Low Power VLSI Architecture for Encoder and Decoder

Haraprasad Tripathy, Sushree Saraswati Panda, Priyabrata Pattanaik

Siksha 'O' Anusandhan (Deemed to be University), Odisha

Abstract- the technology is improved by a "network-on-chip (NoC)" and the additional elements are added for opposing the dissipated power in ion subsystem. The acquiring of "sample adaptive encoder architecture" has been done as in-loop filtering block. The huge quantity of samples are required for obtaining optimum AO parameters by exhaustive operations. This paper presents a low power and high speed encoder decoder at a rate of "½ convolutional coding with a constraint length of K=3. A high speed and low delay is maintained in Spartan 6 FPGA. A low power and high speed encoder decoder is designed by using different FPGA boards and using together both logics in one IC at same time by using extra area of hardware.

Keywords- Integrated Circuit (IC), Adaptive Offset (AO), Network on chip (NoC).

I. INTRODUCTION

A technique of "Forward Error Correction (FEC)" is convolutional decoding. The FEC aims to the improvement of channel capacity by addition of some redundant information to the data whose transmission will be done through channel. This is known as channel coding, as this process adds the redundant information. There are two forms of channel coding, block coding and convolutional coding[1]. Operations of convolutional codes are done on serial data, by using few bits at a time. There are message blocks of up to hundred bytes on which block codes operate. The original data is recovered by using various useful codes and algorithms that are used in decoding[2]. This paper proposes an RTL level decoder having low power and high speed in design environment of standard cell. In "standard cell design environment", VHDL is used for describing a design behaviour. A gate level design is generated by synthesis of a behavioural design. A design layout is generated by placing and routing a gate level design. The "standard cell based design" offers an advantage of faster turn-around time, circuit modelling in an accurate manner and easiness in verification of design. This paper proposes designing of decoders at RTL level[3].

II. EXISTING WORK

There is a demand of techniques that provide effective compression. The video coding has "high efficiency video coding" as a successor of next generation. The HEVC developed ISO/IEC "Moving Picture Experts Group (MPEG) and "Video Coding Experts Group (VCEG)". The HEVC reduced the encoding bit rate by 50%.

"In loop filtering block" is adopted for increasing the efficiency of computation. The Inloop filter comprise of two modules "Aging Offset filter (AO)" and "Deblocking filter (DB)". It mainly works at the elimination of processing based on blocks and quantization artefacts such as ringing, colour biases and blocking. The ringing artefacts are prevented by AO filter. Near sharp transitions and edges of objects, ringing artefacts appeared. This paper prefers a "high performance", "high throughput" and a "low cost VLSI architecture" for estimation of AO parameters. In HEVC, classification of frame is done into "Code tree units (CTUs)"[4][5]. There are three "code tree block (CTBs) components, Luma CTB(L) and two chroma CTBs(Cband Cr) plus syntax element" components in each CTU. A processor based on CTU is AO. There are three modes of operation of AO for a current CTU: a new parameter processing (New/No Merge), Assumption of upper CTU parameters (Merge Up) or assumption of parameters of left CTU (Merge Left)[6]–[8].

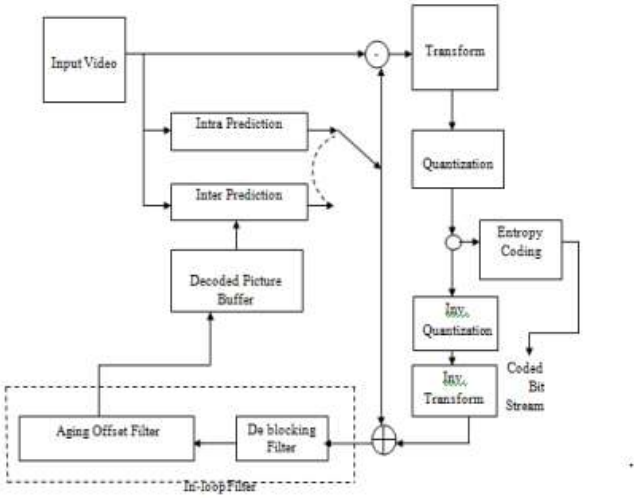

Fig.1. Simplified Block Diagram of HEVC Encoder

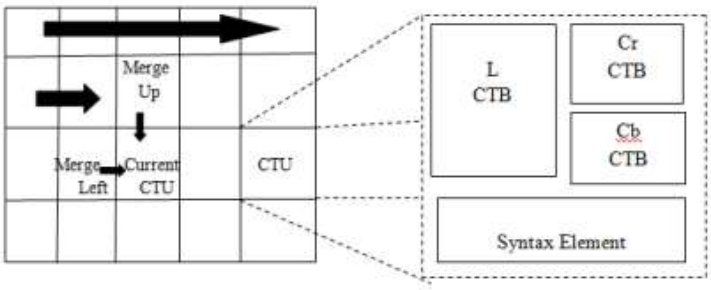

Fig.2. Merge Mode and CTU Component
Edge Offset (EO) type

There are three types in "New" mode, "Band Offset (BO) type, Edge Offset (EO) type or AO not applied (NA)". Based on the optimum type or mode, corresponding parameters are selected according to optimization of rate distortion and histogram analysis[9].

III. PROPOSED SYSTEM

The decoding process involves following tasks:

1) Computation of branch metric
2) Updating State metric: A new branch metric is used for updating a state metric.
3) Recording of survivor path: Survivor path is recorded at every node.
4) Generation of output decision: Information of survivor path is used for generating sequence of decoded output. A proposed decoder is shown in figure 3. In a quantization block, quantization and conversion analog signals into digital signal. The symbol boundaries and frame boundaries of a code word are detected by synchronization blocks. A parallel code symbols in successive manner are received by a proposed decoder, which identifies the boundaries of frames and symbols[10].

A. Branch Matrix Unit: Branch metrics are generated by this, which is a hamming distance (input data from 00, 01, 10 and 11). From an input data, branch metric are calculated for all branches of trellis by using a BM unit. Branch metric is measured by choosing absolute difference. The weights of branches are the branch metrics[11], [12].

B. ACS Unit: It is an Add Compare Unit which computes a new value of every state metrics at every instant of time. At every cycle of clock, state metrics are needed to be updated. The throughput of a system can be increased by a common approach of using pipelining, but it is not applicable. The most area and power are consumed by ACS unit. A 5 bits resolution is essential for state metrics for obtaining required precision but branch metrics need 5 bits. Positive branch metrics are added to state metrics and is a positive number then without normalizing, there would be indefinite grow in accumulated metrics[9], [13], [14].

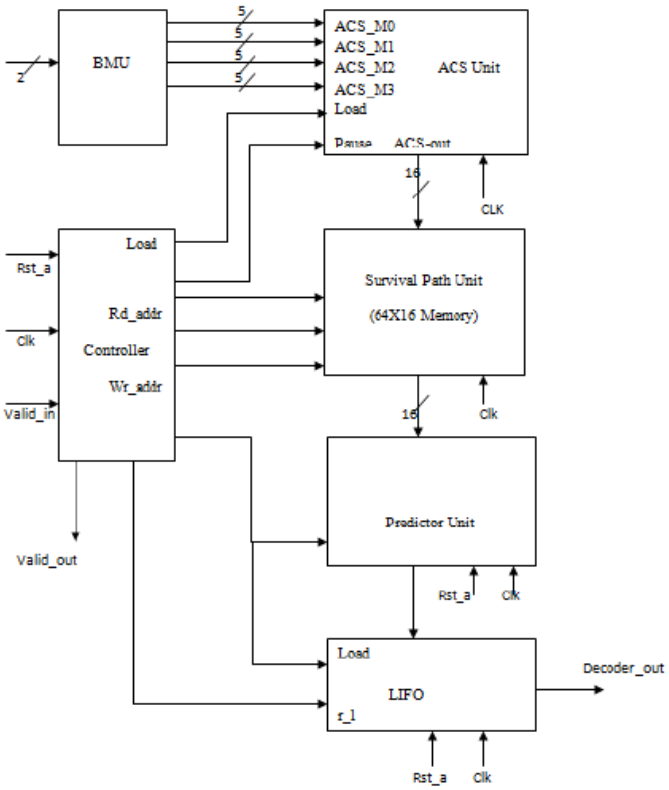

Fig. 3. Internal sub blocks of proposed decoder.

C. Memory: The survivor "Path matrix unit (PMU)" is stored in memory. The quantity of total ACS sub blocks that are used in design are used for memory word length. The length of trellis should be equal to two memory blocks. In this paper, k=5 gives length of trellis equal to 32. Hence a memory block of 64x16 and dual port is used. One post is used for writing down data and other one is used for reading data, as data is written and read simultaneously to and from various addresses.

D. Controller:

The synchronization between different modules of a system is done by a controller. The control signals such as pause, we, oe, wr_addr, rd_addr, valid_out are controlled by a decoder's controller unit. Two six bit counters are included in a controller in which one is responsible for counting up whereas another one counts down. The read and write addresses of a memory are derived by these counters. These counters are stopped by a pause signal which is generated by a controller in order to stop writing and reading of unnecessary data from memory.

E. Predictor Unit: the trellis sequence having length 32 is traced back by a predictor unit and next state is predicted and decoded after rectification of error. After every 32 clock cycles, a minimum "accumulated path metric" is loaded with a state machine. A bit is accessed from a PMMU by using this unit as a state value[15].

F. LIFO Unit: a LIFO unit is necessitated by reversing the transmitted data's order by a predictor unit. There are two registers of 32-bit each in this unit in which one is for read operation while another is for written. Both the registers are read and written alternatively by making selection by a multiplexer in a read mode and output is given in correct sequence[16].

IV. RESULT

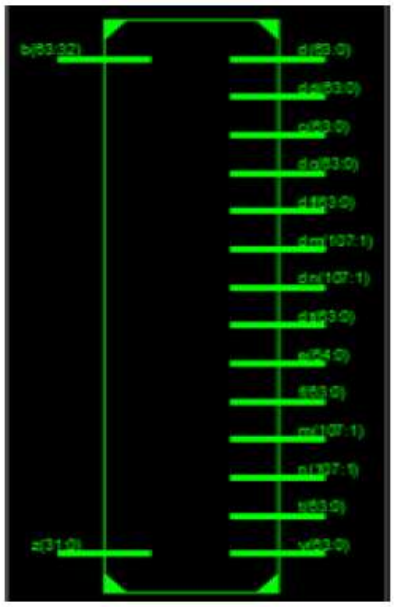

Fig. 4 RTL Schematic

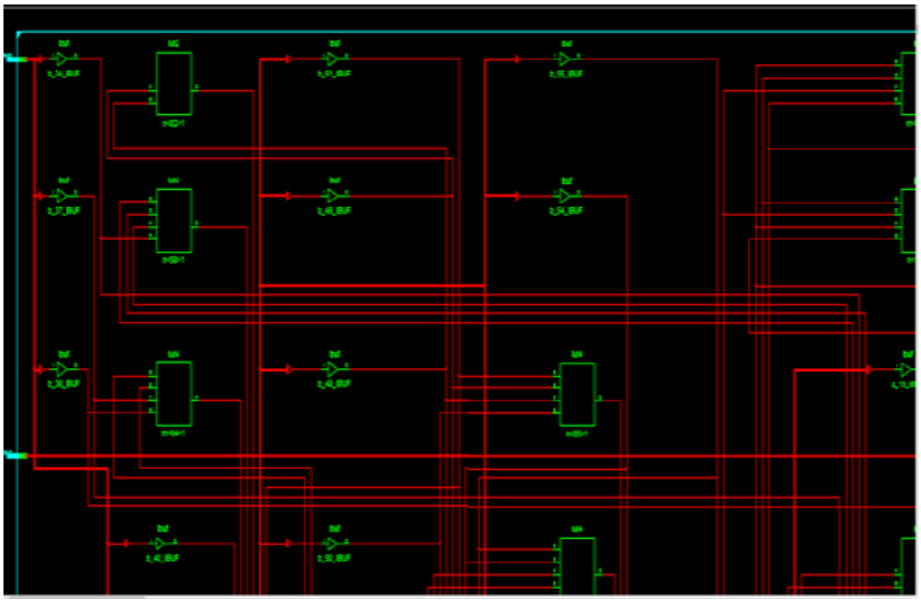

Fig. 5 Technology Schematic

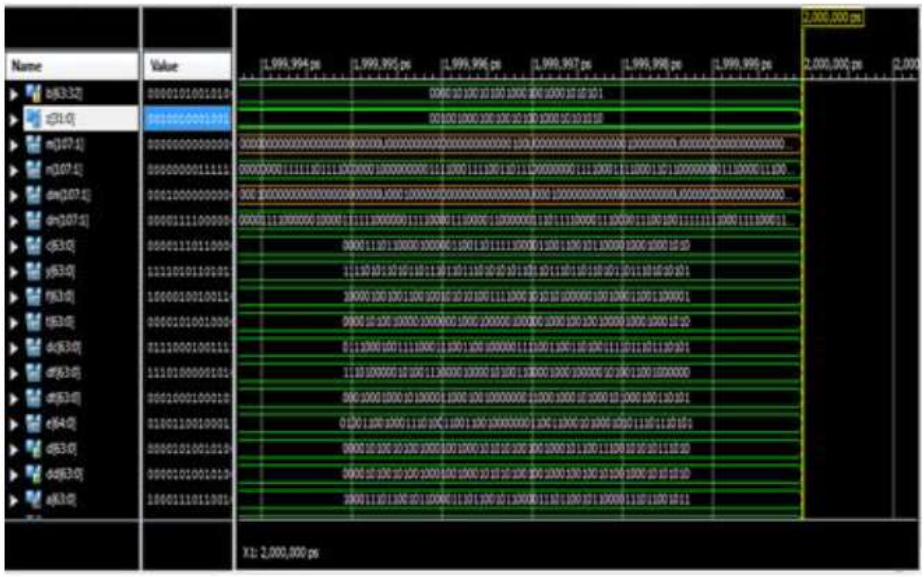

Fig. 6 Output Waveform

V. CONCLUSION

A “low cost high throughput high performance VLSI architecture” has been proposed in this paper for “SAO encoding stage”. A power consumption and high speed decoder has been designed from a proposed model of decoder and encoder and it is used for data transmission of high speed and used in communication protocols having low power and less logic utilization.

REFERENCES

[1] T. Koike-akino *et al.*, “Optimal Layered Scheduling for Hardware-Efficient Windowed Decoding of LDPC Convolutional Codes,” *Proc. Eur. Conf. Opt. Commun.*, pp. 565–567, 2016.

[2] M. Kieffer and P. Duhamel, “Joint Source-Channel Coding and Decoding,” in *Channel Coding: Theory, Algorithms, and Applications: Academic Press Library in Mobile and Wireless Communications*, 2014, pp. 535–582.

[3] G. Ma, B. Ai, F. Wang, and Z. Zhong, “Joint Design of Coded Tandem Spreading Multiple Access and Coded Slotted ALOHA for Massive Machine-type Communications,” *IEEE Trans. Ind. Informatics*, vol. 14, no. 9, pp. 4064–4071, 2018.

[4] A. Thakur and M. K. Chattopadhyay, “Design and Implementation of Viterbi Decoder Using VHDL,” in *IOP Conference Series: Materials Science and Engineering*, 2018.

[5] M. A. Abu, H. Harun, M. Y. Harmin, N. I. A. Wahab, and M. K. A. Kadir, “The design of Viterbi decoder for low power consumption space time trellis code without adder architecture using RTL model,” *World J. Eng.*, 2016.

[6] P. Singh and S. K. Vishvakarma, “Rtl level implementation of high speed-low power viterbi encoder & decoder,” in *2013 IEEE 3rd International Conference on Information Science and Technology, ICIST 2013*, 2013, pp. 345–349.

[7] M. A. Abu, H. Harun, M. Y. Harmin, and N. I. A. Wahab, “Low Complexity of ACS architecture For Space Time Trellis Code Viterbi Decoder,” *Int. J. Appl. Eng. Res.*, vol. 10, no. 4, pp. 8927–8939, 2015.

[8] M. A. Abu, H. Harun, M. Y. Harmin, and N. I. Abdul Wahab, “Power Consumption Optimization Technique in ACS for Space Time Trellis Code Viterbi Decoder,” *Appl. Mech. Mater.*, vol. 785, pp. 734–738, 2015.

[9] S. Sharma, Sunil, H. S. Vasudevamurthy, and N. Valarmathi, “Design and FPGA implementation of block synchronizer for Viterbi decoder,” in *Proceedings of the 2013 International Conference on Advances in Computing, Communications and Informatics, ICACCI 2013*, 2013, pp. 908–912.

[10] H. Yueksel, G. Cherubini, R. D. Cideciyan, A. Burg, and T. Toifl, “Design considerations on sliding-block viterbi detectors for high-speed data transmission,” in *2016, 10th International Conference on Signal Processing and Communication Systems, ICSPCS 2016 - Proceedings*, 2016.

[11] J. (2016). U. P. D. for M.-C. M. M. N. I. C. L. https://doi. org/10. 1109/LCOMM. 2016. 257169. Han, Y., & Lee *et al.*, “High-Rate and Low-Complexity Space-Time Block Codes for <inline-formula> <img src="/images/tex/24701.gif" alt="2 \times 2"> </inline-formula> MIMO Systems,” *IEEE Communications Letters*. 2016.

[12] H. Yueksel *et al.*, “Design Techniques for High-Speed Multi-Level Viterbi Detectors and Trellis-Coded-Modulation Decoders,” *IEEE Trans. Circuits Syst. I Regul. Pap.*, 2018.

[13] H. Yueksel *et al.*, “A 4.1 pJ/b 25.6 Gb/s 4-PAM reduced-state sliding-block Viterbi detector in 14 nm CMOS,” in *European Solid-State Circuits Conference*, 2016, vol. 2016-Octob, pp. 309–312.

[14] M. Véstias, H. Neto, and H. Sarmento, “Sliding block Viterbi decoders in FPGA,” in *Proceedings - 22nd International Conference on Field Programmable Logic and Applications, FPL 2012*, 2012, pp. 595–598.

[15] S. K. Tadse and S. L. Haridas, “A low power asynchronous Viterbi decoder using minimum transition hybrid register exchange method,” in *Proceedings of the 2017 International Conference On Smart Technology for Smart Nation, SmartTechCon 2017*, 2018.

[16] R. Liu and K. K. Parhi, “Low-latency low-complexity architectures for viterbi decoders,” *IEEE Trans. Circuits Syst. I Regul. Pap.*, vol. 56, no. 10, pp. 2315–2324, 2009.