The Time Stopper

M. L. Carmen
Aalborg University
A. C. Meyers Vænge 15 2450
cmunoz19@student.aau.dk

ABSTRACT

The Time Stopper (TTS) is a NIME which main purpose is to “stop the time”. It is comprised of 4 sub instruments which aim to achieve an ambient atmosphere where the player can relax and calm without much worrying about her/his performance or the instrument complexity. TTS in essence is a tribute to Brian Eno [1] and its contribution to ambient genre music, specifically, an attempt to reproduce similar sounds or sensations caused by song “1/1” from “Music for Airports” album [2].

Author Keywords

NIME, multi-instrument, Teensy, Arduino, Ambient music

1. INTRODUCTION AND MOTIVATION

Many NIME have been developed through modern times. Most of them feature a new or different way to play traditional instruments, others try to resemble a synth, others build new manners to interact and play an unique sound… Most of them are thought to be performed on front of an audience or in combination with some other instruments. However, TTS goal is a total different one.

TTS is a personal instrument, though to create an ambient music modelled and configured by the user in a very simple manner and aided by the machine itself. Its music is uniquely played for the ear’s user, avoiding any “tension” to make a good performance for an audience. Its interface is beautifully crafted with LEDs and lights which will accompany the music during its meters/bars. The user can “see” the music going through time at the light’s rhythm and relax by watching the “music” flowing through time.
TTS is not intended to explore new “horizons” or limits of NIME in terms of manual dexterity or accuracy, but to have a time where time stops.

2. RELATED WORK

2.1 Teensy polyphonic synth

Synthesis of sounds is not a new concept in NIME (of which TTS will make use). There are many synth open source constructions on the web [poliphonicsynth] [4]. One of the 4 instrument of TTS will be build using this synthesis idea applied to Teensy microcontroller too.

2.2 The Monolith

The monolith has forty buttons which serve as an 8-step sequencer with five different voices, while touch sensors on the left and right panels serve as a polyphonic arpeggiator and preset controller, respectively [monolith]. It is also a NIME which not only plays “one” instrument, but several at the same time running on a 3.6 Teensy [6].

NIME’20, July 21-25, 2020, Royal Birmingham Conservatoire, Birmingham City University, Birmingham, United Kingdom.

3. MATERIALS FOR THE PROTOTYPE

For TTS implementation, a Teensy 4.0 is used, an audio adapter board, a USB solar battery, a USB cable, headphones, 7 buttons, 18 LEDs (each one with a corresponding 120 Ω resistor), 3 knobs, 2 switches, 1 photoresistor, 1 fader and lots of cables.
Tools needed: welder, cable cutter/scissors, multimeter, 2 breadboards and a plier.
Expendable material: soldering tin.

4. IMPLEMENTATION/ ITERATIONS OF THE PROTOTYPE

4.1 Song analysis

In Brian Eno’s song, it can be observed that outstanding melody notes come from a grand hall type piano (0:02). Low-pitch or bass notes come from an electric piano/synthesizer (0:01, 0:24). There is also a periodical “clunk” that often is in between or separates the song bars (0:14). Finally, there is also vibrant synthesized strings at some song points (0:40, clearer at 2:02 or 9:43). This last one will not be of our interest.
From this analysis, as mentioned beforehand, TTS will be comprised of 4 sub instruments to resemble the atmosphere “1/1” song has.

4.2 Bell synth

A synthesized bell will be representing the continuous “clunk” the song plays along. Bell synthesis is addressed in many different ways; additive synthesis, FM modulation [7], many oscillators, envelopes and modulated amplification to resemble the different modes and enharmonic a physical bell has [8], etc.
Due to performance and dynamic memory limitations, a simple but effective synthesis was implemented.

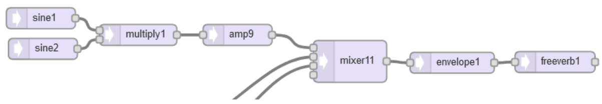

Figure 1. Simplified diagram of one sine treatment in Teensy GUI [teensygui]

4.2.1 Implementation

It is comprised of 9 sines at different frequencies, followed by a multiplication by its detuned versions to create a beating effect [10]. In a real bell, one side of the bell may not be cast to the same thickness as the other side. Consequently, there will be small variations in frequencies which produce beating at various frequencies. For this effect, a multiplication of slightly detuned sines ("enharmonic" partials) are used [11].
Afterwards, amplification effect is used. Amplification follows the spectrum of the natural harmonic series where the amplitude of each harmonic is inversely proportional to its position in the series. Finally, an envelope (ADSR) effect is applied to recreate the natural reproduction of sound followed by a reverberation to “fulfill” the space.

4.2.2 *Logic*

Bell sound is triggered when it is "raining", to be a metaphor of the "clunking" sound effect. This translates to the analogic world into a photoresistor, which when is covered with a piece (it is a cloud) staying in the shadows, will play the bell sound periodically. It is a two-state or ON-OFF instrument. Two LEDs representing a "falling" drop will be sequentially illuminated when bell instrument is ON.

4.3 Bass and Main piano

4.3.1 *Implementation*

For the bass and main piano again, considering memory limitations, wavetable synthesis methods were used. Easy as it could seemed [12], available software for wavetable synthesis (that is essentially raw data samples) were to use only in Teensy 3.2/3.6, not allowing (with more than two or three trials) to have a wide range to select the most similar instrument sounds. Consequently, already provided instruments from Teensy examples were used.

Brian Eno's song was reviewed carefully and the notes for 4 sequences based on the song were written in Ableton and afterwards, written in terms of frequency and duration in Teensy's code for both instruments.

4.3.2 *Bass logic*

To control this instrument, a fader fed with a +3V with 4 possible positions and a knob are used. Knob can have two states, random selection of sequences or the chosen by the fader position. Fader pinout function is analyzed with a multimeter. A LED will be illuminated corresponding to which of the 4 song bass sequences is being played.

4.3.3 *Main piano logic*

For this instrument, 2 switches resembling an airplane control panel are used. Random switch selects if the sequence is been played randomly or linearly. Speed switch selects if the sequence should be played slowly or quickly. The fader is fed with a +3V. A LED will be illuminated corresponding to which of the 4 song melody sequences is being played. Its pinout can be studied using a data sheet or with a multimeter.

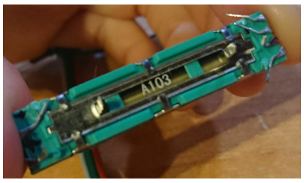

Figure 2. A103 fader

4.4 Solo instrument

4.4.1 *Solo instrument implementation*

The notes of this instrument are straight away played from the SD card using Teensy's audio library function. This is is a straight forward method (not synthesis) to play a sound; solo instrument was desired to have more complex and rich spectrum than other sounds as it is possibly one of most importance to the user as it is the one who will spend more time playing on in comparison to the others. For further enrichment of this instrument, a feedback (echo) delay was applied [13] with two control variables; time delay and feedback. Feedback is controlled varying the gain of the mixers (which is real time efficient) and therefore, how much "feed-back" there is.

As for the delay effect, it is not designed to vary its time delay feature in real time but rather having it predefined from a start for a good performance. Then, a set of amplifiers were applied to the output of the delay effect. Depending on the desired delay time, all but except one were set to 0.

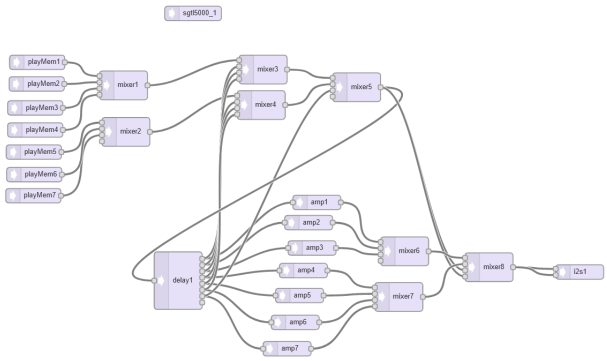

Figure 3. Solo instrument diagram in Teensy GUI

4.4.2 *Solo instrument logic*

This instrument has 7 buttons which will play a piano note on the same key Brian Eno's song has. It has 2 controls; one knob for a feedback delay and another one for setting the time delay. Depending on the played notes, some of the 7 LEDs will be illuminated corresponding to the positions of the played notes.

4.5 Project issues

4.5.1 *Pinout issues*

Teensy 4.0 has limited pinouts: 13 analogic ports and 14 exclusively PWM. 8 of the 13 aforementioned analogic ports have a PWM mode. Ports 24-33 can be accessed by soldering (no direct plug). Ports 34-39 (which are not counted for the previous ports count) can be soldered by SMD soldering (for its tiny little size), and therefore, discarded for this project, as they can result in a hazard for a decent performance.

Moreover, some audio adapter ports [14] are not possible to use from Teensy if a SD card is being used or clock signal is needed, reducing the available ports to use, so final prototype will not have as many LEDs as in the beginning.

Soldered ports offer less accuracy and performance and thus they are designated to low failure risk operations in TTS; switches and LEDs.

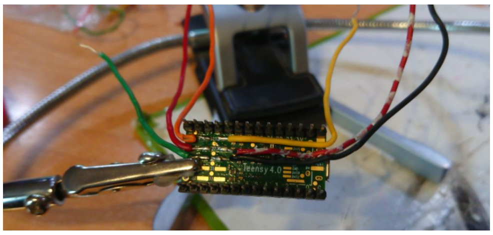

Figure 4. Soldering cables on Teensy ports

See pinout.xlsx [15] for full pinout of Teensy ports of this project.

All these issues can be handled without problem if in possession of a Teensy 4.1.

4.5.2 *Memory issues*

For playing in real time 4 "tracks" at once in a microcontroller, a consideration for dynamic memory and RAM performance should be essential. The dynamic and memory usage by the 4 instruments individually are:

Table 1. Memory and storage usage in Teensy

	Bass and Main piano	Solo instrument (using SD card)	Bell synth
Dyn. Memory	38%	11%	14%
Program storage space	4%	2%	1%

There are only 2 ways of reading audio in Teensy; from flash memory or SD card, and this audio format can be either .wav or raw audio.
If instead of using SD card reading for the solo instrument, flash memory was elected, performance is quite better (latency, time response…), however, dynamic memory increases up to 86%; a percentage that will not allow Teensy to work correctly with other instruments addition.
Also, reading raw audio from SD card instead of .wav, which only removes its header, does not increase significantly usage.
High quality reverb used also increase pretty dynamic memory usage (10% by each one).

4.6 Battery supply

A portable solar USB battery of 9000 mAh is supplying teensy the current and voltage to work. This is located in the box enclosure.

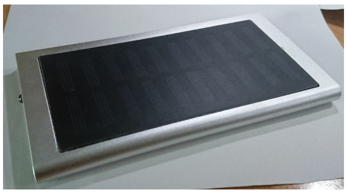

Figure 5. Portable battery used for the project

4.7 Audio output

Audio will play through the audio minijack located in the audio adapter board, which will be inside the box enclosure. Headphones can be and are encouraged to listen to TTS, as performance is personal and individual.

4.8 Sketch design, box enclosure and prototype

Fusion360, a ruler (to measure actual components size) and a pencil where used to create the box enclosure model and the painting designs.

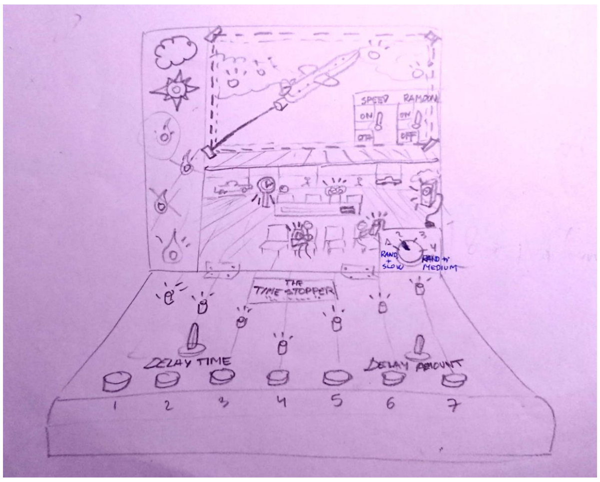

Figure 6. Pencil draft for the TTS

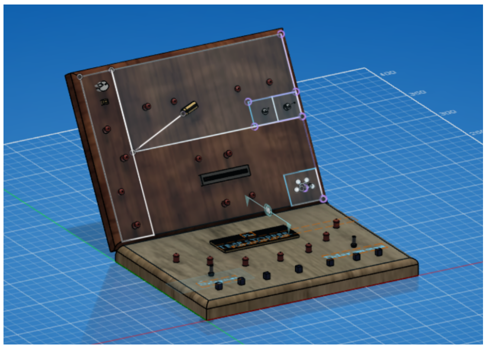

Figure 7. TTS box enclosure on Fusion360

Box enclosure is thought to bend to occupy less space. It would be made out of cardboard and be painted in the inside to recreate the song metaphors.

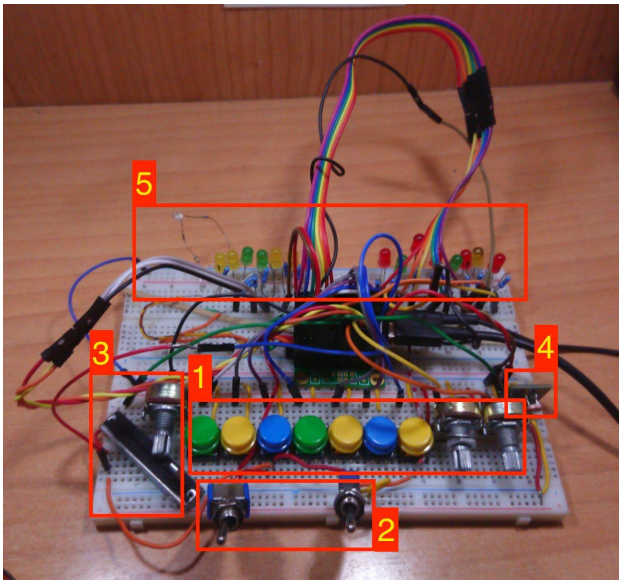

Figure 8. TTS prototype build on to 2 breadboards. Rectangles: 1 Solo instrument – 2 Main piano – 3 Bass – 4 Bell – 5 LEDs

5. EVALUATION PROTOCOL

An evaluation protocol was designed for a complete successful implementation of TTS as considered of importance for NIME community [16]. Brief notes are considered below:

5.1.1 Goals of the evaluation

Getting to know if the Time Stopper is regarded as something appealing in the design and experience, once the true purpose of it. If TTS is a "kind" of instrument which helps to create an atmosphere which is prone to create tranquility, relaxation, calmness and does not lead the user to "exploit" fast, difficult or difficult performance. It is an instrument, crafted and designed to stop the time (like Brian Eno "1/1" song).

5.1.2 Which are not goals of the evaluation

Measure infinite possibilities, latency, battery life, quality of sound, speed.

5.1.3 Evaluation protocol

1. Observations from user and telling him/her the instructions when approaching the instrument, open comments, interview.
2. Questionnaire: SUS [17] and Adjective Rating Scale [18]

1 – *We let the user see the instrument but not to play. One open question will be made:* What do you like this thing? Comment on anything that your mind is telling you. *Further general questions could be considered if the user is "hard" to tell anything.*
Afterwards, we tell the user the brief purpose of the Time Stopper (it's a personal device to stop the time, to slowly, reach tranquility). We let the user play.
We write down any unexpected behavior in Observations.
Afterwards, interview is carried out:

- Is it joyful to play? Is it easy to understand what it makes? Do you like the sounds it produces? Is the physical design appealing to you? What difficulties have you encountered? Is it fun? Which sound do you like the most? Do you like the distribution of the controls?
- Did it evoke any visual images? Have you relaxed in some sense? If you'd need to relax for some moment, would you use it? Would it work for reaching calmness? Is this sound similar to Brian Eno "1/1" song? (this should be played if not known)

2 – SUS is given in a sheet of paper with the proposed modifications of *Determining what individual SUS score mean.* Adjective rating scale is given in another sheet of paper with the following statements:

1. Overall, I would rate the user-friendliness of this instrument as:
2. I would rate the design of the instrument as:
3. I would rate "beauty" of the appearance of this instrument as:
4. I would rate the produced combination of sounds as:
5. I would rate the degree of "calmness" achieves by the device as:

6. UNIMPLEMENTED IDEAS/ MISHAPS

Due to the limited time of this project, the following milestones were not able to be achieved:

- A fully functional model within the box enclosure
- Due to the lack of operative LEDs, not all could be shown to work in real time, although the considered soldered design is prepared to deal with them all
- Due to lack of PWM, laser beam (shoot from the back of the airplane paint) bouncing on the little mirror pieces is not implemented
- Due to the many connections/low quality components, there is a continuous white background noise

7. ETHICAL STANDARDS

Sources of funding: thanks to Aalborg University in Copenhagen for lending some of the materials used for this project. The construction of this project involved the author of this paper and Dan Overholt as supervisor.

8. REFERENCES

[1] Tamm, E. (1995). Brian Eno: His music and the vertical color of sound (pp. 141-146). New York: Da Capo Press
[2] Ambient 1: Music for airports, R.P. (22 May 2020). In Wikipedia. Retrieved from https://en.wikipedia.org/wiki/Ambient_1:_Music_for_Airports
[3] Otem. R.P. (2017,May 13). Teensypolysynth. Retrieved from https://github.com/otem/teensypolysynth/
[4] Dave, R.P. (June 10 2017). TEENSY-Synth PART1. Retrieved from https://www.notesandvolts.com/2018/05/teensy-synth-part1.html
[5] Darcy Neal. Current Projects. Retrieved from http://www.ladybrainstudios.com/works/universal-death-sound-light-cube-no-1-aka-the-waynebrain/
[6] Baichtal, J. R.P. (June 10, 2017). THE MONOLITH BRINGS THE BOOM TO MAKER FAIRE. Retrieved from https://hackaday.com/2017/06/10/the-monolith-brings-the-boom-to-maker-faire/
[7] Serafin, S. Assignment 2: Sound synthesis competition. Retrieved from: https://ccrma.stanford.edu/~serafin/320/assign2/assign2.pdf
[8] Reid, G., R.P. (August 2002). Retrieved from https://www.soundonsound.com/techniques/synthesizing-bells
[9] Stoffregen, P. Teensy Audio Library. Retrieved from https://www.pjrc.com/teensy/td_libs_Audio.html
[10] Blair Vanderbilt university. Creating a Simple Bell Using Additive Synthesis Techniques. Retrieved from https://www.computermusicresource.com/Simple.bell.tutorial.html
[11] Blair Vanderbilt university. BEATING. Retrieved from https://www.computermusicresource.com/Definitions/beating.html
[12] R.P. (Fall 2016). Teensy Wavetable Synthesis. Retrieved from https://teensyaudio.github.io/Wavetable-Synthesis/html/index.html
[13] Stoffregen, P. R.P. (14 November 2015). Microcontroller Audio Workshop & HaD Supercon 2015. Retrieved from https://hackaday.io/project/8292-microcontroller-audio-workshop-had-supercon-2015
[14] Stoffregen, P. Retrieved from https://www.pjrc.com/store/teensy3_audio.html
[15] Muñoz, C. R.P. (2020, May).Retrieved from https://github.com/chachipirulin/theTimeStopper
[16] Barbosa, J., Malloch, J., Wanderley, M. M., & Huot, S. (2015, May). What does" Evaluation" mean for the NIME community?
[17] Brooke, J. (1996). SUS-A quick and dirty usability scale. Usability evaluation in industry, 189(194), 4-7.
[18] Bangor, A., Kortum, P., & Miller, J. (2009). Determining what individual SUS scores mean: Adding an adjective rating scale. Journal of usability studies, 4(3), 114-123.

9. APPENDIX

See GitHub repository for TTS project files [15].

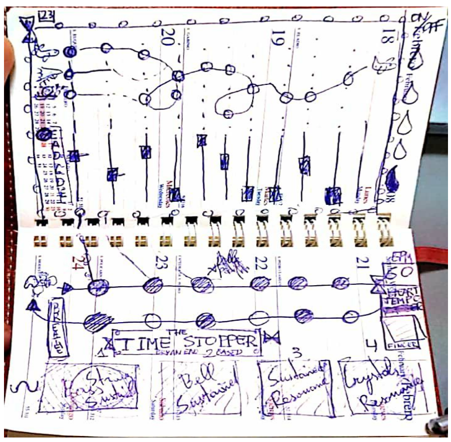

Figure 9. Discarded first idea sketch of T