, and the basic hardware resources (adders, dividers and multipliers). The individual complexities are identified as follows: μ denotes the MAC unit complexity, α represents the CORDIC-based absolute value computation complexity, Σ denotes the adder complexity, χ identifies a divider complexity, and π denotes the complexity of a 16×16 bits multiplier. Recall that the divider used by the MV unit is a simple shifter if the number of SUs is a power of two.

Only higher hierarchy units like MAC, adders and absolute value computers are accounted in the hardware complexity analysis. Basic units like shifters, LUTs and registers are not included here. Moreover, the control unit, although indispens-

able to the proper functioning of the spectrum sensor, does not have its complexity affected by the design variables.

From Table IV it can be concluded that the hardware complexity growths of the designed modules are independent of the number of samples, N . The complexities of the SCM and TSC modules depend on the number of SUs, m , which affects the number of MAC and DAV units needed. The growths in the numbers of MAC and DAV units with m are the same, but their computational burdens (time complexities) are different from each other. The hardware complexity of the DM module is constant and determined by the type of multiplier used to compute $\gamma\Omega_{\text{den}}$.

TABLE IV: Hardware complexities.

	Hardware complexity
SCM module	$\Pi_{\text{SCM}} = \frac{m}{2}(m+1)\mu$
TSC module	$\Pi_{\text{TSC}} = \frac{m}{2}(m+1)\alpha + (m^2 + m - 2)\Sigma + \chi$
DM module	$\Pi_{\text{DM}} = \pi$

The post-implementation design of the PRIDe sensor has achieved a critical path delay² of 5.489 ns and 5.606 ns for PRIDe V1 and PRIDe V2 architectures, respectively. The above critical path delays are translated into maximum clock frequencies of 182 MHz and 178 MHz for PRIDe V1 and PRIDe V2 architectures, respectively. The operating clock frequency used by the proposed implementations considered herein is $f_{\text{clk}} = 166$ MHz. At this frequency, the dynamic power dissipations estimated in Vivado software with SAIF (Switching Activity Interchange Format) file from the post-implementation simulation are 130 mW for PRIDe V1 and 102 mW for PRIDe V2 resulting in dynamic power performances of 0.7831 mW/MHz and 0.6145 mW/MHz, respectively. At the worst-case scenario, operating at their maximum clock frequency, PRIDe V1 exhibits a dynamic power consumption of 141 mW, while PRIDe V2 consumes 109 mW. The thermal profiles of both sensors are depicted in Fig. 8, with reported values reaching a maximum temperature of 26.7°C for PRIDe V1 and 26.5°C for PRIDe V2, both without a heat sink and with zero airflow. The ambient temperature used in the power estimation is 25°C, and Vivado's default parameters include a medium heat sink and 250 linear-feet per minute (LFM) airflow.

The reported device has been designed to process the digital samples received from the SUs. The receiver front-end, the analog-to-digital conversion and other analog signal processing units are not addressed in the present design.

B. Detailed latency analysis

In this subsection, the delays that compose the overall latency of the PRIDe sensor architecture are described.

²The critical path delay is the maximum logical and routing delays among the designed paths. It is used to define the maximum operating frequency that the design can properly operate. If the critical path delay is too long, greater than clock period, the next rising edge clock occurs before a signal arrives at its destination.

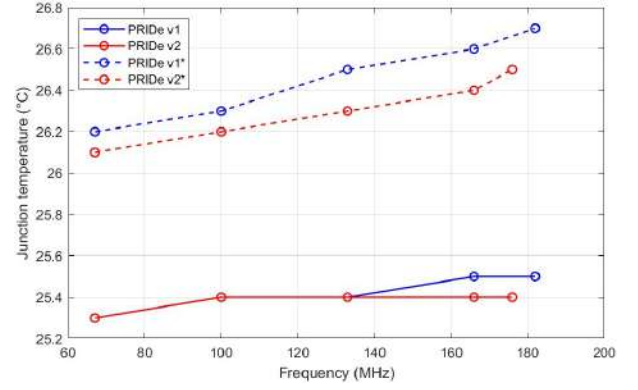


Fig. 8: The thermal profile of junction temperature as a function of frequency. Solid lines represent Vivado's default condition, while dashed lines correspond to scenarios with no heat sink and zero airflow.

The SCM computation module latency is denoted by Δ_1 , whose value is based on the required time by the MACs to process the input samples and compute the SCM. In the proposed architectures, both MAC-I and MAC-II are able to process one sample per clock cycle. On the other hand, MAC-III needs two clock cycles to reuse the same multipliers to calculate real and imaginary parts separately. Thus, the SCM computation module latency is N clock cycles for MAC-I and MAC-II, and $2N$ for MAC-III.

The MV unit latency is denoted by $\Delta_{2.1}$. In the proposed design, the MV unit realizes partial summations of the diagonal and off-diagonal elements of the SCM, left-shifting (equivalent to the multiplication of the off-diagonal elements by two) and right-shifting of the final summation (representing the division of the summation result by m^2), producing a latency equal to 3 clock cycles.

The latency of the absolute value operation inside the DAV unit is $\Delta_{2.2.1}$. The conventional non-restoring square root algorithm produces a latency equal to the word length used to represent each real and imaginary value, which in this architecture design is 13 bits. The CORDIC algorithm with four iterations plus a reset/pre-load clock cycle yields a latency of 5 clock cycles.

The summation latency inside the DAV unit is represented by $\Delta_{2.2.2}$. This summation, following (11), is performed independently on the diagonal and the off-diagonal elements of the SCM. Then, the off-diagonal sum is left-shifted and added to the diagonal sum. This process requires 3 clock cycles.

The extra latency of 1 clock cycle associated with the output enable and synchronization signal used by the DAV control unit is denoted by $\Delta_{2.2.3}$.

The total latency of the TSC module is Δ_2 . The MV unit calculates the mean value in parallel with the test statistic numerator computation, which causes no effect on the TSC module latency because $\Delta_{2.1}$ is always smaller than $\Delta_{2.2.1}$, regardless of the DAV architecture (conventional or CORDIC-based). Moreover, the computation of the denominator starts right after the computation of the absolute value of the numerator is completed. Consequently, the DAV unit produces two

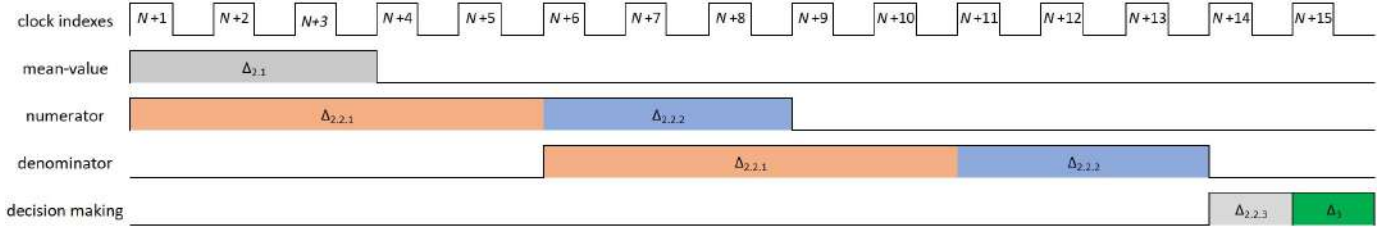


Fig. 9: Detailed timing chart for the TSC (Δ_2) and DM (Δ_3) modules.

times the absolute value operation latency, plus the summation, demux and registering latencies, plus $\Delta_{2,2,3}$. Fig. 9 depicts how this parallel computation is performed. The TSC module delay is $\Delta_2 = 2\Delta_{2,2,1} + \Delta_{2,2,2} + \Delta_{2,2,3}$, yielding a latency of 30 and 14 clock cycles for the TSC module using the conventional and the CORDIC-based absolute value computation approaches, respectively.

The DM module latency, which is 1 clock cycle, is denoted by Δ_3 . This is the time needed to multiply Ω_{den} by γ and compare the result with Ω_{num} to make the decision on the spectrum occupancy state.

Fig. 9 illustrates the timing diagram for the above-described latencies Δ_2 and Δ_3 .

Finally, Δ_{PRIDe} is the total latency (in number of clock cycles) required by the PRIDe spectrum sensor. It is equal to the sum of individual modules latencies described above, that is,

$$\Delta_{\text{PRIDe}} = \Delta_1 + \Delta_2 + \Delta_3. \quad (14)$$

Cognitive radio networks with centralized data-fusion CSS operate with a fixed frame structure that is divided into a spectrum sensing interval, followed by an interval corresponding to the report of the collected samples to the FC, an interval for processing the received samples at the FC and making the spectrum occupancy decision, an interval for spectrum allocation, and an interval for data transmission. Clearly, the reduction of the time spent for sensing, reporting, FC processing, and spectrum allocation allows the cognitive radio network to transmit more data in the frame, increasing its data throughput [31]. Likewise [20], this work focus on the reduction of the processing time at the FC side. Hence, the PRIDe sensor architecture developed herein aims at minimizing the test statistic computation and global decision delays. Given a clock frequency f_{clk} , the time required by these tasks at the FC side is given by

$$\tau_{\text{FC}} = \frac{\Delta_{\text{PRIDe}}}{f_{\text{clk}}}. \quad (15)$$

C. Detection performance analysis

The impact of fixed-point operation and the number of CORDIC iterations on the performance have been assessed by means of Monte Carlo simulations in Matlab. The aim was to set the minimum number of iterations for the CORDIC-based absolute value computations, and design the overall PRIDe sensor architecture with the smallest word lengths

that do not cause noticeable performance loss with respect to the conventional absolute value calculation and floating-point operation. The results found were 6 bits for the fixed-point word length that represent the input samples, and 4 iterations in the CORDIC algorithm.

Starting with the design parameters defined from the simulations, the hardware description language (HDL) code for the proposed sensor architecture has been written in VHDL using the Xilinx Vivado software. Subsequently, the sensor performance has been evaluated by comparing Matlab simulation results with the behavioral (also known as register transfer level, RTL) simulation results, as well as with the post-implementation (also known as functional) simulation results obtained from the Vivado software.

To perform the Monte Carlo simulations in Matlab, 15000 sample matrices $\mathbf{Y}$ of order $m \times N = 4 \times 100$ were generated using the Matlab, under hypothesis $\mathcal{H}_1$ with SNR = -8 dB, and under $\mathcal{H}_0$. These matrices were exported from Matlab and imported by custom read-only memories (ROMs) inside the RTL simulation test bench. The results generated by the RTL simulation followed the reverse export-import path to plot the receiver operating characteristics (ROC) curve of the PRIDe test statistic. The Matlab-generated samples were used to simulate the PRIDe sensor in floating-point operation, and converted into signed fixed-point representation with 6-bit word length and 5-bit fraction to simulate the PRIDe sensor in fixed-point operation.

Fig. 10 shows all ROC curves obtained from the performance analysis of the PRIDe sensor: the ROC obtained via Matlab using the conventional absolute value computation in float-point (double), the ROC obtained via Matlab using the CORDIC-based absolute value computation with samples processed in fixed-point, the ROC obtained from the RTL simulation via Vivado, and a single operating point representing the measured $P_{\text{fa}} = 0.1060$ and $P_{\text{d}} = 0.7424$, obtained from the post-implemented functional simulation using a decision threshold $\gamma = 0.884765625$, whose value was pre-computed targeting $P_{\text{fa}} = 0.1$. From Fig. 10, it can be seen that all ROC curves are superimposed, and that the measured operating point lies on these curves. Thus, the 6-bit word length adopted to represent the input samples in fixed-point notation, and the number of CORDIC iterations equal to 4 are capable of not degrading the PRIDe performance relative to floating-point computations and the use of conventional absolute value computation whose hardware implementation depends on less efficient algorithms (e.g., iterative square root algorithms).

The measured mean squared error (MSE) between the test

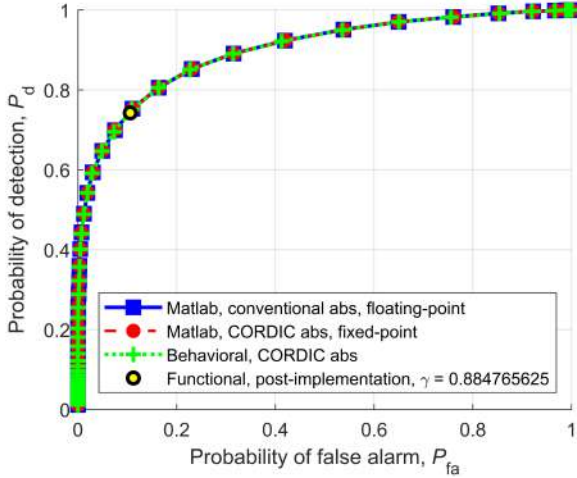


Fig. 10: Simulated performance of the PRIDe sensor.

statistic values calculated by the Matlab simulation and those obtained from the RTL hardware-level simulation using the data-set under hypothesis $\mathcal{H}_0$ was 4.457×10^{-6} . Under $\mathcal{H}_1$, the MSE was 1.130×10^{-4} . In both cases, the CORDIC-based absolute value computation was adopted. From the agreement of the results presented in Fig. 10, it can be concluded that such errors can be considered small enough to cause no impact on the performance of the FPGA and ASIC-implemented PRIDe sensor.

The overall processing times under a clock frequency of 166 MHz were 690 ns and 1290 ns for the PRIDe V1 and the PRIDe V2 architectures, respectively, demonstrating a clear advantage of PRIDe V1 in this regard, and an overall extremely low-latency design.

D. ASIC design and comparisons

The proposed hardware architectures of PRIDe V1 and PRIDe V2 spectrum sensors are ASIC synthesized and post-layout simulated in the 90 nm-CMOS technology node from United Microelectronics Corporation (UMC). The VHDL codes of suggested spectrum sensors are functionally verified, synthesized, post-synthesis simulated and timing analyzed using the NCSim electronic design automation (EDA) tool from Cadence. Further, the timing verified gate-level netlist of our design has been imported to Cadence-Innovus EDA tool, using the 5-metal layer LEF files of 90 nm-CMOS process. Here, the physical design of the spectrum sensor architecture is carried out where floor-plan, power plan, placement, signal routing, clock tree synthesis, and timing verifications are carried out hierarchically. Based on the post placed-&-route timing analysis, PRIDe V1 and PRIDe V2 architectures are capable of delivering maximum clock frequency of 186.6 MHz with the critical path delay of 5.386 ns. Further, post-layout simulations indicate that PRIDe V1 and PRIDe V2 attain latencies – i.e. Δ_{PRIDe} value in (15) – of 115 and 215 clock cycles, respectively, while processing 100 signal samples. As a result, at aforementioned clock frequency (f_{clk}) and latencies, sensing times (i.e. $\tau_{\text{FC}} = \Delta_{\text{PRIDe}}/f_{\text{clk}}$) of $0.616 \mu\text{s}$ and $1.152 \mu\text{s}$

are delivered by PRIDe V1 and PRIDe V2 spectrum sensors, respectively. Comprehensive power analysis has been carried out at the clock frequency of 186 MHz with the supply voltage of 1.2. Thus, PRIDe V1 and PRIDe V2 consume total powers (leakage and dynamic powers) of 15.72 mW and 9.695 mW, respectively. An overall design area of PRIDe V1 is 0.094 mm^2 that incorporates 11700 standard cells. Similarly, PRIDe V2 with 10421 cell-count occupies 0.084 mm^2 of area. Chip layouts in 90 nm-CMOS process of both the proposed PRIDe V1 and PRIDe V2 spectrum sensors are presented in Fig. 11.

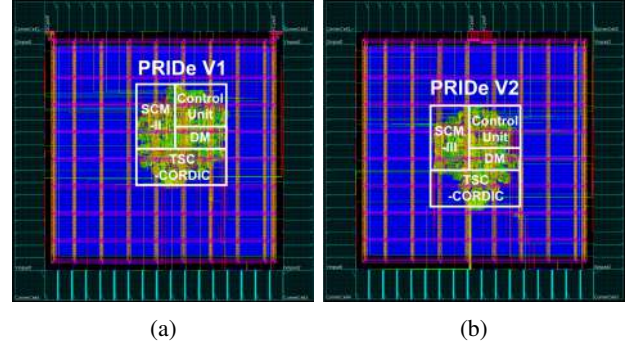


Fig. 11: 5-metal layered ASIC chip-layouts of the proposed (a) PRIDe V1 and (b) PRIDe V2 spectrum sensors in UMC 90 nm-CMOS technology node.

Our ASIC design results of the proposed spectrum sensors are presented and compared with the state-of-the-art implementations in Table V. Here, the comparisons have been carried out with two types of spectrum sensors: cooperative spectrum sensors (CSRs) and stand-alone spectrum sensors (SSSRs). Synthesized and post-layout simulated results of PRIDe V1 and PRIDe V2 are compared with the contemporary Gini index-based [32] and GRCR-based [20] CSRs. Furthermore, implementation of [18] is the unified MED/MMED-based CSR. The reported work of [16] is based on GLRT based CSR that uses iterative power method to compute all the eigenvalues of SCM. Unlike, [17] applies iterative Cholesky method for the same. These CSRs from literature deliver excellent detection performance, under the assumption that the received signal-power and noise-variance, at the cooperating SUs, are uniform. However, in a real-world scenario where the received signal-power and noise-variance are different, SUs fluctuate in both space and time (i.e. non-uniform dynamical noise and received signal-power scenario). Under such realistic scenario, the proposed PRIDe-based CSS algorithm delivers superior performance as compared to CSS algorithms from [16]–[18], [20], [32].

In addition, the implementations reported in [33]–[40] are all SSSRs. Here, SSSR from [33] is a digital baseband processor based on adaptive channel-specific threshold and sensing-time. Similarly, [34] is a rapid interferer detector that uses compressed sampling with a quadrature analog-to-information converter. The SSSR reported in [35] is a 30 MHz to 2.4 GHz CMOS-receiver with an integrated tunable RF-filter and a dynamic-range-scalable energy detector for white-space and interference-level sensing in cognitive-radio systems. Unlike, [36] and [39] are digital SSSRs based on cyclostationary-

TABLE V: Comparison of the proposed PRIDe V1 and PRIDe V2 spectrum sensors with the state-of-the-art implementations.

	Prop. PRIDe V1 ^{2s}	Prop. PRIDe V2 ^{2s}	[32] [‡] TCE-2022	[20] [‡] TVLSI-2022	[18] [‡] ISCAS-2021	[16] [✱] TVLSI-2021	[17] [✱] TCAS-II-2021	[33] [‡] JSSC-2012	[34] [‡] JSSC-2015	[35] [‡] JSSC-2012	[36] [✱] TCAS-II-2018	[37] [‡] TCAS-I-2018	[38] [‡] TCAS-I-2018	[39] [‡] TCAS-I-2019	[40] [✱] TVLSI-2016
Topology	CSR	CSR	CSR	CSR	CSR	CSR	CSR [§]	SSSR	SSSR	SSSR	SSSR	SSSR	SSSR	SSSR	SSSR
Technology (nm)	90 [®]	90 [®]	130 [✱]	130 [®]	130 [†]	90 [†]	90 [†]	65 [⊕]	65 [✱]	90 [‡]	90 [✱]	130 [✱]	65 [✱]	90 [⊖]	130 [✱]
Supply (V)	1.2	1.2	1.2	1.2	1.2	1.2	1.2	1	1.1	1.2	1.2	1.2/1.1	1	1.2	1.5
Area (mm ²)	0.094	0.084	0.35	0.27	0.564	2.41	2.47	1.64	1.96	2.3	0.26 ^d	1.33	2.53	0.42 ^c	0.165
Scaled area [#] (mm ²)	0.094	0.084	0.167 ^α	0.129 ^α	0.27 ^α	2.41	2.47	3.144 ^β	3.76 ^β	2.3	0.26	0.64 ^α	4.85 ^β	0.42	0.08 ^β
Total power (mW)	15.72 [°]	9.695 [°]	8.31 ^δ	6.47 ^δ	8.3809	35.35	31.84	7.4	81	44	39.66	0.878	47.9	38.24	28.5
Scaled total power (mW) [‡]	15.72 [°]	9.695 [°]	5.05 ^δ	3.93 ^δ	5.09	35.35	31.84	17.33	152.33	44	39.66	0.65	112.2	38.24	10.41
Max. clock frequency (MHz)	186.6	186.6	88.8	88.8	88.8	87.71	101.83	-NA-	-NA-	-NA-	-NA-	-NA-	-NA-	404	-NA-
Detection bandwidth ^ε (MHz)	93.3 [⊙]	93.3 [⊙]	44.4 [⊙]	44.4 [⊙]	44.4 [⊙]	43.85 [⊙]	50.91 [⊙]	200 ^{<}	1000 ⁺	0.2-30	200 [⊕]	0.36-0.72 [⊥]	132 [⊥]	>400 [⊥]	40 [⊕]
Sensing time (ms)	0.00062	0.0012	0.043	0.005	0.120/0.236	0.0604	0.133	<50	0.004	-NA-	<5	0.133	0.42	0.0535	1
Scaled sensing time (ms) [‡]	0.00062	0.0012	0.03026	0.00352	0.08445/0.166	0.0604	0.133	<54.11	0.00493	-NA-	<5	0.0865	0.4545	0.0535	0.7727
ATP (mm ² -ms) ^λ	0.00006	0.0001	0.0150	0.0014	0.0678/0.133104	0.1455	0.3285	82	0.007	-NA-	1.3	0.177	1.063	0.0225	0.165
Scaled ATP (mm ² -ms) [‡]	0.00006	0.0001	0.0051	0.00046	0.0228/0.0449	0.1455	0.3285	170.13	0.0185	-NA-	1.3	0.0551	2.20	0.0225	0.0611
PDP (mW-ms) ^ν	0.0097	0.012	0.3573	0.0323	1.00/1.977	2.14	4.23	370	0.324	-NA-	198.3	0.117	20.118	2.046	28.5
Scaled PDP (mW-ms) [‡]	0.0097	0.012	0.1527	0.0138	0.4299/0.8455	2.14	4.23	937.97	0.7506	-NA-	198.3	0.05624	51.00	2.046	8.04

§: CSR based on GLRT CSS-algorithm with Cholesky algorithm; ✱: GID based digital CSR; ⊙: GRCR based digital CSR; ®: PRIDe based digital CSR; ⊕: Wideband digital baseband Processor; λ: ATP (Area Time Product) = Area × Sensing Time; ν: PDP (Power Delay Product) = Total Power × Sensing Time.
 ✱: Wideband Rapid Interferer Detector; ‡: Dynamic-Range-Scalable Energy detector for Cognitive radio; †: Eigenvalue based digital CSR
 ‡: Scaled Area = Area/ s^2 where s = scaling factor; α: $s = (130/90)$; β: $s = (65/90)$; γ: $s = (180/130)$.
 ‡: The scaled metrics for power and delay follow the scaling equations as outlined in [41].
 °: Total power consumption at 186 MHz. δ: Total power consumption at 88 MHz. ⊖: MME Based digital spectrum sensor.
 ◆: Cyclostationary Feature Detection (CFD) based digital spectrum sensor; ◆: ED based analog spectrum sensor; ◆: ED based digital spectrum sensor.
 ✱: Synthesized and post-layout simulated results; ‡: Measured results from chip tape-out.
 Input word lengths of $x[n]$: a : 10 bits; b : 14 bits; c : 28 bits; d : 20 bits; e : 26 bits.
 ⊙: Signal sensing bandwidth of CSR that is situated in the digital-baseband part of spectrum sensing receiver.
 ⊕: Signal sensing bandwidth of Digital SSSR that is situated in the digital-baseband part of spectrum sensing receiver.
 ⊥: Sensing bandwidth of Analog SSSR that is situated in the Analog RF-Frontend part of spectrum sensing receiver.
 ⊙ & ε: Maximum clock frequency (Θ_{max}) $\geq 2 \times f_{bb}$ (Detection Bandwidth); $\therefore f_{bb} \approx \frac{\Theta_{max}}{2}$.
 <: Sensing Bandwidth of 200 MHz with resolution of 200-kHz; +: Sensing Bandwidth of 1000 MHz with resolution of 20 MHz.

feature detection and maximum-minimum eigenvalue based detection techniques, respectively. Further, [37] is successive-approximation-register based analog ED SSSR for ultra-wideband cognitive-radio applications with short sensing time. On the other hand, [40] is an analog CMOS-RF based ED SSSR.

As shown in Table V, both PRIDe V1 and PRIDe V2 spectrum sensors occupy the smallest area in comparison to all the reported implementations. To ensure fair comparison between different semiconductor technologies, scaling equations are adopted following [41]. Here, the area, delay and power consumed by all the implementations are scaled to 90 nm CMOS technology node. In comparison to the smallest area consumed by the state-of-the-art CSR from [20], PRIDe V1 and PRIDe V2 spectrum sensors consume 27.1% and 34.9% lower area, respectively. On the other hand, Table V also shows the comparison of our CSRs with the SSSRs, reported in the literature. Here, only the work presented in [40] delivers smaller area than the proposed CSRs. This SSSR has been designed for ED based spectrum sensing algorithm in the

analog circuit domain that delivers unreliable detection performance under real-world scenario and undergoes SNR wall problem. Nevertheless, ATP and PDP values of the suggested PRIDe V1 and PRIDe V2 are better than ones obtained for [40]. Similarly, sensing times of our CSRs are 5.7× and 2.9× better than the fastest sensing time of contemporary CSR from [20], as presented in Table V. It also shows that the suggested designs have delivered the lowest ATP and PDP among all the reported implementations. Therefore, both PRIDe V1 and PRIDe V2 are the most hardware as well as power efficient CSRs, reported till date.

E. ASIC Fabrication Considerations

The fabrication of the sensor at different technology nodes can improve or worsen the sensor performance. Smaller technology nodes (e.g., 65 nm, 45 nm, 22 nm) can achieve higher maximum operating frequencies, resulting in reduced latency. However, the number of clock cycles required for

output remains constant, determined solely by PRIDe's architectural design. Smaller technology nodes lead to lower power consumption than 90 nm CMOS process, provided they are operating at same clock frequency. In larger technology nodes, latency increases due to lower clock frequencies, while power consumption depends on clock frequency and supply voltage. In larger nodes, clock frequency decreases and supply voltage increases, potentially causing increased power consumption due to the quadratic relationship between total power (mostly dynamic power) and supply voltage.

Irrespective of potential fabrication defects (unavoidable and not under the designer's control), there could be some expected deviations in practical, measured results. One issue is the risk of setup-time violations along the critical path of the proposed architecture. This issue can occur due to underestimating clock skew and jitter, leading to uncertainty. Consequently, the fabricated ASIC chip may be unable to operate at its maximum clock frequency of 186.6 MHz, reducing achievable sensing time and sensing bandwidth. Another challenge is the potential for hold-time violations within the fabricated PRIDe architecture. This is the most challenging problem because it cannot be corrected post-fabrication and remains independent of clock frequency. Finally, the design must ensure there are enough supply voltage pads. Without them, the supply voltage distribution to standard cells may be insufficient, leading to reduced rail-to-rail swing in digital logic, which, in turn, impacts noise margins and overall reliability.

Some specific solutions and precautions can be adopted in the design to mitigate these fabrication challenges. In response to the setup-time violation issue, one viable approach is to modify the design to operate at lower clock frequencies until it meets the set-time condition of the critical path. Alternatively, it is possible to explore the use of better clock sources with reduced jitter and clock skew. A crucial precautionary measure for hold-time violations involves rigorous hold condition checks during the synthesis and post-layout simulation phases. Finally, to address the challenge of insufficient supply voltage distribution, the design must ensure an adequate number of supply pads and also use the appropriate size of routing wires to deliver the necessary supply to standard cells. The mentioned modifications are alternatives to improve the noise margin and reliability of the design.

V. CONCLUSIONS

This article explored an ultra-low latency design and implementation of the PRIDe detector for centralized data-fusion cooperative spectrum sensing in FPGA and ASIC.

New approaches were proposed as alternatives to conventional architectures for the computation of the PRIDe's test statistic, namely: the absolute value of complex quantities, the complex multiplier-accumulator, and the spectrum occupancy decision. The multiplier-accumulator uses only two clock cycles, and the absolute value operation, which is critical to the PRIDe test statistic computational cost, applies the CORDIC algorithm as a much more efficient option in terms of resource usage and latency. The spectrum occupancy decision has been simplified by shifting the denominator of the test statistic to the

decision threshold, avoiding an extra division operation. RTL and Monte Carlo simulations unveiled that the resulting PRIDe sensor yields no performance loss with respect to floating-point simulation results.

The PRIDe sensor, in its most efficient version (PRIDe v1), consumes a silicon area of 0.094 mm², a power consumption of 15.72 mW, and a sensing time of 620 ns. Thus, the proposed sensor presents an ATP of 0.00006 mm²·ms and a PDP of 0.0097 mW·ms, which the respective scaled values are 7.66× and 1.42× smaller than its best competitor, with ATP of 0.00046mm²·ms and PDP of 0.0138mW·ms reported in [20].

These results have shown that the PRIDe spectrum sensor is more resource- and time-efficient for hardware implementations than concurrent state-of-the-art test statistics reported in the literature.

The PRIDe detector is suitable for ordinary spectrum sensing applications, but is especially suitable for those applications that demand fast sensing, for example to scan a wide frequency band in a short time, to fasten the sliding-window approach for detecting pulse radar signals, or simply to reduce the overall sensing time, aiming at increasing the cognitive secondary network data throughput.

CONFLICT OF INTEREST

All authors certify that they have no affiliations with or involvement in any organization or entity with any financial interest or non-financial interest in the subject matter or materials discussed in this manuscript.

REFERENCES

- [1] W. Lehr and J. Chapin, "Mobile broadband growth, spectrum scarcity and sustainable competition," in *Telecommunications Policy Research Conference*, Arlington VA, 2011.
- [2] K. Patil, R. Prasad, and K. Skouby, "A survey of worldwide spectrum occupancy measurement campaigns for cognitive radio," in *2011 International Conference on Devices and Communications (ICDeCom)*, 2011, pp. 1–5.
- [3] Q. Zhao and B. M. Sadler, "A survey of dynamic spectrum access," *IEEE Signal Processing Magazine*, vol. 24, no. 3, pp. 79–89, 2007.
- [4] J. Mitola and G. Q. Maguire, "Cognitive radio: making software radios more personal," *IEEE personal communications*, vol. 6, no. 4, pp. 13–18, 1999.
- [5] H. Urkowitz, "Energy detection of unknown deterministic signals," *Proceedings of the IEEE*, vol. 55, no. 4, pp. 523–531, 1967, doi: 10.1109/PROC.1967.5573.
- [6] Y. Zeng, C. L. Koh, and Y.-C. Liang, "Maximum eigenvalue detection: Theory and application," in *2008 IEEE International Conference on Communications*, 2008, pp. 4160–4164.
- [7] T. J. Lim, R. Zhang, Y. C. Liang, and Y. Zeng, "GLRT-based spectrum sensing for cognitive radio," in *IEEE GLOBECOM 2008 - 2008 IEEE Global Telecommunications Conference*, 2008, pp. 1–5.
- [8] Y. Zeng and Y.-C. Liang, "Maximum-minimum eigenvalue detection for cognitive radio," in *2007 IEEE 18th International Symposium on Personal, Indoor and Mobile Radio Communications*, 2007, pp. 1–5.
- [9] R. Zhang, T. J. Lim, Y. C. Liang, and Y. Zeng, "Multi-antenna based spectrum sensing for cognitive radios: A GLRT approach," *IEEE Trans. Commun.*, vol. 58, no. 1, pp. 84–88, Jan 2010.
- [10] S. Sedighi, A. Taherpour, J. Sala-Alvarez, and T. Khattab, "On the performance of Hadamard ratio detector-based spectrum sensing for cognitive radios," *IEEE Transactions on Signal Processing*, vol. 63, no. 14, pp. 3809–3824, 2015.
- [11] L. Huang, C. Qian, Y. Xiao, and Q. T. Zhang, "Performance analysis of volume-based spectrum sensing for cognitive radio," *IEEE Trans. Wirel. Commun.*, vol. 14, no. 1, pp. 317–330, Jan 2015.
- [12] D. A. Guimarães, "Robust test statistic for cooperative spectrum sensing based on the Gerschgorin circle theorem," *IEEE Access*, vol. 6, pp. 2445–2456, 2018.

- [13] —, “Gini index inspired robust detector for spectrum sensing over Ricean channels,” *Electronics Letters*, 11 2018.
- [14] —, “Pietra-Ricci index detector for centralized data fusion cooperative spectrum sensing,” *IEEE Transactions on Vehicular Technology*, vol. 69, no. 10, pp. 12 354–12 358, 2020.
- [15] M. López-Benítez and F. Casadevall, “Improved energy detection spectrum sensing for cognitive radio,” *IET communications*, vol. 6, no. 8, pp. 785–796, 2012.
- [16] R. B. Chaurasiya and R. Shrestha, “A new hardware-efficient spectrum-sensor VLSI architecture for data-fusion based cooperative cognitive-radio network,” *IEEE Trans. Very Large Scale Integr.*, vol. 29, no. 4, pp. 760–773, April 2021.
- [17] —, “Area-efficient and scalable data-fusion based cooperative spectrum sensor for cognitive radio,” *IEEE Trans. Circuits Syst. II*, vol. 68, no. 4, pp. 1198–1202, April 2021.
- [18] —, “Hardware-efficient ASIC implementation of eigenvalue based spectrum sensor reconfigurable-architecture for cooperative cognitive-radio network,” in *IEEE Int. Symp. on Circuits and Systems (ISCAS)*, May 2021, pp. 1–5.
- [19] G. H. Golub and H. A. van der Vorst, “Eigenvalue computation in the 20th century,” *Journal of Computational and Applied Mathematics*, vol. 123, no. 1, pp. 35–65, 2000, numerical Analysis 2000. Vol. III: Linear Algebra. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0377042700004131>
- [20] R. B. Chaurasiya and R. Shrestha, “Hardware-efficient VLSI architecture and ASIC implementation of GRCC-based cooperative spectrum sensor for cognitive radio network,” *IEEE Trans. Very Large Scale Integr.*, vol. 30, no. 2, pp. 166–176, February 2022.
- [21] D. A. Guimarães and C. H. Lim, “Sliding-window-based detection for spectrum sensing in radar bands,” *IEEE Communications Letters*, vol. 22, no. 7, pp. 1418–1421, 2018.
- [22] S. Zhu, T. S. Ghazany, S. M. R. Jones, R. A. Abd-Alhameed, J. M. Noras, T. Van Buren, J. Wilson, T. Suggett, and S. Marker, “Probability distribution of Rician K -factor in urban, suburban and rural areas using real-world captured data,” *IEEE Trans. Antennas Propag.*, vol. 62, no. 7, pp. 3835–3839, Jul 2014.
- [23] The Institute of Electrical and Electronic Engineers, IEEE. (2011) IEEE 802 Part 22: Cognitive Wireless RAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications: Policies and Procedures for Operation in the TV Bands. [Online]. Available: <http://standards.ieee.org/getieee802/download/802.22-2011.pdf>
- [24] Xilinx, Inc., “LogiCORE IP Complex Multiplier v3.1,” Mar. 2011, accessed on: 14 February, 2023. [Online]. Available: https://www.xilinx.com/content/dam/xilinx/support/documents/ip_documentation/cmpy_ds291.pdf.
- [25] R. Warrier, W. Zhang, and C. H. Vun, “Pipeline reconfigurable DSP for dynamically reconfigurable architectures,” *Circuits, Systems, and Signal Processing*, vol. 36, no. 9, pp. 3799–3824, 2017.
- [26] D. Serpanos and T. Wolf, “Chapter 4 - interconnects and switching fabrics,” in *Architecture of Network Systems*, ser. The Morgan Kaufmann Series in Computer Architecture and Design, D. Serpanos and T. Wolf, Eds. Boston: Morgan Kaufmann, 2011, pp. 35–61. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/B9780123744944000049>
- [27] Y. Li and W. Chu, “A new non-restoring square root algorithm and its VLSI implementations,” in *Proceedings International Conference on Computer Design. VLSI in Computers and Processors*, 1996, pp. 538–544.
- [28] J. E. Volder, “The CORDIC trigonometric computing technique,” *IRE Transactions on Electronic Computers*, vol. EC-8, no. 3, pp. 330–334, 1959.
- [29] R. Andraka, “A survey of CORDIC algorithms for FPGA based computers,” *ACM/SIGDA International Symposium on Field Programmable Gate Arrays - FPGA*, 12 2001.
- [30] S. Kaur, M. Manna, and R. Agarwal, “VHDL implementation of non restoring division algorithm using high speed adder/subtractor,” *International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering*, vol. Vol. 2, pp. 3317–3324, 07 2013.
- [31] M. Ali and H. Nam, “Optimization of spectrum utilization in cooperative spectrum sensing,” *Sensors*, vol. 19, no. 8, 2019. [Online]. Available: <https://www.mdpi.com/1424-8220/19/8/1922>
- [32] R. B. Chaurasiya and R. Shrestha, “Design and ASIC-implementation of hardware-efficient cooperative spectrum sensor for cognitive radio network,” *IEEE Trans. Consum. Electron.*, vol. 68, no. 3, pp. 221–235, August 2022.
- [33] T.-H. Yu, C.-H. Yang, D. Čabrić, and D. Marković, “A 7.4mW 200MS/s wideband spectrum sensing digital baseband processor for cognitive radios,” in *2011 Symposium on VLSI Circuits - Digest of Technical Papers*, 2011, pp. 254–255.
- [34] R. T. Yazicigil, T. Haque, M. R. Whalen, J. Yuan, J. Wright, and P. R. Kinget, “Wideband rapid interferer detector exploiting compressed sampling with a quadrature analog-to-information converter,” *IEEE J. Solid-State Circuits*, vol. 50, no. 12, pp. 3047–3064, December 2015.
- [35] M. Kitsunezuka, H. Kodama, N. Oshima, K. Kunihiro, T. Maeda, and M. Fukaiishi, “A 30-MHz-2.4-GHz CMOS receiver with integrated RF filter and dynamic-range-scalable energy detector for cognitive radio systems,” *IEEE J. Solid-State Circuits*, vol. 47, no. 5, pp. 1084–1093, May 2012.
- [36] M. S. Murty and R. Shrestha, “Reconfigurable & memory-efficient cyclostationary spectrum sensor for cognitive-radio wireless networks,” *IEEE Trans. Circuits Syst. II*, vol. 65, no. 8, pp. 1039–1043, August 2018.
- [37] K. Banović and A. C. Carusone, “A sub-mW integrating mixer SAR spectrum sensor for portable cognitive radio applications,” *IEEE Trans. Circuits Syst. I*, vol. 65, no. 3, pp. 1110–1119, March 2018.
- [38] N.-S. Kim and J. M. Rabaey, “A dual-resolution wavelet-based energy detection spectrum sensing for UWB-based cognitive radios,” *IEEE Trans. Circuits Syst. I*, vol. 65, no. 7, pp. 2279–2292, 2018.
- [39] R. B. Chaurasiya and R. Shrestha, “Hardware-efficient and fast sensing-time maximum-minimum-eigenvalue-based spectrum sensor for cognitive radio network,” *IEEE Trans. Circuits Syst. I*, vol. 66, no. 11, pp. 4448–4461, November 2019.
- [40] V. Khatri and G. Banerjee, “A 0.25-3.25-GHz wideband CMOS-RF spectrum sensor for narrowband energy detection,” *IEEE Trans. Very Large Scale Integr.*, vol. 24, no. 9, pp. 2887–2898, September 2016.
- [41] A. Stillmaker and B. Baas, “Scaling equations for the accurate prediction of cmos device performance from 180nm to 7nm,” *Integration*, vol. 58, pp. 74–81, 2017. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0167926017300755>



Elivander Judas Tadeu Pereira received the degree of Bachelor in Engineering (2018) and the M.Sc. in Telecommunications (2020) from the National Institute of Telecommunications (Inatel), Brazil. He is currently working towards his Doctorate on Telecommunications at Inatel. His research interests are mobile communications, digital transmission, cognitive radio, statistics, hardware development and signal processing.



Dayan Adionel Guimarães received an MSc and a PhD in Electrical Engineering from the State University of Campinas (Unicamp), Brazil, in 1998 and 2003, respectively. He is a Researcher and Senior Lecturer in the National Institute of Telecommunications (Inatel), Brazil. His research focuses the general aspects of wireless communications, specifically radio propagation, digital transmission, dynamic spectrum access, and convex optimization and signal processing applied to telecommunications.



Rahul Shrestha (Senior Member, IEEE) received the PhD degree in Electronics and Electrical Engineering from IIT Guwahati, Guwahati, India, in 2014. He is currently an Associate Professor with the School of Computing and Electrical Engineering, IIT Mandi, Mandi, India. His research team works in various domains, such as efficient micro-architecture design for signal-processing, communication and deep-neural-network applications, forward-error-correction channel decoders, cognitive radio, and spectrum-sensing algorithms and architectures.