Reconfigurable Efficient Design of Viterbi Decoder for Wireless Communication Systems

Swati Gupta[1]

Faculty Member, ECE Department
D.I.E.T
Karnal, India-132001
swatigupta13@gmail.com

Rajesh Mehra[2]

Faculty Member, ECE Department
NITTTR
Chandigarh, India-160019
rajeshmehra@yahoo.com

Abstract—**Viterbi Decoders are employed in digital wireless communication systems to decode the convolution codes which are the forward correction codes. These decoders are quite complex and dissipate large amount of power. With the proliferation of battery powered devices such as cellular phones and laptop computers, power dissipation, along with speed and area, is a major concern in VLSI design. In this paper, a low power and high speed viterbi decoder has been designed. The proposed design has been designed using Matlab, synthesized using Xilinx Synthesis Tool and implemented on Xilinx Virtex-II Pro based XC2vpx30 FPGA device. The results show that the proposed design can operate at an estimated frequency of 62.6 MHz by consuming fewer resources on target device.**

Keywords—*Clock Gating; FPGA; VHDL; Trace Back; Viterbi Decoder.*

I. INTRODUCTION

Viterbi Decoder has been recognized as an attractive solution to a variety of digital estimation problems, as the Kalman filter has been adapted to analog estimation problems. Viterbi algorithm is widely used in many wireless and mobile communication systems for optimal decoding of convolutional codes. Convolutional codes which are forward error correction codes offer a good alternative to block codes for transmission over a noisy channel. The purpose of forward error correction (FEC) is to improve the capacity of a channel by adding some carefully designed redundant information to the data being transmitted through the channel [1]. The Viterbi algorithm essentially performs maximum likelihood decoding to correct the errors in received data which are caused by the channel noise. However it reduces the computational load by taking advantage of special structure in the code trellis. Moreover viterbi decoding has a fixed decoding time which is well suited for hardware decoder implementation [2]. The requirements for the Viterbi decoder, which is a processor that implements the Viterbi algorithm, depend on the application in which it is used. This results in a very wide range of data throughput. The decoder structure is very simple for short constraint length, making the decoding feasible at rates of up to 100 Mbit/s. Viterbi decoder is effective in achieving noise tolerance, but the cost is an exponential growth in memory, computational resources and power consumption.

II. VITERBI DECODER AND ALGORITHM

The Viterbi algorithm is commonly used in a wide range of communication and data storage applications. It is also used for decoding convolutional codes, in base band detection for wireless systems, and for detection of recorded data in magnetic disk drives. The Viterbi detectors used in cellular telephones have low data rates (typically less than 1Mb/s) and should have very low energy consumption. On the opposite end of the scale, very high speed Viterbi detectors are used in magnetic disk drive read channels, with throughputs over 600Mb/s but power consumption is not as critical

Viterbi Maximum Likelihood Algorithm is one of the best techniques for communications, especially wireless where energy efficiency is the most important factor. It works on the principle of selecting a code word closest to the received word. The Viterbi decoder examines an entire sequence of received signal of a given length. The decoder computes a metric for each path and makes a decision based on this metric. The metric is hamming distance between the received branch word and expected branch word [4]. This is just the dot product between the received codeword and the allowable codeword. All paths are followed until two paths converge on one node. Then the path with the lower metric is kept and the one with higher metric is discarded. The paths selected are called the survivors. For an N bit sequence, total numbers of possible received sequences are 2^N. The Viterbi algorithm applies the maximum-likelihood principles to limit the comparison to 2 to the power of kL surviving paths instead of checking all the paths. The selection of survivors lies at the heart of the Viterbi algorithm and ensures that the algorithm terminates with the maximum likelihood path. The algorithm terminates when all of the nodes in the trellis have been labeled and their entering survivors are determined. We then go to the last node in the trellis and trace-back through the trellis. At any given node, we can only continue backward on a path that survived upon entry into that node. Since each node has only one entering survivor, our trace-back operation always yields a unique path. This path is the maximum likelihood estimate that predicts the most likely transmitted sequence. The maximum likelihood is given by:

$$P(Z/U(m')) = \max P(Z/U(m)) \ over \ all \ U(m) \quad (1)$$

where Z is the received sequence, and U (m) is one of the possible transmitted sequences, and chooses the maximum (closest possible received sequence).

III. ARCHITECTURE OF THE VITERBI DECODER

The input to the communication systems is a stream of analog, modulated signals. The primary task of the receiver is the recovery of the carrier signal and also synchronization of bit timing so that the individual received data bits can be removed from the carrier and also separated from one another in an efficient manner. Both tasks are generally performed through the use of phase locked loops [5]. The analog base band signal is applied to the analog-to-digital converter with b-bit quantizer to get a received bit stream. The bit stream is then applied as the input to the Viterbi decoder. In order to compute the branch metrics at any given point in time, the Viterbi decoder must be able to segment the received bit stream into n-bit blocks, each block corresponding to a stage in the trellis.
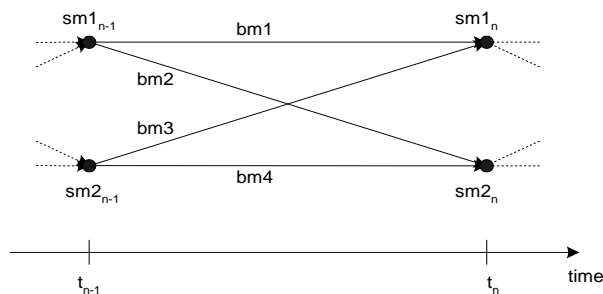

Fig.1 Two State Trellis

A trellis diagram is a time-indexed version of a state machine, and the simplest 2-state trellis is shown in Fig.1. Each state in the trellis corresponds to a possible pattern of recently received data bits and each branch corresponds to a receipt of the next (noisy) input. The goal is to find the path through the trellis of maximum likelihood because that path corresponds to the most likely pattern that the transmitter actually sent [8]. In this paper, we assume that the input to our proposed design is an identified code symbols and frames.

The basic building blocks of viterbi decoder are:-

A. Branch Metric Unit

The branch metric unit (BMU) takes the fuzzy bit and calculates the cost for each branch of the trellis. A simple branch metric unit may use hamming or Euclidean distance as the metric for calculating the cost of the branch [7]. It is based on a look-up table containing the various bit metrics. The computer looks up the n-bit metrics associated with each branch and sums them to obtain the branch metric.
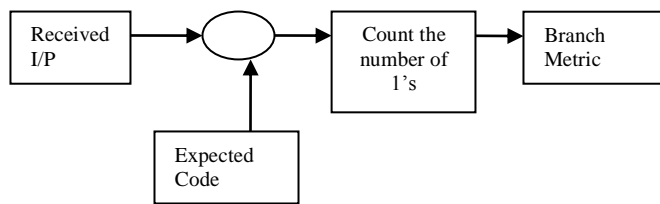

Fig.2 Branch Metric Computer

B. Add-Compare-Select Unit

The add-compare-select unit (ACSU) is the heart of the Viterbi algorithm and calculates the state metrics. It recursively accumulates the branch metrics as the path metrics (PM), compares the incoming path metrics, and makes a decision to select the most likely state transitions for each state of the trellis and generates the corresponding decision bits. The path metrics are added to state metrics from the previous time instant and the smaller sum is selected as the new state metric:

$$sm1_n = \min(sm1_{n-1} + bm_1, sm2_{n-1} + bm_3) \qquad (2)$$

$$sm2_n = \min(sm1_{n-1} + bm_2, sm2_{n-1} + bm_4) \qquad (3)$$

bm_k is the hamming distance between received and expected sequence.

For a given code with rate 1/n and total memory M, the number of ACS required to decode a received sequence of length L is $L \times 2^M$.
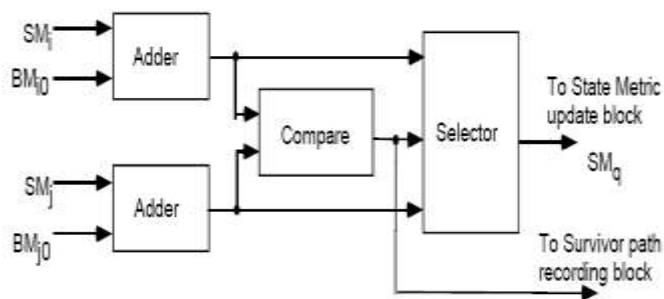

Fig.3 ACS Module

C. Survivor Memory Unit

The survivor memory unit (SMU) is responsible for keeping track of the information bits associated with the surviving paths designated by the path metric updating and storage unit. There are two basic design approaches for SMU:

Register Exchange and Trace Back. In both techniques, a shift register is associated with every trellis node throughout the decoding operation. This register has a length equal to the frame length. The register exchange method works well for small constraint lengths. The traceback method works well for longer constraint length codes. The traceback method stores the decisions from the ACS into a RAM and also the path information in the form of an array of recursive pointers [9]. The best path is determined by reading backwards through the RAM. The general approach to traceback is to accumulate path metrics for up to five times the constraint length ($5 * (K - 1)$), find the node with the largest accumulated cost, and begin traceback from this node [15]. The trace-back unit can then output the sequence of branches used to get to that state. In practice, the survivor paths merge after some number of iterations. The trellis depth at which all the survivor paths merge with high probability is referred to as the survivor path length.

IV. PROPOSED LOW POWER DESIGN

The ACSU and SMU consume most of the power of the decoder. In this paper we will be focusing on Survivor Memory Unit of viterbi decoder to develop a low power model. Among the two memory organization technique in the SMU, i.e. register exchange and trace back, the trace back approach is being used for low power applications. In the traceback approach, each register storing the survivor path information updates its content only once during the entire period of a code word. In contrast, all the registers in the register-exchange approach update their contents for each code symbol. Hence, the switching activity of the registers in a traceback approach is much lower than that for the registers in a register-exchange approach. So low power design techniques can be applied readily to the traceback module. In our work we will be utilizing the benefit of clock gating to develop a low power design [2]. The key issue is that the content of each register does not change as soon as it is updated. This is very useful in our low power design, as we don't have to activate the registers after each updation which reduces the switching activity leading to a reduction in power dissipation. Some blocks of a circuit are used only during a certain period of time. The clock of these blocks can be disabled to eliminate unnecessary switching not in use. Fig. 4 shows clock gating to disable a unit.
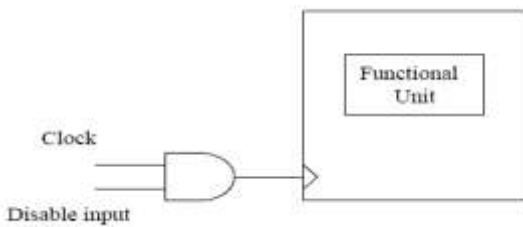

Fig.4 Clock Gating

The survivor path storage block holds the information on survivor paths. When the ith code symbol is received, the survivor path information is obtained and stored in the ith register. At this moment all other registers hold their contents,

and hence their clocks can be gated to save power as shown in Fig. 5. The clock is gated by the information coming from a ring counter that tells what the current state is so far.
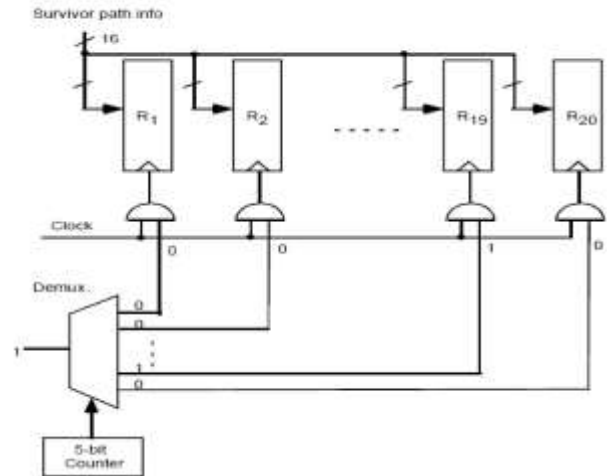

Fig.5 Implementation of Clock Gating

V. HARDWARE IMPLEMENTATION

A Matlab code is initially written for the convolutional encoder with constraint length, K= 7 and rate ½ and the two generator polynomials G1G2 as {171,133} and our proposed viterbi decoder design using trace back with clock gating to evaluate the performance of the proposed design. Fig.6 shows the BER curve vs. Eb /No using the AWGN channel for both the uncoded data and the coded data using convolutional coding with viterbi decoding. The data is decoded every clock cycle but delayed 20 clocks. Since the traceback module is not activated until the end of the frame and only for one clock cycle, this feature helps in saving the power.
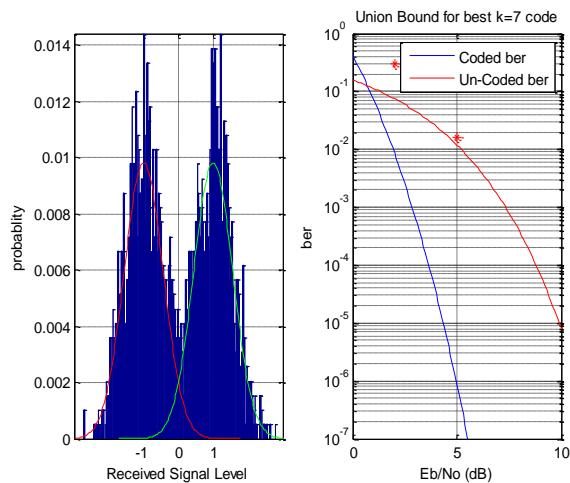

Fig.6 Matlab Simulation for Viterbi Decoder

Next step was the development of VHDL code. The test bench is written for both the convolutional encoder and viterbi decoder using ModelSim SE 6.4b simulator to test the functionality of the implemented decoder. Fig shows the

simulated results after applying an error pattern to show the efficiency of the decoder in correcting those errors.
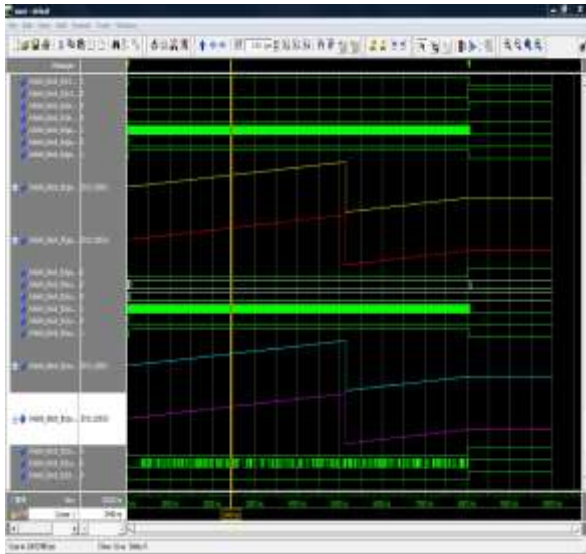

Fig7. ModelSim Simulation for Decoder

Xilinx ISE 10.1 tool has been used to map the design to FPGA Xilinx Virtex-II Pro xc2vpx70 with speed grade -5. The proposed design using the clock gating is then applied to Xilinx Xpower analyzer tool. Table 1 shows the device utilization summary

TABLE 1: DEVICE UTILIZATION SUMMARY

Device Utilization Summary		
Logic Utilization	Used/Available	Utilization
Number of Slices	5687/33088	17%
Number of Slice Flip-flops	2433/66176	3%
Number of 4 input LUTs	10686/66176	16%
Number of IOBs	17/996	1%
Number of MULT 18x18s	4/328	1%

TABLE 2: COMPARISON OF POWER DISSIPATION AND SPEED

	Trace back with Shift Register	Trace back with Shift Update and clock gating [6]	Proposed Design
Speed (In MHz.)	------	47	62.52
Power Consumed (In Watt)	0.246	0.068	0.103

The results in Table 2 shows how the speed is being enhanced and the power is being reduced for the proposed design using traceback with clock gating as compared to other conventional designs.

VI. CONCLUSION

Features like flexibility, re-configurability and shorter time to market provides for a wide range of applications for FPGA. In this paper, a high speed and low power viterbi decoder has been designed which benefits from the concept of clock gating, switching off the blocks when not in use and hence helping in power saving. The design has been described using VHDL and implemented on VirtexII Pro based xc2vpx70 FPGA using ISE10.1. The power analysis has been done using Xilinx Xpower analyzer tool. The overall design shows that the effective speed of operation increases by 24.8% and a reduction in power dissipation to about 45% as compared to the design which was not benefiting clock gating and was using the conventional design using shift registers.

REFERENCES

[1] Yeu-Horng Shiau, Pei-Yin Chen, Hung-Yu Yang, Yi-Ming Lin, Shi-Gi Huang, "An efficient VLSI architecture for convolutional code decoding", International Symposium on Next Generation Electronics, 2010, pp. 223-226

[2] Sherif Welsen Shaker, Salwa Hussien Elramly and Khaled Ali Shehata, "FPGA Implementation of a Configurable Viterbi Decoder for Software Radio Receiver", Autotestcon, IEEE, July 2009, pp. 140 – 144.

[3] Jinjin He, Huaping Liu, Zhongfeng Wang, "A fast ACSU architecture for Viterbi decoder using T-algorithm", Forty-Third Asilomar Conference on signals, Systems and Computers, 2009, pp. 231-235.

[4] Abdallah R.A., Seok-Jun Leey, Goel M., Shanbhag N.R., "Low-power pre-decoding based viterbi decoder for tail-biting convolutional codes", IEEE Workshop on Signal Processing Systems, 2009, pp. 185-190

[5] Abdallah R.A., Shanbhag N.R., "Error-Resilient Low-Power Viterbi Decoder Architectures", IEEE Transactions on Signal Processing, Vol. 57, 2009, pp. 4906-4917

[6] Sherif Welsen Shaker, Salwa Hussien Elramly, and Khaled Ali Shehata, "Design and Implementation of Low-Power Viterbi Decoder for Software-Defined WiMAX Receiver", International Conference on Microelectronics, pp. 264 – 267, 2009.

[7] Adam O., Shengli Fu, Varanasi M, "Hardware Efficient Encryption Encoder and Decoder Unit", Military Communication Conference, IEEE, 16-19 Nov. 2008, pp. 1 –6

[8] Wei Shao, Brackenbury L., "Pre-processing of convolutional codes for reducing decoding power consumption", IEEE International Conference on Acoustics, Speech and Signal Processing, 2008 , pp. 2957 - 2960

[9] Ajay D. Jadhav, Anil. R. Yardi, "Folding ADC Design for Software Defined GSM Radio-Mobile Station Receiver", IJCSNS International Journal of Computer Science and Network Security, vol. 8 No.9, September 2008.

[10] Ying Li, Weijun Lu, Dunshan Yu, Xing Zhang, "Design Technique of Viterbi Decoder in Satellite Communication", IET Conference on Wireless, Mobile and Sensor Networks, 2007, pp. 162-165.

[11] Sun F., Zhang T, "Low-Power State-Parallel Relaxed Adaptive Viterbi Decoder", IEEE Transactions on Circuits and Systems, Vol. 54, Issue. 5, 2007, pp. 1060-1068

[12] Noguera J, Kennedy I.O, "Power Reduction in Network Equipment through Adaptive Partial Reconfiguration", International Conference on Field Programmable Logic and Applications, 2007 , pp. 240 - 245

[13] Guichang Zhong, Willson, A.N., "An Energy-efficient Reconfigurable Viterbi Decoder on a Programmable Multiprocessor", International Symposium on Circuits and Systems, IEEE, May 2007, pp. 1565-1568.

[14] Dong-Sun Kim D.S., Seung-Yerl Lee, S.Y., Kyu-Yeul Wang, K.Y., Jong-Hee Hwang, J.H., Duck-Jin Chung, D.J., "Power Efficient Viterbi Decoder based on Pre-computation Technique for Portable Digital Multimedia Broadcasting Receiver", IEEE Transactions on Consumer Electronics, Vol. 53, Issue. 2, 2007, pp. 350-356

[15] Russell Tessier, Sriram Swaminathan, Ramaswamy Ramaswamy, Dennis Goeckel, and Wayne Burleson, "A Reconfigurable, Power- Efficient

Adaptive Viterbi Decoder", IEEE Transactions on VLSI Systems, Vol.13, No.4, April 2005.

[16] Yao Gang, Ahmet T.Erdogan and Tughrul Arslan, "An effective Pre-trace back architecture for the viterbi decoder Targeting Wireless Communication Applications", IEEE Transactions on Circuits and Systems, Vol.53, No.9, pp. 725-729, Sep 2006.

AUTHORS PROFILE

Ms. **Swati Gupta** is currently Assistant Professor at Doon Valley Institute of Engineering and Technology, Karnal, India. She is pursuing M.E. from NITTTR, Panjab University, Chandigarh, India. Ms. Swati has completed her B.E. (Hons.) from Vaish College of Engineering, Rohtak, India. She has 9 years of academic experience. Ms. Swati has authored 3 review papers and 2 research papers in reputed National Conferences. Her areas of interest include Communication Systems, Wireless and Mobile Communication and Digital Signal Processing.

Mr. Rajesh Mehra is currently Assistant Professor at National Institute of Technical Teachers' Training & Research, Chandigarh, India. He is pursuing his PhD from Panjab University, Chandigarh, India. He has completed his M.E. from NITTTR, Chandigarh, India and B.Tech. from NIT, Jalandhar, India. Mr. Mehra has 15 years of academic experience. He has authored more than 25 research papers in reputed International Journals and 35 research papers in National and International conferences. Mr. Mehra's interest areas include VLSI Design, Embedded System Design, Advanced Digital Signal Processing, Wireless & Mobile Communication and Digital System Design. Mr. Mehra is life member of ISTE.