Design and Implementation of Analog Viterbi Decoder for Convolution Decoding Applications using 0.18µm Technology

Priya Maheshwari¹, Cyril P. Raj², S. Ajay Kumar³, P. Chandra Mohan⁴
1- M. Sc. [Engg.] Student, 2 - Assistant Professor and Course Manager (VSD), 3 - Senior Training Engineer, 4 - Senior Lecturer, VLSI System Design Centre, M.S. Ramaiah School of Advanced Studies, Bangalore 560 054

Abstract
With the continually increasing need for transmission of digital data over noisy channels, need for error control and correction techniques are also raising as well. There are various techniques and methods available for transmission of digital data over noisy channel but Viterbi Decoder is most efficient. It provides better coding gain at lower cost and good performance Convolutional encoding with Viterbi Decoding is a Forward Error Correction technique that is particularly suited to a channel in which the transmitted signal is corrupted mainly by Additive White Gaussian Noise (AWGN).

In this work, a software and hardware reference model of Current Mode Analog Viterbi Decoder is presented. The design is developed for rate ½, constraint length three and speed 50 Mbits/s. The Unit Adder block has reduced from three half adders to one half adder and two ex-or gates. This modification reduces power from 0.062772 mW to 0.036130 mW. Similarly, the PMRU block of design has reduced from three half subtractors to one half subtractor and two ex-or gates. The modified design reduces power from 0.021731 mW to 0.018481 mW.

The software reference model of the design is developed in MATLAB/Simulink. The hardware reference model of the design is developed in HSPICE, Virtuoso Schematic and Layout Editor. The power consumed by BMU block is 0.0171mW, PMRU block is 0.0166 mW and Unit Adder is 0.0199 mW in HSPICE. The area consumed by BMU block is 888.87 µm², PMRU block is 4135.61 µm² and Unit Adder is 5240.39µm² in Virtuoso Layout Editor. The complete design contains 7,500 transistors and 400329.93 µm² area.

Key Words: Convolution Code, Viterbi Decoding, Current Mode Analog Viterbi Decoder

Abbreviations
ACSU Add-Compare-Select Unit
AVD Analog Viterbi Decoder
AWGN Additive White Gaussian Noise
BMC Branch Metric Calculator
ECC Error Correcting Code
FEC Forward Error Correction
PMRU Path Metric Renormalization Unit
REM Register Exchange Method
SSM Storage Survivor Memory
TB Trace Back

1. INTRODUCTION
The reliable transmission of information over noisy channels is one of the basic requirements of digital information and communication systems. Because of this requirement, modern communication systems rely heavily on powerful channel coding methodologies. For practical applications these coding schemes do not only need to have good coding characteristics with respect to the capability of detecting or correcting errors introduced on the channel. They also have to be efficiently implementable, e.g. in digital hardware within integrated circuits. With the increase need for transmission of digital data over a wide variety of mediums, need for error control coding increases as well.

Forward Error Correction is a system of error control for data transmission by adding some redundant symbols to the transmitted information to facilitate error detection and error correction at receiver end. The addition of redundancy in coded messages implies the need for increased transmission bandwidth and also the system complexity [1][2][3]. Forward Error Correction (FEC) in digital communication system improves the error detection as well as error correction capability of the system at the cost of increased bandwidth and system complexity. Using FEC the need for retransmission of data can be avoided hence it is applied in situations where retransmissions are relatively costly or impossible.

FEC can be classified in two categories namely block codes and convolution codes. Block codes work on fixed size blocks of bits whereas convolution codes work on arbitrary length blocks of bits. The convolutional coder is often used in digital communication systems where the signal to noise ratio is very low. In this, the encoding operation may be viewed as discrete time convolution of input sequence with the impulse response of the encoder. “Convolutional” means the current output channel symbols are not only derived from the current input bits, but from a combinatorial logic that is fed by all the bits within a constraint length cycle. Standard convolutional codes denoted by (n,k,L) are usually described by two parameters: the code rate k/n and the constraint length L, in which k is the number of bits input into the convolutional encoder and n is the number of channel symbols output by the convolutional encoder. Constraint length L denotes the number of memory element used. The Convolutional encoder for rate ½ and constraint length 5 is shown in figure 1.

A convolutional encoder belongs to a class of devices known as finite-state machines where the encoder connections are characterized by generator polynomials. One can represent a convolutional encoder with a set of n generator polynomials, one for each of the n modulo-2 adders. There are several ways to...
represent an encoder such as – state diagram, tree diagram and trellis diagram [4][5][6]. State diagram is the time representation of the encoder states whereas the tree diagram adds the dimension of time to the state diagram. One generally prefers trellis representation over tree representation because at any level the number of nodes does not continuously increases as the number of incoming message bits increases.

**Fig. 1 Convolution encoder (rate 1/2, K=3) [2]**

Convolution encoding with Viterbi decoding is a FEC that is suited to a channel in which transmitted signal is corrupted by Additive White Gaussian Noise (AWGN). A Viterbi decoder uses Viterbi Algorithm for decoding a bit stream that has been encoded using (FEC) based on a convolution code. There are other algorithms available like Fano Algorithm but Viterbi Algorithm is widely used because it does the maximum likelihood decoding. Maximim. likelihood decoding means finding a code branch in code trellis that is most likely to be transmitted.

The six major sub-blocks of Viterbi decoder are FE/SH block, Branch Metric Calculation (BMC), Add-Compare-Select (ACS), Storage Survivor Memory (SSM) Path Metric Renormalization (PMR) and clock generator. The Viterbi decoder can be broadly classified in traditional (digital) and analog. In traditional Viterbi decoder, BMC, ACS and PMR unit are implemented in digital domain. The drawbacks of digital Viterbi decoder are requirement of ADC, high power consumption, speed reduction, increased area and high cost. An Analog Viterbi decoder is a Viterbi decoder in which all the metrics are computed and stored in analog form in ACS block[7][8][9]. Analog Viterbi decoders can be further classified into voltage mode, current mode and hybrid mode. Based on the Analog input each mode can be characterized as 1) Hard Decision Digital Decoder 2) 3-Bit Soft Decision Digital Decoder 3) Real Analog Decoder 4) Ideal Analog Decoder. This paper presents the hard decision current mode analog viterbi decoder for 50Mbits/s speed.
multiplexers and dual port memory containing all decision values through trace forward depth. The register exchange method needs same number of multiplexers and dual port memory as number of states multiplied by survivor path depth, and they are activated every cycle to update data in memory.

2. ANALOG VITERBI DECODER
SOFTWARE MODEL

In the initial phase of this work, a software reference model was developed to analyze the performance of various sub-blocks of the design [10][11]. The reference model consists of a message generator, convolution encoder and serial to parallel converter, BMC, PMR and ACS block. The model can process a sine wave generated at 50Mbits/s. The model is designed to support data rate ½ and constraint length 3. The model is designed to simulate a sine wave corrupted by AWGN. The snapshots of the complete design are shown in the figure 3.

3. ANALOG VITERBI DECODER
HARDWARE MODEL

The Hardware model of the Analog Viterbi Decoder was designed and implemented in HSPICE, Virtuoso Schematic Editor and Virtuoso Layout Editor. The hardware model is to be designed to produce maximum throughput, least latency and best clock period. The following sections discusses about the design and implementation of the hardware model of Analog Viterbi Decoder in HSPICE, Virtuoso Schematic Editor and Virtuoso Layout Editor.

3.1 Design of Hardware Model in HSPICE

The hardware model of BMU, PMR, Unit Adder, Comparator and ACS were designed to maximize the throughput, optimum operating frequency and provide flexibility for further hardware modifications. The BMU block input are received LSB bit, received MSB bit, expected LSB bit and expected MSB bit. The inputs of PMR block are constant LSB bit, constant MSB bit, previous stage output LSB bit and MSB bit. Figure 4 and 5 shows the screenshot of BMU and PMR block working in HSPICE.

3.2 Design of Hardware Model in Virtuoso
Schematic Editor

Unit Adder works on BM Unit output and PMR nit output. Its function is to add the branch metrics with their corresponding path metrics. The comparator works on Unit Adders output. It compares the output of two Unit Adders and provides minimum between them at output. Figure 6 and 7 show the screenshot of Unit Adder and Comparator block working in HSPICE.
PMR, unit adder, comparator and memory elements in it. Figure 8, 9 and 10 shows the schematics of BM Unit PMR Unit and ACS Unit.

Fig. 8 Screenshot of BM Unit from virtuoso schematic editor

Fig. 9 Screenshot of PMR Unit from virtuoso schematic editor

Fig. 10 Screenshot of ACS block from virtuoso schematic editor

Fig. 11 Screenshot of BMU layout
3.3 Design of Hardware Model in Virtuoso Layout Editor

The layout for BMU, PMR, unit adder and unit comparator block is developed and checked against all Design Rule Check rules [15][16]. The area and power consumed by each block is calculated and tabulated in Results section. The screenshot of BMU and PMR layouts are shown in figures 11 and 12.

4. RESULTS AND DISCUSSION

The Analog Viterbi Decoder along the generation of analog input is successfully designed in Matlab/Simulink. The results obtained from the design are as follows:

- **Speed**: The design operates at a maximum speed of ~50 Mbits/s.
- **Latency**: The latency of the system is \( N+2 \) clock cycles, where \( N \) is the total number of input message bits.
- **Throughput**: The system requires 1 clock cycle for processing each stage, additional 2 clock cycles, one for initial condition and one for final stage output.

<table>
<thead>
<tr>
<th>Block Name</th>
<th>Average Power</th>
<th>Max. Power</th>
<th>RMS Power</th>
</tr>
</thead>
<tbody>
<tr>
<td>BMU Unit</td>
<td>0.027mW</td>
<td>3.42mW</td>
<td>0.22mW</td>
</tr>
<tr>
<td>PMU Unit</td>
<td>0.195mW</td>
<td>4.10mW</td>
<td>0.20mW</td>
</tr>
<tr>
<td>Unit adder</td>
<td>0.056mW</td>
<td>4.29mW</td>
<td>0.29mW</td>
</tr>
<tr>
<td>Comparator</td>
<td>0.038mW</td>
<td>4.11mW</td>
<td>0.26mW</td>
</tr>
</tbody>
</table>

An Analog Viterbi Decoder is implemented in HSPICE, Virtuoso Schematic and Layout Editor. The results obtained in MATLAB/Simulink are cross verified against the results obtained in HSPICE and Schematic Editor. The power analysis is carried out in HSPICE for various sub-blocks and tabulated as in table 1.

<table>
<thead>
<tr>
<th>Block Name</th>
<th>Average Power</th>
<th>Area (µm²)</th>
</tr>
</thead>
<tbody>
<tr>
<td>BMU Unit</td>
<td>0.196mW</td>
<td>888.87</td>
</tr>
<tr>
<td>PMU Unit</td>
<td>0.196mW</td>
<td>4135.61</td>
</tr>
<tr>
<td>Unit adder</td>
<td>0.196mW</td>
<td>8842.60</td>
</tr>
<tr>
<td>Comparator</td>
<td>0.032mW</td>
<td>5240.39</td>
</tr>
</tbody>
</table>

5. CONCLUSION

The primary objective of this work was to simulate Analog Viterbi Decoder using software model and analyze the performance of the design. Based on the results obtained from the software model, a suitable hardware model was implemented. The conclusions derived from the work carried out are as follows:

- The software model of Analog Viterbi Decoder simulations has proved to be a promising approach for analyzing the performance in terms of operating speed, data rate and constraint length.
- The software model of the design helps to finding out the complexity of design before designing the hardware model.
- The software model approach also helps in generation of input data along with expected output of the Analog Viterbi Decoder for simulations the hardware model.
- The HSPICE model of design helps in understanding the design functionality at hardware level.
- The HSPICE model also helps in finding out total time delay and power consumed by the design.
while schematics help in finding out power and layouts in area calculation.

- The replacement of three half adders from one half adder and two ex-or gates in unit adder design reduces time delay from 0.099702 ns to 0.086490 ns and average power from 0.062772 EmW to 0.036130 mW.

- The modification in PMR block from three half subtractors to one half subtractor and two ex-or gates reduces time delay from 0.15544 ns to 0.14759 ns and average power from 0.021731 mW to 0.018481 mW.

REFERENCES


[8] Andreas Demosthenous and John Taylor *Current-mode Approaches to Implementing Hybrid Analogue Digital Viterbi Decoders*, ICECS ’96


