International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 3 Issue 11, November-2014

Different Approaches for Implementation of Viterbi Decoder on Reconfigurable Platform

Amruta J. Mandwale
Electronics & Telecommunication Department
SKN Sinhgad College of Engg. Korti Pandharpur, India

Altaf O. Mulani
Electronics & Telecommunication Department
SKN Sinhgad College of Engg., Korti, Pandharpur, India

***Abstract*— Generally, the data transmitted over any communication channel is affected due to noise. So, for detecting and correcting the errors due to channel noise, encoding and decoding should be performed at the transmitter and receiver end respectively. The Viterbi algorithm is one of the most popular algorithms for decoding convolution codes. Using VLSI technology, the system requires low power, less area and high speed constraints while designing. A high-speed Viterbi decoder is a challenging task due to the recursive iteration of various steps followed for decoding process. In this paper, different kind of implementation of Viterbi decoder along with their performance have been discussed.**

Keywords*— GDIL, Register exchange, Adaptive viterbi decoder, Non-polynomial-approach, Viterbi decoder, VLSI, FPGA.

I. INTRODUCTION

Now-a-days most of the digital communication systems convolutionally encodes the transmitted data to compensate for various noises. For its efficiency the Viterbi algorithm has proven to be a very efficient algorithm for forward error correction of convolutionally encoded messages [4]. The Viterbi decoding algorithm, proposed by Viterbi, is a decoding process for convolutional codes in memory-less noise. The algorithm can be applied to a host of problems encountered in the design of communication systems. The Viterbi Algorithm finds the most-likely state transition sequence in a state diagram, given a sequence of symbols. The Viterbi algorithm is used to find the most likely noiseless finite-state sequence.

It is well known that data transmissions over wireless channels are affected by attenuation, distortion, interference and noise, which affect the receiver's ability to receive correct information. Convolutional encoding with Viterbi decoding is a powerful method for forward error detection and correction. It has been widely deployed in many wireless communication systems to improve the limited capacity of the communication channels.

Like any error-correcting code, a convolutional code works by adding some structured redundant information to the user's data and then correcting errors using this information. The encoder structure is shown in Fig.1. Assume input bit is 11011.So by this encoder we get encoded output equal to 11 01 01 00 01.

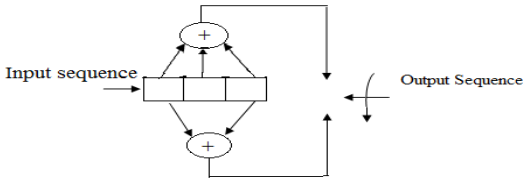

Fig.1: convolution Encoder

Generally, a Viterbi decoder consists of three basic computation units:

Branch Metric Unit (BMU)
Add- Compare-Select Unit (ACSU) and
Trace Back Unit (TBU).

The BMU calculates the branch metrics by the hamming distance or Euclidean distance and the ACSU calculates a summation of the branch metric from the BMU and previous state metrics, which are called the path metrics. After this summation, the value of each state is updated and then the survivor path is chosen by comparing path metrics. The TBU processes the decisions made in the BMU and ACSU and outputs the decoded data. The feedback loop of the ACSU is a major critical path for the Viterbi decoder.

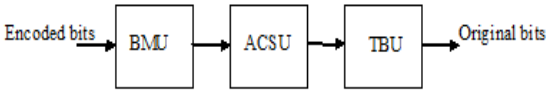

Fig.2: Viterbi decoder data flow

II. LITERATURE REVIEW

[1] In this paper, GDIL technique allows reducing delay, and area of digital circuits while maintaining low complexity of logic design. Comparison with traditional CMOS and various pass-transistor logic design techniques is given in this paper. The GDIL approach allows of a wide range of complex logic functions using only two transistors. Using this method we can achieve fast, low-power circuits, using a reduced number of transistors (as compared to CMOS and existing PTL techniques). The GDI method is based on the use of a simple cell as shown in Fig.3.

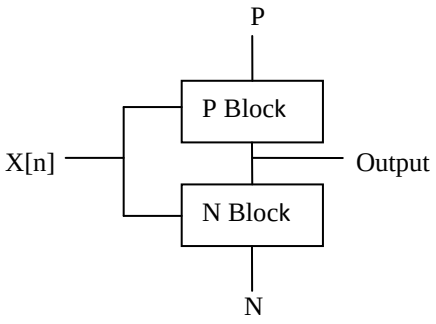

Fig.3: Basic GDI cell

TABLE 1: Truth table for GDI logic

VARIOUS LOGIC FUNCTIONS OF GDI CELL FOR DIFFERENT INPUT CONFIGURATIONS

N	P	G	Out	Function
'0'	B	A	$\overline{A}B$	F1
B	'1'	A	$\overline{A}+B$	F2
'1'	B	A	$A+B$	OR
B	'0'	A	AB	AND
C	B	A	$\overline{A}B+AC$	MUX
'0'	'1'	A	$\overline{A}$	NOT

Advantages of GDI

- For low power digital circuit design which allows reducing power consumption, delay and area of the digital circuit.
- Suitable for design of fast, low power circuits using a reduced number of transistors as compared to CMOS and existing Pass Transistor Logic techniques.
- A simple change of the input configuration of the simple GDI cell corresponds to very different Boolean functions.
- Most of these functions are complex (6–12 transistors) in CMOS, as well as in standard Pass Transistor Logic (PTL) implementations but very simple (only two transistors per function) in the GDI design method.
- This GDIL undoubtedly reduces area as lesser number of LUTs and CLBs are used in FPGA prototyping.

[2]. In this paper, a light-weighted pipelined serial Viterbi Decoder is implemented for resource saving purpose. In this paper they proposed a technique which stores the trace back bits in RAM instead of register array and trace back operation is not needed. Also for string the BMU and ACS values 'Metric RAM' is used. This also supplies the previous values as compared to simple RAM used for trace back.

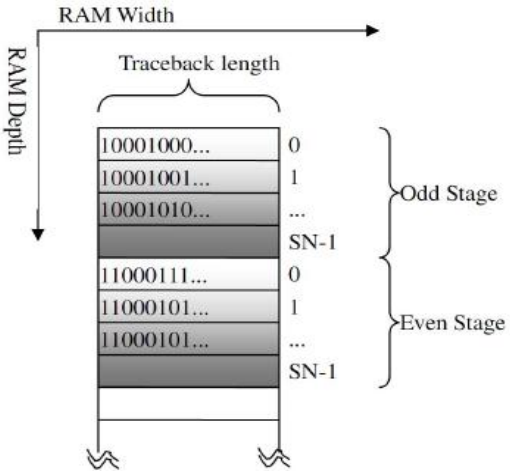

Fig.4: Trace back RAM architecture

[3]. This paper gives information about adaptive Viterbi decoder with different parameters used for every block in designing decoder. Instead of saving data in RAM, data is stored in buffers for designing path storage block. For reducing the time when performing branch metric calculation parallel processing is done with Hamming distance. While designing the ACS unit, ROM and 3bit to 12bit shift register is considered.

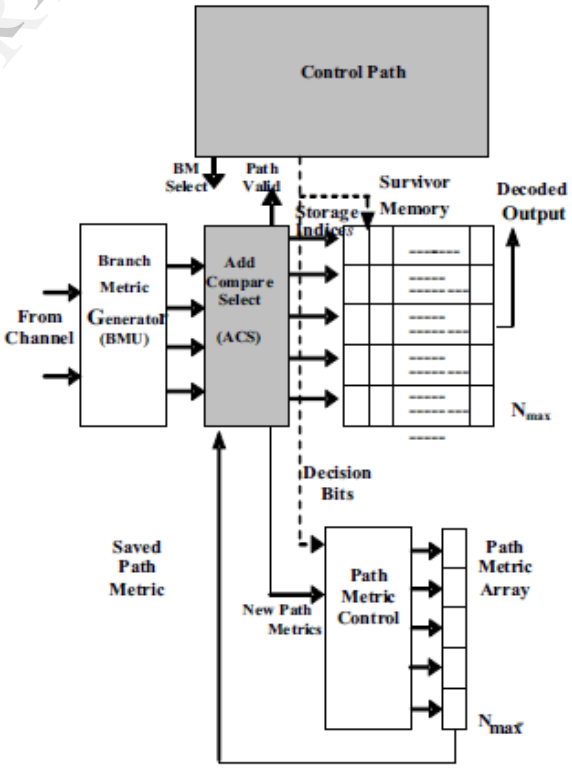

Fig 4: Adaptive Viterbi decoder architecture

[4]. A low probability of error in Viterbi decoder is presented in this paper using non polynomial approach. For BMU, it compares the received bits with expected bits. While designing the ACS unit, the

addition of path metric and then subtraction is done for purpose.TBU is designed with RAM. Fig.4 shows the Viterbi decoder based on non polynomial method.

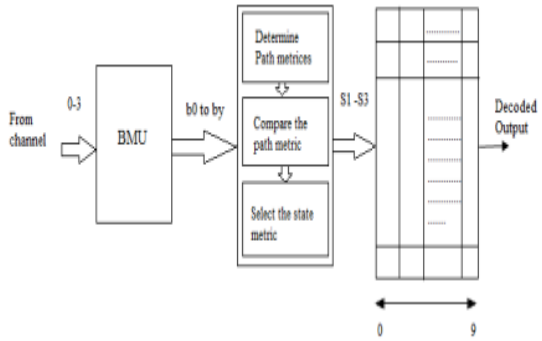

Fig.5: Architecture of the Viterbi decoder based on non polynomial approach

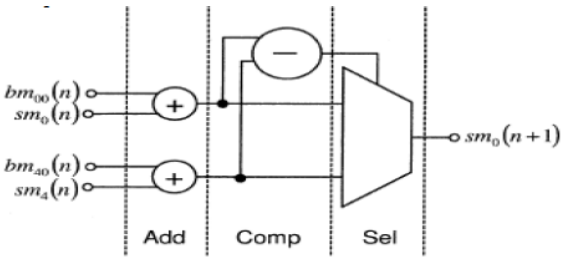

Fig.6: Internal structure of ACS unit

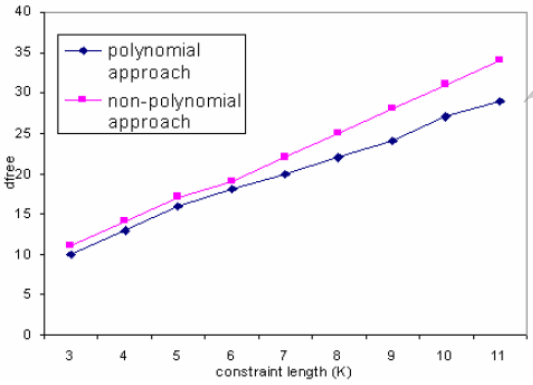

Fig.7:Variation of constraint length Vs dfree

III. DIFFRENCE IN TABULAR FORM

TABLE 3:Diffrence between various

Paper	**Slices**	**Power**	**Time**	**Technology used**
[1]	----	0.003 µw	15.500 ns	GDIL
[2]	116	----	5.556 ns	Reg exchange
[3]	6	62 mw	1.267 ns	Adaptive viterbi
[4]	The decoder is designed for less error probability in decoding bits			Non polynomial approach

IV. CONCLUSION

From the literature review it is clear that VLSI technology requires low power, less area and high speed constraints while implementing the decoder. Also various techniques of implementing the Viterbi decoder on FPGA are discussed by considering various factors.

V. REFRENCES

[1] D. Chakraborty, P. Raha, A. Bhattacharya, R. Dutta , " Speed optimization of a FPGA based modified Viterbi Decoder", International Conference on Computer Communication and Informatics (ICCCI -2013), Jan. 04 – 06, 2013, Coimbatore, INDIA.

[2] Zhenzhi Wu, Shujuan Hou, Hai Li,"A light-weighted Viterbi Decoder implemented by FPGA"IEEE, 2011

[3] Prof. Siddeeq Y. Ameen, Mohammed H. Al-Jammas, Ahmed S. Alenezi, "FPGA Implementation of Modified Architecture for Adaptive Viterbi Decoder" 2011

[4] C. Arun and V. Rajamani,"Design and VLSI implementation of a Low Probability of Error Viterbi decoder"IEEE, 2008

[5] Bernard Sklar, "Digital communication fundamentals & applications".