

# Конфигурирование области ввода/вывода ПЛК

## Руководство пользователя

### Содержание

Введение .....	5
Термины, сокращения и форматы данных, используемые в руководстве .....	6
Используемые термины .....	6
Используемые сокращения .....	7
Форматы данных .....	7
1. Запуск Конфигуратора ПЛК.....	9
1.1. Типы модулей в Конфигурации.....	11
1.2. Представление модулей в экранной форме Конфигуратора ПЛК .....	12
1.3. Описание канала модуля.....	13
1.4. Подчиненные подмодули .....	15
2. Описание фиксированных модулей Конфигуратора ПЛК .....	16
2.1. Модуль дискретных входов (Discrete input).....	16
2.1.1. Параметр «Время фильтрации» (Time of filtration) .....	16
2.1.2. Параметр «Видимость» (Visibility) .....	17
2.1.3. Подмодуль «Триггер» (Trigger) .....	17
2.1.3.1. Параметр «Номер входа» (Number of Input).....	18
2.1.3.2. Параметр «Фронт сигнала» (SenseEdge) .....	18
2.1.4. Подмодуль «Счетчик» (Counter 16bit).....	20
2.1.5. Подмодуль «Энкодер» (Encoder 16bit) .....	20
2.1.5.1. Параметры «Первый вход» (First Input) и «Второй вход» (Second Input) .....	21
2.1.5.2. Параметр «Диапазон изменений» (Range) .....	21
2.1.5.3. Параметр «Тип энкодера» (Encoder Type) .....	21
2.2. Модуль дискретного выхода (Discrete output) .....	23
2.2.1. Параметр «Безопасное значение» (Save Value) .....	24
2.2.2. Подмодуль ШИМ (Pulse-wide modulator) .....	24
2.2.2.1. Параметр «Номер выхода» (Number of output) .....	25
2.2.2.2. Параметр «Период ШИМ, 100 мксек» (Period of PWM in 100 mksec).....	25
2.2.2.3. Параметр «Минимальная длительность импульса ШИМ, 100 мксек» (Minimal duration of impulse in 100 mksec) .....	25
2.3. Специальные модули выходов (Special output) и входов (Special input) .....	26
2.4. Модули аналоговых входов.....	28
2.4.1. Модуль «Датчик унифицированного сигнала» (Unified Signal Sensor) .....	29

2.4.1.1. Параметр «Тип датчика» (Type of sensor).....	30
2.4.1.2. Параметр «Время измерения в секундах» (Measure interval, s) .....	30
2.4.1.3. Параметры «Нижняя и верхняя границы» (Ain low & Ain high) .....	30
2.4.1.4. Параметры «Первая, вторая и третья точка» и «Вводимое изменение» (First point и Delta), (Second point и Delta) и (Third point и Delta).....	31
2.4.2. Модуль «Датчик типа «Термопара»» (Thermocouple sensor)	31
2.4.3. Модуль «Датчик типа «Термосопротивление»» (RTD sensor) .....	33
2.4.4. Модуль «Контактный датчик» (Contact sensor) .....	34
2.5. Модуль аналоговых выходов (Analog output) .....	36
2.5.1. Параметр «Тип выхода» (Type) .....	37
2.5.2. Параметры «Коррекция в начальной точке» (Null correction) и «Коррекция в конечной точке диапазона» (Full range correction).....	37
3. Модули Конфигуратора ПЛК, настраиваемые потребителем .....	38
3.1. Модуль Modbus (Slave).....	38
3.1.1. Параметр «Адрес устройства» (Address) .....	39
3.1.2. Подмодуль Modbus (FIX). Настройка коммуникаций модуля .....	39
3.1.2.1. Параметр «Скорость передачи информации» (CommunicationSpeed, bod) .....	43
3.1.2.2. Параметр «Проверка четности» (Parity) .....	43
3.1.2.3. Параметр «Количество бит данных» (DataBits) .....	43
3.1.2.4. Параметр «Количество стоп-битов» (StopLength) .....	43
3.1.2.5. Параметр «Тип интерфейса» (InterfaceType) .....	43
3.1.2.6. Параметр «Тип протокола обмена» (FrameOriented) ...	43
3.1.2.7. Параметр «Время, на которое необходимо задерживать ответ на запрос в мс» (Framing time) .....	44
3.1.2.8. Параметр «Удаленный порт» (RemotePort) .....	44
3.1.3. Настройка данных информационного обмена .....	44
3.1.3.1. Выравнивание переменных при размещении в области памяти.....	46
3.1.3.2. Отображение устройства в области памяти.....	48
3.2. Модуль Modbus (Master) .....	50
3.2.1. Универсальное устройство Modbus (Master).....	51
3.2.1.1. Параметр «IP-адрес» (Module IP) .....	52
3.2.1.2. Параметр «Максимальный тайм-аут, в мс» (Max timeout).....	52
3.2.1.3. Параметр «Порт TCP» (TCP port) .....	52
3.2.1.4. Параметр «Режим работы в сети» (NetMode) .....	52
3.2.1.5. Параметр «Адрес подчиненного устройства» (ModuleSlaveAddress) .....	53

3.2.1.6. Параметр «Период опроса устройства, в мс» (Polling time) .....	53
3.2.2. Настройка входов и выходов модуля Modbus (Master) .....	53
3.2.2.1. Параметр «Адрес регистра опрашиваемого устройства» (Register Address) .....	54
3.2.2.2. Параметр «Номер команды протокола» (Modbus Command) .....	54
3.3. Модуль «Архиватор» (Archiver) .....	55
3.3.1. Параметр «Режим проведения архивации» (Archive Mode) .....	57
3.3.2. Параметр «Тип данных архивации» (Data Archive Type) .....	58
3.3.3. Параметр «Период архивации, сек» (Period of Archiving) ..	58
3.3.4. Параметр «Имя архива» (Archive Name) .....	58
3.3.5. Параметр «Комментарий архива» (Comment) .....	58
3.3.6. Параметры «Время начала архивации» (Start time) и «Время остановки архивации» (Stop time) .....	58
3.3.7. Параметры «Понедельник» (On Sunday) ... «Воскресенье» (On Saturday) .....	59
3.3.8. Подмодуль архивации информации в файл (File Output) ..	59
3.3.8.1. Параметр «Имя файла» (File name) .....	60
3.3.8.2. Параметр «Режим работы модуля» (Mode) .....	61
3.3.8.3. Параметр «Тип» (Type) .....	61
3.3.8.4. Параметр «Размер записи» (Max file size) .....	61
3.4. Модуль «Owen (Slave)» .....	62
3.4.1. Параметр «Имя прибора» (Slave Name) .....	63
3.4.2. Параметр «Длина адреса устройства» (Address Length) ...	63
3.4.3. Параметр «Адрес устройства» (Address) .....	63
3.4.4. Подмодуль Owen (FIX). Настройка коммуникаций модуля	64
3.4.5. Переменные протокола ОВЕН .....	64
3.4.5.1. Параметр «Имя переменной» (Hash name) .....	65
3.4.5.2. Параметры «Индекс прибора» (Index) и «Использовать индекс?» (Use a index?) .....	65
3.4.5.3. Параметр «Тип числа с плавающей точкой» (Float Type) .....	66
3.4.5.4. Параметр «Точность» (Precision) .....	66
3.5. Модуль «Owen (Spy)» .....	67
3.5.1. Переменные протокола ОВЕН .....	68
3.5.1.1. Параметр «Длина адреса» (Address Length) .....	68
3.5.1.2. Параметр «Адрес устройства» (Address) .....	69
3.6. Модуль «Owen (Master)» .....	70
3.6.1. Переменные протокола ОВЕН .....	71
3.6.1.1. Параметр «Максимальное время задержки ответа, мс» (Max response delay, ms) .....	72
3.7. Модуль «DCON (Master)» .....	73
3.7.1. Универсальное устройство DCON (Universal DCON Device)	74
3.7.1.1. Параметры «Формат запроса» (Request format), «Формат правильного ответа» (Good response format) и «Формат неправильного ответа» (Bad response format) .....	75

3.7.1.2. Параметр «Максимальное время ответа» (Max response timeout).....	78
3.7.1.3. Параметр «Режим работы» (Work mode).....	78
3.8. Модуль статистики (Statistic) .....	80
3.9. Константы (Constant value) .....	82
3.9.1. Параметр «Значение константы» (Constant Value) .....	82
3.9.2. Параметр «Комментарий для программы EasyWorkPLC» (Comment for EasyWorkPLC).....	83
<b>Приложение А. Типы датчиков, используемых в качестве аналоговых входов ОВЕН ПЛК .....</b>	<b>84</b>
<b>А.1. Подключение датчиков типа ТСМ50, ТСП100 и ТСМ100 к ПЛК150 .....</b>	<b>84</b>
<b>А.2. Перечень типов датчиков .....</b>	<b>84</b>
<b>Приложение Б. Примеры настройки опроса переменных по протоколу ОВЕН.....</b>	<b>86</b>
<b>Приложение В. Сообщения об ошибках в ПЛК.....</b>	<b>89</b>
<b>В.1. Ошибки датчиков (аналоговых входов) .....</b>	<b>89</b>
<b>В.2. Коды ошибок модулей «Мастер».....</b>	<b>89</b>
<b>В.2.1. Модуль Modbus (Мастер) .....</b>	<b>89</b>
<b>В.2.2. Модуль ОВЕН (Мастер) .....</b>	<b>90</b>
<b>В.2.3. Модуль DCON (Мастер) .....</b>	<b>93</b>
<b>В.3. Коды ошибок модуля «Архиватор» .....</b>	<b>94</b>
<b>В.4. Коды ошибок модуля архивации информации в файл.....</b>	<b>95</b>
<b>Приложение Г. Примеры настройки опроса переменных по протоколу DCON .....</b>	<b>96</b>
<b>Г.1. Опрос значений входов модулей аналоговых входов IPC-7033 .....</b>	<b>96</b>
<b>Г.2. Запись значения выхода модуля IPC-7021 с периодичностью 1 секунда и при смене значения.....</b>	<b>98</b>

## Введение

Настоящее руководство предназначено для ознакомления пользователя с технологией задания и редактирования конфигурации области ввода/вывода программируемого логического контроллера ОВЕН ПЛК. Область ввода/вывода ПЛК включают в себя (дискретные и аналоговые) входы и выходы, модули расширения функционала (в том числе организующие обмен информацией между ПЛК и отдельными приборами и устройствами, связанными по сети с ПЛК). Обмен данными программа ПЛК осуществляет через область ввода/вывода (**%I** и **%Q**). Размеры областей ввода/вывода задаются в **target**-файле, их объем ограничен лицензионным соглашением. В случае неограниченной лицензии пользователь может изменять размер области ввода/вывода. Программа-конфигуратор, таким образом, является редактором ресурсов ПЛК, относящихся к области ввода/вывода.

Задание конфигурации ПЛК осуществляется в среде разработки **CoDeSys** с помощью утилиты **PLC Configuration (Конфигуратор ПЛК)**, расположенной во вкладке ресурсов Организатора объектов. В экранной форме отображается конфигурация ПЛК – ряд модулей, каналов, параметров в соответствующих окнах. Задача пользователя при работе с Конфигуратором – определить совокупность конфигурационных параметров ПЛК, определяющих конфигурацию области ввода/вывода ПЛК.

## Термины, сокращения и форматы данных, используемые в руководстве

В разделе представлены термины (в т.ч. профильные специализированные) и их определения, сокращения и их расшифровки и форматы данных, использованные при разработке настоящего руководства пользователя.

### Используемые термины

**Битовый канал** – элемент структурирования канала. Создается автоматически. Тип всегда BOOL.

**Имя параметра** – уникальный (в данной конфигурации) набор символов, однозначно определяющий доступ к параметру в контроллере.

**Индекс параметра** – числовое значение, отличающее параметры однотипных элементов с одинаковыми именами.

**Канал конфигурации** – элемент структурирования области памяти (собственно данные ввода/вывода). Принадлежит только к одной из областей памяти. Может иметь типы BOOL, BYTE, WORD, DWORD, FLOAT, STRING. Может иметь параметры. Тип идентифицируется по уникальному Id.

**Конфигуратор ПЛК – PLC-Configuration** – редактор **CoDeSys**, в котором определяется состав аппаратных средств и производится настройка определенных параметров ввода-вывода.

**Конфигурация** – совокупность модулей, каналов и значений их параметров, определяющая структуру области ввода/вывода и функционирование ПЛК.

**Модуль конфигурации** – основной элемент конфигурации – независимая единица аппаратных/программных средств. Может иметь подчиненные модули (субмодули), включает набор каналов ввода-вывода. Как и каждый отдельный канал, модуль может иметь параметры. Каждый тип модуля имеет уникальный идентификатор.

**Название параметра** – словесное описание параметра, отражающее его суть.

**Параметр** – атрибут канала или модуля. Значение параметра устанавливается интерактивно до загрузки проекта в контроллер. Оно передается в ПЛК и влияет на его работу.

**Программный модуль** – блок программы прибора, предназначенный для выполнения конкретного действия.

**Формат данных** – тип значений параметров. Различают следующие форматы: целое число, число с плавающей точкой и др.

**Целевой файл (Target file)** – набор файлов, поставляемых производителем ПЛК и описывающих аппаратные и программные

особенности конкретного ПЛК, что позволяет среде разработки корректно взаимодействовать с ПЛК.

## Используемые сокращения

<b>АЦП</b>	– аналогоцифровой преобразователь
<b>ПК</b>	– персональный компьютер
<b>ПЛК</b>	– программируемый логический контроллер
<b>ПО</b>	– программное обеспечение
<b>ЦАП</b>	– цифроаналоговый преобразователь
<b>SCADA</b>	– <i>Supervisory Control And Data Acquisition. Программа, исполняемая на ПК с целью получения и отображения данных в удобном для пользователя виде, с возможностью управления</i>

## Форматы данных

**Логический тип:**

**BOOL**

**Целочисленные типы:**

**BYTE, WORD, DWORD, SINT, USINT, INT, UINT, DINT и UDINT.**

**С плавающей точкой:**

**REAL** – данные в формате с плавающей запятой, используются для сохранения рациональных чисел.

**Строки:**

Строковый тип **STRING** представляет строки символов. Максимальный размер строки определяет количество резервируемой памяти и указывается при объявлении переменной. Размер задается в круглых или квадратных скобках. Если размер не указан, принимается размер по умолчанию - 80 символов. Длина строки не ограничена в CoDeSys, но строковые функции способны обращаться со строками от 1 до 255 символов.

Типы форматов данных характеризуются различными диапазонами сохраняемых данных и, соответственно, различными требованиями к памяти. Подробно данные характеристики представлены в таблице.

Таблица

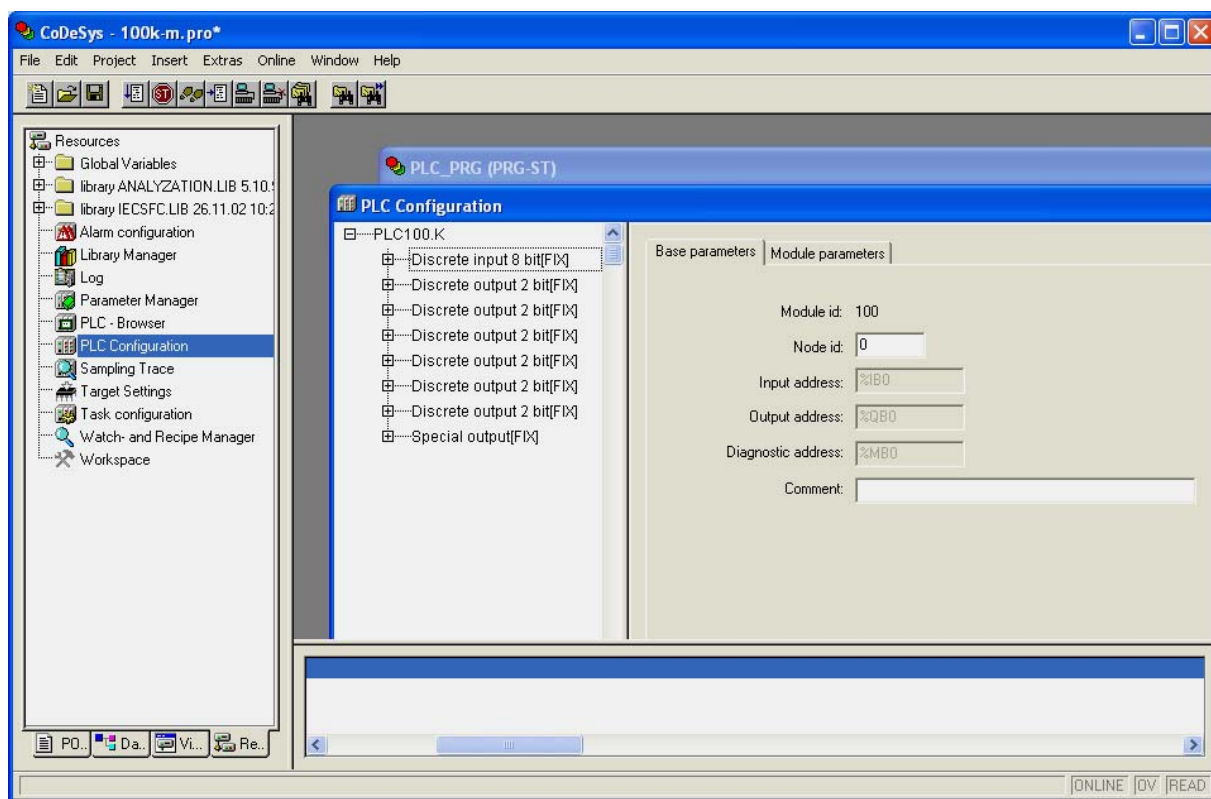
Тип	Нижний предел	Верхний предел	Размер памяти
<b>BYTE:</b>	0	255	8 Бит
<b>WORD:</b>	0	65535	16 Бит
<b>DWORD:</b>	0	4294967295	32 Бита
<b>SINT:</b>	-128	127	8 Бит
<b>USINT:</b>	0	255	8 Бит
<b>INT:</b>	-32768	32767	16 Бита
<b>UINT:</b>	0	65535	16 Бита
<b>DINT:</b>	-2147483648	2147483647	32 Бита
<b>UDINT:</b>	0	4294967295	32 Бита
<b>REAL:</b>	1.175494351e-38F	3.402823466e+38F	32 Бита
<b>STRING:</b>	Любой символ, кроме 0x00		Длина строки в байтах +1 байт на завершающий нуль



# 1. Запуск Конфигуратора ПЛК

Утилита **PLC Configuration (Конфигуратор ПЛК)** доступна пользователю на вкладке ресурсов (**Resources**) Организатора объектов среды разработки **CoDeSys** (далее – **CoDeSys**) (см. рис. 1.1). Конфигуратор является редактором ресурсов ПЛК, относящихся к области ввода-вывода, через которую программа ПЛК осуществляет информационный обмен с внешней средой. В конфигурации присутствуют модули, отвечающие за структурирование областей ввода и/или вывода, каждый из которых может содержать вложенные подэлементы (субмодули и каналы). Для каналов могут быть назначены символические имена. Прямые МЭК адреса отображаются в конфигурации для каждого символического имени.

Первоначальный вид экранной формы редактора конфигурации в окне **CoDeSys** задают файл (файлы) конфигурации **\*.cfg**. Они располагаются в директории, определенной в целевом файле (**Target file**) и считываются при открытии проекта в **CoDeSys**.



**Рис. 1.1. Запуск Конфигуратора ПЛК**

Окно редактора конфигулятора ПЛК разделено на две части. В левой части окна отображается иерархическая структура – дерево **Конфигурации ПЛК**. Структура и компоненты дерева могут быть изменены пользователем **CoDeSys**. В правом окне показаны доступные в момент работы пользователя

**диалоги конфигурации** в виде одной или нескольких табличных вкладок. В этих таблицах задаются значения параметров элементов Конфигурации ПЛК

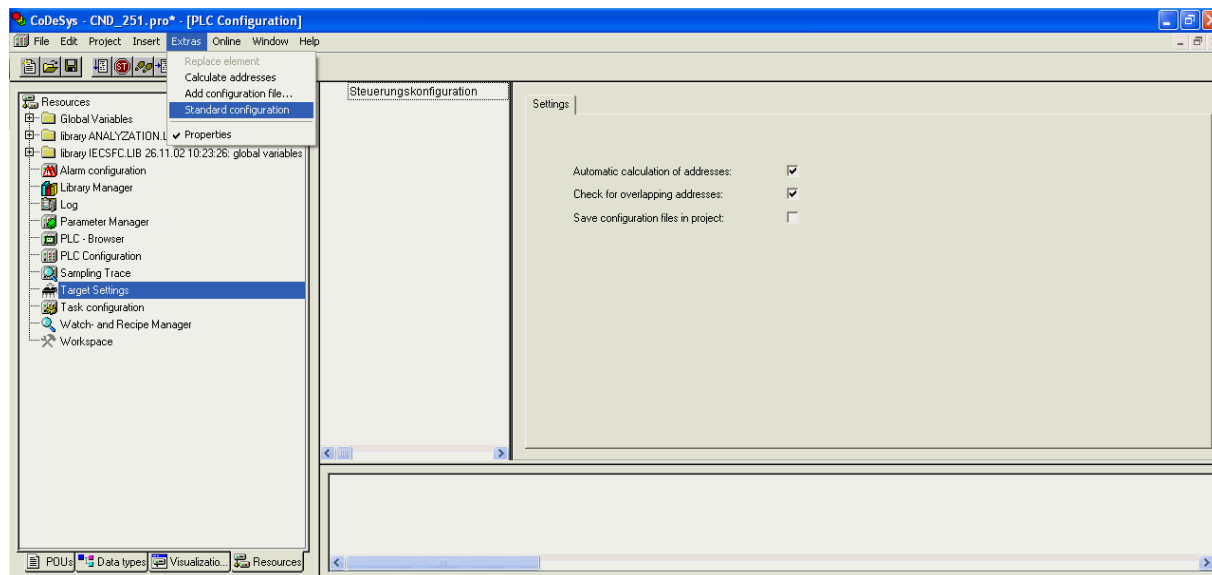
---

**Примечание.** Правая часть окна видна по умолчанию, но, может быть скрыта через опции меню **'Extras' 'Properties'**.

---

**Внимание.** В работе пользователя при переходе от **Target file** иного производителя к **Target file** разработки ОВЕН в окне **PLC Configuration** возникает следующая ситуация: дерево Конфигурации отсутствует. Для создания стандартной Конфигурации пользователь вызывает в главном меню среды разработки опцию **Extras|Standard Configuration** (см. рис. 1.2.), в результате чего появляется экранная форма, аналогичная представленной на рис. 1.1.

**Примечание.** Если потенциально возможен переход от одного **Target File** фирмы Овен к другому, существенно отличающемуся от первого (например, от ПЛК100 к ПЛК150), необходимо задавать переменные в области программ (**POU**) или глобальных переменных (**Global Variables**), т.к. при задании **Standard Configuration** пропадают переменные, заданные в редакторе **PLC Configuration**, и ранее созданное распределение и именование переменных теряется. При объявлении глобальных переменных их имена не будут потеряны, и при переходе к другому **Target File** останется только скорректировать адреса.



**Рис. 1.2. Запуск стандартной Конфигурации**

При настройке конфигурации пользователь может изменить параметры функционирования ПЛК, установленные по умолчанию (см. рис. 1.3):

**Минимальное значение цикла работы ПЛК, в мс (MinCycleLength, ms)** – параметр определяет минимальный период, с которым ПЛК выполняет

полный цикл своей работы. Диапазон значений от 1 до 50 мс, значение по умолчанию – 1 мс.

**Примечание.** Программная реализация ПЛК обеспечивает вызов цикла ПЛК не чаще, чем 1 раз в установленное число мс. В случае, если цикл ПЛК выполняется за время, превышающее установленное значение, его выполнение будет происходить реже.

**Максимальное значение цикла работы ПЛК, в мс (MaxCycleLength, ms)** – параметр определяет максимально допустимое время, за которое ПЛК выполняет полный цикл своей работы. В случае превышения этой величины при работе ПЛК будет принудительно перезагружен. Диапазон значений от 1000 до 10000 мс, значение по умолчанию – 1000 мс.

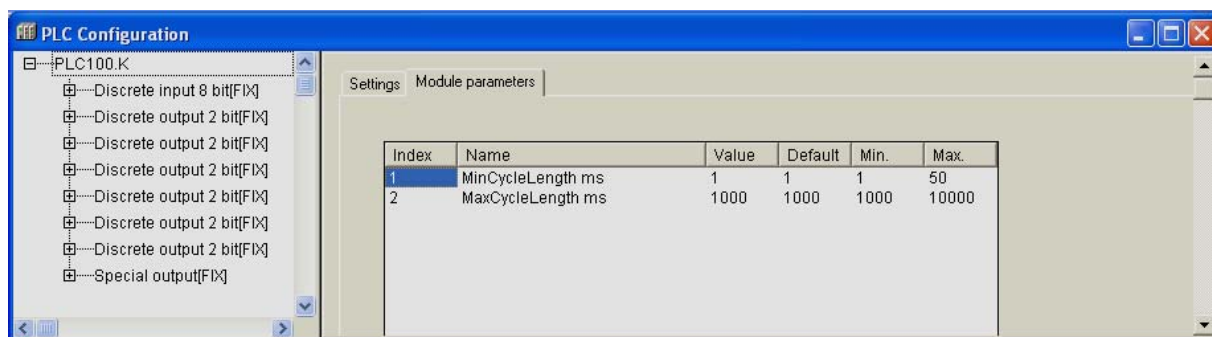


Рис. 1.3. Параметры работы ПЛК

## 1.1. Типы модулей в Конфигурации

В Конфигурации присутствуют 3 типа модулей:

- **фиксированные** – жестко задаются и не могут быть удалены или заменены. Допускается только редактирование их параметров;
- **тип SLOT** – означает, что зарезервировано место для модуля, которое может быть занято или оставлено пустым. На одно зарезервированное место может быть установлен 1 модуль;
- **свободный тип (VAR)** – означает возможность установить любое количество модулей (с учетом физических возможностей области ввода/вывода).

Возможность добавлять модули может быть запрещена или ограничена, т.е. добавлять возможно будет лишь определенные типы модулей.

Добавление модулей типа **SLOT** и **VAR** осуществляется инициированием опции **Append Subelement** («Добавление подэлемента») в контекстном меню, вызываемом нажатием правой кнопки манипулятора «мышь» при установке курсора в дереве **Конфигурации ПЛК**, и последующим выбором конкретного модуля из выпадающего списка. После появления

экранной формы модуля его параметры соответствующим образом настраиваются, как это необходимо для функционирования этого подэлемента.

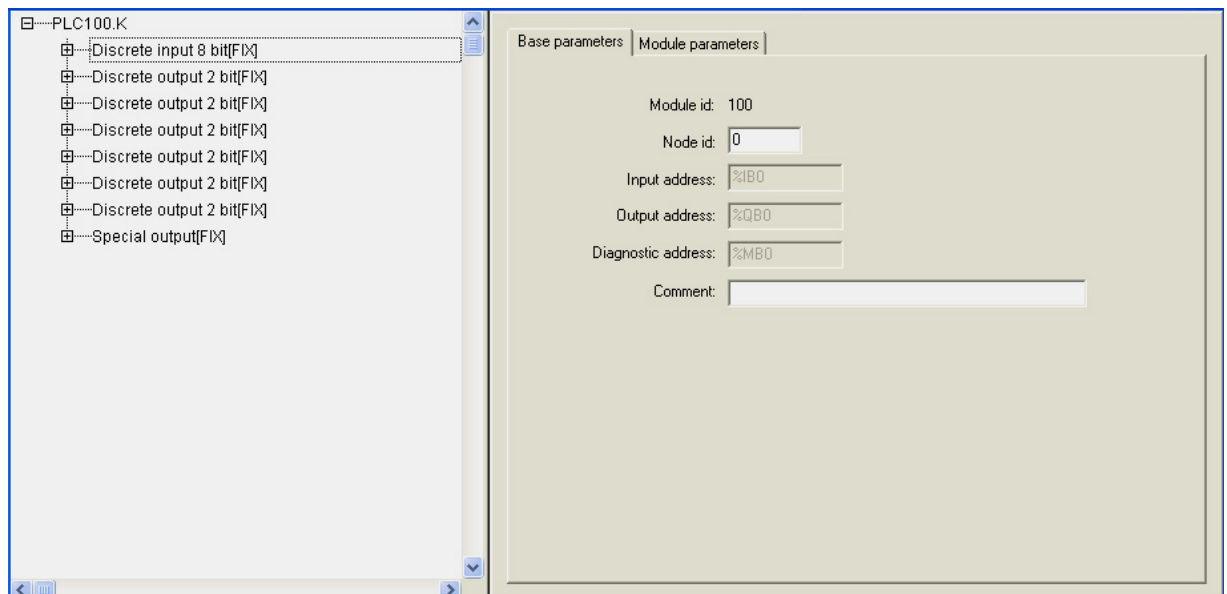
## 1.2. Представление модулей в экранной форме Конфигуратора ПЛК

Окно экранной формы Конфигуратора ПЛК разделено на две части. В левой части окна отображается иерархическая структура Конфигурации ПЛК, в правой – одна из двух вкладок параметров конкретного модуля, выделенного курсором в иерархической структуре, – вкладка базовых параметров или вкладка параметров модуля.

Вкладка базовых параметров (**Base parameters**) идентична для всех модулей и содержит следующие значения:

- идентификационный номер модуля – **Modul id**;
- положение модуля на его уровне иерархии в общей конфигурации – **Node id**. Это значение можно редактировать, в таком случае аналогичные идентификаторы других модулей одного уровня иерархии будут сдвигаться;
- адреса областей ввода-вывода – **Input, Output, Diagnostic Address** – приводятся конкретные номера. Они могут понадобиться при программировании, если будет необходимо обращаться к ним. Значения недоступны для редактирования.

На рис. 1.4. на примере модуля дискретных входов представлена экранная форма Конфигуратора ПЛК, появляющаяся при открытии модуля, с вкладкой базовых параметров в правой части экранной формы.

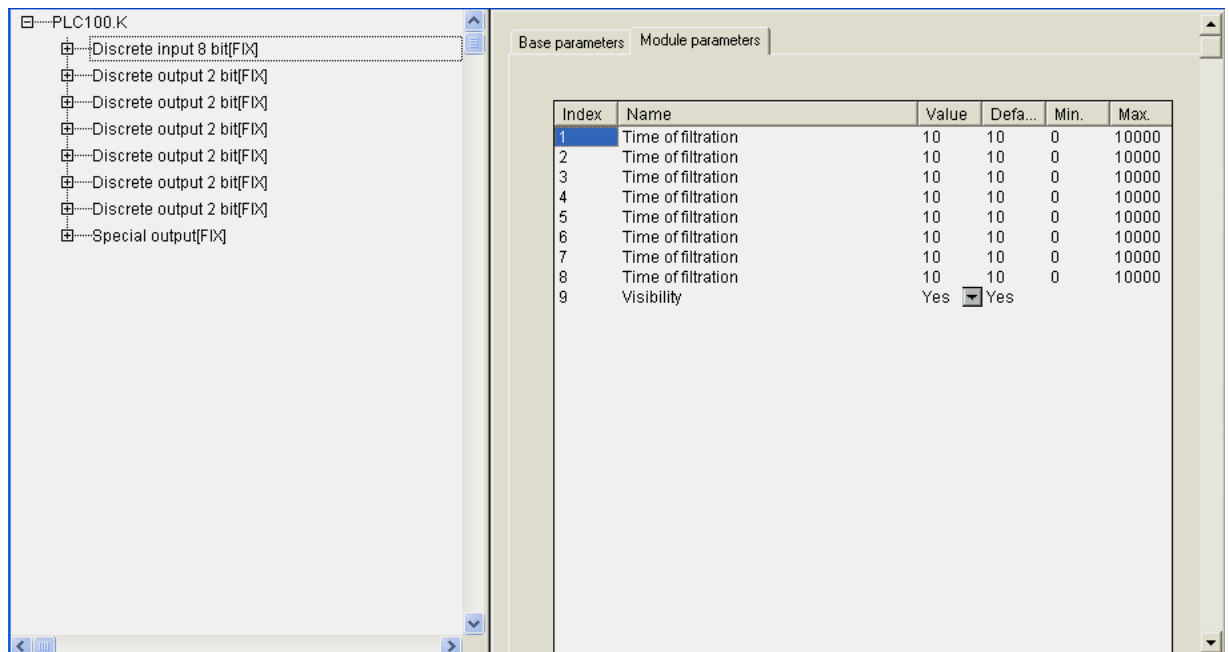


**Рис. 1.4. Модуль дискретных входов.  
Вкладка «Базовые параметры»**

Вторая вкладка правой части экранной формы – «**Параметры модуля**» (**Module parameters**) (см. рис. 1.5) – содержит параметры модуля, представленные в виде таблицы, содержащей столбцы: номер параметра (**Index**), имя (**Name**), значение текущее (**Value**), значение по умолчанию (**Default**) и минимальная (**Min**) и максимальная (**Max**) величины диапазона возможных значений.

**Примечание.** Значения параметров по умолчанию, min-max величины – значения опциональные и не всегда присутствуют во вкладках параметров модулей.

Значения параметров могут быть цифровыми, символьными и выбираемыми из списка.



**Рис. 1.5. Модуль дискретных входов.  
Вкладка «Параметры модуля»**

### 1.3. Описание канала модуля

В составе модуля есть каналы – битовые и байтовые.

Канал – это переносчик единицы информации (данных) от внешнего оборудования в область памяти ввода/вывода. Каждому каналу соответствует переменная в области ввода/вывода.

Канал и соответствующая ему переменная могут быть поименованы, и по присвоенному имени к переменной можно обращаться в программе, либо в программе возможен вызов переменной канала по тому адресу, который у нее установлен аппаратно: например, %IX 0.0.1

8-ми битовый канал может быть использован в программе как 8-битное число, либо как 4 или 8 отдельных друг от друга битов.

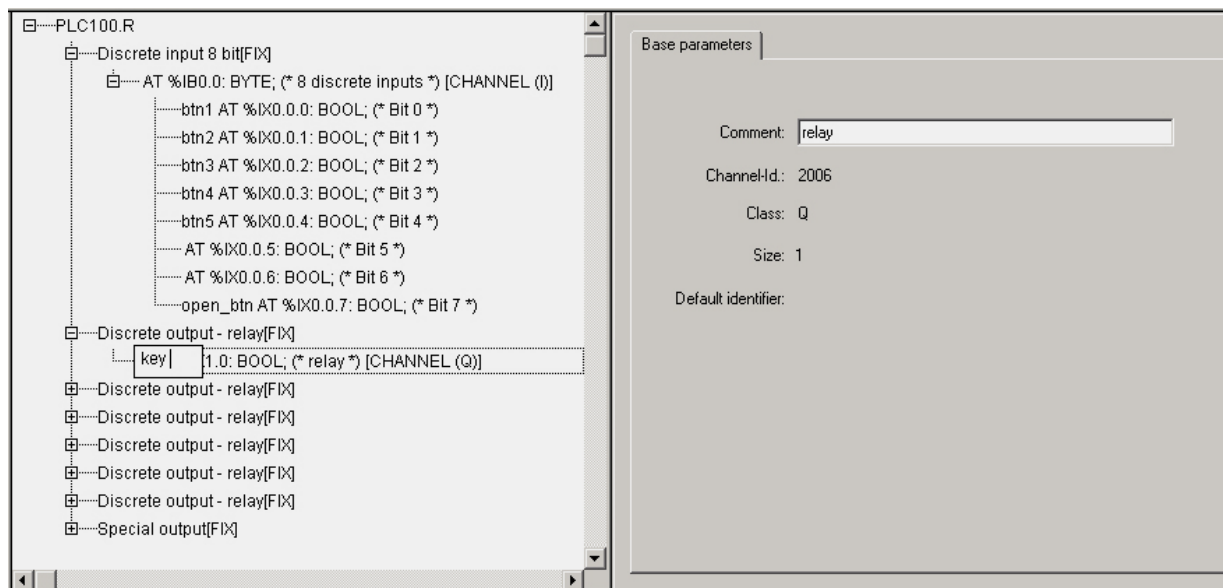
Именованию канала производится следующим образом: двойным щелчком манипулятора «мышь» при курсоре, установленном в начале строки названия канала, осуществляется переход в режим редактирования и вводится имя переменной канала.

---

### Примечания.

1. Имя состоит из латинских букв, цифр и знака «\_» (подчеркивание).
  2. Имя должно начинаться с буквы или знака «\_».
  3. Имя должно быть уникальным.
  4. В некоторых случаях редактирование имен каналов может быть запрещено.
- 

Экранная форма, представленная на рис. 1.6, иллюстрируют процесс именования канала – появление поля ввода символов.



**Рис. 1.6. Ввод и редактирование имени переменной канала**

Данные, появляющиеся во вкладке **«Базовые параметры»**, носят информационный характер и не редактируются.

Для канала программа выводит следующую информацию:

- комментарий – характеристика канала (например, для модуля дискретных входов – «8 discrete inputs» = «8 дискретных входов»);
- идентификационный номер канала в общей иерархии;
- класс;
- размер (в битах).

Для битового канала программа выводит только комментарий с номером битового канала, например, «Bit 3».

## 1.4. Подчиненные подмодули

В Конфигураторе реализована возможность добавления к модулям подчиненных им подмодулей, которые расширяют функционал или изменяют алгоритм работы. Подключение к модулю подчиненного ему подмодуля продемонстрировано на рис. 1.7 на примере модуля дискретных входов.

Добавление подмодулей осуществляется с использованием опции **Append Subelement** (см. п. 1.1) при установке курсора на имени модуля в дереве **Конфигурации ПЛК**, в который добавляется подмодуль.

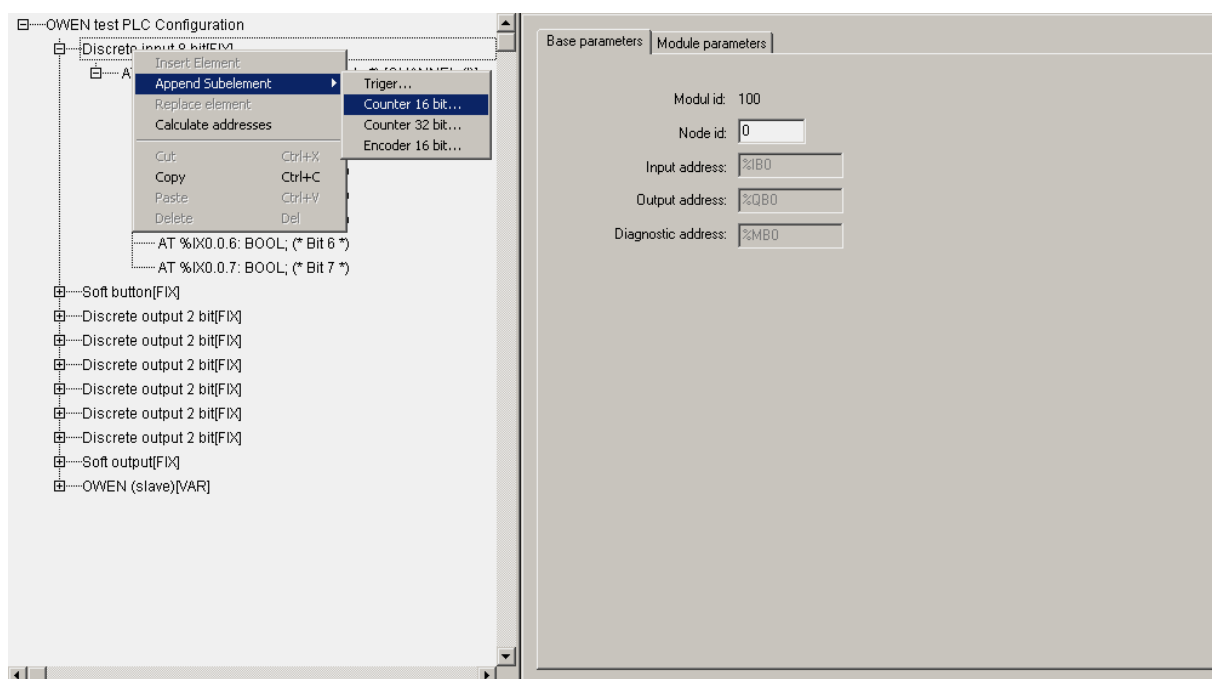


Рис. 1.7. Вызов подчиненного подмодуля

## 2. Описание фиксированных модулей Конфигуратора ПЛК

К фиксированным модулям относятся аппаратные входы и выходы: дискретные или аналоговые.

Другой класс модулей, – модули, вставляемые (добавляемые) в Конфигурацию по желанию пользователя в соответствии с целевыми задачами, – будет описан ниже (см. раздел 3).

### 2.1. Модуль дискретных входов (Discrete input)

Модуль дискретных входов (**Discrete input**) отображает в области памяти ввода/вывода значения дискретных входов ПЛК. **ID=100**.

Модуль имеет 8-ми битовый канал.

**Параметры модуля:**

«**Время фильтрации**» (**Time of filtration**) – диапазон значений от 0 до 10000, значение по умолчанию – 10 (1 ед. = 100 мкс, 10 ед. = 1 мс) (см. п. 2.1.1).

---

**Примечание.** На вкладке модуля дискретных входов представлены восемь одноименных параметров «Время фильтрации» – для каждого битового канала (входа), соответственно.

---

«**Видимость**» (**Visibility**) – задает видимость параметров модуля в программе EasyWorkPLC. Значения выбираются из списка «**yes**» и «**no**», значение по умолчанию – «**yes**» (см. п. 2.1.2).

**Список подмодулей:**

**Триггер (Trigger)** (см. п. 2.1.3), **счетчик (Counter)** (см. п. 2.1.4) и **энкодер (Encoder)** (см. п. 2.1.5).

Вкладки модуля дискретных входов представлены на рис. 1.3. и 1.4.

#### 2.1.1. Параметр «Время фильтрации» (Time of filtration)

**Время фильтрации** – это период 1 опроса значения входного сигнала, задается в сотнях микросекунд (1 ед. = 100 мкс, 10 ед. = 1 мс).

Фильтрация применяется главным образом для подавления дребезжания контактов.

Принцип действия фильтрации:

- при 8 последовательных выборках из входа, если число единиц <2, то результатом фильтрации является логический нуль, >5 – логическая единица



- 5 если единиц 2 – состояние не изменяется.

Фильтрация осуществляется в диапазоне от 1 до 10 000 (0.0001..1 с на выборку).

Режим фильтрации может быть отключен установлением в параметре значения, равного 0. Отключение фильтрации бывает необходимо при работе с подчиненными модулями триггеров, счетчиков и энкодеров для того, чтобы не пропускать высокочастотные сигналы.

### 2.1.2. Параметр «Видимость» (Visibility)

Работа конечного пользователя с Конфигуратором может осуществляться с помощью специализированной программы **EasyWorkPLC** разработки ПО «Овен». При установке для конкретного модуля значения «**yes**» параметра «**Видимость**», параметры этого модуля становятся видны в программе **EasyWorkPLC**.

---

**Внимание.** Параметры разных модулей совпадают (повторяются). Описание конкретных параметров (сущность, качественные характеристики) приводятся, если есть необходимость пояснений, при первом упоминании (появлении) – в теле текущего подраздела или отдельной включаемой статьей.

---

### 2.1.3. Подмодуль «Триггер» (Trigger)

Триггер – программный модуль, позволяющий следить за состоянием входа и подающий соответствующий сигнал об изменении состояния входа с одного на другое.

Модуль «**Триггер**» (**Trigger**) является подчиненным подмодулем модуля дискретных входов и выполняет функцию триггера. **ID=500**.

Модуль имеет битовый канал.

**Параметры модуля:**

«**Номер входа**» (**Number of Input**) – диапазон значений от 0 до 7, значение по умолчанию – 0 (см. п. 2.1.3.1).

«**Фронт сигнала**» (**SenseEdge**) – значения выбираются из списка «**RISE\_EDGE**», «**FALL\_EDGE**» и «**BOTH\_EDGE**», значение по умолчанию – «**RISE\_EDGE**» (см. п. 2.1.3.1).

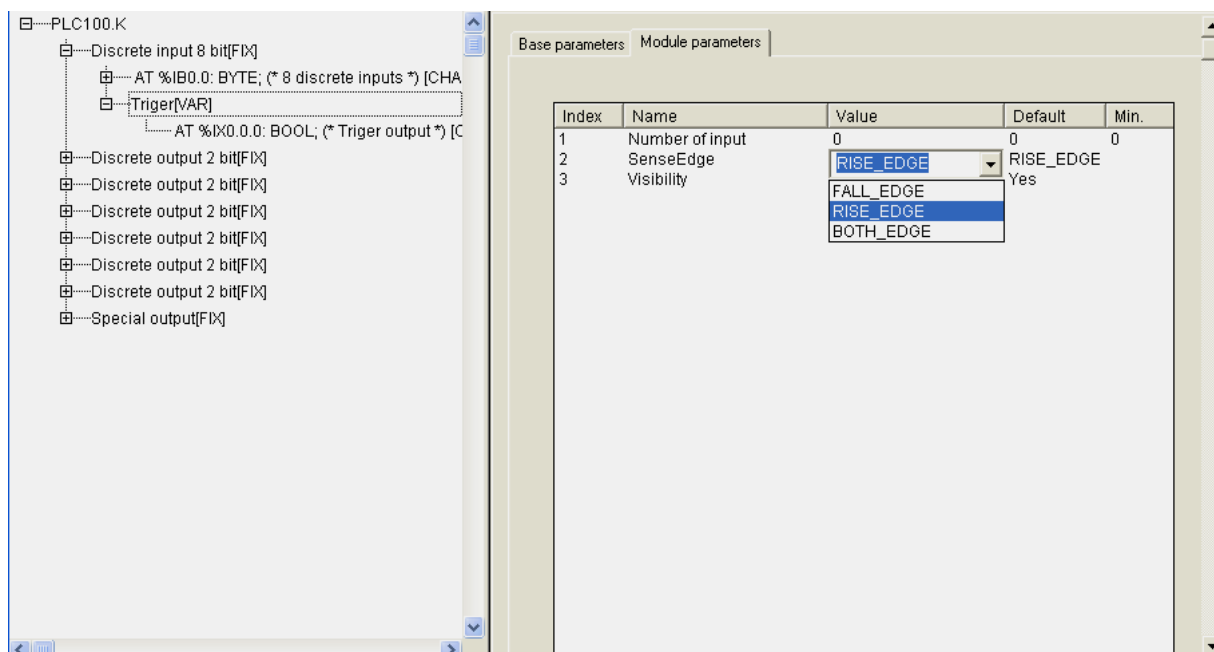
«**Видимость**» (**Visibility**) – задает видимость параметров модуля в программе EasyWorkPLC. Значения выбираются из списка «**yes**» и «**no**», значение по умолчанию – «**yes**» (см. п. 2.1.2).

Модуль «**Триггер**» (**Trigger**) представлен на рис. 2.1.

---

**Внимание.** Значение выхода (в битовом канале) типа BOOL фиксирует состояние прибора: 0 – означает, что с момента предыдущего чтения состояние не изменилось, 1 – зафиксировано изменение.

---



**Рис. 2.1. Модуль «Триггер»**

### **2.1.3.1. Параметр «Номер входа» (Number of Input)**

Параметр **«Номер входа»** – указывает, какой вход модуля дискретных входов из имеющихся будет обрабатываться.

### **2.1.3.2. Параметр «Фронт сигнала» (SenseEdge)**

Параметр **«Фронт сигнала»** – указывает фронт, по которому будет осуществляться работа. Если сигнал представить как прямоугольный импульс, то, очевидно, что у него есть возрастающий и убывающий фронты. Соответственно, может быть задано срабатывание триггера в момент, когда значение сигнала изменяется с 0 на 1, – возрастающий фронт импульса RISE\_EDGE, когда меняется с 1 на 0, – убывающий фронт FALL\_EDGE, или по любому фронту, – BOTH\_EDGE.

---

**Примечание.** Выбор из трех вариантов срабатывания возможен только в том случае, если задан режим фильтрации входа, т.е. значение времени фильтрации больше 0. Если время фильтрации не задано (установлено в 0), особенности аппаратной реализации ОБЕН ПЛК запрещают работу в режиме BOTH\_EDGE.

**Внимание.** Представленное примечание действительно для работы не только триггера, но и счетчика, и энкодера.

---

После чтения программой ПЛК значения триггера, оно обнуляется, т.е. за цикл ПЛК значения не только считываются, но и обнуляются, соответственно, накопления в триггере нет, и, при необходимости суммирования, исполнение этой функции осуществляется в программе.

Графики работы триггера для различных фронтов и сигналов представлены на рис. 2.2.

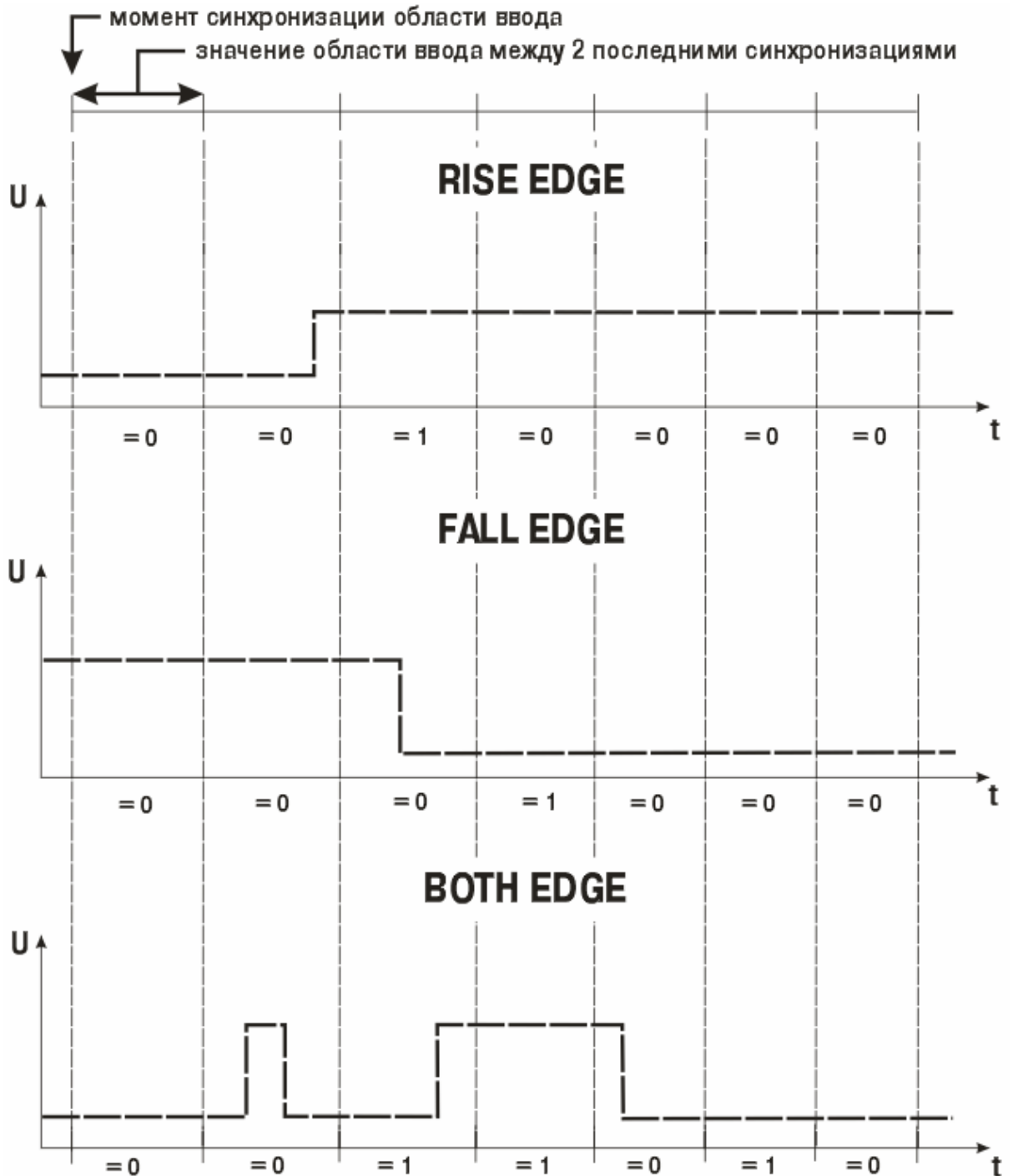


Рис. 2.2. Графики работы триггера для различных фронтов и сигналов

## 2.1.4. Подмодуль «Счетчик» (Counter 16bit)

Счетчик – программный модуль, осуществляющий ведение учета входных импульсов и экспорт учетных данных программе ПЛК, при этом реализована возможность задания метода подсчета импульсов.

Модуль **«Счетчик» (Counter)** является подчиненным подмодулем модуля дискретных входов и выполняет функции счетчика с разрядностью 16 бит. **ID=502.**

Модуль имеет 16-ти битовый канал.

### Параметры модуля:

**«Номер входа» (Number of Input)** – диапазон значений от 0 до 7, значение по умолчанию – 0 (см. п. 2.1.3.1).

**«Фронт сигнала» (SenseEdge)** – значения выбираются из списка **«RISE\_EDGE»**, **«FALL\_EDGE»** и **«BOTH\_EDGE»**, значение по умолчанию – **«RISE\_EDGE»** (см. п. 2.1.3.2).

**«Видимость» (Visibility)** – задает видимость параметров модуля в программе EasyWorkPLC. Значения выбираются из списка **«yes»** и **«no»**, значение по умолчанию – **«yes»** (см. п. 2.1.2).

Счетчик считает количество импульсов, пришедших на дискретный вход за один цикл ПЛК, т.е. между соседними опросами этого счетчика из пользовательской программы. После считывания пользовательской программой значение в счетчике обнуляется.

## 2.1.5. Подмодуль «Энкодер» (Encoder 16bit)

Энкодер – программный модуль, позволяющий осуществлять подключение на 2-х дискретных входах **относительного энкодера** для получения с его помощью данных о вращении или линейном перемещении (относительно конкретного контрольного положения) с последующей передачей информации в цифровой форме в программу ПЛК.

Модуль **«Энкодер» (Encoder)** является подчиненным подмодулем модуля дискретных входов. Запись осуществляется в формате WORD, 16 бит. **ID=504.**

### Параметры модуля:

**«Первый вход» (First Input)** – диапазон значений от 0 до 7, значение по умолчанию – 0 (см. п. 2.1.5.1).

**«Второй вход» (Second Input)** – диапазон значений от 0 до 7, значение по умолчанию – 1 (см. п. 2.1.5.1).

**«Диапазон изменений» (Range)** – диапазон значений от 0 до 65000, значение по умолчанию – 255 (см. п. 2.1.5.2).

**«Тип энкодера» (Encoder Type)** – значения выбираются из списка **«RING»**, **«LINEAR»**, значение по умолчанию – **«RING»** (см. п. 2.1.5.3).

«Видимость» (**Visibility**) – задает видимость параметров модуля в программе EasyWorkPLC. Значения выбираются из списка «yes» и «no», значение по умолчанию – «yes» (см. п. 2.1.2).

Модуль «Энкодер» (**Encoder**) представлен на рис. 2.3.

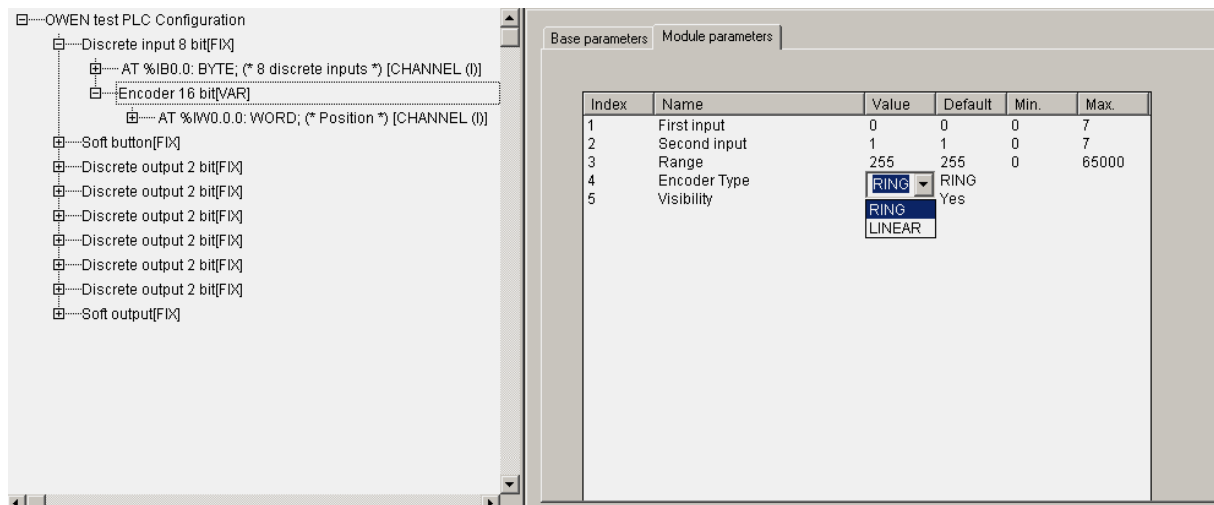


Рис. 2.3. Модуль «Энкодер»

### 2.1.5.1. Параметры «Первый вход» (*First Input*) и «Второй вход» (*Second Input*)

Параметры «Первый вход» (**First Input**) и «Второй вход» (**Second Input**) определяют номера входов ПЛК, к которым подключен энкодер.

---

**Внимание.** При подключении энкодера необходимо соблюдать осторожность, чтобы не поменялось направление вращения; при выходах А и Б подключение осуществляется следующим образом – 1-й вход=А, 2-й вход=Б.

---

### 2.1.5.2. Параметр «Диапазон изменений» (*Range*)

Параметр «Диапазон изменений» (**Range**) – определяет количество импульсов на полный оборот кругового энкодера или на полный ход линейного.

### 2.1.5.3. Параметр «Тип энкодера» (*Encoder Type*)

Параметр «Тип энкодера» (**Encoder Type**) – определяет тип энкодера: либо круговой, либо линейный.

---

**Внимание.** Необходимо учитывать особенность подсчета для типов энкодера:

- при круговом типе – если при вращении осуществляется переход через условный нуль, показания счетчика сбрасываются, и начинается новый отсчет с 0;
  - при линейном типе – осуществляется фиксация при достижении максимума или минимума диапазона, далее приращение не идет.
- 

При считывании информации в программу ПЛК позиция не обнуляется, т.е. энкодер является датчиком положения и фиксирует позицию постоянно в диапазоне от 0 до значения, заданного в параметре «**Диапазон изменений**» (**Range**).

## 2.2. Модуль дискретного выхода (Discrete output)

Модуль дискретного выхода (**Discrete output**) отображает в области памяти ввода/вывода значения дискретного выхода ПЛК. **ID=102, 101**.

Модуль имеет битовый канал (2-х битный или 4-х битный, в зависимости от варианта применяемого выхода).

### Параметры модуля:

«**Безопасное значение**» (**Save Value**) – **TRUE** или **FALSE** для дискретного выхода (см. п. 2.2.1).

«**Видимость**» (**Visibility**) – задает видимость параметров модуля в программе EasyWorkPLC. Значения выбираются из списка «**yes**» и «**no**», значение по умолчанию – «**yes**» (см. п. 2.1.2).

### Список подмодулей:

**ШИМ (Pulse-wide modulator)** (см. п. 2.2.2).

В зависимости от исполнения ПЛК может быть оснащен э/м реле или сдвоенными транзисторными ключами. При комплектации сдвоенными ключами один канал дискретного выхода имеет два битовых канала. При комплектации э/м реле один канал дискретного выхода может содержать один битовый канал или несколько битовых каналов, равных количеству э/м реле в контроллере.

На рис. 2.4 представлены варианты реализации модуля дискретного выхода: 1 битовый канал для э/м реле и 4 битовых канала для 4-х э/м реле.

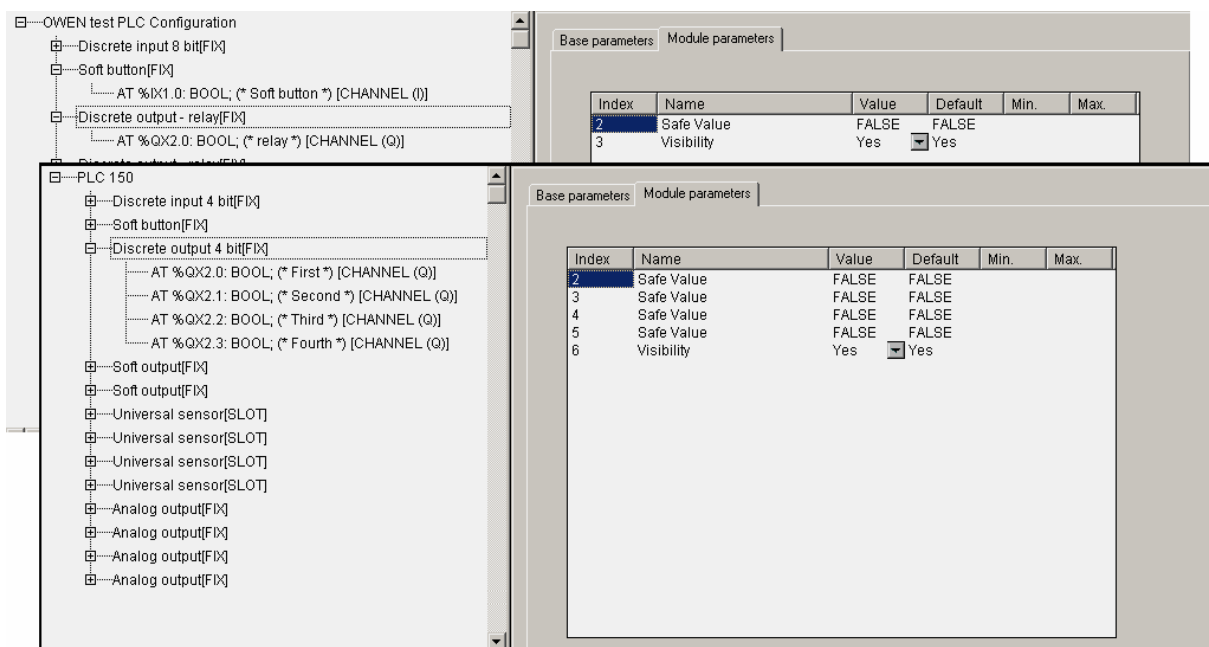


Рис. 2.4. Модуль дискретного выхода

---

**Примечание.** В вариантах применения 2-х битного и 4-х битного дискретных выходов значение параметра **«Безопасное значение» (Save Value)** приводится для каждого однобитного выхода.

---

### 2.2.1. Параметр **«Безопасное значение» (Save Value)**

Назначение параметра **«безопасное значение»** или **«безопасное состояние выхода»** проявляется в следующем. В момент загрузки ПЛК или при каком-либо серьезном сбое, например, при зависании ПЛК и т.п., выходы ПЛК могут оказаться в неопределенном состоянии: выключены или включены. Однако, логика безопасной работы оборудования предполагает, что, как только происходит нештатная ситуация, например, неисправимая авария (ПЛК определяет ее с учетом заданных времен задержек) или при загрузке, ПЛК переводит выходы в заданное заранее в программе безопасное состояние.

Параметр задается в соответствии с типом выхода: для дискретного – **TRUE-FALSE**.

### 2.2.2. Подмодуль ШИМ (Pulse-wide modulator)

Модуль ШИМ – программный модуль, предназначенный для обеспечения функционирования генератора **«широко-импульсный модулятор»**, подключенного на дискретный выход.

Модуль **ШИМ (Pulse-wide modulator)** является подчиненным подмодулем модуля дискретного выхода. Запись осуществляется в формате WORD, 16 бит. **ID=550**.

#### **Параметры модуля:**

**«Номер выхода» (Number of output)** – диапазон значений от 0 до 7, значение по умолчанию – 0 (см. п. 2.2.2.1).

---

**Примечание.** Параметр принимает значения от 0 до (n-1), где n – количество битовых каналов в модуле, где осуществляется подключение.

---

**«Период ШИМ, 100 мксек» (Period of PWM in 100 mksec)** – диапазон значений от 100 до 360000, значение по умолчанию – 1000 (см. п. 2.2.2.2).

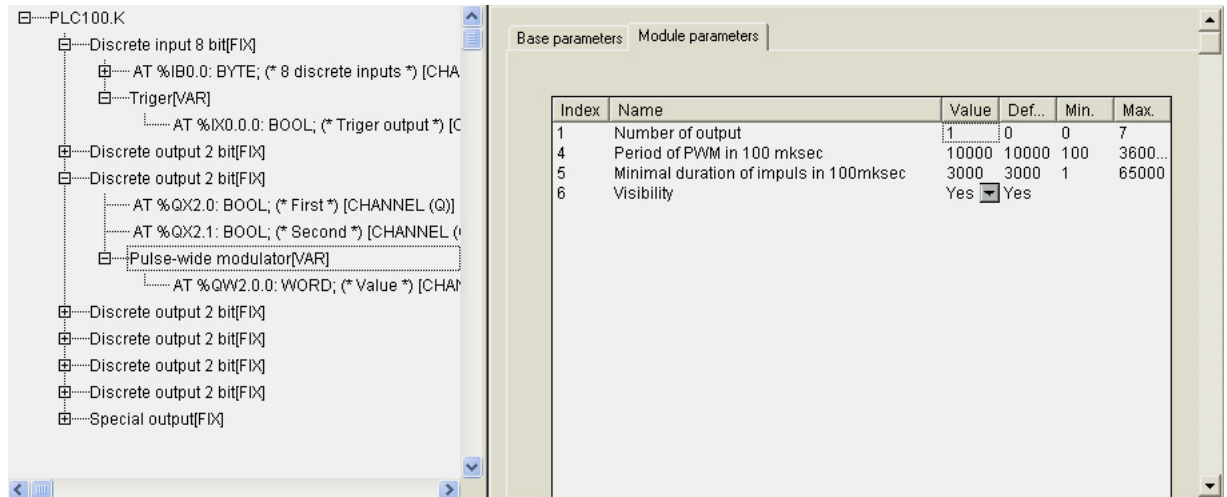
**«Минимальная длительность импульса ШИМ, 100 мксек» (Minimal duration of impulse in 100 mksec)** – диапазон значений от 1 до 65000, значение по умолчанию – 100 (см. п. 2.2.2.3).

**«Видимость» (Visibility)** – задает видимость параметров модуля в программе EasyWorkPLC. Значения выбираются из списка **«yes»** и **«no»**, значение по умолчанию – **«yes»** (см. п. 2.1.2).



При добавлении модуля **ШИМ** появляется канал, в который записывается значение скважности ШИМа (от 0 до 65535 (от 0 до 100% мощности)).

Модуль **ШИМ (Pulse-wide modulator)** представлен на рис. 2.5.



**Рис. 2.5. Модуль ШИМ**

### **2.2.2.1. Параметр «Номер выхода» (Number of output)**

Параметр **«Номер выхода» (Number of output)** – определяет номер выхода ПЛК, к которому подключен ШИМ-генератор.

### **2.2.2.2. Параметр «Период ШИМ, 100 мксек» (Period of PWM in 100 mksec)**

Параметр **«Период ШИМ, 100 мксек» (Period of PWM in 100 mksec)** – определяет длительность одного периода ШИМ-регулирования. Принимает значения от 100 до 360000, соответственно задавая период ШИМ от 10 миллисекунд до 36 секунд.

### **2.2.2.3. Параметр «Минимальная длительность импульса ШИМ, 100 мксек» (Minimal duration of impulse in 100 mksec)**

Параметр **«Минимальная длительность импульса ШИМ, 100 мксек» (Minimal duration of impulse in 100 mksec)** – устанавливает ограничение на минимальную длительность импульса ШИМ.

## 2.3. Специальные модули выходов (**Special output**) и входов (**Special input**)

Модуль специального дискретного входа (**Special input**) – программный модуль, предназначенный для обеспечения функционирования кнопки на верхней панели ПЛК. **ID=103.**

Модуль имеет битовый канал.

### Параметры модуля:

«**Видимость**» (**Visibility**) – задает видимость параметров модуля в программе EasyWorkPLC. Значения выбираются из списка «**yes**» и «**no**», значение по умолчанию – «**yes**» (см. п. 2.1.2).

---

**Примечание.** Число и расположение кнопок зависит от конкретной модели ПЛК.

---

Модуль специального дискретного выхода (**Special output**) – модуль, определяющий переменную бинарного типа, при записи значения в которую происходит включение встроенного в ПЛК устройства подачи звукового сигнала или зажигание специального светодиода на верхней панели ПЛК. **ID=104.**

Модуль имеет битовый канал.

### Параметры модуля:

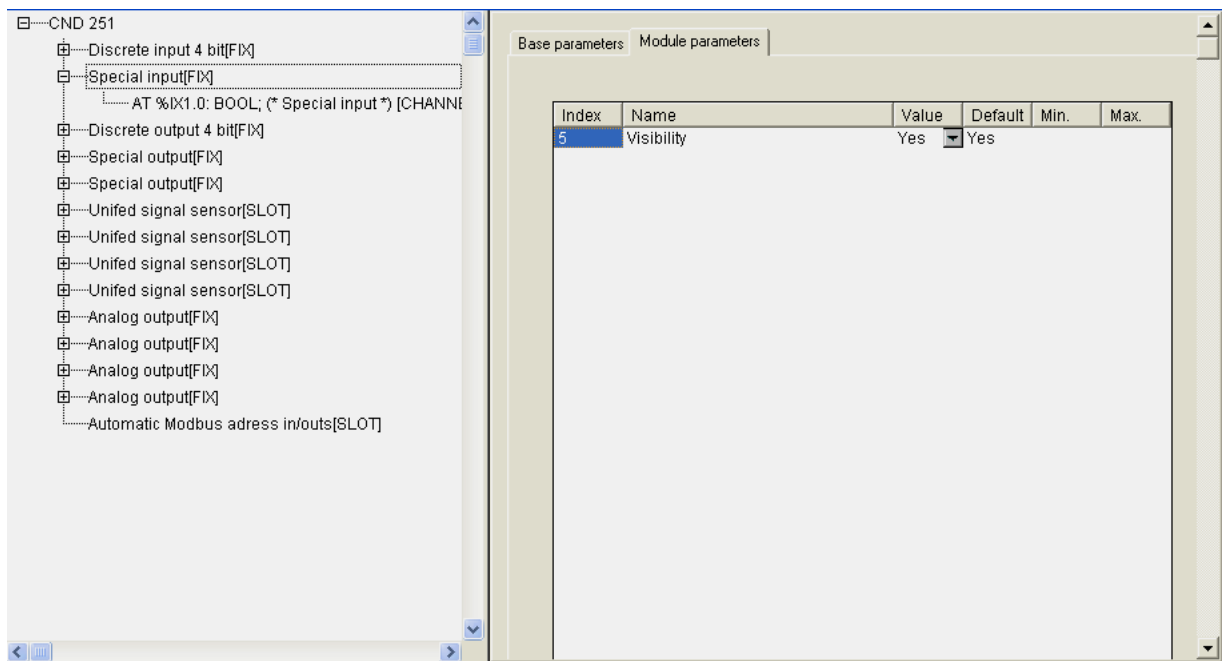
«**Видимость**» (**Visibility**) – задает видимость параметров модуля в программе EasyWorkPLC. Значения выбираются из списка «**yes**» и «**no**», значение по умолчанию – «**yes**» (см. п. 2.1.2).

---

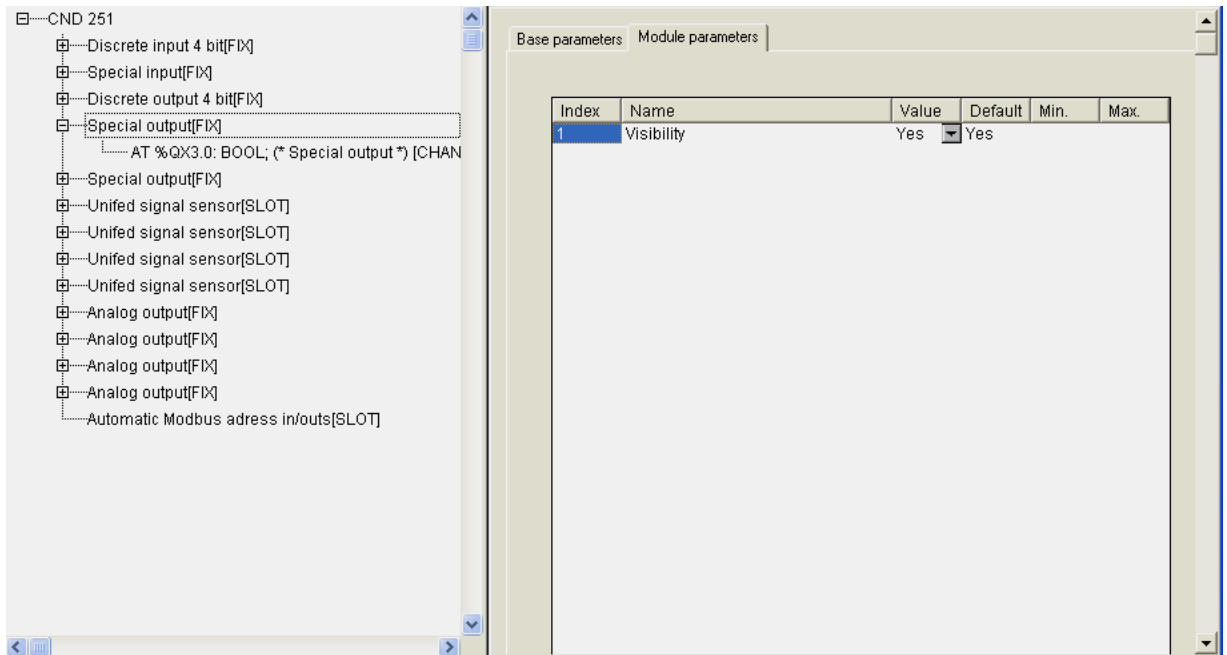
**Примечание.** У отдельных моделей ПЛК может быть только звуковой сигнал, у других – светодиод и звуковой сигнал и т.п.

---

Специальные модули выходов (**Special output**) и входов (**Special input**) отображены на рис. 2.6 и 2.7.



**Рис. 2.6. Модуль битового входа**



**Рис. 2.7. Модуль битового выхода**

## 2.4. Модули аналоговых входов

Аналоговые входы относятся к стационарным модулям. В зависимости от конкретной модели ПЛК их число может меняться, для них в Конфигураторе ПЛК отведено зарезервированное место, в которое по воле пользователя может быть вставлено несколько разных вариантов модулей. Пользователь выбирает необходимые модули функцией контекстного меню **Replace Elements** из 4-х возможных вариантов в выпадающем списке: датчик унифицированного сигнала, датчик типа «термопара», датчик типа «термосопротивление» и контактный датчик.

Модули предназначены для приведения результатов измерений к значениям физической величины, измеряемой датчиком. Приведенное значение физической величины передается в качестве выходного значения в программу ПЛК.

Рис. 2.8 иллюстрирует возможность выбора варианта аналогового входа.

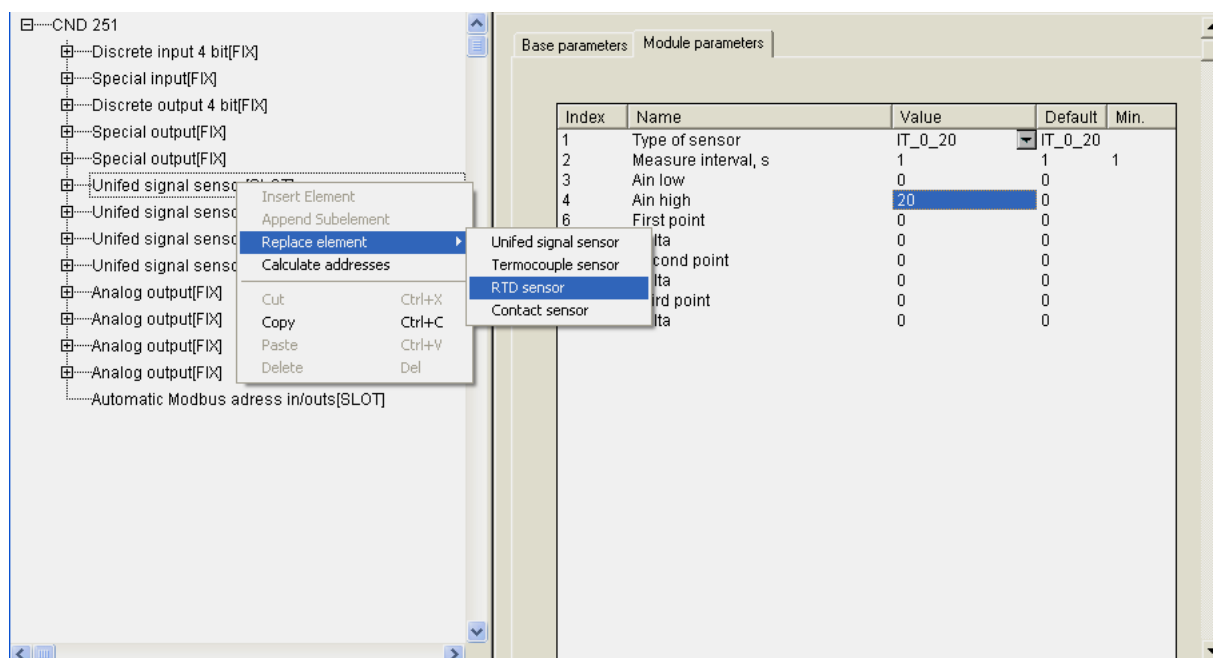


Рис. 2.8. Выбор варианта аналогового входа

У каждого варианта аналоговых входов есть как совпадающие с другими параметры, так и отличающиеся, характерные только для конкретного датчика.

## 2.4.1. Модуль «Датчик унифицированного сигнала» (Unified Signal Sensor)

### Параметры модуля:

«Тип датчика» (**Type of sensor**) – значения выбираются из списка, содержащего 7 возможных вариантов датчиков, значение по умолчанию – **IT\_4\_20** (датчик тока от 4 до 20 мА) (см. п. 2.4.1.1).

«Время измерения в секундах» (**Measure interval, s**) – временной интервал измерения сверху не ограничен, значение может быть любым, в т.ч. с плавающей точкой, но не меньше 1 сек, значение по умолчанию – 1 (см. п. 2.4.1.2).

«Нижняя и верхняя границы» (**Ain low & Ain high**) – границы диапазона измерений физической величины, значения по умолчанию – 0 (см. п. 2.4.1.3).

«Первая, вторая и третья точка» и «Вводимое изменение» (**First point** и **Delta**), (**Second point** и **Delta**) и (**Third point** и **Delta**) – три точки коррекции производимых измерений и вводимые изменения, значения по умолчанию – 0 (см. п. 2.4.1.4).

Выход модуля – **Value («Значение»)**, формат **Real** – значение, получаемое на выходе прибора с учетом всех поправок, приведенное к значениям измеряемой физической величины.

---

**Примечание.** При ошибке датчика (выходе за диапазон) значение не является числом, а содержит код ошибки в специальном формате.

Описание формата и ссылку на функцию извлечения кода ошибки – см. Описание библиотеки функциональных блоков ПО «ОВЕН».

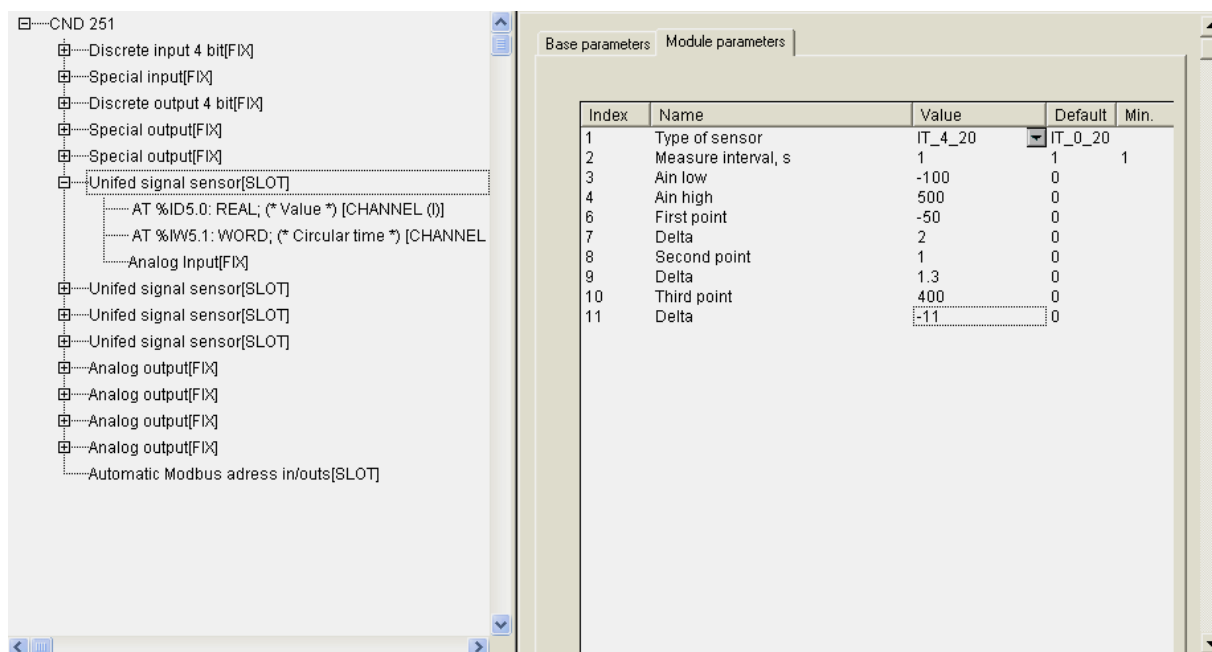
Сообщения об ошибках в ПЛК представлены в Приложении В.

---

**Circular time («Циклическое время»)**, формат **Word** – значение циклического времени измерения в диапазоне от 0 до 65536, сотых долей секунды (точность, необходимая для ПИД-регулирования). Отсчет времени начинается от момента старта программы ПЛК и обнуляется при переходе через максимальное значение.

**Analog Input** – подмодуль, необходимый для организации структуры.

Вкладка параметров аналогового входа «Датчик унифицированного сигнала» представлена на рис. 2.9.



**Рис. 2.9. Вкладка параметров аналогового входа «Датчик унифицированного сигнала»**

#### **2.4.1.1. Параметр «Тип датчика» (Type of sensor)**

Параметр **«Тип датчика» (Type of sensor)** определяет тип датчика, показания которого будут обрабатываться модулем **«Датчик унифицированного сигнала»**. Пользователь осуществляет выбор из 7 возможных вариантов в выпадающем списке: датчик тока от 0 до 20 мА, датчик тока от 4 до 20 мА, датчик тока от 0 до 5 мА, милливольтметр от -50 до +50 мВ, неполярное напряжение от 0 до 1 В, неполярное напряжение от 0 до 10 В и сопротивление от 0 до 5000 Ом.

Перечень используемых типов датчиков представлен в Приложении А.

#### **2.4.1.2. Параметр «Время измерения в секундах» (Measure interval, s)**

Параметр **«Время измерения в секундах» (Measure interval, s)** определяет временной интервал измерений обрабатываемого датчика.

#### **2.4.1.3. Параметры «Нижняя и верхняя границы» (Ain low & Ain high)**

Параметры **«Нижняя и верхняя границы» (Ain low & Ain high)** определяют диапазон измерений датчика унифицированного сигнала.

**Например**, датчик тока от 0 до 20 мА измеряет температуру в диапазоне [-100 °С...+500 °С], и 0 мА ставится в соответствие с нижней границей диапазона измерений – -100 °С, а 20 мА ставится в соответствие с верхней границей диапазона измерений – +500 °С. Тогда выходная величина модуля будет изменяться от -100 до +500. Преобразование линейное на всем диапазоне.

#### **2.4.1.4. Параметры «Первая, вторая и третья точка» и «Вводимое изменение» (First point и Delta), (Second point и Delta) и (Third point и Delta)**

Параметры «Первая, вторая и третья точка» и «Вводимое изменение» (First point и Delta), (Second point и Delta) и (Third point и Delta) определяют корректировку линейности датчика.

Датчик позволяет выполнять коррекцию по трем точкам (полином 2-й степени). Три группы параметров определяют три точки коррекции производимых измерений. По умолчанию значения точек коррекции нулевые, что означает, что ни одна точка не используется. Как только значения [xxx]\_Point устанавливаются ненулевыми, датчик их использует, производя корректировку полинома по заданным точкам измерения и поправкам Delta в этих точках.

Технология коррекции следующая. Например, при калибровке определено, что замер в 1-ой точке – -50 °С, а датчик показывает значение -48 °С. Вносится поправка -2 °С, при сложении с которой получается реальная характеристика, которая в дальнейшем корректирует работу датчика. Также и с остальными точками. Желательно, чтобы при калибровке точки коррекции были расположены далеко друг от друга. При близком расстоянии могут быть большие погрешности: наклон графика будет неверным.

---

**Примечание.** Если есть одна точка – вся кривая смещается на одну корректирующую величину, 2 точки – есть смещение и наклон, 3 – для внесения коррекции используется полином 2-й степени.

Точки задаются, начиная с 1-й. Если будет задана вторая или третья точки, но не задана первая, то контроллер воспримет это, как ошибку.

---

#### **2.4.2. Модуль «Датчик типа «Термопара»» (Thermocouple sensor)**

**Параметры модуля:**

«Тип датчика» (Type of sensor) – значения выбираются из списка, содержащего возможные варианты датчиков, значение по умолчанию – TP\_L

(термопара ТХК (L) с диапазоном контроля [-200 °C...+800 °C]) (см. п. 2.4.1.1).

«**Время измерения в секундах**» (**Measure interval, s**) – временной интервал измерения сверху не ограничен, значение может быть любым, в т.ч. с плавающей точкой, но не меньше 1 сек, значение по умолчанию – 1 (см. п. 2.4.1.2).

«**Нижняя и верхняя границы**» (**Ain low & Ain high**) – границы диапазона измерений физической величины, значения по умолчанию – 0 (см. п. 2.4.1.3).

«**Первая, вторая и третья точка**» и «**Вводимое изменение**» (**First point и Delta**), (**Second point и Delta**) и (**Third point и Delta**) – три точки коррекции производимых измерений и вводимые изменения, значения по умолчанию – 0 (см. п. 2.4.1.4).

Выход модуля – **Value («Значение»)**, **формат Real** – значение, получаемое на выходе прибора с учетом всех поправок, приведенное к значениям измеряемой физической величины.

---

**Примечание.** При ошибке датчика (выходе за диапазон) значение не является числом, а содержит код ошибки в специальном формате.

Описание формата и ссылку на функцию извлечения кода ошибки – см. Описание библиотеки функциональных блоков ПО «ОВЕН».

Сообщения об ошибках в ПЛК представлены в Приложении В.

---

**Circular time («Циклическое время»)**, **формат Word** – значение циклического времени измерения в диапазоне от 0 до 65536, сотых долей секунды (точность, необходимая для ПИД-регулирования). Отсчет времени начинается от момента старта программы ПЛК и обнуляется при переходе через максимальное значение.

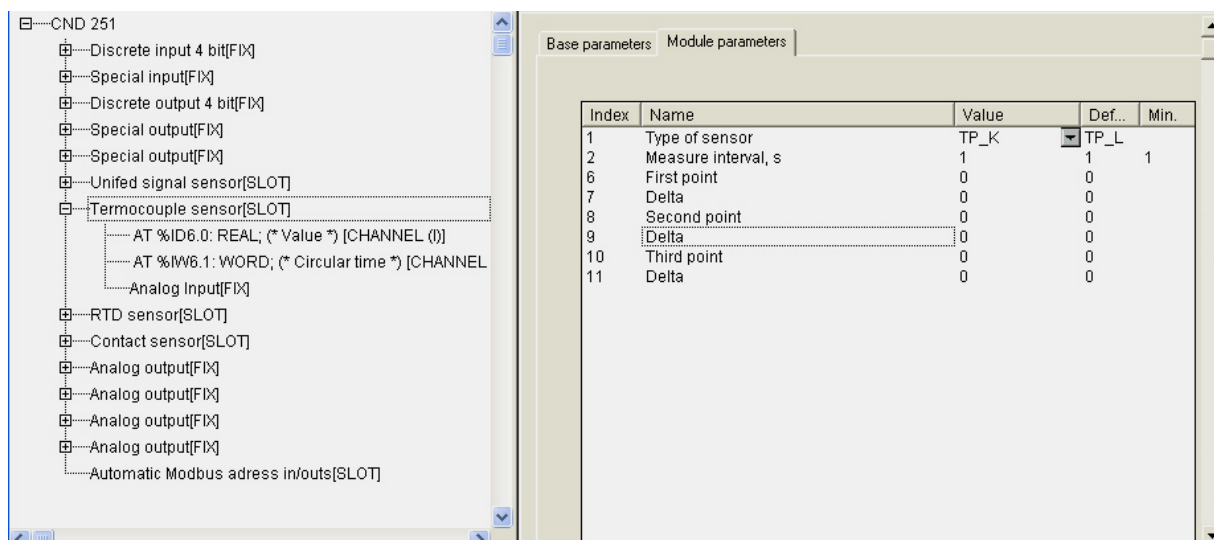
**Analog Input** – подмодуль, необходимый для организации структуры.

Перечень параметров аналогового входа «**Термопара**» идентичен перечню параметров модуля аналогового входа «**Датчик унифицированного сигнала**». Разница – в выборе возможных значений параметра «**Тип датчика**» (**Type of sensor**). Пользователю предоставляется на выбор выпадающий список возможных термопар разных типов. Эти же типы термопар используются во всех приборах с универсальным входом разработки ПО «Овен», например, в ТРМ148.

Перечень используемых типов датчиков представлен в Приложении А.

Вкладка параметров аналогового входа «Термопара» представлена на рис. 2.10.





**Рис. 2.10. Вкладка параметров аналогового входа «Термопара»**

### 2.4.3. Модуль «Датчик типа «Термосопротивление»» (RTD sensor)

#### Параметры модуля:

«**Тип датчика**» (**Type of sensor**) – значения выбираются из списка, содержащего возможные варианты датчиков, значение по умолчанию – **r385\_50** (термопреобразователь сопротивления ТСП (Pt 50)  $W_{100} = 1,3850$  с диапазоном контроля  $[-200\text{ }^{\circ}\text{C}...+750\text{ }^{\circ}\text{C}]$ ) (см. п. 2.4.1.1).

«**Время измерения в секундах**» (**Measure interval, s**) – временной интервал измерения сверху не ограничен, значение может быть любым, в т.ч. с плавающей точкой, но не меньше 1 сек, значение по умолчанию – 1 (см. п. 2.4.1.2).

«**Нижняя и верхняя границы**» (**Ain low & Ain high**) – границы диапазона измерений физической величины, значения по умолчанию – 0 (см. п. 2.4.1.3).

«**Первая, вторая и третья точка**» и «**Вводимое изменение**» (**First point** и **Delta**), (**Second point** и **Delta**) и (**Third point** и **Delta**) – три точки коррекции производимых измерений и вводимые изменения, значения по умолчанию – 0 (см. п. 2.4.1.4).

Выход модуля – **Value** («**Значение**»), формат **Real** – значение, получаемое на выходе прибора с учетом всех поправок, приведенное к значениям измеряемой физической величины.

---

**Примечание.** При ошибке датчика (выходе за диапазон) значение не является числом, а содержит код ошибки в специальном формате.

Описание формата и ссылку на функцию извлечения кода ошибки – см. Описание библиотеки функциональных блоков ПО «ОВЕН».

---

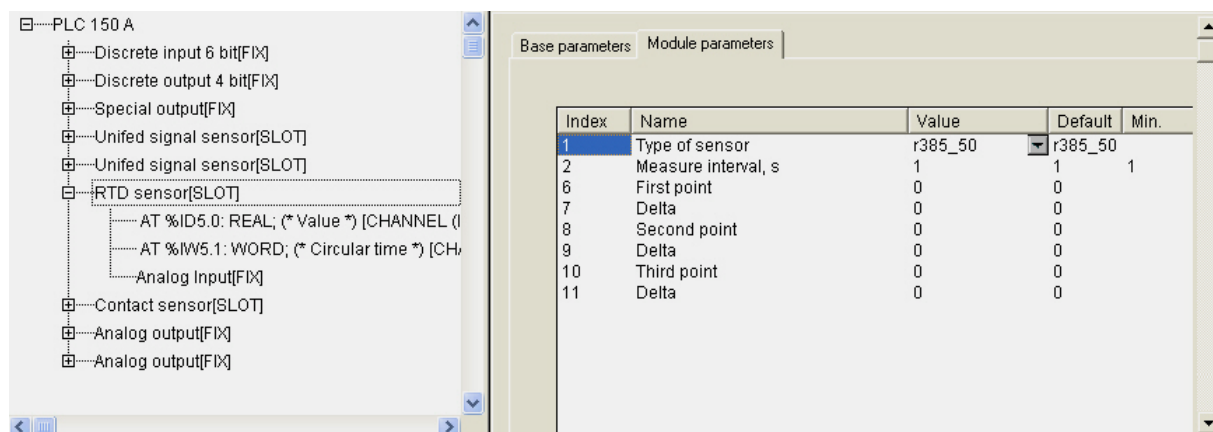
**Circular time («Циклическое время»)**, формат **Word** – значение циклического времени измерения в диапазоне от 0 до 65536, сотых долях секунды (точность, необходимая для ПИД-регулирования). Отсчет времени начинается от момента старта программы ПЛК и обнуляется при переходе через максимальное значение.

**Analog Input** – подмодуль, необходимый для организации структуры.

Перечень параметров аналогового входа **«Термосопротивление»** идентичен перечням параметров аналоговых входов **«Термопара»** и **«Датчик унифицированного сигнала»**. Разница – в выборе возможных значений параметра **«Тип датчика» (Type of sensor)**. Пользователю предоставляется на выбор выпадающий список возможных термосопротивлений разных типов.

Перечень используемых типов датчиков представлен в Приложении А.

Вкладка параметров аналогового входа **«Термосопротивление»** представлена на рис. 2.11.



**Рис. 2.11. Вкладка параметров аналогового входа «Термосопротивление»**

#### **2.4.4. Модуль «Контактный датчик» (Contact sensor)**

Тип аналогового входа – **«Контактный датчик»** – используется в тех случаях, когда при конкретной схеме применения ПЛК не хватает дискретных входов и приходится использовать один из аналоговых входов.

##### **Параметры модуля:**

**«Тип датчика» (Type of sensor)** – значения выбираются из списка, содержащего возможные варианты датчиков, значение по умолчанию – **КТ** (см. п. 2.4.1.1).

«**Время измерения в секундах**» (**Measure interval, s**) – временной интервал измерения сверху не ограничен, значение может быть любым, в т.ч. с плавающей точкой, но не меньше 1 сек, значение по умолчанию – 1 (см. п. 2.4.1.2).

Выход модуля – **Value («Значение»)**, **формат Real** – значение, получаемое на выходе прибора с учетом всех поправок, приведенное к значениям измеряемой физической величины.

---

**Примечание.** При ошибке датчика (выходе за диапазон) значение не является числом, а содержит код ошибки в специальном формате.

Описание формата и ссылку на функцию извлечения кода ошибки – см. Описание библиотеки функциональных блоков ПО «ОВЕН».

Сообщения об ошибках в ПЛК представлены в Приложении В.

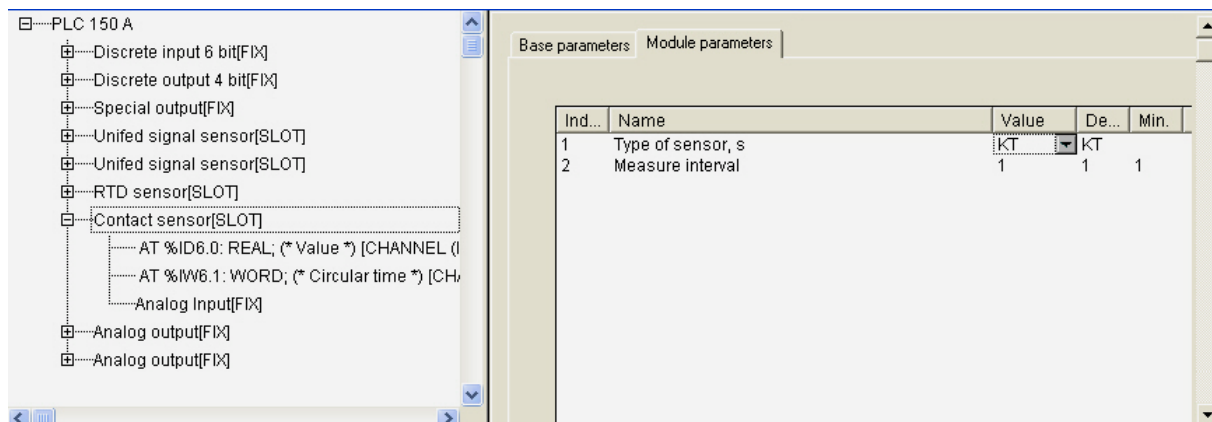
---

**Circular time («Циклическое время»)**, **формат Word** – значение циклического времени измерения в диапазоне от 0 до 65536, сотых долях секунды (точность, необходимая для ПИД-регулирования). Отсчет времени начинается от момента старта программы ПЛК и обнуляется при переходе через максимальное значение.

**Analog Input** – подмодуль, необходимый для организации структуры.

Выходное значение контактного датчика может быть равно 1 или 0. Значение 1 означает, что сопротивление контакта, замыкающего общую клемму аналоговых входов и клемму конкретного входа менее 50 Ом. Размыкание контакта, а также сопротивления более 50 Ом интерпретируются как нулевое выходное значение.

Вкладка параметров аналогового входа «**Контактный датчик**» представлена на рис. 2.12.



**Рис. 2.12. Вкладка параметров аналогового входа «Контактный датчик»**

## 2.5. Модуль аналоговых выходов (Analog output)

Аналоговые выходы относятся к стационарным модулям. В зависимости от установленных ЦАП возможна настройка аналоговых выходов в 3-х вариантах:

- универсальный – программно настраиваемый либо на ток, либо на напряжение;
- фиксированный токовый;
- фиксированный напряжения.

### Параметры модуля:

«Тип выхода» (**Type**) – значения выбираются из списка «**Current 0-20 mA**» и «**Voltage 0-10V**», значение по умолчанию – «**Voltage 0-10V**» (см. п. 2.5.1).

«Коррекция в начальной точке» (**Null correction**) и «Коррекция в конечной точке диапазона» (**Full range correction**) – значения выбираются, исходя из необходимой корректировки, значения по умолчанию – 0 (см. п. 2.5.2).

«Безопасное значение» (**Save Value**) – значение задается в тех же единицах, что и основные, значение по умолчанию – 0 (см. п. 2.2.1).

«Видимость» (**Visibility**) – задает видимость параметров модуля в программе EasyWorkPLC. Значения выбираются из списка «**yes**» и «**no**», значение по умолчанию – «**yes**» (см. п. 2.1.2).

На рис. 2.13 представлена вкладка модуля аналогового выхода.

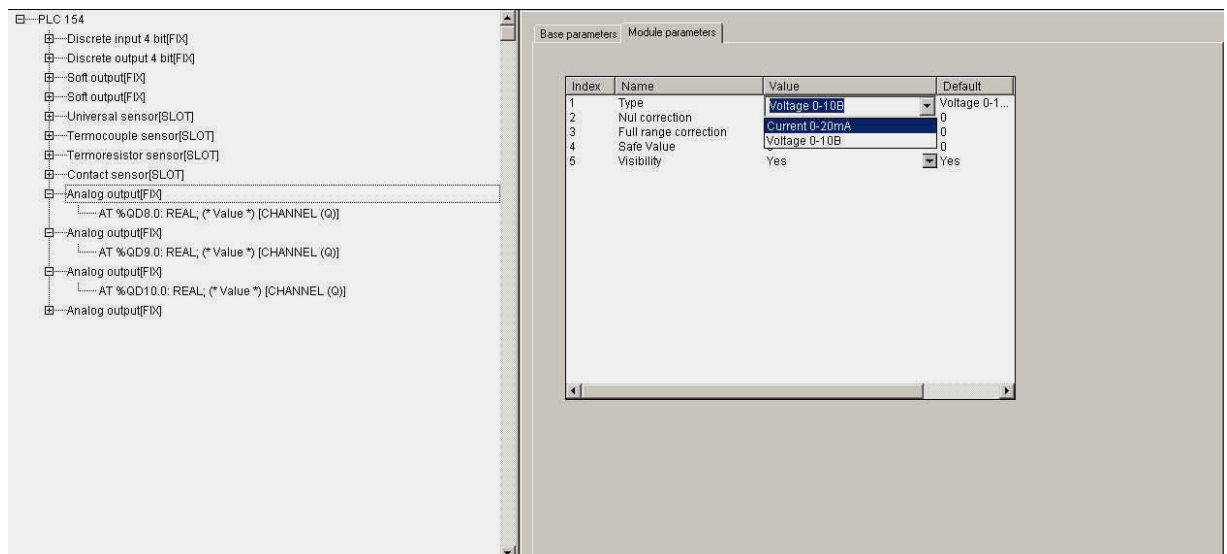


Рис. 2.13. Вкладка параметров модуля аналогового выхода

### 2.5.1. Параметр «Тип выхода» (Type)

Параметр «Тип выхода» (Type)

**Type** – «тип выхода» – параметр, определяющий выбор типа аналогового выхода.

---

**Примечание.** При использовании неуниверсальных ЦАП тип выхода задается жестко и не изменяется.

---

В значение аналогового модуля из программы ПЛК передается число, равное желаемому выходному значению. Для ЦАП типа «Ток» это значения от 4 до 20, для напряжения – от 0 до 10. Значения передаются в формате **Real**.

### 2.5.2. Параметры «Коррекция в начальной точке» (Null correction) и «Коррекция в конечной точке диапазона» (Full range correction)

Группа параметров «Коррекция в начальной точке» (Null correction) и «Коррекция в конечной точке диапазона» (Full range correction) определяет две точки коррекции. Точки выбраны следующим образом: по току – 4 и 15 мА, а по напряжению – 1В и 10В. В параметрах коррекции могут быть заданы корректирующие значения в случае неточной работы ЦАП.

## 3. Модули Конфигуратора ПЛК, настраиваемые потребителем

В данном разделе описаны модули, добавляемые в Конфигурацию по желанию пользователя в соответствии с целевыми задачами и направлениями использования конкретного ПЛК. Эти модули могут добавляться программистом, сопровождающим работу ПЛК, настраиваться соответствующим задачам образом, соответственно, могут и удаляться.

### 3.1. Модуль Modbus (Slave)

**Modbus** – это протокол обмена, и разработанный для ПЛК одноименный модуль обеспечивает работу ПЛК в соответствии с этим протоколом. По протоколу **Modbus** устройства, и ПЛК в т.ч., могут работать в 2-х режимах: режиме «**ведомый (slave)**», когда устройство пассивно отвечает на запросы отдельного Мастера, и в режиме «**ведущий (master)**», когда устройство само формирует запросы другим приборам. В данном разделе рассматривается работа ПЛК в пассивном режиме, т.е. режиме «**slave**». Описываемый модуль – составной и имеет в своем составе подмодули.

#### Параметры модуля:

«**Адрес устройства**» (**Address**) – диапазон значений от 1 до 247, значение по умолчанию – 0 (см. п. 3.1.1).

«**Видимость**» (**Visibility**) – задает видимость параметров модуля в программе EasyWorkPLC. Значения выбираются из списка «**yes**» и «**no**», значение по умолчанию – «**yes**» (см. п. 2.1.2).

#### Список подмодулей:

**Modbus (FIX)** (см. п. 3.1.2).

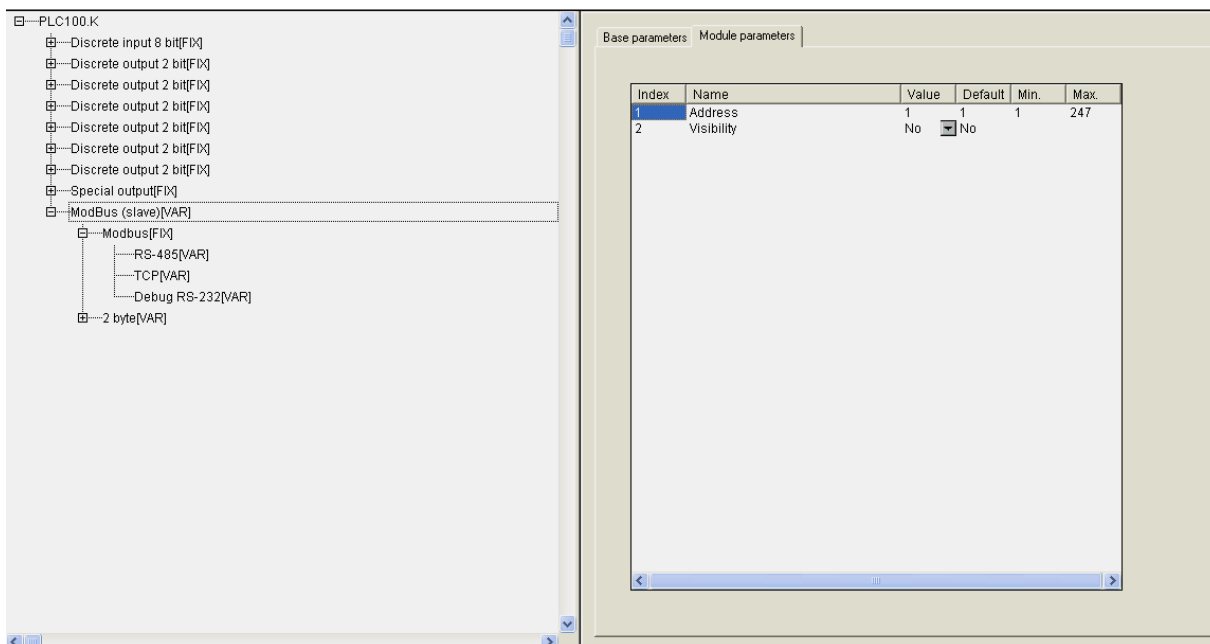
Переменные, которыми будет обмениваться ПЛК по протоколу **Modbus**, выбираются пользователем функцией контекстного меню **Append Subelements**.

---

**Внимание.** При пропадании питания в процессе работы ПЛК последние (текущие) значения переменных сохраняются в энергонезависимой памяти и восстанавливаются при возобновлении работы прибора.

---

Модуль **Modbus (Slave)** представлен на рис. 3.1.



**Рис. 3.1. Модуль Modbus (Slave)**

### 3.1.1. Параметр «Адрес устройства» (Address)

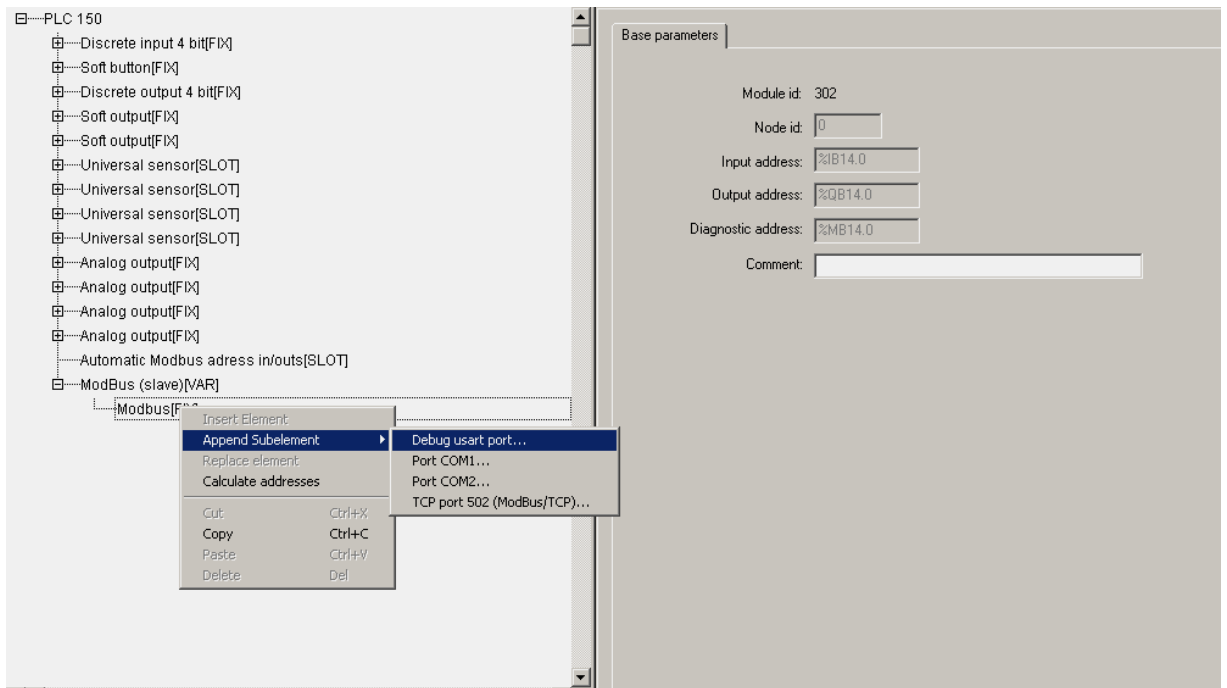
Значение параметра «**Адрес устройства**» (**Address**) – это адрес прибора, по которому посылаются запросы. Параметр имеет значения в диапазоне от 1 до 247, при этом может быть равен 0, и именно это значение выставлено по умолчанию, но применяется в специфических случаях.

### 3.1.2. Подмодуль Modbus (FIX). Настройка коммуникаций модуля

Функционирование ПЛК сопровождается информационным обменом с другими устройствами, включенными наряду с ПЛК в одну систему. Настройка коммуникаций, соответственно, является приоритетной задачей.

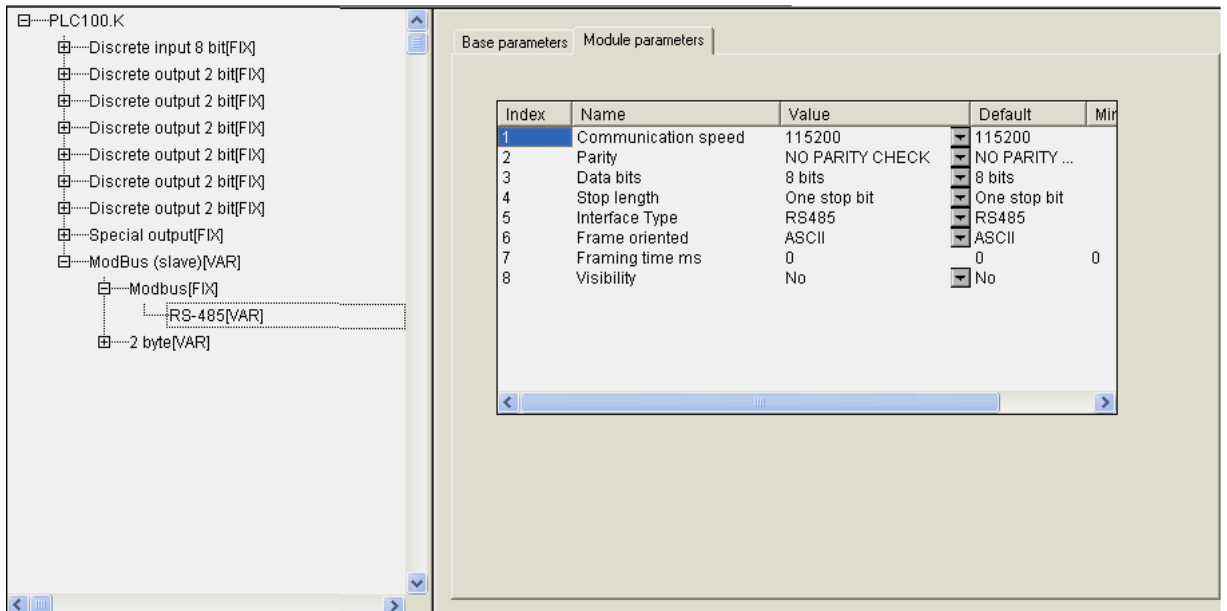
В состав модуля **Modbus (Slave)** фиксировано вставлен подмодуль **Modbus (FIX)** – внутренний модуль, создающий определенную структуру, к которой уже подключается коммутационный интерфейс. ПЛК позволяет обмениваться данными по большому количеству интерфейсов, среди них могут быть **RS-232**, **RS-485** и **TCP (Ethernet)**.

В ПЛК предусмотрены разные подмодули для осуществления обмена по конкретным коммутационным интерфейсам. Рис. 3.2 демонстрирует процесс подключения подмодулей.



**Рис. 3.2. Добавление модулей настройки коммуникаций**

На рис. 3.3. представлена вкладка параметров последовательного порта.



**Рис. 3.3. Вкладка параметров порта**

Параметры последовательных портов, используемых в ПЛК, абсолютно идентичны, при описании их функционирования меняется только название и собственно физический порт, в котором происходит работа.



### **Параметры последовательного порта:**

«**Скорость передачи информации**» (**CommunicationSpeed, bod**) – значения выбираются из списка (2400, 4800, 9600, 14400, 19200, 28800, 38400, 57600, 115200), значение по умолчанию – 115200 (см. п. 3.1.2.1).

«**Проверка четности**» (**Parity**) – значения выбираются из списка, значение по умолчанию – NO PARITY CHECK (отсутствие проверки четности) (см. п. 3.1.2.2).

«**Количество бит данных**» (**DataBits**) – значения в диапазоне от 5 до 8 бит, значение по умолчанию – 8 (см. п. 3.1.2.3).

«**Количество стоп-битов**» (**StopLength**) – значения выбираются из списка (один, полтора или два стоп-бита), значение по умолчанию – один стоп-бит (One stop bit) (см. п. 3.1.2.4).

«**Тип интерфейса**» (**InterfaceType**) – значения выбираются из списка «**RS-232**» и «**RS-485**», значение по умолчанию – **RS-485** (см. п. 3.1.2.5).

«**Тип протокола обмена**» (**FrameOriented**) – значения выбираются из списка «**ASCII**» и «**RTU-режим**», значение по умолчанию – **ASCII** (см. п. 3.1.2.6).

«**Время, на которое необходимо задерживать ответ на запрос в мс**» (**Framing time**) – рекомендуемый диапазон значений от 0 до 50 мс, значение по умолчанию – 0 (см. п. 3.1.2.7).

«**Видимость**» (**Visibility**) – задает видимость параметров модуля в программе EasyWorkPLC. Значения выбираются из списка «**yes**» и «**no**», значение по умолчанию – «**yes**» (см. п. 2.1.2).

Помимо последовательного порта в конкретной системе устройств пользователь может использовать порт TCP.

### **Параметры порта TCP:**

«**Удаленный порт**» (**RemotePort**) – значения устанавливаются из диапазона от 0 до 65535, значение по умолчанию – 502 (см. п. 3.1.2.8).

«**Видимость**» (**Visibility**) – задает видимость параметров модуля в программе EasyWorkPLC. Значения выбираются из списка «**yes**» и «**no**», значение по умолчанию – «**yes**» (см. п. 2.1.2).

---

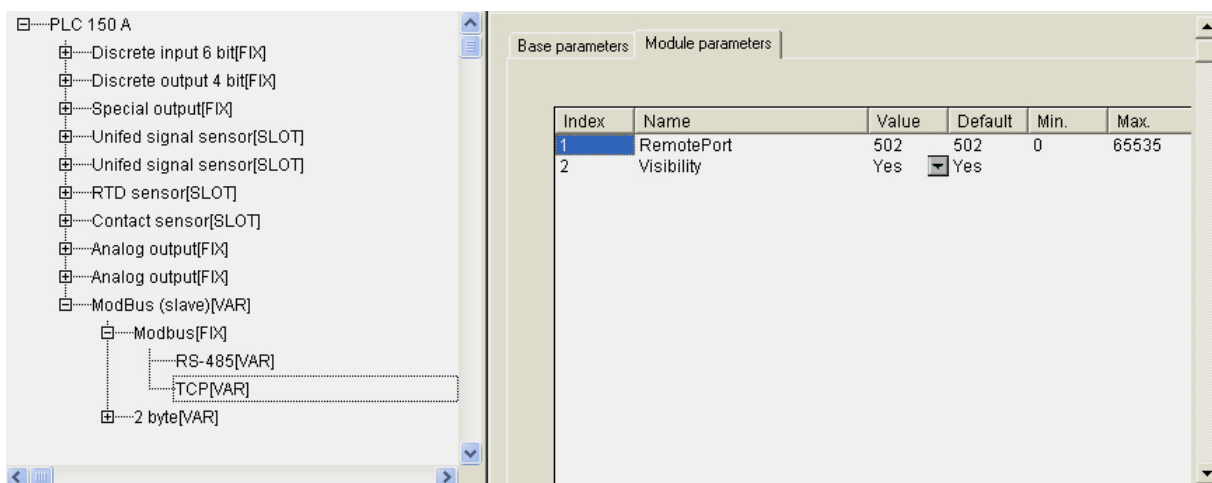
### **Примечание.**

1. IP адрес и MAC адрес указываются для контроллера и задаются на предприятии-изготовителе. При желании пользователя их поменять, это может быть сделано специализированной процедурой (см. **PLC Browser**). В данном случае модуль использует MAC и IP адреса контроллера.

2. Для пользователя никаких дополнительных настроек порта TCP не требуется

---

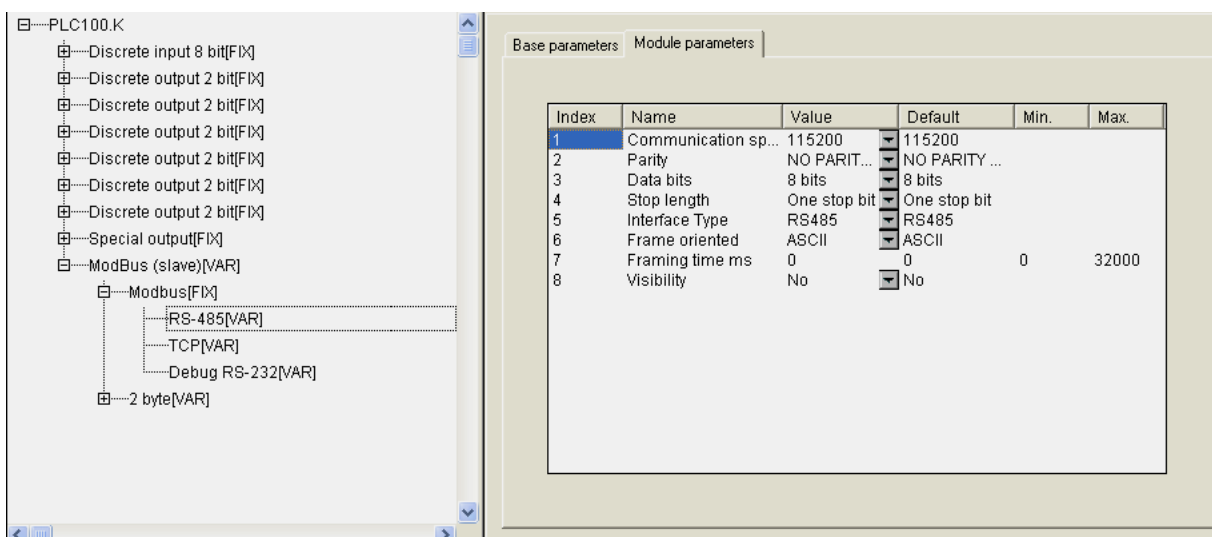
На рис. 3.4. представлена вкладка параметров порта TCP.



**Рис. 3.4. Параметры порта TCP**

При работе ПЛК в режиме «ведомый» имеется возможность использования нескольких разных портов, т.е. опрос может вестись по разным интерфейсам. Таким образом, задавая разные входы, можно один модуль соединить с разными Мастерами по разным физическим линиям (и интерфейсам). Это, как пример, может использоваться для создания межсетевого шлюза и/или линии резервного управления (например, **Scada** подключена в резервном варианте). Можно, например, направлять в одно устройство информацию, полученную по разным линиям, для обмена, – это пассивный вариант работы ПЛК под управлением внешнего Мастера, соответственно, ПЛК отвечает на приходящие запросы.

Пример подключения нескольких портов представлен на рис. 3.5.



**Рис. 3.5. Подключение нескольких портов**

В качестве стандартного варианта можно рассматривать ПЛК, у которого есть один com-порт и один порт Ethernet. При выходе Ethernet из

стройка (авария, сбой и пр.), вышестоящая **Scada** понимает, что произошел сбой, и начинает информационный обмен с этим устройством по резервной линии – через com-порт, с меньшей скоростью, но функционирование устройства сохраняется. Количество подключаемых портов ограничено физическими возможностями самого контроллера.

### **3.1.2.1. Параметр «Скорость передачи информации» (CommunicationSpeed, bod)**

Параметр «Скорость передачи информации» (**CommunicationSpeed, bod**) устанавливает скорость передачи информации через последовательный порт.

### **3.1.2.2. Параметр «Проверка четности» (Parity)**

Параметр «Проверка четности» (**Parity**) определяет наличие бита четности и его значение (четность, нечетность).

### **3.1.2.3. Параметр «Количество бит данных» (DataBits)**

Параметр «Количество бит данных» (**DataBits**) определяет количество бит в одной посылке.

### **3.1.2.4. Параметр «Количество стоп-битов» (StopLength)**

Параметр «Количество стоп-битов» (**StopLength**) определяет количество стоп-битов.

### **3.1.2.5. Параметр «Тип интерфейса» (InterfaceType)**

Параметр «Тип интерфейса» (**InterfaceType**) определяет интерфейс, по которому осуществляется информационный обмен в последовательном порте (**RS-232** или **RS-485**).

### **3.1.2.6. Параметр «Тип протокола обмена» (FrameOriented)**

В ПЛК используются следующие типы протоколов обмена: ориентированный на текстовые символы режим ASCII и ориентированный на поток байтов RTU-режим. Параметр «Тип протокола обмена» (**FrameOriented**) устанавливает используемый для применения ПЛК в конкретной системе режим.

В ASCII информация передается последовательностью читаемых символов, и начало и окончание посылки имеют четко обозначенные специальные символы, обычно это – символы решетки, перевода строки и др.

В RTU иная структура передачи информации: передаются посылочные байты, без какого-либо указания начальных и/или конечных границ (заголовочных и конечных байтов), а сама посылка и ее границы определяются по наличию разрыва. Если время разрыва превышает определенное (достаточно незначительное) время (например, для **Modbus** – 3,5 символа), – устройство определяет, что посылка закончилась, началась другая посылка. Таким образом, посылки разделяются друг от друга и их можно идентифицировать.

### **3.1.2.7. Параметр «Время, на которое необходимо задерживать ответ на запрос в мс» (Framing time)**

Параметр «**Время, на которое необходимо задерживать ответ на запрос в мс**» (**Framing time**) определяет временную задержку между последним байтом принятого пакета и первым байтом, передаваемым в ответ. Задержка бывает необходима для работы с медленными устройствами.

### **3.1.2.8. Параметр «Удаленный порт» (RemotePort)**

Параметр «**Удаленный порт**» (**RemotePort**) устанавливает адрес удаленного порта.

## **3.1.3. Настройка данных информационного обмена**

Выше были рассмотрены вопросы, связанные с настройкой коммуникаций модуля **Modbus (Slave)**. В этом разделе рассматривается механизм распределения адресов ячеек памяти внутри **Modbus (Slave)**.

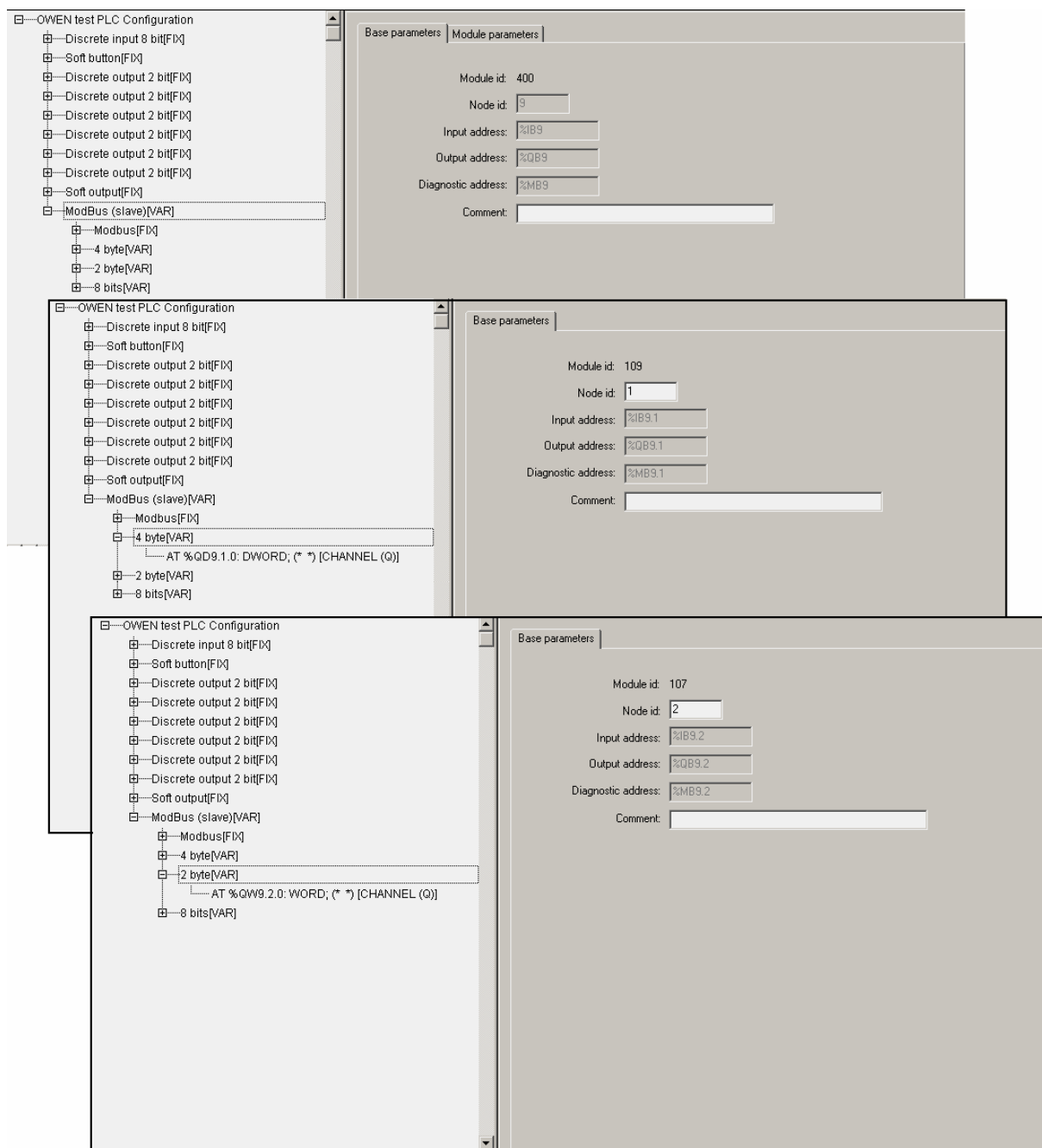
Протокол **Modbus** рассматривает каждое устройство, включая и модуль **Modbus (Slave)**, как память, набор неких конкретных регистров.

---

**Примечание.** При рассмотрении информационного обмена, областей памяти и т.п. используются категории «бит» и «регистр» (численно равный 2 байтам). Считается, что эти два адресных пространства независимы между собой. В некоторых приборах они учитываются раздельно, в других приборах – это общая память, но она адресуется по-разному. Биты адресуются от 0 до n побитно, байты – от 0 до n побайтно, а регистры – также от 0 до n, но по регистрам (т.е. по 2 байта).

---

В **Modbus (Slave)** есть возможность вставить следующие 4 типа элементов (переменных): REAL, 4 байта, 2 байта или 8 бит (см. рис. 3.6).



**Рис. 3.6. Подключение переменных**

Добавление переменных различных размеров требуется для структурирования памяти. С точки зрения Мастера **slave**-устройство представляет собой непрерывно организованную область памяти, которую можно адресовать как по регистрам (с 0-го регистра), так и побитно.

Обращение к внутренней памяти **Modbus (Slave)** происходит следующим образом. При запросе внешнего Мастера, например, «регистра номер 0», устройство считывает первые два байта первой переменной, для «регистра номер 1» – вторые два байта первой переменной (если та имеет длину более 2-х байт) или первые два байта второй переменной, и т.д.

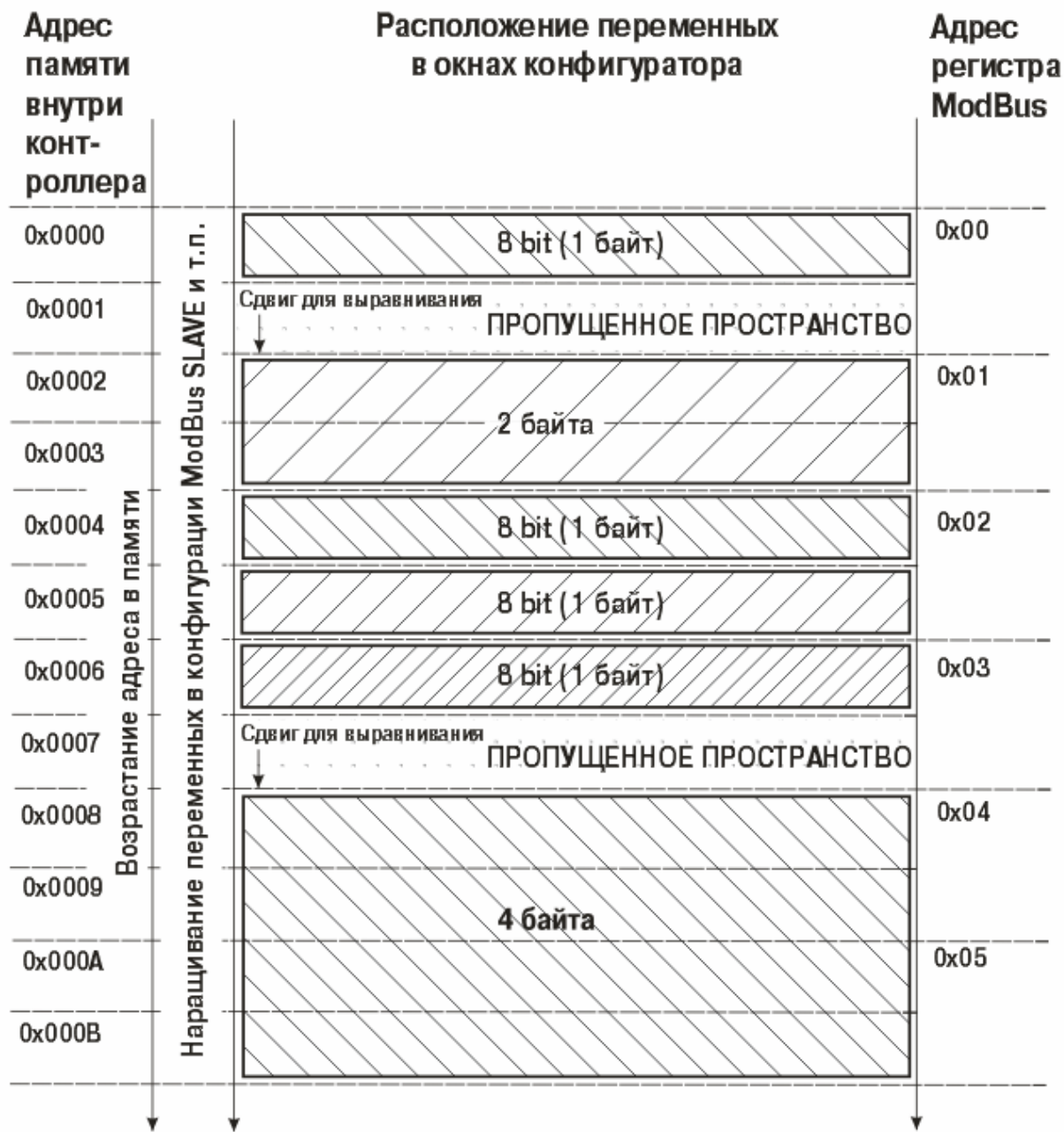
Особенность организации обращений заключается в том, что, если в устройстве установлен (используется) смешанный тип переменных (т.е., одновременно и 2-байтный, и 8-битный, и 4-байтный), то необходимо на всех этапах работы с устройством учитывать особенности выравнивания переменных в области ввода/вывода.

### **3.1.3.1. Выравнивание переменных при размещении в области памяти**

Суть упорядоченного размещения переменных в области памяти или «**выравнивания (align)**», принятого в среде CoDeSys, заключается в организации физической памяти таким образом, что переменные 2-байта, 8-бит и 4-байта должны располагаться только по определенным адресам. Адрес 4-байтной переменной должен быть кратен 4, 2-байтной – кратен 2, а однобайтной (или 8 бит) – кратен 1 и может находиться в любой точке пространства памяти.

Если представить область памяти с возрастающими адресами (от 0 до какого-либо числа) и расположить переменные, то, если идет первая переменная типа «байт», то она будет расположена по адресу 0x00, следующая – 0x01 и т.д. Если дальше идет 4-байтная переменная, она должна располагаться по адресу 0x04, т.е., кратному 4, и т.д. При этом, если однобайтная переменная заняла кратное 4 место, следующая 4-байтная переменная занимает следующее кратное 4 место. Порядок задания переменных может быть произвольным, выравнивание же ставит переменные на кратные их длине адреса. Соответственно, при таком порядке размещения переменных, неизбежно и естественным образом возникают пустые пространства, которые нигде не отображаются, не видны в области ввода/вывода, но в обязательном порядке должны учитываться пользователем: когда производится опрос прибора извне для получения информации, размещенной по конкретному адресу (регистру), пользователь должен учитывать особенность выравнивания, чтобы не получить некорректную информацию, причем должен учитывать еще на стадии задания переменных.

Выравнивание при размещении переменных в области памяти проиллюстрировано на рис. 3.7.



1 байтовая переменная располагается по любому адресу  
 2 байтовая переменная выравнивается по четным адресам

4 байтовые и больше переменные выравниваются по адресу, кратному 4

**Рис. 3.7. Выравнивание переменных при размещении в области памяти**

### 3.1.3.2. Отображение устройства в области памяти

Среди переменных, к которым при работе ПЛК осуществляется обращение, например, внешнего **Мастера**, может быть вставлен и другой модуль, который также отображается, как память. Это важный момент именно для **Modbus (Slave)**, который отображается как память, т.к. во внутреннее пространство этой памяти можно вставить не только переменные, но и устройство, отображаемое как память. Этим вставляемым устройством может быть аналогичный модуль **Modbus (Slave)**, который будет подключен уже к другой физической линии, **Мастер**, опрашивающий подчиненный **Modbus (Slave)**, например, подключенный к другому физическому интерфейсу и имеющий доступ к ограниченному сегменту от общих данных, или другой модуль (список доступных для добавления модулей отображается при попытке вставить новую переменную). За счет подобной организации области памяти могут быть реализованы:

- шлюз для обмена данными;
- сокрытие определенных данных от конкретного пользователя путем предоставления доступа к части переменных и осуществления контроля через внешний **Мастер**, размещенный на компьютере; и др.

Подключение устройств в области памяти продемонстрировано на рис. 3.8.

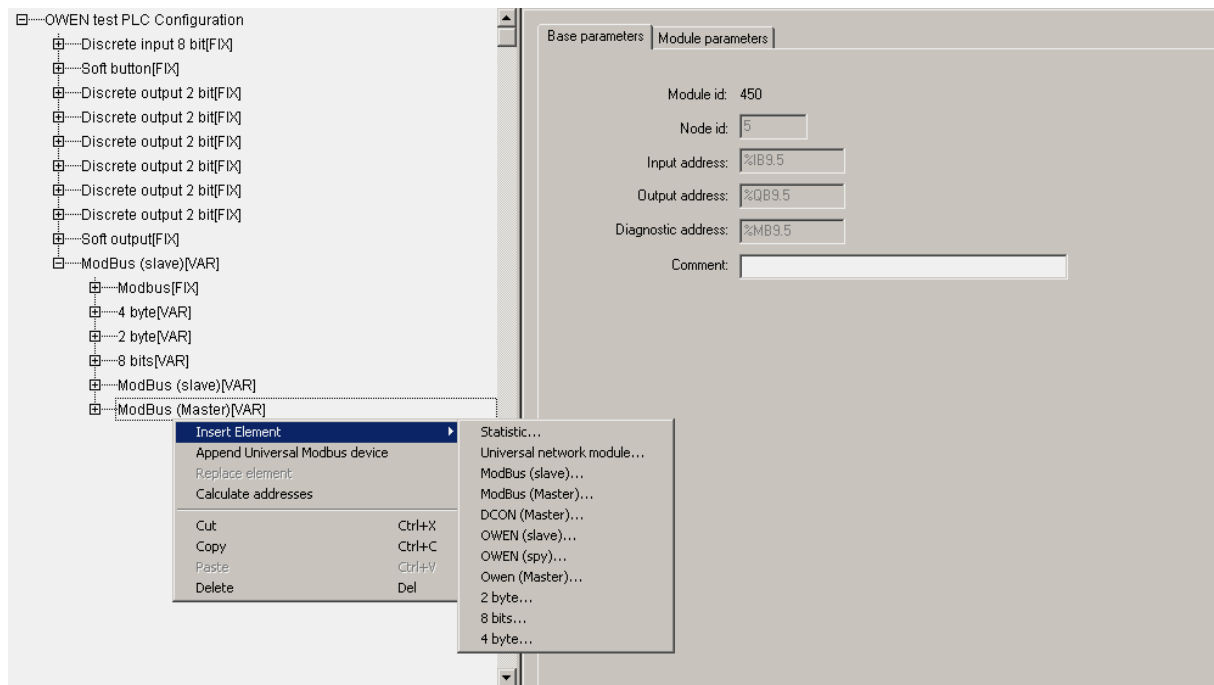


Рис. 3.8. Подключение устройств в области памяти

Этот механизм позволяет с помощью внешней **SCADA**, имея один заголовочный модуль **Modbus (Slave)**, подключить все внутренние модули как подчиненные, соответственно, **SCADA** естественным путем получает доступ ко всем их данным. В этом случае, нет необходимости реализовывать в



программе копирование данных из одного протокола в другой (например, **Овена (Мастера) в Modbus (Slave)**), чтобы затем **SCADA** могла их считать. В описанном случае **SCADA** имеет прямой доступ к информации других устройств, которые подключены и идентифицируются как области памяти, к которым можно обращаться, считывать и пр. Кроме того, реализуется еще и возможность управления: в обратную сторону: **SCADA** может записать данные, которые определенным образом меняют работу контролируемого устройства.

## 3.2. Модуль Modbus (Master)

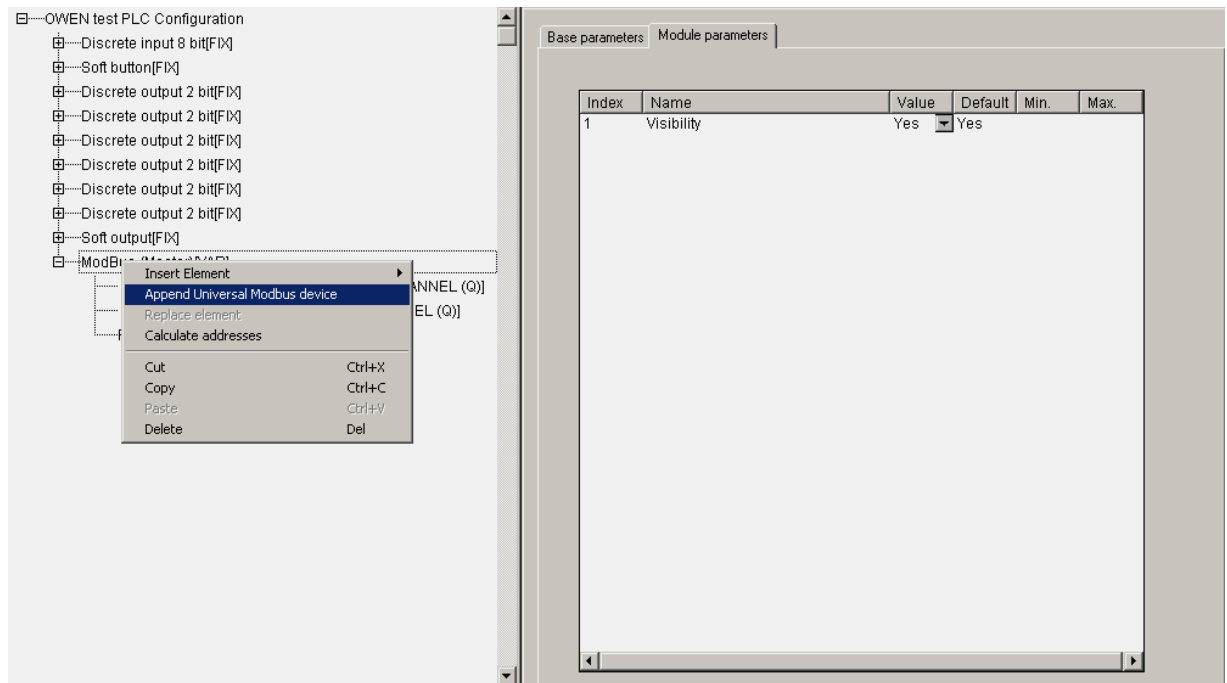
В данном разделе рассматривается работа ПЛК по протоколу **Modbus** в режиме «**ведущего**», т.е. режиме «**Master**».

ПЛК, когда у него установлен модуль **Modbus (Master)**, будет по выбранному коммуникационному интерфейсу работать активно – обмениваться данными с другими устройствами, контролировать работу других устройств и т.д.

### Параметры модуля:

«**Видимость**» (**Visibility**) – задает видимость параметров модуля в программе EasyWorkPLC. Значения выбираются из списка «**yes**» и «**no**», значение по умолчанию – «**yes**» (см. п. 2.1.2).

Модуль **Modbus (Master)** представлен на рис. 3.9.



**Рис. 3.9. Модуль Modbus (Master)**

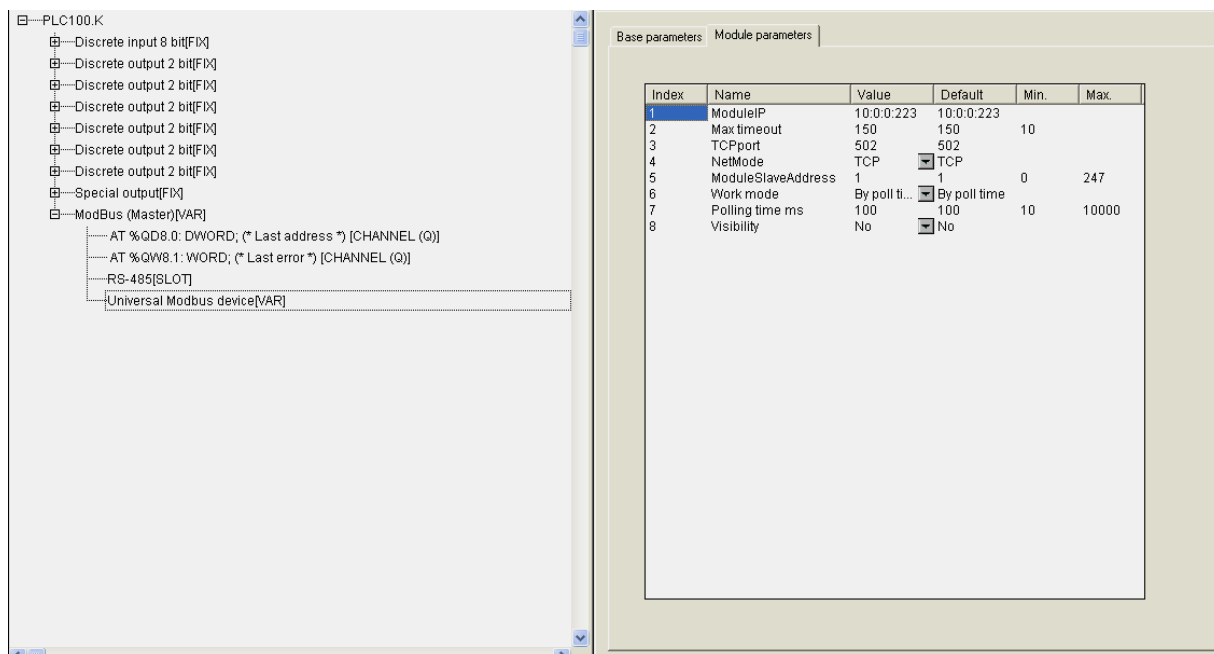
Настройка коммуникационных интерфейсов **Modbus (Master)** такая же, как и описанная выше для **Modbus (Slave)**. Особенностью **Мастера** является следующее: в модуле уже подключен один из интерфейсов последовательных портов, его можно только заменить на аналогичный. Таким образом, если **Мастер** предполагает работать с конкретным устройством по протоколу TCP, то необходимая настройка производится в модуле устройства.

Установка и настройка устройств, опрашиваемых модулем, производится через установку «универсального устройства **Modbus (Master)**» и задание необходимых параметров. Его подключение

выполняется командой «**Append Universal Modbus Device**», вызываемой из контекстного меню.

### 3.2.1. Универсальное устройство Modbus (Master)

Универсальное устройство **Modbus (Master)** представлено на рис. 3.10.



**Рис. 3.10. Универсальное устройство Modbus (Master)**

При вызове универсального устройства появляется шаблон, не заполненный, параметры и идентификаторы которого не привязаны к конкретному модулю. При заполнении полей параметров значениями, определении входов-выходов, устройство определит конкретный вариант вставляемого устройства.

#### **Параметры универсального устройства Modbus (Master):**

«**IP-адрес (Module IP)**» – устанавливается IP-адрес подчиненного устройства (см. п. 3.2.1.1).

«**Максимальный тайм-аут, в мс (Max timeout)**» – значение сверху не ограничено, может быть любым, в т.ч. с плавающей точкой, но не меньше 10 мс, значение по умолчанию – 150 (см. п. 3.2.1.2).

«**Порт TCP (TCP port)**» – устанавливается значение по умолчанию – 502 (см. п. 3.2.1.3).

---

**Внимание.** Значение параметра может быть установлено и другое, при необходимости.

---

«**Режим работы в сети**» (**NetMode**) – значения выбираются из списка «**TCP**» и «**Serial**», значение по умолчанию – «**TCP**» (см. п. 3.2.1.4).

«**Адрес подчиненного устройства**» (**ModuleSlaveAddress**) – диапазон значений от 1 до 247, значение по умолчанию – 1 (см. п. 3.2.1.5).

«**Видимость**» (**Visibility**) – задает видимость параметров модуля в программе EasyWorkPLC. Значения выбираются из списка «**yes**» и «**no**», значение по умолчанию – «**yes**» (см. п. 2.1.2).

«**Период опроса устройства, в мс**» (**Polling time**) – диапазон значений от 10 до 10000, значение по умолчанию – 100 (см. п. 3.2.1.6).

---

**Внимание.** На вкладке представлены параметры, относящиеся как только к режиму TCP, так и параметры, относящиеся как к режиму TCP, так и режиму последовательного интерфейса.

---

### ***3.2.1.1. Параметр «IP-адрес» (Module IP)***

Параметр «**IP-адрес**» (**Module IP**) устанавливает IP-адрес для подчиненного устройства, которым управляет **Мастер**, если оно будет работать в режиме TCP.

### ***3.2.1.2. Параметр «Максимальный тайм-аут, в мс» (Max timeout)***

Параметр «**Максимальный тайм-аут, в мс**» (**Max timeout**) представляет собой максимально время, за которое устройство должно ответить на запрос. Если оно за это время не отвечает, то Мастер считает, что с устройством произошел сбой или авария, и начинается опрос другого устройства. Информация о сбое фиксируется в переменной модуля **Last error**.

### ***3.2.1.3. Параметр «Порт TCP» (TCP port)***

Стандартным значением параметра «**Порт TCP**» (**TCP port**) для протокола **Modbus TCP** является 502, но может быть установлено, при необходимости, и другое значение.

### ***3.2.1.4. Параметр «Режим работы в сети» (NetMode)***

Параметр «**Режим работы в сети**» (**NetMode**) имеет 2 варианта значений – TCP и Serial:

- **TCP** – подчиненное устройство, работает по протоколу TCP.

- **Serial** – устройства, осуществляющие функционирование по последовательному интерфейсу.

---

При задании значения **Serial** обмен ведется через тот последовательный интерфейс, который подключен к модулю Мастера. В случае задания значения **TCP** используется интерфейс **Ethernet**, опрашиваемый модуль идентифицируется по значению IP-адреса.

---

### **3.2.1.5. Параметр «Адрес подчиненного устройства» (ModuleSlaveAddress)**

Параметр **«Адрес подчиненного устройства» (ModuleSlaveAddress)** задает адрес подчиненного устройства. Значения лежат в диапазоне от 1 до 247. Значение 0 специфично и используется для широковещательных сообщений. Например, при работе через шлюз.

### **3.2.1.6. Параметр «Период опроса устройства, в мс» (Polling time)**

Параметр **«Период опроса устройства, в мс» (Polling time)** – задает период в мс, с которым будет опрашиваться устройство.

## **3.2.2. Настройка входов и выходов модуля Modbus (Master)**

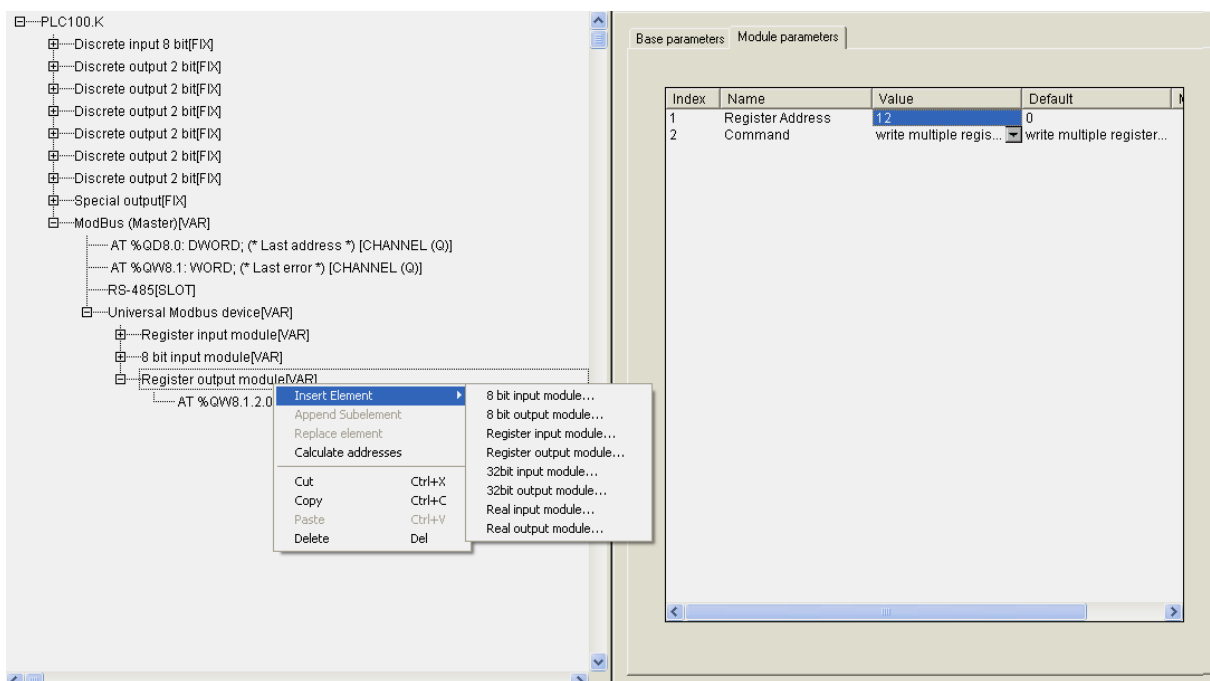
После задания параметров в рамках выстраиваемого модуля настраиваются входы и выходы, которые будут опрашиваться на этом устройстве. Могут быть добавлены четыре типа входов/выходов. В соответствии с протоколом – это могут быть 8-, 16-, 32-битные данные, либо числа с плавающей точкой.

### **Параметры входов и выходов модуля Modbus (Master):**

**«Адрес регистра опрашиваемого устройства» (Register Address)** – устанавливается адрес регистра опрашиваемого устройства (см. п. 3.2.2.1).

**«Номер команды протокола» (Modbus Command)** – значения выбираются из списка возможных вариантов, значение по умолчанию – **«write multiple registers (0x10)»** (см. п. 3.2.2.2).

Рис. 3.11 иллюстрирует процесс настройки входов и выходов универсального устройства.



**Рис. 3.11. Настройка входов и выходов универсального устройства**

При опросе модулем **Modbus (Master)** подчиненных устройств информация о ходе обмена индицируется на его переменных:

**LastAddress** – последний адрес, по которому обращался **Modbus (Master)**. Модуль запрашивает устройство, и, соответственно, тут же меняется значение: показывается значение адреса последнего запроса и код ошибки (**LastError**). Это необходимо для определения, нормально ли работает опрашиваемое устройство.

Сообщения об ошибках в ПЛК представлены в Приложении В.

### **3.2.2.1. Параметр «Адрес регистра опрашиваемого устройства» (Register Address)**

Параметр «**Адрес регистра опрашиваемого устройства**» (**Register Address**) задает адрес регистра опрашиваемого устройства

### **3.2.2.2. Параметр «Номер команды протокола» (Modbus Command)**

Параметр «**Номер команды протокола**» (**Modbus Command**) задает номер команды протокола, по которой будет производиться обмен, т.к. некоторые устройства не поддерживают все команды протокола ModBus.

### 3.3. Модуль «Архиватор» (Archiver)

«Архиватор» – специализированный модуль, призванный облегчить программисту, пишущему программу, архивацию каких-либо необходимых данных. Архивируемые данные могут храниться на **Flash** диске ПЛК или выводиться через коммутационный интерфейс. Например, к ПЛК может быть подключен принтер, работающий по последовательному интерфейсу, и задана распечатка определенных значений. При подключении модуля, у него сразу присутствует один из интерфейсных портов, в который будут выдаваться архивные данные. Пользователю требуется заменить интерфейсный порт на нужный, воспользовавшись командой контекстного меню **Replace element**.

#### Параметры модуля:

«Режим проведения архивация» (**Archive Mode**) – значения выбираются из списка «**By timer**», «**By change value**» и «**By command**», значение по умолчанию – «**By timer**» (см. п. 3.3.1).

«Тип данных архивации» (**Data Archive Type**) – значения выбираются из списка «**ASCII only**» и «**Mixed**», значение по умолчанию – «**ASCII only**» (см. п. 3.3.2).

«Период архивации, сек» (**Period of Archiving**) – диапазон значений от 5 до 65535 сек, значение по умолчанию – 60 (см. п. 3.3.3).

«Имя архива» (**Archive Name**) – устанавливается имя архива (см. п. 3.3.4).

«Комментарий архива» (**Comment**) – вводится комментарий архива в текстовом виде (см. п. 3.3.5).

«Время начала архивации» (**Start time**) – устанавливается время старта архивации (см. п. 3.3.6).

«Время остановки архивации» (**Stop time**) – устанавливается время остановки архивации (см. п. 3.3.6).

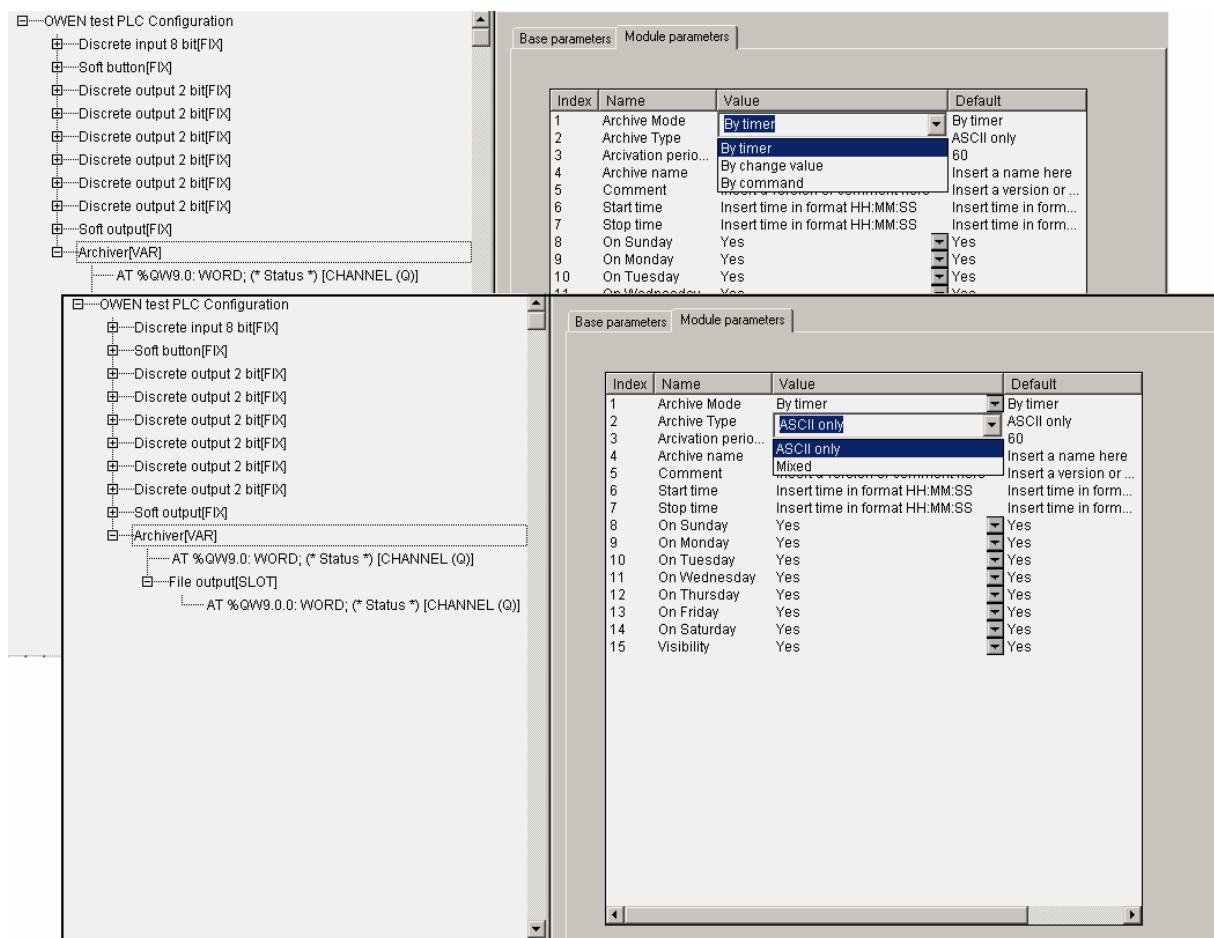
«Понедельник» (**On Sunday**) ... «Воскресенье» (**On Saturday**) – всего 7 параметров – устанавливается день (дни) недели, в который (которые) будет производиться архивация. Значения выбираются из списка «**yes**» и «**no**», значение по умолчанию – «**yes**» (см. п. 3.3.7).

«Видимость» (**Visibility**) – задает видимость параметров модуля в программе EasyWorkPLC. Значения выбираются из списка «**yes**» и «**no**», значение по умолчанию – «**yes**» (см. п. 2.1.2).

#### Список подмодулей:

Подмодуль архивации информации в файл (**File Output**) (см. п. 3.3.8).

Параметры архиватора представлены на рис. 3.12.



**Рис. 3.12. Параметры архиватора**

Внесение переменных в список для последующего архивирования осуществляется по стандартной технологии – иницированием опции **Append Subelement** в контекстном меню, вызываемом нажатием правой клавиши манипулятора «мышь», при выделении курсором на экране монитора модуля «**Archiver**». Могут быть добавлены переменные байтовая, двухбайтовая, 4-байтовая, число с плавающей точкой и текстовая строка (максимум 15 символов+завершающий ноль). При добавлении любой переменной в ее параметрах вводится лишь имя – **Variable Name**, – которое заносится в архивный файл для удобства работы.

В модуле «**Архиватор**» имеется переменная «**File Status**» для отображения статуса и возникающих ошибок. Сообщения об ошибках в ПЛК представлены в Приложении В.

Модуль «**Архиватор**» и организация архивации в необходимый интерфейс представлены на рис. 3.13. Настройка последовательных интерфейсов и интерфейса TCP стандартная и описана в п. 3.2.



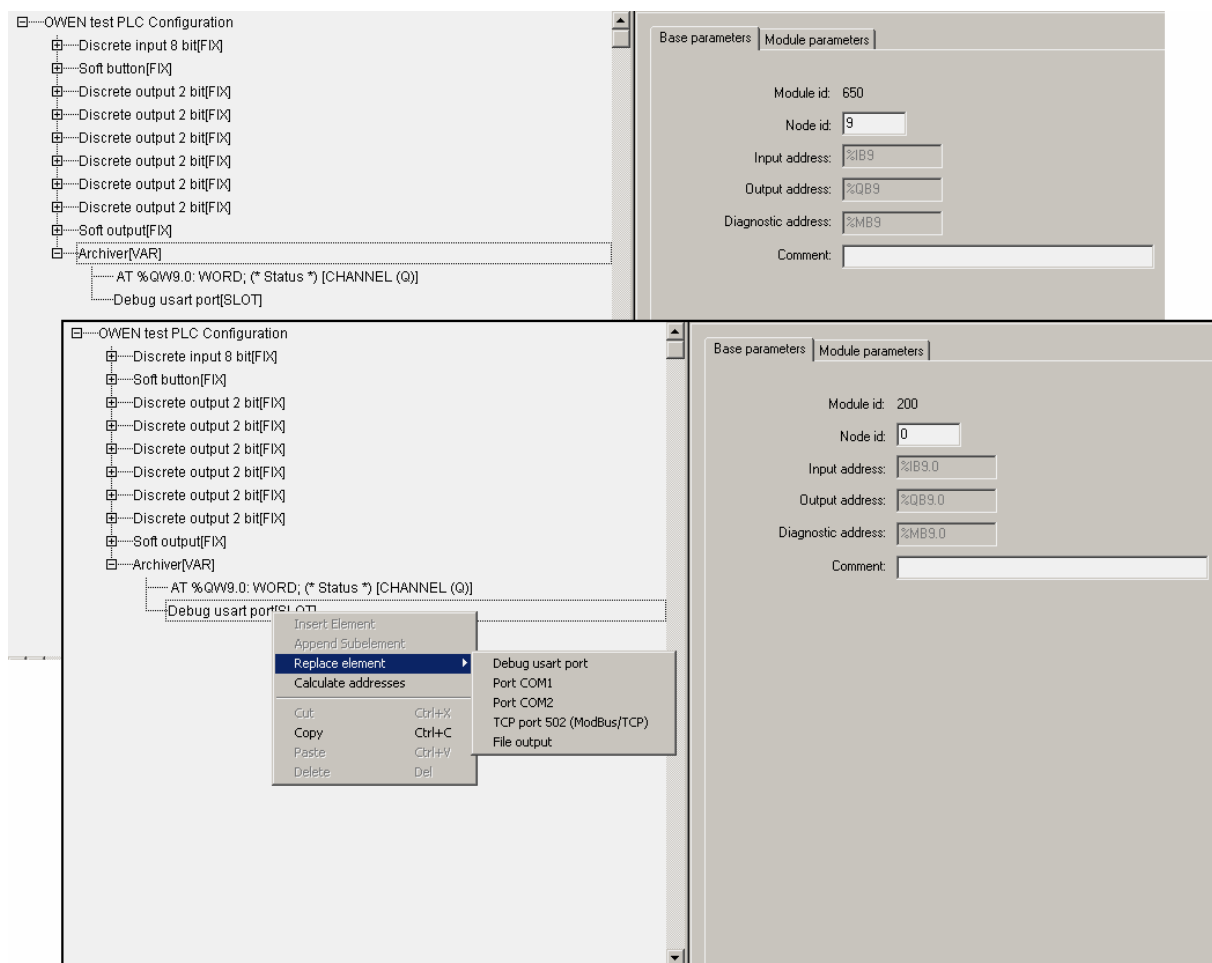


Рис. 3.13. Выбор интерфейса архивации

### 3.3.1. Параметр «Режим проведения архивация» (Archive Mode)

«Режим проведения архивация» (Archive Mode) – определяет выбор режима архивации из 3-х возможных вариантов:

- **By timer** – «по таймеру» – данные записываются в архив с заданным периодом архивации;
- **By change value** – «по изменению значений»: если какая-то переменная, включенная в список архивации, меняет свое значение, то происходит ее архивация, причем именно этой переменной, а не всех. Изменения могут записываться не чаще, чем раз в секунду;
- **By command** – «по команде»: если в переменной **Status** модуля архивации записана специальная команда, то происходит либо старт архивации, либо ее остановка (**0x00FE** – «стоп», а **0x00FF** – «старт»).

### 3.3.2. Параметр «Тип данных архивации» (Data Archive Type)

«Тип данных архивации» (Data Archive Type) – определяет выбор типа записи архивируемых данных из 2-х возможных вариантов:

- **ASCII only** – данные выдаются в текстовом виде, удобном для чтения пользователя, для печати и т.п.;
- **Mixed** – данные выдаются в смешанном виде: запись имеет заголовок архива с надписями названий переменных, архива, временных данных в удобном для чтения виде, а все изменения архива регистрируются в бинарном виде, запакованном для уменьшения занимаемого места.

### 3.3.3. Параметр «Период архивации, сек» (Period of Archiving)

«Период архивации, сек» (Period of Archiving) – задает периодичность обновления архива при работе модуля в режиме «по таймеру».

### 3.3.4. Параметр «Имя архива» (Archive Name)

«Имя архива» (Archive Name) – задает имя архива, которое записывается в начале файла.

### 3.3.5. Параметр «Комментарий архива» (Comment)

«Комментарий архива» (Comment) – задает комментарий архива в текстовом виде. Может быть введена информация, позволяющая пользователю в последующем идентифицировать конкретный архив по дополнительным признакам.

### 3.3.6. Параметры «Время начала архивации» (Start time) и «Время остановки архивации» (Stop time)

«Время начала архивации» (Start time) – задает время старта архивации.

«Время остановки архивации» (Stop time) – определяет время остановки.

---

**Внимание.** Два описываемых параметра, задающих временные рамки работы, независимы друг от друга, т.е. одного или обоих из них может не быть. Для параметров определен формат, в котором они должны задаваться – **ЧЧ:ММ:СС**, – с обязательным использованием полноформатного задания величины и разделителя «двоеточие». При

неполном формате и/или использовании иного разделителя программа проигнорирует информацию, как ошибочную.

---

### 3.3.7. Параметры «Понедельник» (On Sunday) ... «Воскресенье» (On Saturday)

Параметры от **On Sunday** до **On Saturday** – всего 7 параметров – определяют день недели, в который будет производиться архивация.

---

**Внимание.** Установлен определенный приоритет между всеми модификаторами времени. Главный приоритет имеет команда: если в переменной **File Status** записана команда «**стоп**», то операция в любом случае прекратится, если «**старт**» – она будет выполнена минимум 1 раз. Затем по приоритету идет день недели и, далее, время старта и время останова.

---

### 3.3.8. Подмодуль архивации информации в файл (File Output)

Подмодуль архивации информации в файл (**File Output**) – программный модуль, задающий параметры архивации информации в файл.

Модуль «**File Output**» является подчиненным подмодулем модуля «**Архиватор**».

Для подмодуля архивации информации в файл дополнительно в поле «**File Status**» выводится информация о работе подмодуля. Сообщения об ошибках в ПЛК представлены в Приложении В.

#### Параметры модуля:

«**Имя файла**» (**File name**) – устанавливается имя файла, значение по умолчанию – «**File\_name.log**» (см. п. 3.3.8.1).

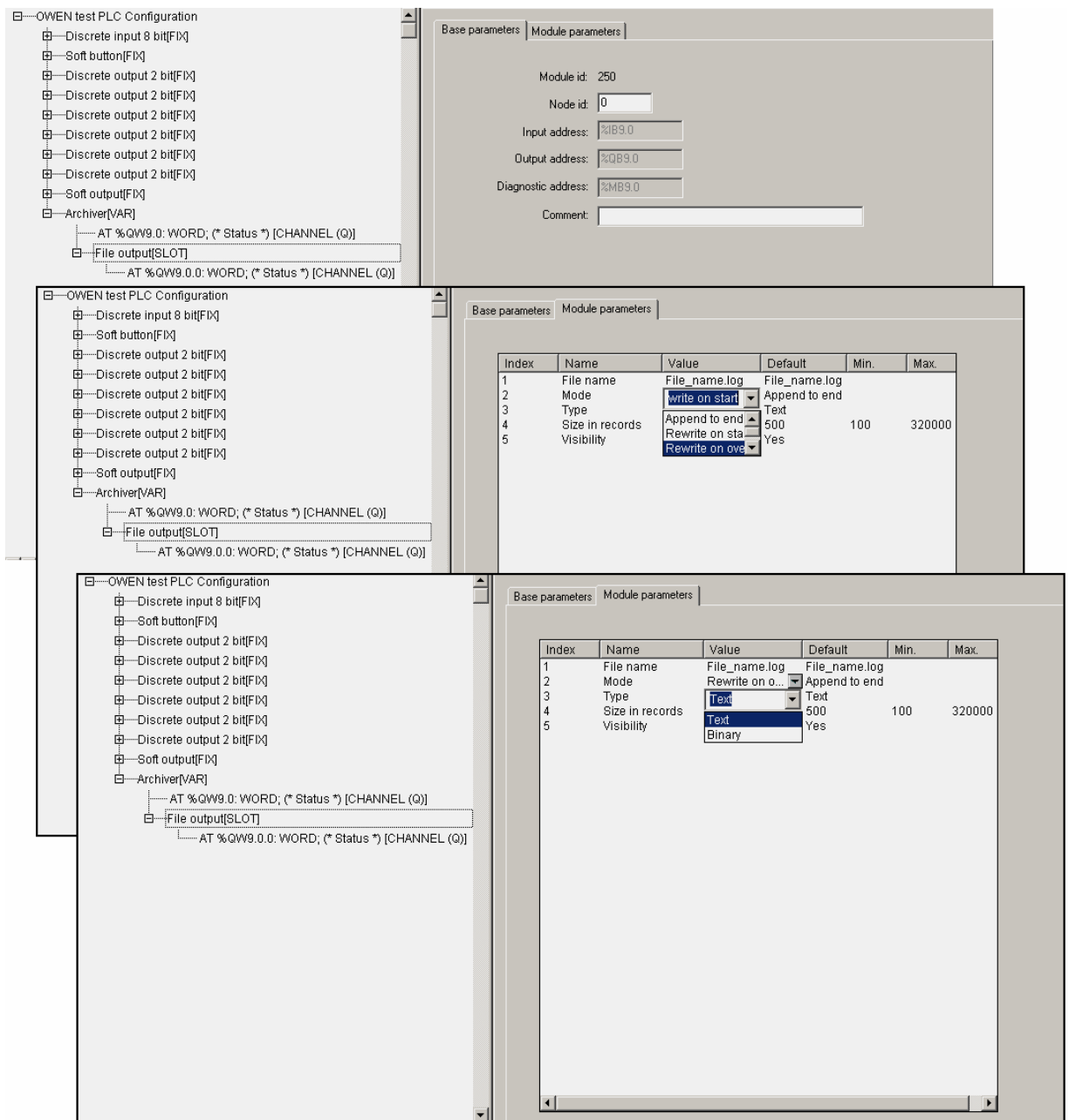
«**Режим работы модуля**» (**Mode**) – значения выбираются из списка «**Append to end**», «**Rewrite on start**», «**Rewrite on oversize**» и «**Shift Mode**», значение по умолчанию – «**Append to end**» (см. п. 3.3.8.2).

«**Тип**» (**Type**) – значения выбираются из списка «**Text**» и «**Binary**», значение по умолчанию – «**Text**» (см. п. 3.3.8.3).

«**Размер записи**» (**Max file size**) – диапазон значений от 100 до 320000, значение по умолчанию – 500 (см. п. 3.3.8.4).

«**Видимость**» (**Visibility**) – задает видимость параметров модуля в программе EasyWorkPLC. Значения выбираются из списка «**yes**» и «**no**», значение по умолчанию – «**yes**» (см. п. 2.1.2).

Параметры подмодуля архивации необходимой информации в файл представлены на рис. 3.14.



**Рис. 3.14. Параметры подмодуля архивации  
необходимой информации в файл**

### **3.3.8.1. Параметр «Имя файла» (File name)**

Параметр **«Имя файла» (File name)** устанавливает имя файла, в который будет писаться архивная информация.

### **3.3.8.2. Параметр «Режим работы модуля» (Mode)**

Параметр **«Режим работы модуля» (Mode)** определяет вариант архивации информации. При установке значения осуществляется выбор из 4-х возможных вариантов:

- **Append to end** – «**добавить в конец**» – вариант работы, при котором информация добавляется в конец файла, и, как только файл переполняется, запись прекращается. Файл имеет ограничение либо по байтам, либо по записям;
- **Rewrite on start** – «**перезапись при старте**» – вариант работы, при котором при старте ПЛК или загрузке новой конфигурации производится стирание старого файла и начинается запись файла с самого начала;
- **Rewrite on oversize** – «**перезапись старого файла при превышении заданного размера**» – вариант работы, при котором при достижении файлом заданного размера, он стирается, и запись начинается сначала;
- **Shift Mode** – «**режим сдвига**» – вариант работы, при котором, при достижении файлом заданного размера, вторая (более поздняя по времени записи) половина файла переносится в начало, запись продолжается, дописывается, т.е. остаются самые последние записи.

### **3.3.8.3. Параметр «Тип» (Type)**

Значение параметра **«Тип» (Type)** определяет, каким образом при работе модуля будет подсчитываться размер файла:

- при текстовом режиме (**Text**) подсчет будет осуществляться по записям, при этом каждая запись заканчивается символом перевода каретки;
- при цифровом (двоичном) режиме (**Binary**) подсчет будет осуществляться по размеру файла в байтах.

### **3.3.8.4. Параметр «Размер записи» (Max file size)**

Параметр **«Размер записи» (Max file size)** устанавливает ограничение на размер записываемого файла, при этом размер определяется в зависимости от типа: при текстовом – количество записей, при бинарном – количество байт.

### 3.4. Модуль «Owen (Slave)»

Модуль **Owen (Slave)** обеспечивает обмен информацией по протоколу ОВЕН.

Как и в **Modbus (Slave)**, в модуле **Owen (Slave)** реализована возможность установить коммуникационный интерфейс, по которому осуществляется информационный обмен.

#### **Параметры модуля:**

«**Имя прибора**» (**Slave Name**) – устанавливается имя ПЛК в сети ОВЕН, значение по умолчанию – «**max 8 sym**» (см. п. 3.4.1).

«**Длина адреса устройства**» (**Address Length**) – значения выбираются из списка «**8 bit**» и «**11 bit**», значение по умолчанию – «**8 bit**» (см. п. 3.4.2).

«**Адрес устройства**» (**Address**) – диапазон значений от 0 до 255 или от 0 до 2048, в зависимости от размера адреса, значение по умолчанию – 1 (см. п. 3.4.3).

«**Видимость**» (**Visibility**) – задает видимость параметров модуля в программе EasyWorkPLC. Значения выбираются из списка «**yes**» и «**no**», значение по умолчанию – «**yes**» (см. п. 2.1.2).

#### **Список подмодулей:**

**Owen (FIX)** (см. п. 3.4.4).

В одном ПЛК может быть вставлено несколько модулей **Owen (Slave)**. В этом случае они представляются в сети как несколько приборов с разными адресами и именами.

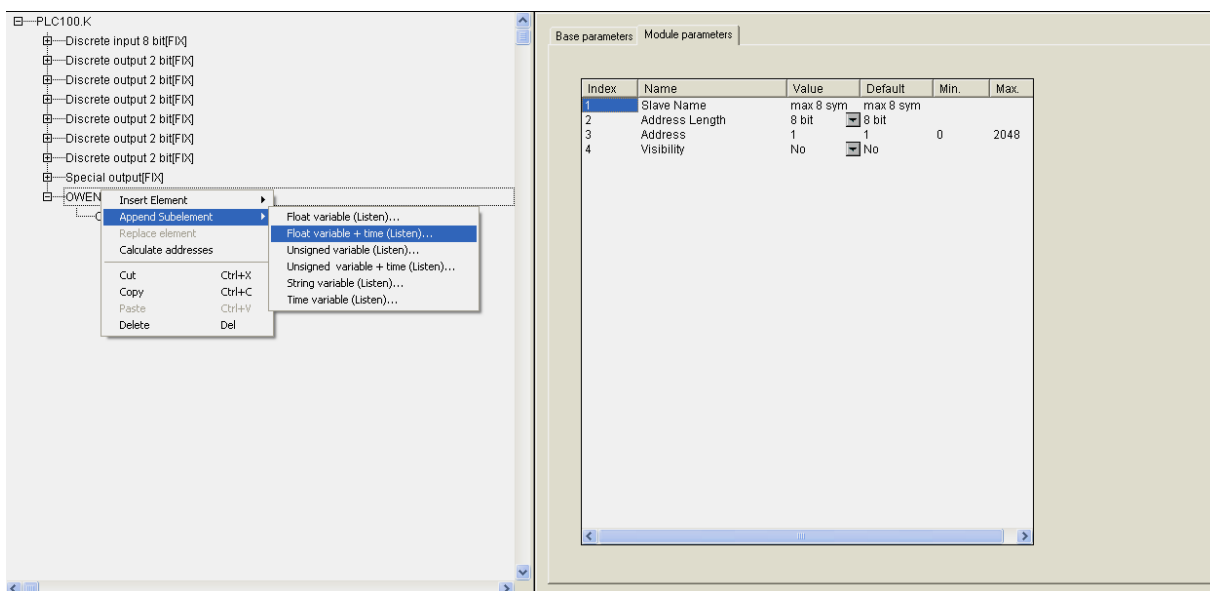
Переменные, которыми будет обмениваться ПЛК по протоколу ОВЕН, выбираются пользователем функцией контекстного меню **Append Subelements**.

---

**Внимание.** При пропадании питания в процессе работы ПЛК последние (текущие) значения переменных сохраняются в энергонезависимой памяти и восстанавливаются при возобновлении работы прибора.

---

Модуль **Owen (Slave)** представлен на рис. 3.15.



**Рис. 3.15. Модуль Owen (Slave)**

### 3.4.1. Параметр «Имя прибора» (Slave Name)

Параметр **«Имя прибора» (Slave Name)** задает имя ПЛК в сети ОВЕН. Имя прибора можно считать, послав запрос значения стандартного параметра **dev**.

### 3.4.2. Параметр «Длина адреса устройства» (Address Length)

Параметр **«Длина адреса устройства» (Address Length)** задает длину адреса в битах. Приборы разработки ПО «Овен» поддерживают два варианта адресации – 8 и 11 битную (на приборе выставляется тот же вариант адресации, что выставлен на управляющей стороне).

### 3.4.3. Параметр «Адрес устройства» (Address)

Значение параметра **«Адрес устройства» (Address)** – это адрес прибора, по которому посылаются запросы. Параметр имеет значения в диапазоне от 0 до 2048.

---

**Внимание.** Параметр идентичен одноименному параметру модуля **Modbus (Slave)**, за исключением диапазона возможных значений.

---

### 3.4.4. Подмодуль Owen (FIX). Настройка коммуникаций модуля

В состав модуля **Owen (Slave)** фиксировано вставлен подмодуль **Owen (FIX)** – внутренний модуль, создающий определенную структуру, к которой уже подключается коммутационный интерфейс.

---

**Внимание.** Параметры настройки интерфейса идентичны параметрам настройки подмодуля **Modbus (FIX)** (см. п. 3.1.2.). Особенности следующие: параметр «**Тип протокола обмена**» (**FrameOriented**) должен быть выставлен в протоколе ASCII, а остальные – по выбору пользователя, в зависимости от предполагаемых условий работы с прибором.

---

### 3.4.5. Переменные протокола OVEN

В информационном обмене ПЛК по протоколу OVEN используются переменные, различающиеся по типу данных (всего – 8 возможных вариантов):

- число с плавающей точкой – **Float variable (Listen)**,
- число с плавающей точкой с модификатором времени – **Float variable + time (Listen)**,
- целочисленная переменная – **Unsigned variable (Listen)**,
- целочисленная переменная с модификатором времени – **Unsigned variable + time (Listen)**,
- строка – **String variable (Listen)**,
- строка с модификатором времени – **String variable + time (Listen)**,
- время в специальном формате – **Time variable (Listen)**,
- время с модификатором времени – **Time variable + time (Listen)**.

---

**Примечание.** Характеристика **Listen** напоминает пользователю о том, что прибор пассивен, ждет запросов («слушает») и отвечает на них.

---

#### Параметры переменных протокола OVEN:

«**Имя переменной**» (**Hash name**) – вводится имя переменной (см. п. 3.4.5.1).

«**Индекс прибора**» (**Index**) – диапазон значений от 0 до 65535, значение по умолчанию – 0 (см. п. 3.4.5.2).

«**Использовать индекс?**» (**Use a index?**) – значения выбираются из списка «**yes**» и «**no**», значение по умолчанию – «**no**» (см. п. 3.4.5.2).

«**Тип числа с плавающей точкой**» (**Float Type**) – значения выбираются из списка «**Float**», «**Float-Pic**», «**Fix point binary**» и «**Fix point BCD**», значение по умолчанию – «**Float**» (см. п. 3.4.5.3).

«**Точность**» (**Precision**) – диапазон значений – от 0 до 7, значение по умолчанию – 2 (см. п. 3.4.5.4).

Параметры переменных протокола OVEN проиллюстрированы на рис. 3.16.



---

**Примечание.** Параметры **Address Length**, **Address** используются в модулях **Owen (Master)** и **Owen (Spy)**, а **Polling time** только в **Owen (Master)**. Для модуля **Owen (Slave)** данные параметры задавать не требуется.

---

Отличие переменной **Float** с модификатором времени состоит в том, что, при тех же параметрах, в поле данных, кроме собственно значения, присутствует еще и время (в сотых долях сек).

Переменная типа **Unsigned** позволяет пользователю передавать любые данные в любом произвольном (в т.ч., собственном специализированном) формате. В переменную типа **Unsigned** командой **Insert element** контекстного меню вставляются переменные длиной 1 байт, 2 байта, либо 4 байта. Допускается вставить не более 4-х переменных суммарной длиной не более 16 байт. При этом необходимо учитывать принцип выравнивания, см. п. 3.1.2.1. При информационных обменах передаются все набранные данные.

Особенность переменной формата **String** – использование максимальной длины до 15 символов в соответствии со стандартом протокола ОВЕН.

Переменная типа **Time** позволяет передать время. Особенность протокола Owen при передаче времени в том, что переменная имеет следующий формат: **год:месяц:день:час:минута:секунда:миллисекунда**. Однако все эти данные нужны полностью не всегда, и можно передавать любой фрагмент общего формата. Параметр **«Наименьшее (по значению) поле времени» (Less Time Field)** определяет последнюю границу.

Примеры задания параметров для различных случаев применения ПЛК представлены в Приложении Б.

#### **3.4.5.1. Параметр «Имя переменной» (Hash name)**

Параметр **«Имя переменной» (Hash name)** – задает имя переменной. Пользователь может вводить любое имя в соответствии с протоколом Owen. Вводимое имя преобразуется в ПЛК в **Hash**-код, который затем и используется при функционировании.

#### **3.4.5.2. Параметры «Индекс прибора» (Index) и «Использовать индекс?» (Use a index?)**

Параметр **«Индекс прибора» (Index)** – задает индекс прибора, параметр **«Использовать индекс?» (Use a index?)** – задает использование индекса. В совокупности применяются для управления конфигурационными

параметрами ПЛК, определяют наличие линейного индекса у параметра и задают значение индекса.

### **3.4.5.3. Параметр «Тип числа с плавающей точкой» (Float Type)**

Параметр «**Тип числа с плавающей точкой**» (**Float Type**) – уточняет тип переменной типа **Float**. Выбирается пользователем из выпадающего списка:

- **Float** – число с плавающей точкой в формате IEEE, обычно используемое в программировании, в **CoDeSys** называется Real, имеет длину 4 байта;
- **Float-Pic** – переменная размером в 3 байта, и один байт из мантииссы удаляется, т.е. число с меньшей точностью, но и размер переменной (количество байт) меньше;
- **Fix point binary** – число с фиксированной точкой в двоичном виде;
- **Fix point BCD** – число с фиксированной точкой в двоично-десятичном виде.

---

**Внимание.** Для переменных иных типов данный параметр отсутствует.

---

### **3.4.5.4. Параметр «Точность» (Precision)**

Параметр «**Точность**» (**Precision**) – определяет точность для параметров с фиксированной точкой: если выбирается значение точности 2, то число 10.12 именно так и будет передано. При 1 – 10.1, при 3 – 10.120. Этот параметр определяет положение десятичной точки. Диапазон значений – от 0 до 7.

### 3.5. Модуль «Owen (Spy)»

**Owen (Spy)** – специализированный модуль, обеспечивающий мониторинг информационных обменов в сети ОВЕН, в которую интегрирован ПЛК.

Технология применения модуля **Owen (Spy)** – добавление в конфигурацию, настраивание работы модуля (установка com-порта, через который будет производиться информационный обмен, настройка параметров, добавление переменных и пр.) и его функционирование по протоколу ОВЕН – аналогична технологии модуля **Owen (Slave)**.

Отличительной особенностью **Owen (Spy)** является то, что это устройство, которое не отвечает ни на чьи запросы, а только улавливает чужой обмен. Модуль может быть настроен таким образом, что, при опросе Мастером сети какого-либо устройства и его ответе, этот ответ модулем **Owen (Spy)** улавливается и записывается к себе в переменную. Данная возможность используется в тех случаях, когда ПЛК необходимо интегрировать в какую-то уже существующую сеть и получать от нее данные для последующей обработки и выполнения заданных действий. В этом случае нет необходимости в смене Мастера корректируемой сети, можно вообще не останавливать работу системы в ходе интеграции. Установка модуля **Owen (Spy)** решает все эти проблемы: ПЛК улавливает необходимые ему данные и выполняет запрограммированные действия.

#### Параметры модуля:

«Видимость» (**Visibility**) – задает видимость параметров модуля в программе EasyWorkPLC. Значения выбираются из списка «**yes**» и «**no**», значение по умолчанию – «**yes**» (см. п. 2.1.2).

Модуль **Owen (Spy)** со вставленной переменной представлен на рис. 3.16.

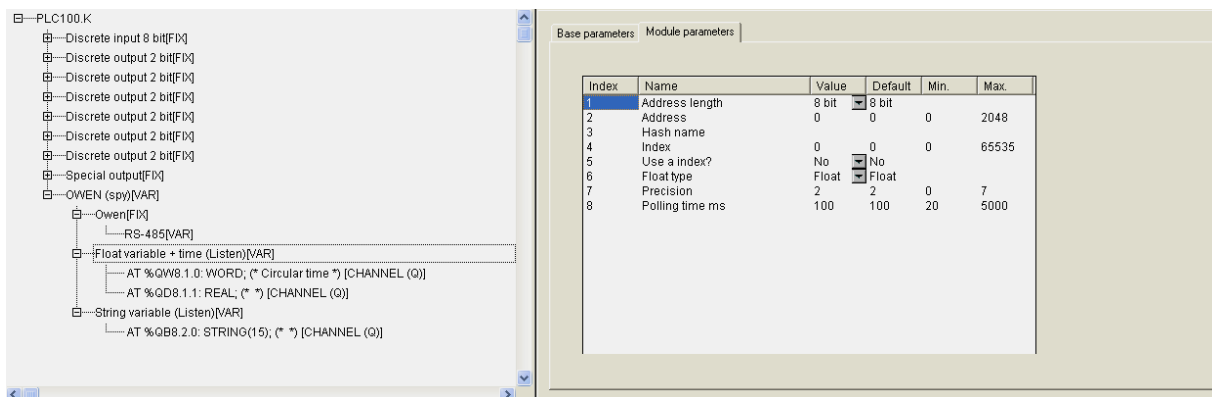


Рис. 3.16. Модуль Owen (Spy)

### 3.5.1. Переменные протокола ОВЕН

В модуле **Owen (Spy)** используются переменные, описанные в п. 3.4.5.

#### Параметры переменных протокола ОВЕН:

«**Длина адреса**» (**Address Length**) – значения выбираются из списка «**8 bit**» и «**11 bit**», значение по умолчанию – «**8 bit**» (см. п. 3.5.1.1).

«**Адрес устройства**» (**Address**) – диапазон значений от 0 до 255 или от 0 до 2048, в зависимости от размера адреса, значение по умолчанию – 0 (см. п. 3.5.1.2).

«**Имя переменной**» (**Hash name**) – вводится имя переменной (см. п. 3.4.5.1).

«**Индекс прибора**» (**Index**) – диапазон значений от 0 до 65535, значение по умолчанию – 0 (см. п. 3.4.5.2).

«**Использовать индекс?**» (**Use a index?**) – значения выбираются из списка «**yes**» и «**no**», значение по умолчанию – «**no**» (см. п. 3.4.5.2).

«**Тип числа с плавающей точкой**» (**Float Type**) – значения выбираются из списка «**Float**», «**Float-Pic**», «**Fix point binary**» и «**Fix point BCD**», значение по умолчанию – «**Float**» (см. п. 3.4.5.3).

«**Точность**» (**Precision**) – диапазон значений – от 0 до 7, значение по умолчанию – 2 (см. п. 3.4.5.4).

---

#### Примечания.

1. Параметры «**Индекс прибора**» (**Index**), «**Использовать индекс?**» (**Use a index?**), «**Тип числа с плавающей точкой**» (**Float Type**), «**Точность**» (**Precision**) описываются и применяются так же, как и в модуле **Owen (Slave)**.

2. Применение параметра «**Имя переменной**» (**Hash name**) имеет отличие: его значение – это «**имя прибора**» – имя, которое записано в документации прибора (МВА, МВУ, ТРМ, др.), ответ которого перехватывается. Пользователь может вводить любое имя в соответствии с протоколом Owen, причем символом может быть как просто символ, так и символ с точкой. Вводимое имя преобразуется в ПЛК в **Hash**-код, который затем и используется при функционировании.

3. Параметр «**Период опроса устройства, сек**» (**Polling time, ms**) используется только в Мастере ОВЕН.

---

#### 3.5.1.1. Параметр «Длина адреса» (**Address Length**)

Параметр «**Длина адреса**» (**Address Length**) – задает в битах длину адреса прибора, ответ которого необходимо перехватить. Приборы разработки ПО «Овен» поддерживают два варианта адресации – 8 и 11 битную.

### **3.5.1.2. Параметр «Адрес устройства» (Address)**

Параметр **«Адрес устройства» (Address)** – задает адрес прибора, ответ которого необходимо перехватить. Значение может быть в диапазоне от 0 до 255 или от 0 до 2047, в зависимости от размера адреса.

### 3.6. Модуль «Owen (Master)»

Функционирование ПЛК по протоколу **Owen** в «**ведущем**» режиме, т.е. режиме «**Master**», осуществляется при установке в ПЛК модуля **Owen (Master)**, который будет по выбранному коммуникационному интерфейсу работать активно – обмениваться данными с другими устройствами.

Настройка **Owen (Master)** аналогична описанной выше для **Owen (Slave)**.

#### Параметры модуля:

«**Максимальное время задержки ответа, мс**» (**Max response delay, ms**) – диапазон значений от 0 до 32000, значение по умолчанию – 50 (см. п. 3.6.1.1).

«**Видимость**» (**Visibility**) – задает видимость параметров модуля в программе EasyWorkPLC. Значения выбираются из списка «**yes**» и «**no**», значение по умолчанию – «**yes**» (см. п. 2.1.2).

Информация о функционировании опрашиваемых приборов выводится в переменные, сигнализирующие об ошибках, чтобы можно было проследить, что конкретно запрашивалось, какая ошибка, какой прибор последним опрашивался (по последнему **Hash-код** и последнему адресу). С этой целью в модуль фиксировано вставлены три переменные, в которых индицируется информация о ходе опроса подчиненных устройств (по аналогии с **Modbus (Master)**):

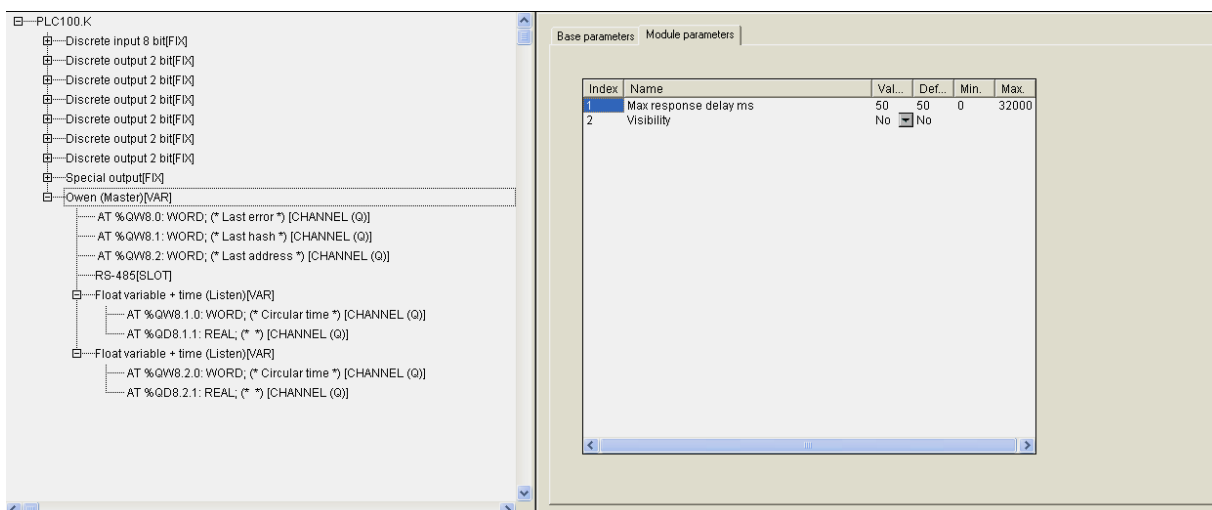
«**Последний адрес**» (**Last address**) – индицируется адрес прибора, по которому обращался **Owen (Master)** последним. Модуль запрашивает устройство, и, соответственно, тут же меняются значения: показывается значение адреса именно последнего запроса,

«**Код последней ошибки**» (**Last error**) – показывается код ошибки, которая произошла при последнем опросе.

«**Последний Hash-код**» (**Last Hash**) – показывается **Hash-код** параметра, который присутствовал в последнем опросе.

Сообщения об ошибках в ПЛК представлены в Приложении В.

Модуль **Owen (Master)** представлен на рис. 3.17.



**Рис. 3.17. Модуль Owen (Master)**

В отличие от **Owen (Slave)** в **Owen (Master)** отсутствует подмодуль **Owen (FIX)** и сразу вставлен модуль интерфейса. Пользователь может заменить его тип командой **Replace element** контекстного меню.

### 3.6.1. Переменные протокола ОВЕН

В модуле **Owen (Master)** используются переменные, описанные в п. 3.4.5.

При вводе переменной, например, переменной **Float variable** (см. рис. 3.16), пользователь задает значения параметров, упомянутых при описании модуля **Owen (Slave)** (см. п. 3.4).

#### Параметры переменных протокола ОВЕН:

«**Длина адреса устройства**» (**Address Length**) – значения выбираются из списка «**8 bit**» и «**11 bit**», значение по умолчанию – «**8 bit**» (см. п. 3.4.2).

«**Адрес устройства**» (**Address**) – диапазон значений от 0 до 255 или от 0 до 2048, в зависимости от размера адреса, значение по умолчанию – 0 (см. п. 3.4.3).

«**Имя переменной**» (**Hash name**) – вводится имя переменной (см. п. 3.4.5.1).

«**Индекс прибора**» (**Index**) – диапазон значений от 0 до 65535, значение по умолчанию – 0 (см. п. 3.4.5.2).

«**Использовать индекс?**» (**Use a index?**) – значения выбираются из списка «**yes**» и «**no**», значение по умолчанию – «**no**» (см. п. 3.4.5.2).

«**Тип числа с плавающей точкой**» (**Float Type**) – значения выбираются из списка «**Float**», «**Float-Pic**», «**Fix point binary**» и «**Fix point BCD**», значение по умолчанию – «**Float**» (см. п. 3.4.5.3).

«**Точность**» (**Precision**) – диапазон значений – от 0 до 7, значение по умолчанию – 2 (см. п. 3.4.5.4).

**«Период опроса устройства, сек» (Polling time, ms)** – диапазон значений – от 20 до 5000, значение по умолчанию – 20 (см. п. 3.2.1.6).

---

**Примечания.**

1. Параметры **«Индекс прибора» (Index)**, **«Использовать индекс?» (Use a index?)**, **«Тип числа с плавающей точкой» (Float Type)**, **«Точность» (Precision)** описываются и применяются также, как и в модуле **Owen (Slave)**.

2. Применение параметра **«Имя переменной» (Hash name)** имеет отличие: его значение – это **«имя прибора»** – имя, которое записано в документации прибора (МВА, МВУ, ТРМ, др.), ответ которого запрашивается. Пользователь может вводить любое имя в соответствии с протоколом Owen, причем символом может быть как просто символ, так и символ с точкой. Вводимое имя преобразуется в ПЛК в **Hash**-код, который затем и используется при функционировании.

3. Параметр **«Период опроса устройства, сек» (Polling time, ms)** идентичен аналогичному параметру, используемому в **Modbus (Master)**, с отличием в значениях параметра: диапазон от 20 мс до 5 сек.

**Внимание.** Пользователь должен учитывать физические ограничения, накладываемые характеристиками сети: скорость информационного обмена в сети ограничена, и, если задается большое количество переменных, которые к тому же должны часто опрашиваться, то информация будет поступать к прибору с запаздыванием. Следовательно, пользователь должен заранее просчитать пропускную способность сети, и, соответственно, либо уменьшить частоту опроса, либо производить опрос по разным линиям и/или др.

---

### **3.6.1.1. Параметр «Максимальное время задержки ответа, мс» (Max response delay, ms)**

Параметр **«Максимальное время задержки ответа, мс» (Max response delay, ms)** задает то время, за которое опрашиваемый прибор должен ответить на запрос **Owen (Master)**. Если он не отвечает, то считается, что прибор отключен или произошел обрыв линии связи.



### 3.7. Модуль «DCON (Master)»

Функционирование ОВЕН ПЛК по протоколу **DCON** реализовано только в режиме «ведущего», т.е. режиме «**Master**», и осуществляется при установке в ПЛК модуля **DCON (Master)**, который по выбранному коммуникационному интерфейсу работает активно – обменивается данными с другими устройствами по протоколу **DCON**.

#### Параметры модуля:

«**Видимость**» (**Visibility**) – задает видимость параметров модуля в программе EasyWorkPLC. Значения выбираются из списка «**yes**» и «**no**», значение по умолчанию – «**yes**» (см. п. 2.1.2).

Модуль **DCON (Master)** представлен на рис. 3.18.

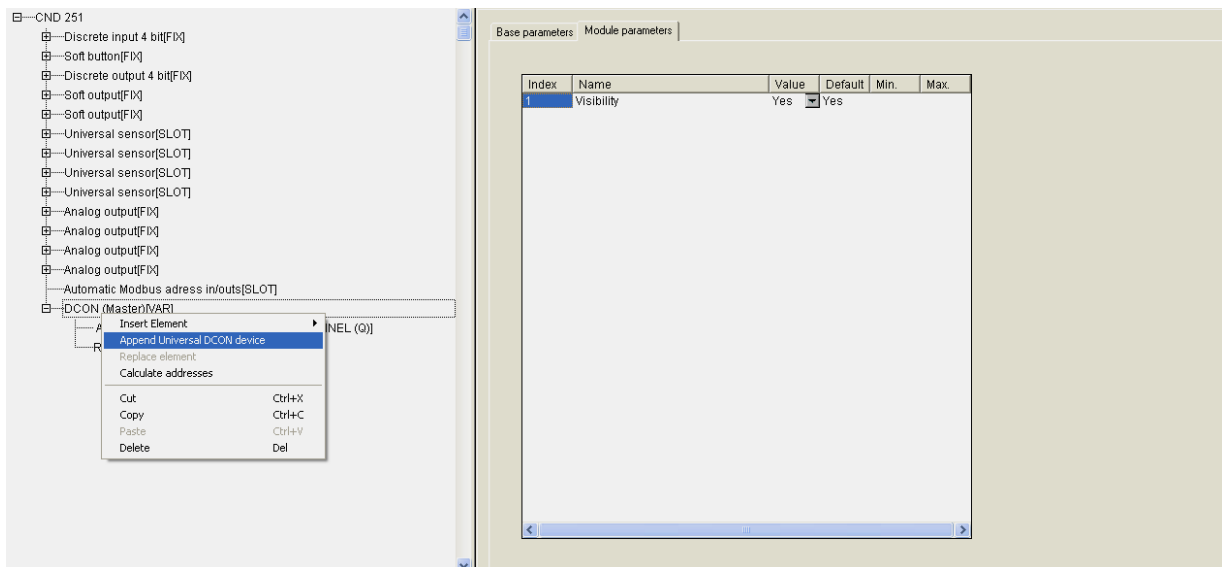


Рис. 3.18. Модуль DCON (Master)

Как и в варианте **Owen (Master)**, в модуле **DCON (Master)** используются только последовательные порты. Настройка последовательного порта описана в п. 1.2.

Информация о функционировании опрашиваемых приборов выводится в переменную, сигнализирующую об ошибках. С этой целью в модуль фиксировано вставлена переменная, в которую индицируется информация о ходе опроса подчиненных устройств (по аналогии с **Modbus (Master)**):

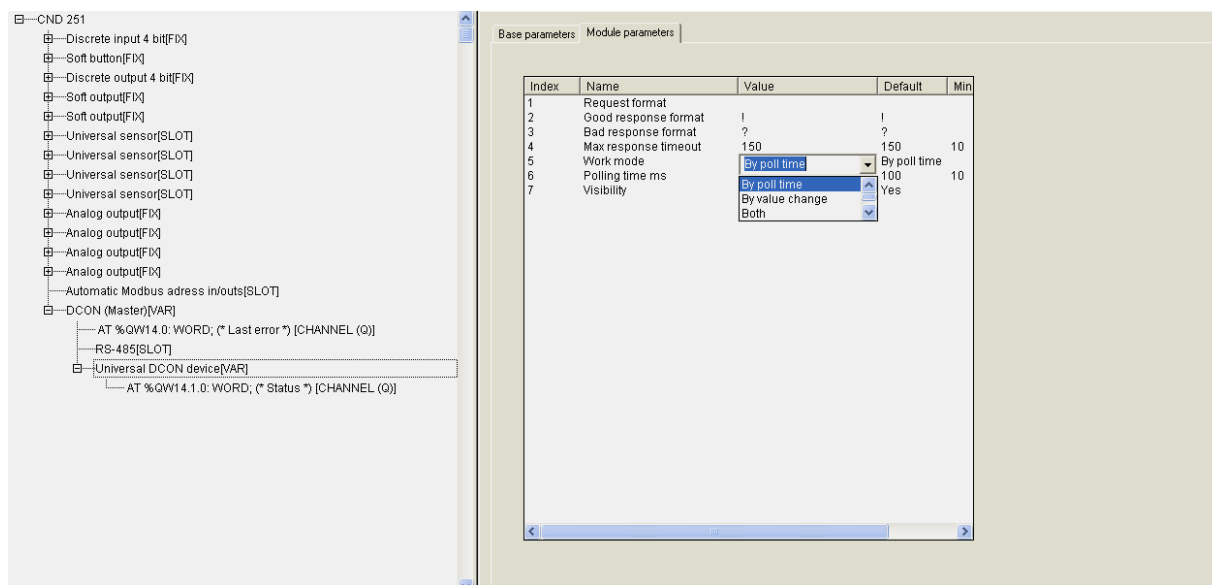
«**Код последней ошибки**» (**Last error**) – показывается код ошибки, которая произошла при последнем опросе.

Организация работы **DCON (Master)** аналогична описанной выше для **Modbus (Master)**: установка и настройка устройств, опрашиваемых модулем, производится через установку «универсального устройства **DCON (Master)**» и задание необходимых параметров. Его подключение

выполняется командой «**Append Universal DCON Device**», вызываемой из контекстного меню.

### 3.7.1. Универсальное устройство DCON (Universal DCON Device)

Универсальное устройство **DCON (Universal DCON Device)** представлено на рис. 3.19.



**Рис. 3.19. Универсальное устройство DCON**

При вызове универсального устройства появляется шаблон, не заполненный, параметры и переменные которого не привязаны к конкретному модулю. При заполнении полей параметров значениями, определении переменных входов-выходов, устройство определит конкретный вариант вставляемого устройства.

Для конфигурирования универсального устройства **DCON** необходимо выполнить следующие действия:

- подключить к модулю необходимое количество входных и выходных каналов (переменных); подключение осуществляется вызовом команды **Append Subelement** контекстного меню;
- задать в параметрах модуля форматы строк запросов и ответов, которые будут посылаться в подключаемое устройство и приниматься от него. Данные из выходных переменных (**output variables**) модуля будут кодироваться в определенный формат и вставляться в передаваемые строки протокола **DCON**. Данные из строк ответов устройства будут декодироваться и вставляться во входные переменные модуля (**input variables**);
- задать прочие параметры модуля.

У устройства **DCON** есть поле информации о собственном статусе, который хранится в переменной **Статус (Status)**. В поле **Статус** фиксируются ошибки обмена с внешним устройством или хранится код, свидетельствующий об успешном обмене.

Переменная **Статус (Status)** может использоваться для управления включением/выключением рабочего состояния модуля: если в переменной **Status** модуля записана специальная команда, то происходит либо старт работы модуля, либо его остановка (**0x00FE** – «стоп», а **0x00FF** – «старт»).

#### **Параметры универсального устройства DCON (Master):**

**«Формат запроса» (Request format)** – устанавливается формат запроса, может быть любым, ограничения не накладываются (см. п. 3.7.1.1).

**«Формат правильного ответа» (Good response format)** – устанавливается формат правильного ответа, значение по умолчанию – «!» (см. п. 3.7.1.1).

**«Формат неправильного ответа» (Bad response format)** – устанавливается формат неправильного ответа, значение по умолчанию – «?» (см. п. 3.7.1.1).

**«Максимальное время ответа» (Max response timeout)** – задает максимальное допустимое время ответа. Значение сверху не ограничено, может быть любым, в т.ч. с плавающей точкой, но не меньше 10 мс, значение по умолчанию – 150 мс (см. п. 3.7.1.2).

**«Режим работы» (Work mode)** – значения выбираются из списка **«По времени» (By poll time)**, **«По изменению значения переменных» (By value change)**, **«По времени и по изменению значения переменных» (Both)** и **«По команде» (By command)**, значение по умолчанию – **«By poll time»** (см. п. 3.7.1.3).

**«Видимость» (Visibility)** – задает видимость параметров модуля в программе EasyWorkPLC. Значения выбираются из списка **«yes»** и **«no»**, значение по умолчанию – **«yes»** (см. п. 2.1.2).

**«Период опроса устройства, в мс» (Polling time)** – диапазон значений от 10 до 10000, значение по умолчанию – 100 (см. п. 3.2.1.6).

#### **3.7.1.1. Параметры «Формат запроса» (Request format), «Формат правильного ответа» (Good response format) и «Формат неправильного ответа» (Bad response format)**

Протокол **DCON** подразумевает, что при организации опроса устройств создается строка запроса, и, при ее послышке, опрашиваемое устройство может вернуть 2 варианта ответа: ответ правильный (команда распознана, данные есть) – один формат, и ответ неправильный (не распознана команда, нет данных и/или пр.) – другой формат.

Строки «**Формат запроса**» (**Request format**), «**Формат правильного ответа**» (**Good response format**), «**Формат неправильного ответа**» (**Bad response format**) используются для задания формата запроса **DCON (Master)** и разбора правильного/неправильного ответа.

Строки формата ответа могут не задаваться, если устройство не отвечает на запрос.

Строка формата представляет собой строку, содержащую символы и спецкоманды.

**Символы** – любой символ, кроме служебных, к которым относятся символы «**\$**» (знак доллара), «**[**» и «**]**» (открывающая и закрывающая квадратные скобки).

---

**Примечание.** При необходимости вывести служебный символ в качестве обычного он вводится в строку два раза подряд.

---

**Спецкоманда** имеет формат: [**{модификатор} действие**].

**Модификатор** – количество символов, обрабатываемых действием. Представляет собой десятичное целое число. Может быть у всех действий, кроме вычисления контрольных сумм. Наличие модификатора необязательно, значение по умолчанию = 1.

**Действие** – отображается в строке спецкоманды одним из символов – **D, H, F, S, \*, +, %**. Регистр символов значения не имеет.

Символы соответствуют следующим видам действий:

- **D** – представляет передаваемую переменную в ASCII-символах в десятичном формате (без знака) или преобразует ASCII-строку из десятичного формата (без знака) в принимаемую переменную. Количество символов задается **модификатором**;
- **H** – представляет передаваемую переменную в ASCII-символах в шестнадцатичном формате или преобразует ASCII-строку из шестнадцатичного формата в принимаемую переменную. Количество символов задается **модификатором**;
- **F** – представляет передаваемую переменную в ASCII-символах в десятичном формате со знаком, разделителем целой и дробной части числа (точкой). Строка имеет фиксированное число символов, заданное **модификатором**. Для принимаемых переменных производит обратное преобразование из ASCII-строки в число;
- **S** – осуществляет прямое копирование из передаваемой строковой переменной в строку запроса числа символов, заданного **модификатором** или обратное копирование из строки ответа в принимаемую переменную строкового типа;
- **\*** – задает в строке ответа набор символов, которые надо пропустить. Количество символов может быть задано **модификатором**;

- + – вставляет в строку запроса контрольную сумму или получает ее в строке ответа. Контрольная сумма вычисляется путем сложения с переполнением по модулю 256. Данное действие не может иметь **модификатора**;
- % – вставляет в строку запроса контрольную сумму или получает ее в строке ответа. Контрольная сумма вычисляется по 8-ми битному полиному (DOW-CRC). Данное действие не может иметь **модификатора**

---

**Примечание.** При работе ОВЕН ПЛК по протоколу DCON есть 3 варианта работы: без расчета контрольных сумм, с расчетом контрольных сумм путем сложения значений всех символов и с расчетом контрольных сумм 8-ми битных. Вариант работы пользователь выбирает в соответствии с тем, какой вариант расчета контрольной суммы используется в опрашиваемом приборе.

---

**Используются следующие алгоритмы преобразования:**

**при формировании запроса** – все символы вне спецкоманд копируются в строку запроса без изменения, спецкоманды заменяются на значения передаваемых (выходных) переменных. Значения переменных кодируются в формате, заданном **действием**, число символов соответствует **модификатору**;

**при разборе ответа** – все символы вне спецкоманд сравниваются с соответствующими позициями ответа и, при нахождении различия, вырабатывается сообщение об ошибке. Данные в позициях ответа, соответствующих спецкомандам, преобразуются и сохраняются в соответствующих принимаемых (входных) переменных.

Если запрос жестко фиксированный, т.е. в строке не содержатся изменяемые данные, то строка набивается без каких-либо команд, в таком виде отсылается, при этом может быть добавлена контрольная сумма.

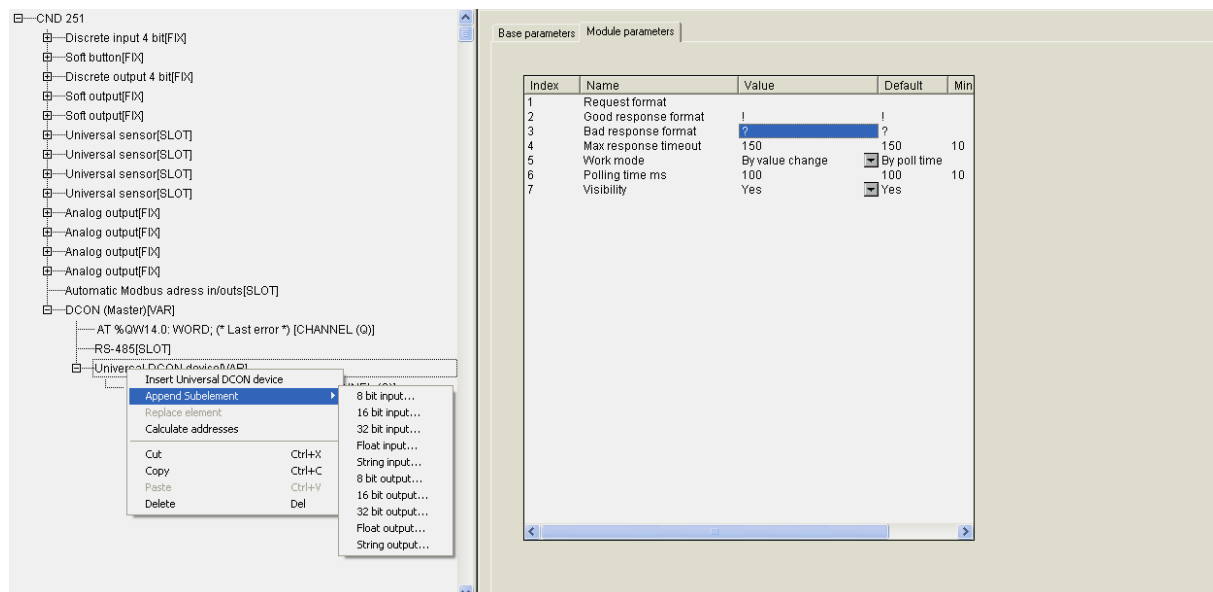
Аналогично с ответом: в случае прихода строки без содержания каких-либо данных (в конце может быть контрольная сумма), это будет означать, что прибор работает, реагирует и пр., т.е. сам факт получения ответа от прибора уже является информацией.

В универсальное устройство добавляются переменные, входящие и выходящие. Входящие переменные (**input variables**) заполняются, когда приходит ответ от опрашиваемого прибора и в них фиксируются значения, выходящие переменные (**output variables**) используются при формировании запроса.

Универсальное устройство позволяет добавлять переменные различного размера (8,16 и 32 бита и 16 байтная строка) и типа (целое беззнаковое, с плавающей точкой или строка). Размер и порядок расположения входных и выходных переменных должны соответствовать строке команды. Переменные, которые будет использоваться ПЛК при опросе

устройств по протоколу **DCON**, выбираются пользователем функцией контекстного меню **Append Subelements** (см. рис 3.20).

Примеры настройки модуля **DCON (Master)** для опроса устройств ввода/вывода представлены в Приложении Г.



**Рис. 3.20. Добавление переменных универсального устройства модуля DCON (Master)**

### **3.7.1.2. Параметр «Максимальное время ответа» (Max response timeout)**

Параметр «**Максимальное время ответа**» (**Max response timeout**) задает то время, за которое опрашиваемый прибор должен ответить на запрос **DCON (Master)**. Если он не отвечает, то считается, что прибор отключен или произошел обрыв линии связи. Информация об этом заносится в переменную «**Код последней ошибки**» (**Last error**).

### **3.7.1.3. Параметр «Режим работы» (Work mode)**

Параметр «**Режим работы**» (**Work mode**) – определяет выбор варианта работы модуля **DCON (Master)** при опросе приборов из 4-х возможных вариантов:

- **By poll time** – «по времени» – контролируемые устройства опрашиваются с периодичностью, заданной в параметре «**Период опроса устройства**» (**Polling time**);

- **By value change** – «по изменению значения переменных» – модуль **DCON (Master)** генерирует запрос устройству при изменении значений выходных переменных модуля;
- **Both** – «оба var» – опрос производится с периодом, заданным в параметре **Polling time** плюс тогда, когда изменилось значение переменных;
- **By command** – «по команде» – производится однократная посылка запроса, когда в переменную **Статус (Status)** записывается значение **0x00FF**.

### 3.8. Модуль статистики (Statistic)

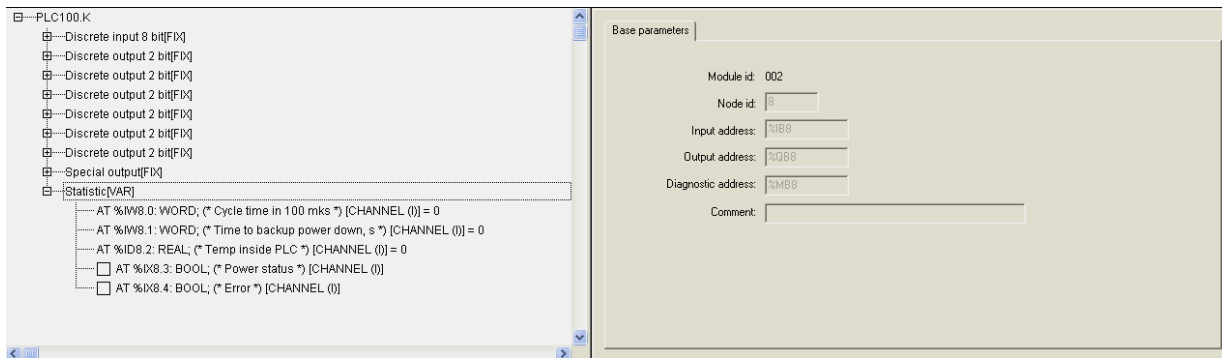
Назначением **модуля статистики (Statistic)** является выдача в программу пользователя информационных данных о функционировании ОВЕН ПЛК:

- наличие/отсутствие питания – отображается в битовом канале **Power status** (питание есть –1);
- информация об ошибках, возникающих при работе ПЛК, – отображается в битовом канале **Error** (0 – ПЛК работает штатно, 1 – произошла ошибка при функционировании ПЛК);
- 3 строки показателей работы ПЛК:
  - значение **цикла работы ПЛК в сотнях мкс (Cycle time in 100 mks)**, позволяет пользователю оценить объем вычислительных ресурсов, который требуется для работы написанной им программы. Если цикл оказывается больше, чем значение параметра **MinCycleLength**, заданного в параметрах работы ПЛК, то оказывается, что программа пользователя слишком требовательна к ресурсам, и этот параметр желательно увеличить, чтобы циклы не перекрывались;
  - значение **времени, оставшегося до выключения ПЛК в сек (Time to backup power down, s)**, отражает временной ресурс для варианта работы ПЛК с отключенным внешним питанием (питание от аккумуляторной батареи). Характеристика оценочная, точный учет влияния всех факторов (температура ПЛК и внешней среды, точность измерений и пр.) затруднителен. В случае работы ПЛК от сети аккумуляторная батарея заряжается, и этот параметр косвенно указывает на процесс зарядки (600 сек соответствует полностью заряженной батарее);
  - **температура внутри ПЛК (Temp inside PLC)**, отражает температуру, замеренную датчиком внутри корпуса ПЛК (у разных моделей ПЛК температура может измеряться на разных платах, определяется интенсивностью нагрева конкретных плат). Характеристика косвенно свидетельствует о рабочем состоянии ПЛК.

Параметры настройки работы модуля отсутствуют.

**Модуль статистики (Statistic)** представлен на рис. 3.21.





**Рис. 3.21. Модуль статистики**

## 3.9. Константы (Constant value)

**Константы (Constant value)** – это модули, содержащие постоянные значения (задаваемые как параметры модулей), помещаемые при загрузке ОВЕН ПЛК в соответствующий канал и считываемые, при необходимости, в программу ПЛК.

Основным назначением модулей **Константы (Constant value)** является их использование при работе программы **EasyWorkPLC**.

### Параметры модуля:

«**Значение константы**» (**Constant Value**) – диапазон значений зависит от типа вводимой константы, значение по умолчанию – 0 (см. п. 3.9.1).

«**Комментарий для программы EasyWorkPLC**» (**Comment for EasyWorkPLC**) – указывается строка комментария для программы **EasyWorkPLC** (см. п. 3.9.2).

«**Видимость**» (**Visibility**) – задает видимость параметров модуля в программе **EasyWorkPLC**. Значения выбираются из списка «**yes**» и «**no**», значение по умолчанию – «**yes**» (см. п. 2.1.2).

Модули **Константы (Constant value)** представлен на рис. 3.22.

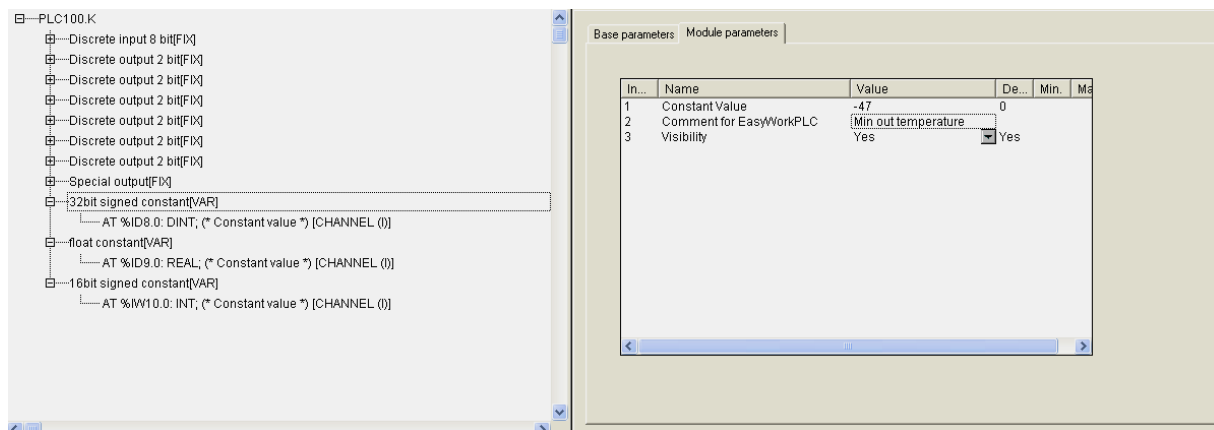


Рис. 3.22. Модули Константы (Constant value)

### 3.9.1. Параметр «Значение константы» (Constant Value)

Параметр «**Значение константы**» (**Constant Value**) устанавливает значение вводимой пользователем константы.

**Примечание.** В зависимости от вида константы пользователь вводит значения, учитывая возможные диапазоны вводимых констант, например: для 32-битной константы диапазон составляет  $[-2 \cdot 10^9 \dots 2 \cdot 10^9]$ , для константы

**Real** – числа с плавающей точкой, для 16-битной константы – [-32768...32768] и т.д.

---

### **3.9.2. Параметр «Комментарий для программы EasyWorkPLC» (Comment for EasyWorkPLC)**

Параметр **Комментарий для программы EasyWorkPLC» (Comment for EasyWorkPLC)** устанавливает значение вводимого пользователем комментария для программы **EasyWorkPLC**. Комментарий может быть написан на русском языке.

При запуске **EasyWorkPLC** в окне программы появляется список всех модулей и значений их параметров, при этом будут видны только параметры тех модулей, у которых значение параметра «**Видимость» (Visibility)** было установлено «**yes**».

При использовании этой программы комментарий высвечивается как имя константы, значение может редактировать пользователь. Таким образом, какие-то настройки на ПЛК можно поменять, и для этого не нужна среда **CoDeSys**, а используется программа **EasyWorkPLC**.

## Приложение А. Типы датчиков, используемых в качестве аналоговых входов ОВЕН ПЛК

### А.1. Подключение датчиков типа ТСМ50, ТСП100 и ТСМ100 к ПЛК150

Датчики типа ТРМ50, ТСП100 и ТСМ100 имеют относительно небольшое сопротивление, сравнимое с сопротивлением проводов, которыми эти датчики подключаются. Из-за этого сопротивление проводов вносит большую дополнительную погрешность. Обычно подключение таких термосопротивлений осуществляется по трехпроводной схеме, позволяющей измерить и компенсировать сопротивление проводов, но ПЛК150 не имеет возможности подключения датчиков по трехпроводной схеме. Поэтому при подключении датчиков ТСМ50, ТСП100 и ТСМ100 по двухпроводной схеме необходимо выполнить следующие действия:

1. Подключить линию связи «прибор-датчик» к контроллеру согласно схемам подключения.
2. Настроить контроллер на работу с датчиком требуемого типа.
3. Со стороны датчика к линии связи подключить магазин сопротивлений с классом точности не более 0.1 или эталонный резистор.
4. Установить на магазине сопротивлений значение 50 Ом для датчиков ТСМ50 или 100 Ом для датчиков ТСМ100 и ТСП100.
5. Снять показания измерений со входа контроллера.
6. В параметре **First point** модуля требуемого аналогового входа задать значение 0 (что соответствует значению 0° С) и в первом параметре **Delta** задать значение, равное измеренному, но с обратным знаком.
7. Повторно снять показания с измерительного входа контроллера и удостовериться, что они равны или близки к 0.
8. Подсоединить вместо магазина сопротивлений датчик.

### А.2. Перечень типов датчиков

Типы датчиков, используемых в качестве аналоговых входов ОВЕН ПЛК представлены в табл. А.1.

Таблица А.1

#### Входные первичные преобразователи

Наименование	Аббревиатура типа в параметрах универсального датчика	Диапазон контроля
<b>Термопреобразователи сопротивления (по ГОСТ 6651-94) (Termoresistor sensors)</b>		
ТСМ (Cu 50) $W_{100} = 1,4260$	R426_50	-50 °С...+200 °С
ТСМ (50М) $W_{100} = 1,4280$	R428_50	-190 °С...+200 °С
ТСП (Pt 50) $W_{100} = 1,3850$	R385_50	-200 °С...+750 °С
ТСП (50П) $W_{100} = 1,3910$	R391_50	-200 °С...+750 °С

ТСМ (Cu 100) $W_{100} = 1,4260$	R426_100	-50 °C...+200 °C
ТСМ (100M) $W_{100} = 1,4280$	R428_100	-190 °C...+200 °C

**Продолжение табл. А.1**

<b>Наименование</b>	<b>Аббревиатура типа в параметрах универсального датчика</b>	<b>Диапазон контроля</b>
ТСП (Pt 100) $W_{100} = 1,3850$	R385_100	-200 °C...+750 °C
ТСП (100П) $W_{100} = 1,3910$	R391_100	-200 °C...+750 °C
ТСН (100Н) $W_{100} = 1,6170$	R617_100	-60 °C...+180 °C
ТСМ (Cu 500) $W_{100} = 1,4260$	R426_500	-50 °C...+200 °C
ТСМ (500M) $W_{100} = 1,4280$	R428_500	-190 °C...+200 °C
ТСП (Pt 500) $W_{100} = 1,3850$	R385_500	-200 °C...+750 °C
ТСП (500П) $W_{100} = 1,3910$	R391_500	-200 °C...+750 °C
ТСН (500Н) $W_{100} = 1,6170$	R617_500	-60 °C...+180 °C
ТСМ (Cu 1000) $W_{100} = 1,4260$	R426_1000	-50 °C...+200 °C
ТСМ (1000M) $W_{100} = 1,4280$	R428_1000	-190 °C...+200 °C
ТСП (Pt 1000) $W_{100} = 1,3850$	R385_1000	-200 °C...+750 °C
ТСП (1000П) $W_{100} = 1,3910$	R391_1000	-200 °C...+750 °C
ТСН (1000Н) $W_{100} = 1,6170$	R617_1000	-60 °C...+180 °C

**Термопары (по ГОСТ Р 8.585-2001) (Thermocouple sensors)**

ТХК (L)	TP_L	-200 °C...+800 °C
ТЖК (J)	TP_J	-200 °C...+1200 °C
ТНН (N)	TP_N	-200 °C...+1300 °C
ТХА (K)	TP_K	-200 °C...+1300 °C
ТПП (S)	TP_S	0 °C...+1600 °C
ТПП (R)	TP_R	0 °C...+1600 °C
ТПР (B)	TP_B	+200 °C...+1800 °C
ТВР (A-1)	TP_WR1	0 °C...+2500 °C
ТВР (A-2)	TP_WR2	0 °C...+1800 °C
ТВР (A-3)	TP_WR3	0 °C...+1600 °C
ТМК (T)	TP_T	-200 °C...+400 °C

**Унифицированные сигналы постоянного напряжения и тока (по ГОСТ 26.011-80) (Universal sensors)**

0...5,0 мА	IT_0_5	0...100 %
0...20,0 мА	IT_0_20	0...100 %
4,0...20,0 мА	IT_4_20	0...100 %
-50,0...+50,0 мВ	mV	0...100 %
0...1,0 В	U0_1	0...100 %
0...10,0 В	U0_10	0...100 %

**Датчики сопротивления (Universal sensors)**

резистивный (0... 5000 Ом)	RO_5000	0...100 %
----------------------------	---------	-----------

**Примечания.**

- $W_{100}$  – отношение сопротивления датчика, измеренное при температуре 100 °C, к его сопротивлению, измеренному при 0 °C ( $R_0$ ).
- Для работы с прибором могут быть использованы только изолированные термопары с незаземленными рабочими спаями.

## Приложение Б. Примеры настройки опроса переменных по протоколу ОВЕН

В Приложении представлены примеры настройки опроса переменных модуля **Owen (Master)** для некоторых наиболее часто встречающихся случаев применения ПЛК (см. табл. Б.1чБ5). Эти примеры можно использовать и при настройке модулей **Owen (Slave)** и **Owen (Spy)**.

**Примечание.** Пользователь должен помнить, что при настройке модулей **Owen (Spy)** и **Owen (Slave)** не используется параметр **Poling Time** («период опроса»). При настройке модуля **Owen (Slave)** также не используются параметры **Address Type** («тип адреса») и **Address** («адрес»). Они задаются в самом модуле **Owen (Slave)**.

Таблица Б.1

Пример 1 настройки опроса переменных модуля Owen (Master)

	Тип данных с числом с плавающей точкой и циклическим временем
<b>Описание переменной</b>	
Прибор	Все приборы с аналоговыми входами
Имя параметра	<b>read</b>
Описание	Значение с измерителя
Диапазон	В соответствии с диапазоном аналогового входа
Диапазон индекса	0 .. 6
<b>Описание настроек в окне конфигурации</b>	
Тип переменной	<b>Float variable + time</b>
Направление опроса	Чтение
<b>Address Length</b>	8/11 бит в зависимости от настроек сети
<b>Address</b>	Адрес прибора + номер канала с нуля
<b>Hash name</b>	read
<b>Index</b>	Индекс содержится в адресе прибора
<b>Use a index?</b>	No
<b>Float type</b>	Float
<b>Precision</b>	Не используется
<b>Polling time</b>	В мс, в зависимости от необходимого периода опроса
<b>Вставленные подмодули</b>	Отсутствуют

Таблица Б.2

## Пример 2 настройки опроса переменных модуля Owen (Master)

	<b>Тип данных с числом с фиксированной точкой</b>
<b>Описание переменной</b>	
Прибор	TPM133
Имя параметра	<b>itrL</b>
Описание	Период опроса датчика
Диапазон	0.1 .. 30
Диапазон индекса	0 .. 6
<b>Описание настроек в окне конфигурации</b>	
Тип переменной	<b>Float variable</b>
Направление опроса	Чтение/Запись (в зависимости от потребностей)
<b>Address Length</b>	8/11 бит в зависимости от настроек сети
<b>Address</b>	Адрес прибора
<b>Hash name</b>	<b>itrL</b>
<b>Index</b>	0-6 (в зависимости от номера канала измерения)
<b>Use a index?</b>	Yes
<b>Float type</b>	<b>Fix point binary</b>
<b>Precision</b>	1
<b>Polling time</b>	В мс, в зависимости от необходимого периода опроса
<b>Вставленные подмодули</b>	Отсутствуют

Таблица Б.3

## Пример 3 настройки опроса переменных модуля Owen (Master)

	<b>Тип данных со знаковым/беззнаковым целым значением</b>
<b>Описание переменной</b>	
Прибор	TPM133, 151, МВА и т.п.
Имя параметра	<b>in-t</b>
Описание	Тип датчика
Диапазон	Не ограничен
Диапазон индекса	0 .. 6
<b>Описание настроек в окне конфигурации</b>	
Тип переменной	<b>Unsigned variable</b>
Направление опроса	Чтение/Запись (в зависимости от потребностей)
<b>Address Length</b>	8/11 бит в зависимости от настроек сети
<b>Address</b>	Адрес прибора
<b>Hash name</b>	<b>in-t</b>
<b>Index</b>	0 .. 6
<b>Use a index?</b>	Yes
<b>Polling time</b>	В мс, в зависимости от необходимого периода опроса
<b>Вставленные подмодули</b>	
1	8 bits

Таблица Б.4

## Пример 4 настройки опроса переменных модуля Owen (Master)

	Тип данных с текстовой информацией
<b>Описание переменной</b>	
Прибор	Любой прибор ОВЕН
Имя параметра	<b>Dev</b>
Описание	Название прибора
Диапазон	Строка
Диапазон индекса	Нет индекса
<b>Описание настроек в окне конфигурации</b>	
Тип переменной	<b>String variable</b>
Направление опроса	Чтение (константное значение)
<b>Address Length</b>	8/11 бит в зависимости от настроек сети
<b>Address</b>	Адрес прибора
<b>Hash name</b>	<b>dev</b>
<b>Index</b>	Не используется
<b>Use a index?</b>	No
<b>Polling time</b>	В мс, в зависимости от необходимого периода опроса
<b>Вставленные подмодули</b>	Отсутствуют

Таблица Б.5

## Пример 5 настройки опроса переменных модуля Owen (Master)

	Тип данных с информацией о времени
<b>Описание переменной</b>	
Прибор	TPM133
Имя параметра	<b>t.val</b>
Описание	Время полного хода 3-х позиционного ИМ, мин:сек
Диапазон	1 .. 999
Диапазон индекса	0 .. 7
<b>Описание настроек в окне конфигурации</b>	
Тип переменной	<b>Time variable</b>
Направление опроса	Чтение/Запись (в зависимости от потребностей)
<b>Address Length</b>	8/11 бит в зависимости от настроек сети
<b>Address</b>	Адрес прибора
<b>Hash name</b>	<b>t.val</b>
<b>Index</b>	0-7 (в зависимости от номера канала)
<b>Use a index?</b>	Yes
<b>Less time field</b>	Seconds
<b>Polling time</b>	В мс, в зависимости от необходимого периода опроса
<b>Вставленные подмодули</b>	
1	Seconds
2	Minutes



## Приложение В. Сообщения об ошибках в ПЛК

В Приложении представлены сообщения об ошибках в ПЛК.

В ПЛК реализована комплексная технология сообщений об ошибках. Коды ошибок доступны при функционировании следующих модулей:

- модули аналогового входа;
- модули «Мастер» (ОВЕН, Modbus, DCON);
- модуль «Архиватор»;
- модуль архивации информации в файл.

### В.1. Ошибки датчиков (аналоговых входов)

Коды ошибок передаются через область памяти канала REAL в датчике. При возникновении ошибки старшие 4 бита выставляются в 1, в следующих 4-х битах следует код ошибки (см. табл. В.1).

Таблица В.1

Коды ошибок аналоговых входов

Краткое наименование	Код ошибки		Описание ошибки
	(hex)	(dec)	
<b>emSensorOff</b>	<b>0xF7</b>	<b>247</b>	Датчик отключен
<b>emHighColdSold</b>	<b>0xF8</b>	<b>248</b>	Велика температура холодного спая
<b>emLowColdSold</b>	<b>0xF9</b>	<b>249</b>	Мала температура холодного спая
<b>emHighValue</b>	<b>0xFA</b>	<b>250</b>	Вычисленное значение слишком велико (за пределами диапазона измерения)
<b>emLowValue</b>	<b>0xFB</b>	<b>251</b>	Вычисленное значение слишком мало (за пределами диапазона измерения)
<b>emShortCircuit</b>	<b>0xFC</b>	<b>252</b>	Короткое замыкание датчика
<b>emSensBreak</b>	<b>0xFD</b>	<b>253</b>	Обрыв датчика
<b>emADCErr</b>	<b>0xFE</b>	<b>254</b>	Отсутствие связи с АЦП
<b>emClbrCoef</b>	<b>0xFF</b>	<b>255</b>	Некорректный калибровочный коэффициент

### В.2. Коды ошибок модулей «Мастер»

#### В.2.1. Модуль Modbus (Мастер)

В модуле **Modbus (Мастер)** используются 2 поля для отображения статуса Мастера и возникающих ошибок.

- «**Last Address**» – содержит адрес последнего обращения (адрес последовательного устройства или IP-адрес, в зависимости от режима работы опрашивавшегося последним устройства);
- «**Last error**» – содержит код ошибки из табл. В.2.

Таблица В.2

## Коды ошибок модуля Modbus (Мастер)

Краткое наименование	Код ошибки		Описание ошибки
	(hex)	(dec)	
<b>OK</b>	<b>0x0000</b>	<b>0</b>	Нет ошибок
<b>NO_DEVICE</b>	<b>0x0051</b>	<b>81</b>	Превышен таймаут ожидания ответа
<b>NO_SOCKET</b>	<b>0x0054</b>	<b>84</b>	Нет свободного сокета для устройства TCP/IP
<b>SOCKET_ERROR</b>	<b>0x0055</b>	<b>85</b>	Ошибка при приеме/передаче по сети TCP/IP

## В.2.2. Модуль ОБЕН (Мастер)

В модуле **ОБЕН (Мастер)** используются 3 поля для отображения статуса мастера и возникающих ошибок:

- «**Last error**» – содержит код ошибки из табл. В.3
- «**Last Address**» – содержит адрес последнего обращения (адрес последовательного устройства или IP-адрес, в зависимости от режима работы опрашивавшегося последним устройством)
- «**Last HASH**» – содержит Hash-код переменной, которая была опрошена последней.

Таблица В.3

## Коды ошибок модуля ОБЕН (Мастер)

Краткое наименование	Код ошибки		Описание ошибки	
	(hex)	(dec)		
<b>OK</b>	<b>0x0000</b>	<b>0</b>	Нет ошибок	
<b>NO_DEVICE</b>	<b>0x0051</b>	<b>81</b>	Превышен таймаут ожидания ответа	
<b>N_ERR</b>	<b>0x0057</b>	<b>87</b>	В ответе устройства содержится Hash-код ошибки <b>n.Err (0x0233)</b> . Уточнение ошибки содержится в старшем байте кода ошибки (см. табл. В.4)	
<b>BAD_HASH</b>	<b>0x4000</b>	<b>16384</b>	Hash-код в ответе не соответствует ожидаемому	Эти коды ошибок могут быть вызваны совместной работой нескольких мастеров в одной сети
<b>BAD_ADDRESS</b>	<b>0x8000</b>	<b>32768</b>	Адрес в ответе не соответствует ожидаемому	

Таблица В.4

## Коды ошибок приборов в сети ОВЕН

Краткое наименование	Код ошибки в старшем байте		Описание ошибки
	(hex)	(dec)	
<b>Определение констант ошибок приема</b>			
OK	0	0	Безошибочный прием кадра
<b>Ошибки записи параметров и атрибутов функцией modifc</b>			
PDOT	2	2	Задано положение точки, превышающее 3
EROM	3	3	Попытка модификации ROM-параметра
ESTR	4	4	Не целое число при записи индекса строки или времени
EDOT	5	5	Неверно задано положение точки (при фиксированной точке)
ERNG	6	6	Значение мантиссы превышает ограничения дескриптора
<b>Ошибки записи атрибутов функциями modAllPermis() и modEditPermis()</b>			
EOWNER	7	7	Несанкционированная попытка редактирования Атрибутов (попытка изменить атрибут пользователем, не являющимся хозяином параметра).
EPERM	8	8	У запрошенного параметра отсутствуют признаки
<b>Стандартные ошибки, присущие протоколу обмена</b>			
AFE	0x21	33	Аппаратная ошибка кадрирования
B8E	0x22	34	Ошибка в 8-ом бите посылки
B9E	0x23	35	Ошибка в 9-ом бите посылки
SBE	0x24	36	Ошибка приема стоп-байта (стоп пришел не вовремя)
OVB	0x25	37	Ошибка переполнения буфера
ERS	0x26	38	Принят недопустимый символ
CRCE	0x27	39	Неверная контрольная сумма кадра
EDESC	0x28	40	Не найден дескриптор
NFNC	0x29	41	Не найдена сетевая функция, хотя дескриптор найден. В нормально функционирующем приборе <b>эта ошибка встречаться НЕ ДОЛЖНА!</b>
<b>Стандартные ошибки, общие для всех модулей</b>			
EDGT	0x30	48	Мантисса двоично-десятичного параметра содержит ошибку
SZE	0x31	49	Размер поля данных не соответствует ожидаемому
EASK	0x32	50	Значение бита запроса не соответствует ожидаемому
EACC	0x33	51	Редактирование параметра запрещено индивидуальным атрибутом
IDXOVF	0x34	52	Недопустимо большой линейный индекс
IDXLIM	0x35	53	Индекс параметра превышает ограничитель индекса

**Продолжение табл. В.4**

Краткое наименование	Код ошибки в старшем байте		Описание ошибки
	(hex)	(dec)	
<b>EXTROM</b>	<b>0x36</b>	<b>54</b>	Индекс параметра превышает ограничитель индекса
<b>RESERVED</b>	<b>0x37</b>	<b>55</b>	Данный код не используется
<b>ГРУППА из 8 кодов ошибок в диапазоне: LEVGRATT...LEVGRATT+L...LEVGRATT+7. Ошибка: «Запись запрещена групповым атрибутом уровня L»</b>			
<b>LEVGRATT</b>	<b>0x38</b>	<b>56</b>	Запрещающий групповой атрибут находится на уровне 0 (в корне)
<b>LEVGRATT1</b>	<b>0x39</b>	<b>57</b>	Запрещающий групповой атрибут находится на уровне 1
<b>LEVGRATT2</b>	<b>0x3A</b>	<b>58</b>	Запрещающий групповой атрибут находится на уровне 2
<b>LEVGRATT3</b>	<b>0x3B</b>	<b>59</b>	Запрещающий групповой атрибут находится на уровне 3
<b>LEVGRATT4</b>	<b>0x3C</b>	<b>60</b>	Запрещающий групповой атрибут находится на уровне 4
<b>LEVGRATT5</b>	<b>0x3D</b>	<b>61</b>	Запрещающий групповой атрибут находится на уровне 5
<b>LEVGRATT6</b>	<b>0x3E</b>	<b>62</b>	Запрещающий групповой атрибут находится на уровне 6
<b>LEVGRATT7</b>	<b>0x3F</b>	<b>63</b>	Запрещающий групповой атрибут находится на уровне 7
<b>Состояния COMMON-сегмента</b>			
<b>__LOCKSEG</b>	<b>0x41</b>	<b>65</b>	Выполняется другая задача (Сегмент COMMON занят)
<b>__FREESEG</b>	<b>0x42</b>	<b>66</b>	Задача еще не запущена (Сегмент COMMON свободен)
<b>__READYSEG</b>	<b>0x43</b>	<b>67</b>	Запрошенная задача уже выполняется
<b>__DEBUGSEG</b>	<b>0x44</b>	<b>68</b>	Программе неизвестна запрошенная функция
<b>__NOWHATCOM</b>	<b>0x45</b>	<b>69</b>	В программе стоит заглушка функции WhatCOMState()
<b>__NORUNCOM</b>	<b>0x46</b>	<b>70</b>	В программе стоит заглушка функции RunCOMTask()
	<b>0x47</b>	<b>71</b>	Недопустимое сочетание значений параметров (Изменение параметра было запрещено функцией Valid)
	<b>0x48</b>	<b>72</b>	Ошибка при чтении EEPROM
<b>Ошибки при редактировании графиков</b>			
	<b>0x49</b>	<b>73</b>	Нарушена упорядоченность узлов X по возрастанию
	<b>0x4A</b>	<b>74</b>	Попытка записи X при ненулевом числе узлов графика
	<b>0x4B</b>	<b>75</b>	Ошибка выполнения функции PrevWriteActions()

**Продолжение табл. В.4**

Краткое наименование	Код ошибки в старшем байте		Описание ошибки
	(hex)	(dec)	
<b>Ошибки мостов и ретрансляторов</b>			
<b>GATE_OVR</b>	<b>0x50</b>		Переполнение буфера моста или ретранслятора
<b>GATE_DERR</b>	<b>0x51</b>	<b>81</b>	Превышение тайм-аута ответа, потеря пакета в дочерней сети (сети, в которую ретранслируется пакет)
<b>GATE_NONET</b>	<b>0x52</b>	<b>82</b>	Запрошенная дочерняя подсеть не доступна (в случае ретрансляции в одну из нескольких дочерних подсетей)
<b>GATE_MERR</b>	<b>0x53</b>	<b>83</b>	Ответ из дочерней сети не может быть ретранслирован в материнскую сеть.

### **В.2.3. Модуль DCON (Мастер)**

В модуле **DCON (Мастер)** используется 1 поле для отображения статуса мастера и возникающих ошибок:

- «**Last error**» – содержит код ошибки из табл. В.5.

**Таблица В.5**

#### **Коды ошибок модуля DCON (Мастер)**

Краткое наименование	Код ошибки		Описание ошибки
	(hex)	(dec)	
<b>OK</b>	<b>0x0000</b>	<b>0</b>	Нет ошибок
<b>NO_DEVICE</b>	<b>0x0051</b>	<b>81</b>	Превышен таймаут ожидания ответа

Информация о статусе работы каждого отдельного устройства DCON выводится в поле «Status» универсального устройства DCON. Коды статуса и ошибок приведены в таблице В.6.

**Таблица В.6**

#### **Коды статуса и ошибок универсального устройства DCON**

Краткое наименование	Код ошибки		Описание ошибки
	(hex)	(dec)	
<b>NOT_INITIALIZED</b>	<b>0x0000</b>	<b>0</b>	Модуль универсального устройства DCON не был проинициализирован корректно
<b>REQUEST</b>	<b>0x0001</b>	<b>1</b>	Послан запрос к устройству
<b>BAD_REQUEST_FORMAT</b>	<b>0x0041</b>	<b>65</b>	Неправильный формат строки запроса
<b>BAD_REQUEST_DATA</b>	<b>0x0081</b>	<b>129</b>	Данные для запроса не соответствуют по формату строке запроса

Продолжение табл. В.6

Краткое наименование	Код ошибки		Описание ошибки
	(hex)	(dec)	
<b>NO_DEVICE</b>	<b>0x0051</b>	<b>81</b>	Превышен таймаут ожидания ответа
<b>UNIDENTIFIED_ANSVER</b>	<b>0x0021</b>	<b>33</b>	Ответ не распознан
<b>OK_ANSVER</b>	<b>0x0003</b>	<b>3</b>	Пришел ответ, соответствующий строке формата для случая положительного ответа
<b>OK_ANSVER_BAD_FORMAT</b>	<b>0x0043</b>	<b>67</b>	Неправильный формат строки разбора положительного ответа
<b>OK_ANSVER_BAD_DATA</b>	<b>0x0083</b>	<b>131</b>	Данные для разбора положительного ответа не соответствуют по формату строке разбора
<b>NEG_ANSVER</b>	<b>0x0023</b>	<b>35</b>	Пришел ответ, соответствующий строке формата для негативного ответа
<b>NEG_ANSVER_BAD_FORMAT</b>	<b>0x0063</b>	<b>99</b>	Неправильный формат строки разбора негативного ответа
<b>NEG_ANSVER_BAD_DATA</b>	<b>0x00A3</b>	<b>163</b>	Данные для разбора негативного ответа не соответствуют по формату строке разбора

**Примечание.** Поле статуса предназначено также для управления работой мастера DCON при настройке универсального устройства DCON в режиме «Опрос по команде» (**Work mode=«By command»**). Для однократного запуска опроса необходимо записать в строку статуса **0xFF**.

### В.3. Коды ошибок модуля «Архиватор»

В модуле «Архиватор» используется 1 поле для отображения статуса и возникающих ошибок:

- «**Status**» – содержит код ошибки из табл. В.7.

Таблица В.7

## Коды ошибок и статуса модуля «Архиватор»

Краткое наименование	Код ошибки		Описание ошибки
	(hex)	(dec)	
<b>STANDBY</b>	<b>0x0000</b>	<b>0</b>	Модуль архиватора остановлен
<b>RUN</b>	<b>0x0001</b>	<b>1</b>	Модуль архиватора запущен и работает без ошибок
<b>DEVICE_ERROR</b>	<b>0x0002</b>	<b>2</b>	Ошибка при записи в устройство
<b>NO_DEVICE</b>	<b>0x0004</b>	<b>4</b>	Устройство вывода отсутствует или неправильно проинициализировано

**Примечание.** При записи команды в поле «Status» архивация может быть запущена или остановлена. Коды команд даны в табл. В.8.

Таблица В.8

## Команды управления для модуля «Архиватор»

Краткое наименование	Код команды		Описание команды
	(hex)	(dec)	
<b>ARCHIVING_STARTED</b>	<b>0x00FF</b>	<b>255</b>	Запуск архивации
<b>ARCHIVING_STOPED</b>	<b>0x00FE</b>	<b>254</b>	Остановка архивации

## В.4. Коды ошибок модуля архивации информации в файл

Для подмодуля архивации информации в файл дополнительно в поле «**File Status**» выводится информация о работе подмодуля. Коды ошибок приведены в таблице В.9.

Таблица В.9

## Коды ошибок и статуса модуля архивации информации в файл

Краткое наименование	Код ошибки		Описание ошибки
	(hex)	(dec)	
<b>OK</b>	<b>0x0000</b>	<b>0</b>	Модуль функционирует нормально
<b>STOPED</b>	<b>0x0001</b>	<b>1</b>	Вывод в файл запрещен
<b>CANT_OPEN</b>	<b>0x0002</b>	<b>2</b>	Невозможно открыть файл
<b>TOO_LARGE</b>	<b>0x0003</b>	<b>3</b>	Размер записи превышает размер файла. Увеличить файл или разбить данные на несколько файлов
<b>FILE_FULL</b>	<b>0x0004</b>	<b>4</b>	Файл заполнен. Перезапись или сдвиг запрещены
<b>GENERAL_ERROR</b>	<b>0x0005</b>	<b>5</b>	Ошибка инициализации

## Приложение Г. Примеры настройки опроса переменных по протоколу DCON

В Приложении представлены примеры настройки модуля **DCON (Master)** для опроса устройств ввода/вывода.

### Г.1. Опрос значений входов модулей аналоговых входов IPC-7033

Производится опрос 3 входов нескольких модулей аналоговых входов IPC-7033.

**Описание прибора:**

**Формат команды:** #AA[CRC][CR]

где	#	–	разделитель;
	AA	–	адрес прибора;
	CRC	–	контрольная сумма;
	CR	–	перевод строки.

**Формат ответа:** >+0255.12+013.45+150.11[CRC][CR]

где	>	–	разделитель в случае положительного ответа;
	+025.12	–	данные 1 канала (5 чисел + знак + точка) итого 7 символов;
	CRC	–	контрольная сумма;
	CR	–	перевод строки.

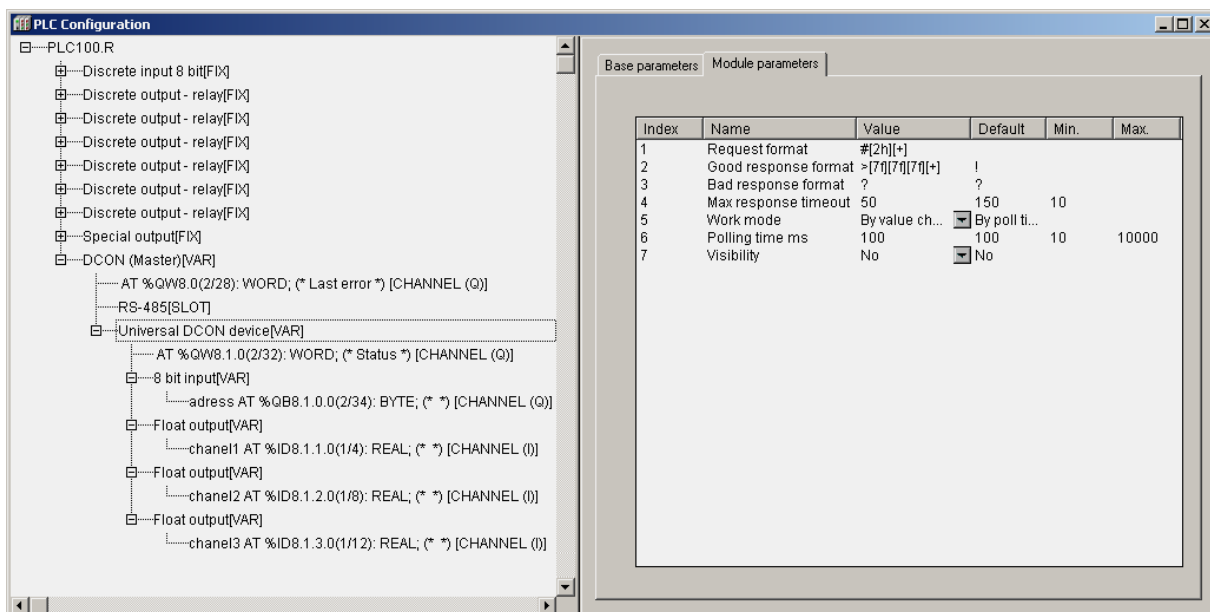
---

**Внимание.** Прибор IPC-7033 поддерживает расчет контрольной суммы только по модулю 256. Другие приборы могут поддерживать более надежный механизм подсчета контрольной суммы, как 8-битного полинома.

---

Работа модуля **DCON (Master)** с подключенным к нему устройством **Universal DCON device**, настроенным для последовательного опроса нескольких модулей **IPC-7033** с различными адресами, проиллюстрирована на рис. Г.1.





**Рис. Г. 1. Настройки устройства Universal DCON device модуля DCON (Master) для опроса входов модулей IPC-7033**

Для задания адреса опрашиваемого прибора используется 8-битная входная переменная «**Address**». Данные каналов опрашиваемого прибора отображаются в 3-х выходных переменных типа **Float (REAL)**.

Модуль **Universal DCON device** настраивается следующим образом:

**Request format** – формат строки запроса – **#[2h][+]**,

- где
- #** – символ разделителя команды опроса входов;
  - [2h]** – формат, указывающий, что в это место запроса необходимо подставить шестнадцатиричный 2-х символьный адрес, значение которого должно быть взято из 1-й входной переменной;
  - [+]** – указание на необходимость подсчета и добавления в конец запроса контрольной суммы по модулю 256.

---

**Внимание.** Символ возврата каретки вставляется автоматически!

---

**Good response format** – формат положительного ответа – **>[7f][7f][7f][+]**,

- где
- >** – символ разделителя в случае положительного ответа;
  - [7f]** – формат, указывающий на то, что 7 символов ответа должны быть преобразованы в число с плавающей точкой и результат преобразования (если на этом месте в ответе содержится число в формате **[знак]число.число.число.число**)

должен быть помещен в 1 выходную переменную, которая должна иметь формат **Float**.

**Внимание.** Для следующих [7f] применяется то же правило преобразования, но результаты помещаются во 2-ю и 3-ю выходные переменные, соответственно.

- [+] – указание на то, что должно быть проанализировано соответствие контрольной суммы принятой посылке. Результат выводится в поле статуса.
- CR** – перевод строки.

**Bad response format** – формат отрицательного ответа – **?**,  
где **?** – начальный символ строки отрицательного ответа. В рассматриваемом случае отрицательный ответ не содержит значащей информации, для его идентификации достаточно одного первого символа.

**Max response timeout** – максимальное время ожидания ответа – 50 мс.  
Задается в соответствии с рекомендациями производителя прибора.

**Work mode** – режим работы – **by change value** (по смене значения одной из входных переменных). Этот режим позволяет генерировать запросы при смене адреса опрашиваемого прибора. Для генерации 1 запроса необходимо записать значение, отличающееся от текущего, во входную переменную **«adress»**. После этого по значению поля статуса определяется окончание обмена данными с опрашиваемым прибором и ликвидность данных в выходных переменных.

---

**Внимание.** Чтение регистра статуса и его анализ должны производиться на следующем цикле работы ПЛК после записи нового адреса.

---

Остальные параметры в данном режиме работы не существенны.

## **Г.2. Запись значения выхода модуля IPC-7021 с периодичностью 1 секунда и при смене значения**

Производится запись 1 выхода у 1 прибора IPC-7021.  
Адрес прибора 18 (шестнадцатиричный).

### Описание прибора:

**Формат команды: #AA(данные)[CRC][CR]**

- где # – разделитель;  
**AA** – адрес прибора;  
**+025.12** – данные 1 канала (5 чисел + знак + точка) итого 7 символов  
**CRC** – контрольная сумма;  
**CR** – перевод строки.

**Формат ответа: ![CRC][CR]**

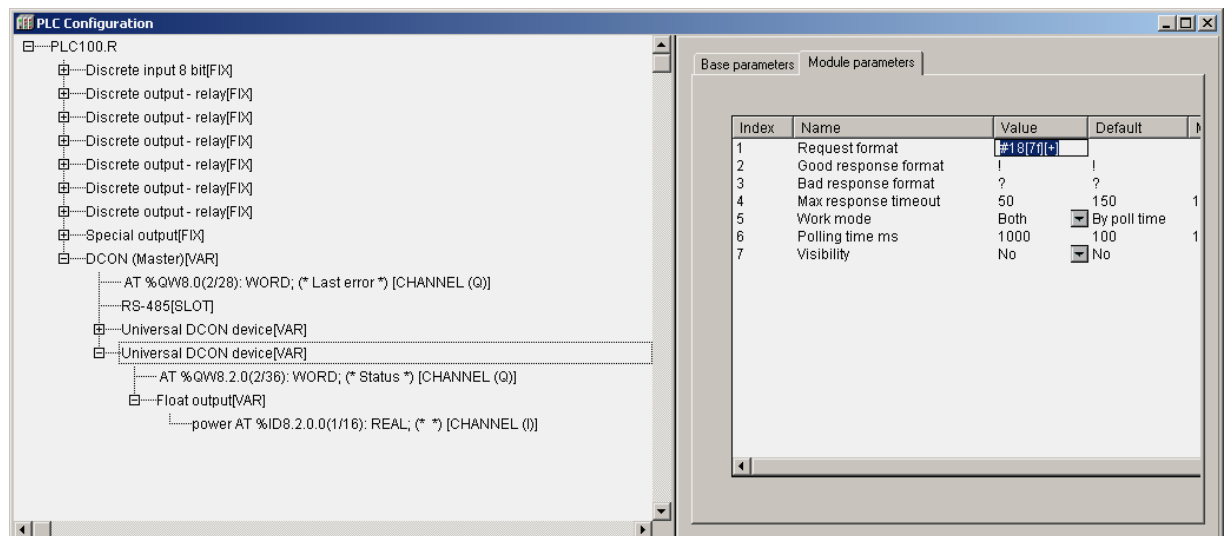
- где ! – разделитель в случае положительного ответа;  
**CRC** – контрольная сумма;  
**CR** – перевод строки.

---

**Внимание.** Прибор IPC-7021 поддерживает расчет контрольной суммы только по модулю 256. Другие приборы могут поддерживать более надежный механизм подсчета контрольной суммы, как 8-битного полинома.

---

Работа модуля **DCON (Master)** с подключенным к нему устройством **Universal DCON device**, настроенным для периодической записи значений выхода в модуль IPC-7021, проиллюстрирована на рис. Г.2.



**Рис. Г.2. Настройки устройства Universal DCON device модуля DCON (Master) для периодической записи значений выхода в модуль IPC-7021**

Данные, посылаемые в выход прибора IPC 7021, задаются во входной переменной «power» типа **Float (REAL)**.

Модуль **Universal DCON device** настраивается следующим образом:

**Request format** – формат строки запроса – **#18[7f][+]**,

- где
- #** – символ разделителя команды опроса входов;
  - 18** – адрес прибора в шестнадцатиричном формате (**для букв используется верхний регистр!**);
  - [7f]** – формат, указывающий на то, что 7 символов запроса должны быть сформированы в виде числа в формате **[знак]число..число.число..число.** Данные должны быть взяты из 1-й входной переменной, которая должна иметь формат **float**;
  - [+]** – указание на необходимость подсчета и добавления в конец запроса контрольной суммы по модулю 256.

---

**Внимание.** Символ возврата каретки вставляется автоматически!

---

**Good response format** – формат положительного ответа – **!**,

- где
- !** – начальный символ строки положительного ответа. В рассматриваемом случае положительный ответ не содержит значащей информации, для его идентификации достаточно одного первого символа.

**Bad response format** – формат отрицательного ответа – **?**,

- где
- ?** – начальный символ строки отрицательного ответа. В рассматриваемом случае отрицательный ответ не содержит информации, для его идентификации достаточно одного первого символа.

**Max response timeout** – максимальное время ожидания ответа – 50 мс.  
Задается в соответствии с рекомендациями производителя прибора.

**Work mode** – режим работы – **both** (по времени опроса и смене значения одной из входных переменных). Этот режим позволяет генерировать запросы по таймеру (параметр **Polling time**) и при смене значения выхода.

**Polling time** – время опроса – 1000 мс.

Задаёт период записи значения в выход прибора IPC-7021.