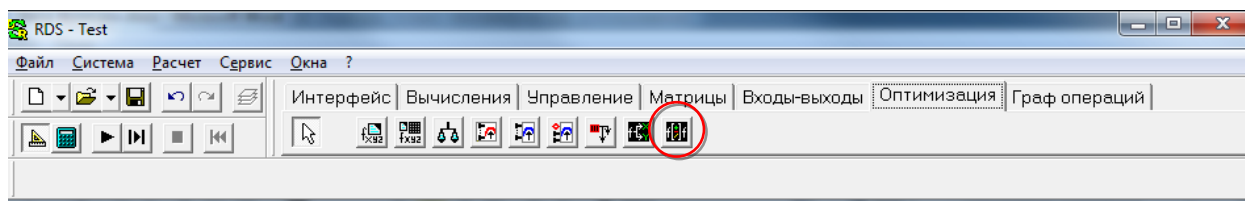


Операции над функциями

Блок “Операции над функциями” предназначен для конструирования новой функции (программы в синтаксисе языка C) из одной или нескольких подобных функций, поступивших на вход блока. С выхода блока сконструированная функция подается на соответствующие входы *Func_in* других блоков.

Размещение на панели блоков:

Вкладка “Оптимизация”:



Внешний вид в схеме:



Входы:

<имя_функции> (*char*) – входы для входных функций (по одному на каждую функцию, предварительно добавленную в настройках)

<имя>, *<имя>_max*, *<имя>_min*, *<имя>_step*, *<имя>_end*, *<имя>_0* (*double*) – входы для параметров функции, создаются для всех параметров типа "число" пришедших по связям функций, а также для всех дополнительных параметров типа "число", созданных в настройках блока.

<имя_матрицы> (*матрица double*) – входы для параметров функции, создаются для всех параметров прочих типов пришедших по связям функций, а также для всех дополнительных параметров прочих типов, созданных в настройках блока.

Index_in (*int*) - вход для индекса переменных. Используется только если в настройках блока включена операция добавления индекса к имени переменной. Считывается перед расчетом, изменение значения входа в процессе расчета считается ошибкой.

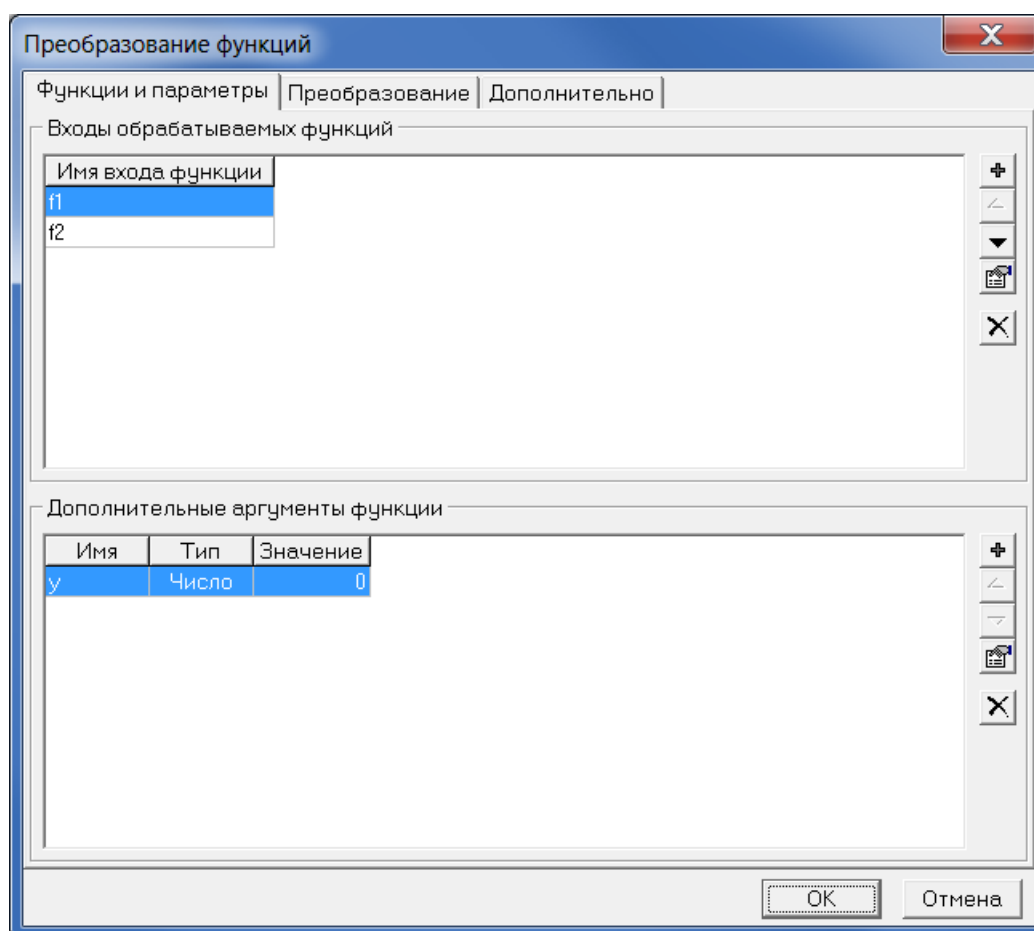
Выходы:

Func_out (*char*) – выход для передачи функции по связи, с которого сформированная функция вместе с ее параметрами подается на входы *Func_in* других блоков.

Настроечные параметры:

Вкладка “Функции и параметры”

Вкладка “Функции и параметры” предназначена для добавления входных функций и дополнительных аргументов для выходной функции.



Имена функций и аргументов могут содержать только заглавные и строчные буквы латинского алфавита (чувствительны к регистру) и цифры. Они могут начинаться только с буквы (за исключением последовательностей символов “rds” или “157”).

Вкладка состоит из двух панелей:

- Панель “Входы обрабатываемых функций”. На панели расположена таблица с именами входных функций блока и панель кнопок для добавления, редактирования и удаления строк данной таблицы.
- Панель “Дополнительные аргументы функции”. На панели расположена таблица параметров выходной функции и панель кнопок для добавления, редактирования и удаления строк данной таблицы. Таблица состоит из следующих столбцов:
 - “Имя”. Столбец, в котором отображаются имена аргументов функции и соответствующих им входов блока.
 - “Тип”. Столбец, в котором отображаются типы аргументов функции.
 - “Значение”. Столбец, в котором отображаются значения аргументов функции, которые будут использоваться при вычислениях, если к соответствующим им входам блока не будут подключены связи.
- Панель кнопок. Включает в себя кнопки для добавления, смены положения, редактирования и удаления строк таблицы “Дополнительные аргументы функции”.

Добавление и редактирование строк таблицы (входов блока) осуществляется с помощью формы, изображенной ниже.

Такие аргументы могут быть различных типов. Таблица соответствия типов аргументов и типов входов блока приведена ниже.

Тип аргумента	Тип входа блока
Число	double
Матрица	матрица double
Массив	массив double
Табличная функция одной переменной	матрица double специального формата (формат приведен ниже)
Табличная функция двух переменных	матрица double специального формата (формат приведен ниже)

Если x – аргумент табличной функции одной переменной $F(x)$, то формат ее матрицы имеет следующий вид

	0	1	2	3
0	$x_1 = x_{\min}$	$F_1(x_1)$	$F_2(x_1)$...
1	x_2	$F_1(x_2)$	$F_2(x_2)$...
...	...			
i	x_{i+1}	$F_1(x_{i+1})$	$F_2(x_{i+1})$...
...	...			
	x_{\max}	$F_1(x_{\max})$	$F_2(x_{\max})$...

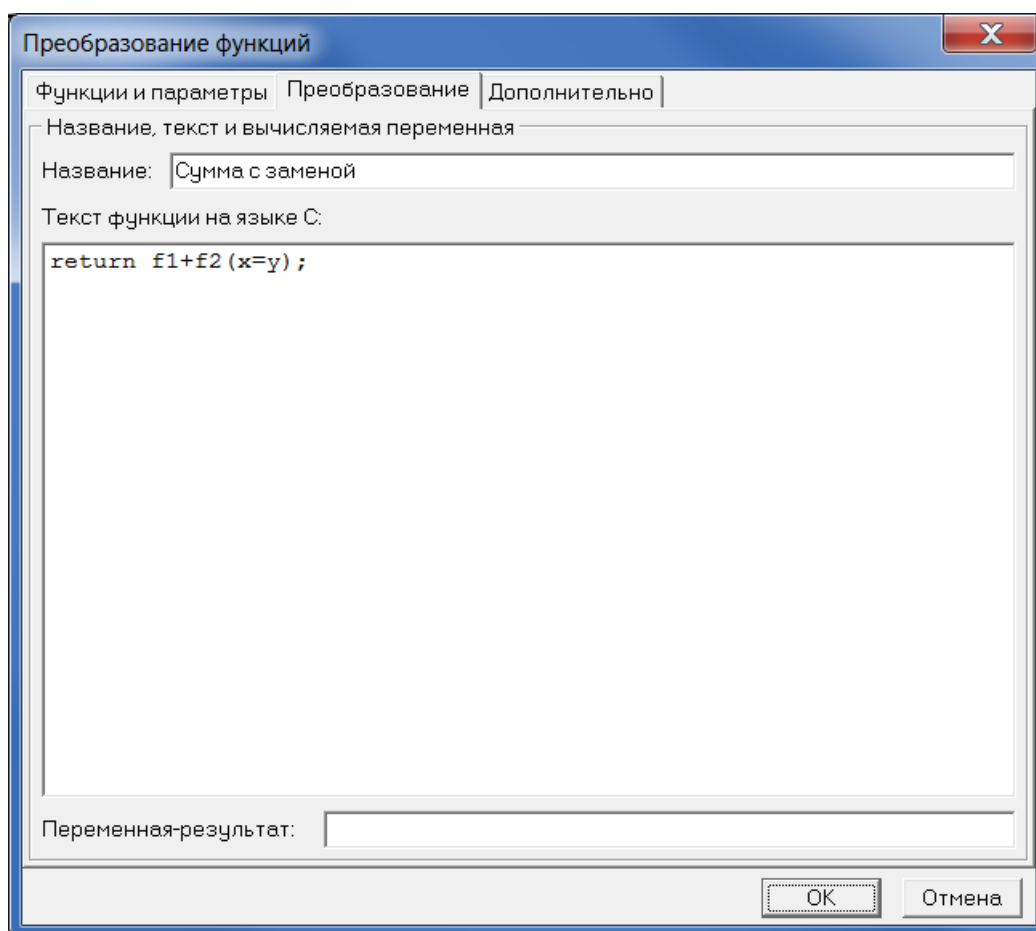
В приведенной выше таблице может располагаться несколько функций с общим аргументом. По умолчанию берется значение функции из столбца с индексом 1.

Если x и y – первый и второй аргументы табличной функции двух переменных $F(x, y)$, то формат ее матрицы имеет следующий вид

	0	1	2	...	j	...	
0	?	$y_1 = y_{\min}$	y_2		y_j		y_{\max}
1	$x_1 = x_{\min}$	$F(x_1, y_1)$	$F(x_1, y_2)$		$F(x_1, y_j)$		$F(x_1, y_{\max})$
2	x_2	$F(x_2, y_1)$	$F(x_2, y_2)$		$F(x_2, y_j)$		$F(x_2, y_{\max})$
...	x_i						
i		$F(x_i, y_1)$	$F(x_i, y_2)$		$F(x_i, y_j)$		$F(x_i, y_{\max})$
...							
	x_{\max}	$F(x_{\max}, y_1)$	$F(x_{\max}, y_2)$		$F(x_{\max}, y_j)$		$F(x_{\max}, y_{\max})$

Ячейка матрицы [0][0] не используется и содержит служебное значение, отображающееся в РДС как вопросительный знак.

Вкладка “Преобразование”



Преобразование функций

Функции и параметры | Преобразование | Дополнительно

Название, текст и вычисляемая переменная

Название: Сумма с заменой

Текст функции на языке C:

```
return f1+f2(x=y);
```

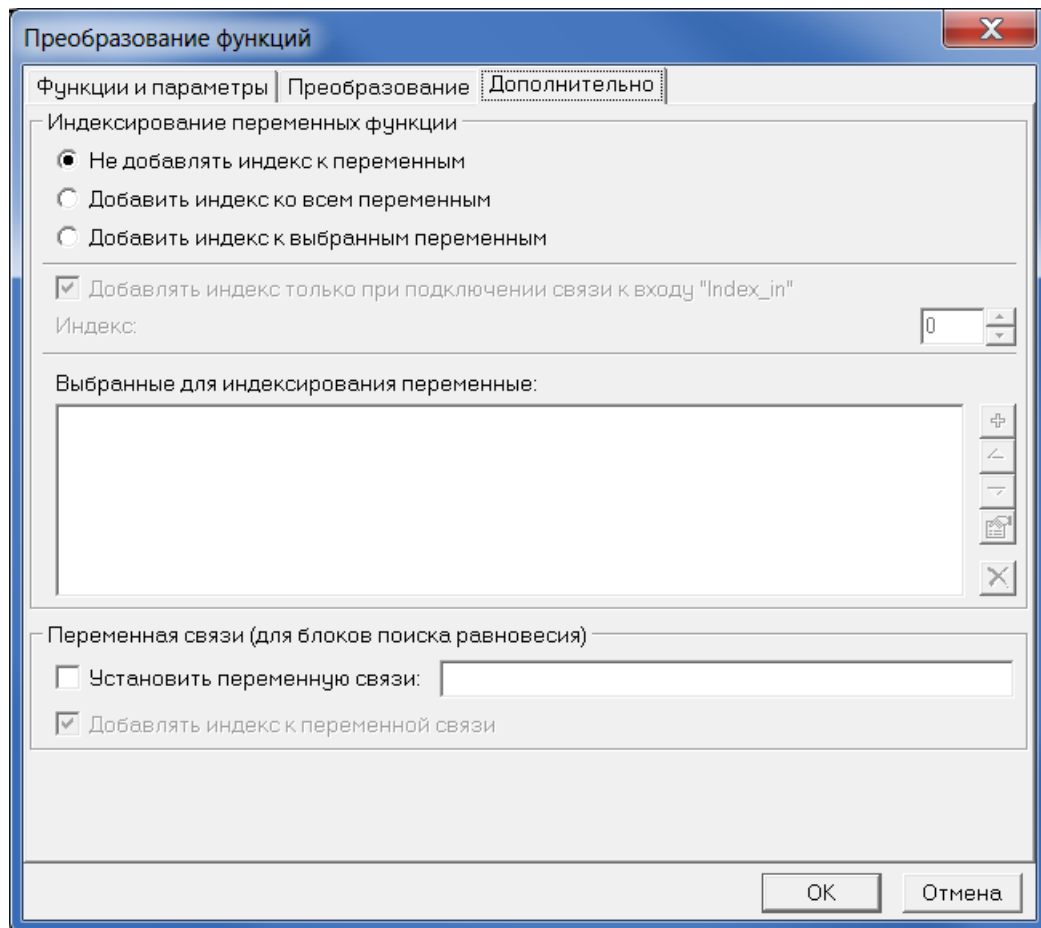
Переменная-результат:

OK Отмена

Вкладка “Преобразование” предназначена для ввода текста программы преобразования входных функций в выходную функцию. На вкладке расположены следующие настройки:

- “Название”. Поле для ввода названия функции. Заполнение не обязательно, используется для пояснения.
- “Текст функции на языке C”. Поле для ввода текста программы преобразования входных функций в выходную в виде выражения или фрагмента программы на языке C (возможно использовать любые операторы C и стандартные функции математической библиотеки). Вызовы входных функций в данной программе преобразования записываются в специфическом формате (отличается от стандартов C). Указанный формат будет описан ниже.
- “Переменная-результат”. Поле для ввода имени переменной, присваивание значения которой можно использовать для возврата значения выходной функции.

Вкладка “Дополнительно”



Вкладка “Дополнительно” предназначена для настройки дополнительных преобразований выходной функции. Вкладка состоит из двух панелей:

- Панель “Индексирование переменных функции”. Индексирование – это добавление к имени аргумента функции числового индекса. Значение индекса либо задается в настройках блока (в поле “Индекс”), либо подается на вход блока *Index_in*. На панели расположены следующие настройки:
 - Группа радиокнопок выбора переменных, к которым будут добавлены индексы
 - “Не добавлять индекс к переменным”. Если выбран данный вариант, то флажок “Добавлять только при подключении связи к входу “Index_in”” станет недоступен
 - “Добавить индекс ко всем переменным”
 - “Добавить индекс к выбранным переменным”. Если выбран данный вариант, то станет доступна для ввода таблица “Выбранные для индексирования переменные:” и панель кнопок для работы с ней
 - “Добавлять только при подключении связи к входу “Index_in””. Если флажок активен, то значение для индекса будет браться только с входа *Index_in* (если подключена связь, и поступившее на него значение неотрицательное, в противном случае индексирование производиться не будет). При этом поле “Индекс” будет недоступно для ввода
При неактивном флажке в случае отсутствия указанной выше связи значение индекса будет браться из поля ввода “Индекс”
 - “Индекс”. Поле для ввода значения добавляемого индекса. Используется при соответствующих настройках, описанных выше.

- Таблица “Выбранные для индексирования переменные:”. Содержит список имен переменных, к которым будет добавляться индекс
- Панель кнопок для работы с таблицей “Выбранные для индексирования переменные:”. С помощью кнопок на данной панели добавляются, редактируются и удаляются строки указанной выше таблицы
- Панель “Переменные связи (для блоков поиска равновесия)”. Панель позволяет задать в формируемой функции имена переменных связи. Имена переменных связи могут не быть аргументами функции, они задаются независимо и используются только в блоке поиска равновесия для связей с другими его функциями. На панели расположены следующие настройки:
 - “Установить переменную связи”. Если флажок активен, то станет доступно поле для ввода имени переменной, которая в выходной функции будет указана в качестве переменной связи
 - “Добавлять индекс к переменной связи”. Если флажок активен, то при включенном индексировании к имени переменной связи (введенному в соответствующее поле) будет добавлен индекс (значение индекса будет взято с входа *Index_in* или из поля “Индекс” панели “Индексирование переменных функции” в зависимости от настроек индексирования)

Описание синтаксиса вызова входных функций

Как уже было упомянуто выше, вызовы входных функций в программе преобразования записываются в специфическом формате (отличается от стандартов C). Приведем описание данного формата и примеры его использования.

Во всех приведенных ниже примерах на одноименные входы поступают функции $f1$ и $f2$:

Входные функции		Параметры функции	
Имя входа блока	Текст программы функции	Имя параметра	Тип параметра
$f1$	$return \sin(w*x);$	x	Число
		w	Число
$f2$	$return A*\cos(x);$	x	Число
		A	Число

Следует учитывать, что формируемая блоком выходная функция внешне не похожа на вычисляемое с ее помощью выражение, т.к. исходные функции в ней скрыты. То есть в настройках одного из принимающих блоков текст функции, полученной в результате преобразования, будет содержать только вызовы исходных функций.

а) Если в данном вызове входная функция должна сохранить параметры и их значения неизменными (т.е. не будут выполняться подстановки и замены), то имя функции записывается без скобок и параметров.

Пример 1

Для указанных выше функций выполним преобразование:

$return f1+f2;$

Результатом преобразования будет функция:

Вычисляемое выражение	Параметры функции	
	Имя параметра	Тип параметра
$\sin wx + A \cdot \cos x$	x	Число
	w	Число
	A	Число

Для вычисления приведенного выше выражения будет сгенерирован следующий текст программы в синтаксисе языка С:

```
return _f_f1(x,w)+_f_f2(A,x);
```

При проведении преобразования поступившие на вход функции становятся внутренними и будут иметь следующий вид:

```
double _f_f1(double x,double w)
{ return sin(w*x); }
double _f_f2(double A,double x)
{ return A*cos(x); }
```

б) Если в данном вызове во входной функции необходимо выполнить подстановку для одного или нескольких параметров (заменить его на константу, другой параметр или сложное выражение, содержащее вызовы других функций), то вызов будет иметь следующий вид:

$\Phi (<\text{имя1}>=<\text{выражение1}> ,<\text{имя2}>=<\text{выражение2}>,...),$

где Φ – имя входной функции;

$<\text{имя1}>, <\text{имя2}>$ – имена параметров, для которых выполняется подстановка;

$<\text{выражение1}>, <\text{выражение 2}>$ – константа, другой параметр или сложное выражение для подстановки.

Может выполняться любое количество подстановок параметров в произвольном порядке. Выражения записываются в синтаксисе С с вставками в виде указанных выше специфических выражений для подстановок. Они могут содержать как собственные параметры входных функций, так и дополнительные параметры, введенные пользователем в настройках блока.

Пример 2

Для указанных выше функций выполним преобразование:

```
return f1(x=y,w=0.5)+f2;
```

Из текста преобразования и входных функций видно, что был добавлен дополнительный параметр y . (Для корректной работы блока необходимо предварительно добавить дополнительный аргумент “ y ” в настройках блока на вкладке “Функции и параметры”). Для “ $f1$ ” выполняются подстановки “ y ” вместо “ x ” и константы “0.5” вместо “ w ”.

Результатом преобразования будет функция:

Вычисляемое выражение	Параметры функции	
	Имя параметра	Тип параметра
$\sin 0.5y + A \cdot \cos x$	x	Число
	y	Число
	A	Число

Пример 3

Для указанных выше функций выполним преобразование:

return f1(w=f2+3);

Для "f1" выполняется подстановка вместо "w" – сумма вызова "f2" с ее собственными параметрами (т.е. "A" и "x") и константы "3".

Результатом преобразования будет функция:

Вычисляемое выражение	Параметры функции	
	Имя параметра	Тип параметра
$\sin((A \cdot \cos x + 3) \cdot x)$	x	Число
	A	Число

Пример 4

Для указанных выше функций выполним преобразование:

return f1(x=f1)+f2(x=f2(x=y));

Из текста преобразования и входных функций видно, что был добавлен дополнительный параметр y. (Для корректной работы блока необходимо предварительно добавить дополнительный аргумент “y” в настройках блока на вкладке “Функции и параметры”). Для функций "f1" выполняется подстановка вместо "x" самой функции "f1", для функций "f2" – вместо "x" самой функции "f2", причем для подставляемой функции "f2" производится замена "x" на "y".

Результатом преобразования будет функция:

Вычисляемое выражение	Параметры функции	
	Имя параметра	Тип параметра
$\sin(w \cdot \sin wx) + A \cdot \cos(A \cdot \cos y)$	x	Число
	y	Число
	w	Число
	A	Число

в) Если в данном вызове функция имеет только один параметр, имя параметра при подстановке можно не указывать.

Пример 5

Если на вход "f1" поступает функция с единственным параметром "x", и при вызове вместо него необходимо подставить значение "2*y", то запись может быть следующей:

*return f1(2*y);*

Следует учитывать, что если у функции более одного параметра, такая запись будет ошибочной и преобразование не будет выполнено.

Описание процедуры индексирования

Индексирование проводится после преобразования, при этом к выбранным именам переменных дописывается числовой индекс, либо введенный в настройках, либо поступивший на вход "Index_in".

Пример 6

Для указанных выше функций выполним преобразование:

return f1(x=y)+f2;

При этом на вкладке "дополнительно" включено индексирование для переменных "w" и "y", т.е. они добавлены в таблицу “Выбранные для индексирования переменные:”, а на

вход *Index_in* перед расчетом поступает значение "3". Для "f1" выполняется подстановка "y" вместо "x". Затем переменные "w" и "y" заменяются на "w3" и "y3" соответственно.

Результатом преобразования и индексирования будет функция:

Вычисляемое выражение (формальное выражение)	Параметры функции	
	Имя параметра	Тип параметра
$\sin(w3 \cdot y3) + A \cdot \cos x$	x	Число
	$w3$	Число
	$y3$	Число
	A	Число

Индексирование переменной связи, если она добавляется в функцию, производится таким же образом.