Nhu Nguyen ES95 Prof. Lehrman Final Project report

# The Desk Instrument

Group: Peter Wu, Paloma Ruiz-Ramon, Nhu Nguyen, and Parker Heyl



#### 1. Introduction:

Our initial goal for the Desk instrument project was to create something that is fun for musicians to play. We were aiming for a design that is intuitive, with large controls to manipulate, but also including simple and familiar features like keyboards or controlling wheels. Our project went through two phases, and the second phase was to renovate the instrument with new features, functions, and new hardware (plus replacing old parts).

We knew that we wanted to output sequences of notes as chords, and initially aimed for some kind of buttons combination set up to achieve this purpose. With so many features that are large for visual purposes (to promote player-instrument interaction), we decided that a desk would be a nice set up to put everything together. Some factors that makes the desk such an ideal frame for the project is that it already has a flat surface to attach controllers, the drawers (moving in the y axis) could be uses to attach IR sensors or slider potentiometers. My playing while standing, the musician is more engaged with the instrument, and more expression could be achieved using both upper (hands) and lower (feet, hip) body parts.

The instrument we created used a variety of sensors that reads in digital signals (push buttons) and analog signals (knobs and sliders). These switches, buttons and sensors are hooked up to an Arduino Mega, than are processed by Max and Reason software to produce interesting sounds.



## 2. Design Goals:

Initially, we came up with a sketch for our design (Figure 1).

Our initial design comprised of an octave length of keyboard keys, a wooden pitch bend bar (which we call "whammy bar"), a rubber ball that detects squeezing for reverb, a set of toggle buttons to play chords, and 2 wheels (a normal size bike wheel and a mini bike wheel). Reed switches and magnets are attached to these two wheels to set off sequences of notes played. Moreover, the two drawers (with IR sensors) on the down-right control reverb and delay, while the foot pedal (with push buttons) allows us to switch between octaves on

the keyboard. Over time, the design changes, however, the

Figure 1. Initial design

functions of the instrument and the feature it has stays pretty much the same.

The aim of using wheels for the instrument is for the looping effect. Our group thought that it would be vibrating if the musician could be able to assign chords while playing, and let the wheels play the notes in the chord.

## 3. Official design:

## Hardware:

List of sensors used:

Slider potentiometer (5): whammy bar, chord volume, lead LFO, Chord LFO, lead noise, and lead volume

IR sensors (2): attached to 2drawers to control reverb and lead echo amount

Push buttons (24): 2 octaves long keyboard set

Toggle buttons (21): 19 chord buttons, one octave setting button, and one to set off a drum loop sample.

Reed switches (2): for two bike wheels. Linear FSR (1): for filter frequency (attached on whammy bar)

List of other parts:

Desk

Bike wheel (2): 1 small and 1 normal size bike wheel

Arp wheel (clothesline pulley)

Whammy bar (wood)

Chord box (acrylic, laser cut)

Keyboard keys (3D printed)

Spring (Whammy bar)

Others: springs, wood blocks, wires, Arduino Mega, etc.

## Construction

One of the main features of the instrument is the looping effect, done by simultaneously changing chord buttons and rotate the small wheel on the right, or spin the big bike wheel, and change its envelope (LFO amount, volume) in real time. While the rotating bike wheel would be use to continuously play the notes set off by the keyboard. In the design process, we envisioned that the patch like set up of the toggle buttons would be used to initialize chords (major, minor, diminish, augmented...), and the arp wheel (smaller wheel) would rotate to set of sequences of notes in the chord. Other effects would be controlled by the slider potentiometer, the whammy bar (the linear FSR on top), and the drawers. For example, pressing down two notes: major (blue) and C (white) simultaneously (Figure 2), while spinning the arp wheel would set off the C major chord note sequence, and loop them over and over (Figure 3).



Figure 2: pressing down to play C major



Figure 3: arp wheel

The toggles buttons are simply push buttons that are connected to the digital port in the Arduino, and pushing down the buttons triggers the chord to be played. The reed sensor that is set up on the side arp wheel responses to the magnet that is attached on the rim of the wheel. Every time the magnet passes by and "closes" the reed switch, it sends digital signal that sets off the sequences of chord notes over and over again. Initially, a metal clothesline wheel was used instead, however, since it failed to be sturdy, it is replaced with a heavier wheel.

To the right of the toggle chord buttons is the two octaves of keyboard keys that we had 3D printed in the lab. The physical keys are used to trigger a system of small push buttons right below



them. There are 24 keys (black and white altogether), requiring 24 push buttons hooking up to the digital ports of the Arduino. The keyboard simply have a wooden dowel run through them as rotating axle, making them flexible to movement, thus, allow us to flip up the keys for trouble shooting (Figure 4). During the second phase in the project, to make the keyboard set sturdier, we attached a long moveable woodblock on top to prevent the keys from shifting up-down while being played (Figure 5).

In the second phase of the project, our team added a toggle button that allows us to assign octaves to the keyboard keys. Pressing the button would change the octave numbers from 1-2 to 3-4, 5-6, 7-8 and back.



Next to the keyboard set on the side of the table is a rotating bike wheel. This big wheel has almost the same setup with almost the same purpose as the arp wheel for the chord buttons. The wheel has sets of magnet attached to it and is allow to spin, which will make the magnets "close" the reed switch near the wheel's rim, creating the LFO effect. Similarly, the bike wheel's purpose is to work with the keyboard. The reed switch, would send out commands to create LFO effect for the last note played on the keyboard every time the magnets pass by. Thus, keep spinning the bike wheel and varying the notes on the keyboard would change the loops of notes played in

Figure 6. Bike wheel with reed switch and magnets

response. The wheel, or the LFO wheel, is hooked up to the digital port in the Arduino.



Another feature that controls and

effect the notes played is the pitch bend bar, or "whammy" bar (Figure 7). The whammy bar is set up to transfer the up and down motion of the wooden bar (with the help of the spring), to the up and down sliding of the slider potentiometer. The slider potentiometer is then hooked up to the Arduino analog port. Slamming the whammy bar up and down while playing the notes on the key board gives the instrument a pretty unique feature. Playing the keyboard while spinning the LFO bike wheel, and using the pitch bend bar at the same time create a plucking eastern string instrument-like sounds!

Eventually, our team came up with addition features that we could implement using the whammy bar. We thought that it would be interesting if the position pressed on the whammy bar could also determine the filter amount. The idea was simple, however, making it work with the right sensor was a struggle! Initially, a long ribbon sensor was perfectly attached along the upper side of the whammy bar. The ribbon sensor has three zones, which act like three different resistor sensor, which promised us that we could implement three more features. However, we could not get nice enough readings from the sensor, and bending it along the curve of the whammy bar greatly interfere with the reading. We finally resorts to a short linear FSR, which works consistently well.

Speaking of analog connection to the Arduino, there are slider potentiometers that determines the chords volume, the lead LFO amount, the chords LFO, the lead noise, and the lead volume. In the beginning, these controllers were rotary switches, which were not as visually appealing. Changing them to slider pots makes it easier for controlling purposes.

Our project goal is to create a fun a visually appealing music instrument out of a desk, so we thought that we should use the drawers that are already there to control some kind of sound effects, We end up turning the upper drawer and lower drawer on the right side of the desk to be reverb amount and lead echo drawers. At the back of each drawers we attached an IR sensor that detects the distance from the attached point to the back of the desk as the drawers slide in and out.

### Circuit and circuit modification

The circuitry for our instrument is fairly straightforward. The switches, toggle buttons, and keyboard keys are hooked up to the Arduino digital ports, while sensors such and slider pots, FSRs are connected to analog ports of the Arduino Mega. In the beginning, our wiring was very crowded since we were connecting the power and ground of the buttons and sensors separately.

Later on, the wiring was modified and organized by grouping together power and ground connections, and running the power and ground port of the Arduino to different part of the desk.

Software: function of Max patch and Reason patch

The Max patch

Besides specific functions, our group's Max patch has a user interface that allow us to read the signal follow what is going on with the instrument. Moreover, in the user interface patch, there is also an initialize button that allow Max to reinitialize all the values every time the Arduino is reset up. This feature allows us to minimize difficulties in the building and testing process, as well as interact better with the instrument in real time.





Figure 10. The keyboard set up



## Figure 11. Data processing for analog input: pitch bend, reverb, lead echo, and LFO

Some modifications our group made for the final phase was to add some a feature for setting off the drum loops, as well as a patch that allowed us to switch between channels and octaves for the keyboard.

The Reason rack used includes two NN19 subtractors (one for the chord buttons notes and the other for the keyboard notes), an RV7000 for reverb sound effect, and an Echo rack. The Reason module basically reads in outputs from Max (Max 1), synthesizes sound through the subtractors and produces the desired sound input. Moreover, it is also programed so that our sound controllers' analog data values (reverb, lead echo, etc) would be rescaled in Max, and output to control the controlling wheel in Reason for real time effects.

In the second phase, we got around to creating a few more synths, more reason patches, and drum loop samples that we could use for the instrument. There is a toggle button to push that



000

00

0

 $(\mathbf{\hat{o}})$ 

00

Ø

Ô

(e) FM (e)

P

Q

•

Ø

0

(

(0)

0

00

0

TRACTOR

Ô

Θ

Ø

ECHO

Q

O O

would set off a drum loop sample.

#### 4. Discussion

During the building process, there were many problems that we encountered and had to overcome. Coming up with the right materials for parts was initially a challenge. We knew that we wanted some kind of box-like set up to layout the chord buttons. In the beginning, we decided on a wooden box. However, we failed trying to drill big holes for toggle buttons on thin wood panels (they cracked), and acrylic sheets were shattering too easily if hand drilled. So we decided to laser cut assembling pieces for the box and glued them together, which became very sturdy and neat.

Before the actual building process started, we all agreed that we wanted a long whammy bar for pitch bend, so I decided to start ahead a build it out of hard wood, with angled tongue-groove joint, which came out to be sturdy for its purpose.

For the rotating wheels (small arp wheel and bike wheel), we wanted them to be able to rotate freely with the least amount of friction as possible. Our choice for the big bike wheel and the clothesline pulley has been working well with our design. Later on, since the pulley was not sturdy enough, we replaced with a smaller bike wheel (heavier).

Another big challenge was 3D printing the keyboard keys and assembling them. The keys had to be design in CAD such that they would be able to grip and push on the push buttons. This tasked required a significant amount of time (especially to get all the keys printed).

Besides material and structure building, a lot of the problems we ran into involved circuit set up and connections. On a regular occasions, the analog signals were messed up when a button is triggered, there were a lot of inconsistent data floating. Most frequently, the high B note went off every time there is a change in signal in the circuit. At first, we thought that it might have been due to several bad ports in the Arduino, but one of our group member (Peter) suggested that it might be insulating problem between the Arduino the the desk surface it was attached to. After we have insulated the Arduino with rubber sheet at the back the noisy signals went away.

One other problem we ran into was that the slider pot's response from the whammy bar significantly delayed, it turned out that we were averaging points using dl.stream on the Max patch. Thus, switching to the zl.stream fixed the problem.

Most of our sensor work very well after being wired up. However, toward the end when we were attaching sensors to the drawers for reverb and echo, we decided to use slider potentiometer to make things easier. It turns out that the slider pots was old and not functioning properly so we switched them out a replaced with IR sensors.

To improve on what we already have, we could add more features to the instrument, such as switches for polyphonic and monophonic mode (it would be fun to play solo when needed). Our two rotating wheels right now have difficulty stopping the loop (we would have to jiggle them a little to stop), which means having terminating buttons would be much appreciated. After the first phase of the project, our team went on to implement the rubber ball from our initial design. The ball works perfectly fine with a square FSR underneath, and the data is consistent after averaging. However, the rubber ball was in the way, and did not turn out to be important, nor cooperated with the player's movement, so we decided to not proceed with that feature.

The team also had an idea about attaching IR sensor in the front drawer to set off drum loop, which we could stop by shutting the drawer to stop (cool beginning of a performance, maybe). Due to time constraint, we could not get the IR sensor to work, but used a toggle button instead. This would be something I would like to improve in the project. Another feature that I think would be something we could have done if we had time was to add a row of small circular FSRs on the front rim of the desk to play drum tap sounds. Furthermore, IR sensors at the bottom of the desk that could be controlled by the player's feet/ standing position would be something really cool to implement. After all, the desk instrument is supposed to be fun to play!

Overall, the project was successful. We got to finish everything that we expected to get done after trials and errors. Our initial goal for the project was to make something big and fun to play, which we have achieved. Big and small wheels, colorful buttons, loop chords as background, pulling desk drawers in and out for sound effects is definitely something you do not get to do every day. Instead of playing a sample, the instrument still allow the player to compose their own music, and put in creativity. Comparing the traditional/classic musical instrument, our desk does not have 3D response (pressure, velocity sensing...) yet, but it allows the musician to loop over chords (which is what pianist usually do), while playing notes on the keyboard! If we could make more improvements to the instrument and make it more responsive to musician's physical touch, we would be able to do so much more with it. Furthermore, since the setup is simple and similar to traditional musical instrument (piano keys, chords), it would not take the user to long to pick up on how to play it.