

TAC Vista®



Разработка программ
в TAC Menta



Предисловие

Перед Вами руководство "Разработка программ в TAC Menta для TAC Vista IV".

Если Вы обнаружили ошибки или неясные описания в этом руководстве, пожалуйста, свяжитесь с Вашим представителем TAC. Вы так же можете воспользоваться электронной почтой info@tac-russia.ru.

Копирайт © 2003 TAC AB. Все права защищены.

Этот документ, также как и продукция, к которой он относится, предназначен только для лицензированных пользователей. TAC AB имеет авторское право на этот документ и мохраняет за собой право производить в нем изменения, добавления или сокращения. TAC AB не несет ответственности за возможные ошибки, которые могли появиться в этом документе.

Не используйте продукт для других целей кроме тех, которые указаны в этом документе.

Только лицензированным пользователям продукта и документа разрешается использовать документ или любую информацию из него. Распространение, раскрытие, копирование, сохранение или использование изделия, информации или иллюстрации, помещенных в документе, нелицензионными потребителями, в электронной или механической форм, как запись или другим способом, включая фотокопирование или системы хранения поиска информации, без специального письменного разрешения TAC AB строго запрещены и буду расценены как нарушение законов об авторском праве.

Торговые марки и зарегистрированные торговые марки являются собственностью их соответствующих владельцев. Microsoft® и Windows® - зарегистрированные торговые марки Microsoft Corporation.

Редакции

Арт. номер.	Комментарии	Редактор	Дата
0-004-7843-0	Новое руководство	ANBL/ POOL	28 июня 2002
0-004-7843-1	Исключение замечаний по автоматической генерации меню панели оператора для Xenta 280. Документация по использованию ед.измерения системы I-P.	KSD	03 декабря 2002
0-004-7843-2	В главе 8.3 Пункт меню Опции - Конфигурация устройства, добавлено "Использование LonMark 3.3" для Xenta 283. Добавлен блок PIDP регулятора в главе В Функциональные блоки (В.36).	KSD	03 декабря 2002
0-004-7485-0 (RU)	Перевод руководства	UP	Июнь 2004

Разработка программ в TAC Menta

Subject to modification.

© 2004 TAC AB

Содержание

ВВЕДЕНИЕ	13
1 Введение	15
1.1 Структура	15
1.2 Условные обозначения	16
ОСНОВНОЙ РАЗДЕЛ	17
2 Введение в TAC Menta	19
2.1 Терминология	19
2.2 TAC Menta, система блоков	20
2.3 Режимы	20
2.4 TAC Menta Программные лицензии	21
2.5 TAC Menta Setup	21
2.6 Запуск TAC Menta	22
3 Основные положения	23
3.1 Функциональные блоки диаграмм (FBD)	23
3.2 Сигналы	24
3.2.1 Типы сигналов	24
3.2.2 Названия сигналов	24
3.2.3 Общедоступные сигналы (Public)	25
3.3 Соединения	25
3.4 Функциональные блоки	27
3.4.1 Простые блоки (Simple Block...)	27
3.4.2 Блоки формул (Expression...)	27
3.4.3 Тестовые блоки (Test Probe Block)	29
3.4.4 Группы функциональных блоков	29
3.5 Константы	29

3.5.1	Названия констант	29
3.5.2	Общедоступные константы	30
3.6	Внешние сигналы I/O ввода/вывода	32
3.6.1	I/O блоки	32
3.6.2	Общедоступные (Public) параметры	36
3.7	Модули (Module)	37
3.8	Иерархические функциональные блоки(HFB)	38
3.9	Правила доступа	39
3.9.1	Сигналы	39
3.9.2	Константы	39
3.10	Расписания (Time Schedules)	39
3.11	Текст аварийных сообщений	40
3.12	Локальный Тренд Лог (Trend Logs...) в TAC Xenta	40
3.13	Файлы данных	40
4	Мышь и функциональные клавиши	43
4.1	Использование мыши	43
4.2	Использование функциональных клавиш	44
5	Режим редактирования (Edit)	47
5.1	Опции меню	47
5.1.1	Пункты меню	47
5.1.2	Всплывающие меню	58
5.2	Функциональные блоки	65
5.2.1	Создание нового блока	65
5.2.2	Редактирование блока	68
5.2.3	Перемещение блока	70
5.2.4	Удаление блока	70
5.2.5	Отсоединить блок	70
5.3	Соединения	72
5.3.1	Рисование соединений	72
5.3.2	Удаление соединения	76
5.3.3	Перемещение узла	76
5.3.4	Разрыв соединения	76
5.3.5	Подсвечивание соединений	77
5.3.6	Ортогональные соединения	78
5.4	Операции с группами	78
5.4.1	Выделение группы	78
5.4.2	Снятие выделения группы	79
5.4.3	Перемещение группы	79
5.4.4	Разъединение группы	79
5.4.5	Удаление группы	81
5.4.6	Ввод или правка названия модуля	81
5.4.7	Копирование и вставка группы	81
5.4.8	Копировать выделенную область в буфер	81
5.4.9	Центрирование выборки	81
5.4.10	Печать выборки	81

5.4.11	Макро-команды в блоках комментариев	82
5.4.12	Сохранение макро-блока	82
5.4.13	Загрузка макроблока	83
5.5	Иерархические функциональные блоки (HFB)	86
5.5.1	Создание HFB	86
5.5.2	Название HFB и соединений	86
5.5.3	Раскрытие и свертывание блока HFB	87
5.5.4	Вывод на печать HFB	87
5.6	Таблица констант	87
5.6.1	Добавление константы	87
5.6.2	Редактирование констант	89
5.6.3	Удаление константы	89
5.7	Таблица конфигурации I/O	89
5.8	Таблица расписаний	90
5.9	Таблица аварийных сообщений	90
5.10	Локальный ТрендЛог в TAC Xenta	90
5.10.1	Выбор Тренд лога (Select Trend Log)	91
5.10.2	Определение тренд лога (Trend Log Definition)	92
5.11	Установка даты и времени (Set Date and Time)	95
5.12	Поиск и замена текста	96
5.13	Отмена (Undo)	97
6	Режим Имитации (Simulate)	99
6.1	Компиляция FBD	99
6.2	Меню Опции	100
6.2.1	Строка меню	100
6.2.2	Всплывающие меню	106
6.3	Выполнение прикладной программы	107
6.3.1	Старт, стоп и инициализация выполнения	108
6.3.2	Операции с сигналами	108
6.3.3	Выделение соединений	109
6.3.4	Режим моделирования исполняемого файла	109
6.4	Имитация внешних входов	109
6.4.1	Ручной режим управления	110
6.4.2	Автоматическая генерация вектора входных значений	110
6.4.3	Имитация с использованием тестовых блоков	112
6.5	Тренд логи	112
6.5.1	Добавление сигналов в регистратор	113
6.5.2	Удаление сигналов из регистратора	113
6.5.3	Очистка регистратора	113
6.5.4	Рестарт регистратора	113
6.5.5	Сброс регистратора	113
6.5.6	Просмотр тренд логов	113
6.5.7	Задание временных выборок	114
6.6	Средства регистрации	115
6.6.1	Старт регистрации	115
6.6.2	Настройка графиков	115

6.6.3	Представление и анализ регистрируемых величин	116
6.6.4	Сохранение и просмотр Лог файла	116
6.6.5	Очистка регистратора	117
6.7	Изменение параметров блока в процессе имитации	118
6.7.1	Параметры функциональных блоков	118
6.7.2	Временные расписания	118
6.7.3	Данные связывания I/O	118
6.7.4	Константы	118
6.7.5	Аварийные сообщения	118
6.7.6	Блок ошибок ERR	119
7	Программа конфигурации панели оператора	121
7.1	Экран структуры меню	122
7.2	Создание структуры меню	123
7.2.1	Добавление пунктов меню	123
7.2.2	Подменю (Submenu)	124
7.2.3	Статус (Status)	124
7.2.4	Авария (Alarm)	126
7.2.5	Код доступа (Access Code)	126
7.2.6	Дата и время (Date and Time)/Летнее время (Daylight Saving)	127
7.2.7	Недельный/Годовой таймер (Week/Holiday Chart)	127
7.2.8	Сервисное меню TAC	127
7.3	Редактирование существующего дерева структуры меню	128
7.3.1	Перемещение пунктов меню	128
7.3.2	Изменение разметки эрана панели оператора	128
7.3.3	Копирование и вставка	128
7.4	Автоматическая генерация дерева меню панели оператора	129
7.5	Файл описания панели оператора	130
7.5.1	Данные и описания синтаксиса в файлах описания панели оператора (.DOP)	131
7.5.2	Импорт файла описания панели оператора	133
7.5.3	Экспорт файла описания панели оператора	134
7.6	Определение файла набора символов	134
7.7	Меню опции	135
7.7.1	Строка меню	135
8	Другие функции и инструменты	141
8.1	Распечатка документации прикладной программы	141
8.2	Связанные текстовые файлы	142
8.3	Конфигурация устройства	143
8.4	Программная спецификация	145
8.5	Генерация исполняемого кода	147
8.6	Вычисление использования памяти	148
8.7	Функции На-линии	149
8.7.1	Режим На-линии (Online)	150
8.8	Загрузка прикладной программы	151
8.9	Загрузка данных приложения из контроллера	153

8.10	Принудительное переопределение сигналов физических входов/выходов (Override of Physical I/O Signals)	153
8.11	Изменение параметров блоков в режиме На-линии	155
8.12	Изменение общедоступных констант в режиме На-линии	155
8.13	Изменение параметров "связывания" (Bind...) в режиме На-линии	155
8.14	Обозначение стандартных приложений/контроллеров	155
8.15	Уникальный ID прикладной программы	156
9	Мастер Загрузки	157
9.1	Как использовать Мастер загрузки (Download Wizard)	158
9.1.1	Диалог	158
9.1.2	Общая процедура загрузки	159
9.2	Совместимость TAC Menta	160
9.2.1	.AUT файлы	160
9.2.2	.COD файлы	161
9.2.3	Файлы дерева меню панели оператора	161
9.3	Обновление устройства TAC Xenta 300 до версии v3	161
9.3.1	Обновление .AUT файла с данными приложения (дополнительно)	161
9.3.2	Обновление системных и прикладных программ	162
9.3.3	Проверка корректности выполнения (дополнительно, но рекомендуется)	163
10	Сообщения об ошибках	165
10.1	Системные ошибки	165
10.2	Компиляция FBD	165
10.3	Сохранение и загрузка прикладной программы в базу данных	167
10.4	Имитация	168
10.5	Генерация кода	169
10.6	Загрузка	170
10.6.1	Коды ошибок	171
10.7	Связь TAC Xenta	172
	СПРАВОЧНЫЙ РАЗДЕЛ	173
A	Введение	175
A.1	Общие сведения	175
A.2	Терминология	176
A.3	Типы сигналов	176
B	Функциональные блоки	177
B.1	Обзор	178
B.1.1	Блоки I/O (Входов/Выходов)	178
B.1.2	Источники сигнала	178
B.1.3	Логические функции	179
B.1.4	Нелинейные функции	179
B.1.5	Блоки задержки	179

V.1.6	Регуляторы и фильтры	180
V.1.7	Сумматоры	180
V.1.8	Системные параметры	180
V.1.9	Расписание и сигнал тревоги	181
V.1.10	Передаточные функции	181
V.2	ACCUM - Сумматор	182
V.3	ANYST - Аналоговый гистерезис	183
V.4	AI - Аналоговый вход	185
V.4.1	Линейный аналоговый вход (Linear Analog Input)	186
V.4.2	Нелинейный аналоговый (термисторный) вход (Non-linear Analog Input)	189
V.4.3	Сетевая переменная (Network Variable)	191
V.4.4	SNVT	193
V.4.5	Постоянное значение (Constant Value)	195
V.5	ALARM - Авария	196
V.6	AND - Логическая функция "И"	199
V.7	АО - Аналоговый выход	200
V.7.1	Физический выход	200
V.7.2	SNVT	202
V.7.3	Не подсоединен (Not connected)	203
V.8	CNT - Цифровой вход - Счетчик импульсов	204
V.8.1	Счетчик импульсов	204
V.8.2	Не подсоединен (Not connected)	205
V.9	CURVE - Линейный график	206
V.10	DATE - Дата	209
V.11	DELAY - Задержка включения/выключения	210
V.12	DELB - Задержка на 1 цикл (бинарный сигнал)	212
V.13	DELI - Задержка на 1 цикл (целый сигнал)	213
V.14	DELR - Задержка на 1 цикл (дробный сигнал)	214
V.15	DI - Цифровой вход	215
V.15.1	Физический цифровой вход	215
V.15.2	Сетевая переменная	216
V.15.3	SNVT	218
V.15.4	Устройство На-линии (Online Device)	219
V.15.5	Постоянное значение (Constant Value)	220
V.16	DO - Цифровой выход	221
V.16.1	Физический цифровой выход (Physical Output)	221
V.16.2	SNVT	222
V.16.3	Не подсоединен (Not connected)	223
V.17	DOPU - Цифровой импульсный выход	224
V.17.1	Цифровой импульсный выход	224
V.17.2	Не подсоединен (Not connected)	225
V.18	ENTH - Расчет энтальпии	226
V.19	ERR - Системные ошибки	230
V.20	FILT - Фильтр первого порядка	232
V.21	HOURL - Текущий час	233
V.22	HYST - Бинарный гистерезис	234

V.23	INTEG - Интегратор	236
V.24	LIMIT - Ограничение по min/max	237
V.25	MAX - Селектор максимального из 2 сигналов	238
V.26	MIN - Селектор минимального из 2 сигналов	239
V.27	MINUTE - Текущая минута	240
V.28	MONTH - Текущий месяц	241
V.29	NCYC - Счетчик программного цикла	242
V.30	NOT - Инверсия бинарного сигнала	243
V.31	OPT - Оптимизатор	244
V.32	OR - логическая функция "ИЛИ" для 2 бинарных сигналов	253
V.33	OSC - Осциллятор	254
V.34	PIDA - PID регулятор - Аналоговый выход	255
V.35	PIDI - PID регулятор - Увеличить/Уменьшить	259
V.36	PIDP - PIDP регулятор - Аналоговый выход	262
V.37	POLY - Полиномы	266
V.38	PRCNT - Процентное соотношение	268
V.39	PULSE - Импульсный генератор	269
V.40	PVB - Бинарная переменная	270
V.41	PVI - Целая переменная	271
V.41.1	PVI - XENTASYSREG	271
V.42	PVR - Дробная переменная	273
V.43	RAMP - Ограничитель скорости изменения сигнала	274
V.44	RST - Рестарт	275
V.45	RT - Учет времени работы	276
V.46	SECOND - Текущая секунда	277
V.47	SEQ - Последовательное управление	278
V.48	SHB - Регистр, сдвигающий вправо, бинарный сигнал	281
V.49	SHI - Регистр, сдвигающий вправо, целый сигнал	282
V.50	SHR - Регистр, сдвигающий вправо, дробный сигнал	283
V.51	SR - PC Триггер	284
V.52	TCYC - Время цикла	285
V.53	TRIG - Триггер	286
V.54	TSCH - Расписание	287
V.55	VECTOR - Векторная функция	290
V.56	WDAY - Текущий день недели	292
V.57	XOR - Инверсия или XOR-функция	293
C	Формулы (Expressions)	295
D	Операторы (Operators)	301
E	Тестовые блоки (Test Probe Blocks)	305
E.1	Краткое описание	305

E.2	ТРАИ - Тестовый блок для Аналог. входа	306
E.3	ТРАО - Тестовый блок для Аналог. выхода	306
E.4	ТПДИ - Тестовый блок для Цифрового входа	307
E.5	ТПДО - Тестовый блок для Цифрового выхода	307
F	Рекомендации по программированию	309
G	Поддерживаемые SNVT	313
G.1	Список используемых SNVT	313
	Алфавитный указатель	317

ВВЕДЕНИЕ

Глава

1 Введение

1 *Введение*

Это руководство рассматривает основные ситуации. Для более детальной информации рекомендуется обращаться к руководствам пользователя, поставляемым с конкретным оборудованием.

Для получения информации по инсталляции программного обеспечения рекомендуется обращаться к инструкциям, поставляемым с программным обеспечением.

1.1 Структура

Руководство делится на три раздела:

- Введение
- Основной раздел
- Справочный раздел

Введение

Раздел Введение содержит информацию о структуре руководства и о наиболее эффективном способе его использования для поиска информации.

Основные положения

Основной раздел содержит подробное описание того, как осуществить проектирование и решение задач, изложенных в этом руководстве.

Справочный раздел

Справочный раздел содержит более подробную информацию о различных частях основного раздела. Он также дает информацию относительно альтернативных решений, не указанных в основном разделе.

При необходимости разделом можно пользоваться по конкретной тематике.

1.2 Условные обозначения

В тексте руководства используются следующие специальные знаки для выделения особо важных участков текста..

**Обратите внимание!**

Используется для выделения некоторой информации.

**Подсказка!**

Используется для выделения действий, облегчающих выполнение операций.

**Предупреждение!**

Используется, когда необходимо предупреждение.

**Предостережения!**

Используется для выделения действий, неточное выполнение которых может вызвать серьезные проблемы.

ОСНОВНОЙ РАЗДЕЛ

Главы

- 2 Введение в ТАС Menta
- 3 Основные положения
- 4 Мышь и функциональные клавиши
- 5 Редактирование
- 6 Имитация
- 7 Инструмент конфигурирования ПО
- 8 Другие инструменты и функции
- 9 Мастер загрузки
- 10 Сообщения об ошибках

2 Введение в TAC Menta

TAC Menta это графический язык программирования, разработанный для контроллеров серии TAC Xenta.

В этом документе использование рассмотренных инструментов приведено с точки зрения пользователя. Предполагается, что пользователь имеет основные знания об оборудовании TAC Xenta и о языке его программирования и поэтому основы здесь не будут освещены. Все функции доступные в языке программирования TAC Xenta описаны в *Справочном разделе данного руководства*.

2.1 Терминология

Холодный старт	Рестарт после пропадания питания, длившегося более чем 72 часа, т.е. когда содержание памяти RAM в TAC Xenta стерто.
FBD	Функциональные блоки диаграмм.
Инструмент конфигурации OP	Инструмент программирования для TAC Xenta панели оператора, включенный в TAC Menta.
OP	Панель оператора для TAC Xenta контроллеров.
SNVT	Стандартные сетевые переменные, позволяющие производить в LonWorks™ сети общение между узлами различных производителей.
TAC Menta	Инструмент создания программы приложения для контроллеров TAC Xenta.
TAC Xenta	Серия специализированных контроллеров (TAC Xenta 3000) и свободно программируемых контроллеров (TAC Xenta 300/401) с блоками расширения входов/выходов и контроллеров с ограниченными возможностями (TAC Xenta 280) без блоков расширения входов/выходов.
Горячий старт	Рестарт после пропадания питания, длившегося менее 72 часов, т.е. когда содержимое RAM-памяти не стерто.
TAC Vista	Инструмент для представления информации и контроля.

2.2 TAC Menta, система блоков

В TAC Menta, ряд блоков используют систему измерения как СИ, так и I-P.

Локальные установки операционной системы Windows© на ПК, на котором создается прикладная программа, определяют систему единиц измерения. Если в операционной системе используется метрическая система измерения, то в блоках будет использована

СИ система измерения, а если используется Американская система измерений, то в блоках будет использована I-P система измерения.

2.3 Режимы

В TAC Menta есть два режима работы, *Редактирование (Edit)* и *Имитация (Simulate)*:

- В режиме *Редактирования (Edit)* инструментарий позволяет нам разрабатывать прикладную программу работая с FBD (функциональные блоки диаграмм) в окне. Режим Редактирования (Edit) подробно описан на [стр. 43. в разделе "Редактирование"](#). В режиме имитации (Simulate) правильно разработанная прикладная программа может быть запущена на выполнение, при этом каждый такт программы будут отображаться значения сигналов в FBD-окне. В тоже время, возможно графически отображать тренд-логи не менее, чем 6 бинарных и/или аналоговых сигналов. У пользователя так же есть возможность выбрать вид физических входных сигналов.
- В *режиме имитации (Simulate)* возможно работать в режиме не-на-линии (Offline) (нет подсоединенных устройств TAC Xenta) и в режиме на-линии (Online) (устройства TAC Xenta подключены к ПК). имитация подробно описана в разделе ["Имитация" на стр. 73.](#)

В обоих режимах можно войти в инструмент конфигурации панели оператора (OP Configuration) (описанном в разделе ["Инструмент конфигурации панели оператора" на стр. 89](#)) для разработки текста и меню панели оператора.

2.4 TAC Menta Программные лицензии

Лицензирование

TAC Menta поставляется с лицензией, подключаемой через Privilege License Server, используемый с TAC Vista IV. Для получения информации об установке TAC Menta и Privilege License Server, пожалуйста, смотрите руководство по инсталляции для TAC Vista IV.

Демо режим

TAC Menta может быть загружена в демо режиме без лицензии. Демо режим позволяет пользователю опробовать все функции программы (кроме сохранения, имитации и редактирования). Тем не менее приложение может быть сохранено как демо файл. Это означает, что будет не возможно загрузить приложение в контроллер. Демо приложение может быть открыто в лицензионной версии TAC Menta, но не может быть сконвертировано в пригодный для загрузки в контроллер формат.



Обратите внимание!

Приложение, сохраненное в версии TAC Menta 4.0 (или более поздней версии) не может быть использовано TAC Vista версии менее чем версия 3.2. Для организации поддержки старых версий TAC Vista, есть возможность приложения, созданные TAC Menta 4.0 сохранять в формате TAC Menta 3.1.

2.5 TAC Menta Setup

TAC Menta Setup - программный модуль для осуществления настройки связи Вашей системы:

Com Порт	Коммуникационный порт ПК.
BPS	Бит в секунду (скорость передачи), всегда должна быть 9600.
Create Com Log	Флаг отладки коммуникации (0= нет записи лога, 1= запись лога). Если ComLog=1 тогда все коммуникации в/из TAC Menta записываются в файл с именем pl_com1.txt (если используется коммуникационный порт 1) в директории "C:\Windows\Temp". (Убедитесь, что данная директория существует на Вашем жестком диске).
	Обратите внимание! Эта опция только для внутренних средств отладки. Лог-файл может стать очень большим.
Time Out, Packet Size, Resends, Com Pause	Коммуникационные параметры, не должны изменяться.

2.6 Запуск TAC Menta

TAC Menta запускается из группы программ TAC Menta, которая содержит иконки программ этой группы. Различные программы могут быть запущены одновременно. Также более чем одна программа может быть запущена, если доступен коммуникационный порт. Запустите программу нажатием на иконке TAC Menta. Перед началом работы с TAC Menta выполните необходимые установки для вашей системы в меню **Свойства - Установки (Preferences - Settings...)** в основной программе TAC Menta.

Каждый FBD-файл может иметь связанный с ним текстовый файл. Активируете текстовый редактор для его использования.

Вы можете использовать библиотеку готовых макро блоков (Load Macro...). Для этого укажите путь, где будет располагаться библиотека.

TAC Menta также может быть запущена из TAC Vista Explorer, и из TAC Vista Device Plug-In, путем наведения курсора на какое-либо устройство, нажатия на нем правой кнопки мыши и выбора в появившемся меню пункта Редактирование (Edit).

3 Основные положения

Программа приложения определяется ее графическим представлением, функциональным блоком диаграмм (FBD). Два фундаментальных элемента FBD это функциональные блоки (FB), которые обрабатывают данные, и соединения, которые связывают сигналы (данные).

В режиме имитации (Simulate) прикладной программы, в FBD блоках могут отображаться текущие значения сигналов. Вы можете изменять значения сигналов во время выполнения программы, выводить изменение сигналов в виде графиков в поле *Trend Log*, и можете устанавливать значения для *физических входов*.

3.1 Функциональные блоки диаграмм (FBD)

В FBD, каждый **функциональный блок** обрабатывает один или несколько *входных сигналов* и генерирует единственный выходной сигнал. Каждый функциональный блок может иметь один или несколько *параметров*, используемых при генерации выходного сигнала. Параметры могут быть как числовыми значениями, так и идентификаторами, *константами*, представляющими постоянные величины.

Выходной сигнал может быть подсоединен к другим блокам в соответствии со связью, установленной **Соединением**, которое определяет прохождение сигнала во время выполнения программы. Путь сигнала обычно должен идти слева направо, за исключением случая, когда соединение используется для замыкания *обратной связи*.

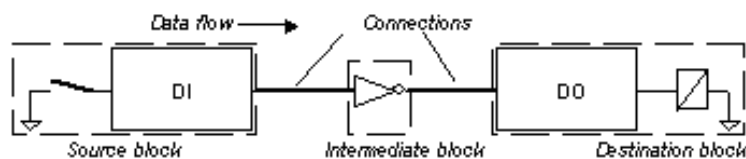


Рисунок 3.1. Простой функциональный блок диаграмм (FBD)

В основном, блоки не имеющие входов, такие как физические входы, (*блоки источники сигналов*) располагаются слева. Блоки, выходы которых не подсоединены к другим блокам (*блоки получатели*), такие как физические выходы, располагаются в правой части. *Промежуточные блоки* располагаются между блоками источниками и блоками получателями, всегда ориентированы в направлении прохождения сигнала.

Программа выполняется циклически. Время цикла является фиксированным. Однократное прохождение цикла называется программным циклом. В каждом программном цикле, изменения

происходящие в блоках источниках передаются в блоки приемники через промежуточные блоки, сигнал следует по связям определенным соединениями и по мере продвижения обрабатывается.

3.2 Сигналы

Сигнал - это базовая единица информации, обрабатываемая прикладной программой, представляющая собой физическую или рассчитанную величину, изменяющуюся во времени. Сигналы изменяются в прикладной программе, которая генерирует значения на выходах функциональных блоков и передает их через соединения на входы других блоков. Все функциональные блоки имеют только один выход, и каждый функциональный блок генерирует свой уникальный сигнал.

3.2.1 Типы сигналов

В FBD соединяемые вход и выход должны быть одинакового типа. Есть три типа данных для сигналов:

Целый (Integer)	16-битное число (диапазон: -32768 to 32767).
Дробный (Real)	действительное 32-битное число в формате IEEE с точностью до 7 знаков (диапазон: 3.4×10^{-38} to 3.4×10^{38}).
Бинарный (Binary)	1-битное число для представления бинарных величин (0/1 = ЛОЖЬ/ИСТИНА). Бинарный тип также называется Булевым или цифровым.

В некоторых случаях мы используем четвертый тип сигнала, так называемый *аналоговый* тип, в независимости от типа, как целые, так и дробные сигналы могут подсоединяться к необходимым входам /выходам.

3.2.2 Названия сигналов

Названия сигналов состоят из двух частей Модуль и Сигнал, разделенных символом "\", например АНУ1\Т1. Модуль и Сигнал представляют собой буквенно-цифровую последовательность. Часть Модуль может включать до 12 символов, но также может быть опущена. Часть Сигнал может включать до 20 символов, такая длина соответствует размеру экрана на панели оператора и максимальной длине строки ID в TAC Vista. Часть Сигнал не может состоять только из цифровых значений. Символы "\", ; , ' - , \$ " используются как разделители или сетевые идентификаторы объектов в TAC Vista, и поэтому не могут применяться в обозначении названия сигнала. Символ пробела нельзя использовать, но символы "/" : "_" разрешены к использованию. Также можно использовать символы русской

раскладки (включая, например, "е д ц з") в названиях сигналов. Название сигнала может быть введено с использованием символа подчеркивание ("Signal_name"), который тем не менее будет автоматически удален. В названиях сигналов нет различия заглавных и прописных символов. Такие названия сигналов считаются идентичными и при компиляции выдадут сообщение об ошибке.

3.2.3 Общедоступные сигналы (*Public*)

Только часть переменных прикладной программы используются другими узлами в сети, среди множества сигналов, применяемых для внутрипрограммных вычислений. Эти сигналы, используемые другими узлами, объявляются как общедоступные сигналы, таким образом позволяя другим узлам в сети читать и, если необходимо, изменять значения таких сигналов. При объявлении всех общедоступных переменных программой приложения формируется программная спецификация, определяющая интерфейс обмена данными с другими узлами в сети.

Общедоступные сигналы в программной спецификации классифицируются как:

DIG	Бинарный (цифровой) сигнал.
ANA	Аналоговый сигнал (целый или дробный).

В зависимости от того могут ли узлы читать и/или изменять значение сигнала, можно также классифицировать сигналы как *только чтение (RO)* или *чтение/запись (RW)*, см. "[Правила доступа](#)" на стр. 38.

3.3 Соединения

Соединение - это связь между выходом одного блока и одним или несколькими входами других блоков. Все соединения типизированы, они наследуют тип выхода, к которому подсоединены. В FBD, бинарное соединение заканчивается незакрашенной стрелкой, а аналоговое (целое, дробное) соединение заканчивается закрашенной стрелкой.

Соединение формируется различными отрезками прямых линий. Точка соединения между отрезками называется узлом. Если узел, объединяет две или более линии, то он отображается как черный кружок, в отличие от места пересечения необъединенных отрезков. Обычно, соединение начинается на выходе блока и заканчивается на входе другого блока. Также возможно наличие неподсоединенных соединений. В этом случае, когда соединение есть только с выходом, то его окончание представляет собой черную точку, а если соединение есть только со входом, окончание представляет черный квадратик. Тем не менее, неподсоединенные

входы не принимаются компилятором TAC Menta, и должны быть подсоединены до запуска режима имитации.

3.4 Функциональные блоки

Функциональные блоки описаны в Справочном разделе этого руководства. Они подразделяются на четыре класса:

- Простые блоки (Simple Block)
- Блоки формул (Expression)
- Операторы (Operator)
- Тестовые блоки (Test Probe Block)

3.4.1 Простые блоки (Simple Block)

Существуют различные типы блоков, каждый из которых выполняет специальные функции. Общее во всех блоках то, что все они имеют фиксированное число входов и параметров. Каждый простой блок генерирует единственный выходной сигнал.

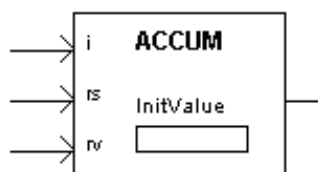


Рисунок 3.2. Простой блок, ACCUM - Сумматор.

Параметры могут быть разных типов, каждый тип допускает определенный диапазон значений, но в основном это числовое значение или ряд числовых значений. Есть два пути определения числа:

Число	Десятичное число в экспоненциальной форме.
Константа	Идентификатор, состоящий из 20 символов, который должен быть определен в <i>Таблице констант (Constants table)</i> (смотри <i>"Константы" на стр. 31</i>).

3.4.2 Блоки формул (Expression)

Блоки формул - это специальные блоки с одним или несколькими входами, арифметическим выражением (может быть составным), но только с одним выходом. Блоки формул имеют столько входов, сколько переменных объявлено в формуле. Графическое

представление блока формул может изменяться в зависимости от размера формулы и количества входов..



Рисунок 3.3. Блок формул.

Входные переменные	<p>Формула может содержать:</p> <p>Одна буква, заглавная ("A") представляет аналоговый сигнал, прописная ("a") соответствует бинарному сигналу. Входные переменные сортируются в алфавитном порядке в левой части блока.</p>
Константы	<p>Могут быть Числовыми: "5", "-13", "0.03", "1.25E12" и т.д. или Буквенно-цифровыми: "PI", "ENERGY" и т.д. Буквенно-цифровые константы - это идентификаторы, состоящие из 20 символов, которые должны быть опеределены в Таблице констант. Константы могут быть введены в кавычках. Общедоступные константы не используются в блоках формул.</p>
Операторы	<p>Операторы такие как "-" (изменение знака), "*" (умножение), "+" (сложение), ">>" (сдвиг вправо), "<" (меньше чем) и "a ? b : c" (ЕСЛИ-ТО-ИНАЧЕ означает: " если a то b иначе c").</p>

Блоки формул разделяются на три типа в зависимости от формируемого выходного сигнала: целый, дробный и бинарный.

Блок формул может быть представлен графически в виде различных блоков операторов, связанных между собой.

Получаемый результат полностью эквивалентен.

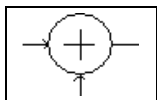


Рисунок 3.4. Оператор, сложение.

Существует несколько типов операторов, идентичных тем, которые используются в блоках формул. **Общедоступные** константы **не** используются в операторах.

Аналоговые входы могут быть соединены с выходами дробного или целого типа, при этом аналоговый выход может быть содинен как с дробным, так и с целым входом. Для перевода целых сигналов в дробные, и аналоговых и бинарные, используются специальные операторы.

Четыре основных логических оператора NOT, AND, OR, XOR, также доступны в виде простых блоков. Логический блок представленный в виде простого блока так и воспринимается компилятором. Но любой логический оператор компилируется как блок формул, поэтому лучше использовать простые блоки вместо операторов для более эффективной работы памяти и процессора. В любом случае различия незначительны, и использование логических простых блоков или логических операторов дело вкуса.

3.4.3 Тестовые блоки (Test Probe Block)

Для имитации прикладной программы, модель контролируемой системы может быть создана при помощи простых функциональных блоков, блоков формул и операторов. Специальные блоки требуются для считывания состояния физических выходов и записи значений, рассчитанных моделью системы для физических входов. Эти тестовые блоки используются только для режима имитации, невозможно *сгенерировать* код прикладной программы, содержащей тестовые блоки.

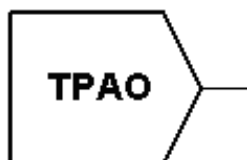


Рисунок 3.5. Тестовый блок, TPAO - Аналоговый выход

Существует четыре типа тестовых блоков, соответствующих четырем типам физических сигналов: **TPAO - Аналоговый выход**, **TPDO - Цифровой выход**, **TPAI - Аналоговый вход** and **TPDI - Цифровой вход**. Все они имеют только один параметр, который определяет название тестируемого входного/выходного блока.

3.4.4 Группы функциональных блоков

Возможно перемещать и копировать целую группу блоков и соединений одновременно, выделив элементы, принадлежащие группе, заключив их в *прямоугольник* (выделяемая область) с помощью мыши.

Группа блоков может быть сохранена как макроблок для использования его в других прикладных программах.

3.5 Константы

TAC Menta позволяет определить идентификаторы, которые представляют постоянные величины для использования их в качестве параметров в блоках вместо числовых значений. Эти идентификаторы называются константами и их значения задаются в *Таблице констант*.

3.5.1 Названия констант

Названия констант (смотри "*Названия сигналов*" на стр. 24) - это буквенно-цифровая последовательность из 20 символов (такая длина соответствует размеру экрана на панели оператора и максимальной длине строки ID в TAC Vista.). Название константы не может состоять только из цифр. Символы " ", ; ' - . \$ " используются как разделители или системные идентификаторы в TAC Vista, и не применяются в названиях констант. Символ

пробел не используется, но символы "/" : "_" используют. Также в названиях допустимо использовать символы русской раскладки (включая, например "е д ц з"). Название константы может быть введено с использованием кавычек ("Constant_name"), который тем не менее будут автоматически удалены.

3.5.2 Общедоступные константы

Константы могут быть определены как *общедоступные (Public)* либо как *внутренние*. После определения, константа не может быть изменена прикладной программой в которой она объявлена, но если она определяется как общедоступная, то она становится доступной другим узлам в сети и может быть изменена, например, с панели оператора. В окне FBD, *общедоступная константа* выделена небольшим кружком "о" справа от поля параметра функционального блока. Название Модуля общедоступной константы, которое совпадает с названием модуля блока, не отображается в FBD (за исключением когда оно отличается от названия Модуля блока).

Рекомендуется всегда давать названия блоков, содержащих общедоступные константы, отличные от названий Модуля, которые константы не могут получить/ изменить.



Обратите внимание!

Важно отметить, что *общедоступные константы могут быть изменены через сеть в процессе выполнения прикладной программы.*

Общедоступные константы могут быть заданы только для одного параметра функционального блока, тем не менее внутренние константы могут быть заданы для любого числа параметров функционального блока. Причина в том, что внутренние названия констант заменяются на соответствующее числовое значение при компиляции. Такое невозможно с общедоступными константами, так как механизм ссылок (указателей) в TAC Menta не позволяет множественные ссылки на одну общедоступную переменную от нескольких параметров функциональных блоков во время выполнения программы.

Общедоступной константе *нельзя* давать тоже название, что и общедоступному сигналу. Также нельзя задавать общедоступную константу, которая *не* используется в каком-либо параметре блока.

Общедоступные константы также включаются в программную спецификацию, но только после компиляции FBD.

Общедоступные константы классифицируются как:

PAB	Бинарная константа
PAI	Целая константа
PAR	Дробная константа

В зависимости от того разрешено другим узлам читать и/или изменять значение константы, их можно классифицировать как *только чтение* или *чтение/запись*, смотри "[Правила доступа](#)" на стр. 40.

3.6 Внешние сигналы I/O ввода/вывода

3.6.1 I/O блоки

Внешние I/O сигналы могут быть получены в FBD посредством четырех *основных I/O блоков* (AI, DI, AO и DO). Основные I/O блоки могут быть связаны как физическим, так и сетевым адресом. Это называется связывание *binding*. Существует два специальных типа I/O блоков, CNT цифрового входа - счетчика импульсов и DOPU для цифрового импульсного выхода. CNT или DOPU блоки *не* могут быть связаны с удаленным сетевым адресом, из-за возможных проблем согласования по времени и синхронизации.

Связывание I/O блоков представлено в *таблице конфигурации I/O*.

Общедоступные сигналы и константы в других устройствах в сети могут *передаваться* через I/O блоки, путем связывания их с ссылками на сетевые адреса, описанные в таблице конфигурации I/O. Типичный случай применения этой функции заключается в использовании несколькими контроллерами совместно сигнала с датчика в общих целях, например, значения датчика температуры наружного воздуха, который физически подсоединен к одному из контроллеров. Передача данных между прикладными программами TAC Xenta организована с помощью блоков (AI и DI) для *чтения* данных из другого узла, а не для *записи* их в узел.

Общедоступные сигналы также могут быть *импортированы/экспортированы* в другие узлы сети через I/O блоки путем связывания их через функцию SNVT. Данная функция изначально предназначалась для осуществления коммуникации между оборудованием разных производителей, например освещения или интеллектуальных приводов, через одну сеть.

При подсоединении к физическому входу/выходу, номер модуля и название входа/выхода отображаются под символом I/O блока, например "*M1-U3*" (модуль номер 1, универсальный вход номер

3). А при подключении к удаленному сетевому сигналу, "*Net*" или "*SNVT*" отображается вместо названия входа/выхода.

В зависимости от типа I/O блока и связывания, используются различные уставки параметров конфигурации, смотри Справочный раздел данного руководства. Каждый I/O- модуль может быть сконфигурирован как:

- Физическое подключение (Physical Input)
- Сетевая переменная (Network Variable)
- Устройство на-линии (Online Device)
- SNVT

Физическое подключение (Physical Input)

Номер модуля (Mod Number)	Для основных устройств отображается в названии, например: Xenta 301. Для модулей входов/выходов отображается как номер модуля и тип модуля, например: M1 (422)
Название входа/ выхода (Terminal Ref)	Последовательность символов указывающая тип и номер входа/выхода (например, В1-В4, U1-U4, X1-X4, Y1-Y4, К1-К4). При выборе входа/выхода в меню выбора входа выхода, символ "#" после адреса входа/выхода указывает, что вход/выход уже занят.

Сетевая переменная (Network Variable)

Сетевой адрес (Network Address)

Ссылка на общедоступный сигнал в другом устройстве TAC Xenta, например, \RPU1\АНУ1\OutdoorTemp (максимальное число знаков \20\12\20). Сетевой адрес в блоке DI всегда должен соответствовать бинарному сигналу в другом устройстве, в тоже время сетевой адрес в блоке AI может соответствовать как целому, так и дробному сигналу.

Чтобы найти сетевой адрес нажмите кнопку обзора ([...]). Если база данных TAC Vista находится в режиме на-линии, просмотрите базу данных для поиска сетевой переменной. Если база данных в режиме не-на-линии, откройте *.MTA файл и просмотрите файл для поиска сетевой переменной.

Обратите внимание!

Выход AI блока всегда дробный.

Дельта (Delta)

Минимальное изменение значения сигнала, которое приводит к обновлению значения сигнала в сети. Значение по умолчанию = 0.5.

Период (Period)

Максимальный временной интервал в секундах (целое значение) между двумя обновлениями импортируемых/экспортируемых данных. Значение по умолчанию = 60. В независимости изменилось значение сигнала или нет, оно будет обновлено, по истечении данного интервала времени после последнего обновления.



Обратите внимание!

Дельта и *Период* должны выбираться таким образом, чтобы избежать сильной загруженности сети. Лучше отправлять сигналы через сеть раз в минуту или в пол-минуты, нежели каждую секунду.

Устройство на-линии (Online Device)

Основной DI блок может быть использован для отслеживания наличия связи с любым удаленным устройством в сети, используя опцию связывания "online device" и задавая сетевой адрес удаленного устройства в таблице конфигурации I/O. Это может быть использовано, например, в приложениях, использующих

**Адрес устройства
(Device Address)**

сигналы от других устройств сети, и в которых при обрыве связи должны быть выполнены дополнительные действия.

Вводится в виде строки. Выход DI блока будет ИСТИНА (1), если удаленное устройство находится на линии, в противном случае значение выхода равно ЛОЖЬ (0). Если в параметре *Адрес устройства* указан адрес программируемого устройства и в сети не найдено данное устройство, находящееся на связи, то выход блока будет иметь значение ЛОЖЬ (0), в противном случае значение на выходе будет ИСТИНА (1).

SNVT

Опция SNVT используется для импорта/экспорта внешних данных через сеть. Сигналу сначала присваивается SNVT имя (которое может отличаться от названия сигнала) в диалоге "связывания", и затем выбранная SNVT переменная "связывается" с внешним сигналом с таким же типом SNVT.

Перечень поддерживаемых SNVT смотри в "[Перечень SNVT](#)" на [стр. 235](#).

**Обратите внимание!**

Передачи данных между двумя связанными SNVT переменными не будет, пока либо выходная переменная не станет в режим периодической *отправки (Send)* нового значения (или значение сигнала изменится на величину *Дельта*), либо входная переменная не будет в режиме периодического *опроса (Poll)* входа.

**Тип
(Type)**

Тип выбирается из определенного списка поддерживаемых типов. Этот список определяется в файле TATYPE.INI. Пожалуйста, свяжитесь с представителями TAC, если необходимый тип SNVT не содержится в списке. Если выбранная SNVT **Структурированного** типа (при выборе структурированного типа все входящие в SNVT переменные автоматически отображаются в поле Поле (Members)), то SNVT содержит не одну, а несколько переменных, и в этом случае каждая переменная должна быть представлена (и связана) в отдельном блоке. Для этих целей существуют готовые Макро блоки для всех структурированных SNVT в библиотеке TAC Макро блоков; всегда используйте их!

**Имя SNVT
(SNVT Name)**

Имя сигнала хранится в .XIF файле и выбирается таким, каким оно должно быть представлено во внешних приложениях. Имя должно состоять не более чем из 16 символов.

Начальное значение (Initial Value)	Этот параметр определяет значение на выходе блока при инициализации.
Период (Period)	После установления внешних связей, соединение запрашивается/обновляется каждый раз, как истекает временной интервал равный <i>Периоду</i> . Если <i>период</i> равен 0 в АО/ДО блоках, то внешняя переменная будет обновляться лишь при изменении значения сигнала на величину, определяемую параметром <i>Дельта</i> .
Дельта (Delta)	Минимальное изменение значения сигнала, которое приводит к обновлению значения сигнала в сети.

3.6.2 Общедоступные (Public) параметры

Чтобы надлежащим образом обеспечить доступность внутренних параметров функциональных блоков для других узлов, эти величины должны быть определены как *общедоступные константы*, смотри "*Константы на стр. 29*". Таким образом общедоступные параметры функциональных блоков могут изменяться через сеть, например, с панели оператора, во время выполнения программы.

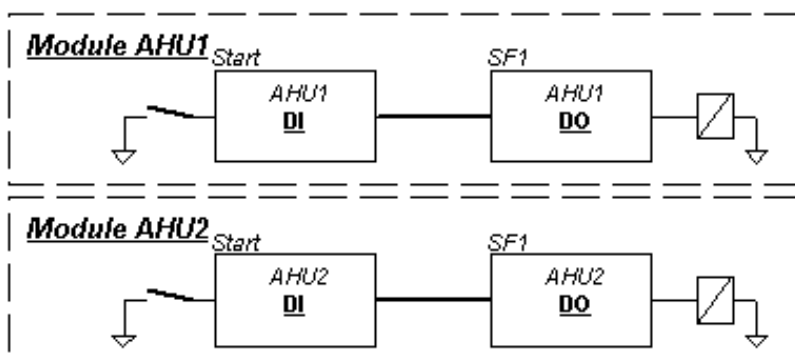


Обратите внимание!

Общедоступная константа может быть задана только для одного параметра одного блока.

3.7 Модули (Module)

В блоке, принадлежность в модулю может быть частью названия сигнала (обозначается: Модуль\Сигнал). Название модуля может быть введено и отредактировано для всех блоков в группе одновременно. То, что блок относится к определенному модулю показывает пурпурное и подчеркнутое имя блока.



Принцип Модулей позволяет разделять программы приложения для различных систем, например, несколько групп отопления. Это дает два основных преимущества:

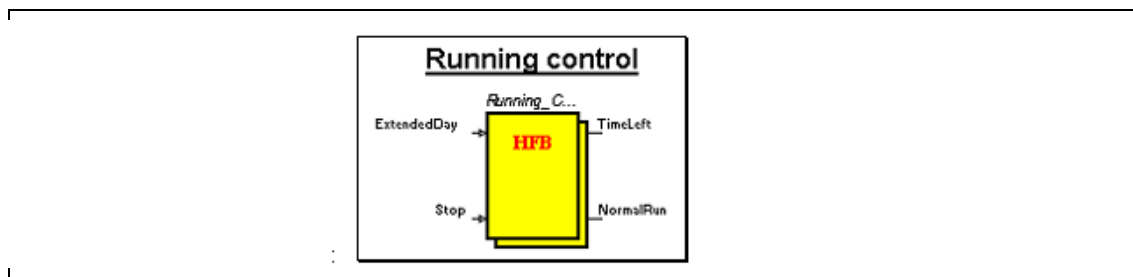
- Все общедоступные сигналы принадлежащие к одному Модулю будут располагаться совместно при переносе их в контролируемую систему (TAC Vista).
- Сигналы в одинаковыми названиями могут использоваться в разных Модулях, например, . АНУ1\Т1_SP и АНУ2\Т1_SP.

Если общедоступные сигналы, принадлежащие Модулю, используются как *Сетевой адрес* в *Сетевых переменных* (TAC-NV), Имя модуля должно быть включено (обозначается: \Устройство\Модуль\Сигнал) где "Устройство" это название устройства TAC Xenta. Сигналы без принадлежности к какому-либо блоку адресуются следующим образом \Устройство\Приложение\Сигнал, где "Приложение" - это название приложения, которое вводится в поле *Имя* в таблице **программной спецификации**. В качестве альтернативы, сигнал без привязки в Модулю может быть адресовано так \Устройство\Сигнал.

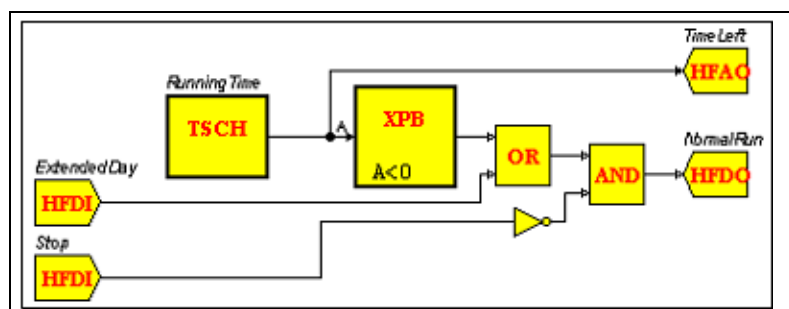
3.8 Иерархические функциональные блоки(HFB)

Лучшее представление FBD может быть достигнуто объединением блоков в иерархические функциональные блоки (HFB). HFB могут включать другие объединенные функциональные блоки (другие HFB) и соединения. Поэтому можно создавать HFB более чем одного уровня. Входы и выходы формируются определенными блоками, например, блок HFAI для входного аналогового сигнала.

HFB - это графический инструмент в TAC Menta для объединения функциональных блоков, они также могут быть распечатаны. Программа разгружается в устройство TAC Xenta device не как иерархические блоки, а как отдельные функциональные блоки FBD. Рисунки, приведенные ниже показывают HFB в свернутом виде и в развернутом виде.



HFB в свернутом виде



HFB в развернутом виде

3.9 Правила доступа

3.9.1 Сигналы

Доступ непосредственно к выходному сигналу задается в его свойствах:

- RO** Только чтение. Сигнал может быть прочитан с панели оператора и/или с других узлов в сети, но не может быть изменен.
- RW** Чтение/запись. Сигнал может быть прочитан и изменен с панели оператора и/или с другого узла в сети.

Для некоторых блоков, единственным доступным классом является класс RO. Это применимо ко всем блокам без внутренней структуры, где рассчитанное значение на выходе зависит только от текущего значения(ий) входного сигнала(ов). Если выходное значение такого блока изменяется через сеть, изменения не окажут влияния на значение сигнала на выходе, рассчитанное прикладной программой в следующем программном цикле. Существуют типы блоков где, тем не менее, имеет смысл допускать внешнее изменение значения выходного сигнала. Это справедливо например, для интеграторов и сумматоров, где новое значение рассчитывается сложением/вычитанием приращения к предыдущему значению сигнала. Тогда задавая выходной сигнал RW можно сбрасывать его значения через сеть.

3.9.2 Константы

Общедоступные константы имеют уровень доступа RW в FBD. Определенный класс доступа для панели оператора (RW или RO) задается пользователем в программе конфигурации панели оператора (OP Configuration).

3.10 Расписания (*Time Schedules*)

Функция расписания используется для того, чтобы запустить или остановить объект, например, вентустановку. Функция дает возможность выбирать *недельный таймер* для запуска и останова вентустановки в разное время в зависимости от дня недели. Также есть возможность выбрать альтернативное расписание, которое называется *годовой таймер*. Расписания представлены в виде блока *Функциональный блок расписания* (TSCH), имеющего выход для целого сигнала, имеющего положительное/отрицательное значение. Отрицательное значение указывает на количество времени (в минутах), оставшееся до конца заданного в расписании интервала времени, а положительное значение - до его начала.

Каждое *Расписание* может содержать несколько *недельных (Week Charts)* и *годовых (Holiday Charts)* таймеров. Количество недельных и годовых таймеров в каждом расписании определяется двумя параметрами в функциональном блоке TSCH. Годовой таймер также может быть задан в TAC Vista по средствам функции глобального расписания (смотри *TAC Vista Basic Functions Manual, (0-004-7647)*).

3.11 Текст аварийных сообщений

Текст аварийных сообщений связан с блоком аварий. Текст аварии включает аварийное сообщение, сгенерированное при переходе аварийного блока в единицу. Можно отредактировать текст аварийного сообщения с помощью функции редактирования аварийных сообщений.

3.12 Локальный Тренд Лог (*Trend Logs*) в TAC Xenta

'Локальный Тренд Лог в TAC Xenta' - это функция для сбора (регистрации) значений и хранения их в TAC Xenta для переноса в TAC Vista в последующее время. В TAC Vista собранные значения могут быть сохранены, обработаны и отображены. Функция в основном используется совместно с удаленным соединением между TAC Vista и TAC Xenta.

Регистрируемый сигнал может быть общедоступным дробным, целым или бинарным сигналом. Можно записывать как быстротекущие процессы за короткое время; в этом случае интервал записи (Log Interval) будет составлять секунды (минимальный интервал записи 10 секунд), так и длительные процессы с интервалом записи равным нескольким дням или неделям, такие как значения потребления воды или энергии с отметкой даты и времени. 'Локальный Тренд Лог в TAC Xenta' может быть задан из TAC Menta, но в ней нельзя отобразить записываемые значения, такое возможно только в TAC Vista.

3.13 Файлы данных

Во время создания окончательной прикладной программы, формируются следующие файлы данных. Пожалуйста обратите внимание, что все файлы, связанные с прикладной программой, должны располагаться в *той же папке*.

.AUT	Текстовый файл с кодом прикладной программы и данными, включающими графические блоки диаграмм.
.TLG	Файл тренд лога.
.MCB	Файл макроблока (Группы блоков).
.COD	Файл в "Машинном коде" в формате ASCII для загрузки в контроллер TAC Xenta.
.XLG	Откомпилированный файл тренд лога.
.ESP	Файл спецификации с общедоступными сигналами, их атрибутами и т.д., сгенерированный TAC Menta. Этот файл содержит входные данные для конфигурации панели оператора.
.OPE	Файл с пустым деревом меню, т.е. файл меню панели оператора без спецификации. Это тип файла, который создается при запуске программы конфигурирования панели оператора.
.OPC	Совокупность .OPE и .ESP файлов. Этот файл создается при запуске программы конфигурации панели оператора из TAC Menta.
.BIN	Двоичный файл с деревом меню панели оператора для загрузки в контроллер TAC Xenta совместно с .COD файлом.
.CHR	Файл установки символов русской раскладки для загрузки в контроллер TAC Xenta совместно с .COD файлом.
.XIF	Файл внешнего интерфейса, содержащий стандартизованное описание сетевых переменных/объектов прикладной программы, которое позволяет осуществлять "связывание" и коммуникацию с узлами LonWorks разных производителей.

.MTA Файл проекта Menta, который может содержать в себе все файлы (все типы упомянутые выше кроме файлов .MSB и .OPE), относящиеся к данной прикладной программе. Файл однажды вставленный (сохраненный) в файле проекта (например, .AUT или .OPC), в любое время может быть извлечен (открыт) как .MTA файл.

Файл .MTA может быть сохранен непосредственно в базе данных TAC Vista.

Обратите внимание!

Файл .MTA может быть сохранен в базе данных TAC Vista даже если он не откомпилирован. Неоткомпилированный файл .MTA конечно не возможно будет загрузить в устройство TAC Xenta. Он должен быть правильным и откомпилированным до того, как загрузка будет возможна.

.В дополнение к перечисленным выше файлам, существуют следующие файлы данных, которые участвуют в создании .BIN файла. Эти файлы расположены в специальной папке для временных файлов.

OPDOC.GOP ASCII файл создаваемый программой конфигурации панели оператора, когда запускается команда Генерировать (Generate). Этот файл конвертируется в .SPT файл, который затем конвертируется в .BIN файл.

OPDOC.SPT Вспомогательный ASCII файл генерируется из .GOP файла.

OPDOC.TMP Вспомогательный ASCII файл генерируется из .GOP файла.

При загрузке временно создается следующий файл:

BPR Файл сетевого окружения, описывающий окружающие узлы и группы устройств TAC Xenta.

4 Мышь и функциональные клавиши

4.1 Использование мыши

Возможности использования мыши различаются в зависимости от режима работы: Редактирование или имитация. В первую очередь они различаются в зависимости от действий. Возможные действия:

- Нажатие левой кнопки.
- Нажатие левой кнопки с удержанием кнопки Shift.
- Нажатие левой кнопки с удержанием кнопки Ctrl.
- Двойное нажатие левой кнопки.
- Нажатие правой кнопки.

Во-вторых, действия имеют разный эффект в зависимости от того где располагается курсор в момент выполнения действия. Возможные области в окне прикладной программы для осуществления действий следующие:

Блок	Желтый прямоугольник (функциональный блок).
Выход	Все функциональные блоки имеют выход, который представляет собой маленький отрезок линии, который выдается справа от прямоугольника.
Вход	Некоторые функциональные блоки имеют вход, который отображается как стрелка.
Соединение	Любые промежуточные точки отрезков линий соединения.
Узел	Окончания отрезков линий соединений, называемые узлами.
Фон	Любая область в окне прикладной программы, отличная от перечисленных выше областей, относится к фону.

В конечном счете, окончательные возможности зависят от условий редактирования. Существуют различные условия редактирования, определяемые по виду курсора:

Нормальный	Курсор в виде классической стрелки.
Вставка блока	Курсор в виде стрелки на фоне блока.
Увеличение	Курсор в виде увеличительного стекла (лупы).
Выделение группы	Курсор в виде раскрытой руки.
Рисование соединений	Курсор в виде X. Символ курсора изменяется на X когда соединение возможно связать со входом.
Выделение прямоугольника	Курсор в виде перекрестия.
Перемещение выделенного	Курсор в виде перекрестия, заканчивающегося стрелками.

4.2 Использование функциональных клавиш

<u>Клавиша</u>	Описание
Delete	Удаляет выделенный блок или группу блоков.
F1	Помощь.
F2	Пуск или стоп (переключатель) выполнения прикладной программы. То же самое, что нажатие кнопки Выполнение в режиме имитации.
F3	Пошаговое исполнение программы приложения. То же, что и нажатие кнопки По шагам в режиме имитации.
F5	Переключатель между <i>FBD</i> и <i>табличным</i> режимами имитации.
F7	Поиск блоков или комментариев, содержащих заданный текст в названии блока, типе блока или параметре. То же, что команда Правка- Найти (Edit - Find) . Клавиша работает в режимах Редактирование и имитации.
F8	Поиск заданного текста в комментариях, названиях блоков и параметрах блоков, и замена его на другой текст. То же, что команда Правка - Переместить (Edit - Replace) в режиме редактирования.
F9	Перемещение выделенной области в центр экрана. То же, что команда Правка- Центрировать выборку (Edit - Center selection) . Клавиши работают в режиме Редактирования и в режиме имитации.

F10	Сбросить счетчик программного цикла в ноль и все сигналы функциональных блоков в их начальные значения. Тоже, что и нажатие на кнопку RST в режиме имитации.
F11	Переключатель между режимами На-линии и Не-на-линии. Тоже, что нажатие на кнопку Online в режиме имитации.
F12	Переключатель между режимами Редактирование и имитации .
CTRL+H	Увеличение масштаба.
CTRL+O	Уменьшение масштаба.
CTRL+X	Вырезать текст.
CTRL+C	Копировать выбранную группу блоков или копировать выделенный текст (тоже что Ctrl-Insert).
CTRL+V	Вставка скопированной группы блоков или выделенного текста (тоже что Shift-Insert).
CTRL+Z	Отмена удаления.
CTRL+Insert	Копировать выбранную группу блоков или копировать выделенный текст.
Shift+Insert	Вставка скопированной группы блоков или вставка выделенного текста.
Tab	Увеличение (тоже что Ctrl-I).
Esc	Отмена действия (тоже что кнопка Cancel в диалоговом окне).
Клавиши курсора	Перемещение окна FBD в направлении клавиш курсора (Влево, Вправо, Вверх и Вниз). Доступно в режиме Редактирования и режиме имитации.
Shift+клавиши курсора	Перемещение окна FBD на один экран в направлении клавиш курсора (Влево, Вправо, Вверх или Вниз). Доступно в режиме Редактирования и режиме имитации.
CTRL+клавиши курсора	Перемещение курсора в окне Тренд логов на один шаг в направлении клавиш навигации (Влево или вправо).

5 Режим редактирования (Edit)

Когда TAC Menta запускается из меню Microsoft® Windows или из проводника, то она всегда запускается в режиме Редактирования. В этом режиме, основное окно содержит единственное под-окно с двумя полосами прокрутки. Это окно называется окном диграмм, в нем отображаются FBD блоки.

5.1 Опции меню

5.1.1 Пункты меню

В пунктах меню содержится 8 групп опций, каждая имеет различные подпункты:

- Файл (File)
- База данных (Vista Database)
- Правка (Edit)
- Свойства (Preferences)
- Опции (Options)
- Масштаб (Zoom)
- Инструменты (Tools)
- Помощь (Help)

Файл (File)

Новый (New)	Закрывает текущую прикладную программу и открывает новый файл. Если текущий файл не был сохранен, программа выдаст сообщение, предлагающее пользователю сохранить файл перед открытием нового.
Открыть (Open)	Загружает прикладную программу. Файлы прикладной программы должны иметь расширение .AUT или .MTA. Текущий файл будет закрыт до открытия нового. Если текущий файл не был сохранен, то программа выдаст сообщение, предлагающее пользователю сохранить файл перед открытием нового. Предлагаемая для сохранения папка, обычно, та, которая использовалась последней.
Сохранить (Save)	Сохраняет текущую прикладную программу с текущим именем.
Сохранить как (Save As...)	Сохраняет текущую прикладную программу, но предлагает пользователю ввести новое имя файла. Меню панели оператора (.OPC) и другие связанные текстовые файлы так же сохраняются с принятым новым именем файла. Если пользователь открывает прикладную программу из предыдущей версии TAC Menta, он попадает в диалоговое окно, в котором предлагается выбрать формат (версию), в котором прикладная программа будет сохранена. Можно сохранить приложение в формате TAC Menta версия 4, TAC Menta версия 3.2 или TAC Menta 3.1.
Сохранить как демо (Save As Demo)	Сохранение текущего приложения как демо-приложения. Демо-приложение не может быть загружено в любой контроллер TAC Xenta, но позволяет пользователю изучить все функции TAC Menta. Сохранение как демо - это единственная опция для сохранения приложений при использовании нелицензионной TAC Menta. Для более подробной информации обращайтесь к "TAC Menta программные лицензии" на стр. 20.
Печать (Print)	Выбор части прикладной программы: FBD, HFB, OP дерево меню, связанные текстовые файлы и/или программные спецификации, и печать ее на принтере.

**Экспорт I/O списка
(Export I/O List)**

Сохранение списка I/O текущей прикладной программы в текстовом ASCII файле (.TXT). Пользователю предлагается ввести имя файла.

**Создание меток
(Create Labels)**

Создает метки в файле Microsoft® Excel для прикладной программы, открытой в TAC Menta. Пользователю предлагается сохранить Excel файл.

Выход (Exit)

Завершение работы с TAC Menta и выход в проводник Microsoft® Windows. Если текущий файл приложения не был сохранен, при выходе появится сообщение с предложением сохранить файл.

База данных (Vista Database)**Открыть ... (Open)**

Перемещение в базе данных TAC Vista и выбор устройств TAC Xenta. Прикладная программа для выбранного устройства открывается в TAC Menta.

Сохранить (Save)

Сохранение прикладной программы, открытой для устройства TAC Xenta.

**Сохранить как...
(Save As...)**

Перемещение в базе данных TAC Vista и выбор устройства TAC Xenta для сохранения прикладной программы. "Сохранить как..." используется для сохранения новых приложений для устройства TAC Xenta или для сохранения прикладной программы для другого устройства TAC Xenta.

Вход в систему (Log In)

Вход в систему в TAC Vista Server использует учетную запись TAC Vista или учетную запись Windows NT. Введите **Имя пользователя (User name)** и **Пароль (Password)** и нажмите **ОК**. Для информации о том, как осуществлять вход в систему в TAC Vista смотрите Руководство по Основам TAC Vista.

Обратите внимание!

Для возможности сохранения прикладных программ в базе данных необходимо осуществить вход в систему в TAC Vista.

**Выход из системы
(Log Out)**

Осуществляет выход из системы в TAC Vista.

Правка (Edit)	
Откат (Undo)	Отмена удаления одного или более выбранных функциональных блоков, включая связанные соединения.
Копировать в буфер (Copy To Clipboard)	FBD целиком или выбранная часть его копируется в буфер в формате мета файла Microsoft® Windows. Он может быть вставлен из буфера в другие приложения Microsoft® Windows.
Копировать (Copy)	Создание копии выбранных блоков и соединений и сохранение их во внешнем буфере. Тоже что команда ГРУППА - Копировать (GROUP - Copy) .
Вставить (Paste)	Вставка скопированных блоков и соединений из внешнего буфера. Тоже что команда НОВЫЙ - Вставить (NEW - Paste) .
Удалить (Delete)	Удаление всех выбранных блоков и соединений.
Выбрать все (Select All)	Открытие минимальновозможного прямоугольника выделения, включающего всю схему FBD.
Найти (Find)	Поиск блоков или комментариев, которые содержат заданный текст в названиях блоков, типах блоков или параметрах. Первый найденный блок подсвечивается зеленым прямоугольником. В диалоговом окне пункта Найти также содержится кнопка для поиска следующего подходящего блока. Последний введенный текст для поиска сохраняется и может быть вновь вызван в диалоговом окне. Функциональная клавиша: F7.
Замена (Replace)	Поиск заданного текста в комментариях, названиях блоков и параметрах блоков и замена его на другой текст. Первый найденный блок подсвечивается зеленым прямоугольником. В диалоговом окне пункта "Найти" также содержится кнопка для поиска следующего подходящего блока и замены всех соответствий. Последний введенный текст для поиска сохраняется и может быть вновь вызван в диалоговом окне. Функциональная клавиша: F8.
Центрировать выборку (Center Selection)	Перемещение выбранной области в центр экрана. Функциональная клавиша: F9.

**Добавить блок
перехода
(Add Boundary Ties)**

Добавить блоки перекрестных ссылок на обеих сторонах маркеров окончания страницы в окне FBD для всех соединений, пересекающих окончание страницы. После подготовки программы к печати, блок перехода на другую страницу отобразит страницу и номер сигнала, где соединение продолжается.

**Удалить блоки
перехода (Remove
Boundary Ties)**

Удаление всех блоков перехода.

**Снять выделение всех
соединений (Unmark
All Connections)**

Возвращение к нормальному отображению всех соединений.

Свойства (Preferences)

Параметры страницы (Page setup)

Выбор ориентации, размера, отступов сверху и снизу для вывода на печать FBD. Параметры страницы отражаются также в окне FBD, где границы страницы, выводимой на печать, отмечены голубыми линиями. Графические размеры приложения определяются числом горизонтальных и вертикальных страниц.

Показывать только общедоступные идентификаторы (View Only Public Identifier)

Если эта опция активирована, только общедоступные идентификаторы общедоступных блоков будут отображаться в диаграммах. См. раздел 4.3.6. По умолчанию: все имена блоков отображаются.

Показывать строку текущего состояния (Show Status Bar)

Поле в панели состояния внизу окна прикладной программы показывает на каком уровне иерархии находится FBD. Информационное сообщение отображается в левой части панели, а положение курсора в координатах X,Y (ячейка) отображается в правой части (вертикальнорасположенная страница A4 составляет поле 160*200 ячеек).

Ортогональные соединения (Orthogonal Connections)

Активирует функцию ортогональных соединений. См. *"Ортогональные соединения"* на стр. 59 для более детальной информации. По умолчанию: активировано.

Черно-белый (Black and White)

При активации этой опции, блоки будут отображаться в черно-белом цвете вместо цветного. Эта функция используется тогда, когда необходимо скопировать блок через блокнот в другое приложение, которому требуется черно-белая графика. По умолчанию: цветной.

Параметры (Settings)

Эта команда позволяет пользователю выбрать путь по умолчанию для модулей библиотеки, вызываемых в командах *Загрузить группу* и *Сохранить группу*, и выбрать текстовый редактор, с помощью которого можно будет просматривать связанные с приложением текстовые файлы. Для это необходимо указать расширение открываемого файла, и задать текстовому редактору путь к программе.

Диалоговое окно также содержит флаговое окошко, установка галочки в котором означает, что текущий FBD будет автоматически сохранен в файл AUTOSAVE.AUT перед переходом из режима редактирования в режим имитации.

Пользователь может выбрать какой коммуникационный порт (COM1, COM2 и т.д.) использовать в данном сеансе, когда запущено несколько сеансов TAC Menta.

Опции (Options)

Имитация (Simulate)	Компиляция прикладной программы и, если ошибок не обнаружено, переход в режим имитации. Функциональная клавиша: F12.
Установка даты и времени (Set Date And Time)	Установка начальных значений даты и времени в режиме не-на-линии (возможность тестирования функций прикладной программы в зависимости от даты и времени суток). Начальные значения даты и времени хранятся в файле кода программы (.AUT или .MTA).
Конфигурация устройства (Device Configuration)	Это пункт позволяет пользователю задавать тип устройства TAC Xenta (301/302/401) и тип модулей входов/выходов, которые используются в приложении.
Программная спецификация (Program Specification)	Показывает список общедоступных сигналов и параметров. В этом диалоговом окне задается название прикладной программы (Name:), короткое имя программы (Abbr:) и длительность программного цикла (Cycle Time). Текущий ID программы (Program ID), записанный в XIF файл тоже показан.
Таблица констант (Constant Table)	Эта опция позволяет изменять, добавлять и удалять константы, которые используются в параметрах блоков.
Таблица конфигурации I/O (I/O Configuration Table)	Открывает таблицу конфигурации блоков I/O для определения и изменения "связывания" блоков I/O с физическими входами/выходами или сетевыми адресами.
Таблица расписаний (Time Schedule Table)	Открывает таблицу расписаний для изменения, дополнения или удаления.
Таблица аварийных сообщений (AlarmText Table)	В ней задаются и изменяются тексты аварийных сообщений, которые содержат сетевое аварийное сообщение, генерируемое при переходе аварийного блока в единицу.
Тренд Логи (Trend Logs)	Открывает диалоговое окно для задания локальных тренд логов в устройстве TAC Xenta.
Использование памяти (Memory Usage)	Происходит считывание и подсчет объема используемой памяти в текущем приложении. В диалоговом окне показана вся используемая память (подразделенная на разные типы загружаемых файлов), доступная память и свободная память в устройстве TAC Xenta. Для получения более точной оценки, необходимо сначала перейти в режим имитации и выбрать подпункт Команды - Генерировать (Commands – Generate) .

Масштаб (Zoom)**Увеличение (Zoom In)**

При выборе этого пункта, форма курсора мыши изменяется на увеличительное стекло. Программа ждет когда будет очерчен прямоугольник в окне FBD, выбирая таким образом участок для увеличения. Если после выбора увеличения нажать на точке в FBD без очерчивания прямоугольника, выбранная тока окажется в центре экрана увеличенная до нормального масштаба. Горячие клавиши: Ctrl I.

**Уменьшение
(Zoom Out)**

При выборе этого пункта, FBD перерисовывается в уменьшенном размере с наименьшим уровнем детализации. Горячие клавиши: Ctrl O.

Нормальный (Normal)

Возвращение к нормальному масштабу.

**Уменьшить масштаб
на 200%
(Zoom Out 200%)**

Устанавливает степень уменьшения масштаба 200%, 300% или 400%. Для возвращения к нормальному масштабу (100%) выберите Normal.

**Уменьшить масштаб
на 300%
(Zoom Out 300%)****Уменьшить масштаб
на 400%
(Zoom Out 400%)**

Инструменты (Tools)

Текстовый файл (Text File)	Вызывает текстовый редактор для редактирования связанных с приложением текстовых файлов, смотри <i>"Связанные текстовые файлы" на стр. 108.</i>
Конфигурация панели оператора (OP Configuration)	Вызывает программу конфигурации панели оператора, смотри <i>"Программа конфигурации панели оператора" на стр. 89.</i>
Панель инструментов (Tool Bar)	Открывает панель инструментов для создания простых блоков или операторов. Не доступен, если TAC Menta запущена под Windows NT или Windows 2000.
Обновить экран (Redraw Screen)	Обновляет экран.

	<i>Помощь (Help)</i>
Содержание (Contents)	Вызывает help файл с основной информацией из этого руководства. Функциональная клавиша: F1.
ТАС в Интернет (ТАС on the Web)	Посещение домашней страницы ТАС.
О программе (About ТАС Menta)	Отображение информации о программе ТАС Menta.

5.1.2 Всплывающие меню

Всплывающие меню появляются при нажатии на правую кнопку мыши в окне прикладной программы и исчезают при нажатии на левую кнопку. Для отключения всплывающего меню без дополнительных действий, челкните на верхней границе или вне поля меню. Нажатие левой кнопки на одном из пунктов меню запускает соответствующую опцию.

Всплывающие меню, появляющиеся в зависимости от *условий редактирования и выбранной области*.

НОВЫЙ (NEW)

	Появляется при нажатии правой кнопкой мыши на фоне в режиме редактирования.
Простой блок (Simple Block)	Создание простого блока.
Оператор (Operator)	Создание блока оператора.
Блок формул (Expression)	Создание блока формул.
Тестовый блок (Test Probe Block)	Создание тестового блока.
Узел (Node)	Создание начала соединения целого, дробного или бинарного типа.
HFB	Создание HFB (иерархический функциональный блок).
HFB I/O	Создание входа или выхода HFB.
Сжать HFB (Compress HFB)	Свертывание раскрытого HFB.
Комментарий (Comment)	Создание блока текстового комментария.
Линия (Line)	Создание начала линии комментария.
Прямоугольник (Rectangle)	Создание прямоугольника из пунктирной линии для выделения функционально законченной области FBD.
Вставка (Paste)	Вставка (после <i>ГРУППА - Копировать (GROUP - Copy)</i>) скопированных блоков или соединений из внешнего буфера.
Загрузка группы (Load Macro)	Загрузка группы блоков и соединений с диска (макроблок).

БЛОК (Block)

Появляется при нажатии правой кнопки мыши на блоке в режиме редактирования.

Правка (Edit)	Открытие окна для правки параметров блока.
Дублировать (Duplicate)	Создание нового блока идентичного одному выбранному.
Отсоединить (Disconnect)	Удаление всех соединений связанных с входами и выходами блока.
Удалить (Delete)	Удалить блок.
Раскрыть HFB (Expand HFB)	Отображение иерархического блока в развернутом виде.

СОЕДИНЕНИЕ (CONNECTION)

Появляется при нажатии правой кнопкой мыши на соединении/ узле в режиме редактирования.

**Переместить узел
(Move Node)**

Перемещение узла с перетаскиванием соединений, объединенных с ним.

**Создать узел
(Create Node)**

Создание ответвления от соединения.

**Удалить узел
(Delete Node)**

Удаление выделенного соединения.

Отсоединить (Detach)

Разрыв соединения пополам в выбранной точке и разделение соединения на два сегмента.

Выделить (Mark)

Подсвечивание выбранного соединения и его ответвлений разными цветами. Существует 6 разных цветов, из которых можно выбрать.

**Снять выделение
(Unmark)**

Возвращение в нормальный режим отображения соединения.

**Снять выделение всех
соединений
(Unmark All)**

Возвращение в нормальный режим отображения всех соединений.

ВХОД (INPUT)

Появляется при нажатии правой кнопкой мыши на входе блока в режиме редактирования.

**Разъединить
(Disconnect)**

Удаление соединения связанного с выбранным входом.

Отсоединить (Detach)

Отсоединение входа без удаления соединения.

ВЫХОД (OUTPUT)

Появляется при нажатии правой кнопкой мыши на выходе блока в режиме редактирования.

**Разъединить
(Disconnect)**

Удаление всех соединений, связанных с выходом блока.

Отсоединить (Detach)

Отсоединение выхода без удаления соединения.

ГРУППА (GROUP)

	Появляется при нажатии правой кнопкой мыши в любом месте окна, когда очерчен прямоугольник выделения.
Снять выделение (Unselect)	Отмена выделения и переход в нормальный режим редактирования.
Копировать (Copy)	Копирование выбранных блоков и соединений во внешний буфер.
Разъединить (Disconnect)	Если соединение выделено частично (выделено только его начало или конечная точка, а не обе одновременно), то соединение будет удалено. Полностью выделенное соединение также будет удалено (при выделении и начальной и конечной точки).
Удалить (Delete)	Удаление всех выделенных блоков и соединений.
Модуль (Module)	Открытие диалогового окна для ввода и изменения названия модуля во всех выбранных блоках.
Сохранить (Save)	Сохранение выбранной части прикладной программы на диск как макроблока (с расширением .MTA/.AUT).
Создание HFB (Create HFB)	Объединение выбранных блоков и соединений в блок HFB.

5.2 *Функциональные блоки*

5.2.1 *Создание нового блока*

Использование всплывающего меню *НОВЫЙ (NEW)*

Для создания нового блока в прикладной программе, необходимо выполнить следующие действия:

1. Щелкните правой кнопкой мыши по фону. В появившемся окне **НОВЫЙ (NEW)** выберите класс блока: простой (Simple Block), оператор (Operator), блок формул (Expression) или тестовый блок (Test Probe Block).



Обратите внимание!

Если класс нового блока такой же, как и у последнего созданного, Вы можете просто дважды нажать на фоне.

2. После выбора класса блока, появится диалоговое окно выбора блока. Вид окна зависит от выбранного класса: для классов **Простой блок**, **Оператор** и **Тестовый блок** окна похожи. Для выбора следует выполнить одно из трех действий:
 1. Двойное нажатие на элементе в списке типов.
 2. Нажатие на элементе в списке и подтверждение (**ОК**).
 3. Введение вручную типа блока и подтверждение (**ОК**).

Выход путем выбора **Cancel** прекратит создание блока.

Для класса **Формула (Expression)** окно выбора такое же как и окно редактирование, его рассмотрение приведено в другом разделе.

3. После закрытия окна, новый блок появится в том месте, где находился курсор при выполнении первого шага. Если местоположение блока не определено, то он появится как плавающее выделение в фоне окна.

Дублирование существующего блока

Простейший путь создания блока, когда эквивалентный уже создан - это дублирование существующего блока. Щелкните правой кнопкой мыши на блоке, который Вы хотите дублировать и выберите *Дублировать (Duplicate)* во всплывающем меню **БЛОК (BLOCK)** для завершения действия. HFВ блок не может быть создан таким образом.

5.2.2 Редактирование блока

Все функциональные блоки имеют определенное число конфигураций, которые определяются в диалоговом окне *Редактирование блока (Edit block)*. Вид окна несущественно зависит от типа выбранного блока.

Есть два способа перехода в окно *Редактирование блока (Edit block)*:

1. Двойным нажатием на блоке.
2. Из всплывающего меню **БЛОК (BLOCK)** выбором подпункта **Правка (Edit)**.



В общих чертах простое диалоговое окно *Редактирование блока (Edit Block)* выглядит следующим образом.

The image shows a dialog box titled "Edit block XXX". It contains the following elements:

- Identifier:** A text input field.
- Unit:** A dropdown menu with a downward arrow.
- Description:** A text input field.
- Options:** Three checkboxes labeled "Public", "RW", and "Backup".
- Status:** A box containing "Type: Real" and "Access: RW".
- Parameters:** A section with two text input fields labeled "Par1:" and "Par2:".
- Buttons:** "OK", "Cancel", and "Help" buttons at the bottom.

В окне Редактирование Блока, могут содержаться следующие поля:

Идентификатор (Identifier)	Это имя блока, появляющийся над левым верхним углом блока. Этот идентификатор также связан с названием выходного сигнала. Обязательно заполнять эту строку, если сигнал является общедоступным (Public), или если это блок входа/выхода, аварии или TSCH блок. Максимальное количество символов - 20. (смотри " <i>Названия сигналов</i> " на стр. 24).
Ед. измерения (Unit)	Выбор Unit (ед. измерения) из предлагаемого списка. Unit передается в TAC Vista, в случае, если это необходимо. Данное поле не обязательно заполнять.
Описание (Description)	В этом поле вводится описание блока. Оно не должно превышать длины в 40 символов, исключая символ '. Описание передается в TAC Vista, в случае необходимости. Заполнение этого поля не обязательно.
Общедоступный (Public)	Если в поле <i>Общедоступный (Public)</i> стоит галочка, выход блока будет определен как общедоступный сигнал и будет включен в программную спецификацию.
Чтение/Запись (R/W)	Если в поле <i>Чтение/Запись (Read/Write)</i> стоит галочка, то выходной сигнал можно изменять из других узлов в сети во время выполнения прикладной программы в устройстве TAC Xenta и вручную во время имитации при отключенном устройстве. Свойство Чтение/Запись не может быть выбрано для блоков с уровнем доступа Только Чтение (RO).
Backup	Если выбрана эта опция, то в RAM-памяти контроллера сохраняются текущие значения сигнала на выходе блока и внутренние уставки, и именно с этих значений начинается работа при горячем старте. Если опция не выбрана, при горячем старте значения сигналов на выходе блока и внутренних уставок сбрасываются в их начальные значения (тоже, что и холодный старт). Если питание контроллера пропадало на период превышающий 72 часа содержимое RAM-памяти стирается, и рестарт в любом случае будет происходить как при холодном старте и все уставки блоков будут сброшены в начальные значения в независимости была выбрана опция Backup или нет.

Режим 2 (Mode 2)	Эта опция, доступна только для блоков операторов с двумя входами, определяет положение входов графически относительно символа оператора: над/под символом оператора ( - <i>Режим 2 (Mode 2)</i> выбран) или слева/под символом оператора ().
Описание (Description)	Это поле показывает тип доступа (RO или RW) и тип данных (дробный, целый или бинарный) выходного сигнала и размер памяти, который блок занимает в RAM-памяти устройства TAC Xenta после компиляции.
Параметры (Parameters)	Появляющийся список параметров может быть пустым или содержать несколько параметров в зависимости от типа данного блока. В блоках входа/выхода, раздел Параметры доступен через кнопку Связь (Bind), которая открывает диалоговое окно задания параметров входа/выхода. Пожалуйста, обратитесь к разделу <i>"Блоки ввода/вывода" на стр. 30</i> для более детальной информации. При нажатии на кнопке <i>Помощь (Help)</i> в правом нижнем углу диалогового окна на экран будет выведено детальное описание данного блока.

5.2.3 Перемещение блока

Для перемещения блока в окне прикладной программы, он должен быть первоначально выбран выделением. Как выделить блок описано в разделе *"Операции над группами" на стр. 71*.

5.2.4 Удаление блока

Для удаления блока из FBD, щелкните правой кнопкой мыши на блоке и в режиме редактирования, при условии вызова всплывающего меню **БЛОК (BLOCK)**, выберите подпункт Удалить (Delete). Можно также выделить блок и нажать кнопку Delete на клавиатуре.

5.2.5 Отсоединить блок

Данная опция предназначена для удаления соединений, которые начинаются или заканчиваются в данном блоке:

Если необходимо отсоединить все соединения от блока, щелкните правой кнопкой мыши на блоке, а затем выберите подпункт Разъединить (Disconnect) во всплывающем меню **БЛОК (BLOCK)**.

Если необходимо удалить только одно соединение, щелкните правой кнопкой мыши на входе, выходе или самом соединении и затем во всплывающем меню выберите пункт *Разъединить/Удалить узел (Disconnect/Delete node)*.

Иногда требуется разъединение соединения без его удаления, например, для замены блока на блок другого типа, или для перемещения соединения из одного места в другое и т.д.:

1. Для отсоединения входа либо щелкнете на входе левой кнопкой мыши, либо щелкните правой кнопкой мыши, и в появившемся меню **ВХОД (INPUT)**, выберите пункт **Отсоединить (Detach)**. В любом случае, соединение перетаскивается курсором мыши и происходит переключение в режим рисования соединения (смотри "[Рисование соединений](#)" на стр. 66).
2. Для отсоединения выхода щелкните на выходе правой кнопкой мыши и в появившемся меню **ВЫХОД (OUTPUT)** выберите пункт **Отсоединить (Detach)**. При этом, группа соединений, начинающихся на выходе перетаскивается и может быть перемещена в любую другую часть программы. (смотри "[Перемещение узла](#)" на стр. 70).

5.3 Соединения

5.3.1 Рисование соединений

При рисовании соединения, курсор мыши превращается в знак X и сопровождается зеленой, ломанной линией, которая всегда фиксируется в одном из ее крайних положений. Для рисования соединений, необходимо выполнить следующие шаги:

1 - Новое соединение

Существует несколько вариантов создания соединения:

- 1 а) Соединение, начинающееся в блоке: щелкните левой кнопкой мыши на выходе блока.
- 1 б) Соединение, начинающееся в узле: щелкните левой кнопкой мыши на узле или щелкните правой кнопкой мыши и в появившемся меню **СОЕДИНЕНИЕ (CONNECTION)** выберите пункт **Создать узел (Create Node)**.
- 1 в) Соединение, начинающее на другом соединении: щелкните правой кнопкой мыши на соединении и в появившемся меню **СОЕДИНЕНИЕ (CONNECTION)** выберите пункт **Создать узел (Create Node)**.
- 1 г) Соединение, начинающееся на фоновом участке: щелкните правой кнопкой мыши на фоне, выберите во всплывающем меню **НОВЫЙ (NEW)** пункт **Узел... (Node ...)** и выберите тип соединения (**дробный, целый** или **бинарный**) в появившемся подменю.

Во всех этих случаях зеленая гибкая линия будет следовать за курсором.

2 - Рисование соединения

Переместите курсор в точку излома соединения и нажмите левую кнопку мыши. Временное соединение будет зафиксировано и новый участок временного соединения начнется с зафиксированной точки излома. Этот шаг может быть повторен столько раз, сколько необходимо, окончательное соединение представляет собой последовательность нескольких сегментов. При удержании клавиши Shift при передвижении курсора (и при отключенной функции ортогональности соединений (Preferences\Orthogonal Connections)) соединительная линия будет ломаться только под углами кратными 45 градусам.

3 - Удаление последнего сегмента соединения

Когда Вы рисуете линию и Вам необходимо удалить последний сегмент, щелкните правой кнопкой мыши и сегмент исчезнет.

4 - Окончание соединения

Для завершения соединения на входе блока, подведите курсор (с зеленой линией соединения) ко входу. Если соединение возможно, вид курсора изменится (X в квадратике). Щелкните левой кнопкой мыши по входу и соединение будет закончено. Чтобы оставить соединение неподсоединенным в какой-либо точке просто щелкните левой кнопкой мыши на фоне два раза.

5.3.2 Удаление соединения

Для удаления соединения щелкните правой кнопкой мыши на любой точке соединения и в появившемся меню **СОЕДИНЕНИЕ (CONNECTION)** выберите пункт **Удалить (Delete)**. Этого можно достичь и из всплывающих меню **ВХОД (INPUT)** и **ВЫХОД (OUTPUT)** в подменю **Разъединить (Disconnect)**. Так же можно выделить соединение и нажать кнопку **Delete**.

5.3.3 Перемещение узла

Для перемещения одного узла щелкните правой кнопкой мыши в любой точке узла и в появившемся меню **СОЕДИНЕНИЕ (CONNECTION)** выбрать пункт **Переместить узел (Move node)**. Также можно выделить узел. При этом узел измениться на временное соединение (зеленая линия) и будет перетаскиваться вместе с курсором. Чтобы зафиксировать узел, нажмите левую кнопку мыши.

5.3.4 Разрыв соединения

Для разрыва соединения в средней точке, щелкните правой кнопкой мыши на соединении и в появившемся меню **СОЕДИНЕНИЕ (CONNECTION)** выберите пункт **Отсоединить (Detach)**. Соединение разорвется на две части и они перейдут в состояние **рисование соединения** (смотри "[Рисование соединений](#)" на стр. 66).

Существует две возможности отделения соединения от входа : щелкните левой кнопкой мыши на входе или щелкните правой кнопкой мыши, и в открывшемся меню **ВХОД (INPUT)** выберите пункт **Отсоединить (Detach)**. В любом случае последний сегмент соединительной линии станет гибким и перейдет в состояние **рисование соединения**.

Для отделения соединения от выхода, щелкните правой кнопкой мыши на выходе и в появившемся меню **ВЫХОД (OUTPUT)** выберите пункт **Отсоединить (Detach)**. Автоматически все подсоединенные соединения станут эластичными и перейдут в состояние **перемещение группы** (смотри "[Перемещение узла](#)" на стр. 70).

5.3.5 Подсвечивание соединений

Для подсвечивания соединений (включая все их ответвления), щелкните правой кнопкой мыши на соединении и в меню **СОЕДИНЕНИЕ (CONNECTION)** выберите пункт **Выделить (Mark)**. Соединение будет отображаться каким-либо цветом, что поможет легче проследить его в теле программы (окно FBD). Для возвращения в нормальный режим отображения соединений, щелкните правой кнопкой мыши на соединении и в появившемся меню **СОЕДИНЕНИЕ (CONNECTION)** выберите пункт **Снять выделение (Unmark)**.

5.3.6 Ортогональные соединения

Режим ортогональности соединений включен по умолчанию, но может быть изменен из **Меню Свойства (Preferences menu)**. При включенном режиме ортогональности два или три ортогональных сегмента создаются при рисовании соединения. При рисовании соединения из узла, создается только два сегмента.

Трехсегментное соединение создается только при рисовании соединения от выхода блока. В этом случае, удержание кнопки CTRL во время рисования соединения, позволяет пользователю зафиксировать вертикальный сегмент в определенном положении.

При рисовании соединения, режим текущего сегмента можно временно поменять с *ортогонального* на *неортогональный* и наоборот, нажимая одновременно кнопки CTRL и Shift.

5.4 Операции с группами

Для осуществления операций с группами, блоки и соединения, принадлежащие группе первоначально следует выделить, при этом курсор примет вид раскрытой руки. В этом случае, при нажатии правой кнопки мыши, появится всплывающее меню **ГРУППА (GROUP)**.

5.4.1 Выделение группы

Для выделения группы, нажмите и удерживайте левую кнопку мыши, появится один из четырех углов области выделения. Курсор мыши изменится на перекрестие, и за ним будут тянуться зеленые линии, образующие прямоугольник. Удерживайте кнопку мыши нажатой и перемещайте курсор в противоположный угол прямоугольника. При отпуске кнопки мыши, все блоки и соединения, попавшие в прямоугольник станут выделенными. Соединение может быть *частично выделено* только если его начальная точка или конечная точка, но не обе крайние точки выделены. Или может быть *полностью выделено*, если обе крайние точки выделены.

Блоки и узлы, находящиеся внутри выделенной области могут быть индивидуально выделены, или с них может быть снято выделение, если щелкнуть на них левой кнопкой мыши удерживать клавишу Shift. При этом, если элемент был выделен, он становится невыделенным и наоборот. При выделении элемента, находящегося за пределами области выделения, ее размеры увеличиваются, чтобы включить элемент.

Для выделения одного блока существует два быстрых способа выделения. Первый - это щелкнуть левой кнопкой мыши на блоке, удерживая нажатой клавишу Shift. Второй - выделить маленькую область, включающую только один блок и совпадающую по размерам с блоком.

Для выделения всей области FBD, существует опция **Выбрать все (Select All)** в меню **Правка (Edit)** в панели меню. Эта опция создает минимально возможную область выделения, содержащую все FBD.

5.4.2 Снятие выделения группы

Для снятия выделения группы самым простым способом, щелкните левой кнопкой мыши вне области выделения. Альтернативный способ - щелкните правой кнопкой мыши внутри выделенной группы и во всплывающем меню **ГРУППА (GROUP)** выберите подпункт Снять выделение (Deselect).

5.4.3 Перемещение группы

Для перемещения выделенной группы нажмите и удерживайте левую кнопку мыши внутри области выделения. Выделенная группа автоматически превратится в "плавающую". Это подразумевает что:

1. Область выделения изменит цвет с зеленого на красный.
2. Курсор мыши примет форму перекрестия со стрелками, показывая, что группу можно перетаскивать.
3. Выделенные блоки и полностью выделенные соединения становятся "плавающими", т.е. блоки и соединения отображаются пунктирными линиями.
4. Отрезки частично выделенных соединений становятся гибкими линиями зеленого цвета.

Пока левая кнопка нажата, группа остается "плавающей" и передвигается вслед за курсором. Гибкие соединения растягиваются, оставаясь связанными с одной из крайних точек, в то время как другие соединения остаются неизменными.

Если курсор мыши передвигается за пределы экрана, он будет прокручен в направлении движения курсора. При отпускании кнопки мыши происходит расчет возможности пересечения между "плавающими" элементами и элементами, находящимися под ними. Это процесс может занять значительный отрезок времени, если группа содержит много элементов. Если нет пересечения между перемещаемыми и стационарными элементами, группа будет вставлена и перерисована в нормальном виде. Тем не менее, если пересечение есть, выбранная группа останется "плавающей" и должна быть перенесена снова в ту область, где пересечения не будет.

5.4.4 Разъединение группы

Для удаления соединений в выделенной группе, выберите пункт **Разъединить (Disconnect)** во всплывающем меню **ГРУППА (GROUP)**. Если существует хотя бы одно *частично выделенное*

соединение в группе, то только этот тип соединений будет удален. Если таких соединений в группе нет, все *полностью выделенные* соединения будут удалены.

5.4.5 Удаление группы

Для удаления группы, выберите **Удалить (Delete)** из всплывающего меню **ГРУППА (GROUP)**, или выделите группу и нажмите клавишу Delete.

5.4.6 Ввод или правка названия модуля

Для вызова диалогового окна ввода или правки названия модуля во всех выделенных блоках, выберите **Модуль (Module)** из всплывающего меню **ГРУППА (GROUP)**.

5.4.7 Копирование и вставка группы

Чтобы скопировать выделенную группу, выберите **Копировать (Copy)** из всплывающего меню **ГРУППА (GROUP)** или **Копировать (Copy)** из меню **Правка (Edit)**. Когда группа скопирована, все выделенные блоки (включая все имена сигналов и параметров и уставки) и все выделенные соединения, копируются во внешний буфер. Чтобы вставить содержимое буфера необходимо снять выделение с выделенной области. Выберите Вставить из меню **НОВЫЙ (NEW)** или **Вставить (Paste)** из меню **Правка (Edit)**.

5.4.8 Копировать выделенную область в буфер

Эта опция используется для копирования выделенной части FBD в буфер для переноса ее в другие приложения в формате метафайла. Это достигается выбором пункта **Копировать в буфер (Copy to clipboard)** из меню **Правка (Edit)** в панели меню. Если необходимо скопировать элементы в **черно-белом** формате, соответствующую опцию следует выбрать из меню **Свойства (Preferences)**.

5.4.9 Центрирование выборки

Для перемещения выделенной группы в центр экрана, выберите опцию **Центрировать выборку (Center selection)** из меню **Правка (Edit)**. Функциональная клавиша F9 является горячей клавишей для этой функции.

5.4.10 Печать выборки

Если выделена выборка и выбран пункт **Печатать (Print)** из меню **Файл (File)**, напечатана будет только выделенная область. Так же, следует выбрать опцию **Масштабировать выборку на одну страницу (Scale output to one page)** во вкладке **Опции (Options)** в диалоговом окне Печатать (Print), в противном случае размер выведенной на печать области будет неверным.

5.4.11 Макро-команды в блоках комментариев.

Перед сохранением группы в виде макроблока, программист может добавить макрокоманду, которая будет выполнена при загрузке макроблока. Типичное использование этой опции - замена установленных по умолчанию текстовых строк в названиях сигналов, сетевых адресах или именах SNVT-переменных, на другой определенный текст. Макрос создается введением следующих команд в блоке комментария:

\$MESSAGEBOXEnter
text\$

Появится окно, содержащее сообщение "Enter text". Нажмите **OK** для продолжения.

\$REPLACEText\$

Появится окно замены. Введите текст и щелкните **OK**, чтобы найти введенный текст "Текст" в комментариях, названиях блоков и параметрах блоков и заменить его на новый введенный текст.

\$REMINDEREnter
text\$

Появится окно сообщения, содержащее сообщение "Enter text". Нажмите **OK** для продолжения. Окно напоминания появится не только при первой загрузке макроблока, но каждый раз при открытии макроса прикладной программой.

\$PLAYSOUND-
WAV.WAV\$

Звук в формате wav.wav будет проигрываться не только при первой загрузке макроблока, но каждый раз при открытии макроса прикладной программой.



Обратите внимание!

- Макро-команды исполняются в порядке сверху вниз.
- Первая команда Заменить (Replace) заменит, во всех блоках, первый совпадающий текст, вторая команда Заменить заменит второй совпадающий текст и т.д.

5.4.12 Сохранение макро-блока

Чтобы сохранить группу блоков и соединений в виде макроблока во всплывающем меню **ГРУППА (GROUP)** выберите пункт **Сохранить (Save)**. При выборе этой опции, появится стандартное окно сохранения файла с запросом имени, расширения (.MTA/.AUT), пути и диска файла, где группа будет сохранена.



Обратите внимание!

Только полностью выделенные соединения сохраняются в макроблоке, смотри *"Выделение группы"* на стр. 73.

Когда выделенная группа блоков сохраняется как MTA/AUT файл, сохраняются только определения блоков и соединения.

Сохранение выделенной группы блоков как AUT файла не одно и то же, что сохранение блоков через команду **Файл - Сохранение (File - Save)**. В первом случае, только информация, хранящаяся в выделенных блоках и соединения включаются в сохраняемый файл, в то время как, во втором случае множество другой информации такой, как величины констант, данные спецификации (автор, тип и т.д.) включаются в файл.

5.4.13 Загрузка макроблока

Использование всплывающего меню *НОВЫЙ*

Ранее сохраненные макроблоки загружаются с диска через меню ***НОВЫЙ***, выбором пункта ***Загрузка группы (Load group)***. При выборе этого пункта появляется стандартное окно открытия файла, позволяющее выбрать имя и путь файла макроблока. Возможно выбирать между старым макроблоком (.МСВ) и прикладной программой (.МТА/.АУТ). Группа будет загружена в месте, где находился курсор мыши при вызове всплывающего меню. При загрузке .МТА/.АУТ файла, который был сохранен как законченная прикладная программа, загрузятся только FBD и имена констант, и никаких других данных спецификации. Новые константы добавляются в таблицу констант, но если константа уже существует, старое значение сохраняется (Это применимо как для общедоступных, так и для локальных констант).

Путь расположения для пункта ***Загрузка группы (Load group)*** по умолчанию, т.е библиотека макроблоков, может быть определен в пункте ***Параметры (Settings)*** в меню ***Свойства (Preferences)***. Изменить эту информацию можно при последующих сеансах.



Обратите внимание!

Только имена констант, но не их величины, хранятся в файле макроблока (.МСВ). Таким образом, величины констант, используемые в макроблоке, не передаются, когда макроблок вставляют в прикладную программу. Если загружен макроблок, содержащий константы, новые константы автоматически не добавляются. Вместо этого при компиляции FBD появляется сообщение "Неопределенная константа" ("Undefined constant"). Также можно создавать .АУТ файлы через команду ***Сохранить группу (Save Group)***.

Использование проводника Windows

В качестве альтернативы, ранее сохраненные макроблоки могут быть загружены с диска через проводник Microsoft® Windows Explorer. Используйте его для загрузки макроблока из библиотеки в FBD.

5.5 Иерархические функциональные блоки (HFB)

5.5.1 Создание HFB

В зависимости от ситуации, можно создать HFB двумя путями:

- Вы можете создать HFB, а затем вставить функциональные блоки и соединения в HFB. В этом случае, выберите **HFB** в меню **НОВЫЙ (NEW)** и назовите HFB. Затем откройте **HFB** через **Раскрыть HFB (Expand HFB)** из всплывающего меню **БЛОК (BLOCK)**.

- Если у Вас уже есть блоки и соединения, которые Вы хотели бы объединить в HFB, очертите курсором прямоугольник вокруг функциональных блоков и выберите **Создать HFB (Create HFB)** из всплывающего меню **ГРУППА (GROUP)**. Пожалуйста, обратите внимание, что Вы не можете отменить команду "Create HFB". По этой причине, рекомендуется сохранить Вашу прикладную программу до создания HFB.

5.5.2 Название HFB и соединений

Назначение имени HFB и соединений осуществляется также, как назначение имени блоков. Тем не менее, сигналы HFB не могут быть определены как общедоступные.

Соотношение имени с HFB осуществляется в диалоговом окне **Правка (Edit)**, которое автоматически появляется при создании иерархического функционального блока. При закрытии диалогового окна нажатием кнопки "Отмена", создание HFB не произойдет. Возможно создать более чем один уровень иерархии блока HFB, это означает что можно вставить один или более HFB в существующем HFB. Поле в строке состояния внизу прикладной программы показывает какой уровень иерархии FBD отображен.

Блок HFB может иметь несколько входов и выходов, которые должны иметь уникальные имена в текущем уровне иерархии HFB. Эти типы входа и выхода блока называются "HFAI" или "HFAO" (аналоговые) и "HFDI" или "HFDO" (цифровые) соответственно. Входы и выходы отображаются только в развернутом виде.



Обратите внимание!

Эти I/O-блоки только графическое представление в TAC Menta. Они не загружаются в TAC Xenta, а значит эти сигналы не читаются в TAC Vista или панели оператора TAC Xenta.

При создании соединений в HFB с помощью команды **HFB I/O** из всплывающего меню **НОВЫЙ (NEW)**, пожалуйста, обратите внимание, что необходимо раскрыть блок HFB до создания его соединений. Если блок HFB создается очерчиванием курсором

существующих блоков с последующим выбором команды **Создать HFB (Create HFB)**, то только соединения частично находящиеся в выделяемой области будут отображены в HFB с именами по умолчанию; заглавные буквы для аналогового типа сигналов и прописные буквы для цифрового типа сигналов.

При создании или переименовании модуля, содержащего HFB, блок HFB и все нижние уровни будут принадлежать этому модулю.

5.5.3 Раскрытие и свертывание блока HFB

Внутренняя структура функциональных блоков блока HFB отображается при выборе команды **Раскрыть HFB (Expand HFB)** из всплывающего меню **БЛОК (BLOCK)** после щелчка правой кнопкой мыши на HFB. Раскрытый HFB сворачивается снова при выборе команды **Свернуть HFB (Compress HFB)** из всплывающего меню **НОВЫЙ (NEW)**.

5.5.4 Вывод на печать HFB

Выбором команды **Печатать (Print)** в меню **Файл (File)** осуществляется вывод на печать блока HFB. В диалоговом окне вывода на печать необходимо поставить галочку в поле "FBD". Если Вы выберете пункт "Все уровни" ("All levels"), то на печать будут выведен каждый иерархический уровень HFB на разных страницах с указанием их имени и номера станицы внизу страницы. Например, если имя блока HFB - "ALARMS", номера страниц будут "ALARMS1.1" и "ALARMS1.2" соответственно, если распечатка блока занимает две страницы.

При выборе пункта "Текущий уровень" ("Current level"), на печать будет выведен текущий иерархический уровень HFB.

5.6 Таблица констант

Константы прописываются в таблице констант. В таблице можно производить добавление, изменения, удаление или просто просмотр. Таблицу констант можно открыть выбрав пункт **Таблица констант (Constants table)** в меню **Опции (Options)** в панели меню.

5.6.1 Добавление константы

При редактировании функционального блока, идентификаторы могут быть определены как параметры, пока еще незадаанные в таблице констант. Затем, при закрытии диалогового окна создания блока нажатием кнопки **ОК**, появится диалог редактирования констант. В этом окне константа появится в поле **Идентификатор (Identifier)** с установленным по умолчанию значением 0. Пользователь может изменить значение и задать физическое устройство, в котором значение параметра измеряется.

Следует также задать будет ли константа общедоступной, т.е. будет ли она отображаться в **Программной спецификации (Program specification)**.

Задаваемые параметры констант такие же, как и у блоков. В поле Описание (Description) можно ввести описание константы. Другая возможность добавить константу в таблицу - выбрать пункт **Добавить... (Add...)** в окне таблицы констант. Появится диалоговое окно редактирования констант как было описано выше, но в этот раз с пустым полем **Identifier**.

5.6.2 Редактирование констант

Для изменения какой-либо константы в таблице констант, дважды щелкните левой кнопкой мыши на константе. Это действие откроет окно редактирования выбранной константы, в котором все поля, кроме *Имя (Name)*, могут быть изменены. Константы можно добавлять и в *режиме Редактирования (Edit)* и в *режиме имитации (Simulate)*.

5.6.3 Удаление константы

Чтобы удалить константу из таблицы, сначала выберите ее нажатием левой кнопки мыши на ней. Выбранная строка будет отображаться в инверсном виде. При нажатии кнопки *Удалить (Delete)*, выбранная константа будет удалена из таблицы.

5.7 Таблица конфигурации I/O

Таблица конфигурации I/O открывается при выборе пункта *I/O Configuration Table* в меню *Опции (Options)* в строке меню. В этом окне можно изменять, задавать и просто просматривать "связывание" входов/выходов. Таблица содержит список всех I/O сигналов в FBD.

В таблице 4 столбца: название сигнала, тип блока I/O, а также тип и адрес внешнего сигнала к которому сигнал подсоединен (физический вход/выход, сетевая переменная и т.п.). Есть также и столбец *Override*.

Сигналы сортируются в алфавитном порядке. Возможно пересортировать список нажатием на заголовке каждого столбца. При нажатии на заголовке столбца "Название", список отсортируется в соответствии с представленными в нем данными в возрастающем порядке. При повторном нажатии на том же заголовке список отсортируется в обратном порядке.

Редактирование "связывания" сигнала I/O можно осуществить, либо щелкнув один раз левой кнопкой мыши на одной из строк в таблице конфигурации I/O, чтобы ее выбрать, а затем - по кнопке *Связь (Bind)*, либо двойным щелчком левой кнопки мыши на любой из строк таблицы. При этом откроется окно Редактирования связи I/O выбранного сигнала, в котором параметры "Связывания" могут быть изменены.

В зависимости от типа I/O блока и типа связи существует ряд различных окон Редактирования связи I/O. Каждый тип связи также имеет свои собственные уставки конфигурационных параметров, пожалуйста обратитесь к справочному разделу данного руководства.

**Обратите внимание!**

-Когда Вы выбираете вход/выход в меню выбора входа/выхода, знак "#" после адреса входа/выхода показывает, что вход/выход уже используется.

- Имена констант не могут быть заданы в конфигурационных параметрах входа/выхода.

5.8 Таблица расписаний

Чтобы открыть окно с таблицей всех расписаний в FBD выберите команду Таблица расписаний в меню *Опции (Options)* в строке меню. Записи в *Таблице расписаний (Time Schedule Table)* отсортированы в алфавитном порядке. Чтобы изменить какое - либо расписание в таблице, дважды щелкните левой кнопкой мыши на строке с расписанием, которое необходимо изменить, или выбрать расписание однократным нажатием кнопки мыши на строке и последующим нажатием кнопки *Правка (Edit)*. При этом появится диалоговое окно редактирования расписаний, в котором недельный таймер и таймер праздничных дней (годовой) можно изменять, дополнять, удалять или просто просматривать.

Расписание может быть изменено во время режима имитации, но число таймеров недельного и годового может быть изменено только в режиме *Редактирования*.

5.9 Таблица аварийных сообщений

Таблица аварийных сообщений открывается через команду *Таблица аварийных сообщений (Alarm Text Table)* в меню *Опции (Options)* в строке меню. При это открывается окно с перечислением всех блоков аварии и соответствующими текстами аварийных сообщений (также текст сообщений для TAC Vista). Столбцы с текстом аварийных сообщений могут быть отсортированы в любом порядке. Чтобы вставить текст аварийного сообщения, щелкните левой кнопкой мыши на строке с аварийным сообщением, которое необходимо редактировать, или выберите аварийное сообщение щелкнув один раз на строке сообщения и нажмите кнопку *Изменить (Change)*. При этом появится диалоговое окно Редактирования текста аварийных сообщений выбранного сообщения, в котором текст аварийного сообщения можно редактировать или просто просматривать.

5.10 Локальный ТрендЛог в TAC Xenta

Локальный тренд лог задается при создании прикладной программы TAC Xenta (т.е. .MTA-файл). Существует возможность задавать в приложении несколько каналов регистрации (логов).

Количество логов ограничено только общим размером приложения.

Для открытия в TAC Menta окна опеределения функции 'Локальный тренд лог в TAC Xenta' ('Local Trend Logging in TAC Xenta') выберите пункт **Тренд Лог (Trend Logs)** в меню **Опции (Options)**. Добавить тренд лог в существующее приложение можно двумя путями:

- В TAC Menta Вы можете экспортировать .MTA-файл из TAC Vista и добавить определение лога в TAC Menta. Вставьте приложение снова в TAC Vista через определение файл и загрузите его в текущее устройство TAC Xenta, или
- Создайте тренд лог объект в TAC Vista и сконфигурируйте его для доступного физического ресурса, который пока не используется для регистрации в .MTA-файле. Задайте лог для общедоступного сигнала и выберите интервал регистрации и размер регистрации. Эти новые данные обновятся в .MTA-файле при экспорте прикладной программы из TAC Vista.



Обратите внимание!

При добавлении атрибутов *регистрируемый сигнал (Logged Signal)*, *интервал регистрации (Log Interval)*, *размер регистрации (Log Space)*, *не останавливать запись при логической активации (No stop if logically activated)*, *очистка лога после активации (Clear Log at start)* или типа ведения записи (Log Type) (циклический/нециклический лог) в описание существующего лога, и загрузке их в устройство TAC Xenta, запущенная регистрация будет остановлена и регистрируемые данные будут утеряны. Об этом Вы получите предупреждение.

5.10.1 Выбор Тренд лога (Select Trend Log)

Максимальное количество логов (Max Number of Logs)	Вводится максимальное число тренд логов, которое возможно задать в текущем приложении (0–50). Это число не может быть изменено в TAC Vista. Не используемые тренд логи, т.е. тренд логи, которые не определены и не названы, будут названы "LOG<no.>", т.е. LOG04, LOG05 и так далее.
Имя (Name)	Представление имени тренд лога (смотри раздел ниже).
Описание (Description)	Представление описания тренд лога (смотри раздел ниже). Можно удалить определенный тренд лог сокращением ' Максимального количества логов ' ('Max number of logs').

5.10.2 Определение тренд лога (Trend Log Definition)

Диалоговое окно для определения выбранного тренд лога вызывается либо нажатием на кнопке **Правка**, либо двойным нажатием на выделенном тренд логе. Окно содержит следующие поля:

Имя (Name) Задается имя тренд лога (максимум 20 символов), например, 'TempRoom205'. Если импортируется файл, созданный в TAC Vista, лог и физический ресурс создаются с этим именем. Имя физического ресурса не может быть изменено в TAC Vista.

Описание (Description) Описание тренд лога (максимум 40 символов), например, 'Temperature logging room 205'.

Регистрируемый сигнал (Logged Signal)	Сигнал (Signal) Нажмите кнопку <i>Обзор (Browse)</i> чтобы выбрать необходимую точку регистрации, которая должна быть общедоступным сигналом (бинарным, целым или дробным) в текущем приложении.
Гистерезис (Hysteresis)	Минимальное изменение регистрируемой величины (целой или дробной), которое приводит к новому значению регистрируемой величины. Например: задайте гистерезис равный 0.5, чтобы игнорировать изменение температуры менее чем на +/- 0.5 °C. Если значение гистерезиса не введено, по умолчанию используется гистерезис равный 0.5. Если введено значение гистерезиса 0.0, все регистрируемые записи будут сохраняться даже, если значение регистрируемого сигнала на последующем шаге не отличается от значения на предыдущем шаге.
Интервал регистрации (Log Interval)	Вводится требуемый интервал регистрации (10 секунд – 530 недель), т.е. время между двумя регистрациями.
Размер регистрации (Log Space)	Вводится требуемый объем регистрации (ед. измерения: время)

Управление регистрацией (Control)

Активация (Activate)	Выберите <i>Ручное-Выкл. (Manual-Off)</i> , <i>Ручное-Вкл. (Manual-On)</i> или <i>Автоматическая (Automatic)</i> .
Время старта (Start Time)	При ручной активации <i>Ручное_Вкл. (Manual-On)</i> можно выбрать дату и время для старта регистрации (по умолчанию автоматически устанавливается текущее время на момент включения режима).
Стартовая переменная (Start Variable)	При автоматической активации (<i>Автоматическая</i>) можно выбрать управление регистрацией через стартовую переменную (Start Variable) (общедоступный бинарный сигнал) в текущем приложении. Выберите сигнал через кнопку <i>Обзор (Browse)</i> .
Не останавливать лог, если активировано логически (No stop if logically activated)	Поставьте галочку в этом поле, чтобы предотвратить остановку регистрации, которая была запущена логической переменной, даже если логическая переменная деактивировала регистрацию. Регистрация будет продолжаться до тех пор пока не будет остановлена вручную выбором <i>Ручное-Выкл.</i> или до тех пор пока лог не заполнится полностью.
Очистка лога после активации (Clear Log at Start)	Поставьте галочку в этом поле, если требуется очистка тренд лога в момент его активации.

Тип регистрации (Log Type)

Выберите один из 4 возможных типов регистрации. Циклический лог/сохранение означает, что запись лога будет продолжаться при полном заполнении размера регистрации, но при этом наиболее старые значения будут затираться новыми.

- *Циклический лог в TAC Xenta, циклическое сохранение в TAC Vista (Circular log in TAC Xenta, circular storage in TAC Vista)*
- *Циклический лог в TAC Xenta, нециклическое сохранение в TAC Vista (Circular log in TAC Xenta, non-circular storage in TAC Vista)*
- *Циклический лог в TAC Xenta без сохранения в TAC Vista (Circular log in TAC Xenta, no storage in TAC Vista)*
- *Нециклический лог в TAC Xenta без сохранения в TAC Vista (Non-circular log in TAC Xenta, no storage in TAC Vista)*

Размер лога в TAC Vista (Log Space in TAC Vista)

Если ранее было выбрано альтернативное сохранение лога в TAC Vista, определите требуемый размер регистрации в TAC Vista, например, 10 недель.

Получить если заполнено на % (Retrieve at % Full)

Определите процентное соотношение (например, 80%), при достижении которого будут выбраны данные лога из устройства TAC Xenta и сохранены в TAC Vista. TAC Xenta устанавливает соединение с TAC Vista при достижении заданного процентного соотношения.

Получить если заполнено на % и есть соединение (Retrieve if connected and %)

Определите процентное соотношение (например, 60%), при достижении которого будут выбраны данные лога из устройства TAC Xenta и сохранены в TAC Vista, при условии что TAC Xenta и TAC Vista находятся на связи по каким-либо причинам. При малой скорости опрашивания регистрируемые величины всегда будут извлекаться.

5.11 Установка даты и времени (Set Date and Time)

Пользователь может установить начальные значения даты и времени для режима имитации не-на-линии (нужно для работы тестовых функций связанных с датой и временем) с помощью команды *Установка даты и времени (Set Date and Time)* в меню *Опции (Options)*. Если установка осуществляется в режиме на-линии, время изменяется и в устройстве TAC Xenta.

5.12 Поиск и замена текста

Окно Поиска запускается через пункт **Поиск (Find)** в меню **Правка (Edit)**, или через функциональную клавишу F7. Поиск ведется в блоках и комментариях, содержащих определенный текст в названии блока, в типе блока или в параметрах. Первый блок, в котором будет найдено соответствие, будет выделен зеленым прямоугольником. Окно также содержит кнопки для поиска следующего совпадения, поиска целых слов и расширенный поиск.

Окно Замены запускается через пункт **Замена (Replace)** в меню **Правка (Edit)**, или через функциональную клавишу F8. Вы можете искать заданный текст в комментариях, названии блока или параметрах блока и можете заменить его на новый введенный текст. Первый блок, в котором будет найдено совпадение, будет выделен зеленым прямоугольником. Затем нажмите кнопку **Замена (Replace)** для замены текста или нажмите кнопку **Найти следующий (Find next)** для поиска следующего совпадения без замены текста. При замене, только первый найденный текст в блоке будет заменен, если выбран пункт *Замена только первого найденного текста (Replace only first appearance)*, в другом случае все совпадения в блоке будут заменены. Окно также содержит кнопки расширенного поиска, поиск по целым словам и замена совпадений во *всех* блоках (**Заменить все**).

**Обратите внимание!**

существуют следующие функции в окне Поиск и Замена:

- Выбор поля *Сигнал и Константы (Signal/Constant Names)* означает, что поиск текста ведется в части Сигнал идентификаторов блока, названий констант, сетевых адресов и имен SNVT.
- Выбор поля *Название Модуля (Module Names)* означает, что поиск ведется в части Модуль идентификаторов блока, названий констант, сетевых адресов и в имен SNVT.
- Выбор обоих полей *Название модуля и Сигнал и Константы* означает, что поиск ведется в части Сигнал и в части Модуль, например, АНУ01\TS, в идентификаторах блока, в названиях констант, в сетевых адресах и в именах SNVT. Обратите внимание, что тем не менее, в этом случае, изменение названия Модуля не возможно.
- Программой не осуществляется проверка длины текстовой строки в названии после выполнения команды *Замена (Replace)*. Поэтому, пожалуйста, внимательнее проверяйте, что *Название Модуля* не превышает 12 символов, а название *Сигнала или Константы* не превышает 20 символов, после выполнения команды **Поиск и Замена**.
- Если поле *Искать...(Search for...)* незаполнено, то поиск будет осуществляться во всех блоках выделенного типа (-ов)(Block Types).

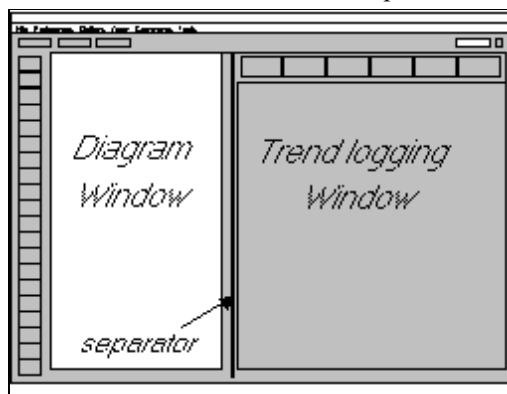
5.13 Отмена (Undo)

Команда **Отмена (Undo)** выполняется через пункт **Отмена (Undo)** в меню **Правка (Edit)**, или через горячие клавиши Ctrl-Z. Можно восстановить удаленные один или более выделенных функциональные блоки, включая соединенные с ними узлы. Если область удаленных функциональных блоков и соединенных с ними узлов, очень велика, чтобы уместиться на экране, будет показан нижний левый угол этой области.

Командой **Отмена (Undo)** возможно сделать откат на один шаг назад. Если выполнялась другая команда, исключая **Файл/Сохранить**, команда **Отмена (Undo)** не действует. Например: если Вы удалили функциональный блок и затем, например, раскрыли иерархический функциональный блок, вы не можете восстановить удаленный функциональный блок.

6 Режим Имитации (Simulate)

Создав и правильно спроектировав FBD, можно перейти из *режима редактирования (Edit)* в *режим имитации (Simulate)*, в котором прикладная программа может быть запущена на выполнение для анализа поведения сигналов. В режиме на-линии, поведение сигнала имитируется в реальном времени. В этом режиме работы окно приложения состоит из двух под-окон: *подокно блоков* и *под-окно тренд логов*:



Два под-окна отделены *разделителем*, который можно свободно перемещать для изменения пропорций окон. Для перемещения разделителя, щелкните левой кнопкой мыши на нем и удерживая кнопку нажатой двигайте курсор в сторону. При отпуске кнопки, размеры двух окон будут пересчитаны. Двойной щелчок левой кнопкой на разделителе осуществляет переключение между горизонтальным и вертикальным разделением под-окон. В *окне блоков* отображается либо FBD, либо таблица (*Табличный режим*) со всеми общедоступными сигналами и параметрами, отсортированными в алфавитном порядке. В окружении под-окон существует ряд кнопок. Кнопки в левой части используются для управления физическими входами, кнопки в верхней части - для управления выполнением прикладной программы.

6.1 Компиляция FBD

Для перехода от *режима редактирования* к *режиму имитации* необходимо выбрать пункт **Имитация (Simulate)** из меню **Опции (Options)** или нажатием функциональной клавиши F12. До изменения режима, текущий FBD автоматически сохраняется в файл AUTOSAVE.AUT, если эта функция (Auto Save .AUT File) выбрана в окне **Параметры (Settings)** в меню **Свойства (Preferences)**. В этом случае FBD сохраняется, даже если программа будет прервана.

При изменении режима, прикладная программа автоматически компилируется. Компилятор в первую очередь определяет те блоки, которые не имеют входов. Это важно для тестовых блоков

(см. 6.4.3 “Имитация с использованием тестовых блоков” на стр. 112), поскольку в противном случае эти блоки могут быть определены случайным образом по отношению к I/O блокам, и тогда сдвиг задержки выполнения между I/O блоком и тестовым блоком и наоборот не будет детерминированным. Всегда ставится один шаг задержки на выполнение.

Если при компиляции, которая может занять несколько секунд в зависимости от сложности прикладной программы, будут обнаружены ошибки, появится диалоговое окно указывающее причину ошибки. В тоже время, блок, вызвавший ошибку, будет выделен зеленым прямоугольником. Блок с ошибкой будет помещен в центр экрана автоматически.

6.2 Меню Опции

6.2.1 Строка меню

Строка меню режима имитации имеет 7 групп опций, каждая с различными подпунктами:

- Файл (File)
- Свойства (Preferences)
- Опции (Options)
- Масштаб (Zoom)
- Команды (Commands)
- Инструменты (Tools)
- Помощь (Help)

Файл (File)

Новый (New)	Закрывает текущую прикладную программу и открывает новый файл. Если текущий файл не был сохранен, появится вопрос о его сохранении пользователем.
Открыть (Open)	Загружает прикладную программу с диска. Файл прикладной программы на диске должен иметь расширение .AUT или.MTA. Текущий файл будет закрыт до открытия нового. Если текущий файл не был сохранен, появится вопрос о его сохранении перед открытием нового. Папка, предлагаемая для сохранения всегда та, которая использовалась в предыдущий раз.
Сохранить (Save)	Сохраняет текущую прикладную программу на диск с тем же именем, под которым она уже существует.
Сохранить как (Save As)	Сохраняет текущую прикладную программу на диск, но позволяет пользователю ввести новое имя файла. Меню панели оператора (.OPC) и связанные текстовые файлы (.DOC) также сохраняются с соответствующими новыми именами файлов.
Печатать (Print)	Выберите часть прикладной программы для распечатки: FBD, HFB, дерево меню панели оператора, связанные текстовые файлы и/или программная спецификация, и затем напечатайте на любом активном принтере.
Экспорт I/O списка (Export I/O List)	Сохраняет список входов/выходов текущей прикладной программы как текстовый файл формата ASCII (.TXT). Пользователю предлагается ввести имя файла.
Выход (Exit)	Выход из программы и возвращение в Windows™. Если текущий файл не был сохранен, до выхода из программы появится окно с предложением его сохранить.
Свойства (Preferences)	
Параметры страницы (Page Setup)	Выбор ориентации (Orientation), размера (Size), полей и заголовков (Page Header) для распечатки FBD. Параметры страницы также влияют на отображение окна FBD, где границы страницы, выводимой на печать, помечены синими линиями. Графический размер приложения, рабочая область окна FBD, определяется числом горизонтальных и вертикальных страниц.
Отображать только общедоступные идентификаторы (View Only Public Identifiers)	При выборе этой опции, только идентификаторы блоков, чьи сигналы являются общедоступными будут отображены в окне блоков. По умолчанию: отображать имена всех блоков.
Отображать значения сигналов (View Signal Values)	При выборе этой опции, значения сигналов будут показываться в блоках в FBD. По умолчанию: отображать значения сигналов.

Отображать блоки (View Diagram)	При выборе этой опции, FBD отображаются в окне блоков, иначе (<i>Табличный режим</i>) отображается таблица общедоступных сигналов и параметров, отсортированная в алфавитном порядке. Функциональная клавиша F5 используется для переключения между режимами.
Показывать строку состояния (Show Status Bar)	Поле в строке состояния внизу окна прикладной программы показывает какой иерархический уровень FBD отображается. Информация представлена в левой части строки, а положение курсора в координатах X- и Y (ячейки) (страница формата А4 в горизонтальном положении содержит 160 * 200 ячеек) представлено в правой части.
Автоматическая генерация дерева меню (Auto Generation of Menu Tree)	При выборе этой опции, команда <i>Генерировать (Generate)</i> или <i>Загрузить (Download)</i> автоматически начнет генерацию стандартного дерева меню панели оператора открытого FBD, включая все общедоступные сигналы.
Параметры (Settings)	<p>Эта команда позволяет пользователю определить путь к библиотеке модулей, который будет использоваться по умолчанию в командах <i>Загрузить группу (Load Group)</i> и <i>Сохранить группу (Save Group)</i>, и выбрать текстовый редактор для связанных текстовых файлов путем задания расширения файла, имени и пути расположения исполняемого файла редактора.</p> <p>Окно также содержит окошко метки для выбора необходимости сохранять текущий FBD в файл AUTOSAVE.AUT до перехода из режимаредктирования в режим имитации.</p> <p>Пользователь может выбрать какой коммуникационный порт (COM1, COM2 и т.д.) будет использоваться в этом сеансе, когда запущено множество сеансов TAC Menta.</p>

Опции (Option)

Правка (Edit)	Возвращение в <i>режим редактирования</i> . Функциональная клавиша: F12.
На-линии (Online)	Переключатель в <i>режим На-линии</i> , смотри 8.7 "Функции на-линии" на стр. 113 . Эта опция эквивалентна нажатию кнопки <i>На-линии</i> . Функциональная клавиша: F11.
Не обновлять константы (Skip Constants)	Выключение онлайн обновления общедоступных констант в режиме <i>На-линии</i> . Это действие эквивалентно нажатию кнопки <i>Не обнов. конст. (Skip Const)</i> .
Установка даты и времени (Set Date and Time)	Установка значений текущей даты и времени в режиме имитации не-на-линии (offline simulation) (разрешает тестирование функций прикладной программы, зависящих от даты и времени). Если текущая дата и время изменяются во время имитации, это не окажет влияния на начальные значения даты и времени, введенные в режиме Редактирования (Edit) . При запущенном режиме На-линии (Offline) , текущая дата и время будут считываться из подсоединенного устройства. Если пользователь затем изменит установки даты и времени в режиме <i>На-линии</i> , изменения будут сохранены в устройстве.
Конфигурация устройства (Device Configuration)	Эта таблица позволяет пользователю задать тип TAC Xenta и I/O модулей, используемых прикладной программой.
Программная спецификация (Program Specification)	Открывает окно, которое показывает программную спецификацию, включающую перечень общедоступных сигналов и параметров. В этом окне определяется название прикладной программы и время программного цикла. Текущий программный ID XIF файла также отображается в этом окне.
Таблица констант (Constants Table)	Показывает список констант. Список может быть изменен.
Таблица конфигурации I/O (I/O Configuration Table)	Открывает таблицу конфигурации I/O для задания и изменения связей блоков I/O с физическими входами/выходами или сетевыми адресами. Пользователь также вручную может удалять I/O сигналы (только в <i>режиме На-линии</i>).
Таблица расписаний (Time Schedule Table)	Открывает таблицу для изменения расписаний. В режиме имитации, не возможно добавлять недельные и годовые таймеры.
Таблица аварийных сообщений (Alarm Text Table)	Здесь задается или изменяется текст аварийных сообщений, который содержится в сетевых аварийных сообщениях, генерируемых при срабатывании аварийных блоков.

Тренд логи (Trend Logs)	Открывает окно для задания локальных тренд логов в TAC Xenta.
Системная информация (System Program)	Считывает <i>Имя системы программы (System program Name)</i> и <i>Версию (Version)</i> , <i>версию оборудования (Hardware version)</i> и <i>версию PROM (PROM version)</i> из подсоединенного устройства TAC Xenta. Эта команда доступна только в <i>режиме На-линии</i> .
Использование памяти (Memory Usage)	Считывает рассчитанную величину использования памяти в текущем приложении. Окно показывает итоговую емкость используемой памяти (даже разделенную на разные типы загружаемых файлов), доступной памяти и свободной памяти в устройстве TAC Xenta. Для получения наилучшей оценки, выберите <i>Команды - Генерировать (Commands – Generate)</i> .

Масштаб (Zoom)	
Увеличение (Zoom In)	При выборе этого пункта курсор мыши принимает форму линзы, и в окне FBD можно очерчивать прямоугольную область, которая будет увеличена. Если после выбора Увеличение (Zoom in) щелкнуть в точке в окне FBD без очерчивания области, выбранная точка будет перемещена в центр экрана в нормальном масштабе. Горячие клавиши: Ctrl I.
Уменьшение (Zoom Out)	При выборе этого пункта размер FBD уменьшается до наименьшего возможного уровня детализации. Горячие клавиши: Ctrl O.
Нормальный (Normal)	Возвращение к нормальному масштабу.
Команды (Commands)	
Сброс (Resest)	Сброс прикладной программы в ее начальные значения и установка счетчика цикла в ноль. Эквивалент - нажатие кнопки RST . Функциональная клавиша: F10.
Шаг (Step)	Выполнение одного программно цикла. Тоже самое, что нажатие на клавишу Шаг (Step) . Функциональные клавиши: F3.
Выполнение (Execute)	Старт на выполнение прикладной программы, тоже что нажатие кнопки Выполнение (Execute) . Функциональная клавиша: F2.
Выполнение n раз (Execute n Times)	Старт на выполнение прикладной программы, но позволяет пользователю задать количество раз выполнения программно цикла.
Выборка (Sampling)	Открытие диалогового окна, в котором может быть определено число программных циклов в выборке регистратора тренд-лога.
Генерировать (Generate)	Генерировать исполняемый код. Файл спецификации (.ESP) и файл исполняемого кода (.COD) генерируются всегда. Если выбрано, и если приложение имеет файл меню панели оператора (.OPC), то новый .OPC файл и .BIN файл (файл исполняемого меню панели оператора) могут быть сгенерированы. Файл .XIF генерируется, если SNVT включены в прикладную программу.
Загрузка в... (Download)	Загрузка прикладной программы и данных с компьютера в контроллер через последовательный коммуникационный порт. Эта команда доступна только в режиме На-линии.
Загрузка из... (Upload)	Загрузка прикладной программы из контроллера в компьютер через последовательный коммуникационный порт. Эта команда доступна только в <i>режиме На-линии</i> .
Холодный старт (Cold Start)	Производится холодный старт подсоединенного устройства. Эта команда доступна только в режиме На-линии.
Горячий старт (Warm Start)	Имитация горячего старта прикладной программы в режиме имитации, или выполнение горячего старта подсоединенного устройства, находящегося в режиме <i>На-линии</i> .

Инструменты (Tools)

Текстовый файл (Text File)	Вызывает текстовый редактор для редактирования связанных текстовых файлов, см. 8.2 "Связанные текстовые файлы" на стр. 108.
Конфигурация ОП (OP Configuration)	Вызывает инструмент конфигурации панели оператора, см. 7 "Программа конфигурации панели оператора" на стр. 89.
Регистратор (Logger)	Вызывает регистратор, см. 6.6.1 "Старт Регистратора" на стр. 86.
Обновить окно (Redraw Screen)	Обновить окно.

Помощь

Содержание (Contents)	Вызывает Help-файл с основной информацией из этого руководства. Функциональная клавиша: F1.
TAC в интернете (TAC on the Web)	Посещение домашней страницы TAC в интернете.
О программе (About TAC Menta)	Появляется окно с краткой информацией о программе TAC Menta.

6.2.2 Всплывающие меню

Всплывающие меню появляются при щелчке правой кнопкой мыши на окне FBD или на окне тренд лога, и исчезает при щелкании левой кнопкой мыши. Также меню исчезает, если щелкнуть на его заголовке или вне меню. Если один из пунктов меню выбран с помощью курсора мыши, запустится соответствующая опция. Всплывающее меню зависит от *объекта* на котором было открыто.

СИГНАЛ (SIGNAL)

Появляется при щелкании на блоке или соединении в окне блоков, или на строке сигнала в *табличном режиме*.

Изменить (Modify)	Открывает окно для изменения значения сигнала. Тоже что и щелкание левой нопкой мыши на блоке или соединении.
Просмотр (View)	Открывает окно редактирования блока прикладной программы, которому соответствует сигнал, но не позволяет производить изменения в полях; они доступны только для просмотра. Тем не менее, "связывание" входов/выходов и уставки расписания могут быть изменены из режима имитации.
Запись (Record)	Добавляет сигнал в регистратор тренд лога в первое пустое поле записи. Если сигнал аналоговый, при первом открытии окна ЗАПИСЬ (RECORD) запрашивается диапазон сигнала, который будет записан.

Удалить (Delete)	Удаляет сигнал из регистратора (сигнал не будет записываться).
Выделить (Mark)	Подсвечивает выбранное соединение и все его ответвления, отображая их в другом цвете. Доступно 6 цветов, из которых можно выбирать.
Снять выделение (Unmark)	Возвращение к нормальному режиму отображения выделенных ранее соединений.
Снять выделение со всех (Unmark All)	Возвращение к нормальному режиму отображения всех соединений.
Раскрыть (Expand)	Показывает внутреннюю структуру функциональных блоков HFB.

ЗАПИСЬ (RECORD)

	Появляется при щелкании на одном из шести полей записей, которые появляются в окне тренд логов при записи сигналов.
Изменить (Modify)	Открывает окно изменения значения регистрируемого сигнала. Это окно также можно вызвать щелканием левой кнопкой мыши на записи.
Изменить диапазон (Edit Range)	Открывает окно, в котором пользователь может изменять диапазон сигнала, выводимого в окне тренд логов. Изменение диапазона не отражается на ранее выведенном графике сигнала.
Удалить (Delete)	Удаляет сигнал из перечня записей.
Останов по пределу (Stop at a limit)	Устанавливается предел (контрольная точка), по достижении которого выполнение прикладной программы останавливается.

ТРЕНД (TREND)

	Появляется при щелкании на графиках тренд лога.
Очистить (Clear)	Удаляет графики из тренд лога.
Рестарт (Restart)	Рестарт рисования графиков с начала без удаления предыдущих графиков.
Сброс (Reset)	Удаление всех регистрируемых сигналов из регистратора.

УПРАВЛЕНИЕ (NAVIGATE)

	Появляется когда в окне FBD отображается раскрытый HFB.
Свернуть HFB (Compress HFB)	Свернуть раскрытый HFB.

6.3 Выполнение прикладной программы

Выполнение прикладной программы в устройстве TAC Xenta состоит из бесконечного цикла, выполняемого за постоянный

временной интервал. В каждой итерации (повторении) цикла (*программного цикла*) все выходные сигналы функциональных блоков обновляются. В TAC Menta у пользователя есть возможность выполнять прикладную программу по шагам, имитируя действия от других узлов сети и, наблюдать изменения всех сигналов, чтобы удостовериться в правильности работы прикладной программы.

6.3.1 Старт, стоп и инициализация выполнения

Запустить прикладную программу на выполнение можно несколькими способами:

Непрерывное выполнение (Continuous Execution)

Нажмите кнопку **Выполнение (Execute)** или функциональную клавишу F2 для непрерывного выполнения прикладной программы. Оно не прекратится до тех пор, пока эта операция не будет повторена еще раз.

Пошаговое выполнение (Step by Step Execution)

Нажмите кнопку **Шаг (Step)** или функциональную клавишу F3 для выполнения одного цикла прикладной программы.

Выполнение n циклов (Execute n Cycles)

Выберите **Выполнить n циклов... (Execute n Times...)** из меню **Команды (Command)**, при этом появиться окно в котором необходимо указать количество выполняемых циклов. Окно закрывается нажатием клавиши Enter.

При непрерывном выполнении прикладной программы, оно будет приостановлено если любая из кнопок **Выполнить (Execute)**, **Шаг (Step)** или **RST** или соответствующие им функциональные кнопки будут нажаты.

В правом верхнем углу окна тренд логов отображается количество выполненных циклов прикладной программы со времени последней реинициализации. Нажмите кнопку **RST** или функциональную клавишу F10, счетчик сбросится в ноль и все сигналы прикладной программы вернуться в их начальные значения.

6.3.2 Операции с сигналами

В любой момент имитации, пользователь может изменить значение сигнала, что эквивалентно действию вышестоящего узла. Не все сигналы прикладной программы могут быть изменены; только RW (чтение/запись), но не RO (только чтение), могут быть изменены.

Чтобы изменить сигнал, щелкните левой кнопкой мыши на блоке или соединении, которые содержат сигнал. В зависимости от того выбран ли аналоговый или бинарный сигнал, окно **ИЗМЕНИТЬ (MODIFY)** или **ЗАДАТЬ (ORDER)** появится соответственно. В окне **ЗАМЕНИТЬ (MODIFY)**, текущее значение сигнала, которое появляется в окне, может быть заменено новым значением, набранным на клавиатуре, чтобы изменения были приняты

необходимо нажать клавишу Enter. В окне ЗАДАТЬ (ORDER), сигнал может принимать два значения: логический ноль, который представляется пустым кружком, и логическая единица, которая представляется заполненным кружком. Щелкая курсором мыши внутри кружка можно изменять его статус. В любом из этих случаев, изменение не вступит в силу, пока окно не будет закрыто. Для этого нажмите клавишу Enter или щелкните на области вне окна. Тем не менее при закрытии окна нажатием клавиши Esc не приведет к изменению значения сигнала.

6.3.3 Выделение соединений

Чтобы выделить соединение (включая все его ветви), щелкните правой кнопкой мыши на соединении и из всплывающего меню **СИГНАЛ (SIGNAL)** выберите пункт **Выделить (Mark)**. Соединение будет выделено другим цветом, что позволяет проследить его через FBD. Чтобы вернуться к нормальному режиму отображения соединений щелкните правой кнопкой мыши на соединении и во всплывающем меню **СИГНАЛ (SIGNAL)** выберите пункт **Снять выделение (Unmark)**.

6.3.4 Режим моделирования исполняемого файла

Выполняемый файл может быть загружен в режиме имитации через команду **Файл-Открыть (File - Open)**, выбрав файл .COD. Файл может быть смоделирован в режиме не-на-линии, но *только в табличной форме*. Исходный код не может быть изменен.

6.4 Имитация внешних входов

Пользователь имеет несколько возможностей имитировать поведение внешних входов контроллера в TAC Menta во время выполнения прикладной программы.

Входные сигналы управляются входными кнопками, расположенными в левой части окна блоков. Каждый входной блок в прикладной программе представлен кнопкой, на которой показано название соответствующего входного блока. TAC Menta поддерживает произвольное число входных блоков в прикладной программе. Если количество используемых входов превышает доступное место на экране, то к панели кнопок физических входов добавляется полоса прокрутки.

Имитированные значения генерируемые кнопками входов не зависят от конфигурационных параметров входов/выходов, например, параметр **Нормально открытый (Normally Open)** в блоке DI не учитывается при имитации. Исключение составляет параметр **Постоянная времени (Time Constant)** в блоке AI, который будет имитировать тот же фильтр, что и в устройстве TAC Xenta в автоматическом режиме (но не в ручном режиме).

6.4.1 Ручной режим управления

Ручной режим - это режим управления физическими входами, выбранный по умолчанию, он заключается в задании вручную значения на входе. Каждый раз, когда происходит изменение входного значения, оно после изменения остается постоянным, пока вновь не будет изменено.

Кнопки, которые представляют физические цифровые входы действуют как переключатели: они переключаются между разомкнутым и замкнутым состоянием каждый раз при нажатии левой кнопкой мыши на кнопке физического входа.

При щелкании левой кнопкой мыши на кнопках, представляющих аналоговые входы, открывается диалоговое окно. В этом окне задается величина сигнала, воспринимаемая физическим входом в *инженерных единицах измерения* (например, С°). Величина должна быть введена в тех же единицах измерения, что и выходной сигнал блока аналогового входа. Чтобы применить новое значение, окно должно быть закрыто нажатием кнопки Enter или щелчком вне области окна.

6.4.2 Автоматическая генерация вектора входных значений

В данном режиме управления, физические входы могут быть заданы произвольными входными функциями, которые изменяются во времени при выполнении прикладной программы. Чтобы выбрать автоматический режим, щелкните правой кнопкой мыши на кнопке физического входа; в зависимости от того аналоговый или цифровой вход, откроются разные диалоговые окна.

Генерация аналоговых колебаний

Для аналоговых входов существует два варианта выбора режима управления, ручной и автоматический. В ручном режиме (Manual), поведение входного сигнала соответствует описанному выше. Если выбран автоматический режим (Automatic), в диалоговом

Форма колебания (Wave Form)	окне необходимо будет задать ряд полей. Колебательный процесс, который может быть задан содержит следующие параметры: Выберите постоянное значение (continuous), меандр (square), синусоидальную (sinusoidal), треугольную (ramp), пилообразную (saw) или импульсную (impulse) форму.
Амплитуда (Amplitude)	Здесь задайте последовательность значений амплитуды, разделенных запятыми. Во время выполнения прикладной программы в первом периоде амплитуда колебаний будет равна первому значению, во втором - второму значению и так далее последовательно. <i>Амплитуды задаются в тех же единицах измерения, что и выходной сигнал блока аналогового входа.</i>
Циклический (Circular)	Если в этом поле стоит галочка, то когда во время выполнения программы пройдет столько циклов, сколько значений амплитуд задано в описанной выше последовательности, цикл продолжится с начала последовательности. Тем не менее, если эта опция не выбрана, по окончании первого цикла последовательности, управление автоматически перейдет в ручной режим.
Среднее значение (Average Value)	Выберите среднее значение для колебательного процесса. <i>Среднее значение задается в тех же единицах измерения, что и выход блока аналогового входа.</i> Например, если среднее значение равно 50 и амплитуда равна 10, синусоидальное колебание будет изменяться в диапазоне от 40 до 60.
Фаза (Phase(degrees))	Это смещение во времени колебания относительно оси. Измеряется в градусах. Полный период равен 360. Например, синусоидальное колебание с фазой равной 90 эквивалентно косинусоидальному колебанию.
Период (Period)	Задайте длительность полного периода колебания, измеряемого в секундах.

Генерация бинарных последовательностей

Для цифровых входов, также существует два режима управления, ручной и автоматический. В ручном режиме входной сигнал ведет себя как было описано в [6.4.1 "Ручной режим управления" на стр. 83](#). Если выбран автоматический режим (Automatic), необходимо рассмотреть некоторые поля в диалоговом окне.

Последовательность (Sequence)	<p>Последовательность, которую можно задать определяется следующими параметрами:</p> <p>Здесь необходимо задать в какой последовательности будут следовать нули и единицы (например, 00000110100011). Каждый элемент последовательности имеет продолжительность равную заданному периоду.</p>
Цикличность (Circular)	<p>При выборе этой опции, если во время выполнения программы последовательность завершится, она будет повторена сначала. Тем не менее, если эта опция не выбрана, при окончании первого цикла, цифровой вход автоматически перейдет в ручной режим управления.</p>
Период (Period (ms.))	<p>Выберите продолжительность элемента последовательности, измеряемую в миллисекундах.</p>

6.4.3 Имитация с использованием тестовых блоков

Тестовые блоки для имитации используются для создания *вспомогательных* прикладных программ, которые имитируют систему, которой Вы хотите управлять, используя *основную* прикладную программу. Если известно приближенное функционирование управляемой системы, то есть возможность спроектировать прикладную программу, которая имитирует поведение этой системы. Тестовые блоки используются следующим образом:

- Основная прикладная программа считывает данные с блоков физических входов и управляет физическими выходами.
- Вспомогательная модель использует ТРАО/ТПДО блоки для считывания данных с физических выходов прикладной программы, и записывают рассчитанные моделью значения в физические входы используя блоки ТРАИ/ТПДИ.

Когда ТРАИ или ТПДИ блоки существуют в функциональных блоках диаграмм, входы/выходы, подсоединенные к этим блокам, могут не управляться кнопками входов ни в ручном, ни в автоматическом режимах. Тем не менее, кнопки могут быть использованы для считывания значений на входах в любой момент. Приложение, содержащее тестовые блоки, не может быть загружено в контроллер TAC Xenta.

6.5 Тренд логи

Регистрация сигналов позволяет пользователю записывать, во время имитации, во временной диаграмме изменения не менее шести сигналов прикладной программы одновременно. Аналоговые и бинарные сигналы могут записываться в регистраторе одновременно.

6.5.1 Добавление сигналов в регистратор

Чтобы записать сигнал, щелкните правой кнопкой мыши на соответствующем блоке и выберите пункт **Запись (Record)** из всплывающего меню **СИГНАЛ (SIGNAL)**. Если сигнал аналоговый, откроется диалоговое окно **ЗАПИСЬ (RECORD)**, где задается диапазон сигнала. Запрашиваемые **максимальное и минимальное значения** соответствуют значениям, которые сигнал принимает соответственно в конце и начале оси ординат диаграммы. Диапазон по умолчанию: 0-100.

6.5.2 Удаление сигналов из регистратора

Существует два способа, чтобы остановить запись сигнала. Первый - щелкнуть правой кнопкой мыши на экране регистратора и в появившемся меню **ЗАПИСЬ (RECORD)** выбрать пункт **Удалить (Delete)**. Второй - щелкнуть правой кнопкой мыши на блоке, соответствующем сигналу, и в появившемся меню **СИГНАЛ (SIGNAL)** выбрать пункт **Удалить (Delete)**.

6.5.3 Очистка регистратора

Очистка регистратора означает только удаление графиков временных диаграмм, при этом записываемые сигналы остаются теми же. Для проведения очистки щелкните правой кнопкой мыши на временной диаграмме и во всплывающем меню **ТРЕНД (TREND)** выберите пункт **Очистить (Clear)**.

6.5.4 Рестарт регистратора

Рестарт регистратора означает, что рисование графиков начнется из начала координат, но без удаления ранее полученных графиков. Для рестарта щелкните правой кнопкой мыши на всплывающем меню **ЗАПИСЬ (RECORD)** и выберите пункт **Рестарт (Restart)**.

6.5.5 Сброс регистратора

Сброс регистратора означает удаление всех сигналов, которые записывались. Для этого удалите все сигналы один за другим используя процедуру, описанную в предшествующем разделе, или удалите их всех одновременно. Последнее достигается щелканием правой кнопкой мыши на временной диаграмме и выборе в появившемся меню **ТРЕНД (TREND)** пункта Сброс. До выполнения операции **Сброс**, пользователю необходимо подтвердить ее.

6.5.6 Просмотр тренд логов

Можно сканировать графики сигналов используя курсор и получать строб (выборку) в любой точке. Для этого просто щелкните левой кнопкой мыши в точке на графике. Курсор появится в выбранной точке в виде пунктирной вертикальной

линии; в тоже время, записи сигналов и счетчик цикла будут обновлены значениями, соответствующими значениям графиков сигналов в этой точке.

Нажав и удерживая левую кнопку при движении мыши влево или право получим следование курсора за движениями мыши, но за один раз курсор будет перемещаться на один шаг. Курсор тренд лога также может быть перемещен с помощью клавиш курсора влево и вправо, при нажатой клавише Ctrl.

Для удаления курсора с временных диаграмм, щелкните правой кнопкой мыши на диаграмме.

6.5.7 Задание временных выборок

По умолчанию, регистратор делает одну выборку значений для каждого регистрируемого сигнала в каждом программном цикле. Тем не менее, возможно задать число программных циклов, выполняемых для получения одной выборки значения сигнала в регистраторе, т.е. график будет продвигаться медленнее и медленно изменяющиеся процессы будет удобнее наблюдать. При выборе пункта **Выборка (Sampling)** из меню **Команды (Commands)** в строке меню откроется окно, в котором можно задать параметры. По умолчанию число программных циклов на выборку равно единице, что эквивалентно отображению каждого программного цикла в тренд логе.

В режиме На-линии, период выборки для тренд лога выражается в секундах вместо программных циклов, как в режиме *Не-на-линии*. Тем не менее, благодаря ограничениям в операционной системе такой период выборки не может быть очень точным.

6.6 Средства регистрации

Средства регистрации - это дополнение к функциям тренд-лога, которое позволяет производить дальнейший анализ регистрируемых величин, и сохранять их в файл. Регистрируемые сигналы те же, что и записываемые в тренд-логе.


6.6.1 Старт регистрации

Запись регистратора запускается в режиме имитации следующим образом:

1. Выберите Регистратор через команду Регистратор (Logger) в меню Инструменты (Tools).
2. Выберите сигнал для регистрации, см. [6.5.1 "Добавление сигналов в регистратор" на стр. 86](#).
3. Запустите программу на выполнение.

6.6.2 Настройка графиков

По средствам настройки временных диаграмм, Вы можете разделить окно Регистрации на несколько под-окон (диаграмм) и установить различные настройки для каждой диаграммы.

Нажмите основную кнопку настройки () , расположенную ниже строки меню регистратора.





В левой части появившегося окна располагается список диаграмм. Добавляйте или удаляйте диаграммы через кнопки **Добавить диаграмму (Add diagram)** и **Удалить диаграмму (Remove diagram)**, расположенные внизу формы окна.

Справа располагается список доступных сигналов. Задайте сигналы для диаграмм с помощью перетаскивания. Двойной щелчек на иконке диаграмм показывает сигналы, связанные с отдельными диаграммами. Затем отмеченные сигналы могут быть удалены из диаграмм через кнопку **Удалить сигнал (Remove signal)**.

Нажмите **ОК** по готовности.

Расположите диаграммы командами **Расположить по горизонтали (Tile)** или **Каскад (Cascade)** в меню **Окно (Window)**.


6.6.3 Представление и анализ регистрируемых величин

	Установите свойства для каждой диаграммы нажав кнопку настройки () в под-окне диаграмм (Diagram 1):
Легенда (Labels)	Пояснительные описания для осей X и Y.
Сетка (Grid)	Включени/выключение сетки.
Тип шкалы (Scale Types)	Выберите <i>Скольльзящая шкала (Slide scale)</i> () , <i>Растущая шкала (Grow scale)</i> () или <i>Фиксированная шкала (Fix scale)</i> () .
Координаты (Coordinates)	Установите минимальное и максимальное значения для осей X и Y.

Нажмите **OK** по выполнении всех операций.


Регистрируемая величина отображается в виде графа на диаграмме в соответствии в установками диаграммы. Название сигнала представлено в верхнем левом углу каждой диаграммы. Над полем диаграммы отображаются последнее записанное значение сигнала и номер программного цикла (выборки) для выбранного сигнала.

Если Вы хотите увеличить и выбрать часть диаграммы для более детального анализа, эту область следует выделить очерчиванием прямоугольника, удерживая при этом нажатой левую кнопку мыши. Масштаб диаграммы автоматически измениться, таким образом выбранная область будет занимать все окно диаграммы. Кнопки со стрелками в правой части диаграммы используются для перемещения настоящего участка вдоль оси X (число программных циклов).

Используйте кнопку масштабирования () , расположенную справа от диаграммы, чтобы вернуть масштаб диаграммы к установкам по умолчанию X = 0-1000, Y = 0-100.

При нажатии кнопки печати справа от диаграмме, текущая диаграмма будет напечатана на активном принтере.

6.6.4 Сохранение и просмотр Лог файла

Все регистрируемые величины могут быть сохранены в файл через кнопку сохранения () , или через команду Файл-Сохранить в

6.7 Изменение параметров блока в процессе имитации

6.7.1 Параметры функциональных блоков

Внутренние параметры блоков не могут быть изменены в режиме имитации. Значения могут быть проверены с помощью команды *Просмотр (View)* во всплывающем меню *СИГНАЛ (SIGNAL)*. Чтобы изменить общедоступные параметры блока в процессе имитации, пользователь должен войти в таблицу констант и изменить величины соответствующих общедоступных констант.

6.7.2 Временные расписания

Любые изменения в режиме имитации в информации TSCN будут сохранены при возвращении в режим редактирования, в независимости произведены изменения в блоке или через таблицу TSCN. В режиме На-линии, изменения в уставках недельного или годового таймеров блока TSCN будут переданы непосредственно в контроллер. Пожалуйста, обратите внимание, если число недельных и годовых таймеров изменилось, должны быть заново сгенерированы и загружены файлы .COD и .BIN.

6.7.3 Данные связывания I/O

Любые изменения в "связывании" входов/выходов в режиме имитации сохраняются при возвращении в режим редактирования. В табличном режиме, информация о "связывании" входов/выходов доступна только через таблицу "связывания" входов/выходов. После изменения данных "связывания" входов/выходов, необходимо заново сгенерировать и загрузить файл .COD в устройство.

6.7.4 Константы

Только общедоступные константы доступны в режиме имитации. Значения необщедоступных констант могут быть просмотрены, но не изменены в таблице констант. Значения общедоступных констант, измененные в таблице констант обновляются в исходном коде, но изменения в табличном режиме не возможно.

6.7.5 Аварийные сообщения

Аварийные сообщения доступны только через таблицу аварийных сообщений. Изменения всегда отражаются на исходном коде. После изменения аварийных сообщений, необходимо сгенерировать и загрузить в контроллер новый файл .COD.

6.7.6 Блок ошибок ERR

В режиме имитации возможно редактировать блок ошибок ERR для имитации различных внутренних сигналов или ошибок от системной программы. Для информации о блоке ошибок ERR см. [B.19 "ERR - системные ошибки"](#) на стр. 171.

7 Программа конфигурации панели оператора

Программа конфигурации панели оператора вызывается командой **Конфигурация панели оператора (OP configuration)** в меню **Инструменты (Tools)**. При выборе этой команды автоматически создается файл спецификации (.ESP) и вызывается программа конфигурации панели оператора с файлом .OPE для текущего приложения. Если этот файл не создан, появится предупреждающее сообщение и пользователю будет предложено выбрать либо файл по умолчанию (DEFAULT.OPE), либо какой-нибудь существующий. Файл .OPE будет использован как файл дерева меню по умолчанию. Нажатие **OK** без выбора другого .OPE файла, приведет к тому, что программа будет вызвана с полностью пустым деревом меню.



Дерево структуры меню	<p>Экран состоит из четырех основных областей:</p> <p>В нем отображается вся структура меню. Оператор может перемещаться по структуре меню, выбирая различные пункты. В зависимости от выбранного пункта различные части дерева в правой части экрана будут показывать разную информацию.</p>
Изменение названия пункта меню	<p>Для изменения названия пункта выберите кнопку Изменить название (Change Name). Для корневых пунктов меню (приложение) отображаются и название и аббревиатура, но ни то, ни другое нельзя изменить, пока они не будут определены в программной спецификации в прикладной программной части TAC Menta. Аббревиатура отображается во всех окнах, поэтому оператор всегда может видеть с каким устройством в сети он в настоящий момент работает. Аббревиатура также используется для указания суммарной аварии в TAC Xenta, текст будет мигать чтобы указать, что в списке аварий есть не подтвержденные аварии.</p>
Область экрана панели оператора (Operator Panel Display)	<p>Устанавливая курсор на области экрана текст можно изменять или добавлять. В зависимости от типа пункта меню различный текст будет копироваться на экран. Аббревиатура приложения всегда будет копироваться в левый верхний угол экрана. Также название пункта меню будет автоматически копироваться в верхнюю строку экрана при его создании.</p> <p>Для некоторых пунктов меню эта область также включает ряд кнопок управления для вставки справочной информации, имен и т.д. Существуют также кнопки управления для задания свойств справочной информации вставляемых сигналов.</p>
Список пунктов меню (список сигналов)	<p>В этой области появляется содержание пунктов меню (подменю) выбранного пункта меню, если он соответствует типу пункта подменю или корневого меню. Так же в этой области может отображаться список сигналов, используемых для вставки информации, названий и т.д. в экран панели оператора.</p>

7.1 Экран структуры меню

Различные типы пунктов меню представляются различными символами в дереве меню. Первые два типа пунктов можно сравнить с символами папок в Microsoft® Windows Explorer:



Пункт корневого меню (Menu root item), также название приложения, которое отображается на экране панели оператора TAC Xenta OP, при заданном сетевом устройстве.



Подменю (Sub menu) пункт меню, это тип пункта меню, которые может включать ряд других пунктов меню, формирующих подменю. Используется для создания иерархических уровней в структуре меню.

Другие типы пунктов, называемые экранными пунктами, можно сравнить с файлами в Microsoft® Windows Explorer, с содержанием "файла" представленным на экране панели оператора:



Статус (Status) пункт меню, этот тип включает ряд страниц/экранов отображающих традиционные сигналы и параметры, такие как значение уставки, измеряемые величины и т.д.



Авария (Alarm) пункт меню, под этим пунктом подразумевается список аварий.



Код доступа (Access Code) пункт меню, этот пункт ведет к экрану, где вводится код доступа.



Изменить код доступа (Edit Access Code) пункт меню, этот пункт ведет к экрану, где изменяется код доступа.



Дата и время (Date and Time) пункт меню, Этот пункт ведет к экрану, где задаются дата и время.



Переход на летнее время (Daylight Saving) пункт меню, этот пункт ведет к экрану, где вводятся параметры перехода на летнее время.





Недельный таймер (Week Chart) пункт меню, этот пункт меню включает ряд недельных таймеров, привязанных к одному расписанию.



Годовой таймер (Holiday Chart) пункт меню, этот пункт содержит ряд годовых таймеров, привязанных к одному расписанию.

7.2 Создание структуры меню

7.2.1 Добавление пунктов меню

Если выбрана ранее не сохраненная структура меню, то в программе конфигурации панели оператора при старте ее по умолчанию появляется только корневой пункт меню в экране структуры меню. Этот корневой пункт имеет название, которое совпадает с названием устройства TAC Xenta (приложения), используемого при представлении панелью оператора всех устройств в сети. Структура меню создается добавлением пунктов меню в корневое меню () и подменю (). Добавление пункта меню выполняется выбором корневого меню или подменю в


экране структуры меню, с дальнейшим выбором кнопки *Добавить (Add)* в области списка пунктов меню. При этом появится окно, в котором можно ввести название пункта меню и выбрать его тип. В зависимости от выбранного типа, будут созданы разные панели экрана. Когда пункт меню содержит подменю, при создании он отмечается символом "+" на иконке пункта подменю в экране структуры меню. При двойном щелчке на этой иконке, в экране структуры меню появляются пункты, входящие в подменю (см. также опцию меню *Дерево - Раскрыть (Tree - Expand)*).

**Обратите внимание!**

Дерево меню панели оператора максимально может содержать до 255 экранов панели оператора.

Для изменения порядка следования пунктов меню, перетащите их в необходимое место в окне списка пунктов меню. Чтобы удалить пункт меню, выделите его (подсветите) в списке пунктов меню и удалите кнопкой *Delete*.


7.2.2 Подменю (Submenu)

Простейший тип пункта меню - это подменю (). У него есть только два свойства; название и уровень доступа. Подменю может иметь пункты меню любого типа в своем списке, включая другие подменю. Тем не менее, максимальное число уровней подменю равно 15.

Подменю может содержать пункты меню разного уровня доступа. При повышении уровня доступа для подменю, уровень вложенных подменю также может быть изменен до того же или более высокого уровня доступа для этих подменю.

При понижении уровня доступа подменю, Вы также можете выбрать изменение уровней доступа для вложенных подменю до того же или более вывокого уровня доступа для этих подменю.

7.2.3 Статус (Status)

Тип пункта меню *Статус* (), используется для создания диалогового окна, в котором параметры и измеряемые величины отображаются и изменяются.

В окне *экрана панели оператора (Operator Panel Display)* выполняется фактическое редактирование формата экрана. Текстовые строки и свойства сигнала (параметры, измеряемые величины) могут быть вставлены в определенное место экрана, указываемое курсором, двойным щелчком курсора мыши на списке сигналов, или используя кнопки *Вставить модуль (Insert Module)* (включая выбранный разделитель), *Вставить название (Insert Name)*, *Вставить сигнал (Insert Signal)* и *Вставить ед. измерения (Insert Unit)*. Эти кнопки работают только в случае, если файл спецификации был загружен. Если курсор был

установлен правильно в окне *экрана панели оператора*, то в этом месте появиться сигнал, выбранный в списке сигналов, расположенном в нижнем правом углу. Используемые сигналы в списке сигналов отображаются прямым начертанием, а еще не использованные сигналы отображаются жирным шрифтом. Если свойства сигналов используются более чем в одном месте в структуре дерева меню, эти сигналы будут отображаться прямым начертанием.

**Обратите внимание!**

Отображаемый список содержит только сигналы, принадлежащие одному модулю, таким образом следует сначала выбрать модуль через окно прокрутки над списком сигналов. Используя различные кнопки "вставка" можно вставить названия, ед.измерения или величины сигналов в окно.

Командная клавиша **Свойства (Property)** используется для вызова окна свойств для подсвеченного сигнала в окне *экрана панели оператора*. В окне есть следующие уставки:

**Атрибуты панели
опреатора (Operator
Panel
Attributes)**

Используется для определения можно ли менять данный сигнал с операторской панели или можно его только наблюдать.

**Свойства параметров
сигнала (Signal
Reference
Properties)**

Эта опция применима только для дробных и целых величин. Свойство Количество разрядов (No. of decimals) используется для задания формата отображения сигнала. Это должно быть установлено в пределах от -4 до 4. Отрицательные числа означают, что величина будет изменяться с шагом равным 10 (Кол-во разрядов = -1), 100, 1000 или 10000 (Кол-во разрядов = -4) при изменении ее оператором.

Если дробная величина требует большего числа зарезервированных разрядов, последующие символы в строке экрана будут перезаписаны. Если значение не умещается в доступном пространстве до конца строки, оно будет показано в научном формате (например, 14.1E6).

Свойства *Минимум* или *Максимум* используются для установки пределов, которые должны быть в диапазоне (-32768, 32767), для изменения сигнала кнопками "+" или "-" с панели оператора. Эти пределы используются только когда оператор изменяет величину сигнала с панели оператора.

Значения по умолчанию для *Минимума*, *Максимума* и *Количества разрядов*, для использования со всеми новыми сигналами, устанавливаются через команду **Свойства сигналов (Signal Properties)** в меню **Форматы (Formats)**.


Выделенные сигналы в окне *экрана панели оператора* можно удалить используя кнопку **Удалить сигнал (Delete Signal)**.

Пункт меню *статус (Status)* может содержать несколько страниц. Используйте командную клавишу добавить страницу для создания дополнительных страниц с параметрами/значениями.

**Обратите внимание!**



Если пункт меню Статус удален, то соответствующие сигналы не будут помечены как неиспользуемые (жирный шрифт) в списке сигналов пока список не будет обновлен, например, через команду *Дерево - Обновить спецификацию (Tree - Update specification)*.

7.2.4 Авария (Alarm)

При создании пункта меню *типа Авария* (), по умолчанию создается экран с привязками к списку аварий. Замечание: даже если несколько пунктов типа Авария будут созданы, все они будут ссылаться на один и тот же список аварий. Никакие другие свойства кроме названия и уровня доступа не следует изменять для этого типа пункта меню. Текст сообщений, который будет появляться на экране панели оператора задается в TAC Menta.



7.2.5 Код доступа (Access Code)

Когда выбран пункт корневого меню в левой части экрана, можно изменить уровень доступа для этого пункта в списке пунктов меню. Один или более пунктов меню можно выбрать, а затем нажать кнопку *Уровень (Level)*, при этом появится окно задания уровня доступа (*низкий(зеленый), средний(желтый) или высокий(красный)*). Уровень доступа для каждого пункта в структуре меню выделяется цветом уровня и буквой (Н, С или В). Замечание: уровень пункта меню типа *Код доступа* всегда должен быть задан как низкий.



Пункт меню типа *Код доступа* () и *Изменить код доступа (Edit Access Code)* () используются для создания диалоговых окон для выбора уровня доступа и кода доступа. Для этих типов пояснительный текст на экране может быть изменен. Это можно сделать поместив курсор (используя мышь) в окно *экрана панели оператора* и перезаписав новый текст.

В каждом устройстве TAC Xenta по умолчанию коды следующие *средний* = 1111 и *высокий* = 2222, они могут быть отредактированы через панель оператора. Замечание: TAC Xenta будет возвращаться к кодам по умолчанию всякий раз, когда прикладная программа будет загружена.

7.2.6 Дата и время (Date and Time)/Летнее время (Daylight Saving)

Пункт меню типа *Дата и время* () и *Летнее время* () используются для создания окон для установки даты и времени и для установки летнего времени. Окно создаваемое по умолчанию может быть изменено или дополнено, добавлением текста вручную в окне *экран панели оператора (Operator Panel Display)*.

7.2.7 Недельный/Годовой таймер (Week/Holiday Chart)

Пункт меню типов *Недельный таймер* () и *Годовой таймер* () используются для создания окон для просмотра и редактирования временных расписаний. Расписание содержит ряд недельных и годовых таймеров. Расписание подразделено на два окна (недельный и годовой таймер) для упрощения взаимодействия оператора через панель оператора. Если свойства связаны с *недельным и годовым таймерами*, то они установлены для одного и того же расписания. Чтобы привязать таймер к расписанию выберите объект расписания в списке сигналов (сравни *Вставить сигнал* в окне *Статус*) и нажмите кнопку "Связать расписание" (Assign time sch). Используемые сигналы в списке сигналов будут отображаться прямым начертанием, а не используемые жирным начертанием. Если сигнал выбран только как недельный таймер или только как годовой таймер, он будет отображаться прямым начертанием. Замечание: показываемый список содержит только сигналы, принадлежащие к одному модулю, таким образом сначала необходимо выбрать модуль через меню прокрутки над списком сигналов.

Для того чтобы уменьшить использование памяти TAC Xenta, надо выбрать через *Свойства (Settings)* в меню *Форматы (Formats)* использование шаблонов временных расписаний. в этом случае окна недельного и годового таймеров будут иметь фиксированный формат. В противном случае, окно создаваемое по умолчанию может быть изменено или дополнено добавлением текста вручную в окне *экран панели оператора*. Сервисное меню

7.2.8 Сервисное меню TAC

Сервисное меню TAC - это неизменное меню, которое делает возможным через панель оператора считывать версию системной программы TAC Xenta, название TAC Xenta и сетевой адрес, конфигурацию модулей I/O и перезапускать TAC Xenta.

В сервисном меню не будет пунктов, пока оно автоматически не будет создано при генерации дерева меню панели оператора, если файл OPDOCADD.GOP определен в папке TAC Menta (может быть определен в *Форматы - Свойства (Formats - Settings)*).

Сервисное меню, которое всегда располагается в конце дерева меню, доступно только через определенный код доступа.

7.3 Редактирование существующего дерева структуры меню

Когда структура меню используется повторно или создается как описано в предыдущих разделах., может появиться необходимость переместить/изменить текст или переделать пункты меню.

Для того чтобы использовать повторно файл дерева меню панели оператора (.OPC) или его часть в других приложениях, следует сохранить его под другим именем (именем нового приложения). Затем выйти из программы конфигурации панели оператора и создать новый файл (.AUT/.MTA) с тем же именем (если он еще не существует). Перезапустите программу конфигурации панели оператора через окно FBD, и Вы сможете редактировать новый файл дерева меню панели оператора. Все свойства сигнала, которые все еще существуют (с теми же именами) в новом файле, останутся нетронутыми, другие свойства будут пусты и могут быть вставлены из списка сигналов нового файла. Новые сигналы, которые не существуют в исходном списке сигналов, доступны в списке сигналов исправленного файла, и также могут быть вставлены.

7.3.1 Перемещение пунктов меню

В "списке пунктов меню" (Menu item list) различные пункты могут быть отсортированы перетаскиванием (выберите пункт меню указывая на нем курсором мыши, нажмите левую кнопку мыши и удерживая ее нажатой перетаскивайте пункт меню на требуемое место). Также возможно перемещать пункты меню в окне дерева структуры меню от одного узла подменю к другому.

7.3.2 Изменение разметки эрана панели оператора

Когда Вы переносите или редактируете текст на экране панели оператора (Operator Panel Display), может понадобится переместить привязки сигналов на экране панели оператора. Для этого выберите привязку курсором мыши и переместите ее перетаскиванием . (Выберите привязку сигнала, указав на нее курсором мыши, нажмите левую кнопку мыши и удерживая ее нажатой перетаскивайте привязку в необходимое место.)

7.3.3 Копирование и вставка

Функции копирования и вставки могут быть использованы для всех пунктов в корневом меню или узлов подменю. Пункты копируются/вставляются со всеми их подменю и сигналами. Скопированные пункты сохраняются в буфере - как текст, так и картинки - и могут затем быть использованы для копирования между двумя окнами файлов описания.

7.4 Автоматическая генерация дерева меню панели оператора

Программа конфигурации панели оператора содержит функцию для автоматической генерации законченного дерева меню панели оператора. По средствам этой функции пользователь может импортировать все общедоступные сигналы из FBD в стандартный формат представления панели оператора, вместо добавления каждого сигнала вручную.

Функция вызывается командами *Генерировать (Generate)* или *Загрузить (Download)* в режиме имитации TAC Menta, если было установлено *Автогенерация дерева меню (Auto generation of menu tree)* в меню *Свойства (Preferences)*. Дерево меню будет создано из пустого стандартного файла дерева меню панели оператора, заданного (Стандатное дерево OP = Path\FILENAME.OPE) в пункте *Форматы - Уставки (Formats - Settings)*. Если стандартный файл дерева меню панели оператора не был определен, используется файл STD_AUT.OPE. Автоматическая генерация также может быть начата из программы конфигурации панели оператора при выборе пункта *Построить (Build)* в меню *Дерево (Tree)*.

Функция автоматической генерации сначала стирает все старые данные в пунктах меню статус. Затем функция выполняет следующее:

- Все RO **бинарные** сигналы будут отсортированы в алфавитном порядке в первом доступном пункте меню *Статус (Status)* низкого уровня доступа, с одним сигналом на строку. Названия сигналов занимают по длине позиции 1-14 (более длинные названия усекаются), привязки сигналов располагаются в позициях 15-17 и ед.измерения сигналов располагаются в позициях 18-20. Новые страницы автоматически добавляются, если это необходимо. Если FBD содержит два или более **Модуля (Modules)**, пункты меню статус будут автоматически перемещены в пункт *Подменю*, содержащий соответствующее число пунктов меню статус, все с теми же названиями, что и модули.
- Все RO **аналоговые** сигналы будут отсортированы в алфавитном порядке во втором пункте меню *Статус (Status)* с низким уровнем доступа, с одним сигналом на строку. Следовательно, должно быть по меньшей мере два статус пункта меню в дереве, или аналоговые сигналы не будут отображаться. Названия сигналов занимают позиции 1-9 (более длинные сигналы усекаются), привязки сигналов (со свойствами по умолчанию) занимают позиции 10-17 и инженерные ед. измерения занимают позиции 18-20. Новые страницы добавляются автоматически по необходимости. Оперирование с Модулем, как описано выше.
- Все **бинарные** сигналы (как RO, так и R/W) будут отсортированы в алфавитном порядке в первом *Статус* пункте меню среднего

или высокого уровня доступа. Разметка и оперирование с модулями, как описано выше.

- Все *аналоговые* сигналы (как RO, так и R/W) будут отсортированы в алфавитном порядке во втором *Статус* пункте меню со средним или высоким уровнем доступа. Разметка и оперирование с модулями, как описано выше.

- Первое доступное *Расписание* будет соотнесено с первым пунктом меню *Недельный таймер* и первым пунктом меню *Годовой таймер*. Второе расписание будет соотнесено со вторым пунктом меню недельный таймер и вторым пунктом меню годовой таймер, и т.д. Замечание: появится сообщение об ошибке, и генерация будет прервана, если в FBD расписаний будет меньше, чем Недельных/Годовых таймеров в дереве меню. Оперирование с модулями, как описано выше.

Дерево меню панели оператора обычно содержит информацию, показывающую тип приложения, версию программного обеспечения и т.д. Этот тип меню создается как *Статус* пункт меню, содержащий только текст. Для предотвращения перезаписи информационного меню при автоматической генерации, оно может быть задано, как Информционный узел (Имя "Info" меню = MENUNAME) в *Форматы - Уставки (Formats - Settings)*. В последующем это меню будет исключено при поиске программой статус пунктов меню.

Сгенерированное дерево меню панели оператора, конечно, может редактироваться вручную в программе конфигурации панели оператора. Замечание: тем не менее, все изменения, внесенные вручную, будут удалены при новой автогенерации дерева меню. Следовательно, всегда пытайтесь подготовить стандартный файл дерева меню панели оператора в котором будет строиться новое дерево меню, в этом случае редактирование вручную не потребуется. Например, возможно изменить образец на характеристики сигналов и/или сохранить их.

7.5 Файл описания панели оператора

Еще один альтернативный способ создать дерево меню панели оператора - это использование файлов описания панели оператора (.DOP). Пользователь описывает дерево меню в текстовом файле, импортирует этот файл в программу конфигурации панели оператора, и файлы .OPC и .BIN автоматически будут созданы и сохранены на диск. Файл .DOP можно отредактировать любым тестовым редактором.

7.5.1 Данные и описания синтаксиса в файлах описания панели оператора (.DOP)

Первая позиция в строке зарезервирована для одного из трех ключевых слов:

OPC	Полный путь\имя файла для создания дерева меню панели оператора, обычно .MTA файл. Также .BIN файл сохраняется по тому же пути.
ESP	Полный путь\имя файла для списка сигналов обычно .MTA файл, который используется как входной файл для .DOP файла.
MENU	Информация о названии меню, типах пунктов меню, уровнях доступа и других атрибутах, описывающих разметку определенного экрана панели оператора.

Строки комментариев помечаются в начале строки символом *.

Каждый пункт меню в дереве меню панели оператора должен быть задан отдельно. Чтобы поместить пункт меню в 'корневой' уровень, введите символ дефиса (-) перед названием меню. Пункты меню (без дефисов), задаются после того как "подменю" будут построены в уровне "подменю" в дереве меню. Для задания пункта меню в другом подменю нежели предыдущее, введите полный путь имени, например, -PARAMETERS- HEATING_COIL-FROST_PROTECTION.

Ключевые слова и атрибуты отделяются пробелом или табуляцией. Следовательно, все тексты атрибутов, содержащие пробел должны быть введены в одинарных кавычках, например, 'подменю'. Следующие типы пунктов меню могут следовать после ключевого слова MENU:

Под-меню	MENU 'MENU NAME' 'Sub menu' High Будет создана стандартная разметка экрана.
Недельный таймер	MENU 'MENU NAME' 'Week chart' Low AHU01\TC_SF01 Будет создана стандартная разметка экрана, но следует обратить внимание, что привязки специальных временных расписаний (например, AHU01\TC_SF01) должны быть включены.
Годовой таймер	MENU 'MENU NAME' 'Holiday chart' Medium AHU01\TC_SF01 Будет создана стандартная разметка экрана, но следует обратить внимание, что привязки специальных временных расписаний (например, AHU01\TC_SF01) должны быть включены.
Статус	MENU 'MENU NAME' Status High 'Text' 'T11 ~##.##°C' T11 0 100 1 RO

Строка за строкой, текст, отображаемый на экране панели оператора, водится в одинарных кавычках, точно так, как он будет появляться на экране. Текст усекается, если он занимает более 20 символов. Пустые линии могут быть заданы как ' ' (символ пробела, но не табуляция!). Новые страницы автоматически создаются по необходимости.

Цифровые сигналы задаются с символами '~#.' в тексте и сопровождаются двумя атрибутами: привязка сигнала (включая модуль) и атрибут запись/чтение (RO/RW), все записанные в одну линию.

Аналоговые сигналы задаются с символами '~#.' в тексте и сопровождаются пятью атрибутами: привязка сигнала (включая модуль), уставка *Минимума*, уставка *Максимума*, уставка *Число разрядов* и атрибут чтение/запись (RO/RW), все записанные в одну линию. Чтобы отображать более одного сигнала в одной строке экрана панели оператора, отделите следующий сигнал и его атрибуты. Максимально 10 сигналов может быть задано в одной строке, и все данные для одной определенной строки на экране должны быть в одной строке .DOP файла.

Access Code	MENU	'MENU NAME'	'Access Code'	Low
Edit Access Code	MENU	'MENU NAME'	'Edit Access Code'	Medium
Alarm	MENU	'MENU NAME'	Alarm	High
Date and Time	MENU	'MENU NAME'	'Date and Time'	Low
Daylight Saving	MENU	'MENU NAME'	'Daylight Saving'	Medium

Для всех этих пунктов меню будет создана стандартная разметка экрана.

7.5.2 Файл описания панели оператора. Пример

```

*****
*      TAC XENTA AHU11      FILE TYPE: TAC XENTA OP
*      PROJECT:            Hospital west
*      AUTHOR:             U. M. Owen
*      DATE:               980401
***** FILE NAME *****
*      Name
OPC    C:\PROGRAM\TAC\PROJECT\HOSPITAL_WEST\AHU11\AHU11
ESP    C:\PROGRAM\TAC\PROJECT\HOSPITAL_WEST\AHU11\AHU11
***** OP INFORMATION *****
*      Name                Type                Access Level
*      'OP text row        ' Signal ref          Min  Max  Dec.  RO/RW
MENU  -INFO                Status                Low
      'Appl: 123456-789'
      'Version: 1.0B'
MENU  '-PLANT STATUS'      Status                Low
      'T11 MV ~###.#°C '   AHU11\T11_MV         0   100  1   RO
      'T11 SP ~###.#°C '   AHU11\T11_SP         0   50   1   RO
      'Outdoor ~###.#°C '  T31_MV               -50  50   1   RO
      'Heating pump '
      'Ind./Man. ~# / ~# '  AHU11\P11_Op  RO    AHU11\P11_M  RO
      'Run time ~##### h' AHU11\P11_RT         0   32500 1   RO
MENU  -ALARM                Alarm                Low
MENU  '-ACCESS CODE'        'Access Code'        Low
MENU  '-TIME PROGRAM'       'Sub menu'            Medium
MENU  'Week program'        'Week chart'          Medium
      AHU11\TC_SF11
MENU  'Holiday program'     'Holiday chart'       Medium
      AHU11\TC_SF11
MENU  -DATE/TIME            'Date and Time'       Medium
MENU  '-CONTROL P&R'        'Sub menu'            Medium
MENU  Room                  Status                Medium
      'Gain Room ~### °C '  AHU11\ROOM_G         0   99   0   RW
      'Itime Room ~### Min' AHU11\ROOM_ITIME     0   99   0   RW
MENU  Heating                Status                Medium
      'Heat P&and ~### °C ' AHU11\HEAT_P&AND     0   99   0   RW
      'Heat Itime ~### Min' AHU11\HEAT_ITIME     0   99   0   RW
MENU  '-CHANGE CODE'        'Edit Access Code'    Medium
***** EMD *****

```

7.5.3 Импорт файла описания панели оператора

Существует два альтернативных пути импорта .DOP файла:

1) Используйте **Файл - Импорт (File - Import)** в программе конфигурации панели оператора. Если дерево меню уже существует, импортируемое дерево будет автоматически вставлено после существующего. Обычно, вопрос "Сохранить?" сохранить появляется при выходе из программы. Импортирование не удалось, если этот вопрос не появился.

2) Используйте команду **Run** (или клавишную комбинацию быстрого вызова) в меню "пуск" Windows. Формат команды: **Run путь\GraphOP.exe /d путь\имя файла.dop**

Обратите внимание что команда **Run** использует DOS; следовательно, *путь\имя файла* должны быть введены в одинарных кавычках, если содержат пробел или их длина более 8 символов.

Программа создаст новый .BIN файл, но не .OPC файл. Если файл .OPC с тем же именем уже существует, он не будет изменен. Чтобы избежать ошибок, следует либо переместить старый .OPC файл, либо запустить программу конфигурации панели оператора и сохранить правильный .OPC файл.

7.5.4 Экспорт файла описания панели оператора

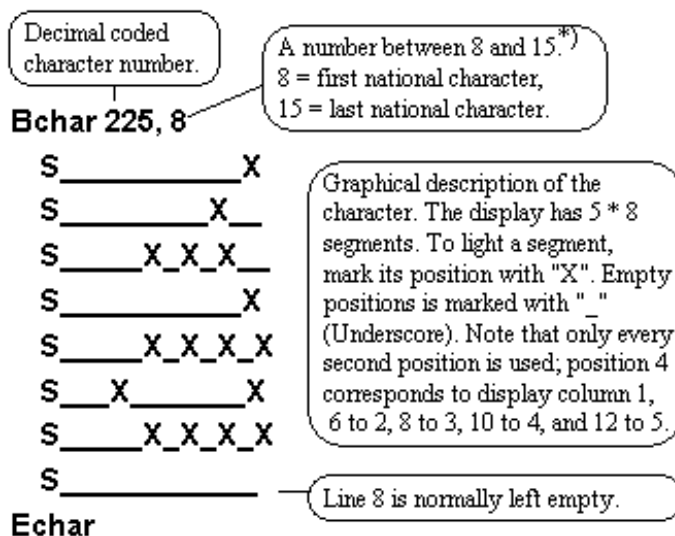
Чтобы экспортировать открытое дерево меню в .DOP файл, используйте **Файл - Экспорт (File - Export)** в программе конфигурации панели оператора. При экспорте в .DOP-файл, атрибуты принадлежащие ключевым словам будут на английском, например, MENU Week chart 'Week chart' Medium. Тем не менее, если Ваша программа TAC Menta не на английском, атрибуты не следует редактировать, пока программа работает с двумя языками, английским и русским, при импорте.

7.6 Определение файла набора символов

Для отображения на LCD экране панели оператора символы, совместно со стандартным отображением в формате ASCII, представляются десятичными кодами от 32 до 125. Дополнительно 8 символов с кодами выше чем 127 могут быть заданы в файле набора символов русской раскладки.

Чтобы задать новый символ в файле набора символов, скопируйте существующий файл CHARSET с новым именем в проводнике Windows™ Explorer. Затем отредактируете новый файл набора

символов, например, в редакторе Notepad. В CHARSET используется следующий формат:



*) Note that national character number 14 and 15 are reserved for special "AM"/ "PM" characters, when using the AM/PM Time Format.

Простейший путь найти номер символа в десятичном коде, запустить программу карты символов (CHARMAP.EXE) в проводнике Windows™ Explorer (обычно располагается в группе связанных программ), и найти символ, например, в отображении Arial True Type. Здесь все символы представлены по порядку, начиная с "пробела" (десятичный код 32) в левом нижнем углу.

В заключении, откройте диалоговое окно **Форматы - Файл набора символов (Formats - Character set file)** в программе конфигурации панели оператора, и выберите новый файл набора символов. В дальнейшем, новые символы будут использоваться при выполнении команды **Генерировать (Generate)**.

7.7 Меню опции

7.7.1 Строка меню

В строке меню, можно найти ряд вложенных меню, каждое с рядом подпунктов, которые рассматриваются ниже:

ФАЙЛ (FILE)

Новый (New)	Закрывает текущий файл дерева меню панели оператора и открывает новый файл. Если текущий файл не был сохранен, то до его закрытия появится сообщение с предложением его сохранить.
Открыть (Open)	Загружает файл дерева меню панели оператора с диска. Файлы на диске должны иметь расширение .OPC (дерево меню + спецификация) или .OPE (только дерево меню). Текущее окно панели оператора будет закрыто до открытия нового файла. Если текущий файл не был сохранен, то до его закрытия появится сообщение с предложением его сохранить..
Сохранить (Save)	Сохраняет текущий файл панели оператора на диск с тем же именем, что и был.
Сохранить как (Save As)	сохраняет текущий файл панели оператора на диск, но позволяет пользователю вести новое имя файла. Возможно сохранить файл как .OPC файл или как .OPE. Чтобы сохранить как .OPE файл, загруженный список сигналов сначала должен быть удален (Дерево - Удалить спецификацию).
Печать (Print)	Распечатывает обзорную или детализированную (полностью раскрытое дерево меню) документацию для панели оператора на активном принтере.
Предварительный просмотр (Print Preview)	Отображает на экране вид распечатки, общий (Overview) или детальный (Detailed).
Параметры печати (Print Setup)	Используется для выбора активного принтера и изменения установок печати.
Импорт (Import)	Используется для импорта дерева меню панели оператора из файла описания панели оператора (.DOP).
Экспорт (Export)	Используется для экспорта дерева меню панели оператора в файл описания панели оператора (.DOP).
Выход (Exit)	Выходит из программы и возвращается в ОС или в TAC Menta, если была запущена отсюда. Если текущий файл не был сохранен, то до его закрытия появится сообщение с предложением его сохранить.

ФОРМАТЫ (FORMATS)**Свойства сигнала
(Signal Properties)**

Здесь устанавливаются значения по умолчанию для Максимума, Минимума и Числа разрядов для использования во всех новых окнах определения параметров сигнала.

Аварии (Alarms)

Окно, в котором можно редактировать текст, который показывает статус аварии в списке аварий. Здесь же задаются аббревиатуры в русской раскладке для слов "активная", "сбросившаяся" и "подтвержденная".

**Дата и время
(Date and Time)**

Окно, в котором устанавливаются форматы представления даты и времени. Выбранный формат используется при отображении аварий, расписаний и т.д. Также русский текст для названий дней недели вводится в этом окне.

**Национальные
названия дней недели
(National Week
Days Text)**

В этом окне задаются русские названия для дней недели. Эти названия используются при отображении даты и времени в панели оператора.

Национальные названия месяцев (National Months Text)	В этом окне задаются русские названия месяцев. Эти названия используются для отображения некоторых форматов даты. Только первые три символа в каждом названии месяца используются в поддерживаемом формате отображения даты.
Файл набора символов (Character Set File)	При использовании символов русской раскладки, отличных от стандартных символов в формате ASCII (0-127), должен использоваться файл набора символов, содержащий описание символов русской раскладки. LCD экран панели оператора поддерживает 8 задаваемых пользователем символов, описываемых в файле набора символов русской раскладки. Файл используется программой при выполнении команды Генерировать (Generate). Программа затем проверит заданный текст для символов в формате ASCII с кодом более 127, и переместит их в файл набора символов. Это позволит корректно отображать русские буквы в панели оператора.
Уставки (Settings)	Используется для установки: <ul style="list-style-type: none">- Временная папка (Temporary directory); задается папка для всех временных файлов дерева меню, создается и используется в процессе Генерации.- Разделитель (Separator); один символ, например "-" или "/", используется для разделения названия модуля и названия сигнала.- Использовать шаблоны временного канала (Use Time Channel Templates); задается, если Шаблоны Расписаний должны быть использованы.- Использовать сервис меню (Use Service menu); задается, если TAC сервис меню автоматически должно быть включено при генерации дерева меню панели оператора.- Стандартное дерево OP (Standart OP tree); путь и имя для пустого файла дерева меню панели оператора, из которого будет построено дерево меню при автоматической генерации.- Название "Инфо" меню (Name "Info" menu); определяется введенное имя статус меню, как информационного узла, которое означает, что этот пункт будет защищен от перезаписи при автоматической генерации дерева меню.- OP; выберите Custom, если используется панель управления с набором русских символов, в противном случае По умолчанию (Default).

**Обратите внимание!**

Если Вы постоянно хотите использовать отличные от предопределенных текст и значения для *Свойств сигнала*, *Аварий*, *Даты и времени*, *Русских названий дней недели* и *Русских названий месяцев*, загружаемых в программе конфигурации панели оператора. Вы можете сохранить эти измененные текст и значения через пункт *Сохранить как* пустой файл дерева меню (.OPE), который Вы потом можете использовать как файл по умолчанию всякий раз, когда Вы будете создавать новое дерево меню.

ДЕРЕВО (TREE)**Раскрыть (Expand)**

Показываются все уровни описаного дерева меню. Это эквивалентно двойному щелчку на всех иконках дерева меню с символом "+" .

Тест (Test)

Производится предкомпиляционная проверка, чтобы убедиться, что получившееся меню панели оператора будет успешно загружено. Этот тест проверяет на следующие ошибки:

- Пункты меню выстроены не по возрастающему уровню доступа.
- Файл спецификации включает незагруженный список сигналов.
- Набор русских символов не выбран.
- Пункт корневого меню пуст.
- Привязки расписания в недельном таймере не определены.
- Привязки расписания в годовом таймере не определены.

Этот тест также проверит следующие предупреждения:

- Не все сигналы в файле спецификации используются в меню ОР.
- Расписание в файле спецификации не используется в меню ОР.

Генерировать (Generate)

Сначала законченное меню ОР тестируется (см. выше). Затем, если ошбок не обнаружено, после нажатия кнопки **OK** структура меню компилируется в файл, который будет загружен совместно с приложением из TAC Menta. Результат компиляции представляется динамично в окне на экране. Закройте это окно для продолжения Вашей работы.

Построить (Build)

Запустить автоматическую генерацию дерева меню панели оператора.

Загрузить спецификацию (Load Specification)

Используется для загрузки новой спецификации сигналов.

Обновление спецификации (Update Specification)

Обновляет активную спецификацию сигналов. Используется при советном программировании FBD и работе программы конфигурации панели оператора. Тем не менее, новая версия файла спецификации (.ESP) должна быть сначала создана в режиме имитации командой Генерировать до того, как она будет обновлена.

**Удалить
спецификацию
(Remove Specification)**

Удаляет загруженный список сигналов из памяти.

**Просмотр
спецификации
(View Specification)**

Используется для отображения спецификации сигналов.

**Копировать в буфер
(Copy to clipboard)**

Структура дерева меню копируется в буфер обмена в формате мета-файла Microsoft® Windows и в текстовом формате. Эта копия в свою очередь может быть экспортирована из буфера обмена в другие приложения Microsoft® Windows. Она может быть вставлена как текст или как картинка , например, в Microsoft® Word.

ПОМОЩЬ (HELP)

Содержание (Contents)

Вызывает файл помощи с основной информацией из этого руководства.

**О программе (About
OP/32 Configuration
Tool)**

Отображает информацию о текущей версии программы.

8 Другие функции и инструменты

8.1 Распечатка документации прикладной программы

Команда **Параметры страницы (Page setup)** в меню **Свойства(Preferences)** используется для установки вида распечатки FBD и вид на экране окна FBD, где границы печатной страницы помечаются как синие линии. Здесь же устанавливаются ориентация бумаги при печати (**Горизонтальное** или **Вертикальное**), **Поля** и **Размеры блоков**. Размер блоков 100% выбран по умолчанию, но меньшие размеры блоков могут использоваться для получения лучшего общего представления, и большие размеры блоков могут использоваться чтобы сделать распечатанную страницу более удобной для чтения. Графический размер приложения, рабочая область окна FBD (**Страницы**), задаются числом горизонтальных и вертикальных страниц. Рабочая область никогда не может быть меньше, чем отображаемое количество страниц при максимальном уменьшении.

Заголовок, печатающийся на всех страницах создается в окне Параметры страницы. Строка 1 заголовка - это всегда название приложения. Строка 2 содержит информацию введенную в поле **Заголовок страницы (Page header)**, которое может содержать три позиции, текст из поля таблицы **Программная спецификация (#author, #abbrev, #type и #data)**, время и дата распечатки (**#prdate** и **#prtime**) и символ табуляции (**#tab**).

Синие линии на экране, которые помечают границы печатных страниц, могут быть включены/выключены в окне Параметры страницы. Команда **Добавить ссылки соединений (Add boundary ties)** в меню **Правка (Edit)** добавляет указатель перекрестных ссылок с обеих сторон границы печатной страницы для всех соединений, пересекающих границу. Ссылки соединений можно передвигать и/или удалять. При распечатке ссылки соединений будут указывать номер страницы и сигнала, где продолжается соединений.

Документация распечатывается при выборе команды **Печатать (Print)** в меню **Файл (File)**. Сначала пользователь выбирает информацию для распечатки: спецификация (устройства, общедоступные сигналы, список I/O, аварии. константы, расписания и параметры блоков), FBD, связанные текстовые файлы и/или дерево меню OP. Ориентация FBD выбирается в окне Параметры страницы, другая информация всегда печатается в соответствии с установками принтера в Microsoft® Windows. Выберите принтер из списка принтеров.

Если выбрана опция текстового файла, связанный текстовый файл (если доступен) распечатывается после программной спецификации.

Опция *Масштабирование распечатки к одной странице (Scale output to one page)* в окне *Печать (Print)* делает возможным печатать один уровень FBD на одной странице, или одна страница на уровень. Опция *Цветная распечатка (Color printout)* используется для цветных принтеров. Опция Пропускать пустые страницы (Skip empty pages) означает, что полностью пустые страницы окна FBD не будут печататься.

Если выбрана группа блоков при выполнении команды *Печать (Print)*, выбранная группа распечатывается взамен всего FBD.

Если функциональный блок диаграмм содержит HFB, Вы можете либо распечатать фактический уровень FBD, либо распечатать все уровни.

Дерево меню панели оператора может быть распечатано на двух уровнях; как обзорная часть или как детализированная часть, включающая полное расширенное дерево меню.

Как альтернатива распечатки спецификации, можно сохранить список входов/выходов в виде текстового файла формата ASCII (.TXT) посредством команды *Экспорт списка I/O (Export I/O list)* из меню Файл (File). Текстовый файл в свою очередь можно импортировать в другие программы, например, САД программы. Список I/O в текстовом файле содержит следующие столбцы: № модуля I/O (0 = TAC Xenta 280/301/302/401 и т.д.), тип TAC Xenta или модуля входов/выходов, определение входа/выхода и название сигнала.

Если в графе "Параметры блока" в окне *Распечатка (Printout)* стоит галочка, может быть распечатаны комментарии определенные для TAC Vista.

8.2 Связанные текстовые файлы

Каждый FBD (.AUT или .MTA) файл может иметь связанный с ним текстовый файл с тем же именем, но с другим расширением (например, .TXT или .DOC). Этот файл можно использовать, например, для текстового описания функций. Текстовый редактор (или текстовый процессор) и расширение связанного текстового файла задаются пользователем. Имя исполняемого файла определяется в команде *Параметры (Settings)* из меню *Свойства (Preferences)*. Изменения в этой информации сохраняются при последующих сеансах.

Редактор текстовых файлов открывается при выборе команды *Текстовый файл (Text File)* в меню *ИНСТРУМЕНТЫ (Tools)*. Выбранный редактор текстовых файлов или текстовый процессор запускается с текстовым файлом, соответствующим имени файла текущей программы, для редактирования. Если текстовый файл не

существует в той же папке, что и .AUT файл, файл DEFAULT.xxx (где "xxx" - выбранное пользователем расширение текстового файла) копируется в папку FBD и получает тоже имя, что и FBD файл. Файл DEFAULT.xxx должен располагаться по тому ж пути, что и TAC32.EXE. Файл DEFAULT.xxx может быть задан пользователем и должен, конечно, быть совместим с выбранным текстовым редактором.

8.3 Конфигурация устройства

Команда **Конфигурация устройства (Device Configuration)** в меню **Опции (Options)** объявляет тип TAC Xenta и I/O модулей для использования их приложением.

Тип базового устройства (Base unit Type)	Тип TAC Xenta (TAC Xenta 280/300/401 и т.д.) для текущего приложения, выбранный из predetermined списка поддерживаемых типов. Для TAC Xenta 301/302/401 список задается в файле TATYPE.INI.
Системная версия (System Version)	Определение версии системной программы TAC Xenta. Если выбрано 1.x, команды Генерировать и Загрузка создадут загружаемый в TAC Xenta файл кода в формате версии 1.1, взамен TAC Xenta формата версии 3 (по умолчанию).
Аппаратная версия (Hardware Version)	Показывает аппаратную версию для выбранного устройства TAC Xenta (281/282/283/301/302/401 и т.д.), и не должна изменяться. Если приложение в TAC Xenta 300 должно содержать Тренд Лог, то должна быть использована аппаратная версия 2.

XIF заголовок, генерируемый в соответствии со стандартом LonMark (XIF Header generated according to LonMark standard)

TAC Menta может, для TAC Xenta начиная с версии 3.3 системной программы, генерировать XIF файл, в котором программный ID устройства, одобренный LonMark включается в первую строку параметров. Предыдущие версии не содержат этой информации, но выбрав нужную версию, например, при модернизации устройства TAC Xenta, правильная информация будет включена. **ОБРАТИТЕ ВНИМАНИЕ!** Если XIF файл изменяется таким образом, следует переустановить все его "связи" в программном обеспечении сетевого управления.

Добавление новой сетевой переменной (SNVT) последней в XIF файл (Add new network variables (SNVT) last in the XIF file)

Определяет чтобы все новые сетевые переменные (SNVT) располагались последними в XIF файле, при редактировании существующей прикладной программы. Это означает что только эти новые переменные необходимо связывать в программном обеспечении сетевого управления; а переделки новых связей не должно быть. Замечание: тем не менее, если сетевая переменная была удалена, или если существующий блок I/O был изменен, например, с 'Нелинейный аналоговый вход' на 'SNVT', следует переделать "связывание". Также самый первый раз, когда функция используется (которая задана по умолчанию в TAC Menta версии 3.24) в существующей прикладной программе, все связи должны быть "пересвязаны".

Использование LonMark 3.3 (Use LonMark 3.3)

TAC Xenta 283 может работать как одобренный LONMARK 3.3 табельщик реального времени, используя UTC время как входной сигнал и возвращенное локальное время как выходной сигнал. Тем не менее, это доступно только для системной версии 3.6 или выше. Если эта функция не выбрана будет создаваться LONMARK версии 3 XIF файл .

Таблица модулей расширения входов/выходов (IO Expansion Module Table) создается с кнопками **Правка (Edit)** и **Добавить (Add)**. Следующие параметры могут быть заданы/изменены:

Модуль (Module)

Определение номера (Module1, Module2 и т.д..) для каждого I/O модуля в текущем приложении. Новые модули могут быть добавлены, но только последний модуль в списке может быть удален.

Тип (Type)	Тип (TAC Xenta 411, TAC Xenta 412 и т.д.) для каждого I/O модуля в текущем приложении, выбираемый из вложенного меню списка поддерживаемых типов. Список определяется в файле TATYPE.INI.
Мин. время посылки (Min send time)	Определяет как часто I/O модуль может связываться с TAC Xenta. Минимально возможное значение (оно стоит по умолчанию) равно 500 мс, но значение может быть увеличено для уменьшения сетевого трафика.
Отчет о статусе CNT (Fast CNT reporting)	<p>Определяет используется ли создание отчета об изменениях состояний на импульсном входе (CNT). Функция используется, когда подключается выключатель для управления светом. Замечание: данная функция увеличивает трафик в сети. По умолчанию функция не выбрана; отправка отчета об изменении состояния будет осуществляться через 30 секунд, что например можно использовать при импульсном измерении энергии.</p> <p>Когда конфигурация устройства завершена, все таблицы связей будут содержать информацию о текущей конфигурации. Параметр Номер модуля (Mod Number) будет показывать номера TAC Xenta и определенных модулей I/O. Параметр Свойства входа/выхода (Terminal Ref) будет показывать адреса физических входов/выходов выбранного модуля. Если "связывание" уже было проведено и затем конфигурация устройства изменилась, "связи" останутся теми же, если те же привязки входов/выходов существуют в новом модуле, в противном случае связи будут пустыми.</p>

8.4 Программная спецификация

Команда **Программная спецификация (Program Specification)** находится в меню **Опции (Options)**. Программная спецификация включает всю информацию о прикладной программе, которая требуется программе конфигурации для загрузки прикладной

программы в устройство TAC Xenta.

Имя (Name)	Строка длиной до 20 буквенно-цифровых символов, определяемая пользователем вместе с именем прикладной программы. Символы ", ; ' - . \$ " используются как разделители или идентификаторы системных объектов в TAC Vista, и не применимы в именах приложений. Символ пробела не используется, но символы " _ : / " можно использовать. Можно использовать символы русской раскладки (включая, например, "е д ц з") в имени приложения.
Аббревиатура (Abbreviation)	Строка длиной до 4 символов. Эта строка используется только панелью оператора, см. главу 6, таким образом разрешены все символы, которые использует панель оператора.
Тип (Type)	Строка переменной длины, определяемая пользователем, описывает тип прикладной программы.
Автор (Author)	Строка переменной длины с именем автора прикладной программы.
Дата (Date)	Поле только для чтения с последней датой редактирования прикладной программы.
Период цикла (Cycle time)	Интервал (в мс) выполнения прикладной программы при выполнении ее в контроллере. Это параметр не отражается на выполнении прикладной программы во время режима имитации не-на-линии. Значение по умолчанию равно 1000 мс. Максимальное значение программного цикла равно 65 536 мс. Минимальное значение программного цикла равно 500 мс.
Блоки (Blocks)	Количество блоков в программе. Эта информация доступна только после компиляции программы.
I/O сигналы (I/O signals)	Выдает количество физических сигналов, выходов и входов, аналоговых и цифровых, которые используются прикладной программой. Это информация доступна только после компиляции программы.
Таблица общедоступных сигналов (Public signal table)	Список только для чтения всех общедоступных сигналов прикладной программы. Список содержит 4 столбца, отображающих идентификатор сигнала, тип, класс доступа и единицу измерения. Параметры таблицы констант, которые определяются как общедоступные также включаются в список, и все внешние I/O сигналы. Сигналы/параметры сортируются в алфавитном порядке.
Стандартное Прил. (Standard App.)	Переключатель только для чтения, который определяет будет ли текущее приложение Стандартным приложением или <i>определяемым Пользователем приложением</i> (если галочка= Стандартное приложение), см. " Выделение стандартных приложение/контроллеров " на стр. 118.

8.5 Генерация исполняемого кода

Выполняемый код прикладной программы генерируется при выборе команды *Генерировать (Generate)* в меню *Команды (Commands)* с режиме имитации. Обратите внимание, что файлы загружаемого кода определяются уставками в *Системной версии (System version)* в команде *Конфигурация устройства (Device configuration)* в меню *Опции (Options)*. Создаются следующие файлы, где "*" означает имя файла источника кода прикладной программы (.MTA/.AUT):

- *.COD Файл TAC Xenta "машинного кода" в формате ASCII. Выполняемый файл для загрузки в контроллер.
- *.ESP Файл спецификации: таблица с общедоступными сигналами, их атрибутами и т.д. Этот файл используется как файл входных данных для программы конфигурации панели оператора.
- *.XIF Файл внешнего интерфейса, генерируется только если SNVT содержатся в прикладной программе. Он содержит стандартизированное описание всех сетевых переменных/объектов прикладной программы, которые делают доступными "связи" и взаимодействие с другими узлами LonWorks других производителей.

Имя файла и дата файла исходного кода (.MTA/.AUT) включены в объектный и исполняемый файлы. Эти данные содержатся в информации, передаваемой в контроллер с ПК во время загрузки данных из контроллера, см. ["Загрузка данных приложения из контроллера" на стр. 119](#).

При генерации выполняемого кода, может быть сгенерирован и новый файл панели оператора, либо автоматически, либо после выбора пользователем во всплывающем меню. Таким образом, программа конфигурации панели оператора вызывается с .OPC файлом текущего приложения, но "именем" и "аббревиатурой" взятыми из текущей программной спецификации, при этом автоматически генерируются следующие файлы:

- *.OPC Файл дерева меню панели оператора, который состоит из совокупности .OPE (пустое дерево меню) и .ESP файлов.
- *.BIN Бинарный файл с данными дерева меню для загрузки в контроллер TAC Xenta совместно с .COD файлом.
- *.CHR Файл национального набора символов для загрузки в контроллер TAC Xenta совместно с файлом .COD.

8.6 Вычисление использования памяти

Можно отобразить рассчитанное использование памяти в текущем приложении через команду *Использование памяти (Memory Usage)* в меню *Опции (Options)*. Это окно также содержит текст, указывающий подходит ли приложение для TAC Xenta или нет.



Обратите внимание!

Приложение может быть загружено даже, если оно не подходит для TAC Xenta. В этом случае сетевая конфигурация должна быть переопределена.

Отображаемые величины рассчитываются для устройств TAC Xenta версии 3, и могут отличаться немного от реального использования памяти в TAC Xenta. Кроме того, есть дополнительные максимальные пределы в устройстве TAC Xenta, которые могут остановить попытку загрузки и выдать сообщение "*Синтаксическая ошибка кода*" ("*Parser code error*") при попытке загрузки. Пожалуйста, свяжитесь с TAC Helpdesk, если Вы столкнулись с такого рода проблемами!

Общая доступная память зависит от устройства TAC Xenta (280/300/400) и используемой версии. TAC Xenta использует следующие области памяти (три в TAC Xenta 300; четыре в TAC Xenta 400) для хранения прикладной программы и данных. В TAC Xenta 300 *Файла приложения* и *Приложение* соединены в одной области памяти.

Файлы приложения (Application files)

Файлы загружаемой программы (*.COD, *.BIN, *.CHR и *.BPR).

Приложение (Application)

Данные, извлекаемые из этих файлов, например, таблицы общедоступных сигналов, расписания и тексты аварий.

Параметры (Parameters)

Параметры, используемые функциональными блоками, например, начальные значения в PV блоках и координаты в блоке CURVE.

Рабочая область (Work area)

Данные создаваемые при выполнении прикладной программы, например, состояние выхода и расписание.

**Количество объектов
(Number of objects)**

Также отображаются:

Функциональные блоки и опеределения TAC Menta в TAC Xenta преобразуются в объекты. Для каждой области памяти, окно показывает общую используемую память, доступную память и свободную память в устройстве TAC Xenta. Для памяти Приложения, также показываются размеры различных типов загружаемых файлов. Если свободная память одной из трех областей стремится к уменьшению, размер прикладной программы должен быть уменьшен. В этом случае обратите внимание на следующее:

- Использование памяти приложением может быть уменьшено уменьшением размера файлов прикладной программы. Особенно дерево меню ОР и блоки формул (а также операторы, откомпилированные как блоки формул) используют много памяти.

- Свободная рабочая область пространства памяти может быть увеличена уменьшением числа недельных и годовых таймеров.

Окно использования памяти также содержит кнопку *Дополнительно (Advanced)*. При нажатии на эту кнопку появляется диалоговое окно со следующими темогу расчетами вероятного использования памяти:

TACN/SNVT вх/вых

Каждый тип устройства TAC Xenta имеет фиксированное число входов и выходов зарезервированных для сетевых переменных (TACNV) и SNVT. Для определенного (в окне *Конфигурация устройства*) типа устройства TAC Xenta, в окне представлено число *заданных* и свободных входов в текущем приложении.

**Вычислитель размера
BPR**

Если размер файла сетевого окружения (.BPR) не известен при создании FBD, он может быть рассчитан введением предполагаемого числа TAC-групп и усредненного числа устройств TAC Xenta (узлов) в каждой группе, вместе с усредненным числом символов в названиях Группы и Узла. Нажатие на кнопке *Рассчитать* приведет к представлению расчета размера .BPR, который лучше предварительно узнать дабы избежать нехватки памяти Приложения.

8.7 Функции На-линии

С TAC Menta, пользователь может считывать/записывать общедоступные сигналы и параметры в режиме на-линии, т.е. пока прикладная программа выполняется в подсоединенном устройстве TAC Xenta. До шести сигналов, включенных в трен лог обновляются чаще, чем другие сигналы. Возможно установить соединение на-линии для локального устройства TAC Xenta, только если оно физически подключено к ПК.

Сетевые коммуникации TAC Xenta могут быть сконфигурированы без доступа к исходному коду. Информация о "связывание" I/O может быть загружена из контроллера, изменена и загружена обратно в контроллер. Также возможно определить сетевые переменные в сети в режиме Не-на-линии в *режиме Редактирования (Edit)* и загрузить информацию позже.

8.7.1 Режим На-линии (Online)

Режим работы На-линии вызывается нажатием кнопки *На-линии (Online)* в режиме имитации, или выбором команды *На-линии (Online)* в меню *Опции (Options)*.

Не возможно ввести режим На-линии, когда в FBD содержатся тестовы блоки, также как подобная программа не может быть выполнена в контроллере. Если программа содержит тестовые блоки, появится сообщение об ошибке и операция будет прервана.

После того, как соединение *на-линии* установлено, возможны следующие ситуации, в зависимости загружена ли прикладная программа в TAC Menta в подсоединенное устройство TAC Xenta или нет:

1. Прикладная программа не загружена ни в TAC Menta ни в подсоединенном устройстве TAC Xenta

TAC Menta сразу переходит в режим *На-линии (Online)*. В этом случае Вы сможете только установить название подсоединенного устройства, пока прикладная программа не будет загружена.

2. Прикладная программа загружена на ПК и/или в подсоединенном устройстве

Пользователь может выбрать *Соединение (Connect)*, *Загрузка из... (Upload)*, *Загрузка в... (Download)* или *Отмена (Cancel)*.

Соединение (Connect)

Устройство TAC Xenta сразу же подсоединяется, если программный ID текущего FBD и приложения совпадают, в ином случае оператору предлагается выполнить операцию загрузки.

Загрузка (Download)

Когда команда *Загрузка* выполняется, сначала проверяется, что ни одно из следующих утверждений не выполняется, иначе операция прерывается:

- Нет FBD в TAC Menta
- Стандартное приложение загружается в устройство, предназначенное для свободного программирования.
- Приложение для свободнопрограммируемого устройства загружается в стандартное устройство.

Если ошибки не были обнаружены, приложение в TAC Menta сбрасывает состояние в начальное и генерируется новый .COD файл. Если требуется, пользователь может выбрать следует или нет генерировать новый ОР файл. Если ответ - да, запускается программа конфигурации панели оператора с .OPC файлом текущего приложения, но с "Именем" и "аббревиатурой", взятыми из текущей программной спецификации, и автоматически генерируется новый .BIN файл. Окончательно файлы .COD, .CHR и .BIN загружаются в подсоединенное устройство.

**Обратите внимание!**

Формат версии (1 или 3) файлов загружаемого кода может быть определен уставкой *Системная программа (System program)* в команде *Конфигурация устройства (Device configuration)* меню *Опции (Options)*, если файлы сгенерированы до подключения устройства TAC Xenta в режиме На-линии.

Загрузка из контроллера

При выполнении команды *Загрузить из... (Upload)*, сначала проверяется есть ли в устройстве программа (FBD), в ином случае операция прерывается.

Если программный ID TAC Menta FBD и программы в подсоединенном устройстве не совпадают, FBD в TAC Menta сбрасывается. Пользователю предлагается ввести имя для загружаемого в ПК .COD файла, и после загрузки работа будет возможна только в табличном режиме.

Если нет FBD, загруженного в TAC Menta, пользователю предлагается ввести имя файла для загруженного из контроллера .COD файла.

Если ошибок не обнаружено, приложение из контроллера загружается в ПК. Затем загружаются .BIN, .CHR (if they exist), .COD и .BPR файлы.

8.8 Загрузка прикладной программы

Обычно загрузка выполняется в TAC Vista. Функция *загрузка* в устройство TAC Xenta из TAC Menta через последовательный порт доступно посредством команды *Загрузка (Download)* в меню *Команды (Commands)*. Эта команда доступна только в режиме На-линии.

При загрузке приложения как файла в TAC Menta, для загрузки через последовательный интерфейс, информация о сетевом окружении извлекается из устройства TAC Xenta. При открытии приложения из базы данных TAC Vista, информация о сетевом окружении извлекается из базы данных TAC Vista.

Перед отправкой .COD файла в устройство, сохраненный на диске .COD файл обновляется со всеми изменениями, следанными в режиме На-линии. (Теперь .COD файл сохраняется еще раз с тем же (выбранным пользователем) именем, что и загружаемый файл). Программа затем ищет файл на жестком диске с определенным именем, и отправляет копию в подсоединенное устройство.

8.9 Загрузка данных приложения из контроллера

Функция *загрузка из...* в устройстве TAC Xenta через последовательный порт доступна с помощью команды *Загрузить из... (Upload)* в меню *Команды (Commands)*. Эта команда доступна только в режиме *На-линии (Online)*. Загружаются файлы .COD, .BIN (если существует) и .BPR.

При выполнении команды *Загрузить из...*, Пользователю сначала предлагается определить имя файла для загружаемого .COD файла. Программа затем загружает .COD файл из подсоединенного устройства, и сохраняет его на жестком диске с выбранным именем. В конце TAC Menta открывает .COD файл на диске.

После загрузки данных приложения, пользователь может проверить имя и дату файла прикладной программы в программной спецификации, что таким образом позволяет ему найти соответствующий исходный код приложения (если доступно) на диске. Правильный исходный код приложения должен быть загружен в TAC Menta с жесткого диска, чтобы возможно было использовать FBD в режиме *На-линии (Online)*, поскольку графическая информация не сохраняется в TAC Xenta.

8.10 Принудительное переопределение сигналов физических входов/выходов (*Override of Physical I/O Signals*)

Для целей наладки системы, существует функция принудительного переопределения во всех драйверах входа/

**Переопределение
(Override)**

выхода, кроме блоков CNT (функция доступна только в режиме Online):

Бинарный параметр (ДА/НЕТ), по умолчанию = НЕТ (галочка не стоит).

**Значение
переопределения
(Override Value)**

Параметр того же типа, что и выходной сигнал. Когда переключатель переопределения установлен (галочка стоит), переопределяемая величина непосредственно влияет на выход I/O - блока^а. Переопределяемая величина обычно получает те же единицы измерения, что и выход блока, кроме случая аналогового выхода, где переопределяемая величина задается в %.

Переключатель и значение переопределения предполагаются для использования в целях наладки системы и таким образом возможно управление через соединение. Принудительное переопределение можно считать он-лайн эквивалентом установок вручную значений физических входов в режиме имитации не-на-линии.

Функция принудительного переопределения выполняется через кнопку **Переопределение (Override)** в **таблице конфигурирования I/O (I/O configuration table)** (*только в режиме на-линии*).

а.Это означает, что переопределение влияет на сигнал немедленно, без проведения фильтрации.

8.11 Изменение параметров блоков в режиме На-линии

Изменение параметров (PVB/PVI/PVR) состояний выходных блоков в режиме На-линии осуществляется следующим образом:

Если в поле *Backup* установлена галочка, тогда при изменении состояния выхода новое значение копируется в параметр *Начальное значение (InitValue)* во Flash памяти. Новое *Начальное значение (InitValue)* будет таким образом также включено в .COD файл, когда он будет загружен.

Если в поле *Backup* не установлена галочка, изменение состояния выхода PVx не копируется и не сохраняется в *Начальном значении (InitValue)* во Flash памяти. В этом случае выход блока будет собран в *Начальное значение (InitValue)* при рестарте. Измененные значения выхода блока не включаются в .COD файл.

8.12 Изменение общедоступных констант в режиме На-линии

Если общедоступная константа изменится из TAC Menta в режиме На-линии, новое значение будет немедленно передано в TAC Xenta, и станет видимым с панели оператора.

Если значение общедоступной константы изменяется с панели оператора, эти изменения передадутся в TAC Menta только когда приложение будет загружено из контроллера.

Динамичное обновление общедоступных констант может замедлять представление сигналов в режиме На-линии. Поэтому, возможно выключать он-лайн обновление общедоступных констант используя кнопку *Пропустить константу (Skip const)*.

8.13 Изменение параметров "связывания" (Bind) в режиме На-линии

Когда параметр "связывания" в физическом (AI/AO/CNT/DI/DO/DOPU) I/O блоке изменяется из TAC Menta в режиме На-линии, новое значение не передается в TAC Xenta, пока следующая операция загрузки (включая новую генерацию .COD файла) не выполнена.

8.14 Обозначение стандартных приложений/ контроллеров

Контроллеры TAC Xenta обозначаются либо как *Стандартные*, либо как *Свободнопрограммируемые* (программируемые пользователем) контроллеры. Обозначение, которое уникально

для каждого контроллера, генерируется из нейрон ID по специальному алгоритму. Если контроллер обозначен как Стандартный, только стандартные приложения могут быть загружены в этот контроллер. Если контроллер обозначен как Свободнопрограммируемый, только созданные пользователем приложения могут быть загружены в него. Таким образом, не возможно загрузить стандартное приложение в свободнопрограммируемый контроллер и наоборот.

Файлы стандартного исходного кода приложения обозначаются уникальным паролем, сгенерированным из контрольной суммы FBD по специальному алгоритму. Контрольная сумма FBD также используется как уникальный программный ID.

8.15 Уникальный ID прикладной программы

Для того чтобы разъяснить ответственности сервиса/техподдержки и т.п., требуется наличие возможности проверки на месте идентичности прикладной программы, установленной в устройстве, и исходного кода, например, из библиотеки стандартных приложений TAC.

Программный ID - это контрольная сумма, относящаяся только к структурной и функциональной частям программы, не к значениям параметров или данным конфигурации входов/выходов. Поэтому возможно изменять параметры и/или данные конфигурации без воздействия на программную идентификацию. Программный ID отображается в окне программной спецификации.

9 Мастер Загрузки

**Обратите внимание!**

Для TAC Vista IV Мастер загрузки СЛЕДУЕТ ИСПОЛЬЗОВАТЬ ТОЛЬКО для замены системных программ в устройствах TAC Xenta.

Использование Мастера загрузки для загрузки приложений и сетевой конфигурации допустимо только для TAC Vista 3.x и TAC Menta 3.x.

Зайдите в меню Пуск Microsoft® Windows. Выберите TAC и TAC Tools и Вы найдете Мастер загрузки (Download Wizard).

Мастер загрузки - это управляющая загрузочная процедура для обновления системной программы одного или нескольких устройств TAC Xenta.

**Обратите внимание!**

Возможность повторного использования/обновления программного модуля при обновлении системной программы зависит от совместимости версий программ. Также, различные дополнительные подготовительные действия могут потребоваться. Например, TAC Xenta 300 с системной программой версии 1.01 не может быть обновлена до версии с 1.13 (или более поздней), если и исходный код приложения (.AUT) и сетевая конфигурация (.NWC) запущены при работе процедуры. Следовательно, всегда считывайте вложенную обновляемую информацию до инсталляции новой версии программы.

**Предостережение!**

НИКОГДА не загружайте системные программы для TAC Xenta 300 в TAC Xenta 280.

Это приведет к выходу TAC Xenta 280 из строя.

9.1 Как использовать Мастер загрузки (Download Wizard)

9.1.1 Диалог

Диалог Мастера загрузки содержит две страницы, отображаемые через использование кнопок *Следующая (Next)* и *Предыдущая (Previous)*. Вторая страница содержит следующие секции:

Система (System)

Сначала проверьте *тип устройства (Unit type)* TAC Xenta. Выберите *Сохранить текущую (Retain current)* системную программу или *Загрузить новую (Download new)*. Если будет загружена новая программа, необходимо ввести имя и путь для *Системного файла (System file)*(.MOT).

Приложение (Application)

Выберите *Сохранить текущую (Retain current)* прикладную программу или *Загрузить новую (Download new)*. При выборе опции *Очистить память (Clear memory)*, появится второе окно в котором будут предложены два альтернативных ответа. Выбор кнопки *Yes* приведет к полной очистке памяти, ее следует выбирать в том случае, когда в устройстве TAC Xenta не должно быть прикладной программы. Очистка может привести к проблемам при использовании устройства в некоторых типах сетей. Выбор кнопки *No* приведет к возвращению устройства к его заводским уставкам. Рекомендуется выбирать ответ Нет. Если загружается новое приложение, следует ввести имя и путь файла приложения.

Конфигурация (Configuration)

Выберите *Сохранение текущей (Retain current)* сетевой конфигурации или *Загрузить новую (Download new)*. Если поставить галочку напротив слов *Проверка нейрон ID (Tick Check Neuron ID)*, и Мастер загрузки будет искать потерянные и несовпадающие нейрон ID в базе данных. Если загружается новая конфигурация, имя и путь *файла базы данных (Database file)* должны быть указаны.

Подсоединенное устройство (Connected unit)

Отображается текущий узел или имя подсоединенного устройства TAC Xenta. Введите желаемое новое имя узла (обычно тоже, что и *Подсоединенное устройство*) из списка *узлов для загрузки (Node to download)*, который отображает все узлы в выбранном файле базы данных.

Информация (Information)	Диалоговое окно также содержит следующие кнопки: Отображает тип устройства, аппаратную версию, версию системной программы и версию PROM подсоединенного устройства TAC Xenta, и информирует Вас, если выбранный файл базы данных помечен флагом изменения (т.е сетевая база данных содержит данные, которые не были загружены в TAC Xenta).
Принять (Apply)	Запускает процедуру загрузки. Результат загрузки отображается в окне выполнения загрузки.
Закреть (Close)	Закрывает окно диалога.

9.1.2 Общая процедура загрузки

Если загрузка включает прикладную программу, мы рекомендуем открыть соответствующий .AUT файл и затем немедленно снова его закрыть. В этом случае Вы присвоите файлу новую дату и избежите дополнительного вопроса о генерации .BIN файла из TAC Menat во время процедуры загрузки.

1. Запустите "мастер загрузки" из группы программ TAC Menta.
2. Соедините кабелем RS232 определенный Вами коммуникационный порт и контроллер TAC Xenta.
3. Следуйте указаниям "Мастера" и определите какое ПО необходимо загрузить.
4. В области Конфигурация можно поставить галочку проверки нейрон ID. Если галочка стоит, Мастер сравнит ID устройства с ID, записанных в базе данных.
 - Если ID отсутствует в базе данных, он будет добавлен.
 - Если ID различаются, появится вопрос об обновлении базы данных.
5. Если галочка была установлена, нажмите "Применить" и ждите завершения процедуры. Результат будет показан в окне выполнения загрузки. Появление сообщения "Удаленное устройство повторно не подключено!" обычно можно игнорировать.

**Предостережение!**

Процесс системной загрузки в TAC Xenta очень критичен; на момент загрузки системы все другие приложения в компьютере должны быть закрыты, функция сохранения экрана должна быть выключена, компьютер и загружаемый контроллер желательно подключить к бесперебойному питанию! Если процедура прервалась, то может оказаться невозможным перезагрузить заново системное ПО и, естественно, дальнейшая эксплуатация данного контроллера будет не возможна. Восстановление подобной неисправности возможно только на заводе-изготовителе в Швеции.

6. Если загрузка была подтверждена, процедура может быть повторена с шага 2.
7. Накопленные результаты всех загрузок указываются в файле "dwiz.log", обычно расположенном в папке программы TAC Menta. Запись содержит информацию о
 - времени старта и неисправных узлах
 - выполненных операциях с временными метками
 - общем требуемом времени
 - результате операций
 - разделители
 - в конце: нейрон ID, если обновлялась база данных.
8. Когда загрузка будет завершена, закройте окно "Мастера загрузки".

9.2 Совместимость TAC Menta

TAC Menta v4 полностью совместима с TAC Xenta 280/300/3000/400/901 и TAC Vista v3. Также возможно осуществление соединения и сгенерировать код для TAC Xenta 280/300/400 v1.1 (См. окно **Конфигурация устройства (Device Configuration)**), подразумевается что используются только функции совместимые с TAC Menta v1.2 и TAC Xenta v1.1.

9.2.1 .AUT файлы

Ранние версии файлов исходного кода приложения TAC Menta могут быть открыты и отредактированы. Когда открывается старый файл, он автоматически конвертируется в формат последней версии. Если файл затем сохраняется вручную через команду Сохранить (Save), он сохраняется в новом формате. Не возможно сохранить новый файл в более старом формате.

По причине изменения размера страницы, выводимой на печать и функции разделения на страницы в TAC Menta v3, обновление приложения из версии v1 может потребовать корректировки вручную графического отображения FBD.

9.2.2 .COD файлы

После обновления TAC Xenta системной программы из версии v1.x в версию v3, .COD файл должен быть сгенерирован заново из .AUT файла до загрузки, из-за различий в формате загружаемого (.COD) файла между TAC Menta v1.2 и v3. Таким образом, исходный код приложения должен быть доступен при обновлении системной программы контроллера.

При соединении с устройством TAC Xenta v1.x, TAC Menta по умолчанию сгенерирует и загрузит .COD файлы в формате v1.x. Перед началом соединения, специальное сообщение отправляется в устройство TAC Xenta, ответ на которое определяет какой тип протокола использовать.

При соединении с устройством TAC Xenta v1.x, также возможно загрузить приложение из контроллера. Если до загрузки открыть соответствующий .AUT файл, Вы можете затем регенерировать .COD файл (с загружаемыми из контроллера параметрами) в формате TAC Menta v3. *Это, тем не менее, не возможно, если нет .AUT файла.* В настоящее время нет способа конвертировать .COD файл из версии v1.x в версию v3 без наличия доступа к .AUT файлу.

9.2.3 Файлы дерева меню панели оператора

Файлы (.BIN) дерева меню панели оператора версии V1.x используют старый, занимающий больше памяти для временных расписаний формат, принимаемый TAC Xenta v3. Тем не менее, если наличие свободной памяти - это проблема (что обычно является причиной), пользователю убедительно рекомендуется сгенерировать снова OP файл в формате v3, использующем 'Шаблоны временных расписаний' для сохранения места.

9.3 Обновление устройства TAC Xenta 300 до версии v3

При обновлении устройства TAC Xenta 300 из системной программы версии v1.x в версию v3, следует нижеописанная процедура. Все TAC Xenta 280/301/302/401 в сети должны иметь одинаковые версии системного ПО (v1 или v3).

9.3.1 Обновление .AUT файла с данными приложения (дополнительно)

- Запустите TAC Menta, и откройте обновляемый .AUT файл для приложения в устройстве TAC Xenta.
- Загрузите .COD файл из устройства TAC Xenta.
- Сохраните файл прикладной программы в режиме имитации (ответ "Да" на вопрос "...Хотите ли Вы сохранить?"). Затем он

автоматически будет сконвертирован в формат TAC Menta v3, и обновлен с загруженными из устройства параметрами.

Эти шаги можно исключить, если .AUT файл, сохраненный на диске, имеет обновленные параметры. Эта процедура тем не менее, необходима, если Вы уверены, что уставки параметров были изменены с панели оператора или из TAC Vista, и что эти уставки сохранялись после обновления.

9.3.2 Обновление системных и прикладных программ

В зависимости от обстоятельств принимается либо первая, либо вторая, из описанных ниже, альтернатив.

Альтернатива 1 - Нет изменений в сетевой конфигурации

- Запустите Мастер загрузки, и выберите следующие опции: Загрузить новую систему, Загрузить новое приложение и Оставить текущую конфигурацию.
- Выберите соответствующий файл системной программы и файл прикладной программы (выберите файл, созданный в шаге 1, если он подходит).
- Щелкните на кнопке Применить для завершения процедуры обновления. После того, как системная программа будет загружена, новый .COD файл в формате v3 будет сгенерирован из выбранного файла прикладной программы и загружен вместе с .BIN файлом (если он подходит) и загруженным из контроллера .BPR файлом.

Альтернатива 2 - Новая или измененная сетевая конфигурация

- Запустите Мастер загрузки, и выберите следующие опции: Загрузить новую систему, *Загрузить новое приложение и Загрузить новую конфигурацию.*
- Выберите соответствующие системную программу, прикладную программу (выберите файл, созданный в шаге 1, если он подходит) и файлы сетевой базы данных.
- Щелкните на кнопке ***Применить*** для завершения процедуры обновления. После загрузки системной программы, .BPR файл для подсоединенного устройства будет сгенерирован из выбранного файла сетевой базы данных, .COD файл в формате v3 будет сгенерирован из выбранного файла прикладной программы и, если подойдет, также .BIN файл. Окончательно, .COD файл будет загружен вместе с .BIN файлом или .BPR файлом.

9.3.3 Проверка корректности выполнения (дополнительно, но рекомендуется)

- Запустите TAC Menta, и откройте файл прикладной программы с исходных кодом приложения.
- Установите связь с устройством, и проверьте, что прикладная программа в контроллере выполняется правильно.

10 Сообщения об ошибках

При работе TAC Menta выдает некоторую информацию, предупреждения и сообщения об ошибках. В этом разделе возможные причины некоторых из этих сообщений представлены совместно с некоторыми рекомендациями как найти и исправить эти ошибки.

10.1 Системные ошибки

Не хватает памяти для системы, Не хватает памяти, Не хватает памяти для компилятора, ошибка API, Таймер не локализован, Ошибка печати (System out of memory, Out of memory, Compiler out of memory, API error, No timer allocated, Printing error)

Все эти ошибки могут возникнуть, если системные ресурсы сильно ограничены, хотя другие ошибки могут вызывать некоторые из этих сообщений. Закройте все другие программы и/или перезагрузите Windows.

Файл описания типов TATYPE.INI не найден (Type description file TATYPE.INI not found)

Программные файлы были удалены или перемещены из папки TAC Menta. Переустановите TAC Menta.

10.2 Компиляция FBD

Во время компиляции FBD, т.е. при переходе из режима Редактирования в режим Иммитации в основной программе TAC Menta, могут быть обнаружены ошибки синтаксиса. Блок, вызвавший ошибку будет выделен и помещен в центр.

Тип датчика 0..1В может давать недостаточное разрешение (Sensor type 0..1V may give poor resolution)

Не ошибка, только предупреждение, что значение уставки по умолчанию в поле Датчик в параметре *AI - линейный аналоговый вход* не было изменено. В большинстве случаев лучшим выбором являются значения 0..10V или 2..10V.

Неподсоединенный вход (Unconnected input)

Входы блоков не могут быть оставлены "висящими" (неподсоединенными). Когда эта ошибка появляется, найдите выделенный блок, и соедините вход или входы, которые не подсоединены.

Обнаружена неверно замкнутая обратная связь (Illegal closed loop has been detected)

Обратные связи разрешены в языке программирования; тем не менее, по меньшей мере один RW блок, т.е. блок с внутренней задержкой между входным и выходным сигналом, должен существовать в цикле. Если все блоки, которые составляют обратную связь типа RO это сообщение об ошибке появится. Один из блоков, входящих в обратную связь, будет выделен. Устранение данного типа ошибки заключается в введении RW блока, обычно блока задержки, в цикл обратной связи.

Выход соединен с несовместимыми входами (Output is connected to incompatible inputs)

Данная ошибка появляется только, если оператор одновременно с аналоговым выходом соединен со входом дробного типа и входом целого типа. Для устранения этой ошибки, мы рекомендуем дублировать оператор и соединить каждый из операторов с одним входом.

Требования физических IO превышают возможности выбранного устройства (The physical IO requirements exceed the capabilities of the chosen device).

Эта ошибка возникает, когда одно и тоже соединение используется в нескольких блоках.

Переопределите идентификатор (Redefined identifier)

Эта ошибка возникает, когда два блока, чьи выходы определены как общедоступные сигналы, имеют одинаковые идентификаторы. Чтобы исправить эту ошибку, необходимо изменить название одного из блоков.

Неопределенная константа (Undefined constant)

Эта ошибка появляется, когда идентификатор константы, используемый в блоке или блоках, не задан в таблице констант.

Блок должен иметь имя (Block must have a name)

Эта ошибка возникает, когда общедоступному выходному сигналу блока не дано название (идентификатор).

Общедоступная константа не поддерживается в блоке формул (Public constant not supported in expression blocks)

Общедоступные константы нельзя использовать в блоках формул. Решение - переместить константу в PVB/PVI/PVR блок и

соединить его с блоком формул через вход (или использовать необщедоступную константу).

Общедоступная константа используется более, чем один раз (Public constant used more than once)

Общедоступные константы не могут быть использованы в нескольких параметрах блоков в одно и тоже время. Решение - выбрать разные имена констант для каждого параметра блока (если они должны быть общедоступными) или использовать необщедоступные константы.

Идентификатор общедоступной константы уже используется (Public constant identifier is already used)

Идентификатор сигнала не может существовать одновременно как общедоступная константа и общедоступный сигнал. Константа должна быть либо переименована, либо сделана необщедоступной.

Общедоступная константа не используется (Public constant is not used)

Нельзя определить общедоступную константу и затем ее не использовать в параметрах блока. Она должна быть либо переименована, либо сделана необщедоступной. .

Вход/выход используется повторно в блоке I/O (I/O terminal reused in I/O block)

Либо два I/O блока используют один и тот же вход/выход, либо I/O блок не имеет входов/выходов вообще.

Имя узла слишком длинное (Node name too long)

Название узла слишком длинное. Для информации о синтаксических требованиях см. "[Названия сигналов](#)" на стр. 24.

Число обратных косых должно быть 2 или 3 (The number of backslashes must be 2 or 3)

Число обратных косых линий неправильное. Допускается использование только 2 или 3 обратных косых линии. Для информации о синтаксических требованиях см. "[Названия сигналов](#)" на стр. 24.

10.3 Сохранение и загрузка прикладной программы в базу данных

Нельзя сохранить mta файл (Could not save mta file)

Mta файл не может быть сохранен в базе данных TAC Vista.

Ошибка при обновлении базы данных (Failed to update the database (reserve after inject))

Прикладная программа не может быть сохранена в базе данных TAC Vista.

Не возможно загрузить mta файл.

Файл не может быть открыт из базы данных TAC Vista (File could not be opened from the TAC Vista database)

Приложение не может быть открыто. Проверьте правильность работы TAC Vista Server.

Файл занят другим приложением (File object is locked by another application)

Другое приложение (или пользователь) в текущее время работают с файлом и он не доступен.

Невозможно прочесть временный файл (Could not read temporary file)

При работе с приложением используется временный файл. Эта ошибка показывает, что временный файл поврежден или испорчен.

Файл не содержит фактический MTA файл (File object contains no actual MTA file)

Эта ошибка может появиться, если сохранить пустую прикладную программу в базе данных TAC Vista.

Невозможно сохранить файл. Приложение содержит <№> блоков, но только <№> допустимо (Unable to save file. The application contains <no> blocks, but only <no> are allowed).

Эта ошибка возникает при попытке сохранить прикладную программу, которая содержит более 4000 блоков.

10.4 Имитация

Деление на ноль, переполнение разрядной сетки (Division by zero, Numerical overflow)

Логическая ошибка прикладной программы. Перепроектируйте Ваш FBD.

Выполнение остановлено, блок с именем: <имя> и типом: <тип> достиг своего предела (Execution

stopped, the block with name: <name> and type: <type> passed its limit)

Сообщение об ошибке, появляющееся, когда значение сигнала достигло заданного предела (*Stop at a limit*). Процесс имитации при этом останавливается.

10.5 Генерация кода

Символьная таблица переполнена, Величина не умещается в байт (Symbol table full, Value doesn't fit in byte)

В некоторых случаях приложение содержит слишком много сигналов, блокирующих генерацию дерева меню панели оператора. Запустите *Программу конфигурации панели оператора (OP Configuration Tool)* и удалите часть разделов дерева меню.

**Обратите внимание!**

Если в пункте меню *Свойства* стоит галочка "Автоматическая генерация дерева меню", снимите ее или новое дерево меню будет перезаписано при следующей генерации дерева меню панели оператора.

Ошибка синтаксиса в выходном файле, Ошибка синтаксиса в генерируемом файле (Software error parsing output file, Software error parsing generated file)

В TAC Menta невозможно запустить на выполнение приложение пока не закончится процесс генерации кода, тем не менее в большинстве случаев ошибок в FBD нет. Попробуйте еще раз!

FBD с тестовыми блоками не может выполняться в режиме он-лайн, Невозможно сгенерировать COD файл с тестовыми блоками (An FBD with test probe cannot be executed online, Can't generate a COD file with test probes)

Прикладная программа, содержащая тестовые блоки не может быть сгенерирована в исполняемый код и загружена в контроллер. Это сообщение об ошибке также может появляться в случае, если ошибка, которая не имеет своего сообщения об ошибке, появится в процессе генерации кода.

Ошибка синтаксиса в коде (Parser code error)

Приложение может использовать слишком много памяти. Пожалуйста, свяжитесь со службой техподдержки TAC.

Превышено максимальное число блоков аварий в приложении. Проблема должна быть устранена, чтобы приложение можно было сгенерировать в код (The maximum number of Alarm blocks is exceeded in the application. The problem must be solved before the application can be generated)

Приложение может содержать не более чем 127 блоков аварийных сообщений. Число блоков аварийных сообщений не должно превышать 127.

Обработка TACSnvtCfg.dll завершилась неуспешно! Не возможно использовать SNVT в приложении (The TACSnvtCfg.dll was not instantiated successfully! It will not be possible to use SNVTs in the applications)

Библиотека TACSnvtCfg.dll была удалена из папки C:\Program Files\Common Files\TAC Shared. Возможно это произошло вследствие удаления программы. Пожалуйста, переустановите TAC Menta, если Вы хотите иметь доступ к SNVT в приложении.

Данная SNVT не может быть использована, поскольку ни одна единица измерения для нее не доступна (This SNVT could not be used because all engineering units are not available for it)

Описанная SNVT не доступна в файле Taxif.ini. Пожалуйста, свяжитесь для подробной информации со службой техподдержки TAC.

10.6 Загрузка

Если ошибка появляется в процессе загрузки в устройство TAC Xenta, TAC Menta показывает сообщение об ошибке и открывает .COD файл с выделенной подозреваемой ошибкой. Код ошибки отображается в той части .COD файла, где загрузка была прервана. Эта информация может пригодится при устранении неисправностей.

Обновление описание сетевого окружения из TAC Vista Server не выполнено (Update of description of network neighbourhood from TAC Vista Server is not performed)

Эта ошибка появляется, когда для загрузки используется последовательный порт (RS-232) в устройстве TAC Xenta. Появление этой ошибки говорит о том, что сетевое окружение не обновляется, как при загрузке через TAC Vista.

10.6.1 Коды ошибок

№.	Ошибка	Комментарий
1	Создание объекта	
2	Распределение флэш	
3	Сохранение модуля I/O	
4	Версия	
5	Название приложения	
6	Аббревиатура	
7	Тип	
8	Исполняемый файл	
9	Сигналы	
10	Текст аварии	
11	Расписание	
12	I/O модуль	
13	I/O	Контакт возможно не существует с этим типом ед.измерения
14	Время цикла	
15	Код	
16	DST	
17	Контрольная сумма	
18	EOF	
	Распределение памяти	Приложение слишком велико

10.7 Связь TAC Xenta

Истекло время ожидания установления соединения, Истекло время ожидания, Неизвестная ошибка от удаленного узла (Communication time-out, Time-out, Unknown error from remote unit)

Проблемы связи; узел TAC Xenta не отвечает. Попробуйте еще раз.

Ошибка связи, Ошибка при загрузке в..., Ошибка при загрузке из... (Communications failed, Upload failed, Download failed)

Устройство TAC Xenta может прервать соединение. Попробуйте установить его еще раз или перезапустите устройство TAC Xenta вручную (например, выключив его питание). Перед этим сообщением об ошибке может появиться другое, объясняющее в дальнейшем проблему.

Ошибка перезапуска устройства (Error restarting unit)

Устройство TAC Xenta слишком занято, чтобы ответить на вопрос. Попробуйте еще раз.

СПРАВОЧНЫЙ РАЗДЕЛ

Главы

- A Введение**
- B Функциональные блоки**
- C Формулы**
- D Операторы**
- E Тестовые блоки**
- F Рекомендации по
программированию**
- G Поддерживаемые SNVT**

А Введение

А.1 Общие сведения

В данном руководстве описываются все основные функции, доступные графическому языку программирования TAC Menta. Вопросы по использованию программного обеспечения TAC Menta освещены во втором разделе данного руководства.

В главе В "Функциональные блоки" 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:

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 С "Формулы" на стр. 217 содержит описание **Блока формул (Expression)**.

Глава D "Операторы" на стр. 223 содержит описание **Операторов (Operator)**.

Глава E "Тестовые блоки" на стр. 227 содержит описание **Тестовых блоков (Test Probe Block)**.

A.1 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):

В = термистор

U = универсальный вход

X = цифровой вход

Y = аналоговый вход

K = релейный выход

A.1 Типы сигналов

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

Целые (Integer) 16 бит (диапазон: от -32 768 до 32 767)

Дробные (Real) 32 бита в IEEE формате с точностью до 7 знаков (диапазон: от $3,4 \cdot 10^{-38}$ до $3,4 \cdot 10^{38}$)

Бинарные (Binary) 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.

В Функциональные блоки

V.1 Обзор

V.1.1 Блоки I/O (Входов/Выходов)

Сокращение	Короткое описание
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.

V.1.2 Источники сигнала

Сокращение	Краткое описание
NCYC	Счетчик программного цикла.
OSC	Генератор периодических импульсов.
PVB	Бинарная переменная.
PVI	Целая переменная.
PVR	Дробная переменная.

V.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, выдающий на выходе одиночный импульс длительностью в один цикл при изменении входного сигнала
XOR	XOR-функция на 2 бинарных сигнала.

V.1.4 Нелинейные функции

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

V.1.5 Блоки задержки

Сокращение	Короткое 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.

V.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).
PIDP	PID-регулятор (Аналоговый выход).
RAMP	Ограничитель скорости изменения сигнала
SEQ	Блок последовательного управления.

V.1.7 Сумматоры

Сокращение	Краткое описание
ACCUM	Сумматор дробных сигналов.
INTEG	Интегратор (целые сигналы).
RT	Учет времени работы.

V.1.8 Системные параметры

Сокращение	Краткое описание
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 TAC Xenta. По этой причине все блоки данной группы относятся к так называемым источникам сигналов, поскольку у них нет входов.

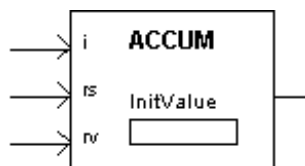
V.1.9 Расписание и сигнал тревоги

Сокращение	Краткое описание
ALARM	Функция выдачи аварийного сообщения
TSCH	Расписание

V.1.10 Передаточные функции

Сокращение	Краткое описание
CURVE	Линейный график.
ENTH	Расчет энтальпии.
POLY	Полином.
PRCNT	Проценты.
VECTOR	Вектор.

B.2 ACCUM - Сумматор



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	

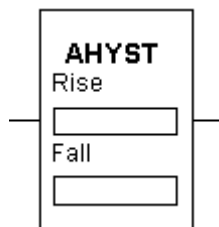
Описание Этот блок используется для суммирования общего приращения за время одного программного цикла. Рассчитывается сумма изменения *Приращения* за время цикла. Суммирование производится с большой цифровой точностью во избежание ошибок в случае малого приращения.

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.

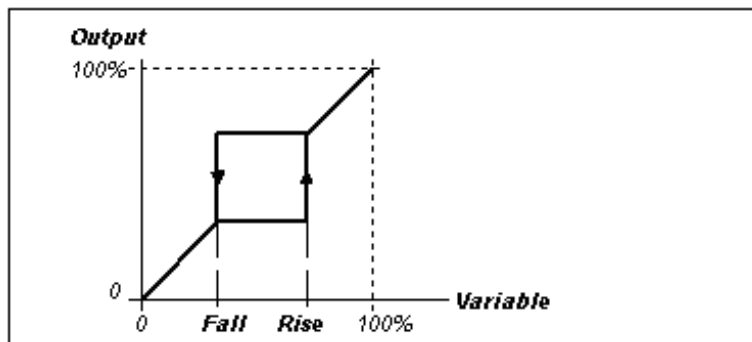
B.3 АНУСТ - Аналоговый гистерезис



ВХОДЫ	Переменная	ДРОБНЫЙ	Входной сигнал
ПАРАМЕТРЫ	Подъем	ДРОБНЫЙ	Значение входного сигнала при подъеме в области гистерезиса
	Спад	ДРОБНЫЙ	Значение входящего сигнала при спаде в области гистерезиса
ТИП ВЫХОДА		ДОСТУП	
ДРОБНЫЙ		R/W	

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 ANYST-блоков, которое будет соответствовать числу областей гистерезиса в желаемой функции передачи.

V.4 AI - Аналоговый вход



AI-блок имеет 5 различных режимов, настраиваемых пользователем:

- Линейный аналоговый вход (Linear Analog input)
- Нелинейный аналоговый вход (Non-linear Analog input)
- Сетевая переменная (Network variable)
- SNVT
- Постоянное значение (Constant value)

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.



Обратите внимание!

Если Вы используете ТАС Menta версии 4.0 и систему версии 3.5 устройства ТАС Xenta, возможно задать AI блоку использование системы измерения СИ или I-P для определенных параметров.

В.4.1 Линейный аналоговый вход (Linear Analog Input)

ПАРАМЕТРЫ	Номер модуля (Mod Number)	ЦЕЛЫЙ	Номер I/O-модуля.
	Название контакта (Terminal Ref)	СПИСОК	Название контакта.
	Тип датчика (Sensor Type)	СПИСОК	Тип датчика.
	Min значение (Min Value)	ДРОБНЫЙ	Нижняя граница диапазона входящего сигнала.
	Max значение (Max Value)	ДРОБНЫЙ	Верхняя граница диапазона входящего сигнала.
	Постоянная времени (Time Const)	ДРОБНЫЙ	Постоянная времени для фильтра (сек).
	Начальное значение (Initial Value)	ДРОБНЫЙ	Начальное значение на выходе (только для I/O-модуля) Значение по умолчанию = 0.
ТИП ВЫХОДА		ДОСТУП	
ДРОБНЫЙ		RO	

Описание	<p>В режиме <i>Линейный аналоговый вход (Linear Analog Input)</i> блок получает значение с аналогового входа от линейного датчика. Значение на выходе блока представляет собой значение на аналоговом входе, преобразованное в соответствующие единицы измерения.</p> <p>Преобразование в различные единицы измерения определяется, когда поля <i>Min Значение</i> и <i>Max Значение</i> определяются в зависимости от выбранного типа датчика.</p> <p>Параметр <i>Номер модуля</i> представляет собой номер I/O-модуля (0 = основное устройство). Параметр <i>Название контакта</i> задает тип и номер контакта (напр.: V1-V4, U1-U4).</p> <p>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 <i>Min значение</i> и <i>Max значение</i> представляют собой нижнюю и верхнюю границы диапазона возможного значения на датчике, напр.: 0-40 оС, соответствующие наименьшему и наибольшему значению входного электрического сигнала.</p> <p>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 <i>Постоянная времени</i>.</p> <p>Алгоритм работы фильтра следующий</p> $y(k) = y(k-1) + \frac{1}{1 + \frac{T}{h}}(u(k) - y(k-1))$ <p>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.</p> <p>Данный алгоритм работы фильтра основан на приближенном значении <i>Обратное дифференцирование</i> оператора бесконечной производной времени. За более подробным объяснением обращайтесь, например, к изданию Острема и Виттенмарка "<i>Системы, контролируемые компьютером - Теория и дизайн</i>". (Astrom and Wittenmark: "Computer Controlled Systems - Theory and design". Prentice-Hall, Englewood Cliffs (1984)).</p>
-----------------	--

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.

V.4.2 Нелинейный аналоговый (термисторный) вход (Non-linear Analog Input)

ПАРАМЕТРЫ	Номер модуля (Mod Number)	СПИСОК	Название I/O-модуля.
	Название контакта (Terminal Ref)	СПИСОК	Название контакта.
	Тип датчика (Sensor Type)	СПИСОК	Тип датчика.
	Постоянная времени (Time Const)	ДРОБНЫЙ	Постоянная времени для фильтра (сек).
	Начальное значение (Initial Value)	ДРОБНЫЙ	Начальное значение на выходе (только для I/O-модуля). Значение по умолчанию = 0.
ТИП ВЫХОДА		ДОСТУП	
ДРОБНЫЙ		RO	

Описание	<p>В режиме Термисторный вход (Нелинейный аналоговый вход) блок получает значение с аналогового входа от нелинейного датчика, т.е. термистора.</p> <p>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.</p> <p>Единицы измерения при выборе в поле <i>Датчик</i>: либо единиц оС, либо оF. Вход блока будет представлять значение соответствующее выбранной единице измерения.</p> <p>Параметр <i>Номер модуля</i> представляет название I/O-модуля или основного устройства. Параметр <i>Название контакта</i> задает тип и номер контакта (напр.: В1 - В4, U1 - U4)..</p> <p>Преобразование значения сигнала в единицу измерения задается параметром <i>Тип датчика</i>. Последний необходимо выбрать из предложенного списка. Выбор <i>SP adjust</i> производится, если ко входу подсоединен потенциометр от комнатного термостата серии ZS 100.</p> <p>После преобразования, значение сигнала может быть пропущено через фильтр первого порядка, входящий в функциональный блок. Постоянная времени для фильтра задается в секундах параметром <i>Постоянная времени</i>.</p> <p>Алгоритм работы фильтра будет тот же, что и при конфигурации Линейный аналоговый вход.</p> <p>При подключении входа к I/O модулю, начальное значение на выходе блока (до получения значения сигнала с I/O модуля) задается параметром <i>Начальное значение</i>. При отключении I/O модуля, выход блока сохранит последнее значение, полученное с I/O модуля.</p>
-----------------	--

V.4.3 Сетевая переменная (Network Variable)

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

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

Сетевой адрес представляет собой общедоступный сигнал в другом блоке TAC, представленный в виде строки, напр.: \RPU1\АНУ1\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 используется). 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!

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.

B.4.4 SNVT

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

Описание В режиме SNVT блок используется для передачи через сеть сигнала из внешнего оборудования в программу приложения. Блок входа 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).

SNVT намеренно представляются в единицах измерения системы СИ, но могут быть установлены для представления в единицах системы измерения I-P. Для перехода к другой системе измерения надо выбрать национальные уставки в ОС Windows на компьютере, на котором создается прикладная программа. Использование метрической системы измерения создаст блоки, в которых значения SNVT будут в единицах измерения СИ. Использование Американской системы измерения приведет к созданию блоков с I-P единицами измерения.

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 за время *Периода*.

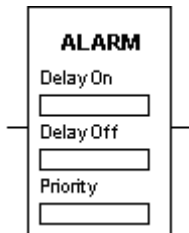
В.4.5 Постоянное значение (Constant Value)

ПАРАМЕТРЫ	Начальное значение (Initial Value)	ДРОБНОЕ	Начальное значение на выходе.
ТИП ВЫХОДА	ДОСТУП		
ДРОБНЫЙ	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 (Initial Value)*.

B.5 ALARM - Авария



ВХОДЫ	Вход (Input)	БИНАРНЫЙ	Сигнал на входе.
ПАРАМЕТРЫ	Задержка включения (DelayOn)	ДРОБНЫЙ	Задержка перед включением аварийной функции.
	Задержка выключения (DelayOff)	ДРОБНЫЙ	Задержка перед выключением аварийной функции.
	Приоритет (Priority)	ЦЕЛЫЙ	Уровень приоритета аварии.
	Текст аварии (AlarmText)	СПИСОК	Текст аварии.
ТИП ВЫХОДА		ДОСТУП	
ДРОБНЫЙ		RO	

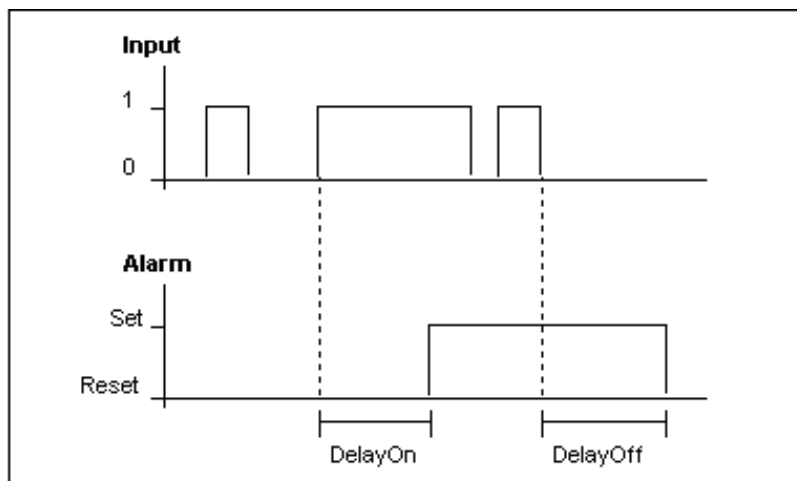
Описание

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*, соответственно. 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.

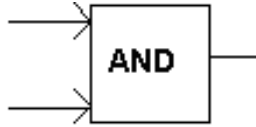
Уровень приоритета аварии задается параметром *Приоритет*. Данное значение не производит никаких действий в программе приложения. Оно всего лишь является составной частью той информации, которая входит в выдаваемое аварийное сообщение. Аварийные сообщения хранятся в специальном аварийном списке, который обрабатывается системной программой. Обработку можно осуществлять, например, через панель оператора. Когда сигнал аварии посылается в TAC 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 TAC Vista, а также *не* отображаются в списке аварий.



Обратите внимание!

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.

V.6 AND - Логическая функция "И"



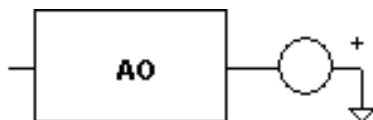
ВХОДЫ	Вход1	БИНАРНЫЙ	
	Вход2	БИНАРНЫЙ	
ТИП ВЫХОДА		ДОСТУП	
ДРОБНЫЙ		RO	

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

B.7 АО - Аналоговый выход



АО-блок имеет 3 различных режима, настраиваемых пользователем:

- Физический выход (Physical output)
- SNVT
- Не подсоединен (Not connected)



Обратите внимание!

Если Вы используете TAC Menta версии 4.0 и системную программу версии 3.5 в устройстве TAC Xenta, возможно задание в блоке АО использования для определенных параметров единиц измерения Си или I-P.

B.7.1 Физический выход

ВХОД	Вход (Input)	ДРОБНЫЙ	Сигнал на входе, 0-100%
ПАРАМЕТРЫ	Номер модуля (Mod Number)	СПИСОК	Название I/O модуля.
	Название контакта (Terminal Ref)	СПИСОК	Название контакта.
	Напряжение 0% (Voltage 0%)	ДРОБНЫЙ	Сигнал на выходе, если входящий сигнал 0% (Вольт)
	Напряжение 100% (Voltage 100%)	ДРОБНЫЙ	Сигнал на выходе, если входящий сигнал 100% (Вольт)
	Начальное значение (Initial Value)	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 название I/O-модуля или основного устройства. 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.

B.7.2 SNVT

ПАРАМЕТРЫ	Тип (Type)	СПИСОК	Тип SNVT.
	Поле (Member)	СПИСОК	Тип SNVT -поля (Применяется только после выбора структурированного типа SNVT).
	SNVT-название (SNVT Name)	СПИСОК	ID сигнала в сети, максимум 16 знаков.
	Нач.значение (Initial Value)	ДРОБНЫЙ	Начальное значение на выходе. Значение по умолчанию = 0
	Отправить (Send)	БИНАРНЫЙ	Определяет, должен ли внешний сигнал обновляться автоматически или нет.
	Дельта (Delta)	ДРОБНЫЙ	Наименьшее изменение значения входящего сигнала, который будет передаваться на вход подсоединенного устройства. Значение по умолчанию = 0,5
	Период (Period)	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., *OutdoorTemp*) и в сети (*SNVT - Имя*, напр., *nvooutdoortemp*).

SNVT представляются в единицах измерения СИ, но могут быть также представлены и в единицах измерения системы I-P. Выбор осуществляется выбором национальных установок в операционной системе Windows на ПК, где создается прикладная программа. Использование метрической системы приведет к созданию блока с единицами измерения системы СИ. Использование американской системы приведет к созданию блоков с единицами измерения системы I-P.

Значение параметров *Тун* 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 изменения на величину *Delta*.

Если функция *Отправить* не выбрана, для получения обновленных значений внешнее оборудование должно использовать функцию *Опрос*. Полученное значение будет обновляться лишь в случае отклонения значения сигнала на входе блока от последнего записанного значения на величину, превышающую *Delta*.

В.7.3 Не подсоединен (Not connected)

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

В.8 CNT - Цифровой вход - Счетчик импульсов



CNT-блок, используя режимы I/O, может быть подключен либо в режиме счетчика импульсов (Pulse counter), либо в режиме Не подсоединен (Not connected).

В.8.1 Счетчик импульсов

ПАРАМЕТРЫ	Номер модуля (Mod Number)	СПИСОК	Название I/O - модуля.
	Название контакта (Terminal Ref)	СПИСОК	Название контакта.
	Полярность (Normally Open)	БИНАРНЫЙ	Если метка стоит на "закрыт" (Close), то переключение будет по переднему фронту (по умолчанию). Если метка стоит на "открыт" (Open), то переключение будет по заднему фронту.
	Множитель (Multiplier)	ДРОБНЫЙ	Фактор шкалы. Значение по умолчанию = 1.0
ТИП ВЫХОДА		ДОСТУП	
ДРОБНЫЙ		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*.

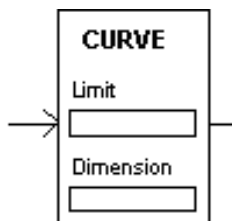
Параметр *Номер модуля* представляет собой название I/O-модуля или основного устройства. Параметр *Название контакта* задает тип и номер контакта на входе (напр.: U1-U4, X1-X4).

В зависимости от заданного значения параметра *Полярность (Normaly Open)*, счетчик переключается либо по переднему, либо по заднему фронту входного цифрового сигнала:
Если метка стоит на "закрыт" (Close), то переключение будет по переднему фронту (по умолчанию).
Если метка стоит на "открыт" (Open), то переключение будет по заднему фронту.

V.8.2 Не подсоединен (Not connected)

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

V.9 CURVE - Линейный график



ВХОДЫ	Вход (Input)	ДРОБНЫЙ	
ПАРАМЕТРЫ	Граница (Limit)	БИНАРНЫЙ	Выбор функции ограничения (граница = 1) или линейной экстраполяции (граница = 0).
	Координаты x,y (Dimension (Pair list x,y))	ДРОБНЫЙ	Список пар координат x,y, задающих место изгиба (точки) графика.
ТИП ВЫХОДА		ДОСТУП	
ДРОБНЫЙ		RO	

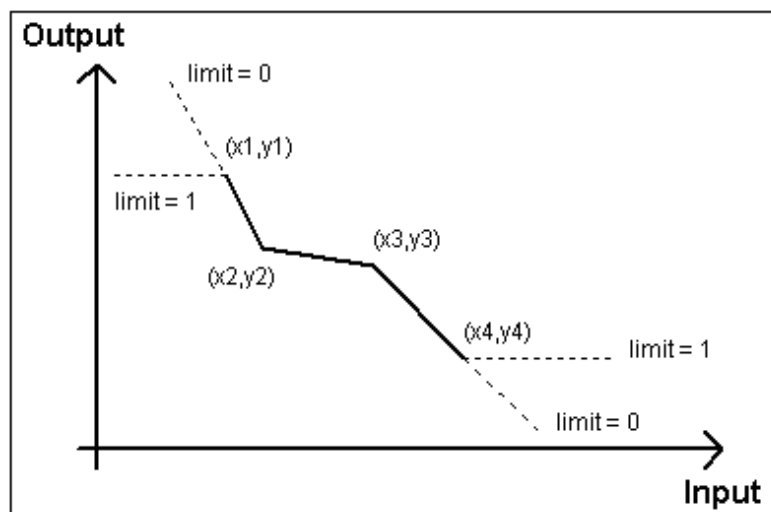
Описание

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 строке. Значения координат x вводятся в список по порядку, т.е. $x_{i-1} < x_i < x_{i+1}$ (x на верхней строке всегда должен быть меньше того, который снизу).

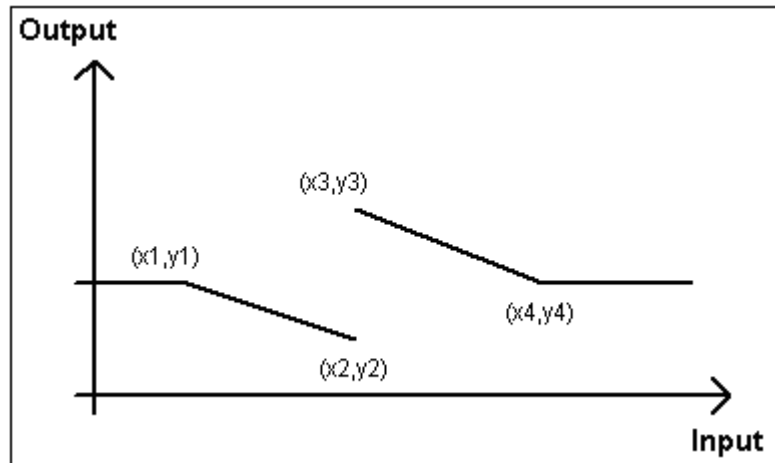
Значение входящего сигнала сопоставляется с координатами x точек изгиба, по порядку. Первая пара точек координат, в которой значение x больше значения входящего сигнала принимается за конечную точку линейного отрезка с началом в предыдущей точке. Значение выходящего сигнала вычисляется через линейную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

При установке параметра Граница (Limit) 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 (т.е. Limit = 0), 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 .

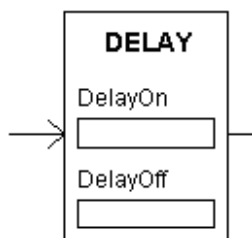
V.10 DATE - Дата



ТИИ ВЫХОДА	ДОСТУП	
ЦЕЛЫЙ	RO	

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

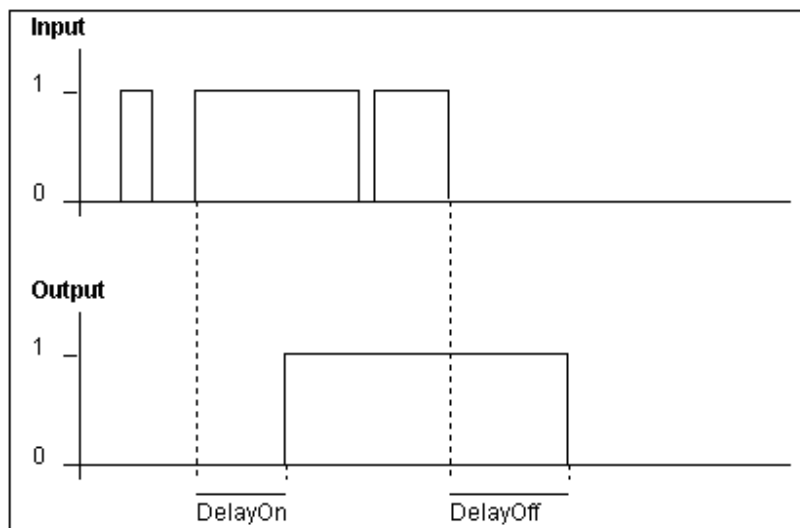
B.11 DELAY - Задержка включения/выключения



ВХОДЫ	Вход (state)	БИНАРНЫЙ	Входной сигнал
ПАРАМЕТРЫ	Задержка вкл. (DelayOn)	ДРОБНЫЙ	Задержка включения в секундах
	Задержка выкл. (DelayOff)	ДРОБНЫЙ	Задержка выключения в секундах
ТИП ВЫХОДА		ДОСТУП	
БИНАРНЫЙ		RO	

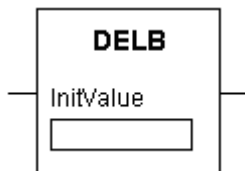
Описание

Данный блок задерживает изменение состояния входного 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, чтобы на выходе появилась "1" время входного положительного сигнала (1) должно превышать время задержки на включение (см. нижеприведенную диаграмму времени). Для переключения выхода на значение ЛОЖЬ, входной сигнал должен иметь значение ЛОЖЬ (0) 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.

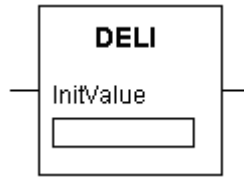
V.12 DELB - Задержка на 1 цикл (бинарный сигнал)



ВХОДЫ	Вход (state)	БИНАРНЫЙ	Бинарный входной сигнал.
ПАРАМЕТРЫ	Нач.значение (InitValue)	БИНАРНЫЙ	Начальное значение сигнала на выходе
ТИП ВЫХОДА		ДОСТУП	
БИНАРНЫЙ		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.

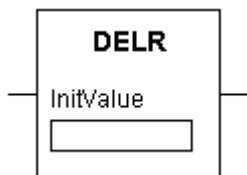
V.13 DELI - Задержка на 1 цикл (целый сигнал)



ВХОДЫ	Значение (variable)	ЦЕЛЫЙ	Входной сигнал.
ПАРАМЕТРЫ	Нач.значение (InitValue)	ЦЕЛЫЙ	Начальное значение сигнала на выходе.
ТИП ВЫХОДА		ДОСТУП	
ЦЕЛЫЙ		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.

V.14 DELR - Задержка на 1 цикл (дробный сигнал)



ВХОДЫ	Значение (variable)	ДРОБНЫЙ	Входной аналоговый сигнал
ПАРАМЕТРЫ	Нач. значение (InitValue)	ДРОБНЫЙ	Начальное значение сигнала на выходе.
ТИП ВЫХОДА		ДОСТУП	
ДРОБНЫЙ		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.

B.15 DI - Цифровой вход



DI-блок имеет 5 различных режимов, настраиваемых пользователем:

- Физический цифровой вход (Physical digital input)
- Сетевая переменная (Network variable)
- SNVT
- Устройство На-линии (Online device)
- Постоянное значение (Constant value)

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.

B.15.1 Физический цифровой вход

ПАРАМЕТРЫ	Номер модуля (Mod Number)	СПИСОК	Название I/O-модуля.
	Название контакта (Terminal Ref)	СПИСОК	Название контакта.
	Полярность (Normally Open)	БИНАРНЫЙ	Если метка стоит на "закрыт" (Close), то переключение будет по переднему фронту (по умолчанию). Если метка стоит на "открыт" (Open), то переключение будет по заднему фронту.
	Нач.значение (Initial Value)	БИНАРНЫЙ	Начальное значение на выходе (только I/O модуль); Вкл. или Выкл. Значение по умолчанию = Off (Выкл).

OUTPUT TYPE	ACCESS	
BINARY	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 *Полярность (Normaly Open)*. Если метка стоит на "закрыт" (Close), то переключение будет по переднему фронту (по умолчанию). Если метка стоит на "открыт" (Open), то переключение будет по заднему фронту. 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.

V.15.2 Сетевая переменная

ПАРАМЕТРЫ	Сетевой адрес (Network Address)	СПИСОК	Соотносится с внешним сигналом сети TAC
	Начальное значение (Initial Value)	БИНАРНЫЙ	Начальное значение на выходе; Вкл.(1)/ Выкл.(0). Значение по умолчанию=Off.
	Период (Period)	ЦЕЛЫЙ	Максимальный интервал времени (в секундах) между двумя обновлениями посланного значения. Значение по умолчанию = 60.
ТИП ВЫХОДА		ДОСТУП	
БИНАРНЫЙ		RO	

Описание В режиме *Сетевая переменная* блок используется для получения сигнала от других устройств TAC через сеть. *Сетевой адрес* представляет собой адресобщедоступного (Public) бинарного сигнала в другом устройстве TAC, представленный в виде строки, напр: \RPU1\AHU1\ExtendedOp, где RPU1 - название устройства в сетевой конфигурации, AHU1 - название модуля (если он есть), в котором расположен интересующий нас сигнал, ExtendedOp - название бинарного сигнала в приложении другого устройства, показания которого нас интересуют (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 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 входного сигнала. Для каждого сигнала используется только один блок, который подключается к другим блокам в схеме FBD, использующей посланный сигнал.

B.15.3 SNVT

ПАРАМЕТРЫ	Тип (Type)	СПИСОК	Тип SNVT.
	Поле (Member)	СПИСОК	Тип SNVT -поля (Применяется только после выбора структурированного типа)
	SNVT-название (SNVT Name)	СПИСОК	ID сигнала в сети, максимум 16 знаков.
	Нач.значение (Initial Value)	БИНАРНЫЙ	Начальное значение на выходе.Значение по умолчанию = Выключить (0).
	Опрос (Poll)	БИНАРНЫЙ	Определяет, должно ли быть запрошено значение внешнего сигнала на выходе.
	Период (Period)	ЦЕЛЫЙ	Максимальный интервал времени (в секундах) между двумя обновлениями посланного значения.Значение по
ТИП ВЫХОДА		ДОСТУП	
БИНАРНОЙ		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*..

V.15.4 Устройство На-линии (Online Device)

ПАРАМЕТРЫ	Адрес устройства (Device Address)	СТРОКА	Сетевой адрес инспектируемого устройства или номер модуля расширения.
ТИП ВЫХОДА		ДОСТУП	
БИНАРНЫЙ		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 TAC Vista, в качестве названия устройства через программу Сетевая конфигурация должно вводиться имя LonWorks network. Обратите внимание на тот факт, что контроллер TAC 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).

V.15.5 Постоянное значение (Constant Value)

ПАРАМЕТРЫ	Нач.значение (Initial Value)	БИНАРНЫЙ	Начальное значение на выходе; Включить (1)/Выключить (0). Значение по умолчанию = Выключить (0).
ТИП ВЫХОДА		ДОСТУП	
БИНАРНЫЙ		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 (Initial Value)*.

V.16 DO - Цифровой выход



DI-блок имеет четыре различных режима, настраиваемых пользователем:

- Физический выход (Physical output)
- SNVT
- Не подсоединен (Not connected)

V.16.1 Физический цифровой выход (Physical Output)

ВХОДЫ	Вход (Input)	БИНАРНЫЙ	Сигнал на входе.
ПАРАМЕТРЫ	Номер модуля (Mod Nuber)	СПИСОК	Название I/O - модуля.
	Название контакта (Terminal Ref)	СПИСОК	Название контакта.
	Нач. значение (Initial Value)	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 название I/O-модуля или основного устройства. 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.

B.16.2 SNVT

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

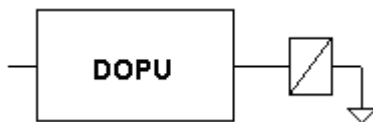
Если выбрана функция *Отправить*, значение внешнего выходного сигнала будет каждый раз обновляться при изменении значения сигнала на входе блока. Независимо от факта изменения значения входного сигнала, внешний выходной сигнал будет обновляться по прохождении времени, заданного параметром *Период*, после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

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

V.16.3 Не подсоединен (Not connected)

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

V.17 DOPU - Цифровой импульсный выход



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

V.17.1 Цифровой импульсный выход

ВХОДЫ	Вход (Input)	ДРОБНЫЙ	Длина импульсного сигнала (сек.).
ПАРАМЕТРЫ	Номер модуля (Mod Number)	НОМЕР	Номер I/O - модуля.
	Название контакта (Terminal Ref)	СПИСОК	Название контакта.
	Мин. импульс (Min Pulse)	ДРОБНЫЙ	Минимальное значение длины импульсного сигнала на выходе (сек) Значение по умолчанию = 0,5 сек
	Нач. значение (Initial Value)	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 название I/O-модуля или основного устройства. Параметр *Название контакта* задает тип и номер контакта на выходе (напр.: K1-K4).

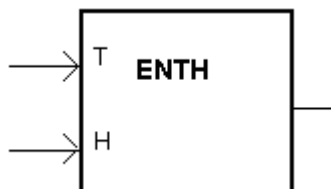
Параметр *Мин. Импульс* задает минимальное значение длины импульсного сигнала на выходе (в секундах). Все значения входящего сигнала, превышающие данное минимальное значение, накапливаются для следующего цикла в программе приложения. Если протяженность импульсного сигнала на входе превышает по времени 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.

V.17.2 Не подсоединен (Not connected)

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

B.18 ENTH - Расчет энтальпии



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

Описание

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



Обратите внимание!

Если Вы используете TAC Menta версии 4.0 и системную программу версии 3.5 в устройстве TAC Xenta, можно задать использование в блоке ENTH системы единиц измерения "СИ" или "P-I".

Блок ENTH будет использовать значения входа Т (температура) в единицах измерения "СИ" или I-P, в зависимости от национальных установок в ОС Windows на Вашем компьютере, где создается прикладная программа. Использование метрической системы приведет к созданию блока с входом Т в оС, а использование американской системы приведет к созданию блока в входом Т в оF.

Ранее упомянутые установки также определяют единицы измерения расчетного значения энтальпии. Метрическая система подразумевает использование кДж/кг. Американская система - BTU/фунт. Рассчитанное значение энтальпии для разных систем измерения различается и не может быть сконвертировано из одних единиц измерения в другие.

Алгоритм для СИ-системы

Давление насыщенного водяного пара при температуре от 0 до 200 оС вычисляется по следующей функции :

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

где

T = Абсолютная температура (К),

p_{ws} = давление насыщения (Па) ,

a

$c_8 = -5.8002206E3$;

$c_9 = 1.3914993$;

$c_{10} = -4.8640239E-2$;

$c_{11} = 4.1764768E-5$;

$c_{12} = -1.4452093E-8$;

$c_{13} = 6.5459673$;

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

$$p_{wsat} = p_{ws}(t+273.15)$$

$$p_w = F \cdot p_{wsat} / 100 \quad (22)$$

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

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

в которых

p_{wsat} = давление насыщенного водяного пара (Па)

t = температура по сухому термометру (°C)

F = 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)



Обратите внимание!

См.: ASHRAE Руководство 1989 - Основы (Глава 6, Психрометрия).- ASHRAE Handbook 1989 - Fundamentals (Chapter 6, Psychrometrics). Неизвестные в уравнении в скобках соотносятся с соответствующими уравнениями в Руководстве ASHRAE.

Алгоритм для системы I-P

Давление насыщенного водяного пара при температуре от -80 до 300 °F вычисляется по следующей функции :

$$pws(T) = \exp(c8/T + c9 + c10*T + c11*Ti + c12*Ti + c13*\ln(T)) \quad (6)$$

где

T= абсолютная температура (oR),

pws = давление насыщения (psia)

а

$$c8 = -1.0440397E+04;$$

$$c9 = -1.1294650E+01;$$

$$c10 = -2.7022355E-02;$$

$$c11 = 1.2890360E-05;$$

$$c12 = -2.4780681E-09;$$

$$c13 = 6.5459673E+00;$$

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

$$pw_{sat} = pws(t+459.67)$$

$$pw = \Phi * pw_{sat} / 100 \quad (24)$$

$$W = 0.62198 * pw / (p - pw) \quad (22)$$

$$h = 0.240 * t + W * (1061 + 0.444 * t) \quad (32)$$

где

pw_{sat} = давление насыщенного водяного пара (psi)

t = температура по сухому термометру (°F)

Φ = относительная влажность (%)

pw = парциальное давление водяного пара во влажном воздухе (psi)

p = давление в окружающей среде (101325 psi)

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

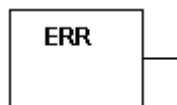
h = энтальпия (кДж/кг)



Обратите внимание!

См.: ASHRAE Руководство 1989 - Основы (Глава 6, Психрометрия).- ASHRAE Handbook 1989 - Fundamentals (Chapter 6, Psychrometrics). Неизвестные в уравнении в скобках соотносятся с соответствующими уравнениями в Руководстве ASHRAE.

B.19 ERR - Системные ошибки



ТИП ВЫХОДА	ДОСТУП	
ЦЕЛЫЙ	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.

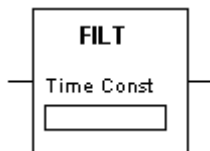
Коды ошибок

№ бита	Знач. вых. 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. Прикладная программа не может быть выполнена в течении времени, приведенного в ее спецификации
6	32	Как минимум один из переключателей одного из блоков расширения (I/O) управляет в ручном режиме (ON, OFF, MAN)
7	64	Меню OP заблокировано желтым или красным уровнем доступа (средний или высокий уровень доступа) (применяется в Xenta системной версии v3.51)
8	128	

9	256	Повреждение на линии, соединяющий модем и устройство ТАС Xenta
10	512	Обновленное значение сетевой переменной не было получено в течение заданного времени
11	1024	Существует неподтвержденный активный сигнал аварии с приоритетом 1 в списке аварий
12	2048	Отсутствие связи в одним или несколькими I/O модулями
13	4096	I/O в принудительном режиме
14	8192	Наличие активного неподтвержденного сигнала аварии в списке аварий
15	16384	Наличие активного и несброшенного аварийного сигнала в списке аварий

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

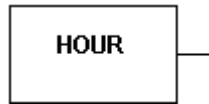
V.20 FILT - Фильтр первого порядка



ВХОДЫ	Переменная (variable)	ДРОБНЫЙ	Входной сигнал
ПАРАМЕТРЫ	Постоянная времени (TimeConst)	ДРОБНЫЙ	Постоянная времени фильтра, измеряемая в секундах.
ТИП ВЫХОДА		ДОСТУП	
ДРОБНЫЙ		R/W	

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

V.21 HOUR - Текущий час

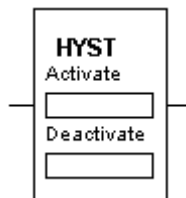


ТИП ВЫХОДА	ДОСТУП	
ЦЕЛЫЙ	RO	

Описание

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

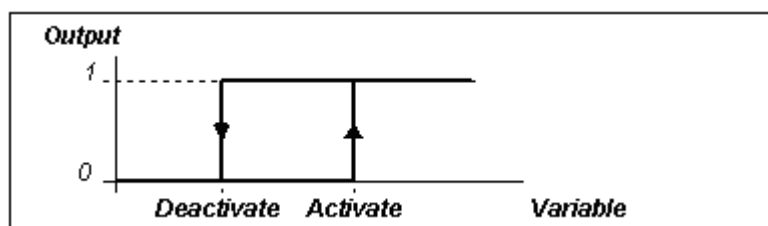
B.22 HYST - Бинарный гистерезис



ВХОДЫ	Переменная (variable)	ДРОБНЫЙ	Входной сигнал.
ПАРАМЕТРЫ	Активный (Activate)	ДРОБНЫЙ	Крайнее значение входного сигнала для активации выходного сигнала.
	Пассивный (Deactivate)	ДРОБНЫЙ	Крайнее значение входного сигнала для дезактивации выходного сигнала.
ТИП ВЫХОДА		ДОСТУП	
ДРОБНЫЙ		RO	

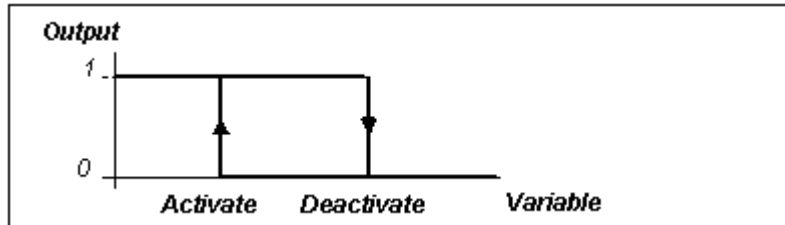
Описание

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 дезактивации, значение на выходе становится ЛОЖЬ (0). В том случае, если входящий сигнал по своему значению находится между границами областей активации и дезактивации, выход блока сохраняет свое первоначальное состояние..

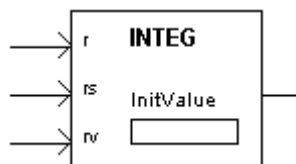


В том случае, когда значение параметра *Активный* меньше, чем значение параметра *Пассивный*, данный блок будет выполнять следующую функцию: если значение на выходе блока ИСТИНА (1), а значение входящего сигнала превышает величину допустимого крайнего значения области дезактивации, то

значение на выходе изменяется на ЛОЖЬ (0). Если значение на выходе блока ЛОЖЬ (0), а значение входящего сигнала меньше значения на границе области дезактивации, значение на выходе становится ИСТИНА (1). В том случае, если входящий сигнал по своему значению находится между границами областей активации и дезактивации, выход блока сохраняет свое первоначальное состояние.



B.23 INTEG - Интегратор



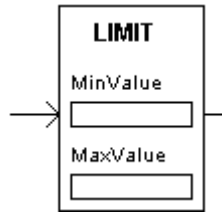
ВХОДЫ	Скорость (r)	ДРОБНЫЙ	Интегрируемый сигнал на входе
	Сброс(rs)	БИНАРНЫЙ	Вход сброса(1 = сброс)
	Предустановка (rv)	ДРОБНЫЙ	Значение на выходе при активизированном входе сброса.
ПАРАМЕТРЫ	Нач.значение (InitValue)	ДРОБНЫЙ	Начальное значение интегратора
ТИП ВЫХОДА		ДОСТУП	
ДРОБНЫЙ		R/W	

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.

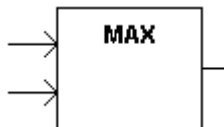
B.24 LIMIT - Ограничение по min/max



ВХОДЫ	Вход (Input)	ДРОБНЫЙ	Входной сигнал
ПАРАМЕТРЫ	Мин.значение (MinValue)	ДРОБНЫЙ	Ограничение по миним. значению
	Макс.значение (MaxValue)	ДРОБНЫЙ	Ограничение по максим. значению
ТИП ВЫХОДА		ДОСТУП	
ДРОБНЫЙ		RO	

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

V.25 MAX - Селектор максимального из 2 сигналов

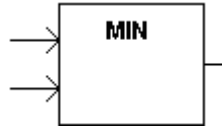


ВХОДЫ	Вход1	ДРОБНЫЙ	Первый входной сигнал
	Вход2	ДРОБНЫЙ	Второй входной сигнал
ТИП ВЫХОДА		ДОСТУП	
ДРОБНЫЙ		RO	

Description

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

V.26 MIN - Селектор минимального из 2 сигналов

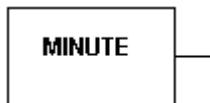


ВХОДЫ	Вход1	ДРОБНЫЙ	Первый входной сигнал
	Вход2	ДРОБНЫЙ	Второй входной сигнал
ТИП ВЫХОДА		ДОСТУП	
ДРОБНЫЙ		RO	

Описание

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

V.27 MINUTE - Текущая минута



ТИП ВЫХОДА	ДОСТУП	
ЦЕЛЫЙ	RO	

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

V.28 MONTH - Текущий месяц

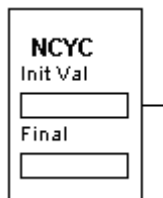


ТИП ВЫХОДА	ДОСТУП	
ЦЕЛЬЙ	RO	

Описание

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

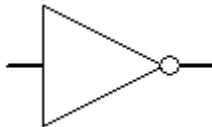
V.29 NCYC - Счетчик программного цикла



ПАРАМЕТРЫ	Нач.значение (InitValue)	ЦЕЛЫЙ	Начальное значение на счетчике.
	Конеч.значение (Final)	ЦЕЛЫЙ	Конечное значение на счетчике.
ТИП ВЫХОДА		ДОСТУП	
ЦЕЛЫЙ		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 при обратном соотношении *Начального* и *Конечного значения*. Когда счетчик доходит до конечного значения, он сбрасывается и начинает счет в следующем цикле с начального значения.

V.30 NOT - Инверсия бинарного сигнала



ВХОДЫ	Вход	БИНАРНЫЙ	Входной сигнал
ТИП ВЫХОДА		ДОСТУП	
БИНАРНЫЙ		RO	

Описание Данный блок инвертирует Бинарный сигнал.

V.31 OPT - Оптимизатор

	OPT	
RoT		Max Start Ti <input type="text"/>
SP	AutoAdjust <input type="text"/>	ETol <input type="text"/>
OuT	Room Temp F <input type="text"/>	Stop Lo Temp <input type="text"/>
Opl	OperIndic F <input type="text"/>	Stop Time Lo <input type="text"/>
	Start Time Lo <input type="text"/>	Stop Hi Temp <input type="text"/>
TmL	Start Time Hi <input type="text"/>	Stop Time Hi <input type="text"/>
Mode	Holiday Comp <input type="text"/>	Stop Time Err <input type="text"/>

ВХОДЫ	RoT	ДРОБНЫЙ	Температура в помещении оС (оF).
	SP	ДРОБНЫЙ	Уставка для температуры в комнате в рабочее время оС (оF).
	OuT	ДРОБНЫЙ	Наружная температура оС (оF).
	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, ОРГ 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	Переключатель автоматической настройки AutoAdjust = 0 => без настройки AutoAdjust = 1 => настройка точек графика AutoAdjust = 2 => настройка точек графика и компенсации для выходных. По умолчанию 2.
	RoomTempF	БИНАРНЫЙ	Флаг датчика температуры комнаты RoomTempF = 0 => нет датчика RoomTempF = 1 => есть датчик по умолчанию 1.

ПАРАМЕТРЫ	OperInicF	БИНАРНЫЙ	Флаг выбора наличия внешнего индикатора работы OperInicF = 0 => нет индикатора OperInicF = 1 => есть индикатор по умолчанию 0.
	StartTimeLo	ДРОБНЫЙ	Время запуска (минуты) при низкой наружной температуре -10 °C (14°F), 10 °C (50 °F) (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 более 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		ДОСТУП	
ЦЕЛЬ		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 $Mode = -1$.

**Обратите внимание!**

Если Вы используете TAC Menta версии 4.0 систему версии 3.5 в устройстве TAC Xenta, то можно задать в блоке ОРТ использование либо системы измерения "СИ", либо системы измерения I-P.

Блок будет использовать на входах *RoT*, *SP*, and *Out ed.* измерения "СИ" или I-P в зависимости от национальных уставок в ОС Windows на компьютере, на котором создается прикладная программа. Использование метрической системы приведет к тому, что значения на входах будут в оС, а использование американской системы измерения приведет к тому, что значения на входах будут в оF.

Параметры блока *StartTimeLo*, *StartTimeHi*, определяющие часть координат для граничных точек, используют фиксированную температуру -10 оС (14 оF), 10 оС (50 оF), и 30 оС (86 оF). Введение значений для времени запуска содержит определение координат.

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 °С), 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 $2^{\circ}C$ ниже уставки.

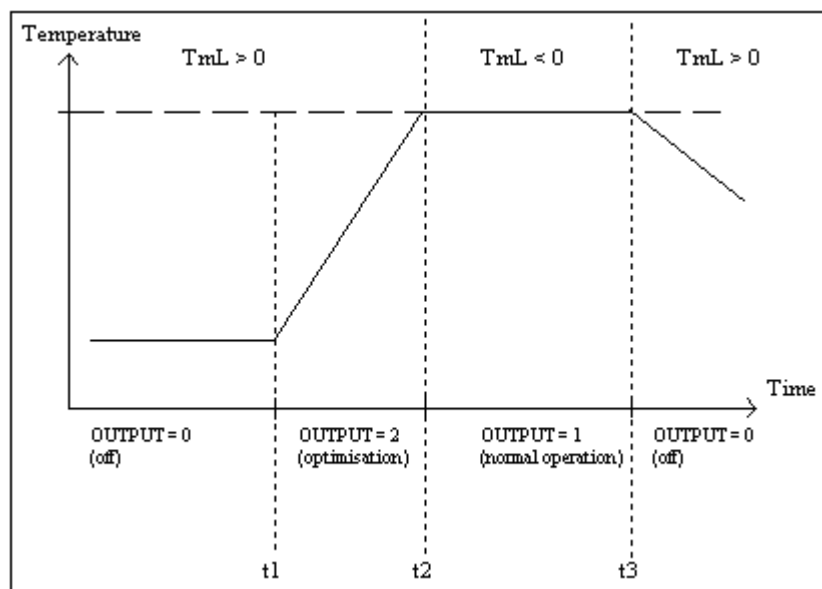


Рис.1 Принцип оптимизации запуска.

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

выход 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 °C и 10 °C для режима нагрева, 10 °C и 30 °C для режима охлаждения. Значения по умолчанию времени запуска 105 минут (*StartTimeLo*) и 45 минут (*StartTimeHi*). Эти точки могут автоматически изменяться, смотри ниже.

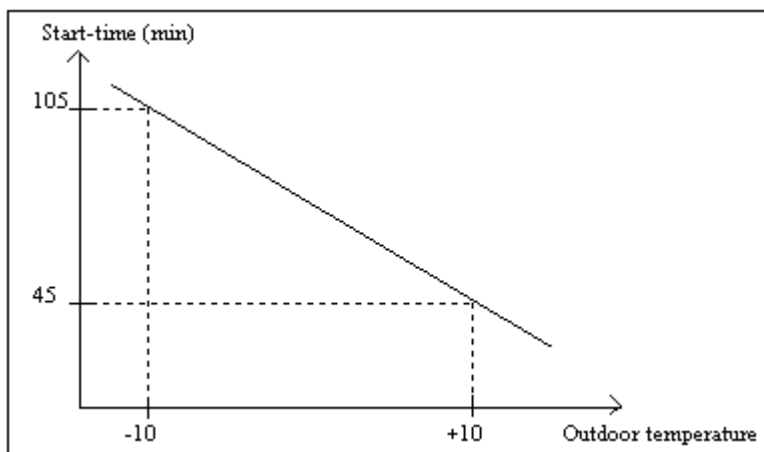


Рис. 2. График описывает зависимость между наружной температурой и временем запуска. График показан для режима обогрева.

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.

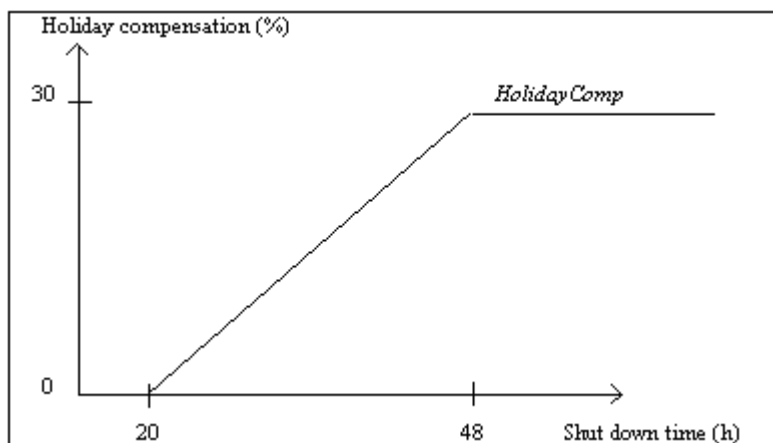


Рис. 3. График зависимости компенсации от длительности выключения. Компенсация выходных = 30%.

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. Необходимо заметить, что точки графика и *HolidayComp* никогда не заданы одновременно после процесса оптимизации. Точки графика могут быть заданы, если установка была выключена менее чем 20 часов, тогда как *HolidayComp* может быть задано только, если установка была отключена более чем 30 часов. Это также подразумевает, что определение этих параметров в целом возможно, если время выключения установки находится между 20 и 30 часами.

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 - X (*StopLoTemp* и *StopHiTemp*) и по оси - Y (*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.

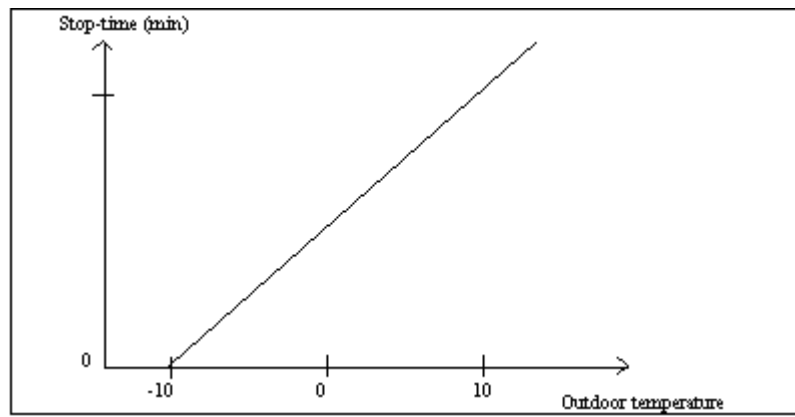
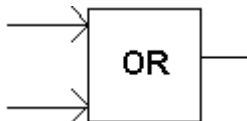


Рис. 4. График отношения наружной температуры и времени остановки. Значение по умолчанию на графике не указано.

V.32 OR - логическая функция "ИЛИ" для 2 бинарных сигналов



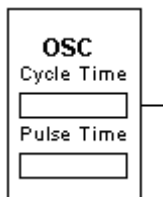
ВХОДЫ	Вход1	БИНАРНЫЙ	Входной сигнал 1
	Вход2	БИНАРНЫЙ	Входной сигнал 2
ТИП ВЫХОДА		ДОСТУП	
БИНАРНЫЙ		RO	

Описание

Данный блок рассчитывает булеву "ИЛИ" -функцию для Входа1 и Входа2 согласно нижеприведенной таблицы истинности:

Вход 1	Вход 2	Выход
0	0	0
0	1	1
1	0	1
1	1	1

B.33 OSC - Осциллятор



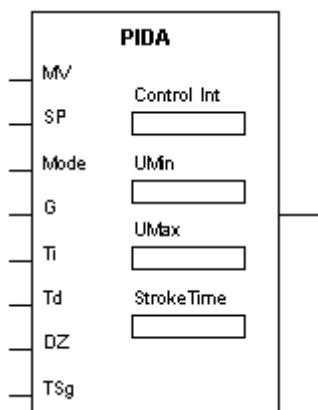
ПАРАМЕТРЫ	Время Цикла (CycleTime)	ДРОБНЫЙ	Период генерирования (секунды)
	Время Импульса (PulseTime)	ДРОБНЫЙ	Длительность импульса (секунды)
ТИП ВЫХОДА		ДОСТУП	
БИНАРНЫЙ		RO	

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.

B.34 PIDA - PID регулятор - Аналоговый выход



ВХОДЫ	MV	ДРОБНЫЙ	Измеряемое значение.
	SP	ДРОБНЫЙ	Заданное значение (уставка).
	Mode	ЦЕЛЫЙ	Рабочий режим регулятора.
	G	ДРОБНЫЙ	Пропорциональное усиление.
	Ti	ДРОБНЫЙ	Время интегрирования (сек).
	Td	ДРОБНЫЙ	Время дифференцирования (сек).
	DZ	ДРОБНЫЙ	Мертвая зона.
	TSg	ДРОБНЫЙ	Сопровождающий сигнал (предшествующее реальное значение управляющего сигнала).
ПАРАМЕТРЫ	ControlInt	ДРОБНЫЙ	Интервал управления (сек).
	UMin	ДРОБНЫЙ	Минимальновозможный управляющий сигнал.
	UMax	ДРОБНЫЙ	Максимальновозможный управляющий сигнал.
	StrokeTime	ДРОБНЫЙ	Время полного хода штока привода (сек)
ТИП ВЫХОДА		ДОСТУП	
ДРОБНЫЙ		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 (уставка) для другого управляющего контура (каскадное управление). Алгоритм управления представляет собой пошаговый PID алгоритм для дискретного времени; изменение управляющего $du(t)$ рассчитывается по формуле:

$$(1) \quad du(t) = G \cdot \left(e(t) - e(t-h) + \frac{h}{T_i} \cdot e(t) - T_d \cdot \frac{y(t) - 2 \cdot y(t-h) + y(t-2h)}{h} \right)$$

где e - ошибка регулирования, y - измеряемое значение (MV), G - усиление регулятора, T_i - время интеграции, T_d - 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 . По умолчанию интервал управления установлен равным 1.

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. $abs(e) < DZ$, величина изменения значения на выходе регулятора, $du(t)$, будет равна нулю. Единицы измерения для параметров Мертвая зона, Измеряемое значение и Заданное значение должны совпадать.

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:

$$(3) \quad P_{band} = \frac{U_{Max} - U_{Min}}{G}$$

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 T_i или T_d соответственно задается равным 0. По аналогии с этим установите функции PI-регулятора, значение $T_i \neq 0$, а $T_d = 0$. 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 P и PD управления, в котором управляющий сигнал вычисляется по формуле (4):

$$(4) \quad u(t) = G \cdot e(t) - T_d \cdot \frac{(y(t) - y(t-h))}{h} + \frac{(U_{Max} + U_{Min})}{h}$$

где U_{Max} - максимальный, а U_{Min} - минимальный возможный управляющий сигнал.

Ограничение значения управляющего сигнала

Максимальная величина изменения значения на выходе за время одного интервала управления, Du_{Max} , зависит от времени хода привода и вычисляется по формуле

$$(5) \quad Du_{Max} = \frac{(U_{max} - U_{Min}) \cdot h}{StrokeTime}$$

Рассчитанная величина изменения значения на выходе регулятора, $du(t)$, должна находиться в диапазоне $\pm Du_{Max}$ до вычисления абсолютного уровня управляющего сигнала. Полученное значение нового управляющего сигнала $u(t)$ ограничена интервалом (U_{Min} , U_{Max}). Если U_{Min} и/или U_{Max} не определены, данное ограничение не выполняется. Параметры U_{Min} и U_{Max} 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 (уставки) для других регуляторов, и если отсутствует необходимость ограничения изменения управляющего сигнала, параметр `StrokeTime` нужно задать равным 0.

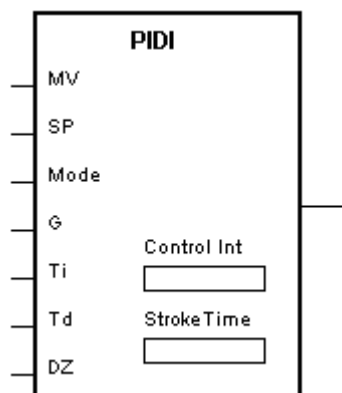
Рабочий режим

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

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

V.35 PIDI - PID регулятор - Увеличить/ Уменьшить



ВХОДЫ	MV	ДРОБНЫЙ	Измеряемое значение.
	SP	ДРОБНЫЙ	Заданное значение (уставка).
	Mode	ЦЕЛЫЙ	Рабочий режим регулятора.
	G	ДРОБНЫЙ	Пропорциональное усиление.
	Ti	ДРОБНЫЙ	Время интегрирования (сек).
	Td	ДРОБНЫЙ	Время дифференцирования (сек).
	DZ	ДРОБНЫЙ	Мертвая зона.
ПАРАМЕТРЫ	ControlInt	ДРОБНЫЙ	Интервал управления (сек).
	StrokeTime	ДРОБНЫЙ	Время полного хода штока привода (сек)
ТИП ВЫХОДА		ДОСТУП	
ДРОБНЫЙ		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:

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

где *StrokeTime* - время полного хода штока привода (в секундах). Обратите внимание, что величина *StrokeTime* соответствует 100% изменению позиции привода. Значение на выходе блока PID1 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 PID1 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 и *dt* = 5,5 с, сигнал на выходе PID1 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:

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

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 $Ti \neq 0$, а $Td = 0$. При усиление *G* равном 0, программа будет выполняться, но управляющий сигнал не изменится, независимо от размера ошибки.

Ограничение значения управляющего сигнала

Рассчитываемое время хода ограничено интервалом $\pm \text{ControlInt}$ (сек), поскольку это максимальное значение времени хода штока

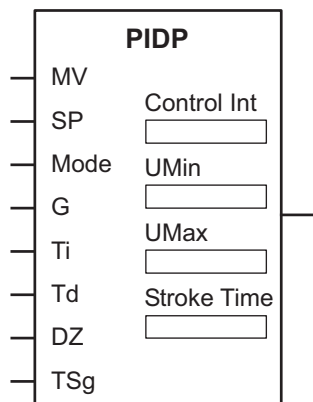
привода, которое возможно в пределах одного управляющего интервала.

Operating Mode

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

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).

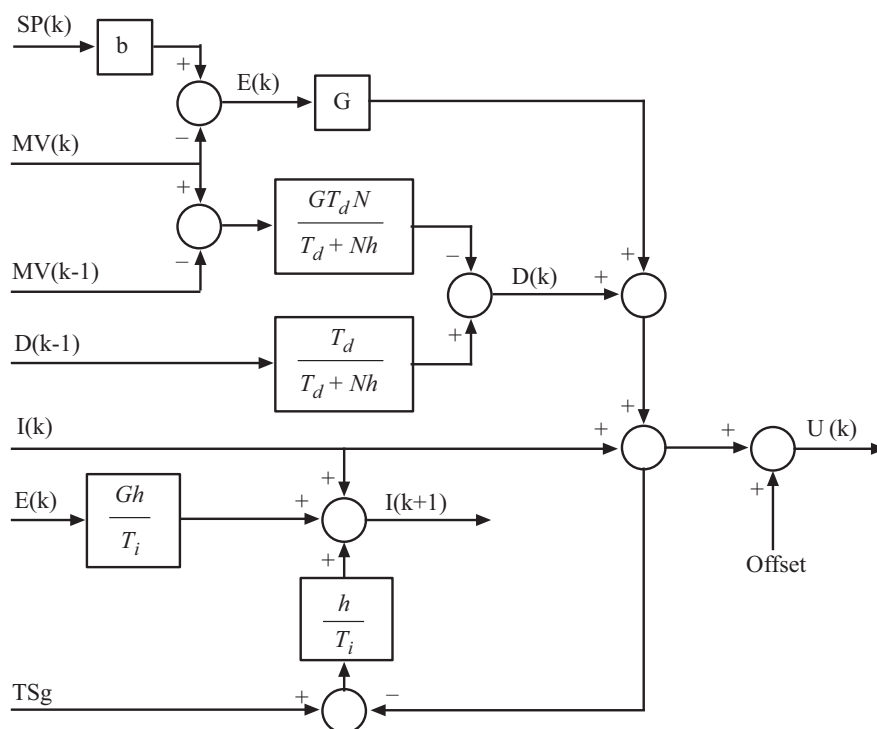
V.36 PIDP - PIDP регулятор - Аналоговый выход

ВХОДЫ	MV	ДРОБНЫЙ	Измеряемое значение.
	SP	ДРОБНЫЙ	Заданное значение (уставка).
	Mode	ЦЕЛЫЙ	Рабочий режим регулятора .
	G	ДРОБНЫЙ	Пропорциональное усиление.
	Ti	ДРОБНЫЙ	Время интегрирования (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	Минимальновозможный управляющий сигнал.
	UMax	ДРОБНЫЙ	Максимальновозможный управляющий сигнал.
	StrokeTime	ДРОБНЫЙ	Время полного хода штока привода (сек).
ТИП ВЫХОДА	ДОСТУП		
ДРОБНЫЙ	R/W		

Описание

Алгоритм управления

Блок PIDP 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 $U(k)$ рассчитывается по формуле:

**PIDP блок**

Здесь SP - это заданное значение, MV - измеряемое значение, G - усиление регулятора, T_i - время интегрирования, T_d - время дифференцирования, а h - интервал управления (*Control Int*), т.е. отрезок времени между двумя успешными обновлениями значения сигнала на выходе регулятора. Если значение параметра *Control Int* задать как 0, то интервал управления автоматически станет равным времени программного цикла.

Индекс времени k представляет собой текущее значение переменной, k-1 является значением алгоритма управления, полученным при предшествующем обновлении и т.д.

Блок PID функционирует с интервалом времени $h \times$ секунд, где $h \times$ секунды соответствуют интервалу времени между исполнениями программы приложения, даже если выбранный интервал управления больше. Интервал управления, h , должен быть кратным времени $h \times$. Если данное условие не выполняется, то алгоритм PID автоматически выберет в качестве управляющего сигнала ближайшее кратное значение, меньшее, чем h . По умолчанию интервал управления равен 1.

В том случае, если ошибка управления меньше значения в мертвой зоне, т.е. $\text{abs}(e) < DZ$, измеряемое значение будет равно значению выходного сигнала на предыдущем шаге, значения на выходе регулятора не изменится. Единицы измерения для параметров Мертвая зона, Измеряемое значение и Заданное значение должны совпадать.

Другие типы регуляторов

Алгоритм, описанный ранее, может быть использован для всех типов регуляторов. При PI или PID регулировании пропорциональная часть не зависит от разницы между заданным значением и измеряемым, а зависит только от измеряемого значения, MV.

Если используется P или PD регулирование, P-функция будет автоматически изменяться в зависимости от ошибки регулирования (SP-MV). При использовании этих регуляторов, значение отклонения регулирования также добавляется к выходному сигналу:

$$\text{Offset} = \frac{(U_{max} + U_{min})}{2}$$

где U_{max} - максимально допустимое значение управляющего сигнала, а U_{min} - минимально допустимое значение управляющего сигнала.

В том случае, если для работы требуется регулятор без I- или D-функций (I - интегрирование, D - дифференцирование), то значение для T_i или T_d соответственно задается равным 0. По аналогии с этим установите функции PI-регулятора, значение $T_i \neq 0$, а $T_d = 0$. При усилении G равном 0, программа будет выполняться, но управляющий сигнал не изменится, независимо от размера ошибки.

Пожалуйста, обратите внимание, что когда любой из параметров G , T_i и T_d изменяется, выполняется автоматическое внутреннее обновление в контроллере. Не следует принимать никаких мер в приложении, чтобы избежать скачков управляющего сигнала.

Ограничение значения управляющего сигнала

Параметр *StrokeTime* необходим для определения времени реального полного хода штока привода. Обратите внимание на то, что *StrokeTime* может быть использован для ограничения величины изменения управляющего сигнала даже, если выход не соединен с приводом.

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, что соответствует значению по умолчанию.

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

$$DuMax = \frac{(U_{max} - U_{min}) \times h}{StrokeTime}$$

Что соответствует максимальному изменению между двумя последующими выходными сигналами PIDP блока.

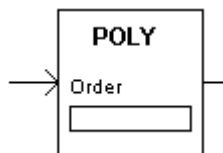
Полученное значение нового управляющего сигнала $u(t)$ ограничено интервалом (U_{min}, U_{max}) . Параметры *UMin* и *UMax* задаются в соответствующих единицах измерения. Значения по умолчанию 0 и 100 (%) соответственно.

Рабочий режим

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

- *Mode* = 0 => Выключен, регулятор остановлен
- *Mode* = 1 => Нормальный режим управления.
- *Mode* = 2 => Значение на выходе регулятора равно *UMax*.
- *Mode* = 3 => Значение на выходе регулятора равно *UMin*.

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

V.37 POLY - Полиномы

ВХОДЫ	Переменная (variable)	ДРОБНЫЙ	
ПАРАМЕТРЫ	список коэффициентов (list of coefficients)	ДРОБНЫЙ	Список полиномиальных коэффициентов a_0, \dots, a_n
ТИП ВЫХОДА		ДОСТУП	
ДРОБНЫЙ		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 полинома (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: Froberg, 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 $p(x) = b_n$:

$$b_0 = a_n$$

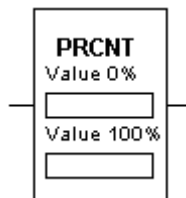
$$b_1 = b_0 \cdot x + a_{n-1}$$

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

....

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

V.38 PRCNT - Процентное соотношение



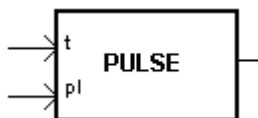
ВХОДЫ	вход (variable)	ДРОБНЫЙ	входной сигнал
ПАРАМЕТРЫ	значение 0% (value 0%)	ДРОБНЫЙ	Значение сигнала на входе, который составляет 0% на выходе
	значение 100% (value 100%)	ДРОБНЫЙ	Значение сигнала на входе, который составляет 100% на выходе
ТИП ВЫХОДА		ДОСТУП	
ДРОБНЫЙ		RO	

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.

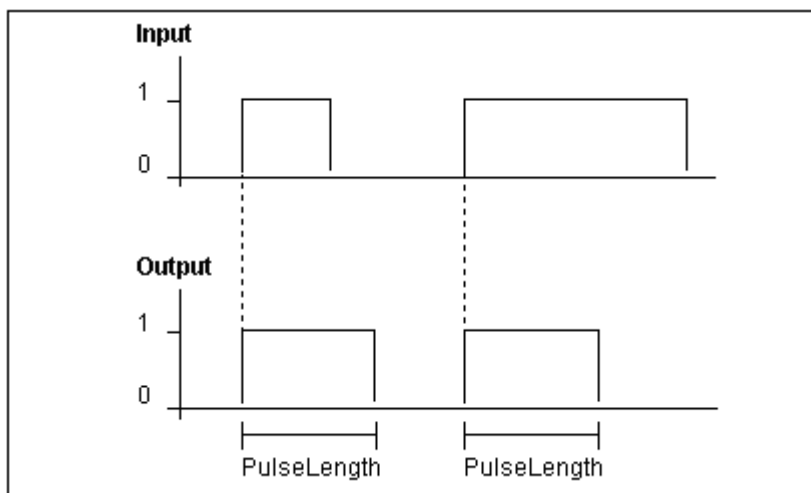
B.39 PULSE - Импульсный генератор



ВХОДЫ	Триггер (t)	БИНАРНЫЙ	Входной сигнал
	Длина импульса (pl)	ДРОБНЫЙ	Продолжительность выходного импульса (сек)
ТИП ВЫХОДА		ДОСТУП	
БИНАРНЫЙ		RO	

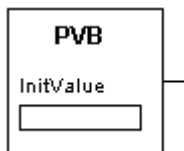
Описание

Функциональный блок с моностабильным бинарным выходом с изменяемой длительностью импульса (импульсный генератор). Длина импульса задается значением входящего сигнала *Длина импульса*. Данный блок имеет только одно стабильное состояние выхода (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. Импульс на выходе появляется только по переднему фронту входного сигнала, при неизменном входном сигнале импульса на выходе не будет. Таким образом длина выходящего импульса не зависит от длины входного импульсного сигнала.



Длина 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.

V.40 PVB - Бинарная переменная

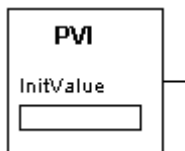


ПАРАМЕТРЫ	Нач.значение (InitValue)	БИНАРНЫЙ	Начальное значение на выходе
ТИП ВЫХОДА		ДОПУСК	
БИНАРНЫЙ		R/W	

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 общедоступный (Public). Начальное значение на выходе задается параметром *Начальное значение (Initial Value)*.

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

B.41 PVI - Целая переменная



ПАРАМЕТРЫ	Нач.значение (InitValue)	ЦЕЛЫЙ	Начальное значение на выходе.
ТИП ВЫХОДА		ДОСТУП	
ЦЕЛЫЙ		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 общедоступный. Начальное значение на выходе задается параметром Начальное значение (Initial Value).

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

B.41.1 PVI - XENTASYSREG

Особый тип целой переменной PVI, в котором доступны две новые функции:

- Отображение дерева меню без сброса через 30 минут – обычно дерево меню на экране панели оператора сбрасывается для отображения дерева меню всех установок через 30 минут. При активации этой функции экран панели оператора не будет сбрасываться для отображения списка всех установок через 30 минут.
- Функция перехода на летнее время для Австралии — данная функция приводит переход на летнее время в соответствии с условиями перехода в Австралии, которые противоположны условиям перехода в Европе.



Обратите внимание!

Чтобы данную функцию можно было использовать требуется версия 3.5 (или выше) системной программы и программы для панели оператора.

Чтобы использовать одну или обе из этих функций блоку PVI следует задать имя "XENTASYSREG" и одно из следующих значений должно быть введено для "XENTASYSREG" блока PVI:

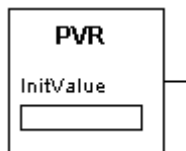
Value = 0. *Ни одна из функций не активна.*

Value = 1. Активирована функция отображения дерева меню без сброса через 30 минут.

Value = 2. Активирована функция перехода на летнее время для Австралии.

Value = 3. Обе функции активированы.

B.42 PVR - Дробная переменная



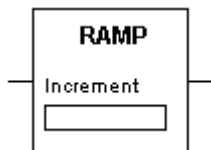
ПАРАМЕТРЫ	Нач.значение (InitValue)	ДРОБНЫЙ	Начальное значение на выходе.
ТИП ВЫХОДА		ДОСТУП	
ДРОБНЫЙ		R/W	

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 общедоступный (Public). Начальное значение на выходе задается параметром *Начальное значение (Initial Value)*.

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

V.43 RAMP - Ограничитель скорости изменения сигнала



ВХОДЫ	Вход (variable)	ДРОБНЫЙ	Входной сигнал
ПАРАМЕТРЫ	Увеличение (Increment)	ДРОБНЫЙ	Максимальное увеличение скорости сигнала в секунду.
ТИП ВЫХОДА		ДОСТУП	
ДРОБНЫЙ		R/W	

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

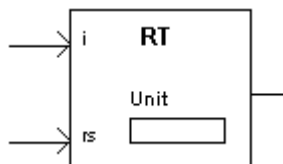
B.44 RST - Рестарт



ТИП ВЫХОДА	ДОСТУП	
БИНАРНЫЙ	RO	

Описание Выход активизируется в первом программном цикле после теплого старта.

B.45 RT - Учет времени работы



ВХОДЫ	Отсчет (i)	БИНАРНЫЙ	Вход для запуска отсчета
	Сброс (rs)	БИНАРНЫЙ	Вход для сброса суммы (1 = сброс)
ПАРАМЕТРЫ	Ед.измерения (Unit)	ЦЕЛЫЙ	Ед.измерения времени на выходе (0 = часы, 1 = минуты, 2 = секунды). Значение по умолчанию = 0 (часы)

ТИП ВЫХОДА	ДОСТУП	
ЦЕЛЫЙ	R/W	

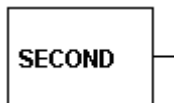
Описание

Данный блок используется для того, чтобы суммировать время, в течение которого активизирован бинарный сигнал *Отсчет* (= 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 (ЛОЖЬ). 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.

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

B.46 SECOND - Текущая секунда



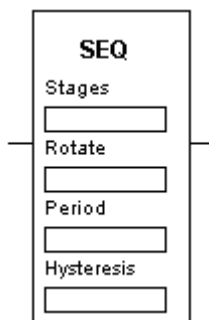
ТИП ВЫХОДА	ДОСТУП	
ЦЕЛЫЙ	RO	

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



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

B.47 SEQ - Последовательное управление



ВХОДЫ	Вход (Input)	ДРОБНЫЙ	Входной сигнал (%)
ПАРАМЕТРЫ	Ступень (Stages)	ДРОБНЫЙ	Количество ступеней гистерезиса на выходе (1 до 16).
	Поворот (Rotation)	БИНАРНЫЙ	Определяет наличие поворота ступеней гистерезиса.
	Период (Period)	ЦЕЛЫЙ	Задержка одновременной активизации ступеней гистерезиса, в миллисекундах.
	Гистерезис (Hysteresis)	ДРОБНЫЙ	Ширина гистерезиса (%).
ТИП ВЫХОДА		ДОСТУП	
ЦЕЛЫЙ		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:

$$N=(M+1)*Input/100$$

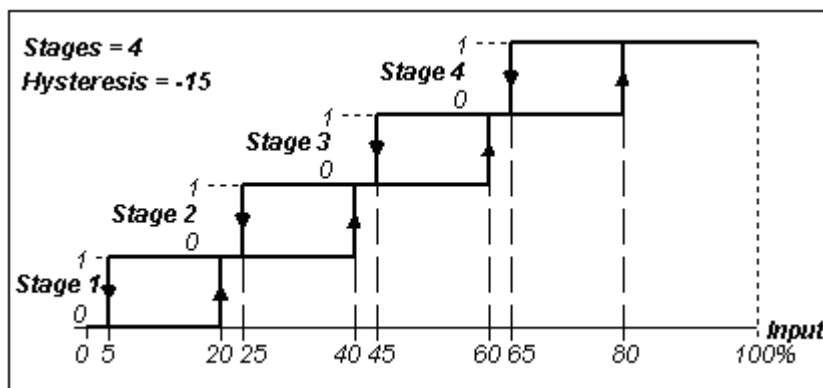
где *Вход (Input)*- значение между 0% и 100%. Ступени M представляют собой первые M-биты от целого значения выходного сигнала блока. Первая величина N из данных битов будет равна 1 (ИСТИНА), а все остальные - 0 (ЛОЖЬ).

Пример: Если параметр *Ступени (M)* задан равным 4, а *Гистерезис (Hysteresis)* и *Поворот (Rotate)* 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 TAC 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 входным сигналом. Если *Гистерезис* имеет ненулевое значение, петля гистерезиса будет находиться 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%, на 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. активация 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.

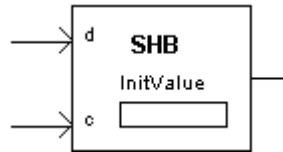


Параметр *Период (Period)* используется для предотвращения одновременной активации двух или более ступеней. Если значение периода не равно нулю, ступени будут всегда активизироваться последовательно с интервалом, соответствующим значению *Периода* (миллисекунды), даже при условии значительного изменения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

V.48 SHB - Регистр, сдвигающий вправо, бинарный сигнал



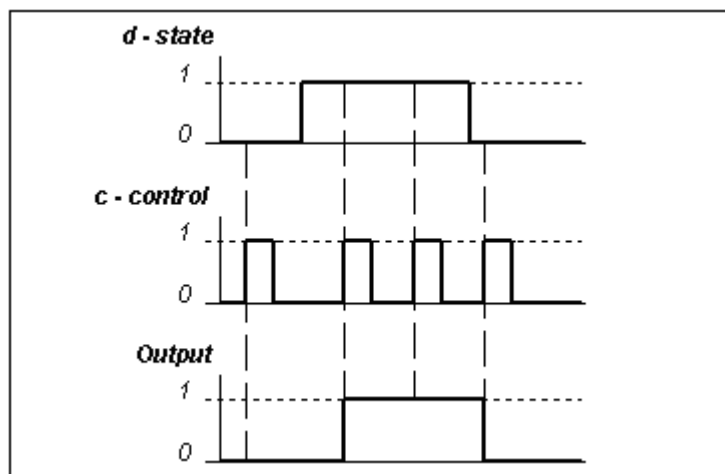
ВХОДЫ	Состояние (d)	БИНАРНЫЙ	Входной бинарный сигнал.
	Управление (c)	БИНАРНЫЙ	Управляющий сигнал.
ПАРАМЕТРЫ	Нач.значение (InitValue)	БИНАРНЫЙ	Начальное значение выходного сигнала.
ТИП ВЫХОДА		ДОСТУП	
БИНАРНЫЙ		R/W	

Описание

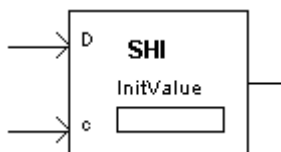
Функция регистра, сдвигающего вправо, выполняется согласно нижеприведенной таблице истинности:

состояние (t)	управление (t)	выход (t+1)
0	0	output (t)
1	0	output (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.



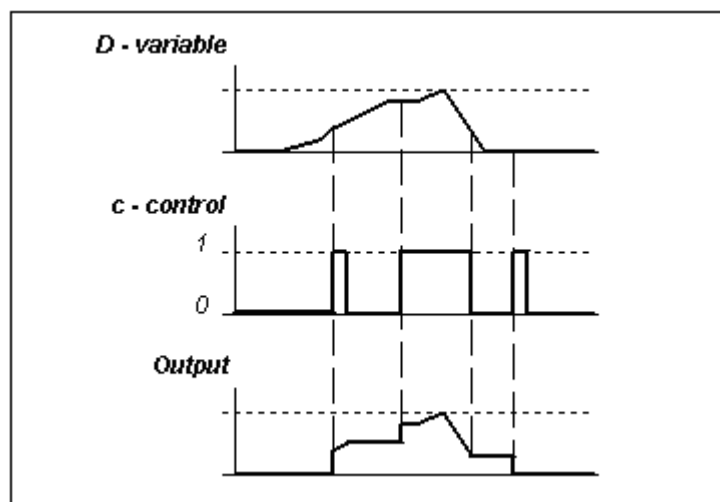
V.49 SHI - Регистр, сдвигающий вправо, целый сигнал



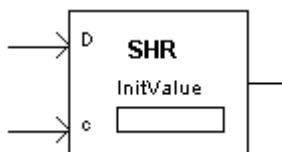
ВХОДЫ	переменная(D)	ЦЕЛЫЙ	Аналоговый входной сигнал.
	управление (c)	БИНАРНЫЙ	Управляющий сигнал.
ПАРАМЕТРЫ	Нач.значение (InitValue)	ЦЕЛЫЙ	Начальное значение на выходе.
ТИП ВЫХОДА		ДОСТУП	
ЦЕЛЫЙ		R/W	

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.

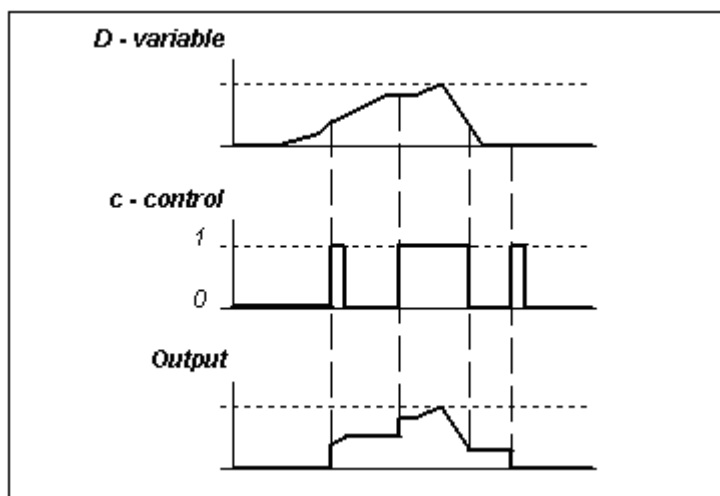


V.50 SHR - Регистр, сдвигающий вправо, дробный сигнал

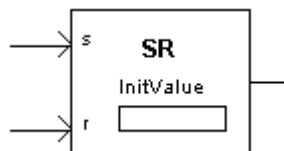


ВХОДЫ	Переменная (D)	ДРОБНЫЙ	Аналоговый входной сигнал.
	Управление (c)	БИНАРНЫЙ	Управляющий сигнал.
ПАРАМЕТРЫ	Нач.значение (InitValue)	ДРОБНЫЙ	Начальное значение на выходе.
ТИП ВЫХОДА		ДОСТУП	
ДРОБНЫЙ		R/W	

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 входного сигнала из предыдущего цикла.



B.51 SR - PC Триггер



ВХОДЫ	Уставка (s)	БИНАРНЫЙ	Активация выходного сигнала.
	Сброс (r)	БИНАРНЫЙ	Деактивация выходного сигнала.
ПАРАМЕТРЫ	Нач.значение (InitValue)	БИНАРНЫЙ	Начальное значение на выходе.

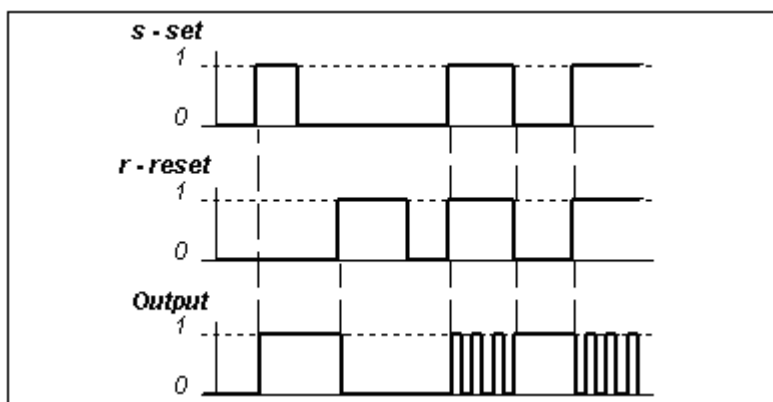
ТИП ВЫХОДА	ДОСТУП	
БИНАРНЫЙ	R/W	

Описание

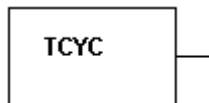
PC Триггер представляет собой бистабильный блок с двумя входами: уставка и сброс. В зависимости от значения сигнала на данных входах на заданный момент времени, для следующего программного цикла выход блока получит значение в соответствии с данными, приведенными в таблице:

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.



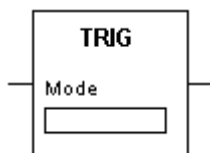
V.52 ТСУС - Время цикла



ТИП ВЫХОДА	ДОСТУП	
ДРОБНЫЙ	RO	

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

B.53 TRIG - Триггер



ВХОД	Состояние (State)	БИНАРНЫЙ	Сигнал триггера.
ПАРАМЕТРЫ	Режим (Mode)	ЦЕЛЫЙ	Режим работы 1,2 или 3
ТИП ВЫХОДА		ДОСТУП	
БИНАРНЫЙ		RO	

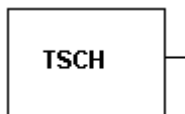
Описание

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 выдавать импульс на выходе при следующих условиях:

- **Режим 0** и **1**: при изменении от низкого к высокому (по переднему фронту).
- **Режим 2**: при изменении от высокого к низкому (по заднему фронту входного сигнала).
- **Режим 3**: при изменении любого типа (по обоим фронтам).

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.

B.54 TSCH - Расписание



ПАРАМЕТРЫ	Недельный график, Мах. (Week charts, Max.)	ЦЕЛЫЙ	Число недельных графиков.
	Годовой график, Мах. (Holiday charts, Max.)	ЦЕЛЫЙ	Число годовых графиков
ТИП ВЫХОДА		ДОСТУП	
ЦЕЛЫЙ		RO	



Обратите внимание!

Прикладная программа для устройства TAC Xenta 280 может содержать только один TSCH блок.

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!

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 TAC Xenta OP.

Дополнительные недельные графики могут выбираться при помощи панели оператора непосредственно во время работы.

Количество графиков выходных задается при помощи параметра *Годовой график*, 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 TAC Xenta OP. Дополнительные графики выходных могут выбираться при помощи панели оператора непосредственно во время работы.

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

Графики выходных можно задавать в TAC Vista, путем использования функции центрального расписания времени.

Пример

Предположим, что АНУ следует работать в следующем режиме:

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 сочельник АНУ должна работать с 15:00 до 16:00, если сочельник в понедельник, вторник, среду, четверг или пятницу.

Также должна быть возможность задания двух праздничных графиков с панели оператора TAC Xenta OP.

Подобное расписание будет выглядеть следующим образом:

Недельный график, Мах. = 4:

Время старта	Время останова	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, Мах. = 3:

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



Обратите внимание!

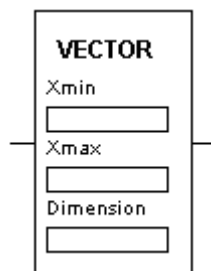
Примечание 1: Символ "звездочка" в колонках "дата старта" и "дата останова" обозначает возможность задавать определенный выходной день каждый год. Такое возможно только для графиков выходных.

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.

B.55 VECTOR - Векторная функция



ВХОДЫ	Вход(Input)	ДРОБНЫЙ	Входной сигнал.
ПАРАМЕТРЫ	Xmin	ДРОБНЫЙ	Нижняя граница диапазона входного сигнала.
	Xmax	ДРОБНЫЙ	Верхняя граница диапазона входящего сигнала.
	Размеры (Y(X)) (Dimension (Y(X)))	ДРОБНЫЙ	Список значений линейной функции (от 2 до 255). Каждое значение в списке должно вводиться отдельной строкой.
ТИП ВЫХОДА		ДОСТУП	
ДРОБНЫЙ		RO	

Описание

Блок VECTOR позволяет определить любой отрезок линейной функции для входящего сигнала, ограниченный параметрами Xmax и Xmin. 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 Xmin, функция определяется значением, заданным для первой точки интервала. Если входящий сигнал больше Xmax, функция будет определяться значением, заданным для последней точки интервала.

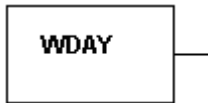
Обратите внимание!

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: $8 - 5 + 1 = 4$).

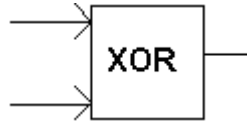
B.56 WDAY - Текущий день недели



ТИП ВЫХОДА	ДОСТУП	
ЦЕЛЫЙ	RO	

Описание Данный блок показывает текущий день недели, согласно внутренним часам. Значение 1 на входе соответствует понедельнику, 7 - воскресенью.

V.57 XOR - Инверсия или XOR-функция



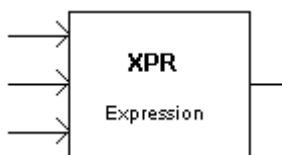
ВХОДЫ	состояние1	БИНАРНЫЙ	
	состояние2	БИНАРНЫЙ	
ТИП ВЫХОДА		ДОСТУП	
БИНАРНЫЙ		RO	

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:

состояние 1	состояние 2	ВЫХОД
0	0	0
0	1	1
1	0	1
1	1	0

С **Формулы (Expressions)**



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>Комментар.</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 (объекты действия):

Входные переменные

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

Числовые константы

Целая часть константы может вводиться со знаками + или -. За целой частью может следовать запятая (,) или 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 константы

Константы данного типа представляют собой буквенно-цифровые идентификаторы не более 20 значимых символов, заданных в таблице констант. Имя константы должно быть заключено в кавычки ("), если в нем присутствуют знаки / или :. Общедоступные константы 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)
Троичные операторы		логическая "ИЛИ"-функция (побитовая)
	? :	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)	десятичный логарифм (по основанию 10)
EXP (x)	Экспонента e в степени x
COS (x)	Косинус x (радиан), для всех x.
SIN (x)	Синус x (радиан), для всех x.
TAN (x)	Тангс x (radians), для всех x.
ACOS (x)	Арккосинус x, где $-1 < x < 1$, результат между 0 и $\pi/2$ радиан.
ASIN (x)	Арсинус x, где $-1 < x < 1$, результат между $-\pi/2$ и $\pi/2$ радиан.
ATAN (x)	Арктангенс x, для всех x, результат между $-\pi/2$ и $\pi/2$ радиан.
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 $\%$, \ll , \gg , $\&$, \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 $>$, $<$, $>=$, $<=$, $=$, \neq преобразуют операнды в Дробные значения, выполняют сравнение и возвращают Бинарный результат.

Выход

Блок формул может иметь следующие типы выходного сигнала

XPB Block

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 Block

Результат преобразован в 16-битное **Целое** число..

XPR Block

Сигнал на выходе имеет **Дробное** значение, полученное в результате вычислений по формуле. .

D Операторы (Operators)

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**

<i>Название</i>	<i>Комментарии</i>
Бинарная константа (Binary const)	Нет входа → Бинарный выход
Целая константа (Integer const)	Нет входа → Целый выход
Дробная константа (Real const)	Нет входа → Дробный выход

Общедоступные константы не используются в качестве операторов.

- **Логические операторы**

<i>Название</i>	<i>Комментарии</i>
NOT	Бинарный вход → Бинар. выход
AND	Бинарный входы → Бинар. выход
OR	Бинарный входы → Бинар. выход
XOR	Бинарный входы → Бинар. выход

• Математические операторы

<i>Название</i>	<i>Комментарии</i>
Отрицание (Negate)	Аналог. вход → Аналог. выход
Сложение (Addition)	Аналог. входы → Аналог. выход
Вычитание (Subtraction)	Аналог. входы → Аналог. выход
Умножение (Product)	Аналог. входы → Аналог. выход
Деление (Division)	Аналог. входы → Аналог. выход
Модуль (Module)	Аналог. входы → Аналог. выход

• Сопоставление

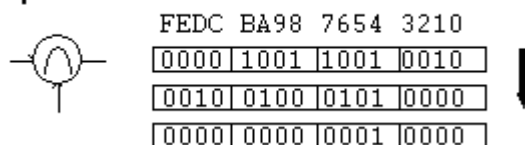
<i>Название</i>	<i>Комментарий</i>
Меньше чем (Less than)	Аналог. входы → Бинар. выход
Больше чем (Greater than)	Аналог. входы → Бинар. выход
Равенство (Equal)	Аналог. входы → Бинар. выход
Неравенство (Not equal)	Аналог. входы → Бинар. выход
Больше или равно (Greater or equal)	Аналог. входы → Бинар. выход
Меньше или равно (Less or equal)	Аналог. входы → Бинар. выход

• **Битовая операция**

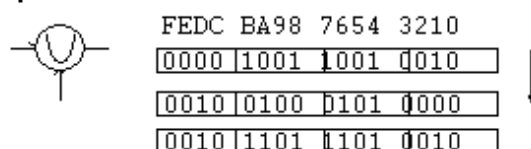
Название	Комментарии
bit AND	Аналог. входы → Аналог. выход
bit OR	Аналог. входы → Аналог. выход
bit XOR	Аналог. входы → Аналог. выход
Сдвиг вправо (Shift right)	Аналог. входы → Аналог. выход
Сдвиг влево (Shift left)	Аналог. входы → Аналог. выход

Примеры

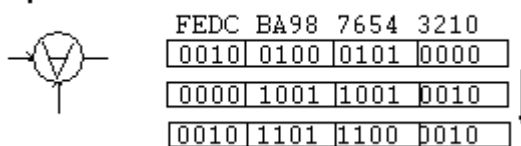
Operator bit AND



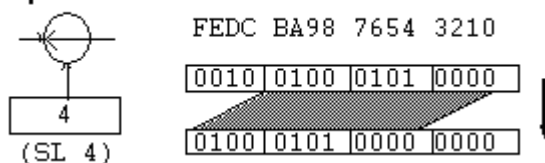
Operator bit OR



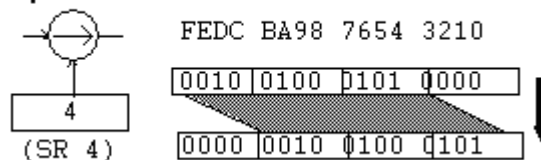
Operator bit XOR



Operator <<



Operator >>



Обратите внимание!

Сдвигаются все 16 позиций, 0 сдвигается в пустые позиции. Любое число сдвигов более 16 обуславливает значение на выходе, равное 0.

• **Другие**

<i>Название</i>	<i>Комментарии</i>
D/A преобразователь (D/A converter)	Бинарный вход → Аналог.выход
A/D преобразователь (A/D converter)	Аналог.вход → Бинар.выход
Аналоговый мультиплексор (Analog multiplexer)	Аналог.входы → Аналог.выход
Бинарный мультиплексор (Binary multiplexer)	Бинар.входы → Бинар.выход
Преобразование AA (Conversion 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 (ясно, недвусмысленно). 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 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 входных сигналов в зависимости от значения Бинарного переключателя.

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

E Тестовые блоки (Test Probe Blocks)

E.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 (IO), поскольку выполняет тестовые функции, аналогичные функциям блока IO.



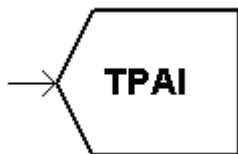
Обратите внимание!

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
ТРАИ	Тестовый блок для Аналогового входа
ТРДИ	Тестовый блок для Цифрового входа

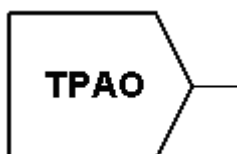
E.2 ТРАІ - Тестовый блок для Аналог. входа



ВХОДЫ	Вход	ДРОБНЫЙ	Входной сигнал
ТИП ВЫХОДА		ДОСТУП	
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.

E.3 ТРАО - Тестовый блок для Аналог. выхода



ВХОДЫ			Блок не имеет входов.
ТИП ВЫХОДА		ДОСТУП	
ДРОБНЫЙ		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".

E.4 TPDI - Тестовый блок для Цифрового входа



ВХОДЫ	Вход	БИНАРНЫЙ	Входной сигнал
ТИП ВЫХОДА		ДОСТУП	
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.

E.5 TPDO - Тестовый блок для Цифрового выхода



ВХОДЫ			Блок не имеет входов.
ТИП ВЫХОДА		ДОСТУП	
БИНАРНЫЙ		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.

F Рекомендации по программированию

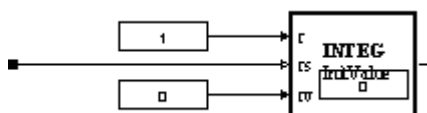
Время программного цикла

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.



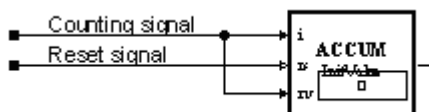
Счетчики времени могут быть созданы посредством блоков PULSE и RT. Только построения из блоков RT и INTEG могут сбрасывать значения при подсчете. Счетчик INTEG используется, в основном, при подсчете (времени, мощности и т.д.) за определенный период.

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:



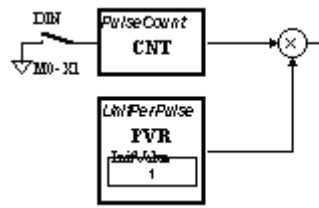
Побитовые логические операторы

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.

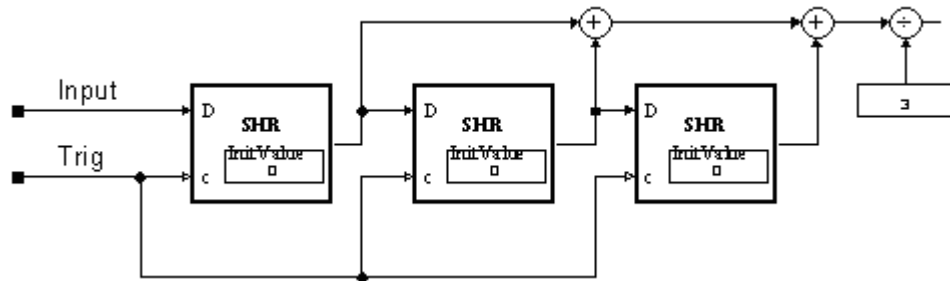
Параметр Множителя в блоке CNT

Внутренний параметр множителя в блоке не относится к общедоступным константам, 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. .



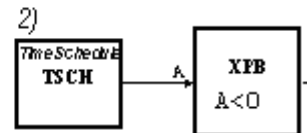
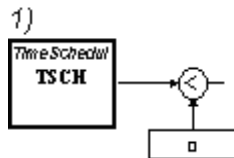
Скользящее среднее значение

Пример программы со скользящим средним значением трех величин:



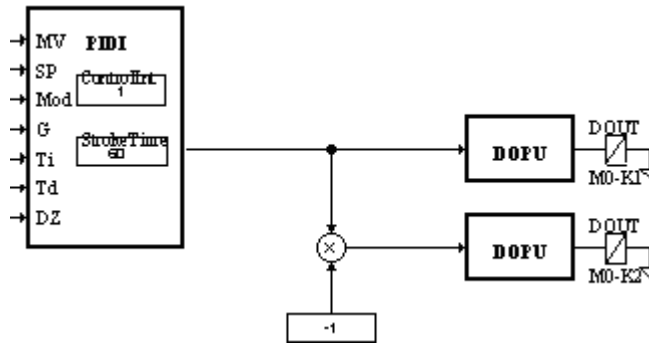
Выход TSCH

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



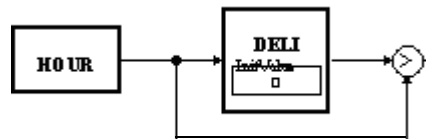
PIDI - DOPU

Стандартные функции блоков PIDI и DOPU;



Смена суток

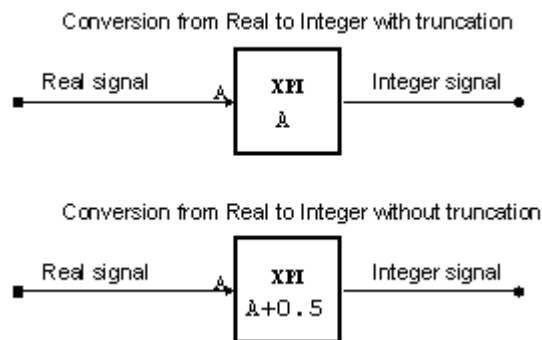
Смена суток произошла, если значение нового часа меньше, чем полученное за предыдущий час.



Блоки формул

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

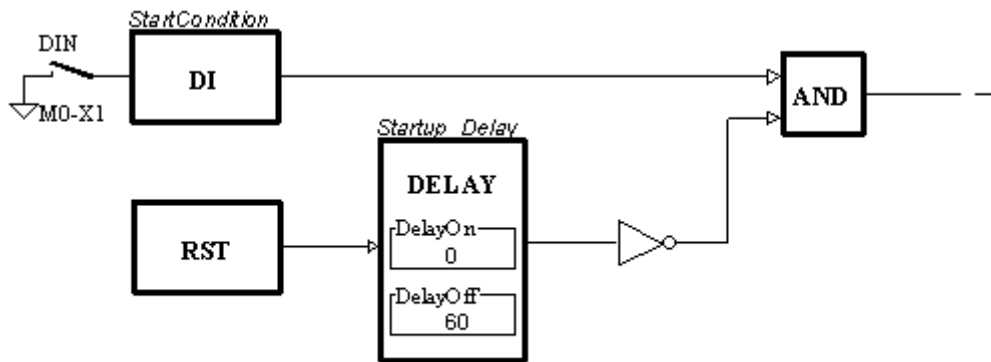
Блоки формул могут использоваться для преобразования Дробного сигнала в Целый и наоборот:



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):



G Поддерживаемые SNVT

Список SNVT, поддерживаемых TAC Menta регулярно обновляется, за последней информацией обращайтесь в службу технической поддержки TAC Menta по электронной почте info@tac-russia.ru.

G.1 Список используемых SNVT

TAC Menta поддерживает следующие типы SNVT:

SNVT	Number
SNVT_alarm	88
SNVT_amp	1
SNVT_amp_f	48
SNVT_amp_mil	2
SNVT_angle	3
SNVT_angle_deg	104
SNVT_angle_vel	4
SNVT_btu_f	67
SNVT_char_ascii	7
SNVT_chlr_status	127
SNVT_config_src	69
SNVT_count	8
SNVT_count_f	51
SNVT_count_inc	9
SNVT_count_inc_f	52
SNVT_date_day	11
SNVT_elapsed_tm	87
SNVT_elec_kwh	13

SNVT	Number
SNVT_elec_whr	68
SNVT_flow	15
SNVT_flow_f	53
SNVT_flow_p	161
SNVT_freq_f	75
SNVT_freq_hz	76
SNVT_freq_kilohz	77
SNVT_hvac_emerg	103
SNVT_hvac_mode	108
SNVT_hvac_overid	111
SNVT_hvac_status	112
SNVT_length_mil	20
SNVT_lev_count	21
SNVT_lev_count_f	55
SNVT_lev_disc	22
SNVT_lev_percent	81
SNVT_lux	79
SNVT_magcard	86
SNVT_mass_f	56
SNVT_muldiv	91
SNVT_multiplier	82
SNVT_obj_request	92
SNVT_obj_status	93
SNVT_occupancy	109
SNVT_power	27
SNVT_power_f	57
SNVT_power_kilo	28
SNVT_ppm	29
SNVT_ppm_f	58
SNVT_preset	94
SNVT_press	30
SNVT_press_f	59
SNVT_press_p	113

SNVT	Number
SNVT_pwr_fact	98
SNVT_pwr_fact_f	99
SNVT_regval	136
SNVT_regval_ts	137
SNVT_res_f	60
SNVT_rpm	102
SNVT_scene	115
SNVT_scene_cfg	116
SNVT_setting	117
SNVT_speed	34
SNVT_speed_f	62
SNVT_state	83
SNVT_str_asc	36
SNVT_str_int	37
SNVT_switch	95
SNVT_temp	39
SNVT_temp_f	63
SNVT_temp_p	105
SNVT_temp_setpt	106
SNVT_time_f	64
SNVT_time_hour	124
SNVT_time_min	123
SNVT_time_sec	107
SNVT_time_stamp	84
SNVT_tod_event	128
SNVT_vol	41
SNVT_vol_f	65
SNVT_vol_kilo	42
SNVT_volt	44
SNVT_volt_f	66
SNVT_zero_span	85

Алфавитный указатель

A

ACCUM - Сумматор 176
ANYST - Аналоговый гистерезис 177
AI - Аналоговый вход 179
ALARM - Авария 189
AND - Логическая функция "И" 192
АО - Аналоговый выход 193

B

BPS 22

C

CNT 32
CNT - Цифровой вход - Счетчик
импульсов 196
Com пауза 22
Com Порт 21
Ctrl-Insert 45
CURVE - Линейный график 198

D

DATE - Дата 200
DELAY - Задержка включения/
выключения 201
DELB - Задержка на 1 цикл (бинарный) 203
DELI - Задержка на 1 цикл (целый) 204
DELR - Задержка на 1 цикл (дробный) 205
DI - Цифровой вход 206
DO - Цифровой выход 212
DOPU 32
DOPU - Цифровой импульсный выход 215

E

ENTH - Расчет энтальпии 217
ERR - Системный ошибки 221

F

FBD 19, 23, 44
FILT - Фильтр первого порядка 223

H

HFB 38, 85
HOUR - Текущий час 224
HYST - Бинарный гистерезис 225

I

ID прикладной программы 152
INTEG - Интегратор 227

L

LIMIT - Ограничение по min/max 228
LonWorks 19

M

MAX - Селектор максимального из 2
сигналов 229
MIN - Селектор минимального из 2
сигналов 230
MINUTE - Текущая минута 231
MONTH - Текущий месяц 232

N

NCYC - Счетчик программного цикла 233
NOT - Инверсия бинарного сигнала 234

O

OP 19
OPT - Оптимизатор 235

OR - логическая функция "ИЛИ" 244
OSC - Осциллятор 245

P

PIDA - PID регулятор 246
PID1 - DOPU 301
PID1 - PID регулятор - Увеличить/
Уменьшить 250
PIDP - PIDP регулятор 253
POLY - Полиномы 257
PRCNT - Процентное соотношение 259
PULSE - Импульсный генератор 260
PVB - Бинарная переменная 261
PVI - Целая переменная 262
PVR - Дробная переменная 264

R

RAMP - Ограничитель скорости изменения
сигнала 265
RST 45
RST - Рестарт 266
RT - Учет времени работы 267

S

SECOND - Текущая секунда 268
SEQ - Последовательное управление 269
SHB - Регистр, сдвигающий вправо
бинарный сигнал 272
Shift-Insert 45
SHR - Регистр, сдвигающий вправо
дробный сигнал 274
SNVT 19, 35
SR - PC-Триггер 275

T

TAC Menta 19
TAC Menta v3 совместимость 156
TAC Xenta 19
обновление 157

ТСУС - Время цикла 276
text file 22
TРАI - аналоговый вход 29
TРАI - Тестовый блок для Аналогового
входа 295
TРАO - аналоговый выход 29
TРАO - Тестовый блок для Аналогового
выхода 296
TрD1 - Тестовый блок для Цифрового
входа 296
TрD1 - цифровой вход 29
TрDО - Тестовый блок для Цифрового
выхода 297
TрDО - цифровой выход 29
TRIG - Триггер 277
TSCH 39
TSCH - Расписание 278

V

VECTOR - Векторная функция 281

W

WDAY - Текущий день недели 283

X

XOR - Инверсия или XOR-функция 284

A

Автоматическая генерация
Вектор входных значений 108
Аналоговые колебания
Амплитуда 108
Период 109
Среднее значение 109
Фаза 109
Циклические 108
Форма колебаний 108
аналоговый 24

арифметическое выражение 27

Б

библиотека 22

Бинарные последовательности

Период 109

Последовательность 109

Цикличность 109

Бинарные последовательности, генерация
109

БЛОК 59

Раскрыть HFB 59

Рассоединить 59

Дублировать 59

Правка 59

Удалить 59

блок 66

блоки источники сигналов 23

блоки получатели 23

Блоки формул 301

Бит в секунду 22

В

Временные выборки

Задание 111

Время программного цикла 299

Вектор входных значений

Автоматическая генерация 108

внутренние константы 30

Всплывающее меню

ЗАПИСЬ 105

ТРЕНД 105

УПРАВЛЕНИЕ 105

Всплывающие меню

СИГНАЛ 104

ВХОД 61

Разъединить 61

Отсоединить 61

входные сигналы 23

входы 27

Выполнение 44

ВЫХОД 62

Разъединить 62

Отсоединить 62

Выход TSCH 300

выходной сигнал 27

Г

ГРУППА 63

Разъединить 63

Копировать 63

Модуль 63

Создание HFB 63

Сохранить 63

Снять выделение 63

Удалить 63

Генерация

Бинарные последовательности 109

Аналоговые колебания 108

Генерация исполняемого кода 143

генерация прикладной программы 29

Горячий старт 19

годовой таймер 39

Д

Другие функции и инструменты 137

Дерево меню панели оператора

Автоматическая генерация 125

ДЕРЕВО

Раскрыть 135

Генерировать 135

Загрузить спецификацию 135

Копировать в буфер 136

Просмотр спецификации 136

Обновление спецификации 135

Построить 135

Удалить спецификацию 136

Тест 135

Данные приложения

Загрузка из... 149

Дата и Время 94

Документация прикладной программы

Распечатка 137

З

Загрузка данных приложения из контроллера 149

Загрузка прикладной программы 147

Задержка включения 301

ЗАПИСЬ

Изменить 105

Изменить диапазон 105

Останов по пределу 105

Удалить 105

Запуск TAC Menta 22

И

Иерархические функциональные блоки 38, 85

Раскрытие и свертывание блока HFB 86

Вывод на печать HFB 86

Название HFB и соединений 85
создание HFB 85

Изменение параметров блока 115

Временные расписания 115

Данные связывания I/O 115

Константы 115

Параметры функционального блока 115

Изменение параметров связывания 151

Изменение общедоступных констант 151

Имитация 43, 45

Внешние входы 107

Всплывающие меню 104

Изменение параметров блока 115

Команды 103

Инструменты 104

Исполняемые файлы 107

Меню Опции 98

Масштаб 103

Помощь 104

Опции 101

Свойства 99

Файл 99

Тестовые блоки 109

Инструменты 55

Регистратор 104

Конфигурация панели оператора 55

Панель инструментов 55

Обновить 55

Обновить окно 104

Текстовый файл 55, 104

Исполняемый код

Генерация 143

BIN 143

CHR 143

COD 143

ESP 143

OPC 143

XIF 143

Исполняемый файлы, имитация 107

Использование памяти

Рабочая область 144

Вычисление 144

Вычислитель размера BPR 145

Количество объектов 145

Приложение 144

Параметры 144

TACN/SNVT вх/вых 145

К

Корректность выполнения

Проверка 158

логический оператор 28

Команды

Генерировать 103

Горячий старт 103

Вполнение 103

Выборка 103

Выполнение n раз 103

Загрузка в 103

Загрузка из 103

Сброс 103

Шаг 103

Холодный старт 103
Константы 29
константы 23
Конфигурация панели оператора
 Дерево 135
 ПОМОЩЬ 136
 Список сигналов 118
 Строка меню 131
 Форматы 133
Конфигурация устройства 139
 Добавление новой сетевой перменной
 (SNVT) последней в файл XIF
 140
 Аппаратная версия 139
 Использование LonMark 3.3 140
 Системная версия 139
 Тип бавого устройства 139
 XIF заголовок, генерируемый в
 соответствии со стандартом
 LonMark 140
контур регулирования 23

М

Маркер
 Сохранить как 99
макро блоки 22
Меню опции 131
Мастер загрузки 153
 Диалог 154
 Общая процедура загрузки 155
Масштаб 54
 Нормальный 54
 нормальный 103
 Увеличение 54, 103
 Уменьшение 54, 103
Модули 37
модуль 24
Множественная привязка 22
Мышь 43

Н

недельный график 39
На-линии 45
НОВЫЙ 58, 65
 Вставка 58
 Загрузка группы 58
 Линия 58
 Комментарий 58
 HFB I/O 58
 Простой блок 58
 Прямоугольник 58
 Оператор 58
 Сжать HFB 58
 Тестовый блок 58
 Функция 58

О

обратная связь 23
Обозначение стандартных приложений/
контроллеров 151
Обновление
 AUTфайл 157
 Системные и прикладные программы
 157
Обновление TAC Xenta 300 до v3 157
Общедоступные константы 28
 Изменение в режиме На-линии 151
общедоступные константы 28, 30, 36
Общедоступные параметры 36
Определение файла надора символов 130
Операци над группами
 Ввод или правка названия модуля 80
Операции над группами
 Разъединение 78
 Выделение 77
 Загрузка макроблока 82
 Копирование и вставка 80
 Копия выделенной области в буфер 80
 Перемещение 78
 Печать выборки 80
 Сохранение макроблока 81

- Снятие выделения 78
- Удаление 80
- Центрирование выборки 80
- Операции с группами 77
- Операции с сигналами 106
- Опции 53
 - Ручной режим 107
 - Имитация 53
 - Конфигурация устройства 53, 101
 - Использование памяти 53, 102
 - Правка 101
 - Программная спецификация 53, 101
 - Не обновлять константы 101
 - На-линии 101
 - Системная информация 102
 - Тренд Логи 53, 102
 - Таблица расписаний 53, 101
 - Таблица аварийных сообщений 53, 101
 - Таблица констант 101
 - таблица констант 53
 - Таблица конфигурации I/O 53
 - таблица конфигурации I/O 101
 - Установка даты и времени 53, 101
- Опции меню 47
- основные I/O блоки 32
- Основные положения 23
- отрезки 25
- Отмена 95
- Отсоединить блок 69

- П**
- Правка 51
 - Добавить блок перехода 51
 - Вставить 51
 - Выбрать все 51
 - Замена 51
 - Копировать 51
 - Копировать в буфер 51
 - Найти 51
 - Откат 51
 - Снять выделение всех соединений 51
 - Удалить 51
 - удалить блоки перехода 51
 - Центрировать выборку 51
- Правка - Найти 44
- Правка-Переместить 44
- Правка-Центрировать выборку 44
- Правила доступа 39
- Прикладная программа
 - Загрузка 147
- Прикладная программа, Выполнение 105
- Принудительное переопределение сигналов физических входов/выходов
 - Переопределение 150
- Программная спецификация 141
 - Дата 142
 - Автор 142
 - Блоки 142
 - Имя 142
 - I/O сигналы 142
 - Период цикла 142
 - Стандартное прил. 142
 - Таблица общедоступных сигналов 142
 - Тип 142
- Программные лицензии 21
- программный цикл 23
- Проверка корректности выполнения 158
- пропадание питания 19
- промежуточные блоки 23
- Просмотр
 - Тренд Лог 111
- Простые блоки 27
- Параметр Множителя в блоке CNT 300
- параметры 23
- Параметры блока
 - Изменение 115
- Параметры блоков
 - Изменение в режиме На-Линии 151
- Параметры связывания
 - Изменение в режиме На-Линии 151
- Перемещение узла 75
- По шагам 44
- Побитовые логические операторы 299

- Повторная отправка 22
ПОМОЩЬ
 О программе 136
 Содержание 136
Помощь 56
 О программе 56, 104
 Содержание 56, 104
Пункты меню 47
- Р**
- рассчитанная величина 24
Редактирование 43, 45
Редактирование блока 67
Равенство 299
Режим редактирования 20, 47
Режим имитации 20
Режимы работы 20–21
Регистратор
 Рестарт 110
 Добавление сигналов 110
 Сброс 111
 Удаление 110
 Удаление сигналов 110
Размер пакета 22
Рапечатка документации прикладной программы 137
Расписание 39
Расписания 39
Рестарт 19
- С**
- Средства регистрации 112
 Представление регистрируемых величин 113
 Очистка 114
 Настройка графиков 112
 Старт 112
Счетчик времени 299
Счетчик сброса 299
Свойства 22, 52
 Черно-белый 52
- Автоматическая генерация дерева меню 100
Ортогональные соединения 52
Параметры 52, 100
Параметры страницы 52, 99
Показывать строку состояния 100
Показывать строку текущего состояния 52
Показывать только общедоступные идентификаторы 52
Отображать блоки 100
Отображать значения сигналов 99
Отображать только общедоступные идентификаторы 99
Сетевая переменная 33
Сетевой адрес 37
Связанные текстовые файлы 138
связанный текстовый файл 22
связывание 32
СИГНАЛ
 Раскрыть 105
 Выделить 105
 Запись 104
 Изменить 104
 Просмотр 104
 Снять выделение 105
 Снять выделение со всех 105
 Удалить 104
Сигнал
 Добавление сигналов в регистратор 110
Сигналы 39
скорость передачи 22
Скользящее среднее значение 300
СОЕДИНЕНИЕ 60
 Выделить 60
 Переместить узел 60
 Отсоединить 60
 Удалить соединение 60
Соединение
 Разрыв 75
 Окончание 75
 Новое 72

- Подсвечивание 76
 - создать 73
 - Удаление 75
 - удалить последний сегмент 74
 - Соединения 23, 25, 71
 - Выделение 76, 107
 - соединения 23, 25
 - Смена суток 301
 - Совместимость
 - AUT файлы 156
 - COD файлы 156
 - Файлы дерева меню панели оператора 156
 - TAC Menta v3 156
 - СОЕДИНЕНИЕ
 - Создать узел 60
 - создание лога Com 22
 - Строка меню 98
 - Структура меню
 - Редактирование существующего дерева меню 124
 - Авария 122
 - Дата и Время/Летнее время 122
 - Добавление пунктов 119
 - Код доступа 122
 - Копирование и вставка 124
 - Перемещение пунктов меню 124
 - Подменю 120
 - Сервисное меню TAC 123
 - Создание 119
 - Статус 120
 - структурированный тип 36
 - Стандартные приложения/контроллеры
 - Обозначение 151
 - Стандартные сетевые переменные 19
- T**
- Тренд Лог 40, 89
 - Выбор Тренд Лога 90
 - Локальный 89
 - Просмотр 111
 - Определение 91
 - Сигнал 92
 - Тип регистрации 94
 - Тренд Логи 110
 - ТРЕНД
 - Рестарт 105
 - Очистить 105
 - Сброс 105
 - Терминология 19
 - Удаление сигналов, из регистратора 110
 - табличный режим 44
 - Таблица расписаний 89
 - Таблица аварийных сообщений 89
 - Таблица констант 29
 - Редактирование констант 88
 - Добавление константы 86
 - Удаление констант 88
 - таблица конфигурирования I/O 32
 - Таблица модулей расширения I/O 140
 - Мин. время посылки 141
 - Отчет о статусе CNT 141
 - Тип 141
 - Табица модулей расширения I/O
 - Модуль 140
 - Тайм аут 22
 - Тестовые блоки 29
 - только чтение 25, 31
- У**
- УПРАВЛЕНИЕ**
- Свернуть HFB 105
 - Устройство На-линии 35
 - Установки 22
- Ф**
- ФАЙЛ** 132
- Экспорт 132
 - Выход 132
 - Импорт 132
 - Предварительный просмотр 132
 - Параметры печати 132

- Печатать 132
 - Новый 132
 - Открыть 132
 - Сохранить 132
 - Сохранить как 132
 - Файл
 - Экспорт I/O списка 99
 - Выход 99
 - Печатать 99
 - Новый 99
 - Открыть 99
 - Сохранить 99
 - Файл набора символов
 - Определение 130
 - Файл описания панели оператора 126
 - Файлы описания панели оператора
 - Экспорт 130
 - Данные и описания синтаксиса 127
 - Access Code 128
 - Alarm 128
 - Date and Time 128
 - Daylight Saving 128
 - DOP 127
 - Edit Access Code 128
 - Импорт 129
 - ESP 127
 - Пример 129
 - MENU 127
 - OPC 127
 - физическая величина 24
 - Физическое подключение 33
 - ФОРМАТЫ
 - Русские названия дней недели 133
 - Русские названия месяцев 134
 - Аварии 133
 - Дата и время 133
 - Свойства сигнала 133
 - Файл набора символов 134
 - Уставки 134
 - Функции На-линии 145
 - Режим На-линии 146
 - Функциональные блоки 27, 64
 - Функциональные клавиши 43
 - функциональные клавиши 44
 - функциональный блок 23
 - Функциональный блок диаграмм 19
- Х**
- Холодный старт 19
- Ц**
- Числовой 28
 - числовое значение 27
 - чтение/запись 25, 31
- Э**
- Экран структуры меню 118



TAC helps us all to feel and function better - as a direct result of greater comfort. This is made possible by TAC's concept for efficient buildings - Open Systems for Building IT™. This provides our clients with advantages such as energy savings, wider choice and greater flexibility, security and user friendliness.

Trademarks and registered trademarks are the property of their respective owners.

TAC Vista®, TAC Menta®, TAC Xenta® and TAC I-talk® are registered trademarks of TAC AB.





TAC helps us all to feel and function better - as a direct result of greater comfort. This is made possible by TAC's concept for efficient buildings - Open Systems for Building IT™. This provides our clients with advantages such as energy savings, wider choice and greater flexibility, security and user friendliness.

Trademarks and registered trademarks are the property of their respective owners.

TAC Vista®, TAC Menta®, TAC Xenta® and TAC I-talk® are registered trademarks of TAC AB.

