Journal of Engineering Sciences
Vol 11, Issue 2, FEB/2020
ISSN NO:0377-9254

A Configurable and Low Power Hard-Decision Viterbi Decoder in VLSI Architecture

Vamsikrishna Jupudi[1].,
Department Of Electronics and Communication, Engineering,
Samskruti College of Engg & Tech, Ghatkesar, Telangana.India,
Email: jupudi.vamsikrishna8@gmail.com

Vikram kumar Meesala[2] ,
Department Of Electronics and Communication, Samskruti
College of Engg & Tech, Ghatkesar, Telangana,India.
Email: vikramkumar1416@gmail.com

Abstract - Convolutional encoding and viterbi algorithm are basic concepts of error correction method. Viterbi algorithm is one of decoding method for data error correction. In VLSI area, the design challenges are usually about its power, area consumption, speed, complexity, and configurability. This paper proposed a configurable and low power design for hard-decision viterbi decoder in VLSI. For any number of traceback the design can be configured by increasing or decreasing the size of traceback parameters. It needs N+2 clock cycles latency to complete the process, where N is the number of traceback. In this research, configuration test have been conducted for N=32 and N=64. The design also has been synthesized in Xilinx as target boards. It gives good synthesis results in operational speed and area consumption.

Key Words: Viterbi decoder, convolution encoder, low power, VLSI.

1. INTRODUCTION

In the error correction field, there are many methods proposed. One of the best known method is using viterbi algorithm. Its complete error correction method consists of three main parts: the convolutional encoding, the error disturbance, and the viterbi decoding. The original data will be convoluted using its specific convolution formula to produce a codeword. Each codeword consists of 2-bits. Codeword is representation of both original data and its redundant. Hence, if errors occur in the data transmission, we can reconstruct the correct data using viterbi algorithm.

Habib et.al [1] discussed about design space exploration in hard decision viterbi algorithm and its VLSI implementation. Jinjin He et.al [2] proposed a high-speed and low-power viterbi decoder by using T-algorithm as precomputation architecture for trellis coded modulation (TCM) system. Low-power issue is important because, viterbi decoder is power hungry module in TCM [3]. Regarding this issue, Chakraborty et.al [4] also proposed a design to reduce power consumption in viterbi decoder. Another issue is about operational speed. Jinjin He et.al and Azhar et.al discussed about how to increase the operational speed of viterbi decoder.

There is still a chance to explore about design method for configurable and low power viterbi decoder. The configurable design means that we can change the traceback umber by changing the traceback parameters without any additional major modifications in RTL design. The low power design means that the design consumes low power. Those are goals of this research. Hard decision category is chosen because of its simplicity for basic fundamental research. Hard decision design means viterbi decoder only use two level decision, high "1" and low "0".

2. CONVOLUTION ENCODER

Convolutional codes are sometimes referred as trellis codes. Convolutional encoding is simple, but decoding is much more difficult. Convolutional codes are usually characterized by two parameters and the patterns of *n* modulo-2 adders. The two important parameters are the code rate and constraint length. The code rate is the number of transmitted bits per input bit, e.g., a rate 1/2 encodes 1 bit and produces 2 bits for transmission.

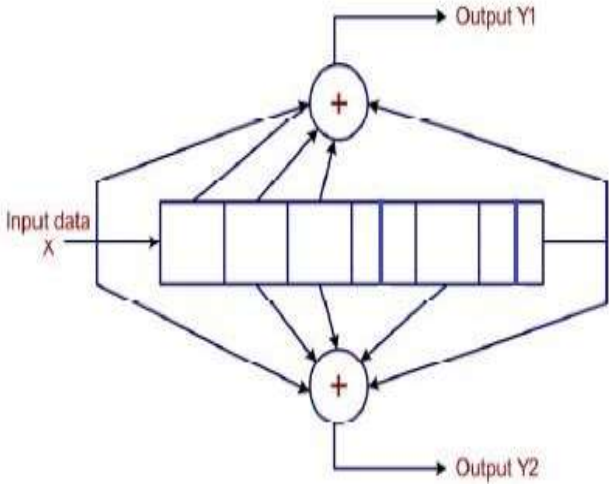


Figure 1: Convolutional encoder with constraint length k=9 and code rate (k/n)=1/2

Convolutional codes are sometimes referred as trellis codes. Convolutional encoding is simple, but decoding is much more difficult. Convolutional codes are usually characterized by two parameters and the patterns of *n* modulo-2 adders. The two important parameters are the code rate and constraint length. The code rate is the number of transmitted bits per input bit, e.g., a rate 1/2 encodes 1 bit and produces 2 bits for transmission.

The encoder has *n* modulo-2 adders, and *n* generator polynomials, one for each adder. An input bit *m1* is fed into the leftmost register. Using the generator polynomials and the existing values in the remaining registers, the encoder outputs *n* bits. As shown in Figure 1, where we have a general encoder designed with a code rate (k/n) of 1/2 and an information sequence that is being shifted in to the register *m* of 1 bit at a time. The shift register has a constraint length (K) of 3, equal to the number of stages in the register. The encoder output is called code symbols. At initialization all stages in the encoder shall be initially set to zero. The output of the encoder is determined by the generator polynomial equations. Since the complexity of the encoder increases exponentially with the constraint length, none of the encoders uses more than a constraint length of 9, for practical reasons [5].

Convolutional code definition parameters are the following: code rate (R), generating polynomial g (n), and number of input bits (k), number of output bits (n) and constraint length (K). 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. Passing the information sequence to be transmitted through a linear finite shift register generates a convolutional code. Tree diagram, state diagram and trellis diagram are the three alternative methods that are often used to describe the convolutional code.

3. VITERBI DECODER

The basic building blocks of Viterbi decoder are branch metric unit (BMU), path metric unit (PMU), add compare and select unit (ACSU) and survivor memory management unit (SMU).

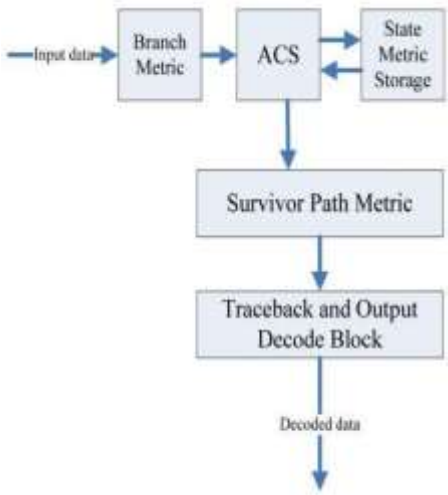


Figure 2: Block diagram of Viterbi decoder

3.1 BRANCH METRIC UNIT (BMU)

The first unit is called branch metric unit. Here the received data symbols are compared to the ideal outputs of the encoder from the transmitter and branch metric is calculated. Hamming distance or the Euclidean distance is used for branch metric computation.

3.2 PATH METRIC UNIT

The second unit, called path metric computation unit, calculates the path metrics of a stage by adding the branch metrics, associated with a received symbol, to the path metrics from the previous stage of the trellis.

3.3 ADD COMPARE AND SELECT UNIT (ACSU)

The ACSU unit is composed of 64 ACS units, each is composed of an ACS butterfly module, which adds the corresponding BM to corresponding PM, compares the new PM, feeds the selected PM to ACSU unit and generates the decision bits. The decoder implements adders for computing the path metric and a compare-select section to decide on the best path. Using bit serial architecture can reduce the power of Viterbi decoder.

3.3.1 BUTTERFLY STRUCTURE

The two ACS processors that share the same inputs can be grouped in a Butterfly processor. There are 64 states so 32 butterflies are needed. Because of butterfly structure, interconnections are reduced due to which power is reduced. Figure shows the radix 2- butterfly module. The inputs j and j + N/2 are shared for the output j and 2j+1.

There are two paths reaching towards each node. First is for transition 0 and other is for transition 1. When input 0 is given to current state, the next state is 2j but when input to current state is 1, output is 2j+1.The path with lower value of accumulated path metric is assigned. The branch metric and path metric of the ith path and jth path are accumulated. If the accumulated path metric of ith path is smaller, decision is made in the favor of ith path and decision bit generated is 0. The new path metric will be the accumulated path metric of ith path.

If

BM i(p)+PM i(p)<BM(j,p)+PM(j,p)

then

Dec(p)=0

PM (p) = PM i(p) + BM i(p)

If the accumulated path metric of jth path is smaller, decision is made in the favor of jth path and decision bit generated is 1. The new path metric will be the accumulated path metric of jth path

If

BM i(p) + PM i(p) > BM (j, p)+PM (j, p)

Dec(p)=1

PM(p) = PM (j, p) + BM (j, p)

If

BM i(q) + PM i(q) < BM (j, p)+PM (j, q)

Then

Dec(q)=0

PM (q) = PM (j, q) + BM (j, q)

If

BM i(q) + PM i(q) > BM (j, q) +PM (j, q)

Then

Dec(q)=1

PM (q) = PM (j, q) + BM (j, q)

According to the symmetric characteristics of viterbi decoder

BM i(q) = BM (j, p)

BM i(p) = BM (j, q)

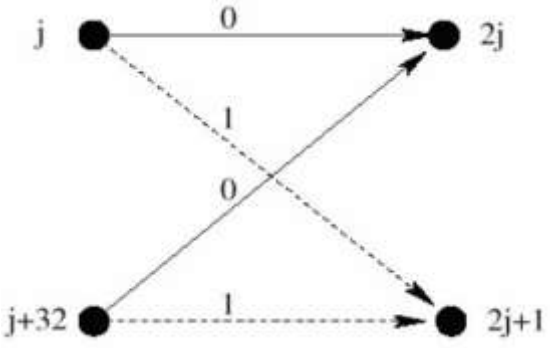


Figure 3: Radix 2 Butterfly Structure

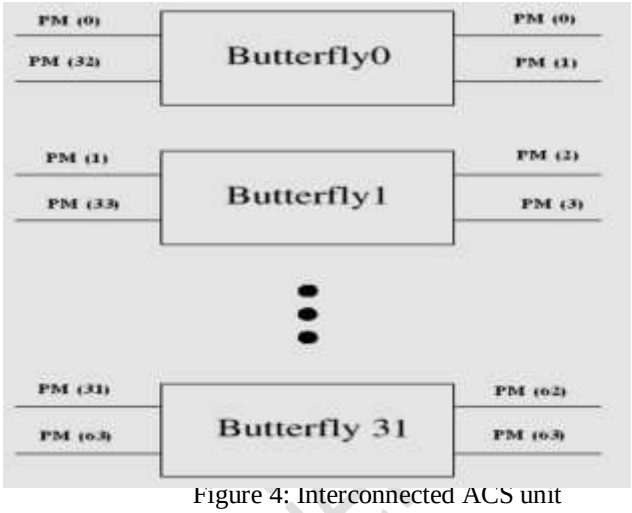
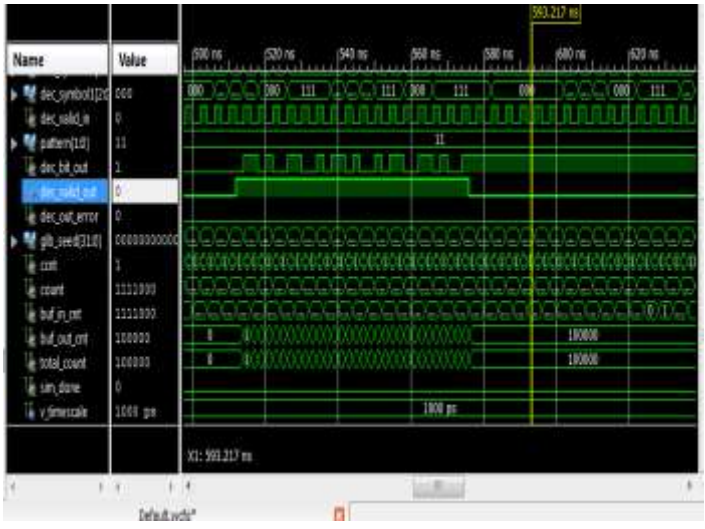


Figure 4: Interconnected ACS unit

3.4 TRACEBACK METHOD

In the TB method, the storage can be implemented as RAM and is called the path memory. Comparisons in the ACS unit and does not store the actual survivors. After at least L branches have been processed, the trellis connections are recalled in the reverse order and through the trellis diagram the path is traced back. Beginning from the state with the minimum PM, the TB method extracts the decoded bits. Beginning at this state and tracing backward in time by following the survivor path, a unique path is identified which originally contributed to the current PM. While tracing back through the trellis, the decoded output sequence, is generated in the reverse order [8], corresponding to the traced branches. Trace back architecture limits the decoding speed [9], since it has a limited memory bandwidth in nature.

4. RESULT



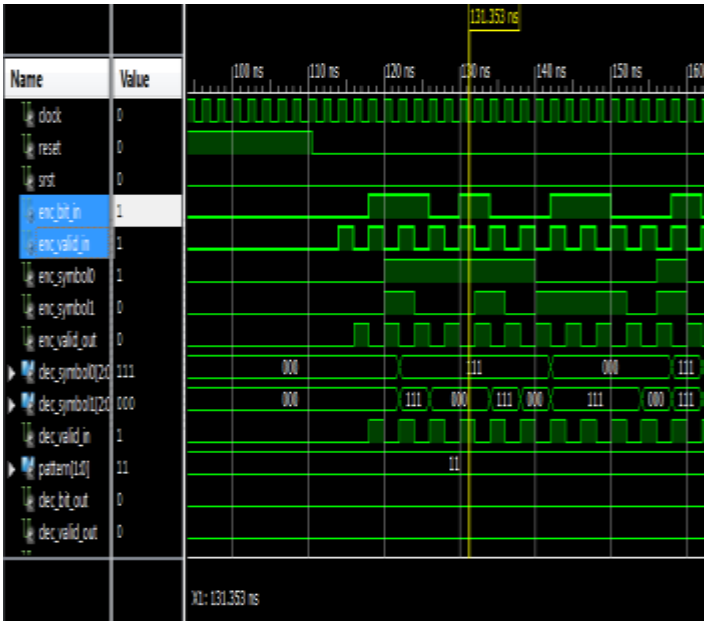


Figure 5: Simulation result for encoder and decoder

5. REFERENCES

[1] S. K. Hasnain, and Azam Beg and "Performance Analysis of Viterbi Decoder Using a DSP Technique", 8th IEEE International Multitopic Conference(ITMIC'04), Dec 2004,pp.201-207.

[2] S. Ranpara, On a Viterbi Decoder Design for LowPower Dissipation, M.S. Thesis, Dept. of Electrical and Computer Eng., Virginia Polytechnic Institute and State University, April 1999.

[3] "Design of a High-Throughput Low-Power IS95 Viterbi Decoder" Xun Liu Marios C. Papaefthymiou. Advanced Computer Architecture Laboratory, Department of Electrical Engineering and Computer Science, University of Michigan, Ann Arbor, Michigan.

[4] R.A.Abdallah, N.R.Shanbhag, "Error-resilient low-power viterbi decoder architectures", IEEE Trans. Signal Process., vol. 57, no. 12, pp. 4906-4917, Dec. 2009 Modulation Detection and Coding" Tommy Oberg (2001). Wiley and Sons. Pp 81- 86.

[5] "VLSI Design and Implementation of High Speed Viterbi Decoder" WANG Jin-xiang, YOU Yu-xin, LAI Feng-chang and YE Yi-zheng (2002).

[6] "A Convolutional Code Decoder Design Using Viterbi Algorithm with Register Exchange History Unit" Vasily P. Pribylov, Alexander I. Plyasunov (2005). SIBCON. IEEE.

[7]The CDMA wireless phone QCP-800 specifications, 1996.

[8]"Efficient Scalable Architectures for Viterbi Decoders"Stefan Bitterlich and Heinrich Meyr (1993). Aachen University of Technology, Templergraben , Germany. pp 89-100.