IOSR Journal of Electrical and Electronics Engineering (IOSR-JEEE)
e-ISSN: 2278-1676,p-ISSN: 2320-3331, Volume 6, Issue 3 (May. - Jun. 2013), PP 10-15
www.iosrjournals.org

Vlsi Implementation of Low Power Convolutional Coding With Viterbi Decoding Using Fsm

S.Gnanamurugan[1], A.Sindhu[2],

Assistant professor Department of ECE, Arunai college of Engineering[1], Tiruvannamalai
PG Scholar, Department of ECE, Arunai college of Engineering[2], Tiruvannamalai

Abstract: *Convolution Encoding with Viterbi Decoding is a powerful method for Forward Error correction and Detection. It has been deployed in many Wireless Communication Systems to improve the limited capacity of the Communication channels. Forward Error Correction techniques that is particularly suited to a channel in which the transmitted signal is corrupted mainly by Additive White Gaussian Noise. Convolution codes which are designed with state diagram method,Viterbi Decoder includes Branch Metric Unit, Add-Compare Select unit, Survivor-Path Memory involving choices of area and power. VLSI process technologies make it possible to realize one chip high speed encoder and decoders. Fixed set of parameters of 1/2 code rate with a constraint length of 3, encoder polynomials and trace back depth is complex. This complexity is reduced by different techniques with respect to reduction in area size increase the speed and low power consumption.*

***Keywords:** FEC, ConvolutionalEncoder, Viterbi Decoder.*

I. Introduction

Convolution coding is a popular error-correcting coding method used in digital communications. A message is convoluted, and then transmitted into a noisy channel. This convolution operation encodes some redundant information into the transmitted signal, thereby improving the data capacity of the channel. The Viterbi algorithm is a popular method used to decode convolutionally coded messages. The algorithm tracks down the most likely state sequences the encoder went through in encoding the message, and uses this information to determine the original message.

Instead of estimating a message based on each individual sample in the signal, the convolution encoding and Viterbi decoding process packages and encodes a message as a sequence, providing a level of correlation between each sample in the signal. As the convolution codes are used mostly for the channel encoding of data to achieve low-error-rate in latest wireless communication

Standards like3GPP, GSM and WLAN; the use of optimal decoding Viterbi algorithm will suffice. All communication channels are subject to the additive white Gaussian noise (AWGN) around the environment. The block codes can be applied only for the block of data where as convolution coding has can be applied to a continuous data stream as well as to blocks of data. Convolution Encoding with Viterbi decoding is a powerful FEC technique that is particularly suited to a channel in which the Transmitted signal is corrupted mainly by AWGN. It operates on data stream and has memory that uses previous bits to encode.

FEC techniques are used in the transmitter to encode the data stream and receiver to detect and correct bits in errors, hence minimize the bit error rate (BER) to improve the performance. RS decoding algorithm complexity is relatively low and can be implemented in hardware at very high data rates. It seems to be an ideal code attributes for any application. However, RS codes perform very poorly in AWGN channel.

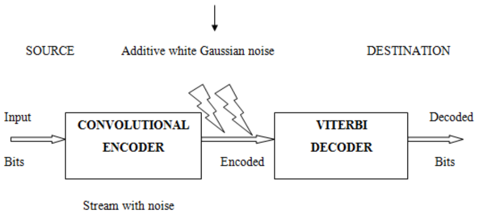


Fig.1.1 Digital Communication system

Convolution encoding with Viterbi decoding is a powerful FEC technique that is Particularly suited to a channel in which the transmitted signal is corrupted mainly by AWGN. It operates on data stream and has memory that uses previous bits to encode. It is simple and has good performance with low implementation cost. The Viterbi algorithm (VA) is used for decoding a bit stream that has been encoded using FEC code. The convolution encoder adds redundancy to a continuous stream of input data by using a linear shift register. The convolution Encoder and Viterbi Decoder used in the Digital Communications System is shown in Fig.1.1.

II. Forward Error Correction

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 message simples the need for increased transmission bandwidth and also the system complexity .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 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 where as convolution codes work on arbitrary length blocks of bits. The convolution 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.

Error detection and correction or error control is techniques that enable reliable delivery of digital data over unreliable communication channels. Many communication channels are subject to channel noise, and thus errors may be introduced during transmission from the source to a receiver. Error detection techniques allow detecting such errors, while error correction enables reconstruction of the original data.

III. Convolutional Encoder

The convolution encoder consists of a shift register, which shifts in a number of the bits from I at a time, and then produces a set of output bits based on logical operations carried out on parts of I in the register memory. This process is often referred to as convolution encoding. The encoder introduces redundancy into the output code, producing more output bits than input bits shifted into its memory. As a bit is shifted along the register it becomes part of other output symbols sent. Thus the present output bit that is observed by the VA has information about previous bits in I, so that if one of these symbols becomes corrupted then the VA can still decode the original bits in I by using information from the previous and subsequent observation symbols.
A diagram of the convolution encoder used shown in Figure 3.1. It is assumed here that the shift register only shifts in one bit at a time and outputs two bits, though other combinations of input to output bits are possible.

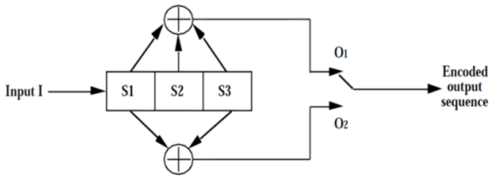


Fig.3.1: Convolution Encoder

At the initial point the shift registers in the convolution encoder are at reset position which is all-zero content. As illustrated in Fig2.Whenever a data bit enters from the input port, encoder produces two encoded bits and both of the encoded bits are correlated with instant and K-1 number of previous input bits, where K refers to the constraint length of the convolution encoder.

IV. State Diagram

State diagram is the graphical way to show state table. In the state diagram bubbles are the states and the indicators of the states are written inside the bubbles. Arrows are the state transitions according to input values. The value on the arrows corresponds to the output of the encoder while the transition takes place w ith respect to the input value as shown in fig 4.1.

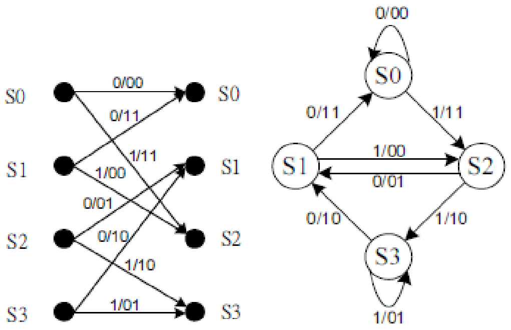


Fig.4.1: State Diagram

An input sequence I, of 0 1 1 0 0 0 is to be transmitted across the BSC, using the convolution encoder described above, then the output obtained from the encoder will be 00 11 01 01 11 00, as shown in Transition Table 4.2.

INPUT I	S1	S2	S3	O1	O2
0	0	0	0	0	0
1	1	0	0	1	1
1	1	1	0	0	1
0	0	1	1	0	1
0	0	0	1	1	1
0	0	0	0	0	0

TABLE 4.2 Transition Table

V. Encoder Output Sequence

The output is termed as the Encoder Output Sequence (EOS).shows the corresponding contents of each memory element of the shift register, where each element is assumed to be initialized to zero's at the start of encoding. As the EOS is constructed by the encoder, the part of the EOS already formed is transmitted across the channel. At the receiving end of the channel the following noisy sequence of bits may be received, 01 11 01 00 11 00. As can be seen there are two bit errors in this sequence, the 00 at the beginning has changed to 01, and similarly the fourth symbol has changed to 00 from 01. It is the job of the Viterbi Algorithm to find the most likely set of states visited by the original FSM and thus determine the original input sequence.

VI. Viterbi Decoder

The receiver can deliver either hard or soft symbols to the Viterbi decoder. A hard symbol is equivalent to a binary +/-1. A soft symbol, on the other hand, is multileveled to represent the confidence in the bit being positive or negative. For instance, if the channel is non-fading and Gaussian, the output of the matched filter quantified to a given number of bits is a suitable soft input. In both cases, 0 is used to represent a punctured bit.

The basic units of viterbi decoder are branch metrics, Add compare select and Survivor management unit. Figure5.1.shows the general structure of a Viterbi decoder. The quality of Viterbi decoder design is mainly measured by three criteria.

- Coding gain
- Throughput
- Power dissipation.

It consist of three blocks: the branch metric unit (BMU), which computes metrics, the add–compare–select unit (ACSU), which selects the survivor paths for each trellis state, also finds the minimum path metric of the survivor paths and the survivor management unit (SMU), that is responsible for selecting the output based on the minimum path metric.

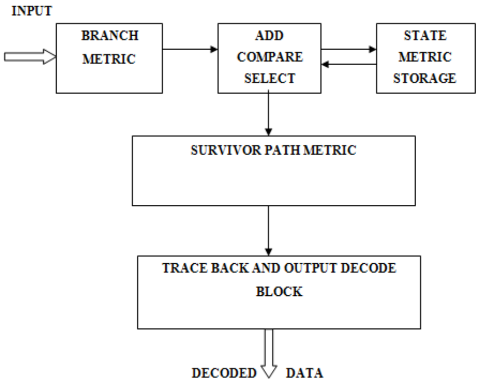


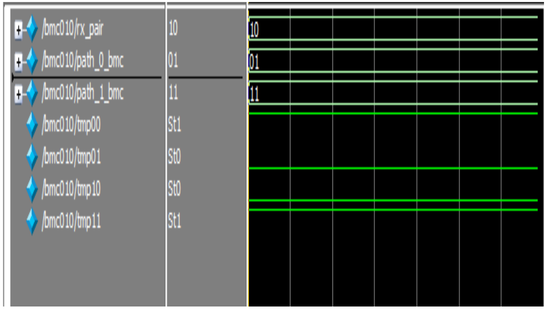
Fig.5.1: Structure of a Viterbi Decoder

VII. Viterbi Algorithm

The Viterbi algorithm was originally invented to detect convolutionally encoded data symbols[2].The Algorithm (VA) finds a maximum like hood (ML) estimate of a code sequence c from the corresponding received sequence r by maximizing the probability(r|c) that sequence is received conditioned on the estimated code sequence must be a valid coded sequence. The Viterbi algorithm utilizes the trellis diagram to compute path metrics. The channel is assumed to be memory less, i.e. the noise sample affecting a received bit is independent from the noise sample affecting the other bits. The decoding operation starts from state '00', i.e. with the assumption that the initial state of the encoder is '00'.With receipt of one noisy code word, the decoding operation progresses by one step deeper in to the trellis diagram. In the process, the 'accumulated path metric' is updated by adding the metric of the incoming branch with the 'accumulated path metric' of the state from where the branch originated. No decision about a received codeword is taken such operations and the decoding decision is deliberately delayed to reduce the possibility of erroneous decision.

VIII. Simulation Results

BRANCH METRIC UNIT STATE-10(0)

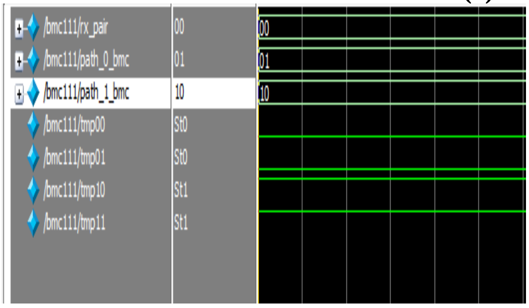


```
/bmc010/rx_pair      10
/bmc010/path_0_bmc   01
/bmc010/path_1_bmc   11
/bmc010/tmp00        St1
/bmc010/tmp01        St0
/bmc010/tmp10        St0
/bmc010/tmp11        St1
```

FIG.8.1: BMU STATE 0

BMU at the state 10 Path_0 (01), Path_1 (11), BMP 00-1, 01-0, 10-0, 11-1.

BRANCH METRIC UNIT STATE-11(1)

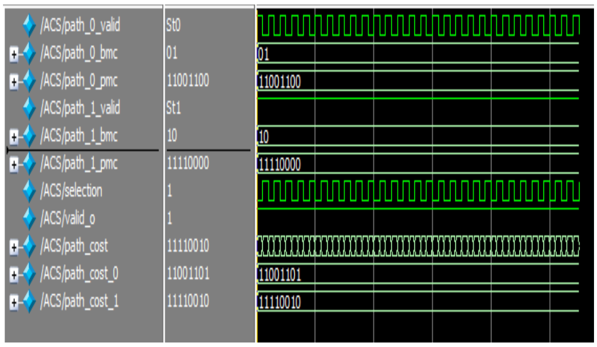


```
/bmc111/rx_pair      00
/bmc111/path_0_bmc   01
/bmc111/path_1_bmc   10
/bmc111/tmp00        St0
/bmc111/tmp01        St0
/bmc111/tmp10        St1
/bmc111/tmp11        St1
```

FIG.8.2: BMU STATE 1

BMU at the state 11 Path_0 (01), Path_1 (10), BMP 00-0, 01-0, 10-1, 11-1.

ADD-COMPPARE SELECT UNIT



```
/ACS/path_0_valid    St0
/ACS/path_0_bmc      01
/ACS/path_0_pmc      11001100
/ACS/path_1_valid    St1
/ACS/path_1_bmc      10
/ACS/path_1_pmc      11110000
/ACS/selection       1
/ACS/valid_o         1
/ACS/path_cost       11110010
/ACS/path_cost_0     11001101
/ACS/path_cost_1     11110010
```

FIG.8.3: ACSU

ACSU at the state 0 (01) Path_0 11001100 at the stae 1(10) Path_1 11001101.

TRACE BACK UNIT

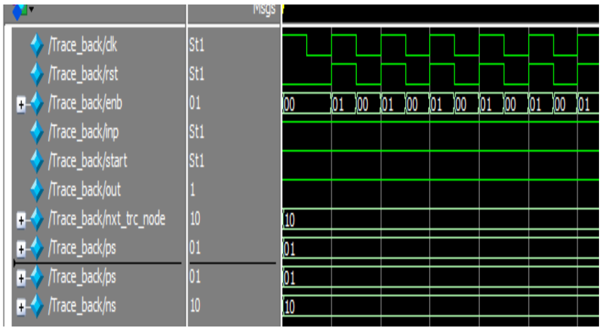


FIG.8.4:TBU

TBU Clk-1,Rst-0,Enable-00,Present State-01,Next State-10.

VITERBI TX_RX TOP MODULE

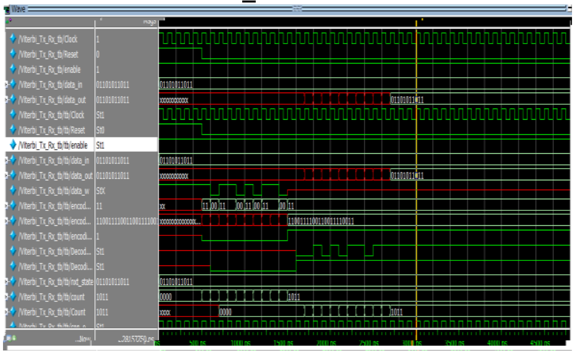

Wave
Msgs
/Viterbi_Tx_Rx_tb/Clock 1
/Viterbi_Tx_Rx_tb/Reset 0
/Viterbi_Tx_Rx_tb/enable 1
/Viterbi_Tx_Rx_tb/data_in 01101011011
/Viterbi_Tx_Rx_tb/data_out 01101011011
/Viterbi_Tx_Rx_tb/tb/Clock St1
/Viterbi_Tx_Rx_tb/tb/Reset St0
/Viterbi_Tx_Rx_tb/tb/enable St1
/Viterbi_Tx_Rx_tb/tb/data_in 01101011011
/Viterbi_Tx_Rx_tb/tb/data_out 01101011011
/Viterbi_Tx_Rx_tb/tb/data_w StX
/Viterbi_Tx_Rx_tb/tb/encod... 11
/Viterbi_Tx_Rx_tb/tb/encod... 1100111100110011110011
/Viterbi_Tx_Rx_tb/tb/encodi... 1
/Viterbi_Tx_Rx_tb/tb/Decod... St1
/Viterbi_Tx_Rx_tb/tb/Decodi... St1
/Viterbi_Tx_Rx_tb/tb/nxt_state 01101011011
/Viterbi_Tx_Rx_tb/tb/count 1011
/Viterbi_Tx_Rx_tb/tb/Count 1011
Now
01101011011
xxxxxxxxxx
01101011011
11 00 11 00 11 00 11 00 11
1100111100110011110011
0000
1011
xxxx
0000
1011
500 ns
1000 ns
1500 ns
2000 ns
2500 ns
3000 ns
3500 ns
4000 ns
4500 ns

FIG.8.5: VITERBI TX_RX

Simulation results show input **0110101011011** is encoded using **Finite State machines.** It generate the data to be transmitted through the channel-result is binary data bits convolutionally encode the data-result is channel symbols. After all of the inputs have been presented to the encoder,the output sequence

11 00 11 11 00 11 00 11 11 00 11

Viterbi decoder receives a bit stream containing information about the reliability of each received symbol.Trace-Back unit restores an (almost) maximum-likelihood path from the decisions made by PMU. Since it does it in inverse direction, a viterbi decoder comprises a FILO (first-in-last-out) buffer to reconstruct a correct order.

IX. Conclusion

Viterbi Algorithm allows safe data transmission via error correction and original message can be recovered accurately without any noise. It was concluded from that if trace back is started after going deeper into trellis diagram then more accurate data can be achieved but it results in complex hardware design and latency in the received signal. Viterbi algorithm of any rate can be designed using same basic principles and techniques.

References

[1] Jinjin He,Huaping Liu"High-speed Low-power Viterbi Decoder design for TCM Decoders"IEEE Trans. VLSI. vol 20,Apr 2012.

[2] P.Subhashini, D.R.Mahesh Varma, Implementation "Analysis of adaptive Viterbi Decoder for High Speed Applications" International Journal of Computer Applications (0975 – 8887) Volume 31– No.2, October 2011.

[3] Anubhuti Khare, Manish Saxena, Jagdish Patel, "FPGA Based Efficient Implementation of Viterbi Decoder" International Journal of Engineering and Advanced Technology (IJEAT)ISSN: 2249 – 8958, Volume-1, Issue-1, October 2011.

[4] J. He, Z. Wang, and H. Liu, "An efficient 4-D 8PSK TCM decoder architecture," IEEE Trans. Very Large Scale Integer. (VLSI) Syst., vol.18,no.5,May 2010.

[5] J. He, H. Liu, and Z. Wang, "A fast ACSU architecture for viterbi decoder using T - algorithm," in Proc. 43rd IEEE Asilomar Conf. Signals,Syst. Comput., Nov. 2009.

[6] R. A. Abdullah and N. R. Shanbhag, "Error-resilient low-power viterbi decoder architectures," IEEE Trans. Signal Process., vol. 57, no. 12,pp. 4906–4917, Dec. 2009.

[7] . Jin and C.-Y. Tsui, "Low-power limited-search parallel state viterbi decoder implementation based on scarece state transition," IEEE Trans.Very Large Scale Integer.(VLSI) Syst., vol. 15, no. 11, pp. 1172–1176,Oct. 2007.

[8] F. Sun and T. Zhang, "Low power state-parallel relaxed adaptive viterbi decoder design and implementation," in Proc. IEEE ISCAS, May 2006,pp. 4811–4814.

[9] "Bandwidth-efficient modulations," Consultative Committee For Space Data System, Matera, Italy, CCSDS 401(3.3.6) Green Book, Issue 1, Apr. 2003.

[10] F. Chan and D. Haccoun, "Adaptive viterbi decoding of convolutional codes over memory less channels," IEEE Trans. Commun., vol. 45, no.11, pp. 1389–1400, Nov. 1997.

[11] J. B. Anderson and E. Offer, "Reduced-state sequence detection with convolutional codes," IEEE Trans. Inf. Theory, vol. 40, no. 3, pp.965–972, May 1994.

[12] S. J. Simmons, "Breadth-first trellis decoding with adaptive effort,"IEEE Trans. Commun., vol. 38, no. 1, pp. 3–12, Jan. 1990,

[13] C. F. Lin and J. B. Anderson, "M-algorithm decoding of channel convolutional codes, "presented at the Princeton Conf. Info. Sci. Syst., Princeton, NJ, Mar. 1986.

Online References:

1. http://en.wikipedia.org/wiki/Viterbi_algorithm

2. http://www.1core.com/library/comm/viterbi/
viterbi.pdf

3.http://web.aanet.com.au/~ospiropo/tech/viterbi/TIC1Tutorial_Viterbi.pdf.

Text Book Reference:

[1]Simon Haykins, "Communication Systems".