ELT-3050 Final Project

5/1/21

Kevin Kunker, Josh Remillard

**Preliminary Design Proposal**

**Overview:**

The purpose of this this project is to produce an arbitrary waveform generator (AWG). AWG’s can synthesize a general selection of waveforms usually ranging from sine, square, or triangle patterns.

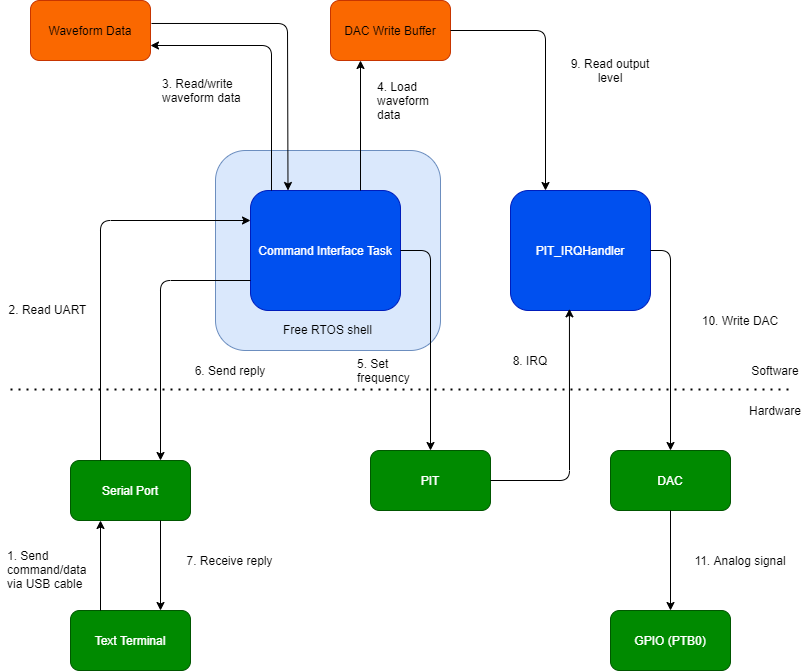
They can also produce waveforms which are custom tailored to meet certain criteria an end user may require.

**Scope of project:**

An AWG will be created using the FRDM-KL25Z development board, along with the accompanying VTC shield board. The AWG will have the ability to generate three pre-defined waveforms, as well as two user specified arbitrary waveforms. The AWG will have four selectable output frequencies, each having a maximum voltage output of 3.3 volts. AWG functionality will be controlled via serial communication using predetermined commands.

**Project Design:**

The initial design for the AWG consists of both hardware peripherals, and source code incorporated into a Free RTOS environment. The Open SDA serial port is used to acquire commands, as well as reading in ARB waveforms. Serial data will pass through a stream buffer to a task which will convert it into appropriate tokens. Command tokens will control the waveform output, whereas ARB tokens will be loaded into the specified ARB buffer.



**Figure I: Block Diagram of AWG**

The AWG shall be controlled by a text terminal such as TeraTerm which can send commands and arbitrary waveform data to the microcontroller. The commands/data travel via a USB cable to one of two serial ports on the KL25Z.

The Command Interface Task is a task created in the main function using FreeRTOS, so that other possible tasks can run concurrently. It processes commands and reads/writes waveform data or sets the timer frequency if necessary. The Command Interface Task also has the important function of loading the DAC write buffer with the correct waveform data to be sent to the DAC.

The PIT\_IRQHandler is the KL25Z’s standard ISR for the PIT. In this case, it is used to write the DAC output at a particular frequency. The PIT\_IRQHandler is triggered by the PIT, and then reads one integer in the DAC write buffer at the current index. The PIT\_IRQHandler then writes the level to the DAC and updates the index. The index starts at 0 and wraps back to 0 after the end of the write buffer is reached.

**Project Timeline:**

This proposal will be presented for review on 05/01/2021. Upon approval, a functional prototype will be developed for a final review with an anticipated date of 05/12/2021. Any necessary revisions will be made at that time. The finished AWG, including documentation will be submitted 05/21/2021.