ECSE-323
Digital System Design

Lab #5 – *System Integration for the Card Game* Winter 2017

Introduction .

In this lab you will put together all of the parts for the complete card game system and implement it on the Altera board.

Learning Outcomes .

After completing this lab you should know how to:

• Get a complete digital system working on the Altera board, and will have gained experience in implementing user interfaces.

Table of Contents .

This lab consists of the following stages:

1. Design and simulation of the computer player FSM
2. Design of the complete system Datapath
3. Design of the complete system controller FSM
4. Testing of the complete system on the Altera board
5. Writeup of the lab report

The Complete Card Game System

In this lab you will combine all of the parts you have developed in the previous labs and create the full Card Game system. Most of the work has been done in the previous labs, but in this lab you will need to construct the ***user interface*** circuitry, which permits a human player to control his/her play, and the ***computer player*** circuitry, which automatically makes a correct play during the computer's turn. The user interface will make use of the switches, buttons, and LED displays on the Altera DE1 board. You will also need to construct a controller (FSM) that will control the game play process.

The system should be designed to be able to play the ***Crazy-Eights*** game. The game should continue until either the ***Play Deck***, ***Human Player's Hand***, or the ***Computer Player's Hand*** has no cards. (see http://en.wikipedia.org/wiki/Crazy_eights for official rules)

Once you have constructed the full system, it will be downloaded to the Altera board hardware and tested (played!).

1. The Card Game User Interface

The interface of the system to the human player should have the following inputs and outputs:

USER INPUTS:

- Reset pushbutton - starts a new game
- Play pushbutton – plays the selected card from the Human Player's Hand
- Human Player Hand Scroller pushbuttons (up and down) – selects card to play
- Display Select (switches to select which value is being displayed on the 7-segment LEDs)
-

USER DISPLAYS (to be selected by the *Display Select* setting):

- Card on top of the Play Pile
- Selected Card in the Human Player's Hand (selected by hand scrolling input)
- Number of Cards in the Human Player's Hand (0-26)
- Number of Cards in the Computer Player's Hand (0-26)
- Number of Cards left in the Deck (0-52)
- Turn Indicator (i.e. is it the Human or Computer Player's turn)
- Invalid Play Indicator
- Game Over Indicator (when there are no cards in either the Deck or one of the players' hands)
- Winner Indicator (human win, computer win, or draw)

2. Design of the Computer Player FSM

You will construct an FSM that will control the play of the Computer Player.

The Computer Player operational sequence should be as follows:

- Wait for its turn (i.e. look at the TURN control signal). De-assert the DONE signal when the TURN signal indicates the Human Player's turn.

- Scan the cards in the Computer Player's Hand (stack) one at a time. For each card, check the validity of playing it, using the rules module from the previous lab. If a card is valid then play it (pop it and transfer it to the play pile), and assert the DONE signal (the main controller will then change the turn indicator to point to the human player). Stop scanning once a valid card is found.

- If no valid card is found, then draw a card from the Play Deck (i.e. pop the play deck and transfer (push) the card to the computer player's hand). Assert the DONE signal.

2. The Computer Player FSM Design

Begin the design process by drawing the state transition diagram for the computer player FSM. Show this diagram to the TA.

Next, write a VHDL description of the Computer Player FSM, and show it to the TA.

Compile the FSM and perform a functional simulation, entering suitable control signal values (e.g. TURN signal) by hand.

2. The Complete System Datapath Design

Now you will put together all of the parts for the complete system. Begin by deciding on what modules you will need to provide the proper functionality (i.e. the datapath, which will also include the computer player FSM).

You should layout the datapath using schematic capture in Quartus, so that you have a nice block diagram for the report.

2. The Complete System Datapath Design

You can't just add in modules at will, without considering the FPGA resource utilization. In particular, you should go back to your report for lab #3 and check how many Embedded Array blocks were used for your stack circuit. How many stacks can you fit in the Cyclone II FPGA, without modification of the circuit?

At first glance, it would appear that you would need to have ***4*** stacks in your system, one for the Deck, one for the Human Player's Hand, one for the Computer Player's Hand, and one for the Play Pile.

But we do not need a stack for the Play Pile, since only the top card is relevant to the play. So all that is needed to implement the Play Pile is a 6-bit register to hold the top card.

In addition, the stacks for the two Players' Hands need only have 26 elements, since each player can have at most 26 cards.

2. The Complete System Datapath Design

The Embedded Array blocks in the Cyclone II chip are used for the ***lpm_rom*** block employed in the ***pop_enable*** module, which is part of the stack circuit.

We can therefore minimize the usage of Embedded Array blocks by ***sharing*** a single ***pop_enable*** block amongst the three stacks.

That is, you will have one ***pop_enable*** circuit which is connected to all 3 of the stacks, but only one stack at a time is operated. You will need to have a control signal to select which stack is being operated at any given time. This control signal will be generated by the system controller FSM, to be developed next.

Once you have laid out the complete system datapath in Quartus, show its schematic diagram to the TA.

3. The Complete System Controller Design

The last step in the design of the complete system is the design of the system controller. This controller is an FSM that generates the control signals for the datapath module which produces the proper game playing behaviour.

Begin by drawing the state transition diagram for the FSM. Take care that you properly detect transitions on various external input and datapath status lines (i.e. to detect a rising edge, first wait for the signal to go low then wait for it to go high). Don't try to do too much in a single state, use as many states as you need. Show the state diagram to the TA.

Next, write a VHDL description of the system FSM, and show it to the TA.

4. Testing of the Complete System on the Altera Board.

Finally, connect the system controller FSM up to the datapath and to the external inputs and outputs. Then compile the full circuit and download it to the Altera board.

Demonstrate the functioning of your system to the TA. Make sure to demonstrate the following:

- **Resetting of the system** (i.e. beginning new games). Make sure that the human player's hand is randomized, and that each player gets the right number of cards to start with.
- **Proper Display of all required items.**
- **Detection of Invalid Plays.**
- **Correct Computer Player Play.**
- **End-of-Game Detection and Winner Determination**

Note the speeds at which the computer player plays. It may be difficult to check whether the computer is playing correctly!

5. Writeup of the Lab Report .

Write up a report for the complete ***Card Game*** system that you designed.

The report must include the following items:

- A header listing the group number (and company name if you have one), the names and student numbers of each group member.
- A title, giving the name of your system.
- A description of the system's features (i.e. what it does).
- A block diagram of the entire system.
- A description of how your system works, referring to the block diagram.
- A description of the user interface (e.g. a users guide to operation).
- A discussion of how the complete system was tested.
- A summary of the FPGA resource utilization.
- A conclusion section discussing problems or significant issues that arose during the design process, as well as a discussion of possible enhancements or extensions that could be made to your system.

Some items to remember when preparing the report for lab #5:

- All diagrams, tables, and figures should have captions describing the contents of the figure and be given a figure number.
- All such figures must be referred to in the text of the report. Do not just throw a figure into the report without referring to it. Figures should be used to augment the textual matter.
- Hand-drawn figures and diagrams are not acceptable!
- Turn off the grid in the Quartus schematics when making screenshots!
- Break large block diagrams into multiple smaller ones when inserting them into the report.
- Do not include long vhdl descriptions into the report. Instead, include the .vhd files in your report submission zip file.

Don’t wait until the last day to start writing the report. Start writing it early on in the lab 5 period. Remember, it is due on the last day of classes, and not one week later!

The report should be done in html or pdf (preferred), or in Microsoft Word, and uploaded to the myCourses site using the lab #5 assignment submission icon.

The presentation of the report (grammar, spelling, and clarity of the diagrams) will also be graded. *Professor Clark will grade the final reports.*

Make sure that you have uploaded ***all*** of the design files (e.g. .bdf and .vhd files) used in your project. Also include the device programming file (.sof or .pof file) so that Prof. Clark can download your design to his Altera board.

The report is due at midnight on the last day of classes, *Tuesday April 11*.
Late submissions will be accepted, but will be assessed a penalty of *10% mark per day*, (out of a possible maximum of 100%) up to (and including) *Friday April 21*. No submissions will be accepted after that date.

Grade Sheet for Lab #5

Winter 2017.

Group Number: _____.

Group Member Name: _____. Student Number: _____.

Group Member Name: _____. Student Number: _____.

Marks			TA Signatures
	1.	State diagram for the Computer player FSM	.
	2.	VHDL Description of the Computer player FSM	.
	3.	Simulation of the Computer player FSM	.
	4.	State diagram for the system controller	.
	5.	Demonstration of system reset (game begin)	.
	6.	Demonstration of display of required items	.
	7.	Detection of invalid play	.
	8.	Demonstration of correct computer play	.
	9.	Detection of end-of-game and determination of winner	.

Each part should be demonstrated to one of the TAs who will then give a grade and sign the grade sheet. Grades for each part will be either 0, 1, or 2. A mark of 2 will be given if everything is done correctly. A grade of 1 will be given if there are significant problems, but an attempt was made. A grade of 0 will be given for parts that were not done at all, or for which there is no TA signature.