Indian Journal of Science and Technology, Vol 12(36), DOI: 10.17485/ijst/2019/v12i36/147574, September 2019 ISSN (Online): 0974-5645

# Design and Implementation of High Speed Viterbi Decoder and Convolution Encoder for SDR

R. J. Reshmi\* and C. Raji

School of ECE, REVA University, Bangalore – 560064, Karnataka, India; reshuachu26@gmail.com, raji.c@reva.edu.in

### **Abstract**

**Objectives:** Data transmission in wireless communication system will be affected by noise. The Viterbi decoder and convolutional encoder are best suited for forward error detection and correcting codes for a channel. **Methods:** The study proposes convolutional encoder with 1/2 code rate and constrained length of 3. An improved architecture which optimizes critical path delay is proposed to achieve higher speeds. The design is carried out in MATLAB. The simulation of proposed architecture is done using XILINX 14.1 and implementation is done using FPGA SPARATAN 3AN. **Findings:** Software defined radio uses different modulation and encoding procedures and techniques by varying its configuration. In recent years, it has earned a great reputation for its flexible ability to adapt different procedures and techniques without changing the existing hardware. SDR offers a flexible method of communication and decreasing the cost complexity problem. The Viterbi decoder exploits advantages in SDR applications mainly because of area efficiency and speed. **Application:** The Viterbi decoder will be able to correct and detect one-bit error at the input bit stream of data. For high speed SDR applications the proposed architecture is suitable.

**Keywords:** Convolutional Encoder, Viterbi, FPGA, SDR

## 1. Introduction

Convolutional codes are best suited for forward error correcting that has been extensively used in digital communication systems making transmission more reliable. To transfer data along a noisy channel, convolution codes offer good outcomes compared to block codes. Additional bits are given to input bit in convolutional encoding to reduce the probability of error. The effective technique for decoding the convolutional codes is viterbi decoder<sup>1,2</sup>. A Finite State Machine can be used to design the Convolution encoder. For each clock pulse, the encoder divides a single input bit into 2 or more bits corresponding to a polynomial generator. According to generator polynomial logic '1' denotes that there is connection between the stages and logic '0' denotes that there is no connection between the stages. The convolution encoder having a code rate of ½ is shown in Figure 1.

Using trellis diagram, state diagram and state table, convolutional encoder can be represented. The shift register content of the encoder represents the states Figure 1. State diagram and trellis diagram of convolution encoder are displayed in Figure 2(a) and 2(b)<sup>3</sup>. Trellis diagram is being used to decide the cumulative distances from the received input sequence to obtain similar transmissible sequence.

#### 1.2 Viterbi Decoder

Viterbi decoding is known for implementing the technique of maximum likelihood decoding. The Viterbi decoder examines the complete received input sequence. The Viterbi decoder uses the Viterbi algorithm for decoding the bit stream which is encoded using a Convolutional code. The hard decision Viterbi decoding, uses hamming distance which depends on possibilities of 0 and 1. To determine the count of bits on the operation of x or gate, the hamming distance is used<sup>4</sup>.

<sup>\*</sup>Author for correspondence

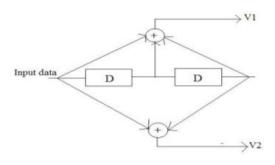


Figure 1. Convolution encoder flip flop diagram.

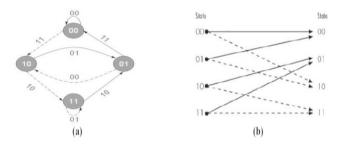
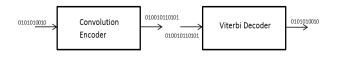


Figure 2. (a) State diagram (b) Trellis diagram.

# 2. Methodology

The Viterbi decoder comprised of following major block:

- 1. Path Metric Unit (PMU)
- 2. Branch Metric Unit (BMU)
- 3. Trace Back Unit (TBU).



**Figure 3.** Viterbi decoder and Convolutional Encoder Block diagram.

#### 2.1 BMU

The purpose of the BM unit is to determine branch metric state values. The hard decision type decoder is taken using the hamming distance metric values are computed. It executes the EX-OR operation between each state's possible output symbol and the symbol received.

#### **2.2 ACSU**

The ACSU consists of Comparator, Multiplexer and adder.

The state metric is the sum of the minimum value of previous state metric at that state and current branch metric. Adder adds the previous state of the state metric and branch metric of the current state. At each state the state metric value is compared using comparator and multiplexer selects the state metric with the minimum value<sup>5.6</sup>.

#### **2.3 PMU**

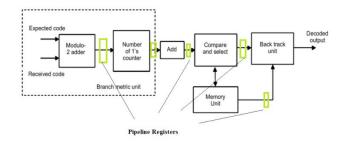
PMU is the summation of the pervious iteration of path metric and the current input of the branch metric. At each node of trellis the partial path metric is computed and branch metric computed at the BMU is taken. The memory element stores the metric for next iteration and to find the path metric with minimum value at the last stage (iteration) of the trellis<sup>7.8</sup>.

#### **2.4 TBU**

TBU recognise the output and survivor path. The original bit given to the convolutional encoder is extracted by the trace back unit<sup>2</sup>.

The Computation of Hamming Distance Module

This module differentiates the obtained codes with the actual expected input codes of the current state in order to calculate the hamming distance between them. Figure 5 shows the schematic diagram for calculating the hamming distance.



Proposed Pipeline Architecture

Figure 4. Proposed architecture diagram.

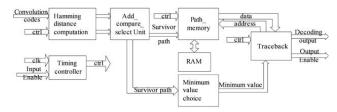


Figure 5. Schematic diagram of Hamming Distance computation.

# 3. Viterbi Algorithm

Viterbi algorithm is termed efficient algorithm as it reduces error possibilities<sup>10</sup>. The following three steps can explain the Viterbi algorithm briefly:

- 1. Trellis is weighed. Branch metrics are calculated.
- 2. In terms of the minimum path to n time, recursively calculate the minimum path to n-1 time. Procedure has been used in this step to repeatedly refresh the signal survivor path. This is called the recursion Add Compare Select (ACS).
- 3. Iteratively find the shortest distance that uses Step 2 decisions leads to each trellis state. Survivor path decoding finds the shortest path for the state. Lastly, the survivor paths incorporate the most likely signal path into a unique path when they are traced back in time. The flow chart for the Viterbi algorithm is shown in the Figure 6.

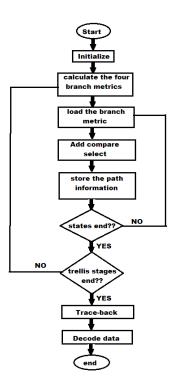


Figure 6. Flow chart of Viterbi trace back algorithm.

# 4. Result and Discussion

Matlab is used to determine, next state and output for the present state of convolutional encoder as shown in Figures 7 and 8.

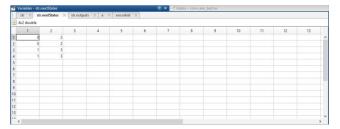


Figure 7. Matlab result for next state for logic 1 and logic 0.



Figure 8. Output for logic 1 and logic 0.

The min and max values for next state and output is obtained as shown in Figure 9.

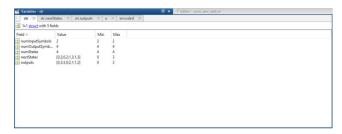


Figure 9. Min and max values for next state and output.

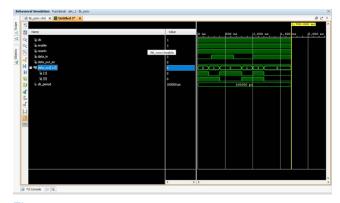


Figure 10. Convolutional Encoder- simulation result.

The Viterbi Decoder and Convolutional Encoder is designed and its behavioural simulation is verified using Xilinx ISE 14.1. The device timing summary for Viterbi decoder and convolutional encoder was obtained. The simulation result for encoder and decoder is shown in the Figures 10 and 11. The RTL schematic for the Viterbi

decoder and convolutional encoder is obtained using Xilinx ISE 14.1 as shown in the Figure 12.

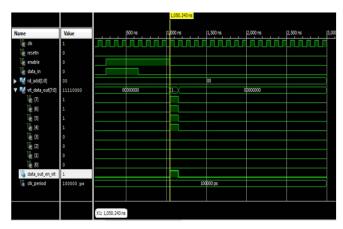


Figure 11. Viterbi decoder-simulation result.

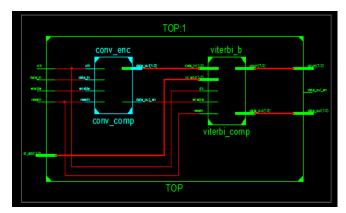


Figure 12. Viterbi decoder and convolutional encoder -RTL schematic.

Table 1. Timing Summary (Convolutional encoder)

Min.period (max.freq:1919.01Mhz)	0.521ns
Before Clock,Min. input arrival time	0.544ns
After Clock,required Max output time	0.992ns
Combinational path delay (maximum)	No path

Table 2. Timing summary (Viterbi Decoder)

Min.period (max.freq:561.798Mhz)	1.780ns
Before Clock,Min. input arrival time	0.548ns
After Clock,required Max output time	0.511ns
Combinational path delay (maximum)	No path

The timing utilization is obtained using Xilinx ISE14.1. The critical path delay, combinational path delay and also maximum and minimum time required before

and after the clock for encoder and decoder is determined as shown in Tables 1 and 2.

The Verilog HDL is used in Xilinx ISE 14.1 to design and synthesize the proposed architecture. The RTL schematic and timing summary for Viterbi decoder and convolutional encoder was obtained.

### 5. Conclusion

SDR is the technology for controlling radio performance through software. The design convolutional encoder and Viterbi decoder using trace back method and its performance characteristics in terms of, timing, summary and HDL synthesis report was successfully obtained. A bit of error was detected and corrected by the Viterbi decoder in the data input bit stream. The frequency of convolutional encoder is 1919.010 MHz and Viterbi decoder is 561.798 MHz. Thus, the implemented method where the speed is high, making the proposed architecture better suited for SDR applications with high speed.

### 6. References

- Lin KC, Liu W, Yeh W, Chang L, Hwu W, Chen S, Hsiung P. A tiling-scheme viterbi decoder in software defined radio for GPUs. 17th International Conference on Wireless Communications, Networking and Zobile Computing; 2011. p. 1–4.
- Viterbi AJ. Convolution codes and their performance in communication systems. IEEE Transactions on Communication Technology; 1971. p. 751–72. https://doi. org/10.1109/TCOM.1971.1090700
- Bhowal S. Transformation of ACS module to CSA module of low-power Viterbi decoder for digital wireless communication applications. International Conference on Advances in Computing, Communications and Informatics; 2013. https://doi.org/10.1109/ICACCI.2013.6637182
- Lai KY-T. A high-speed low-power pipelined viterbi decoder: Breaking the ACS-bottleneck. International Conference on Green Circuits and Systems; 2010.
- Lai KY-T. An efficient metric normalization architechture for high-speed low-power viterbi decoder. TENCON 2007 - 2007 IEEE Region 10 Conference; 2007.
- Middya A, Dhar AS. Real-time area efficient and high speed architecture design of viterbi decoder. 2016 2nd International Conference on Advances in Electrical, Electronics, Information, Communication and Bio-Informatics (AEEICB); 2016

- 7. Hariprasad SA, Arunlal KS. An efficient viterbi decoder. International Journal of Advanced Information Technology. 2012; 2(1):1–16. https://doi.org/10.5121/ijcsea.2012.2110
- 8. Yihua C, Meilin S, Yifan N. FPGA implementation of trellis coded modulation encode on SDR communication system. IEEE 11th International Conference on Electronic Measurement & Instruments; 2013.
- 9. Haridas SL, Choudhari NK. Design of viterbi decoder with modified traceback and hybrid register exchange pro-
- cessing. Proceedings of the International Conference on Advances in Computing, Communication and Control; 2009. p. 223–26. https://doi.org/10.1145/1523103.1523150
- 10. Singh M. Performance and analysis of viterbi decoder using VHDL. International Journal of Innovative Research in Technology. 2014; 1(3):1-9.