



ТАС Мента версия 3

Руководство

Предисловие

Перед Вами руководство для ТАС Menta версия 3.

Если в данном руководстве Вы обнаружите ошибки и/или неточные описания, пожалуйста, свяжитесь с ближайшим к Вам представительством ТАС. Вы также можете связаться с нами по e-mail:

info@tac-russia.ru

Копирайт © 2000 ТАС АВ. Все права защищены.

Изготовитель сохраняет за собой право вносить незначительные изменения.

Данный документ, а также продукция, которая в нем упоминается, предназначены исключительно для пользователей, имеющих соответствующую лицензию. ТАС АВ обладает всеми правами на данный документ и сохраняет за собой право вносить необходимые дополнения и изменения. ТАС АВ не несет никакой ответственности за возможные ошибки в данном документе.

Продукция, описанная в документе, должна использоваться только по своему прямому назначению.

Только пользователи, имеющие соответствующую лицензию, имеют право использовать информацию из данного документа. Распространение, разглашение, копирование, хранение или использование нелегализованными пользователями продукции, информации или иллюстраций из данного документа в электронной или иной форме, например, в виде записей на пленку, ксерокопий и т.п., без письменного разрешения компании ТАС АВ считается нарушением авторского права и преследуется по закону.

ТАС Vista, ТАС Menta и ТАС Xenta – зарегистрированные товарные знаки, принадлежащие компании ТАС АВ в Швеции и других странах. Все другие упомянутые наименования продукции являются официальной собственностью своих владельцев.

Изменения

| Номер док. | Комментарии | Редактор | Дата |
|--------------------|---|-----------------|-------------|
| 0-004-7609-3 (GB) | Изменения для ТАС Menta версия 3.4. | СРНА | 2000-05-15 |
| 0-004-7484-0 (Рус) | Были внесены следующие изменения: <ul style="list-style-type: none">- Глава 2.4.1: Блоки расширения I/O с текущими входами, например, с конфигурацией 4-20 mA.- Глава 2.7.1: Начальное значение сигнала для аналогового выхода в блоке I/O задается как V.- Глава 2.19: Блок ERR (Системные ошибки) показывает, что выход блока I/O управляется в ручном режиме. Блок ERR необходимо использовать только в программах приложения.- Глава 6: Добавлен пример задержки старта.- Приложение 1: Расширен список доступных SNVT (№№ 48, 66, 92, 93, 94, 136, 137). Единицы измерения, диапазон и разрешения добавлены в таблицу. | | |

ТАС Мента версия 3

Руководство

Содержание

| | |
|--|-----------|
| ТАС Мента версия 3 | 1 |
| Руководство | 1 |
| Изменения | 2 |
| 1. Введение | 7 |
| 1.1. Общие сведения | 7 |
| 1.2. Терминология | 8 |
| 1.3. Типы сигналов | 8 |
| 2. Функциональные блоки | 9 |
| 2.1. Обзор | 9 |
| 2.1.1. Блоки I/O (Входов/Выходов) | 9 |
| 2.1.2. Источники сигнала | 9 |
| 2.1.3. Логические функции | 10 |
| 2.1.4. Нелинейные функции | 10 |
| 2.1.5. Блоки задержки | 10 |
| 2.1.6. Регуляторы и фильтры | 10 |
| 2.1.7. Сумматор | 11 |
| 2.1.8. Системные параметры | 11 |
| 2.1.9. Расписание и сигнал тревоги | 11 |
| 2.1.10. Передаточные функции | 11 |
| 2.2. ACCUM - Сумматор ACCUM | 12 |
| 2.3. ANYST - Аналоговый гистерезис ANYST | 13 |
| 2.4. AI - Аналоговый вход AI | 14 |
| 2.4.1. Линейный аналоговый вход | 14 |
| 2.4.2. Термисторный вход (Нелинейный аналоговый вход) | 15 |
| 2.4.3. Сетевая переменная | 16 |
| 2.4.4. SNVT | 18 |
| 2.4.5. Постоянное значение | 19 |
| 2.5. ALARM - Авария ALARM | 20 |
| 2.6. AND - Логическая «И»-функция AND | 22 |
| 2.7. AO - Аналоговый выход AO | 23 |
| 2.7.1. Физический выход | 23 |
| 2.7.2. SNVT | 24 |
| 2.7.3. Не подсоединен | 25 |
| 2.8. CNT - Цифровой вход –счетчик импульсов CNT | 26 |
| 2.8.1. Счетчик импульсов | 26 |
| 2.8.2. Не подсоединен | 27 |

| | | |
|---|---------------------|-----------|
| 2.9. CURVE – Линейный график | CURVE | 28 |
| 2.10. DATE - Дата | DATE | 30 |
| 2.11. DELAY - Задержка Включения/Выключения | DELAY | 31 |
| 2.12. DELB -Задержка на 1 цикл (Бинарный сигнал) | DELB | 33 |
| 2.13. DELI - Задержка на 1 цикл (Целый сигнал) | DELI | 33 |
| 2.14. DELR - Задержка на 1 цикл (Дробный сигнал) | DELR | 34 |
| 2.15. DI - Цифровой вход | DI | 35 |
| 2.15.1. Физический цифровой вход | | 35 |
| 2.15.2. Сетевая переменная..... | | 36 |
| 2.15.3. SNVT..... | | 37 |
| 2.15.4. Устройство HA-Линии..... | | 38 |
| 2.15.5. Постоянное значение | | 38 |
| 2.16. DO - Цифровой выход | DO | 39 |
| 2.16.1. Физический цифровой выход..... | | 39 |
| 2.16.2. SNVT..... | | 40 |
| 2.16.3. Не подсоединен | | 41 |
| 2.17. DOPU -Цифровой импульсный выход | DOPU | 42 |
| 2.17.1. Цифровой импульсный выход..... | | 42 |
| 2.17.2. Не подсоединен | | 43 |
| 2.18. ENTH - Расчет энтальпии | ENTH | 44 |
| 2.19. ERR - Системные ошибки | ERR | 46 |
| 2.20. FILT - Фильтры первого порядка | FILT | 47 |
| 2.21. HOUR - Текущий час | HOUR | 47 |
| 2.22. HYST - Бинарный гистерезис | HYST | 48 |
| 2.23. INTEG - Интегратор (Целые сигналы) | INTEG | 50 |
| 2.24. LIMIT – Ограничение по max/min | LIMIT | 51 |
| 2.25. MAX – Max показатель для 2 сигналов | MAX | 51 |
| 2.26. MIN – Min показатель для 2 сигналов | MIN | 52 |
| 2.27. MINUTE – Текущая минута | MINUTE | 52 |
| 2.28. MONTH – Текущий месяц | MONTH | 53 |
| 2.29. NCYC – Счетчик цикла программы | NCYC | 53 |
| 2.30. NOT – Инверсия бинарного сигнала | NOT | 54 |
| 2.31. OPT - Оптимизатор | OPT | 55 |
| 2.32. OR – «ИЛИ» - функция на 2 бинарных сигнала | OR | 62 |
| 2.33. OSC – Осциллятор | OSC | 62 |
| 2.34. PIDA - PID регулятор – Аналог. выход | PIDA | 64 |
| 2.35. PIDI-PID регулятор - Увеличить/Уменьшить | PIDI | 68 |
| 2.36. POLY – Полиномы | POLY | 71 |
| 2.37. PRCNT – Процентное соотношение | PRCNT | 72 |
| 2.38. PULSE – Импульсный генератор | PULSE | 73 |
| 2.39. PVB – Бинарная переменная | PVB | 74 |

| | | |
|--|--------------|------------|
| 2.40. PVI – Целая переменная | PVI..... | 74 |
| 2.41. PVR – Дробная переменная | PVR | 76 |
| 2.42. RAMP – Ограничитель скорости изменения сигнала | RAMP..... | 77 |
| 2.43. RST – Рестарт | RST..... | 78 |
| 2.44. RT – Учет времени работы | RT..... | 78 |
| 2.45. SECOND – Текущая секунда | SECOND..... | 79 |
| 2.46. SEQ – Последовательное управление | SEQ..... | 80 |
| 2.47. SHB – Регистр сдвигающий вправо | SHB..... | 83 |
| 2.48. SHI - Регистр сдвигающий вправо | SHI..... | 84 |
| 2.49. SHR - Регистр сдвигающий вправо | SHR..... | 85 |
| 2.50. SR – PC Триггер | SR | 86 |
| 2.51. TCYC – Время цикла | TCYC | 87 |
| 2.52. TRIG - Триггер | TRIG | 87 |
| 2.53. TSCH – Расписание | TSCH | 88 |
| 2.54. VECTOR - Векторы | VECTOR | 91 |
| 2.55. WDAY – Текущий день недели | WDAY | 92 |
| 2.56. XOR – Инверсия или XOR-функция | XOR..... | 92 |
| 3. Формулы | | 93 |
| 4. Операторы | | 97 |
| 5. Тестовые блоки | | 101 |
| 5.1. Краткое описание | | 101 |
| 5.2. TPAI – Тестовый блок для Аналог. входа | TPAI | 102 |
| 5.3. TPAO - Тестовый блок для Аналог. выхода | TPAO..... | 102 |
| 5.4. TPDI - Тестовый блок для Цифр. входа | TPDI | 103 |
| 5.5. TPDO - Тестовый блок для Цифр. выхода | TPDO..... | 103 |
| 6. Рекомендации по программированию | | 107 |
| Приложение 1 Используемые SNVTs | | 110 |

Всего: 114 страниц

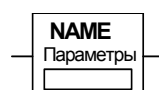
1. Введение

1.1. Общие сведения

В данном руководстве описываются все основные функции, доступные графическому языку программирования TAC Menta. Вопросы по использованию программного обеспечения TAC Menta освещены в *Руководстве пользователя TAC Menta*.

В *Главе 2* данного руководства содержится подробное описание всех типов *Функциональных блоков*, входящих в TAC Menta.

Описание каждого типа блока включает в себя следующую информацию:



| ВХОДЫ | Наименование | ТИП | описание |
|------------|--------------|-----|----------|
| ПАРАМЕТРЫ | Наименование | ТИП | описание |
| ТИП ВЫХОДА | ДОСТУП | | |
| ТИП | RO | | |

Описание: Описание...

- Входы** Тип входа и краткое описание его функций.
- Параметры** Параметры функциональных блоков (FB) и краткое описание их функций.
- Тип выхода** Тип данных для выхода FB. В случае соединения двух блоков друг с другом, тип выхода должен совпадать с типом входа.
- Доступ** Вид сигнала, который передается блоком. Сигнал может быть либо «только чтение» (RO), либо «чтение/запись» (R/W). Изменены могут быть только блоки с обозначением R/W (чтение/запись).
- Описание** Краткое функциональное описание блока.

Глава 3 содержит описание *Блока формул*.

Глава 4 содержит описание *Операторов*.

Глава 5 содержит описание *Тестовых блоков*.

1.2. Терминология

В тексте данного документа используются следующие сокращения:

FB Функциональный блок

FBD Схема функциональных блоков

Следующие буквы используются для обозначения контактов, к которым осуществляется подключение блоков I/O (входов/выходов):

V = термистор

U = универсальный вход

X = цифровой вход

Y = аналоговый вход

K = релейный выход

1.3. Типы сигналов

В FBD возможны соединения только между подобными сигналами. Существует три основных типа сигналов:

- Целые** 16 бит (диапазон: от -32 768 до 32 767)
- Дробные** 32 бита в IEEE формате с точностью до 7 знаков (диапазон: от $3,4 \cdot 10^{-38}$ до $3,4 \cdot 10^{38}$)
- Бинарные** 1 бит для двоичной информации (0/1 = ЛОЖЬ/ИСТИНА). Бинарный тип сигнала также называют *цифровым*.

В некоторых случаях может использоваться четвертый тип сигнала, а именно **Аналоговый сигнал**. Это значит, что, независимо от типа, как целые, так и дробные сигналы могут подсоединяться к необходимым входам/выходам.

2. Функциональные блоки

2.1. Обзор

2.1.1. Блоки I/O (Входов/Выходов)

| <i>Сокращение</i> | <i>Краткое описание</i> |
|-------------------|--|
| AI | Аналоговый вход |
| AO | Аналоговый выход. |
| CNT | Счетчик импульсов (количество импульсов за последний цикл программы) |
| DI | Цифровой вход. |
| DO | Цифровой выход. |
| DOPU | Цифровой импульсный выход |

Внешние сигналы входа/выхода могут соединяться с программой приложения через четыре *программных блока I/O* (AI, DI, AO и DO).

Программные блоки I/O могут соединяться либо с контактами, либо с сетевыми адресами. Подобное соединение называется *привязкой*.

Существует также два специальных типа блоков I/O. Первый предназначен для подсчета входящих цифровых импульсов (CNT), второй – для цифровых выходов с изменяющейся длительностью импульса (DOPU=ШИМ). Блоки CNT и DOPU нельзя подсоединять к внешним сетевым адресам, по причине возможности возникновения проблем, связанных с синхронизацией и временными характеристиками.

2.1.2. Источники сигнала

| <i>Сокращение</i> | <i>Краткое описание</i> |
|-------------------|-----------------------------------|
| NCYC | Счетчик цикла программы. |
| OSC | Генератор периодических импульсов |
| PVB | Бинарная переменная |
| PVI | Целая переменная |
| PVR | Дробная переменная |

2.1.3. Логические функции

| Сокращение | Краткое описание |
|------------|---|
| AND | «И»-функция на 2 бинарных сигнала |
| NOT | Инверсия бинарного сигнала |
| OR | «ИЛИ»-функция на 2 бинарных сигнала |
| PULSE | Моностабильный импульсный генератор с изменяемой длительностью импульса |
| SR | Триггер (РС-триггер) |
| TRIG | Переключатель, действующий при изменении входного сигнала |
| XOR | Инверсия или XOR-функция на 2 бинарных сигнала. |

2.1.4. Нелинейные функции

| Сокращение | Краткое описание |
|------------|--|
| ANYST | Аналоговый гистерезис |
| HYST | Бинарный гистерезис |
| LIMIT | Ограничение по max/min |
| MAX | Максимальный показатель для 2 сигналов |
| MIN | Минимальный показатель для 2 сигналов |

2.1.5. Блоки задержки

| Сокращение | Краткое описание |
|------------|---|
| DELAY | Задержка включение/ задержка выключение. |
| DELB | Задержка на 1 цикл (бинарный сигнал). |
| DELI | Задержка на 1 цикл (целый сигнал) |
| DELR | Задержка на 1 цикл (дробный сигнал) |
| SHB | Регистр, сдвигающий вправо; бинарный сигнал |
| SHI | Регистр, сдвигающий вправо; целый сигнал |
| SHR | Регистр, сдвигающий вправо; дробный сигнал |

Блок задержки имеет один вход того же типа, что и выход.

Блок «Регистр, сдвигающий вправо» имеет 2 входа: вход «данные»(и выход того же типа), а также «управляющий» вход. При поступлении сигнала ИСТИНА на «управляющий» вход, осуществляется запись информации со входа «данные». Когда управляющий сигнал ЛОЖЬ, выход сохраняет последнее значение, записанное при управляющем сигнале ИСТИНА.

2.1.6. Регуляторы и фильтры

| Сокращение | Краткое описание |
|------------|--|
| FILT | Фильтр первого порядка |
| OPT | Оптимизация времени запуска/ остановки |
| PIDA | PID-регулятор (Аналоговый выход) |
| PIDI | PID-регулятор (выход увеличить/ уменьшить) |
| RAMP | Ограничитель скорости изменения сигнала |
| SEQ | Блок последовательного управления |

2.1.7. Сумматор

| <i>Сокращение</i> | <i>Краткое описание</i> |
|-------------------|----------------------------|
| ACCUM | Сумматор дробных сигналов |
| INTEG | Интегратор (целые сигналы) |
| RT | Учет времени работы |

2.1.8. Системные параметры

| <i>Сокращение</i> | <i>Краткое описание</i> |
|-------------------|---|
| DATE | Текущая дата. |
| ERR | Системные ошибки |
| HOUR | Текущий час |
| MINUTE | Текущая минута |
| MONTH | Текущий месяц |
| RST | Рестарт. Выход, активизируется при первом цикле программы после теплого старта. |
| SECOND | Текущая секунда |
| TCYC | Номер цикла программы приложения |
| WDAY | Текущий день недели. 1 = понедельник |

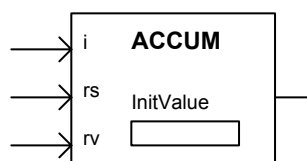
Данная группа состоит из функциональных блоков, выходы которых содержат внутренние параметры ТАС Хента. По этой причине все блоки данной группы относятся к так называемым источникам сигналов, поскольку у них нет входов.

2.1.9. Расписание и сигнал тревоги

| <i>Сокращение</i> | <i>Краткое описание</i> |
|-------------------|-------------------------------------|
| ALARM | Функция выдачи аварийного сообщения |
| TSCH | Расписание |

2.1.10. Передаточные функции

| <i>Сокращение</i> | <i>Краткое описание</i> |
|-------------------|-------------------------|
| CURVE | Линейный график |
| ENTH | Расчет энтальпии |
| POLY | Полиномы |
| PRCNT | Проценты |
| VECTOR | Векторы |

2.2. ACCUM - Сумматор**ACCUM**

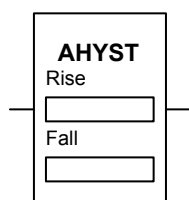
| | | | |
|----------------|--------------------|----------|--|
| ВХОДЫ | Приращение (i) | ДРОБНЫЙ | Величина, суммируемая для каждого цикла программы |
| | Сброс (rs) | БИНАРНЫЙ | Вход сброса (1 = сброс) |
| | Предустановка (rv) | ДРОБНЫЙ | Значение на выходе при активизированном входе сброса |
| ПАРАМЕТРЫ | Нач. значение | ДРОБНЫЙ | Начальное значение сумматора |
| ТИП ВЫХОДА | | ДОСТУП | |
| ДРОБНЫЙ СИГНАЛ | | R/W | |

Описание: Этот блок используется для суммирования общего приращения за время одного программного цикла. Рассчитывается сумма изменения *Приращения* за время цикла. Суммирование производится с большой цифровой точностью во избежание ошибок в случае малого приращения.

Следует обратить особое внимание на то, что суммарное значение появляется на выходе только по прохождении программного цикла.

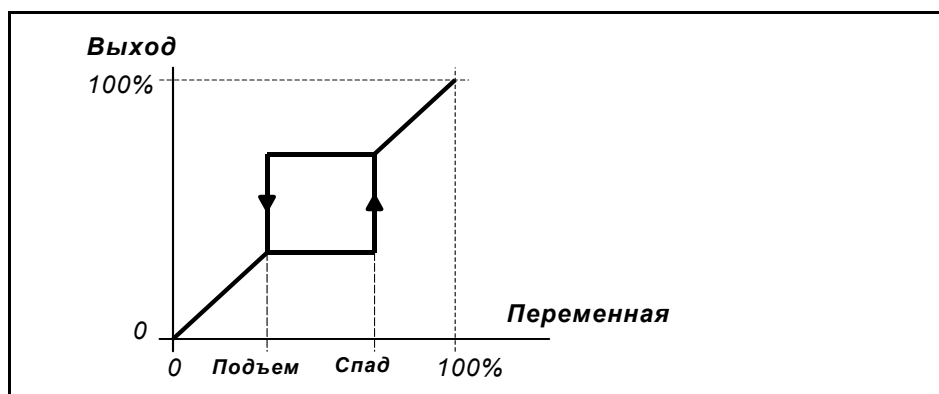
Значение на выходе сумматора будет соответствовать *Начальному значению* в его начальном состоянии. При активации входа *Сброс*, значение на выходе сумматора сбрасывается до величины, которая была подана на вход *Предустановка*. При деактивации входа *Сброс*, суммирование ведется от последнего значения на входе *Предустановка*.

Верхняя и нижняя границы значения на выходе определяются исходя из максимального значения дробного сигнала, установленного в контроллере.

2.3. АНУСТ - Аналоговый гистерезис**АНУСТ**

| | | | |
|----------------|------------|---------|--|
| ВХОДЫ | Переменная | ДРОБНЫЙ | Входящий сигнал |
| ПАРАМЕТРЫ | Подъем | ДРОБНЫЙ | Значение входящего сигнала при подъеме в области гистерезиса |
| | Спад | ДРОБНЫЙ | Значение входящего сигнала при спаде в области гистерезиса |
| ТИП ВЫХОДА | ДОСТУП | | |
| ДРОБНЫЙ СИГНАЛ | RO | | |

Описание: Данный блок выполняет функцию аналогового гистерезиса. Значение на выходе соответствует значению на входе до тех пор, пока значение на входе находится за пределами области гистерезиса. Когда переменная на входе входит в область гистерезиса, выход принимает значение, ограниченное гистерезисом, см. график:



Если $\text{Подъем} > \text{Спад}$, процесс гистерезиса будет направлен против часовой стрелки, как показано на графике. Если $\text{Подъем} < \text{Спад}$, процесс гистерезиса будет проходить в направлении по часовой стрелке.

При возникновении необходимости в комплексной функции с несколькими областями гистерезиса, рекомендуется использовать другой функциональный блок (POLY, CURVE, VECTOR, и т.п.) и каскадом подключить то число АНУСТ-блоков, которое будет соответствовать числу областей гистерезиса в желаемой функции передачи.

2.4. AI - Аналоговый вход

AI



AI-блок имеет 5 различных режимов, настраиваемых пользователем:

- Линейный аналоговый вход
- Термисторный вход (температурный)
- Сетевая переменная
- SNVT
- Постоянное значение (константа)

Значение выхода блока обновляется один раз за программный цикл. Это значит, что изменения входящего сигнала протяженностью менее одного программного цикла не будут учитываться программой приложения.

2.4.1. Линейный аналоговый вход

| | | | |
|----------------|--------------------|---------|---|
| ПАРАМЕТРЫ | Номер модуля | НОМЕР | Номер I/O-модуля |
| | Название контакта | СПИСОК | Название контакта |
| | Тип датчика | СПИСОК | Тип датчика |
| | Min значение | ДРОБНЫЙ | Нижняя граница диапазона входящего сигнала |
| | Max значение | ДРОБНЫЙ | Верхняя граница диапазона входящего сигнала |
| | Постоянная времени | ДРОБНЫЙ | Постоянная времени для фильтра (сек) |
| | Начальное значение | ДРОБНЫЙ | Начальное значение на выходе (только для I/O-модуля) Значение по умолчанию = 0 |
| ТИП ВЫХОДА | | ДОСТУП | |
| ДРОБНЫЙ СИГНАЛ | | RO | |

Описание: В режиме *Линейный аналоговый вход* блок получает значение с аналогового входа от линейного датчика. Значение на выходе блока представляет собой значение на аналоговом входе, преобразованное в соответствующие единицы измерения.

Параметр Номер модуля представляет собой номер I/O-модуля (0 = основное устройство). Параметр Название контакта задает тип и номер контакта (напр.: V1-V4, U1-U4).

Тип датчика для установленного I/O-модуля или основного устройства, выбирается из предложенного списка, напр.: 0-10 В

или 2-10 В. Преобразованные в соответствующие единицы измерения значения параметров Min значение и Max значение представляют собой нижнюю и верхнюю границы диапазона возможного значения на датчике, напр.: 0-40 °С, соответствующие наименьшему и наибольшему значению входящего электрического сигнала.

После преобразования, значения с датчика могут пропускаться через фильтр первого порядка, включенный в функциональный блок. Постоянная времени для фильтра выражается в секундах при помощи параметра Постоянная времени.

Алгоритм работы фильтра следующий

$$y(k) = y(k-1) + \frac{1}{1 + \frac{T}{h}} (u(k) - y(k-1))$$

где $y(k)$ и $u(k)$ представляют собой профильтрованное и, соответственно, непрофильтрованное значение в единицу времени k . h – это выбранный интервал времени (т.е. время цикла программы приложения), а T – это постоянная времени для фильтра. Если постоянная времени меньше или равна нулю, фильтрация не производится. Для получения функции фильтрации, значение постоянной времени должно быть выбрано больше величины продолжительности программного цикла программы приложения.

Данный алгоритм работы фильтра основан на приближенном значении *Обратное дифференцирование* оператора бесконечной производной времени. За более подробным объяснением обращайтесь, например, к изданию Острема и Виттенмарка «Системы, контролируемые компьютером – Теория и дизайн». (Åström and Wittenmark: "Computer Controlled Systems - Theory and design". Prentice-Hall, Englewood Cliffs (1984)).

При подключении входа к I/O модулю, начальное значение на выходе блока (до получения значения сигнала с I/O модуля) задается параметром *Начальное значение*. При отключении I/O модуля, выход блока сохранит последнее значение, полученное с I/O модуля.

2.4.2. Термисторный вход (Нелинейный аналоговый вход)

| | | | |
|----------------|--------------------|--------|---|
| ПАРАМЕТРЫ | Номер модуля | НОМЕР | Номер I/O-модуля |
| | Название контакта | | Название контакта |
| | Тип датчика | | Тип датчика |
| | Постоянная времени | | Постоянная времени для фильтра (сек) |
| | Начальное значение | | Начальное значение на выходе (только для I/O-модуля) Значение по умолчанию = 0 |
| ТИП ВЫХОДА | | ACCESS | |
| ДРОБНЫЙ СИГНАЛ | | RO | |

Описание: В режиме Термисторный вход (Нелинейный аналоговый вход) блок получает значение с аналогового входа от нелинейного датчика, т.е. термистора.

Значение на выходе блока представляет собой значение на аналоговом входе, преобразованное в соответствующую единицу измерения. Преобразование производится путем использования заранее известной кривой, определяющей характеристики датчика.

Параметр *Номер модуля* представляет собой номер I/O-модуля (0 = основное устройство). Параметр *Название контакта* задает тип и номер контакта (напр.: В1 - В4, U1 - U4).

Преобразование значения сигнала в единицу измерения задается параметром *Тип датчика*. Последний необходимо выбрать из предложенного списка. Выбор *SP adjust* производится при подсоединении сигнала потенциометра от комнатного термостата серии ZS 100.

После преобразования, значение сигнала может быть пропущено через фильтр первого порядка, входящий в функциональный блок. Постоянная времени для фильтра задается в секундах параметром *Постоянная времени*.

Алгоритм работы фильтра будет тот же, что и при конфигурации Линейный аналоговый вход.

При подключении входа к I/O модулю, начальное значение на выходе блока (до получения значения сигнала с I/O модуля) задается параметром *Начальное значение*. При отключении I/O модуля, выход блока сохранит последнее значение, полученное с I/O модуля.

2.4.3. Сетевая переменная

| | | | |
|----------------|--------------------|---------|---|
| ПАРАМЕТРЫ | Сетевой адрес | СТРОКА | Соотносится с внешним сигналом сети ТАС. |
| | Начальное значение | ДРОБНЫЙ | Начальное значение на выходе. Значение по умолчанию = 0 |
| | Дельта | ДРОБНЫЙ | Наименьшее изменение значения внешнего сигнала, которое положит начало обновлению полученного значения.. Значение по умолчанию = 0.5 |
| | Период | ЦЕЛЫЙ | Максимальный интервал времени (в секундах) между двумя обновлениями посланного значения. Значение по умолчанию =60 |
| ТИП ВЫХОДА | | ДОСТУП | |
| ДРОБНЫЙ СИГНАЛ | | RO | |

Описание: В режиме *Сетевая переменная* блок используется для получения сигнала от других блоков ТАС к программе приложения через сеть.

Сетевой адрес представляет собой “Public” сигнал в другом блоке TAC, представленный в виде строки, напр.: \RPU1\AHU1\OutdoorTemp (максимальное число знаков \20\12\20). Изменять сетевой адрес в процессе работы блока нельзя.

Начальное значение на выходе блока (до получения им какого-либо значения сигнала через сеть) задается параметром *Начальное значение*. В случае возникновения ошибки при коммуникации, значение на выходе блока не будет обновляться (т.е. выход всегда будет сохранять последнее записанное значение сигнала, принятого через сеть). После холодного старта и установки RAM-памяти контроллера в начальное положение (=0), значение на выходе блока сбросится до *Начального значения*.

Полученное значение обновляется каждый раз, когда значение отличается от последнего посланного значения больше, чем на величину *Дельта*. (Если *Дельта* имеет отрицательное значение, то данная функция *не работает*). Независимо от того, происходит изменение сигнала или нет, полученное значение будет обновляться также по истечении времени, заданного параметром *Период*, с момента последнего обновления. (Если значение *Периода* меньше 1 сек, задайте *Период = 1 сек*).

Обратите внимание на то, что использование одного сетевого адреса для нескольких AI-блоков, входящих в одну схему FBD, не имеет смысла, т.к. в сети возникнет ненужная нагрузка, если будет послано несколько запросов на получение одного входящего сигнала.

2.4.4. SNVT

| | | | |
|----------------|--------------------|----------|---|
| ПАРАМЕТРЫ | Тип | СПИСОК | Тип SNVT. |
| | Поле | СПИСОК | Тип SNVT –поля (Применяется только после выбора структурированного типа SNVT). |
| | SNVT - название | СПИСОК | ID сигнала в сети, максимум 16 знаков. |
| | Начальное значение | ДРОБНЫЙ | Начальное значение на выходе. Значение по умолчанию = 0 |
| | Опрос | БИНАРНЫЙ | Определяет, должно ли быть запрошено значение внешнего сигнала на выходе. |
| | Период | ЦЕЛЫЙ | Максимальный интервал времени (в секундах) между двумя обновлениями полученного значения. Значение по умолчанию =60 |
| ТИП ВЫХОДА | | ДОСТУП | |
| ДРОБНЫЙ СИГНАЛ | | RO | |

Описание: В режиме *SNVT* блок используется для передачи через сеть сигнала из внешнего оборудования в программу приложения. Блок входов должен также соединяться с выходящим сигналом внешнего оборудования (которое должно быть такого же *SNVT*-типа) при помощи программы сетевой адресации. Обратите внимание на то, что сигналу могут присваиваться разные имена в приложении (имя блока, напр. *OutdoorTemp*) и в сети (*имя SNVT name*, напр. *nvioutdoortemp*).

Значение параметров *Тип* и *Поле* выбираются из предоставленного списка для блоков *SNVT*- типа.

Начальное значение на выходе блока (до получения им какого-либо значения сигнала через сеть) задается параметром *Начальное значение*. После холодного старта и установки RAM-памяти контроллера в начальное положение (=0), значение на выходе блока сбросится до *Начального значения*.

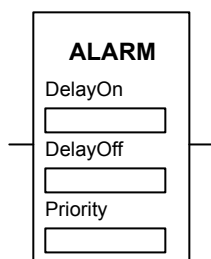
В случае возникновения ошибки при коммуникации, значение на выходе блока не будет обновляться (т.е. выход всегда будет сохранять последнее записанное значение сигнала, принятого через сеть).

После адресации связей, внешний сигнал на выходе будет опрашиваться автоматически, если выбрана функция *Опрос* и если не произошло обновления значений во время *Периода*.

2.4.5. Постоянное значение

| | | | |
|----------------|--------------------|---------|-------------------------------|
| ПАРАМЕТРЫ | Начальное значение | ДРОБНЫЙ | Начальное значение на выходе. |
| ТИП ВЫХОДА | ДОСТУП | | |
| ДРОБНЫЙ СИГНАЛ | RO | | |

Описание: Вход блока не имеет ни физического, ни сетевого адреса. Вместо этого данная конфигурация предоставляет на выход блока постоянное значение, заданное параметром *Начальное значение*.

2.5. ALARM - Авария**ALARM**

| ВХОДЫ | Вход | БИНАРНЫЙ | Сигнал на входе |
|-----------------|---------------------|----------|--|
| ПАРАМЕТРЫ | Задержка включения | ДРОБНЫЙ | Задержка перед включением аварийной функции |
| | Задержка выключения | ДРОБНЫЙ | Задержка перед выключением аварийной функции |
| | Приоритет | ЦЕЛЫЙ | Уровень приоритета аварии |
| | Текст аварии | СПИСОК | Текст аварии |
| ТИП ВЫХОДА | | ДОСТУП | |
| БИНАРНЫЙ СИГНАЛ | | RO | |

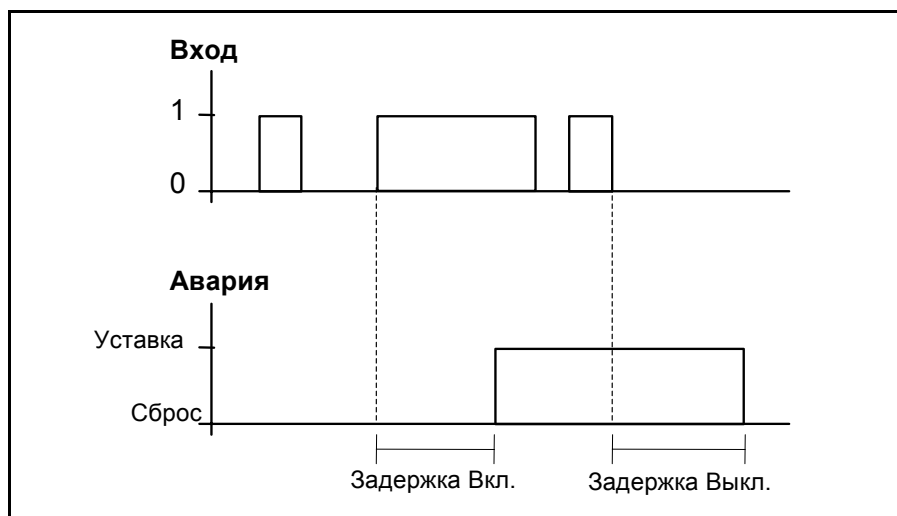
Описание: ALARM-блок отслеживает статус входящего бинарного сигнала. При задержке включения (переход от 0 до 1), начинается отсчет времени, на протяжении которого входящий сигнал будет иметь значение ИСТИНА (1). Если данное время превышает время *Задержки включения*, то аварийное сообщение, содержащее информацию о состоянии, времени, имени сигнала, приоритете, а также текст аварии, выдаются системной программой. После выдачи сообщения, ALARM-блок будет ожидать изменения значения входящего сигнала на ЛОЖЬ (0). При *Задержке выключения* входящего сигнала начинается отсчет времени подобно тому, как это происходило при задержке включения. По прохождении времени *Задержки выключения* меняется состояние аварийного сообщения.

Сигнал на выходе ALARM-блока показывает состояние текущего аварийного сообщения. 1 = активное состояние аварийного сообщения, 0 = пассивное состояние аварийного сообщения.

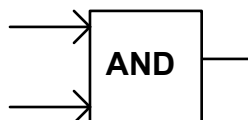
Время задержки включения и выключения задается в секундах параметрами *Задержка включения* и *Задержка выключения*. Приведенная ниже временная диаграмма показывает включение и выключение функции аварийного сообщения. Если в период задержки включения/ выключения случится сбой напряжения питания, недостающее время задержки будет сохранено только в том случае, если выбрана метка *Backup*.

При помощи кнопки **Edit** параметра *TAC Vista Alarm* можно ввести в TAC Vista специальные данные для управления аварийным сообщением (максимально 20 знаков) и текстом аварии (выдача или сброс аварийного сообщения, максимально 255 знаков для каждой функции).

Уровень приоритета аварии задается параметром *Приоритет*. Данное значение не производит никаких действий в программе приложения. Оно всего лишь является составной частью той информации, которая входит в выдаваемое аварийное сообщение. Аварийные сообщения хранятся в специальном аварийном списке, который обрабатывается системной программой. Обработку можно осуществлять, например, через панель оператора. Когда сигнал аварии посылается в ТАС Vista, параметр *Приоритет* используется для сортировки аварийных событий согласно их степени важности (10 уровней: от 0 до 9). Аварийные события с приоритетом = 0 (Информационное сообщение) обладают наименьшим приоритетом и *не попадают* в базу данных аварии в ТАС Vista, а также *не отображаются* в списке аварий.



Обратите внимание на то, что в течение интервала времени, превышающего время *Задержки включения*, входящий сигнал должен иметь значение ИСТИНА для того, чтобы вызвать новое аварийное сообщение. Значение выходящего сигнала в течение интервала времени, превышающего время *Задержки выключения*, должно быть ЛОЖЬ для того, чтобы отключить аварийное сообщение.

2.6. AND - Логическая «И»-функция**AND**

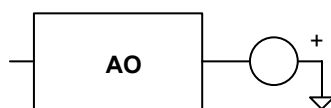
| | | | |
|-----------------|--------|----------|--|
| ВХОДЫ | Вход 1 | БИНАРНЫЙ | |
| | Вход 2 | БИНАРНЫЙ | |
| ТИП ВЫХОДА | | ДОСТУП | |
| БИНАРНЫЙ СИГНАЛ | | RO | |

Описание: AND-блок рассчитывает бинарную «И»-функцию для *Входа 1* и *Входа 2*, согласно нижеприведенной таблице истинности:

| вход 1 | вход 2 | выход |
|--------|--------|-------|
| 0 | 0 | 0 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 1 |

2.7. АО - Аналоговый выход

АО



АО-блок имеет 3 различных режима, настраиваемых пользователем:

- Физический выход
- SNVT
- Не подсоединен

2.7.1. Физический выход

| ВХОДЫ | Вход | ДРОБНЫЙ | Сигнал на входе, 0-100% |
|----------------------|--------------------|---------|---|
| ПАРАМЕТРЫ | Номер модуля | НОМЕР | Номер I/O – модуля. |
| | Название контакта | СПИСОК | Название контакта |
| | Напряжение 0% | ДРОБНЫЙ | Сигнал на выходе, если входящий сигнал 0% (Вольт) |
| | Напряжение 100% | ДРОБНЫЙ | Сигнал на выходе, если входящий сигнал 100% (Вольт) |
| | Начальное значение | ДРОБНЫЙ | Начальное значение на выходе (только для I/O-модуля) Значение по умолчанию = 0 (Вольт) |
| ТИП ВЫХОДА | | ДОСТУП | |
| Блок не имеет выхода | | RO | |

Описание: Блок «Аналоговый выход» преобразует аналоговый сигнал 0-100%, поступающий из программы приложения, в физический сигнал напряжения .

Параметр *Номер модуля* представляет собой номер I/O-модуля (0 = основное устройство). Параметр *Название контакта* задает тип и номер контакта на выходе (напр.: Y1-Y4).

Значение напряжения на выходе, соответствующее 0% и 100% входящего сигнала, задается (в Вольтах) параметрами *Напряжение 0%* и *Напряжение 100%*. Обратите внимание на то, что данные параметры могут приобретать обратное значение, напр.: *Напряжение 0%* = 10 V и *Напряжение 100%* = 2 V.

При соединении блока с I/O-модулем, параметр *Начальное значение* задает значение выходящего сигнала, если I/O-модуль отключен, напр., при повторном запуске блока в результате сбоя питания.

2.7.2. SNVT

| | | | |
|----------------------|-----------------|----------|---|
| ПАРАМЕТРЫ | Тип | СПИСОК | Тип SNVT. |
| | Поле | СПИСОК | Тип SNVT –поля (Применяется только после выбора структурированного типа SNVT). |
| | SNVT - название | СПИСОК | ID сигнала в сети, максимум 16 знаков. |
| | Нач. значение | ДРОБНЫЙ | Начальное значение на выходе. Значение по умолчанию = 0 |
| | Отправить | БИНАРНЫЙ | Определяет, должен ли внешний сигнал обновляться автоматически или нет |
| | Дельта | ДРОБНЫЙ | Наименьшее изменение значения входящего сигнала, который будет передаваться на вход подсоединенного устройства. Значение по умолчанию = 0,5 |
| | Период | ЦЕЛЫЙ | Максимальный интервал времени (в секундах) между двумя обновлениями посланного значения. Значение по умолчанию =60 |
| OUTPUT TYPE | | ДОСТУП | |
| Блок не имеет выхода | | RO | |

Описание: В режиме *SNVT* блок используется для передачи через сеть сигнала из программы приложения во внешнее оборудование. Блок выходов должен быть «привязан» к входящему сигналу внешнего оборудования (которое должно быть такого же *SNVT*-типа) при помощи программы сетевой привязки. Обратите внимание на то, что сигналу могут присваиваться разные имена в приложении (имя блока, напр., *OutdoorTemp*) и в сети (*SNVT – Имя*, напр., *nvooutdoortemp*).

Значение параметров *Тип* и *Поле* выбираются из предоставленного списка для блоков *SNVT*- типа.

Начальное значение на выходе блока (перед выполнением первого программного цикла) задается параметром *Начальное значение*.

Если выбрана функция *Отправить*, значение внешнего входящего сигнала будет каждый раз обновляться при отклонении значения сигнала на входе блока от последнего записанного значения на величину, превышающую *Delta*. Независимо от факта изменения значения входящего сигнала, внешний входящий сигнал будет обновляться по прохождении времени, заданного параметром *Период*, после последнего обновления. Если значение *Периода* = 0, внешний сигнал на входе будет обновляться только в случае его отклонения от величины *Delta*.

Если функция *Отправить* не выбрана, для получения обновленных значений внешнее оборудование должно использовать функцию *Опрос*. Полученное значение будет обновляться лишь в случае отклонения значение сигнала на входе

блока от последнего записанного значения на величину, превышающую *Delta*.

2.7.3. Не подсоединен

В режиме *Не подсоединен* блок не подсоединен ни к физическому, ни к сетевому адресу. Данный режим не имеет параметров.

2.8. CNT - Цифровой вход –счетчик импульсов CNT



CNT-блок, используя режимы I/O, может быть подключен либо в режиме счетчика импульсов, либо в режиме *Не подсоединен*.

2.8.1. Счетчик импульсов

| ПАРАМЕТРЫ | Номер модуля | НОМЕР | Номер I/O - модуля. |
|----------------|-------------------|----------|---|
| | Название контакта | СПИСОК | Название контакта. |
| | Нормально открыт | БИНАРНЫЙ | Метка выбрана = переключение по переднему фронту т.е. нормально открыт -> закрыт (По умолчанию) Метка не выбрана = переключение по заднему фронту, т.е. нормально закрыт -> открыт. |
| | Множитель | ДРОБНЫЙ | Фактор шкалы. Значение по умолчанию = 1.0 |
| ТИП ВЫХОДА | | ДОСТУП | |
| ДРОБНЫЙ СИГНАЛ | | RO | |

Описание: Данный блок подсчитывает количество импульсов, поступивших на цифровой вход в течение последнего программного цикла. Счетчик прекратит подсчет (не сбрасывая показания до нуля), если количество импульсов достигнет максимального Целого значения (32767) до начала следующего программного цикла. Минимальное измеряемое значение длины импульса зависит от используемого программного обеспечения.

Количество полученных импульсов переводится в соответствующие единицы измерения путем умножения значения со счетчика импульсов на фактор шкалы *Множитель*.

Параметр *Номер модуля* представляет собой номер I/O-модуля (0 = основное устройство). Параметр *Название контакта* задает тип и номер контакта на входе (напр.: U1-U4, X1-X4).

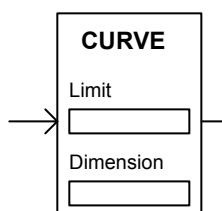
В зависимости от заданного значения параметра *Нормально открыт*, счетчик переключается либо по переднему, либо по заднему фронту входного цифрового сигнала:

Метка выбрана = переключение по переднему фронту (нормально открыт -> закрыт)

Метка не выбрана = переключение по заднему фронту (нормально закрыт -> открыт).

2.8.2. Не подсоединен

В режиме *Не подсоединен* блок не подсоединен ни к физическому, ни к сетевому адресу. Данный режим не имеет параметров.

2.9. CURVE – Линейный график**CURVE**

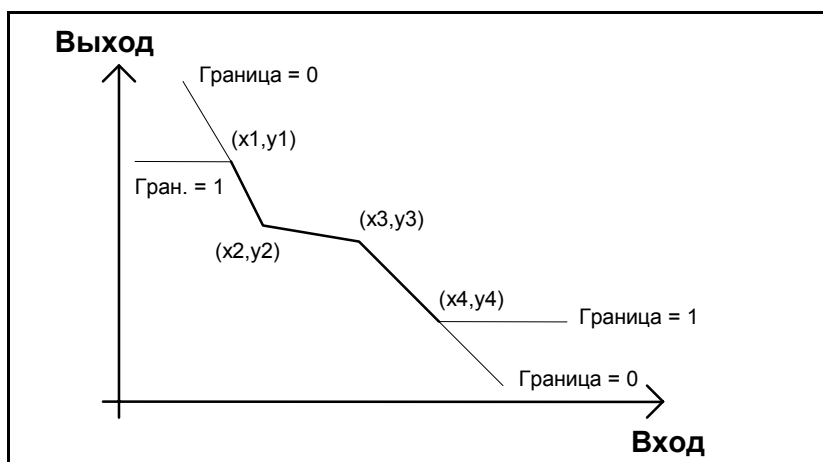
| ВХОДЫ | Вход | ДРОБНЫЙ | |
|----------------|----------------|----------|---|
| ПАРАМЕТРЫ | Граница | БИНАРНЫЙ | Выбор функции ограничения (граница = 1) или линейной экстраполяции (граница = 0). |
| | Координаты x,y | ДРОБНЫЙ | Список пар координат x,y, задающих место изгиба графика. |
| ТИП ВЫХОДА | | ДОСТУП | |
| ДРОБНЫЙ СИГНАЛ | | RO | |

Описание: Данный блок позволяет строить любой ломаный график, состоящий из линейных отрезков, заданных, исходя из количества точек изгиба графика (x_i, y_i), т.е. значение функции $y = f(x)$ для заданных значений x на входе. Общее количество точек изгиба не должно превышать 127. Число заданных точек изгиба (N) отображается в ячейке *Значение*, расположенной на схеме функционального блока. Точки изгиба вводятся списком Дробных значений x,y через запятую, по 2 точки координат на каждой строчке. Значения координат x вводятся в список по порядку, т.е. $x_{i-1} < x_i < x_{i+1}$.

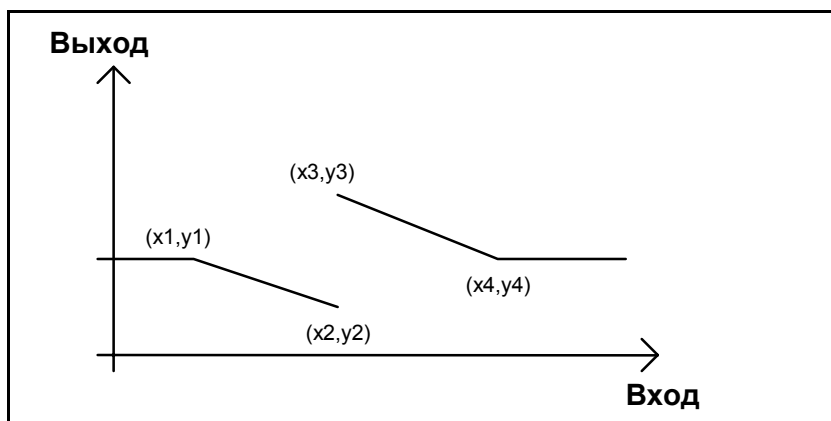
Значение входящего сигнала сопоставляется с координатами x точек изгиба, по порядку. Первая пара точек координат, в которой значение x больше значения входящего сигнала принимается за конечную точку линейного отрезка с началом в предыдущей точке. Значение выходящего сигнала вычисляется через линейную функцию на данном линейном отрезке.

При установке параметра *Граница* на 1, значение на выходе будет равно y_1 , если значение на входе меньше x_1 или y_N , если значение на входе больше x_N . При отключении функции *Граница*, значение на выходе вычисляется с линейной экстраполяцией каждый раз, когда входящий сигнал выходит за область, заданную координатами (x_1, x_N) .

На нижеприведенном рисунке представлен график непрерывной функции:



Если значение двух точек координат совпадают, график будет выглядеть следующим образом:



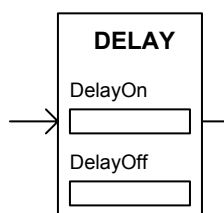
При разрывной функции, значение на входе = x_2 , значение на выходе блока будет = y_2 .

2.10. DATE - Дата**DATE**

| | | |
|--------------|--------|--|
| ТИП ВЫХОДА | ДОСТУП | |
| ЦЕЛЫЙ СИГНАЛ | RO | |

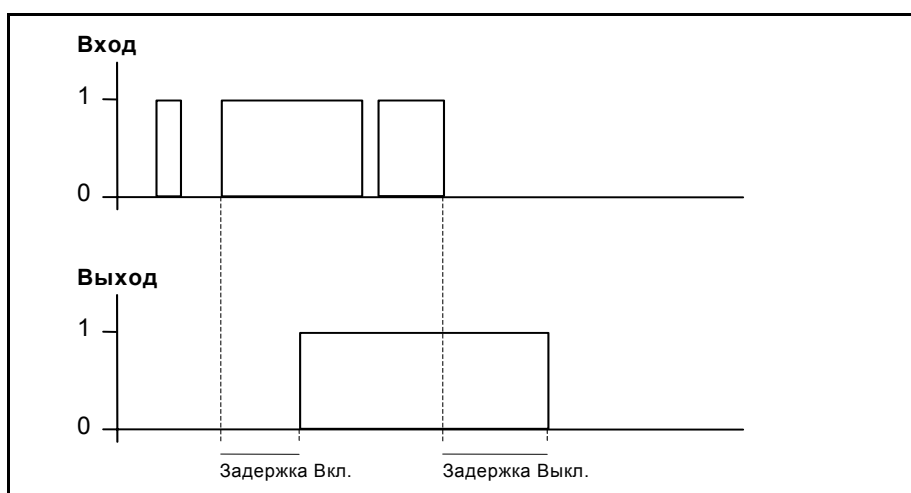
Описание: DATE-блок показывает дату (1-31) согласно данным внутренних часов.

2.11. DELAY - Задержка Включения/Выключения DELAY



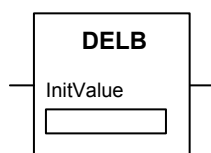
| | | | |
|-----------------|---------------------|----------|--------------------------------|
| ВХОДЫ | Вход | БИНАРНЫЙ | Сигнал на входе |
| ПАРАМЕТРЫ | Задержка включения | ДРОБНЫЙ | Задержка включения в секундах |
| | Задержка выключения | ДРОБНЫЙ | Задержка выключения в секундах |
| ТИП ВЫХОДА | | ДОСТУП | |
| БИНАРНЫЙ СИГНАЛ | | RO | |

Описание: Данный блок задерживает изменение состояния входящего сигнала на время, заданное в секундах параметрами *Задержка включения* (переход от 0 до 1) и *Задержка выключения* (переход от 1 до 0). Обратите внимание на тот факт, что входящий сигнал должен иметь значение ИСТИНА на протяжении времени, превышающего по длительности время *Задержки включения*, для того, чтобы генерировать импульсы на выходе FB (см. нижеприведенную диаграмму времени). Для переключения выхода на значение ЛОЖЬ, входящий сигнал должен иметь значение ЛОЖЬ на протяжении времени, превышающего время *Задержки выключения*.



Если в период задержки включения/ выключения случится сбой напряжения питания, недостающее время задержки будет сохранено только в том случае, если выбрана метка *Backup*.

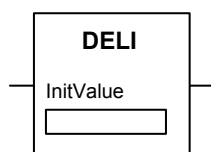
2.12. DELB - Задержка на 1 цикл (Бинарный сигнал) DELB



| | | | |
|-----------------|---------------|----------|--------------------------------------|
| ВХОДЫ | Значение | БИНАРНЫЙ | Бинарный сигнал на входе |
| ПАРАМЕТРЫ | Нач. значение | БИНАРНЫЙ | Начальное значение сигнала на выходе |
| ТИП ВЫХОДА | | ДОСТУП | |
| БИНАРНЫЙ СИГНАЛ | | R/W | |

Описание: Данный блок выполняет функцию задержки бинарного сигнала на 1 программный цикл. Значение сигнала на выходе обновляется в каждом программном цикле с учетом значения входящего сигнала из предыдущего цикла.

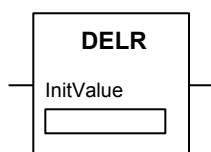
2.13. DELI - Задержка на 1 цикл (Целый сигнал) DELI



| | | | |
|--------------|---------------|--------|--------------------------------------|
| ВХОДЫ | Значение | ЦЕЛЫЙ | Сигнал на входе |
| ПАРАМЕТРЫ | Нач. значение | ЦЕЛЫЙ | Начальное значение сигнала на выходе |
| ТИП ВЫХОДА | | ДОСТУП | |
| ЦЕЛЫЙ СИГНАЛ | | R/W | |

Описание: Данный блок выполняет функцию задержки целого сигнала на 1 программный цикл. Значение сигнала на выходе обновляется в каждом программном цикле с учетом значения входящего сигнала из предыдущего цикла.

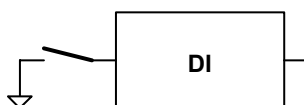
2.14. DELR - Задержка на 1 цикл (Дробный сигнал) DELR



| | | | |
|----------------|---------------|---------|--------------------------------------|
| ВХОДЫ | Значение | ДРОБНЫЙ | Сигнал на входе |
| ПАРАМЕТРЫ | Нач. значение | ДРОБНЫЙ | Начальное значение сигнала на выходе |
| ТИП ВЫХОДА | | ДОСТУП | |
| ДРОБНЫЙ СИГНАЛ | | R/W | |

Описание: Данный блок выполняет функцию задержки дробного сигнала на 1 программный цикл. Значение сигнала на выходе обновляется в каждом программном цикле с учетом значения входящего сигнала из предыдущего цикла.

2.15. DI - Цифровой вход

DI


DI-блок имеет 5 различных режимов, настраиваемых пользователем:

- Физический цифровой вход
- Сетевая переменная
- SNVT
- Устройство НА-Линии
- Постоянное значение (константа)

Значение выхода блока обновляется один раз за программный цикл. Это значит, что изменения входящего сигнала протяженностью менее одного программного цикла не будут учитываться программой приложения.

2.15.1. Физический цифровой вход

| | | | |
|-----------------|-------------------|----------|---|
| ПАРАМЕТРЫ | Номер модуля | НОМЕР | Номер I/O-модуля |
| | Название контакта | СПИСОК | Название контакта |
| | Тип датчика | СПИСОК | Тип датчика |
| | Нормально открыт | БИНАРНЫЙ | Метка выбрана = открытый контакт воспринимается как ЛОЖЬ (0); Значение по умолчанию. Метка не выбрана = открытый контакт воспринимается как ИСТИНА (1). |
| ТИП ВЫХОДА | | ДОСТУП | |
| БИНАРНЫЙ СИГНАЛ | | RO | |

Описание: В режиме *Физический цифровой вход* блок получает значение с цифрового входа и преобразует его в бинарное.

Параметр *Номер модуля* представляет собой номер I/O-модуля (0 = основное устройство). Параметр *Название контакта* задает тип и номер контакта (напр.: U1-U4, X1-X4).

Функция распознавания значения сигнала для физического цифрового входа задается параметром *Нормально открыт*. Если *Нормально открыт* = 1, то это означает, что открытый контакт присваивает выходу блока значение 0 (ЛОЖЬ). Если *Нормально открыт* = 0, то открытый контакт присваивает выходу блока значение 1 (ИСТИНА).

При подключении входа к I/O модулю, начальное значение на выходе блока (до получения значения сигнала с I/O модуля) задается параметром *Начальное значение*. При отключении I/O модуля, выход блока сохранит последнее значение, полученное с I/O модуля.

2.15.2. Сетевая переменная

| | | | |
|-----------------|--------------------|----------|--|
| ПАРАМЕТРЫ | Сетевой адрес | СТРОКА | Соотносится с внешним сигналом сети ТАС. |
| | Начальное значение | БИНАРНЫЙ | Начальное значение на выходе; Включить (1)/Выключить (0). Значение по умолчанию = Выключить (0). |
| | Период | ЦЕЛЫЙ | Максимальный интервал времени (в секундах) между двумя обновлениями посланного значения. Значение по умолчанию = 60 |
| ТИП ВЫХОДА | | ДОСТУП | |
| БИНАРНЫЙ СИГНАЛ | | RO | |

Описание: В режиме *Сетевая переменная* блок используется для получения сигнала от других устройств ТАС к программе приложения через сеть. *Сетевой адрес* представляет собой бинарный сигнал “Public” в другом устройстве ТАС, представленный в виде строки, напр: \RPU1\АНУ1\ExtendedOp (максимальное число знаков \20\20). Изменять сетевой адрес в процессе работы блока нельзя.

Начальное значение на выходе блока (до получения им какого-либо значения сигнала через сеть) задается параметром *Начальное значение*. В случае возникновения ошибки при коммуникации, значение на выходе блока не будет обновляться (т.е. выход всегда будет сохранять последнее записанное значение сигнала, принятого через сеть). После холодного старта и установки RAM-памяти контроллера в начальное положение (=0), значение на выходе блока сбросится до *Начального значения*.

Полученное значение обновляется каждый раз, когда сигнал изменяет свое значение. Независимо от того, происходит изменение сигнала или нет, полученное значение будет обновляться также по истечении времени, заданного параметром *Период*, с момента последнего обновления. (Если значение *Периода* меньше 1 сек, задайте *Период = 1 сек*). Обратите внимание на то, что использование одного сетевого адреса для нескольких DI-блоков, входящих в одну схему FBD, не имеет смысла, т.к. в сети возникнет ненужная нагрузка, если будет послано несколько запросов на получение одного входящего сигнала. Для каждого сигнала используется только один блок, который подключается к другим блокам в схеме FBD, использующей посланный сигнал.

2.15.3. SNVT

| | | | |
|-----------------|--------------------|----------|---|
| ПАРАМЕТРЫ | Тип | СПИСОК | Тип SNVT. |
| | Поле | СПИСОК | Тип SNVT –поля (Применяется только после выбора структурированного типа SNVT). |
| | SNVT - название | СПИСОК | ID сигнала в сети, максимум 16 знаков. |
| | Начальное значение | БИНАРНЫЙ | Начальное значение на выходе. Значение по умолчанию = Выключить (0) |
| | Опрос | БИНАРНЫЙ | Определяет, должно ли быть запрошено значение внешнего сигнала на выходе. |
| | Период | ЦЕЛЫЙ | Максимальный интервал времени (в секундах) между двумя обновлениями посланного значения. Значение по умолчанию =60 |
| ТИП ВЫХОДА | | ДОПУСК | |
| БИНАРНЫЙ СИГНАЛ | | RO | |

Описание: В режиме *SNVT* блок используется для передачи через сеть сигнала от внешнего оборудования в программу приложения. Блок входов должен также быть «привязан» к выходящему сигналу внешнего оборудования (которое должно быть такого же *SNVT*- типа) при помощи программы сетевой адресации. Обратите внимание на то, что сигналу могут присваиваться разные имена в приложении (имя блока, напр., StartButton) и в сети (имя *SNVT*, напр., nvistartbutton).

Значение параметров *Тип* и *Поле* выбираются из предоставленного списка для блоков *SNVT*- типа. Для бинарного сигнала может быть использован переключатель *SNVT_switch*.

Начальное значение на выходе блока (до получения им какого-либо значения сигнала через сеть) задается параметром *Начальное значение*.

В случае возникновения ошибки при коммуникации, значение на выходе блока не будет обновляться (т.е. выход всегда будет сохранять последнее записанное значение сигнала, принятого через сеть).

После «привязки», внешний сигнал на выходе будет опрашиваться автоматически, если выбрана функция *Опрос* и если не произошло обновления значений во время *Периода*.

2.15.4. Устройство НА-Линии

| | | | |
|-----------------|------------------|--------|---------------------------|
| ПАРАМЕТРЫ | Адрес устройства | СТРОКА | Инспектируемое устройство |
| ТИП ВЫХОДА | ДОСТУП | | |
| БИНАРНЫЙ СИГНАЛ | RO | | |

Описание: В режиме *Устройство НА-Линии* блок используется для отслеживания наличия связи с I/O-модулем или любым другим устройством в сети. Данный блок может быть использован, например, в приложениях, в которых сигналы посылаются с других устройств в сети и в которых при обрыве связи должны быть выполнены дополнительные действия. Название устройства (или номер I/O-модуля: 1, 2,...) задается параметром *Адрес устройства*, который вводится в виде строки. Для отслеживания наличия связи системы диспетчеризации ТАС Vista, в качестве названия устройства через программу *Сетевая конфигурация* должно вводиться имя **LonWorks network**. Обратите внимание на тот факт, что контроллер ТАС Xenta может контролировать наличие связи только того I/O-модуля, который входит в его программу приложения.

Выход блока будет иметь значение ИСТИНА (1), если устройство по введенному *Адресу устройства* находится на линии, в противном случае значение на выходе блока будет ЛОЖЬ (0). Если в параметре *Адрес устройства* указан адрес программируемого устройства и в сети не найдено какое-либо другое устройство, находящееся на связи, то выход блока будет иметь значение ЛОЖЬ (0), В противном случае значение на выходе будет ИСТИНА (1).

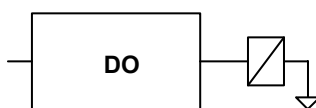
2.15.5. Постоянное значение

| | | | |
|-----------------|--------------------|----------|---|
| ПАРАМЕТРЫ | Начальное значение | БИНАРНЫЙ | Начальное значение на выходе; Включить (1)/Выключить (0). Значение по умолчанию = Выключить (0) |
| ТИП ВЫХОДА | ДОПУСК | | |
| БИНАРНЫЙ СИГНАЛ | RO | | |

Описание: Вход блока не имеет ни физического, ни сетевого адреса. Вместо этого данная конфигурация предоставляет на выход блока постоянное значение, заданное параметром *Начальное значение*.

2.16. DO - Цифровой выход

DO



DI-блок имеет четыре различных режима, настраиваемых пользователем:

- Физический выход
- SNVT
- Не подсоединен

2.16.1. Физический цифровой выход

| ВХОДЫ | Вход | БИНАРНЫЙ | Сигнал на входе |
|----------------------|--------------------|----------|---|
| ПАРАМЕТРЫ | Номер модуля | НОМЕР | Номер I/O - модуля |
| | Название контакта | СПИСОК | Название контакта |
| | Начальное значение | БИНАРНЫЙ | Начальное значение на выходе (только для I/O-модуля); Включить (1)/Выключить (0). Значение по умолчанию = Выключить (0) |
| ТИП ВЫХОДА | | ДОСТУП | |
| Блок не имеет выхода | | RO | |

Описание: Данный блок задает состояние на физическом цифровом выходе.

Параметр *Номер модуля* представляет собой номер I/O-модуля (0 = основное устройство). Параметр *Название контакта* задает тип и номер контакта на выходе (напр.: K1-K4).

При соединении блока с I/O-модулем, параметр *Начальное значение* задает значение выходящего сигнала, если I/O-модуль отключен, напр., при повторном запуске блока в результате сбоя питания.

2.16.2. SNVT

| | | | |
|----------------------|--------------------|----------|---|
| ПАРАМЕТРЫ | Тип | СПИСОК | Тип SNVT. |
| | Поле | СПИСОК | Тип SNVT –поля (Применяется только после выбора структурированного типа SNVT). |
| | SNVT - название | СПИСОК | ID сигнала в сети, максимум 16 знаков. |
| | Начальное значение | БИНАРНЫЙ | Начальное значение на выходе. Значение по умолчанию = Выключить (0) |
| | Отправить | БИНАРНЫЙ | Определяет, должен ли внешний сигнал обновляться автоматически или нет |
| | Период | ЦЕЛЫЙ | Максимальный интервал времени (в секундах) между двумя обновлениями посланного значения. Значение по умолчанию =60 |
| ТИП ВЫХОДА | | ДОСТУП | |
| Блок не имеет выхода | | RO | |

Описание: В режиме *SNVT* блок используется для передачи через сеть сигнала из программы приложения во внешнее оборудование. Блок выходов должен быть «привязан» к входящему сигналу внешнего оборудования (которое должно быть такого же *SNVT*-типа) при помощи программы сетевой привязки. Обратите внимание на то, что сигналу могут присваиваться разные имена в приложении (имя блока, напр., *StartButton*) и в сети (*Имя SNVT*, напр., *nvostartbutton*).

Значение параметров *Тип* и *Поле* выбираются из предоставленного списка для блоков *SNVT*- типа. Для бинарного сигнала может быть использован переключатель *SNVT_switch*.

Начальное значение на выходе блока (перед выполнением первого программного цикла) задается параметром *Начальное значение*.

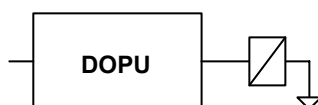
Если выбрана функция *Отправить*, значение внешнего входящего сигнала будет каждый раз обновляться при изменении значения сигнала на входе блока. Независимо от факта изменения значения входящего сигнала, внешний входящий сигнал будет обновляться по прохождении времени, заданного параметром *Период*, после последнего обновления. Если значение *Периода* = 0, внешний сигнал на входе будет обновляться только в случае изменения его статуса.

Если функция *Отправить* не выбрана, для получения обновленных значений внешнее оборудование должно опрашиваться дополнительно.

2.16.3. Не подсоединен

В режиме *Не подсоединен* блок не подсоединен ни к физическому, ни к сетевому адресу. Данный режим не имеет параметров.

2.17. DOPU -Цифровой импульсный выход DOPU



DOPU - блок имеет следующие режимы, настраиваемые пользователем: Цифровой импульсный выход (Физический выход) и Не подсоединен.

2.17.1. Цифровой импульсный выход

| ВХОДЫ | Вход | ДРОБНЫЙ | Длина импульсного сигнала |
|----------------------|--------------------|---------|--|
| ПАРАМЕТРЫ | Номер модуля | НОМЕР | Номер I/O - модуля |
| | Название контакта | СПИСОК | Название контакта |
| | Мин. Импульс | ДРОБНЫЙ | Минимальное значение длины импульсного сигнала на выходе (сек) Значение по умолчанию = 0,5 сек |
| | Начальное значение | ДРОБНЫЙ | Начальное значение импульсного сигнала на выходе (только для I/O-модуля); Значение по умолчанию = 0 сек |
| ТИП ВЫХОДА | | ДОСТУП | |
| Блок не имеет выхода | | RO | |

Описание: Данный блок задает состояние на физическом цифровом импульсном выходе. Длина импульсного сигнала на выходе определяется, исходя из значения входящего сигнала блока (в секундах). DOPU-блок специально разработан для совместного использования с PID-регулятором (Увеличить/Уменьшить) (PIDI-блок). Отрицательные значения на входе блока не учитываются.

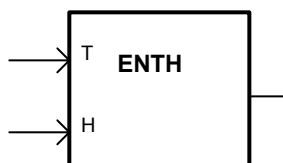
Параметр *Номер модуля* представляет собой номер I/O-модуля (0 = основное устройство). Параметр *Название контакта* задает тип и номер контакта на выходе (напр.: K1-K4).

Параметр *Мин. Импульс* задает минимальное значение длины импульсного сигнала на выходе (в секундах). Все значения входящего сигнала, превышающие данное минимальное значение, накапливаются для следующего цикла в программе приложения. Если протяженность импульсного сигнала на входе превышает по времени цикл программы приложения, то она сокращается до величины, равной времени выполнения программного цикла.

При соединении блока с I/O-модулем, параметр *Начальное значение* задает значение выходящего сигнала в секундах, если I/O-модуль отключен, напр., при повторном запуске блока в результате сбоя питания.

2.17.2. Не подсоединен

В режиме *Не подсоединен* блок не подсоединен ни к физическому, ни к сетевому адресу. Данный режим не имеет параметров.

2.18. ENTH - Расчет энтальпии**ENTH**

| | | | |
|----------------|-----------------------|---------|---------------------------------------|
| ВХОДЫ | Температура (Т) | ДРОБНЫЙ | Температура по сухому термометру (°С) |
| | Влажность воздуха (Н) | ДРОБНЫЙ | Относительная влажность воздуха (%) |
| ТИП ВЫХОДА | | ДОСТУП | |
| ДРОБНЫЙ СИГНАЛ | | RO | |

Описание: Величина энтальпии (кДж/кг) для влажного воздуха при нормальном атмосферном давлении рассчитывается как функция температуры по сухому термометру (°С) и относительной влажности воздуха (%).

См.: *ASHRAE Руководство 1989 – Основы* (Глава 6, Психрометрия).- *ASHRAE Handbook 1989 - Fundamentals* (Chapter 6, Psychrometrics).

Алгоритм:

Давление насыщенного водяного пара при температуре от 0 до 200 °С вычисляется по следующей функции¹:

$$pws(T) = \exp(c8/T + c9 + c10*T + c11*T^2 + c12*T^3 + c13*\ln(T)) \quad (4)$$

где

T = Абсолютная температура (К),
pws = давление насыщения (Па),

а

c8 = -5.8002206E3;
c9 = 1.3914993;
c10 = -4.8640239E-2;
c11 = 4.1764768E-5;
c12 = -1.4452093E-8;
c13 = 6.5459673;

¹ Неизвестные в уравнении в скобках соотносятся с соответствующими уравнениями в Руководстве ASHRAE.

Величина энтальпии влажного воздуха при нормальном атмосферном давлении, рассчитываемая как функция температуры по сухому термометру и относительной влажности воздуха, задается следующими уравнениями:

$$p_{w_{sat}} = p_{ws}(t+273.15)$$

$$p_w = \Phi * p_{w_{sat}} / 100 \quad (22)$$

$$W = 0.62198 * p_w / (p - p_w) \quad (20)$$

$$h = t + W * (2501 + 1.805 * t) \quad (30)$$

в которых

$p_{w_{sat}}$ = давление насыщенного водяного пара (Па)

t = температура по сухому термометру (°C)

Φ = относительная влажность (%)

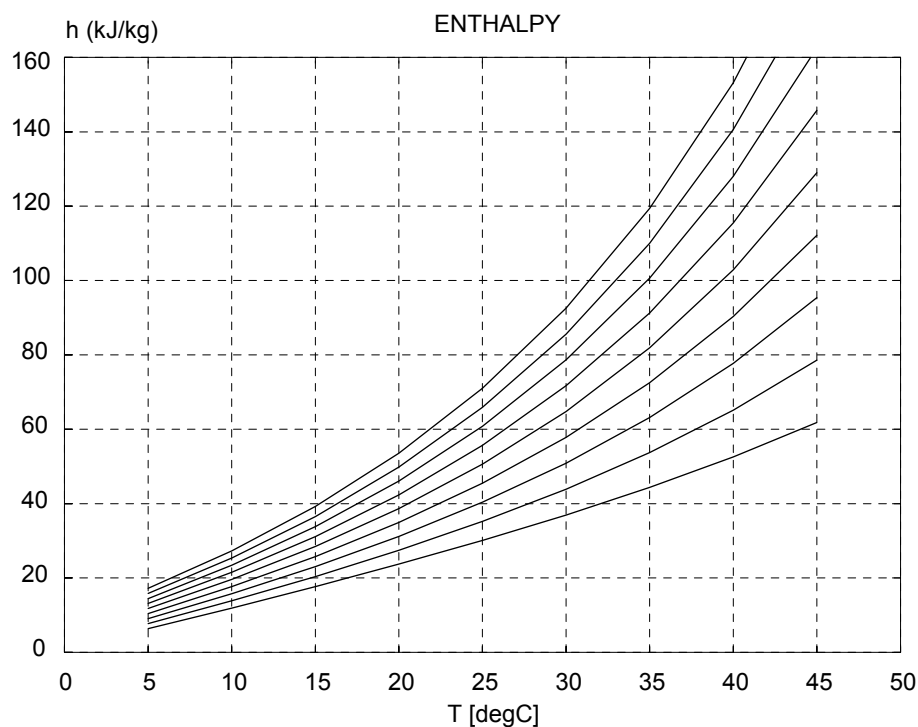
p_w = парциальное давление водяного пара во влажном воздухе (Па)

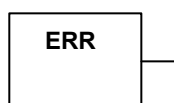
p = давление в окружающей среде (101325 Па)

W = коэффициент влажности, масса воды на общую массу сухого воздуха

h = энтальпия (кДж/кг)

На нижеприведенном графике функции температуры по сухому термометру, энтальпия показана для ряда значений коэффициента влажности. Берущие свое начало в нижней части, кривые графика функции соответствуют коэффициенту влажности воздуха 10%, 20%, 30%, ... 90%.



2.19. ERR - Системные ошибки**ERR**

| | | |
|--------------|--------|--|
| ТИП ВЫХОДА | ДОСТУП | |
| ЦЕЛЫЙ СИГНАЛ | RO | |

Описание: Сигнал на выходе данного блока представляет собой целый сигнал, каждый бит которого является внутренним сигналом или ошибкой системной программы. Каждый бит выходящего сигнала задается при появлении ошибки или во время действия первого программного цикла после повторного запуска программы из-за сбоя питания. Значение сигнала на выходе равно нулю, если ошибок нет.

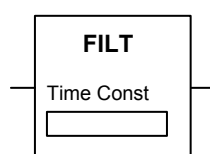
Коды ошибок:

| № Бита | Знач. выход. сигнала | Описание |
|--------|----------------------|---|
| 1 | 1 | Рестарт после сбоя питания |
| 2 | 2 | Исчезновение/переполнение целым сигналом |
| 3 | 4 | Переполнение CNT-блока |
| 4 | 8 | Переполнение RT-блока |
| 5 | 16 | Превышение времени выполнения главной программы. Программа приложения не может быть выполнена в течение времени, приведенного в ее спецификации |
| 6 | 32 | Как минимум один из переключателей одного из блоков расширения (I/O) управляется в ручном режиме (ON, OFF, MAN) |
| 7 | 64 | |
| 8 | 128 | |
| 9 | 256 | Повреждение на линии, соединяющей модем и устройство ТАС Xenta |
| 10 | 512 | Обновленное значение сетевой переменной не было получено в течение заданного времени |
| 11 | 1024 | Существует сигнал аварии первого приоритета в списке аварий |
| 12 | 2048 | Отсутствие связи с одним или несколькими I/O-модулями |

| | | |
|----|-------|--|
| 13 | 4096 | I/O в принудительном режиме |
| 14 | 8192 | Наличие сработавшего и неподтвержденного аварийного сигнала в списке |
| 15 | 16384 | Наличие сработавшего и не сброшенного аварийного сигнала в списке |

ERR -блок используется только в программе приложения.
Подключение нескольких блоков одновременно может привести к значительной перегрузке памяти устройства TAC Xenta.

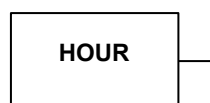
2.20. *FILT* - Фильтры первого порядка *FILT*



| | | | |
|----------------|--------------------|---------|---|
| ВХОДЫ | Переменная | ДРОБНЫЙ | Входящий сигнал |
| ПАРАМЕТРЫ | Постоянная времени | ДРОБНЫЙ | Постоянная времени для фильтра, измеряемая в секундах |
| ТИП ВЫХОДА | ДОСТУП | | |
| ДРОБНЫЙ СИГНАЛ | R/W | | |

Описание: Данный блок представляет собой фильтр первого порядка с постоянной времени, заданной параметром *Постоянная времени*. Начальное значение на выходе блока равно нулю.

2.21. *HOURL* - Текущий час *HOURL*

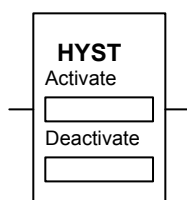


| | | |
|--------------|--------|--|
| ТИП ВЫХОДА | ДОСТУП | |
| ЦЕЛЫЙ СИГНАЛ | RO | |

Описание: Данный блок показывает текущий час (0-23), согласно данным внутренних часов.

2.22. HYST - Бинарный гистерезис

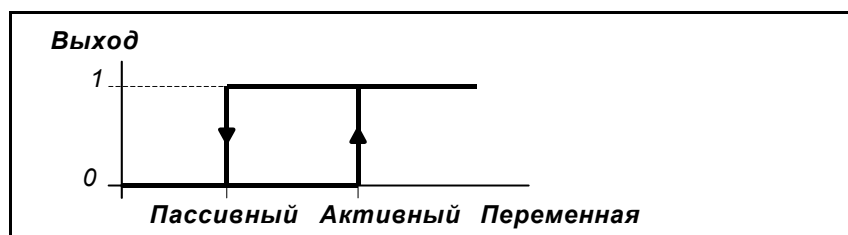
HYST



| | | | |
|-----------------|------------|---------|---|
| ВХОДЫ | Переменная | ДРОБНЫЙ | Входящий сигнал |
| ПАРАМЕТРЫ | Активный | ДРОБНЫЙ | Крайнее значение входящего сигнала для активации исходящего сигнала |
| | Пассивный | ДРОБНЫЙ | Крайнее значение входящего сигнала для деактивации исходящего сигнала |
| ТИП ВЫХОДА | ДОСТУП | | |
| БИНАРНЫЙ СИГНАЛ | RO | | |

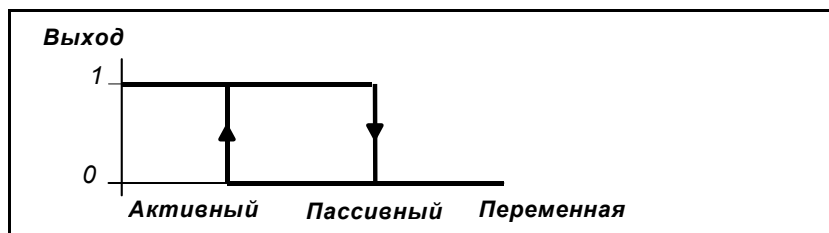
Описание: Данный блок выполняет релейную функцию с гистерезисом.

В том случае, когда значение параметра *Активный* больше, чем значение параметра *Пассивный*, данный блок будет выполнять следующую функцию: если значение на выходе блока ЛОЖЬ (0), а значение входящего сигнала превышает границу области активации, то значение на выходе изменяется на ИСТИНА (1). Если значение на выходе блока ИСТИНА (1), а значение входящего сигнала меньше значения на границе области деактивации, значение на выходе становится ЛОЖЬ (0). В том случае, если входящий сигнал по своему значению находится между границами областей активации и деактивации, выход блока сохраняет свое первоначальное состояние.

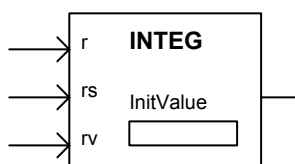


В том случае, когда значение параметра *Активный* меньше, чем значение параметра *Пассивный*, данный блок будет выполнять следующую функцию: если значение на выходе блока ИСТИНА (1), а значение входящего сигнала превышает величину допустимого крайнего значения области деактивации, то значение на выходе изменяется на ЛОЖЬ (0). Если значение на выходе блока ЛОЖЬ (0), а значение входящего сигнала меньше значения на границе области деактивации, значение на выходе становится ИСТИНА (1). В том случае, если входящий сигнал по своему значению находится между границами областей

активации и деактивации, выход блока сохраняет свое первоначальное состояние.



2.23. INTEG - Интегратор (Целые сигналы) INTEG

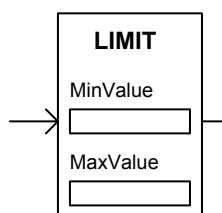


| | | | |
|----------------|--------------------|----------|--|
| ВХОДЫ | Скорость (r) | ДРОБНЫЙ | Интегрируемый сигнал на входе |
| | Сброс (rs) | БИНАРНЫЙ | Вход сброса (1 = сброс) |
| | Предустановка (rv) | ДРОБНЫЙ | Значение на выходе при активизированном входе сброса |
| ПАРАМЕТРЫ | Нач. значение | ДРОБНЫЙ | Начальное значение интегратора |
| ТИП ВЫХОДА | | ДОСТУП | |
| ДРОБНЫЙ СИГНАЛ | | R/W | |

Описание: Данный блок позволяет интегрировать скорость или расход за время цикла. Интеграл вычисляется как сумма скорости, умноженная на инкремент времени, т.е. на время программного цикла. Суммирование производится с большой цифровой точностью во избежание ошибок в случае малой скорости.

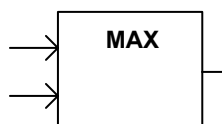
Значение на выходе интегратора будет соответствовать *Начальному значению* в его изначальном состоянии. При активации входа *Сброс*, значение на выходе интегратора сбрасывается до того значения, которое было установлено функцией *Предустановка* входящего сигнала. В случае прекращения использования входа *Сброс*, интегрирование ведется от последнего значения на входе *Предустановка*.

Верхняя и нижняя границы значения на выходе определяются исходя из максимального значения дробного сигнала, установленного в контроллере.

2.24. LIMIT – Ограничение по max/min**LIMIT**

| | | | |
|----------------|----------------|---------|-------------------------------|
| ВХОДЫ | Вход | ДРОБНЫЙ | Сигнал на входе |
| ПАРАМЕТРЫ | Мин. Значение | ДРОБНЫЙ | Ограничение по мин. значению |
| | Макс. Значение | ДРОБНЫЙ | Ограничение по макс. значению |
| ТИП ВЫХОДА | ДРОБНЫЙ | | |
| ДРОБНЫЙ СИГНАЛ | RO | | |

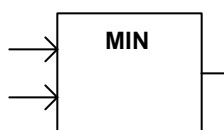
Описание: Сигнал на *Входе* ограничен диапазоном [*Мин. Значение*, *Макс. Значение*].

2.25. MAX – Max показатель для 2 сигналов MAX

| | | | |
|----------------|--------|---------|------------------------|
| ВХОДЫ | Вход1 | ДРОБНЫЙ | Первый входящий сигнал |
| | Вход2 | ДРОБНЫЙ | Второй входящий сигнал |
| ТИП ВЫХОДА | ДОСТУП | | |
| ДРОБНЫЙ СИГНАЛ | RO | | |

Описание: Сигнал на выходе блока MAX задается максимальным значением параметров *Вход1* и *Вход2*.

2.26. MIN – Min показатель для 2 сигналов **MIN**



| | | | |
|----------------|-------|---------|------------------------|
| ВХОДЫ | Вход1 | ДРОБНЫЙ | Первый входящий сигнал |
| | Вход2 | ДРОБНЫЙ | Второй входящий сигнал |
| ТИП ВЫХОДА | | ДОСТУП | |
| ДРОБНЫЙ СИГНАЛ | | RO | |

Описание: Сигнал на выходе блока MIN задается минимальным значением параметров *Вход1* и *Вход2*.

2.27. MINUTE – Текущая минута **MINUTE**



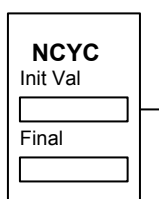
| | | |
|--------------|--------|--|
| ТИП ВЫХОДА | ДОПУСК | |
| ЦЕЛЫЙ СИГНАЛ | RO | |

Описание: Данный блок показывает текущую минуту (0-59) согласно данным внутренних часов.

2.28. MONTH – Текущий месяц**MONTH**

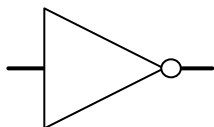
| | | |
|--------------|--------|--|
| ТИП ВЫХОДА | ДОПУСК | |
| ЦЕЛЫЙ СИГНАЛ | RO | |

Описание: Данный блок показывает текущий месяц согласно данным внутренних часов. Месяц 1 – означает Январь, а месяц 12, соответственно, Декабрь.

2.29. NSYC – Счетчик цикла программы**NSYC**

| | | | |
|--------------|-----------------|-------|--------------------------------|
| ПАРАМЕТРЫ | Нач. значение | ЦЕЛЫЙ | Начальное значение на счетчике |
| | Конеч. значение | ЦЕЛЫЙ | Конечное значение на счетчике |
| ТИП ВЫХОДА | ДОПУСК | | |
| ЦЕЛЫЙ СИГНАЛ | R/W | | |

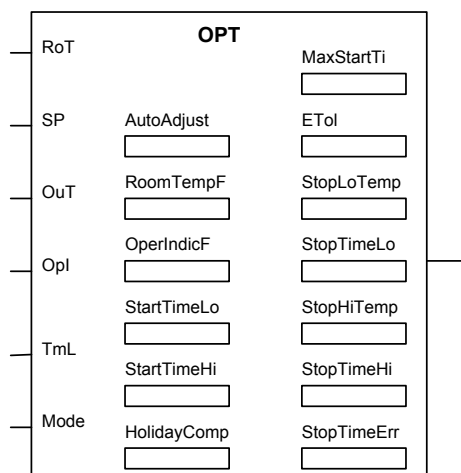
Описание: Данный блок увеличивает значение на выходе на одну единицу в каждом программном цикле, если *Начальное значение* меньше, чем *Конечное*, или уменьшает его в при обратном соотношении *Начального* и *Конечного значения*. Когда счетчик достигает до конечного значения, он сбрасывается и начинает счет в следующем цикле с начального значения.

2.30. NOT – Инверсия бинарного сигнала NOT

| | | | |
|-----------------|--------|----------|-----------------|
| ВХОДЫ | Вход | БИНАРНЫЙ | Входящий сигнал |
| ТИП ВЫХОДА | ДОПУСК | | |
| БИНАРНЫЙ СИГНАЛ | RO | | |

Описание: Данный блок преобразовывает Бинарный сигнал.

2.31. OPT - Оптимизатор **OPT**



| | | | |
|-------|-----------|------------|--|
| ВХОДЫ | RoT | ДРОБНЫЙ | Температура в помещении. |
| | SP | ДРОБНЫЙ | Уставка для температуры в комнате для рабочего времени |
| | OuT | ДРОБНЫЙ | Наружная температура. |
| | OpI | БИНАРНЫЙ | Индикация работы, т.е. внешний сигнал определяющий работает оборудование или нет. OpL=0 => не работает OpL=1 => работает |
| | TmL | ЦЕЛЫЙ | Счетчик времени в минутах показывающий как долго длится рабочее время. Если TmL отрицательное, OPT интерпретирует это значение по истечении которого оборудование должно быть выключено. |
| | Mode | ЦЕЛЫЙ | Выбор режима работы для отключения оптимизации Mode = 0 => обогрев Mode = 1 => охлаждение Mode = -1 => без оптимизации по умолчанию 0. |
| | ПАРАМЕТРЫ | AutoAdjust | ЦЕЛЫЙ |

| | | | |
|--------------|-------------|----------|---|
| ПАРАМЕТРЫ | RoomTempF | БИНАРНЫЙ | Флаг датчика температуры комнаты RoomTempF = 0 => нет датчика RoomTempF = 1 => есть датчик по умолчанию 1. |
| | OperIndicF | БИНАРНЫЙ | Флаг выбора наличия внешнего индикатора работы OperInicF = 0 => нет индикатора OperInicF = 1 => есть индикатор по умолчанию 0. |
| | StartTimeLo | ДРОБНЫЙ | Время запуска (в минутах) при низкой наружной температуре (-10°C/10°C) (верхняя точка по оси Y для графика) по умолчанию 105. |
| | StartTimeHi | ДРОБНЫЙ | Время запуска (в минутах) при высокой наружной температуре (10°C/30°C) (нижняя точка по оси Y для графика) по умолчанию 45. |
| | HolidayComp | ДРОБНЫЙ | Компенсация выходных (%) когда оборудование выкл. На срок больше 48 часов. Максимальная точка для графика компенсации. По умолчанию 0. |
| | MaxStartTi | ДРОБНЫЙ | Максимальное время запуска (часы). По умолчанию 12 |
| | ETol | ДРОБНЫЙ | Отклонение измеряемой температуры от уставки когда происходит переключение с оптимизации на нормальный режим работы. по умолчанию 0.5. |
| | StopLoTemp | ДРОБНЫЙ | Точка низкой наружной температуры для остановки (°C) по умолчанию -10°C |
| | StopTimeLo | ДРОБНЫЙ | Время остановки (минуты) когда наружная температура = StopLoTemp. по умолчанию = 0. |
| | StopHiTemp | ДРОБНЫЙ | Точка высокой наружной температуры для остановки (°C) по умолчанию 10°C |
| | StopTimeHi | ДРОБНЫЙ | Время остановки (минуты) когда наружная температура = StopHiTemp. по умолчанию = 0. |
| | StopTimeErr | ДРОБНЫЙ | Допустимое отклонение от уставки для прекращения нормального режима. Используется для подсчета времени остановки. По умолчанию 0.5 |
| ТИП ВЫХОДА | | ДОПУСК | |
| ЦЕЛЬЙ СИГНАЛ | | R/W | |

Описание: Общее

ОРТ обеспечивает оптимизацию запуска и остановки. ОРТ может применяться как в отопительных, так и в охлаждающих приложениях. Он может использоваться наравне с другими функциональными блоками в приложении. Тем не менее, основной алгоритм работы блока пересчитывается только один раз в минуту. Цель оптимизации времени запуска системы нагрева/охлаждения состоит в получении правильной температуры к началу работы. Цель оптимизации времени остановки системы перед окончанием нормальной работы - в ограничении работы системы отопления/охлаждения перед тем, как помещение будет закрыто на ночь.

Функция оптимизации может быть заблокирована установкой параметра $Mode=-1$. В этом случае Выход блока ОРТ будет повторять сигнал с блока времени, т.е. выход =0, когда $TmL>0$ и выход =1, когда $TmL<0$. Точки графика и *HolidayComp* не настраиваются при $Mode = -1$.

Оптимизация запуска

Время старта подсчитывается каждую минуту, когда $TmL>0$ и функция оптимизации еще не запущена. Если время старта превышает время, оставшееся до начала нормальной работы, то выход блока ОРТ становится =2 (режим оптимизации).

При достижении комнатной температуры значения уставки (SP) минус настраиваемое отклонение $ETol$ (по умолчанию $0,5\text{ }^{\circ}\text{C}$), выход блока будет установлен в 1 (нормальная работа) т.к. цель оптимизации достигнута и нормальная работа может начинаться. Если комнатная температура не достигла значения $SP-ETol$ непосредственно перед переключением $TmL<0$, то выход блока сразу устанавливается равным 1 (нормальная работа). В режиме охлаждения выход устанавливается в 1, когда комнатная температура упадет ниже, чем $SP+ETol$. Принцип оптимизации запуска показан на рисунке 1. На рисунке изображена ситуация когда требуемая температура была достигнута в тот же момент, когда величина TmL стала отрицательной.

Время оптимизации определяется по графику в зависимости от наружной температуры.

Если датчик комнатной температуры не используется, то время старта получается непосредственно с графика с изменением в выходные дни, о чем смотри ниже. Когда подсоединен комнатный датчик, то время старта получается умножением времени полученного с графика на величину разницы между уставкой в помещении SP и температурой в помещении RoT . Например, время старта будет 120 минут если значение получаемое с графика в зависимости от наружной температуры равно 60 мин. И комнатная температура на 2°C ниже уставки.

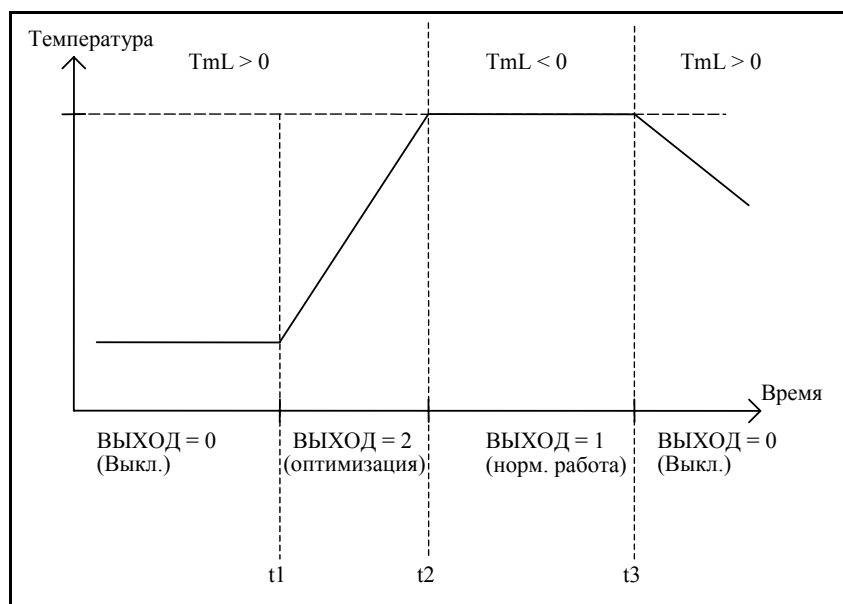


Рис. 1. Принцип оптимизации запуска.

Для получения правильного времени старта после длительного периода выключения, например после выходных, дополнительная величина может быть добавлена ко времени старта. Если параметр *OperIndicF* установлен в 1, блок ОПТ использует вход *OpL* для определения работает оборудование или нет. Если *OperIndicF* установлен в 0, блок ОПТ будет игнорировать вход *OpL* и принимать решение о работе оборудования если выход блока равен 1 (нормальная работа) или 2 (оптимизация).

Время запуска ограничено параметром (*MaxStartTi*).

Выходные значения ОРТ

Как показано на рисунке 1, блок ОРТ может иметь несколько выходных значений:

- Выход = 0 => оборудование выключено (ночь).
- Выход = 1 => оборудование включено (день)
- Выход = 2 => оптимизация

График, время запуска как функция от наружной температуры.

График описывает зависимость между наружной температурой и временем запуска, строится по двум точкам см. рисунок 2. При наличии комнатного датчика время старта будет изменено относительно значения получаемого с графика в зависимости от разницы отклонения уставки от показаний датчика комнатной температуры. Точки для наружной температуры определяются тем какой выбран режим *Mode* (охлаждение или нагревание), время запуска определяется параметрами *StartTimeLo* и *StartTimeHi*. Точки для наружной температуры равны -10 °С и 10 °С для режима нагрева, 10 °С и 30 °С для режима охлаждения. Значения по умолчанию времени запуска 105 минут (*StartTimeLo*) и 45 минут (*StartTimeHi*). Эти точки могут автоматически изменяться, смотри ниже.

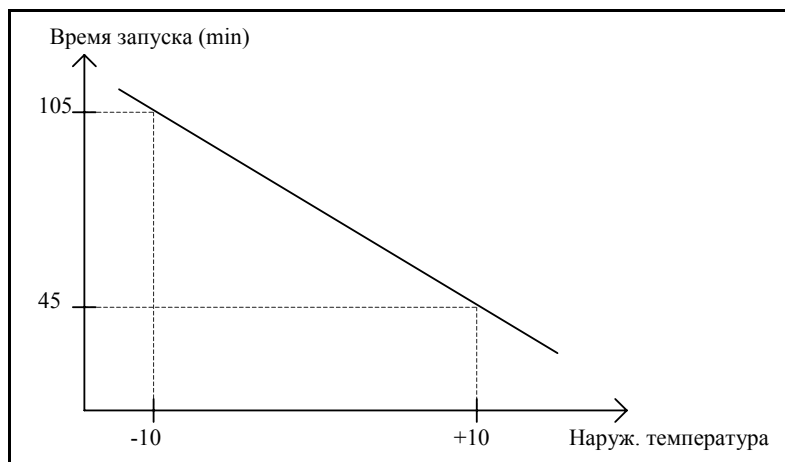


Рис. 2. График описывает зависимость между наружной температурой и временем запуска.

Автоматическая настройка точек графика.

Для корректировки температурных процессов происходящих в здании используется механизм настройки точек графика. Если требуемая температура в помещении не будет достигнута по истечении времени оптимизации (или, наоборот будет достигнута слишком рано) механизм настройки подсчитывает требуемое время оптимизации и изменяет точки графика в соответствии с этим. Этот режим работает только при наличии датчика комнатной температуры.

Настройка точек графика будет производиться только в случае нормального запуска после оптимизации. Для этого необходимо выбрать значение *AutoAdjust* >0 . Настройка графика не будет происходить если оборудование выключено на период больше чем 20 часов.

В любое время вы можете вносить изменения в точки графика используя панель оператора. Если эти изменения были произведены в течении времени оптимизации то изменения вступят в силу только в следующем цикле.

Компенсация выходных (эффект понедельника).

Когда оборудование выключается на долгий период, то для достижения нормальной температуры требуется большее время прогрева т.к. здание было значительно охлаждено. Для компенсации охлаждения происходит добавление времени старта в процентах, если оборудование было выключено на время больше 20 часов, как показано на рисунке 3. Величина максимальной компенсации не возрастает больше чем значение *HolidayComp* и это значение будет при времени выключения больше 48 часов. Проценты увеличиваются линейно с 0% при 20 часах выключения до *HolidayComp* при 48 часах выключения..

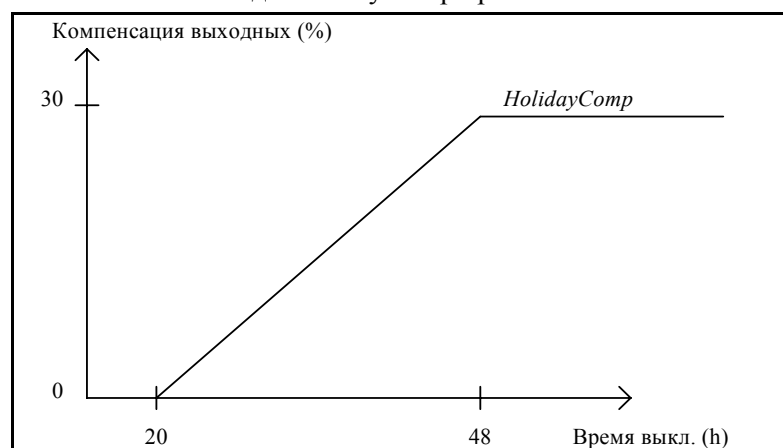


Рис. 3. График зависимости компенсации от длительности выключения. Компенсация выходных = 30%.

Данная функция более важна для систем без комнатного датчика. В системах с таким датчиком запуск будет производиться раньше, т.к. температура в помещениях упадет ниже, чем в обычные выключения.

Автоматическая настройка компенсации выходных

Автоматическая настройка необходима в случае когда оборудование включилось в нормальный режим после фазы оптимизации но перед этим было выключение на период больше 30 часов. Для выполнения этой настройки необходимо выбрать параметр *AutoAdjust* = 2. Если выбрано другое значение то компенсация не будет происходить.

Оптимизация времени выключения

Причиной использования времени выключения является экономия энергии при остановке процесса отопления/охлаждения перед тем, как помещение освободится. Важно следить за тем, чтобы комнатная температура не упала слишком низко, если помещение еще занято. При выборе времени остановки следует учитывать падение наружной температуры в соответствии с падением температуры в помещении, поэтому обычно этот режим используют только в системах с комнатным датчиком. Время остановки определяется графиком, описывающим зависимость между наружной температурой и временем остановки. График дает время остановки на каждый градус отклонения между комнатной температурой и нижней допустимой температурой в комнате перед концом рабочего дня, в случае использования комнатного сенсора. Нижняя допустимая температура получается при вычитании из *SP* величины *StopTimeErr* в режиме отопления. Величина *StopTimeErr* по умолчанию = 0,5 °C. При режиме охлаждения верхняя допустимая температура получается при сложении *SP* и *StopTimeErr*. Если комнатный датчик не используется, то функция оптимизации берет за комнатную температуру значение превышающее *SP* на один градус при отоплении и на один градус ниже при охлаждении. Точки по оси-Х (*StopLoTemp* и *StopHiTemp*) и по оси - У (*StopTimeLo* и *StopTimeHi*) являются свободно изменяемыми. Значения по умолчанию (-10°C, 0 мин.) и (10°C, 0 мин.). Это означает, что не происходит оптимизации времени остановки, если эти параметры не изменены. Эта оптимизация используется относительно редко.

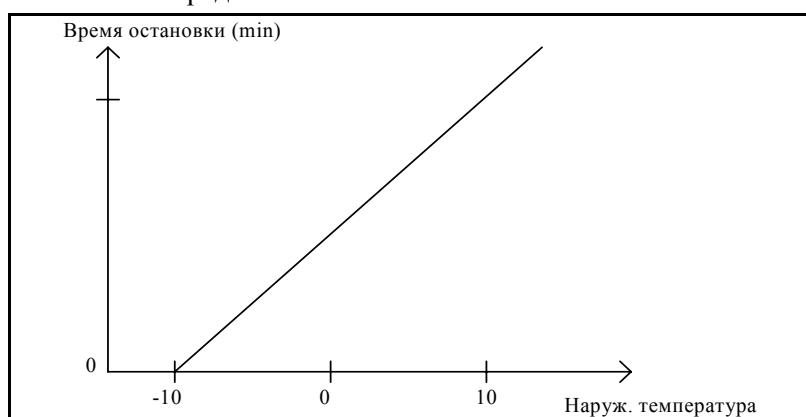
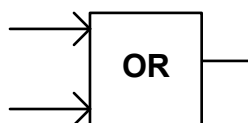


Рис. 4. График отношения наружной температуры и времени остановки. Значение по умолчанию на графике не указано.

2.32. OR – «ИЛИ» - функция на 2 бинарных сигнала OR



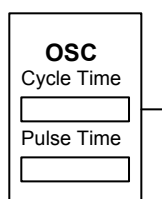
| | | | |
|------------|-------|----------|-------------------|
| ВХОДЫ | Вход1 | БИНАРНЫЙ | Входящий сигнал 1 |
| | Вход2 | БИНАРНЫЙ | Входящий сигнал 2 |
| ТИП ВЫХОДА | | ДОПУСК | |
| БИНАРНЫЙ | | RO | |

Описание: Данный блок рассчитывает булеву «ИЛИ» -функцию для *Входа1* и *Входа2* согласно нижеприведенной таблицы истинности:

| Вход1 | Вход2 | Выход |
|-------|-------|-------|
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 1 |

2.33. OSC – Осциллятор

OSC



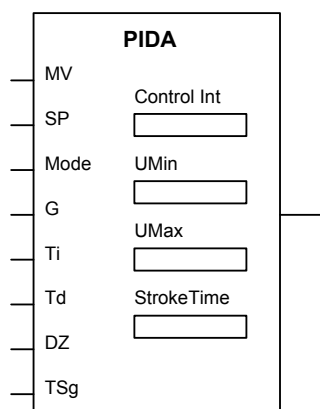
| | | | |
|-----------------|---------------|---------|--------------------------------|
| ПАРАМЕТРЫ | ВремяЦикла | ДРОБНЫЙ | Период генерирования (секунды) |
| | ВремяИмпульса | ДРОБНЫЙ | Длина импульса (секунды) |
| ТИП ВЫХОДА | | ДОПУСК | |
| БИНАРНЫЙ СИГНАЛ | | RO | |

Описание: Данный блок генерирует импульсы длиной, равной значению параметра *ВремяИмпульса*, с периодичностью, заданной параметром *ВремяЦикла*.

Последовательность импульсов всегда кратна времени программного цикла, т.е. если *ВремяЦикла* равно 7,4 секунды,

ВремяИмпульса – 5,2 секунды, а время программного цикла составляет 1 секунду, то каждые 8 секунд подается импульс, продолжительностью 6 секунд

2.34. PIDA - PID регулятор – Аналог. выход PIDA



| | | | |
|----------------|------------|---------|--|
| ВХОДЫ | MV | ДРОБНЫЙ | Измеряемое значение |
| | SP | ДРОБНЫЙ | Заданное значение |
| | Mode | ЦЕЛЫЙ | Рабочий режим регулятора. |
| | G | ДРОБНЫЙ | Пропорциональное усиление |
| | Ti | ДРОБНЫЙ | Время интеграции (сек). |
| | Td | ДРОБНЫЙ | Время дифференцирования (сек). |
| | DZ | ДРОБНЫЙ | Мертвая зона |
| | TSg | ДРОБНЫЙ | Сопровождающий сигнал (актуальное значение предшествующего управляющего сигнала) |
| ПАРАМЕТРЫ | ControlInt | ДРОБНЫЙ | Интервал управления (сек) |
| | UMin | ДРОБНЫЙ | Минимальный возможный управляющий сигнал |
| | UMax | ДРОБНЫЙ | Максимальный возможный управляющий сигнал |
| | StrokeTime | ДРОБНЫЙ | Время полного хода (поршня) привода (сек) |
| ТИП ВЫХОДА | | ДОПУСК | |
| ДРОБНЫЙ СИГНАЛ | | R/W | |

Описание: Алгоритм управления

Блок PIDA предназначен для использования в управляющем контуре, в котором выход регулятора либо соединен с аналоговым физическим выходом, либо используется как заданное значение для другого управляющего контура (каскадное управление). Алгоритм управления представляет собой пошаговый PID алгоритм для дискретного времени; изменение управляющего $du(t)$ рассчитывается по формуле:

$$(1) \quad du(t) = G \cdot (e(t) - e(t-h)) + \frac{h}{Ti} \cdot e(t) - Td \cdot \frac{y(t) - 2 \cdot y(t-h) + y(t-2h)}{h}$$

где e – ошибка регулирования, y – измеряемое значение (MV), G – усиление регулятора, Ti – время интеграции, Td –

дифференциальное время, а h – интервал управления (*ControlInt*), т.е. отрезок времени между двумя успешными обновлениями значения сигнала на выходе регулятора. Если значение параметра *ControlInt* задать как 0, то интервал управления автоматически станет равным времени программного цикла. Индекс времени t представляет собой текущее значение переменной, $t-h$ является значением алгоритма управления, полученным при предшествующем обновлении и т.д.

Блок PID функционирует с интервалом времени hx секунд, где hx секунды соответствуют интервалу времени между исполнениями программы приложения, даже если выбранный интервал управления больше. Интервал управления, h , должен быть кратным времени hx . Если данное условие не выполняется, то алгоритм PID автоматически выберет в качестве управляющего сигнала ближайшее кратное значение, меньшее, чем h .

Ошибка регулирования e определяется как $e = SP - MV$. Таким образом, если измеряемое значение меньше заданного, а усиление G положительно, значение на выходе регулятора увеличится (управление нагреванием). При отрицательном усилении G значение на выходе регулятора, наоборот, станет меньше (управление охлаждением). В том случае, если ошибка управления меньше значения в мертвой зоне, т.е. $\text{abs}(e) < DZ$, величина изменения значения на выходе регулятора, $du(t)$, будет равна нулю. Единицы измерения для параметров *Мертвая зона*, *Измеряемое значение* и *Заданное значение* совпадают.

Значение управляющего сигнала на выходе регулятора можно рассчитать по формуле:

$$(2) \quad u(t) = u(t-h) + du(t)$$

где $u(t)$ это текущий сигнал управления, а $u(t-h)$ - предыдущее значение данного сигнала. Значение для $u(t-h)$ берется со входа *TSg*, который представляет собой сопровождающий сигнал, т.е. *актуальное значение предшествующего сигнала управления*, учитывающий любые внешние ограничения и/или функции опережения программы приложения. Как правило, вход *TSg* соединяется непосредственно с выходом регулятора.

Пропорциональный диапазон соотносится с определенным пропорциональным усилением и вычисляется по формуле:

$$(3) \quad Pband = \frac{UMax - UMin}{G}$$

***P* и *PD* регулятор**

Описанный выше алгоритм применяется при наличии у регулятора функции интегрирования. В том случае, если для работы требуется регулятор без I- или D- функций (I – интегрирование, D - дифференцирование), то значение для Ti или Td соответственно задается равным 0. По аналогии с этим установите функции PI-регулятора, значение $Ti \neq 0$, а $Td = 0$. При усилении G равном 0, программа будет выполняться, но

управляющий сигнал не изменится, независимо от размера ошибки.

Если в пошаговом алгоритме управления – формула (1) – не используется функция интегрирования, существует вероятность возникновения относительно большой ошибки, величина которой не обязательно будет уменьшаться при увеличении усиления регулятора. В следствие этого нами используется специальный алгоритм для Р и PD управления, в котором управляющий сигнал вычисляется по формуле (4):

$$(4) u(t) = G \cdot (e(t) - Td \cdot \frac{(y(t) - y(t-h))}{h}) + \frac{(UMax + UMin)}{2}$$

где $UMax$ – максимальный, а $UMin$ – минимальный возможный управляющий сигнал.

Ограничение значения управляющего сигнала

Максимальная величина изменения значения на выходе за время одного интервала управления, $DuMax$, зависит от времени хода привода и вычисляется по формуле

$$(5) \quad DuMax = \frac{(UMax - UMin) \cdot h}{StrokeTime}$$

Рассчитанная величина изменения значения на выходе регулятора, $du(t)$, должна находиться в диапазоне $\pm DuMax$ до вычисления абсолютного уровня управляющего сигнала. Полученное значение нового управляющего сигнала $u(t)$ ограничена интервалом $(UMin, UMax)$. Если $UMin$ и/или $UMax$ не определены, данное ограничение не выполняется. Параметры $UMin$ и $UMax$ задаются в соответствующих единицах измерения. Значения по умолчанию 0 и 100 (%) соответственно.

Параметр $StrokeTime$ необходим для определения времени реального полного хода привода. Обратите внимание на то, что $StrokeTime$ может быть использован для ограничения величины изменения управляющего сигнала даже, если выход не соединен с приводом. В подобном случае параметр $StrokeTime$ представляет собой минимальное возможное время, которое необходимо для изменения значения управляющего сигнала с $UMin$ на $UMax$. Если значение на выходе регулятора используется в качестве задания для других регуляторов, и если отсутствует необходимость ограничения изменения управляющего сигнала, параметр $StrokeTime$ нужно задать равным 0.

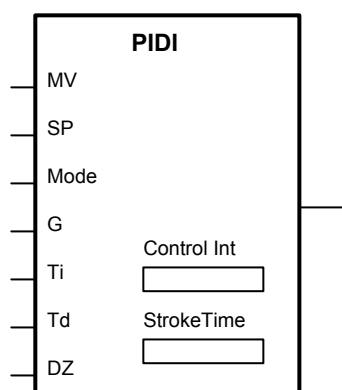
Рабочий режим

Рабочий режим регулятора зависит от значения входящего сигнала $Mode$, см. приведенное ниже описание:

- $Mode = 0 \Rightarrow$ Выключен, регулятор остановлен ($du = 0$)
- $Mode = 1 \Rightarrow$ Нормальный режим управления.
- $Mode = 2 \Rightarrow$ Значение на выходе регулятора равно $UMax$.
- $Mode = 3 \Rightarrow$ Значение на выходе регулятора равно $UMin$.

При $Mode = 0$, выход регулятора будет поддерживать сигнал, соответствующий *Сопровождающему сигналу* (TSg) на входе. Если $Mode < 0$ или $Mode > 3$, то рабочий режим регулятора будет в состоянии Выключен (также, как при $Mode = 0$).

2.35. PIDI-PID регулятор - Увеличить/Уменьшить PIDI



| | | | |
|----------------|------------|---------|----------------------------------|
| ВХОДЫ | MV | ДРОБНЫЙ | Измеряемое значение. |
| | SP | ДРОБНЫЙ | Заданное значение |
| | Mode | ЦЕЛЫЙ | Рабочий режим регулятора |
| | G | ДРОБНЫЙ | Пропорциональное усиление |
| | Ti | ДРОБНЫЙ | Время интеграции (сек). |
| | Td | ДРОБНЫЙ | Время дифференцирования (сек). |
| | DZ | ДРОБНЫЙ | Мертвая зона |
| ПАРАМЕТРЫ | ControlInt | ДРОБНЫЙ | Интервал управления (сек) |
| | StrokeTime | ДРОБНЫЙ | Время полного хода привода (сек) |
| ТИП ВЫХОДА | | ДОПУСК | |
| ДРОБНЫЙ СИГНАЛ | | R/W | |

Описание: Алгоритм управления

Блок PIDI предназначен для использования вместе с двумя блоками цифрового импульсного выхода (DOPU) в управляющем контуре с приводами типа Увеличить/Уменьшить. Алгоритм управления представляет собой пошаговый PID алгоритм для дискретного времени, в котором вычисляемая величина изменения (приращение) управляющего сигнала преобразована в соответствующее время хода привода. Изменение управляющего сигнала $du(t)$ вычисляется по той же формуле, что и для блока PIDA (см. уравнение (1) в описании PIDA блока).

Вычисляемое изменение управляющего сигнала $du(t)$ преобразовывается в величину, соответствующую времени хода привода (в секундах) согласно приведенной ниже формуле:

$$(2) \quad dt = \frac{du(\%)}{100\%} \cdot StrokeTime$$

где $StrokeTime$ – время полного хода привода (в секундах). Обратите внимание, что величина $StrokeTime$ соответствует 100% изменению позиции привода. Значение на выходе блока PIDI представляет собой рассчитанную величину dt .

Если *StrokeTime* равно 0, то время хода автоматически задается в 60 секунд.

Если *ControlInt* равен 0, то интервал управления автоматически становится равным времени цикла программы приложения. Однако, если *ControlInt* больше времени цикла (*hx*), то необходимо принять во внимание, что DOPU блок не работает дольше *hx* секунд до того, как начнется вычисление нового значения на выходе PIDI блока. Это означает, что время на выходе *dt*, рассчитываемое по формуле (2), должно быть поделено на отрезки, продолжительность которых не может превышать *hx*.

Например: если *ControlInt* = 10 с, *hx* = 1 с and *dt* = 5,5 с, сигнал на выходе PIDI блока будет равен 1 для 5 циклов, затем 0,5 для шестого цикла и , наконец, 0 для следующих 4 циклов, пока не наступит время для нового расчета *dt*.

Пропорциональный диапазон соотносится с определенным пропорциональным усилением и вычисляется по формуле:

$$(3) \quad Pband = \frac{100\%}{G}$$

***P* и *PD* регулятор**

Регулятор с выходом увеличить/уменьшить плохо функционирует при отсутствии у него I-функции. Существует возможность возникновения относительно большой постоянной ошибки, которая не обязательно будет уменьшаться при увеличении усиления регулятора. По этой причине лучше отказаться от использования регулятора с выходом увеличить/уменьшить, не имеющего I-функции. Однако, если вы все-таки желаете работать с регулятором без I-функции, задайте значение параметров *Ti* и *Td*, соответственно, равным 0. По аналогии с этим установите функции PI-регулятора: значение *Ti* ### 0, а *Td* = 0. При усилении *G* равном 0, программа будет выполняться, но управляющий сигнал не изменится, независимо от размера ошибки.

Ограничение значения управляющего сигнала

Рассчитываемое время хода ограничено интервалом \pm *ControlInt* (сек), поскольку это максимальное значение времени хода штока привода, которое возможно в пределах одного управляющего интервала.

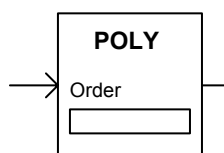
Рабочий режим

Рабочий режим регулятора зависит от значения входящего сигнала *Mode*, см. приведенное ниже описание:

- *Mode* = 0 => Выключен, регулятор остановлен (*dt* = 0)
- *Mode* = 1 => Нормальный режим управления.
- *Mode* = 2 => Значение на выходе регулятора приближается к max (*dt* = *ControlInt* сек).

- $Mode = 3 \Rightarrow$ Значение на выходе регулятора приближается к $\min (dt = -ControlInt \text{ sec})$.

Обратите внимание на то, что при $Mode = 2$ или 3 время хода задается как минимальное, что позволяет приводу безостановочно перемещаться в желаемом направлении. Если $Mode < 0$ или $Mode > 3$, то рабочий режим регулятора будет в состоянии Выключен (также, как при $Mode = 0$).

2.36. POLY – Полиномы**POLY**

| | | | |
|----------------|----------------------|---------|---|
| ВХОДЫ | переменная | ДРОБНЫЙ | |
| ПАРАМЕТРЫ | Список коэффициентов | ДРОБНЫЙ | Список полиномиальных коэффициентов a_0, \dots, a_n |
| ТИП ВЫХОДА | ДОСТУП | | |
| ДРОБНЫЙ СИГНАЛ | RO | | |

Описание: Этот блок вычисляет полиномиальную функцию, заданную выражением:

$$p(x) = a_n \cdot x^n + a_{n-1} \cdot x^{n-1} + \dots + a_1 \cdot x + a_0$$

Порядок полиномов (n) равен номеру коэффициента минус один, что отражается на графическом изображении функционального блока. Максимальное число коэффициентов равно 225.

Вычисление значения полинома $p(x)$ осуществляется по алгоритму, известному как схема Хорнера (Horner's scheme) [См: Fröberg, Carl-Erik: "Numerical Mathematics - Theory and Computer Applications", Addison-Wesley (1985)] для того, чтобы увеличить точность вычислений и сократить количество операций с плавающей запятой. Алгоритм может быть представлен в виде следующих уравнений, где

$$p(x) = b_n:$$

$$b_0 = a_n$$

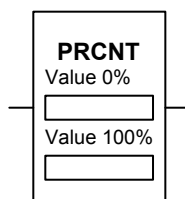
$$b_1 = b_0 \cdot x + a_{n-1}$$

$$b_2 = b_1 \cdot x + a_{n-2}$$

....

$$b_n = b_{n-1} \cdot x + a_0$$

2.37. PRCNT – Процентное соотношение PRCNT



| ВХОДЫ | вход | ДРОБНЫЙ | сигнал на входе |
|----------------|--------------|---------|--|
| ПАРАМЕТРЫ | значение0% | ДРОБНЫЙ | Значение сигнала на входе, который составляет 0% на выходе |
| | значение100% | ДРОБНЫЙ | Значение сигнала на входе, который составляет 100% на выходе |
| ТИП ВЫХОДА | | ДОПУСК | |
| ДРОБНЫЙ СИГНАЛ | | RO | |

Описание: Выполняет простое линейное преобразование входящего сигнала с использованием трансформатора:

$$\text{выход} = 100 * (\text{вход} - \text{значение}0\%) / (\text{значение}100\% - \text{значение}0\%)$$

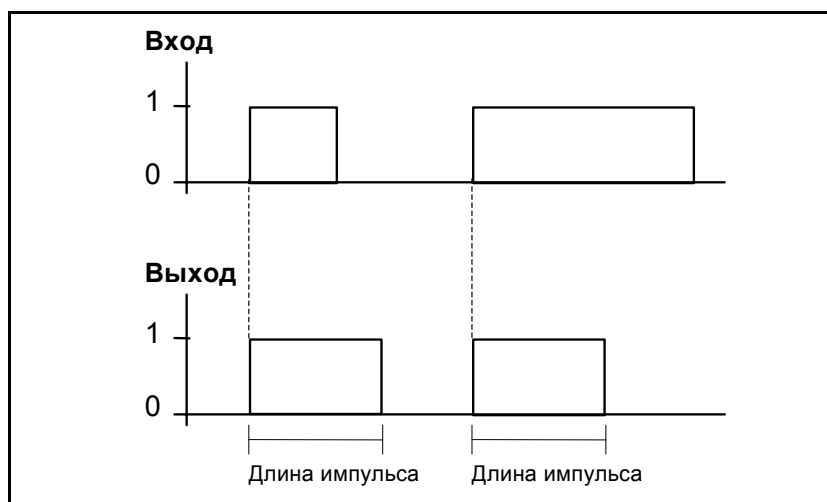
Значение на выходе всегда находится в пределах 0 и 100.

2.38. PULSE – Импульсный генератор PULSE

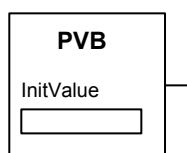


| | | | |
|-----------------|--------------------|----------|---|
| ВХОДЫ | Триггер (t) | БИНАРНЫЙ | Входящий сигнал |
| | Длина импульса(pl) | ДРОБНЫЙ | Продолжительность выходящего импульса (секунды) |
| ТИП ВЫХОДА | | ДОПУСК | |
| БИНАРНЫЙ СИГНАЛ | | RO | |

Описание: Функциональный блок с моностабильным бинарным выходом с изменяемой длительностью импульса (импульсный генератор). Длительность импульса задается значением входящего сигнала *Длина импульса*. Данный блок имеет только один стабильный выход (0). Когда входящий сигнал *Триггер* переключается с 0 на 1, то выход переходит в нестабильное состояние (1) и остается в нем на протяжении времени импульса, после чего возвращается в 0. Выходящий импульс переключается под воздействием восходящего сигнала на входе, а не ровного. Таким образом длительность выходящего импульса не зависит от длительности входящего импульсного сигнала.



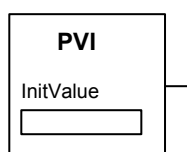
Длительность выходящего импульса всегда кратна времени программного цикла, т.е. если значение на входе *Длина импульса* задано как 5.2 сек, а время программного цикла составляет 1 сек, тогда продолжительность выходящего импульса будет равна 6 сек.

2.39. PVB – Бинарная переменная**PVB**

| | | | |
|-----------------|--------------------|----------|------------------------------|
| ПАРАМЕТРЫ | Начальное значение | БИНАРНЫЙ | Начальное значение на выходе |
| ТИП ВЫХОДА | | ДОПУСК | |
| БИНАРНЫЙ СИГНАЛ | | R/W | |

Описание: Данный блок присваивает выбранное пользователем переменное бинарное значение сигналу на входе другого блока. Для того, чтобы иметь доступ к данному блоку с панели оператора (OP) или через сеть, выход блока должен быть обозначен как “Public”. Начальное значение на выходе задается параметром *Начальное значение*.

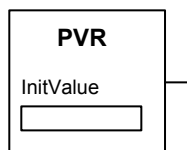
Если выходящий сигнал не является “Public”, данный блок функционирует в качестве параметра постоянного значения для входа другого блока, т.к. данное значение остается неизменным во время выполнения программы приложения.

2.40. PVI – Целая переменная**PVI**

| | | | |
|--------------|--------------------|--------|------------------------------|
| ПАРАМЕТРЫ | Начальное значение | ЦЕЛЫЙ | Начальное значение на выходе |
| ТИП ВЫХОДА | | ДОПУСК | |
| ЦЕЛЫЙ СИГНАЛ | | R/W | |

Описание: Данный блок присваивает выбранное пользователем переменное целое значение сигналу на входе другого блока. Для того, чтобы иметь доступ к данному блоку с панели оператора (OP) или через сеть, выход блока должен быть обозначен как “Public”. Начальное значение на выходе задается параметром *Начальное значение*.

Если выходящий сигнал не является “Public”, данный блок функционирует в качестве параметра постоянного значения для входа другого блока, т.к. данное значение остается неизменным во время выполнения программы приложения.

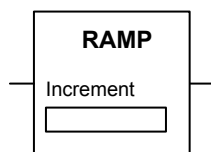
2.41. PVR – Дробная переменная**PVR**

| ПАРАМЕТРЫ | Начальное значение | ДРОБНЫЙ | Начальное значение на выходе |
|----------------|--------------------|---------|------------------------------|
| ТИП ВЫХОДА | | ДОПУСК | |
| ДРОБНЫЙ СИГНАЛ | | R/W | |

Описание: Данный блок присваивает выбранное пользователем переменное дробное значение сигналу на входе другого блока. Для того, чтобы иметь доступ к данному блоку с панели оператора (OP) или через сеть, выход блока должен быть обозначен как “Public”. Начальное значение на выходе задается параметром *Начальное значение*.

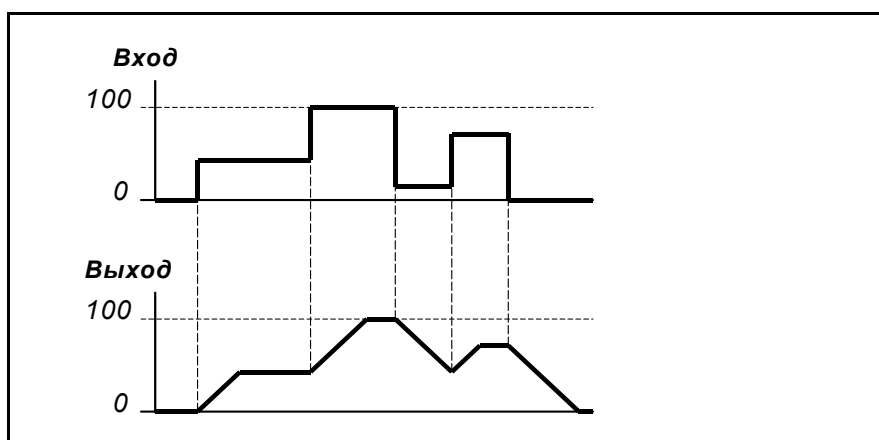
Если выходящий сигнал не является “Public”, данный блок функционирует в качестве параметра постоянного значения для входа другого блока, т.к. данное значение остается неизменным во время выполнения программы приложения.

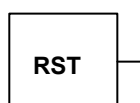
2.42. RAMP – Ограничитель скорости RAMP
изменения сигнала



| | | | |
|----------------|------------|---------|---|
| ВХОДЫ | Вход | ДРОБНЫЙ | Входящий сигнал |
| ПАРАМЕТРЫ | Увеличение | ДРОБНЫЙ | Максимальное увеличение скорости сигнала в секунду. |
| ТИП ВЫХОДА | ДОПУСК | | |
| ДРОБНЫЙ СИГНАЛ | R/W | | |

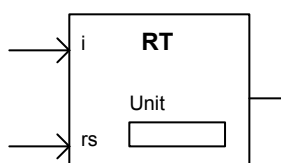
Описание: Данный фильтр функционирует в качестве ограничителя скорости, т.е. он ограничивает скорость изменения сигнала на *Входе*. Максимальное увеличение скорости выходящего сигнала в сек задается параметром *Увеличение*. Если параметр *Увеличение* не используется, то в качестве ограничителя скорости используется абсолютное значение.



2.43. RST – Рестарт**RST**

| | | |
|-----------------|--------|--|
| ТИП ВЫХОДА | ДОПУСК | |
| БИНАРНЫЙ СИГНАЛ | RO | |

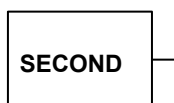
Описание: Выход активизируется в первом программном цикле после теплого старта.

2.44. RT – Учет времени работы**RT**

| | | | |
|--------------|-------------------|----------|---|
| ВХОДЫ | Индикация (i) | БИНАРНЫЙ | Running indication to be measured |
| | Сброс (rs) | БИНАРНЫЙ | Сброс на входе (1 = сброс) |
| ПАРАМЕТРЫ | Единица измерения | ЦЕЛЫЙ | Единица измерения времени на выходе (0 = часы, 1 = минуты, 2 = секунды). Значение по умолчанию = 0 (часы) |
| ТИП ВЫХОДА | ДОПУСК | | |
| ЦЕЛЫЙ СИГНАЛ | R/W | | |

Описание: Данный блок используется для того, чтобы суммировать время, в течение которого активизирован бинарный сигнал *Индикация* (= истина). Единица измерения времени на выходе (часы, минуты, секунды) задается параметром *Единица измерения*. При инициализации или при активации входа *Сброс* значение на выходе становится равным 0 (ЛЮЖБ). В случае деактивации входа *Сброс*, суммирование времени продолжится.

По достижении выходом максимального целого предела (32767), процесс суммирования прекращается, но значение на выходе *не сбрасывается* до нуля.

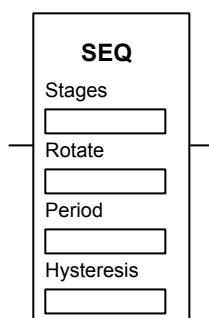
2.45. SECOND – Текущая секунда**SECOND**

| | | |
|--------------|--------|--|
| ТИП ВЫХОДА | ДОПУСК | |
| ЦЕЛЫЙ СИГНАЛ | RO | |

Описание: Данный блок показывает текущую секунду (0-59) согласно данным внутренних часов.

Обратите внимание, что выходного сигнала не меняется в течении одного программного цикла. Данный факт необходимо учитывать при использовании рабочих циклов продолжительностью более 1 сек.

2.46. SEQ – Последовательное управление SEQ



| ВХОДЫ | Вход | ДРОБНЫЙ | Сигнал на входе (%). |
|--------------|------------|----------|---|
| ПАРАМЕТРЫ | Ступень | ДРОБНЫЙ | Кол-во ступеней гистерезиса на выходе (1-16). |
| | Поворот | БИНАРНЫЙ | Определяет наличие поворота ступеней гистерезиса. |
| | Период | ЦЕЛЫЙ | Задержка одновременной активизации ступеней гистерезиса, в миллисекундах. |
| | Гистерезис | ДРОБНЫЙ | Ширина гистерезиса (%). |
| ТИП ВЫХОДА | | ДОПУСК | |
| ЦЕЛЫЙ СИГНАЛ | | RO | |

Описание: Данный блок используется для активации N из M, где M – общее число ступеней на выходе, заданное параметром *Ступени* (до 16 ступеней max), а N – целая часть числа, полученного в результате следующих вычислений:

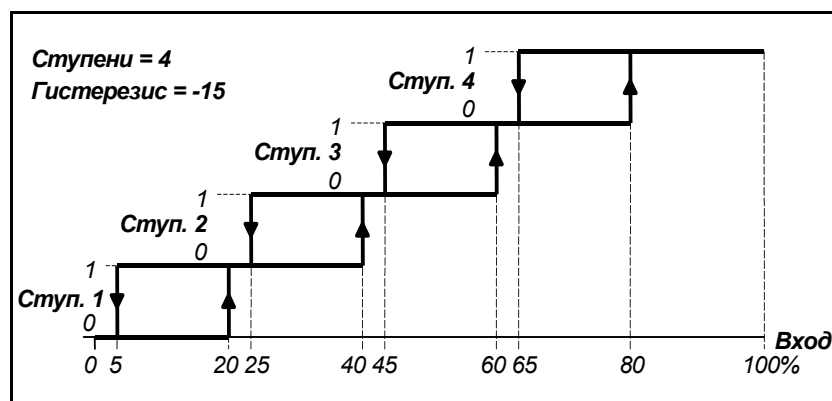
$$N = \frac{(M + 1) \cdot \text{Вход}}{100}$$

Вход – значение между 0% и 100%. Ступени M представляют собой первые M-биты от целого значения выходного сигнала блока. Первая величина N из данных битов будет равна 1 (ИСТИНА), а все остальные – 0 (ЛОЖЬ).

Пример: Если параметр *Ступени* задан равным 4, а *Гистерезис* и *Поворот* не используются, произойдет активация первой ступени (Выход = 1; 0001) при 20% сигнала на *Входе*, второй ступени (Выход = 3; 0011) при 40%, третьей ступени (Выход = 7; 0111) при 60% и четвертой ступени (Выход = 15; 1111) при 80%. Обратите внимание, что в ТАС Menta заданные целые значения выводятся в дополнительном двоичном коде, т.е. сигнал на выходе для активации 16 ступеней будет -1 (1111111111111111).

Если параметр *Гистерезис* равен нулю, то приведенная выше формула позволит Вам рассчитать число ступеней, активизированных входящим сигналом. Если *Гистерезис* имеет ненулевое значение, петля гистерезиса будет находится слева или справа (в зависимости от отрицательного или положительного значения *Гистерезиса*) от точек, полученных при помощи приведенной формулы. В подобных случаях, значения активизированных ступеней будут деактивироваться в обратном порядке. Например, при расчете 4-ступенчатой последовательности активация ступеней согласно вышеприведенной формуле произойдет на одной ступени при 20%, на двух ступенях - при 40%, трех ступенях - 60% и на четырех ступенях - 80%. Если гистерезис имеет положительное значение, например, равен 10, активация ступеней будет смещена вправо на 10%, т.е. произойдет при 30%, 50%, 70% и 90% соответственно. При этом деактивация будет выполняться по первоначальным значениям.

С другой стороны, если гистерезис имеет отрицательное значение, например, -15%, деактивация сместится влево и, таким образом, ступени будут активизироваться при 20%, 40%, 60% и 80% , а деактивироваться при 5%, 25%, 45% и 65% соответственно.



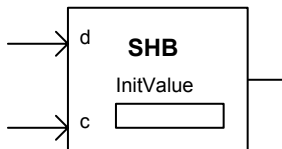
Параметр *Период* используется для предотвращения одновременной активации двух или более ступеней. Если значение периода не равно нулю, ступени будут всегда активизироваться последовательно с интервалом, соответствующим значению *Периода* (миллисекунды), даже при условии значительного изменения входящего сигнала, способного вызывать одновременную активацию ступеней. Однако, если значение периода равно нулю, существует вероятность одновременной активации ступеней.

Параметр *Поворот* определяет, должны ли поворачиваться активные ступени. Отличие между *Поворотом* = 1 (с поворотом) и *Поворотом* = 0 (без поворота) состоит в том, что в первом случае ступени будут деактивироваться в том же порядке, в каком они были активированы. Таким образом, ступень, которая была активизирована в течении более длительного времени, будет всегда деактивироваться в первую очередь. Если значение

Поворота равно 0, порядок деактивации ступеней будет прямо противоположным порядку активации. В том случае, если последовательность задается с поворотом, можно предположить, что постепенно период времени, в течение которого каждая ступень остается активной, станет практически одинаковым для всех ступеней.

2.47. SHB – Регистр сдвигающий вправо SHB

Значение бинарного сигнала

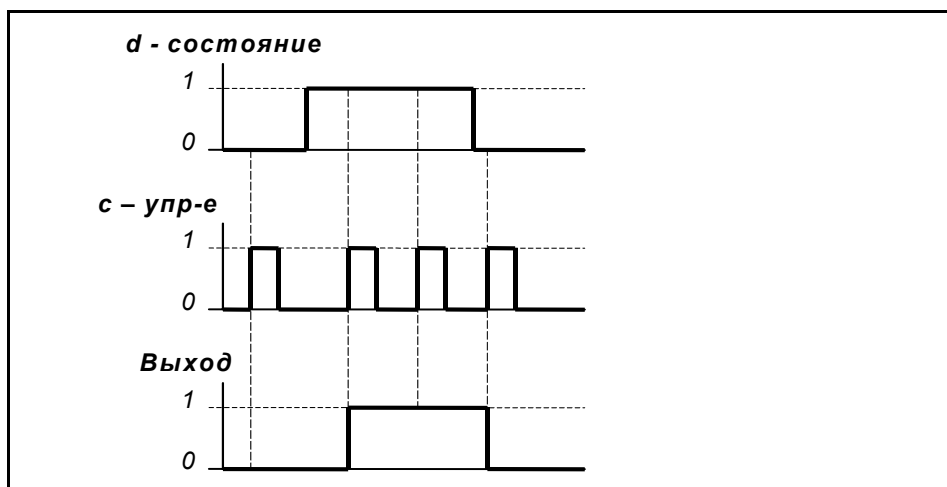


| | | | |
|-----------------|--------------------|----------|--|
| ВХОДЫ | состояние (d) | БИНАРНЫЙ | Входящий бинарный сигнал. |
| | управление (c) | БИНАРНЫЙ | Управляющий сигнал. |
| ПАРАМЕТРЫ | Начальное значение | БИНАРНЫЙ | Начальное значение управляющего сигнала. |
| ТИП ВЫХОДА | | ДОПУСК | |
| БИНАРНЫЙ СИГНАЛ | | R/W | |

Описание: Функция регистра, сдвигающего вправо, выполняется согласно нижеприведенной таблице истинности:

| состояние (t) | управление (t) | выход (t+1) |
|---------------|----------------|-------------|
| 0 | 0 | выход (t) |
| 1 | 0 | выход (t) |
| 0 | 1 | 0 |
| 1 | 1 | 1 |

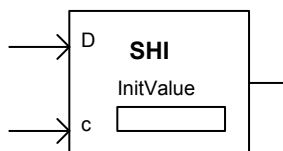
Данный блок копирует состояние входа на выход при активном управляющем сигнале. В случае, если управляющий сигнал неактивен, состояние выхода не меняется.



2.48. SHI - Регистр сдвигающий вправо

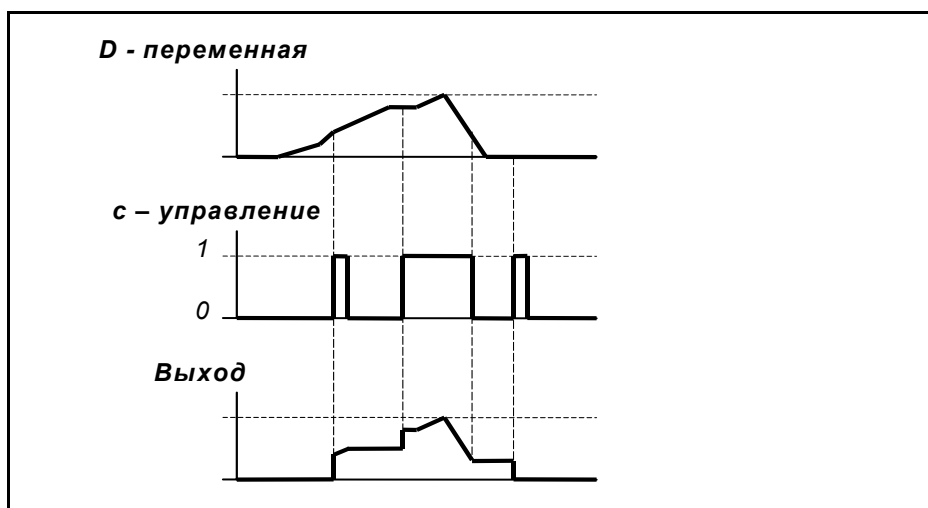
SHI

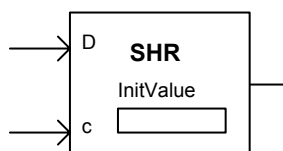
Значение целого сигнала



| | | | |
|--------------|--------------------|----------|-------------------------------|
| ВХОДЫ | переменная (D) | ЦЕЛЫЙ | Аналоговый входящий сигнал. |
| | управление (c) | БИНАРНЫЙ | Управляющий сигнал. |
| ПАРАМЕТРЫ | Начальное значение | ЦЕЛЫЙ | Начальное значение на выходе. |
| ТИП ВЫХОДА | | ДОПУСК | |
| ЦЕЛЫЙ СИГНАЛ | | R/W | |

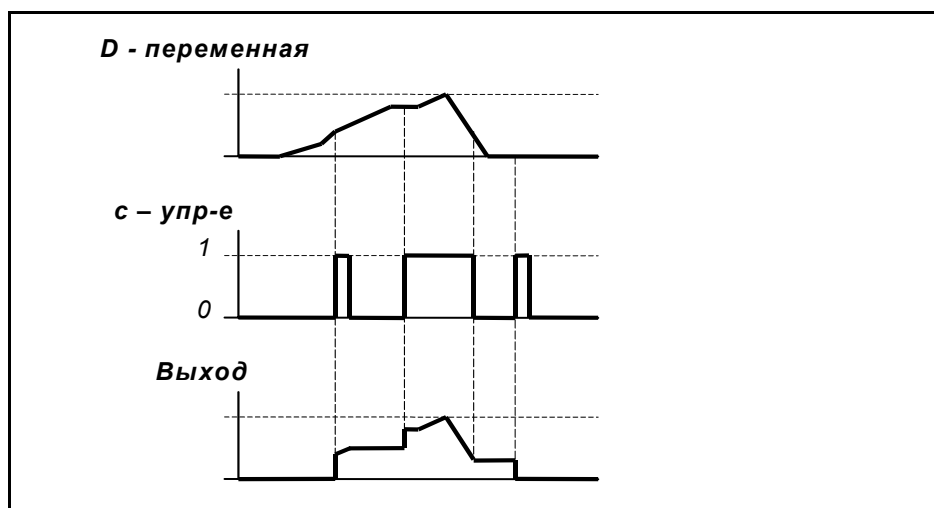
Описание: Данный блок функционирует подобно блоку DELI в ситуации, при которой управляющий сигнал на входе остается активным. Данный блок выполняет функцию задержки целого сигнала на 1 программный цикл. Значение сигнала на выходе обновляется в каждом программном цикле с учетом значения входящего сигнала из предыдущего цикла.



2.49. SHR - Регистр сдвигающий вправо**SHR****Значение дробного сигнала**

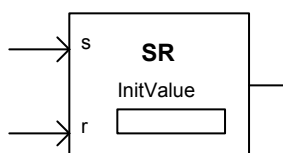
| | | | |
|----------------|--------------------|----------|-------------------------------|
| ВХОДЫ | переменная (D) | ДРОБНЫЙ | Аналоговый входящий сигнал. |
| | управление (c) | БИНАРНЫЙ | Управляющий сигнал. |
| ПАРАМЕТРЫ | Начальное значение | ДРОБНЫЙ | Начальное значение на выходе. |
| ТИП ВЫХОДА | | ДОПУСК | |
| ДРОБНЫЙ СИГНАЛ | | R/W | |

Описание: Данный блок функционирует подобно блоку DELR в ситуации, при которой управляющий сигнал на входе остается активным. Данный блок выполняет функцию задержки дробного сигнала на 1 программный цикл. Значение сигнала на выходе обновляется в каждом программном цикле с учетом значения входящего сигнала из предыдущего цикла..



2.50. SR – PC Триггер

SR

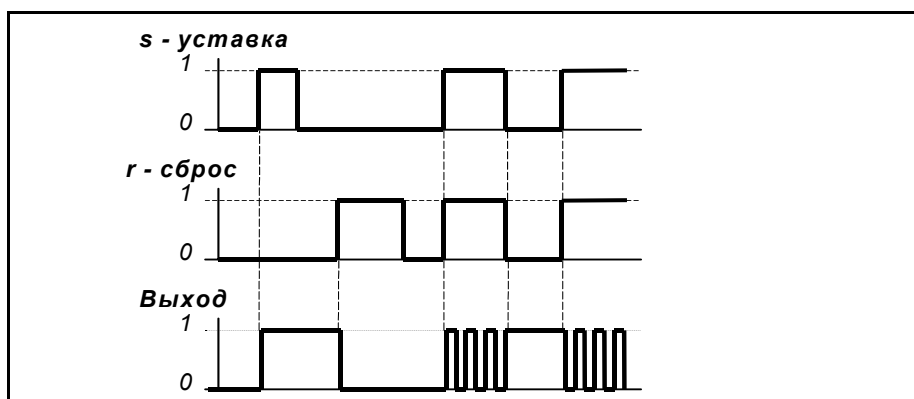


| | | | |
|-----------------|--------------------|----------|---------------------------------------|
| ВХОДЫ | Уставка (s) | БИНАРНЫЙ | Активация выходящего сигнала. |
| | Сброс (r) | БИНАРНЫЙ | Деактивация выходящего сигнала. |
| ПАРАМЕТРЫ | Начальное значение | БИНАРНЫЙ | Начальное значение сигнала на выходе. |
| ТИП ВЫХОДА | | ДОПУСК | |
| БИНАРНЫЙ СИГНАЛ | | R/W | |

Описание: PC Триггер представляет собой бистабильный блок с двумя входами: *уставка* и *сброс*. В зависимости от значения сигнала на данных входах на заданный момент времени, для следующего программного цикла выход блока получит значение в соответствии с данными, приведенными в таблице:

| уставка (t) | сброс (t) | выход (t+1) |
|-------------|-----------|-----------------|
| 0 | 0 | выход (t) |
| 0 | 1 | 0 |
| 1 | 0 | 1 |
| 1 | 1 | нет (выход (t)) |

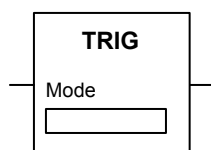
Сигнал на выходе в момент времени t+1 равен сигналу на входе в момент t, при условии, что два входа неактивны. При активных входах выход меняет свое состояние в каждом цикле программы. Если активен только один из входов, то выход будет активироваться или деактивироваться в зависимости от того, какая из переменных входа активна, *уставка* или *сброс*.



2.51. ТСУС – Время цикла**ТСУС**

| | | |
|----------------|--------|--|
| ТИП ВЫХОДА | ДОПУСК | |
| ДРОБНЫЙ СИГНАЛ | RO | |

Описание: Данный блок выдает продолжительность одного программного цикла (в секундах), согласно таблице спецификации.

2.52. TRIG - Триггер**TRIG**

| | | | |
|-----------------|-----------|----------|-----------------------|
| ВХОДЫ | Состояние | БИНАРНЫЙ | Сигнал триггера |
| ПАРАМЕТРЫ | Режим | ЦЕЛЫЙ | Режим работы 1,2 or 3 |
| ТИП ВЫХОДА | ДОПУСК | | |
| БИНАРНЫЙ СИГНАЛ | RO | | |

Описание: Триггер (переключатель) является моностабильным и меняет состояние при наличии изменения во входном сигнале. Данный блок генерирует импульсный сигнал, который по продолжительности равен времени одного программного цикла. В зависимости от выбранного *режима* работы триггер может менять состояние следующим образом:

- **Режим 0 и 1:** при изменении от низкого к высокому.
- **Режим 2:** при изменении от высокого к низкому.
- **Режим 3:** при изменении любого типа.

Режим может вводиться как целое число от 0 до 255, однако любой режим больше 3 будет работать как режим 3, т.е. триггер будет менять состояние при изменении любого типа.

2.53. TSCH – Расписание**TSCH**

| | | | |
|--------------|---------------------------|--------|--------------------------|
| ПАРАМЕТРЫ | Недельный график, Мах. | ЦЕЛЫЙ | Число недельных графиков |
| | График выходных, Мах. | ЦЕЛЫЙ | Число графиков выходных |
| ТИП ВЫХОДА | | ДОПУСК | |
| ЦЕЛЫЙ СИГНАЛ | | RO | |

Описание: Блок расписания времени используется для построения недельных графиков, которые необходимы для запуска или остановки, например, блока АНУ в разное время в зависимости от определенного дня недели. При помощи данного блока также можно задавать временные интервалы (праздники), во время которых график работы изменяется с нормального режима на альтернативный.

Блок TSCH имеет выход для Целого сигнала с положительным/отрицательным значением. Отрицательный сигнал указывает на количество времени (в мин.), оставшееся до конца заданного в расписании интервала времени, а положительный сигнал – до его начала.

Обратите внимание, что метка *Backup* используется только для состояния блока на выходе, т.е. в течение времени, оставшегося до следующего состояния. Точки уставки недельных графиков и графиков выходных **всегда** сохраняются во Flash-памяти контроллера, независимо от наличия метки *Backup*.

Количество недельных графиков в расписании задается при помощи параметра *Недельный график, Мах.* Необходимость в определении всех нужных графиков во время прикладного программирования отсутствует. Параметр *Недельный график, Мах.* также позволяет определять количество недельных графиков для использования в панели оператора ТАС Xenta OP. Дополнительные недельные графики могут выбираться при помощи панели оператора непосредственно во время работы.

Количество графиков выходных задается при помощи параметра *График выходных, Мах.* Необходимость в определении всех нужных графиков во время прикладного программирования отсутствует. Параметр *График выходных, Мах.* также позволяет определять количество графиков для использования в панели оператора ТАС Xenta OP. Дополнительные графики выходных могут выбираться при помощи панели оператора непосредственно во время работы.

Общее количество недельных графиков и графиков выходных ограничено размером памяти контроллера ТАС Xenta. Поскольку каждый из таких графиков занимает достаточно много места в памяти контроллера, прежде, чем увеличить значение параметра **Max.**, необходимо четко определить цели использования блока расписания времени.

Графики выходным можно задавать в ТАС Vista, путем использования функции центрального расписания времени.

Пример:

Предположим, что блок АНУ имеет следующие часы работы:

08:00 - 12:00 и 13:00 - 17:30 с понедельника по пятницу

09:00 - 14:00 в субботу

10:00 - 12:00 в воскресенье

В канун рождества время работы блока АНУ должно быть с 15:00 до 16:00, если рождество приходится на понедельник, вторник, среду, четверг и пятницу.

Также необходимо иметь возможность задавать 2 дополнительных выходных дня при помощи панели оператора ТАС Xenta OP.

Подобное расписание будет выглядеть следующим образом:

Недельный график, Max. = 4:

| Время старта | Время остановки | Дни недели |
|--------------|-----------------|--------------------|
| 08:00 | 12:00 | Пн, Вт, Ср, Чт, Пт |
| 13:00 | 17:30 | Пн, Вт, Ср, Чт, Пт |
| 09:00 | 14:00 | Сб |
| 10:00 | 12:00 | Вс |

График выходных, Max. = 3:

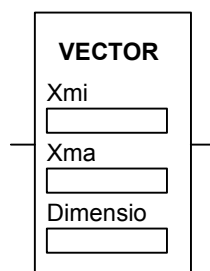
| Дата старта | Дата остановки | Время старта | Время остановки | Дни недели |
|-------------|----------------|--------------|-----------------|--------------------|
| *-12-24 | *-12-24 | 15:00 | 16:00 | Пн, Вт, Ср, Чт, Пн |
| | | | | |
| | | | | |

Примечание 1: Символ «звездочка» в колонках «дата старта» и «дата остановки» обозначает возможность задавать определенный выходной день каждый год. Такое возможно только для графиков выходных.

Примечание 2: Пользователю разрешается вводить время 24:00. Часы работы 00:00-24:00 обозначают, что регулируемое оборудование будет выполнять свои функции в течение полных 24 часов.

Примечание 3: Пользователю разрешается вводить временной интервал 23:00-04:00 Пн, что означает, что регулируемое оборудование будет работать с 23:00 до 24:00 в заданный день недели (Пн), но продолжит работу с 00:00 до 04:00 на следующий день (Вт).

Примечание 4: нулевой интервал 00:00-00:00 может быть задан для графика выходных с целью обеспечить отключение регулируемого оборудования на полные 24 часа. Время работы, введенное в виде 03:34-03:34 будет считаться эквивалентным 00:00-00:00, т.е. регулируемое оборудование будет отключено.

2.54. VECTOR - Векторы**VECTOR**

| ВХОДЫ | Вход | ДРОБНЫЙ | Входящий сигнал. |
|----------------|----------------|---------|--|
| ПАРАМЕТРЫ | Xmin | ДРОБНЫЙ | Нижняя граница диапазона входящего сигнала. |
| | Xmax | ДРОБНЫЙ | Верхняя граница диапазона входящего сигнала. |
| | Размеры (Y(X)) | ДРОБНЫЙ | Список значений линейной функции (от 2 до 255). Каждое значение в списке должно вводиться отдельной строкой. |
| ТИП ВЫХОДА | | ДОПУСК | |
| ДРОБНЫЙ СИГНАЛ | | RO | |

Описание: Блок VECTOR позволяет определить любой отрезок линейной функции для входящего сигнала, ограниченный параметрами X_{max} и X_{min} . Функция определяется путем указания значения функции на выходе $y = f(x)$ для N числа значений (x), полученных делением границ входного числа на равные отрезки. Для значения на входе между двумя точками значение функции рассчитывается при помощи линейной интерполяции. Для входного сигнала, меньше X_{min} , функция определяется значением, заданным для первой точки интервала. Если входящий сигнал больше X_{max} , функция будет определяться значением, заданным для последней точки интервала.

Обратите внимание, что данный блок может использоваться как ранжированная матрица Дробных значений для просмотра таблицы, подобной той, что приведена ниже:

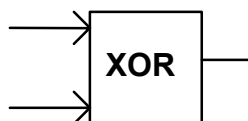
| Вход | Выход |
|------|-------|
| 5 | 34,5 |
| 6 | 28,0 |
| 7 | 42,5 |
| 8 | 33,9 |

Для осуществления данной функции, значение на входе должно всегда быть Целым числом X_{min} и X_{max} (в таблице: $X_{min} = 5$, $X_{max} = 8$), а число элементов в списке должно равняться $X_{max} - X_{min} + 1$ (в таблице: $8 - 5 + 1 = 4$).

2.55. WDAY – Текущий день недели**WDAY**

| | | |
|--------------|--------|--|
| ТИП ВЫХОДА | ДОПУСК | |
| ЦЕЛЫЙ СИГНАЛ | RO | |

Описание: Данный блок показывает текущий день недели, согласно внутренним часам. Значение 1 на входе соответствует понедельнику, 7 – воскресенье.

2.56. XOR – Инверсия или XOR-функция**XOR**

| | | | |
|-----------------|------------|----------|--|
| ВХОДЫ | состояние1 | БИНАРНЫЙ | |
| | состояние2 | БИНАРНЫЙ | |
| ТИП ВЫХОДА | ДОПУСК | | |
| БИНАРНЫЙ СИГНАЛ | RO | | |

Описание: Данный блок используется для определения булевой функции отрицания или XOR-функции на 2 бинарных сигнала: *состояние 1* и *состояние2*, в соответствии со следующей таблицей истинности:

| состояние1 | состояние2 | выход |
|------------|------------|-------|
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |

3. Формулы



Блок формул является специфичным и содержит всего один параметр, арифметическую формулу, которая может быть как простой, так и сложной. В зависимости от формулы блок имеет разное кол-во входов. Графическое изображение блока будет менять свой размер, что обусловлено величиной формулы и/или количеством входов.

Блок формул имеет только один выходной сигнал. Тип блока формул зависит от типа сигнала на выходе, который может быть Дробным, Целым или Бинарным. Таким образом, в таблице представлены три возможных типа данного блока:

| <i>Сокращение</i> | <i>Краткое описание</i> | <i>Коммент.</i> |
|-------------------|------------------------------|-----------------|
| XPB | Блок формул. Бинарный выход. | |
| XPI | Блок формул. Целый выход. | |
| XPR | Блок формул. Дробный выход. | |

Блоки формул, как правило, доступны только в режиме просмотра. Размер памяти, которую занимает подобный блок, зависит от степени сложности формулы.

• Операнды

В формулах могут использоваться следующие операнды:

Входящие переменные

Входящие переменные, обозначенные в формулах заглавными буквами (**A**, **B**, **C**, ...) относятся к Аналоговому входу, в то время как строчные буквы (**a**, **b**, **c**, ...) относятся к переменным Бинарного входа. В одной и той же формуле запрещается использование одной и той же буквы в качестве прописной и строчной, например, "A" и "a". Входящие переменные приводятся с левой стороны блока формул в алфавитном порядке.

Числовые константы Константы относятся к Целой части, которая может вводиться знаками + или -. За целой частью может следовать запятая (,)/точка (.), отделяющая целую часть от десятичной дроби. Также может присутствовать часть показательной функции, вводимая буквами *e* или **E**, за которыми следует целое двузначное число.

Буквенно-цифровые конст. Константы данного типа представляют собой буквенно-цифровые идентификаторы 20 значимых символов, заданных в таблице констант. Имя константы должно быть заключено в кавычки ("), если в нем присутствуют знаки / или :. Константы **Public** в блоках формул *не* используются.

• Операторы

В блоках формул операнды сочетаются с **операторами**. Описание последних приведено ниже, **в порядке старшинства**.

| | | |
|-------------------------------|-----------|--|
| Единичные операторы | - , ! | изменение знака, логическое отрицание |
| Бинарные операторы | * , / , % | умножение, деление и модуль |
| | + , - | сложение, вычитание |
| | << , >> | сдвиг влево, сдвиг вправо |
| | < , > | меньше, больше |
| | <= , >= | меньше/равно, больше/равно |
| | = , != | равно, не равно |
| | & | логическая «И»-функция (побитовая) |
| | ^ | логическое отрицание или XOR-функция (побитовая) |
| | | логическая «ИЛИ»-функция (побитовая) |
| Троичные операторы | ? : | выражение «ЕСЛИ-ТО-ИНАЧЕ». "a ? b : c" означает "если a то b иначе c". |
| Арифметические функции | $x^{**}y$ | x в степени y |
| | LN (x) | Натуральный логарифм |
| | LOG (x) | Base 10 логарифм |
| | EXP (x) | Показательная функция e в степени x |
| | COS (x) | Косинус x (радиан), для всех x. |
| | SIN (x) | Синус x (радиан), для всех x. |
| | TAN (x) | Тангенс x (радиан), для всех x. |
| | ACOS (x) | Арккосинус x, где $-1 < x < 1$, результат между 0 и $\pi/2$ радианами. |
| | ASIN (x) | Арксинус x, где $-1 < x < 1$, результат между $-\pi/2$ и $\pi/2$ радианами. |
| | ATAN (x) | Арктангенс x, для всех x, результат между $-\pi/2$ и $\pi/2$ радианами. |
| | SQRT (x) | Квадратный корень из x |

| | |
|----------------|----------------------------------|
| ABS (x) | Абсолютное значение x |
| INT (x) | Перевод в целое число (усечение) |

В формулах могут использоваться круглые скобки для обозначения порядка вычисления, который отличается от порядка расположения операторов по старшинству.

Три типа переменных могут без проблем сочетаться с оператором, поскольку преобразования необходимых типов выполняются автоматически, согласно описываемым ниже правилам:

- Для выполнения операций $+$, $-$, $*$ и $/$ все значения преобразуются в Дробные.
- Для выполнения операций $\%$, $<<$, $>>$, $\&$, \wedge и $|$ все значения преобразуются в целые (Дробные значения округляются).
- Для выполнения операции **!** Дробные или Целые значения преобразуются в Бинарные согласно следующему правилу: если значение равно нулю, то оно преобразуется в ноль. Если значение не равно нулю, то оно преобразуется в то значение, которое предшествовало операции логического отрицания. "Истинная" **побитовая операция отрицания** целого числа A может быть осуществлена путем $A \wedge (-1)$, т.е. A XOR (-1).
- Операции $>$, $<$, $>=$, $<=$, $=$, $!=$ преобразуют операнды в Дробные значения, выполняют сравнение и возвращают Бинарный результат.

• Выход

Блок формул может иметь следующие типы выходящего сигнала:

- XRV блок** **Бинарный** выход. Если результат равен нулю, сигнал на выходе блока будет равен нулю. Если результат не равен нулю, сигнал на выходе блока будет равен единице.
- XPI блок** Результат преобразован в 16-битное **Целое** число.
- XPR блок** Сигнал на выходе имеет **Дробное** значение, полученное в результате вычислений по формуле.

4. Операторы

Блок формул может быть представлен *графически* в виде различных **блоков операторов**, связанных между собой. Существует несколько типов операторов, идентичных тем, которые используются в блоках формул. Получаемый результат полностью эквивалентен: каждый оператор привязан к определенному блоку формул, так, как будто он был бы записан в блок формул. Выходящие сигналы операторов могут свободно считываться пользователем.

Аналоговые входы операторов могут соединяться с Дробными или Целыми выходами. Аналоговые выходы подсоединяются к Дробным или Целым входам, не делая между ними различий. При соединении выхода Аналогового оператора с функциональным блоком, имеющим Целый вход, выход оператора преобразуется в Целый. Таким образом, в случае подключения другого функционального блока с Дробными входами к такому же выходу оператору, входы приобретут Целое значение. Бинарные входы могут соединяться только с Бинарными выходами.

Существующие операторы приведены в следующих таблицах. Для их презентации используются символы стандартного графического режима. Все операторы разделены на логические группы подобно простым блокам.

• Константы

| Название | Комментарии |
|--------------------|----------------------------|
| Бинарная константа | Нет входа → Бинарный выход |
| Целая константа | Нет входа → Целый выход |
| Дробная константа | Нет входа → Дробный выход |

Константы **Public** не используются в качестве операторов.

• Логические операторы

| Название | Комментарии |
|----------|-------------------------------|
| NOT | Бинарный вход → Бинар. выход |
| AND | Бинарные входы → Бинар. выход |
| OR | Бинарные входы → Бинар. выход |
| XOR | Бинарные входы → Бинар. выход |

• Математические операторы

| Название | Комментарии |
|-----------|-------------------------------|
| Отрицание | Аналог. вход → Аналог. выход |
| Сложение | Аналог. входы → Аналог. выход |
| Вычитание | Аналог. входы → Аналог. выход |
| Умножение | Аналог. входы → Аналог. выход |
| Деление | Аналог. входы → Аналог. выход |
| Модуль | Аналог. входы → Аналог. выход |

• Сопоставление


| Название | Комментарии |
|------------------|------------------------------|
| Больше | Аналог. входы → Бинар. выход |
| Меньше | Аналог. входы → Бинар. выход |
| Равенство | Аналог. входы → Бинар. выход |
| Неравенство | Аналог. входы → Бинар. выход |
| Больше или равно | Аналог. входы → Бинар. выход |
| Меньше или равно | Аналог. входы → Бинар. выход |

• Битовая операция

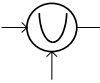
| Название | Комментарии |
|--------------|-------------------------------|
| bit AND | Аналог. входы → Аналог. выход |
| bit OR | Аналог. входы → Аналог. выход |
| bit XOR | Аналог. входы → Аналог. выход |
| Сдвиг вправо | Аналог. входы → Аналог. выход |
| Сдвиг влево | Аналог. входы → Аналог. выход |

Примеры

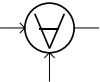
Оператор bit AND

| | |
|---|---------------------|
|  | FEDC BA98 7654 3210 |
| | 0000 1001 1001 0010 |
| | 0010 0100 0101 0000 |
| | 0000 0000 0001 0000 |

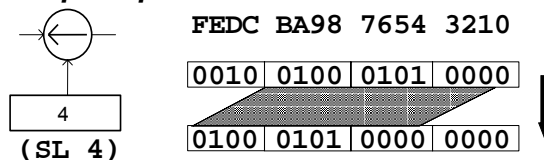
Оператор bit OR

| | |
|---|---------------------|
|  | FEDC BA98 7654 3210 |
| | 0000 1001 1001 0010 |
| | 0010 0100 0101 0000 |
| | 0010 1101 1101 0010 |

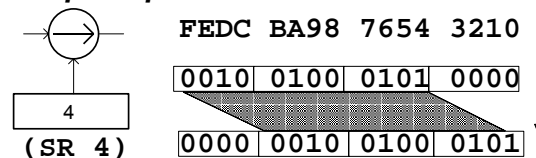
Оператор bit XOR

| | |
|---|---------------------|
|  | FEDC BA98 7654 3210 |
| | 0010 0100 0101 0000 |
| | 0000 1001 1001 0010 |
| | 0010 1101 1100 0010 |

Оператор <<



Оператор >>



Обратите внимание, что все сдвигаются все 16 позиций. 0 сдвигается в пустые позиции. Любой число сдвигов ≥ 16 обуславливает значение на выходе, равное 0.

• Другое

| Название | Комментарии |
|-----------------------------------|-------------------------------|
| D/A преобразователь | Бинар. вход → Аналог. выход |
| A/D преобразователь | Аналог. вход → Бинар. выход |
| Мультиплексор Аналоговых сигналов | Аналог. входы → Аналог. выход |
| Мультиплексор Бинарных сигналов | Бинар. входы → Бинар. выход |
| Преобразование AA | Целый сигнал ↔ Дробный сигнал |

В блоке формул операнды разных типов могут комбинироваться посредством любого оператора. Однако, при использовании блоков операторов, преобразования должны выполняться эксплицитно. Операторы D/A и A/D используются исключительно для преобразования Бинарного сигнала в Аналоговый и наоборот. Эти операторы необходимы, т.к. остальные операторы имеют маркированные типы входов. Соответственно, без D/A и A/D, например, было бы невозможно вычислить сумму Бинарного и Аналогового сигнала. В блоках формул подобные преобразования производятся неявно, т.е. данные операторы не включаются в формулу. Существуют следующие правила выполнения преобразований:

- Дробное или Целое число приобретает в Бинарное значение путем присвоения логического 0, если значение на выходе равно 0, или 1, если значение на выходе не равно нулю..
- Бинарный сигнал преобразуется в Аналоговый следующим образом: логический 0 присваивает число 0, а логическая 1 присваивает число 1.

Операторы-мультиплексоры представляют собой переключатели, которые выбирают один из двух входящих сигналов в зависимости от значения Бинарного переключателя.

Оператор «ПреобразованиеАА» используется при соединении целого выхода с дробным входом и наоборот. Поскольку операторы не делают различий между Дробными и Целыми значениями, оператор «ПреобразованиеАА» не выполняет каких-либо явных преобразований с округлением или усечением. Подобные типы преобразований производятся в блоке путем использования выходящего сигнала «ПреобразованиеАА» в качестве входящего. В случае возникновения необходимости в выполнении преобразования явного типа с округлением или усечением рекомендуется использовать блоки формул.

5. Тестовые блоки

5.1. Краткое описание

В некоторых случаях использование программ приложения для тестирования и отладки упрощается при закрытом контуре управления с обратной связью. В этом случае может использоваться вспомогательная модель, имитирующая работу системы, например, системы вентиляции. Такая модель программируется при помощи функциональных блоков и включается в приложение FBD. Для того, чтобы сделать программу приложения независимой от вспомогательной модели, необходимы специальные блоки, которые способны считывать состояния физических выходов и записывать значения, рассчитанные моделью системы для физических входов. Такие специальные блоки называются *тестовыми*. Тестовый блок относится к определенному блоку входа/выхода (Ю), поскольку выполняет тестовые функции, аналогичные функциям блока Ю.

Обратите внимание, что тестовые блоки используются исключительно для имитации работы системы. Программа тестового блока не способна генерировать код для последующей загрузки ее в контроллер. Таким образом, любая вспомогательная модель, входящая в состав программы приложения, должна быть удалена прежде, чем будет генерироваться исполняемый код.

Существует четыре типа тестовых блоков, соответствующих четырем типам физических сигналов I/O:

| <i>Сокращение</i> | <i>Краткое описание</i> |
|-------------------|---------------------------------------|
| ТРАО | Тестовый блок для Аналогового выхода. |
| ТРДО | Тестовый блок для Цифрового выхода. |
| ТРАІ | Тестовый блок для Аналогового входа. |
| ТРДІ | Тестовый блок для Цифрового входа. |

5.2. ТРАІ – Тестовый блок для Аналог. входа ТРАІ



| | | | |
|-----------------------|------|--|-----------------|
| ВХОДЫ | Вход | ДРОБНЫЙ | Входящий сигнал |
| ТИП ВЫХОДА | | ДОПУСК | |
| Блок не имеет выходов | | Блок используется только для имитации. | |

Описание: Блок ТРАІ записывает значение на блок «Аналоговый вход» под тем же именем (идентификатором), как и в тестовом блоке. Обратите внимание, что блок ТРАІ может использоваться со всеми типами блоков «Аналоговый вход». Значение блока «Аналоговый вход» записывается на соответствующий блок ТРАІ, т.е. без дополнительного пересчета.

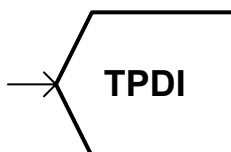
5.3. ТРАО - Тестовый блок для Аналог. выхода ТРАО



| | | | |
|----------------|--|--|-----------------------|
| ВХОДЫ | | | Блок не имеет входов. |
| ТИП ВЫХОДА | | ДОПУСК | |
| ДРОБНЫЙ СИГНАЛ | | Блок используется только для имитации. | |

Описание: Блок ТРАО считывает значение с блока «Аналоговый выход» под тем же именем (идентификатором), как и в тестовом блоке.. Выход блока ТРАО имеет то же значение, что и вход блока «Аналоговый выход».

5.4. TPDI - Тестовый блок для Цифр. входа TPDI



| | | | |
|-----------------------|------|--|-----------------|
| ВХОДЫ | Вход | БИНАРНЫЙ | Входящий сигнал |
| ТИП ВЫХОДА | | ДОПУСК | |
| Блок не имеет выходов | | Блок используется только для имитации. | |

Описание: Блок TPDI записывает значение *Вход* на блок «Цифровой вход» под тем же именем (идентификатором), как и в тестовом блоке.

TPDI также может соединяться с блоком CNT. При этом будет происходить подсчет только одного импульса на один цикл программы приложения. Подсчет импульсов происходит при изменении значения входного сигнала от 1 к 0.

5.5. TPDO - Тестовый блок для Цифр. выхода TPDO



| | | | |
|-----------------|--|--|-----------------------|
| ВХОДЫ | | | Блок не имеет входов. |
| ТИП ВЫХОДА | | ДОПУСК | |
| БИНАРНЫЙ СИГНАЛ | | Блок используется только для имитации. | |

Описание: Блок TPDO считывает состояние блока «Цифровой выход» под тем же именем (идентификатором), как и в тестовом блоке.

TPDO также может соединяться с блоком DOPU. При этом параметр *Мин. Импульс* блока DOPU задается равным времени исполнения цикла программы приложения (сек.), в противном случае Вы получите ошибочные значения.

6. Рекомендации по программированию

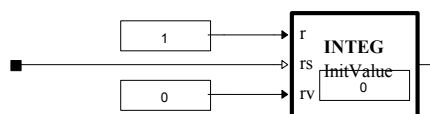
• **Время программного цикла**

Время цикла программы приложения изменяемо. Не забывайте об этом в процессе создания программы приложения.

Пример: Блок INTEG может использоваться для расчета потребляемой Энергии исходя из входной Мощности. При сбросе всех значений блока в начале нового интервала вычислений, величина начала должна быть равна счетчику цикла!

• **Счетчик времени**

Простой в использовании счетчик времени, считающий секунды, отвечает за реальное время цикла программы приложения. Сброс счетчика производится Бинарным сигналом..



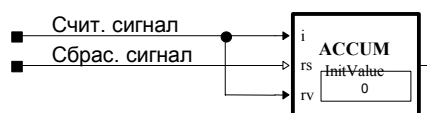
Счетчики времени могут быть созданы посредством блоков PULSE и RT. Только построения из блоков RT и INTEG могут сбрасывать значения при подсчете. Счетчик INTEG используется, в основном, при подсчете (времени, мощности и т.д.) за определенный период.

• **Равенство**

Если при тестировании два значения равны, убедитесь, что это значения Бинарного или Целого типа. Не выполняйте тест на тождественность с Дробными значениями.

• **Счетчик сброса**

Например, сброс счетчика энергии без потери импульсов во время сброса:

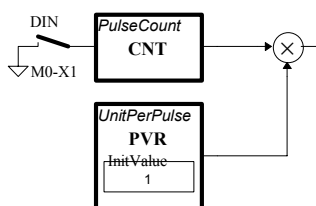


• Побитовые логические операторы

Выход побитового оператора AND (&) имеет Целое значение, что является результатом «И»-операции с двумя Аналоговыми сигналами. Выполнив данную операцию в блоке формул с Бинарным выходом, Вы получите нулевое значение на Бинарном выходе при отсутствии совпадения, и единицу при совпадении, как минимум, 1 бита у двух Аналоговых сигналов.

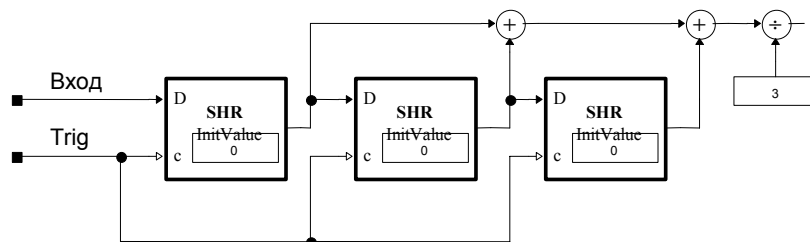
• Параметр Множителя в блоке CNT

Внутренний параметр множителя в блоке не относится к константам «Public», т.к. является параметром конфигурации. При возникновении необходимости в изменении значения данного параметра с панели оператора, потребуется специальная программа, отвечающая за подсчет другими блоками количества импульсов, поступающих к оборудованию.



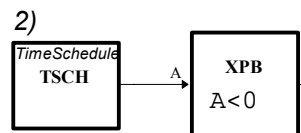
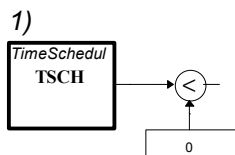
• Скользящее среднее значение

Пример программы со скользящим средним значением трех величин:



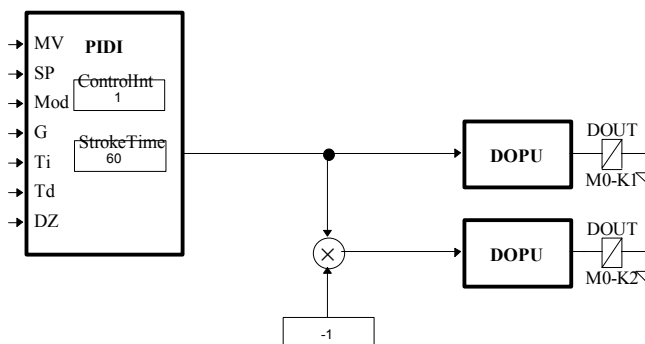
• Выход TSCH

Выход блока «Расписание» имеет целое значение и показывает, сколько минут осталось до следующего изменения состоянием блоком. Существует несколько возможностей для получения Бинарного сигнала:



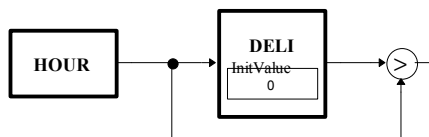
• PIDI - DOPU

Стандартные функции блоков PIDI и DOPU;



• Смена суток

Смена суток произошла, если значение нового часа меньше, чем полученное за предыдущий час.

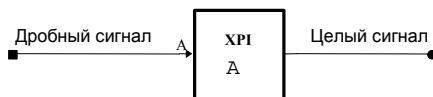


• Блоки формул

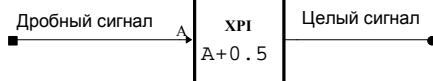
Если выход блока формул соединен с узлом, то тип выхода не изменяется.

Блоки формул могут использоваться для преобразования Дробного сигнала в Целый и наоборот:

Преобр. Дробного сигнала в Целый с усечением

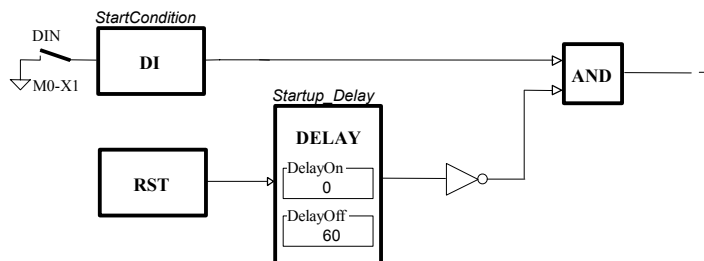


Преобр. Дробного сигнала в Целый без усечения



• Задержка включения

Блок RST используется для осуществления задержки включения после теплого старта контроллера TAC Xenta. Задайте разные периоды задержки для разных АНУ, например, путем изменения времени **Задержки выключения** (60 сек в примере):



Приложение 1 Используемые SNVTs

TAC Menta v3.4 поддерживает следующие типы SNVT:

| <i>SNVT</i> | <i>№.</i> | <i>Комментарии</i> |
|--------------------|-----------|--|
| SNVT amp | 1 | Ток: -3,276.8 .. 3,276.7 A (0.1 A) |
| SNVT amp_f | 48 | Ток: -1E38 .. 1E38 A |
| SNVT amp_mil | 2 | Ток: -3,276.8 .. 3,276.7 mA (0.1 mA) |
| SNVT angle | 3 | Угол: 0 .. 65.535 rad (0.001 rad) |
| SNVT angle_deg | 104 | Угол: -359.98 .. +360.00° (0.02°) |
| SNVT count | 8 | Счетчик: 0 .. 65,535 |
| SNVT count_f | 51 | Счетчик: -1E38 .. 1E38 |
| SNVT count_inc | 9 | Счетчик: -32,768 .. 32,767 |
| SNVT count_inc_f | 52 | Счетчик: -1E38 .. 1E38 |
| SNVT_elapsed_tm | 87 | <i>Структурированная SNVT. Структурные члены:</i> день, час, минута, сек, мсек |
| SNVT elec_kwh | 13 | Электроэнергия: 0 .. 65,535 kWh (1 kWh) |
| SNVT elec_whr_f | 68 | Электроэнергия: 0 .. 1E38 Wh |
| SNVT flow | 15 | расход: 0 .. 65,534 l/s (1 l/s) |
| SNVT flow_f | 53 | Расход: -1E38 .. 1E38 l/s |
| SNVT freq_f | 75 | Частота: -1E38 .. 1E38 Hz |
| SNVT freq_hz | 76 | Частота: 0 .. 6553.5 Hz (0.1 Hz) |
| SNVT freq_kilohz | 77 | Частота: 0 .. 6553.5 kHz (0.1 kHz) |
| SNVT_hvac_emerg | 103 | 0 = EMERG_NORMAL 1 = EMERG_PRESSURIZE 2 = EMERG_DEPRESSURIZE 3 = EMERG_PURGE 4 = EMERG_SHUTDOWN |
| SNVT_hvac_mode | 108 | 0 = HVAC_AUTO 1 = HVAC_HEAT 2 = HVAC_MRNG_WRMUP 3 = HVAC_COOL 4 = HVAC_NIGHT_PURGE 5 = HVAC_PRE_COOL 6 = HVAC_OFF 7 = HVAC_TEST 8 = HVAC_EMERG_HEAT 9 = HVAC_FAN_ONLY |
| SNVT_hvac_override | 111 | <i>Структурированная SNVT: состояние, %, расход</i> <i>Справедливые значения для состояния:</i> 0 = HVO_OFF 1 = HVO_POSITION 2 = HVO_FLOW_VALUE 3 = HVO_FLOW_PERCENT 4 = HVO_OPEN 5 = HVO_CLOSE 6 = HVO_MINIMUM 7 = HVO_MAXIMUM |
| SNVT_hvac_status | 112 | <i>Структур. SNVT. Структ. члены:</i> mode, heat o p, |

| | | |
|------------------|-----|--|
| | | heat o s, cool out, econ out, fan out, in alarm |
| SNVT lev cont | 21 | Уровень: 0 .. 100 % (0.5%) |
| SNVT lev cont f | 55 | Уровень: 0 .. 100 % |
| SNVT_lev_disc | 22 | 0 = ST_OFF 1 = ST_LOW 2 = ST_MED 3 = ST_HIGH 4 = ST_ON |
| SNVT lev percent | 81 | Уровень: -163.84% .. 163.83% (0.005%) |
| SNVT lux | 79 | Освещение: 0 .. 65,535 lux (1 lux) |
| SNVT multiplier | 82 | 0 .. 32.7675 (0.0005) |
| SNVT obj req | 92 | Структур. SNVT. Струк. члены: id, request |
| SNVT_obj_status | 93 | Структур. SNVT. Струк. члены: obj_id, byte1, byte2, byte3, reserved2 |
| SNVT_occupancy | 109 | 0 = OC_OCCUPIED 1 = OC_UNOCCUPIED 2 = OC_BYPASS 3 = OC_STANDBY |
| SNVT power | 27 | Мощность: 0 .. 6,553.5 W (0.1 W) |
| SNVT power f | 57 | Мощность: -1E38 .. 1E38 W |
| SNVT power kilo | 28 | Мощность: 0 .. 6,553.5 kW (0.1 kW) |
| SNVT ppm | 29 | Концентрация: 0 .. 65,535 ppm (1 ppm) |
| SNVT ppm f | 58 | Концентрация: 0 .. 1E38 ppm |
| SNVT_preset2w | 94 | Структур. SNVT. Струк. члены: learn, selector, valueH, valueL, день, час, мин, сек, мсек |
| SNVT press | 30 | Давление: -3,276.8 .. 3,276.7 kPa (0.1 kPa) |
| SNVT press f | 59 | Давление: -1E38 .. 1E38 Pa |
| SNVT press p | 113 | Давление: -32,768 .. 32,766 Pa (1 Pa) |
| SNVT pwr fact | 98 | -1.0 .. 1.0 (0.00005) |
| SNVT_reg_val | 136 | Структур. SNVT: raw ms, raw ls, unit, misc |
| SNVT_reg_val_ts | 137 | Структур. SNVT. Струк. члены: raw_ms, raw_ls, unit, misc, год, месяц, день, час, мин, сек |
| SNVT rpm | 102 | Скорость: 0 .. 65,534 rpm (1 rpm) |
| SNVT scene | 115 | Структур. SNVT: function, scene number |
| SNVT_scene_cfg | 116 | Структур. SNVT. Струк. члены: function, scene_number, setting, rotation, fade_time, delay_time, priority |
| SNVT speed | 34 | Скорость: 0 .. 6,553.5 m/s (0.1 m/s) |
| SNVT speed f | 62 | Скорость: -1E38 .. 1E38 m/s |
| SNVT state | 83 | Состояние: 16 bits |
| SNVT setting | 117 | Структур. SNVT: сост., уставка, угол |
| SNVT switch | 95 | Структур. SNVT. Струк. члены: знач., состояние |
| SNVT temp | 39 | Температура: -274 .. 6,279.5 °C (0.1 °C) |
| SNVT temp f | 63 | Температура: -273.17 .. 1E38 °C |
| SNVT temp p | 105 | Температура: -273.17 .. +327.66 °C (0.01 °C) |
| SNVT_temp_setpt | 106 | Структур. SNVT. Струк. члены: occupied_cool, standby_cool, unoccupied_cool, occupied_heat, standby_heat, unoccupied_heat |
| SNVT time f | 64 | Время: -1E38 .. 1E38 s |
| SNVT time sec | 107 | Время: 0.0 .. 6553.4 s (0.1 s) |
| SNVT_time_stamp | 84 | Структур. SNVT. Струк. члены: год, месяц, день, час, мин, сек |
| SNVT vol | 41 | Объем: 0 .. 6,553.5 l (0.1 l) |
| SNVT vol f | 65 | Объем: 0 .. 1E38 l |
| SNVT vol kilo | 42 | Объем: 0 .. 6,553.5 kl (0.1 kl) |
| SNVT volt | 44 | Объем: -3,276.8 .. 3,276.7 V (0.1 V) |
| SNVT volt f | 66 | Объем: -1E38 .. 1E38 V |
| SNVT zerospan | 85 | Структур. SNVT. Струк. члены: ноль, диапазон |



ООО «ТАС»
129626 Россия, Москва,
Ул. Новоалексеевская 13/1,
Тел.: (095) 937 40 88,
Факс: (095) 937 40 85

ПАРТНЕРЫ
Петроспек
(095) 965 07 80
Спецмонтажэлектро
(095) 232 21 11

ТБН Энергосервис
(095) 978 61 01
HCM Engineering
(Киев, Украина)
(38 044) 238 64 94

Петроспек-Холдинг
(Санкт-Петербург)
(812) 102 08 80
Блисе (Самара)
(8462) 92 63 80

Климат-Проф
(Санкт-Петербург)
(812) 327 12 00

www.tac-russia.ru

www.tac-global.com



Компания ТАС создает все необходимые условия для улучшения микроклимата в помещении, что позволяет современному человеку, проводящему до 90 процентов своего времени в офисе, квартире или собственном доме, чувствовать себя комфортно. Благодаря новой концепции деятельности ТАС – «Открытые системы для интеллектуальных зданий» (Open Systems for Building IT™), клиенты, сотрудничающие с нашей компанией, получают такие преимущества, как сокращение расхода электроэнергии, свобода выбора системы и возможность ее изменения, надежность в работе и индивидуальный подход к каждому заказчику. Компания ТАС, владельцем которой является крупная инвестиционная компания EQT, имеет дочерние отделения и партнеров более чем в 70 странах мира. В ТАС работают около 2 000 сотрудников, и в этом году полный оборот компании должен составить более 300 миллионов долларов США. Главный офис ТАС, координирующий деятельность в европейских странах, расположен в г. Мальме, Швеция. Региональное представительство ТАС, ответственное за работу на территории Южной и Северной Америки находится в Далласе, США, а офис ТАС в г. Перт, Австралия, отвечает за азиатско-австралийский регион.

