THEN – Final Project

Galvanic Skin Response Gamified

Ariane Hipsagh – 22689812

CART 360 – Tangible Media and Physical Computing

GitHub at

https://github.com/Ahipsagh/CART360/tree/master/Assignments/THEN/FinalProject

Professor Elio Bidinost

December 7, 2018

From Proposal to Mid Fidelity Prototype

After our proposal meeting about the THEN proposal which was the creation of Galvanic Skin Response (GSR) fingerless mittens, we brainstormed a two-player gamified proposition using the GSR technology.

Phase 1 – Revise proposal to a Game

Correlate GSR and Pulse Monitor data to confirm a specific state or to indicate another emotional state that requires both collections of data.

The revised version would use the GSR sensors that presented in the proposal. Conductive hook and loop would replace the aluminum and regular hook and loop. The audio cable wire is sewn with conductive tape directly to the conductive hook and loop to improve the electrical signals communicated from the two fingers into an analog port on the Arduino. Each player wears the sensor on two fingers of one hand. The signals from the respective sensors are amplified through the breadboard setup based on the MakeProjects.com Truth Meter project including schematic into respective analog ports on the Arduino. The Truth Meter project uses hardware that is supposed to use the Arduino to simulate the polygraph technology which was the brainchild of Cleve Backster (1924 – 2013). According to the video on for the Truth Meter Project, the LED on the breadboard will turn on when the person lies. The participant must be motionless while wearing the sensor. Each player each inserts their hand into an object (which was to be determined at this time) with the goal of invoking a strong emotional response that could be measured by the sensors. The object would be similar in concept to a Twiddle Muff which is a mitten that has several textured items on the inside that has been effective therapy for Alzheimer victims. In the context of the game, the inside of the object would contain unknown items that would provoke an emotional response from the player. The new goal for the GSR sensors was to have two players synchronize one of three emotional responses which is visualized by a Neopixel light strip and at the the same time could be discerned by the sensors using Arduino Stress Test programs. When the players are calm the Neopixel strip would be blue, when happy, the lights would be green and when angry or scared, it would flash red. When the players synched to a colour the Arduino program would invoke the servo motor to open the 3D printed flower to signify the end of the game.

Phase 2 – shopping

Where there is smoke…

This project required several components to be purchased. Specifically, the conductive hook and loop tape and audio cables was bought from a Montreal electronics store. The op-amp is available online. While waiting for the 8-pin op-amp to arrive, I was able to acquire a 14-pin op-amp from the IMCA Concordia department. The “hurry up and wait” project needed several iterations on the bread board including two different setups for the op-amps.

Phase 3 – remove object from the game

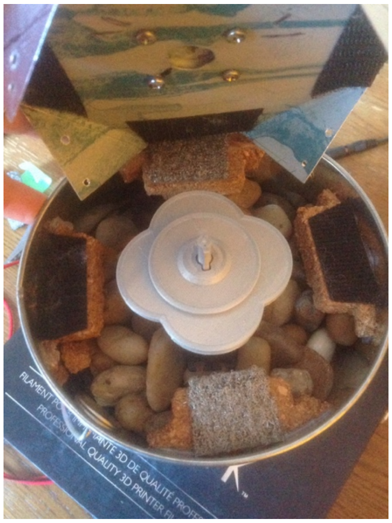
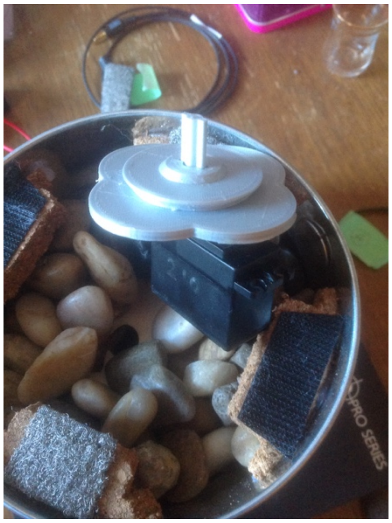
On our next meeting, we tested the breadboard with both the 8-pin op-amp and the 14-pin op-amp configurations. Ultimately, there is no evidence that the GSR works nor that there is any correlation between strong emotional responses and the analog readings that were captured by the Arduino analog port. The good news is that the readings were able to be translated into three distinct value ranges that could be shown by the Neopixel strip in the blue, green, and red lights.

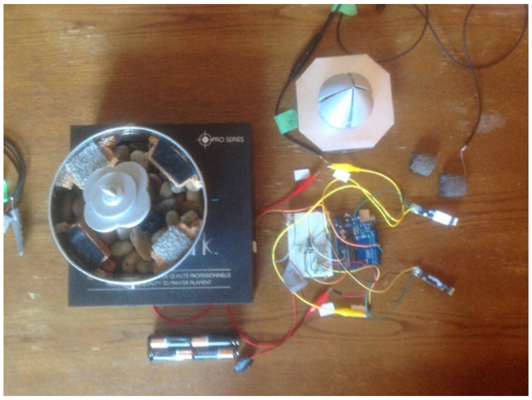
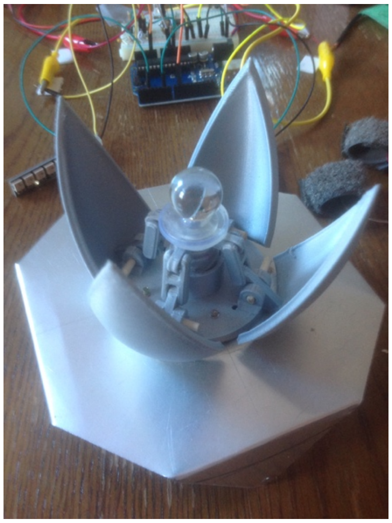
As a result, there is no longer was a requirement for the object because there is no way to provoke an emotional response from the player. The next step is to build the other breadboard in an identical setup to the first and have the players synch to a common color and have the servo motor open the 3D flower to end the game.

Phase 4 – where there is smoke …

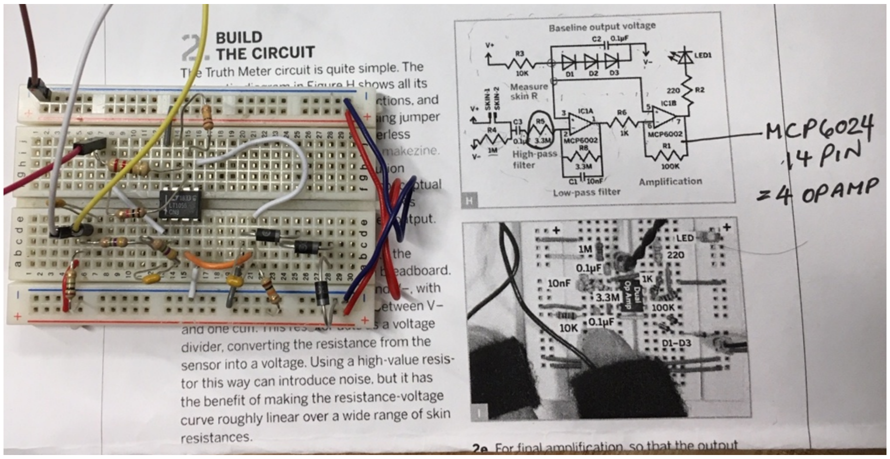
Unfortunately, when the other breadboard was replaced in the original configuration, the readings did not match in any way to the original set up. In fact the readings were completely different and was not able to discern three colour ranges required for the game. Thankfully, when the original breadboard was put back into place, it worked properly. As a result, I decided to create a two-player game, where the other player is the Arduino for the mid-fidelity prototype presentation.

THEN GSR Game

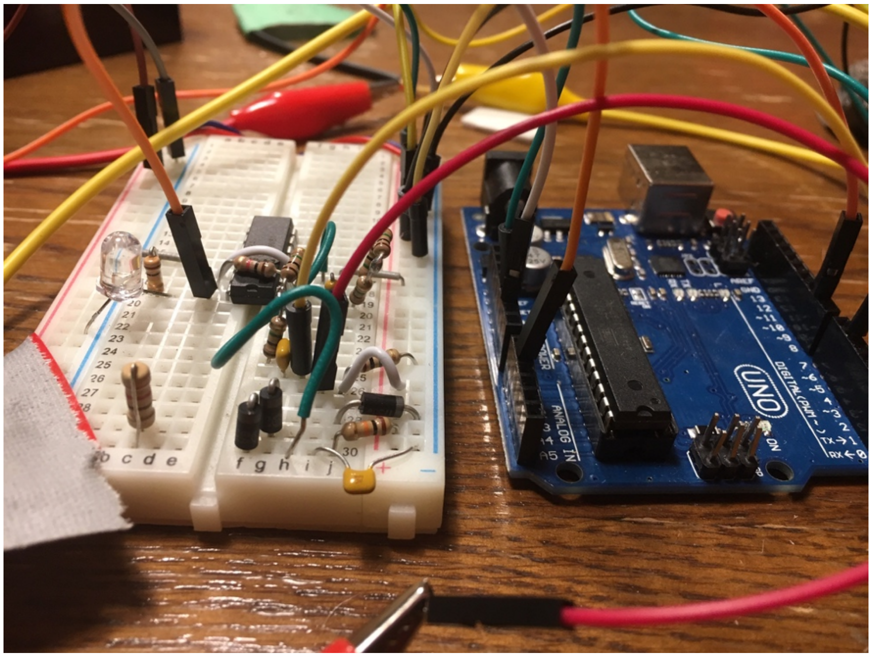
3D flower	3D flower motor
	
Lots of hook and loop tape holds this project in place. Here is the under the hood look for the 3D printed flower installation in the flower pot.	Servo motor attached to the 3D printed flower. The rocks hold the motor in place.

THEN GSR Game components	3D open flower
	
Components include 3D flower motorized by servo motor, four AA batteries, breadboard, Arduino, audio cable sensor	3D flower in opened state includes a transparent marble centerpiece

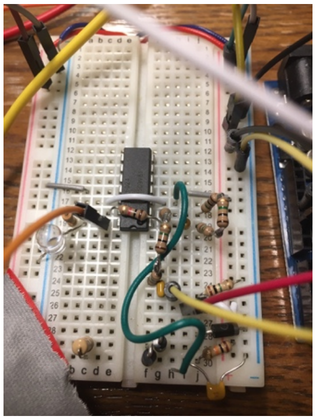
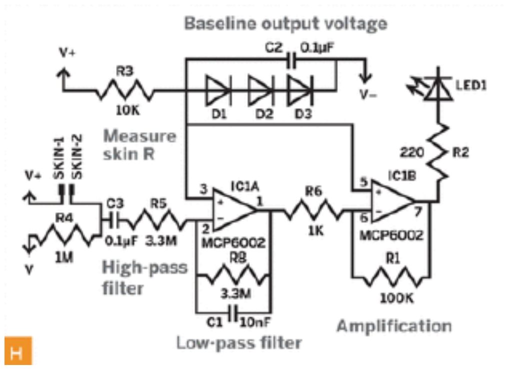
8-pin op-amp breadboard configuration

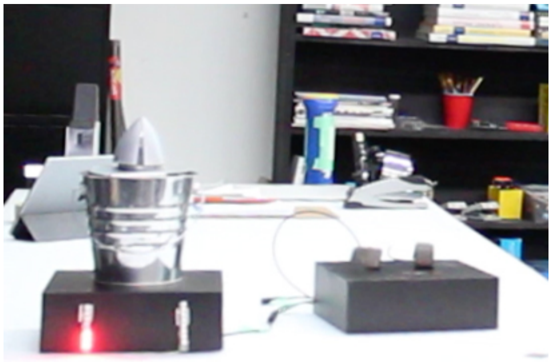
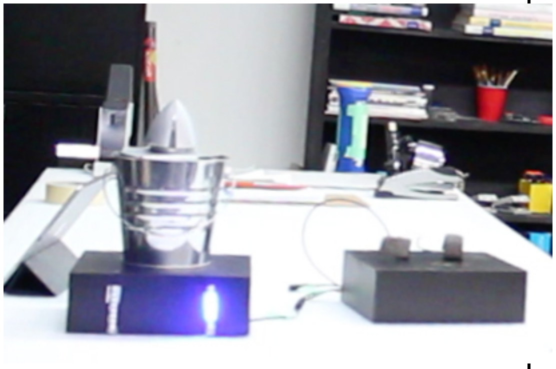


14-pin op-amp configuration used in GSR Game



Setup for 14-pin op-amp breadboard configuration to Arduino analog port.

14-pin op-amp configuration	Truth Meter schematic
	 Baseline output voltage C2 0.1µF V+ R3 10K D1 D2 D3 V– LED1 Measure skin R SKIN-1 SKIN-2 220 R2 IC1B V+ R4 C3 R5 3 IC1A 1 R6 5 7 6 2 0.1µF 3.3M 1K MCP6002 MCP6002 1M V R8 R1 High-pass filter 3.3M 100K C1 10nF Amplification Low-pass filter H
Close-up view of 14-pin op-amp breadboard configuration	Schematic of Truth Meter from MAKEVol26TruthMeter.pdf used for THEN GSR Game.

Game begins	The player state should be off
	
Red NeoPixel strip is the Arduino state	Blue Neopixel strip is the player state

Green player state	Game over – flower blooms
Green player state indicated on player Neopixel strip.	Game ends when the sensor matches the Arduino colour

Post Mortem

It was interesting to notice that when I presented the mid-fidelity prototype to the class, the player colour was flashing between blue and green constantly. Throughout the testing phase, when the sensor's were not in use, the player colour was off. After the presentation, when the class moved away for the game, the player light turned off.

The game is its current state can be changed in several ways. For example, one player can see the Arduino's colour and not his own. The next version of the Arduino program can have the colour randomized.

Research Sources

Galvanic_Skin_Response___The_Complete_Pocket_Guide.pdf (Github)

MAKEVol26TruthMeter.pdf (Github)

10556fd.pdf (Github) – 8-pin op-amp datasheet

21685B.pdf (Github) – 14-pin op-amp datasheet

https://wiki.pumpingstationone.org/Biosensor_Array_Galvanic_Skin_Response

https://forum.arduino.cc/index.php?topic=455550.0

http://thomaskosch.com/index.php/2017/12/17/galvanic-skin-response-powered-by-arduino/

https://www.media.mit.edu/galvactivator/faq.html

https://www.instructables.com/id/Making-Galvanic-Skin-Response-Finger-Electrodes/

https://youtu.be/IjVQpwVHpOo

https://www.instructables.com/id/Pulse-Sensor-With-Arduino-Tutorial/