



# UM Control: связь с Matlab/Simulink, SimInTech, и по UDP протоколу

Это руководство поможет вам изучить как особенности интеграции моделей, созданных в среде Matlab/Simulink и SimInTech, в модели программного комплекса «Универсальный механизм», так и наоборот, интеграцию моделей программного комплекса «Универсальный механизм» в модели, созданные в среде Matlab/Simulink и SimInTech

## Оглавление

<b>НАЧИНАЕМ РАБОТАТЬ В ПРОГРАММНОМ КОМПЛЕКСЕ «УНИВЕРСАЛЬНЫЙ МЕХАНИЗМ»: ИНТЕРФЕЙС С MATLAB/SIMULINK, SIMINTECH И ПО UDP ПРОТОКОЛУ .....</b>	<b>4</b>
<b>1. ВВЕДЕНИЕ .....</b>	<b>9</b>
<b>2. ЗНАКОМСТВО С МОДЕЛЯМИ.....</b>	<b>11</b>
<b>2.1. ПЕРЕВЕРНУТЫЙ МАЯТНИК.....</b>	<b>11</b>
2.1.1. Описание модели .....	11
2.1.2. Подготовка модели на этапе создания .....	13
<b>2.2. ДВИГАТЕЛЬ ПОСТОЯННОГО ТОКА .....</b>	<b>14</b>
2.2.1. Описание модели .....	14
2.2.2. Особенности описания механической части .....	16
2.2.3. Особенности модели двигателя .....	17
<b>2.3. ОСОБЕННОСТИ И ОГРАНИЧЕНИЯ.....</b>	<b>18</b>
<b>3. РАБОТА С MATLAB/SIMULINK .....</b>	<b>19</b>
<b>3.1. МОДЕЛИРОВАНИЕ С ИСПОЛЬЗОВАНИЕМ MATLAB IMPORT .....</b>	<b>19</b>
3.1.1. Порядок работы.....	19
3.1.2. Перевернутый маятник.....	20
3.1.2.1. Экспорт системы управления из Matlab/Simulink.....	21
3.1.2.2. Подключение библиотеки Matlab/Simulink в UM.....	30
3.1.2.3. Моделирование движения.....	37
3.1.3. Двигатель постоянного тока .....	40
3.1.3.1. Описание системы управления.....	41
3.1.3.2. Экспорт двигателя из Matlab/Simulink.....	42
3.1.3.3. Подключение библиотеки Matlab/Simulink в UM.....	43
3.1.3.4. Моделирование движения.....	48
<b>3.2. МОДЕЛИРОВАНИЕ С ИСПОЛЬЗОВАНИЕМ MATLAB COSIMULATION.....</b>	<b>51</b>
3.2.1. Порядок работы.....	52
3.2.2. Перевернутый маятник.....	53
3.2.2.1. Подготовка системы управления в Matlab/Simulink.....	54
3.2.2.2. Экспорт модели механической системы из UM .....	55
3.2.2.3. Создание связи между UM-моделью и Matlab/Simulink .....	62
3.2.2.4. Моделирование движения.....	63
3.2.3. Двигатель постоянного тока .....	65
3.2.3.1. Подготовка системы управления в Matlab/Simulink.....	66
3.2.3.2. Экспорт модели механической системы из UM .....	67
3.2.3.3. Создание связи между UM-моделью и Matlab/Simulink .....	70
3.2.3.4. Моделирование движения.....	71
3.2.4. Антиблокировочная система автомобиля (ABS) .....	72
3.2.4.1. Подготовка системы управления в Matlab/Simulink.....	73
3.2.4.2. Экспорт модели механической системы из UM .....	73
3.2.4.3. Моделирование движения.....	75
3.2.4.3.1. Загрузка модели .....	75
3.2.4.3.2. Результаты моделирования движения с ABS.....	76
3.2.4.3.3. Результаты моделирования движения без ABS .....	79
<b>4. РАБОТА С SIMINTECH.....</b>	<b>82</b>
<b>4.1. О SIMINTECH.....</b>	<b>82</b>

<b>4.2. МОДЕЛИРОВАНИЕ С ИСПОЛЬЗОВАНИЕМ SIMINTECH IMPORT .....</b>	<b>82</b>
4.2.1. Порядок работы.....	82
4.2.2. Перевернутый маятник.....	84
4.2.2.1. Экспорт системы управления из SimInTech.....	85
4.2.2.2. Подключение библиотеки SimInTech в UM.....	90
4.2.2.3. Моделирование движения.....	96
<b>4.3. МОДЕЛИРОВАНИЕ С ИСПОЛЬЗОВАНИЕМ SIMINTECH CoSIMULATION.....</b>	<b>99</b>
4.3.1. Порядок работы.....	100
4.3.2. Перевернутый маятник.....	101
4.3.2.1. Подготовка системы управления в SimInTech .....	102
4.3.2.2. Экспорт модели механической системы из UM .....	103
4.3.2.3. Создание связи между UM-моделью и SimInTech .....	109
4.3.2.4. Моделирование движения.....	110
<b>4.4. СВЯЗЬ С SIMINTECH ПО UDP ПРОТОКОЛУ .....</b>	<b>114</b>
4.4.1. Подключение расчёта силы в демпфере .....	114
4.4.2. Перевернутый маятник.....	119

## Начинаем работать в программном комплексе «Универсальный механизм»: интерфейс с Matlab/Simulink, SimInTech и по UDP протоколу

Это руководство поможет вам изучить как особенности интеграции моделей, созданных в среде Matlab/Simulink и SimInTech, в модели программного комплекса (ПК) «Универсальный механизм», так и наоборот, интеграцию моделей программного комплекса «Универсальный механизм» в модели, созданные в среде Matlab/Simulink и SimInTech. Также рассмотрен инструмент **UDP Bridge** из модуля **UM Control**, который позволяет связывать модели, созданные в «Универсальном механизме», с другими программами по сетевому протоколу UDP.

Предполагается, что вы уже изучили раздел, посвященный введению в моделирование в ПК УМ, который находится в файле **gs\_UM.pdf**<sup>1</sup>, и умеете выполнять в УМ простые действия: создавать новую модель, добавлять тела и шарниры (**UM Input**) и работать в программе моделирования движения (**UM Simulation**).

В настоящем руководстве мы шаг за шагом рассмотрим два примера. Первый пример продемонстрирует моделирование перевернутого маятника с системой управления, которая удерживает его в вертикальном перевернутом положении. Механическая часть системы – это обычная модель ПК «Универсальный механизм», модель системы управления создается в среде Matlab/Simulink или SimInTech. Затем в зависимости от выбранного подхода либо (1) модель Matlab/Simulink или SimInTech импортируется в ПК «Универсальный механизм» и связывается с моделью механической части с помощью инструментов **Matlab Import** или **SimInTech Import**, входящих в модуль **UM Control**, либо (2) модель механической части импортируется и включается в модель системы управления на стороне Matlab/Simulink или SimInTech с помощью инструментов **Matlab CoSimulation** или **SimInTech CoSimulation**, входящих в тот же модуль. Оба способа позволяют проанализировать динамику управляемого движения.

Далее мы рассмотрим пример (только для Matlab/Simulink), который иллюстрирует подключение двигателя постоянного тока к валу. Модель двигателя создана в Matlab/Simulink. Этот пример иллюстрирует разгон вала двигателем до номинальной угловой скорости с последующим приложением нагрузки. В конце этого примера мы построим графики силы тока, напряжения и электромагнитного момента на двигателе, а также график угловой скорости вала.

### Совместимость с Matlab/Simulink

#### UM Control / Matlab Import

Импорт моделей Matlab/Simulink в УМ доступен только в поставке с инструментом **UM Control / Matlab Import**. Для проверки наличия этого инструмента запустите программу **UM Input** или **UM Simulation**, выберите пункт меню **Помощь | О программе...** В

---

<sup>1</sup> [www.universalmechanism.com/download/90/rus/g\\_s\\_um.pdf](http://www.universalmechanism.com/download/90/rus/g_s_um.pdf)

появившемся окне в разделе **Конфигурация** вы увидите список доступных модулей и инструментов.

В настоящей версии программы поддерживается интерфейс с Matlab 7.12-13 (релизы R2011), Matlab 8.X (релизы R2012-R2015) и Matlab 9.X / R2016-R2021.

Для компиляции моделей Simulink необходимо чтобы в Matlab обязательно были установлены два следующих компонента: **Matlab Coder** и **Simulink Coder**. Наличие этих компонентов в установленной версии Matlab вы можете проверить с помощью команды **ver**, запускаемой из командного окна Matlab.

Для компиляции моделей Simulink необходимо наличие компилятора Microsoft Visual C++ подходящей версии. Никакие другие компиляторы, включая MinGW и Intel Parallel Studio, не поддерживаются. Для [актуального](#) и [предыдущих](#) релизов Matlab вы можете посмотреть версии поддерживаемых компиляторов на соответствующим web-ссылкам. Обратите внимание, что и более поздние и более ранние версии компилятора не подходят. Нужно установить строго одну из версий, перечисленных в списке поддерживаемых компиляторов. Щелкнув по одной из ссылок выше перейдите в раздел поддерживаемых компиляторов, см. рис. ниже. Далее найдите поддерживаемые компиляторы в столбце **MATLAB Coder**.

Previous Releases: System Requirements and Supported Compilers

Release	Windows	Linux	Mac	Solaris/UNIX	Supported Compilers	Platform Availability
R2021a (MATLAB 9.10)	Details	Details	Details	N/A	Details	Details
R2020b (MATLAB 9.9)	Details	Details	Details	N/A	Details	Details
R2020a (MATLAB 9.8)	Details	Details	Details	N/A	Details	Details
R2019b (MATLAB 9.7)	Details	Details	Details	N/A	Details	Details

MATLAB Product Family – Release 2021a									
Compiler	MATLAB	MATLAB Coder	GPU Coder	SimBiology	Fixed-Point Designer	HDL Coder	HDL Verifier	Audio Toolbox	ROS Toolbox
	For MEX-file compilation, loadlibrary, C++ interface, and external usage of MATLAB Engine and MAT-file APIs	For all features	For all features	For accelerated computation	For accelerated computation	For accelerated testbench simulation	For DPI and TLM component generation	For validating and generating audio plugins	For custom messages and code generation
MinGW 6.3 C/C++ (Distributor: mingw-w64) <a href="#">Download Now</a> Available at no charge	✓	✓		✓	✓	✓	✓		
Microsoft Visual C++ 2019 product family	✓	✓	✓	✓	✓			✓	
Microsoft Visual C++ 2017 product family <sup>10</sup>	✓	✓	✓	✓	✓	✓	✓	✓	✓
Microsoft Visual C++ 2015 Professional <sup>8</sup>	✓	✓	✓	✓	✓	✓	✓	✓	
Intel Parallel Studio XE 2020 for C/C++ <sup>5</sup>	✓	✓		✓	✓				
Intel Parallel Studio XE 2019 for C/C++ <sup>5</sup>	✓	✓		✓	✓				
Intel Parallel Studio XE 2018 for C/C++ <sup>5</sup>	✓	✓		✓	✓				
Intel Parallel Studio XE 2020 for Fortran <sup>5</sup>	✓			✓					
Intel Parallel Studio XE 2019 for Fortran <sup>5</sup>	✓			✓					
Intel Parallel Studio XE 2018 for Fortran <sup>5</sup>	✓			✓					

Модели, описанные в Matlab/Simulink для успешного импорта в UM должны содержать только непрерывные элементы, не допускается использование в импортируемой модели дискретных элементов.

Начиная с версии Matlab 8.X подготовленные DLL требуют наличия на компьютере Matlab/Simulink и неработоспособны при их использовании на компьютере без установленного Matlab/Simulink.

**UM Control / Matlab CoSimulation**

Экспорт моделей UM в Matlab/Simulink доступен только в поставке с инструментом **UM Control / Matlab CoSimulation**. Для поддержки расчета динамики UM-модели из Matlab/Simulink необходимо, чтобы на компьютере было установлено динамическое ядро UM в виде COM-сервера (модуль **UM COM Server**). Это делается автоматически при установке Универсального механизма.

В настоящей версии программы поддерживается интерфейс с Matlab 7.X и 8.X (релизы R14, R2006 и более поздние).

**Внимание!** ПК "Универсальный механизм" начиная с версии 9 является 64-разрядным приложением. Для совместной работы с помощью модуля **UM Control / Matlab CoSimulation** вам понадобится 64-разрядная версия Matlab.

## Совместимость с SimInTech

### UM Control / SimInTech Import

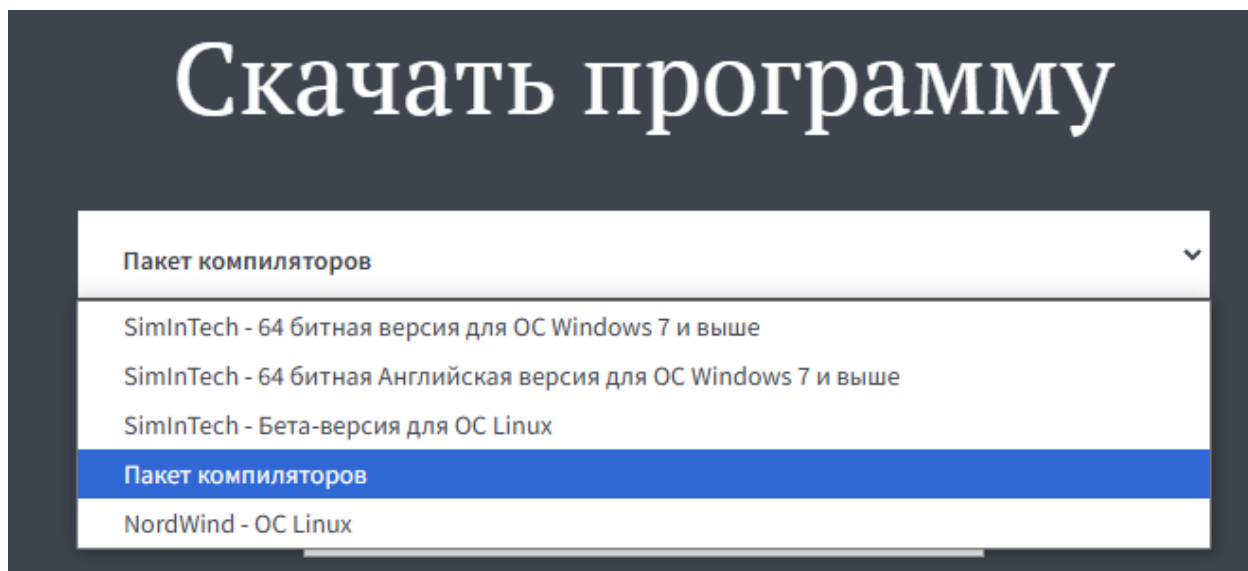
Импорт расчетных схем SimInTech в UM доступен только в поставке с инструментом **UM Control / SimInTech Import**. Для проверки наличия этого инструмента запустите программу **UM Input** или **UM Simulation**, выберите пункт меню **Помощь | О программе...** В появившемся окне в разделе **Конфигурация** вы увидите список доступных модулей и инструментов.

Для успешной работы динамически загружаемой библиотеки, созданной в SimInTech, в Универсальном механизме, требуется, чтобы оба программных продукта – и Универсальный механизм, и SimInTech – были 64-разрядными приложениями для Windows.

Для генерации кода на основе разработанной модели SimInTech требуется лицензия на инструмент **Кодогенератор Си**. Проверьте наличие такой лицензии с помощью пункта главного меню SimInTech **Справка/О программе** либо запросите её у поставщика.



После генерации исходного кода на Си для его компиляции в динамически загружаемую библиотеку DLL требуется скачать и установить **Пакет компиляторов**, который доступен на сайте <https://simintech.ru> в разделе **Скачать дистрибутив (Скачать программу)**.



### UM Control / SimInTech CoSimulation

Экспорт моделей UM в SimInTech доступен только в поставке с инструментом **UM Control / SimInTech CoSimulation**. Для поддержки расчета динамики UM-модели из SimInTech необходимо, чтобы на компьютере было установлено динамическое ядро UM в виде COM-сервера (модуль **UM COM Server**). Это делается автоматически при установке Универсального механизма.

Работа UM-моделей в среде SimInTech поддерживается специальным блоком «Связь с расчетным кодом Универсальный механизм», расположенным на вкладке «Данные». Этот блок доступен для пользователей SimInTech начиная с версии 1.17.9.29.

**Внимание!** Инструмент **UM Control / SimInTech CoSimulation** работает только для 64-разрядных версий **SimInTech** и Универсального механизма, 32-разрядные версии не поддерживаются.

### Ответственность и авторские права

Данное руководство может изменяться время от времени. Авторы не несут никакой ответственности за любые ошибки и несоответствия, которые могут иметь место в данном документе.

ООО «Вычислительная механика». Все права защищены ©, 2026.

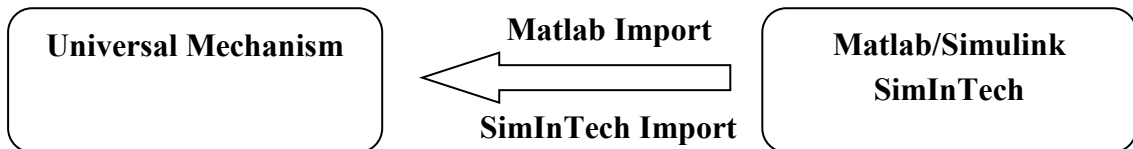
Все товарные знаки принадлежат их законным владельцам.

# 1. Введение

В данном руководстве будут рассмотрены два подхода, позволяющие связать механическую систему, созданную в ПК «Универсальный механизм», с расчетными схемами, построенными в среде Matlab/Simulink или SimInTech. Остановимся подробнее на особенностях каждого из предлагаемых вариантов моделирования.

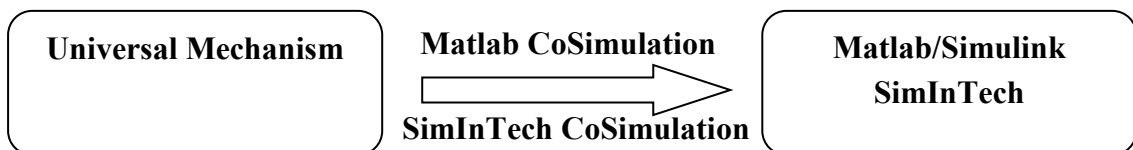
## Инструменты Matlab Import и SimInTech Import

Этот подход предполагает, что расчетная схема, созданная в среде Matlab/Simulink или SimInTech, сначала компилируется в динамическую библиотеку (DLL), а затем уже эта динамически загружаемая библиотека средствами ПК «Универсальный механизм» подключается к модели механической части. Процедура подключения динамической библиотеки производится с помощью **Мастера связи с внешними библиотеками** в ПК «Универсальный механизм». Собственно моделирование динамики связанной системы происходит в среде ПК «Универсальный механизм».



## Инструменты Matlab CoSimulation и SimInTech CoSimulation

Другой вариант моделирования предполагает, что в расчетную схему Matlab/Simulink или SimInTech включается отдельный блок (для Matlab это S-функция, а для SimInTech это специальный блок «Универсальный механизм»), который представляет собой модель механической системы ПК «Универсальный механизм». **Мастер экспорта** позволяет описать входы и выходы блока UM для Matlab/Simulink или SimInTech и формирует все необходимые вспомогательные файлы для работы UM-модели. Моделирование динамики связанной системы происходит на стороне Matlab/Simulink или SimInTech.



На рисунках выше показаны пути экспорта/импорта моделей и приложение, в котором будет выполняться моделирование составной модели, включающей UM-модель и модель Matlab/Simulink или SimInTech. Поток данных в процессе численного моделирования уравнений движения в любом случае идет двунаправленно.

## Инструмент UDP Bridge и обмен данными с Matlab/Simulink и SimInTech по протоколу UDP

Инструмент **UDP Bridge** из модуля **UM Control** позволяет связывать модели, созданные в «Универсальном механизме», с другими программами по сетевому протоколу UDP.

Этот инструмент позволяет отправлять результаты моделирования в "Универсальном механизме" на удалённый компьютер, а также принимать сигналы с удалённого компьютера. Из "Универсального механизма" можно отправлять любую переменную, созданную в **Мастере переменных**. Обычно отправляются кинематические величины: координаты, скорости, ускорения. Принимаемые сигналы присваиваются идентификаторам модели и обычно это идентификаторы, описывающие силовое воздействие.

Подробнее инструмент UDP Bridge описан в главе [19\\_um\\_cosimulation](#), п. 19.2. "Инструмент UDP Bridge".

Ниже мы подробно рассмотрим примеры с подключением по UDP протоколу к «Универсальному механизму» расчётных схем, созданных в SimInTech и Matlab/Simulink. Модель в «Универсальном механизме» и расчётные схемы для SimInTech и Matlab/Simulink уже разработаны и находятся в примерах, поставляемых в инсталляции.

## 2. Знакомство с моделями

В этой части руководства будут рассмотрены модели перевернутого маятника и двигателя постоянного тока. Мы подробно рассмотрим физическую составляющую предложенных моделей, дадим характеристику основным параметрам моделей, остановимся на особенностях подготовки этих моделей для дальнейшей связи с расчетными схемами Matlab/Simulink или SimInTech.

### 2.1. Перевернутый маятник

#### 2.1.1. Описание модели

Модель, представленная на рис. 2.1, состоит из тележки, массой  $M$  и перевернутого маятника массой  $m$ , момент инерции которого относительно оси вращения равен  $I$ . Параметры модели даны в табл. 2.1. На вход системы управления подается угол отклонения маятника от вертикали  $\psi$ , с выхода системы управления снимается величина управляющей силы  $F$ , которую нужно приложить к тележке для балансировки перевернутого маятника.

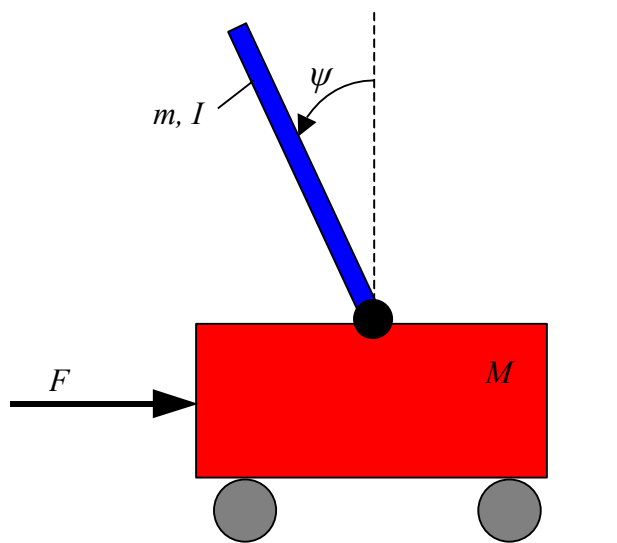


Рис. 2.1. Расчетная схема модели

Таблица 2.1

**Параметры модели**

Параметр на рисунке	Параметр модели	Комментарий	Значение
$F$	force	Сила, приложенная к тележке	
$M$	mass_cart	Масса тележки	0,5 кг
$M$	mass_pend	Масса маятника	0,2 кг
$I$	ix	Момент инерции маятника относительно оси вращения	0,006 кг·м <sup>2</sup>
	l	Длина от оси вращения до центра масс маятника	0,3 м
	b	Коэффициент вязкого трения для тележки	0,1 Н/м/с
$\psi$		Угол отклонения маятника от вертикали	рад

## 2.1.2. Подготовка модели на этапе создания

Следует отметить, что предусмотреть возможность дальнейшей связи UM-модели механической системы с системой управления необходимо уже на этапе описания модели в программе **UM Input**.

Если система управления возвращает силы или моменты, которые нужно приложить к механической системе, то на этапе описания модели такие силы или моменты нужно в ней предусмотреть.

В нашей модели перевернутого маятника в модель механической части добавлена Т-сила **Control force**, действующая вдоль оси Y. Величина силы задается параметром модели **force**. В дальнейшем мы свяжем выходную величину системы управления, смоделированной в **Matlab/Simulink** или **SimInTech**, с этим параметром, и, таким образом, приложим к системе управляющее усилие.

Имя: Control force + + -

Комментарий/Текстовый атрибут C

Тело1: Base0 Тело2: Cart

Задать в СК тела Base0

Точка приведения : Cart

-0.3

Способ задания

Выражение  Файл

Сила:

force

Момент:

T= 10 dT= 0.01

Рис. 2.2. Описание Т-силы

## 2.2. Двигатель постоянного тока

### 2.2.1. Описание модели

В качестве другого примера рассмотрим модель вала, на который действует активный момент  $M_a$  и момент сил сопротивления  $M_c$ . Активный момент действует со стороны двигателя постоянного тока. Общий вид модели приведен на рис. 2.3.

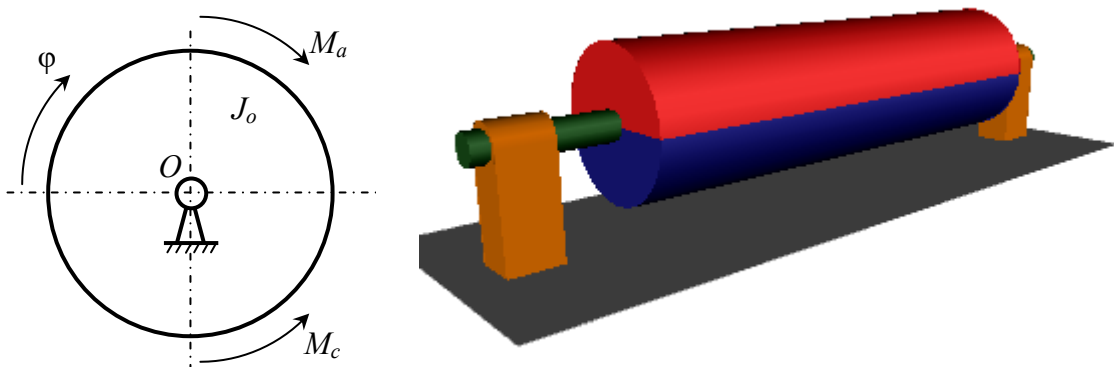


Рис. 2.3. Расчетная схема и общий вид модели

Электромеханическая модель, представленная на рис. 2.3, состоит из вала, момент инерции которого относительно оси вращения равен  $J_o$ . Вал соединен вращательным шарниром с инерциальной системой координат и, соответственно, имеет одну вращательную степень свободы.

Модель двигателя постоянного тока описана в Matlab/Simulink. На вход модели двигателя подается угловая скорость вала – предполагается, что ротор двигателя и вал жестко связаны между собой. С выхода модели двигателя снимается три величины: активный электромагнитный момент  $M_a$ , ток и напряжение цепи якоря.

Электромагнитный момент  $M_a$  описан как активный момент. Момент сил сопротивления описан как шарнирный момент. Момент сил сопротивления вводится как функция времени для иллюстрации реакции двигателя на ступенчатую нагрузку, см. рис. 2.4.

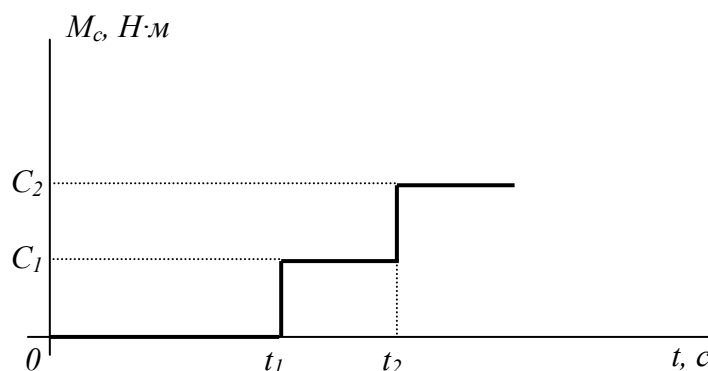


Рис. 2.4. График момента сил сопротивления (ступенчатая нагрузка)

Таблица 2.2

**Параметры модели**

Параметр на рисунке	Параметр модели	Комментарий	Значение
$Ma$	driving_torque	Активный электромагнитный момент	
$Mc$		Момент сил сопротивления	$F(t)$
$Jo$	$Iy$	Момент инерции вала относительно оси вращения	$0,1 \text{ кг}\cdot\text{м}^2$
$t_1$	t1	Момент времени $t_1$ , см. рис. 2.4	5 с
$t_2$	t2	Момент времени $t_2$ , см. рис. 2.4	8 с
$C_1$	C1	Величина момента сил сопротивления, начиная с $t=t_1$ , см. рис. 2.4	5 Н·м
$C_2$	C2	Величина момента сил сопротивления, начиная с $t=t_2$ , см. рис. 2.4	10 Н·м

### 2.2.2. Особенности описания механической части

Как уже было сказано выше, предусмотреть возможность дальнейшей связи UM-модели механической системы с системой управления необходимо еще на этапе описания модели в программе **UM Input**. Если система управления возвращает силы или моменты, которые нужно приложить к механической системе, то на этапе описания модели такие силы или моменты нужно в ней предусмотреть.

В рассматриваемой модели добавлена Т-сила **Driving torque** (движущий момент), см. рис. 2.5, слева. Эта сила имеет один ненулевой компонент – момент относительно оси Y, заданный идентификатором **Ma**, который в дальнейшем мы и свяжем с электромагнитным моментом, вычисляемым в модели Matlab/Simulink.

Момент сил сопротивления в данной модели вводится как шарнирный момент, заданный функцией времени. Значение момента задается выражением

$$M_r = c1*heavi(t-t1)+(c2-c1)*heavi(t-t2), \text{ где}$$

t1, t2, c1 и c2 – параметры модели, см. рис. 2.4, причем

$$heavi(t) = \begin{cases} 1, & t > 0 \\ 0, & t \leq 0 \end{cases}$$

Для того чтобы впоследствии при моделировании вывести величины тока и напряжения на двигателе, в модель формально введены два параметра: **I** (ток якоря) и **U** (напряжение цепи якоря).

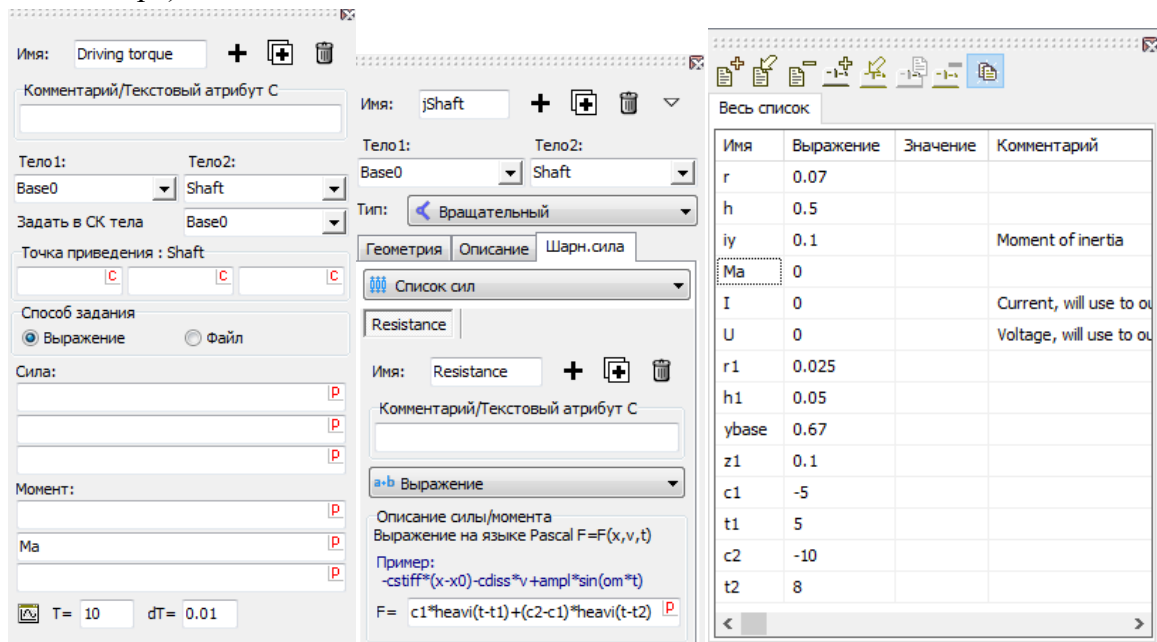


Рис. 2.5. Описание активного момента, момента сил сопротивления и параметров модели

### 2.2.3. Особенности модели двигателя

Для данного примера выберем двигатель типа 2ПН112МУХЛ4 – двигатель постоянного тока с независимым возбуждением и постоянным магнитным потоком. Справочные данные для данного типа двигателя приведены в табл. 2.3.

Таблица 2.3

Параметры двигателя		
Параметр	Значение	
Мощность, кВт	3,6	
Напряжение, В	220	
Номинальная частота вращения, об/мин	3000	
Максимальная частота вращения, об/мин	4000	
КПД, %	79	
Сопротивление обмотки при 15°C, Ом		
	якоря	0,42
	добавочных полюсов	0,356
	возбуждения	129
Индуктивность цепи якоря, мГн		4,8

Двигатель включается при помощи двухступенчатого пускового реостата. Первая ступень сопротивлением 2,5 Ом выключается через 1,2 с после включения двигателя. Вторая ступень сопротивлением 2,035 Ом выключается через 2,5 с.

## 2.3. Особенности и ограничения

Существует ограничение на связь идентификаторов UM-модели с выходными величинами Matlab/Simulink модели, которое заключается в следующем. С выходными величинами Matlab/Simulink модели можно связывать только те идентификаторы, которые параметризуют силовые элементы: координаты точек прикрепления/действия сил, параметры жесткости, диссипации и прочее.

Согласно общим принципам UM запрещается назначать выходные величины расчетных схем Matlab/Simulink или SimInTech модели идентификаторам UM-модели, параметризующим инерционные параметры тел (массу и моменты инерции) и кинематику шарниров (положение шарнирных точек и прочее). Это приведет к получению неправильных результатов. Значения таких параметров можно менять между отдельными численными экспериментами, но не в процессе проведения одного численного эксперимента.

## 3. Работа с Matlab/Simulink

### 3.1. Моделирование с использованием Matlab Import

В этой части руководства мы подробно рассмотрим особенности подключения систем управления Matlab/Simulink к механическим моделям «Универсального механизма» посредством экспорта систем управления из Matlab/Simulink как динамических библиотек (DLL). Вначале остановимся на основных этапах, которые необходимо выполнить, чтобы правильно подключить систему управления к модели механической системы и выполнить моделирование динамики управляемого движения.

#### 3.1.1. Порядок работы

В общем случае моделирования динамики механических систем с подключением библиотек Matlab/Simulink предполагает выполнение следующих этапов.

- Описание модели системы управления в среде **Matlab/Simulink**.
- Экспорт созданной модели из **Matlab/Simulink** в виде DLL-библиотеки.
- Создание модели механической системы в программе **UM Input**.
- Загрузка подготовленной модели механической части в программу **UM Simulation**. Загрузка и подключение DLL-библиотеки Matlab/Simulink с помощью **Мастера связи с внешними библиотеками**.
- Моделирование динамики управляемого движения.

Модель системы управления из **Matlab/Simulink**, подключаемая в UM, рассматривается как черный ящик, который по некоторому закону преобразует входные величины в выходные. При добавлении системы управления в модель механической системы на входы системы управления назначаются переменные, которые создаются при помощи **Мастера переменных**. Выходные величины связываются с параметрами UM-модели.

Для реализации управляющих усилий со стороны системы управления в модель механической системы вводятся силы/моменты, значения или характеристики (например, коэффициент жесткости, коэффициент диссипации) которых задаются параметрами модели. Затем с помощью **Мастера связи с внешними библиотеками** эти параметры связываются с выходами системы управления.

### 3.1.2. Перевернутый маятник

Модель перевернутого маятника без подключенной системы управления находится в каталоге [{Данные УМ}\SAMPLES\TUTORIAL\inv\\_pend](#). Перед началом урока, проверьте ее наличие в указанном каталоге. Если такой модели нет, то скачайте ее из интернета по адресу: [www.universalmecanism.com/download/10/inv\\_pend.zip](http://www.universalmecanism.com/download/10/inv_pend.zip). В данном уроке мы не будем подробно разбирать этапы создания модели механической части, ограничимся только описанием связи механической части с системой управления Matlab/Simulink.

Готовая модель перевернутого маятника с подключенной системой управления находится в каталоге [{Данные УМ}\SAMPLES\Simulink\inv\\_pend\\_ctrl](#)<sup>2</sup>.

---

<sup>2</sup> [www.universalmecanism.com/download/90/inv\\_pend\\_ctrl.zip](http://www.universalmecanism.com/download/90/inv_pend_ctrl.zip)

### 3.1.2.1. Экспорт системы управления из Matlab/Simulink

Экспорт системы управления из Matlab/Simulink заключается в компиляции подготовленной модели в динамическую библиотеку (DLL), которая впоследствии подключается к программному комплексу «Универсальный механизм».

**Замечание.** Здесь предполагается, что вы знакомы с основами моделирования в среде Matlab/Simulink. В любом случае сейчас вы можете пропустить данный параграф и вернуться к нему позднее. Результатом экспорта системы управления из Matlab/Simulink является библиотека `inv_pend\pendpid_cntr.dll`. Вы можете пропустить этап создания dll-библиотеки для системы управления и воспользоваться уже подготовленной.

Процесс экспорта модели системы управления состоит из следующих этапов.

1. Подготовка модели системы управления к экспорту.
2. Копирование необходимых для компиляции модели файлов в каталог с моделью системы управления.
3. Настройка процесса компиляции и собственно компиляция модели.

#### Подготовка модели системы управления к экспорту

Для того чтобы модель системы управления было возможно экспортировать в dll-библиотеку, необходимо исключить из нее компоненты производящие какие-либо операции ввода/вывода, также компоненты, для которых в системе Matlab/Simulink отсутствуют исходные коды или компиляция которых невозможна.

Помимо этого в модель системы управления следует включить компоненты «In» и «Out», которые будут обеспечивать связь модели системы управления с моделью ПК «Универсальный механизм». В нашей модели перевернутого маятника входной переменной для системы управления является угол отклонения маятника от вертикали. Соответственно нужно предусмотреть один вход в модель системы управления. В свою очередь для организации управления перевернутым маятником в нашей модели системы управления предусмотрен один выход – усилие, подаваемое на тележку.

Подготовленная модель системы управления перевернутым маятником в Matlab/Simulink представлена на рис. 3.1.

**Замечание.** Вы можете самостоятельно собрать модель, представленную на рис. 3.1 или же загрузить готовую модель Matlab/Simulink из файла `SAMPLES\TUTORIAL\inv_pend\pendpid_cntr.mdl`.

**Замечание.** В силу особенностей компилирования моделей Matlab в них нельзя использовать блок **Derivative**. Вместо него необходимо использовать блок **Transfer Fcn** с соответствующей передаточной функцией.

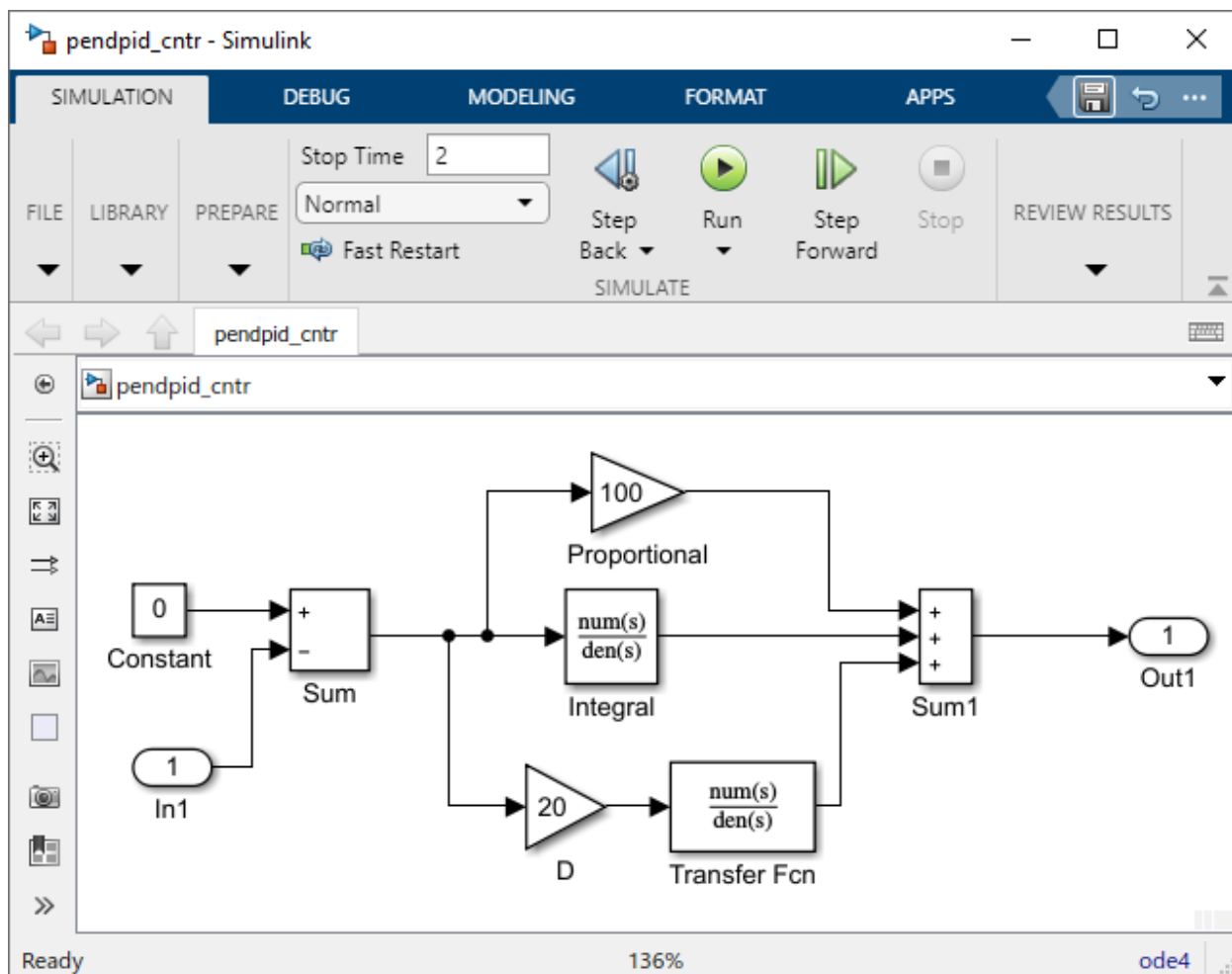


Рис. 3.1. Модель системы стабилизации перевернутого маятника

### Копирование файлов и настройка путей

1. Для компиляции модели системы управления необходимо скопировать некоторые вспомогательные файлы в каталог, где размещена модель Matlab/Simulink. Дальнейшие действия зависят от версии Matlab, которую вы используете.

#### Matlab 8.X / R2013-R2015 32-bit

Из каталога {Данные УМ}\Simulink\R2013\_R2015\x32 в каталог, содержащий файл модели, скопируйте следующие файлы: **rsim.tlc**, **rsim\_vc.tmf** и **um.tls**.

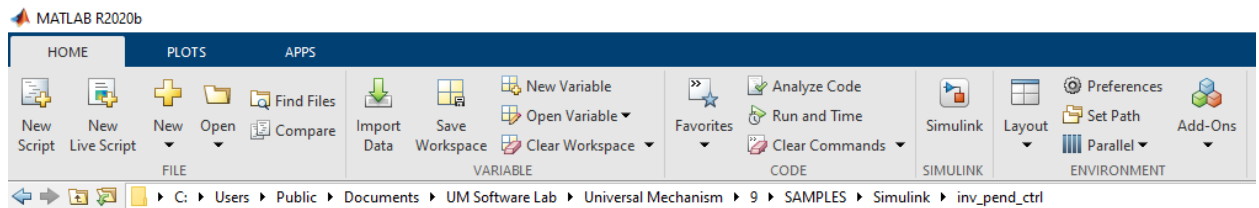
#### Matlab 8.X / R2013-R2015 64-bit

Из каталога {Данные УМ}\Simulink\R2013\_R2015\x64 в каталог, содержащий файл модели, скопируйте следующие файлы: **rsim.tlc**, **rsim\_vc.tmf** и **um.tls**.

#### Matlab 9.X / R2016-R2021 64-bit

Из каталога {Данные УМ}\Simulink\R2016\_R2020x64 скопируйте все файлы в каталог, содержащий файл модели (**rsim.tlc**, **rsim\_vc.tmf** и **um.tls**).

- Помимо этого необходимо, чтобы поле **Current Directory (Current Folder)** имело значение каталога, в котором находится файл экспортируемой модели системы управления.



## Настройка процесса компиляции и компиляция модели

Перейдите в командное окно системы Matlab. С помощью команды **"mex -setup"** (в более свежих версиях – **"mex -setup C++"**) выберите один из поддерживаемых вашей версией Matlab компиляторов семейства **Microsoft Visual C/C++**.

```
Command Window
>> mex -setup C++
MEX configured to use 'Microsoft Visual C++ 2019' for C++ language compilation.
fx >> |
```

**Замечание.** Для компиляции ваших моделей Matlab/Simulink используйте только компиляторы из семейства **Microsoft Visual C/C++**. Обратите внимание, что каждая конкретная версия Matlab поддерживает только определенные версии компиляторов **Microsoft Visual C/C++**. В выходе новых релизов Matlab добавляется поддержка новых релизов **Microsoft Visual C++** и прекращается поддержка устаревших релизов **Microsoft Visual C++**.

Таблицу совместимости компиляторов для разных версий Matlab можно найти по следующей ссылке:

[https://www.mathworks.com/support/sysreq/previous\\_releases.html](https://www.mathworks.com/support/sysreq/previous_releases.html).

Дальнейшие действия по компиляции модели зависят от версии Matlab, которую вы используете.

**Компиляция модели в Matlab 8.X / R2013-R2015**

1. Перейдите в окно модели. В меню **Code** выберите пункт **C/C++ Code**, после чего выберите пункт **Code Generation Options**.
2. В поле **System target file** задайте значение **rsim.tlc**.
3. В поле **Language** установите значение **C**.
4. Убедитесь, что в поле **Make command** задано **make\_rtw**.
5. В поле **Template makefile** выберите **rsim\_vc.tmf**.
6. В дереве объектов конфигурирования (диалоговое окно, слева) выберите **Solver** и в поле **Solver options | Type** выберите значение **Fixed Step**.
7. В поле **Solver options | Solver** выберите наиболее подходящий, по вашему мнению, численный метод решения, например **ode4 (Runge-Kutta)**.
8. Для запуска процесса компиляции и сборки dll-библиотеки вернитесь в поле **Code Generation** и нажмите кнопку **Build**.

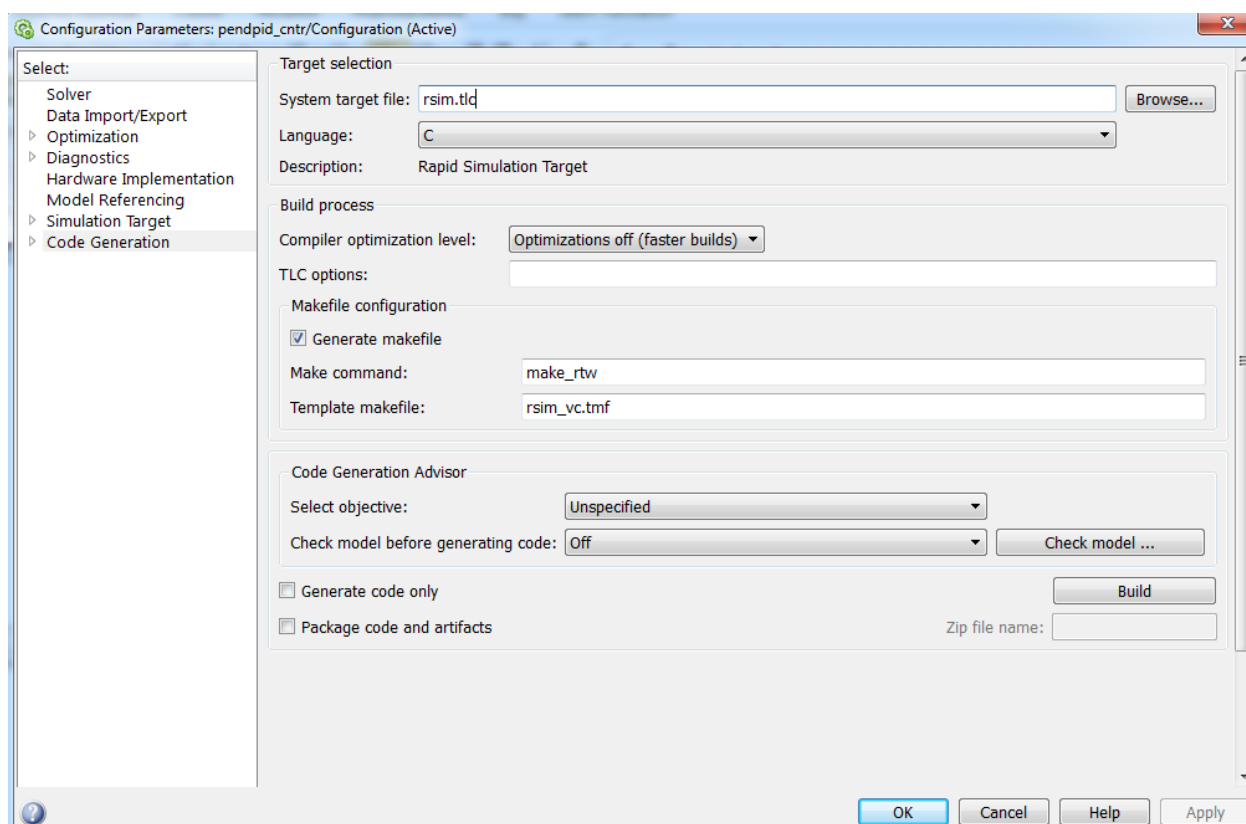


Рис. 3.2. Настройка процесса компиляции в Matlab R2012-R2015

Процесс компиляции сопровождается выводом сообщений о ходе процесса в командное окно системы Matlab/Simulink. Успешная компиляция библиотеки в частности для нашей модели системы управления завершается сообщением «Creating library ..\pendpid\_cntr.lib and object ..\pendpid\_cntr.exp».

**Замечание.** Для работы динамической библиотеки созданной в релизах Matlab R2011-R2015 в программном комплексе UM необходимо, чтобы на компьютере была установлена соответствующая версия Matlab.

В результате выполнения компиляции в рабочем каталоге появятся файлы библиотеки. Для нашего примера это следующие файлы: **pendpid\_cntr.dll**, **pendpid\_cntr.lib**, **pendpid\_cntr.exp**. А также подкаталог, содержащий исходные коды скомпилированной библиотеки. Для нашей модели он будет называться **pendpid\_cntr\_rsim\_rtw**.

### Компиляция модели в Matlab 9.X / R2016-R2020

1. Перейдите в окно модели [pendpid\_ctrl], см. рис. 3.1.
2. В верхнем меню в галерее **MODELING** выберите **Model Settings / Model Settings** (см. рис. 3.3) или же щёлкните правой кнопкой мыши на белом фоне модели и в контекстном меню выберите пункт **Model Configuration Parameters**. Появится окно **Configuration Parameters**, см. рис. 3.4.

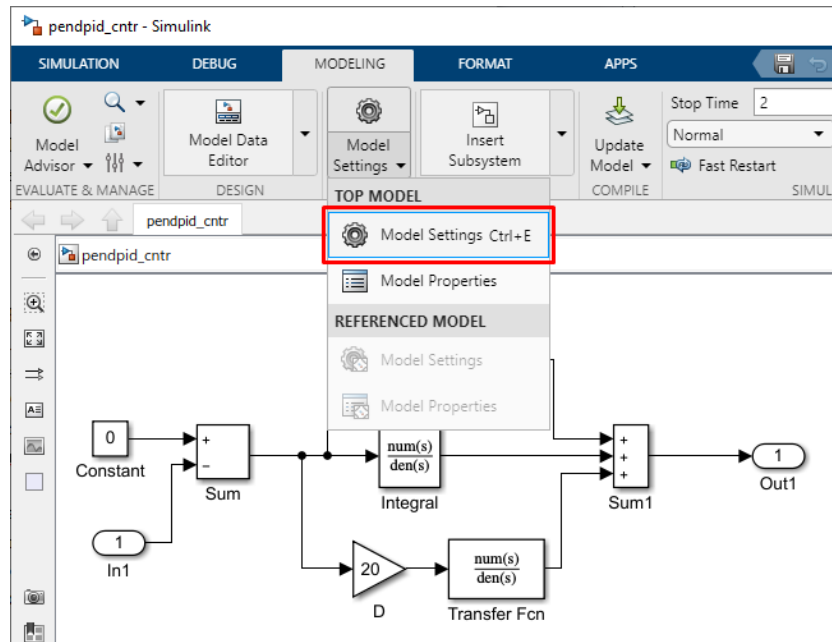


Рис. 3.3. Вызов окна параметров модели в Matlab 9.X (R2016-R2020)

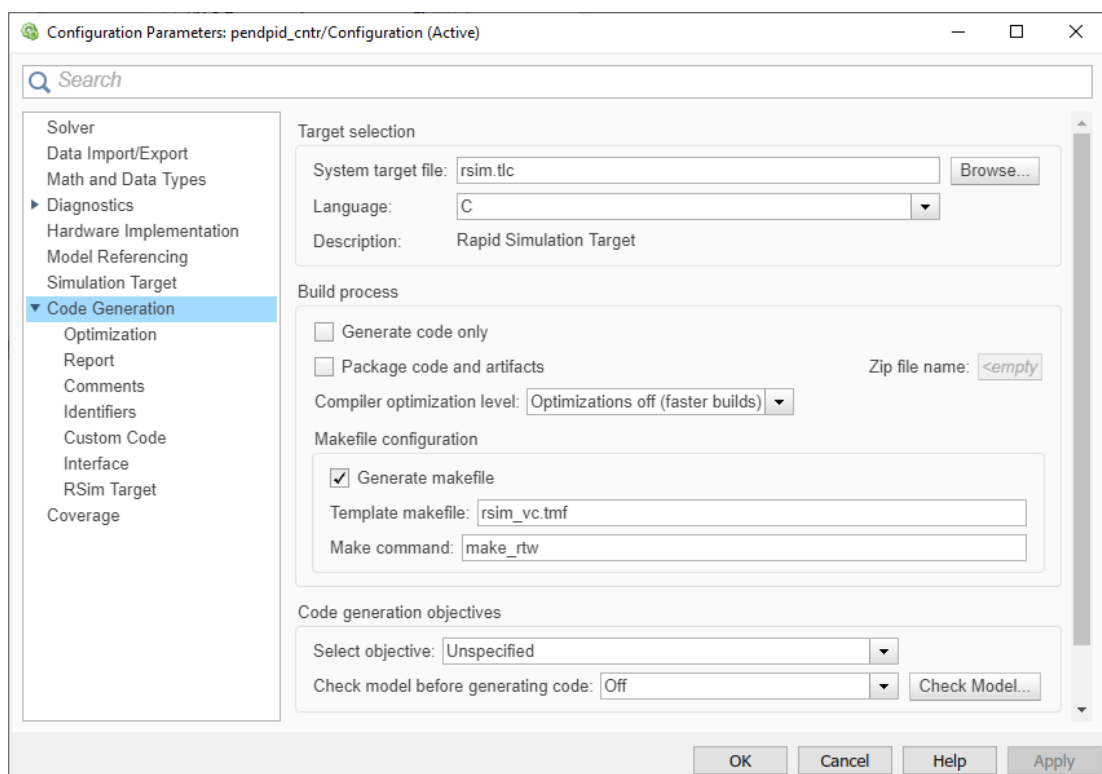


Рис. 3.4. Настройка процесса компиляции в Matlab 9.X (R2016-R2020)

3. В дереве объектов конфигурирования слева выберите **Code Generation**, как это показано на рис. 3.4.
4. В поле **System target file** задайте значение **rsim.tlc**.
5. В поле **Language** установите значение **C**.
6. В поле **Template makefile** выберите **rsim\_vc.tmf**.
7. Убедитесь, что в поле **Make command** задано **make\_rtw**.
8. Затем в дереве объектов конфигурирования (диалоговое окно, слева) выберите **Solver**.
9. В поле **Solver options | Type** выберите значение **Fixed Step**.
10. В поле **Solver options | Solver** выберите наиболее подходящий, по вашему мнению, численный метод решения, например **ode4 (Runge-Kutta)**.
11. Далее в разделе **Code Generation / Optimization** в выпадающем списке **Default parameter behavior** выберите значение **Tunable**, см. рис. 3.5. Это даст возможность менять параметры системы управления на стороне УМ после импорта DLL. Если не проделать эту операцию, то список параметров модели после импорта в УМ будет пустой и менять их будет невозможно.
12. Закройте окно **Configuration Parameters**.

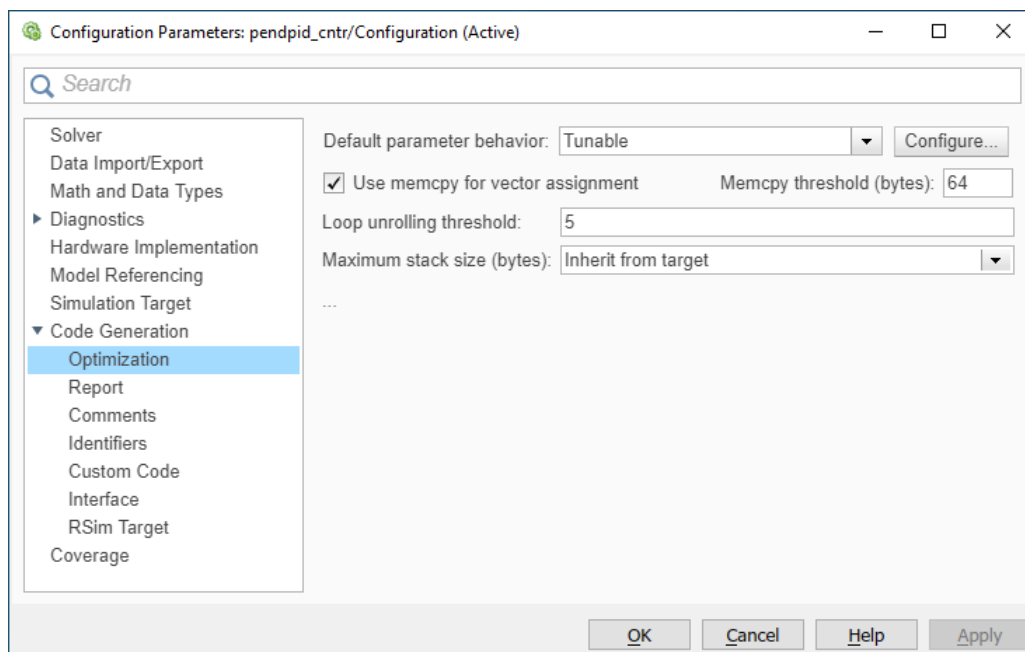


Рис. 3.5. Задание настраиваемых параметров расчётной схемы в Matlab 9.X (R2016-R2020)

13. Для запуска процесса компиляции и сборки dll-библиотеки в окне модели в галерее **APPS** выберите **Simulink Coder** затем нажмите кнопку **Generate code and build model**, см. рис. 3.6.

Процесс компиляции сопровождается выводом сообщений о ходе процесса в окно **Diagnostic Viewer** системы Matlab/Simulink, см. рис. 3.7. Успешная компиляция библиотеки в частности для нашей модели системы управления завершается сообщением «*Build process completed successfully*», см. рис. 3.7.

**Замечание.** Для работы динамической библиотеки в программном комплексе UM необходимо, чтобы на компьютере была установлена соответствующая версия Matlab.

В результате выполнения компиляции в рабочем каталоге появится файл библиотеки `pendpid_ctr.dll` и ряд сопутствующих файлов. Кроме того появится подкаталог, содержащий исходные коды скомпилированной библиотеки. Для нашей модели он будет называться `pendpid_ctr_rsim_rtw`.

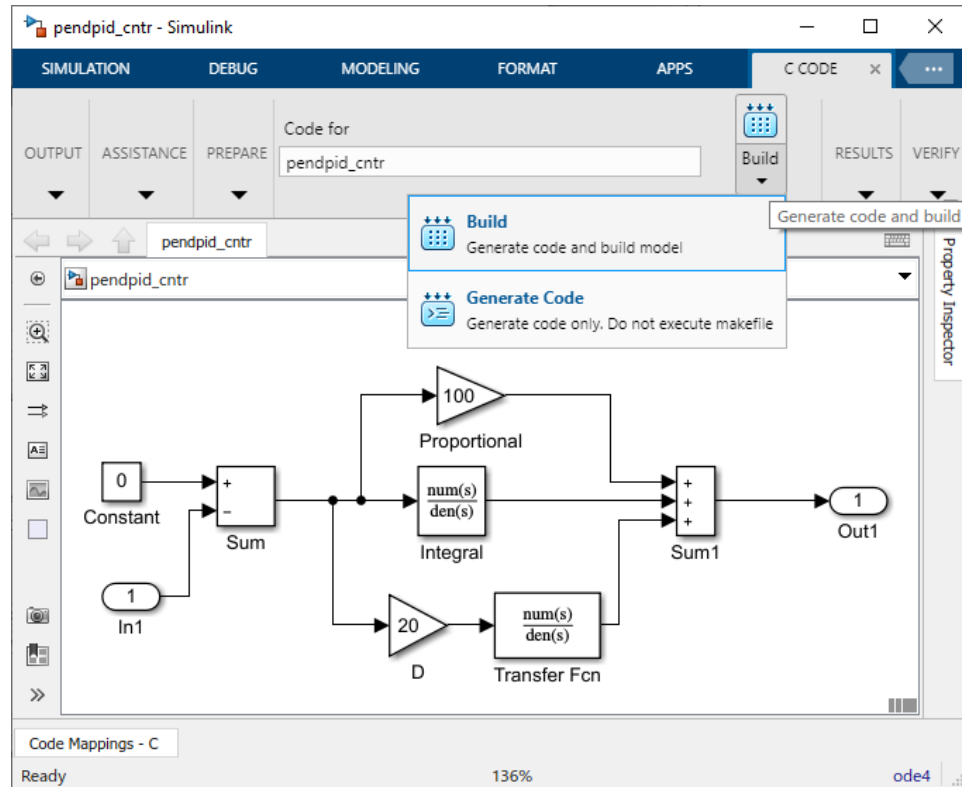


Рис. 3.6. Кнопка генерации кода и сборки DLL в Matlab 9.X (R2016-R2020)

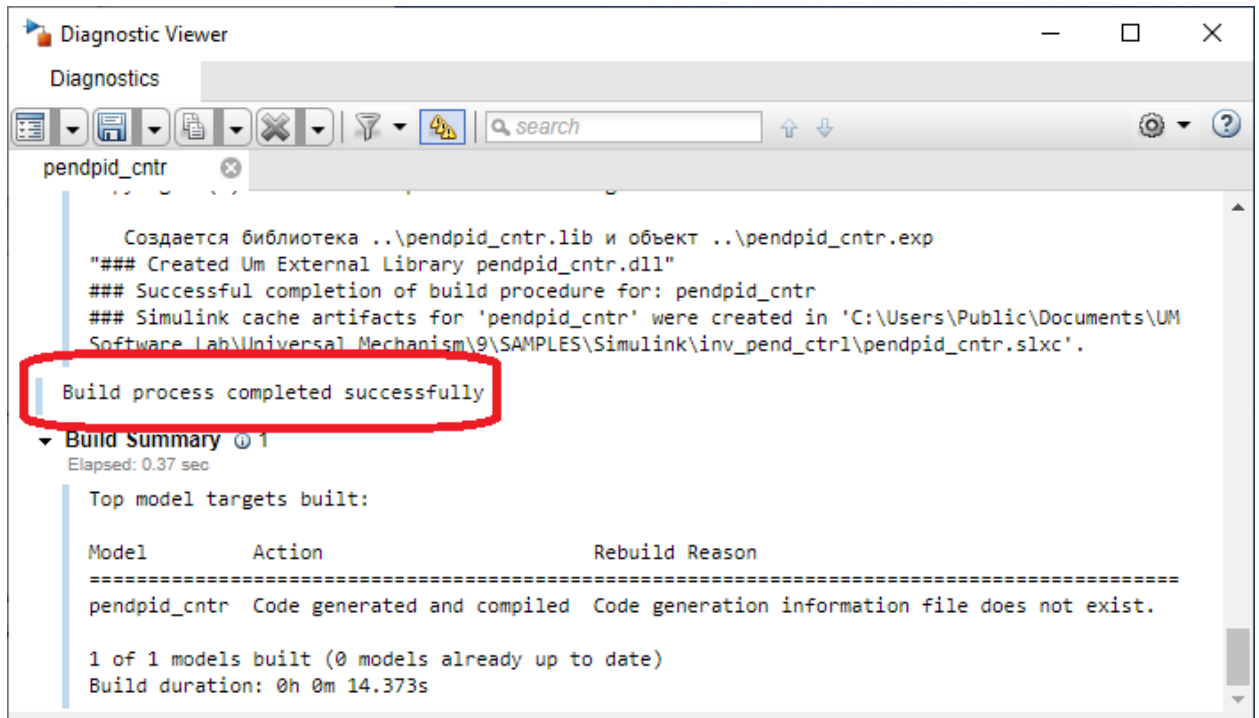


Рис. 3.7. Окно Diagnostic Viewer в Matlab 9.X (R2016-R2020)

### 3.1.2.2. Подключение библиотеки Matlab/Simulink в UM

#### Загрузка модели механической части

1. Запустите программу моделирования **UM Simulation**.
2. Загрузите модель [{Данные УМ}\SAMPLES\TUTORIAL\inv\\_pend](#).
3. Откройте новое анимационное окно.

#### Загрузка модели Matlab/Simulink

1. Выберите пункт меню **Инструменты | Внешние интерфейсы | Интерфейс с внешними библиотеками...**

Появится окно **Мастера связи с внешними библиотеками**.

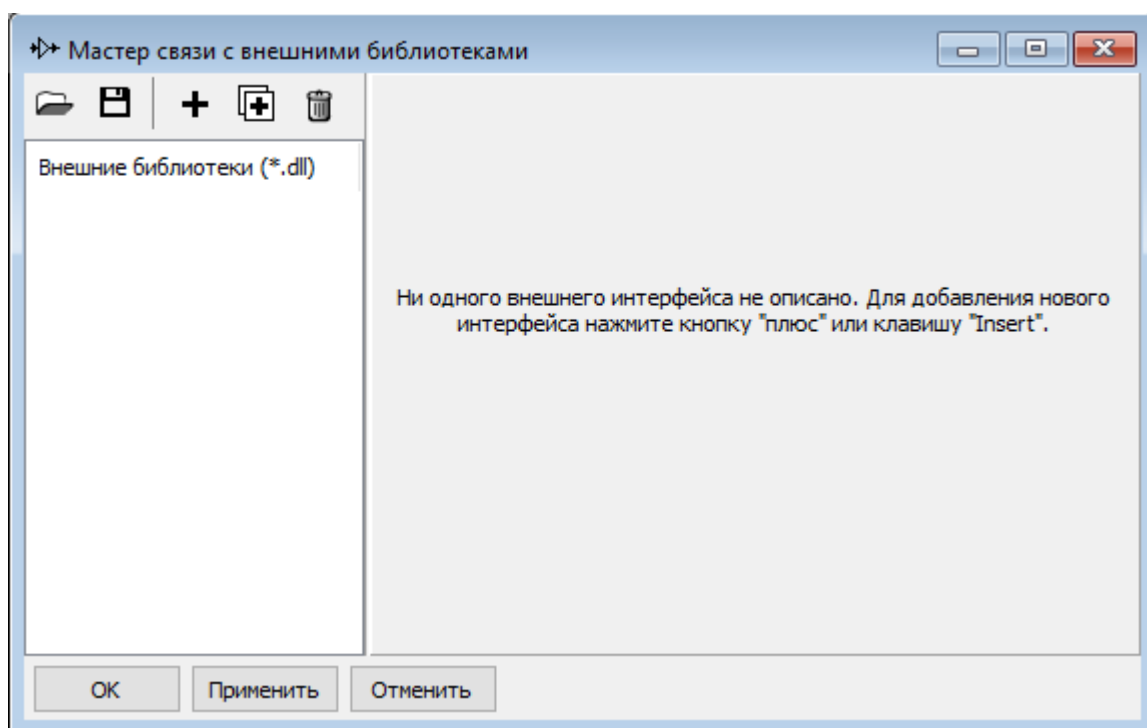
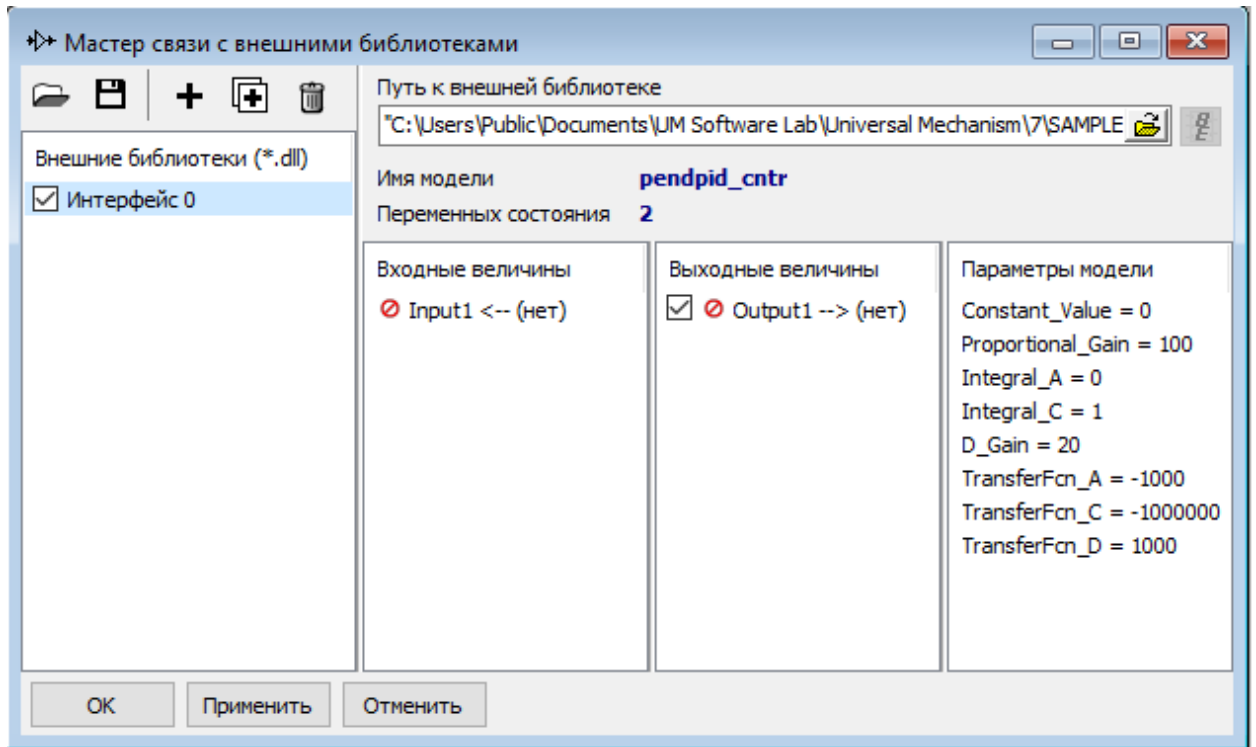


Рис. 3.8. Мастер связи с внешними библиотеками

2. Нажмите кнопку **+** для добавления новой библиотеки **Matlab/Simulink**.
3. В поле **Путь к внешней библиотеке** выберите файл **pendpid\_cntr.dll**. Если вы создали выходную dll самостоятельно, то выберите ее, иначе выберите готовую dll системы управления из каталога **..\inv\_pend**.

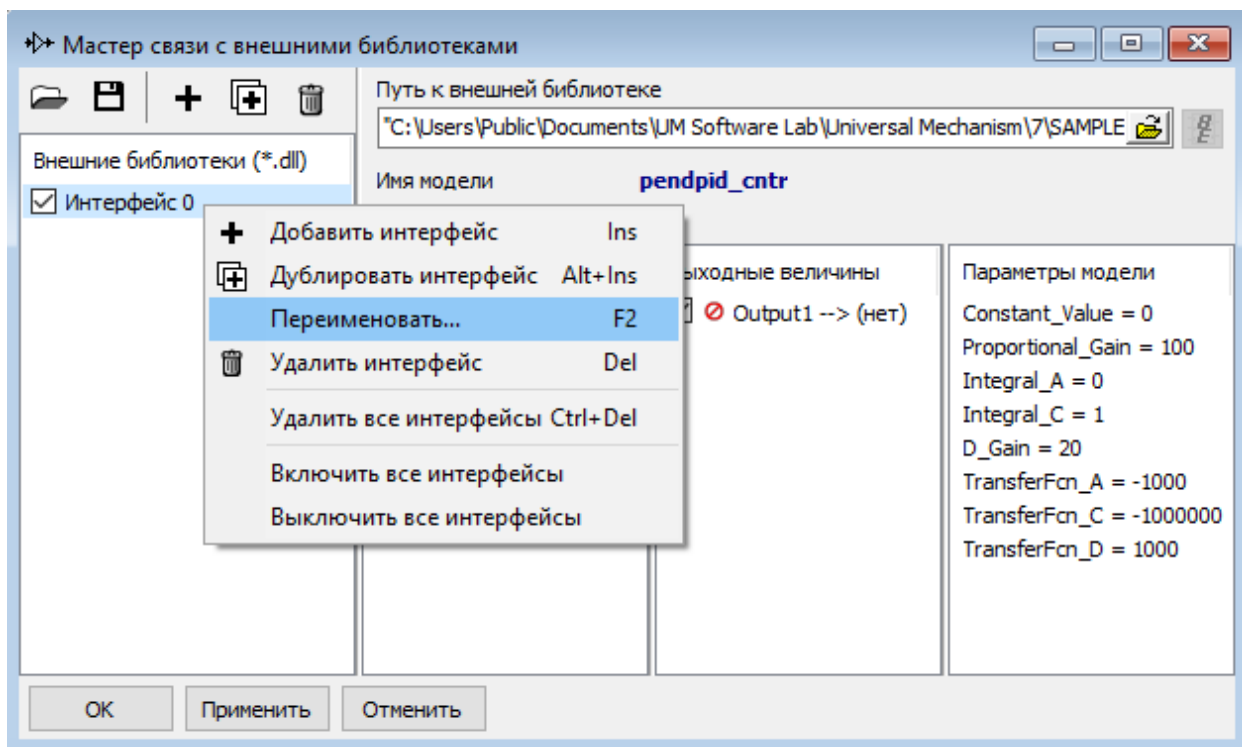
**Мастер связи** загрузит указанную модель и установит число входных и выходных величин модели системы управления, а также число параметров этой модели и переменных состояния.

В данном случае в окне появится одна входная и одна выходная величина, см. рис. ниже. На вход системы управления мы подадим угол отклонения маятника от вертикали, а выходную величину подадим в качестве усилия на тележку.




## Переименование интерфейса

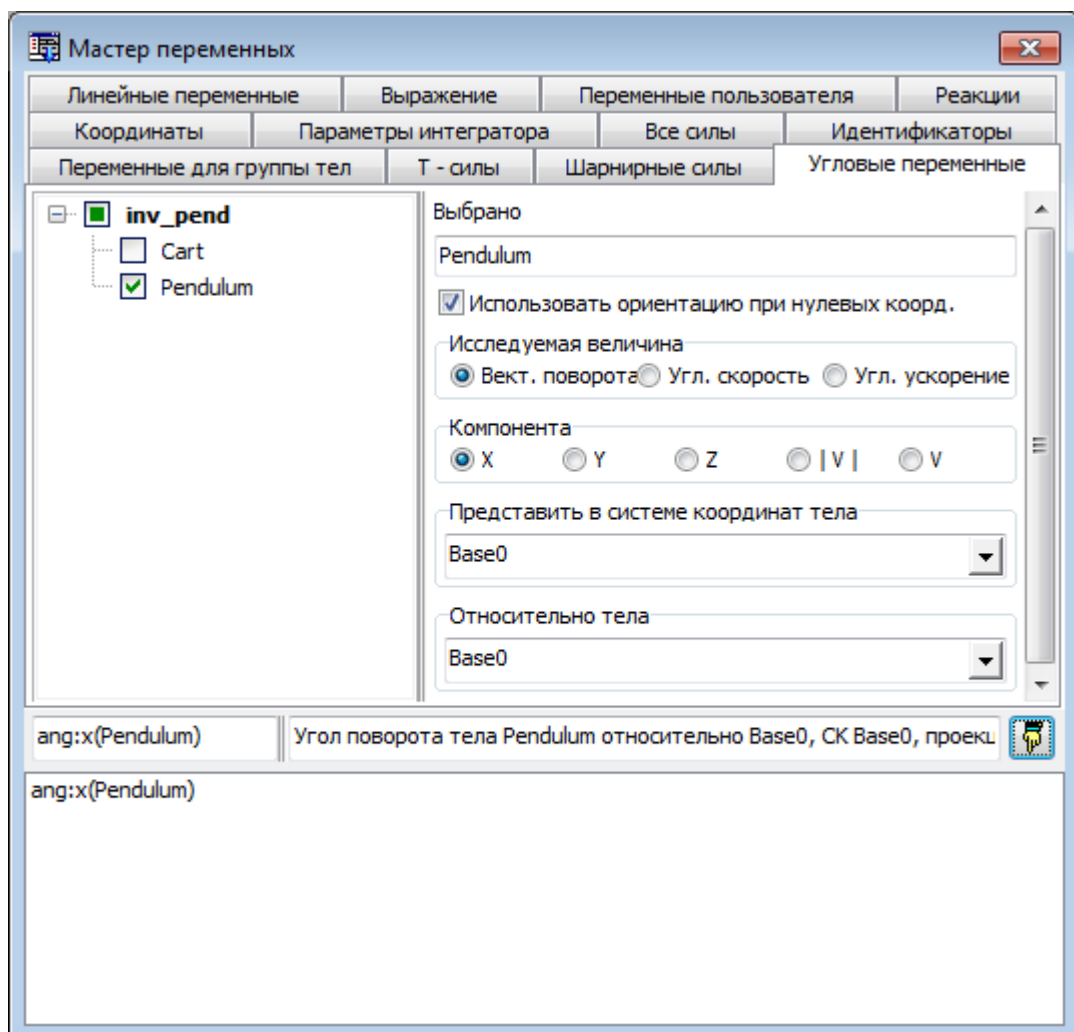
1. Выберите в списке **Внешние библиотеки (\*.dll)** текущий интерфейс **Интерфейс 0**, правой кнопкой мыши вызовите контекстное меню и в нем выберите пункт **Переименовать...**, или нажмите клавишу **F2**.
2. В поле редактирования названия введите **Control force** и нажмите клавишу **Enter**.



## Назначение входных величин модели Matlab/Simulink

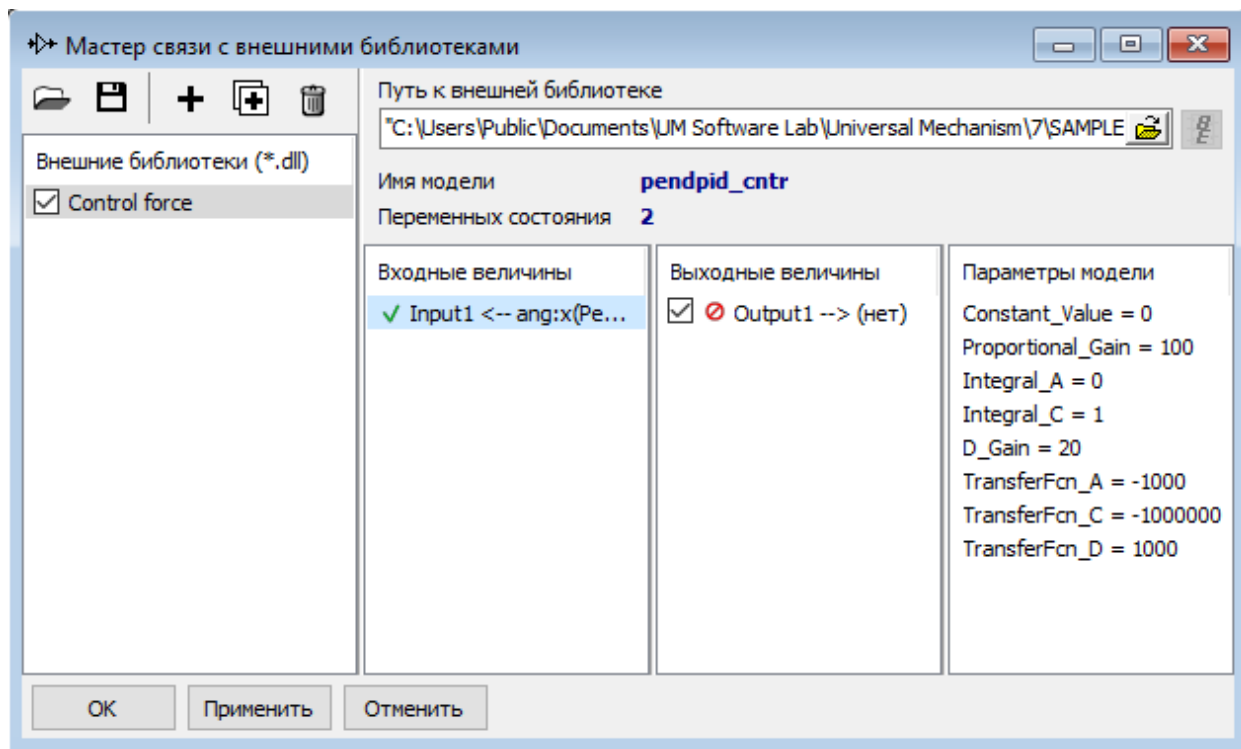
С помощью мастера переменных сформируем переменную «угол отклонения маятника от вертикали» и назначим ее в качестве входной величины для системы управления.

1. Запустите **Мастер переменных** (пункт меню **Инструменты | Переменные | Мастер переменных...**).
2. Перейдите на вкладку **Угловые переменные**.
3. В списке тел слева выберите **Pendulum**, в поле **Исследуемая величина** выберите **Вект.(ор) поворота**, а в поле **Компонента** выберите **X**.
4. Создайте переменную кнопкой . Переменная **ang:x(Pendulum)** появится в контейнере переменных.



5. Перетащите мышкой созданную переменную **ang:x(Pendulum)** в окно **Мастера связи с внешними библиотеками** на вход **Input1**.

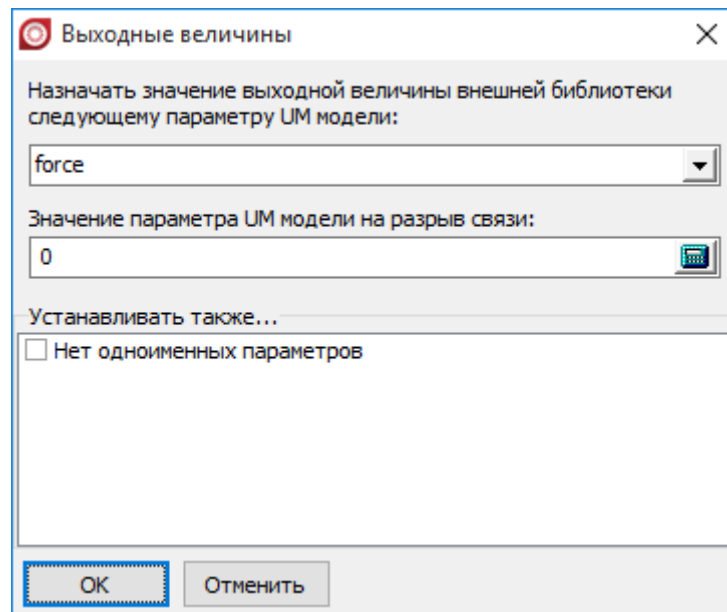
Вход модели **Input1** станет помечен зеленой галочкой, что означает, что на данный вход модели Matlab/Simulink назначена переменная.



**Назначение выходных величин модели Matlab/Simulink**

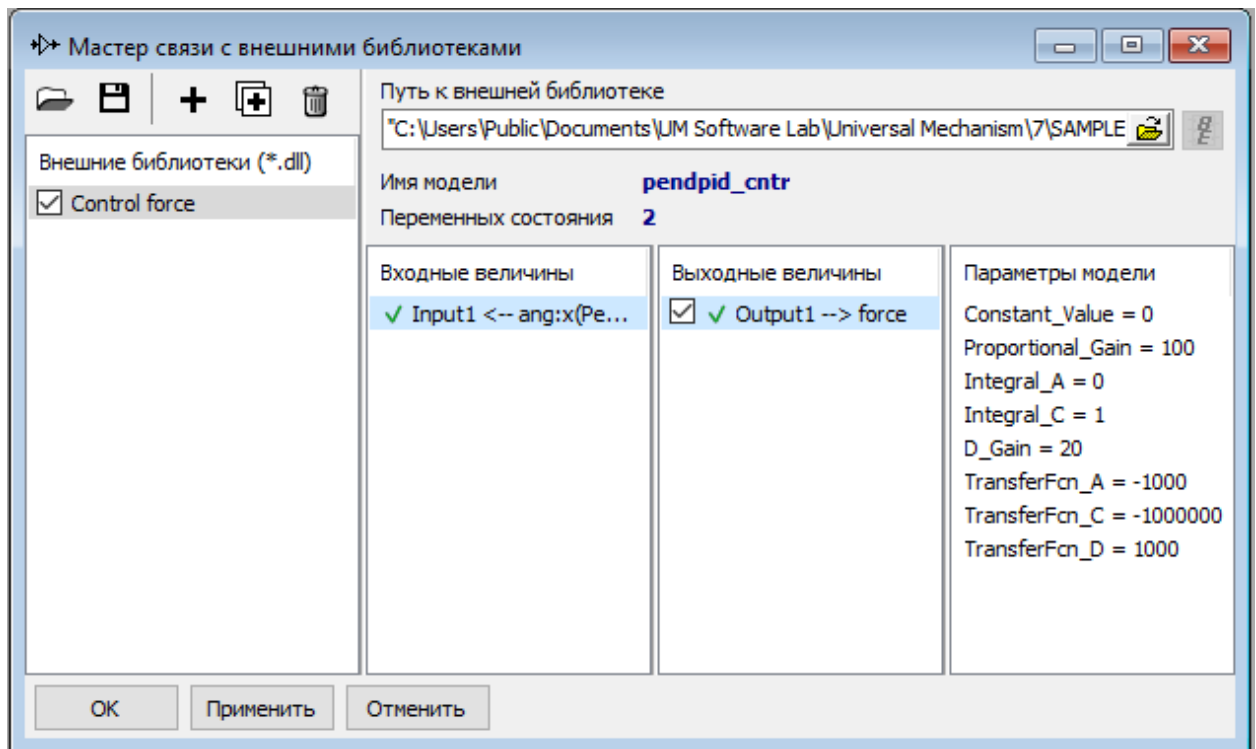
Выходные величины модели Matlab/Simulink связываются с параметрами UM-модели.

1. Дважды щелкните на элементе **Выходные величины | Output1**. Появится диалоговое окно назначения параметров UM-модели на выход модели Matlab/Simulink.
2. В поле **Параметр UM модели** выберите **force**.
3. Закройте окно кнопкой **ОК**.



Модель Matlab/Simulink полностью связана с моделью механической системы.

4. Нажмите **Применить** для актуализации настроек без закрытия окна **Мастера связи с внешними библиотеками** или **ОК** для сохранения данных с закрытием окна.

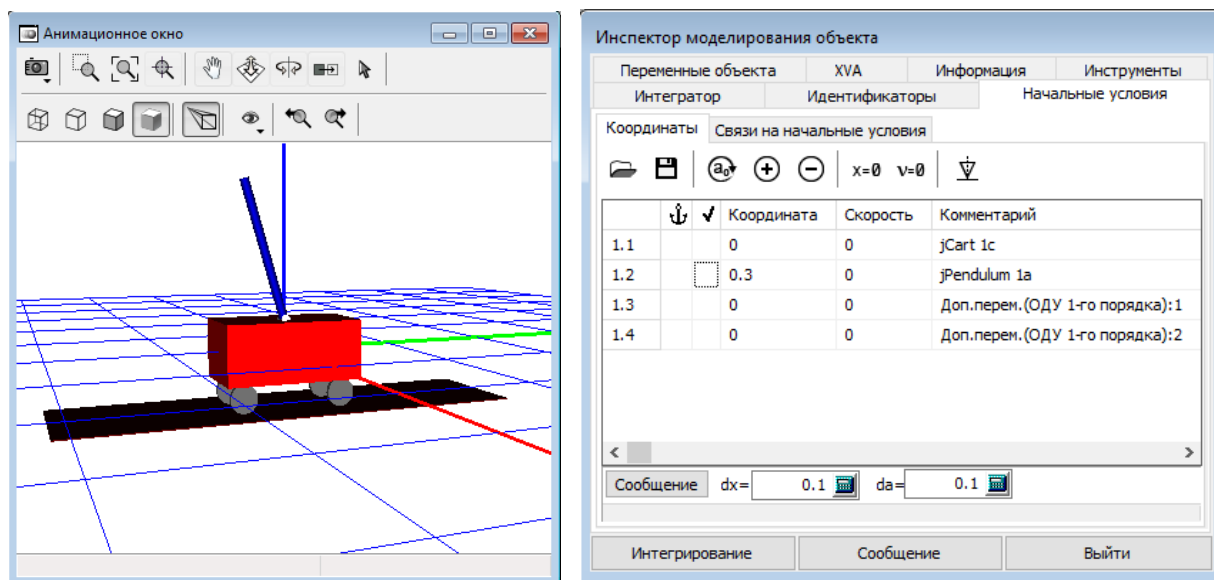


### 3.1.2.3. Моделирование движения

Теперь наша модель перевернутого маятника полностью описана. Сейчас мы отклоним маятник из положения равновесия и проверим, действительно ли система управления работает.

1. Выберите пункт меню **Анализ | Моделирование...** Откроется **Инспектор моделирования объекта**.
2. Перейдите на вкладку **Начальные условия**.
3. Отклоним маятник на 0.3 радиана от положения равновесия. В поле второй координаты (Координата/1.2) введите **0.3**.

**Замечание.** Первая координата – положение тележки относительно оси Y. Третья и четвертая координаты – переменные состояния модели Matlab/Simulink.



4. Перейдите на вкладку **Интегратор**, в поле **Численный метод** выберите **Метод Park**, в разделе установки условий завершения установите **Время > 0.1**, в поле **Шаг представления результатов** установите значение **0.0002**, а в поле **Погрешность** установите **1E-7**.
5. Нажмите кнопку **Интегрирование**.



Действительно, как вы можете наблюдать, результаты моделирования показывают, что система управления стабилизировала перевернутый маятник.

### Построение вектора управляющей силы и графиков переменных

Выведем в анимационное окно график управляющей силы, действующей на тележку, а в графические окна – величину этой силы и угол поворота маятника.

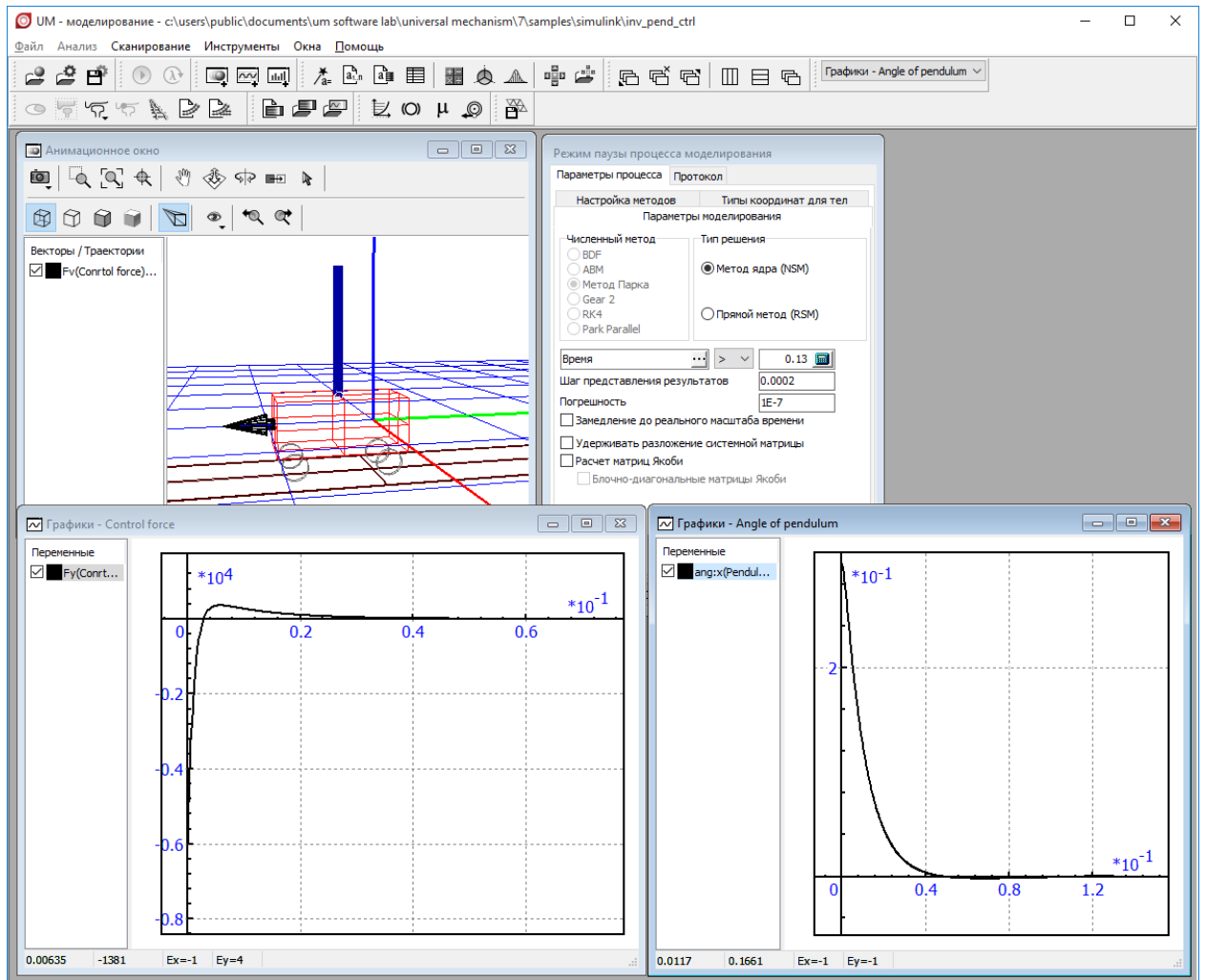
Сформируем переменную – вектор силы, для отрисовки в анимационном окне.

1. Откройте **Мастер переменных**.

2. Перейдите на вкладку **T-силы**.
3. В списке слева поставьте галочку возле силы **Control force**, в поле **Исследуемая величина** выберите **Сила**, в поле **Компонента** – **V** (вектор), в поле **Действует на тело** – **Cart**.
4. Нажав кнопку  поместите сформированную переменную в контейнер переменных и перетащите ее в анимационное окно.
5. Кнопкой  переключите анимационное окно в режим проволочной графики. Это позволит наблюдать вектор силы, даже когда он будет изображаться внутри тележки.

А теперь сформируем две переменные для отрисовки в графических окнах: величину управляющей силы и угол поворота маятника.

6. Откройте новое графическое окно. В нем будем строить график управляющей силы.
7. Перейдите обратно в **Мастер переменных**, на вкладку **T-силы**. В поле **Компонента** выберите **Y**. Поместите переменную в контейнер и перетащите ее в графическое окно.
8. Откройте еще одно графическое окно. В этом окне мы будем строить зависимость угла поворота маятника от времени.
9. Перейдите в окно **Мастера переменных** на вкладку **Угловые переменные**.
10. В списке тел выберите **Pendulum**, в поле **Исследуемая величина** выберите **Вект.(ор) поворота**, в поле **Компонента** – **X**.
11. Поместите переменную в контейнер переменных, а затем перетащите ее в графическое окно.
12. Закройте окно **Мастера переменных**.
13. Запустите процесс моделирования.



### 3.1.3. Двигатель постоянного тока

Модель механической части системы без подключенного электрического двигателя находится в каталоге [{Данные УМ}\SAMPLES\TUTORIAL\dc\\_motor](#). Перед началом урока, проверьте ее наличие в указанном каталоге. Если такой модели нет, то скачайте ее из интернета по адресу: [www.universalmechanism.com/download/90/dc\\_motor.zip](http://www.universalmechanism.com/download/90/dc_motor.zip).

Как и в предыдущем уроке, мы не будем подробно разбирать этапы создания моделей механической и электрической частей. Предполагается, что читатель может самостоятельно открыть и изучить модель механической части в Универсальном механизме, и модель двигателя постоянного тока – в Matlab/Simulink. В данном примере мы ограничимся только описанием связи механической части с двигателем, импортированным из Matlab/Simulink.

Готовая модель электромеханической системы находится в каталоге [{Данные УМ}\SAMPLES\Simulink\dc\\_motor\\_fin](#)<sup>3</sup>.

---

<sup>3</sup> [www.universalmechanism.com/download/90/dc\\_motor\\_fin.zip](http://www.universalmechanism.com/download/90/dc_motor_fin.zip)

### 3.1.3.1. Описание системы управления

Модель двигателя имеет один вход – угловую скорость ротора (рад/с) и три выхода: (1) электромагнитный момент, (2) ток якоря и (3) напряжения цепи якоря. Копии экрана Simulink-модели рассматриваемого двигателя приведены на рис. 3.9 и рис. 3.10.

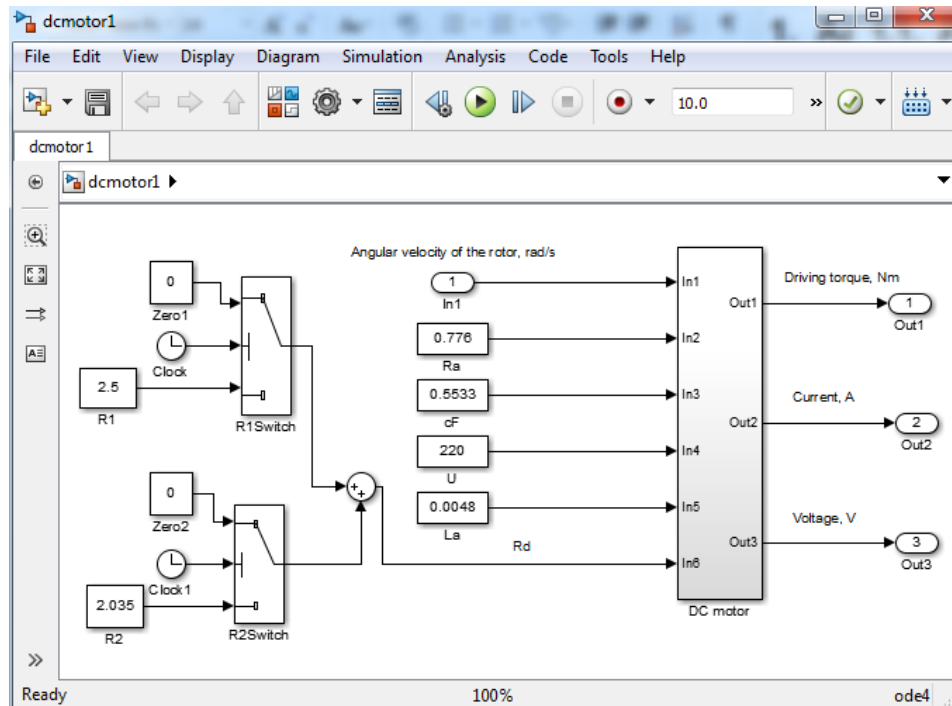


Рис. 3.9. Модель двигателя в Matlab/Simulink

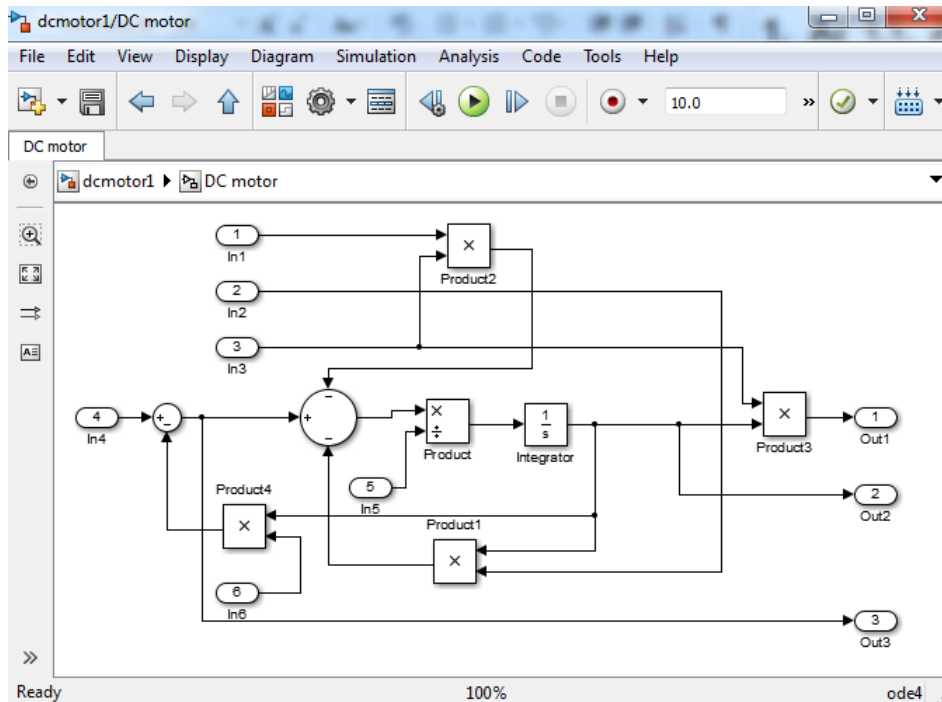


Рис. 3.10. Подсистема DC motor

### 3.1.3.2. Экспорт двигателя из Matlab/Simulink

Сейчас мы должны из \*.mdl файла, который содержит нашу Simulink-модель двигателя, получить файл \*.dll.

Matlab/Simulink модель двигателя постоянного тока находится в файле `..\dc_motor\dcmotor1.mdl`. Вы можете либо откомпилировать данный \*.mdl файл пользуясь инструкциями, изложенным в п. 3.1.2.1, либо пропустить этап компиляции файла \*.mdl и воспользоваться уже готовым файлом **dcmotor1.dll**.

### 3.1.3.3. Подключение библиотеки Matlab/Simulink в UM

#### Загрузка модели механической части

1. Запустите программу моделирования **UM Simulation**.
2. Откройте модель [{Данные УМ}\SAMPLES\TUTORIAL\dc\\_motor](#).
3. Откройте новое анимационное окно.

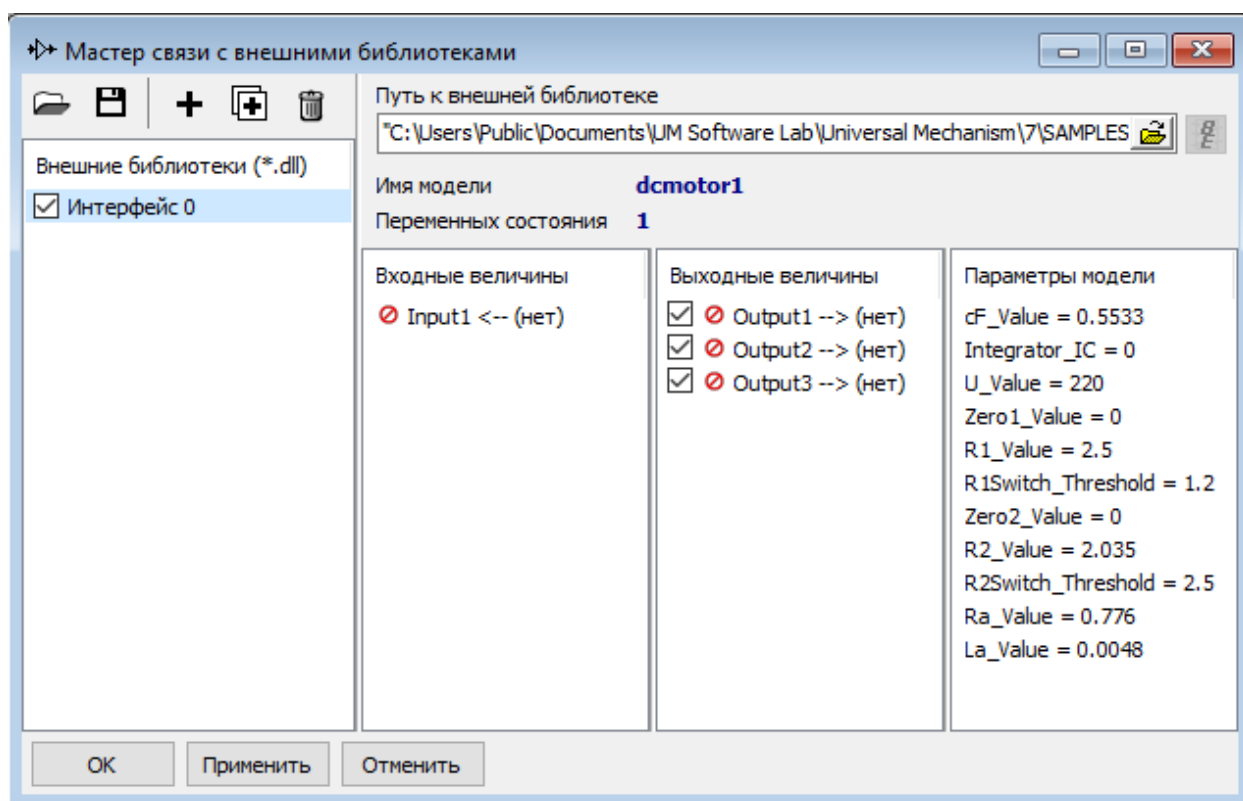
#### Загрузка модели Matlab/Simulink

4. Выберите пункт меню **Инструменты | Внешние интерфейсы | Интерфейс с внешними библиотеками...**

Появится окно **Мастера связи с внешними библиотеками**.

5. Нажмите кнопку **+** для добавления новой внешней библиотеки.
6. В поле **Путь к внешней библиотеке** выберите файл **dcmotor1.dll**. Если вы создали выходную библиотеку dll самостоятельно, то выберите ее, иначе выберите готовую dll системы управления из каталога **..\dc\_model**.

**Мастер связи** загрузит указанную модель, установит число входных и выходных величин модели двигателя, а также число параметров модели и переменных состояния. В данном случае в окне появится одна входная и три выходных величины, см. рис. ниже. На вход системы управления мы подадим угол поворота вала, первый выход назначим параметру **Ma**, второй и третий – параметрам **I** и **U**, соответственно.




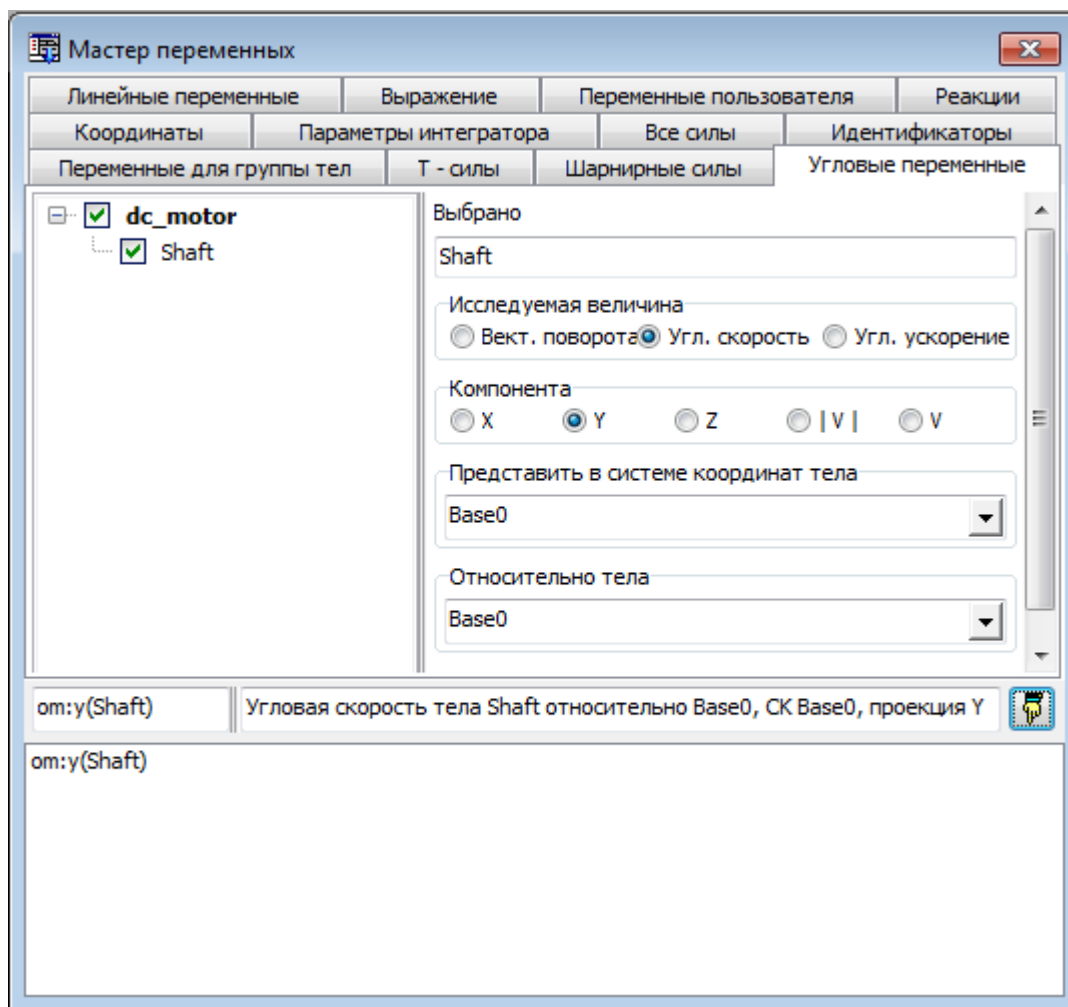
### Переименование интерфейса

1. В списке **Внешние библиотеки** выберите текущий интерфейс **Интерфейс 0**, правой кнопкой мыши вызовите контекстное меню и в нем выберите пункт **Переименовать...**, или нажмите клавишу **F2**.
2. В поле редактирования названия введите **DC motor** и нажмите клавишу **Enter**.

### Назначение входных величин модели Matlab/Simulink

С помощью мастера переменных сформируем переменную «угловая скорость вала» и назначим ее в качестве входной величины для системы управления.

1. Запустите **Мастер переменных** (пункт меню **Инструменты | Переменные | Мастер переменных...**).
2. Перейдите на вкладку **Угловые переменные**.
3. В списке тел слева выберите **Shaft**, в поле **Исследуемая величина** выберите **Угл. (овая) скорость**, а в поле **Компонента** выберите **Y**.
4. Создайте переменную кнопкой . Переменная **om:y(Shaft)** появится в контейнере переменных.



5. Перетащите мышкой созданную переменную **om:y(Shaft)** в окно **Мастера связи с внешними библиотеками** на вход **Input1**.

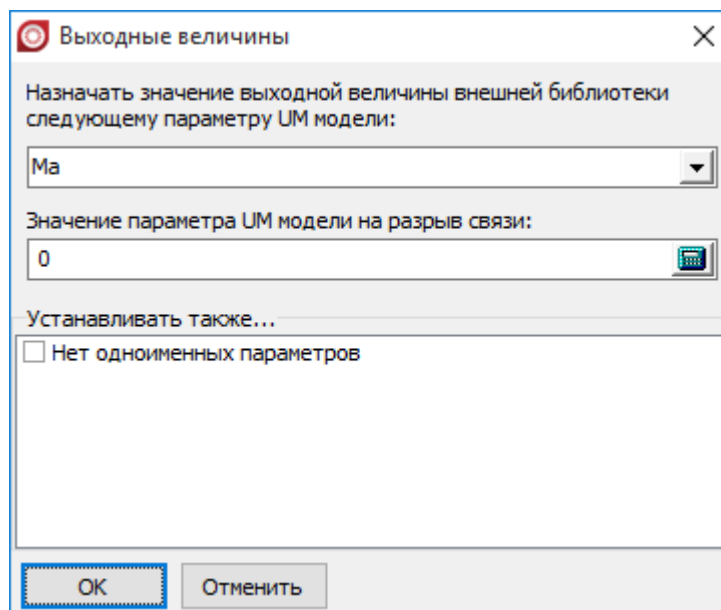
Вход модели **Input1** станет помечен зеленой галочкой, что означает, что на данный вход модели Matlab/Simulink назначена переменная.

6. Закройте окно **Мастера переменных**.

### Назначение выходных величин модели Matlab/Simulink

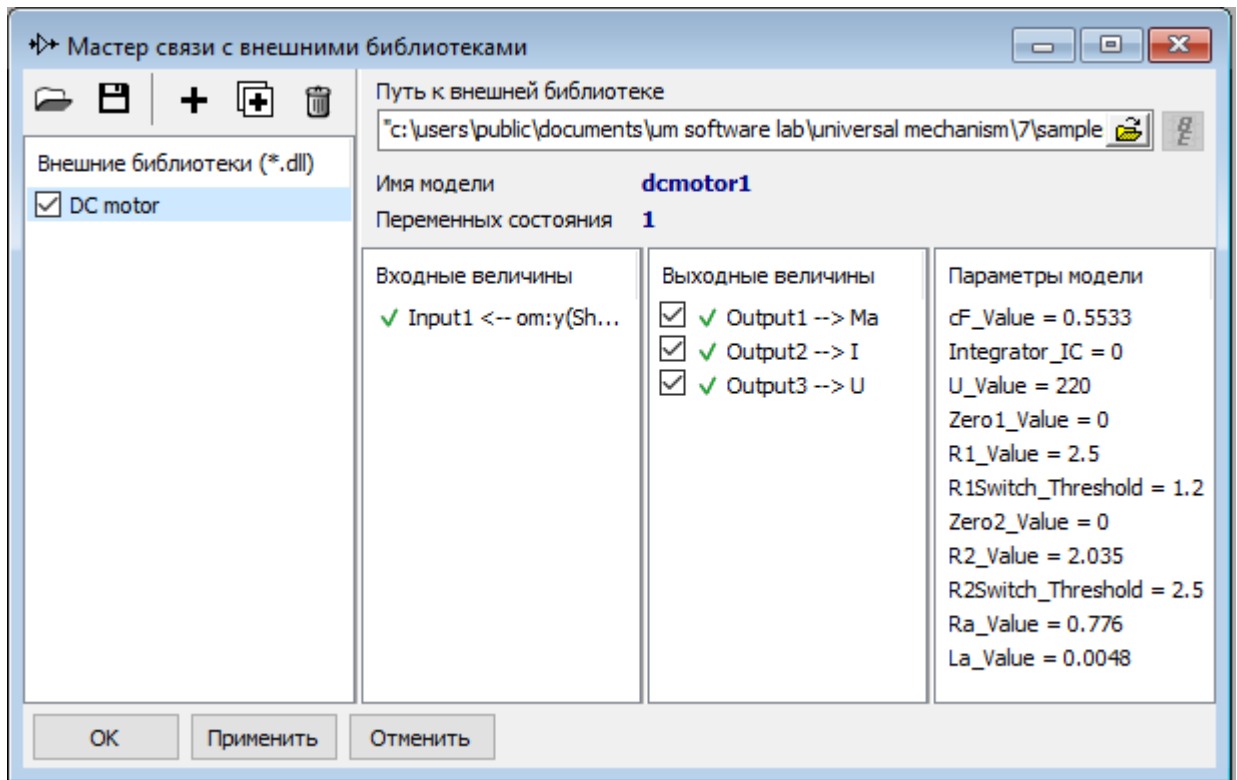
Выходные величины модели Matlab/Simulink связываются с параметрами UM-модели.

1. Дважды щелкните на элементе **Выходные величины | Output1**. Появится диалоговое окно связывания параметров UM-модели с выходами модели Matlab/Simulink.
2. В поле **Параметр UM модели** выберите **Ma**.
3. Закройте окно кнопкой **ОК**.



4. Выход **Output2** назначьте параметру **I**.
5. Выход **Output3** назначьте параметру **U**.

6. Нажмите кнопку **ОК** для сохранения данных с закрытием окна **Мастера связи с Matlab/Simulink**.

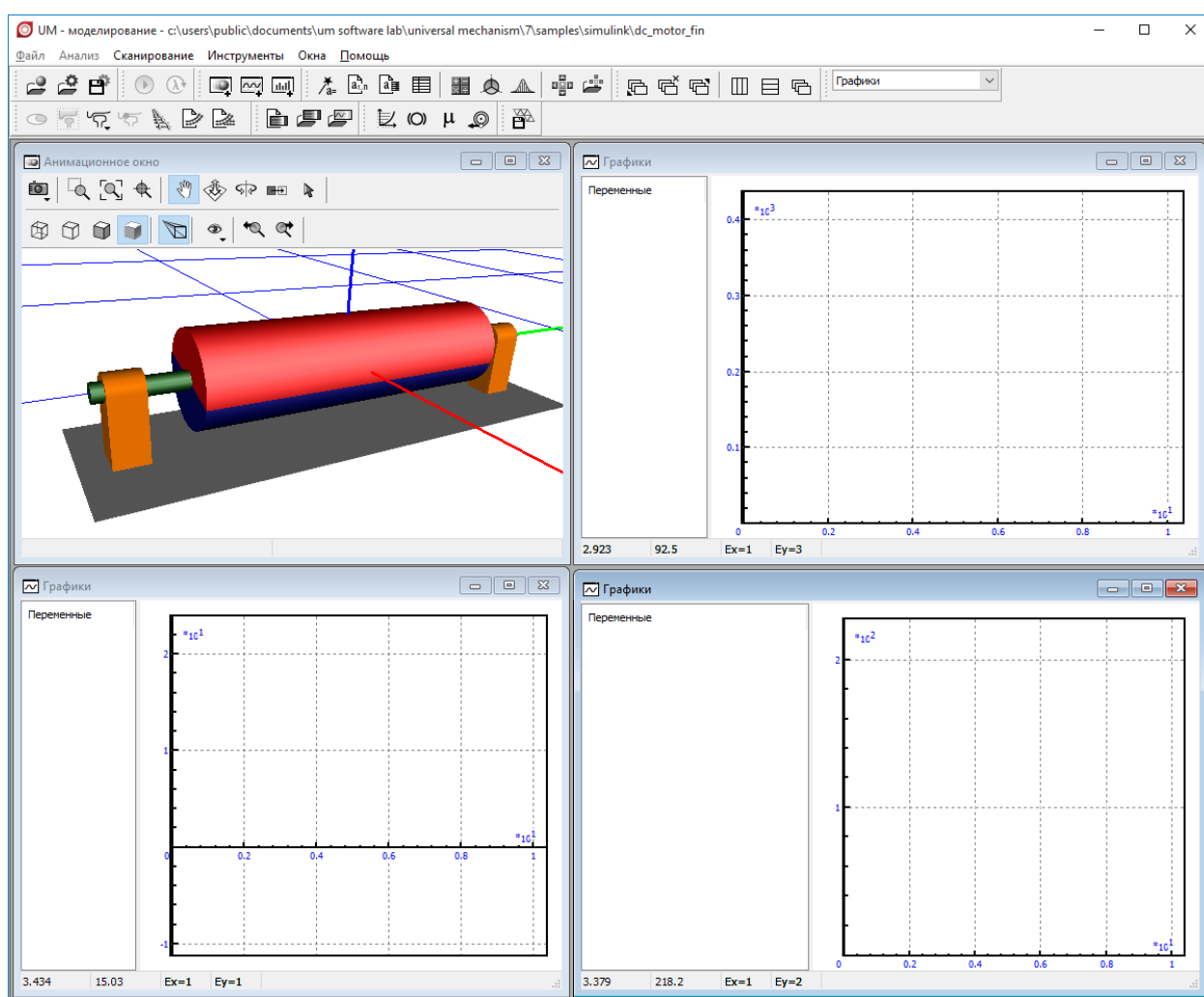


### 3.1.3.4. Моделирование движения

Модель подготовлена для дальнейшего анализа. Сейчас мы откроем одно анимационное окно и несколько графических окон. В графических окнах будем строить графики следующих переменных: активного момента и момента сил сопротивления, угловой скорости вала, а также тока и напряжения на двигателе.



#### Подготовка моделирования

1. Для создания нового анимационного окна выберите пункт меню **Инструменты | Анимационное окно...**
2. При помощи пункта меню **Инструменты | Графическое окно...** создайте три новых графических окна. Разместите окна так, как показано на рисунке ниже.



3. Откройте окно **Мастера переменных** (пункт меню **Инструменты | Переменные | Мастер переменных...**).

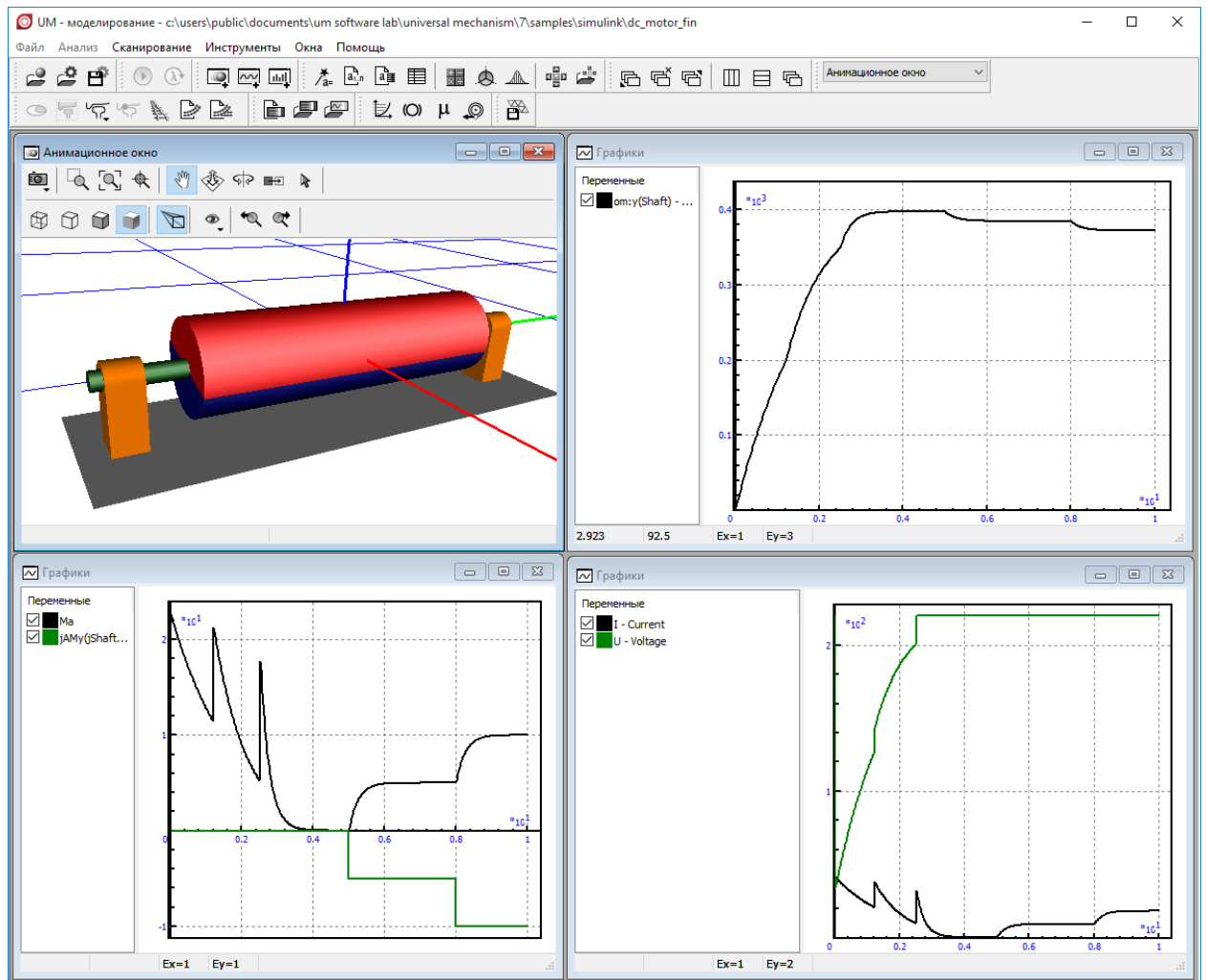
Выведем на график значение электромагнитного момента, действующего на вал со стороны электродвигателя, а также момент сил сопротивления.

4. В **Мастере переменных** перейдите на вкладку **Идентификаторы**. В списке идентификатором модели слева выберите **Ma**. Создайте переменную кнопкой . Переменная **Ma** появится в контейнере переменных снизу.
5. Перейдите на вкладку **Шарнирные силы**. В поле **Исследуемая величина** выберите **Момент**, в поле **Компонента** выберите **Y**. В поле **Действует на** выберите тело **2: Shaft**. Создайте переменную кнопкой .
6. Контейнер переменных **Мастера переменных** сейчас содержит две переменные: активный момент и момент сил сопротивления. Выделите эти переменные и мышкой перетащите их в левое нижнее графическое окно.
7. Создайте и перетащите в правое верхнее окно переменную – угловую скорость вала: вкладка **Угловые переменные**, **Исследуемая величина = Угл. скорость**, **Компонента = Y**.
8. Создайте и перетащите в правое нижнее окно две переменные – ток и напряжение на двигателе: вкладка **Идентификаторы**, идентификаторы **I** и **U**.
9. Закройте **Мастер переменных**.

### Моделирование динамики системы

1. Выберите пункт меню **Анализ | Моделирование...** Откроется **Мастер моделирования объекта**.
2. Перейдите на вкладку **Интегратор | Параметры моделирования** и проверьте установки. В поле **Численный метод** должен быть выбран **BDF** метод, в разделе установки условий завершения установите **Время > 10 с**, **Шаг представления результатов – 0.02**, **Погрешность – 0.001**.
3. Нажмите кнопку **Интегрирование**.
4. В процессе расчета будут построены графики, см. рис. ниже.

Проанализируем полученные результаты. На графике электромагнитного момента и тока четко видны скачки, соответствующие выключению ступеней пускового реостата. До номинальной угловой скорости вал разгоняется примерно за 4 с, после чего электромагнитный момент становится равным нулю. Далее, в момент времени 5 секунд происходит ступенчатое увеличение момента сил сопротивления. Вслед за внешним моментом сил сопротивления электромагнитный момент примерно за одну секунду выходит на новое установившееся значение. Увеличение нагрузки отрабатывается увеличением электромагнитного момента, тока якоря и снижением угловой скорости вала.



## 3.2. Моделирование с использованием Matlab CoSimulation

В этой части руководства будут рассмотрены примеры моделей систем управления с использованием инструмента **Matlab CoSimulation** из модуля **UM Control**. Это подразумевает, что модель механической части будет экспортирована из UM и импортирована в Matlab/Simulink, где будет представляться в системе управления в виде одного блока, имеющего некоторое количество входов и выходов. Остановимся на основных моментах, поясняющих суть предлагаемой технологии моделирования.

### 3.2.1. Порядок работы

В общем случае моделирования динамики механических систем с использованием Matlab **CoSimulation** предполагает выполнение следующих этапов.

- Описание модели системы управления в среде Matlab/Simulink.
- Включение в эту модель блока S-функции, который представляет собой модель механической системы.
- Создание модели механической системы в программе **UM Input**.
- Загрузка подготовленной модели механической части в программу **UM Simulation**. Формирование файлов настроек и управляющего m-файла с помощью **Мастера экспорта** для подключения этой модели в систему управления Matlab/Simulink.
- Создание связи между UM-моделью и системой управления **Matlab/Simulink**.
- Моделирование динамики управляемого движения в среде **Matlab/Simulink**.

Модель механической системы UM, подключаемая в систему управления Matlab/Simulink, рассматривается как черный ящик, который по некоторому закону преобразует входные величины в выходные. При включении механической системы в модель системы управления на выходы механической системы назначаются *переменные*, которые создаются при помощи **Мастера переменных**. Входные величины и параметры S-функции связываются с параметрами UM-модели.

Для реализации управляющих усилий со стороны системы управления в модель механической системы вводятся силы/моменты, значения или характеристики (например, коэффициент жесткости, коэффициент диссипации) которых задаются параметрами модели. Затем с помощью **Мастера экспорта** эти параметры связываются с величинами, которые являются входами S-функции.

### 3.2.2. Перевернутый маятник

Модель перевернутого маятника с готовыми файлами настроек и управляющим m-файлом находится в каталоге [{Данные УМ}\SAMPLES\Cosimulation\inv\\_pend\\_cosim](#). Перед началом урока, проверьте ее наличие в указанном каталоге. Если такой модели нет, то скачайте ее из интернета по адресу:

[www.universalmechanism.com/download/90/inv\\_pend\\_cosim.zip](http://www.universalmechanism.com/download/90/inv_pend_cosim.zip).

В данном уроке мы не будем подробно разбирать этапы создания модели механической части, ограничимся только описанием связи механической части с системой управления Matlab/Simulink.

### 3.2.2.1. Подготовка системы управления в Matlab/Simulink

Модель системы управления Matlab/Simulink для инструмента Matlab CoSimulation, см. рис. 3.11, очень похожа на ту модель, которая рассматривалась ранее применительно к инструменту **Matlab Import**, рис. 3.1, но имеет одно существенное отличие: модель механической системы представляет собой отдельный блок – S-функцию. Этот компонент можно найти в библиотеке Simulink **User-Defined Functions**. При первом добавлении S-функции в модель системы управления она будет иметь название **System**.

Именно S-функция обеспечивает связь модели системы управления с моделью ПК «Универсальный механизм». В нашей модели для организации управления перевернутым маятником входной переменной для S-функции будет величина силы, подаваемой на тележку, а выходной величиной S-функции будет угол отклонения маятника от вертикали.

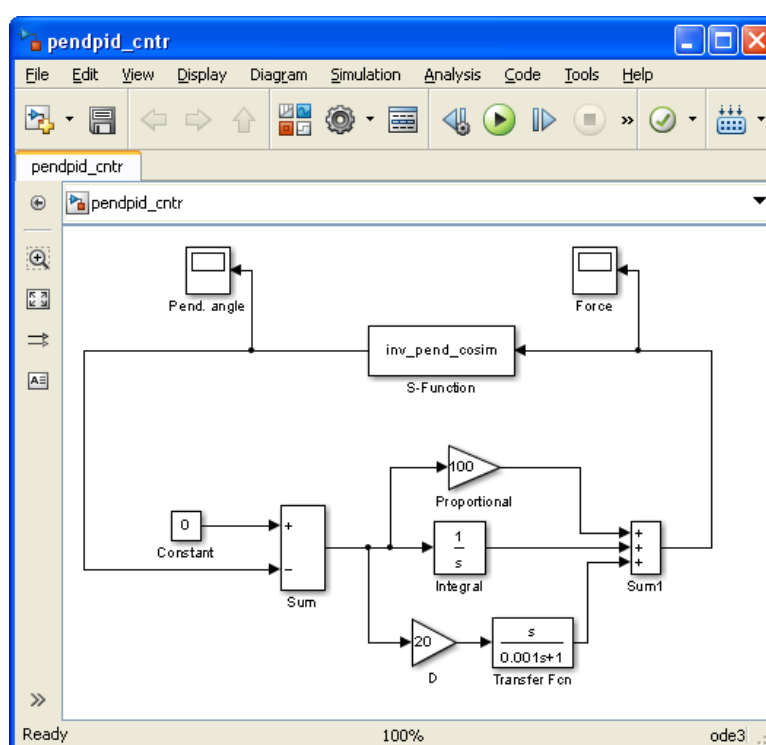


Рис. 3.11. Система стабилизации перевернутого маятника с механической системой как S-функцией

Подготовленную модель системы управления перевернутым маятником в Matlab/Simulink вы можете найти в файле [{Данные УМ}\SAMPLES\Cosimulation\inv\\_pend\\_cosim\pendpid\\_cntr.mdl](#), рис. 3.11. К модели в Matlab/Simulink мы обратимся в п. 3.2.2.3, а пока рассмотрим особенности экспорта механической части из UM.

### 3.2.2.2. Экспорт модели механической системы из UM

На следующем шаге нам необходимо указать, какие именно входные и выходные величины S-функции потребуются для моделирования, сохранить связанные с этим настройки и сгенерировать специальный m-файл, который будет управлять S-функцией.

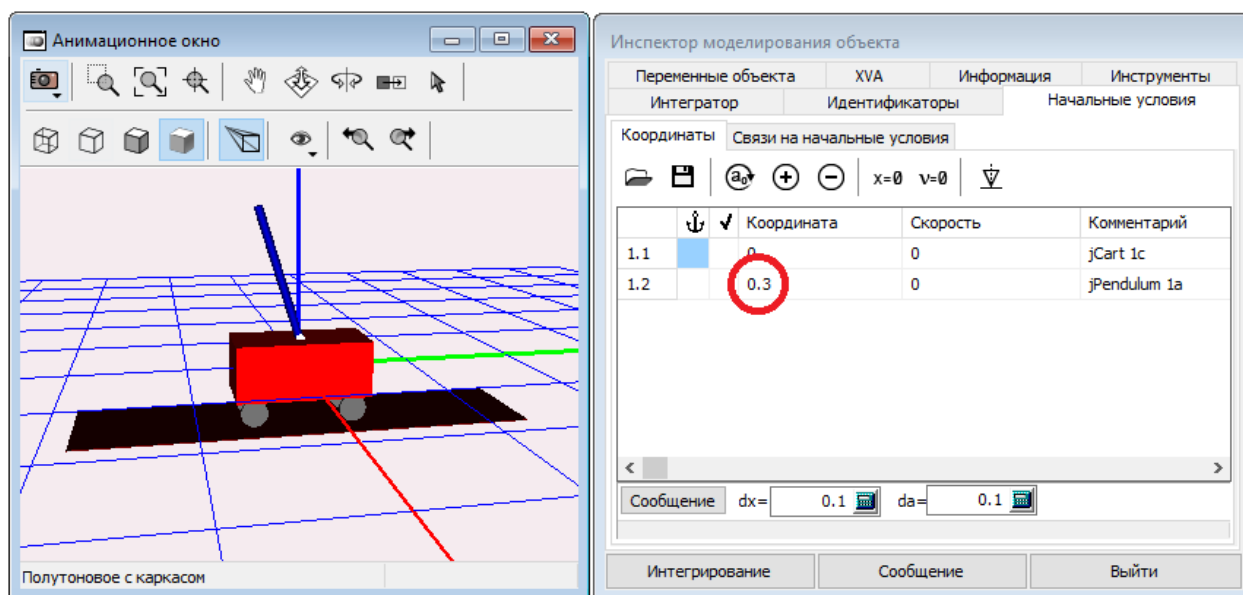
#### Загрузка модели механической части

1. Запустите программу моделирования **UM Simulation**.
2. Загрузите модель [{Данные UM}\SAMPLES\Cosimulation\inv\\_pend\\_cosim](#).

#### Задание начальных условий

Перед выполнением дальнейших действий убедитесь в том, что перевернутый маятник отклонен от вертикали и при необходимости задайте это отклонение.

1. Выберите пункт меню **Анализ | Моделирование...**
2. В появившемся окне **Инспектора моделирования объекта** перейдите на вкладку **Начальные условия | Координаты** и убедитесь, что координата **1.2** имеет значение **0.3** (рад), а маятник в анимационном окне отклонен от вертикали.



#### Мастер экспорта в Matlab/Simulink

3. Выберите пункт меню **Инструменты | Мастер экспорта**.

Появится окно **Мастера экспорта в Matlab/Simulink**.

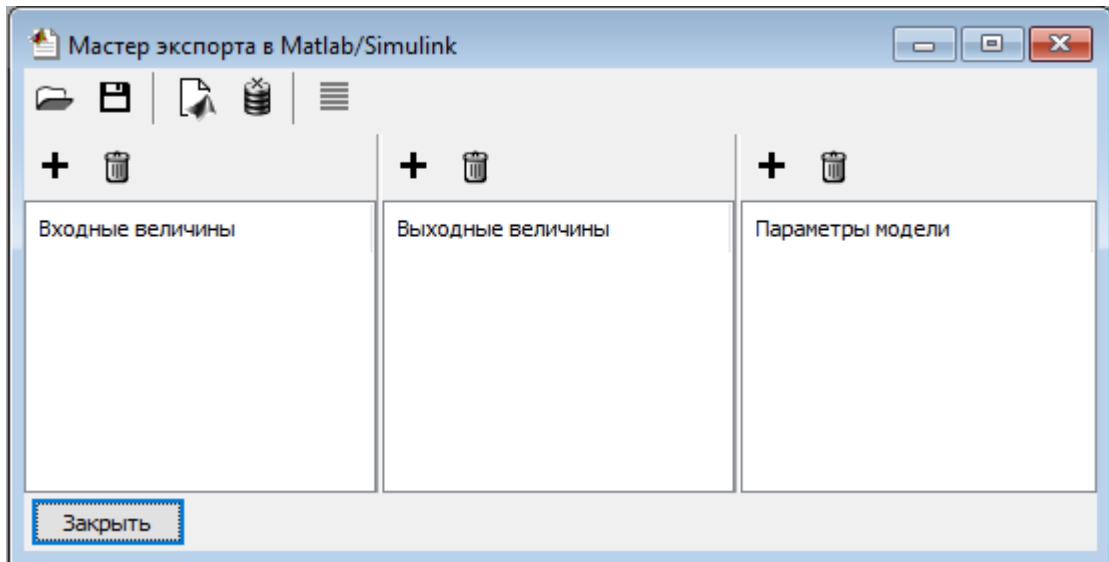


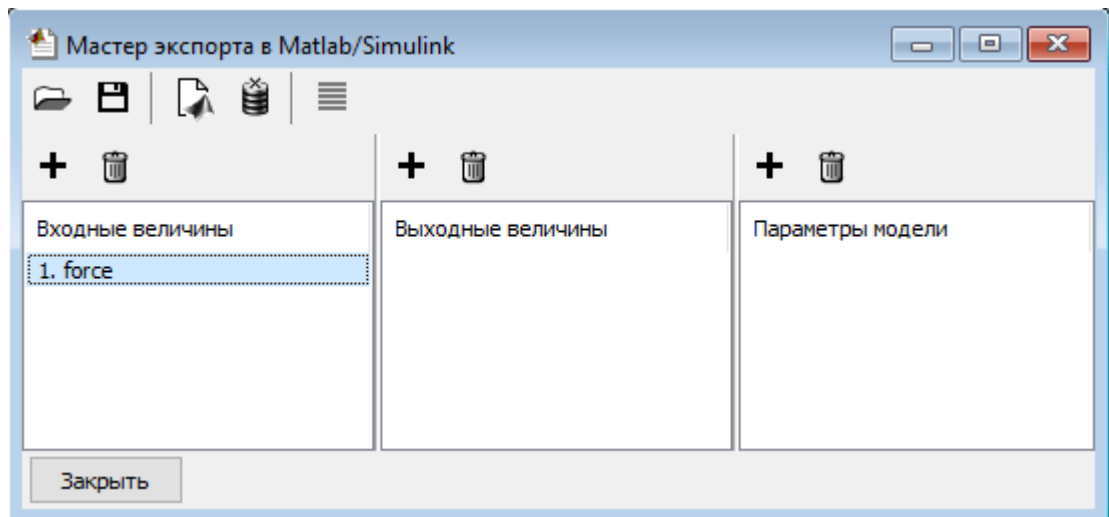
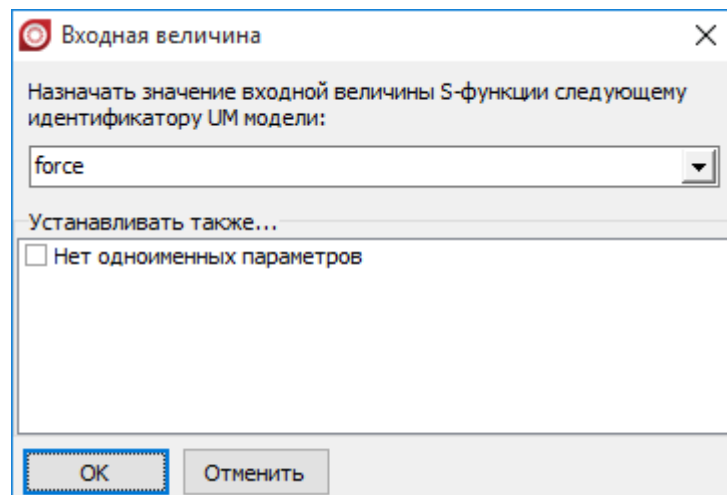
Рис. 3.12. Мастер экспорта в Matlab/Simulink

В данном случае нам необходимо назначить одну входную и одну выходную величины. На вход модели перевернутого маятника мы подадим величину усилия на тележку, а в качестве выходной величины назначим угол отклонения маятника от вертикали.

## Назначение входных величин

Для того чтобы добавить новую входную величину для UM-модели необходимо выполнить следующие действия.

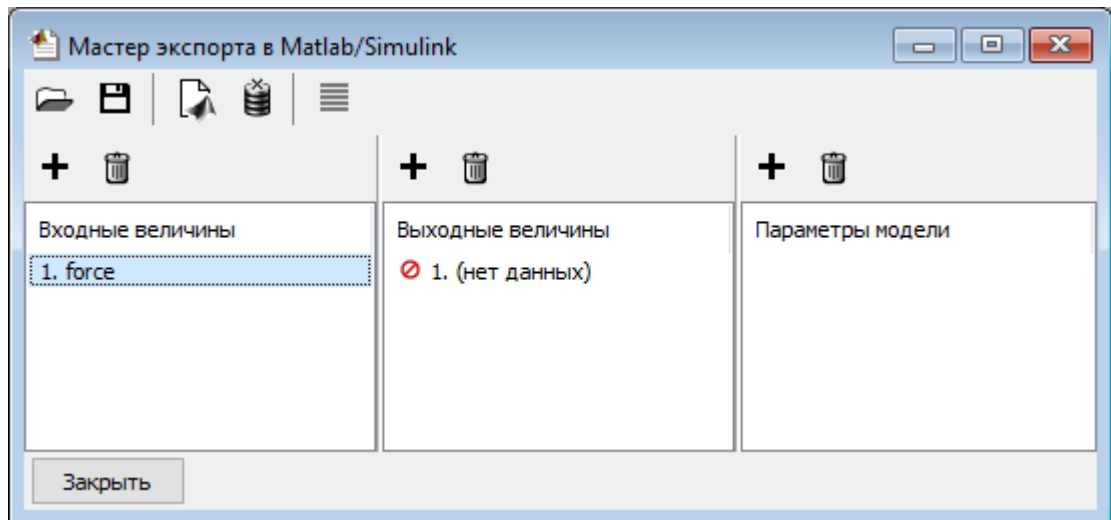
1. Нажмите кнопку **+** над списком **Входные величины**. Появится диалоговое окно назначения входных величин S-функции идентификаторам UM-модели.
2. В поле **Назначать значение входной величины...** выберите идентификатор модели **force**.
3. Закройте окно кнопкой **ОК**.




### Назначение выходных величин для модели перевернутого маятника

С помощью **Мастера переменных** сформируем переменную **Угол отклонения маятника от вертикали** и назначим ее в качестве выходной величины S-функции.

1. Нажмите кнопку **+** над списком **Выходные величины**. В списке выходных величин появится одна новая величина.



2. Запустите **Мастер переменных** (пункт меню **Инструменты | Переменные | Мастер переменных...**).
3. Перейдите на вкладку **Угловые переменные**, см. рис. 3.13.
4. В списке тел слева выберите **Pendulum**, выключите флажок **Использовать ориентацию при нулевых коорд.(инатах)**, в поле **Исследуемая величина** выберите **Вект.(ор) поворота**, а в поле **Компонента** выберите **X**.
5. Создайте переменную кнопкой . Переменная **ang:x(Pendulum)** появится в контейнере переменных.

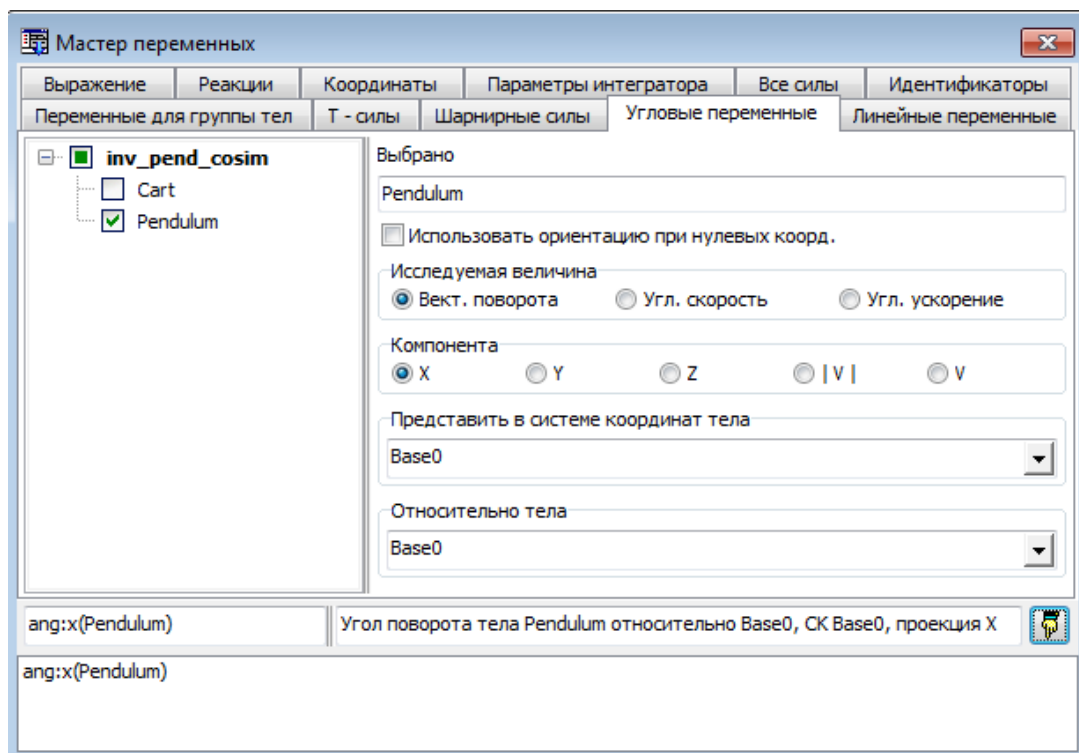
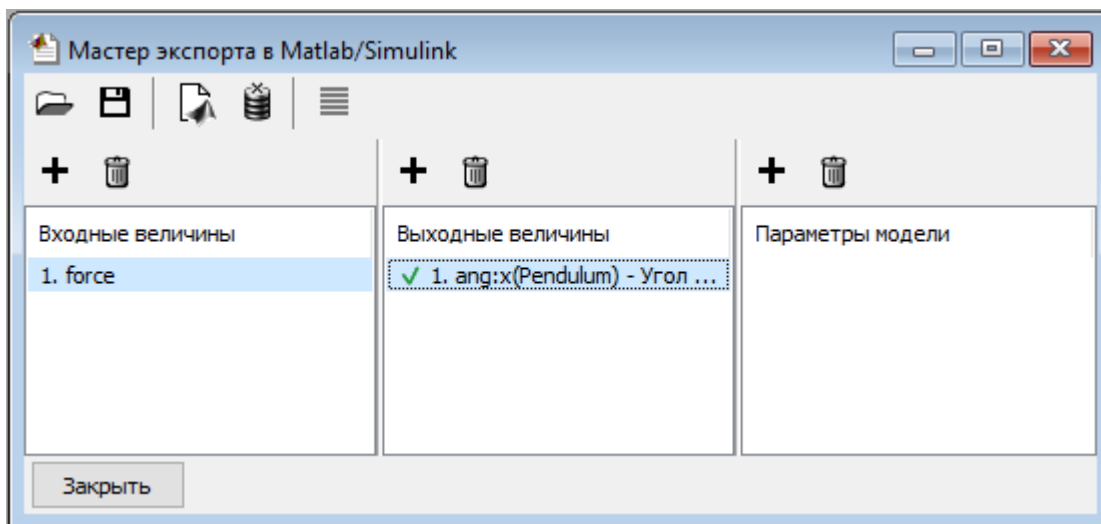


Рис. 3.13. Создание переменной

6. Перетащите мышкой созданную переменную **ang:x(Pendulum)** в окно **Мастера экспорта в Matlab/Simulink** и назначьте ее в качестве выходной величины.

Выход модели станет помечен зеленой галочкой, что означает, что на данный выход S-функции назначена переменная.



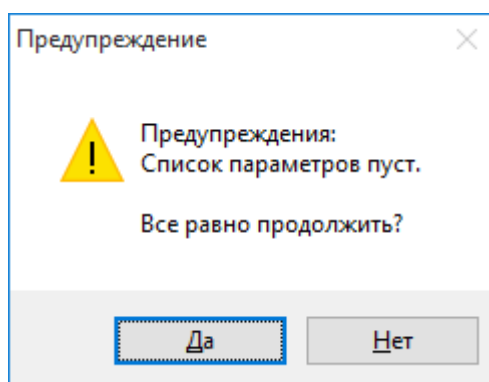
Все необходимые настройки для экспорта UM-модели в Matlab/Simulink закончены. Далее сгенерируем m-файл, который обеспечит работу UM-модели, как S-функции, одновременно сохраним файл настроек \*.cosim.

## Сохранение управляющего m-файла

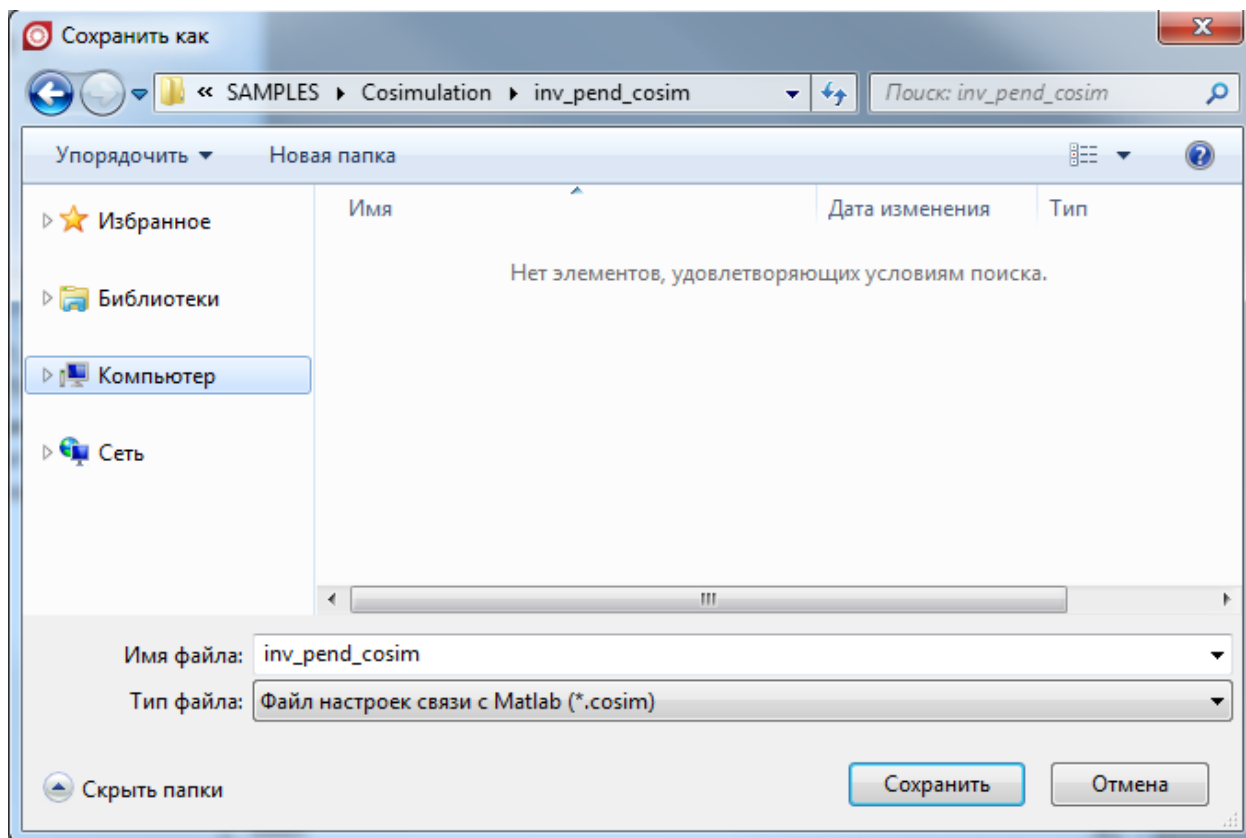
Теперь, когда мы описали количество входов и выходов, а также связали *идентификаторы* UM модели с входами, а *переменные* – с выходами, нужно сформировать m-файл, который необходим для связи UM-модели и системы управления Matlab/Simulink. Для этого выполните следующие действия.

1. Нажмите кнопку  для вызова диалога создания m-файла.

При этом появится предупреждение о том, что список параметров модели пуст. Это сообщение можно проигнорировать и продолжить работу, нажав на кнопку **Да**.



2. В появившемся окне диалога сохранения файла необходимо указать в названии файла **inv\_pend\_cosim** и сохранить его. Одновременно с созданием m-файла будет сохранен одноименный файл **inv\_pend\_cosim.cosim**, который будет использоваться «Универсальным механизмом» при работе под управлением Matlab/Simulink.

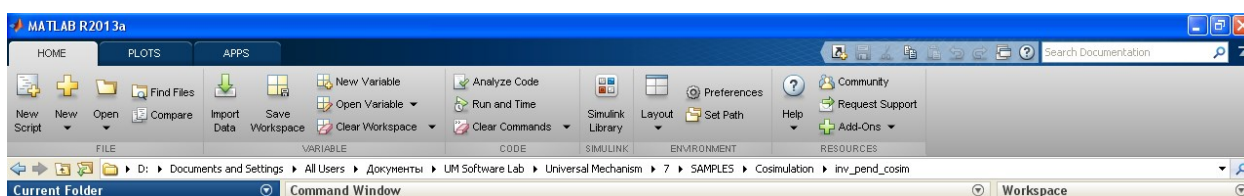


На этом действия на стороне UM закончены. Переходим к описанию связи на стороне Matlab/Simulink.

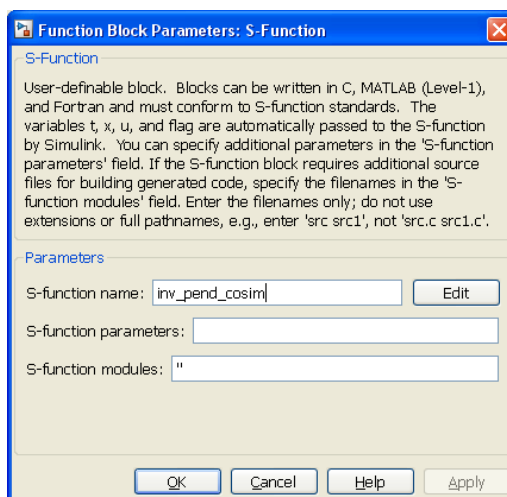
### 3.2.2.3. Создание связи между UM-моделью и Matlab/Simulink

Подготовленный на предыдущем этапе m-файл, который по умолчанию сохранен в каталоге UM-модели, в общем случае необходимо скопировать в каталог, где находится система управления Matlab/Simulink. В нашем случае файл системы управления (.mdl) также находится в каталоге модели, поэтому копировать m-файл в данном случае никуда не нужно. Для создания связи между системой управления Matlab/Simulink и механической моделью выполните следующие действия.

1. Запустите **Matlab**
2. В поле **Current Directory** выберите каталог **{Данные UM}\ SAMPLES\Cosimulation\inv\_pend\_cosim**.



3. Откройте файл **pendpid\_cntr.mdl**, в котором хранится модель системы управления перевернутым маятником.
4. Двойным щелчком мыши на **S-функции** вызовите окно параметров этого блока.
5. Убедитесь, что в поле **S-function name** установлено **inv\_pend\_cosim**, что соответствует названию ранее подготовленного m-файла.



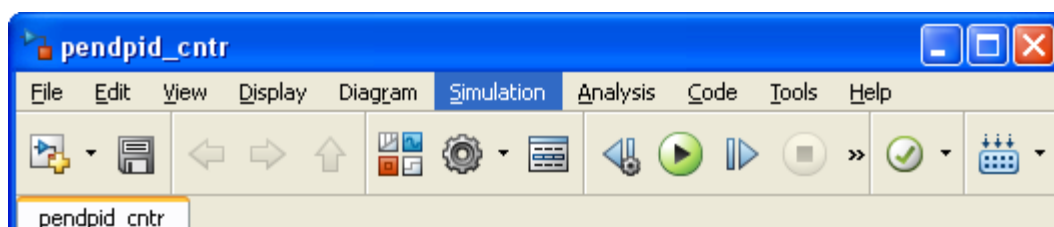
6. Нажмите **ОК**.

### 3.2.2.4. Моделирование движения

Модель подготовлена для дальнейшего анализа. Сейчас мы установим параметры моделирования, а затем запустим модель на выполнение.

#### Установка параметров моделирования

Выберите пункт меню **Simulation | Model Configuration Parameters**.



Перейдите на вкладку **Solver**.

В поле **Start time** установите **0.0**, в поле **Stop time** – значение **0.3**, в поле **Type** выберите **Fixed-step**, в поле **Fixed-step size** установите **1E-4**, в поле **Solver** выберите метод **ode3(Bogacki-Shampine)**. Если вы все сделали верно, то ваши настройки должны совпадать с теми, которые приведены на рис. 3.14 (этот пример приведен для версии Matlab R2013a).

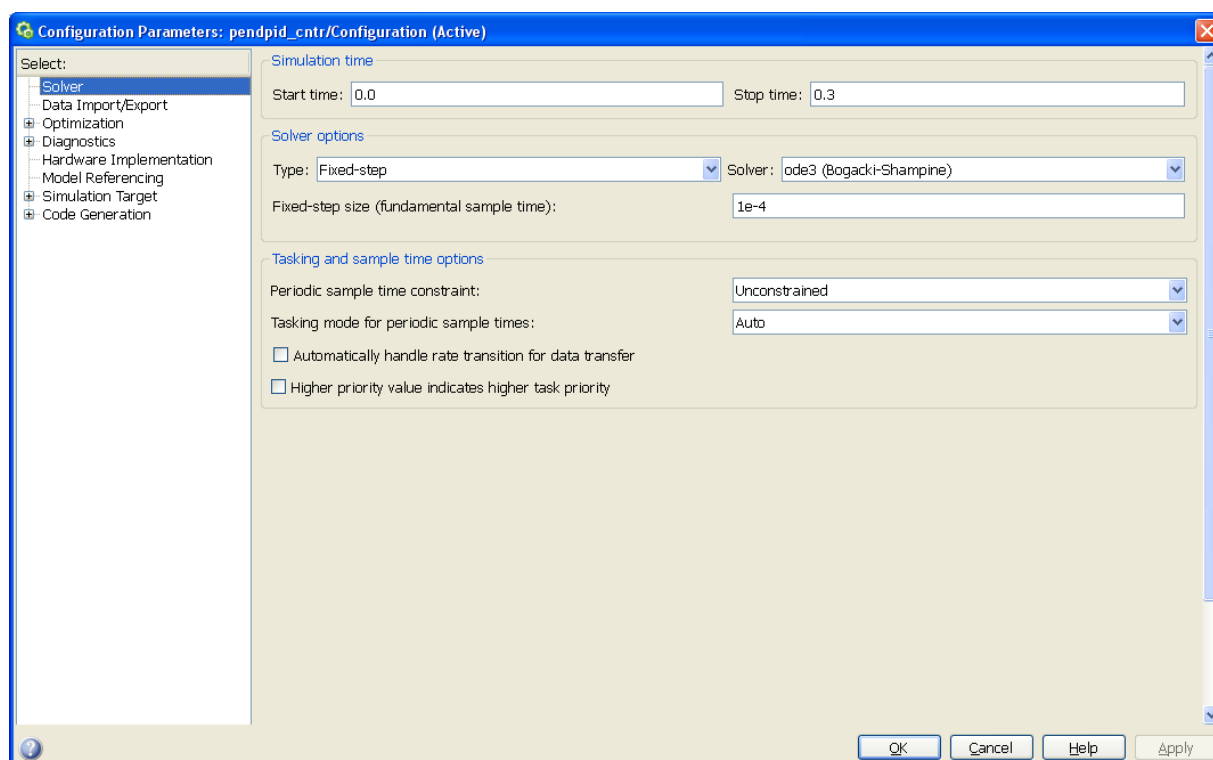



Рис. 3.14. Параметры конфигурации модели

Нажмите **ОК**.

## Моделирование динамики системы

1. Двойным щелчком мыши откройте окна **Scope**, в которых будут отображаться угол отклонения маятника от вертикали и величина усилия, подаваемого на тележку.
2. Кнопкой  запустите модель на выполнение.
3. По истечении времени моделирования вы увидите результаты, приведенные на рис. 3.15 и рис. 3.16.

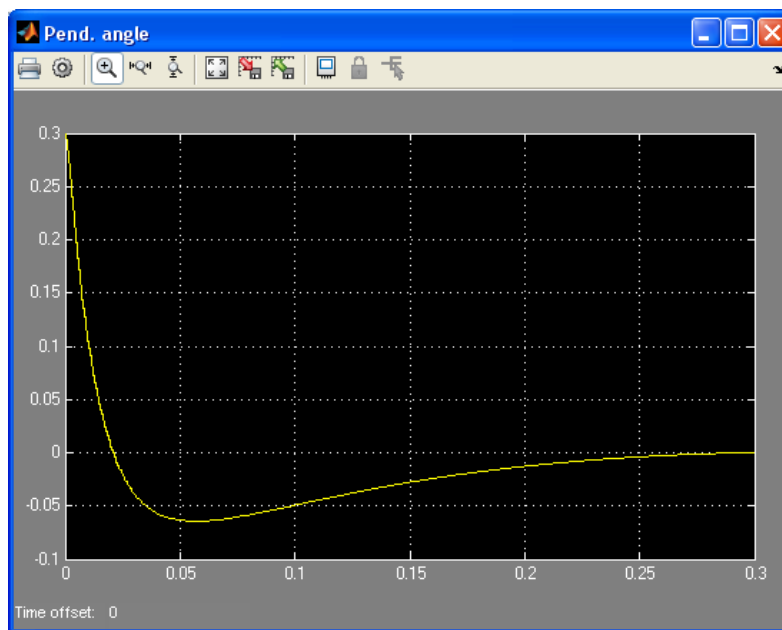


Рис. 3.15. График изменения угла отклонения маятника от вертикали

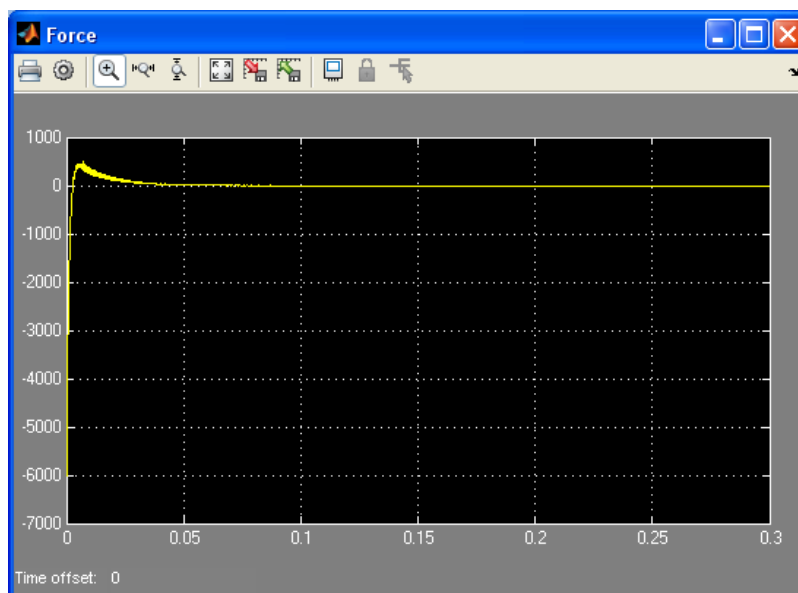


Рис. 3.16. График изменения величины усилия, подаваемого на тележку

### 3.2.3. Двигатель постоянного тока

UM и Matlab/Simulink модели двигателя постоянного тока с готовыми файлами настроек и управляющим m-файлом находится в каталоге [{Данные UM}\SAMPLES\Cosimulation\dcmotor\\_cosim](#). Перед началом урока, проверьте наличие этого каталога. Если такого каталога нет, то скачайте его из интернета по адресу: [www.universalmechanism.com/download/90/dcmotor\\_cosim.zip](http://www.universalmechanism.com/download/90/dcmotor_cosim.zip).

В данном уроке мы не будем подробно разбирать этапы создания модели механической части, ограничимся только описанием связи механической части с системой управления Matlab/Simulink.

### 3.2.3.1. Подготовка системы управления в Matlab/Simulink

Подготовленная модель системы управления двигателя постоянного тока в Matlab/Simulink представлена на рис. 3.17. Подсистема **DC motor** показана на рис. 3.10.

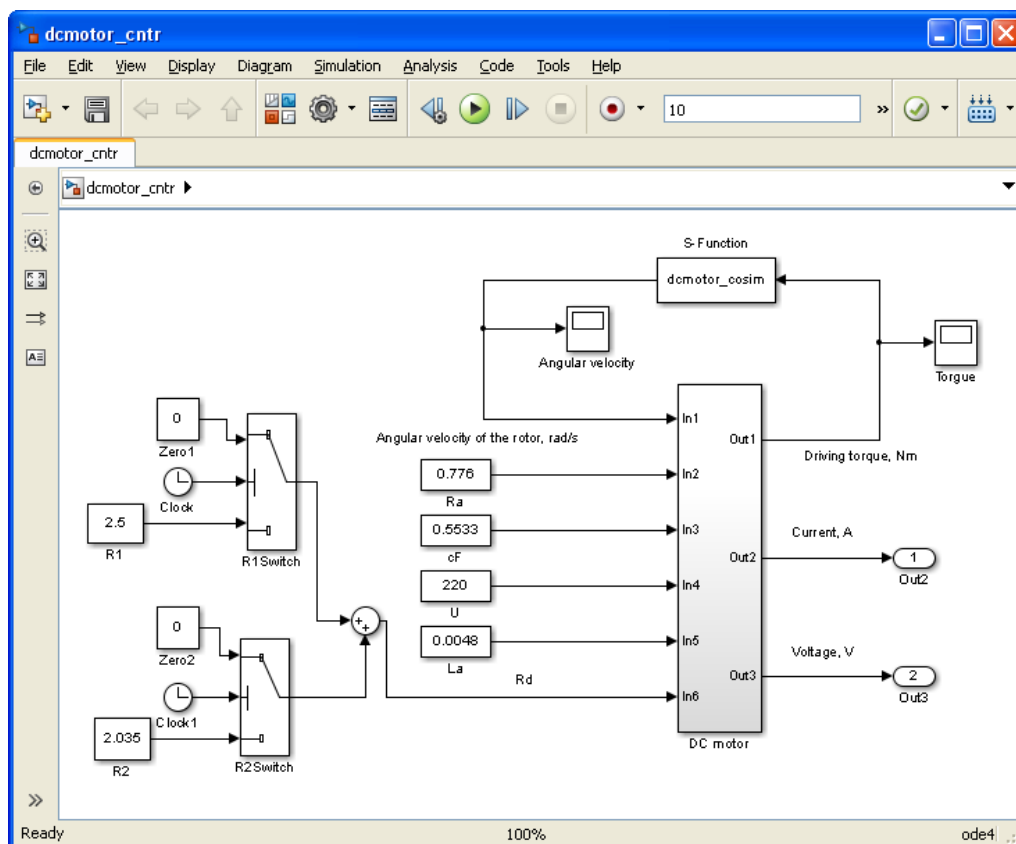


Рис. 3.17. Модель двигателя в Matlab/Simulink

В нашей модели для организации управления двигателем постоянным током входной переменной для S-функции будет величина вращательного момента, подаваемого на вал со стороны электродвигателя. В свою очередь, выходной величиной S-функции будет являться угловая скорость вала.

Подготовленную модель системы управления перевернутым маятником в Matlab/Simulink вы можете найти в файле [{Данные УМ}\SAMPLES\Cosimulation\dcmotor\\_cosim\dcmotor\\_cntr.mdl](#), рис. 3.17. К модели в Matlab/Simulink мы обратимся в п. 3.2.3.3, а пока рассмотрим особенности экспорта механической части из UM.

### 3.2.3.2. Экспорт модели механической системы из UM

#### Загрузка модели механической части

1. Запустите программу моделирования **UM Simulation**.
2. Загрузите модель [{Данные UM}\SAMPLES\cosimulation\dcmotor\\_cosim](#).

#### Начало экспорта в Matlab/Simulink

1. Выберите пункт меню **Инструменты | Мастер экспорта в Matlab/ Simulink...** Появится окно **Мастера экспорта**.

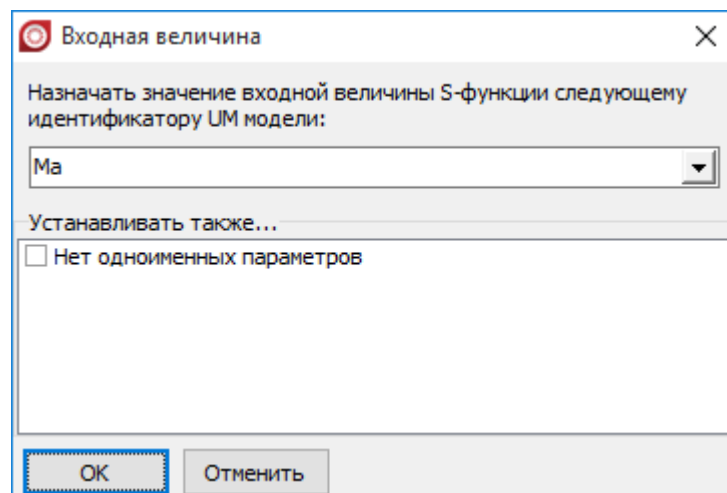
В данном случае нам необходимо назначить одну входную и одну выходную величины. На вход модели двигателя постоянного тока мы подадим величину вращательного момента, а в качестве выходной величины назначим угловую скорость вала.

#### Назначение входных величин

Добавим новую входную величину.


Нажмите кнопку **+** над списком входных величин. Появится диалоговое окно назначения входных величин S-функции идентификаторам UM-модели.

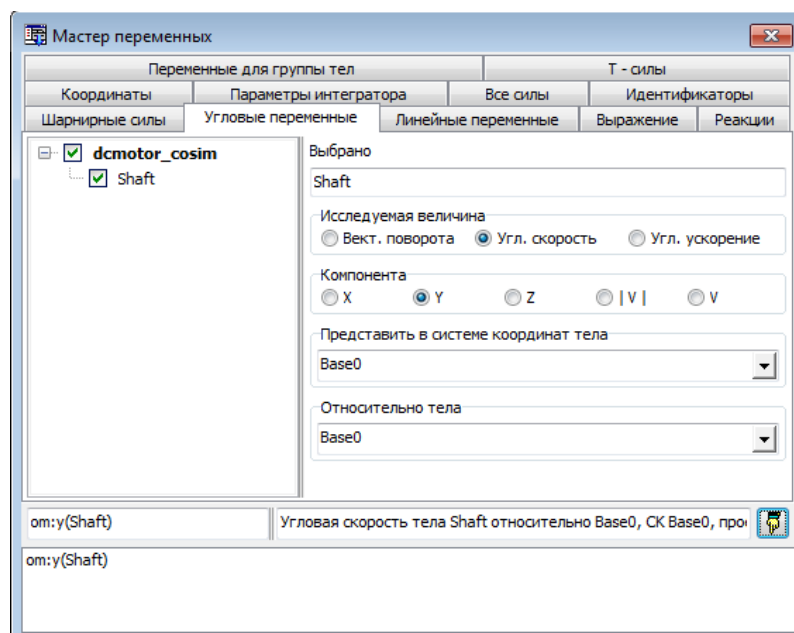
В поле **Назначать значение входной величины...** выберите **Ma**. Закройте окно кнопкой **ОК**.



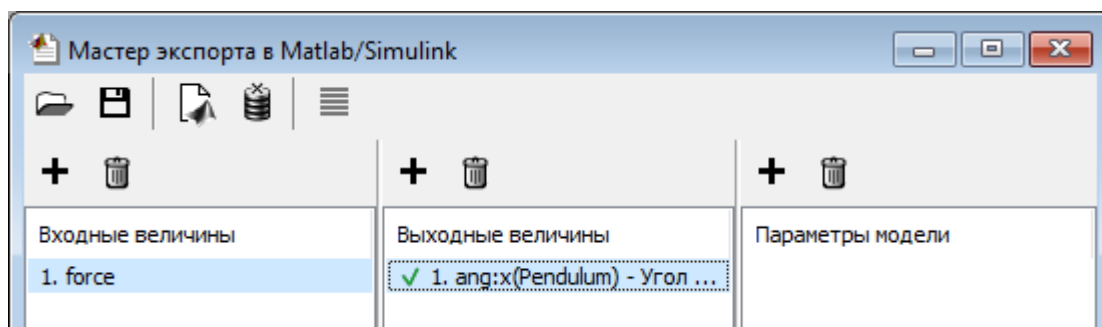
## Назначение выходных величин

С помощью **Мастера переменных** сформируем переменную «угловая скорость вала» и назначим ее в качестве выходной величины S-функции.


1. Создайте новую выходную величину нажав кнопку **+** над списком **Выходные величины**. В списке выходных величин появится одна новая величина.
2. Запустите **Мастер переменных** (пункт меню **Инструменты | Переменные | Мастер переменных...**).
3. Перейдите на вкладку **Угловые переменные**. В списке тел слева выберите **Shaft**, в поле **Исследуемая величина** выберите **Угл. скорость**, а в поле **Компонента** выберите **Y**. Создайте переменную кнопкой . Переменная **om:y(Shaft)** появится в контейнере переменных.

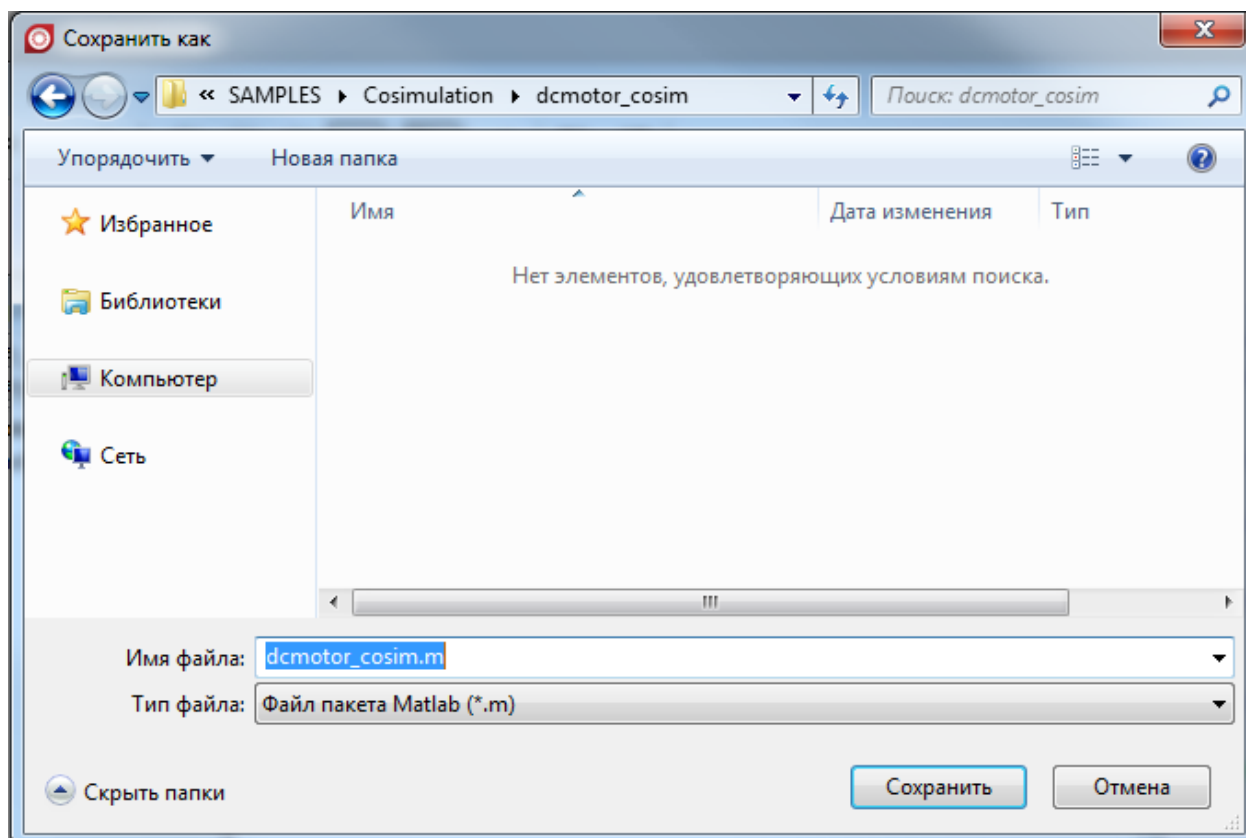


4. Перетащите мышкой созданную переменную **om:y(Shaft)** в окно **Мастера экспорта в Matlab/Simulink** и назначьте ее в качестве выходной величины. Выход модели станет помечен зеленой галочкой, что означает, что на данный выход S-функции назначена переменная.



## Сохранение управляющего m-файла

1. Нажмите кнопку  для вызова диалога сохранения m-файла. При этом появится предупреждение о том, что список параметров модели пуст. Это сообщение нужно проигнорировать и продолжить работу, нажав на кнопку **Да**.
2. В появившемся окне диалога сохранения файла необходимо указать в названии файла **dcmotor\_cosim** и сохранить его.

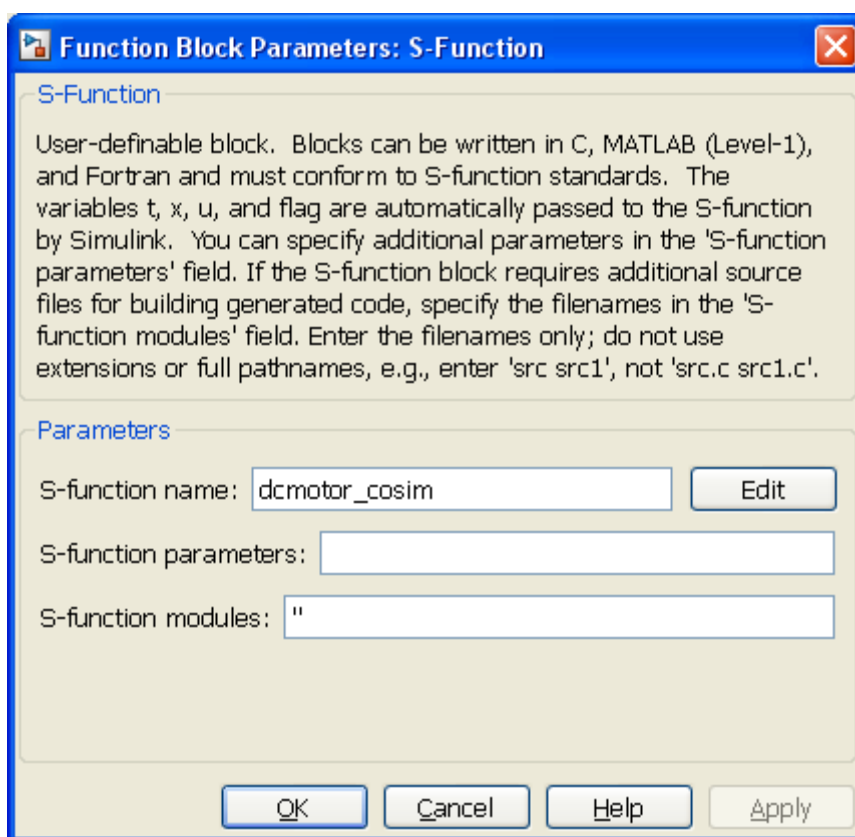


На этом действия на стороне UM закончены. Переходим к описанию связи на стороне Matlab/Simulink.

### 3.2.3.3. Создание связи между UM-моделью и Matlab/Simulink

Для создания связи между системой управления Matlab/Simulink и механической моделью выполните следующие действия.

1. Запустите **Matlab**.
2. В поле **Current Directory** выберите каталог **{Данные УМ}\ SAMPLES\Cosimulation\dcmotor\_cosim**.
3. Откройте файл **dcmotor\_cntr.mdl**, в котором хранится модель системы управления двигателем постоянного тока.
4. Двойным щелчком мыши вызовите окно параметров S-функции.
5. Убедитесь, что в поле **S-function name** установлено **dcmotor\_cosim**, что соответствует названию m-файла.



6. Нажмите кнопку **OK**.

Модель готова для дальнейшего анализа.

### 3.2.3.4. Моделирование движения

#### Моделирование динамики системы

1. Двойным щелчком мыши откройте окна **Scope**, в которых будут отображаться угловая скорость вала и величина вращательного момента.
2. Запустите модель на выполнение.
3. По истечении времени моделирования вы увидите следующие результаты, см. рис. ниже.

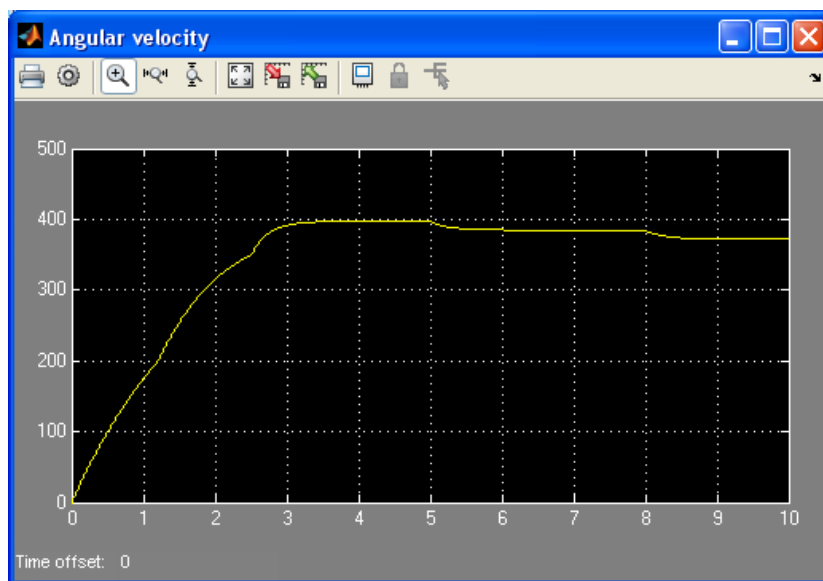


Рис. 3.18. Изменение угловой скорости вала

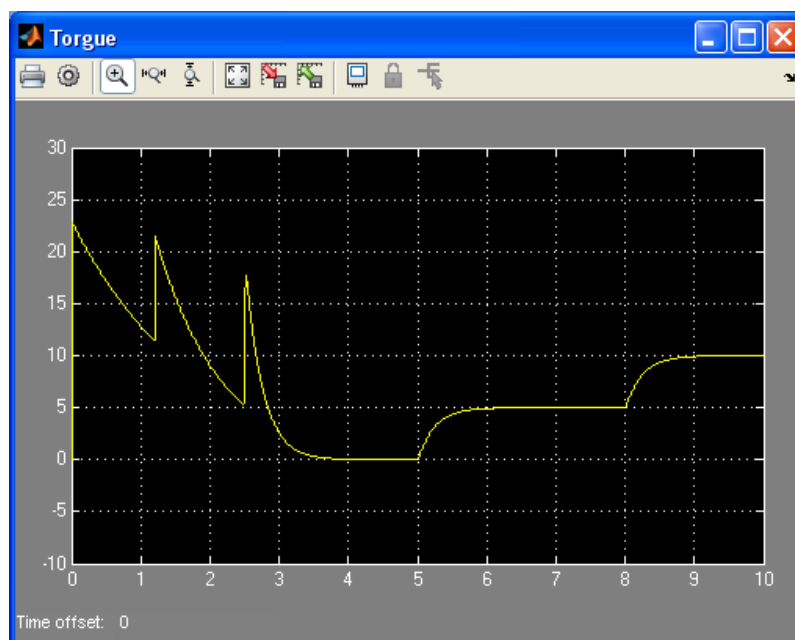


Рис. 3.19. Изменение активного электромагнитного момента на валу

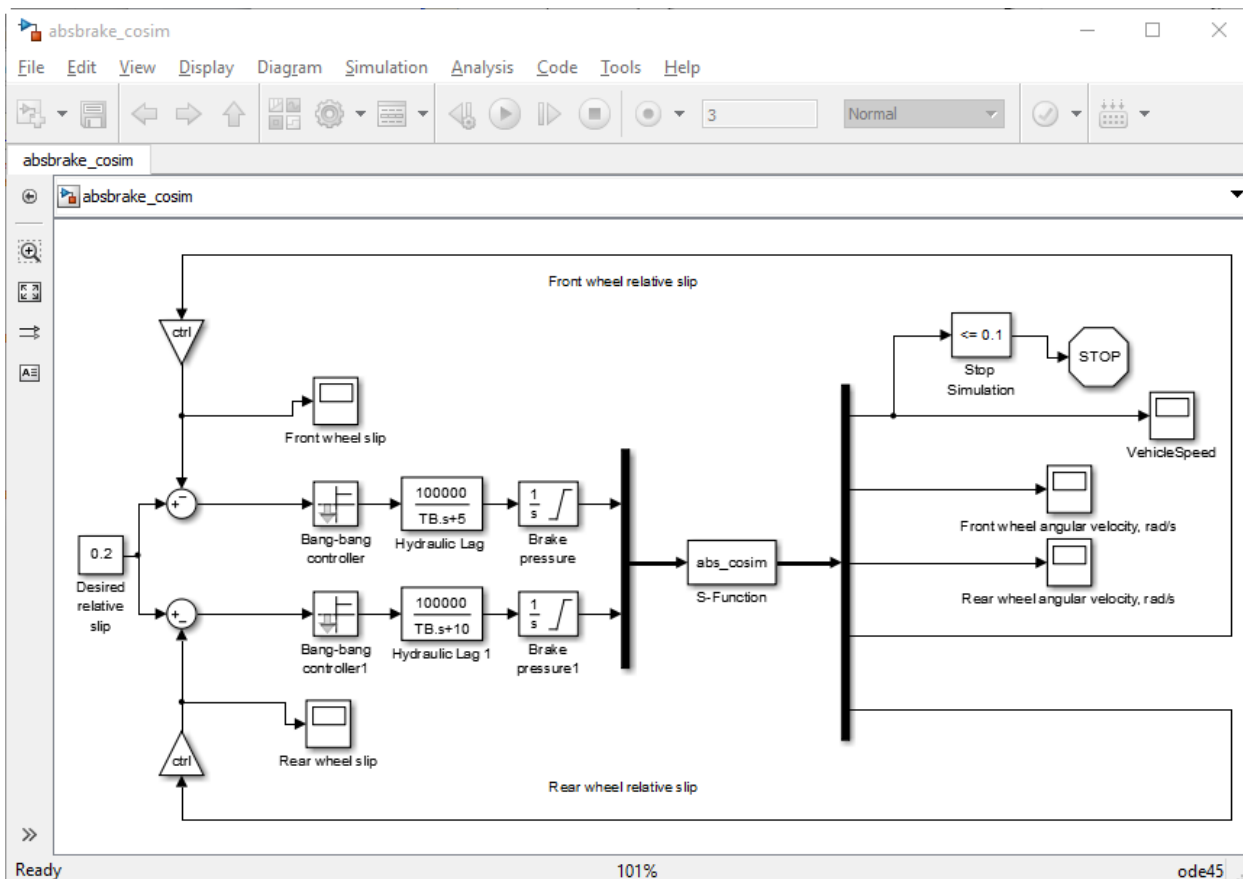
### 3.2.4. Антиблокировочная система автомобиля (ABS)

Модель автомобиля с готовыми файлами настроек, управляющим m-файлом и моделью Matlab/Simulink находится в каталоге [{Данные УМ}\SAMPLES\Cosimulation\vaz21\\_09abs](#). Перед началом урока, проверьте ее наличие в указанном каталоге. Если такой модели нет, то скачайте ее из интернета по адресу: [www.universalmecanism.com/download/90/vaz21\\_09abs.zip](http://www.universalmecanism.com/download/90/vaz21_09abs.zip).

В данном уроке мы не будем подробно разбирать этапы создания модели механической части, ограничимся только описанием связи механической части с системой управления Matlab/Simulink.

### 3.2.4.1. Подготовка системы управления в Matlab/Simulink

Электронный контроль системы ABS позволяет достигать гораздо лучшего торможения в аварийных ситуациях. Главной задачей системы ABS является предотвращение полной блокировки колес и тем самым обеспечение максимально возможного сцепления колес с дорогой и сохранения управляемости, в том числе на скользкой и мокрой дороге и гравии. За счет этого сохраняется управляемость автомобиля при любых состояниях дорожного покрытия. Простая модель ABS с поосным регулированием приведена на рисунке ниже.




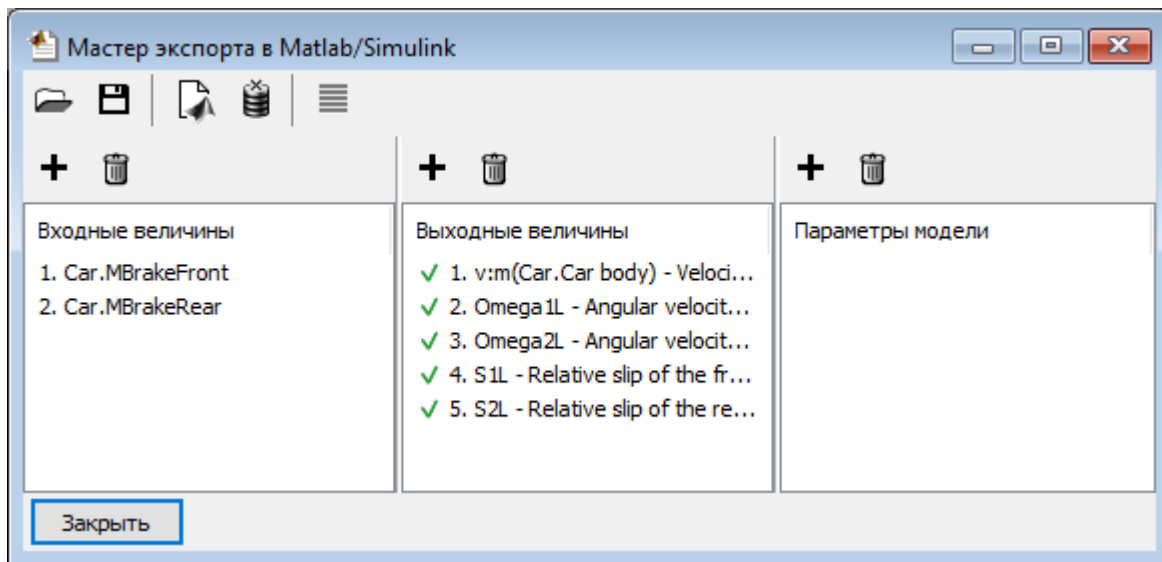
В нашей модели для организации управления автомобилем входными переменными для S-функции будут тормозные усилия, подаваемые на переднюю и заднюю оси автомобиля. В свою очередь, выходными величинами S-функции будут являться скорость движения автомобиля, угловые скорости передних и задних колес, пробуксовка на передних и задних колесах и давление, подаваемое на передние колеса.

### 3.2.4.2. Экспорт модели механической системы из UM

Не будем подробно описывать создание и сохранением настроек модели автомобиля, ограничимся лишь копиями экранов для иллюстрации проделанной работы.

Окно **Мастера экспорта в Matlab/Simulink** со списками входных и выходных величин представлено ниже. Для моделирования динамики в среде Matlab/Simulink необходимо заново сгенерировать управляющий m-файл.

1. Запустите **UM Simulation**.
2. Загрузите модель [{Данные УМ}\SAMPLES\Cosimulation\vaz21\\_09abs](#).
3. Откройте окно **Мастера экспорта** (меню **Инструменты | Мастер экспорта в Matlab/Simulink...**).
4. В **Мастере экспорта в Matlab/Simulink** загрузите файл **abs\_cosim.cosim**.
5. Ответьте **Да** на приглашение загрузить дополнительные файлы настроек.
6. Создайте заново m-файл щелкнув по кнопке .



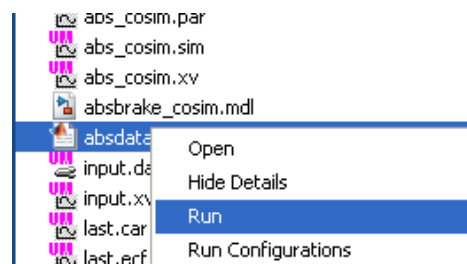
Сохранив управляющий m-файл, а вместе с ним и файлы, можно переходить к моделированию работы ABS.

### 3.2.4.3. Моделирование движения

Рассмотрим торможение автомобиля сначала с включенной, а затем с выключенной ABS. Ниже приведены зависимости скорости движения автомобиля, угловых скоростей передних и задних колес, пробуксовки передних и задних колес от времени.

#### 3.2.4.3.1. Загрузка модели

1. Запустите **Matlab**.
2. В поле **Current Directory** выберите каталог {Данные УМ} \ SAMPLES\ Cosimulation \ vaz21\_09abs.
3. Выполните инициализацию констант системы ABS. В списке файлов выделите **absdata.m** и в контекстном меню выберите команду **Run**.



4. Откройте файл **absbrake\_cosim.mdl**, в котором хранится модель системы управления антиблокировочной системой.
5. Запустите моделирование.

**3.2.4.3.2. Результаты моделирования движения с ABS**

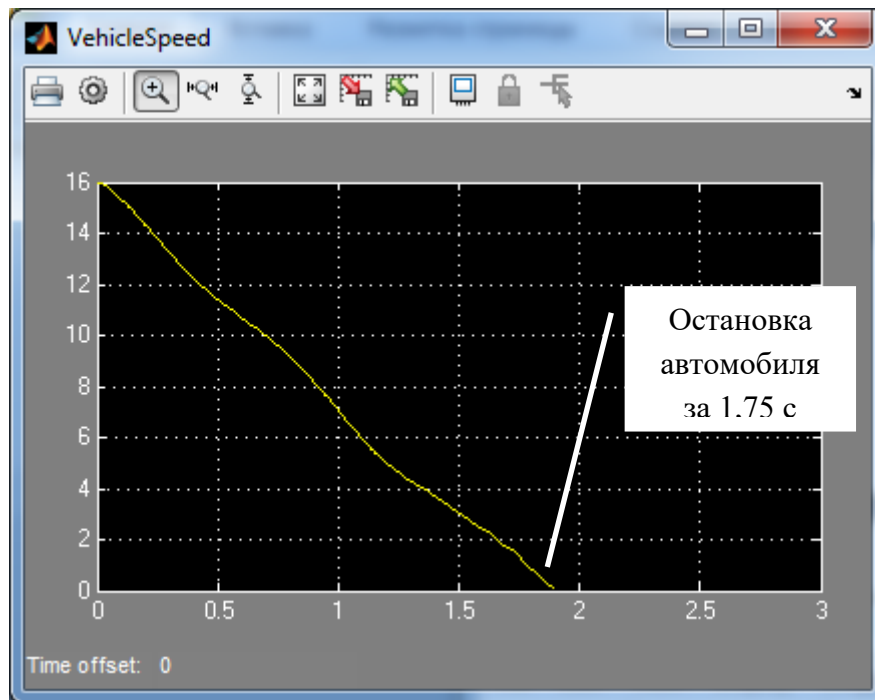


Рис. 3.20. Скорость автомобиля с включенной ABS

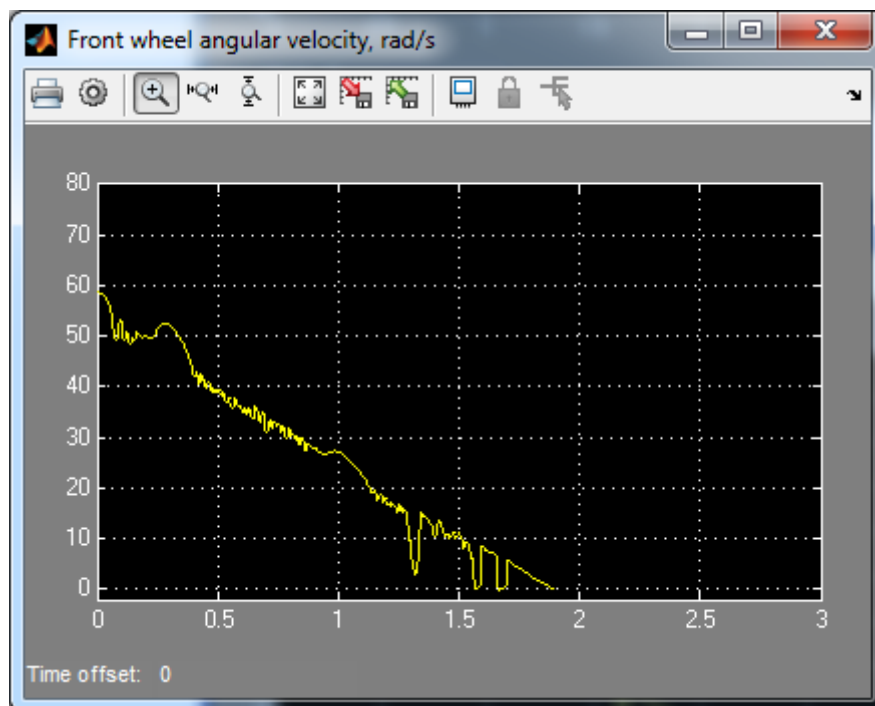


Рис. 3.21. Угловая скорость переднего колеса

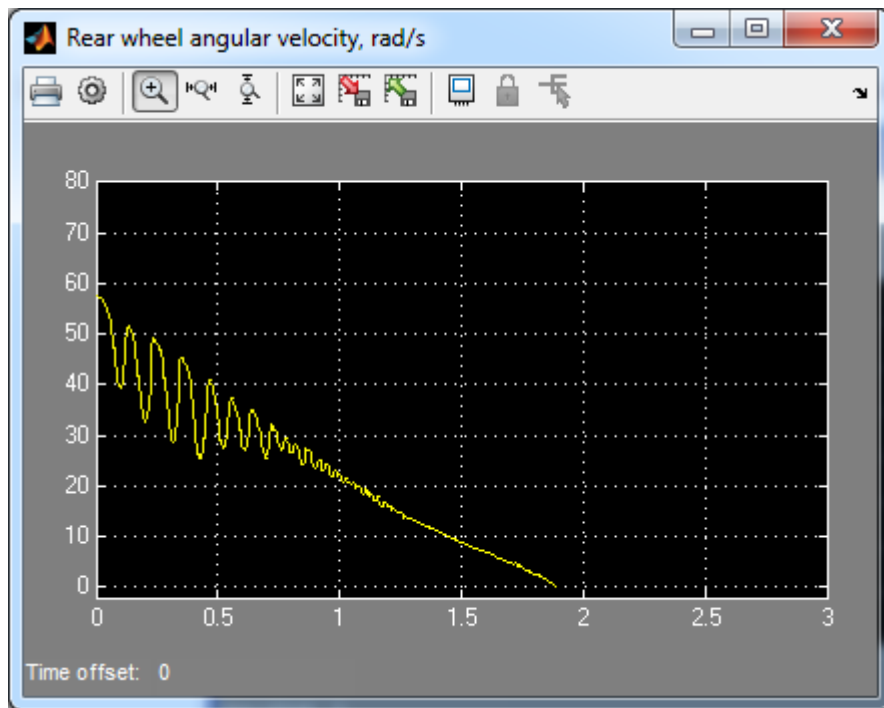


Рис. 3.22. Угловая скорость заднего колеса

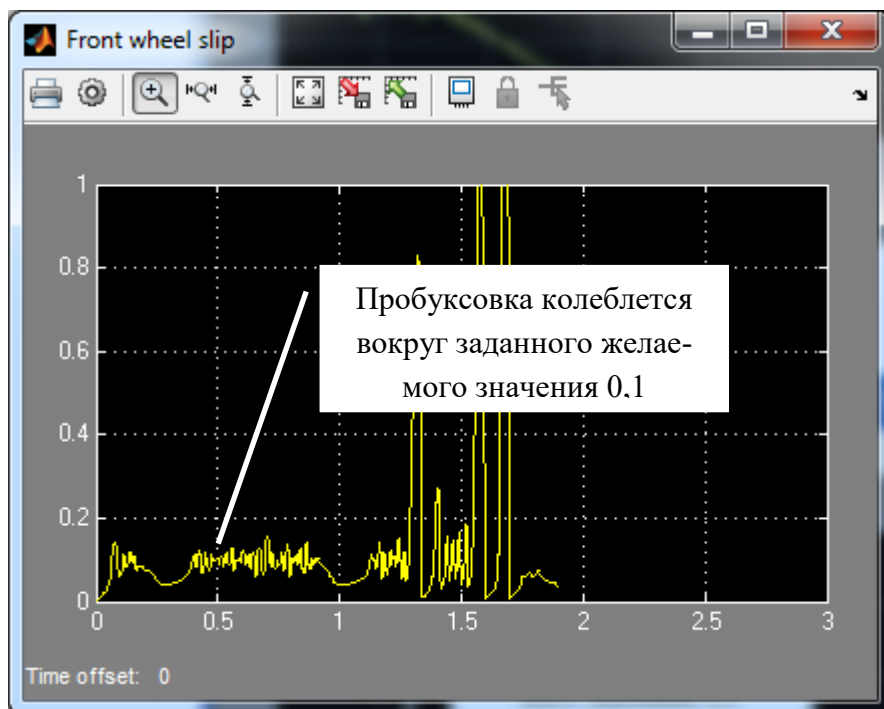


Рис. 3.23. Пробуксовка передних колес

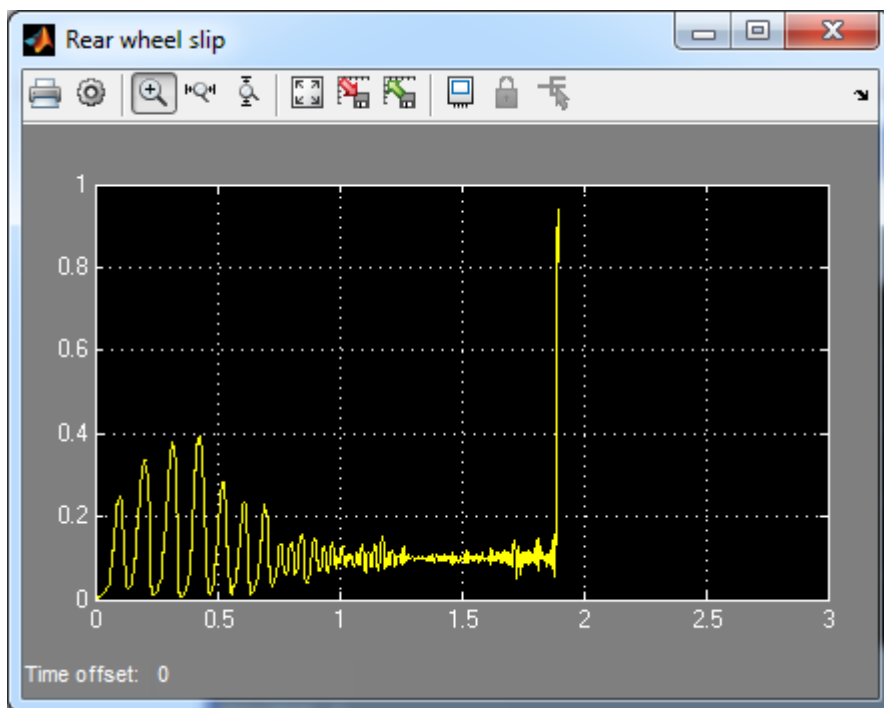


Рис. 3.24. Пробуксовка задних колес

### 3.2.4.3.3. Результаты моделирования движения без ABS

При отключенной системе ABS графики зависимости представленных выше величин от времени имеют следующий вид.

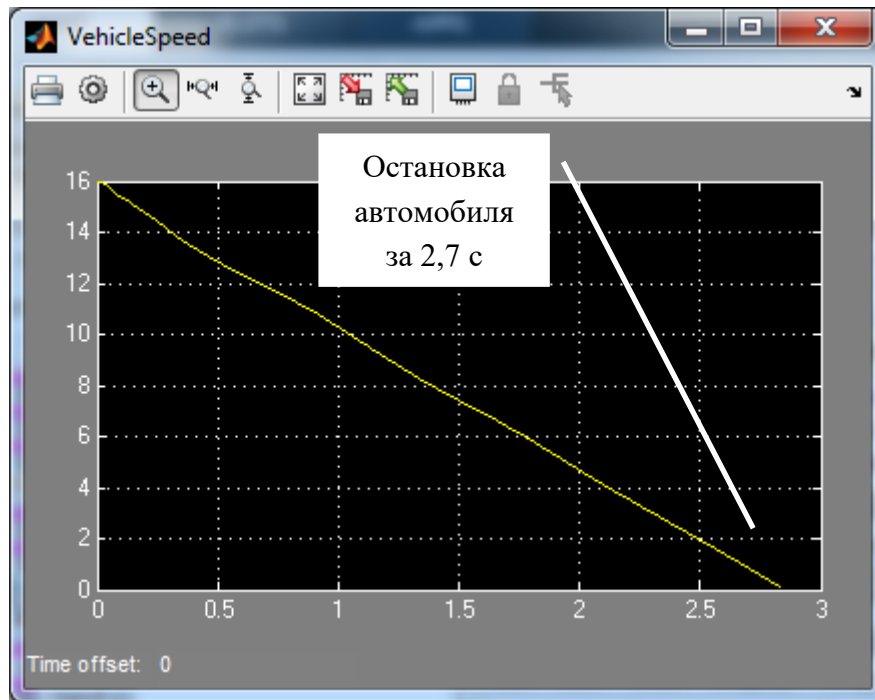


Рис. 3.25. Скорость автомобиля с выключенной ABS

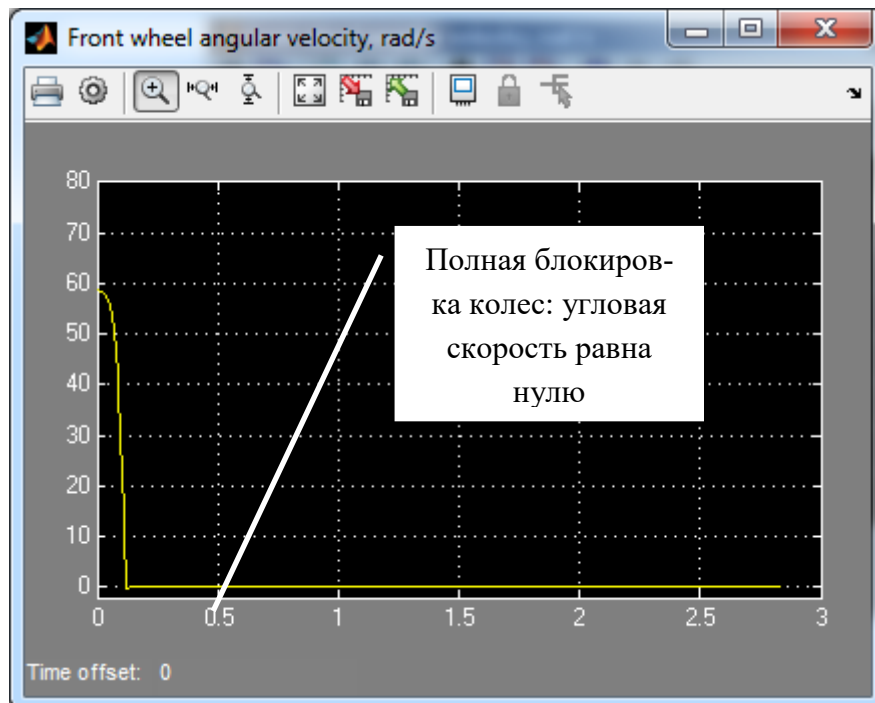


Рис. 3.26. Угловая скорость переднего колеса

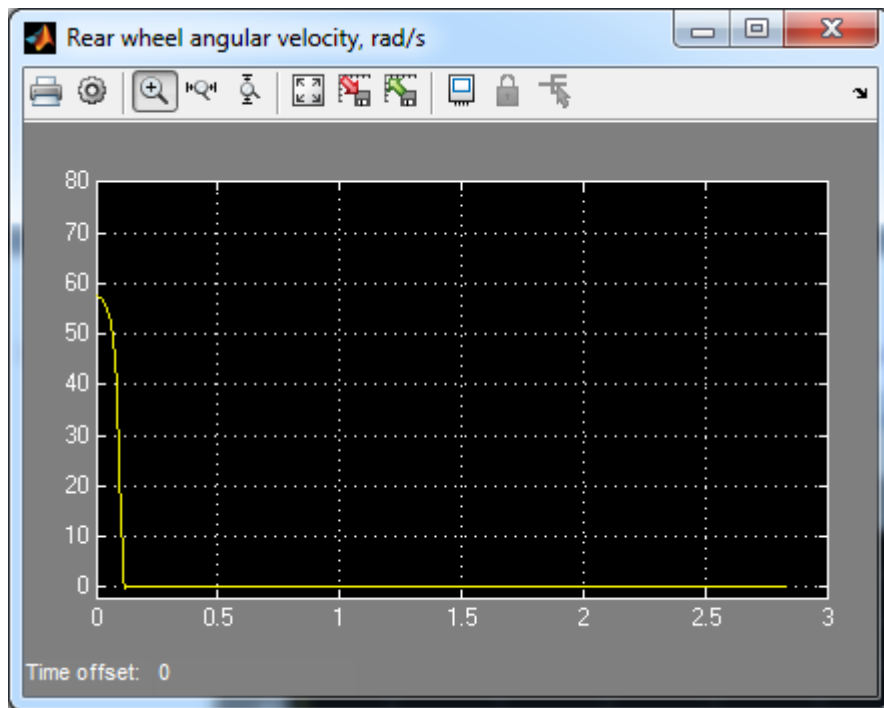


Рис. 3.27. Угловая скорость заднего колеса

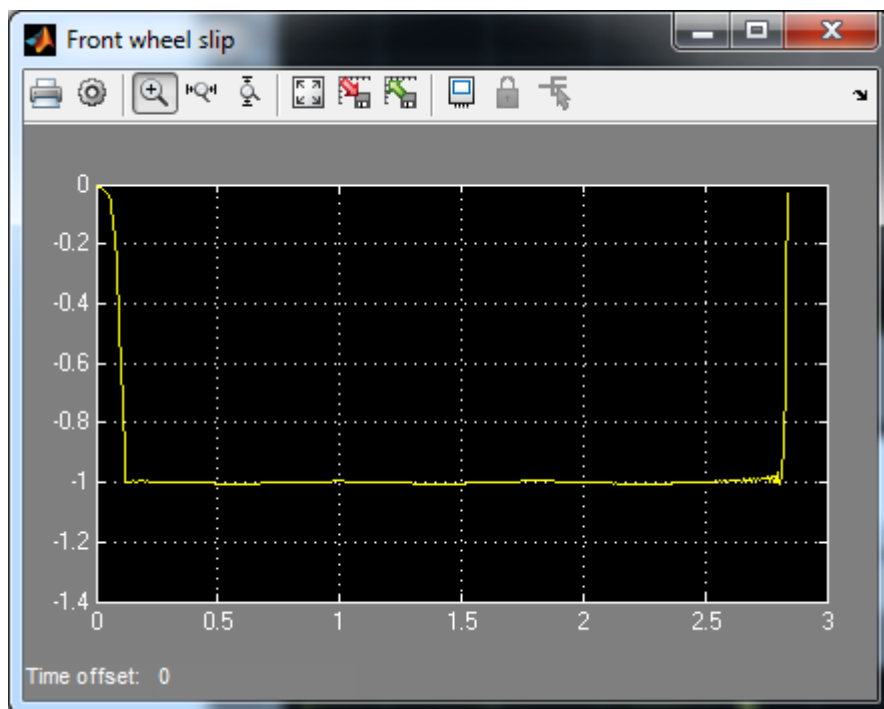


Рис. 3.28. Пробуксовка передних колес

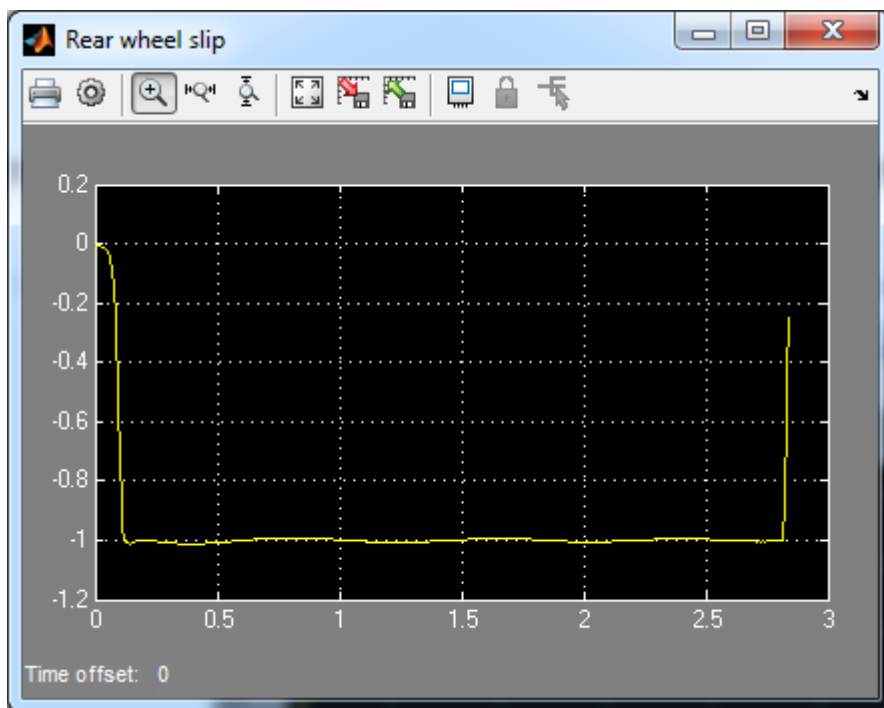


Рис. 3.29. Пробуксовка задних колес

При отключенной системе ABS остановка автомобиля происходит за больший промежуток времени: 2,7 с против 1,75 с. Из графиков зависимости угловых скоростей колес автомобиля от времени видно, что при отключенной ABS угловая скорость колес падает, они блокируются, и автомобиль начинает скользить по дороге с полностью заблокированными колесами, что увеличивает тормозной путь.

## 4. Работа с SimInTech

### 4.1. О SimInTech

SimInTech – это российская система модельно-ориентированного проектирования систем автоматического управления. SimInTech предназначен для детального исследования и анализа нестационарных процессов в различных объектах управления. Разработка математических моделей и алгоритмов управления в SimInTech происходит в виде структурного проектирования логико-динамических систем, описываемых во входо-выходных отношениях, в виде систем обыкновенных дифференциальных уравнений и/или дифференциально-алгебраических уравнений.

SimInTech – это альтернатива таким программным продуктам как Matlab/Simulink (MathWorks), Dymola (Dassault), AMESim (Siemens), SCADE (ANSYS), VisSim (Visual Solutions), SimulationX (ESI ITI GmbH), Simscape, Modelica, LabVIEW (National Instruments) и другим.

Подробнее о системе SimInTech можно узнать на сайте [simintech.ru](http://simintech.ru).

### 4.2. Моделирование с использованием SimInTech Import

В этой части руководства мы подробно рассмотрим особенности подключения расчетных схем, разработанных в SimInTech к механическим моделям «Универсального механизма» посредством экспорта расчетных схем из SimInTech как динамических библиотек (DLL). Вначале остановимся на основных этапах, которые необходимо выполнить, чтобы правильно подключить расчетную схему SimInTech к модели механической системы и выполнить моделирование динамики совместной (составной) модели.

#### 4.2.1. Порядок работы

В общем случае моделирования динамики механических систем с подключением библиотек SimInTech предполагает выполнение следующих этапов.

- Описание расчетной схемы в среде **SimInTech**.
- Экспорт созданной модели из **SimInTech** в виде DLL-библиотеки.
- Создание модели механической системы в программе **UM Input**.
- Загрузка подготовленной модели механической части в программу **UM Simulation**. Загрузка и подключение DLL-библиотеки SimInTech с помощью **Мастера связи с внешними библиотеками**.
- Моделирование динамики управляемого движения.

Расчетная схема **SimInTech**, подключаемая в UM, рассматривается как черный ящик, который по некоторому закону преобразует входные величины в выходные. При добавлении системы управления в модель механической системы на входы системы управления назначаются переменные, которые создаются при помощи **Мастера переменных**. Выходные величины связываются с параметрами UM-модели.

Для реализации управляющих усилий со стороны системы управления в модель механической системы вводятся силы/моменты, значения или характеристики (например, коэффициент жесткости, коэффициент диссипации) которых задаются параметрами модели. Затем с помощью **Мастера связи с внешними библиотеками** эти параметры связываются с выходами расчетной схемы **SimInTech**.

## 4.2.2. Перевернутый маятник

Модель перевернутого маятника без подключенной системы управления находится в каталоге [{Данные УМ}\SAMPLES\TUTORIAL\inv\\_pend](#). Перед началом урока, проверьте ее наличие в указанном каталоге. Если такой модели нет, то скачайте ее из интернета по адресу: [www.universalmechanism.com/download/10/inv\\_pend.zip](http://www.universalmechanism.com/download/10/inv_pend.zip). В данном уроке мы не будем подробно разбирать этапы создания модели механической части, ограничимся только описанием связи механической части с системой управления SimInTech.

Готовая модель перевернутого маятника с подключенной системой управления находится в каталоге [{Данные УМ}\SAMPLES\SimInTech\inv\\_pend\\_simintech](#)<sup>4</sup>.

---

<sup>4</sup> [www.universalmechanism.com/download/10/inv\\_pend\\_simintech.zip](http://www.universalmechanism.com/download/10/inv_pend_simintech.zip)

### 4.2.2.1. Экспорт системы управления из SimInTech

Экспорт системы управления перевернутым маятником из SimInTech заключается в компиляции подготовленной модели в динамическую библиотеку (DLL), которая впоследствии подключается к программному комплексу «Универсальный механизм».

**Замечание.** Здесь предполагается, что вы знакомы с основами моделирования в среде SimInTech. В любом случае сейчас вы можете пропустить данный параграф и вернуться к нему позднее. Результатом экспорта системы управления из SimInTech является библиотека `inv_pend\pid.dll`. Вы можете пропустить этап создания dll-библиотеки для системы управления и воспользоваться уже подготовленной.

Процесс экспорта модели системы управления состоит из следующих этапов.

1. Подготовка модели системы управления к экспорту.
2. Настройка процесса компиляции и собственно компиляция модели.

#### Подготовка модели системы управления к экспорту

В модель системы управления следует включить блоки «Входной контакт» и «Выходной контакт» с вкладки «Сигналы», которые будут обеспечивать связь модели системы управления с моделью ПК «Универсальный механизм», см. рис. 4.1.

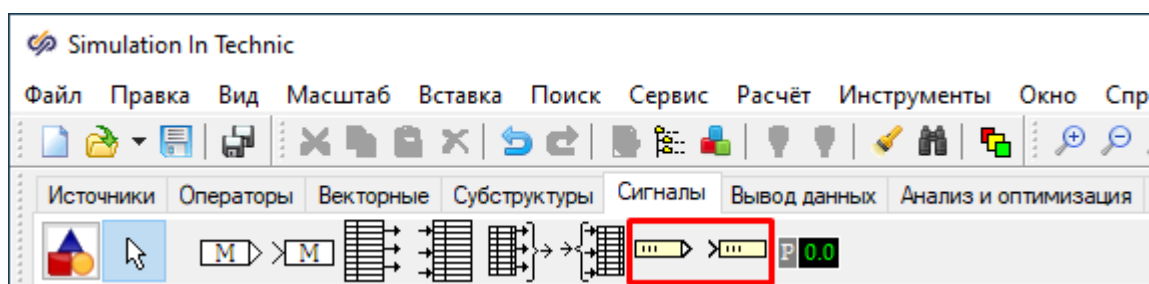


Рис. 4.1. Блоки «Входной контакт» и «Выходной контакт» на вкладке «Сигналы». Копия экрана из SimInTech версии 2.25 от 1 марта 2025 г. Визуальное представление главного окна SimInTech в более поздних версиях программы может меняться.

**Замечание.** Обратите внимание, что обмен данными между «Универсальным механизмом» и DLL, сгенерированной с помощью SimInTech, поддерживается только через указанные выше сигналы «Входной контакт» и «Выходной контакт». Поддерживаются только скалярные сигналы типа Float. Другие блоки сигналов с этой вкладки, а также векторные, матричные и другие типы сигналов в УМ не поддерживаются для описания входов/выходов.

В нашей модели перевернутого маятника входной переменной для системы управления является угол отклонения маятника от вертикали. Соответственно нужно предусмотреть один вход в модель системы управления. В свою очередь для организации управле-

ния перевернутым маятником в нашей модели системы управления предусмотрен один выход – усилие, подаваемое на тележку.

Подготовленная модель системы управления перевернутым маятником в SimInTech представлена на рис. 4.2.

**Замечание.** Вы можете самостоятельно собрать модель, представленную на рис. 4.2 или же загрузить готовую модель SimInTech из файла [{Данные UM}\SAMPLES\TUTORIAL\inv\\_pend\pid.prt](#).

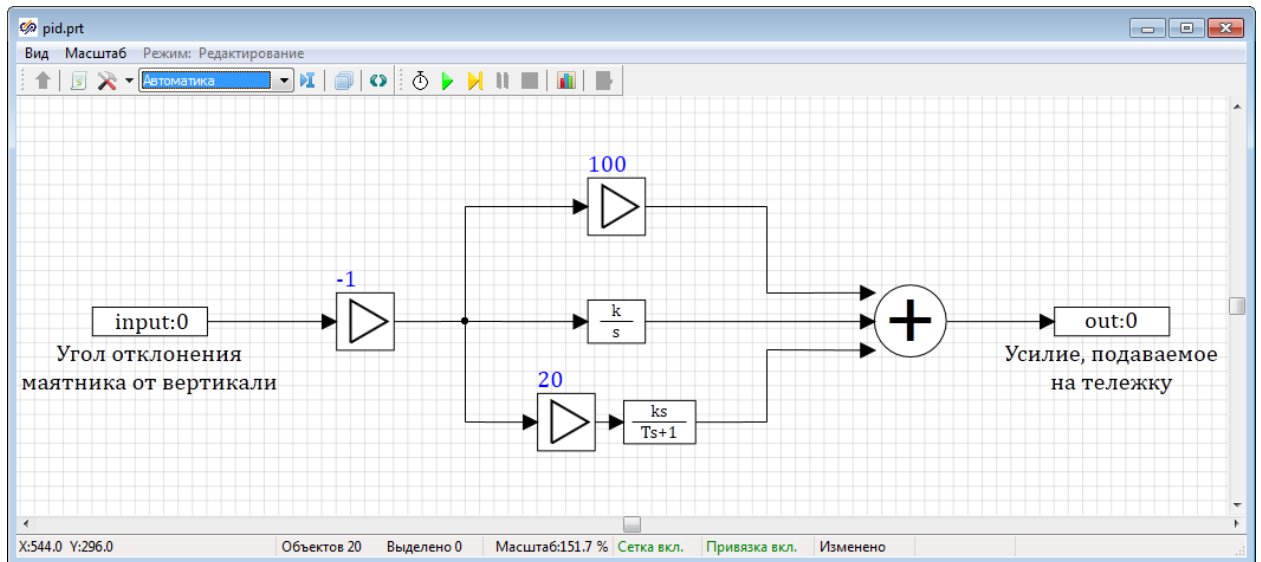


Рис. 4.2. Модель системы стабилизации перевернутого маятника

## Настройка процесса компиляции и компиляция модели

В меню **Кодогенератор** выберите пункт **Кодогенератор СИ**, (рис. 4.3).

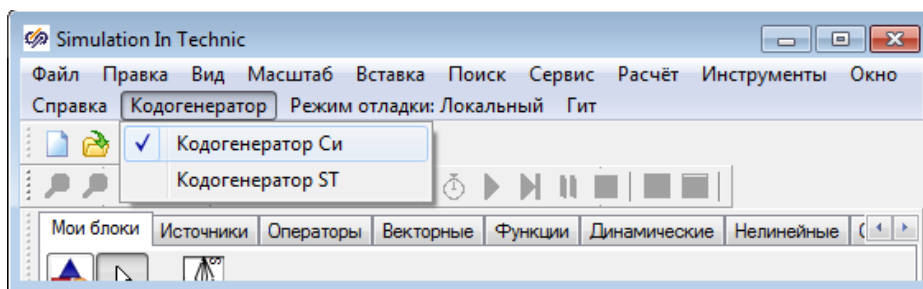


Рис. 4.3 Меню **Кодогенератор**

В появившемся диалоговом окне перейдите на вкладку **Настройки**. Убедитесь, что поле **Директория исходников** пусто. В поле **Директория шаблона кода** задайте `%codetemplates%MinGW_DLL\`, как показано на рис. 4.4. Значения остальных полей оставьте по умолчанию, как показано на рис. 4.5.

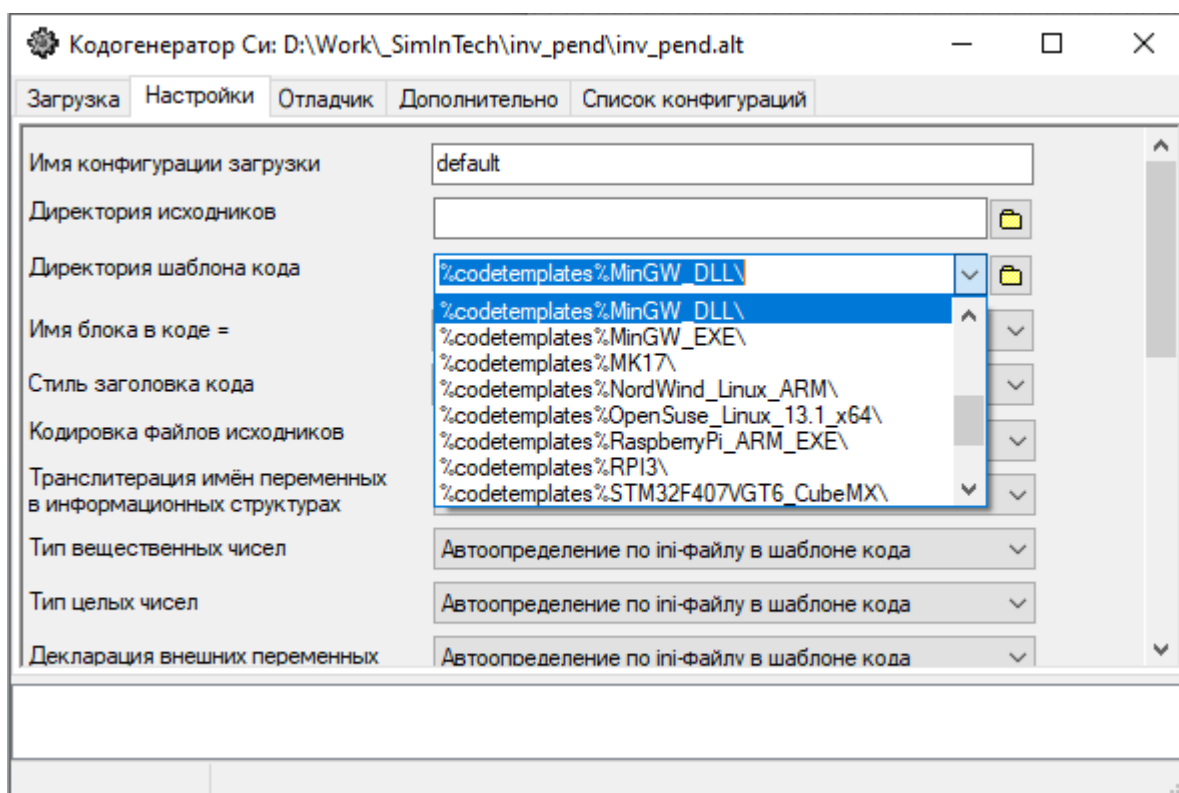


Рис. 4.4 Настройка процесса компиляции в SimInTech – шаблон кода

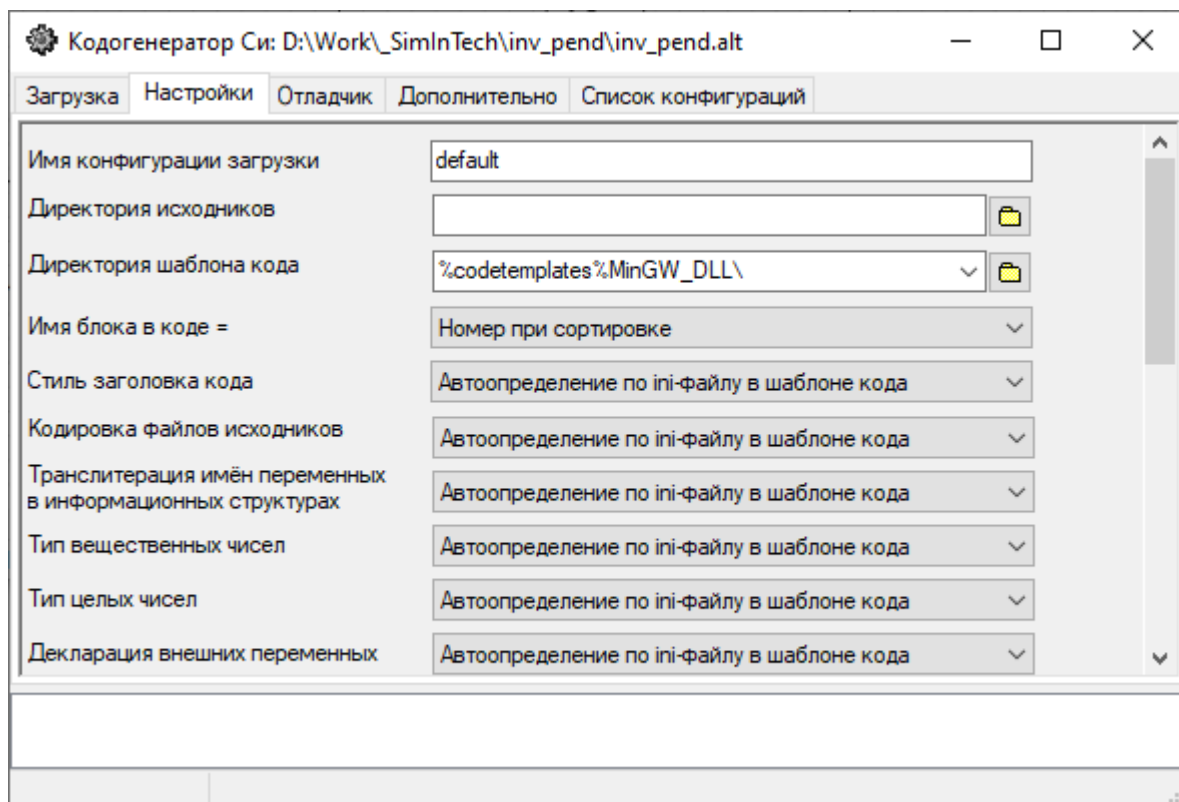


Рис. 4.5 Настройка процесса компиляции в SimInTech – остальные поля

Далее перейдите на вкладку **Загрузка**. Нажмите **Сохранить конфигурацию** (рис. 4.6). После этого окно кодогенератора можно закрыть.

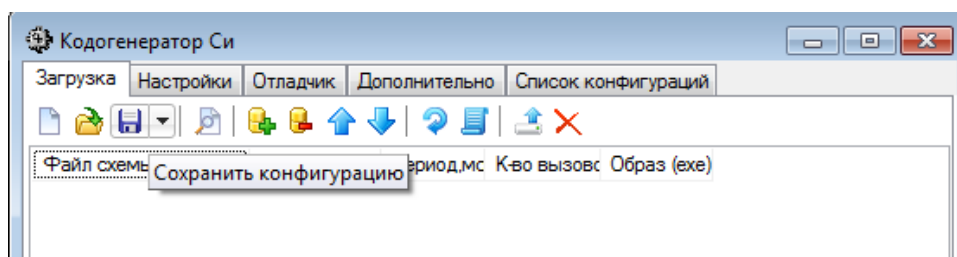
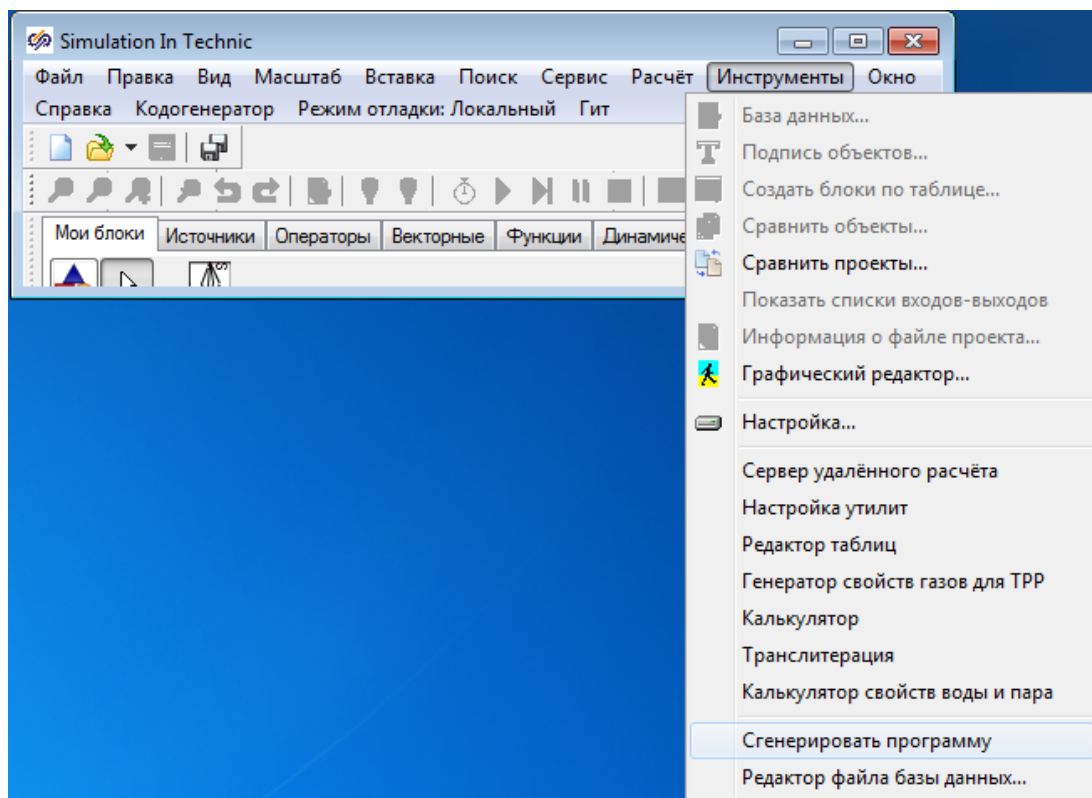


Рис. 4.6 Сохранение конфигурации

Для запуска процесса компиляции и сборки dll-библиотеки необходимо в меню **Инструменты** выбрать пункт **Сгенерировать программу**, рис. 4.7.

Рис. 4.7 Меню **Инструменты**

Процесс компиляции сопровождается выводом сообщений о ходе процесса в командное окно системы SimInTech.

В результате выполнения компиляции в каталоге модели появятся сгенерированные исходные файлы библиотеки и откомпилированная DLL. Для нашего примера это **pid.dll**.

#### 4.2.2.2. Подключение библиотеки SimInTech в UM

##### Загрузка модели механической части

Запустите программу моделирования **UM Simulation**.

Загрузите модель [{Данные UM}\SAMPLES\TUTORIAL\inv\\_pend](#).

Откройте новое анимационное окно.

##### Загрузка модели SimInTech

Выберите пункт меню **Инструменты | Внешние интерфейсы | Интерфейс с внешними библиотеками...**

Появится окно **Мастера связи с внешними библиотеками**.

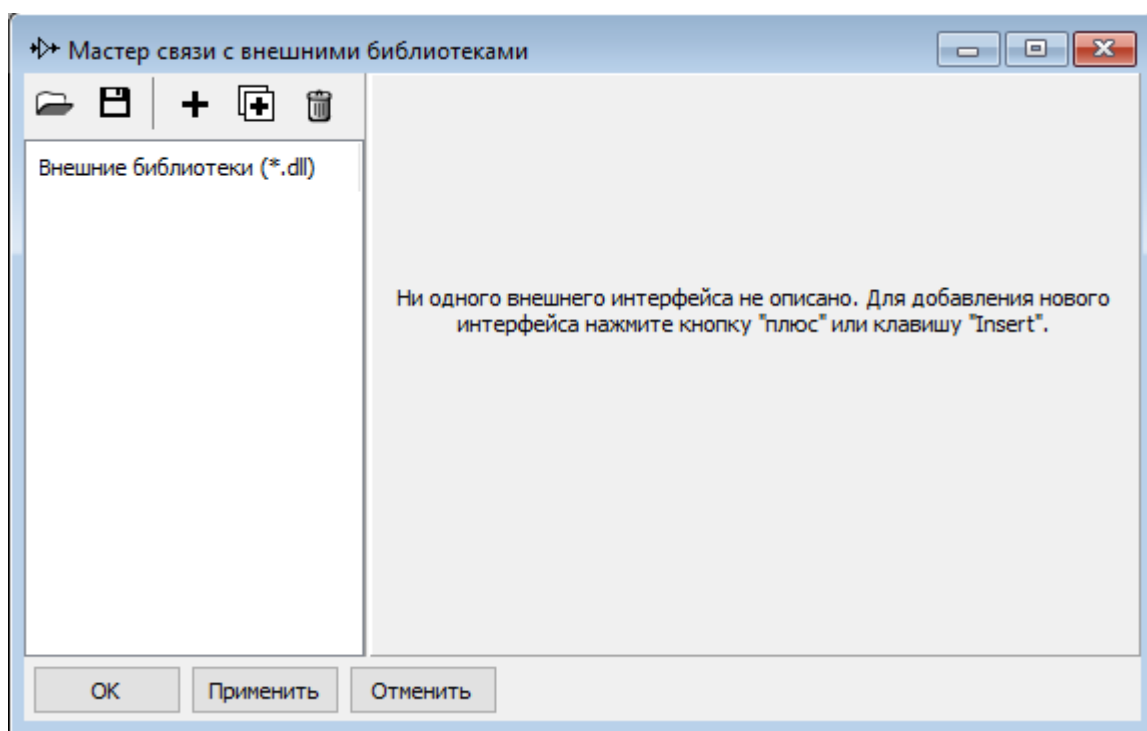


Рис. 4.8. Мастер связи с внешними библиотеками

Нажмите кнопку **+** для добавления новой библиотеки **SimInTech**.

В поле **Путь к внешней библиотеке** выберите файл **pid.dll**. Если вы создали выходную dll самостоятельно, то выберите ее, иначе выберите готовую dll системы управления из каталога **..\inv\_pend**.

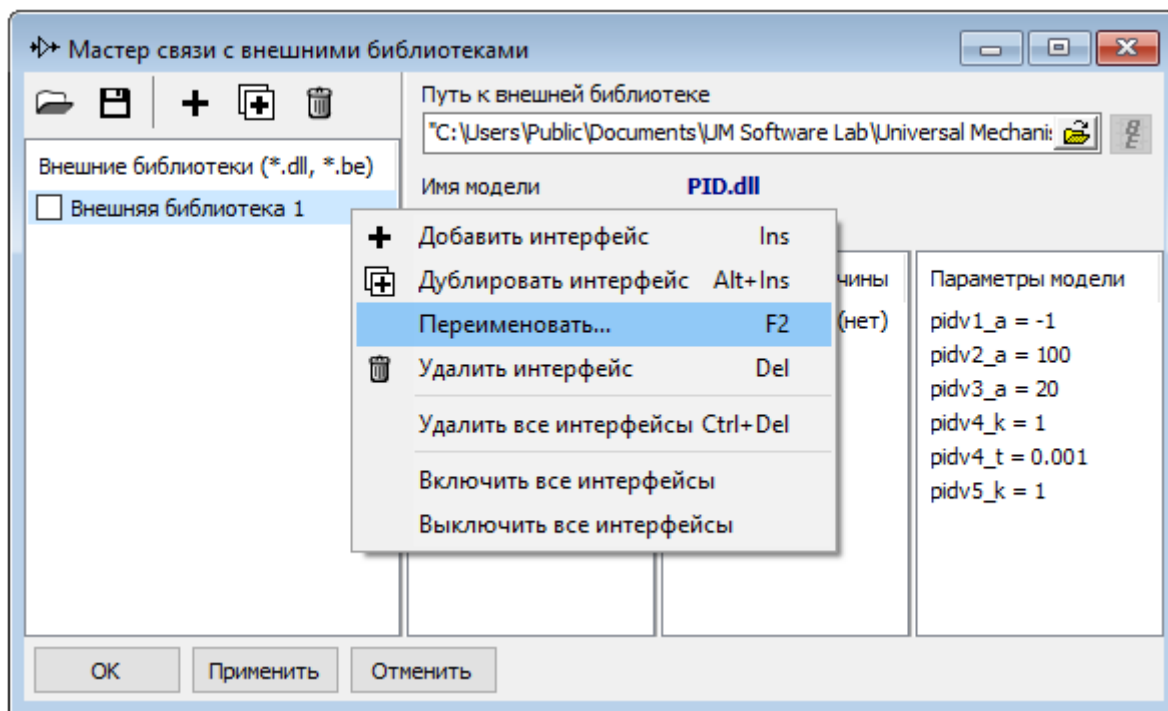
**Мастер связи** загрузит указанную модель и установит число входных и выходных величин модели системы управления, а также число параметров этой модели и переменных состояния.

В данном случае в окне появится одна входная и одна выходная величина, см. рис. ниже. На вход системы управления мы подадим угол отклонения маятника от вертикали, а выходную величину подадим в качестве усилия на тележку.

## Переименование интерфейса

Выберите в списке **Внешние библиотеки (\*.dll)** текущую **Внешнюю библиотеку 1**, правой кнопкой мыши вызовите контекстное меню и в нем выберите пункт **Переименовать...**, или нажмите клавишу **F2**.

В поле редактирования названия введите **Control force** и нажмите клавишу **Enter**.




## Назначение входных величин модели SimInTech

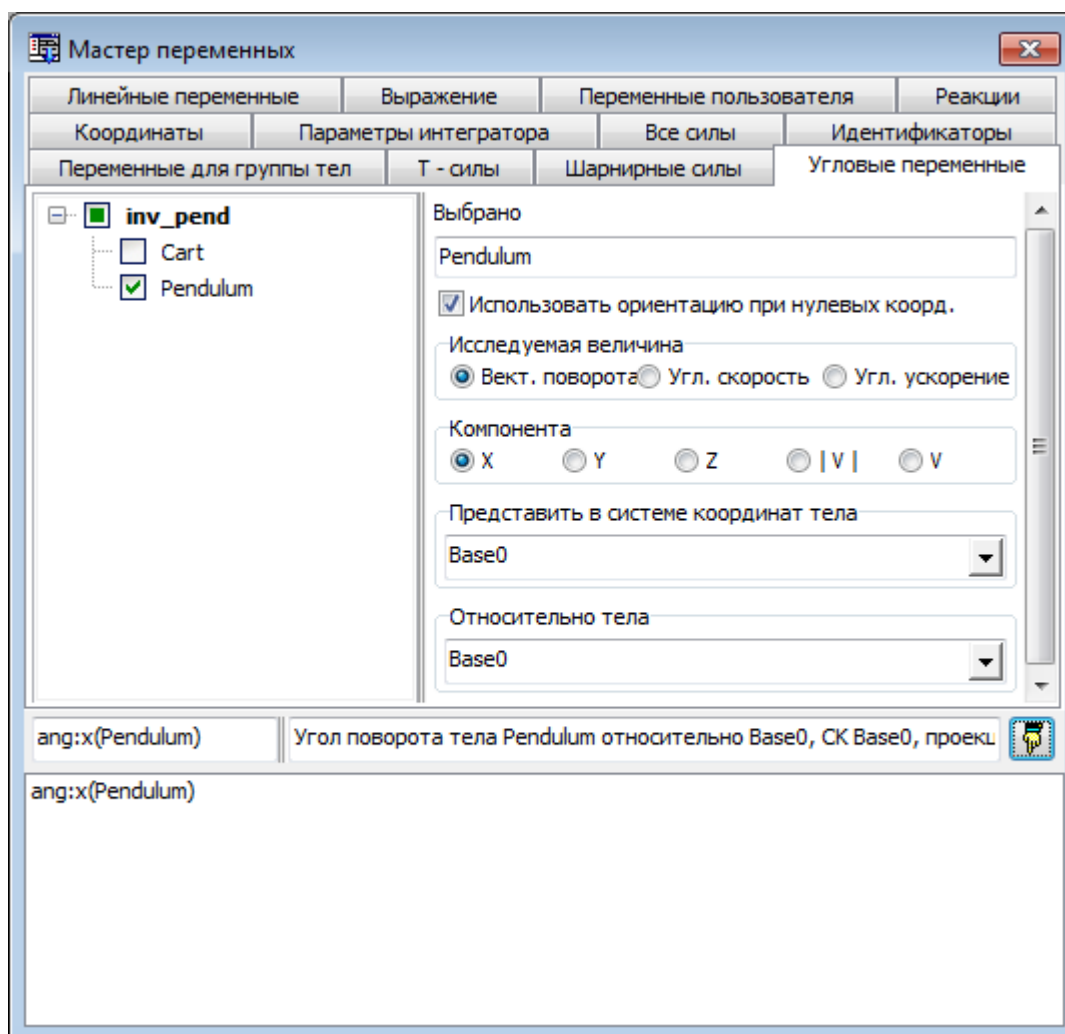
С помощью мастера переменных сформируем переменную «угол отклонения маятника от вертикали» и назначим ее в качестве входной величины для системы управления.

Запустите **Мастер переменных**.

Перейдите на вкладку **Угловые переменные**.

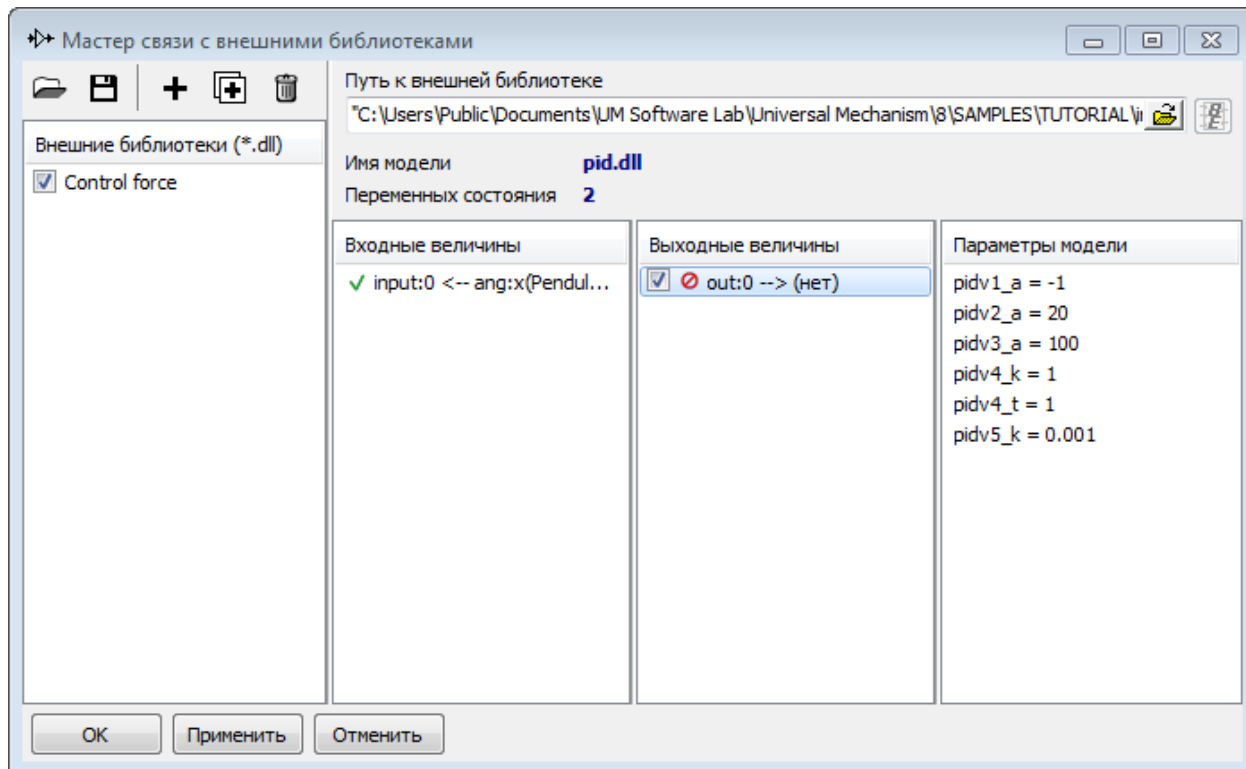
В списке тел слева выберите **Pendulum**, в поле **Исследуемая величина** выберите **Вект.(ор) поворота**, а в поле **Компонента** выберите **X**.

Создайте переменную кнопкой . Переменная **ang:x(Pendulum)** появится в контейнере переменных.



Перетащите мышкой созданную переменную **ang:x(Pendulum)** в окно **Мастера связи с внешними библиотеками** на вход **Input:0**.

Вход модели **Input:0** станет помечен зеленой галочкой, что означает, что на данный вход модели SimInTech назначена переменная.



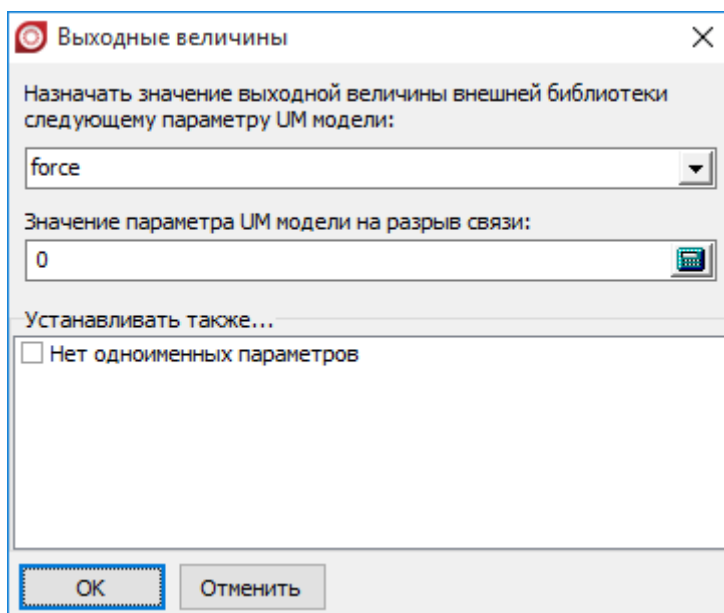
## Назначение выходных величин модели SimInTech

Выходные величины модели SimInTech связываются с параметрами UM-модели.

Дважды щелкните на элементе **Выходные величины** | **out:0**. Появится диалоговое окно назначения параметров UM-модели на выход модели SimInTech.

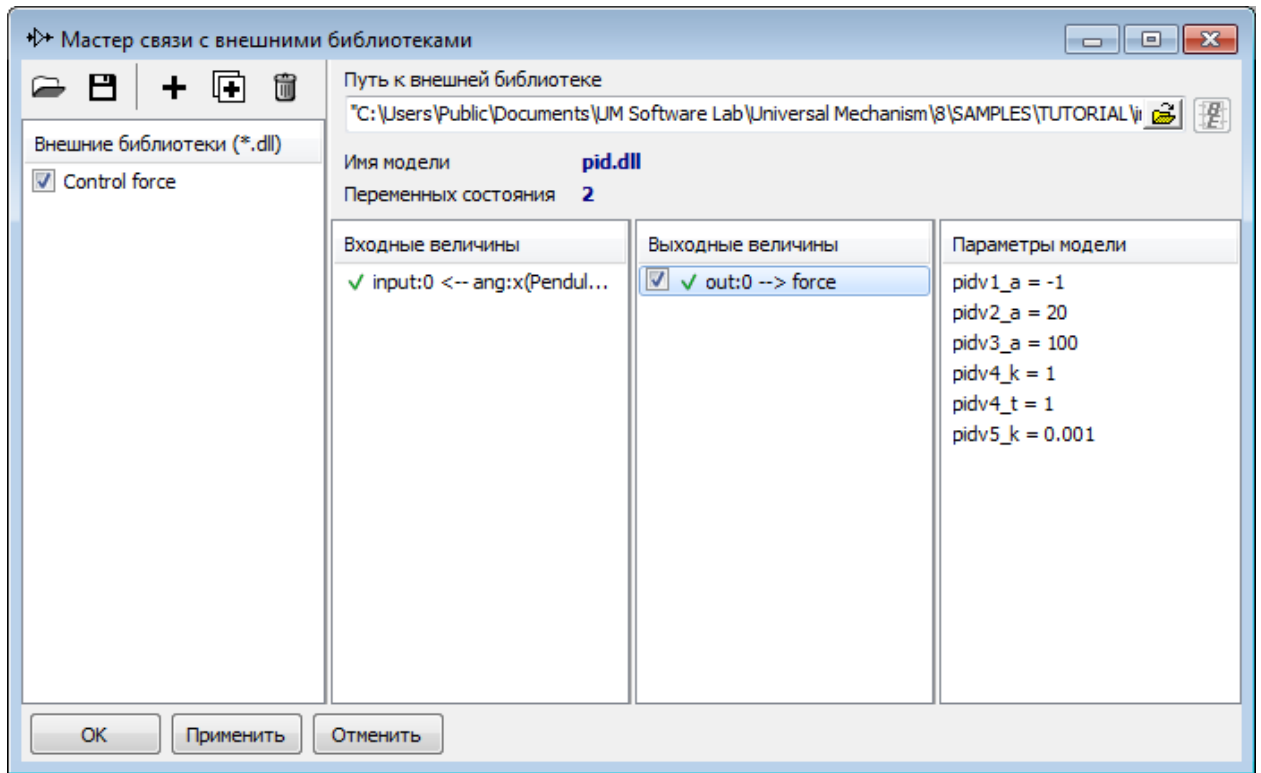
В поле **Параметр UM модели** выберите **force**.

Закройте окно кнопкой **ОК**.



Теперь модель SimInTech полностью связана с моделью механической системы.

Нажмите **Применить** для актуализации настроек без закрытия окна **Мастера связи с внешними библиотеками** или **ОК** для сохранения данных с закрытием окна.



### 4.2.2.3. Моделирование движения

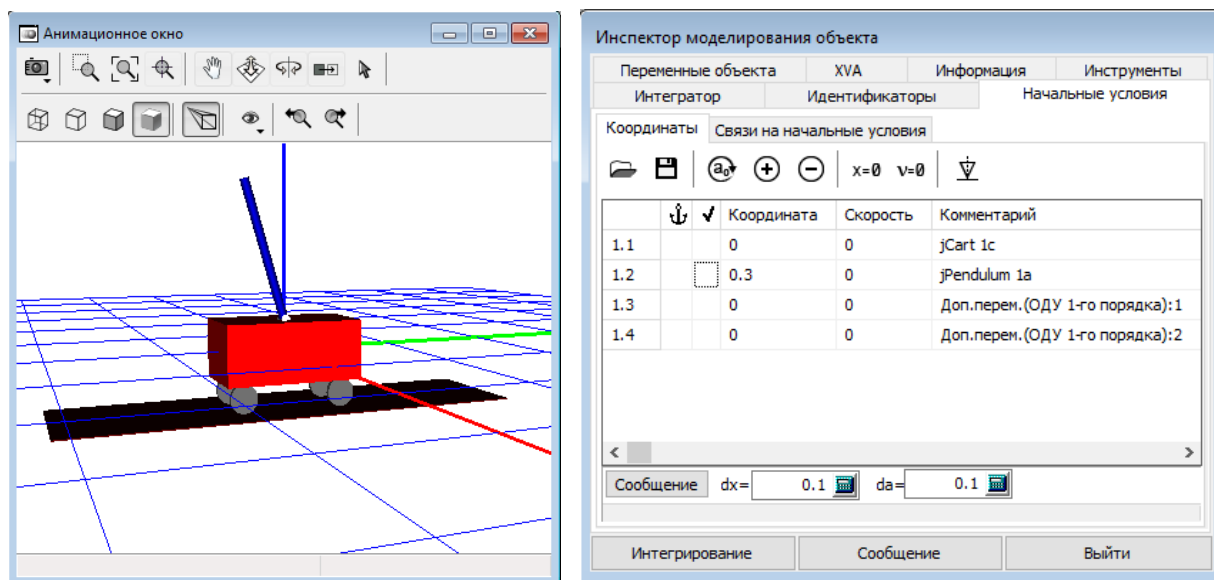
Теперь наша модель перевернутого маятника полностью описана. Сейчас мы отклоним маятник из положения равновесия и проверим, действительно ли система управления работает.

Выберите пункт меню **Анализ | Моделирование...** Откроется **Мастер моделирования объекта**.

Перейдите на вкладку **Начальные условия**.

Отклоним маятник на 0.3 радиана от положения равновесия. В поле второй координаты (Координата/1.2) введите **0.3**.

**Замечание.** Первая координата – положение тележки относительно оси Y. Третья и четвертая координаты – переменные состояния модели SimInTech.



Перейдите на вкладку **Интегратор**, в поле **Численный метод** выберите **Park**. Ниже в секции описания условий завершения введите **Время > 0.1**, в поле **Шаг представления результатов** установите значение **0.0002**, а в поле **Погрешность** установите **1E-7**.

Нажмите кнопку **Интегрирование**.

Действительно, как вы можете наблюдать, результаты моделирования показывают, что система управления стабилизировала перевернутый маятник.

### Построение вектора управляющей силы и графиков переменных


Выведем в анимационное окно график управляющей силы, действующей на тележку, а в графические окна – величину этой силы и угол поворота маятника.


Сформируем переменную – вектор силы, для отрисовки в анимационном окне.

Откройте **Мастер переменных**.

Перейдите на вкладку **T-силы**.

В списке слева поставьте галочку возле силы **Control force**, в поле **Исследуемая величина** выберите **Сила**, в поле **Компонента** – **V** (вектор), в поле **Действует на тело** – **Cart**.

Нажав кнопку  поместите сформированную переменную в контейнер переменных и перетащите ее в анимационное окно.

Кнопкой  переключите анимационное окно в режим проволочной графики. Это позволит наблюдать вектор силы, даже когда он будет изображаться внутри тележки.

А теперь сформируем две переменные для представления в графических окнах: величину управляющей силы и угол поворота маятника.

Откройте новое графическое окно. В нем будем строить график управляющей силы.

Перейдите обратно в **Мастер переменных**, на вкладку **T-силы**. В поле **Компонента** выберите **Y**. Поместите переменную в контейнер и перетащите ее в графическое окно.

Откройте еще одно графическое окно. В этом окне мы будем строить зависимость угла поворота маятника от времени.

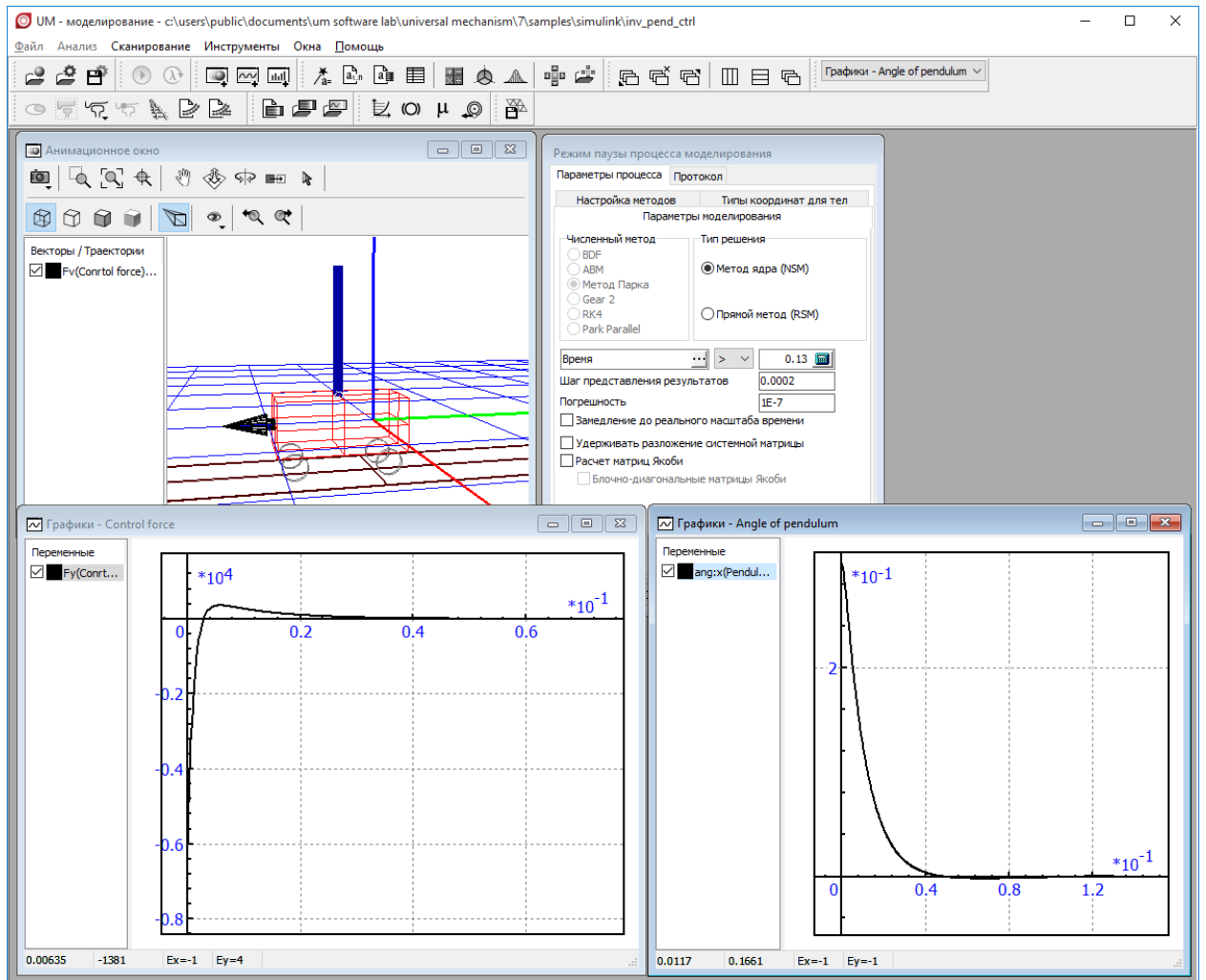
Перейдите в окно **Мастера переменных** на вкладку **Угловые переменные**.

В списке тел выберите **Pendulum**, в поле **Исследуемая величина** выберите **Вект.(ор) поворота**, в поле **Компонента** – **X**.

Поместите переменную в контейнер переменных, а затем перетащите ее в графическое окно.

Закройте окно **Мастера переменных**.

Запустите процесс моделирования.



## 4.3. Моделирование с использованием SimInTech CoSimulation

В этой части руководства будут рассмотрены примеры использования инструмента **SimInTech CoSimulation** из модуля **UM Control**. Этот подход подразумевает, что модель механической части будет экспортирована из UM и импортирована в SimInTech, где будет представляться в системе управления в виде отдельного блока, имеющего некоторое количество входов и выходов. Остановимся на основных моментах, поясняющих суть предлагаемой технологии совместного моделирования (косимуляции).

Работа UM-моделей в среде SimInTech поддерживается специальным блоком «Связь с расчетным кодом Универсальный механизм», расположенным на вкладке «Данные». Этот блок доступен для пользователей SimInTech начиная с версии 1.17.9.29.

### 4.3.1. Порядок работы

В общем случае моделирования динамики механических систем с использованием **SimInTech CoSimulation** предполагается выполнение следующих этапов.

- Описание модели системы управления в среде **SimInTech**.
- Включение в эту модель блока «Универсальный механизм», который представляет собой модель механической системы.
- Создание модели механической системы в программе **UM Input**.
- Загрузка подготовленной модели механической части в программу **UM Simulation**. Формирование входов, выходов и файлов настроек с помощью **Мастера экспорта** для подключения этой модели в систему управления **SimInTech**.
- Создание связи между UM-моделью и системой управления **SimInTech**.
- Моделирование динамики управляемого движения в среде **SimInTech**.

Модель механической системы UM, подключаемая в систему управления **SimInTech**, рассматривается как черный ящик, который по некоторому закону преобразует входные величины в выходные. При включении механической системы в модель системы управления на выходы механической системы назначаются *переменные*, которые создаются при помощи **Мастера переменных**. Входные величины блока «Универсальный механизм» связываются с параметрами UM-модели.

Для реализации управляющих усилий со стороны системы управления в модель механической системы вводятся силы/моменты, значения или характеристики (например, коэффициент жесткости, коэффициент диссипации) которых задаются параметрами модели. Затем с помощью **Мастера экспорта** эти параметры связываются с величинами, которые являются входами блока **SimInTech**.

### 4.3.2. Перевернутый маятник

Модель перевернутого маятника с готовыми файлами настроек и управляющим m-файлом находится в каталоге `{Данные УМ}\SAMPLES\Cosimulation\inv_pend_cosim`. Перед началом урока, проверьте ее наличие в указанном каталоге. Если такой модели нет, то скачайте ее из интернета по адресу:

[www.universalmechanism.com/download/10/inv\\_pend\\_cosim.zip](http://www.universalmechanism.com/download/10/inv_pend_cosim.zip).

В данном уроке мы не будем подробно разбирать этапы создания модели механической части, ограничимся только описанием связи механической части с системой управления в SimInTech.

### 4.3.2.1. Подготовка системы управления в SimInTech

Модель системы управления перевернутым маятником для SimInTech CoSimulation, см. рис. 4.9, очень похожа на ту модель, которая рассматривалась ранее применительно к инструменту **SimInTech Import**, рис. 4.2, но имеет одно существенное отличие: модель механической системы представляет собой отдельный блок «Связь с расчетным кодом Универсальный механизм» – далее «УМ», который доступен на вкладке «Данные» в среде SimInTech.

Именно этот блок обеспечивает связь модели системы управления с моделью ПК «Универсальный механизм». В нашей модели для организации управления перевернутым маятником входной переменной для блока «УМ» будет величина силы, подаваемой на тележку, а выходной величиной блока «УМ» будет угол отклонения маятника от вертикали.

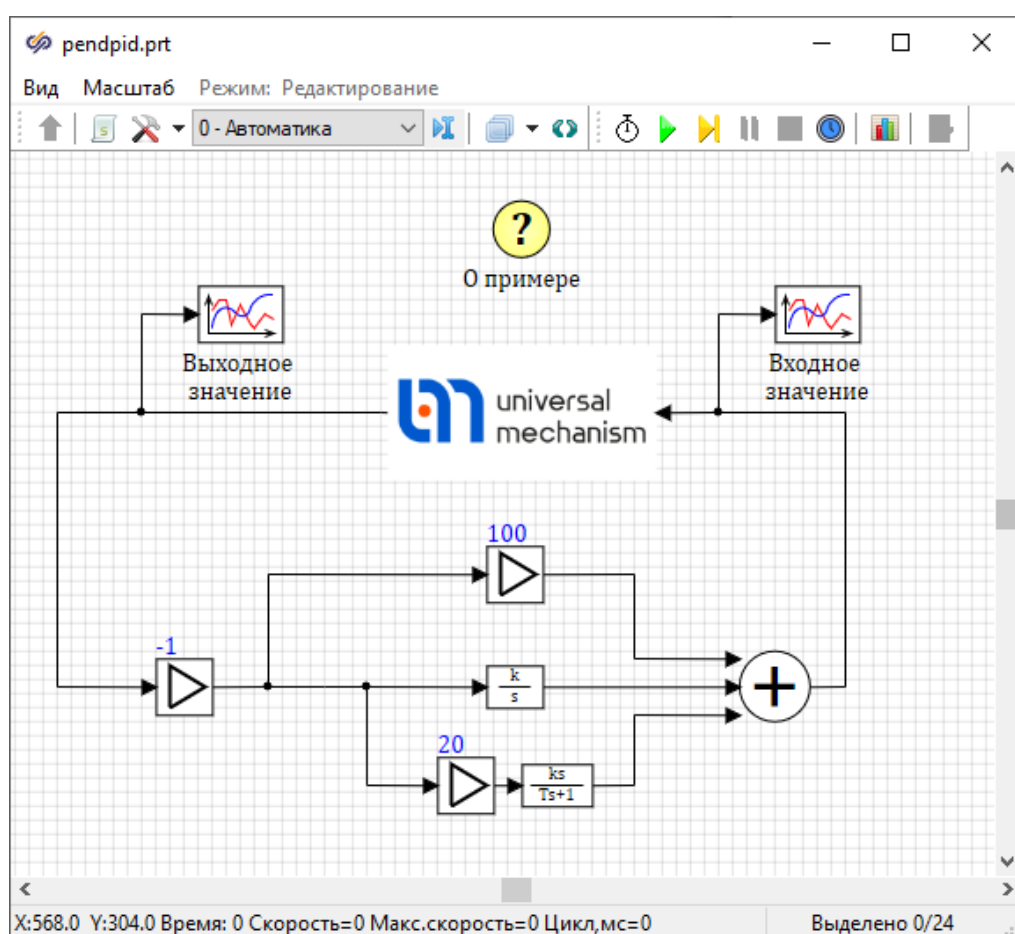


Рис. 4.9. Система стабилизации перевернутого маятника с механической системой как блок «УМ»

Подготовленную модель системы управления перевернутым маятником в SimInTech вы можете найти в файле

[{Данные УМ}\SAMPLES\Cosimulation\inv\\_pend\\_cosim\pendpid.prt](#), рис. 4.9.

К модели в SimInTech мы обратимся в п. 4.3.2.3, а пока рассмотрим особенности экспорта механической части из UM.

### 4.3.2.2. Экспорт модели механической системы из UM

На следующем шаге нам необходимо указать, какие именно входные и выходные величины блока «УМ» потребуются для моделирования, сохранить связанные с этим настройки и сгенерировать специальный cosim-файл, который будет управлять блоком «УМ».

#### Загрузка модели механической части

Запустите программу моделирования **UM Simulation**.

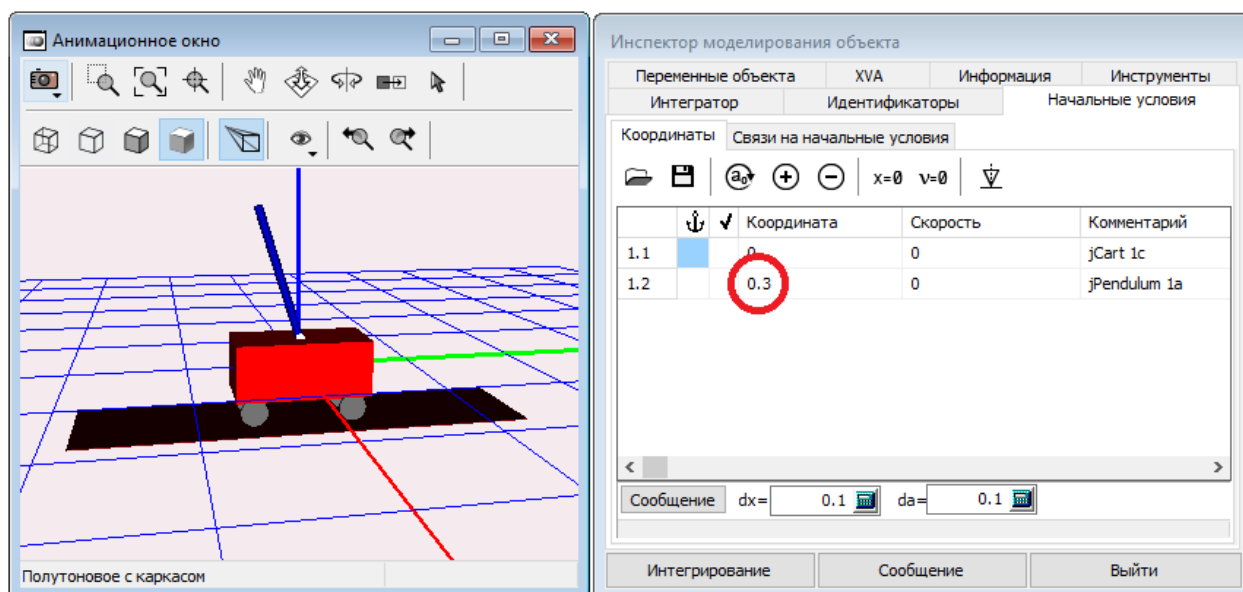
Загрузите модель [{Данные УМ}\SAMPLES\Cosimulation\inv\\_pend\\_cosim](#).

#### Задание начальных условий

Перед выполнением дальнейших действий убедитесь в том, что перевернутый маятник отклонен от вертикали и при необходимости задайте это отклонение.

Выберите пункт меню **Анализ | Моделирование...**

В появившемся окне **Инспектора моделирования объекта** перейдите на вкладку **Начальные условия | Координаты** и убедитесь, что координата **1.2** имеет значение **0.3** (рад), а маятник в анимационном окне отклонен от вертикали.



#### Мастер экспорта в SimInTech

Выберите пункт меню **Инструменты | Мастер экспорта...**

Появится окно **Мастера экспорта**.

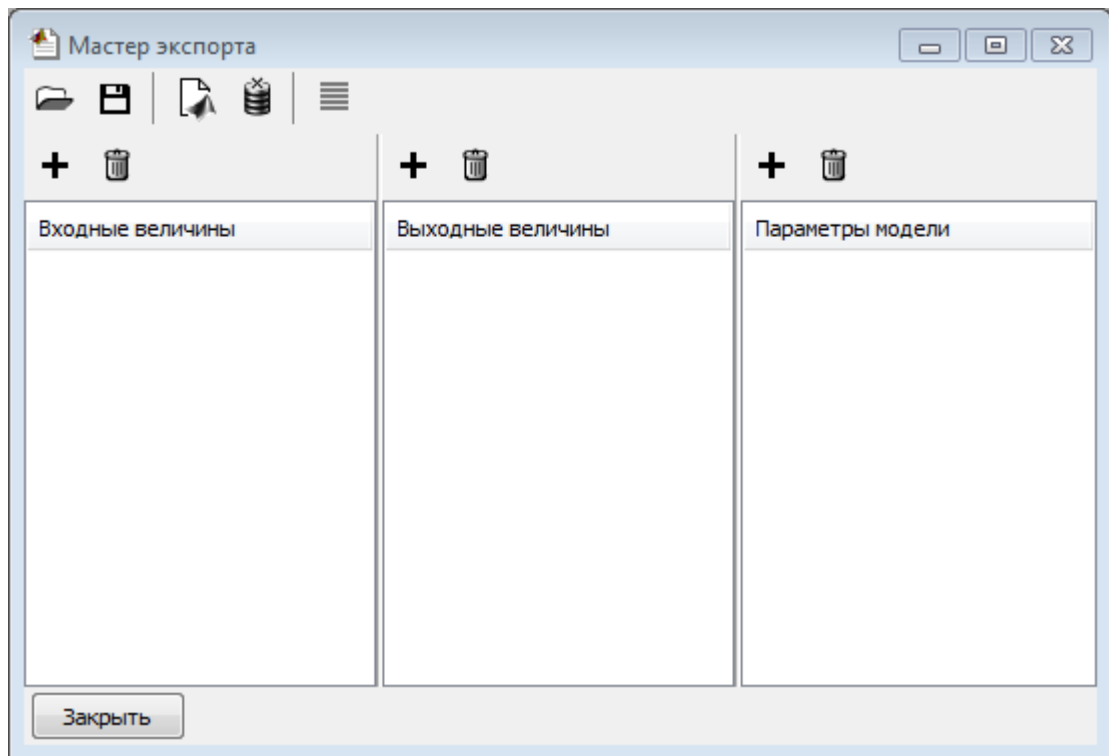


Рис. 4.10. Мастер экспорта

В данном случае нам необходимо назначить одну входную и одну выходную величины. На вход модели перевернутого маятника мы подадим величину усилия на тележку, а в качестве выходной величины назначим угол отклонения маятника от вертикали.

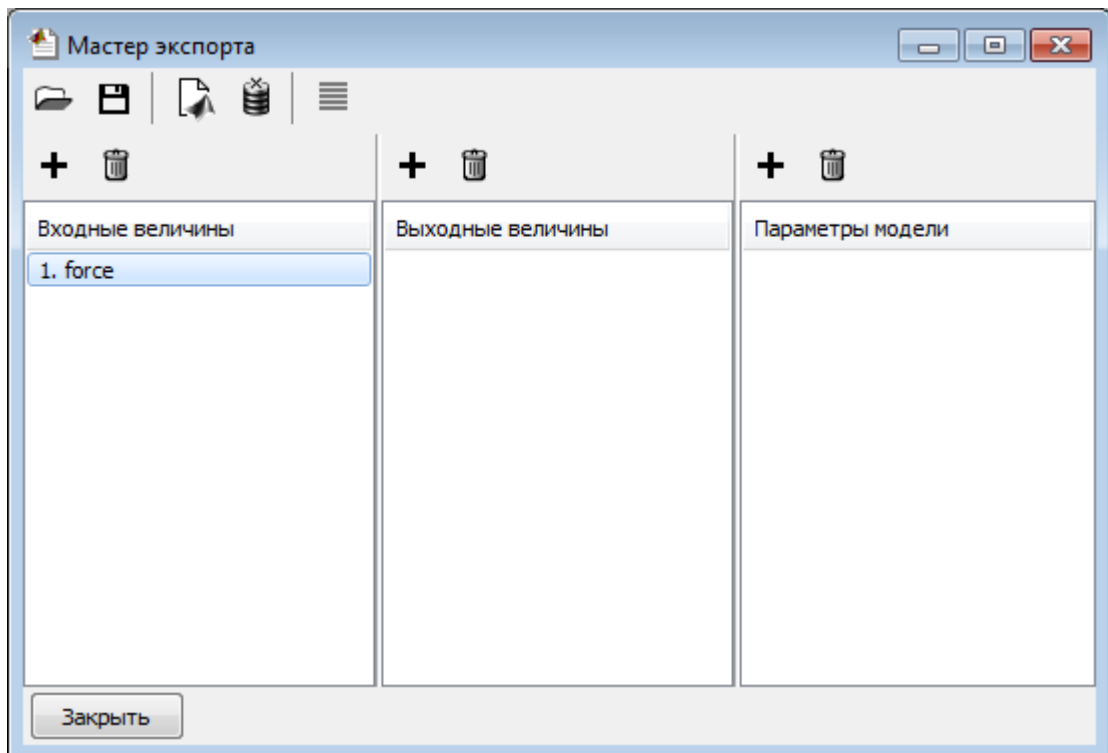
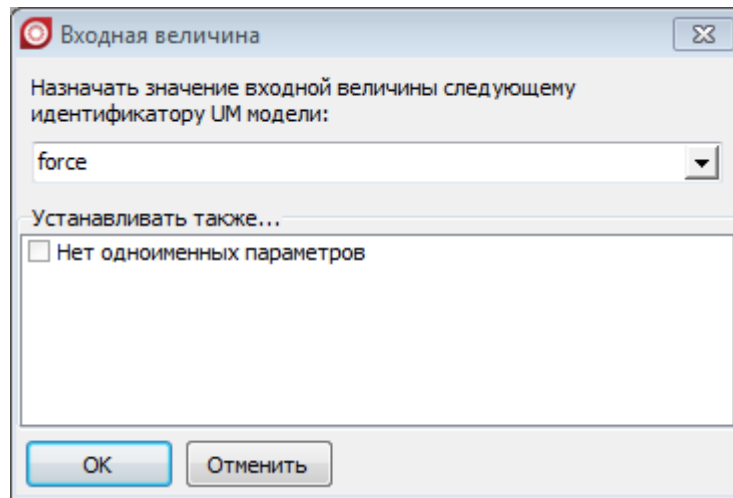
## Назначение входных величин

Для того чтобы добавить новую входную величину для UM-модели необходимо выполнить следующие действия.

Нажмите кнопку **+** над списком **Входные величины**. Появится диалоговое окно назначения входных величин блока «УМ» идентификаторам UM-модели.

В поле **Назначать значение входной величины...** выберите идентификатор модели **force**.

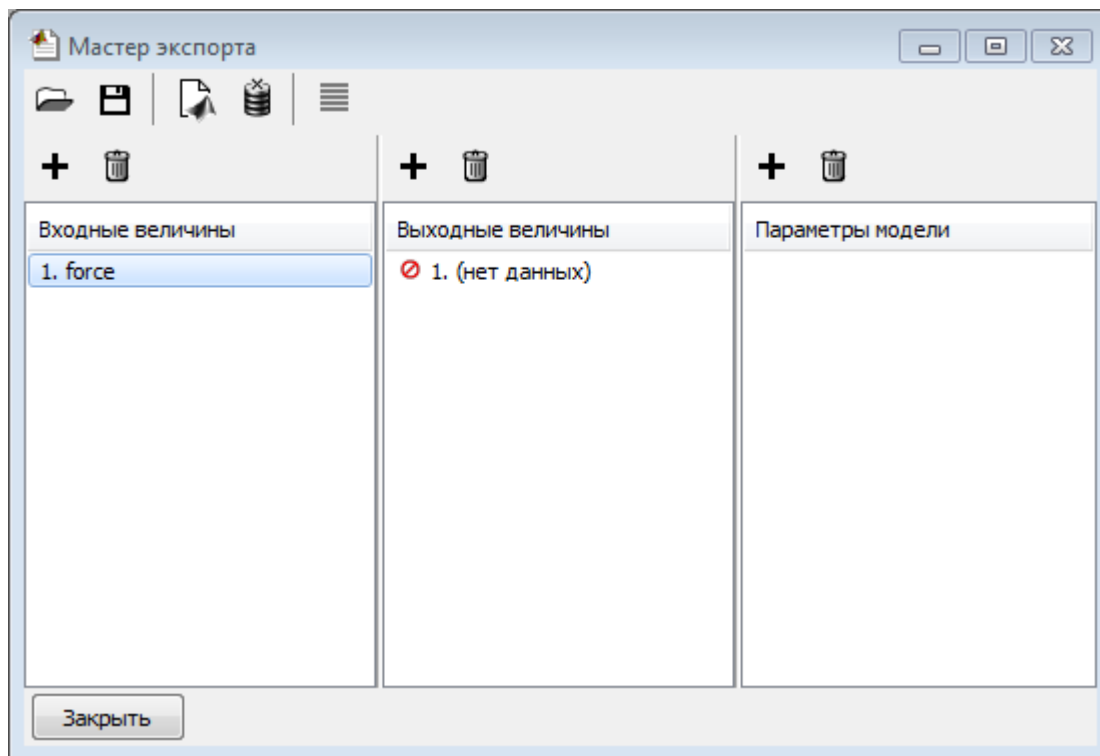
Закройте окно кнопкой **ОК**.



### Назначение выходных величин для модели перевернутого маятника

С помощью **Мастера переменных** сформируем переменную **Угол отклонения маятника от вертикали** и назначим ее в качестве выходной величины блока «УМ».


Нажмите кнопку **+** над списком **Выходные величины**. В списке выходных величин появится одна новая величина.



Запустите **Мастер переменных** (пункт меню **Инструменты | Переменные | Мастер переменных...**).

Перейдите на вкладку **Угловые переменные**, см. рис. 4.11.

В списке тел слева выберите **Pendulum**, выключите флажок **Использовать ориентацию при нулевых коорд.(инатах)**, в поле **Исследуемая величина** выберите **Вект.(ор) поворота**, а в поле **Компонента** выберите **X**.

Создайте переменную кнопкой . Переменная **ang:x(Pendulum)** появится в контейнере переменных.

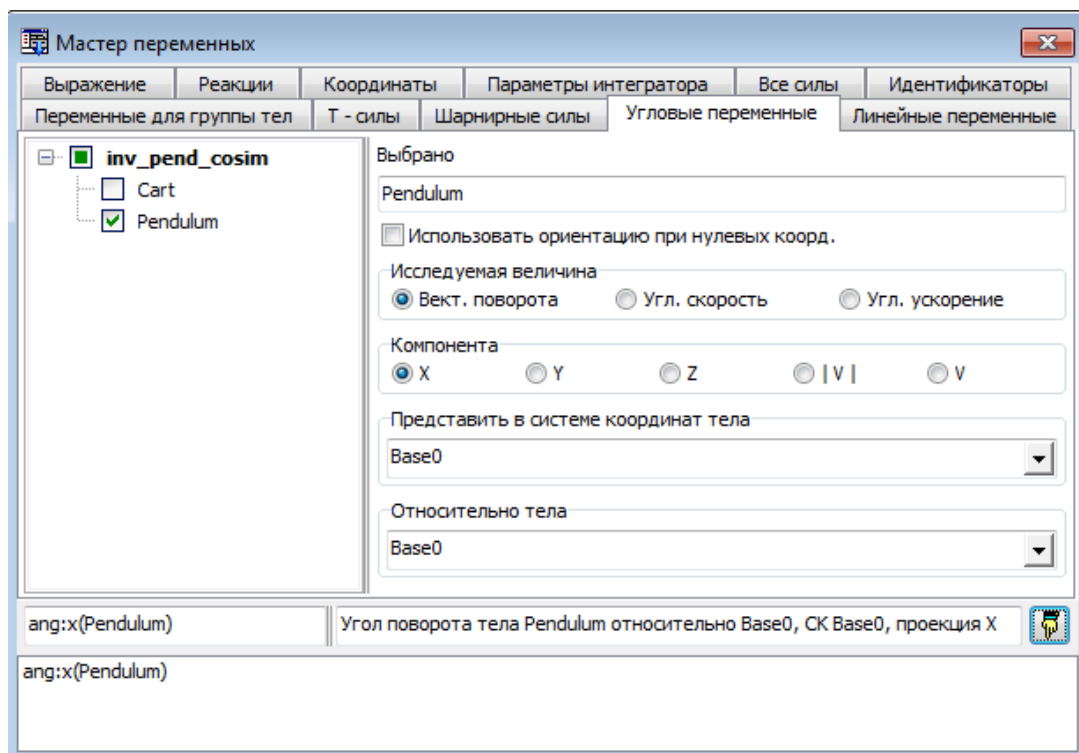
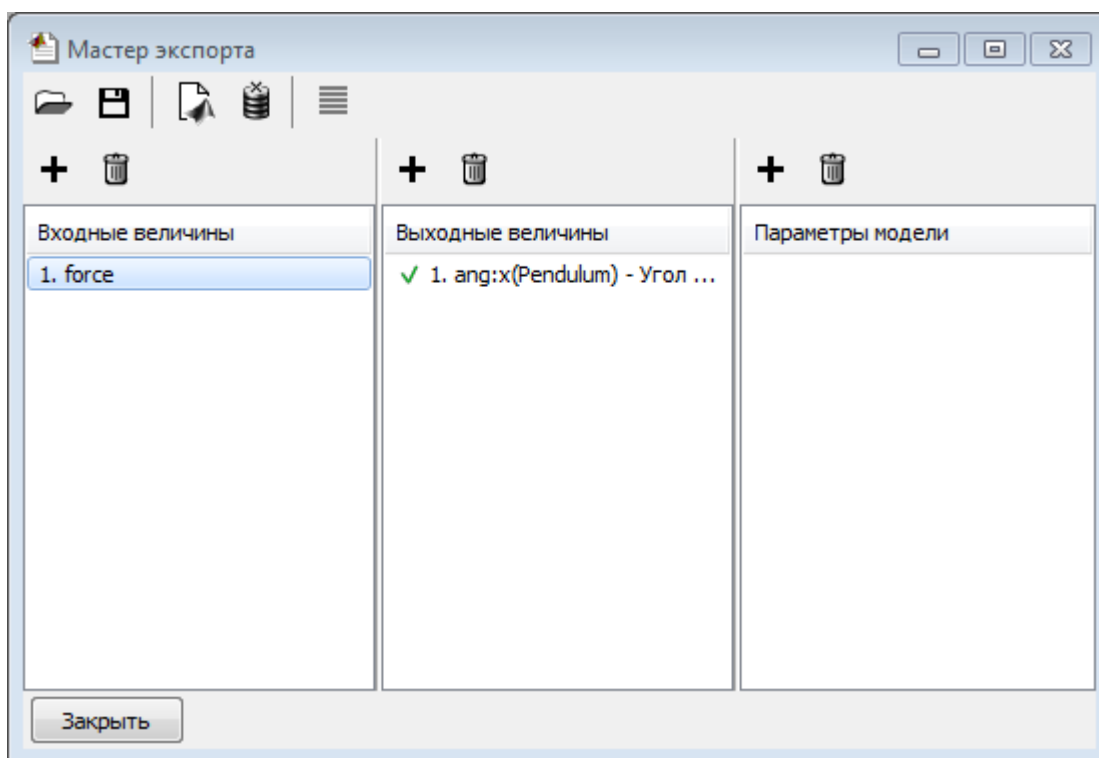



Рис. 4.11. Создание переменной

Перетащите мышкой созданную переменную **ang:x(Pendulum)** в окно **Мастера экспорта...** и назначьте ее в качестве выходной величины.

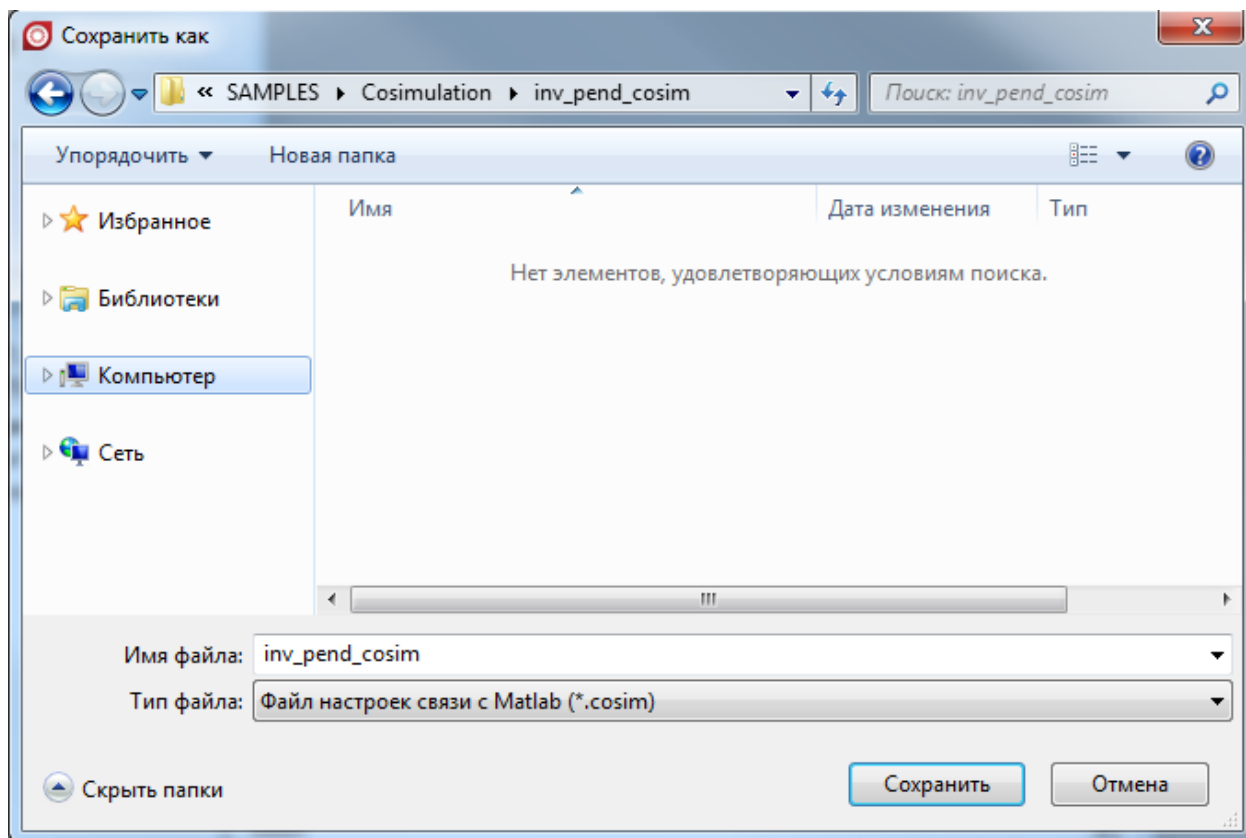
Выход модели станет помечен зеленой галочкой, что означает, что на данный выход блока «УМ» назначена переменная.



Все необходимые настройки для экспорта UM-модели в SimInTech закончены. Далее сохраним настройки в файл \*.cosim.

Сохраните заданную конфигурацию в **inv\_pend\_cosim.cosim** нажав кнопку .

В появившемся окне диалога сохранения файла необходимо указать в названии файла **inv\_pend\_cosim** и сохранить его. Одновременно с созданием файла \*.cosim будут сохранены все файлы конфигурации (файл параметров, начальных условий и т.д.), которые будут использоваться «Универсальным механизмом» при работе под управлением SimInTech.



На этом действия на стороне UM закончены. Переходим к описанию связи на стороне SimInTech.

### 4.3.2.3. Создание связи между UM-моделью и SimInTech

Для создания связи между системой управления SimInTech и механической моделью выполните следующие действия.

Запустите **SimInTech**.

В меню **Файл** нажмите **Открыть...** и выберите каталог {Данные УМ}\SAMPLES\Cosimulation\inv\_pend\_cosim.

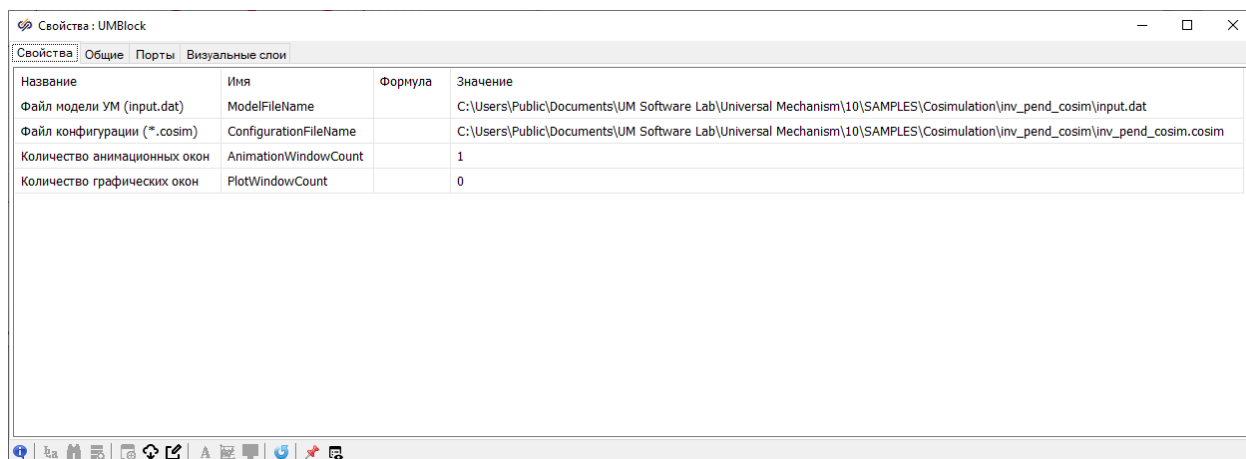
Откройте файл **pendpid.prt**, в котором хранится модель системы управления перевернутым маятником.

Двойным щелчком мыши на блок «УМ» вызовите окно параметров этого блока.

В строке **Файл модели УМ (input.dat)** в столбце **Значение** введите полный путь к модели в ПК «Универсальный механизм», включая файл input.dat.

В строке **Файл конфигурации (\*.cosim)** в столбце **Значение** установите полное имя файла конфигурации, включая файл inv\_pend\_cosim.cosim.

**Замечание.** Обратите внимание, что значения в столбце **Формула** нужно оставить пустыми.



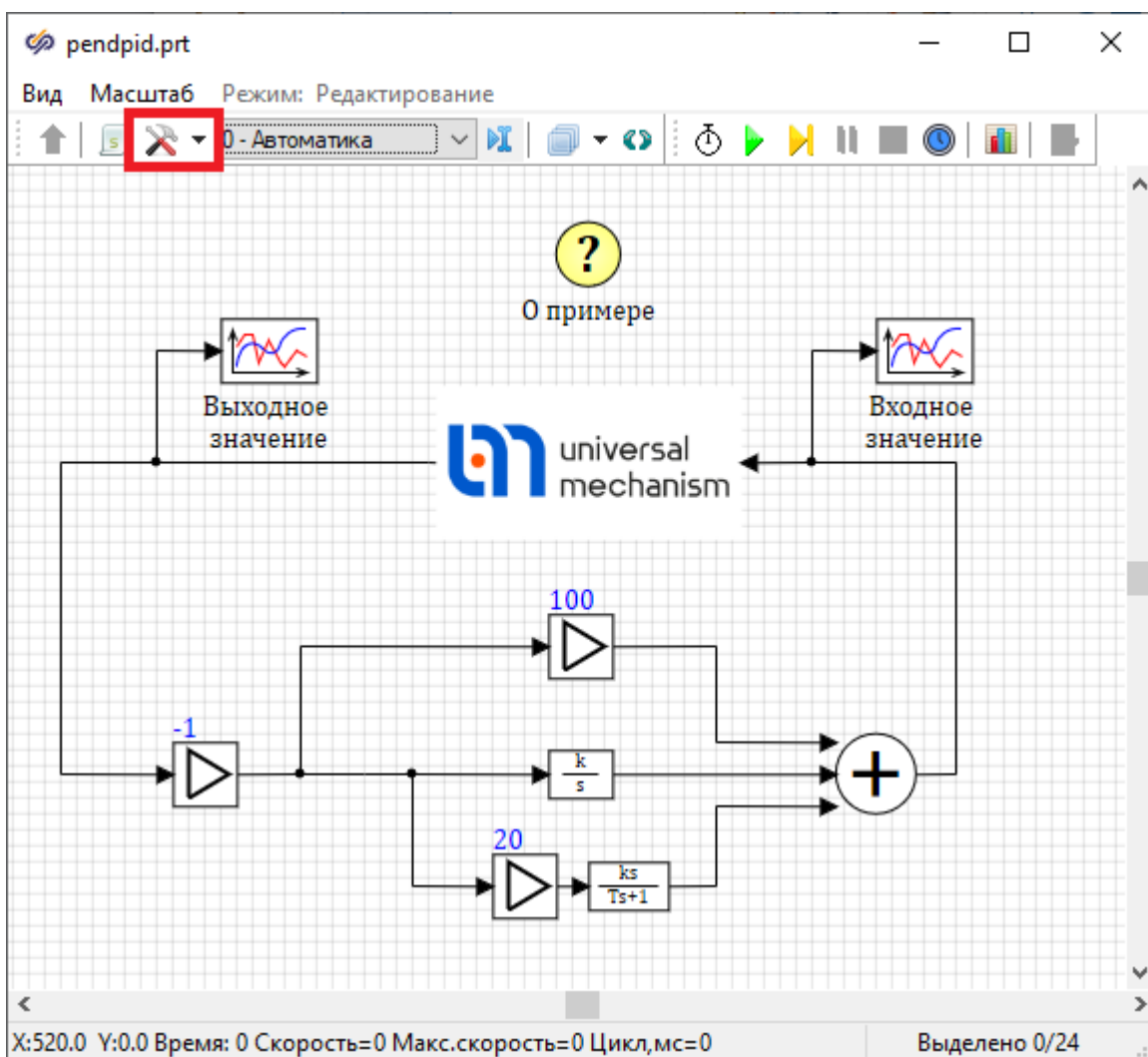
Закройте окно со свойствами блока.

#### 4.3.2.4. Моделирование движения

Модель подготовлена для дальнейшего анализа. Сейчас мы установим параметры моделирования, а затем запустим совместное моделирование.

##### Установка параметров моделирования

В окне модели нажмите на кнопку **Параметры расчета**.



На вкладке **Параметры расчета** в поле **Минимальный шаг** установите **0.00001**, в поле **Максимальный шаг** установите **0.001**.

Закройте окно **Параметры расчёта**.

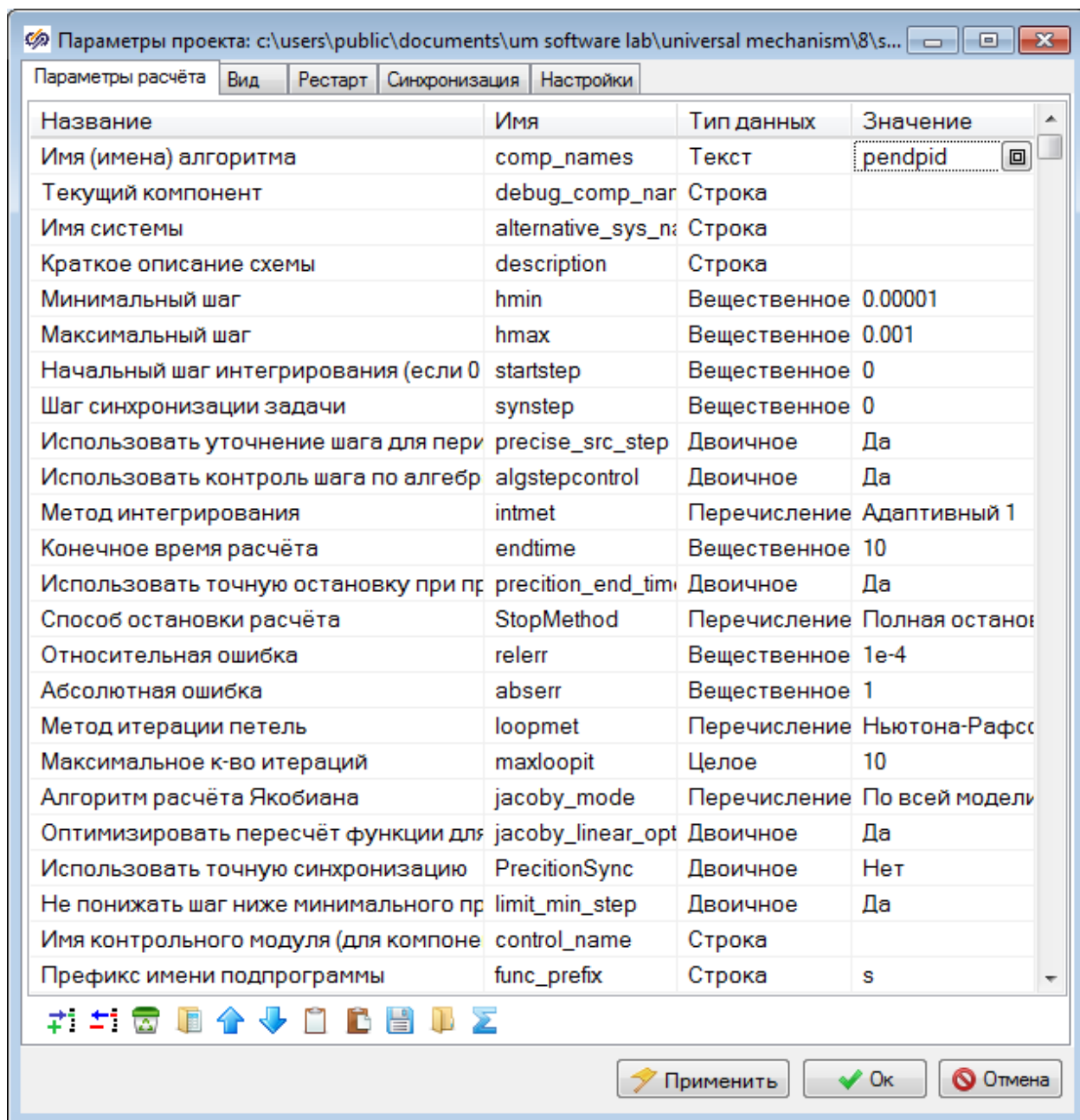



Рис. 4.12. Параметры конфигурации модели

**Моделирование динамики системы**

1. Двойным щелчком мыши откройте окна **Временной график**, в которых будут отображаться угол отклонения маятника от вертикали и величина усилия, подаваемого на тележку.
2. Кнопкой **Пуск**  запустите моделирование.
3. По истечении времени моделирования вы увидите результаты, приведенные на рис. 4.13 и рис. 4.14.

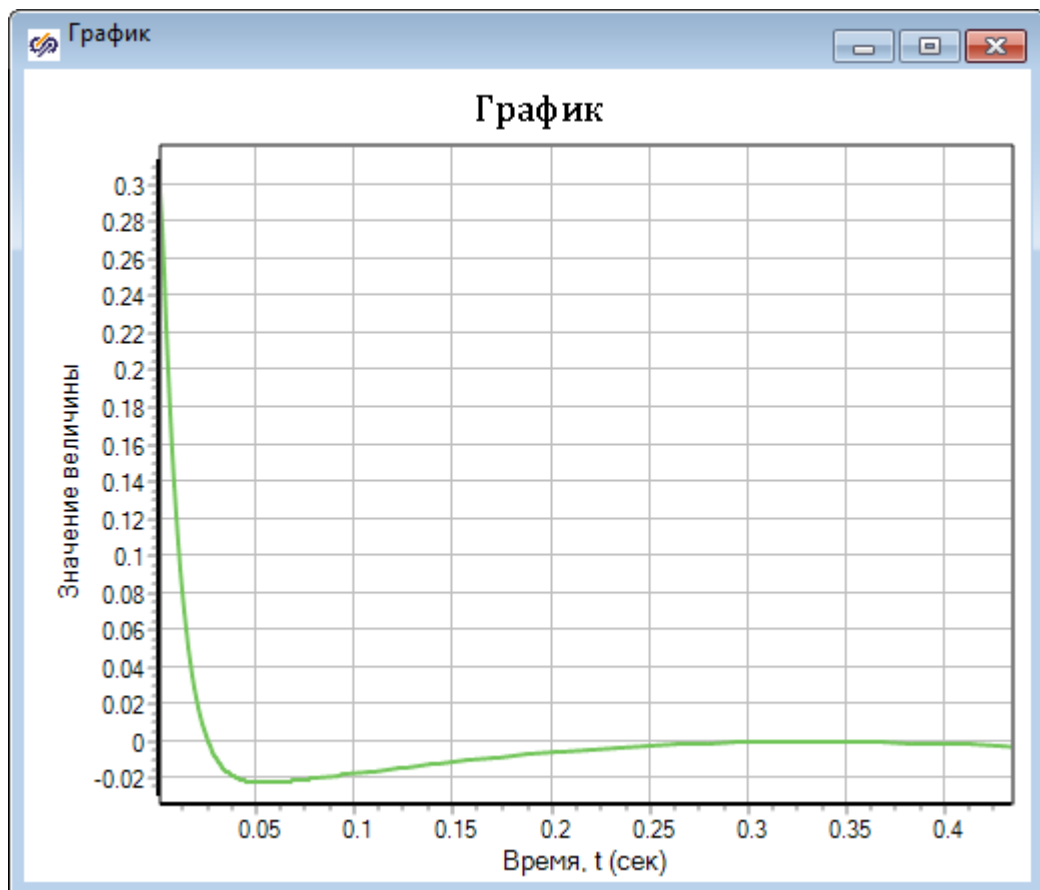


Рис. 4.13. График изменения угла отклонения маятника от вертикали

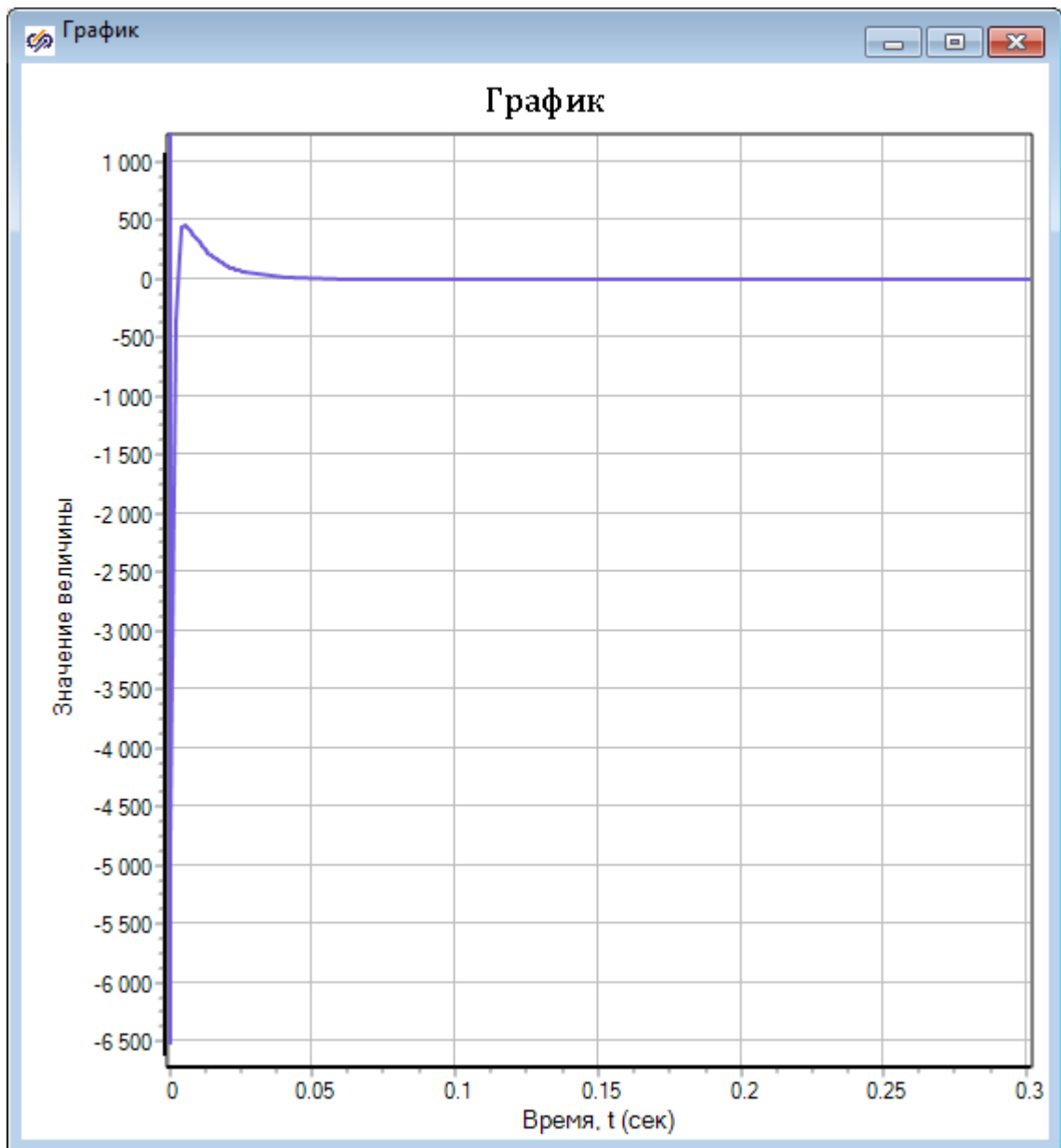


Рис. 4.14. График изменения величины усилия, подаваемого на тележку

## 4.4. Связь с SimInTech по UDP протоколу

### 4.4.1. Подключение расчёта силы в демпфере

#### Загрузка модели механической части в Универсальный механизм

В качестве примера рассмотрим очень простую модель, которая раньше уже встречалась в наших уроках – модель груза на пружине, см. рис. 4.15. В этом примере мы будем вычислять усилие в гидравлическом демпфере в расчётной схеме SimInTech, а обмен данными будем осуществлять по UDP протоколу. Для этого скорость штока демпфера относительно его корпуса будем отправлять в расчётную схему SimInTech, а принимать будем рассчитанное усилие в этом демпфере. Для реализации такого взаимодействия модель в УМ немного переработана – усилие в демпфере задаётся напрямую с помощью идентификатора **damperforce**.

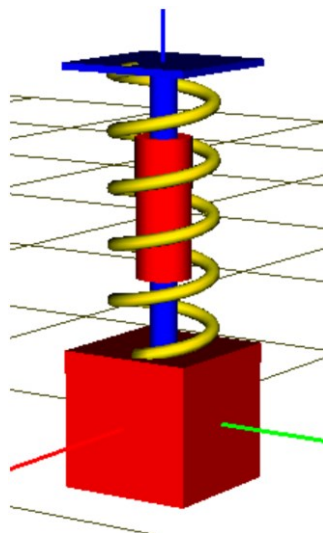


Рис. 4.15. Модель груза на пружине

1. Запустите программу моделирования **UM Simulation**.
2. Загрузите модель [{Данные УМ}\SAMPLES\udp\oscillator\\_udp](#).
3. Откройте окно **Мастера связи по UDP протоколу** с помощью пункта меню **Инструменты / Внешние интерфейсы / UDP интерфейс**, см. рис. 4.16. Настройки UDP интерфейса загрузились автоматически при загрузке модели из файла **last\_udp**. Модель в «Универсальном механизме» полностью настроена и готова к работе. Ниже рассмотрим основные особенности модели и подключения по UDP.

Как мы видим в описании UDP протокола на рис. 4.16, сигнал, который приходит в порт 50891, назначается идентификатору **damperforce**, который непосредственно и задаёт величину силы в демпфере, см. рис. 4.17. Диссипативную силу в расчётной схеме в SimInTech будем считать как произведение относительной скорости штока и цилиндра демпфера и коэффициента диссипации. Для этого будем отправлять в порт 50890 на нашем локальном компьютере скорость груза относительно точки подвеса, см. рис. 4.18. Ниже на рис. 4.19 показано, как в **Мастере переменных** была создана переменная для от-

правки в расчётную схему **SimInTech**. Эту переменную не нужно заново создавать и назначать. Обмен данными уже сконфигурирован в модели, рис. 4.19 приводится только для иллюстрации того, как это было сделано.

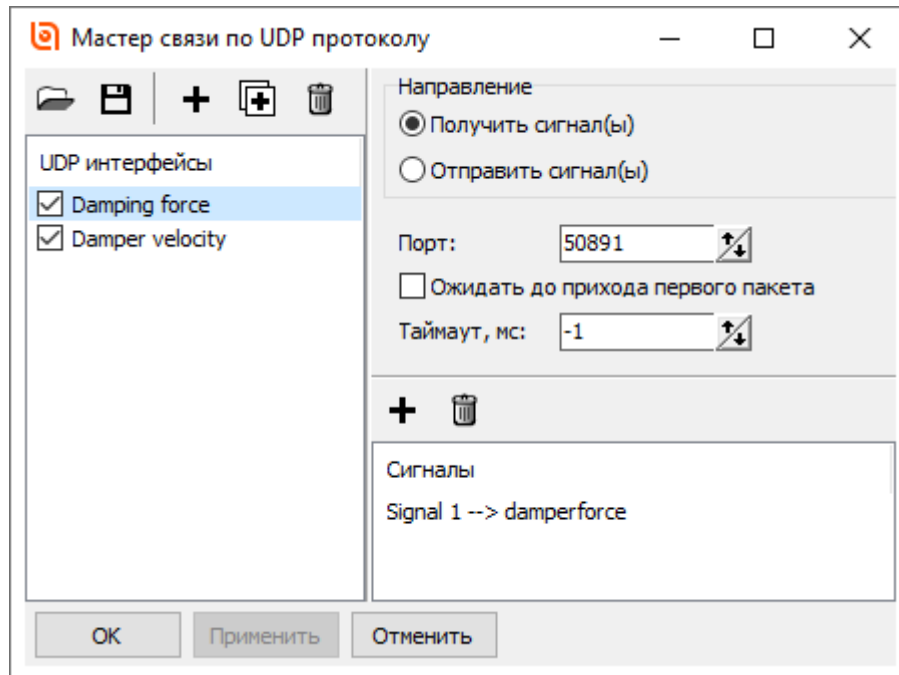


Рис. 4.16. Описание входных сигналов в **Мастере связи по UDP протоколу**

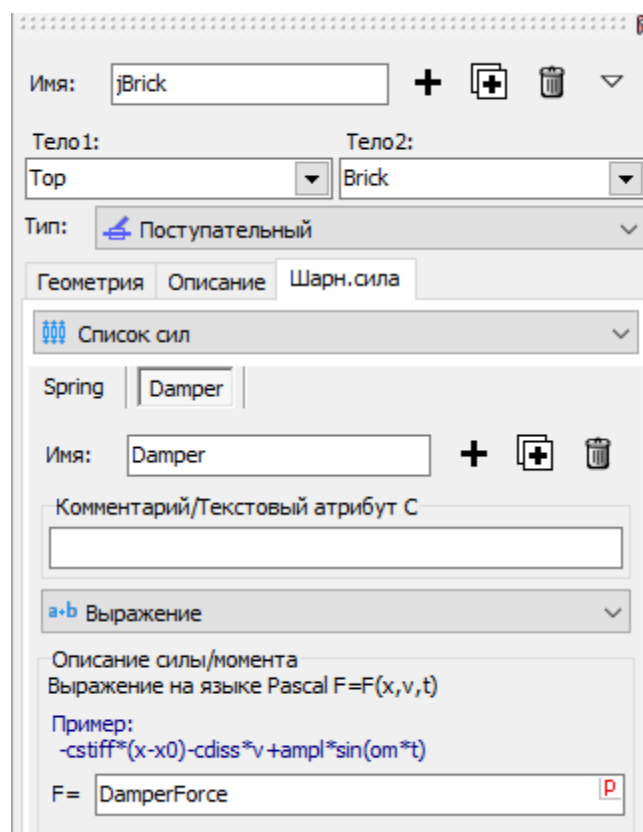


Рис. 4.17. Описание диссипативной силы в программе **UM Input**

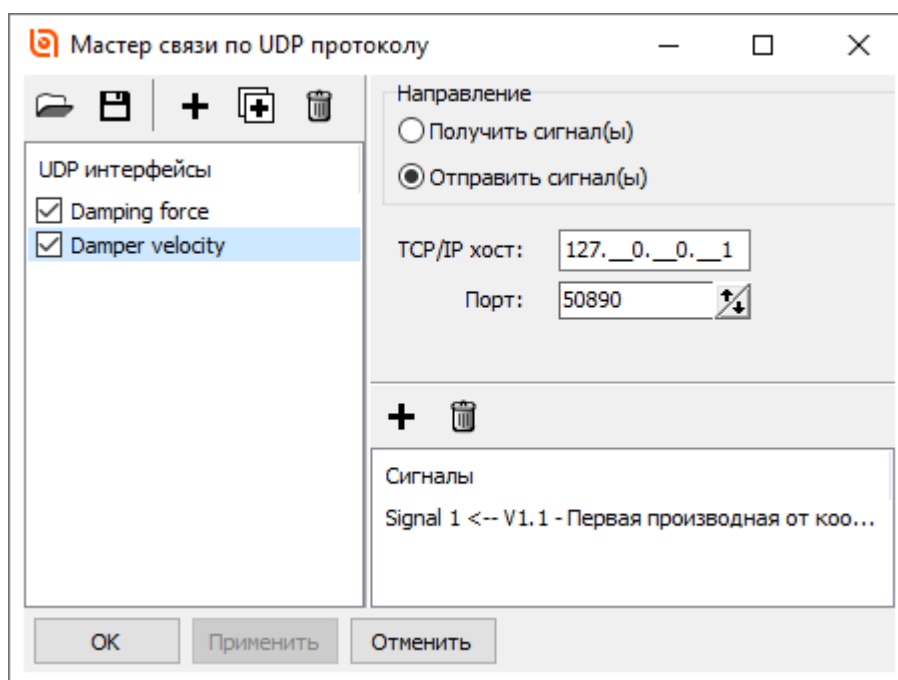


Рис. 4.18. Описание выходных сигналов в **Мастере связи по UDP протоколу**

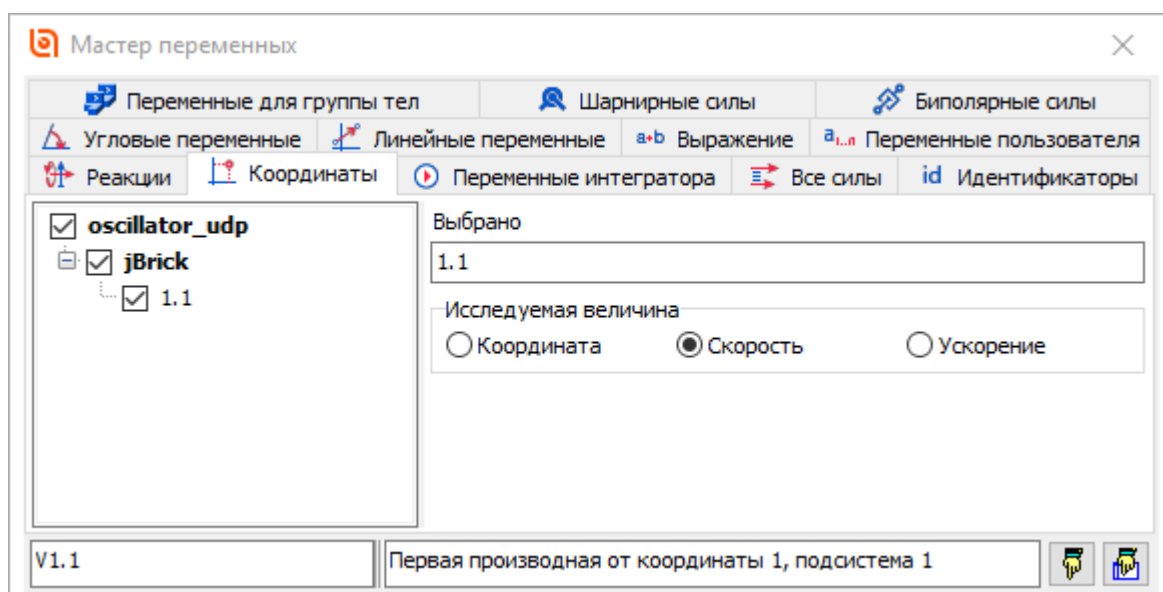


Рис. 4.19. Создание переменной для отправки по UDP протоколу в **Мастере переменных**

### Загрузка расчётной схемы в SimInTech

4. Запустите **SimInTech**.
5. Из каталога [{Данные УМ}\SAMPLES\udp\oscillator\\_udp](#) загрузите файл `damper_udp_rus.prt`, см. рис. 4.20. Обмен данными по протоколу UDP в SimInTech поддерживается блоками **Сервер UDP** и **Клиент UDP** с вкладки **Обмен данными**, см. рис. 4.21

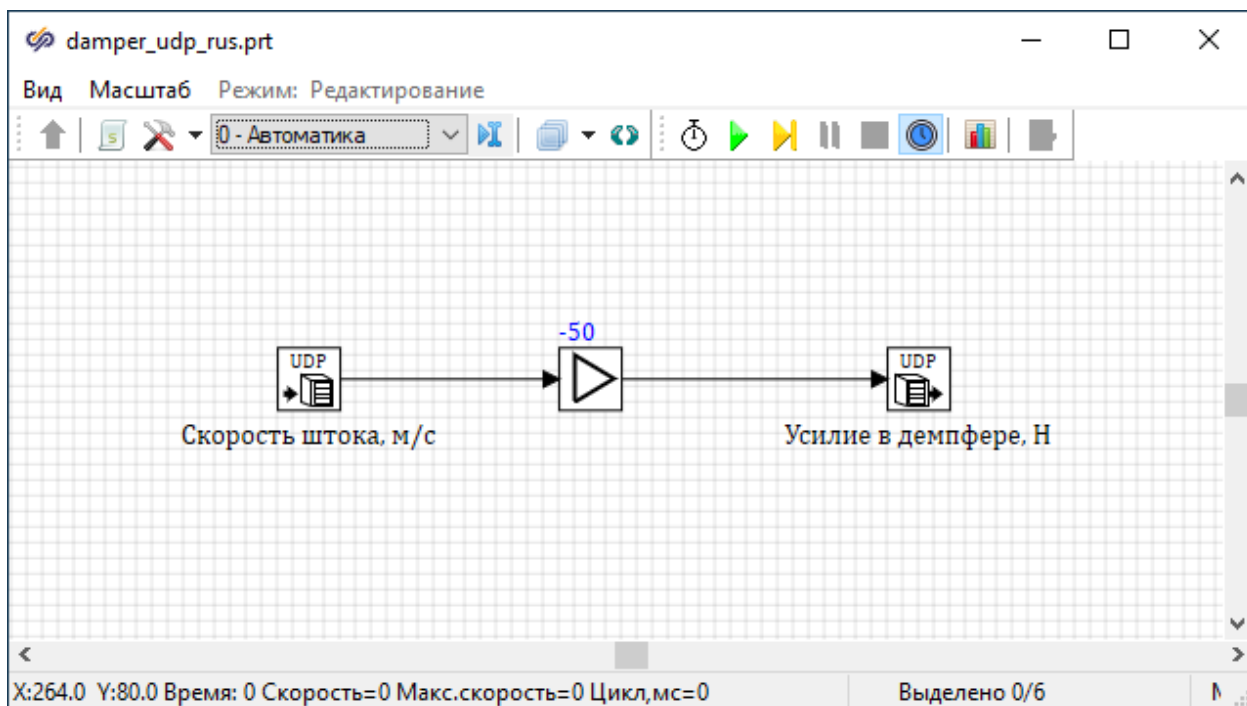


Рис. 4.20. Расчетная схема в SimInTech

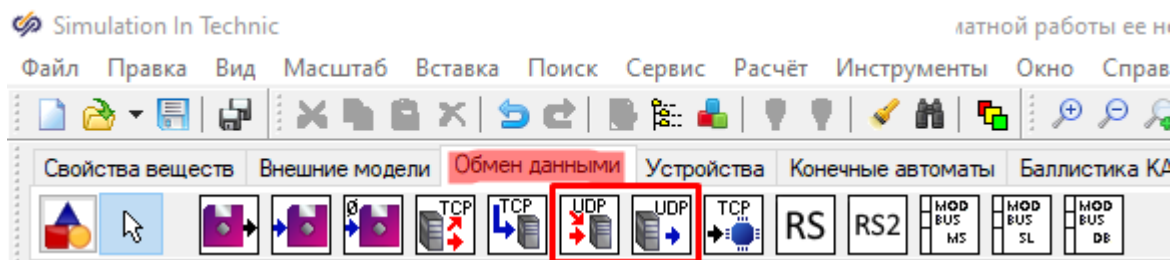


Рис. 4.21. Блоки работы с UDP протоколом в SimInTech

### Запуск совместного моделирования

6. Запустите моделирование в **SimInTech** кнопкой **Пуск**.

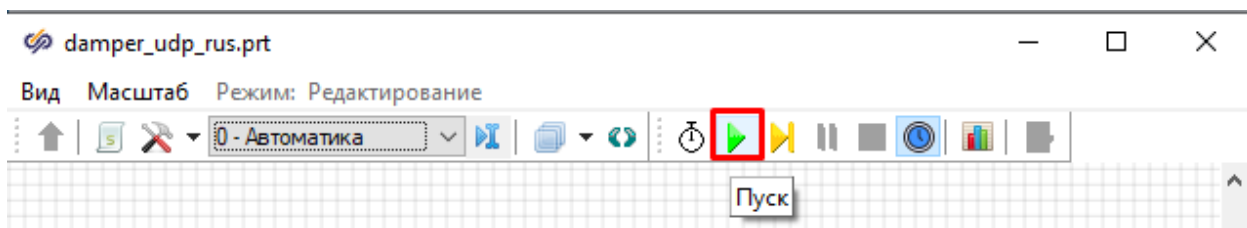


Рис. 4.22. Запуск расчётной схемы в SimInTech

7. Перейдите в «Универсальный механизм», с помощью пункта меню **Анализ / Моделирование** вызовите **Инспектор моделирования объекта**. Далее для запуска расчётов нажмите кнопку **Интегрирование** на вкладке **Интегратор**. По истечении 10 секунд расчёт в «Универсальном механизме» завершится. После этого остановите и расчётную схему в SimInTech.

При совместном моделировании вы увидите затухающие колебания груза на пружине. Проведите несколько численных экспериментов с другими значениями коэффициента диссипации, заданным коэффициентом усиления в расчётной схеме в SimInTech.

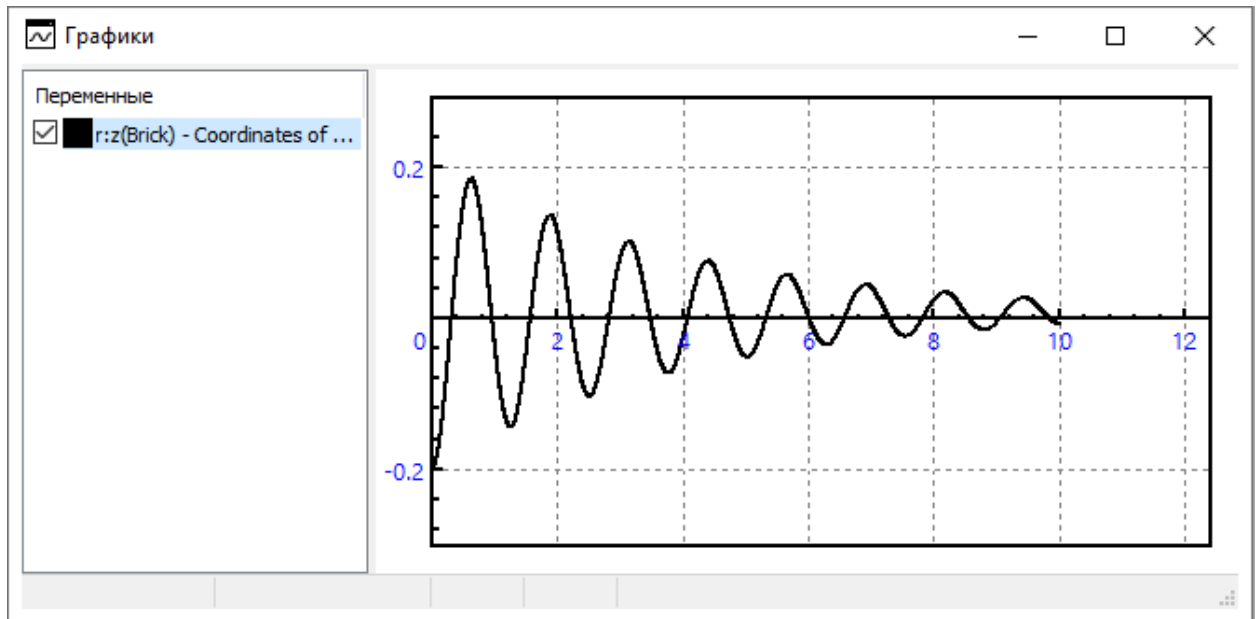


Рис. 4.23. Затухающие колебания как результат совместного моделирования

## 4.4.2. Перевернутый маятник

В этом параграфе рассмотрим подключение к УМ системы управления для стабилизации перевернутого маятника в неустойчивом положении равновесия с помощью обмена данными по UDP протоколу. Описание модели механической части дано в п. 2.1 «Перевернутый маятник», стр. 11.

### Загрузка модели механической части в Универсальный механизм

1. Запустите программу моделирования **UM Simulation**.
2. Загрузите модель `{Данные УМ}\SAMPLES\udp\inv_pend_udp`.
3. Откройте окно **Мастера связи по UDP протоколу** с помощью пункта меню **Инструменты / Внешние интерфейсы / UDP интерфейс**, см. рис. 4.24. Настройки UDP интерфейса загрузились автоматически при загрузке модели из файла last.udp. Модель в «Универсальном механизме» полностью настроена и готова к работе. Ниже рассмотрим основные особенности модели и подключения по UDP.

Как мы видим в описании UDP протокола на рис. 4.24, угол отклонения маятника отправляется в порт 50890, а управляющая сила приходит в порт 50891, см. рис. 4.25.

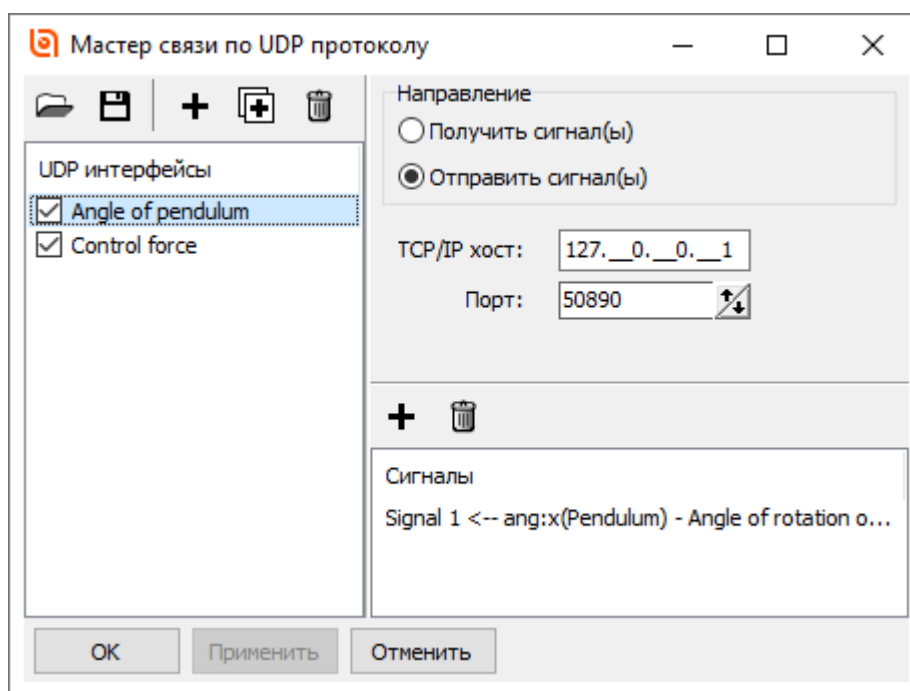


Рис. 4.24. Описание выходных сигналов в **Мастере связи по UDP протоколу**

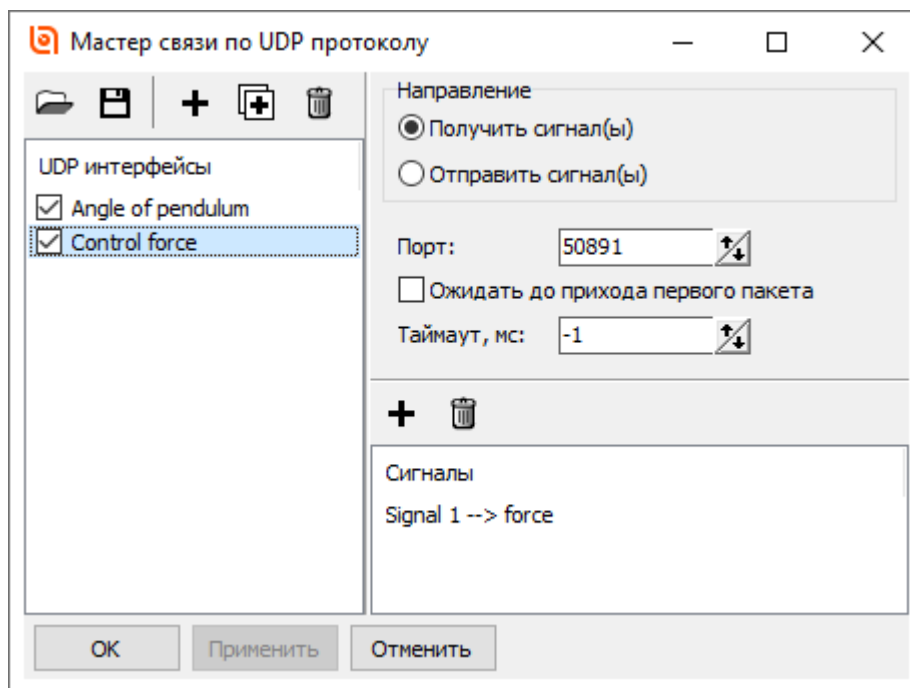


Рис. 4.25. Описание входных сигналов в **Мастере связи по UDP протоколу**

### Загрузка расчётной схемы в SimInTech

4. Запустите **SimInTech**.
5. Из каталога [{Данные УМ}\SAMPLES\udp\inv\\_pend\\_udp](#) загрузите файл `pid_udp_rus.prt`, см. рис. 4.26.

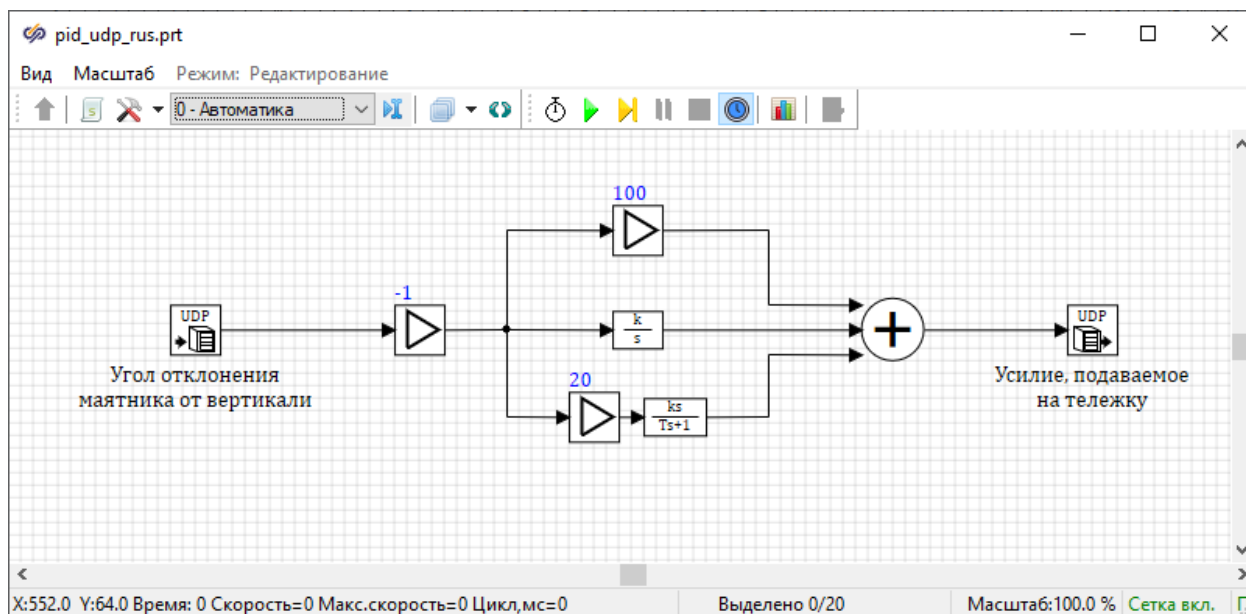


Рис. 4.26. Расчетная схема в SimInTech

### Запуск совместного моделирования

6. Запустите моделирование в **SimInTech** кнопкой **Пуск**.

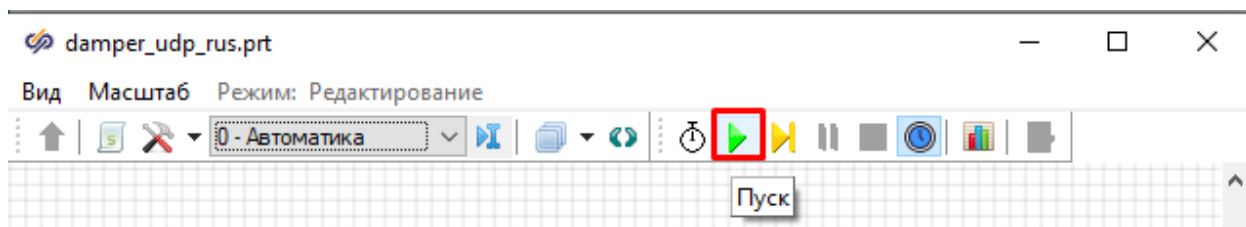


Рис. 4.27. Запуск расчётной схемы в SimInTech

7. Перейдите в «Универсальный механизм», с помощью пункта меню **Анализ / Моделирование** вызовите **Инспектор моделирования объекта**. Далее для запуска расчётов нажмите кнопку **Интегрирование** на вкладке **Интегратор**. По истечении 3 секунд расчёт в «Универсальном механизме» завершится. После этого остановите и расчётную схему в SimInTech.

При совместном моделировании вы увидите стабилизацию перевернутого маятника. Будут построены графики управляющей силы и угла отклонения маятника от вертикали. Обратите внимание, что в начале моделирования идентификатор **force** будет иметь значение, которое было назначено ему последним в предыдущем сеансе моделирования. При повторении расчётов перейдите на вкладку **Идентификаторы / Список идентификаторов Инспектора моделирования объекта** и установите значение управляющей силы **force = 0**.

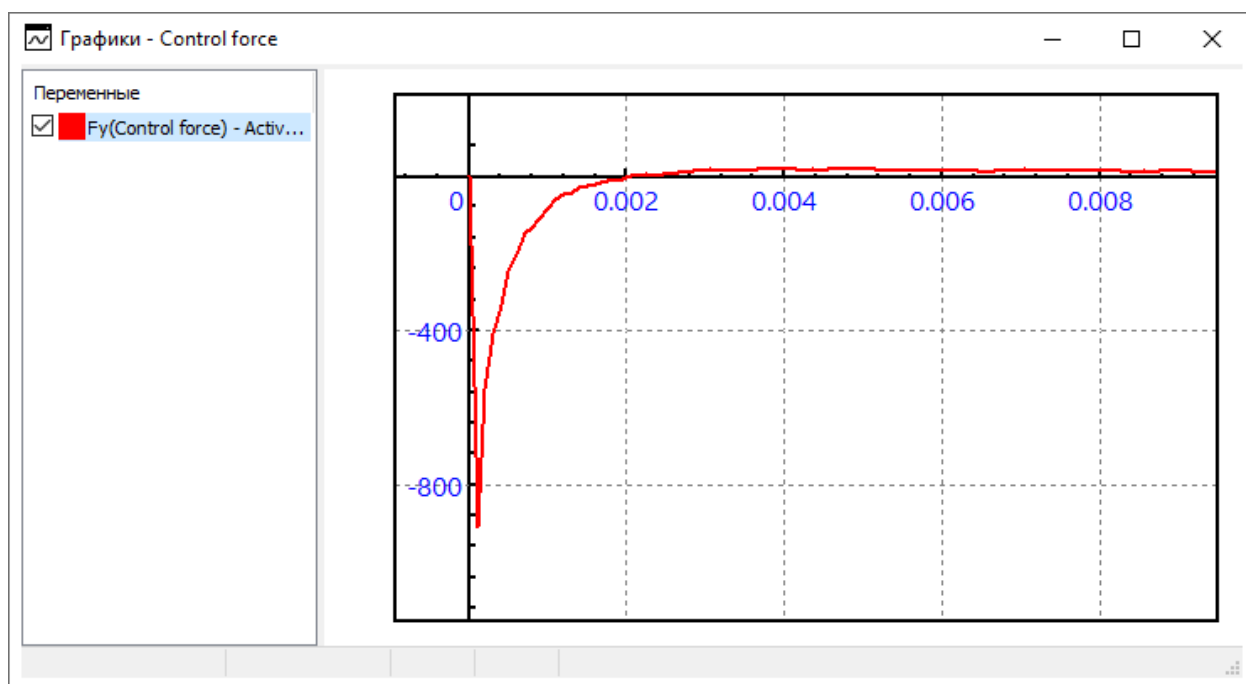


Рис. 4.28. Стартовый бросок управляющей силы

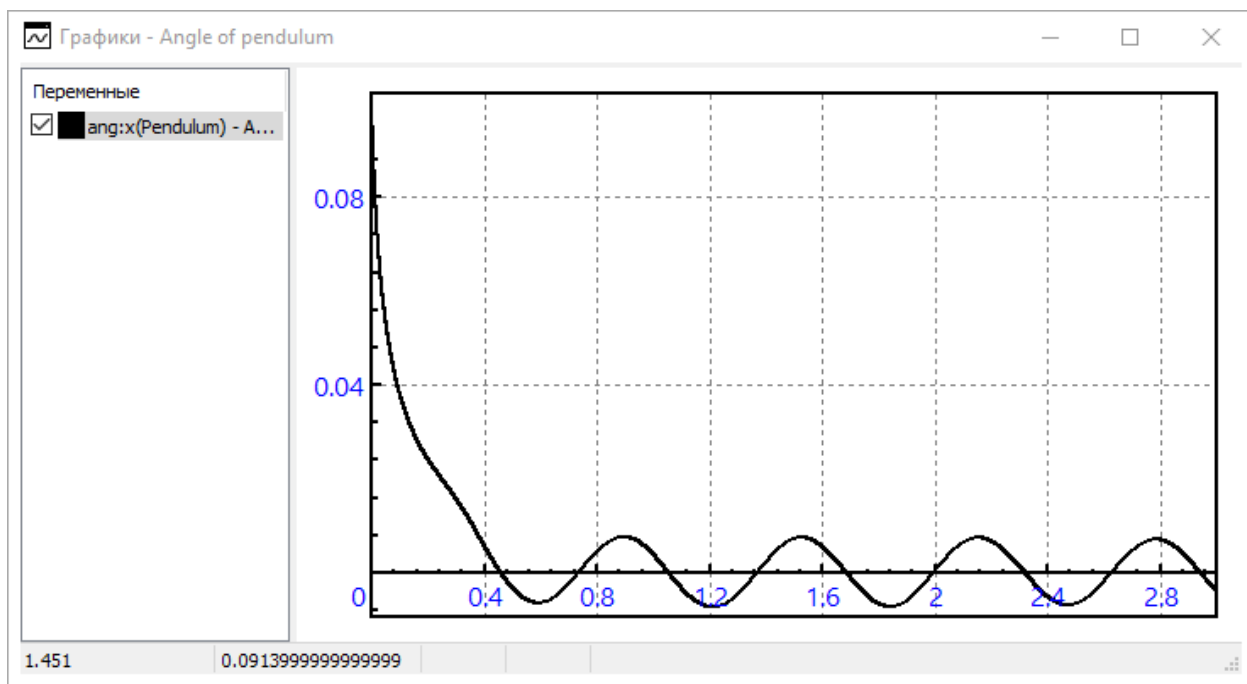


Рис. 4.29. Угол отклонения маятника от вертикали