High- speed- Low-Power Viterbi Decoder Design

V.D.M. Jabez Daniel

Assistant Professor,
Dr. Sivanthi Aditanar College of Engineering,
jabezdaniel@gmail.com

P. Arul Sindhia

Assistant Professor,
Infant Jesus College of Engineering and Technology,
sindhiaphilip@gmail.com

Abstract:- High-speed, low-power design of Viterbi decoders for trellis coded modulation (TCM) systems is presented in this paper. It is well known that the Viterbi decoder (VD) is the dominant module determining the overall power consumption of TCM decoders. We propose a pre-computation architecture incorporated with -algorithm for VD, which can effectively reduce the power consumption without degrading the decoding speed much. A general solution to derive the optimal pre-computation steps is also given in the paper. Implementation result of a VD for a rate-3/4 convolutional code used in a TCM system shows that compared with the full trellis VD, the precomputation architecture reduces the power consumption.

———————————————————————————————*********———————————————————————————————

I. INTRODUCTION

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 Algorithm (VA) finds the most-likely state transition sequence in a state diagram, given a sequence of symbols. The Viterbi algorithm is used to find the most likely noiseless finite-state sequence, given a sequence of finite-state signals that are corrupted by noise.

In order to reduce the power consumption, and increase the speed, an asynchronous technique that is Delay Insensitive Null Convention Logic (NCL) for Viterbi Decoder (VD) and its Encoder using Dual rail signal [5] is proposed in this paper. In VLSI design the major cause for the power dissipation is the dynamic power dissipation about 80 to 90 percent of total power dissipation. NCL reduces the dynamic power consumption in terms of reducing the switching activity and also it reduces the Glitch power significantly, thereby achieving the lower power. The basic Viterbi algorithm [12] was applied in digital communication systems, speech and character recognition. It focused on the operations and the practical memory requirement to implement the Viterbi algorithm in real-time. Based on the data generated and decoded [10] from the zero Hamming distance path, unnecessary computations in the Viterbi decoder was avoided. Speed and power were not considered. Low-power bit-serial Viterbi decoder chip [1] for the code rate r = 1/ 3 and the constraint length K = 9 (256 states) was discussed. The add-compare-select (ACS) module was based on the bit-serial arithmetic and implemented with the pass transistor logic circuit. The Scarce state transition (SST) scheme [14] employed a simple pre decoder followed by a pre encoder to reduce the transitions of the Viterbi decoder.

II. LITERATURE SURVEY

General solutions for low-power VD design have been well studied by existing work. Power reduction in VDs could be achieved by reducing the number of states (for example, reduced-state sequence decoding (RSSD) [2], T-algorithm [3] and T -algorithm [4], [5]) or by over-scaling the supply voltage [6]. Over-scaling of the supply voltage usually needs to take into consideration the whole system that includes the VD (whether the system allows such an over-scaling or not), which is not the main focus of our research. RSSD is in general not as efficient as the M-algorithm [2] and T -algorithm is more commonly used than M-algorithm in practical applications, because the T-algorithm requires a sorting process in a feedback loop while M -algorithm only searches for the optimal path metric (PM), that is, the minimum value or the maximum value of all PMs. Algorithm has been shown to be very efficient in reducing the power consumption [7], [8]. However, searching for the optimal PM in the feedback loop still reduces the decoding speed. To overcome this drawback, two variations of the T -algorithm have been proposed: the relaxed adaptive VD [7], which suggests using an estimated optimal PM, instead of finding the real one each cycle and the limited-search parallel state VD based on scarce state transition (SST) [8]. In our

418

preliminary work [9], we have shown that when applied to high-rate convolutional codes, the relaxed adaptive VD suffers a severe degradation of bit-error-rate (BER) performance due to the inherent drifting error between the estimated optimal PM and the accurate one. On the other hand, the SST based scheme requires predecoding and re-encoding processes and is not suitable for TCM decoders. In TCM, the encoded data are always associated with a complex multi-level modulation scheme like 8-ary phase-shift keying (8PSK) or 16/64-ary quadrature amplitude modulation (16/64QAM) through a constellation point mapper. At the receiver, a soft-input VD should be employed to guarantee a good coding gain. Therefore, the computational overhead and decoding latency due to predecoding and re-encoding of the TCM signal become high. In our preliminary work [9], we proposed an add-compare-select unit (ACSU) architecture based on precomputation for VDs incorporating T -algorithm, which efficiently improves the clock speed of a VD with T -algorithm for a rate-3/4 code. In this work, we further analyze the precomputation algorithm. A systematic way to determine the optimal precomputation steps is presented, where the minimum number of steps for the critical path to achieve the theoretical iteration bound is calculated and the computational complexity overhead due to pre-computation is evaluated. Then, we discuss a complete low-power high-speed VD design for the rate-3/4 convolutional code [1]. Finally ASIC implementation results of the VD are reported, which have not been obtained in our previous work in [9].
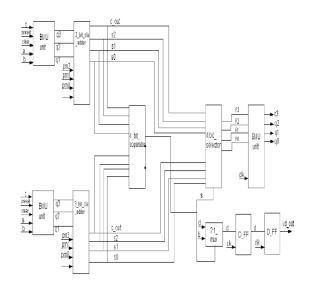
III. VITERBI DECODER

A typical functional block diagram of a Viterbi decoder is shown in Fig. 1. First, branch metrics (BMs) are calculated in the BM unit (BMU) from the received symbols. In a TCM decoder, this module is replaced by transition metrics unit (TMU), which is more complex than the BMU. Then, BMs are fed into the ACSU that recursively computes the PMs and outputs decision bits for each possible state transition. After that, the decision bits are stored in and retrieved from the SMU in order to decode the source bits along the final survivor path. The PMs of the current iteration are stored in the PM unit (PMU). T -algorithm requires extra computation in the ACSU loop for calculating the optimal PM and puncturing states. Therefore, a straightforward implementation of T -algorithm will dramatically reduce the decoding speed. The key point of improving the clock speed of T –algorithm is to quickly find the optimal PM.

3.1. DESIGN OF VITERBI DECODER USING NCL

The Viterbi decoder consists of the Branch Metric Unit, Add compare and Select Unit, and the Survivor Path Memory Unit. This chapter explains the internal blocks of the all the three units and how that blocks are designed using the Null Convention Logic.

3.1.1. Branch Metric Unit

The Branch Metric Unit (BMU) consists of two input Exor gate and three bit counter. The branch metric computation block compares the received code symbol with the expected code symbol and counts the number of differing bits. If the received sequence and expected sequence are different then the output of the exor gate becomes high and the nme f1scutduig the counter. The block diagram of BMU using NCL is shown in Figure 2. The NCL EXOR gate has two dual rail inputs are X (X^0, X^1) and Y (Y^0, Y^1) and a single dual rail output is Z (Z^0, Z^1). 3-bit counter is designed by cascading the T-FF and the output of the one flip flop is given as clock input for the next flip flop. Further the T input for all the flip flops are tied to HIGH input. The preset and clear input is used to make the counter working as asynchronous counter. Initially the counter output q3 $(q3^0, q3^1)$ q2 $(q2^0, q2^1)$ q1 $(q1^0, q1^1$ s="1 01" hntepee=1 nla=1.Nwi h is lcsapidte h one trscutn h lcyl s"01" eeteEoaeotusthe clock input of 3 bit counter. For the next clock cycle the output of the counter q3 $(q3^0, q3^1)$ q2 $(q2^0, q2^1)$ q1 $(q1^0, q1^1)$ is"10 0" iial h one ilcutaltecokcce n twlot"00"we ece 11"
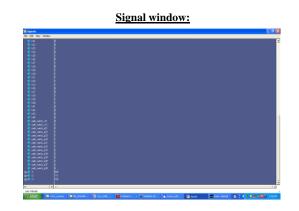

3.1.2. Add Compare and Select Unit

The Add Compare Select Unit (ACSU) which adds the BMs to the corresponding Path Metrics(PM), compares the new PMs, and then stores the selected PMs in the Path Metric Memory (PMM); at the same time, the ACSU stores the associated survivor path decisions in the Survivor Memory Unit (SMU). The PM of the survivor path of each state is updated and stored back into the PMM. Each butterfly wing is usually implemented by a module called ACS module. The output from the BMU and PM are processed as input to the two adder units.The two dual rail 3 bit inputs of the NCL adder units area a3 ($a3^0$, $a3^1$), a2 ($a2^0$, $a2^1$), a1 ($a1^0,a1^1$) and b3 ($b3^0,b3^1$), b2 ($b2^0,b2^1$), b1 ($b1^0$, $b1^1$). The output of the adder units are sum bits that is s3 ($s3^0$, $s3^1$), s2 ($s2^0$, $s2^1$), s1 ($s1^0$, $s1^1$) and carry bit that is cout ($cout^0$, $cout^1$). The comparator unit has two dual rail four bit inputs that can be obtained from the output of two adder units. The inputs are a3($a3^0,a3^1$), a2($a2^0,a2^1$), a1($a1^0,a1^1$), a0(a0 $^0,a0^1$) and b3($b3^0,b3^1$), b3($b2^0,b2^1$), b1($b1^0,b1^1$), b0($b0^0,b0^1$). The outputs are LT(LT^0,LT^1), EQ(EQ^0,EQ^1) and GT(GT^0,GT^1). When the value of a is less than the value of b then the LT output is high that is LT^0=0 and LT^1=1. When the value of a is greater than the value of b then the GT output is high that is GT^0=0 and GT^1=1. When the values of a and b are equal then the EQ output is high that is EQ^0=0 and EQ^1=1.The selector unit consists of four 2:1 multiplexor. The dual rail select input is s (s^0, s^1) which is from the LT output of the comparator. The dual rail two 4 bit inputs are from the output of adder units and the four bit outputs are f4($f4^0,f4^1$), f3($f3^0,f3^1$), f2($f2^0,f2^1$) and f1($f1^0,f1^1$).

IV. EXPERIMENTAL RESULTS AND DISCUSSIONS

The viterbi decoder is designed using the Null Convention Logic and the simulation results are verified using TANNER TOOL (SCH and TSPICE) in the 1.25µm technology, 3V Vdd and at a frequency of 2GHz.The output waveform of the Viterbi decoder using NCL is shown in figure 6. The dual rail inputs of the Viterbi decoder are the received sequence and the expected sequence. The dual rail output of the Viterbi decoder is VD_out0 and VD_out1. The block diagram of Viterbi decoder uses two branch metric units since each state have two branches in the trellis. Here the received sequence is a=c="11 01 11" and the Expected sequence for the first branch metric is b="00 10 01" and expected sequence for the second branch metric is d="11 01 10" and the decoded output sequence is VD out="11 01 10".
a

Signal window:

V. REFERENCES

[1] Chien-Ching Lin, Yen-Hsu Shih, Hsie-Chia Chang, and Chen-Yi Lee, 2005. Design of a Power-Reduction Viterbi Decoder for WLAN Applications, IEEE Transactions on Circuits and System-I: regular papers, 52(6), 321-328G.

[2] IrfanHabib, OzgunPaker, and Sergei Sawitzki,2009, Design Space Exploration of Hard-Decision Viterbi Decoding:

[3] ann S. YuanandWeidongKuang, 2004, Teaching Asynchronous Design in Digital Integrated Circuits, IEEE transactions on education,47(3),397-404

[4] Injin He, Zhongfeng Wang, Zhiqiang Cui, and Li Li, 2009, Towards an Optimal Trade-off of Viterbi Decoder Design, IEEE conferecne,3030-3033

[5] Joshi M.V., Gosavi S., Jegadeesan V., Basu A., Jaiswal S., Al-Assadi W.K. and Smith S.C. 2007, NCL Implementation of Dual-Rail 2s Complement 8×8 Booth2 Multiplier using Static and Semi-Static Primitives, IEEE region 5 Technical Conference, April 20-21, Fayetteville,59- 64.

[6] Jun Jin Kong, Keshhab K Parhi., 2004 Low-Latency Architectures for High-Throughput Rate Viterbi Decoder, IEEE Transactions on VLSI System, 12(6), 642-651.

[7] Meilana Siswanto1, Masuri Othman, Edmond Zahedi,2006 VLSI Implementation of 1/2 Viterbi Decoder for IEEE P802.15-3a UWB Communication, IEEE ICSE2006 Proc., Kuala Lumpur, Malaysia,666 – 670.

[8] Qing Li, Xuan-zhong Li, Han-hong Jiang and Wen-hao He 2008, A High-Speed Viterbi Decoder, Fourth International Conference on Natural Computation IEEE.,p.p. 313-316.

[9] Yao Gang, Ahmet T., Erdogan, and TughrulArslan, 2006, An Efficient Pre- Traceback Architecture for the Viterbi Decoder Targeting Wireless Communication Applications,IEEE Transactions on Circuits and Systems-I: regular papers, 53(9),423-432

[10] Yun-Ching Tang, Do-Chen Hu, Weiyi Wei, Wen-Chung Lin and HongchinLin , 2009. A Memory-Efficient Architecture for Low Latency Viterbi Decoders, IEEE.335-338

[11] Ajay Dholakia, 1994. Introduction to Convolutional Codes with Applications.Kluwer Academic Publishers.

[12] G. Forney, 1973. The Viterbi Algorithm, Proceedings of the IEEE, 61(3),268-278.

[13] Dalia A., El-Dib and Elmasry M.I. 2004. Modified Register-Exchange Viterbi Decoder for Low-Power Wireless Communications, IEEE Transactions on Circuits and Systems I, ,51(2), 371- 378.

[14] Lang L, Tsui C.Y and Cheng R.S.1997. Low power soft output Viterbi decoder scheme for turbo code decoding, IEEE Conference-Paper, ISCAS „97,New York, USA, 24, 1369-1372.