Realistic Car Controller

First, thank you for purchasing and using Realistic Car Controller!

Contents

Realistic Car Controller 1

Contents 1

First to Do! 2

Script Execution Order 3

Overview 3

RCC_CarControllerV4.cs 4

RCC Settings 6

Configurable Ground Materials 9

RCC Scene Manager 10

Controller Types 10

Mobile Controller 11

About Mobile Use On City Scene 12

RCC Camera 13

Record / Replay 16

Customization 17

How The Customization Panel Works 17

Credits 19

License 19

You can find more updated details on

http://www.bonecrackergames.com/realistic-car-controller

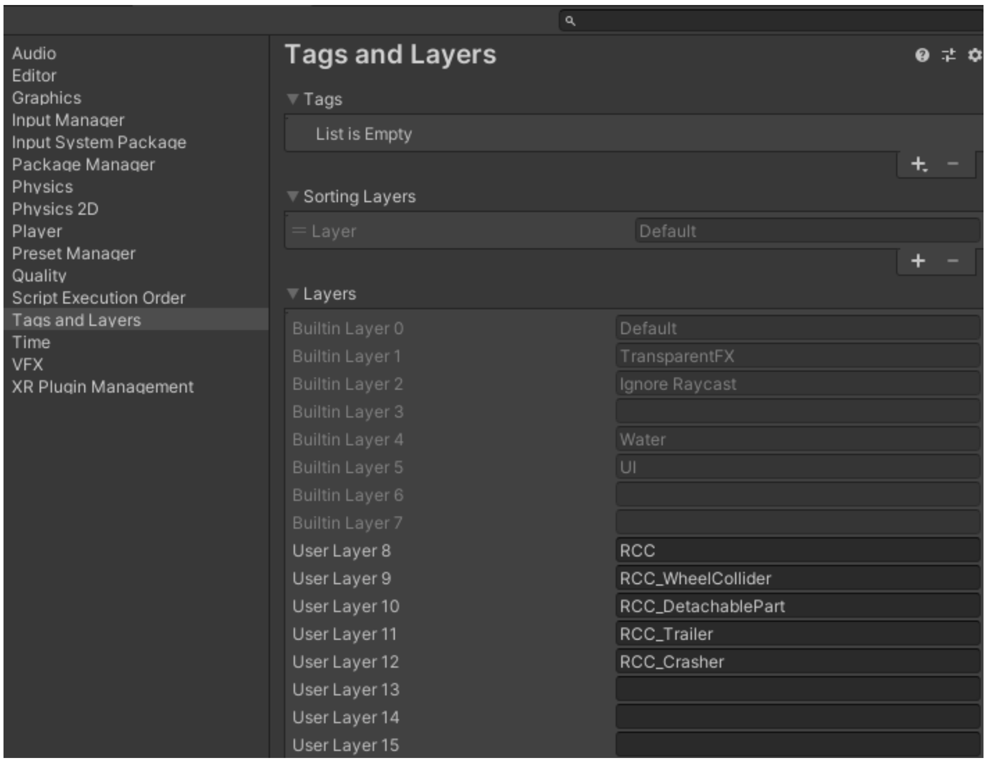
https://www.youtube.com/playlist?list=PLRXTqAVrLDpoW58lKf8XA1AWD6kDkoKb1

(You can zoom in with CTRL + ScrollUp for enlarge PDF pages)

First to Do!

Always backup your project before updating any asset or Unity Editor. Keep your own assets outside of the RealisticCarControllerV4 folder. Delete the entire folder and import the updated version.

RCC uses LayerMask to avoid unwanted raycast hits. Necessary layers will be created automatically, but feel free to check them after the import. These layers must be selected in the RCC Settings. Also, you can import it from the Welcome Screen, but it will overwrite your **Tags & Layers**.



Tags and Layers

Tags

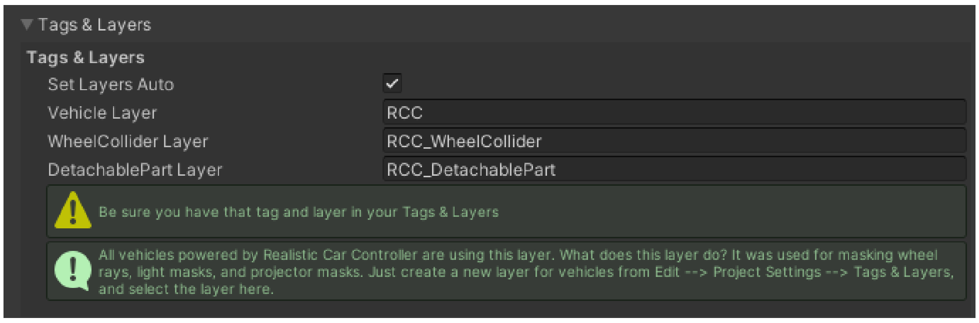
List is Empty

Sorting Layers

Layer	Default

Layers

Builtin Layer 0	Default
Builtin Layer 1	TransparentFX
Builtin Layer 2	Ignore Raycast
Builtin Layer 3	
Builtin Layer 4	Water
Builtin Layer 5	UI
Builtin Layer 6	
Builtin Layer 7	
User Layer 8	RCC
User Layer 9	RCC_WheelCollider
User Layer 10	RCC_DetachablePart
User Layer 11	RCC_Trailer
User Layer 12	RCC_Crasher
User Layer 13	
User Layer 14	
User Layer 15	



Tags & Layers

Tags & Layers

Set Layers Auto	✓
Vehicle Layer	RCC
WheelCollider Layer	RCC_WheelCollider
DetachablePart Layer	RCC_DetachablePart

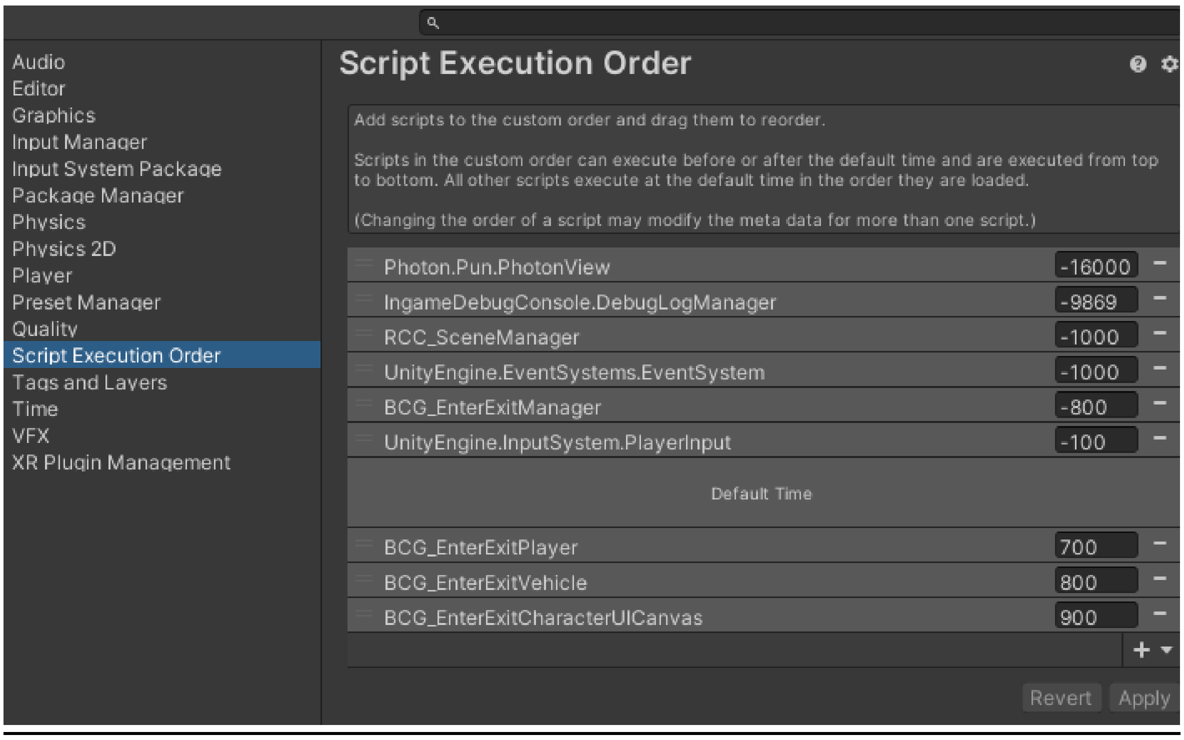
Be sure you have that tag and layer in your Tags & Layers

All vehicles powered by Realistic Car Controller are using this layer. What does this layer do? It was used for masking wheel rays, light masks, and projector masks. Just create a new layer for vehicles from Edit --> Project Settings --> Tags & Layers, and select the layer here.

(Tools → BoneCracker Games → Realistic Car Controller → RCC Settings)

Script Execution Order

RCC is using **Script Execution Order** to avoid unexpected event conflicts. This should be imported successfully when RCC installed and doesn't require any action. Just make sure you have this order. You can check it in **Edit → Project Settings → Script Execution Order**.



Script Execution Order

Add scripts to the custom order and drag them to reorder.

Scripts in the custom order can execute before or after the default time and are executed from top to bottom. All other scripts execute at the default time in the order they are loaded.

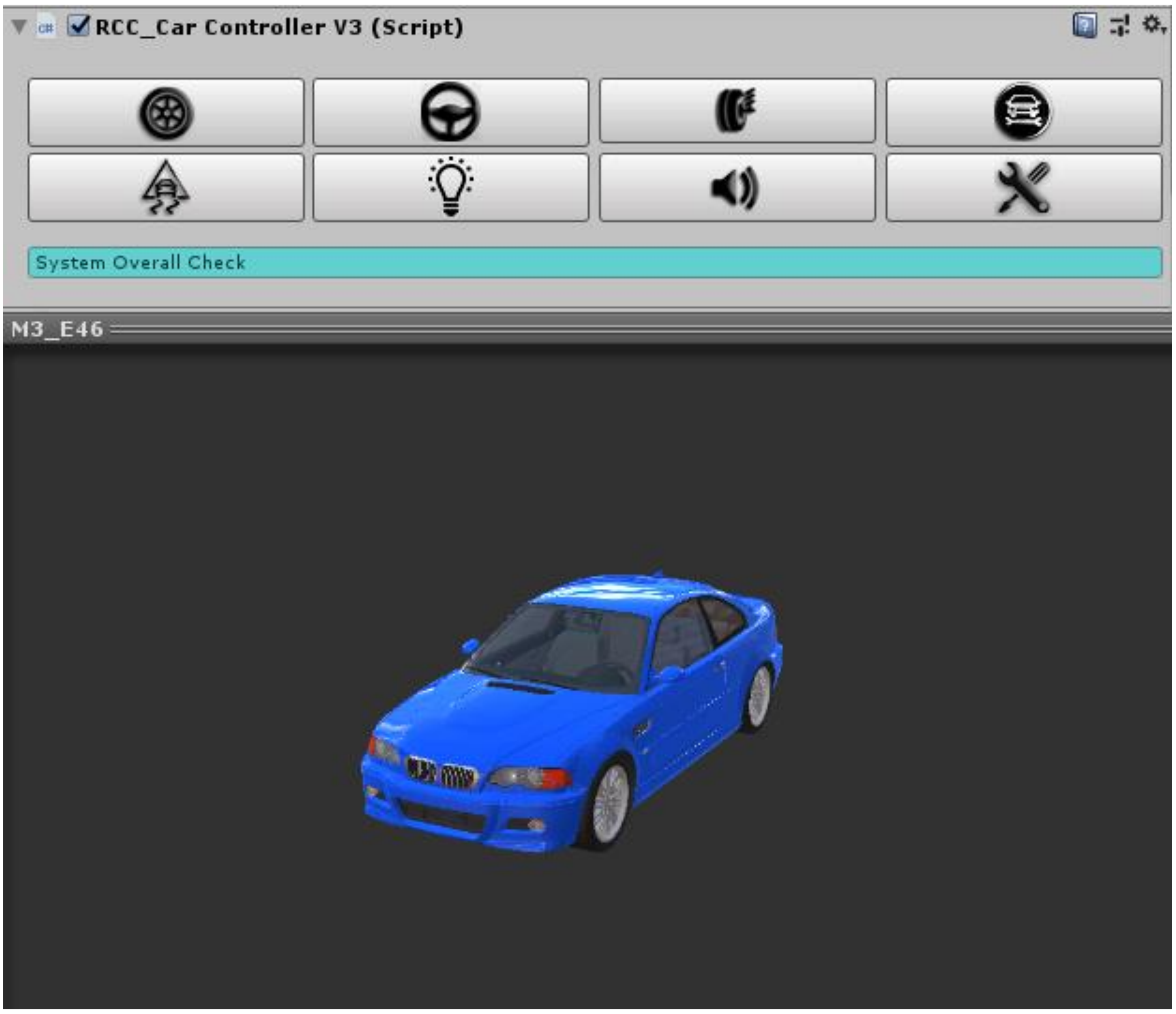
(Changing the order of a script may modify the meta data for more than one script.)

Photon.Pun.PhotonView	-16000
IngameDebugConsole.DebugLogManager	-9869
RCC_SceneManager	-1000
UnityEngine.EventSystems.EventSystem	-1000
BCG_EnterExitManager	-800
UnityEngine.InputSystem.PlayerInput	-100
Default Time	
BCG_EnterExitPlayer	700
BCG_EnterExitVehicle	800
BCG_EnterExitCharacterUICanvas	900

Overview

Each vehicle has it's own **RCC_CarControllerV3.cs** script. Each vehicle is responsible for own **RCC_CarControllerV3.cs**. All global shared settings are in the **RCC Settings** (**Tools → BCG → RCC → Edit Settings**). Lights, cameras, and exhausts are addons and not required as an essential. Inputs are processed by the **RCC_InputManager.cs** script. It will receive corresponding inputs from the selected device. **RCC_SceneManager.cs** is managing active player vehicle, other vehicles, AI vehicles, record/replay, UI canvases, etc. All other main topics can be found below.

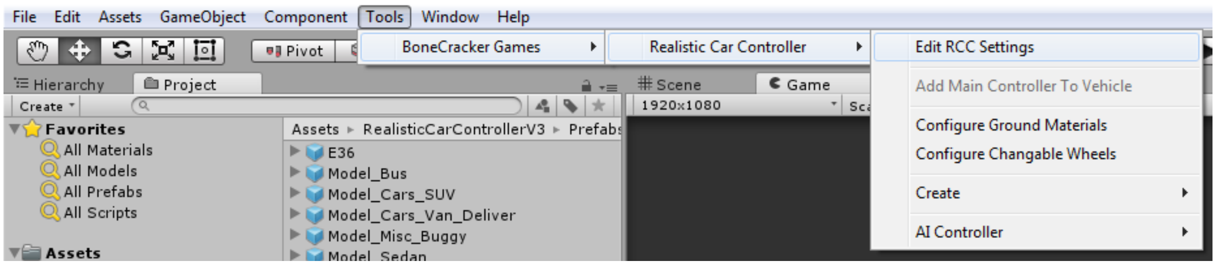
RCC_CarControllerV4.cs



8 Main Categories for easily and understandable creating / configurating vehicles.

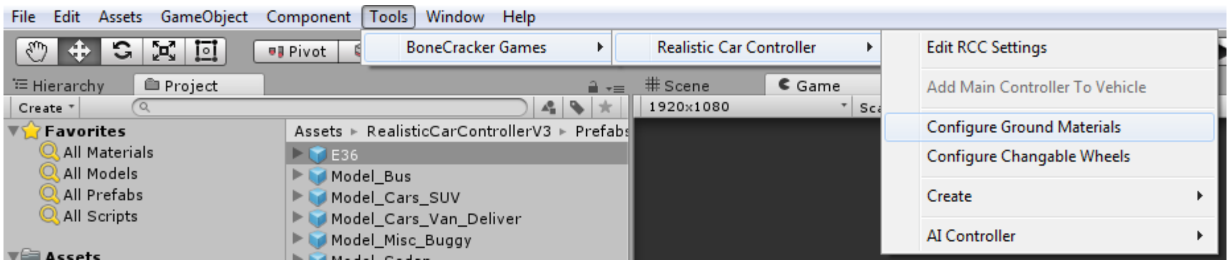
Wheels, **Steering**, **Suspensions**, **Mechanic Configuration**, **Stability**, **Lights**, **Sounds**, and **Damage**.

All vehicles are sharing global settings, sounds, and configurations via **RCC Settings**.

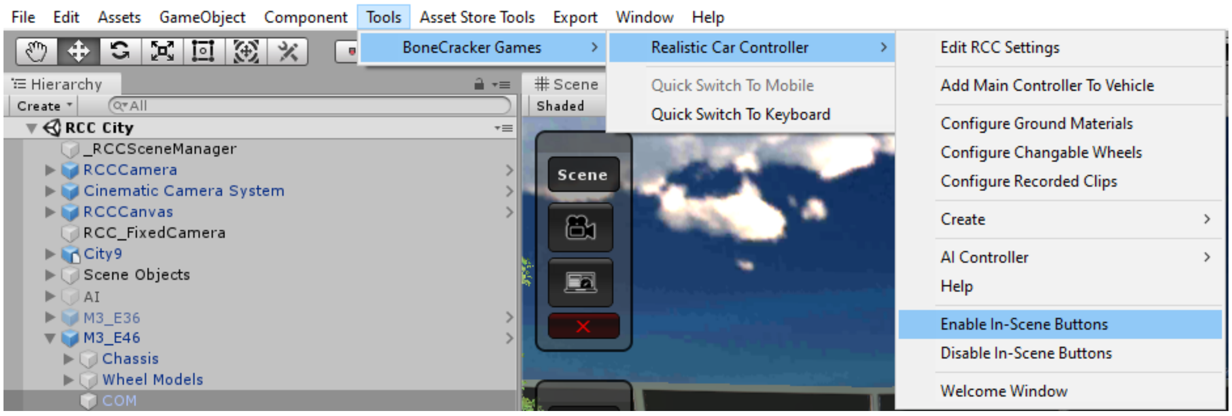


Creating new vehicles have been explained in documentation named "**Realistic Car Controller How to Create New Vehicles**."

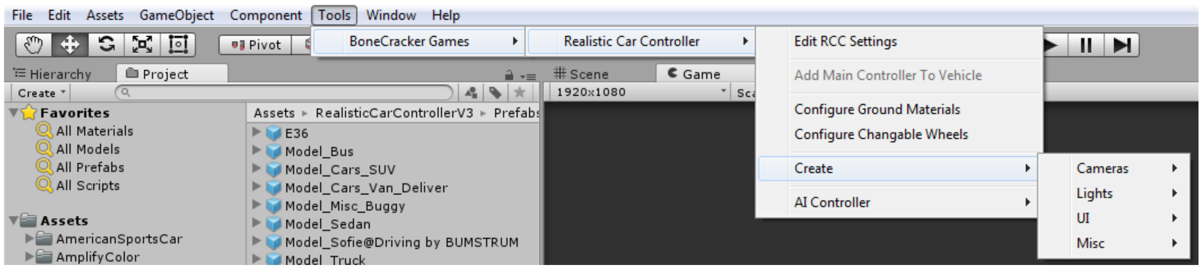
Changing ground materials physics, particles, sounds, etc. in the **Tools → BoneCracker Games → Realistic Car Controller → Configure Ground Materials**. (Detailed explanation in the documentation named "**Realistic Car Controller RCC_GroundMaterials**")



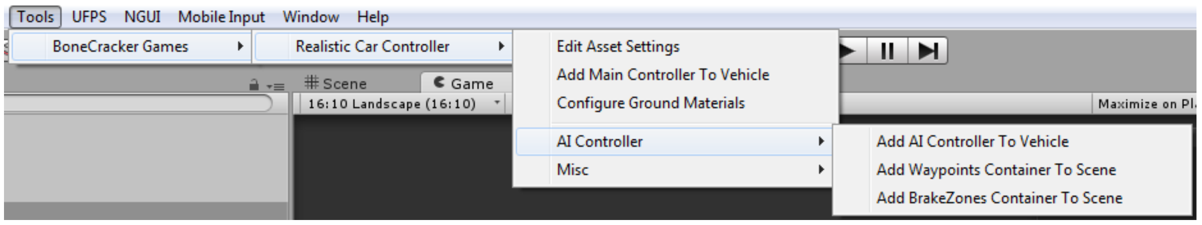
You may want to enable In-Scene buttons to create addons with fastest way. **Tools → BoneCracker Games → Realistic Car Controller → Enable In-Scene Buttons**. (Detailed explanation in the documentation named "**Realistic Car Controller How to Create New Vehicles**")



Creating lights, exhausts, mirrors, cameras, etc. in the **Tools → BoneCracker Games → Realistic Car Controller → Create**. (Detailed explanation in the documentation named "**Realistic Car Controller How to Create New Vehicles**")

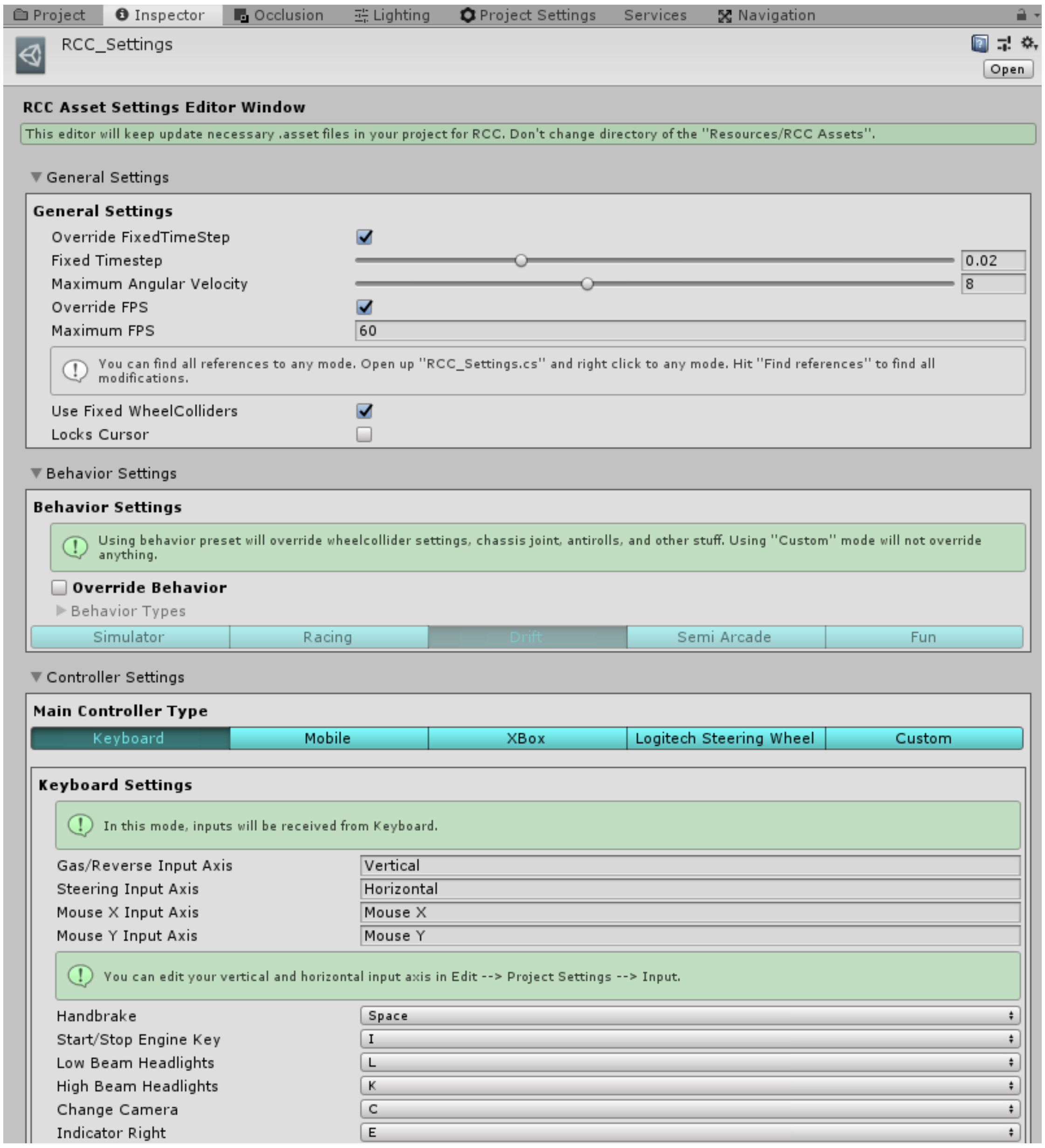


Making vehicles controlled by AI by **Tools → BoneCracker Games → Realistic Car Controller → AI Controller**. (Detailed explanation in the documentation named "**Realistic Car Controller AI**")



RCC Settings

Main RCC Settings. It's shared by all vehicles powered by RCC. **Tools → BoneCracker Games → Realistic Car Controller → RCC Settings**. (Detailed explanation in the documentation named "**Realistic Car Controller RCC_Settings**")


Project | Inspector | Occlusion | Lighting | Project Settings | Services | Navigation

RCC_Settings

Open

RCC Asset Settings Editor Window

This editor will keep update necessary .asset files in your project for RCC. Don't change directory of the "Resources/RCC Assets".

General Settings

General Settings

Override FixedTimeStep	✓
Fixed Timestep	0.02
Maximum Angular Velocity	8
Override FPS	✓
Maximum FPS	60

You can find all references to any mode. Open up "RCC_Settings.cs" and right click to any mode. Hit "Find references" to find all modifications.

Use Fixed WheelColliders	✓
Locks Cursor	☐

Behavior Settings

Behavior Settings

Using behavior preset will override wheelcollider settings, chassis joint, antirolls, and other stuff. Using "Custom" mode will not override anything.

☐ **Override Behavior**

Behavior Types

Simulator | Racing | Drift | Semi Arcade | Fun

Controller Settings

Main Controller Type

[✓ Keyboard] | Mobile | XBox | Logitech Steering Wheel | Custom

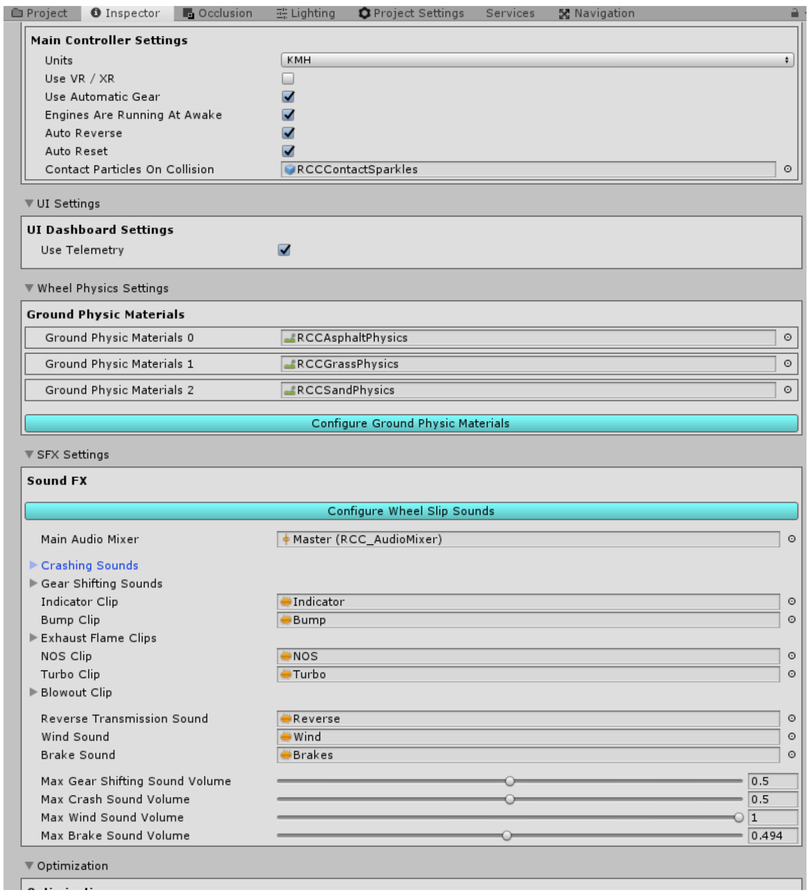
Keyboard Settings

In this mode, inputs will be received from Keyboard.

Gas/Reverse Input Axis	Vertical
Steering Input Axis	Horizontal
Mouse X Input Axis	Mouse X
Mouse Y Input Axis	Mouse Y

You can edit your vertical and horizontal input axis in Edit --> Project Settings --> Input.

Handbrake	Space
Start/Stop Engine Key	I
Low Beam Headlights	L
High Beam Headlights	K
Change Camera	C
Indicator Right	E



Main Controller Settings

Units	KMH
Use VR / XR	☐
Use Automatic Gear	✓
Engines Are Running At Awake	✓
Auto Reverse	✓
Auto Reset	✓
Contact Particles On Collision	RCCContactSparkles

▼ UI Settings

UI Dashboard Settings

Use Telemetry	✓

▼ Wheel Physics Settings

Ground Physic Materials

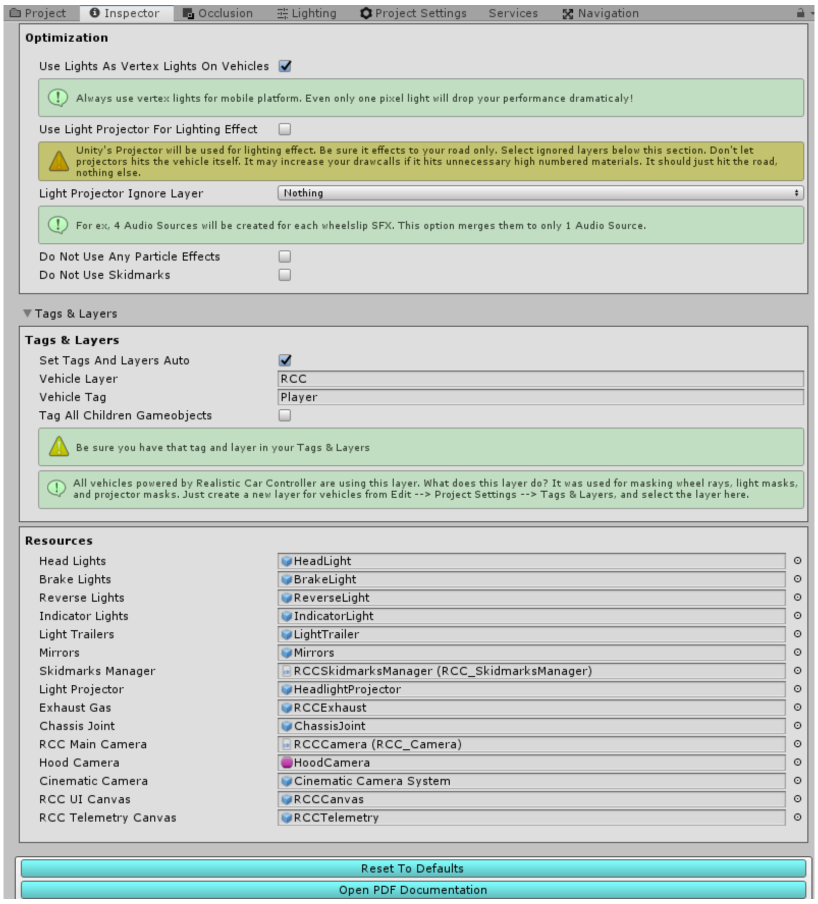
Ground Physic Materials 0	RCCAsphaltPhysics
Ground Physic Materials 1	RCCGrassPhysics
Ground Physic Materials 2	RCCSandPhysics

▼ SFX Settings

Sound FX

Main Audio Mixer	Master (RCC_AudioMixer)
▶ Crashing Sounds	
▶ Gear Shifting Sounds	
Indicator Clip	Indicator
Bump Clip	Bump
▶ Exhaust Flame Clips	
NOS Clip	NOS
Turbo Clip	Turbo
▶ Blowout Clip	
Reverse Transmission Sound	Reverse
Wind Sound	Wind
Brake Sound	Brakes
Max Gear Shifting Sound Volume	0.5
Max Crash Sound Volume	0.5
Max Wind Sound Volume	1
Max Brake Sound Volume	0.494

▼ Optimization



Optimization

Use Lights As Vertex Lights On Vehicles	✓

Always use vertex lights for mobile platform. Even only one pixel light will drop your performance dramaticaly!

Use Light Projector For Lighting Effect	☐

Unity's Projector will be used for lighting effect. Be sure it effects to your road only. Select ignored layers below this section. Don't let projectors hits the vehicle itself. It may increase your drawcalls if it hits unnecessary high numbered materials. It should just hit the road, nothing else.

Light Projector Ignore Layer	Nothing

For ex, 4 Audio Sources will be created for each wheelslip SFX. This option merges them to only 1 Audio Source.

Do Not Use Any Particle Effects	☐
Do Not Use Skidmarks	☐

▼ Tags & Layers

Tags & Layers

Set Tags And Layers Auto	✓
Vehicle Layer	RCC
Vehicle Tag	Player
Tag All Children Gameobjects	☐

Be sure you have that tag and layer in your Tags & Layers

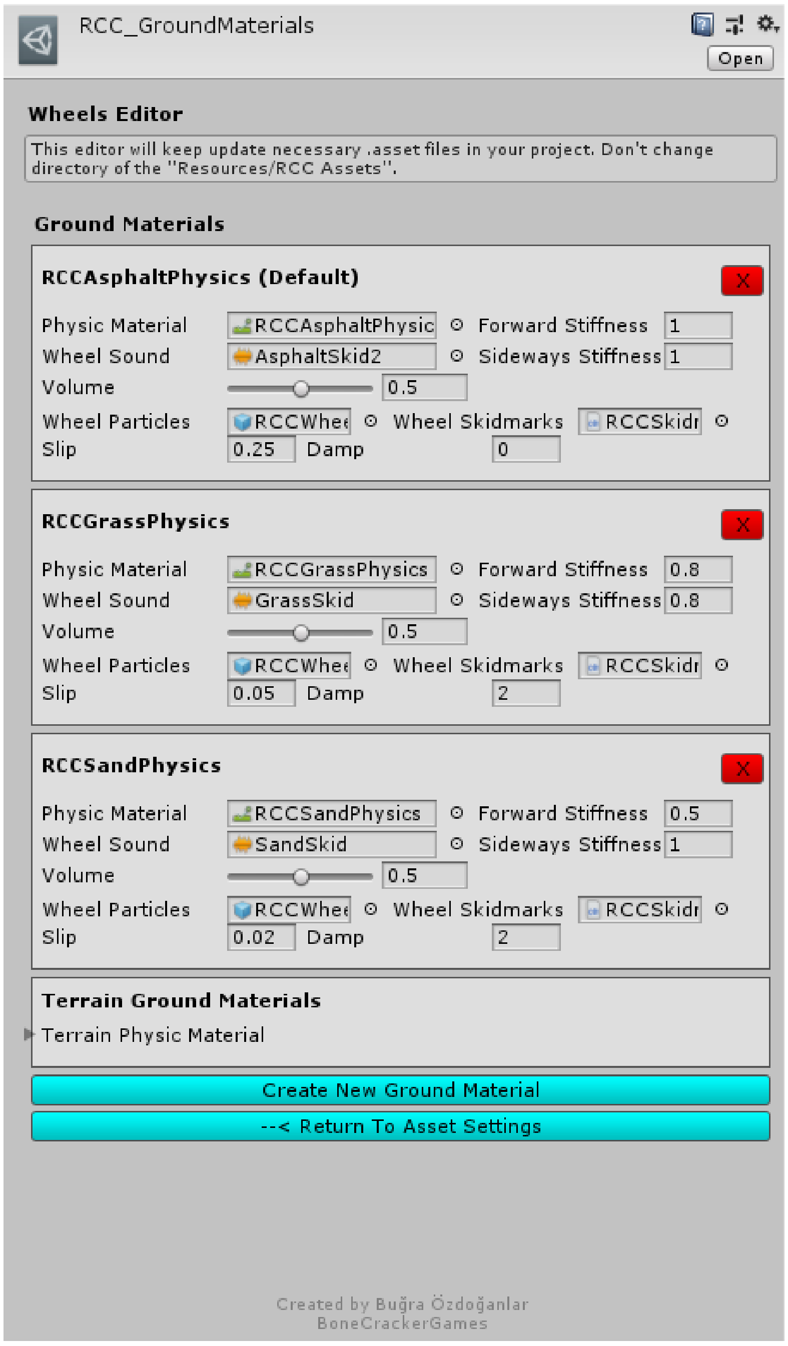
All vehicles powered by Realistic Car Controller are using this layer. What does this layer do? It was used for masking wheel rays, light masks, and projector masks. Just create a new layer for vehicles from Edit --> Project Settings --> Tags & Layers, and select the layer here.

Resources

Head Lights	HeadLight
Brake Lights	BrakeLight
Reverse Lights	ReverseLight
Indicator Lights	IndicatorLight
Light Trailers	LightTrailer
Mirrors	Mirrors
Skidmarks Manager	RCCSkidmarksManager (RCC_SkidmarksManager)
Light Projector	HeadlightProjector
Exhaust Gas	RCCExhaust
Chassis Joint	ChassisJoint
RCC Main Camera	RCCCamera (RCC_Camera)
Hood Camera	HoodCamera
Cinematic Camera	Cinematic Camera System
RCC UI Canvas	RCCCanvas
RCC Telemetry Canvas	RCCTelemetry

Configurable Ground Materials

Changing or adding new ground materials, physics, particles, damps, sounds, etc. in **Tools → BoneCracker Games → Realistic Car Controller → Configure Ground Materials**. (Detailed explanation in the documentation named "**Realistic Car Controller RCC_GroundMaterials**")



RCC_GroundMaterials

Wheels Editor

This editor will keep update necessary .asset files in your project. Don't change directory of the "Resources/RCC Assets".

Ground Materials

RCCAsphaltPhysics (Default)

Physic Material	RCCAsphaltPhysic	Forward Stiffness	1
Wheel Sound	AsphaltSkid2	Sideways Stiffness	1
Volume	0.5		
Wheel Particles	RCCWhee	Wheel Skidmarks	RCCSkidr
Slip	0.25	Damp	0

RCCGrassPhysics

Physic Material	RCCGrassPhysics	Forward Stiffness	0.8
Wheel Sound	GrassSkid	Sideways Stiffness	0.8
Volume	0.5		
Wheel Particles	RCCWhee	Wheel Skidmarks	RCCSkidr
Slip	0.05	Damp	2

RCCSandPhysics

Physic Material	RCCSandPhysics	Forward Stiffness	0.5
Wheel Sound	SandSkid	Sideways Stiffness	1
Volume	0.5		
Wheel Particles	RCCWhee	Wheel Skidmarks	RCCSkidr
Slip	0.02	Damp	2

Terrain Ground Materials

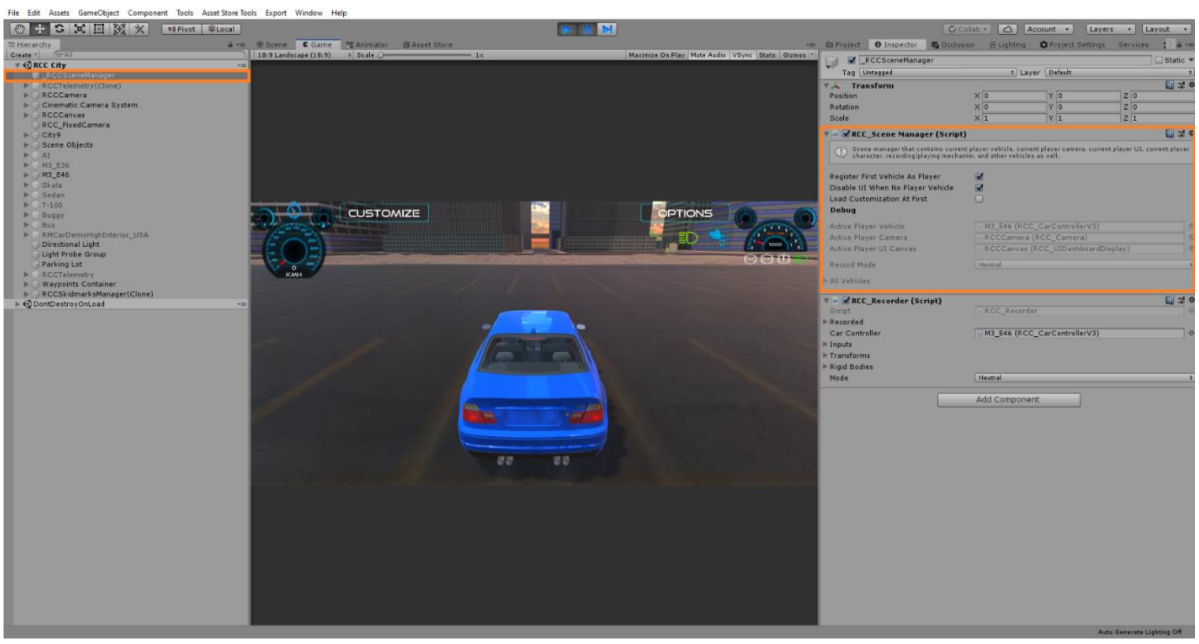
Terrain Physic Material

Open
X
Create New Ground Material
--< Return To Asset Settings
Created by Buğra Özdoğanlar
BoneCrackerGames

If wheelcollider hits a collider with one of the physic material in the list, changes will be applied to the wheelcollider. You can check out demo scenes.

RCC Scene Manager

Every scene will have this manager automatically. RCC Scene Manager contains current player vehicle, current player camera, current player UI, current player character, recording / replay mechanism, and other vehicles as well. Instead of finding current car controller, or camera on scene, RCC Scene Manager will find it and manage it only. All other scripts depending on the player vehicle will take reference of the RCC Scene Manager. For example, finding player vehicle on scene is **RCC_SceneManager.Instance.activePlayerVehicle**. All other codes can be found in scripts documentation.



_RCCSceneManager — Static
Tag: Untagged — Layer: Default

Transform
Position X 0 Y 0 Z 0
Rotation X 0 Y 0 Z 0
Scale X 1 Y 1 Z 1

RCC_Scene Manager (Script)
Scene manager that contains current player vehicle, current player camera, current player UI, current player character, recording/playing mechanim, and other vehicles as well.

Register First Vehicle As Player ✓
Disable UI When No Player Vehicle ✓
Load Customization At First ☐
Debug
Active Player Vehicle: M3_E46 (RCC_CarControllerV3)
Active Player Camera: RCCCamera (RCC_Camera)
Active Player UI Canvas: RCCCanvas (RCC_UIDashboardDisplay)
Record Mode: Neutral
All Vehicles

RCC_Recorder (Script)
Script: RCC_Recorder
Recorded
Car Controller: M3_E46 (RCC_CarControllerV3)
Inputs
Transforms
Rigid Bodies
Mode: Neutral

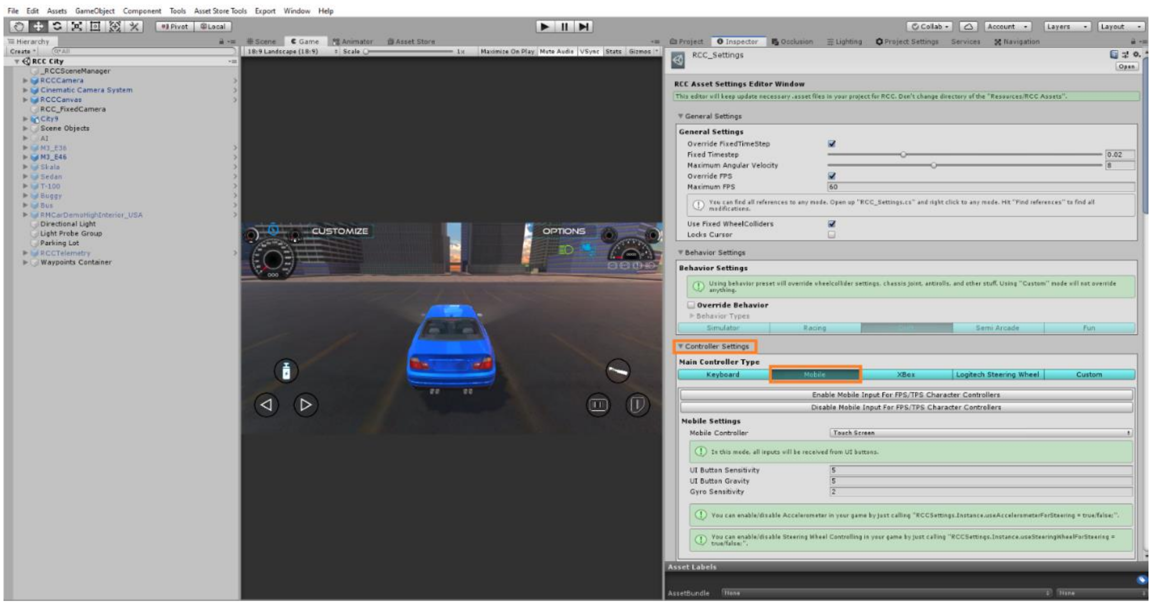
Controller Types

RCC supports all controller types with the new input system. Each controller can be changed directly from the RCC_InputActions (Detailed explanation in documentation named "**Realistic Car Controller New Input System**")

Logitech Steering requires **Logitech Gaming SDK** installed in your project.

Mobile Controller

Mobile controller is using my own input system instead of the new input manager. Each UI controller button has “RCC_UIController” script for inputs. These buttons feeds RCC_InputManager with normalized float values. You can adjust UI buttons sensitivity and gravity from the RCC Settings. Switching the mobile controller to the new input manager is easy, however I don’t recommend to do this. Because UI buttons will simulate gamepad buttons in this case.



RCC_Settings

RCC Asset Settings Editor Window

This editor will keep update necessary .asset files in your project for RCC. Don't change directory of the "Resources/RCC Assets".

General Settings

Override FixedTimeStep	✓
Fixed Timestep	0.02
Maximum Angular Velocity	8
Override FPS	✓
Maximum FPS	60

You can find all references to any mode. Open up "RCC_Settings.cs" and right click to any mode. Hit "Find references" to find all modifications.

Use Fixed WheelColliders	✓
Locks Cursor	☐

Behavior Settings

Using behavior preset will override wheelcollider settings, chassis joint, antirolls, and other stuff. Using "Custom" mode will not override anything.

☐ Override Behavior

Controller Settings

Main Controller Type: Keyboard [✓ Mobile] XBox Logitech Steering Wheel Custom

Mobile Settings

Mobile Controller	Touch Screen

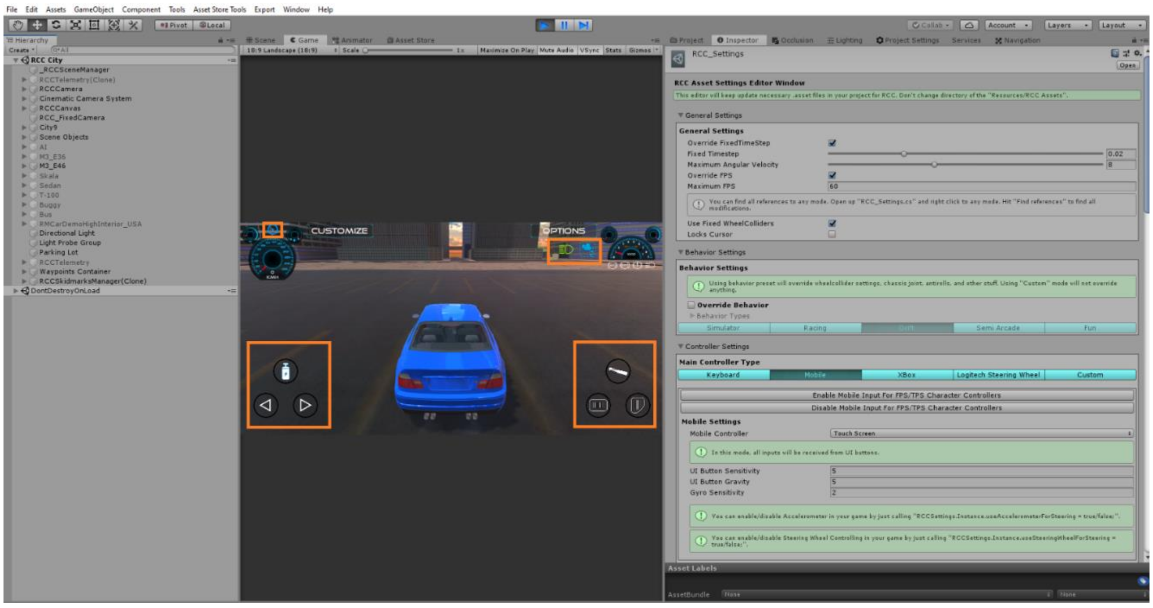
In this mode, all inputs will be received from UI buttons.

UI Button Sensitivity	5
UI Button Gravity	5
Gyro Sensitivity	2

You can enable/disable Accelerometer in your game by just calling "RCCSettings.Instance.useAccelerometerForSteering = true/false;".

You can enable/disable Steering Wheel Controlling in your game by just calling "RCCSettings.Instance.useSteeringWheelForSteering = true/false;".

CUSTOMIZE
OPTIONS
Enable Mobile Input For FPS/TPS Character Controllers
Disable Mobile Input For FPS/TPS Character Controllers
Simulator
Racing
Semi Arcade
Fun



RCC_Settings

RCC Asset Settings Editor Window

This editor will keep update necessary .asset files in your project for RCC. Don't change directory of the "Resources/RCC Assets".

General Settings

Override FixedTimeStep	✓
Fixed Timestep	0.02
Maximum Angular Velocity	8
Override FPS	✓
Maximum FPS	60

You can find all references to any mode. Open up "RCC_Settings.cs" and right click to any mode. Hit "Find references" to find all modifications.

Use Fixed WheelColliders	✓
Locks Cursor	☐

Behavior Settings

Using behavior preset will override wheelcollider settings, chassis joint, antirolls, and other stuff. Using "Custom" mode will not override anything.

☐ **Override Behavior**

Behavior Types

Simulator | Racing | Drift | Semi Arcade | Fun

Controller Settings

Main Controller Type

Keyboard | [✓ Mobile] | XBox | Logitech Steering Wheel | Custom

Enable Mobile Input For FPS/TPS Character Controllers

Disable Mobile Input For FPS/TPS Character Controllers

Mobile Settings

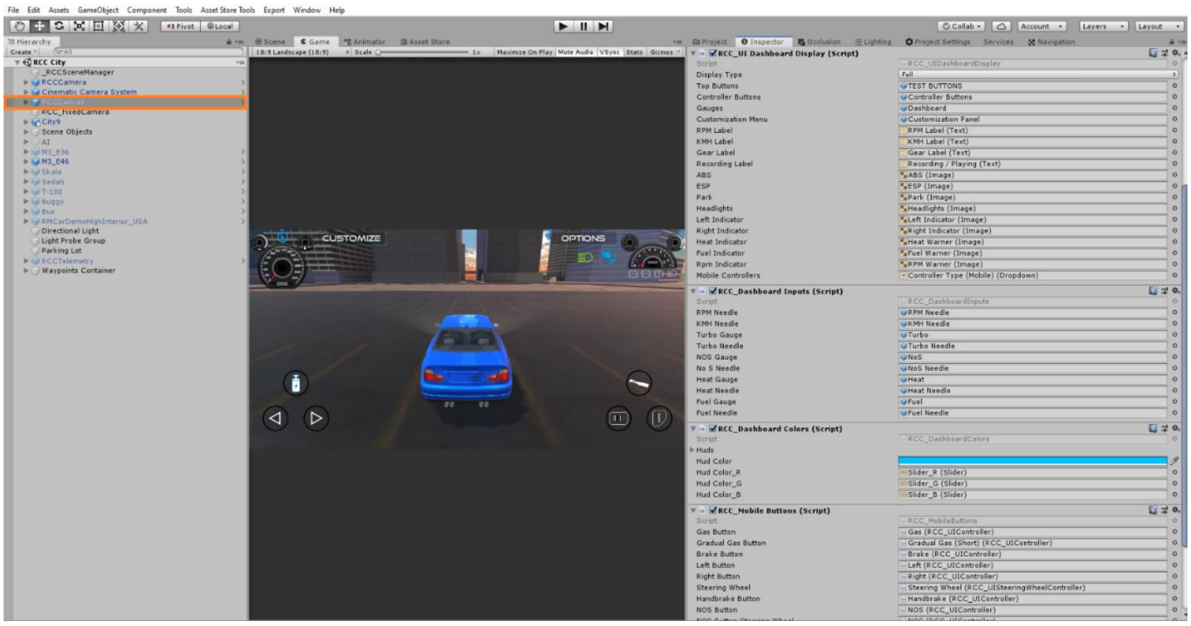
Mobile Controller	Touch Screen

In this mode, all inputs will be received from UI buttons.

UI Button Sensitivity	5
UI Button Gravity	5
Gyro Sensitivity	2

You can enable/disable Accelerometer in your game by just calling "RCCSettings.Instance.useAccelerometerForSteering = true/false;".

You can enable/disable Steering Wheel Controlling in your game by just calling "RCCSettings.Instance.useSteeringWheelForSteering = true/false;".



RCC_UI Dashboard Display (Script)

Script	RCC_UIDashboardDisplay
Display Type	Full
Top Buttons	TEST BUTTONS
Controller Buttons	Controller Buttons
Gauges	Dashboard
Customization Menu	Customization Panel
RPM Label	RPM Label (Text)
KMH Label	KMH Label (Text)
Gear Label	Gear Label (Text)
Recording Label	Recording / Playing (Text)
ABS	ABS (Image)
ESP	ESP (Image)
Park	Park (Image)
Headlights	Headlights (Image)
Left Indicator	Left Indicator (Image)
Right Indicator	Right Indicator (Image)
Heat Indicator	Heat Warner (Image)
Fuel Indicator	Fuel Warner (Image)
Rpm Indicator	RPM Warner (Image)
Mobile Controllers	Controller Type (Mobile) (Dropdown)

RCC_Dashboard Inputs (Script)

Script	RCC_DashboardInputs
RPM Needle	RPM Needle
KMH Needle	KMH Needle
Turbo Gauge	Turbo
Turbo Needle	Turbo Needle
NOS Gauge	NoS
No S Needle	NoS Needle
Heat Gauge	Heat
Heat Needle	Heat Needle
Fuel Gauge	Fuel
Fuel Needle	Fuel Needle

RCC_Dashboard Colors (Script)

Script	RCC_DashboardColors
Huds	
Hud Color	🟦
Hud Color_R	Slider_R (Slider)
Hud Color_G	Slider_G (Slider)
Hud Color_B	Slider_B (Slider)

RCC_Mobile Buttons (Script)

Script	RCC_MobileButtons
Gas Button	Gas (RCC_UIController)
Gradual Gas Button	Gradual Gas (Short) (RCC_UIController)
Brake Button	Brake (RCC_UIController)
Left Button	Left (RCC_UIController)
Right Button	Right (RCC_UIController)
Steering Wheel	Steering Wheel (RCC_UISteeringWheelController)
Handbrake Button	Handbrake (RCC_UIController)
NOS Button	NOS (RCC_UIController)

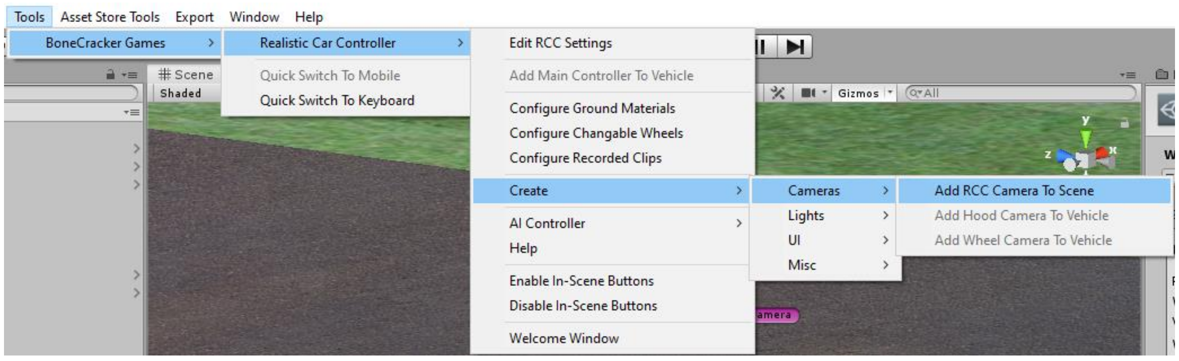
About Mobile Use on City Scene

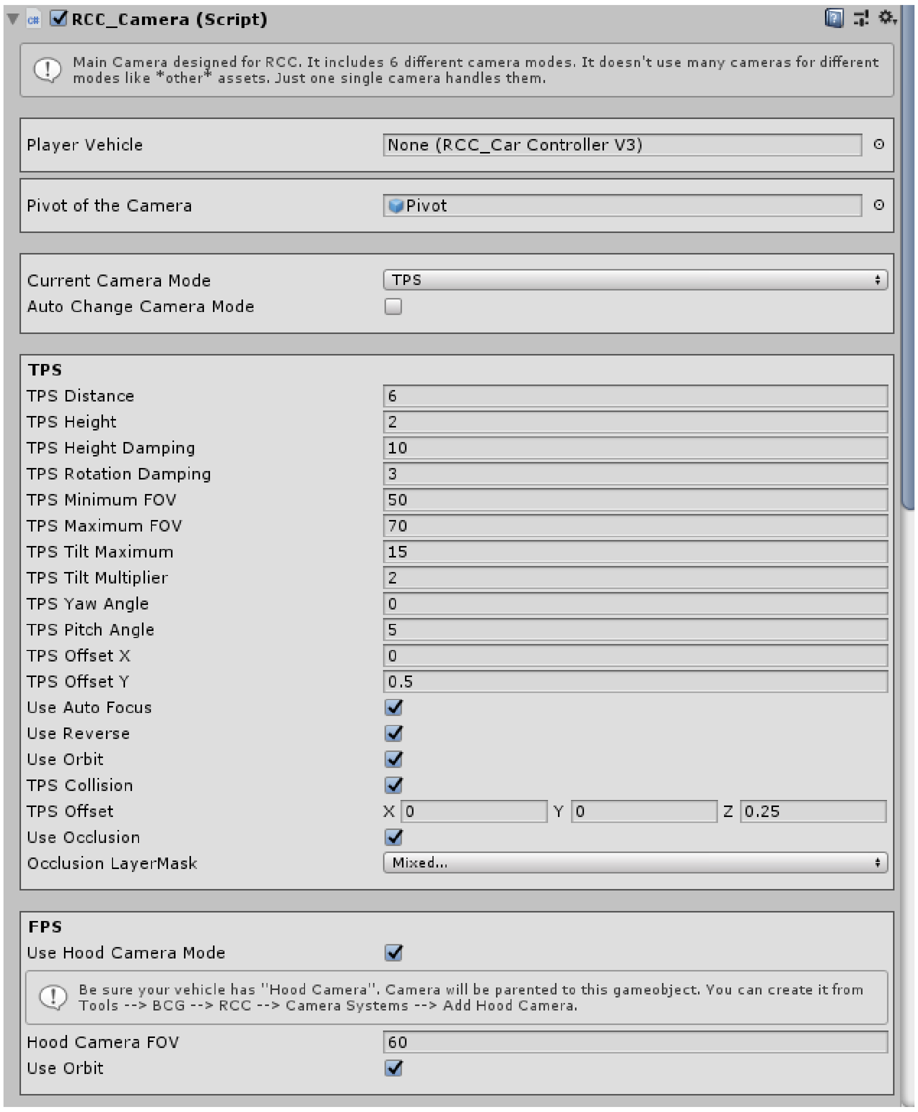
The city scene has a lot of specular maps with alpha channels. Textures with alpha channels and bump maps are heavy for mobile devices. In Demo APK in my website is not using any texture with alpha channels. Also, all standard shaders have been replaced with mobile shaders in the **RCC City Mobile** scene at the demo. If you build an APK without editing materials, you may get performance loss on low-end devices.

RCC Camera

Main camera system designed for using with RCC. Related with vehicle stats and includes six different camera modes with many customizable settings. It doesn't use different individual cameras on your scene. Simply it parents the camera to their positions, and that's all.

If your scene doesn’t include RCC Camera, you can create it from **Tools → BoneCracker Games → Realistic Car Controller → Create → Cameras → Add RCC Camera To Scene**.





RCC_Camera (Script)

Main Camera designed for RCC. It includes 6 different camera modes. It doesn't use many cameras for different modes like *other* assets. Just one single camera handles them.

Player Vehicle	None (RCC_Car Controller V3)
Pivot of the Camera	Pivot
Current Camera Mode	TPS
Auto Change Camera Mode	☐

TPS

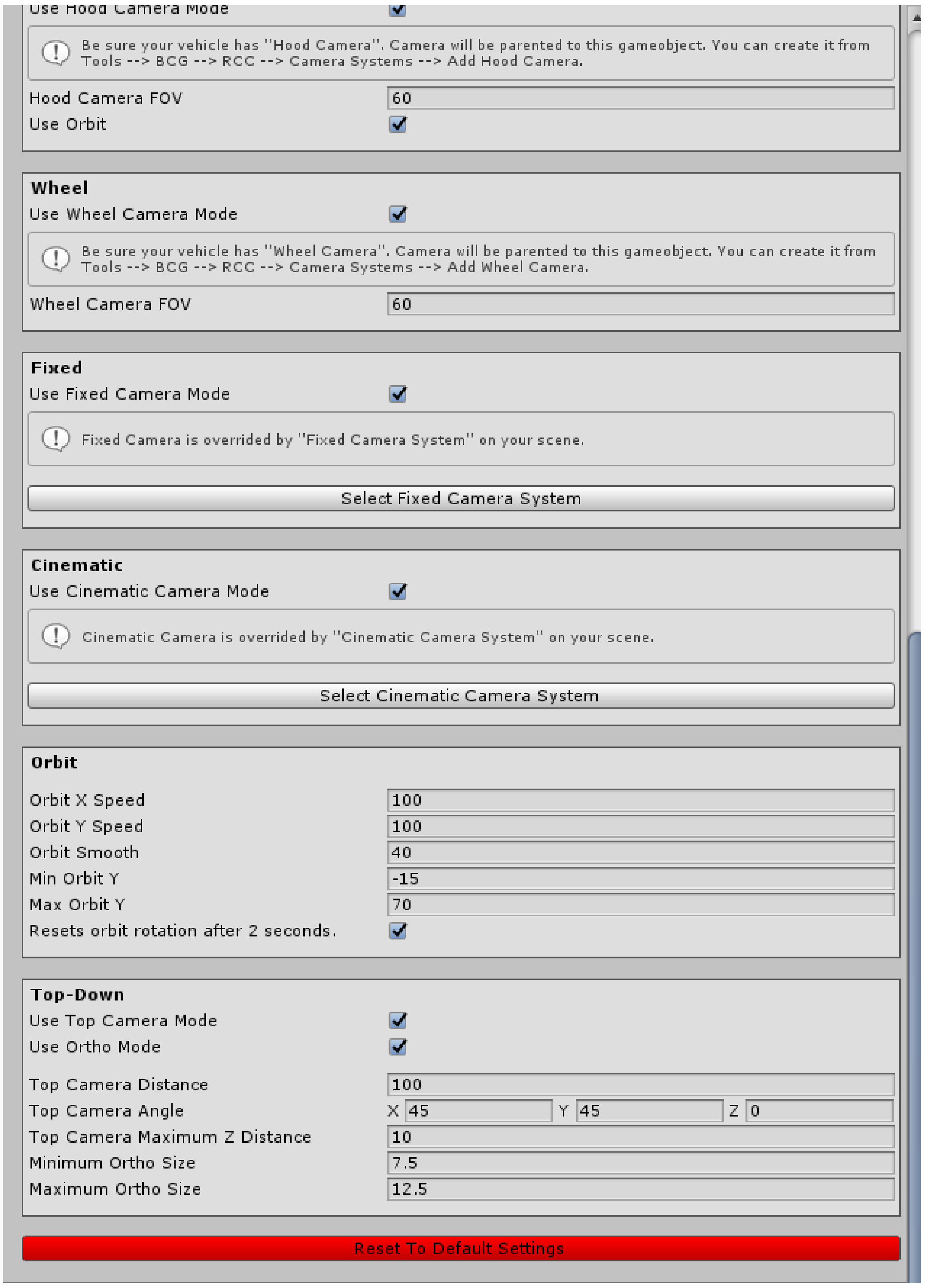
TPS Distance	6
TPS Height	2
TPS Height Damping	10
TPS Rotation Damping	3
TPS Minimum FOV	50
TPS Maximum FOV	70
TPS Tilt Maximum	15
TPS Tilt Multiplier	2
TPS Yaw Angle	0
TPS Pitch Angle	5
TPS Offset X	0
TPS Offset Y	0.5
Use Auto Focus	✓
Use Reverse	✓
Use Orbit	✓
TPS Collision	✓
TPS Offset	X 0 Y 0 Z 0.25
Use Occlusion	✓
Occlusion LayerMask	Mixed...

FPS

Use Hood Camera Mode	✓

Be sure your vehicle has ''Hood Camera''. Camera will be parented to this gameobject. You can create it from Tools --> BCG --> RCC --> Camera Systems --> Add Hood Camera.

Hood Camera FOV	60
Use Orbit	✓


Use Hood Camera Mode ✓

Be sure your vehicle has "Hood Camera". Camera will be parented to this gameobject. You can create it from Tools --> BCG --> RCC --> Camera Systems --> Add Hood Camera.

Hood Camera FOV	60
Use Orbit	✓

Wheel

Use Wheel Camera Mode ✓

Be sure your vehicle has "Wheel Camera". Camera will be parented to this gameobject. You can create it from Tools --> BCG --> RCC --> Camera Systems --> Add Wheel Camera.

Wheel Camera FOV 60

Fixed

Use Fixed Camera Mode ✓

Fixed Camera is overrided by "Fixed Camera System" on your scene.

Select Fixed Camera System

Cinematic

Use Cinematic Camera Mode ✓

Cinematic Camera is overrided by "Cinematic Camera System" on your scene.

Select Cinematic Camera System

Orbit

Orbit X Speed	100
Orbit Y Speed	100
Orbit Smooth	40
Min Orbit Y	-15
Max Orbit Y	70
Resets orbit rotation after 2 seconds.	✓

Top-Down

Use Top Camera Mode	✓
Use Ortho Mode	✓
Top Camera Distance	100
Top Camera Angle	X 45 Y 45 Z 0
Top Camera Maximum Z Distance	10
Minimum Ortho Size	7.5
Maximum Ortho Size	12.5

Reset To Default Settings

Each camera mode can be customized here. TPS mode is required, and all other modes are optional. If you don't want to use hood, wheel, fixed, cinematic camera, top-down modes, you can just disable them here.

Record / Replay

Complete physics and input based record / replay system. Player vehicle and all active AI vehicles can record / replay. All you must do is press "**R**" for start recording, and "**P**" for start replay. These buttons can be changed in the **RCC_InputActions**. And of course, there is a UI button for mobile.

RCC_Recorder can be found attached to **RCC_SceneManager** on your scene. You can enable or disable it. Script will be added at awake, or you can add it by manually if enabled. You can use RCC's API to start record / replay at runtime. For example;

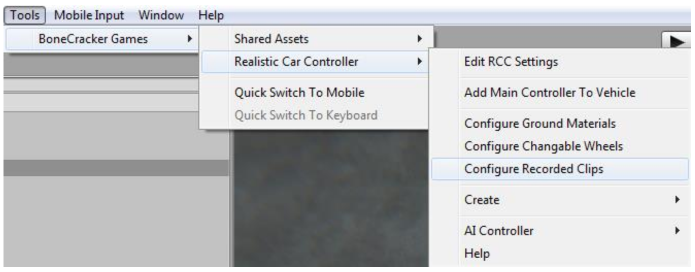
RCC. StartStopReplay ();

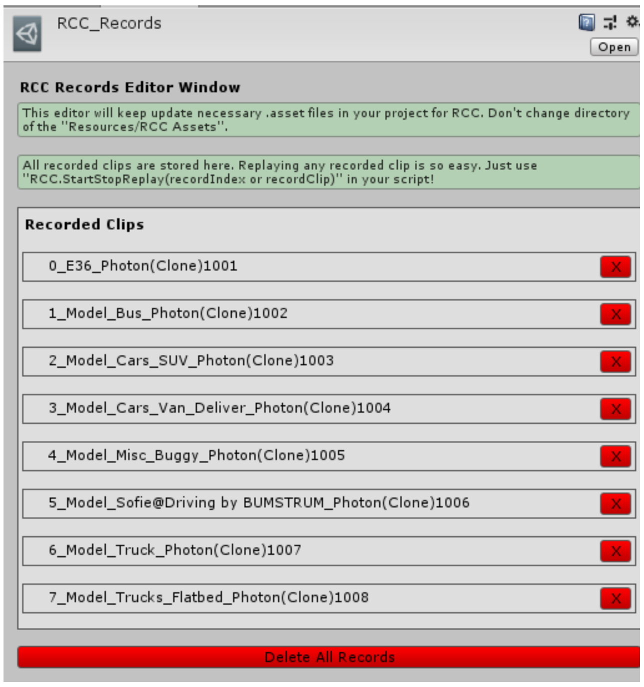
RCC. StartStopReplay (RCC_Recorder.Recorded recordedClip);

RCC. StartStopReplay (int index);

RCC. StartStopReplay (RCC_Recorder.Recorded recordedClip);

All records are stored in the **RCC_Records**. You can access it from the **Tools → BCG → RCC → Configure Recorded Clips**.





RCC Records Editor Window

This editor will keep update necessary .asset files in your project for RCC. Don't change directory of the "Resources/RCC Assets".

All recorded clips are stored here. Replaying any recorded clip is so easy. Just use "RCC.StartStopReplay(recordIndex or recordClip)" in your script!

Recorded Clips

- 0_E36_Photon(Clone)1001
- 1_Model_Bus_Photon(Clone)1002
- 2_Model_Cars_SUV_Photon(Clone)1003
- 3_Model_Cars_Van_Deliver_Photon(Clone)1004
- 4_Model_Misc_Buggy_Photon(Clone)1005
- 5_Model_Sofie@Driving by BUMSTRUM_Photon(Clone)1006
- 6_Model_Truck_Photon(Clone)1007
- 7_Model_Trucks_Flatbed_Photon(Clone)1008

Customization

You can customize your vehicles by just calling a single method. Please look at "**Realistic Car Controller Scripts**" documentation. All methods in the **RCC_Customization** have been explained there.

How The Customization Panel Works

I've written an example script called "**RCC_CustomizerExample.cs**" which uses static methods in the **RCC_Customization**. Script is attached to the **RCC_Canvas**. UI buttons in customization panel send methods to this example script. And this example script uses static methods in the **RCC_Customization** for making changes. Let me explain it with simple examples.

We want to change the front suspension distance of our vehicle. So, we have to use;

RCC_Customization.SetFrontSuspensionsDistances (targetRCC, targetValue);

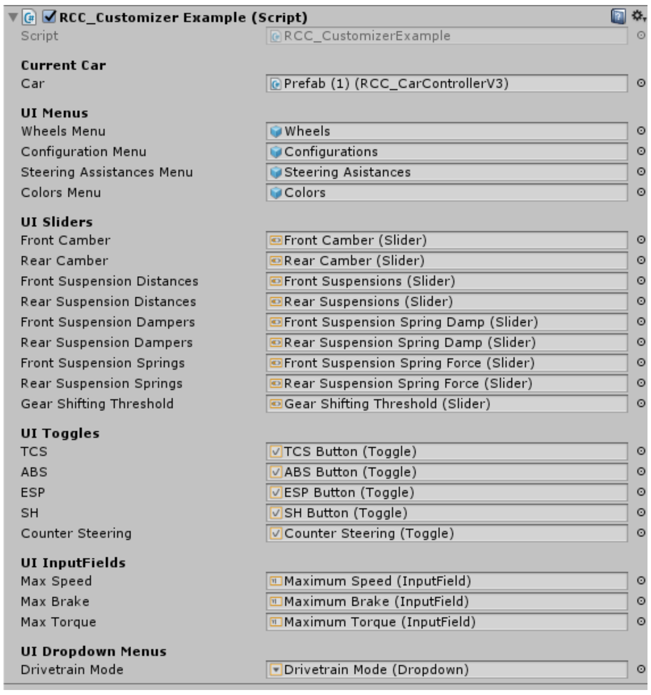
We want to repair our car. So, we have to use;

RCC_Customization. RepairCar (targetRCC);

We want to change the drivetrain of our car to AWD. So, we have to use;

```
RCC_Customization. SetDrivetrainMode (targetRCC,
RCC_CarControllerV3 WheelType.AWD);
```

And goes on... Simply look at all methods in **RCC_CustomizerExample** script, you will see how I customized the player vehicle by using **RCC_Customization** script.



RCC_Customizer Example (Script)

Script	RCC_CustomizerExample
Current Car	
Car	Prefab (1) (RCC_CarControllerV3)
UI Menus	
Wheels Menu	Wheels
Configuration Menu	Configurations
Steering Assistances Menu	Steering Asistances
Colors Menu	Colors
UI Sliders	
Front Camber	Front Camber (Slider)
Rear Camber	Rear Camber (Slider)
Front Suspension Distances	Front Suspensions (Slider)
Rear Suspension Distances	Rear Suspensions (Slider)
Front Suspension Dampers	Front Suspension Spring Damp (Slider)
Rear Suspension Dampers	Rear Suspension Spring Damp (Slider)
Front Suspension Springs	Front Suspension Spring Force (Slider)
Rear Suspension Springs	Rear Suspension Spring Force (Slider)
Gear Shifting Threshold	Gear Shifting Threshold (Slider)
UI Toggles	
TCS	TCS Button (Toggle)
ABS	ABS Button (Toggle)
ESP	ESP Button (Toggle)
SH	SH Button (Toggle)
Counter Steering	Counter Steering (Toggle)
UI InputFields	
Max Speed	Maximum Speed (InputField)
Max Brake	Maximum Brake (InputField)
Max Torque	Maximum Torque (InputField)
UI Dropdown Menus	
Drivetrain Mode	Drivetrain Mode (Dropdown)

This example script handles all UI menus, buttons, sliders, toggles, input fields, and dropdown menus of the customization panel. It just receives inputs from the UI, and fires necessary actions.

Credits

Driver Sofie, her animations, and her car model made by 3DMaesen. You can access 3DMaesen asset store from this link.

http://u3d.as/2vg

All sounds in the package are completely royalty free. You can use them on any personal or commercial projects. You can't redistribute / resell them.

License

You can ask me anything about my assets! If you want to change **minor things** in the package, don't waste your time by editing scripts. Just tell me, I'll do my best with no cost. I don't take any projects right now, and I'm not available for hire. Please email me if you used any of my assets in your game, I'd like to see it in action!

Ekrem Bugra Ozdoganlar

Bonecrackergames@gmail.com