International Journal of Engineering and Technical Research (IJETR)
ISSN: 2321-0869, Volume-2, Issue-5, May 2014

# Implementation of forward error correction technique using Convolutional Encoding with Viterbi Decoding

**Ashima Sood, Nagendra Sah**

***Abstract*— For the last few decades there exist a growing need for reliable data transmission systems. In digital communication systems data gets corrupted due to noise when it flows from source to destination. This is why many error detection and correction methods have been developed in order to ensure a satisfiable quality of data transmission. Convolutional encoding is a forward error correction technique that is used for correction of errors at the receiver end. Convolutional codes protect information by adding redundant bits to the binary data. Viterbi decoding is the technique for decoding the Convolutional codes. The Viterbi algorithm estimates the maximum likelihood path through a trellis based on received symbols. This paper deals with implementation of Convolutional encoder and viterbi decoder using Verilog HDL.**

***Index Terms*— Add-Compare-select(ACS), Convolutional encoder, Verilog HDl, Viterbi decoder***

## I. INTRODUCTION

### *A. Convolutional encoders*

Convolutional codes are widely used as channel codes in practical communication systems for error correction [1].Convolution coding with Viterbi decoding is a FEC technique that is particularly suited to a channel in which transmitted signal is corrupted mainly by additive white Gaussian noise (AWGN).

Convolutional code definition parameters are the following: code rate (R), generating polynomial g (n), number of input bits (k), number of output bits (n) and constraint length [2]. The code rate is the number of transmitted bits per input bit, e.g., a rate ½ encodes 1 bit and produces 2 bits for transmission. One generating polynomial stands for one output. The constraint length is the length of the generating polynomial in bits. The higher it is, the more robust is the code. Passing the information sequence to be transmitted through a linear finite shift register generates a Convolutional code. The shift register consists of k bit stages and n linear algebraic function generators. The contents of shift register are multiplied by respective term in generator matrix and are then XORed together to generate respective code words [3].

Manuscript received May 12, 2014.

### *B. Viterbi decoders*

Viterbi decoders work on Viterbi algorithm to decode the encoded data. The Viterbi decoding algorithm was discovered and analyzed by Viterbi in 1967. The Viterbi algorithm essentially performs maximum likelihood decoding; however, it reduces the computational load by taking advantage of the special structure in the code trellis[4]. The algorithm involves calculating a measure of similarity, or distance between the received signal, at time $t_i$, and all the trellis paths entering each state at time ti. The Viterbi algorithm removes from consideration those trellis paths that could not possibly be

candidates for the maximum likelihood choice[5]. When two paths enter the same state, the one having the best metric is chosen; this path is called the surviving path. This selection of surviving paths is performed for all the states. The decoder continues in this way to advance deeper into the trellis, making decisions by eliminating the least likely paths[6]. The early rejection of the unlikely paths reduces the decoding complexity. In 1969, Omura demonstrated that the Viterbi algorithm is, in fact, maximum likelihood. Note that the goal of selecting the optimum path can be expressed, equivalently, as choosing the codeword with the maximum likelihood metric or as choosing the codeword with the minimum distance metric [7].

## II. IMPLEMENTATION

### *A. Convolutional encoder:*

The various modules of Convolutional encoder and Viterbi decoder is implemented in Verilog HDL and results are shown. The Convolutional encoder considered for implementation is (2,1,3) . The parameters of Convolutional encoder are shown in table 1.1.

Table 1.1: Parameters of the Convolutional encoder

| Definition | Symbol | Value |
|---|---|---|
| Input bits | K | 1 |
| Output bits | N | 2 |
| Code rate | K/N | ½ |
| Constrain length | L | 3 |
| Generator sequence | G | [111;101] |

The schematic of Convolutional encoder with above parameters is shown in figure 1.1

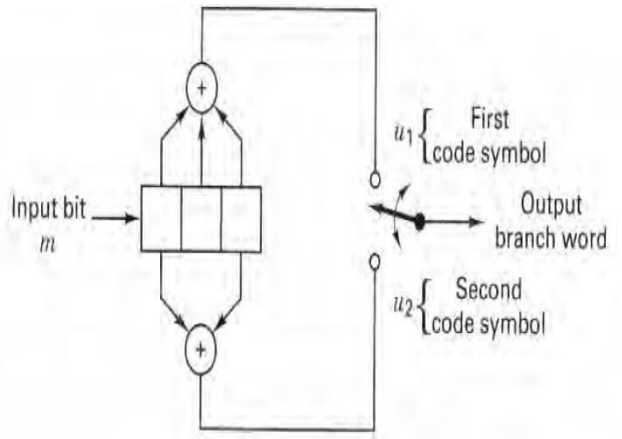
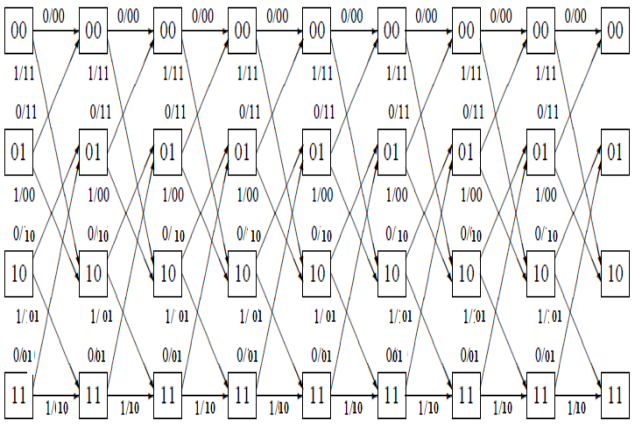



Figure 1.1: Convolutional encoder

Convolutional encoder can be represented in various ways: State table, State diagram and Trellis diagram. The state table of above encoder is shown in table 1.2.

Table 1.2 State table

| INPUT | PRESENT STATE | ENCODED OUTPUT | NEXT STATE |
|---|---|---|---|
| 0 | 0 0 | 0 0 | 0 0 |
| 1 | 0 0 | 1 1 | 1 0 |
| 0 | 0 1 | 1 1 | 0 0 |
| 1 | 0 1 | 0 0 | 1 0 |
| 0 | 1 0 | 1 0 | 0 1 |
| 1 | 1 0 | 0 1 | 1 1 |
| 0 | 1 1 | 0 1 | 0 1 |
| 1 | 1 1 | 1 0 | 1 1 |



Trellis diagram of (2, 1 ,3) encoder

*B. Viterbi decoder:*

A viterbi decoder consist of three basic computation units as shown in figure 1.2

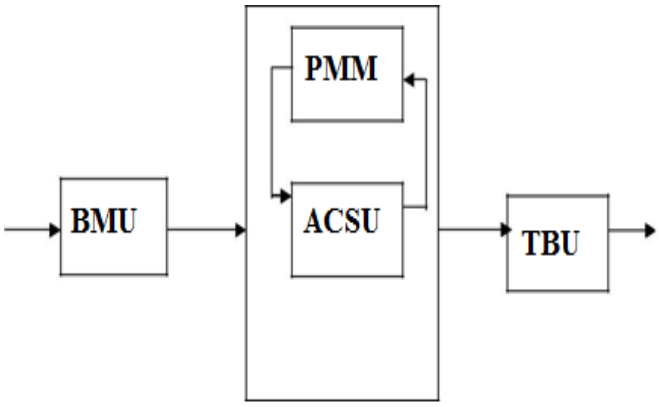



Figure 1.2: Viterbi decoder block diagram

**1) Branch metric unit (BMU):** The branch metric unit calculates the branch metrics by the hamming distance or Euclidean distance. Hard decision viterbi decoder calculates the hamming distance ,while soft decision calculate Euclidean distance[8]. We have considered hard decision viterbi decoder for our implementation. BMU compares the received code symbol with the expected code symbol and counts the number of differing bits.

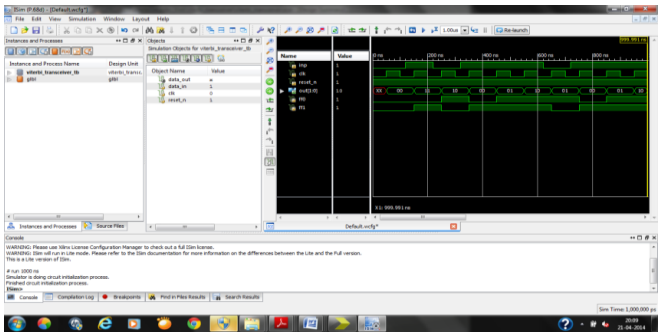
**2) Add-Compare Select unit (ACSU):**The Add-Compare Select Unit (ACSU) adds the Branch Metrics (BM) to the partial Path Metrics (PM) to obtain new path metric.When two paths enter the same state, it compares the new PMs and the one having minimum metric is chosen , this path is called survivor path. The selection for survivor path is done for all states. It then stores the selected PMs in the Path Metric Memory (PMM). The PM of the survivor path of each state is updated and stored back into the PMM .

**3) Trace Back Unit (TBU):**The final unit is the trace-back process or register exchange method, where the survivor path and the output data are identified. The trace-back (TB) and the register-exchange (RE) methods are the two major techniques used for decoding the output sequence. The TB method takes up less area but requires much more time as compared to RE method because it needs to search or trace the survivor path back sequentially. Also, extra hardware is required to reverse the decoded bits. The major disadvantage of the RE approach is that it's routing cost is very high especially in the case of long-constraint lengths. Thus for high performance viterbi decoder trace-back method is preferred [9].

## III. SIMULATION RESULTS

*A. Convolutional encoder:*

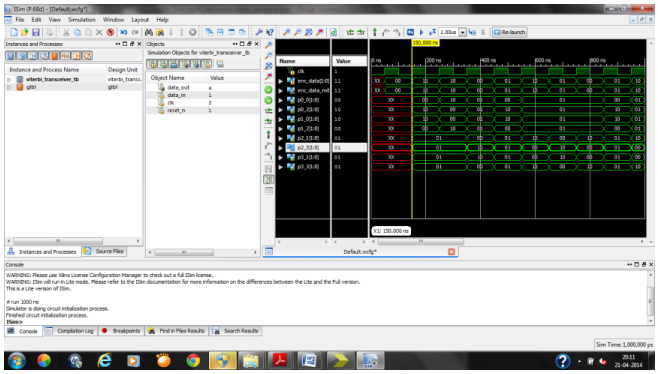
This module provides the encoded output for each incoming bit. This is designed using XOR gate and shift register.



*B. Viterbi decoder:*

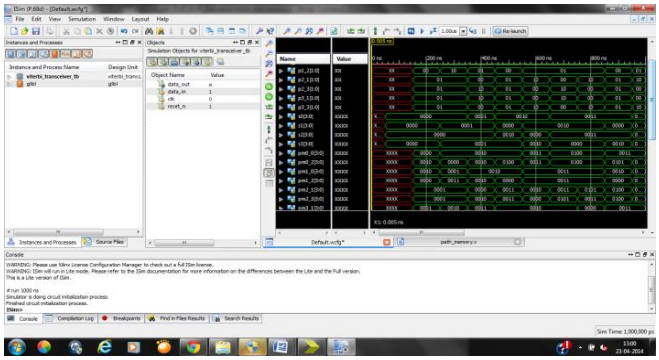
***Branch metric unit(BMU)*** *:*

The encoded output coming from Convolutional encoder module is fed as input to BMU . BMU calculates the hamming distance between received symbol and expected symbol. For (2,1,3) Convolutional encoder three hamming distance are possible:00,01,10.We store the possible hamming distance in hamming distance look up table and use in futher modules. The output waveform of BMU module is shown below:

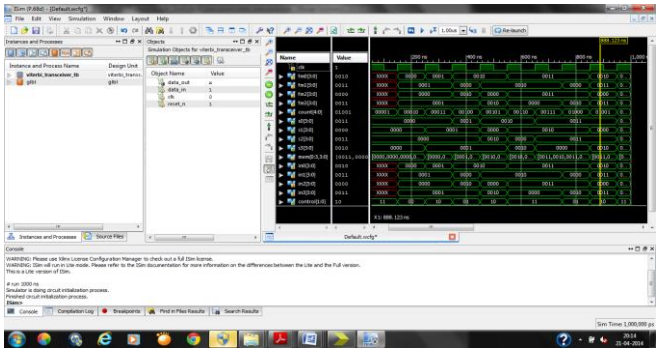


***Add-compare-select Unit (ACSU):***

This module add the incoming branch metrics and partial path metrics stored in PMM to get new path metric[10]. When two paths enter the same state, it compares the new PMs obtained in adder module and the one having minimum metric is chosen , this path is called survivor path. The selection for survivor path is done for all states[11].This module also generate 4 control signals which are required for trace-backing. The waveform is shown below:

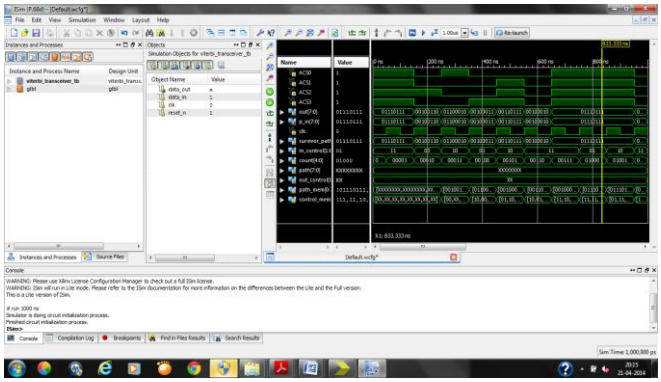


***Path metric memory(PMM)****:*The selected PMs are stored in Path Metric Memory (PMM) module. The PM of the survivor path of each state is updated and stored back into the PMM. Simulation results of PMM module is shown below:

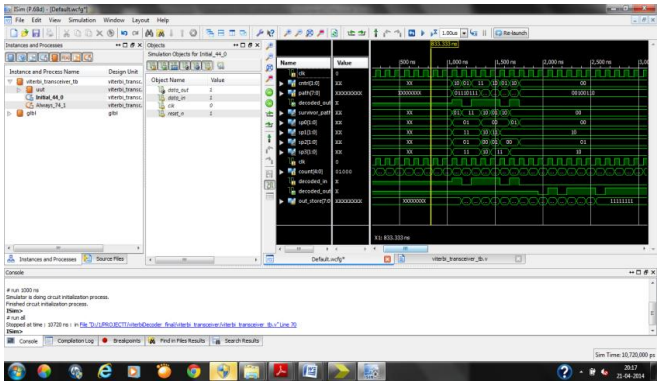


***Trace back unit(TBU):*** Trace Back module extracts the decoded bits, beginning from the state with the minimum PM. The modules required to implement trace-back are:

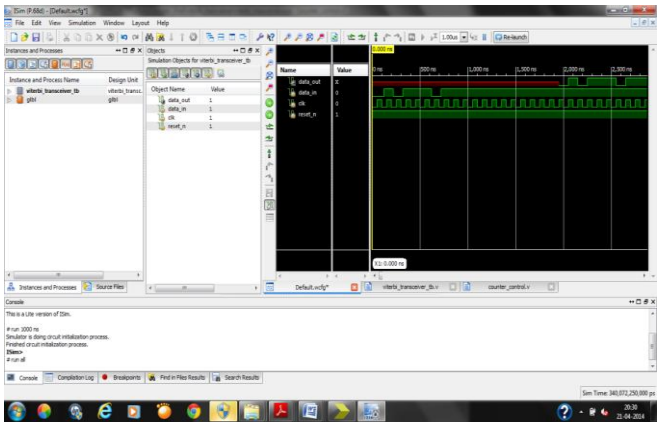
*Path and path- memory*: These modules store the predecessor states (with minimum metric) and present state (with minimum metric) information. 8 registers of 8-bit are used to store the state (predecessor and present) information for all time instants in trellis. The results of both the modules are shown below:



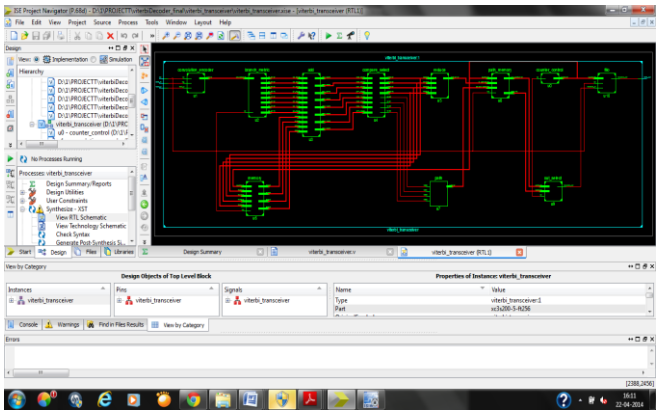
*First-in-last out(FILO):* Once trellis is finished we trace-back from the state with minimum metric and following the survivor path, which originally contributed to the current PM. While tracing back through the trellis, the decoded output sequence, corresponding to the traced branches, is generated in the reverse order. FILO module provide the output in correct order.



**Convolutional encoder and Viterbi decoder:** All sub-modules are integrated in this module. The waveform is shown below:



**RTL Schematic of encoder and decoder:**



## IV. SYNTEHSIS AND TIMING RESULTS

### *C. Synthesis results*

**Device Utilization Summary**

| **Logic Utilization** | **Used** | **Available** | **Utilization** |
|---|---|---|---|
| Total Number Slice Registers | 111 | 3,840 | 2% |
| Number of 4 input LUTs | 203 | 3,840 | 5% |
| Number of occupied Slices | 125 | 1,920 | 6% |
| Total Number of 4 input LUTs | 203 | 3,840 | 5% |
| Number of bonded IOBs | 4 | 173 | 2% |

### *B. Timing Analysis*

| TIMIMG REPORT | VITERBI DECODER |
|---|---|
| Target device | *Spartan3*<br>*XC3S200* |
| Minimum clock period | *15.940ns* |
| Maximum clock freq | *62.736MHz* |
| Clock to set-up cycle time | *12.095 ns* |
| Set-up to clock pad delay | *0.885ns* |
| Clock pad to output pad delay | *8.329 ns* |

## V. CONCLUSION AND FUTURE SCOPE

Convolutional encoder and viterbi decoder are widely used in digital communication systems. Viterbi decoder is based on viterbi algorithm .The Viterbi algorithm (VA) is also known as maximum likelihood (ML)-decoding algorithm for Convolutional codes. Maximum likelihood decoding means finding the code branch in the code trellis that was most likely to be transmitted.. We have implemented both encoder and decoder in Verilog HDL and results are obtained. It has been found that the decoding speed of VD is dominated by its ACS recursion i.e. its feedback loop. For the time issue, we do not implement a higher performance Viterbi decoder. So in future, the speed of viterbi decoder can be improved using some optimization techniques.

## REFERENCES


[1] R. W. Hamming, "Error detecting and correcting codes," Bell Sys. Tech.J, vol. 29, pp. 147–160, 1950.

[2] Anubhuti khare, Manish saxena, Jagdish patel, "FPGA based implementation of Viterbi decoder",IJEAT ISSN 2249-8959 Vol- 1Issue-1efficient, October 2011.

[3] Chakraborty, P. Raha, A.Bhattacharya, R Dutta, "Speed optimization Of a FPGA based modified Viterbi Decoder", IEEE, 978-1-4673- 2907- 1/13/,2013.

[4] Chip fleming"A tutorial on Convolutional coding with viterbi decoder.

[5] G. Davis Forney Jr., "The Viterbi algorithm," Proc. IEEE, vol. 61, pp 268-278, March 1973.

[6] http://en.wikipedia.org/wiki/Coding_theory.

[7] Varsha P. Patil , Prof. D. G. Chougule , Radhika.R.Naik "Viterbi Algorithm for error detection and correction", *ISSN: 2278-2834-, ISBN: 2278-8735, PP: 60-65*

[8] G.S. Suganya, G.kavya ,"RTL Design and VLSI Implementation of an efficient Convolutional Encoder and Adaptive Viterbi Decoder", 978- 1-4673-4866-9/13/$31.00 ©2013 IEEE.

[9] M. Kivioja, J. Isoaho and L.Vanska, "Design and implementation of Viterbi decoder with FPGAs," Journal of VLSI Signal Processing Systems for Signal, Image, and Video Technology, Kluwer Academic Publishers, vol. 21,no 1, pp. 5-14, May 1999.

[10] Wong, Y.S. et.al "Implementation of convolutional encoder and Viterbi decoder using VHDL" IEEE Tran. on Inform. Theory, Pp. 22- 25, Nov. 2009.

[11] Hema.S, Suresh Babu.V, Ramesh P, "FPGA Implementation of Viterbi Decoder", 6th WSEAS Int. Conf. on Electronics,February 2007.