

# Performance Analysis of Convolutional Encoder and Viterbi Decoder Using FPGA

Shaina Suresh, Ch. Kranthi Rekha, Faisal Sani Bala

Musaliar College of Engineering, Talla Padmavathy College of Engineering, Linton University College

**Abstract**— Wireless devices such as hand phones and broadband modems rely heavily on forward error correction techniques for their proper functioning, thus sending and receiving information with minimal or no error, while utilizing the available bandwidth. Major requirements for modern digital wireless communication systems include high throughput, low power consumption and physical size. This paper presents the design of an efficient coding technique for wireless communication, using FPGA, a four state convolutional encoder and decoder were designed respectively, an efficient decoder with high speed and low power consumption was designed using the memory less decoder design, the maximum frequency of the device clock was recorded as 185 MHz, with a coding gain of 4dB over uncoded BPSK for BER of  $10^{-10}$  found through MATLAB simulation. The encoder and decoder were implemented on altera DE1 board, with cyclone II FPGA having a maximum frequency of 50MHz.

**Index Terms**— Bit Error Rate, Convolutional Encoder, FPGA, Speed, Viterbi Decoder.

## I. INTRODUCTION

The ever increasing use of wireless digital communication has led to a lot of effort invested in FEC (Forward Error Correction). Wireless digital communication devices such as hand phones and broadband modems rely heavily on forward error correction techniques for their proper functioning, thus sending and receiving information with minimal or no error, while utilizing the available bandwidth. The FEC is computationally intensive, and it is traditionally done using digital signal processors or application specific integrated circuits (ASICs). The convolution encoding with Viterbi decoding, which provides optimum decoding [5] is the most common FEC used in wireless digital communication devices. Thus these encoding and decoding techniques have been implemented using digital signal processors and ASICs. However, in recent years, study of DSP platform performance has revealed that for higher level system performance, while considering budget constraints, DSP processors are not the best options for implementing digital communication algorithms; cost and performance issues regarding DSP processors have been discussed by [6]. Major requirements for modern digital wireless communication systems include high throughput, low power consumption and physical size.

### A. Convolutional encoder

The convolutional encoder maps a continuous information bit stream into a continuous bit stream of encoder output. The convolutional encoder is a finite state machine, which is a

machine having memory of past inputs and also having a finite number of different states it will encounter [1].

The convolutional encoder is characterized by certain parameters, which are used for design, analysis and specification of the encoder.

- 1) Input bit stream: This is the number of data bits entering the encoder input, represented as 'k' [1].
- 2) Output bit stream: This is the total number of bits of a given code word coming out of the encoder, the symbolic representation is 'n' [1].
- 3) Constraint length: The constraint length of the convolutional encoder represents the number of stages which the input bit goes through in the encoding shift register, this is represented as K [1].
- 4) Code rate: This is the ratio of the input bit to output bit, given as k/n.
- 5) The convolutional encoder has several kinds of representations, namely schematic diagram representation, polynomial representation (connection representation), state diagram representation (Finite State Mealy Machine), tree diagram and trellis diagram.
- 6) The aim of this work is to design and implement an efficient convolutional encoder and viterbi decoder for wireless communication, using FPGA (Field Programmable Gate Array).

## II. PROPOSED METHOD

### A. Encoder

A Convolutional encoder introduces redundant bits into the data stream through the use of linear shift registers. Convolutional codes are most commonly specified by three parameters; n, k, K. The information bits are input into shift registers and the output encoded bits are obtained by modulo-2 addition of the input information bits and the contents of the shift registers where two outputs are multiplexed to give the single output encoded bits.

The proposed encoder has the following specification below and schematic in Figure 1.

- Constraint length: K=3
- Quantization level: 8 level (Soft decision)
- Input bit : n=1
- Output bit: k=3
- Generator polynomials:

$$G_1 = 1 + X + X^2 \quad (1)$$

$$G_2 = 1 + X + X^2 \quad (2)$$

$$G_3 = 1 + X^2 \quad (3)$$

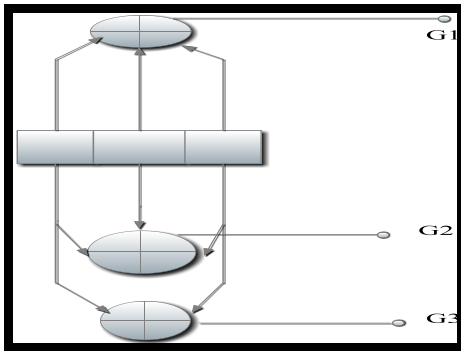


Fig1: Proposed Convolutional Encoder

**B. Viterbi decoder**

An efficient Viterbi decoder must have consideration for both speed and power dissipation, as such a lot of work has been reported in literatures, where architectures to achieve either high speed or low power dissipation were discussed. Some of the methods in literatures for low power dissipation are,

**1) Clock Gating**

This technique of low power design is implemented at the gate level. In clock gating, clock of the blocks in the circuit which are only used for certain periods are disabled, thus getting rid of unnecessary switching. This powerful technique is mostly used in the trace back circuit of the Viterbi decoder. [3] Reported an FPGA implementation of a reconfigurable Viterbi decoder for wimax receiver using clock gating power reduction technique. Another work was also reported by [4], clock gating of the traceback memory was employed as the power saving scheme, for both the two designs, and there is a substantial decrease in power consumption of the decoders. However, the drawback of clock gating is the price paid for additional hardware, thus increasing the size of the circuit.

**2) Toggle Filtering**

When the inputs of a combinational circuit block are delayed relative to one another, the circuit switches continuously before properly having all the inputs, thus consuming dynamic power before a valid output can be obtained. To address the problem, early input signals are disabled until all other input signals are ready. Toggle filtering and clock gating have been applied jointly for low power design as reported by [2]. However, the system design when different schemes are put together can become complex. On the other hand, the speed of the Viterbi decoder is primarily determined by the Add Compare Select Unit (ACSU), which is responsible for the most intensive computations in the decoder and by the type of trace back scheme used to record the decision bits before producing the output. In terms of the memory type for the decoded sequence generation, trace back memory consumes less power, however the first output bit can only be ready when the decoder is deep into the trellis, while the register exchange is faster, on the other hand, the power dissipation of it is far more than that of the trace back, because it requires copying contents of state registers during each stage of the decoding.

Thus the reason why register exchange is used for decoders with small constraint length. Considering the ACSU unit, its architecture and configuration is critical to the device speed. The sharing of hardware resource in Viterbi design is widely used in order to reduce amount of logic elements required, however, such practice slows the speed of computation by the device. Among the methods used to improve the speed of the ACSU or the decoder is pipelining [8]. Also, parallel architecture has been used extensively to speed up the computation in ACSU, as reported by [4]. However, routing the connection between the ACSU processors, which are of equal number with the states in the trellis is the drawback of the parallel ACSU processing, as it increases the use of logic and register resources and increases the routing overhead. Nonetheless synthesis tools for FPGA design can take care of the routings and connections.

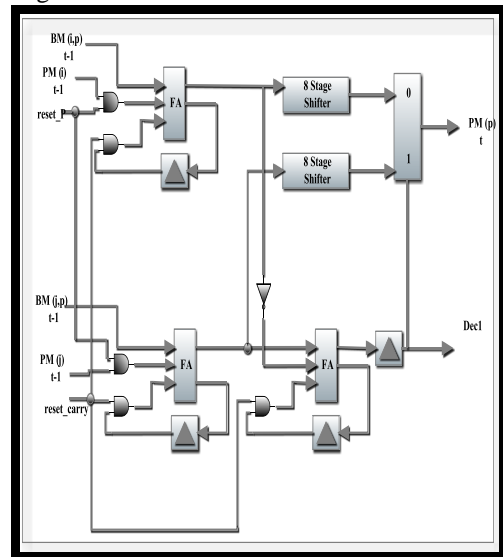


Fig 2. Bit Serial ACSU

The design proposed in this work is the Memory less Viterbi Decoder [5]. This design has extreme low power consumption and low latency. The design consists of namely the ACSU block, Add Compare Select to Survivor Memory (ACSTOSM), Parallel to Serial Converter, Branch Metric Unit (BMU), Most Significant Bit (MSB), and Pointer. The low power design is based on the pointer concept applied to a register exchange, with decoder having knowledge of its initial state, only corresponding row is necessary for decoding, using the fact that register exchange provides decoded bit in the right order, even the row memory is not of any use, thus requiring reset of the encoder only after every survivor path length. The BMU of the design is in five bit two's complement format, while the ACSU is having a bit serial architecture initially proposed by [11]. However further modification of the architecture was done [12] thus making it faster and compatible with the memory less architecture, thus having a parallel to serial converter between the ACSU and BMU. Each state has an ACSU butterfly dedicated for it, therefore making parallel computation possible, there by speeding up the decoder. Figure 3 shows proposed design of the decoder.

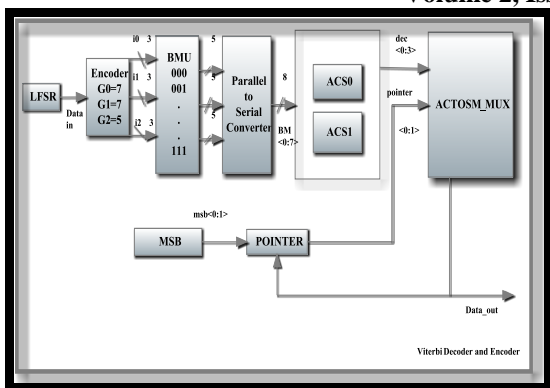


Fig 3: Proposed Decoder

C. Design Flow

Considering the butterfly diagram [13] in Figure 4, which is a signal flow for the Viterbi decoder, derived from the trellis diagram, connections of the path metric and branch metric to the ACSU unit can be established.

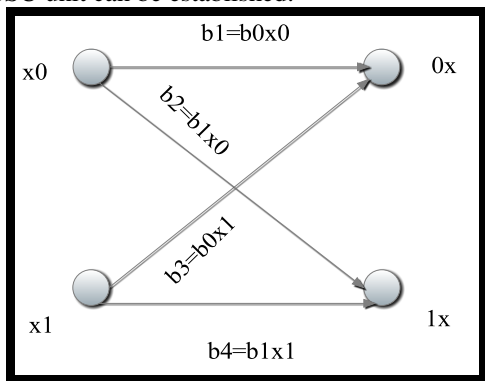


Fig 4. Butterfly Diagram

Since the encoder has four states, 'x', in Figure 4, carries value of 0 and 1, thus resulting in the butterfly diagrams in Figure 5 and Figure 6 respectively. From Figure 4(a), the initial states of the encoder are 00 and 01, while the next states are 00 and 10; transition from 00 to 00 has a branch metric value of b000, while transition from 00 and 10 has a branch metric value of b100. On the other hand, transition from 01 to 00 has the branch metric of b001 and from 01 and 10 has the Branch metric of 101. The same transition sequence holds for Figure 4(b).

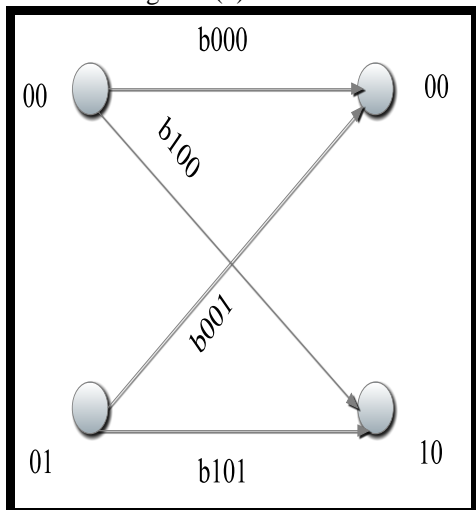


Fig 4(a). Butterfly diagram for x=0

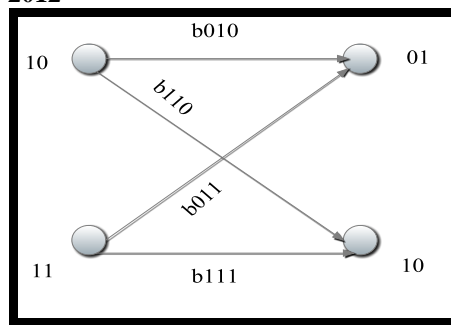


Fig 4(b). Butterfly Diagram for x=1

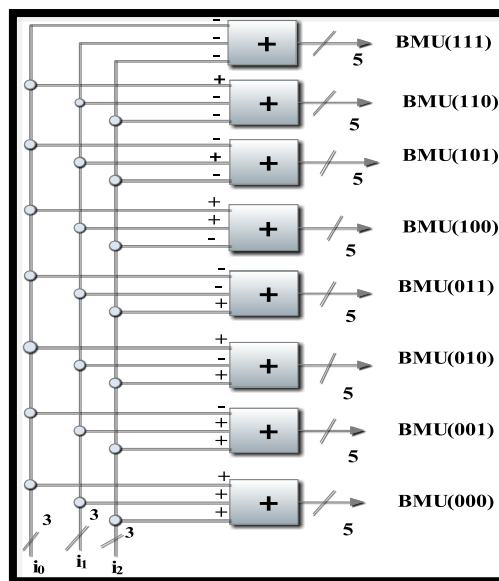


Fig 5. Branch Metric Unit

The Branch metric has symmetric property [13] which can be used to reduce the routing or wiring of all the branch metrics to the ACSU. The Branch Metric Unit architecture is given in Figure 5 and its RTL view is given in Figure 8. With reference to Figure 4., the expression is given as,

$$b_1 = b_2' = b_3' = b_4$$

For x = 0

$$BMU(000) = BMU(100)' = BMU(001)' = BMU(101) \quad (4)$$

$$\therefore BMU(000) = BMU(011) = BMU(110) = BMU(101) \quad (5)$$

For x = 1

$$BMU(010) = BMU(110)' = BMU(011)' = BMU(111) \quad (6)$$

$$\therefore BMU(010) = BMU(001) = BMU(100) = BMU(111) \quad (7)$$

From the derivation given in equations 5,6,7 and 8, it can be seen that only four Branch Metrics are required for the path metric computation, namely BMU (000), BMU (111), BMU (100) and BMU (011). The Add Compare Select to Survivor Memory (ACSTOSM), shown in Figure 7 and RTL view shown in Figure 10 is a four to one multiplexer, which routes the decision bits of the ACS module as required to the output. The select signal for the multiplexer is the output of pointer block. From the derivation given in equations 5,6,7 and 8, it can be seen that only four Branch Metrics are required for the path metric computation, namely BMU (000), BMU (111), BMU (100) and BMU (011). The Add

Compare Select to Survivor Memory (ACSTOSM), shown in Figure 8 and RTL view shown in Figure 11 is a four to one multiplexer, which routes the decision bits of the ACS module as required to the output. The select signal for the multiplexer is the output of pointer block.

A. Quartus II Simulations

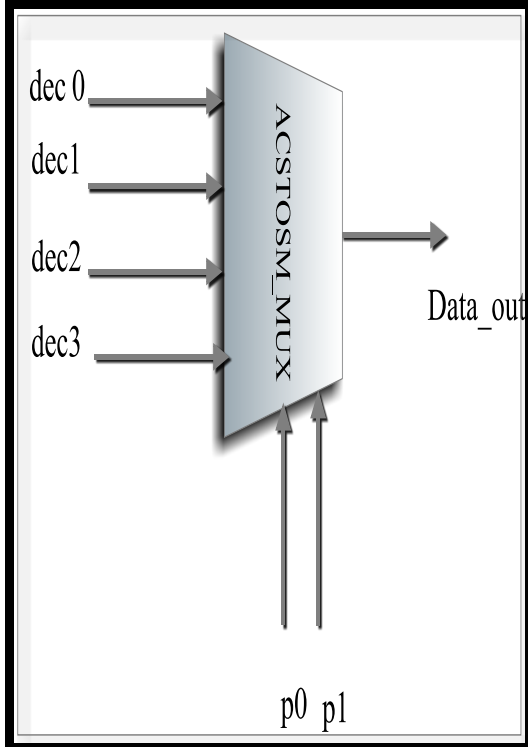


Fig 7. Add Compare Select to Survivor Memory (ACSTOSM)

The pointer block, RTL view of which is shown in Figure 9 keeps track of the decoded bit state, which is recorded as two bits, thus its content is updated for each bit decoded. The last block is the MSB block, RTL view shown in Figure 11, which has two outputs that point to the two bits of the pointer in parallel. The second bit of the MSB block is enabled at reset; this causes the pointer's second bit to be the MSB. After accessing incoming code word in the ACSU, a decision bit is written into the MSB of the pointer, and the output of the MSB block is shifted to the right so that the last bit is enabled.

III. RESULTS

Matlab simulation was conducted for the proposed Viterbi decoder design, and the BER performance of the proposed Viterbi decoder was plotted, alongside an uncoded QPSK, as shown in Figure 17. The VHDL codes were written at Register Transfer Level for the proposed design, in order to enable a built-in self test. The RTL view of the synthesized circuit is provided in the Figures 9, 10, 11 and 12. An LFSR was synthesized together with the encoder and decoder, so as to provide a random input to the system. Functional simulation and synthesis were conducted using the Quartus II version 9 software, Figure 13 shows the functional simulation, while Figure 14 shows the speed analysis result and Figure 15 shows the synthesis report.

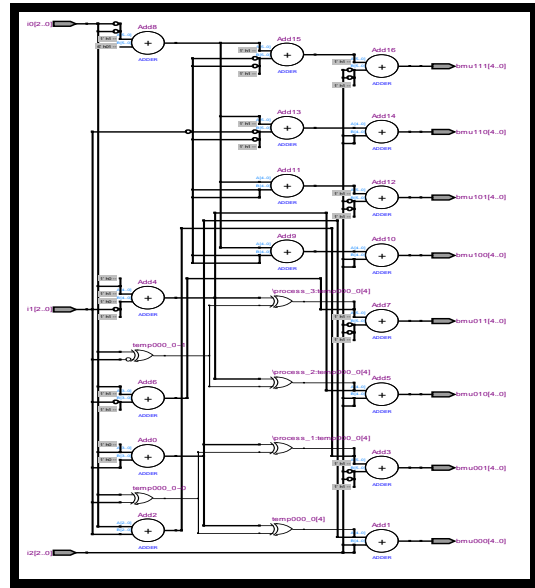


Fig 8. RTL view of the Branch Metric Unit

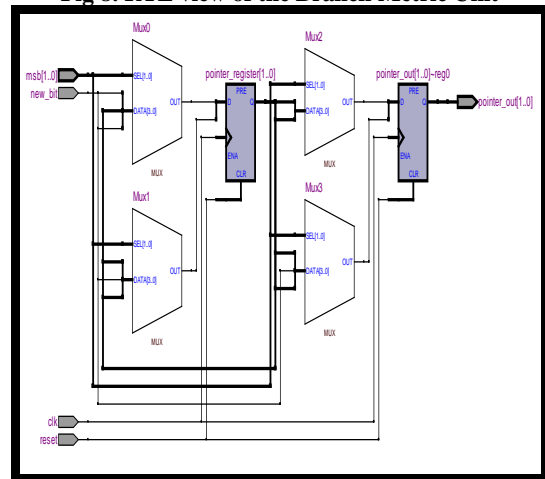


Fig 9. RTL View of the Pointer

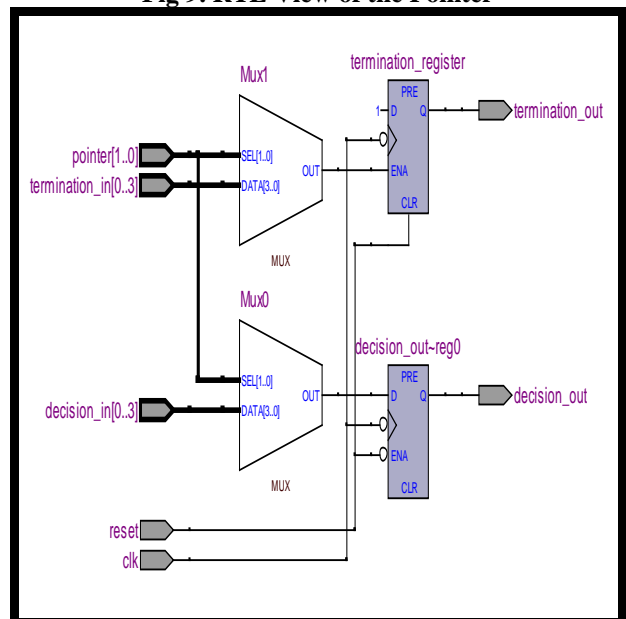


Fig 10. RTL view of the ACSTOSM

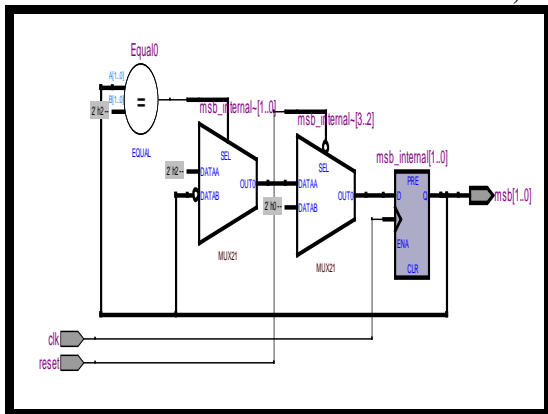


Fig 11. RTL view of the MSB block

**B. Functional Simulation**

The functional simulation of the Viterbi decoder was conducted after synthesis. Results given in Figure 13 shows that the output of the decoder is a delayed version of the random input test sequence, the output data appears after four clock cycles. Since the output sequence and the random input sequence have the same pattern, the designed decoder functions correctly.

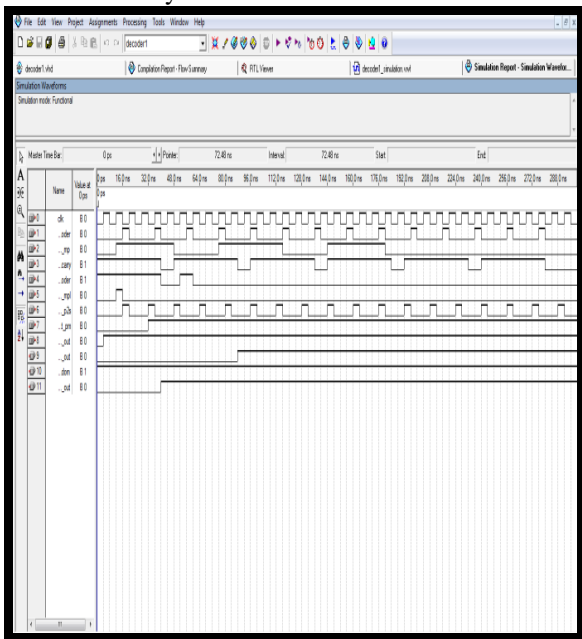


Fig 12: Functional Simulation of the Viterbi decoder

**C. Device Speed and Resource utilization**

Another important parameter is the maximum frequency of the decoder. From the compilation report, it is seen that the system has a maximum frequency of 185.39MHz as shown in Figure 13. This maximum frequency is the inverse of the time delay of longest register to register path, with a high maximum frequency, the decoder will operate at high speed, thus asserting the decoder's efficiency. This high speed of the decoder has been achieved by the parallel ACSU computations, instead of sharing a single ACSU; the parallel ACSU assigned for each state has made it possible to compute the decision bits and path metrics in parallel.

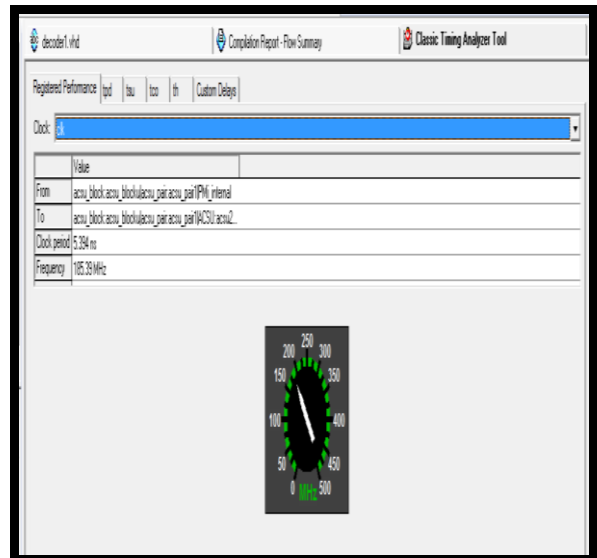


Fig 13: Timing Analysis Result

|                                    |   |
|------------------------------------|---|
| Flow Status                        | Successful - Thu Mar 22 15:54:40 2012   |
| Quartus II Version                 | 9.0 Build 132 02/25/2009 SJ Web Edition |
| Revision Name                      | decoder1                                |
| Top-level Entity Name              | decoder1                                |
| Family                             | Cyclone II                              |
| Met timing requirements            | Yes                                     |
| Total logic elements               | 125 / 4,608 (3 %)                       |
| Total combinational functions      | 115 / 4,608 (2 %)                       |
| Dedicated logic registers          | 63 / 4,608 (1 %)                        |
| Total registers                    | 63                                      |
| Total pins                         | 12 / 89 (13 %)                          |
| Total virtual pins                 | 0                                       |
| Total memory bits                  | 48 / 119,808 (< 1 %)                    |
| Embedded Multiplier 9-bit elements | 0 / 26 (0 %)                            |
| Total PLLs                         | 0 / 2 (0 %)                             |
| Device                             | EP2C5T144C6                             |
| Timing Models                      | Final                                   |

Fig 14: Compilation Summary

The FPGA resource utilization, given in Figure 14 shows that only 125 logic elements, 115 combinational functions and 63 registers were used.

**D. Matlab Simulations**

Varying the code rate for a constant constraint length, as seen from Figure 15, the performance decreases as the code rate increases, thus as the redundancy of the code is increased (which implies more output bit for a single input bit), the performance increases. For Eb/No of 3dB, it is seen that the uncoded QPSK has BER of 0.02288, while for 1/3 code rate, it is 0.003647 and 0.001468 for the 1/5 code rate. Thus the smallest code rate which is 1/5 has the best BER performance as compared with the higher code rate and the uncoded QPSK. This trend shows that entropy is inversely proportional to BER performance.

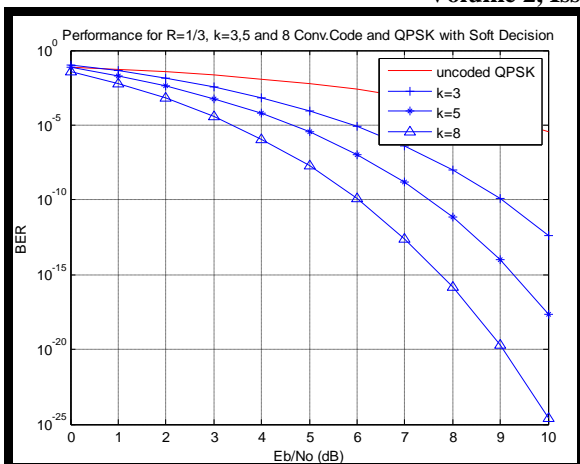


Fig 15. BER Performance for Various Constraint Lengths

The BER performance for the system is thus deduced from the simulation result given in Figure 16. It is seen that the decoder has a coding gain of 4 dB at BER of  $10^{-10}$  over an uncoded QPSK.

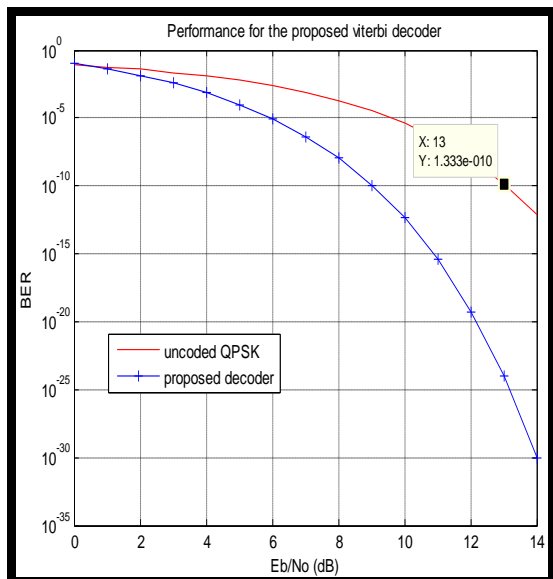


Fig 16. Plot for Performance of the Proposed Viterbi Decoder In Comparison With the Uncoded QPSK

#### IV. CONCLUSION

Finally, comparing the speed and code rate obtained, with that of a previous work [7], there is improvement over the maximum clocking frequency at the same constraint length of 3. In this work a frequency of 185 MHz has been achieved, with a code rate of 1/3, while the previous work recorded a frequency of 85 MHz while having a code rate of 1/2. Thus there is improvement in this work in terms of speed and also coding gain. The designed Viterbi decoder has low power dissipation and also high speed; however, due to various clocking signals in the system, there should be proper clock control, to enable better operation. Another major challenge of the memory less low power design is to make the decoder have knowledge of the current state of the encoder, this calls for synchronization between the transmission and reception terminals, subsequent design should include synchronization

circuit. In terms of handling burst error, the encoder and decoder should be enhanced with inter leaver and de-inter leaver respectively. The design can also be made more flexible through parameterization, so as to enable it to be used for different encoding and decoding requirements. Finally, the highest frequency of clock obtained from the FPGA board used is 50MHz, while the system can run on a higher frequency than that, thus a higher end board should be used to obtain higher clock frequency.

#### REFERENCES

- [1] B. Sklar, "Digital Communications: Fundamentals and Applications", 2<sup>nd</sup> edn, Prentice Hall, 2006.
- [2] C. Arun, and V. Rajamani, "A Low Power and High Speed Viterbi Decoder Based on Deep Pipelined, Clock Blocking and Hazards Filtering", International Journal of Communications, Network and System Sciences, 6, 2009, pp. 575-582.
- [3] E. Dalia, "Low Power Register Exchange Viterbi Decoder for Wireless Applications", PhD thesis, University of Waterloo, Ontario, 2004.
- [4] G. Fettweis, and H. Meyr, "High-Speed Parallel Viterbi Decoding: Algorithm and VLSI Architecture". IEEE Communications Magazine, 1991, pp. 46-55.
- [5] G. Forney, "The Viterbi algorithm", Proceedings of the IEEE, vol.61, no. 3, 1973, pp. 268-278.
- [6] M. Hosemann, R. Habendorf, and P. Gerhard, "Hardware-Software Co design Of A 14.4mbit - 64 State - Viterbi Decoder For An Application-Specific Digital Signal Processor", 2003.
- [7] J. Lin, "High-Speed Viterbi Decoder Design and Implementation with FPGA", Master's thesis, University of Manitoba, 2000.
- [8] S. Ranpara, "On a Viterbi decoder design for low power dissipation, Master's thesis", Virginia Polytechnic Institute and State University, 1999.
- [9] G. Swati, and M. Rajesh, "Reconfigurable Efficient Design of Viterbi Decoder for Wireless Communication Systems". International Journal of Advanced Computer Science, 2011.
- [10] S. Welsen, S. Hussien, and K. Ali, "Design and Implementation of Low-Power Viterbi Decoder for Software-Defined WiMAX Receiver". TELFOR IEEE, 2009.
- [11] Y.N. Chang, H. Suzuki, and K. Parhi, "A 2-mb/s 256-state 10-mw rate-1/3 Viterbi decoder", IEEE J. Solid-State Circuits, vol. 35, no.6, 2000, pp. 826-834.
- [12] R. Sivasubramanian, and A. Varadhan, "An Efficient Implementation of IS-95A CDMA Transceivers through FPGA". ICGST DSP Journal, vol 6, issue 1, 2006, pp. 23-30.
- [13] K. Tsung-Sheng, "Low Complexity Convolutional Codes Using Branch Symmetry", Ph D Desertation, Institute of Communication Engineering Tatung University, 2007.
- [14] S. Viraktamath, and G. Attimarad, "Impact of constraint length on performance of Convolutional CODEC in AWGN channel for image application". International Journal of Engineering Science and Technology, vol. 2(9), 2010, pp. 4696-4700.

## AUTHOR'S PROFILE



**Shaina Suresh** has received her B.E in Electronics from Pune University, Pune in 1998 and M.Tech in Applied Electronics from Dr. M.G.R. University, Chennai in 2007. Currently she is working as Assoc. Prof. in Musaliar College of Engineering and Technology, Pathanamthitta. She has more than 13 years of teaching experience. Her areas of interests are Embedded Systems, Digital System Design, VLSI and Signal Processing, Neural Networks, Digital Signal Processing and Image Processing.



**CH.Kranthi Rekha** had received her B.E in Electronics and Communication Engineering from Madurai Kamaraj University in 2000 and Completed M.Tech from JNTUH, Hyderabad in 2010. Presently she is working as Assoc. Prof. in Talla Padmavathy college of Engineering, India; she has more than 12 years of teaching experience. She is the Author of two Books (Digital communications and Digital Image processing). She was resource person to talk on Image processing. Her area of interest are Neural networks, Image processing, Signal processing, VLSI, Communications. She is life member of ISTE, IETE. She has published number of papers in Journals, national conferences and international conferences.

**Faisal Sani Bala** has received his B.Eng (Hons) in Electrical and Electronic Engineering from University of East London, UK in 2011. He has a sound knowledge in Designing Digital systems and strong in VHDL programming. His areas of interest are Digital system design using VHDL, Digital signal processing, ASIC Design and Embedded systems.