sdmay19-11: MIDI Zeusaphone (Singing Tesla Coil)
Week Report 10
October 31 - November 7
Client
Joseph Zambreno
Advisor
Craig Rupp
Team Members
Gunnar Andrews — *Webmaster*
Leo Freier — *Interrupter and Micro Controller Lead*
Luke Heilman — *Technical Architect*
William Brandt — *Pulse Width Modulation Expert*
Greg Harmon — *Tesla Coil Construction Expert*
Jacob Feddersen — *Communications Specialist*

Summary of Progress this Report

- Fixed timing issues in pulse generation using hardware-implemented Pulse-Width Modulation
- Updated status report errors and image re-formatting
- Started researching hosting a GUI on the PI, also created scripts to run C code

Past Week Accomplishments

- Raspberry Pi Timing Issues Resolution - Jacob Feddersen
    - Previously, we were using a timing loop to generate the output wave
        - Timing could be interrupted by kernel
        - Not acceptable for our requirements
    - Use hardware-implemented PWM to generate signals instead
        - Pros: perfect timing accuracy
        - Cons: limited to two channels, Raspberry Pi only has two PWM
- Virtual MIDI Input Device - Luke Heilman
    - We needed a way to test the program that receives live MIDI input from a keyboard
    - Solution was to use the RtMidi library, which we used to receive live MIDI input
        - An "RtMidi out" object is used, which the OS interprets as a connected MIDI device
        - This object can then output MIDI messages, which are picked up by the OS
        - This output is easily sent to the MIDI keyboard app for testing
        - The testing itself uses the Python wrapper library for the RtMidi C++ library
- IEEE Standard for Safety Levels with human exposure - Greg Harmon & William Brandt
    - For frequencies between 3kHz and 5MHz, the max possible exposure for magnetic field strength ranges from 182 A/m to 163 A/m RMS. The range for Electric field is 614 V/m RMS.
- Updated team pictures on website - Gunnar Andrews
    - Trying to work on updating the team pictures on the website so that they don't look awful
- Created a checklist for hosting a GUI on the Raspberry PI
    - Will have to first host a website, then create scripts that can be ran from the website so that we can

access the code from any PC, as well as make/run the code on any PC.
- Scripts to run C code - Gunnar Andrews
    - Along with researching the GUI I noticed I would need to write scripts that can run the C code on the PI
    - Started experimenting with how to do this in python, that way it works well on the PI
- Worked on scripts and GUI prototype - Leo Freier
    - Created a script to run the wave program. It takes in an argument that will be the .mid file to run the program with. I tried to make it modular so implementing it in a GUI will be easier and users will be able to select and load .mid files using the source folder.
    - Prototyped a GUI in Java, just to see what is possible in a GUI. We likely won't make the GUI in Java, but I was just experimenting with how to go about creating it.

Pending Issues
Nothing to report.

Plans for Upcoming Reporting Period
- Gunnar Andrews
    - Finish updating team pictures on website
    - Work with Leo to create a hosted GUI for running code from the PI on a PC
    - Create make files for all C code that needs to be compiled and ran
- Leo Freier
    - Read more of the OneTesla manual.
    - Test the various potential solutions on simulating multiple channels.
    - If the coil does not arrive, continue GUI work.
- Greg Harmon
    - Create a procedure to follow when testing and analyzing the device
    - Research the effects of EM waves on pacemakers and other life-supporting devices
        - Search for datasheets/manufacturer-provided data
    - Simulate the driver circuit
    - Look into construction of a faraday cage
        - Size, specific specs, material
- William Brandt
    - Design test cases
    - Work on design for control circuit
- Jacob Feddersen
    - Continue working on driver software, based on the research from this week
    - Work on design for circuit to combine PWM output pins into a single signal
- Luke Heilman
    - Try to get the test scripts for the virtual MIDI device to pass the auto-testing Git pipeline
    - Start creating interrupter design

Individual Contributions

Team Member	Contribution	Weekly Hours	Total Hours
Gunnar Andrews	- Found a checklist for hosting a website on the Raspberry PI - Started updating image sizes on team page of website - Updated website with status reports and changed errors with dates on them - Started writing scripts to execute C code	6	52.5
Leo Freier	- Found some more potential solutions on the multiple channel simulation. - Created scripts and a basic GUI for running the wave program. - Read into the manual and a tutorial on building the OneTesla	6	52
Luke Heilman	- Continued implementation of virtual MIDI device for testing - Began viewing interrupter circuits	6.5	58.5
William Brandt	- Researched driver circuits - Researched maximum permissible exposure to find safe levels	8	49
Greg Harmon	- Read through parts of the IEEE Standard for Safety levels with RF EMF. - Found a max possible exposure level for Magnetic and Electric Fields - Refined tests for the tesla coil to ensure proper operation - Researched test equipment	6	58
Jacob Feddersen	- Help with issues with virtual instruments for testing - Research accurate waveform generation for the Raspberry PI - Implement Hardware-based PWM	7.5	60.5

Gitlab Activity Summary

commit 06f364d66bd4960203f77b9935a9c7ad69f7b5b8 (origin/pi_generate_wave)
Author: Leo Freier <lmfreier@iastate.edu>
Date: Wed Nov 7 18:44:22 2018 -0600

    Addes script with more appropriate name.

commit a05fbf4245b56d84714fa8ba0e762f8154ec2d44
Author: Leo Freier <lmfreier@iastate.edu>
Date: Wed Nov 7 18:37:27 2018 -0600

    Added the linux script rough draft.

commit 14046471cc2aa945c22427962feb2a7501938e2b (origin/make-tests, make-tests)
Author: Jake <jtfedd@iastate.edu>
Date: Tue Nov 6 21:40:00 2018 -0600

    Update automated scripts to use a different socket file

commit 0eeb563d94e84cebba3243a09f29e8ead232ce01
Author: Jake <jtfedd@iastate.edu>
Date: Tue Nov 6 21:38:38 2018 -0600

    Add socket file parameters to test scripts

commit 93b8e4c9381cf6b2777748cf00cbfd5a12b72b9d
Author: Jake <jtfedd@iastate.edu>
Date: Tue Nov 6 20:49:32 2018 -0600

    Add socket parameter to driver emulator

commit 8d0addec101841befcf745233a921d5a0362e26e
Author: Jake <jtfedd@iastate.edu>
Date: Tue Nov 6 20:32:03 2018 -0600

    Add socket parameter to midi file app

commit 81371f7bd5e6b0011d86e0d52f34e149ee7cc982
Author: Jake <jtfedd@iastate.edu>
Date: Tue Nov 6 20:22:12 2018 -0600

    Add socket parameter to captureMidiStream

commit db9f9a541d491a427ea1a4ff7272669456dda121
Author: Jake <jtfedd@iastate.edu>
Date: Tue Nov 6 19:34:45 2018 -0600

    Fix makefile so it updates midi keyboard app correctly

commit d64dc8a93c9fc1918abf622ac6c218fb282795c4
Author: Leo Freier <lmfreier@iastate.edu>
Date: Tue Nov 6 17:02:22 2018 -0600

   More multi-channel implementation attempts.

commit 5912b05174324e71536afe18008e7e9deb1a2a4c
Author: Jake <jtfedd@iastate.edu>
Date: Tue Nov 6 15:28:40 2018 -0600

   Adjust sleep time in keyboard test

commit afdad2e240e628741474540e4740b8e5f7c49088
Author: Jake <jtfedd@iastate.edu>
Date: Tue Nov 6 15:24:24 2018 -0600

   Fix gitlab-runner permissions

commit 6ac83a8b65de49e705b584f51348abe9b22a209a
Author: Jake <jtfedd@iastate.edu>
Date: Tue Nov 6 09:13:35 2018 -0600

   Fix midi_keyboard_test.py with async call

commit f7842b7d337e4e3ac42da42f25b4274d561a1cfd
Author: Jake <jtfedd@iastate.edu>
Date: Tue Nov 6 00:18:58 2018 -0600

   Add midi_file_app_test.py

commit d096987314f8fe8d0389b5ab13e97f1286366a80 (origin/keyboard-emulator)
Author: Jake <jtfedd@iastate.edu>
Date: Mon Nov 5 23:02:28 2018 -0600

   Cleaned up midiout_test.cpp