Emma Pannullo
EMID Final Project Report
5/10/19

Musical Drive Console

Introduction:

The motivation behind this project was to use parts of a car driver console to create a MIDI controller. Many people listen to music in their cars, be it on the radio, on a CD, or using their smartphone. I know I associate being in the car with listening to music so we thought what if instead of listening to music in your car, you could actually create music using your car. Additionally, we spend a lot of time driving in cars- we are very familiar with the layout and many of the controls come second nature to us. For example, if you want to turn left, you probably just put your turn signal on and then turn left without even thinking about it. Sometimes new instruments are difficult to learn because the layout is foreign. This instrument has a very familiar layout, making it easy to use.

Construction:

We first picked out some of the key features of the drive console: steering wheel, gear shifter, windshield wipers, turn signals, gas pedal, and buttons on the steering wheel. We wanted to focus on building the driver console first. The console is made almost entirely out of wood and is a portable tabletop instrument. There is a large wooden piece that sits at an angle, similar to a dashboard in a real car, and is supported by two triangular wedges and a large wooden base. These pieces were cut and glued together to make the main support for the instrument. The steering wheel we bought to modify originally had a center button, but that was removed in order to attach a round wooden piece and a wooden dowel to the back of the wheel. A circular hole was cut in the board to allow the wheel dowel to go through and additional wooden supports were added to allow the wheel to be held up and to also rotate when one turns it. The wooden dowel was also hollowed out through the middle in order for wires to be fed through it to the back of the drive console. This way the wires would also turn with the wheel, providing some flexibility and also allowing the wires to come through the back of the console and avoid messy wiring.

Other components required mounting on the console- this was mainly done using the X-carve to create holes in the console and hot glue to hold components in place. The joystick was mounted on the console by adding additional material between its two plastic plates, a small piece of wood was attached to the button mechanism in order to help it reach the button connect when pressed. To try and keep the instrument clean and make components flush, all of the components were measured ahead of time. A test board was used to make test cuts using the

X-carve. This helped ensure that the pieces would fit snugly into the board. For the windshield wiper and turn signal pieces, we chose to use more robust switches as the first ones we picked were too small and pushing the switch just a little turned it on, where as we wanted there to be a more significant gesture attached to flicking the switch on. Using these new switches, an extension piece was made using the 3D printer to make a piece that attached to the lever of the switch and extended out at an angle to clear the steering wheel.

We also chose to include some indicator LEDs in three different forms: a circular strip, a straight strip, and then four individual multi-colored LEDs. Similarly to the other parts, we used the X-carve to cut holes in the wooden dashboard to fit these components. The entire wooden structure was painted black and other components either had accent colors of blue, red, or orange to keep a sleek look. A few laser cut pieces were created as well, mainly to help describe some of the indicator LEDs. For example, the straight LED strip is used to indicate to the user what octave is currently being played, so each LED on the strip lines up with a number cut into a blue piece of acrylic.

Sensors and Electronics / Use of Arduino:

Most of our sensors were digital sensors: LEDs, switches, buttons, rotary encoder, and a joystick. The joystick was just four switches placed in a square with a button in the middle for a total of 5 digital outputs. The only analog component used was a softpot. All of the components except the softpot and the individual LEDs could be connected directly to pins on the Arduino- since the code was reading ground as “high” the common pin for the buttons and switches was tied to ground. The LEDs worked the same way but needed also required a resistor in their circuit. We created a common ground board to take the ground of the Arduino and share it with several different pins in order to prevent a ground loop and to clean up wiring.

Additionally, the softpots needed a resistor tied between the middle output pin and ground in an effort to reduce noise. The LEDs had data pins going to the Arduino and the 4 individual LEDs were powered off of a separate 5V power supply as to not over-exert the Arduino power. Separate Arduino code was written to control the LEDs. The circular strip represented the starting note of the scale and the straight strip represented the current octave- so these two needed to share data. The 4 individual LEDs were indicator LEDs that flash accordingly when the joystick is moved in a certain direction. Figure 1 below shows all of the sensors on the drive console.

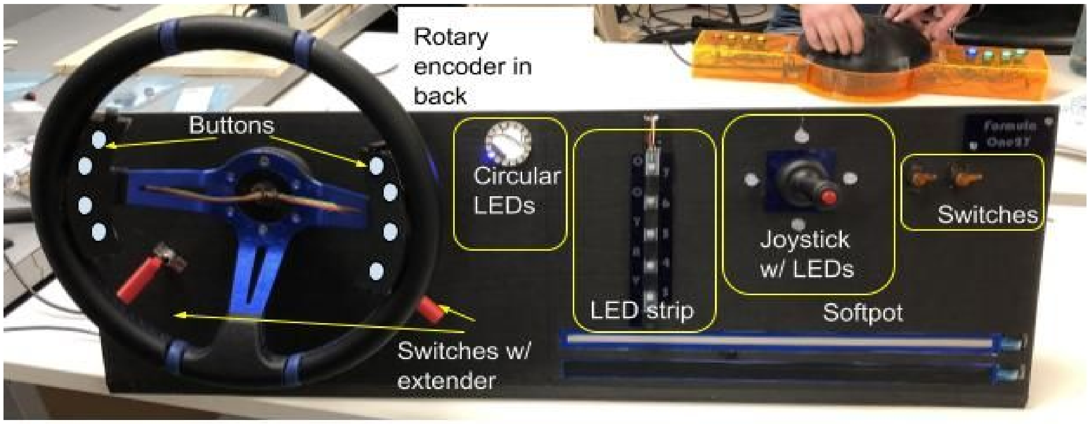

Figure 1: Console Diagram

MAX Processing:

Table 1 below describes the functionality of each sensor as it is defined in the Max code. For the button presses, when the button is pushed down, Max sends a note on. When the button comes back up, Max sends a note off. To switch between the different scales, there are subpatches used for each note as well as gate objects to determine which scale to send the data to. The analog softpot had a range of 0 to 1024 so that needed to be scaled down to a scale of 0 to 127 in Max. For the rotary encoded, each state changes the starting note. So as you turn the wheel, the notes increase clockwise or decrease counterclockwise.

Component	Control
8 Buttons	Each button plays a note in a scale
Right Shifter	○ MIDI channel select
Left Shifter	○ Scale select
Joystick	○ ↑ up an octave ○ ↓ down an octave ○ → down whole step ○ ← up whole step ○ Button: turns on / off the second filter
Softpot	Pitch bend

Right Switch	LFO 1 on Connected to the filter frequency
Left Switch	LFO 2 on Changes vibrato [Only for the first 2 patches since third is a sample patch]
Rotary Encoder	Starting note of scale

Table 1: Component Gestures

Figure 2 shows the subpatches for the Arduino 2 Max code. As you can see the analog and digital components are broken up. There are also subpatches that deal with changing the value of the notes- be it a whole step, half step, octave, or changing the entire scale to be major, minor, or blues. Additionally, there are subpatches that handle changing the MIDI channel to be used in Reason as well as utilize pitch bend, filter, and LFOs.

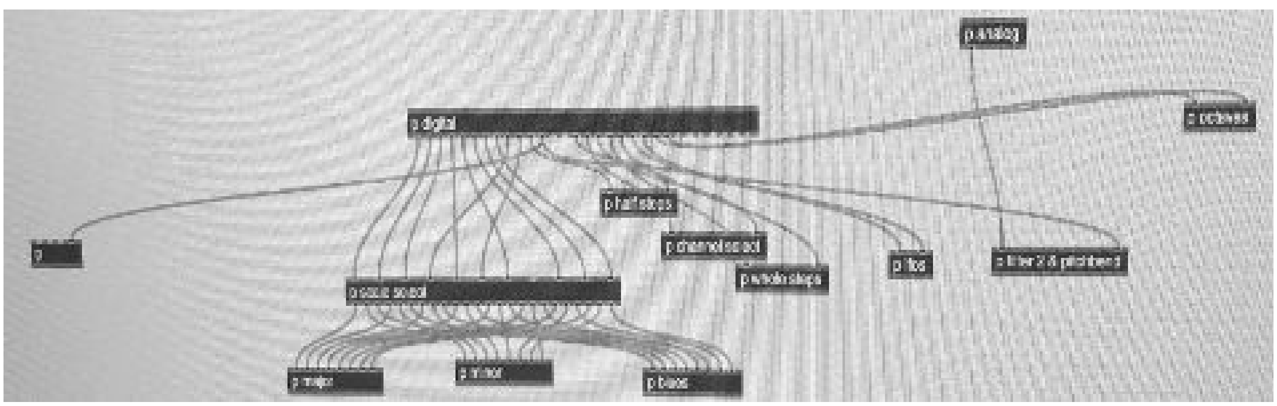

Figure 2: Arduino 2 MAX

Synth / Sound Effects:

Our Reason patch consists of three different channels. Two channels are subtractor patches and the third is a sample patch using the NN-19 sampler. The first channel was made to sound like a car engine starting with a bass tone and using medium LFO rate and a low pass filter to get deep undertones. The next channel was made to sound like a car horn beeping with much higher frequencies and relying less on oscillations. This was a modified synth patch keeping more of the higher frequencies. A fun additional patch is the sample patch which maps several different car horn sound samples to different note regions. The way this works is as you turn the wheel you discover new samples.

Next Steps:

Given more time, there are some additional things we did not get to or could improve upon. The rotary encoder code was not sensitive enough for what we were trying to use it for. A full turn of the wheel only went up or down about 3 notes. We wanted it to be more dramatic such that turning a little changed the notes more. Additionally, we were encountering some issues when linking the LEDs and the joystick between Arduino and Max so the indicator LEDs were off from the actual data. We also ran out of time but including a "gas" pedal to sustain notes would have also been a nice touch. Another idea suggested that we wish we had time to implement was to make force sensing grips on the steering wheel, possibly to be used for volume or maybe attack of each note played. I think having a few switches that triggered a drum sequence (maybe also made up of car doors slamming or something fun related) would have been a nice addition as well. As it stands, the instrument is fun but could benefit by adding some depth so playing over a drum or chord sequencer might be a good next step.

Member Tasks:

Although we all contributed to all aspects of the project in some way, I think we all found our own special rolls. Will contributed a lot to the construction of the instrument. He spent a lot of time in Nolop cutting and measuring and gluing and coming up with clever ways to attach components to the drive shaft. He also spent a lot of time making test cuts with the X-carve to ensure we were being precise and would not ruin our final wooden board.

Brandon helped with soldering as well as some cutting of the delicate wooden pieces of the button housing. He also helped Tim do some debugging with the Reason patch and figured out how the rotary encoder worked as well as the circular LED strip. Brandon had his hands on a little bit of everything- he also helped reduce the noise on the softpot and overall helped everyone out. Tim put most of his efforts into the Max and Reason software but also helped some with construction. He wrote most of the Max patch, with some of the teams input on what should be mapped to what and helping debug when necessary.

I found myself trying to manage the project- setting up meeting times and making sure we were all on the same page. I also worked mostly on the wiring of everything. I wired components and decided where the components should go, how they needed to be housed, how the wires would have to fit though the system. I tried to label all of the wires so we could keep track of how things needed to be connected to the Arduino to make Tim and Brandon's jobs easier.

Team Evaluation:

Working with this team was really great- we all worked really well together and there was always a set of helping hands at the ready, no matter what the task. That being said, we also all found our piece of the project because our skill sets were well distributed. I believe that everyone put in the same amount of time and passion into creating and developing this project. We all picked up some new skills from each other as well as used skills we personally had that could help out the team- we also had a lot of fun together.