

Analysis and Perform an convolutional encoder

Patel Pratik J.¹ Tarunkumar C. Lad²

¹M.Tech (Purs.) ²M.Tech

¹CGPIT, Maliba Campus, UTU

²SVNIT, SURAT, India

Abstract--- this paper focused on the two different system of the convolutional encoder. We have to present a Spartan 3E field programmable gate array (FPGA) implementation of Convolutional Encoder with a constraint length of 3 and two different code rate 1/2 and 1/3. The purpose of a convolutional encoder is to a single input and generates encoded outputs. In this paper analysis and perform different type of system in the convolutional encoder. The system implication in VHDL and also simulated and synthesizes using Xilinx for RTL design.

Keywords: Forward Error correction, Xilinx, RTL simulation, SPRTAN 3E FPGA

I. INTRODUCTION

Convolutional coding one type of decoding algorithm, this Algorithm are currently used in cellphones or wireless communication. However, the largest current consumer of Viterbi algorithm processor cycles is probably digital video broadcasting.

Convolutional encoder is encoding the information this information adding redundancy bit to original information. The Viterbi decoder is decoding convolutional code. Digital commination is complex in Viterbi decoder. Encoder adds redundant bits to sender's bit stream to create a codeword. Decoder received error correcting code, a convolutional code works by adding redundant information to the user's data and correcting errors using this information. Convolutional codes offer an alternative to block codes for transmission over a noisy channel.

A complete code word is received and decoded output sequence is readily available. In the traceback scheme, only the survivor path information for each state is stored for each code symbol. Once the complete code word is received, a traceback block executed the decoded output sequence using the survivor path information. We considered two different methods for the traceback approach, shift update and selective update. In shift update method, the survivor path information is shifted in to the registers. In selected update method the survivor path information is routed by a multiplexer to appropriate registers. We investigated in power dissipation register-exchange and traceback approaches and the power dissipation of shift and selective update methods. Each encoded bit is a function of the present input bits and their past ones.

II. CONVOLUTIONAL CODER

Convolutional coding has been used in communication system including space communication. Convolutional coding can be performing to a continuous input block or

stream. A (n,k,m) convolutional code can be implication with a k - input sequence , n - output sequence , m -memory(flip-flops).it is considered as $k < n$,but the memory 'm' must large to achieved low error probabilities, code rate($R=k/n$)is defined ratios of number of input convolutional encoder to the number of bit in output symbol which not only input bit ,but also previous bit($k-1$). A convolutional encoder is implication in final state machine, also $k-1$ shift register.

A convolutional encoder is a Mealy machine. The output is a function of the current input and the current stage. In Mealy machine, one or more shift registers and multiple XOR gates. The stream of information bits flows in to the shift register from one end and is shifted out at the other end. XOR gates are connected to some stages of the shift registers as well as to the current input to generate the output. There is no basis for the optimal location of the shift register stages to be connected to XOR gates. [1]

A convolutional encoder consists of one or more shift registers and multiple XOR gates. The information bit shifted from shit register, process will continue with end of information bit. XOR gates are connected to some stages of the shift registers as well as to the current input to generate the output.

In this paper, we focus on convolutional code with $(n,1,m)$ encoder with the rate $1/n$, number of input $n=3$. Consider the (n,k,m) values for $(3,1,2)$.Generator generate three polynomials are $G1 = 1 + x + x^2$, $G2 = 1 + x + x^2$, $G3 = 1 + x^2$

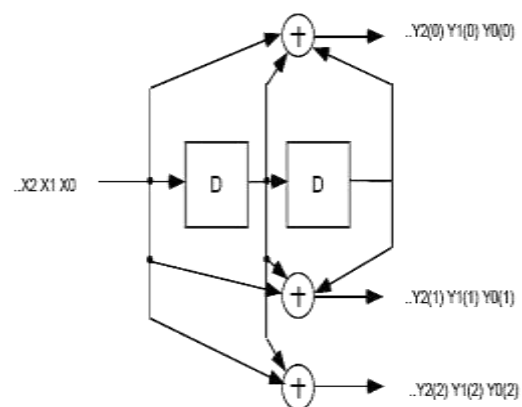


Fig. 1: convolutional encoder of rate 1/3

The above figure 1 shows the convolutional Encoder with coder rate $(K)=1/3$.encoder can be described in terms of state table and state diagram.

Figure 2 shows the convolutional encoder with coder rate $(K/n)=1/2$.

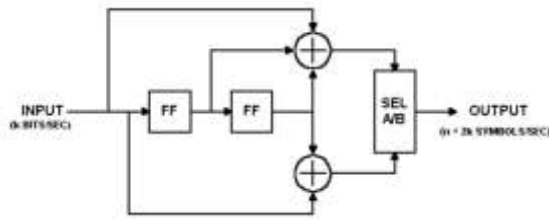


Fig. 2: convolutional encoder of rate 1/2.

In this figure 2 shows block diagram with code rate $(k/n)=1/2$, this second method for convolutional encoder with different code rate. Generator polynomials $G1 = 1 + x^2$, $G2 = 1 + x + x^2$ with consider length $k=3$ and $n=2$. [4]

III. TRELIS DIAGRAM

A trellis diagram is an extension of a state diagram that explicitly shows the passage of time. A convolutional code described by four state trellis is shows in fig.3. which can find optimal path with best maximum path matrix. To find optimal path compute in path matrix. Viterbi algorithm is basically implemented to decode the errors found in convolution encoded sequence. The Viterbi decoder the decoding complicity is proportional to the number of state, i.e, 2^k where k –length of convolutional code.

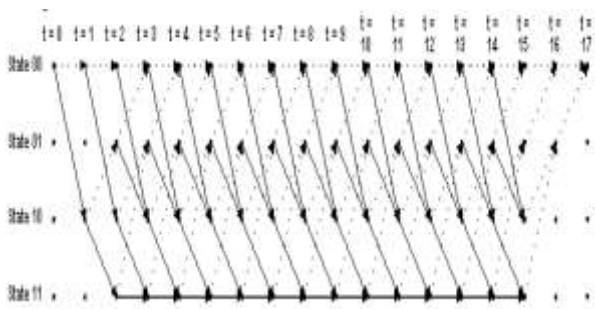


Fig 3: four state trellis diagram

Above figure 3 consider with four state. Let $s_0, s_1, s_2, s_3, s_4, \dots, s_{n-1}$ be consider being all state in trellis diagram for convolutional code to be decoder. A trellis diagram is an extension of a state diagram that explicitly shows the passage of time. In this figure 3 nodes $(s_0, s_1, s_2, s_3, s_4, \dots, s_{n-1})$ is represent states of the encoder. From an initial state s_0 the trellis records the possible transitions to the next states for each possible input pattern. For the encoder in Figure 3, there are two encoded symbols corresponding to input bit '0' and '1'. At the stage $t=1$ there are two states s_0 and s_1 , and each state has two transitions corresponding to input bits '0' and '1'. Hence the trellis grows up to the maximum number of states or nodes, which is decided by the number of memory elements for an encoder. After all the encoded symbols of the information bits are transmitted, the encoder is usually forced back into the initial state by applying a fixed input sequence called reset input sequence. The fixed input sequence reduces the possible transitions. In this manner, the trellis shrinks until it reaches the initial state.

IV. PROGRAMMABLE DEVICES

Programmable devices are those devices which can program by user requirement. Many programmable devices in market

like PLDs, CPLDs, ASICs and FPGAs.

A. Field programmable Gate Arrays

A field-programmable gate array (FPGA) is an integrated circuit designed to be configured by a customer requirement. The FPGA configuration is generally specified in a hardware description language (HDL), similar to that used for an application-specific integrated circuit (ASIC). FPGA contain many programmable logic block, and a hierarchy of reconfigurable interconnection between two node or block by using wired. In most FPGAs, the logic blocks also include memory elements. This memory elements identified be simple flip-flops or complete blocks of memory. As FPGA designs employ very fast I/Os and bidirectional data buses it becomes a challenge to verify correct timing of valid data within setup time and hold time. Floor planning is identified by time constraints in FPGA.

CPLD contend much large number of gate (logic gate) up to thousands to ten thousands, which can implementation complicated data processing devices. In CPLD very offer very predictable timing characteristics and ideal for critical control applications. The advantages of the FPGA over DSP implication include higher sampling rate, this sampling rate available in DSP chip, lower cost than ASIC. The FPGA also included design flexibility with optimal device utilization and system power in case with DSP chip.

B. Spartan family

The Spartan®-3A family of Field-Programmable Gate Arrays (FPGAs) solves the design challenges in most high-volume, cost-sensitive, I/O-intensive electronic applications. The five-member family offers densities ranging from 50,000 to 1.4 million system gates. The Spartan-3A FPGAs are part of the Extended Spartan-3A family, which also include the non-volatile Spartan-3AN and the higher density Spartan-3A DSP FPGAs. The Spartan-3A family builds on the success of the earlier Spartan-3E and Spartan-3 FPGA families. New features improve system performance and reduce the cost of configuration. These Spartan-3A family enhancements, combined with proven 90 nm process technology, deliver more functionality and bandwidth per dollar than ever before, setting the new standard in the programmable logic industry. Because of their exceptionally low cost, Spartan-3A FPGAs are ideally suited to a wide range of consumer electronics applications, including broadband access, home networking, display/projection, and digital television equipment. The Spartan-3A family is a superior alternative to mask programmed ASICs. FPGAs avoid the high initial cost, lengthy development cycles, and the inherent inflexibility of conventional ASICs, and permit field design upgrades. [4]

V. RESULTS ANALYSIS OF CONVOLUTIONAL ENCODER

Synthesis is a process of constructing a gate level net list from a register transfer level model of a circuit described in VHDL. Convolutional coder are design in Spartan 3E FPGA. In SPRTAN 3E FPGA Increasing design size and complexity, and also improvements in design synthesis tools and simulation tools, have made in HDL Languages the preferred design languages of most integrated circuit

designers.

A. Device utilization Report

After the completing implementation, you can verify the device utilization. The Synthesis report is generated after the compilation of design for the targeted Xilinx Spartan 3E based FPGA Device. This design is not implementation on the FPGA device. This report is used component in design. It is also compression of two code rate system, code rate 1/2 system is batter then coder rate 1/ system.

Selected Device	3s100evq100-5	
Number of Slices	2 out of 960	0%
Number of Slice Flip Flops	4 out of 1920	0%
Number of 4 input LUTs	2 out of 1920	0%
Number of IOs	4	
Number of bonded IOBs	4 out of 66	6%
Number of GCLKs	1 out of 24	4%

Table 1: Device Utilization with Code Rate 1/2

Selected Device	3s100evq100-5	
Number of Slices	2 out of 768	0%
Number of Slice Flip Flops	3 out of 1536	0%
Number of 4 input LUTs	2 out of 1536	0%
Number of IOs	6	
Number of bonded IOBs	6 out of 63	9%
Number of GCLKs	1 out of 8	12%

Table 2: Device utilization with code rate 1/3

B. Timing and Power Summary

The synthesis Report and timing diagram generated to input requirement. With the help of timing diagram speed Grade, input arrival time, output required time and combinational path delay calculated after the applying clock.

Timing Summary:

Speed Grade	5
Minimum period	1.843ns
Maximum Frequency	542.608MHz
Minimum input arrival time before clock	2.589ns
Maximum output required time after clock	4.040ns
Maximum combinational path delay	No path found

Table 3: Timing summary report for 1/2 code rate

Speed Grade	5
Minimum period	1.573ns
Maximum Frequency	653.748MHz
Minimum input arrival time before clock	2.378ns
Maximum output required time after clock	7.825ns
Maximum combinational path delay	7.789ns

Table 4: Timing summary report for 1/3 code rate

C. RTL schematic of convolutional encoder

We have to obtain two type of RTL schematic with different code rate. To observe the speed and power utilization in convolutional encoder as well as improvement in design simulation tools, have made in HDL. The two leading HDL languages are VHDL and Verilog. This both languages stranded in IEEE. Show the RTL view in fig4.

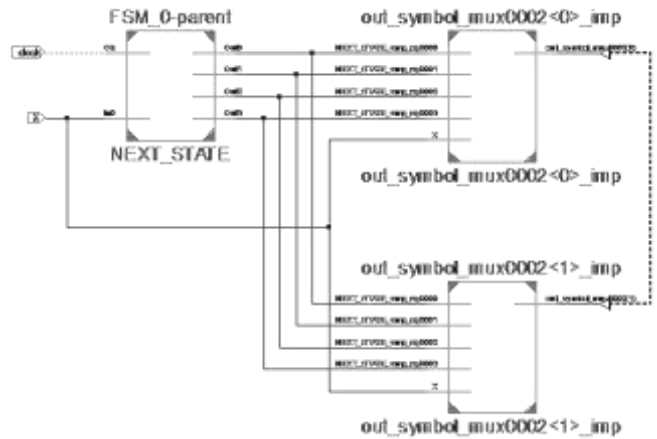


Fig. 4: RTL schematic of encoder with code rate 1/2

The other system are consider with coder rate 1/3.with have a different RTL view.

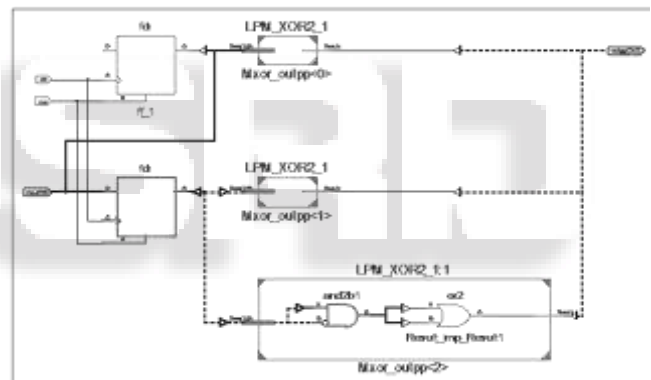


Fig. 5: RTL schematic of encoder with code rate 1/3

D. Behavior simulation

The Behavior Simulation for consider constraint length k= Code rate 1/2 and 1/3 has been developed and synthesis in Spartan 3E FPGA. The Behavior simulation show in figure 6 and 7.

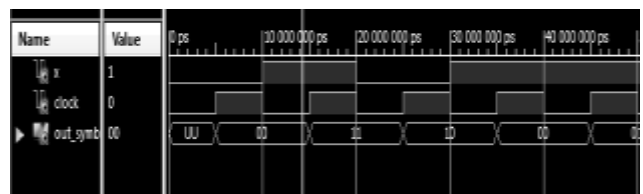


Fig. 6: Behavior simulation with code rate 1/2



Fig. 7: Behavior simulation with code rate 1/3.

E. Post-Route simulation

We can measure the input and output arrival time in wave form. The input arrival time is defined by input is stable before some amount of time. We have to measure the input arrival time is 4.03ns for 1/3 code rate dependent system. This arrival time show in fig 8.

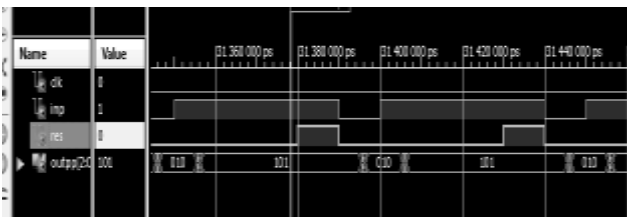


Fig. 8: Post-route simulation for 1/3 code rate system-



Fig. 9: Post-route simulation for 1/2 code rate system

F. Xpower Analyzer

We can use Xpower Analyzer to identify total power consumption in system. Two power consumption in system, first one is Quiescent power and other are Dynamical power. If power ration is low then system is batter.

Name	Value	Used	Total Available	Utilization (%)
Clock	0.00001 (W)	1	---	---
Logic	0.00002 (W)	4	1920	0.2
Signals	0.00006 (W)	9	---	---
I/Os	0.00992 (W)	6	66	9.1
Total Quiescent Power 0.03367 (W)				
Total Dynamic Power 0.01000 (W)				
Total Power 0.04367 (W)				

Fig 10: Xpower Analyzer with code rate 1/3

Name	Value	Used	Total Available	Utilization (%)
Logic	0.00013 (W)	8	1920	0.4
Signals	0.00040 (W)	17	---	---
I/Os	0.07130 (W)	22	66	33.3
Total Quiescent Power 0.03414 (W)				
Total Dynamic Power 0.07184 (W)				
Total Power 0.10598 (W)				
Junction Temp	30.2 (degrees C)			

Fig 11: Xpower Analyzer with code rate 1/3

ACKNOWLEDGMENT

In this paper optimized resource convolutional encoder has been design in Spartan 3E FPGA. The proposed convolutional encoder has been designed in VHDL using finite state Machine (FSM).The designed convolutional encoder has been simulated using Xilinx ISE simulator and synthesized with XST. The simulation and synthesized result show that proposed design can work frequency of 31.320MHz to 542.608MHz.The system is analysis by Xpower Analyzer and post routing simulation.

REFERENCES

[1] Ms.G.S. Suganya, Ms. G.Kavya, "RTL Design and VLSI Implementation of an efficient Convolutional Encoder and Adaptive Viterbi Decoder," IEEE conf. on commination and signal processing, April 3-5, 2013, India.

[2] FEI SUN, TONG ZHANG , " Low-power state-Parallel Relaxed Adaptive Viterbi Decoder," IEEE Trans. On circuits and System, Vol.54, No.5, May2007.

[3] MING-BO, "New path history management circuit for Viterbi decoder," in IEEE Transaction on Commination Vol 48, no.10, October 2000.

[4] HIRAL PUJARA, PANKAJ PRAJAPATI," RTL implementation of Viterbi decoder using VHDL," IOSR Journal of VLSI and signal processing vol 2. Issue 1 mar-Apr.2013.

[5] C. Arun, V. Rajamani, Design and VLSI implementation of a Low Probability of Error Viterbi decoder, First International Conference on Emerging Trends in Engineering and Technology, IEEE 2008.

[6] Matthias Kamuf, Member, IEEE, Viktor Öwall, Member, IEEE, and John B. Anderson, Fellow, IEEE, Optimization and Implementation of a Viterbi Decoder under Flexibility Constraints, IEEE Transactions on Circuits and Systems—I: Regular Papers, Vol. 55, No. 8, September 2008.

[7] J. Bhaskar, A VHDL Primer, Third Edition MaheJabeen and Salma Khan, Design of Convolution Encoder and Reconfigurable Viterbi Decoder, International Journal of Engineering and Science, Vol. 1, No.3, Sep 2012.