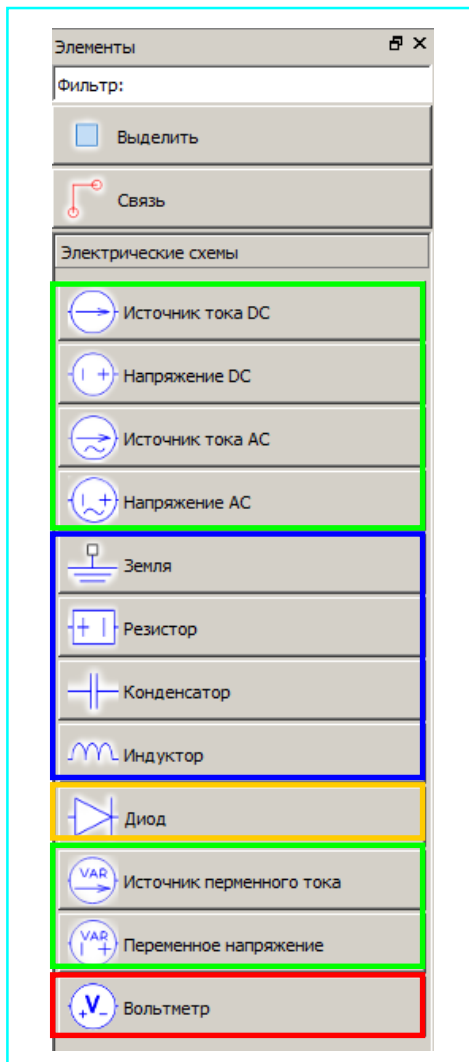




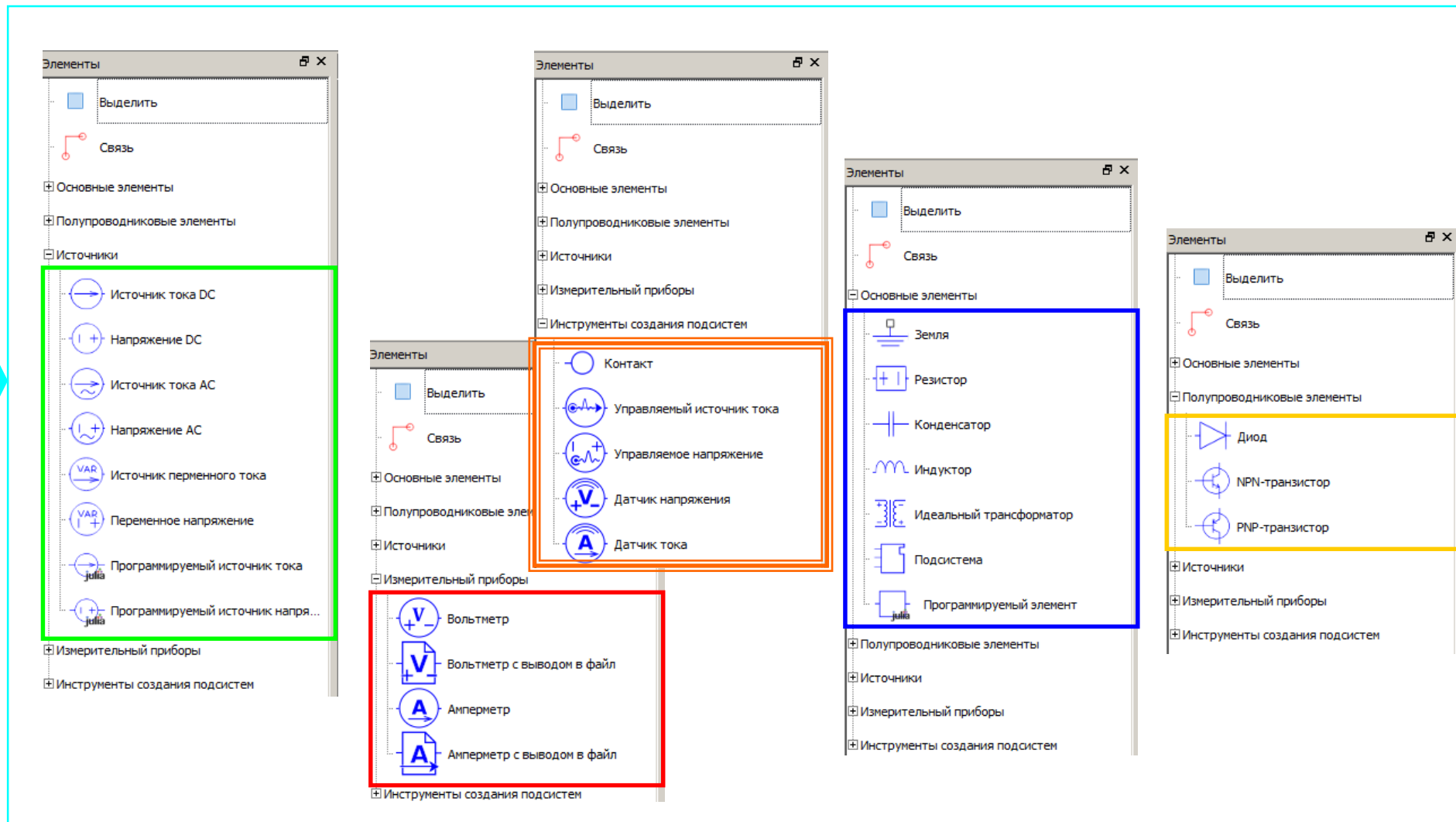
**От подсистемы к системе! АРМ ЕСА –
еще один шаг к построению систем
автоматизированного управления!**

- **Расширение элементной базы**
- Параметризация
- Программируемые пользовательские элементы
- Работа с подсистемами
- Имитационные схемы
- Взаимосвязь между электрическими и имитационными схемами
- Автоматические системы управления

APM ECA v15

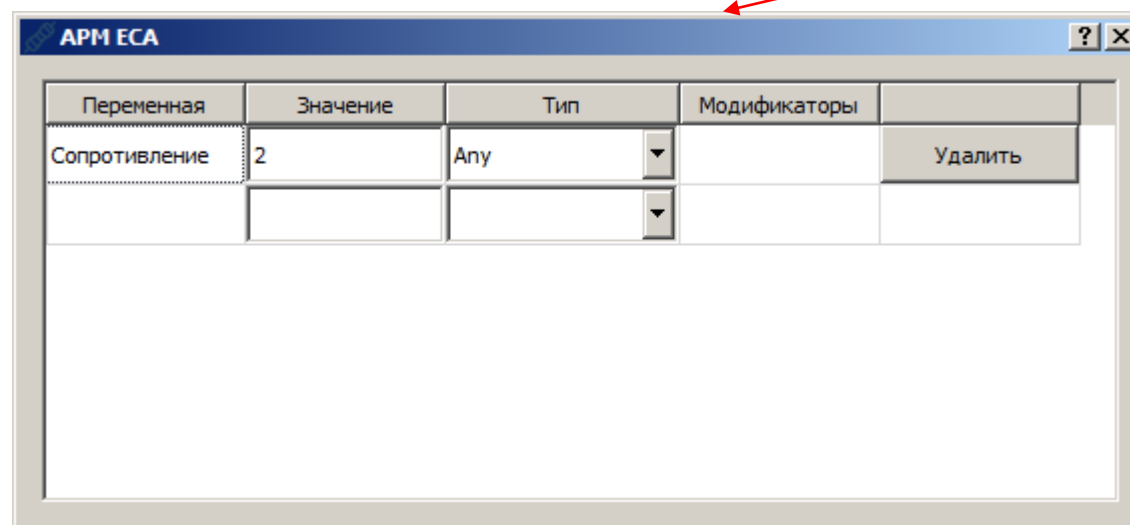
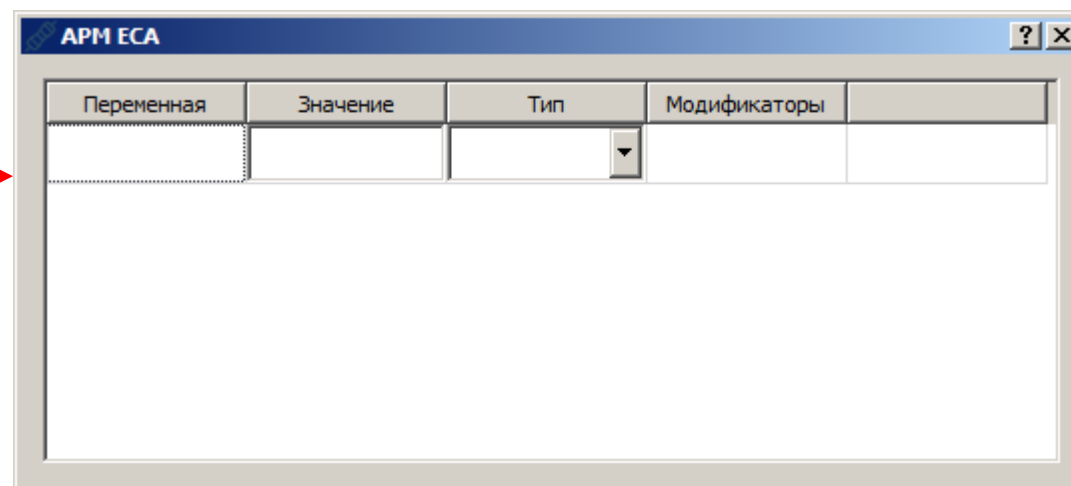
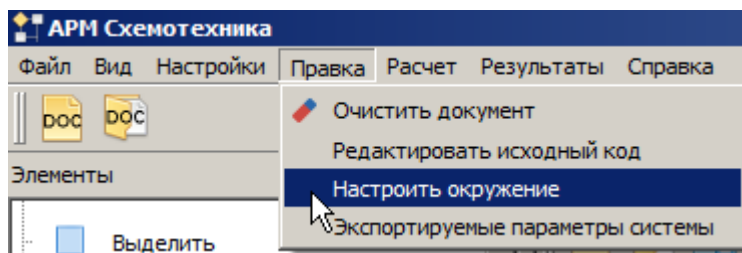


APM ECA v16



- Расширение элементной базы
- **Параметризация**
- Программируемые пользовательские элементы
- Работа с подсистемами
- Имитационные схемы
- Взаимосвязь между электрическими и имитационными схемами
- Автоматические системы управления

Создание переменной



Использование в значениях математических операций и значений других переменных

Переменная	Значение	Тип	Модификаторы	
Сопротивление	2	Any		Удалить
Сопротивление1	$\sin(\pi/3) + 25^2 \cdot \sqrt{3}$	Any		Удалить
X	Сопротивление + 5 * Сопротивление1	Any		Удалить
Y	X * Z	Any		Удалить
Z	Y	Any		Удалить

Ошибка!!!

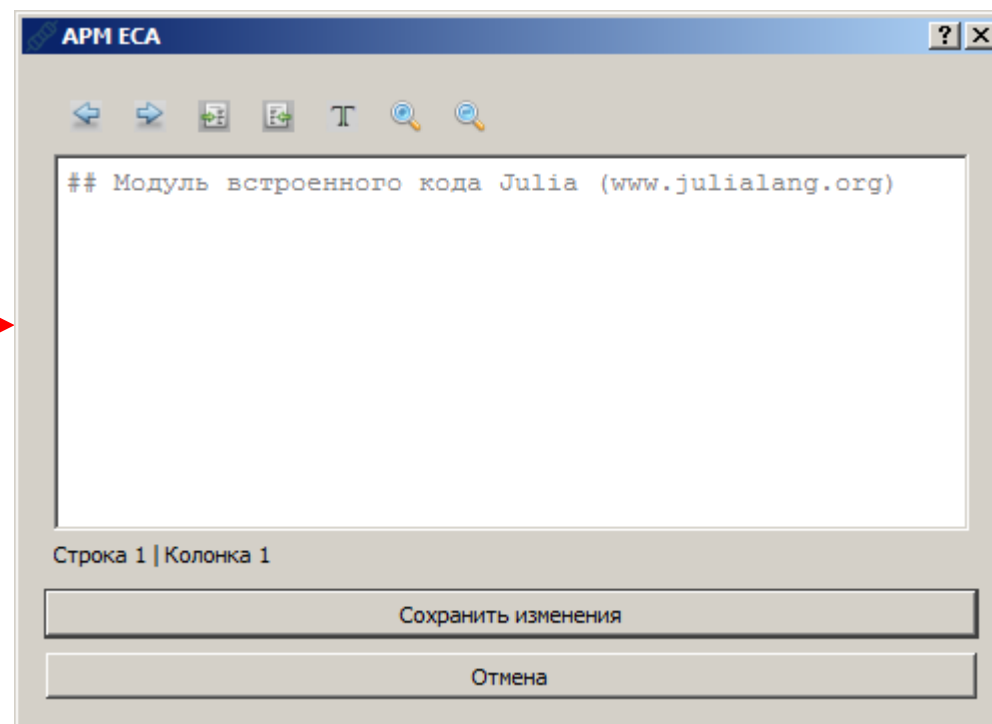
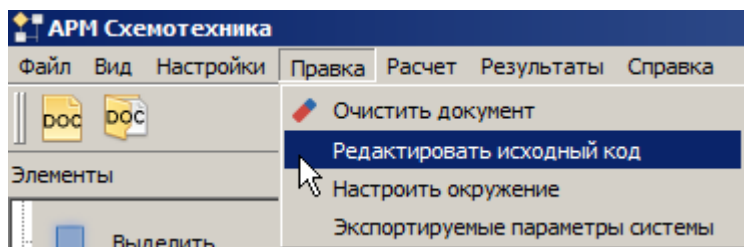
Переменная Y ссылается на переменную Z,
а переменная Z ссылается на переменную Y

- Расширение элементной базы
- Параметризация
- **Программируемые пользовательские элементы**
- Работа с подсистемами
- Имитационные схемы
- Взаимосвязь между электрическими и имитационными схемами
- Автоматические системы управления

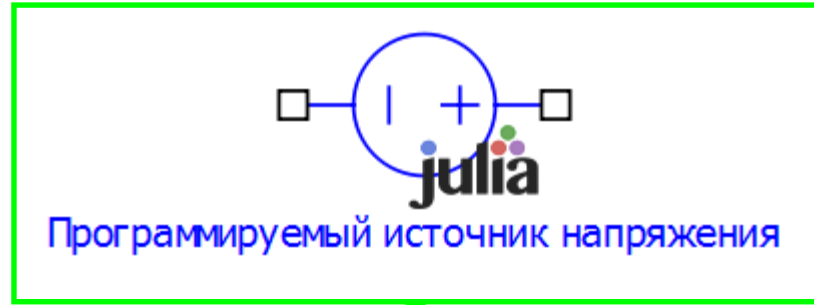
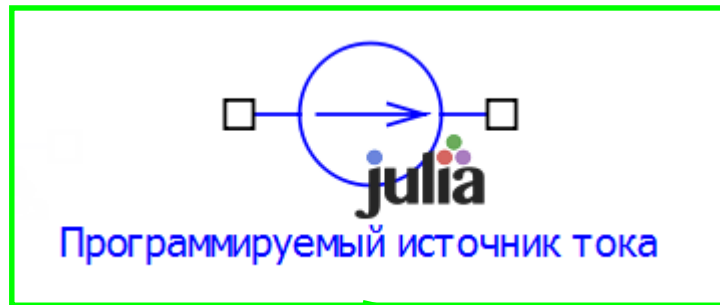
Программирование пользовательских элементов осуществляется на языке Julia

Julia – высокоуровневый, высокопроизводительный, динамический язык программирования
для научных и инженерных расчетов

Julia – Free, Open Source, Library-Friendly Software (www.julialang.org)



Типы программируемых элементов



Свойства элемента

Название вызываемой функции:

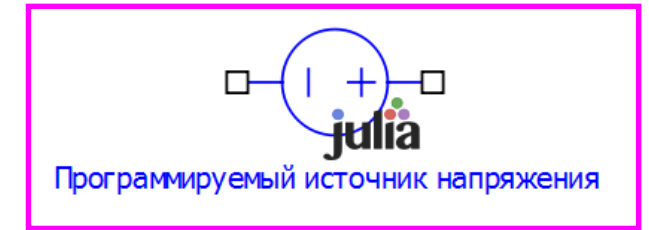
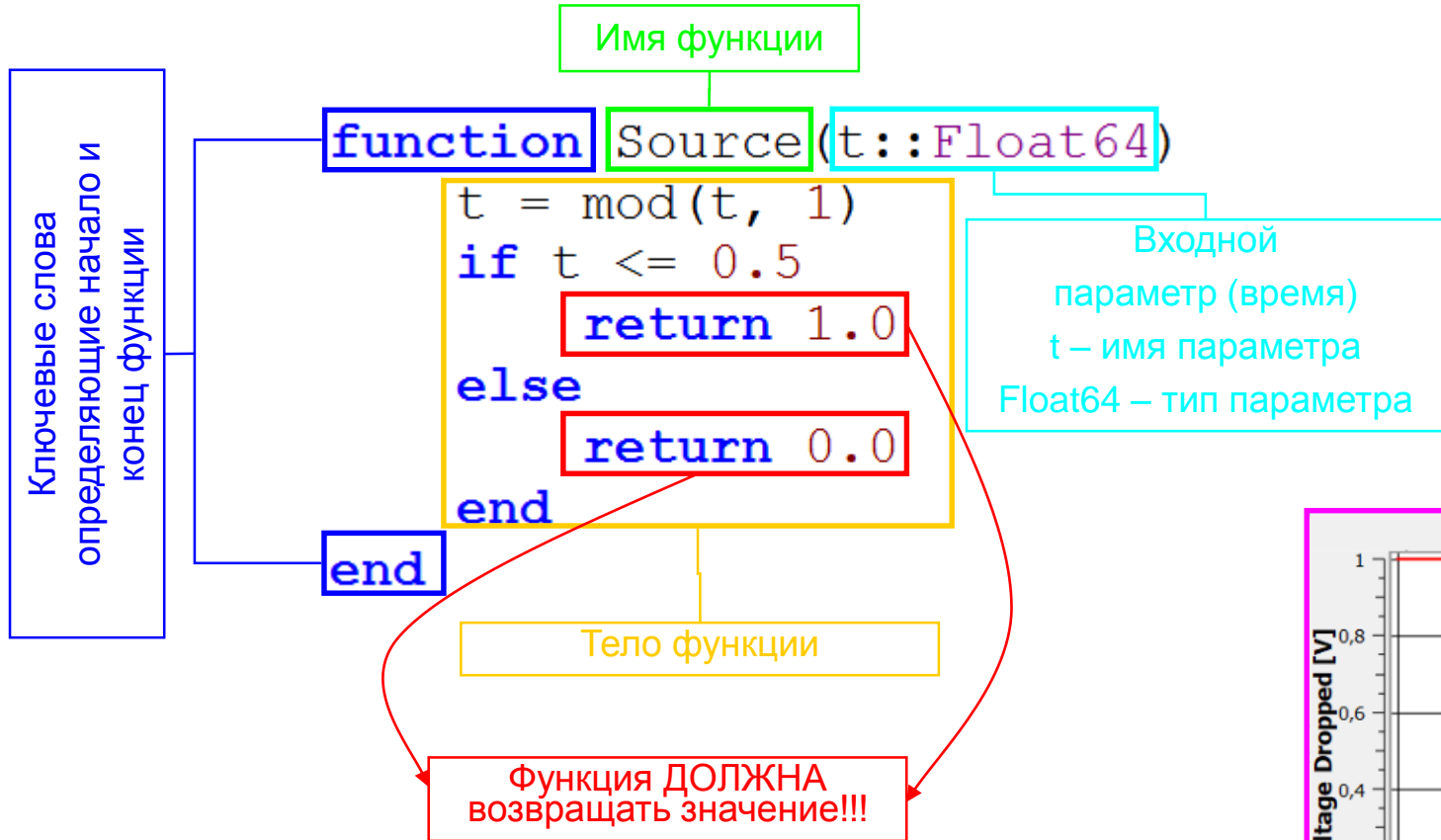
Функция

Свойства элемента

Тип элемента:

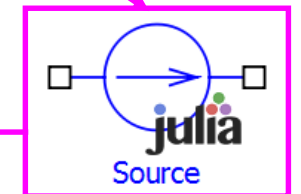
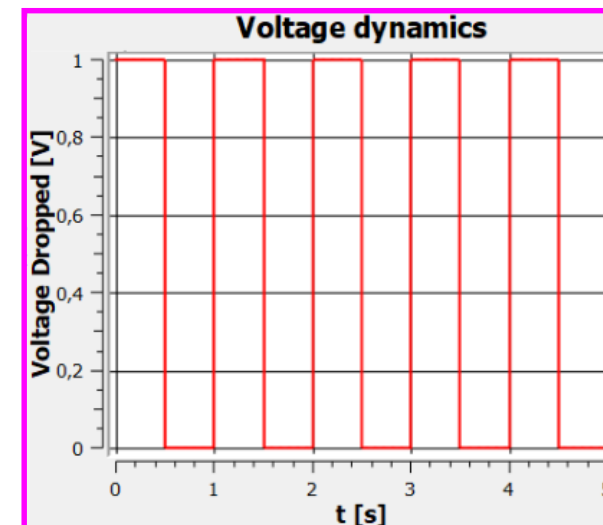
Элемент

Функция на языке Julia

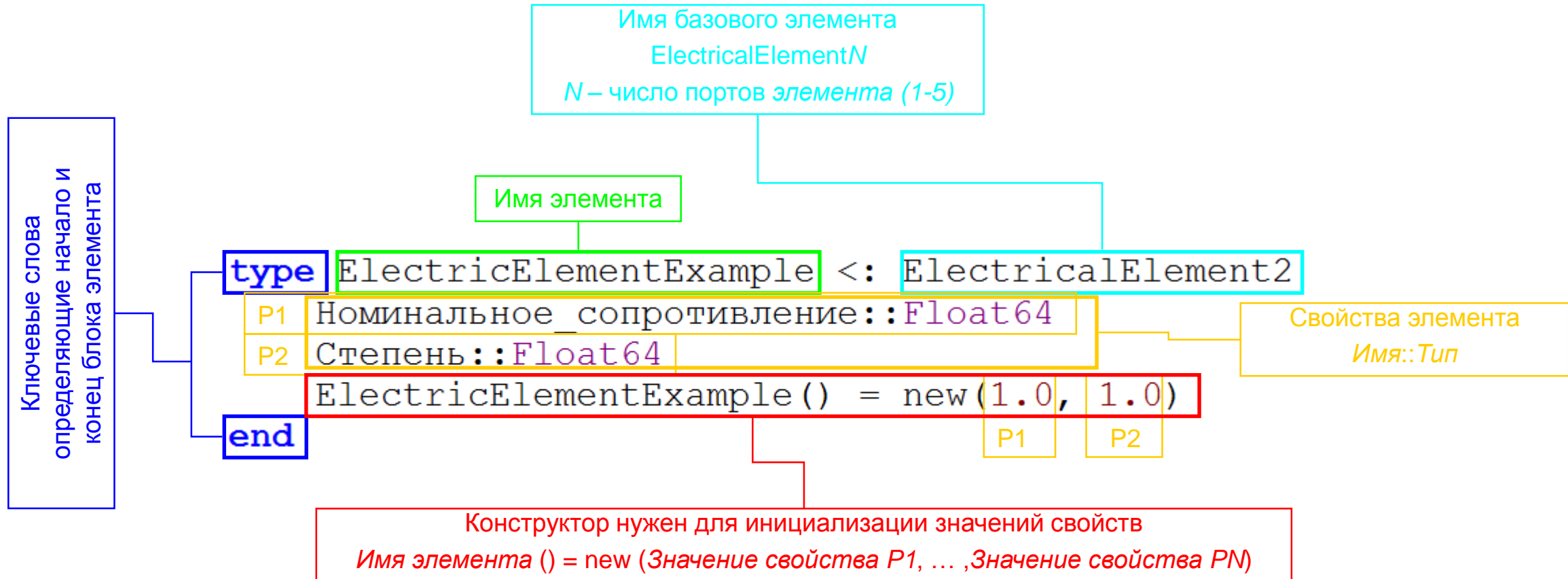


Свойства элемента

Название вызываемой функции:



Элемент на языке Julia



Элемент на языке Julia

Входной массив узловых потенциалов

q – имя массива

Array – указывает что входной параметр массив

Float64 – тип элементов массива

Размер массива N – число портов элемента

Элемент

Имя элемента

```
function (e::ElectricElementExample) (q::Array{Float64})
    U = q[1] - q[2]
    R = (abs(U) ^ e.Степень + 1) * e.Номинальное_сопротивление
    I = U / R
    return [-I; I]
end
```

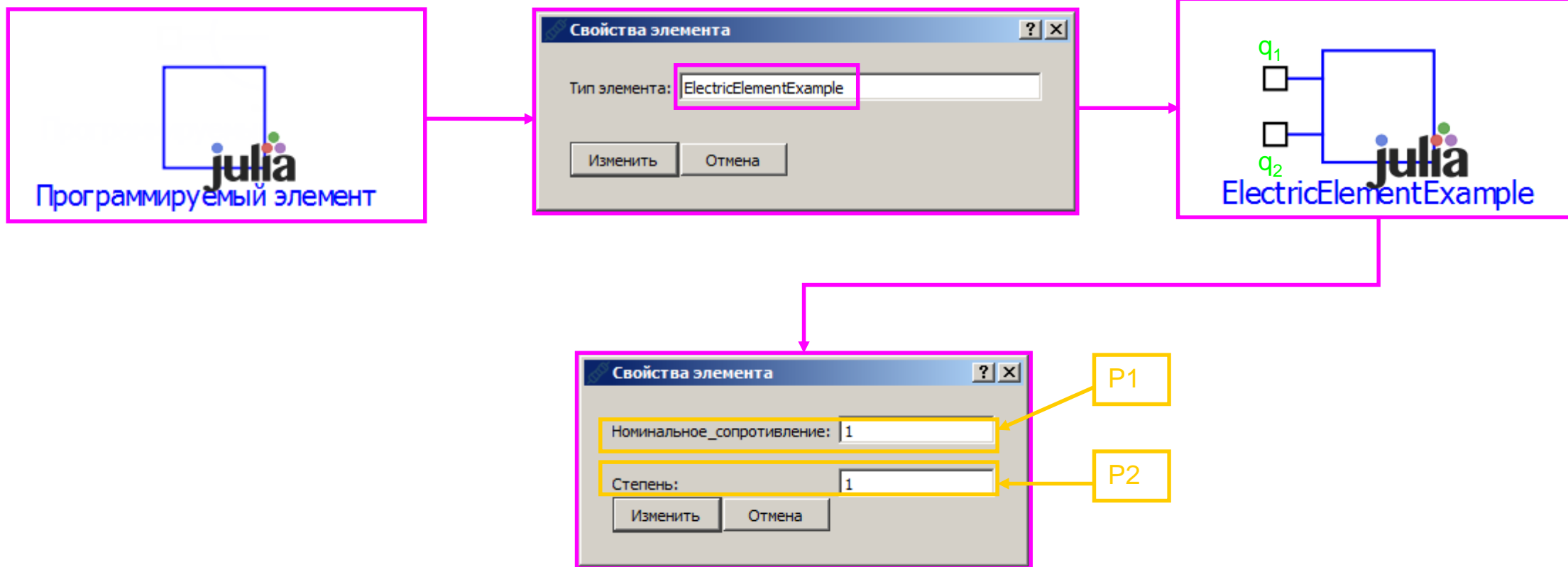
Функция ДОЛЖНА возвращать массив значений токов в портах элемента соответствующих элементам массива узловых потенциалов!!!

Обращение к свойствам элемента
Элемент.Свойство элемента

Ключевые слова
определяющие начало и
конец функции

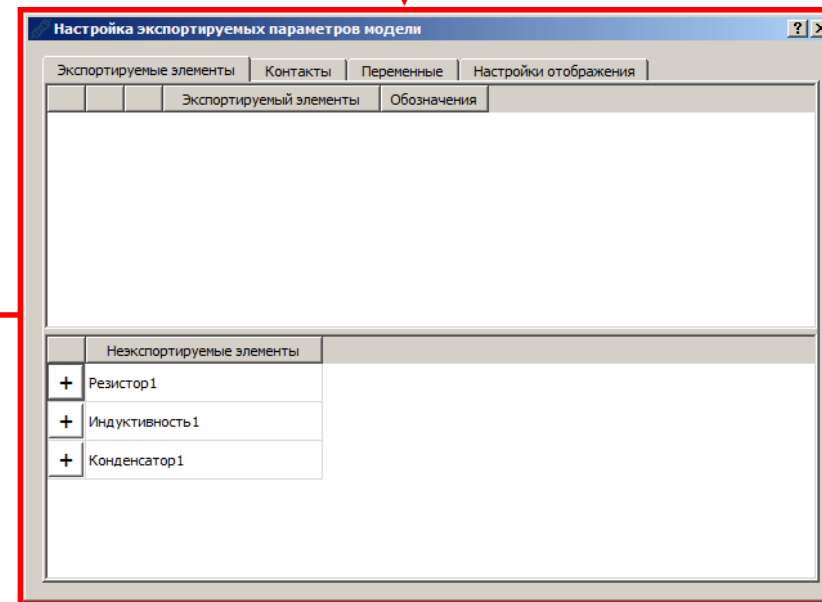
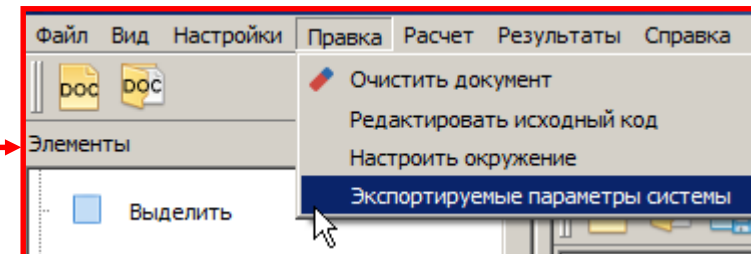
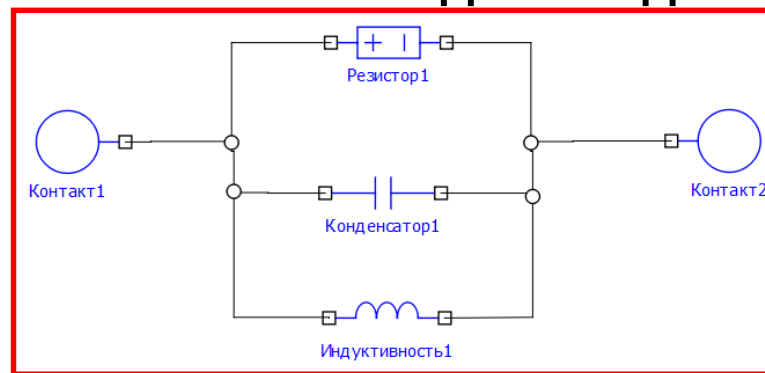
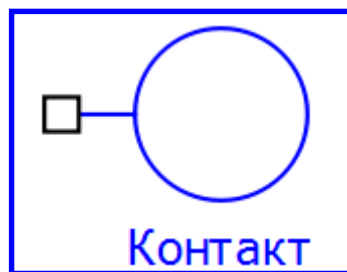
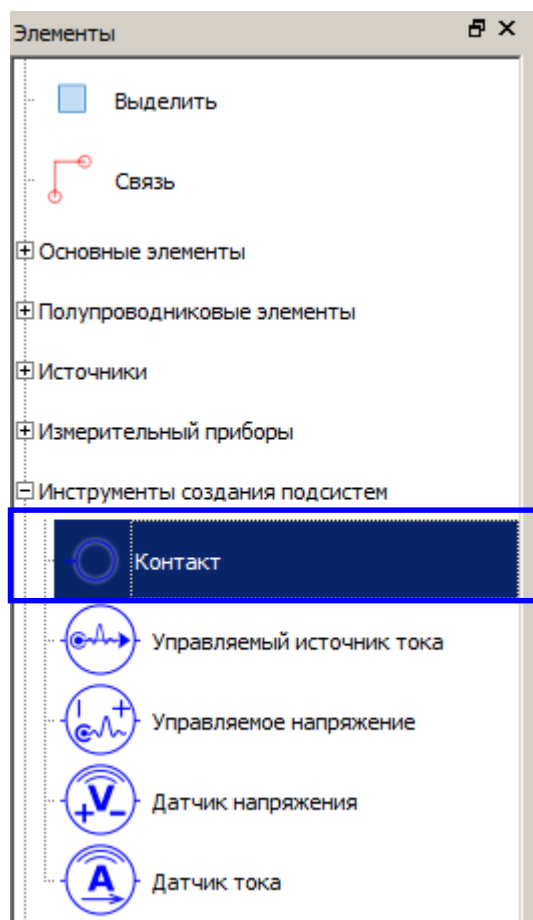
Вспомогательные
вычисления

Элемент на языке Julia

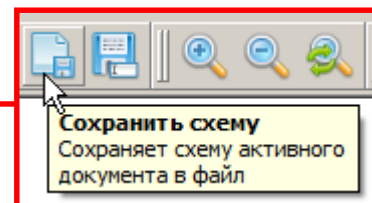


- Расширение элементной базы
- Параметризация
- Программируемые пользовательские элементы
- **Работа с подсистемами**
- Имитационные схемы
- Взаимосвязь между электрическими и имитационными схемами
- Автоматические системы управления

Контакт – основной элемент для создания подсистем



**ПОДСИСТЕМА
ГОТОВА К
ИСПОЛЬЗОВАНИЮ!!!**



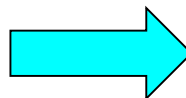
Настройка экспортируемых элементов

Настройка экспортируемых параметров модели

Экспортируемые элементы | Контакты | Переменные | Настройки отображения

	Экспортируемый элемент	Обозначения
--	------------------------	-------------

	Неэкспортируемые элементы
+	Конденсатор1
+	Резистор1
+	Индуктивность1



Настройка экспортируемых параметров модели

Экспортируемые элементы | Контакты | Переменные | Настройки отображения

		Экспортируемый элемент	Обозначения
	-	Резистор1	R1
	-	Индуктивность1	L1

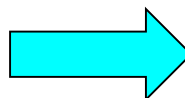
	Неэкспортируемые элементы
+	Конденсатор1

Настройка контактов

Настройка экспортируемых параметров модели

Экспортируемые элементы | Контакты | Переменные | Настройки отображения

		Метка	Положение
1		Контакт1	слева
2		Контакт2	слева



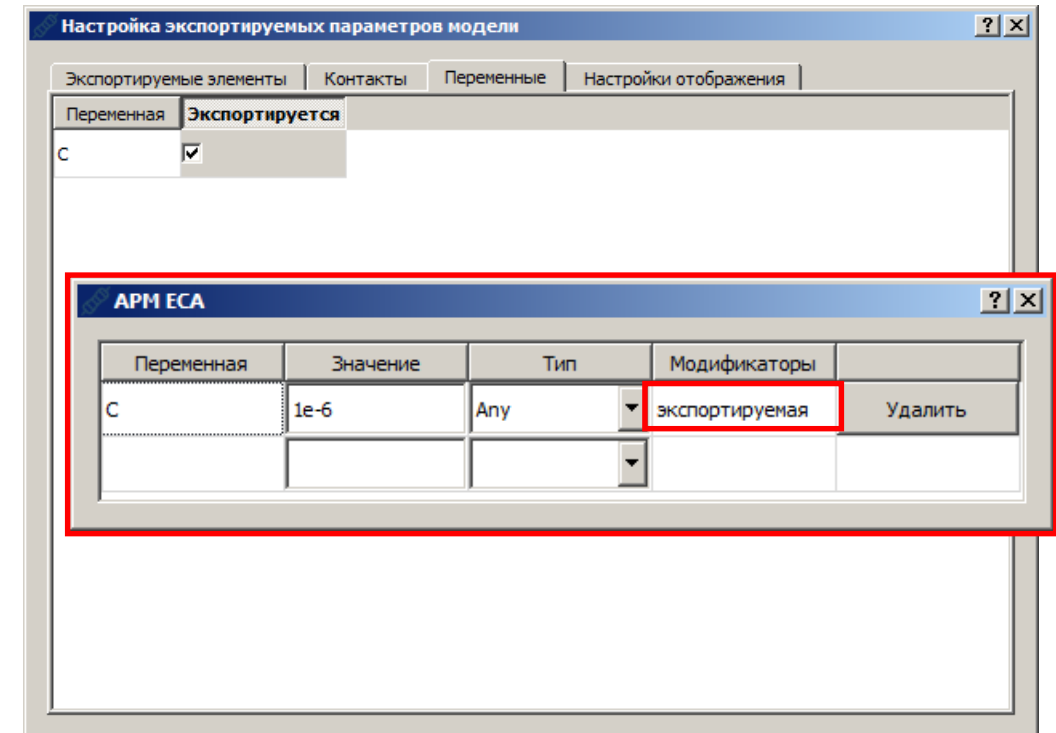
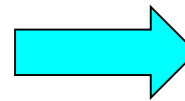
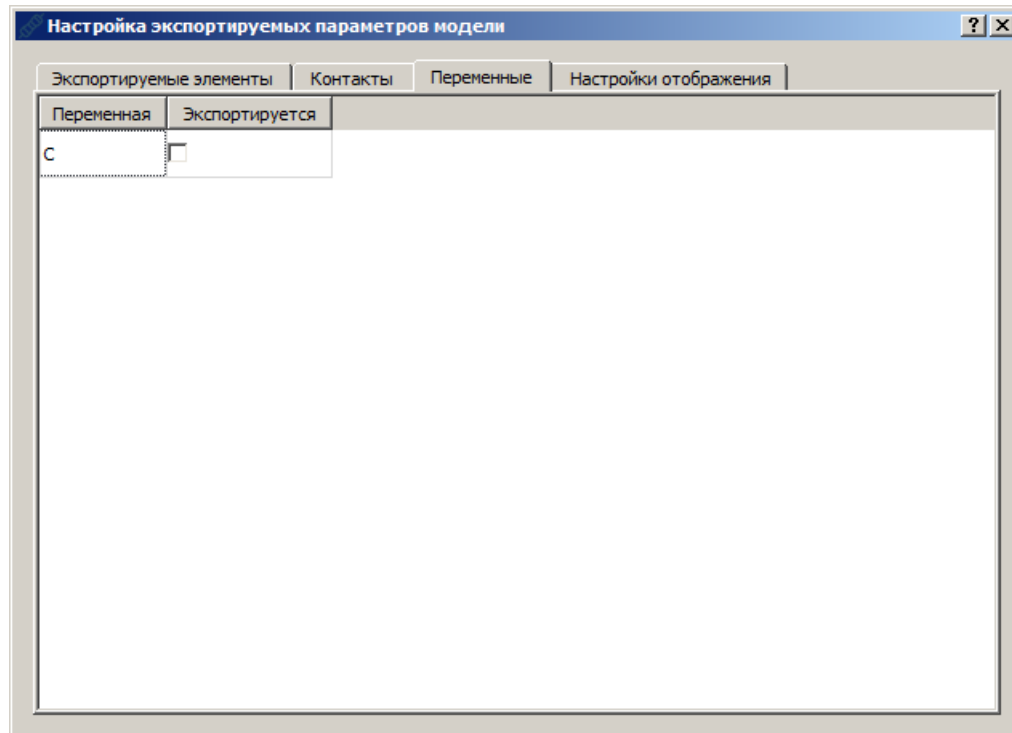
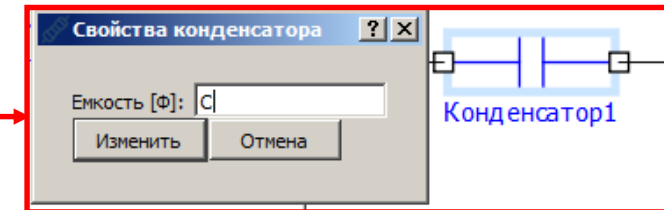
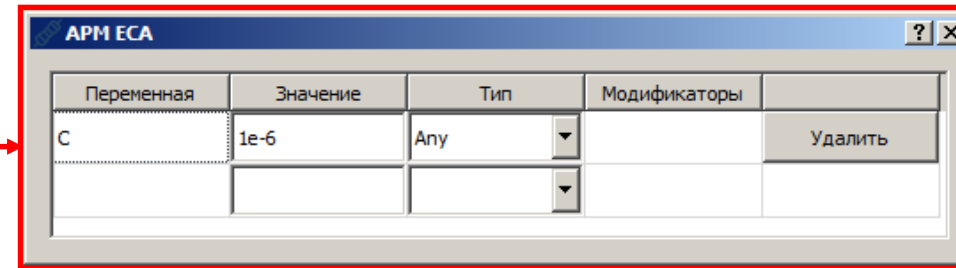
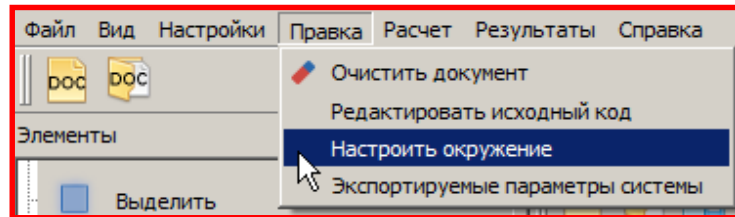
Настройка экспортируемых параметров модели

Экспортируемые элементы | Контакты | Переменные | Настройки отображения

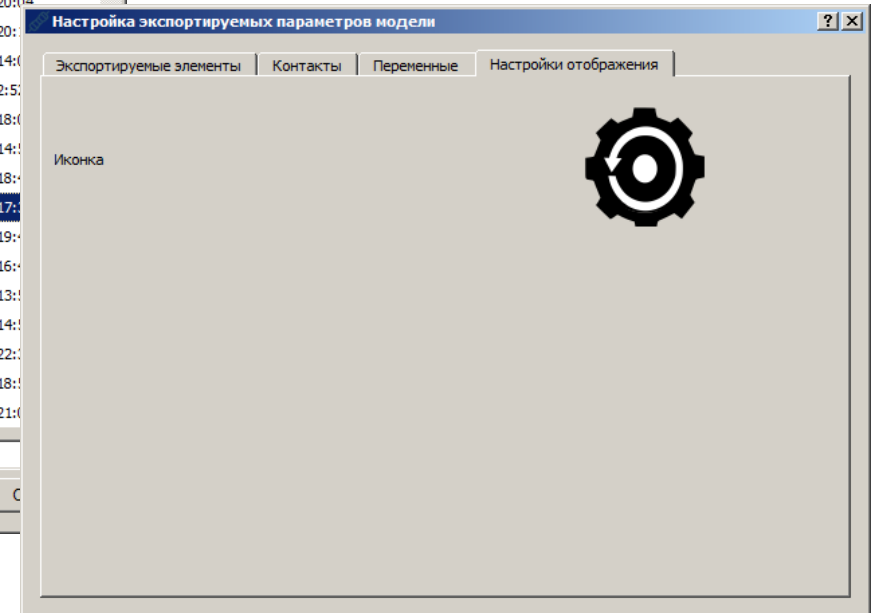
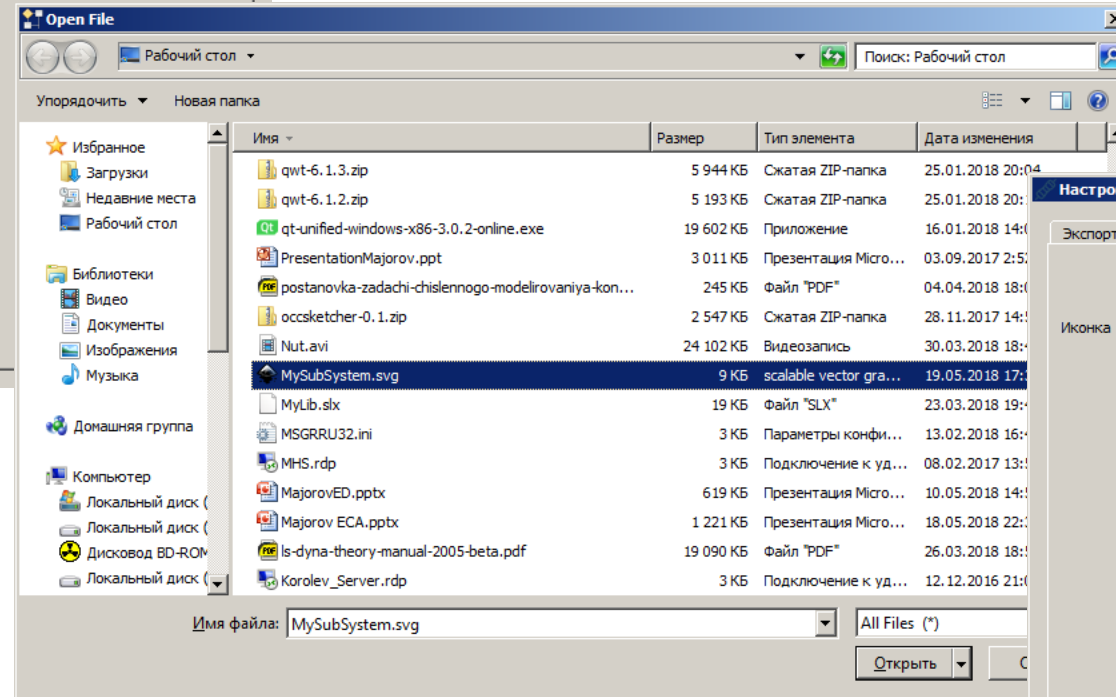
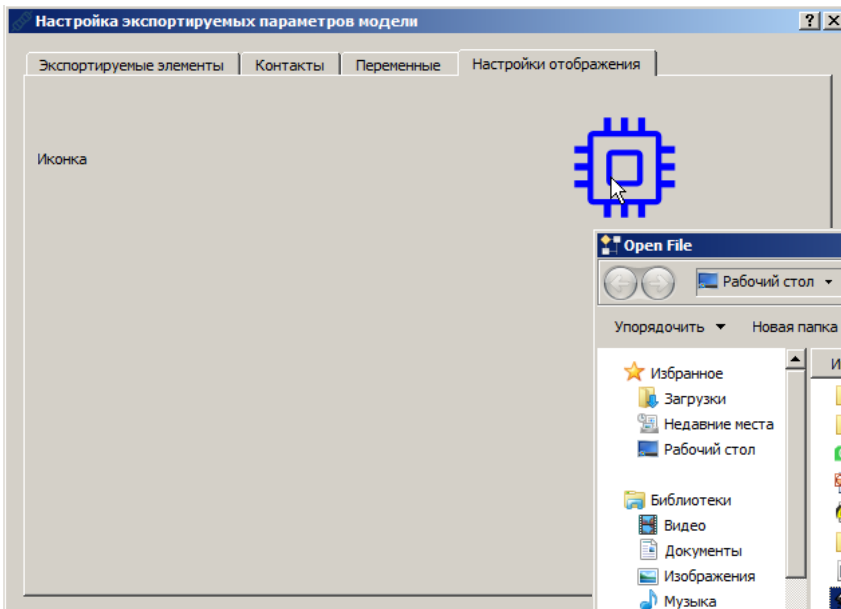
		Метка	Положение
1		Контакт1	слева
2		Контакт2	слева

- слева
- справа**
- сверху
- снизу

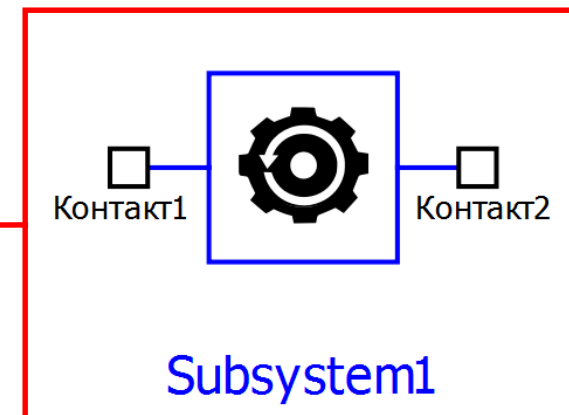
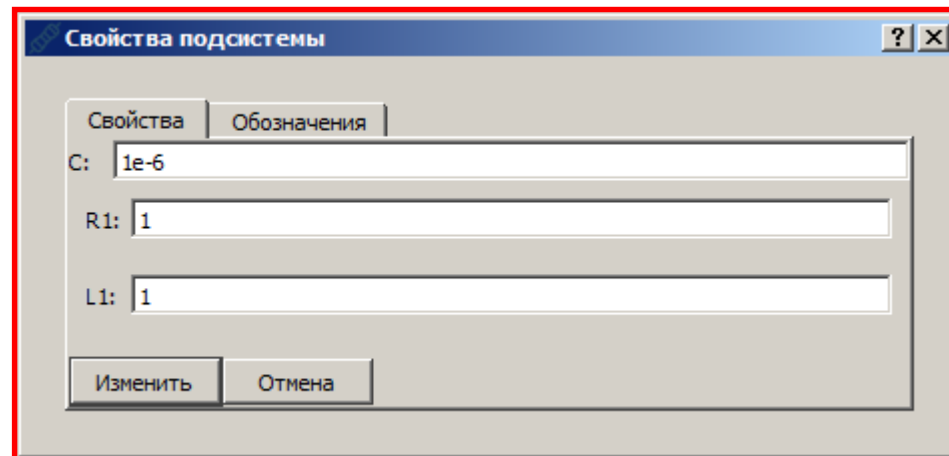
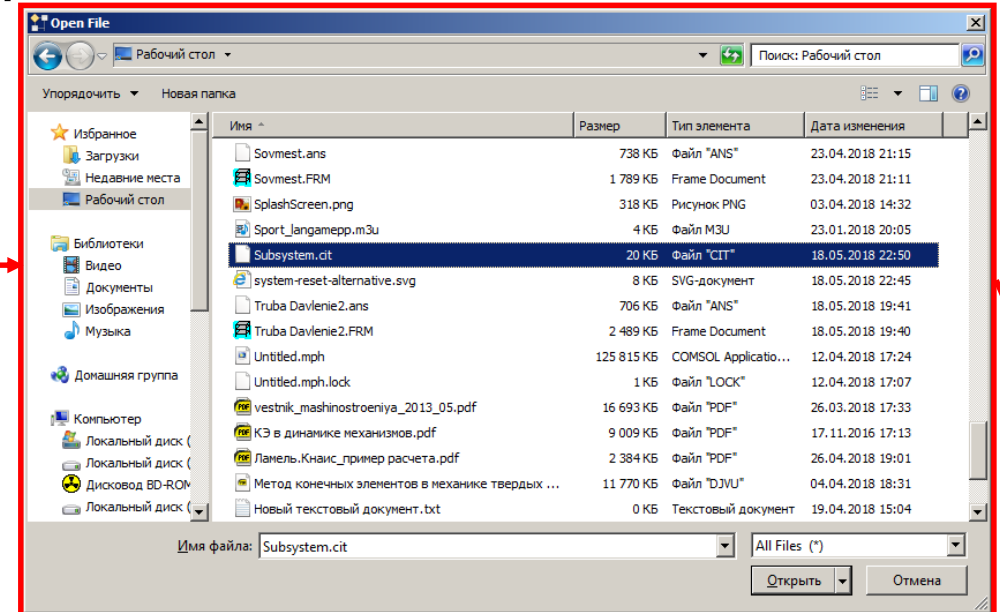
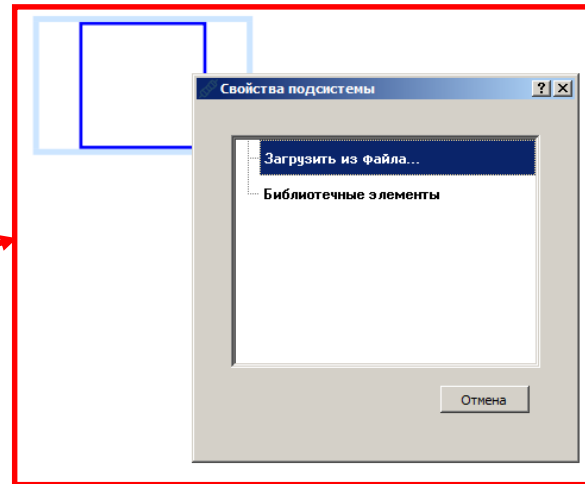
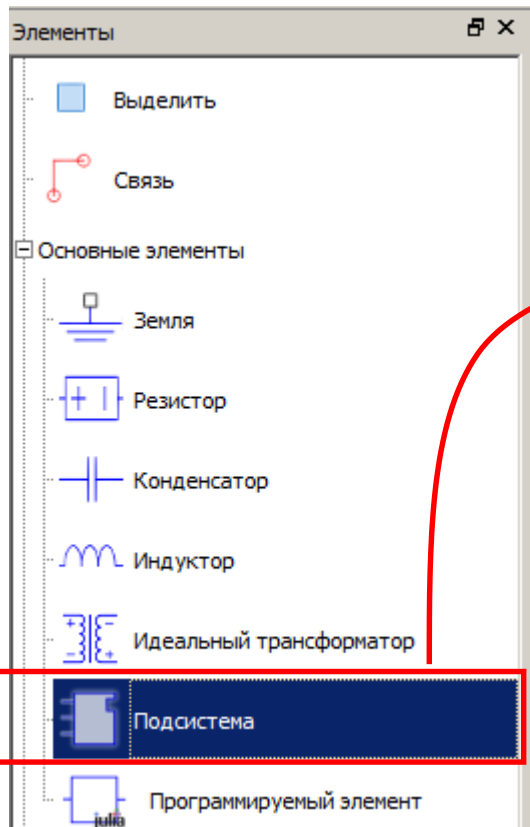
Настройка экспортируемых переменных



Настройка отображения подсистемы



Использование подсистемы



Настройка подсистемы

Свойства подсистемы

Свойства Обозначения

C:

R1:

L1:

Свойства подсистемы

Свойства Обозначения

Путь к документу подсистемы

Порт #0 Контакт1

Порт #1 Контакт2

Свойства подсистемы

Свойства Обозначения

Путь к документу подсистемы

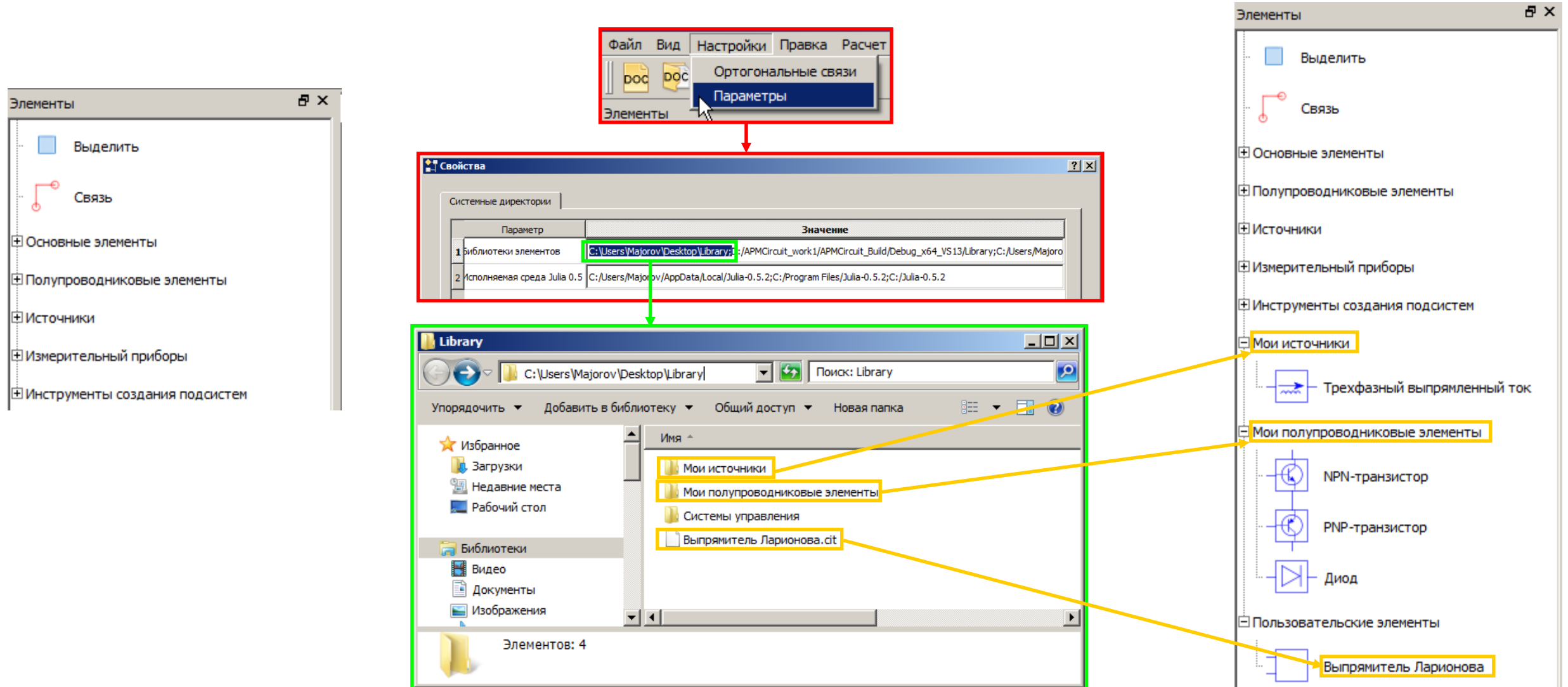
Порт #0 Контакт1

Порт #1 Контакт2

Необходимо хранить
файлы с подсистемой в
указанной директории!!!

Подсистема включена в текущий
файл и в дальнейшем файл
содержащий ее не потребуется

Организация библиотек элементов пользователя



The image illustrates the steps to organize user element libraries in a software application. It shows the 'Настройки' (Settings) menu, the 'Свойства' (Properties) dialog box, and the 'Library' folder view.

Настройки (Settings) Menu: The 'Настройки' (Settings) menu is highlighted, showing options like 'Ортогональные связи' (Orthogonal connections) and 'Параметры' (Parameters).

Свойства (Properties) Dialog Box: The 'Свойства' (Properties) dialog box is shown, displaying the 'Системные директории' (System directories) tab. The 'Параметр' (Parameter) list includes 'Библиотеки элементов' (Element libraries) and 'Исполняемая среда Julia 0.5' (Julia 0.5 executable environment). The 'Значение' (Value) for 'Библиотеки элементов' is set to 'C:\Users\Majorov\Desktop\Library;./APMCircuit_work1/APMCircuit_Build/Debug_x64_VS13/Library;C:\Users\Majorov\'. The 'Значение' for 'Исполняемая среда Julia 0.5' is 'C:\Users\Majorov\AppData\Local\Julia-0.5.2;C:\Program Files\Julia-0.5.2;C:\Julia-0.5.2'.

Library Folder View: The 'Library' folder view is shown, displaying the contents of the 'C:\Users\Majorov\Desktop\Library' folder. The folder contains several subfolders and files, including 'Мои источники' (My sources), 'Мои полупроводниковые элементы' (My semiconductor elements), 'Системы управления' (Control systems), and 'Выпрямитель Ларионова.ct' (Larionov rectifier.ct).

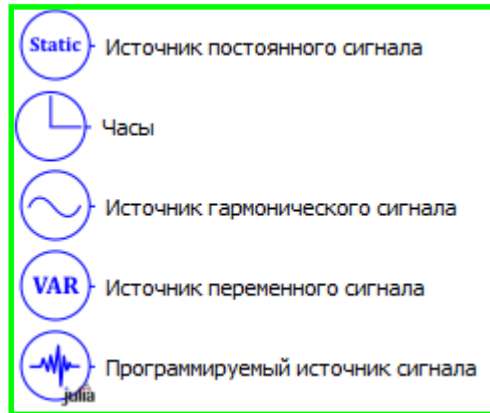
Элементы (Elements) Panel: The 'Элементы' (Elements) panel is shown on the right, displaying a list of elements categorized by type. The categories include 'Основные элементы' (Basic elements), 'Полупроводниковые элементы' (Semiconductor elements), 'Источники' (Sources), 'Измерительные приборы' (Measuring instruments), 'Инструменты создания подсистем' (Subsystem creation tools), 'Мои источники' (My sources), 'Мои полупроводниковые элементы' (My semiconductor elements), and 'Пользовательские элементы' (User elements). The 'Мои источники' and 'Мои полупроводниковые элементы' categories are highlighted with yellow boxes. The 'Пользовательские элементы' category is also highlighted, showing the 'Выпрямитель Ларионова' (Larionov rectifier) element.

- Расширение элементной базы
- Параметризация
- Программируемые пользовательские элементы
- Работа с подсистемами
- **Имитационные схемы**
- Взаимосвязь между электрическими и имитационными схемами
- Автоматические системы управления

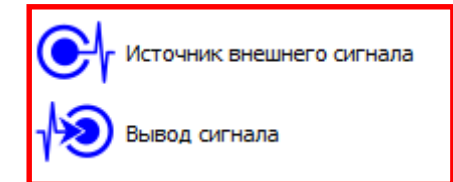
Элементная база

Основные элементы

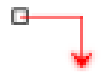
Источники сигнала

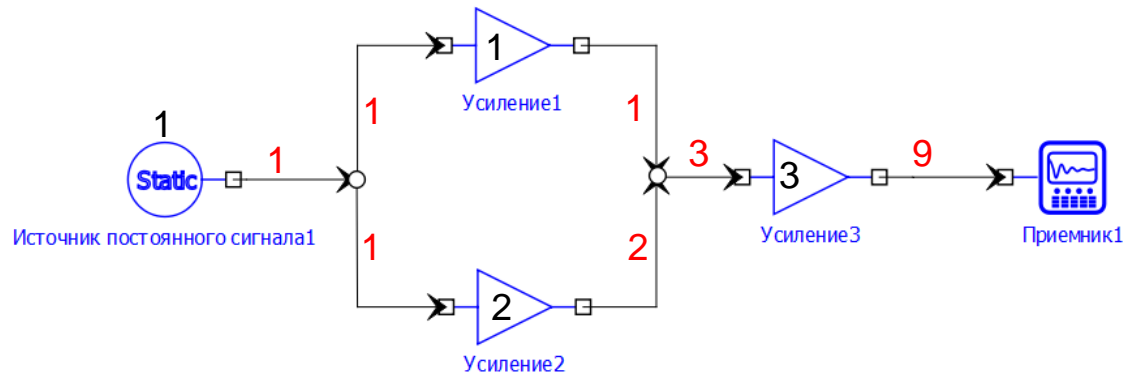



Инструменты формирования подсистем

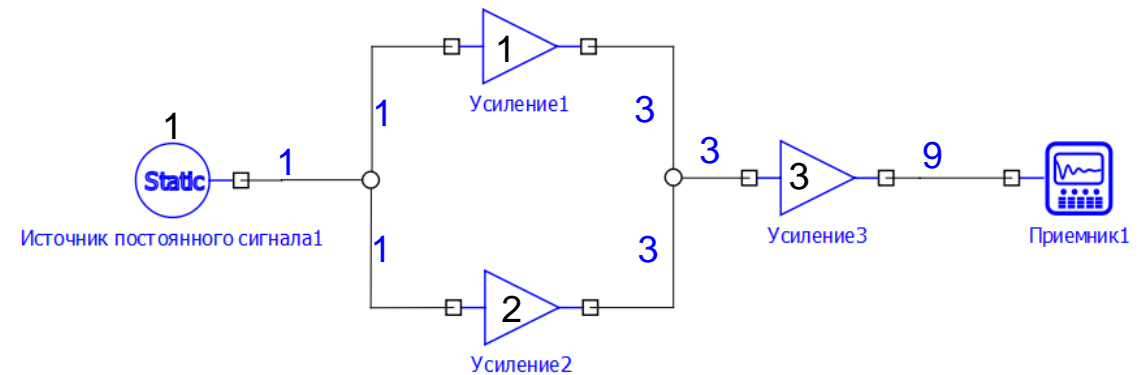


Направленная и объединяющая связь

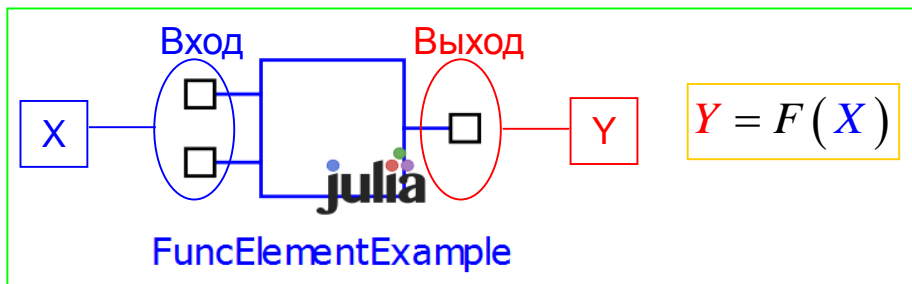
 Направленная связь



 Связь



Программируемый функциональный элемент



```
type FuncElementExample <: SignalFuncElement
    a1::Float64
    a2::Float64
    FuncElementExample() = new(1.0, 1.0)
end
```

Имя базового элемента

```
function numPorts(e::FuncElementExample)
    return (2, 1)
end
```

Число входных портов

Число выходных портов

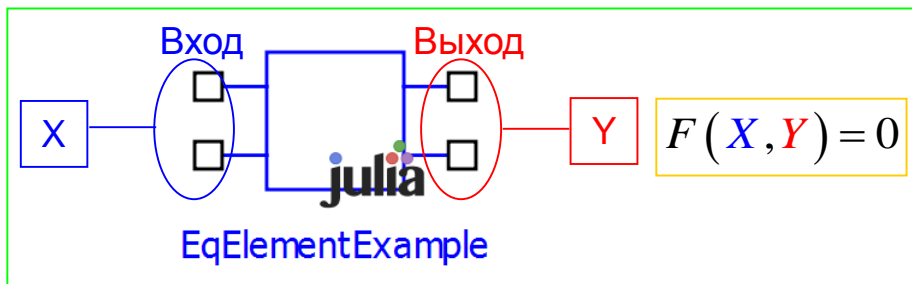
Массив значений сигналов на входе
In – имя массива
Array – указывает что входной параметр массив
Float64 – тип элементов массива
Размер массива N – число входных портов элемента

```
function (e::FuncElementExample)(t::Float64, In::Array{Float64})
    Out = (In[1]^e.a1 + In[2]^e.a2)*sin(2*pi*t)
    return [Out]
end
```

Функция ДОЛЖНА возвращать массив значений сигналов на выходе.
Размер массива M – число выходных портов элемента

Функциональная зависимость

Программируемый элемент, определяемый уравнениями



```
type EqElementExample <: SignalEquationElement
  a1::Float64
  a2::Float64
  EqElementExample() = new(2.0, 1.5)
end
```

```
function numPorts(e::EqElementExample)
  return (2, 2)
end
```

Число входных портов

Число выходных портов

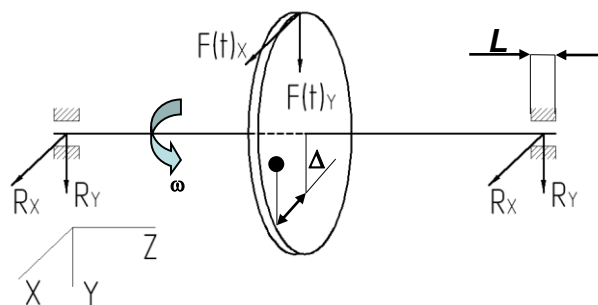
Массив значений сигналов на выходе
Out – имя массива
Array – указывает что входной параметр массив
Float64 – тип элементов массива
Размер массива M – число входных портов элемента

```
function (e::EqElementExample)(t::Float64, In::Array{Float64},
  Out::Array{Float64})
  f1 = log( e.a1 + In[2]*Out[1] ) + Out[1]
  f2 = log( e.a2 + In[1]*Out[2] ) + Out[2]
  return [f1; f2]
end
```

Функция ДОЛЖНА возвращать массив значений
рассчитанных по уравнениям.
Размер массива M – число выходных портов элемента

Функциональная
зависимость

Динамика ротора на подшипниках скольжения



m – масса ротора

R_x, R_y – проекции реакции подшипника

F_x, F_y – проекции внешних сил

ω – угловая скорость ротора

Δ – статический дисбаланс

t – время

h_0 – радиальный зазор подшипника

R – радиус подшипника

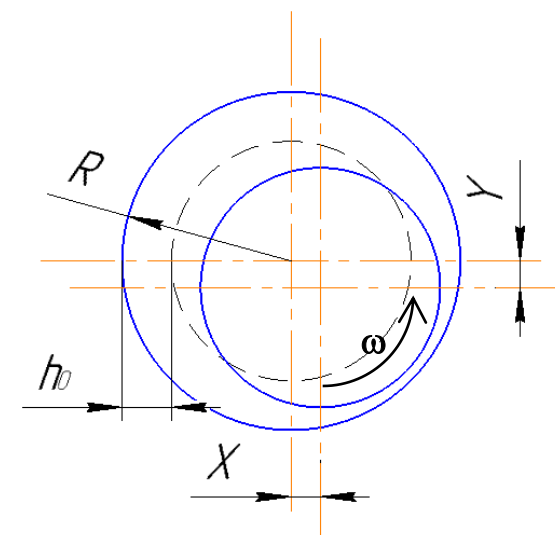
L – длина подшипника

X, Y – координаты центра ротора

μ – динамическая вязкость смазки

Уравнения движения

$$\begin{cases} m\ddot{X} = 2R_x + m\Delta\omega^2 \sin \omega t + F_x \\ m\ddot{Y} = 2R_y + m\Delta\omega^2 \cos \omega t + F_y \end{cases}$$

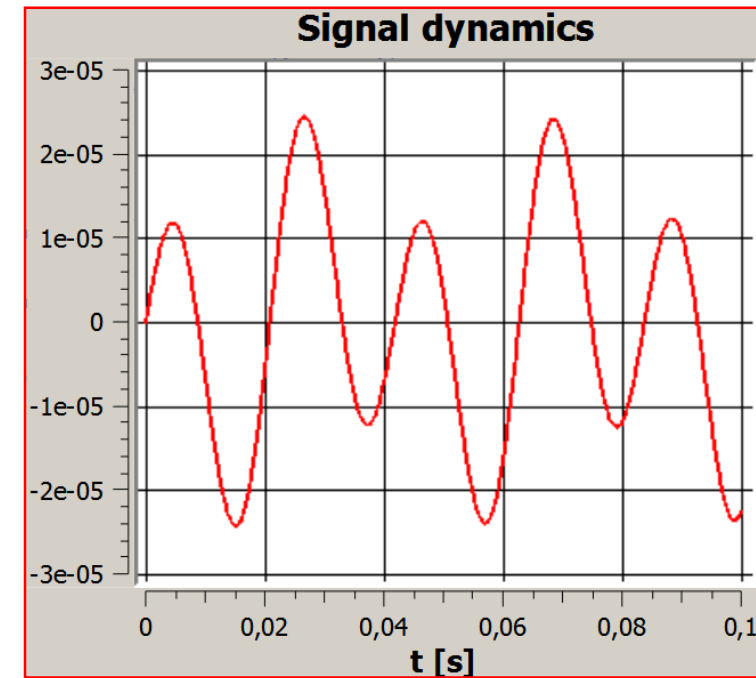
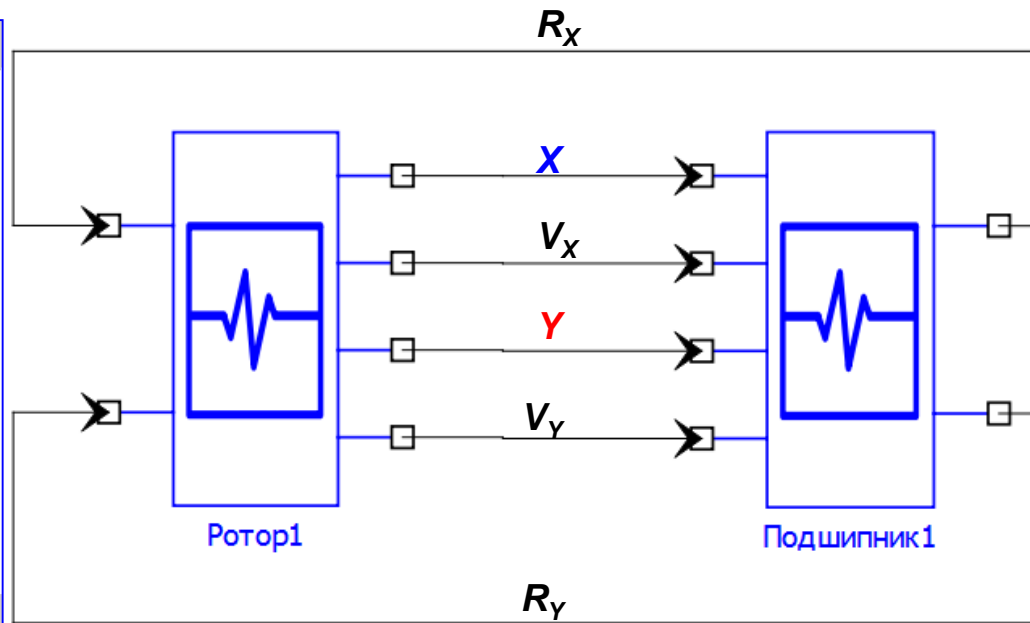
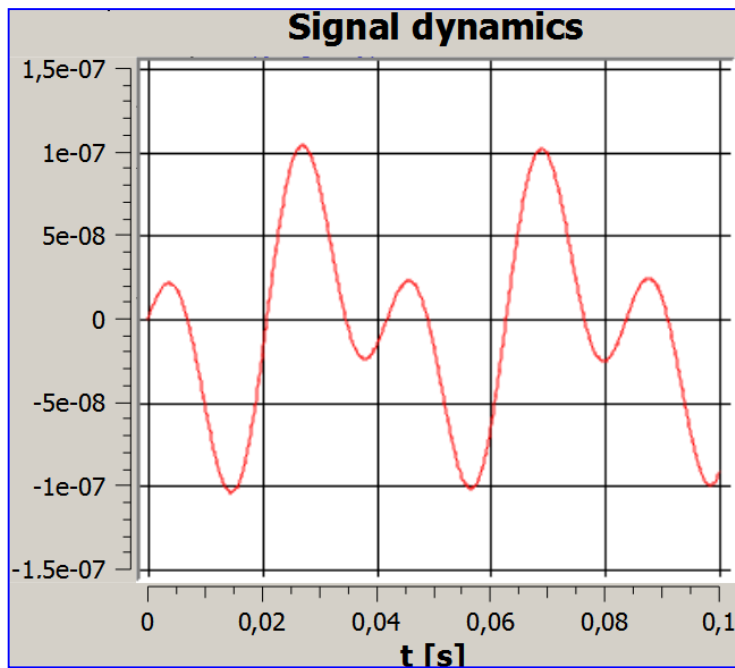


Реакции подшипника

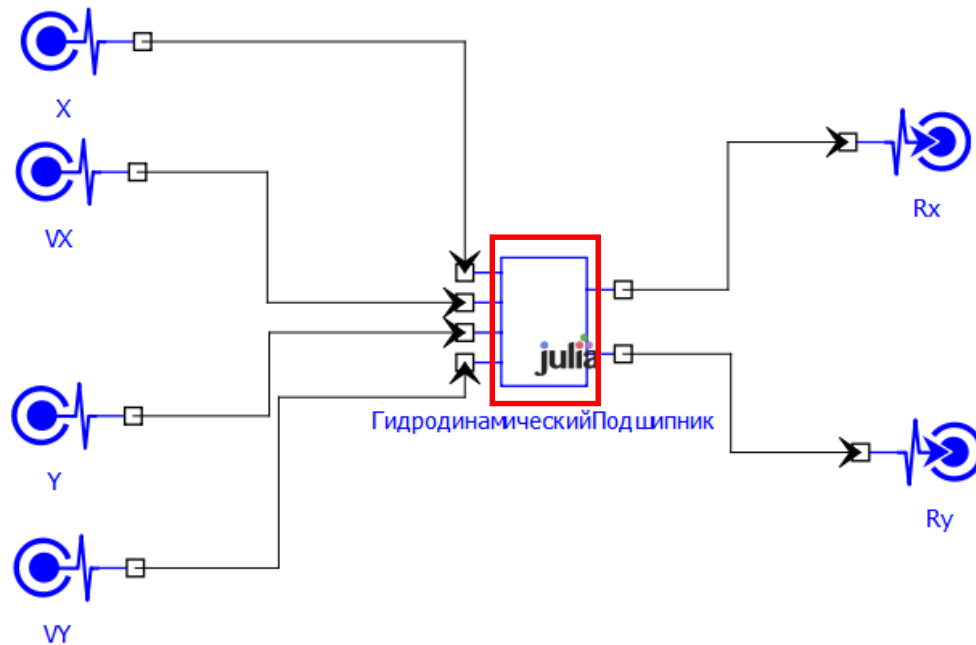
$$R_x = -\mu\pi RL^3 \left(\frac{\omega Y + 2\dot{X}}{2(h_0^2 - X^2 - Y^2)^{3/2}} + \frac{3X(X\dot{X} + Y\dot{Y})}{(h_0^2 - X^2 - Y^2)^{5/2}} \right)$$

$$R_y = -\mu\pi RL^3 \left(\frac{2\dot{Y} - \omega X}{2(h_0^2 - X^2 - Y^2)^{3/2}} + \frac{3Y(X\dot{X} + Y\dot{Y})}{(h_0^2 - X^2 - Y^2)^{5/2}} \right)$$

Динамика ротора на подшипниках скольжения. Система



Подсистема подшипника



```

type ГидродинамическийПодшипник <: SignalFuncElement
    Длина::Float64
    Радиус::Float64
    Номинальный_зазор::Float64
    Динамическая_вязкость::Float64
    Окружная_скорость::Float64
    ГидродинамическийПодшипник() = new(0.02, 0.02, 1e-5, 1e-3, 300.0)
end

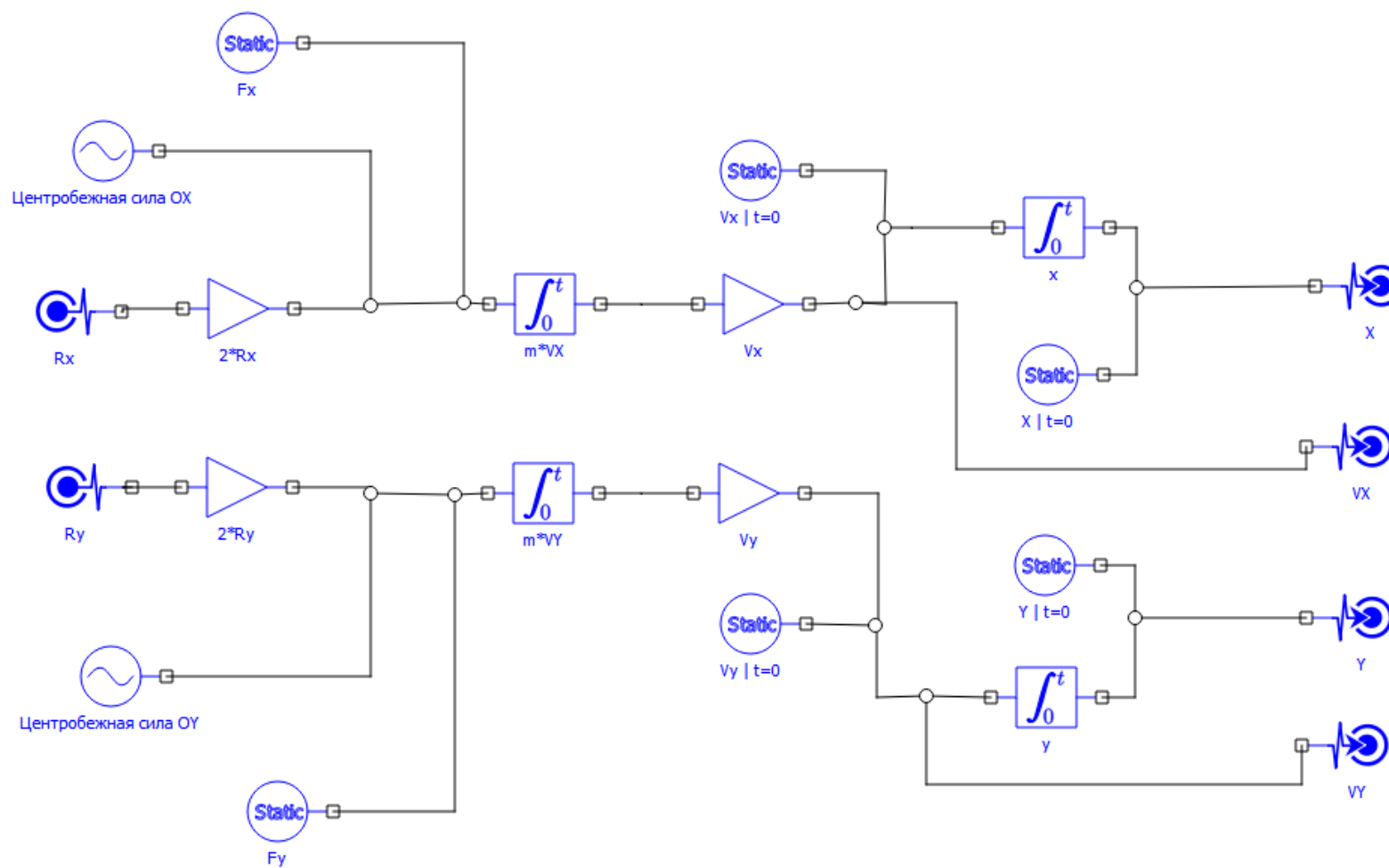
function numPorts(e::ГидродинамическийПодшипник)
    return (4, 2)
end

function (e::ГидродинамическийПодшипник)(t::Float64, Inputs::Array{Float64})
    return ShortBearingReactions(e.Длина, e.Радиус, e.Номинальный_зазор,
        e.Окружная_скорость, e.Динамическая_вязкость,
        Inputs[1], Inputs[2], Inputs[3], Inputs[4])
end

function ShortBearingReactions(l::Float64, R::Float64, h0::Float64,
    omega::Float64, mu::Float64, X::Float64,
    Vx::Float64, Y::Float64, Vy::Float64)

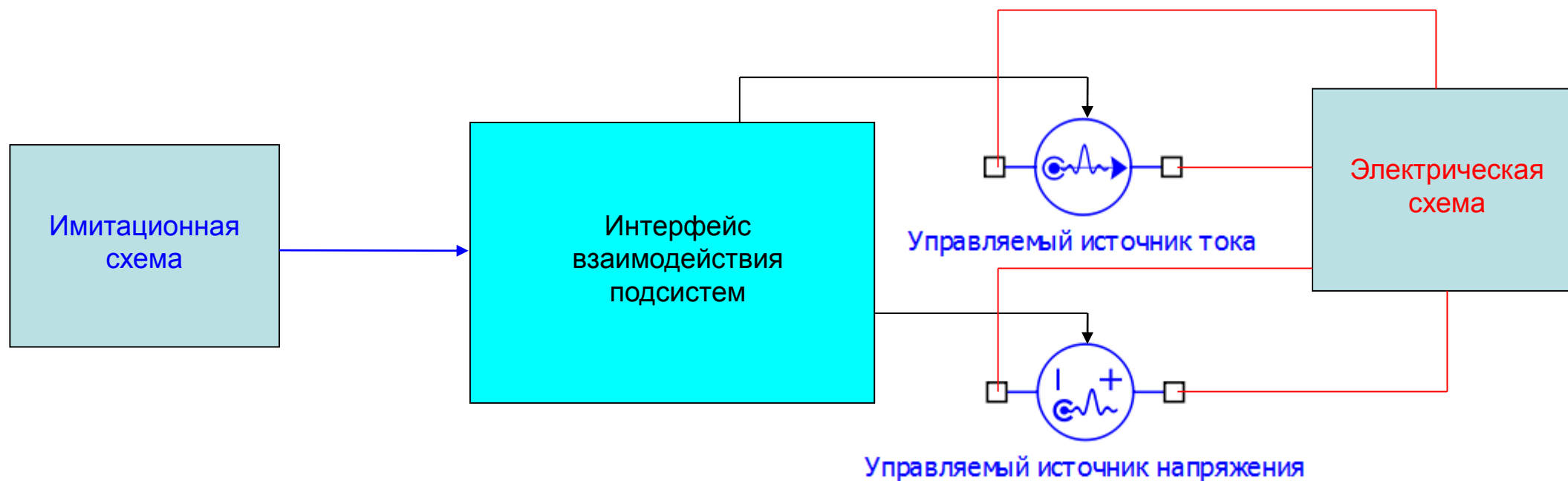
    a = -mu*pi*R*l^3
    b = sqrt(h0^2 - X^2 - Y^2)
    c = 3*(X*Vx + Y*Vy)
    Rx = a*( (omega*Y + 2*Vx)/(2*b^3) + X*c/b^5 )
    Ry = a*( (-omega*X + 2*Vy)/(2*b^3) + Y*c/b^5 )
    return [Rx; Ry]
end
    
```

Подсистема ротора

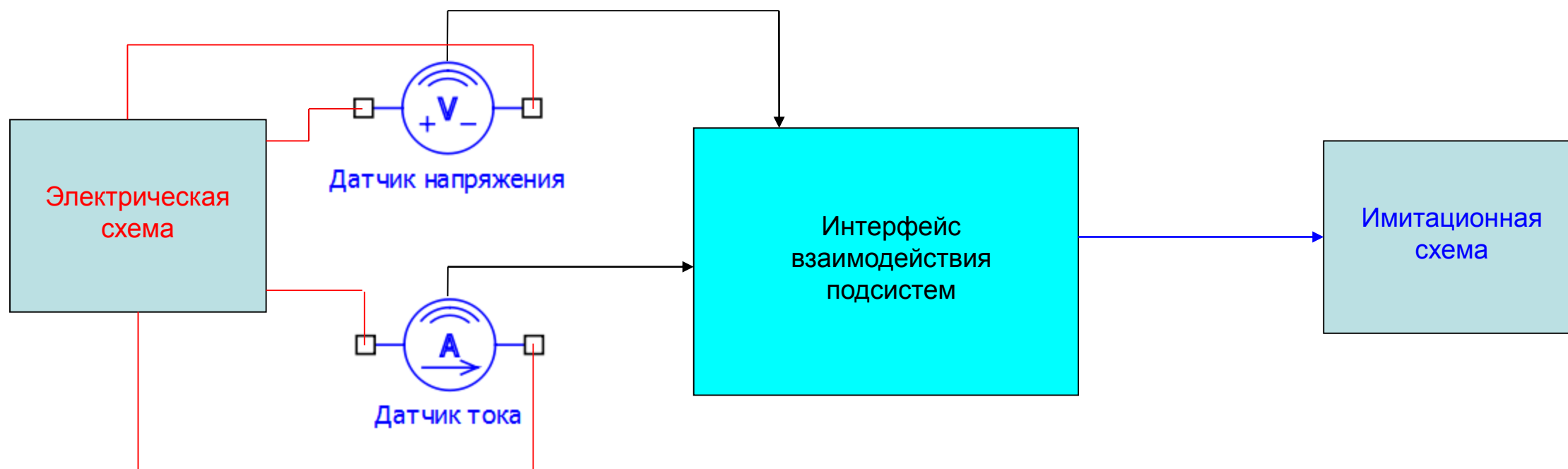


- Расширение элементной базы
- Параметризация
- Программируемые пользовательские элементы
- Работа с подсистемами
- Имитационные схемы
- **Взаимосвязь между электрическими и имитационными схемами**
- Автоматические системы управления

Управляемые электрические источники

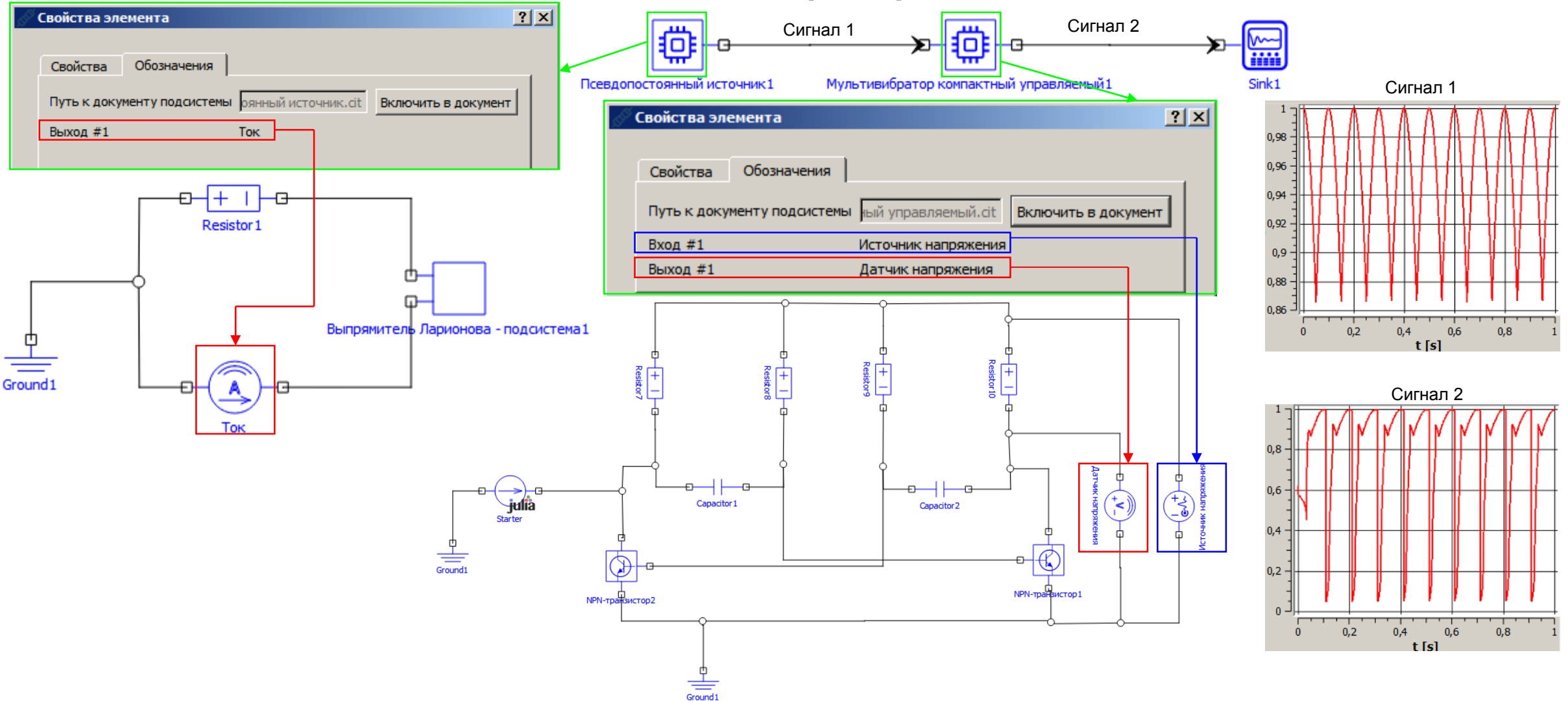


Датчики тока и напряжения

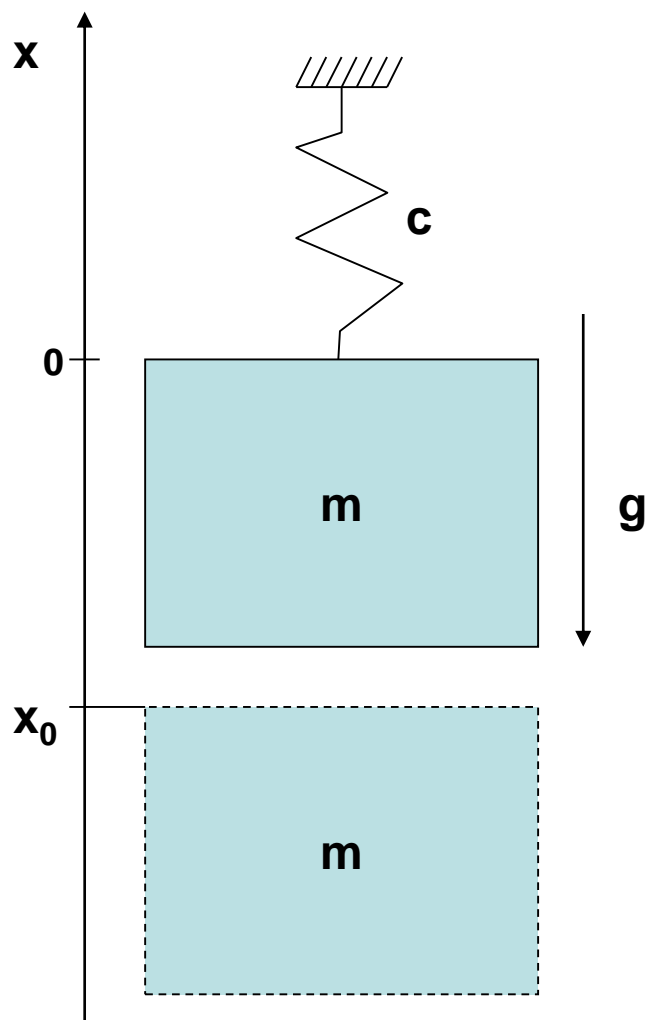


Взаимосвязь между электрическими и имитационными схемами

Пример



- Расширение элементной базы
- Параметризация
- Программируемые пользовательские элементы
- Работа с подсистемами
- Имитационные схемы
- Взаимосвязь между электрическими и имитационными схемами
- **Автоматические системы управления**



Элементарный осциллятор

Уравнение движения

$$m\ddot{x} = -cx - mg$$

Начальные условия

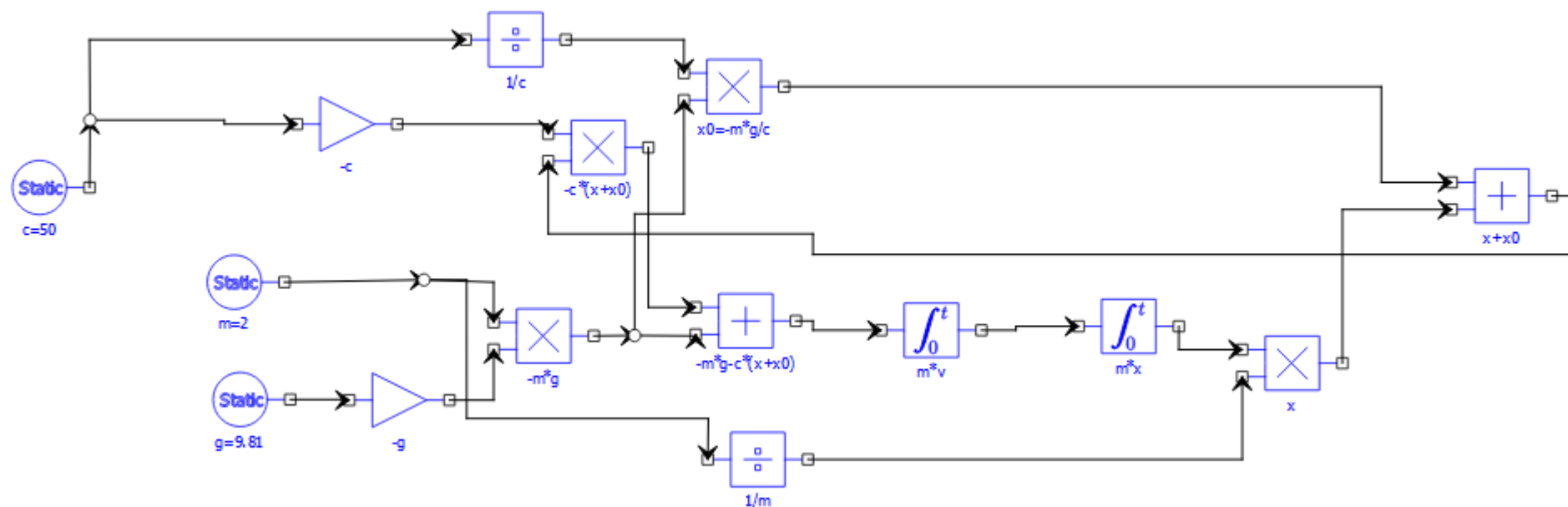
1. Состояние покоя:

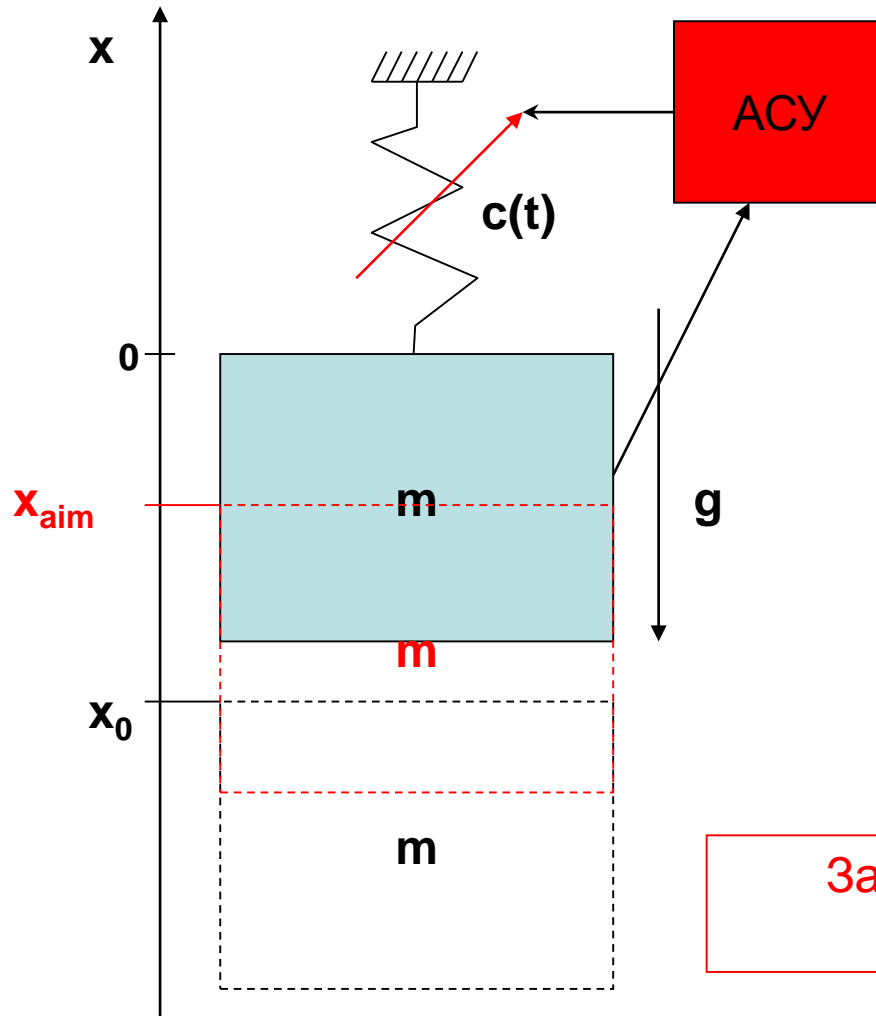
$$\dot{x}(0) = 0$$

2. Положение равновесия:

$$x(0) = x_0 = -\frac{mg}{c}$$

Элементарный осциллятор. Схема





Элементарный осциллятор с управляемой жесткостью

Уравнение движения

$$m\ddot{x} = -c(t)x - mg$$

Начальные условия

1. Состояние покоя:

$$\dot{x}(0) = 0$$

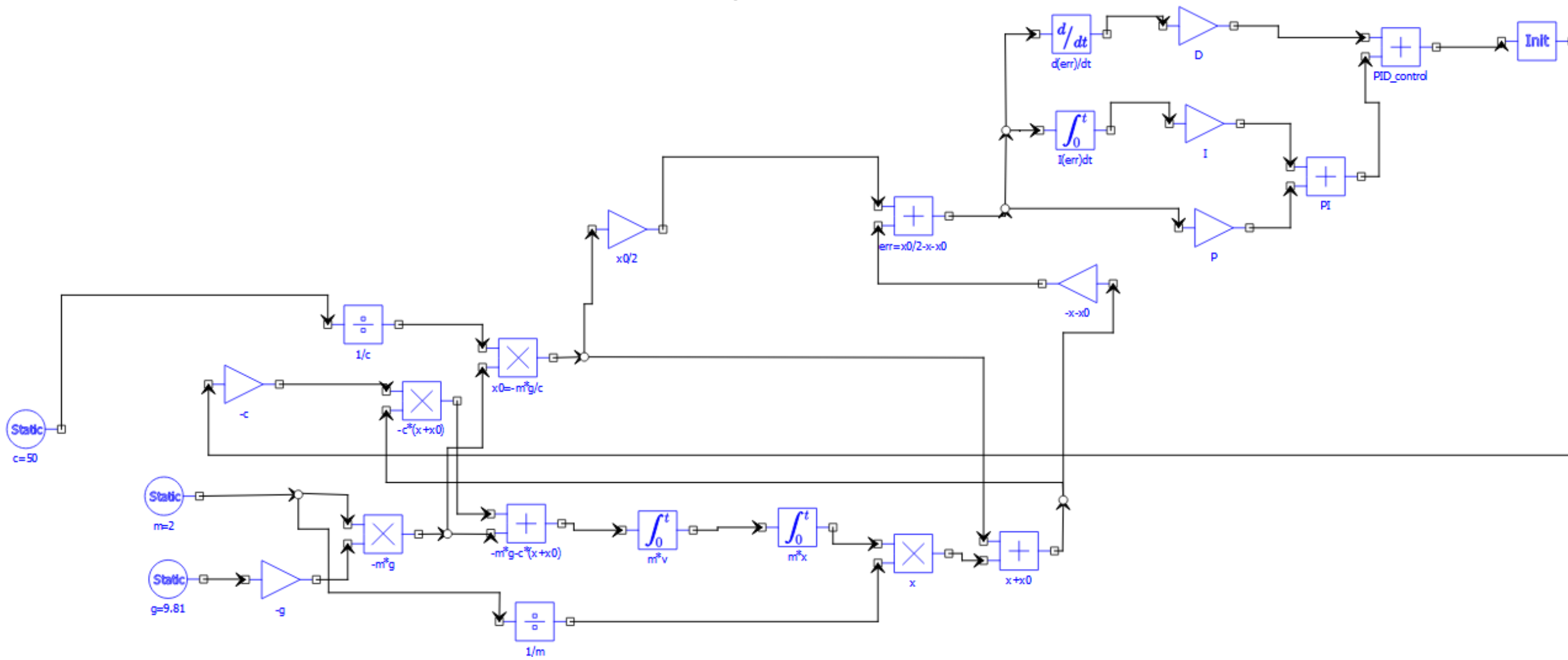
2. Положение равновесия:

$$x(0) = x_0 = -\frac{mg}{c}$$

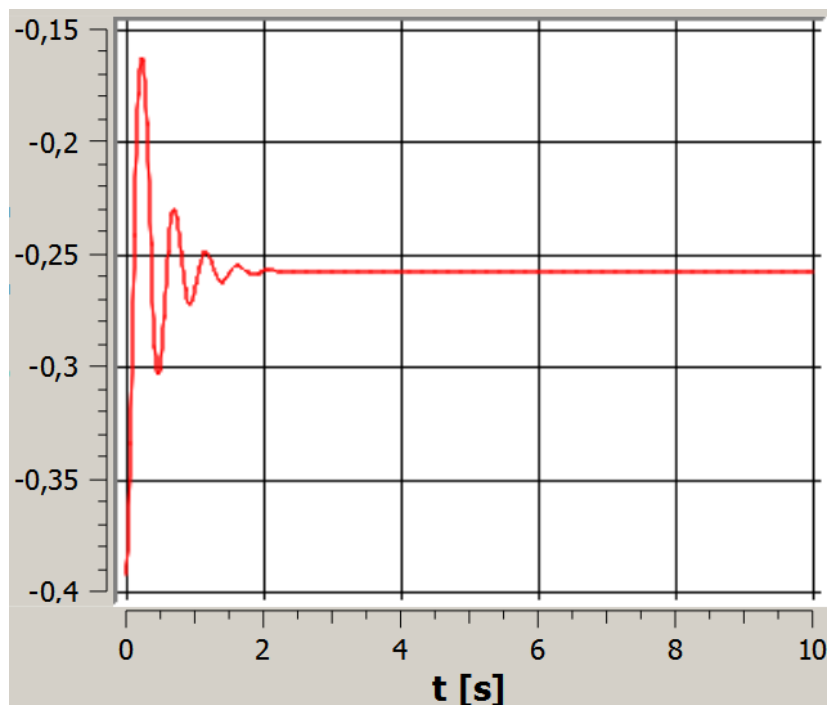
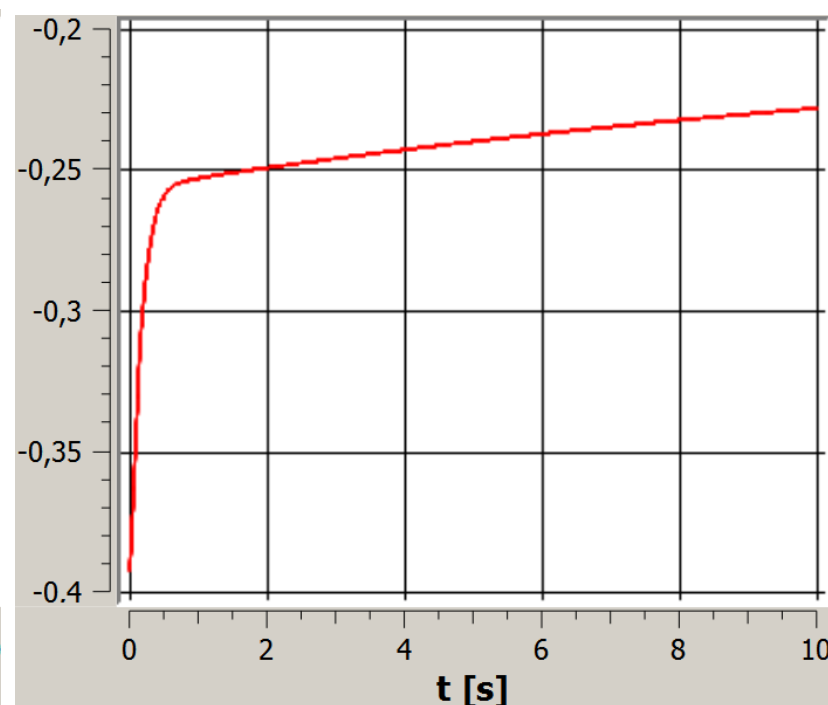
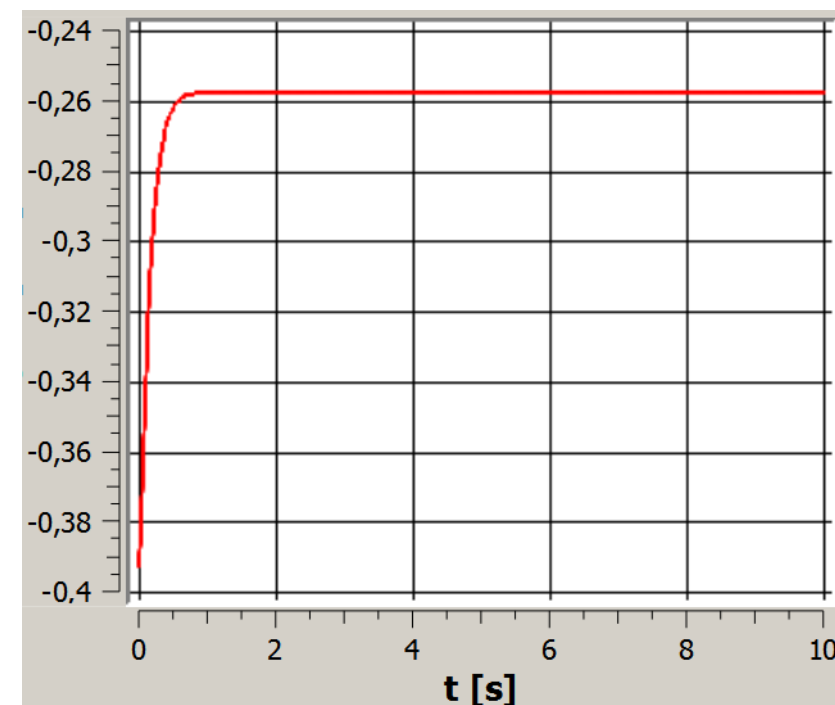
Задача АСУ: перевести систему из начального положения равновесия x_0 в некоторое заданное (целевое) x_{aim} .

Примем: $m=2$ кг; $c=50$ Н/м; $g=9,81$ м/с²; $x_{aim}=x_0/2$

Элементарный осциллятор с управляемой жесткостью. Схема



Результаты настройки регулятора

 $P=1240 \ I=0.1 \ D=25$  $P=1240 \ I=100 \ D=250$  $P=1240 \ I=0.1 \ D=250$ 

Спасибо за внимание!

**Компания НТЦ «АПМ»
(научно-технический центр)
Московская область, г. Королев
Октябрьский бульвар, д. 14, офис 6
Тел.: (498) 600-25-10, (495) 514-84-19
Internet: www.apm.ru, www.cae.apm.ru
E-mail: com@apm.ru**