

# VR Movement for Oculus

revised: 20 May 19

---

## Quick Links:

- [FAQ](#)
- [Quick Start For Rift and Quest](#)
- [Quick Start For Oculus Go](#)
- [Module Information](#)

If you have any problems or questions please contact at: <https://discord.gg/W3CVAPu>

This is a comprehensive movement system for the Oculus Touch. The goal of this asset is to give you alternatives to VR's standard (arc) teleportation movement system. The Vive version is also available for sale.

To Try the Demo Go Here: <http://3lbgames.com/vr-movement-system-2-oculus/>

We have improved v2 with an easier-to-use Inspector and have made the movement systems modular. This will allow you to test different movement systems more quickly and efficiently. You can also modify them in order to create your own new movement systems.

### - New -

- Oculus Go Support (Some Modules not Compatible)
- Thrown Object Teleporter
- Drag World Movement
- Zero G Thrusters
- Module System

### - Features -

- Includes Low Poly Touch Models
- Thrown Object Teleporter
- Drag World Movement
- Thrusters
- Hand-Guided Movement Mode
- Point-and-Shoot Rotation Mode
- Teleportation Mode
- Flight/Grounded Mode
- Stick Rotation Mode
- Quick Stick Rotate Mode
- Blink Mode System
- Keyboard/Controller Debug Movement

- Line Arc System
- Rubber Band Movement
- VR Fade System

## Features

---

### Thrown Object Teleporter

Generates an object you throw and then teleport to wherever you throw it. It can even stick to walls.

### Drag World Movement

This system allows you to push or pull yourself through the world. Grip the world and move it around.

### Thrusters

Thrusters simulate water or space, allowing you to float around the world. You can edit drag and can even give players a brake button to stop themselves from moving.

### Hand-Guided Movement Mode

Your movement follows the direction in which you point the controller. Point forward to go forward; point left and you drift in that direction.

### Point-and-Shoot Rotation Mode

This rotation mode is best for quick player turns in VR. Simply point the controller in the direction you want to face and press the button.

### Teleportation Mode

This simple teleportation scheme can work with a NavMesh, Tagged Colliders, or any other Colliders. Teleportation contains a fading system.

### Flight/Grounded Mode

This is a standard FPS movement system that allows you to determine whether the camera is attached to a fixed point above the ground or has freedom of mobility. That is, you can determine whether the camera will move along the ground/terrain or fly.

## Stick Rotation Mode

Those with harder constitutions can use the control stick on the Oculus Touch.

## Quick Stick Rotate

Turn a set number of degrees, combined with a fade, using the control stick.

## Button Rotate LR

Allows you to select a button, which will then turn you a set number of degrees.

## Forward-Blink Mode

The character blinks (teleports) to a determinable point.

## Keyboard/Controller Debug Movement

This mode allows you to use the system without the Oculus Touch controllers, using either the provided remote or the included Xbox controller.

## Line Arc System

A utility script that allows you to draw lines using the line renderer. This system connects two points and generates a parabola. You travel along that path.

## Rubber Band Movement

A new, unique system that allows you to place a point. From this point, your speed and direction are based on the distance and location you drag the controller.

## VR Fade System

Enable a fade for rotation or teleportation fade-out/fade-in. This works with most headsets. The shader is included.

# FAQ

---

**Q: Do any of the modules work with controllers or keyboard?**

A: No, the modules are designed specifically to work with the VR controllers.

**Q: I am getting a error that OVRInput.Axis2D.PrimaryTouchpad is not found?**

A: Your Oculus Intergration is out of date please delete the OVR in your project and update it using <https://assetstore.unity.com/packages/tools/integration/oculus-integration-82022/>

**Q: Why isn't the Teleporter more featured**

A: The whole point of this movement system is to avoid teleportation.

**Q: Slow Stick gives me simulator sickness.**

A: Slow Stick is not recommended. If you get motion sickness from it, use one of the other rotational systems, such as Quick Stick.

**Q: X module makes me sick.**

A: Simulator sickness affects everyone differently. We test our movement systems to ensure they cause the least amount of sickness, but at high speeds and with fast jerks, they can still cause some sickness even in experienced VR users.

**Q: Do I need DOTween?**

A: Yes, DOTween is required for the rotational and teleportation modules. It smooths out the rotations to help with simulator sickness. DOTween is a fantastic package that can save you loads of time. You can find more info here:

<https://www.assetstore.unity3d.com/en/#!/content/32416>

**Q: I am getting a compiler error: *The type or namespace name 'OVRInput' could not be found.***

A: You did not import the OVR Utilities. They can be found below:

<https://assetstore.unity.com/packages/tools/integration/oculus-integration-82022/>

**Q: Does this work for SteamVR?**

A: Yes. We have a version, found here:

<https://assetstore.unity.com/packages/tools/input-management/vr-movement-system-for-htc-vive-47329>

**Q: Does this work for Windows Mixed Reality?**

A: We are working on a version. As soon as we get our hands on a WMR headset we plan to work on it.

**Q: Can you add X to your movement system**

A: Sure send us an email and we will look into developing it.

**Q: Does this work with VRTK?**

A: We have recently added a section explaining how to use our system in VRTK [Go here](#)

**Q: Does this work with Oculus Go?**

A: Yes follow the instructions here [Oculus Go](#)

**Q: What versions of Unity does this work with?**

A: This has been tested up to Unity 2018.4 we recommend Unity 2018.2

**Q: What versions of Oculus Utilities?**

A: It works with all recent versions of the Oculus Utilities

**Q: Can the steamVR and Oculus versions work together?**

A: Yes turn them on and off depending on what VR mode you are in. You can detect this using code like below:

```
public enum VrMode
{
    Oculus,
    OpenVR
}

public static VRModeDetector.VrMode VRMode;

public static void DetectVRMode()
{
    Debug.Log(XRSettings.loadedDeviceName);
    if (XRSettings.loadedDeviceName == "Oculus")
    {
        VRModeDetector.VRMode = VRModeDetector.VrMode.Oculus;
    }
    if (XRSettings.loadedDeviceName == "OpenVR")
    {
        VRModeDetector.VRMode = VRModeDetector.VrMode.OpenVR;
    }
}
```

# Quick Start For Rift or Quest

---

To start with the movement systems, we highly recommend you begin with one of the bare bones systems we included. The guide below will tell you how to start from scratch.

**If you have compiler errors after Import, take these steps first. If you have any problems or questions please contact at: <https://discord.gg/W3CVAPu>**

- 1) OVR Integration is Imported:

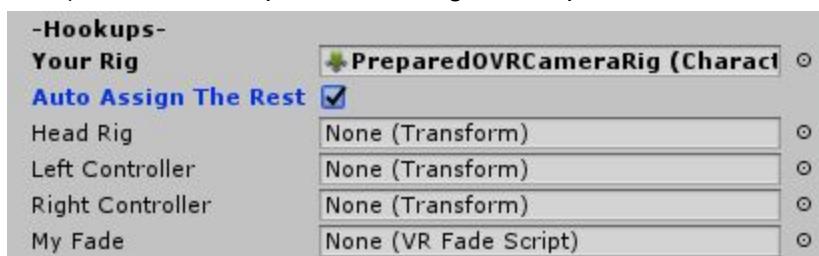
<https://assetstore.unity.com/packages/tools/integration/oculus-integration-82022>

## To Begin

- 1) Bring the **PreparedOVRCameraRig** Prefab
- 2) Adjust the character controller to your specifications or leave it alone.
- 3) Create an Empty GameObject
- 4) Add the **VRMovementOculus** Script to it
- 5) Pick the hand you prefer to control.



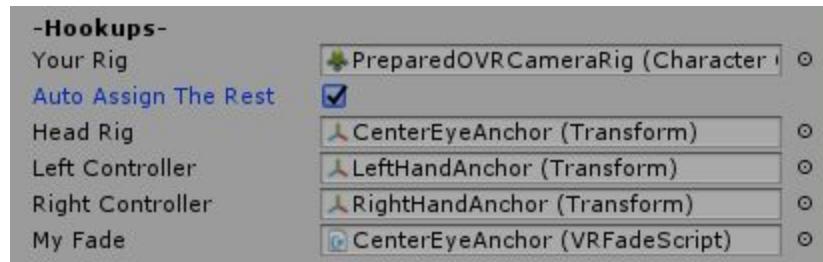
- 6) In the Hookups section drag the PreparedOVRCameraRig to Your Rig



- 7) Ensure Auto Assign is Checked
- 8) Press Play your Done!

If you need the old controllers use the OldOVRCameraRig

During Play Mode with our **VRMovementOculus** Hookups should look like:



You can modify the Prepared rig as you see fit or make your own rigs. Extra movement types are in the module section. Experiment away!

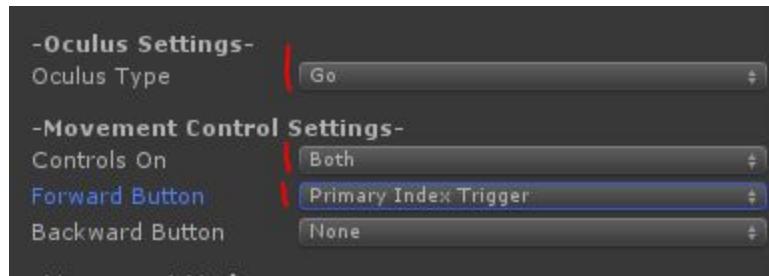
## Oculus Go Setup

---

First Ensure you are Up to date on the Oculus Integration

<https://assetstore.unity.com/packages/tools/integration/oculus-integration-82022/>

Follow Quick start using PreparedGoOVRCameraRig instead of PreparedOVRCameraRig and then make these changes to **VRMovementOculus** change the Oculus type to Go. It is also recommended that you set controls on to Both.



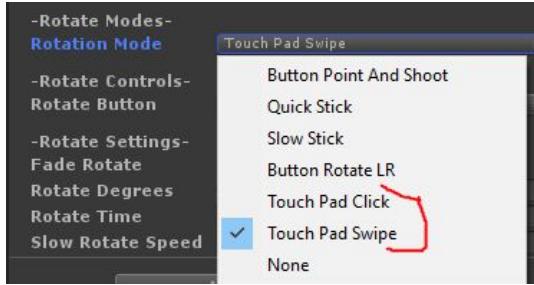
Note: Oculus Go does not have as many buttons as the oculus for example the Hand Trigger does not exist on the Oculus Go use **index trigger** like pictured above.

**The following modules are NOT compatible Oculus Go due to it not having positional tracking: Other modules may also not act as intended.**

- RubberBand

- Teleport Thrower
- Drag Movement

**VRMoveRotate** - you will have to use Touch Pad Click or Touch Pad Swipe



## VRTK Setup (1.0 not 2.0)

---

Our movement system does sorta work the VRTK but you have just a tad more setup to do:

First ensure that the OVRCameraRig is set to reset tracker on load this is inside of your VRTK\_SDKManager



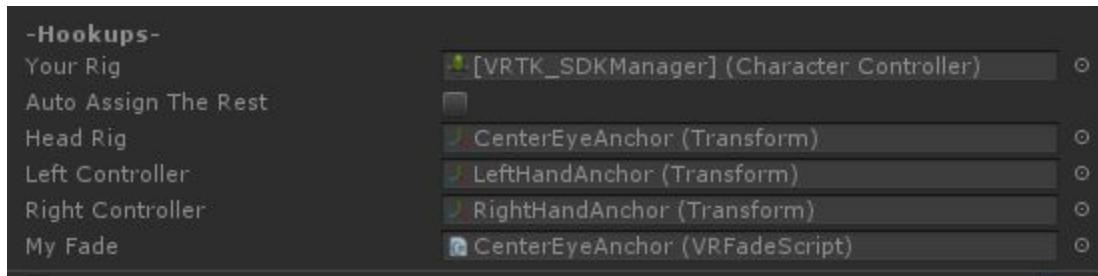
Add an empty object and assign VRTouchMove2 like usual.

### IMPORTANT:

*Character height is set by OculusVR ensure the Oculus is set to Eye level/ additionally the Character controllers collider can interfere with picking up objects.*

Add the Character Controller to the [VRTK\_SDKManager] adjusting height as needed.

Then setup your Hookups like this:



Add the modules as you see fit and you should be good to go!

We recommend you take VR Movement and parent it to the OculusVR Rig so that it will be turned on and off by the VRTK SDKManager. This will allow you to easily add our movement system for SteamVR. In the same manner.

## Module Information

---

Modules are used to enhance or replace the default Hand-Guided Movement. Some modules require additional setup information, which can be found by clicking on each of the movement setups.



[Hand-Guided Movement \(Default\)](#)



[Rotation System](#)



[Blink System](#)



[Teleportation](#)



[Drag World Movement](#)



[Rubber band Movement](#)



[Thrown Object Teleporter](#)



[Thrusters](#)

# Hand-Guided Movement



**VR Movement Oculus (Script)**

**Oculus Settings-**  
Oculus Type: Rift

**Movement Control Settings-**  
Controls On: Both  
Forward Button: Primary Hand Trigger  
Backward Button: None

**Movement Modes-**  
Can Move:   
Movement Mode: Flight

**General Settings-**  
Head Is Forward:   
Move Speed: 4  
Player Gravity: 300  
Basic Acceleration: 10

**Fade Settings-**  
Fade Time: 0.3

**Hookups-**  
Your Rig: PreparedOVRCharacterRig (Character Controller)  
Auto Assign The Rest:   
Head Rig: CenterEyeAnchor (Transform)  
Left Controller: LeftHandAnchor (Transform)  
Right Controller: RightHandAnchor (Transform)  
My Fade: CenterEyeAnchor (VRFadeScript)

**Oculus Type:**

Here you pick which Oculus hardware you are planning to use.

**Controls On:**

The hand you choose on this setting will translate to all movement modules.

Left - Controls only on left

Right - Controls only on right

Both - Controls either left or right

**Forward Button:**

Forward/Move button is used to move forward and activate modules.

**Backward Button:**

backward button is used to move backwards and is only used on the base module.

**Can Move:**

This is the toggle that can enable the movement part of the VRMovement system. This does not affect teleport/blink or the rotate module

**Movement Modes:**

Flight - Movement Without Gravity. You hover when you stop.

Grounded - Gravity is applied and pulls the player down.

None - Disabled Movement Mode (Not Modules)

None With Gravity - Disables Movement (Not Modules)

Remote - Debug Movement using Remote

Controller - Debug Movement using Controller

Keyboard - Debug Movement using Keyboard

**Head is Forward:**

This alter your movement direction to be based on the head direction rather then the controller.

**This will only work with Basic Movement.** The other modules require controller rotation

**Move Speed:**

This is the speed of the movement. In VR, it is hard to quantify speed in something like KM/H.

This is because in VR, size, scale, and points of reference can greatly change how you perceive speed. This setting is used by most of the modules.

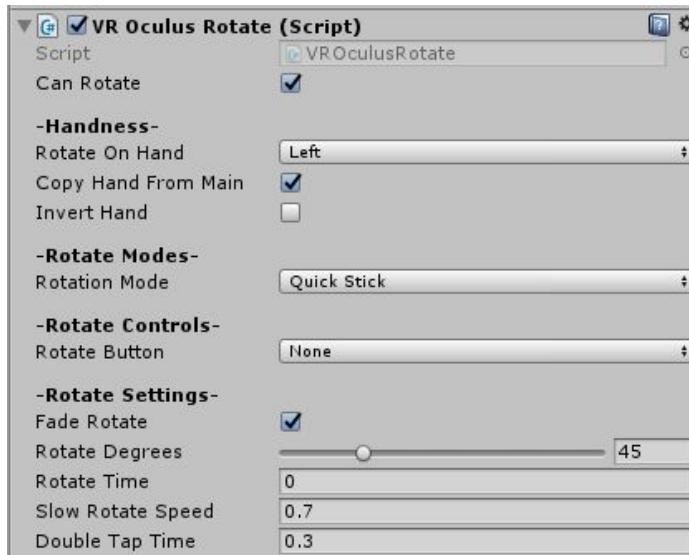
**Player Gravity:**

The gravity of the player applies a downward force when Movement mode is set to grounded or None with Gravity.

**Fade Time:**

This is how long it takes to fade from black to white in VR.

## Rotate Module (VR Oculus Rotate)



The Rotate module is designed to rotate the main camera rig. This allows the player in VR to stay facing forward while still getting a 360-degree experience. Facing forward also gives you the best current tracking for the controllers. We have three kinds of rotation.

### Handness

Rotate on Hand:

Which hand you will rotate on

Copy from Main:

This will copy the Controls on list from VRMovementOculus

Invert Hand:

This will invert the hand given to it so if the hand is set to left you will rotate on the right hand.

*Quick Stick:*

Quick Stick rotates you quickly 45 degrees left or right. It is activated by pressing left or right on the stick. Rotate Time is the amount of time it takes to do that rotation. The faster the better, and it can be faded.

Recommended: Quick Stick with a rotation speed below .3f

If your Fade Rotate is true, it is recommended you set Rotate Time to Zero.

*Point and Shoot*

Point and Shoot is used to orient the player by using the controller to pick a facing direction. When you press the Point and Shoot Rotate button, the player's view will orient to where they are pointing the controller.

*Slow Stick:*

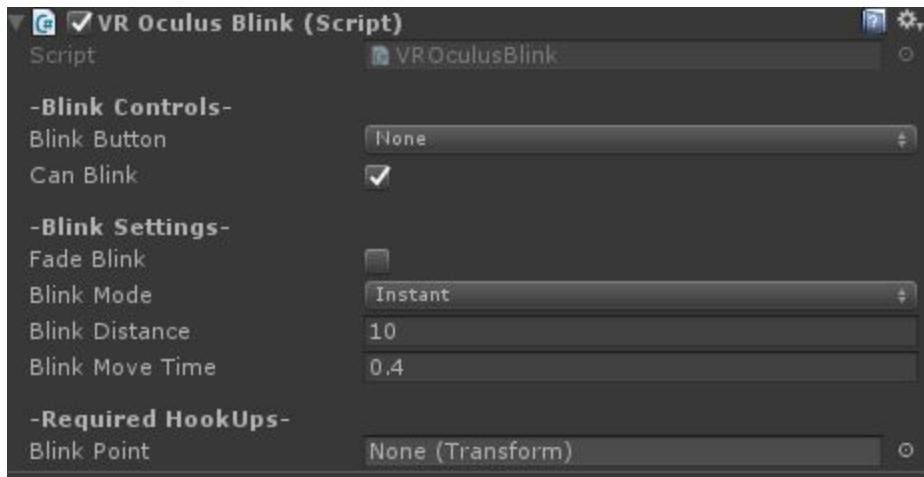
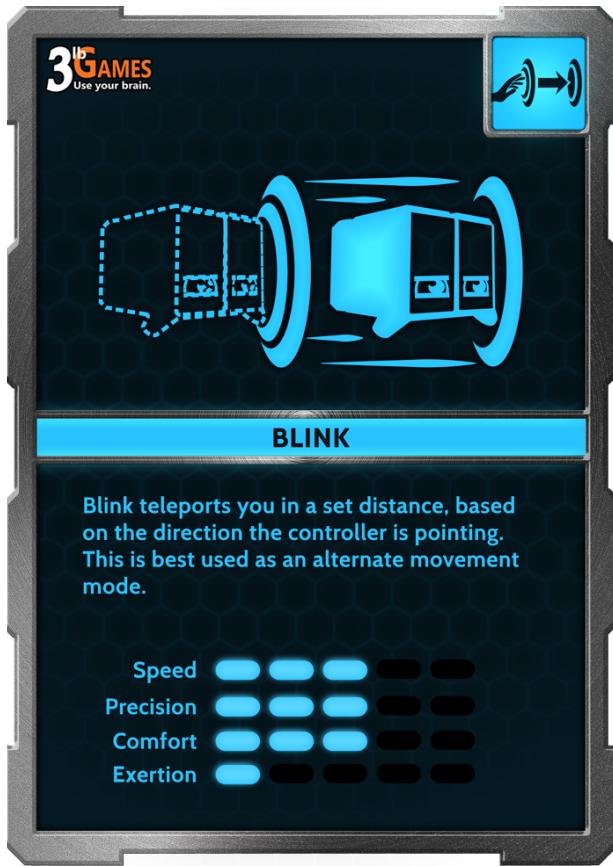
Slow Stick rotates the player slowly by pressing left or right on the stick.\*

\*Disclaimer: Even at slow speeds, this can cause simulator sickness in people. You should not use this method unless absolutely necessary. This cannot be faded!

*Button Rotate LR*

Using the rotation button will turn you a set number of degrees via controller with button press. This can be faded.

## Blink Module (VR Oculus Blink)



The Blink Module is a short-range teleporter. It can be aimed in mid-air allowing [place type of jumping here]jumping. Blink works using a ray cast. The ray is blocked by any collider, and will

reduce the blink distance accordingly. This, like almost all modules, works by pointing the controller. **Blink requires a hook up of the Blink Point**

*Blink Button:*

The button is used to activate blink.

*Can Blink:*

A toggle to turn blinking on and off.

*Fade Blink*

Use the fade when blinking.

*Blink Mode:*

Instant - The Blink button will teleport to where you point the controller.

Hold & Release - Hold the button and point the controller. On release, you blink to the point.

*Blink Distance:*

This is the distance of Unity Units you can blink.

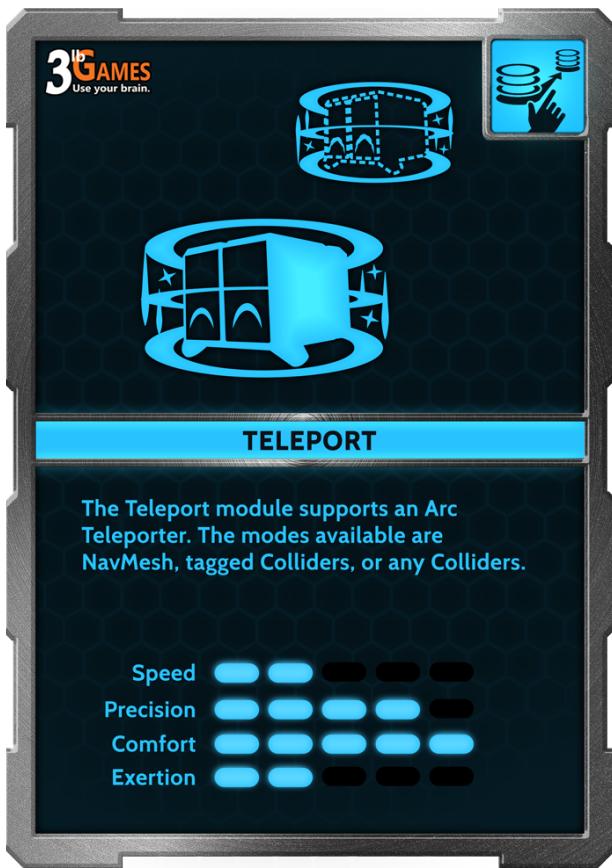
*Blink Move Time:*

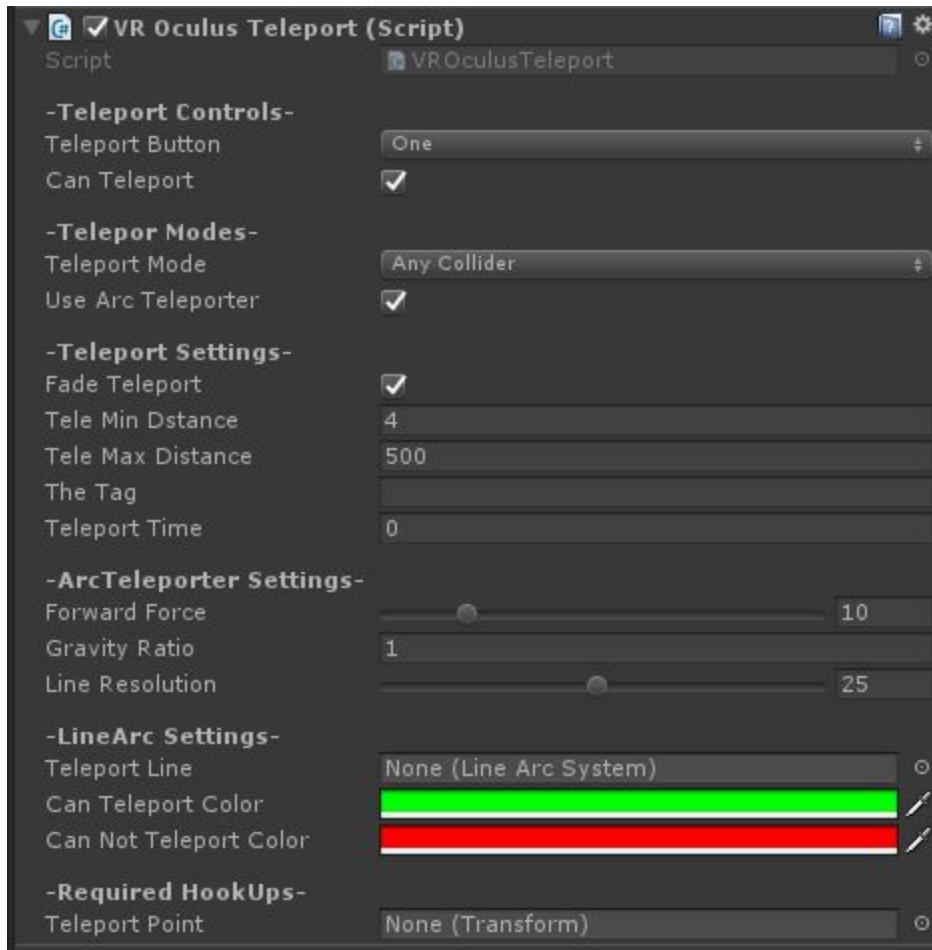
Time, in seconds, that it takes to blink to a selected point. Zero (0) is instant.

*Blink Point:*

The Blink Point is required and can be any gameobject without a collider, or any collider on a layer set to ignore raycasts. This hookup is required for the module to work correctly. If you don't want it to be visible, use an empty gameobject.

## Teleportation Module (VR Oculus Teleport)





Teleportation is what most VR developers default to when creating a movement system. We at 3lb Games believe that's a mistake. Teleportation is one of the reasons we created this asset. That being said, teleportation can occasionally be the best option. Please try the other movement modules before defaulting to teleport.

Teleport requires a hook

#### *Teleport Button:*

This is the button that uses the teleport this teleportation module is a hold and release system. This is the button used to activate teleport. When released, it will activate teleport if it finds a teleport able point. These points are determined by the teleportation mode. This teleport module only supports hold and release.

#### *Can Teleport:*

A toggle Boolean that enables and disables teleportation.

#### *Teleport Modes:*

#### **Any Collider**

The teleporter works on any collider that can detect raycasts.

#### **Tagged Point**

The teleporter will only interact with the set tag, and will always teleport to the center of the object. This is best used when you are creating set points where a player can teleport.

#### **Tag**

The teleporter will only interact with the set tag.

#### **Nav Mesh**

The teleporter will only interact with NavMeshes. **You must have baked navmesh in order for this to work correctly.**

#### ***Use Arch Teleporter:***

This alters teleporter to function like an [Arc Teleporter](#).

#### ***Fade Teleport:***

Use the fade when teleporting.

#### ***TeleMinDistance / TeleMaxDistance***

These variables are the minimum and the maximum range of the teleporter. These do not work if you are using the arch teleporter.

#### ***The Tag:***

Place the tag for Tagged and Tagged Point teleportation mode here.

#### ***Teleport Time:***

The time, in seconds, that it takes to move to the teleport point. Zero (0) is instant.

#### ***Arc Teleport Settings:***

(This requires the TeleportLine)

#### ***Forward Force:***

Amount of the forward force that the arch teleporter has. The more force, the further forward it goes before it falls to the ground.

#### ***Gravity Ratio:***

This is the amount of gravity used on the arch teleporter.

#### ***Line Resolution:***

This is the amount of points in the line arch. The more you have, the smoother the line will be.

*Teleport Line*

This is a LineArc Hookup. This is required.

*Can Teleport Color*

Changes the Teleport Line color when you can teleport.

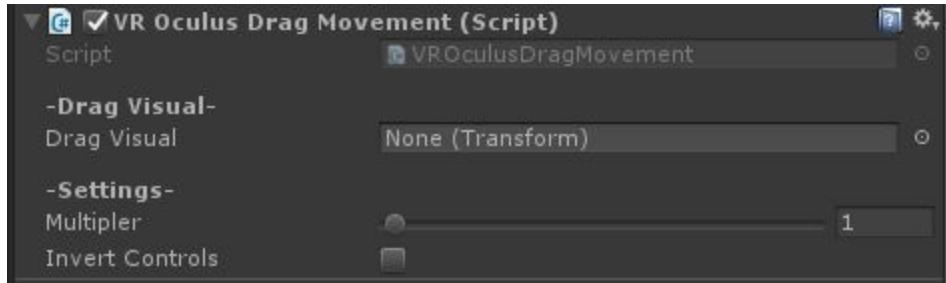
*Cannot Teleport Color*

Changes the Teleport Line color when you cannot teleport.

*Teleport Point:*

This is the Teleport point. It is required to be any gameobject without a collider (or a collider ignoring raycasts). This hookup is required for the module to work correctly. If you don't want it to be visible, use an empty gameobject.

## Drag Movement Module (VR Oculus Drag Movement)



The Drag movement module allows the player to drag themselves across space. By pressing the forward movement button, you grab on to a point in space and can pull yourself around. This is a natural movement.

### *Drag Visual:*

This allows you to create a visual where the drag point is added. This is optional.

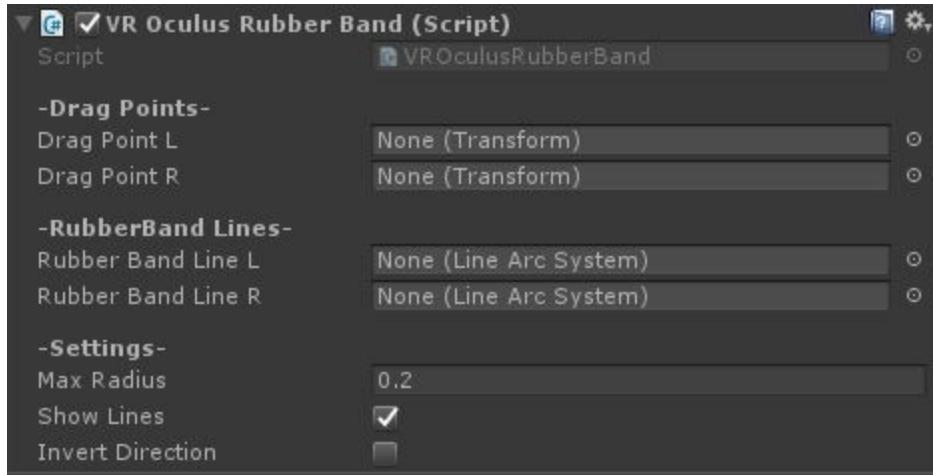
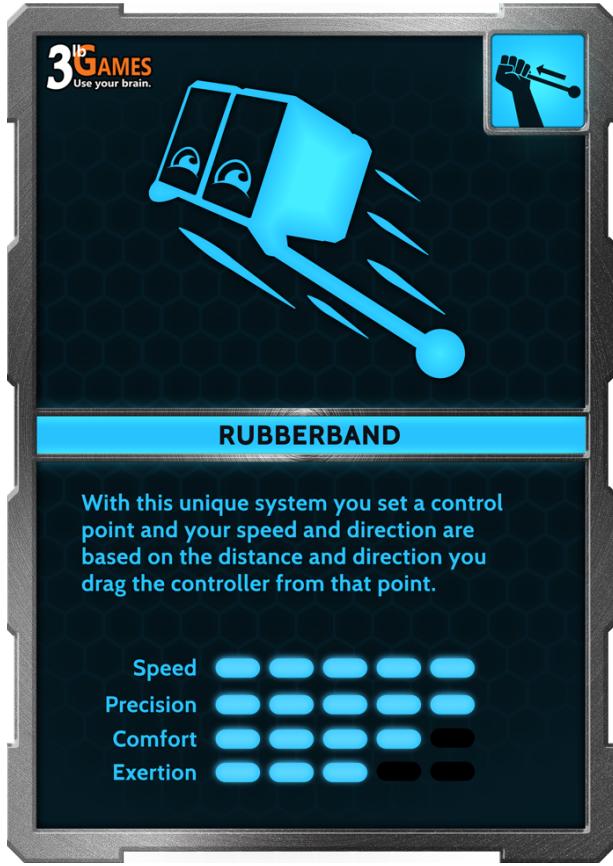
***Multiplier***

This magnifies the movement based on the amount set. Setting this to 2 means that for every meter you move your arm, you will move 2 meters, and so on.

***Invert Controls:***

Checking this box reverses the dragging. Instead of pulling, you will push yourself.

## Rubber Band Movement Module (VR Oculus Rubber Band)



Rubber Band works by pressing the forward button and placing a point and then using that point as a virtual joystick. **Rubber Band requires Drag Points and Rubber Band Lines to work.**

*Drag Point L / Drag Point R*

These are the drag points. They are required to be any gameobject without a collider (or a collider ignoring raycasts). This hookup is required for the module to work correctly. If you don't want it to be visible, use an empty gameobject.

*Rubber Band Line L / Rubber Band Line R:*

These are the Line arcs for the left and the right. These are required to make this plugin work correctly.

*Max Radius:*

This float is the max distance before you go at full speed. This protects the movement system for someone with longer arms moving too fast.

*Show Lines:*

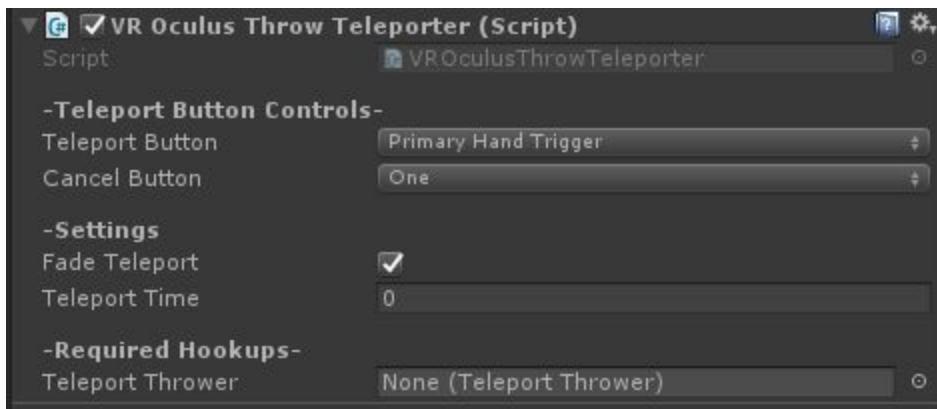
If checked, the Rubber Band lines will appear graphically

*Invert Direction:*

This will reverse the direction, pushing instead of pulling.

## Throw Teleport Module (VR Oculus Throw Teleporter)





This module allows a player to create an object and throw it to a location with a collider. When pressing the teleport button again, the player will teleport to the object's location. It is recommended that you use this in an enclosed environment or with kill zones for the teleporter so players don't warp outside the map.

*Teleport Button:*

Button you use to summon a Teleport Thrower and teleport to it.

*Cancel Button*

Cancels and destroys the throw teleporter.

*Fade Teleport:*

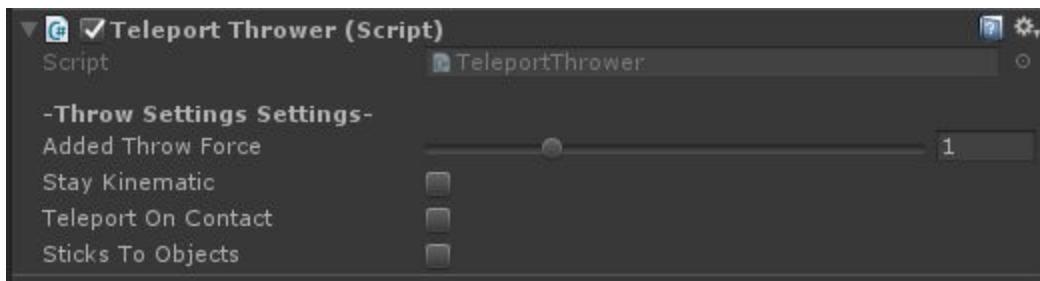
Use the fade when teleporting

*Teleport Time:*

The time it takes to move to the teleport point. Zero is instant.

*Throw Teleporter*

This takes an object with the TeleportThrower & a RigidBody attached to it. This can be found in the prefabs folder.



The Teleporter Thrower has its own set of variables that alter how this functions.

***Added Throw Force:***

Multiplies how powerful the throw is.

***Stay Kinematic:***

This forces the ThrowTeleporter to stay Kinematic. It will stick exactly where you place it.

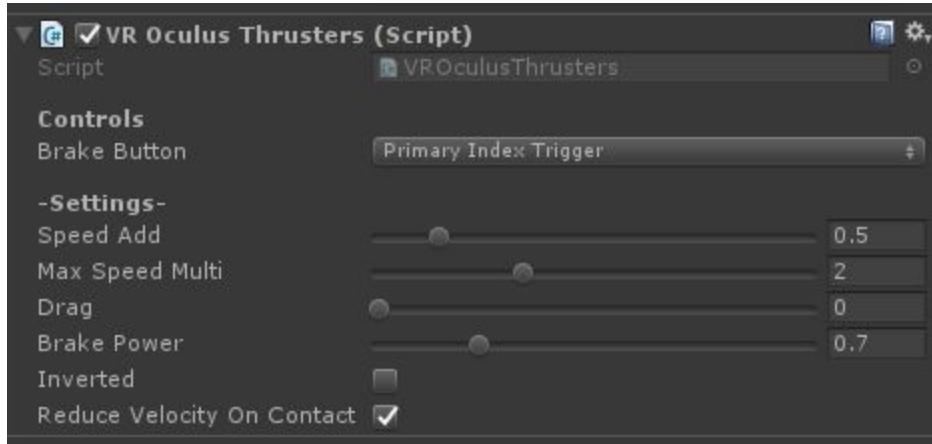
***Teleport on Contact:***

If the object hits a collider, it will force the player to teleport to it. *Do not use with Sticks To Objects.*

***Sticks to Objects:***

When the object hits a collider, it will stick to it, waiting for the player to teleport to it. *Do not use with Teleport on Contact*

## Thrusters Module (VR Oculus Thrusters)



This module is designed to work in zero gravity, simulating water or space. You use the controllers to create thrust to push yourself around. This module also can reduce velocity on contact with a collider. Using the move forward button, you activate the thrusters.

*Brake Button:*

This button will reduce the player's velocity to zero based on the brake power.

*Speed Add:*

This setting is how much speed is added when holding down the move forward button.

*Max Speed Multi:*

This setting clamps the max movement speed the player can go in thrusters. For example, 2x means they can go twice as fast as the move speed. This keeps the player from going too fast.

*Drag:*

This setting applies drag to the player at all times this can be used to simulate water or to smooth out movement. If this is set to zero, it will function like zero gravity, meaning the player will keep going indefinitely.

*Brake Power:*

The more brake power, the faster the player comes to a complete stop.

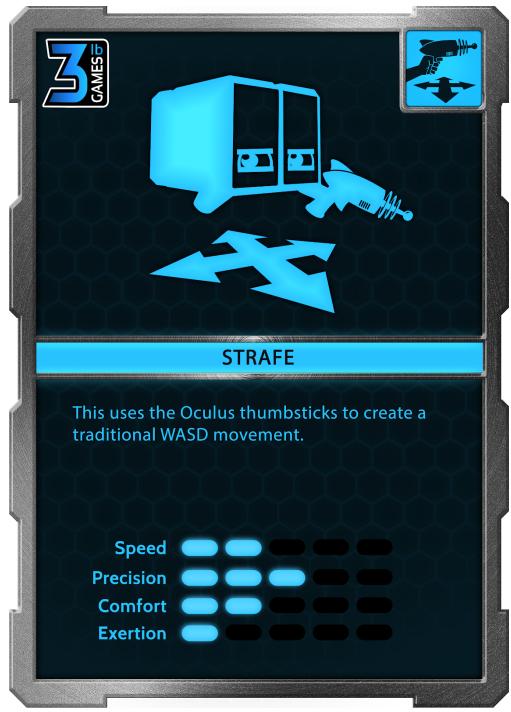
*Inverted:*

This inverts the movement. Instead of the force coming from the forward direction of the controller, it comes from the back direction.

*Reduce Velocity on Contact:*

This setting is recommended to be on, as it applies brakes as soon a player hits an object, meaning if they run into a wall, they will slow down and eventually stop.

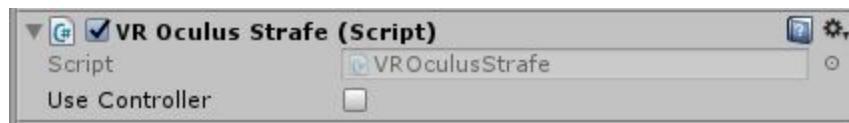
## Strafe Module (VR OculusStrafe)



This module is designed to emulate the movement that you see in FPS's using Analog sticks. Note that this module is for more advanced users and can cause simulator sickness in novice players. By default it uses the players Head as the forward direction.

This cannot be used with VR Rotate Quick stick on the same hand see the Strafe Prefab for proper setup.

This module does not work with the Oculus Go



### *Use Controller:*

Alters the forward direction from the default head to the controller