

MY PRAYERS ARE MADE OF SILICON: PURIFY

FOR NETWORKED MOTION CONTROLLER QUARTET, FIXED AND
LIVE ELECTRONICS, AND LAPTOP SCREENS

COMMISSIONED BY ALARM WILL SOUND WITH THE
SUPPORT OF THE ERNST VON SIEMENS MUSIC FOUNDATION

ROHAN CHANDER / BAKUDI SCREAM

2020

DURATION: 11' 14"

PROGRAM NOTES:

My Prayers Are Made of Silicon began as an evening-length commission but whose development was curtailed due to the onset of the pandemic. In response, *Purify* was written as an exploratory microcosm that could be realized by performers sheltering at home and in various parts of the country.

The vision for the work was to explore what a communal sacral process could look like when separated by screens and/or digital interfaces. The result was, perhaps ironically, a sacralization of those same mediated forces, set in a hyperbolic reality wherein religiosity is a digitized practice.

The work draws performers into a cyberpunk ritual with my laptop and their screens. Each performer's movements are wirelessly configured to control parameters of a "master" computer (played by my own laptop), which then outputs sound into a shared Zoom call. This computer also streams a scrolling "score" to each performer. Each performer's score is made of a video that mirrors choreography while beaming a sequence of colored lights onto the performers.

This work is divided into three major sections, each with their own ritual function. Drawing from a variety of cyberpunk and religious literature, the movements are designed to loosely embody the actions of cutting, pasting, and sharing, three of the primary tools in digital text-editing. These movements are accompanied by three distinctive sonic palettes - a granular engine comprised of heavily processed samples, downtempo synth wave commonly associated with the 1980s capitalistic vision of the future, and Palestrina's *Jesu, rex admirabilis*. The three work to paint a chronologically porous view of religiosity, making past and future projections of faith part of a singular, anthropomorphized continuity.

The documentation you are viewing of the work is of a live Zoom recording conducted in 2021. I was interested in the poetry of a telecommunication format, the homogenizing of users in a group call, and the voyeurism of watching someone on camera. I was also particularly interested in the information communicated when you can tell that someone on a call is looking at a different tab or window, usually by noticing a change of lighting on their body or face.

In an exhibition, a series of Zoom "breakout rooms" have audience members confronted with a series of flashing lights and text prompts. This culminates in the audience attending a webinar featuring 50 bots and the four performers of the ritual. Each bot has a pre-recorded camera that documents, on webcam, a person watching a video of flashing lights. The technology for this was popularized in anonymous online chatrooms, most often in sex-tortion and blackmail cases. However, this tech, along with the flashing lights, reconfigures this into an image of a hive-mind being "blessed" by digital holy water, represented by the flashing lights in both the footage and score.

My Prayers are Made of Silicon: Purify was commissioned by Alarm Will Sound through support of the Ernst Von Siemens Music Foundation and the Matt Marks Impact Fund as part of their 2020-21 Video Chat Variations series.

PERFORMANCE NOTES:

This work has no written score. Instead, the piece is performed by streaming four separate video files (each functioning as a "part") to independent performers in real-time. A full score rendering of the four parts **can be viewed here.**

The piece operates via networked motion controllers that stream motion data to my laptop via internet protocol. The data is fed into a session that manipulates synthesizers and wavetable engines in real time. A governing midi time clock is streamed back to each performer and controls the place in your score for rehearsal purposes. This allows for performers to play and rehearse the piece anywhere across the United States on a shared Zoom or cloud communication program.

Your part, in addition to communicating the choreography and when to activate your motion control, also flashes lights in your space. It is important to calibrate the brightness on your laptop screen and your surrounding environment for this effect to be effective.

To perform this piece, you will need the following:

- 1 Wii remote, 1 modded Roku TV remote and their AA and AAA batteries respectively (to all be shipped)

- Max MSP (software)

- OSCulator (software)

The following information will explain how these are used to perform the work.

GLOVE V.2.0 AND WII REMOTE:

The apparatus works through use of an infrared camera (Wii remote) and a lightly modded Roku TV remote. Each player will have one of each. The Wii Remote acts as a sensor, reading the position of your TV remote as you move it around.

As a result, the Wii Remote is completely fixed in this piece. You will not be holding it at any point during the performance. **The remote can be placed in a variety of ways, however for optimal results, it should be placed approximately 8.5 feet away at upper mid torso height with the camera/ front end pointed towards you.** There should be nothing between you and the Wii Remote aside from your laptop/computer screen. The photo below demonstrates a possible setup for the remote:



The “glove” is worn by affixing the remote to your **right hand** with an elastic band. The face of the remote should be facing your knuckles. Slip the finger sock around your index finger and position the switch so your thumb can easily press it with the support of your middle finger. The following images demonstrate this:



While the TV Remote has an added functionality of an angle of dispersion, it’s still important that **you contour your wrist as you move towards the main face of the Wii Remote, so the head of the TV Remote is always pointed towards the Wiimote when you move.**

SETTING UP THE REMOTE FOR USE:

The Wii Remote works by transmitting a relative position to the TV Remote to a computer via Bluetooth. Depending on the make and model of your computer, the process of setting up the Bluetooth connection varies. The following instructions communicate a set up on a Macintosh computer. If you have trouble setting this connection, please feel free to email at rchandermusic@gmail.com at any point.

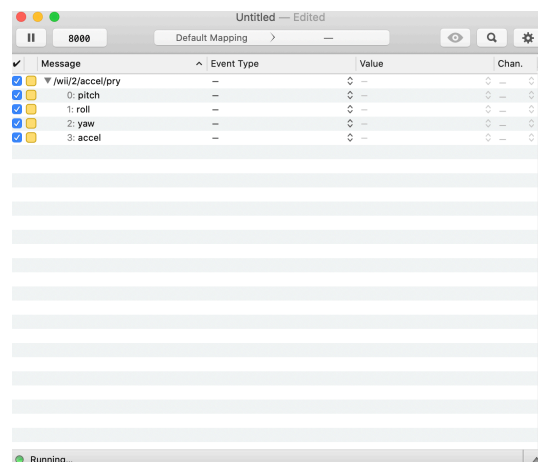
1. Activate bluetooth on your computer.
2. Under system preferences, open Bluetooth.
3. Remove the back cover of the Wii Remote and press the red sync button.
4. You'll see a new device appear with a series of cryptic letters numbers. Pair the device.

If you do not see the device, do the following:

1. Turn off your Bluetooth and turn it back on. Try again.
2. Try resetting the Bluetooth module. You can do this by going to the Bluetooth icon at the top of your computer screen and clicking it while holding Shift and Option. In the drop down menu, you'll see an option called "Debug." Mouse over to reset the bluetooth module. If that doesn't work, try removing other devices.

If the remote asks for a password, ignore it and try connecting. Once it's successfully connected, you'll notice the blue lights on your wii remote will continuously flash.

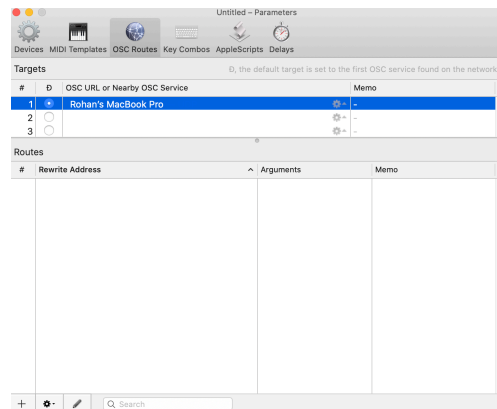
Once you've completed this stage, you're ready to begin transmitting data from the remote. Begin by opening OSCulator. As soon as you open OSCulator, you'll notice your remote will now display a singular LED and a menu till populate with items:



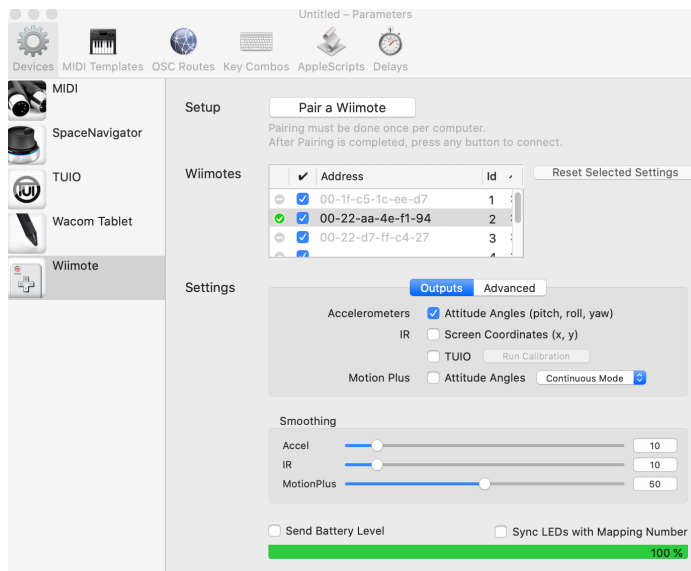
If you see this, your Wii Remote has successfully paired to OSCulator. In order for the data to be transmitted in real time to my computer, it must be packaged in an OSC message. OSCulator acts as a routing device for OSC data allows the Wii Remote data to transmit to any device via the internet protocol. Now on to the next stage:

CALIBRATING THE REMOTE:

Now that you're in OSCulator, you can prepare the necessary data to be transmitted. Begin by clicking the gear icon in the top right hand corner:



From here, click on Devices.

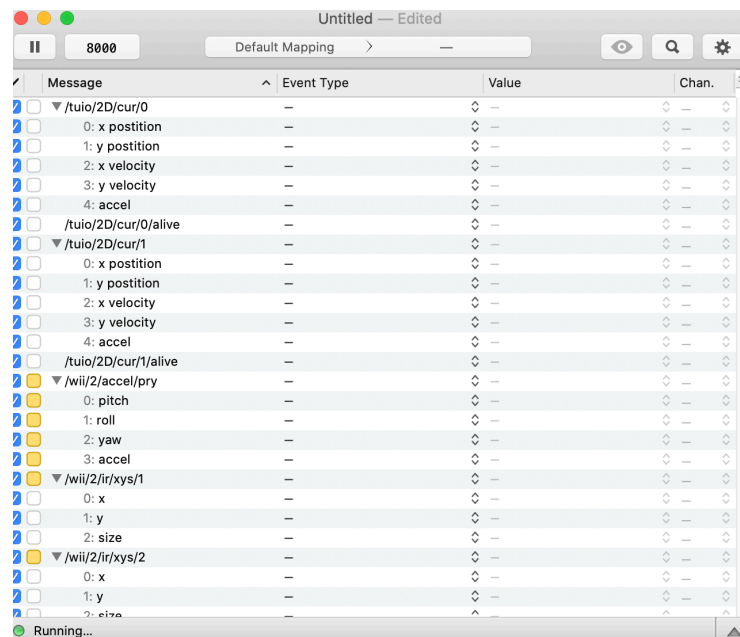


In Wiimotes, select your currently active Wiimote. Underneath, click on advanced and activate "Raw Bright Dots Coordinates" and increase the IR sensitivity to +2. Once you do that, return to outputs and check TUIO. While we won't be using TUIO, we can use the calibration to make sure the wii remote and TV remote are operating correctly. Toggle TUIO and click on "Run Calibration."

A screen with 4 targets in each corner should pop up, along with a button "Start Calibration." Click start calibration. Once you do this, take your TV remote and point it towards the Wii Remote (this can be from any distance). Press the switch on the TV Remote and keep it pointed to the Wii Remote till the target gets checked. Repeat the exact procedure for the remaining three targets. Again, this is mostly to ensure that both remotes are operating smoothly- if you're having difficulty with this, there may be an issue with your hardware.

SENDING THE OSC MESSAGE:

Close out of the settings window to return back to the main OSCulator window. You'll notice that it's more populated than before:



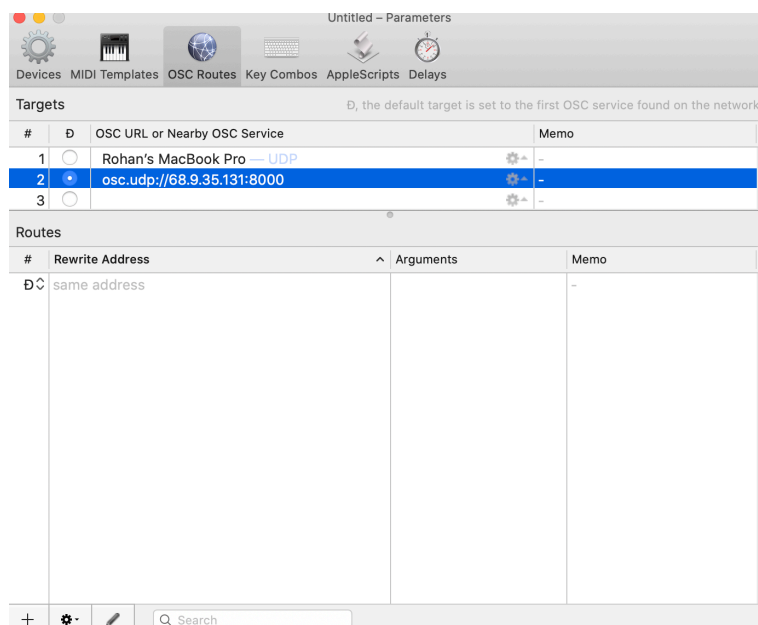
For this iteration of the project, we're only going to be concerned with the heading /wii/(number)/ir/xys/1. Please note that I use (number) here as you might not have the 2 appear in the directory name. We're going to be specifically using the two beneath that directory labeled X and Y.

In the event type column, on the same row as the X category, click and select OSC message. In the column over, for value, click on it and press New. From here you'll be taken back to the OSC routing menu we were in before.

You'll notice that menu is split into two sections, Targets and Routes. Under targets, you might notice the name of your own laptop. In the row beneath that, double click and write the following:

osc.udp://(my public IP address):8000

My public IP address might change (unlikely, but possible) so for that reason I'll disclose that to you when we do some test runs/rehearsals. The URL should look like this:



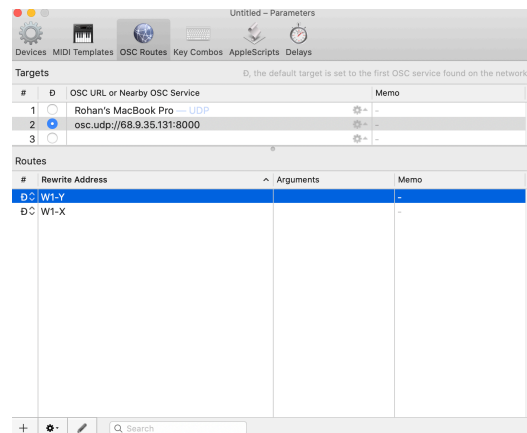
Once you've done this, double click on the greyed out text that reads "same address" in the Routes section. In the very top Rewrite Address section, type:

W(your part number)-X

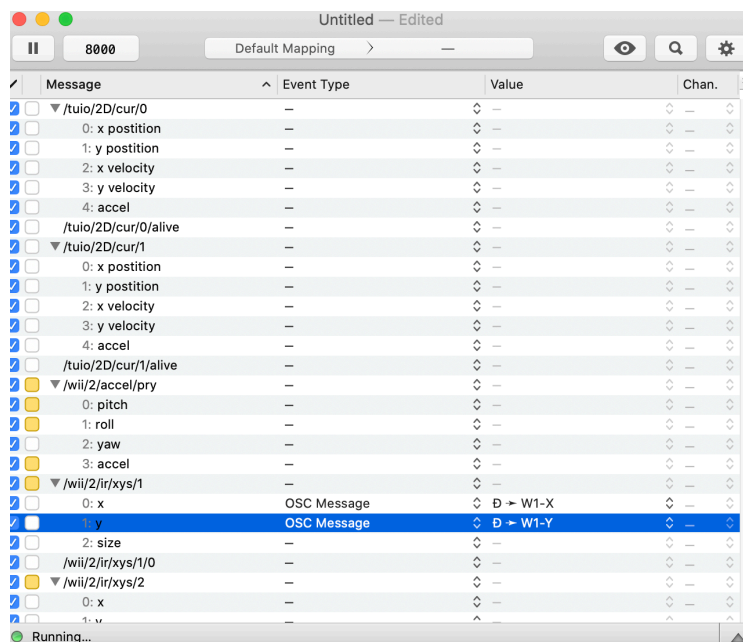
Close out and add another one by clicking the + icon in the bottom left corner. Type, in the same location:

W(your part number)-Y

Once you've done all that, your OSC Routes menu should look like this:



Return back to the main OSCulator page. For the X row of the /xys/1 directory, change the value to the X, for the Y, change to value to the Y routing address. It should look as follows:

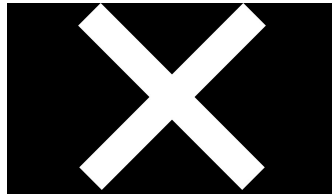


Your Wii remote is now transmitting X-Y coordinates of your Roku remote to my computer. To check, select the X row and the Y row and click on the eyeball icon in the top right of the window. Point your Roku remote, press a button, and move around. Notice how the graphs move as you move. Try standing further away- you'll notice you need to increase your range of motion to create the same movement of data. This is one way we'll be scaling the performance according to your environment.

NOTES ON THE SCORE:

The piece is performed by mimicking me as you perform. Each part can be found on a Google drive or by contacting me, as well as the full score (all four parts in split screen). I encourage you to take a look at the full score to get a sense how the gestures interact with other gestures.

Sometimes you may be moving but not triggering sound. To indicate whether your Roku remote is to be pressed, a large X will appear on the bottom of the screen like this:



SETTING UP THE SCORE PATCH:

In order to successfully operate this patch, you need to perform an internet routing technique called **port forwarding**. Because this varies extensively depending on the make and model of your router, best option is to set up a brief Zoom/phone call where we can talk through it. You can reach out to me at rchandermusic@gmail.com.