

Группа УМИ - ООО «УМИКОН»

**КОМПЛЕКС
ИЗМЕРИТЕЛЬНО-ИНФОРМАЦИОННЫЙ И
УПРАВЛЯЮЩИЙ
(ПРОГРАММНО-ТЕХНИЧЕСКИЙ)
«УМИКОН»**

**КОМПЛЕКС ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ
«МИКСИС»**

Система алгоблочного программирования верхнего уровня

RSPROG

РУКОВОДСТВО ПОЛЬЗОВАТЕЛЯ

4252-0050-85646258 ИЗ

1. ОБЩИЕ СВЕДЕНИЯ	4
2. ЗАПУСК ПРОГРАММЫ	6
3. ОСНОВНЫЕ ФУНКЦИИ	7
4. УПРАВЛЕНИЕ ИСПОЛНЕНИЕМ.....	10
5. РЕДАКТИРОВАНИЕ ПРОЦЕДУР И АЛГОБЛОКОВ	14
6. ПРОСМОТР ЗНАЧЕНИЙ	28
7. ПРОСМОТР И ИНИЦИАЛИЗАЦИЯ БАЗ.....	29
8. РЕДАКТОР БАЗ	31
9. ГРУППЫ УЧЕТА.....	34
ПРИЛОЖЕНИЕ А	37
A.1 Общие операции	38
A.1.1 Копирование.....	38
A.1.2 Загрузка и сохранение БД НСИ	39
A.1.3 Чтение из трендов.....	40
A.1.4 Описание сигнала.....	41
A.2 Управление	43
A.2.1 Вызов процедуры.....	43
A.2.2 Вызов подпрограммы	43
A.2.3 Группы учета	45
A.2.4 Переход.....	45
A.2.5 Возврат	46
A.3 Арифметические операции	48
A.3.1 Элементарные бинарные операции.....	48
A.3.2 Элементарные унарные операции	51
A.3.3 Взвешенные операции	53
A.3.4 Попадание в Диапазон (арифметическое).....	55
A.3.5 Селектор Арифметический.....	56
A.3.6 Преобразование.....	57
A.3.7 Прогрессия	61
A.3.8 Система линейных уравнений	61
A.3.9 Определение коэффициентов полинома	62
A.3.10 Масштабирование.....	63
A.3.11 Выборка	64
A.3.12 Коммутатор.....	66
A.4 Логические операции	69
A.4.1 AND, OR, XOR, Штрих Шеффера, Стрелка Пирса, Эквивалентность, Следование, НЕ Следование, Обратное Следование, НЕ Обратное Следование	70
A.4.2 Операции сравнения с уставками.....	71
A.4.3 Операции сравнения значений: <=,>=,#,<,>	73
A.4.4 Операция отрицания NOT	73
A.4.5 Попадание в диапазон (логическое)	74
A.4.6 Селектор логический	77
A.4.7 Триггер	79
A.4.8 Фильтр дискретный.....	82
A.4.9 Генератор импульсов.....	84
A.4.10 Фильтр Аперидический	86
A.4.11 Отклонения	90
A.4.12 Квитирование	92
A.4.13 Анализ статуса (Win&Lin)	94

A.5 Динамика	96
A.5.1 Интегратор по методу прямоугольника	96
A.5.2 Дифференциатор.....	98
A.5.3 Звено запаздывания	100
A.5.4 Фильтр скользящего среднего	101
A.5.5 Ограничение	104
A.5.6 ШИМ	107
A.5.7 Аперiodическое / Колебательное звено	109
A.5.8 Статистика (среднее, минимальное, максимальное)	112
A.5.9 Фильтр	113
A.5.10 Идентификация модели	116
A.6 Таймеры	118
A.6.1 Таймер циклический.....	118
A.6.2 Таймер астрономический.....	121
A.6.3 Хронограф	124
A.6.4 Циклограмма	128
A.6.5 Управление оборудованием	130
A.6.6 Командный таймер	132
A.6.7 Дата и время	133
A.6.8 Интервал	134
A.7 Регуляторы	136
A.7.1 ПТОTo2 П - регулятор.....	136
A.7.2 ПТОTo2 И - регулятор	136
A.7.3 ПТО Т П - регулятор	137
A.7.4 ПТО Т И - регулятор.....	138
A.7.5 ПТО Т ПР - регулятор.....	139
A.7.6 ПТО Т ПРБ - регулятор.....	140
A.7.7 Использование операндов в регуляторах.....	141
A.7.8 Формирование выхода в регуляторах.....	142
A.7.9 Идентификация объекта управления	143
A.7.10 Изменение такта регулирования	144
A.8 Резервирование	145
A.8.1 Резервирование	145
A.8.2 Исправность.....	148
A.8.3 Управление БПР	149
A.8.4 Диагностика	150
A.9 Теплофизика	152
A.9.1 Энтальпия.....	152
A.9.2 Плотность	152
A.10 Специализированные алгоблоки	154
A.10.1 Расчет угла поворота сельсинной пары	154
A.11 Учет и ТЭП	155
A.11.1 Приведение накопления	155
A.12 Циклы	156
A.12.1 Цикл обхода (алгоблок цикла).....	156

1. ОБЩИЕ СВЕДЕНИЯ

1.1. Система программирования на языке алгоблоков предназначена для создания пользовательских процедур, исполняемых программой MWBridge в режиме реального времени.

1.2. Алгоблок - это набор двоичных данных, определяющих требуемую операцию следующим образом:

- операнды алгоблока (один или два в зависимости от типа операции);
- результат исполнения операции;
- тип операции;
- условия выполнения операции;
- условия использования операндов;
- результат (логический) выполнения операции.

Каждый набор данных также включает в себя дополнительные флаги, уточняющие режимы использования данных наборов данных.

1.3. Процедуры представляют собой набор алгоблоков, исполняемых последовательно друг за другом.

Исполнение процедур производится по специальному циклическому алгоритму разделения времени.

Процедуры, входящие в цикл исполнения, называются циклами обхода. Количество циклов обхода (т.е. постоянно выполняющихся процедур) определяется оператором на этапе конструирования алгоблоков (циклы обхода образуют первые процедуры). Первая процедура присутствует всегда и образует быстрый цикл обхода, безусловно исполняемый на каждом такте работы ядра реального времени. Для организации вычислительного процесса оператору предоставляется специальный алгоблок цикла **Переход на предыдущий цикл обхода**, который позволяет прервать исполнение текущего цикла обхода (кроме первого) и перейти к исполнению предыдущего более быстрого цикла обхода (исполняется процедура с меньшим номером). Данный алгоблок может быть сделан условным (по значению определенной переменной или времени), т.о. алгоритм организации вычислительного процесса может быть изменен во время работы программы.

Пример организации вычислительного процесса для случая:

- циклов обхода - 3;
- 2-й и 3-й циклы обхода имеют внутри себя алгоблок цикла, делящий эти процедуры на две части: 2-1,2-2 и 3-1,3-2 соответственно (алгоблок цикла осуществляет безусловную передачу управления предыдущему циклу обхода).

Номер такта ядра реального времени	Исполняемые процедуры
1	1 + 2-1
2	1 + 2-2
3	1 + 3-1
4	1 + 2-1
5	1 + 2-2
6	1 + 3-2

Далее последовательность повторяется.

Таким образом, в данном примере 1-й цикл обхода (процедура N1) выполняется в 3 раза чаще 2-го и в 6 раз чаще 3-го.

Реальное исполняемое количество циклов обхода может быть меньше заданного в том случае, если какая-то из процедур, входящих в циклы обхода не определена (количество алгоблоков в ней равно 0).

Процедуры, с номерами большими количества циклов обхода, не входят в алгоритм непрерывного циклического исполнения и могут быть запущены только, используя специальный алгоблок **Вызов процедуры**.

Для создания процедур на основе алгоблоков используется программа-конструктор алгоблоков rsprog.exe.

2. ЗАПУСК ПРОГРАММЫ

2.1. Подсистема исполнения алгоблоков запускается автоматически программой MWBridge.exe.

2.2. Программа-конструктор алгоблоков может быть запущена одним из следующих способов:

- из программы MWBridge путем вызова пункта меню АлгоВУ.
- из редактора мнемосхем системы отображения Display посредством вызова пункта меню Инструменты->АлгоВУ;
- автономно, путем запуска исполняемого файла rsprog.exe из каталога установки системы отображения Display.

2.3. Для работы программы необходимо указание текущего рабочего проекта. Если в момент запуска программы, на компьютере работает программа MWBridge, то в качестве рабочего проекта будет использован текущий рабочий проект MWBridge, в противном случае программа выдаст диалог выбора каталога проекта.

3. ОСНОВНЫЕ ФУНКЦИИ

3.1. После запуска программы на экране отобразится главное окно, представленное на рис.1.

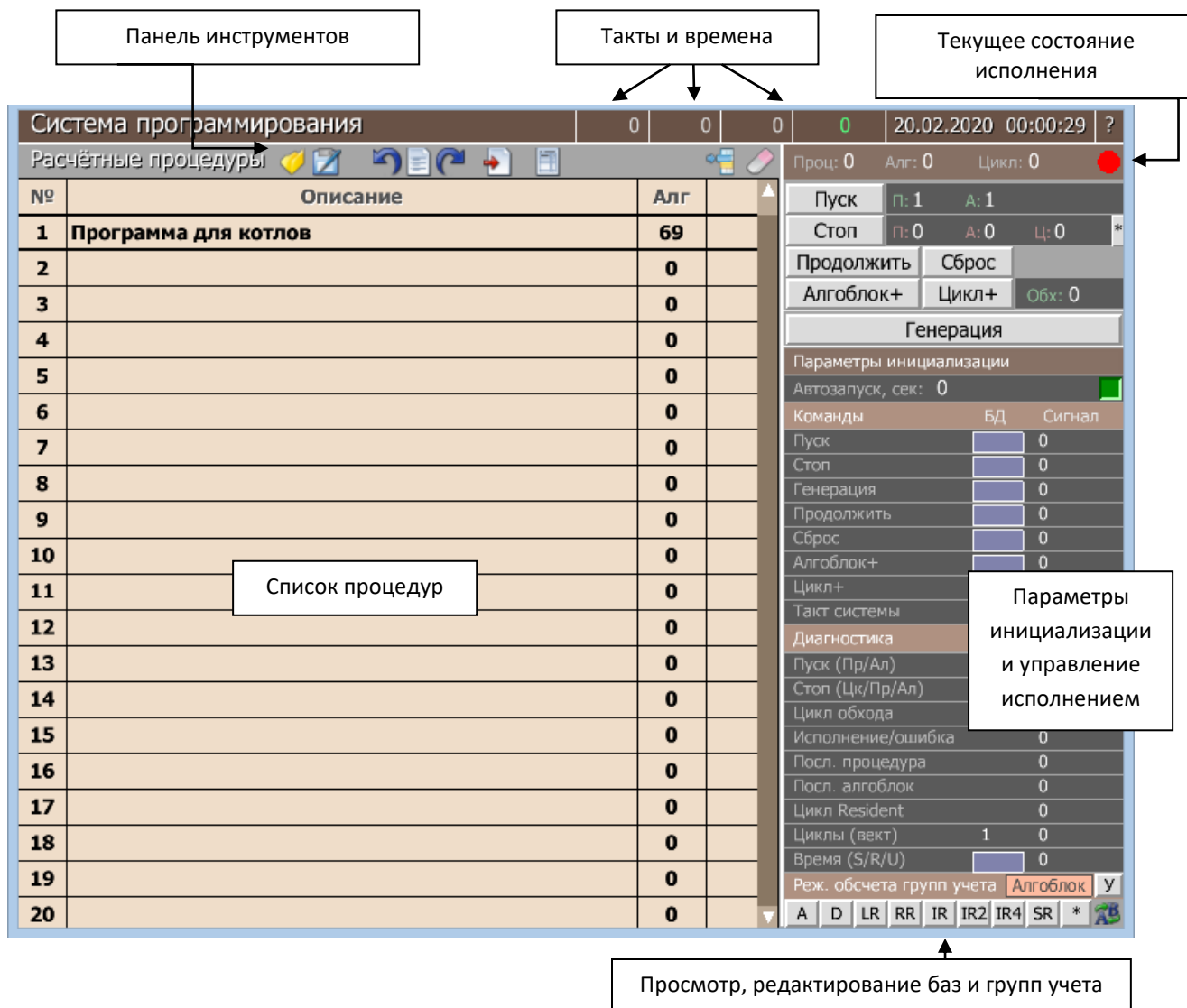


Рис. 1 – Главное окно программы

Панель инструментов содержит в себе инструменты для выполнения стандартных операции для работы с изображением и буфером обмена:

- открыть файл процедур;
- сохранить;
- отменить последнее действие;
- история правок;
- повторить последнее отмененное действие;
- импорт процедур из другого проекта;



- листинг программы;



- заменить процедуру;



- стереть содержимое процедуры.

Область **такты и времена** содержит четыре поля:

- текущий такт работы системы, в миллисекундах;
- время работы исполняющей системы, в тактах;
- время работы алгоблочной программы на такте, в миллисекундах;
- поле для установки нового такта работы системы в миллисекундах.



Текущее состояние исполнения отображает номер текущей выполняемой процедуры, номер текущего выполняемого алгоблока, номер цикла, а также общее состояние исполняющей системы (зеленый индикатор – работает, красный индикатор – остановлено).


Область **параметров инициализации и управления исполнением** предназначена для запуска, останова, пошагового исполнения (отладки) программы, а также настройки автоматического срабатывания этих функций по сигналу привязанной переменной БД.

Кнопки **просмотра и редактирования баз** предназначены для просмотра в табличном виде сигналов баз данных системы, редактирования их параметров и начальных значений, а также анализа использования сигналов в качестве операндов или результата алгоблоков.

Список процедур представляет собой таблицу с перечнем процедур алгоблочной программы. Таблица содержит четыре столбца:

- Порядковый номер процедуры;
- Описание процедуры;
- Количество алгоблоков в процедуре;
- Флаг выделения процедуры.

3.2. Все производимые действия по редактированию алгоблочной программы записываются в журнал правок. Отменить последнюю правку можно путем нажатия кнопки  на панели инструментов. Повторное нажатие кнопки произведёт отмену предпоследней правки и т.д. Заново произвести отмененное действие можно путем нажатия кнопки .

Полный перечень записей журнала правок можно увидеть в окне журнала правок, которое отображается по нажатию кнопки  на панели инструментов. Окно журнала правок представлено на рис.2.

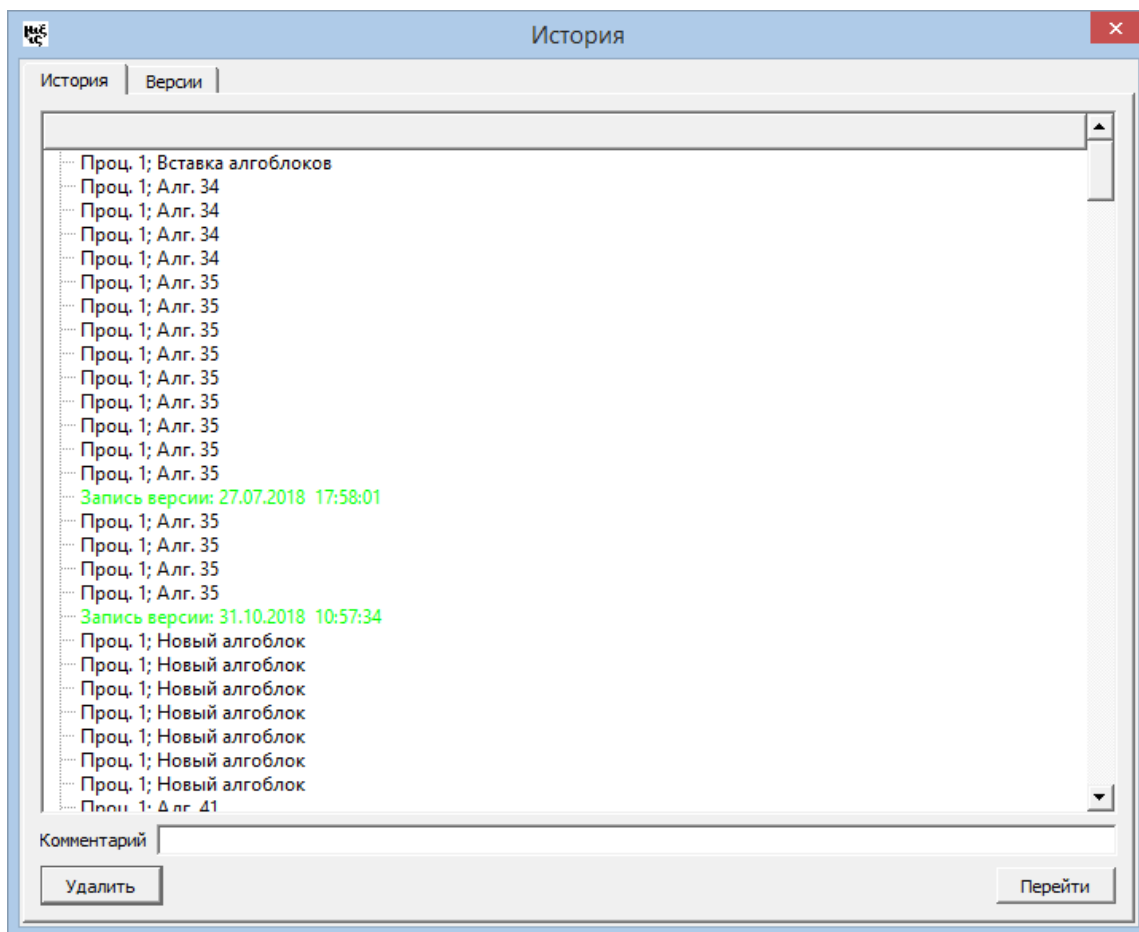


Рис.2 – История правок

Каждая запись журнала сопровождается информацией о том, в какой процедуре и в каком алгоблоке произведена соответствующая правка. Журнал позволяет откатить произвольное количество шагов и восстановить состояние редактируемой программы на момент до внесения соответствующих изменений.

Если после отката некоторого количества изменений начать редактирование программы, то отмененные изменения не теряются, а выделяются в отдельную ветвь правок, т.е. журнал изменений обладает древовидной структурой.

Каждое сохранение алгоблочной программы в файл называется «версией», таким образом логически завершая некоторую последовательность правок программы. Версии представлены в окне журнала на отдельной вкладке «Версии».

4. УПРАВЛЕНИЕ ИСПОЛНЕНИЕМ

4.1. Кнопки управления исполняющей системой располагаются в правой части главного окна и приведены на рис.3.

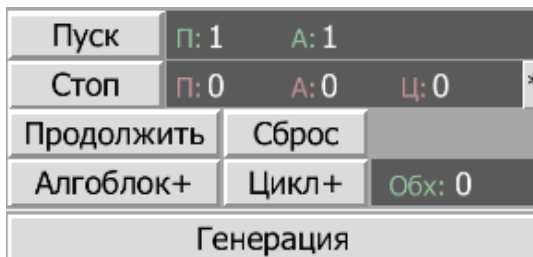


Рис.3 – Кнопки управления исполнением

Кнопка **Пуск** выполняет старт программы с указанной процедуры (П) и алгоблока (А). Если номера процедуры (П), алгоблока (А), и цикла обхода (Ц) напротив кнопки **Стоп** не равны нулю, то программа автоматически остановится по их достижении.

Кнопка **Стоп** выполняет останов выполнения программы. При этом в полях П, А и Ц отобразятся номера процедуры, алгоблока и цикла обхода, на которых произведен останов.

Кнопка **Продолжить** запускает выполнение программы после останова с текущей позиции.

Кнопка **Сброс** выполняет обнуление счетчиков исполнения для всех процедур, входящих в циклы обхода.

Кнопка **Алгоблок+** предназначена для пошагового исполнения программы в режиме останова и выполняет следующий по порядку алгоблок. При этом в полях П, А и Ц отобразятся номера процедуры, алгоблока и цикла обхода, на которых произведен останов после выполнения алгоблока.

Кнопка **Цикл+** производит исполнение программы на один цикл указанной в поле Обх процедуры. При этом в полях П, А и Ц отобразятся номера процедуры, алгоблока и цикла обхода, на которых произведен останов после выполнения алгоблока.

Кнопка **Генерация** производит перезагрузку расчетных массивов и программы. Если программа до этого исполнялась, производится повторный запуск.

Кнопка * отображает окно редактирования точек останова. Точки останова представляют собой наборы номеров процедур, алгоблоков и циклов обхода, по достижению которых произойдет автоматическая остановка исполнения программы. Точки останова используются в целях отладки. Подробнее см. далее п.4.5.

4.2. **Параметры инициализации** включают в себя флаг **Автозапуск**, определяющий режим запуска программы после загрузки. При включенном флаге автозапуска программа начинается исполняться сразу после загрузки с опциональной задержкой, указываемой в секундах. Если режим автозапуска отключен, то программа не исполняется после старта системы; запуск возможен только двумя способами: с использованием кнопок управления исполнением или путем установки единицы в управляющей переменной, привязанной к команде «Пуск» (см. далее п.4.3).

4.3. Блок **Команды** определяет сигналы БД, используемые для передачи команд исполняющей системе. Для установки сигнала следует в столбце «БД» в выпадающем меню выбрать тип БД, а в поле «Сигнал» указать номер сигнала этой БД, который будет использоваться в качестве управляющего.

Наличие команды определяется значением переменной управляющего сигнала:

0 – нет команды;

1 – есть команда; после исполнения команды исполняющая система сбрасывает управляющую переменную в 0.

Команды	БД	Сигнал
Пуск	<input type="text"/>	0
Стоп	<input type="text"/>	0
Генерация	<input type="text"/>	0
Продолжить	<input type="text"/>	0
Сброс	<input type="text"/>	0
Алгоблок+	<input type="text"/>	0
Цикл+	<input type="text"/>	0
Такт системы	<input type="text"/>	0

Рис.4 – Привязка команд к сигналам БД

4.4. Блок **Диагностика** предназначен для указания сигналов БД, используемых для передачи параметров в определенных командах, а также для приема результатов работы исполняющей системы.

Параметры полей блока «Диагностика» приведены в таблице:

Параметр	Значение
Пуск (Пр/Ал)	Определяют стартовую процедуру и алгоблок
Стоп (Цк/Пр/Ал)	При старте: определяют точку останова; При останове: возвращают точку останова
Цикл обхода	Для команды «Цикл+» определяет номер процедуры, входящей в цикл обхода
Исполнение/Ошибка	Возвращает значение «1» при работающей программе и «0» при неработающей. Отрицательные значения означают ошибки: -1 – ошибка загрузки Progres.nsi; -2 – ошибка подключения к БД MWBridge; -3..-8 – ошибка обработки файлов БД RS\$RR,RS\$LR,RS\$SR,RS\$IR,RS\$IR2, RS\$IR4 соответственно; -9 – недостаточно оперативной памяти для загрузки расчетных БД; -10 – недостаточно оперативной памяти для загрузки процедуры; -11 – Ошибка обработки файла процедур; -12 - Win-программа не может исполняться под DOS; -13 – ошибка распределения памяти под область предыдущих значений; -14 – отсутствует файл процедур; -15 – размер файла процедур не соответствует требуемому; -16 – разрушена внутренняя структура файла процедур; -17 – не заданы процедуры для обсчета; -18 – ошибка создания файла Procedur.prs;

	-19 – ошибка записи в файл Procedur.prs; -20 – ошибка контрольной суммы.
Посл. процедура Посл. алгоблок	Возвращает номера процедуры и алгоблока, на которых завершено исполнение на текущем такте работы исполняющей системы
Цикл Resident	Возвращает текущее значение цикла: 0 – старт, 1 – исполнение, 2 – завершение
Цикл (вектор)	Возвращает текущие значения счетчиков циклов для процедур, входящих в цикл обхода (количество процедур определяется размерностью вектора, но не больше циклов обхода)
Время (S/R/U)	Возвращает время в миллисекундах: S – основной такт работы системы; R – такт работы Resident; U – такт работы исполняющей системы обработки алгоблоков

4.5. Окно редактирования точек останова представлено на рис.5.

Точки останова 							Загрузить точки останова
№	Вкл	Процедура	Алгоблок	Цикл	Упр.сиг.	Инв.	Комментарий
1		1	3	1			
2		1	4	2			
3							
4							
5							
6							
7							
8							
9							
10							
11							
12							
13							
14							
15							
16							
17							
18							
19							
20							
21							
22							
23							
24							
25							
26							
27							
28							
29							
30							
31							
32							

Рис.5 – Точки останова

Для включения режима обработки точек останова следует нажать кнопку переключателя, расположенную возле заголовка окна.

Окно включает в себя таблицу с перечнем точек останова. Для каждой точки останова определяются следующие параметры:

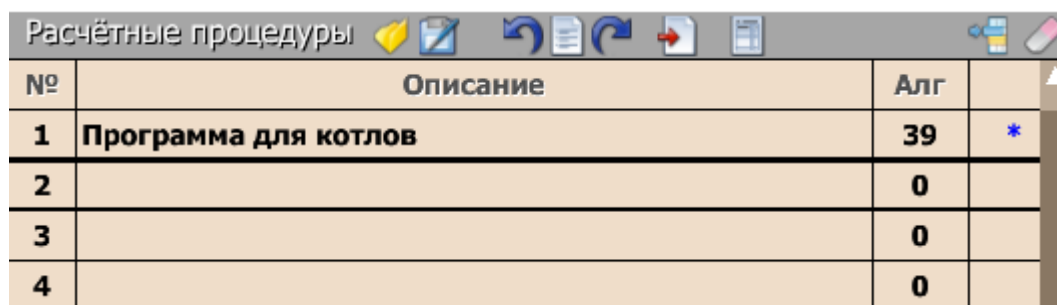
- индивидуальный переключатель активности точки останова (вкл/выкл);
- номер процедуры;
- номер алгоблока;

- номер цикла;
- управляющий сигнал БД, а также флаг инверсии значения управляющего сигнала;
- комментарий.

Кнопка **Загрузить точки останова** производит загрузку точек останова в исполняющую систему в режиме «налету».

5. РЕДАКТИРОВАНИЕ ПРОЦЕДУР И АЛГОБЛОКОВ

5.1. Список процедур алгоблочной программы располагается в левой части главного окна и представлен на рис.6.





№	Описание	Алг	
1	Программа для котлов	39	*
2		0	
3		0	
4		0	

Рис.6 – Список процедур

В первом столбце отображается порядковый номер процедуры, во втором столбце отображается символьное описание процедуры, в третьем столбце приведено количество алгоблоков в процедуре, четвертый столбец служит для выделения процедур.

Нажатие мышью в крайнем левом столбце списка процедур позволяет **выбрать** текущую процедуру. Выбранная процедура помечается рамкой. Нажатие мышью в крайнем правом столбце списка процедур осуществляет **выделение** процедуры. Выделенная процедура помечается символом *. При этом удержание клавиши Ctrl позволяет добавить выделяемую процедуру к ранее выделенным, удержание клавиши Shift позволяет выделить диапазон процедур, удержание клавиши Alt снимает выделение с процедуры.

Кнопка  осуществляет **замену** выбранной процедуры выделенной процедурой (дублирование). Если выделено несколько процедур, то будет осуществлена замена нескольких процедур, начиная с выбранной.

Кнопка  осуществляет **стирание** выделенных процедур: очистку описание и удаление всех алгоблоков.

5.2. Нажатие мышью во втором столбце открывает окно процедуры, приведенное на рис.7.

Для редактирования **описания процедуры** следует нажать мышью в поле Описание процедуры, ввести новое описание и нажать клавишу Enter. Нажатие клавиши Esc осуществляет выход из режима редактирования описания без сохранения изменений.

Список алгоблоков представляет собой таблицу, каждая строка которой соответствует одному алгоблоку. Первый столбец таблицы содержит порядковый номер алгоблока, во втором столбце отображается символьное описание алгоблока, в третьем столбце приведено состояние алгоблока (вкл/выкл), четвертый столбец содержит название выполняемой операции, пятый столбец служит для выделения алгоблоков.

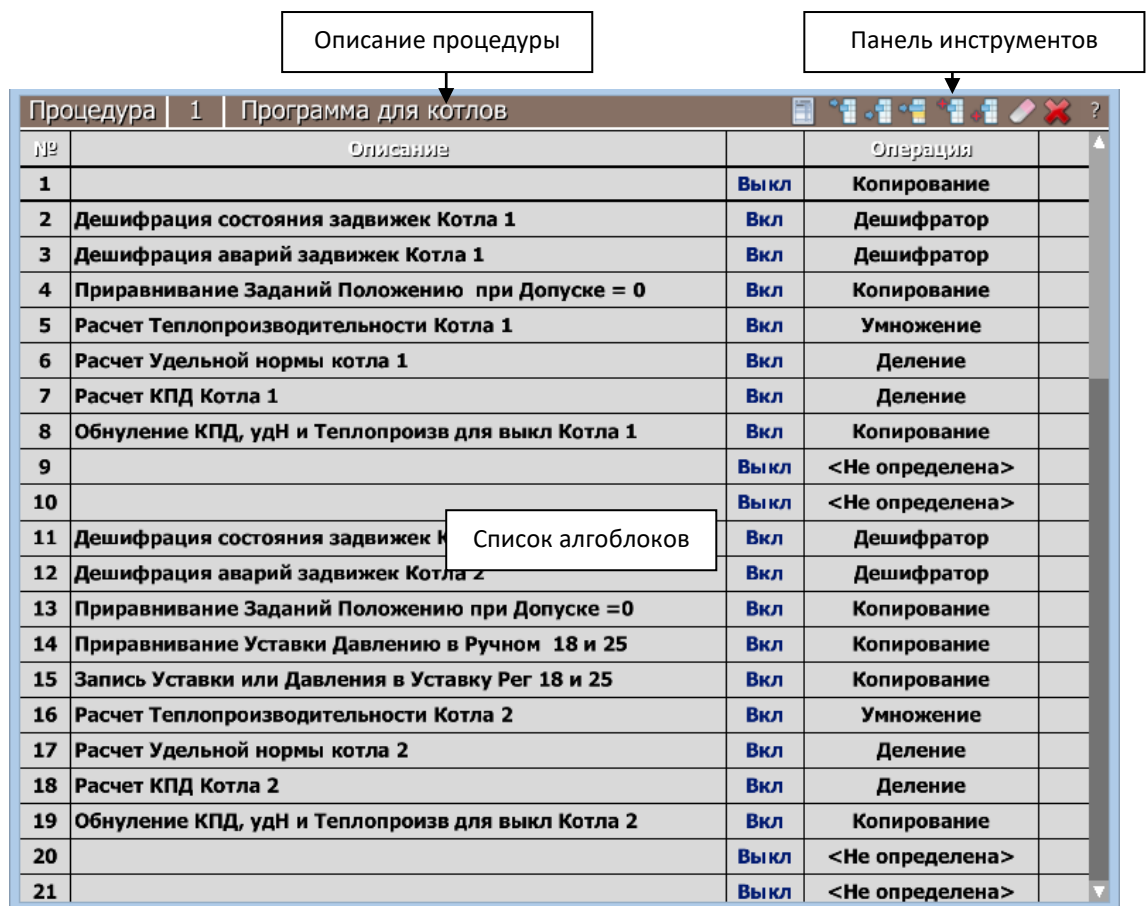


Рис.7 – Окно процедуры

Нажатие мышью в крайнем левом столбце списка процедур позволяет **выбрать** текущий алгоблок. Выбранный алгоблок помечается рамкой. Нажатие мышью в крайнем правом столбце списка алгоблоков осуществляет **выделение** алгоблока. Выделенный алгоблок помечается символом *. При этом удержание клавиши Ctrl позволяет добавить выделяемый алгоблок к ранее выделенным, удержание клавиши Shift позволяет выделить диапазон алгоблоков, удержание клавиши Alt снимает выделение с алгоблока.

Панель управления содержит в себе следующие инструменты:

- сохраняет **листинг процедуры** в указанный каталог для последующей печати;
- дублирует текущий выделенный алгоблок и **вставляет перед** текущим выбранным;
- дублирует текущий выделенный алгоблок и **вставляет после** текущего выбранного;
- дублирует текущий выделенный алгоблок и **заменяет** текущий выбранный;
- **добавляет** новый пустой алгоблок **перед** текущим выбранным;
- **добавляет** новый пустой алгоблок **после** текущим выбранным;
- **очищает** текущий выделенный алгоблок (удаление без сдвига остальных алгоблоков);
- **удаляет** текущий выделенный алгоблок (со сдвигом остальных алгоблоков);
- вывод справочной информации по процедуре.

5.3. Нажатие мышью во втором столбце списка алгоблоков открывает окно алгоблока, приведенное на рис.8.

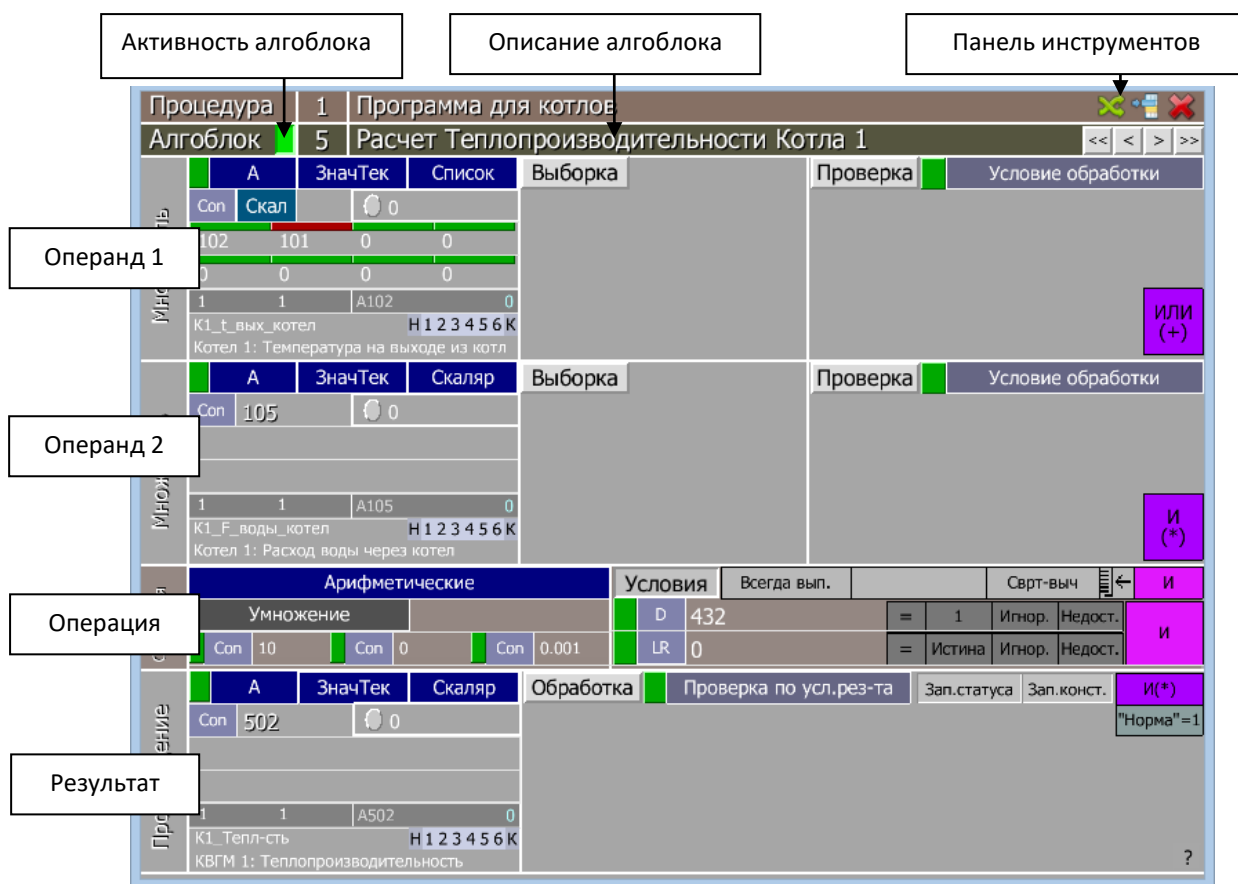






Рис.8 – Окно алгоблока




Для включения алгоблока следует нажать левой клавишей мыши на кнопке **Активность алгоблока**. Нажатие правой клавишей мыши выключает алгоблок.

Для редактирования **описания алгоблока** следует нажать мышью в поле Описание алгоблока, ввести новое описание и нажать клавишу Enter. Нажатие клавиши Esc осуществляет выход из режима редактирования описания без сохранения изменений.

Нажатие мышью в крайнем левом поле операндов, операции и результата позволяет **выделить** соответствующий элемент для последующих манипуляций над ними с помощью инструментов панели управления. Выделенный элемент помечается символом *.

Панель управления содержит в себе следующие инструменты:

-  - **меняет местами операнды** алгоблока;
-  - дублирует выделенный элемент алгоблока (в текущем окне или в окне другого алгоблока) и **заменяет** соответствующий элемент алгоблока в текущем окне;
-  - **удаляет** (очищает) выделенный элемент алгоблока;
-  - переключает текущий алгоблок в окне на предыдущий алгоблок;

-  - переключает текущий алгоблок в окне на следующий алгоблок;
-  - перелистывает на 10 алгоблоков назад;
-  - перелистывает на 10 алгоблоков вперед.

5.4. Поле операнда состоит из трех секций:

- источник данных;
- выборка;
- проверка.

Источник данных определяет тип и формат источника данных операнда. Выборка задаёт фильтрация источника данных по определенным критериям. Секция проверки позволяет задать параметры проверки операндов на достоверность.

5.5. Секция источника данных приведена на рис. 9.

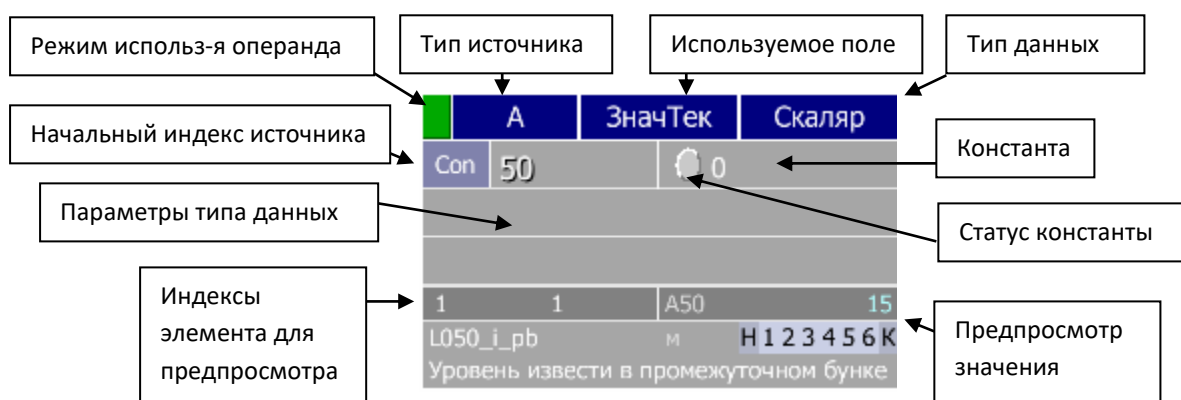






Рис.9 – Источник данных

Режим использования операнда переключается циклично по нажатию в поле кнопкой мыши и имеет следующие состояния:

-  - норма (операнд используется без изменений);
-  - отрицание (операнд используется с отрицанием);
-  - модуль (операнд используется по модулю);
-  - антимодуль (используется обратная величина 1/x).

Тип источника и **используемое поле** представляют собой выпадающее меню с возможностью выбора типа источника: константа, базы MWBridge, базы RSPROG; а также конкретного поля (параметра) сигнала БД, используемого в качестве значения операнда. В зависимости от типа источника меняется перечень возможных для указания полей сигнала.

Тип данных имеет следующие варианты для выбора:

- скаляр;
- вектор;
- матрица;

– список.

Начальный индекс источника задаёт номер сигнала, используемого в качестве первого элемента множества, в соответствии с выбранным типом данных: для скаляра – собственно скаляр; для вектора – первый элемент вектора; для матрицы – первый элемент первой строки матрицы; для списка – первый элемент списка. Если в сопутствующем выпадающем меню выбран пункт, отличный от Con, то актуальный начальный индекс источника будет считываться **по ссылке**, из указанной БД и сигнала с указанным номером.

Константа и **статус константы** используются в случае, если в качестве типа источника выбрана константа. Для задания константы следует нажать мышью в поле константа, ввести с клавиатуры необходимое значение и нажать Enter. Также существует возможность установки одного из предустановленных значений. Для этого следует, удерживая клавишу Shift или Alt, нажать левой или правой кнопкой мыши в поле константа. При удержании клавиши Shift циклично перебираются константы из следующего списка: 0.0, 1.175494e-38 (min_float), 0.5, 0.70710678118 (sqrt(2)/2), 0.86602540378 (sqrt(3)/2), 1.0, 1.41421356237 (sqrt(2)), 1.73205080756 (sqrt(3)), 2.71828182845 (e), 3.14159265358 (pi), 9.80665 (g), 3.402823e38 (max_float). При удержании клавиши Alt циклично перебираются константы из следующего списка: 0.0, 0.52359877560 (pi/6), 0.78539816340 (pi/4), 1.04719755120 (pi/3), 1.57079632679 (pi/2), 2.09439510239 (2*pi/3), 2.35619449019 (3*pi/4), 2.61799387799 (5*pi/6), 3.14159265359 (pi), 3.66519142919 (7*pi/6), 3.92699081699 (5*pi/4), 4.18879020479 (4*pi/3), 4.71238898038 (3*pi/2), 5.23598775598 (5*pi/3), 5.49778714378 (7*pi/4), 5.75958653158 (11*pi/6), 6.28318530718 (2*pi). Статус константы имеет два состояния: норма и недостоверно.

Поле **предварительного просмотра** предназначено для оценки используемого в операнде значения, а также статуса этого значения. Индексы элемента позволяют выбрать конкретный элемент из множества операндов в соответствии с выбранным типом данных (строка и столбец в матрице). Для просмотра всего множества значений операнда следует нажать мышью на поле предпросмотра.

Поле **параметров типа данных** имеет разную конфигурацию в зависимости от выбранного типа данных.

Для скаляров отсутствуют параметры типа данных.

Для вектора поле параметров имеет следующий вид, представленный на рис.10.

	A	ЗначТек	Вектор
Con	50	<input type="radio"/>	0
Con	2	Con	1

Количество элементов →

Шаг между элементами ←

Рис.10 – Параметры вектора

Параметры вектора задают **количество элементов вектора** и **шаг между элементами**. Номера элементов вектора рассчитываются начиная от начального индекса источника с заданным шагом. Как и начальный индекс источника, оба эти параметра могут быть заданы по ссылке.

Для матриц поле параметров имеет следующий вид, представленный на рис.11.

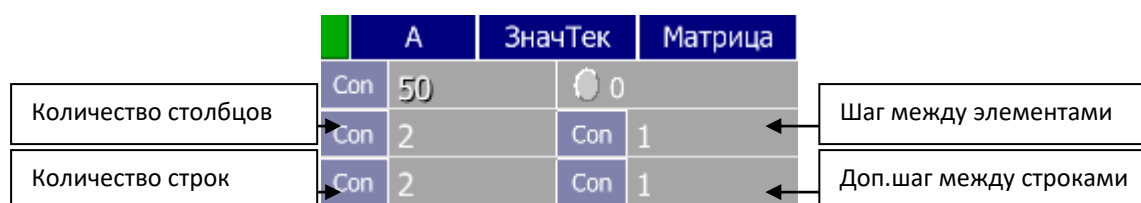


Рис.11 – Параметры матрицы

Параметры матрицы задают **количество столбцов**, **количество строк**, **шаг между элементами** в строке и **дополнительный шаг** между последним элементом строки и первым элементом следующей строки. Номера элементов вектора рассчитываются начиная от начального индекса источника. Как и начальный индекс источника, параметры матрицы могут быть заданы по ссылке.

Для списка поле параметров имеет вид, представленный на рис.12.

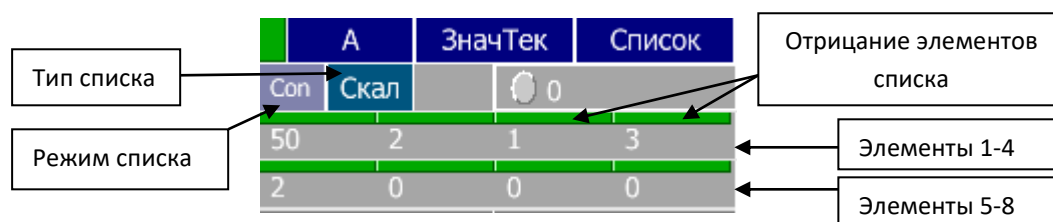


Рис.12 – Параметры списка

Режим предусматривает следующие варианты:

- задание константами;
- задание ссылками на БД.

Тип списка предусматривает следующие варианты

- список скаляров (образует вектор - обрабатывается как вектор);
- список векторов (образует матрицу - обрабатывается как матрица);
- список списков (образует матрицу - обрабатывается как матрица).

Использование элементов списка разного типа в различных режимах определяется в соответствии с таблицей:

Тип списка	Режим задания	
	Константы	Ссылки
Список скаляров (вектор)	Sp(1..8) – список, заданный константами. Размер вектора – до 8 элементов.	Sp(1) – ссылка на начало списка; Sp(2) – количество элементов списка (размер вектора)
Список векторов (матрица)	Sp(1..6) – список, заданный константами (максимум 6 элементов) – размер вектора; Sp(7) – количество векторов (размер матрицы); Sp(8) – смещение между векторами матрицы; Образуется матрица 6XSp(7)	Sp(1) – ссылка на начало списка (элемент матрицы (1,1)); Sp(2)-количество элементов списка (размер вектора); Sp(3)-количество векторов (размер матрицы); Sp(4)-смещение между векторами матрицы; Образуется матрица Sp(2)XSp(3)

Список списков (матрица)	<p>Sp(1..4) – список, заданный константами (максимум 4 элемента) - размер вектора;</p> <p>Sp(5..8) – смещения между векторами матрицы, отсчитанные от 1-го вектора;</p> <p>Образуется матрица 4X5</p>	<p>Sp(1) – ссылка на начало списка (элемент матрицы (1,1));</p> <p>Sp(2) – количество элементов списка (размер вектора);</p> <p>Sp(3) – количество векторов (размер матрицы);</p> <p>Sp(4) – ссылка на список смещений векторов матрицы, отсчитываемых от 1-го вектора матрицы (размер списка равен sp(3)-1);</p> <p>Образуется матрица Sp(2)XSp(3)</p>
--------------------------	---	---

Замечания

1. Списки, заданные константами:

- имеют индивидуальные флаги отрицания;
- элементы списка, меньшие или равные нулю и не входящие в область нормальных номеров (за ними нет нормальных номеров) отбрасываются.

2. Списки, заданные ссылками:

- если элемент списка задан отрицательным числом, то считается, что он имеет отрицание - для выборки элемента используется абсолютное значение;
- никаких других проверок не производится - нули не отбрасываются.

3. Индивидуальное отрицание элемента списка и общее отрицание всего операнда объединяются по правилу «Исключительное ИЛИ», таким образом два отрицания отменяют отрицание.

Отрицание элементов производится после чтения или перед записью по следующим правилам:

- для чисел производится умножение на -1;
- для логических элементов выполняется функция NOT;
- для символов:
 - если код символа меньше 128, то к нему прибавляется 128;
 - если код символа больше 127, то от него вычитается 128.

5.6. Секция выборки приведена на рис.13.

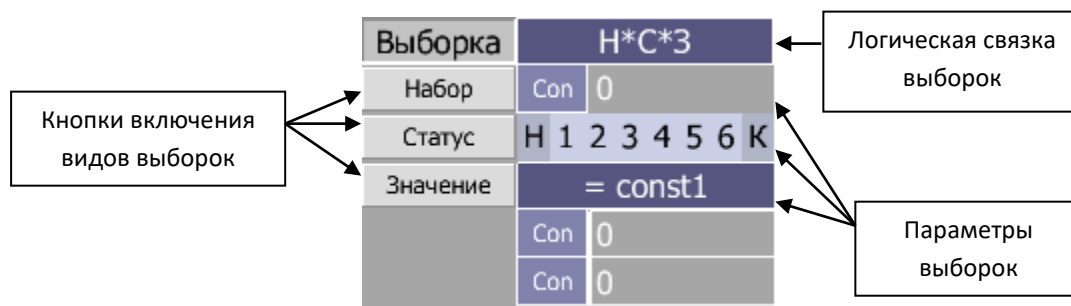


Рис.13 – Секция выборки операндов

Операция выборки операндов предназначена для фильтрации операндов по определенным критериям. Операнды, не прошедшие выборку, не участвуют в операции. Выборка может быть организована по следующим критериям:

- по номеру набора;
- по статусу параметра;
- по значению параметра.

Выборка **по номеру набора** возможна только для БД MWBridge. Параметр считается удовлетворяющим данному критерию выборки, если номер набора параметра равен заданному.

Выборка **по статусу** возможна только для БД MWBridge и вектора RR. Для организации выборки используется статусный байт, в котором побитно (6 битов анализа уставок, бит достоверности и бит квитирования) указывается:

- Использование данного бита (установка бита в 1);
- Использование данного бита (сброс бита в 0);
- Отказ от анализа данного бита.

Выборка считается пройденной, если все биты, для которых определена проверка, имеют указанные значения.

Выборка **по значению** возможна для всех БД. Параметр считается удовлетворяющим данному критерию выборки, если текущее значение параметра удовлетворяет заданному условию проверки. Возможны следующие проверки значения:

- равно константе;
- не равно константе;
- больше константы;
- меньше константы;
- находится в диапазоне <константа1 - константа2>;
- находится вне диапазона <константа1 - константа2>.

Результаты выборки по указанным выше критериям **объединяются в единый результат логическими операциями** «И» и «ИЛИ», что дает возможность составлять сложные логические конструкции для выборки операндов.

Выборка возможна для следующих операций:

- общие:
 - копирование;
- арифметические:
 - все бинарные операции;
 - все унарные операции;
 - взвешенное сложение и умножение, а также средневзвешенное;
 - попадание в диапазон;
 - прогрессия;
- логические:
 - отрицание (NOT);
 - И, ИЛИ, Исключительное ИЛИ, НЕ И, НЕ ИЛИ, НЕ Исключительное ИЛИ, Следование, НЕ Следование,
 - сравнение с уставками;
 - сравнение значений;

- попадание в диапазон;
- динамика:
 - идентификация модели.

Включение выборки производится следующим образом:

- установить общий флаг включения выборки;
- включить необходимые выборки (по набору, статусу и значению);
- для каждой включенной выборки установить требуемые значения для проверки;
- определить логическую связку выборок (если используется более одной выборки).

5.7. Секция проверки приведена на рис.14.

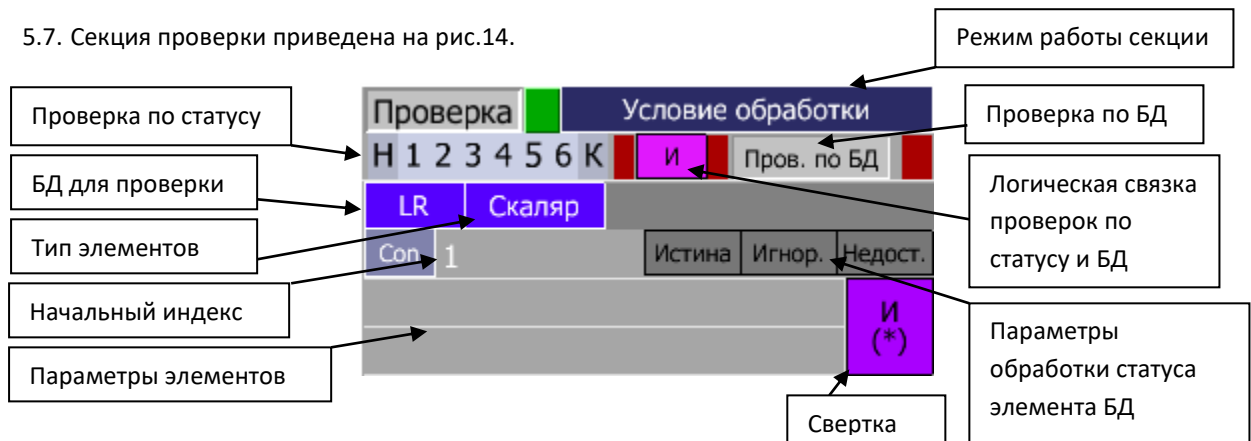


Рис.14 – Секция проверки

Перед использованием операндов в операции может быть произведена проверка операндов. Проверка данных на достоверность производится после проведения выборки данных (если выборка определена). Возможны два типа проверок, **объединяемые логически** по «И»/«ИЛИ»:

- проверка по статусу;
- проверка по базе данных.

Проверка по статусу предназначена для данных, имеющих статусные биты состояния сигнала: БД MWBridge, а также расчетная БД Rsprog RR, если в ней сформированы статусные биты.

Для БД Rsprog RR возможно включение проверок следующих состояний:

- норма (исключаются значения со статусами 1,2,5,6,7);
- за верхней уставкой (исключаются значения со статусом 3);
- за нижней уставкой (исключаются значения со статусом 4);
- любые комбинации из них, объединенные логической комбинацией «ИЛИ» (в т.ч. наиболее распространенный вариант проверки - проверка только на «Норму» - в этом случае достоверными будут считаться значениями только со статусами 0,3 или 4).

Для БД MWBridge проверки включаются побитно:

- бит «Н» - проверка на достоверность;
- бит «К» - бит квитирования;
- 6 битов уставок (в случае достоверности – бит «Н» не взведен)/типов недостоверности(в случае недостоверности – бит «Н» взведен).

Проверка по БД сводится к анализу состояния указанного элемента БД (скаляра, вектора или матрицы) - элемент проверяется на значение Истина/Ложь в зависимости от указанных флагов проверки. Приведение различных типов данных к логическому типу осуществляется следующим образом:

- для числовых данных (любые вещественные и целые):
 значение, равное нулю - Ложь;
 значение, неравное нулю - Истина;
- для символьных данных используется код символа, далее обработка как для целых чисел.

Если для проверки по БД указан не скалярный тип элемента, то производится **свертка** результатов проверки индивидуальных элементов.

Свертка по флагу "И": результирующий свернутый операнд объявляется недостоверным, если хотя бы одно из сворачиваемых данных было недостоверным.

Свертка по флагу "ИЛИ": результирующий свернутый операнд объявляется недостоверным, если все сворачиваемые данные были недостоверными.

5.8. Вместо проверки по БД можно использовать БД, указанную в секции проверки, как расширение операнда. Расширение БД операнда позволяет использовать в операциях дополнительные БД. Расширение включается с помощью выпадающего меню **Режим работы секции**. При этом для различных операций название пункта меню, определяющего режим расширения операнда, может отличаться.

Правила расширения:

- расширение производится простым наращением размерностей:

Структура операнда	Структура расширения	Рез. структура	Примечание
1. Скаляр	Скаляр	Вектор (2)	
	Вектор (n)	Вектор (1+n)	
	Матрица (n,m)	Вектор (1+n)	
2. Вектор (l)	Скаляр	Вектор (l+1)	
	Вектор (n)	Вектор (l+n)	
	Матрица (n,m)	Вектор (l+n)	
3. Матрица (l,k)	Скаляр	Матрица (l,k)	Нет расширения
	Вектор (n)	Матрица (l,k)	Нет расширения
	Матрица (n,m)	Матрица (l+n,k)	Есть только если $m \geq k$

- при расширении для дополнительных БД используется тот же тип значений, что и для основного операнда; если такого типа значений для дополнительной БД не существует, то используется текущее значение.

5.9. Поле операции состоит из двух секций:

- тип операции;
- условия выполнения операции.

5.10. Секция типа операции представлена на рис.15.

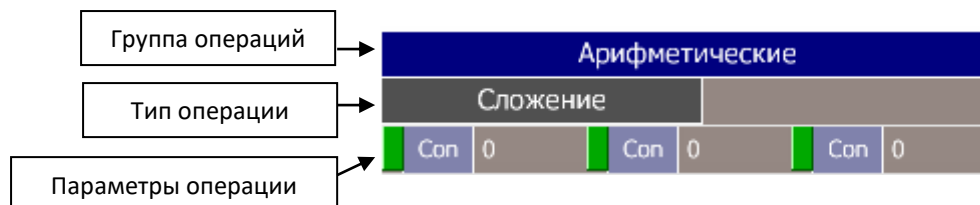


Рис.15 – Тип и параметры операции

Все типы операций для удобства разделены на группы согласно их сути. Для выбора выполняемой в алгоблоке операции необходимо в выпадающих меню указать **группу и тип операции**.

Для большинства операций есть возможность указать до трех дополнительных **параметров**, которые могут оказывать влияние на выполнение операции.

Полный перечень поддерживаемых операций и их параметров приведен в **Приложении А**.

5.11. Секция условий выполнения операции приведена на рис.16.

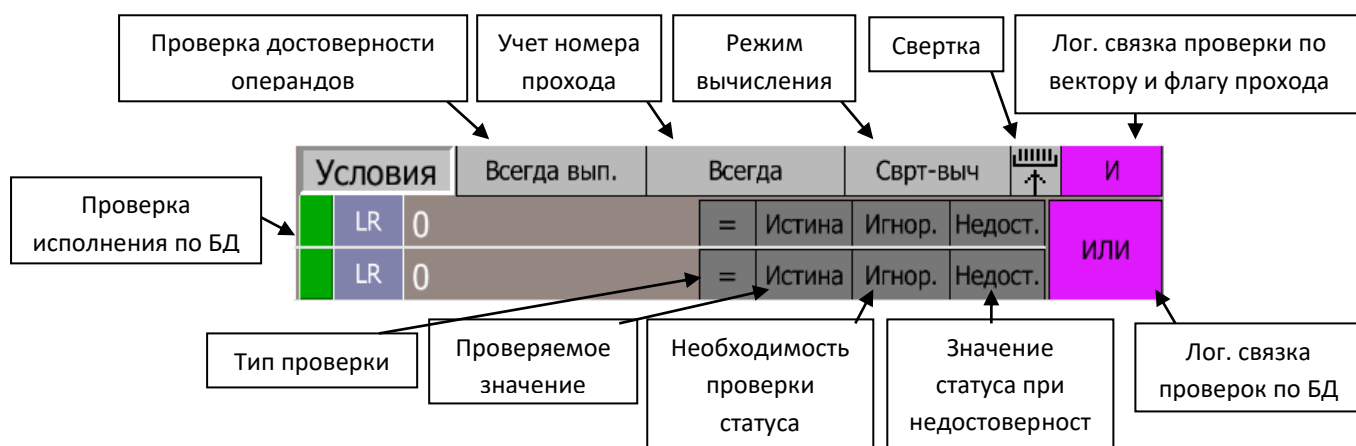


Рис.16 – Условия выполнения операции

Для каждой операции оператор может задать режимы использования результатов **проверки операндов на достоверность**. Для операции определен флаг задания проверки, имеющий следующие значения:

- НПР (не проверять операнды на достоверность);
- Люб (хотя бы один из операндов д.б. достоверен);
- Оба (проверять оба операнда);

а также флаг выполнимости операции в зависимости от результатов проверки:

- Вып (операция выполняется);
- Нет (операция не выполняется).

Если требуется, производится свертка операндов по «И» или «ИЛИ»

Учет недостоверных данных при свертке производится следующим образом:

Проверка операндов	Свертка по «И»	Свертка по «ИЛИ»
НЕ проверять	Используется константа операнда	Операнд пропускается
Проверять	Используется сам операнд	Операнд пропускается

НПР (не проверять)		ЛЮБОЙ / ОБА					
		Выполнять операцию			НЕ выполнять операцию		
1 операнд	2 операнда	1 операнд	2 операнда		1 операнд	2 операнда	
			ЛЮБОЙ	ОБА		ЛЮБОЙ	ОБА
ОП ДСТ – 0	Оба ОП ДСТ – 0	ОП ДСТ – 0	Оба ОП ДСТ – 0	Оба ОП ДСТ – 0	ОП ДСТ – 0	Оба ОП ДСТ – 0	Оба ОП ДСТ – 0
ОП НЕДСТ – 1	Один ОП НЕДСТ – 2	ОП НЕДСТ – 4	Один ОП НЕДСТ – 5	Один ОП НЕДСТ – 7	ОП НЕДСТ – 9	Один ОП НЕДСТ – 10	Один ОП НЕДСТ – 12
	Оба ОП НЕДСТ – 3		Оба ОП НЕДСТ – 6	Оба ОП НЕДСТ – 8		Оба ОП НЕДСТ – 11	Оба ОП НЕДСТ – 13

где ОП ДСТ – Операнд достоверен; ОП НЕДСТ – Операнд недостоверен.

Случай	Действие	Записываемый статус
0	Тривиальный случай: операция выполняется над операндами	0
1	Вместо НЕДСТ операнда используется КОНСТАНТА ОПЕРАНДА	0
2	Если операнд сворачивался по "И": КОНСТАНТА ОПЕРАНДА "ИЛИ": операнд НЕ УЧИТЫВАЕТСЯ	0
3	=== 1	
4	Используется Сам Операнд	5
5	НЕДСТ операнд НЕ УЧИТЫВАЕТСЯ	0
6	=== 4	
7	=== 4	
8	=== 4	
9	Операция НЕ ВЫПОЛНЯЕТСЯ: используется КОНСТАНТА РЕЗУЛЬТАТА (если разрешена); Константа пишется со своим СТАТУСОМ, флаг записи статуса игнорируется. Но, если задана запись статуса, то статус константы будет затерт статусом операции	5
10	=== 5	
11	=== 9	
12	=== 9	
13	=== 9	

Если операция не может быть выполнена (деление на 0 и т.д.), то в результат пишется константа результата (если разрешено) и статус 5 (если разрешена запись статуса).

Если в результате выполнения свертки по флагу "ИЛИ" (с пропуском недостоверных операндов) нет ни одного достоверного данного, то такой операнд считается отсутствующим.

Если обнаружено отсутствие хотя бы одного операнда, то:

- при значениях флагов Люб или Оба и флаге выполнимости операции Да в результат записывается константа результата;
- при всех остальных комбинациях флагов результат не изменяется.

5.12. Поле результата состоит из двух секций:

- результат;
- обработка.

Секция результата аналогична секции источника данных операнда и определяет то, куда будет записан результат выполнения операции.

Секция обработки приведена на рис.17.

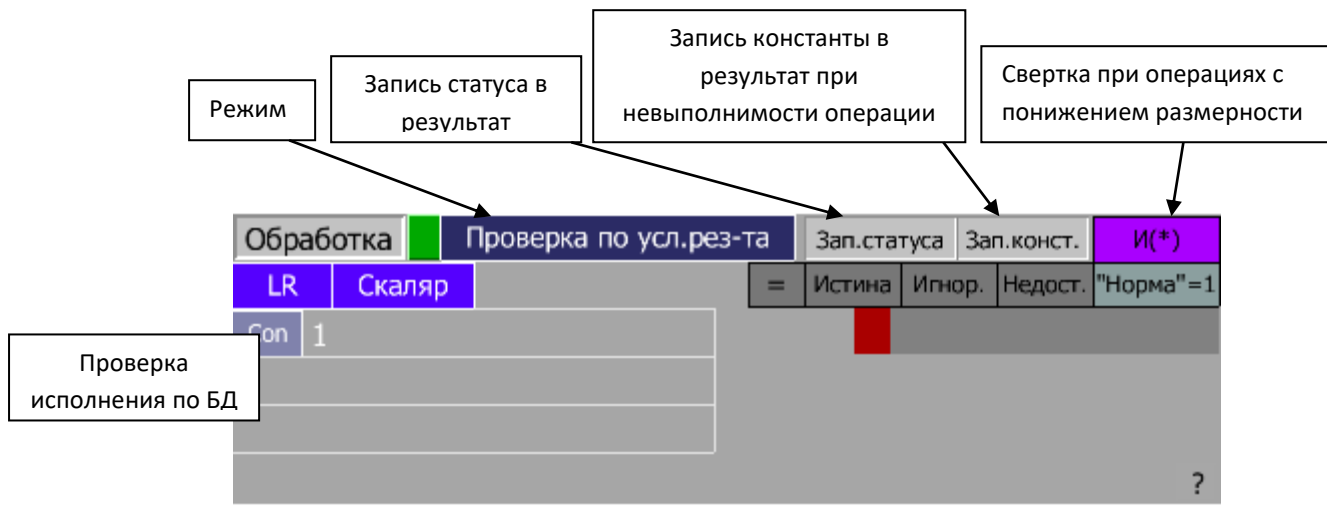


Рис.17 – Логическая обработка результата

Проверка необходимости исполнения алгоблока может быть произведена непосредственно на этапе расчета. Для этого используются условия результата.

Для включения проверки необходимо:

- установить режим проверки по условиям результата;
- определить значение для проверки (Истина, Ложь, 0 ... 13);
- установить флаг проверки статуса используемых для проверки данных;
- установить результат проверки по умолчанию (Дост, Недост), используемый при недостоверных данных или неверной адресации условий проверки.

Заданная таким образом проверка позволяет выполнять операции индивидуально. Например, при операции сложения векторов с записью в вектор результата можно определить вектор условий результата, разрешающий исполнение сложения только для избранных элементов векторов (для тех элементов векторов результата, для которых соответствующие элементы вектора условий результата имеют значение, равное проверяемому).


Проверка по условиям результата применима для следующих операций:

- общие операции:
 - копирование;
- все арифметические операции (кроме прогрессии, решения системы линейных уравнений, а также определения коэффициентов полинома);
- все логические операции, кроме апериодического фильтра и отклонения;
- операции динамики:
 - ограничение;

- ШИМ;
- таймеры:
 - циклический;
 - астрономический.

5.13. Вместо проверки по БД можно использовать БД, указанную в секции обработки, как расширение результата аналогично расширению БД операнда. Расширение включается с помощью выпадающего меню **Режим**. При этом для различных операций название пункта меню, определяющего режим расширения операнда, может отличаться.


6. ПРОСМОТР ЗНАЧЕНИЙ

6.1. Форма просмотра значений предоставляет удобный интерфейс для просмотра значения и статуса некоторого набора сигналов из разных БД на одном экране. Окно просмотра значений запускается по кнопке  из главного окна и приведено на рис.18.

[illegible]

Рис.18 – Просмотр значений

Добавление сигнала в таблицу для просмотра значения и статуса осуществляется путем нажатия на кнопку **+** на необходимой строке таблицы. В появившемся диалоге следует выбрать необходимую БД и сигнал, после чего нажать кнопку ОК.

Для удаления сигнала из таблицы следует нажать кнопку  на необходимой строке таблицы. В появившемся диалоге следует снять флаг Активно и нажать кнопку ОК.

7. ПРОСМОТР И ИНИЦИАЛИЗАЦИЯ БАЗ

7.1. В нижней части главного окна программы располагается ряд кнопок с названиями БД системы, представленный на рис.19.



Рис.19 – Кнопки просмотра баз

При нажатии кнопки открывается окно с перечнем сигналов выбранной базы данных, пример которого представлен на рис.20.

Массив RR					>>	Количество эл-тов	10000	Записать
№	Название	Описание	Нач.знач.	Тек.знач.				
RR_1	RR_000001	RR элемент [00001]	0	0				
RR_2	RR_000002	RR элемент [00002]	0	0				
RR_3	RR_000003	RR элемент [00003]	0	0				
RR_4	RR_000004	RR элемент [00004]	0	0				
RR_5	RR_000005	RR элемент [00005]	0	0				
RR_6	RR_000006	RR элемент [00006]	0	0				
RR_7	RR_000007	RR элемент [00007]	0	0				
RR_8	RR_000008	RR элемент [00008]	0	0				
RR_9	RR_000009	RR элемент [00009]	0	0				
RR_10	RR_000010	RR элемент [00010]	0	0				
RR_11	RR_000011	RR элемент [00011]	0	0				
RR_12	RR_000012	RR элемент [00012]	0	0				
RR_13	RR_000013	RR элемент [00013]	0	0				
RR_14	RR_000014	RR элемент [00014]	0	0				
RR_15	RR_000015	RR элемент [00015]	0	0				
RR_16	RR_000016	RR элемент [00016]	0	0				
RR_17	RR_000017	RR элемент [00017]	0	0				
RR_18	RR_000018	RR элемент [00018]	0	0				
RR_19	RR_000019	RR элемент [00019]	0	0				
RR_20	RR_000020	RR элемент [00020]	0	0				


Рис.20 – Окно просмотра БД

Редактированию в таблице сигналов подлежат:

- название сигнала;
- описание сигнала;
- начальное (инициализационное) значение;
- текущее значение (только при загруженной системе исполнения алгоблоков).

Для баз LR, RR, IR, IR2, IR4 также существует возможность установить размер БД.

Таблица сигналов поддерживает операции по работе с буфером обмена: копирование (Ctrl+C), вставку (Ctrl+V). Операции осуществляются с выделенным диапазоном ячеек таблицы. Выбор ячеек осуществляется мышью, удерживая левую кнопку.

7.2. Кнопка  открывает дополнительное окно для текущей базы данных, в котором содержатся данные **анализа использования элементов БД**. Окно анализа представлено на рис.21.

База А		Использование						Изменение					
№	Всего	№	прц	алг			Всего	№	прц	алг			
A1	0					< >	0					< >	
A2	0					< >	0					< >	
A3	0					< >	0					< >	
A4	0					< >	0					< >	
A5	0					< >	0					< >	
A6	0					< >	0					< >	
A7	0					< >	0					< >	
A8	0					< >	0					< >	
A9	0					< >	0					< >	
A10	0					< >	0					< >	
A11	0					< >	0					< >	
A12	0					< >	0					< >	
A13	0					< >	0					< >	
A14	0					< >	0					< >	
A15	0					< >	0					< >	
A16	0					< >	0					< >	
A17	0					< >	0					< >	
A18	0					< >	0					< >	
A19	0					< >	0					< >	
A20	0					< >	0					< >	

Рис.21 – Анализ использования элементов БД


Блок столбцов **использование** позволяет определить те места в программе (номер процедуры, алгоблока и область в алгоблоке), где данный элемент используется, или как данное, или как флаг, или как ссылочный массив.

Блок столбцов **изменение** позволяет определить те места в программе (номер процедуры, алгоблока и область в алгоблоке), где данный элемент присваивается, т.е. меняет значение.

В обоих случаях использование и присвоение может производиться через ссылку, т.е. зависеть от текущего значения других элементов.

Анализ ссылочной информации возможен только при загруженной исполняющей системе.

8. РЕДАКТОР БАЗ

8.1. Редактор баз предназначен для автоматизации редактирования баз данных, выполнения групповых операций над параметрами сигналов базы. Для запуска редактора баз следует нажать кнопку  на главном окне программы. Окно редактора баз представлено на рис.22.

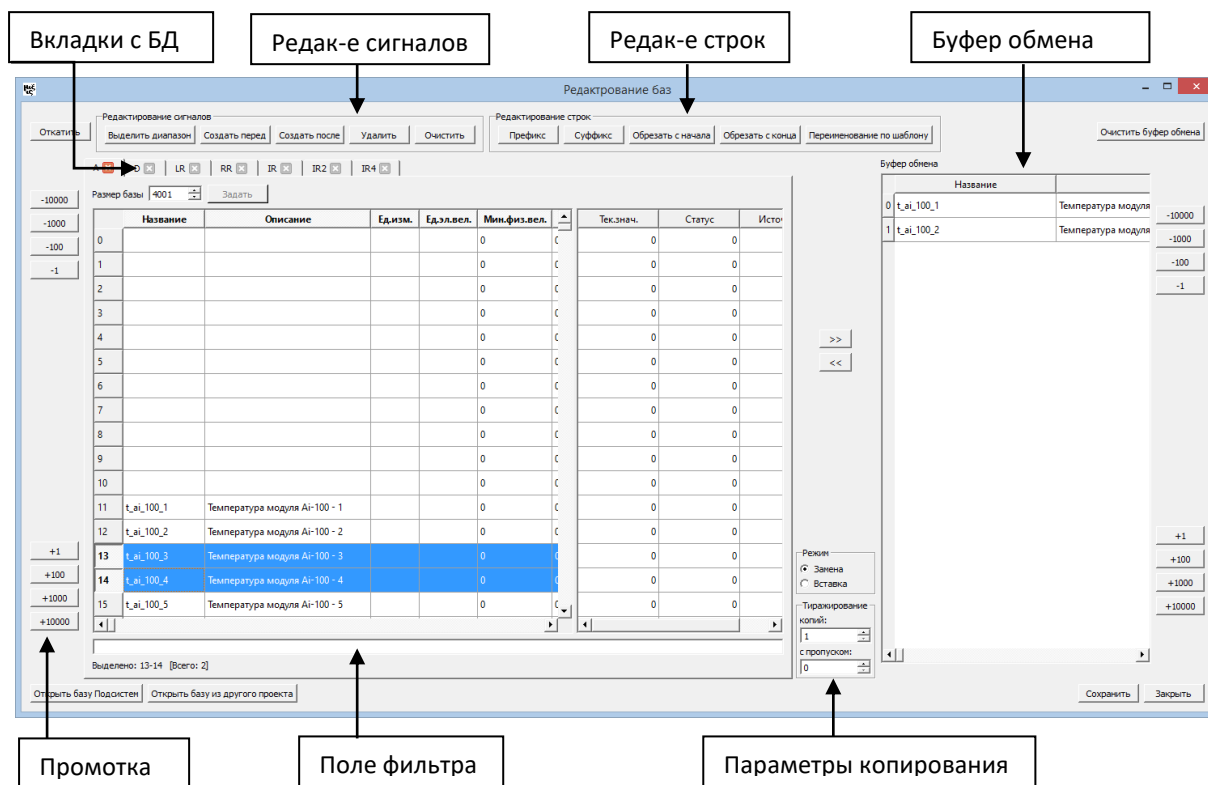



Рис.22 - Редактор баз

В левой части окна редактора баз располагаются **вкладки**, каждая из которых представляет одну базу данных. На каждой вкладке находится таблица с перечнем сигналов базы. Для баз аналогов (А) и дискретов (D) также присутствуют дополнительные столбцы, отображающие текущее значение, статус и источник для каждого сигнала (при работающем MWBridge). Помимо собственных баз проекта есть возможность работы со сторонними БД (подсистемы или из другого проекта). Сторонние БД подгружаются по кнопкам **Открыть базу Подсистем** или **Открыть базу из другого проекта**, в виде дополнительных вкладок. После работы со сторонними базами, они могут быть закрыты путем нажатия кнопки  на соответствующей вкладке.

Кнопки промотки **-10000, -1000, -100, -1, +1, +100, +1000, +10000** позволяют осуществлять перелистывание таблицы сигналов на заданное число строк.

Поле фильтра предназначено для быстрого поиска сигналов базы по подстроке, содержащейся в названии или описании. Для этого необходимо ввести в поле искомую строку (подстроку), после чего в таблице сигналов будут отображены только те сигналы, название или описание которых содержит указанную строку (без учёта регистра букв).

В верхней части вкладки с БД располагается поле ввода, позволяющее **задать размер БД**. Для изменения размера БД следует указать количество сигналов и нажать кнопку **Задать**. Не допускается изменять размер БД при работающем MWBridge.

8.2. Индивидуальное редактирование существующих сигналов БД осуществляется напрямую в таблице сигналов. Для редактирования доступны все отображаемые поля сигнала.

Секция **Редактирование сигналов** содержит кнопки, предназначенные для группового редактирования сигналов, а также изменения их состава.

Кнопка **Выделить диапазон** позволяет осуществить выделение сигналов в таблице путём явного указания номеров сигналов начала и конца диапазона. Опционально выделяемый диапазон сигналов может быть добавлен к уже имеющемуся выделению, позволяя таким образом набирать произвольное подмножество сигналов из общего списка для последующего редактирования.

Кнопки **Создать перед** и **Создать после** позволяют вставить указанное количество пустых сигналов перед первым сигналом или после последнего сигнала выделенного множества сигналов соответственно, со сдвигом остальных сигналов.

Кнопка **Удалить** осуществляет удаление выделенных сигналов, со сдвигом остальных сигналов.

Кнопка **Очистить** осуществляет удаление выделенных сигналов, без сдвига остальных сигналов (т.е. замену удаляемых сигналов пустыми сигналами).

Секция **Редактирование строк** содержит кнопки, предназначенные для группового редактирования строковых полей сигналов (названия, описания). Редактированию подвергаются только выделенные поля в таблице сигналов.

Кнопки **Префикс** и **Суффикс** позволяют добавить указанную строку в начало и конец каждой строки из выделенного множества соответственно.

Кнопки **Обрезать с начала** и **Обрезать с конца** позволяют удалить указанное количество символов в начале и конце каждой строки выделенного множества соответственно.

Кнопка **Переименование по шаблону** позволяет осуществить преобразование выбранных строк в соответствии с шаблоном специального формата. Окно переименования представлено на рис.23.

Шаблон

[N] - исходная строка
[x-y] - диапазон символов исходной строки (x - начальный символ, y - конечный символ)
[Cx,y,z] - счетчик (x - число цифр, y - начальное число, z - шаг)

Найти: 100 -
Занести на: 100-

	1	2	3	4	5	
1	t_ai_100_3	Температура модуля Ai-100 - 3			0	0
2	t_ai_100_4	Температура модуля Ai-100 - 4			0	0

	1	2	3	4	5	
1	t_ai_100_3	Температура модуля Ai-100-3			0	0
2	t_ai_100_4	Температура модуля Ai-100-4			0	0

OK Отмена

Рис.23 – Переименование по шаблону

В качестве шаблона используется строка, в которой могут применяться специальные маркеры:

[N] – маркер заменяется на исходную строку;


[Nx-y] – маркер позволяет скопировать в результат диапазон символов исходной строки (подстроку), начиная с символа **x**, оканчивая символом **y**. Например [N3-5] копирует в результат 3,4 и 5 символы исходной строки.


[Cx,y,z] – маркер позволяет вставить в результат числовое значение счётчика с количеством цифр **x**, начальным числом **y** и шагом **z**.

Пример использования

Исходные строки	Шаблон: [N1-14]камере [C2,1,1]
Температура в печи 1	Температура в камере 01
Температура в печи 2	Температура в камере 02
Температура в печи 3	Температура в камере 03
Температура в печи 4	Температура в камере 04
Температура в печи 5	Температура в камере 05
Температура в печи 6	Температура в камере 06

- 8.3. В правой части окна редактора баз располагается буфер обмена. **Буфер обмена** представляет собой виртуальную базу данных и предназначен для временного хранения отдельных сигналов, осуществления операций редактирования над ними и последующей вставки подготовленных сигналов в целевую БД.

Для того, чтобы поместить (скопировать) сигналы в буфер обмена, следует выделить необходимые сигналы в таблице сигналов исходной БД и нажать кнопку . Копируемые сигналы будут помещены сразу за последней выделенной строкой в таблице сигналов буфера обмена.

Для того, чтобы поместить сигналы из буфера обмена в целевую БД, следует выделить необходимые сигналы в буфере обмена, выделить опорный сигнал в целевой БД, задать параметры копирования и нажать кнопку . Сигналы будут помещены в целевую БД согласно параметрам копирования.

Параметры копирования включают в себя следующие опции:

– **Режим** копирования:

Замена – копируемые сигналы заменяют существующие сигналы в целевой БД, начиная с опорного;

Вставка – копируемые сигналы вставляются сразу за последним выделенным сигналом в целевой БД, со сдвигом остальных сигналов базы.

– **Количество копий** – перемещаемые сигналы будут продублированы и скопированы в соответствии с указанным режимом заданное количество раз.

– **С пропуском** – параметр позволяет задавать смещение между копиями перемещаемых сигналов в целевой БД. Если используется режим замены, то пропускаемые сигналы остаются без изменений. Если используется режим вставки, то между копиями перемещаемых сигналов создаются пустые сигналы в количестве указанного смещения.

Так как буфер обмена является виртуальной БД, то при работе с сигналами в буфере обмена доступны операции редактирования сигналов и редактирования строк, аналогично работе с сигналами БД.

9. ГРУППЫ УЧЕТА

9.1. Программа позволяет управлять двумя типами групп:

- учетные группы;
- группы расчета расходов и тепла.

Поле **Группы учета** в главном окне предназначено для работы с учетными группами и представлено на рис.24.



Рис.24

Поле **Режим обсчета групп учета** позволяет установить способ обработки групп учета:

- перед исполнением программы;
- после исполнения программы;
- в теле самой программы с помощью алгоблока «Группы учета» (в этом случае возможно задание номеров обсчитываемых групп; в остальных случаях всегда обсчитываются все группы).

9.2. При нажатии кнопки **Y** в поле групп учета главного окна открывается окно редактора групп учета, представленное на рис.25.

№	Знак	Множитель	Сигнал БД	Тек.знач.
1	+	1	A11; t_ai_100_1; Температура модуля Ai-100 - 1	0
2	+	1	A12; t_ai_100_2; Температура модуля Ai-100 - 2	0
3	+	1	A15; t_ai_100_5; Температура модуля Ai-100 - 5	0
4	+	1		
5	+	1		
6	+	1		
7	+	1		

№	Название	Сигнал ошибки	Время посл. измер.	Матер. суж. ус-ва	Матер. изм. тр-да	Экв. шерох.	Экв. шерох.
1	Группа N 1	Среда	природный газ	0.00			
2	Группа N 2	Расход	объемный, м3/час		8		
3	Группа N 3	Суж. у-во	диафрагма		8		
4	Группа N 4	ØИз. Тр., мм	0.00	ØСУ, мм	0.00	CO2	0.000
5	Группа N 5	Способ отбора	угловой	N2	0.00	R кромки	0.00
6	Группа N 6	№	1	Температура среды, град.С			
7	Группа N 7	№	2	Давл. среды избыточное, кгс/см2			
8	Группа N 8	№	3	Перепад давл. на суж. у-ве, кгс/см2			
9	Группа N 9	№	4	Давл. атмосферное, мм.рт.ст.			
10	Группа N 10	Расход среды объемный, м3/час					
11	Группа N 11	Количество тепла, кДж/ч					

Рис.25 – Редактирование групп учета

9.3. В верхней части окна осуществляется редактирование учетных групп. Данный тип групп позволяет вычислять следующее математическое выражение:

$$R=K1*P1+K2*P2+...+Kn*Pn$$

где:

- R - результат вычислений;
- K1..Kn - коэффициенты учета параметров;
- P1..Pn - значения учитываемых параметров;
- N - количество параметров в группе.

Слева располагается список учетных групп. При выборе в списке конкретной группы справа отображаются параметры выбранной группы.

Таблица **сигналы** служит для редактирования слагаемых вычисляемого выражения, и позволяет задать знак, множитель (коэффициент) и сигнал БД для каждого из параметров выражения. Панель инструментов позволяет добавлять или удалять параметры в выражение.

В поле **Результат** необходимо указать сигнал БД для записи результата вычисления.

Кнопка **Генерация** осуществляет сохранение настроек и их загрузку в исполняющую систему.

9.4. В нижней части окна осуществляется редактирование групп расчетов расходов и тепла. Слева располагается список групп. При выборе в списке конкретной группы справа отображаются параметры выбранной группы.

Данный тип групп позволяет вычислять расходы и тепло при различных настроечных параметрах:

- среда;
- единицы измерения расхода (объемный в м3/час или массовый в т/час);
- тип сужающего устройства;
- диаметры сужающего устройства (СУ) и измерительного трубопровода (ИТ) (в мм);
- способ отбора (для диафрагмы);
- время с момента последнего измерения (в годах);
- материалы сужающего устройства и измерительного трубопровода;
- эквивалентная шероховатость (Ra);
- радиус закругления кромки сужающего устройства (в мм);
- молярные доли CO2 и N2 (для природного газа).

Для расчета обязательно должны быть заданы сигналы, определяющие входные параметры:

- температура среды (град.С);
- давление среды избыточное (кгс/см2);
- перепад давлений на СУ (кгс/см2);
- давление атмосферное (мм.рт.ст.);

а также результат расчета (расход объемный или массовый).

Опционально может быть задан сигнал для расчета теплоты и вспомогательный сигнал ошибки.

Сигнал ошибки принимает следующие значения:

= 0 - норма;

В остальных случаях (значение меньше 0) при данном значении определенных параметров расчет невозможен (результатирующее значение равно 0 с недоверенным статусом):

- = -1 - неверные значения параметров (диаметр СУ при рабочей температуре Dсу, диаметр ИТ при рабочей температуре Дит, относительный диаметр СУ);
- = -2 - отношение dp/p (относительное изменения давления) > 0.25;
- = -3 – невозможно определить коэффициент шероховатости при данном значении числа Рейнольдса;

- = -4 - неверное отношение $Ra/D_{ит}$;
- = -5 – неверное значение число Рейнольдса;
- = -6 – неверно задано атмосферное давление;
- = -100 – не все входные данные достоверны;
- = -101 - не определяется фактор сжимаемости природного газа.

Более подробно см. ГОСТ 8.586.1-5.

Примечание. Реализованы расчеты только для природного газа, водяного пара и воды.

Кнопка **Генерация** осуществляет сохранение настроек и их загрузку в исполняющую систему.

Все операции подразделяются на следующие классы:

- Общие;
- Управление;
- Арифметические;
- Логические;
- РЕЗЕРВ;
- Динамика;
- Таймеры;
- Регуляторы;
- Резервирование;
- Теплофизика;
- Специализированные;
- Учет и ТЭП;
- РЕЗЕРВ;
- РЕЗЕРВ;
- РЕЗЕРВ;
- Циклы.

A.1 Общие операции

1. Копирование
2. Загрузка БД НСИ
3. Сохранение БД НСИ

A.1.1 Копирование

Копирование осуществляется по типу результата (операнды сразу вычитываются с преобразованием к типу результата). Копирование производится из 1-го операнда в результат. Операнды не проверяются на достоверность, а копируются, как есть вместе со статусом.

Параметр 1

- = 0 - копирование из 1-го операнда с тиражированием результата (если возможно);
- = 1 - копирование из 1-го операнда с использованием 2-го операнда для дозаписи в свободное место результата (если оно есть);
- = 10, 11 – аналогично случаям 0 и 1 за исключением того, что запись в результат производится с блокированием записи из внешних источников.

Параметр 2

- = 0 - обычное копирование;
- = 10 - копирование с промежуточной буферизацией (возможность копирования в ту же область данных - Перестановка).

Для символьных источников и приемников данных возможно задание перекодировки:

- для обычного копирования:

- 1 - перекодировка Win->DOS;
- 2 - перекодировка Win->Linux;
- 3 - перекодировка DOS->Win;
- 4 - перекодировка DOS->Linux;
- 5 - перекодировка Linux->Win;
- 6 - перекодировка Linux->DOS;

- для копирования с буферизацией:

- 11 - перекодировка Win->DOS;
- 12 - перекодировка Win->Linux;
- 13 - перекодировка DOS->Win;
- 14 - перекодировка DOS->Linux;
- 15 - перекодировка Linux->Win;
- 16 - перекодировка Linux->DOS.

Копирование при значении параметра 2 >= 10 позволяет осуществлять копирование в ту же область данных, что и у операндов - то есть производить перестановку элементов.

Параметр 3 - уплотнение при записи в результат:

- = 0 - без уплотнения;
- = 1 - с уплотнением пропущенных в результате элементов (пропуски могут возникать при использовании функции выборки операндов);

Проверка по условиям результата

Если включена проверка по условиям результата, то:

- копирование производится только в те элементы результата, для которых проверка пройдена;

- в случае тиражирования результата (скаляр - вектор, матрица или вектор - матрица):
 - если запись в основной (тиражируемый элемент) возможна, то тиражирование производится только в те элементы результата, для которых проверка пройдена;
 - если запись в основной (тиражируемый элемент) не производится, то тиражирования также не производится.

Запись в условия результата

Если разрешена запись в условия результата, то:

- если размерности приемника условий результата больше или равны размерностям приемника самого результата,
 - то: для всех записанных в результат значений в соответствующих элементах условия результата записывается «1»; кроме этого в 1-х элементах векторов условий результата записывается суммарное количество записанных в этот вектор значений;
 - иначе:
 - если размерность вектора условий результата больше или равна количеству векторов в самом результате (2-я размерность результата),
 - то: для каждого записанного вектора результата в условия результата записывается суммарное количество записанных в этот вектор значений;
 - иначе: в условия результата (1-й элемент) записывается общее количество записанных в результат значений.

Для данной операции возможно указание **выборки данных**.

A.1.2 Загрузка и сохранение БД НСИ

Данные алгоблоки используются для загрузки и сохранения внешних БД НСИ, созданных пользователем.

Параметры операции не используются.

Операнды содержат имена внешних БД НСИ; имена указываются без расширения; БД НСИ должны находиться в текущем каталоге.

Операнды используются только в том случае, если:

- указывают на символьную БД (SR);
- имеют размерность не ниже вектора; матрица рассматривается как группа строк, каждая из которых определяют свою БД.

Длина имени файла определяется:

- или размерностью вектора;
- или ограничивается байтом с кодом, меньшим или равным коду "пробела".

Максимальное количество внешних БД - 10.

Используйте загрузку БД под флагом 1-го прохода !

Используйте сохранение БД под флагом последнего прохода !

А.1.3 Чтение из трендов

Алгоблок вычитывает данные из трендов.

Чтение из трендов возможно только под Windows из исполняющей системы Slot_algo. Для чтения используется библиотека доступа к трендам Slot_UmiTrends.dll.

Параметр 1 - тип тренда и тип задания времени:

- время задано локально:

- = 0 – тренд реального времени (PB);
- = 1 – минутный тренд;
- = 2 – часовой тренд;

- время задано по Гринвичу:

- = 100 – тренд реального времени (PB);
- = 101 – минутный тренд;
- = 102 – часовой тренд.

Параметр 2 – тип требуемых данных:

- = 0 - усредненные данные (только для часового и минутного трендов; для тренда PB используются мгновенные значения);
- = 1 - усредненные данные без учета выхода за физический диапазон (только для часового и минутного трендов; для тренда PB используются мгновенные значения);
- = 2 - мгновенные значения;
- = 3 - минимальные значения (только для часовых и минутных трендов; для тренда PB только при заданном усреднении (см. Параметр 3), иначе – мгновенные значения);
- = 4 - максимальные значения (только для часовых и минутных трендов; для тренда PB только при заданном усреднении (см. Параметр 3), иначе – мгновенные значения).

Параметр 3 – режим усреднения данных:

- = 0 - без усреднения;
- > 0 - с усреднением заданным в секундах (возможно с дробной частью для задания мсек).

Операнд 1 - описывает требуемые сигналы для чтения;

Операнд 2 - описывает временной диапазон запроса:

- 1-й вектор (из 7-ми элементов) - дата начала выборки;
- 2-й вектор (из 7-ми элементов) - дата конца выборки; если дата конца не задана, предполагается текущий момент времени.

Формат вектора времени:

(1) - год (2)-месяц (3)-день (4)-час (5)-минута (6) - секунда (7) - мсек

Результат - в общем случае, содержит массив данных, где 1-я размерность меняется от 1 до количества сигналов, определенных операндом 1, а 2-я - количество прочитанных точек по времени.

1. Если в результате определено условие результата, то в его 1-й элемент записывается код завершения:

- = 0 - данные прочитаны нормально;

Отрицательные значения означают ошибки:

- 101 - не задан путь трендов;
- 102 - Ошибка загрузки Slot_Umitrends или получения адреса umi_read_for_algo;
- 103 - Ошибка выделения памяти алгоблоку
- 99 - Ошибка распределения памяти внутри umi_read_for_algo;
- 1 - Не найдено трендов по запросу;
- 2 - Не найдено данных по запросу;

- 3 - Ошибка чтения трендов;
- 4 - Структура трендов разрушена;
- 5 - Неудача определения количества точек в тренде для запроса;
- 6 - Диапазон запроса меньше периода усреднения;
- 33 - Неверный диапазон запроса;

2. Если в условии результата более одного элемента, то во 2-й записывается реальное количество точек по времени, считанных из файла.

3. Путь к трендам запоминается при первом проходе. Если требуется изменить, необходима "Генерация".

Использование флага свертки операндов в алгоблоке

Возможно изменение направления расположения данных:

Стандартно:

- 1-я размерность – сигналы;
- 2-я размерность – точки;

Инверсно:

- 1-я размерность – точки;
- 2-я размерность – сигналы.

А.1.4 Описание сигнала

Данный алгоблок предназначен для чтения символьных описателей сигналов БД MWBRidge (A и D).

Операнд 1 - задает структуру исходных данных (по векторам A и D); в качестве источника м.б. заданы только БД A и D, остальные игнорируются; допустимо раздельное использование расширения операнда для другой БД;

Операнд 2

1. если в качестве источника данных заданы A или D, то операнд рассматривается как источник данных наравне с операндом 1, запись в результат производится в свободные вектора после записи данных операнда 1; расширение операнда может использоваться для любой из баз – A, D или SR
2. если источник данных - символьный вектор SR, то данные в нем рассматриваются как разделители между выводимыми полями одного сигнала. Возможно задание до 3-х разделителей, которые используются после выдачи соответствующего поля (порядок выдачи определяется параметром 1). Если SR не используется, то используется разделитель полей по умолчанию вида "_:_:"; если источник данных не A,D или SR - операнд 2 не используется.

Расширение операнда 2 может использоваться для любой из баз – A, D или SR

Результат - в качестве приемника данных возможно использование только символьного массива SR.

Размерность результата д.б. по крайней мере на 1 выше размерности операндов. Запись производится в вектора результата.

Параметр 1 - определяет выводимые поля сигнала тип перекодировки. Используется следующая цифровая кодировка:

- 1 - имя;
- 2 - описание;
- 3 - ед.измерения физ. величины;
- 4 - ед.измерения элек. величины.

Последняя цифра числа (самая младшая) определяет тип перекодировки и может иметь следующие значения:

- 0 - без перекодировки;
- 1 - перекодировка Win->DOS;
- 2 - перекодировка Win->Linux;
- 3 - перекодировка DOS->Win;
- 4 - перекодировка DOS->Linux;
- 5 - перекодировка Linux->Win;
- 6 - перекодировка Linux->DOS;

Цифры могут располагаться в любой последовательности (но не более 5 цифр в числе).

Например, число 21432 задает вывод данных в следующей последовательности:

описание, имя, ед.изм.элек.вел., ед.изм.физ.вел. с перекодировкой Win->DOS. Между выдаваемыми полями располагаются разделители. При выводе в каждом поле убираются ведущие и завершающие пробелы и табуляции. Вывод строки всегда завершается 0-байтом. Если заданная размерность вектора результата меньше длины сформированной строки, то строка укорачивается до приемлемого размера.

Параметр 2 - используется только при выдаче описания: определяет количество пропускаемых символов с начала строки описания (еще до убирания ведущих пробелов и табуляций);

Параметр 3 - используется только при выдаче описания: определяет общее количество символов в строке описания (еще до убирания ведущих пробелов и табуляций).

A.2 Управление

1. Вызов процедуры
2. Вызов подпрограммы
3. Группы учета
4. Переход
5. Возврат

A.2.1 Вызов процедуры

Производится вызов алгоблочной процедуры.

Параметр 1 - номер вызываемой алгоблочной процедуры;

Параметр 2 - номер 1-го исполняемого алгоблока в вызываемой процедуре;

Параметр 3 - номер последнего исполняемого алгоблока в вызываемой процедуре.

Все параметры рассматриваются как константы (режим ссылок на БД игнорируются).

Абсолютный номер вызываемой процедуры должен отличаться от номера текущей процедуры и не входить в список процедур, образующих цикл обхода.

При вызове производится передача управления на указанные параметрами 1 и 2 процедуру и алгоблок. Завершается процедура или:

- по достижении последнего исполняемого алгоблока (заданного параметром 3);
- по выполнению алгоблока **Возврат** (см. далее).

При вызове процедуры автоматически производится сохранение индивидуальных параметров алгоблоков, входящих в процедуру (например, для таймеров - засечки времени, режим работы таймера и т.д.), что позволяет вызывать одну и ту же процедуру несколько раз в течение одного такта исполнения.

Если номер процедуры задан **отрицательным числом**, то производится передача входных параметров в вызванную процедуру и возврат выходных параметров.

Входными параметрами являются данные операндов 1 и 2 вместе с их возможными расширениями, заданными в алгоблоке **Вызов процедуры**. Эти данные копируются в соответствующие операнды первого исполняемого алгоблока вызванной процедуры.

Выходными параметрами являются данные результата (с возможным расширением) последнего исполняемого алгоблока вызванной процедуры (в частности, это м.б. алгоблок **Возврат**). Эти данные копируются в структуру результата, заданную при вызове процедуры в алгоблоке **Вызов процедуры**.

После завершения исполнения вызванной процедуры управление передается в основную (текущую) процедуру на следующий алгоблок.

A.2.2 Вызов подпрограммы

Производится вызов подпрограммы алгоблоков среднего уровня (только Windows).

Параметр 1 - номер исполняемой подпрограммы (1-256); если параметр 1 находится вне указанного диапазона, то вычисления не производятся;

Параметры 2, 3 - входные данные (см. ниже).

Сначала (однократно) загружается файл подпрограмм `Procedur.alg`. Файл должен располагаться в каталоге проекта. При ошибке загрузки вычисления не проводятся. Далее на основании данных операндов (с возможными расширениями) формируются входные данные для алгоблока вызова подпрограммы. После этого исполняется заданная подпрограмма и производится выборка результатов в поле результата (с возможным расширением).

Входные данные подпрограммы (<=24)

Входные данные для алгоблока вызова подпрограммы формируются на основании данных операндов, расширений операндов и параметров 2 и 3.

Последовательность обхода следующая:

Оп1 ... Расш.Оп1 ... Оп2 ... Расш.Оп2 ... Пар2 ... Пар3

Количество элементов данных, используемых в каждом элементе обхода определяется величиной 1-й размерности (параметры 2 и 3 всегда рассматриваются как скаляры). Общее количество входов не больше 24.

Если общее количество входов больше 24 или реально используемого подпрограммой, то остальные входы не используются.

Выходные данные подпрограммы (<=24)

Выходные данные записываются в 1-е размерности результата и расширения результата (если оно задано).

Если общее количество выходов больше 24 или реально используемого подпрограммой, то остальные выходы не используются.

Цикличность вызовов определяется 2-й размерностью операндов и результата. Количество циклов равно минимуму 2-й размерности результата и наибольшей 2-й размерности используемых операндов.

Если цикличность отсутствует, то подпрограмма выполняется однократно, но результат может быть тиражирован:

- для матрицы результата: если 2-я размерность больше цикличности, то последний рассчитанный вектор копируется в оставшиеся свободные вектора;
- для вектора результата: в результат записывается максимально возможное количество порций результата.

Обработка недостоверных входных данных

1. Проверка достоверности не задана:

- для недостоверных операндов используется константа;

2. Проверка достоверности включена и

2.1 Включено исполнение операции:

- для недостоверных используется сам операнд;
- в результат записывается недостоверный статус операции;

2.2 Операция не исполняется:

- если есть недостоверные данные операция НЕ выполняется;

- используется константа результата (если разрешена); константа пишется со своим статусом, флаг записи статуса игнорируется;
- записывается недостоверный статус операции. Если задана запись статуса, то статус константы будет затерт статусом операции.

A.2.3 Группы учета

Производится обсчет учетных групп, заранее созданных пользователем.

Параметр 1 - тип обсчитываемых групп:

- = 0 - оба типа;
- = 1 - только учетные группы;
- = 2 - только расходы и тепло.

Параметры 2 и 3 определяют начальный и конечный номера групп, участвующих в обсчете (для случая Параметр1 = 0 - определяют группы обоих типов). Если номера заданы равными 0, то производится обсчет всех групп.

A.2.4 Переход

Операция предназначена для изменения порядка обработки алгоблоков в цикле исполнения.

Параметр 1 - номер алгоблока перехода в той же процедуре;

- < 0 - количество алгоблоков перед текущим алгоблоком;
- > 0 - абсолютный номер алгоблока в процедуре;
- = 0 - алгоблок НЕ исполняется;

- Если алгоблок, на который осуществляется переход разделен от текущего алгоблоком цикла переход на предыдущий цикл обхода:

- управление передается на требуемый алгоблок, но при этом производится выход из цикла расчета на данном такте; на следующем такте расчет начнется с алгоблока, на который осуществлен переход.

- Переход к новому алгоблоку не может пересекать границы процедуры:

- если он больше количества алгоблоков в процедуре, то производится **выход** из процедуры (при этом переход через алгоблок цикла НЕ учитывается);
- если он меньше 1-го, то переход производится на 1-й алгоблок процедуры.

Параметр 2 - количество проходов через данный алгоблок на данном цикле исполнения процедуры:

- <= 0 - алгоблок НЕ исполняется.

Параметр 3 - общее время прохождения через данный алгоблок в мсек:

- <= 0 - время прохождения НЕ учитывается.

Результат содержит информацию о выполнении перехода:

- 1-й элемент результата содержит счетчик проходов внутри такта;
- 2-й элемент результата содержит текущее время;
- 1-й элемент условия результата содержит флаг перехода. (Подробно см. ниже).

Алгоритм функционирования алгоблока

Необходимое условие осуществления передачи управления:

$\text{abs}(\text{ОП1}) > \text{abs}(\text{ОП2})$, где:

ОП1 - операнд 1, свернутый суммой к скаляру;

ОП2 - операнд 2, свернутый суммой к скаляру;

$\text{abs}()$ - абсолютное значение;

иначе управление передается, как обычно, следующему алгоблоку.

1. Проверка Пар1 и Пар2 на 0; если есть хотя бы один 0 - переход к п.9;
2. Проверка $\text{abs}(\text{ОП1}) > \text{abs}(\text{ОП2})$; если нет, то переход к п.8;
3. Нарастание значения счетчика проходов через алгоблок (count) и времени прохождения (Time) (если использование времени определено Пар3);
4. Проверка: $\text{Count} \leq \text{Пар2}$ И $\text{Time} \leq \text{Пар3}$; если НЕТ, то переход к п.7;
5. Запись в результат:
 $\text{РЕЗ}(1)=\text{Count}$, $\text{РЕЗ}(2)=\text{Time}$
запись в условие результата:
 $\text{УСРЕЗ}(1)=\text{«ИСТИНА»}$
6. Осуществление передачи управления на требуемый алгоблок и ВЫХОД
7. Запись в результат:
 $\text{РЕЗ}(1)=\text{Count}$, $\text{РЕЗ}(2)=\text{Time}$
запись в условие результата:
 $\text{УСРЕЗ}(1)=\text{«ЛОЖЬ»}$
и ВЫХОД (НЕТ передачи управления)
8. Если $\text{Time} = 0$ (1-й вход в алгоблок на данном такте работы), то
Запись в результат:
 $\text{РЕЗ}(1)=0$, $\text{РЕЗ}(2)=0$
запись в условие результата:
 $\text{УСРЕЗ}(1)=\text{«ИСТИНА»}$
, иначе результат НЕ меняется
и ВЫХОД (НЕТ передачи управления)
9. Запись в результат:
 $\text{РЕЗ}(1)=0$, $\text{РЕЗ}(2)=0$
запись в условие результата:
 $\text{УСРЕЗ}(1)=\text{«ЛОЖЬ»}$
и ВЫХОД (НЕТ передачи управления)

А.2.5 Возврат

Данный алгоблок осуществляет передачу управления:

- для процедуры, вызванной алгоблоком **Вызов процедуры** - на следующий алгоблок вызывающей процедуры;
- для процедур основного цикла производится выход из данной процедуры.

Если алгоблок используется для возврата из процедуры, вызванной алгоблоком **Вызов процедуры** с передачей параметров (номер процедуры при вызове задан **отрицательным числом**), производится копирование данных результата данного алгоблока в структуру результата, заданную при вызове процедуры в алгоблоке **Вызов процедуры**.

Если алгоблок используется для возврата из процедуры, вызванной алгоблоком **Вызов процедуры** без передачи параметров (номер процедуры при вызове задан **положительным числом**), то возможно задание режима копирования операндов в результат:

Параметр 1 - режим копирования:

= 0 - без копирования;

= 1 - копируется только операнд 1;

= 2 - копируется только операнд 2;

= 3 - копируется сначала операнд 1, затем в свободное место - операнд 2.

А.3 Арифметические операции

1. Элементарные бинарные операции
2. Элементарные унарные операции
3. Взвешенные операции
4. Попадание в Диапазон (арифметическое)
5. Селектор Арифметический
6. Преобразование
7. Прогрессия
8. Система линейных уравнений
9. Определение коэффициентов полинома
10. Масштабирование
11. Выборка

Для первых 6-ти типов операций, а также для операций «Масштабирование» и «Выборка» возможно использование **проверки исполнения по условиям результата**.

Для первых 4-х типов операций, а также для операции «Прогрессия», «Масштабирование» и «Выборка» возможно указание **выборки данных**.

А.3.1 Элементарные бинарные операции

Сложение, Вычитание, Умножение, Деление, Остаток от деления, Деление нацело, Среднее, Минимум, Максимум

Параметр 1: использование операндов:

- = 0 - используются оба операнда; (свертка по типу операции);
- = 1 - используется один операнд - 1-й; (свертка по типу операции);
- = 2 - используется один операнд - 2-й; (свертка по типу операции);
- = 10 - используются оба операнда; (свертка по флагу «И»/«ИЛИ»; кроме операций: Среднее, Минимум, Максимум);
- = 11 - используется один операнд - 1-й; (свертка по флагу «И»/«ИЛИ»; кроме операций: Среднее, Минимум, Максимум);
- = 12 - используется один операнд - 2-й; (свертка по флагу «И»/«ИЛИ»; кроме операций: Среднее, Минимум, Максимум).

Для операций Сложение, Вычитание, Умножение, Деление могут выполняться предоперации перед основной операцией. Тип предоперации определяется флагом свертки соответствующего операнда:

- если Параметр 2 # 0 выполняется предоперация:

Оп1*3нПар2 - для флага свертки «И»;

Оп1+3нПар2 - для флага свертки «ИЛИ»;

-если Параметр 3 # 0 выполняется предоперация:

Оп2*3нПар3 - для флага свертки «И»;

Оп2+3нПар3 - для флага свертки «ИЛИ».

При выполнении операции при необходимости операнды приводятся к размерности результата сворачиванием:

Свертка по типу операции

- сложением для операций сложения, вычитания, среднего, минимума и максимума;
- умножением для операций умножения и любых операций деления.

Свертка по флагу «И»/«ИЛИ»

- сложением по флагу «ИЛИ»;
- умножением по флагу «И».

Алгоритм приведения операндов в зависимости от размерностей

Результат - Скаляр

Оба операнда свертываются к скаляру

Результат - Вектор

	Операнд 1	Скаляр	Вектор	Матрица
Операнд 2				
Скаляр		1	2	3
Вектор		4	5	6
Матрица		7	8	9

Сл.1 - Тривиальный случай; высчитывается скаляр, который тиражируется на вектор результата;

Сл.2,4 - Образуется вектор результата - используется вектор одного операнда и скаляр другого для всех элементов вектора;

Сл.3,7 - Матрицы свертываются к вектору - далее, как в сл.2 и 4;

Сл.5 - Тривиальный расчет по минимальным размерностям векторов;

Сл.6,8 - Соответствующая матрица свертывается к вектору, далее, как в сл.5;

Сл.9 - Матрицы свертываются к векторам, далее, как в сл.5.

Использование флага свертки операндов в алгоблоке

В случае результата - вектора возможно изменение направления свертки.

Размерность:

- для однооперандных операций равна минимуму 1-й(2) размерности Оп и Рез;

- для двухоперандных операций со скаляром равна минимуму 1-й(2) не скалярного Оп и 1-й размерности Рез;

- для двухоперандных операций без скаляров равна минимуму 1-х(2-х) размерностей Оп1 и Оп2 и 1-й размерности Рез.

Оп1 и Оп2 сворачиваются по 2-й или 1-й размерности.

Примечание. В скобках указана размерность, используемая при инверсной свертке.

Результат - Матрица

	Операнд 1	Скаляр	Вектор	Матрица
Операнд 2				
Скаляр		11	12	13
Вектор		14	15	16
Матрица		17	18	19

Сл.11 - Вычисление скаляра с тиражированием результата на матрицу;

Сл.12,14 - Образуется вектор результата - используется вектор одного операнда и скаляр другого для всех элементов вектора - далее результирующий вектор тиражируется на матрицу;

Сл.13,17 - Вычисляется матрица результата по матрице операнда и скаляру (по всем элементам матрицы);

Сл.15: - для бинарных операций (кроме минимума, максимума и среднего) - выполняется операция, аналогичная каноническому произведению векторов:

$\text{Рез}(i,j) = \text{Оп1}(i) \text{ F } \text{Оп2}(j)$, где

$\text{Рез}(i,j)$ – элементы матрицы результата;

$\text{Оп1}(i)$, $\text{Оп2}(j)$ – элементы векторов операндов;

F – исполняемая операция;

- в остальных случаях: расчет по минимальным размерностям векторов с тиражированием на вектора матрицы;

Сл.16,18 - Матрица результата получается обработкой соответствующих векторов матрицы одного операнда и вектора другого;

Сл.19 - Матрицы обрабатываются поэлементно по минимальным размерностям.

Использование флага свертки операндов в алгоблоке

В случае 15 (BBM) возможно изменение направления свертки.

Размерность матрицы:

1-я - 1-я размерность Оп1 и 1(2) размерность Рез;

2-я - 1-я размерность Оп1 и 2(1) размерность Рез.

Примечание. В скобках указана размерность, используемая при инверсной свертке.

При использовании только одного операнда (или в том случае, если второй операнд является недоверным) результат вычисляется следующим образом:

Операция	1-й операнд	2-й операнд
Сложение, умножение, среднее, минимум, максимум	Результатом является достоверный операнд	
Вычитание	Операнд	- Операнд
Деление	Операнд	1 / Операнд
Остаток от деления	Дробная часть операнда	Остаток от деления: 1 / операнд
Деление нацело	Целая часть операнда	Целая часть от деления: 1 / операнд

А.3.2 Элементарные унарные операции

Квадрат, Корень, Экспонента, $10^{**}X$, $2^{**}X$, $X^{**}Y$, Логарифм натуральный, Логарифм десятичный, Логарифм двоичный, Логарифм по основанию A, Синус, Косинус, Тангенс, Котангенс, Арксинус, Арккосинус, Арктангенс, Арккотангенс

Операнд 1 - используемый операнд.

Операнд 2 - дополнительный операнд, если его использование задано параметром 1 алгоблока.

Параметр 1

=0 - используется только 1-й операнд (свертка по типу операции);

=1 - используется также 2-й операнд (свертка по типу операции);

=10 - используется только 1-й операнд (свертка по флагу «И»/«ИЛИ»);

=11 - используется также 2-й операнд (свертка по флагу «И»/«ИЛИ»).

Параметр 3 – определяет режим использования операнда 2 (если параметр 1 равен 1 или 11):

=0 – сначала производится вычисление аргумента функции добавлением операнда 2 к операнду 1 (по типу свертки), затем производится вычисление функции;

#0 – сначала вычисляется функция (как аргумент используется только 1-й операнд), затем полученный результат домножается на 2-й операнд.

Для операций $X^{**}Y$ и $\text{Loga}X$ всегда используются оба операнда:

- $X^{**}Y$: X - 1-й операнд;

Y - 2-й операнд.

- $\text{Loga}X$: X - 1-й операнд;

A - 2-й операнд.

Свертка по типу операции

В зависимости от типа операции свертка 1-го (или 1-го и 2-го) операндов, а также свертка между операндами (при использовании двух операндов) производится следующим образом:

- для операций: **Квадрат, Корень**, а также все **логарифмические операции**: произведением;

- для всех остальных операций (**показательные и тригонометрические функции**): суммированием;

- для операции $X^{**}Y$:

X - свертывается произведением, Y - суммированием;

- для операции $\text{Loga}X$:

X - произведением; A - суммированием.

Свертка по флагу «И»/«ИЛИ»

- сложением по флагу «ИЛИ»;

- умножением по флагу «И».

Алгоритм приведения операндов в зависимости от размерностей (кроме операций $X^{}Y$ и $\text{LOGa}X$).**

Результат - Скаляр

Оба операнда свертываются к скаляру

Результат - Вектор

	Операнд 1	Скаляр	Вектор	Матрица
Операнд 2				
Скаляр		1	2	3
Вектор		4	5	6
Матрица		7	8	9

Сл.1 - Тривиальный случай; высчитывается скаляр, который тиражируется на вектор результата;

Сл.2,4 - Образуется вектор результата - используется вектор одного операнда и скаляр другого для всех элементов вектора;

Сл.3,7 - Матрицы свертываются к вектору - далее как в сл.2 и 4;

Сл.5 - Тривиальный расчет по минимальным размерностям векторов;

Сл.6,8 - Соответствующая матрица свертывается к вектору, далее как в сл.5;

Сл.9 - Матрицы свертываются к векторам, далее как в сл.5.

Результат - Матрица

	Операнд 1	Скаляр	Вектор	Матрица
Операнд 2				
Скаляр		11	12	13
Вектор		14	15	16
Матрица		17	18	19

Сл.11 - Вычисление скаляра с тиражированием результата на матрицу;

Сл.12,14 - Образуется вектор результата - используется вектор одного операнда и скаляр другого для всех элементов вектора - далее результирующий вектор тиражируется на матрицу;

Сл.13,17 - Вычисляется матрица результата по матрице операнда и скаляру (по всем элементам матрицы);

Сл.15 - Расчет по минимальным размерностям векторов с тиражированием на вектора матрицы;

Сл.16,18 - Матрица результата получается обработкой соответствующих векторов матрицы одного операнда и вектора другого;

Сл.19 - Матрицы обрабатываются поэлементно по минимальным размерностям.

Использование флага свертки операндов в алгоблоке

Здесь возможно изменение направления свертки.

Размерность:

- для однооперандных операций равна минимуму 1-й (2) размерности Оп и Рез;

- для двухоперандных операций со скаляром равна минимуму 1-й (2) не скалярного Оп и 1-й размерности Рез;

- для двухоперандных операций без скаляров равна минимуму 1-х(2-х) размерностей Оп1 и Оп2 и 1-й размерности Рез.

Оп1 и Оп2 сворачиваются по 2-й или 1-й размерности.

Примечание. В скобках указана размерность, используемая при инверсной свертке.

А.3.3 Взвешенные операции

Взвешенное Сложение, Взвешенное Умножение, Средневзвешенное

Если 1-й Параметр # 0, то:

2-й параметр - минимально-возможный результат;

3-й параметр - максимально-возможный результат;

Рассчитанный результат будет в границах: $P(2) \leq X \leq P(3)$.

Формулы расчета:

- **Взвешенное Сложение:**

$$X = Op1(1) * Op2(1) + \dots + Op1(n) * Op2(n)$$

- **Взвешенное Умножение:**

$$X = Op1(1) ** Op2(1) * \dots * Op1(n) ** Op2(n)$$

- **Средневзвешенное:**

$$X = (Op1(1) ** Op2(1) * \dots * Op1(n) ** Op2(n)) / SiOp2(i)$$

, где $SiOp2(i)$ - сумма элементов операнда 2 с 1 по N.

Исключения при взвешенном умножении:

1. Если $Op1(i) < 0$ и $Op2(i)$ -дробное --> $X = -ABS(Op1(i)) ** Op2(i)$
2. Если $Op1(i) = 0$ и $Op2(i) < 0$ --> $X = \text{Константа результата}$

Режим приведения операндов перед вычислением (понижение размерности операнда):

Взвешенное сложение, Средневзвешенное	Взвешенное умножение
Сложением	Умножением

Операции понижают размерность, поэтому максимальная размерность результата - вектор.

Расчеты (формулы даны для взвешенного сложения)

Результат – Скаляр

	Операнд 2	Скаляр	Вектор A2(n2)	Матрица A2(n2,m2)
Операнд 1				
Скаляр		1	2	3
Вектор A1(n1)		4	5	6
Матрица A1(n1,m1)		7	8	9

Сл.1,5 - Основной расчет:

$X = SiA1(i) * A2(i)$, где

X - результат

Si - знак суммы по индексу i, $i = 1..min(n1, n2)$

Для случая 1 : $i=1..1$

Сл.2,4 - Один из Оп - скаляр - по всему вектору;

Сл.3 - Свертка матрицы к вектору по 2-й размерности, далее по 2;

Сл.7 - Свертка матрицы к вектору по 2-й размерности, далее по 4;

Сл.6,8,9 - Свертка матриц к векторам и далее по 5.

Использование флага свертки операндов в алгоблоке

Здесь возможно изменение направления свертки.

Количество пар (только изменяемые случаи):

Оп1/Оп2

СМ - 1(2) размерность Оп2;

ВМ - 1(2) размерность Оп2;

МС - 1(2) размерность Оп1;

МВ - 1(2) размерность Оп1 и 1-я размерность Оп2;

ММ - 1(2) размерность Оп1 и 1(2) размерность Оп2;

Свертка Оп проводится размерности по 2(1) размерности.

Примечание. В скобках указана размерность, используемая при инверсной свертке.

Результат - Вектор R(n3) или матрица R(n3,m3)

	Операнд 2	Скаляр	Вектор A2(n2)	Матрица A2(n2,m2)
Операнд 1				
Скаляр		11	12	13
Вектор A1(n1)		14	15	16
Матрица A1(n1,m1)		17	18	19

Сл.11,15 - Обычное Вычисление по формуле (1 или 5) с тиражированием результата;

Сл.12,14 - Один из Оп - скаляр (2 или 4) с тиражированием результата;

Сл.19 - Основной расчет:

$R(i)=S_j A1(j,i) * A2(j,i)$, где

S_j - сумма по j, $j=1..min(n1,n2)$

$R(i)$ - вектор результата $i=1..min(m1,m2,n3)$

Сл.16:

$R(i)=S_j A1(j) * A2(j,i)$, где

S_j - сумма по j, $j=1..min(n1,n2)$

$R(i)$ - вектор результата $i=1..min(m2,n3)$

Сл.18:

$R(i)=S_j A1(j,i) * A2(j)$, где

S_j - сумма по j, $j=1..min(n1,n2)$

$R(i)$ - вектор результата $i=1..min(m1,n3)$

Сл.13:

$R(i)=S_j A1(1) * A2(j,i)$, где

S_j - сумма по j, $j=1..n2$

$R(i)$ - вектор результата $i=1..min(m2,n3)$

Сл.17:

$R(i) = \sum_j A1(j,i) * A2(1)$, где
 \sum_j - сумма по j, $j=1..n1$
 $R(i)$ - вектор результата $i=1..\min(m1,n3)$

Использование флага свертки операндов в алгоблоке

Здесь возможно изменение направления свертки.

2-я размерность - размерность свертки

Количество пар (только изменяемые случаи):

ОП1/ОП2

СМ : 1-я размерность вычислений равна 2(1) размерности Оп2 и 1-й размерности Рез;

2-я размерность свертки равна 1(2) размерности Оп2;

ВМ : 1-я размерность вычислений равна минимуму 2(1) размерности Оп2 и 1-й размерности Рез;

2-я размерность свертки равна минимуму 1-й размерности Оп1 и 1(2) размерности Оп2;

МС : 1-я размерность вычислений равна минимуму 2(1) размерности Оп1 и 1-й Рез;

2-я размерность свертки равна 1(2) размерности Оп1;

МВ : 1-я размерность вычислений равна минимуму 2(1) размерности Оп1 и 1-й размерности Рез;

2-я размерность свертки равна минимуму 1(2) размерности Оп1 и 1-й размерности Оп2;

ММ : 1-я размерность вычислений равна минимуму 2(1) размерности Оп1, 2(1) размерности Оп2 и 1-й размерности Рез;

2-я размерность свертки равна минимуму 1(2) размерности Оп1 и 1(2) размерности ОП2.

Примечание. В скобках указана размерность, используемая при инверсной свертке.

Тиражирование результата производится, если:

Результат возможный (определяется указанным типом)	Результат фактический	Действие
Вектор	Скаляр	Скаляр тиражируется на весь вектор
Матрица	Скаляр	Скаляр тиражируется на всю матрицу
Матрица	Вектор	Вектор тиражируется на все вектора

А.3.4 Попадание в Диапазон (арифметическое)

Операнд 1 - Анализируемые данные.

Операнд 2 - Набор диапазонов.

Результат - Номера диапазонов, в которые попадают анализируемые данные.

Производится анализ попадания значения в диапазон, при этом результатом является номер диапазона.

При выборке числа и диапазона операнды проверяются на достоверность только по статусу (проверка по БД не используется).

Параметр 1 - режим задания диапазонов:

= 0 - последовательно: каждый элемент является верхней границей предыдущего диапазона и нижней следующего. Матрица рассматривается как непрерывный вектор.

= 1 - диапазоны задаются парами чисел.

= 2 - аналогичен случаю Параметр1=1, только индекс попадания является в случае матрицы диапазонов номером вектора.

Параметры 2 и 3 - режимы задания границ диапазонов:

Параметр 2	Параметр 3	Границы
0	0	< ... =<
1	0	=< ... =<
0	1	< ... <
1	1	=< ... <

Если граница отсутствует - операция производится, но в результат ставится бит недостоверности.

A.3.5 Селектор Арифметический

Операнд 1 - источник выбора.

Операнд 2 - индекс (рассматривается как элемент типа 2-х байтового целого).

Производится выборка из источника по значению индекса и записывается в результат. Данные из источника выбираются в соответствии с типом приемника результата.

Параметр 1 - режим приведения (где требуется) векторов и матриц к скалярному индексу:

= 0 - использовать как индекс 1-й элемент вектора или матрицы;

= 1 - просуммировать вектор или матрицу и получить индекс (недостоверные данные в сумму не включаются);

Параметр 2 - использование индексов, выходящих за диапазон источника выбора:

= 0 - если индекс больше максимально-возможного, используется этот максимально возможный;

- если индекс меньше минимально-возможного, используется этот минимально возможный.

= 1 - если индекс больше максимально-возможного или меньше минимально-возможного, копирование в результат не производится.

Если индекс является недостоверным - копирование не производится.

Выбираемый операнд 1 не проверяется на достоверность и по векторам, а копируется как есть со своим статусом.

Алгоритм выборки с учетом размерностей

Операнд 1 – Источник	Операнд 2 – Индекс	Результат		
		Скаляр	Вектор	Матрица
Скаляр	Скаляр Вектор Матрица	Копирование в 1-й элемент результата с тиражированием в вектор и матрицу		
Вектор	Скаляр	Выборка из вектора	Копирование вектора с тиражированием в матрицу	
Вектор	Вектор Матрица	Приведение вектора к скаляру по Параметру 1	Копирование вектора с тиражированием в матрицу	
Матрица	Скаляр	Матрица рассматривается как вектор	Скаляр используется как номер вектора	Копирование всей матрицы
Матрица	Матрица	Матрица рассматривается как вектор; Приведение вектора к скаляру по Параметру 1		Копирование всей матрицы
Матрица	Вектор	Матрица рассматривается как вектор; Приведение вектора к скаляру по Параметру 1		Копирование векторов матрицы ОП1 в результат

Для случая Матрица-Вектор-Матрица возможно изменение направления свертки.

А.3.6 Преобразование

Операнд 1 - преобразуемая величина;

Операнд 2 - данные для преобразования (таблица или коэффициенты полинома).

Параметр 1 - режим выбора преобразования:

= 0 - по данным из операнда 2;

= N - по таблице преобразований из БД MIKROB;

= -1 - двумерная аппроксимация по данным операнда 2.

Параметр 2 - тип преобразования:

= 0 - кусочно-линейная аппроксимация;

= 1 - полиномиальная аппроксимация;

Параметр 2 используется только при задании преобразования по данным операнда 2 (параметр 1 = 0, -1).

Преобразование по данным в операнде 2 (параметр 1 = 0)

Кусочно-линейная аппроксимация

Операнд 2 рассматривается как таблица, состоящая из пар чисел:

$(X_1, Y_1), (X_2, Y_2), \dots, (X_n, Y_n)$ и для любого i : $X_{i+1} \geq X_i$
(расположены по возрастанию)

где:

X_i - значения, используемые для анализа попадания операнда 1 в диапазон;

Y_i - значения, по которым формируется выходное значение.

Размер таблицы определяется размерностью операнда 2:

$N = \text{Разм_1} * \text{Разм_2} / 2$, где:

Разм_1 - 1-я размерность операнда 2;

Разм_2 - 2-я размерность операнда 2.

После нахождения диапазона:

$X_i \leq A \leq X_{i+1}$ (A - значение операнда 1)

определяются коэффициенты для уравнения прямой $Y = K * X + B$:

$K = (Y_{i+1} - Y_i) / (X_{i+1} - X_i)$

$B = Y_i - K * X_i$

и находится результирующее значение:

$R = K * A + B$

Полиномиальная аппроксимация

Расчет результирующего значения производится по формуле:

$Y = A_0 + A_1 * X + A_2 * X^2 + \dots + A_N * X^N$

, где:

Y - результат,

A_0, \dots, A_N - коэффициенты полинома

X - значение операнда 1

Операнд 2 в данном случае содержит коэффициенты полинома, начиная с A_0 .

Количество коэффициентов определяется размерностью операнда 2:

$N = \text{Разм_1} * \text{Разм_2}$

, где: Разм_1 - 1-я размерность операнда 2;

Разм_2 - 2-я размерность операнда 2.

Структурирование размерностей (параметр 1 ≥ 0)

Результат – скаляр

Операнд 1 сводится к скаляру сложением.

Результат - вектор $V_3(n_3)$

Операнд 1

C1 - скалярный результат формируется в B3(1), на остальные элементы вектора производится тиражирование;

B1(n1) - формируется вектор результатов по минимальным размерностям: $\text{Min}(n1, n3)$;

M1(n1, m1) - операнд 1 сворачивается сложением по 2-й размерности к вектору B1(n1); далее как в случае векторного результата.

Использование флага свертки операндов в алгоблоке

Здесь возможно изменение направления свертки.

Количество результатов равно минимуму 1(2) размерности Оп1 и 1 размерности Рез.

Свертка Оп1 проводится по 2(1) размерности.

Примечание. В скобках указана размерность, используемая при инверсной свертке.

Результат - матрица M3(n3, m3)

Операнд 1

C1 - скалярный результат формируется в M3(1,1), на остальные элементы матрицы производится тиражирование;

B1(n1) - формируется вектор результатов по минимальным размерностям: $\text{Min}(n1, n3)$; далее производится тиражирование векторного результата на остальные вектора матрицы;

M1(n1, m1) - формируется матричный результат по минимальным размерностям: $\text{Min}(n1, n3)$ и $\text{Min}(m1, m3)$.

Двумерная аппроксимация (параметр 1 = -1)

Операнд 1 описывает оба входных параметра. Входные параметры должны следовать парами:

(P1(1), P2(1)), (P1(2), P2(2)), ... , где P1 - 1-я входная величина, P2 - 2-я. Т.о. входной вектор операнда 1 размерностью 10 описывает 5 пар входных точек, и, следовательно, требует в результате как минимум вектор размерностью 5. Если операнд 1 описан матрицей M1(N1, K1), то она будет рассматриваться как вектор вида:

M1(1,1), M1(2,1), ..., M1(N1,1), M1(1,2), M1(2,2), ..., M1(N1, K1)

В случае недостоверности входных данных для P1 будет использована константа операнда 1, а для P2 константа операнда 2.

Кусочно-линейная аппроксимация (параметр 2 = 0)

Операнд 2 должен содержать матрицу преобразования M2(N2, K2) вида:

M2(1,1) - не используется;

M2(1,2) .. M2(1, K2) - K2-1 точка, описывающая значения параметра 1 (P1), расположенные по возрастанию;

M2(2,1) .. M2(N2,1) - N2-1 точка, описывающая значения параметра 2 (P2), расположенные по возрастанию.

Остальные точки матрицы (кроме 1-го столбца и 1-й строки) содержат опорные значения выходной величины для расчета.

Полином аппроксимации имеет вид:

$$A_0 + A_1 * P_1 + A_2 * P_2 + A_3 * P_1 * P_1 + A_4 * P_1 * P_2 + A_5 * P_2 * P_2 + \dots$$

,где А - коэффициенты полинома;

P1 и P2 - входные параметры.

Операнд 2 должен содержать коэффициенты полинома А, начиная с младшего (А0). Количество используемых коэффициентов равно размерности операнда 2:

$$K_{kf} = 1_Размерность_ОП2 * 2_Размерность_ОП2$$

Структурирование размерностей

Количество входных точек определяется как:

$$K_{vx} = (1_Размерность_ОП1 * 2_Размерность_ОП1) / 2$$

Количество точек в результате:

$$K_{pz} = 1_Размерность_РЕЗ * 2_Размерность_РЕЗ$$

Количество рассчитываемых точек равно: $\text{Min}(K_{vx}, K_{pz})$

А.3.7 Прогрессия

Данный алгоблок позволяет реализовать арифметическую и геометрическую прогрессии.

Операнд 1 - начальное значение;

Операнд 2 - шаг прогрессии;

Параметр 1 - тип прогрессии:

= 0 – арифметическая;

= 1 – геометрическая;

Свертка операнда 1 (при необходимости) производится по типу прогрессии:

- для арифметической – сложением;

- для геометрической – умножением.

= 10 – арифметическая;

= 11 – геометрическая;

Свертка операнда 1 (при необходимости) производится по флагу свертки «И»/«ИЛИ»:

- для «ИЛИ» – сложением;

- для «И» – умножением.

Параметры 2,3 - диапазон прогрессии (если не равны).

Если диапазон задан, прогрессия закольцовывается внутри диапазона.

Структурирование размерностей

Операнд 1 рассматривается, как имеющий размерность минимум на 1 меньше размерности результата.

При необходимости, производится свертка операнда 1 по типу прогрессии: для арифметической - сложением, для геометрической - умножением.

Использование флага свертки операндов в алгоблоке

Для результата - матрица возможно изменение направления свертки.

Количество прогрессий равно минимуму 1(2) размерности Оп1 и 2(1) размерности Рез.

Свертка Оп1 проводится по 2(1) размерности.

Примечание. В скобках указана размерность, используемая при инверсной свертке.

А.3.8 Система линейных уравнений

Данный алгоблок позволяет решить систему линейных уравнений вида:

$$A_{11} \cdot X_1 + A_{12} \cdot X_2 + \dots + A_{1N} \cdot X_N = C_1$$

...

$$A_{N1} \cdot X_1 + A_{N2} \cdot X_2 + \dots + A_{NN} \cdot X_N = C_N$$

Операнд 1 - $A(N,N)$ - матрица коэффициентов (квадратная);

Операнд 2 - $C(N,K)$ - вектор или матрица ($k \neq 1$) свободных членов;

Результат - $X(N,K)$ - вектор или матрица ($k \# 1$) результата.

Параметр 1 - логический вектор $LSA(N)$, если задан ссылкой, иначе все элементы вектора считаются равными "Истина"; LSA - вектор, элемент которого указывает, что соответствующий столбец и строка матрицы A ненулевые;

Параметр 2 - вектор $MC(N)$, если задан ссылкой, иначе считается каждый элемент равен N ; MC - вектор, каждый элемент которого указывает, где заканчиваются ненулевые коэффициенты в строке (столбце), $(MC(1) \leq MC(2) \leq MC(3) \leq \dots)$;

Если в результате получается вектор, то он тиражируется на матрицу результата, если возможно. Перед вычислением предыдущий результат вычитывается в вектор X , формируются вектора Lsa и Mc , после вычисления X копируется в результат.

Размерности N и K ограничены 10 и 5 соответственно.

А.3.9 Определение коэффициентов полинома

Данный алгоблок позволяет определить коэффициенты полинома вида:

$$Y = A_0 * X^{**0} + A_1 * X^{**1} + A_2 * X^{**2} + \dots + A_N * X^{**N}$$

Операнд 1 - вектор входа $X(nr)$;

Операнд 2 - вектор выхода $Y(nr)$;

Nr - количество пар точек.

Результат - вектор коэффициентов зависимости $A(nr_m)$, Nr_m - количество коэффициентов.

Параметр 1 - если задан ссылкой, то по ссылке вычитывается логический вектор $lfp(nr_m)$ - признак включения коэффициента зависимости в расчет (если равен "Истина"); если не задан ссылкой, то считается, что каждый элемент равен "Истина";

Параметр 2 - если задан ссылкой, то по ссылке вычитывается вещественный вектор $Ast(nr)$, содержащий веса точек; если не задан ссылкой, то веса считаются равными 1.0.

Количество пар точек Nr определяется по минимальным размерностям векторов X и Y . Количество коэффициентов Nr_m определяется размерностью результата.

ПРИМЕЧАНИЕ:

- Ограничение на размер входных векторов: 10;
- Ограничение на размер выходного вектора коэффициентов: 10.

А.3.10 Масштабирование

Данный алгоблок позволяет производить масштабирование входной величины (преобразование по выбранной формуле с учетом диапазонов).

Операнд 1 - масштабируемая величина;

Операнд 2 - параметры масштабирования (задаются вектором максимум из 4-х элементов для каждого входного элемента);

Результат - масштабированная величина;

Параметр 1 - тип шкалы преобразования:

= 0 - линейная

1 - ROT

2 - QRT

3 - SIN

4 - COS

5 - EXP

6 - 1/X

Остальные параметры не используются.

Выполнение расчета зависит от значения параметра 1 и размерности вектора операнда 2.

Для параметра 1 = 0:

Размерность вектора ОП2	Формула
1	Рез=Оп2(1)*Оп1
2	Рез=Оп2(1)*Оп1+Оп2(2)
3	Рез=(Оп1-Оп2(1))/(Оп2(2)-Оп2(1))*Оп2(3)
4	Рез=(Оп1-Оп2(1))/(Оп2(2)-Оп2(1))*(Оп2(4)-Оп2(3))+Оп2(3)

Для параметра 1 > 0:

Размерность вектора ОП2	Формула
1	Рез=F(Оп1)*Оп2(1)
2	Рез=F(Оп1)*Оп2(1)+Оп2(2)
3	Рез=((F((Оп1-Оп2(1))/(Оп2(2)-Оп2(1))))-F(0))/(F(1)-F(0))*Оп2(3) , F(0) F(1) для функций, где F(0)≠0 и F(1)≠1 (exp) (для 1/X - используются F(1) и F(2) соответственно) , иначе полагается F(1)=1 F(0)=0
4	Рез=(F(Pi/2*(Оп1-Оп2(1))/(Оп2(2)-Оп2(1))))*Оп2(3)
4	Рез=(F(Pi/2*(Оп1-Оп2(1))/(Оп2(2)-Оп2(1))))*(Оп2(4)-Оп2(3))+Оп2(3)
4	Рез=((F(Оп1-Оп2(1))/(Оп2(2)-Оп2(1)))-F(0))*(Оп2(4)-Оп2(3))/(F(1)-F(0))+Оп2(3) , F(0) F(1) - для функций, где F(0)≠0 и F(1)≠1 (exp)

(для $1/X$ - используются $F(1)$ и $F(2)$ соответственно)
, иначе полагается $F(1)=1$ $F(0)=0$

Структурирование размерностей

Размерность вычислений определяется минимальными размерностями операнда 1 и результата Оп1Рез.

Учет операнда 2 (шкалы)

Оп1Рез	C1	V1(n1)	M1(n1,m1)
Оп2			
C2	1	2	3
V2(n2)	4	5	6
M2(n2,m2)	7	8	9

Сл.1-7: единая шкала для всех входных данных; берется из Оп2(1..4) (для сл.7 используется 1-й вектор Оп2);

Сл.8 : для всех $i=1, \dots, \min(n1, m2)$ преобразование берется из $M2(1..4, i)$;

Сл.9 : аналог сл.8 - все вектора $M1()$ преобразуются как в сл.8 по одинаковой шкале.

А.3.11 Выборка

Операция выборки предназначена для поиска такого элемента в источнике, значение которого удовлетворяет заданному условию. Условие выборки может быть определено:

- или как операция сравнения с соответствующим элементом операнда 2;
- или как поиск максимального или минимального элемента (в т.ч. по модулю) (параметр 3 больше 3).

Результатом данной операции является индекс выбранного элемента или 0 (в случае режима сравнения), если ни один элемент не найден.

Операнд 1 - источник выборки; операнд проверяется на достоверность по статусу и вектору проверки; также м.б. указано условие выборки;

Операнд 2 - значение для проверки; не проверяется на достоверность, выборка не используется; если операция сравнения с элементом не проводится, операнд не используется.

В режиме сравнения с операндом 2 (параметр 3 меньше 4) результатом операции является индекс найденного элемента (по индексации источника или по абсолютной индексации всего расчетного вектора) или 0, если нет соответствующего элемента в источнике. Для случаев 11, 14, 17 (см. ниже) всегда определяется номер вектора; для случаев 12, 13, 15, 16, 18, 19 (см. ниже) определяется номер элемента в векторе.

В режиме поиска минимального/максимального (в т.ч. по модулю) значения результат всегда определен – индекс соответствующего элемента.

Для результата м.б. указана проверка записи по логическому вектору.

Параметр 1 - индекс начала просмотра; в режиме сравнения с операндом 2 для определенных случаев (см. ниже сл.11, 14, 17), где источником является матрица, индекс определяет номер вектора в матрице; для случаев 12, 13, 15, 16, 18, 19 (см. ниже) индекс определяет номер элемента в векторе матрицы. Если индекс начала меньше 1, он считается равным 1, если больше максимального количества анализируемых элементов, то равен максимальному количеству.

Параметр 2 - направление просмотра:

= 0 - вперед;

= 1 - назад;

Результатом является найденный индекс по индексации внутри источника;

= 10 - вперед;

= 11 - назад;

Результатом является найденный индекс по абсолютной индексации расчетного вектора;

Параметр 3 - условие проверки:

= 0 - проверка на равенство (=);

= 1 - проверка на неравенство (#);

= 2 - проверка на больше (>);

= 3 - проверка на меньше (<);

= 4 - поиск элемента с минимальным значением;

= 5 - поиск элемента с максимальным значением;

= 6 - поиск элемента с минимальным по модулю значением;

= 7 - поиск элемента с максимальным по модулю значением;

В режиме сравнения элемент считается удовлетворяющим условию проверки, если выполняется условие:

Оп1 F(П3) Оп2 = "Истина",

где F(П3) - функция проверки, определенная параметром 3.

Структурирование размерностей для режима сравнения с операндом 2

1. Оп1 - скаляр С или вектор В(N1)

Рез	С	В(N3)	М(N3,K3)
Оп2			
С	1	2	3
В(N2)	4	5	6
М(N2,K2)	7	8	9

Сл.1 - Стандартная выборка

Сл.2/3 - сл.1 с тиражированием результата

Сл.4/5 - Свертка Оп2 к скаляру, далее как сл.1

Сл.5/6/8/9 - Свертка Оп2 к скаляру, далее, как сл.2/3

2. Оп1 - матрица М(N1,K1)

Рез	С	В(N3)	М(N3,K3)
Оп2			
С	11	12	13

B(N2)	14	15	16
M(N2,K2)	17	18	19

Сл.11 - Выборка вектора из K1 векторов Оп1 (каждый элемент вектора сравнивается с скаляром Оп2; результат сравнения определяется значением флага свертки результата **И/ИЛИ** - если он равен **"И"**, то все элементы вектора должны удовлетворять условию проверки, если **"ИЛИ"**, то хотя бы один)

Сл.14/17- Свертка Оп2 к скаляру, далее, как сл.11

Сл.12 - Выборка элемента внутри каждого вектора матрицы Оп1; количество анализируемых векторов определяется как $\min(K1, N3)$; сравнение производится с одним значением - скаляр Оп2

Сл.13 - как в сл.12 с тиражированием на 2-ю размерность результата

Сл.15 - Выборка элемента внутри каждого вектора матрицы Оп1; количество анализируемых векторов определяется как $\min(K1, N2, N3)$; производится поэлементное сравнение каждого вектора с соответствующим элементом вектора Оп2

Сл.18 - Свертка Оп2 к вектору по 2-й размерности, далее, как сл.15

Сл.16 - как в сл.15 с тиражированием на 2-ю размерность результата

Сл.19 - как в сл.18 с тиражированием на 2-ю размерность результата.

Использование флага свертки операндов в алгоблоке

Для случаев 15,16,18,19 возможно изменение направления свертки ОП2.

Количество результатов равно минимуму 2 размерности Оп1 , 1(2) размерности Оп2 и 1 размерности Рез.

Свертка Оп2 к вектору проводится по 2(1) размерности.

Примечание. В скобках указана размерность, используемая при инверсной свертке.

Структурирование размерностей для режима поиска минимума / максимума

В случае, когда операнд 1 является скаляром или вектором, а также в случае, если операнд 1 – матрица, а результат – скаляр выполняется поиск минимумов / максимумов по всему операнду 1, результат тиражируется на все возможные размерности.

В случае, когда операнд 1 является матрицей:

- если результат - матрица, то производится копирование операнда 1 в результат по минимальным размерностям;

- если результат – вектор, то производится поиск по отдельным векторам операнда 1; количество результатов определяется минимумом размерностей: 1-й размерности вектора результата и 2-й размерности операнда 1.

А.3.12 Коммутатор

Алгоблок предназначен для перекрестного копирования данных (коммутации). В прямом режиме работы в управляющих сигналах указывается вход для соответствующего выхода (выход определяется номером управляющего сигнала). В обратном режиме в управляющих сигналах указывается выход для соответствующего входа (вход определяется номером управляющего сигнала).

Операнд 1 - источник данных;

Операнд 2 - управляющие сигналы;

Результат - приемник данных;

Параметр 1 - режим работы:

= 0 - прямой: задается вход;

= 1 - реверсивный: задается выход;

Параметр 2 - определяет номер используемого вектора для матриц (см. ниже). Если параметр задан константой, то считается, что значение параметра считается равным 1; в остальных случаях значение параметра определяется стандартно.

Во всех случаях тип "скаляр" интерпретируется как тип "вектор" единичной длины.

Возможно указание выборки для операнда 2, проверки на запись в условиях результата, а также изменение режима свертки (прямая (сворачиваются столбцы) – обратная (сворачиваются строки)).

Действия в соответствии с размерностями.

Определяющими являются размерности ОП2 и соответствующие размерности ОП1 или РЕЗ в зависимости от режима работы.

Количество обрабатываемых элементов:

K - количество элементов в каждом векторе.

L – количество векторов в операции.

ОП1	V1(n1)	M1(n1,m1)
Рез		
V3(n3)	1	2
M3(n3,m3)	3	4

Сл.1:

```
ОП2:  V2(n2): V3()=V1() F V2();
      Для П1=0:      K=min(n2,n3);
      П1#0:          K=n2;
      M2(n2,m2):
      Для прямой свертки:
      V3()=V1() F M2( ,П2);
      Для П1=0:      K=min(n2,n3);
      П1#0:          K=n2;
      Для обратной свертки:
      V3()=V1() F M2(П2,);
      Для П1=0:      K=min(m2,n3);
      П1#0:          K=m2;
```

Сл.2:

```
ОП2:  V2(n2):
      Для прямой свертки:
      V3()=V2() F M1( ,П2);
      Для П1=0:      K=min(n2,n3);
      Для П1#0:      K=min(n2,n1);
      Для обратной свертки:
      V3()=V2() F M1(П2, );
```

		Для П1=0: $K = \min(m_2, n_3);$ Для П1#0: $K = \min(m_2, n_1);$
	$M_2(n_2, m_2):$	Для прямой свертки: $V_3() = M_2(, P_2) \text{ F } M_1(, P_2);$ Для П1=0: $K = \min(n_2, n_3);$ Для П1#0: $K = \min(n_2, n_1);$ Для обратной свертки: $V_3() = M_2(P_2,) \text{ F } M_1(P_2,);$ Для П1=0: $K = \min(m_2, n_3);$ Для П1#0: $K = \min(m_2, n_1);$
Сл.3	Оп2: $B_2(n_2):$	Для Прямой свертки: $M_3(, P_2) = B_2() \text{ F } V_1();$ Для П1=0: $K = \min(n_2, n_3);$ Для П1#0: $K = \min(n_2, n_1);$ Тиражирование на остальные вектора РЕЗ Для обратной свертки: $M_3(P_2,) = B_2() \text{ F } V_1();$ Для П1=0: $K = \min(n_2, m_3);$ Для П1#0: $K = \min(n_2, n_1);$ Тиражирование на остальные вектора РЕЗ
	$M_2(n_2, m_2):$	Для прямой свертки: $M_3(, i) = M_2(, i) \text{ F } V_1();$ Для П1=0: $K = \min(n_2, n_3)$ в каждом векторе, Для П1#0: $K = \min(n_2, n_1)$ в каждом векторе, $L = \min(m_2, m_3)$ - кол-во векторов, $i = 1, \dots, L$ Для обратной свертки: $M_3(i,) = M_2(i,) \text{ F } V_1();$ Для П1=0: $K = \min(m_2, m_3)$ в каждом векторе, Для П1#0: $K = \min(m_2, n_1)$ в каждом векторе, $L = \min(n_2, n_3)$ - кол-во векторов, $i = 1, \dots, L$
Сл.4	Оп2: $B_2(n_2):$	Для прямой свертки: $M_3(, i) = M_1(, i) \text{ F } V_2();$ Для П1=0: $K = \min(n_2, n_3)$ в каждом векторе Для П1#0: $K = \min(n_2, n_1)$ в каждом векторе $L = \min(m_1, m_3)$ - кол-во векторов, $i = 1, \dots, L$ Для обратной свертки: $M_3(i,) = M_1(i,) \text{ F } V_2();$ Для П1=0: $K = \min(n_2, m_3)$ в каждом векторе Для П1#0: $K = \min(n_2, m_1)$ в каждом векторе $L = \min(n_1, n_3)$ - кол-во векторов, $i = 1, \dots, L$
	$M_2(n_2, m_2):$	

Есть расширение ОП2 - $MP2(nr2, m2)$ (основная - $MO2(no2, m2)$)

1-й индекс определяется основной матрицей

2-й индекс расширением

Для прямой свертки:

Для $P1=0$:

$i=1, \dots, K$, $K=\min(no2, nr2, n3)$ в каждом векторе

$j=1, \dots, L$, $L=\min(m2, m3)$ векторов

$M3(i, j) = M2(i, j) \text{ F } M1(, j)$

Для $P1 \neq 0$:

$i=1, \dots, K$, $K=\min(no2, nr2, n1)$ в каждом векторе

$j=1, \dots, L$, $L=\min(m2, m1)$ векторов

$M3(, j) = M2(i, j) \text{ F } M1(i, j)$

Для обратной свертки:

Для $P1=0$:

$j=1, \dots, K$, $K=\min(m2, m3)$

$i=1, \dots, L$, $L=\min(no2, nr2, n3)$

$M3(i, j) = M2(i, j) \text{ F } M1(, j)$

Для $P1 \neq 0$:

$j=1, \dots, K$, $K=\min(m2, m1)$

$i=1, \dots, L$, $L=\min(no2, nr2, n1)$

$M3(, j) = M2(i, j) \text{ F } M1(i, j)$

Нет расширения:

Поэлементные операции:

Для прямой свертки:

$M3(i, j) = M2(i, j) \text{ F } M1(i, j);$

Для $P1=0$:

$K=\min(n2, n3)$ в каждом векторе

$i=1, \dots, K$

$L=\min(m2, m3)$ - кол-во векторов

$j=1, \dots, L$

Для $P1 \neq 0$:

$K=\min(n2, n1)$ в каждом векторе

$i=1, \dots, K$

$L=\min(m2, m1)$ - кол-во векторов

$j=1, \dots, L$

Для обратной свертки:

$M3(i, j) = M2(i, j) \text{ F } M1(i, j);$

Для $P1=0$:

$K=\min(m2, m3)$ в каждом векторе

$j=1, \dots, K$

$L=\min(n2, n3)$ - кол-во векторов

$i=1, \dots, L$

Для $P1 \neq 0$:

$K=\min(m2, m1)$ в каждом векторе

$j=1, \dots, K$

$L=\min(n2, n1)$ - кол-во векторов

$i=1, \dots, L$

А.4 Логические операции

1. AND, OR, XOR, NAND, NOR, NXOR, Операции Следования
2. Операции сравнения с уставками
3. Операции сравнения значений
4. Операция отрицания NOT
5. Попадание в диапазон (логическое)
6. Селектор логический
7. Триггер
8. Фильтр дискретный
9. Генератор импульсов
10. Фильтр Аperiodический
11. Отклонения

Для всех логических операций (кроме аperiodического фильтра и отклонения) возможно использование **проверки исполнения по условиям результата**.

Для первых 5-ти типов операций возможно указание **выборки данных**.

A.4.1 AND, OR, XOR, Штрих Шеффера, Стрелка Пирса, Эквивалентность, Следование, НЕ Следование, Обратное Следование, НЕ Обратное Следование

1-й параметр операции:

- = 0 - используются оба операнда;
- = 1 - используется один операнд - 1-й;
- = 2 - используется один операнд - 2-й;

Для параметра 1 = 0,1,2 свертка выполняется по типу операции;

- = 10 - используются оба операнда;
- = 11 - используется один операнд - 1-й;
- = 12 - используется один операнд - 2-й;

Свертка операндов для параметра 1 = 10,11,12 производится по флагу свертки.

Параметры 2 и 3 определяют использование предоперации перед выполнением основной операции в том случае, если заданы через ссылку:

Параметр 2 - если задан через ссылку, то:

после свертки ОП1 производится предоперация:

ОП1 AND/OR ЗнПар2,

тип операции AND/OR определяется флагом свертки ОП1; ЗнПар2 рассматривается как логическое;

Параметр 3 - если задан через ссылку, то:

после свертки ОП2 производится предоперация:

ОП2 AND/OR ЗнПар3,

тип операции AND/OR определяется флагом свертки ОП2; ЗнПар3 рассматривается как логическое.

Для операций Следование, НЕ Следование, Обратное Следование и НЕ Обратное Следование параметры алгоблока игнорируются:

- свертка всегда производится по флагу «И»/«ИЛИ»;
- всегда требуются оба операнда.

Таблица истинности для операций

Операнд1	Ложь	Ложь	Истина	Истина
Операнд2	Ложь	Истина	Ложь	Ложь
<u>И</u>	Ложь	Ложь	Ложь	Истина
<u>ИЛИ</u>	Ложь	Истина	Истина	Истина
<u>Исключающее ИЛИ</u>	Ложь	Истина	Истина	Ложь
<u>Штрих Шеффера (НЕ И)</u>	Истина	Истина	Истина	Ложь
<u>Стрелка Пирса (НЕ ИЛИ)</u>	Истина	Ложь	Ложь	Ложь
<u>Эквивалентность (НЕ Искл.ИЛИ)</u>	Истина	Ложь	Ложь	Истина
<u>Следование</u>	Истина	Истина	Ложь	Истина
<u>Обратное Следование</u>	Истина	Ложь	Истина	Истина

Для операций с префиксом **НЕ** производится инвертирование результата.

В случае бинарной операции над векторами с результатом, определенным матрицей выполняется операция, аналогичная каноническому произведению векторов:

$Рез(i,j) = Оп1(i) \text{ } F \text{ } Оп2(j)$, где

$Рез(i,j)$ – элементы матрицы результата;

$Оп1(i)$, $Оп2(j)$ – элементы векторов операндов;

F – исполняемая операция.

В остальных случаях обработка аналогична случаю бинарных арифметических операций.

Использование флага свертки операндов в алгоблоке

В случае результата - вектор возможно изменение направления свертки $Оп1$ и/или $Оп2$:

Стандартно:

- количество результатов равно минимуму 1-х размерностей $Оп1$ и/или $Оп2$ и $Рез$;
- свертка матриц $Оп1$ и/или $Оп2$ к вектору (сл.3,6,7,8,9) проводится по 2-й размерности;

Инверсно:

- количество результатов равно минимуму 2-х размерностей $Оп1$ и/или $Оп2$ и 2-й размерности $Рез$;
- свертка матриц $Оп1$ и/или $Оп2$ к вектору (сл.3,6,7,8,9) проводится по 1-й размерности.

А.4.2 Операции сравнения с уставками

1. Не за верхней уставкой

2. Не за нижней
3. За нижней
4. За верхней
5. Вне уставок
6. В уставках (норма)
7. Недостоверно
8. Достоверно

Для операций всегда требуются **оба операнда**.

Свертка (при необходимости) производится по флагу «И»/«ИЛИ» результата.

Алгоритм приведения операндов в зависимости от размерностей

Результат - Скаляр

Оба операнда свертываются к скаляру

Результат - Вектор

	Операнд 1	Скаляр	Вектор	Матрица
Операнд 2				
Скаляр		1	2	3
Вектор		4	5	6
Матрица		7	8	9

Сл.1 - Тривиальный случай; высчитывается скаляр, который тиражируется на вектор результата;

Сл.2,4 - Образуется вектор результата - используется вектор одного операнда и скаляр другого для всех элементов вектора;

Сл.3,7 - Матрицы свертываются к вектору - далее, как в сл.2 и 4;

Сл.5 - Тривиальный расчет по минимальным размерностям векторов;

Сл.6,8 - Соответствующая матрица свертывается к вектору, далее, как в сл.5;

Сл.9 - Матрицы свертываются к векторам, далее, как в сл.5.

Результат - Матрица

	Операнд 1	Скаляр	Вектор	Матрица
Операнд 2				
Скаляр		11	12	13
Вектор		14	15	16
Матрица		17	18	19

Сл.11 - Вычисление скаляра с тиражированием результата на матрицу;

Сл.12,14 - Образуется вектор результата - используется вектор одного операнда и скаляр другого для всех элементов вектора - далее результирующий вектор тиражируется на матрицу;

Сл.13,17 - Вычисляется матрица результата по матрице операнда и скаляру (по всем элементам матрицы);

Сл.15: - Расчет по минимальным размерностям векторов с тиражированием на вектора матрицы;
Сл.16,18 - Матрица результата получается обработкой соответствующих векторов матрицы одного операнда и вектора другого;
Сл.19 - Матрицы обрабатываются поэлементно по минимальным размерностям.

Использование флага свертки операндов в алгоблоке

В случае результата - вектор возможно изменение направления свертки Оп1 и/или Оп2:

Стандартно:

- количество результатов равно минимуму 1-х размерностей Оп1 и/или Оп2 и Рез;
- свертка матриц Оп1 и/или Оп2 к вектору проводится по 2-й размерности;

Инверсно:

- количество результатов равно минимуму 2-х размерностей Оп1 и/или Оп2 и 2-й размерности Рез;
- свертка матриц Оп1 и/или Оп2 к вектору проводится по 1-й размерности;

А.4.3 Операции сравнения значений: <=,>=,#,<,>

Для операций всегда требуются **оба операнда**.

Параметр 1 - зона нечувствительности (для операций =,#) или 1-я полуширина гистерезиса (для операций <=,>=,<,>);

Параметр 2

0 - 2-я полуширина гистерезиса;

= 0 - 2-я полуширина = 1-ой полуширине;

Алгоритм сравнения для операций

1. >,>=

- Если Рез'="Ложь", то перед сравнением ОП2=ОП2+mod(par1)
="Истина", то перед сравнением ОП2=ОП2-mod(par2)

2. <,<=

- Если Рез'="Истина", то перед сравнением ОП2=ОП2+mod(par1)
="Ложь", то перед сравнением ОП2=ОП2-mod(par2)

3. =,# - при ненулевой зоне нечувствительности операции преобразуются в операции попадания/не попадания в диапазон соответственно.

(ОП1, ОП2 – операнды; mod() – функция взятия абсолютного значения; Рез' - значение сравнения с предыдущего такта).

А.4.4 Операция отрицания NOT

Операция выполняется аналогично копированию за исключением того, что операнды используются как логические.

Параметр 1 используется для указания дозаписи в результат (если =1).

А.4.5 Попадание в диапазон (логическое)

Операнд 1 - Анализируемые данные;

Операнд 2 - Набор диапазонов;

Матрица рассматривается как набор векторов по 1-й размерности (у каждого вектора может быть "- бесконечность" и "+ бесконечность" по краям).

При попадании в диапазон в соответствующем поле результата устанавливается значение ИСТИНА, при непадании - значение ЛОЖЬ.

При выборке числа и диапазона данные проверяются только по статусам (проверка по векторам не производится).

Параметр1 - режим задания диапазонов:

= 0 - последовательно: каждый элемент является верхней границей предыдущего диапазона и нижней следующего;

= 1 - диапазоны задаются парами чисел.

Если включен учет бесконечностей, то добавляется бесконечность в начало и/или конец (при любом значении параметра 1).

Параметры 2 и 3 - режимы задания границ диапазонов:

В общем виде, количество диапазонов для матрицы $M(l,n)$:

Параметр 2	Параметр 3	Границы	Учет «бесконечностей»		Количество диапазонов	
			«-»	«+»	Параметр 1 = 0	Параметр 1 = 1
0	0	< ... =<	Да	Да	$(L+1)*N$	$((L+2)/2)*N$
1	0	=< ... =<	Да	Да	-"-	-"-
0	1	< ... <	Да	Да	-"-	-"-
1	1	=< ... <	Да	Да	-"-	-"-
2	0	< ... =<	Нет	Да	$L*N$	$((L+1)/2)*N$
3	0	=< ... =<	Нет	Да	-"-	-"-
2	1	< ... <	Нет	Да	-"-	-"-
3	1	=< ... <	Нет	Да	-"-	-"-
0	2	< ... =<	Да	Нет	$L*N$	$((L+1)/2)*N$
1	2	=< ... =<	Да	Нет	-"-	-"-
0	3	< ... <	Да	Нет	-"-	-"-
1	3	=< ... <	Да	Нет	-"-	-"-
2	2	< ... =<	Нет	Нет	$(L-1)*N$	$L/2*N$
3	2	=< ... =<	Нет	Нет	-"-	-"-
2	3	< ... <	Нет	Нет	-"-	-"-
3	3	=< ... <	Нет	Нет	-"-	-"-

Если граница определена некорректно – операция проверки производится и:

- Если нет свертки одного операнда по границам: ставится статус недоверности 5;
- Если есть свертка одного операнда по границам:
 - «И»:
 - если объединенный результат - **Ложь**:
 - если при этом есть хотя бы один корректный диапазон, для которого значение попадания - **Ложь**, то пишется достоверный статус 0;
 - во всех остальных случаях - недоверный статус 5;
 - если объединенный результат - **Истина**:
 - если при этом есть хотя бы один некорректный диапазон, то пишется недоверный статус 5;
 - во всех остальных случаях - достоверный статус 0;
 - «ИЛИ»:
 - если объединенный результат - **Ложь**:
 - во всех случаях пишется недоверный статус 5;
 - если объединенный результат - **Истина**:
 - если при этом есть хотя бы один корректный диапазон с результатом **Истина**, то пишется достоверный статус 0;
 - во всех остальных случаях - недоверный статус 5;

Если анализируемый операнд недоверен - с ним операций не производится:

- Во всех случаях результату присваивается значение **Ложь** и статус 5, кроме случаев:
 - Результаты сравнения нескольких операндов по одному диапазону объединялись по «ИЛИ» и было найдено хотя бы одно попадание в диапазон - результат - **Истина** со статусом 0;
 - Результаты сравнения нескольких операндов по одному диапазону объединялись по «И» и было найдено хотя бы одно не попадание в диапазон - результат **Ложь** со статусом 0.

Алгоритмы анализа в зависимости от размерностей операндов и результата.

Так как количество диапазонов, определяемых операндом 2, может не соответствовать размерности, то все дальнейшие алгоритмы приведены к случаям одного диапазона (скаляр), N диапазонов, заданных вектором (вектор) и N*M диапазонов, определяемых матрицей операнда (матрица).

Тип результата: Скаляр

	Операнд 1			
Операнд 2 (ДП)		Скаляр	Вектор	Матрица
Скаляр		1	2	3
Вектор		4	5	6
Матрица		7	8	9

Сл.1 - тривиальная проверка одного операнда по одному диапазону;

Сл.2,3 - проверка всех операндов по одному диапазону, результат формируется объединением (сверткой) по «И»/«ИЛИ» в зависимости от значения флага свертки, установленного в условии операнда 1 (флаг переключается по значениям «И»/«ИЛИ», для его использования необходимо также включить общий флаг использования условий операнда 1 - иначе значение флага неопределено !);

Сл.4,7 - проверка одного операнда по всем диапазонам, результат формируется модифицированной сверткой:

- для последовательно заданных границ (Параметр 1 = 0) - всегда по «ИЛИ»;
- для парно заданных границ (Параметр 1 =1) - по флагу свертки «И»/«ИЛИ»;

Сл.5,9 - каждое значение проверяется по своему диапазону, результат формируется сверткой по флагу «И»/«ИЛИ»;

Сл.6 - каждый элемент вектора матрицы операнда проверяется по соответствующему ему диапазону из вектора диапазонов; результат образуется сверткой по флагу «И»/«ИЛИ»;

Сл.8 - каждый элемент вектора значений проверяется по всем возможным диапазонам (результат образуется модифицированной сверткой); далее полученные результаты для каждого значения операнда сворачиваются по флагу «И»/«ИЛИ», образуя скалярный результат.

Тип результата: Вектор B(R)

	Операнд 1	Скаляр	Вектор B(K)	Матрица M(K,S)
Операнд 2 (ДП)				
Скаляр		11	12	13
Вектор B(L)		14	15	16
Матрица M(L,N)		17	18	19

Сл.11 - аналог сл.1 с добавлением тиражирования скалярного результата на весь результирующий вектор;

Сл.12 - поэлементная проверка каждого элемента операнда по единственному диапазону; размерность результата: Min(R,K);

Сл.14 - поддиапазонная проверка единственного операнда по всем диапазонам; размерность результата: Min(R,L);

Сл.15 - каждый элемент операнда 1 проверяется по своему диапазону; размерность результата: Min(R,K,L);

Сл.13,16 - каждый вектор операнда проверяется по одному диапазону (в Сл.13 диапазон общий) и сворачивается по флагу «И»/«ИЛИ»; размерность результата: Min(R,S); размерность внутренней проверки: Min(L,K) для Сл.16 или K для Сл.13;

Сл.17 - поддиапазонная проверка операнда по вектору диапазонов (результат образуется модифицированной сверткой); размерность результата: Min(R,N); размерность внутренней проверки - L;

Сл.18 - каждый элемент вектора значений проверяется по вектору диапазонов (далее используется модифицированная свертка); размерность результата: Min(R,K,N); размерность внутренней проверки - L;

Сл.19 - каждый элемент матрицы значений проверяется по своему диапазону из матрицы диапазонов, результат образуется сворачиванием по флагу «И»/«ИЛИ»; размерность результата: Min(R,S,N); размерность внутренней проверки: Min(L,K);

Тип результата: Матрица M(R,T)

	Операнд 1	Скаляр	Вектор B(K)	Матрица M(K,S)
Операнд 2 (ДП)				
Скаляр		21	22	23

Вектор В(L)	24	25	26
Матрица М(L,N)	27	28	29

Сл.21 - аналог сл.1 с добавлением тиражирования скалярного результата на всю результирующую матрицу;

Сл.22 - аналог сл.12 с тиражированием вектора на всю результирующую матрицу;

Сл.23 - поэлементная проверка матрицы значений по единственному диапазону;

Сл.24 - аналог сл.14 с тиражированием векторов на всю результирующую матрицу;

Сл.27 - поддиапазонная проверка скаляра по матрице диапазонов;

Сл.25 - вектора результирующей матрицы образуются проверкой каждого элемента из вектора значений по всем диапазонам из вектора диапазонов; размерность результата: Min(L,R),Min(K,T);

Сл.26 - каждый элемент вектора матрицы значений проверяется по своему диапазону из вектора диапазонов; размерность результата: Min(K,L,R),Min(S,T);

Сл.28 - каждый элемент вектора значений проверяется по своему диапазону из матрицы диапазонов; размерность результата: Min(K,L,R),Min(T,N).

Сл.29 - поэлементная проверка каждого элемента матрицы значений на своем диапазоне из матрицы диапазонов.

A.4.6 Селектор логический

Операнд 1 - Источник выбора.

Операнд 2 - Признак (маска) выбора.

В **результат** копируются те данные из операнда 1, для которых соответствующие им данные из операнда 2 равны **Истина** (#0).

Если операнд 2 - недостоверен (достоверность проверяется только по статусу), выборки не производится.

Для операнда 1 никаких проверок не проводится, а копируется операнд со своим статусом.

Операнд 2 всегда рассматривается как логический.

Операнд 1 всегда рассматривается как имеющий тип результата.

В тех случаях (отмечены символом '*'), когда необходимо понизить размерность Операнда 1:

- для логического типа результата используется свертка по «И»/«ИЛИ» в зависимости от флага;
- для числового и символьного типов используется сложение (для символьного по модулю 255).

Параметр 1 - приоритет выбора из операнда 2 (используется в случаях, когда необходимо понизить размерность выборки до размерности результата - далее отмечены символом '*'):

= 0 - с начала операндов (для поиска значений **Истина** операнд 2 просматривается с начала);

= 1 - с конца операндов (для поиска значений **Истина** операнд 2 просматривается с конца).

Алгоритмы выборки в соответствии с размерностями операндов и результата

Результат - Скаляр

	Операнд 1 – источник	Скаляр	Вектор	Матрица
Операнд 2 – маска выбора				
Скаляр		1	2	3
Вектор		4	5	6
Матрица		7	8	9

Сл.1 Копирование, если Оп2=**Истина**.

Сл.2 и 3 Копирование, если Оп2=**Истина**. Перед копированием производится свертка Оп1 (**).

Сл.4 и 7 Свертка Оп2 по флагу «И»/«ИЛИ» к скаляру и далее как по Сл.1.

Сл.5 Основной случай. Копирование в результат того элемента вектора источника, для которого соответствующий элемент маски в Оп2=**Истина**; порядок просмотра Оп2(приоритет) определяется параметром 1 операции (*).

Сл.6 Аналог сл.5 - перед копированием - свертка Оп1 (**) к вектору.

Сл.8 Свертка матрицы Оп2 по флагу «И»/«ИЛИ» к вектору и далее аналогично случаю 5.

Сл.9 Полный аналог сл.5 за исключением того, что просматриваются матрицы.

Результат - Вектор

	Операнд 1 – источник	Скаляр	Вектор	Матрица
Операнд 2 – маска выбора				
Скаляр		11	12	13
Вектор		14	15	16
Матрица		17	18	19

Сл.11 Аналогичен сл.1 с тиражированием результата на вектор.

Сл.12 Копирование вектора Оп1 в результат, если Оп2=**Истина**.

Сл.13 Матрица Оп1 сворачивается по (*) к вектору и далее аналогично сл.12.

Сл.14 Копирование скаляра Оп1 в результирующий вектор по тем индексам, для которых элементы вектора Оп2 имеют значение **Истина**.

Сл.15 Аналог сл.14, но копируется не скаляр, а соответствующие элементы вектора Оп1.

Сл.16 Из матрицы Оп1 выбирается вектор по значениям соответствующих элементов вектора Оп2 и копируется в результат; просмотр вектора Оп2 определяется параметром 1 операции (**).

Сл.17 Матрица Оп2 сворачивается к вектору по флагу «И»/«ИЛИ», далее действия аналогичны сл.14.

Сл.18 Матрица Оп2 сворачивается к вектору по флагу «И»/«ИЛИ», далее действия аналогичны сл.15.

Сл.19 Каждый элемент результирующего вектора получается как результат обработки соответствующих векторов Оп1 и Оп2(см.сл.5).

Результат - Матрица

	Операнд 1 – источник	Скаляр	Вектор	Матрица
Операнд 2 – маска выбора				

Скаляр	21	22	23
Вектор	24	25	26
Матрица	27	28	29

Сл.21 Аналогичен сл.1 с тиражированием результата на матрицу.

Сл.22 Аналогичен сл.12 с тиражированием результата на матрицу.

Сл.23 Копирование матрицы, если Оп2=**Истина**.

Сл.24 Аналогичен сл.14 с тиражированием.

Сл.25 Для всех элементов вектора Оп2=**Истина** производится копирование всего вектора Оп1 в соответствующий вектор матрицы результата.

Сл.26 Для всех элементов вектора Оп2=**Истина** производится копирование соответствующего вектора Оп1 в соответствующий вектор матрицы результата.

Сл.27 Скаляр Оп1 копируется по маске матрицы Оп2(=**Истина**) в соответствующие элементы матрицы результата.

Сл.28 Аналогичен сл.27, только копирование производится из вектора.

Сл.29 Производится копирование матрицы Оп1 в матрицу результата по маске из матрицы Оп2.

Использование флага свертки операндов в алгоблоке

Для случаев MBC и MCB возможно изменение направления свертки Оп1.

Стандартно:

- количество вычислений равно минимуму 1-х размерностей Оп1 и Оп2 (MBC) или Оп1 и Рез (MCB);
- свертка Оп1 проводится по 2-й размерности операнда;

Инверсно:

- количество вычислений равно минимуму 2-й размерности Оп1 и 1-й размерности Оп2 (MBC) или 2-й размерности Оп1 и 1-й размерности Рез (MCB);
- свертка Оп1 проводится по 1-й размерности операнда;

А.4.7 Триггер

Алгоблок реализует триггеры: JK,RS,T,D.

Триггеры могут иметь различные типы входов:

- реагирующие на значение входного сигнала (потенциальные);
- реагирующие на изменение значения входного сигнала (импульсные), в том числе:
 - изменение с 0 до 1;
 - изменение с 1 до 0;
 - любое изменение значения входного сигнала.

RS-триггер представлен в 2-х модификациях в соответствии со значением выхода для запрещенной комбинации входов 1-1:

RS-0 - устанавливается значение 0;

RS-1 - устанавливается значение 1.

Параметр 1 - тип триггера:

= 0 – JK по фронту 0 -> 1

1 - RS-0 по фронту 0 -> 1

2 - RS-1	по фронту 0 -> 1
3 - T	по фронту 0 -> 1
4 - D	по фронту 0 -> 1
5 - JK	по фронту 1 -> 0
6 - RS-0	по фронту 1 -> 0
7 - RS-1	по фронту 1 -> 0
8 - T	по фронту 1 -> 0
9 - D	по фронту 1 -> 0
10 - JK	по любому фронту
11 - RS-0	по любому фронту
12 - RS-1	по любому фронту
13 - T	по любому фронту
14 - D	по любому фронту

Параметр 2 - ссылка на область сохранения значений 1-х операндов JK/RS/T/D-триггеров - J,R,T,D соответственно.

Параметр 3 - ссылка на область сохранения значений:

- 2-х операндов для JK и RS-триггеров - K и S соответственно;
- наличие импульса для D-триггера.

Если ссылка для соответствующего входа задана, т.е. источник соответствующего параметра не является константой и значение параметра не равно 0, то такой вход является импульсным в соответствии со значением параметра 1; если же ссылка не задана, то вход является потенциальным.

Для D-триггера:

- если задана одна ссылка - триггер - потенциальный;
- если обе (параметры 2 и 3)- импульсный.

В одном триггере (типа JK- RS-) допустима любая комбинация типов входов.

T-триггер является импульсным, если определена ссылка для параметра 2. Если ссылка не определена, триггер является потенциальными.

Количество триггеров определяется размерностями результата.

Операнд 1 - вход в зависимости от типа триггера:

- J для JK-триггера;
- R для RS-триггера;
- T для T-триггера;
- D для D-триггера.

Операнд 2 - вход в зависимости от типа триггеров:

- K для JK-триггера;
- S для RS-триггера.

Операнды всегда приводятся к типу и размеру результата:

- сворачиванием по флагу «И»/«ИЛИ» для уменьшения размерностей;
- тиражированием для увеличения размерностей.

В области сохранения предыдущих значений входов (область, заданная через параметры 2 и 3 для JK- и RS-триггеров, параметр 2 для T- и D-триггера) записывается приведенное к размерности результата значение. Для D-триггера в области, определенной параметром 3 фиксируется наличие импульса на входе. Размер области, выделенной через параметры 2 и 3, должен соответствовать размерности результата, т.е. количеству используемых триггеров.

Таблицы истинности

JK-триггер

J	K	Q-Выход
0	0	Q'
0	1	0
1	0	1
1	1	NOT Q'

RS-триггер

R	S	Q-Выход
0	0	Q'
0	1	1
1	0	0
1	1	0 или 1

T-триггер

T	Q-Выход
0	Q'
1	NOT Q'

D-триггер

D	Q-Выход
0	D'
1	D'

, где:

Q' - значение выхода на предыдущем такте

D' - значение входа D-триггера на предыдущем такте

0,1 - для потенциального входа:

- значение входа

- для импульсного входа:

0 - отсутствие изменения входа (отсутствие импульса);

1 - наличие требуемого изменения входа (наличие импульса).

Использование флага свертки операндов в алгоблоке

Возможно изменение направления свертки Оп1 и Оп2.

Стандартно:

- свертка Оп1 и Оп2 проводится по 2-й размерности операнда;

Инверсно:

- свертка Оп1 и Оп2 проводится по 1-й размерности операнда;

А.4.8 Фильтр дискретный

Операнд 1 - фильтруемая величина (рассматривается как логическая);

Операнд 2 - буфер побитного хранения предыдущих значений операнда 1; глубина буфера определяется размером элемента операнда 2:

integer*1(8 бит) - 8 предыдущих значений операнда 1,

integer*2(16 бит) - 16 значений и т.д.

Операнд 2 не может быть константой.

Параметр 1 - режим работы:

= 0 – фильтр;

= 1 - звено запаздывания;

Параметр 2 - для фильтра: режим задания приоритета при равенстве "0" и "1":

= 0 - приоритет определяется предыдущим значением результата;

= 1 - приоритет определяется значением параметра 3.

- для звена запаздывания: глубина выборки - определяет выбираемое предыдущее значение:

= 0 - текущее значение(значение операнда 1с текущего такта);

= 1 - предыдущее значение операнда 1и т.д. (ограничивается глубиной буфера).

Параметр 3:

- для фильтра:

- приоритет выбора при равенстве "0" и "1" - 0 или 1, а также начальное значение буфера;

- для звена запаздывания:

- начальное значение буфера;

При старте весь буфер прописывается значением, определяемым параметром 3.

Все необходимые свертки результата: вектора в скаляр (см. ниже сл.14,17,24), матрицы в скаляр (см. ниже сл.27), матрицы в вектор (см. ниже сл.28) проводятся по флагу операнда 2 «И»/«ИЛИ».

Алгоритм работы

Фильтр

1. Подсчет количества "0" и "1" в буфере с учетом текущего значения операнда 1 и результата с предыдущего такта:

- если больше "1", на выход "1";

- если больше "0", на выход "0";

- если поровну, на выход:

- или предыдущее значение результата;

- или значение, определяемое параметром 3;

2. Сохранение значения операнда 1 в буфере;

Звено запаздывания

1. На выход из буфера (операнд 2) подается значение, определяемое глубиной выборки (параметр 2);

2. Сохранение значения операнда 1 в буфере.

При сохранении значений производится перемещение указателя буфера, сами данные не перемещаются.

Определяющей размерностью является размерность буфера (операнда 2).

Приведение операнда 1 и результата в зависимости от размерности операнда 2.

Операнд 2 - скаляр

	Результат	Скаляр C3	Вектор V3(n3)	Матрица M3(n3,m3)
Операнд 1				
Скаляр C1		1	2	3
Вектор V1(n1)		4	5	6
Матрица M1(n1,m1)		7	8	9

Сл.1 Тривиальный

Сл.2 и 3 Аналог сл.1; скалярный результат тиражируется на вектор или матрицу результата;

Сл.4 и 7 Операнд 1 свертывается к скаляру; далее аналог сл.1;

Сл.5,6,8,9 Операнд 1 свертывается к скаляру; далее аналог сл.1 с тиражированием результата.

Операнд 2 - вектор V2(n2)

	Результат	Скаляр C3	Вектор V3(n3)	Матрица M3(n3,m3)
Операнд 1				
Скаляр C1		11	12	13
Вектор V1(n1)		14	15	16
Матрица M1(n1,m1)		17	18	19

Сл.11 Используется элемент V2(1); аналог сл.1;

Сл.14 Работа по $\min(n1,n2)$; векторный результат свертывается к скаляру;

Сл.17 Матрица операнда 1 свертывается к вектору; далее аналог сл.14;

Сл.12 и 13 Используется V2(1) для хранения; скалярный результат тиражируется на вектор или матрицу;

Сл.15 Работа по $\min(n1,n2,n3)$

Сл.18 Матрица операнда 1 свертывается к вектору; далее аналог сл.15;

Сл.16 Аналог сл.15 с тиражированием вектора результата на матрицу;

Сл.19 Матрица операнда 1 свертывается к вектору; далее аналог сл.16.

Операнд 2 - матрица M2(n2,m2)

	Результат	Скаляр C3	Вектор V3(n3)	Матрица M3(n3,m3)
Операнд 1				
Скаляр C1		21	22	23
Вектор V1(n1)		24	25	26
Матрица M1(n1,m1)		27	28	29

Сл.21 Используется M2(1,1); аналог сл.1

Сл.24 Работа по $\min(n1,n2)$; векторный результат свертывается в скаляр;

Сл.27 Работа по $\min(n1,n2)$ и $\min(m1,m2)$; матричный результат свертывается в скаляр;

Сл.22 и 23 Используется $M2(1,1)$ для хранения; скалярный результат тиражируется на вектор или матрицу;
 Сл.25 Работа по $\text{Min}(n1,n2,n3)$; для хранения используется 1-й вектор операнда 2;
 Сл.26 Аналог сл.25 с тиражированием вектора результата на матрицу;
 Сл.28 Работа по $\text{Min}(n1,n2,n3)$ и $\text{Min}(m1,m2)$; Матрица результата сворачивается к вектору;
 Сл.29 Работа по $\text{Min}(n1,n2,n3)$ и $\text{Min}(m1,m2,m3)$.

Использование флага свертки операндов в алгоблоке

Для случаев MBC, MBV и MBM возможно изменение направления свертки Оп1.
 Стандартно:

- количество результатов равно:
 - для MBV и MBM минимуму 1-х размерностей Оп1, Оп2 и Рез;
 - для MBC минимуму 1-х размерностей Оп1 и Оп2;
- свертка Оп1 проводится по 2-й размерности;

Инверсно:

- количество результатов равно:
 - для MBV и MBM минимуму 2-й размерности Оп1 и 1-х размерностей Оп2 и Рез;
 - для MBC минимуму 2-й размерности Оп1 и 1-й размерности Оп2;
- свертка Оп1 проводится по 1-й размерности;

А.4.9 Генератор импульсов

Операнд 1 - инициатор генерации импульсов.

Параметр 1 - тип генератора:

- = 0 – одновибратор;
- = 1 (#0) – мультивибратор;

Параметр 2 - режим запуска (в зависимости от типа генератора):

- для одновибратора (генерация включается по фронтам операнда 1):
 - генерация однократных импульсов:
 - = 0 - генерация импульса по фронту операнда 1: 0 -> 1;
 - = 1 - генерация импульса по фронту операнда 1: 1 -> 0;
 - = 2 - генерация импульса по любому фронту операнда 1: 0 -> 1 или 1 -> 0;
 - генерация импульсов длительностью, определяемой операндом 1:
- = 3 - генерация импульса:
 - начало импульса по фронту 0 -> 1;
 - завершение импульса по фронту 0 -> 1;
- = 4 - генерация импульса:
 - начало импульса по фронту 1 -> 0;
 - завершение импульса по фронту 1 -> 0;
- для мультивибратора импульсы (генерация включается потенциально по значению операнда 1):
 - = 0 - генерация однократных импульсов по значению операнда 1 = **Ложь**;
 - = 1 (#0) - генерация однократных импульсов по значению операнда 1 = **Истина**;

Однотактный импульс - импульс начинается и завершается на следующих друг за другом проходах данного алгоблока.

Для одновибратора требуется сохранение значений операнда 1 - для этого используется операнд 2. Для мультивибратора **операнд 2 не используется**.

Приведение размерностей операндов и результата

1. Одновибратор

Определяющей является размерность операнда 2.

Операнд 2 - Скаляр

	Результат	Скаляр С3	Вектор В3(п3)	Матрица М3(п3,л3)
Операнд 1				
Скаляр С1		1	2	3
Вектор В1(п1)		4	5	6
Матрица М1(п1,л1)		7	8	9

Во всех случаях, где операнд 1 не скаляр - он свертывается к скаляру по флагу «И»/«ИЛИ».
В сл.2 и 3 производится тиражирование результирующего скаляра на вектор и матрицу результата соответственно.

Операнд 2 - Вектор В2(п2)

	Результат	Скаляр С3	Вектор В3(п3)	Матрица М3(п3,л3)
Операнд 1				
Скаляр С1		11	12	13
Вектор В1(п1)		14	15	16
Матрица М1(п1,л1)		17	18	19

Сл.11 - тривиальный (для хранения используется только В2(1))

Сл.12 и 13 - аналогичны сл.11 с тиражированием результирующего скаляра на вектор и матрицу соответственно

Сл.15 - общий случай: работа по минимальным размерностям (N1,N2,N3)

Сл.16 - аналог сл.15 с тиражированием вектора результата на все вектора матрицы

Сл.18 - свертка операнда 1 к вектору по 2-й размерности; далее аналог сл.15

Сл.19 - свертка операнда 1 к вектору по 2-й размерности; далее аналог сл.16

Сл.14 и 17 - свертка операнда 1 к скаляру; для запоминания используется 1-й элемент вектора операнда 2

Операнд 2 - Матрица М2(п2,л2)

	Результат	Скаляр С3	Вектор В3(п3)	Матрица М3(п3,л3)
Операнд 1				
Скаляр С1		21	22	23

Вектор $V1(n1)$	24	25	26
Матрица $M1(n1,l1)$	27	28	29

Сл.21 - тривиальный

Сл.22 и 23 - аналог сл.21 с тиражированием скаляра на вектор и матрицу соответственно

Сл.25 - работа по минимальным размерностям ($N1, N2, N3$)

Сл.29 - общий случай: работа по минимальным размерностям: ($N1, N2, N3$) и ($L1, L2, L3$)

Сл.26 - аналог сл.25 с тиражированием результирующего вектора на матрицу

Сл.28 - сворачивание матрицы операнда 1 к вектору по 2-й размерности; далее аналог сл.25

Сл.24 и 27 - свертка операнда 1 к скаляру (запоминание в 1-м элементе операнда 2); далее аналог сл.21

Использование флага свертки операндов в алгоблоке

Для случаев $BBV, MBV, BBM, MBM, BMB, MMB, BMM$ возможно указание инверсного направления свертки $Op1$.

Стандартно:

- количество элементов результата равно минимуму 1-х размерностей $Op1$, $Op2$ и $Рез$;
- свертка $Op1$ из матрицы к вектору проводится по 2-й размерности;

Инверсно:

- количество элементов результата равно минимуму 2-й размерности $Op1$ и 1-х размерностей $Op2$ и $Рез$;
- свертка $Op1$ из матрицы к вектору проводится по 1-й размерности.

2. Мультивибратор

	Результат	Скаляр $C3$	Вектор $V3(n3)$	Матрица $M3(n3,l3)$
Операнд 1				
Скаляр $C1$		1	2	3
Вектор $V1(n1)$		4	5	6
Матрица $M1(n1,l1)$		7	8	9

Сл.1 - тривиальный

Сл.2 и 3 - аналог сл.1 с тиражированием скалярного результата на вектор и матрицу соответственно

Сл.4 и 7 - свертка вектора и матрицы к скаляру по флагу «И»/«ИЛИ» операнда 1 ; далее аналог сл.1

Сл.5 и 9 - работа по соответствующим минимальным размерностям

Сл.6 - аналог сл.5 с тиражированием вектора результата на все вектора матрицы

Сл.8 - Свертка матрицы операнда 1 к вектору по 2-й размерности по флагу операнда 1 «И»/«ИЛИ»; далее аналог сл.5

А.4.10 Фильтр Апероидический

Операнд 1 - источник данных (рассматриваются как логические);

Операнд 2 - значение апероидики (вещественное);

Результат – логический.

Параметр 1 - постоянная времени (или вектор постоянных времени, если задан ссылкой);

Параметр 2 - размерность вектора постоянных времени, определенного Параметром 1

Параметр 3 - приоритет обработки: принимает значения $<0>$, $<1>$, $<\text{любое другое}>$

Алгоритм вычисления результата

1. $Op2 = Op1 * Dt / T + Op2' * (1 - Dt / T)$,

где:

Op2 - значение операнда 2;

Op1 - значение операнда 1;

Dt - время между последовательными вызовами алгоблока;

T - постоянная времени;

Op2' - предыдущее значение Op2.

2. Если ПЗ=0:

- если $Op2 \geq 0.64$, то Рез=**Истина**

$Op2 < 0.64$, то Рез=**Ложь**

Если $(Op2 > Op2')$ и $(Op2 \geq 0.64)$ и $(Op2' < 0.64)$ (пересечение границы 0.64 вверх), то $Op2 = 1.0$

Если $(Op2 < Op2')$ и $(Op2 \leq 0.64)$ и $(Op2' > 0.64)$ (пересечение границы 0.64 вниз), то $Op2 = 0.0$

Если ПЗ=1:

- если $Op2 \geq 0.36$, то Рез=**Истина**

$Op2 < 0.36$, то Рез=**Ложь**

Если $(Op2 > Op2')$ и $(Op2 \geq 0.36)$ и $(Op2' < 0.36)$ (пересечение границы 0.36 вверх), то $Op2 = 1.0$

Если $(Op2 < Op2')$ и $(Op2 \leq 0.36)$ и $(Op2' > 0.36)$ (пересечение границы 0.36 вниз), то $Op2 = 0.0$

Если ПЗ=<любой другой>:

- если $Op2 > Op2'$, то:

- если $Op2 \geq 0.64$, то Рез=**Истина**

$Op2 < 0.64$, то Рез=**Ложь**

- если $Op2 < Op2'$, то:

- если $Op2 \geq 0.36$, то Рез=**Истина**

$Op2 < 0.36$, то Рез=**Ложь**

- если $Op2 = Op2'$, то Рез НЕ меняется

Если $(Op2 > Op2')$ и $(Op2 \geq 0.64)$ и $(Op2' < 0.64)$ (пересечение границы 0.64 вверх), то $Op2 = 1.0$

Если $(Op2 < Op2')$ и $(Op2 \leq 0.36)$ и $(Op2' > 0.36)$ (пересечение границы 0.36 вниз), то $Op2 = 0.0$

Структурирование размерностей

P - количество постоянных времени (определяется параметром 2, если параметр 1 определен ссылкой).

1. Операнд 2 - скаляр C2

	Операнд 1	Скаляр C1	Вектор	Матрица
--	-----------	-----------	--------	---------

Результат			$V1(n1)$	$M1(n1,m1)$
Скаляр $C3$		1	2	3
Вектор $V3(n3)$		4	5	6
Матрица $M3(n3,m3)$		7	8	9

2. Операнд 2 - вектор $V2(n2)$

	Операнд 1	Скаляр $C1$	Вектор $V1(n1)$	Матрица $M1(n1,m1)$
Результат				
Скаляр $C3$		11	12	13
Вектор $V3(n3)$		14	15	16
Матрица $M3(n3,m3)$		17	18	19

3. Операнд 2 - матрица $M(n2,m2)$

	Операнд 1	Скаляр $C1$	Вектор $V1(n1)$	Матрица $M1(n1,m1)$
Результат				
Скаляр $C3$		21	22	23
Вектор $V3(n3)$		24	25	26
Матрица $M3(n3,m3)$		27	28	29

Случай	Количество фильтров	Свертка ОП1 к Ск	Свертка ОП1 к Вк	Тираж СкРез на Вк/Мт	Тираж ВкРез на Мт	Свертка ВкРез в Ск	Свертка МтРез в Ск	Свертка МтРез в Вк
1	$1 = \min(n_1, n_2, n_3)$	-	-	-	-	-	-	-
2	“-	+	-	-	-	-	-	-
3	“-	+	-	-	-	-	-	-
4	“-	-	-	+	-	-	-	-
5	“-	+	-	+	-	-	-	-
6	“-	+	-	+	-	-	-	-
7	“-	-	-	+	-	-	-	-
8	“-	+	-	+	-	-	-	-
9	“-	+	-	+	-	-	-	-
11	$\min(n_2, P)$	-	-	-	-	$+(P>1);$ $-(P=1);$	-	-
12	$\min(n_1, n_2)$	-	-	-	-	+	-	-
13	$\min(n_1, n_2)$	-	+	-	-	+	-	-
14	$\min(n_2, n_3, P)$	-	-	$+(P=1);$ - $(P>1);$	-	-	-	-
15	$\min(n_1, n_2, n_3)$	-	-	-	-	-	-	-
16	$\min(n_1, n_2, n_3)$	-	+	-	-	-	-	-
17	$\min(n_2, n_3, P)$	-	-	$+(P=1: C > M);$ $(P>1: B > M);$	-	-	-	-
18	$\min(n_1, n_2, n_3)$	-	-	-	+	-	-	-
19	$\min(n_1, n_2, n_3)$	-	+	-	+	-	-	-
21	$\min(n_2, P)$	-	-	-	-	$+(P>1);$ $-(P=1);$	-	-
22	$\min(n_1, n_2) * \min(m_2, P)$	-	-	-	-	$+(P=1);$	$+(P>1);$	-
23	$\min(n_1, n_2) * \min(m_1, m_2, P)$, если $P>1$	-	-	-	-	-	+	-
24	$\min(n_2, n_3, P)$	-	-	$+(P=1);$ - $(P>1);$	-	-	-	-
25	$\min(n_1, n_2, n_3) * \min(m_3, P)$	-	-	-	-	-	-	$-(P=1);$ $+(P>1: M > B);$
26	$\min(n_1, n_2, n_3) * \min(m_1, m_2, P)$, если $P>1$	-	-	-	-	-	-	+
27	$\min(n_2, n_3, P)$	-	-	$+(P=1: C > M);$ $(P>1: B > M);$	-	-	-	-

28	Min(n1,n2,n3)* Min(P,m2,m3)	-	-	-	+(P=1); -(P=0)	-	-	-
29	Min(n1,n2,n3)* Min(m1,m2,m3, P), если P>1	-	-	-	-	-	-	-

Свертка результата (где необходимо) производится по флагу «И»/«ИЛИ» Операнда 2.

Использование флага свертки операндов в алгоблоке

Для случаев 13,16,19 возможно изменение направления свертки Оп1.

Стандартно:

- количество результатов равно:

- для случая 13 минимуму 1-х размерностей Оп1 и Оп2;
- для случаев 16 и 19 минимуму 1-х размерностей Оп1, Оп2 и Рез;

- свертка Оп1 проводится по 2-й размерности;

Инверсно:

- количество результатов равно:

- для случая 13 минимуму 2-й размерности Оп1 и 1-й размерности Оп2;
- для случаев 16 и 19 минимуму 2-й размерности Оп1 и 1-х размерностей Оп2 и Рез;

- свертка Оп1 проводится по 1-й размерности;

А.4.11 Отклонения

Вычисление отклонений значений от запомненных. При наличии отклонения на выход записывается "Истина", иначе "Ложь".

Использование параметров операции:

Параметр 1 - зона нечувствительности Delta

- для Пар3=0 - Delta - абсолютное значение в физических единицах;
- для Пар3=1,2 - Delta - процент в диапазоне 0.0 - 1.0.

Параметр 2 - режим использования зоны нечувствительности:

- = 0 - $Zn = Abs(delta)$ - абсолютное значение;
- = 1 - $Zn = Delta$.

Параметр 3 - формула анализа:

- = 0 - абсолютное изменение:
 $RZ > Zn$
- = 1 - процентное изменение от шкалы сигнала
 $RZ/Sh > Zn$
- = 2 - процентное изменение от текущих значений
 $RZ/R > Zn$, где:

$RZ = Abs(Тек - Пред)$ для Пар2=0

$RZ = Тек - Пред$ для Пар2=1

Тек - текущее значение;

Пред - предыдущее значение;

Zn - зона нечувствительности:

$Z_n = \text{Abs}(\text{delta})$ для Пар2=0
 $Z_n = \text{Delta}$ для Пар2=1
 Sh - диапазон сигнала
 (если не задан, то расчет
 производится по Пар3=2);
 $R = (\text{abs}(\text{Тек}) - \text{abs}(\text{Пред})) / 2$ - среднее
 абсолютных значений;

Операнд 1 - анализируемые данные;

Операнд 2 - область сохранения операнда 1 (должна соответствовать по структуре операнду 1);

Результат – логический.

- Сохранение в операнде 2 производится только, если обнаружено отклонение.

Для БД AD, AO, RP, RR также может осуществляться проверка по статусу:

- если текущий и запомненный статусы отличаются - есть отклонение.

Для включения проверки по статусу:

- включить общий флаг проверки операнда 1;
- включить проверку на норму.

Для БД LR зона нечувствительности игнорируется.

Структурирование размерностей

Совместная размерность ОП1 и ОП2 : ОП12

	Операнд 1	Скаляр C1	Вектор B1(n1)	Матрица M1(n1,m1)
Операнд 2				
Скаляр C2		1	2	3
Вектор B2(n2)		4	5	6
Матрица M2(n2,m2)		7	8	9

ОП12:

Сл.1,5,8 - C, B, M соответственно по минимальным размерностям n1, m1, n2, m2;

Сл.2,3,4,7 – C;

Сл.6,8 - B(min(n1,n2));

Формирование результата

	Результат	Скаляр C3	Вектор B3(n3)	Матрица M3(n3,m3)
ОП12				
Скаляр C12		1	2	3

Вектор B12(n12)	4	5	6
Матрица M12(n12,m12)	7	8	9

- Сл.1,5,9- тривиальные случаи по минимальным размерностям;
Сл.2,3 - результат формируется в 1-й элемент (в остальные - тиражирование);
Сл.4,7 - результат формируется в 1-й элемент;
Сл.6 - анализируется вектор по $\min(n12, n3)$ с тиражированием на вектора матрицы результата;
Сл.8 - анализ векторов размерностью n12; кол-во векторов: $\min(m12, n3)$;

Для случаев 4,7,8 при обнаружении отклонения хотя бы одного элемента производится сохранение текущих значений всего операнда 1.

Использование флага свертки операндов в алгоблоке

Для случая 8 возможно изменение количества векторов.

Стандартно: минимум 1-й размерности результата и 2-й размерности объединенного операнда.

Инверсно: минимум 1-й размерности результата и 1-й размерности объединенного операнда.

А.4.12 Квити́рование

Алгоблок предназначен для квитирования аварийных ситуаций.

Аварийные ситуации могут возникать:

- при изменении значения дискретных сигналов:
(как 0->1, так и 1->0);
- при изменении статуса аналоговых сигналов:
(0,1,2,6 -> 3,4,5,7).

Операнд 1 и 2 - анализируемые параметры; допустимы проверки по статусам и выборка операндов.

Результат - результат анализа (см.ниже); допустима проверка по условиям результата.

Параметр 1 - квитирующий сигнал (КС) (номер определяется значением параметра, БД - указанной БД); если номер задан отрицательным числом, то КС НЕ сбрасывается в 0.

Параметры 2 и 3 - устанавливаемые значения в результате анализа (см. ниже).

Квити́рующий сигнал:

- если не задан - считается, что имеет значение 0
- - если ≤ 0 - произвести анализ данных;
- - если > 0 и $КС' > 0$ - сброс в 0 (если возможно) и анализ;
- - если > 0 - выполнить:
 - для дискретных операндов присвоить в результат значение дискрета;
 - для аналоговых операндов:

- если текущий статус = 0,1,2,6 - записать 0;
- если текущий статус = 3,4,5,7 - записать 1.

- Если операнд использует в качестве источника константу или БД SR, то такой операнд исключается из обработки и, следовательно, не занимает места в результате;

- Если для операнда указана выборка и она не пройдена – такой сигнал не обрабатывается и результат для него не меняется;

- Обработка операндов производится поэлементно без сверток: сначала обрабатывается операнд 1, затем операнд 2;

- Если размерность результата больше суммарной размерности операндов (т.е. в результате есть свободное место) или определено условие результата для записи, то в эти элементы записывается:

- 0, если в результате нет ни одного значения, равного

или

если Пар2 #0 и #1 - Пар2

иначе - 3

или

если Пар3 #0 и #1 - Пар3

- 1, если такие значения есть.

Не обрабатываются следующие сигналы:

- недостоверные данные операнда;
- данные операнда, не прошедшие выборку (если задана);
- данные операнда с источником данных константа или символьный массив;
- данные, для результата которых не пройдена проверка по условиям результата.

Анализ основывается на фиксации переходов.

Для дискретных операндов:

- в операнде 1:
 - прямой переход: изменение значения 0->1;
 - обратный переход: 1->0;
- в операнде 2:
 - прямой переход: изменение значения 1->0;
 - обратный переход: 0->1;

Диагностика перехода производится на основании текущего и предыдущего значений.

Для аналоговых операндов:

- прямой переход: изменение статуса из 0,1,2,6 в 3,4,5,7
- обратный переход: 3,4,5,7 в 0,1,2,6

Диагностика перехода основывается на текущем статусе и значении результата с предыдущего такта:

Текущий статус		Текущее значение	
-----		-----	
3,4,5,7	И	= 0	- прямой переход
0,1,2,6	И	= 1	- обратный переход
		или	
		если Пар2 #0 и #1, то Пар2	

Анализ

- ```

- Если есть прямой переход, то:
 - если Пар2 = 0 или 1, то:
 - Рез=3
 - иначе:
 - Рез=Пар2;
- Если есть обратный переход, то:
 - если Рез'=Пар2 или 3, то:
 - если Пар3 # 0 и # 1, то:
 - Рез=Пар3;
 - иначе:
 - для дискретных:
 - Рез=ЗнДис
 - для аналоговых:
 - Рез=0
 - иначе:
 - для дискретных:
 - Рез=ЗнДис
 - для аналоговых:
 - Рез=0
- Если нет перехода:
 - Рез НЕ меняется.

```

**A.4.13 Анализ статуса (Win&Lin)**

Данный алгоблок предназначен для анализа статуса. В алгоблоке производится побитная обработка статуса. Алгоблок может использоваться только в исполняющих системах, работающих под Windows или Linux.

**Параметр 1** - количество используемых операндов:

= 1(11) - используется только 1-й операнд;

= 2(12) - используется только 2-й операнд;

= 0(10) - используются оба операнда.

Значения 10,11,12 - аналогичны 0,1,2 по использованию операндов, но отличны по режиму проверки битов:

- 0,1,2 - проверка битов по «И» - все проверяемые биты должны иметь указанное значение;

- 10,11,12 - проверка битов по «ИЛИ» - достаточно, чтобы хотя бы один бит имел указанное значение.

**Параметр 2** - используемые для анализа биты статуса:

= 0 - бит не анализируется;

# 0 - бит анализируется;

**Параметр 3** - проверяемое значение:

= 0 - бит д.б. = 0

# 0 - бит д.б. = 1

Порядок задания битов в параметрах:

<Квит><УС6><УС5><УС4><УС3><УС2><УС1><ДСТ>

Результат проверки:

- режим "И" (пар-р1 = 0,1,2):

- Если все анализируемые биты имеют указанное значение, то результат равен **Истина**, иначе **Ложь**;

- режим "ИЛИ" (пар-р1 = 10,11,12):

- Если хотя бы один анализируемый бит имеет указанное значение, то результат равен **Истина**, иначе **Ложь**.

### Пример

Пар2 = 1011 - Проверяются биты <УС3>..**<УС1><ДСТ>**

Пар3 = 0010(или просто 10) - Проверяемые значения:

<ДСТ> - д.б. = 0

<УС1> - = 1

<УС3> - = 0

Состояние остальных битов неважно.

Свертка операндов определяется флагом свертки «И»/«ИЛИ» результата.

## А.5 Динамика

1. Интегратор по методу прямоугольника
2. Дифференциатор
3. Звено запаздывания
4. Фильтр скользящего среднего
5. Ограничение
6. ШИМ
7. Аperiodическое звено
8. Статистика (среднее, минимум, максимум)
9. Фильтр
10. Идентификация модели

Для операций **Ограничение, ШИМ** возможно использование **проверки исполнения по условиям результата**. Для операции **Идентификация модели** возможно использование **выборки операндов**.

### А.5.1 Интегратор по методу прямоугольника

**Операнд 1** - Интегрируемая величина.

**Операнд 2** - Максимальное значение интеграла.

Запуск интегратора производится по условию операнда 1.

#### Параметры операции:

- 1 - #0 - режим домножения на DT - время между запусками алгоблока;
- 2 - #0 - коэффициент интегрирования;
- 3 - диапазон результата:
  - Если задан в виде Const:
    - Если  $Const > 0$  :  $[0, Const]$
    - Если  $Const < 0$  :  $[Const, 0]$
    - Если  $Const = 0$  : Диапазон не задан
  - Если задан ссылкой, то по ссылке берутся 2 значения:
    - 1-е - Min, 2-е - Max; проверка производится, если  $Min < Max$ .

#### Алгоритм работы

Если интегратор только что запущен, то сразу переход на Шаг4, если был запущен ранее, то на Шаг1.

Шаг1:  $Рез = Рез + Оп1 * P(1) * DT$ ; домножение на  $P(1)$  и  $DT$  только если указано;

Шаг2: Проверка результата на диапазон, если задана;

Шаг3: Проверка результата на максимальное значение, заданное в операнде 2:

- если значение больше (или равно)  $Оп2$  - записать в логический результат - **Истина**;
- если значение меньше  $Оп2$  - записать в логический результат - **Ложь**;

Шаг4: Запомнить текущее время.

#### Алгоритм приведения размерностей

|  |           |           |        |         |
|--|-----------|-----------|--------|---------|
|  | Результат | Скаляр C1 | Вектор | Матрица |
|--|-----------|-----------|--------|---------|



| Операнд 1         |  |   | $B1(n1)$ | $M1(n1,m1)$ |
|-------------------|--|---|----------|-------------|
| Скаляр СЗ         |  | 1 | 2        | 3           |
| Вектор ВЗ(пЗ)     |  | 4 | 5        | 6           |
| Матрица МЗ(пЗ,мЗ) |  | 7 | 8        | 9           |

Сл.1 - обычная операция;

Сл.2,3 - используется 1-й эл-т вектора или матрицы - на остальные проводится тиражирование;

Сл.4,7 - вектор/матрица операнда 1 сводятся к скаляру суммированием, далее, как сл.1;

Сл.5,9 - поэлементные операции по минимальным размерностям операнда 1 и результата;

Сл.6 - операции над 1-м вектором матрицы с учетом размерностей - на остальные вектора матрицы результата проводится тиражирование;

Сл.8 - матрица суммированием сводится к вектору по 2-й размерности суммированием, далее, как в сл.5.

### Использование Операнда 2

- в случаях 1,2,3,4,7 (фактический результат - скаляр) используется 1-й элемент операнда 2;
- в случаях 5,6,8 (фактический результат - вектор) используются значения 1-го вектора операнда 2;
- в случае 9 используется весь операнд 2.

### Формирование логического результата

- в случаях 1,2,3,4,7 (фактический результат - скаляр):

Если логический Результат:

- скаляр - запись в скаляр;
- вектор, матрица - тиражирование скаляра на вектор или матрицу соответственно;

- в случаях 5,6,8 (фактический результат - вектор):

Если логический Результат:

- скаляр - проводится свертка вектора в скаляр по флагу «И»/«ИЛИ»;
- вектор - поэлементная запись в вектор;
- матрица - тиражирование результирующего вектора на все вектора матрицы;

- в случае 9 (фактический результат - матрица):

Если логический Результат:

- скаляр - свертка матрицы в скаляр по флагу «И»/«ИЛИ»;
- вектор - свертка матрицы к вектору по флагу «И»/«ИЛИ» (по 2-й размерности);
- матрица - поэлементная запись.

### Использование флага свертки операндов в алгоблоке

В случае результата - вектор возможно изменение направления свертки Оп1:

Стандартно:

- количество результатов равно минимуму 1-х размерностей Оп1 и Рез;
- свертка матрицы Оп1 к вектору проводится по 2-й размерности;

Инверсно:

- количество результатов равно минимуму 2-й размерности Оп1 и 1-й размерности Рез;
- свертка матрицы Оп1 к вектору проводится по 1-й размерности;

## А.5.2 Дифференциатор

**Операнд 1** - дифференцируемая величина (скаляр, вектор, для матрицы используется 1-й вектор);

**Операнд 2** - буфер (вектор, матрица, скаляр рассматривается как вектор единичной длины).

Буфер содержит:

- предыдущие значения величины (со статусом);
- засечки времени, если требуется (разницу времени между вызовами алгоблока).

Размерность результата совпадает с размерностью операнда 1.

Алгоблок включается в работу по условию операнда 1:

- **Истина** - работа;
- **Ложь** - отключен.

**Параметр 1** - режим работы:

= 0 - дифференциатор без учета времени (рассчитывается только по значениям дифференцируемой величины):

Рез=Тек\_Зн-Сохран\_Зн;

= 1 - дифференциатор с учетом времени (глубина буфера дифференцирования задается параметром 2 в СЕК.):

Рез=(Тек\_Зн-Сохран\_Зн)/Время;

= 2 - дифференциатор с учетом времени (глубина буфера дифференцирования задается параметром 2 - количество точек дифференцирования):

Рез=(Тек\_Зн-Сохран\_Зн)/Время;

= 3 - дифференциатор для обработки счетчиков с учетом времени (предполагается, что дифференцируемой величиной являются показания счетчиков; глубина буфера дифференцирования фиксирована одной точкой; время в операнде 2 НЕ сохраняется):

Рез=(Тек\_Зн-Сохран\_Зн)/Время;

Если (Тек\_Зн-Сохран\_Зн)<0, то

Рез=(Тек\_Зн-Сохран\_Зн+Макс)/Время;

,где

Тек\_Зн - текущее значение дифференцируемой величины, взятое на данном такте из Операнда 1;

Сохран\_Зн - выбранное из буфера (операнд 2) сохраненное значение (глубина выборки определяется параметром 2 алгоблока);

Макс - максимальное значение счетчика (определяется параметром 2);

Время - разница времени между данными значениями.

**Параметр 2:**

- для случаев Параметр 1 =0,1,2:

- глубина буфера дифференцирования. Режим задания зависит от Параметра 1:

| Параметр 1 | Задание                        |
|------------|--------------------------------|
| 0, 2       | Количество точек               |
| 1          | Время дифференцирования в сек. |

Размер буфера N также зависит от Параметра 1:

| Параметр 1 | Размер буфера N                                                |
|------------|----------------------------------------------------------------|
| 0          | K                                                              |
| 1, 2       | K/2 (требуется хранить также засечки времени для каждой точки) |

,где

K - Общее количество элементов хранения в операнде 2 для одной дифференцируемой величины.

Ограничение на размер - соответствующая размерность операнда 2.

- для случая Параметр 1=3:
- период расчета выходного значения в секундах; в остальное время выходное значение не меняется.

Параметр 3:

- для случаев Параметр 1=0,1,2:
  - T - время сглаживания (если #0). Включает функцию экспоненциального сглаживания рассчитанной производной.

Задание в зависимости от параметра 1:

| Параметр 1 | Параметр 3 |
|------------|------------|
| 0, 2       | В тактах   |
| 1          | В секундах |

С учетом сглаживания результирующая величина рассчитывается следующим образом:

$\text{Рез} = \text{Рез} * \text{Dt} / \text{T} + \text{Рез}' * (1 - \text{Dt} / \text{T})$  для задания в секундах;

$\text{Рез} = \text{Рез} / \text{T} + \text{Рез}' * (1 - 1 / \text{T})$  для задания в тактах

, где

Рез - текущий результат на данном такте;  
Рез'- результат с предыдущего такта;  
Dt - время между тактами;  
T - время сглаживания в тактах или секундах;

Если Dt/T или 1/T больше или равно 1, то предыдущее значение не используется.

- для случая Параметр 1=3:
- максимальное значение счетчика.

#### Алгоритм работы при обработке счетчиков (П1=3)

Если период вызова, определенный параметром 2 больше 1 сек и:

- укладывается целое число раз в 60 сек (1 минуте), то:

- расчет "привязывается" к астрономической кратности периода в минуте (например, для периода 12 сек расчет производится на 0,12,24,36 и 48 секундах каждой минуты);
- укладывается целое число раз в 3600 сек (1 часе), то:
- расчет "привязывается" к астрономической кратности периода в часе;
- укладывается целое число раз в 86400 сек (1 сутках), то:
- расчет "привязывается" к астрономической кратности периода в сутках.

Во всех остальных случаях (период меньше 1 сек или отсутствует кратность) "привязка" к астрономическому времени не производится.

### **Использование флага свертки операндов в алгоблоке**

Для случаев BBB, MBV, BMB, MMB, BBM, MBM, BMM, MMM возможно изменение расположения элементов в буфере Оп2:

Стандартно:

- по 1-му индексу - номер звена запаздывания, дифференциатора или фильтра.
- по 2-му индексу – буфер;

Инверсно:

- по 1-му индексу – буфер;
- по 2-му индексу - номер звена запаздывания, дифференциатора или фильтра.

### **А.5.3 Звено запаздывания**

**Операнд 1** - обрабатываемая величина (скаляр, вектор, для матрицы используется 1-й вектор);

**Операнд 2** - буфер (вектор, матрица, скаляр рассматривается как вектор единичной длины).

Буфер содержит:

- предыдущие значения величины (со статусом);
- засечки времени, если требуется (разницу времени между вызовами алгоблока);

Размерность результата совпадает с размерностью операнда 1.

Алгоблок включается в работу по условию операнда 1:

- **Истина** - работа;
- **Ложь** - отключен.

**Параметр 1** - режим работы:

- = 0 - звено запаздывания с глубиной буфера, заданной в точках (параметр 2);
- = 1 - звено запаздывания с глубиной буфера, заданной временем в сек. (параметр 2);

**Параметр 2** - глубина буфера. Режим задания зависит от Параметра 1:

| Параметр 1 | Задание                   |
|------------|---------------------------|
| 0          | Количество точек          |
| 1          | Время запаздывания в сек. |

Размер буфера N также зависит от Параметра 1:

| Параметр 1 | Размер буфера N                                                |
|------------|----------------------------------------------------------------|
| 0          | K                                                              |
| 1          | K/2 (требуется хранить также засечки времени для каждой точки) |

, где

K - Общее количество элементов хранения в операнде 2 для одной величины.

Ограничение на размер - соответствующая размерность операнда 2.

#### Использование флага свертки операндов в алгоблоке

Для случаев BBV, MBV, BMB, MMB, VBM, MBM, BMM, MMM возможно изменение расположения элементов в буфере Оп2:

Стандартно:

- по 1-му индексу - номер звена запаздывания, дифференциатора или фильтра.
- по 2-му индексу – буфер;

Инверсно:

- по 1-му индексу – буфер;
- по 2-му индексу - номер звена запаздывания, дифференциатора или фильтра.

#### А.5.4 Фильтр скользящего среднего

**Операнд 1** - фильтруемая величина (скаляр, вектор, для матрицы используется 1-й вектор);

**Операнд 2** - буфер (вектор, матрица, скаляр рассматривается как вектор единичной длины).

Буфер содержит:

- предыдущие значения величины (со статусом);
- засечки времени, если требуется (разницу времени между вызовами алгоблока);

Размерность результата совпадает с размерностью операнда 1.

Алгоблок включается в работу по условию операнда 1:

- **Истина** - работа;
- **Ложь** - отключен.

**Параметр 1** - режим работы:

= 0 - фильтр без учета времени (среднее значение рассчитывается по значениям величины и количеству точек):

$\text{Рез} = \text{SnAi} / N$ , где:

$\sum A_i$  - сумма значений  $A_i$  ( $i=1,...,N$ )

$N$  - количество точек;

= 1 - фильтр с учетом времени (глубина выборки фильтра задается параметром 2 в сек.):

$\text{Рез} = \sum A_i \cdot T_i / \sum T_i$ , где:

$\sum A_i \cdot T_i$  - сумма произведений: значение  $A_i$  на время  $T_i$  (время от предыдущего замера до текущего); ( $i=1,...,N$ );  $N$  - глубина выборки определяется временем, заданным в параметре 2;

= 2 - фильтр с учетом времени (глубина буфера фильтра задается параметром 2 - количество точек):

$\text{Рез} = \sum A_i \cdot T_i / \sum T_i$ , где:

$\sum A_i \cdot T_i$  - сумма произведений: значение  $A_i$  на время  $T_i$  (время от предыдущего замера до текущего); ( $i=1,...,N$ );  $N$  - глубина выборки определяется количеством точек, заданным в параметре 2.

**Параметр 2** - глубина буфера. Режим задания зависит от Параметра 1:

| Параметр 1 | Задание          |
|------------|------------------|
| 0          | Количество точек |
| 1, 2       | Время в сек.     |

Размер буфера  $N$  также зависит от Параметра 1:

| Параметр 1 | Размер буфера $N$                                                |
|------------|------------------------------------------------------------------|
| 0          | $K$                                                              |
| 1, 2       | $K/2$ (требуется хранить также засечки времени для каждой точки) |

, где

$K$  - Общее количество элементов хранения в операнде 2 для одной величины.

Ограничение на размер - соответствующая размерность операнда 2.

#### Замечания

- При отключенном дифференциаторе (звене запаздывания/фильтре) все равно производится накопление данных в буфере - не формируется только результат.

- На 1-м шаге (ситуация при запуске программы, когда буфер еще пуст) на выход подается 0 (для фильтра - текущее значение), на каждом следующем шаге алгоритм работает по текущей глубине буфера пока буфер не заполнится полностью - устанавливается нормальный режим работы.

- Если глубина буфера равна 1 и требуется учитывать время, то разница времен получается следующим образом:

текущее время - запомненное время предыдущего запуска алгоритма.

#### Учет размерностей

Результат – скаляр

|  |           |        |        |         |
|--|-----------|--------|--------|---------|
|  | Операнд 1 | Скаляр | Вектор | Матрица |
|--|-----------|--------|--------|---------|

|           |  |   |   |   |
|-----------|--|---|---|---|
| Операнд 2 |  |   |   |   |
| Скаляр    |  | 1 | 2 | 3 |
| Вектор    |  | 4 | 5 | 6 |
| Матрица   |  | 7 | 8 | 9 |

Сл.1,2,3 - Глубина буфера ограничена 1 числом; Используется только 1-й элемент операнда 1;

Сл.4,5,6 - Используется только 1-й элемент операнда 1: размер буфера определяется размером вектора;

Сл.7,8,9 - Используется только 1-й элемент операнда 1: размер буфера определяется размером матрицы.

Результат – вектор

|           |           |        |        |         |
|-----------|-----------|--------|--------|---------|
|           | Операнд 1 | Скаляр | Вектор | Матрица |
| Операнд 2 |           |        |        |         |
| Скаляр    |           | 11     | 12     | 13      |
| Вектор    |           | 14     | 15     | 16      |
| Матрица   |           | 17     | 18     | 19      |

Сл.11,12,13 - Глубина буфера ограничена 1 числом; Всегда используется 1-й элемент операнда 1; Всегда тиражирование на вектор;

Сл.14 - Тиражирование результата на вектор;

Сл.15 - Используется каждый элемент вектора Оп1 (глубина буфера -1);

Сл.16 - Аналог сл.15 (производится свертка матрицы операнда 1 к вектору сложением);

Сл.17 - Аналог сл.14 (глубина буфера определяется матрицей);

Сл.18 - Используется каждый элемент вектора операнда 1 – буфер создается по соответствующей 2-й размерности матрицы операнда 2;

Сл.19 - Аналог сл.18 (свертка матрицы операнда 1 к вектору сложением).

Результат – матрица

|           |           |        |        |         |
|-----------|-----------|--------|--------|---------|
|           | Операнд 1 | Скаляр | Вектор | Матрица |
| Операнд 2 |           |        |        |         |
| Скаляр    |           | 21     | 22     | 23      |
| Вектор    |           | 24     | 25     | 26      |
| Матрица   |           | 27     | 28     | 29      |

Сл.21,22,23 - Глубина буфера ограничена 1 числом; Всегда используется 1-й элемент Оп1; Всегда тиражирование скаляра на матрицу;

Сл.24 - Тиражирование скаляра на матрицу;

Сл.25 - Используется каждый элемент вектора Оп1 (глубина буфера -1); Тиражирование на матрицу;

Сл.26 - Аналог сл.25 (свертка матрицы операнда 1 к вектору сложением);

Сл.27 - Аналог сл.24 (глубина буфера определяется матрицей);

Сл.28 - Используется каждый элемент вектора операнда 1 – буфер создается по соответствующей 2-й размерности матрицы операнда 2; Тиражирование на матрицу;

Сл.29 - Аналог сл.28 (свертка матрицы операнда 1 к вектору сложением).

**Использование флага свертки операндов в алгоблоке**

Для случаев ВВВ, МВВ, ВМВ, ММВ, ВВМ, МВМ, ВММ, МММ возможно изменение расположения элементов в буфере Оп2:

Стандартно:

- по 1-му индексу - номер звена запаздывания, дифференциатора или фильтра.
- по 2-му индексу – буфер;

Инверсно:

- по 1-му индексу – буфер;
- по 2-му индексу - номер звена запаздывания, дифференциатора или фильтра.

#### A.5.5 Ограничение

**Операнд 1** - аналоговый источник входных данных.

**Операнд 2** - аналоговый источник задания.

**Результат** - разность: Оп1-Оп2.

Существуют 3 вида запретов:

1. Запрет на изменение выходного сигнала в «+» или «-» по 2 входным дискретным сигналам (при одновременной подаче входных дискретных сигналов на выходе устанавливается 0);
2. Запрет на изменение выходного сигнала выше или ниже (по абсолютной величине) того значения, которое выход имел в момент запрета. Запрет формируется по 2 входным дискретным сигналам (один - запрет на увеличение, 2- запрет на уменьшение). При одновременной подаче дискретных сигналов выход «замораживается».
3. Запрет на выход за пределы диапазона, определяемого параметрами 1 и 2: если значение меньше параметра 1, то на выход пишется параметр 1; если значение больше параметра 2, то на выход пишется параметр 2. Дискретные сигналы запрета в данном случае не используются.

Условие операнда 1 - 1-й дискретный сигнал запрета.

Условие операнда 2 - 2-й дискретный сигнал запрета.

**Параметр 3** - режим работы:

= 0 - используется запрет 3;

= 1 - используются запреты 1 и 2.

**Параметр 1:**

- если Пар3 = 0 - нижняя граница диапазона;
- если Пар3 = 1 - режим работы 1-го дискретного сигнала запрета:
  - = 0 - не учитывается;
  - = 1 - запрет на >0;
  - = 2 - запрет на увеличение.

**Параметр 2:**

- если Пар3 = 0 - верхняя граница диапазона;
- если Пар3 = 1 - режим работы 2-го дискретного сигнала запрета:
  - = 0 - не учитывается;
  - = 1 - запрет на <0;
  - = 2 - запрет на уменьшение.



Перед проведением расчета сами операнды и их условия (запреты) приводятся к типу результата:

- операнды:

- сложением для уменьшения размерности;
- тиражированием для увеличения размерности;

- условия операндов:

- сворачиванием по флагу «И»/«ИЛИ» в соответствии с установленным флагом для уменьшения размерности;
- тиражированием для увеличения размерности.

Таким образом, при выполнении расчета все составляющие (операнды, их условия и результат) имеют одинаковую размерность.

Для выполнения расчета оба операнда д.б. достоверны, иначе никаких действий не производится.

Для включения в работу условий операндов необходимо включить общий флаг проверки операндов, а также флаг проверки по векторам.

### Алгоритм функционирования с учетом запрета 3

$X_r$  - расчетная разность  $Op1 - Op2$ .

$X_u$  - устанавливаемое значение выхода в результате работы данного алгоблока.

$X_r = Op1 - Op2$

$X_u = X_r$ ;

- если  $X_r < Пар1$ , то

$X_u = Пар1$ ;

- иначе:

- если  $X_r > Пар2$ , то

$X_u = Пар2$ .

**Алгоритм функционирования с учетом запретов 1 и 2** (даны для скаляров, для остальных размерностей аналогично):

$X_r$  - расчетная разность  $Op1 - Op2$ ;

$X_r$  - предыдущее значение выхода (на момент запуска алгоблока);

$X_u$  - устанавливаемое значение выхода в результате работы данного алгоблока.

|   | Параметр 1 | 0 | 1 | 2 |
|---|------------|---|---|---|
|   | Параметр 2 |   |   |   |
| 0 |            | 1 | 2 | 3 |
| 1 |            | 4 | 5 | 6 |
| 2 |            | 7 | 8 | 9 |

Сл.1 Запреты не используются:  $X_u = X_r$ ;

Сл.2 Есть запрет на  $>0$ :

- Если запрет включен (условие операнда 1 равно Истине),  
  - то
  - если  $X_r > 0$ , то  $X_u = 0$
  - иначе:  $X_u = X_r$
  - иначе:  $X_u = X_r$

Сл.3 Есть запрет на увеличение:

- Если запрет включен (условие операнда 1 равно Истине),  
  - то
  - если  $X_r > X_r$ , то  $X_u = X_r$  (на выход ничего не пишем)
  - иначе:  $X_u = X_r$ 
    - иначе:  $X_u = X_r$

Сл.4 Есть запрет на  $<0$ :

- Если запрет включен (условие операнда 2 равно Истине),  
  - то
  - если  $X_r < 0$ , то  $X_u = 0$
  - иначе:  $X_u = X_r$ 
    - иначе:  $X_u = X_r$

Сл.7 Есть запрет на уменьшение:

- Если запрет включен (условие операнда 2 равно Истине),  
  - то
  - если  $X_r < X_r$ , то  $X_u = X_r$  (на выход ничего не пишем)
  - иначе:  $X_u = X_r$ 
    - иначе:  $X_u = X_r$

Сл.5 Есть запреты на  $>0$  и  $<0$ :

- Если запрет на  $>0$  включен (Ус.Оп1=И) и запрет на  $<0$  включен (Ус.Оп2=И), то  $X_u = 0$
- Если оба запрета отключены:  $X_u = X_r$
- Если включен какой-то один запрет - см.Сл.2 и 4

Сл.6 Есть запрет на увеличение и на  $<0$ :

- Если оба запрета включены:
  - если  $X_r > X_r$ , то  $X_u = X_u$  (не пишем)
- иначе,
- если  $X_r < 0$ , то  $X_u = 0$
- иначе  $X_u = X_r$ 
  - Если оба запрета отключены:  $X_u = X_r$
  - Если включен какой-то один запрет - см.Сл.3 и 4

Сл.8 Есть запрет на  $>0$  и на уменьшение:

- Если оба запрета включены:
  - если  $X_r > 0$ , то  $X_u = 0$
  - иначе,
- если  $X_r < X_r$ , то  $X_u = X_u$  (не пишем)
- иначе  $X_u = X_r$ 
  - Если оба запрета отключены:  $X_u = X_r$
  - Если включен какой-то один запрет - см.Сл.2 и 7

Сл.9 Есть запреты на увеличение и уменьшение

- Если оба запрета включены:
  - $X_u = X_u$  (не пишем)
- Если оба запрета отключены:  $X_u = X_r$

- Если включен какой-то один запрет - см.Сл.3 и 7

### Использование флага свертки операндов в алгоблоке

Стандартно:

- количество результатов равно минимуму 1-й размерности Оп1 и 2-й размерности Рез;
- свертка Оп1/Оп2 (при Рез-Вектор и Оп1/Оп2-Матрица) проводится по 2-й размерности операнда;
- при одинаковых размерностях Оп и Рез (или размерность Рез больше размерности операнда) чтение Оп производится по индексам:
  - 1-й : 1..1-я размерность Оп;
  - 2-й : 1..2-я размерность Рез;

Инверсно:

- количество результатов равно минимуму 2-й размерности Оп1 и 2-й размерности Рез;
- свертка Оп1/Оп2 (при Рез-Вектор и Оп1/Оп2-Матрица) проводится по 1-й размерности операнда;
- при одинаковых размерностях Оп и Рез (или размерность Рез больше размерности операнда) чтение Оп производится по индексам:
  - 1-й : 1..2-я размерность Оп;
  - 2-й : 1..2-я размерность Рез.

### А.5.6 ШИМ

**Операнд 1** - параметр модуляции: время или количество шагов.

**Операнд 2** - модулируемая величина.

**Параметры операции:**

1-й - режим задания параметра модуляции (операнд 1):

= 0 - шаги;

= 1 - время;

2-й - значение для выхода (когда не выдается 1-й операнд).

3-й - структура области данных, определенной параметром 2:

= 0 - скаляр;

= 1 - вектор;

= 2 - матрица.

Структуры типа вектор или матрица могут использоваться только, если параметр 2 задан ссылкой, иначе он считается скаляром.

### Алгоритм работы алгоблока

Шаг1. - Если значение в Операнде 1 больше нуля, то на выход подается значение из Операнда 2;

- Если значение в Операнде 1 меньше или равно нулю, то на выход подается значение из области данных, определенной 2-м параметром.

Шаг2. - Если значение в Операнде 1 больше нуля, то из него вычитается:

- DT (время между запусками алгоблока), если параметр 1 равен 1;
- 1, если параметр 1 равен 0.

### Структурирование размерностей

Во всех далее приведенных случаях, использование области, заданной параметром 2 равнозначно использованию операнда 2.

|           | Результат | Скаляр | Вектор | Матрица |
|-----------|-----------|--------|--------|---------|
| Операнд 1 |           |        |        |         |
| Скаляр    |           | 1      | 2      | 3       |
| Вектор    |           | 4      | 5      | 6       |
| Матрица   |           | 7      | 8      | 9       |

Сл.1 Тривиальный случай;

Сл.2 ШИМ-вектор с единым параметром;

Сл.3 ШИМ-матрица с единым параметром;

Сл.4,7 Аналог сл.1; используются 1-е элементы вектора или матрицы операнда 1;

Сл.5 Количество ШИМ - минимум 1-й размерности результата и операнда 1;

Сл.6 ШИМ-матрица; каждый вектор результата имеет свой параметр - соответствующий элемент вектора операнда 1;

Сл.8 Аналог сл.5; используется 1-й вектор операнда 1;

Сл.9 Количество ШИМ - минимум всех размерностей операнда 1 и результата.

Количество ШИМ:

- случаи: 1,4,7 - ШИМ-скаляр;
- случаи: 2,5,8 - ШИМ-вектор;
- случай: 3,6,9 - ШИМ-матрица.

#### Использование Операнда 2 и области, заданной параметром 2

|             | Операнд 2 | Скаляр | Вектор | Матрица |
|-------------|-----------|--------|--------|---------|
| ШИМ         |           |        |        |         |
| ШИМ-скаляр  |           | 1      | 2      | 3       |
| ШИМ-вектор  |           | 4      | 5      | 6       |
| ШИМ-матрица |           | 7      | 8      | 9       |

Сл.1 - тривиальный случай: используется скаляр;

Сл.2,3 - свертка операнда 2(области) к скаляру (из вектора или матрицы соответственно);

Сл.4,7 - использование скаляра для всех элементов ШИМ-вектора или матрицы;

Сл.5 - тривиальный случай: используются элементы вектора;

Сл.6 - свертка матрицы к вектору;

Сл.8 - использование вектора для всех векторов ШИМ-матрицы;

Сл.9 - тривиальный случай: используются все элементы матрицы.

#### Использование флага свертки операндов в алгоблоке

Свертка Оп2 (или области, заданной через Параметр 2) стандартно проводится по 2-й размерности (свертка типа матрица -> вектор).

Инверсная свертка проводится по 1-й размерности.

Случаи проведения сверток Матрица -> Вектор: ШИМ-вектор и Оп2 (или область) имеет тип матрицы.

### А.5.7 Апериодическое / Колебательное звено

**Операнд 1** - обрабатываемая величина;

**Операнд 2** - буфер хранения предыдущих значений.

**Параметр 3** - режим работы:

= 0 - апериодическое звено

# 0 - колебательное звено

**Для апериодических звеньев:**

**Параметр 1** - T1 - постоянная времени первого апериодического звена;

**Параметр 2** - T2 - постоянная времени второго апериодического звена.

Признаком задания **неустойчивого апериодического звена** является флаг отрицания соответствующего параметра.

**Для колебательного звена:**

**Параметр 1** - T - постоянная времени;

**Параметр 2** - E - коэффициент затухания.

Если Параметр 1 и Параметр 2 равны 0 - алгоблок не работает.

Для колебательного звена и двойного апериодического звена должны быть заданы оба параметра и буфер хранения в операнде 2 (операнд 2 не может быть константой).

Алгоблок включается и отключается по соответствующему значению условия операнда 1:

- Истина - работа;
- Ложь - останов.

#### Формулы расчета

##### 1. Апериодические звенья

###### Апериодическое звено

$$\text{Рез} = \text{Оп1} * K + \text{Рез}' * (1 - K)$$

$$K = dT * (3 * T + dT) / 3 * T * (T + dT)$$

###### Неустойчивое апериодическое звено

$$\text{Рез} = \text{Оп1} * M + \text{Рез}' * (1 + M)$$

$$M = dT / (T - dT)$$

, где:

- Рез - формируемый результат;
- Оп1 - текущее значение операнда 1;
- dT - интервал времени между вызовами алгоблока;
- Рез' - предыдущее значение результата;
- T - постоянная времени, заданная одним из параметров.

- Если  $K \geq 1$  для устойчивого звена - то второе слагаемое отбрасывается;
- Если  $dT \geq T$  для неустойчивого звена, то звено не работает - выход не меняется:  
Рез=Рез' - результат не меняется
- Если апериодика отключена (соответствующее условие операнда 1 равно "Ложь"):  
Рез=Оп1
- Если апериодика включена на самом первом исполнении алгоблока (соответствующее условие операнда 1 равно "Истина" и  $dT=0.0$ ):  
Рез=Рез' - результат не меняется.

### Двойные апериодические звенья

| Устойчивое                         | Неустойчивое                       |
|------------------------------------|------------------------------------|
| $Оп2 = Оп1 * K1 + Оп2' * (1 - K1)$ | $Оп2 = Оп1 * M1 + Оп2' * (1 + M1)$ |
| $Рез = Оп2 * K2 + Рез' * (1 - K2)$ | $Рез = Оп2 * M2 + Рез' * (1 + M2)$ |

, где:

$$K1 = dT * (3 * T1 + dT) / 3 * T1 * (T1 + dT)$$

$$K2 = dT * (3 * T2 + dT) / 3 * T2 * (T2 + dT)$$

$$M1 = dT / (T1 - dT)$$

$$M2 = dT / (T2 - dT)$$

Оп2' - сохраненное в операнде 2 значение операнда 1 с предыдущего такта;

Оп2 - новое сохраняемое значение в операнде 2;

Рез - формируемый результат;

Оп1 - текущее значение операнда 1;

dT - интервал времени между вызовами алгоблока;

Рез' - предыдущее значение результата;

T1 и T2 - постоянные времени, заданные параметрами.

Звенья могут комбинироваться любым образом:

2 устойчивых;

2 неустойчивых

устойчивое и неустойчивое;

неустойчивое и устойчивое.

- Если  $K1, K2 \geq 1$  - то второе слагаемое отбрасывается.
- Если  $dT \geq T1, dT \geq T2$  для неустойчивого звена, то звено не работает - выход не меняется;
- Если апериодика отключена (соответствующее условие операнда 1 равно "Ложь"):  
Оп2=Оп1  
Рез=Оп1
- Если апериодика включена на самом первом исполнении алгоблока (соответствующее условие операнда 1 равно "Истина" и  $dT=0.0$ ):  
Оп2=Оп2' - операнд 2 не меняется;  
Рез=Рез' - результат не меняется.

## 2. Колебательное звено

S=Рез' - сохранение результата;

Рез=(dT\*dT\*Оп1 + (2\*T\*T + 2\*E\*T\*dT)\*S - T\*T\*Оп2) / (dT\*dT + 2\*E\*T\*dT + T\*T)

Оп2=S - новое предпредыдущее значение результата;

, где

S,Рез' - предыдущее значение результата;

Рез - формируемый результат;

dT - интервал времени между вызовами алгоблока;

Оп1 - текущее значение операнда 1;

T - постоянная времени;

E - коэффициент затухания;

Оп2 - сохраненное в операнде 2 предпредыдущее значение результата;

#### Для коэффициента затухания E:

0 =< E < 1 - колебательное звено;

-1 =< E < 0 - неустойчивое колебательное звено;

1 =< E - двойное аperiodическое звено;

-1 > E - двойное неустойчивое аperiodическое звено.

Реально, из-за квантования времени момент перехода от колебательного звена к неустойчивому колебательному звену равен не 0, а, примерно, может быть определен по формуле:

$E = -dT / (2 * T)$

#### Структурирование размерностей

Результат - Скаляр

|                   | Операнд 1 | Скаляр C1 | Вектор V1(n1) | Матрица M1(n1,m1) |
|-------------------|-----------|-----------|---------------|-------------------|
| Операнд 2         |           |           |               |                   |
| Скаляр C2         |           | 1         | 2             | 3                 |
| Вектор B2(n2)     |           | 4         | 5             | 6                 |
| Матрица M2(n2,m2) |           | 7         | 8             | 9                 |

Сл.1 - Основной скалярный вариант

Сл.4 и 7 - аналог сл.1 (используется только 1-й элемент вектора или матрицы)

Сл.2 и 3 - свертка операнда 1 к скаляру сложением; далее аналог сл.1

Сл.5,6,8 и 9 - комбинации сл.4,7 и 2,3.

Результат - Вектор B(n3)

|                   | Операнд 1 | Скаляр C1 | Вектор V1(n1) | Матрица M1(n1,m1) |
|-------------------|-----------|-----------|---------------|-------------------|
| Операнд 2         |           |           |               |                   |
| Скаляр C2         |           | 11        | 12            | 13                |
| Вектор B2(n2)     |           | 14        | 15            | 16                |
| Матрица M2(n2,m2) |           | 17        | 18            | 19                |

Сл.11 - Аналог сл.1 с тиражированием скалярного результата на вектор  
 Сл.14 и 17 - аналог сл.11 (используется только 1-й элемент вектора или матрицы)  
 Сл.12 и 13 - свертка операнда 1 к скаляру сложением; далее аналог сл.11  
 Сл.15 - Основной векторный вариант: расчет по минимальным размерностям:  $\min(n_1, n_2, n_3)$   
 Сл.16 - Свертка операнда 1 к вектору по 2-й размерности матрицы; далее аналог сл.15  
 Сл.18 - Использование 1-го вектора матрицы операнда 2 для хранения; далее аналог сл.15  
 Сл.19 - комбинация сл.16 и 18

Результат - Матрица  $M3(n_3, m_3)$

|                   | Операнд 1 | Скаляр C1 | Вектор B1(n1) | Матрица M1(n1,m1) |
|-------------------|-----------|-----------|---------------|-------------------|
| Операнд 2         |           |           |               |                   |
| Скаляр C2         |           | 21        | 22            | 23                |
| Вектор B2(n2)     |           | 24        | 25            | 26                |
| Матрица M2(n2,m2) |           | 27        | 28            | 29                |

Сл.21 - Аналог сл.1 с тиражированием скалярного результата на матрицу  
 Сл.24 и 27 - аналог сл.21 (используется только 1-й элемент вектора или матрицы)  
 Сл.22 и 23 - свертка операнда 1 к скаляру сложением; далее аналог сл.21  
 Сл.25 - Аналог сл.15 с тиражированием вектора результата на матрицу  
 Сл.26 - Свертка матрицы операнда 1 к вектору сложением; далее аналог сл.25  
 Сл.28 - Использование 1-го вектора матрицы операнда 2 для хранения данных; далее аналог сл.25  
 Сл.29 - Основной матричный вариант: расчет по минимальным размерностям:  $\min(n_1, n_2, n_3), \min(m_1, m_2, m_3)$ .

#### Использование флага свертки операндов в алгоблоке

Для случаев векторных операций (15,16,18,19,25,26,28) возможна смена направления свертки матрицы операнда 1 к вектору.

Стандартно:

- количество результатов: минимум 1-х размерностей Оп1, Оп2 и Рез;
- включение звена определяется сверткой условия Оп1 по 2-й размерности;
- для случаев матрицы в Оп1 (16,19,26) свертка проводится по 2-й размерности

Инверсно:

- количество результатов: минимум 2-й размерности Оп1 и 1-х размерностей Оп2 и Рез;
- включение звена определяется сверткой условия Оп1 по 1-й размерности;
- для случаев матрицы в Оп1 (16,19,26) свертка проводится по 1-й размерности

#### А.5.8 Статистика (среднее, минимальное, максимальное)

Данный алгоблок позволяет рассчитывать текущее среднее, минимальное или максимальное значение для входного сигнала.

**Операнд 1** - источник данных (данные проверяются только по статусам);

**Операнд 2** - используется только при расчете среднего для хранения счетчика количества измерений; счетчик сбрасывается или:



- при достижении максимального значения счетчика для данного типа данных операнда 2:
  - для типа Integer\*1 - 127; Integer\*2 - 32767; Integer\*4 - 2147483647; Real\*4 - 99999999; Logical\*2, **Character\*1 - не ведется!**
- при возникновении сигнала **Истина** на соответствующем входе условия операнда 1 (см. ниже);
- при истечении указанного интервала времени (см. Параметры 2 и 3);

Условие операнда 1 - управление сбросом (индивидуально для каждого сигнала) - **Истина** – произвести сброс; после проведения действий по сбросу соответствующее условие сбрасывается в **Ложь**.

Условие операнда 2 - управление паузой (индивидуально для каждого сигнала, кроме режима - параметр 2 = 0 - общий сигнал **Пауза** для всех) (**Истина** - пауза); в течении паузы никакие действия с сигналом не производятся (в режиме параметр 2 = 0 - таймер останавливается).

**Параметр 1** - режим работы:

- = 0 - расчет среднего;
- = 1 - расчет минимального (операнд 2 не используется);
- = 2 - расчет максимального (операнд 2 не используется);

**Параметр 2** - размерность периода накопления, заданного в параметре 3:

- = 0 - в секундах, отсчитываемых от текущего момента;
  - = 1 - в минутах от начала часа;
  - = 2 - в часах от начала суток;
  - = 3 - в сутках от начала месяца;
- По истечении интервала сразу начинается следующий интервал.

**Параметр 3** - величина интервала.

*Пример.* Для Параметр 2 = 1 и Параметра 3 = 10:

- образуется 6 интервалов в течение каждого часа: сбросы производятся в 0,10,20,30,40,50 мин. каждого часа.

**Примечание.** Для Параметра 2 = 0 (сек) при возникновении условий сброса по значению условия операнда 1:

- обнуляется накопление для соответствующего сигнала;
- засечка времени сбрасывается (единая для всех сигналов).

### Структурирование размерностей

- Для расчета среднего: размерность результата определяется по минимальным размерностям обеих операндов и результата;
- Для расчетов минимального и максимального: размерность результата определяется по минимальным размерностям операнда 1 и результата.

### А.5.9 Фильтр

**Операнд 1** - источник данных.

**Операнд 2** - буфер хранения результата фильтрации.

Расширение операнда 1 – период запуска алгоблока в секундах (если задан);

Условие операнда 2 – управление сбросом (при установке в значение **Истина** значение из операнда 1 переписывается в результат и операнд 2).

**Параметр 1** - постоянная времени фильтра экспоненциального сглаживания;

**Параметр 2** - допустимый выброс;

**Параметр 3** - учитываемый выброс.

#### Алгоритм функционирования фильтра

- если условие ОП2 = **Истина**, то:
  - РЕЗ=ОП1
  - ОП2=ОП1
- расчет  $\Delta = \text{abs}(\text{ОП1} - \text{ОП2})$
- проверка значения Пар1:
  - если Пар1 = 0 , то
    - если  $\Delta > \text{abs}(\text{Пар2})$  , то
      - если  $\Delta < \text{abs}(\text{Пар3})$  , то
        - ОП2=ОП1
      - если  $\Delta \leq \text{abs}(\text{Пар2})$  , то
        - РЕЗ=ОП1
        - ОП2=ОП1
    - если Пар1  $\neq 0$  , то
      - если  $\Delta > \text{abs}(\text{Пар2})$  , то
        - если  $\Delta < \text{abs}(\text{Пар3})$  , то
          - $\text{ОП2} = \text{ОП1} * \text{DT} / \text{T} + \text{ОП2}' * (1 - \text{DT} / \text{T})$  и
          - $\text{РЕЗ} = \text{ОП2} * \text{DT} / \text{T} + \text{РЕЗ}' * (1 - \text{DT} / \text{T})$
        - если  $\Delta \leq \text{abs}(\text{Пар2})$  , то
          - $\text{ОП2} = \text{ОП1} * \text{DT} / \text{T} + \text{ОП2}' * (1 - \text{DT} / \text{T})$
          - РЕЗ=ОП2

, где ОП1 - значение операнда 1;

ОП2 - значение операнда 2;

РЕЗ - результат;

ОП2' - предыдущее значение ОП2;

РЕЗ' - предыдущее значение РЕЗ

DT - время между вызовами алгоблока

T - постоянная времени в сек (задана Параметром 1)

abs – модуль

Если  $\text{DT} / \text{T} > 1$  , то полагается  $\text{DT} / \text{T} = 1$ .

- сброс условий ОП2.

Если в результате работы алгоритма запись в ОП2 и РЕЗ не производится, то для данного фильтра включается счетчик, фиксирующий количество следующих друг за другом тактов, при которых не производится запись. При достижении определенного максимального количества для данного фильтра

формируется состояние **Сброса** (аналогично взводу УСОР2 для данного фильтра) и производятся соответствующие действия. Для данной версии установлено максимальное значение счетчика – 16.

### Структурирование размерностей

Результат - Скаляр

|                   | Операнд 1 | Скаляр C1 | Вектор V1(n1) | Матрица M1(n1,m1) |
|-------------------|-----------|-----------|---------------|-------------------|
| Операнд 2         |           |           |               |                   |
| Скаляр C2         |           | 1         | 2             | 3                 |
| Вектор B2(n2)     |           | 4         | 5             | 6                 |
| Матрица M2(n2,m2) |           | 7         | 8             | 9                 |

Сл.1 - Основной скалярный вариант

Сл.4 и 7 - аналог сл.1 (используется только 1-й элемент вектора или матрицы)

Сл.2 и 3 - свертка операнда 1 к скаляру сложением; далее аналог сл.1

Сл.5,6,8 и 9 - комбинации сл.4,7 и 2,3

Результат - Вектор V(n3)

|                   | Операнд 1 | Скаляр C1 | Вектор V1(n1) | Матрица M1(n1,m1) |
|-------------------|-----------|-----------|---------------|-------------------|
| Операнд 2         |           |           |               |                   |
| Скаляр C2         |           | 11        | 12            | 13                |
| Вектор B2(n2)     |           | 14        | 15            | 16                |
| Матрица M2(n2,m2) |           | 17        | 18            | 19                |

Сл.11 - Аналог сл.1 с тиражированием скалярного результата на вектор

Сл.14 и 17 - аналог сл.11 (используется только 1-й элемент вектора или матрицы)

Сл.12 и 13 - свертка операнда 1 к скаляру сложением; далее аналог сл.11

Сл.15 - Основной векторный вариант: расчет по минимальным размерностям:  $\min(n1, n2, n3)$

Сл.16 - Свертка операнда 1 к вектору по 2-й размерности матрицы; далее аналог сл.15

Сл.18 - Использование 1-го вектора матрицы операнда 2 для хранения; далее аналог сл.15

Сл.19 - комбинация сл.16 и 18

Результат - Матрица M3(n3,m3)

|                   | Операнд 1 | Скаляр C1 | Вектор V1(n1) | Матрица M1(n1,m1) |
|-------------------|-----------|-----------|---------------|-------------------|
| Операнд 2         |           |           |               |                   |
| Скаляр C2         |           | 21        | 22            | 23                |
| Вектор B2(n2)     |           | 24        | 25            | 26                |
| Матрица M2(n2,m2) |           | 27        | 28            | 29                |

Сл.21 - Аналог сл.1 с тиражированием скалярного результата на матрицу

Сл.24 и 27 - аналог сл.21 (используется только 1-й элемент вектора или матрицы)

Сл.22 и 23 - свертка операнда 1 к скаляру сложением; далее аналог сл.21

Сл.25 - Аналог сл.15 с тиражированием вектора результата на матрицу

Сл.26 - Свертка матрицы операнда 1 к вектору сложением; далее аналог сл.25

Сл.28 - Использование 1-го вектора матрицы операнда 2 для хранения данных; далее аналог сл.25  
Сл.29 - Основной векторный вариант: расчет по минимальным размерностям:  $\min(n1, n2, n3)$ ,  
 $\min(m1, m2, m3)$

### Использование флага свертки операндов в алгоблоке

Для случаев векторных операций (15,16,18,19,25,26,28) возможна смена направления свертки матрицы операнда 1 к вектору.

Стандартно:

- количество результатов: минимум 1-х размерностей Оп1, Оп2 и Рез (для Оп1 без учета расширения, если оно задано);
- для случаев матрицы в Оп1 (16,19,26) свертка проводится по 2-й размерности

Инверсно:

- количество результатов: минимум 2-й размерности Оп1 и 1-х размерностей Оп2 и Рез;
- для случаев матрицы в Оп1 (16,19,26) свертка проводится по 1-й размерности (Оп1 без учета расширения, если оно задано).

### А.5.10 Идентификация модели

Данный алгоблок предназначен для идентификации параметров динамической модели объекта 2-го порядка.

**Параметр 1** - период вызова алгоблока в секундах;

**Параметр 2** - режим учета предыдущих результатов:

= 0 - без учета предыдущих результатов;

= 1 - с учетом.

**Параметр 3** - режим пересчета:

= 0 - расчет по 2-й апериодике:  $1^*/(T1S+1)(T2S+1)$

= 1 - расчет с учетом наличия интегратора:  $K*1/(T2S+1)$

**Операнд 1** - U - управление, подаваемое на объект (приводится к скаляру суммированием);

**Операнд 2** - A - выход объекта (приводится к скаляру суммированием);

**Результат** - 1-й вектор:

- коэффициенты конечно-разностной модели;

- 2-й вектор:

K - коэффициент усиления;

T1 - 1-я постоянная времени;

T2 - 2-я постоянная времени.

Условие операнда 1 (1-й элемент) используется для запуска расчета. В момент установки этого элемента в значение **Истина** производится фиксация начальных значений управления и состояния объекта, которые используются в дальнейшем как смещения текущих показателей управления и состояния объекта.

Условие операнда 2 используется как пауза.

### Алгоритм работы

Алгоблок использует дополнительный буфер памяти, выделяемой при 1-ом исполнении. Если не удастся получить необходимую область, то алгоблок не исполняется.

- 1-й проход: сохранение U,A;
- 2-й проход или УСОР1 = "Ложь":
  - сохранение U,A;
  - обнуление буфера;
- Проход не 1 и не 2 и УСОР1 = "Истина":
  - вычитывание буфера;
  - вычитать предыдущий результат;
  - произвести расчет параметров модели;
  - записать результат;
  - сохранить буфера;
- сохранение U,A.

## А.6 Таймеры

1. Таймер циклический
2. Таймер астрономический
3. Хронограф
4. Циклограмма
5. Управление оборудованием
6. Командный таймер

Для операций **таймер циклический**, **таймер астрономический** возможно использование **проверки исполнения по условиям результата**.

### А.6.1 Таймер циклический

Включение и отключение таймера производится по значению 1-го элемента условия операнда 1.

Таймер может управляться как импульсно, так и по значению.

Импульсное управление включением/отключением таймера реализовано по переднему фронту импульса (переход "0"-"1"). При необходимости использовать задний фронт (переход "1"-"0"):

- или установить отрицание первого элемента условия операнда 1;
- или в поле "П" изменить проверяемое значение с "И" (**истина**) (по умолчанию) на "Л" (**ложь**).

2-й элемент условия операнда 1 управляет включением паузы.

**Операнд 1** - буфер длительностей шагов (набор засечек времени в сек.).

**Операнд 2** - буфер значений, выдаваемых на выход.

**Параметр 1** - тип таймера:

- = 0 - однократный таймер с потенциальным управлением;
- = 1 - циклический таймер с потенциальным управлением;
- = 2 - однократный таймер с импульсным управлением;
- = 3 - циклический таймер с импульсным управлением;
- = 4..7 - аналогично значениям 0..3 соответственно с функцией "самоподхвата" (игнорирование команды остановки таймера);
- = 10..17 - аналогично значениям 0..7 соответственно, но выход меняется линейно от текущего значения к следующему.

**Параметр 2** = N - количество полных циклов исполнения для циклического таймера;

По завершении при потенциальном управлении флаг запуска сбрасывается.

**Параметр 3** = S - Для циклического таймера: число тактов исполнения на последнем цикле (размерность буфера шагов).

Варианты для циклического таймера

---

| Параметр 2 | Параметр 3 | Таймер             |
|------------|------------|--------------------|
| 0          | 0          | Бесконечный таймер |

|   |   |                                                     |
|---|---|-----------------------------------------------------|
| N | 0 | Конечный таймер: N циклов                           |
| N | S | Конечный таймер: N циклов<br>+ S шагов на N+1 цикле |
| 0 | S | Конечный таймер: 0 циклов<br>+ S шагов              |

Для всех конечных циклических таймеров и однократных таймеров при потенциальном управлении производится сброс флага включения (Условие операнда 1).

В условия результата записываются (в условии результата д.б. включена опция "Запись"):

- в 1-й элемент – счетчик шагов;
- во 2-й элемент – счетчик циклов.

Условия операнда 2 используются для маскирования выхода:

- если соответствующее значение условия ОП2 равно **Истине**, то значение на выход НЕ выдается.

Если результат имеет тип скаляр, то условия ОП2 управляют маскированием выхода по шагам цикла.

Если результат имеет тип вектор или матрица, то действия по маскированию определяются соответствием размерностей ОП2 и условий ОП2. Например, если ОП2 - матрица (2,3) (2 выхода по 3 шага) и условия ОП2 также имеют размерность матрицы (2,3), то происходит индивидуальное маскирование каждого выхода по шагам, если же условия имеют размерность вектора (2), то происходит маскирование по выходам независимо от шага.

### Алгоритм работы таймера

П1: Проверить состояние таймера:

Отключен - П2, Включен - П4, Пауза - П16

П2: Проверить управление на включение таймера:

Есть - П3, Нет - П99(Завершение)

П3: Установить Цикл=1, Шаг=1;

переход к П18

П4: Проверить управление на отключение:

Есть - П5, Нет - П6

П5: Завершение работы таймера;

Для потенциально-управляемого таймера - снять управление

Переход к П99

П6: Проверить тайм-аут:

Не истек - П7, истек - П10

П7: Проверить управление на паузу

Пауза есть - П8, нет - П9

П8: Коррекция времени засечки;

Переход к П99

П9: Повторить текущий выходной сигнал

Переход к П99

П10: Есть новый шаг:

Да - П11, Нет - П12

П11: Шаг=Шаг+1;

Переход к П18  
П12: Есть ли новый цикл:  
Есть - П13, Нет - П14  
П13: Цикл=Цикл+1, Шаг=1  
Переход к П18  
П14: Есть ли выдача по завершению работы таймера:  
Нет - П5, Есть - П15  
П15: Выдать выходной сигнал;  
Переход к П5  
П16: Проверить снятие паузы:  
Не снята - П99, Снята - П17  
П17: Восстановить параметры таймера;  
Переход к П18  
П18: Установить тайм-аут  
Переход к П19  
П19: Проверить команду установки Паузы  
Есть Пауза - П8, нет - П20  
П20: Выдать выходной сигнал  
Переход к П99  
П99: Завершение работы на данном такте

#### Алгоритм с учетом размерностей

**Операнд 1** - Буфер засечек времени.

**Операнд 2** - Буфер выходных сигналов.

##### 1. Результат - скаляр

|                 | Операнд 1 | Скаляр Ct | Вектор Bt(N) | Матрица Mt(N,M) |
|-----------------|-----------|-----------|--------------|-----------------|
| Операнд 2       |           |           |              |                 |
| Скаляр Cz       |           | 1         | 2            | 3               |
| Вектор Vz(K)    |           | 4         | 5            | 6               |
| Матрица Mz(K,L) |           | 7         | 8            | 9               |

Сл.1 обычная операция: Засечка на Ст с выдачей Cz; после завершения времени выдача прекращается;  
Сл.4 сначала аналог п.1 (выдается Vz(1)), после завершения засечки выдается Vz(2),если существует;  
Сл.7 используется 1-й вектор аналогично п.4 (Выдаются Mz(1,1) и Mz(2,1));  
Сл.2 засечка на суммарное время (по всему вектору) - аналог п.1;  
Сл.5 последовательная выдача значений Vz(1)...Vz(min(K,N)) по временам Bt(1)...Bt(min(K,N)). Если после завершения есть еще элементы для выдачи (K>N), то выдается Vz(N+1);  
Сл.8 аналог сл.5 - используется только 1-й вектор матрицы значений Mz(1,1)...Mz(min(K,N),1). Если (K>N)- в конце выдается Mz(N+1,1);  
Случаи 3,6,9 аналогичны 2,5,8 - используется только 1-й вектор матрицы.

##### 2. Результат - Вектор(Bp(R))

|  | Операнд 1 | Скаляр Ct | Вектор Bt(N) | Матрица |
|--|-----------|-----------|--------------|---------|
|--|-----------|-----------|--------------|---------|



|                 |  |    |    |         |
|-----------------|--|----|----|---------|
| Операнд 2       |  |    |    | Mt(N,M) |
| Скаляр Сз       |  | 11 | 12 | 13      |
| Вектор Вз(K)    |  | 14 | 15 | 16      |
| Матрица Мз(K,L) |  | 17 | 18 | 19      |

Сл.11 обычная операция: Аналогично п.1 с тиражированием;

Сл.14 выдача из Вз(1) на время работы таймера и Вз(2) по завершении;

Сл.17 используется 1-й вектор (Выдается Мз(1..Min(K,R),1). Если есть 2-й вектор, то он выдается по завершению засечки времени;

Сл.12 засечка на суммарное время (по всему вектору) - аналог п.4 с тиражированием;

Сл.15 поэлементная выдача из Вз(1..min(K,N)) по временам Вт(1..min(K,N));

Сл.18 по каждой засечке времени Вт(1..min(N,L)) выдается вектор Мз(min(K,R),1..min(N,L)). Если L>N ,то после последней засечки выдается вектор Мз(min(K,R),N+1);

Случаи 13,16,19 аналогичны 12,15,18 – используется только 1-й вектор матрицы. Для случаев 16,19 производится свертка матрицы Оп1 к вектору суммированием по 2-й размерности.

### 3. Результат – Матрица (Mp(R,T))

Все делается аналогично случаю, когда результат - вектор с добавлением тиражирования:  
тиражирование векторов для случаев 17-19.

#### А.6.2 Таймер астрономический

Данный таймер позволяет формировать выходной сигнал по наступлению указанной астрономической даты и времени.

**Операнд 1** - дата установки выхода в **Истину**.

**Операнд 2** - дата сброса выхода в **Ложь**.

Условие операнда 1 используется для запуска таймера (потенциальное управление). Таймер работает только, если соответствующее условие операнда 1 истинно.

Параметры алгоблока не используются.

Задание даты производится вектором из 6 целых чисел:

(1) (2) (3) (4) (5) (6)

секунда минута час день месяц год

допустимые значения:

0-59 0-59 0-23 1-кол-во 1-12 2000-бесконечность  
дней в месяце

Если вектор короче 6 элементов - то отбрасываются старшие, начиная с года. Если значение элемента выходит за пределы допустимости, то такие элементы также отбрасываются. Отброшенные элементы даты в момент исполнения подменяются текущими значениями даты.

| Обработка дат с отброшенными элементами (-) |        |     |      |       |     |       |
|---------------------------------------------|--------|-----|------|-------|-----|-------|
| Секунд                                      | минута | час | день | месяц | Год | Выход |

|   |   |   |   |   |   |                                        |
|---|---|---|---|---|---|----------------------------------------|
| a |   |   |   |   |   |                                        |
| + | + | + | + | + | - | Установка 1 раз<br>каждый год          |
| + | + | + | + | - | - | Установка 12 раз в год<br>каждый месяц |
| + | + | + | - | - | - | Ежедневная установка                   |
| + | + | - | - | - | - | Ежечасная установка                    |
| + | - | - | - | - | - | Ежеминутная установка                  |

Ежесекундная установка возможна при задании единичного вектора (аналогично ежеминутной установке) и указанном количестве секунд вне диапазона 0-59.

**Пример** с отброшенными внутри элементами:

+ + + - + - Установка выхода 1 раз  
в день (по чч.мм.сс) для указанного месяца.

2-й операнд всегда согласуется по размерности с 1-м операндом:

- если соответствующий вектор даты/времени 2-го операнда больше 1-го, то он обрезается, начиная со старших элементов (год и т.д.);
- если соответствующий вектор даты/времени 2-го операнда меньше 1-го, то недостающие элементы вектора добавляются из вектора операнда 1, начиная со старших.

**Алгоритм работы для одного таймера (T1 - дата/время старта из ОП1, T2 - дата/время останова из ОП2), ТС - текущее дата/время, ТП - предыдущее дата/время запуска алгоблока**

- если  $T2 \leq T1$  - формирование импульсного выхода:
  - если  $ТС \geq T1$ , то:
    - если  $ТП < T1$  - установить "1" на выход;
    - если  $ТП \geq T1$  - установить "0" на выход ("1" была установлена на предыдущем такте); "0" будет далее постоянно устанавливаться до перехода на новый интервал взвода таймера (например, для секундного таймера до наступления следующей минуты);
    - если  $ТС < T1$  - ожидание времени старта;
- если  $T2 > T1$  - формирование потенциального (длинный импульс) выхода:
  - если  $ТС \geq T1$  и  $ТС < T2$  - установить "1" на выходе;
  - если  $ТС \geq T1$  и  $ТС \geq T2$  - сбросить выход в "0";
  - если  $ТС < T1$  - ожидание времени старта.

#### **Работа астрономического таймера с учетом размерностей**

Во всех случаях для операндов 1 и 2 скаляр рассматривается как вектор единичной размерности (заданы секунды). Определяется количество таймеров КТ.

## 1. Операнд 1 - вектор $V1(n1)$

---

Результат

Скаляр -  $KT=1$ .

Вектор, Матрица - аналогично случаю со скаляром с тиражированием результата.

В операнде 2 всегда используется 1-й вектор или скаляр для указания времени сброса выхода.

## 2. Операнд 1 - матрица $M1(n1,m1)$

---

Результат

Скаляр  $C3$  -  $KT=m1$  (2-й размерности операнда 1); происходит многократное управление одной переменной;

Вектор  $V3(n3)$  -  $KT=\min(n3,m1)$

Матрица  $M3(n3,m3)$  - аналог случая для вектора с тиражированием на матрицу.

Во всех случаях операнд 2 определяет количество точек сброса:

ОП2 -  $C, V2(n2)$ : - одинаковое время стопа для всех  $KT$  таймеров;

-  $M2(n2,m2)$  - количество точек сброса определяется 2-й размерностью матрицы; в случае, если  $m2 < KT$  для всех таймеров, больше  $m2$  используется последняя точка сброса.

### Проверка условий операнда 1 для включения и отключения таймеров

Операнд 1

$C$  - обычная свертка  $услОп1$  к скаляру;

$V(n1)$  - при проверке вектор подменяется скаляром и далее производится обработка как для скаляра;

$M(n1,m1)$  - при проверке матрица подменяется вектором  $V(m1)$  и вектор проверяется поэлементно.

### Перспектива

---

Параметр 1

= 0 - 4-й элемент вектора даты/времени - сутки

=  $N > 0$  - 4-й элемент вектора даты/времени -  $N$  часов в сутках

Параметр 2

= 0 - 5-й элемент вектора даты/времени - месяц

=  $N > 0$  - 5-й элемент вектора даты/времени -  $N$  суток в месяце

=  $N < 0$  :

= -1 - недели

= -2 - декады

= -3 - кварталы

= -4 - полугодия

= -5 - годы

### А.6.3 Хронограф

Хронограф позволяет фиксировать интервалы времени.

**Операнд 1** - источник включения хронографа; включение нового подсчета времени может происходить или:

А. - по изменению входного сигнала (импульс на входе);

Б. - по наличию определенного значения на входе (плато или потенциал);

**Операнд 2** - см. далее в зависимости от режима работы;

**Результат:** подсчитанное время в видах:

1) - вектор времени в формате:

(1) эл-т вектора - мсек, (2) - сек, (3) - мин

и т.д. до Лет; максимальная используемая размерность вектора - 7

или

2) - вектор времени, образующий кольцевой буфер по

размерности вектора, где каждый элемент представляет

собой подсчитанное время в мсек за указанный операндом1

интервал.

**Параметр 1** - режим работы и представления выходных данных:

= 0 - включение нового подсчета по импульсу на входе Оп1

(вар.А); предыдущее значение ОП1 запоминается в ОП2;

результат (2)) представлен в виде кольцевого буфера

времен - при переходе к новому интервалу

происходит сдвиг вперед по вектору, заполняется

всегда 1-й элемент вектора;

= 1 - включение нового подсчета при попадании на заданное

плато на входе Оп1 (вар.Б); Операнд 2 используется для

фиксации попадания Оп1 на плато;

результат (2)) представлен в виде кольцевого буфера

времен - при переходе к новому интервалу

происходит сдвиг вперед по вектору, заполняется

всегда 1-й элемент вектора;

= 2 - подсчет времени ведется всегда при нахождении Оп1 на

заданном плато на входе Оп1 (вар.Б); Обнуление

выходного вектора времени (для вар.1)) или переход

со сдвигом к новому интервалу (для вар.2))

производится по наличию "1" в соответствующем элементе

операнда 2 (Операнд 2 **не используется** для фиксации

нахождения ОП1 на плато);

результат представлен в виде кольцевого буфера

времен - при переходе к новому интервалу происходит

сдвиг вперед по вектору, заполняется всегда 1-й

элемент вектора;

= 10,11,12 - аналогичны соответственно случаям для значений

0,1,2 соответственно. Отличие в том, что результатом

является вектор времени (вар.1)).

**Параметр 2** - = Delta;

для варианта А:

величина, определяющая наличие импульса на входе ОП1 (изменение значения).

**Параметр 3** - = Plato - значение плато.

Добавление времени в результат производится:

- для Пар1=0,10 - всегда

- для Пар1=1,2,11,12 - при нахождении на плато (для Delta < 0 - вне плато);

**Фиксация изменения значения для варианта А**

- Параметр 1 = 0 :

- если Plato = 0 , то:

- если Delta >= 0 , то:

- если  $\text{abs}(\text{ТЗОП1} - \text{ПЗОП1}) > \text{abs}(\text{Delta})$  - ЕСТЬ изменение

- если Delta < 0 , то:

- если  $\text{abs}(\text{ТЗОП1} - \text{ПЗОП1}) > 10 * \text{abs}(\text{Delta})$  , то

ЕСТЬ изменение БЕЗ сдвига КБ

- иначе:

- если  $\text{abs}(\text{ТЗОП1} - \text{ПЗОП1}) > \text{abs}(\text{Delta})$  - ЕСТЬ  
изменение

- если Plato ≠ 0 , то:

- если  $\text{abs}(\text{Plato}) > \text{abs}(\text{Delta})$  , то

- если  $\text{abs}(\text{ТЗОП1} - \text{ПЗОП1}) > \text{abs}(\text{Plato})$  , то ЕСТЬ изменение  
БЕЗ сдвига КБ

- иначе:

- если Plato > 0 , то

- если  $\text{ТЗОП1} - \text{ПЗОП1} > \text{Delta}$  - ЕСТЬ изменение

- если Plato <= 0 , то

- если  $\text{ПЗОП1} - \text{ТЗОП1} > \text{Delta}$  - ЕСТЬ изменение

- если  $\text{abs}(\text{Plato}) \leq \text{abs}(\text{Delta})$  , то

- если Plato > 0 , то

- если  $\text{ТЗОП1} - \text{ПЗОП1} > \text{Delta}$  - ЕСТЬ изменение

- если Plato <= 0 , то

- если  $\text{ПЗОП1} - \text{ТЗОП1} > \text{Delta}$  - ЕСТЬ изменение

- Параметр 1 = 10 :

- если Plato = 0 , то

- если  $\text{abs}(\text{ТЗОП1} - \text{ПЗОП1}) > \text{abs}(\text{Delta})$  - ЕСТЬ изменение

- если Plato > 0 , то

- если  $\text{ТЗОП1} - \text{ПЗОП1} > \text{Delta}$  - ЕСТЬ изменение

- если Plato < 0 , то

- если  $\text{ПЗОП1} - \text{ТЗОП1} > \text{Delta}$  - ЕСТЬ изменение

**Фиксация попадания на плато для варианта Б (для Пар1= 1,2,11,12)**

- если  $\Delta > 0$ :
    - если  $\text{ТЗОП1} \geq \text{Plato} - \Delta$   
И
    - если  $\text{ТЗОП1} \leq \text{Plato} + \Delta$   
то значение ОП1 находится на плато и время добавляется.
  - если  $\Delta < 0$ :
    - если  $\text{ТЗОП1} > \text{Plato} + \text{abs}(\Delta)$   
ИЛИ
    - если  $\text{ТЗОП1} < \text{Plato} - \text{abs}(\Delta)$   
то значение ОП1 находится вне плато и время добавляется.
  - если  $\Delta = 0$ :
    - если  $\text{ТЗОП1} > \text{Plato}$   
то значение ОП1 находится на плато и время добавляется.
- , где      ТЗОП1 - текущее значение ОП1;  
              ПЗОП1 - предыдущее значение ОП1  
                       (сохранено в ОП2);  
              abs - модуль.

#### **Сдвиги и обнуления:**

- для случая Пар1=0 сдвиг кольцевого буфера времен производится каждый раз при изменении значения на входе ОП1;
- для случая Пар1=10 обнуление вектора времени производится каждый раз при изменении значения на входе ОП1;
- для случая Пар1=1 сдвиг кольцевого буфера на выходе производится при попадании на плато (для  $\Delta < 0$  - при ухода с плато);
- для случая Пар1=11 обнуление выходного вектора времени производится при попадании на плато (для  $\Delta < 0$  - при ухода с плато);
- для случая Пар1=2 сдвиг кольцевого буфера на выходе производится при наличии в соответствующем элементе ОП2 "1";
- для случая Пар1=12 обнуление выходного вектора времени производится при наличии в соответствующем элементе ОП2 "1".

Для режимов Параметр1 = 2,12 - **ВСЕ элементы ОП2 при выходе обнуляются !**

#### **Структурирование размерностей; определение количества хронографов КН**

КБ - кольцевой буфер; РВР - размерность вектора времени

1. Оп1 - скаляр С1 - всегда 1 хронограф

---

|  |           |           |        |         |
|--|-----------|-----------|--------|---------|
|  | Результат | Скаляр С3 | Вектор | Матрица |
|--|-----------|-----------|--------|---------|

|                     |  |   |          |             |
|---------------------|--|---|----------|-------------|
| Операнд 2           |  |   | $V3(n3)$ | $M3(n3,m3)$ |
| Скаляр $C2$         |  | 1 | 2        | 3           |
| Вектор $V2(n2)$     |  | 4 | 5        | 6           |
| Матрица $M2(n1,m2)$ |  | 7 | 8        | 9           |

| Случай | Параметр 1 |            |             |             |
|--------|------------|------------|-------------|-------------|
|        | 0/1        | 2          | 10/11       | 12          |
| 1      | 1 КБ=1     | 1 КБ=1     | 1 PBP=1     | 1 PBP=1     |
| 2      | 1 КБ=n3    | 1 КБ=n3    | 1 PBP=n3    | 1 PBP=n3    |
| 3      | 1 КБ=n3+тм | 1 КБ=n3+тм | 1 PBP=n3+тм | 1 PBP=n3+тм |
| 4      | 1 КБ=1     | 1 КБ=1     | 1 PBP=1     | 1 PBP=1     |
| 5      | 1 КБ=n3    | 1 КБ=n3    | 1 PBP=n3    | 1 PBP=n3    |
| 6      | 1 КБ=n3+тм | 1 КБ=n3+тм | 1 PBP=n3+тм | 1 PBP=n3    |
| 7      | 1 КБ=1     | 1 КБ=1     | 1 PBP=1     | 1 PBP=1     |
| 8      | =5         | =5         | =5          | =5          |
| 9      | =6         | =6         | =6          | =6          |

## 2. Оп1 - вектор $V1(n1)$

---

|                     | Результат | Скаляр $C3$ | Вектор $V3(n3)$ | Матрица $M3(n3,m3)$ |
|---------------------|-----------|-------------|-----------------|---------------------|
| Операнд 2           |           |             |                 |                     |
| Скаляр $C2$         |           | 11          | 12              | 13                  |
| Вектор $V2(n2)$     |           | 14          | 15              | 16                  |
| Матрица $M2(n1,m2)$ |           | 17          | 18              | 19                  |

| Случай | Параметр 1                            |                                  |                           |                      |
|--------|---------------------------------------|----------------------------------|---------------------------|----------------------|
|        | 0/1                                   | 2                                | 10/11                     | 12                   |
| 11     | 1 КБ=1                                | 1 КБ=1                           | 1 PBP=1                   | 1 PBP=1              |
| 12     | 1 КБ=n3                               | 1 КБ=n3                          | 1 PBP=n3                  | 1 PBP=n3             |
| 13     | 1 КБ=n3+тм                            | Min(n1,m3)<br>КБ=n3              | 1 PBP=n3+тм               | Min(n1,m3)<br>PBP=n3 |
| 14     | 1 КБ=1                                | 1 КБ=1                           | 1 PBP=1                   | 1 PBP=1              |
| 15     | Min(n1,n2,n3)<br>) КБ=1 без<br>сдвига | Min(n1,n3)<br>КБ=1 без<br>сдвига | 1 PBP=n3                  | 1 PBP=n3             |
| 16     | Min(n1,n2,m3)<br>) КБ=n3              | Min(n1,m3)<br>КБ=n3              | Min(n1,n2,m3)<br>) PBP=n3 | Min(n1,m3)<br>PBP=n3 |
| 17     | 1 КБ=1                                | 1 КБ=1                           | 1 PBP=1                   | 1 PBP=1              |
| 18     | =15                                   | =15                              | =15                       | =15                  |
| 19     | =16                                   | =16                              | =16                       | =16                  |

## 3. Оп1 - матрица $M1(n1,m1)$

---

Аналог случая для вектора - используется только 1-й вектор матрицы.

#### А.6.4 Циклограмма

Данный алгоблок предназначен для организации циклического управления оборудованием с ожиданием срабатывания и контролем по времени.

**Операнд 1** - засечки времен (задаются вектором) и обработка ошибок (если размерность операнда равна матрице). Для засечек времен используется 1-й вектор матрицы (или просто вектор, если операнд не является матрицей):

- если значение элемента вектора  $T > 0$ , то данный шаг полностью выдерживается по времени независимо от срабатывания оборудования;
- если значение элемента вектора  $T < 0$ , то переход к следующему шагу осуществляется сразу по срабатывании оборудования, не дожидаясь выдержки времени.

Если операнд определен как матрица, то:

- второй вектор используется для организации обработки ошибок в случае их возникновения на данном шаге (не срабатывания оборудования); соответствующий элемент этого вектора  $lerrb$  определяет:
  - если  $lerrb > 0$  - следующий исполняемый шаг; (шаг завершения определяется соответствующим элементом 3-го вектора  $lerre$  (см. ниже); если 3-й вектор не задан, то шагом завершения является максимально-возможный шаг);
  - если  $lerrb \leq 0$  - циклограмма завершается; при этом, если  $lerrb < 0$ , то это значение является кодом завершения; если  $lerrb = 0$ , то кодом завершения является соответствующее значение из 3-его вектора операнда  $lerre$  (если он задан).
- третий вектор определяет:
  - или последний исполняемый шаг при обработке ошибок на данном основном шаге программы (если  $lerrb > 0$ );
  - или код завершения по ошибке (если  $lerrb = 0$ ).

Условие операнда 1:

- (1-й элемент) - управление включением ("**Истина**") и отключением ("**Ложь**");
- (2-й элемент) - управление паузой ("**Истина**" - включена пауза);

**Операнд 2** - матрица управлений (значения для подачи на выход; при этом, если значение равно -1 - управление не подается); значения на выход подаются однократно при начале текущего шага циклограммы;

Расширение Операнда 2 - матрица ожидаемых значений после подачи управления на данном шаге:

- если указано значение -1, то анализ для данного сигнала на данном шаге не проводится;
- если указано значение  $< -1$  (-2,...), то данный сигнал рассматривается на данном шаге как аварийный: при наличии в нем значения "**Истина**" ( $>0$ ) диагностируется ошибка и программа переходит к обработке ошибочной ситуации аналогично случаю истечения таймаута.

**Результат** - вектор выходов для подачи управления;

Расширение Результата - вектор входов для анализа срабатывания.

Если **Параметр 1**  $> 0$  и **Параметр 2**  $> 0$  - циклограмма выполняется однократно с шага П1 до шага П2;



Если **Параметр 1** > 0 и **Параметр 2** = 0 - выполняется П1 циклов (количество шагов цикла определяется по размерностям операндов и результата);

Если **Параметр 1** = 0 (**Параметр 2** - любой) - бесконечное исполнение циклограммы;

**Параметр 3** - определяет расположение итогового элемента (определяется БД и номером в ней); в итоговый может элемент записываться:

- <N> (от 1 до максимального количества шагов) - номер текущего исполняемого шага цикла - при исполнении циклограммы;

- <-N> (от 1 до максимального количества шагов) - отрицательный номер последнего исполняемого шага - при возникновении ошибки на данном шаге (не срабатывание оборудования);

- <N+1> - номер последнего исполненного шага +1 - при нормальном завершении циклограммы;

- <N+2> - при прекращении действия алгоблока оператором (сброс управления УсОп1(1)).

Определение максимального количества шагов - см. ниже.

При завершении исполнения циклограммы всегда сбрасывается флаг включения (УсОп1(1)).

### Структурирование размерностей

Оп1 - В1(n1) - кол-во меток времени

Оп2 - М2(n2,l2) - матрица управлений:

1...N2 - кол-во элементов управления

1...L2 - кол-во шагов по времени

РасшОп2 - МР2(nr2,lr2) - матрица ожидаемых значений:

1...Nr2 - количество элементов управления

1...Lr2 - кол-во шагов по времени

Рез - В3(n3) - вектор выходов для управления

РасшРез - ВР3(nr3) - вектор входов для анализа

Количество шагов по времени определяется:  $N = \min(n1, l2, lr2)$

Количество управляемых элементов определяется:  $U = \min(n2, n3)$

Количество анализируемых входных данных:  $A = \min(nr2, nr3)$

### Пример

#### 1. Пример 1

4 шага во времени            n1,l2,lr2=4

3 управляющих сигнала        n2,n3=3

5 анализируемых сигналов     nr2,nr3=5

В1=-5,-10,-15,30        - время в сек (последний шаг всегда выдерживается полностью)

Оп2(3,4)=    1  0  0 - 1-й шаг

0  1  0 - 2-й

0  0  1 - 3-й

-1 -1 -1 - 4-й (нет управления)

РасшОп2(5,4)= 1  0  0 -1 -1 - анализ на 1-м шаге

0  1  0 -1 -1 - 2-м

0 0 1 1 1 - 3-м (срабатывание 3-х сигналов)  
 -1 -1 -1 -1 -1 - 4-м (нет анализа - просто выдержка времени).

## 2. Пример 2 (с заданием обработки ошибок)

Операнд 1 определен матрицей M1(4,3)  
 4 шага во времени n1,l2,lr2=4  
 4 управляющих сигнала n2,n3=4  
 4 анализируемых сигналов nr2,nr3=4

Параметрами П1 и П2 задаются шаги исполнения основного цикла: с 1 по 3. 4-й шаг используется для обработки ошибок.

Оп1(4,3)=-5,-10,-15,10 - засечки времени в сек (последний шаг всегда выдерживается полностью);  
 4 4 0 -99 - 2-й вектор определяет:  
 - переход на 4-й шаг при возникновении ошибки на шагах 1 и 2;  
 - при ошибке на шаге 3 циклограмма завершается (исп-ся код ошибки из 3-его вектора -98);  
 - код завершения (-99) при ошибке на шаге 4;  
 4 4 -98 0 - шаги завершения (4) при возникновении ошибки на 1 и 2 шагах;  
 - код ошибки (-98) при ошибке на 3-м шаге.  
 Оп2(4,4)= 1 0 0 0 - 1-й шаг  
 0 1 0 0 - 2-й  
 0 0 1 1 - 3-й  
 0 0 0 0 - 4-й (сброс управления)  
 РасшОп2(4,4)= 1 0 0 -1 - анализ на 1-м шаге  
 0 1 0 -1 - 2-м  
 0 0 1 1 - 3-м (срабатывание 2-х сигналов)  
 -1 -1 -1 -1 - 4-м (нет анализа - просто выдержка времени).

## А.6.5 Управление оборудованием

Алгоблок предназначен для управления задвижками, клапанами, вентилями, насосами, двигателями.

**Операнд 1** - определяет сигналы телесигнализации о состоянии оборудования;

**Операнд 2** - ожидаемые значения по окончании управления;

**Результат** - определяет сигналы телеуправления для выдачи команд и сводный сигнал диагностики управления;

Условие операнда 1 (1-й элемент) - включение/отключение алгоблока;

**Параметр 1** - время ожидания срабатывания оборудования (сек);

**Параметр 2** - значение управления для выдачи в результат:  
 - устанавливается однократно при 1-м проходе;  
 - сбрасывается в зависимости от Параметра 3;

**Параметр 3** - флаг необходимости сброса управления:

= 0 - сбрасывать ненулевое управление в 0 при достижении ожидаемого состояния;

=1 – сбрасывать ненулевое управление в 0 только, если все устройства достигли ожидаемого состояния или по завершении времени управления;  
= 2 - не сбрасывать управление.

Для данного алгоблока возможно указание выборки по операнду 1 и проверки исполнения по условиям результата.

#### **Алгоритм работы**

1. По включению алгоблока (УсОп1) установить таймаут (П1).
2. Для каждого устройства:
  - проверить состояние (Оп1) на равенство ожидаемому значению (Оп2):
    - если равны, то:
      - если ПЗ равен 0 выдать на выход «0»;
      - если не равны, то выдать на выход значение из П2(только на 1-м проходе);
3. Если по завершении цикла по всем устройствам состояния их равны ожидаемому, то:
  - сбросить управление алгоблоком в «0» (УсОп1);
  - если в результате есть сводный сигнал диагностики, то в него записывается «1»;
  - если ПЗ=1, то произвести сброс управления для всех устройств;
4. Если по завершении цикла остались устройства с неустановившимся состоянием, то проверить таймаут:
  - если истек, то:
    - сбросить управление алгоблоком в «0» (УсОп1);
    - если ПЗ=1, то произвести сброс управления для всех устройств;
    - если в результате есть сводный сигнал диагностики, то в него записывается «-1»;
  - если нет, то:
    - если в результате есть сводный сигнал, то в него записывается «0»;
5. Если алгоблок отключается извне (по УсОп1), то:
  - ненулевое управление сбрасывается в 0 (при ПЗ#2);
  - в сводный сигнал диагностики записывается значение «2».

#### **Структурирование размерностей**

- Работа производится по минимальным размерностям операнда 1 и результата;
- Свободный элемент результата (сводный сигнал диагностики управления) определяется:
  - если 1-я размерность результата больше минимума (1-я размерность операнда 1 и 1-я размерность результата) - свободный элемент в 1-й размерности;
  - иначе:
    - если 2-я размерность результата больше минимума (2-я размерность операнда 1 и 2-я размерность результата) - свободный элемент во 2-й размерности.

### А.6.6 Командный таймер

Данный алгоблок предназначен для организации последовательных цепочек управления оборудованием с ожиданием подтверждения срабатывания.

**Операнд 1** - время шагов в секундах (может задаваться меньшим количеством, чем вектор команд и подтверждений - в этом случае предполагается одинаковое время);

Расширение/Условие Оп1 - вектор команд (0/1) по шагам;

**Операнд 2** - подтверждение обработки;

Расширение/Условие Оп2 - разрешение выполнения команды (может не задаваться; по умолчанию считается, что разрешение есть);

Результат - выходы, устанавливаемые при нормальном завершении шага;

Расширение/Условие результата - выходы, устанавливаемые при ненормальном завершении шага.

**Параметр 1** - сигнал **Пауза** - при возникновении данного сигнала алгоблок сохраняет все значения (в т.ч. величину таймаута) и никаких действий не производит до снятия сигнала **Пауза**;

**Параметр 2** - действия при исчезновении сигнала разрешения:

= 0 - нет действий

= 1 - взвести соответствующий сигнал РРез/УСРез

= 2 - взвести соответствующий сигнал Рез

= 3 - взвести соответствующие сигналы РРез/УСРез и Рез

**Параметр 3** - действия при исчезновении сигнала команды:

= 0 - нет действий

= 1 - взвести соответствующий сигнал РРез/УСРез

= 2 - взвести соответствующий сигнал Рез

= 3 - взвести соответствующие сигналы РРез/УСРез и Рез

#### Алгоритм

При возникновении команды "1" в условии/расширении Оп1 и наличии разрешения в условии/расширении Оп2 включается таймер (на время, определенное в ОП1). В течении работы таймера ожидается приход подтверждения ("1" в ОП2). Если подтверждение уже есть, то таймер не включается, команда сбрасывается в "0" и в соответствующий элемент Рез устанавливается "1".

Если пришло подтверждение, то команда сбрасывается в "0" и в соответствующий элемент Рез устанавливается "1".

Если подтверждения за время работы таймера не пришло, то команда сбрасывается в "0" и "1" устанавливается в соответствующий элемент РРез.

Если во время отсчета таймаута пропало разрешение, то команда сбрасывается в "0", а дальнейшие действия алгоблока зависят от значения в Пар2.

Если во время отсчета таймаута пропала команда, то действия алгоблока зависят от значения в Пар3.

После завершения шага производится переход к следующему шагу.

#### Структурирование размерностей

Количество шагов таймера определяется по минимуму размерностей РОП1/УСОП1, ОП2, РЕЗ и РРЕЗ/УСРЕЗ. ОП1 (время) м.б. общее (одинаковое) для всех шагов; разрешение в РОП2/УСОП2 может не задаваться, тогда считается, что разрешение есть всегда.

#### А.6.7 Дата и время

Данный алгоблок предназначен для получения параметров текущей даты и времени, а также дня недели и номера недели в году.

Дата и время выдаются в результат в виде вектора максимум из 9 элементов. Элементы с 1 по 7 - дата и время (последовательность зависит от параметра 1). День недели (8-й элемент) и номер недели в году (9-й элемент) выдаются только, если в векторе результата есть место.

**Параметр 1** - режим выдачи даты времени:

= 0 - последовательность:

(1)-мсек., (2)-сек., (3)-мин., (4)-час,  
(5)-день, (6)-месяц, (7)-год;

# 0 - обратная последовательность:

(1)-год, (2)-месяц, (3)-день, (4)-час,  
(5)-мин., (6)-сек., (7)-мсек.

**Параметр 2** - режим выдачи времени:

= 0 - выдаются локальные дата и время;

= 1 - дата и время по Гринвичу (только для WIN);

= 2 - локальное поясное (зимнее) дата и время;

= -1 - локальное время (аналог режима 0) с учетом переходов зимнее/летнее при смещении даты/времени вследствие суммирования с операндами.

Если размерность вектора результата меньше 7, то производится усечение элементов даты/времени, начиная со старших элементов, например, для случая из 4 элементов и параметра 1 = 0 в результате получим вектор:

(1)-мсек.,(2)-сек,(3)-мин,(4)-час.

Операнды используются для получения смещенных векторов даты и времени.

Если в результате определен 1 вектор, то используется только 1-й операнд, если 2 и более, то для 1-го используется 1-й операнд для 2-го вектора - 2-й операнд.

**Операнд 1/2** - вектор смещений параметров даты и времени; последовательность параметров соответствует последовательности, определенной параметром 1; смещения могут задаваться со знаком, т.е. если значение параметра больше нуля, то производится увеличение соответствующего параметра (движение вперед от текущего времени), если меньше нуля, то производится уменьшение параметра (движение назад от текущего времени). Значения смещений **НЕ** ограничиваются диапазонами изменения соответствующих параметров.

### А.6.8 Интервал

Данный алгоблок предназначен для определения промежутков времени от начала и конца указанного интервала до текущего момента времени.

**Параметр 1** - диапазон отсчета интервалов:

- = 0 - год
- 1 - полугодие
- 2 - квартал
- 3 - месяц
- 4 - декада
- 5 - неделя
- 6 - сутки
- 7 - смена
- 8 - час
- 9 - минута

**Параметры 2 и 3** - используются для задания смены:

Пар2 - длительность смены в минутах

Пар3 - смещение от начала суток до начала 1-й смены в минутах

В остальных случаях параметры 2 и 3 не используются.

**Операнд 1** - кратный интервал; задается в виде вектора вида:

- (1) - секунды
- (2) - минуты
- (3) - часы
- (4) - сутки

Значения элементов не ограничены;

Суммарное ограничение: размер текущего года, если размерность вектора меньше 4, то отбрасываются старшие элементы, начиная с суток.

**Операнд 2** - одиночный интервал; задается кодировкой:

- 1 - минута
- 2 - час
- 3 - смена
- 4 - сутки
- 5 - неделя
- 6 - декада
- 7 - месяц
- 8 - квартал
- 9 - полугодие
- 10 - год

Если задан Оп2 (1..10) - то используется только указанный текущий интервал: текущая минута ... текущий год; Оп1 и Параметры НЕ используются (кроме задания смены в параметрах 2 и 3).

Если Оп2 не задан - то интервал рассчитывается из Оп1; диапазон расположения интервалов определяется параметром 1. Если указанный интервал не кратен диапазону (Пар1), то последний интервал "обрезается" по границе диапазона. Если размер интервала превышает диапазон, то используется следующий (больший по размеру) диапазон.

**Результат:**

В результат записывается вектор (максимум из 4-х элементов) - остаток времени от текущего момента до конца текущего интервала.

В расширение результата (если оно задано) записывается вектор (максимум из 4-х элементов) - остаток времени от начала текущего интервала до текущего момента времени.

Формат вектора результата:

- (1) - секунды
- (2) - минуты
- (3) - часы
- (4) - сутки

Если старшие элементы не заданы, то они пересчитываются в следующий (младший по порядку) элемент.

## A.7 Регуляторы

1. ПТоTo2 П - регулятор
2. ПТоTo2 И - регулятор
3. ПТо Т П - регулятор
4. Пто Т И - регулятор
5. ПТо Т ПР - регулятор
6. ПТо Т ПРБ - регулятор

### A.7.1 ПТоTo2 П - регулятор

Регулятор с выходом на пропорциональный исполнительный механизм.

**Параметр 1** - пропорциональная составляющая.

**Параметр 2** - постоянная времени объекта (в сек.)

**Параметр 3** - 2-я постоянная времени объекта (в сек.)

**Операнд 1** - уставка;

**Операнд 2** - регулируемая величина;

Условие операнда 1:

1-й элемент - режим работы регулятора:

= 0 - ручной режим;

= 1 - автоматический режим.

2-й элемент - флаг расчета такта регулирования по постоянной времени объекта;

3-й..5-й элементы - управление идентификацией объекта:

3-й элемент - включение алгоритма идентификации;

4-й элемент - включение коррекции параметров регулятора;

5-й элемент - включение режима **Пауза**.

Условие операнда 2:

1-й элемент - режим безударного включения;

2-й ... - флаги учета возмущения.

Интеграл ошибки и предыдущее значение интегрального коэффициента хранятся внутри алгоблока.

**Использование операндов в регуляторах** – см.А.7.7

**Формирование выхода в регуляторах** – см.А.7.8

**Идентификация объекта управления** – см.А.7.9

**Изменение такта регулирования** – см.А.7.10

### A.7.2 ПТоTo2 И - регулятор

Регулятор с выходом на интегрирующий исполнительный механизм (выход на ШИМ).



**Параметр 1** - пропорциональная составляющая.  
**Параметр 2** - постоянная времени объекта (в сек.)  
**Параметр 3** - 2-я постоянная времени объекта (в сек.)  
**Операнд 1** - уставка;  
**Операнд 2** - регулируемая величина;

Условие операнда 1:

1-й элемент - режим работы регулятора:

- = 0 - ручной режим;
- = 1 - автоматический режим.

2-й элемент - флаг расчета такта регулирования по постоянной времени объекта;

3-й..5-й элементы - управление идентификацией объекта:

3-й элемент - включение алгоритма идентификации;

4-й элемент - включение коррекции параметров регулятора;

5-й элемент - включение режима **Пауза**.

Условие операнда 2:

1-й элемент - режим безударного включения;

2-й ... - флаги учета возмущения.

**Использование операндов в регуляторах** – см.А.7.7

**Формирование выхода в регуляторах** – см.А.7.8

**Идентификация объекта управления** – см.А.7.9

**Изменение такта регулирования** – см.А.7.10

### А.7.3 ПТО Т П - регулятор

Регулятор с выходом на пропорциональный исполнительный механизм с варьiruемым периодом вызова.

**Параметр 1** - пропорциональная составляющая.

**Параметр 2** - постоянная времени объекта (в сек.)

**Параметр 3** - :

>=0 - период вызова регулятора (в сек.);

< 0 и параметр определен через БД – параметр рассматривается как ссылка на область параметров:

- 2-я постоянная времени (если не равна 0);
- период вызова в сек.;
- коэффициент возмущения;
- коэффициент неравномерности;
- зона нечувствительности.

**Операнд 1** - уставка;

**Операнд 2** - регулируемая величина;

Условие операнда 1:

1-й элемент - режим работы регулятора:

= 0 - ручной режим;

= 1 - автоматический режим.

2-й элемент - флаг расчета такта регулирования по постоянной времени объекта (для данного регулятора **не используется**);

3-й..5-й элементы - управление идентификацией объекта:

3-й элемент - включение алгоритма идентификации;

4-й элемент - включение коррекции параметров регулятора;

5-й элемент - включение режима **Пауза**.

Условие операнда 2:

1-й элемент - режим безударного включения;

2-й ... - флаги учета возмущения.

Интеграл ошибки и предыдущее значение интегрального коэффициента хранятся внутри алгоблока.

**Использование операндов в регуляторах** – см.А.7.7

**Формирование выхода в регуляторах** – см.А.7.8

**Идентификация объекта управления** – см.А.7.9

#### А.7.4 ПТО Т И - регулятор

Регулятор с выходом на интегрирующий исполнительный механизм (выход на ШИМ) с варьируемым периодом вызова.

**Параметр 1** - пропорциональная составляющая.

**Параметр 2** - постоянная времени объекта (в сек.)

**Параметр 3** - :

>=0 - период вызова регулятора (в сек.);

< 0 и параметр определен через БД – параметр рассматривается как ссылка на область параметров:

- 2-я постоянная времени (если не равна 0);
- период вызова в сек.;
- коэффициент возмущения;
- коэффициент неравномерности;
- зона нечувствительности.

**Операнд 1** - уставка;

**Операнд 2** - регулируемая величина;

Условие операнда 1:

1-й элемент - режим работы регулятора:

= 0 - ручной режим;

= 1 - автоматический режим.

2-й элемент - флаг расчета такта регулирования по постоянной времени объекта (для данного регулятора **не используется**);

3-й..5-й элементы - управление идентификацией объекта:

3-й элемент - включение алгоритма идентификации;

4-й элемент - включение коррекции параметров регулятора;

5-й элемент - включение режима **Пауза**.

Условие операнда 2:

1-й элемент - режим безударного включения;

2-й ... - флаги учета возмущения.

**Использование операндов в регуляторах** – см.А.7.7

**Формирование выхода в регуляторах** – см.А.7.8

**Идентификация объекта управления** – см.А.7.9

#### А.7.5 ПТО Т ПР - регулятор

Регулятор с выходом на релейный исполнительный механизм.

**Параметр 1** - пропорциональная составляющая.

**Параметр 2** - постоянная времени объекта (в сек.)

**Параметр 3** - :

>=0 - период вызова регулятора (в сек.);

< 0 и параметр определен через БД – параметр рассматривается как ссылка на область параметров:

- 2-я постоянная времени (если не равна 0);
- период вызова в сек.;
- коэффициент возмущения;
- коэффициент неравномерности;
- зона нечувствительности.

**Операнд 1** - уставка;

**Операнд 2** - регулируемая величина;

Условие операнда 1:

1-й элемент - режим работы регулятора:

= 0 - ручной режим;

= 1 - автоматический режим.

2-й элемент - флаг расчета такта регулирования по постоянной времени объекта;

3-й..5-й элементы - управление идентификацией объекта:

3-й элемент - включение алгоритма идентификации;

4-й элемент - включение коррекции параметров регулятора;

5-й элемент - включение режима **Пауза**.

Условие операнда 2:

1-й элемент - режим безударного включения;

2-й ... - флаги учета возмущения.

Интеграл ошибки и предыдущее значение интегрального коэффициента хранятся внутри алгоблока.

**Использование операндов в регуляторах** – см.А.7.7

**Формирование выхода в регуляторах** – см.А.7.8

**Идентификация объекта управления** – см.А.7.9

**Изменение такта регулирования** – см.А.7.10

#### А.7.6 ПТО Т ПРБ - регулятор

Регулятор с выходом на релейный биполярный исполнительный механизм.

**Использование параметров операции:**

**Параметр 1** - пропорциональная составляющая.

**Параметр 2** - постоянная времени объекта (в сек.)

**Параметр 3** - :

>=0 - период вызова регулятора (в сек.);

< 0 и параметр определен через БД – параметр рассматривается как ссылка на область параметров:

- 2-я постоянная времени (если не равна 0);
- период вызова в сек.;
- коэффициент возмущения;
- коэффициент неравномерности;
- зона нечувствительности.

**Операнд 1** - уставка;

**Операнд 2** - регулируемая величина;

Условие операнда 1:

1-й элемент - режим работы регулятора:

= 0 - ручной режим;

= 1 - автоматический режим.

2-й элемент - флаг расчета такта регулирования по постоянной времени объекта;

3-й..5-й элементы - управление идентификацией объекта:

3-й элемент - включение алгоритма идентификации;

4-й элемент - включение коррекции параметров регулятора;

5-й элемент - включение режима **Пауза**.

Условие операнда 2:

1-й элемент - режим безударного включения;

2-й ... - флаги учета возмущения.

Интеграл ошибки и предыдущее значение интегрального коэффициента хранятся внутри алгоблока.

**Использование операндов в регуляторах** – см.А.7.7

**Формирование выхода в регуляторах** – см.А.7.8

**Идентификация объекта управления** – см.А.7.9

**Изменение такта регулирования** – см.А.7.10

### A.7.7 Использование операндов в регуляторах

**Операнд 1** – уставка; если задана вектором или матрицей, то приводится сложением к скаляру.

Формирование уставки суммой всех элементов операнда 1 регулируется значениями соответствующих элементов условия операнда 1 (кроме 1-го элемента). При значении условия "Истина" соответствующий элемент операнда 1 включается в сумму.

**НЕ** может быть константой.

**Операнд 2** - 1-й элемент - регулируемая величина; все остальные элементы, если есть считаются возмущением и приводятся к скаляру сложением.

Формирование возмущения суммой всех элементов операнда 2 регулируется значениями соответствующих элементов условия операнда 2 (начиная со 2-го элемента). При значении условия "Истина" соответствующий элемент операнда 2 включается в сумму.

Условие операнда 1 - режим работы регулятора:

= 0 - ручной режим;

= 1 - автоматический режим.

Режим определяется по состоянию 1-го элемента условия.

**В ручном режиме (1-й элемент - Ложь):**

- при включенном режиме безударного включения:

(режим безударного включения определяется значением 1-го элемента условия операнда 2):

- регулируемая величина (1-й элемент операнда 2) записывается в уставку (1-й элемент операнда 1);

Остальные элементы операнда 1 обнуляются.

- обнуляется также сохраняемое предыдущее значение ошибки.

Никаких других действий не производится;

Собственно, регулятор не работает.

**В автоматическом режиме (1-й элемент условия - Истина):**

- уставка формируется суммированием для тех элементов операнда 1, где соответствующее им условие = **Истина**;

- возмущение, если есть формируется суммированием;

- рассчитывается значение выхода.

Условие операнда 2 (1-й элемент) определяет режим безударного включения. Значение рассматривается **инверсно** (если значение равно "Ложь" - режим включен).

Если условие операнда 2 не включено, то режим безударного включения рассматривается как включенный.

#### **Работа регулятора при недостоверных входных данных**

1. Включение проверки входных данных на достоверность производится следующим образом:

- взвести флаги проверки статусов в условиях операндов 1 (уставка) и 2 (регулируемая величина и возмущение); флаги могут взводиться индивидуально для операндов 1 и 2; для условий операндов 1 и 2 необходимо также взвести общий флаг включения проверки по условиям;

- установить режим проверки флагами: "Люб" или "Оба" (установка в режим "НПР" отменяет проверку на достоверность !); при значении "Оба" проверяются и уставка и регулируемая величина (с возмущением); при значении "Люб" проверяется только регулируемая величина;
- установить флаг выполнимости операции в значение "Д" или "Н".

## 2. Обработка недостоверных входных данных

### 2.1 Уставка

|                          | Свертка «ИЛИ»                      | Свертка «И»                        |
|--------------------------|------------------------------------|------------------------------------|
| Проверка «Оба» включена  | Недостоверный операнд пропускается | Уставка считается НЕДОСТОВЕРНОЙ;   |
| Проверка «Оба» отключена |                                    | Используется константа операнда 1; |

Если при свертке по "ИЛИ" вообще не было достоверных операндов:

- включена проверка "Оба":
  - уставка считается НЕДОСТОВЕРНОЙ;
- отключена проверка "Оба":
  - в качестве уставки используется константа операнда 1.

### 2.2 Регулируемая величина

- При включенной проверке "Оба" или "Люб" регулируемая величина считается НЕДОСТОВЕРНОЙ;
- При отключенной проверке (стоит "НПР") используется константа операнда 2.

### 2.3 Возмущение

|                                  | Свертка «ИЛИ»                      | Свертка «И»                         |
|----------------------------------|------------------------------------|-------------------------------------|
| Проверка «Оба» / «Люб» включена  | Недостоверный операнд пропускается | Возмущение считается НЕДОСТОВЕРНЫМ; |
| Проверка «Оба» / «Люб» отключена |                                    | Используется константа операнда 2;  |

## 3. Действия по результатам проверки

Регулятор работает стандартным образом только при условии **достоверности всех входных данных**.

Если по условиям проверки есть недостоверные данные, то:

- если флаг выполнимости операции установлен в значение "Д":
  - в результат производится запись константы результата со своим статусом;
- если флаг выполнимости операции установлен в значение "Н":
  - выход НЕ меняется.

### A.7.8 Формирование выхода в регуляторах

**Для регуляторов с выходом на ШИМ, а также Пто Т ПРБ:**

- Если размерность приемника данных алгоблока:
  - скаляр:
    - Рез=Outr
  - вектор, список или матрица из 2-х элементов:
    - Outr>0:
      - Рез(1)=Outr
      - Рез(2)=0

-  $Out_r < 0$ :

-  $Рез(1) = 0$

$Рез(2) = ABS(out_r)$

- вектор, список или матрица из 3-х элементов:

Аналог случая с 2-мя элементами с добавлением:

$Рез(3) = Integ \text{ (Пто Т ПРБ)}$

$Рез(4) = Out_r$

**Для регуляторов с выходом на пропорциональный исполнительный механизм, а также Пто Т ПР:**

- Если размерность приемника данных алгоблока:

- скаляр:

$Рез = Out_r$

- вектор, список или матрица из 2-х элементов:

$Рез(1) = Out_r$

$Рез(2) = Integ$

, где  $Out_r$  - рассчитанное выходное воздействие

$Integ$  - значение интеграла.

Если используется запись в условия результата, т.е. взведены флаги «Общее включение условий результата» и «Запись в условия результата», а также в качестве приемника задан числовой вектор, то интеграл ( $Integ$ ) будет сохраняться именно там, а не во 2-м элементе результата.

Для регуляторов с выходом на пропорциональный исполнительный механизм по возможности производится тиражирование результата

#### A.7.9 Идентификация объекта управления

В регуляторах возможно включение алгоритма идентификации объекта управления с целью определения параметров его передаточной функции. Рассчитанные таким образом параметры в дальнейшем можно использовать для настройки регулятора.

Для включения алгоритма идентификации используются 3-й, 4-й и 5-й элементы условия операнда 1:

- если 3-й элемент равен **Истина** - включается алгоритм идентификации в **диагностическом режиме**:

- в момент установки в значение **Истина** запоминаются начальные значения управления и выхода объекта, которые в дальнейшем используются как смещения соответствующих параметров;

- в свободное место результата (после результата и интеграла для пропорционального или после результата в виде 3-х чисел для регулятора с ШИМ-выходом) записывается:

1.1 Для регулятора с одной постоянной времени записывается вектор из 2-х чисел:

АК - коэффициент усиления;

АКТ1 - 1-я постоянная времени;

1.2 Для регулятора с двумя постоянными времени записывается вектор из 3-х чисел:

АК - коэффициент усиления;

АКТ1 - 1-я постоянная времени;

АКТ2 - 2-я постоянная времени;

- если 3-й и 4-й элемент условия операнда 1 установлены в значение **Истина**, то идентификатор включается в **рабочий режим**. Наряду с записью вышеуказанных данных в результат, производится коррекция параметров регулятора: коэффициента усиления и постоянных времени следующим образом:

Вариант

1.1 - АК / АК1

1.2 - АК / АК1/ АК2

Модификация производится даже, если в результате не было места для записи рассчитанных показателей в диагностическом режиме.

- если 5-й элемент условия операнда 1 установлены в значение **Истина**, то идентификатор переключается в **режим паузы**. При этом производится только накопление данных – расчет коэффициентов и коррекция не производится.

#### **A.7.10 Изменение такта регулирования**

Условие операнда 1 (2-й элемент) является флагом расчета такта регулирования по постоянной времени объекта (кроме регуляторов ПТо Т П и ПТо Т И):

= Ложь - отключен пересчет;

= Истина - включен пересчет.



## А.8 Резервирование

### А.8.1 Резервирование

Резервирование - выбор по критерию из операнда числа и запись в результат.

#### Параметр 1

= 0 - операнд 2 используется совместно с операндом 1 в качестве источника выбора. Обработка производится по минимальным размерностям операндов и результата с возможным тиражированием в результате. Последовательность выбора элемента из группы [достоверных элементов](#):

использовать предыдущий достоверный операнд.

Если в векторах результата остается свободное место, то оно используется для хранения индексов операндов (=0 – значение выбиралось из операнда 1, =1 - значение выбиралось из операнда 2). Эти значение могут быть изменены другими алгоблоками и, следовательно, изменен порядок выбора.

= 1 - операнд 2 используется совместно с операндом 1 в качестве источника выбора. Обработка производится по минимальным размерностям операндов и результата с возможным тиражированием в результате. Последовательность выбора элемента из группы [достоверных элементов](#):

- поиск элемента, наиболее близкого к результату;
- поиск элемента из числа равноудаленных к результату, имеющего индекс, наиболее близкий к индексу, определенному в Параметре 2.

= 2 - операнд 2 используется совместно с операндом 1 в качестве источника выбора. Обработка производится по минимальным размерностям операндов и результата с возможным тиражированием в результате. Последовательность выбора элемента из группы [достоверных элементов](#):

- поиск элемента, имеющего индекс, наиболее близкий к индексу, определенному в Параметре 2.

= 10 - операнд 2 используется совместно с операндом 1 в качестве источника выбора;

последовательность выбора элемента из группы [достоверных элементов](#):

- мажорирование по значению;
- поиск элемента, наиболее близкого к результату (если при мажорировании выбрано более одного элемента);
- поиск элемента из числа равноудаленных к результату, имеющего индекс, наиболее близкий к индексу, определенному в Параметре 2.

= 11 - операнд 2 используется совместно с операндом 1 в качестве источника выбора;

последовательность выбора элемента из группы [достоверных элементов](#):

- поиск элемента, наиболее близкого к результату;
- поиск элемента из числа равноудаленных к результату, имеющего индекс, наиболее близкий к индексу, определенному в Параметре 2.

= 12 - операнд 2 используется совместно с операндом 1 в качестве источника выбора;

последовательность выбора элемента из группы [достоверных элементов](#):

- поиск элемента, имеющего индекс, наиболее близкий к индексу, определенному в Параметре 2.

= 20 - использование операнда 2 для хранения индексов выбранных элементов; последовательность выбора элемента из группы [достоверных элементов](#):

- поиск в группе элемента, имеющего индекс, равный индексу в операнде 2;
- мажорирование по значению;

- поиск элемента, наиболее близкого к результату (если при мажорировании выбрано более одного элемента);
  - поиск элемента из числа равноудаленных к результату, имеющего индекс, наиболее близкий к индексу, определенному в Параметре 2.
- = 21 - использование операнда 2 для хранения индексов выбранных элементов; последовательность выбора элемента из группы [достоверных элементов](#):
- мажорирование по значению;
  - поиск элемента, наиболее близкого к результату (если при мажорировании выбрано более одного элемента);
  - поиск элемента из числа равноудаленных к результату, имеющего индекс, наиболее близкий к индексу, определенному в Параметре 2.
- = 22 - использование операнда 2 для хранения индексов выбранных элементов; последовательность выбора элемента из группы [достоверных элементов](#):
- поиск элемента, наиболее близкого к результату;
  - поиск элемента из числа равноудаленных к результату, имеющего индекс, наиболее близкий к индексу, определенному в Параметре 2.

### **Параметр 2**

= N - номер желаемого индекса выбора (используется при выборке). Если в источнике нет желаемого индекса, используется ближайший к нему индекс. Индексом выборки называется индекс по источнику; для скаляра индекс всегда = 1; при использовании операнда 2 совместно с операндом 1 используется сквозная индексация.

**Параметр 3** - Используется в случае, когда все элементы источника выборки недостоверны.

= 0 - записать в результат константу результата, если запись константы разрешена;

= 1 - записать в результат значение по сохраненному индексу в операнде 2 или из параметра 2 (в параметре 2 выбранный индекс не сохраняется).

Выбор производится **только из достоверных элементов**.

Обычное включение проверки на достоверность по статусу производится включением поля "**Н**" (норма) в области условия операнда (в этом случае достоверными считаются данные со статусами 0, 3 или 4).

Для более жесткой проверки требуется

включить поле "**Стрелка вверх**" (это исключает статус 3) и/или поле "**Стрелка вниз**" (это исключает статус 4). Т.о. при включенных обеих стрелках достоверными будут данные только со статусом 0.

Проверка на достоверность по массиву позволяет формировать состояние данных по заранее сформированному массиву (при этом условие операнда должно использоваться именно как условие, а не как расширение операнда).

Если никакая из проверок не включена, то все входные данные будут считаться достоверными.

Если же, наоборот, в результате проверок состояния не оказывается достоверных данных, то:

- в результат записывается 0 и устанавливается статус недостоверности (если разрешена запись статуса);
- в логический массив результата (если он не используется как расширение результата) пишется **Ложь**.

### **Определение основного операнда (Параметр 1=10..12)**

Основным является операнд, имеющий более высокую размерность:

|           |           |        |        |         |
|-----------|-----------|--------|--------|---------|
|           | Операнд 2 | Скаляр | Вектор | Матрица |
| Операнд 1 |           |        |        |         |
| Скаляр    |           | ОП1    | ОП2    | ОП2     |
| Вектор    |           | ОП1    | ОП1    | ОП2     |
| Матрица   |           | ОП1    | ОП1    | ОП1     |

#### Образование источника с дополнением (Параметр 1=10..12)

|                 |              |           |                |                                                                                                 |
|-----------------|--------------|-----------|----------------|-------------------------------------------------------------------------------------------------|
|                 | Осн. операнд | Скаляр    | Вектор (No)    | Матрица (No,Mo)                                                                                 |
| Доп. операнд    |              |           |                |                                                                                                 |
| Скаляр          |              | Вектор(2) | Вектор (No+1)  | Матрица (No+1,Mo)                                                                               |
| Вектор (Nd)     |              | -         | Вектор (No+Nd) | Матрица (No+1,K),<br>K=min(Mo,Nd); если<br>K<Mo, то остальная<br>часть матрицы<br>(No,K+1..Mo)  |
| Матрица (Nd,Md) |              | -         | -              | Матрица (No+Nd,K),<br>K=min(Md,Mo); если<br>K<Mo, то остальная<br>часть матрицы<br>(No,K+1..Mo) |

#### Структурирование по размерностям

##### 1. Параметр 1 = 0..2

Операции производятся по минимальным размерностям операндов и результата; выбор всегда производится между двумя элементами: из операнда 1 и операнда 2. Если размерность результата больше размерности вычислений, производится тиражирование данных.

##### 2. Параметр 1 = 10..22

###### - Вариант 1:

Параметр 1 = 20..22:

Операнд 1 - матрица, операнд 2 и результат - вектора или матрицы.

Производится выбор N элементов, где N - минимум 2-й размерности операнда 1 и 1-х размерностей операнда 2 и результата.

Параметр 1 = 10..12:

Операнд 1+операнд 2 - матрица, результат - вектор или матрица.

Производится выбор N элементов, где N - минимум 2-й размерности операнда 1+операнда 2 и 1-й размерности результата.

Тиражирование производится, если результатом является матрица (тиражирование векторов).

###### - Вариант 2: Все остальные случаи размерностей.

Производится выбор одного элемента.

Операнд 1 (или операнд 1+операнд 2) рассматривается как вектор, в результате (или результате и операнде 2) используется только 1-й элемент; для результата производится тиражирование.

### А.8.2 Исправность

Данный алгоблок предназначен для диагностики исправности оборудования.

**Операнд 1 и Операнд 2** - проверяемые каналы (всегда используется чтение статуса для определения исправности)

**Результат** - результат проверки статусов операндов.

Канал считается неисправным, если в его статусе установлены биты 2 и 3 (нумерация с 1) (XXXXX11X) (X-значение бита - любое) или взведен бит 2 и любой из битов, указанных в Параметре 1 (DDDD01X) (D-биты, которые могут проверяться).

В случае исправности в соответствующий элемент результата пишется значение **Истина**, иначе **Ложь**. Одновременно формируется сводный сигнал исправности, являющийся объединением индивидуальных сигналов по **И**.

**Параметр 1** - определяет дополнительные коды недостоверных статусов:

Указываются биты (от 4-го до 8-го), которые будут проверены на установку в "1". Номера битов указываются в виде цифр числа (например, 58 определяет проверку двух битов - 5-го и 8-го, т.о. недостоверными будут объявлены статусы 1XXXX01X, XXX1X01X).

**Параметр 2** - данный параметр определяет дополнительный выходной сигнал:

номер определяется значением параметра, БД – указанной БД.

**Параметр 3** - определяет значение, выдаваемое в выходной сигнал, определенный параметром 2.

- Если параметром 2 определен дополнительный выходной сигнал, то:

- если сводный сигнал имеет значение **Истина**, то:

- если значение параметра 3 больше нуля, оно выдается в дополнительный сигнал (предполагается, что это длина импульса, формирующего меандр готовности);

- если значение параметра 3 меньше или равно нулю, то в дополнительном сигнале формируется меандр;

- если сводный сигнал равен **Ложь**, то на выход пишется **0**.

- Сам сводный сигнал записывается в результат в том случае, если размерность результата больше суммы размерностей операндов 1 и 2;

- Если суммарная размерность операндов 1 и 2 меньше размерности результата на 2 и более (т.е. есть место не только для сводного сигнала, то оставшееся место является управляющим); определение управляющего места:

- если результат вектор:

1-й индекс = 1-й размерности результата

2-й индекс = 1

- если результат матрица:

- есть возможность записи по 2-й размерности, то

1-й индекс = 1-й размерности результата

- 2-й индекс = 2-й размерности результата
- есть возможность записи по 1-й размерности, то
- 1-й индекс = 1-й размерности результата
- 2-й индекс = 1

Если значение управляющего сигнала равно **Ложь**, то сводный сигнал формируется только на основании данных 2-го операнда и значение сводного сигнала меняется с **Ложь/Истина** на **0/2** соответственно.

Для данной операции:

- проверки по статусам и логическим массивам отключены;
- разрешены **выборки данных**;
- разрешена **проверка исполнения по условиям результата**.

### Структурирование по размерностям

#### 1. Результат - скаляр или вектор В3(N3)

- Операнд 1 - (если матрица, сворачивается по "И" по 2-й размерности) поэлементно анализируется и записывается в соответствующие элементы результата;
- Операнд 2 аналогично операнду 1 записывается далее в результат (если есть место);
- Если осталось место, сводный сигнал записывается во все свободные элементы результата.

#### 2. Результат - матрица

- Поэлементный анализ и запись сначала операнда 1, затем, если есть место, операнда 2.

### А.8.3 Управление БПР

Данный алгоблок предназначен для управления БПР на основе анализа состояния каналов. Каналы анализируются попарно (соответствующие элементы операндов 1 и 2). Канал считается неисправным, если статус элемента равен значению 2 или любому из дополнительных значений статусов, определенных в параметре 1. В остальных случаях канал считается исправным. Для каждой пары анализируемых каналов в результате должна быть определена соответствующая пара каналов управления: 1-й элемент из пары управляет подключением канала из операнда 1, 2-й элемент управляет подключением канала из операнда 2. Если в результате анализа состояния каналов:

- оба канала исправны или оба неисправны, то никаких действий не производится;
- если один канал исправен, а другой неисправен, то производится подача команды на переключение БПР:
- для исправного канала на выход подается значение **"1"**;
- для неисправного канала на выход подается значение **"0"**.

Возможно указание проверки операндов по векторам. Если проверка не пройдена, канал также считается неисправным.

Для результата возможно указание проверки исполнения по условиям результата (индивидуально для каждого выхода).

Для операндов возможно указание выборки данных, при этом, если хотя бы один из операндов не прошел выборку, анализ состояния не производится и выход соответственно не меняется.

**Параметр 1** - определяет дополнительные статусы для проверки (допустимы значения 5, 6, 7). Статусы указываются в виде цифр числа, например, 65 (или 56) определяет еще 2 дополнительных статуса 5 и 6.

**Операнды 1 и 2** - данные от проверяемых каналов;

**Результат** - управление переключением БПР.

#### Структурирование размерностей

Операнды и результат рассматриваются как вектора.

Общее количество обрабатываемых пар каналов определяется

минимальными размерностями операндов и результата:

$$\text{Min}((R1OP1 * R2OP1), (R1OP2 * R2OP2), (R1REZ * R1REZ) / 2),$$

где

$R1(2)OP1(2)$  - 1-я(2) размерности операнда 1(2)

$R1(2)REZ$  - 1-я (2) размерности результата.

#### А.8.4 Диагностика

Данный алгоблок предназначен для контроля исправности оборудования. Контроль основывается на фиксации изменений входного сигнала за определенный интервал времени. Предполагается, что на вход алгоблока подается периодически изменяющийся сигнал, диагностирующий исправность оборудования. При отсутствии изменений входного сигнала диагностируется отказ оборудования.

**Параметр 1** - режим работы алгоблока:

= 0 - контроль исправности;

# 0 - контроль исправности с дополнительной диагностикой;

**Параметр 2** - начальное значение результата (записывается на старте и определяет состояние оборудования до истечения первого интервала исправности):

= 0 - в результат пишется значение "Истина" (оборудование исправно);

# 0 - в результат пишется значение "Ложь" (оборудование не исправно);

**Операнд 1** - входной периодически изменяющийся сигнал (меандр, счетчик и т.д.);

**Операнд 2** - заданный интервал контроля исправности, заданный в секундах;

Расширение операнда 2 - в режиме исправности (параметр 1 = 0) - интервал исправности;

- в режиме диагностики (параметр 1 # 0) - интервал диагностики;

**Результат** - результат контроля: "Истина" - если были изменения входного сигнала, "Ложь" - если не было (отказ);

Расширение результата - в режиме исправности (параметр 1 = 0) - результат анализа исправности ("Истина"/"Ложь");

- в режиме диагностики (параметр 1 # 0) - количество интервалов диагностики без изменений входного сигнала.

Возможно более точное определение состояния оборудования – режим диагностики (Параметр 1 # 0).

В этом режиме в расширении операнда 2 указывается интервал диагностики.

В расширение результата записывается количество интервалов диагностики, в течение которых не было зафиксировано изменений входного сигнала. Сброс этого значения производится по истечении соответствующего интервала исправности.

**Размерности вычислений.**

Параметр 1 = 0: размерности контроля исправности определяются минимальными размерностями операнда 1 и результата (включая расширения):

$$P_i = \min(OP1, PE3(\text{с расш}));$$

Параметр 1 # 0: размерности контроля исправности определяются минимальными размерностями операнда 1 и результата (без расширения):

$$P_i = \min(OP1, PE3(\text{без расш}));$$

размерности контроля диагностики определяются:

$$P_d = \min(P_i, \text{РасшРез}).$$

## А.9 Теплофизика

### А.9.1 Энтальпия

Данный алгоблок производит расчет энтальпии.

**Операнд 1** - температура: 0 - 180 °С;

**Операнд 2** - давление: 0 - 2,0 ата;

**Результат** - энтальпия: [ккал/кг]

**1-й параметр** алгоблока:

= 0 - расчет энтальпии воды;

= 1 - расчет энтальпии пара;

**2-й параметр** алгоблока:

= 0 – расчет энтальпии по давлению и температуре;

= 1 - расчет температуры насыщения и энтальпии пара по давлению;

= 2 - расчет давления насыщения и энтальпии пара по температуре;

**3-й параметр** алгоблока:

- значение, добавляемое к давлению.

Рассчитанные значения давления и температуры (режим: параметр 2 равен 1 или 2) записываются в соответствующий операнд.

Операция производится по минимальным размерностям операндов и результата. При необходимости операнды сворачиваются усреднением, начиная со старших размерностей.

#### Использование флага свертки операндов в алгоблоке

Для всех размерностей возможно изменение направления использования Оп1 и Оп2.

### А.9.2 Плотность

Данный алгоблок производит расчет плотности воды или пара.

**Операнд 1** - температура;

**Операнд 2** - давление;

**Результат** - плотность;

**1-й параметр** алгоблока:

= 0 - расчет плотности воды;

= 1 - расчет плотности пара;

**2-й параметр** алгоблока:

= 0 – расчет плотности по давлению и температуре;

= 1 – расчет плотности и температуры по давлению на линии насыщения;

= 2 – расчет плотности и давления по температуре на линии насыщения;

Рассчитанные значения давления и температуры (режим: параметр 2 равен 1 или 2) записываются в соответствующий операнд.



Операция производится по минимальным размерностям операндов и результата. При необходимости операнды сворачиваются усреднением, начиная со старших размерностей.

Для всех размерностей возможно изменение направления использования ОП1 и ОП2.

## А.10 Специализированные алгоблоки

### А.10.1 Расчет угла поворота сельсинной пары

Данный алгоблок предназначен для расчета угла поворота сельсинной пары по 2-м напряжениям.

**Операнд 1** - напряжение  $U_1$ ;

**Операнд 2** - напряжение  $U_2$ ;

Расширение ОП1 (если задано) - фаза напряжения  $U_1$  относительно напряжения питания (0-360град).

Если значение от  $0 \leq \text{фаза} < 180$ :  $U_1 > 0$ ;

Если значение от  $180 \leq \text{фаза} < 360$ :  $U_1 < 0$

Если расширение не задано, то считается, что  $U_1$  уже имеет знак.

Расширение ОП2 (если задано) - фаза напряжения  $U_2$  относительно напряжения питания (0-360град).

Если значение от  $0 \leq \text{фаза} < 180$ :  $U_2 > 0$ ;

Если значение от  $180 \leq \text{фаза} < 360$ :  $U_2 < 0$ ;

Если расширение не задано, то считается, что  $U_2$  уже имеет знак.

**Результат** - рассчитанный гол.

**Параметр 1** - тип результата:

= 0 - результат в градусах

# 0 - в радианах

#### Структурирование размерностей

Все операции выполняются по минимальным размерностям операндов и результата (и расширений операндов, если они заданы).

## A.11 Учет и ТЭП

### A.11.1 Приведение накопления

Данный алгоблок предназначен для приведения "пилообразноменяющегося" входного сигнала к формату минутных или часовых трендов.

**Операнд 1** - входное значение;  
**Операнд 2** - буфер сохранения;  
**Результат** - приведенный сигнал;

**Параметр 1** - режим работы:  
= 0 - приведение к часовому тренду (такт - 30 мин)  
1 - приведение к минутному тренду (такт - 15 сек)

Структурирование размерностей

Все операции выполняются по минимальным размерностям операндов и результата.

Алгоблок использует возможности слота Slot\_Umitrends.dll для определения моментов записи в тренды.

При ошибке стыковки с Slot\_Umitrends или получения адреса необходимой функции код ошибки (-102) запоминается в 1-м элементе условия результата.

## A.12 Циклы

### A.12.1 Цикл обхода (алгоблок цикла)

Данный алгоблок служит для организации управления вычислительным процессом. Алгоблок осуществляет передачу управления на предыдущий цикл обхода и, следовательно, может использоваться только на циклах обхода с номером, большим 1. Алгоритм подробно описан в разделе основные положения.

Алгоблок цикла, как и всякий другой имеет возможность условного исполнения в зависимости от значений указанных переменных, что позволяет использовать его более гибко.