Text

Description automatically generated

**School of Electrical and Computer Engineering**

**Embedded System Design - ECE 5721 - Fall 2022**

**Final Project Progress Report**

**Richard Pinto, Daniel Funke, Jared Panizzoli, Dmytro Melnikov**

**November 11, 2022**

**Table of Contents**

[1. Introduction 3](#_Toc119101074)

[2. System Description 4](#_Toc119101075)

[2.1 Hardware Configuration 4](#_Toc119101076)

[2.1.1 Signal Generation Component 5](#_Toc119101077)

[2.1.2 Analog Filter Stage 6](#_Toc119101078)

[2.1.3 Data Acquisition and Processing Unit 7](#_Toc119101079)

[3. Team Member Updates 9](#_Toc119101080)

[3.1 Dan Funke 9](#_Toc119101081)

[3.2 Jared Panizzoli 9](#_Toc119101082)

[3.3 Richard Pinto 9](#_Toc119101083)

[3.4 Dmytro Melnikov 9](#_Toc119101084)

[4. References 10](#_Toc119101085)

[5. ​Appendix 11](#_Toc119101086)

[5.1 Appendix 1: DAC Initialization Code 11](#_Toc119101087)

[5.2 Appendix 2: ADC Initialization Code 11](#_Toc119101088)

[5.3 Appendix 3: SCG GPIO Initialization Code 12](#_Toc119101089)

[5.4 DAPU GPIO Initialization Code 13](#_Toc119101090)

# Introduction

Fatigued driving is a serious problem that results in thousands of crashes, injuries, and deaths every year. These crashes impose monumental physical, emotional, and financial hardships for the individuals affected by the crash, as well as economic and logistical complications for trucking companies, insurance firms, and state and local governments.

Research has shown the Electroencephalography (EEG) is one of the most reliable tools for detecting sleep onset while driving [1]. It is possible that a device that detects the onset of driver fatigue and implements some form of corrective action could help mitigate the risk of driver drowsiness and thereby reduce the likeliness of crashes.

# System Description

For this project, we propose to design and prototype an EEG data acquisition and feature extraction device to detect drowsiness in drivers. The project will consist of three main parts: a surrogate “brain” to produce test signals to feed into the system, an analog filter stage for initial signal conditioning, and a digital filter, feature detection, and output stage. The relationship between these three blocks is depicted graphically in Figure 1.

Diagram, schematic

Description automatically generated

Figure 1: High-level system block diagram

The left-most block, representing the surrogate brain, will consist of a FRDM-KL25Z development board whose sole purpose will be to produce simulation waveforms to be collected for analysis by the fatigue detection system. The surrogate brain will produce a series of complex waveforms comprised of pure sine waves of varying frequencies and amplitudes mixed with random, high-frequency noise to simulate real EEG. These signals will be generated by the device’s onboard digital-to-analog converter peripheral.

The middle stage consists of some simple analog signal conditioning. This includes input buffers and analog filters. Normal human EEG activity typically falls within a frequency range of 1-30Hz, so a low-pass filter will be developed and implemented to attenuate high frequency noise before passing the signals to the digital filter stage.

The final block depicted in Figure 1 is the Digital Filter/Feature Detection/Output stage. This block will also consist of a FRDM-KL25Z development board. We will utilize the four differential channels of the device’s analog-to-digital converter (ADC) to digitize the signals as they exit the analog filter stage. We will then implement some software filters to further reduce high frequency noise before performing a Fourier transform on the input signals.

## Hardware Configuration

This section describes the individual aspects of the project that utilize various aspects of the MKL25Z development board. This section is subdivided into two parts, with the first part describing the signal generation component (SGC) and the second describing the data acquisition and processing unit (DAPU)

### Signal Generation Component

The first segment of our project, the Signal Generation Component, serves as an artificial brain that produces the surrogate EEG signals that will be consumed in the later stage. This segment utilizes the MKL25Z’s onboard digital-to-analog converter (DAC) to produce a test waveform. This waveform will be a composite sinusoidal signal comprised of four pure sinusoid waves with frequencies of 3 Hz, 5 Hz, 11 Hz, and 16 Hz to represent Delta, Theta, Alpha, and Beta EEG waves, respectively. Sample code for initializing the DAC can be found in Appendix 1: DAC Initialization Code.

This signal will also include high frequency noise produced by adding random values to the composite waveform. This will simulate real-world conditions in which small EEG signals are obscured by ambient electromagnetic interference.

The instantaneous voltage of each component waveform will be calculated using the standard C library function sin(x). A counter from the Timer and Pulse Width Modulation peripheral will be utilized to generate an interrupt that will update a variable that represents time. This time variable will be passed to the individual sin() functions. The outputs of the sin() functions will be summed together and then scaled by the desired peak output voltage. Every time a new value is calculated, the DAC output will be updated. The signals we are generating have very low frequencies. Consequently, the rate at which the DAC output needs to be updated to generate a coherent waveform also relatively low. We chose a factor of 10x the fastest waveform (160Hz, or every 6.25 ms) to configure our update timer.

The Signal Generation Component will also provide a user interface to control the relative power distribution amongst the component waveforms. Increment and decrement buttons will allow the user to allocate signal power to different bands to simulate various levels of drowsiness. To implement this interface, GPIO pins will be configured to read input from external buttons.

A top-level block diagram of this component is shown in Figure 2.

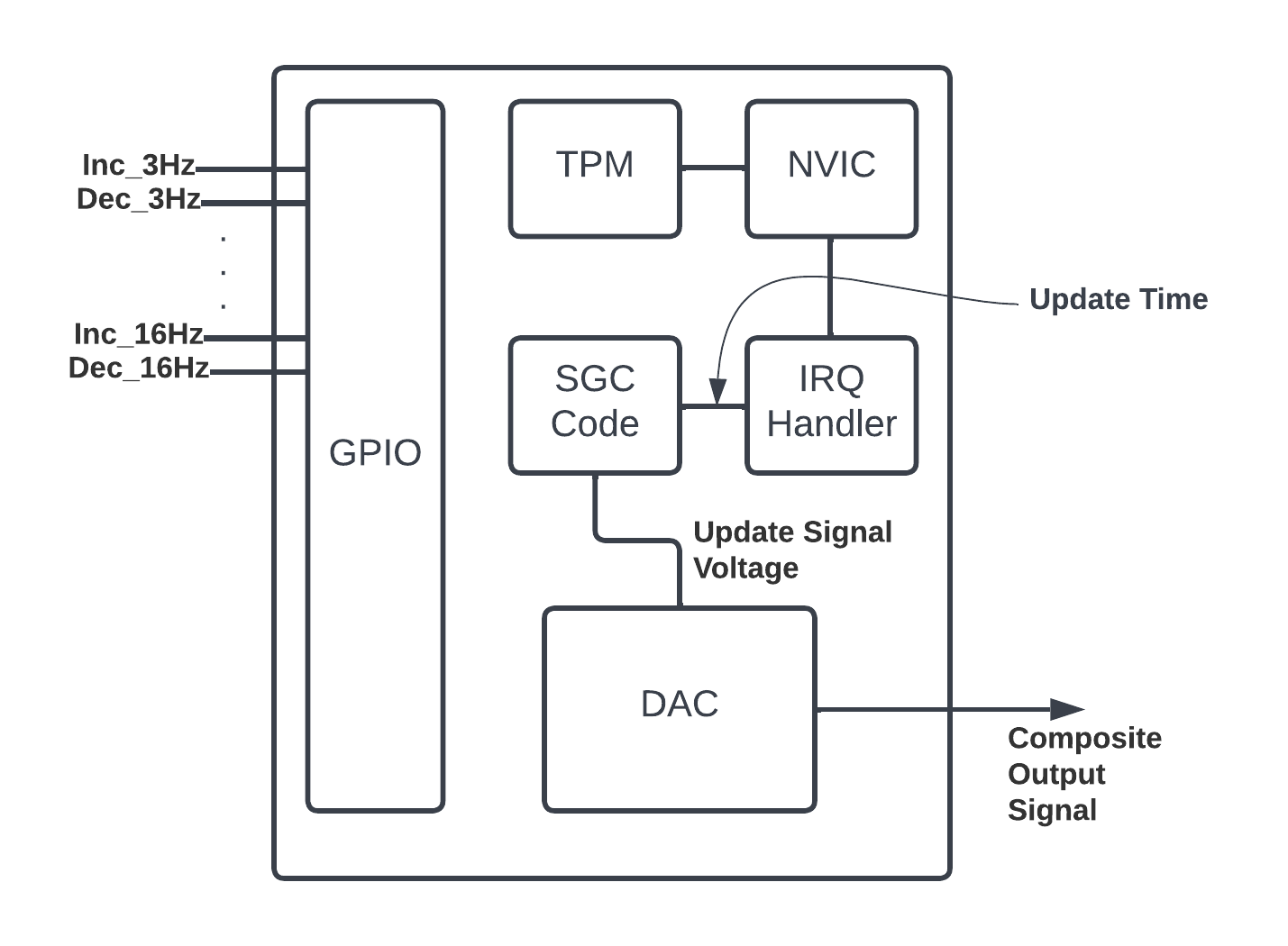


Figure : SCG Architecture

### Analog Filter Stage

The analog filter stage must consist of a low-pass filter in order to reduce high frequency noise on the signal. Since normal human EEG activity falls between the range of 0-30Hz, the 30Hz becomes the target cutoff frequency of the filter. A simple implementation of a low-pass filter consists of a resistor and capacitor in series. The signal will then be measured across the capacitor. An example circuit is shown below in Figure 3.

Diagram

Description automatically generated

Figure : RC Low-Pass Filter Circuit

In figure 2, a low-pass filter circuit is constructed using a resistor and capacitor in series. To calculate the impedance values, the equation shown in Figure 4 can be used.

Figure : Formula to Calculate RC Values

The formula to calculate the cutoff frequency of the low-pass filter can be algebraically manipulated to solve for the target resistor value.

Figure : Modified Formula to Calculate Resistance

Since our target cutoff frequency is 30Hz, it will be a constant in the calculation. The capacitor value will also be chosen to be 100nF. Based on this formula, the calculated resistance is 53kΩ.

### Data Acquisition and Processing Unit

The Data Acquisition and Processing Unit will primarily make use of the MKL25Z ADC peripheral, as its main function is to acquire input signals from the SCG after they’ve gone through the first stage of analog filtering (described in an earlier section). A timer will force the DAPU to periodically read the ADC to acquire a signal sample and append it to an array. Once the array is filled, a discrete Fourier transform (DFT) will be performed to calculate the frequency power spectrum. A block diagram of the intended program sequence is shown in Figure 6. Additionally, example code for initializing the chip’s onboard ADC is shown in Appendix 2: ADC Initialization Code.

Diagram

Description automatically generated

Figure : High Level Program Sequence

Based on the results of the DFT, the DAPU will produce an output that will alert the user should a drowsy state be detected. This output will take the form of flashing lights and an audible alarm. These elements will be controlled by toggling GPIO pins. Example code for initializing this peripheral is shown in Appendix 3: SCG GPIO Initialization Code.

# Team Member Updates

## Dan Funke

Dan is currently working on implementing the Signal Generation Component. He has developed a top- level block diagram of the individual components that need to be configured to generate the test signals for use in the Data Acquisition phase. His intention is to develop the software modules to control the peripheral modules and produce unit tests to confirm their functionality. He will the integrate the modules and test their system-level functionality by measuring the output waveforms using an oscilloscope. He plans to work closely with group member Dmytro Melnikov to test this functionality in the lab.

## Jared Panizzoli

Jared is currently working on design and implementation of the analog filtering stage. The necessary filter is chosen, and impedance values calculated. His next goal is to work with the team to ensure that the filter is properly receiving the signal from the signal generator and outputting the refined signal to the data acquisition unit.

## Richard Pinto

Richard has worked with Dan on the system design/block diagram and began creating a high-level version of the detection logic that will run on the data acquisition unit. The next goal is to research implementation of the Fourier transform in C, as well as the implementation of the detection logic running on the data acquisition unit.

## Dmytro Melnikov

Dmytro is working on implementing button input portion of the SCG module, as well as the alarm output portion of the DAPU. This includes configuring and testing the GPIO components of both project segments. He is also working with Dan Funke to verify and test the DAC component of the SGC.

# References

|  |  |
| --- | --- |
| [1] | Z. Mardi, S. N. M. Ashtiani and M. Mikaili, "EEG-based Drowsiness Detection for Safe Driving Using Chaotic Features and Statistical Tests," *Journal of Medical Signals and Sensors,* vol. 1, no. 2, pp. 130-137, 2011. |
| [2] | NXP, "KL25 Sub-Family Reference Manual, Rev. 3," 2012. |
| [3] | J. M. Wise, K. Heaton and P. Patrician, "Fatigue in Long-Haul Truck Drivers: A Concept Analysis," *Workplace Health & Safety,* vol. 67, no. 2, pp. 68-77, 2019. |
| [4] | M. M. Mitler, J. C. Miller, J. J. Lipsitz, J. K. Walsh and C. D. Wylie, "THE SLEEP OF LONG-HAUL TRUCK DRIVERS," *New England Journal of Medicine,* vol. 337, no. 11, pp. 755-761, 1997. |

# ​Appendix

## Appendix 1: DAC Initialization Code

The following code snipped was produced using the CMSIS header file and the NXP Technical Reference Manual for the KL25Z [2].

Text

Description automatically generated

## Appendix 2: ADC Initialization Code

The following code snipped was produced using the CMSIS header file and the NXP Technical Reference Manual for the KL25Z [2].

Text

Description automatically generated

## Appendix 3: SCG GPIO Initialization Code

The following code snipped was produced using the CMSIS header file and the NXP Technical Reference Manual for the KL25Z [2].

Table

Description automatically generated

## DAPU GPIO Initialization Code

The following code snipped was produced using the CMSIS header file and the NXP Technical Reference Manual for the KL25Z [2].

