20/09/2019 – 11am

Sophie

Harry

Kelvin

Using a private repo on GitLab to communicate, keep track of HW and SW changes.

Software branching on GitLab:

* Using the Feature Branch Workflow

Hardware Purchases

* 1x IR LED (Existing remote??)
* 1x Photo sensor (only detects IR)
* 1-2x Integrator (op-amp 550)
* Necessary resistors and capacitors for above

GTK Wave – wave viewing

GHDL – compiling VHDL

Hardware Plan: Output waveforms to double integrator to make into a triangle-like wave to ensure better sound quality to speaker.

Problem: Reading the end terminating character when put on the board. Planning to use a RAM module.

Problem:

Digilent Adept Suite – Adept 2 SDK – compatible with VS 2019

SDL 2 C++ or C#/Java???

Infinite scrolling???

Creation/transfer of the music

Possible device monitoring

List of Activities – Hardware Plan:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Activity** | **Description** | **Dependency** | **Time Estimate** | **Expected Time** |
|  | START | NULL |  |  |
|  | HW Planning phase | NULL |  |  |
|  | Design and test of clock division circuit design for quavers/semi-quavers | A |  |  |
|  | Design and test of circuit for outputting of relevant frequency waveforms to double integrator | C |  |  |
|  | Implementation of hardware circuit using components designed in C, and D for basic 60bpm tempo | C, D |  |  |
|  | **Demonstration of D on FPGA** | **C, D, E,** |  |  |
|  | Design and test circuit for tempo changing hardware | F |  |  |
|  | Interfacing with IR LED + remote control to control tempo changes | F |  |  |
|  | Implementation of circuit containing components designed in G, and H. | G, H, |  |  |
|  | **Demonstration of I on FPGA** | **G, H, I,** |  |  |
|  | Integration with SW … TBD |  |  |  |

Software Design Plan:

Draft requirements:

Requirements:

- Create music based on a trebleclef interface

- Translate the given music notes/quavers/semi-quavers to the relevant text file

- User can click desired note duration

- User then clicks the desired note on a visual piano

- Communicate the text file to the board via EPP

- Adjust the speed of the resulting file using a slider

Visual piano:

- Audibly plays note to the user when clicked

- Adds note to score only when relevant "beat" is selected first

- When adding note to score, will play the elected note

Music Score:

- Can add/remove notes

- Can add notes via the piano or right click on the specified location and selecting "add <beat value here>"

- Can add/remove pauses

- After removing a note, can replace it with a newer note otherwise a pause will be placed in when trying to export

- Can extend the number of notes available as going along

General Interface:

- Play entire score

- Play selection of score (later)

- Delete section of score (later)

- Adjust default tempo (later) (60-120 BPM selected by sliding scale)

- Import/edit a song (later)