Modern Applied Science; Vol. 6, No. 11; 2012
ISSN 1913-1844 E-ISSN 1913-1852
Published by Canadian Center of Science and Education

The Design of Efficient Viterbi Decoder and Realization by FPGA

Liu Yanyan[1], Yang Xiaohui[1] & Chen Dianren[1]

[1] Institute of Electronic and Information Engineering, Changchun University of Science and Technology, China

Correspondence: Liu Yanyan, Institute of Electronic and Information Engineering, Changchun University of Science and Technology, Weixing Road No. 7089, Changchun City 130022, Jilin Province, China. Tel: 86-431-8558-2643. E-mail: liuyy306@163.com

Received: August 29, 2012 Accepted: October 20, 2012 Online Published: October 26, 2012

doi:10.5539/mas.v6n11p44 URL: http://dx.doi.org/10.5539/mas.v6n11p44

Abstract

Convolution code is a kind of widely used error-correcting codes in the error control field, in order to solve the Viterbi decoding of higher complex degree and lower speed etc. problem, a kind of efficient and reliable Viterbi decode method has been put forward specially. Firstly, the principle of Viterbi decode has been introduced by detail; Secondly, in order to improve the parallel decoding speed, Viterbi decoding algorithm is improved; And then, according to the improved algorithm to achieve high speed and parallel Viterbi decoding method, which is realized easily by FPGA; Finally, the function simulation and test for (2, 1, 7) convolution code has been carried out. The experimental results show that: when the system clock is 64 MHz, eventually the decoding rate of not less than 16 Mbps, improved Viterbi decoding algorithm has lower complexity, improved Viterbi decoding efficiency.

Keywords: error-correcting code, convolution code, Viterbi decode, FPGA

1. Introduction

The convolution code is a kind of linear coding method with memory packet, and equivalent to convolute between the input information stream and an impulse response. The constraint length of the convolutional code determines the coded error resilience ability, constraint length is longer, error resilience capability is stronger, but the decoding complexity is increased significantly. Complex degree of decode also large increase. Parallel decode complex degree is and restrain length index relation. The most important decode method of convolution code is Viterbi decode, in the decoding process, not only from the current moment received code group extracted decoding information, but also to use before or after each time of received code extracting relevant information (Sun & Ding, 2012; Niu & Ma, 2011; Cholan, 2012).

In order to reduce the decoding complexity and improve the decoding speed, this paper presents a kind of high speed and parallel method of Viterbi decode realized easily by FPGA.

2. The Work Principle of Viterbi Decode

For better understanding Viterbi decode, below give a specific decoding process. If the input to the encoder information sequences M = (1011100), by the output of the encoder code sequence C = (11100001100111), through the BSC is fed into the decoder in the sequence of R = (10100001110111), there are two error, and using the Viterbi decoding algorithm decoder output valuation information sequences $\hat{M}$ and $\hat{C}$ code sequence. Figure 1 and Figure 2 were painted in third time and the seventh time in each state to keep selected path and measure d (minimum distance), as well as the corresponding decoder valuation information sequences $\hat{M}$. When the seventh time, four have chosen path is only one, it is the output of the decoder valuation sequence $\hat{C}$ = (11 10 00 01 10 01 11), the corresponding valuation information sequences $\hat{M}$ = (1011100), R in the two error corrected.

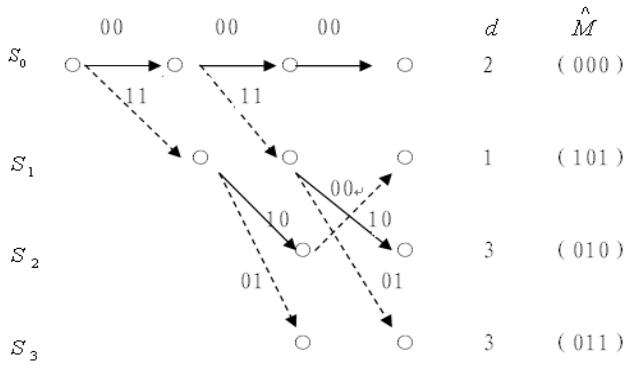

Figure 1. The selected path and measure d in third time

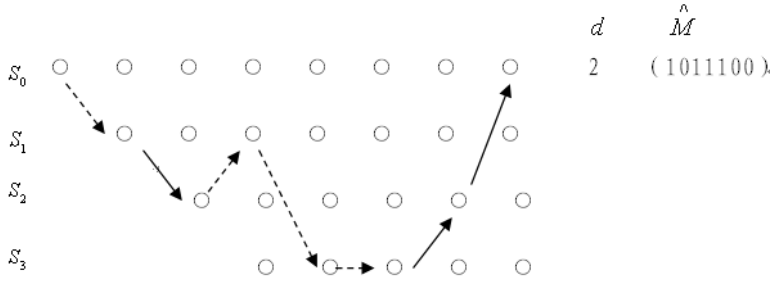

Figure 2. The selected path and measured in seventh time

In third time, enter S_0 state to keep selected path determination process is as follows: enter S_0 state there are two paths, one is by the (00) branch coupled with the branch before a moment (second time) for selected path of C_{01} =（00 00）connected to the path ($\hat{C}_{01}$ 00)=(00 00 00), d (R_2, 00)= d (00 00)=0, and d ($\hat{C}_{01}$ 00, R_0, R_1R_2)= d ($\hat{C}_{01}$, R_0R_1)+ d (R_2, 00)=2+0=2, so the path metric value is 2; the other path is composed of (11) branch with this branch is connected with the previous time (second time) for selected path of $\hat{C}_{01}'$=(11 10) connected to the path ($\hat{C}_{01}'$ 11)=(11 10 11), the metric value $d = d$ ($\hat{C}_{01}'$ 11, $R_0R_1R_2$)= d ($\hat{C}_{01}'$, R_0R_1)+ d (R_2 11)=1+2=3. Based on the minimum Hamming distance criterion available, in third time S_0 left the selected path is $\hat{C}_{012}$ =(00 00 00), the metric value d =2.

3. The Improvement Algorithm of Viterbi Decode

The Viterbi algorithm can be described as follows: in stage i, state $S_{j,i}$, at each grid node assignment $V(S_{j,i})$. The value of the node according to the following steps of calculation:

1) Receiving sequence has been divided into L length for n_0 code section, and draw the length of L+m segment of the grid graph;

2) Suppose $V(S_{0,0}) = 0$, $i = 1$;

3) In the phase of $i = 1$, calculated into each state branch portion of the path length, selected and stored minimal path length path as well as the length of $V(S_{j,1})$, call this part of path for the surviving path;

4) i increased 1, the stage to each state branch of all the branch lengths, and the same branch connected to the previous phase of the surviving path length $V(S_{j,i})$ addition, selected minimum to be stored and deleted all the other path, the phase of the surviving path and length of $V(S_{j,i+1})$, the surviving path to increase the length of one branch;

5) If j <L, then return to (4) step, otherwise jumped into (6) step;

6) From the L+1 phase of the whole 0 state, through the mesh of the surviving path back to the original 0 state, the path is the maximum likelihood path, corresponding to its input bit sequence is the maximum likelihood decoding information sequence.

4. The Design of the FPGA Structure of Viterbi Decoder

Viterbi decoding algorithm FPGA realized, the current commonly used in three ways: serial, parallel and series-parallel combination, in the serial implementation using only a serial processing unit to achieve the various state of the path metric values are updated, so that to achieve the most prominent advantages of saving hardware resources, but this approach has relatively the obvious disadvantages: low throughput and sequential complex decoder. For example, the realization of a constraint length K=7 decoder, there are a total of $2^{K-1} = 64$ state, so that each receives a code requires at least 64 master clock cycles to complete the treatment process, due to the specific implementation of each functional unit when the internal need of water treatment, such processing in 64 clock periods is not enough, and so will make the internal timing is quite complex, the need for precision control, thus greatly increasing the hardware design work load.

The parallel algorithm of Viterbi decoder, need to have the same number of state of ACJ (accumulator, comparator, judgement ware) unit, at a bit rate clock cycle to complete all state path metric values are updated, which can greatly improve the Viterbi decoding throughput, and the whole system just a bit rate clock is can work, timing is also greatly simplified, but the disadvantage is the consumption of resources, the constraint length K is bigger, especially the consumption of resources. But with the FPGA rapid development of manufacturing technology, chip scale problem has been not system bottleneck problem, now there are up to 8000000 FPGA comes out, thus solving the parallel algorithm hardware resource consumption problem (Zhang, Zhang, & Yao, 2011).

With the FPGA in resources and the operation speed of development, with area change rate method can make the decoder, including Viterbi channel codec performance close to the theoretical value, higher work frequency. The overall design scheme of Viterbi decoder shows as Figure 3.

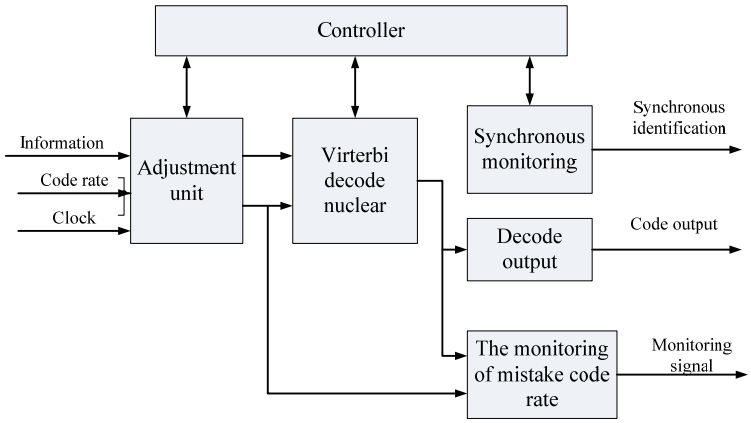

Figure 3. The overall design scheme of Viterbi decoder

The received information, rate and clock to adjust unit undertakes adjustment, output effective encoding information into the Viterbi decoding nuclear, controller realization of the various parts of the work, Viterbi decoding nuclear output decoding stream bit error rate monitoring, based on bit error rate statistics to determine whether the data synchronous receiving, at the same time the information into a self synchronous monitoring unit, for adjusting the synchronization using. When the decoding system synchronization, the output synchronization identification, representation system is synchronous. Viterbi decoder will eventually output decoding stream, synchronization identification and monitoring signal is fed to the next step of receiving unit (Angarita, Canet, & Sansaloni, 2008).

A complete Viterbi decoding nuclear structure includes accumulator, comparator, judgement ware, tolerance value register, information sequence register decision device, control circuit, as shown in Figure 4.

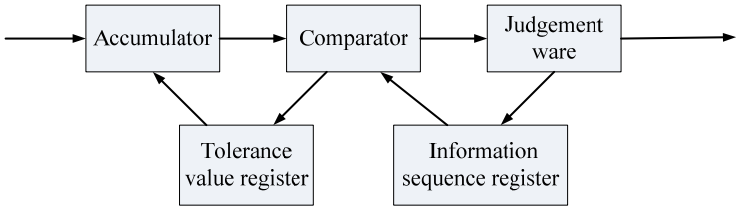

Figure 4. The basic structure of Viterbi decoder

R = 1/2 Viterbi decoding module according to the above algorithm, for a soft decision decoder, should possess the following several part:

(1) Tolerance value register: The tolerance that is used to stock each route value. Its former level should still have a state generator, produces 64 state and branch value.

(2) The accumulator, comparator and decision device. Were used to carry out soft distance accumulation, the path metric values and select output information element value.

(3) The path register: used to store the surviving path.

The branch metric value calculation section, first of all to receive the soft decision information to calculate each branch metrics, in adjusting module output disable pulse position can't be measured value. The result into the plus selection circuit, the comparison and selection circuit receives the selected path to leave, this information is fed to the path register. When the path register is in a 64 state path metric is equal when, after large number decision circuit outputs the decoded information, into the lower bit error monitoring and self synchronization circuit (Guo, Ahmad, & Swamay, 2005; Hsu, Kuo, & Hsu, 2007).

For (2, 1, 7) convolutional code, in a decoding cycle, accumulator completed 64 branch metric calculation, the comparator group completed the same state route distance value is compared, will be less in 64 metric value register. Decision device select 64 information sequence of registers in the minimum, and the corresponding information sequence decoding result output register.

In addition, we can also in throughput and scale of hardware on a compromise, with strings and combined algorithm to achieve the Vietbri decoder, for example to the constraint length of 7 convolutional code, we can use the 4 ACJ unit to realize the 64 state path metric update operation, this kind of decoder each receive a set of codewords, need at least 16 processing clock cycles to complete path metric value update. In practice, we can according to the specific situation, choose the ACJ unit number.

5. The Synthesis and Simulation of Viterbi Decode

The design of the Viterbi decode for (2, 1, 7) convolution code of this paper, take the resource utilization rate of FPGA specially than decoder as Figure 5 shows, FPGA chooses XCV300 of Virtex series, in which, the resource utilization rate of Slices is 50%.

Device Utilization Summary				
Logic Utilization	**Used**	**Available**	**Utilization**	**Note(s)**
Number of Slice Flip Flops	1,209	6,144	19%	
Number of 4 input LUTs	2,214	6,144	36%	
Logic Distribution				
Number of occupied Slices	1,564	3,072	50%	
Number of Slices containing only related logic	1,564	1,564	100%	
Number of Slices containing unrelated logic	0	1,564	0%	
Total Number 4 input LUTs	2,609	6,144	42%	
Number used as logic	2,214			
Number used as a route-thru	388			
Number used as Shift registers	7			
Number of bonded IOBs	128	162	79%	
IOB Flip Flops	81			
Number of GCLKs	4	4	100%	
Number of GCLKIOBs	1	4	25%	
Number of DLLs	1	4	25%	
Total equivalent gate count for design	35,898			
Additional JTAG gate count for IOBs	6,192			

Figure 5. The resource utilization rate that takes FPGA specially of Viterbi decoder

On the design of Viterbi decoder undertook careful analysis, given the various modules of the detailed design method, and the corresponding simulation, verified the correctness of the design of each module. The following Viterbi decoder is verified, Viterbi decoder overall diagram as shown in Figure 6. Clk is the input clock signal, rst is reset signal, enable is an enable signal, co_din is decoder input data, co_out[1:0] is the decoder output signal.

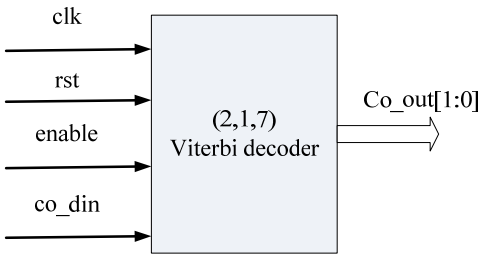

Figure 6. The overall diagram of Viterbi decoder

Under the integrated software environmental ISE design software of Xilinx, simulation and test have been carried out using simulation tool ModelSim for Viterbi decoder of (2, 1, 7) convolution code. The simulation waveform shows as Figure 7. Firstly using encoder as the sequence that had known carries out coding, the coding word that produces this input sequence goes on for the coding word that produces, is artificial to add to disturb, to the ability of error correction of verifying the Viterbi decoder designed for wrong information (Yin, Wen, & Jin, 2009). Through the sequence of contrast original coding sequence and decoder output, can find out, the sequence of input with decode export sequence consistent, so, can prove the correctness of Viterbi decoder design. Under systematic clock 64 MHz last decode speed do not be lower than 16 Mbps.

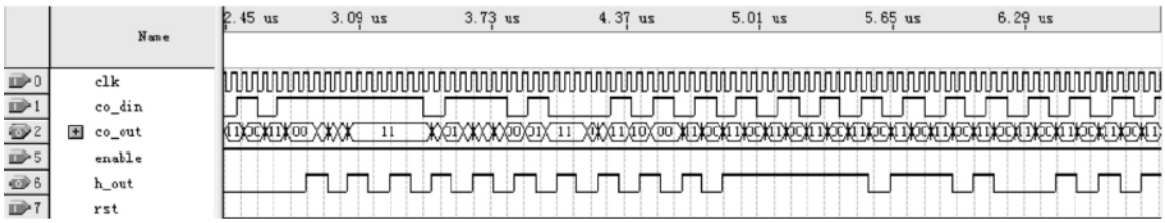

Figure 7. The simulation waveform of Viterbi decoder

6. Conclusions

This article on the Virterbi decoding principle and process are introduced in detail, it is easy to put forward a kind of FPGA to achieve high efficiency, reliable Viterbi decoding algorithm, and gives the concrete realization of the process, Viterbi decoding were functional simulation, the results show that when system clock is 64 MHz, eventually the decoding rate is not less than 16 Mbps, therefore, Viterbi decode algorithm real time better, which reliability is higher.

References

Angarita, F., Canet, M. J., & Sansaloni, T. (2008). Architectures for the implementation of a OFDM-WLAN Viterbi decoder. *Journal of Signal Processing Systems, 52*(1), 35-44. http://dx.doi.org/10.1007/s11265-007-0071-8

Cholan, K. (2012). Design and implementation of low power high speed Viterbi decoder. *Procedia Engineering, 30*, 61-68. http://dx.doi.org/10.1016/j.proeng.2012.01.834

Guo, M., Ahmad, M. O., & Swamay, M. N. S. (2005). FPGA design and implementation of a low-power systolic array-based adaptive Viterbi decoder. *IEEE Transactions on Circuits and Systems I: Regular Papers, 52*(2), 350-365. http://dx.doi.org/10.1109/TCSI.2004.838266

Hsu, C. Y., Kuo, T. S., & Hsu, Y. H. (2007). Low complexity radix-4 butterfly design for the soft-decision Viterbi decoder. *Microprocessors and Microsystems, 31*(8), 529-536. http://dx.doi.org/10.1016/j.micpro.2006.02.007

Niu, Y., & Ma, Z. S. (2011). High-speed Virterbi decoder based on FPGA. *Research & Development, 30*(8), 63-65.

Sun, Y., & Ding, Z. Z. (2012). FPGA Design and Implementation of a Convolutional Encoder and a Viterbi Decoder Based on 802.11a for OFDM. *Wireless Engineering and Technology, 3*, 125-131. http://dx.doi.org/10.4236/wet.2012.33019

Yin, S. W., Wen, J. O., & Jin, H. C. (2009). Implementation of convolutional encoder and Viterbi decoder using VHDL. *Proceedings of 2009 IEEE Student Conference on Research and Development*, 22-25.

Zhang, H. L., Zhang, G. Y., & Yao, G. Y. (2011). A novel structure of FPGA-based Viterbi decoder. *ICIC Express Letters, 5*(1), 163-168.