**Digital Audio**

**Engineering 155 Lab IV Report**

**October 5th, 2021**

John Hearn

A picture containing text, office

Description automatically generated

**Introduction:**

The goal of this lab was to build a circuit to enable an I/O pin from the MCU to drive an 8-ohm speaker. We were to implement the timer functionality for this purpose by reading the datasheet and writing our own library in C using PlatformIO. Using this design, we were to read in an array of notes in the form of frequencies and durations to play Fur Elise.

**Design Methodology:**

The main component of the design for this lab came from the software side. It was determined that it would be effective to use two timers, TIM10 and TIM11, for calculating both the duration of the notes being played and the pitches that they would be played at.

It was necessary to create structs to access each field of the timers, shown in the Technical Documentation section. Looking through the reference manual, it was determined that the fields used within the configPitchT() and configLenT() functions were necessary for the configuration of the two timers.

Because TIM10 was used for the pitches, it determined that the timer frequency should be scaled down to 14 MHz from 84 MHz to produce a frequency able to be picked up by humans. To achieve this, we divided 84 MHz by 6, which meant setting the prescaler to 5, as shown below in Figure 1.

Text, letter

Description automatically generated

Fig. 1: PSC calculation TIM10

TIM11 was used for the note durations and similarly, it was determined that 1 MHz was the easiest frequency to make calculations with. We divided 84 MHz by 84 to achieve this frequency, meaning a prescaler value of 83. The calculation is shown below in Figure 2.

Text, letter

Description automatically generated

Figure 2: PSC calculation TIM11

After this, it was necessary to create functions updating the duration and pitch based on the notes passed in. These functions were used within a for-loop that iterated through all the notes to play the correct note for the correct durations.

**Technical Documentation:**

Diagram, schematic

Description automatically generated

Figure 3: Circuit schematic

Text

Description automatically generated

Figure 4: GPIO.c file

Text

Description automatically generated

Figure 5: RCC.c file

Text

Description automatically generated

Figure 6: Flash.c file

Text

Description automatically generated

Figure 7: TIM.c file

Text

Description automatically generated

Figure 8: main.c file

**Results and Discussion:**

The song played. The ending note lingers but otherwise it plays perfectly with no issues of tone.

If I were to redo this lab I would start earlier. This lab took everything from me to complete.

**Conclusion:**

I was successful in creating a system that plays Fur Elise through the speaker. This lab took 21 hours over two days to complete.