

Low complexity VLSI Architecture Design methodology for Wigner Ville Distribution

Suresh Mopuri, Amit Acharyya, *Member, IEEE*

Abstract—In this paper, we propose a low complexity VLSI architecture design Methodology for Wigner Ville Distribution (WVD) computation. The proposed methodology performs both auto and cross WVD computations using only the half number of Fast Fourier transform (FFT) computations as opposed to the state of the art methodologies. The FPGA implementation for proposed methodology was performed for 16 bit fixed point and 32 bit single precision floating point numbers on the Xilinx Virtex-7 FPGA (XC7vx485tffg). The proposed methodology saves 49% energy consumption when compared with the state of the art methodology. However it can be noted that the proposed methodology is independent of the VLSI implementation platform and technology node.

Index Terms—Wigner Ville Distribution (WVD), VLSI Architecture, Fast Fourier transform (FFT), Auto WVD, Cross WVD.

I. INTRODUCTION

TIME FREQUENCY signal analysis tools have been used in different practical signal processing applications where the signals are non-stationary [1]-[4]. Wigner Ville Distribution (WVD) is one of the most fundamental tools in quadratic Time Frequency distributions which offers high Time-Frequency (TF) resolution among TF signal analysis tools [5]-[17]. WVD suffers with cross frequency terms which can be removed by applying the approaches proposed in [18]-[22]. However, WVD has been used in the computationally intensive source separation algorithms as for instance Underdetermined Blind Source Separation (UBSS)[5]-[17] where Spatial Time Frequency Distribution matrix generator is one of the most important modules comprising of both Auto WVD (AWVD) and cross WVD (XWVD) of input mixture signals [23]-[24]. Such systems have been found to be useful in the resource constrained applications including remote health monitoring where the device has to be portable and backed up by the limited battery power. In such applications, energy efficiency and low-complexity holistic algorithm-architecture design methodology would play a significant role. Hence, a low-complexity architecture design of AWVD and XWVD would help accomplishing such efficient system designs which is the objective of this paper.

The WVD is non-casual and computationally intensive in nature. The computation of the WVD requires the analytical signal computation [25]-[26], kernel computation and Fast Fourier Transform (FFT) computation [27]. In AWVD computation, the kernel is computed using instantaneous auto correlation of the input signal whereas in XWVD computation, the kernel is computed using instantaneous cross correlation of the input signals. An efficient methodology was reported in [27] for AWVD computation by exploiting the conjugate symmetry property possessed by AWVD kernel resulting reduction of number of FFT computations by half. Another methodology

was reported in [28] for AWVD computation which was faster than [27] for signal length shorter than 1024 samples and FFT length less than 256 points. On the other hand, the XWVD is implemented in [29], [30] using kernel computation followed by FFT computation. However, the XWVD kernel does not have such conjugate symmetry property as that of the AWVD kernel. Hence methodologies reported in [27]-[28] cannot be applicable here for the implementation of the XWVD. The existing XWVD design methodologies as reported in [29], [30] require more number of FFT computations than that is required in AWVD computation as reported in [27], [28]. The state of the art methodologies for WVD computation can be implemented using kernel computation followed by either single FFT or parallel FFTs [27]-[30]. For example, the single FFT implementation for N point AWVD computation [27], [28], the FFT should be reused $\frac{N}{2}$ number of times whereas in parallel FFT implementation requires $\frac{N}{2}$ parallel FFT computations. Similarly for N point XWVD computation [29], [30], the single FFT implementation requires to reuse the FFT N number of times whereas in parallel FFT implementation requires N parallel FFT computations.

Motivated by the fore mentioned arguments and applications, in this paper, we introduce a low complexity design methodology of WVD to minimize the need of significant number of FFT computations resulting reduction of hardware complexity, energy consumption and also resulting the computational speed up of the WVD computation. The highlights of this paper are:

- Exploration of partial conjugate symmetry in the XWVD kernel. The partial conjugate symmetry in XWVD computation saves half the number of FFT computations.
- The proposed methodology works for both AWVD and XWVD computations unlike the state of the art methodologies [28], [30].
- An extensive vector matching is performed to validate the proposed methodology using MATLAB simulation.
- The architecture has been coded in VHDL and FPGA prototyped on the Xilinx Virtex-7 FPGA (XC7vx485tffg). The proposed methodology saves 49% energy consumption when compared with the state of the art methodologies [28], [30].

II. PROPOSED METHODOLOGY

The WVD of any two square integral signals $x_i(t)$ and $x_j(t)$ is defined [27] as

$$\rho_{z_i z_j}(t, f) = \int_{-\infty}^{\infty} z_i^*(t - \frac{\tau}{2}) z_j(t + \frac{\tau}{2}) e^{-j2\pi f\tau} d\tau \quad (1)$$

where $z_i(t)$ and $z_j(t)$ are the analytic associates of $x_i(t)$ and $x_j(t)$. The analytic associate of $x_i(t)$ is defined as $z_i(t) = x_i(t) + j\hat{H}\{x_i(t)\}$, where $\hat{H}\{x_i(t)\}$ represents Hilbert transform of $x_i(t)$. In (1), if $i = j$ then the WVD is called AWVD, otherwise for $i \neq j$, it is called as XWVD. In practical applications XWVD could be intercepted, if $z_i(t)$ is

The manuscript contains a supplementary material. S.M and A.A are with Department of Electrical Engineering, Indian Institute of Technology, Hyderabad, India-502285, e-mails: {ee13p0004 and amit_acharyya}@iith.ac.in. This work is partially supported by the project "i-MOBILYZE" funded by the Xilinx Inc., USA (No: IITH/EE/F091/S8). The design tools were supported under SMDP-C2S program of MEITY, Govt. of India.

a reference signal and $z_j(t)$ is the signal under analysis. For example, in radar, sonar, seismic and acoustic applications, transmitted signal is the reference signal and received echo is the signal under analysis. In other applications like signal characteristic analysis where reference signal is not available, the AWVD is used for spectral analysis [12]-[14]. The discrete equivalent of (1) is called Discrete time WVD (DWVD) and it can be expressed as

$$\rho_{z_i z_j}(n, f) = 2 \sum_{m=-\infty}^{m=\infty} z_i^*(n-m) z_j(n+m) e^{-j2\pi f m} \quad (2)$$

The above DWVD formula is not feasible for real time implementation due to infinite duration in length. To overcome these problems, windowing technique can be applied on the input signals by multiplying with time window function. The windowed DWVD is called as Pseudo DWVD (PDWVD) and expressed as $2 \sum_{m=-M}^{m=M} z_i^*(n-m) z_j(n+m) e^{-j2\pi f m} w^*(-m) w(m) e^{-j2\pi \Delta f m}$, where $w(m)$ is a window function, of length M , symmetric and real. The DWVD kernel sequence at n^{th} instant is defined as

$$k_{z_i z_j}^n(m) = z_i^*(n-m) z_j(n+m) w^*(-m) w(m) \quad (3)$$

Still, the PDWVD formula in (2) is not feasible for implementation because the frequency variable f is a continuous variable. After sampling the frequency variable f for length N , (2) can be rewritten as

$$\rho_{z_i z_j}(n, l) = 2 \sum_{m=-M}^{m=M} k_{z_i z_j}^n(m) e^{-j2\pi l \Delta f m} \quad (4)$$

where $\Delta f = \frac{1}{N} = \frac{1}{2M+1}$ [27]. Now the PDWVD formula shown in (4) is feasible for implementation. The PDWVD shown in (4) can be performed using using FFT. The FFT requires the input sequence indexed from 0 to $N-1$ but (4) has the input sequence from $-M$ to M . The positive indexing can be achieved by making a periodic extension of the kernel sequence considering $N = 2M+1$ [27]. The extended kernel sequence is expressed in following equation [27]

$$k_{z_i z_j}^n(m) = \begin{cases} k_{z_i z_j}^n(m) & 0 \leq m \leq \frac{N}{2} - 1 \\ k_{z_i z_j}^{n*}(m-N) & \frac{N}{2} + 1 \leq m \leq N-1 \\ 0 & m = \frac{N}{2} \end{cases} \quad (5)$$

The PWVD computation shown in (4) is similar to the FFT computation. In general, the FFT computations are optimized based on symmetry properties of the input signal. Hence, in this proposed methodology, the number of FFT computations can be optimized based on the symmetry properties of the kernel sequence $k_{z_i z_j}^n(m)$. First, we investigate the properties of AWVD kernel sequence when $i = j$. The PDWVD kernel shown in (5) possess conjugate symmetry property when $i = j$

$$k_{z_i z_i}^n(m) = k_{z_i z_i}^{n*}(-m) \quad (6)$$

For conjugate symmetry kernel, the FFT will be real. Let the PDWVD kernel sequences at two successive instances n and $n+1$ are combined as follows

$$k_{z_i z_i}^{\text{comb}}(m) = k_{z_i z_i}^n(m) + j * k_{z_i z_i}^{n+1}(m) \quad (7)$$

Then FFT of combined kernel sequence can be computed using following equation

$$K_{z_i z_i}^{\text{comb}}(l) = FFT(k_{z_i z_i}^{\text{comb}}(m)) \quad (8)$$

Now the individual FFT of kernel sequences $K_{z_i z_i}^n(l)$ and $K_{z_i z_i}^{n+1}(l)$ can be computed using single FFT as follows

$$\rho_{z_i z_i}(n, l) = K_{z_i z_i}^n(l) = \text{real}(K_{z_i z_i}^{\text{comb}}(l)) \quad (9)$$

$$\rho_{z_i z_i}(n+1, l) = K_{z_i z_i}^{n+1}(l) = \text{imag}(K_{z_i z_i}^{\text{comb}}(l)) \quad (10)$$

From (9) and (10), it is important to note that by combining successive pairs of kernel sequences $k_{z_i z_i}^n(m)$ and $k_{z_i z_i}^{n+1}(m)$, the number of FFT computations will be halved in AWVD computation.

Now we will investigate the properties of PDWVD kernel sequence when $i \neq j$ for XWVD computation. The $k_{z_i z_j}^n(m)$ cannot possess conjugate symmetry property as shown in (6). Hence, we define a virtual kernel sequence $p(m)$ having length N using $k_{z_i z_j}^n(m)$. The $k_{z_i z_j}^n(m)$ and $p(m)$ have the following relation.

$$p(m) = \begin{cases} k_{z_i z_j}^n(m) & 1 \leq m \leq N-1 \\ \text{real}(k_{z_i z_j}^n(0)) & m = 0 \end{cases} \quad (11)$$

From (5), it is important to note that the $k_{z_i z_j}^n(m)$ possess conjugate symmetry property except at $m = 0$ when $i \neq j$. Therefore, the virtual kernel $p(m)$ possess conjugate symmetry property i.e. $p(m) = p^*(-m)$.

The FFT of $k_{z_i z_j}^n(m)$ can be computed as follows:

$$K_{z_i z_j}^n(l) = \sum_{m=0}^{m=N-1} k_{z_i z_j}^n(m) e^{-j2\pi l m} \quad (12)$$

By substituting (5) in (12) and expanding (12), we get:

$$K_{z_i z_j}^n(l) = k_{z_i z_j}^n(0) + \sum_{m=0}^{m=\frac{N}{2}-1} k_{z_i z_j}^n(m) e^{-j2\pi l m} + \sum_{m=\frac{N}{2}+1}^{m=N-1} k_{z_i z_j}^n(m) e^{-j2\pi l m} \quad (13)$$

The $k_{z_i z_j}^n(m)$ satisfies the conjugate symmetry as shown in (6) except at $m = 0$. Then (13) can be rewritten as follows:

$$K_{z_i z_j}^n(l) = k_{z_i z_j}^n(0) + \sum_{m=0}^{m=\frac{N}{2}-1} k_{z_i z_j}^n(m) e^{-j2\pi l m} + \sum_{m=0}^{m=\frac{N}{2}-1} k_{z_i z_j}^{n*}(m) e^{j2\pi l m} \quad (14)$$

From (14), the real and imaginary parts of $K_{z_i z_j}^n(l)$ can be computed as follows:

$$\text{real}(K_{z_i z_j}^n(l)) = \text{real}(k_{z_i z_j}^n(0)) + \sum_{m=0}^{m=\frac{N}{2}-1} k_{z_i z_j}^n(m) e^{-j2\pi l m} + \sum_{m=0}^{m=\frac{N}{2}-1} k_{z_i z_j}^{n*}(m) e^{j2\pi l m} \quad (15)$$

$$\text{imag}(K_{z_i z_j}^n(l)) = \text{imag}(k_{z_i z_j}^n(0)) \quad (16)$$

The FFT of $p(m)$ as computed as follows:

$$P(l) = \sum_{m=0}^{m=N-1} p(m) e^{-j2\pi l m} \quad (17)$$

By substituting (11) in (17), we get:

$$P(l) = \text{real}(k_{z_i z_j}^n(0)) + \sum_{m=0}^{m=\frac{N}{2}-1} k_{z_i z_j}^n(m) e^{-j2\pi l m} + \sum_{m=0}^{m=\frac{N}{2}-1} k_{z_i z_j}^{n*}(m) e^{j2\pi l m} \quad (18)$$

From (15), (16) and (18) the relation between $K_{z_i z_j}^n(l)$ and $P(l)$ can be defined as follows

$$\begin{aligned} \text{real}(K_{z_i z_j}^n(l)) &= P(l) \\ \text{imag}(K_{z_i z_j}^n(l)) &= \text{imag}(k_{z_i z_j}^n(0)) \end{aligned} \quad (19)$$

Consider the XWVD kernel sequences at two successive instances n and $n+1$ which are $k_{z_i z_j}^n(m)$ and $k_{z_i z_j}^{n+1}(m)$ respectively. From (11), their virtual kernel sequences $p^n(m)$ and $p^{n+1}(m)$ can be expressed as follows:

$$p^n(m) = \begin{cases} k_{z_i z_j}^n(m) & 1 \leq m \leq N-1 \\ \text{real}(k_{z_i z_j}^n(0)) & m = 0 \end{cases} \quad (20)$$

$$p^{n+1}(m) = \begin{cases} k_{z_i z_j}^{n+1}(m) & 1 \leq m \leq N-1 \\ \text{real}(k_{z_i z_j}^{n+1}(0)) & m = 0 \end{cases} \quad (21)$$

The XWVD kernel $k_{z_i z_j}^n(m)$ possess conjugate symmetry except at $m = 0$. Hence the defined virtual kernels possess conjugate symmetry property as shown in (6). The $p^n(m)$ and $p^{n+1}(m)$ can be combined as follows:

$$p^{comb}(m) = p^n(m) + j * p^{n+1}(m) \quad (22)$$

Then the FFT of combined kernel sequence can be computed using following equation:

$$P^{comb}(l) = FFT(p^{comb}(m)) = P^n(l) + j * P^{n+1}(l) \quad (23)$$

Now using (19) and (23) the individual FFT of XWVD kernel sequences $K_{z_i z_j}^n(l)$ and $K_{z_i z_j}^{n+1}(l)$ can be computed as follows:

$$\begin{aligned} \rho_{z_i z_j}(n, l) &= K_{z_i z_j}^n(l) = P^n(l) + j * \text{imag}(K_{z_i z_j}^n(l)) \\ &= \text{real}(P^{comb}(l)) + j * \text{imag}(k_{z_i z_j}^n(0)) \end{aligned} \quad (24)$$

$$\begin{aligned} \rho_{z_i z_j}(n, l) &= K_{z_i z_j}^{n+1}(l) = P^{n+1}(l) + j * \text{imag}(K_{z_i z_j}^{n+1}(l)) \\ &= \text{imag}(P^{comb}(l)) + j * \text{imag}(k_{z_i z_j}^{n+1}(0)) \end{aligned} \quad (25)$$

From (22), (23), (24) and (25), it can be noted that by combining successive pairs of virtual kernel sequences $p^n(m)$ and $p^{n+1}(m)$, the number of FFT computations will be halved in XWVD computation.

Subsequently an architecture has been designed for both AWVD and XWVD computation based on (20)-(25) is depicted in Fig.1(a). Here the analytical signals are computed using re-configurable DHT presented in [25]. The functionality of architecture is explained with the help of pseudo code as shown in Algorithm1 (Please see the supplementary material). Two port memories are used in this design for storing the input signals $z_i(n)$ and $z_j(n)$. The controller module consists of counters to generate the memory indices m , $m+n$, $m-n$ as shown in pseudo code (Algorithm1) (Please see the supplementary material). This module also generates the read write signals for memories, FIFO and control signals for other sub modules based on the proposed methodology. The kernel sequence $k_{z_i z_j}^n(m)$ is computed using (3) at n and $n+1$ instants. The window coefficients are precomputed and stored in the window coefficient memory. The stored coefficients are fetched and multiplied as shown in (3). By using (20) and (21), the corresponding virtual sequence $p^n(m)$ and $p^{n+1}(m)$ are computed. In this virtual sequence computation, the imaginary parts of $k_{z_i z_j}^n(m)$ and $k_{z_i z_j}^{n+1}(m)$ at $m = 0$ are stored in a buffer. If $i = j$, these imaginary parts will become 0. The imaginary part selection module is designed to select the

imaginary part of WVD as shown in (24) and (25). The virtual kernel sequence $p^n(m)$ and $p^{n+1}(m)$ are combined using (22). Combining of the virtual kernel sequences eliminates the additional computation of FFT in the WVD design.

III. EXPERIMENTAL RESULTS AND DISCUSSION

A. Verification of the proposed methodology

In this subsection, before implementing the proposed methodology on hardware, we modeled in MATLAB for double precision floating point numbers to check the functionality and simulate the absolute errors. The functionality of the AWVD and XWVD is cross verified with ‘‘WVD function’’ and ‘‘XWVD function’’ in Time Frequency Tool Box (TFTB) of MATLAB [31]. The MATLAB model is simulated by supplying the non-stationary input signals such as chirp signals, ECG and EEG signals to validate the accuracy of the proposed hardware design in comparison with the software version of the WVD. For example, the AWVD computation of a linear chirp signal of bandwidth 2 MHz and 100 microseconds in length is shown in Fig.S1(a) (Please see the supplementary material). The XWVD computation between the chirp signal and Gaussian signal of bandwidth 1.4 MHz and 100 microsecond in length is shown in Fig.S1(d) (Please see the supplementary material). The AWVD and XWVD using TFTB are shown in Fig.S1(b) and Fig.S1(e) respectively (Please see the supplementary material). The ‘‘WVD function’’ and ‘‘XWVD function’’ in TFTB are implemented using the state of the art methodologies[27], [29], [30]. The absolute error between the proposed method and state of the art [27], [29], [30] is zero. Therefore, the proposed methodology does not have any negative impact on the functionality but optimizes the requirement of number of FFT computations. Similarly, the MATLAB model for the proposed methodology is tested for different input lengths and for different input signals such as non-linear chirp, ECG and EEG signal shown in Fig.S2 (ECG signal for 120 seconds, EEG signal for 100 seconds linear Chirp and logarithmic chirp signals for duration of 200 micro seconds and band width of 4 MHz) (Please see the supplementary material).

B. FPGA Implementation of the proposed methodology

The proposed methodology is independent of either single FFT or parallel FFTs implementation and saves half number of FFT computations. Therefore the proposed methodology can be implemented using either single FFT or parallel FFTs. In this paper, the proposed architecture, state of the art architectures [27], [28] and [30] are implemented using kernel computation followed by single FFT structure as shown in Fig.1(a) to compare the design performance on a uniform platform. The proposed architecture and state of the art architectures [28]-[30] are coded in VHDL for signal length $T = 1024$, FFT length $N = 512$ (minimum frequency resolution required is $\frac{T}{2} = 512$). The proposed methodology is independent of technology node, fixed or floating point implementation and saves the half number of FFT computations when compared with the state of the art methodology. However, to provide more insight to the reader the proposed methodology is implemented for both fixed point (Fxp) and floating point (FP). The FPGA prototype for the proposed architecture has been done on VC707 evaluation platform which is having the

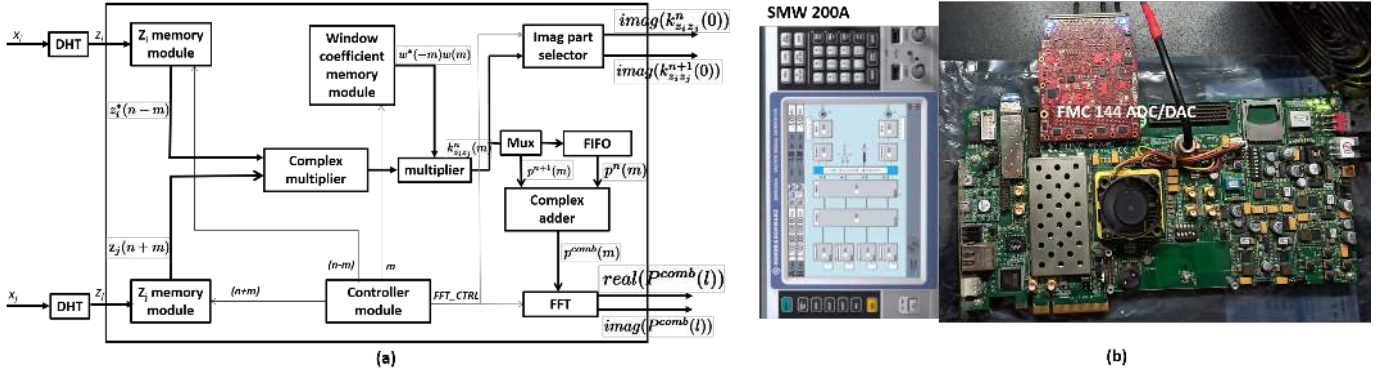


Fig. 1. (a) WVD Architecture for AWVD and XWVD computation based on the proposed methodology (b) Experimental setup for verification of the proposed methodology using Xilinx Virtex-7 FPGA (XC7vx485tffg) and FMC 144ADC/DAC card

Xilinx Virtex-7 FPGA (XC7vx485tffg). The implementation results are shown in Table I. The FPGA implementation for proposed methodology is performed for 16 bit fixed point and 32 bit single precision floating point.

As shown in the Table I, the proposed method requires 0.904% higher LUT requirement than [28] but the methodology in [28] performs only AWVD computation whereas proposed methodology performs both AWVD and XWVD computations. On the other hand, the proposed methodology requires 0.904% higher LUT requirement than [30]. However, it is found that the proposed methodology saves 50% FFT computations resulting 49% of reduction in the energy consumption. It can be noted that if the proposed methodology and state of the architectures are implemented using kernel computation followed by parallel FFT structure, the state of the methodology requires more FPGA resources than proposed methodology. In this example, the proposed methodology requires the resources for 512 FFTs whereas state of the art methodology requires the resources for 1024 FFTs. The parallel FFTs structure implementation requires more area, energy consumption and faster than the single FFT implementation. In order to provide more insight on the FPGA implementation for different signal lengths, the proposed methodology is implemented for signal length $T = 2048$, FFT lengths $N = 1024$ and $N = 2048$. The synthesis results are tabulated in Table SI. As signal dimension increases, the resource requirement will increase however the proposed methodology saves the half number of FFT computations irrespective of signal length and FFT length.

The Hardware implementation of the proposed methodology is tested using the experimental setup shown in Fig.1(b). The SMW200A is a dual channel vector signal generator which consists of arbitrary waveform memory which provides an option to generate any waveform. For example, the chirp signals, ECG and EEG signals generated in MATLAB simulation section III-A can be stored and playback. The SMW200A is stored with chirp signals for pulsed of 100 microseconds. The SMW200A is connected to VC707 board through FMC144 ADC/DAC quad channel daughter card. THE FMC 144 is tuned to 10 MHz sampling clock using FMC analyzer software. The VHDL reference design of FMC144 is integrated with proposed VHDL design and generated a bit file. The functionality is verified using chip scope.

Accuracy of the proposed design is determined by comparing outputs from FPGA with modeled function in MATLAB.

The HDL outputs for the proposed AWVD and XWVD are shown in Fig.S1(c) and Fig.S1(f) respectively for the same test bench used in MATLAB simulation. The signature of spectrogram in the MATLAB simulation and VHDL simulations is same for the respective AWVD and XWVD computations. However, to provide insight into the accuracy of the proposed architecture, Mean Square Error (MSE) and signal to noise ratio (SNR) are computed using the approach in [32]-[33]. The SNR and MSE variations with respect to word length is shown in Fig.2. From Fig.2, the observed MSE is 1.2×10^{-12} and SNR is 120dB for both proposed and state of the art AWVD and XWVD computations. The MSE and SNR for the proposed and state of the art architecture are same due to both followed similar methodology but the proposed method saves 50% FFT computations.

TABLE I
PERFORMANCE OF PROPOSED ALGORITHM ON FPGA

| Parameter | Proposed | AWVD [27] | AWVD [28] | XWVD [30] | Proposed |
|------------------------------------|-------------|------------|------------|------------|------------|
| AWVD / XWVD | AWVD & XWVD | AWVD | AWVD | AWVD& XWVD | AWVD& XWVD |
| Word length | Fxp 16 bit | Fxp 16 bit | Fxp 16 bit | Fxp 16 bit | FP 32 bit |
| Number of FFT computations (F) | 512 | 512 | 512 | 1024 | 512 |
| Maximum clock frequency (f) | 280MHz | | | | 172MHz |
| Energy consumption (nano Joule) | 68.2423 | 68.2423 | 67.9954 | 135.99 | 330.13 |
| Slice LUTs | 7521 | 7453 | 7453 | 7453 | 34740 |
| DSP slices | 35 | 35 | 35 | 35 | 83 |
| 18k BRams | 12 | 12 | 12 | 12 | 25 |
| 36k BRams | 2 | 2 | 2 | 2 | 6 |
| SNR | 120 | 119.89 | 120 | 120.1 | 123.25 |

The timing of the proposed methodology is analyzed in terms of the latency and execution time in the supplementary material. For example, for $T = 1024$, the time required for kernel computation (T_{kernel}) is 1024 clock cycles. From (S3), the latency of the proposed methodology is 1024 clock cycles more than the state of the art methodology [30]. However, from

(S6), $ETS = 512 * T_{FFT}$ so that the proposed methodology saves the ET of 512 number of FFTs when compared with the state of the art methodology [30].

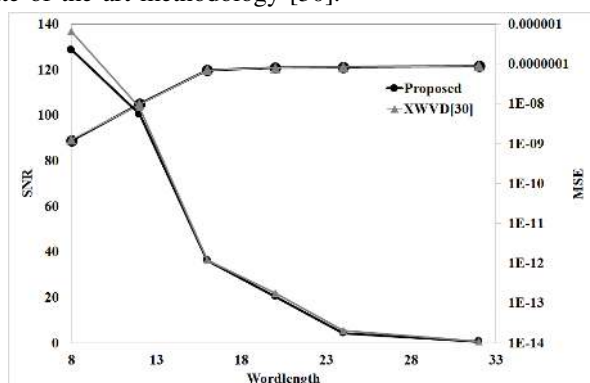


Fig. 2. SNR and MSE vs Word length for the proposed methodology

IV. CONCLUSION

In this paper, we proposed an architecture design methodology for low complex Wigner Ville Distribution (WVD) computation which performs both AWVD and XWVD and saves 50% FFT computations when compared with the state of the art methodologies. It also significantly reduces the energy consumption by approximately 49%. The architecture based on the proposed methodology has been designed and implemented on FPGA (Xilinx Virtex-7 FPGA (XC7vx485tffg)) platform and performance has been evaluated in terms of area, energy consumption and accuracy of the algorithm. The proposed methodology presents the details for implementation of basic WVD which suffers with cross frequency terms. The approaches introduced in [18]-[22] require the WVD, the proposed methodology can be used for the implementation of the WVD and the approaches [18]-[22] can be applied to reduce the cross frequency terms.

REFERENCES

- [1] A. Acharyya et al., "Memory Reduction Methodology for Distributed-Arithmetic-Based DWT/IDWT Exploiting Data Symmetry," in IEEE Transactions on Circuits and Systems II: Express Briefs, vol. 56, no. 4, pp. 285-289, April 2009.
- [2] N. Zarić et al., "An Implementation of the L-Estimate Distributions for Analysis of Signals in Heavy-Tailed Noise," in IEEE Transactions on Circuits and Systems II: Express Briefs, vol. 58, no. 7, pp. 427-431, July 2011.
- [3] I. Orović et al., "A Suitable Hardware Realization for the Cohen Class Distributions," in IEEE Transactions on Circuits and Systems II: Express Briefs, vol. 60, no. 9, pp. 607-611, Sept. 2013.
- [4] L. Stanković, "On the STFT Inversion Redundancy," in IEEE Transactions on Circuits and Systems II: Express Briefs, vol. 63, no. 3, pp. 284-288, March 2016.
- [5] L. F. Chaparro et al., "Asynchronous Representation and Processing of Nonstationary Signals: A Time-Frequency Framework," in IEEE Signal Processing Magazine, vol. 30, no. 6, pp. 42-52, Nov. 2013.
- [6] K. M. M. Prabhu et al., "Fixed-point error analysis of discrete Wigner-Ville distribution," in IEEE Transactions on Signal Processing, vol. 45, no. 10, pp. 2579-2582, Oct. 1997.
- [7] E. Palmer et al., "Preliminary investigation into the use of the cross Wigner-Ville distribution to detect high impedance earth faults in power systems," Signal Processing and Its Applications, 2003. Proceedings. Seventh International Symposium on, 2003, pp. 597-600 vol.1.
- [8] S. Rajagopalan et al., "Nonstationary Motor Fault Detection Using Recent Quadratic Time-Frequency Representations," in IEEE Transactions on Industry Applications, vol. 44, no. 3, pp. 735-744, May-june 2008.
- [9] F. Auger et al., "Time-Frequency Reassignment and Synchrosqueezing: An Overview," in IEEE Signal Processing Magazine, vol. 30, no. 6, pp. 32-41, Nov. 2013.
- [10] Sharma et al., "Improved Eigenvalue Decomposition-Based Approach for Reducing Cross-Terms in Wigner-Ville Distribution." Circuits, Systems, and Signal Processing 37.8 (2018): 3330-3350.
- [11] C. Y. Mei et al., "Classification of digitally modulated signals using cross time-frequency distribution," 2015 IEEE International Conference on Signal and Image Processing Applications (ICSIPA), Kuala Lumpur, 2015, pp. 1-6.
- [12] S. Barbarossa et al., "Detection and imaging of moving objects with synthetic aperture radar. 2. Joint time-frequency analysis by Wigner-Ville distribution," in IEE Proceedings F - Radar and Signal Processing, vol. 139, no. 1, pp. 89-97, Feb 1992.
- [13] N. Allan et al., "Dualpolarized Doppler radar measurements of oceanic fronts," in IEEE Transactions on Geoscience and Remote Sensing, vol. 37, no. 1, pp. 395-417, Jan 1999.
- [14] D. Borio et al., "Time-Frequency Excision for GNSS Applications," in IEEE Systems Journal, vol. 2, no. 1, pp. 27-37, March 2008.
- [15] K. Cai et al., "Semi-Blind Fetal Electrocardiogram Extraction by Eliminating the Cross-Terms of the Wigner-Ville Representations," Bioinformatics and Biomedical Engineering, (iCBBE) 2011 5th International Conference on, Wuhan, 2011, pp. 1-4.
- [16] A. Monti et al., "Instantaneous parameter estimation in cardiovascular time series by harmonic and time-frequency analysis," in IEEE Transactions on Biomedical Engineering, vol. 49, no. 12, pp. 1547-1556, Dec. 2002.
- [17] C. Guanghua et al., "Wigner-Ville distribution and cross Wigner-Ville distribution of noisy signals," in Journal of Systems Engineering and Electronics, vol. 19, no. 5, pp. 1053-1057, Oct. 2008.
- [18] Ram Bilas Pachori et al., "Cross-terms reduction in the Wigner-Ville distribution using tunable-Q wavelet transform, Signal Processing, Volume 120, 2016, Pages 288-304, ISSN 0165-1684.
- [19] Pooja Jain et al., "Marginal energy density over the low frequency range as a feature for voiced/non-voiced detection in noisy speech signals, Journal of the Franklin Institute, Volume 350, Issue 4, 2013, Pages 698-716, ISSN 0016-0032.
- [20] Varun Bajaj et al., "Automatic classification of sleep stages based on the time-frequency image of EEG signals, Computer Methods and Programs in Biomedicine, Volume 112, Issue 3, 2013, Pages 320-328, ISSN 0169-2607.
- [21] Ram Bilas Pachori et al., "A new technique to reduce cross terms in the Wigner distribution, Digital Signal Processing, Volume 17, Issue 2, 2007, Pages 466-474, ISSN 1051-2004.
- [22] Sharma, R.R. et al., "An empirical wavelet transform-based approach for cross-terms-free Wigner-Ville distribution. SIViP (2019).
- [23] S. Mopuri, et al., "Low complexity underdetermined blind source separation system architecture for emerging remote healthcare applications," 2014 36th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, Chicago, IL, 2014, pp. 3833-3836.
- [24] S. Mopuri et al., "Fast underdetermined BSS architecture design methodology for real time applications," 2015 37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), Milan, 2015, pp. 5408-5411.
- [25] P. S. Reddy et al., "A Reconfigurable High Speed Architecture Design for Discrete Hilbert Transform," in IEEE Signal Processing Letters, vol. 21, no. 11, pp. 1413-1417, Nov. 2014.
- [26] Suresh Mopuri et al., "Low-complexity and Reconfigurable DHT Architecture Design Methodology" Journal of Low Power Electronics, American Scientific Publishers, Vol. 14, No: 2, June 2018.
- [27] B. Boashash et al., "An efficient real-time implementation of the Wigner-Ville distribution," in IEEE Transactions on Acoustics, Speech, and Signal Processing, vol. 35, no. 11, pp. 1611-1618, Nov 1987.
- [28] Hua, X., et al., "A novel fast algorithm for the pseudo Wigner-Ville distribution." Journal of Communications Technology and Electronics 60.11 (2015).
- [29] Boualem Boashash. 2003. Time Frequency Analysis(1st ed.). Elsevier Science, San Diego, USA.
- [30] Boashash B. Time-frequency signal analysis and processing: a comprehensive reference. Academic Press; 2015 Dec 11.
- [31] Boualem Boashash, et al., Efficient software platform TFSAP 7.1 and Matlab package to compute Time-Frequency Distributions and related Time-Scale methods with extraction of signal characteristics, SoftwareX, Volume 8, 2018, Pages 48-52, ISSN 2352-7110.
- [32] Sang Yoon Park et al., "Fixed-point error analysis of CORDIC processor based on the variance propagation formula," in IEEE Transactions on Circuits and Systems I: Regular Papers, vol. 51, no. 3, pp. 573-584, March 2004.
- [33] W. Chang et al., "On the Fixed-Point Accuracy Analysis of FFT Algorithms," in IEEE Transactions on Signal Processing, vol. 56, no. 10, pp. 4673-4682, Oct. 2008.