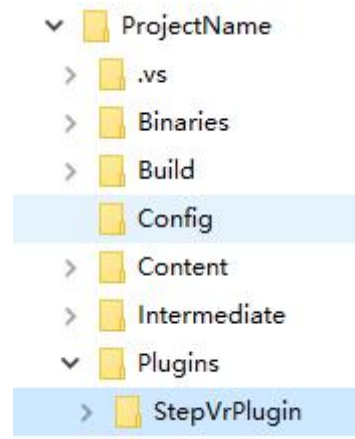


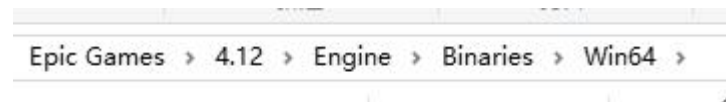
# UE4 插件说明

## 配置 Windows

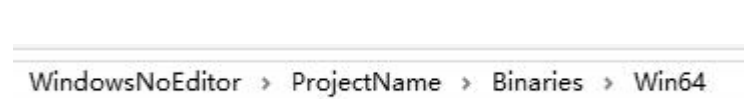
一 将解压后的 StepVrPlugin 文件夹 放到目标工程的 Plugins 目录下（如果没有 Plugins 文件夹，需要在 Content 同级目录下创建）



二 开发时将标定的矩阵文件 TransMat.txt 放到对应版本引擎的 Engine\Binaries\Win64 下（所有工程都可以使用）



项目打包后需将矩阵复制到 WindowsNoEditor\ProjectName\Binaries\Win64 下



## 使用范例

一 Camera 设置范例：

1 创建一个蓝图继承自 Pawn 并添加 StepVr 组件



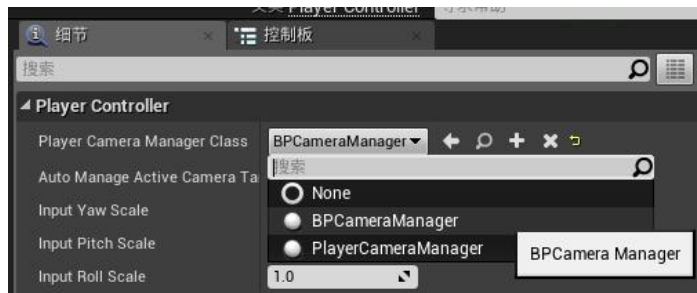
- 2 创建一个蓝图继承自 GameMode
- 3 创建一个蓝图继承自 PlayerController
- 4 创建一个蓝图继承自 CameraManager



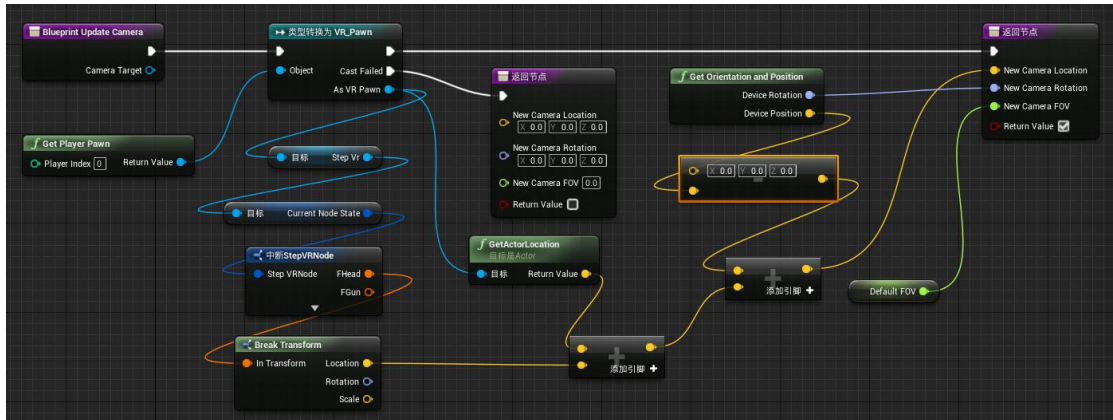
- 5 查看世界设置中 GameMode 属性栏  
设置 GameMode Override 为步骤 2 中创建的  
设置 PlayerController Class 为步骤 3 中创建的



- 6 设置 PlayerController 蓝图中 CameraManger Class, 选择步骤 3 中的 CameraManager 蓝  
图



- 7 设置 CameraManager 蓝图，刷新摄像机  
先覆盖函数 BlueprintUpdateCamera

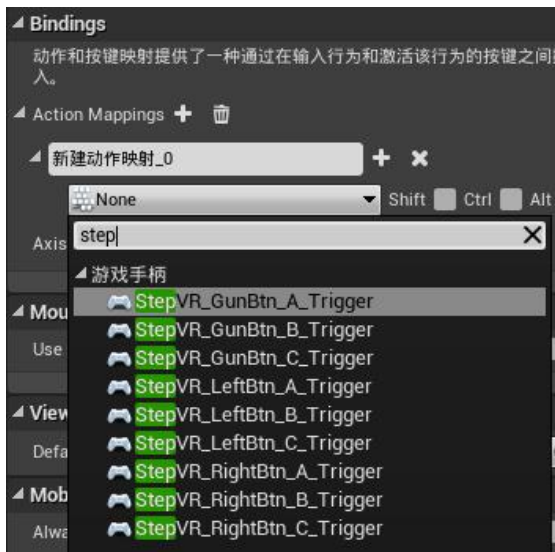


完成以上步骤，选择虚拟现实预览，并运行查看效果



## 二 获取按键事件

选择：设置→项目设置→引擎→输入→Bindings→Action Mappings,添加动作映射



## 三 使用 MotionController

在 Pawn 蓝图中添加组件 MotionController



运动控制器和 StepVr 设备的映射关系



左手柄: Left

右手柄: Right

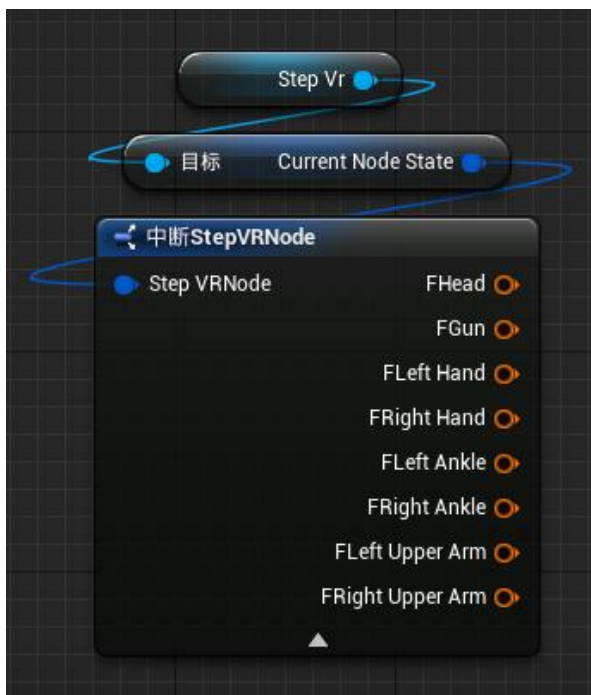
枪: Pad

#### 四 获取所有设备 TransForm 信息自定义使用方法

重复步骤 一.1(如过存在 StepVrComponent 组件, 跳过此步骤)

获取 CurrentNodeState 成员变量

查看 FStepVRNode 结构体, 获取相应的节点数据, 自行操作



## 注意事项（数据获取异常）

- 一 使用前需先插上 AP 设备,然后再从 VS 启动编辑器(拔插 AP 需要重新启动编辑器) 或者 开始独立游戏 , 否则获取不到数据,甚至可能会引起崩溃
- 二 查看设备 ID 号是否对应
- 三 矩阵文件是否放在正确的位置 (查看配置 Windows)

## 当前插件设备对应 ID

左手柄 ID: 1  
右手柄 ID: 2  
枪 ID: 4  
头 ID: 6