A SIMPLE DESIGN OF VHDL BASED CHESS CLOCK INTEGRATED WITH CHESS BOARD

Sarita Sharma[1], Aditya Pundir[2], Manish Kumar Gupta[3]

[1]Department of Electronics and Communication Engineering, Jaipur National University, Jaipur
[2]Department of Electronics and Communication Engineering, Jaipur National University, Jaipur
[3]Department of Electronics and Communication Engineering, Vivekananda Global University, Jaipur

Abstract

In this paper, we are proposing the simplest algorithm with VHDL code to design a chess clock. This algorithm is based on basic simplified fundamentals of chess game. This design is been made for Modelsim simulator and can easily be implemented using FPGA kit. This algorithm is designed to get the least delays and possibilities of errors. RTL equivalent of this algorithm is also shown in the paper. This design can further be enhanced for multi player chess clocks for multiplayer rapid/blitz chess games.

Keywords: *Chess Clock, Technical Chess, Algorithms, VHDL, FPGA*

1. INTRODUCTION

Rapid/blitz chess clock is very interesting clock based on timer switch phenomena. This clock introduces some difficulties to the players. The basic difficulty is to synchronize a player's mind both with the complexity of the game and the clock trigging [1]. Development of a self switching chess clock is been in great interest of electronic designers [4-5]. There are some codes available in different language to implement the clock but all codes are lengthy and with large delays [2,3]. Considering all the facts, we are here presenting the simplest algorithm based VHDL code for the chess clock. This code runs with zero errors and optimized synchronization of player's move with the clock.

This design is based on chess rules. Though all we need to change in existing 'touch and move' rule with 'touch, pick and move' rule. According to 'touch and move', A chess player has to move the same piece which is he/she has touched. This rule allows them to keep the piece back at same position a while and think before making move. This rule is to be modified with 'touch, pick and move', i.e. once a player has touched and picked a player he cannot put it back at same position although he/she can take it in hands to think a while. This is all which will make this algorithm to work perfectly.

2. THE ALGORITHM

The beauty of blitz/rapid chess game is every player has to push the clock every time after he/she makes the move. Again, a move is considered complete only when the player put a piece back on the chess board. It is to be noted that, in the case of killing any piece of opposite side, it may happen that a player picks more than one player during his move (one his own, and one of opposition) but number of pieces put back on the board will be always one. This basic and simple concept is the key of this algorithm. This design deals with the logic handsomely with a very small programming.

Logic table for the chess clock can be written as:

RST	D[i], 64 bit			Output
0	any D[i] 1	0	→	0
1	any D[i] 1	0	→	Previous Stage
0	any D[i] 0	1	→	0
1	any D[i] 0	1	→	Toggle

We can write the above situation in algorithm as:

Let A[i] be the 64 bit array representing 64 blocks of the chess board.

Then

```
For every 'i'
    if RST = 1 then
        if any A[i] changes to 1
        then clock = toggle
        else
        clock = no change
    Else No change
```

The above mentioned algorithm is the simplest way to implement a chess clock. Since it has only one loop with only one conditional statement, it will run within the least delays and the clock will be optimized with the player's moves.

3. VHDL IMPLEMENTATION/RTL DESIGN

The VHDL code of the above algorithm can be done easily. This algorithm is the shortest possible algorithm for the purpose as per author's best knowledge. This code is been implemented using Xilinx and the Output generated from modelsim is as shown in this paper.

RTL structure generated from Xilinx is as given below:

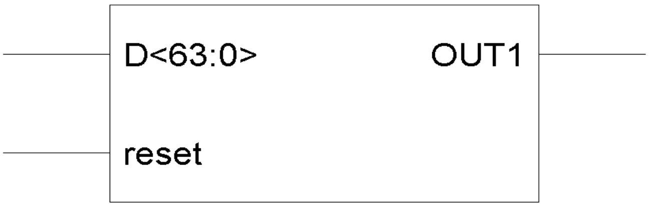

Fig 1: RTL Schematic for the proposed algorithm

This Design is expended in the diagram given below:

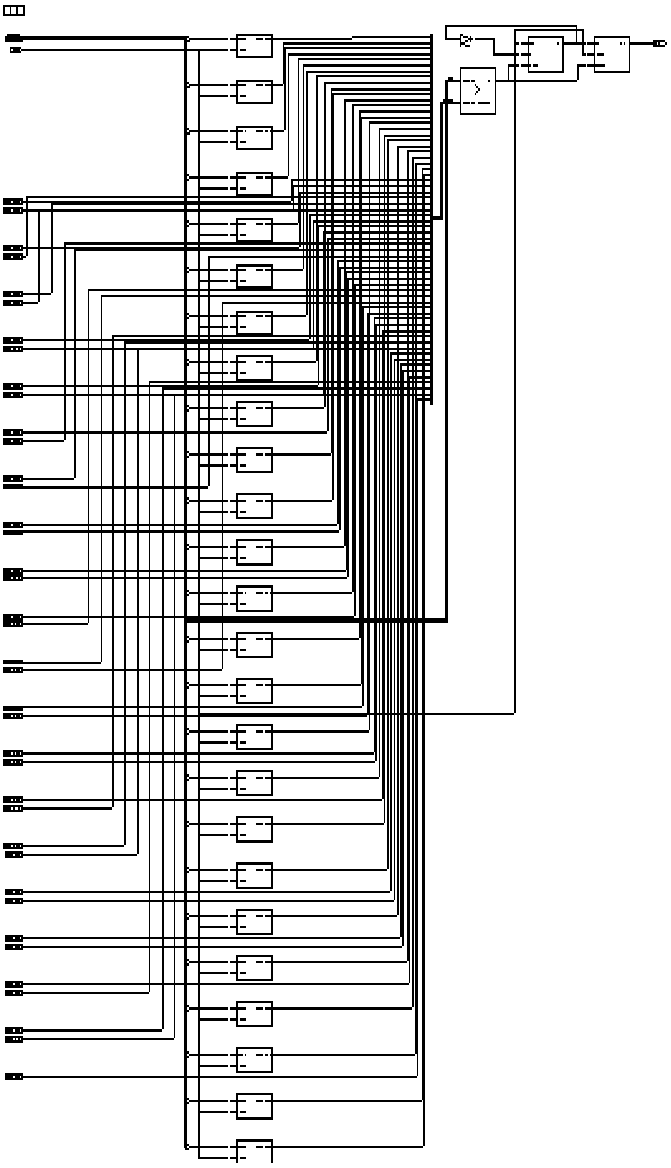

Fig 2: Expanded RTL for the proposed algorithm

4. RESULTS

The proposed algorithm is complete solution of the automatic chess clock and the results are shown in the previous section. This design is the simplest possible design as it has least complexity in the algorithm.

5. CONCLUSIONS

The proposed algorithm is very simple and easy to implement using any VHDL tool. This algorithm avoids lengthy coding and memory requirements as reported in earlier algorithms. The physical implementation of the algorithm will make rapid/blitz chess players relaxed from burden of pushing chess clock.

This all can be implemented by making a small changes in chess rules, i.e. ‘touch, pick and move’ instead of ‘touch and move’.

FUTURE WORK

Hardware implementation of the proposed algorithm is our first priority at present stage. Though, this algorithm can also be implemented for three player rapid chess clock and six player rapid chess clock. Another parallel approach is being done to find the possibilities in minimization the RTL design.

REFERENCES

[1] Marc Boulé and Zeljko Zilic, *An FPGA Based Move Generator for the Game of Ches*s, Custom Integrated Circuits Conference, Proceedings of the IEEE 2002

[2] Mustafa Mukadam, Qianliang Liu, Ke Ma, *Luminous Chessboard*, Final report, University Of Illinois At Urbana - Champaign

[3] Peter Minns and Ian Elliott, *FSM-based Digital Design using Verilog HDL*, John Wiley & Sons, Ltd. 2008

[4] Niels Grootscholten, Thusitha Mabotuwana, Winnie Jiang, David Wang, *Engineering Design 3CS, Fruits Basket*, Final Report, The University of Auckland, 2003

[5] Gurjit Kaur, Anil K. Yadav, Varun Anand, *Design and Implementation of Artificially Intelligent Microcontroller based Chess Opponent*, Proceedings of the World Congress on Engineering, London, U.K., Vol 1, 2010