

F-GEAR



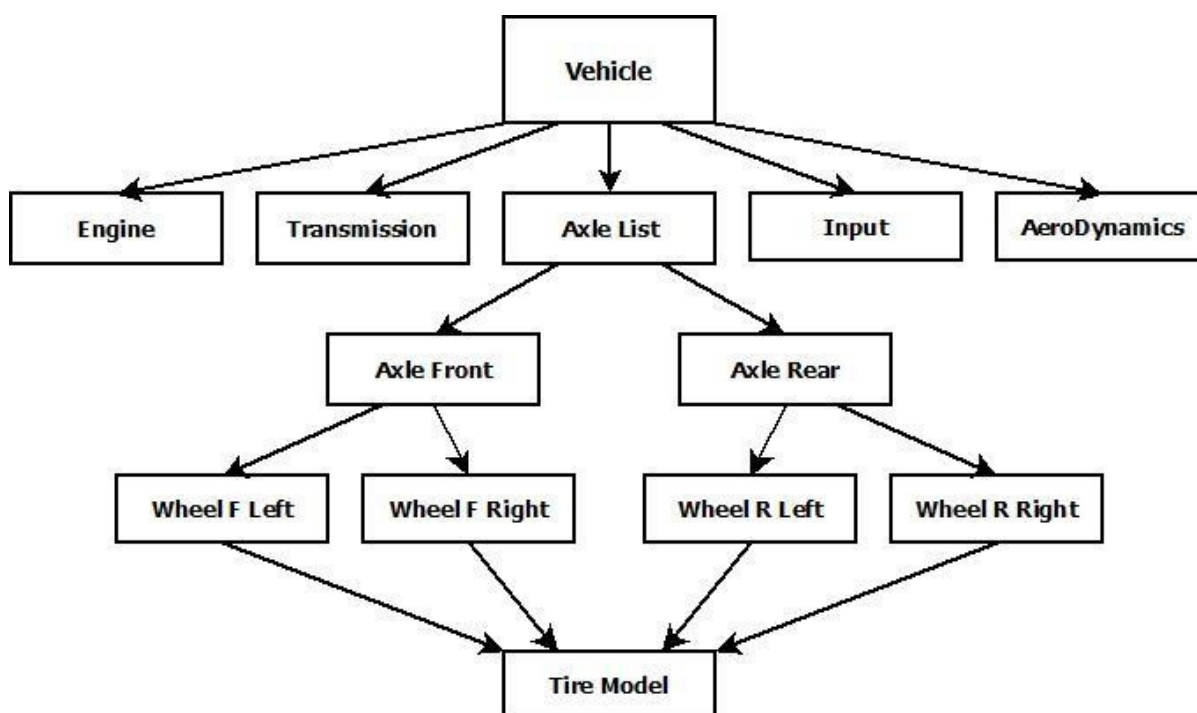
Обзор FGear

FGear - это пользовательское решение физики транспортных средств для unity, позволяющее создавать гоночные игры в стиле аркады/семиаркады. Пакет состоит из основного модуля и некоторых дополнительных функций, которые были использованы при создании демо-сцен. Пакет ориентирован в основном на 4-колесные автомобили, но возможны и многоосные варианты.

Основные возможности:

- Настольный или мобильный.
- Единый скриптовый компонент для управляемого транспортного средства.
- Полный исходный код.
- Регулируемые шаги интеграции.
- Упрощенные варианты моделей шин расеjka, расеjka96 и MF6.1.
- Поддержка 3D-колес (множественные лучи, сфера, выпуклый слепок).
- Простая модель двигателя с кривой крутящего момента.
- Настраиваемые автоматические/последовательные/ручные (h-паттерн) передачи.
- Варианты открытого, заблокированного и lsd-дифференциала.
- Рулевое управление Ackerman, развал-схождение.
- Регулируемое распределение крутящего момента и тормозов по осям.
- Простые пружинно-демпферные подвески с возможностью преднатяга.
- Компонент AeroDynamics для учета эффектов сопротивления и прижимной силы.
- Стандартный менеджер ввода с поддержкой клавиатуры, джойстика и колеса+шифтера.
- Такие средства обеспечения безопасности движения, как abs, asr, esp и антипробуксовочная система.
- Подробный интерфейс телеметрии.
- Базовая система сохранения/загрузки для быстрого создания прототипов.
- Различные установки и демонстрации образцов.

На приведенной ниже схеме показано, как устроен основной модуль:



- **Vehicle** : Здесь хранятся все компоненты автомобиля, и это единственный скрипт, который необходимо добавить к игровому объекту для создания управляемого автомобиля. Помимо управления компонентами, он также содержит реализацию большинства ассистентов вождения.
- **Двигатель**: Использует кривую крутящего момента и значения оборотов для создания мощности автомобиля.
- **Трансмиссия**: Это трансмиссия, которая может управляться вручную или использовать настраиваемые значения для автоматического переключения передач. Возможны Н-образное или последовательное переключение передач, а также простая реализация сцепления, которое автоматически управляется в последовательном режиме.
- **AeroDynamics** : Этот компонент выполняет две задачи. Во-первых, по мере ускорения автомобиля возникает сила сопротивления, а во-вторых, прикладывается прижимная сила. Они рассчитываются в соответствии с заданными параметрами и постоянным значением плотности воздуха.
- **Ввод**: Первая задача этого компонента - считывать единые значения входных осей и применять их к автомобилю, например, к дросселю или тормозам. Можно использовать клавиатуру, джойстик или колесо+переключатель. Опционально он изменяет входные сигналы рулевого управления в соответствии с некоторыми параметрами для обеспечения лучшего управления.
- **Оси** Основная функция моста - передача мощности. Крутящий момент двигателя передается на мост, а мост через дифференциал передает крутящий момент на колеса. Автомобиль имеет 2 или более осей, на одной оси закреплены 2 колеса.
- **Колеса** : Компонент колеса создает как силы в шинах, так и силы в подвеске. Для определения сжатия подвески используется один или несколько рейкастов, и рассчитывается сила, необходимая для отрыва автомобиля от земли. Для генерации поперечных и продольных сил используется модель шины. Сгенерированные силы прикладываются к кузову автомобиля, а также передают крутящий момент реакции на

двигатель.

- **Tire** : Wheels использует модель шины для расчета силы тяги. По умолчанию используется модель шины расейка96. Упрощенная расейка также является достойной и более дешевой в вычислениях. Наиболее продвинутой является модель шины MF6.1. Некоторые возможности, например, эффект развала-схождения, недоступны в простой модели. Выбор модели зависит от ваших требований. Подробнее о моделях шин можно прочитать в документации по CoreModule.
- **Дополнительные возможности** : Для примеров сцен используются некоторые дополнительные материалы. Их можно использовать для создания прототипов и обучения, но не рекомендуется применять в производстве. Вот эти дополнительные возможности:
- **AIController** : Это скрипт ai, который управляет автомобилями в гоночном примере. Этому скрипту необходим экземпляр SplineTool в сцене, который представляет собой гоночную линию.
- **ArcadeAssists** : Этот скрипт содержит несколько ненастоящих, но эффективных помощников при вождении. Они могут быть полезны в играх аркадного стиля.
- **Car Manager** : Это утилита, позволяющая пользователю легко переключаться между автомобилями в сцене. Кроме того, она использует скрипт SplineTool для расчета расположения автомобилей и отображения их положения на экране.
- **Эффекты** : Прежде всего, этот скрипт добавляет звуковые эффекты двигателя и заноса, а также сетчатый эффект TireSkid и частицы дыма. Кроме того, скрипт содержит пример системы, которая определяет физический материал или текстуру местности при ударе колеса о поверхность и изменяет значения трения в соответствии с заданными параметрами.
- **ForceFeedback** : Это пример скрипта, использующего LogitechSteeringWheelSDK для добавления некоторых простых эффектов обратной связи. Для использования этого сценария необходимо установить sdk.
- **Gauge UI** : Набор активов и скрипт, отображающий прикольный hud с индикацией скорости, оборотов и текущей передачи.
- **JoystickVibration** : Это пример скрипта, который использует XInput для добавления эффекта вибрации к джойстикам. Для его использования необходимо установить библиотеку XInput.
- **MinimapTool** : Использует SplineTool и LineRenderer для создания визуального представления трассы. Кроме того, для отображения карты и автомобилей на экране используется ортографическая камера.
- **MobileInput** : Это сценарий, который управляет мобильными входами в мобильной выборке.
- **OrbitCamera** : Простая настраиваемая камера, которая вращается вокруг заданной трансформации. Она также отображает простой пользовательский интерфейс, позволяющий изменять ее настройки в режиме игры.
- **SplineTool** : Этот скрипт строит сплайн из дочерних преобразований присоединенного игрового объекта. Построенные кривые рисуются в редакторе в виде гизмо.
- **Статистика** : Постоянно измеряет такие показатели, как 0-100, 0-200, 100-0, а также, если предусмотрен SplineTool, вычисляет и отображает текущее и последнее время прохождения круга.
- **Suspension Constraint** : Добавляет настраиваемый шарнир к каждому колесу, который останавливает сжатие подвески на жестком пределе.
- **RuntimeSample** : Создает простое транспортное средство из кода, проверьте

runtime.scene, чтобы увидеть его в действии.

- **NetworkSample** : Использует сеть photon unity для установления соединения и инстанцирует транспортное средство с поддержкой сети. Для его использования необходимо установить библиотеку PUN2.
- **NetworkSync** : Прикрепляется к транспортному средству для синхронизации по сети. Для его использования необходимо установить библиотеку PUN2.

Для быстрого начала работы перейдите на страницу **1-Quickstart** или см. <https://youtu.be/WzAV3pWq70U>.

Подробное описание основного модуля приведено в разделе

2-CoreModule. Подробное описание дополнительных

модулей приведено в разделе **3-Extras**.

Получить api документ* с сайта <https://www.dropbox.com/s/nn7m1zshcygui57/api.chm?dl=0>

За любой поддержкой обращайтесь к нам по адресу lazybitgames@gmail.com.

*Если содержимое api документов не отображается, щелкните правой кнопкой мыши на файле .chm и разблокируйте файл в свойствах:

