IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

IMPLEMENTATION OF SEQUENCE DETECTOR USING REVERSIBLE LOGIC

N Srinivasa Rao

Dept. of ETE,

BMS College of Engineering, Bengaluru, India,

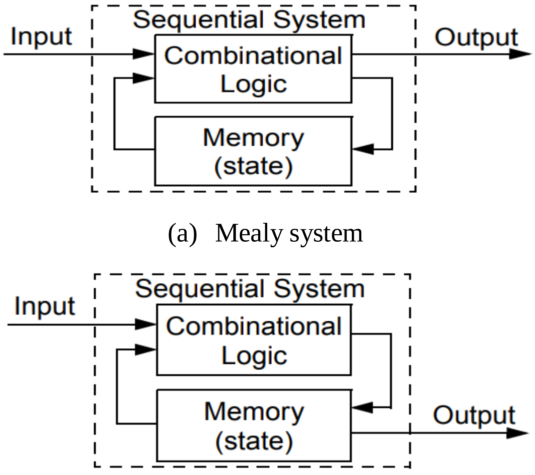
ABSTRACT

For any complex digital systems, design and analysis through state machine approach is preferred. In this paper, sequence detector using Mealy finite state machine is designed with reversible logic circuits. Reducing power dissipation is the main requirement for low power VLSI design. Reversible logic circuits have received more attention in the recent years because of their ability to reduce the power dissipation. The behavior of these circuits are described in Verilog HDL, Simulated and Synthesized in Xilinx ISE environment. In this work, sequence detectors with and without overlap sequence are considered

Key words: Finite state machine, Sequence detector, Verilog HDL, Xilinx, Reversible logic.

1. INTRODUCTION

A Finite state machine is a graphical approach to design complex digital systems. Finite state machines can be used to model real world problems in many fields, including traffic signals, artificial intelligence, and vending machine. The basic concept of an FSM is to store a sequence of states and transition between the states depending on the values of the external inputs and the current state of the machine [1]. The FSM can be of two types: Moore FSM and Mealy FSM. In Moore FSM, the output of the state machine is purely dependent on the current state variables, whereas the output can depend on the current state variable values and current input values in Mealy system. The general structure of Mealy and Moore FSM is shown in Figure 1.1.



(a) Mealy system

(b) Moore system

Figure 1.1. General structure of Finite state machines

This paper is organized as follows: Section II describes the brief introduction of the reversible logic gates and gates required for the present work. In Section III, the sequence detectors are described. Section IV describes the implementation using Mealy sequence detectors with reversible gates. The simulation and Synthesis results are shown in section V. Finally, Section VI concludes with scope for improvement.

2. REVERSIBLE GATES

A reversible logic gate is an nxn logic device with one-to-one mapping [2]. In reversible logic gates, the outputs are determined from the inputs and also the inputs can be uniquely recovered from the outputs. There are a number of reversible gates exists in the literature. Some of the widely used gates are Feynman, Fredkin, Toffoli and Peres gates [3,4,5]. A reversible circuit should be designed using less number of reversible logic gates [6]. In this work, Feynman gates and Fredkin are used to construct the reversible sequence detector. A brief description of these gates are given below:

2.1 Feynman Gate

Figure 2.1 shows a 2*2 Feynman gate. The Feynman gates has two inputs A, B and two outputs X, Y. The outputs are defined by X = A, Y = A⊕B.

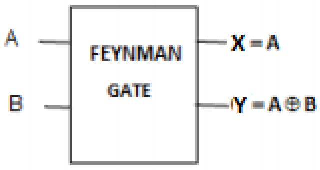
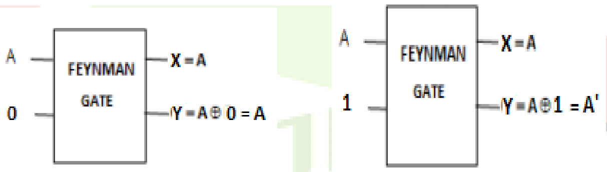


Figure 2.1. Feynman gate

Feynman gate is used to copy the input or complementing the input. This is shown in figure 2.2



(a) copier (b) complementer

Figure 2.2. Feynman gate as copier and complementer

2.2 Fredkin Gate

Figure 2.3 shows a 3*3 Fredkin gate [4]. It has three inputs A, B, C and three outputs X, Y and Z. The outputs are defined by the relation X = A, Y = A′B⊕AC and Z = A′C⊕AB. Figure 2.4 shows the realization of Fredkin gate as OR gate and AND gates.

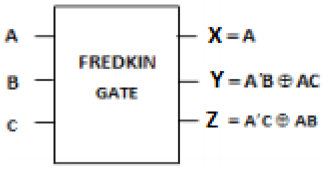
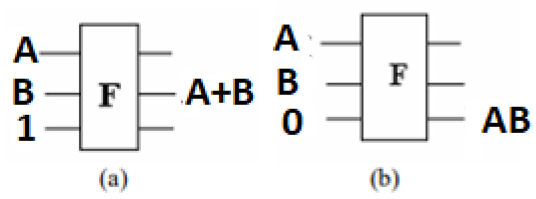


Figure 2.3. Fredkin gate



(a) OR gate (b) AND gate

Figure 2.4. Fredkin gate as OR gate and AND gate

3. SEQUENCE DETECTOR

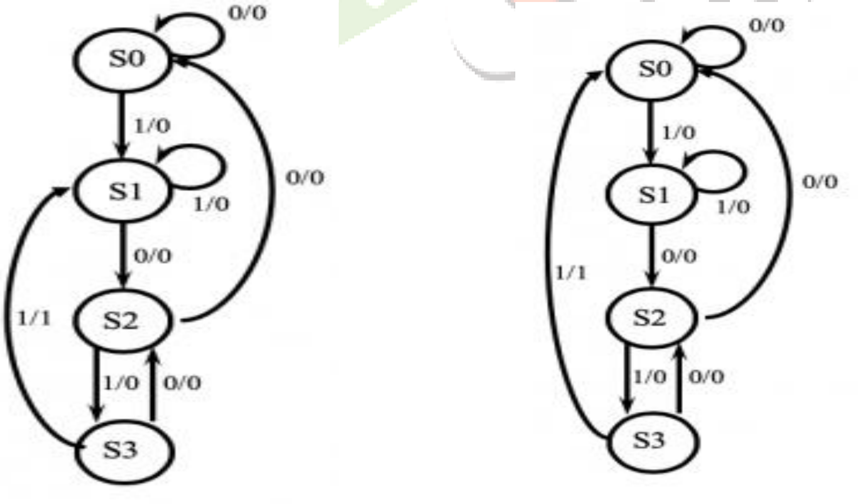
It is a sequential FSM which receives input a string of bits and generates an output [8]. The output goes high whenever the required sequence has been detected. There are two basic types of sequence detectors: **overlapping** and **non-overlapping.** In an over lapping sequence detector, **t**he last bit of one sequence becomes the first bit of next sequence where as in non-overlapping sequence detector the last bit of one sequence does not become the first bit of next sequence. In the following example, detection of the sequence 1011 using Mealy FSM is considered.
Examples:

Overlapping case
Input string: 010110110110
Output: 000010010010
Non-overlapping case
Input string: 010110110110
Output: 000010000010

4. IMPLEMENTATION

4.1 Sequence detector using Mealy FSM:

In Mealy FSM, the output depends on the current input and current state variables. The state diagrams for the sequence 1011 with and without overlap condition is shown in figure 4.1.



(a) Non-over lapping (b) Over lapping

Figure 4.1. State diagram for Mealy FSM

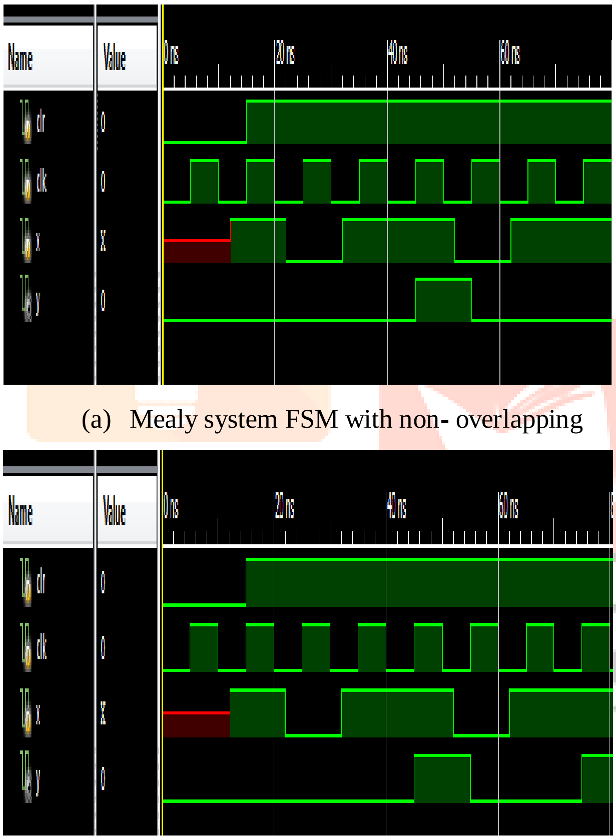
In Non-over lapping sequence, the output goes high after detecting all the four bits and goes to initial state for the next cycle. This is shown in figure 4.1(a) where as in overlapping sequence which is shown in figure 4.1 (b), the output goes high after detecting the all the four bits and last bit of the first cycle is first bit of the next sequence.

4.2 Design equations:

In the proposed work, Feynman and Fredkin gates are used to design all the components with minimum gates and garbage outputs. The prososed design is carriedout using D-FFs by assigning the binary values to the states, s0=00,s1=01,s2=10 and s3=11. The flip-flop input and output equations of the system for the non-overlapping system are DA = AB’X + BX’, DB = A’X + B’X, Y = ABX respectively where as for the overlapping system the design equations are DA = AB’X + BX’, DB = X and Y = ABX. The combinational logic and the memory elements D-FF are designed using Fredkin and Feynman gates.

5. RESULTS AND DISCUSSIONS

The behavior of the Mealy FSM was described using Verilog HDL and simulated using ISim in Xilinx environment. The simulated results are shown in figure 5.1. In the figure 5.1 (a), the input string is 10110111 and the output sequence is 00010000. whereas in figure 5.1 (b), the output sequence is 00010010 for the same input sequence. Here the input and output are changing with positive edge of clock pulse.

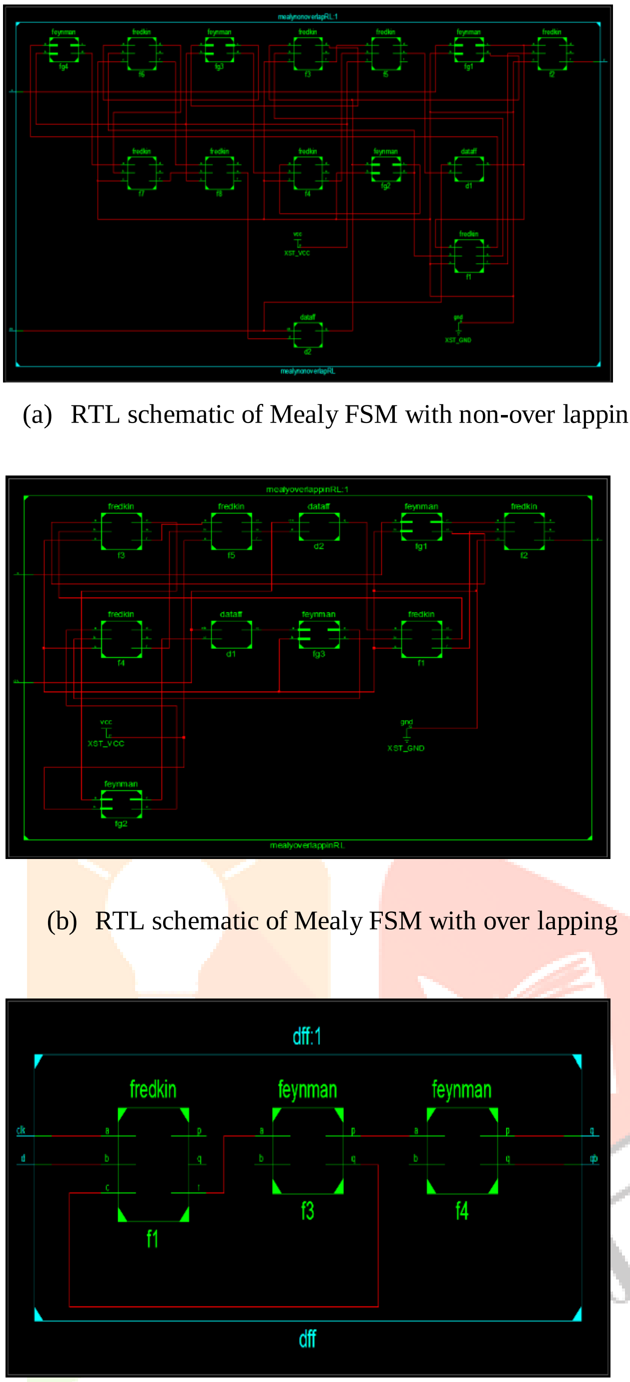


(a) Mealy system FSM with non- overlapping

(b) Mealy FSM with overlapping

Figure 5.1. Simulation results

The designs are synthesized using Spartan 6 device and these are shown in figure 5.2. The performance metrics such as number of reversible gates, number of constant inputs and number of garbage outputs are given in Table 1.



(a) RTL schematic of Mealy FSM with non-over lapping

(b) RTL schematic of Mealy FSM with over lapping

(c) RTL schematic of D-FF

Figure 5.2. Synthesis results

Table 1: performance metrics

Circuit		No. of reversible gates	No. of Constant inputs	No. of Garbage Outputs
Mealy FSM	Overlapping	14	12	14
	Non-over lapping	18	16	20

6. CONCLUSION AND FUTURE SCOPE

In this paper, a sequence detector 1011 with and without overlap using Mealy FSM was designed using reversible logic gates such as Feynman and Fredkin gates. The behavior is coded in Verilog HDL and the design is implemented in Xilinx ISE environment. It was simulated using lSim simulator and then synthesized using Spartan-6 device 6SLX45TFGG484-3. The proposed work is optimized in terms of number of reversible gates and garbage outputs. The proposed design will provide a new paradigm in the area of reversible circuits. One of the limitations of our work is that it does not calculate the delay and quantum cost. The next future step in the design of reversible sequential circuits is to investigate the synthesis of reversible sequential circuits with above parameters.

REFERENCES

[1] N. Sharma, R. Sachdeva, R. Yadav, U. Saraswat and G. Kaur, "Novel power efficient bit-wise sequence detector (Non-overlapping)," *2016* 3rd International Conference on Computing for Sustainable Global Development (INDIACom), New Delhi, India, 2016, pp. 2358-2361.

[2] R. Landauer, “Irreversibility and Heat Generation in the Computational Process”, IBM Journal of Research and Development, pp. 183-191, 1961.

[3] C.H. Bennett, “Logical Reversibility of Computation”, IBM J. Research and Development, pp. 525-532, November 1973.

[4] E. Fredkin, T Toffoli, “Conservative Logic”, International Journal of Theor. Physics, 21(1982), pp. 219-253.

[5] T. Toffoli., “Reversible Computing”, Tech memo MIT/LCS/TM-151, MIT Lab for Computer Science, 1980.

[6] Alberto LEPORATI, Claudio ZANDRON, Giancarlo MAURI," Simulating the Fredkin Gate with Energy Based P Systems", Journal of Universal Computer Science, Volume 10, Issue 5, pp 600 - 619.

[7] M. P. Frank, “Introduction to reversible computing: motivation, progress, and challenges”, In Proceedings of the 2nd Conference on Computing Fron-tiers, pages 385–390, 2005.

[8] M. Morrison and N. Ranganathan, "Design of a Moore finite state machine using a novel reversible logic gate, decoder and synchronous up-counter," 2011 11th IEEE International Conference on Nanotechnology, Portland, OR, USA, 2011, pp. 1445-1449,

[9] Stephen Brown and Zvonko Vranesic, “Fundamentals of Digital Logic Design with VHDL”, Tata McGraw-Hill

[10] M.Morris Mano and Michael D Ciletti, “Digital Design” Pearson Education

[11] Samir Palnitkar, “Verilog HDL”, Prentice Hall publication