Design Of Cricket Game and Display System Using Verilog HDL

Mogulluri Sreehari[1], P.Sreeram Sarath[2], P.Vinod Kumar[3] Fazal Noorbasha[4]

Department of Electronics and Communication Engineering,
KL University, Vaddeswaram, Guntur, A.P., India – 522 202
[1]smogulluri007@gmail.com, [2]roxstasarath@gmail.com
[3]paritalavinodchowdary@gmail.com, [4]fazalnoorbasha@kluniversity.in

Abstract

In this Paper we developed cricket game using Verilog HDL. This system contains linear feedback shift register (LFSR), counters and adders. The sequence of LFSR and counters are used as runs, balls and wickets. To display the total runs, balls and wickets we have used adder circuits. This system supports only game between two teams. Display system will show who is winner. We have implementation this system in two models, gate level logic circuit is designed using DSCH tool and FPGA circuit is designed using Verilog HDL. Here we have explained this system implementation and working principle with results.

Keywords— **Adder, LFSR, Verilog HDL, Counter, Display System**

I. INTRODUCTION

The creation of games was always in the mind of the engineers. From the start of engineering scientists always thought about how to create something to entertain themselves, not only the required research work. We chose this cricket game, because it one of the big classic games and it needs not so much drawing, but it's really much fun to play[1].

Random numbers are required in a wide assortment of utilizations, including information encryption, circuit testing, frame work reproduction and Monte Carlo technique [2]. In this paper we exhibit 4-bit random number generator utilizing Linear Feedback Shift Register. Using Linear Feedback Shift Register we will generate a sequence of random numbers and we will use this same numbers as runs and wickets. We will run this cycle for two times and it will be given to two players and finally compare these two team results and a winner is emerged by comparing who scored more [3]. If the score is tied we will compare wickets. If score and wickets are tied we will compare the number of boundaries scored and produce a winner.

For display system design, runs, wickets and balls are taken as inputs from keyboard, and the score is displayed as the real world display system in cricket grounds, televisions etc. Before going to implement this cricket game, we have implemented the total circuit using DSCH tool after we have used Verilog HDL code to implement this cricket game and display FPGA system [4].

II. PROPOSED CRICKET GAME

2.1 Game Design and Algorithm

The output of LFSR Flip Flops generates a random sequence [5]. The output of the random sequence generator obtained from LFSR assigned as wickets and runs and then

assigned variables like WICKETS and SCORE. If any wicket falls wickets will increase by one and if any batsmen scores run the score count will increase by the respected score. Now the game is played between two teams and to decide winner scores are compared.

The sequence we are generating is 4 bit sequence [6]. Therefore the resulting number range will be 0-15.So we assign 0-6 and 10-15 as Runs and 7, 8, 9 as wickets. 10-15 are assigned as 1-6 respectively. If any of 7, 8, and 9 occurs it will count as a wicket it is shown in Table 1.

LFSR Output	Reassigned Number(a)	Output
0-6	0-6	Score = Score+a;
10	1	Score=score+1;
11	2	Score=score+2;
12	3	Score=score+3;
13	4	Score=score+4;
14	5	Score=score+5;
15	6	Score=score+6;
7,8,9	R=1	Wickets= Wickets + 1;

Table.1LFSR state Vs score and wickets

The sequences of numbers are counted for runs and wickets. A 4 bit adder [7] is used to count runs and wickets. Runs is a 7 bit output variable and wickets is a 4 bit output variable. The score count will be stopped if the number of wickets is reached to 10. This cycle contains 120 balls that mean 120 clock pulses. Two Scores will be noted and compared to get a winner. Fig 1 illustrates the complete algorithm about the game design.

Display System is designed by taking the inputs like Dot Ball, Single, Four, Six, Wide, No ball, Wicket, Ball and if any one given input it will result in their respective output variables (Score, Extras, Wickets, Balls) for example, if any single or four or six is scored it will result on Score board, similarly Score, wickets and Balls will displayed on the score board.

We have used a binary number system and before going to display this will be converted into decimal number system. Thus, each and every variable is taken as an input is one bit and it results change of 7 bit or 4 bit output variables. If the input is given to single bit, it will increment one to the score variable which is a 7 bit variable, if wicket is incremented it will result in incrementing of a 4 bit variable Wickets. In this way it will result in incrementing in 4 bit variable extras if any input is given to wide or no ball.

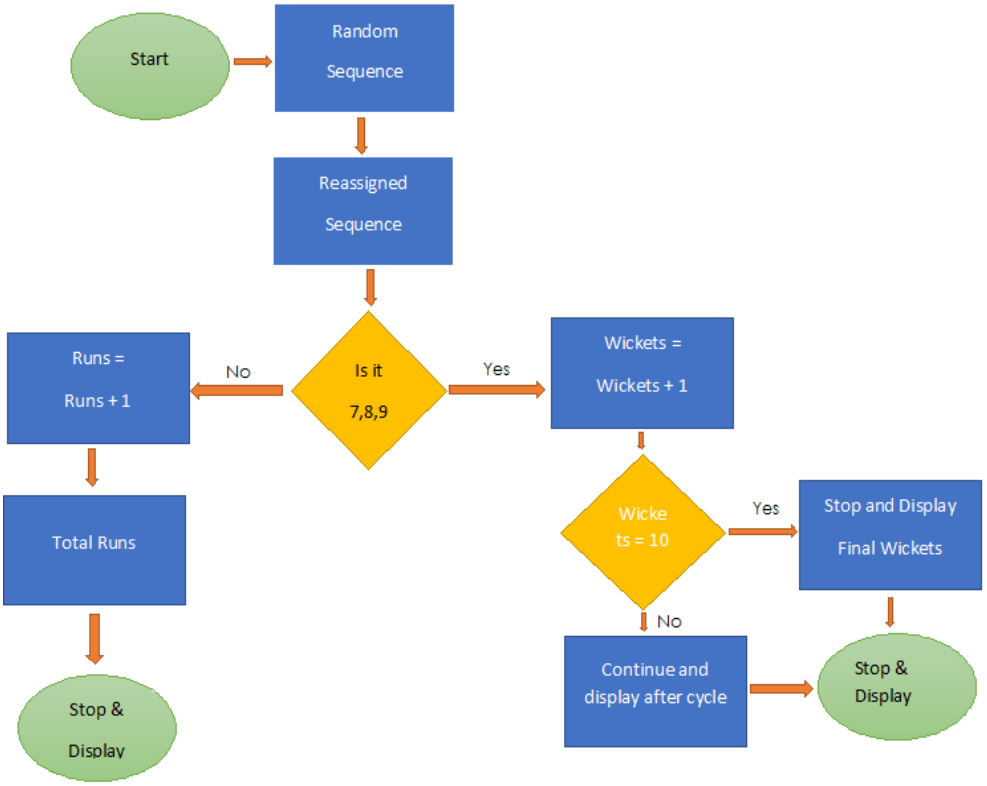

Fig.1. Algorithm of Game Design

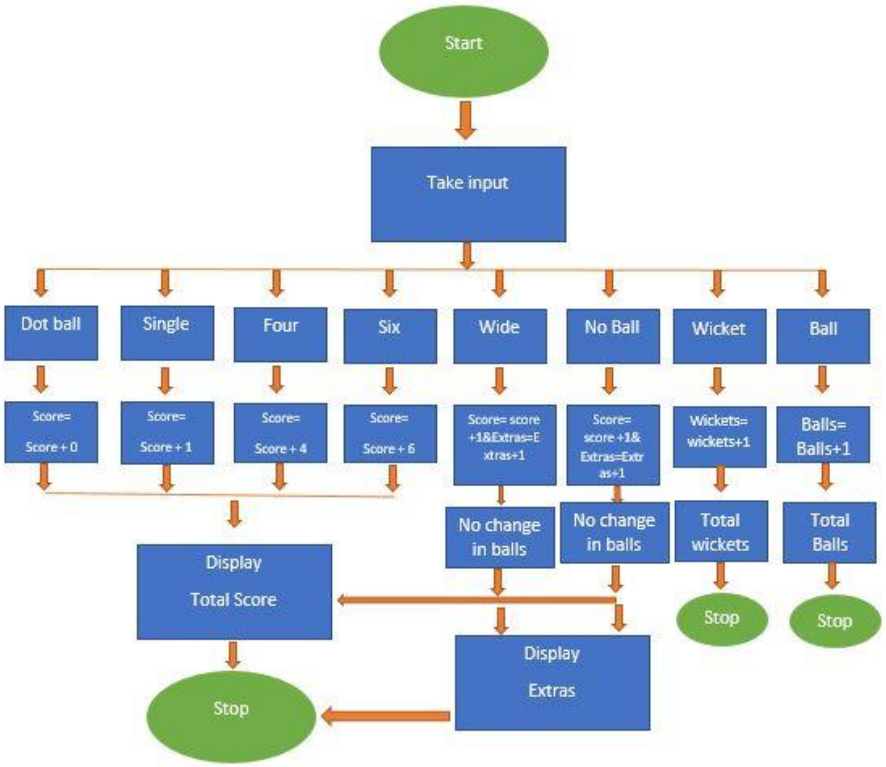

Fig.2. Algorithm of Display Design System

2.2 Display System Design and Algorithm

Likewise the total cricket match score can be calculated and the final score can be given in format or Runs – Wickets – Extras –Total Balls resulting in the Display System that is similar to the real life computerized Score Board that is displaying in the cricket matches. The inputs are given in test bench code. Fig 2 illustrates the complete algorithm about the display system design.

III. DESIGN OF CRICKET SYSTEM

3.1 FPGA Implementation

Linear Feedback Shift Register is used to design random number generator. LFSR is designed using flip-flops and it is used to produce a random number sequence as explained in Fig.3. LFSR is designed by using flip-flops and XOR gate .LFSR mainly works on previous state which is a linear function of its previous state. Table 2 gives the information about the next state values that are generated by previous state value.The RTL schematic of the simulated LFSR design in given in Fig.4 and the simulated output timing waveforms are given in Fig.5.

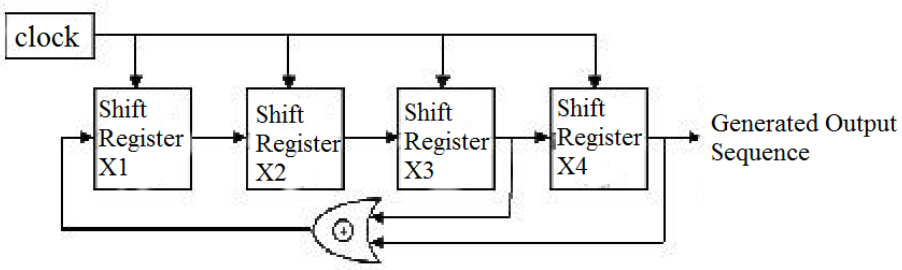

Fig.3. Block Diagram of LFSR

Present State (input)	Next State(output)
0000	0011
0001	0101
1010	1100
1111	0010
1001	0110
1010	1011
0101	1100
1110	0100

Table 2. LFSR Present State and Next State

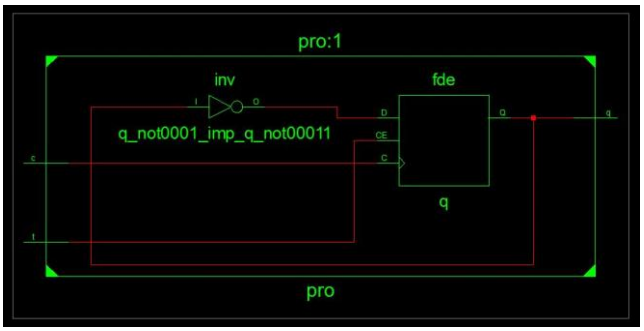

Fig.4.RTL Schematic of LFSR

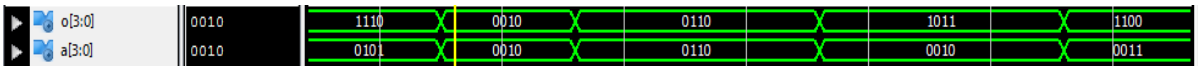

Fig.5. Waveforms of simulated LFSR (o – output of LFSR & a –re assigned values of LFSR)

We have used different types of adders as per our requirements. An 8 bit adder [8] for calculation of score in Game Design and Display System and 4 bit adder [8] for all the remaining variables in both Game Design and Display System Design.

A 4 bit synchronous up counter is used to calculate the number of wickets in game design. If any of the wickets (7, 8, 9 in the random number sequence generated by LFSR) falls the up counter will increase the count of the variable wickets. The counter will count up to 10 as the highest number of wickets in a innings have to be 10 that's why we are using a 4 bit up counter.

3.2 Gate Level Circuit Design:

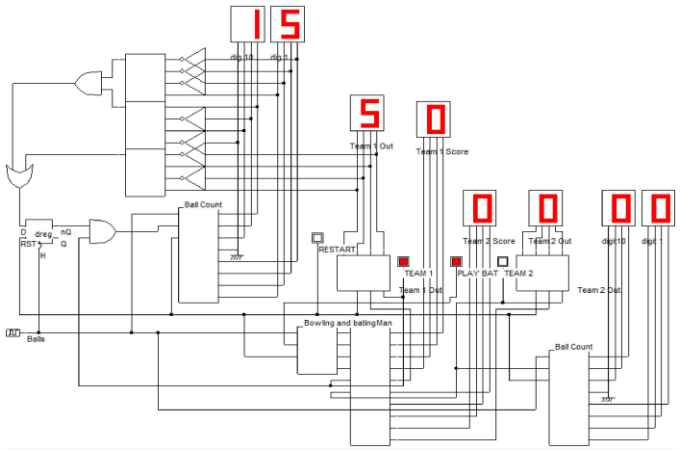

Fig .6. Gate Level Circuit Design of Game Design

Fig. 6 shows the gate level circuit of cricket game and display system. The total system is divided in to three parts, balls counter and disply, wickets counter and display and runs counter and display. For balls we have used upcounter and for wickets and runs we have used LFSR. For display we have used adder circuit and output of the adder circuit will display on the display unit. Finally we know the who is the winner.

IV. Results and Discussion

4.2 Cricket Game FPGA Design Results

Game display system RTL Schematic is shown in Fig.7. We can see the Block diagram of number of slices, number of flip-flops, number of look up tables, number of input output blocks and number of GCLKs used in the design. Table 3(Device Utilization Summary) gives the information about number of slices, number of flip-flops, number of look up tables, number of input output blocks and number of GCLKs used in the design.

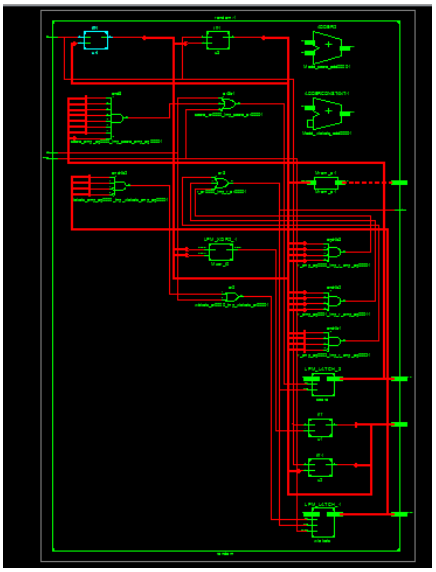

Fig.7. Game Display System RTL Schematic

Logic Utilization	Used	Available	Utilization
Number of Slices	11	4656	0%
Number of Slice Flip Flops	16	9312	0%
Number of 4 Input LUTs	25	9312	0%
Number of bonded IOBs	24	232	10%
Number of GCLKs	2	24	8%

Table 3. Device Utilization Summary

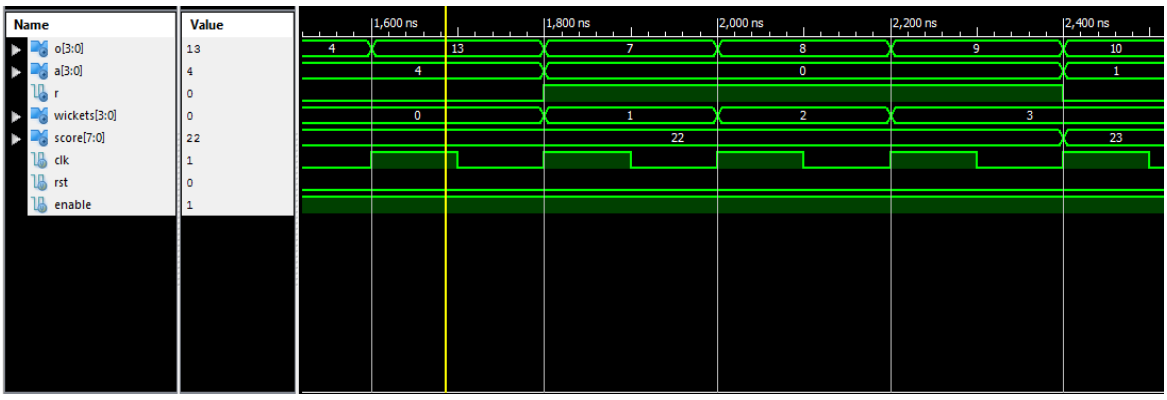

Fig.8. Timing waveforms of Game Design in Decimal Values

Fig. 8 shows the information about how the game is working and how the wickets and score is displayed in decimal form.

4.2 Display System Design results

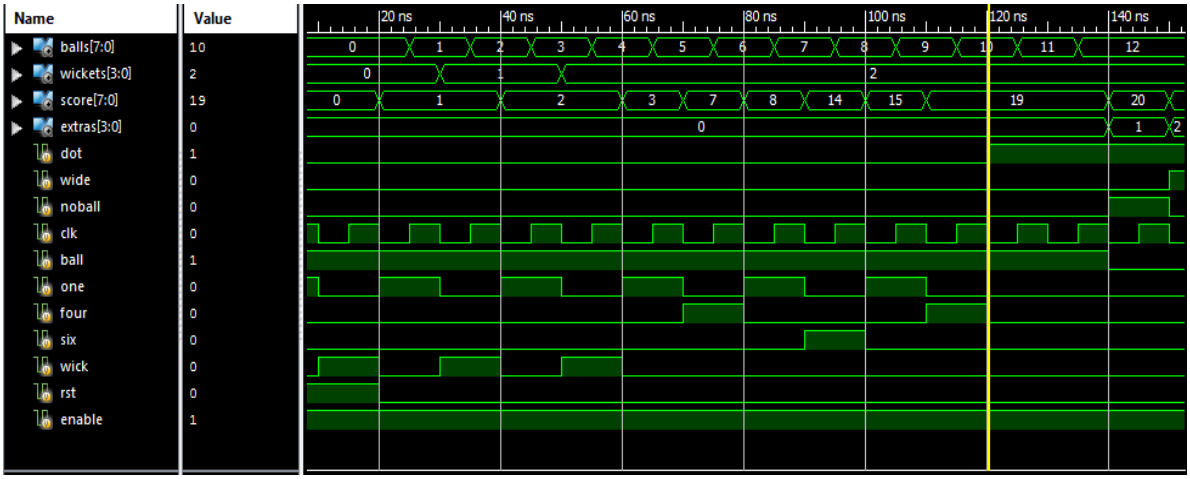

Fig.9. Timing waveforms of Display in Decimal Form

Fig. 9 shows the information about how the cricket display system is working and how the wickets, score, Extras and number of balls bowled is displayed and how the input and output variables vary is displayed in decimal form.

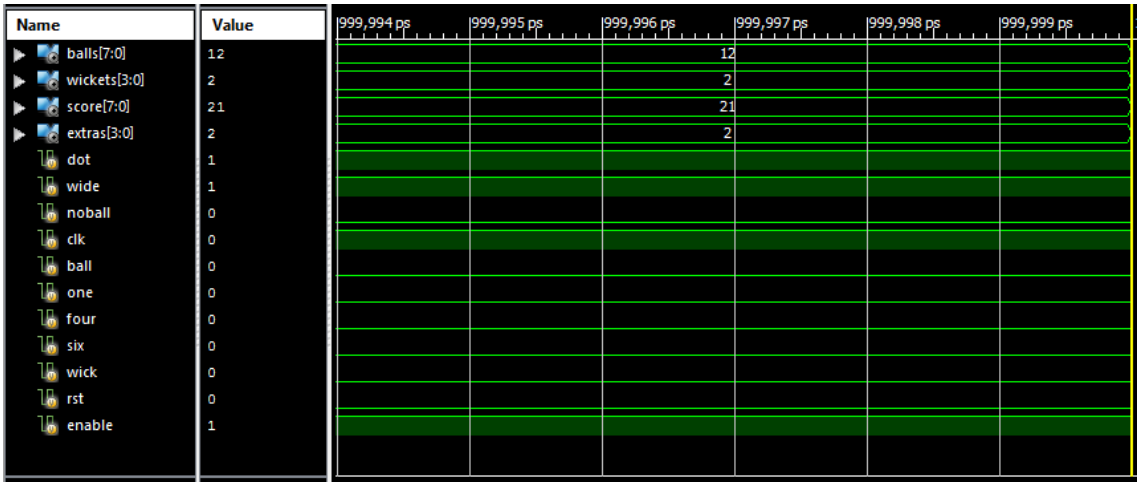

Fig.10. Timing waveforms of Final Score Displayed after completion of Game

Fig. 10 shows the information about how the cricket display system is working and how the wickets, score, Extras and number of balls bowled is displayed and how the input and output variables varied and it displays the final score that is displayed in the Display System Design.

V. CONCLUSION

As we could see we created a cricket game on FPGA. It was a good choice to create it on FPGA, because we had a suitable development board which has VGA port for interfacing a computer monitor and push buttons for controls. We also chose this platform, because this way we could create an ASIC, a standalone chip, by converting the Verilog with Mentor Graphic tools and we could create the layout of the chip for the final product. Thus by developing cricket game it will be addictive if a group of people play and it is user friendly. By developing Display System design it will be easy and user friendly to people who play cricket and suffering with score calculation issues while playing cricket. This will be very much cost effective and portable.

REFERENCES

[1] Kai, Liu, Yuliang, Yang, Yanlin, Zhu (2012): Tetris game design based on the FPGA. 2nd International Conference on Consumer Electronics, Communications and Networks, pp. 2925– 2928.

[2] Szabó, R., Gontean, A. (2014a): Pong Game on FPGA with CRT or LCD Display and Push Button Controls. Federated Conference on Computer Science and Information Systems, pp. 735–740.

[3] JiangHao, Li Zheying, “On the Production of Pseudo-random Numbers in Cryptography” in Journal Of Changzhou Teachers College of Technology, Vo1. 7 , No. 4, Dec. 2001.

[4] B.Umamaheswari and N.Rajeswaran ,“Simulation and Analysis of Genetic Algorithm Based on FPGA” in IJCER , Mar-Apr 2012 , Vol. 2 , Issue No.2 , pp 596-598 .

[5] Shruti Hathwalia, Design and Analysis of a 32 Bit Linear Feedback Shift Register Using VHDL Int. Journal of Engineering Research and Applications. www.ijera.com ISSN : 2248-9622, Vol. 4, Issue 6(Version 6), June 2014, pp.99-102 .

[6] K.Lavanya,V.Surendra Babu “Design of 8-bit Ripple Carry Adder Using Constant Delay Logic” International Journal of Electrical, Electronics and Computer Systems (IJEECS) , ISSN (Online): 2347-2820, Volume -2, Issue-8,9 2014.

[7] Maroju SaiKumar, Dr. P. Samundiswary, “Design and Performance Analysis of Various Adders using Verilog”, IJCSMC, Vol. 2, Issue. 9, September 2013, pg.128 – 138.

[8] C. Suba, S. Karthick, M. Prakash,” Analysis of Different Bit Carry Lookahead Adder with Reconfigurability in Low Power VLSI Using Verilog Code” International Journal of Innovative Research in Computer and Communication Engineering(IJIRCCE), Vol. 2, Issue 11, November 2014 .