EECS 388

LABORATORY EXERCISE II

SPEAKERBUZZ

September 22, 2018

Benjamin Streit

LABORATORY OVERVIEW

This laboratory involves the implementation of a new task to be run on the Tiva TM4C1294 evaluation board with the TI Audio BoosterPack add-on evaluation board. This new task will generate a tone using the Digital to Analog (DAC) and speaker that is housed on the TI Audio BoosterPack evaluation board.

TONE GENERATION TASK

The structure of the tone generation task, named `Task_Speakerbuzz.c`, is as follows:

```
// Include DAC drivers
// Define state variables
// Initialize the DAC
// Initialize state value
// Enter endless while loop
    // If state variable is high, toggle state variables and write state to DAC
    // Else, toggle state variables and write state to DAC
    // Delay
    // Loop again
```

In more detail: The DAC drivers are included using `#include "Drivers/EECS388_DAC.h"`. State variables defined are `DAC_State` as a `uint32_t`, and `high` as a `bool`. The DAC is then initialized using the call `EECS388_DAC_Initialization();`, and `high` is set to `false`. Now the loop is entered, and both state variables are toggled depending on the current value of `high`, and the newly toggled `DAC_State` value is written to the DAC using the call `EECS388_WriteDAC( DAC_State );`. Afterwards, a delay is introduced using the call `vTaskDelay( ( x * configTICK_RATE_HZ ) / 10000 );`, where the value of `x` is determined and explained in the following Analysis section. Finally, the loop begins again.

In addition to the writing of `Task_Speakerbuzz.c`, this function had to be included in the `EECS_388_Program_Base_Fa18.c` file. This was done by declaring the function outside the `int main( void )` function, and within the function using a `xTaskCreate()` call.

ANALYSIS

For this laboratory exercise, exploration of the relationship between the speaker buzz frequency requested by the TA, and the task delay the `Task_Speakerbuzz.c` is to employ. Say

the requested frequency is `300 Hz`. The period is $\frac{1}{300\text{Hz}}$. The creation of the buzz sources from the toggling of the DAC. If the period is $\frac{1}{300\text{Hz}}$, the DAC must spend half of this period in each state, or $\frac{1}{600\text{Hz}}$. Meaning, the value passed into `vTaskDelay` must equal half the period. With this in mind, one can follow this process and solve for `x` for any requested frequency.

CODE SCREENSHOTS

Images of code discussed included below for reference.