Design and Implementation of High Throughput and Area Efficient Hard Decision Viterbi Decoder in 65nm Technology

Labishetty Bhanuteja1, M Venkateshwarlu2

1PG Scholar, Sri KS Raju Institute of Technology and Sciences, Hyderabad, TS, India.
2Associate Professor, Sri KS Raju Institute of Technology and Sciences, Hyderabad, TS, India.

Abstract: The best way of decoding against random errors is to compute the received sequence with every possible code sequence. This is called maximum likelihood (ML) decoding. The criterion for deciding between two paths is to select the one having the smaller metric. The rule maximizes the probability of a correct decision. Initially the digital signal is applied to the constellation mapper then the obtained output is given to puncture block. Punctured codes can be used in communication systems with a rate-adaption mechanism. It converts the code rate from one to two bits which can be carried by conventional encoder. The viterbi decoder performs the decoding mechanism by accepting the encoder output through the interface which can be converted into original form through the constellation demapper blocks. The MB OFDM reduces the number of computations and maintains the orthogonality. The Proposed Viterbi decoder reduces memory and the hardware resources. Exploiting parallelism with n way parallel architecture enables to keep throughput constraint at n times lower clock speeds. The proposed architecture checks every node for path metric value and eliminates the path that is found if it is not having minimum distance. This architecture is simulated, synthesized and implemented by VERILOG language using XILINX ISE Tool.

Keywords: Puncturer, Viterbi, BMU, ACS.

I. INTRODUCTION

In telecommunication, a convolutional code is a type of error-correcting code that generates parity symbols via the sliding application of a boolean polynomial function to a data stream. The sliding application represents the 'convolution' of the encoder over the data, which gives rise to the term 'convolutional coding.' The sliding nature of the convolutional codes facilitates trellis decoding using a time invariant trellis. Time invariant trellis decoding allows convolutional codes to be maximum likelihood soft decision decoded with reasonable complexity. The ability to perform economical maximum likelihood soft decision decoding is one of the major benefits of convolutional codes. This is in contrast to classic block codes which are generally represented by a time variant trellis and therefore are typically hard decision decoded. Convolutional codes are often characterized by the base code rate and the depth (or memory) of the encoder [n,k,K]. The base code rate is typically given as n/k, where n is the input data rate and k is the output symbol rate. The depth is often called the "constraint length" K, where the output is a function of the previous K-1 inputs. The depth may also be given as the number of memory elements 'v' in the polynomial or the maximum possible number of states of the encoder (typically 2^v). Convolutional codes are often described as continuous. However, it may also be said that convolutional codes have arbitrary block length, rather than that they are continuous, since most real world convolutional encoding is performed on blocks of data.

Convolutionally encoded block codes typically employ termination. The arbitrary block length of convolutional codes can also be contrasted to classic block codes, which generally have fixed block lengths that are determined by algebraic properties. The Viterbi decoding algorithm, proposed in 1967 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 decoding algorithm provides both a maximum-likelihood and a maximum a posteriori algorithm. A maximum a posteriori algorithm identifies a code word that maximizes the conditional probability of the received code word against the received code word, in contrast a maximum likelihood algorithm identifies a code word that maximizes the conditional probability of the received code word against the decoded code word. The two algorithms give the same results when the source information has a uniform distribution. Traditionally, performance and silicon area are the two most important concerns in VLSI design. Recently, power dissipation has also become an important concern, especially in battery- powered applications, such as cellular phones, pagers and laptop computers.

Power dissipation can be classified into two categories, static power dissipation and dynamic power dissipation. Typically, static power dissipation is due to various leakage currents, while dynamic power dissipation is a result of charging and discharging the parasitic capacitance of transistors and wires. Since the dynamic
power dissipation accounts for about 80 to 90 percent of overall power dissipation in CMOS circuits; numerous techniques have been proposed to reduce dynamic power dissipation. These techniques can be applied at different levels of digital design, such as the algorithmic level, the architectural level, the gate level and, the circuit level. A viterbi decoder uses the Viterbi algorithm for decoding a bit stream that has been encoded using Forward error correction based on a Convolutional code. The Viterbi algorithm is commonly used in a wide range of communications and data storage applications. It is used for decoding convolutional codes, in baseband detection for wireless systems, and also for detection of recorded data in magnetic disk drives. The requirements for the Viterbi decoder or Viterbi detector, which is a processor that implements the Viterbi algorithm, depend on the applications where they are used. This results in very wide range of required data throughputs and power or area requirements.

II. THE VITERBI DECODER ALGORITHM

The Viterbi decoding algorithm is a decoding process for convolutional codes for memory-less channel. Fig.1 depicts the normal flow of information over a noisy channel. For the purpose of error recovery, the encoder adds redundant information to the original Information, and the output is transmitted through a channel. Input at receiver end (r) is the information with redundancy and possibly, noise. The receiver tries to extract the original information through a decoding algorithm and generates an estimate (e). A decoding algorithm that maximizes the probability p(r|e) is a maximum likelihood (ML) algorithm. An algorithm which maximizes the p(r|e) through the proper selection of the estimate (e) is called a maximum a posteriori (MAP) algorithm. The two algorithms have identical results when the source information i has a uniform distribution.

Fig.1. The Convolutional Decoding.

The Viterbi Algorithm was developed by Andrew J. Viterbi and first published in the IEEE transactions journal on Information theory in 1967 [1]. It is a maximum likelihood decoding algorithm for convolutional codes. This algorithm provides a method of finding the branch in the trellis diagram that has the highest probability of matching the actual transmitted sequence of bits. Since being discovered, it has become one of the most popular algorithms in use for convolutional decoding. Apart from being an efficient and robust error detection code, it has the advantage of having a fixed decoding time. This makes it suitable for hardware implementation. This block generates the decoded output sequence. In the trace back approach, the block incorporates combinational logic, which traces back along the survivor path and latches the path (equivalently the decoded output sequence) to a register.

Fig.2. The block diagram of a proposed system.

A. Encoding Mechanism

Data is coded by using a convolutional encoder. It consists of a series of shift registers and an associated combinational logic. The combinational logic is usually a series of exclusive-or gates. The conventional encoder K=9 used for the purpose of this project. The octal numbers 171 and 133 when represented in binary form correspond to the connection of the shift registers to the upper and lower exclusive-or gates respectively. Fig.2 represents this convolutional encoder that will be used for the project.

B. Decoding Mechanism

There are two main mechanisms by which Viterbi decoding may be carried out namely, the Register Exchange mechanism and the Traceback mechanism. Register exchange mechanisms, as explained by Ranpara and Sam Ha store the partially decoded output sequence along the path. The advantage of this approach is that it eliminates the need for traceback and hence reduces latency. However at each stage, the contents of each register needs to be copied to the next stage. This makes the hardware complex and more energy consuming than the traceback mechanism. Traceback mechanisms use a single bit to indicate whether the survivor branch came from the upper or lower path. This information is used to traceback the surviving path from the final state to the initial state. This path can then be used to obtain the decoded sequence. Traceback mechanisms prove to be less energy consuming and will hence be the approach followed in this project. Decoding may be done using either hard decision inputs or soft decision inputs. Inputs that arrive at the receiver may not be exactly zero or one. Having been affected by noise, they will have values in between and even higher or lower than zero and one.

The values may also be complex in nature. In the hard decision Viterbi decoder, each input that arrives at the receiver is converted into a binary value (either 0 or 1). In the soft decision Viterbi decoder, several levels are created and the arriving input is categorized into a level that is closest to its value. If the possible values are split into 8 decision
Design and Implementation of High Throughput and Area Efficient Hard Decision Viterbi Decoder in 65nm Technology

levels, these levels may be represented by 3 bits and this is known as a 3 bit Soft decision. This project uses a hard decision Viterbi decoder for the purpose of developing and verifying the new energy saving algorithm. Once the algorithm is verified, a soft decision Viterbi decoder may be used in place of the hard decision decoder. Fig.3 shows the various stages required to decode data using the Viterbi Algorithm. The decoding mechanism comprises of three major stages namely the Branch Metric Computation Unit, the Path Metric Computation and Add-Compare-Select (ACS) Unit and the Traceback Unit. A schematic representation of the decoder is described below.

Fig.3. Schematic representation of the Viterbi decoding block.

C. Branch Metric Computation (BMC)

For each state, the Hamming distance between the received bits and the expected bits is calculated as shown in Fig.4. Hamming distance between two symbols of the same length is calculated as the number of bits that are different between them. These branch metric values are passed to Block 2. If soft decision inputs were to be used, branch metric would be calculated as the squared Euclidean distance between the received symbols. The squared Euclidean distance is given as \((a_1-b_1)^2 + (a_2-b_2)^2 + (a_3-b_3)^2\) where \(a_1, a_2, a_3\) and \(b_1, b_2, b_3\) are the three soft decision bits of the received and expected bits respectively.

Fig.4. Branch metric unit.

D. Trace Back Unit

The global winner for the current state is received from Block 2. Its predecessor is selected in the manner. In this way, working backwards through the trellis, the path with the minimum accumulated path metric is selected. This path is known as the traceback path. A diagrammatic description will help visualize this process. The state having minimum accumulated error at the last time instant is State 10 and traceback is started here. Moving backwards through the trellis, the minimum error path out of the two possible predecessors from that state is selected. This path is marked in blue. The actual received data is described at the bottom while the expected data written in blue along the selected path. It is observed that at time slot three there was an error in received data (11). This was corrected to (10) by the decoder.

Fig.5. Selected minimum error path for a ½ code rate.

III. REGISTER EXCHANGE METHOD

The register exchange (RE) method is the simplest conceptually and a commonly used technique. Because of the large power consumption and large area required in VLSI implementations of the RE method, the trace back method (TB) method is the preferred method in the design of large constraint length, high performance Viterbi decoders. In the register exchange, a register assigned to each state contains information bits for the survivor path from the initial state to the current state. In fact, the register keeps the partially decoded output sequence along the path, as illustrated in Fig.6. The register of state S1 at \(t=3\) contains ‘101’. This is the decoded output sequence along the hold path from the initial state.
The register-exchange method eliminates the need to trace back since the register of the final state contains the decoded output sequence. However, this method results in complex hardware due to the need to copy the contents of all the registers in a stage to the next stage. The survivor path information is applied to the least significant bit of each register, and all the registers perform a shift left operation at each stage to make room for the next bits. Hence, each register fills in the survivor path information from the least significant bit toward the most significant bit. The scheme is called shift update. The shift update method is simple in implementation but causes high switching activity due to the shift operation and, hence, results in high power dissipation.

In order to realize a certain coding scheme a suitable measure of similarity or distance metric between two code words is vital. The two important metrics used to measure the distance between two code words are the Hamming distance and Euclidian distance adopted by the decoder depending on the code scheme, required accuracy, channel characteristics and demodulator type.

A. Trellis Diagram

A convolutional encoder is often seen as a finite state machine. Each state corresponds to some value of the encoder's register. Given the input bit value, from a certain state the encoder can move to two other states. These state transitions constitute a diagram which is called a trellis diagram. A trellis diagram for the code on the Figure 3.2 is depicted on the Figure 3.3. A solid line corresponds to input 0, a dotted line – to input 1 (note that encoder states are designated in such a way that the rightmost bit is the newest one). Each path on the trellis diagram corresponds to a valid sequence from the encoder's output. Conversely, any valid sequence from the encoder's output can be represented as a path on the trellis diagram. One of the possible paths is denoted as red (as an example). Note that each state transition on the diagram corresponds to a pair of output bits. There are only two allowed transitions for every state, so there are two allowed pairs of output bits, and the two other pairs are forbidden. If an error occurs, it is very likely that the receiver will get a set of forbidden pairs, which don't constitute a path on the trellis diagram. So, the task of the decoder is to find a path on the trellis diagram which is the closest match to the received sequence. Let's define a free distance as a minimal Hamming distance between two different allowed binary sequences (a Hamming distance is defined as a number of differing bits).

IV. RESULTS

Results of this paper is shown in bellow Figs.8 to 10.

Fig.6. Register Exchange Method.

Fig.7. A trellis diagram corresponding to the encoder.

Fig.8. schematic diagram.

Fig.9. RTL Schematic.
V. CONCLUSION

We have proposed a high speed VD design for TCM systems. The precomputation architecture that incorporates T-algorithm efficiently reducing the decoding speed appreciably. We have also analysed precomputation algorithm, where the optimal pre computation steps are calculated and discussed. This algorithm is suitable for TCM systems which always employ high rate convolution code. Finally we presented a design case. Both the ACSU and SMU are modified to correctly to decode the signal. synthesis results show that VD could improving the maximum decoding speed. By using FPGA device and hybrid microprocessor the decoding benefits can be achieved in future. In future to improve the decoder performance the Viterbi algorithm is carried out in reconfigurable hardware. Power saving architecture can be designed for the above decoder which is executable in the mobile devices. Viterbi decoder can also be implemented using JAVA. Therefore in the future Viterbi algorithm may be used for various scenarios. So in the future the complexity can be greatly reduced.

VI. REFERENCES
