A COMMON OPERATOR ARCHITECTURE FOR DISCRETE WAVELET TRANSFORM AND CONVOLUTIONAL ENCODER

V.RAJESH[1] AND V.PALANISAMY[2]
[1]Assistant Professor, Department of Electrical &Electronics Engineering
Mahendra Institute of Technology, Namakkal, Tamilnadu
[2]Principal, Info Institute of Engineering, Coimbatore, Tamilnadu
E-mail: [1]rajeshsrija@gmail.com

ABSTRACT

In Software defined radio system, the transform coding is performed by data compression and encryption techniques. In this paper, a parameterization approach is considered to represent a transform coding technique, this was formulated and represented over Discrete Wavelet Transform (DWT) and convolutional coding. Thus, a flexible multi-standard transform coding system is designed under parameterization approach which explores the architecture to extract the operators to be designed. Here, two different standards for whom the Common Operator is designed are Lattice DWT and Convolutional Encoder, and the hardware complexity for this multi-standard approach is reduced by 32.48% and the power analyzed is lowered by 47.09%.

Keywords: *Common Operator, Parameterization, Discrete Wavelet Transform (DWT), Convolutional Encoder, Lattice.*

1. INTRODUCTION

In the last few years, a proliferation of communication standards has substantially increased the complexity of the mobile radios. In typical designs, the communication standards are implemented separately using dedicated representations which are difficult to upgrade for their support of new features. In the present days, the concept of Software Radio (SWR), prefaced in [13], emerged from military research to become a cornerstone of modern communication systems. The SWR technique becomes the way to design flexible and reconfigurable architectures capable of supporting different transmission standards in a single platform. However, although there is a common agreement on the SWR aim and benefit. The way of implementing SWR, also known as Software Defined Radio (SDR) varies, considering various tradeoffs such as cost, flexibility, complexity, power consumption, speed, etc. requested by actual design.

Software-defined radio (SDR) has emerged as a revolutionary approach that offers a flexible mechanism for developing and operating communication radios to implement radio functionality, such as signal generation, coding, and modulation for a broad range of domains including commercial, military, and public service[22, 23]. Advances in digital signal processing and computing power have enabled the evolution of radio implementations from primarily electronically based to firmware and/or software-based. Each of these application domains has leveraged reconfigurable hardware to satisfy the specific desired operating requirements [23, 4–4]. This gives rise to the possibility of adapting the radio to user’s preferences and the operating environment and of supporting multiple standards without requiring separate hardware for each standard. In many of these domains, engineers have pursued standards or other sets of rules to establish the policies required to achieve a desired degree of commonality within the domain for development and operations [23, 3–7, 11].

A digital communication chain, when supporting different standards, uses typical signal processing operations such as modulation, channel coding, equalization, etc. These common functions can be identified and then explored to take advantage from the commonalities among common tasks in order to enhance power efficiency and area utilization [11]. In this context, parameterization technique has been introduced in [4] and [3]. It consists in identifying the common aspects among the targeted modes and standards in order to define a generic operation capable of handling the required tasks. This generic operation can switch from a configuration to another by a simple change of its parameters.

A work on parameterization approach is presented in this paper. There are lot of works presented on parameterization technique in [12], [11], that can be reckoned to construct a general method subject of endorsing a vast array of communication standards, this approach is known as the Common Operator Technique. The fundamental approach of the Common Operator technique is to distinguish the common elements built-up on tiny systems that could be reused to a considerable degree throughout the functions. Depending upon medium granularity operators, the common operator method targets to project reconfigurable structures larger than basic logic cells and smaller than Velcro Method or Common function[13]. Likewise, a Common Operator similar to flip flop or logic gate can be used irrespective of the function carried out by. From this point, the CO operator technique claims to be less standard dependent than classical approach [6] where the entire specific element required by a standard are implemented and executed when needed. It is expected that the reduction of the exploration space to signal processing functions will help defining these Common Operators. Thus, we have designed a common operator for transform and encoder under transform coding in SDR. In communication system, the signal transmission and reception is the main concern and hence to perform healthy communication data compression and encryption at transmission side as well data decompression and decryption at the receiver side is necessary. Our intention is to design and implement a parameterization approach for compression and encryption present at the transmission side of the SDR. The resulted implementation is believed to be flexible and configurable to a broad array of measures.

Previously, several works were focused on defining [9, 12] implementing and managing [16] the Common Operators. The works presented in literature are mostly designed as the common operator or reconfigurable cells for either the Viterbi and FFT algorithms or the FEC code and FFT algorithms. Our work deals with, combining two widely used algorithms in wireless communication systems: Convolutional code and DWT algorithms. A configurable cell that can be reprocessed throughout these functions is proposed. The proposed common operator is a configurable cell that pools convolutional encoder and lattice DWT algorithm. The literature, research methodology, proposed concept, results and conclusion are described in the following sections. Initially, the literature survey for the available common operator is presented in section 2. With the information and approach presented in the literature, the research methodology is summarized in section 3. Based on the concept presented in previous section we design a common operator for Encoder and DWT in section 4 and for the same, results generated are briefly presented in section 5 and finally, the paper is concluded.

2. RELATED WORK

Here, the reviews of recent researches are available in the literature for a Common Operator Technique on FPGA platform is presented.

Earlier in 2006 J.Takala et.al [9] presented partial-column radix-2 FFT processors and realizations of butterfly operations. Mean while in 2007 *S.T. Gul et. al* [12] presented an approach with two scenarios for designing flexible multi-standard radio systems. In [9] The area and power-efficiency of butterfly units to be used in their processor organization based on bit-parallel multipliers, distributed arithmetic and CORDIC were analyzed and compared. Their processor organization permits the area of the FFT implementation to be traded against the computation time, thus the final structure was easily tailored according to the requirements of the given application. The power consumption comparisons showed that butterflies based on bit-parallel multipliers were power-efficient but have limitations on clock frequency. Butterflies based on distributed arithmetic could be used when higher clock frequencies were used. If extremely long FFTs were needed, the CORDIC based butterflies were applicable. Whereas, the multi-standard radio systems in [12] explored the design of multi-standard systems at different levels of granularity and selected the convenient level depending on each designer's needs. Their approach was initially to prepare a graph description and then explore the architecture to extract the operators of the multi-standard system to be designed. These operators were requested to have enhanced reconfiguration capabilities so that a fast reconfiguration was implementable. Furthermore their approach presented the scheduling issues related to their design approach.

Recently in year 2010, *Malek Naoues et. al* [18] designed a common operator for the FFT and Viterbi algorithms and also for FFT and FEC decoding algorithms in [17]. Both of these common operators benefited their regular architecture open to future function mapping and adapted to accommodate silicon technology variability through dependable design. Thereafter, A Multi Standard Terminal was defined by *L. Alaus et.al* [15], their refined Common Operator (CO) technique enhanced the reconfigurability and the scalability of the design but this lead to a complex management of data dependencies and scheduling of each operator for its correct execution in the terminal. The COB created a scalable design, and reduced the number of operators limiting scheduling of each operator. They applied a tri-standard terminal, which lowered the hardware complexity by up to 40%. And the Common Functions and Common Operators were the two different techniques presented by *L. Alaus et.al* [13]. They considered the parameterization as a digital radio design methodology; this was developed and illustrated over two well-known functions such as, Fast Fourier Transform (FFT) and the Reconfigurable Linear Feedback Shift Register (R-LFSR), which was derived from the classical Linear Feedback Shift Register (LFSR) structure.

3. RESEARCH METHODOLOGY

My work is to present a research methodology based upon the parameterization approach that defines the design and implementation of a common operator technique or a reconfigurable cell. Earlier in the literature, I have discovered that there are few algorithms utilized under wireless communication systems for which a CO is defined. They in particular are designed [18] to present a common structure for the FFT and Viterbi algorithms. This is benefited the regularity in architecture which made it open to function mapping and adopted to accommodate silicon technology variability through dependable design. A multi-standard terminal based on a limited set of common operators is defined [15]. This approach in [6] enhanced the reconfigurability and the scalability but lead to complex management of data dependencies and scheduling of each operator for its correct execution in the terminal. Then a COB was utilized which undo the complexity issue upto some extent. In [12] a flexible multi-standard radio system was designed at different levels of granularity and selected the convenient level. And to enhance the reconfigurable capabilities there by implementing the fast reconfigurable cell, their approach first represented a graph description of the multi-standard system to be designed and then, an architectural consideration drew out the operators of a given multi-standard device. A parameterization technique for digital radio design was presented in [13] where, common function and common operator techniques were considered. This approach was illustrated with FFT and the Reconfigurable Linear Feedback Shift Register (R-LFSR). Thus, motivated from all these concepts presented by respective authors in the literature, in this paper a Common Operator for the data compression block at the transmitter side in transform coding of SWR will be designed. This block will transform the original data using DWT (Analysis part of discrete wavelet transform) which will produce coefficients as the information in the source signal. But this will not compress the data and thus the transformed data i.e., coefficients are compressed by the coder. There are various types of coders available for implementation but the best suitable will be designed for common operator architecture. The CO architecture for the Data Compression will be designed in Verilog and implemented will be synthesized in Xilinx ISE tool. The CO for the Data Compression block designed will provide effective results.

4. PROPOSED SYSTEM

Transform coding is a data compression technique. This technique is commonly used over data like audio signals or image information. Transform coding is carried out in two steps. Transform signal using different transform classes and then encoding the information using dedicated encoding techniques. For example a transform coding of image information into JPEG image format.

Block diagram of transform coding

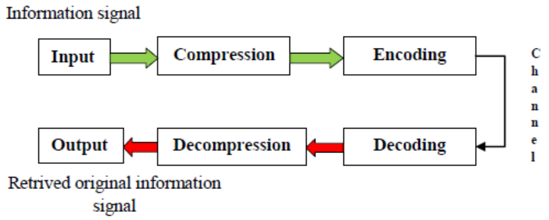


Fig. 1 Transform Coding Process In Block Representation

The Compression of transform coding technique may compress the following media types must be; Data, Audio, Video, Image, etc. In a digital system, the bit rate is the product of the sampling rate and the number of bits in each sample. The difference between the information rate of a signal and its bit rate is known as the redundancy. Compression systems are designed to eliminate this redundancy. These redundancies can be coding redundancies, perceptual redundancy, temporal redundancy, etc. For these compression inputs some of the following methods can be adopted viz; RLE, Huffman encoding, sub-band coding, CELP, JPEG, Wavelet, MPEG. In this paper, as we are designing a common operator for the SWR we have to compress an audio signal. Because the growth of the computer industry has invariably led to the demand for quality audio data, we consider Wavelet transform for compression of the audio signal. The DWT analysis side will be considered for the compression of the input signal.

Daubechies-4 lattice DWT

For signal processing and analysis the preferred tool by researchers is the DWT, because it is a powerful temporal resolution tool [21-2]. Passing, a signal $X(n)$ through a series of filters is the process for computing DWT of a signal. Passing, the samples through a low pass filter with impulse response $H_0(n)$contributes to approximation coefficients while a high pass filter $G_0(n)$ decomposes a signal simultaneously which contributes to the detailed coefficients. A processing element of the lattice DWT structure, for both DWT and IDWT consists of two adders and two multipliers. A delay element exists ahead of the PE in DWT and it follows the PE in IDWT with a scaling factor at the end of both DWT and IDWT.

$$\begin{bmatrix} y_0 \\ y_1 \end{bmatrix} = H_l(z) \cdot \begin{bmatrix} x_e \\ z^{-1} x_o \end{bmatrix} \tag{1}$$

The lattice daubechies-4 IDWT processing element is represented in Fig. 2. Fig.2: The lattice DWT processing element consists of $x_e(i)$ and $x_o(i)$ even and odd inputs of the i^{th} PE, α_i is the lattice coefficient at the i^{th} PE, and $y_0(i)$ & $y_1(i)$ are the low pass and High pass outputs of the i^{th} PE, respectively. x_e and X_o are the even and odd samples of input X , and $H_l(z)$ is the lattice matrix.

$$H_l(z) = \begin{bmatrix} \beta(1-\alpha_0\alpha_1 z^{-1}) & \beta(\alpha_0+\alpha_1 z^{-1}) \\ \beta(-\alpha_1 z-\alpha_0) & \beta(-\alpha_0\alpha_1 z+1) \end{bmatrix} \tag{2}$$

Likewise, the lattice IDWT structure is a mirror image of its DWT. Here, we propose a common operator for the lattice DWT and Convolutional code. Therefore, the mathematical representation of the DWT architecture is discussed for its Daubechies-4 lattice filter bank representation.

Daubechies Lattice DWT architecture:

From the lattice DWT structure in Fig.2, we define its computation method for Daubechies-4 lattice filter bank operations [10]. The equations (3) and (4) define the lattice DWT operation.

$$y_1 = [((x_o(n)z - \alpha_1 \cdot x_e(n))z^{-1} - \alpha_0(x_e(n) + x_o(n)z \cdot \alpha_1))z^{-1}] \cdot \beta \tag{3}$$

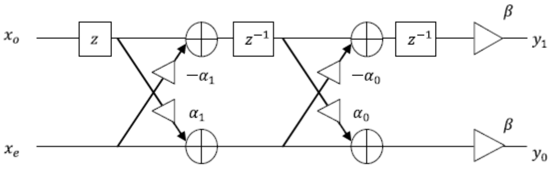


Fig.2. Daubechies-4 Lattice Filter.

$$y_0 = [((x_o(n)z \cdot \alpha_1 + x_e(n)) + \alpha_0((x_o(n)z - \alpha_1 \cdot x_e(n))z^{-1})] \cdot \beta \tag{4}$$

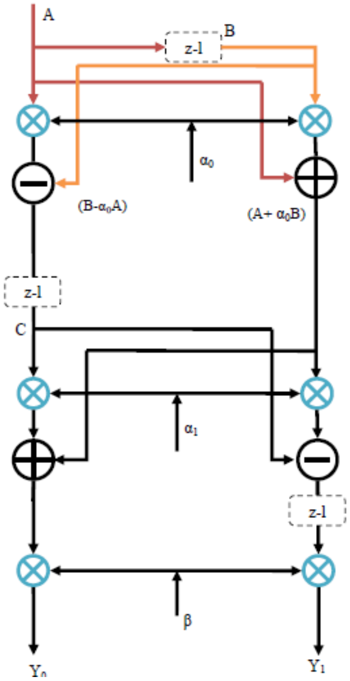


Fig.3. Realization Of Daubechies-4 Lattice DWT Architecture

Convolutional Code

A convolutional code does work by summing up some integrated excess information to the user's information and then compensating errors using this data, similar to any other error-correcting codes. A convolutional encoder is a linear system. A binary convolutional encoder can be staged as a array of shift registers. The outputs of the encoder are summed values present in the register's cells using modulo-2 summation. Either the unencoded sequence or the unencoded sequence added with the values of some register's cells are considered as the input to the encoder. Convolutional codes can be classified as systematic and non-systematic type. Therefore, an unencoded sequence is a part of the output sequence those are addressed as Systematic codes which is always recursive. While, non-recursive codes are always addressed as non-systematic. A recursive code or systematic code is the resultant polynomial of the combination of register's cells that forms one of the output streams.

Here, 'm' bit info symbol is encoded into 'n' bit coded symbols. $\frac{m}{n} = coderate(n \geq m)$ and K be the constraint length of code.

Convolutional codes are used widely in loads of applications, prepared to accomplish reliable information transfer, such as digital video, radio, mobile communication, and satellite communication.

'k' memory registers, each holding 1 input bit are considered to convolutionally encode data. All memory registers are having default value of 0, unless we assign it. The encoder has n modulo-2 adders (a modulo 2 adder can be implemented with a single Boolean XOR gate, where the logic is: 0+0 = 0, 0+1 = 1, 1+0 = 1, 1+1 = 0), and n generator polynomials — one for each adder (see figure below). An input bit m1 is fed into the left most register. Using the generator polynomials and the existing values in the remaining registers, the encoder outputs n bits. Now bit shift all register values to the right (m1 moves to m0, m0 moves to m-1) and wait for the next input bit. If there are no remaining input bits, the encoder continues output until all registers have returned to the zero state.

The figure below is a rate 1/2 (m/n) encoder with constraint length (k= 3). Generator polynomials are G1 = (1,1,1), and G2 = (1,0,1). Therefore, output bits are calculated (modulo 2) as follows:

$$n1 = m1 + m0 + m\text{-}1, \quad (5)$$
$$n2 = m1 + 0 + m\text{-}1. \quad (6)$$

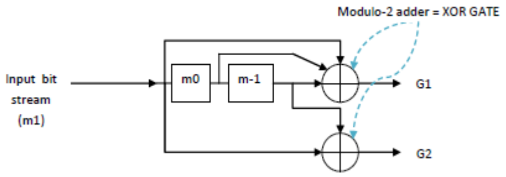


Fig.4. Convolutional Code Basic Structure

Convolutional encoder Architecture

We can derive the operations required by the encoder implementation, from equation (5) & (6). As illustrated in Fig.4, computation of every Generator polynomial involves at least an adder and a couple of delay elements. Equation (5) and (6) can be represented in generator polynomial form

n1 = m1 + m0 + m-1,
n2 = m1 + 0 + m-1.
As Generator Polynomials (G(D)) = [1+D+D^2, 1+D^2]

Therefore, the proposed architecture of Convolution encoder at the coding rate of ½=m/n.

The encoded output will be

$$O1= I*G(D1) \quad (7)$$
$$O2= I*G(D2) \quad (8)$$

Where, G(D1)= 1+D+D^2 and

G(D2)= 1+D^2.

As demonstrated in this section, the DWT and convolutional coder have solid resemblances if we examine their structures. This alikeness can be canvased to build a common structure for these two algorithms.

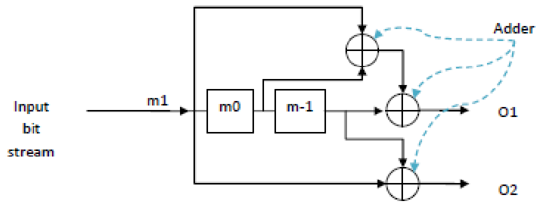


Fig.5. Convolutional Encoder

Proposed Common operator

In this segment of the paper, we are discussing a common operator system proposed for the two algorithms presented viz; DWT and Convolution encoder. This architecture performs the arithmetic operations of the lattice-DWT algorithm and the modulo-2 addition operations of the convolutional code algorithm. The common operator is designed in this paper by identifying some of the common arithmetic operations between the two different algorithms. Here, if you observe the architecture in fig(3) and in fig(4), they have a pair of outputs for a input source. They are defined as the lowpass and highpass outputs for DWT and a few output bits per a input bit in case of the convlutional code (code rate = 1/2) which is a result of generator polynomials with constraint length of k=3. From this similarity of the number of outputs of both DWT and encoder algorithms, we identify a common measure between them.

Next, the entire DWT architecture is divided/portioned into three different stages in order to realize the encoder structure with in, such that the common structure for both the algorithms is accommodated. And this actually is recognized from the structure of dwt and encoder where a combination of a pair of Daubechies(db)-2 lattice DWT and a pair of convolution code with rate=1/2 and constraint length k=2 is addressed. Therefore, the DWT archiotecture in 3 stages are recognized as common operator at first two stages and a DWT at the final stage. The stage1 and stage2 comprises of two db-2 lattice filters, while the encoder of rate=1/2 and length k=2. The figure below represents this exploration idea to accommodate the common operator architecture for DWT and Convolution encoder.

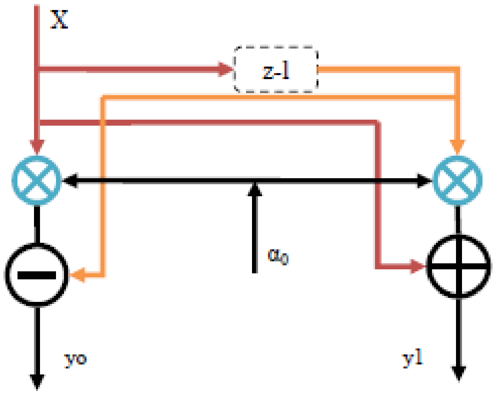


Fig.6. Daubechies-2 Filter.

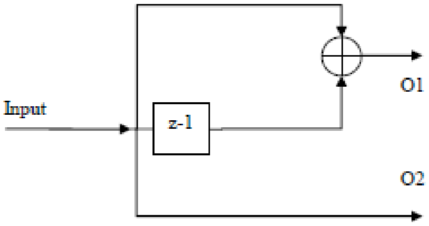


Fig. 7. Convolutional Code For Code Rate=1/2.

Whereas, stage3 is the part of DWt architecture and it performs scaling of the DWT output with scaling factor.

As presented in Fig.3, the Daubechiese-4 lattice DWT structure is composed by 6 multipliers and 4 Adder/Subtractor (A/S) whereas; the convolution coding structure is composed by 3 A/S unit with 2 delay elements or D-flip flops. In stage1, the encoder requires 2 A/S and a`delay element while for DWT it is 2 A/S, 2 multipliers and a`delay element. Whereas in stage2 also, the lattice-DWT comprises of 2 A/S, 2 multipliers and a delay element but in case of encoder 2 A/S and a delay element is required. As illustrated in the Fig.3 and Fig.4, the blocks are not interconnected in the same way for the two algorithms. Thus, to build a common structure for this stage we subdivide the architecture as in fig (6) & fig (7) which shows a common operator for the first two satges of the Convolution encoder and lattice DWT.

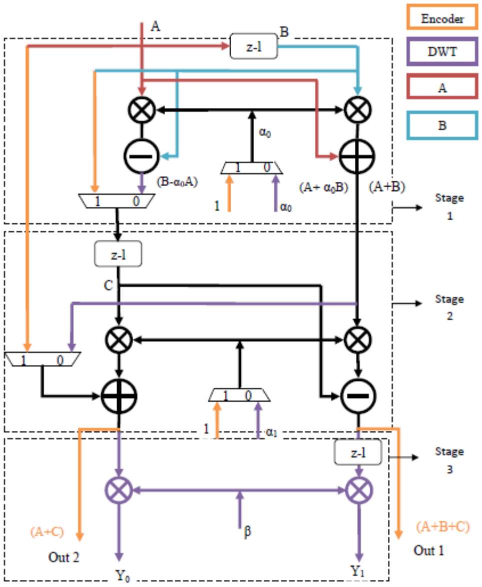


Fig.8. Proposed Common Operator For DWT And Convolutional Encoder

Finally, the common operator is designed with a reference signal to switch between the algorithms to be used and this reference signal will choose between DWT and convolutional encoder signals or Data. The DWT coefficient will be multiplied with the input signal at reference is’0’ then multiplexer selects the coefficient while when the reference is ‘1’ then multiplexer selects the encoder operation and inputs. The output of stage2 is considered as convoltional encoder output when reference is ‘1’. The output of stage3 is considered as DWT lowpass and highpass outputs when reference is ‘0’.

5. RESULTS AND COMPARISON

The experimental results of our proposed method are presented below. The proposed design is simulated and synthesized using Xilinx ISE 10.1. Result for the proposed Common Operator approach for DWT and Convolutional encoding algorithm is obtained. And all the experiments were performed on 3.00GHz Intel(R) Pentium(R) D, 1.00GB RAM, and 32-bit operating system with windows7 professional.

The simulated result for the convolutional encoder and lattice-DWT in the common operator is outlined below. I have encoded an input bit stream with the proposed CO. The proposed system delivers desired results when compared to the matlab output. While for the lattice-DWT the information signal format obtained from matlab is applied to the analysis side and the output obtained when compared with matlab gives RMSE of 0.6730.

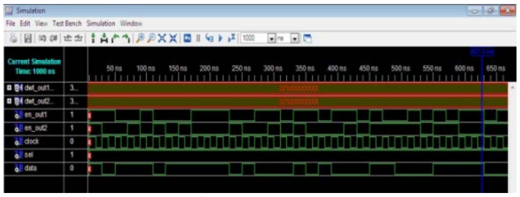

Simulation
File Edit View Test Bench Simulation Window
Current Simulation Time: 1000 ns
dwt_out1...
dwt_out2...
en_out1
en_out2
clock
sel
data

Fig.9.1 simulated result for the Convolution encoder with input “101110111111010001010011100011110” for the proposed Common Operator and its corresponding encoded output at rate= ½, “11 10 00 01 10 01 00 01 10 10 10 01 00 10 11 00 11 10 00 10 11 00 11 01 01 11 00 11 01 10 10 01”.

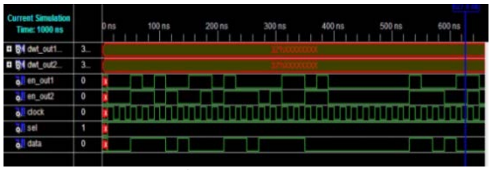


Fig.9.2 simulated result for the Convolution encoder with input “00111010001101111000000000110100” for the proposed Common Operator and its corresponding output at rate= ½, “00 00 11 01 10 01 00 10 11 00 11 01 01 00 01 10 10 01 11 00 00 00 00 00 00 00 11 01 01 00 10 11”.

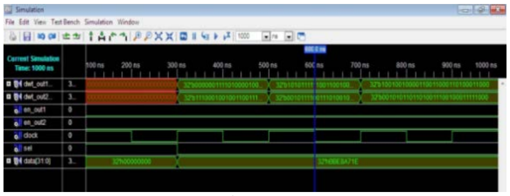

Simulation
File Edit View Test Bench Simulation Window
Current Simulation Time: 1000 ns
dwt_out1...
dwt_out2...
en_out1
en_out2
clock
sel
data[31:0]

Fig.9.3 simulated result for the lattice-DWT with input “101110111111010001010011100011110” for the proposed Common Operator and its corresponding output with RMSE “10010010000110011000110100011000” for the proposed Common Operator.

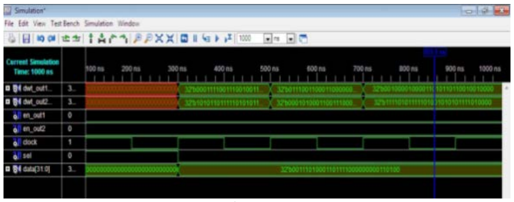

Simulator*
File Edit View Test Bench Simulation Window
Current Simulation Time: 1000 ns
dwt_out1...
dwt_out2...
en_out1
en_out2
clock
sel
data[31:0]

Fig.9.4 simulated result for the lattice-DWT with input “101110111111010001010011100011110” for the proposed Common Operator and its corresponding output with RMSE “00100001000011010110110010010000” for the proposed Common Operator.

DWT in CO is has several metrics that tend to be indicative of audio quality. The one that appear frequently when comparing original and decomposed or approximated data is root mean square error(RMSE).This measure will not be seriously reoriented by a single anomaly, as they are measuring average state. RMSE brings forth the same units as the original audio data, so its results are easy to represent. The results of our proposed DWT-CO architecture have RMSE is 0.6730.

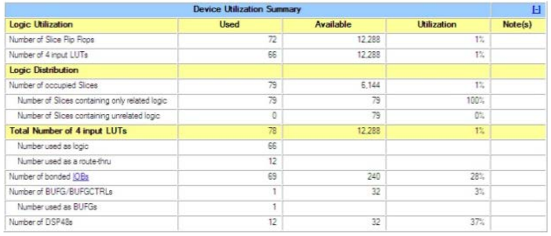
$$RMSE=\sqrt{\left[\frac{1}{N}\sum_{i=1}^{N}(in_i-out_i)^2\right]} \qquad (9)$$

Where; ‘N’ represents size of the image,

in_i Represents input image and

out_i Represents output image.

Table. 1
Device Utilization Table For Common Operator

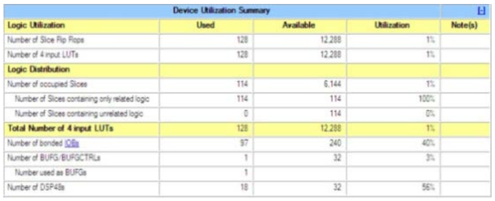


Device Utilization Summary				
Logic Utilization	**Used**	**Available**	**Utilization**	**Note(s)**
Number of Slice Flip Flops	72	12,288	1%	
Number of 4 input LUTs	66	12,288	1%	
Logic Distribution				
Number of occupied Slices	79	6,144	1%	
Number of Slices containing only related logic	79	79	100%	
Number of Slices containing unrelated logic	0	79	0%	
Total Number of 4 input LUTs	78	12,288	1%	
Number used as logic	66			
Number used as a route-thru	12			
Number of bonded IOBs	69	240	28%	
Number of BUFG/BUFGCTRLs	1	32	3%	
Number used as BUFGs	1			
Number of DSP48s	12	32	37%	

Fig.10. Proposed common operator device utilization table.

The device utilization table for the proposed common operator in fig.9 and otherwise for the DWT and Convolution encoder structure in fig.11 and fig.12 respectively gives an idea that, how well a common operator utilizes the available resources of one structure for the operation of the other structure in the same system.

Table..2
Device Utilization Table For Dwt Structure

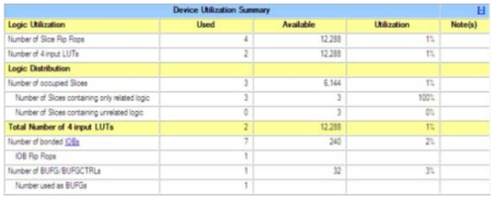


Device Utilization Summary

Logic Utilization	Used	Available	Utilization	Note(s)
Number of Slice Flip Flops	128	12,288	1%	
Number of 4 input LUTs	128	12,288	1%	
Logic Distribution				
Number of occupied Slices	114	6,144	1%	
Number of Slices containing only related logic	114	114	100%	
Number of Slices containing unrelated logic	0	114	0%	
Total Number of 4 input LUTs	128	12,288	1%	
Number of bonded IOBs	97	240	40%	
Number of BUFG/BUFGCTRLs	1	32	3%	
Number used as BUFGs	1			
Number of DSP48s	18	32	56%	

Fig.11. DWT structure logic utilization summary.

Table.3
Device Utilization For Convolution Encoder Decoder.



Device Utilization Summary

Logic Utilization	Used	Available	Utilization	Note(s)
Number of Slice Flip Flops	4	12,288	1%	
Number of 4 input LUTs	2	12,288	1%	
Logic Distribution				
Number of occupied Slices	3	6,144	1%	
Number of Slices containing only related logic	3	3	100%	
Number of Slices containing unrelated logic	0	3	0%	
Total Number of 4 input LUTs	2	12,288	1%	
Number of bonded IOBs	7	240	2%	
IOB Flip Flops	1			
Number of BUFG/BUFGCTRLs	1	32	3%	
Number used as BUFGs	1			

Fig.12. represents the logic utilization by the Convolution encoder decoder block.

Comparison of the Common Operator and the individual DWT and Convolution encoder structure:

Table. 4
Architecture Comparison Table.

Structure	Occupied slices	Power(W)
Proposed-CO	79	0.01210
DWT	114	0.01732
encoder	3	0.00555

Fig.13 presents the comparison for the common operator and the DWT and Convolution encoder structures.

As we can observe from the comparison table in fig.13 that, the sum of area required by the individual structures of the two algorithms DWT and the Convolution encoder is more as compared to a Common Operator. Thus, the proposed common operator has low power consumption and efficient area occupied.

6. CONCLUSION

A common operator for Lattice Discrete Wavelet Transform and Convolution encoder coding algorithm was proposed. The CO structure for both the algorithms was realized by the parameterization approach. The CO system was designed and synthesized using XILINX ISE tool, for the proposed system was analyzed for its logic utilization and power consumption with respect to the conventional DWT and encoder architectures. The presented architecture was synthesized and from the generated synthesis report the no. of occupied slices required by the CO was 32.48% reduced compared to the sum of no. of slices required by the two individual architectures Also, power consumption was analyzed, where the presented approach had reduced its required power by 47.09%.

REFERENCES

[1] Cooley, J. W. and Tukey, J. W, “An Algorithm for the Machine Calculation of Complex Fourier Series”, *Math. Computat*., 19, 297–301, 1965.

[2] J. Mitola, “The software radio architecture,” IEEE Communications Magazine, 33, pp. 26-38, 1995.

[3] R. I. Lackey and D. W. Upmal, “Speakeasy: The military software radio,” *IEEE Commun. Mag*., May 1995.

[4] D. Efstathiou, L. Fridman, Z. Zvonar, and Analog Devices Inc., “Recent developments in enabling technologies for software defined radio,” *IEEE Commun*. Mag., Aug. 1999.

[5] F. Jondral, “Software Defined Radio: Enabling technologies”, *Wiley* 2002.

[6] F. Jondral, “Parameter Controlled Software Defined Radio,” *Software Defined Radio Technical Conference, San Diego*, Nov 2002.

[7] L. Biard and D. Noguet, “An adaptable architecture for the Viterbi algorithm”, *the 7th International Symposium on Wireless Personal Multimedia Communications*, 2004.

[8] KemihKarim, KimoucheMosbah, BenslamaMalek and BenslamaAchour, “Performance of Discrete Wavelet Transform with Convolutional Code in Chaotic DS-CDMA System for Microsatellite Communication,” *Geoscience and Remote Sensing Symposium, 2004.IGARSS '04. Proceedings*, 2004 IEEE International, Volume: 1.

[9] J.Takala, K. Punkka, “Scalable FFT Processors and Pipelined Butterfly Units”, *The Journal of VLSI Signal Processing, Springer Netherlands* 2006

[10] C. Moy, J. Palicot, V. Rodriguez, D. Giri, “Optimal Determination of Common Operators for Multi-Standard Software-Defined Radio”, *4th Karlsruhe Workshop on Software Radios*, March 2006.

[11] JTRS, Software Commun. Architecture v2.2.2, *Washington, DC*, Apr.2006.

[12] S.T. Gul, C. Moy and J. Palicot, “Two scenarios of flexible Multi-standard architecture designs using a multi-granularity exploration”, *the 18th Annual IEEE International Symposium on Personal, Indoor and Mobile Radio Communications*, 2007.

[13] L. Alaus, J. Palicot, C. Roland, Y. Louet, D. Noguet, "Promising Technique of Parameterization For Reconfigurable Radio, the Common Operators Technique: Fundamentals and Examples" *Journal of Signal Processing Systems*, Springer, 2009.

[14] W. Tuttlebee, "Software Defined Radio – Baseband Technology for 3G Handsets and Basestations," *Communications Engineer* vol. 2, no. 2, pp. 46-47.

[15] L. Alaus, D. Noguet, J. Palicot, "A Common Operator Bank to resolve scheduling issue on a SDR Terminal". *The 6th Advanced international conference on telecommunications*, 2010.

[16] D. Noguet, G. Masera, V. Ramakrishnan, M.Belleville, D. Morche, G. Asheid, "Considering microelectronic trends in advanced wireless system design," *Advances in Electronics and Telecommunications Journal*, April 2010.

[17] D.Vijendra Babu and Dr.N.R.Alamelu, "Implementation of Energy Efficient Integer Wavelet Transform in Spartan 3E FPGA," *International Journal of Computer Applications*-2010, Vol-1, Issue-12.

[18] MalekNaoues, Laurent Alaus, Dominique Noguet, "A Common Operator for FFT and Viterbi algorithms," *Digital System Design: Architectures, Methods and Tools (DSD)*, 2010 13th Euro micro Conference on1-3 Sept. 2010, pp. 309 – 313.

[19] Malek Naoues, Dominique Noguet, Laurent Alaus and Yves Louët, "A Common Operator for FFT and FEC Decoding," *Microprocessors and Microsystems Journal (2011)*, 35, 8, 708-715.

[20] Akhilesh Bijalwan, Aditya Goyal and Nidhi Sethi, "Wavelet Transform Based Image Denoise Using Threshold Approaches," *International Journal of Engineering and Advanced Technology (IJEAT)*, Vol-1, Issue-5, June 2012.

[21] Malek Naoues, Dominique Noguet, Yves Louët, Khaled Grati and Ghazel, A, "An Efficient Flixible Common Operator for FFT and Viterbi Algorithms," *Vehicular Technology Conference (VTCSpring), IEEE 73rd*, May-2011, pp.1-5.

[22] Chunxiao Li, Raghunathan A, Jha, N.K, "An Architecture for Secure Software Defined Radio," *Design, Automation & Test in Europe Conference & Exhibition*, 2009, pp: 448-453.

[23] Richard C. Reinhart, Sandra K. Johnson, Thomas J. Kacpura, Charles S. Hall, Carl R. Smith, and John Liebetreu, "Open Architecture Standard for NASA's Software-Defined Space Telecommunications Radio Systems," *IEEE proceeding*, 2007, Vol. 95, Issue. 10, pp: 1986-1993.