AUBURN UNIVERSITY

Samuel Ginn College of Engineering

3D Modeling of Detached Metal Whiskers

User Operations Manual

Version 1.16

Mech 4250 – Group 8

7/16/2024

Table of Contents

Table of Figures

Introduction

This manual contains information regarding the 3D Modeling of Detached Metal Whiskers Simulation, created by a team of Senior Mechanical Engineering students at Auburn University. This program is designed to utilize the Unity Game and Physics Engine to run simulations that analyze the odds of whisker bridging across circuit boards in various environments and settings. It is intended for use in reliability testing and predictive analysis in the field of electronics, offering a user-friendly interface to assess and mitigate the risks associated with whiskers on and around circuit boards.

The program allows users to customize their entire simulation, providing control over whisker dimensions and properties, spawning locations, external forces, and circuit board analysis, among other features. It aims to give users a realistic simulation of their own boards, which can be easily imported into the program. With highly customizable parameters, users can tailor any simulation to specific, real-world scenarios.

This user manual is designed to be a comprehensive guide to effectively utilizing the 3D Modeling of Detached Metal Whiskers Simulation. By following the instructions and insights provided, you will be able to maximize the program's potential, making informed decisions to enhance the reliability of your electronic designs. If this is the first time accessing and learning how to utilize this program, continue on to: **Getting Started.** If the program is already functioning and a faster, more condensed version of the instructions are needed, refer to **Quick Overview.**

Getting Started

Section 1: GitHub and Repository Access.

To gain access to the simulation, you will need to obtain it through GitHub. GitHub is a powerful version control and software management tool that enables developers to collaborate efficiently and manage their code effectively. By using GitHub, you can easily access the latest version of the simulation, ensuring that you are always up to date with the most recent changes and improvements. GitHub can also facilitate easy updates, allowing users to synchronize their local copies with the latest changes made by developers. This process eliminates the need for extensive redownloading. Additionally, GitHub's version control system tracks all modifications made to the code, allowing developers to provide a comprehensive history of changes to the users as updates come out. This makes it easy to revert to previous versions if needed and understand the evolution of the project over time. The utilization of GitHub for this simulation was vital to allow users to easily access and manage the program files, along with staying updated as future teams develop the program further. The following sections explain how to gain access to this program

There are two main methods that can be used to download and gain access to this simulation. Both methods require a GitHub account. If you do not have an account already, it is vital to create one using the following link and instructions listed on the website:

https://github.com/

Once an account has been made, you may follow one of the two methods to download or gain access to the repository.

Method 1: Using GitHub Desktop (Recommended):

- **a.** If you do not have GitHub Desktop already, refer to the following link to download:

https://github.com/apps/desktop

- **b.** Once downloaded, open the app and log in.
- **c.** From there, look to the top left of the app for the File tab. Click this, and find the option: Clone Repository

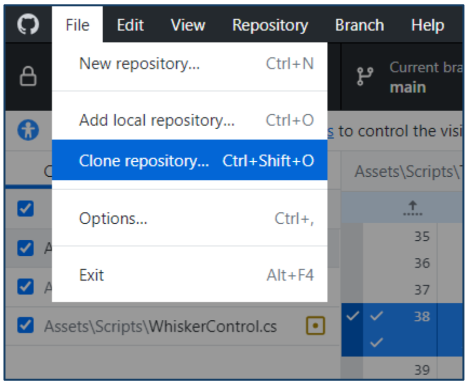
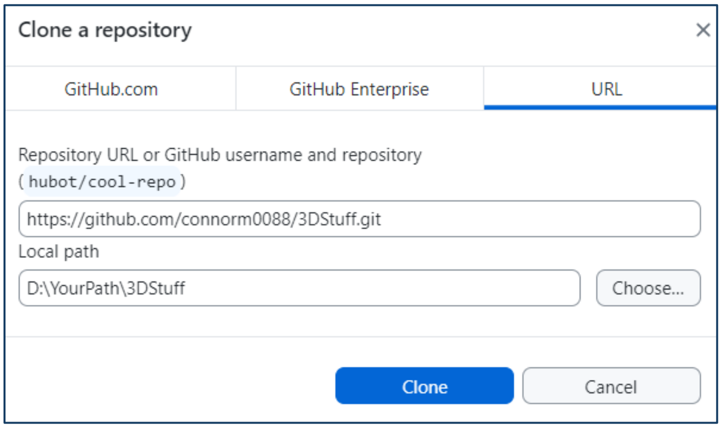


Figure 1: Cloning Repository

- **d.** This will open a new screen. Click on the tab titled URL, and enter in the following URL to the input field:

https://github.com/connorm0088/3DStuff.git

- **e.** Define a directory for the repository to be cloned into. Then, click Clone.



Repository URL or GitHub username and repository (`hubot/cool-repo`)

https://github.com/connorm0088/3DStuff.git

Local path

D:\YourPath\3DStuff

Figure 2: GitHub URL

The repository containing the simulation is now saved to your device, and you may skip to Unity Startup

Method 2: GitHub Website:

This method is more prone to issues, as it revolves around downloading the files instead of cloning them directly onto your device, which takes up more space and is much slower. This method is not recommended but can be used if GitHub Desktop is not accessible.

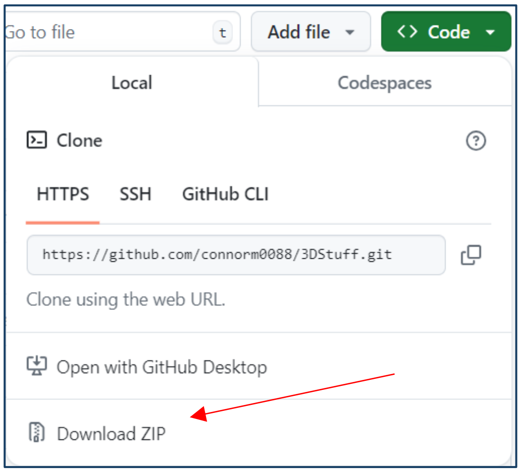
a. Assuming a GitHub account has been made, navigate to the following link to access the online version of the simulation repository:

https://github.com/connorm0088/3DStuff

b. Click the green Code button and click Download Zip. This will begin the installation of the program files.

c. Once fully downloaded, extract the files to a directory of your choice on your device.

The repository containing the simulation is now saved to your device, and you may continue on to Unity Startup.



https://github.com/connorm0088/3DStuff.git

Clone using the web URL.

Figure 3: GitHub Repository Webpage

Unity Startup

Section 1: Downloading Unity.

As stated previously, this program runs off of the Unity Game Engine, so it is vital to have the Unity Editor and Unity Hub installed for this program to work as intended. The both the Editor and the Hub can be downloaded by following the official tutorial found on the link below. This program was developed in version 2022.3 of the Editor. It is highly recommended to get this same version for running the simulation as other versions might have different features and layouts, making it more complicated to follow the instructions.

Unity Hub and Editor Download

Section 2: Unity Hub

Once completed, you will have access to the Unity Editor and the Unity Hub. The Hub is a Unity program used to control and manage all of the projects users may have saved. This will be the main way to access and run the simulation.

To access the simulation, the repository saved in **Getting Started, Section 1** must be added to your Unity Hub. Refer to the following instructions to complete this process:

a. Open Unity Hub.

b. In the Projects tab, click Add. This will bring up your file explorer. *The image for your reference already has projects listed, however, your Unity Hub will most likely be empty.

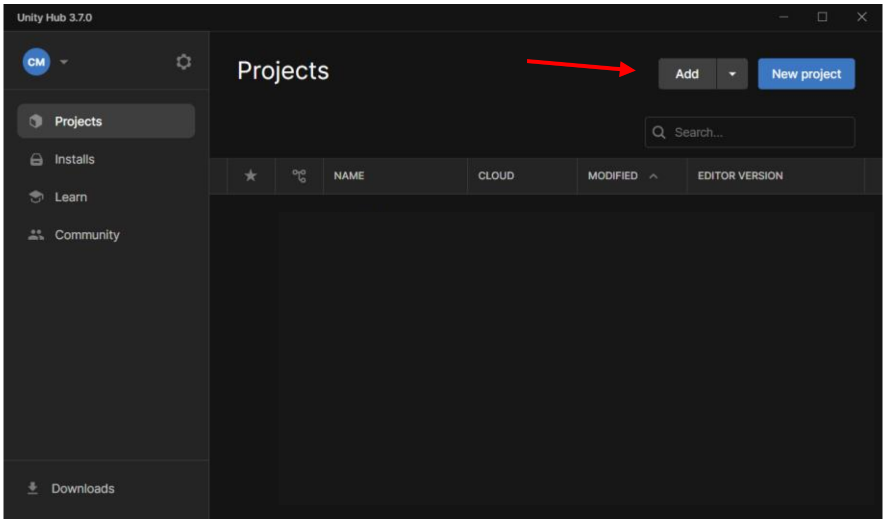
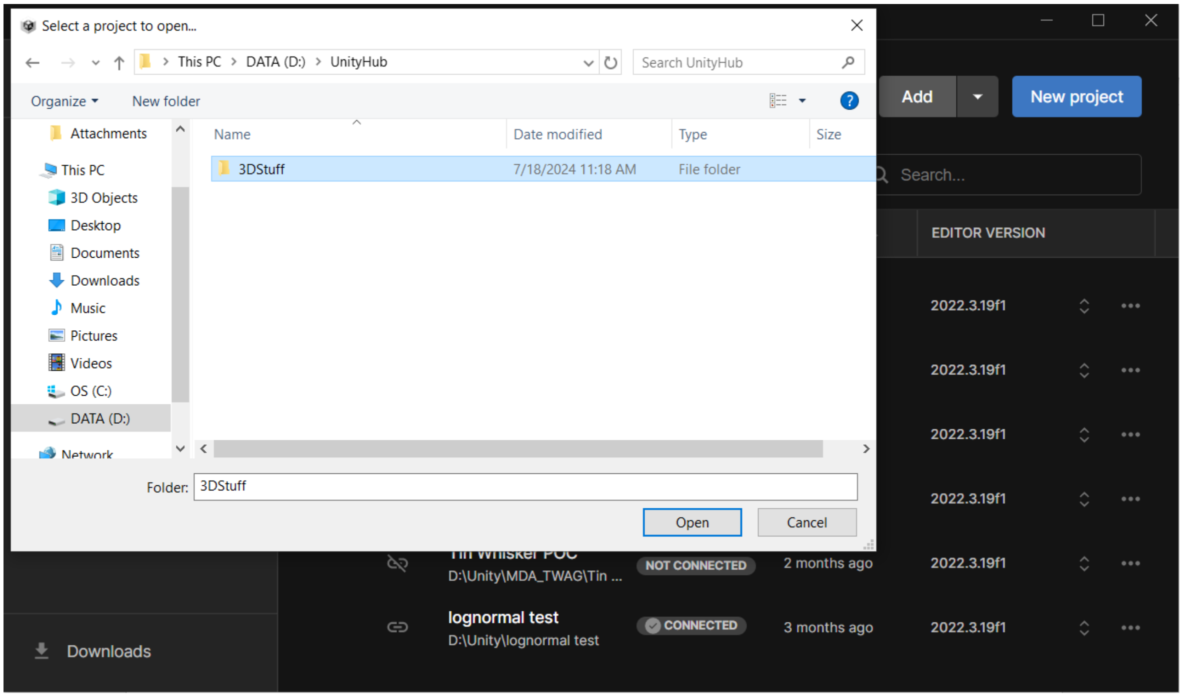


Figure 4: Unity Hub Interface

c. Navigate to the directory where the repository was saved. Click the repository and click Open.



Name	Date modified	Type	Size
3DStuff	7/18/2024 11:18 AM	File folder	

Folder: 3DStuff

			2022.3.19f1
			2022.3.19f1
			2022.3.19f1
			2022.3.19f1
Tin Whisker POC D:\Unity\MDA_TWAG\Tin ...	NOT CONNECTED	2 months ago	2022.3.19f1
lognormal test D:\Unity\lognormal test	CONNECTED	3 months ago	2022.3.19f1

Figure 5: Selecting the Repository

The project should now be available in the Unity Hub. You may move to Section 3: Unity Editor. *Note, repository name may be different from the one shown in Figure 5.

Section 3: Unity Editor

Now that the simulation has been cloned from GitHub and added to your Unity Hub, you may now open the simulation. Clicking on the project will open it in the Editor. Please allow a few moments for this to complete. When the project is opened, you should see Unity's interface in the Editor. There are four main sections of the Editor: Hierarchy, Scene/Game window, Inspector, and the Project/Console Tabs. If this is the first time the Editor is being opened, it should be blank with nothing in the main sections. As a brief overview, the following table describes what each section handles when using the Editor:

Editor Layout

Hierarchy	On the left side of the Editor, the hierarchy lists all the Game Objects in the current scene in a hierarchical order.
Scene/Game Window	Scene: Interactive view into the simulation where objects can be manipulated. Game: Preview of sim. before being ran.
Inspector	On the Right side of the Editor, it is filled whenever a Game Object is selected, and provides important information about the object.
Project/Console Tabs	At the bottom left side of the Editor, these tabs contain project folders and any error messages being printed from the console

It is not necessary to be fully familiar with the Editor and its features to run this program. However, if more information is desired, the official Unity website has countless tutorials and help pages to answer any questions. Figure 6 shows a fresh Editor when opened:

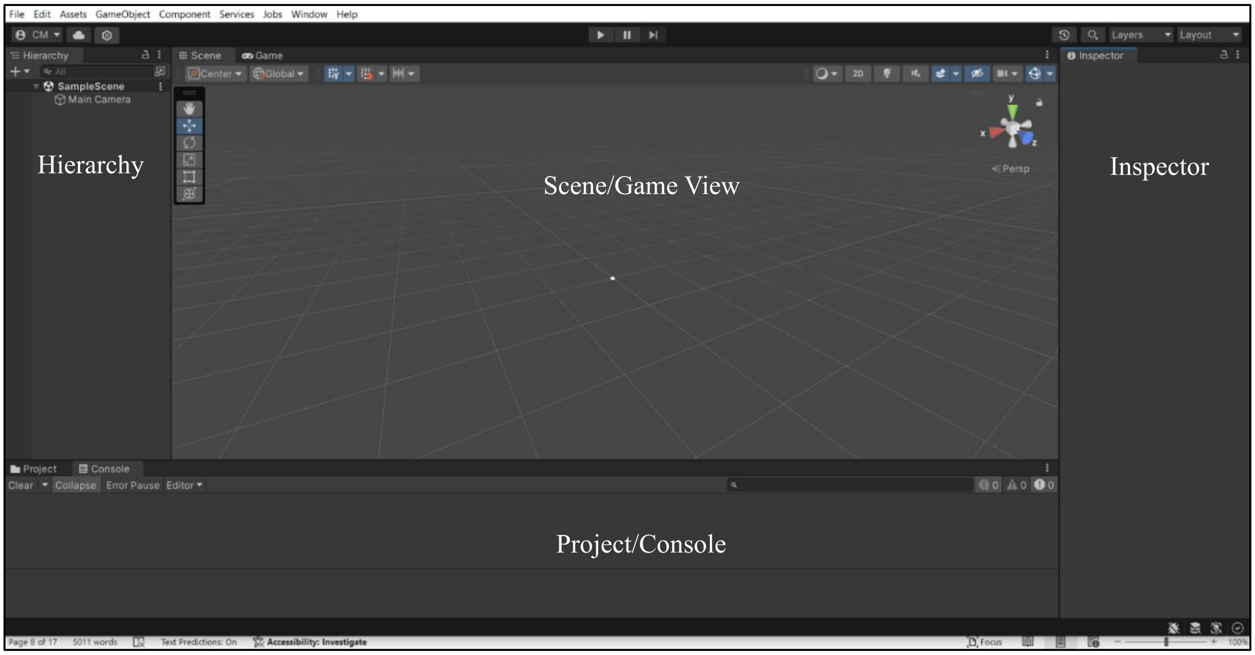


Figure 6: Empty Unity Editor

Section 4: Importing the Simulation

Once in the Editor, you may now complete the process of accessing the simulation. Refer to the following instructions to finalize this process:

a. To fully import the simulation into the fresh Unity Editor, click on Project in the Project/Console Tabs. On the left side of the tab, there should be a folder icon labeled Assets. Click the dropdown next to this folder.

b. Once opened, there will be a file labeled Scenes. Click into this folder.

c. Once inside, there will be one object: a Unity logo with the name SampleScene. Double click this object to fully import the Simulation into your Editor.

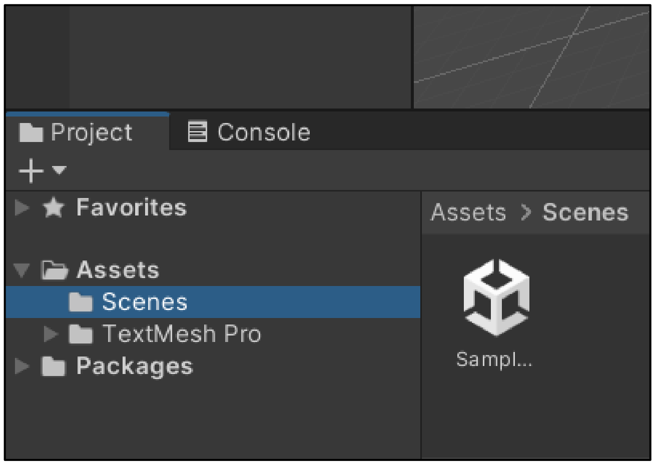


Figure 7: Sample Scene

Once these steps have been completed, the simulation is now being hosted by the Editor. To verify this was done correctly, refer to Figure 8 to compare your Editor. These steps will not have to be repeated as the simulation is now saved to your Unity Hub and can be reopened at any time.

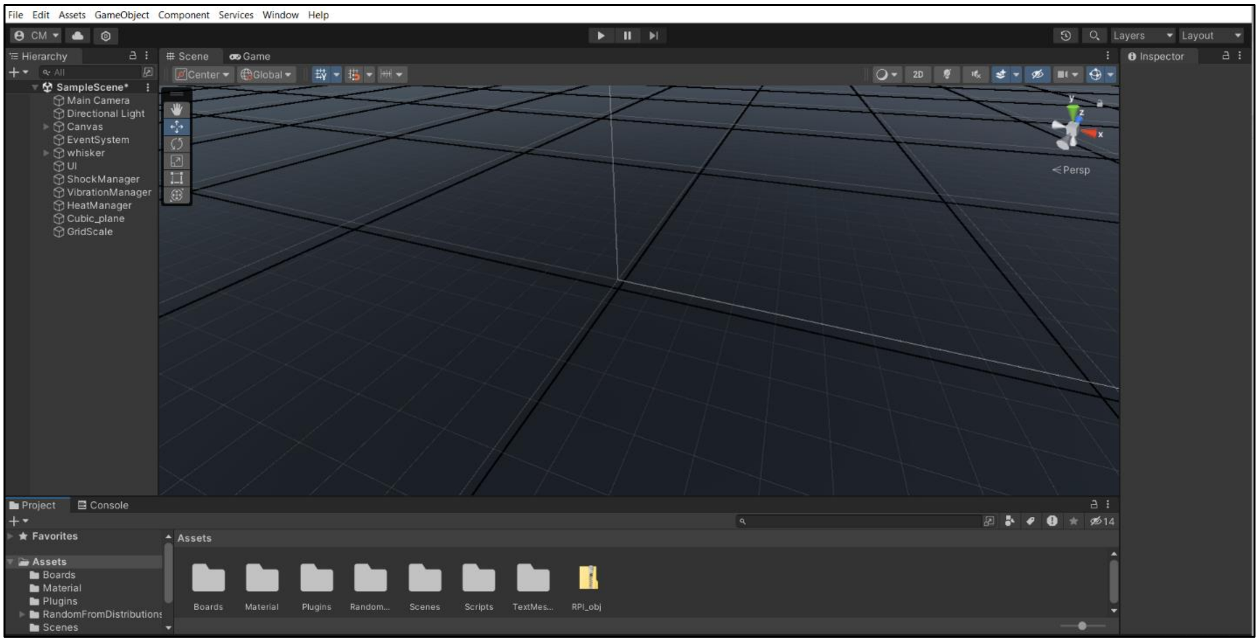


Figure 8: Simulation Editor Screen

You may now proceed to **Setting up your Simulation.**

Setting up your Simulation:

Section 1: Loading in your Desired Board

To get the most accurate results for your desired board, it will need to be imported into Unity as a .obj file. Other file types can be imported into Unity successfully, but some manipulation of the components may be required to get them to function properly. Many programs such as Altium, Blendr, Autodesk Eagle, and others have the capabilities to produce .obj files while maintaining the important contours and components of your circuit board.

a. Once you have your desired PCB in a compatible file type, open the folder containing the 3D object. Next you will need to access the Boards folder that came with the simulation. This can be done through the Unity Editor, under the Project tab. Click to open the Assets folder, and find a new folder titled Boards, then click into the folder. Once here, drag and drop your board into the folder. The board will now be present in the Project tab. ** Disregard preexisting boards in Figure 10. These will not be present in your project.

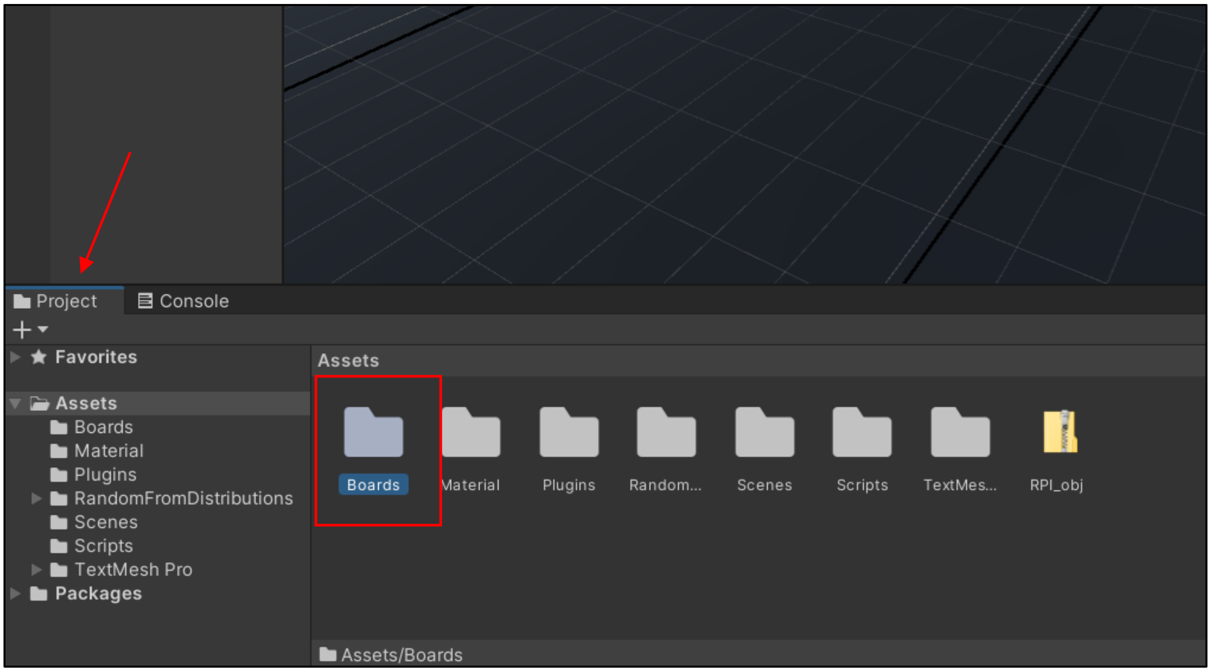


Figure 9: Accessing Board Folder

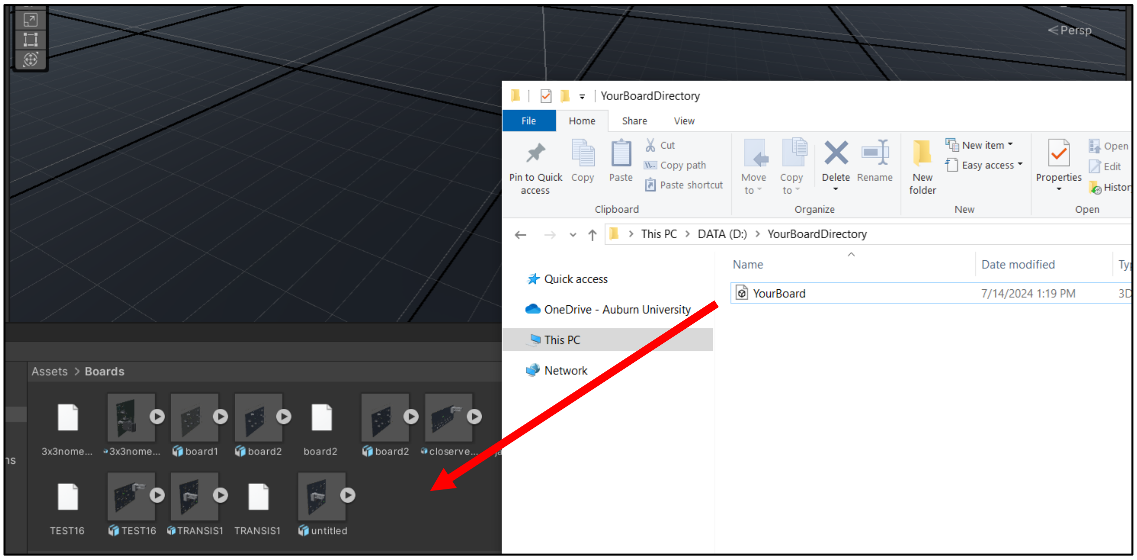


Figure 10: Importing your PCB

b. Click on your board and drag it into the Scene View. It does not matter where it is placed in the Scene, as through the following steps and the background code, it will be placed for you. Figure 11 demonstrates this task. *Disregard excess boards shown in figure.

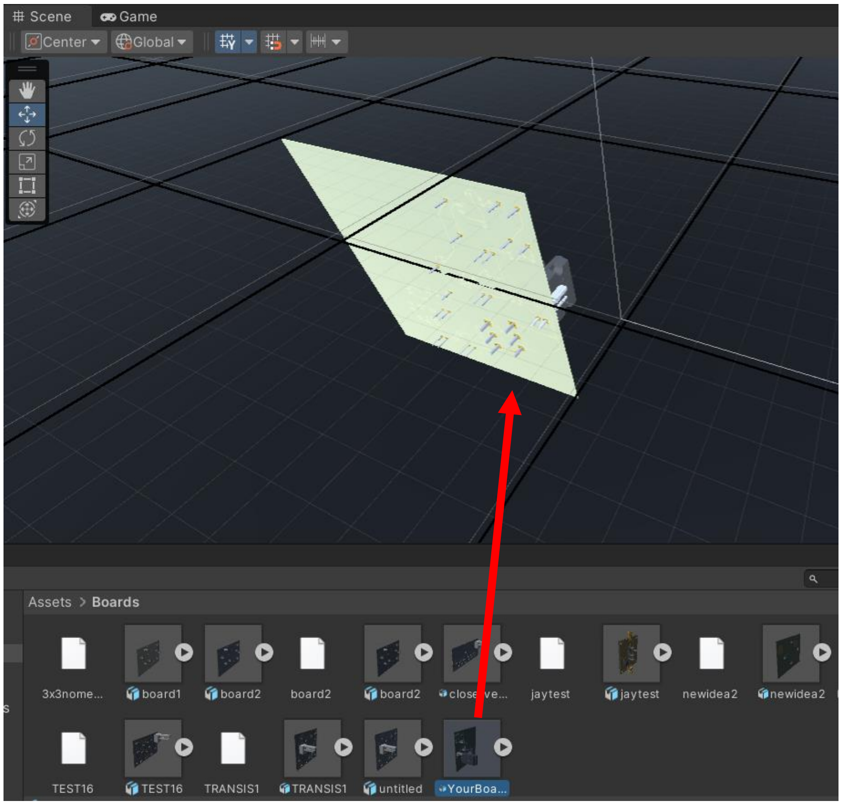
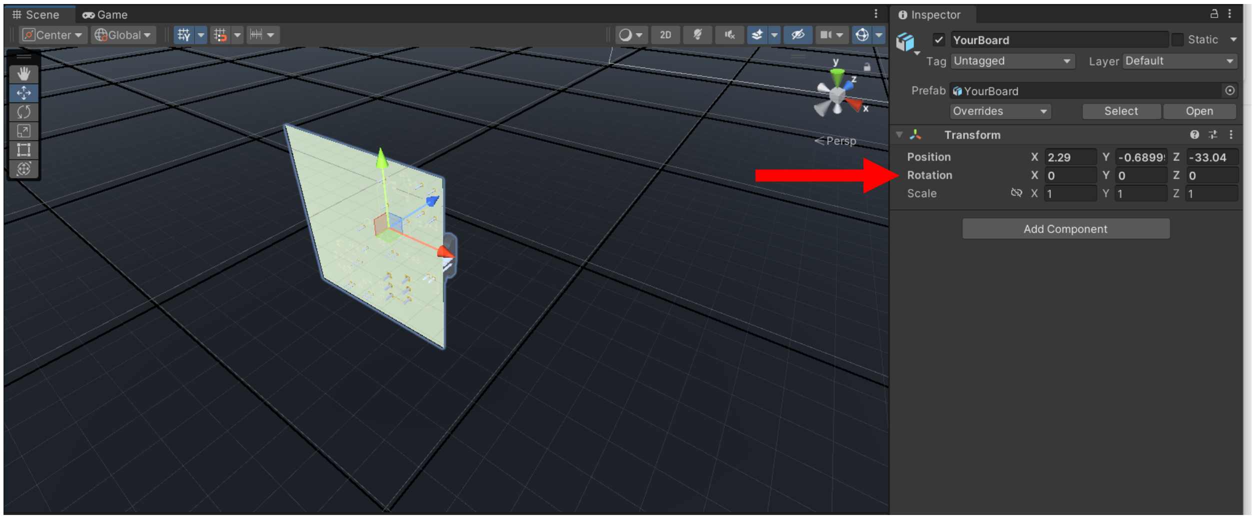


Figure 11: Dragging in the PCB

c. Correct the Orientation of your board in the case it does not import correctly. This can be done by selecting the board, and then looking in the Inspector tab. There should be a subsection called Transform, Rotation, where you can edit the rotation of the board to orient it properly. Example: typing in 90 under the X axis rotation will rotate the board 90 degrees in the positive X direction. Once this is done, the board should now be ready for the setup.



YourBoard
Tag Untagged Layer Default
Prefab YourBoard

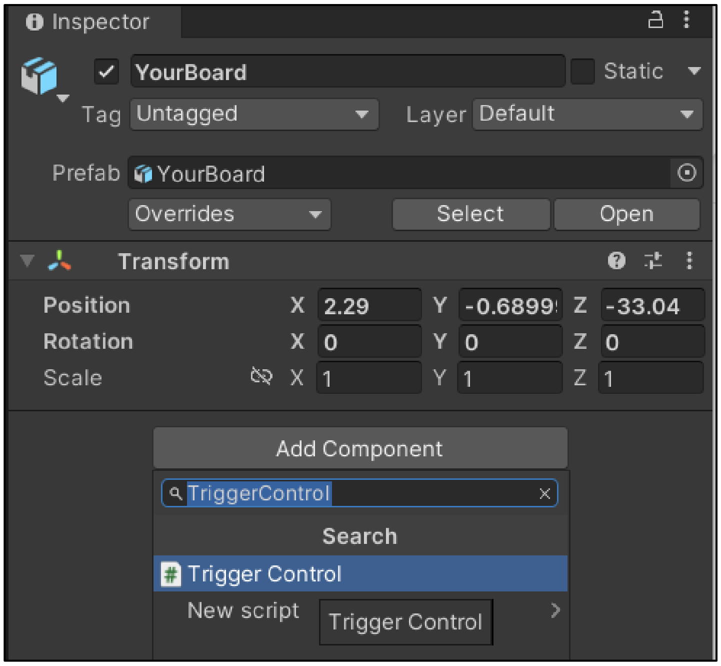
Position	X	2.29	Y	-0.6899	Z	-33.04
Rotation	X	0	Y	0	Z	0
Scale	X	1	Y	1	Z	1

Figure 12: Orienting your Board

Section 2: Preparing your board for Simulation

A few easy steps are required in order for the simulation to run as needed. These involve using the unity editor to attach certain scripts/objects to the board in order to effectively run the simulation.

a. If the previous section was completed properly, the board should now be in the Scene. Select your board and look to the Inspector. There will be a button called Add Component. Clicking this will bring up a search bar. Type in the bar: TriggerControl, and an icon that looks like a paper with a green pound symbol with the matching name should show up. Click this item. The script will be added as a component, and a new field will be added to the Inspector for your board.



YourBoard
Tag Untagged Layer Default
Prefab YourBoard

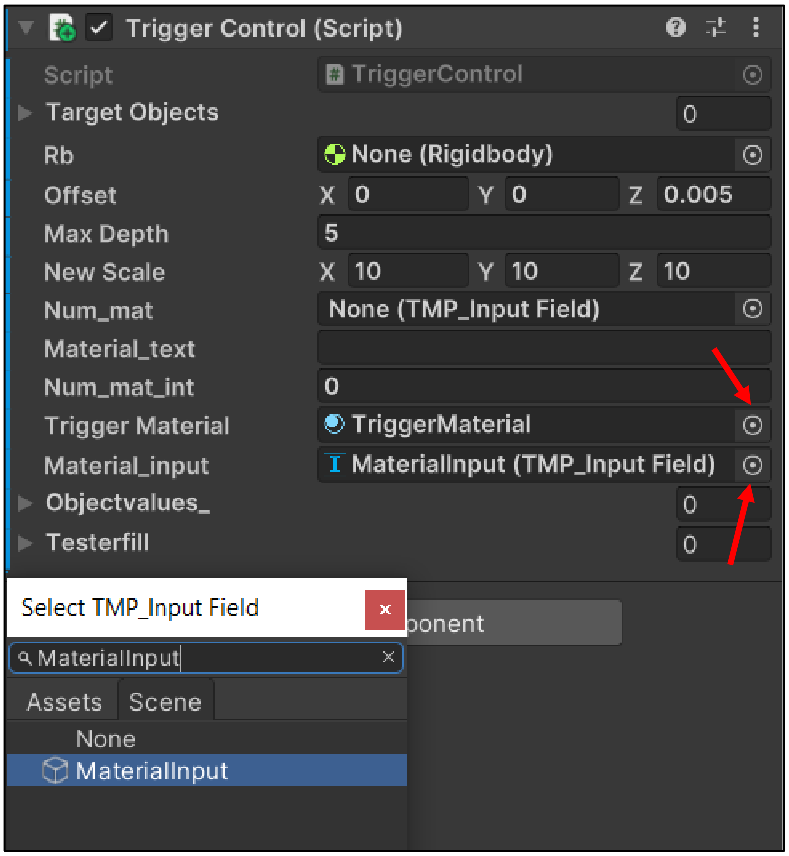
Position	X	2.29	Y	-0.6899	Z	-33.04
Rotation	X	0	Y	0	Z	0
Scale	X	1	Y	1	Z	1

TriggerControl
Trigger Control
Trigger Control

Figure 13: Adding Components

b. Now, you should see a few new fields in the Inspector under the section Trigger Control. There is an empty field titled Trigger Material. Click the small circle to the right of this field, which will bring up a text input field. Type "TriggerMaterial" and click the item that comes up.

c. Finally, directly under the Trigger Material field should be one titled Material Input. Similar to the last step, click the small circle to the right of the field, but instead type type "MaterialInput", and click the item that comes up. Leave all other fields in this section untouched. Your board is now ready to be run through the simulation. Figure 14 shows what the fields should look like once completed properly.



Trigger Control (Script)

Script	TriggerControl
Target Objects	0
Rb	None (Rigidbody)
Offset	X 0 Y 0 Z 0.005
Max Depth	5
New Scale	X 10 Y 10 Z 10
Num_mat	None (TMP_Input Field)
Material_text	
Num_mat_int	0
Trigger Material	TriggerMaterial
Material_input	MaterialInput (TMP_Input Field)
Objectvalues_	0
Testerfill	0

Select TMP_Input Field

MaterialInput

Figure 14: Finalizing the Board

The PCB of choice can now be correctly analyzed in the simulation. Keep in mind that these steps will need to be repeated as more boards are imported into the simulation but will not need to be repeated for ones already set up. If there are multiple boards in the Scene and only one is required for testing, you can turn off the others by selecting them and looking at the top left of the Inspector window. There, you will see a small check mark indicating that the board is 'on.' Clicking this check mark will turn the board off, allowing Unity to retain the board's setup but remove it from the Scene until it is needed again.

Running the Simulation (Walkthrough):

Section 1: Camera Controls

Now that the boards are set up, the simulation can be run. To do this, look to the top of your Unity Editor. Click the Play Button to start the program.

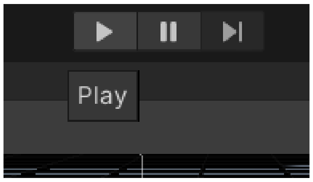
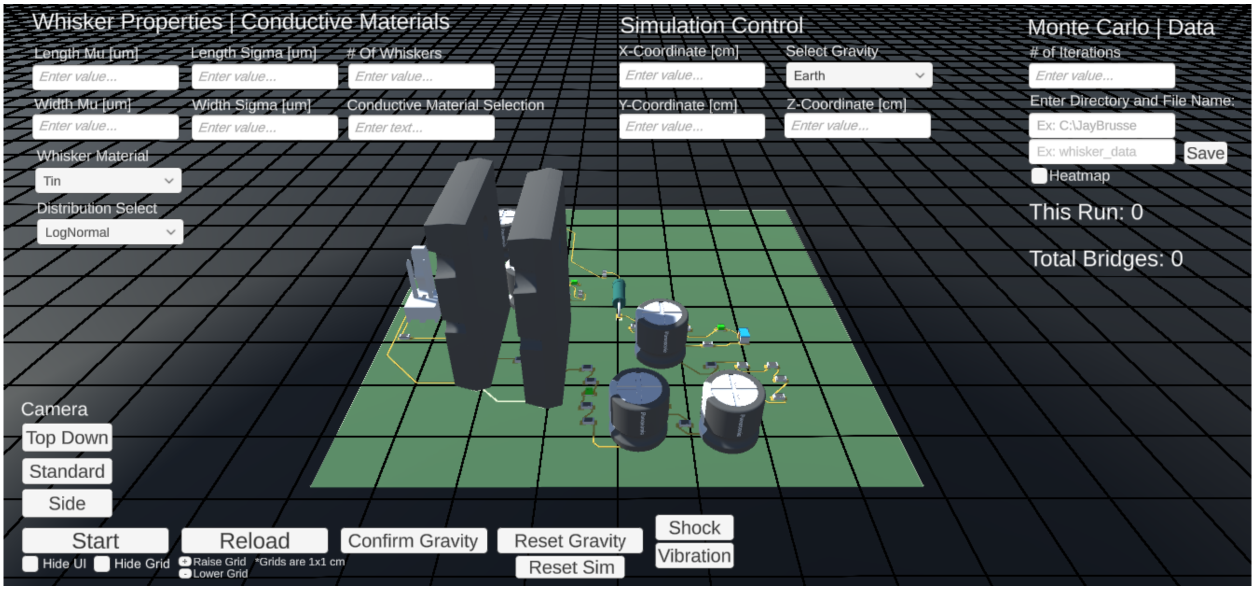


Figure 15: Play Button

a. When in Play Mode, users can navigate the camera around the simulation to get the best views of their PCB/whisker bridges. To rotate the camera, hold the left mouse button and move the mouse in any direction. Additionally, use the W, A, S, and D keys to move the camera forward, left, right, and backward from the direction the camera is facing. The E, and Q, keys can be used to move the camera vertically up or down, and clicking the right mouse button will reset the camera orientation to level with the board. On the bottom left of the Interface, there is a section titled ‘Camera’, which has three buttons that when pressed, set the camera to preset conditions. Standard is a front facing ariel view, Top Down is a birds eye view from directly over the board, and Side is a side facing view from the left side of the PCB.

Figure 16 shows what the simulation would look like with a PCB imported in. There are four main areas of the interface that will be covered in the following sections in detail. As a brief overview: each section handles a different part of the simulation and allows users to either give inputs to fully customize the simulation and save any data necessary or run and manage the simulation. Each simulation comes with a scale grid, which are the black lines laid across the board and the entire Scene. These grids are all 1x1 cm squares.



Whisker Material: Tin
Distribution Select: LogNormal
Select Gravity: Earth
This Run: 0
Total Bridges: 0

Figure 16: Simulation Interface

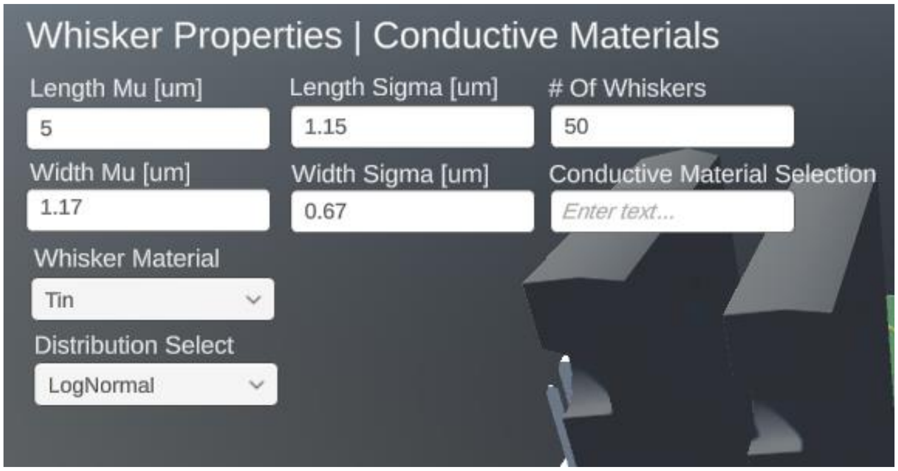
Section 2: Whisker Properties | Conductive Materials

This is the section of the Interface that handles all of the user inputs required to generate whiskers into the Scene, along with the input field that allows users to tell the program what materials on their board are to be seen as conductive. Below is a table giving a description of what each part of the interface does:

Whisker Properties Inputs

Length Mu [µm]	Takes in the value and uses it as the location parameter or the mean value for the Lengths of the whiskers depending on if the distribution is Lognormal or Normal.
Length Sigma [µm]	Takes in the value and uses it as the scale parameter or the standard deviation for the Lengths of the whiskers depending on if the distribution is Lognormal or Normal.
Width Mu [µm]	Takes in the value and uses it as the location parameter or the mean value for the diameters of the whiskers depending on if the distribution is Lognormal or Normal.
Width Sigma [µm]	Takes in the value and uses it as the scale parameter or the standard deviation for the diameters of the whiskers depending on if the distribution is Lognormal or Normal.
# of Whiskers	Sets how many whiskers are to be generated.
Whisker Material	A dropdown list that allows users to select from Tin, Zinc, or Cadmium, changing the base properties of the whiskers.
Distribution Selection	A dropdown list that sets what distribution type is to be used for the generation of the whiskers. Defaults to Lognormal.
Conductive Material Selection*	A text field that when material names are entered in, the program searches the components of the board for the specific material listed, and if it finds a match, it labels that object as Conductive.

a. You can now begin to enter in values for your Mu and Sigma inputs in micrometers and choose your number of whiskers, along with selecting any desired options from the dropdown lists. Verify that these parameters are in line with the distribution being, as the whiskers may not spawn in or be visible depending on the size of the inputs and the distribution being selected. The whisker material may now be selected from the dropdown as well.


Whisker Properties | Conductive Materials

Length Mu [um]: 5

Length Sigma [um]: 1.15

Of Whiskers: 50

Width Mu [um]: 1.17

Width Sigma [um]: 0.67

Conductive Material Selection: Enter text...

Whisker Material: Tin

Distribution Select: LogNormal

Figure 17: Example Inputs

b. Finally, the conductive material can be set. This will be done by entering the name of the material into the input field, and then clearing the field before entering in more conductive materials if there are multiple. This can handle any number of conductive materials, but once the conductive materials are set, they cannot be undone except by exiting Play mode, and refreshing the simulation.

*NOTE: Many outside programs used for creating models of circuit boards do not handle the materials of the components very well, regardless of which program is being used. Often times, instead of a material being labeled correctly (e.g. copper), it will be labeled as mat_12, or a variation of that format, which will then be how it appears in Unity. It is important to scan through your board after importing and see what material names have been assigned to properly use this program. These could change depending on the program the board gets exported from. To view these materials, open the Boards folder in the Project tab, under Assets. Click the small arrow next to your board. This will open up a list of all the components on the board and any materials that were imported alongside.

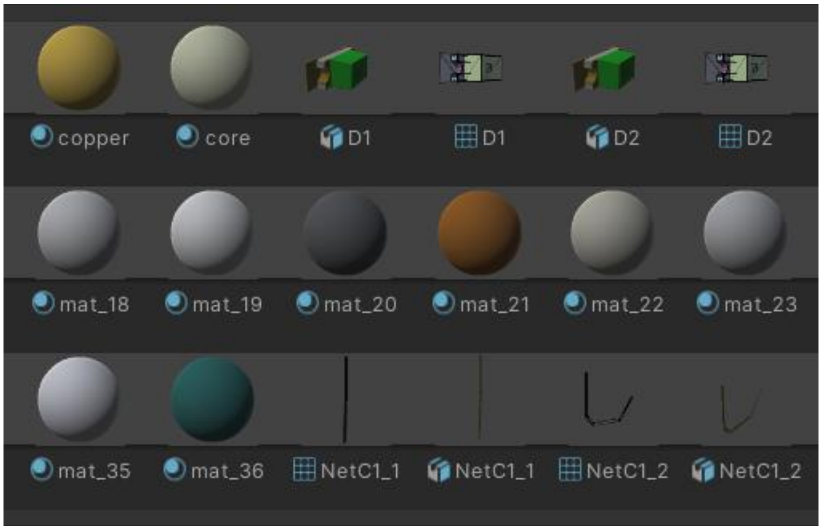


Figure 18: Materials Imported from Altium

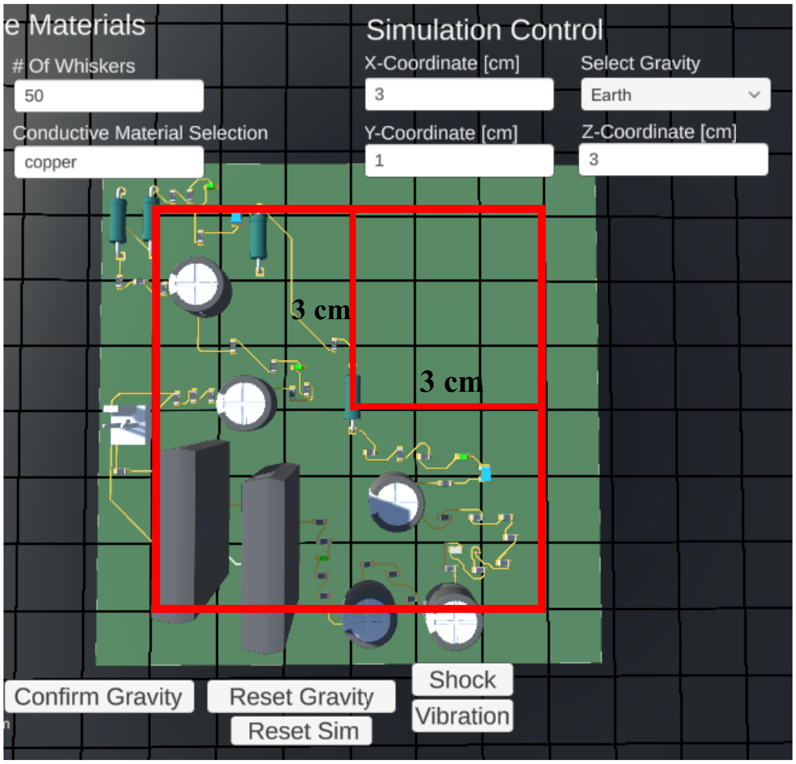
Section 3: Simulation Control

This section of the Interface handles where the whiskers will spawn in from, along with what gravity scale will be applied upon confirming the gravity later in the walkthrough. Below is a table showing what each input field handles.

Simulation Control Inputs

X – Coordinate [cm]	Set the bounds in the X direction where the whiskers can spawn in from.
Y – Coordinate [cm]	Set the bounds in the Y direction where the whiskers can spawn in from.
Z – Coordinate [cm]	Set the bounds in the Z direction where the whiskers can spawn in from.
Select Gravity	A dropdown list that allows users to select what gravity scale is being applied with a selection of Earth, Moon, or Mars.

a. The spawning locations can now be set using the coordinate system. This system works by effectively creating a cube that the whiskers can spawn into. When a value is entered into the coordinate field, it sets the boundaries for spawning objects in both directions. For example, if you type 3 into the coordinate field for the x direction, the objects will be able to spawn anywhere from -3 to +3 on the x-axis, providing a total span of 6 centimeters. Similarly, the same applies to the y and z directions if values are entered for those fields. This allows PCBs of any size to be effectively covered by whiskers. Figure 19 demonstrates this:



e Materials

Of Whiskers: 50

Conductive Material Selection: copper

Simulation Control

X-Coordinate [cm]: 3

Select Gravity: Earth

Y-Coordinate [cm]: 1

Z-Coordinate [cm]: 3

Figure 19: Whisker Spawning Location

It can be seen from the red box where the whiskers would be able to spawn in from given an input of 3 cm for the X and Z axis, as when the simulation is run, the board is instantly placed at (0, 0, 0) in the Scene.

b. Next, the gravity scale can be selected. Clicking the dropdown will show the list of options: Earth, Moon, or Mars. Clicking one of the options will set the gravitational acceleration to the respective value.

Section 4: Monte Carlo | Data Storage

Once the inputs covered in sections 2-3 have been filled in, the final inputs can be set. This section allows the user to run the simulation through multiple iterations, automatically, along with providing a directory and file name for the data to be stored in order to save information from the simulation. This section also has the toggle button used to turn on the Heatmap, which will be discussed later on in the walkthrough. The table listed below describes what this section of the interface provides:

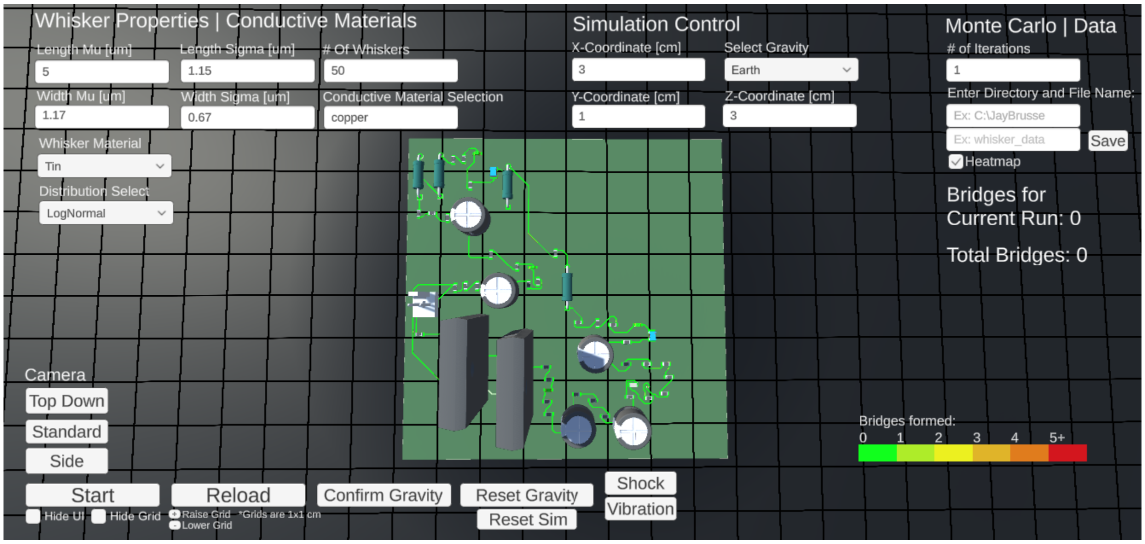
Monte Carlo | Data Storage Inputs

# of Iterations	Tells the simulation how many iterations to run before coming to a stop.
Directory	Input field for users to place their specific/desired location for their data on their device.
File Name	Input field for users to set the file name of their .csv sheet, which will contain the simulation data.
Save Button	Saves the Directory and File Name inputs into the code.
Heatmap Toggle	Toggles the Heatmap on or off.
Bridges for Current Run	Displays number of bridges detected for the current iteration.
Total Bridges	Displays the total number of bridges detected across all iterations.

a. The number of iterations for the simulation to run through can be set. It is recommended to enter in “1” for this field until the whiskers have their desired dimensions and spawning area.

b. Enter in a directory on your personal device and a file name for the csv file to give the simulation a location to store the data into. Press Save once ready. If an incorrect directory or file name is entered, reenter the correct name and press Save again.

c. Optionally, click the Heatmap Toggle to turn on the Heatmap. The components containing the conductive materials you have given in **Section 2-b** will be highlighted in green, and a scale of the gradient will appear on the screen, ranging from 0-5+ whiskers, moving from light green to dark red. More details about this feature can be found in **Running the Simulation: Section 6.**



Whisker Properties | Conductive Materials

Length Mu [um]: 5
Length Sigma [um]: 1.15
Of Whiskers: 50
Width Mu [um]: 1.17
Width Sigma [um]: 0.67
Conductive Material Selection: copper
Whisker Material: Tin
Distribution Select: LogNormal

Simulation Control

X-Coordinate [cm]: 3
Select Gravity: Earth
Y-Coordinate [cm]: 1
Z-Coordinate [cm]: 3

Monte Carlo | Data

# of Iterations: 1
Enter Directory and File Name:
Ex: C:\JayBrusse
Ex: whisker_data
☑ Heatmap

Bridges for Current Run: 0

Total Bridges: 0

Figure 20: Heatmap Toggled

Section 5: Simulation Management

This section of the Interface allows users to spawn in whiskers, reload them to better fit to the PCB, begin and stop the simulation, along with adding external forces onto the board. The external forces are covered in detail in **Section 7: External Forces**. Below is the final table describing what each Interface object handles in this area:

Simulation Management Controls

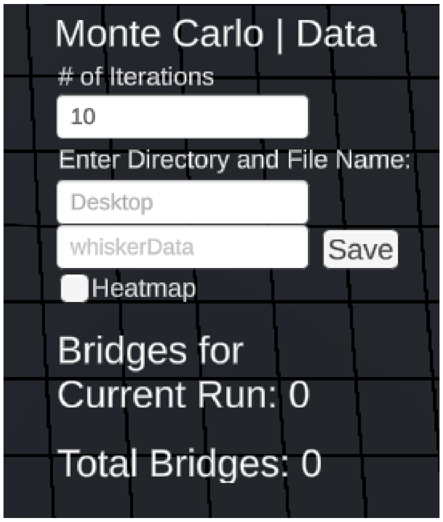
Start	Initially spawns in whiskers based off all of the previously described inputs.
Reload	Allows users to load fresh whiskers into the Scene after making a change to either their dimensions, number in the Scene, or coordinate location over the board.
Confirm Gravity	Confirms the gravity and officially starts the simulation. Pressing this will run the simulation through all the iterations desired.
Reset Gravity	Turns off the gravity, and the iteration counter, which allows the whiskers or data storage to be changed to different parameters.
Reset Sim	Clears and stops the iteration counter, and clears bridges formed - total and per run.
Shock	Opens the inputs for the external force: Mechanical shock or impact.
Vibration	Opens the inputs for the external force: Physical vibration.
Hide UI	Toggle that hides the user inputs.
Hide Grid/Raise/Lower Grid	Series of a toggle and buttons that allow the user to turn off or on the scale grid, along with raise or lower it to better fit their PCB.

a. Once the previous sections and steps are completed, all inputs should be filled in and any selections from the dropdown lists should have been made. The Start button should now be pressed to begin the simulation. This will spawn in the whiskers based off the parameters chosen.

b. Next, you should verify that the whiskers are spawning in with the desired dimensions over the appropriate areas. If any parameter must be changed, they can be altered and pressing the Reload button will spawn new whiskers into the scene based off of the new parameters. Repeat this step until the whiskers are generating over the entire board and are the appropriate size for your simulation needs.

c. If the whiskers are correctly sized and generating over your PCB, confirm that the directory and file path are as desired, along with the number of iterations for the simulation to run through. Figure 21 demonstrates these inputs being filled.

d. Finally, press the Confirm Gravity button to confirm all settings, and fully begin the simulation. The whiskers should now be dropping towards the board, and every 5 seconds, they will be cleared along with generating a new set with the same parameters you have entered in. The whiskers creating bridges will be highlighted red and will be tracked, while the number of bridges being made will be printed to the screen on the right. The data from the simulation will be stored into the file name you have provided. Figure 22 provides a look at what the simulation looks like while running, while Figure 23 shows a bridged whisker being highlighted and compared to a non-bridged whisker. *NOTE the whiskers used in Figure 22 were generated using a Normal Distribution with extremely exaggerated parameters for demonstration purposes.



Monte Carlo | Data

# of Iterations: 10

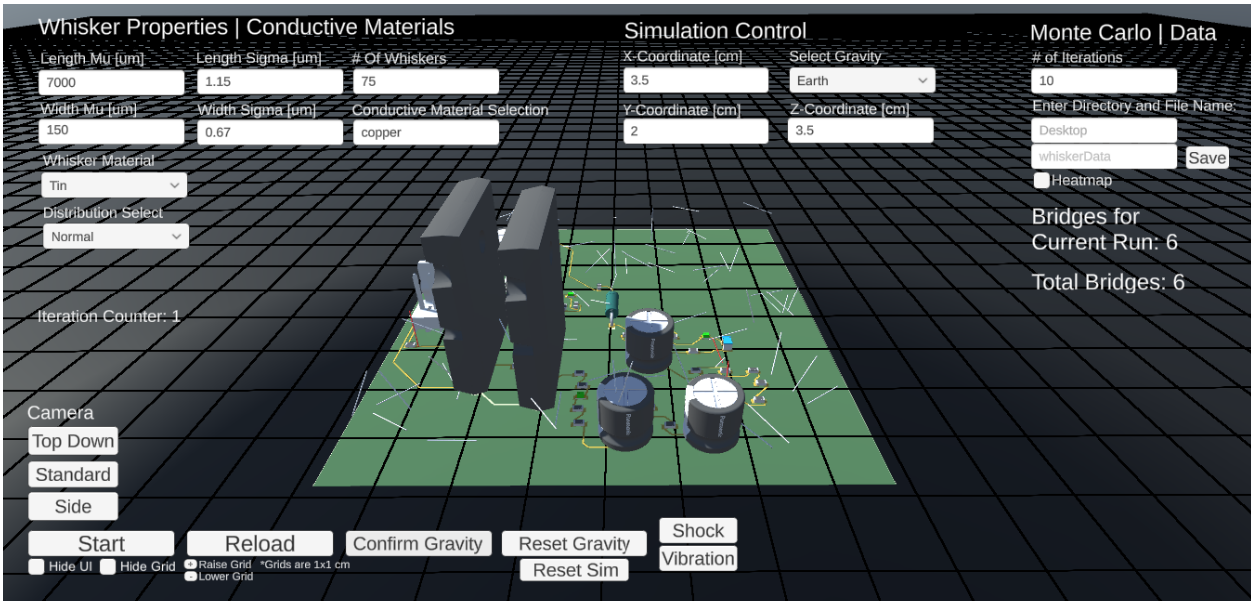
Enter Directory and File Name: Desktop / whiskerData

□ Heatmap

Bridges for Current Run: 0

Total Bridges: 0

Figure 21: Data Storage Input



Whisker Properties | Conductive Materials

Length Mu [um]: 7000

Length Sigma [um]: 1.15

Of Whiskers: 75

Width Mu [um]: 150

Width Sigma [um]: 0.67

Conductive Material Selection: copper

Whisker Material: Tin

Distribution Select: Normal

Iteration Counter: 1

Simulation Control

X-Coordinate [cm]: 3.5

Select Gravity: Earth

Y-Coordinate [cm]: 2

Z-Coordinate [cm]: 3.5

Monte Carlo | Data

# of Iterations: 10

Enter Directory and File Name: Desktop / whiskerData

□ Heatmap

Bridges for Current Run: 6

Total Bridges: 6

□ Hide UI □ Hide Grid

*Grids are 1x1 cm

Figure 22: Simulation Running

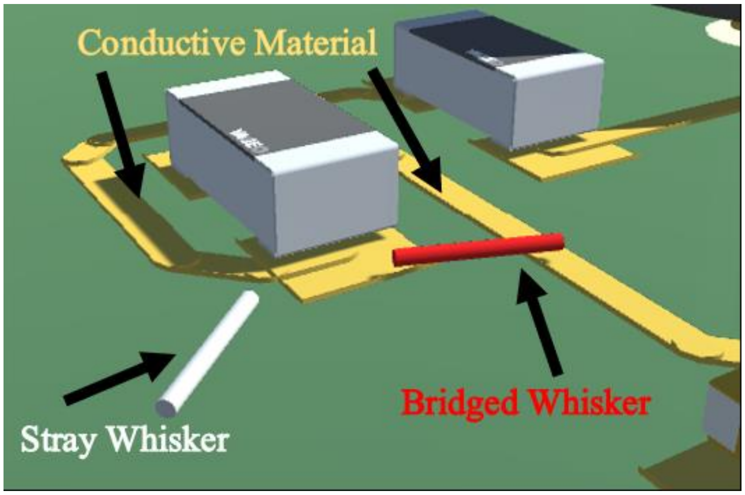


Figure 23: Bridged vs Stray Whiskers

Section 6: Heatmap

Once your simulation has been completed. A message will appear on the left side of the screen in green: "Simulation Complete!". The Heatmap toggle located in the **Monte Carlo | Data Storage** section of the Interface can be turned on if it has not been already. Post-simulation, assuming there were some bridges formed, some components will have changed colors representing how many bridges each component contributed to making. This allows a visual of what components are most likely to fail due to whisker bridging.

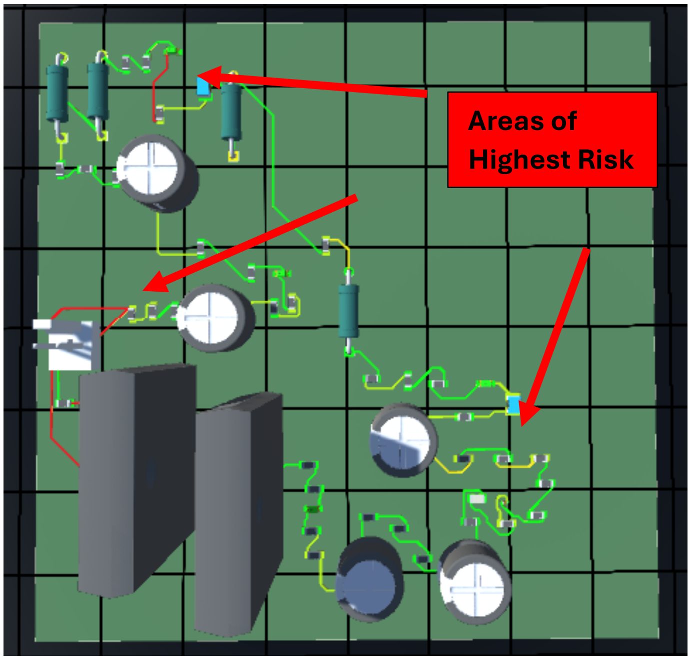


Figure 24: Heatmap – Post Simulation

Section 7: External Forces

As mentioned in Section 5, there are buttons in the user interface that open the external force input fields. To use them properly, please refer to the following steps:

Shock | Impact

a. To apply the mechanical shock or impact force to the board, press the Shock button. This will open the Shock Force Interface. There will be an input field for the value of force, a dropdown list that allows the force to be applied to any axis, a toggle, and a new Shock button underneath the dropdown.

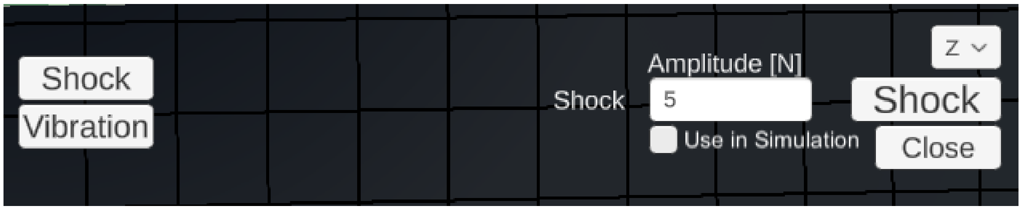
b. **Enter in a value for the Amplitude (N) text field. This will tell the program how many Newtons of force to apply to the board. Decimal and negative values are accepted. Entering in a negative value will push the board in the opposite direction.

c. Select an axis to apply the force to through the dropdown. The options are X, Y, and Z.

d. Finally, press the Shock button on the right of the screen to apply the shock. This will apply the force to the board based off the input and direction given.

e. The toggle can then be checked. Clicking this will run the shock continuously, using the inputs provided by the user, while running multiple iterations of the simulation.

** NOTE, any shock value over 10 N could cause whiskers to be pushed dramatically in the direction of choice. It is highly recommended to operate in the 0-10 N range.


Z

Amplitude [N]

5

Figure 25: Sample Shock Input

Vibration

a. To apply a vibration to the board, press the Vibration button. This will open the Vibration Interface. Similarly to the shock, there will be an input field for the Amplitude of the vibration, a dropdown list that can select the axis to apply the vibration, a toggle, a new vibration button, and inputs for the Frequency and Duration of the vibration.

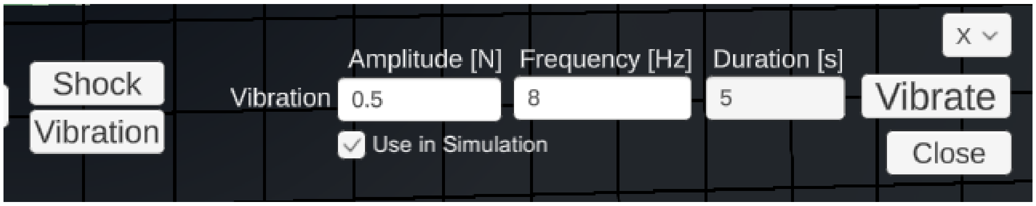
b. Enter in values for the Amplitude, Frequency, and Duration. A smaller amplitude (e.g. 0.2) and a larger frequency (e.g. 5-10) will provide a proper vibration, whereas a larger amplitude with a smaller frequency will provide a much smoother and wider vibration. The amplitude can be a negative value, while the Frequency and Duration cannot be.

c. Select an axis to apply the vibration on from the dropdown list.

d. Press the vibration button to apply the vibration.

e. Check the toggle to on to apply the vibration across multiple iterations when running the simulation fully.

**Set the vibration duration to 5 seconds or less before pressing Confirm Gravity. Failure to do so will result in the vibrations overlapping and moving the board out of place.



Vibration Amplitude [N] 0.5 Frequency [Hz] 8 Duration [s] 5

☑ Use in Simulation

Figure 26: Sample Vibration Input

Section 8: Resetting the Simulation | Saving New Data

To reset your simulation, start a new set of iterations, or save a new set of data, you will have to reset the simulation. There are a few ways to do this. The first method involves using the Reset Sim button located near the bottom of the screen when viewing the Interface.

a. Click Reset Gravity. This will set the gravity back to 0 and prevent the iterations from running until any new changes to the inputs are made and ready, along with preventing any excess or unwanted data from being stored.
b. (Optional) Clear the Directory and File Name input fields and press Save.
c. Set the Number of Whiskers field to 0 and Press Reload to remove the whiskers from the surface of the board, preventing premature bridging detection.
d. Click the Reset Sim button. This will clear the bridges detected and the iteration counters so a new run can be completed.
e. Enter new parameters into the input fields as needed.
f. Fill in the Directory and input a new File Name to store the new data set into. Press Save. (e.g. if the first .csv file was titled ‘whisker_data’, name the next one something new like ‘vibrationOnData’ or ‘whisker2_data’)
g. Press Reload with the new or original inputs until whisker are desired size and locations.
h. The simulation should now be ready to be run again by pressing Confirm Gravity.

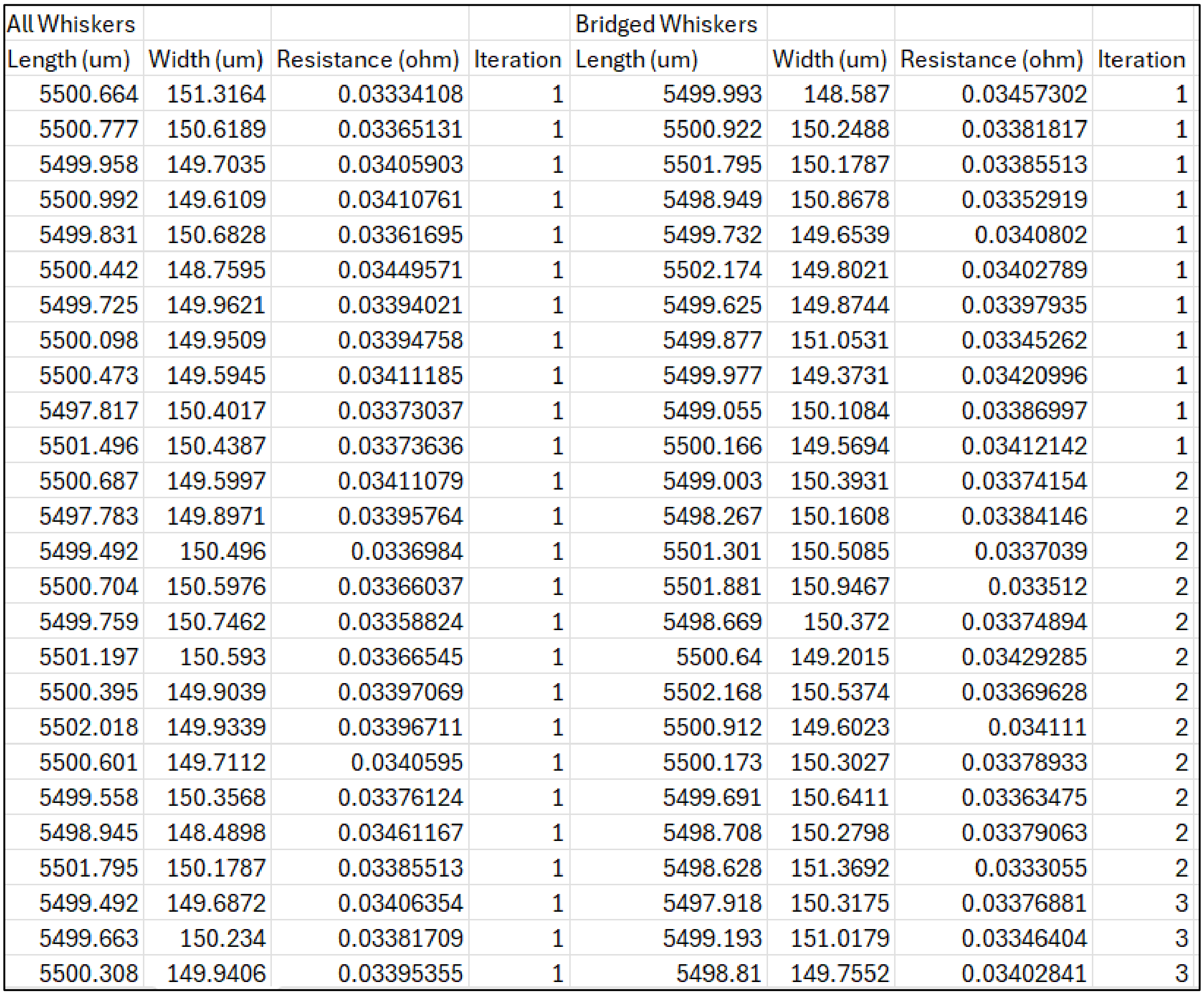
Alternatively, you can clear all inputs and values and fully reset the simulation by simply pressing the Play button at the top of your Unity Editor to fully stop the program, and then pressing it again to reopen it.

Data Processing

Section 1: Using WhiskerResults

After the data has been saved into your file of choice, you may navigate to the directory you have provided, and open the file. You will see that the data has been saved and formatted. Additionally, in the program files that were saved to your device previously from GitHub, there was an Excel file titled WhiskerResults. This .xlsm file is equipped with Excel macros and instructions which allows more in-depth results from the simulation to be generated and viewed.

Figure 27 shows an example of what the data might look like when stored into the .csv file you have generated. Notice that the file keeps track of every whisker that is generated, along with its dimensions, resistance, and what iteration it was generated in. The same applies to the whiskers that created a bridge. **Data values in the example are highly exaggerated.



All Whiskers				Bridged Whiskers			
Length (um)	Width (um)	Resistance (ohm)	Iteration	Length (um)	Width (um)	Resistance (ohm)	Iteration
5500.664	151.3164	0.03334108	1	5499.993	148.587	0.03457302	1
5500.777	150.6189	0.03365131	1	5500.922	150.2488	0.03381817	1
5499.958	149.7035	0.03405903	1	5501.795	150.1787	0.03385513	1
5500.992	149.6109	0.03410761	1	5498.949	150.8678	0.03352919	1
5499.831	150.6828	0.03361695	1	5499.732	149.6539	0.0340802	1
5500.442	148.7595	0.03449571	1	5502.174	149.8021	0.03402789	1
5499.725	149.9621	0.03394021	1	5499.625	149.8744	0.03397935	1
5500.098	149.9509	0.03394758	1	5499.877	151.0531	0.03345262	1
5500.473	149.5945	0.03411185	1	5499.977	149.3731	0.03420996	1
5497.817	150.4017	0.03373037	1	5499.055	150.1084	0.03386997	1
5501.496	150.4387	0.03373636	1	5500.166	149.5694	0.03412142	1
5500.687	149.5997	0.03411079	1	5499.003	150.3931	0.03374154	2
5497.783	149.8971	0.03395764	1	5498.267	150.1608	0.03384146	2
5499.492	150.496	0.0336984	1	5501.301	150.5085	0.0337039	2
5500.704	150.5976	0.03366037	1	5501.881	150.9467	0.033512	2
5499.759	150.7462	0.03358824	1	5498.669	150.372	0.03374894	2
5501.197	150.593	0.03366545	1	5500.64	149.2015	0.03429285	2
5500.395	149.9039	0.03397069	1	5502.168	150.5374	0.03369628	2
5502.018	149.9339	0.03396711	1	5500.912	149.6023	0.034111	2
5500.601	149.7112	0.0340595	1	5500.173	150.3027	0.03378933	2
5499.558	150.3568	0.03376124	1	5499.691	150.6411	0.03363475	2
5498.945	148.4898	0.03461167	1	5498.708	150.2798	0.03379063	2
5501.795	150.1787	0.03385513	1	5498.628	151.3692	0.0333055	2
5499.492	149.6872	0.03406354	1	5497.918	150.3175	0.03376881	3
5499.663	150.234	0.03381709	1	5499.193	151.0179	0.03346404	3
5500.308	149.9406	0.03395355	1	5498.81	149.7552	0.03402841	3

Figure 27: CSV File Data

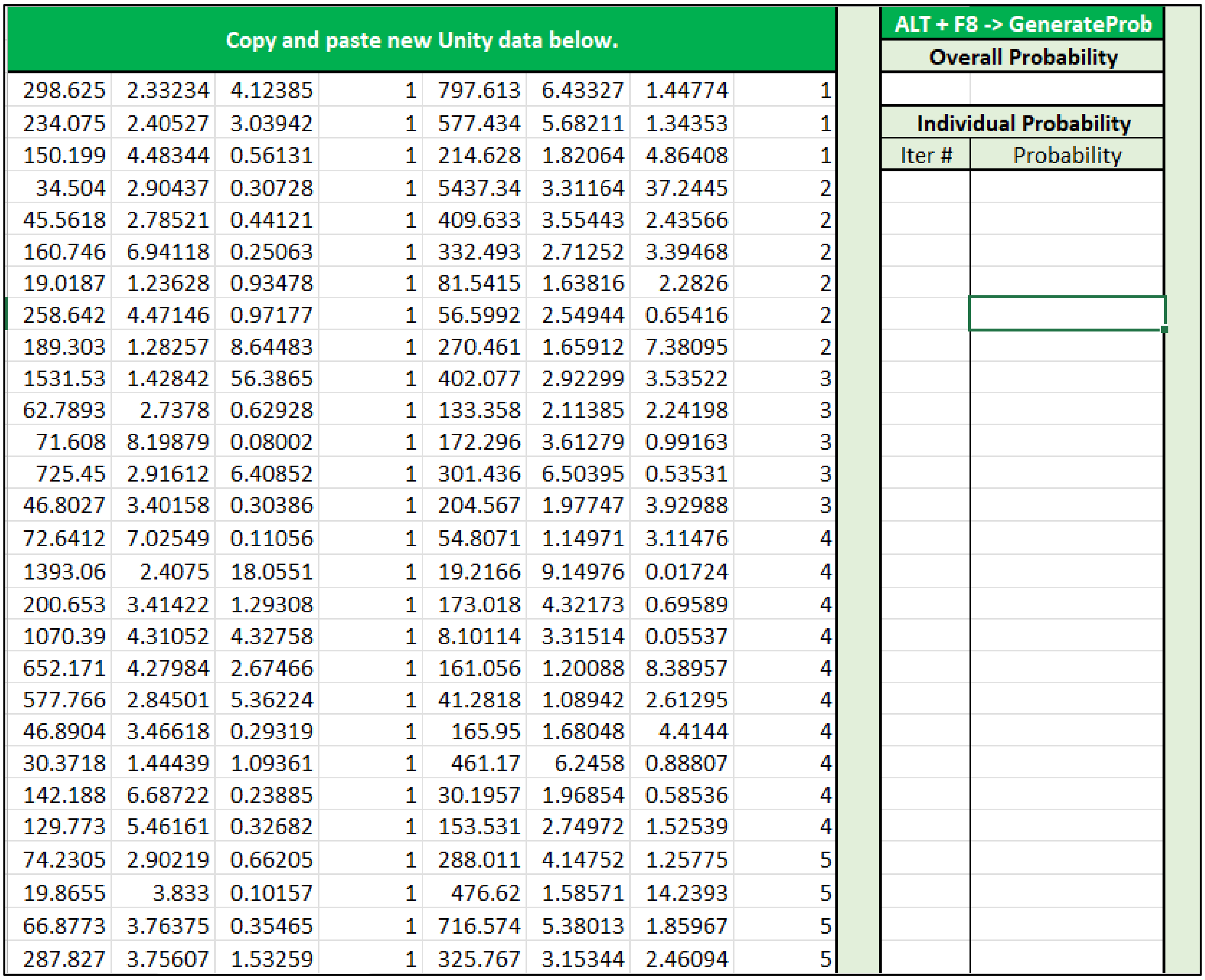
To take this data and get the results properly, refer to the following instructions for guidance:

a. Open WhiskerResults.xlsm.

b. Open the file containing your simulation data.

c. Copy and paste the values from your data set to the designated location in WhiskerResults.

d. Once all of the values are set into the proper location, there will be instructions listed inside of the file that give details on how to properly analyze the data.



Copy and paste new Unity data below.

298.625	2.33234	4.12385	1	797.613	6.43327	1.44774	1
234.075	2.40527	3.03942	1	577.434	5.68211	1.34353	1
150.199	4.48344	0.56131	1	214.628	1.82064	4.86408	1
34.504	2.90437	0.30728	1	5437.34	3.31164	37.2445	2
45.5618	2.78521	0.44121	1	409.633	3.55443	2.43566	2
160.746	6.94118	0.25063	1	332.493	2.71252	3.39468	2
19.0187	1.23628	0.93478	1	81.5415	1.63816	2.2826	2
258.642	4.47146	0.97177	1	56.5992	2.54944	0.65416	2
189.303	1.28257	8.64483	1	270.461	1.65912	7.38095	2
1531.53	1.42842	56.3865	1	402.077	2.92299	3.53522	3
62.7893	2.7378	0.62928	1	133.358	2.11385	2.24198	3
71.608	8.19879	0.08002	1	172.296	3.61279	0.99163	3
725.45	2.91612	6.40852	1	301.436	6.50395	0.53531	3
46.8027	3.40158	0.30386	1	204.567	1.97747	3.92988	3
72.6412	7.02549	0.11056	1	54.8071	1.14971	3.11476	4
1393.06	2.4075	18.0551	1	19.2166	9.14976	0.01724	4
200.653	3.41422	1.29308	1	173.018	4.32173	0.69589	4
1070.39	4.31052	4.32758	1	8.10114	3.31514	0.05537	4
652.171	4.27984	2.67466	1	161.056	1.20088	8.38957	4
577.766	2.84501	5.36224	1	41.2818	1.08942	2.61295	4
46.8904	3.46618	0.29319	1	165.95	1.68048	4.4144	4
30.3718	1.44439	1.09361	1	461.17	6.2458	0.88807	4
142.188	6.68722	0.23885	1	30.1957	1.96854	0.58536	4
129.773	5.46161	0.32682	1	153.531	2.74972	1.52539	4
74.2305	2.90219	0.66205	1	288.011	4.14752	1.25775	5
19.8655	3.833	0.10157	1	476.62	1.58571	14.2393	5
66.8773	3.76375	0.35465	1	716.574	5.38013	1.85967	5
287.827	3.75607	1.53259	1	325.767	3.15344	2.46094	5

ALT + F8 -> GenerateProb

Overall Probability

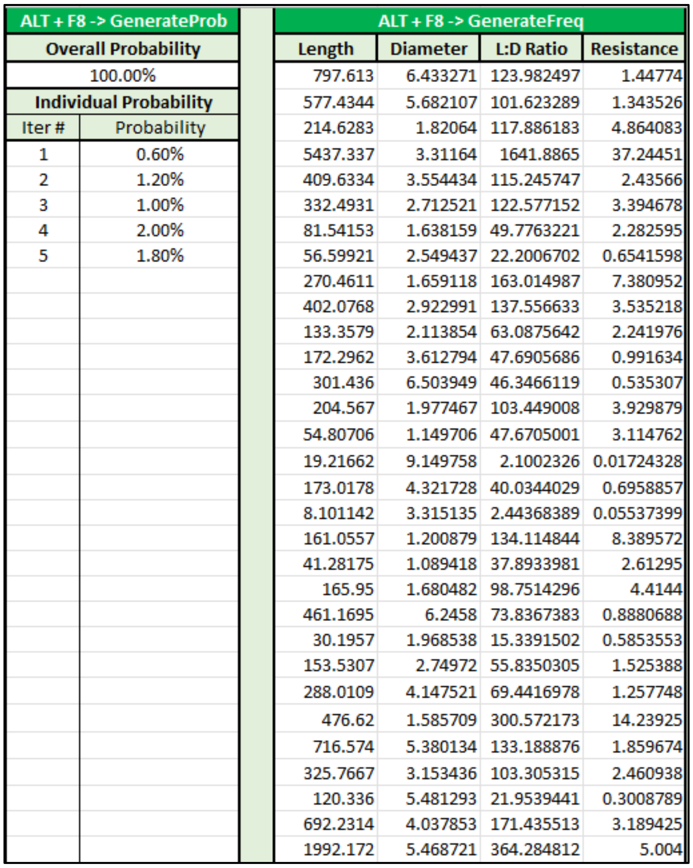
Individual Probability

Iter #	Probability

Figure 28: Sample Data in WhiskerResults

e. Pressing the Alt key and F8 at the same time will open the macro list used to generate the data.

f. Selecting the GenerateAll macro will analyze the data and produce something similar to what is seen in Figure 29 and Figure 30.



ALT + F8 -> GenerateProb	
Overall Probability	
100.00%	
Individual Probability	
Iter #	Probability
1	0.60%
2	1.20%
3	1.00%
4	2.00%
5	1.80%

ALT + F8 -> GenerateFreq			
Length	Diameter	L:D Ratio	Resistance
797.613	6.433271	123.982497	1.44774
577.4344	5.682107	101.623289	1.343526
214.6283	1.82064	117.886183	4.864083
5437.337	3.31164	1641.8865	37.24451
409.6334	3.554434	115.245747	2.43566
332.4931	2.712521	122.577152	3.394678
81.54153	1.638159	49.7763221	2.282595
56.59921	2.549437	22.2006702	0.6541598
270.4611	1.659118	163.014987	7.380952
402.0768	2.922991	137.556633	3.535218
133.3579	2.113854	63.0875642	2.241976
172.2962	3.612794	47.6905686	0.991634
301.436	6.503949	46.3466119	0.535307
204.567	1.977467	103.449008	3.929879
54.80706	1.149706	47.6705001	3.114762
19.21662	9.149758	2.1002326	0.01724328
173.0178	4.321728	40.0344029	0.6958857
8.101142	3.315135	2.44368389	0.05537399
161.0557	1.200879	134.114844	8.389572
41.28175	1.089418	37.8933981	2.61295
165.95	1.680482	98.7514296	4.4144
461.1695	6.2458	73.8367383	0.8880688
30.1957	1.968538	15.3391502	0.5853553
153.5307	2.74972	55.8350305	1.525388
288.0109	4.147521	69.4416978	1.257748
476.62	1.585709	300.572173	14.23925
716.574	5.380134	133.188876	1.859674
325.7667	3.153436	103.305315	2.460938
120.336	5.481293	21.9539441	0.3008789
692.2314	4.037853	171.435513	3.189425
1992.172	5.468721	364.284812	5.004

Figure 29: Probability and Frequency Data

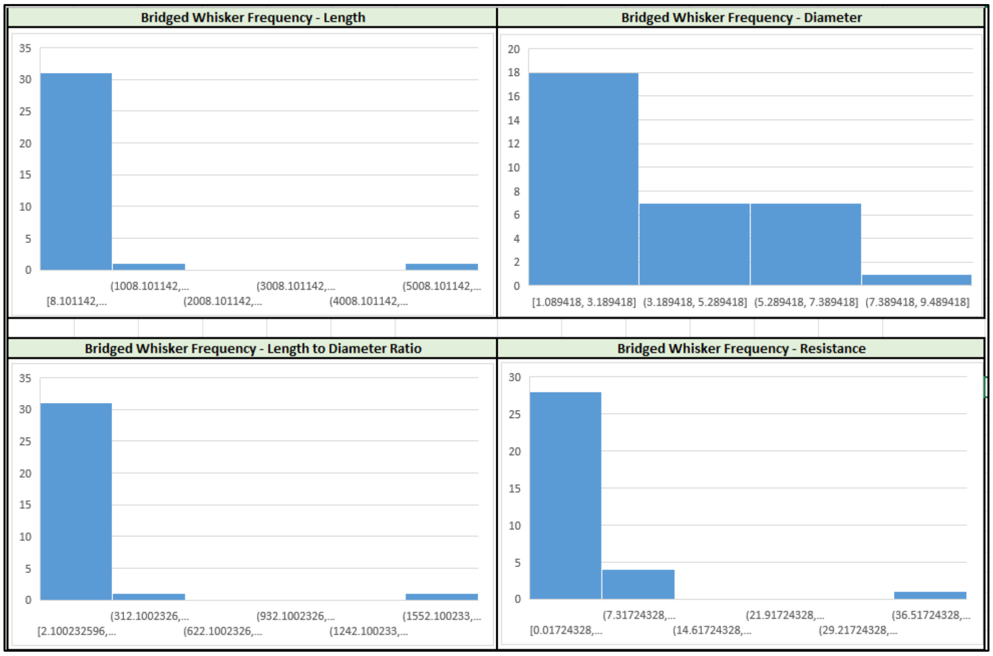


Figure 30: Frequency Plots

Extra Features

Section 1: Filtering Objects Based on Material

This is a built-in feature that the Unity Editor offers. What this allows users to do is when in the Scene view (Simulation is not running, users are examining their board or fixing orientation), they can filter objects based on the materials the components are made of. This can be useful when figuring out which components need to be labeled as conductive, as is done in **Running the Simulation, Section 2.**

a. To do this, navigate to your Project tab, into the Assets folder, and then to the Boards folder where your Board should be stored.

b. Press the small arrow to the right of your PCB icon to open up all components of the board.

c. Find any material you wish to see highlighted, and right click to bring up a new menu.

d. Click on Find References in Scene. This will highlight all objects containing the material you selected.

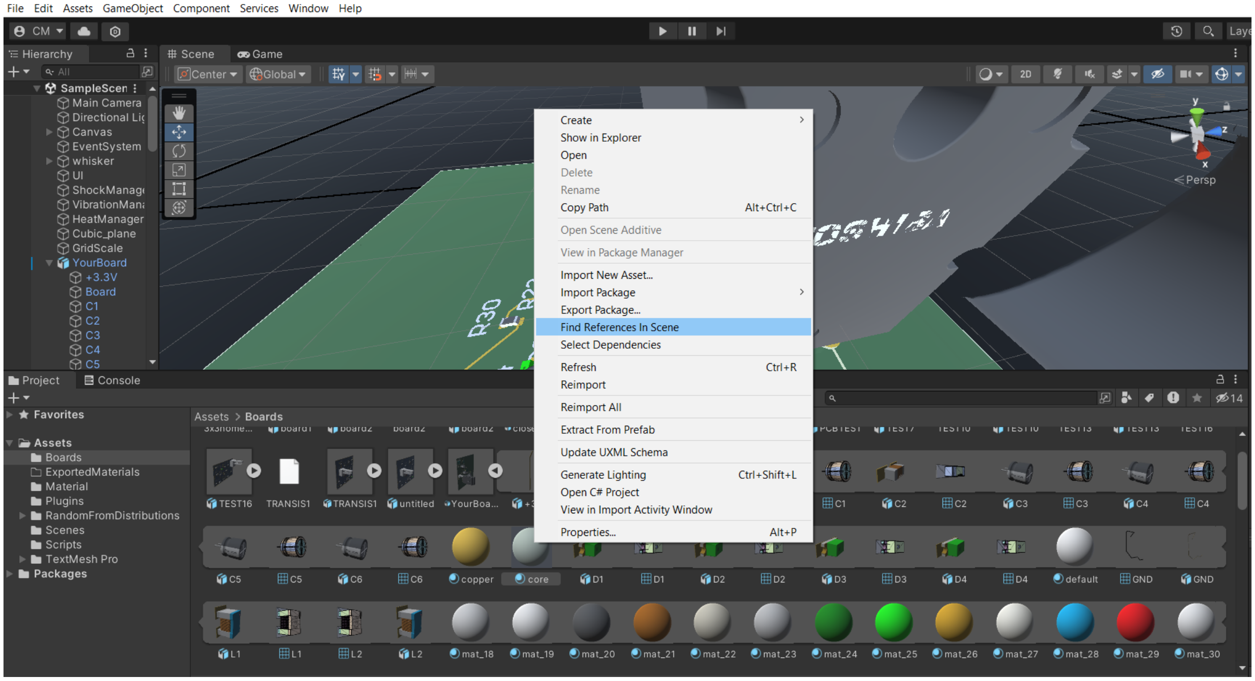


Figure 31: Material Filtering

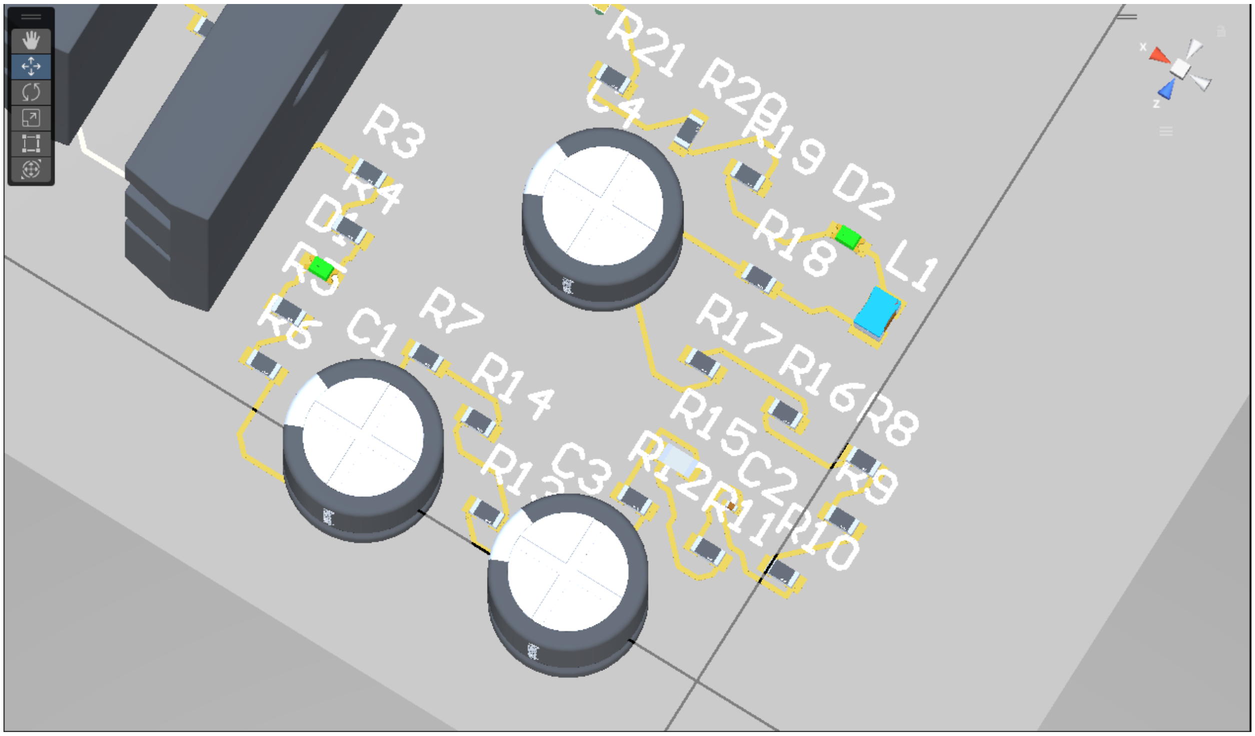


Figure 32: Filtering for Copper

Quick Overview

This section of the manual was designed to give users who are familiar with the simulation and already have it saved into their Unity Editor a brief overview of the instructions as a reference or reminder.

a. Import your board into the scene: Drag and drop your board from the file explorer on your personal device into the Boards folder in the Project tab, under Assets.
b. Orient the board if needed using the rotation parameters in the Inspector.
c. Add the Trigger Control script to the imported board and assign the Trigger Material and Material Input objects in their respective fields.
d. Press the Play button to run the simulation.
e. Fill in your Whisker Properties and search the board for any particular materials using the Conductive Material Selection input field.
f. Fill in your Simulation Control Parameters.
g. Set your desired number of iterations.
h. Press Start
i. Change parameters and inputs if needed, and press Reload to generate new whiskers.
j. If you are ready to save data, input a directory and file name into the inputs, and press Save.
k. If you desire to have one of the external forces to be applied, open the desired interface, fill in the inputs, and click the toggle to tell the simulation the forces should be applied.
l. Press Confirm Gravity to begin the simulation

m. Once the simulation is complete, the data will be stored into the csv file with the name you have given.

n. Copy and paste the data from the csv file into the WhiskerResults file included in the repository.

o. To reset the simulation, press Reset Gravity > Cler the Directory and File Name and press Save > Enter in new parameters and press Reload > Press Reset Sim

p. Alternatively, press the Play button again to fully stop the simulation, and press again to reopen.

Closing

This program has been designed with reliability and ease of use in mind. The goal of this user manual has been to provide a useful guide on how to use this program to its fullest potential. By following the instructions and tips provided, you'll be equipped to navigate every aspect of the program, regardless of experience with programs such as Unity or GitHub. It is highly encouraged to experiment with this tool and get the very most out of what it has to offer. Watch for updates being pushed out by Auburn's secondary team of engineers as they are made available, as they have been tasked with improving this simulation design further.