

# A LOW POWER VITERBI DECODER DESIGN WITH MINIMUM TRANSITION HYBRID REGISTER EXCHANGE PROCESSING FOR WIRELESS APPLICATIONS

S. L. Haridas<sup>1</sup> and Dr. N. K. Choudhari<sup>2</sup>

<sup>1</sup>Prof. & Head of E&T Engg., B. D. College of Engg., Sevagram (M.S.), India  
*slhleec@rediffmail.com*

<sup>2</sup>Principal, Smt. Bhagvati Chaturvedi College of Engg., Nagpur (M.S.), India  
*drnitinchoudhari@gmail.com*

## **ABSTRACT**

*This work proposes the low power implementation of Viterbi Decoder. Majority of viterbi decoder designs in the past use simple Register Exchange or Traceback method to achieve very high speed and low power decoding respectively, but it suffers from both complex routing and high switching activity. Here simplification is made in survivor memory unit by storing only m-1 bits to identify previous state in the survivor path, and by assigning m-1 registers to decision vectors. This approach eliminates unnecessary shift operations. Also for storing the decoded data only half memory is required than register exchange method. In this paper Hybrid approach that combines both Traceback and Register Exchange schemes has been applied to the viterbi decoder design. By using distance properties of encoder we further modified to minimum transition hybrid register exchange method. It leads to lower dynamic power consumption because of lower switching activity. Dynamic power estimation obtained through gate level simulation indicates that the proposed design reduces the power dissipation of a conventional viterbi decoder design by 30%.*

## **KEYWORDS**

*Traceback method, Register Exchange method, Hybrid Register Exchange method, Minimum Transition Register Exchange Method.*

## **1. INTRODUCTION**

The Viterbi Algorithm [1], can find the most likely sequence of state transitions, and has been widely used in digital communication and signal processing, was introduced in 1967 as a method of decoding convolutional codes [2]. In digital communication systems, convolutional coding and viterbi decoding are used to enhance the performance of the systems. In recent years due to battery operated devices, the design of viterbi decoder has been investigated intensively.

Numerous techniques for reducing power dissipation have been proposed. Scarce state transition architecture [3] has been used to reduce switching activity which incorporates a predecoder and preencoder to minimize transitions at the input of viterbi decoder. This reduces dynamic power dissipation. Kang and Wilson [4] suggesting partitioning major blocks at the system level. High throughput rate can be achieved using the concept of strongly connected trellis which was introduced by Chang and Yao [5]. Two decoding methods are mainly used, Register exchange method (REM) and Traceback method (TBM). The TBM is more popular since used for large number of states and constraint length, therefore continuous modification has been take place in TBM. While REM is used for small number of states and small constraint length. Very few attempts to combine the advantages of TBM and REM are takes place in [6] and [7] where they

proposed Hybrid Traceback and Hybrid Register exchange (HREM) methods. In HREM, reduces memory, reduces number of data transfer operations and ultimately the switching activity. In this paper we present the design of low power viterbi decoder in the IS-95 CDMA environment. We used minimum transition hybrid register exchange method (MTHREM) [8], which further reduces the switching activity by minimizing the data transfer operations using a free distance property of encoder.

The following sections are organised as follows. We first briefly review the viterbi algorithm with all components in section 2. In section 3 explains the system architecture. Section 4 devotes to various decoding schemes, while in section 5 we proposed detail implementation of MTHREM with free distance and error correcting capability of viterbi decoder and few low power techniques. Finally in section 6 remaining design steps and results of the actual implementation are reported.

## 2. VITERBI ALGORITHM

The Viterbi algorithm is an optimal algorithm for estimating the state sequence of a finite state process observed in the presence of memory less noise.

### 2.1 Convolutional Encoder

A Viterbi decoder and a convolutional encoder operate by finding the most likely decoding sequences for an input code symbol stream. A convolutional encoder is selected for error-correction with digital mobile communication. We consider a convolutional code (3, 1, 2) which has a code rate  $r$  of  $1/3$  and constraint length  $K$  of 3 and number of memory element  $m$  is 2, depicted in figure 1. The encoder shown contains a shift registers tapped at various positions. The shift register taps terminate at modulo-2 adders forming generator function ( $g_2 = 101$ ,  $g_1 = 111$ ,  $g_0 = 111$ ). Input bit enter the shift register by one bit at a time. The outputs of the generator functions produce an output of three symbols for each input bit, which corresponds to a code rate  $1/3$ . As an example of how the encoding process operates, consider the binary sequence 110110. The resulting output sequence of the encoder will be 111, 100, 100, 000, 100 and 100.

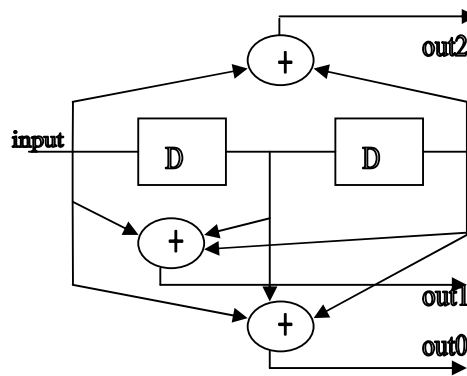


Figure 1. A (3, 1, 2) convolutional encoder

### 2.2 Trellis Diagram

Since there are  $2^m = 4$  states in the encoder, we have four states, e.g.,  $S_0 = (00)$ ,  $S_1 = (01)$ ,  $S_2 = (10)$ , and  $S_3 = (11)$ . The trellis diagram of figure 1 is depicted in figure 2, where the solid line represents the transition branch when the input bit is 0, whereas the dashed line represents the transition branch when the input bit is 1, and the two bits marked on the transition branch

denotes the output codeword. For all convolution encoders, there are two possible branches entering each node, and two possible branches leaving the node, regardless of the code rate.

If we assume a code sequence  $\{c_0, c_1, c_2, \dots\}$  is transmitted over a channel and  $\{r_0, r_1, r_2, \dots\}$  is the received sequence then the branch metric  $S_i$  at time  $t-1$  to state  $S_j$  at time  $t$  is defined by []

$$BM = d_{t-1,t}(S_i, S_j) = \|r_{t-1} - c_{i,j}\|$$

Where  $\|r_{t-1} - c_{i,j}\|$  denotes the hamming distance between received word and transmitted code word of the transition branch. The path metric  $PM_j$  of state  $S_j$  at time  $t$  can be expressed as

$$PM_j^t = \min\{PM_i^{t-1} + d_{t-1,t}(S_i, S_j)\}$$

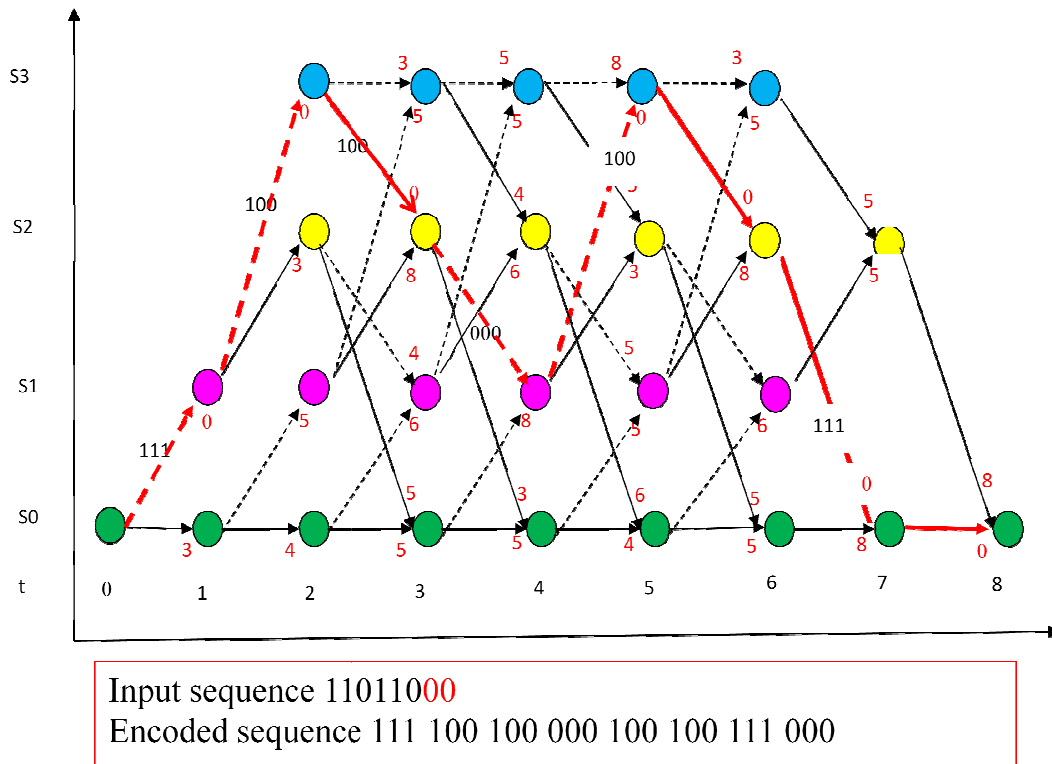


Figure 2. Trellis diagram of Encoder

### 2.3 Viterbi Decoding Algorithm

We can view the Viterbi algorithm as a dynamic programming algorithm for finding the shortest path through a trellis, and the algorithm can be broken down into the following three steps.

1. Calculate the branch metrics by comparing the received code symbol and expected symbols.
2. Recursively computes the shortest paths to time  $n$ , in terms of the shortest paths to time  $n-1$ . In this step, decisions are used to recursively update the survivor path of the signal. This is known as add-compare-select (ACS) recursion.
3. Recursively find the shortest path leading to each trellis state using the decisions from Step 2. The shortest path is called the survivor path for that state and the process is referred to as survivor path decode. Finally, if all survivor paths are traced back in time, they merge into a unique path, which is the most likely signal path that we are trying to find.

From the Fig. 3, associated with each trellis state  $S$  at time  $t$  is a state metric  $SM_p$  and  $SM_q$  which is the accumulated metric along the shortest path leading to that state. The state metrics at time  $t$  can be recursively calculated in terms of the state metrics of the previous iteration as follows:

For state metric  $SM_t$

$$PM^p = \min (PM^i + BM^{i1}, PM^j + BM^{j1})$$

$$PM^q = \min (PM^i + BM^{i0}, PM^j + BM^{j0})$$

In order that the input sequence can be decoded, the survivor path (shortest path) or signal through the trellis must be traced and decode.

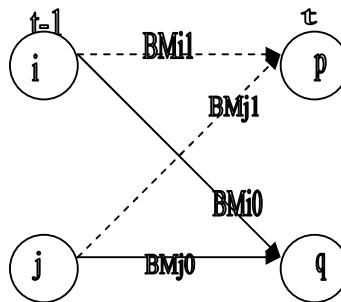


Figure 3. Butterfly structure of encoder

### 3. SYSTEM ARCHITECTURE

The implementation of the viterbi algorithm, referred to as the viterbi decoder. The major blocks of a viterbi decoder are shown in Fig. 4. The role of each block is described briefly below.

**Branch Metric Unit:** This unit computes the branch metric of each transition, which is a hamming distance between the received symbol and expected symbol.

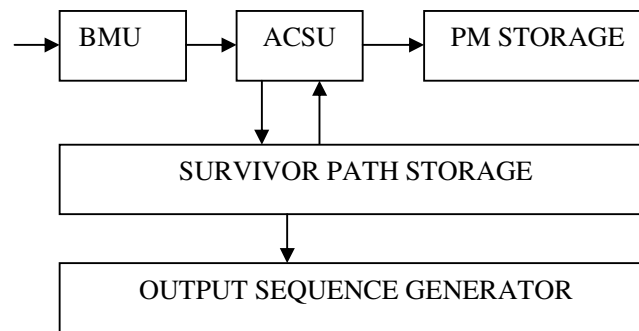


Figure 4. Block Diagram of Viterbi decoder

**ACSU:** The Add-compare-select unit is a collection of ACS units. For a single state, an ACS unit receives two branch metric and two path metrics. It adds each incoming branch metric to the corresponding path metric and compares the two results to select a smaller one. The path metric of the state is updated with the selected one.

**PM storage:** This unit stores the path metric of each state at the current stage.

**Survivor path storage:** It stores the survivor sequence of each state selected by the ACSU.

**Output generator:** It generates the decoded output sequence. In the traceback method, this block incorporates combinational logic, which traces back along the survivor path starting from

the final state to the initial state and generates the decoded output. For the register exchange method the decoded output is the content of the register of the final state.

In this paper, we consider the 16 state, rate 1/3 convolutional encoder. We limit the number of symbols to 20 for our Viterbi decoder. The tail sequence in each frame is fixed to '0' and resets the convolutional encoder to its initial state.

## 4. DECODING METHODS

### 4.1 Register Exchange Method

In Register Exchange Method, Figure 5, a register is assigned to each state contains information bits for the survivor path throughout the trellis. The register keeps the partially decoded output sequence along the path. The register exchange method eliminates the need to traceback since the register of final state contains the decoded output. This approach results in complex hardware and high switching activity due to the need to copy the content of all the registers from state to state.

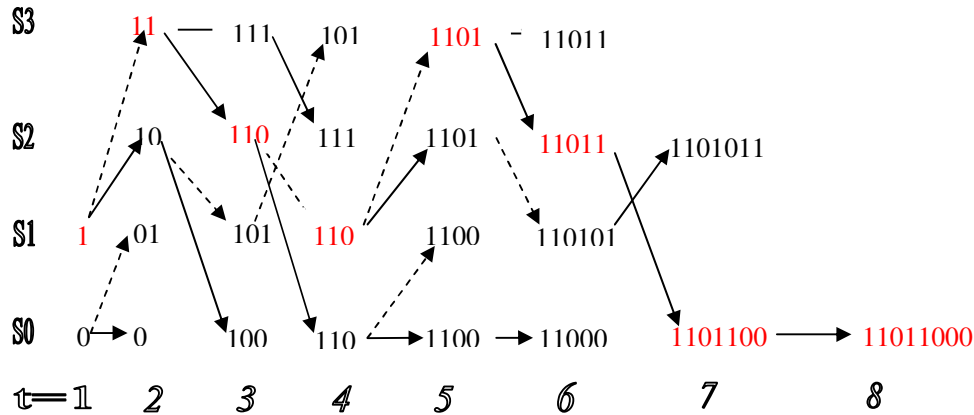


Figure 5. Register exchange method

### 4.2 Hybrid Register Exchange Method

In this method register exchange and traceback method is combined, therefore the name Hybrid register exchange method [7], which reduces further the switching activity and power. Here we are using a property of trellis is that, if we go forward for  $m$  cycles then the data bits will be the corresponding state bits irrespective of the initial state from where the data gets transferred. To find the initial state we have to traceback through an  $m$  cycles by observing the survivor memory. And then transfer the partial decoded data from initial state to the next state which is  $m$  cycle later and not a subsequent cycle. Again consider the previous (3, 1, 2) encoder and trellis diagram as shown in figure 1 and figure 2 respectively. Now if the trellis is strongly connected, then the states on survivor path will correspond to the input bits. We define survivor state as a state at time  $t_1$  that is traced from initial state  $S_i$  at time  $t_2$  ( $t_2 > t_1$ ). It is noted that survivor path memory is the output of ACSU at time  $t$ . For instance, in fig. 6 the survivor state at time  $t = 2$ , traced from  $S_3$  at time  $t = 4$  is 10 i.e.  $S_2$ .

$$SUR(S_3;2,4) = 10,$$

$$SUR(S_2;2,4) = 11$$

$$SUR(S_1;2,4) = 11$$

$$SUR(S_0;2,4) = 11$$

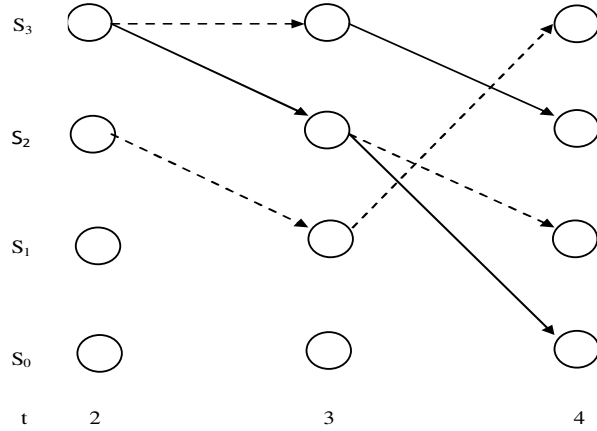


Figure 6. Survivor states for t = 2 to 4

The corresponding nature of lines indicates the corresponding data inputs which are equivalent to the current state. For if current state S3, the survivor state is S2 and data input is 11. So the previous data from state S2 will get transfer to state S3 and new data is added as 11. Here in the Fig. 7 survivor state at t = 4 is 11 for state S1 therefore the content of S1 state register at t = 4 is the data of S3 at t = 2 and the state S1 01 itself i.e. 1101. And at t = 6 the survivor information of state S2 is 01, therefore the content of S2 state register at t = 6 is the data of S1 at t = 4 and the state S2 itself i.e. 110110. This process goes continue, the final state register of S0 contains the decoded output. The memory operation is not at every cycle, and it gets reduced by a factor of m. Also the shifting of data from one register to another is reduced that is the switching activity will reduce.

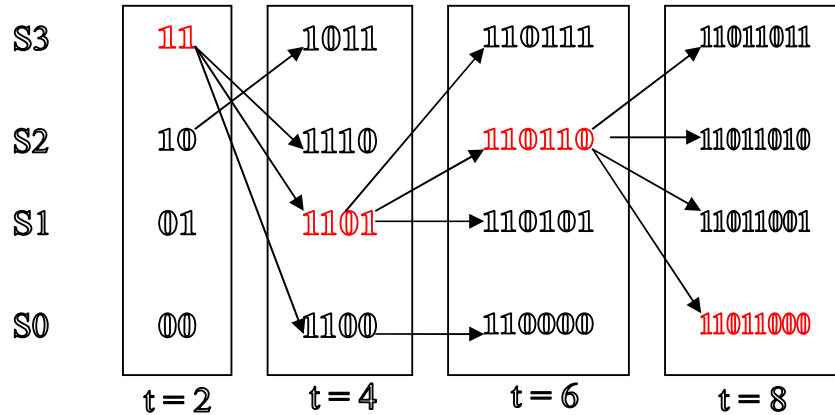


Figure 7. Hybrid Register Exchange Method

### 5. PROPOSED LOW POWER DESIGN

From the state diagram we can obtain a closed form expression whose expansion yields all distant information directly. In figure 8, labeling the branches  $D^2$ ,  $D$  and  $D^0 = 1$ , where the exponent corresponds to the distance of particular branch from the corresponding branch of all zero path. Also we split the node  $S_0$ , since circulation around this self loop simply corresponds to branches of the all zero path. Adding branch exponents we see that path  $S_0-S_1-S_2-S_0$  is at distance 8 from the correct path, and  $S_0-S_1-S_3-S_2-S_0$  and  $S_0-S_1-S_2-S_1-S_2-S_0$  has distance 8 and 10 respectively.  $S_0-S_1-S_2-S_0$  Differ in one input bit from all zero paths and other are two and more.

So the free distance will be 8. This implies that three bit of errors over the channel can be corrected. For three or fewer errors will cause the received sequence to be at most distance three from transmitted sequence but it will be at least, distance 5 from any other possible code sequence.

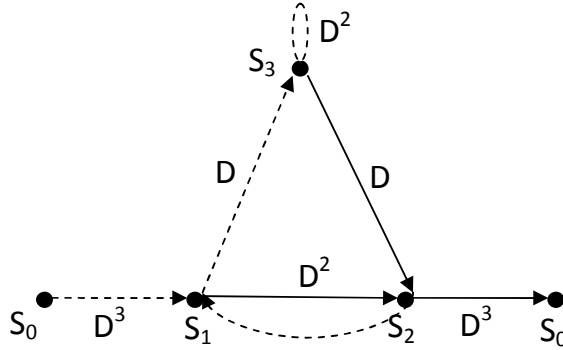


Figure 8. Distance properties of encoder

### 5.1 MINIMUM TRANSITION HYBRID REGISTER EXCHANGE METHOD

In Register Exchange method the data has to be transfer from one register ( $2^m$ ) to other at every instant, and therefore the switching activity is much more. While in Hybrid register exchange method data transfer is taking place at alternate instant of time therefore switching activity is less. One of the solutions to reduce the switching activity further is to avoid unwanted data transfer from one register to other at alternate time interval. Since the path metric at every time interval describes the how much bits they are differ from the transmitted sequence. In our fig. 5 the metric at time  $t = 8$  for final state  $S_0$  is '0'; since the transmitted and receive sequence is same. Every decoder has error correcting capability depending on free distance. Here for our encoder free distance are 8 and therefore error correction ability is three. The path metric less than or equal to three leads to correct path and the state which has path metric more than three may lead to wrong detection. For the encoder maximum half the state has less or less than three path metrics and half the state has path metric more than three. Hence there is absolute no need to trace the path which has higher path metric than three. So memory requirement will get reduced. Earlier for register exchange and hybrid register exchange method the  $2^m$  registers are required to save decoded data while in this approach it reduces to  $2^m/2$ , so the data transfer operations will reduces and ultimately the switching activity will get reduced. Therefore as shown in Figure 9, we can avoid these states or path from calculation and data transfer, so that the data bit transitions could not takes place and avoid undesirable switching.

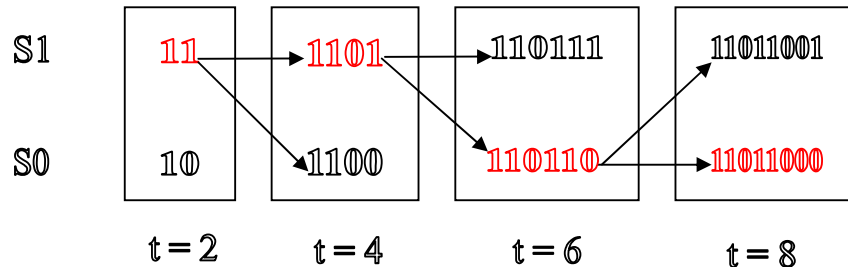


Figure 9. Minimum Transition Hybrid Register Exchange Method

**Clock gating:** Here register holds the survivor path information of all states. The five bit counter keeps the track of the current stage. When the code symbol is received, the particular register clock is enabled, and the survivor path information of that stage is recorded in that register. All other registers hold their state, since the clock of other registers is disabled. Therefore the survivor path storage reduces switching activity.

**Toggle-filtering:** If signals arriving at the inputs of a combinational block at different times, the block may go through several intermediate transitions before it settle down. By blocking early signals until all input signals arrive, intermediate transitions and ultimately dynamic power dissipation can be reduced. In the viterbi decoder, when trace back path is constructed information of this path is read from the survivor memory, and then fed to the output generator block.

## 6. EXPERIMENTAL RESULT

We measured the power dissipation of the three different implementations of rate 1/3, constraint length 5 (scale down version), viterbi decoder; the register exchange, hybrid register exchange; and the proposed low power minimum transition hybrid register exchange method.

Our proposed design, viterbi decoder using Minimum Transition Hybrid Register Exchange Method, is coded in VHDL and synthesized with synopsys tools using the TSMC 150 nm technology library for which supply voltage is 1.65V. Circuit is simulated with a representative set of input vectors. Transitions in different nodes are recorded in an output file which is read by synopsys power estimation tool that is power compiler. This information besides the power attributes of each gate which is extracted from synopsys library is used to estimate power consumption of the circuit.

Experimental results of our design and some references are shown in Table I. All references have the same configuration of viterbi decoder (rate = 1/3, K = 5) and they are recently only available, therefore we are considering here. Our design that is viterbi decoder using Minimum Transition Hybrid Register Exchange Method has require less power than the register exchange and Hybrid register exchange designs. The proposed method reduces the power dissipation by 30% than our designed register exchange method, 15% than our designed hybrid register exchange method, and 73% and 53.75% compared with the design mentioned in [9] and [10] respectively.

Table I. Result of power measurement

Method/References	Our REM	Our HREM	[9]	[10]	Proposed design
Power dissipation ( $\mu$ W)	264.79	214.97	683	400	185.84

## 7. CONCLUSION

Viterbi decoders employed in digital wireless communication are complex and dissipate high power. In this paper, a novel decoding scheme based on minimum transition hybrid register exchange approach has been implemented for the viterbi decoder of rate 1/3 and constraint length of 5. We achieve this low power implementation by algorithmic modification with some low power design techniques. Resulting architecture will lead to low power consumption among the entire configuration available. The same technique can be applicable to higher constraint length viterbi decoder such as K = 7, 9... Table I compares the proposed design with other implementation shown in the literature. Also the memory requirement for decoding is reduced



by half. So this is one of the alternatives for register exchange method for decoding in viterbi algorithm for low power applications.

## REFERENCES

- [1] A. Viterbi, "Error bounds for convolutional codes and asymptotically optimum decoding algorithm" *IEEE Trans. Info. Theory*, Vol. IT 13, no.2, pp 260-269, Apr 1967.
- [2] G. Forney, "The Viterbi Algorithm" *Proc. IEEE*, vol61, no. 3, pp 268-278, Mar. 1973.
- [3] K. Seki; S. Kubota; "Very Low Power Consumption Viterbi Decoder LSIC Employing the SST (Scarce State Transition) Scheme for Multimedia Mobile Communications," *Electronics-Letter, IEE*, Vol.30, no. 8, pp.637-639, April 1994.
- [4] Kang and A. N. Willson Jr., "Low-power Viterbi decoder for CDMA Mobile Terminals," *IEEE-journal of Solid- State-Circuits*. Vol.33, no.3, pp473-82, March 1998.
- [5] C. Y. Chang and K. Yao "Systolic Array Processing of the Viterbi Algorithm," *IEEE Trans. Inform. Theory*, Vol. IT-35, no.1, pp 76-86, Jan. 1989.
- [6] Yun-Nan Chang "Design of Soft Output Viterbi Decoders with Hybrid Traceback Processing," *IEEE Int. Conf. ISCAS(2)* 2003.
- [7] S. L. Haridas and Dr. N. K. Choudhari "Design of Viterbi Decoder with Modified Traceback and Hybrid Register Exchange Processing," *ACM Int. Conf. ICAC3'09*, pp 223-226, Jan 2009.
- [8] S. L. Haridas and Dr. N. K. Choudhari "Design of Viterbi Decoder with Minimum Transition Hybrid Register Exchange Processing," *ACM Int. Conf. ICWET'10*, Feb.26-27, 2010.
- [9] S. Ranpara and Dong Sam Ha, "A low power Viterbi Decoder design for wireless communications applications" *Int. ASIC conference*, Sept. 1999.
- [10] Jing-ling Yang, Alfred, K.K. Wong, "Designing of Precomputational based low power viterbi decoder" *IEEE 6<sup>th</sup> CAS Symp. On Emerging Technologies: Mobile and Wireless Comm., Shanghai, China*, May 31 – June 2, 2004.

## Authors

**S. L. Haridas** received Bachelor of Engineering and Master of Engineering degrees from Nagpur University, M.S., India in 1988 and 1995 respectively. Currently he is pursuing the Ph. D. degree at R. T. M. Nagpur University. He is a Professor and Head of Electronics and Telecommunication Engineering department at B. D. College of Engineering, Sevagram, M. S., India.



Dr. N. K. Choudhari received Bachelor of Engineering and Master of Engineering degrees from Nagpur University, M.S., India in 1987 and 1993 respectively. He received Ph. D. degree from J. M. University, New Delhi, India. Presently he is working as a Principal at Smt. Bhagvati Chaturvedi College of Engineering, Nagpur, M. S., India. He published almost 30 research papers in various conferences and reputed journals. He held different positions on various committees and bodies at university level. He guided nearly 10 research scholars in electronics engineering.

