

# MonoWheel Board

This package contains a monowheel board, example code on how to use it on a character class actor, along with animations for the rider, and audio for its movement. There is a demo third person character (or 2 for UE5 versions) set up with the code to spawn and ride the monowheel board meant to be referenced so you can copy out the required code into your own characters. Do note you'll probably need to know the basics of blueprinting to do this properly, and may need to make some basic tweaks so when toggling between movement modes it returns to your own characters default movement logic in case it's different from default third person character types, such as modifying the move speed, orient settings, etc... when turning off the monowheel riding mode.

There are two board meshes, the normal and an outlined version if you want black outlines/anime style. There are three sets of materials for the board, and they all allow for color masking to modify and adjust the appearance to better match your own projects needs. This pack comes with assets and example characters for both UE4 and UE5 mannequins/skeletons making integration as quick and easy as possible.

The code is also replicated so you can use this in multiplayer.





## How To Use

This is how to copy the code over to your own character, it can be found in the demo characters.

Step 1. Go to class settings, then the Interfaces category in the details panel it brings up, and then add the MonoWheel\_Board\_BPI.

Step 2. Compile.

Step 3. Copy/Paste the code for the "MonoWheel\_Board\_Get\_IK" Interface function into your own blueprint from this demos interface.

Interfaces are in lower left "My Blueprint" tab, with the variables, functions, graphs, etc... under an "interfaces" section.

If your characters foot bones aren't positioned quite the same (like UE4 to UE5 mannequins) then create new sockets for it on the MonoWheel\_Board skeleton) and use those for the "get transform" nodes.

Step 4. Copy and paste this section of code for the spawning and despawning of the monowheel mesh, and changes to movement settings.

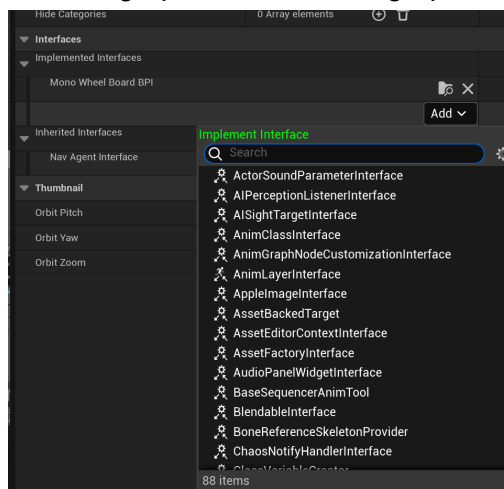
Be sure to make adjustments to the despawning section of movement code so when the monowheel despawns, it matches your own characters movement code/settings for walking/running.

Associate values can be found in beginplay node for creating character anim instance too.

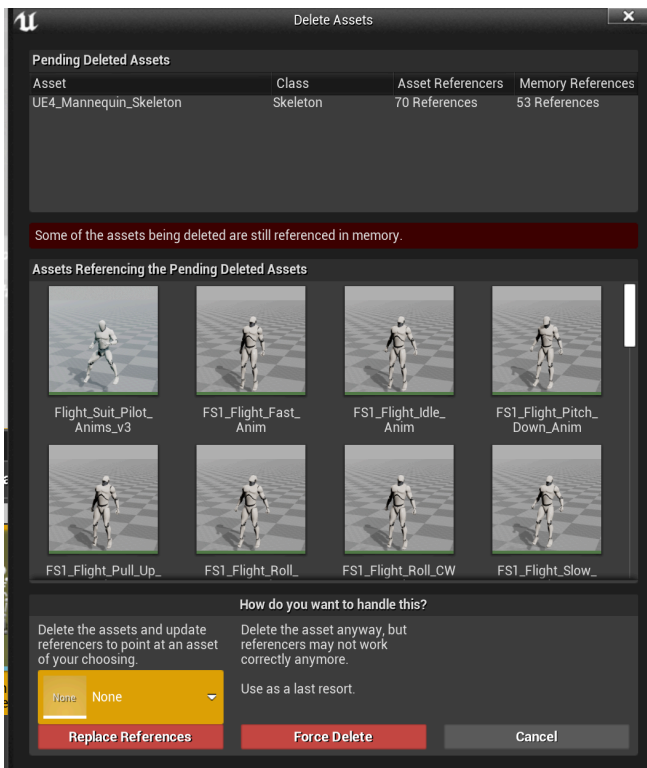
Step 5. Copy the modifier code into your own input/movement code. This is basically just replacing the inputs for move right to be zero if using monowheel. A throttle value for forward movement, and blocking jump when riding.

Step 6. Find the skeleton in this project for your given skeleton type (ue4 or ue5) and hit delete, then when it shows the references, in the lower left select the replace references option and find your own skeleton there. Then once set, delete and replace references so all the animations in the demo project are now on your own characters skeleton (your skeleton must be using the UE4 or UE5 mannequin skeleton respectively, if not, instead use retargeting to apply them to your owns keleton).

Step 7. Go into your characters animaiton blueprint. Add the MonoWheel\_Board\_BPI to it's interfaces list. Compile it. And then, copy/paste over the code from this demo characters animation blueprint. Be sure to copy both in the event graph, and the anim graph.



Example of adding the blueprint interface in the “interface” category. Remember to click class settings up top, then add it here, and then hit compile before trying to copy/paste over code that uses this BPI for stuff.



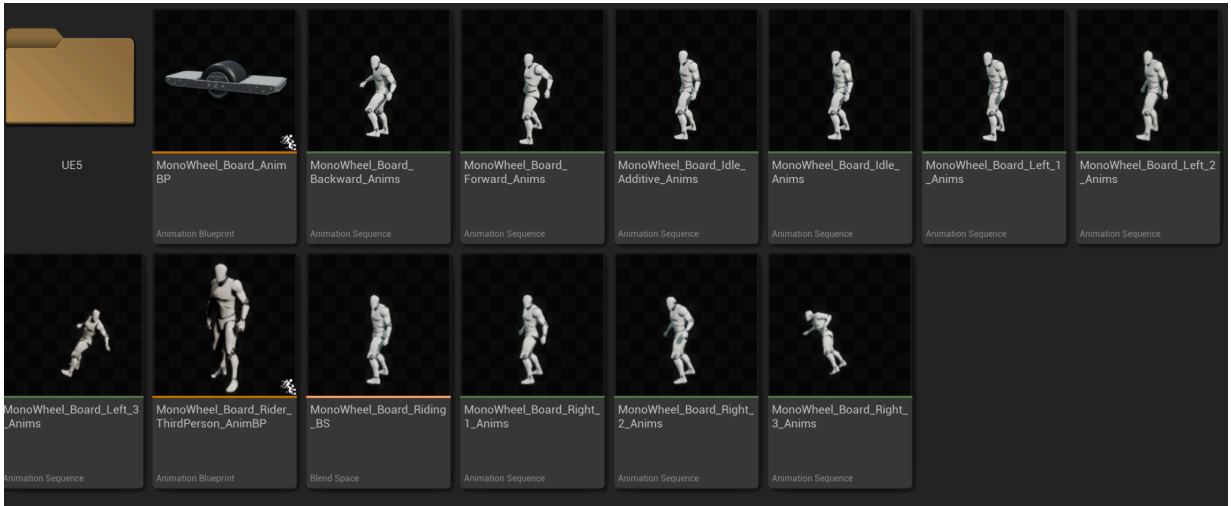
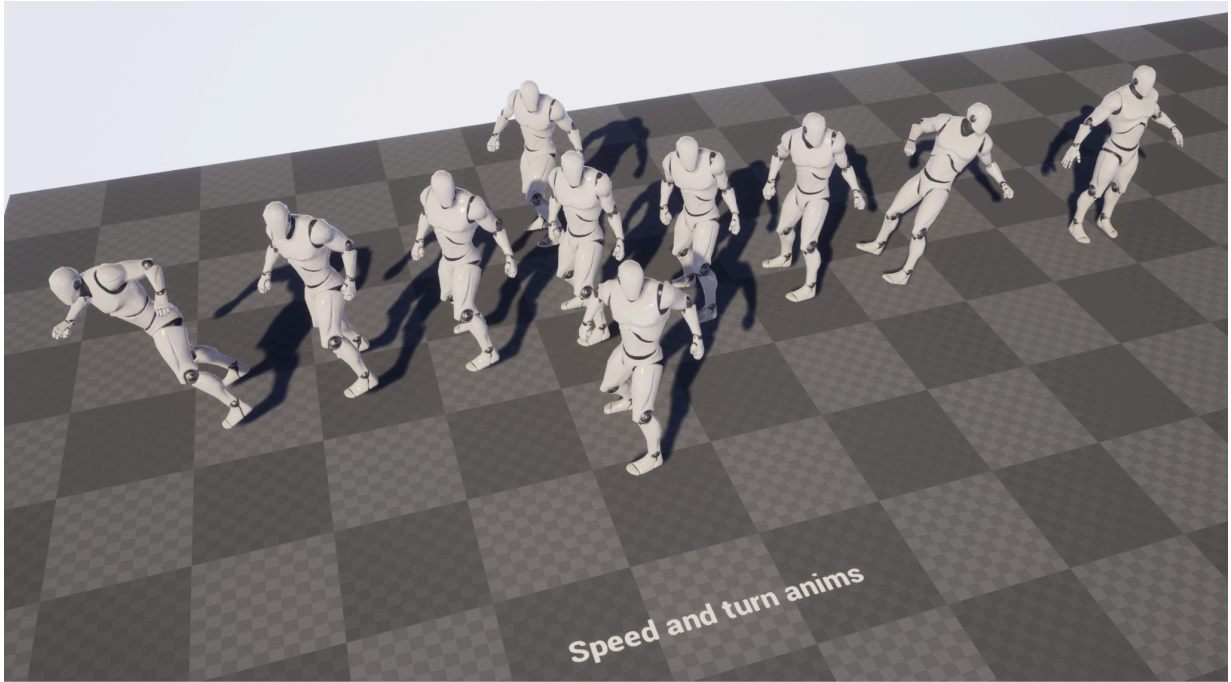
An example of deleting a skeleton, or other asset with references. Lower left, where it says none by default, you can select other assets that might take on these references. In this case, a compatible skeleton could be selected, then “Replace References” clicked and it will delete this skeleton and assign all animations/physics assets/animation blueprints/etc... using it, to your newly set skeleton.

Also as a side note, you might want to change the blur settings in your project to make the wheel look a bit better. Create a post process volume, and adjust the blur. In general the key part is to lower the Max value, and raising the FPS target can help too. But lowering blur in general just looks better in my opinion.



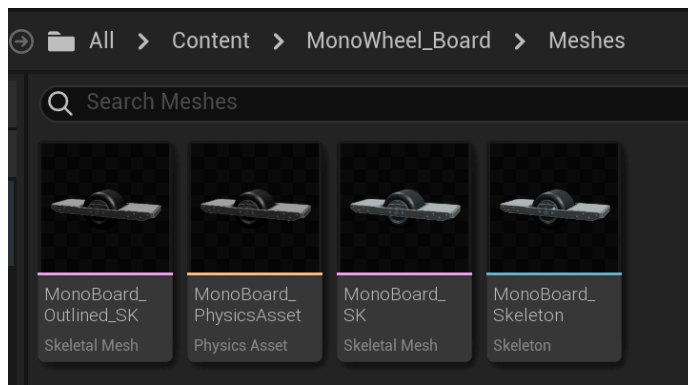
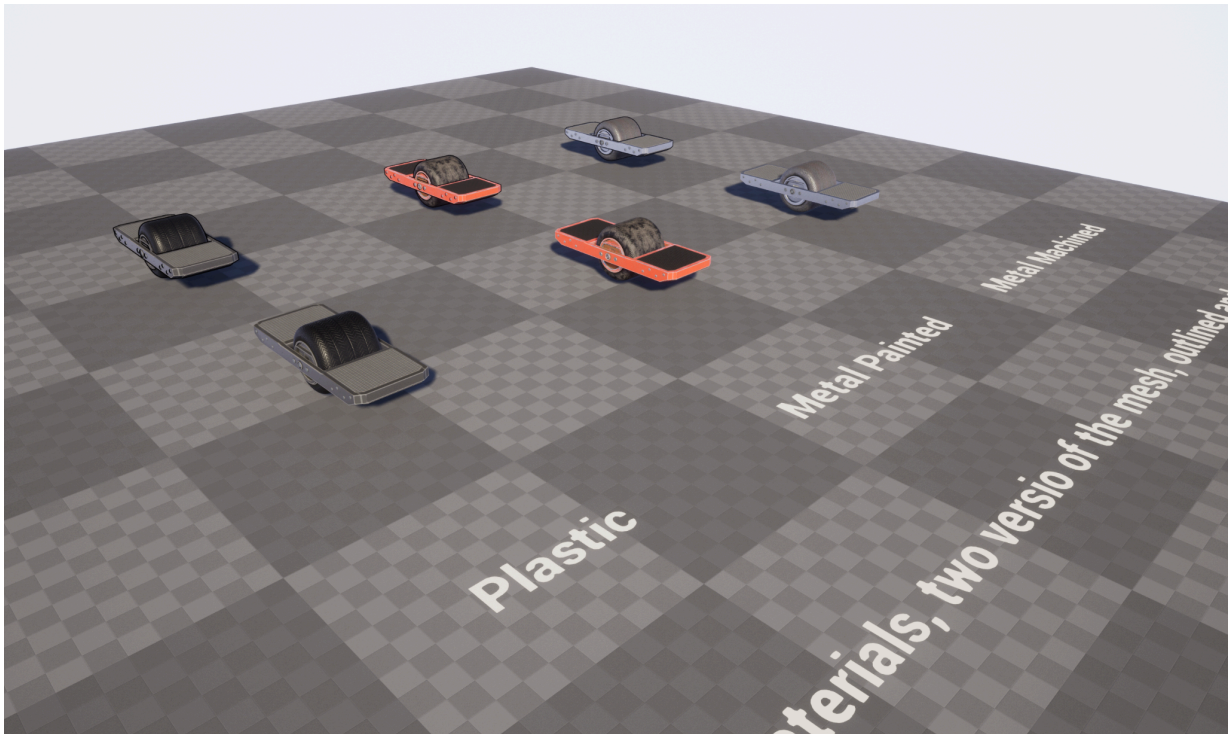
# Animations

There are ten rider animations, 2 blendspaces, and the animation blueprint for the rider. There are also copies of these all for the UE5 skeleton version in UE5 versions.



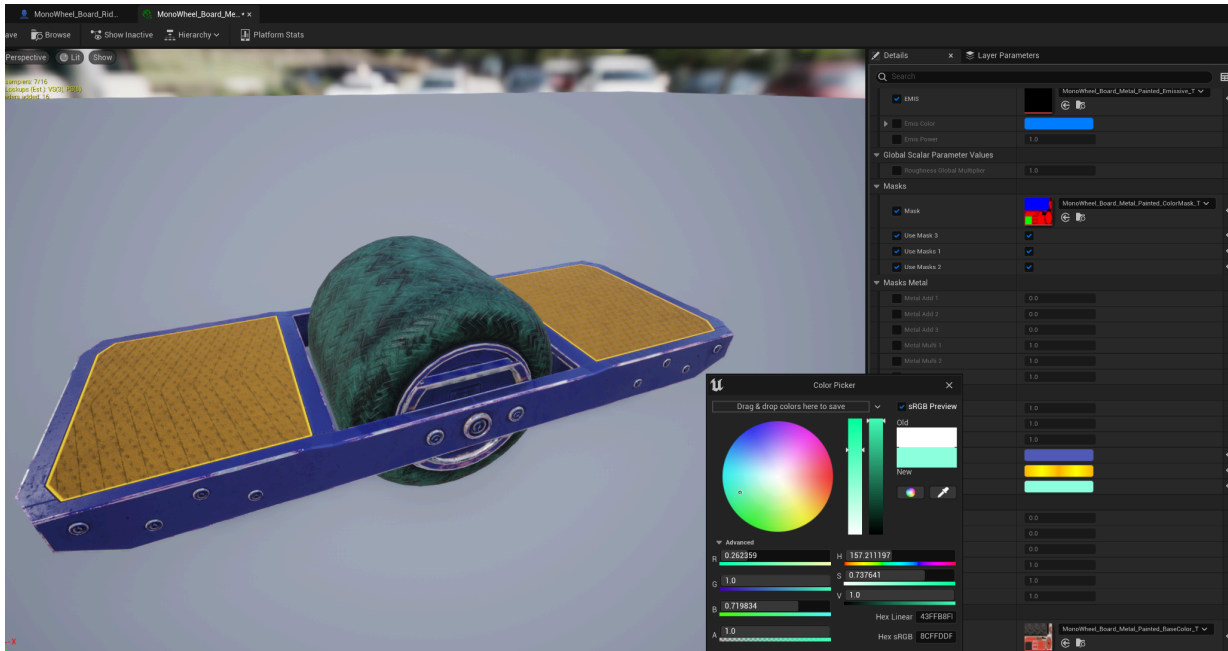
# Meshes

Here are the main meshes for the board.  
6K vertices. 6.5K polygons.  
Outlined version is basically double that.

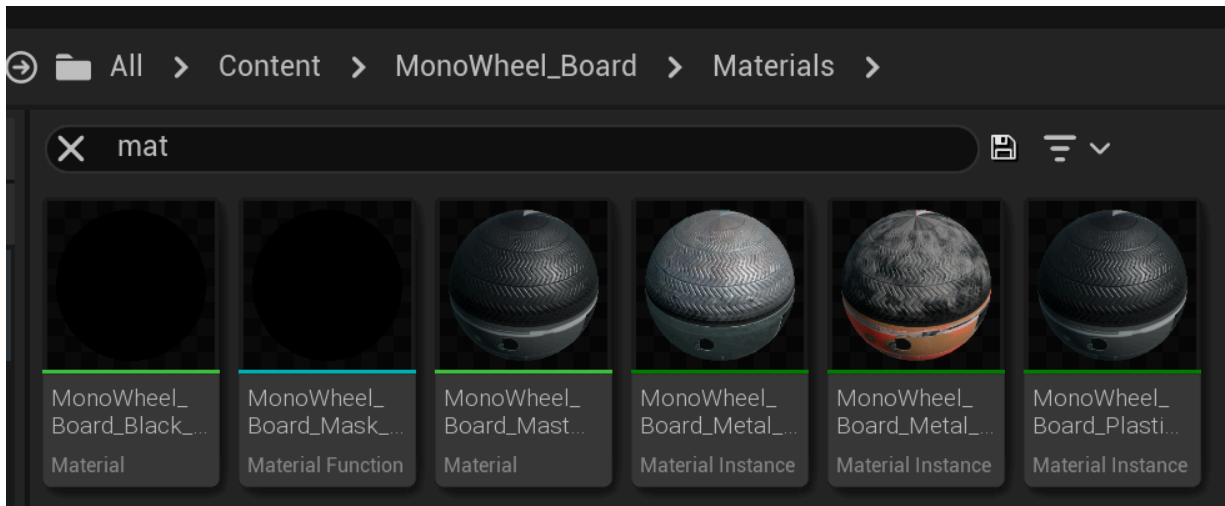


# Materials

Materials include some options to allow for basic changes to each material, an example of the “Metal\_Painted” material type being turned from it’s default red with grey grip to blue with orange grips and greenish tire.



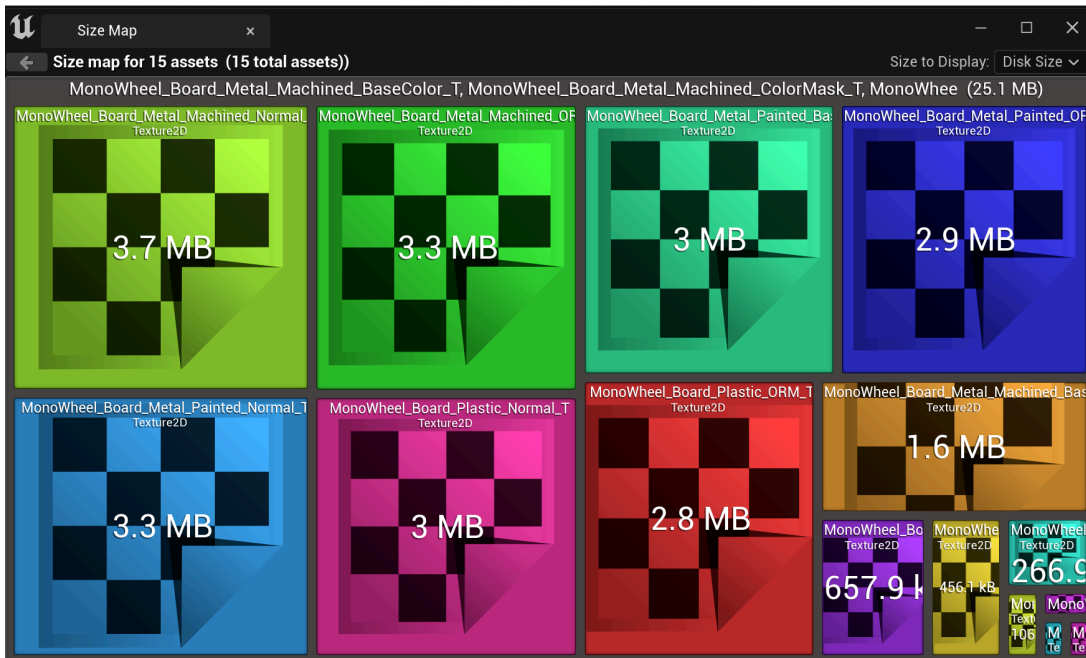
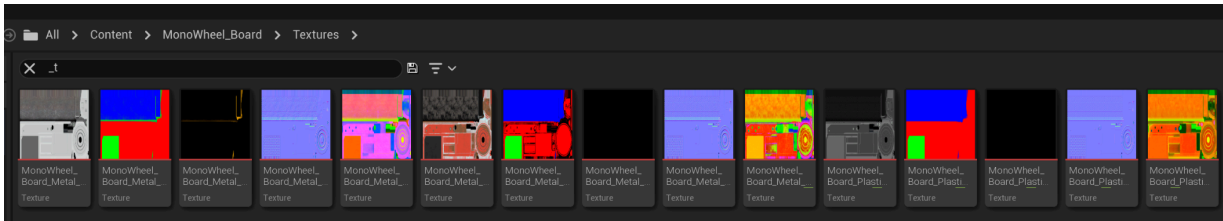
The pack contains 1 material function, 2 master materials, 3 material instances.



## Textures

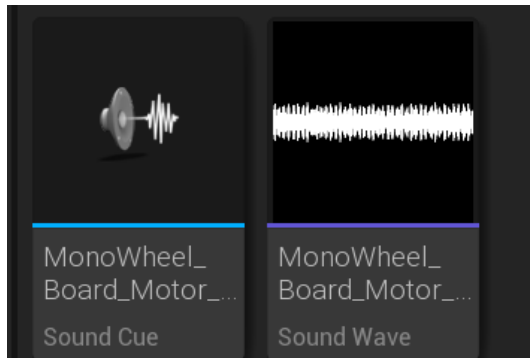
There are 15 textures for the MonoWheel Board. This is in 3 sets for the board, plastic, painted metal, and machined metal.

Total size is 25.1MB. Textures are 2048\*1024.



## Audio

There is 1 audio wave included in the project, plus the corresponding cue.





## Contact Info

Contact me at [Pineconedemon@gmail.com](mailto:Pineconedemon@gmail.com) or go to my discord channel here where you can chat with others or ask questions. <https://discord.gg/d7paEng>.

## Update History

