**ECE324 Lab 8: Keyboard Music**

Name(s):

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

|  |  |  |
| --- | --- | --- |
| **Exercise** | **Course outcome** | **Grade** |
| Lab8 Demo | 2.a, 2.d, 5.c, 7.b | /15 |
| Lab8 Extra Credit | 2.a, 2.d, 5.c, 7.b | /3 |
| Lab8 Report | 2.a, 5.c, 7.b | /25 |
|  | **TOTAL:** | /40 |

2.a. Define engineering problems from specified needs for digital systems including implementation on FPGAs using HDL programming.

2.d. Produce FPGA designs that meet specified needs.

5.c. Collaborate with individuals with diverse backgrounds, skills and perspectives.

7.b. Employ appropriate learning strategies such as communicating with an expert, using external resources, experimentation, simulation, etc.

# Learning objectives

Learn how to modify a PS/2 keyboard interface, and an audio generation circuit, in a SystemVerilog HDL FPGA design.

# New code provided

Lab8\_KeyboardMusic.sv, ps2\_rx.sv, adsr.sv, ddfs.sv, sin\_rom.sv, sin\_table.txt, ds\_1bit\_dac.sv, and Lab8\_KeyboardMusic.xdc

# Procedure

1. Download, build, and observe Lab8\_KeyboardMusic’s starter design in operation.
   1. Synthesize and implement the starter code, and generate a programming file. Expect to get some warnings, but no errors nor critical warnings.
   2. With a USB keyboard connected to the USB connector next to the power switch on the Nexys4DDR board, download the programming file.
   3. With a speaker connected on the top right of the Nexys4DDR board, play a note, and lower the volume to your liking using the left Ctrl key on the keyboard (note for now the only way to raise the volume again is to download the programming file again).
   4. Starting with a finger on the P key, play a song such as “Mary Had a Little Lamb” or Beethoven’s “Ode to Joy” from his 9th Symphony, as if the computer keyboard were a piano keyboard (but don’t press more than one key at a time).
   5. You don’t need to demonstrate this yet.
2. Change Lab8\_KeyboardMusic.sv so that the note A5 will be played when pressing the backslash (\) key (between the Backspace and Enter keys). You don’t need to synthesize and demonstrate this yet.
3. Change the state machine in Lab8\_KeyboardMusic.sv so that the right Ctrl key will increase the volume, by uncommenting a line in state IDLE, and writing your logic to implement states EXT\_BYTE\_AFTER\_IDLE and BREAK\_BYTE\_AFTER\_EXT\_BYTE. The hexadecimal make code sequence for the right Ctrl key is E0 14, and its break code sequence is E0 F0 14. Synthesize and demonstrate its operation to the instructor or lab assistant.

# Extra Credit (up to 3 points)

Without resynthesizing, change the ADSR envelope to configure each of the attack, decay, sustain, and release phases to be one second each, and a sustain level ratio of 1/2, such that you can hear each phase. Demonstrate this to the instructor or lab assistant.

# Deliverables

1. Demonstrate operation of your completed FPGA design to the instructor or lab assistant.
2. Write a brief (1-2 pages) lab report including the following items:
   1. Cover sheet with names, course number, assignment number, grade box and ABET outcomes.
   2. A written description of your SystemVerilog code: describe the overall function of your design and the operation of each module, including any salient details.
   3. Report the number of Flip-flops and LUTs used. Also report the utilization percentage for flip-flops and LUTs.
   4. Report the design’s minimum clock period, and calculate and report the maximum possible clock frequency.
3. Document any SystemVerilog code you modified. Your SystemVerilog code ***must include header comments stating your names, date and class number.*** Any changes to the SystemVerilog code ***must also include comments explaining the operation of the code***.
4. Upload your written lab report ***in .pdf format***, upload all of the .sv and .xdf text files used for your solution (zipped together if you like), and then hit the submit button just once.

# References

Chu, Pong, *FPGA Prototyping by SystemVerilog Examples*, Wiley 2018, ISBN 9781119282662, Chapters 18-20.

# Warning to Students

In case you intend to use WSU’s keyboards for your term project, I found while developing this lab that some of the keys on the right side of the keyboards (to the right of the Enter key) didn’t work as documented in Chu’s book and on the web.