

Dynamic Reconfiguration of Approximate Arithmetic Unit for video Encoding

Rahul Gupta and S. Yuvaraj

ABSTRACT

Low Power is a crucial requirement for portable multimedia devices making use of various signal processing algorithms and architectures. Human beings are unable to mark slightly erroneous Outputs which most of the multimedia applications produces. Consequently, we do not require to produce precisely correct numerical outputs. In this paper, we recommend logic complexity reduction at the transistor level. We demonstrate this concept, by proposing various approximate adder due to which complexity is reduced as the number of transistor is reduced and using them to design approximate adders. When compared to existing implementations using accurate adder, simulation results specifies power saving using the proposed approximate adders. By utilizing approximate adder, 16-bit CLA is implemented consisting of four different types of basic blocks depending upon the presence of carry propagation (P), carry generation (G), sum (S) and C_{out} at different levels.

Index Terms: Approximate circuits, approximate computing, low power design.

1. INTRODUCTION

The backbone of different types of multimedia applications is DIGITAL SIGNAL processing (DSP) blocks used in portable devices. Image and Video processing algorithms are implemented by various DSP blocks, in which either an image or a video for human consumption is the Ultimate Output. Human beings have finite processing sensory information capability when interpreting an image or a video[1]. This permits the Outcome of this algorithm to be integrally approximate rather than precise. This diversion on numerical exactness provides some freedom to carry out imprecise or approximate computation. Today, there are an increasing number of portable applications with limited amount of power available, requiring low power, small area and high throughput circuitry. Therefore, the circuits which consume low power are the major concern for designing system components. Low power VLSI systems have become prominent as exceeding in demand and the research effort in low power microelectronics has been increased. In highly integrated Nano design, reliability issues resulting from PVT (process, voltage and temperature) variations, Moreover static power and leakage are major concerns for the high power consumption. Different methods of power-reduction include algorithmic modifications [2], [3], voltage over-scaling [4] and imprecise computation of metrics [5]. A possible solution to lower power dissipation is to employ approximate circuit design[3]. Mostly used Multimedia applications have digital signal processing (DSP) blocks as key block. The ultimate output for most of these DSP blocks implement algorithms, is either a video or an image for human analysis and presentation[2]. For example, the limited perception of human vision allows the outputs of these algorithms to be numerically approximate rather than accurate.

1.1. CONVENTIONAL ADDER

Adder is one of the most important components of a Arithmetic logic unit (ALU), floating point unit and CPU (central processing unit), and address generation of memory or cache access unit. Due to increasing

* Mtech VLSI design, ECE Department SRM UNIVERSITY CHENNAI 603203 INDIA, Email: guptarahul609@gmail.com

** Asst.Proff(O.G), ECE Department SRM UNIVERSITY CHENNAI 603203 INDIA, Email: yuvasivasanthi@gmail.com

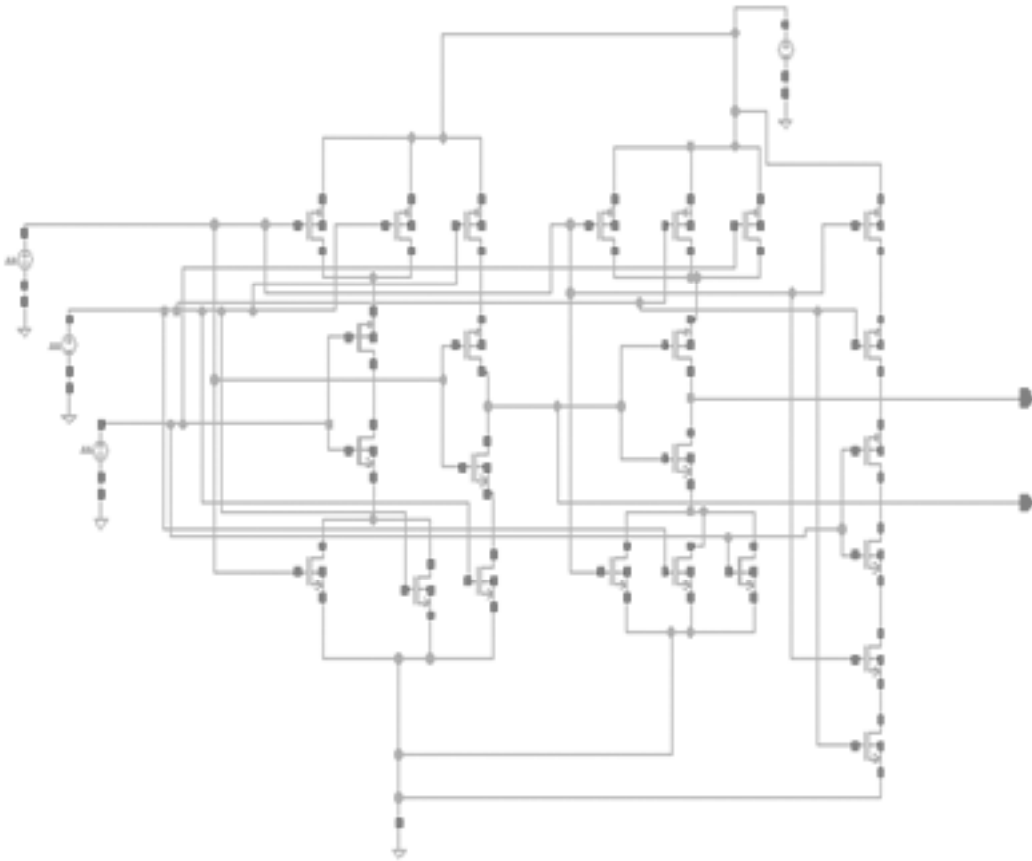


Figure 1(a): Schematic design of Conventional Full adder

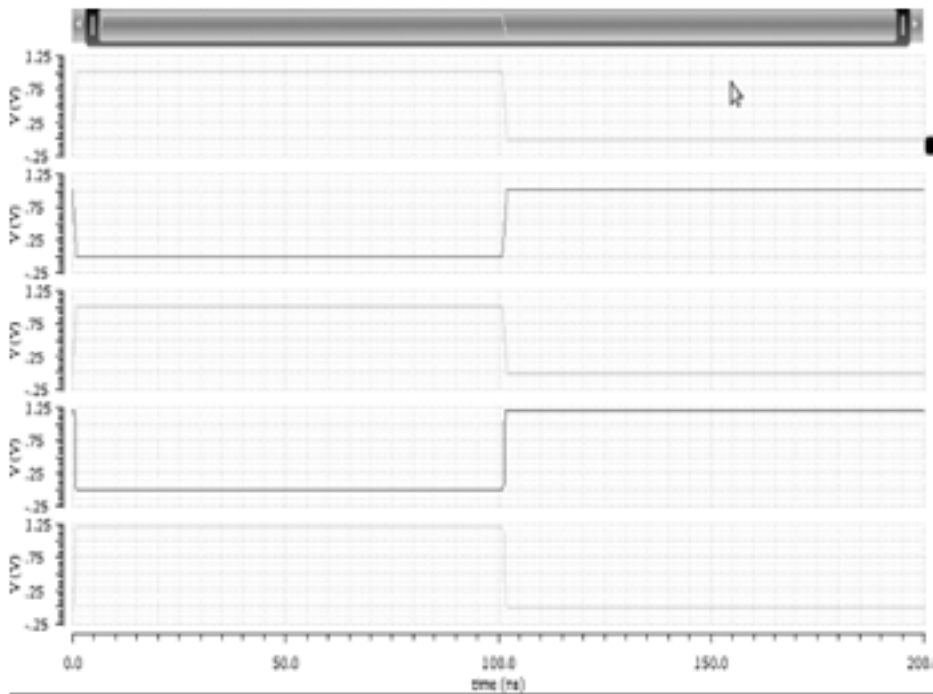


Figure 1(b): Waveform of Conventional Full adder

demand for portable equipments the need of using area and Power efficient VLSI circuits has arised, such as personal digital assistant (PDA), Notebook personal computer and cellular phones. Fig.1 shows the schematic design of a conventional Adder which is most common way of implementing an Full Adder. Total number of transistor in implementing Full adder is 24 transistors. As this is not based on complementary

CMOS logic implementation, hence it provides a good opportunity to design an approximate version by removing the selected transistors.

1.2. Approximate adder

In various approximate implementations multiple – bit adders are classified into two modules: the (accurate) upper part of more significant bits and the (approximate) lower part of less significant bit[6],[7]. A close observation of the Full Adder truth table shows that $C_{out} = A$ for six out of eight cases. Similarly, $C_{out} = B$ for six out of eight cases. Since A and B are interchangeable, we consider $C_{out} = A$. Hence, we propose a fourth approximation where we use an inverter with input A to calculate C_{out} bar and Sum is calculated similar to approximation 4. This introduces two errors in C_{out} and three errors in Sum.

2. 1-BIT DUAL MODE FULL ADDER (DMFA)

The proposed scheme replace each full adder cell with a dual-mode full adder (DMFA) cell (Fig. 3) in which full adder can either operate in fully accurate mode or in approximation mode which depend on the state of the control signal APP. A high logic value of the approximation signal indicates that the DMFA is operating in approximate mode [$APP = 1$]. A low logic value of the approximation signal indicates that the DMFA is operating in accurate mode [$APP = 0$].

This adder & subtractors is termed as RABs. When operating in the approximate mode, it is important to note that Full adder cell is power-gated. Synthesis and evaluation of power consumption were performed in cadence. Our experimental results show a negligible difference in power consumption

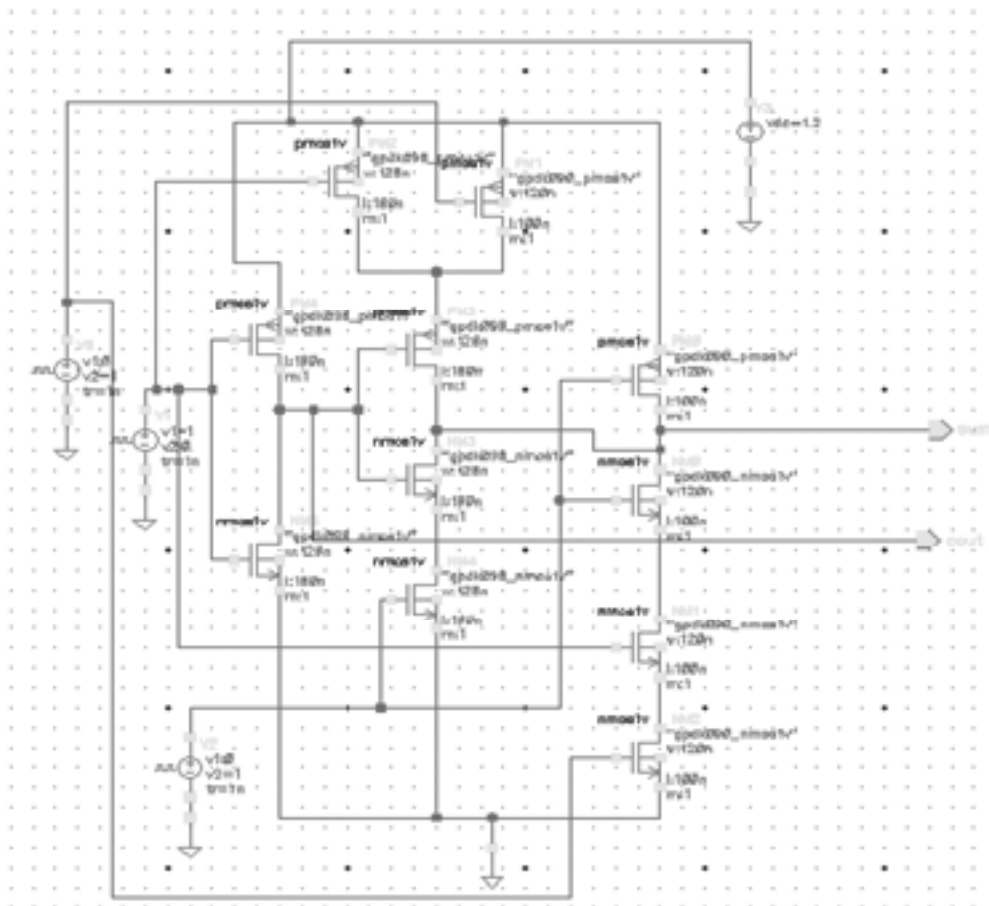


Figure 2: Schematic design of approximation adder

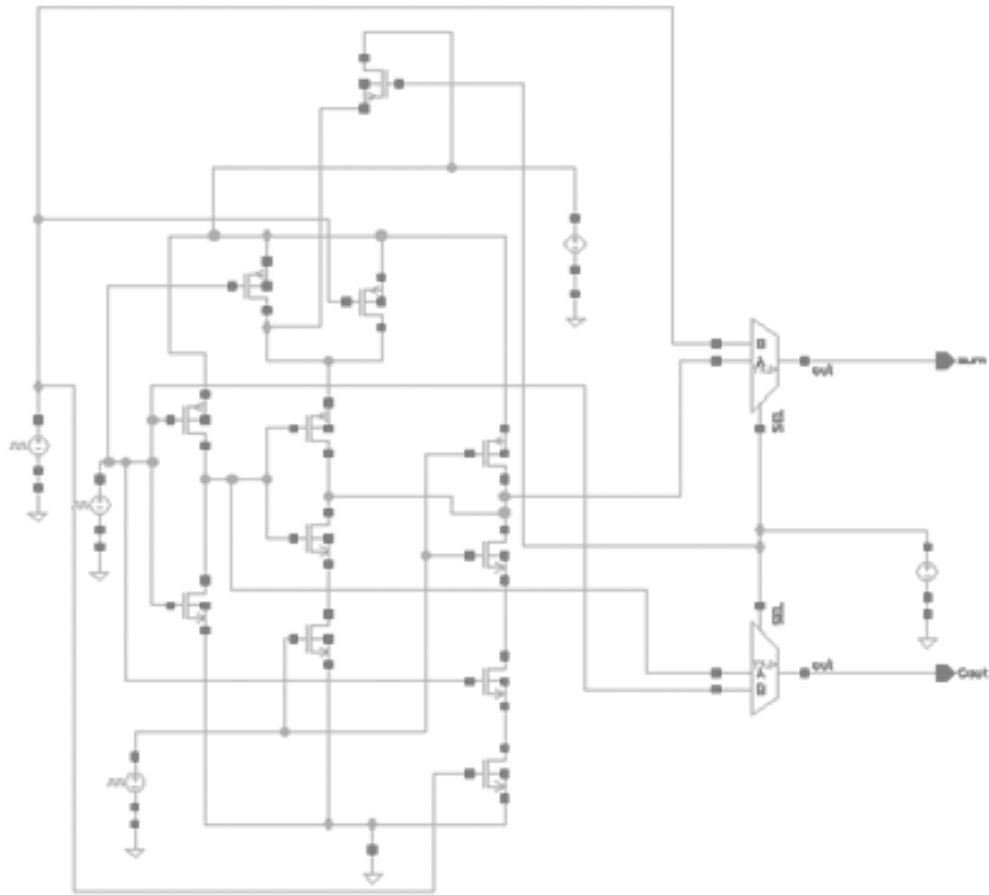


Figure 3(a): Schematic design of DMFA operating in approximate mode and in accurate mode

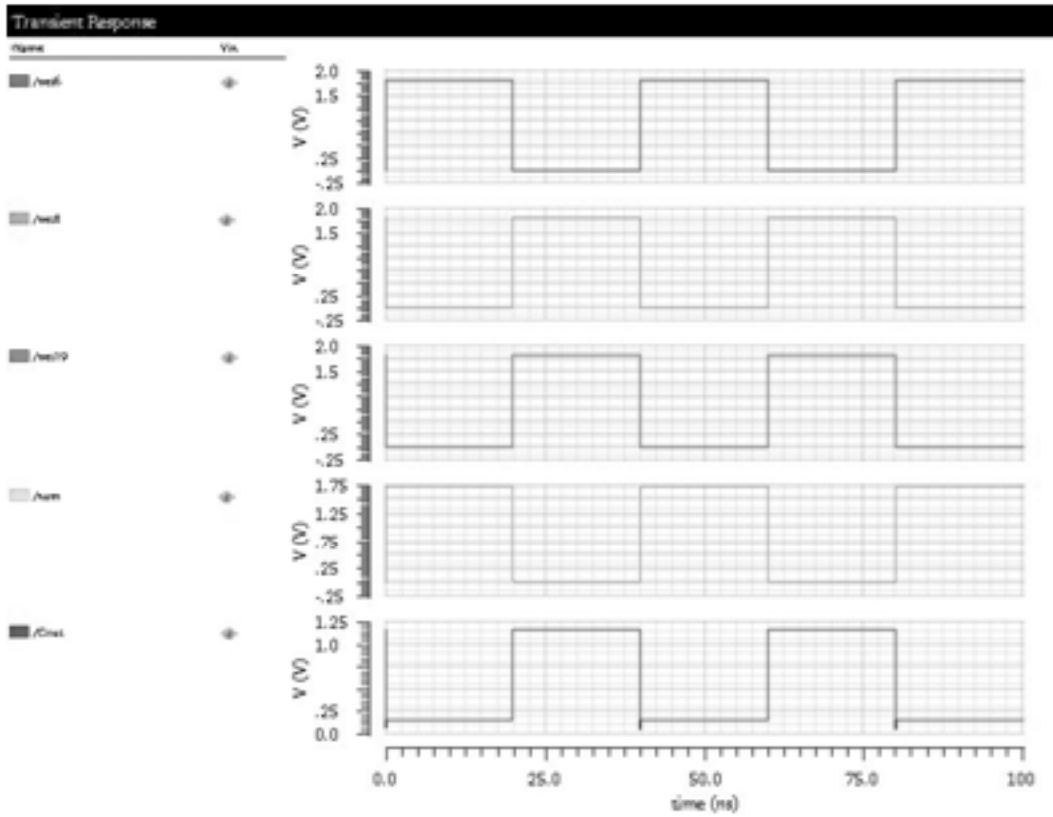


Figure 3(b): Waveform of DMFA

Table 1
Power Consumption Of Different Dmfa Modes

Conventionalfull adder	Dual mode full adder operating in Approximate mode	Dual mode full adder operating in Accurate mode
8uW	9.4Uw	2.8Uw

when operated in either of the two approximation modes of DMFA. Thus, approximate adder was selected for its higher probability of providing the correct output result than truncation, without any loss of generality[8]. Fig. 5 displays the logic block diagram of the DMFAcell, which replaces the constituent Full adder cells of 8-bit RCA. From the point of controlling the approximation magnitude, a multimode Full Adder cell would provide better alternative to the DMFA. Moreover, this increases the complexity of the decoder block used for asserting the right select signals to the multiplexers as well as the logic overhead for the multiplexers themselves. This reduces the power of the primary objective as most of the power savings that we get from approximating the bits are lost[9]. Therefore, the 2:1 multiplexers and the two-mode decoder have negligible overhead and also provide adequate command over the approximation degree

2.1. DMFA overhead

To incur the least possible overhead the power gating transistor and the multiplexers of the DMFA are designed. Our experimental result shows switching power of CMOS transistors contributes most of the power consumption of DMFA and full adder blocks. Table I presents the power consumption of conventional adder, DMFA operating in approximation mode and DMFA operating in accurate mode obtained by exhaustive simulation in cadence. Simulation result shows that power increases when we operate DMFA in accurate mode as compared with the conventional Adder block. Due to the additional number of input capacitance of the interfaced multiplexers, this difference in power can be attributed mainly to increase in load capacitance of the FA block. The additional switching of the multiplexers contributes a small portion of the total power. The input switching activity of the multiplexers is reduced which is also a secondary cause for this small amount of power.

The concept of RAB can also be extended to other adder architectures as well. Adder architectures, such as CBA and CSA, which also contain FA as the fundamental building block, can be made accuracy configurable by direct substitution of the FAs with DMFAs. Other varieties, like CLA and tree adders, use different types of carry propagate and generate blocks as their basic building units, and hence require some additional modifications to function as RABs.

3. 16-BIT RECONFIGURABLE CLA BLOCK

In this paper implemented a 16-bit CLA which consist of four different types of basic blocks (Fig.4) depending upon the presence of sum (S), Cout, carry propagation (P), and carry generation (G) at different levels. We address this basic blocks present at the first (or lowermost) level of a carry lookahead adder, which have inputs coming in directly, as carry lookahead blocks, CLB1 and CLB2[10]. The difference among them being that CLB1 produces an additional Cout signal compared with CLB2. Their corresponding dual-mode versions, DMCLB1 and DMCLB2, have both S and P approximated by input operand B and both Cout and G approximated by input operand A , as shown in Fig. 4. The basic blocks present at the higher levels of CLA hierarchy are denoted as propagate and generate blocks, PGB1 and PGB2. In this case, PGB1 produces an extra Cout output as compared with PGB2. As shown in Fig. 4, the configurable dual-mode versions, DMPGB1 and DMPGB2, use inputs PA and GB as approximations for outputs P and G respectively, when operating in the approximate mode.

Table 2
Dual-mode Block Outputs For Accurate And Approximate Modes

Basic Block(adder type)	Outputs for App = 0 (accurate mode)	Output for APP = 1 (approximate mode)
DMFA(RCA, CBA,CSA)	$S=A\oplus B\oplus C_{in}$ $C_{out}=AB+Bc_{in}+AC_{in}$	$S=BC_{out}=A$
DMCLB1(CLA)	$P=A\oplus BG=ABS=P\oplus C_{in}$ $C_{out}=G+PC_{in}$	$P=AG=AS=BC_{out}=A$
DMCLB2(CLA)	$P=A\oplus BG=ABS=P\oplus C_{in}$	$P=BG=AS=B$
DMPGB1(CLA)	$P=PA$ $PBG=GB+GAP$ $C_{out}=G+PC_{in}$	$P=PAG=GB$ $C_{out}=G+PC_{in}$
DMPGB2(CLA)	$P=PA$ $PBG=GB+GA$ PB	$P=PAG=GB$

Table 3
Power Consumption

DMCLB1	DMCLB2	DMPGB1	DMPGB2
140uW	128uW	108uW	95uW

For a reconfigurable CLA, DMCLB1 and DMCLB2 blocks are approximated in accordance with the DA. However, the DMPGB1 and DMPGB2 blocks are approximated only when each and every DMCLB1, DMCLB2, DMPGB1, and DMPGB2 block, which belongs to the transitive fan-in cones of the concerned block, is approximated. Otherwise, the block is operated in the accurate mode. For example, any DMPGB block at the second level of CLA can be made to operate in approximate mode, if and only if, both of its constituent DMCLB1 and DMCLB2 blocks are operating in the approximate mode. Similar protocol is ensued for the blocks residing at higher levels of the tree, where each DMPGB block can be approximated only when both of its constituent DMPGB1 and DMPGB2 blocks are approximated.

For implementing 16-bit reconfigurable CLA, we have designed 5 different blocks: DMCLB1, DMCLB2, DMPGB1, DMPGB2 and Decoder. In this circuit, all blocks APP signal is connected to Decoder. Accordingly for all DMCLB1 blocks Cout is connected to DMCLB2 blocks Cin. The difference among them being that DMCLB1 & DMPGB1 produces an additional Cout signal compared with DMCLB2 & DMPGB2. The propagate and generate signal which is produced by DMCLB1 is used as input PA, GA for DMPGB1 and accordingly the propagate and generate signal which is produced by DMCLB2 is used as input PB, GB for DMPGB1 and accordingly all the blocks are connected. In this circuit, we have taken $V_{dc} = 1.8V$, the

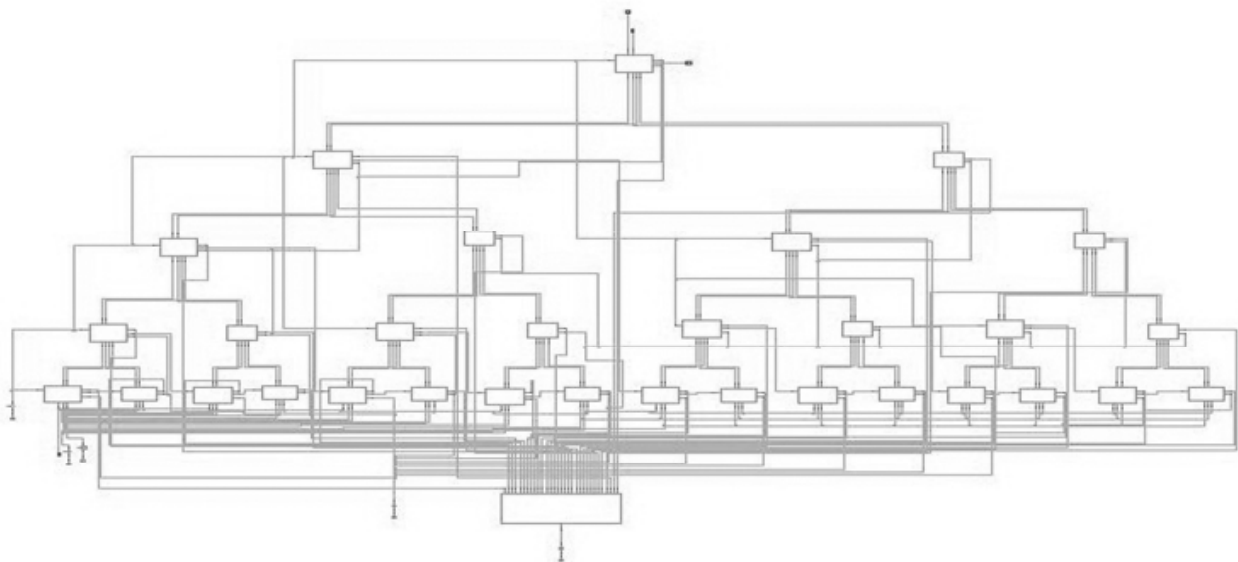


Figure 4: (a) Schematic design of 16 Bit reconfigurable Carry look ahead adder blo

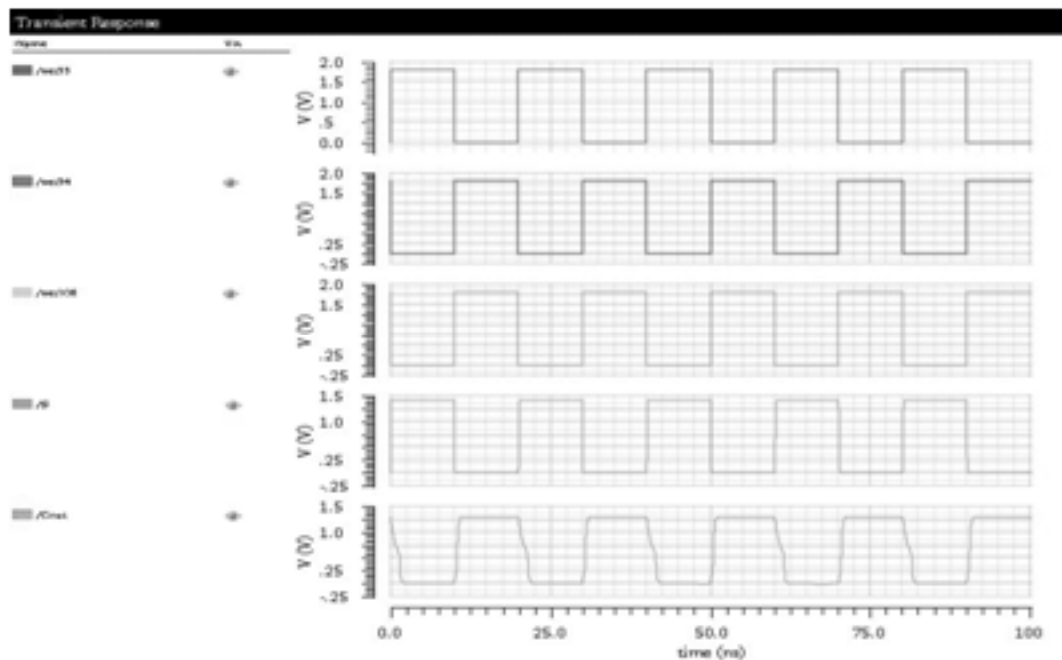


Figure 4: (b) Waveform of 16 bit reconfigurable CLA

simulation and implementation of the circuit is done in cadence virtuoso 180nm. The power for all the 4 blocks is calculated separately in given table no:III.

CONCLUSION

In this paper, we proposed approximate adders that effectively utilized to trade off power and quality for error-resilient DSP systems. Our approach aim is to simplify the complexity of a conventional full adder cell by decreasing the number of transistors and also the load capacitances. When the errors introduced by these approximations implementation were reflected at a high level in a typical DSP algorithm, the impact on output quality was very little. A decrease in the number of series connected transistors which help in reducing the effective switched capacitance and achieving voltage scaling. Our experimental results show that the proposed architecture results in power savings. The proposed approximate adders is used on top of already existing low-power techniques like SDC and ANT to extract multifold benefits with a very minimal loss in output quality. Future work include design of 16 Bit reconfigurable Carrylookahead adder block using adiabatic gates.

REFERENCES

- [1] IEEE TRANSACTIONS ON VERY LARGE SCALE INTEGRATION (VLSI) SYSTEMS Arnab Raha, Student Member, IEEE, Hrishikesh Jayakumar, Student Member, IEEE, and Vijay Raghunathan, Member, IEEE.
- [2] 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.
- [3] 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.
- [4] 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.
- [5] 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.
- [6] V. Gupta, D. Mohapatra, S. P. Park, A. Raghunathan, and K. Roy, "IMPACT: IMPrecise adders for low-power approximate computing," in *Proc. 17th IEEE/ACM Int. Symp. Low-Power Electron. Design (ISLPED)*, Aug. 2011, pp. 409–414.

- [7] 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. 124137, Jan. 2013.
- [8] S.Venkataramani, A. Sabne,V. Kozhikkottu, K. Roy, and A. Raghunathan, "SALSA: Systematiclogic synthesis of approximate circuits," in *Proc. 49th Annu.Design Autom. Conf. (DAC)*, Jun. 2012, pp. 796801.
- [9] A. Raha, H. Jayakumar, and V. Raghunathan," A power efficient video encoderusing recon figurable approximate arithmeticunits.
- [10] P. Kulkarni, P. Gupta, and M. Ercegovac, "Trading accuracy for power with an underdesigned multiplier architecture," in *Proc. 24th IEEE Int.Conf. VLSI Design*, Jan. 2011, pp. 346–351.