Крошечный автомобильный контроллер Advance

Руководство пользователя v2.0

Оглавление

Описание

Этот контроллер позволяет вам создавать и управлять базовым транспортным средством с полностью настраиваемой аркадной физикой.

Вместо использования стандартного колесного коллайдера Unity, который часто излишне сложен и подвержен сбоям, колеса этого контроллера состоят из сферических коллайдеров с несколькими специально настроенными соединениями, имитирующими двигатель, рулевое управление и подвеску автомобиля.

Функции

- Простая и беспроблемная настройка
- Управляйте ускорением, скоростью, трением, трансмиссией, столкновениями и многим другим.
- Использует физический движок Unity для идеальной совместимости с другими ресурсами.
- Легкий, идеально подходит для мобильных игр.
- Включает примеры сценариев для управления вводом и камерой.
- Совместимость с любой ОС, конвейером рендеринга или версией Unity.

Требования

Рекомендуется Unity версии 2019.4 или новее, но она также должна работать с любыми более новыми версиями.

Настоятельно рекомендуется иметь средние знания программирования на C#.

Контакт

Дэвид Джалберт (программист)

Электронная почта:jalbert.d@hotmail.com

Мастодонт:https://mastodon.gamedev.place/@davidjayindie

Твиттер:https://twitter.com/DavidJayIndie

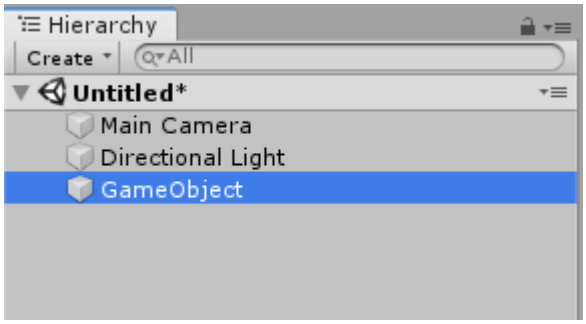
Пакет хранилища ресурсов Unity https://assetstore.unity.com/packages/slug/198873

Демо-версия WebGL

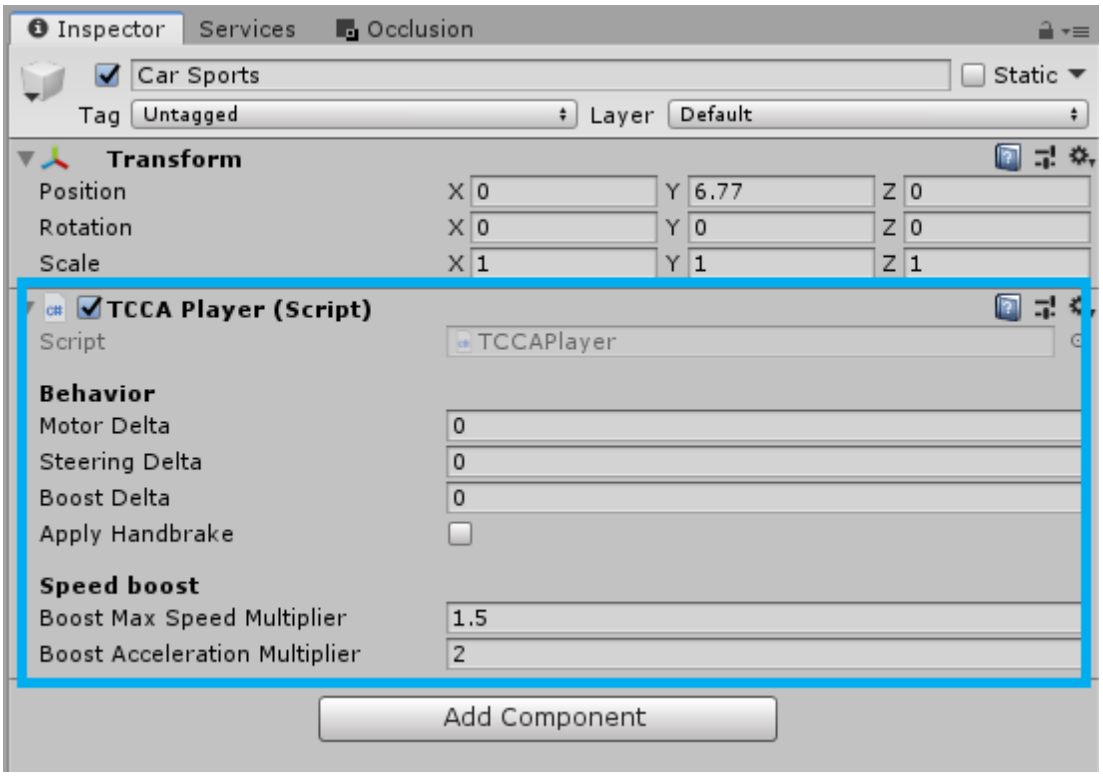
https://davidjalbert.itch.io/tiny-car-controller-advance-webgl-demo

Минимальная настройка

1) Создайте пустой GameObject в своей сцене.

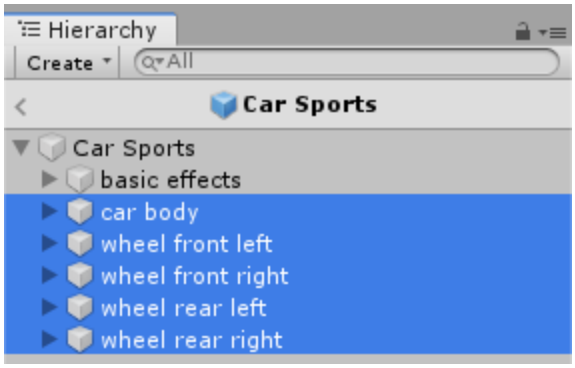


2) Добавьте скрипт «Assets/DavidJalbert/TinyCarControllerAdvance/Scripts/Core/TCCAPlayer.cs» в пустой GameObject.

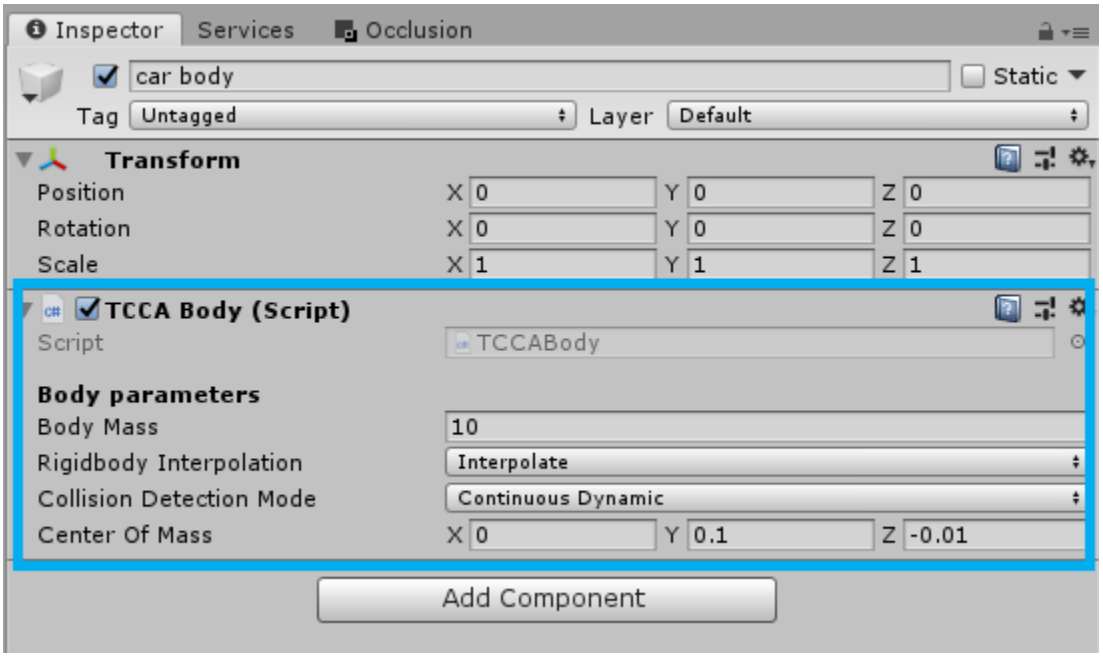


3) Создайте детей для кузова автомобиля и его колес. Поместите свои 3D-модели внутрь этих объектов.

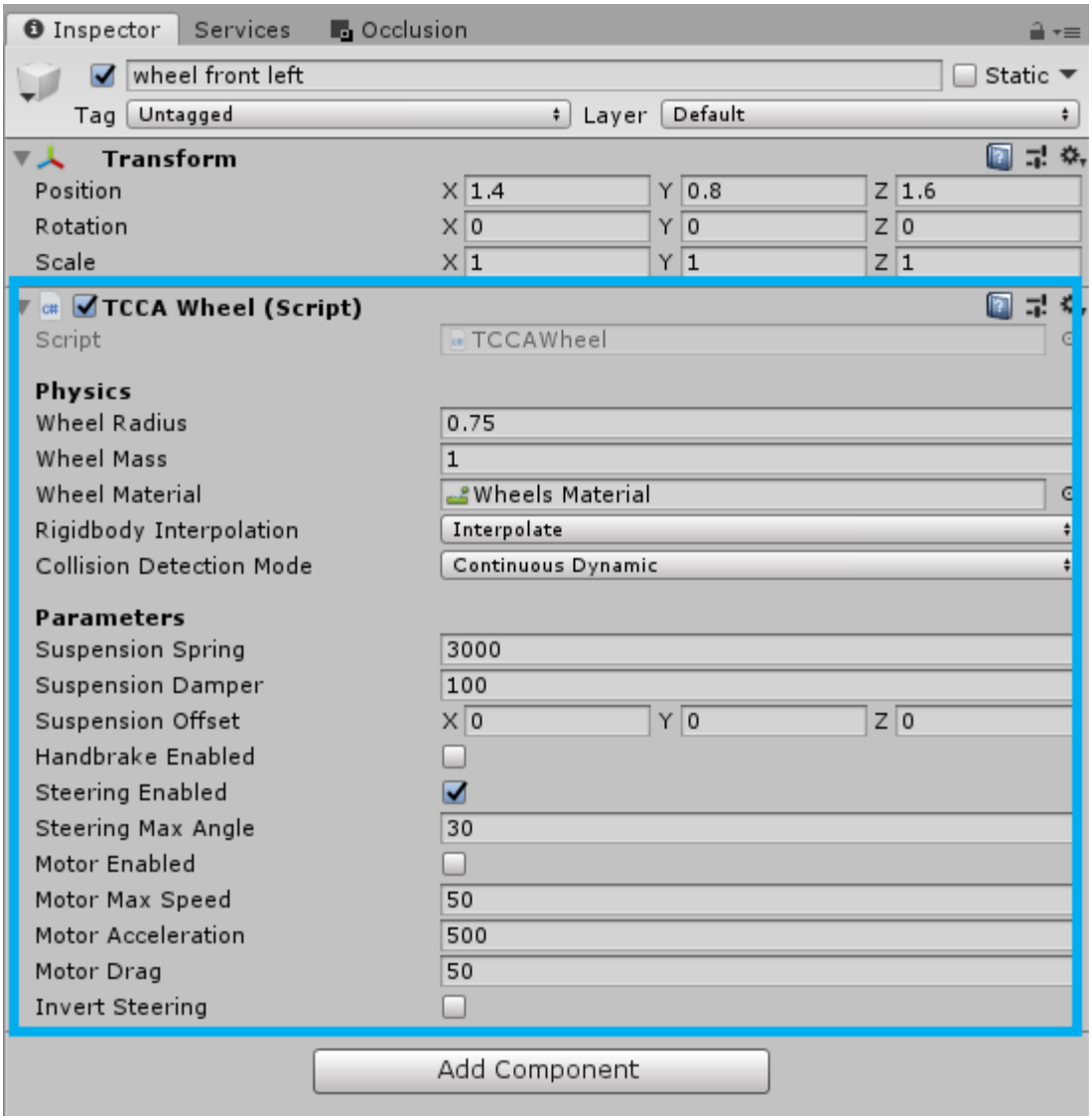
20 апреля



4) Добавьте сценарий «Assets/DavidJalbert/TinyCarControllerAdvance/Scripts/Core/TCCABody.cs» к объекту кузова автомобиля.



5) Добавьте сценарий «Assets/DavidJalbert/TinyCarControllerAdvance/Scripts/Core/TCCAWheel.cs» к каждому объекту колеса автомобиля. Обратите внимание: если в моделях колес есть коллайдеры, они будут отключены при запуске игры.



6) Установите параметры трех скриптов по своему вкусу (пояснения см. ниже).

На этом этапе ваш контроллер готов к использованию, но у него не будет никаких входов или управления камерой. Для этого вам потребуются дополнительные скрипты, которые используются в примере сцены. Посмотрите сцену «DavidJalbert\TinyCarControllerAdvance», чтобы узнать, как их использовать.

20.06.20

Параметры

TCCA-плеер

Мотор Дельта

Какой крутящий момент прикладывать к колесам. 1 — полная скорость вперед, -1 — полная скорость назад, 0 — отдых.

Рулевая дельта

Насколько сильно поворачивать колеса. 1 — направо, -1 — налево, 0 — прямо.

Повышение Дельты

Какое усиление применить к колесам. 1 — полное усиление, 0 — отсутствие усиления.

Применить ручной тормоз

Стоит ли задействовать ручник на колесах.

Увеличение множителя максимальной скорости

Множитель скорости, применяемый при использовании ускорения.

Увеличение множителя ускорения

Множитель ускорения, применяемый при использовании ускорения.

Зафиксировать положение X/Y/Z

Предотвращает движение автомобиля вдоль оси.

Тело TCCA

Масса тела

Масса, которая будет нанесена на тело.

Интерполяция твердого тела

Применять ли интерполяцию к телу.

Режим обнаружения столкновений

Какой режим обнаружения столкновений использовать на кузове.

Центр массы

Центр масс тела в локальном пространстве. В идеале это должен быть центр автомобиля на уровне земли. Измените значение Z, чтобы машина наклонялась назад или вперед в воздухе.

Режим счетчика рулонов

Когда применять противодействующую силу крену.

Целевой угол счетчика крена

Угол в градусах, на который следует повернуть автомобиль. Установите значение 0, чтобы катиться идеально вертикально.

Противодействующая сила крена

Какую силу необходимо приложить, чтобы повернуть автомобиль в вертикальное положение, если он перевернется?

Вращающееся встречное сглаживание

Как быстро повернуть автомобиль в вертикальное положение, если он перевернется. Установите на ноль, чтобы сделать это мгновенно.

Счетчик крена с превышением скорости

Какую силу (от 0 (нет) до 1 (максимальная) следует применять в зависимости от скорости автомобиля, от 0 (стационар) до 1 (максимальная скорость).

Режим счетчика шага

Когда применять силу противодействия тангажу.

Целевой угол счетчика тангажа

Угол в градусах, на который следует повернуть автомобиль. Установите значение 0, чтобы выровнять идеально прямо.

Сила противодействия шагу

Какую силу приложить, чтобы выровнять автомобиль.

Сглаживание счетчика высоты тона

Как быстро выровнять машину. Установите на ноль, чтобы сделать это мгновенно.

Счетчик шага над скоростью

Какую силу (от 0 (нет) до 1 (максимальная) следует применять в зависимости от скорости автомобиля, от 0 (стационар) до 1 (максимальная скорость).

Режим рулевого стабилизатора

Когда применять силу для стабилизации направления рулевого управления. Это добавляет к кузову дополнительную угловую силу, соответствующую текущему значению рулевого управления.

Усилие рулевого стабилизатора

Какое усилие приложить для стабилизации рулевого управления.

Сглаживание рулевого стабилизатора

Как быстро стабилизировать рулевое управление. Установите на ноль, чтобы сделать это мгновенно.

Рулевой стабилизатор превышает скорость

Насколько стабилизируется автомобиль в зависимости от его скорости. Обычно вы хотите, чтобы он был равен нулю, когда он не движется (при нулевой скорости), и увеличивайте его до единицы (полная стабилизация), когда скорость начинает расти.

Колесо TCCA

Радиус колеса

Радиус колеса коллайдера. Он должен быть равен размеру модели колеса.

Масса колеса

Масса жесткого корпуса колеса.

Материал колеса

Материал жесткого корпуса колеса.

Интерполяция твердого тела

Использовать ли интерполяцию для твердого тела колеса.

Режим обнаружения столкновений

Какой режим обнаружения столкновений использовать для коллайдера колес.

Подвеска Пружина

Сила, приложенная к подвеске. Более высокие значения делают подвеску более жесткой.

Подвеска Демпфер

В подвеске установлен демпфер. Более высокие значения ускоряют стабилизацию подвески.

Смещение подвески

Смещает положение колеса относительно его исходного положения. Полезно для имитации гидравлики.

Рулевое управление включено

Разрешить ли колесу поворачиваться влево или вправо.

Максимальный угол поворота рулевого управления

Максимальный угол, на который может повернуться колесо.

Превышение скорости рулевого управления

Насколько сильное рулевое управление должно применяться в диапазоне от 0 (нет) до 1 (макс.) в зависимости от скорости автомобиля, от 0 (неподвижно) до 1 (максимальная скорость).

Рулевая пружина

Величина силы, прикладываемой к рулевой оси.

Рулевой демпфер

Величина трения, применяемая к рулевой оси.

Инвертировать рулевое управление

Следует ли инвертировать рулевое управление. Полезно для задних колес, если вы хотите иметь четырехколесное рулевое управление.

Двигатель включен

Разрешить ли колесу ускоряться.

Максимальная скорость двигателя

Максимальная скорость, до которой колесо может продолжать ускоряться при использовании двигателя.

Ускорение двигателя

Максимальное ускорение, применимое к колесу при использовании двигателя.

Ускорение двигателя выше скорости

Какое ускорение (от 0 (нет) до 1 (максимальное) следует применять относительно скорости автомобиля, от 0 (стационар) до 1 (максимальная скорость).

Мотор Дрэг

Скорость, с которой колесо будет замедляться, если не ускоряться.

Ручной тормоз включен

Разрешить ли колесу использовать ручник.

Показать коллайдеры

По умолчанию объекты и соединения коллайдера скрыты в инспекторе. С помощью этой опции вы можете сделать их видимыми.

Справочник по сценариям

TCCAPlayer

Инициализирует и контролирует поведение колес и кузова, а также включает функции для установки и возврата положения и вращения транспортного средства, проверки состояния заземления, скорости и т. д.

общедоступный TCCAbody getCarBody()

Возвращает связанный компонент TCCAbody.

общедоступное жесткое тело getRigidbody()

Возвращает компонент Rigidbody объекта TCCAbody.

public void setMotor(float d)

Устанавливает крутящий момент, приложенный к каждому дочернему колесу, у которого включен двигатель. Принимает любое значение от -1 до 1.

общедоступный поплавок getMotor()

Возвращает крутящий момент, приложенный к колесам в данный момент.

public void setSteering (float d)

Устанавливает направление рулевого управления для каждого дочернего колеса, для которого включено рулевое управление. -1, чтобы повернуть налево, 1, чтобы повернуть направо.

общедоступное число с плавающей запятой getSteering()

Возвращает направление поворота, примененное в данный момент к колесам.

public void setHandbrake(bool e)

Устанавливает, следует ли блокировать крутящий момент колес.

public bool getHandbrake()

Возвращает, заблокированы ли колеса в данный момент.

public void setBoost (float d)

Какое усиление применить к колесам. 1 — полное усиление, 0 — отсутствие усиления.

общедоступное число с плавающей запятой getBoost()

Возвращает текущий объем усиления, примененного к транспортному средству.

общедоступное число с плавающей запятой getWheelsMaxSpin (int Direction = 0)

Возвращает максимальный крутящий момент колес данного автомобиля. «направление» определяет, следует ли учитывать крутящий момент колеса, только если оно движется в определенном направлении. 1 — вперед, -1 — назад, 0 — для подсчета вращений вперед и назад.

общедоступное плавающее значение getWheelsMaxSpeed()

Возвращает максимальную максимальную скорость колес этого автомобиля.

общедоступное число с плавающей запятой getPitchAngle()

Возвращает текущий наклон (ось X в градусах) кузова транспортного средства.

общедоступное число с плавающей запятой getRollAngle()

Возвращает текущий крен (ось Z в градусах) кузова транспортного средства.

общедоступное число с плавающей запятой getForwardVelocity()

Возвращает текущую скорость транспортного средства относительно его направления вперед.

общедоступное число с плавающей запятой getForwardVelocityDelta()

Возвращает текущую скорость транспортного средства относительно его направления вперед, деленную на максимальную скорость колес.

общедоступное число с плавающей запятой getLateralVelocity()

Возвращает текущую скорость транспортного средства относительно его правильного направления.

общедоступное число с плавающей запятой getVerticalVelocity()

Возвращает текущую скорость транспортного средства относительно его направления вверх.

public bool isPartiallyGrounded()

Возвращает информацию о том, заземлено ли хотя бы одно колесо, но не все.

общественный логический isGrounded()

Возвращает, заземлено ли хотя бы одно колесо.

public bool isFullyGrounded()

Возвращает, заземлены ли все колеса.

публичная пустота, иммобилизация()

Устанавливает все скорости и крутящие моменты на ноль.

public void Translate (позиция Vector3)

Добавляет значение «position» к текущему положению автомобиля.

public void Rotate (вращение кватерниона)

Добавляет значение «ротации» к текущему повороту автомобиля.

публичный недействительный рецентратор()

Если корневой GameObject не находится в нулевой позиции, он

сбрасывается. **public void setPosition (позиция Vector3)**

Устанавливает положение транспортного средства в «position».

public void setRotation (вращение кватерниона)

Устанавливает вращение транспортного средства на «ротацию».

общедоступный Vector3 getPosition()

Возвращает текущую позицию автомобиля в мировом пространстве.

общедоступный кватернион getRotation()

Возвращает текущее вращение автомобиля в мировом

пространстве. **общедоступный Vector3 getInitialPosition()**

Возвращает положение транспортного средства, в котором он был впервые создан.

общедоступный кватернион getInitialRotation()

Возвращает поворот транспортного средства при первом создании экземпляра.

public void setParent (родительский элемент преобразования)

Устанавливает преобразование, в котором будет содержаться транспортное средство.

TCCAbody

Управляет основным корпусом и реализует силы противодействия крену. Добавляет компонент «TCCABodyCollider» к этому GameObject для управления обнаружением столкновений.

публичная виртуальная пустота onCollisionStay (столкновение)

публичная виртуальная пустота onCollisionEnter (столкновение)

публичная виртуальная пустота onCollisionExit (столкновение)

публичная виртуальная пустота onTriggerStay (другое коллайдер)

публичная виртуальная пустота onTriggerEnter (другой коллайдер)

публичная виртуальная пустота onTriggerExit (другой коллайдер)

Вызывается из одноименных функций в TCCABodyCollider. Вы можете создать собственный класс, расширяющий TCCAbody, и переопределить вышеуказанные функции для обнаружения коллизий и триггеров.

public void инициализировать (родительский элемент TCCAPlayer)

Инициализирует кузов транспортного средства с указанным компонентом TCCAPlayer.

публичное обновление void (float deltaTime)

Обновляет параметры и физику кузова автомобиля. Вызывается один раз в каждом кадре в Update(). **общедоступное**

число с плавающей запятой getPitchAngle()

Возвращает текущий наклон (ось X в градусах) кузова транспортного средства.

общедоступное число с плавающей запятой getRollAngle()

Возвращает текущий крен (ось Z в градусах) кузова транспортного

средства. **public bool canCounterRotation (CounterMode m)**

Возвращает, будет ли транспортное средство в данный момент пытаться сбалансироваться с указанным режимом противодействия вращению.

общедоступный Vector3 getForwardAngularVelocity()

Возвращает скорость вращения транспортного средства, представленную в радианах в секунду.

общественная пустота setForwardAngularVelocity (Vector3 v)

Устанавливает скорость вращения транспортного средства, представленную в радианах в секунду.

общедоступное число с плавающей запятой getForwardVelocity()

Возвращает текущую скорость тела относительно его направления вперед.

общедоступное число с плавающей запятой getLateralVelocity()

Возвращает текущую скорость тела относительно его правого направления.

общедоступное число с плавающей запятой getVerticalVelocity()

Возвращает текущую скорость транспортного средства относительно его направления вверх.

общедоступный TCCAPlayer getParentPlayer()

Возвращает компонент TCCAPlayer, подключенный к этому телу.

общедоступный Vector3 getPosition()

Возвращает текущую позицию тела в мировом пространстве.

общедоступный кватернион getRotation()

Возвращает текущее вращение тела в мировом пространстве.

public void setPosition (позиция Vector3)

Устанавливает текущее положение тела в мировом пространстве.

public void setRotation (вращение кватерниона)

Устанавливает текущее вращение тела в мировом пространстве.

общественный недействительный перевод (смещение Vector3)

Добавляет значение «position» к текущей позиции тела.

public void Rotate (вращение кватерниона)

Добавляет значение «ротации» к текущему вращению тела.

публичная пустота, иммобилизация()

Устанавливает все скорости и крутящие моменты на ноль.

public void setParent (родительский элемент преобразования)

Устанавливает Transform, в котором должно содержаться это тело. В обычных обстоятельствах пользователь не должен вызывать эту функцию.

TCCAWheel

Создает необходимые объекты и компоненты для создания колеса с рулевым управлением и подвеской. Новые объекты создаются в объекте, содержащем сценарий TCCAPlayer.

публичная виртуальная пустота onCollisionStay (столкновение)

публичная виртуальная пустота onCollisionEnter (столкновение)

публичная виртуальная пустота onCollisionExit (столкновение)

публичная виртуальная пустота onTriggerStay (другое коллайдер)

публичная виртуальная пустота onTriggerEnter (другой коллайдер)

публичная виртуальная пустота onTriggerExit (другой коллайдер)

Вызывается из одноименных функций в TCCAWheelCollider. Вы можете создать собственный класс, расширяющий TCCAWheel, и переопределить вышеуказанные функции для обнаружения коллизий и триггеров.

public void инициализировать (родительский элемент TCCAPlayer)

Инициализирует кузов транспортного средства с указанным компонентом TCCAPlayer.

публичное обновление void (float deltaTime)

Обновляет параметры и физику колеса автомобиля. Вызывается один раз в каждом кадре в Update().

общедоступный TCCAPlayer getParentPlayer()

Возвращает компонент TCCAPlayer, подключенный к этому рулю.

public void setSpeedMultiplier (float m)

Значение, на которое умножается максимальная скорость колеса. По умолчанию — 1.

public void setAccelerationMultiplier (float m)

Значение, на которое умножается ускорение колеса. По умолчанию —

1. **общественный SphereCollider getCollider()**

Возвращает компонент SphereCollider колеса.

public bool isTouchingGround()

Возвращает, касается ли колесо земли в данный момент.

public void setSteering (плавающее значение)

Устанавливает направление поворота колеса, если рулевое управление включено. -1, чтобы повернуть налево, 1, чтобы повернуть направо.

public void setMotor (плавающее значение)

Устанавливает крутящий момент, приложенный к колесу, если у него включен двигатель. Принимает любое значение от -1 до 1.

public void setHandbrake(bool e)

Устанавливает, следует ли блокировать крутящий момент колеса.

общедоступное число с плавающей запятой getSteering()

Возвращает направление поворота, примененное к рулю в данный момент.

общедоступный поплавок getMotor()

Возвращает крутящий момент, приложенный к колесу в данный момент.

общедоступный Vector3 getForwardAngularVelocity()

Возвращает скорость вращения колеса, выраженную в радианах в секунду.

общественная пустота setForwardAngularVelocity (Vector3 v)

Устанавливает скорость вращения колеса, выраженную в радианах в секунду.

общедоступное число с плавающей запятой getForwardSpinVelocity()

Возвращает расчетную скорость колеса в метрах в секунду в зависимости от его крутящего момента.

public void setForwardSpinVelocity (float v)

Устанавливает крутящий момент колеса в соответствии с желаемой скоростью в метрах в секунду.

общедоступное число с плавающей запятой getForwardVelocity()

Возвращает текущую скорость колеса относительно его прямого направления.

общедоступное число с плавающей запятой getLateralVelocity()

Возвращает текущую скорость колеса относительно его правого направления.

общедоступное число с плавающей запятой getVerticalVelocity()

Возвращает текущую скорость колеса относительно его направления

вверх. **общедоступный Vector3 getRelativeVelocity()**

Возвращает текущую скорость колеса в локальных координатах.

публичная пустота, иммобилизация()

Устанавливает все скорости и крутящие моменты на ноль.

общедоступный Vector3 getPosition()

Возвращает текущую позицию колеса в мировом пространстве.

общедоступный кватернион getRotation()

Возвращает текущее вращение колеса в мировом пространстве.

public void setParent (родительский элемент преобразования)

Устанавливает Transform, в котором будет содержаться колесо. В обычных обстоятельствах пользователь не должен вызывать эту функцию.

Поиск неисправностей

Могу ли я использовать для колес другие коллайдеры, отличные от сферы по умолчанию?

К сожалению, этот контроллер был разработан для использования определенной комбинации твердых тел, коллайдеров и соединений для колес, поэтому вы не можете использовать собственные коллайдеры. Однако вы можете использовать любой тип коллайдера для кузова автомобиля, и колеса не будут с ним сталкиваться, поэтому, если вы хотите, чтобы колеса касались только земли, а не сторон другого объекта, вы можете создать коллайдер на кузове автомобиля, пересекающий колеса.

Как я могу обнаружить, когда колесо/тело сталкивается с пользовательским объектом?

Вы можете создать сценарий, расширяющий сценарий TCCAWheel или TCCAbody, и использовать его вместо этого. Затем вы можете переопределить следующие функции столкновений, которые вызываются одновременно с их аналогами из Rigidbody;

- публичная виртуальная пустота onCollisionStay (столкновение) { }
- public virtual void onCollisionEnter (столкновение) { }
- public virtual void onCollisionExit (столкновение) { }
- публичная виртуальная пустота onTriggerStay (Collider другое) { }
- public virtual void onTriggerEnter (Collider другое) { }
- публичная виртуальная пустота onTriggerExit (Collider другое) { }

Машина немного подпрыгивает при наезде на неровности дороги. Как я могу это исправить?

Попробуйте изменить значения параметра «Смещение контакта по умолчанию» на вкладке «Физика» в настройках проекта, чтобы минимизировать эффекты «призрачных столкновений». В идеале этот контроллер лучше всего работает на связанных сетках.

Как изменить положение и поворот автомобиля?

Вы можете установить положение и поворот автомобиля с помощью функций «TCCAPlayer.setPosition(Vector3)» и «TCCAPlayer.setRotation(Quaternion)». Они вернут объект-контейнер к абсолютному положению/повороту кузова транспортного средства и переместит контейнер к значениям, переданным в качестве параметров.

Если вам необходимо полностью сбросить настройки автомобиля, вы также можете вызвать функцию «TCCAPlayer.immobilize()», которая установит скорость автомобиля на ноль.

Вы также можете использовать функции «TCCAPlayer.translate(Vector3)» и «TCCAPlayer.rotate(Quaternion)», если вы просто хотите добавить значения к текущей позиции и повороту.

При рулевом управлении автомобиль сам слегка поворачивает влево и вправо.

Попробуйте поднять массу кузова и колес до тех пор, пока поворот не станет едва заметным.

Колеса не остаются на месте при слишком быстрой езде.

Попробуйте повысить жесткость (рессору) подвески до тех пор, пока дрожание не станет едва заметным.