

ТАЛЛИНСКИЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
Факультет инфотехнологий
Институт информатики

ISS40LT

Александр Кузьмин 120763IASB

МОДЕЛИРОВАНИЕ ОБРАТНОГО МАЯТНИКА В ВИРТУАЛЬНОЙ РЕАЛЬНОСТИ

Бакалаврская работа

Супервизор: Алексей Тепляков
MSc (2011)
Инженер

Tallinn 2016

TALLINNA TEHNIKAÜLIKOOL
Infotehnoloogia teaduskond
Informaatikainstituut

ISS40LT

Aleksandr Kuzmin 120763IASB

PÖÖRDPENDLI MODELLEERIMINE VIRTUAALREAALSUSES

Bakalaureusetöö

Juhendaja: Aleksei Tepljakov
MSc (2011)
Insener

Tallinn 2016

Autorideklaratsioon

Kinnitan, et olen koostanud antud lõputöö iseseisvalt ning seda ei ole kellegi teise poolt varem kaitsmisele esitatud. Kõik töö koostamisel kasutatud teiste autorite tööd, olulised seisukohad, kirjandusallikatest ja mujalt pärinevad andmed on töös viidatud.

Autor: Aleksandr Kuzmin

25.05.2016

Аннотация

Моделирование обратного маятника в виртуальной реальности.

Данная работа посвящена построению модели обратного маятника, конвертации, а также, последующей реализации модели в виртуальной реальности.

Модель будет построена в системе автоматизированного проектирования AutoCAD, разработанной компанией Autodesk. Анимация будет реализована на игровом движке Unreal Engine 4, поддерживаемым компанией Epic Games. Реализация модели будет происходить, при помощи математической модели обратного маятника в MATLAB. Для конвертации трехмерной модели, будет использована программная система 3Ds Max, до разработанная компанией Autodesk.

В ходе работы автор познакомит со всеми, вышеперечисленными программами, опишет основные проблемы построения, конвертации и реализации данной модели. Будут рассмотрены и объяснены те, или иные решения, как к ним пришел автор и почему именно они получили право на существование. Работа также содержит чертежи модели и алгоритмы вычисления поведения анимации.

Созданная автором анимированная модель обратного маятника в виртуальной реальности, может быть использована, как наглядный пример того, как ведет себя модель в реальном времени, что поможет как с пониманием физических свойств модели, так и для решения задач, связанных с данной тематикой.

За основу будет взята, реально существующая модель обратного маятника, собранная компанией Inteco.

Дипломная работа написана на русском языке и содержит текст на 39 страниц, 6 глав, 26 картинок, 1 таблиц.

Annotatsioon

Pöördpendli modelleerimine virtuaalreaalsuses.

See töö on pühendatud pöördpendli mudeli järjestusele, teisendamisele ja samuti järgneva mudeli realiseerimises virtuaalses reaalsuses.

Mudel saab olema ehitatud automatiseeritud projekteerimis süsteemis AutoCAD, mis on väljatöötatud Autodesk firma poolt. 3D mudeli teisendamiseks, saab olema kasutatud 3Ds Max tarkvarasüsteem, mis on lõpuni ehitatud Autodesk firma poolt. Animatsioon saab olema realiseeritud Unreal Engine 4 alusel, firma Epic Games poolt. Mudeli realiseerimine saab olema tagatud pöördpendli matemaatilise mudeli abiga MATLAB-is.

Töö käigus tutvustab autor kõiki eelnimetatud programme, kirjeldab peamisi järjestus (построения!), teisendamis ja realiseerimis probleeme selles mudelis. Saavad olema läbivaadatud ja seletatud ühed või teised lahendused, kuidas nendeni on jõudnud autor ja miks just need lahendused said olema tehtud. Töö omab endas ka mudeli jooniseid ja animatsiooni käitumise arvutuste algorütme.

Autori poolt loodud pöördpendli animeeritud mudel virtuaalses reaalsuses, võib olla kasutatud kui näide sellest, kuidas käitub mudel käivas ajas, mis aitab aru saada nii mudeli füüsilisi omadusi kui ülesannete lahendamist mis on seotud selle temaatikaga.

Aluseks saab olema võetud, reaalselt olemasolev pöördpendli mudel, mis on ehitatud firma Inteco poolt.

Lõputöö on kirjutatud vene keeles keeles ning sisaldab teksti 39 leheküljel, 6 peatükki, 26 joonist, 1 tabelit.

Abstract

Implementation of an Inverted Pendulum Model in Virtual Reality

This work is dedicated to the construction of the inverted pendulum model, converting and the following implementation of the model in a virtual reality.

The model will be built in the computer-aided design system AutoCAD, developed by Autodesk. Animation will be implemented on the game engine Unreal Engine 4, supported by Epic Games. The implementation of the model will take place by applying a mathematical model of inverted pendulum in MATLAB. Computer software system 3Ds Max, further developed by Autodesk, will be used to convert the three-dimensional model.

In the course of this work author is going to give sight of all the above programs and describe the main problems of construction, converting and implementation of the current model. Various courses of solutions will be considered and explained as well as the way author came to them and why do they have the right to exist. The work also contains graphical representations of models and calculation algorithms of animation's behavior.

Animated inverted pendulum model in virtual reality, created by author, could be used as demonstrative example of model behavior in the real time, which could help not only with understanding of the physical nature of the model, but also with solving problems, related the current subject.

The existing inverted pendulum model, assembled by Inteco company, will be taken as a base.

The thesis is in Russian and contains 39 pages of text, 6 chapters, 26 figures, 1 tables.

Lühendite ja mõistete sõnastik

PIVOT POINT	Точка, центр поворота и масштабирования объекта
COMMUNITY	Сообщество пользователей
2D	Двухмерное представление объекта
3D	Трехмерное представление объекта
ПСК	Пиктограмма системы координат
ACTOR	Родительский класс, содержащий все объекты, имеет координаты

Sisukord

1 Введение	13
1.1 Цели	14
1.2 Проблемы	14
2 Система моделирования AutoCAD	15
2.1 Инструменты трехмерного моделирования	16
2.2 Инструменты двухмерного моделирования	17
2.3 Группировка объектов и слои	17
2.4 Модель обратного маятника	18
2.4.1 Вертикальная планка	19
2.4.2 Рельсовая пара	20
2.4.3 Система ведущих звезд	20
2.4.4 Каретка	21
2.4.5 Итоговая модель	22
3 Конвертация модели из AutoCAD в Unreal Engine 4	24
3.1 Экспорт на прямую	25
3.2 Экспорт при помощи программной системы 3Ds Max	26
3.2.1 Импорт всей модели	27
3.2.2 Централизация pivot point	27
4 Добавление анимации на игровом движке Unreal Engine	29
4.1 Анимация передвижения каретки	30
4.2 Анимация поворота спицы	31
4.3 Направленное движение спицы	31
5 Реализация модели при помощи программы MATLAB	33
5.1 Математическая модель обратного маятника	34
5.2 Визуальная модель обратного маятника	34
6 Вывод	36
Kokkuvõte	37
Summary	38
Источники	39

Дополнение – Чертежи модели обратного маятника, спроектированные в системе AutoCAD.....	40
---	----

Жooniste loetelu

Рисунок 1. Логотип системы автоматизированного моделирования AutoCAD. ...	15
Рисунок 2. Главная панель инструментов, при трехмерном моделировании в системе AutoCAD.	16
Рисунок 3. Главная панель инструментов, при двухмерном моделировании в системе AutoCAD.	17
Рисунок 4. Диспетчер слоев в системе AutoCAD.....	18
Рисунок 5. Реальная модель обратного маятника.	19
Рисунок 6. Визуализация вертикальной планки трехмерной модели в AutoCAD.	19
Рисунок 7. Визуализация рельсовой пары в трехмерной модели в AutoCAD.	20
Рисунок 8. Двухмерная фигура при создании сложного трехмерного объекта (красный треугольник).....	21
Рисунок 9. Визуализация системы ведущих звезд в трехмерной модели в AutoCAD.....	21
Рисунок 10. Визуализация каретки в трехмерной модели в AutoCAD.	22
Рисунок 11. Визуализация полной трехмерной модели обратного маятника в AutoCAD.....	23
Рисунок 12. Вид модели при экспорте из AutoCAD.	25
Рисунок 13. Вид модели при импорте в Unreal Engine 4.	25
Рисунок 14. Логотип системы создания и редактирования трехмерной графики 3Ds Max.	26
Рисунок 15. Вид модели при импорте в 3Ds Max.	27
Рисунок 16. Вид модели после централизации pivot point.	27
Рисунок 17. Инструмент TS_Tools для 3Ds Max.....	28
Рисунок 18. Логотип игрового движка Unreal Engine 4.	29
Рисунок 19. Логика относительного перемещения каретки.....	30
Рисунок 20. Логика перемещения каретки.	30
Рисунок 21. Анимационная логика поворота спицы.	31
Рисунок 22. Логика относительного направленного перемещения каретки.	32
Рисунок 23. Логика направленного перемещения каретки.	32

Рисунок 24. Логотип пакета прикладных программ MATLAB.	33
Рисунок 25. Передатчик структуры данных в среде Simulink.	34
Рисунок 26. Приемник структуры данных в среде Unreal Engine 4.	35

Tabelite loetelu

Таблица 1. Таблица экспортных форматов системы AutoCAD [4]. 24

1 Введение

На сегодняшний день, каждый человек так или иначе имел дело с отдельными проявлениями виртуальной реальности, порой сам этого не подозревая. Будь это работа в графическом редакторе, или же простой разговор по телефону, через звуковую виртуальную реальность. Произошло это во многом благодаря стремительно развивающимся компьютерным технологиям, что позволило создать новый феномен нашего времени, который в научной литературе получил название “Virtual reality”.

Погружаясь в виртуальную реальность, человек перестает быть простым наблюдателем, где он может не только смотреть на образы, но и создавать их, влиять на их поведение, действовать самостоятельно. Такую возможность не могли предоставить прочтение книги, просмотр фильмов или созерцание картины. Поэтому интерес к данной теме, не будет утихать еще очень долго.

Будучи студентом, решая задачи системы обратного маятника в программе MATLAB, я до последнего не мог представить, что реально из себя представляет модель. Простой картинке не было достаточно для передачи всей информации об объекте исследования. Поэтому при выборе задачи для дипломной работы, я отнесся к данной теме очень положительно, так как анимированная модель сможет передать все тонкости поведения.

Данная дипломная работа послужит хорошим обучающим материалом для введения в трехмерное моделирование, и последующее анимирование модели. Работа не только познакомит с графическими редакторами, но и предоставит интересные решения тех, или иных проблем, связанных с данными программами. Будут рассмотрены ошибки, которым уделяется мало внимания, и их последующее решение.

1.1 Цели

Данная дипломная работа преследует следующие цели:

- Создание трехмерной модели в системе моделирования AutoCAD,
- Конвертация модели, при помощи программной системы 3Ds Max,
- Добавление анимации на игровом движке Unreal Engine 4,
- Реализация модели при помощи программы MATLAB.

1.2 Проблемы

В ходе изучения использованных в работе программ, были выявлены следующие проблемы:

- Построение сложных, не тривиальных, объектов для достижения реализма с исходной моделью от Intesco,
- Конвертация форматов из одной системы в другую,
- Централизация опорных точек (pivot point),
- Разработка алгоритмов поведения анимации.

2 Система моделирования AutoCAD

AutoCAD - двух- и трёхмерная система автоматизированного проектирования и черчения, разработанная компанией Autodesk, первая версия которой, была выпущена в 1982 году. Является самой популярной системой проектирования и черчения. Насчитывает около 6 млн пользователей [1].

Ранние версии оперировали примитивным набором инструментов, такие как круги, линии, дуги и текст. Однако на современном этапе, программа включает в себя полный перечень средств, для трехмерного моделирования, что позволяет создавать модели различной степени сложности. Данная программа отлично подходит для моделирования системы обратного маятника.

Главные причины выбора системы AutoCAD:

- Распространение на 18 языках, включая русский,
- Набор инструментов для комплексного трёхмерного моделирования,
- Упрощенная документация, большое 'community',
- Возможность персонализации интерфейса,
- Студенческая лицензия (бесплатная подписка).

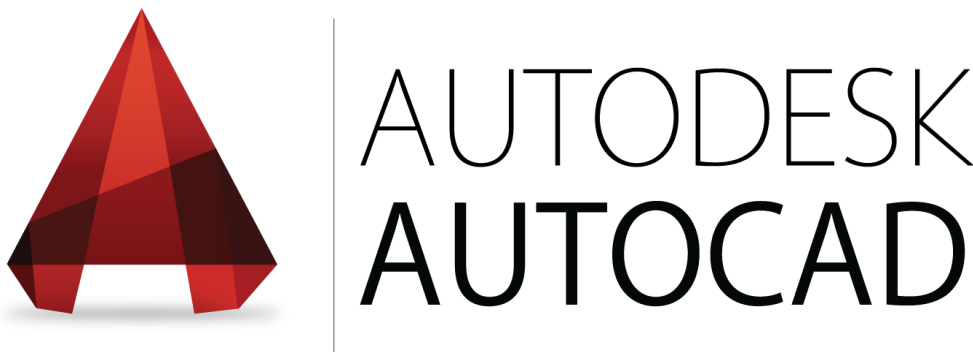


Рисунок 1. Логотип системы автоматизированного моделирования AutoCAD.

2.1 Инструменты трехмерного моделирования

Трехмерные модели в AutoCAD бывают трех типов: каркасные, поверхностные и твердотельные. Объединение разных типов объектов в одной чертеже не рекомендуется из-за ограниченности возможности преобразование одного объекта в другой.

- Каркасная модель – представлена только ребрами граней, представляет собой прозрачный объект.
- Поверхностная модель – представлена не только ребрами, но и гранями, может быть представлена сеткой, позволяет скрывать невидимые части объекта.
- Твердотельная модель – занимает все пространство, описанное его гранями.

AutoCAD имеет богатый выбор инструментов для построения и редактирования трехмерных объектов. Однако самым популярным способом является построение сложных деталей за счет совокупности примитивных объектов, таких как ящик и цилиндр. При этом, зачастую, есть необходимость прибегать к построению двухмерной модели. Объект можно рассмотреть со всех сторон, посмотреть в разрезе или сечении.

Для манипуляции с 3D телами существует панель редактирования тела, которая позволяет подгонять объекты по размерам, объединять, разрезать и смещать.

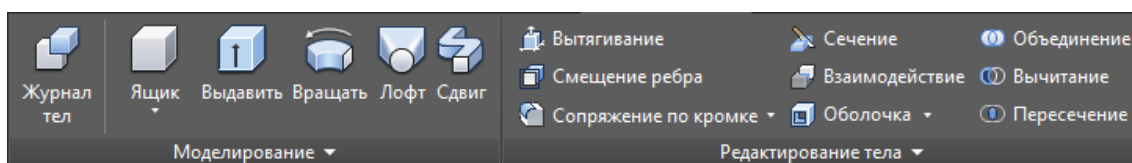


Рисунок 2. Главная панель инструментов, при трехмерном моделировании в системе AutoCAD.

Более сложные фигуры, такие как 'звезда' и рельсы были построены за счет инструментов двухмерного моделирования AutoCAD, что позволяет создавать более точные объекты, что не всегда можно сделать с 3D телом.

2.2 Инструменты двумерного моделирования

Для черчения и моделирования используются практически одни и те же принципы, поэтому, не зная основ черчения, невозможно грамотно построить трехмерную модель. Основными инструментами черчения являются: отрезок, круг, дуга и прямоугольник. Из них можно построить двумерную фигуру любой сложности.

Для редактирования двумерных объектов предоставляются такие инструменты как: перенести объект, копировать, повернуть, отзеркалить, обрезать и создать массив.

Данного набора инструментов вполне достаточно для построения полноценного чертежа.

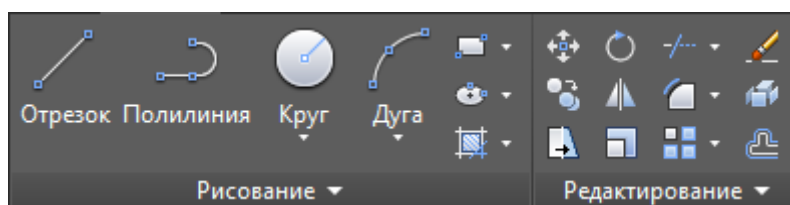


Рисунок 3. Главная панель инструментов, при двумерном моделировании в системе AutoCAD.

2.3 Группировка объектов и слои

Когда примитивов на чертеже становится слишком много, их нужно группировать в зависимости от свойства или местоположения объекта. Существует 2 способа объединить объекты: группа(Group) и слой(Layer).

Каждый примитив чертежа может иметь свойства, такие как цвет, вес(толщина) линии, тип линии и так далее. Для того, чтобы группировать объекты относительно общих свойств у AutoCAD есть отдельный инструмент - Layer(Слой). Слой также является объектом и может иметь присущие ему свойства. Изначально пользователю предоставляется дефолтный слой – Layer0. Остальные слои зависят только от фантазии пользователя.

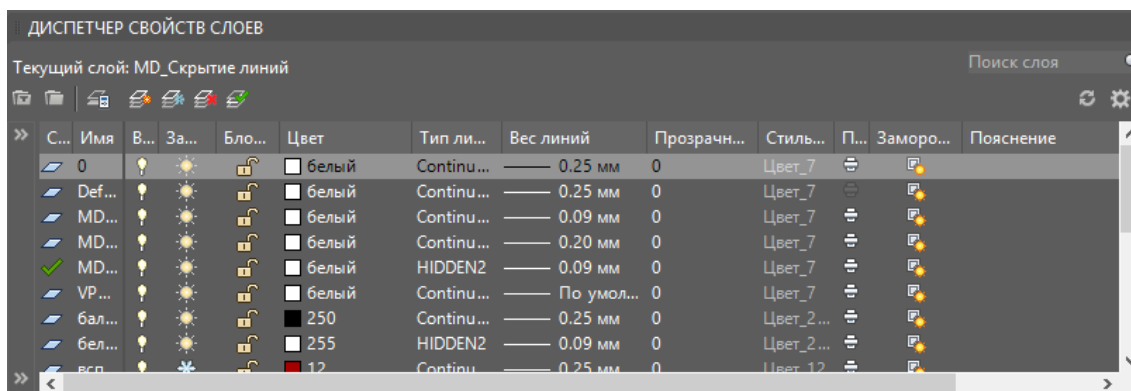


Рисунок 4. Диспетчер слоев в системе AutoCAD.

Группы в AutoCAD предназначены для объединения объектов с разными свойствами, расположенных на разных слоях. Группы являются удобным инструментом для ускорения задач выбора, перемещения и редактирования объектов. Операция группировки предназначена для фиксации взаимного расположения выделенных объектов. Группа также может состоять из нескольких групп.

Без группировки объектов и распределения их по слоям, в зависимости от свойств, является важным умением при работе в AutoCAD. Если пренебрегать данными инструментами, то количество объектов станет настолько огромным, что с проектом невозможно будет работать.

2.4 Модель обратного маятника

В данной главе будет продемонстрирована полностью спроектированная модель в системе автоматизированного проектирования AutoCAD. За основу будет взята, реально существующая модель обратного маятника, собранная компанией Intesco. Модель находится в лаборатории Таллиннского Технического Университета.

Все измерения были проведены с точностью до 1 сантиметра. Цвета и материал подобраны в соответствии с реальной моделью.

Чертежи спроектированной модели будут добавлены в дополнение к дипломной работы.

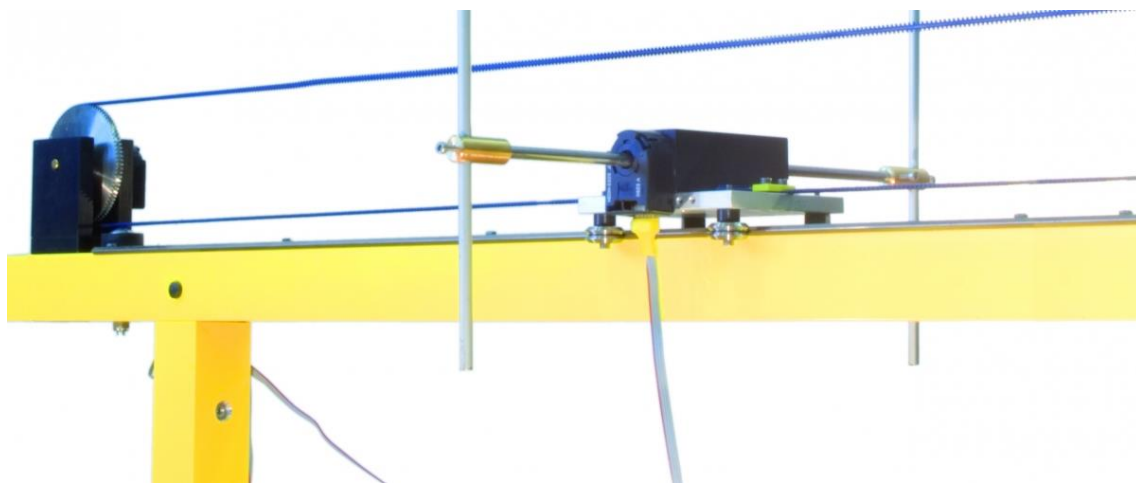


Рисунок 5. Реальная модель обратного маятника.

2.4.1 Вертикальная планка

Один из самых простых групп объектов, было создание(проектирование) вертикальной планки. Она полностью состоит из вытянутых ящиков. Для достижения большей реалистичности, были добавлены черные заглушки на концах ножек и балки.

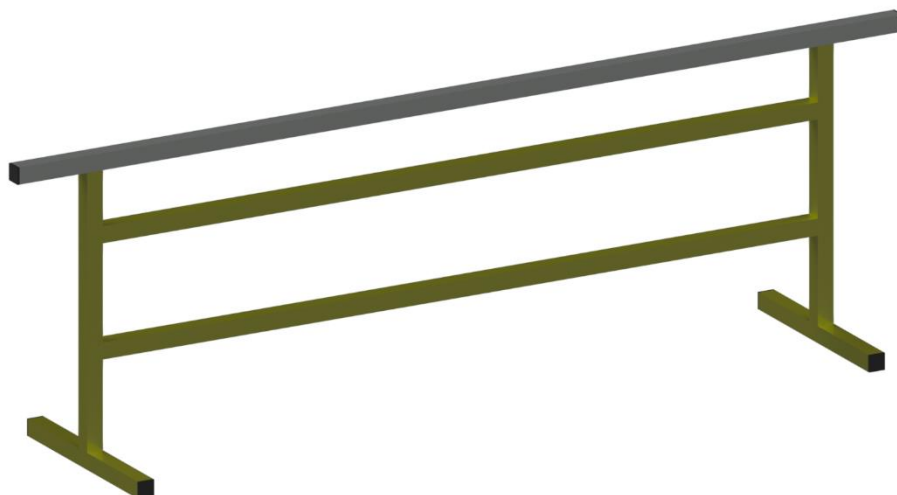


Рисунок 6. Визуализация вертикальной планки трехмерной модели в AutoCAD.

2.4.2 Рельсовая пара

В отличие от вертикальной планки, для построения рельс использовать вытянутый ящик было не достаточно. По краям рельсовых пар находятся вытянутая плоскость равнобедренного треугольника. После, все объекты были объединены в один, что сделало рельсы целостным объектом.

Это был первый случай использования двумерных фигур и инструмента сдвига трехмерной модели для построения более сложной фигуры, что подтверждает, что сложные детали состоят из совокупности примитивных.

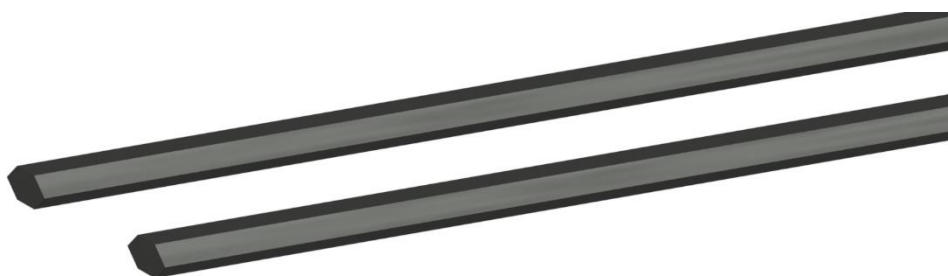


Рисунок 7. Визуализация рельсовой пары в трехмерной модели в AutoCAD.

2.4.3 Система ведущих звёзд

Одной из самых интересных частей модели обратного маятника была система ведущих звезд. Инструменты 3D моделирования не позволяли сделать ее быстро и просто. Было принято решение реализации через чертеж двумерной звезды.

Для начала, был нарисован простой равнобедренный треугольник без основания. После, был создан круговой массив данного треугольника. В итоге получилась целостная фигура, являющаяся гранью звезды.



Рисунок 8. Двухмерная фигура при создании сложного трехмерного объекта (красный треугольник).

Чтобы получить ребро звезды, достаточно сделать поверхность ее грани. Используя инструмент вытягивания, получается настоящий 3D объект, объёмная звезда. После копирования нового объекта, вставляем между звездами цилиндр, в итоге получается ведущая система звезд, схожая с реальным объектом.

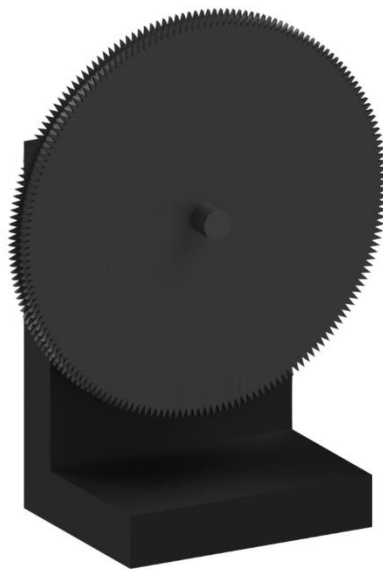


Рисунок 9. Визуализация системы ведущих звёзд в трехмерной модели в AutoCAD.

2.4.4 Каретка

Каретку можно разделить на три части: основание каретки, колесные пары, спицы. Построение основания каретки не представляет ничего сложного, оно состоит из вытянутых ящиков и одного цилиндра. Спицы состоят только из цилиндров. Самый сложный элемент этого объекта является колесная пара.

Колесные пары должны быть не только привязаны к каретке, но и еще идеально подходить к рельсам, которые расположены на вертикальной балке. В то время, как сама каретка состоит в связи с ведущими звездами через ремень. В данном случае помогут только точные измерения реальной модели, и точное расположение объектов виртуальной модели, в системе координат.

Колесная пара представляет из себя цилиндр, вокруг которого сдвинута плоскость треугольника по оси данного цилиндра.



Рисунок 10. Визуализация каретки в трехмерной модели в AutoCAD.

2.4.5 Итоговая модель

Благодаря точным измерениям, удалось достичь целостности всей модели, никакие объекты не наезжают друг на друга и не висят в воздухе. Вся модель, не смотря на ее сложность, была построена из простых объектов с использованием минимального количества инструментов для редактирования. AutoCAD имеет внутреннюю документацию, что способствует быстрому обучению тому или иному инструменту. Инновационная 3D визуализация позволила без проблем рассмотреть объекты со всех сторон. Группировка объектов и распределение слоев позволило упростить редактирование, перемещение и поиск, конкретных деталей. Именно поэтому AutoCAD является лучшей системой двухмерного и трехмерного моделирования.



Рисунок 11. Визуализация полной трехмерной модели обратного маятника в AutoCAD.

3 Конвертация модели из AutoCAD в Unreal Engine 4

Основным форматом файла AutoCAD является DWG. Единственный формат, который доступен для импорта в Unreal Engine 4 - FBX.

- DWG – бинарный формат файла, используемый для хранения двухмерных (2D) и трёхмерных (3D) проектных данных и метаданных [2].
- FBX - используется для обеспечения совместимости различных программ трехмерной графики [3].

3D DWF (*.dwf) 3D DWFx (*.dwfx)	Autodesk Design Web Format
ACIS (*.sat)	ACIS solid object file
Bitmap (*.bmp)	Device-independent bitmap file
Block (*.dwg)	Drawing file
DXX Extract (*.dxx)	Attribute extract DXF file
Encapsulated PS (*.eps)	Encapsulated PostScript file
IGES (*.iges; *.igs)	IGES files
FBX files (*.fbx)	Autodesk FBX files
Lithography (*.stl)	Solid object stereolithography file
Metafile (*.wmf)	Microsoft Windows Metafile
V7 DGN (*.dgn)	MicroStation DGN file
V8 DGN (*.dgn)	MicroStation DGN file

Таблица 1. Таблица экспортных форматов системы AutoCAD [4].

Конвертация трехмерной модели обратного маятника будет проходить через экспорт формата FBX, с последующим импортом на движок Unreal Engine 4.

3.1 Экспорт на прямую

Экспорт и импорт файлов будет происходить напрямую, не используя сторонние программы. AutoCAD имеет возможность экспорта модели в формате FBX, Unreal Engine 4 способен обрабатывать данный формат.

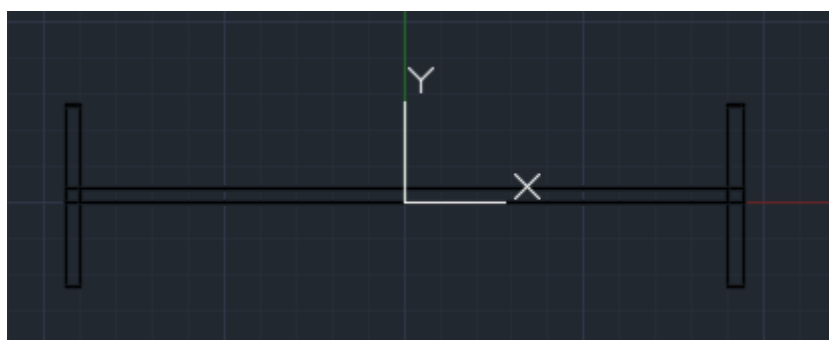


Рисунок 12. Вид модели при экспорте из AutoCAD.

При импорте FBX файла на движок Unreal Engine 4, начинают возникать проблемы с местонахождением pivot point, точка возвращается в мировое положение всей модели, несмотря на то, что при экспорте точка была централизована с объектом, из-за чего, начинает возникать проблемы с добавлением анимации и перемещением объекта на движке. Ротация начинает происходить не вокруг центра объекта, а вокруг pivot point.

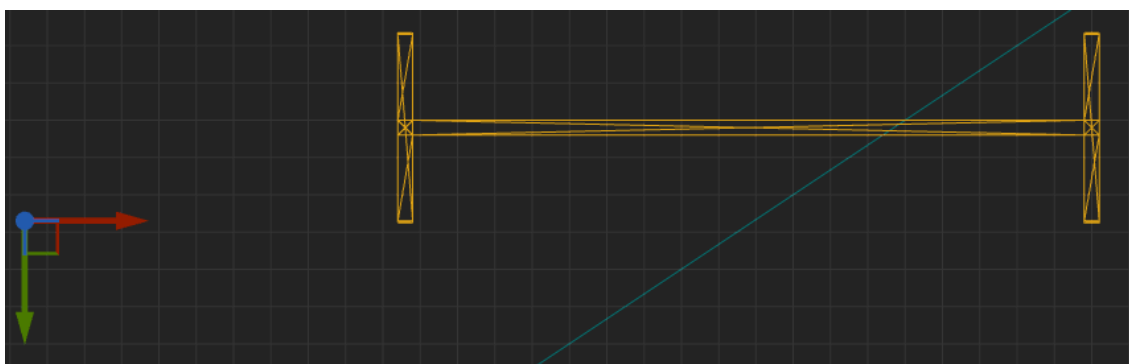


Рисунок 13. Вид модели при импорте в Unreal Engine 4.

Существует 2 варианта решения этой проблемы:

- Перенос каждого объекта к центру ПСК(AutoCAD)
- Перенос pivot point к центру объекта (Unreal Engine 4)

Первый вариант подойдет для простых моделей, но, когда объектов слишком много, задача становится не выполнимой, также будет теряться общее местоположение объектов.

Второй вариант не подходит для сложных моделей. Так как pivot point будет смещаться только относительно данного объекта, относительно других, его pivot point останется на прежнем месте.

3.2 Экспорт при помощи программной системы 3Ds Max

3Ds Max - полнофункциональная профессиональная программная система для создания и редактирования трёхмерной графики и анимации, разработанная компанией Autodesk [5]. Так же как и AutoCAD доступен по студенческой(бесплатной) лицензии. 3Ds Max также способен работать с форматом FBX. Способ конвертации посредством программы 3Ds Max можно разделить на три части: импорт всей модели, централизация pivot point относительно каждого объекта, экспорт каждого объекта в отдельный FBX файл.



Рисунок 14. Логотип системы создания и редактирования трехмерной графики 3Ds Max.

3.2.1 Импорт всей модели.

Модель не нужно разделять на отдельные объекты. В отличие от Unreal Engine 4, 3Ds Max сам найдет все объекты данной модели из формата FBX. Для экспорта понадобится всего один FBX файл.

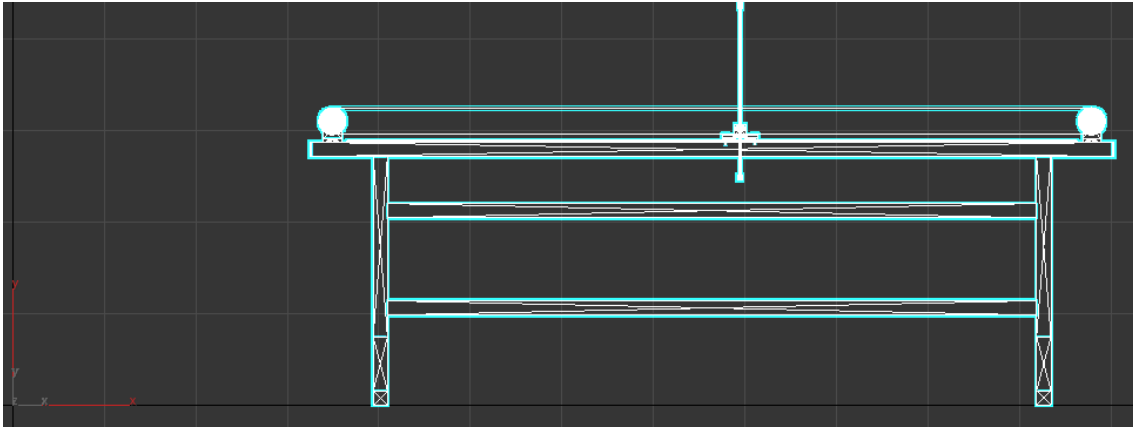


Рисунок 15. Вид модели при импорте в 3Ds Max.

Однако начинают возникать проблемы с передачей материалов объекта, поэтому эту часть лучше предоставить конечному варианту модели, в самой среде Unreal Engine. Pivot point всех объектов будет находиться в мировой точке, благодаря чему, модель не потеряет целостность.

3.2.2 Централизация pivot point

Программа 3Ds Max имеет родной инструмент, с помощью которого можно централизовать все pivot point'ы, относительно каждого отдельного объекта.

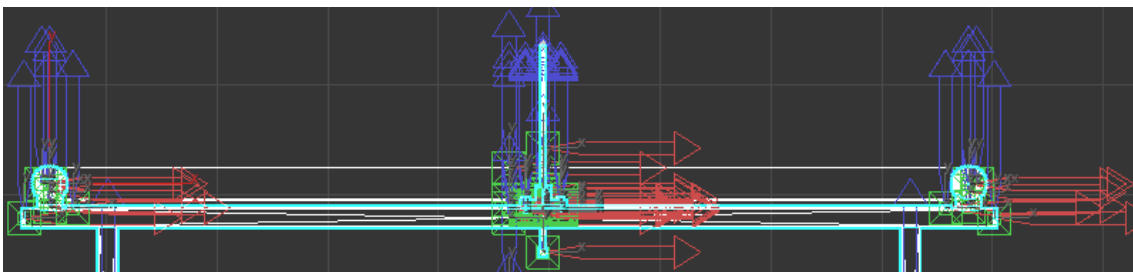


Рисунок 16. Вид модели после централизации pivot point.

Однако данные изменения будут работать только в 3Ds Max'е. При экспорте объектов в формат FBX, pivot точки сбрасываются на мировую. Для решения этой проблемы существует отдельный инструмент, не зависящий от 3Ds Max - TS_Tools.

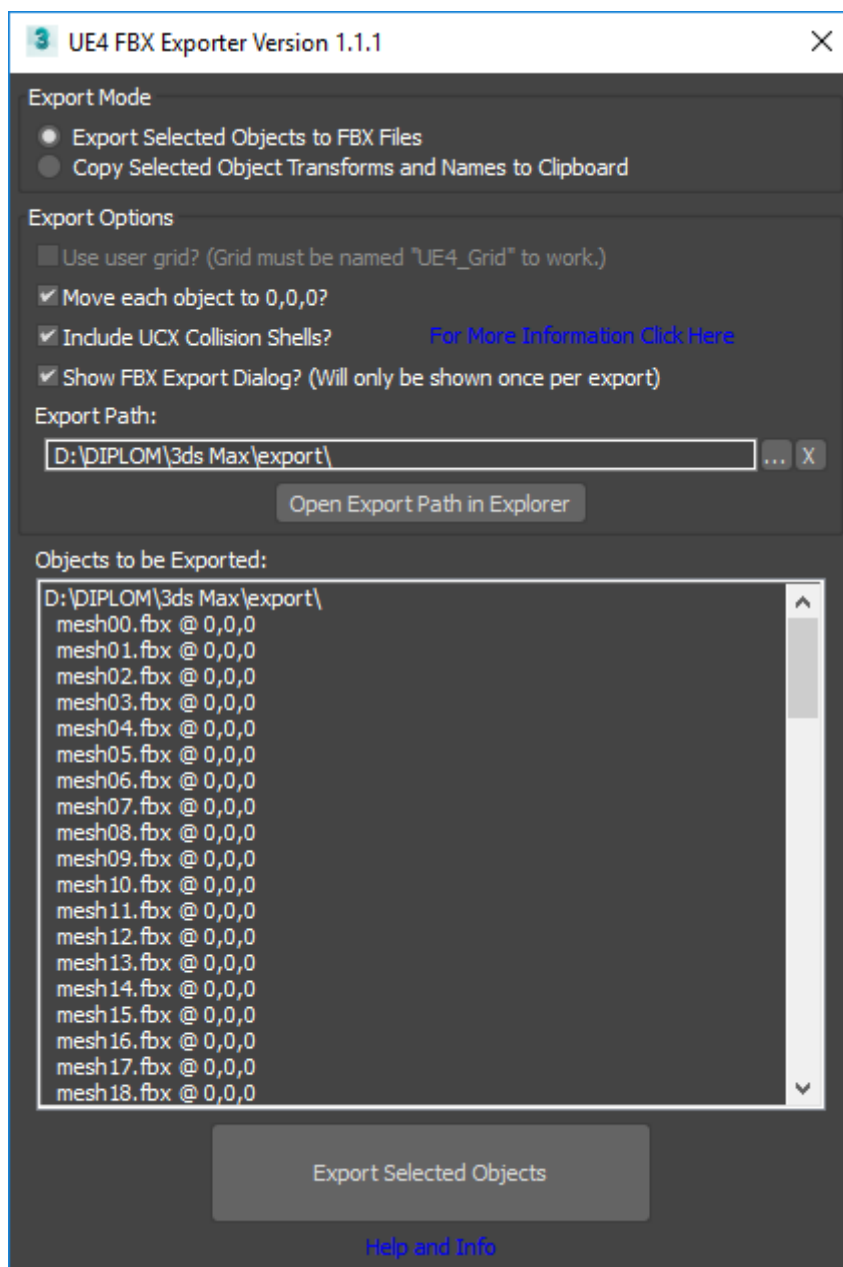


Рисунок 17. Инструмент TS_Tools для 3Ds Max.

После инсталляции скрипта, появится возможность экспортировать все объекты с централизованными pivot точками. Также этот скрипт позволяет создать список 'актеров' для импорта в UE4, с сохранением местоположения всех объектов относительно друг друга.

4 Добавление анимации на игровом движке Unreal Engine

Unreal Engine — игровой движок, разрабатываемый и поддерживаемый компанией Epic Games. Первая игра на этом движке была создана в 1998 году. Движок написан на языке C++, что дает возможность работать с большинством операционных систем [6]. В данной работе был использован движок версии 4.

Движок позволяет писать логику на языке C++, а также с помощью визуальной системы программирования – blueprint. Это визуальный скриптовый язык, который дает такую возможность, как писать логику, не зная ни одного языка программирования. Несмотря на то, что это визуальная система, blueprint позволяет создавать почти все что угодно, решать, как простые задачи, так и сложные.

В данной работе логика анимации была полностью построена с помощью blueprint'ов. Вся анимация опирается на 3 объекта: движение каретки, поворот спицы, направленное движение спицы.



Рисунок 18. Логотип игрового движка Unreal Engine 4.

4.1 Анимация передвижения каретки

Движение каретки происходит только вдоль оси X, координаты Z и Y являются константными значениями, которые не изменяются во время всей анимации модели. Логика принимает на вход: длину балки (x_real_length) и текущую позицию каретки(x), относительно этой длины.

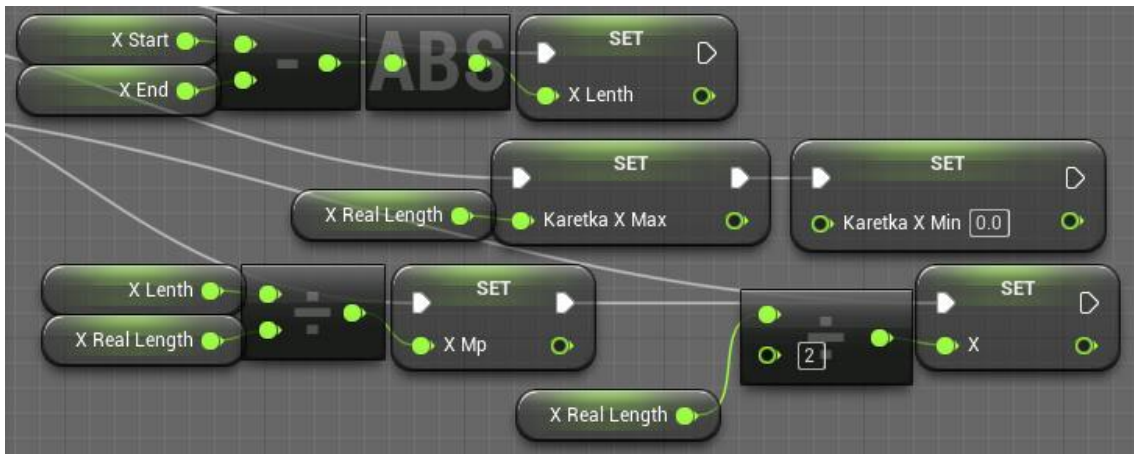


Рисунок 19. Логика относительного перемещения каретки.

При запуске анимации, вычисляется длинна трехмерной балки(x_length). Ставятся максимальная(karetka_x_max) и минимальная(karetka_x_min) длинна пути каретки. Чтобы перемещение каретки соответствовало реальному, создается переменная, отвечающая за коэффициент относительности(x_mp).

При изменении позиции X, каретка автоматически смещается на нужную координату относительно реальной модели.

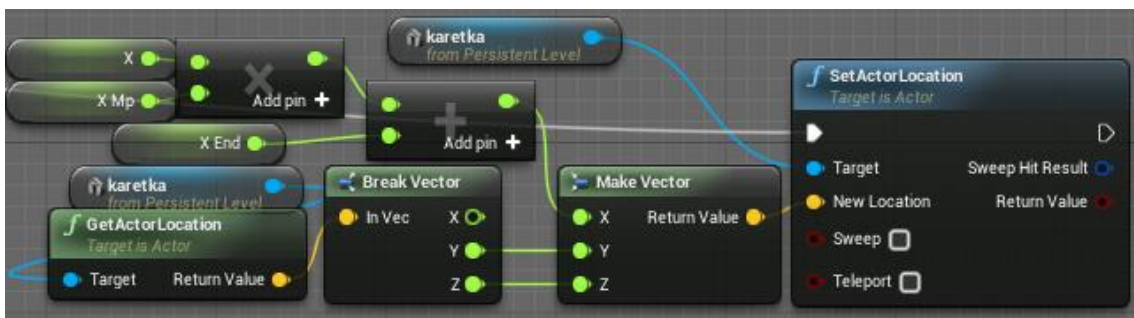


Рисунок 20. Логика перемещения каретки.

4.2 Анимация поворота спицы

Логика поворота спицы является самой простой в модели обратного маятника. Зависит от одного входного параметра – угол поворота.

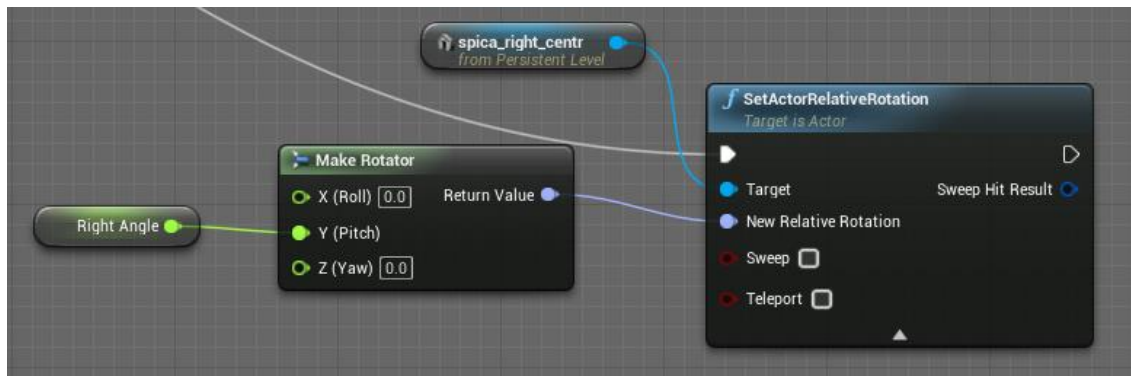


Рисунок 21. Анимационная логика поворота спицы.

В зависимости от входного параметра, спица поворачивается относительно центра объекта, к которому она прикреплена.

4.3 Направленное движение спицы

Движение каретки и поворот спицы вполне достаточно для простых математических моделей, для более сложных, где спица не закреплена и имеет свободное направленное движение, нужно писать дополнительную логику. Входными параметрами служат: длина(spica_length) спицы, позиция(spica_pos) спицы относительно этой длины (например, длина спицы 24, позиция может колебаться от 0 до 24). Все координаты объектов берутся относительно мировой точки.

Входной параметр длины спицы, может отличаться, от длины трехмерной модели, поэтому для начала нужно вычислить коэффициент относительности

$$K_{mp} = \frac{|\vec{A}_{bot} - \vec{A}_{top}|}{A_{length}}$$

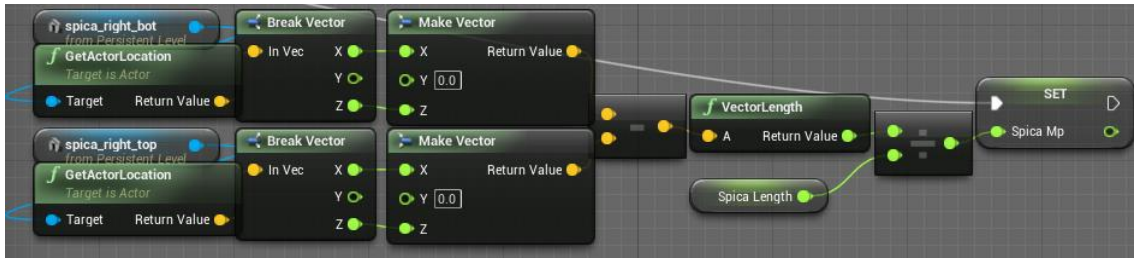


Рисунок 22. Логика относительного направленного перемещения каретки.

Для смещения спицы на новую позицию требуется знать ее текущее местоположение (P_{curr}), относительно которого будет происходить смещение.

$$P_{curr} = |\overrightarrow{A_{bot}} - \overrightarrow{A_{krest}}|$$

Следующим этапом находим разницу текущей позиции (P_{curr}) от новой (P_{pos}), относительно реальной модели.

$$P_{\Delta} = P_{curr} - P_{pos}K_{mp}$$

Новые координаты спицы находятся за счет суммы старой координаты спицы ($\overrightarrow{A_{old}}$) и произведения одиночного вектора по направлению ($\overrightarrow{A_1}$) с разницей текущей позиции (P_{Δ}).

$$\overrightarrow{A_{new}} = P_{\Delta}\overrightarrow{A_1} + \overrightarrow{A_{old}}$$

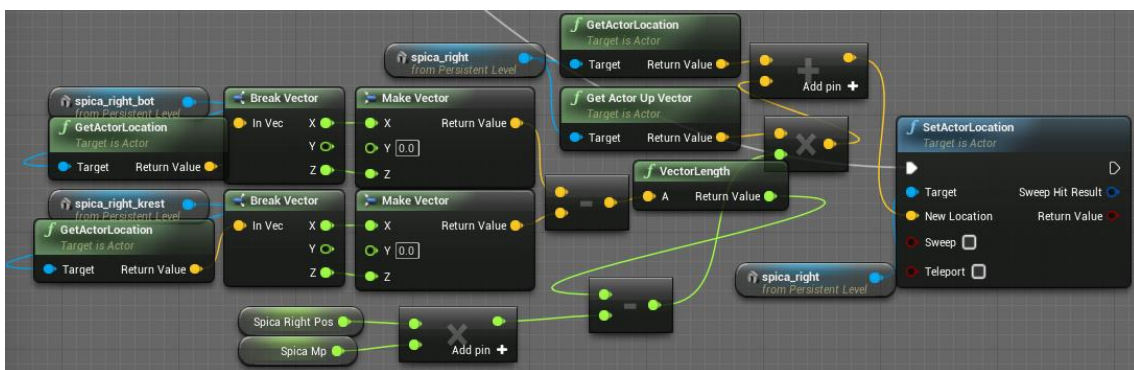


Рисунок 23. Логика направленного перемещения каретки.

5 Реализация модели при помощи программы MATLAB

MATLAB – пакет прикладных программ для решения задач технических вычислений, работает на большинстве операционных систем. Представляет собой основу всего семейства продуктов MathWorks. Ядро MATLAB позволяет максимально просто работать с матрицами разных типов. Предоставляет множество методов для анализа данных, разработки алгоритмов и создания моделей [7].

В данной работе MATLAB будет использоваться исключительно для передачи координат анимированной виртуальной модели. Будет рассмотрена простая математическая модель, где спицы фиксированы, изменяется только местоположение каретки и угол поворота спицы. Усложняя математическую модель и увеличивая выходные данные, можно визуализировать так и сложные математические модели.

Данные будут передаваться по UDP, для этого в UE4 существует отдельный ‘actor’, способный принимать пакеты и парсить их в структуру переданных данных, в нашем случае координат перемещения каретки и поворота спицы.



Рисунок 24. Логотип пакета прикладных программ MATLAB.

5.1 Математическая модель обратного маятника

Данная работа не подразумевает под собой создание математической модели обратного маятника, поэтому будет взята уже существующая, способная передавать нужные нам структуры данных по UDP протоколу. Существует возможность усложнения математической модели для более реальной визуализации. Визуальная модель обратного маятника способна визуализировать относительные передвижения.

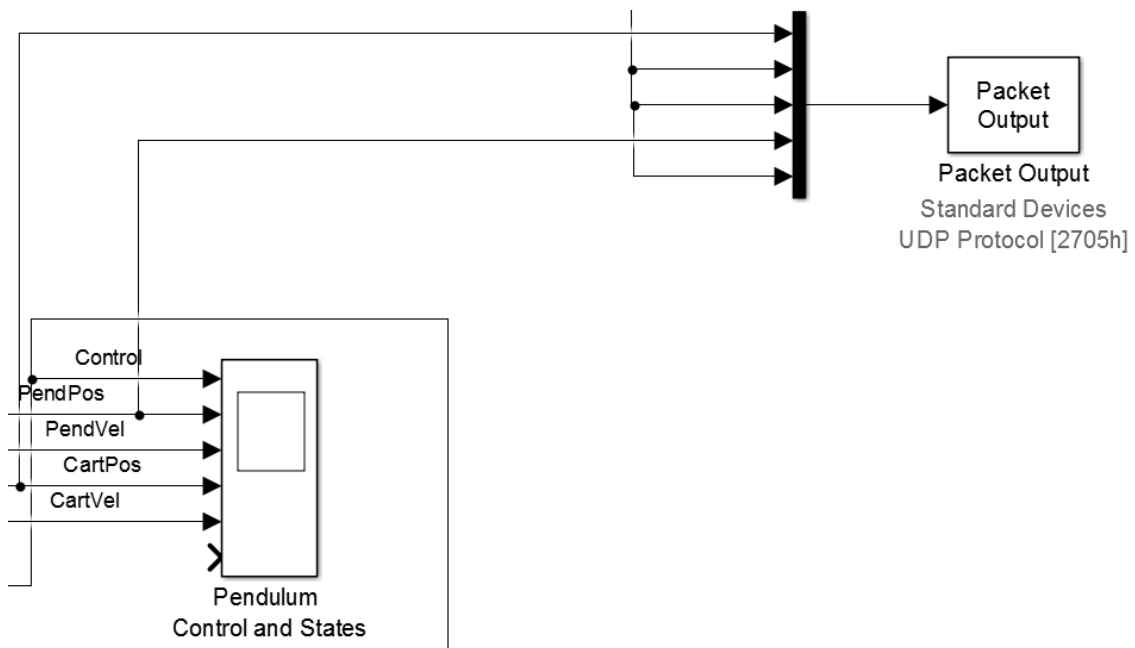


Рисунок 25. Передатчик структуры данных в среде Simulink.

5.2 Визуальная модель обратного маятника

Трехмерная модель и ее анимация уже были рассмотрены ранее. Добавив соответствующий event в blueprint, который имеет UDP приемник, написанный на C++ языке программирования, нам остается только сопоставить выходные параметры с входными. Во время запуска имитации математической модели, данные будут передаваться напрямую в этот event, что даст доступ к координатам каретки и углу поворота спицы. Анимация и запущенная математическая модель могут находиться на разных машинах. Правильность анимации можно проверить, сопоставив их с графиками, которые выдает MATLAB. В итоге получится

анимация виртуальной модели обратного маятника на основе реальной математической модели.

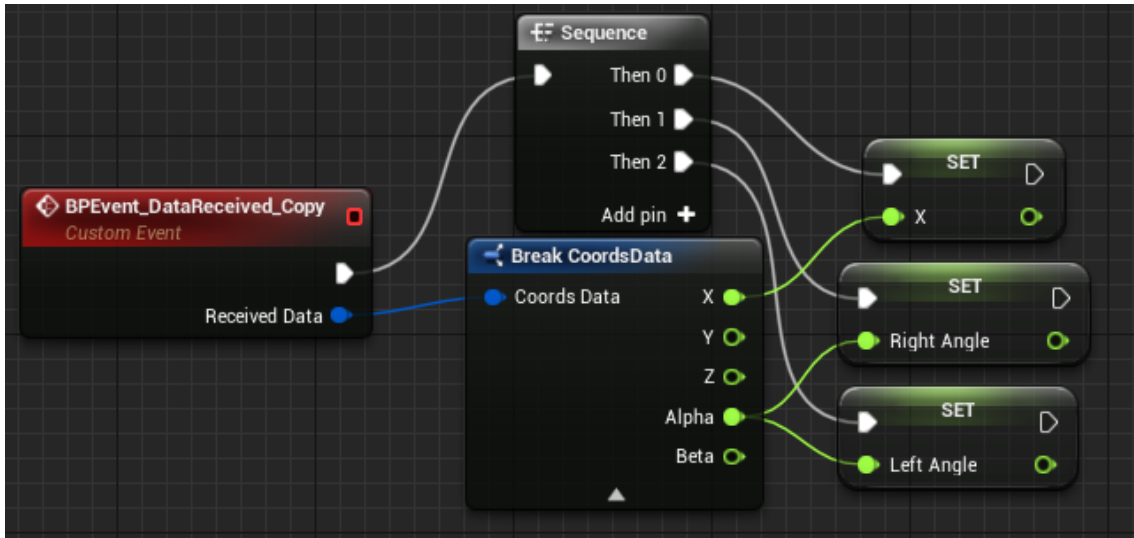


Рисунок 26. Приемник структуры данных в среде Unreal Engine 4.

6 Вывод

Проектирование трехмерной модели обратного маятника происходило в системе автоматизированного проектирования AutoCAD. Данная программа имеет большой набор инструментов как для трехмерного, так и для двухмерного проектирования. За счет встроенной обширной документации, доступной на разных языках, знакомство с новыми инструментами было простым. Сложные детали были построены за счет совокупности более простых. Данная система позволила максимально точно передать детали реальной модели, что повлияло на реалистичность виртуальной модели.

Конвертировать трехмерную модель на игровой движок Unreal Engine 4 напрямую не удалось. Был рассмотрен вариант конвертации посредством системы создания и редактирования трехмерной графики 3Ds Max. Проблема сброса pivot точки была решена путем использования стороннего скрипта TS_Tools. В итоге все pivot точки стали централизованы относительно центра соответствующего объекта.

Анимация модели была создана на игровом движке Unreal Engine 4. Логика анимации написана при помощи визуальной системы программирования, через blueprint. Движение объектов модели соответствует реальному, за счет коэффициента относительности между виртуальной и реальной модели.

Входные параметры модели были вычислены за счет математической модели обратного маятника в программе MATLAB. Данная программа используется только для вычисления математической модели и передачи координат передвижения каретки и угла поворота по UDP протоколу.

Kokkuvõte

Pöörpendli kolmemõõtmelise mudeli projekteerimine toimus automatiseeritud projekteerimise programmis AutoCAD. Antud programmis on olemas suur instrumentide valik kolmemõõtmelise kui ka kahemõõtmelise projekteerimiseks. Dokumentatsioon on lai ja mahukas ning on antud erinevates keeltes. Selle abil tutvumine uue instrumentidega oli kerge. Keerulised detallid olid konstrueeritud kergemate detallide eest. Antud süsteem andis võimalust maksimaalselt täpselt konstrueerida detallide mudelit. Sellepärast virtuaalne mudel sai realistlikuks.

Otse kolmemõõtmelise mudeli konverteerimine ebaõnnestus mänguliugurile Unreal Engine 4. Oli tehtud konverteerimise katse kolmemõõtmelise graafiku programmis 3Ds Max. Pivot punkti taastamise probleem oli lahendatud kõrvalise TS_Tools skripti kasutamise abil. Tulemuses kõik pivot punktid said tsentraliseeritud antud objekti keskpunkti suhtes.

Mudeli animatsioon oli tehtud mänguliuguril Unreal Engine 4. Animatsiooni loogika oli kirjutatud visuaalse programmeerimise süsteemi abil blueprint kasutamises. Objekti mudeli liikumine vastab reaalse mudelile. See toimus virtuaalse ja reaalse mudelite vahel suhtelisuse teguri abil.

Mudeli sisendparameetrid olid arvutatud pöörpendli matemaatilise mudeli abil MATLAB programmis. Antud programmi kasutatakse matemaatilise mudeli arvutamiseks, vankri liikumise koordinaadi edastamiseks ja nurga pöördamiseks UDP protokollil peal.

Summary

The design engineering of the inverted pendulum three-dimensional model took place in computer-aided design system AutoCAD. This program has a great selection of tools for three-dimensional as well as two-dimensional design engineering. Due to the extensive built-in documentation available in different languages, familiarity with the new tools was simple. Complex details were built by a combination of simpler ones. Current system allowed to pass on the details of real model as accurate as possible, which had an impact on the realism of the virtual model.

Conversion of a three-dimensional model directly to the game engine Unreal Engine 4 was not successful. There was considered a conversion option by the usage of creating and editing the three-dimensional graphics system 3Ds Max. The problem of resetting the pivot points was resolved by the usage of a third-party script TS_Tools. As a result, all the pivot points became centralized with respect to the center of the corresponding object.

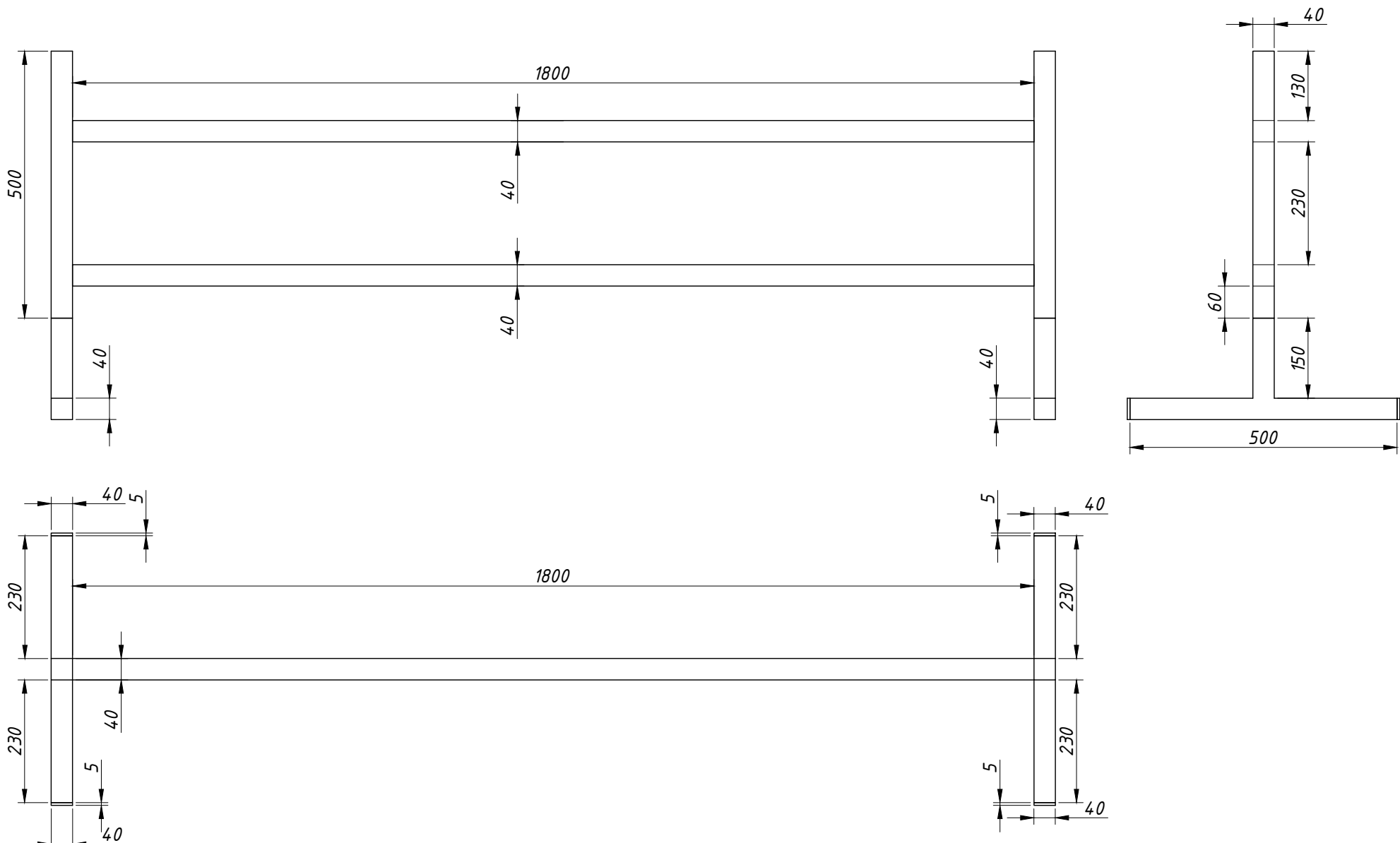
Model animation was implemented on the game engine Unreal Engine 4. Animation logic was written using visual programming system through the blueprint. The objects' movement patterns correspond to the real ones due to the coefficient of relativity between vertical and real models.

The input parameters of the model were calculated by applying a mathematical model of inverted pendulum in MATLAB. This program is only used to calculate the mathematical model and the motion coordinates of the carriage and the rotation angle transmission over UDP.

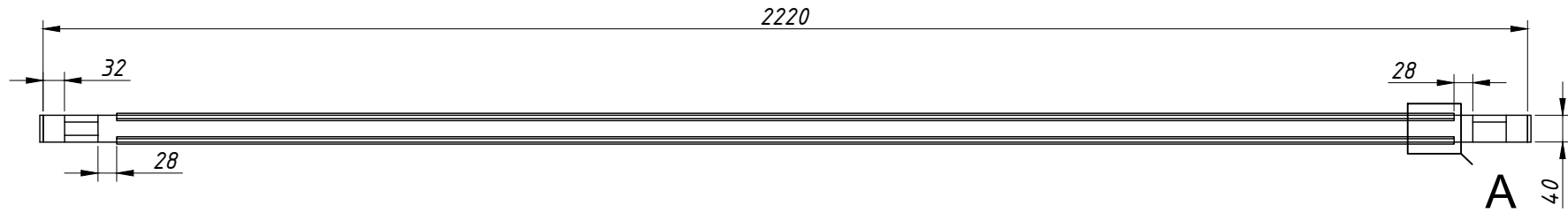
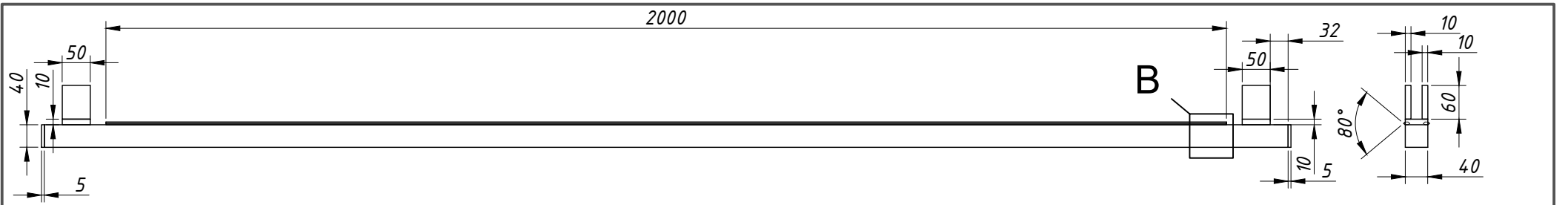
Источники

- [1] "AutoCAD," [Online]. Available: <https://ru.wikipedia.org>. [Accessed 30 May 2016].
- [2] "DWG," [Online]. Available: <https://ru.wikipedia.org>. [Accessed 30 May 2016].
- [3] "FBX," [Online]. Available: <https://ru.wikipedia.org>. [Accessed 30 May 2016].
- [4] "EXPORT (Command)," [Online]. Available: <https://knowledge.autodesk.com>. [Accessed 30 May 2016].
- [5] "Autodesk 3ds Max," [Online]. Available: <https://ru.wikipedia.org>. [Accessed 30 May 2016].
- [6] "Unreal Engine," [Online]. Available: <https://ru.wikipedia.org>. [Accessed 30 May 2016].
- [7] "MATLAB," [Online]. Available: <https://ru.wikipedia.org>. [Accessed 30 May 2016].

**Дополнение – Чертежи модели обратного маятника,
спроектированные в системе AutoCAD**

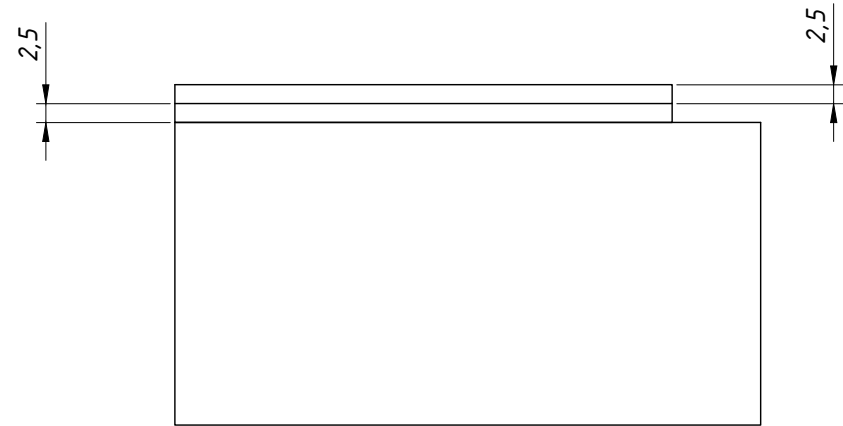
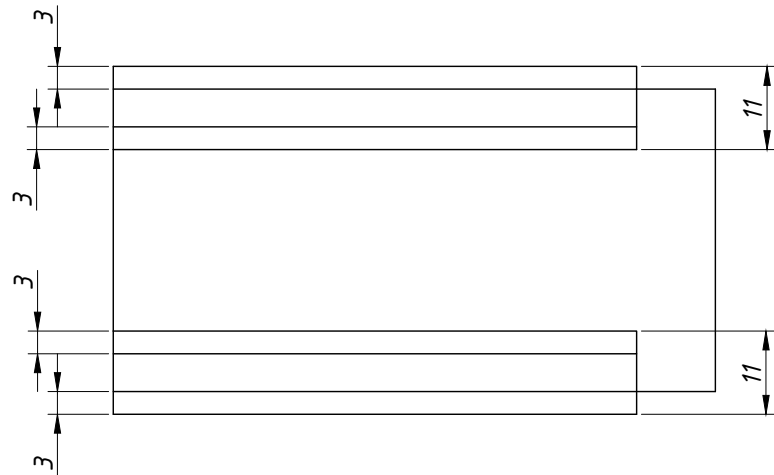


<i>AutoCad model</i>			<i>TTU</i>
<i>12.05.16.</i>	<i>Aleksandr Kuzmin</i>	<i>1:1</i>	<i>IASB</i>
<i>Cart vertical plane legs</i>			

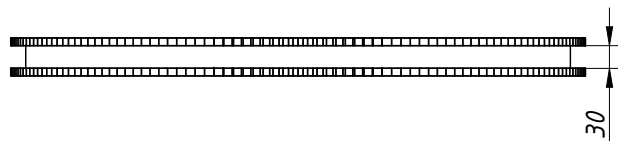
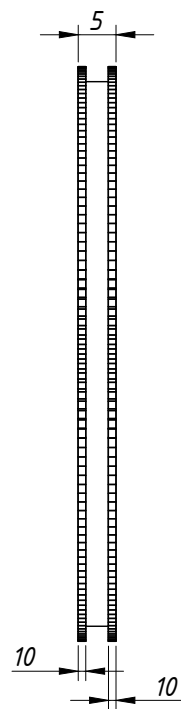
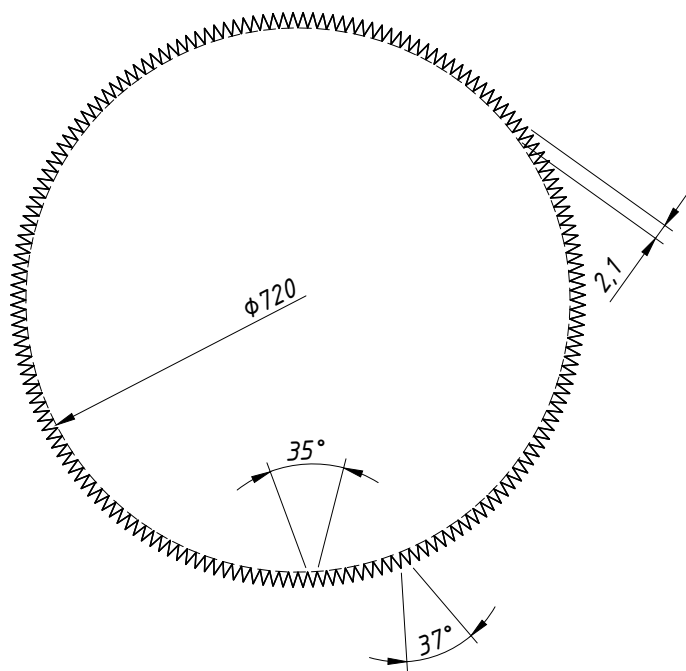


A (10:1)

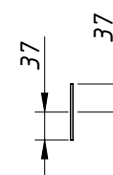
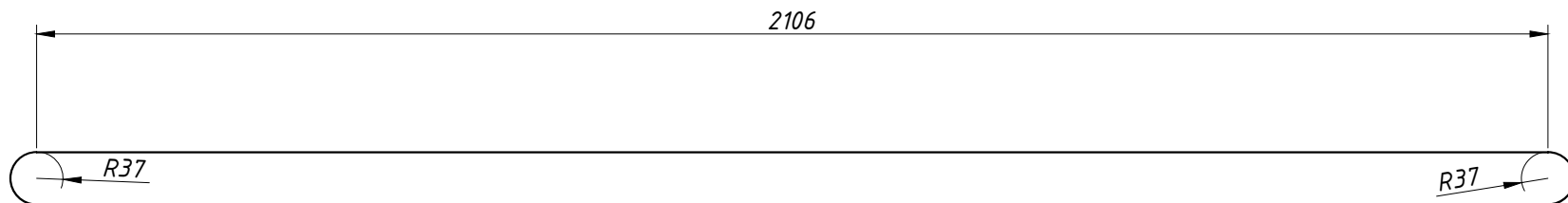
B (10:1)



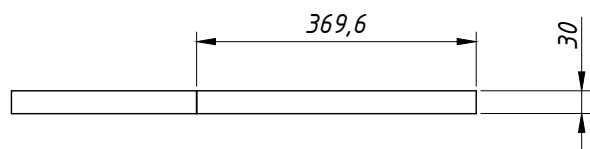
<i>AutoCad model</i>			<i>TTU</i>
<i>12.05.16.</i>	<i>Aleksandr Kuzmin</i>	<i>1:1</i>	<i>IASB</i>
<i>Cart vertical plane with rails</i>			



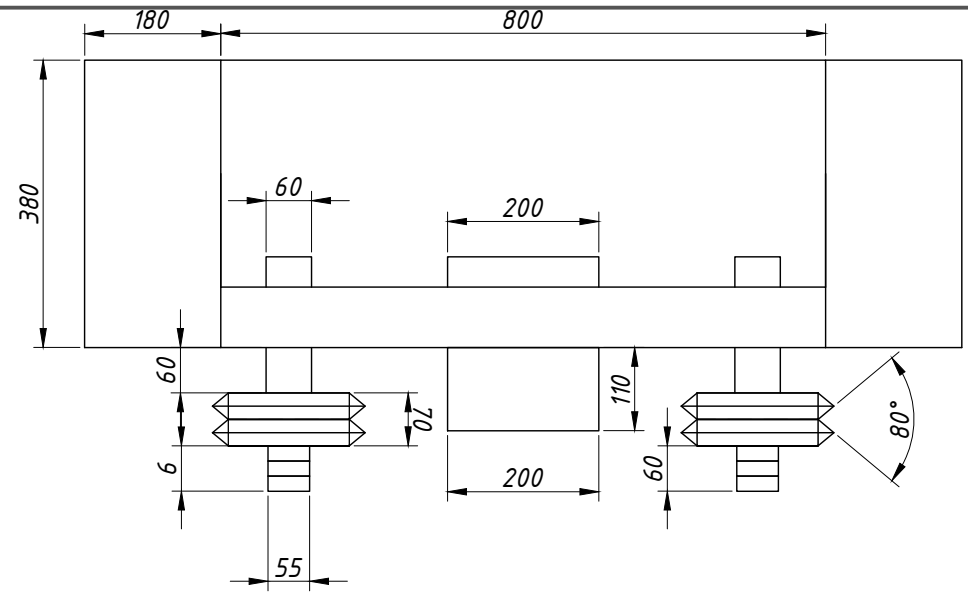
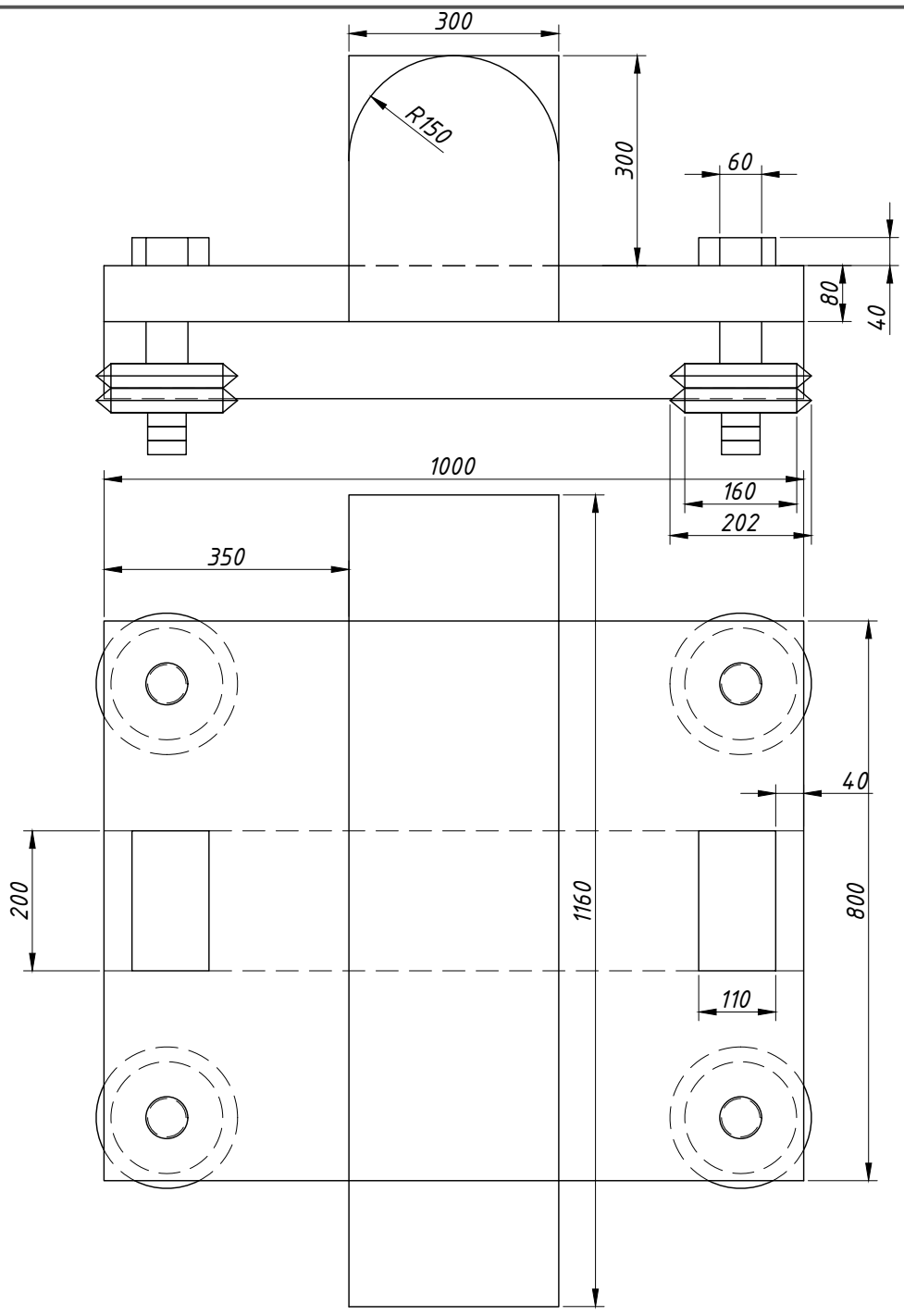
<i>AutoCad model</i>			<i>TTU</i>
<i>12.05.16.</i>	<i>Aleksandr Kuzmin</i>	<i>10:1</i>	<i>IASB</i>
<i>Star for wire</i>			



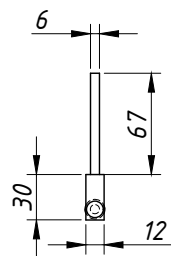
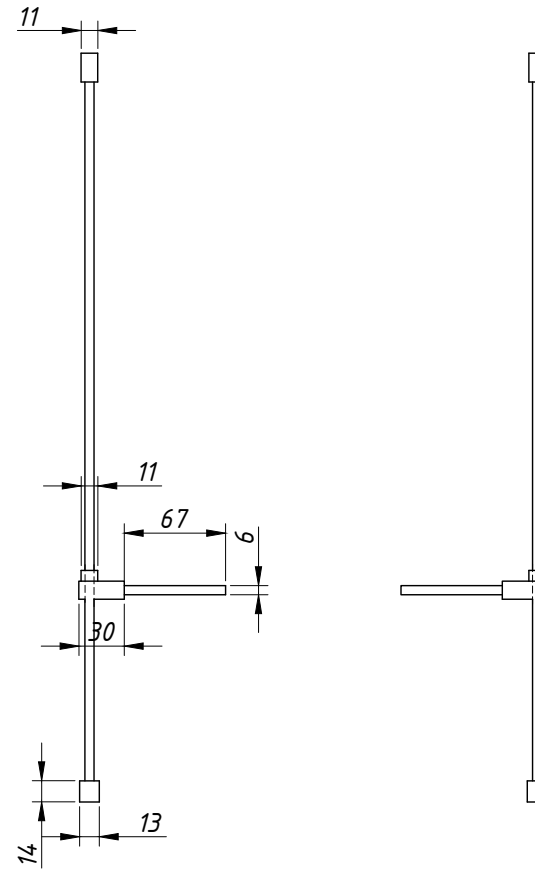
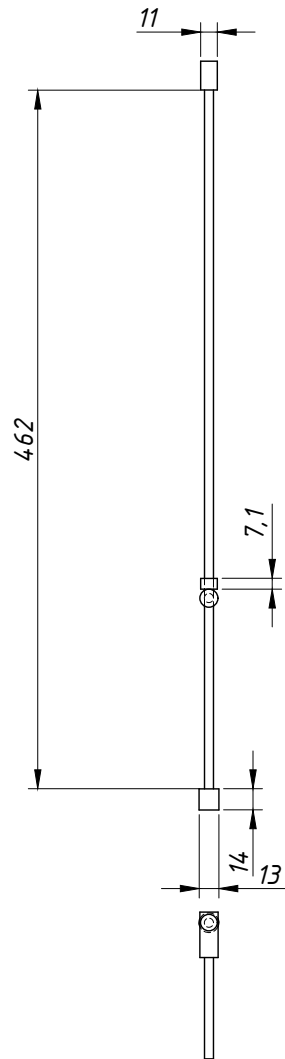
C
(10:1)



<i>AutoCad model</i>			<i>TTU</i>
<i>12.05.16.</i>	<i>Aleksandr Kuzmin</i>	<i>1:1</i>	<i>IASB</i>
<i>Wire</i>			



<i>AutoCad model</i>			<i>TTU</i>
<i>12.05.16.</i>	<i>Aleksandr Kuzmin</i>	<i>10:1</i>	<i>IASB</i>
<i>Cart</i>			



<i>AutoCad model</i>			<i>TTU</i>
<i>12.05.16.</i>	<i>Aleksandr Kuzmin</i>	<i>2:1</i>	<i>IASB</i>
<i>Pendulum</i>			