

Arcade Car Physics (ACP)

[asset store page](#) [extented version asset store page](#)

This unity package provides scripts and examples to build arcade cars in Unity3D using the built-in wheel colliders. While the wheel colliders have a reputation to be buggy and unstable I found that it is not true but they are a bit tricky to get right.

In this package you'll find one demo scene with 4 example prefabs with low poly models made using [kenney's asset forge](#).

Getting Started

First start by changing the inputs in the DemoInputs asset to use inputs in your project settings (Edit > Project Settings > Inputs)

- I have errors about missing inputs! Go to your project settings and add the missing inputs
- Materials are pink/missing! Unity's rendering pipelines have changed since this plugin started, feel free to change the meshes for your own
- How do I change the meshes? Look in the car prefabs, you should find the Body object and replace its children with your own mesh and move the wheels

[More documentation can be found here](#) [How to create a car, video tutorial](#), I'm not very good at tutorials so this is just a video of me building one of the cars from the demo.

Vehicle

The typical vehicle's composition is as follows:

```
├─ CarPrefab
│   ├─ Body
│   │   └─ 3D mesh REPLACE THIS WITH YOUR OWN MESH
│   ├─ Front Left Wheel
│   │   ├─ Wheelcollider
│   │   └─ Suspension (Script)
│   │   └─ Trail Emitter (Script)
│   ├─ Front Right Wheel
│   │   ├─ Wheelcollider
│   │   └─ Suspension (Script)
│   │   └─ Trail Emitter (Script)
│   ├─ Back Left Wheel
│   │   ├─ Wheelcollider
│   │   └─ Suspension (Script)
│   │   └─ Trail Emitter (Script)
│   ├─ Back Right Wheel
│   │   ├─ Wheelcollider
│   │   └─ Suspension (Script)
│   │   └─ Trail Emitter (Script)
│   └─ CenterOfMass (Transform)
└─ Boost and smokes particles...
```

WheelVehicle script

This is the main script of the package controlling the behaviour of the cars, inputs and effects. The inspector is divided into the next 5 parts

Inputs

This is the list of input names used by the car. You can change them as you want and you can also use the same input for drift and boost to add some dramatic drifts to the cars. If you do not want the car to be controlled with Inputs just set `IsPlayer` to false. You can then use the variables Speed, Brake, Jump, Boost and Drift from scripts to control the cars. This is useful if you're making an IA or controlling the cars in other ways. For example this was used with a hinge joint to manually steer a vehicle in VR.

Finally an AnimationCurve name Turn Input Curve is used to have a non-linear steering. The value used in the prefabs is a good example giving more precision to the player

Wheels

You have two lists of wheels: the drive wheels and the turn wheels. 4-wheel steering isn't supported in the current WheelVehicle script as they all steer in the same direction but it can be added if needed, just let me know.

Behaviour

- Motor Torque: Animation curve representing the motor torque in Newton depending on the speed of the car. Having a negative torque after the top speed prevents going faster in some conditions

- Diff Gearing: The differential gearing applied between the motor and the wheels.
- Brake Force: The power of your brakes, mostly determines if we can lock the wheels.
- Steer Angle: Maximum angle of the turn wheels
- Steer Speed: Reaction speed to the inputs, basically go low for old cars and high for cars with power steering
- Jump Vel: How high do you want to jump?
- Drift Intensity
- Center Of Mass: I'd recommend using a center of mass a bit lower than the middle of the wheels to stay upright but don't go under the ground.
- Downforce: How much downforce the car generates when rolling, should be high for sports car
- Handbreak: bool
- Speed: read only indication of speed

Particles

List of gas particles, the emission rate is then indexed on the throttle value.

Boost

- Boost: current boost available
- MaxBoost: Max value allowed for Boost
- Boost Regen: per second
- Boost force: the boost is a force applied to the body, you should adjust this depending on the mass of the vehicle.
- Boost particles and sounds

Other Scripts

Engine sound manager

Goes next to the wheel vehicle and emits sound depending on the state of the vehicle. This is a very basic implementation and I would recommend looking at some FMod tutorials to make your own improved car engine sounds ;)

Suspension

This script goes next to WheelColliders and offers a way to keep the wheel model where the WheelColliders is on realtime depending on the terrain. You might need to adjust the local rotation offset to get the wheel's rotation right

Trail emitter

This script emits trails when the wheels start sliding around and drifting. The generated mesh as it's defaults and will be updated in the future

Camera Follow

Basic script used to have a camera following the cars in the demo scene, feel free to reuse it.

Ghost

A Ghost script and Ghost recorder are provided to save your cars time and movement. Basically use the GhostRecorder to record and put a Ghost script on the ghost you want to replay. Example will come but don't hesitate to ask me if you have any questions about this.

Extended version

The [extended version](#) adds some prefabs to show a bit more of what you can do with ACP. The most interesting examples are the drift car, the motorbike and the tank.

Example

You can find a project I made using this asset and all the features on my [GitHub under the name Autober](#). If you just want to test it you can find a compiled version of [autober on itch.io](#)