|  |  |
| --- | --- |
|  | **ООО «3В Сервис»**  РФ, 127051, Москва, ул. Трубная 25 стр 1 офис 6  Тел./ф (495) 221-22-53  [www.3v-services.com](http://www.3v-services.com/) |

**Утверждаю**

генеральный директор

ООО «ЗВ Сервис»

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Петухов В.Н.

****

**Среда динамического моделирования технических систем SimInTech™**

**Техническое задание**

Модуль генерации кода для систем реального времени

Приложение А

Приложение Б

ШИФР ГК16ТЗП

Москва, 2016

**Реферат**

Отчет 109 с., 54 рис., 25 табл.,.

SimInTech, QNX, приборы; графическая оболочка, программное обеспечение, программный комплекс

Целью настоящей работы является разработка системы программирования для вычислительных приборов на базе программного обеспечения SimInTech. Комплексная система моделирования систем управления и программирования приборов, включает в себя:

* генератор кода для автоматической генерации исходных кодов и исполняемых модулей;
* среду разработки для проектирования алгоритмов управления в виде наглядной функциональной блочной диаграммы;
* система исполнения программ для приборов, для выполнения сгенерированных при помощи генератора кода исполняемых модулей на приборах.

Под прибором следует понимать программируемые логические контроллеры (ПЛК) в составе промышленных компьютеров, работающие под управление POSIX совместимых операционных систем реального времени.

Результаты настоящей работы дадут возможность создавать ПО для приборов с меньшим количеством алгоритмических и программных ошибок за счёт отладки алгоритмов на динамической модели объекта и автоматической генерации исходных кодов по отлаженным схемам алгоритмов

Область применения – системы управления.

Содержание

Содержание 3

Термины и определения 4

Введение 5

Приложение А. Описание функционала и совойств блоков, для которых возможна генерация кода 7

Приложение Б. Список функций языка программирования для генерации кода. 99

Лист регистрации изменений 109

Термины и определения

|  |  |  |
| --- | --- | --- |
| ОС | - | операционная система; |
| ОСРВ | - | операционная система реального времени; |
| ПК | - | программные комплекс; |
| ПО | - | программное обеспечение; |

Введение

Руководство пользователя содержит описание правил и приемов работы с системой программирования для вычислительных приборов под управлением операционной системы реального времени QNX.

Опытный образец системы программирования вычислительных приборов разработан на базе программного обеспечения SimInTech.

Данное руководство описывает методику и основные приемы работы с разработанным программным обеспечением для программирования приборов.

Данное руководство предназначено для специалистов осуществляющих разработку алгоритмов управления и программирование приборов.

Приложение А. Описание функционала и совойств блоков, для которых возможна генерация кода

(справочное)

Список бловов, для которых возможна генерация кода представлен в таблице В.1.

Таблица В.1

| Пиктограмма блока | Наименование блока | Описание функционала и свойства блока | Объём памяти данных, байт |
| --- | --- | --- | --- |
| Закладка «Источники» главной панели инструментов SimInTech | | | |
|  | Шаг интегрирования | Блок возвращает текущий шаг интегрирования (расчёта) задачи.  Блок не имеет настраиваемых свойств. | 8 |
|  | Часы | Блок формирует на выходе сигнал, численно равный текущему значению модельного времени задачи.  Блок не имеет настраиваемых свойств. |  |
|  | Константа | Блок формирует на выходе заданную пользователем постоянную величину.  Свойства блока:   * Значение – значение на выходе блока, может быть векторным; * Тип данных (для генерации кода) – тип данных выходной переменной при генерации кода для полномасштабной модели; * Название – название величины, для справки.   Свойство k может быть вектором: k=[k1,k2,…,kn], в этом случае на выходе будет также вектор: y(t)=k. | (datasize\*<размерность значения>)  datasize для типов данных:  Двоичный:1  Целый:4  Вещественный:8 |
|  | Ступенька | Блок формирует значение выходного сигнала, вычисленное по алгоритму:  Свойства блока:   * Время срабатывания – t0; * Начальное состояние – y0; * Конечное состояние – y1.   Свойства могут быть векторами  t0 = [t01,t02,...t0n], y0=[y01,y02,...y0n] и y1 = [y11,y12,...y1n], размерности которых должны быть одинаковы, в этом случае на выходе будет также вектор: |  |
|  | Синусоида | Блок векторизован, формирует синусоидальный выходной сигнал:  Свойства блока:   * Амплитуда – коэффициент а; * Частота – коэффициент ω; * Фаза – коэффициент f;   Свойства могут быть векторами *a* = [*a*1,*a*2,…,*a*n], *ω* = [*ω*1,*ω*2,…,*ω*n] и *f* = [*f*1,*f*2,…,*f*n], размерности которых должны быть одинаковы, в этом случае на выходе будет также вектор:  *y*i(t) = *a*i·sin(*ωit*+*f*i), *i* = 1..n. |  |
|  | Экспонента | Блок формирует значение выходного сигнала, вычисленное по формуле:  Свойства блока:   * Коэффициет усиления – коэффициент a. * Обратная постоянная времени – коэффициент b. * Сдвиг – коэффициент c.   Свойства могут быть векторами, размерности которых должны быть одинаковы, в этом случае на выходе будет также вектор. |  |
|  | Равномерный шум | Блок генерирует псевдослучайную последовательность вещественных чисел, равномерно распределенных в диапазоне от *y*min до *y*max.  Свойства блока:   * Минимальное значение – ymin; * Максимальное значение – ymax; * Период квантования – T – время в секундах, через которое будет обновляться случайный сигнал на выходе блока. Если T = 0, то случайный сигнал на выходе блока будет обновляться после выполнения каждого очередного шага интегрирования.   Свойства могут быть векторами:  *y*min = [*y*min 1, *y*min 2,…,*y*min n],  *y*max= [*y*max 1, *y*max 2,…,*y*max n] и  *T* = [*T*1, *T*2,…,*T*n] , размерности которых должны быть одинаковы, в этом случае на выходе будет также формироваться векторный сигнал. |  |
|  | Кусочно-линейная | Блок формирует на выходе произвольный кусочно-линейный выходной сигнал, являющийся суперпозицией линейных воздействий.  Свойства блока:   * Время – t – массив временных отметок, в секундах; * Значение функции – y – массив значений выходного сигнала.   Свойства являются матрицами, где в каждой строке задается вектор времён или значений для соответствующего элемента выходного вектора |  |
|  | Меандр | Реализует источник прямоугольных импульсов с заданными временными и амплитудными параметрами.  Свойства блока:   * Длительность 1-го полупериода; * Значение 1-го полупериода; * Длительность 2-го полупериода; * Значение 2-го полупериода. | 26\*<размерность длительности 1-го полупериода> |
|  | Циклограмма | Блок реализует функции управляемого функционального генератора и предоставляет возможность задания произвольных периодических функций в виде последовательности опорных точек, которые интерполируются в соответствии с выбранным методом. Позволяет заменить блоки типа «[Меандр](file:///C:\SimInTech\webhelp\biblioteki_blokov\bloki_vse\0015.html#topic_x5n_gv5_qf)», «[Треугольный сигнал](file:///C:\SimInTech\webhelp\biblioteki_blokov\bloki_vse\0014.html#topic_x5n_gv5_qf)», «[Пила](file:///C:\SimInTech\webhelp\biblioteki_blokov\bloki_vse\0011.html#topic_x5n_gv5_qf)», «[Обратная пила](file:///C:\SimInTech\webhelp\biblioteki_blokov\bloki_vse\0012.html)», «[Кусочно-постоянная](file:///C:\SimInTech\webhelp\biblioteki_blokov\bloki_vse\0113.html)» и расширить их возможности. |  |
|  | Управляемый синусоидальный генератор | Блок векторизован, представляет собой управляемый источник синусоидального напряжения. Отличается от блока «[Синусоида](file:///C:\SimInTech\webhelp\biblioteki_blokov\bloki_vse\0006.html)» наличием управляющего входа (StateFlag), а также тем, что задание параметров сигнала происходит снаружи блока.  Свойства блока:  Вывод времени цикла – включение дополнительного выхода блока, показывающего текущее значение времени цикла синусоиды. Изменяется от нуля до T, где T – период синусоиды. |  |
|  |  |  |  |
|  | Входной контакт s3 | Реализует считывание переменной и передачу её на выход блока. В режиме локальной отладки эквивалентен константе. При удалённой отладке блок считывает переменную из исполнительной системы.  Свойства блока:   * Значение по умолчанию – значение на выходе блока, может быть векторным; * Имя контакта – имя переменной s3 считываемой блоком; * Тип контакта – тип данных выходной переменной при генерации кода для полномасштабной модели; * Формат имени переменной – правило формирования имени считываемой переменной; * Добавить в базу сигналов– флаг, указывающий необходимо ли создать данную переменную в базе сигналов системы s3; * Запрос на имя из базы – запрос из базы данных для формирования имени считываемой переменной; * Автоматически сбрасывать – сброс значения переменной в ноль после её считывания; * Операция для нескольких контактов – способ ассоциирования группы переменных и выхода блока; * Функциональный план – имя файла диаграммы, на который необходимо перейти при двойном щелчке по объекту; * Отлаживаемый компонент – имя компонента, который будет установлен при переходе на другую диаграмму в режиме отладки; * Описание входа (не более 65 символов) – текстовое поле с переменной, декларируемое в файлах описания переменных; * Дополнительный шаблон для файла DBM – скрипт для автоматического создания вспомогательных переменных; * Порт сортировки – при включеннии блок сортируется как функциональный строго после того блока в которому подключен вспомогательный порт. | datasize\*2\*<количество сигналов> - для режима упаковки вектора  datasize + datasize\*<количество сигналов> - для режима объединения по ИЛИ.  datasize для типов данных:  Двоичный:1  Целый:4  Вещественный:8 |
|  | Выходной контакт s3 | Реализует запись переменной исполнительной системы со входа блока.  Свойства блока:   * Значение по умолчанию – значение переменной в начальный момент времени (если надо задать отличное от того что вычисляется автоматически); * Имя контакта – имя переменной s3 считываемой блоком. * Формат имени переменной – правило формирования имени считываемой переменной. * Добавить в базу сигналов – флаг, указывающий необходимо ли создать данную переменную в базе сигналов системы. * Запрос на имя из базы – запрос из базы данных для формирования имени считываемой переменной. * Транслировать в исполнительную систему – при выставлении данного флага происходит трансляция входа в переменную на исполнительной системе; * Применять операцию ИЛИ – если установлен данный флаг, то <переменная> = <переменная> ИЛИ <вход блока>; * Операция для нескольких контактов – способ привязки входа блока к группе переменных; * Функциональный план – имя файла диаграммы, на который необходимо перейти при двойном щелчке по объекту. * Отлаживаемый компонент – имя компонента, который будет установлен при переходе на другую диаграмму в режиме отладки; * Описание выхода (не более 65 символов) – текстовое поле с описанием; * Порт сортировки – при включеннии блок сортируется как функциональный строго после того блока в которому подключен вспомогательный порт. | datasize\*<количество сигналов> - для режима упаковки вектора  datasize для типов данных:  Двоичный:1  Целый:4  Вещественный:8 |
| Закладка «Операторы» главной панели инструментов SimInTech | | | |
|  | Сумматор | Реализует поэлементное суммирование входных величин.  Свойства блока:   * Весовые множители для каждого из входов – числа на которые умножаются значения входных портов блока. Число портов блока равно размерности этого параметра. | 8\*<размерность выхода> |
|  | Сравнивающее устройство | То же самое, что и сумматор, но по умолчанию весовые множители равны [1, -1].  Свойства блока:   * Весовые множители для каждого из входов – числа на которые умножаются значения входных портов блока. Число портов блока равно размерности этого параметра. | 8\*<размерность выхода> |
|  | Сложение вектора с числом | Реализует суммирование каждого из элементов векторной величины на первом входе со скалярной величиной на втором входе. Размерность выхода равна размерности первого входа.  Блок не имеет настраиваемых свойств. | 8\*<размерность выхода> |
|  | Суммирование элементов вектора | Реализует суммирование всех элементов векторов входов. Выход – скалярная величина.  Свойства блока:   * Весовые коэффициенты входов – числа, на которые умножаются значения входных портов блока. Число портов блока равно размерности этого параметра; * Суммировать через (к-во элементов, k) – задаёт правило суммирования элементов – т.е. единичному элементу выхода сопоставляются не все элементы входа, а Nвх/k | 4+8\*<размерность выхода> |
|  | Перемножитель | Реализует поэлементное умножение входных величин блока.  Свойства блока:   * Количество портов – количество портов блока. | 8\*<размерность выхода> |
|  | Перемножение элементов вектора | Реализует перемножение всех элементов векторов входов. Выход – скалярная величина.  Свойства блока:   * Количество входов – указывает к-во входов блока (и сотвественно перемножаемых значений). | 8\*<размерность выхода> |
|  | Умножение на число | Реализует умножение каждого из элементов векторной величины на первом входе со скалярной величиной на втором входе.  Блок не имеет настраиваемых свойств. | 8\*<размерность выхода> |
|  | Делитель | Реализует поэлементное деление первой входной величины на вторую.  Блок не имеет настраиваемых свойств. | 8\*<размерность выхода> |
|  | Деление скаляра на вектор | Реализует деление первой скалярной входной величины на каждый из элементов второй величины. Размерность выхода равна размерности второго входа.  Блок не имеет настраиваемых свойств. | 8\*<размерность выхода> |
|  | Усилитель | Производит поэлементное умножение входной величины на заданный коэффициент усиления. Размерность выхода равна размерности входа.  Свойства блока:   * Коэффициент усиления – скалярный множитель для входной величины. | 8+8\*<размерность выхода> |
|  | Векторный усилитель | Производит умножение элементов входной величины на элементы заданного векторного коэффициента усиления. Размерность выхода равна размерности входа и коэффициента усиления.  Свойства блока:   * Вектор к-в усиления – массив множителей на которые домножается каждый элемент входа. | 16\*<размерность выхода> |
|  | Абсолютное значение | Реализует вычисление модуля каждого из элементов входной векторной величины.  Блок не имеет настраиваемых свойств. | 8\*<размерность выхода> |
|  | Размножитель | Реализует умножение элементов входа на векторы коэффициентов усиления.  Yi = ki\*U – для каждого элемента входа.  где Yi – i-й элемент выхода;  ki – i-й множитель;  U - i-й элемент входа.  Свойства блока:   * Коэффициенты размножения – множители на которые домножается входная величина. | 16\*<размерность выхода> |
|  | Знак | Реализует вычисление знака каждого из элементов входной векторной величины.  -1 – число отрицательное.  0 – число равно нулю.  1 – число больше нуля. | 8\*<размерность выхода> |
|  | Номер активного элемента | Блок выдает порядковый номер первой единицы в векторе двоичных сигналов, поступающем на вход.  Блок не имеет настраиваемых свойств. |  |
|  | Динамическая выборка | Блок выполняет процедуру выбора элемента из вектора по указанному номеру. Алгоритм работы блока:  где *u*1(*t*) – первых входной скалярный сигнал, определяющий номер выбираемого элемента, начиная с 1; *u*2,i(*t*) – i-ый элемент второго входного векторного сигнала *u*2(*t*), из которого осуществляется выборка; *n*≥1 – размерность второго входа, натуральное число.  Блок не имеет настраиваемых свойств. |  |
|  | Стоп-расчет | Блок выполняет процедуру остановки расчёта или временное прерывание расчета при вхождении значения входного сигнала в заданный диапазон:  Где *u*(*t*) – входной скалярный сигнал;  *u*min(*t*), *u*max(*t*) – границы стоп-интервала.  Свойства блока:   * Левая граница стоп-интервала – число *u*min; * Правая граница стоп-интервала – число *u*max. * Сообщение при остановке расчёта – текст, выводимый пользователю при вхождении сигнала в стоп-интервал и, следовательно, остановке расчета. |  |
|  | Линейный преобразователь | Реализует линейную интерполяцию входной величины. При этом значение входа равное минимальной входной величине соответствует нулю на выходе, а значение входа равное максимальной входной величине соответствует единице на выходе.  Свойства блока:   * Минимальное значение входа – значение входной величины соотвествущее нулю на выходе блока; * Максимальное значение входа – значение входной величины соотвествущее единице на выходе блока. | 24\*<размерность выхода> |
|  | Выборка по активному элементу | Блок выполняет процедуру выбора элемента из вектора по активному элементу управляющего вектора по следующему алгоритму:  ЦИКЛ i ОТ 1 ДО N  ЕСЛИ u1[i] > 0 TO Y=u2[i];ВЫХОД ИЗ ЦИКЛА  КОНЕЦ ЦИКЛА  где u1 – управляющий вектор;  u2 – вектор выбираемых значений;  n – размерность входов.  Блок не имеет настраиваемых свойств. | datasize  datasize для типов данных входа блока:  двоичный:1  целый:4  вещественный:8 |
|  | Целая часть | Блок реализует операцию выделения целой части входного сигнала. Под целой частью понимается ближайшее целое число в меньшую сторону (по абсолютной величине). То есть целая часть положительного числа – меньше или равна числу, а целая часть отрицательного числа – больше или равна числу.  Блок не имеет настраиваемых свойств. |  |
|  | Дробная часть | Блок реализует операцию выделения дробной части входного сигнала. Под дробной частью понимается разность между сигналом и его целой частью. Дробная часть положительного числа – неотрицательна, а дробная часть отрицательного числа – неположительна.  Блок не имеет настраиваемых свойств. |  |
|  | Округление | Выполняет округление (ограничение точности) входного действительного числа с задаваемой пользователем точностью.  Свойства блока:  Шаг точности – параметр *p.* |  |
| Закладка «Векторные» главной панели инструментов SimInTech | | | |
|  | Мультиплексор | Блок реализует «сжатие» нескольких скалярных или векторных входных сигналов в один векторный выходной сигнал типа «шина» данных. Размерность выходного вектора равна сумме размерностей входов.  Свойства блока:   * Количество портов – количетсво портов входа блока. | datasize\*<размерность выхода>  datasize для типов данных входа блока:  двоичный:1  целый:4  вещественный:8  тип выхода определяется исходя из максимальной длины типа данных входа. |
|  | Демультиплексор | Блок реализует «расщепление» векторного входного сигнала в отдельные выходные сигналы. Выходами могут быть как скалярные, так и векторные («многожильные») сигналы. По умолчанию данный блок реализует демультиплексирование «двухжильного» входного векторного сигнала в скалярные сигналы.  Свойства блока:   * Массив размерностей выходов – массив, описыващий количество элементов для каждого из выходов блока. | datasize\*<сумма размерностей выходов>  datasize для типов данных входа блока:  двоичный:1  целый:4  вещественный:8 |
|  | Распаковка матрицы | Блок реализует преобразование одного входного векторного сигнала на несколько векторных выходных сигналов.  Свойства блока:   * Число строк в матрице – число M. * Число столбцов в матрице – число N. * Матрица запакована по – вариант запаковки входной матрицы: по строкам или по столбцам. * Матрицу распаковать по – вариант «распаковки» входной матрицы. |  |
|  | Выборка из вектора | Блок реализует выборку элементов из входного векторного сигнала в соответствии с заданным типом выборки.  Свойства блока:   * Тип выборки:  1. Ручная 2. Четные элементы 3. Нечетные элементы 4. Первые N элементов 5. Последние N элементов 6. В обратном порядке 7. По возрастанию 8. По убыванию  * Номера элементов (ручная выборка) – номера элементов для типа выборки Ручная. * Параметр N – количество элементов для типов выборки Первые N элементов и Последние N элементов. |  |
| Закладка «Динамические» главной панели инструментов SimInTech | | | |
|  | Язык программирования | Реализует произвольный алгоритм, написанный на встроенном языке программирования комплекса. При генерации кода для блока может быть задан шаблон генерации произвольного кода на языке программирования Си.  Свойства блока:   * Тип сортировки – флаг, определяющий порядок расчёта блока в общей схеме; * Типы выходов – список, в котором можно принудительно задать типы каждого из выходов блока, отличающиеся от заданных в скрипте; * Заголовок модуля – заголовок текста при генерации кода, здесь декларируются специальные комментарии для системы s3; * Секция декларации переменных – текст, вставляемый в начало программы при генерации кода; * Основная секция кода – основной текст программы при генерации кода для блока; * Секция запоминания состояний – текст, вставляемый в конец программы при генерации кода, здесь производиться вычисление переменных состояния; * Декларации переменных DBM – текст, вставляемый в select-файл для описания переменных состояния блока; * Значения переменных по умолчанию – тест для установки начальных значений переменных состояния блока. * Код вычисления производных – блок кода на Си, который вызывается при вычислении производных динамических блоков; * Коды вычисления алгебраических переменных – блок кода на Си, который вызывается при вычислении алгебраических функций; * Код остановки задачи – блок кода на Си, вызываемый при остановке вычисления. | Все пользовательские данные задаются в блоке впрямую как шаблон. Размер блока зависит от того что задаётся в шаблоне. По умолчанию размер данных равен сумме прозведений размерностей выходов на размер их типов данных.  Размер для типов данных выходов:  двоичный:1  целый:4  вещественный:8 |
|  | Интегратор | Реализует интегрирование элементов вектора входной величины.  Свойства блока:   * Коэффициенты усиления – коэффициент, на который умножается входная величина; * Начальные условия – начальное значение выходной величины блока. | 24\*<размерность входа> |
|  | Инерционное звено 1-го порядка | Реализует вычисление выхода апериодического (инерционного) звена первого порядка.  Свойства блока:   * Коэффициенты усиления – коэффициент, на который умножается входная величина. * Постоянные времени; * Начальные условия – начальное значение выходной величины блока. | 32\*<размерность входа> |
|  | Идеальное транспортное запаздываение | Блок векторизован, реализует математическую модель звена, динамика которого описывается линейным уравнением (ЛУ) с запаздыванием:  Где *xi*(*t*) – i-ый элемент входного сигнала в блок, *xi*(*t*-*Ti*) – значение *i*-го элемента входного сигнала, запомненное в блоке *Ti* секунд назад, по отношению к текущему модельному времени; *Ti* – элемент вектора времён запаздывания, *yi*(*t*) – элемент выходного сигнала из блока.  Блок не имеет настраиваемых параметров. |  |
|  | Инерционно-дифференцирующее звено | Реализует вычисление инерционно-дифференцирующего звена.  Свойства блока:   * Коэффициенты усиления – коэффициент, на который умножается входная величина. * Постоянные времени. * Начальные условия – начальное значение выходной величины блока. | 40\*<размерность входа> |
|  | Производная | Блок реализует численное определение производной:  на каждом временном шаге как отношение приращений динамической переменной и времени, где *xi*(*t*) – i-ый элемент входного сигнала в блок, *yi*(*t*) – i-ый элемент выходного сигнала из блока. Размерности сигналов входа и выхода совпадают.  Блок не имеет настраиваемых параметров. |  |
|  | Инерционно-интегрирующее звено | Реализует вычисление выхода инерционно-интегрирующего звена.  Свойства блока:   * Коэффициенты усиления – коэффициент, на который умножается входная величина; * Постоянные времени; * Начальные условия – начальное значение выходной величины блока; * Начальные условия по производной – начальные условия для производной выходной величины. | 48\*<размерность входа> |
|  | Интегратор с ограничением | Реализует интегрирование элементов вектора входной величины с ограничением значения выходной величины.  Свойства блока:   * Коэффициенты усиления – коэффициент, на который умножается входная величина; * Минимальное значение – минимальное значение выхода; * Максимальное значение – максимальное значение выхода; * Начальные условия – начальное значение выходной величины блока. | 40\*<размерность входа> |
|  | Интегратор с изменяемыми начальными условиями | Блок реализует интегрирование входного сигнала (сигналов):  где *xi*(*t*) – элементы вектора входного сигнала, *yi*(*t*) – элементы вектора выходного сигнала, *t*1 и *t*2 – времена переключения начальных условий.  Свойства блока:  Способ сброса - позволяет выбрать момент выполнения сброса. |  |
|  | Интегратор с ограничением и сбросом начальных условий | Блок реализует интегрирование входного сигнала (сигналов) с ограничением на значение интеграла и сбросом состояния:  где *xi*(*t*) – элементы вектора входного сигнала, *yi*(*t*) – элементы вектора выходного сигнала, *t*1 и *t*2 – времена переключения начальных условий.  Свойства блока:   * Коэффициенты усиления – вектор коэффициентов ki, на который умножается входная величина; * Начальные условия – вектор начальных значений yi(0) выходной величины блока; * Минимальное значение – минимальное значение выхода ymin; * Максимальное значение – максимальное значение выхода ymax; * Начальные условия брать из – настройка, откуда блок будет считывать начальные условия, а именно: из свойства «Начальные условия», или из дополнительного входного порта Y0. |  |
|  | Интегратор с ограничением и сбросом начальных условий | Блок реализует интегрирование входного сигнала (сигналов) с ограничением на значение интеграла и сбросом состояния:  где *xi*(*t*) – элементы вектора входного сигнала, *yi*(*t*) – элементы вектора выходного сигнала, *t*1 и *t*2 – времена переключения начальных условий, *k*i – элемент вектора коэффициентов усиления. |  |
|  | Круговой интегратор | Блок векторизован, реализует математическую модель звена, динамика которого описывается линейным обыкновенным дифференциальным уравнением (ОДУ) вида:  где *xi*(*t*) – i-ый элемент входного сигнала в блок, *ki* – элемент вектора коэффициентов усиления, *yi*(*t*) – элемент выходного сигнала из блока.  Свойства блока:   * Коэффициенты усиления – вектор коэффициентов ki, на который умножается входная величина; * Начальные условия – вектор начальных значений yi(0) выходной величины блока. |  |
|  | Аналитическая апериодика 1-го порядка | Реализует вычисление инерционного звена первого порядка по аналитической формуле.  Свойства блока:   * Коэффициенты усиления – коэффициент, на который умножается входная величина; * Постоянные времени – вектор постоянных времени звена для каждого из элементов входного вектора; * Начальные условия – начальное значение выходной величины блока; * Дополнительные входы – указывает наличие дополнительных входов блока; * Начальные условия берутся из – указатель источника начальных условий блока. | 24\*<размерность входа> |
|  | Дискретная апериодика 1-го порядка | Реализует вычисление выхода апериодического (инерционного) звена 1-го порядка по неявной численной схеме.  Свойства блока:   * Коэффициенты усиления – коэффициент, на который умножается входная величина; * Постоянные времени; * Начальные условия – начальное значение выходной величины блока; * Дополнительные входы – указывает наличие дополнительных входов блока; * Начальные условия берутся из – указатель источника начальных условий блока. | 24\*<размерность входа> |
|  | Интегратор на усилителях | Блок выполняет численное интегрирование с ограничением. Блок векторизован, является субмоделью (сборкой). Для того чтобы посмотреть структуру алгоритма — нажмите правой кнопкой Действия → Войти в субмодель. Алгоритм работы блока реализует численное интегрирование входного сигнала (сигналов).  Свойства блока:   * Число переменных — количество интегрируемых переменных во входном векторе; * Начальные состояние — значения начальных состояний для интегралов входных переменных. |  |
|  | ДИФ-производная | Выполняет численное дифференцирование, с использованием блока «Дискретная апериодика 1-го порядка». Блок векторизован, представляет собой субмодель (сборку).  Свойства блока:   * Число переменных — количество дифференцируемых переменных во входном векторе. |  |
|  | Фильтрация сигнала | Выполняет численную фильтрацию, с использованием блока «Дискретная апериодика 1-го порядка». Блок представляет собой субмодель (сборку).  Свойства блока:   * Постоянная демпфирования — параметр, задающий постоянную времени для апериодики 1-го порядка и определяющая степень «сглаживания» или фильтрации входного сигнала. * Число переменных — число переменных во входном векторе. |  |
| Закладка «Нелинейные» главной панели инструментов SimInTech | | | |
|  | Линейное с насыщением | Блок векторизован и реализует преобразование вектора входных сигналов посредством единой нелинейной статической характеристики типа «насыщение» по следующему алгоритму:    y ( t ) = K \* x ( t ), если a < x ( t ) < b;  y ( t ) = y1, если x ( t ) <= a;  y ( t ) = y2, если x ( t ) >= b,  где x(t) – вектор входных сигналов;  y(t) - вектор выходных сигналов; К = (y2-y1)/(b - a).  Свойства блока:   * Нижняя граница аргумента – a; * Верхняя граница аргумента – b; * Нижнее зн-е функции – y1; * Верхнее зн-е функции – y2. | 32 + 8\*<размерность входа> |
|  | Линейное с зоной нечувствительности | Блок реализует преобразование вектора входных сигналов посредством единой нелинейной статической характеристики типа "линейная с зоной нечувствительности" по следующему алгоритму:    где *x*(*t*) – вектор входных сигналов, *y*(*t*) – вектор выходных сигналов, *k* – коэффициент усиления.  Свойства блока:   * Нижняя граница аргумента – параметр a. * Верхняя граница аргумента – параметр b. * Коэффициент усиления – параметр k. |  |
|  | Релейное неоднозначное (гистерезис) | Блок реализует релейное звено с гистерезисом.  Свойства блока:   * Нижняя граница переключения – минимальное значение входной величины, при котором значение на выходе равно нижнему значению; * Верхняя граница переключения – максимальное значение входной величины, при котором значение на выходе равно верхнему значению; * Нижнее зн-е функции – минимальное значение на выходе блока; * Верхнее зн-е функции – максимальное значение на выходе блока; * Начальные условия (0;1) – начальное состояние реле (включено/выключено). | 32 + <размерность входа> |
|  | Релейное c зоной нечувствительности | Реализует релейное звено с гистерезисом и зоной нечувствительности.  Свойства блока:   * Нижняя граница переключения – минимальное значение входной величины, при котором значение на выходе равно нижнему значению; * Нижняя граница зоны нечувствительности; * Верхняя граница зоны нечувствительности; * Верхняя граница переключения – максимальное значение входной величины, при котором значение на выходе равно верхнему значению; * Нижнее зн-е функции – минимальное значение на выходе блока; * Верхнее зн-е функции – максимальное значение на выходе блока; * Начальные условия (0;1) – начальное состояние реле (включено/выключено). | 48 + 5\*<размерность входа> |
|  | Минимум по входам | Блок производит поэлементное вычисление минимального или максимального из текущих входных значений:  Yi = min(u1i, u2i),  где i – элемент вектора;  u1 – первый входной вектор;  u2 – второй входной вектор;  Y – выходной вектор.  Свойства блока:   * Количество портов – количество портов блока; * Тип операции – тип операции: минимум или максимум; | datasize\*<размерность выхода>  datasize для типов данных выходов:  двоичный:1  целый:4  вещественный:8 |
|  | Минимум по всем элементам вектора | Блок производит вычисление минимального или максимального значений из элементов входного вектора:  Y= min(un,i),  где i – элемент вектора;  un,i – первый входной вектор;  Y – выходной вектор.  Свойства блока:   * Количество портов – количество портов блока; * Тип операции – тип операции: минимум или максимум. | datasize  datasize для типов данных выходов:  двоичный:1  целый:4  вещественный:8 |
|  | Излом | Блок векторизован и реализует преобразование вектора входных сигналов посредством единой нелинейной статической характеристики типа "Излом" по следующему алгоритму:    y ( t ) = k1 \* x ( t ), если x ( t ) < x0;  y ( t ) = k2 \* x ( t ), если x ( t ) >= x0,  где x(t) – вектор входных сигналов; y(t) - вектор выходных сигналов.  Параметры:   * К при X<0 – коэффициент k1 * K при X>0 – коэффициент k2 * Точка излома – x0 | 24 + 8\*<размерность входа> |
|  | Ломаная статическая характеристика | Блок реализует преобразование вектора входных сигналов посредством единой однозначной нелинейной статической характеристики произвольного вида, аппроксимируемой кусочно-ломаной кривой:    Если входной сигнал меньше *x*1 или больше *xn*, то на выходе блока формируется соответствующая постоянная величина:  где *x*(*t*) – вектор входных сигналов; *y*(*t*) – вектор выходных сигналов.  Свойства блока:   * Вектор значений аргумента – массив аргументов xi; * Вектор значений функции – массив значений выходного сигнала yi; * Экстраполировать границы - опция управляет поведением функции за пределами области определения. | 8\*<размерность входа> + 16\*<размерность вектора аргументов функции> |
|  | Многомерная интерполяция | Блок позволяет осуществлять линейную интерполяцию многомерной таблично-заданной функции.  Свойства блока:   * Матрица аргументов по размерностям - сетка известных аргументов функции, заданная матрицей [[значения аргументов по размерности 1],... [значения сетки по размерности N]]; * Вектор значений функций - массив значений заданной многомерной функции, в точках, задаваемых известными аргументами. Если N=3 и размерности x[1], x[2] и x[3] равны соответственно nx, ny, nz, то номер элемента (начиная с 0) массива функции, соответствующий данным координатам равен i+nx(j+ny k), где i, j, k – номера, под которыми стоят координаты в векторах x[1], x[2] и x[3]. Для значений координат не являющихся табличными выполняется линейная интерполяция по соседним точкам. * Метод интерполяции - желаемый способ интерполяции: линейная между двумя соседними точками, либо ступенчатая, сохраняющая значение предыдущей точки; * Экстраполировать границы — способ определения значения функции за пределами сетки аргументов. |  |
|  | Интерполяция | Блок предназначен для интерполяции таблично заданных функций. Блок имеет три векторных входа и один векторный выход. Реализованы методы интерполяции полиномами Лагранжа и кубическими сплайнами. Порядок интерполяции (Order) и индекс начала интерполяции (Ind) имеют значение только для интерполяции полиномами Лагранжа. Значения Order и Ind определяют области интерполяции и экстраполяции. Внутри интервала [x(Ind), x(Ind+Order)] функция интерполируется, вне этого интервала – экстраполируется. При интерполяции сплайнами на границах интервала задано условие равенства 0 вторых производных. |  |
|  | Зазор | Блок реализует преобразование вектора входных сигналов посредством единой нелинейной статической характеристики типа "Зазор" при различных начальных условиях по следующему алгоритму:    где *x*(*t*) – вектор входных сигналов; *y*(*t*) – вектор выходных сигналов, Δt – предыдущий временной шаг интегрирования.  Свойства блока:   * Половина ширины зоны зазора – b; * Коэффициент усиления – k; * Начальные условия – вектор начальных условий y(0). |  |
|  | Люфт | Блок реализует преобразование вектора входных сигналов посредством единой нелинейной статической характеристики типа "Люфт" при различных начальных условиях по следующему алгоритму:    где *x*(*t*) – вектор входных сигналов; *y*(*t*) – вектор выходных сигналов, Δ*t* – предыдущий временной шаг интегрирования.  Свойства блока:   * Половина ширины зоны зазора – b; * Коэффициент усиления – k; * Начальные условия – вектор начальных условий y(0); * Нинжнее значение функции – y1; * Верхнее значение функции – y1. |  |
|  | Квантатель по уравнению | Блок производит квантование сигнала по дискретным уровням с заданным шагом по уровню. Алгоритм работы блока:  где *xi*(*t*) – *i*-ый элемент входного сигнала в блок, *di* – *i*-ый элемент вектора шагов квантования по уровню, *floor* – функция округления до меньшего целого числа, *sign* – функция знака, *yi*(*t*) – элемент выходного сигнала из блока.  Свойства блока:   * Шаг квантования – вектор d шагов квантования по каждому элементу входного вектора. Размерность параметра должна совпадать с размерностью входного вектора. |  |
|  | Запоминание значений сигналов | Блок производит запоминание сигнала (сигналов) по следующему алгоритму:  где *x*1,*i*(*t*), *x*2,*i*(*t*) – i-ые элементы первого и второго векторных входных сигналов, *yi*(*t*) – *i*-ый элемент векторного выходного сигнала.  Блок не имеет настраиваемых свойств. |  |
|  | Запоминие длительности истинного сигнала | Блок производит запоминание длительности сигнала (длительности наличия сигнала или сигналов на входе) по следующему алгоритму:  где xᵢ(t) – i-й элемент первого векторного входного сигнала, yᵢ(t) – i-ый элемент векторного выходного сигнала, t – текущее модельное время,– временная отметка начала импульса.  Блок не имеет настраиваемых свойств. |  |
|  | Задержка на шаг интегрирования | Блок реализует задержку входной величины на один шаг моделирования (интегрирования):  где *xi*(*t* - Δ*t*) – i-ый элемент входного сигнала в блок на предыдущем шаге моделирования, *yi*(*t*) – элемент выходного сигнала из блока.  Свойства блока:   * Развязка алгебраических петель – флаг, показывающий как должен сортироваться блок. Если указано что должен, то блок будет отсортирован в первую очередь. Другими словами это означает, что блок используется для развязки алгебраических петель. * Начальные условия – начальные условия (вектор начальных условий) на выходе блока на первом шаге задачи. |  |
|  | Субмодель релейное с переменными ЗН и ЗВ | Реализует субмодель для реле с гистерезисом и зоной нечувствительности с переменными параметрами.  Блок не имеет настраиваемых свойств. | 80\*<размерность входа> + 56 |
|  | Субмодель релейное плюс с переменными ЗН и ЗВ | Реализует субмодель для реле с гистерезисом и зоной нечувствительности с переменными параметрами.  Блок не имеет настраиваемых свойств. | 80\*<размерность входа> + 56 |
|  | Субмодель релейное минус с переменными ЗН и ЗВ | Реализует субмодель для реле с гистерезисом и зоной нечувствительности с переменными параметрами.  Блок не имеет настраиваемых свойств. | 80\*<размерность входа> + 56 |
|  | Сигнал больше уставки | Блок представляет собой релейное звено с зоной нечувствительности: на выходе блок формирует логическую 1, если входной сигнал превысил значение уставки, и логический 0, если сигнал стал меньше уставки (с учетом зоны возврата, если она задана ненулевой):  где x(t) – входной сигнал; y(t) – выходной сигнал; a, x0 – величина зоны возврата и уставка. Блок представляет собой субмодель.  Свойства блока:   * Уставка - величина уставки, с которой сравнивается входной сигнал. Другими словами, верхняя граница переключения релейного звена. * Зона возврата - величина, которая вычитается из уставки для формирования нижней границы переключения релейного звена. |  |
|  | Сигнал меньше уставки | Блок представляет собой релейное звено с зоной нечувствительности: на выходе блок формирует логическую 1, если входной сигнал меньше значения уставки, и логический 0, если сигнал стал больше уставки (с учетом зоны возврата, если она задана ненулевой):  где x(t) – входной сигнал; y(t) – выходной сигнал; a, x0 – величина зоны возврата и уставка. Блок представляет собой субмодель.  Свойства блока:   * Уставка - величина уставки, с которой сравнивается входной сигнал. Другими словами, верхняя граница переключения релейного звена. * Зона возврата - величина, которая вычитается из уставки для формирования нижней границы переключения релейного звена. |  |
|  | ПОР ИКР (Порог срабатывания) | Блок представляет собой релейное звено с зоной нечувствительности, на выходе формирует логическую 1 или 0 по следующему алгоритму: На вход блока подаются два сигнала, которые сравниваются между собой. Если их разность превысит порог срабатывания, то блок формирует логическую 1 на выходе. Если разность между первым и вторым входом станет меньше порога срабатывания (с учетом зоны возврата), то на выходе блок формирует логический 0:  где *x*1(*t*), *x*2(*t*) – входные сигналы блока; *y*(*t*) – выходной сигнал; *a*, *ε*0 – величина зоны возврата и порог срабатывания.  Свойства блока:   * x1 - первый входной сигнал; * x2 - первый входной сигнал. |  |
|  | НОР ИКП | Блок представляет собой релейное звено с зоной нечувствительности и с двумя выходами, на выходах формирует логические 1 или 0 по следующему алгоритму: На входе в блок подаются два сигнала, которые сравниваются между собой. Если их разность превысит верхний предел, то блок формирует логическую 1 на первом выходе. Если разность между первым и вторым входом станет меньше нижнего предела, то на втором выходе блок формирует логическую 1. В других случаях (если разность находится между верхним и нижним пределами, с учетом величины зоны возврата), на выходах формируется логический 0:  где *x*1(*t*), *x*2(*t*) – входные сигналы блока; *y*1(*t*), *y*2(*t*) – выходные сигналы; *a*, *ε*1, *ε*2 – величина зоны возврата и пределы изменения разности (пределы срабатывания).  Свойства блока:   * x1 - первый входной сигнал; * x2 - первый входной сигнал. |  |
|  | Гистерезис входного сигнала | Блок при помощи релейного неоднозначного звена типа [Гистерезис](file:///D:\%D0%97%D0%B0%D0%B4%D0%B0%D0%BD%D0%B8%D0%B5%20%D0%9F%D0%95%D1%82%D1%83%D1%85%D0%BE%D0%B2%20%D0%BE%D1%82%2009.03.2017\webhelp\biblioteki_blokov\bloki_vse\0071.html) формирует на выходе два признака — превысил ли сигнал на первом входе верхний предел, или снизился ли он до уровня менее нижнего предела. При этом учитывается наличие сигнала на втором логическом входе — при отсутствии сигнала на нём блок «не работает» и оба выхода равны логическому нулю. Блок имеет один аналоговый вход и один дискретный вход - разрешение работы. Выходы дискретные, второй из них – инверсный. Зависимость выходов от аналогового входа – нелинейная, с гистерезисом:  Свойства блока:   * верхний предел - значение верхнего предела сигнала; * нижний предел - значение нижнего предела сигнала. |  |
|  | Мертвая зона (изменяемая) | Блок реализует алгоритм линейного звена с коэффициентом усиления равным 1 и зоной нечувствительности, которая изменяется в зависимости от входного сигнала на втором входном порте. Блок представляет собой субмодель. Для того чтобы посмотреть структуру алгоритма – нажмите правой кнопкой Действия → Войти в субмодель:  Блок не имеет настраиваемых свойств. |  |
|  | ФНЗ (формирование зоны нечуствительности) | Блок реализует алгоритм линейного звена с коэффициентом усиления равным 1 и зоной нечувствительности, задаваемой в параметрах блока. Блок представляет собой субмодель. Для того чтобы посмотреть структуру алгоритма – нажмите правой кнопкой Действия → Войти в субмодель:    Свойства блока:   * Зона нечувствительности — абсолютная величина верхней и нижней границы переключения для блока «Релейное с зоной нечувствительности». * Зона возврата — величина зоны возврата для верхней и нижней границ переключения. |  |
|  | Импульсное открытие/закрытие | Блок импульсно выдает логическую единицу на своем выходе, на протяжение всего времени, пока подана единица на вход. Длина импульса и длина паузы задаются в секундах в свойствах блока.  Свойства блока:   * Длительность импульса, с - продолжительность выходного импульса в секундах; * Длительность паузы, с - продолжительность паузы между выходными импульсами. |  |
|  | Запоминание сигнала | Блок передает входной аналоговый сигнал на выход, если разрешена трансляция (x2=1). Если признак трансляции равен 0, то на выходе фиксируется значение, которое было на момент запрещения трансляции. Если в момент инициализации проекта трансляция запрещена, то на выходе блока будет нулевой сигнал. На изображении блока на схеме отображается значение выходного сигнала, которое равно запомненному либо совпадает с входным, если трансляция разрешена. Таблица истинности блока выглядит следующим образом:  Свойства блока:   * Выход блока - значение выхода блока если трансляция входа не происходит. |  |
|  | Линейное с переменным ограничением | Блок реализует алгоритм линейного звена с коэффициентом усиления равным 1 и с ограничением сигнала, которое изменяется в зависимости от входных сигналов на втором и третьем входных портах. Алгоритм работы блока:  где *y*1(*t*), *y*2(*t*), *y*3(*t*) – сигналы на выходных портах, *x*(*t*) – сигнал на первом входном порте (слева), *x*min(*t*), *x*max(*t*) – сигнал на втором и третьем входных портах (сверху).  Блок не имеет настраиваемых свойств. |  |
| Закладка «Субструктуры» главной панели инструментов SimInTech | | | |
|  | Субмодель | Блок, содержащий внутри себя вложенную схему. Для доступа к содержимому субмодели необходимо произвести по нему двойной щелчок. | |
|  | Порт входа | Блок, который вставляется внутрь субмодели и определяющий наличие у субмодели входного порта. Служит для соединения внешней части схемы со схемой находящейся внутри субмодели. Для указания порта соединения необходимо произвести двойной щелчок по блоку. | |
|  | Порт выхода | Блок, который вставляется внутрь субмодели и определяющий наличие у субмодели выходного порта. Служит для соединения внешней части схемы со схемой находящейся внутри субмодели. Для указания порта соединения необходимо произвести двойной щелчок по блоку. | |
|  | В память | Блок предназначенный для соединения одной части схемы с другой, находящейся в произвольном месте проекта. Для указания названия точки соединения необходимо произвести двойной щелчок по блоку. | |
|  | Из памяти | Блок предназначенный для соединения одной части схемы с другой, находящейся в произвольном месте проекта. Для выбора источника, заданного при помощи блока «В память» необходимо произвести двойной щелчок по блоку и выбрать в списке нужную точку соединения. | |
|  | Двунаправленная шина (вход) | Блок предназначенный для создания двунаправленной линии связи. Данный блок в сочетании с блоком типа «Двунаправленная шина (выход)» используется для создание ненаправленных абстрактных соединений.  Т.е. у блока типа «Двунаправленная шина (вход)» с левой стороны можно создать набор именованных портов (входа или выхода). Аналогичные порты, но с противоположным направлением можно создать для блока «Двунаправленная шина (выход)». При этом порты «MAIN» данных блоков будут соединены между собой линией связи, то это означает, что порты с соотвествующим именем у одного блока (например вход A на блоке «Двунаправленная шина (вход)» и выход A у блока «Двунаправленная шина (выход)») будут соединены напрямую топологически.    то есть в данном случае схема, изображённая на рисунке:    Будет эквивалентна такой схеме:    Использование данных блоков в сочетании с субмоделями позволяет создават модели физических систем, описываемых в терминах узловых величин (например модели электрических сетей). | |
|  | Двунаправленная шина (выход) | Блок предназначенный для создания двунаправленной линии связи. Описание использования – см. в блоке «Двунаправленная шина (вход)» | |
|  | Комментарий | Блок, в котором можно вставить произвольный комментарий в формате RTF. Для редактирования комментария необходимо произвести двойной щелчок по блоку. | |
|  | Заметка | Произвольный текст, который можно вставить на схему. | |
|  | Уровень не подсоединённых портов | Данный блок позволяет подключить все неподключенные порты блоков к единому источнику. Для этого необходимо вставить на схему данный блок и соединить его с выходом источника. | |
|  | Блок записи свойств | Блок позволяющий задать свойство блока из заданного источника. Для этого надо вставить данный блок в другой и указать в нём имя присваиваемого свойства блока.  Параметры:  Имя свойства – имя свойства блока, в который вставлен данный блок, которое должно быть присвоено из источника. | |
|  | Условное выполнение субмодели | Если внутри субмолели установлен данный блок, то все блоки внутри неё выполняются только если на входе данного блока логическая 1 (>0.5). | 1 |
|  | Указатель сортировки блоков | Данный блок выполняет вспомогательную функцию. Он обозначает, что функциональный блок, подключенный к выходу данного блока сортируется строго после блоков, которые подключены к его входам. | |
|  | Внешняя DLL | Блок является инструментом для подключения к расчетной схеме ранее скомпилированной библиотеки. При этом достигается существенный прирост скорости расчета, а также возможность по одному и тому же алгоритму обсчитывать несколько типовых объектов. Кроме этого, блок позволяет вести расчет в несколько потоков.  Свойства блока:   * Количество портов — количество входных сигналов в расчетную схему (в DLL); * Массив размерностей выходов — массив, в котором указываются размерности выходов из DLL; * Имена загружаемых DLL — имена предварительно скомпилированных DLL, которые загружаются в схему. Указывается одна DLL или через точку с запятой несколько DLL; * Имена оборудования — имена оборудования (компонентов), которые подставляются в входы/выходы схемы DLL. Одна DLL позволяет обсчитывать много объектов. Имена указываются через точку с запятой; * Имена файлов проектов для отладки — имена файлов проектов, соответствующих загружаемым DLL, используются для отладки (в процессе расчета) схем, заложенных в DLL; * Создавать по умолчанию не существующие переменные — создавать переменные в базе данных, которых не существует, для раcсчитываемого оборудования (компонентов); * Количество потоков — количество потоков, в которых будут обсчитываться объекты (компоненты) по данной DLL. | |
| Закладка «Данные» главной панели инструментов SimInTech | | | |
|  | Чтение из списка сигналов | Производит считывание данных из локального списка сигналов проекта (или из списка сигналов подключенной БД). На выходе блока возвращается вектор, размерность которого равна сумме размерностей сигналов считываемых блоком.  Параметры:   * Имена сигналов – список имён считываемых блоком сигналов через точку с запятой. В списке имён допустимо использовать форматирование имён и создание запросов к БД; * Рассчитывать с задержкой на шаг; | 2\*datasize\*<количество сигналов> - для режима упаковки вектора  datasize для типов данных:  Двоичный:1  Целый:4  Вещественный:8 |
|  | Запись в список сигналов | Производит запись данных в локальный список сигналов проекта (или список сигналов подключенной БД).  Параметры:   * Имена сигналов – список имён считываемых блоком сигналов через точку с запятой. В списке имён допустимо использовать форматирование имён и создание запросов к БД; | datasize\*<количество сигналов> - для режима упаковки вектора  datasize для типов данных:  Двоичный:1  Целый:4  Вещественный:8 |
|  | Выход алгоритма | Производит запись данных в локальный список сигналов проекта (или список сигналов подключенной БД). При двойном щелчке по блоку появляется редактор списка записываемых переменных. Имя переменной задаётся в поле «Имя сигнала». Каждому входу блока соответствует свой сигнал. | datasize\*<количество сигналов> - для режима упаковки вектора  datasize для типов данных:  Двоичный:1  Целый:4  Вещественный:8 |
|  | Чтение сигналов | Производит считывание данных из локального списка сигналов проекта (или из списка сигналов подключенной БД). При двойном щелчке по блоку появляется редактор списка считываемых переменных. Имя переменной задаётся в поле «Имя сигнала». Каждому входу блока соответствует свой сигнал. | 2\*datasize\*<количество сигналов> - для режима упаковки вектора  datasize для типов данных:  Двоичный:1  Целый:4  Вещественный:8 |
|  | Выход алгоритма (векторный) | Производит запись данных в локальный список сигналов проекта (или список сигналов подключенной БД). При двойном щелчке по блоку появляется редактор списка записываемых переменных. Имя переменной задаётся в поле «Имя сигнала». Размерность входного вектора равна сумме размерностей записываемых сигналов. | datasize\*<количество сигналов> - для режима упаковки вектора  datasize для типов данных:  Двоичный:1  Целый:4  Вещественный:8 |
|  | Чтение сигналов (векторный) | Производит считывание данных из локального списка сигналов проекта (или из списка сигналов подключенной БД). При двойном щелчке по блоку появляется редактор списка считываемых переменных. Имя переменной задаётся в поле «Имя сигнала». На выходе блока создаётся вектор, размерность которого равна сумме размерностей считываемых сигналов. | datasize\*<количество сигналов> - для режима упаковки вектора  datasize для типов данных:  Двоичный:1  Целый:4  Вещественный:8 |
|  | Временной график | Блок, реализующий график величины в зависимости от модельного времени задачи. Блок служит для вывода информации в оболочке, код по нему не генерируется. | 0 |
|  | Фазовый портрет | Блок, реализующий вывод фазовой диаграммы, т.е. для каждой точки значение координаты X берётся с первого входа, а Y – со второго входа. Блок служит для вывода информации в оболочке, код по нему не генерируется. | 0 |
|  | График Y от X | Блок реализующий вывод зависимости вектора Y(второй вход) от вектора X (первый вход). Блок служит для вывода информации в оболочке, код по нему не генерируется. | 0 |
|  | Входной контакт | Реализует считывание переменной и передачу её на выход блока. В режиме локальной отладки эквивалентен константе. При удалённой отладке блок считывает переменную из исполнительной системы.  Свойства блока:   * Значение по умолчанию - значение на выходе блока, может быть векторным. * Тип контакта - тип данных выходной переменной при генерации кода для полномасштабной модели. * Имя контакта - имя переменной, считываемой блоком. * Формат имени переменной - правило формирования имени считываемой переменной. * Декларировать в базе - флаг, указывающий необходимость создания данной переменную в базе сигналов системы. * Запрос на имя из базы - запрос из базы данных для формирования имени считываемой переменной. * Функциональный план - имя файла диаграммы, на который необходимо перейти при двойном щелчке по объекту. * Отлаживаемый компонент — имя компонента, который будет установлен при переходе на другую диаграмму в режиме отладки. |  |
|  | Выходной контакт | Реализует запись переменной исполнительной системы со входа блока.  Свойства блока:   * Имя контакта - имя переменной, в которую блок записывает сигнал. * Формат имени переменной - правило формирования имени считываемой переменной. * Декларировать в базе - флаг, указывающий необходимо ли создать данную переменную в базе сигналов системы. * Запрос на имя из базы - запрос из базы данных для формирования имени считываемой переменной. * Функциональный план - имя файла диаграммы, на который необходимо перейти при двойном щелчке по объекту. * Отлаживаемый компонент - имя компонента, который будет установлен при переходе на другую диаграмму в режиме отладки. * Транслировать в s3 - флаг, указывающий надо ли передавать переменную с диаграммы в исполнительную систему в режиме удалённой отладки. |  |
|  | Датчик | Блок аналогичен блоку [Входной контакт](file:///D:\%D0%97%D0%B0%D0%B4%D0%B0%D0%BD%D0%B8%D0%B5%20%D0%9F%D0%95%D1%82%D1%83%D1%85%D0%BE%D0%B2%20%D0%BE%D1%82%2009.03.2017\webhelp\biblioteki_blokov\bloki_vse\3001.html), с добавлением отображения измеряемой величины. Реализует считывание переменной, передачу её на выход блока, а также отображение значения переменной на схемное окно в заданном пользователем формате. В режиме локальной отладки эквивалентен константе. При удалённой отладке блок считывает переменную из исполнительной системы.  Свойства блока:   * Значение по умолчанию – значение на выходе блока, может быть векторным. * Тип контакта – тип данных выходной переменной при генерации кода для полномасштабной модели. * Имя контакта – имя переменной, считываемой блоком. * Формат имени переменной – правило формирования имени считываемой переменной. * Добавить в базу сигналов – флаг, указывающий необходимо ли создать данную переменную в базе сигналов системы. * Запрос на имя из базы – запрос из базы данных для формирования имени считываемой переменной. * Функциональный план – имя файла диаграммы, на который необходимо перейти при двойном щелчке по объекту. * Отлаживаемый компонент – имя компонента, который будет установлен при переходе на другую диаграмму в режиме отладки. |  |
|  | Выход алгоритма (векторный) | Блок производит запись значений сигналов (запись данных) в локальный список сигналов проекта или в список сигналов подключенной базы данных. Блок имеет один вход, в отличие от блока «Выход алгоритма», где количество входных портов равно количеству записываемых сигналов.  Свойства блока:   * Выбор максимума (ИЛИ). * Имя алгоритма. * Транслировать в исполнительную систему. * Транслировать из исполнительной системы. * Перезаписать сигнал после рестарта. |  |
|  | Чтение сигналов (векторный) | Блок производит считывание данных из локального списка сигналов проекта (или из списка сигналов подключенной базы данных). На выходе блока создаётся вектор, размерность которого равна сумме размерностей считываемых сигналов.  Свойства блока:   * Автосброс. * Рассчитывать с задержкой на шаг. * Транслировать в исполнительную систему. * Использовать сам сигнал как значение выхода при отладке |  |
|  | Клиент ОРС | Блок производит считывание и запись сигналов в другие программы с использованием интерфейса обмена данными по протоколу OPC. Размерность входа равна сумме размерностей сигналов, заданных в соответствующем параметре блока. Размерность выхода равна сумме размерностей сигналов, заданных в соответствующем параметре блока.  Свойства блока:   * Шаг обмена данными, сек. * Имя OPC-сервера – имя COM-сервера, с которым производится обмен данными. * Задержка при старте, мсек – задержка в мсек при подсоединении к COM-серверу. * Имена считываемых параметров – список имён сигналов в OPC-сервере, которые блок читает (разделитель ; или перевод каретки). * Имена записываемых параметров – список имён сигналов в OPC-сервере, которые блок записывает (разделитель ; или перевод каретки). * Размерности считываемых параметров. * Размерности записываемых параметров. |  |
| Закладка «Ключи» главной панели инструментов SimInTech | | | |
|  | Ключ ручной перекидной | Блок формирует на выходе сигнал, равный входному сигналу по первому ИЛИ второму входу, в зависимости от своего «состояния»:  Состояние ключа управляется (изменяется) однократным нажатием левой кнопкой мыши по блоку в процессе расчета. Если схема находится в режиме «Индикация», то при наведении на блок курсора мыши блок обрамляется синей рамкой, которая указывает на «интерактивность» блока.  Блок не имеет настраиваемых свойств. |  |
|  | Ключ-0 | Блок векторизован и реализует функцию управляемого ключа по алгоритму:  y ( t ) = Y0, если x ( t ) < К;  y ( t ) = x ( t ), если x ( t ) >= К,  где x(t) - вектор входных сигналов;  y(t) - вектор выходных сигналов;  К - вектор "порогов" (параметров уставок);  Y0 - вектор выходных сигналов блока при входных сигналах, меньших пороговых значений. | 24\*<размерность входа> |
|  | Ключ-1 | Блок векторизован и реализует функцию управляемого ключа по алгоритму:  y1 ( t ) = x ( t ), если x ( t ) >= К;  y1 ( t ) = Y0, если x ( t ) < К;  y2 ( t ) = x ( t ), если x ( t ) < К;  y2 ( t ) = Y0, если x ( t ) >= К,  где x(t) - вектор входных сигналов;  y1(t) - вектор сигналов на 1-ом выходном порте;  y2(t) - вектор сигналов на 2-ом выходном порте;  К - вектор пороговых значений (уставок), определяющих условия перекоммутации сигналов на выходных портах блока;  Y0 - вектор сигналов на 1-ом выходном порте блока, при входных сигналах, меньших пороговых значений К или вектор сигналов на 2-ом выходном порте блока, при входных сигналах, больших пороговых значений К. | 32\*<размерность входа> |
|  | Ключ-2 | Блок векторизован и реализует функцию управляемого ключа по алгоритму:  y1 ( t ) = x ( t ), если x ( t ) <= К1;  y1 ( t ) = Y10, если x ( t ) > К1;  y2 ( t ) = x ( t ), если x ( t ) > К2;  y2 ( t ) = Y20, если x ( t ) <= К2,  где x(t) - вектор входных сигналов;  y1(t) - вектор сигналов на 1-ом выходном порте;  y2(t) - вектор сигналов на 2-ом выходном порте;  К1, K2 - векторы пороговых значений (уставок), определяющих условия перекоммутации сигналов на выходных портах блока;  Y10 - вектор сигналов на 1-ом выходном порте блока, при входных сигналах, больших пороговых значений К1;  Y20 - вектор сигналов на 2-ом выходном порте блока, при входных сигналах, меньших пороговых значений К2. | 32\*<размерность входа> |
|  | Ключ-3 | Блок векторизован и реализует функцию управляемого ключа по алгоритму:  y ( t ) = x1 ( t ), если x2 ( t ) < К;  y ( t ) = x3 ( t ), если x2 ( t ) >= К,  где x1(t) - вектор входных сигналов на 1-ом входном порте;  x2(t) - вектор управляющих сигналов на 2-ом входном порте;  x3(t) - вектор входных сигналов на 3-ом входном порте;  y(t) - вектор сигналов на выходном порте;  К - вектор пороговых значений управляющих сигналов (обычно логических уставок), определяющих условия перекоммутации входных сигналов на выход блока; | 16\*<размерность входа> |
|  | Ключ интегратора | Блок реализует функцию управляемого ключа для интегрирующего привода (типа интегратора или инерционно-интегрирующего звена) по следующему алгоритму:  y ( t ) = x1 ( t ), если a1 < x2 ( t ) < a2;  y ( t ) = 0, если x2 ( t ) <= a1 и x1 ( t ) <= 0,  или x2 ( t ) >= a2 и x1 ( t ) >= 0;  y ( t ) = x1 ( t ), если x2 ( t ) <= a1 и x1 ( t ) > 0,  или x2 ( t ) >= a2 и x1 ( t ) < 0, | 24\*<размерность входа> |
|  | Ключ управляемый нормально разомкнутый тип 1 | Блок векторизован и реализует функцию управляемого ключа по алгоритму:  где *x*(*t*) - входной сигнал блока (или вектор сигналов), *y*(*t*) - выходной сигнал блока (или вектор сигналов). Управляющий сигнал входит в блок сверху. В процессе расчета изображение блока меняется, в зависимости от наличия управляющего сигнала: в нормальном положении ключ разомкнут, при подаче управляющего сигнала ключ замкнут:    Блок не имеет настраиваемых свойств. |  |
|  | Ключ управляемый нормально разомкнутый тип 2 | Блок векторизован и реализует функцию управляемого ключа по алгоритму:  где *x*(*t*) - входной сигнал блока (или вектор сигналов), *y*(*t*) - выходной сигнал блока (или вектор сигналов). Управляющий сигнал входит в блок сверху. В процессе расчета изображение блока меняется, в зависимости от наличия управляющего сигнала: в нормальном положении ключ разомкнут, при подаче управляющего сигнала ключ замкнут:    Блок не имеет настраиваемых свойств. |  |
|  | Блок «Ключ управляемый нормально замкнутый тип 1 | Блок реализует функцию управляемого ключа по алгоритму:  где *x*(*t*) — входной сигнал блока (или вектор сигналов), *y*(*t*) — выходной сигнал блока (или вектор сигналов). Управляющий сигнал входит в блок сверху. В процессе расчета изображение блока меняется, в зависимости от наличия управляющего сигнала: в нормальном положении ключ замкнут, при подаче управляющего сигнала ключ разомкнут:    Блок не имеет настраиваемых свойств. |  |
|  | Ключ управляемый нормально замкнутый тип 2 | Блок реализует функцию управляемого ключа по алгоритму:  где *x*(*t*) — входной сигнал блока (или вектор сигналов), *y*(*t*) — выходной сигнал блока (или вектор сигналов). Управляющий сигнал входит в блок сверху. В процессе расчета изображение блока меняется, в зависимости от наличия управляющего сигнала: в нормальном положении ключ замкнут, при подаче управляющего сигнала ключ разомкнут:    Блок не имеет настраиваемых свойств. |  |
|  | Ключ управляемый нормально разомкнутый тип 3 | Блок реализует функцию управляемого ключа по алгоритму:  где *x*1(*t*), *x*2(*t*) — входные сигналы блока (или векторы сигналов), *y*1(*t*), *y*2(*t*) — выходные сигналы блока (или векторы сигналов). Управляющий сигнал входит в блок сверху. В процессе расчета изображение блока меняется, в зависимости от наличия управляющего сигнала: в нормальном положении ключ разомкнут, при подаче управляющего сигнала ключ замкнут:    Блок не имеет настраиваемых свойств. |  |
|  | Ключ управляемый нормально разомкнутый тип 4 | Блок реализует функцию управляемого ключа по алгоритму:  где x1(t), x2(t) — входные сигналы блока (или векторы сигналов), y1(t), y2(t) — выходные сигналы блока (или векторы сигналов). Управляющий сигнал входит в блок сверху. В процессе расчета изображение блока меняется, в зависимости от наличия управляющего сигнала: в нормальном положении ключ разомкнут, при подаче управляющего сигнала ключ замкнут:    Блок не имеет настраиваемых свойств. |  |
|  | Ключ управляемый нормально замкнутый тип 3 | Блок реализует функцию управляемого ключа по алгоритму:  где *x*1(*t*), *x*2(*t*) — входные сигналы блока (или векторы сигналов), *y*1(*t*), *y*2(*t*) — выходные сигналы блока (или векторы сигналов). Управляющий сигнал входит в блок сверху. В процессе расчета изображение блока меняется, в зависимости от наличия управляющего сигнала: в нормальном положении ключ замкнут, при подаче управляющего сигнала ключ разомкнут:    Блок не имеет настраиваемых свойств. |  |
|  | Ключ управляемый нормально замкнутый тип 4 | Блок реализует функцию управляемого ключа по алгоритму:  где *x*1(*t*), *x*2(*t*) — входные сигналы блока (или векторы сигналов), *y*1(*t*), *y*2(*t*) — выходные сигналы блока (или векторы сигналов). Управляющий сигнал входит в блок сверху. В процессе расчета изображение блока меняется, в зависимости от наличия управляющего сигнала: в нормальном положении ключ замкнут, при подаче управляющего сигнала ключ разомкнут:    Блок не имеет настраиваемых свойств. |  |
|  | Ключ управляемый перекидной по входам тип 1 | Блок векторизован и реализует функцию управляемого ключа по алгоритму:  где *x*1(*t*), *x*2(*t*) - входные сигналы блока (или векторы сигналов), *y*(*t*) - выходной сигнал блока (или вектор сигналов). Управляющий сигнал входит в блок сверху. В процессе расчета изображение блока меняется, в зависимости от наличия управляющего сигнала: в нормальном положении ключ перекинут на первый вход, при подаче управляющего сигнала ключ перекидывается на второй вход:    Блок не имеет настраиваемых свойств. |  |
|  | Коммутатор входных сигналов | Блок представляет собой коммутатор, подключающий к выходу один из аналоговых входных сигналов (X1 или X2). Управление коммутацией происходит с помощью двух дискретных входов (X3 и X4). Логика работы блока для двух входных сигналов, представлена в таблице.    Блок не имеет настраиваемых свойств. |  |
|  | Выделение минимума с формированием признака | Блок пропускает на аналоговый выход минимальный из двух аналоговых входных сигналов либо повторяет значение сигнала, если оно одинаково для обоих входов. Дискретный выход признака становится равен 1, если минимальным является значение на втором входе. В противном случае признак равен 0.  Блок не имеет настраиваемых свойств. |  |
|  | Выделение максимума с формированием признака | Блок пропускает на аналоговый выход максимальный из двух аналоговых входных сигналов либо повторяет значение сигнала, если оно одинаково для обоих входов. Дискретный выход признака становится равен 1, если максимальным является значение на втором входе. В противном случае признак равен 0.  Блок не имеет настраиваемых свойств. |  |
|  | Задатчик с варьируемым значением | Блок передает на аналоговый выход значение первого входа (x1), если разрешена трансляция (если вход x2 имеет значение 1) и если значение сигнала x1 входит в диапазон между максимальным и минимальным значением, задаваемым соответственно сигналами на входах x3 и x4. Если величина сигнала не входит в диапазон x4÷x3, то сигнал на выходе приобретает соответственно значение либо верхнего предела (x3), либо нижнего (x4). В случае, если трансляция запрещена (x2=0), то на выходе блока фиксируется то значение, которое было на момент запрещения трансляции.  Свойства блока:   * Название. * Начальное значение. |  |
| Закладка «Логические» главной панели инструментов SimInTech | | | |
|  | Оператор И | Реализует операцию логического И.  Параметры:   * Количество портов | 1\*<размерность входа> |
|  | Оператор ИЛИ | Реализует операцию логического ИЛИ.  Параметры:   * Количество портов | 1\*<размерность входа> |
|  | Оператор НЕ | Реализует операцию логического НЕ. | 1\*<размерность входа> |
|  | Логические операции | Реализует произвольную логическую операцию.  Параметры:   * Тип второго операнда – указывает тип второго операнда. Если вектор – то делается поэлементная операция. Скаляр – производиться операция с каждым элементов первого входа и значением второго входа. * Тип операнда – тип логической операции. | 1\*<размерность входа> |
|  | Операция БОЛЬШЕ | Возвращает логическую единицу, если первый вход больше второго.  Параметры:   * Тип второго операнда – указывает тип второго операнда. Если вектор – то делается поэлементная операция. Скаляр – производиться операция с каждым элементов первого входа и значением второго входа. | 1\*<размерность входа> |
|  | Операция МЕНЬШЕ | Возвращает логическую единицу, если первый вход меньше второго.  Параметры:   * Тип второго операнда – указывает тип второго операнда. Если вектор – то делается поэлементная операция. Скаляр – производиться операция с каждым элементов первого входа и значением второго входа. | 1\*<размерность входа> |
|  | Операция РАВНО | Возвращает логическую единицу, если первый вход равен второму.  Параметры:   * Тип второго операнда – указывает тип второго операнда. Если вектор – то делается поэлементная операция. Скаляр – производиться операция с каждым элементов первого входа и значением второго входа. | 1\*<размерность входа> |
|  | Операция НЕ РАВНО | Возвращает логическую единицу, если первый вход не равен второму.  Параметры:   * Тип второго операнда – указывает тип второго операнда. Если вектор – то делается поэлементная операция. Скаляр – производиться операция с каждым элементов первого входа и значением второго входа. | 1\*<размерность входа> |
|  | БОЛЬШЕ ИЛИ РАВНО | Возвращает логическую единицу если первый вход больше или равен второму.  Параметры:   * Тип второго операнда – указывает тип второго операнда. Если вектор – то делается поэлементная операция. Скаляр – производиться операция с каждым элементов первого входа и значением второго входа. | 1\*<размерность входа> |
|  | МЕНЬШЕ ИЛИ РАВНО | Возвращает логическую единицу если первый вход меньше или равен второму.  Параметры:   * Тип второго операнда – указывает тип второго операнда. Если вектор – то делается поэлементная операция. Скаляр – производиться операция с каждым элементов первого входа и значением второго входа. | 1\*<размерность входа> |
|  | XOR | Производит операцию поэлементного XOR (ИСКЛЮЧАЮЩЕЕ ИЛИ) для входного вектора. | 1\*<размерность входа> |
|  | NOT XOR | Производит операцию поэлементного NOT XOR (ИСКЛЮЧАЮЩЕЕ НЕ-ИЛИ) для входного вектора. | 1\*<размерность входа> |
|  | M из N | Возвращает логическую единицу, если количество входов, на которых значение равно логической единице больше заданного количества. При этом операция производится над всеми элементами всех входов, а выход блока – скалярный.  Параметры:   * Количество портов * Число входных True для срабатывания | 4+<размерность входа> |
|  | M из N поэлементное | Возвращает логические единицы в векторе выхода, если количество входов, на которых значение равно логической единице больше заданного количества. При этом операция производится поэлементно для каждого из входов и выхода.  Параметры:   * Количество портов * Число входных True для срабатывания | 4+<размерность входа> |
|  | Один из многих | Блок реализует специальный алгоритм формирования логических сигналов в векторе выхода в зависимости от изменения типа логического сигнала в векторе входа.  Алгоритм работы блока: До начала процесса моделирования в диалоговом окне блока только один сигнал (например, N-ый) в векторе выхода может быть задан как сигнал типа TRUE (активный), независимо от типа соответствующего (N-го) сигнала в векторе входа. Все остальные выходные сигналы считаются сигналами типа FALSE (неактивный), независимо от типа соответствующих сигналов в векторе входа. Если в процессе моделирования какой-то логический сигнал (любой, кроме N-го) в векторе входа изменил свой тип с FALSE на TRUE, то соответствующий сигнал в векторе выхода изменяет свой тип на TRUE (активный) и одновременно N-ый сигнал в векторе выхода становится типа FALSE (неактивный).  Свойства блока:   * Начальный номер активного элемента входного вектора - номер активного элемента в начальный момент. |  |
|  | Один из многих по выбору | Блок реализует специальный алгоритм формирования вектора выходных логических сигналов в зависимости от значения входного скалярного сигнала, округленного до целого.  Алгоритм работы блока:  Если в процессе моделирования скалярный входной сигнал при округлению до целого равен М (1 =< M =< N, где N – размерность вектора выхода), то в векторе выхода тип TRUE (активный) имеет только М-ый сигнал, а все остальные сигналы в векторе выхода соответствуют типу FALSE (неактивный).  Если значение входного скалярного сигнала, округленное до целого, меньше 1 или больше N, то все сигналы в векторе выхода соответствуют типу FALSE. | 1\*<размерность входа> |
|  | Счётчик | Производит подсчёт входных импульсов на входе. Данный блок производит инкремент элемента выходного вектора, если элемент входного вектора удовлетворяет условию перехода в заданную область определения (т.е. выход инкрементируется при переходе). Счётчик не сбрасывается. Выходное значение – вектор целых чисел с размерностью равной размерности входа. Начальное значение выхода счётчика равно 0. Если вход удовлетворяет условию то на первом шаге выходного значение будет инкрементировано. При необходимости организации сложных счётчиков с управляемым состоянием рекоментуется использовать модель счётчика из сумматора и задержки на шаг.  Параметры:   * Нижний предел нечувствительности – нижняя граница области определения * Верхний предел нечувствительности – верхняя граница области определения * Счётчик если вход (Внутри диапазона, Вне диапазона) – тип области определения. | 21\*<размерность входа> |
|  | Векторное И | Блок является аналогом логического блока «И», векторизован и работает по следующему алгоритму:  где *y*(*t*) – выходной скалярный сигнал блока, *xi,j*(*t*) – i-ый элемент j-ого входного векторного сигнала.  Свойства блока:   * Количество портов - количество входных сигналов блока. |  |
|  | Векторное ИЛИ | Блок является аналогом логического блока «ИЛИ», векторизован и работает по следующему алгоритму:  где *y*(*t*) – выходной скалярный сигнал блока, *xi,j*(*t*) – i-ый элемент j-ого входного векторного сигнала.  Свойства блока:  Количество портов - количество входных сигналов блока. |  |
|  | Логическое И с пассивным входом | Блок является аналогом логического блока «И», с выделением активного (первого) и т. н. «пассивного» (второго) порта. Результат работы блока зависит от последовательности во времени моментов появления логических единиц на входах, а именно:   * если первой появилась единица на первом (активном) входе, а единица на втором (пассивном) входе появилась после первой, то выход блока становится равен 1; * если первой появилась единица на втором (пассивном) входе, а единица на первом входе «пришла» после второй, то выход блока не устанавливается в 1.   Блок не имеет настраиваемых свойств. |  |
| Закладка «Триггеры» главной панели инструментов SimInTech | | | |
|  | RS-триггер с приоритетом по сбросу | Реализует RS-триггер с приоритетом по сбросу, т.е. если уровень логической единицы присутствует на входах R и S, то выход принимает значение логического нуля.  Параметры:   * Начальные условия – начальные значение выхода. * Добавить инверсный выход – указывает наличие инверсного выхода. | 1\*<размерность входа> |
|  | RS-триггер с приоритетом по установке | Реализует RS-триггер с приоритетом по сбросу, т.е. если уровень логической единицы присутствует на входах R и S, то выход принимает значение логической единицы.  Параметры:   * Начальные условия – начальные значение выхода. * Добавить инверсный выход – указывает наличие инверсного выхода. | 1\*<размерность входа> |
|  | Триггер T | Реализует счётный триггер, изменяющий своё состояние при изменении значение счётного входа.  Параметры:   * Начальные условия – начальные значение выхода. * Добавить инверсный выход – указывает наличие инверсного выхода. * Тип счётного входа (по фронту, по спаду) – условие срабатывания счётного входа. | 2\*<размерность входа> |
|  | Триггер TR | Реализует RS-триггер с приоритетом по сбросу, т.е. если уровень логической единицы присутствует на входах R и S, то выход принимает значение логического нуля, и счётным входом.  Параметры:   * Начальные условия – начальные значение выхода. * Добавить инверсный выход – указывает наличие инверсного выхода. * Тип счётного входа (по фронту, по спаду) – условие срабатывания счётного входа. | 2\*<размерность входа> |
|  | Триггер TS | Реализует RS-триггер с приоритетом по сбросу, т.е. если уровень логической единицы присутствует на входах R и S, то выход принимает значение логической единицы, и счётным входом.  Параметры:   * Начальные условия – начальные значение выхода. * Добавить инверсный выход – указывает наличие инверсного выхода. * Тип счётного входа (по фронту, по спаду) – условие срабатывания счётного входа. | 2\*<размерность входа> |
|  | Определение первого события | Реализует RD-триггер (приоритет по сбросу), определяющий первое событие (появление фронта) по входным портам с данными (D-порты, то есть 2-ой, 3-ий и последующие порты при их наличии). Блок дискретный. Является вариантом реализации D-триггера. Запоминает какое из событий (логическая единица), поданное на D-входы, возникло первым. После возникновения события, на соответствующем выходе формируется логическая единица и состояния всех выходов фиксируется до возникновения сигнала сброса R, установка которого сбрасывает все выходы в 0, а снятие возвращает блок в работу. Если одновременно возникло более одного события, то состояние выходов блока не меняется до тех пор, пока на входы не будет действовать только один сигнал, который будет считаться за первый. Количество входных сигнальных портов настраивается в свойствах блока. Ему соответствует такое же количество выходов. Также всегда присутствует управляющий порт сброса R.  Свойства блока:   * Количество сигнальных портов - задатчик количества входных портов с данными; * Не реагировать при нескольких событиях сразу - включение игнорирования одновременного появления нескольких событий. |  |
| Закладка «Задержки и импульсы» главной панели инструментов SimInTech | | | |
|  | Задержка по включению | Реализует задержку входного сигнала на заданное время при изменении входного сигнала от логического нуля до единицы.  Параметры:   * Массив времён подтверждения – время задержки включения. * Способ задания времени подтверждения – если указан способ задания «Вход», то время задержки берётся со второго входа. | 18\*<размерность входа> |
|  | Задержка по включению и выключению | Реализует задержку входного сигнала на заданное время при любом изменении входного сигнала.  Параметры:   * Массив времён подтверждения – время задержки включения. * Способ задания времени подтверждения – если указан способ задания «Вход», то время задержки берётся со второго входа. | 35\*<размерность входа> |
|  | Задержка по выключению | Реализует задержку входного сигнала на заданное время при изменении входного сигнала от логической единицы до нуля.  Параметры:   * Массив времён подтверждения – время задержки включения. * Способ задания времени подтверждения – если указан способ задания «Вход», то время задержки берётся со второго входа. | 18\*<размерность входа> |
|  | Импульс по фронту | Формирует импульс величиной 1, длительностью в один шаг интегрирования, при изменении входной величины от нуля до единицы. | 2\*<размерность входа> |
|  | Импульс по срезу | Формирует импульс величиной 1, длительностью в один шаг интегрирования, при изменении входной величины от единицы до нуля. | 2\*<размерность входа> |
|  | Импульс по фронту или срезу | Формирует импульс величиной 1, длительностью в один шаг интегрирования, при изменении входной величины. | 2\*<размерность входа> |
|  | Импульс | Формирует импульс заданной длительности при изменении входа от нуля до единицы.  Алгоритм работы блока: если входной сигнал изменился с нулевого значения на единичное (или на ненулевое), то блок формирует на выходе логическую единицу в течение заданного времени. При этом, в течение всего импульса (пока на выходе 1), блок **не учитывает** изменения входного сигнала.  **Параметры:**   * Длительность импульса — время, в течение которого блок формирует логическую единицу на выходе. * Длительность задается через – способ задания длительности импульса: через дополнительной вход или через параметр блока «Длительность импульса».   **Сравнительный пример работы блока:**  Рассмотрим три блока с длительностью импульсов 1, 4 и 10 секунд. При этом на вход в блоки будем подавать один и тот же сигнал типа «меандр» с полным периодом 7 секунд (длительность первого полупериода 2 с, второго 5 с). Время расчета — 30 с:  D:\DCAD\help\html_help\images\3034_example.png  Рассмотрим графики получаемых импульсов на выходе из блоков (розовый график — входной сигнал типа «меандр», черный график — импульс):  D:\DCAD\help\html_help\images\3034_example_1.png  D:\DCAD\help\html_help\images\3034_example_2.png  D:\DCAD\help\html_help\images\3034_example_3.png   * Как видно из графиков, блок генерирует импульсы строго заданной длительности и только в тот момент, когда входной сигнал изменяется от 0 до 1. При этом, если импульс уже генерируется, то блок не принимает в расчёт изменения входного сигнала (см. график с импульсом 7 с). | 18\*<размерность входа> |
|  | Импульс длительностью не более заданной | Формирует импульс заданной длительности и не более при изменении входа от нуля до единицы.  Алгоритм работы блока: если входной сигнал изменился с нулевого значения на единичное (или на ненулевое), то блок формирует на выходе логическую единицу в течение заданного времени. При этом, если входной сигнал исчез ранее заданной длительности импульса (стал равным 0), блок **учитывает** это и прекращает подачу импульса на выходе.  **Параметры:**   * Длительность импульса — время, в течение которого блок формирует логическую единицу на выходе. Длительность импульса может быть меньше заданной, при исчезновении входного сигнала. * Длительность задается через – способ задания длительности импульса: через дополнительный вход или через параметр блока «Длительность импульса».   **Сравнительный пример работы блока:**  Рассмотрим три блока с длительностью импульсов не более 1, 4 и 10 секунд. При этом на вход в блоки будем подавать один и тот же сигнал типа «меандр» с полным периодом 7 секунд (длительность первого полупериода 2 с, второго 5 с). Время расчета — 30 с:  D:\DCAD\help\html_help\images\3035_example.png  Рассмотрим графики получаемых импульсов на выходе из блоков (розовый график — входной сигнал типа «меандр», черный график — импульс):  D:\DCAD\help\html_help\images\3035_example_1.png  D:\DCAD\help\html_help\images\3035_example_2.png  D:\DCAD\help\html_help\images\3035_example_3.png   * Как видно из графиков, блок генерирует импульсы не более заданной длительности и только до тех пор, пока входной сигнал изменялся от 0 до 1 и остается равным 1. Два последних графика совпадают, хотя заданная длительность импульса в блоках разная. | 18\*<размерность входа> |
|  | Импульс с пролонгированием | Формирует импульс не менее заданной длительности при изменении входа от нуля до единицы. При этом импульс продляется вновь на заданную длительность, если происходит очередное изменение входа от 0 до 1 в процессе формирования предыдущего импульса.  Алгоритм работы блока: если входной сигнал изменился с нулевого значения на единичное (или на ненулевое), то блок формирует на выходе логическую единицу в течение заданного времени. При этом, в течение всего импульса (пока на выходе блока 1), блок также **учитывает** изменения входного сигнала и при очередном изменении от 0 до 1 импульс формируется заново («подхватывается»), а длительность импульса считается от момент последнего изменения входа от 0 до 1.  **Параметры:**   * Длительность импульса — время, в течение которого блок формирует логическую единицу на выходе. * Длительность задается через – способ задания длительности импульса: через дополнительной вход или через параметр блока «Длительность импульса».   **Сравнительный пример работы блока:**  Рассмотрим три блока с длительностью импульсов 1, 4 и 10 секунд, с пролонгированием. При этом на вход в блоки будем подавать один и тот же сигнал типа «меандр» с полным периодом 7 секунд (длительность первого полупериода 2 с, второго 5 с). Время расчета — 30 с:  D:\DCAD\help\html_help\images\3036_example.png  Рассмотрим графики получаемых импульсов на выходе из блоков (розовый график — входной сигнал типа «меандр», черный график — импульс):  D:\DCAD\help\html_help\images\3036_example_1.png  D:\DCAD\help\html_help\images\3036_example_2.png  D:\DCAD\help\html_help\images\3036_example_3.png   * Как видно из графиков, блок генерирует импульсы длительностью не менее заданной. При этом, если импульс уже генерируется, то блок **принимает в расчёт** новые изменения входного сигнала от 0 до 1 (см. график с импульсом 7 с). | 18\*<размерность входа> |
|  | Временное подтверждение | Изменяет значение на выходе при выполнении условия более заданного времени.  Параметры:   * Вектор времён подтверждения. * Подтверждение требуется – условие при котором формируется импульс. * Тип выхода – тип выхода блока. Обычный или инверсный. Блок является устаревшим. | 18\*<размерность входа> |
|  | Одновибратор | Формирует импульс заданной длительности при изменении входной величины до заданного порога.  Параметры:   * Длительность импульса * Порог срабатывания. * Тип выхода – тип выхода блока. Обычный или инверсный. Блок является устаревшим. | 18\*<размерность входа> |
| Закладка «Релейные блоки» главной панели инструментов SimInTech | | |  |
|  | Обмотка реле | Субмодель, реализующая передачу сигнала от входа данного блока к контактам реле.  Параметры:   * Имя реле – имя, по которому контакты ассоциируются с данным блоком. | 1\*<размерность входа> |
|  | Двухпозиционное реле (SET) | Субмодель, реализующая установку состояния логической единицы двухпозиционного реле.  Параметры:   * Имя реле – имя, по которому контакты ассоциируются с данным блоком. | 1\*<размерность входа> |
|  | Двухпозиционное реле (RESET) | Субмодель, реализующая установку состояние логического нуля двухпозиционного реле.  Параметры:   * Имя реле – имя, по которому контакты ассоциируются с данным блоком. | 1\*<размерность входа> |
|  | Замыкающий контакт реле | Реализует операцию логического И для входа блока и значения, поданного на соответствующую управляющую обмотку реле.  Параметры:   * Имя реле – имя, по которому определяется управляющее воздействие. | 1\*<размерность входа> |
|  | Размыкающий контакт реле | Реализует операцию логического И-НЕ для входа блока и значения, поданного на соответствующую управляющую обмотку реле.  Параметры:   * Имя реле – имя, по которому определяется управляющее воздействие. | 1\*<размерность входа> |
|  | Переключающий контакт реле | Реализует операцию переключения входных контактов в зависимости от значения на управляющей обмотке.  Параметры:  Имя реле – имя, по которому определяется управляющее воздействие. | 10\*<размерность входа> |
|  | Замыкающий контакт реле с задержкой по замыканию | Реализует операцию логического И с задержкой по включению для входа блока и значения, поданного на соответствующую управляющую обмотку реле.  Параметры:   * Имя реле – имя, по которому определяется управляющее воздействие. * Время задержки – время задержки при включении контакта; | 19\*<размерность входа> |
|  | Замыкающий контакт реле с задержкой по размыканию | Реализует операцию логического И с задержкой по выключению для входа блока и значения, поданного на соответствующую управляющую обмотку реле.  Параметры:   * Имя реле – имя, по которому определяется управляющее воздействие. * Время задержки – время задержки при выключении контакта; | 19\*<размерность входа> |
|  | Размыкающий контакт реле с задержкой по замыканию | Реализует операцию логического И-НЕ с задержкой по выключению обмотки (замыканию) для входа блока и значения, поданного на соответствующую управляющую обмотку реле.  Параметры:   * Имя реле – имя, по которому определяется управляющее воздействие. * Время задержки – время задержки при замыкании контакта; | 19\*<размерность входа> |
|  | Размыкающий контакт реле с задержкой по размыканию | Реализует операцию логического И-НЕ с задержкой по включению обмотки (размыканию) для входа блока и значения, поданного на соответствующую управляющую обмотку реле.  Параметры:   * Имя реле – имя, по которому определяется управляющее воздействие. * Время задержки – время задержки при размыкании контакта. | 19\*<размерность входа> |
|  | Внешний контакт реле | Блок представляет собой субмодель. Для того чтобы посмотреть структуру алгоритма – нажмите правой кнопкой Действия → Войти в субмодель. Реализует операцию логического И для входа блока и значения, указанного в параметрах блока (аналог управляющей обмотки реле для блока [Замыкающий контакт реле](file:///D:\%D0%97%D0%B0%D0%B4%D0%B0%D0%BD%D0%B8%D0%B5%20%D0%9F%D0%95%D1%82%D1%83%D1%85%D0%BE%D0%B2%20%D0%BE%D1%82%2009.03.2017\webhelp\biblioteki_blokov\bloki_vse\3041.html)). Таким образом, имя контакта указывается непосредственно внутри субмодели.  Свойства блока:   * Имя переменной – имя переменной, для которой (и для входа блока) выполняется операция И. . * Начальное состояние (0 или 1) - начальное состояние выхода блока. |  |
|  | Двухпозиционное реле (RS) | Блок представляет собой субмодель, содержащую RS-триггер с приоритетом по сбросу.  Свойства блока:   * Имя реле - имя переменной на выходе RS-триггера, передаваемое в список сигналов «В память». |  |
|  | Внешний размыкающий контакт реле | Блок представляет собой субмодель. Для того чтобы посмотреть структуру алгоритма – нажмите правой кнопкой Действия → Войти в субмодель. Реализует операцию логического И-НЕ для входа блока и значения, указанного в параметрах блока (аналог управляющей обмотки реле для блока [Размыкающий контакт реле](file:///D:\%D0%97%D0%B0%D0%B4%D0%B0%D0%BD%D0%B8%D0%B5%20%D0%9F%D0%95%D1%82%D1%83%D1%85%D0%BE%D0%B2%20%D0%BE%D1%82%2009.03.2017\webhelp\biblioteki_blokov\bloki_vse\3042.html)). Таким образом, имя контакта указывается непосредственно внутри субмодели.  Свойства блока:   * Имя переменной - имя переменной, для которой (и для входа блока) выполняется операция И-НЕ. * Начальное состояние (0 или 1) - начальное состояние выхода блока. |  |
| Закладка «Дискретные» главной панели инструментов SimInTech | | | |
|  | Экстраполятор | Блок **векторизован** и реализует экстраполяцию нулевого порядка:  **y(t) = u(t[k]),**  где **y(t)** - значение выходного сигнала в текущий момент модельного времени, **u(t[k])** - значение входного сигнала в последний момент дискретизации **t[k].**  **Примечания:**  Для правильной работы данного блока необходимо задать максимальный шаг интегрирования не больше, чем **T/2**. При интегрировании с постоянным шагом рекомендуется задать шаг таким, чтобы период квантования был кратен шагу интегрирования (**T = k\*h**).  Параметры:   * Период квантования – период квантования расчёта блока. | 24\*<размерность входа> |
| Закладка «Функции» главной панели инструментов SimInTech | | | |
|  | Линейная функция | Блок возвращает выходное значение вычисленное по формуле:  Y = a + b\*x,  Где a,b – коэффициенты,  x – входное значение.  Параметры блока:   * Свободный член – коэффициент a * Коэффициент при t – коэффициент b   Параметры могут быть векторами, размерности которых должны быть равны размерности входа, в этом случае на выходе будет также вектор. | 24\*<размерность входа> |
|  | Параболическая функция | Блок реализует преобразование входного сигнала (скалярного или векторного) по формуле:  где *a*0, *a*1, *a*2 – коэффициенты параболы, *x*(*t*), *y*(*t*) – входной и выходной сигналы.  Параметры блока:   * Свободный член, в формуле a0+a1·x+a2·x² – коэффициент a0. * Коэффициент при x – коэффициент a1. * Коэффициент при x² – коэффициент a2.   Свойства могут быть векторами, размерности которых должны быть равны размерности входа, в этом случае на выходе будет также вектор. |  |
|  | Синусоидальная функция | Блок возвращает выходное значение вычисленное по формуле:  Y = a\*sin(w\*x + f),  Где a, w, f – амплитуда, круговая частота и фаза,  x – входное значение.  Параметры блока:   * Амплитуда– коэффициент a * Частота– коэффициент w * Фаза - коэффициент f   Параметры могут быть векторами, размерности которых должны быть равны размерности входа, в этом случае на выходе будет также вектор. | 32\*<размерность входа> |
|  | Экспоненциальная функция | Блок возвращает выходное значение вычисленное по формуле:  Y = a\*exp(b\*x + c),  Где a, b, c – коэффициенты,  x – входное значение.  Параметры блока:   * К-т усиления– коэффициент a * Обратная постоянная времени– коэффициент b * Сдвиг - коэффициент c   Параметры могут быть векторами, размерности которых должны быть равны размерности входа, в этом случае на выходе будет также вектор. | 32\*<размерность входа> |
|  | Гиперболическая функция | Блок возвращает выходное значение вычисленное по формуле:  Y = k/(eps + x),  Где k, eps – коэффициенты,  x – входное значение.  Параметры блока:   * Числитель– коэффициент k * Минимальное значение знаменателя– коэффициент eps.   Параметры могут быть векторами, размерности которых должны быть равны размерности входа, в этом случае на выходе будет также вектор. | 24\*<размерность входа> |
|  | Арксинус | Блок возвращает выходное значение, вычисленное по формуле:  где *a*, *ω*, *f* – амплитуда, круговая частота и фаза, *x*(*t*), *y*(*t*) – входной и выходной сигналы.  Параметры блока:   * Амплитуда – коэффициент a. * Частота – коэффициент ω. * Фаза – коэффициент f.   Свойства могут быть векторами, размерности которых должны быть равны размерности входа, в этом случае на выходе будет также вектор. |  |
|  | Арккосинус | Блок возвращает выходное значение, вычисленное по формуле:  где *a*, *ω*, *f* – амплитуда, круговая частота и фаза, *x*(*t*), *y*(*t*) – входной и выходной сигналы.  Параметры блока:   * Амплитуда – коэффициент a. * Частота – коэффициент ω. * Фаза – коэффициент f.   Свойства могут быть векторами, размерности которых должны быть равны размерности входа, в этом случае на выходе будет также вектор. |  |
|  | Арктангенс | Блок возвращает выходное значение вычисленное по формуле:  Y = a\*arctg(w\*x + f),  Где a, w, f – амплитуда, круговая частота и фаза,  x – входное значение.  Параметры блока:   * Амплитуда– коэффициент a * Частота– коэффициент w * Фаза - коэффициент f   Параметры могут быть векторами, размерности которых должны быть равны размерности входа, в этом случае на выходе будет также вектор. | 32\*<размерность входа> |
|  | Гиперболический синус | Блок возвращает выходное значение, вычисленное по формуле:  где *a*, *ω*, *f* – амплитуда, круговая частота и фаза, *x*(*t*), *y*(*t*) – входной и выходной сигналы.  Параметры блока:   * Амплитуда – коэффициент a. * Частота – коэффициент ω. * Фаза – коэффициент f. |  |
|  | Гиперболический косинус | Блок возвращает выходное значение, вычисленное по формуле:  где *a*, *ω*, *f* – амплитуда, круговая частота и фаза, *x*(*t*), *y*(*t*) – входной и выходной сигналы.  Параметры блока:   * Амплитуда – коэффициент a. * Частота – коэффициент ω. * Фаза – коэффициент f. |  |
|  | Гиперболический тангенс | Блок возвращает выходное значение, вычисленное по формуле:  где *a*, *ω*, *f* – амплитуда, круговая частота и фаза, *x*(*t*), *y*(*t*) – входной и выходной сигналы.  Параметры блока:   * Амплитуда – коэффициент a. * Частота – коэффициент ω. * Фаза – коэффициент f. |  |
|  | Степенная функция | Блок возвращает выходное значение вычисленное по формуле:  Y = xa,  Где a – показатель степени,  x – входное значение.  Параметры блока:   * Показатель степени – коэффициент a   Параметры могут быть векторами, размерности которых должны быть равны размерности входа, в этом случае на выходе будет также вектор. | 16\*<размерность входа> |
|  | Логарифм натуральный | Блок возвращает выходное значение вычисленное по формуле:  Y = a\*ln(w\*x + f),  Где a, w, f – амплитуда, круговая частота и фаза,  x – входное значение.  Параметры блока:   * Амплитуда– коэффициент a * Частота– коэффициент w * Фаза - коэффициент f   Параметры могут быть векторами, размерности которых должны быть равны размерности входа, в этом случае на выходе будет также вектор. | 32\*<размерность входа> |
|  | Логарифм десятичный | Блок возвращает выходное значение вычисленное по формуле:  Y = a\*lg(w\*x + f),  Где a, w, f – амплитуда, круговая частота и фаза,  x – входное значение.  Параметры блока:   * Амплитуда– коэффициент a * Частота– коэффициент w * Фаза - коэффициент f   Параметры могут быть векторами, размерности которых должны быть равны размерности входа, в этом случае на выходе будет также вектор. | 32\*<размерность входа> |
|  | Логарифм натуральный с защитой 0 | Блок векторизован, возвращает выходное значение, вычисленное по алгоритму:  где *a*, *ω*, *f*– амплитуда, круговая частота и фаза, *x*(*t*), *y*(*t*) – входной и выходной сигналы. *b* - константа, определяющая ограничение «снизу». Таким образом, блок работоспособен при любых изменениях входного сигнала, а расчет не останавливается при пересечении входным сигналом нуля.  Параметры блока:   * Амплитуда – коэффициент a. * Частота – коэффициент ω. * Фаза – коэффициент f.   Свойства могут быть векторами, размерности которых должны быть равны размерности входа, в этом случае на выходе будет также вектор. |  |
|  | Логарифм десятичный с защитой 0 | Блок векторизован, возвращает выходное значение, вычисленное по алгоритму:  где a, ω, f– амплитуда, круговая частота и фаза, x(t), y(t) – входной и выходной сигналы. b - константа, определяющая ограничение «снизу». Таким образом, блок работоспособен при любых изменениях входного сигнала, а расчет не останавливается при пересечении входным сигналом нуля.  Параметры блока:   * Амплитуда – коэффициент a. * Частота – коэффициент ω. * Фаза – коэффициент f.   Свойства могут быть векторами, размерности которых должны быть равны размерности входа, в этом случае на выходе будет также вектор. |  |
|  | Корень квадратный | Блок реализует вычисление квадратного корня числа (для каждого элемента входного вектора):    Где x – входное значение.  Размерность выходного вектора равна размерности входного. | 8\*<размерность входа> |
|  | Синус с косинусом | Блок вычисляет значение синуса и косинуса одного аргумента. Блок векторизован, то есть можно получать значения синуса и косинуса нескольких входных значений (вектора).  Блок не имеет настраиваемых параметров. |  |
| Закладка «Статистика» главной панели инструментов SimInTech | | | |
|  | Среднее арифметическое | Блок реализует вычисление среднего арифметического значения входного сигнала. Вычисление среднего значения сигнала производится по формуле:  где *N* – объем выборки (количество значений), *xi* – значения *i*-го элемента выборки.  Свойства блока:   * Размер серии – значение N этого свойства определяет размер серии в выборке. * Способ вывода данных: По всей выборке - среднее значение уточняется сразу же после получения очередного нового значения сигнала на входе в блок. По сериям - среднее значение вычисляется по отдельным сериям. Скользящее - среднее значение вычисляется по сериям, причем на каждом шаге учитываются последние N точек, где N – размер серии. * Период квантования, c – значение этого свойства определяет длительность временного интервала между двумя последовательными считываниями значений сигнала на входе блока. Если это свойство равно 0 (нулю), то считывание производится с периодом, равным шагу интегрирования. |  |
|  | Среднеквадратичное отклонение | Блок реализует вычисление среднеквадратического отклонения значения входного сигнала. Среднеквадратическое отклонение подсчитывается по следующей формуле:  где *N* – объем выборки (количество значений), *xi* – значения *i*-го элемента выборки, *M* – математическое ожидание случайной величины.  Свойства блока:   * Размер серии – значение N этого свойства определяет размер серии в выборке. * Способ вывода данных: По всей выборке - среднеквадратическое отклонение уточняется сразу же после получения очередного нового значения сигнала на входе в блок, По сериям - среднеквадратическое отклонение вычисляется по отдельным сериям, Скользящее - среднеквадратическое отклонение вычисляется по сериям, причем на каждом шаге учитываются последние N точек, где N – размер серии. * Период квантования, c – значение этого свойства определяет длительность временного интервала между двумя последовательными считываниями значений сигнала на входе блока. Если это свойство равно 0 (нулю), то считывание производится с периодом, равным шагу интегрирования. * Удаление линейного тренда (Да или Нет) – если это свойство имеет значение Да, то из массива накопленных значений входного сигнала предварительно вычитается линейный тренд. |  |
| Закладка «Обработка сигналов» главной панели инструментов SimInTech | | | |
|  | Буфер последовательного ввода данных | Блок предназначен для накопления и выдачи в виде вектора, набора дискретных отсчетов входного сигнала.  Свойства блока:   * Размер серии – величина формируемого вектора данных (целое число). |  |
|  | Буфер последовательного вывода данных | Блок предназначен для последовательного вывода содержимого вектора дискретных отсчетов.  Свойства блока:   * Размер серии – величина формируемого вектора данных (целое число). |  |
|  | Быстрое преобразование Фурье | Блок предназначен для применения преобразования Фурье к последовательностям действительных чисел.  Свойства блока:   * Размер серии – величина серии обрабатываемых данных (целое число). |  |
|  | Обратное быстрое преобразование Фурье | Блок предназначен для применения обратного преобразования Фурье к последовательностям комплексных чисел.  Свойства блока:   * Размер серии – величина серии обрабатываемых данных (целое число). |  |

Средний размер области данных на 1 блок единичной размерности составляет 25-30 байт для всей номенклатуры блоков.

Приложение Б. Список функций языка программирования для генерации кода.

| Наименование функции | Описание и свойства функций | Объём памяти данных, байт |
| --- | --- | --- |
| Оператор **+** | Сложение двух выражений. Для строк сложение соответствует конкатенации. Матрицы и вектора складываются поэлементно. |  |
| Оператор **-** | Вычитание из первого выражения второго. Для строк не определено. Матрицы и вектора вычитаются поэлементно. |  |
| Оператор **\*** | Умножение одной переменной на другую |  |
| Оператор **/** | Деление первого выражения на второе |  |
| Оператор **^** | Возведение первого выражение в степень второго. Вектора обрабатываются поэлементно. Матрицы возводятся в целую степень по правилам умножения матриц. |  |
| Оператор **.\*** | Поэлементное перемножение действительных или комплексных матриц |  |
| Оператор **./** | Поэлементное деление действительных или комплексных матриц |  |
| Оператор **.^** | Поэлементное возведение в степень двух матриц или матрицы в действительную степень |  |
| Оператор **>** | Один элемент больше другого |  |
| Оператор **<** | Один элемент меньше другого |  |
| Оператор **>=** | Один элемент больше или равен другому |  |
| Оператор **<=** | Один элемент меньше или равен другому |  |
| Оператор **<>** | Элементы не равны друг другу |  |
| Оператор **not** | Оператор целочисленного или логического отрицания |  |
| Оператор **and** | Оператор побитового логического И |  |
| Оператор **or** | Оператор побитового логического ИЛИ |  |
| Оператор **xor** | Оператор побитового логического исключающего ИЛИ |  |
| Оператор **shl** | Оператор побитового сдвига влево |  |
| Оператор **shr** | Оператор побитового сдвига вправо |  |
| Оператор **mod** | Оператор получения остатка от целочисленного деления |  |
| Оператор **div** | Оператор целочисленного деления |  |
| Оператор **#** | Оператор размножения |  |
| Оператор **..** | Оператор вычисления интервала целых чисел |  |
| Оператор [] | Операторы доступа к элементу массива и упаковки элементов в вектор |  |
| Функция **abs** | Функция получения модуля вещественного или комплексного числа или вектора  x-входное значение или входной массив, содержащий элементы вектора  Входные значения могут быть как вещественными, так и комплексными числами. |  |
| Функция **sin** | Функция вычисления синуса вещественного или комплексного числа  x – входное значение.  Входное значение может быть как вещественным, так и комплексным числом. Входное значение может задаваться как заранее определенная переменная или как постоянное число. Комплексное число задается выражением a+bi, где a и b вещественные и мнимые части числа соответственно. |  |
| Функция **cos** | Функция вычисления косинуса вещественного или комплексного числа  x – входное значение.  Входное значение может быть как вещественным, так и комплексным числом. Входное значение может задаваться как заранее определенная переменная или как постоянное число. Комплексное число задается выражением a+bi, где a и b вещественные и мнимые части числа соответственно. |  |
| Функция **ch** | Функция вычисления гиперболического косинуса вещественного или комплексного числа  x – входное значение.  Входное значение может быть как вещественным, так и комплексным числом. Входное значение может задаваться как заранее определенная переменная или как постоянное число. Комплексное число задается выражением a+bi, где a и b вещественные и мнимые части числа соответственно. |  |
| Функция **sh** | Функция вычисления гиперболического синуса вещественного или комплексного числа  x – входное значение.  Входное значение может быть как вещественным, так и комплексным числом. Входное значение может задаваться как заранее определенная переменная или как постоянное число. Комплексное число задается выражением a+bi, где a и b вещественные и мнимые части числа соответственно. |  |
| Функция **th** | Функция вычисления гиперболического тангенса вещественного или комплексного числа  x – входное значение.  Входное значение может быть как вещественным, так и комплексным числом. Входное значение может задаваться как заранее определенная переменная или как постоянное число. Комплексное число задается выражением a+bi, где a и b вещественные и мнимые части числа соответственно. |  |
| Функция **exp** | Функция вычисления экспоненты вещественного или комплексного числа  x – входное значение.  Входное значение может быть как вещественным, так и комплексным числом. Входное значение может задаваться как заранее определенная переменная или как постоянное число. Комплексное число задается выражением a+bi, где a и b вещественные и комплексные части числа соответственно. |  |
| Функция **sqrt** | Функция вычисления корня квадратного вещественного или комплексного числа  x – входное значение.  Входное значение должно быть положительным числом. Входное значение может задаваться как заранее определенная переменная или как постоянное число. Входное значение может быть как вещественным, так и комплексным числом. Если входное значение – вещественное число, то возвращаемое значение тоже вещественное число. Комплексное число задается выражением a+bi, где a и b вещественные и комплексные части числа соответственно. |  |
| Функция **tg** | Функция вычисления тангенса вещественного или комплексного числа  x – входное значение.  Входное значение должно быть положительным числом. Входное значение может задаваться как заранее определенная переменная или как постоянное число. Входное значение может быть как вещественным, так и комплексным числом. Если входное значение – вещественное число, то возвращаемое значение тоже вещественное число. Комплексное число задается выражением a+bi, где a и b вещественные и комплексные части числа соответственно. |  |
| Функция **arctg** | Функция вычисления арктангенса вещественного или комплексного числа  x – входное значение.  Входное значение может быть как вещественным, так и комплексным числом. Вещественное входное значение ограничено условием: -1≤x≤1. Входное значение может задаваться как заранее определенная переменная или как постоянное число. Комплексное число задается выражением a+bi, где a и b вещественные и мнимые части числа соответственно. |  |
| Функция **arcsin** | Функция вычисления арксинуса вещественного или комплексного числа  x – входное значение.  Входное значение может быть как вещественным, так и комплексным числом. Вещественное входное значение ограничено условием:-1≤x≤1. Входное значение может задаваться как заранее определенная переменная или как постоянное число. Комплексное число задается выражением a+bi, где a и b вещественные и мнимые части числа соответственно. |  |
| Функция **arccos** | Функция вычисления арккосинуса вещественного или комплексного числа  x – входное значение.  Входное значение может быть как вещественным, так и комплексным числом. Вещественное входное значение ограничено условием:-1≤x≤1. Входное значение может задаваться как заранее определенная переменная или как постоянное число. Комплексное число задается выражением a+bi, где a и b вещественные и мнимые части числа соответственно. |  |
| Функция **ln** | Функция вычисления натурального логарифма вещественного или комплексного числа  x – входное значение.  Входное значение может быть как вещественным, так и комплексным числом. Входное значение может задаваться как заранее определенная переменная или как постоянное число. Комплексное число задается выражением a+bi, где a и b вещественные и комплексные части числа соответственно. |  |
| Функция **lg** | Функция вычисления десятичного логарифма вещественного или комплексного числа  x – входное значение.  Входное значение может быть как вещественным, так и комплексным числом. Входное значение может задаваться как заранее определенная переменная или как постоянное число. Комплексное число задается выражением a+bi, где a и b вещественные и комплексные части числа соответственно. |  |
| Функция **complex** | Функция получения комплексного числа  a – вещественная часть комплексного числа,  b – мнимая часть комплексного числа |  |
| Функция **arg** | Функция получения аргумента комплексного числа  x – входное значение комплексного числа.  Аргументом комплексного числа называется величина угла, образованного на комплексной плоскости вектором, определяемым комплексным числом, с вещественной осью. |  |
| Функция **real** | Функция получения вещественной части комплексного числа  x – входное значение комплексного числа. |  |
| Функция **case** | функция выбора одного из входных аргументов  x0, x2, …, xn по номеру, определяемому аргументом i. Нумерация аргументов начинается с нуля.  i – номер выбираемого аргумента, имеет тип integer,  x0, x2, …, xn – входные аргументы. |  |
| Функция **imag** | Функция получения мнимой части комплексного числа  x – входное значение комплексного числа. |  |
| Функция **atan2** | Функция вычисления арктангенса точки с учетом квадранта  y – координата по оси ординат (действительное число или массив действительных чисел);  x – координата по оси абсцисс (действительное число или массив действительных чисел). |  |
| Функция **min** | Функция вычисления минимального значения из двух значений или из значений вектора  a – первое входное значение,  b – второе входное значение,  Х – входной массив, содержащий элементы вектора. |  |
| Функция **max** | Функция вычисления максимального значения из двух значений или из значений вектора  a – первое входное значение,  b – второе входное значение,  Х – входной массив, содержащий элементы вектора. |  |
| Функция **sign** | Функция определяет знак числа  x – входное значение, число. |  |
| Функция **extract** | Функция извлечения элементов из матрицы или вектора по номерам  M – входная матрица,  A – входной массив номеров строк извлекаемых элементов матрицы M,  B – входной массив номеров столбцов извлекаемых элементов матрицы M,  X – входной вектор,  Y – входной массив номеров извлекаемых элементов вектора X. |  |
| Функция **cols** | Функция возвращает количество столбцов матрицы или количество элементов массива  Х – входной массив.  M – входная матрица. |  |
| Функция **transp** | Функция транспонирования матрицы или вектора  M – массив элементов входной матрицы или вектора. |  |
| Функция **lsolve** | Функция решения системы линейных алгебраических уравнений (СЛАУ)  X – входной массив элементов [[a11, ..,a1n],…,[ am1, ..,amn]] , определяющий матрицу системы линейных алгебраических уравнений А,  Y – входной массив элементов [b11, ..,b1m], определяющий вектор B системы линейных алгебраических уравнений А. |  |
| Функция **det** | Функция возвращает значение детерминанта матрицы. Матрица M должна быть квадратной.  M – входная матрица. |  |
| Функция **polyroots** | Функция вычисления корней характеристического полинома вещественной матрицы  M – входная матрица. |  |
| Функция **Interpol** | Функция двумерной линейной, трехмерной линейной, кубической сплайн- интерполяции аргумента x  Х – входной массив координат точек по оси x для вычисления линейной интерполяции,  Y – входной массив координат точек по оси y для вычисления линейной интерполяции,  Z – входной массив координат точек по оси z для вычисления линейной интерполяции,  x,y – аргументы для интерполяции,  M – матрица коэффициентов кубической сплайн-интерполяции. |  |
| Функция **vector** | Функция создания нулевого вектора  n – количество элементов нулевого вектора. |  |

Лист регистрации изменений

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Изм. | Номера листов | | | | Всего листов в документе | № документа | Вх. № сопроводительного документа и дата | Подп. | Дата |
| измененных | замененных | новых | аннулиро-  ванных |
|  |  |  |  |  |  |  |  |  |  |