IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received September 4, 2020, accepted September 24, 2020, date of publication October 6, 2020, date of current version October 19, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3029149

An Approximate Low-Power Lifting Scheme Using Reversible Logic

SITHARA RAVEENDRAN, (Member, IEEE), PRANOSE J. EDAVOOR, (Member, IEEE), NITHIN Y. B. KUMAR, (Senior Member, IEEE), AND M. H. VASANTHA, (Member, IEEE)
National Institute of Technology at Goa, Goa 403401, India

Corresponding author: Sithara Raveendran (sitharaveendrann@gmail.com)

ABSTRACT Haar Wavelet transform is an efficacious class of wavelet transform that satisfies both symmetry and orthogonality properties which are crucial in handling boundary distortion and energy preservation in image processing applications. Such applications demand power efficient design solutions that deliver high performance. Reversible logic has emerged as a solution that incorporates logical and physical reversibility to realise low power designs. This paper presents a reversible logic based design of Haar wavelet transform and lifting scheme for Haar wavelet transform, a first in literature of reversible logic. The designs are analysed to measure the efficiency of reversible logic implementations in terms of Quantum Cost (QC), Constant Inputs (CI), Garbage Outputs (GO) and Gate Count (GC). Furthermore, this paper proposes two architectures for Reversible Approximate Full Adder (RAFA) - RAFA-1 and RAFA-2; optimised explicitly for reversible logic based implementation. The proposed architectures have 25% Error Rate (ER) and optimised QC, CI, GC and GO when compared to existing exact and approximate full adder architectures implemented using reversible logic. Functional verification of the proposed architectures are performed on FPGA using 512 × 512 image. The efficiency of the image processing application is projected in terms of Structural Similarity Index Measure (SSIM) and Peak Signal to Noise Ratio (PSNR). Average SSIM and average PSNR are found to be 0.9679 and 31.81dB for RAFA-1 and 0.9696 and 32.15dB for RAFA-2 which are comparable with exact full adder based design.

INDEX TERMS Reversible logic, Haar wavelet transform, lifting scheme for Haar wavelet transform, approximate full adders, image processing.

I. INTRODUCTION

Wavelet transforms have emerged as one of the effective class of transforms and find application in fields like image processing, video processing, sound analysis, machine learning and Orthogonal Frequency Division Multiplexing (OFDM). Wavelet transforms can be classified as orthogonal and bi-orthogonal wavelets. Orthogonal wavelets preserve energy due to its orthogonality property. Bi-orthogonal wavelets have linear phase due to its symmetry which is essential in handling boundary distortions. Haar wavelet transform is one of the basic wavelets that has advantages of both orthogonal and bi-orthogonal wavelet transforms. It satisfies both symmetry and orthogonality properties. Even though the filter coefficients of Haar wavelet transform are integers, the memory requirement during computations is a bottleneck in their usage. Therefore, lifting scheme was introduced to achieve simple and structured VLSI architectures for wavelet based image processing applications [1].

One of the major design challenges in present day high performance VLSI architectures is the power dissipation at different levels of abstraction in the design. There is always a trade off between area, performance and power dissipation. Scaling down devices was the ground breaking solution to make electronic gadgets deliver high performance with compact hardware. As Moore's law nears its fundamental physical limits [2], other alternatives are to be explored to meet the low-power, high performance demands of the digital world. Moreover, Landauer's study showed that irrespective of the technology used, irreversible operations in conventional systems result in an energy loss of $KTln2$ J for every bit loss [3]. Here, K is the Boltzmann's constant and T is the temperature. This small power dissipation that was insignificant in predecessor device technologies has become

The associate editor coordinating the review of this manuscript and approving it for publication was Amit Singh.

significant as the devices are scaled. Bennett's study shows that, a design that is physically and logically reversible will result in architectures with zero power dissipation [4]. Therefore, reversible logic based circuits can be one of the alternatives to realise high speed, low-power image processing applications where Moore's law reaches its physical limits.

Furthermore, the accuracy of the image processing algorithms can be compromised to reduce the area and power and to increase the speed of operation within the tolerance range of PSNR. The image processing applications are able to utilise this trade off due to the disability of Human Visual System (HVS) to identify the finer details in images. To achieve this, approximate arithmetic circuits are introduced to reduce the computations by introducing errors in acceptable range. Conventional full adder circuit has been explored to incorporate approximation for error tolerant computation process in multimedia and deep learning based applications. Gupta *et al.* [5] have proposed three architectures for full adders with reduced logic complexity for Digital Signal Processing (DSP) systems. The three architectures - Approximation 1, 2 and 3 (GA1, GA2 and GA3 respectively) have ER of 37.5%, 37.5%, 50% respectively and have Error Distance (ED) of $(+1, -1, -1)$, $(-2, -1, +1)$, and $(+1, -1, -1, +1)$ respectively. Yang *et al.* [6] have proposed two architectures for approximate full adders (TGA1 and TGA2) that are optimised for implementation using transmission gates. TGA1 and TGA2 have ER of 25 % and ED of $(+1, -1)$ and $(+1, +1)$ respectively. Liu *et al.* [7] have proposed two architectures (MLAFA1 and MLAFA2) for majority logic based approximate adders. Both the adders have an ER of 25% with ED of $(+1, -1)$. Anusha and Deepa [8] have presented an approximate framework for multiplication operations in image processing by proposing seven approximate full adder designs - AA6, AA7, AA8, AA9, AA10, AA11, AA12 with ER of 37.5%, 25%, 37.5%, 12.5%, 37.5%, 25%, 12.5% and ED of $(+1, -1, +1)$, $(+1, -1)$, $(-1, +1, -1)$, $(+1)$, $(+1, +1, -1)$, $(+1, -1)$ and $(+1)$ respectively.

In order to achieve a power efficient architecture for error resilient image processing applications, this paper proposes a reversible logic based architecture which incorporates approximation while ensuring the accuracy drop to be in permissible range. The contributions of the paper are as listed below.

- First of its kind implementation of Haar wavelet transform and lifting scheme for Haar wavelet transform using reversible logic.
- To optimise the architectures for image processing applications, two reversible logic compatible Approximate Full Adders (AFA) are proposed.
- Unlike the conventional AFA which are optimised for CMOS implementation, this paper optimises the approximate full adder design for reversible logic, based on the inherent XOR operation in basic reversible logic gates.
- Reversible logic based Haar wavelet transform and lifting scheme for Haar wavelet transform is tested using standard 'Lena' image [9] on Kintex 7 FPGA platform. The efficiency of the architectures are projected in terms of SSIM and PSNR.

Section II presents a brief description of Haar wavelet transform and lifting scheme for Haar wavelet transform. An introduction for reversible logic and the performance parameters for measuring the efficiency of reversible logic based designs are presented in Section III. The computational units along with its implementation using reversible logic is presented in Section IV. Section V presents the proposed approximate full adder architectures using reversible logic. The proposed reversible logic based architectures for Haar wavelet filter and lifting scheme for Haar wavelet filter is presented in Section VI. Experimental results are presented in Section VII followed by conclusion in Section VIII.

II. HAAR WAVELET TRANSFORM AND LIFTING SCHEME FOR HAAR WAVELET TRANSFORM

The Haar (mother wavelet) function $\psi(x)$ [1] can be represented as

$$\psi(x) = \begin{cases} 1 & \text{if } 0 \le x \le \frac{1}{2} \\ -1 & \text{if } \frac{1}{2} \le x \le 1 \\ 0 & otherwise \end{cases} \tag{1}$$

The child wavelets are obtained by dilating the mother wavelet by $2^{i/2}$ to meet perfect reconstruction conditions.

$$\psi_{i,j}(x) = \psi(2^i x - j)2^{\frac{i}{2}} \quad i, j \in \mathbb{Z} \tag{2}$$

Let $g(x)$ represent a discrete sequence acquired by sampling a continuous signal. If A and B are two adjacent samples in the sequence, they show high degree of correlation. Upon applying Haar wavelet transform (analysis High Pass Filter (HPF) and analysis Low Pass Filter (LPF)), it results in difference (Dif) and average (Avg) components respectively and is given in equation (3).

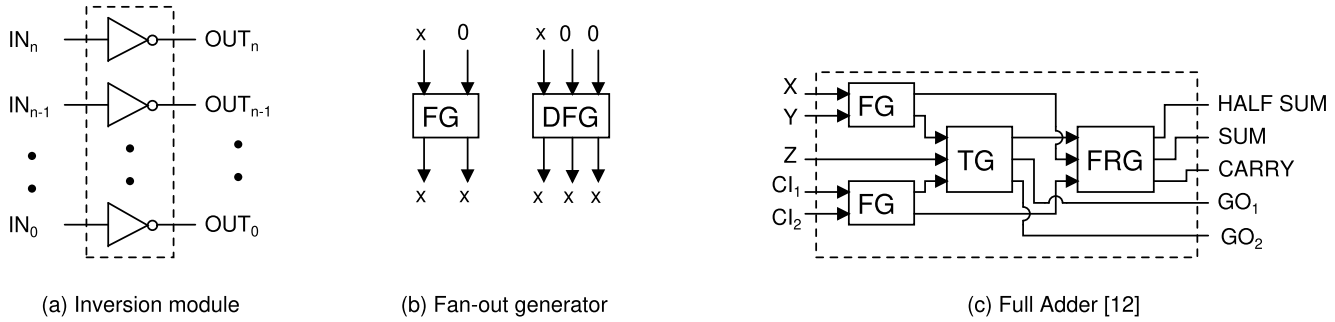
$$Dif = \text{B-A} \quad Avg = \frac{A+B}{2} \tag{3}$$

Inverse Haar (synthesis LPF and synthesis HPF) is applied for reconstruction and can be represented by equation (4).

$$A = Avg - \frac{Dif}{2} \quad B = Avg + \frac{Dif}{2} \tag{4}$$

This approach incurs huge memory and computations in multimedia applications. In order to address this issue, lifting scheme was introduced, where Haar wavelet transform can be computed faster. Lifting scheme achieves this by reducing the computations and by using in-place memory to reduce the memory usage. In this paper, the reduction in computations and thereby an improvement in speed and area is addressed in the implementation. Forward lifting (using analysis HPF and analysis LPF) is applied by utilising the memory where the input samples are stored and the computations are given by equation (5).

$$Dif = \text{B-A} \quad Avg = A + \frac{Dif}{2} \tag{5}$$



(a) Inversion module (b) Fan-out generator (c) Full Adder [12]

FIGURE 1. Basic reversible computation units used in realising Haar wavelet and Lifting scheme for Haar wavelet.

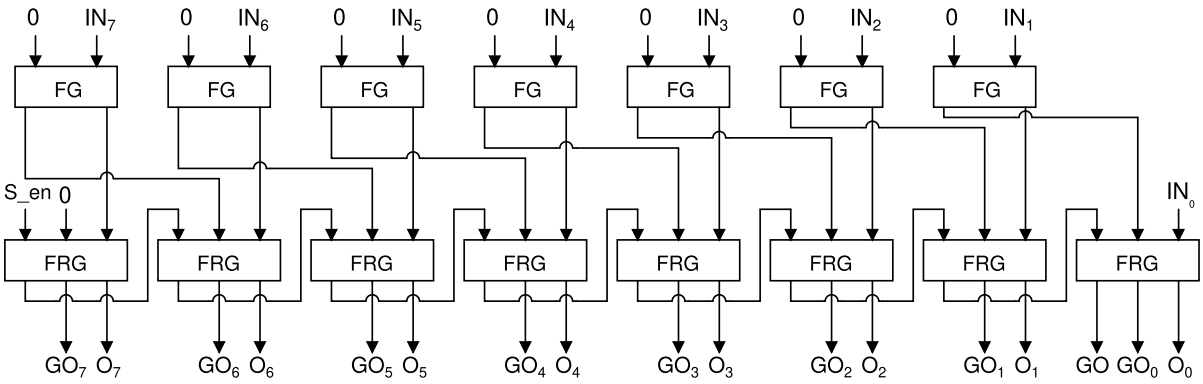


FIGURE 2. Reversible division by 2 [13].

The *Dif* generated is stored in the location where B is stored and *Dif* is used to generate *Avg* unlike the individual data path for generation of *Dif* and *Avg* in Haar transform without lifting. Similarly, the inverse lifting scheme retrieves the samples and is shown in equation (6).

$$A = Avg - \frac{Dif}{2} \quad B = Dif + A \tag{6}$$

III. REVERSIBLE LOGIC

A circuit is said to be reversible, if there is a unique input to output mapping. An $n \times n$ reversible circuit has n inputs, n outputs and 2^n unique input to output combinations. The basic reversible gates that are used in this paper are Fredkin Gate (FRG), Feynman Gate (FG), Double Feynman Gate (DFG), Toffoli Gate (TG), NOT gate (NOT), BJN gate (BJN) [10] and M-Peres Gate (MPG) [11].

There are different implementation parameters (QC, GO, CI, GC) to measure the efficiency of a reversible circuit. QC gives a count of primitive gates used in a design. This gives a measure of complexity of the circuit. GO count gives the number of outputs that are not a part of relevant or primary outputs. CI count gives the number of inputs that are not a part of relevant or primary inputs. GO and CI are added for ensuring logical reversibility. GC gives the number of reversible logic gates used in the circuit.

IV. REVERSIBLE LOGIC BASED COMPUTATIONAL UNITS USED IN THE PROPOSED DESIGNS

A. INVERSION MODULE

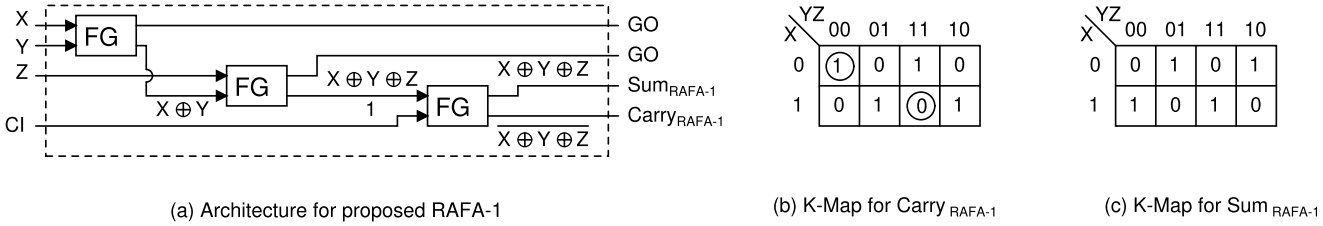
The inversion module is used to invert the subtrahend in 2's complement subtraction operations. This module consists of an array of conventional NOT gates and GC depends on the input word length. The inversion unit for input width 'n' is shown in Figure 1 (a).

B. FAN-OUT GENERATOR

In contrast to the conventional irreversible circuits, reversible logic does not support fan-out to maintain physical and logical reversibility. Fan-out or duplication of a signal is attained by using Feynman and double Feynman gates. Figure 1 (b) shows fan out generator implementation using FG and DFG.

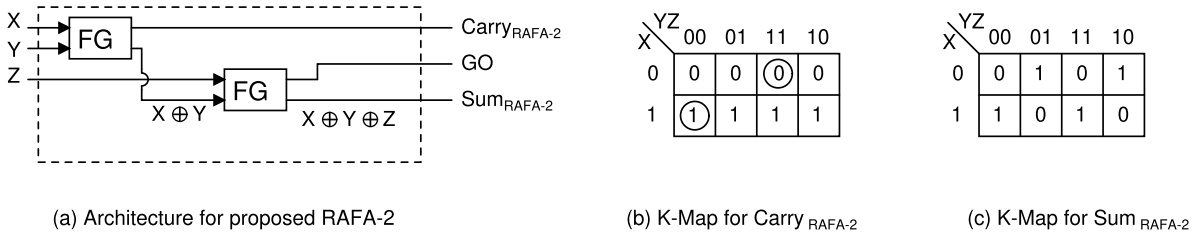
C. FULL ADDER

A reversible full adder [12] is shown in Figure 1 (c). The full adder architecture has three inputs and two constant inputs. In the five outputs, apart from *SUM* and *CARRY*, a half adder sum (*HALF SUM*) and two garbage outputs are generated. The full adder is included with Over-Flow Correction (OFC) when 2's complement subtraction operation is carried out.



(a) Architecture for proposed RAFA-1 (b) K-Map for Carry RAFA-1 (c) K-Map for Sum RAFA-1

FIGURE 3. Proposed Reversible logic based approximate full adder - RAFA-1.



(a) Architecture for proposed RAFA-2 (b) K-Map for Carry RAFA-2 (c) K-Map for Sum RAFA-2

FIGURE 4. Proposed Reversible logic based approximate full adder - RAFA-2.

D. DIVISION BY 2

Division by 2 is required for computing average for forward Haar wavelet transform. Shifter based architecture proposed by Kotiyal *et al.* [13] is employed to realise division by 2. A division by 2 module for 8-bit input is shown in Figure 2. It consists of an array of 7 Feynman gates and an array of 8 Fredkin gates with a select line S_{en}. When S_{en} is high, the output will be half the input and input is passed as it is when S_{en} is low.

V. PROPOSED APPROXIMATE FULL ADDERS FOR REVERSIBLE LOGIC BASED IMPLEMENTATIONS

Image processing applications exploit the spatial redundancies that cannot be detected by human eye to optimise the hardware and memory requirements. Addition operation is one of the commonly used operations in linear and non-linear transformations. To optimise the full adder architecture in scales of QC, GO, CI and GC, this paper proposes two approximate adders using reversible logic. The fundamental reversible gates like TG, FG, DFG and FRG have an inherent *XOR* logic operation at the output. Other gates with logic operations like *AND*, *OR*, *NAND* and *NOR* incur huge overhead in terms of QC [10]. Therefore, this paper proposes two approximate full adders, which optimises the output function in terms of *XOR* operations to minimise QC and GC.

The expression for *Sum* and *Carry* of exact full adder is given as

$$Sum = X \oplus Y \oplus Z \tag{7}$$

$$Carry = (X \oplus Y) \cdot Z + XY \tag{8}$$

A. PROPOSED APPROXIMATE FULL ADDER - RAFA-1

The expression for *Sum* and *Carry* for proposed RAFA-1 shown in Figure 3 (a) is given as

$$Sum_{RAFA-1} = X \oplus Y \oplus Z \tag{9}$$

$$Carry_{RAFA-1} = \overline{X \oplus Y \oplus Z} = \overline{Sum_{RAFA-1}} \tag{10}$$

The K-Map for $Carry_{RAFA-1}$ and Sum_{RAFA-1} are shown in Figure 3 (b) and Figure 3 (c) respectively. The errors introduced are represented using ① and ⓪. ① represents the introduction of error where a zero in the output function is replaced by one. Similarly, ⓪ represents a change of one to zero when the error is introduced. From Figure 3 (b) it can be seen that, error is introduced for input combinations '000' and '111'. Errors are not introduced for Sum_{RAFA-1} and hence the K-Map remains the same as an exact full adder.

B. PROPOSED APPROXIMATE FULL ADDER - RAFA-2

The expression for *Sum* and *Carry* for proposed RAFA-2 shown in Figure 4 (a) is given as

$$Sum_{RAFA-2} = X \oplus Y \oplus Z \tag{11}$$

$$Carry_{RAFA-2} = X \tag{12}$$

The K-Map for $Carry_{RAFA-2}$ and Sum_{RAFA-2} are shown in Figure 4 (b) and Figure 4 (c) respectively. From Figure 4 (b) it can be seen that, error is introduced for input combinations '011' and '100'. Errors are not introduced for Sum_{RAFA-2} and hence the K-Map remains the same as an exact full adder. The primary motive in retaining the expression for *Sum* as that of exact full adder is to take advantage of using inherent XOR operation in reversible gates and realising the adder using minimum QC and GC.

The truth table for the proposed architectures for approximate full adder is shown in Table 1. Approximate circuits are measured in terms of ER and ED. ER, represented in percentage gives the number of errors introduced in the output. ED represents the maximum deviation of the errors introduced from the actual output. The proposed designs introduce equal deviation in positive and negative directions, which aids in reduction of error when the adders are cascaded in forward and inverse transforms. The proposed designs have

TABLE 1. Truth table for exact full adder (EFA) and proposed reversible logic based approximate full adder (RAFA-1 and RAFA-2).

Inputs			Exact Full Adder		Approximate Full Adder (RAFA-1)		ED	Approximate Full Adder (RAFA-2)		ED
X	Y	Z	$Carry$	Sum	$Carry_{RAFA-1}$	Sum_{RAFA-1}		$Carry_{RAFA-2}$	Sum_{RAFA-2}	
0	0	0	0	0	1	0	**+2**	0	0	0
0	0	1	0	1	0	1	0	0	1	0
0	1	0	0	1	0	1	0	0	1	0
0	1	1	1	0	1	0	0	0	0	**-2**
1	0	0	0	1	0	1	0	1	1	**+2**
1	0	1	1	0	1	0	0	1	0	0
1	1	0	1	0	1	0	0	1	0	0
1	1	1	1	1	0	1	**-2**	1	1	0

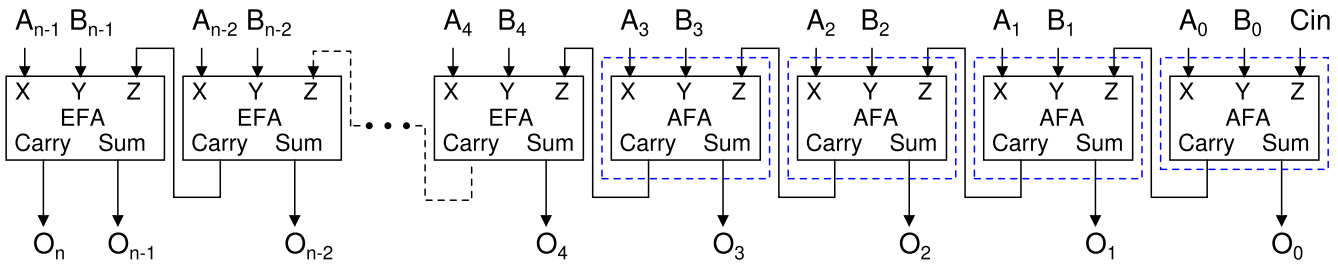


FIGURE 5. n-bit adder with approximation.

an ER of 25% and ED ±2. The proposed architectures are able to reduce the stages required for generation of *Sum* and *Carry*. RAFA-1 generate *Sum* and *Carry* in three stages. And, RAFA-2 generates *Sum* in two stages and *Carry* in one stage.

In EFA based n-bit adders, all the full adders used are exact full adders presented in [12], which is the best in literature for EFA. When approximation is introduced in n-bit adders, the four least significant adders are replaced with the approximate adders and the other adders are exact full adders and is shown in Figure 5. The n-bit adder has two n-bit inputs A and B and (n + 1)-bit output O. When approximation is introduced in an n-bit adder, the adders used to generate O_3, O_2, O_1 and O_0 are approximate full adders and the rest are exact full adders. The exact full adders are marked by EFA and the approximate full adders are marked by AFA and blue dashed boxes. This general structure is considered for all the adders used in the design of architectures for Haar wavelet filter and lifting scheme for Haar wavelet filter.

VI. PROPOSED REVERSIBLE LOGIC BASED ARCHITECTURE FOR HAAR WAVELET FILTER AND LIFTING SCHEME FOR HAAR WAVELET FILTER

The different computational and logical operation modules used to design the proposed architectures are abbreviated and the notations are listed in Table 2.

A. FORWARD HAAR WAVELET FILTER

The proposed architecture for forward Haar wavelet filter is shown in Figure 6. To generate *Avg* and *Dif*, the two samples are scaled by 2 using two reversible shifter based divider (division by 2) modules. The scaled outputs are then passed onto fan-out generators to make two copies. To generate *Avg*, the output from each fan-out generator is added using an 8-bit reversible adder. For *dif*, $\frac{B}{2}$ is subtracted from $\frac{A}{2}$ by two's complement subtraction. A copy of $\frac{B}{2}$ at the output of fan-out generator is inverted and C_{IN} to the second 8-bit adder is set to one. An overflow correction for two's complement data representation is also included in the second adder, where subtraction operation is performed. The computational units that employs the proposed approximate full adder architectures are enclosed in blue dashed boxes.

TABLE 2. Notations used in proposed architectures using reversible gates.

Design	Notation Used
Fan out generator (2 copies)	FOG-2
Reversible divider (Division by 2)	RD-2
Inversion Module	INV
Reversible 8-bit adder	R8BA
Reversible 9-bit adder	R9BA
Reversible 8-bit adder with overflow correction	R8BA OFC
Reversible 9-bit adder with overflow correction	R9BA OFC

In Figure 6, R8BA and R8BA OFC employs RAFA-1/RAFA-2. In all the architectures, approximate full adders replace the exact full adders in the adders for four least significant bits.

B. INVERSE HAAR WAVELET FILTER

The proposed architecture for inverse Haar wavelet filter is shown in Figure 7. To generate the samples from the *Dif* and

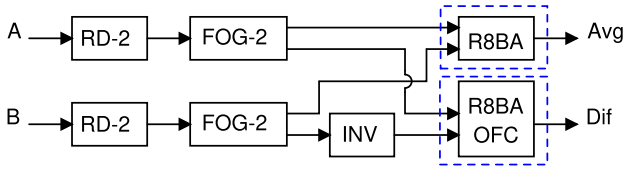


FIGURE 6. Proposed architecture for one level forward Haar wavelet filter using reversible logic.

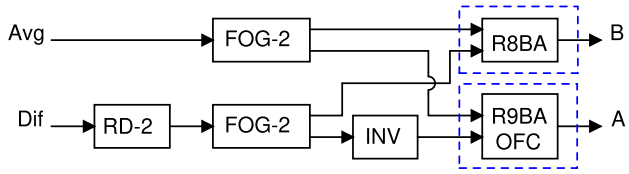


FIGURE 7. Proposed architecture for one level Inverse Haar Wavelet filter using reversible logic.

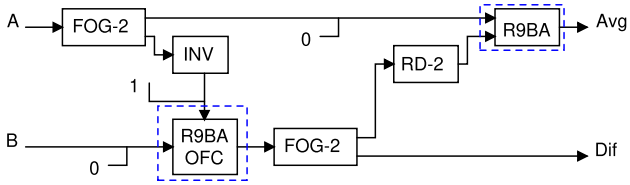


FIGURE 8. Proposed architecture for one level Lifting scheme for forward Haar Wavelet filter using reversible logic.

Avg, the first step is to generate two copies of the $\frac{Dif}{2}$ and *Avg* using fan-out generator.

Further, addition and subtraction of the two are performed using an 8-bit reversible adder and a 9-bit reversible adder respectively. The computational units that employs the proposed approximate full adder architectures are R8BA and R9BA OFC. The blue dashed boxes in Figure 7 represents the units with approximate full adders.

C. FORWARD HAAR WAVELET FILTER USING LIFTING SCHEME

The proposed architecture for forward Haar wavelet filter using lifting scheme is shown in Figure 8. To generate *Avg* and *Dif*, sample A is duplicated using a fan-out generator. Further, $B - A$ is generated using a 9-bit adder using two's complement subtraction operation. The difference is passed onto a fan-out generator, out of which one copy is passed to the output of the filter as *Dif* and the second copy is scaled down by 2 using reversible divider. The scaled output is added with sample A to generate *Avg*. The computational units that employ the proposed approximate full adder architectures (RAFA-1 and RAFA-2) are R9BA and R9BA OFC. The units that employa approximate adders are enclosed in blue dashed boxes in Figure 8.

D. INVERSE HAAR WAVELET FILTER USING LIFTING SCHEME

The proposed architecture for inverse Haar wavelet filter is shown in Figure 9. Firstly, two copies of *Dif* is generated

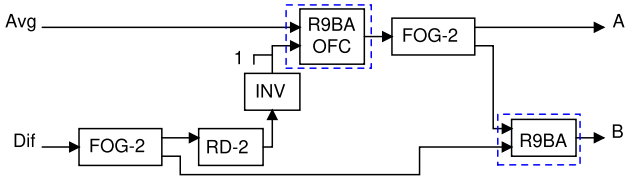


FIGURE 9. Proposed architecture for one level Lifting scheme for Inverse Haar wavelet filter using reversible logic.

using a fan-out generator. One of the copies of *Dif* is scaled down by 2 with a reversible division by 2 module. The scaled output is subtracted from *Avg* to generate sample A. A fan-out generator is used to generate two copies of the sample *A*; one copy is directed to the output and the other copy is added with *Dif* to obtain sample *B*. The computational units that employ the proposed approximate full adder architectures (RAFA-1 and RAFA-2) are R9BA and R9BA OFC. The units where approximate adders are used are enclosed in blue dashed boxes in Figure 9.

VII. EXPERIMENTAL RESULTS

A. REVERSIBLE LOGIC IMPLEMENTATION PARAMETERS FOR PROPOSED REVERSIBLE APPROXIMATE FULL ADDERS

A comparison of the proposed AFA architectures with the existing full adder architectures is presented in Table 3. Among the exact full adders, gate count is the least in case of Awais *et al.* [11] and highest in the case of Parmahi *et al.* [14]. However, QC is the least in case of Aditya *et al.* [12] and highest QC is for Parmahi *et al.* [14]. Highest CI and GO are for Bruce *et al.* [15] and the lowest is for Aditya *et al.* [12]. The least number of stage to generate *Sum* and *Carry* is for the EFA proposed by Aditya *et al.* [12]. Among the approximate full adders, highest ER of 37.5% is for GA1, GA2, AA6, AA8 and AA10. The lowest ER of 12.5% is for AA9 and AA12. The reversible logic implementation parameters CI, GO, GC and QC are the highest in case of MLAFA1 and AA11. From Table 3, it is evident that the approximate full adders proposed in this paper have the least CI, GO, GC and QC. The number of stages to generate *Sum* and *Carry* are the highest in case of AA11 and MLAFA1 respectively. The *Sum* is generated in two stages and *Carry* is generated in one stage in RAFA-2, a minimum among existing approximate full adder architectures.

B. REVERSIBLE LOGIC IMPLEMENTATION PARAMETERS AND IMAGE PROCESSING METRICS FOR PROPOSED REVERSIBLE LOGIC BASED HAAR AND LIFTING SCHEME FOR HAAR WAVELET FILTERS

The proposed designs are analysed to measure the efficacy in scales of reversible logic implementation parameters (QC, CI, GC, GO) and image processing metrics (SSIM [16] and PSNR [16]). In EFA based designs, all the full adders used in 8-bit, 9-bit and 10-bit adders are exact full adders presented in [12]. In approximate full adder based designs, only the four least significant adders are replaced with the

TABLE 3. Comparison of proposed reversible compatible approximate full adder (RAFA-1 and RAFA-2) with existing reversible logic based full adder designs.

Design	ED	ER	Gates Used	Implementation Parameters				No. of Stages	
				CI	GO	GC	QC	Sum	Carry
Exact Full Adder Architectures									
Bruce *et al.* [15]	-	-	FRG(5)	4	5	5	25	4	3
Parmahi *et al.* [14]	-	-	FG (1), FRG (5)	3	4	6	26	3	3
Awais *et al.* [11]	-	-	TG (1), MPG (2)	3	4	3	13	2	3
Aditya *et al.* [12]	-	-	FG (2), TG (1), FRG (1)	2	2	4	12	3	2
Approximate Full Adder Architectures									
Gupta *et al.* [5] Approximation 1 (GA1)	+1, -1, -1	37.5%	NOT (1), TG (1), FG (1), BJN (1)	3	4	4	12	4	3
Gupta *et al.* [5] Approximation 2 (GA2)	-2, -1,+1	37.5%	NOT (1), TG (2), FG (1), BJN (1)	4	5	5	17	5	1
Yang *et al.* [6] (TGA1)	+1, -1	25%	NOT (2), FG (3), TG (2), BJN (1)	5	6	8	20	5	1
Yang *et al.* [6] (TGA2)	+1, +1	25%	NOT (1), TG (1), FG (1), BJN (1)	2	3	4	12	4	1
Liu *et al.* [7] (MLAFA1)	+1, -1	25%	NOT (3), TG (6), BJN (4)	10	11	13	53	5	9
Liu *et al.* [7] (MLAFA2)	+1, -1	25%	NOT (1), TG (3), FG (1), BJN (2)	6	7	7	27	5	2
Anusha and Deepa [8] (AA6)	+1, -1, +1	37.5%	NOT (1), FG (1), TG (1), BJN (1)	3	4	4	12	3	1
Anusha and Deepa [8] (AA7)	+1, -1	25%	NOT (1), TG (4), BJN (4)	8	9	9	41	6	8
Anusha and Deepa [8] (AA8)	-1, +1, -1	37.5%	NOT (1), TG (4), BJN (3)	7	8	8	36	5	4
Anusha and Deepa [8] (AA9)	+1	12.5%	NOT (3), TG (4), FG (1), BJN (3)	7	8	11	39	6	5
Anusha and Deepa [8] (AA10)	+1, +1, -1	37.5%	NOT (1), TG (3), BJN (3)	6	7	7	31	4	7
Anusha and Deepa [8] (AA11)	+1, -1	25%	NOT (3), TG (6), BJN (4)	10	11	13	53	9	4
Anusha and Deepa [8] (AA12)	+1	12.5%	NOT (1), TG (4), BJN (3)	7	8	8	36	4	5
Proposed RAFA-1	+2, -2	25%	FG (3)	1	2	3	3	3	3
Proposed RAFA-2	+2, -2	25%	FG (2)	0	1	2	2	2	1

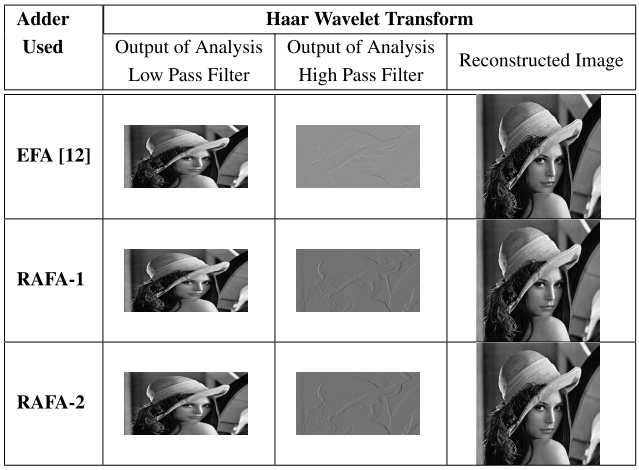


FIGURE 10. Transformed images using proposed reversible architectures for Haar wavelet Transform using EFA, RAFA-1 and RAFA-2.

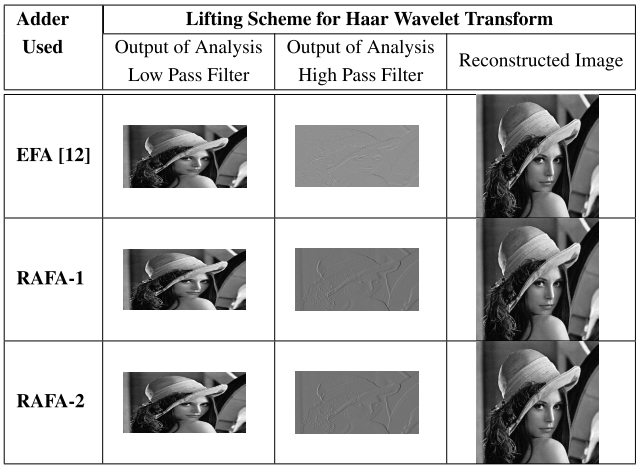


FIGURE 11. Transformed images using proposed reversible architectures for Lifting scheme for Haar wavelet transform using EFA, RAFA-1 and RAFA-2.

approximate adders and the other adders are exact full adders. The reversible logic implementation parameters for proposed designs of Haar wavelet transform and lifting scheme for Haar wavelet transform are presented in Table 4 and Table 5. In Table 4 and Table 5, for both forward and inverse transforms, maximum GC, QC, CI and GO is for MLAFA1 and AA11. The least is for proposed RAFA-1 and RAFA-2 based designs.

It can be observed from Table 4 and Table 5 that, with the introduction of approximation (compatible with reversible logic implementation) in the proposed designs, the reversible logic implementation parameters have been reduced. The reduction in the performance parameters is evident in the case of architectures that utilise the proposed approximate full adders (RAFA-1 and RAFA-2) as they are optimised for reversible logic based implementations.

TABLE 4. Reversible logic parameters for Haar Wavelet Transform using Exact Full Adder (EFA) and Approximate Full Adders (AFA).

Design	Full Adder Architecture Used	ED	GC	QC	CI	GO
1-level Haar Transform (Forward)	**EFA**					
	Aditya *et al.* [12]	-	119	311	68	66
	AFA					
	Gupta *et al.* [5] Approximation 1 (GA1)	+1, -1, -1	119	311	74	74
	Gupta *et al.* [5] Approximation 2 (GA2)	-2, -1, +1	127	351	82	82
	Yang *et al.* [6] (TGA1)	+1, -1	151	375	90	90
	Yang *et al.* [6] (TGA2)	+1, +1	119	311	66	66
	Liu *et al.* [7] (MLAFA1)	+1,-1	191	639	130	130
	Liu *et al.* [7] (MLAFA2)	+1,-1	143	431	98	98
	Anusha and Deepa [8] (AA6)	+1,-1,+1	119	311	74	74
	Anusha and Deepa [8] (AA7)	+1,-1	159	543	114	114
	Anusha and Deepa [8] (AA8)	-1, +1, -1	151	503	106	106
	Anusha and Deepa [8] (AA9)	+1	167	527	106	106
	Anusha and Deepa [8] (AA10)	+1, +1, -1	143	463	106	90
	Anusha and Deepa [8] (AA11)	+1, -1	191	639	130	130
	Anusha and Deepa [8] (AA12)	+1	151	503	106	106
	Proposed RAFA-1	+2, -2	111	239	60	58
	Proposed RAFA-2	+2, -2	103	231	50	50
1-level Haar Transform (Inverse)	**EFA**					
	Aditya *et al.* [12]	-	116	307	66	64
	AFA					
	Gupta *et al.* [5] Approximation 1 (GA1)	+1, -1,-1	116	307	72	72
	Gupta *et al.* [5] Approximation 2 (GA2)	-2, -1, +1	124	347	80	80
	Yang *et al.* [6] (TGA1)	+1, -1	148	371	88	88
	Yang *et al.* [6] (TGA2)	+1, +1	116	307	64	64
	Liu *et al.* [7] (MLAFA1)	+1, -1	188	635	128	128
	Liu *et al.* [7] (MLAFA2)	+1, -1	140	427	96	96
	Anusha and Deepa [8] (AA6)	+1, -1, +1	116	307	72	72
	Anusha and Deepa [8] (AA7)	+1, -1	156	539	112	112
	Anusha and Deepa [8] (AA8)	-1, +1, -1	148	499	104	104
	Anusha and Deepa [8] (AA9)	+1	164	523	104	104
	Anusha and Deepa [8] (AA10)	+1, +1, -1	140	459	104	104
	Anusha and Deepa [8] (AA11)	+1, -1	188	635	128	128
	Anusha and Deepa [8] (AA12)	+1	148	499	104	104
	Proposed RAFA-1	+2, -2	108	235	58	56
	Proposed RAFA-2	+2, -2	100	227	48	48

For functional verification of the proposed circuits, the designs are implemented using Verilog and tested on Kintex -7 FPGA using 512 × 512 standard Lena image [9]. The images obtained by using proposed reversible logic based 1-level forward and inverse Haar wavelet transform, and lifting scheme are shown in Figure 10 and Figure 11 respectively. The quality of output images are measured in terms of SSIM and PSNR.

On of the commonly used metric to assess the quality of an image processing algorithm or architecture is PSNR. To compute PSNR, the quality of the output image from design that employs approximate computational units is compared with the output image of the same design with exact computational units. For this, Mean Squared Error (MSE) is calculated first and then the PSNR is estimated. The equations for MSE and PSNR are given in equations (13) and (14) respectively.

$$MSE = \frac{1}{a \times b} \sum_{j=0}^{a-1} \sum_{k=0}^{b-1} \Big[Ex(j,k) - App(j,k) \Big]^2 \tag{13}$$

$$PSNR = 10log_{10}\left(\frac{\big[Max_{pixel}\big]^2}{MSE} \right) \tag{14}$$

TABLE 5. Reversible logic parameters for Lifting Scheme for Haar Wavelet Transform using Exact Full Adder (EFA) and Approximate Full Adders (AFA).

Design	Full Adder Architecture Used	ED	GC	QC	CI	GO
1-level Lifting scheme for Haar Transform (Forward)	**EFA**					
	Aditya *et al.* [12]	-	114	305	65	64
	AFA					
	Gupta *et al.* [5] Approximation 1 (GA1)	+1, -1, -1	114	305	95	96
	Gupta *et al.* [5] Approximation 2 (GA2)	-2, -1, +1	122	345	79	80
	Yang *et al.* [6] (TGA1)	+1, -1	146	369	87	88
	Yang *et al.* [6] (TGA2)	+1, +1	114	305	63	64
	Liu *et al.* [7] (MLAFA1)	+1, -1	186	633	127	128
	Liu *et al.* [7] (MLAFA2)	+1, -1	138	425	95	96
	Anusha and Deepa [8] (AA6)	+1, -1, +1	114	305	71	72
	Anusha and Deepa [8] (AA7)	+1, -1	154	537	111	112
	Anusha and Deepa [8] (AA8)	-1, +1, -1	146	497	103	104
	Anusha and Deepa [8] (AA9)	+1	162	521	103	104
	Anusha and Deepa [8] (AA10)	+1, +1, -1	138	457	103	88
	Anusha and Deepa [8] (AA11)	+1, -1	186	633	127	128
	Anusha and Deepa [8] (AA12)	+1	146	497	103	104
	Proposed RAFA-1	+2, -2	106	233	57	56
	Proposed RAFA-2	+2, -2	98	225	47	48
1-level Lifting scheme for Haar Transform (Inverse)	**EFA**					
	Aditya *et al.* [12]	-	120	319	67	67
	AFA					
	Gupta *et al.* [5] Approximation 1 (GA1)	+1, -1, -1	120	319	74	87
	Gupta *et al.* [5] Approximation 2 (GA2)	-2, -1, +1	128	359	82	83
	Yang *et al.* [6] (TGA1)	+1, -1	152	383	90	91
	Yang *et al.* [6] (TGA2)	+1, +1	120	319	66	67
	Liu *et al.* [7](MLAFA1)	+1, -1	192	647	130	131
	Liu *et al.* [7] (MLAFA2)	+1, -1	144	439	98	99
	Anusha and Deepa [8] (AA6)	+1, -1, +1	120	319	74	75
	Anusha and Deepa [8] (AA7)	+1, -1	160	551	114	115
	Anusha and Deepa [8] (AA8)	-1, +1, -1	152	511	106	107
	Anusha and Deepa [8] (AA9)	+1	168	535	106	107
	Anusha and Deepa [8] (AA10)	+1, +1, -1	144	471	106	91
	Anusha and Deepa [8] (AA11)	+1, -1	192	647	130	131
	Anusha and Deepa [8] (AA12)	+1	152	511	106	107
	Proposed RAFA-1	+2, -2	112	247	60	59
	Proposed RAFA-2	+2, -2	104	239	50	51

where a and b are the dimensions of the image. $Ex(j,k)$ and $App(j,k)$ represents the corresponding pixels in exact and approximate image outputs respectively. And, Max_{pixel} represents the maximum value of a pixel in the image.

However, PSNR fails to have consistency in the image quality as perceived by human eye [16]. To measure the efficacy of image processing applications in accordance with the perception of human visual system, another metric SSIM is used. It measures the structural similarity of the output images from approximate architecture and the output image from the exact computation based architecture. The expression for SSIM is given in equation (15).

$$SSIM(h,i) = \left[lumin(h,i)^{\alpha}\, contr(h,i)^{\beta}\, struct(h,i)^{\gamma}\right] \quad (15)$$

SSIM extracts the luminance ($lumin(h,i)$, contrast ($contr(h,i)$) and structure ($struct(h,i)$)) information of the

TABLE 6. SSIM and PSNR (dB) for Lena using the proposed reversible architectures with EFA and approximate full adders.

Design	Haar wavelet transform		Lifting scheme for Haar wavelet transform	
	SSIM	PSNR	SSIM	PSNR
EFA [12]	0.9820	31.99	0.9960	38.47
GA-1 [5]	0.8962	28.67	0.9001	33.14
GA-2 [5]	0.8317	24.35	0.8412	30.20
TGA1 [6]	0.9732	30.07	0.9867	36.27
TGA2 [6]	0.9695	29.32	0.9793	35.41
MLAFA-1 [7]	0.9773	30.89	0.9895	36.94
MLAFA-2 [7]	0.9767	30.15	0.9884	36.42
Anusha and Deepa [8] (AA6)	0.8732	28.42	0.8913	32.86
Anusha and Deepa [8] (AA7)	0.9716	29.94	0.9825	35.98
Anusha and Deepa [8] (AA8)	0.8843	28.42	0.8913	32.86
Anusha and Deepa [8] (AA9)	0.9792	31.08	0.9901	37.36
Anusha and Deepa [8] (AA10)	0.8784	28.05	0.8546	32.53
Anusha and Deepa [8] (AA11)	0.9725	30.01	0.9833	36.14
Anusha and Deepa [8] (AA12)	0.9789	31.42	0.9924	37.51
Proposed RAFA-1	0.9613	28.95	0.9746	34.67
Proposed RAFA-2	0.9634	29.18	0.9758	35.11

two images and compares it to estimate the image quality. Here h and i represent the non-negative image signal values. α, β and γ are positive parameters that can be dynamically varied while estimating SSIM to change the weightage of luminance, contrast and structure respectively. A detailed discussion of SSIM and PSNR is presented in [9], [16] and [16].

In this work, the image obtained from the FPGA implementation is compared with the reference input image using MATLAB. SSIM and PSNR for the proposed architectures are presented in Table 6. SSIM and PSNR are projected for proposed architectures using exact full adder and proposed approximate full adder. On an average, SSIM is found to be 0.9890 for exact FA based architectures and 0.9679 and 0.9696 for the proposed approximate FA based architectures RAFA-1 and RAFA-2 respectively. Similarly, PSNR estimated is found to be more for Lifting scheme, when the exact full adders are employed in the architectures. When the proposed approximate adders are used, the PSNR is reduced by an average of 3.5 dB, resulting in a PSNR above the acceptable level of 20 dB.

To further assess the efficiency of the proposed approximate adder based architectures for Haar wavelet transform and Lifting scheme for Haar wavelet transforms, QC × SSIM

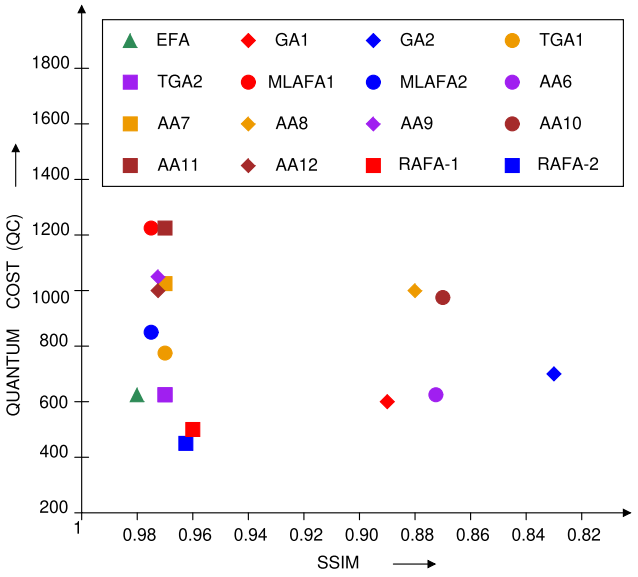


FIGURE 12. Figure of Merit - QC × SSIM for Haar wavelet transform.

(Figure of Merit (FOM)) is estimated and the graph is plotted for Haar wavelet transform and is shown in Figure 12. From Figure 12 it is evident that, MLAFA1 and MLAFA2 have the best SSIM close to EFA, but QC is as high as 1274 and 858 respectively. GA1 and AA10 has reduced QC at the cost of accuracy. This is due to the unequal numbers in ED introduced in positive and negative directions. EFA based architecture has QC of 618 and 0.9820 SSIM. The proposed RAFA-1 and RAFA-2 based architectures have the best optimisation among the approximate adders. These architectures are able to achieve more that 50% reduction in QC compared to MLAFA1 and MLAFA2 without significant reduction in SSIM. In addition, the proposed architectures are able to achieve comparable accuracy as that of EFA based architecture (with the least QC) when compared to existing approximate adder based architectures implemented using reversible logic.

VIII. CONCLUSION

This paper presents a novel architecture for Haar wavelet transform and lifting scheme for Haar wavelet transform using reversible logic gates. Architectures for one level forward and inverse haar wavelet transforms are implemented using reversible logic in this paper. The proposed architectures can be cascaded to obtain two level 2-D 2 × 2, one level 2-D 4 × 4 and two level 2-D 4 × 4 Haar transforms. The proposed designs are optimised for QC, CI, GO and GC. In addition, this paper proposes two approximate adders optimised for reversible logic based implementations. This can be employed in image processing applications, where the accuracy of computations can be compromised to expedite operations. The functional verification of the proposed designs was carried out for 512 × 512 image on FPGA platform. The image quality quantified in scales of SSIM and PSNR are in acceptable range for image processing applications. A first

in literature, the proposed work is a solution for low power image processing applications in quantum computers.

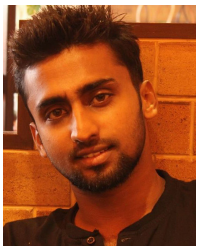
ACKNOWLEDGMENT

This Publication is an outcome of the Research and Development work undertaken in the project under the SMDP-C2SD, Visvesvaraya Ph.D. Scheme of Ministry of Electronics and Information Technology, Government of India, being implemented by Digital India Corporation (formerly Media Lab Asia).

REFERENCES

[1] P. J. Edavoor and A. D. Rahulkar, "Design and implementation of a novel low complexity symmetric orthogonal wavelet filter-bank," *IET Image Process.*, vol. 13, no. 5, pp. 785–793, Apr. 2019.
[2] G. E. Moore, "Cramming more components onto integrated circuits," *Proc. IEEE*, vol. 86, no. 1, pp. 82–85, Jan. 1998.
[3] R. Landauer, "Irreversibility and heat generation in the computing process," *IBM J. Res. Devel.*, vol. 5, no. 3, pp. 183–191, 1961.
[4] C. H. Bennett, "Logical reversibility of computation," *IBM J. Res. Develop.*, vol. 17, no. 6, pp. 525–532, Nov. 1973.
[5] V. Gupta, D. Mohapatra, S. P. Park, A. Raghunathan, and K. Roy, "IMPACT: IMPrecise adders for low-power approximate computing," in *Proc. IEEE/ACM Int. Symp. Low Power Electron. Design*, Aug. 2011, pp. 409–414.
[6] Z. Yang, J. Han, and F. Lombardi, "Transmission gate-based approximate adders for inexact computing," in *Proc. IEEE/ACM Int. Symp. Nanosc. Architectures (NANOARCH)*, Jul. 2015, pp. 145–150.
[7] W. Liu, T. Zhang, E. McLarnon, M. Oneill, P. Montuschi, and F. Lombardi, "Design and analysis of majority logic based approximate adders and multipliers," *IEEE Trans. Emerg. Topics Comput.*, early access, Jul. 17, 2019, doi: 10.1109/TETC.2019.2929100.
[8] G. Anusha and P. Deepa, "Design of approximate adders and multipliers for error tolerant image processing," *Microprocessors Microsyst.*, vol. 72, Feb. 2020, Art. no. 102940.
[9] (2016). *Sipi.usc.edu. SIPI Image Database*. [Online]. Available: http://sipi.usc.edu/database/
[10] L. Touil and B. Ouni, "Design of hardware RGB to HMMD converter based on reversible logic," *IET Image Process.*, vol. 11, no. 8, pp. 646–655, Aug. 2017.
[11] M. Awais, A. Razzaq, A. Ahmed, and G. Masera, "LDPC check node implementation using reversible logic," *IET Circuits, Devices Syst.*, vol. 13, no. 4, pp. 443–455, Jul. 2019.
[12] M. Aditya, Y. B. N. Kumar, and M. H. Vasantha, "Reversible full/half adder with optimum power dissipation," in *Proc. 10th Int. Conf. Intell. Syst. Control (ISCO)*, Jan. 2016, pp. 1–4.
[13] S. Kotiyal, H. Thapliyal, and N. Ranganathan, "Design of a reversible bidirectional barrel shifter," in *Proc. 11th IEEE Int. Conf. Nanotechnol.*, Aug. 2011, pp. 463–468.
[14] B. Parhami, "Fault-tolerant reversible circuits," in *Proc. 40th Asilomar Conf. Signals, Syst. Comput.*, Oct. 2006, pp. 1726–1729.
[15] J. W. Bruce, M. A. Thornton, L. Shivakumaraiah, P. S. Kokate, and X. Li, "Efficient adder circuits based on a conservative reversible logic gate," in *Proc. IEEE Comput. Soc. Annu. Symp. VLSI. New Paradigms VLSI Syst. Design. ISVLSI*, Apr. 2002, pp. 74–79.
[16] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity," *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600–612, Apr. 2004.

SITHARA RAVEENDRAN (Member, IEEE) received the bachelor's degree in electronics and communication engineering from the Cochin university of Science and Technology, Kerala, India, in 2008, and the master's degree in VLSI from the National Institute of Technology Goa, India, in 2016, where she is currently pursuing the Ph.D. degree with the Electronics and Communication Department. She has more than five years of industrial experience in design and verification of digital circuits (SOC level). Her research interests include digital design, reversible logic, FPGA, and ASIC design.



PRANOSE J. EDAVOOR (Member, IEEE) received the bachelor's degree in electronics and communication engineering from the Cochin university of Science and Technology, Kerala, India, in 2012, and the master's degree in VLSI from the National Institute of Technology Goa, India, in 2016, where he is currently pursuing the Ph.D. degree with the Electrical and Electronics Department. His research interests include digital design, FPGA accelerators, ASIC design, wavelets, and deep neural networks.

NITHIN Y. B. KUMAR (Senior Member, IEEE) received the B.E. degree from the Manipal Institute of Technology, Manipal, India, the M.Tech. degree from the National Institute of Technology Karnataka, Surathkal, India, and the Ph.D. degree from IIT Kharagpur, Kharagpur, India. He is currently an Assistant Professor with the National Institute of Technology Goa, Ponda, India, and a Visiting Researcher with the University of Pavia, Pavia, Italy.

M. H. VASANTHA (Member, IEEE) received the B.Tech. degree in electrical and electronics engineering from the NMAM Institute of Technology, Nitte, India, in 1998, the M.Tech. degree in microelectronics and very large scale integration design from IIT Madras, Chennai, India, in 2003, and the Ph.D. degree from the Department of Electronics and Communication Engineering, National Institute of Technology Karnataka, Surathkal, India, in 2014.