ISSN 2348 – 9928
IJAICT Volume 3, Issue 10, October 2016 Doi:01.0401/ijaict.2014.07.07 Published on 05 (10) 2016

A REVIEW ON LOW POWER DESIGN FOR ASYNCHRONOUS VITERBI DECODER

Mr. Nilesh S. Wange
PG Scholar,

Dr. Sanjay L. Haridas
Professor and Head,

Mr. Sanjay. B. Tembhurne
Assistant Professor,

Electronics and Communication Engineering,
G. H. Raisoni Academy of Engineering and Technology,
Nagpur, Maharashtra, India

Abstract*— In digital communication system, channel coding techniques are mostly use convolutional codes for wireless Applications. For decoding the convolutional codes Viterbi decoder is commonly used, because of its high speed performance. Because of fast developments in the communication systems have generated in increasing demand for high speed and low power Viterbi decoders with long battery life, low power dissipation and low weight. In an Asynchronous Technique various Handshaking signals are used to communicate between blocks. Hence asynchronous designs are inherently data driven and are active only when doing useful work, enabling considerable saving in power and operating at the average speed of all components. The Proposed method is focused on the design of low power consumption of a Viterbi decoder for rate of r=1/2, with a constraint length K=3 by using asynchronous technique. This design will be implemented in Virtex7 field Programmable Gate Array (FPGA) kit.

***Keywords*— Asynchronous Viterbi Decoder, VHDL.**

I. INTRODUCTION

Convolutional (Viterbi) decoding is a Forward Error Correction (FEC) technique [1]. The propose of FEC is to improve the capacity of a channel by adding some carefully designed redundant information to the data being transmitted through the channel. The process of adding this redundant information is known as channel coding. Convolutional coding and block coding are the two major forms of channel coding. Convolutional codes operate on serial data, one or a few bits at a time. Block codes operate on relatively large (typically, up to a couple of hundred bytes) message blocks. There are a variety of useful convolutional and block codes, and a variety of algorithms for decoding the received coded information sequences to recover the original data.

1.1 Convolutional Encoder

The Convolutional encoder can be viewed as a finite state machine. It generates a coded output data stream from an input data stream. It is usually composed of shift registers and a network of XOR (Exclusive-OR) gates as shown in Fig. 1. The encoder produces two bits of encoded information for each bit of input information, so it is called a rate 1/2 encoder.

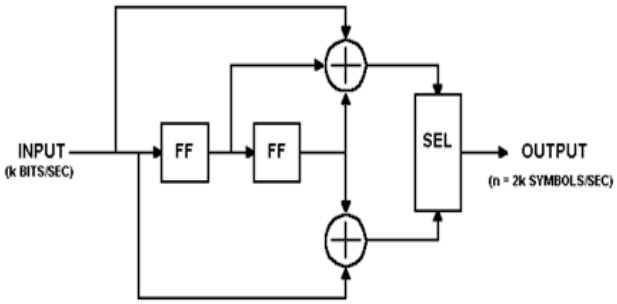

Fig 1 : Convolutional Encoder with rate=1/2 and Constraint length=3

A Convolutional encoder is generally characterized in (n, k, m) format, where n is number of outputs of the encoder; k is number of inputs of the encoder; m is number of memory elements (flip-flops) of the longest shift register of the encoder. The rate of a (n, k, m) encoder is k/n. The encoder shown in the figure is a (2, 1, 4) encoder with rate 1/2.

1.2 Viterbi decoder

A Viterbi decoder uses the Viterbi algorithm for decoding a bit stream that has been encoded using a convolutional code. Viterbi decoders are widely used in digital transmission and recording systems and are expected to be used in next generation wireless applications [2][3]. Two design styles are available for the implementation of Processors: Synchronous and asynchronous. Conventional synchronous processor designs are controlled by a global clock, running throughout the entire system. Asynchronous designs, on the other hand, are locally rather than globally synchronized and use handshaking signals between their components in order to perform the necessary synchronization and sequencing of events. There are many advantages to be gained from migrating to asynchronous designs and giving up the use of a global clock, with all its inherent overhead. In terms of power

consumption, synchronous systems involve higher switching activity than asynchronous ones. High switching activity translates into a large amount of wasted power. On the other hand, in data driven asynchronous systems, idle parts consume negligible power and switching activity is associated only with useful work being done: a valuable feature for battery operated systems.

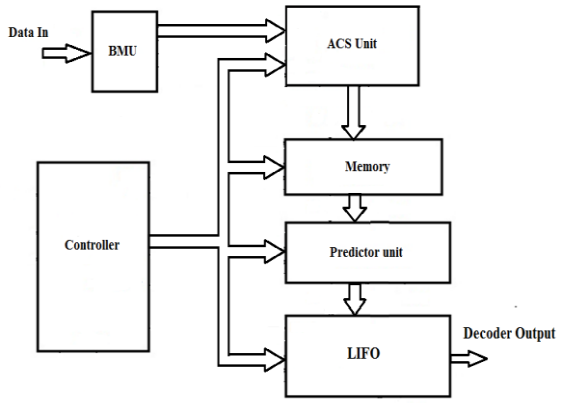

Fig 2 : Internal block diagram of Viterbi decoder.

II. DESIGN METHODOLOGY

The Design Methodology Viterbi decoding algorithm given below it consists of various steps to find the decoded data. In this, a Convolutional encoder generate the code using VHDL,which is pass as input to the Branch Metric unit(BMU),it find out the hamming distance by comparing bit by bit using adder circuit. The output of BMU pass to Add Compare Select unit (ACSU) as an input, it add and compare the data. The output of ACSU given to memory unit to store and process the data in register and the process data pass to the output unit to maintain data on its stack register all the procedure it will generate the output.

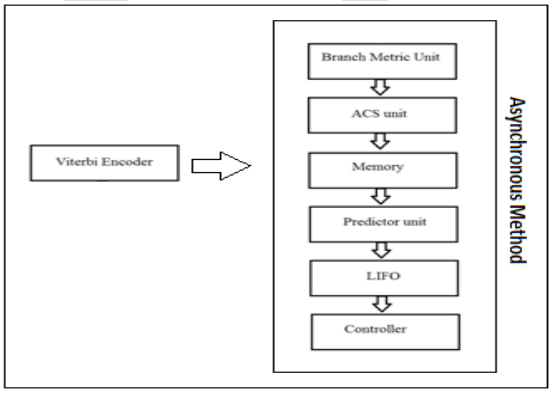

Fig 3 : Design Flow Process.

Every decoded bits put out by the predictor unit is in reverse order of the transmitted data, this necessitates a LIFO unit. This unit has 2 set of registers in which one of the register is read while the other is written. The two set of registers are read and write alternatively and simultaneously selected by the multiplexer which is in the read mode and gives the output in correct sequence. Controller block will use for handshaking signals.

III. UNITS

3.1. Branch Metric Unit

A branch metric unit's function is to calculate *branch metrics*, which are hamming distances between every possible symbol in the code alphabet, and the received symbol. It is used to generate branch metrics, which are hamming distances of input data from 00, 01, 10 and 11.

3.2. Add and Compare Select Unit

A new value of the state metrics has to be computed at each time instant. In other words, the state metrics have to be updated every clock cycle. Because of this recursion, pipelining, a common approach to increase the throughput of the system, is not applicable. The Add Compare-Select (ACS) unit hence is the module that consumes the most power and area.

3.3. Memory Unit

Memory is required to store the survivor Path Matrix Unit (PMU) . The word length of the memory depends on the number of the ACS sub-blocks used in the design or the total number of states in the decoder or 4 and the depth of the memory depends on the trellis length. We have for our project K = 3 and trellis length equal to 12, so the memory block used is 4 x12. The memory used is dual port. One port for writing the data and other for reading the data, as we need to write and read the data simultaneously and that to from different addresses. Memory should write data synchronously but the reading of the data should be asynchronous to keep the latency low or better manage the synchronous behaviour of the full system.

3.4. Predictor Unit

Predictor unit is used to trace back the trellis sequence of length 12 and predict the next state and actual decoded bit after rectifying the error. This unit is a state machine that is loaded with the state with minimum accumulated path metric after every 12 clock cycles. This unit uses the state value to access a bit from the path metric memory unit (PMM) or memory unit.

3.5. Controller

A controller is used to synchronize between the different modules of the system. The controller also includes two six bit counters of which one counts up another counts down. These counters drive the write and read addresses of the memory.

IV. LITERATURE REVIEW

In 2012, T. Kalavati Devi and C Venkatesh [4] worked on "Design of Low Power Viterbi Decoder using Asynchronous Techniques "in this paper Viterbi decoders employed in digital mobile communications are complex in its implementation and dissipate large power. The proposed Viterbi decoder uses asynchronous design techniques to reduce power consumption. The asynchronous design was based upon Quasi Delay Insensitive (QDI) timing model implemented in DCVSL which is used for robust and low power applications. The simulation results show the asynchronous design has the decrease in power consumption by 56.20% with increase in transistor count by 1.8 times in relative to synchronous Viterbi decoder with code rate of ½ and constraint length of 3 to 7 in 0.25µm CMOS technology with a power supply of 2.5V.

In 2013 Luis Alberto Luna Espinosa, Juan de Dios Lopez Sanchez,. Worked on "Viterbi decoders generation for FPGA platforms,"*IEEE*.[5]In this paper, they presented a relation that predicts the way that the states will be interconnected to implement a specific decoder using the Viterbi algorithm and ACS cells. Furthermore, the implementation of a script that makes use of basic entities allowing the generation of several types of decoders without presenting any difficulty in its development, this enable a quick set up for several communication systems giving time of development in other aspects of the system. The behaviour of the complexity of the decoder is also verified by them.

In 2013, Pooran Singh and Santosh Kr.Vishvakarma worked on, "RTL level implementation of High Speed-Low power viterbi encoder and Decoder, "*IEEE*, in this paper Viterbi encoder decoder with K=3 is done [6][7]. They worked based on about to make a Viterbi Encoder- Decoder fast with low power consumption. So for getting high speed we used Virtex 7 Kit to generate a 393.544 MHz clock frequency of our proposed encoder decoder and for low power optimization we used Spartan 6 Low voltage Xilinx FPGA which shows a power reduction of Viterbi decoder upto 11.34 mW. The results are calculated in Xilinx synthesis tool and the power optimization is done in X-power Estimator in Xilinx. The results shows that there is about 11.34 mW of Dynamic Power consumed by the logic and it run with 393.544 MHz Clock frequency.

V. CONCLUSION AND FUTURE WORK

Viterbi decoders employed in digital communications are complex in its implementation and dissipate large power. so overcome this problem, We will design asynchronous Viterbi Decoder for wireless applications, because of asynchronous design is inherently data driven and are active only when doing useful work, enabling considerable savings in power and operating at the average speed of all components and Power consumption will be low as possible.

References

[1] Pooran Singh and Santosh Kr.Vishvakarma,"RTL level implementation of High Speed-Low power viterbi encoder and Decoder, "IEEE, third International conference on information Science and Technology,China,2013

[2] Luis Alberto Luna Espinosa, Juan de Dios López Sánchez, Juan Ivan Nieto Hipolito, Mabel Vazquez Briseño, Aldo E. Pérez Ramos, Salvador Villareal Reyes."Viterbi decoders generation for FPGA platforms,"IEEE, 2013 International Conference on Mechatronics, Electronics and Automotive Engineering.

[3] T. kalavati Devi and C Venkatesh "Design of Low Power Viterbi Decoder using Asynchronous Techniques" International Journal of Advances in Engineering & Technology, Vol 4, Issue 1, pp. 561-570, July 2012

[4] S.L.Haridas, Dr. N. K. Choudhari 'Design of Viterbi Decoder with Minimum Transition Hybrid Register Exchange Processing' International Conference and Workshop on Emerging Trends in Technology, ICWET 2010- TCET, Mumbai, India

[5] Chih-Jhen Chen, Chu Yu, Mao-Hsu Yen, Pao-Ann Hsiung, and Sao-Jie Chen "Design of a Low Power Viterbi Decoder for Wireless Communication Applications"2010 IEEE 14th International Symposium on Consumer Electronics

[6] T. kalavati Devi and C Venkatesh"High Performance and Low Power VLSIC Architecture of Viterbi Decoder using Asynchronous QDI Techniques" International Journal of Recent Trends in Enginering, Vol 2, No. 6, November 2009.

[7] Mohamed Kawokgy C. André T. Salama, "Low Power Asynchronous Viterbi Decoder for Wireless Applications," International Symposium on low power electronics and Design (ISPELD) 204.