虚拟摇杆：ETC Joystick组件

JoystickName：摇杆名字

Activated：不管这个虚拟摇杆是否被启用，当这个选项不打勾，则这个摇杆游戏对象都是可见的

Visible : 显示或不显示虚拟摇杆

Use Fixed Update : 是否启用固定频率的更新

Unregister at disabling time : 当摇杆在禁用的时候是否需要注销,建议保持默认值

ETC Joystick.Position & Size（设置摇杆在UI中的位置）

Type：定义摇杆的类型（静态/动态），建议保持默认值

Anchor:摇杆在UI中的位置

No offset thumb:勾选了，那么手指在按下摇杆的那一下，只要是在灰色区域内，那么摇杆会自动向着你手指的位置移动（即使你的手指在按下去之后没有滑动），如果不勾选，那么手指在按到灰色区域以内并且有滑动，摇杆才会移动

No return of the thumb:玩家松开虚拟摇杆的时候，虚拟摇杆是否会自动复位到原来位置

Background size:摇杆的背景大小

Thumb size:摇杆的大小

Radius based on:虚拟摇杆能够移动的区域直径，以虚拟摇杆的背景大小为基准

当Type选择为动态的情况：

AllowOverTouched：勾选上了，Joystick area才会发挥作用

Joystick area：当Type为动态的时候，虚拟摇杆默认是在UI界面中隐藏的，当手指触摸到屏幕的时候，在屏幕的哪部分显示虚拟摇杆，如果Joystick area选为UserDefine，则用户自定义摇杆显示区域，此区域游戏对象要右键-EasyTouchControls-Area来创建。当然，要想Joystick area这选项发挥作用，则AllowOverTouched必须勾选上

摇杆自定义区域：ETCArea

创建：Hierarchy面板右键==>EasyTouchControls==>Area

Show at runtime：是否在游戏运行的时候显示这块摇杆显示的自定义区域

Preset：锚点的选择

ETC Joystick.Axes properties:

Enable Unity axes:是否让虚拟摇杆与Unity的轴输入起关联

Allow Unity axes on standalone:在虚拟摇杆之外，允许Unity的轴单独作用

Force visible：显示力（暂时不明有何作用，建议保持默认勾选）

Turn & Move direction Action:是否启用虚拟摇杆控制角色旋转和移动

Auto link on tag：此功能控制的物体所在的标签是什么

Move curve：要以那种线性规律进行移动操作

Move speed：要以多大的速度移动

Intial rotation：初始的旋转角度为多少

Lock in jump：在跳跃的过程中是否需要锁住游戏物体不让他进行移动和旋转

Gravity：因为要模拟现实的情况，所以可以设置摩擦力，而摩擦力则与重力有关，所以这里设置的是重力的大小

Horizontal axis / Vertical axis:是否启用EasyTouch的水平/垂直轴，这里可以设置EasyTouch的轴的名字，但是切记，轴的名字在一个场景中必须是唯一的

React on:什么情况下，我们的轴才有输入

Dead zone method：当手指触碰到摇杆的不灵敏区域的时候（例如摇杆在摆幅过小的情况），那么应该对输入自动进行怎样的处理，Classical是默认处理，Curve则是自定义的曲线处理

Dead length：这个值决定了从中心开始每个轴向上多远的距离内是无效区域（不灵敏区域），即每个轴向上移动距离若少于最远距离的这个比例则该轴向上是没有返回值的。（建议保留默认值）

Inverted axis：是否需要将摇杆的输入取反，例如跑跑卡丁车里面，中了反转效果的时候，你的输入全部取反了

On/Off Threshold：开启关闭阈值（与down触发事件相关，可参考文章下面down events部分）

Speed：这个数值直接用于操作摇杆移动的时候所控制物体的运动速度，这个数值可以通过ETCInput.GetAxisSpeed接口获得（关于接口的详细信息可以参考ETCInput\_API.PDF文档）。

Auto link on tag：摇杆控制哪个标签的游戏对象，当游戏对象是从prefab拉出的时候要必须勾上，不然的话摇杆是控制不了这游戏对象的

Action：是要控制这个游戏对象进行什么操作

Affected axis：控制这个游戏对象的哪个轴，进行Action的操作

Gravity：重力

Enable inertia:是否开启惯性

Inertia：惯性的大小

Threshold：惯性的阀值

当Action选择为Jump，Relative Torque，Torque，RelativeForce，Force，Scale，Rotat，Translate Local的时候Gravity-Inertia-smoothing...选项卡只有Enable inertia

当Action为Translate的时候，Direction action（optional）选项卡会多出Lock in jump选项

Lock in jump：是否在移动的过程中不准许跳跃

当Action为RotateLocal的时候，Gravity-Inertia-smoothing...选项卡会多出Automatic stabilization和Clamp rotation选项

Automatic stabilization:当这个功能开启后，当玩家松开摇杆，物体会自动复原到原来的转角

Speed：自动复原的速度

Threshold：活动中的轴的值要大于这个阀值才进行自动复原

Clamp rotation：当这个选项被勾选上，则可以设置最大旋偏转角和最小偏转角

Max angle：最大旋转角（+）

Min angle：最小旋转角（-）

Unity axes：EasyTouch的轴与Unity的哪个轴进行绑定

ETC Joystick.Camera:

Enable tracking:勾选上，则将摄像机交给EasyTouch来控制

Auto link on tag:勾选上，则EasyTouch自动对所设置的tag所对应的摄像机进行控制，不勾选则手动选择摄像机游戏对象

Target mode:摄像机所要跟随的游戏对象

Camera mode：设置普通跟随还是平滑跟随（插值跟随）

Wall detection：是否设置墙壁，因为当游戏对象碰到墙壁的时候，摄像机可能会穿墙。当着选项被勾选上，游戏对象碰到墙壁的时候，EasyTouch会自动调整摄像机的位置

Wall Layer：哪个一层的游戏对象算作墙壁

Distance/Height/Height damping/Rotation dampping：这些选项都是用来设置摄像机相对于其跟随的游戏对象的位置

ETC Joystick.Sprite:

用来设置摇杆背景图和摇杆的图片

ETC Joystick.MoveEvents:

On Move Start：当用户第一次移动轴的时候调用（当返回到0点位置时触发时机将会被重置），也就是说，刚开始移动的那一瞬间调用

On Move（Vector2）：移动摇杆的过程或者摇杆thumb键没有回到原点时将会持续触发，将摇杆每个轴上移动的距离（值范围为-1~1）作为方法参数传入。

On Move Speed（Vector2）：移动摇杆的过程或者摇杆thumb键没有回到原点时将会持续触发，将摇杆每个轴上axisValue \* axisSensitivity\*Time.deltaTime作为方法参数传入。

On Move End：当摇杆thumb键回到原点或者用户释放摇杆时触发

ETC Joystick.TouchEvents:

On Touch Start：当用户刚触摸摇杆的那一瞬间调用

On Touch Up：当用户释放摇杆的时候调用

ETC Joystick.Down Events:

On Down Up：用户第一次按下摇杆并向前移动时触发（当摇杆thumb键回到原点或摇杆位置在对应轴threshold值范围内时重置该函数触发条件）

On Down Right：用户第一次按下摇杆并向右移动时触发（当摇杆thumb键回到原点或摇杆位置在对应轴threshold值范围内时重置该函数触发条件）

On Down Down：用户第一次按下摇杆并向后移动时触发（当摇杆thumb键回到原点或摇杆位置在对应轴threshold值范围内时重置该函数触发条件）

On Down Left：用户第一次按下摇杆并向左移动时触发（当摇杆thumb键回到原点或摇杆位置在对应轴threshold值范围内时重置该函数触发条件）

ETC Joystick.Press Events:

On Press Up：按下摇杆并向上移动时触发

On Press Right：按下摇杆并向右移动时触发

On Press Down：按下摇杆并向下移动时触发

On Left Right：按下摇杆并向左移动时触发

ETC Button:

Button name:Button的名字

Activated：是否被激活

Visible：是否需要隐藏不显示

Use fixed Update：是否需要开启固定帧来做检测

Unregister at disabling time : 当摇杆在禁用的时候是否需要注销,建议保持默认值

ETC Button.Position & Size：

Anchor：按钮位于UI的位置

Offset：在Anchor的基础上的偏移

Size：按钮要多大

ETC Button.Behaviour：

Enable Unity axes：是否与Unity的轴起关联

Allow Unity axes on Standalone：是否允许Unity轴单独作用

Force visible：是否显示力（暂不明确功用，建议保留默认值）

Swipe in：：意思是，当你手指刚开始的触摸并不是这个按钮，但是手指不断的贴着屏幕滑动，当滑动到这按钮的时候，按钮会被自动视作被按下

Swipe out：意思是，当你手指刚开始的触摸是这个按钮，但是手指不断的贴着屏幕滑动，当滑动到这按钮之外的时候，按钮仍然会被视作被按下

Value over the time：Value是否开启渐进加速（如果不开启此选项，则被控制物体会立刻达到移动速度或者转速的最大值），如果开启，就会多出以下两个选项，并且，物体的最终速度由Max calue和Value共同作用，所以在这里建议把Max value和Value设置为一模一样的

Step：轴的值按多大数值递增

Max value：轴能够递增到多大的绝对值

Value：此值是直接作用于Direction ation选项卡中的Action选项的，并且在脚本中可以用ETCInput.GetButtonValue的方法得到返回值(详情请查看ETCInput\_API.PDF)，如果Value over the time勾选了，那么ETCInput.GetButtonValue所得到的值是递增的并且最大值是Max value

Auto link on tag:按钮要控制哪个游戏对象

Action on：按钮要怎被怎样按下才会被触发

Action：按钮要控制这个游戏对象具体做啥

Gravity：重力

Unity axes：按钮要绑定Unity的哪个轴

ETC Button.Sprite：设置按钮图和按钮背景图

ETC Button.Event：

On Down：按钮被按下的那一刹那触发

On Pressed：按钮被按住的时候触发

On Pressed Value（Single）：按钮被按住的时候触发

On Up：按钮被松开的时候触发

ETC Touch Pad：

TouchPad name ：名字

Visible at runtime ：在项目运行的时候是否显示触摸垫的触摸范围

Use Fixed Update : 是否启用固定频率的更新

Unregister at disabling time : 当触摸垫在禁用的时候是否需要注销,建议保持默认值

ETC Touch Pad.Position & Size：

Anchor：按钮位于UI的位置

Offset：在Anchor的基础上的偏移

Width：触摸垫的宽

Height：触摸垫的高

ETC Touch Pad.Axes：

Enable key simulation：是否启用模拟键

Allow simulation on standalone：是否允许模拟键独自作用

DPI：开启了这个选项，触摸垫会把屏幕的分辨率也考虑进去，因此不管屏幕大小怎么变化，触摸垫所返回的数据都是相同的

Swipe in：：意思是，当你手指刚开始的触摸并不是这个触摸垫的区域，但是手指不断的贴着屏幕滑动，当滑动到这触摸垫区域的时候，触摸垫会自动认为你触摸到自己，进而工作

Swipe out：意思是，当你手指刚开始的触摸是在这个触摸垫的区域内，但是手指不断的贴着屏幕滑动，当滑动到这触摸垫区域外的时候，只要你的手指仍然紧贴屏幕没松开，那么触摸垫仍然认为你触摸到了自己，进而仍然继续工作

Horizontal Axis / Vertical Axis：里面的选项属性解释均与ETC Joystick.Axes properties中的对应属性选项的解释相同，不同的地方只有一处：

Sensitivity：触摸垫灵敏度

ETC Touch Pad.Sprites：触摸垫的图片

ETC Touch Pad.Move Evnets：

On Move Start（）：当用户的手指在触摸垫上刚开始滑动的那一刻时调用

On Move（Vector2）：当用户的手指在触摸垫上持续滑动时调用，并且会返回一个你手指在触摸垫上的位置信息

On Move Speed（Vector2）：和On Move（Vector2）大致相同，但是返回的值是手指的滑动速度（轴值 \* 触摸垫灵敏度 \* Time.deltatime）

On Move End：当轴回到0，或到用户手指离开触摸垫时调用

ETC Touch Pad.Touch Evnets：

On Touch Start（）：当手指刚触碰到触摸垫的那一刹那调用

On Touch Up（）：当手指离开触摸垫的那一刹那调用

ETC Touch Pad.Down Evnets：

On Down Up：手指在触摸到触摸垫后，持续向触摸垫上方匀速移动的那一刹那调用

On Down Down：手指在触摸到触摸垫后，持续向触摸垫下方匀速移动的那一刹那调用

On Down Right：手指在触摸到触摸垫后，持续向触摸垫右方匀速移动的那一刹那调用

On Down Left：手指在触摸到触摸垫后，持续向触摸垫左方匀速移动的那一刹那调用

ETC Touch Pad.Press Evnets：

On Press Up：手指在触摸到触摸垫后，向触摸垫上方持续移动时被调用

On Press Down：手指在触摸到触摸垫后，向触摸垫下方持续移动时被调用

On Press Right：手指在触摸到触摸垫后，向触摸垫右方持续移动时被调用

On Press Left：手指在触摸到触摸垫后，向触摸垫左方持续移动时被调用

ETCD Pad:

Activated：是否被激活

Visible：是否需要隐藏不显示

Use fixed Update：是否需要开启固定帧来做检测

Unregister at disabling time : 当D型触摸垫在禁用的时候是否需要注销,建议保持默认值

ETCD Pad.Position & Size：

Anchor：按钮位于UI的位置

Offset：在Anchor的基础上的偏移

Size：D型触摸垫要多大

Button size coef：按钮大小系数，暂不明确功能，建议保留默认值

ETCD Pad.Axes：

Enable key simulation：是否启用模拟键

Allow simulation on standalone：是否允许模拟键独自作用

Force visible：显示力（暂不明确功能，建议保留默认值）

Axes count：设置由D型触摸垫所管理的轴数,如果是两条轴，则当手指在D型触发垫规规矩矩的上下左右的UI上才能有反应，如果是四条轴，则手指除了在D型触发垫的上下左右外，还有左上，左下，右上，右下能够让D型触发垫有反应

Horizontal axis / Vertical axis的选项与ETC Joystick中的对应选项的解释一样，此处不做累述

ETCD Pad.Axes：设置D型触发垫的图片

ETCD Pad.Move Event：与ETC Touch Pad对应选项的解释雷同此处不做累述

ETCD Pad.Touch Event：与ETC Touch Pad对应选项的解释雷同此处不做累述

ETCD Pad.Down Event：与ETC Touch Pad对应选项的解释雷同此处不做累述

ETCD Pad.Press Event：与ETC Touch Pad对应选项的解释雷同此处不做累述

InputManager：如果要让以上虚拟杆起作用，并且把检测到的所有数据提供给你的脚本调用的话，场景中必须要有InputManager这个脚本所挂的游戏对象在，当然，如果你忘记为你的场景加上这么个游戏对象，当你的游戏运行的时候，EasyTouch会为你自动加上