© 2017 IJSRST | Volume 3 | Issue 4 | Print ISSN: 2395-6011 | Online ISSN: 2395-602X
Themed Section: Science and Technology

VLSI Architecture of Configurable Low Complexity Hard Decision Viterbi Decoder

Prashant Shirke
Electronics Department, G. H. Raisoni College of Engineering, Nagpur, Maharashtra, India

ABSTRACT

A Viterbi algorithm has served as a powerful method for decoding of the convolutional code so as to control errors in data transmission over a noisy channel. It is based on maximum likelihood algorithm for decoding the data. However, the hardware implementation of Viterbi algorithm become crucial as it consumes large resources due to its complexity. This paper discusses the implementation of an efficient VHDL implementation of a Viterbi decoder using the concept of pipelining to reduce the critical path (maximum combinational path delay), thereby improving the operating frequency of the design and improving the throughput using ModelSim and Xilinx ISE tools for simulation and synthesis of modules respectively.

Keywords : Error correction codes; Maximum likelihood decoding; Convolutional codes; Viterbi algorithm; VHDL.

I. INTRODUCTION

This Viterbi decoder is used for error correction in satellite and deep space communication. The Viterbi algorithm is known to be an efficient method for the realization of maximum- likelihood (ML) decoding of the convolutional codes. Viterbi decoder is one of the efficient methods of decoding of convolutional codes [1]. To transmit data over a noisy channel, convolutional code offers much better results in contrast to block codes. Some extra bit is added from encoder side for reducing the probability of error of the signal. The convolution coding technique can take continuous stream of bits to encode. In addition to this, convolution code also gives efficient hard decisions.

Viterbi decoder mainly comprises of four basic units [2]: Branch Metric Unit (BMU), Add Compare Select Unit (ACSU), Survivor Path Memory Unit (SPMU), and Trace Back Unit (TBU). The function of BMU is to determine the branch metric, which is the hamming distance between the received signal and expected signal. ACSU calculates the decisions bit that is used to determine the survivor path and store in SPMU. TBU is used to trace the original bit that was given to the encoder input [3]. There are many mechanisms for implementation of high speed and area efficient Viterbi decoder.

Trellis coded modulation (TCM) schemes are used in many bandwidth-efficient systems. Typically, a TCM system employs a high-rate convolutional code, which leads to a high complexity of the Viterbi decoder (VD) for the TCM decoder, even if the constraint length of the convolutional code is moderate. For example, the rate-3/4 convolutional code used in a 4-dimensional TCM system for deep space communications [1] has a constraint length of 7; however, the computational complexity of the corresponding VD is equivalent to that of a VD for a rate-1/2 convolutional code with a constraint length of 9 due to the large number of transitions in the trellis. Therefore, in terms of power consumption, the Viterbi decoder is the dominant module in a TCM decoder. In order to reduce the computational complexity as well as the power consumption, low-power schemes should be exploited for the VD in a TCM decoder.

Error correcting code is a method to find the errors sequentially and correct it based on the remaining constraints such as bit length etc. This study is so called as coding theory. Error detection is the simple method that is checking of digits, applications are credit card

number mistakes. It is also called as the block codes. These are mainly used in the CD players and mobile phones. The sum of the digits given the module of some number.

In the coding theory forward error correction is a method used for controlling errors in the data exchange in noisy channels. The main theme is the sender encode his message using error correcting codes. The receiver receives the encoded message and find the errors in the message avoiding the retransmission. This gives the ability to avoid resending the data in the reverse channel in fixed band width. This technique is applied where the situations are about to retransmission is more cost effective and one way communication. The information which is sending is stored in the storage devices to reveal the corrupted data. It uses digital bit stream with the modular carrier. It is used in multicast communications also.

FEC processing in a receiver may be applied to a digital bit stream or in the demodulation of a digitally modulated carrier. It is an integral part of the initial analog- to-digital conversion. Most of the researchers uses bit error rate to find the bit errors and it depends on the length of the message. Viterbi decoder implements a soft-decision algorithm to demodulate digital data from an analog signal corrupted by noise.

Forward Error Correction codes are divided into two types. Block codes work on fixed-size blocks (packets) of bits or symbols of predetermined size. Practical block codes can decoded in polynomial time to their block length.

Convolutional codes work on bit or symbol streams of arbitrary length. They are decoded with the Viterbi algorithm and other algorithms are used. Viterbi decoding allows asymptotically optimal decoding efficiency with increasing constraint length of the convolutional code and at the expense of exponentially increasing complexity. It can be turned into a block code, if desired, by "tail-biting".

II. METHODS AND MATERIAL

1. Related Work

(1) High-speed Parallel Viterbi Decoding: Algorithm and VLSI-Architecture, Gerhard Fettweis, Heinrich Meyr, May 199 - IEEE Communications Magazine

The Viterbi algorithm is only one specific example of a large set of algorithms which is of interest to design parallel process- ing architectures. Therefore, it is important to derive metho- dologies and tools of how to introduce additional parallelism into algorithms. The basic tools applied here were CS- arithmetic at the bit-level, semi-ring algebra at the word-level, and exploiting acquisition properties at the algorithm-level. We believe that the generalizations of all three methods of finding solutions at each level are a step in this direction [9]. However, a lot of work remains to be done.

(2) High-Speed Low-Power Viterbi Decoder Design for TCM Decoders, Jinjin He, Huaping Liu, Zhongfeng Wang, Xinming Huang, and Kai Zhang, IEEE Trans. VLSI Syst., vol. 18, no. 5, pp. 808-817, May 2010.

The authors have proposed a high-speed low-power VD design for TCM systems. The pre-computation architecture that incorporates T-algorithm efficiently reduces the power consumption of VDs without reducing the decoding speed appreciably. We have also analyzed the pre-computation algorithm, where the optimal pre-computation steps are calculated and discussed. This algorithm is suitable for TCM systems which always employ high-rate convolutional codes. Finally, we presented a design case. Both the ACSU and SMU are modified to correctly decode the signal. ASIC synthesis and power estimation results show that, compared with the full-trellis VD without a low-power scheme, the pre-computation VD could reduce the power consumption by 70% with only 11% reduction of the maximum decoding speed.

(3) High Speed Architecture Design Of Viterbi Decoder Using Verilog HDL, Karri Megha Shyam, International Journal of Engineering Trends and Technology (IJETT) – Volume 4 Issue 10 - Oct 2013

The main purpose of this study is to yield the gains obtained by the developers with the usage of Viterbi algorithm. Research mainly centers on the grandness of Viterbi algorithm in the practical applications with the

VHDL code. Research not only helps the students related to the communications but it also helps the people who are in the field of decoders as it is one of the efficient method for reducing the errors while communication procedure is in advance. Here, VHDL code is used in order to implement the Viterbi algorithm in a proper way. Apart from various codes, researcher selected VHDL code for this research as it offers the high capability in designing the electronic systems. Students and the business people and one can easily understand and analyze the Viterbi algorithm concepts and can gain more knowledge on the VHDL code and the tools that are used in this research.

(4) High- speed- Low-Power Viterbi Decoder Design, V.D.M. Jabez Daniel, International Journal on Recent and Innovation Trends in Computing and Communication 2014 Volume: 2 Issue: 3

High-speed, low-power design of Viterbi decoders for trellis coded modulation (TCM) systems is presented in this paper. It is well known that the Viterbi decoder (VD) is the dominant module determining the overall power consumption of TCM decoders. We propose a pre-computation architecture incorporated with - algorithm for VD, which can effectively reduce the power consumption without degrading the decoding speed much. A general solution to derive the optimal pre-computation steps is also given in the paper. Implementation result of a VD for a rate-3/4 convolutional code used in a TCM system shows that compared with the full trellis VD, the precomputation architecture reduces the power consumption.

(5) Performance Evaluation of 802.11ah Viterbi Decoder for IoT Applications, Thi Hong Tran, Hiromasa Kato, Shinya Takamaeda-Yamazaki, Yasuhiko Nakashima, 2015 International Conference on Advanced Technologies for Communications (ATC)

In this paper the authors have exposed our first research results on Viterbi decoder for IoT application. They have built a 802.11ah simulator, run the simulation and evaluated the system perfor- mance in relation with Viterbi decoder's parameters such as trace-back length L, input data bit-width D, and LLR truncated value E. Simulation results have shown that both BER and PER performance are improved if L value increased. However, if L is large enough the performance improvement becomes insignificant. Selecting L in the range from 20 to 40 is our recommendation. In addition, they have shown that if D value increases from 1 bit to 4 bits, both BER and PER performance are improved. However, as they continue to increase D value, both BER and PER performance are almost unimproved. Especially, when D increases from 2 to 3, they see the best improvement of both BER performance (i.e., 1.8 dB) and PER performance (i.e., 1.5 dB). They suggest to use D = 3 for hardware development. Finally, they have shown that if using D = 3 bits, we should truncate the LLR value around E = 1 75. Doing so we can achieve the best BER and PER performance without any hardware trade-off.

(6) Performance Analysis of Turbo Decoder using Soft Output Viterbi Algorithm, Shweta Ramteke, Sandeep Kakde, Y ogesh Suryawanshi, Mangesh Meshram, Student Member, IEEE ICCSP 2015 conference

This paper presents the design and implementation of an efficient VLSI architecture for 3GPP-LET. Turbo decoder mainly consists of soft-input soft-output (SISO) decoders to achieve high throughput and interleaver and deinterleaver. Turbo decoder comprises of Branch Metric Unit(BMU), State Metric unit(SMU), Log-Likelihood Ratio Computation Unit(LLR), Add Compare Select Unit. Throughput of Turbo decoder depends on speed of ACS (Add Compare & Select) This paper, proposed a new ACS(Add Compare Select) unit consists of Carry-Iookahed Adder, Digital Comparator & Multiplexer which increases the throughput & reduces the area of the design. In data transmission, turbo coding helps achieve near Shannon limit performance. Turbo coding is an advanced error correction technique widely used in the communications industry. Turbo encoders and decoders are important blocks in today's communication systems to achieve the best possible data reception with the fewest possible errors. The proposed turbo decoder is based on the Soft Output Viterbi Algorithm (SOV A). For low bit error rate the Turbo Decoder Simulated using MATLAB. The entire architecture of Turbo decoder is coded using Verilog HDL and it is synthesized using Xilinx EDA with Spartan 3E.

(7) Real-Time Area Efficient and High Speed Architecture Design of Viterbi Decoder, Amitava Middya, Anindya S. Dhar, International Conference on Advances in Electrical, Electronics, Information, Communication and Bio-Informatics (AEEICB16)

An area efficient and high speed architecture design of hard decision Viterbi decoder with encoding rate of 1/2 and constraint length of k = 3 is presented in this paper for the application in satellite communication. The proposed Viterbi decoder is implemented in field programmable gate array (FPGA) and also in application specific integrated circuit (ASIC) using UMC 0.18 µm technology where the maximum frequency of operation is 423.566 MHz in FPGA and 427 MHz in ASIC. The proposed architecture takes 112 Look Up Tables in FPGA and 23842 µm2 in ASIC. The proposed architecture takes less area as well as offers high speed.

III. PROPOSED WORK

In this work we have proposed to design an efficient VHDL implementation of a Viterbi decoder using the concept of pipelining to reduce the critical path (maximum combinational path delay), thereby improving the operating frequency of the design and improving the throughput.

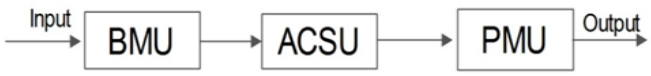

Figure 1. Block diagram of Viterbi decoder

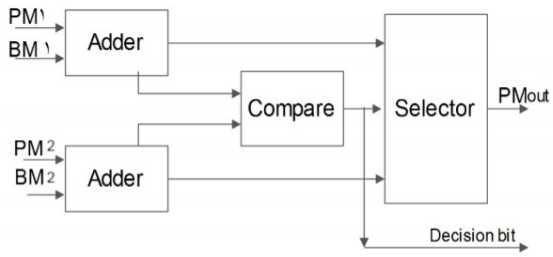

Fig. 2 : Architecture of ACS

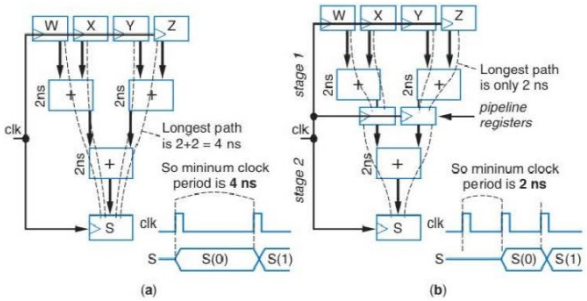

Figure 3. Proposed Pipelining

Pipelining a design is achieved by adding pipeline registers to combinational paths in order to reduce the critical path delay. In the design above, the nonpipelined design has a critical path delay of 4 ns (figure a). When pipeline registers are added to the combinational paths, the critical path delay is reduced from 4 ns to 2 ns. This allows the pipelined design to run twice as fast as the non pipelined design, thus improved performance by a factor of 2. However, the faster clock will result in higher power dissipation.

While critical path reduction is the outcome of pipelining, there is an increase in latency as pipeline depth increases. In the example above, the pipeline depth is 1, so a latency of 1 clock cycle is introduced at output S.

IV. RESULT AND DISCUSSION

Pipelining a design is achieved by adding pipeline registers to combinational paths in order to reduce the critical path delay. In the design above, the nonpipelined design has a critical path delay of 4 ns (figure3 a). When pipeline registers are added to the combinational paths, the critical path delay is reduced from 4 ns to 2 ns. This allows the pipelined design to run twice as fast as the non pipelined design, thus improved performance by a factor of 2. However, the faster clock will result in higher power dissipation.

While critical path reduction is the outcome of pipelining, there is an increase in latency as pipeline depth increases. In the example above, the pipeline depth is 1, so a latency of 1 clock cycle is introduced at output S. Another drawback of pipelining is the increase in area. Very deep pipelines can cause an increase in area by up to 20%.

The non pipelined Viterbi decoder has an area utilization of about 21% whereas the pipelined Viterbi decoder has an area utilization of about 35% (please refer to the area reports).

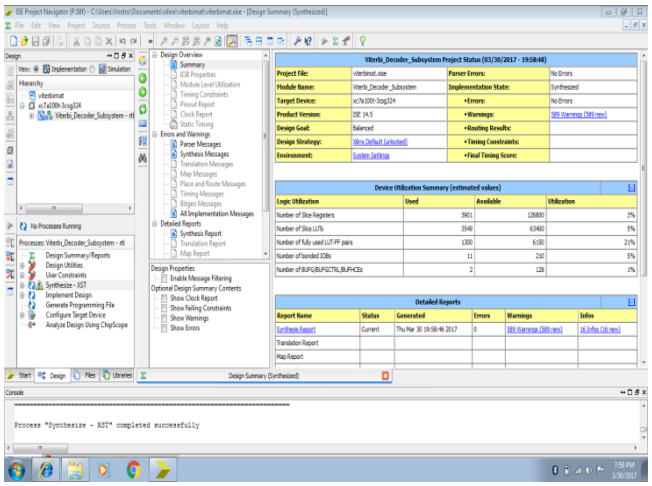

Figure 4. Nonpipelined design area report

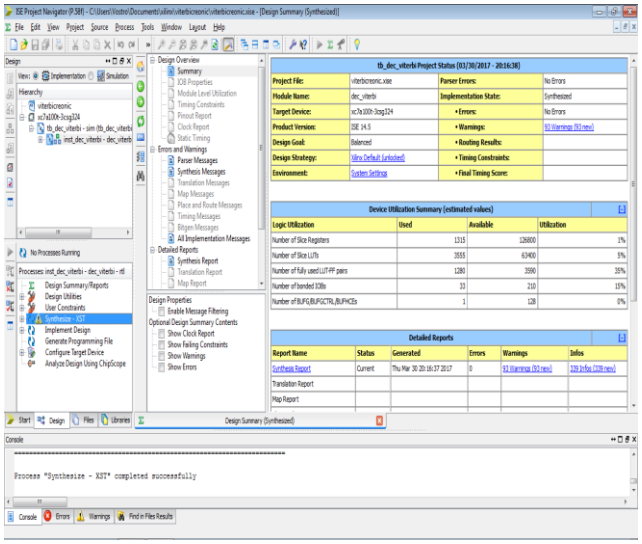

Figure 5. Pipelined design area report

Because of critical path reduction, the pipelined decoder is able to operate at 274 MHz as opposed to 127 MHz for the nonpipelined decoder (please refer to the timing reports).

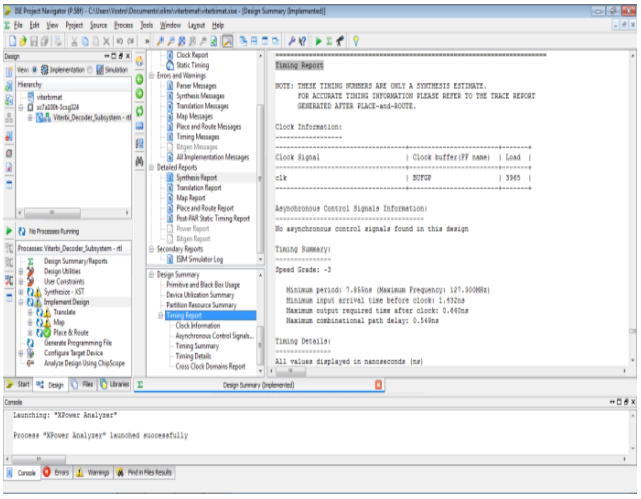

Figure 6. Non pipelined design timing report

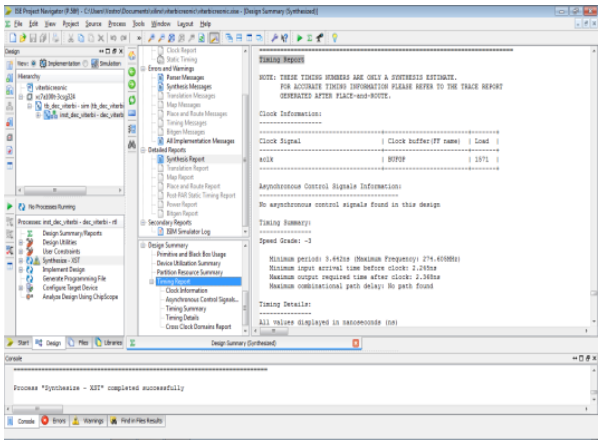

Figure 7. Pipelined design timing report

Also, because of the faster operating clock frequency, the pipelined decoder dissipates 157 mW compared to the nonpipelined design, which dissipates only 149 mW (please refer to the power reports).

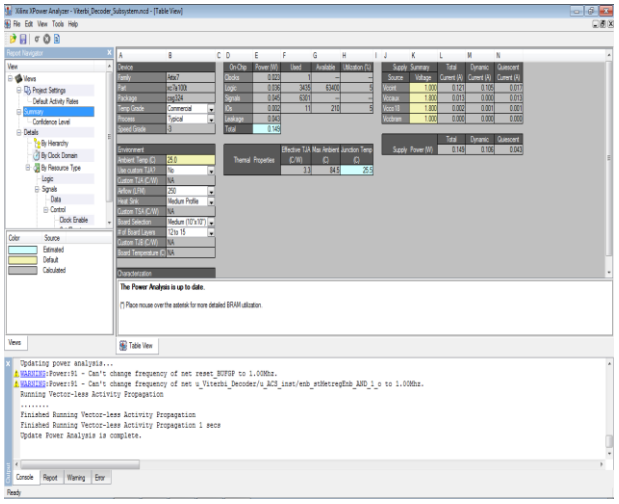

Figure 8. Nonpipelined design power report

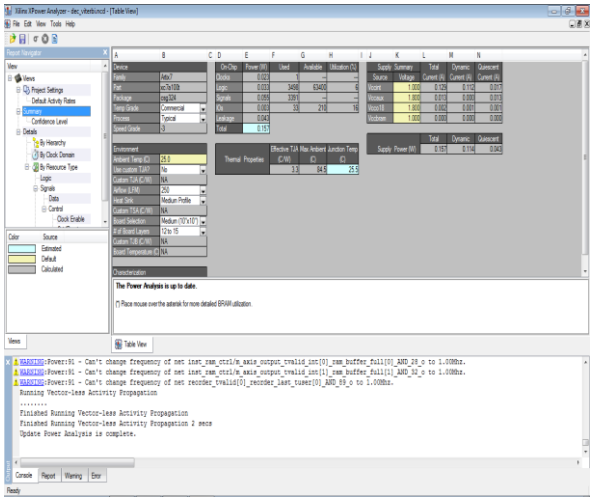

Figure 9. Pipelined design power report

Increased latency is shown in the simulation results, where latency stands at 5000 ps for the pipelined decoder, and at 2000 ps for the nonpipelined design (please refer to the simulation waveforms).

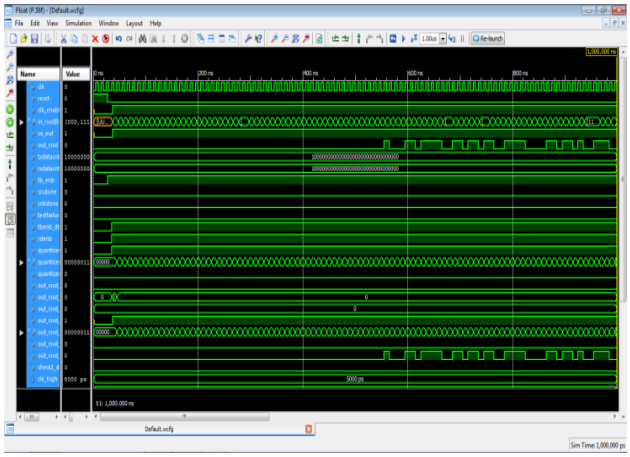

Figure 10. Nonpipelined design simulation results

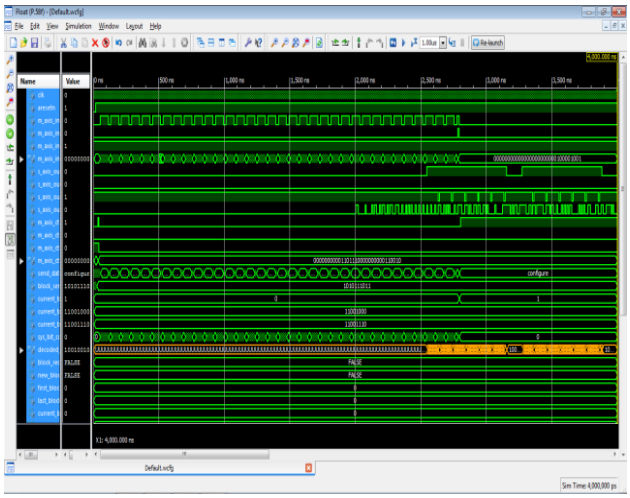

Figure 11. Pipelined design simulation results

The RTL view shows the presence of pipelined registers in the critical path, in case of the modified design, and these registers are absent in the original design (please refer to the RTL views).

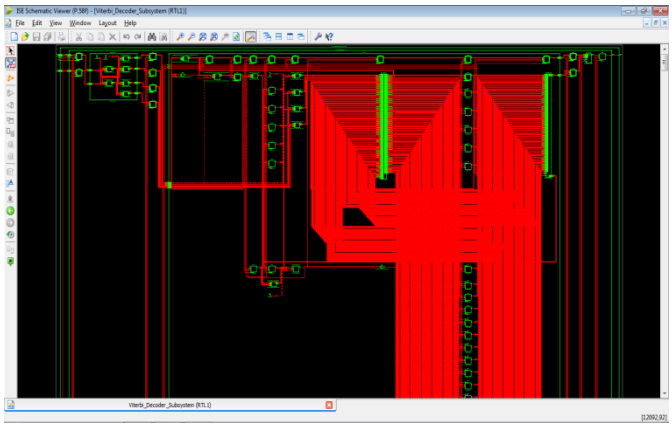

Figure 5. Nonpipelined design RTL view

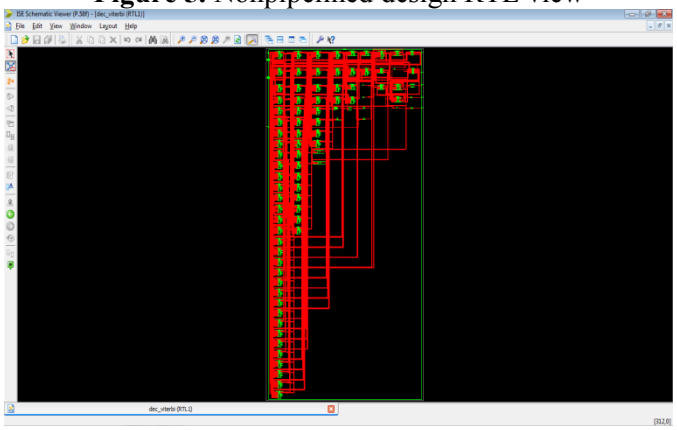

Figure 13. Pipelined design RTL view

V. CONCLUSION

Thus we have been able to design an efficient VHDL implementation of a Viterbi decoder using the concept of pipelining to reduce the critical path (maximum combinational path delay), thereby improving the operating frequency of the design and improving the throughput.

VI.REFERENCES

[1] A. J. Viterbi, "Convolutional codes and their performance in communication systems," IEEE Transactions on Communication Technology, vol. COM-19, pp. 751-772, Oct. 1971.

[2] P. Singh, S. Vishvakarma, "RTL level implementation of high speed- low power viterbi

encoder & decoder," Proc. IEEE ICIST 2013 pp. 345-347, March 2013.

[3] G. Ungerboeck , "Trellis coded modulation with redundant single set part I : Introduction" IEEE Journal of communication Magazine, vol 25,no 2, pp.5-11, February 1987.

[4] M. Veshala, T. Padmaja, K. Ghanta, "FPGA based design and implementation of modified viterbi decoder for a wi-fi receiver ," Proc. IEEE ICT 2013, pp. 525-529, April 2013.

[5] D. Chakraborty, P. Raha, A. Bhattacharya, R dutta, "Speed optimization of a FPGA based modified Viterbi Decoder," Proc. IEEE ICCCI 2013, pp. 1-6, January 2013.

[6] Kelvin Yi- Tse Lai, "An efficient metric normalization architecture for high-speed low-power viterbi decoder" Proc. IEEE TENCON 2007.

[7] Shu Lin and Costello, "Convolutional Codes", in Error Control Coding , 2nd ed, India: Dorling Kindersley Pvt. Ltd., 2011, pp. 454-485.

[8] H-D Lin, "Architectures for Viterbi Decoding and ML Estimators for Controllable Markov Sources," Master's thesis, University of California at Berkeley, May. 1988.

[9] G. Ungerboeck, "Adaptive maximum likelihood receiver for carrier modulated data-transmission systems," IEEE Trans. Communications, Vol. 22, no. 5, pp. 626-636, May. 1974.

[10] C. B. Shung, P. H. Siegel, G. Ungerboeck, and H. K. Thaper, "VLSI architectures for metric normalization in the Viterbi algorithm," IEEE ICC, Vol. 4, pp. 1723-1728, Apr. 1990.