



ISSN: 2454-9940



**INTERNATIONAL JOURNAL OF APPLIED
SCIENCE ENGINEERING AND MANAGEMENT**

E-Mail :
editor.ijasem@gmail.com
editor@ijasem.org

www.ijasem.org

Input-Based Dynamic Reconfiguration of Approximate Arithmetic Units for Video Encoding

K. NAVYA¹, G. LAKSHMI BHARATH²

¹PG Student, Dept of ECE, SITS, Kadapa.

²Assistant Professor, Dept of ECE, SITS, Kadapa.

Abstract –

The field of approximate computing has received significant attention from the research community in the past few years, especially in the context of various signal processing applications. Image and video compression algorithms, such as JPEG, MPEG, and so on, are particularly attractive candidates for approximate computing, since they are tolerant of computing imprecision due to human imperceptibility, which can be exploited to realize highly power-efficient implementations of these algorithms. However, existing approximate architectures typically fix the level of hardware approximation statically and are not adaptive to input data. For example, if a fixed approximate hardware configuration is used for an MPEG encoder (i.e., a fixed level of approximation), the output quality varies greatly for different input videos. This paper addresses this issue by proposing a reconfigurable approximate architecture for MPEG encoders that optimizes power consumption with the goal of maintaining a particular Peak Signal-to-Noise Ratio (PSNR) threshold for any video. Toward this end, we design reconfigurable adder/subtractor blocks (RABs), which have the ability to modulate their degree of approximation, and subsequently integrate these blocks in the motion estimation and discrete cosine transform modules of the MPEG encoder. We propose two heuristics for automatically tuning the approximation degree of the RABs in these two modules during runtime based on the characteristics of each individual video. Experimental results show that our approach of dynamically adjusting the degree of hardware approximation based on the input video respects the given quality bound (PSNR degradation of 1%–10%) across different videos while achieving a power saving up to 38% over a conventional non-approximate MPEG encoder architecture. Note that although the proposed reconfigurable approximate architecture is presented for the specific case of an MPEG encoder, it can be easily extended to other DSP applications.

Keywords – Approximate circuits, approximate computing, low power design, quality configurable.

I. INTRODUCTION

Introducing a limited amount of computing imprecision in image and video processing algorithms often results in a negligible amount of perceptible visual change in the output, which makes these algorithms as ideal candidates for the use of approximate computing architectures. Approximate computing architectures exploit the fact that a small relaxation in output correctness can result in significantly simpler and lower power implementations. However, most approximate hardware architectures proposed so far suffer from the limitation that, for widely varying input parameters, it becomes very hard to provide a quality bound on the output, and in some cases, the output quality may be severely degraded. The main reason for this output quality fluctuation is that the degree of approximation (DA) in the hardware architecture is fixed statically and cannot be customized for different inputs. In the proposed new video coding benchmarks, the intricacy of an encoder is by and large significantly higher than that of a decoder since some encoding particular segments, for example, movement estimation, have computationally escalated

operations notwithstanding when proficient quick movement seek is utilized. Such design is reasonable for downlink transmission model of TV communicate when the framework has few encoders and various decoders. In contemporary media-rich uplink remote video transmission, for example, a camcorder phone transmits video remotely to the base station, multifaceted nature is of fundamental concern since battery-fueled versatile handheld gadgets as a rule have restricted preparing influence and memory. In this way, it is alluring to have a low intricacy video encoder to meet the asset restrictions. As of late, dispersed video coding plans have been proposed to give potential turn around in computational multifaceted nature for decoder and encoder. The hypothetical foundation of these plans depends on Slepian-Wolf and Wyner-Ziv appropriated source coding speculations. The Slepian-Wolf hypothesis expresses that two measurably subordinate discrete signs, for a Gaussian irregular flag and its side data, the restrictive Rate Mean Squared Error Distortion work for this flag is the same as the situation when its side data is accessible just at the decoder. One possible remedy is to adopt a conservative approach and use a very low DA in the hardware so that the output accuracy is not drastically affected. However, such a conservative approach will, as expected, drastically impact the power savings as well. This paper adopts a different approach to addressing this problem by dynamically reconfiguring the approximate hardware architecture depending on the inputs.

II. PROPOSED ARCHITECTURE

This area depicts the distinctive strides followed in developing our proposed reconfigurable design and how it was installed inside the MPEG encoder.

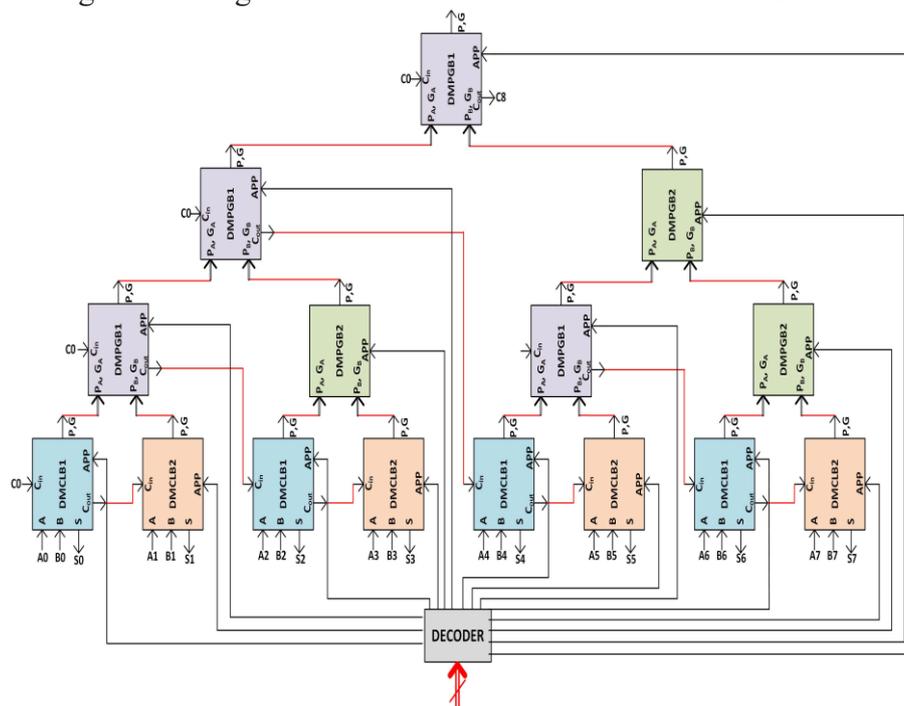


Fig:1. 8-bit reconfigurable CLA block.

A. Reconfigurable Adder/Subtractor Blocks

Dynamic variety of the DA should be possible when each of the snake/subtractor pieces is outfitted with one or more of its estimated duplicates and it can switch between them according to prerequisite. This reconfigurable engineering can incorporate any estimated rendition of the adders/subtractors. As a kind of perspective, proposed six various types of surmised circuits for adders. In any case, it likewise should be guaranteed that the extra territory over heads required for building the reconfigurable rough circuits are

negligible with sufficiently vast power reserve funds. As cases, we have picked the two most credulous techniques exhibited, to be specific, truncation and estimate 5, for approximating the snake/subtractor squares. The last one can likewise be conceptualized as an upgraded rendition of truncation as it just transfers the two 1-bit inputs, one as Sum and alternate as Carry Out (Choice 2). On the off chance that A, B, and C in are the 1-bit inputs to the full adder (FA), at that point the outputs are Sum= B and Cout = A. The resultant truth-table demonstrates that the yields are right for the greater part of all information mixes, in this way ended up being a superior guess mode than truncation. The proposed conspire replaces every FA cell of the adders/subtractors with a double mode FA (DMFA) cell in which every FA cell can work either in completely exact or in some estimation mode relying upon the condition of the control flag APP.

A rationale high estimation of the APP flag means that the DMFA is working in the rough mode. We term these adders/subtractors as RABs. Note that the FA cell is control gated while working in the inexact mode. Union and assessment of energy utilization of a 16-bit RCA were performed in Synopsys Design and Power Compiler and the relating comes about are portrayed in Table I. Our analyses have demonstrated an insignificant distinction in the power utilization of DMFA when worked in both of the two estimation modes. Thus, with no loss of sweeping statement, estimation 5 was decided for its higher probability of giving the right yield result than truncation, which constantly yields 0 independent of the information. Figure demonstrates the rationale square chart of the DMFA cell, which replaces the constituent FA cells of an 8-bit RCA, as appeared in Figure. What's more, it additionally comprises of the estimate controller for producing the proper select signs for the multiplexers. A multimode FA cell would provide even a superior contrasting option to the DMFA from the purpose of controlling the estimate extent. Be that as it may, it likewise expands the many-sided quality of the decoder piece utilized for affirming the privilege select signs to the multiplexers and in addition the rationale overhead for the multiplexers themselves. This undermines the essential target as the majority of the power funds that we get from approximating the bits are lost. Rather, the two-mode decoder and the 2:1 multiplexers have immaterial overhead and furthermore give sufficient summon over the guess degree. 1) DMFA Overhead: The power gating transistor and the multiplexers of the DMFA are intended to bring about the minimum conceivable overhead. Our trials demonstrate that exchanging energy of the CMOS transistors contributes toward a large portion of the aggregate power utilization of the FA and DMFA squares. It exhibits the power utilization of FA and DMFA for various modes gotten by comprehensive reproduction in Synopsys Nano Sim. It demonstrates that the power increments by 0.21 μ W when we work DMFA in precise mode as contrasted and the first FA piece. This distinction in power can be ascribed principally to the expansion in stack capacitance of the FA hinder because of the expansion of the information capacitance of the interfaced multiplexers. A little segment of the aggregate power is contributed by the extra exchanging of the multiplexers. Decrease in the info exchanging movement of the multiplexers is additionally an auxiliary reason for this little measure of energy. The extra overhead because of exchanging of the power gating transistor can be ignored, since its exchanging action is little because of the idea of our exchanging calculations.

This is basically due to the spatial and worldly region of the pixel esteems crosswise over continuous edges. The idea of RAB can likewise be reached out to other snake designs too. Snake structures, for example, CBA and CSA, which additionally contain FA as the crucial building square, can be made precision configurable by coordinate substitution of the FAs with DMFAs. Different assortments, as CLA and tree adders, utilize diverse sorts of convey engender and produce obstructs as their fundamental building units, and henceforth require some extra modifications to work as RABs. For instance, we executed a 16-bit CLA

comprising of four distinct sorts of fundamental pieces contingent on the nearness of total (S), Cout, convey engendering (P), and convey era (G) at various levels. We address the fundamental squares display at the first (or lowermost) level of a CLA, which have inputs coming in specifically, as convey lookahead pieces, CLB1 and CLB2. The distinction among them being that CLB1 produces an extra C out signal compared with CLB2. Their relating double mode renditions, DMCLB1 and DMCLB2, have both S and P approximated by input operand B and both Cout and G approximated by input operand An, as appeared in Figure. The fundamental pieces introduce at the larger amounts of CLA progressive system are indicated as proliferate and produce squares, PGB1 and PGB2. For this situation, PGB1 produces an additional Cout yield as contrasted and PGB2. As appeared in Figure, the configurable double mode forms, DMPGB1 and DMPGB2, utilize inputs PA and GB as approximations for yields P and G, individually, while working in the inexact mode. These approximations were chosen observationally guaranteeing that the proportion of the likelihood of correct output to the extra circuit overhead for each of the pieces is extensive. Table II compresses the yields of each of the double mode squares while working in either exact or surmised mode. For a reconfigurable CLA, DMCLB1 and DMCLB2 squares are approximated as per the DA. Be that as it may, the DMPGB1 and DMPGB2 pieces are approximated just when every last DMCLB1, DMCLB2, DMPGB1, and DMPGB2 square, which has a place with the transitive fan-in cones of the concerned square, is approximated. Something else, the square is worked in the precise mode.

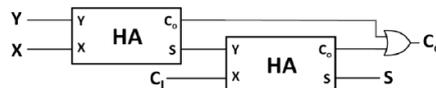
For instance, any DMPGB hinder at the second level of CLA can be made to work in surmised mode, if and just if, both of its constituent DMCLB1 and DMCLB2 squares are working in the inexact mode. Comparative convention is followed for the squares dwelling at larger amounts of the tree, where each DMPGB piece can be approximated just when both of its constituent DMPGB1 and DMPGB2 pieces are approximated. This engineering can be effortlessly extrapolated to other comparable sort CLAs, for example, Kogge–Stone, Brent–Kung, Manchester-convey chain, et cetera. Figure demonstrate a near investigation of the power utilization of the diverse sorts of adders when the DA is fluctuated. Specifically, the figures signify the standardized power utilization of the distinctive sorts of RABs when the quantity of bits approximated is fluctuated. A fascinating perception for CSA is that approximating its MSBs gives more noteworthy power reserve funds than LSB estimation per bit. This can be ascribed to the design of the convey spare adders, where approximating each piece in the MSB brings about power gating of two FAs contrasted and one FA when the LSBs are approximated. The graphs additionally portray that genuine power funds are started when the DA is equivalent to or over 5. This is the point where the investment funds because of estimate outperforms the overhead brought about because of the extra multiplexers, control gating transistors, and controller. The inalienable blunder versatility appeared by the ME and the little contributions to the DCT piece give sufficient chances to accomplishing a high DA (significantly more prominent than 5) and in this way high power reserve funds.

B. Full adder:

In a previous work, we saw how a half adder can be used to determine the sum and carry of two input bits. Consider the possibility that we have three info bits—X, Y, and CI, where CI is a carry in that speaks to the complete from the past less critical piece expansion. In this circumstance, we have what is known as a FULL ADDER—a circuit that includes three one-piece esteems. These qualities are the addends X a Y, and convey in CI. At the point when three single-piece esteems are included, the most astounding conceivable outcome would be $1 + 1 + 1 = 11$, which is the paired portrayal of the decimal number 3. The whole truth table for the FULL ADDER would resemble this:

X	Y	CARRY IN C_i	CARRY OUT C_o	SUM S
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1

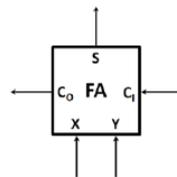
CoS taken together, represents the binary sum $X+Y+C_i$. For example, when CoS is 11 (corresponding to decimal 3), then $X+Y+C_i = 1+1+1$.



One approach to construct a FULL ADDER is to utilize two half adders as appeared in this circuit chart:

The half adder on the left figures the entirety and convey for the addends X and Y. This aggregate and the convey in are at that point included by the half-adder the privilege, creating a last total and a convey bit. There is a CO do assuming either or both of the two convey bits are ON—clarifying the utilization of the OR entryway on the far upper right of the circuit graph.

The FULL ADDER (FA for short) circuit can be represented in a way that hides its innerworkings: The FULL ADDER can then be assembled into a cascade of full adders to add two binary numbers. For example the diagram below shows how one could add two 4-bit binary numbers $X_3X_2X_1X_0$ and $Y_3Y_2Y_1Y_0$ to obtain the sum $S_3S_2S_1S_0$ with a final carry-out C_4 .



C. 4-bit Carry Ripple Adder:

Assume you want to add two operands A and B were

$A = A_3 A_2 A_1 A_0$

$B = B_3 B_2 B_1 B_0$

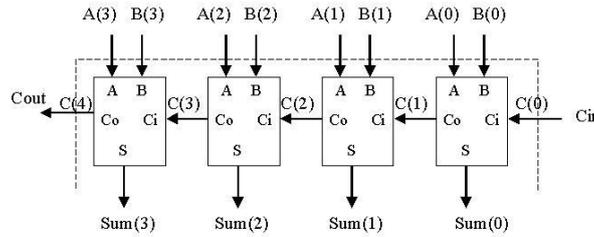
For example:

$A = 1\ 0\ 1\ 1 +$

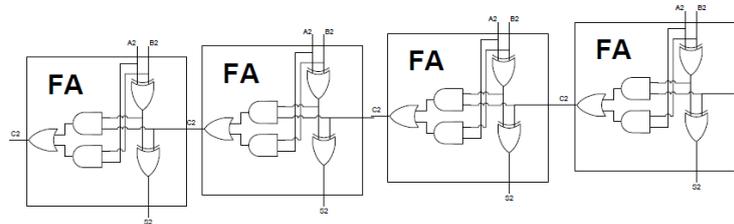
$B = 1\ 1\ 0\ 1$

$A+B = 11\ 0\ 0\ 0 = C\ out\ S_3\ S_2\ S_1\ S_0$

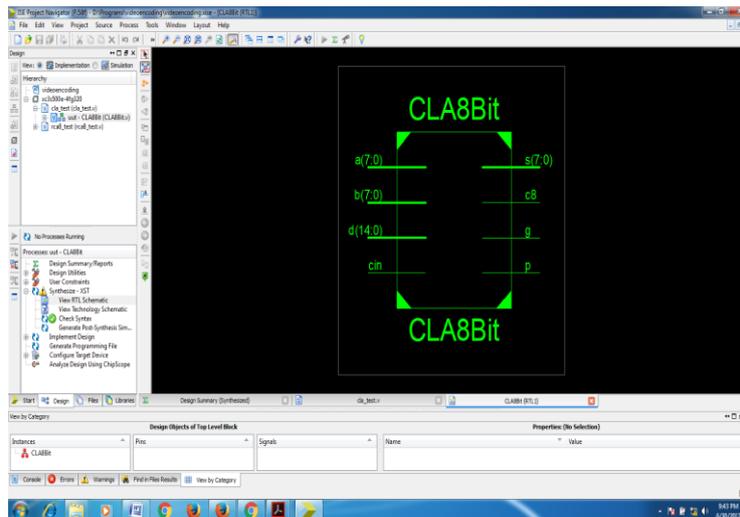
From the example above it can be seen that we are adding 3 bits at a time sequentially until all bits are added. A full adder is a combinational circuit that plays out the number-crunching aggregate of three information bits: augends A_i , numbers to be added B_i and carry in from the past stage. Its outcomes contain the whole S_i and the do, to the following stage. in $C\ out\ C$



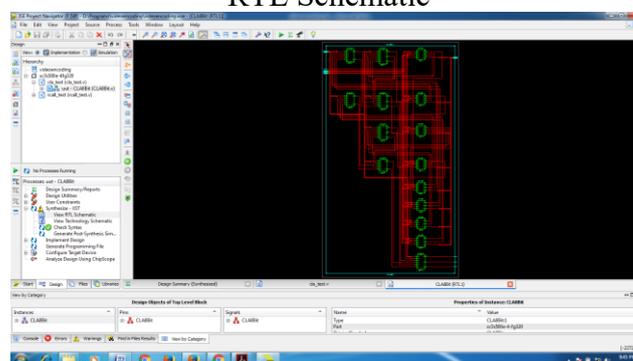
So, to design a 4-bit adder circuit we start by designing the 1-bit full adder then connecting the four 1-bit full adders to get the 4-bit adder as shown in the diagram above. For the 1-bit full adder, the design begins by drawing the Truth Table for the three input and the corresponding output SUM and CARRY. The Boolean Expression describing the binary adder circuit is then deduced. The binary full adder is a three-input combinational circuit which satisfies the truth table above.



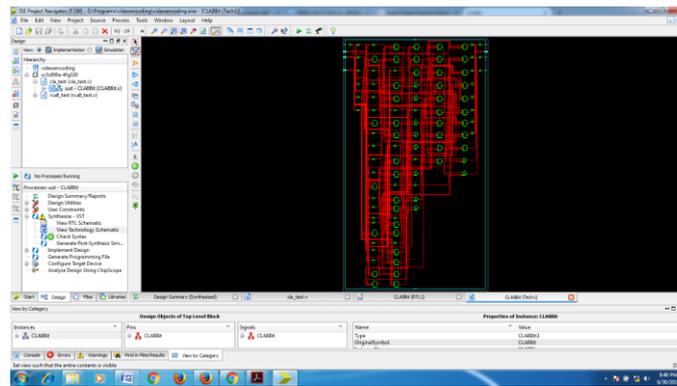
III. OUTPUT RESULTS



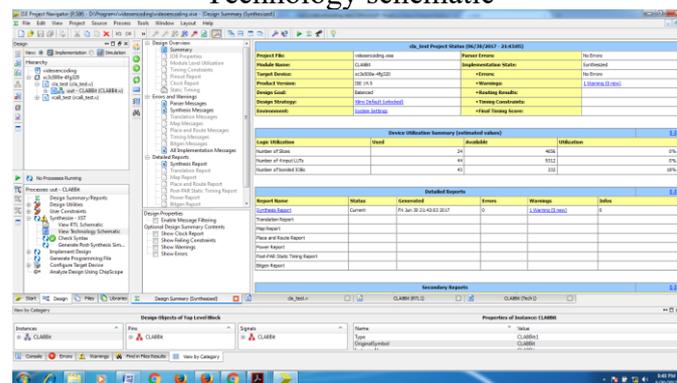
RTL Schematic



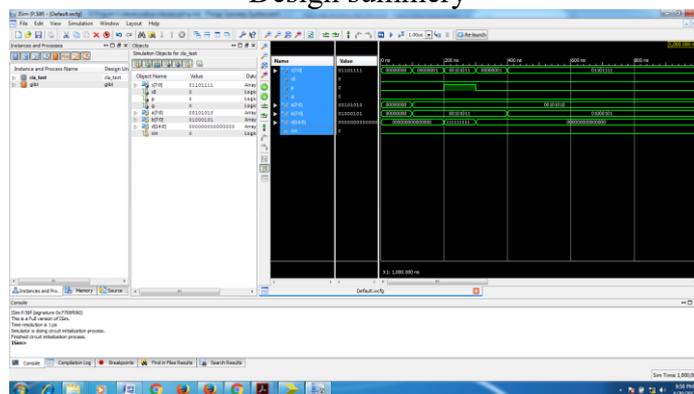
RTL Schematic internal diagram



Technology schematic



Design summary



Simulation output

TIMING REPORT

NOTE: THESE TIMING NUMBERS ARE ONLY A SYNTHESIS ESTIMATE.

FOR ACCURATE TIMING INFORMATION PLEASE REFER TO THE TRACE REPORT

GENERATED AFTER PLACE-and-ROUTE.

Clock Information:

No clock signals found in this design

Asynchronous Control Signals Information:

No asynchronous control signals found in this design

Timing Summary:

Speed Grade: -4

Minimum period: No path found

Minimum input arrival time before clock: No path found
 Maximum output required time after clock: No path found
 Maximum combinational path delay: 11.822ns

Timing Detail:

 All values displayed in nanoseconds (ns)

=====

Timing constraint: Default path analysis

Total number of paths / destination ports: 435 / 11

Delay: 11.822ns (Levels of Logic = 8)

Source: a<0> (PAD)

Destination: s<5> (PAD)

Data Path: a<0> to s<5>

Cell:in->out	Gate	Net	Delay	Delay	Logical Name (Net Name)
IBUF:I->O	4	1.218	0.762	a_0_IBUF	(a_0_IBUF)
LUT3:I0->O	4	0.704	0.622	clb11/g1	(g0)
LUT4:I2->O	1	0.704	0.000	pgb11/g_G	(N80)
MUXF5:I1->O	3	0.321	0.706	pgb11/g	(g8)
LUT3:I0->O	2	0.704	0.482	pgb13/cout1	(c7)
LUT4:I2->O	1	0.704	0.499	clb13/cout1	(c3)
LUT4:I1->O	1	0.704	0.420	clb23/s1	(s_5_OBUF)
OBUF:I->O		3.272		s_5_OBUF	(s<5>)

Total		11.822ns (8.331ns logic, 3.491ns route)			
		(70.5% logic, 29.5% route)			

Total REAL time to Xst completion: 8.00 secs

Total CPU time to Xst completion: 8.48 secs

CONCLUSION

This paper proposed a reconfigurable approximate architecture for the MPEG encoders that optimize power consumption while maintaining output quality across different input videos. The proposed architecture is based on the concept of dynamically reconfiguring the level of approximation in the hardware based on the input characteristics. It requires the user to specify only the overall minimum quality for videos instead of having to decide the level of hardware approximation. Our simulation results show that the proposed architecture results in power savings equivalent to a baseline approach that uses fixed approximate hardware while respecting quality constraints across different videos. Future work includes the incorporation of other approximation techniques and extending the approximations to other arithmetic and functional blocks.

REFERENCES

- [1] M. Elgamel, A. M. Shams, and M. A. Bayoumi, "A comparative analysis for low power motion estimation VLSI architectures," in *Proc. IEEE Workshop Signal Process. Syst. (SiPS)*, Oct. 2000, pp. 149–158.
- [2] F. Dufaux and F. Moscheni, "Motion estimation techniques for digital TV: A review and a new contribution," *Proc. IEEE*, vol. 83, no. 6, pp. 858–876, Jun. 1995.

- [3] I. S. Chong and A. Ortega, "Dynamic voltage scaling algorithms for power constrained motion estimation," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, vol. 2, Apr. 2007, pp. II-101–II-104.
- [4] I. S. Chong and A. Ortega, "Power efficient motion estimation using multiple imprecise metric computations," in *Proc. IEEE Int. Conf. Multimedia Expo*, Jul. 2007, pp. 2046–2049.
- [5] D. Mohapatra, G. Karakonstantis, and K. Roy, "Significance driven computation: A voltage-scalable, variation-aware, quality-tuning motion estimator," in *Proc. 14th ACM/IEEE Int. Symp. Low Power Electron. Design (ISLPED)*, 2009, pp. 195–200.
- [6] J. George, B. Marr, B. E. S. Akgul, and K. V. Palem, "Probabilistic arithmetic and energy efficient embedded signal processing," in *Proc. Int. Conf. Compil., Archit., Synth. Embedded Syst. (CASES)*, 2006, pp. 158–168.
- [7] D. Shin and S. K. Gupta, "A re-design technique for data path modules in error tolerant applications," in *Proc. 17th Asian Test Symp. (ATS)*, 2008, pp. 431–437.
- [8] S. Venkataramani, A. Sabne, V. Kozhikkottu, K. Roy, and A. Raghunathan, "SALSA: Systematic logic synthesis of approximate circuits," in *Proc. 49th Annu. Design Autom. Conf. (DAC)*, Jun. 2012, pp. 796–801.
- [9] V. Gupta, D. Mohapatra, S. P. Park, A. Raghunathan, and K. Roy, "IMPACT: IM Precise adders for low-power approximate computing," in *Proc. 17th IEEE/ACM Int. Symp. Low-Power Electron. Design (ISLPED)*, Aug. 2011, pp. 409–414.
- [10] V. Gupta, D. Mohapatra, A. Raghunathan, and K. Roy, "Low power digital signal processing using approximate adders," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 32, no. 1, pp. 124–137, Jan. 2013.
- [11] V. G. Moshnyaga, K. Inoue, and M. Fukagawa, "Reducing energy consumption of video memory by bit-width compression," in *Proc. Int. Symp. Low Power Electron. Design (ISLPED)*, 2002, pp. 142–147.
- [12] Z. He and M. L. Liou, "Reducing hardware complexity of motion estimation algorithms using truncated pixels," in *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS)*, vol. 4, Jun. 1997, pp. 2809–2812.
- [13] Z.-L. He, C.-Y. Tsui, K.-K. Chan, and M. L. Liou, "Low-power VLSI design for motion estimation using adaptive pixel truncation," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 10, no. 5, pp. 669–678, Aug. 2000.
- [14] A. Raha, H. Jayakumar, and V. Raghunathan, "A power efficient video encoder using reconfigurable approximate arithmetic units," in *Proc. 27th Int. Conf. VLSI Design, 13th Int. Conf. Embedded Syst.*, Jan. 2014, pp. 324–329.
- [15] P. M. Kuhn, *Algorithms, Complexity Analysis and VLSI Architectures for MPEG-4 Motion Estimation*, 1st ed. Norwell, MA, USA: Kluwer, 1999.
- [16] K. Seshadrinathan, R. Soundararajan, A. C. Bovik, and L. K. Cormack, "Study of subjective and objective quality assessment of video," *IEEE Trans. Image Process.*, vol. 19, no. 6, pp. 1427–1441, Jun. 2010.
- [17] S. Winkler, "Video quality measurement standards—Current status and trends," in *Proc. 7th Int. Conf. Inf., Commun., Signal Process. (ICICS)*, Dec. 2009, pp. 1–5.