

# Конфигурирование области ввода/вывода ПЛК

## Руководство пользователя

### Содержание

Введение .....	6
Термины, сокращения и форматы данных, используемые в руководстве .....	7
Используемые термины .....	7
Используемые сокращения .....	8
Форматы данных .....	8
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.....	14
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» (Range) .....	21
2.1.5.3. Параметр «Тип энкодера» (Encoder Type) .....	22
2.1.6. Подмодуль «Счетчик специализированный» (Counter SP). ..	22
2.1.6.1. Параметры «Номер входа старта» (Start Pin Num), «Номер входа остановки» (Stop Pin Num) и «Номер входа счета» (Count Pin Num) .....	23
2.1.6.2. Параметры «Фронт сигнала на входе старта» (Start Sense Edge), «Фронт сигнала на входе остановки» (Stop Sense Edge) и «Фронт сигнала на входе счета» (Count Sense Edge)....	24
2.2. Модуль дискретного выхода (Discrete output) .....	25
2.2.1. Параметр «Безопасное состояние выхода» (Save Value) ...	26
2.2.2. Подмодуль ШИМ (Pulse-wide modulator) .....	26
2.2.2.1. Параметр «Номер выхода» (Number of output).....	27
2.2.2.2. Параметр «Период ШИМ в 100 мксек» (Period of PWM in 100 mksec).....	27

2.2.2.3. Параметр «Минимальная длительность импульса ШИМ в 100 мксек» (Minimal duration of impulse in 100 mksec) .....	27
2.2.3. Подмодуль «Генератор» (Generator) .....	28
2.2.3.1. Параметр «Инверсия сигнала» (Invertion) .....	29
2.3. Модули специальных выходов (Special output) и входов (Special input) .....	30
2.4. Модули аналоговых входов.....	32
2.4.1. Модуль «Датчик унифицированного сигнала» (Unified Signal Sensor) .....	33
2.4.1.1. Параметр «Тип датчика» (Type of sensor).....	34
2.4.1.2. Параметр «Время между измерениями в секундах» (Measure interval, s) .....	34
2.4.1.3. Параметры «Нижняя и верхняя границы» (Ain low & Ain high) .....	34
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).....	35
2.4.2. Модуль «Датчик типа «Термопара»» (Thermocouple sensor).....	35
2.4.2.1. Параметр «Компенсация холодного спая» (Cold junction compensation) .....	37
2.4.3. Модуль «Датчик типа «Термосопротивление»» (RTD sensor) .....	37
2.4.4. Модуль «Контактный датчик» (Contact sensor) .....	39
2.5. Модуль аналоговых выходов (Analog output) .....	41
2.5.1. Параметр «Тип выхода» (Type) .....	42
2.5.2. Параметры «Коррекция в начальной точке» (Null correction) и «Коррекция в конечной точке диапазона» (Full range correction).....	42
3. Описание добавляемых модулей Конфигуратора ПЛК .....	43
3.1. Модуль Modbus (Slave).....	43
3.1.1. Параметр «Адрес устройства» (Address) .....	44
3.1.2. Подмодуль Modbus (FIX). Настройка коммуникаций модуля .....	44
3.1.2.1. Параметр «Скорость передачи информации» (CommunicationSpeed, bod) .....	48
3.1.2.2. Параметр «Проверка четности» (Parity) .....	48
3.1.2.3. Параметр «Количество бит данных» (DataBits) .....	48
3.1.2.4. Параметр «Количество стоп-битов» (StopLength) .....	48
3.1.2.5. Параметр «Тип интерфейса» (InterfaceType).....	48
3.1.2.6. Параметр «Тип протокола обмена» (FrameOriented) ...	48
3.1.2.7. Параметр «Время, на которое необходимо задерживать ответ на запрос в мс» (Framing time) .....	49
3.1.2.8. Параметр «Удаленный порт» (RemotePort) .....	49
3.1.3. Настройка данных информационного обмена .....	49
3.1.3.1. Выравнивание переменных при размещении в области памяти.....	51

3.1.3.2. Отображение устройства в области памяти.....	52
3.2. Модуль Modbus (Master) .....	54
3.2.1. Модуль «Универсальное устройство Modbus (Master)» ....	55
3.2.1.1. Параметр «IP-адрес» (Module IP) .....	57
3.2.1.2. Параметр «Максимальный тайм-аут, в мс» (Max timeout).....	57
3.2.1.3. Параметр «Порт TCP» (TCP port) .....	57
3.2.1.4. Параметр «Режим работы в сети» (NetMode) .....	57
3.2.1.5. Параметр «Адрес подчиненного устройства» (ModuleSlaveAddress) .....	58
3.2.1.6. Параметр «Режим работы» (Work mode).....	58
3.2.1.7. Параметр «Число повторов» (Amount Repeat).....	58
3.2.1.8. Параметр «Порядок передачи байтов посылки» (Byte Sequence).....	59
3.2.1.9. Параметр «Период опроса устройства, в мс» (Polling time) .....	59
3.2.2. Настройка входов и выходов универсального устройства Modbus (Master) .....	59
3.2.2.1. Параметр «Номер команды протокола» (Command) ...	60
3.2.2.2. Параметр «Адрес регистра опрашиваемого устройства» (Start Address) .....	60
3.2.2.3. Параметр «Размер переменной» (Amount byte) .....	60
3.2.3. Шаблоны применяемых устройств.....	60
3.3. Модуль «Архиватор» (Archiver).....	62
3.3.1. Параметр «Режим проведения архивации» (Archive Mode) 64	
3.3.2. Параметр «Тип архива» (Archive Type).....	65
3.3.3. Параметр «Период архивации, сек» (Period of Archiving) ..	65
3.3.4. Параметр «Имя архива» (Archive Name) .....	65
3.3.5. Параметр «Комментарий архива» (Comment) .....	65
3.3.6. Параметры «Время начала архивации» (Start time) и «Время остановки архивации» (Stop time) .....	65
3.3.7. Параметры «Понедельник» (On Sunday) ... «Воскресенье» (On Saturday) .....	66
3.3.8. Подмодуль интерфейсного порта (File Output) .....	66
3.3.8.1. Параметр «Имя файла» (File name).....	67
3.3.8.2. Параметр «Режим работы модуля» (Mode) .....	68
3.3.8.3. Параметр «Тип» (Type).....	68
3.3.8.4. Параметр «Максимальный размер файла» (Max file size) .....	68
3.4. Модуль «Owen (Slave)» .....	69
3.4.1. Параметр «Имя прибора» (Slave Name) .....	70
3.4.2. Параметр «Длина адреса устройства» (Address Length)...	70
3.4.3. Параметр «Адрес устройства» (Address) .....	70
3.4.4. Подмодуль Owen (FIX). Настройка коммуникаций модуля	71
3.4.5. Переменные протокола ОВЕН .....	71
3.4.5.1. Параметр «Сетевое имя переменной» (Hash name) ....	73

3.4.5.2. Параметры «Индекс прибора» (Index) и «Использовать индекс?» (Use a index?) .....	73
3.4.5.3. Параметр «Тип числа с плавающей точкой» (Float Type) .....	73
3.4.5.4. Параметр «Точность» (Precision).....	73
3.5. Модуль «Owen (Spy)» .....	74
3.5.1. Переменные протокола ОВЕН для Owen (Spy) .....	74
3.5.1.1. Параметр «Длина адреса» (Address Length) .....	75
3.5.1.2. Параметр «Адрес устройства» (Address) .....	75
3.6. Модуль «Owen (Master)» .....	76
3.6.1. Переменные протокола ОВЕН для Owen (Master) .....	77
3.6.1.1. Параметр «Максимальное время задержки ответа, мс» (Max response delay, ms) .....	80
3.6.1.2. Параметр «Режим работы» (Work mode).....	80
3.7. Модуль «DCON (Master)».....	81
3.7.1. Модуль «Универсальное устройство DCON» (Universal DCON Device) .....	82
3.7.1.1. Параметры «Формат запроса» (Request format), «Формат правильного ответа» (Good response format) и «Формат неправильного ответа» (Bad response format) .....	83
3.7.1.2. Параметр «Максимальное время ответа» (Max response timeout).....	86
3.7.1.3. Параметр «Режим работы» (Work mode).....	86
3.8. Модуль статистики (Statistic) .....	88
3.9. Модуль «Константа» (Constant value) .....	89
3.9.1. Параметр «Значение константы» (Constant Value) .....	89
3.9.2. Параметр «Комментарий для программы EasyWorkPLC» (Comment for EasyWorkPLC).....	90
<b>Приложение А. Типы датчиков, поддерживаемые аналоговыми входами ОВЕН ПЛК .....</b>	<b>91</b>
<b>А.1. Подключение датчиков типа ТСМ50, ТСП100 и ТСМ100 к ПЛК150 .....</b>	<b>91</b>
<b>А.2. Перечень поддерживаемых датчиков .....</b>	<b>91</b>
<b>Приложение Б. Примеры настройки опроса переменных по протоколу ОВЕН .....</b>	<b>93</b>
<b>Приложение В. Сообщения об ошибках в ПЛК .....</b>	<b>97</b>
<b>В.1. Ошибки датчиков (аналоговых входов) .....</b>	<b>97</b>
<b>В.2. Коды ошибок модулей «Мастер».....</b>	<b>97</b>
<b>В.2.1. Модуль Modbus (Мастер) .....</b>	<b>97</b>
<b>В.2.2. Модуль ОВЕН (Мастер) .....</b>	<b>98</b>
<b>В.2.3. Модуль DCON (Мастер) .....</b>	<b>101</b>
<b>В.3. Коды ошибок модуля «Архиватор» .....</b>	<b>102</b>
<b>В.4. Коды ошибок модуля архивирования информации в файл ...</b>	<b>103</b>
<b>Приложение Г. Примеры настройки опроса по протоколу DCON в режиме Master .....</b>	<b>104</b>

<b>Г.1. Опрос модулей аналоговых входов IPC-7033 .....</b>	<b>104</b>
<b>Г.2. Установка выходного значения модуля аналогового вывода IPC-7021 с периодичностью 1 секунда и при необходимости изменения значения .....</b>	<b>106</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 конфигурации, описывающий переменную в памяти ввода/вывода, через которую модуль, содержащий канал, взаимодействует с программой ПЛК. Переменная канала может быть типа BOOL, BYTE, WORD, DWORD, FLOAT, STRING. Канал не имеет параметров.

**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 (модуль)** – основной конфигурируемый элемент, подключаемый в PLC-Configuration. Модуль соответствует единице аппаратных/программных средств, с которым работает контроллер.

**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**

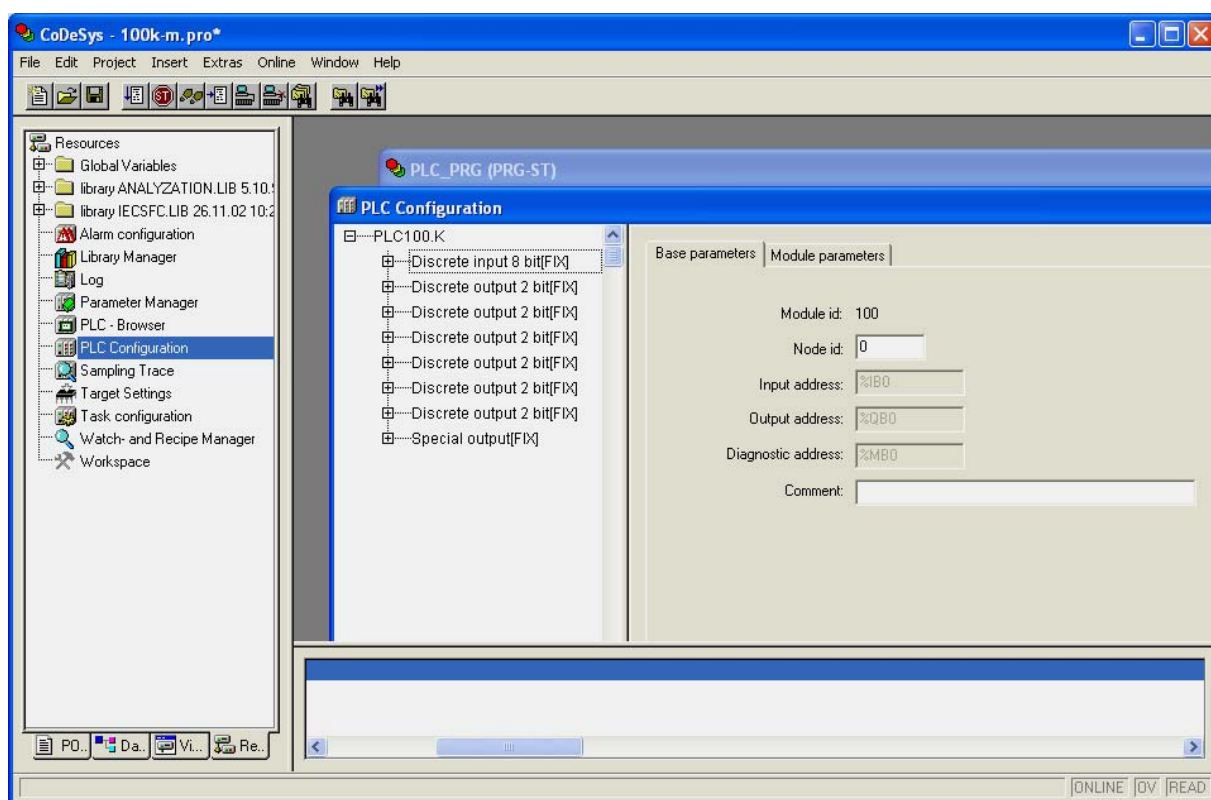
<b>Тип</b>	<b>Нижний предел</b>	<b>Верхний предел</b>	<b>Размер памяти</b>
<b>BYTE:</b>	0	255	8 Бит
<b>WORD:</b>	0	65535	16 Бит
<b>DWORD:</b>	0	4294967295	32 Бита
<b>SINT:</b>	-128	127	8 Бит
<b>USINT:</b>	0	255	8 Бит
<b>INT:</b>	-32768	32767	16 Бита
<b>UINT:</b>	0	65535	16 Бита
<b>DINT:</b>	-2147483648	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

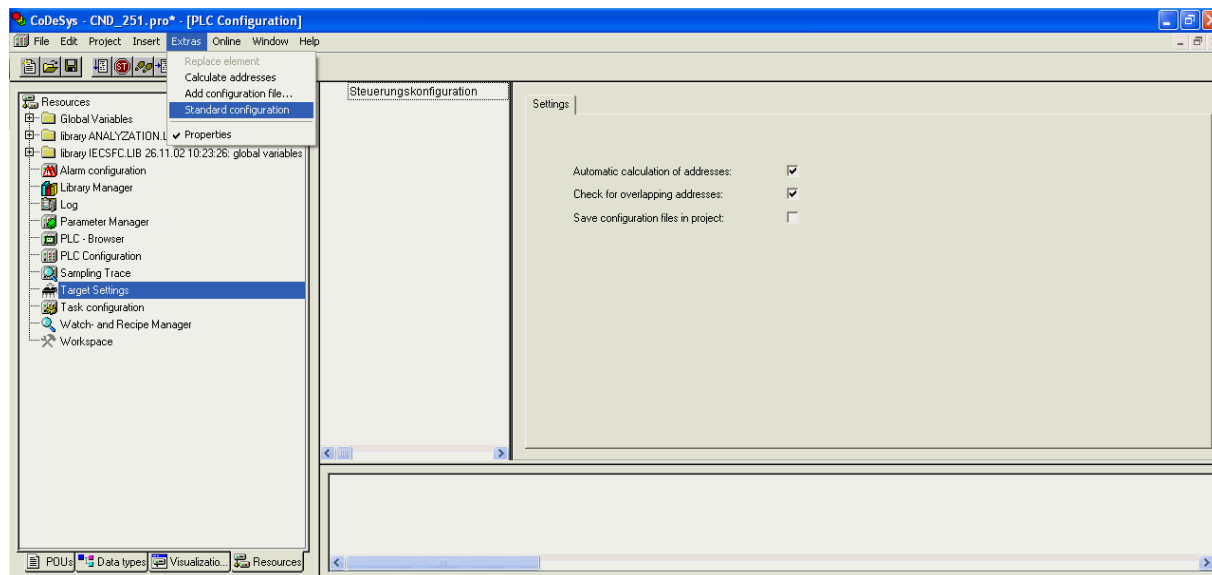
---

**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 0 до 50 мс, значение по умолчанию – 1 мс.

**Примечание.** Программная реализация ПЛК обеспечивает вызов цикла ПЛК не чаще, чем 1 раз в установленное число миллисекунд. В случае, если цикл ПЛК выполняется за время, превышающее установленное значение, его выполнение будет происходить реже. При установке значения периода цикла ПЛК, равного нулю, отключается контроль периода цикла, и цикл ПЛК выполняется с максимально возможной частотой.

**Внимание.** В режиме работы ПЛК с максимально возможной частотой не гарантируется стабильная частота выполнения циклов ПЛК, которая может изменяться в зависимости от интенсивности обмена по сети, отладки, записи данных во Flash и т.д. В этом режиме рекомендуется выдерживать временные интервалы в программе ПЛК с использованием таймера высокого разрешения библиотеки **SysLibTime** (разрешение 100 мкс).

**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.

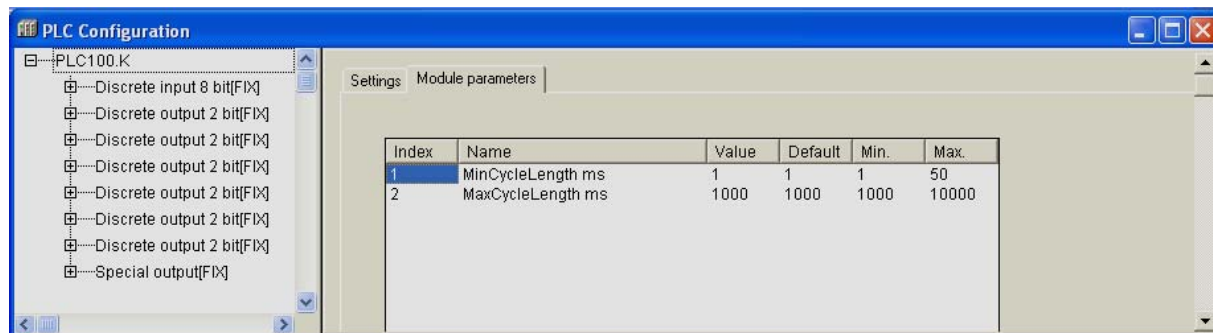


Рис. 1.3. Окно настройки параметров работы ПЛК

## 1.1. Типы модулей в Конфигурации

Существует два вида модулей:

- **фиксированные** – жестко задаются и не могут быть удалены или заменены. Допускается только редактирование их параметров;
- **добавляемые** – вставляются по желанию пользователя во время конфигурирования, подразделяются на два типа:

- **тип SLOT** – означает, что зарезервировано место для модуля, которое может быть занято или оставлено пустым. На одно зарезервированное место может быть установлен один 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. добавить можно лишь определенные типы модулей.

Добавление модулей 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.

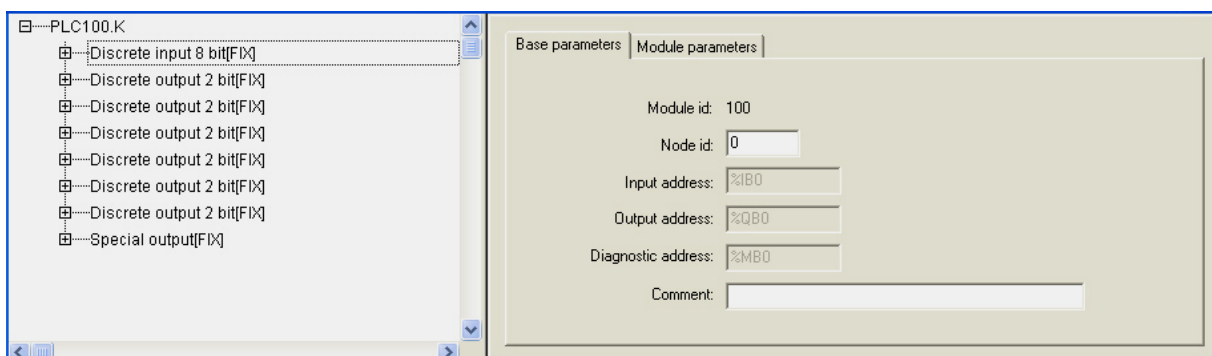
## 1.2. Представление модулей в экранной форме Конфигуратора ПЛК

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 окна конфигурирования модуля, с вкладкой базовых параметров в правой части экранной формы.

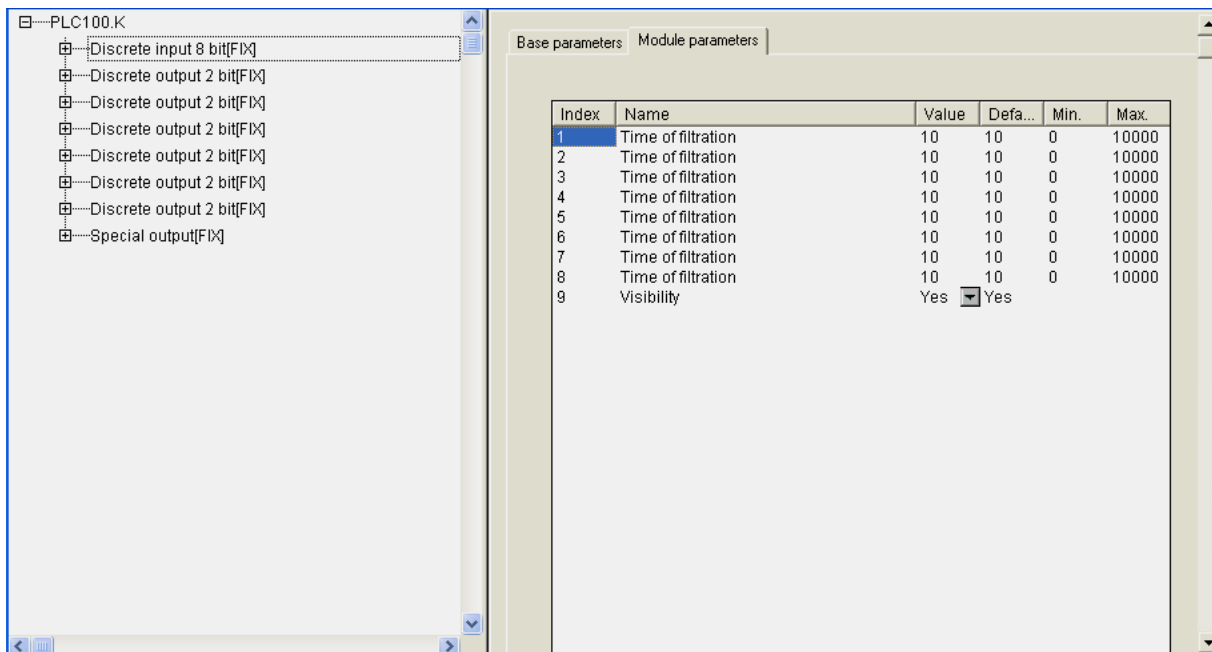


**Рис. 1.4. Окно конфигурирования модуля дискретных входов. Вкладка «Базовые параметры»**

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 – минимальные и максимальные – значения опциональные и не всегда присутствуют во вкладках параметров модулей.

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



**Рис. 1.5. Окно конфигурирования модуля 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.

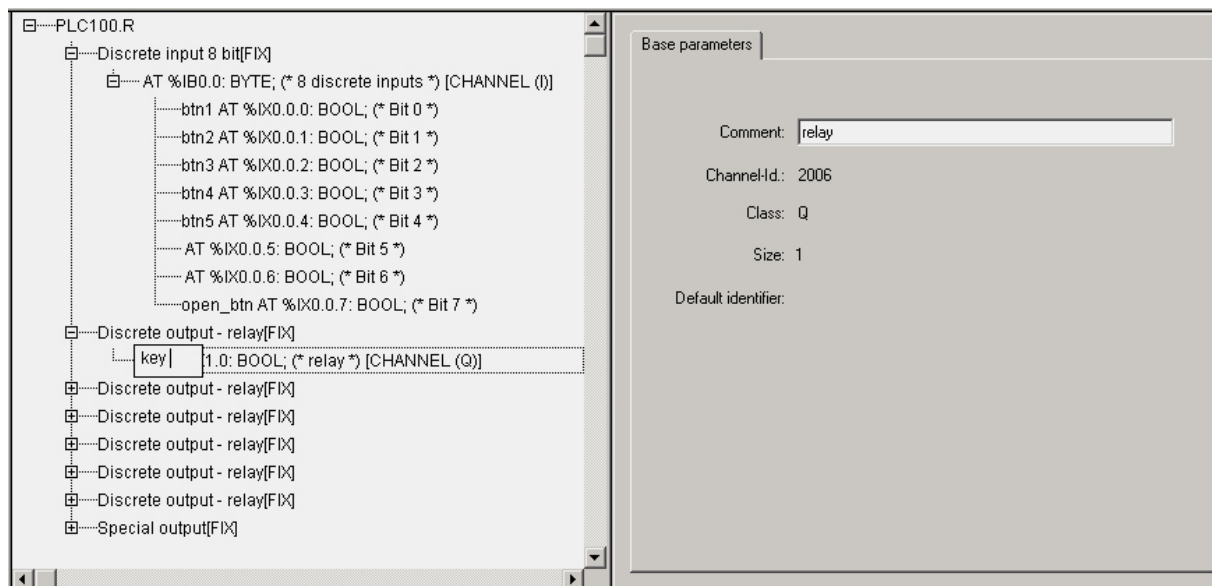
---

#### Примечания.

1. Имя может состоять 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, имеющиеся 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.

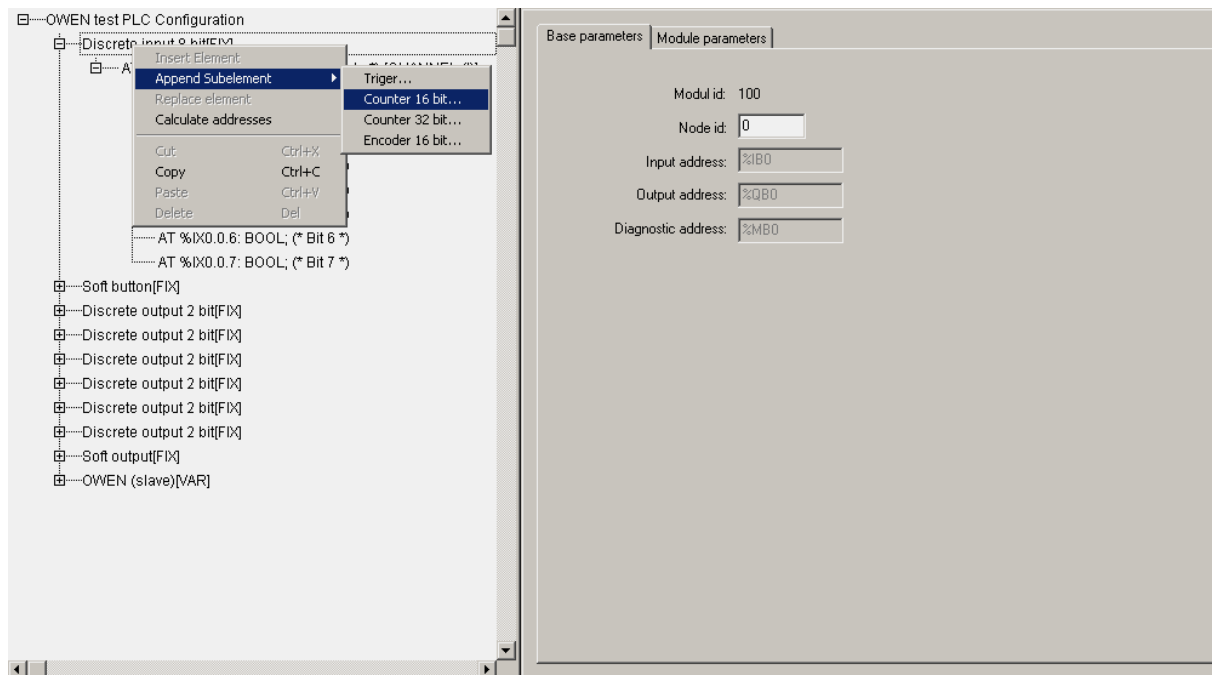


Рис. 1.7. Вызов подчиненного подмодуля

## 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 ввода/вывода значения, характеризующие состояния дискретных входов ПЛК. 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 – «**no**» (см. п. 2.1.2).

#### Список подмодулей:

**Триггер (Trigger)** (см. п. 2.1.3), **счетчик (Counter)** (см. п. 2.1.4), **энкодер (Encoder)** (см. п. 2.1.5) и **счетчик специализированный» (Counter SP)** (см. п. 2.1.6).

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

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

Фильтрация применяется главным образом для подавления «дребезга» контактов.

**Время фильтрации** – это период опроса состояния одного дискретного входа, задается в сотнях микросекунд (1 ед. = 100 мкс, 10 ед. = 1 мс).

Принцип действия фильтрации:

- в сдвиговом регистре в драйвере каждого дискретного входа накапливаются значения восьми последних состояний, полученных в



результате опроса с периодом, заданным в параметре «Время фильтрации»;

- если состояние битового канала дискретного входа равно **1 (TRUE)**, а количество единиц в сдвиговом регистре менее двух, то битовый канал переключается на **0 (FALSE)**;
- если состояние битового канала равно **0 (FALSE)**, а количество единиц в сдвиговом регистре больше пяти, то битовый канал переключается на **1 (TRUE)**;
- если количество единиц в сдвиговом регистре от 2 до 5, то состояние битового канала дискретного входа не меняется.

Режим фильтрации может быть отключен установлением в параметре значения, равного «-1». Отключение фильтрации необходимо при работе с подчиненными модулями энкодеров для того, чтобы не пропускать высокочастотные сигналы, а также в тех случаях, когда ПЛК функционирует без ограничения цикла по частоте, т.е. на максимально возможной частоте.

## 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**.

## 2.1.3. Подмодуль «Триггер» (Trigger)

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

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

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).

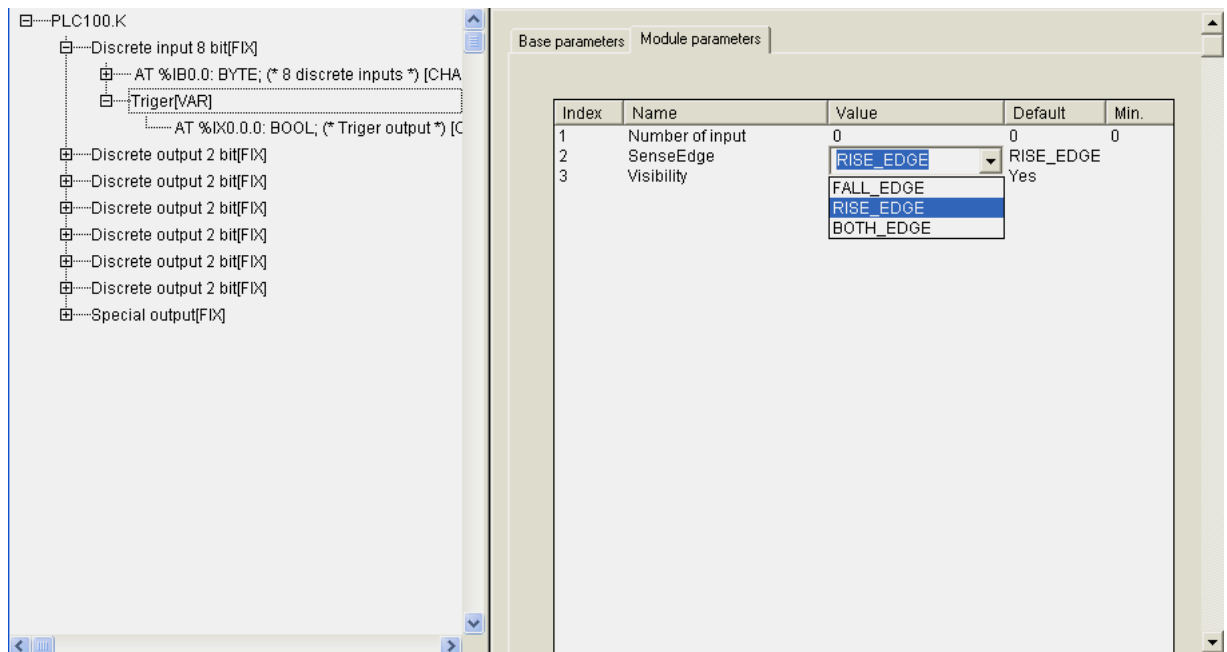
«**Видимость**» (**Visibility**) – задает видимость параметров модуля в программе EasyWorkPLC. Значения выбираются из списка «**yes**» и «**no**», значение по умолчанию – «**no**» (см. п. 2.1.2).

Окно конфигурирования модуля «**Триггер**» (**Trigger**) представлено 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. Окно конфигурирования модуля «Триггер»**

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

Параметр **«Номер входа»** – указывает, какой дискретный вход контроллера (считая от 0) 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 ОВЕН ПЛК запрещают работу в режиме BOTH\_EDGE.

---

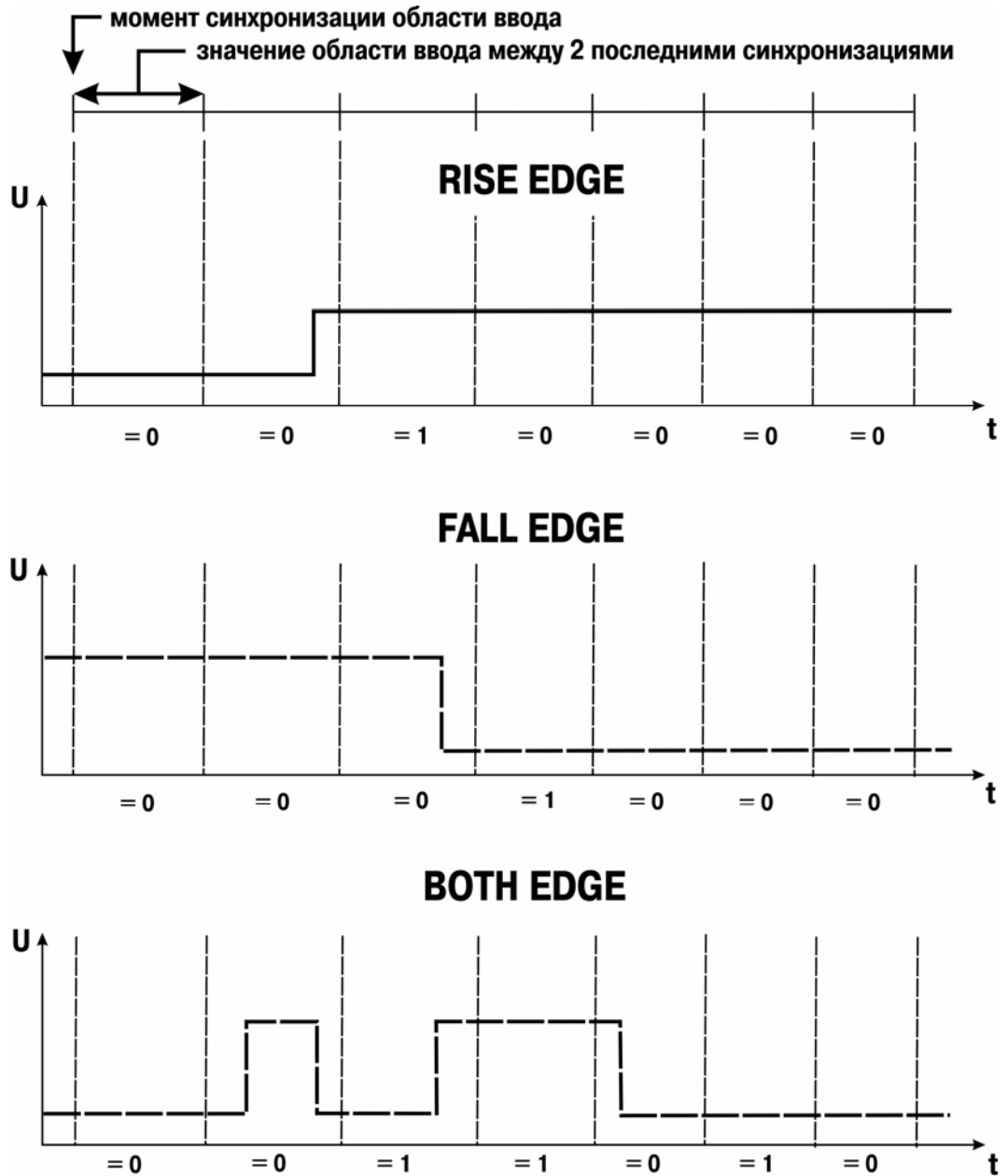
---

**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 изменения сигнала на входе и соответствующее состояние триггера**

## 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 – **«no»** (см. п. 2.1.2).

Счетчик считает количество импульсов, пришедших на дискретный вход за один цикл ПЛК. По завершению каждого цикла ПЛК 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 контролируемого механизма с последующей передачей информации в цифровой форме в программу ПЛК. Для работы с высокочастотными энкодерами необходимо отключать режим фильтрации по входу, подробнее см. п. 2.1.1.

Модуль **«Энкодер» (Encoder)** является подчиненным подмодулем модуля дискретных входов.

Модуль имеет 16-ти битовый канал (формат WORD).

### 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» (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 – «no» (см. п. 2.1.2).

Окно конфигурирования модуля «Энкодер» (**Encoder**) представлено на рис. 2.3.

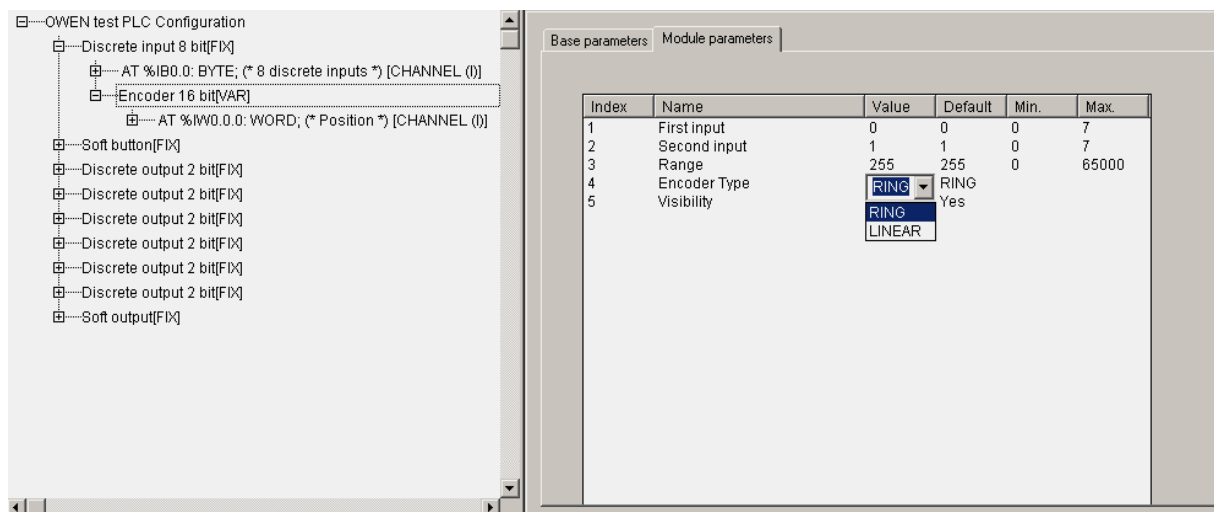


Рис. 2.3. Окно конфигурирования модуля «Энкодер»

### 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» (*Range*)

Параметр «Диапазон» (**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» (Range).

### 2.1.6. Подмодуль «Счетчик специализированный» (Counter SP)

Счетчик специализированный – программный модуль, осуществляющий обработку цифровых входов и управляемый специальным образом. Его отличие от **Счетчика (Counter 16bit)**, подключаемого к какому-либо входу, заключается в том, что **Счетчик специализированный** дополнительно использует два аппаратных входа (которые задаются через параметры счетчика) для того, чтобы извне возможно было управлять стартом и остановкой функционирования счетчика.

Модуль «**Счетчик специализированный**» (Counter SP) является подчиненным подмодулем модуля дискретных входов.

Модуль имеет 16-ти битовый канал (формат WORD).

#### Параметры модуля:

«**Номер входа старта**» (Start Pin Num) – диапазон значений от 0 до 5, значение по умолчанию – 5 (см. п. 2.1.6.1).

«**Номер входа остановки**» (Stop Pin Num) – диапазон значений от 0 до 6, значение по умолчанию – 6 (см. п. 2.1.6.1).

«**Номер входа счета**» (Count Pin Num) – диапазон значений от 0 до 7, значение по умолчанию – 7 (см. п. 2.1.6.1).

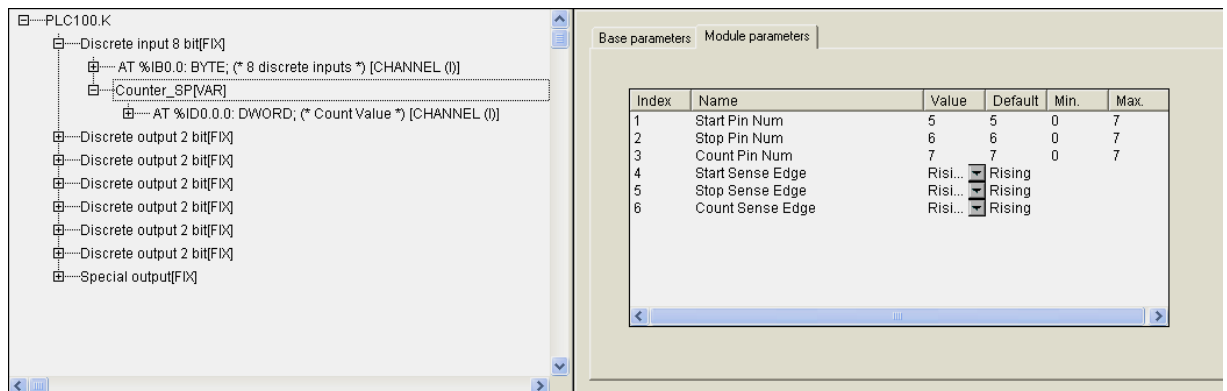
«**Фронт сигнала на входе старта**» (Start Sense Edge) – значения выбираются из списка «**RISING\_EDGE**», «**FALL\_EDGE**» и «**BOTH\_EDGE**», значение по умолчанию – «**RISING\_EDGE**» (см. п. 2.1.6.2).

«**Фронт сигнала на входе остановки**» (Stop Sense Edge) – значения выбираются из списка «**RISING\_EDGE**», «**FALL\_EDGE**» и «**BOTH\_EDGE**», значение по умолчанию – «**RISING\_EDGE**» (см. п. 2.1.6.2).

«Фронт сигнала на входе счета» (**Count Sense Edge**) – значения выбираются из списка «**RISING\_EDGE**», «**FALL\_EDGE**» и «**BOTH\_EDGE**», значение по умолчанию – «**RISING\_EDGE**» (см. п. 2.1.6.2).

Специализированный Счетчик считает количество импульсов, пришедших на дискретный вход. В канале Специализированного Счетчика значение не обнуляется при каждом цикле ПЛК (в отличие от модуля Счетчика), а накапливается. Обнуление данных в канале производится при каждом новом старте.

Окно конфигурирования модуля «**Счетчик специализированный**» (**Counter SP**) представлено на рис. 2.4.



**Рис. 2.4. Окно конфигурирования модуля «Счетчик специализированный»**

### **2.1.6.1. Параметры «Номер входа старта» (Start Pin Num), «Номер входа остановки» (Stop Pin Num) и «Номер входа счета» (Count Pin Num)**

Параметры «**Номер входа старта**» (**Start Pin Num**) и «**Номер входа остановки**» (**Stop Pin Num**) определяют номера входов ПЛК, через которые осуществляется управление стартом и остановкой функционирования счетчика.

Параметр «**Номер входа счета**» (**Count Pin Num**) определяет номер входа ПЛК, на котором производится накопление (суммирование) импульсов, учтенных счетчиком.

### **2.1.6.2. Параметры «Фронт сигнала на входе старта» (Start Sense Edge), «Фронт сигнала на входе остановки» (Stop Sense Edge) и «Фронт сигнала на входе счета» (Count Sense Edge)**

Параметры «Фронт сигнала на входе старта» (Start Sense Edge), «Фронт сигнала на входе остановки» (Stop Sense Edge) и «Фронт сигнала на входе счета» (Count Sense Edge) определяют фронты сигналов, по которым будет осуществляться работа (см. п. 2.1.3.2) – старт и остановка счета и собственно счет импульсов.

---

**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**.

---



## 2.2. Модуль дискретного выхода (Discrete output)

Модуль дискретного выхода (**Discrete output**) отображает в области памяти ввода/вывода значения дискретного выхода ПЛК.

Модуль имеет несколько битовых каналов (количество каналов зависит от варианта исполнения контроллера).

### 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 – «**no**» (см. п. 2.1.2).

### Список подмодулей:

**ШИМ (Pulse-wide modulator)** (см. п. 2.2.2) и **генератор (Generator)** (см. п. 2.2.3).

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. 2.5 представлены варианты модуля дискретного выхода: содержащий один битовый канал и четыре битовых канала.

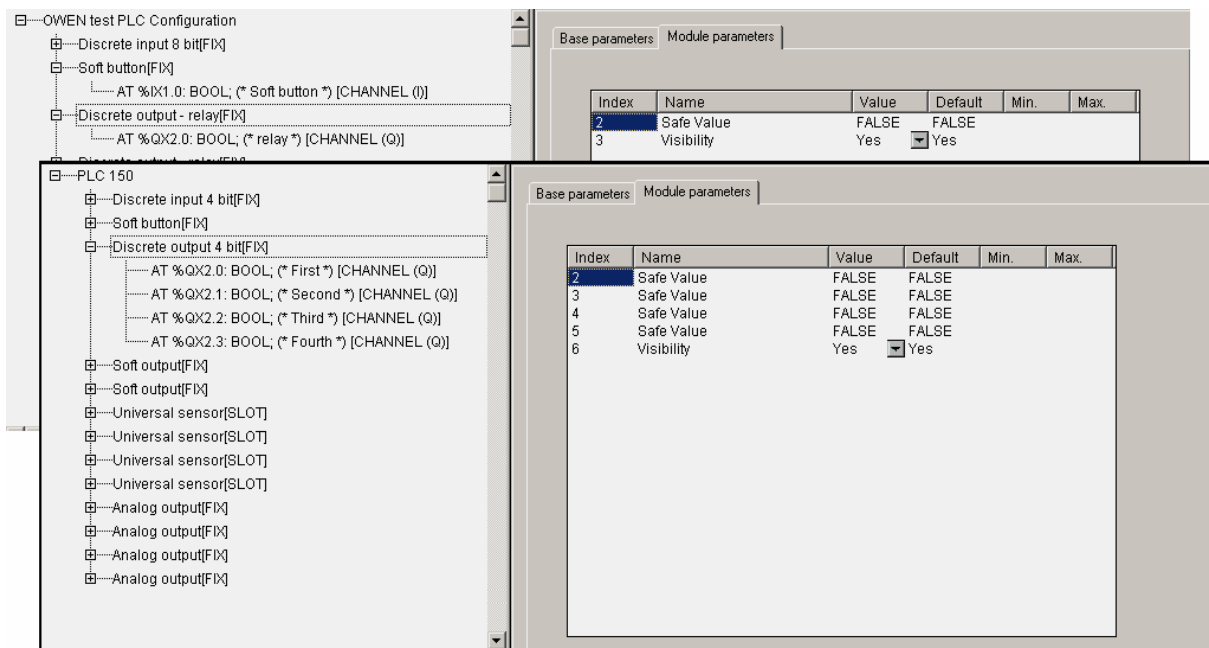


Рис. 2.5. Окно конфигурирования модуля дискретного выхода

---

**Примечание.** В вариантах модуля дискретных входов, содержащих два битовых и четыре битовых канала, значение параметра «**Безопасное состояние выхода**» (**Save Value**) устанавливается для каждого битового канала.

---

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

Назначение параметра «**Безопасное состояние выхода**» следующее.

В момент загрузки ПЛК или при каком-либо серьезном сбое, например при «зависании» ПЛК и т.п., его выходы могут оказаться в неопределенном состоянии: выключены или включены. Однако, это состояние может оказаться недопустимым при эксплуатации управляемого оборудования. Для исключения такой ситуации ПЛК переводит выходы при «зависании» или во время загрузки в состояние, заданное в параметре «**Безопасное состояние выхода**» (**Save Value**). Значения параметра: **FALSE** означает, что выход выключен (=0), **TRUE** – выход включён (=1).

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

Модуль ШИМ – программный модуль, предназначенный для обеспечения функционирования генератора **широотно-импульсной модуляции**, подключенного к дискретному выходу.

Модуль **ШИМ (Pulse-wide modulator)** является подчиненным подмодулем модуля дискретного выхода.

Модуль имеет 16-ти битовый канал (формат WORD).

#### **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, к которому осуществляется подключение.

---

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

«**Минимальная длительность импульса ШИМ в 100 мксек**» (**Minimal duration of impulse in 100 mksec**) – диапазон значений от 1 до 65000, значение по умолчанию – 30 ед. (см. п. 2.2.2.3).

---

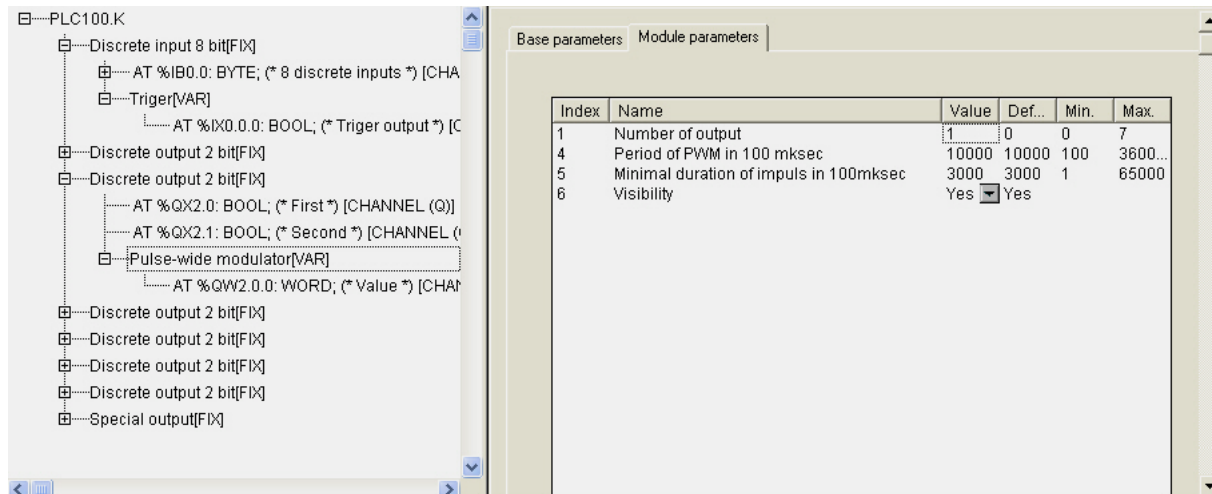
**Примечание.** 1 ед. = 100 мкс.

---

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

Окно конфигурирования модуля **ШИМ (Pulse-wide modulator)** представлено на рис. 2.6.



**Рис. 2.6. Окно конфигурирования модуля ШИМ**

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

В параметре «**Номер выхода**» (**Number of output**) в модуле **ШИМ** – задаётся номер выхода ПЛК, к которому подключен ШИМ-генератор.

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

В параметре «**Период ШИМ в 100 мкс**» (**Period of PWM in 100 mksec**) – задаётся длительность одного периода ШИМ-регулирования. Принимает значения от 100 до 360000 ед., соответственно задавая период ШИМ от 10 миллисекунд до 36 секунд.

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

В параметре «**Минимальная длительность импульса ШИМ в 100 мкс**» (**Minimal duration of impulse in 100 mksec**) устанавливается ограничение на минимальную длительность импульса ШИМ.

### 2.2.3. Подмодуль «Генератор» (**Generator**)

Модуль «**Генератор**» (**Generator**) – программный модуль, расширяющий функционал дискретного выхода и осуществляющий за счет аппаратного обеспечения процессора функционирование выхода в режиме высокоточного и высокоскоростного генератора с изменяемой скважностью.

---

**Внимание.** Модуль «**Генератор**» (**Generator**) может быть применен и в качестве ШИМ-генератора за счет изменения скважности.

**Примечание.** Особенности аппаратной реализации ОВЕН ПЛК позволяют осуществлять генерирование сигнала только для ПЛК100-К и только через последний, двенадцатый выход. В других модификациях ОВЕН ПЛК модуль «**Генератор**» (**Generator**) не реализован. При этом выход ПЛК100-К способен выдавать импульсы, как единичные, так и нулевые, длительностью не менее 30мкс.

---

Модуль «**Генератор**» (**Generator**) является подчиненным подмодулем модуля дискретного выхода.

Модуль «**Генератор**» имеет **3 канала** вывода, в которых записываются значения:

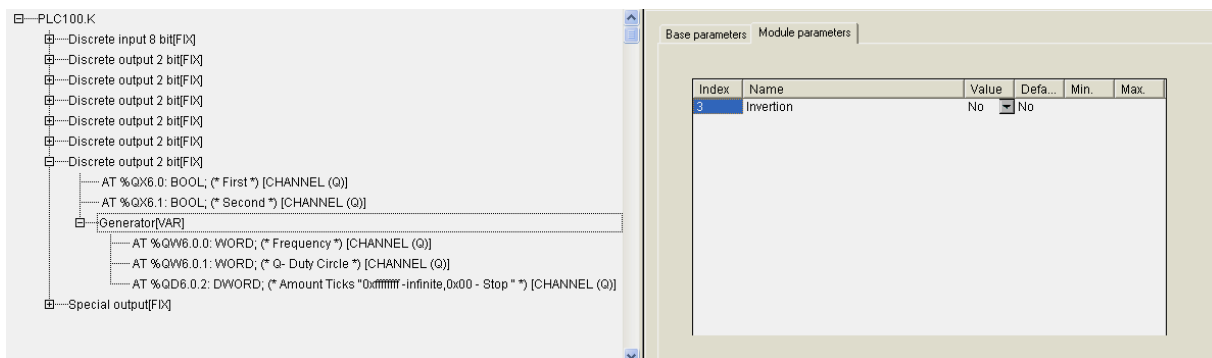
- частоты («**Frequency**») - от 1 до 10 КГц,
- скважности («**Q-Duty Circle**») – от 0 до 999 (от 0 до 99.9% с точностью до 0.1%);
- число импульсов/состояние модуля («**Amount Ticks**»).

В последний канал записывается число импульсов, которое требуется сгенерировать. При записи числа 0x0 генерация импульсов прекращается, при записи числа 0xffffffff происходит бесконечно.

#### **Параметры модуля:**

«**Инверсия сигнала**» (**Inverction**) – значения выбираются из списка «**yes**» и «**no**», значение по умолчанию – «**no**» (см. п. 2.2.3.1).

Окно конфигурирования модуля «**Генератор**» (**Generator**) представлено на рис. 2.7.



**Рис. 2.7. Окно конфигурирования модуля «Генератор»**

### **2.2.3.1. Параметр «Инверсия сигнала» (Inversion)**

В параметре **«Инверсия сигнала» (Inversion)** в модуле **«Генератор» (Generator)** определяется необходимость инверсии сигнала на выходе модуля.

## 2.3. Модули специальных выходов (**Special output**) и входов (**Special input**)

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

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 – «**no**» (см. п. 2.1.2).

---

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

---

Модуль специального дискретного выхода (**Special output**) – модуль, содержащий битовую переменную, управляющую специальным оборудованием (устройством подачи звукового сигнала или светодиодом на верхней панели). При значении переменной – **TRUE** включается специальное оборудование.

Модуль имеет битовый канал.

### Параметры модуля:

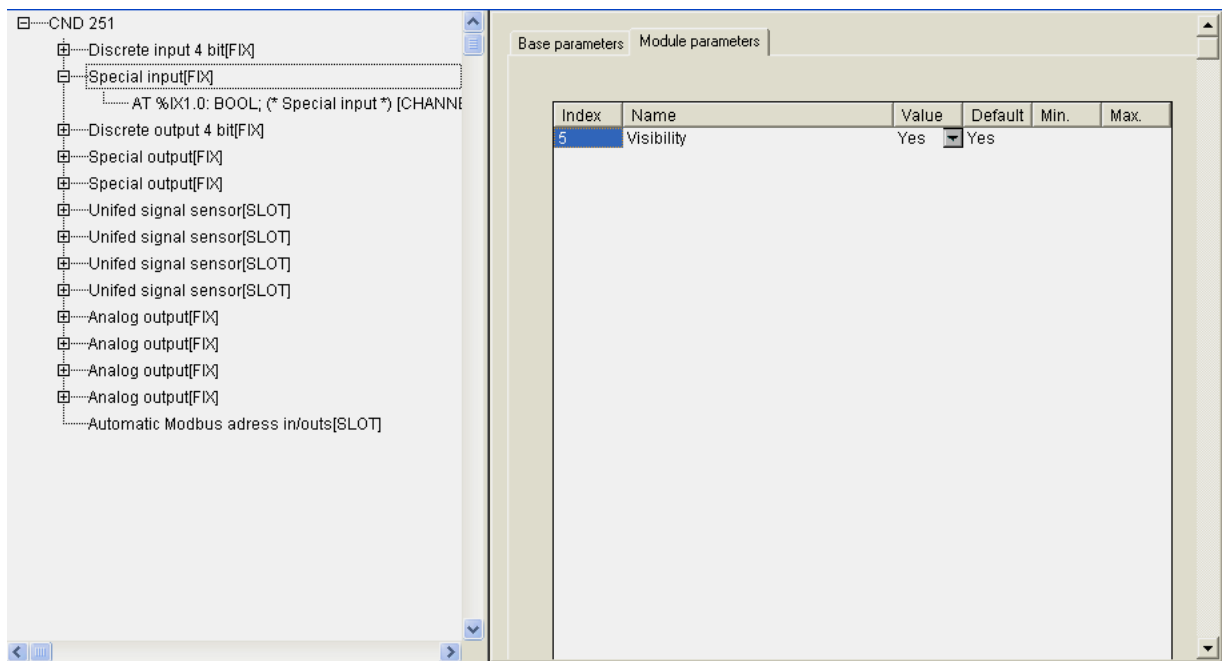
«Видимость» (**Visibility**) – задает видимость параметров модуля в программе EasyWorkPLC. Значения выбираются из списка «**yes**» и «**no**», значение по умолчанию – «**no**» (см. п. 2.1.2).

---

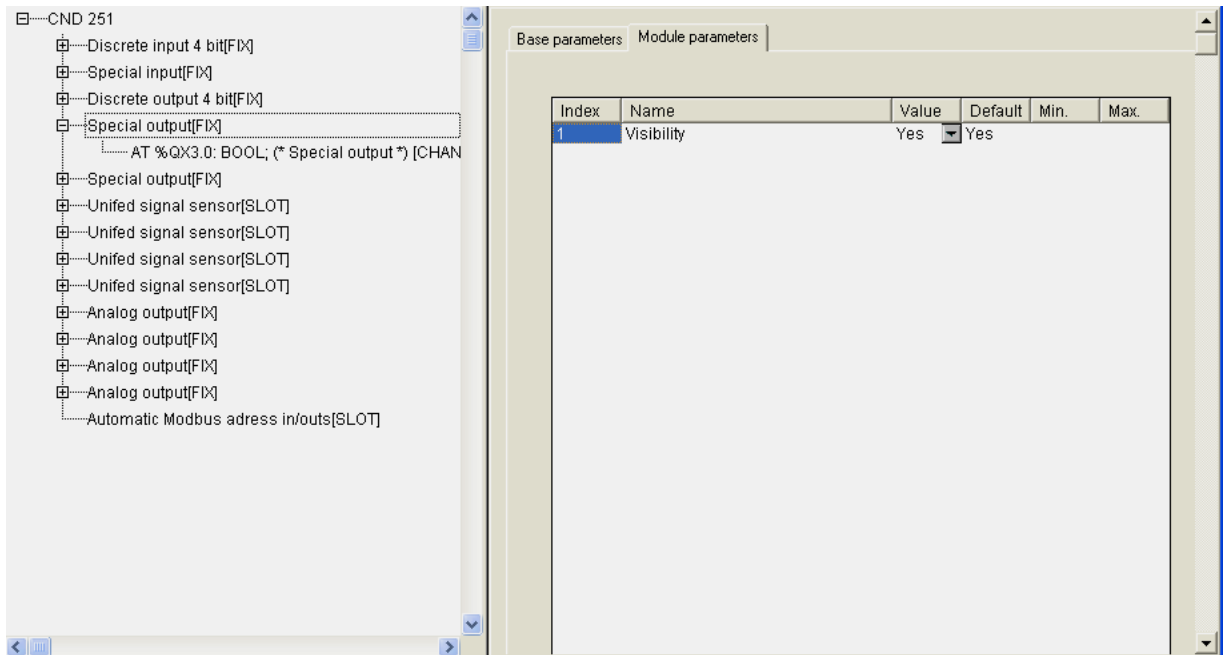
**Примечание.** В зависимости от модели ПЛК оснащён устройством подачи звукового сигнал и/или светодиодом.

---

Окна конфигурирования модулей специального дискретного выхода (**Special output**) и дискретного входа (**Special input**) отображены на рис. 2.8 и 2.9.



**Рис. 2.8. Окно конфигурирования модуля специального входа**



**Рис. 2.9. Окно конфигурирования модуля дискретного выхода**

## 2.4. Модули аналоговых входов

Модули аналоговых входов относятся к стационарным модулям. В зависимости от конкретной модели ПЛК их количество может быть разным. Для них в Конфигураторе ПЛК заложено соответствующее количество модулей аналоговых входов. В зависимости от поставленной задачи могут быть использованы разные модули. Пользователь выбирает необходимые модули функцией контекстного меню **Replace Elements** из четырёх возможных вариантов в выпадающем списке: датчик унифицированного сигнала, датчик типа «термопара», датчик типа «термосопротивление» и контактный датчик.

Модули предназначены для приведения результатов измерения к значениям физической величины, измеряемой датчиком. Приведенное значение физической величины может быть считано в программу ПЛК.

Рис. 2.10 иллюстрирует возможность выбора варианта аналогового входа.

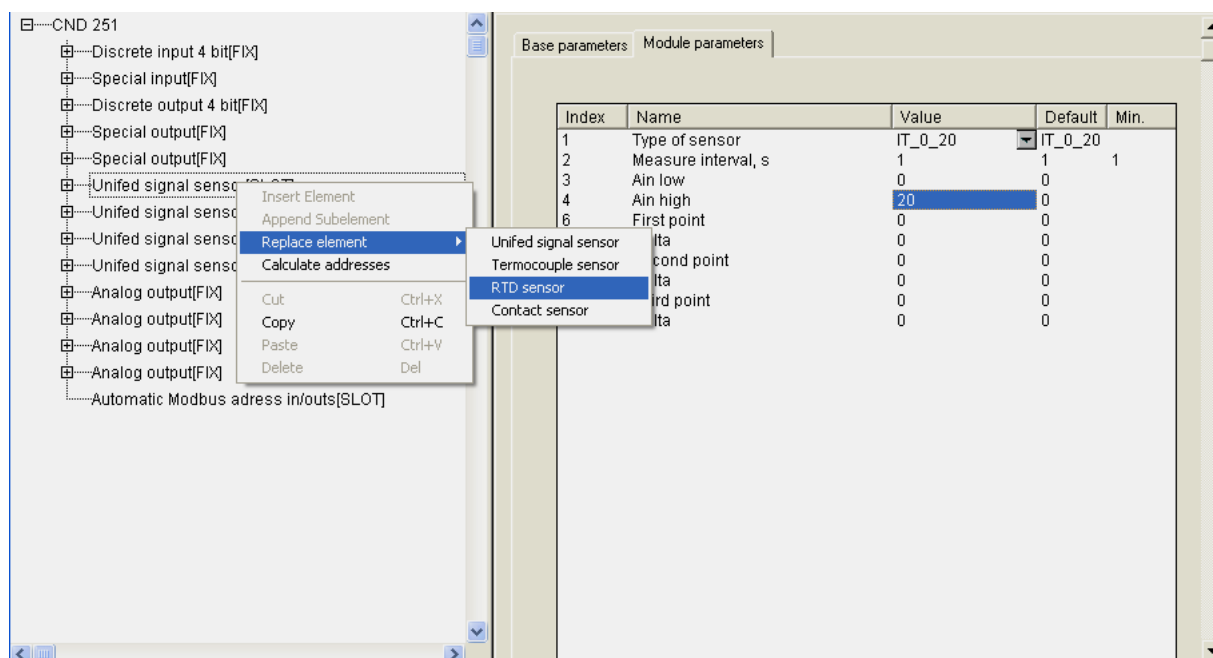


Рис. 2.10. Окно выбора типа модуля аналогового входа

У каждого типа модуля 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 семь возможных типов датчиков, значение по умолчанию – **IT\_4\_20** (датчик с унифицированным сигналом постоянного тока от 4 до 20 мА) (см. п. 2.4.1.1).

«Время между измерениями в секундах» (**Measure interval, s**) – максимальное время между замерами не ограничено, значение может быть любым, в т.ч. дробным, минимальное – 1 сек, значение по умолчанию – 1 (см. п. 2.4.1.2).

«Нижняя и верхняя границы» (**Ain low & Ain high**) – границы диапазона измерения 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 производимого измерения и вводимые изменения, значения по умолчанию – 0 (см. п. 2.4.1.4).

Модуль имеет два канала:

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

---

**Примечание.** При ошибке датчика (и выходе за диапазон) значение содержит код ошибки в специальном формате.

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

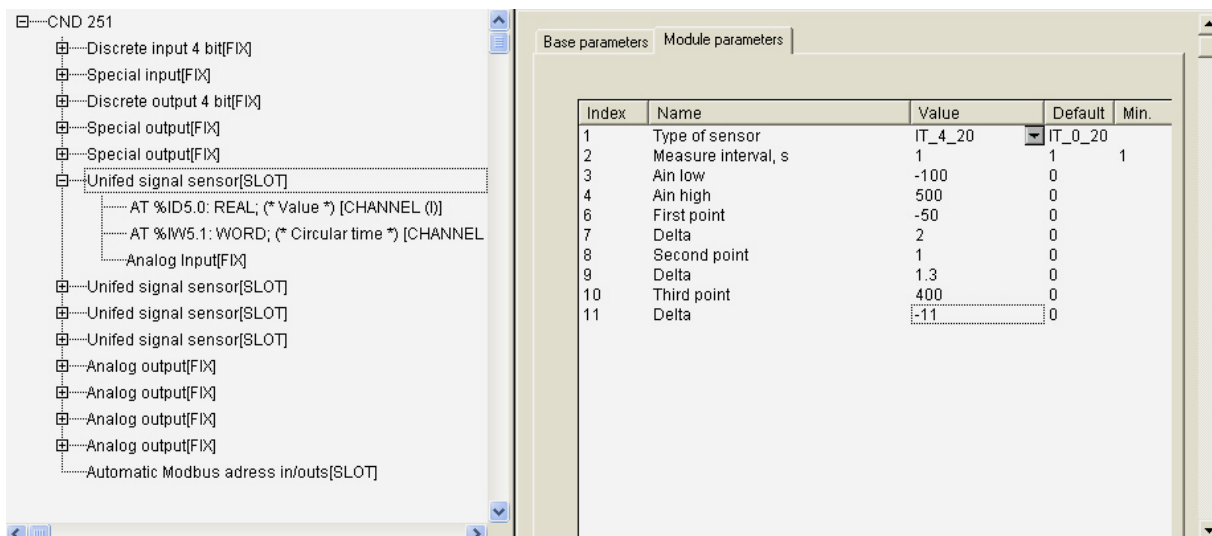
---

Сообщения об ошибках в ПЛК представлены в Приложении В.

- Канал **Circular time («Циклическое время»)**, формат **Word** – значение циклического времени измерения в диапазоне от 0 до 65536 ед. (1 ед.=10 мс). Отсчет времени начинается от момента старта программы ПЛК и обнуляется при переходе через максимальное значение.

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

Окно конфигурирования модуля аналогового входа «Датчик унифицированного сигнала» с вкладкой параметров представлено на рис. 2.11.



**Рис. 2.11. Окно конфигурирования модуля аналогового входа «Датчик унифицированного сигнала» с вкладкой параметров**

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

В параметре «**Тип датчика**» (**Type of sensor**) задается тип датчика, показания которого будут обрабатываться модулем «**Датчик унифицированного сигнала**». Пользователь осуществляет выбор из семи возможных вариантов в выпадающем списке: датчики с унифицированным сигналом тока 0 ... 20 мА, 4 ... 20 мА, 0 ... 5 мА, датчики с унифицированным сигналом напряжения минус 50... +50 мВ, 0 ... 1 В, 0 ... 10 В и сопротивление от 0 до 5000 Ом.

Перечень используемых типов датчиков представлен в Приложении А.

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

В параметре «**Время между измерениями в секундах**» (**Measure interval, s**) задают интервал времени между опросами датчика.

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

В параметрах «**Нижняя и верхняя границы**» (**Ain low & Ain high**) задают диапазон измерений датчика унифицированного сигнала.

**Например**, датчик с унифицированным сигналом «параметр – ток 0...20 мА» измеряет температуру в диапазоне минус 100 °С...+500 °С, и 0 мА соответствует нижней границе диапазона измерения – минус

100 °С, а 20 мА соответствует верхней границе диапазона измерения – +500 °С. Тогда выходная величина модуля будет изменяться от минус 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)**

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

Модуль 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 заданными точкам измерения и поправкам Delta в этих точках.

Технология коррекции следующая. Например, при калибровке определено, что при замере в 1-ой точке – минус 50 °С, датчик показывает значение минус 48 °С. Вносится поправка –2 °С, при сложении с которой получается реальное значение, которое используется в дальнейшей корректировке работы датчика. Также и с остальными точками. Желательно, чтобы при калибровке точки коррекции были расположены далеко друг от друга. В противном случае могут быть большие погрешности, наклон графика будет неверным.

---

**Примечание.** Если коррекция происходит по одной точке, то вся кривая смещается на одну корректирующую величину, по двум точкам – смещается и изменяется наклон, по трём – для внесения коррекции используется полином 2-й степени (меняется форма кривой).

Точки задаются, начиная с первой. 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 типы датчиков, значение по умолчанию – TP\_L (термопара ТХК (L) с диапазоном измерения [–200 °С...+800 °С]) (см. п. 2.4.1.1).

**«Время между измерениями в секундах» (Measure interval, s)** – максимальное время между замерами не ограничено, значение может быть любым, в т.ч. дробным, минимальное – 1 сек, значение по умолчанию – 1 (см. п. 2.4.1.2).

**«Компенсация холодного спая» (Cold junction compensation)** – значения выбираются из списка **«On» (Включить)** и **«Off» (Выключить)**, значение по умолчанию – **«On»** (см. п. 2.4.2.1).

**«Нижняя и верхняя границы» (Ain low & Ain high)** – границы диапазона измерения 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 производимого измерения и вводимые изменения, значения по умолчанию – 0 (см. п. 2.4.1.4).

Модуль имеет два канала:

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

---

**Примечание.** При ошибке датчика (и выходе за диапазон) значение содержит код ошибки в специальном формате.

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

Сообщения об ошибках в ПЛК представлены в Приложении В.

---

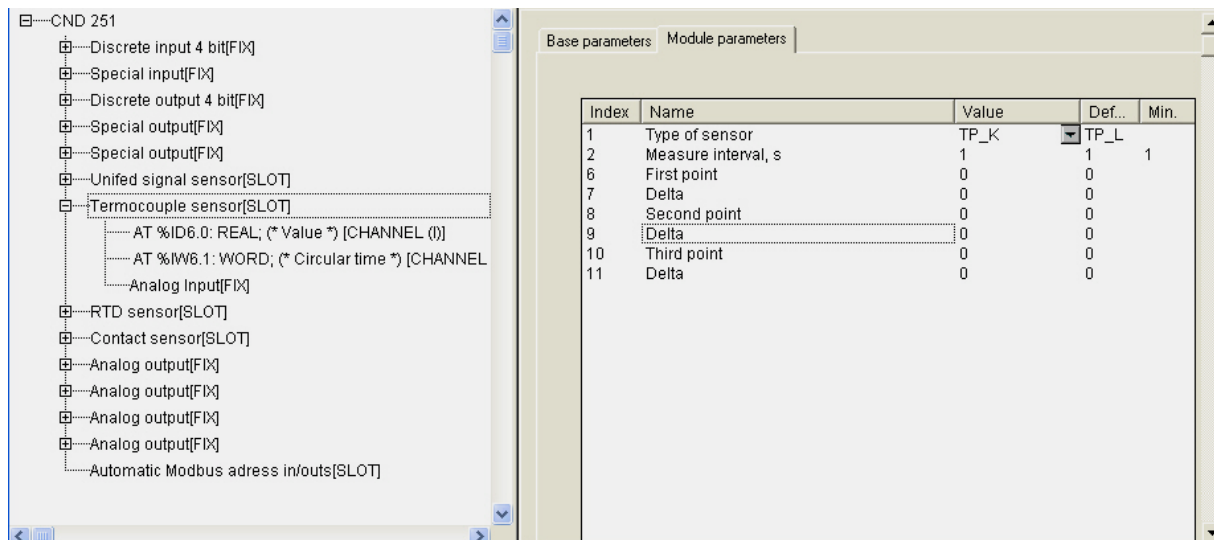
- Канал **Circular time («Циклическое время»), формат Word** – значение циклического времени измерения в диапазоне от 0 до 65536 ед. (1 ед.=10 мс). Отсчет времени начинается от момента старта программы ПЛК и обнуляется при переходе через максимальное значение.

**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.

Окно конфигурирования аналогового входа «Термопара» с вкладкой параметров представлено на рис. 2.12.



**Рис. 2.12. Окно конфигурирования аналогового входа «Термопара» с вкладкой параметров**

### **2.4.2.1. Параметр «Компенсация холодного спая» (Cold junction compensation)**

В параметре «Компенсация холодного спая» (**Cold junction compensation**) определяют необходимость компенсации холодного спая. Отключение компенсации холодного спая требуется для проведения метрологической поверки аналоговых входов ПЛК. В обычном режиме работы компенсацию холодного спая рекомендуется включать.

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

При подключении датчика типа "Термосопротивление" по двухпроводной схеме необходимо выполнить компенсацию сопротивления соединительных проводов по методике, изложенной в Приложении А.

#### **Параметры модуля:**

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

«**Время между измерениями в секундах» (Measure interval, s)** – максимальное время между замерами не ограничено, значение может быть любым, в т.ч. дробным, минимальное – 1 сек, значение по умолчанию – 1 (см. п. 2.4.1.2).

**«Нижняя и верхняя границы» (Ain low & Ain high)** – границы диапазона измерения 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).

Модуль имеет два канала:

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

---

**Примечание.** При ошибке датчика (и выходе из диапазона) значение содержит код ошибки в специальном формате.

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

Сообщения об ошибках в ПЛК представлены в Приложении В.

---

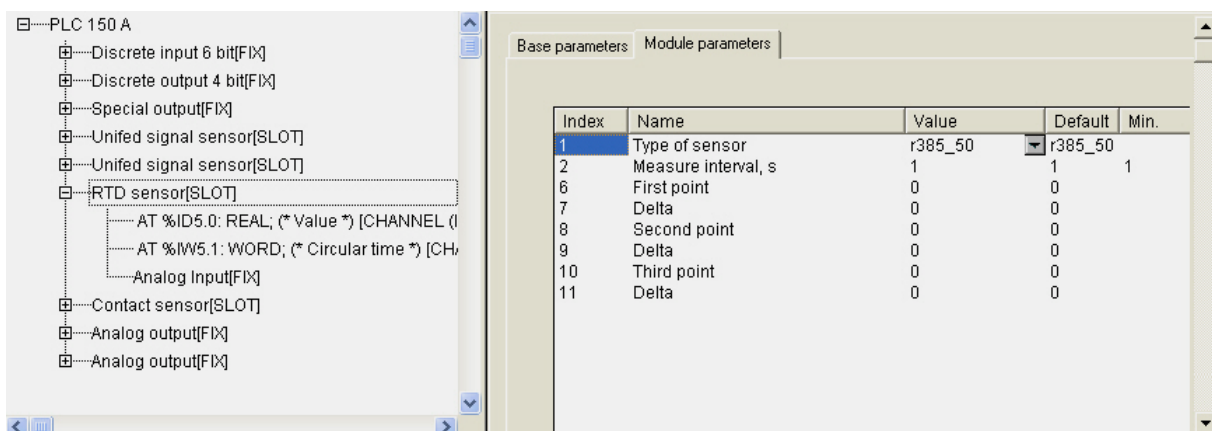
- Канал **Circular time («Циклическое время»), формат Word** – значение циклического времени измерения в диапазоне от 0 до 65536 ед. (1 ед.=10 мс). Отсчет времени начинается от момента старта программы ПЛК и обнуляется при переходе через максимальное значение.

**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 типов термосопротивлений.

Перечень используемых типов датчиков представлен в Приложении А.

Окно конфигурирования аналогового входа **«Термосопротивление»** представлено с вкладкой параметров на рис. 2.13.



**Рис. 2.13. Окно конфигурирования аналогового входа «Термосопротивление» с вкладкой параметров**

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

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

##### Параметры модуля:

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

«**Время между измерениями в секундах**» (**Measure interval, s**) – максимальное время между замерами не ограничено, значение может быть любым, в т.ч. дробным, минимальное – 1 сек, значение по умолчанию – 1 (см. п. 2.4.1.2).

Модуль имеет два канала:

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

---

**Примечание.** При ошибке датчика (и выходе из диапазона) значение содержит код ошибки в специальном формате.

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

Сообщения об ошибках в ПЛК представлены в Приложении В.

---

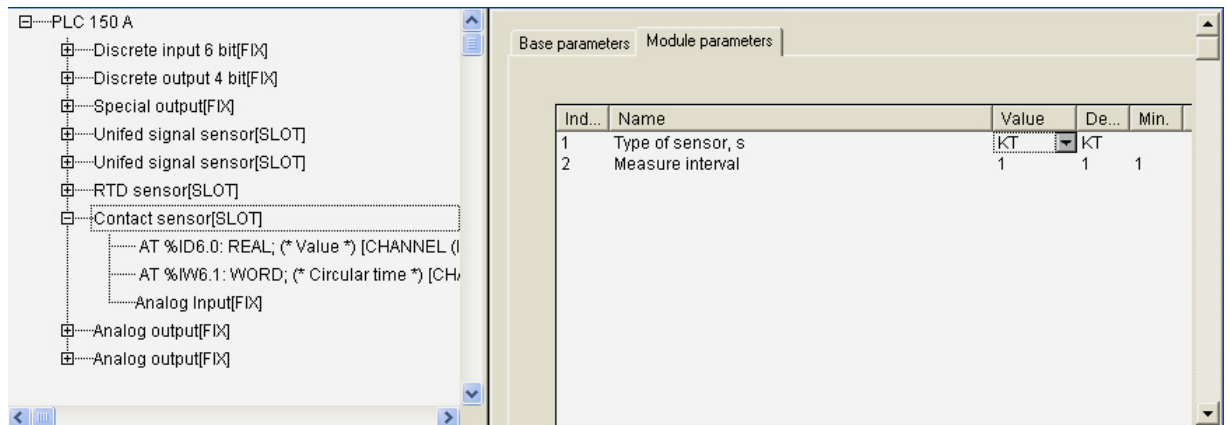
- Канал **Circular time («Циклическое время»)**, формат **Word** – значение циклического времени измерения в диапазоне от 0 до 65536 ед. (1 ед.=10 мс). Отсчет времени начинается от момента

старта программы ПЛК и обнуляется при переходе через максимальное значение.

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

Выходное значение модуля «**Контактный датчик**» 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.

Окно конфигурирования аналогового входа «**Контактный датчик**» с вкладкой параметров представлена на рис. 2.14.



**Рис. 2.14. Окно конфигурирования аналогового входа «Контактный датчик» с вкладкой параметров**



## 2.5. Модуль аналоговых выходов (Analog output)

Аналоговые выходы относятся к стационарным модулям. В зависимости от типа установленных в ПЛК ЦАП существуют три варианта настройки модуля аналоговых выходов:

- универсальный – программно настраиваемый либо на ЦАП «параметр – ток», либо на ЦАП «параметр – напряжение»;
- фиксированный, предназначенный для ЦАП «параметр – ток»;
- фиксированный, предназначенный для ЦАП «параметр – напряжение».

### Параметры модуля:

«Тип выхода» (**Type**) – значения выбираются из списка «**Current 0 – 20 mA**» и «**Voltage 0 – 10 V**», значение по умолчанию – зависит от типа ЦАП (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, значение 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 – «**no**» (см. п. 2.1.2).

На рис. 2.15 представлено окно настройки модуля аналогового выхода с вкладкой параметров.

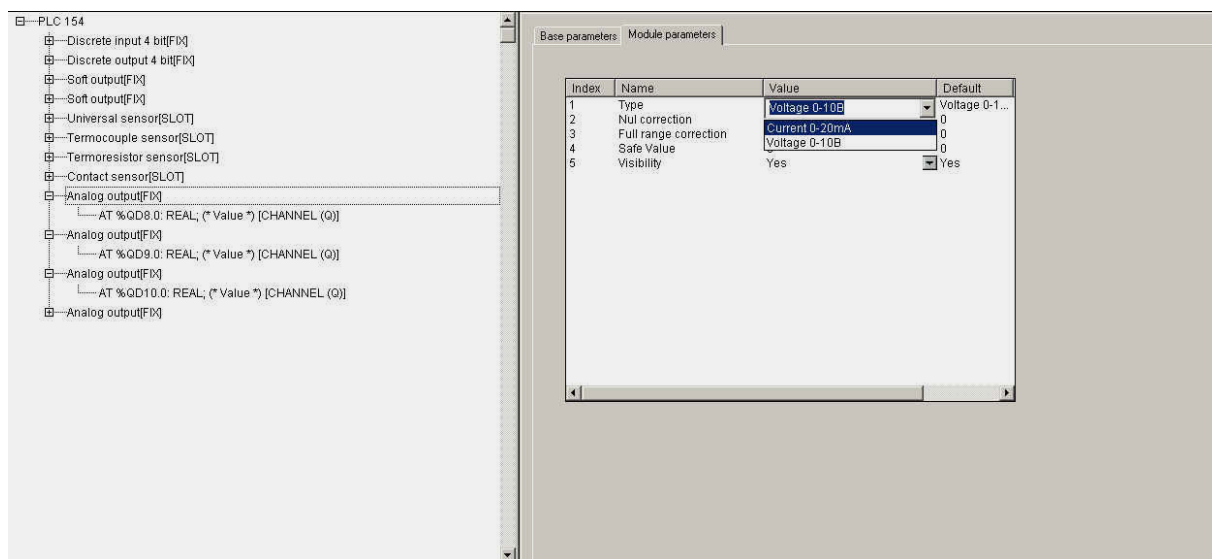


Рис. 2.15. Вкладка параметров модуля аналогового выхода

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

Параметр «**Тип выхода**» (Type) задает тип аналогового выхода. Возможные значения: «параметр – ток 4...20 мА» или «параметр – напряжение 0...10 В».

---

**Примечание.** Изменение значения параметра может быть заблокировано при установке в ПЛК ЦАП фиксированного типа.

---

**В канал** аналогового модуля из программы ПЛК передается число, равное желаемому выходному значению фиксированного модуля аналогового входа. 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. В параметрах коррекции могут быть заданы корректирующие значения в случае неточной работы ЦАП. Точки выбраны следующим образом: для ЦАП «параметр – ток» – 4 и 15 мА, а для ЦАП «параметр – напряжение» – 1 В и 10 В.

## 3. Описание добавляемых модулей Конфигуратора ПЛК

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 в соответствии с поставленной задачей или 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 двух 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.

**Каналы 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 – «**no**» (см. п. 2.1.2).

**Список подмодулей:**

**Modbus (FIX)** (см. п. 3.1.2).

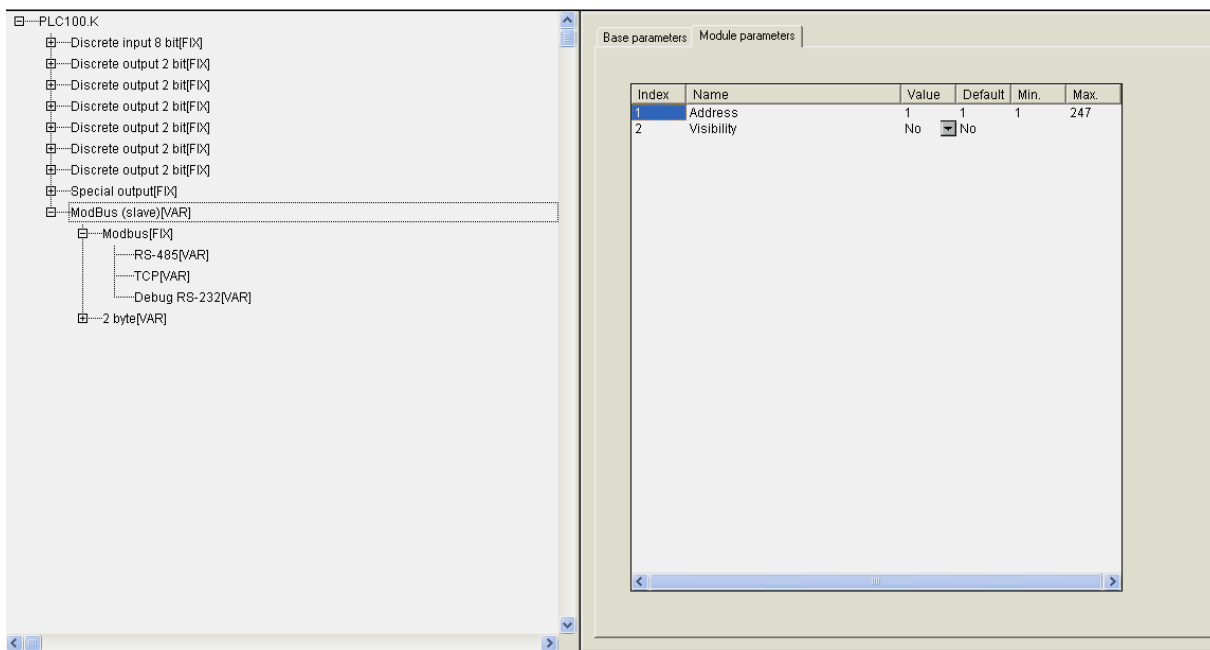
Переменные, которыми будет обмениваться ПЛК по протоколу **Modbus**, выбираются пользователем функцией контекстного меню **Append Subelements**.

---

**Внимание!** При случайном отключении питания во время работы ПЛК последние (текущие) значения переменных сохраняются в энергонезависимой памяти и восстанавливаются при возобновлении работы прибора.

---

Окно конфигурирования модуля **Modbus (Slave)** представлено на рис. 3.1.



**Рис. 3.1. Окно конфигурирования модуля Modbus (Slave)**

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

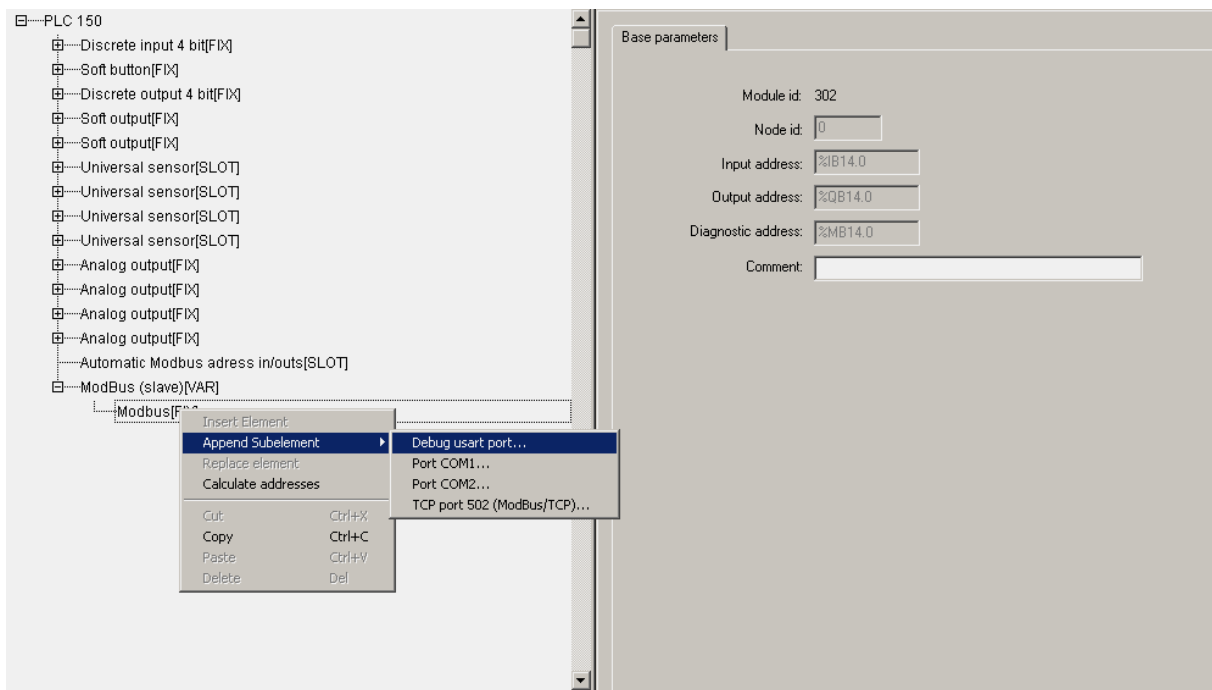
Значение параметра «**Адрес устройства**» (**Address**) – это адрес прибора. Параметр имеет значения в диапазоне от 1 до 247, при этом может быть равен 0, и именно это значение выставлено по умолчанию, но применяется в специфических случаях.

### 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.

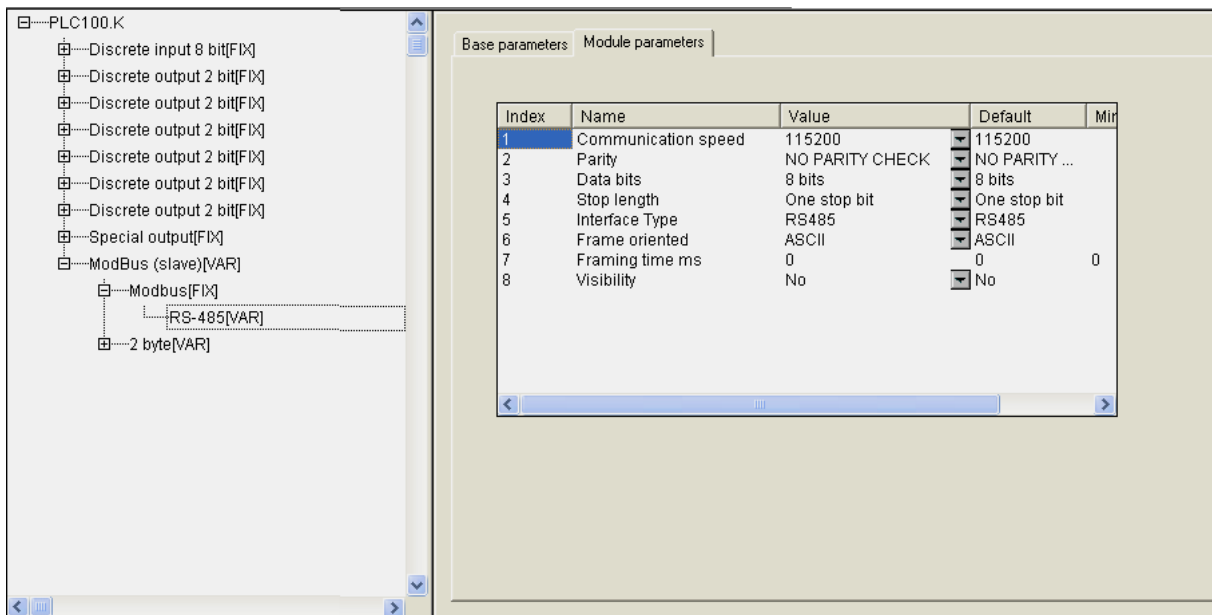
В состав модуля **Modbus (Slave)** фиксировано вставлен подмодуль **Modbus (FIX)** – внутренний модуль, создающий определенную структуру, к которой в свою очередь подключается коммуникационный интерфейс. В ПЛК предусмотрена возможность обмена данными по интерфейсам: **RS-232**, **RS-485** и **TCP (Ethernet)**.

Для работы с разными коммуникационными интерфейсами в ПЛК предусмотрены соответствующие подмодули. Рис. 3.2 демонстрирует процесс подключения подмодулей.



**Рис. 3.2. Окно добавления модулей настройки коммуникаций**

На рис. 3.3. представлена вкладка параметров последовательного порта.



**Рис. 3.3. Окно задания 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 б, 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 – «**no**» (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 в диапазоне от 0 до 65535, значение по умолчанию – 502 (см. п. 3.1.2.8).

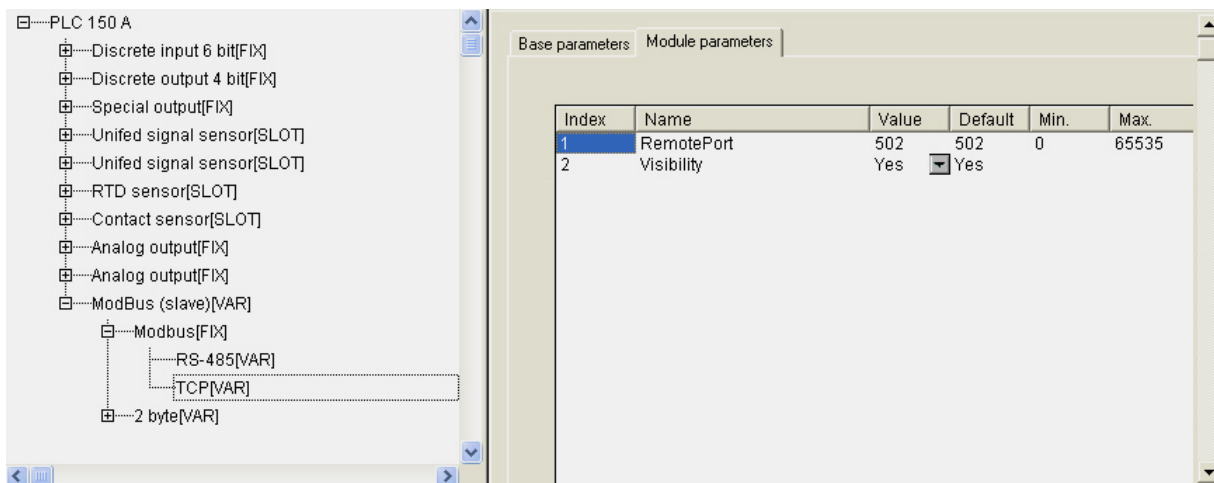
«**Видимость**» (**Visibility**) – задает видимость параметров модуля в программе EasyWorkPLC. Значения выбираются из списка «**yes**» и «**no**», значение по умолчанию – «**no**» (см. п. 2.1.2).

---

**Примечание.** IP-адрес и MAC-адрес указываются для контроллера и задаются на предприятии-изготовителе. Адреса можно поменять специальной процедурой (см. **PLC Browser**). В данном случае модуль использует MAC- и IP-адреса контроллера.

---

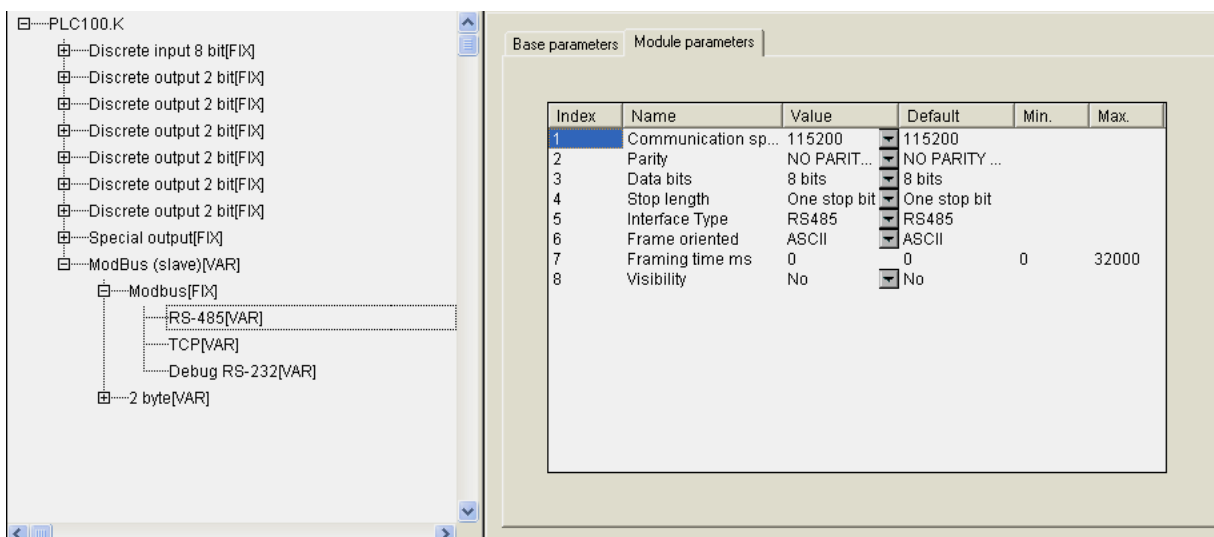
На рис. 3.4. представлено окно конфигурирования порта TCP с вкладкой параметров.



**Рис. 3.4. Окно конфигурирования порта TCP со вкладкой параметров**

При работе ПЛК в режиме «ведомый» возможно использование нескольких разных портов, т.е. опрос может вестись по разным интерфейсам. Таким образом, подключая несколько разных портов, 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. Окно подключения 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)**

В параметре «Скорость передачи информации» (**CommunicationSpeed, bod**) устанавливаются скорость передачи информации через последовательный порт.

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

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

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

В параметре «Количество бит данных» (**DataBits**) задают количество значащих бит в одном байте посылки.

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

Параметром «Количество стоп-битов» (**StopLength**) задают количество стоп-битов.

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

В параметре «Тип интерфейса» (**InterfaceType**) задают тип последовательного интерфейса, по которому осуществляется информационный обмен (**RS-232** или **RS-485**).

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

В ПЛК используются следующие типы протоколов обмена: ориентированный на передачу текстовых символов режим ASCII и ориентированный на передачу потока байтов (RTU-режим). В параметре «Тип



**протокола обмена» (FrameOriented)** задаётся тип протокола обмена (RTU или ASCII).

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

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

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

В параметре **«Время, на которое необходимо задерживать ответ на запрос в мс» (Framing time)** задают временную задержку между последним байтом принятого пакета и первым байтом, передаваемым в ответ. Задержка бывает необходима для работы с устройствами с низкими скоростями информационного обмена.

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

В параметре **«Удаленный порт» (RemotePort)** задают 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 выделения памяти и распределения адресов ячеек памяти внутри **Modbus (Slave)**.

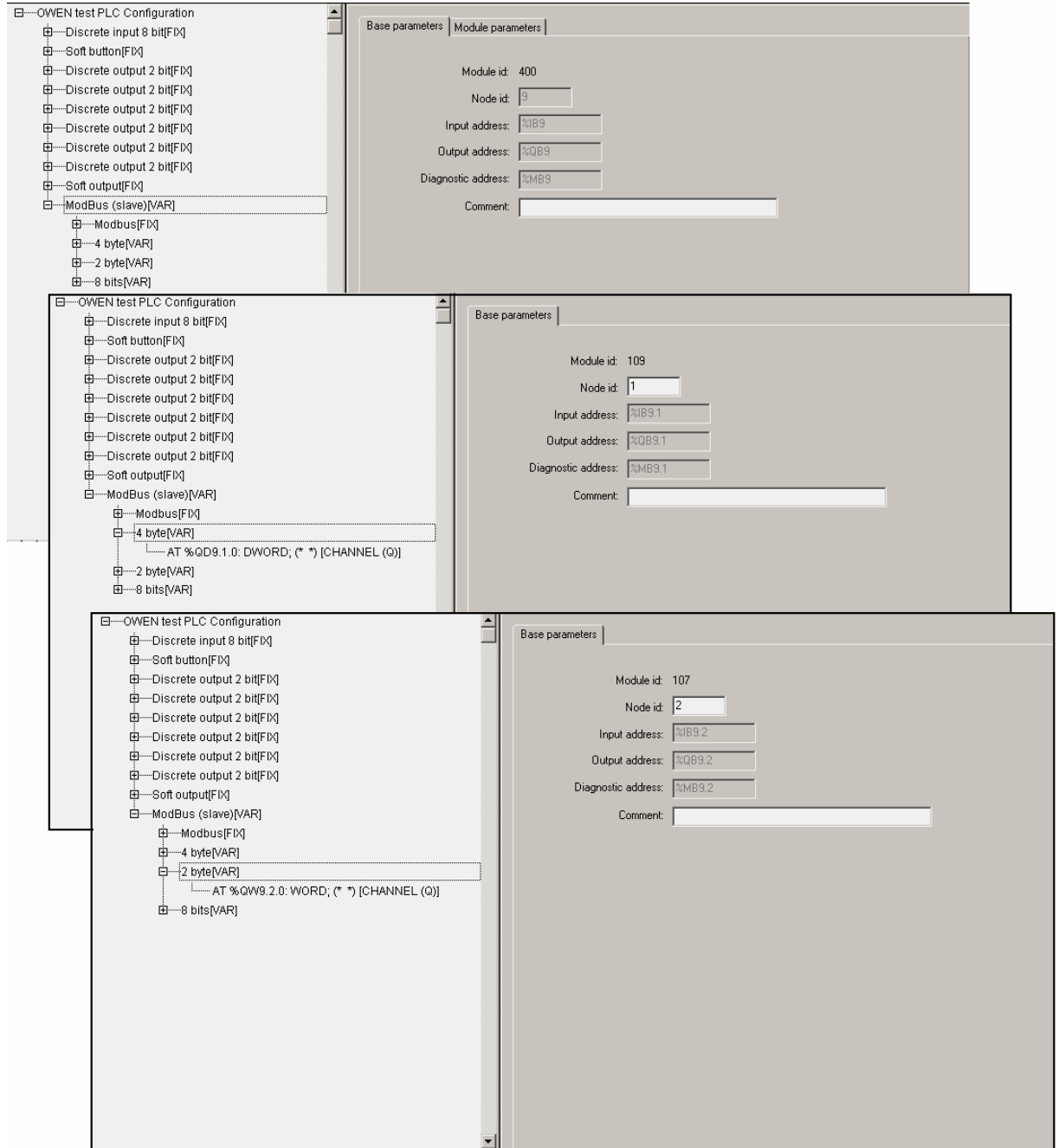
Протокол **Modbus** рассматривает каждое устройство, включая и модуль **Modbus (Slave)**, как память, набор неких конкретных регистров.

---

**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 друг от друга. В некоторых приборах они учитываются раздельно, в других приборах – это общая память, но к ней можно обращаться по-разному. Биты нумеруются 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 добавить следующие четыре типа каналов (переменных): REAL, 4 байта, 2 байта или 8 бит (см. рис. 3.6).



**Рис. 3.6. Окна Modbus (Slave). Добавление переменных**

Добавление переменных различных размеров требуется для структурирования памяти. С точки зрения Мастера **slave**-устройство представляет собой непрерывно организованную область памяти, к которой можно обращаться 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

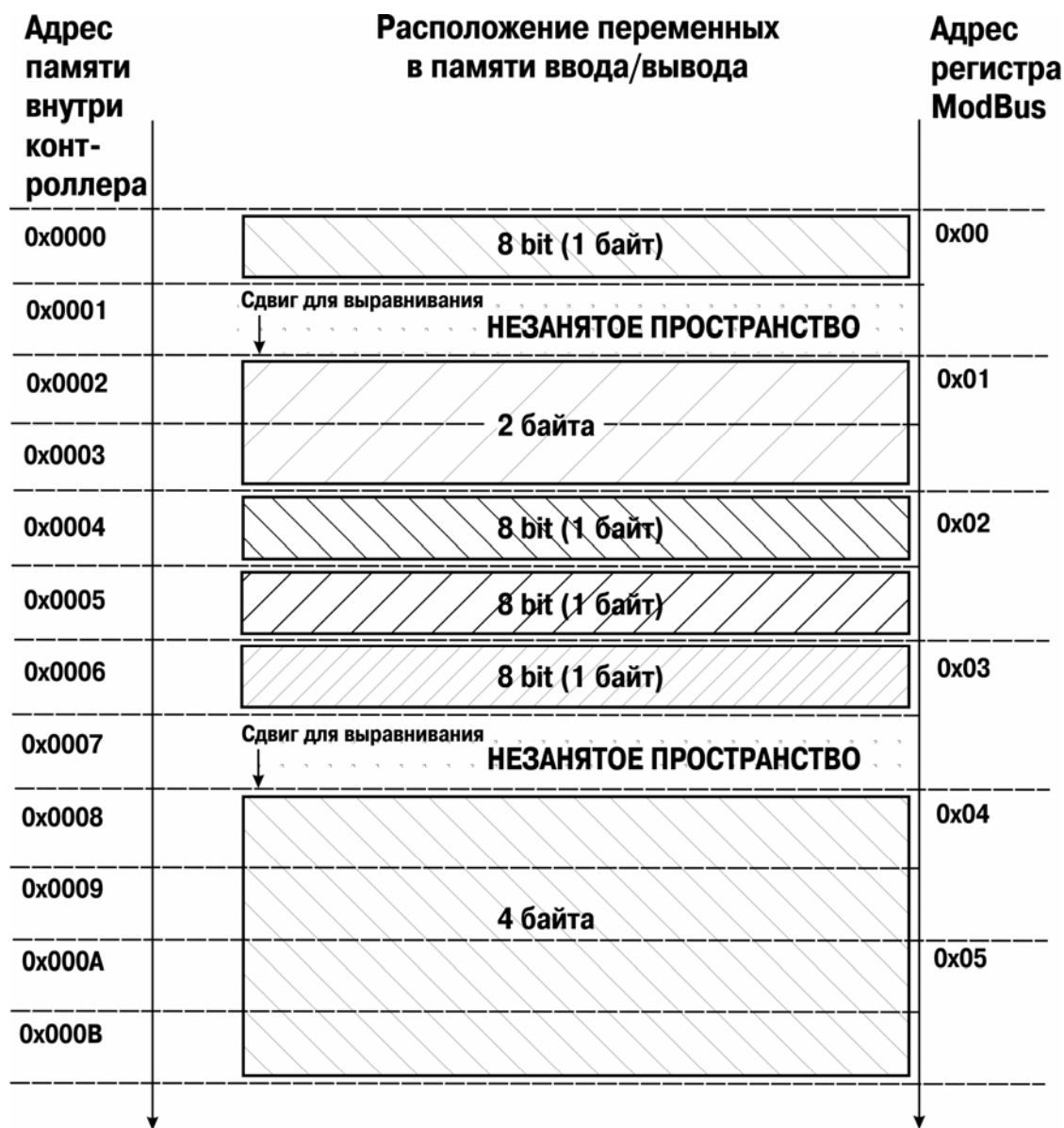
номер 0», устройство считывает первые два байта первой переменной, для «регистра номер 1» – вторые два байта первой переменной (если та имеет длину более двух байт) или первые два байта второй переменной, и т.д. Особенность организации обращений заключается в том, что, если в устройстве установлен (используется) 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 размером 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 место, кратное четырём, 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 возникают не занятые пространства памяти, которые нигде не отображаются, не видны в области ввода/вывода, но обязательно 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. Пример размещения переменных в области ввода/вывода

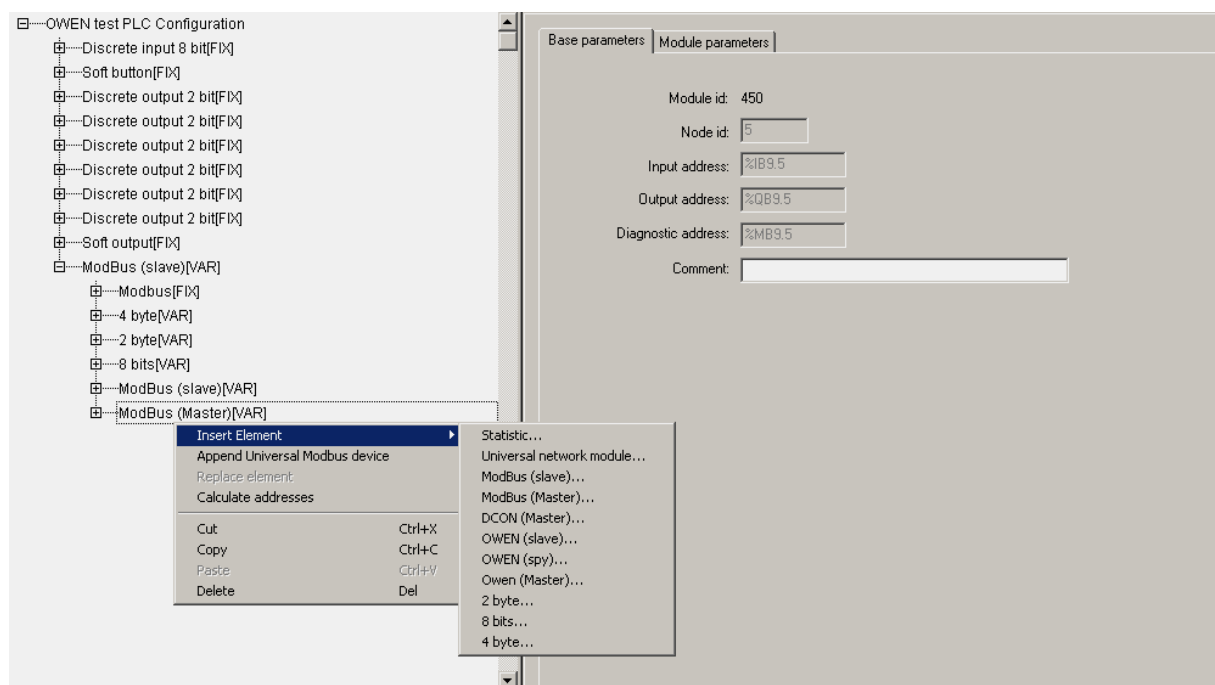
### 3.1.3.2. Отображение устройства в области памяти

Наряду с переменными, 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). При 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.

Окно подключения 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)

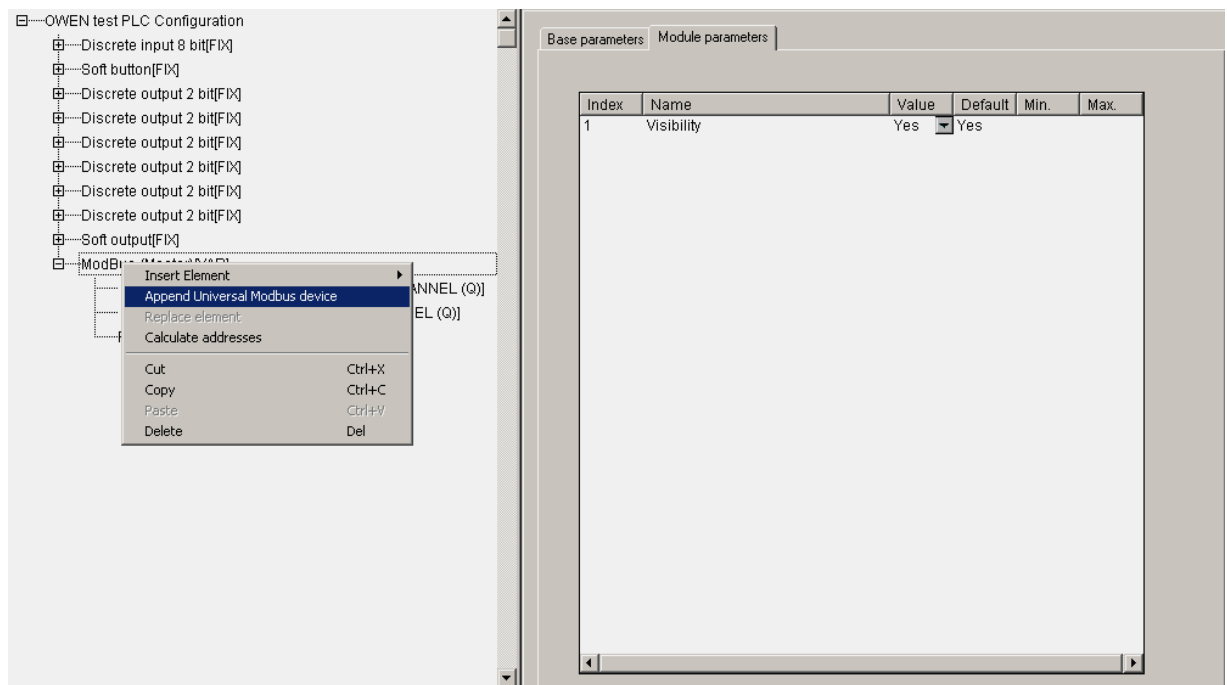
В данном разделе рассматривается работа ПЛК по протоколу **Modbus** в режиме «**ведущего**», т.е. режиме «**Master**».

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 – «**no**» (см. п. 2.1.2).

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



**Рис. 3.9. Окно конфигурирования модуля 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 последовательного порта, его можно только заменить другим. Если предполагается, что **Мастер** будет работать с конкретным устройством по протоколу TCP, то необходимая настройка производится в подмодуле устройства (подмодуль **Universal Modbus Device**).

Установка и настройка устройств, опрашиваемых **Мастером**, производится через установку подмодуля «универсальное устройство **Modbus (Master)**» и задание необходимых параметров этому подмодулю. Его

подключение выполняется командой «**Append Universal Modbus Device**», вызываемой из контекстного меню (см. рис. 3.9).

При опросе модулем **Modbus (Master)** подчиненных устройств информация о ходе обмена записывается в его переменных.

#### Каналы модуля:

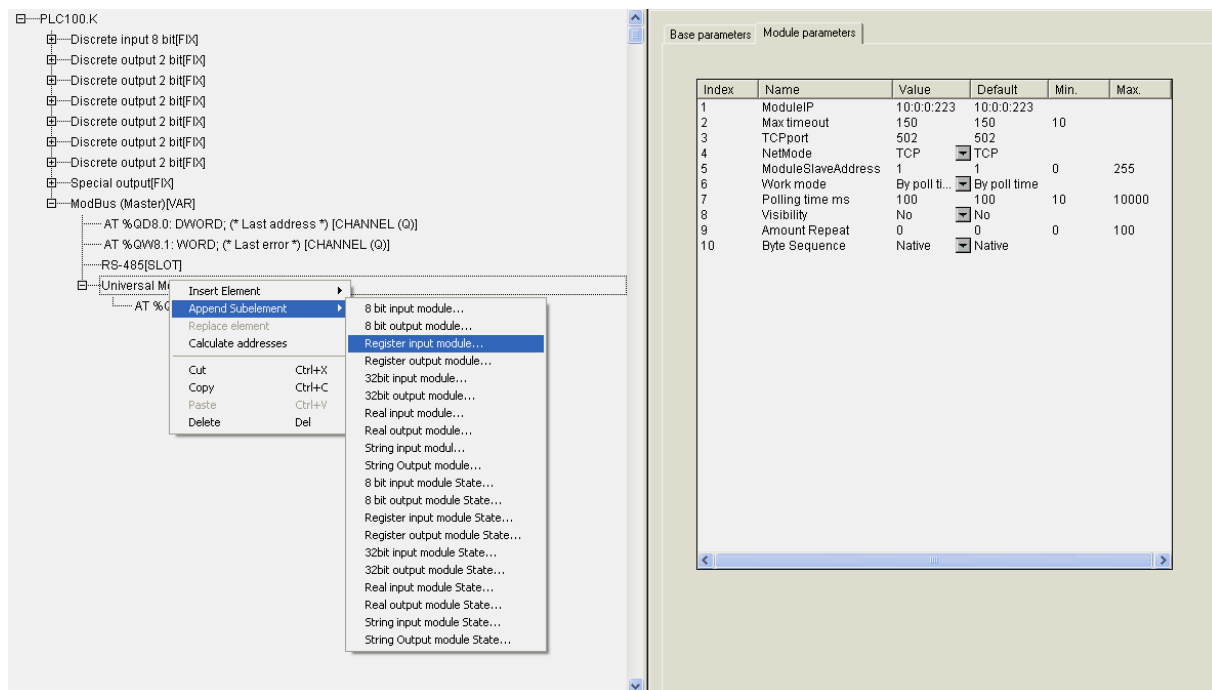
**LastAddress** – адрес последнего опрошенного **Modbus (Master)** устройства. Модуль запрашивает устройство, и, соответственно, тут же меняется значение: показывается значение адреса последнего запроса.

**LastError** – код ошибки. В переменной отображается код ошибки, если информационный обмен прошёл неудачно. Это необходимо для корректности работы опрашиваемого устройства.

Коды ошибок ПЛК представлены в Приложении В.

### 3.2.1. Модуль «Универсальное устройство Modbus (Master)»

Окно настройки модуля «Универсальное устройство **Modbus (Master)**» представлено на рис. 3.10.



**Рис. 3.10. Окно конфигурирования модуля «Универсальное устройство Modbus (Master)»**

При вызове модуля универсального устройства появляется незаполненный шаблон, параметры и идентификаторы которого не привязаны к конкретному внешнему устройству (модулю ввода-вывода, операторской панели). При задании значений параметров модуля, конфигурируется конкретный вариант внешнего устройства.

**Универсальное устройство Modbus** имеет канал **Start/Stop**. Если в него записывается значение **0x00FF**, то происходит старт работы данного устройства Modbus, если же в канал записано значение **0x00FE**, то происходит его остановка и прекращение всех посылок в сеть. Если модуль уже запущен, то повторная запись в канал значения **0x00FF** приводит к внеочередному запросу одной очередной переменной устройства Modbus.

#### **Параметры универсального устройства Modbus:**

«**IP-адрес**» (**Module IP**) – 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. дробным, но не меньше 10 мс, значение по умолчанию – 150 (см. п. 3.2.1.2).

«**Порт TCP**» (**TCP port**) – 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).

«**Режим работы**» (**Work mode**) – значения выбираются из списка «**По времени**» (**Poll time**), «**По изменению значения переменных**» (**Value change**), «**По времени и по изменению значения переменных**» (**Both**) и «**По команде**» (**By Command**), значение по умолчанию – «**Polling time**» (см. п. 3.2.1.6).

«**Число повторов**» (**Amount Repeat**) – рекомендуемый диапазон значений от 0 до 5, значение по умолчанию – 0 (см. п. 3.2.1.7).

«**Порядок передачи байтов посылки**» (**Byte Sequence**) – значения выбираются из списка «**Порядок байтов, используемый в ПЛК**» (**Native**) и «**Порядок байтов, используемый в программе Trace mode**» (**Trace mode**), значение по умолчанию – «**Native**» (см. п. 3.2.1.8).

«**Период опроса устройства, в мс**» (**Polling time**) – диапазон значений от 10 до 10000, значение по умолчанию – 100 (см. п. 3.2.1.9).

---

**Примечание.** В Мастере, когда он работает в режиме «**По изменению значения переменных**» или «**По команде**», нельзя ставить значение параметра **Polling time** слишком маленьким. По умолчанию его значение 100 мс и в этих режимах оно не влияет на периодичность посылки запросов мастера. Однако, если на реальном проекте будет замечено, что Мастер при загрузке программы или при **Login** формирует лишние пакеты и/или запросы, которых не должно быть, значение параметра увеличивают (до 200, 300 и т.д.) до предотвращения появления ложных пакетов.

---



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

---

**Внимание!** На вкладке, изображенной на рис. 3.10, представлены параметры, относящиеся как исключительно к режиму TCP, так и общие, относящиеся к обоим режимам – TCP и Serial.

---

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

В параметре «**IP-адрес**» (**Module IP**) задают IP-адрес подчиненного устройства, которым управляет **Мастер**, если оно будет работать в режиме TCP.

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

В параметре «**Максимальный тайм-аут, в мс**» (**Max timeout**) устанавливают максимальное время, в течение которого устройство должно ответить на запрос. Если по истечении этого времени **Мастер** не получил ответ на запрос, то это значит, что произошел сбой или авария. Информация о сбое фиксируется в переменной модуля **Last error**. **Мастер** продолжает опрос других устройств.

### **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, работающее по протоколу TCP.
- **Serial** – устройства, осуществляющие обмен данными через последовательный интерфейс.

---

При задании режима **Serial** обмен ведется через тот последовательный интерфейс, который подключен к модулю Мастера. В случае задания режима **TCP** используется интерфейс **Ethernet**. В этом случае опрашиваемое внешнее устройство идентифицируется по IP-адресу.

---

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

В параметре «Адрес подчиненного устройства» (**ModuleSlaveAddress**) задают 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 «Режим работы» (Work mode)**

В параметре «Режим работы» (**Work mode**) – выбирают режим работы модуля **Owen (Master)** при опросе внешних устройств:

- **Poll time** – «по времени» – контролируемые устройства опрашиваются с периодичностью, заданной в параметре «Период опроса устройства» (**Polling time**);
- **Value change** – «по изменению значения переменных» – модуль **Owen (Master)** генерирует запрос устройству при изменении значений выходных переменных модуля;
- **Both** – «оба варианта» – опрос производится с временным интервалом, заданным в параметре **Polling time** и тогда, когда изменяются значения выходных переменных;
- **By Command** – «по команде» – производится однократная посылка запроса, когда в командный канал **Command** переменной записывается значение **0x00FF**.

---

**Внимание.** Для переменных с командным каналом при работе в режиме «По команде» (**By Command**) управление осуществляется следующим образом: первая посылка значения **0x00FF** в командный канал включает функционирование этой переменной, повторная посылка значения **0x00FF** инициирует проведение опроса. Аналогично опрос инициируется для переменных с командным каналом при работе в других режимах. При посылке в командный канал значения **0x00FE** переменная выключается из цикла опроса мастера.

---

### **3.2.1.7. Параметр «Число повторов» (Amount Repeat)**

В параметре «Число повторов» (**Amount Repeat**) определяют число повторов чтения/записи переменных при неудачном сеансе связи. В режиме «По времени» (**Polling time**) значение этого параметра не используется.

### **3.2.1.8. Параметр «Порядок передачи байтов посылки» (Byte Sequence)**

В параметре «**Порядок передачи байтов посылки**» (**Byte Sequence**) – определяют, в каком порядке будут передаваться байты посылки протокола **Modbus** для переменных длиной **32 бита**. У устройств разных производителей этот порядок разный, он не стандартизирован в самом протоколе, поэтому необходимо обговаривать его для конкретных устройств.

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

В параметре «**Период опроса устройства, в мс**» (**Polling time**) задают период опроса внешнего устройства.

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

После задания значений параметров модуля **Universal Modbus Device** к нему требуется подключить каналы, задающие переменные ввода/вывода модуля. Могут быть подключены следующие типы каналов (переменных): REAL, STRING, 4 байта, 2 байта или 8 бит. Кроме того, реализована возможность работы по команде: вставляются переменные с обозначением «**Comm**», содержащие дополнительный управляющий канал (см. также п. 3.2.1.6).

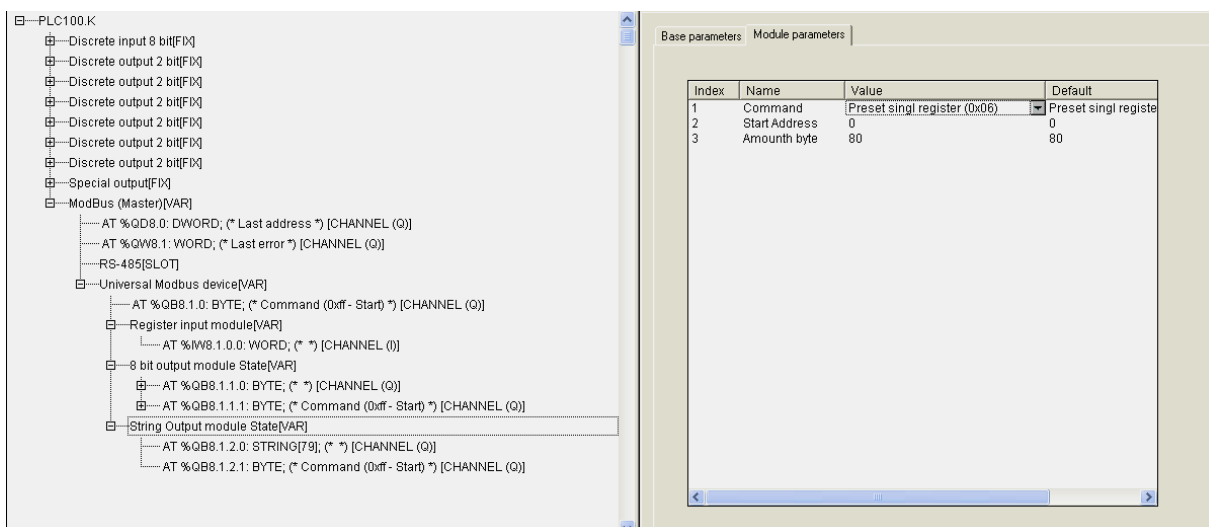
#### **Параметры каналов ввода/вывода модуля Modbus (Master):**

«**Номер команды протокола**» (**Command**) – значения выбираются из списка возможных вариантов, значение по умолчанию – «**preset single register (0x06)**» для выходной строковой переменной («**write multiple registers (0x10)**» для 2-байтной переменной и др.) (см. п. 3.2.2.1).

«**Адрес регистра опрашиваемого устройства**» (**Start Address**) – устанавливают адрес регистра опрашиваемого устройства (см. п. 3.2.2.2).

«**Размер переменной**» (**Amount byte**) – для переменных типа **String**, диапазон значений от 1 до 80, значение по умолчанию – 80 (см. п. 3.2.2.3).

Рис. 3.11 иллюстрирует окно настройки каналов ввода/вывода универсального устройства.



**Рис. 3.11. Окно настройки каналов ввода/вывода модуля «Универсальное устройство»**

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

В параметре «**Номер команды протокола**» (**Command**) задают номер команды (номер функции) протокола, по которой будет производиться обмен, т.к. некоторые устройства поддерживают не все команды (функции) протокола Modbus.

### **3.2.2.2. Параметр «Адрес регистра опрашиваемого устройства» (Start Address)**

В параметре «**Адрес регистра опрашиваемого устройства**» (**Start Address**) задают адрес регистра опрашиваемого устройства.

### **3.2.2.3. Параметр «Размер переменной» (Amount byte)**

В параметре «**Размер переменной**» (**Amount byte**) задают размер строки в байтах, в соответствии с которым будет приниматься и передаваться информация по сети.

## **3.2.3. Шаблоны применяемых устройств**

В **Modbus (Master)** реализована возможность вставлять шаблоны с наиболее часто применяемых устройств – МВА, МВУ и МДВВ. При вставлении шаблона пользователь получает настроенные каналы выходов и выходов с настроенными номерами регистров, командами и пр., и ему остается в

универсальном устройстве настроить тип сервера, адрес устройства, выбрать режим работы и задать период опроса.

**Примечание.** Особенности **CoDeSys** не допускают удаление настроенного в шаблоне канала по команде «Delete». По этому, если пользователю не нужны какие-то каналы и для понижения расхода памяти области ввода-вывода, канал заменяется на специальный модуль **Empty**. При этом определенный параметр фактически исключается из опроса.

Шаблон может быть дополнен дополнительными каналами, добавление каналов производится аналогично добавлению каналов в Универсальное устройство Modbus.

Окно подключения шаблонов **Modbus (Master)**» представлено на рис. 3.12.

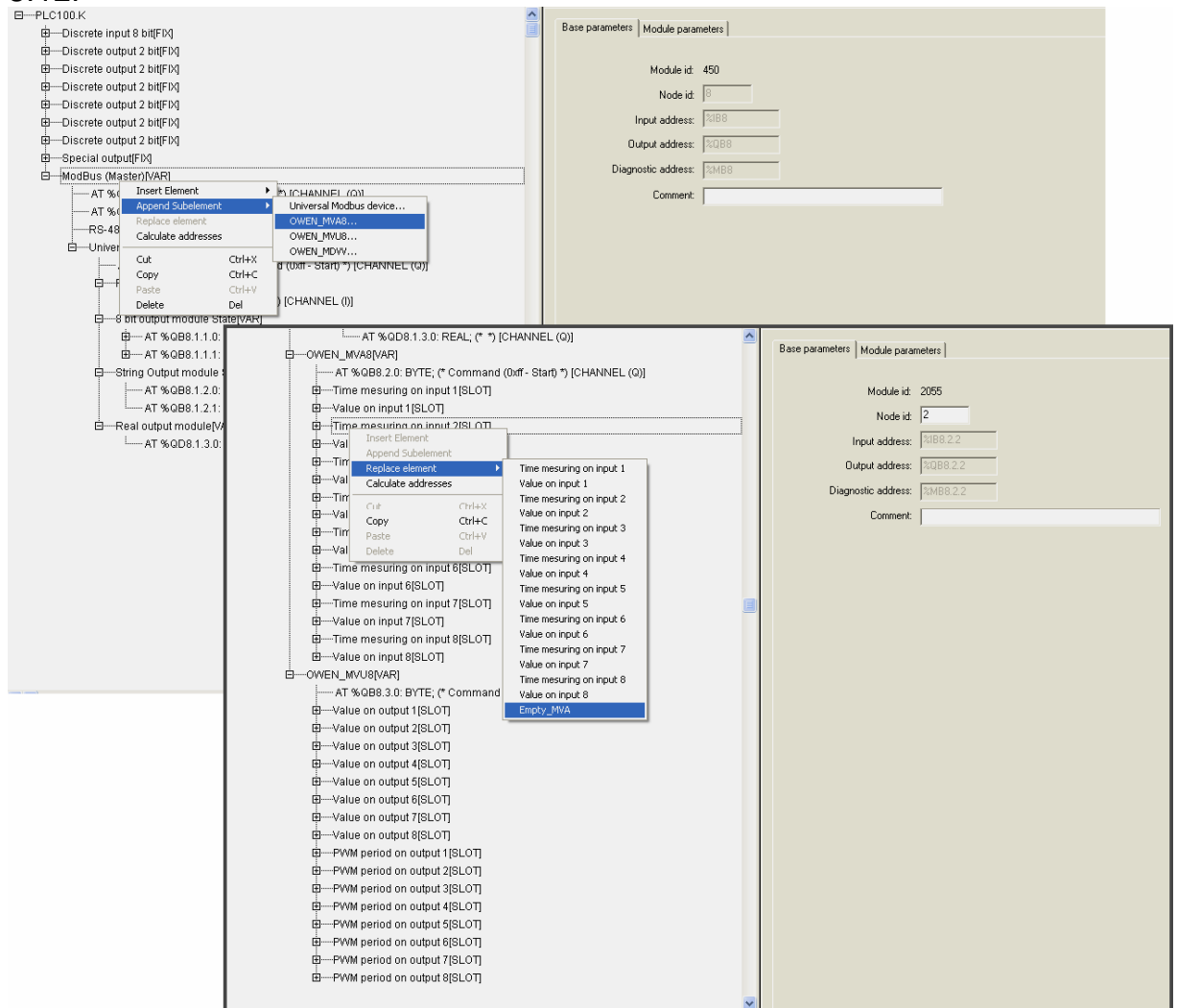


Рис. 3.12. Окно подключения шаблонов **Modbus (Master)**»

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

«Архиватор» – специализированный модуль, призванный облегчить программисту, пишущему программу, архивирование каких-либо необходимых данных. Архивируемые данные могут храниться на **Flash** диске ПЛК или выводиться через коммуникационный интерфейс. Например, к ПЛК через последовательный интерфейс может быть подключен принтер, и задана распечатка определенных значений. К модулю по умолчанию подключен один из интерфейсных портов, через который будут передаваться архивные данные. Пользователь может заменить интерфейсный порт на нужный, воспользовавшись командой контекстного меню **Replace element**.

#### Параметры модуля:

«Режим проведения архивации» (**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).

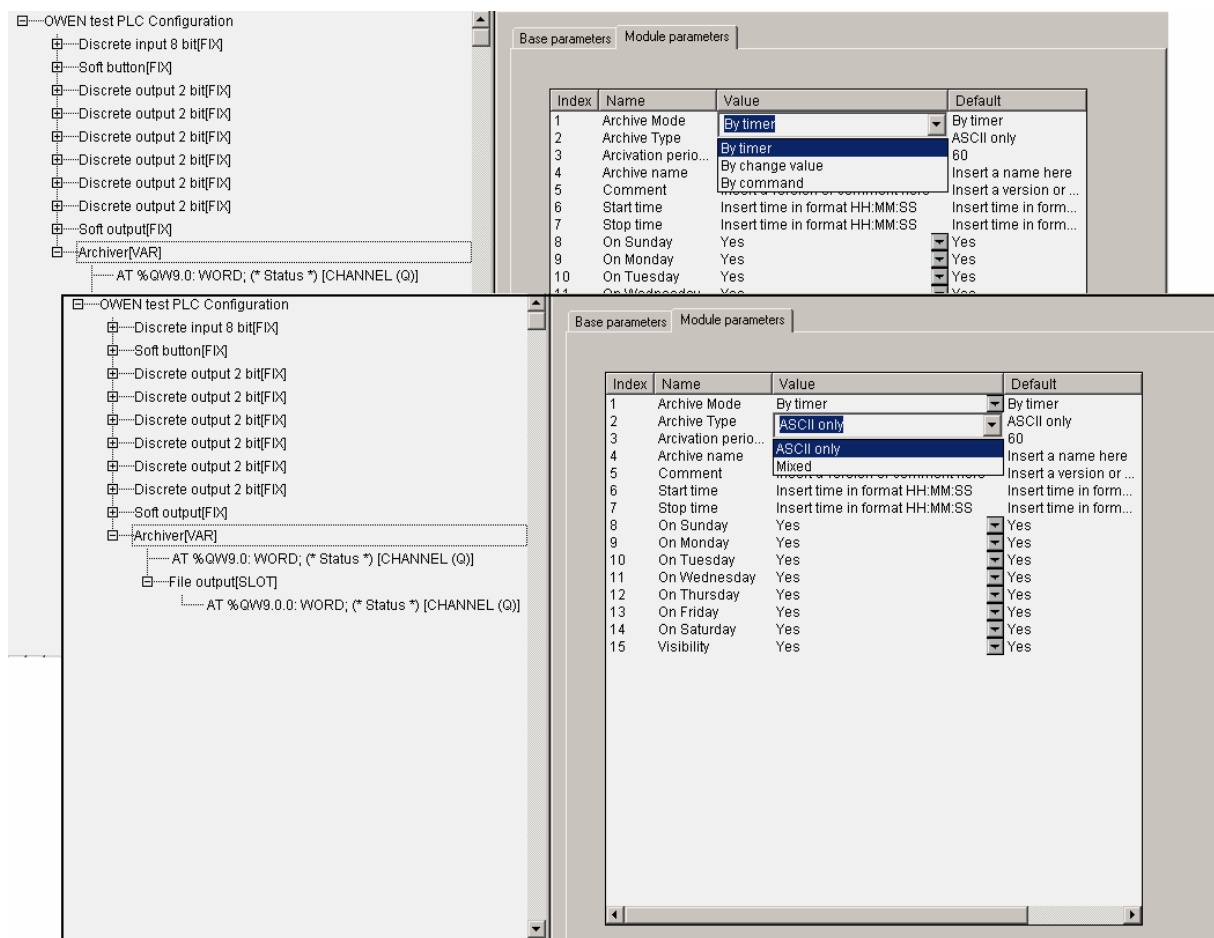
«Понедельник» (**On Sunday**) ... «Воскресенье» (**On Saturday**) – всего семь параметров – назначается день (дни) недели, когда будет производиться архивация. Значения выбираются из списка «**yes**» и «**no**», значение по умолчанию – «**yes**» (см. п. 3.3.7).

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

#### Список подмодулей:

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

Окна настройки параметров модуля «Архиватор» представлены на рис. 3.13.

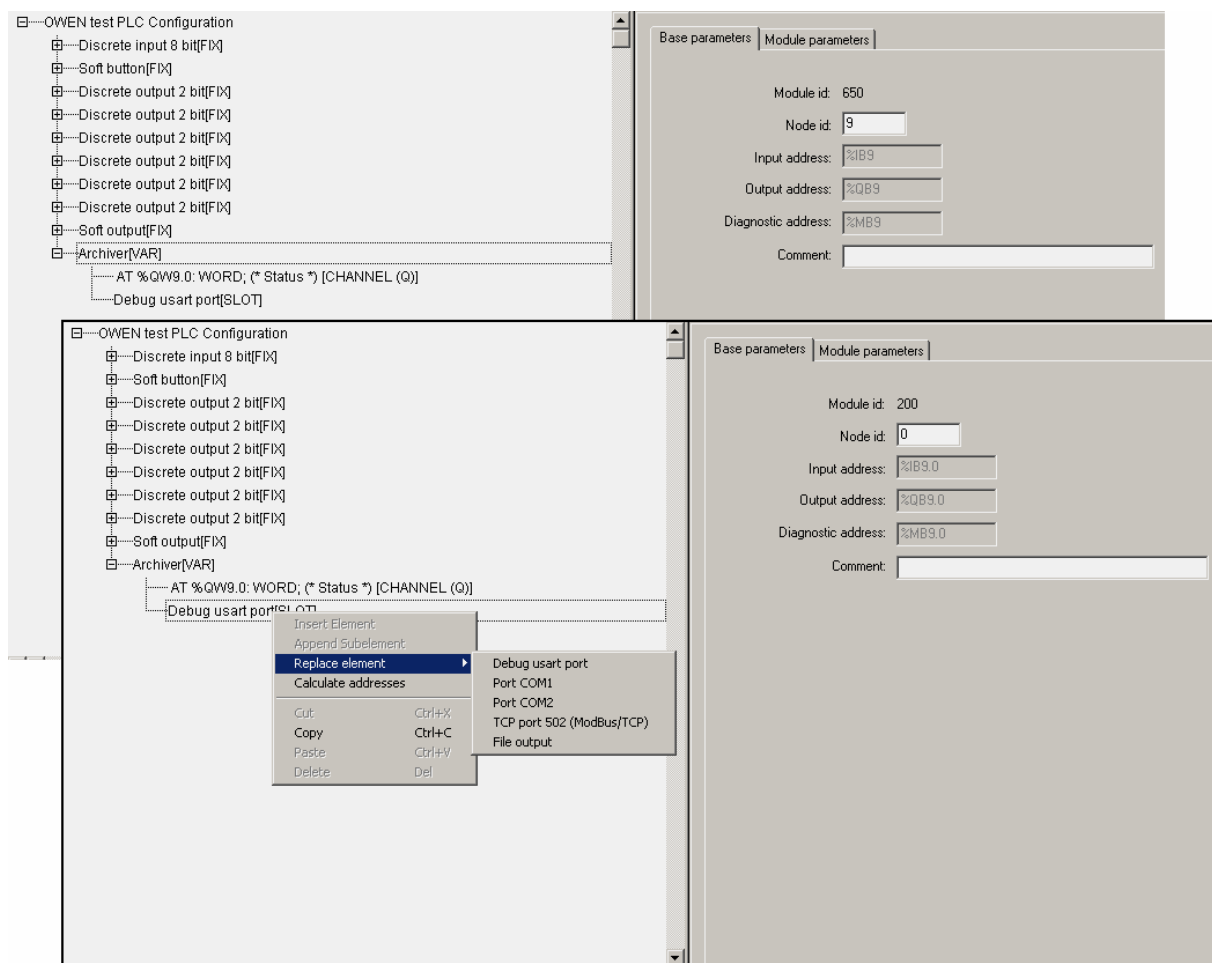


**Рис. 3.13. Окна настройки параметров модуля «Архиватор»**

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**». Так 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 необходимо задать имя – **Variable Name**, – которым потом переменная будет именоваться в архивном файле.

В модуле «**Архиватор**» имеется переменная «**Status**», отображающая статус архива и возникающие ошибки. Коды ошибок ПЛК представлены в Приложении В.

Окно конфигурирования модуля «**Архиватор**» и вид окна при замене интерфейсного порта представлены на рис. 3.14. Настройка последовательных интерфейсов и интерфейса TCP стандартная и описана в п. 3.2.



**Рис. 3.14. Окна настройки модуля «Архиватор» и замены интерфейсного порта**

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

В параметре **«Режим проведения архивации» (Archive Mode)** – выбирают режим архивации из трёх 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 только этой переменной. 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 архива» (Archive Type)

В параметре «Тип архива» (Archive Type) задают тип записи архивируемых данных из двух 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 именами переменных, архива, временные данные в удобном для чтения виде, а все архивируемые переменные записываются в бинарном виде.

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

В параметре «Период архивации, сек» (Period of Archiving) задают периодичность обновления данных архива при работе модуля в режиме «по таймеру».

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

В параметре «Имя архива» (Archive Name) задают имя архива, которое записывается в начале файла.

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

В параметре «Комментарий архива» (Comment) задают 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)

В параметре «Время начала архивации» (Start time) задают время старта архивации.

В параметре «Время остановки архивации» (Stop time) задают время остановки.

---

**Внимание!** Описываемые параметры, задающие временные рамки процесса архивирования, независимы друг от друга, т.е. один или оба параметра могут быть не заданы. 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)

В параметрах от **On Sunday** до **On Saturday** – всего семь параметров – задают день недели, в который будет производиться архивация.

---

**Внимание!** Установлен приоритет между всеми условиями старта и остановки архивирования. Главный приоритет имеет переменная **File Status**, если в ней записана команда «**стоп**», то операция в любом случае прекратится, если «**старт**» – она будет выполнена минимум один раз. Затем по приоритету идет день недели и, далее, время старта и время останова.

---

### 3.3.8. Подмодуль интерфейсного порта (File Output)

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

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

Подмодуль «**File Output**» имеет собственную переменную «**File Status**», в которой сохраняется информация о работе подмодуля. Коды ошибок 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).

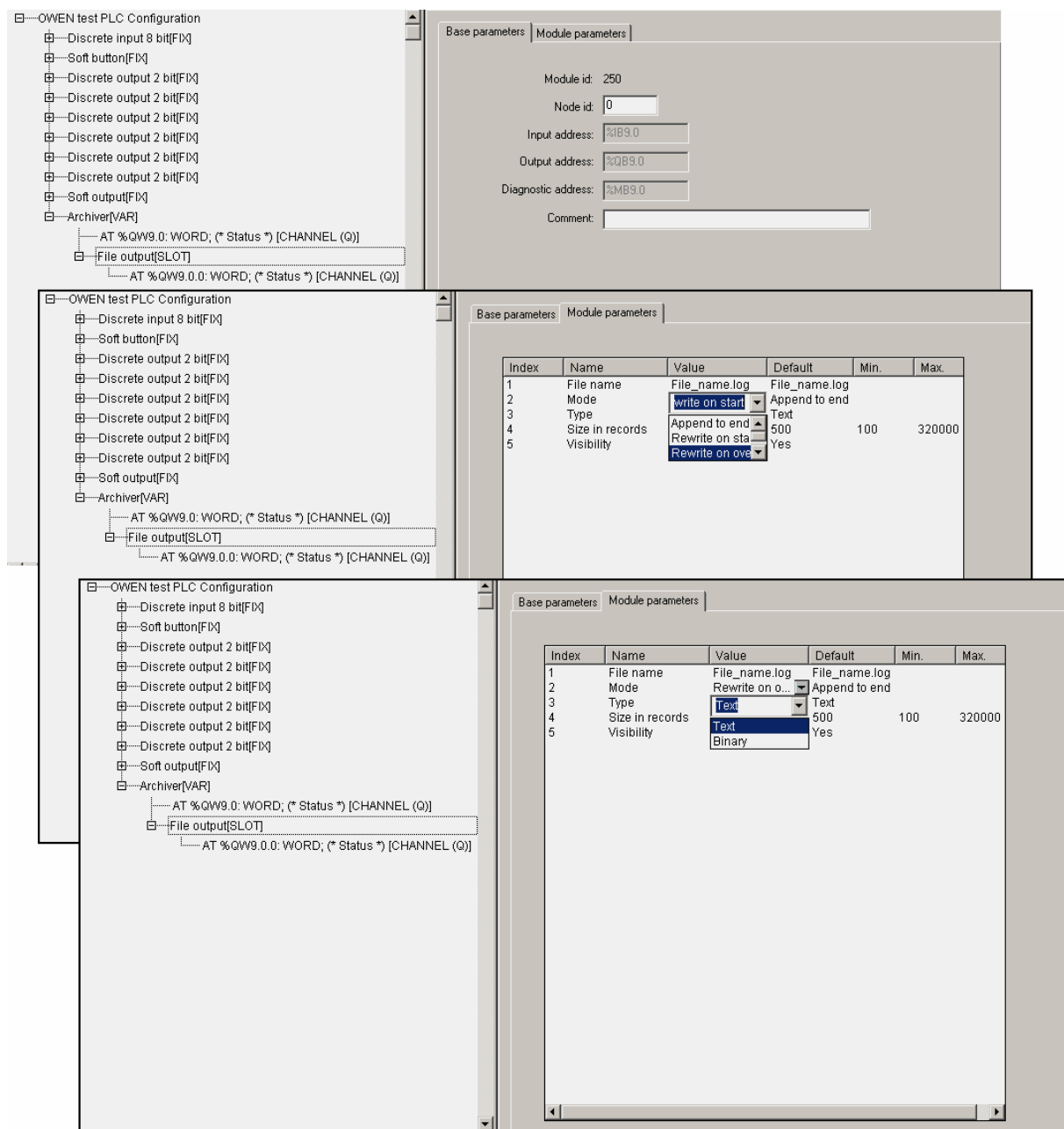
«**Режим работы модуля**» (**Mode**) – значения выбираются из списка «**Append to end**», «**Rewrite on start**», «**Rewrite on oversize**» и «**Shift Mode**», значение по умолчанию – «**Append to end**» (см. п. 3.3.8.2).

«**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 – «**no**» (см. п. 2.1.2).

Окна настройки подмодуля **File Output** представлено на рис. 3.15.



**Рис. 3.15. Окна настройки подмодуля File Output**

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

В параметре «Имя файла» (File name) задают имя файла, в который будет записываться архивная информация.

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

В параметре «**Режим работы модуля**» (**Mode**) выбирают вариант архивации информации из четырёх возможных:

- **Append to end** – «**добавить в конец**» – информация добавляется в конец файла, и, как только файл переполняется, запись прекращается. Файл имеет ограничение либо по размеру (в байтах), либо по количеству записей (задаваемому в параметре **Max file size**);
- **Rewrite on start** – «**перезапись при старте**» – старый файл стирается при старте ПЛК или загрузке новой конфигурации и начинается запись файла с самого начала;
- **Rewrite on oversize** – «**перезапись старого файла при превышении заданного размера**» – файл стирается при достижении им заданного размера, 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 подсчитывается размер файла:

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

### **3.3.8.4. Параметр «Максимальный размер файла» (Max file size)**

В параметре «**Максимальный размер файла**» (**Max file size**) устанавливают ограничение размера 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 «**1 1 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 – «**no**» (см. п. 2.1.2).

#### Список подмодулей:

**Owen (FIX)** (см. п. 3.4.4).

В **PLC Configuration** может быть добавлено несколько модулей **Owen (Slave)**. В этом случае один ПЛК представляется в сети как несколько приборов с разными адресами и именами.

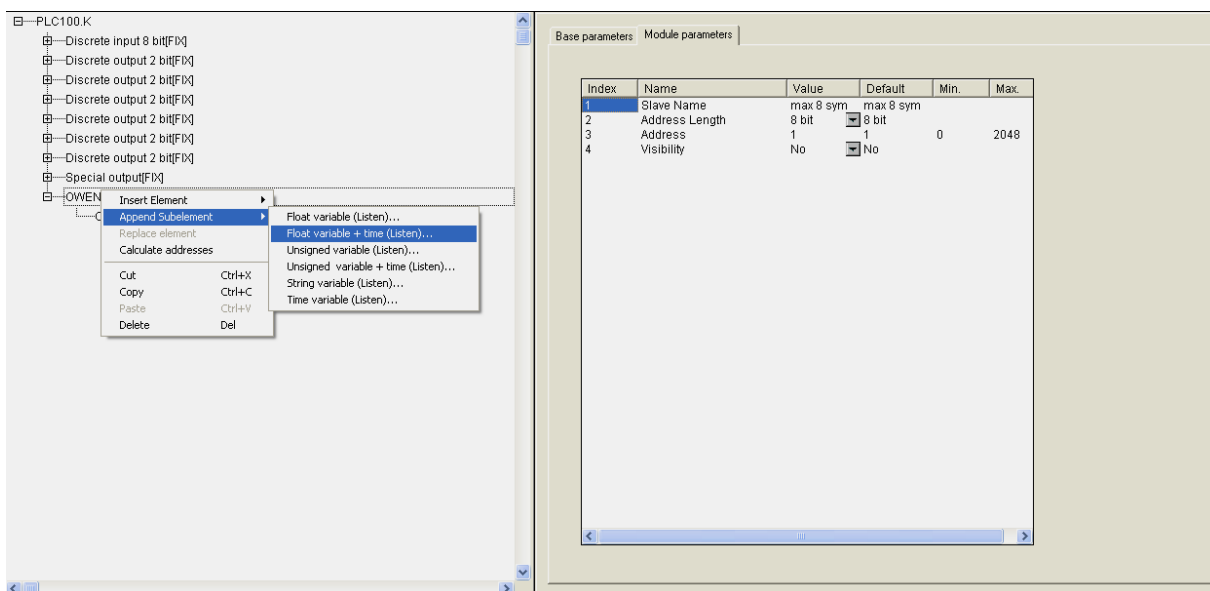
Переменные, которыми будет обмениваться ПЛК по протоколу ОВЕН, выбираются пользователем функцией контекстного меню **Append Subelements**.

---

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

---

Окно конфигурирования модуля **Owen (Slave)** представлено на рис. 3.16.



**Рис. 3.16. Окно конфигурирования модуля Owen (Slave)**

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

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

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

В параметре «Длина адреса устройства» (**Address Length**) задают длину адреса в битах. Приборы разработки ПО «ОВЕН» поддерживают два варианта адресов – 8-ми и 11-ти битовые (задаётся тот же вариант адреса, 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 в свою очередь подключается коммуникационный интерфейс.

---

**Внимание!** Набор параметров настройки интерфейса совпадает с набором параметров настройки подмодуля **Modbus (FIX)** (см. п. 3.1.2.). Особенности следующие: в параметр «**Тип протокола обмена**» (**FrameOriented**) должно быть выставлено значение «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 – восемь типов):

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

---

#### Параметры переменных протокола OVEN:

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

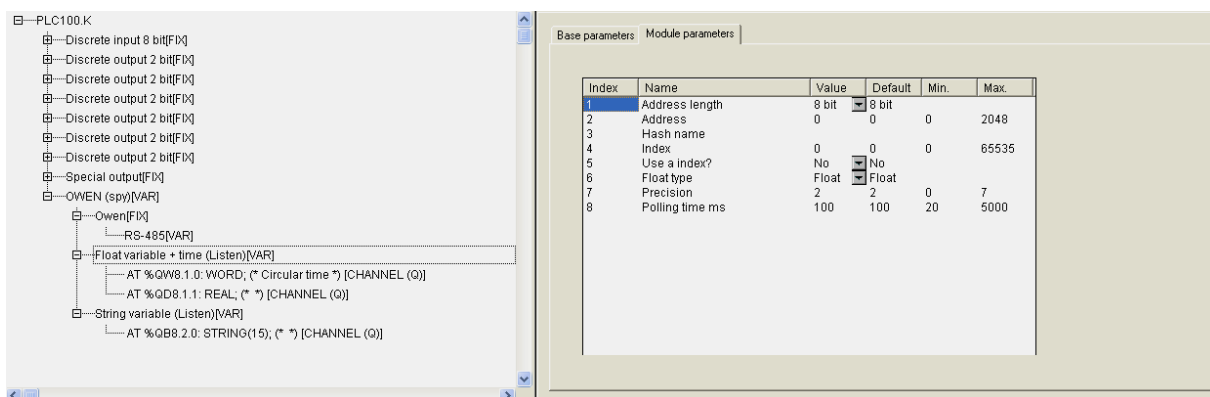
«**Индекс прибора**» (**Index**) – диапазон значений от 0 до 65535, значение по умолчанию – 0 (см. п. 3.4.5.2).

«**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).

Окно задания параметров переменных протокола OVEN проиллюстрированы на рис. 3.17.



**Рис. 3.17. Окно задания параметров переменных протокола OWEN**

**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 OWEN.

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 «Сетевое имя переменной» (Hash name)**

В параметре «Сетевое имя переменной» (**Hash name**) указывается сетевое 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 обмене по сети RS-485.

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

В параметре «Индекс прибора» (**Index**) задают индекс прибора, в параметре «Использовать индекс?» (**Use a index?**) задают 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 вид 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)**

В параметре «Точность» (**Precision**) задают точность для параметров с фиксированной точкой: если выбирается значение точности «2», то число 10,12 именно так и будет передано. При значении «1» – 10,1, при значении «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 Owen – аналогична технологии применения модуля **Owen (Slave)**.

Отличительной особенностью **Owen (Spy)** является то, что это модуль, который не отвечает ни на чьи запросы, а только прослушивает обмен данными в сети RS-485. Модуль может быть настроен таким образом, что, при опросе Мастером сети какого-либо устройства, ответ устройства прослушивается модулем **Owen (Spy)** 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 прослушивает 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 – «**no**» (см. п. 2.1.2).

Окно конфигурирования модуля **Owen (Spy)** со вставленной переменной см. на рис. 3.17.

#### 3.5.1. Переменные протокола ОВЕН для Owen (Spy)

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).

«**Сетевое имя переменной**» (**Hash name**) – 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. Применение параметров **«Индекс прибора» (Index)**, **«Использовать индекс?» (Use a index?)**, **«Тип числа с плавающей точкой» (Float Type)**, **«Точность» (Precision)** соответствует описанному для модуля **Owen (Slave)**.

2. В параметре **«Сетевое имя переменной» (Hash name)** задают имя переменной ведомого устройства, опрашиваемого Мастером. Если в ответе ведомого устройства фигурирует это имя, то данные из ответа будут записаны в переменную (канал) модуля **Owen (Spy)**.

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

---

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

В параметре **«Длина адреса» (Address Length)** задают в битах размер адреса ведомого устройства, ответ которого необходимо прослушать. Приборы разработки ПО «ОВЕН» поддерживают два варианта адресов – 8-ми и 11-ти битовые.

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

В параметре **«Адрес устройства» (Address)** – задают адрес ведомого устройства, ответ которого необходимо прослушать. Значение может быть в диапазоне от 0 до 255 или от 0 до 2047, в зависимости от заданного размера адреса.

### 3.6. Модуль «Owen (Master)»

Функционирование ПЛК по протоколу **Owen** в режиме ведущего, 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 – «**no**» (см. п. 2.1.2).

В модуле **Owen (Master)** имеются несколько переменных, в которые записывается информация о функционировании опрашиваемых приборов и о произошедших ошибках.

#### Каналы модуля:

«**Последний адрес**» (**Last address**) – последний адрес прибора, по которому обращался **Owen (Master)**.

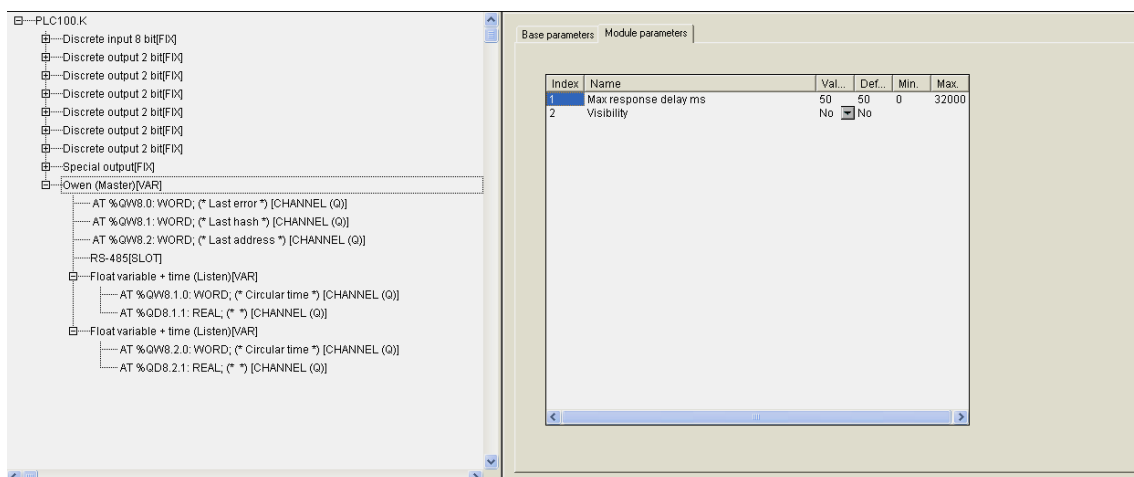
«**Код последней ошибки**» (**Last error**) – код ошибки, которая произошла при последнем опросе.

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

«**Старт/Стоп**» (**Start/Stop**) – используется для управления включением/выключением работы модуля мастера: если в канал записывается значение **0x00FF**, то происходит старт работы модуля, если же в канал записано значение **0x00FE**, то происходит его остановка и прекращение всех посылок в сеть. Если модуль уже запущен, то повторная запись в канал значения **0x00FF** приводит к внеочередному запросу очередной переменной протокола ОВЕН.

Коды ошибок работы ПЛК и пояснения к ним представлены в Приложении В.

Окно конфигурирования модуля **Owen (Master)** представлено на рис. 3.18.



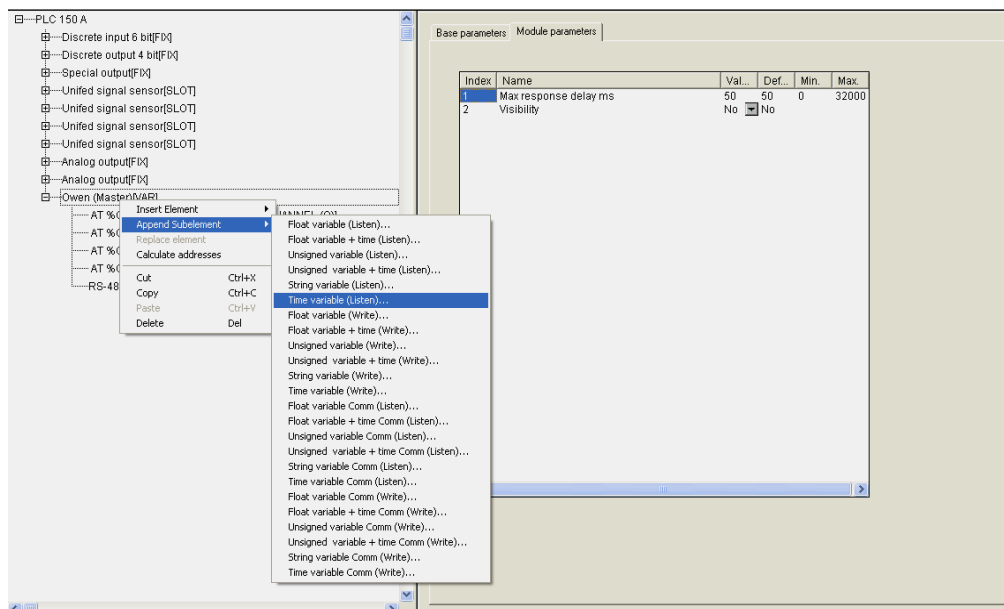
**Рис. 3.18. Окно конфигурирования модуля Owen (Master)**

В отличие от **Owen (Slave)** в **Owen (Master)** отсутствует подмодуль **Owen (FIX)** и по умолчанию вставлен коммуникационный интерфейс. Пользователь может заменить тип интерфейса командой **Replace element** контекстного меню.

### 3.6.1. Переменные протокола ОВЕН для Owen (Master)

В модуле **Owen (Master)** используются переменные, описанные в п. 3.4.5. Кроме того, реализована возможность работы по команде: вставляются переменные протокола ОВЕН с обозначением «**Comm**», которые имеют дополнительный командный управляющий канал.

Переменные протокола ОВЕН в модуле **Owen (Master)** представлены на рис. 3.19.



### Рис. 3.19. Переменные протокола ОВЕН в Owen (Master)

Переменные протокола ОВЕН подразделяются на два типа: предназначенные для чтения (обозначаемые **Listen**) и предназначенные для записи (обозначаемые **Write**).

#### 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).

«**Сетевое имя переменной**» (**Hash name**) – 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).

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

«**Режим работы**» (**Work mode**) – значения выбираются из списка «**По времени**» (**Polling time**), «**По изменению значения переменных**» (**Value change**), «**По времени и по изменению значения переменных**» (**Both**) и «**По команде**» (**By Command**), значение по умолчанию – «**Polling time**» (см. п. 3.6.1.2).

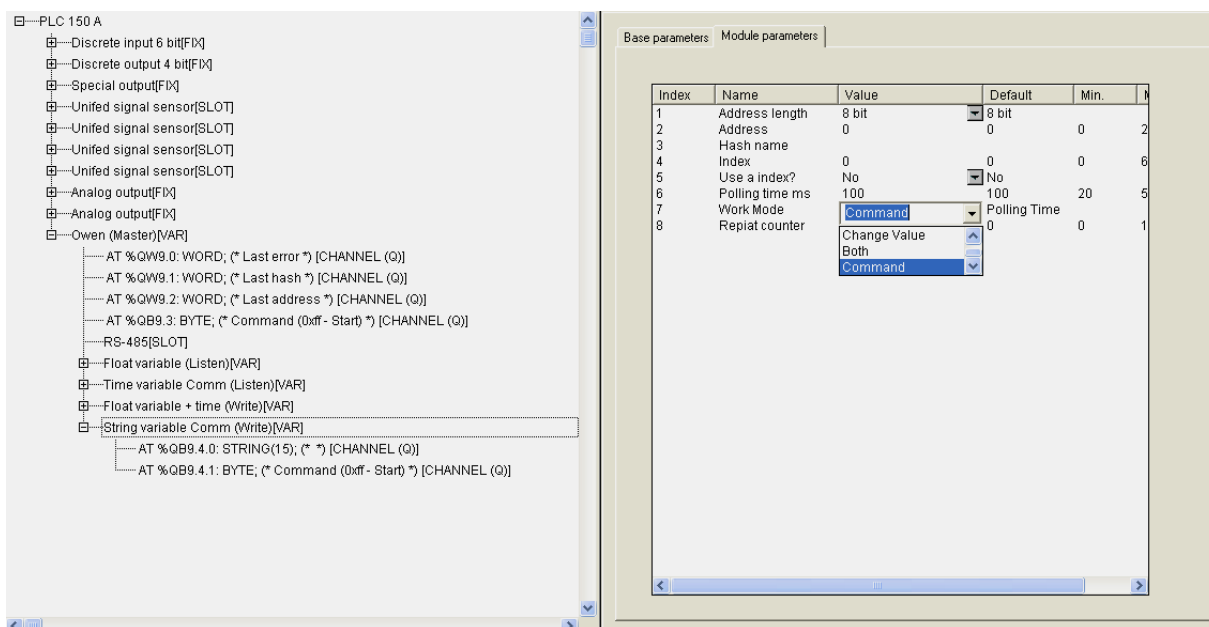
---

**Примечание.** В Мастере, когда он работает в режиме «**По изменению значения переменных**» или «**По команде**», нельзя ставить значение параметра **Polling time** слишком маленьким. По умолчанию его значение 100 мс. Однако, если на реальном проекте будет замечено, что Мастер при загрузке программы или при **Login** формирует лишние пакеты и/или запросы, которых не должно быть, значение параметра увеличивают (до 200, 300 и т.д.) до предотвращения появления ложных пакетов.

---

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

Задание параметров при вводе переменной протокола ОВЕН типа **String variable** проиллюстрировано на рис. 3.20.



**Рис. 3.20. Параметры переменной протокола ОВЕН типа String variable**

### Примечания.

1. Применение параметров «**Индекс прибора**» (**Index**), «**Использовать индекс?**» (**Use a index?**) соответствует описанному для модуля **Owen (Slave)**.

2. Применение параметра «**Сетевое имя переменной**» (**Hash name**): в нём задают имя переменной ведомого прибора, которую запрашивает модуль **Owen (Master)**. Имена переменных ведомых приборов указываются в руководствах по эксплуатации этих приборов. Имя преобразуется в ПЛК в **Hash**-код, который затем и используется при обмене по сети RS-485.

3. Назначение параметра «**Период опроса устройства, мс**» (**Polling time, ms**) аналогично назначению одноименного параметра, используемого 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, значения которых часто запрашиваются, 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)**

В параметре «**Максимальное время задержки ответа, мс**» (**Max response delay, ms**) задают 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.

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

В параметре «**Режим работы**» (**Work mode**) – выбирают режим работы модуля **Owen (Master)** при опросе внешних устройств:

- **Polling time** – «**по времени**» – контролируемые устройства опрашиваются с периодичностью, заданной в параметре «**Период опроса устройства**» (**Polling time**);
- **Value change** – «**по изменению значения переменных**» – модуль **Owen (Master)** генерирует запрос устройству при изменении значений выходных переменных модуля;
- **Both** – «**оба варианта**» – опрос производится с временным интервалом, заданным в параметре **Polling time** и тогда, когда изменяются значения выходных переменных;
- **By Command** – «**по команде**» – производится однократная посылка запроса, когда в командный канал **Command** переменной, имеющей такой канал, записывается значение **0x00FF**.

---

#### **Внимание.**

1. Для считываемых переменных (тип «**Listen**») доступен только режим «**По времени**» (**Polling time**), если же есть командный канал, то и режим «**По команде**» (**Command**). Для записываемых переменных (тип «**Write**») доступны режимы «**По времени**» (**Polling time**), «**По изменению значения переменных**» (**Value change**) и «**По времени и по изменению значения переменных**» (**Both**), если же есть командный канал, то и режим «**По команде**» (**By Command**).
  2. Для переменных с командным каналом при работе в режиме «**По команде**» (**By Command**) управление осуществляется следующим образом: первая посылка значения **0x00FF** в командный канал включает функционирование этой переменной, повторная посылка значения **0x00FF** инициирует проведение опроса. Аналогично опрос инициируется для переменных с командным каналом при работе в других режимах. При посылке в командный канал значения **0x00FE** переменная выключается из цикла опроса мастера.
-



### 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 – «**no**» (см. п. 2.1.2).

Окно конфигурирования модуля **DCON (Master)** представлено на рис. 3.21.

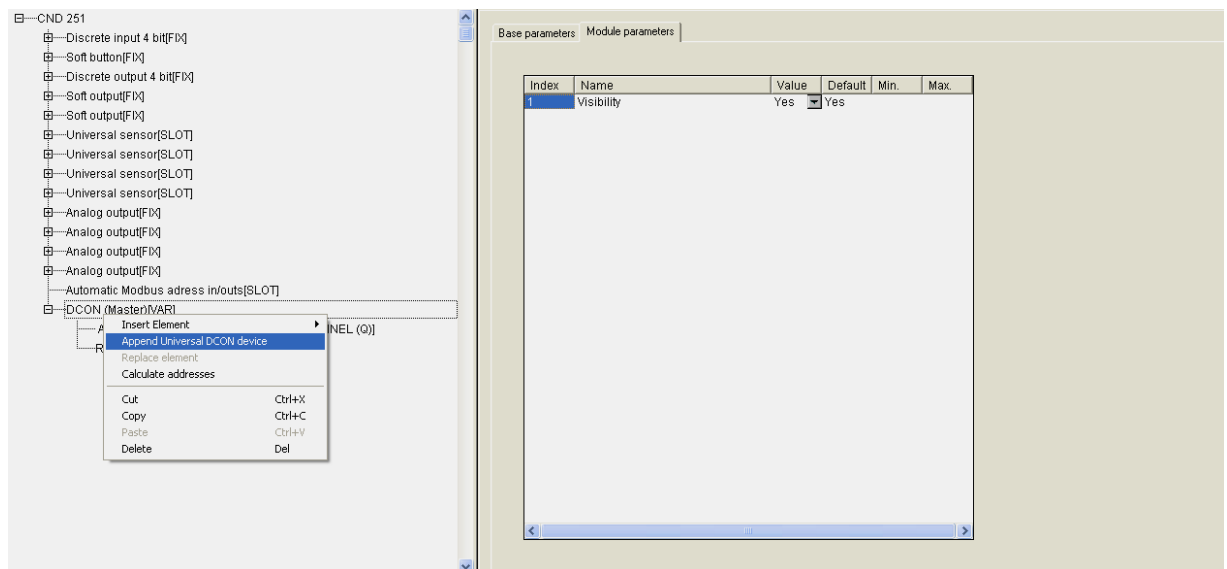


Рис. 3.21. Окно конфигурирования модуля **DCON (Master)**

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

В модуле **DCON (Master)** имеется переменная «**Код последней ошибки**» (**Last error**), в которую записывается код последней ошибки, произошедшей при опросе ведомых приборов.

Организация работы **DCON (Master)** аналогична описанной выше для **Modbus (Master)**: установка и настройка устройств, опрашиваемых модулем, производится через установку модуля «**Универсальное устройство DCON**» и задание необходимых параметров. Его подключение выполняется командой «**Append Universal DCON Device**», вызываемой из контекстного меню.

### 3.7.1. Модуль «Универсальное устройство DCON» (Universal DCON Device)

Окно конфигурирования модуля «Универсальное устройство DCON (Universal DCON Device)» представлено на рис. 3.22.

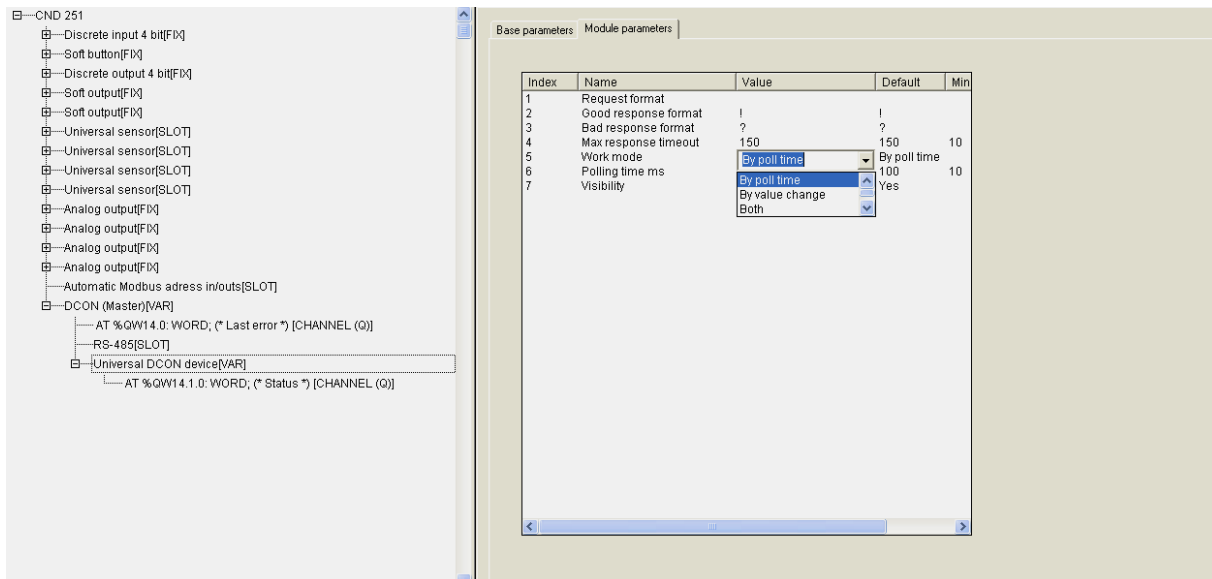


Рис. 3.22. Окно конфигурирования модуля «Универсальное устройство DCON»

При вызове модуля универсального устройства появляется незаполненный шаблон, параметры и переменные которого не привязаны к конкретному внешнему устройству. При заполнении полей параметров значениями, определении переменных входов-выходов, устройство определит конкретный тип вставляемого устройства.

Для конфигурирования модуля 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. Значения 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 записываться во входные переменные модуля (**input variables**);
- задать прочие параметры модуля.

У устройства **DCON** есть переменная **Статус (Status)**. В переменной **Статус** фиксируются ошибки обмена с внешним устройством или хранится код, свидетельствующий об успешном обмене.

Переменная **Статус (Status)** может использоваться для управления включением/выключением работы 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)**» – формат запроса, может быть любым, ограничения не накладываются (см. п. 3.7.1.1).

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

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

«**Максимальное время ответа (Max response timeout)**» – максимальное допустимое время ответа. Значение сверху не ограничено, может быть любым, в т.ч. дробным, 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 – «**no**» (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).

---

**Примечание.** В Мастере, когда он работает в режиме «**По изменению значения переменных**» или «**По команде**», нельзя ставить значение параметра **Polling time** слишком маленьким. По умолчанию его значение 100 мс. Однако, если на реальном проекте будет замечено, что Мастер при загрузке программы или при **Login** формирует лишние пакеты и/или запросы, которых не должно быть, значение параметра увеличивают (до 200, 300 и т.д.) до предотвращения появления ложных пакетов.

---

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

В протоколе **DCON** при организации опроса устройств создается строка запроса, и, при ее посылке, опрашиваемое устройство может вернуть два 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 ОБЕН ПЛК по протоколу DCON есть три варианта работы: без расчета контрольных сумм, с расчетом контрольных сумм путем сложения значений всех символов и с расчетом контрольных сумм 8-ми битовых. 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.

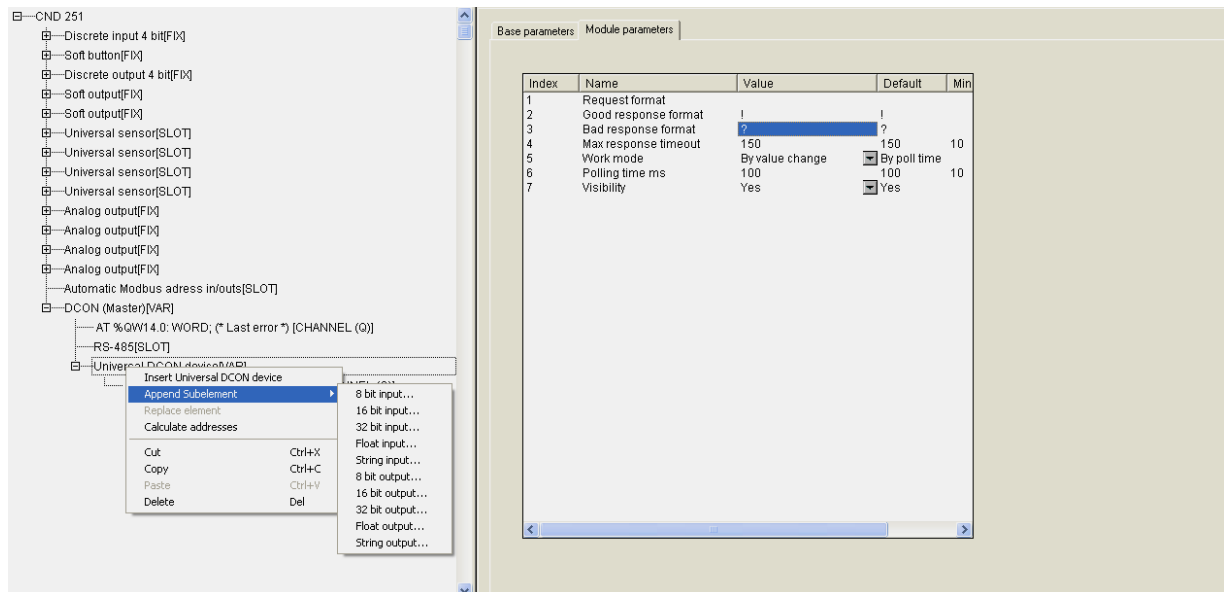
Если запрос жестко фиксированный, т.е. в строке не содержатся изменяемые данные, то строка набивается без каких-либо команд, в таком виде отсылается, при этом может быть добавлена контрольная сумма.

Аналогично с ответом: если приходит строка, не содержащая каких-либо данных (в конце может быть контрольная сумма), это означает, что прибор работает, реагирует и пр., т.е. сам факт получения ответа от прибора уже является информацией.

В модуль «**Универсальное устройство DCON**» добавляются входные и выходные переменные. Во входных переменных (**input variables**) значения записываются, когда приходит ответ от опрашиваемого прибора и в них значения фиксируются, выходные переменные (**output variables**) используются при формировании запроса.

В модуль «**Универсальное устройство DCON**» можно добавлять переменные различного размера (8, 16 и 32 бита и 16 байтовая строка) и типа (целое беззнаковое, с плавающей точкой или строка). Тип и порядок расположения входных и выходных переменных в модуле должны соответствовать строкам команд в полях «**Request format**» и «**Good response format**» (см. рис. 3.20). Добавление переменной производится функцией контекстного меню **Append Subelements** (см. рис. 3.23).

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



**Рис. 3.23. Окно добавления переменных модуля «Универсальное устройство DCON (Master)»**

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

В параметре «**Максимальное время ответа**» (**Max response timeout**) задают время, за которое опрашиваемый прибор должен ответить на запрос **DCON (Master)**. Если в течение этого времени прибор не отвечает, то считается, что он 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)**

В параметре «**Режим работы**» (**Work mode**) – выбирают режим работы модуля **DCON (Master)** при опросе внешних устройств:

- **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 варианта» – опрос производится с временным интервалом, заданным в параметре **Polling time** и тогда, когда изменяются значения выходных переменных;
- **By command** – «по команде» – производится однократная посылка запроса, когда в канал **Статус (Status)** модуля **Универсальное устройство DCON** записывается значение **0x00FF**.

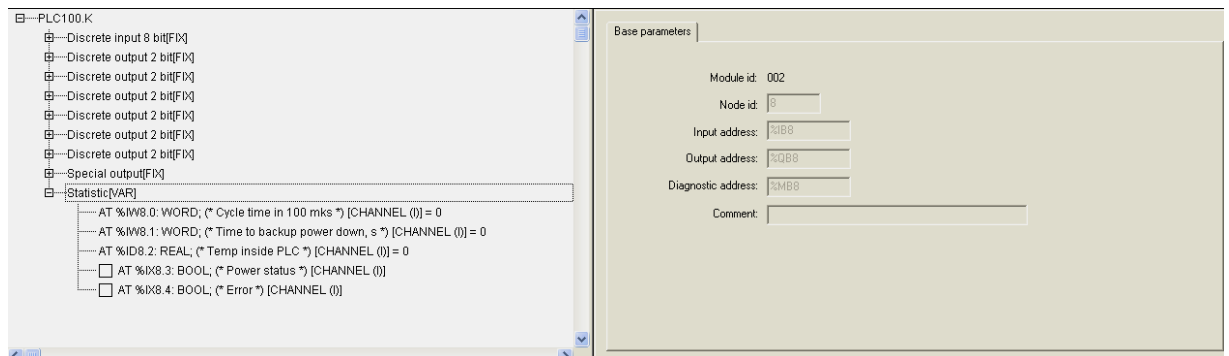
### 3.8. Модуль статистики (Statistic)

**Модуль статистики (Statistic)** предназначен для выдачи 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);
- три 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, заданного в параметрах работы ПЛК параметра **MinCycleLength**, то оказывается, что программа пользователя слишком требовательна к ресурсам, и этот параметр желательно увеличить, чтобы циклы не перекрывались;
  - **время, оставшееся до выключения ПЛК, сек (Time to backup power down, s)**, отражает ресурс времени работы ПЛК от аккумуляторных батарей (с отключенным внешним питанием). Характеристика оценочная, точный учет влияния всех факторов (температура ПЛК и внешней среды, точность измерения и пр.) затруднителен. В случае работы ПЛК от сети аккумуляторная батарея заряжается, и этот параметр косвенно указывает на процесс зарядки (600 с 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.

Окно конфигурирования **модуля статистики (Statistic)** представлено на рис. 3.24.



**Рис. 3.24. Окно конфигурирования модуля статистики**



## 3.9. Модуль «Константа» (Constant value)

**Модуль «Константа» (Constant value)** – содержит значение константы (задаваемой в параметре модуля). При загрузке конфигурации данное значение записывается в канал модуля и может быть считано в программу ПЛК.

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

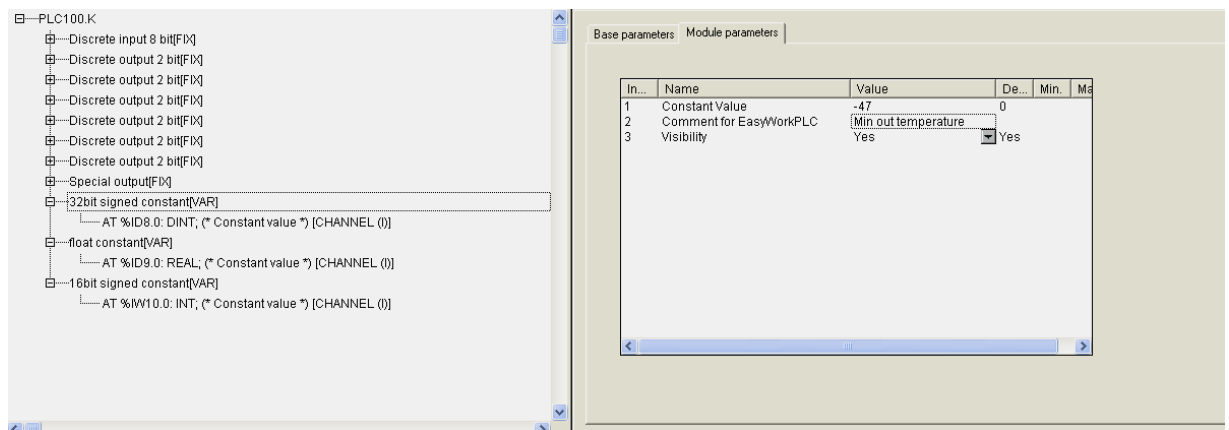
### Параметры модуля:

«**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).

Окно конфигурирования модуля «**Константа» (Constant value)** представлено на рис. 3.25.



**Рис. 3.25.** Окно конфигурирования модуля «Константа» (Constant value)

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

В параметре «**Значение константы» (Constant Value)** устанавливается 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)

В параметре **Комментарий для программы EasyWorkPLC» (Comment for EasyWorkPLC)** записывается вводимый пользователем комментарий к константе 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 константы пользователь может редактировать. Таким образом, какие-то настройки на ПЛК можно поменять, и для этого не нужна среда **CoDeSys**, а используется программа **EasyWorkPLC**.

## Приложение А. Типы датчиков, поддерживаемые аналоговыми входами 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 соединительных проводов. Из-за этого сопротивление проводов вносит большую дополнительную погрешность. Обычно подключение таких термосопротивлений осуществляется по трехпроводной схеме, позволяющей измерять 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 менее 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. Зафиксировать результаты измерения на аналоговом входе контроллера. (Результаты измерения можно вызвать на вкладке, двойным нажатием левой кнопкой манипулятора «мышь» при установке курсора на канале модуля).
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 зафиксировать результаты измерения на аналоговом входе контроллера и удостовериться, что они равны или близки к 0.
8. Подсоединить вместо магазина сопротивлений датчик.

### А.2. Перечень поддерживаемых датчиков

Таблица А.1

#### Датчики и входные сигналы

Тип датчика или входного сигнала	Обозначение в параметре «Type of sensor»	Диапазон измерений
<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

Продолжение табл. А.1

Тип датчика или входного сигнала	Обозначение в параметре «Type of sensor»	Диапазон измерений
ТСП (50П) $W_{100} = 1,3910$	R391_50	-200 °C...+750 °C
ТСМ (Cu 100) $W_{100} = 1,4260$	R426_100	-50 °C...+200 °C
ТСМ (100М) $W_{100} = 1,4280$	R428_100	-190 °C...+200 °C
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 (500М) $W_{100} = 1,4280$	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 (1000М) $W_{100} = 1,4280$	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
<b>Термопары (по ГОСТ Р 8.585-2001) (Termocouple sensors)</b>		
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
ТВР (А-1)	TP_WR1	0 °C...+2500 °C
ТВР (А-2)	TP_WR2	0 °C...+1800 °C
ТВР (А-3)	TP_WR3	0 °C...+1600 °C
ТМК (Т)	TP_T	-200 °C...+400 °C
<b>Унифицированные сигналы постоянного напряжения и тока (по ГОСТ 26.011-80) (Universal sensors)</b>		
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 %
<b>Датчики сопротивления (Universal sensors)</b>		
резистивный (0... 5000 Ом)	RO_5000	0...100 %
<b>Примечания.</b>		
1. $W_{100}$ – отношение сопротивления датчика, измеренное при температуре 100 °C, к его сопротивлению, измеренному при 0 °C ( $R_0$ ).		
2. 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.1-Б5). 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**»).

### Пример 1

Таблица Б.1

#### Описание переменной

Характеристика	Значение
<b>Прибор</b>	Все приборы с аналоговыми входами
<b>Тип переменной</b>	число с плавающей точкой и циклическим временем
<b>Имя параметра</b>	<b>read</b>
<b>Описание</b>	Значение с измерителя
<b>Диапазон</b>	В соответствии с диапазоном аналогового входа
<b>Диапазон индекса</b>	0 .. 6

Таблица Б.2

#### Настройки модуля Owen (Master)

Характеристика	Значение
<b>Тип переменной</b>	<b>Float variable + time</b>
<b>Направление опроса</b>	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

### Пример 2

Таблица Б.3

#### Описание переменной

Характеристика	Значение
Прибор	TPM133, MVA8, TPM151, TPM148
Тип переменной	Тип данных с числом с фиксированной точкой
Имя параметра	itrL
Описание	Период опроса датчика
Диапазон	0,1 ... 30
Диапазон индекса	0 ... 6

Таблица Б.4

#### Настройки модуля Owen (Master)

Характеристика	Значение
Тип переменной	Float variable
Направление опроса	Чтение/Запись (в зависимости от потребностей)
Address Length	8/11 бит в зависимости от настроек сети
Address	Адрес прибора
Hash name	itrL
Index	0-6 (в зависимости от номера канала измерения)
Use a index?	Yes
Float type	Fix point binary
Precision	1
Polling time	В мс, в зависимости от необходимого периода опроса
Вставленные подмодули	Отсутствуют

### Пример 3

Таблица Б.5

#### Описание переменной

Характеристика	Значение
Прибор	TPM133, MVA8, TPM151, TPM148
Тип переменной	Тип данных со знаковым/беззнаковым целым значением
Имя параметра	in-t
Описание	Тип датчика
Диапазон	Не ограничен
Диапазон индекса	0 ... 6

Таблица Б.6

## Настройки модуля Owen (Master)

Характеристика	Значение
Тип переменной	Unsigned variable
Направление опроса	Чтение/Запись (в зависимости от потребностей)
Address Length	8/11 бит в зависимости от настроек сети
Address	Адрес прибора
Hash name	in-t
Index	0...6 (в зависимости от номера канала измерения)
Use a index?	Yes
Polling time	В мс, в зависимости от необходимого периода опроса
Вставленные подмодули	
1	8 bits

## Пример 4

Таблица Б.7

## Описание переменной

Характеристика	Значение
Прибор	Любой прибор ОВЕН
Тип переменной	Тип данных с текстовой информацией
Имя параметра	Dev
Описание	Название прибора
Диапазон	Строка
Диапазон индекса	Нет индекса

Таблица Б.8

## Настройки модуля Owen (Master)

Характеристика	Значение
Тип переменной	String variable
Направление опроса	Чтение (константное значение)
Address Length	8/11 бит в зависимости от настроек сети
Address	Адрес прибора
Hash name	dev
Index	Не используется
Use a index?	No
Polling time	В мс, в зависимости от необходимого периода опроса
Вставленные подмодули	Отсутствуют

Пример 5

Таблица Б.9

Описание переменной

Характеристика	Значение
Прибор	TPM133
Тип переменной	Тип данных с информацией о времени
Имя параметра	<b>t.val</b>
Описание	Время полного хода 3-х позиционного ИМ, мин:сек
Диапазон	1 ... 999
Диапазон индекса	0 ... 7

Таблица Б.10

Настройки модуля Owen (Master)

Характеристика	Значение
Тип переменной	<b>Time variable</b>
Направление опроса	Чтение/Запись (в зависимости от потребностей)
Address Length	8/11 бит в зависимости от настроек сети
Address	Адрес прибора
Hash name	<b>t.val</b>
Index	0-7 (в зависимости от номера канала)
Use a index?	Yes
Less time field	Seconds
Polling time	В мс, в зависимости от необходимого периода опроса
<b>Вставленные подмодули</b>	
<b>1</b>	Seconds
<b>2</b>	Minutes



## Приложение В. Сообщения об ошибках в ПЛК

В Приложении представлены коды и причины возможных ошибок, возникающих в процессе работы модулей, и хранящиеся в переменных этих модулей:

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

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

Коды ошибок передаются через канал **Value**.

При возникновении ошибки, в старших четырёх битах выставляются единицы, в следующих 4-х битах следует код ошибки (см. табл. В.1). Остальные три байта заполняются незначащей информацией.

Таблица В.1

Ошибки работы аналоговых входов

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

### В.2. Коды ошибок модулей «Мастер»

#### В.2.1. Модуль Modbus (Мастер)

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

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

Таблица В.2

## Ошибки работы модуля Modbus (Мастер)

Краткое наименование	Код ошибки		Причины ошибок
	(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 три канала для отображения статуса мастера и возникающих ошибок:

- «**Last error**» – содержит код ошибки (см. табл. В.3)
- «**Last Address**» – содержит адрес последнего опрошенного Мастером устройства
- «**Last HASH**» – содержит Hash-код переменной, которая была опрошена последней.

Таблица В.3

## Ошибки работы модуля ОБЕН (Мастер)

Краткое наименование	Код ошибки		Причины ошибок
	(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*
<b>BAD_ADDRESS</b>	<b>0x8000</b>	<b>32768</b>	Адрес в ответе не соответствует ожидаемому*

\* Эти коды ошибок могут быть вызваны совместной работой нескольких мастеров в одной сети

Таблица В.4

## Коды ошибок приборов в сети ОВЕН

Краткое наименование	Код ошибки в старшем байте		Причины ошибок
	(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		Причины ошибок
	(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>«Запись запрещена групповым атрибутом уровня»</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		Причины ошибок
	(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.
<b>Примечание.</b> Коды ошибок приборов в сети ОВЕН приведены в соответствии с описанием протокола приборов ПО ОВЕН по RS-485			

### В.2.3. Модуль DCON (Мастер)

В модуле **DCON (Мастер)** используется один канал для отображения статуса Мастера и возникающих ошибок:

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

**Таблица В.5**

#### Ошибки работы модуля DCON (Мастер)

Краткое наименование	Код ошибки		Причины ошибок
	(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 работе каждого отдельного устройства DCON выводится в поле «**Status**» универсального устройства DCON. Коды ошибок приведены в таблице В.6.

**Таблица В.6**

#### Ошибки работы универсального устройства DCON

Краткое наименование	Код ошибки		Причины ошибок
	(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		Причины ошибок
	(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.** Переменная **Status** предназначена 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 переменную значение **0xFF**.

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

В модуле «Архиватор» используется один канал для отображения статуса и возникающих ошибок:

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

Таблица В.7

## Ошибки работы модуля «Архиватор»

Краткое наименование	Код ошибки		Причины ошибок
	(hex)	(dec)	
<b>STANDBY</b>	<b>0x0000</b>	<b>0</b>	Модуль «Архиватор» остановлен
<b>RUN</b>	<b>0x0001</b>	<b>1</b>	Модуль «Архиватор» 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 переменную «**Status**» архивирование может быть запущено или остановлено. Коды команд даны в табл. В.8.

Таблица В.8

## Коды команд управления модулем «Архиватор»

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

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

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

Таблица В.9

## Ошибки работы модуля архивирования информации в файл

Краткое наименование	Код ошибки		Причины ошибок
	(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 по протоколу DCON в режиме Master

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

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

Производится опрос первых трёх входов нескольких модулей аналоговых входов IPC-7033.

#### Описание формата обмена с модулем IPC-7033:

##### Формат запроса: #AA[CRC][CR]

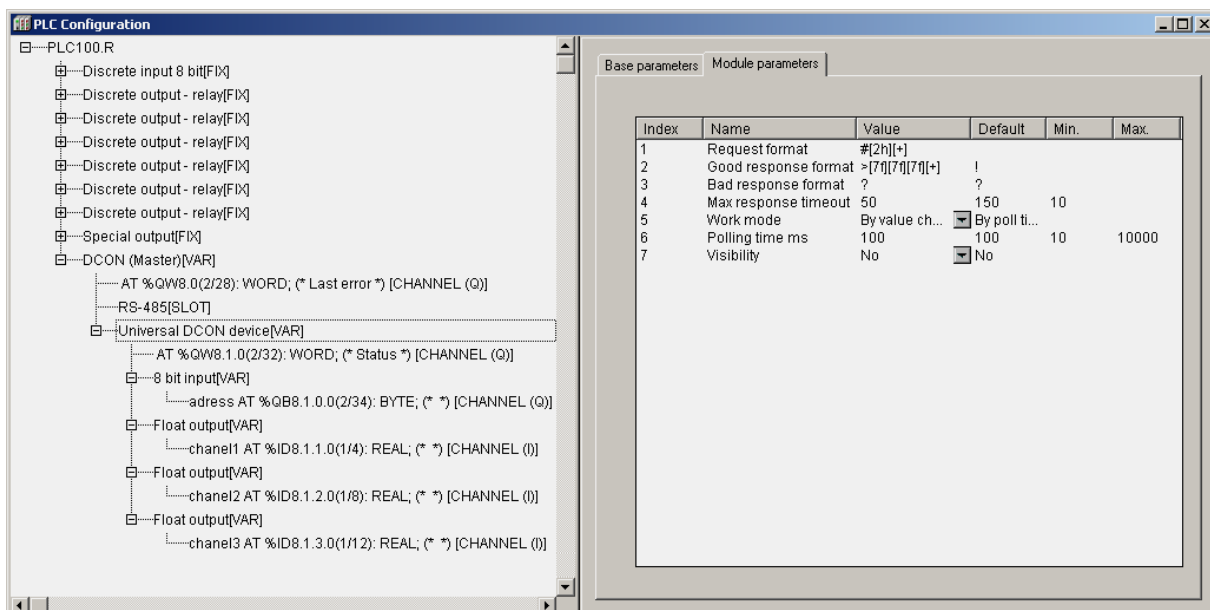
где	#	–	разделитель;
	AA	–	адрес прибора;
	CRC	–	контрольная сумма;
	CR	–	перевод строки.

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

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

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





**Рис. Г. 1. Окно конфигурирования модуля «Universal DCON device» модуля DCON (Master) для опроса входов модулей IPC-7033**

Для задания адреса опрашиваемого прибора используется 8-битовая входная переменная **«Address»**. Значения со входов опрашиваемого модуля IPC-7033 отображаются в трёх 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]** – спецкоманда, указывающая, что в это место запроса подставляется шестнадцатиричный двух символный адрес, значение которого должно быть взято из входной переменной;
  - [+]** – спецкоманда 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]** – спецкоманда, указывающая на то, что семь символов ответа должны быть преобразованы в число с плавающей точкой и результат преобразования должен быть помещен в первую выходную переменную, которая имеет формат **Float**.

**Внимание!** Для следующих спецкоманд [7f] применяется то же правило преобразования, но результаты помещаются во вторую и третью выходные переменные, соответственно.

- [+] – спецкоманда, указывающая на то, что должно быть проанализировано правильность контрольной суммы в принятой посылке. Результат записывается в переменную «**Status**».
- 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 изменению значения одной из входных переменных). Этот режим позволяет генерировать запросы при изменении адреса опрашиваемого прибора. Для генерации одного запроса необходимо записать значение, отличающееся от текущего, во входную переменную «**Adress**». После этого по значению переменной «**Status**» определяется окончание обмена данными с опрашиваемым прибором и корректность данных в выходных переменных.

---

**Внимание!** Чтение переменной «**Status**» и её анализ должны производиться на следующем цикле работы ПЛК после записи нового адреса.

---

Остальные параметры в данном режиме работы не существенны.

## **Г.2. Установка выходного значения модуля аналогового вывода IPC-7021 с периодичностью 1 секунда и при необходимости изменения значения**

Производится запись одного выходного значения у модуля IPC-7021 с шестнадцатиричным адресом 18.

**Описание формата обмена с модулем IPC-7021:**

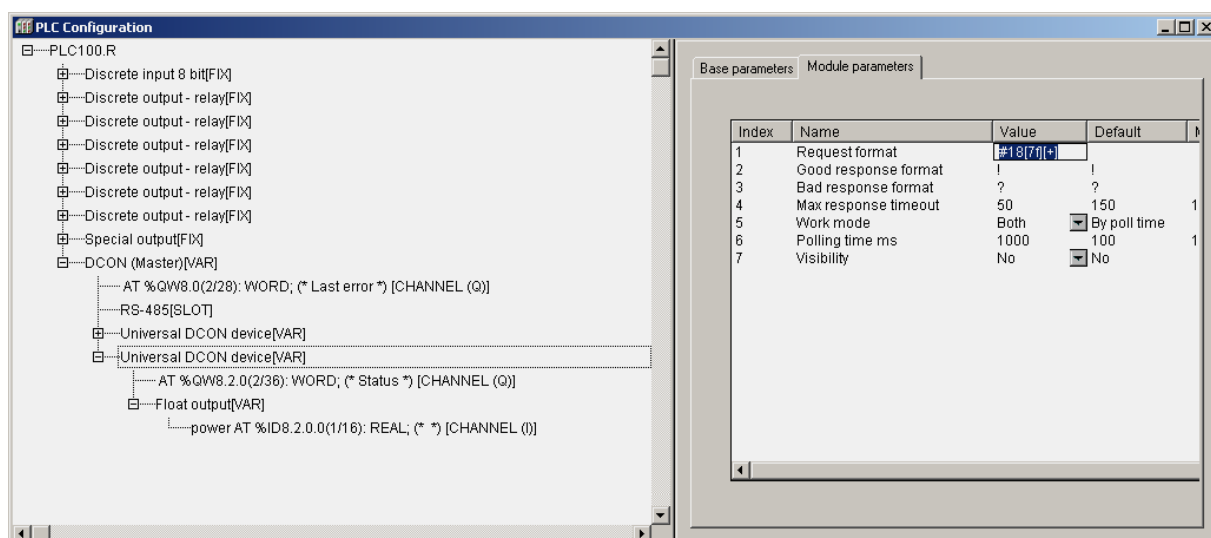
**Формат запроса: #AA(данные)[CRC][CR]**

- где # – разделитель;  
**AA** – адрес прибора;  
**+025.12** – выходное значение (5 чисел + знак + точка)  
 итого семь символов  
**CRC** – контрольная сумма;  
**CR** – перевод строки.

**Формат ответа: ![CRC][CR]**

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

Окно конфигурирования модуля **DCON (Master)** с подключенным к нему модулем **Universal DCON device**, настроенным для периодической записи выходных значений в модуль IPC-7021, проиллюстрировано на рис. Г.2.



**Рис. Г.2. Окно конфигурирования модуля Universal DCON device модуля DCON (Master) для периодической записи выходных значений в модуль IPC-7021**

Данные, посылаемые в модуль IPC 7021, задаются во входной переменной «**power**» типа **Float (REAL)** модуля **Universal DCON device**.

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]** – спецкоманда, указывающая на то, что семь символов запроса должны быть сформированы в виде числа в формате [знак]число..число.число..число. Данные

- должны быть взяты из входной переменной, которая должна иметь формат **float**;
- [+] – спецкоманда 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 изменении значения входной переменной модуля.

**Polling time** – время опроса – 1000 мс.  
Задаёт период записи значения в модуль **IPC-7021**.