Journal of **N**onlinear **A**nalysis and **O**ptimization
Vol. 10, No. 2 : 2019
ISSN : **1906-9685**

LOW COST VLSI ARCHITECTURE FOR PROPOSED ADIABATIC OFFSET ENCODER AND DECODER

[1]D.Surendra,[2]S.Priyanka,[3]C.Rudramurthy,[4]KAndula Sucharitha

[1,2,3]Assistant Professor,[4]Student
Department of ECE
Gouthami Institute Of Technology & Management For Women, Proddatur, Ysr Kadapa, A.P

ABSTRACT

A network-on-chip (NoC) improves the technology and the power dissipated starts to opposed with by the additional elements of the correspond ion subsystem. Sample adaptive encoder architecture has been acquired as a new in-loop filtering block. To get the optimum AO parameters exhaustive operations are required because of the huge amount of samples. In this paper, High speed and low power Proposed Encoder Decoder of rate ½ convolutional coding with a constraint length K = 3 is presented. A high speed it also maintain a low delay in Spartan 6 FPGA. Since the FPGA boards used are different and from that we justified that using both logics together in one Integrated Circuit (IC) we can create a high speed and low power Proposed encoder decoder at the same time with some extra hardware area.

Keywords : Network on chip[NOC], Adaptive offset[AO], Integrated Circuit [IC]

I. INTRODUCTION

Convolutional decoding is a Forward Error Correction (FEC) technique. The purpose of FEC is to improve the capacity of a channel by adding some carefully designed redundant information to the data being transmitted through the channel. The process of adding this redundant information is called as channel coding. Channel coding is in two forms i.e, Convolutional coding and block coding. Convolutional codes operate on serial data, one or a few bits at a time. Block codes operate on relatively large typically, up to a couple of hundred bytes message blocks. There are a various useful convolutional and block codes, and a various algorithms for decoding the received coded information sequences to recover the original data.

In this paper, a design of high speed low power Proposed decoder at the RTL level in the standard cell design environment is proposed. In the standard cell design environment, the behavior of a design is described in VHDL. The behavioral design is synthesized to generate a gate level design. The gatelevel design is placed and routed to generate a layout of the design. The advantages of a standard cell based design over full custom design are faster turn around time for the design, ease in design verification and more accurate modeling of the circuit. Design of Proposed decoders at the RTL-level is focused here. Proposed algorithms and implementation of Proposed decoders were investigated intensively in the past three decades.

II. EXISTED SYSTEM

The effective compression techniques are in demand. High efficiency video coding is regulated to be the next generation of video coding as a successor. The ISO/IEC Moving Picture Experts Group (MPEG) and Video Coding Experts Group (VCEG) was developed the HEVC. 50% bit rate encoding reduction is achieved by the HEVC. For increasing the compression efficiency In-loop filtering block has been adopted.

There are two successive modules i.e, Deblocking Filter (DB) and Aging Offset filter (AO) in the Inloop filter. For eliminating block-based processing and quantization artifacts, such as blocking, ringing and color biases it mainly works on it. AO filter works for preventing ringing artifacts which appeared. Obviously, ringing artifacts appeared near the object's edges or sharp transitions. High throughput, high performance, and low cost VLSI architecture for AO Parameter estimation block is preferred in this paper.

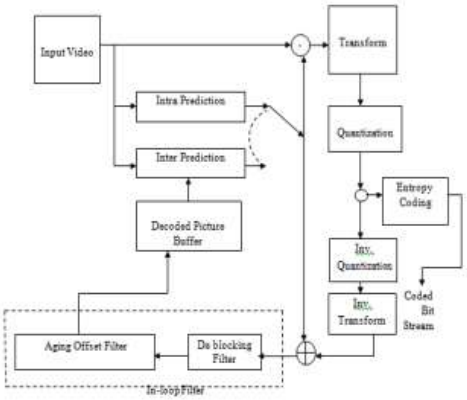


Fig.1.Simplified Block Diagram of HEVC Encoder

The frame is classified into Code tree units (CTUs) in HEVC. Each CTU contains three code tree blocks (CTBs) components, Luma CTB (L) and two chroma CTBs (Cband Cr) plus syntax element. AO is a processor based on code tree unit (CTU). AO is operated in three modes for the current CTU: processing a new parameter (New/No Merge), assumes parameters of the upper CTU (Merge Up), or assume parameters of the left CTU (Merge Left).

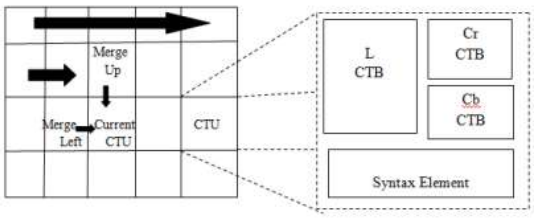


Fig.2.Merge Mode and CTU Component Edge Offset (EO) type

In "New" mode, it is one of the three types, Band Offset (BO) type, Edge Offset (EO) type or AO not applied (NA). The optimum mode/type and the corresponding parameter are selected based on histogram analysis and rate distortion optimization.

III. PROPOSED SYSTEM

The major tasks in the proposed decoding process are as follows:

1) Branch metric computation.

2) State metric update: Update the state metric using the new branch metric.

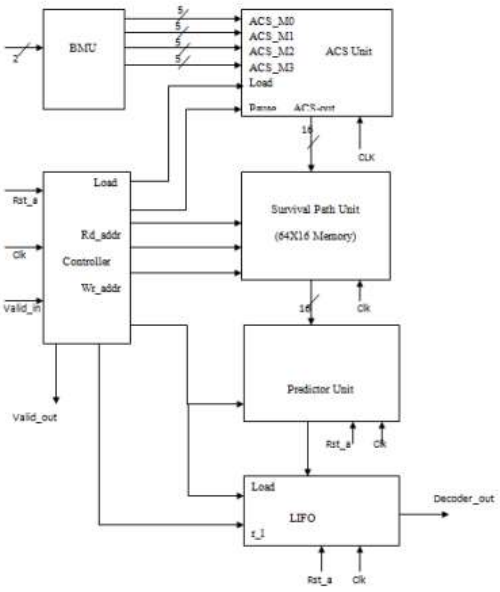
3) Survivor path recording: Tag the surviving path at each node.

4) Output decision generation: Generation of the decoded output sequence based on the survivor path information. Fig. 3 shows the proposed decoder. Analog signals are quantized and converted into digital signals in the quantization block. The synchronization block detects the frame boundaries of code words and symbol boundaries. We assume that a proposed decoder receives parallel successive code symbols, in which the boundaries of the symbols and the frames have been identified.

A. Branch Matrix Unit: It is used to generate branch metrics, which are hamming distances of input data from 00, 01, 10 and 11. The BM unit is used to calculate branch metric for all trellis branches from the input data. We choose absolute difference as measure for branch metric. These branch metrics are viewed as being the weights of the branches.

B.ACS (Add Compare Select) Unit: A new value of the state metrics has to be computed at each time instant. The state metrics have to be updated every clock cycle. Because of this recursion, pipelining, a common approach is to increase the throughput of the system which is not applicable. The AddCompare-Select (ACS) unit is the module that consumes the most power and area. In order to obtain the required precision, a resolution of 5 bits for the state metrics is essential, while 5 bits are needed for the branch metrics. Since the state metrics are always positive numbers and only positive branch metrics are added to them, the accumulated metrics would grow indefinitely without normalization.

Fig. 3. Internal sub blocks of proposed decoder.



C. Memory: Memory is required to store the survivor Path Matrix Unit (PMU). The word length of the memory depends on the number of the ACS subblocks used in the design or the total number of states in the decoder or k^2 (where k is the constraint length, 5 in our case), and the depth of the memory depends on the trellis length. The memory depth usually should be kept two times the trellis length or two blocks of memory equal to trellis length. We have for our project k = 5 and trellis length equal to 32, so the memory block used is 64x16. The memory used is dual port. One port for writing the data and other for reading the data, as we need to write and read the data simultaneously and that to from different addresses.

D. Controller: A controller is used to synchronize between the different modules of the system. The controller unit of decoder controls signals like we, pause, valid_out, oe, rd_addr, wr_addr. The controller also includes two six bit counters of which one counts up another counts down. These counters drive the write and read addresses of the memory. The pause signal generated by the controller also stops these counters for a while so that no unnecessary data is written onto the memory or read from it.

E. Predictor Unit: Predictor unit is used to trace back the trellis sequence of length 32 and predict the next state and actual decoded bit after rectifying the error. This unit is a state machine that is loaded with the state with minimum accumulated path metric after every 32 clock cycles. This unit uses the state value to access a bit from the path metric memory unit (PMM) or memory unit.

F. LIFO Unit: Every 32 decoded bits put out by the predictor unit is in reverse order of the transmitted data, this necessitates a LIFO unit. This unit has 2 32- bit registers

in which one of the register is read while the other is written. The two 32- bit registers are read and write alternatively and simultaneously selected by the multiplexer which is in the read mode and gives the output in correct sequence.

IV. RESULTS

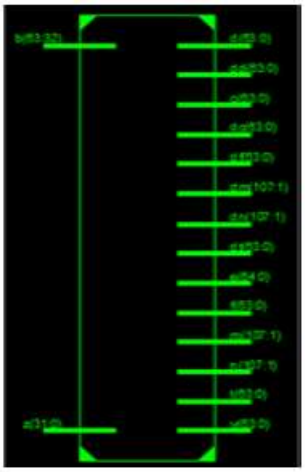


Fig.4 RTL Schematic

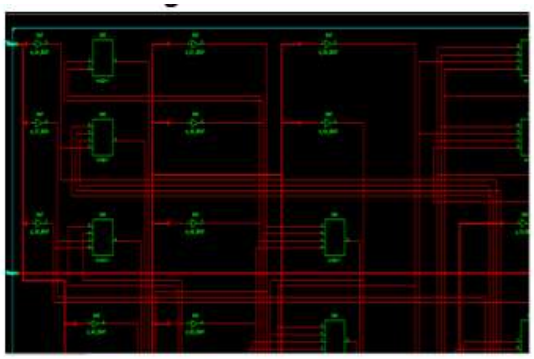


Fig.5 Technology Schematic

Fig.6 Output waveform

V. CONCLUSION

This paper presents a low cost high throughput high performance VLSI architecture for SAO encoding stage. Hence from our proposed encoder decoder we designed a high speed and power consumption decoder and which can be used for communication protocols like CDMA for high speed data transmission with nearly low power with less amount extra logic utilization.

VI. REFERENCES

[1]. Tomas, V., "Decoding of Convolutional Codes Over the Erasure Channel," IEEE Trans. on Information Theory, vol.58, no.1, pp. 90-108, Jan. 2012.

[2]. Viterbi AJ. , "Error bounds for convolutional codes and an asymptotically optimum decoding algorithm," IEEE Transactions on Information Theory, Vol. 13, no. 2, pp. 260-269, April 1967.

[3]. Bupesh Pandita and Subir K Roy. , "Design and Implementation of Viterbi Decoder Using FPGAs," VLSID '99 Proceedings of the 12th International Conference on VLSI Design, page 611, 1999.

[4]. P.J. Black and T.H. Meng., "A 1-Gb/s, four-state, sliding block Viterbi decoder," IEEE Journal of Solid-State Circuits, Vol. 32 no.6, pp.797-805, 1997.

[5]. M. Boo, F. Arguello, JD Bruguera, R. Doallo, and EL Zapata., "High-performance VLSI architecture for the Viterbi algorithm," IEEE Trans. on communications, Vol. 45, no.2 pp.168- 176, 1997.

[6]. O. Collins and F. Pollara., "Memory management in traceback Viterbi decoders," TDA Prog. Rep, pages 42-99, 1989.

[7]. G. Fettweis and H. Meyr., "Parallel Viterbi decoding by breaking the compare select feedback bottleneck," Communications, 201:88, 1988.

[8]. G. Feygin and P. Gulak., "Architectural tradeoffs for survivor sequence memory management in Viterbi decoders," IEEE Transactions on Communications, Vol. 41, no.3, pp. 425- 429, 1993.