

Unreal Engine 4 - Marketplace Asset

First Person Puzzle Template

Developed by Divivor | Dennis Kastel



Documentation

Last update of this document: 03.02.2020

Description

The First Person Puzzle Template (in short: FPPT) is a game template which will allow you to use basic puzzle functionality based on logic gates and a power system. This asset was developed completely in blueprints and has all the basic functions included to easily expand the existing modules with your own creations.

This Asset can be integrated into existing projects without creating a new one.

Features:

- Power Transport with dynamic glowing Materials.
- Multiple Modules and easy expansion
- Logic Modules: AND, OR, XOR, Inverter, Delay, Splitter, Do Once
- Interactable Modules: Bridge, Door, Elevator, Moving Platform
- Each Module can play custom sounds on Activation/Deactivation
- Simple First Person Player Pawn included with footstep sounds
- Cube Highlighting and Collision Sounds
- Cube Activator Areas to activate/deactivate a Power Cube

Contents of this Documentation

Description	1
Contents of this Documentation	2
Changelog	3
Version 1.2	3
Version 1.1	3
Information	4
Enabling Cube Highlighting	4
Changing Power Types and Colors	4
Adding New Modules	5
Logic Gates and Activation in Editor	6
Integration for a custom Player Pawn	7
Modules	10
Getting Started	11

Changelog

Version 1.2

- Fixed an error where the Glow of the Emissive Material wasn't displayed correctly in the editor.
- Added comments to Blueprints.
- Updated the Documentation.
- And a little bit of cleaning.

Version 1.1

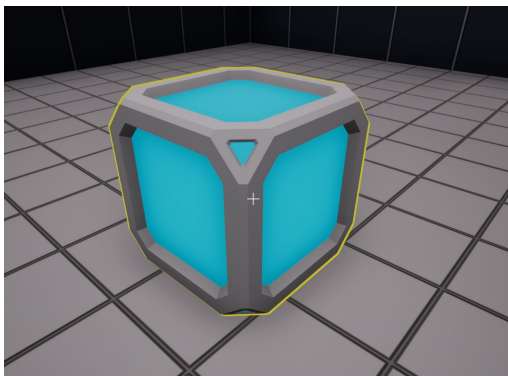
- Added Cube Activators to change the active state of a power cube.
 - BP_FPPT_CubeActivator Base Blueprint
 - BP_FPPT_CubeActivatorBox Trigger
 - BP_FPPT_CubeActivatorSphere Trigger
- Added a simple Grab Component to make it easier to integrate the Grab Mechanic into existing Projects.
 - BP_FPPT_GrabComponent
- Added a Demo Level with a complete Level that showcases most of the functionality provided in this asset.
- Added a Do Once Logic Gate. It allows you to cut off power and save the state of the target modules.
 - BP_FPPT_DoOnceGate
- Added a Moving Platform to the Interactable Modules. It plays a sound while it is moving around.
 - BP_FPPT_MovingPlatform
- Added the ability to add sounds to Modules. You can customize Sounds for each Module and even spawned instances for them.
 - Activation
 - Activation Completed
 - Deactivation
 - Deactivation Completed
- Added Collision Sounds for the Power Cubes.
- Added simple footstep sounds to the First Person Player Pawn.
- There was a bug with the glow of modules if they haven't been activated during runtime. The glow would be less bright than it should be. This has been fixed.
- The folder structure has been restructured.
- Added 7 sounds including footsteps and activation sounds.
- Added Disco Cubes. *It's not a bug, it's a feature.*

Information

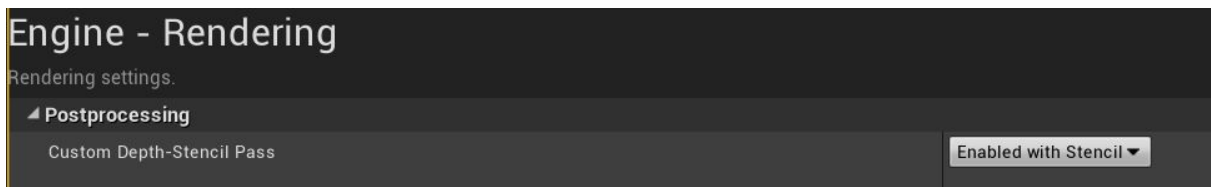
Before you begin using this asset, please make sure to read this page since it contains some information about the usage in the editor.

The controls of the Player Pawn that is included with this product has directly integrated the control schemes for a keyboard and mouse layout. You may want to change this to your Projects Input Events, as this is not recommended for use in a final product.

Enabling Cube Highlighting



To enable the highlight Outlines for Power Cubes, you have to edit your Project Settings. Search for “Custom Depth-Stencil Pass” under Engine > Rendering and change it to “Enabled with Stencil”.



Changing Power Types and Colors

Since you want to edit this asset to be included into your own project, you probably want to use different colors.

To add, remove or change the names of the power types you will have to edit the **Enum_FPPT_Colors** Enumeration File as well as the **DT_FPPT_Colors** Datatable File.

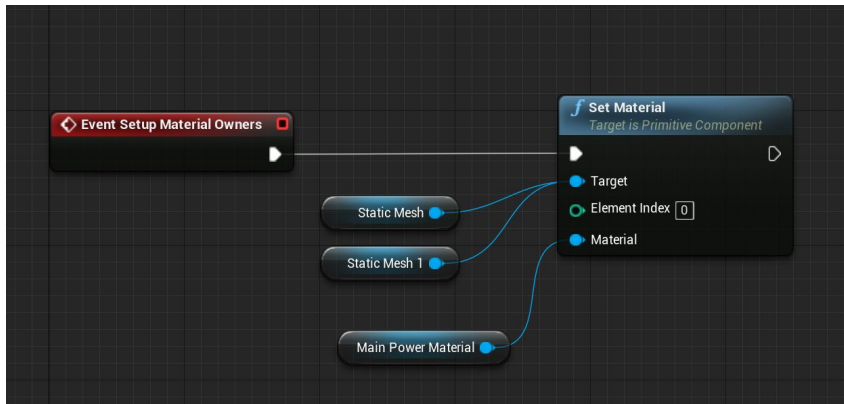
The Display Name of the Enumerators is equal to the Row Name in the Datatable. They have to be the same in order to work.

To change the Colors for each Power Type you have the options to edit **Base Color**, **Emissive Color** and **Emissive Amount** inside the Datatable File.

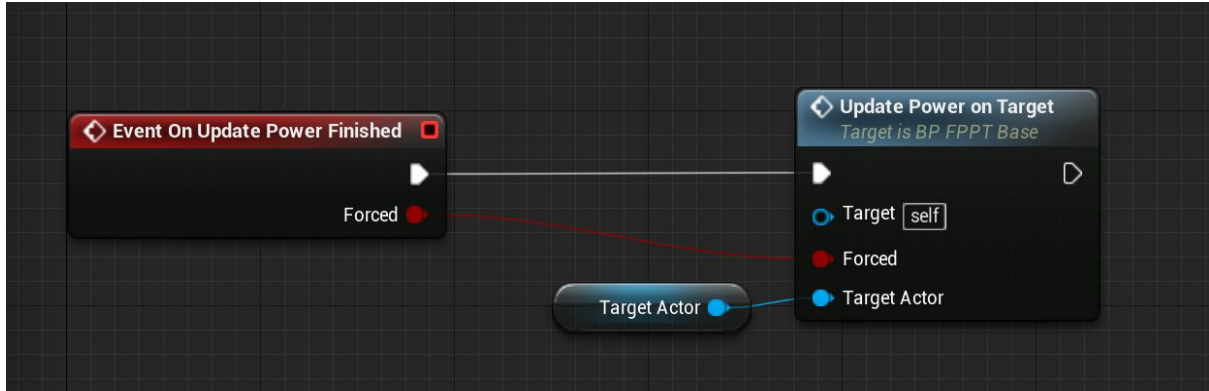
Adding New Modules

In order to add new modules for Logic Gates, Activation or Interactables, you will have to make a Blueprint based on the **BP_FPPT_Base** Blueprint.

The important part of adding new Modules is to add the dynamic Power Material, you will achieve this by overriding the function **“Setup Material Owners”** and set the material of your meshes to our **“Main Power Material”**.



You may also want to add a new Target Actor to give out the power to another module. For this you will have to add a new Variable with Type **Actor** and override the **“On Update Power Finished”** Function and add **“Update Power on Target”**.

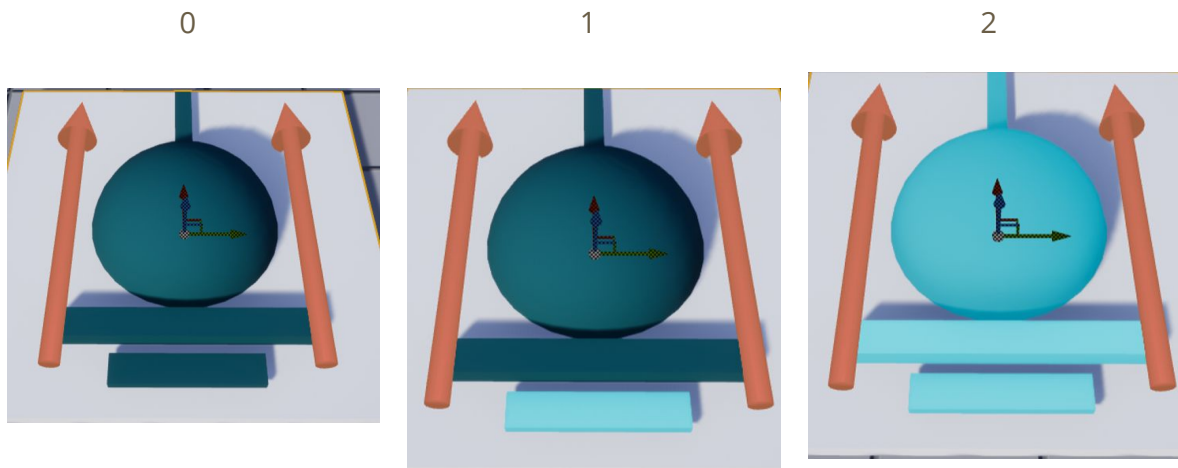


For more customization and functionality you can view the existing Blueprint modules to understand how they were made.

Logic Gates and Activation in Editor

Changing the Power Type and Active State of a Module will update all other modules after it while you are in the editor. This may act a little weird on the Logic Gates and does not look right. The reason for this are some limitations of the Unreal Engine Editor itself. To overcome this problem some of the logic gates have a "Custom Power Amount" Value.

Here is a short example based on the AND Gate:



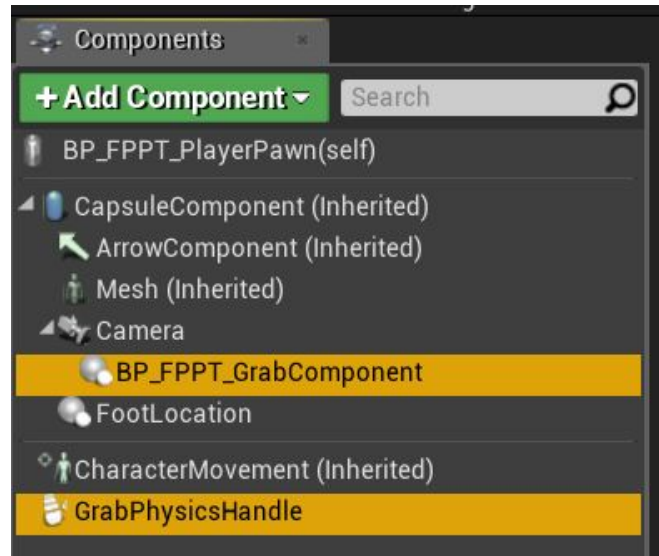
Integration for a custom Player Pawn

You may want to interact with the Puzzle System from your own Player Pawn or Controller. There is currently only 1 interaction that is implemented with the FPPT. The ability to move around a physics object, which in this case is the Power Cube.

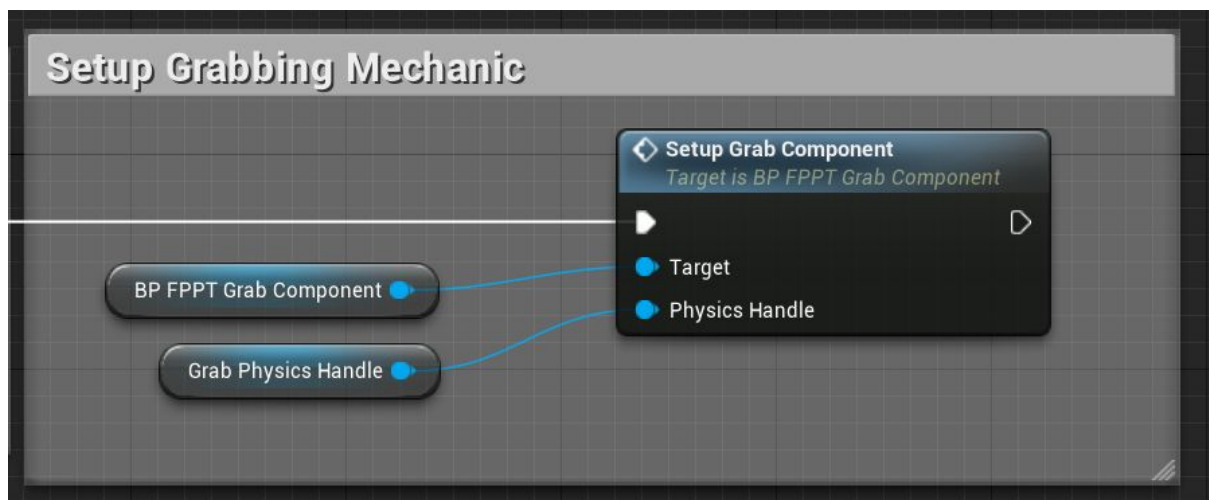
To integrate this functionality easier, the FPPT uses a custom Grab Component which you can attach to your custom Player Pawn.

First you need to add the **BP_FPPT_GrabComponent** to your Actor, as well as an **Physics Handle**.

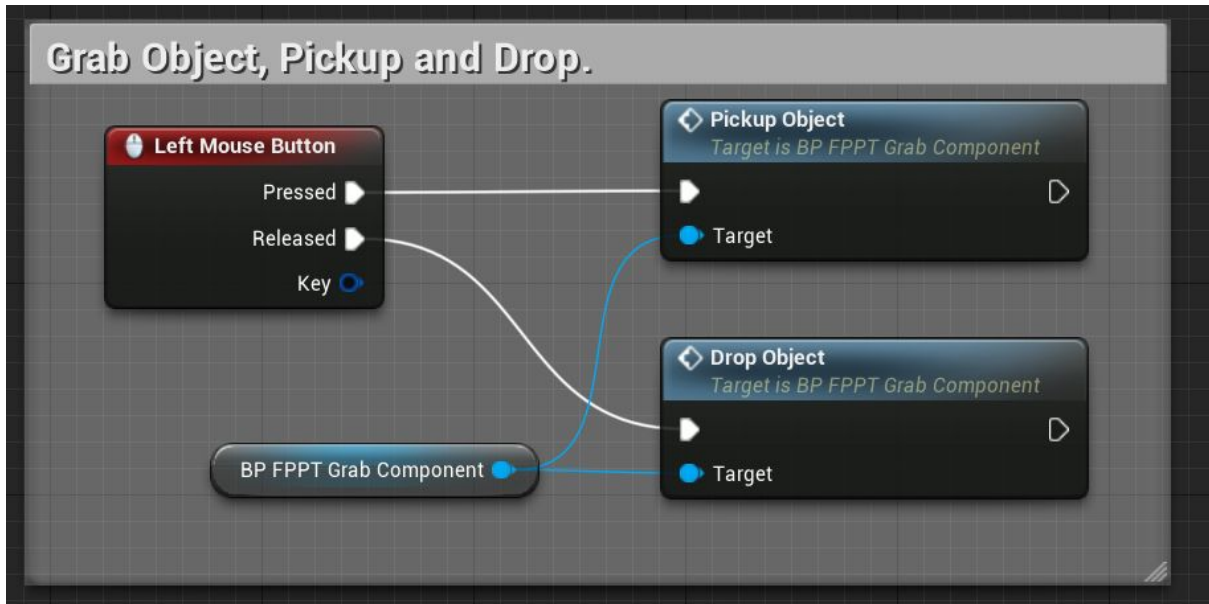
In the case of the default FPPT Player Pawn, the Grab Component has been placed inside the Camera. The forward direction of the Grab Component is the direction it uses to trace for grabbable Objects like the Power Cubes. So place it wherever you like.



Next you have to call the Setup Function during BeginPlay.

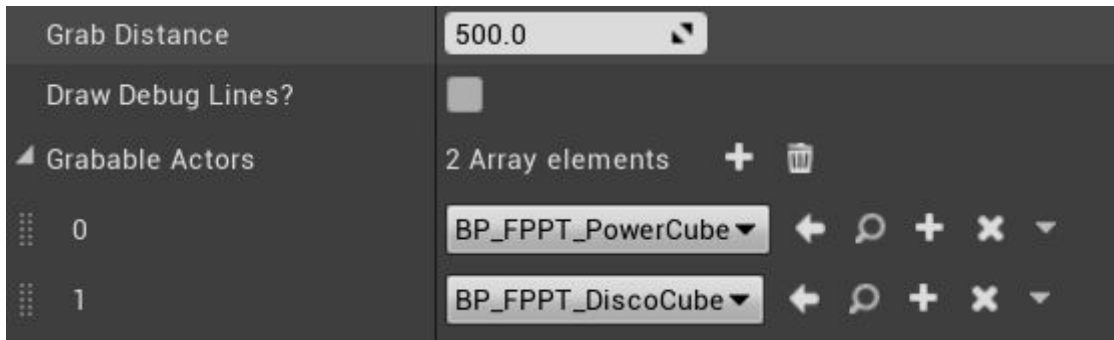


The last thing you have to add are the Pickup and Drop Functions from the Grab Component. In this case the Player should pickup the Cube with the Left Mouse Click.



That is all you need to implement the Physics Grabbing into your own Pawn.

There are also some additional Options on the Grab Component. You may want to increase your Grab Distance or enable other Objects to be picked up.

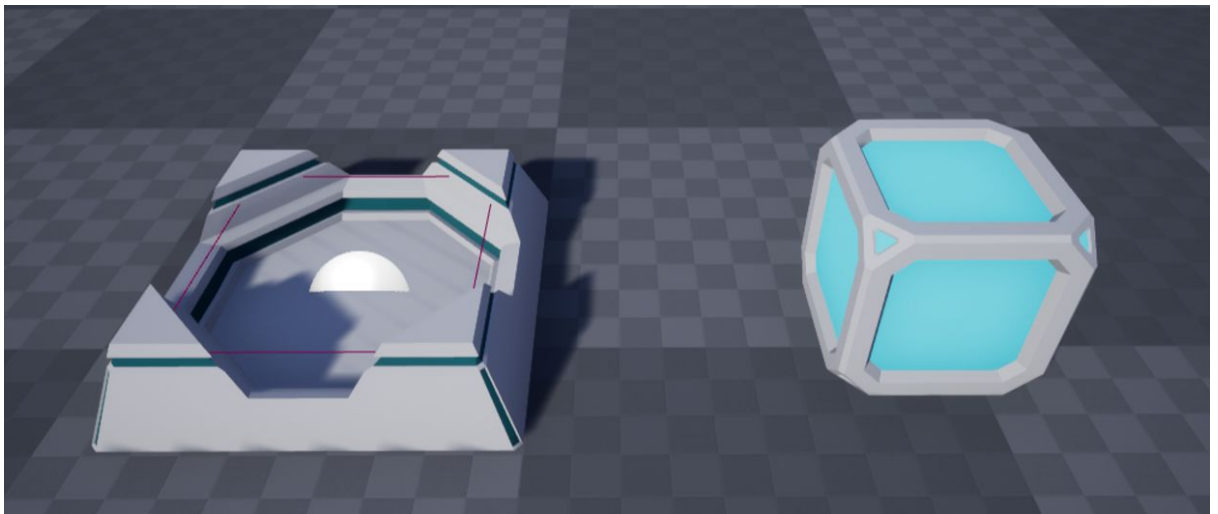


Modules

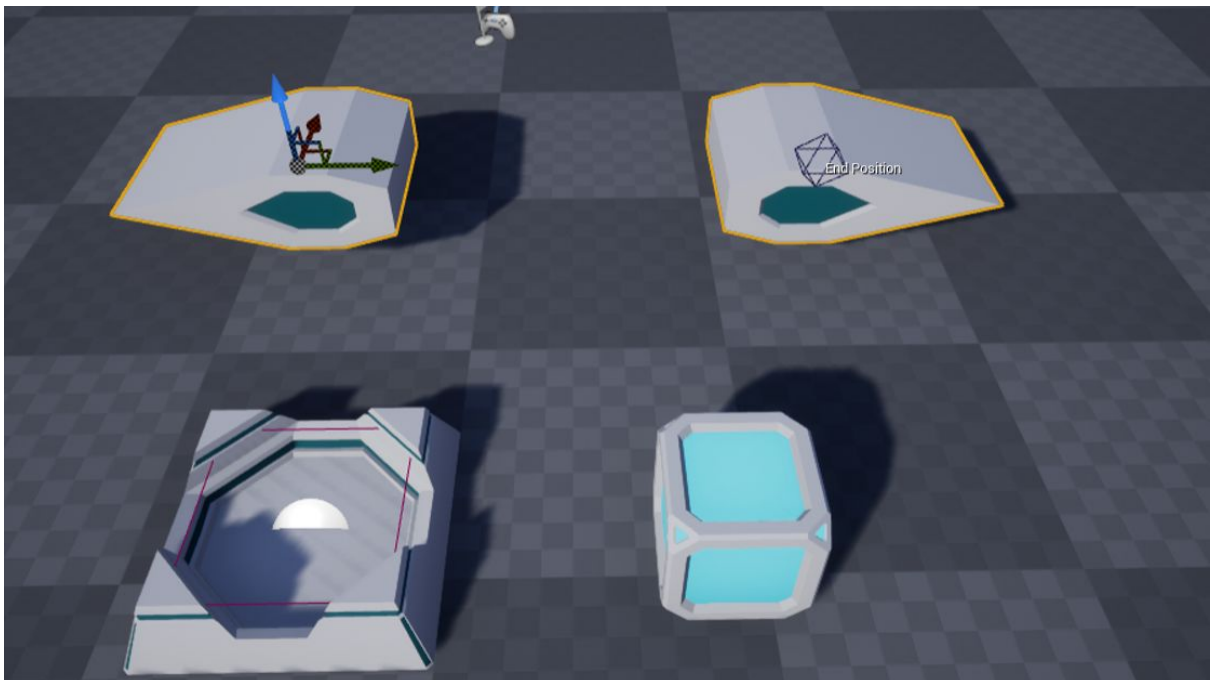
- Activation Modules
 - Power Cube
 - Power Switch
- Power Transport Modules
 - Power Line
 - Delayer Module
- Logic Modules
 - AND
 - OR
 - XOR
 - Inverter
 - Splitter
- Interactable Modules
 - Door
 - Bridge
 - Elevator
 - Moving Platforms
- Player Pawns
 - First Person Pawn

Getting Started

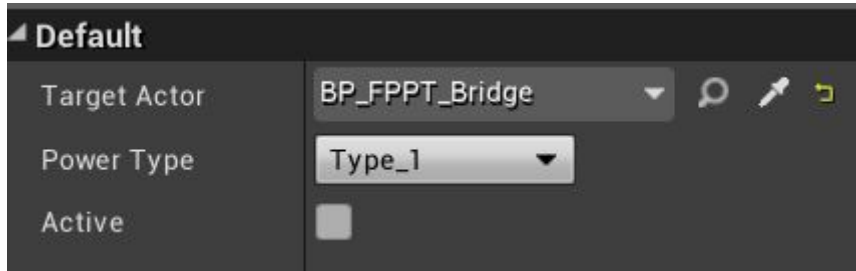
1. At first you need to place a PowerSwitch and a Power Cube inside your Level. These can be found under **FPPTemplate -> Blueprints -> BP_FPPT_PowerSwitch / BP_FPPT_Cube**



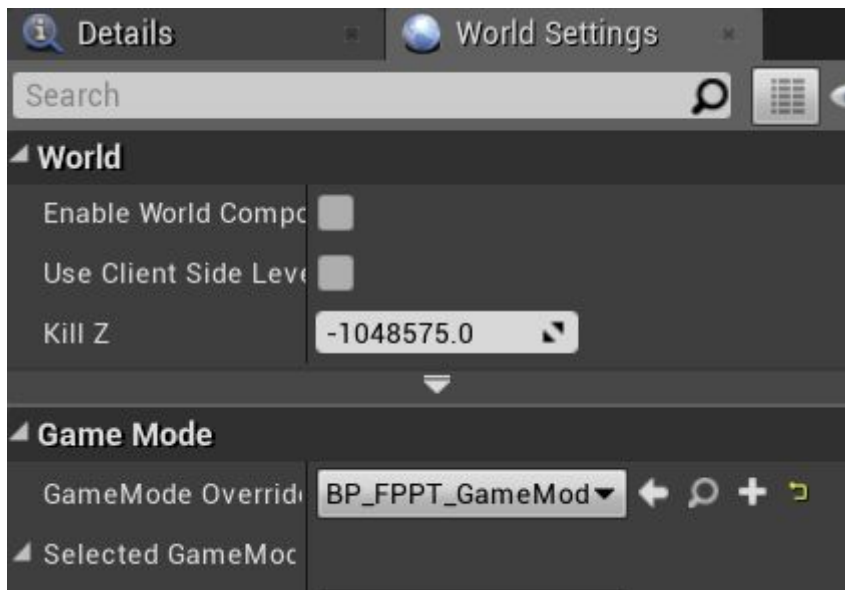
2. Now add an Interactable Module like the Bridge to the scene. You can find the Module in the Folder **FPPTemplate -> Blueprints -> Interactables -> BP_FPPT_Bridge**



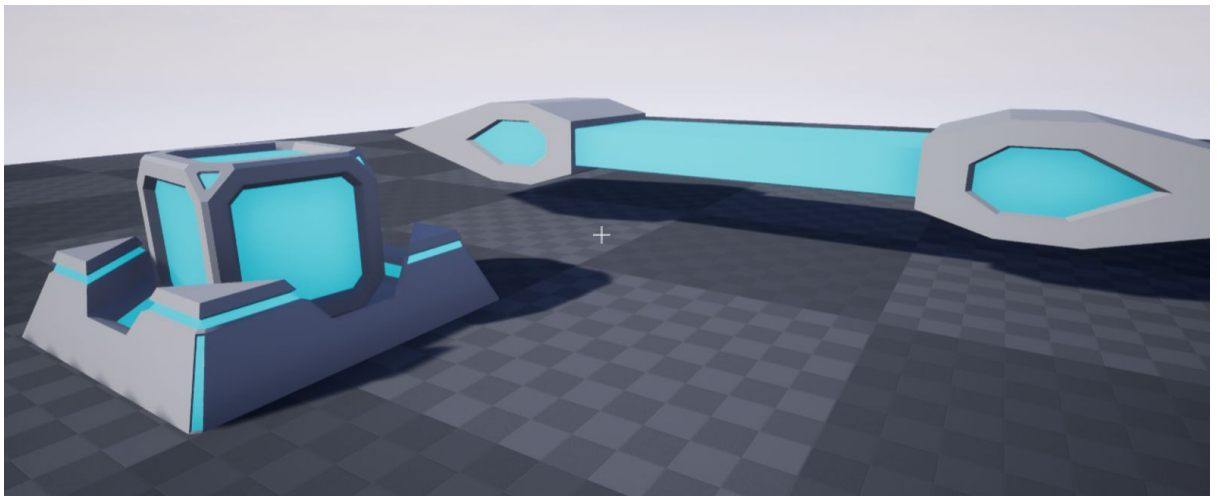
3. Connect the Power Switch to the Bridge by selecting the Target Actor of the Switch Module.



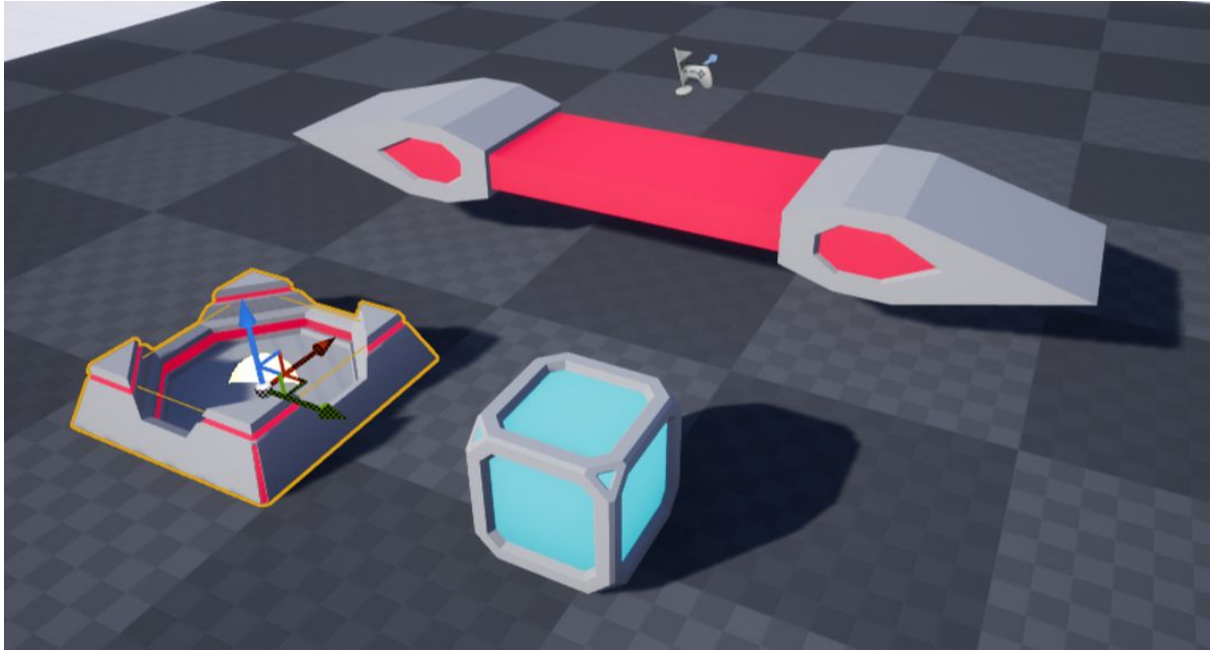
4. In **World Settings**, select the **GameMode BP_FPPT_GameMode** to add the First Person Pawn and try the Power System.



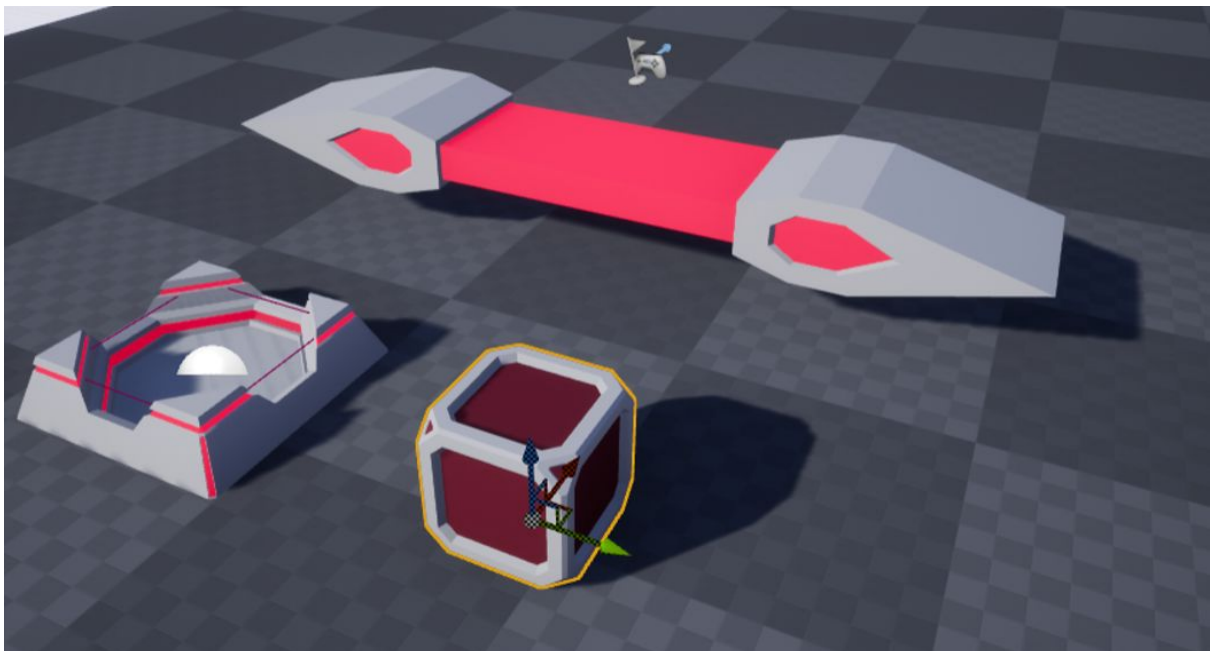
You can now play the level. Left click with your Mouse on the Power Cube and Place it inside the Switch. You will see the Modules lighting up.



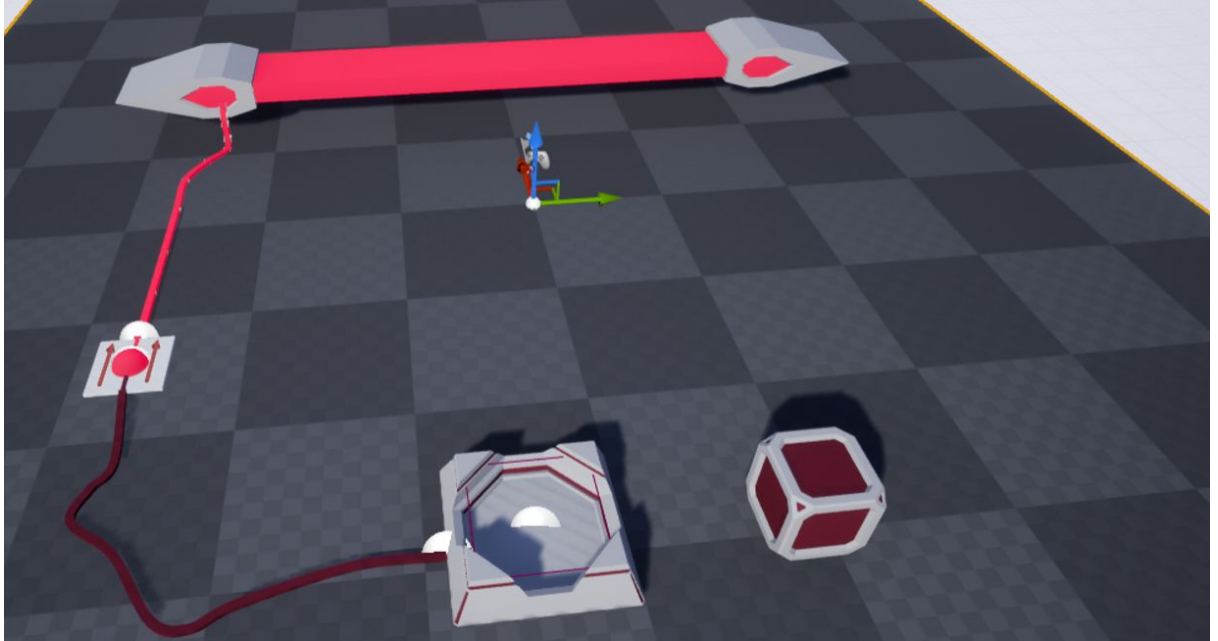
- Let's change the color of our Modules by selecting the Power Switch and changing the Power Type to Type_4 to make it red. You can also enable Active while in the editor to see how the modules will look when enabled.



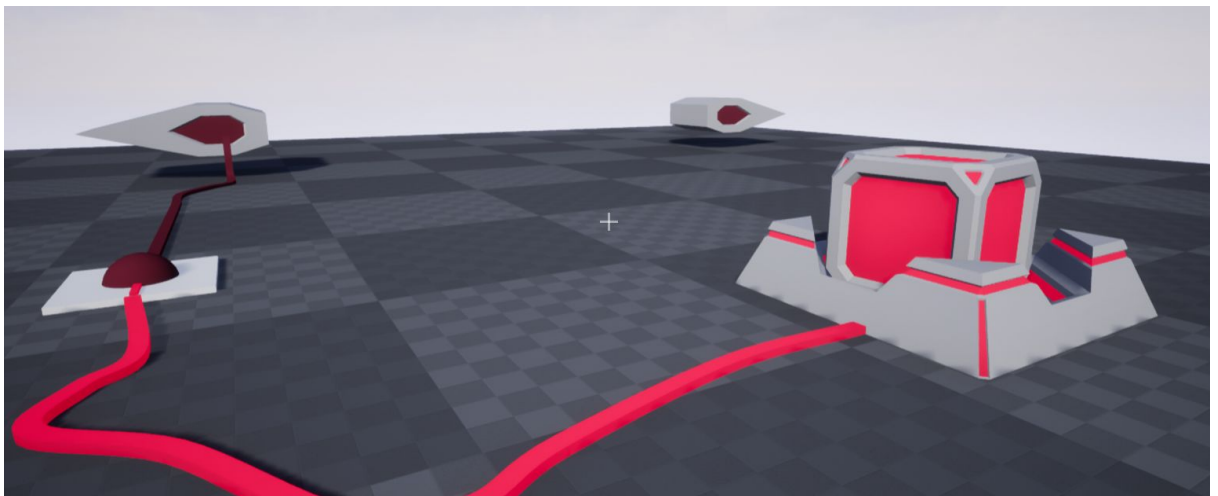
Since the cubes can be used on multiple switches, you will have to set its Power Type also to Type_4 to make it the same color. Different Power Types can't interact with each other, so you can't use blue cubes on red switches and vice versa. You can also disable a cube by ticking off the Active Bool. This will make the cube unusable on any Switch.



- Let's add some variation to this with more Modules. First we will add a Power Line Spline and connect it to an Inverter Module. Connect the inverter to a second Power Line and add the Bridge to it. Note: Because of limitations with the editor you may have to manually enable or disable Modules if there are Logic Gates involved.



- Play the game and test the Inverter out.



- You now know the basics of this asset. Go ahead and test out the other Modules and try to do some puzzles with them.

Something to note: Some Logic Modules use a Custom Power Amount Integer Value for activation. Use this instead of the Active Checkbox to make your Puzzle Logic ready in the Editor.