

GEORGIA INSTITUTE OF TECHNOLOGY
SCHOOL of ELECTRICAL and COMPUTER ENGINEERING
ECE 4813A / 8813A Spring 2018
Homework #8

Assigned: 28-Feb-18
Due Date: 12-Mar-18

For HW 8 you will implement the DTFS sound generator you designed in HW 7.

Turn In:

Turn in hw8_tb.v, hw8_top.v, defines.v, number.hex (USE THE EXACT SAME FILENAMES!)

Details:

1. DUT module declaration should use this exact same definition:

```
`include `./defines.v"
module hw8_top(
    input wire clk,
    input wire play,
    output reg [23:0] audio)
```

bit-width of audio generated is your choice but you should scale it to 24-bits before driving audio bus.

2. Do not hardcode any numbers in the code. Derive parameters from the following `define definitions in testbench. These definitions may change during grading and your code is expected to work:

```
`timescale 1ns/1ns
//define these values specific to DE1 board!
`define TCLK ?? //t nanoseconds time period of CLK
`define SAMPLING_RATE ?? //M Hertz audio
`define NUMBER_OF_DIGITS ?? //Number of lines in number.hex
`define TONE_DURATION 300 //in milliseconds
`define AUDIO_DEPTH 24 //N-bit audio
//CHANGE THIS:
`define USER `"jojujosephzajo`"
```

3. Use ram.v and number.hex to model the SD card. Number should be stored in number.hex under the run directory. Each digit should be in HEX format on a separate line for \$readmemh to work.

Eg. To store 911 in RAM, contents of number.hex:

```
9
1
1
```

4. Testbench: Use hw8_tb.v by filling the correct values for `define constants and other parameters. The TB will sample the DUT audio output signal at the audio sampling rate for DE1 board and log them in a python script <USER>.py. You may run the python script to generate the <USER>.wav. On successful simulation, it will dump a python script. Run the python script to generate .wav file corresponding to your DTMF output.

Python dependencies can be resolved by:

```
pip install NumPy  
pip install scipy
```