Visualize the Invisible

Beverly Yee
College of Engineering
University of Utah
bevyujw@gmail.com

Alex Gray
College of Engineering
University of Utah
agray0659@gmail.com

***Abstract*—The modern world offers a variety of high-tech devices on which we can listen to music. Most of those devices, however, cannot provide visual stimulation to go alongside the audio stimulus. The first solution to this was a visualizer that reacted to the music's intensity/volume followed by one that is simply a single column of lights. Neither of these solutions was stimulating enough, leading to the development of a new solution–the music frequency visualizer. In this method, small sections of frequencies are analyzed to create multiple columns of visualizations.**

This project will create an indoor water show like that at the Bellagio Hotel to enhance the user's music-listening experience. But instead of pre-set patterns scheduled to run at certain times, like the show at the hotel, real-time calculations will be performed to generate a unique visual response based on an analysis of the music being played. This report will describe the materials and knowledge required to realize the concept of a wired or wireless indoor music-reactive water system.

I. Introduction & Motivation

In the beginning, music visualizers began as sheet music as a method of helping musicians perform [5]. Now, in the 21st century, music visualizers have become synonymous with software that "captures data from a music audio file" and creates color, shapes, and images based on that data [10]. In a way, they allow listeners to "see" the songs they're listening to, which is especially important for those who are deaf or hard of hearing [4]. Since they cannot or have difficulties hearing, it takes away the experience of music. Through visualization, they could gain an experience similar to hearing audiences.

The first visualizers–electronic visualizers–were created by Robert Brown in 1976 [8], the same person who created the game, Pong. At first, these electronic music visualizers were primarily used in video games. As the years passed, the visualizers transitioned from video games to personal computers. An older example of one such version would be the Windows Media Player 7, rolled out in the mid-2000s-[7].

Today, visualizers appear on the internet as freeware–that is, free software or open source–for the public to make as their own and exist in a vast variety. On the computer side, this software is mainly used to create music videos for platforms such as YouTube, Facebook, or Twitter. There are also hardware-based music visualizers–DJs use them in clubs to create lighting effects on the dance floor.

Commercially, there are music speakers sold that have attachments to give a similar effect, such as lava lamps and water pumps. The latter partially inspired the creation this project, as it also uses water pumps and LED lights. The downside of these speakers is that there are only 4 pumps in each speaker and the height of the water is fixed, dictated by the total intensity of the music playing at that moment.

A. *Motivation*

The goal of this project is to recreate the water pump speaker with some added complexity. 12 pumps will be used instead of 4 and will operate independently of one another. A strip of LEDs will encircle the pumps to continue the theme of using light to enhance the water columns. The pumps and the LEDs will be separate from the speakers, which will be attached to the sides, as the "water show" is the main focus.

The operation of the pumps and LEDs are entirely dependent on the audio given to the microcontroller overseeing them. The idea is to control and vary the height of each pump to reflect that audio's frequency at a certain time. The LEDs are to pulse through the audio, preferably at key moments of the audio, such as the switch of a beat in a song.

To put it simply, the mission is this: The Bellagio's mini-me: a dance of water fountains to the beat of any chosen musical piece and is highlighted by a spectrum of LEDs. The height of the water and the intensity or color of the LEDs is determined by the frequency/pitch of the music. Visualize the Invisible.

II. Primary Project Tasks

The following section will go over the primary tasks needed to complete the project.

*1) **Select a Microcontroller**:* Currently, there is not an exact microcontroller decided for the project. That said, the STM32 Discovery Board (STM32f072RBx) was used in the demonstration of the prototype and verification that the pumps work as expected. It has 12 pins capable of PWM generation, 4 sort of the desired 16. Additionally, there doesn't seem to be many readily available online resources for the board should there be issues.

The Arduino UNO was briefly considered as an alternative, but it only has half the pins capable of PWM the Discovery Board has. For the time being, the Discovery Board will be used. The 12 pins will not be detrimental in the operation or performance of the water show–it would only decrease the quality of the sampling.

*2) **Select Method of Input**:* The input required for the project to work is an audio file of some sort, preferably music. The method of receiving that input will be through the STX-3120-3B stereo AUX port (shown in 1), which takes the standard 3.5mm headphone jack and will be connected to the microcontroller.

Fig. 1. STX-3120-3B Stereo AUX Port

Reading the signal depends on the microcontroller used. However, in terms of the Discovery Board, there are two options that can be used: an Analog to Digital Converter or the SPI interface in I^2S mode (Inter-IC Sound). The latter is a serial bus interface specifically designed for audio.

*3) **Fast Fourier Transform Analysis**:* Fast Fourier Transform (henceforth referred to as FFT) is an algorithm for analyzing signals to convert them into sections that describes the frequencies present. The frequency resolution of each spectral line is equal to the Sampling Rate divided by the number of points in the FFT. Therefore, a higher spectral resolution can be achieved by increasing the number of points to sample [2]. The downside to this, however, is that the more points given to the FFT, the more time it will take to calculate.

FFT works by taking a signal–in this case, the music piece–and sampling it over a period of time. That sample is then divided into its frequency components. Each of these components is a discrete, sinusoidal oscillation with a unique frequency, amplitude, and phase [3]. Fig. 2 represents the frequency graph of a sample signal in which 3 distinct frequencies were found.

Whenever there is a spike in the frequency of the original signal, that spike is reflected as the amplitude of the individual frequencies. That amplitude can then be converted into a usable, numerical value to represent the height of each bar of the visualizer; in this case, the height of the water column.

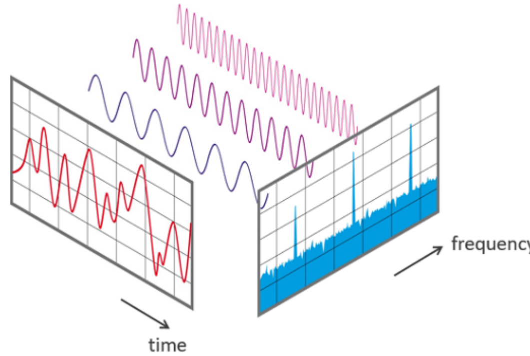


Fig. 2. A signal broken up into the 3 distinct frequencies that it is made of [3].

There are multiple versions of the FFT algorithm. The one used for this project is called the Cooley-Tukey FFT algorithm. Compared to the others, this algorithm has an improved Big O time complexity (Eq. 1) than the standard discrete Fourier Transform algorithm (Eq. 2) [6].

$$O(N * logN) \tag{1}$$

$$O(N^2) \tag{2}$$

An improved Big O time complexity is important for calculations as it allows close to real-time computation of signals such that there is not a lag between the audio and the visualization of said audio.

The method for computing the FFT recursively re-expresses the discrete Fourier transform (DFT) as a number of smaller DFTs recursively. The algorithm formula for DFTs can be seen below in (Eq. 3).

$$x_j = \sum_{n=0}^{N-1} x_n \exp\left(-\frac{2\pi i}{N} * nj\right) \quad j = 0, \cdots, N-1 \tag{3}$$

*4) **Create a Pump Control**:* The pumps used are pumps sold for small aquarium tanks and run between 3 to 5 volts. Controlling the voltage will in turn control the intensity/height of the water, which will be used to represent the intensity of each frequency. To do so, a PCB is created to be used a driver for the pumps (Fig. 3). It will connect to all of the pumps to the microcontroller, where it will receive and use the numerical data given by the FFT.

Since the Discovery Board only has 12 channels of PWM, the schematic was designed for 12 pumps only, but can be increased or decreased as necessary. Each pump utilizes an N-channel MOSFET, a flyback diode (to protect the MOSFET from a spike in voltage due to the pumps being turned off), and a 1k Ohm resister (to serve as a discharge path for the gate pin of the MOSFET). Each pump will need a 5V power supply in order to be able to run at its full potential.

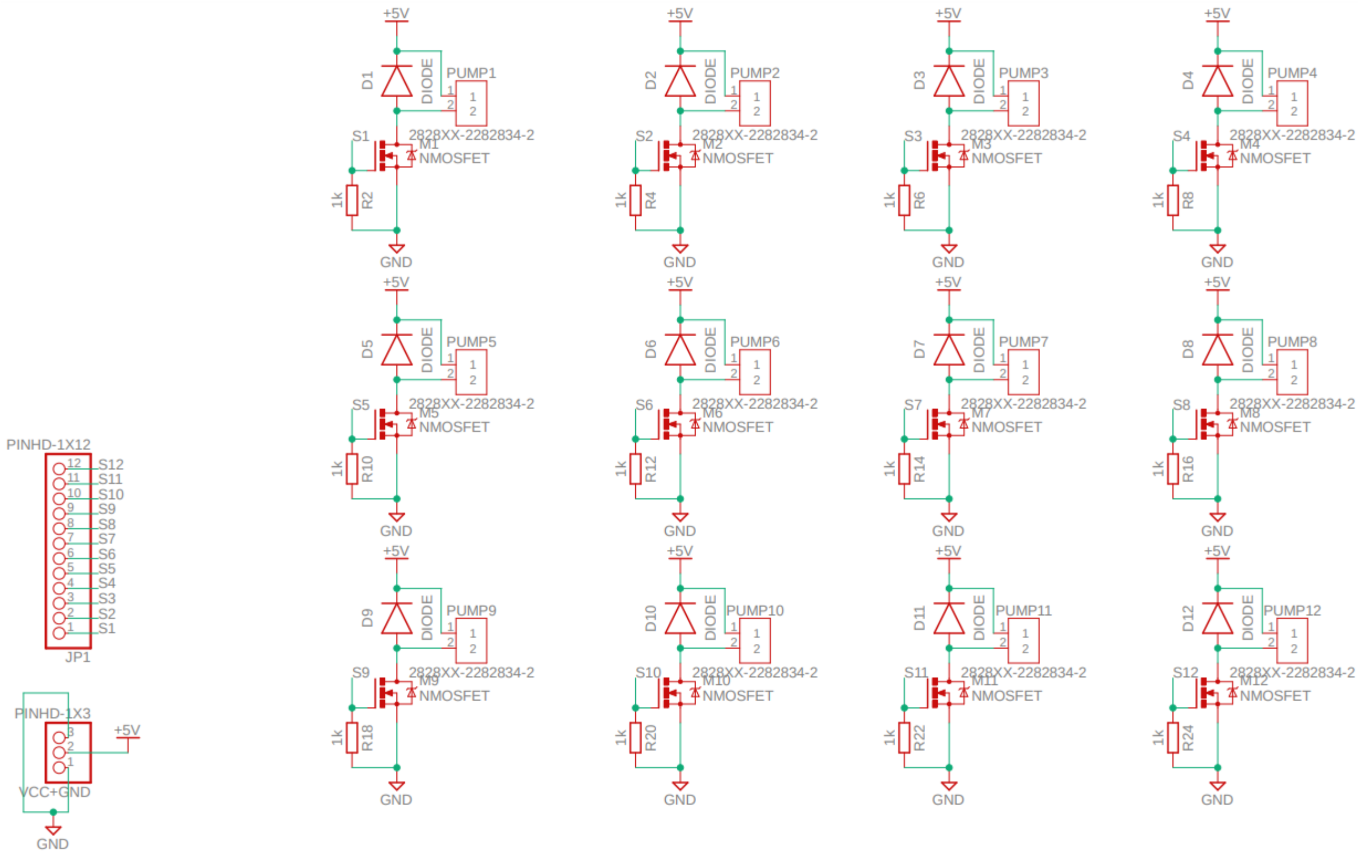


Fig. 3. Pump Driver Schematic

The final PCB design for the 12-pump driver can be seen in Fig. 4.

*5) **Acquire an External Power Source**:* In order to power all 12 pumps to the maximum flow rate, an alternate power source is needed as the 5V power pin of the Discovery Board would not be enough. This specific pin only provides around 100mA of current and each individual pump has a load-rated current 0.18A. Multiplying that by 12 results a total current draw of 2.16A with all pumps at their maximum. In other words, the power source necessary for operation is one that will provide 5V with a high enough current draw threshold.

Initially, the MB102-PS 3.3V/5V power supply board was considered, but the board only had a max output current of 700mA. Then, an AC to DC power supply adaptor was found on Amazon that took input power from a wall outlet and converted it to a 5V/5A supply. Not only does it have a screw terminal for wires, it also has more than enough current to power the pumps.

*6) **Determine Lighting for Enhancements**:* Depending on the LEDs found, there are a few ways the lights can be controlled and displayed.

1) Use individual LED lights, such as the ones found in the Arduino or Discovery starter kits.
    - The advantages to these are the lower costs for the materials and the freedom to determine where the LEDs go.
    - The disadvantages are:
        - The LEDs, whether they are surface-mount diode (SMD) LEDs or through-hole LEDs, will need a custom printed circuit board (PCB) to control.
        - There is the possibility that more calculations are needed to ensure the right voltage/current is delivered to each LED.
        - There needs to be a way to waterproof the LEDs in the event water splashes up to their location.
        - Finally, more work is required to get the hardware pieced together, which may create a product that does not look as nice as manufactured products.
2) Use pre-made LED ring lights that can fit perfectly around the tubing of the pumps.
    - The biggest advantage of these lights is the appeal. Since there is a set of LEDs surrounding each outlet of the water pump, the columns of water would have more emphasis on them as they shoot out.
    - The disadvantage is the cost. The cheapest option

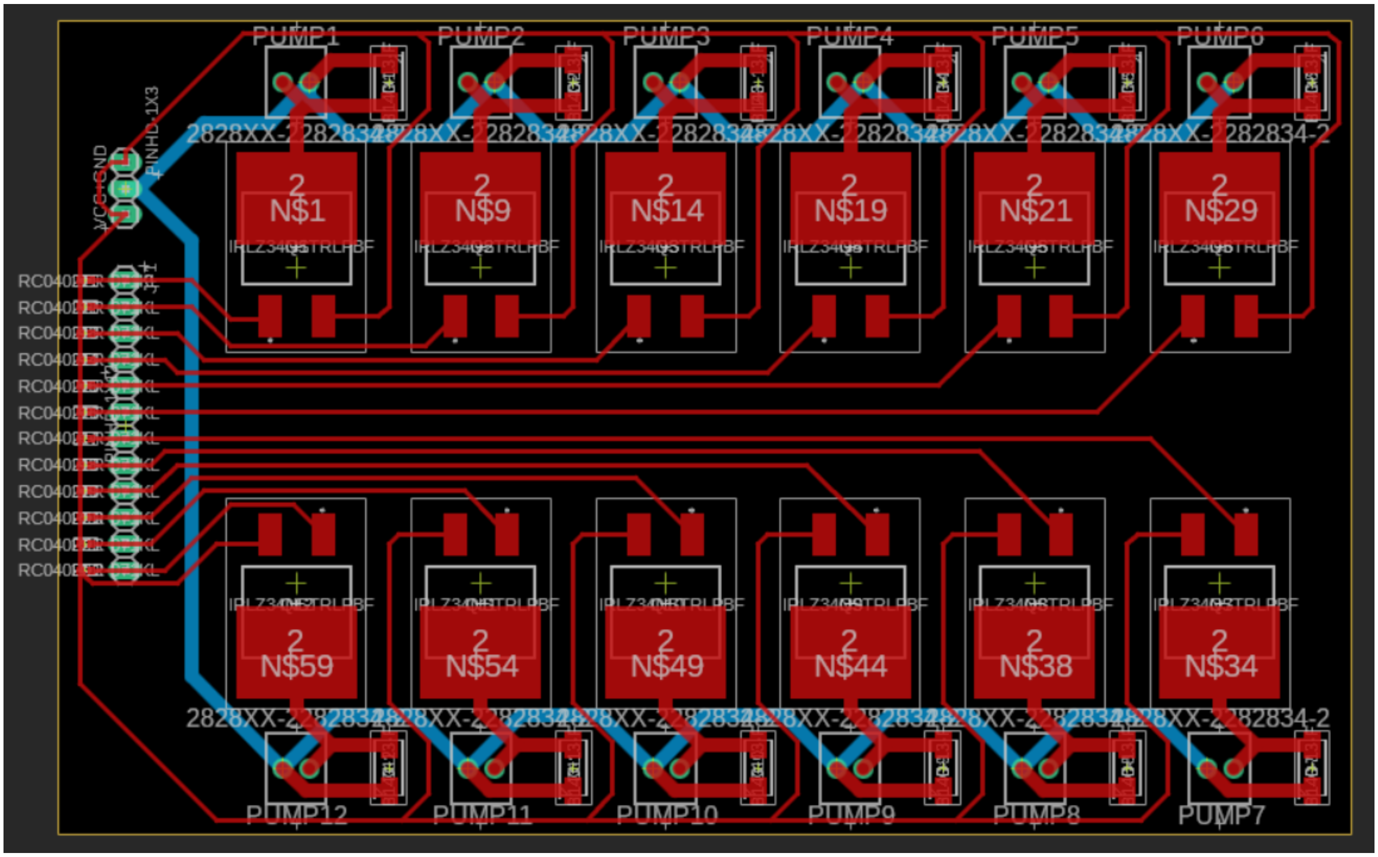


Fig. 4. Pump Driver PCB

found on Amazon were roughly $15 each. Collecting enough for 8-16 pumps would result in one or two hundred dollars, just for lighting alone.

3) Commission an LED specialty company to create customized LED rings that are both sized to the opening of the tubes and waterproof.
   - Similar to the option mentioned previously, the LEDs here will provide more emphasis on the water columns as they appear. Additionally, since they are customized, the types and number of LEDs can be changed to fit the specifications of the project.
   - The disadvantage is, again, the cost. Contracting a company to create a custom PCB is more expensive than simply buying a set on Amazon. Additionally, the wiring involved to connect each ring could detract from the final appearance.
4) Use LED light strips that will be adhered along the inside of the container.
   - This is by far the easiest option as only 1 LED PCB driver is needed to control the entire strip. In theory, it should still be able to provide plenty of emphasis on the water.
   - The disadvantage would be positioning the LED strip at the wrong angle such that the LEDs are only there to look pretty. Additionally, the strip needs to be already waterproofed for possible underwater use or kept out of any possible contact with water.

As for controlling the LEDs for the options that don't have a driver built in, the TLC5947 LED driver from Adafruit is heavily considered. It has 24 channels of 12-bit PWM, which is more than enough to control separate, multiple LED circuits, should that route be chosen.

A good alternative is the TLC5940 breakout from SparkFun. It only has 16 channels, but they are all 12-bit PWM and more than enough for the 12-pump schematic. Both can be daisy-chained if necessary and used as drivers for servos, which could be used for the pumps. The TLC5947 is noted to not good to be used for servos while the TLC5940 is [1] [9].

*7) **Obtain Speakers for Audio Output**:* While it would be nice to see what the song looks like as a water show, it is not complete without audio. If anything, users would be staring at a bunch of water columns of varying height and color without a clue as to what the water is dancing to. The pair of speakers to be used will be generic speakers with a 3.5mm jack output loaned by the professor.

*8) **Final Assembly**:* The final product, roughly sketched in Fig. 9, will be an acrylic encapsulation filled partially with water. It is composed of 5 areas: tank, stage, service box, speakers, and input.

The details of the final sketch will be covered in Section VII Final Sketch Details, but as a brief overview:

- The tank contains the water, pumps, tubing, and LEDs. It's also where the water is dumped back into after being shot out from the tubing.
- As the name suggests, the stage is where the water show happens. It sits above the tank and is surrounded by a clear wall to keep the water from splashing outwards.
- All of the logic of the project is contained in the service box, attached either to the side or the bottom of the tank.
- Speakers are to be placed on either side of the tank.
- Input will be taken from the back of the tank, nearby the service box. It is an AUX port that is connected to a music-playing device via an AUX cord.

III. Stretch Goals

There are a couple of ideas the collaborators would like to see added to the project, should time permit following the completion of the main goals. Each stretch goal is listed below, beginning at the highest priority.

*1) **Beat Sync**:* This particular goal is the most highly anticipated one as the desired output will step up the visualization of the input music. While the exact point is still in discussion, each time there is a drop in beat or the bass is particularly strong, both the LED and water streams will respond with increased intensity equivalent to that beat.

The method of recognizing the beat for this goal will be a little more complicated. In addition to computing the FFT of the signal, an analysis of the amplitude data is needed to find the moment when one amplitude (on the lower end of the spectrum) is significantly louder than the neighboring amplitudes and/or a set threshold. This is because the beat is usually the loudest part of the entire spectrum when it hits.

The drawback lies in the accuracy of the detection at the cost of the speed of the calculation. As such, this goal remains a stretch goal as of the moment such that the processing speed of the signal remains in real-time.

*2) **User Interface**:* For this goal, the idea is to set up an LCD screen with buttons so a user can adjust the visualizer to their preferences, such as changing the song or switching up the settings of the LEDs to different options (eg. rainbow cycle, solid color, beat sync, etc). Having a user interface is not crucially necessary for the project to work, so it is considered a stretch goal.

*3) **Bluetooth**:* As the title suggests, instead of using an audio jack to receive audio, attach a module to the microcontroller so the visualizer can connect to Bluetooth-enabled devices, such as a smartphone. With little experience in dealing with Bluetooth, however, this stretch goal will be the last one to be considered.

IV. Testing

The following sections break down how testing will be done for each part of the project.

A. *Pumps*

During the early stages of the project, each pump received must be tested individually to verify that is not a defective product. Once verified, the pumps will need to be tested at differing voltages to see how it operates and to give the range of values that can be used with the values given by the FFT. Ideally, each stream of water is to land/meet at the very center of the fountain when their pump is operating at max intensity.

B. *Acrylic Casing*

There is not much to test for on the enclosure. One of the requirements is for it to be able to be cut by one of the machines provided. Additionally, as it is difficult to find acrylic cylinders tall enough for this project's purposes, several shallow trays were bought. The bottoms of one or two of these trays may need to be cut out and glued onto another to create a taller enclosure. For this case, a water test is needed to ensure there is no leakage from the seams.

Water tests will need to be performed periodically throughout. The most important thing is for water not to get to the service box else the destruction of all electrical components are imminent.

C. *FFT*

The FFT algorithm will be a major component of the software portion of this project so we will thoroughly test this as we progress. We can start with basic software testing code to see what values return based on input. We can then write code to create a basic visualizer on our computers and use a song from a YouTube video that features a visualizer to compare. Once we know our algorithm works, we will begin implementing the code into the microcontroller to affect the pumps.

D. *Pump PCB*

Once the PCB is received and the parts soldered on, the first task is to test the circuit with a voltmeter and verify that all parts are performing as intended. The pumps and microcontroller will be attached to the PCB only when the test passed such that damage to the other components is mitigated.

Since the PCB is being manufactured by OshPark, there will be a total of 3 boards in one order/shipment. This gives some leeway as 2 more PCBs are readily available should one end up being faulty or is somehow fried during the building process. The final PCB size is 4.6” x 3”, roughly the size of an index card.

E. *LED PCB*

In the event an LED PCB is created, the testing procedure will be very similar to the pump PCB mentioned above. The LED PCB will be tested as soon as it is received and assembled. To avoid damage to the LEDs, each component will be tested individually first before being connected to a strip or multiple.

F. *Audio*

Testing the audio means to see if the microcontroller is receiving anything first. Second, to check to see if the audio received is the exact same as the audio expected, hook up a pair of speakers to an output of the microcontroller and give that output the data received.

V. Integration Strategy

The idea is to get the individual parts of the project working first before putting them all together. When the project reaches the point where all the parts can be put together, do so incrementally such that bugs are easier to find.

A. *Encapsulation*

The final assembly for everything needs to be done after the trays and acrylic pieces have gone through the laser cutter and glued together. Two of the trays will have the entire bottom cut out. One will also have the bottom cut out, except for a 1” strip lining the edge. This tray will be glued upside down on the fourth and final tray to create the tank. The acrylic pieces will be used as the stand for the tank such it rests roughly 1” above the surface. It also is the channel that feeds the wire from the pumps to the PCB.

For the pumps, only the tank and stand are needed for placement of the pumps the tubes extruding their outlets. The last two trays will be added in junction with the LEDs and likely before the completion of the service box. This way, water tests can be conducted without having to worry about ruining circuits.

B. *Pumps*

Start with 1 pump and determine how PWM will work with it using a potentiometer as an input and the STM32 Discovery Board as the medium. This also ended up being the prototype used to demonstrate a portion of the final project. The circuit used to drive the pump (Fig. 5) is the same drawn in Fig. 3 and repeated 12 times.

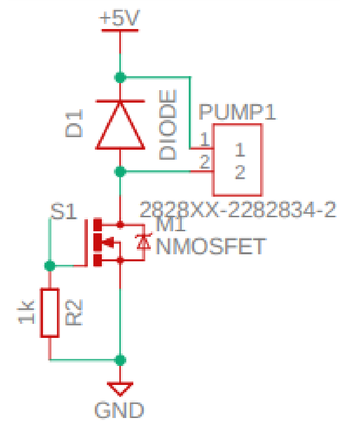


Fig. 5. Circuit schematic for a singular pump used for the prototype.

Fig. 6 shows the prototype in mid-demo. To see the full video of the demo, please contact one of the authors.

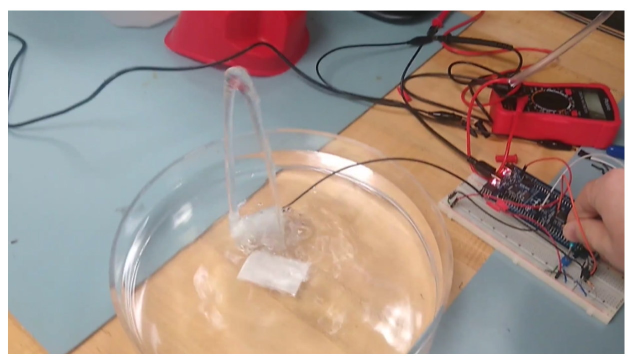


Fig. 6. Prototype.

It should be noted heavily that the microcontroller pin where input is being read in needs to be set to a continuous-read mode. Additionally, the max range given by the input must be taken into consideration. The input’s max range may differ from the max range the PWM can generate/read. Failure to do either may result in a non-functioning system.

Since 1 pump has been verified to work with the potentiometer, the next step is to control multiple pumps with the PCB. With the PCB, it is probably best to start with a single pump with a potentiometer as an input again. Once 1 pump is verified to still be working, add another and keep doing so until all pumps are working as intended. Afterwards, secure them to the encapsulation and replace the potentiometer with a wire to the microcontroller. The wire will be the output of the PWM to the pumps. Before heading to the next step, the FFT must control the pumps properly, meaning the FFT and audio input/output are likely integrated around the same time as the pumps.

C. LEDs

Integration for the LEDs will be similar to the integration of the pumps. Since the style of lighting has yet to be fully determined at the writing of this proposal, the exact steps to get the system working has not yet been conceived. It will need to work in tandem with the pumps even though the PWM signal it's receiving is from a different pin.

The LEDs are the last step of the integration process, but before testing of the entire project as a hole, a box needs to be created for all the electrical component. Additionally, the tank must be completely sealed at the point where the pump wires come out and at the edges were the two trays meet.

VI. Risk Assessment

*1) **Audio Delay**:* Despite using an improved FFT algorithm to improve time, there is still the chance that real-time computation cannot be achievable. In that case, a small delay introduced at the beginning of the audio input may need to be introduced.

The idea is to jump-start the calculations such that the algorithm is looking ahead. What the user is seeing and hearing is upwards to several seconds behind what the algorithm is computing, thus creating the illusion that the audio is being processed in real time.

*2) **Variable Voltage**:* The voltages between the PCB and the microcontroller are incompatible or the difference is too much to buffer. Or there was a miscalculation on the power needed for all the pumps. Adding too much power to the component would only result in magic smokethat is, the component would be rendered useless. At that point, a new component is required. In case time is not friendly, spares will be set aside for this event.

Too little power, on the other hand, will result in the components not working as expected. For this scenario, heading back to the calculations and conversation with fellow Computer Engineers will, hopefully, solve that issue.

*3) **Lack of Sealant**:* Insufficient sealant was added around the wires where they exit the enclosure. Water leakage from those points can get to the circuit boards and fry the components. Just like in the previous point, new components would be needed should this happen. The hope is to apply enough sealant such that such a scenario doesn't come to pass. At the same time, however, too much sealant cannot be applied else risk the enclosure looking ugly.

*4) **Visual Delay**:* Much like the risk discussed in VI-1, one of the many challenges faced in this project is getting the computations done quickly enough such the effects of the visualizer displays in real-time. As with the Audio Delay, a small delay may need to be introduced in the beginning such that the algorithm is ahead of what is being seen and heard.

*5) **Synchronization**:* Alongside computational delay, both the audio and visual effects need to be efficiently synchronized such that one is not faster or slower than the other. While it is not on par with the frustration found with un-synchronized audio/video in a movie, an un-synchronized visualizer will likely kill the simulation it's supposed to bring.

VII. Final Sketch Details

As mentioned in Section II-8 Final Assembly, the final product will tentatively be composed of 5 areas: the tank, stage, service box, speakers, and input.

*1) **Tank**:* Contains the pumps, water, and LEDs of the project. It's constructed of 2 trays, 10" in diameter and 2.5" in height. The openings of these trays are glued together with the bottom of the upper tray hollowed out to 1" from the edge. This 1" strip is the spot where the tubes from the pumps are set into place. The hollowed out part of the tray is the area the water falls back into after exiting the tubes (Fig. 7, #1).

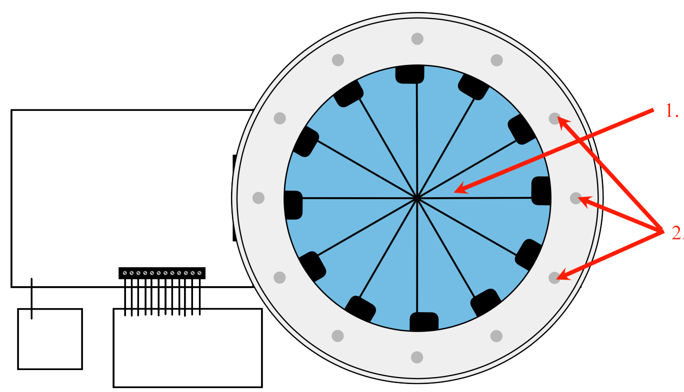


Fig. 7. Top-down view of the final product at the stage area. #1 is the open area where the water falls through to the tank to be reused. #2 are the holes to be drilled for the tubes and are positioned slightly more inward than the spouts of the pumps to ensure the water flows inward instead of straight up.

The pumps will line the bottom, evenly spaced (Fig. 8), and submerged in water that is at least the height of one tray (Fig. 9). The LEDs, although not shown will line the edge of the upper tray. The bottom of the trays are slightly slanted inward, but additional material may need to be added such that the LEDs hit the water at the right angle to illuminate them.

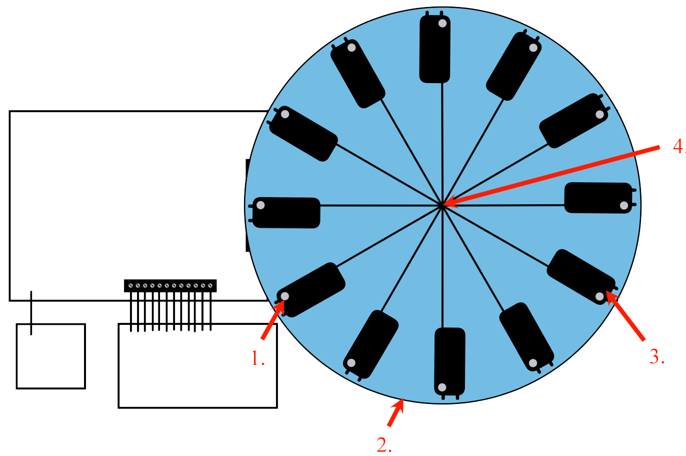


Fig. 8. Top-down view of the final product on the tank area. #1 is the location of the pump spout. It's not very well shown, but #2 points to the fact that the bottom of the tray is slightly smaller than the top. #3. The pumps may need an adhesive to stay in the same spot. #4 is the point where the pump wires go through the trays to reach the PCB.

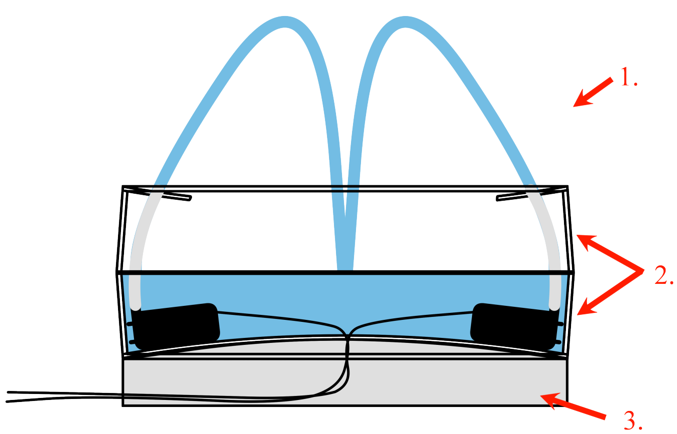


Fig. 9. Side view sketch of the final product where: 1. Either use the other 2 containers to encase the water that comes out of the tubes or get some stiff vinyl to line the sides; 2. Trays are the same–the top tray is the bottom that is flipped with the bottom cut off except for 1̃"; and 3. Stand to raise tank and allow wires to pass through.

*2) **Stage**:* The stage sits on top of the tank. It is the focus of the entire project where the water and light dances to the tune of the music. As noted in the caption of Fig. 9, the stage can either be encapsulated with 2 additional trays, their bottoms completely hollowed out, or stiff, clear vinyl. The idea with the latter is is to create additional strength for the tank as the vinyl would also encase the outside of the tank. However, due to the slanted nature of the outsides of the tray, other material or glue are needed to fill the gaps and create that strength.

*3) **Service Box**:* The bottom of the tank will have a small hole at the very center that acts as the outlet for the pump wires (Fig. 8, #4). Two pieces of custom cut acrylic set at about 2" away from the edge of the tank and about 1" above the surface will act as a gateway for the wires to the PCB.

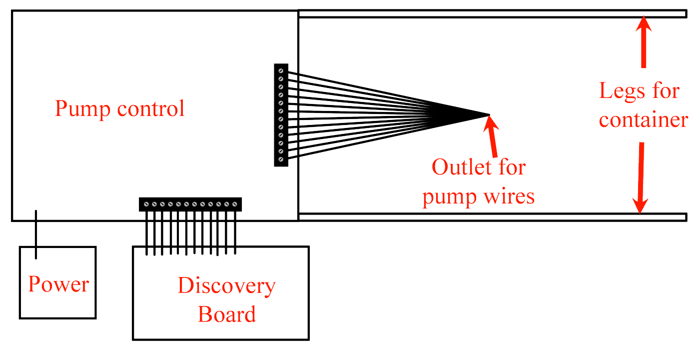


Fig. 10. Top-down view of the final product at the first layer, the service box.

In Fig. 10, the PCB is shown to be the same size as the gateway when in reality, the gateway is about 5" wide while the PCB is 3" wide. This allows for the PCB to be hidden underneath the tank and maybe even the Discovery Board. If not, then the gateway will lead into a small box where all the exposed electronics will stay. This box, the service box, will have a removable top for easy access and two holes where the wires for the AUX cord and the speakers exit.

For appearances, the service box is to be facing away from the user at all times.

*4) **Audio Input/Output**:* Not shown in any of the sketches is the input/output for audio, otherwise known as the speakers and input areas of the project. As mentioned earlier in the proposal, two speakers will serve as the output and will be placed on either side of the encapsulation. The input is an 3.5mm audio jack that can be accessed via AUX cord coming from the service box.

It should be noted that Fig. 7, Fig. 8, Fig. 9, and Fig. 10are initial sketches of what the final product should look like. The actual, physical construction may change as the project progresses to account for any obstacles that may appear.

VIII. Bill of Materials

The following list contains the materials required for this project:

1) Water pumps + tubing - specifically ones used in aquariums such that they are smaller and more wallet-friendly.
2) Speakers - one pair to start with; more will be added later.
3) Water pump driver PCB - either one created by the students or found online.
4) LED driver PCB - either one created by the students or found online.

5) LED strip
6) Microelectronic components - for PCBs
7) Container - preferably clear acrylic for components and water.
8) Sealant - to ensure water doesn't leak to where electrical components are stored.
9) Micro-controller - STM32F072RBx (already own)
10) Aux port part - for input
11) External power source - capable of 5V and at least 5A
12) Water jug (1 gallon) - to be used as the source of water during testing and final product

This list may not reflect everything that will be used in the final assembly of the project.

A. *Vendor List*

1) Amazon
   - 4 water pumps + tubing - $10.59 x4
     - ASIN #: B097F4576N
     - ordered 3/17/2023 - $45.44
   - 5V/5A AC to DC Power Supply Adapter - $16.49
     - ASIN # B078RT3ZPS
   - Smart LED strip lights - $23.60
     - ASIN #: B07N1CMGQQ
     - ordered 2022-Jun-15
2) Lowe's
   - Sealant - $7.00 (Includes 10% employee savings)
     - item #: 727380
   - Waterproof double-sided tape - $8.98 (Includes 10% employee savings)
     - item #: 394705
   - Clear waterproof one-sided tape - $8.08 (Includes 10% employee savings)
     - item #: 488028
   - acrylic sheet 8-in W x 10-in - $4.14 (Includes 10% employee savings)
     - item #: 55844
3) LO Florist Supplies
   - Acrylic Trays 10" x 2.5" - $7.99 x4
     - SKU #: ACYL102
     - ordered 2023-Mar-17 - $45.02
4) Digikey/Mouser
   - Audio jack port - $0.84 x1
     - model #: STX-3120-3B
   - Transistors (Surface Mount) - $1.71 x12
     - model #: IRLZ34NSTRLPBF
   - Resistors (Surface Mount) - $0.10 x12
     - model #: RC0402FR-071KL
   - Diodes (Surface Mount) - $0.40 x12
     - model #: B140-13-F
   - Terminal blocks - $1.38 x12
     - model #: 282834-2
5) UofU ECE Stockroom
   - any additional microelectronic parts or tools we will need
6) OshPark
   - PCB manufacturing
     - Pump driver (3 boards)- $71.45

IX. Group Logistics

There are only two collaborators in this project, so rather than a group or team dynamic, the collaboration is more of a partnership. The work is to be equally divided, although further discussion on future tasks and how they are to be divided may/will be necessary.

A. *Communication Plan*

Current communication is mostly through Discord and brief after-class discussions. The set schedule as things progress will be Mondays and Wednesdays and will take place either virtually for programming components or at the senior lab for hardware components.

B. *Schedule & Milestones*

The Gantt chart below (Fig. 11) is a tentative schedule for how the project will proceed.

The intention is to get all hardware purchased and assembled by the end of August with some programming ready to load onto the microcontroller. This way, the remainder of the semester can be focused on the rest of the programming and final testing and debugging.

X. Conclusion

All-in-all, this capstone project is like a combination of the Bellagio Hotel water show and commercially-sold speakers with the four water pumps. The main difference is it shrinks the Bellagio's fountain to a more personable size and can be operated at any time of day. It also doubles, triples, or even quadruples the number of pumps used to create more of an effect that is in tune with the music.

To create such an effect, the Cooley-Tukey FFT algorithm will be used to quickly create digital data from the audio signal, from where the varying heights of the water columns are computed. LEDs are added to the mix to enhance the experience further. The calculations for how those will appear in terms of color and maybe even pulse is also determined by the data given by the FFT. The algorithm will be found on the microcontroller to mastermind the entire water show. Here is also where the speakers and controller PCBs for the LED and pumps will be connected.

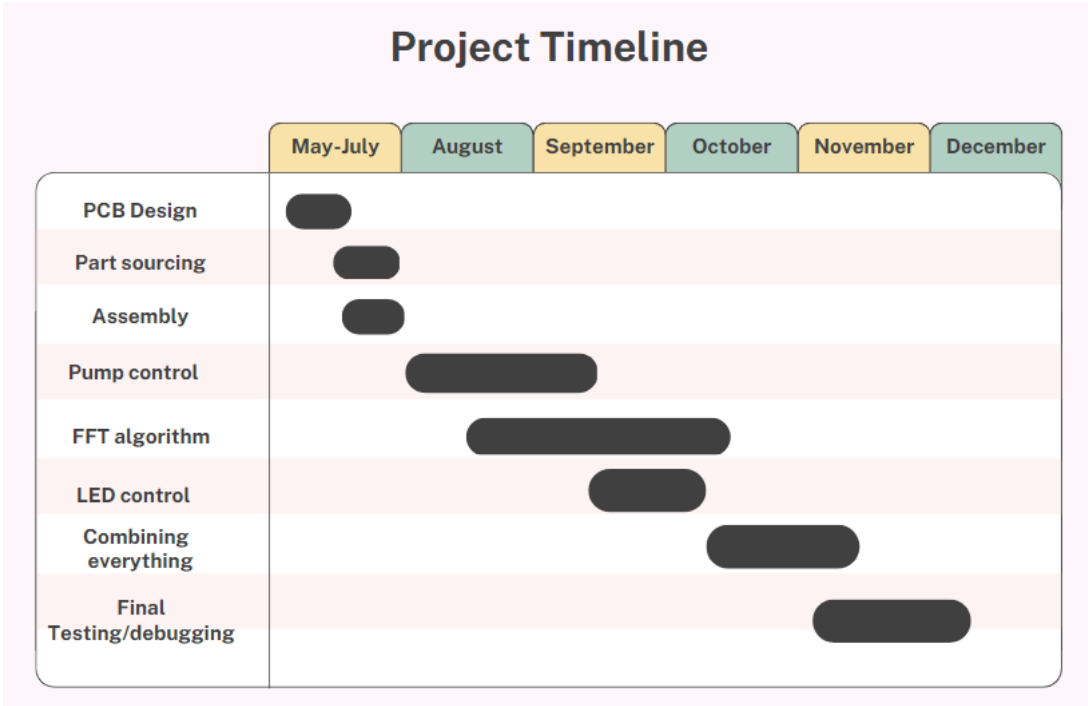


Fig. 11. Tentative project schedule.

The main jewel of the project is a clear, acrylic cylinder that is 10" in diameter and several inches tall. It contains the water, LEDs, pumps, and tubing to create the waterworks. A set of speakers accompany the container on the sides to output the audio originally given. Finally, on either the back or bottom of the container is the "service box" in which the electrical components are stored.

XI. Links

Website: https://my.eng.utah.edu/~byee/index.html

Repository: https://github.com/BYee7127/public_html

References

[1] Adafruit. *24-Channel, 12-Bit PWM LED Driver with Internal Oscillator*. 2008. URL: https://cdn-shop.adafruit.com/datasheets/tlc5947.pdf.

[2] Anon. *FFT Size*. [Online]. URL: https://www.spectraplus.com/DT_help/fft_size.htm. [Accessed: 2023-Mar-22].

[3] NTi Audio. *Fast Fourier Transformation FFT - Basics*. [Online]. 2019. URL: https://www.nti-audio.com/en/support/know-how/fast-fourier-transform-fft. [Accessed: 2023-Mar-22].

[4] David. Forney and Deborah I. Fels. "Creating Access to Music Through Visualization". In: *2009 IEE Toronto International Conference Science and Technology for Humanity (TIC-STH)*. Toronto, ON, Canada, Sept. 27, 2009, pp. 939–944. URL: https://ieeexplore.ieee.org/abstract/document/5444364.

[5] Eric J Isaacson. "What You See Is What You Get: on Visualizing Music." In: *ISMIR*. 2005, pp. 389–395. URL: https://ismir2005.ismir.net/proceedings/1129.pdf.

[6] Ben Levinson. *The Fast-Fourier Transform and Spectrograms for Audio Visualization*. [Online]. Aug. 27, 2017. URL: https://benlevinson.com/projects/spectrograms. Accessed: 2023-Mar-22.

[7] Microsoft. *Microsoft Windows Media Player 7 Brings Click and Play Digital Media To Millions Around the Globe*. [Online]. July 17, 2000. URL: https://news.microsoft.com/2000/07/17/microsoft-windows-media-player-7-brings-click-and-play-digital-media-to-millions-around-the-globe/. [Accessed: 2023-Mar-16].

[8] A. Rothstein. *The Basics of Music Visualizers*. [Online]. Dec. 6, 2019. URL: https://www.ipr.edu/blogs/audio-production/the-basics-of-music-visualizers/. [Accessed: 2023-Mar-16].

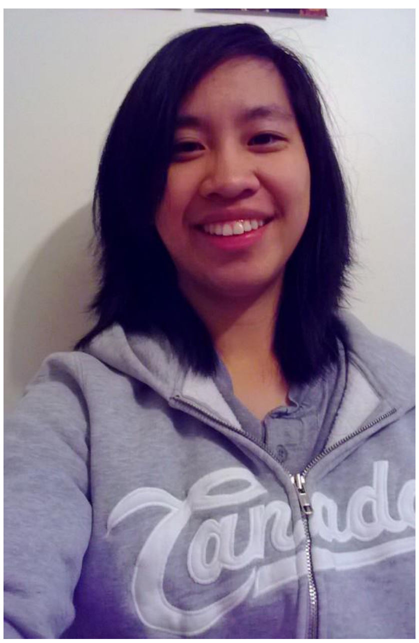
[9] SparkFun. *16 Channel LED Driver with Dot Correction and Grayscale PWM Control*. 2004. URL: https://www.sparkfun.com/datasheets/Components/General/tlc5940.pdf.

[10] Wanda Marie Thibodeaux. *What is Music Visulaization?* [Online]. Feb. 7, 2023. URL: https://www.musicalexpert.

org/what-is-music-visualization.htm. [Accessed: 2023-Mar-16].

XII. Biographies

Alex Gray is a student at the University of Utah pursuing a Bachelor of Science degree with a major in Computer Engineering and a minor in Arts Technology. He has done work with embedded system design and software development with a focus on the digital media and arts industry, including image and audio processing as well as interactive computer graphics. He is passionate about utilizing computers in the media and arts industry thanks to his experience with graphic design, electronic music production, and digital photography.



A 2nd generation Chinese-American, Beverly Yee is a senior at the University of Utah pursuing a Bachelor of Science degree in Computer Engineering. Her specialty lies in web programming and digital media, but has worked with computer hardware, digital system design, and software development. Beverly has experience with graphic art design and does sketch/digital art as a hobby.