

Документ подписан простой электронной подписью

Информация о подписи

ФИО: Минцаев Магомед Шавалович

Должность: Ректор

Дата подписания: 23.11.2023 00:07:47

Уникальный программный ключ:

236bcc35c296f119d6aafdc22836b21db52dbc07971a86865a5825f9fa4304cc

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

ГРОЗНЕНСКИЙ ГОСУДАРСТВЕННЫЙ НЕФТЯНОЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ

имени академика М. Д. Миллионщикова

«УТВЕРЖДАЮ»

Первый проректор
И.Г. Гайрабеков



29 2020г.

РАБОЧАЯ ПРОГРАММА

дисциплины

«Промышленные интерфейсы»

Направление подготовки

15.03.04 Автоматизация технологических процессов и производств

Профиль

«Автоматизация технологических процессов и производств»

Квалификация

Бакалавр

Грозный 2020

1. Цели и задачи дисциплины

Целью преподавания дисциплины "Промышленные интерфейсы" является:

- является овладение студентами основ построения и анализа промышленных сетей и применения промышленных интерфейсов

Основными задачами дисциплины являются:

- изучение моделей, типов, стандартов, интерфейсов и протоколов промышленных сетей;
- формирование умения применения промышленных сетевых технологий для достижения требуемого сетевого сервиса (безотказность, устойчивость к вредным факторам среды, производительность и т.д.);
- формирование навыков обоснованного выбора промышленного телекоммуникационного оборудования

2. Место дисциплины в структуре образовательной программы

Дисциплина относится к вариативной части профессионального цикла программы бакалавриата с присвоением квалификации «Бакалавр» по направлению подготовки 15.03.04 "Автоматизация технологических процессов и производств"

Для изучения дисциплины "Промышленные интерфейсы" требуются знания по следующим дисциплинам: «Теория автоматического управления», «Управляющие микропроцессорные комплексы», «Электротехника», «Вычислительные системы, сети и телекоммуникации», «Физика», «Математика».

В свою очередь, данная дисциплина "Промышленные интерфейсы" является предшествующей для дисциплин: «Системы телемеханики и аппаратура передачи данных», «Системы числового и программного управления», «Проектирование автоматизированных систем».

3. Требования к результатам освоения дисциплины

Процесс изучения дисциплины направлен на формирование следующих компетенций:

- способностью к самоорганизации и самообразованию (ОК-5);
- способностью использовать современные информационные технологии, технику, прикладные программные средства при решении задач профессиональной деятельности (ОПК-3);
- способностью собирать и анализировать исходные информационные данные для проектирования технологических процессов изготовления продукции, средств и систем автоматизации, контроля, технологического оснащения, диагностики, испытаний, управления процессами, жизненным циклом продукции и ее качеством; участвовать в работах по расчету и проектированию процессов изготовления продукции и указанных средств и систем с использованием современных информационных технологий, методов и средств проектирования (ПК-1);
- способностью проводить диагностику состояния и динамики производственных объектов производств с использованием необходимых методов и средств анализ(ПК-6);
- способностью выбирать методы и средства измерения эксплуатационных характеристик оборудования, средств и систем автоматизации, контроля, диагностики, испытаний и управления,

настройки и обслуживания: системного, инструментального и прикладного программного обеспечения данных средств и систем (ПК-24).

В результате освоения дисциплины студент должен

знать:

- Особенности архитектуры распределенных систем сбора данных и управления.
- Модель открытой промышленной сети.
- Используемые топологии сетей.
- Физические среды передачи данных в промышленных сетях.
- Основные компоненты промышленных сетей.
- Протоколы обмена информацией.
- Распространенные стандартные промышленные сети.
- Основные характеристики промышленных сетей.

уметь:

- Оценивать требования к сетям передачи информации.
- Выбирать тип промышленной сети, физический канал и протокол.
- Проектировать простейшие средства сопряжения с сетью.

владеть:

- навыками настройки основных коммуникационных устройств и промышленных интерфейсов.
- навыками проектирования и наладки сетевых структур и интерфейсов.

4. Объем дисциплины и виды учебной работы

Таблица 1

Вид учебной работы	Всего часов/зач.ед.		Семестры	
	ОФО	ЗФО	ОФО	ЗФО
	6	6	6	6
Контактная работа (всего)	64/1,78	20/0,56	64/1,78	20/0,56
В том числе:				
Лекции	32/0,89	10/0,28	32/0,89	10/0,28
Практические занятия (ПЗ)	32/0,89	10/0,28	32/0,89	10/0,28
Самостоятельная работа (всего)	80/2,22	124/3,44	80/2,22	124/3,44
В том числе:				
Реферат (презентация)	20/0,56	54/1,5	20/0,56	54/1,5
Подготовка к практическим занятиям	30/0,83	40/1,11	30/0,83	40/1,11
Подготовка к зачету	30/0,83	30/0,83	30/0,83	30/0,83
Вид промежуточной аттестации	зачет	зачет	зачет	зачет
Общая трудоемкость дисциплины Час./ Зач. ед.	144/4	144/4	144/4	144/4

5. Содержание дисциплины

5.1. Разделы дисциплины и виды занятий

Таблица 2

№ п/п	Наименование раздела дисциплины	Лекц. зан. часы		Практ. зан. часы		Всего часов	
		ОФО	ЗФО	ОФО	ЗФО	ОФО	ЗФО
Модуль 1							
1.	Основы построения промышленных сетей и интерфейсов	10	2	10	2	20	4
Модуль 2							
2.	Основные промышленные протоколы	12	4	12	4	24	8
Модуль 3							
3.	Аппаратное обеспечение и защита от помех.	10	4	10	4	20	8
Всего часов		32	10	32	10	64	20

5.2. Лекционные занятия

Таблица 3

№ п/п	Наименование раздела дисциплины	Содержание раздела
Модуль 1		
1.	Основы построения промышленных сетей и интерфейсов	Общие сведения о промышленных сетях. Модель OSI. Интерфейсы RS-485, RS-422 и RS-232. Интерфейс "токовая петля".
Модуль 2		
2.	Основные промышленные протоколы	HART-протокол. CAN. Profibus. Modbus. Промышленный Ethernet. Беспроводные локальные сети
Модуль 3		
3.	Аппаратное обеспечение и защита от помех.	Сетевое оборудование. Кабели для промышленных сетей. Защита от помех. Заземление. Проводные каналы передачи сигналов. Паразитные связи. Методы экранирования и заземления. Защита промышленных сетей от молнии.

5.3. Практические занятия

Таблица 4

№ п/п	Наименование раздела дисциплины	Наименование практической работы
1	Основы построения промышленных сетей и интерфейсов	Настройка панели СП307 для сети Modbus RTU. Настройка связи СП307 с ПЛК110 по протоколу Modbus RTU (Master/Slave). Настройка связи ПЛК110 с MasterSCADA 4D по протоколу Modbus RTU (Master/Slave).

2	Основные промышленные протоколы	Настройка связи СП307 с ПР114 по протоколу Modbus RTU (Master/Slave). Настройка связи СП307 с ПЧВ1 по протоколу Modbus RTU (Master/Slave).
3	Аппаратное обеспечение и защита от помех.	Обмен по протоколу Modbus TCP через Ethernet. Подключение OPC CoDeSyS V2.3. Реализация WEB-сервера.

6. Самостоятельная работа студентов по дисциплине

Самостоятельная работа включает: повторение студентом изложенного на лекциях и практических занятиях учебного материала, решение индивидуальных домашних задач, подготовку к контрольному опросу и экзамену.

Самостоятельная работа, направленная на углубление и закрепление знаний, а также развитие практических умений заключается в:

- анализе теоретических и фактических материалов по заданной теме, проведении расчетов, составлении схем и моделей на основе сценариев работы технологического оборудования и производства,
- изучении тем, вынесенных на самостоятельную проработку,
- изучении теоретического материала к практическим занятиям,
- выполнении заданий на практические занятия,
- подготовке к экзамену.

6.1. Подготовка к практическим занятиям

№ п/п	Наименование раздела дисциплины	Наименование практической работы
1	Основы построения промышленных сетей и интерфейсов	Настройка панели СП307 для сети Modbus RTU. Настройка связи СП307 с ПЛК110 по протоколу Modbus RTU (Master/Slave). Настройка связи ПЛК110 с MasterSCADA 4D по протоколу Modbus RTU (Master/Slave).
2	Основные промышленные протоколы	Настройка связи СП307 с ПРxx по протоколу Modbus RTU (Master/Slave). Настройка связи СП307 с ПЧВ1 по протоколу Modbus RTU (Master/Slave).
3	Аппаратное обеспечение и защита от помех.	Обмен по протоколу Modbus TCP через Ethernet. Подключение OPC CoDeSyS V2.3. Реализация WEB-сервера.

6.2. Темы рефератов

- Промышленная шина AS-interface
- Промышленная шина CANopen
- Промышленные беспроводные сети
- Промышленная шина Profibus
- Промышленная шина MODBUS
- Промышленная шина HART
- Промышленная шина Profinet
- Модель OSI

Учебно- методическое обеспечение самостоятельной работы студентов

1. Борисов А.М. Основы построения промышленных сетей автоматизации /А.М. Борисов. – Челябинск: Издательский центр ЮУрГУ, 2012. – 108 с.
2. Кангин В.В. Аппаратные и программные средства систем управления. Промышленные сети и контроллеры: учебное пособие / В. В. Кангин, В. Н. Козлов. — М.: Бинوم. Лаборатория знаний, 2010.
3. Парк Дж., Маккей С., Райт Э. Передача данных в системах контроля и управления: практическое руководство / перевод с англ. Савельева В. В.. — М.: ООО "Группа ИТД", 2007. — 480 с.

7. Оценочные средства.

Текущий контроль

Вопросы к первой аттестации

1. Общие сведения о промышленных сетях.
2. Модель OSI. Физический уровень.
3. Модель OSI. Канальный уровень.
4. Модель OSI. Сетевой уровень.
5. Модель OSI. Транспортный уровень.
6. Модель OSI. Сеансовый уровень.
7. Модель OSI. Уровень представления.
8. Модель OSI. Прикладной уровень.
9. Интерфейсы RS-485, RS-422 и RS-232. Дифференциальная передача сигнала.
10. Интерфейсы RS-485, RS-422 и RS-232. "Третье" состояние выходов.
11. Интерфейсы RS-485, RS-422 и RS-232. Четырехпроводной интерфейс.
12. Интерфейсы RS-485, RS-422 и RS-232. Режим приема эха.
13. Интерфейсы RS-485, RS-422 и RS-232. Заземление, гальваническая изоляция и защита от молнии.
14. Интерфейсы RS-485, RS-422 и RS-232. Стандартные параметры.
15. Согласование линии с передатчиком и приемником.
16. Топология сети на основе интерфейса RS-485.
17. Устранение состояния неопределенности линии.
18. Сквозные токи.
19. Выбор кабеля.
20. Расширение предельных возможностей.
21. Интерфейсы RS-232 и RS-422.
22. Интерфейс "токовая петля".
23. HART-протокол.
24. HART-протокол. Принципы построения.
25. Сеть на основе HART-протокола
26. HART-протокол. Адресация.
27. Команды HART
28. HART-протокол. Язык описания устройств DDL. Разновидности.
29. CAN. Физический уровень.
30. Электрические соединения в сети CAN.
31. Трансивер CAN
32. CAN. Канальный уровень.
33. CAN. Достоверность передачи
34. CAN. Передача сообщений.
35. Прикладной уровень: CANopen.
36. Profibus. Физический уровень.
37. Канальный уровень Profibus DP
38. Profibus. Коммуникационный профиль DP.
39. Profibus. Передача сообщений.

Вопросы к зачету по дисциплине

1. Общие сведения о промышленных сетях.
2. Модель OSI. Физический уровень.
3. Модель OSI. Канальный уровень.
4. Модель OSI. Сетевой уровень.
5. Модель OSI. Транспортный уровень.
6. Модель OSI. Сеансовый уровень.
7. Модель OSI. Уровень представления.
8. Модель OSI. Прикладной уровень.
9. Интерфейсы RS-485, RS-422 и RS-232. Дифференциальная передача сигнала.
10. Интерфейсы RS-485, RS-422 и RS-232. "Третье" состояние выходов.
11. Интерфейсы RS-485, RS-422 и RS-232. Четырехпроводной интерфейс.
12. Интерфейсы RS-485, RS-422 и RS-232. Режим приема эха.
13. Интерфейсы RS-485, RS-422 и RS-232. Заземление, гальваническая изоляция и защита от молнии.
14. Интерфейсы RS-485, RS-422 и RS-232. Стандартные параметры.
15. Согласование линии с передатчиком и приемником.
16. Топология сети на основе интерфейса RS-485.
17. Устранение состояния неопределенности линии.
18. Сквозные токи.
19. Выбор кабеля.
20. Расширение предельных возможностей.
21. Интерфейсы RS-232 и RS-422.
22. Интерфейс "токовая петля".
23. HART-протокол.
24. HART-протокол. Принципы построения.
25. Сеть на основе HART-протокола
26. HART-протокол. Адресация.
27. Команды HART
28. HART-протокол. Язык описания устройств DDL. Разновидности.
29. CAN. Физический уровень.
30. Электрические соединения в сети CAN.
31. Трансивер CAN
32. CAN. Канальный уровень.
33. CAN. Достоверность передачи
34. CAN. Передача сообщений.
35. Прикладной уровень: CANopen.
36. Profibus. Физический уровень.
37. Канальный уровень Profibus DP
38. Profibus. Коммуникационный профиль DP.
39. Profibus. Передача сообщений.
40. Profibus. Резервирование.
41. Profibus. Описание устройств.
42. Modbus. Физический уровень.
43. Modbus. Канальный уровень.
44. Modbus. Прикладной уровень
45. Промышленный ETHERNET.
46. Промышленный ETHERNET. Отличительные особенности.
47. Промышленный ETHERNET. Физический уровень.
48. Промышленный ETHERNET. Канальный уровень
49. Промышленный ETHERNET. Modbus TCP.
50. Промышленный ETHERNET. Profinet.
51. Беспроводные локальные сети.
52. Беспроводные локальные сети. Проблемы беспроводных сетей.
53. Беспроводные локальные сети. Bluetooth.

54. Беспроводные локальные сети. ZigBee и IEEE 802.15.4.
55. Беспроводные локальные сети. Wi-Fi и IEEE 802.11.
56. Сравнение беспроводных сетей.
57. Сетевое оборудование. Повторители интерфейса.
58. Сетевое оборудование. Концентраторы (хабы).
59. Сетевое оборудование. Преобразователи интерфейса.
60. Сетевое оборудование. Адресуемые преобразователи интерфейса.
61. Сетевое оборудование. Межсетевые шлюзы.
62. Сетевое оборудование. Другое сетевое оборудование
63. Сетевое оборудование. Кабели для промышленных сетей.
64. Защита от помех.
65. Источники помех.
66. Характеристики помех.
67. Помехи из сети электроснабжения.
68. Молния и атмосферное электричество.
69. Статическое электричество.
70. Помехи через кондуктивные связи.
71. Электромагнитные помехи.
72. Другие типы помех.
73. Заземление. Определения.
74. Цели заземления.
75. Защитное заземление зданий.
76. Автономное заземление.
77. Заземляющие проводники.
78. Модель «земли».
79. Виды заземлений.
80. Проводные каналы передачи сигналов.
81. Источники сигнала.
82. Приемники сигнала.
83. Прием сигнала заземленного источника.
84. Прием сигнала незаземленных источников.
85. Дифференциальные каналы передачи сигнала.
86. Паразитные связи.
87. Модели компонентов систем автоматизации.
88. Паразитные кондуктивные связи.
89. Индуктивные и емкостные связи.
90. Методы экранирования и заземления.
91. Гальванически связанные цепи.
92. Экранирование сигнальных кабелей.
93. Гальванически развязанные цепи.
94. Экраны кабелей на электрических подстанциях.
95. Экраны кабелей для защиты от молнии.
96. Заземление при дифференциальных измерениях.
97. Интеллектуальные датчики.
98. Монтажные шкафы.
99. Распределенные системы управления.
100. Чувствительные измерительные цепи.
101. Исполнительное оборудование и приводы.
102. Заземление на взрывоопасных объектах.
103. Гальваническая развязка.
104. Защита промышленных сетей от молнии.
105. Пути прохождения импульса молнии.
106. Средства защиты от молнии.
107. Стандарты и методы испытаний по ЭМС.
108. Верификация заземления и экранирования.

ПЛК110-32.P-M [M02]				СП307	
Переменные для чтения из панели		Переменные для записи в панель		Тип данных	Регистр
Программа	Modbus Master (конфигурация)	Программа	Modbus Master (конфигурация)		
bBoolRead	bInput	bBoolWrite	bOutput	BOOL	PSB300
wWordRead	wInput	wWordWrite	wOutput	WORD	PSW300
rRealRead	rInput	rRealWrite	rOutput	REAL	PSW310-311
sStringRead	sInput	sStringWrite	sOutput	STRING	PSW320-322

Настройки СП307 (Modbus RTU Slave)

2.1. Настройки PLC-порта

Для того чтобы использовать панель СП307 в режиме **Modbus RTU Slave**, необходимо в настройках проекта на вкладке **Устройство** настроить режимы работы последовательных портов. В нашем примере мы будем использовать **PLC-порт**. Настроим его в качестве **Modbus Slave** согласно [табл. 1](#): скорость – **115200**, бит данных – **8**, стоп бит – **1**, контроль четности – **нет**. Адрес порта (**Slave ID**) будет равен **1**.

Обратите внимание на наличие галочки **Передача данных** – при ее отсутствии ПЛК не сможет считывать данные с панели (но сможет их записывать).

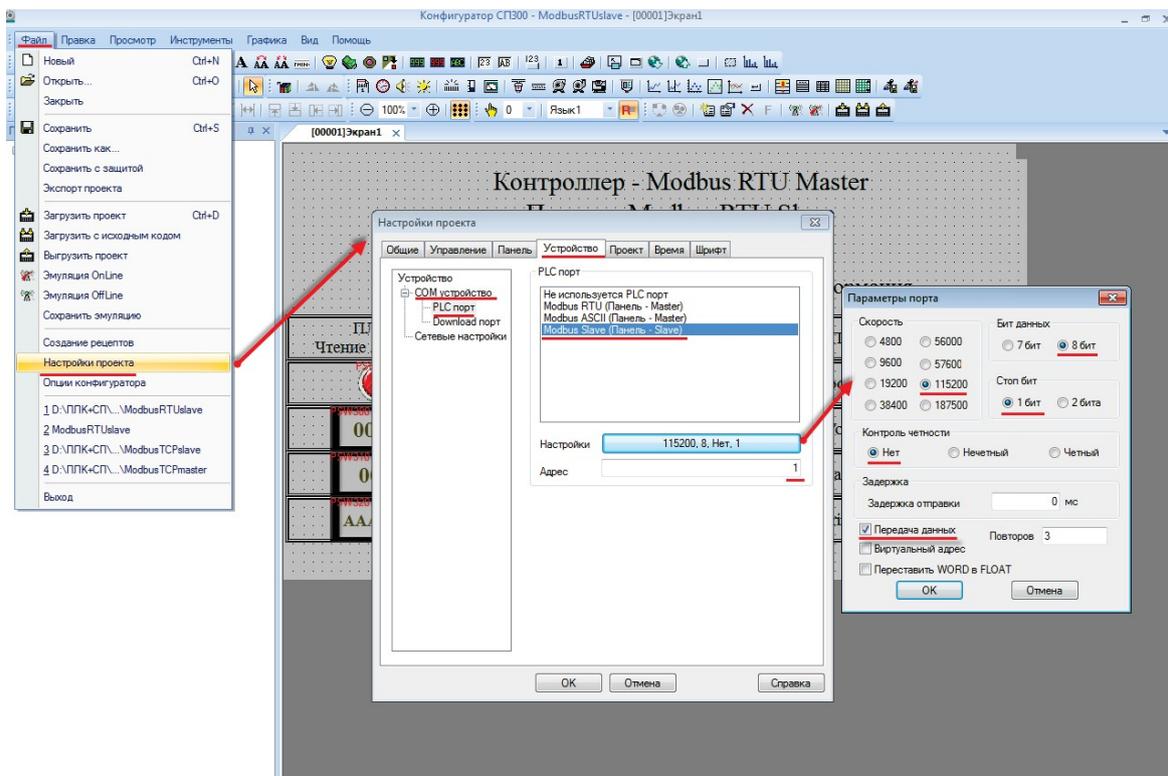


Рис. 1. Настройки PLC-порта панели (режим Modbus RTU Slave)

2.2. Экран проекта

Проект содержит один экран, на котором осуществляется отображение данных, которые записывает ПЛК, и ввод данных, которые ПЛК считывает; при этом и отображение, и ввод для каждого типа данных реализованы через один элемент.

Соответственно, экран содержит четыре активных элемента:

1. **Переключатель с индикацией** с привязанным регистром **PSB300**. ПЛК считывает значение этого регистра в переменную **bBoolRead** и по команде пользователя записывает в него значение переменной **bBoolWrite**. Обе переменные имеют тип **BOOL**.
2. **Цифровой ввод** с привязанным регистром **PSW300**. ПЛК считывает значение этого регистра в переменную **wWordRead** и по команде пользователя записывает в него значение переменной **wWordWrite**. Обе переменные имеют тип **WORD**.
3. **Цифровой ввод** с привязанными регистрами **PSW310-311**. ПЛК считывает значения этих регистров в переменную **rRealRead** и по команде пользователя записывает в них значение переменной **rRealWrite**. Обе переменные имеют тип **REAL**.
4. **Текстовый ввод** с привязанными регистрами **PSW320-322**. ПЛК считывает значения этих регистров в переменную **sStringRead** и по команде пользователя записывает в них значение переменной **sStringWrite**. Обе переменные имеют тип **STRING**.

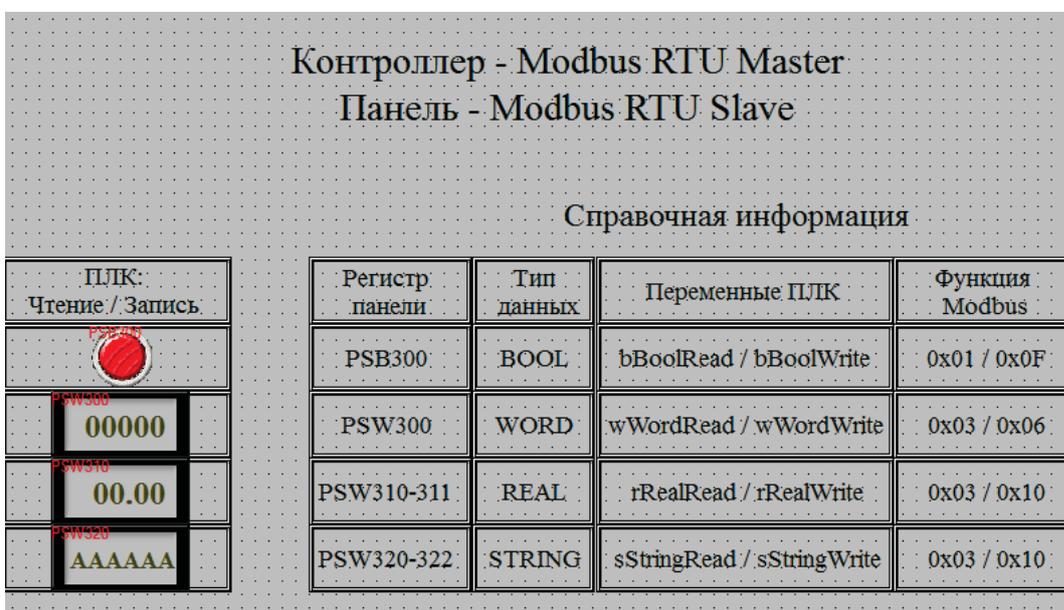


Рис. 2. Внешний вид экрана проекта

Настройки элементов приведены ниже.

2.2.1. Переключатель с индикацией (BOOL)

Элемент **Переключатель с индикацией** используется для отображения и ввода значений переменных ПЛК типа **BOOL**. Переменные этого типа в каждый момент времени находятся в одном из двух возможных состояний: **TRUE** или **FALSE** (иными словами, **Включено/Выключено**).



Рис. 3. Внешний вид элемента **Переключатель с индикацией**

В настройках элемента на вкладке **Регистр элемента** выберем регистр **PSB300** (согласно [табл. 2](#)).

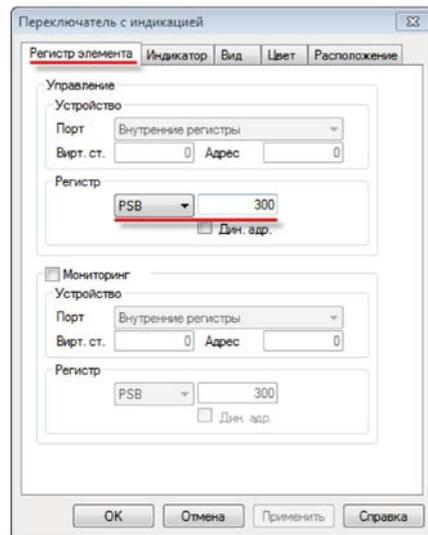


Рис. 4. Настройки элемента **Переключатель с индикацией**, вкладка **Регистр элемента**

На вкладке **Индикатор** для настройки **Переключение бита в состояние** выберем значение **Инверсия**. Это нужно для того, чтобы свободно менять значение элемента с дисплея панели.

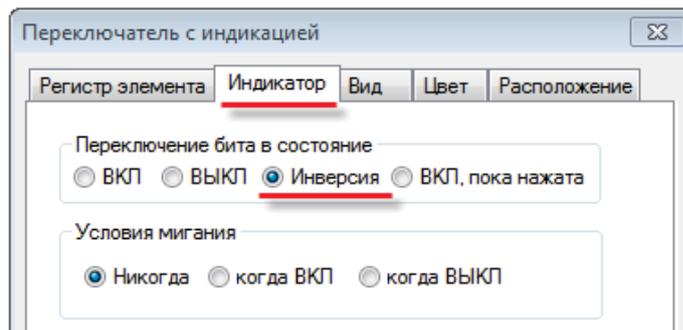


Рис. 5. Настройки элемента **Переключатель с индикацией**, вкладка **Индикатор**

2.2.2. Цифровой ввод (WORD)

Элемент **Цифровой ввод (WORD)** используется для отображения и ввода значений переменных ПЛК типа **WORD**. Переменные этого типа являются *целочисленными* и могут принимать значение от 0 до 65535.



Рис. 6. Внешний вид элемента **Цифровой ввод**

В настройках элемента на вкладке **Регистр элемента** выберем регистр **PSW300** (согласно [табл. 2](#)). **Обратите внимание**, что параметр **Тип** должен иметь значение **WORD**.

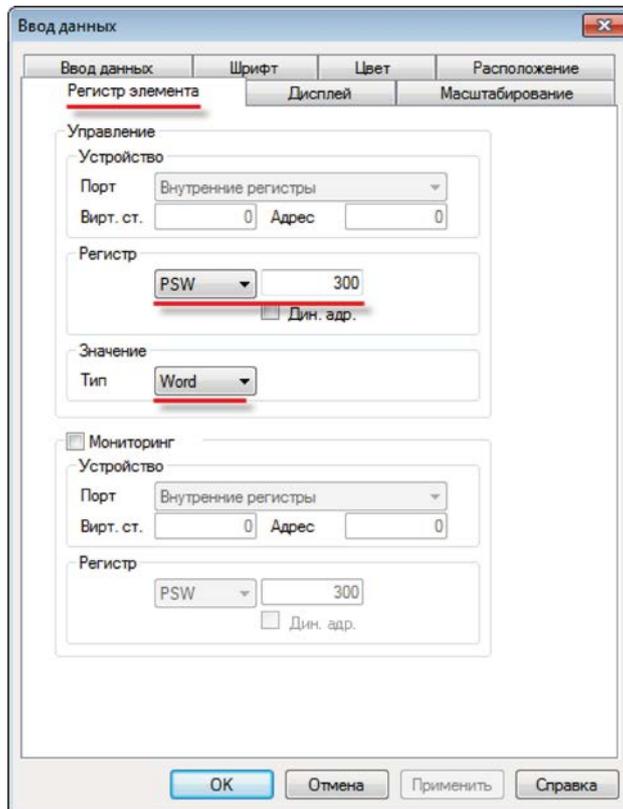


Рис. 7. Настройки элемента **Цифровой ввод**, вкладка **Регистр элемента**

На вкладке **Дисплей** выберем формат **Unsigned** (беззнаковый). Для параметра **Разрядность/Всего знаков** зададим значение **5**, поскольку переменные типа **WORD** не способны принимать шестизначные значения.

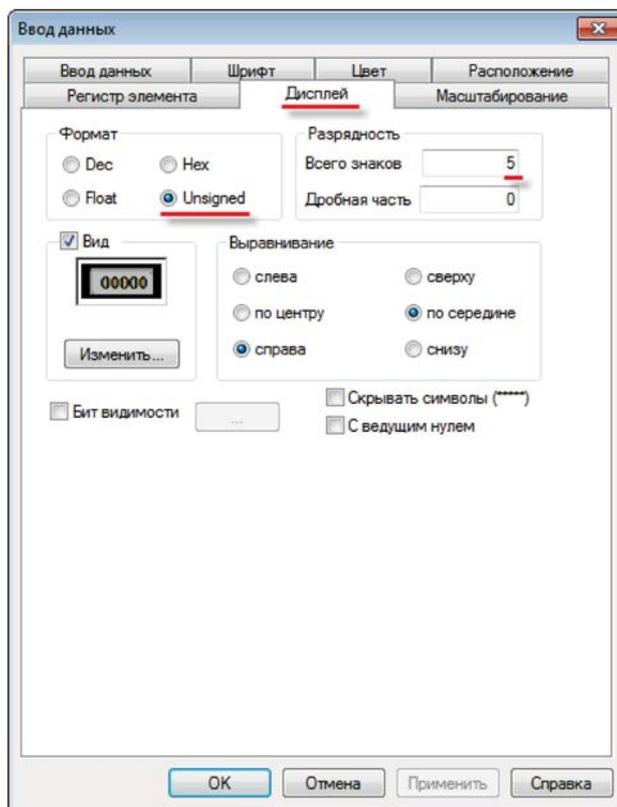


Рис. 8. Настройки элемента **Цифровой ввод**, вкладка **Дисплей**

Обратите внимание, что при необходимости работать с отрицательными целыми числами следует использовать формат **Dec**, а в ПЛК преобразовать полученное значение стандартной функцией **WORD_TO_INT** и записывать в переменную типа **INT**.

2.2.3. Цифровой ввод (real)

Элемент **Цифровой ввод (REAL)** используется для отображения и ввода значений переменных ПЛК типа **REAL**. Переменные этого типа используются для работы с [числами с плавающей точкой](#).



Рис. 9. Внешний вид элемента **Цифровой ввод**

В настройках элемента на вкладке **Регистр элемента** выберем регистр **PSW310** и тип **DWORD** (поскольку переменные типа **REAL** занимают два регистра). В результате, данные элемента будут храниться в регистрах **PSW310-311** (согласно [табл. 2](#)).

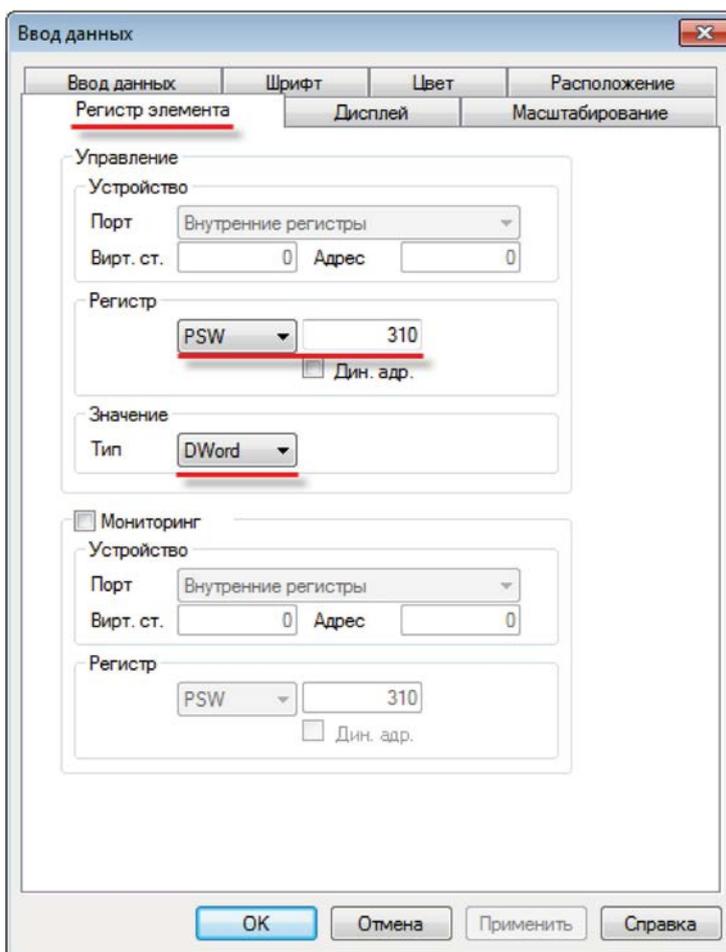


Рис. 10. Настройки элемента **Цифровой ввод**, вкладка **Регистр элемента**

На вкладке **Дисплей** выберем формат **Float** (с плавающей точкой). Для параметров **Разрядность/Всего знаков** и **Разрядность/Дробная часть** зададим значения **4** и **2**, т.е. элемент сможет отображать значения от -99.99 до 99.99.

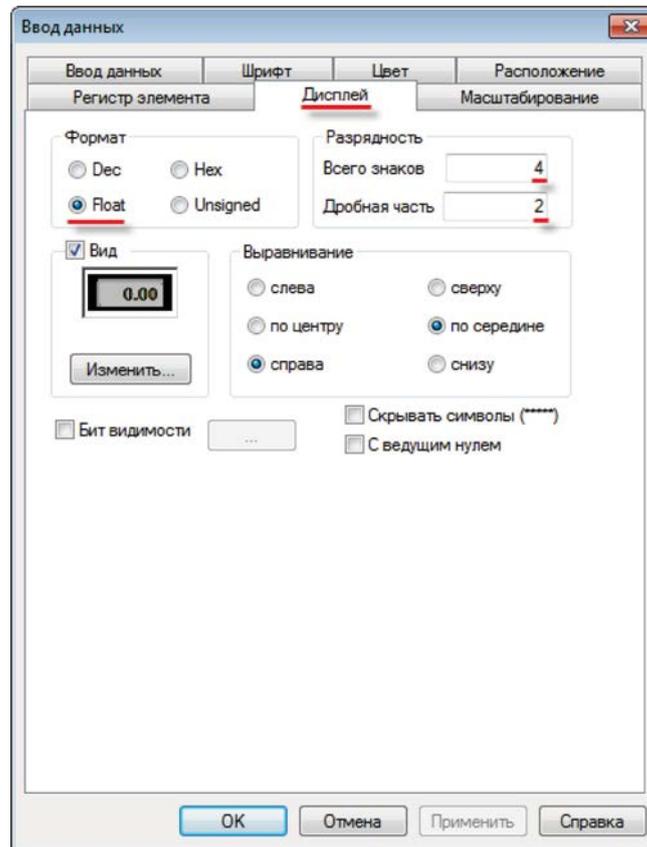


Рис. 11. Настройки элемента **Цифровой ввод**, вкладка **Дисплей**

2.2.4. Текстовый ввод (STRING)

Элемент **Текстовый ввод** используется для отображения и ввода значений переменных ПЛК типа **STRING**. Переменные этого типа представляют собой текстовые строки.



Рис. 12. Внешний вид элемента **Текстовый ввод**

В настройках элемента на вкладке **Регистр элемента** выберем регистр **PSW320** и кол-во регистров **3**. В результате, данные элемента будут храниться в регистрах **PSW320-322** (согласно [табл. 2](#)). Каждый регистр может содержать два символа, поэтому с помощью элемента будет осуществляться отображение и ввод текстовых строк длиной до **6** символов.

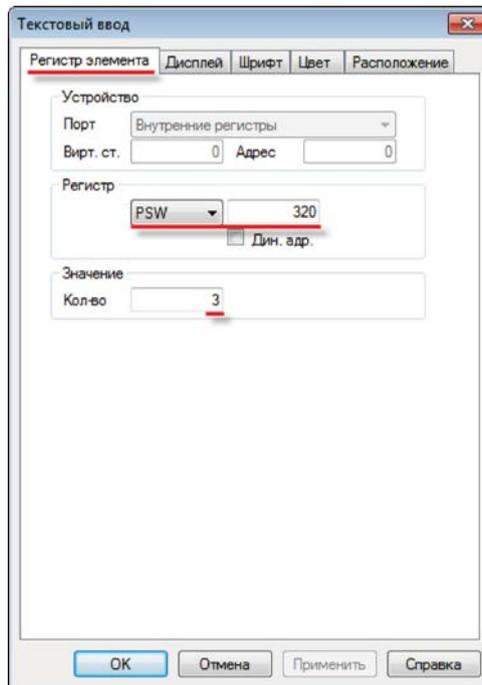


Рис. 13. Настройки элемента **Текстовый ввод**, вкладка **Регистр элемента**

Обратите внимание, что при необходимости ввода с дисплея панели символов кириллицы следует на вкладке **Дисплей** в параметре **Клавиатура** выбрать клавиатуру **60043**.

3. Настройки ПЛК110 [M02] (Modbus RTU Master)

3.1. Конфигурация ПЛК

В компоненте **Конфигурация ПЛК** (вкладка **Ресурсы**) пользователь производит настройку каналов Modbus и привязывает к ним переменные.

Нажмем **ПКМ** на название контроллера (в нашем примере - **PLC110_32**) и добавим подэлемент **Modbus (Master)**:

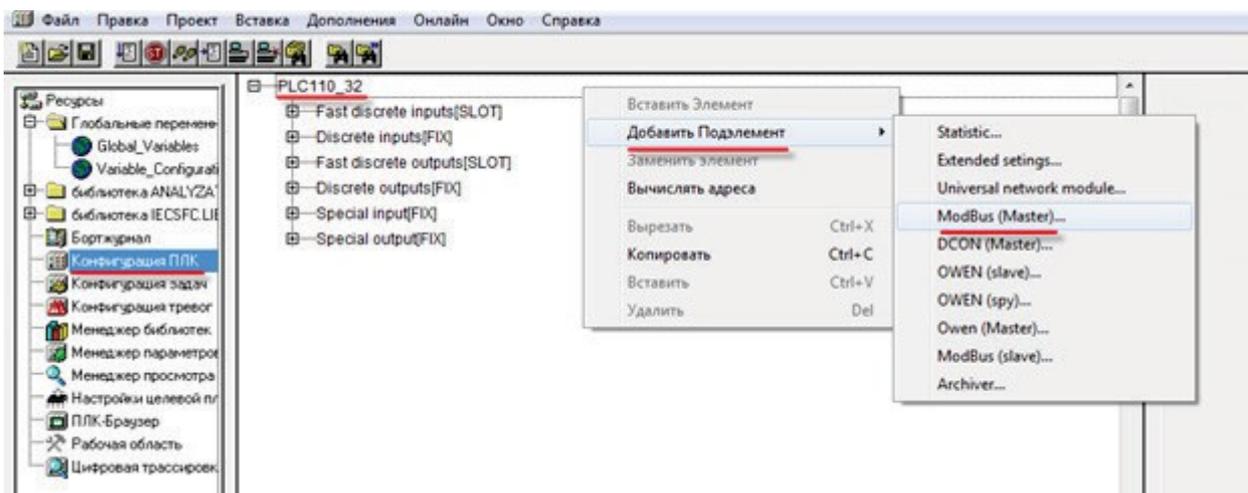


Рис. 14. Конфигурация ПЛК. Добавление **Modbus (Master)**

Этот элемент не нуждается в настройках.

Выберем порт ПЛК, который будет использоваться для связи с панелью. Для этого в элементе **Modbus (Master)** нажмем **ПКМ** на порт **Debug RS-232** и в контекстном меню

выберем команду **Заменить элемент**. В нашем примере мы используем **RS-485**; при необходимости, можно использовать **RS-232** или **Debug RS-232** – это повлияет только на порт, к которому необходимо будет подключить панель, и кабель, с помощью которого будут соединены панель и ПЛК.

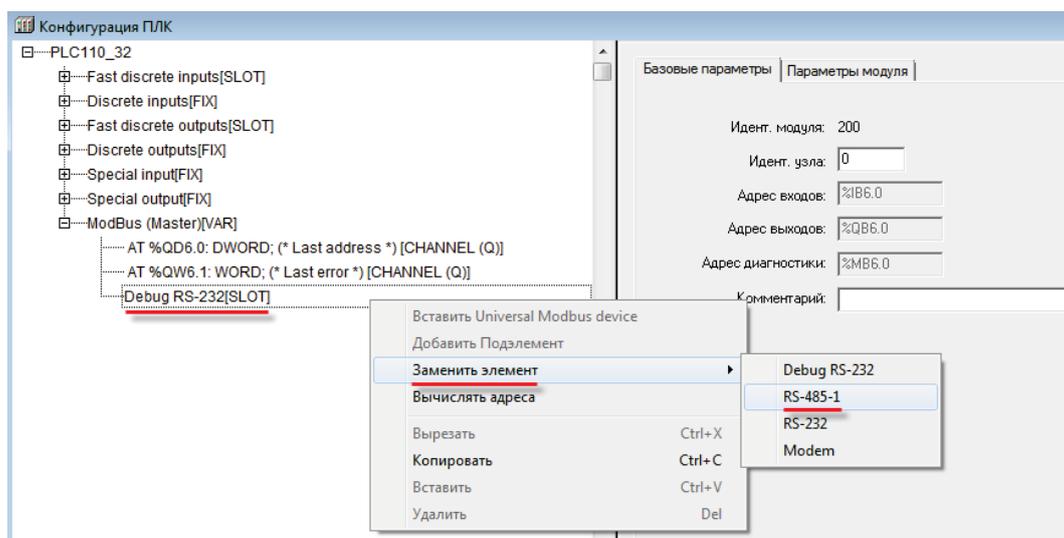


Рис. 15. Конфигурация ПЛК. Выбор порта

Настройки порта по умолчанию соответствуют тем настройкам, которые мы задали панели (согласно [табл. 1](#)): скорость – **115200**, бит данных – **8**, стоп бит – **1**, контроль четности – **нет**. Для параметра **Frame Oriented** выберем значение **RTU**.

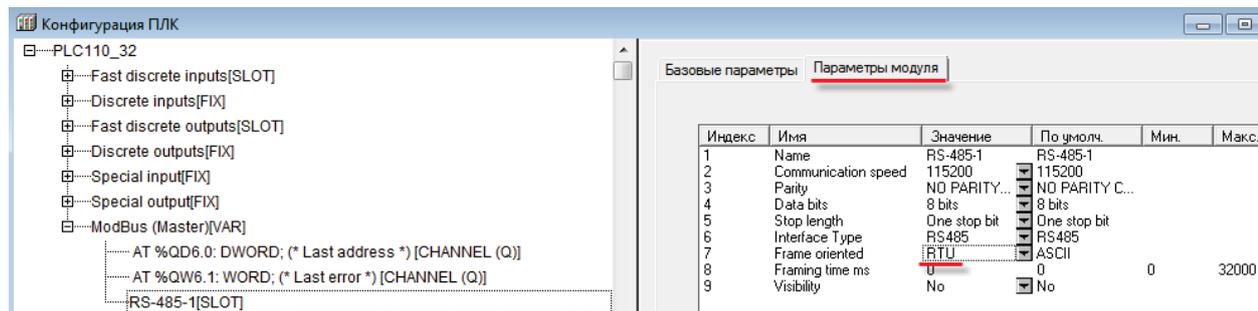


Рис. 16. Конфигурация ПЛК. Настройки порта

Нажмем **ПКМ** на элемент **Modbus (Master)** и добавим два подэлемента **Universal Modbus Device**:

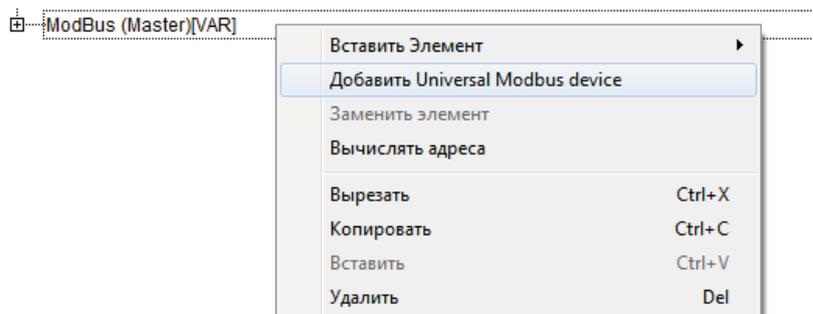


Рис. 17. Конфигурация ПЛК. Добавление **Universal Modbus Device**

Один из них будет использоваться для чтения значений из панели, второй – для записи.

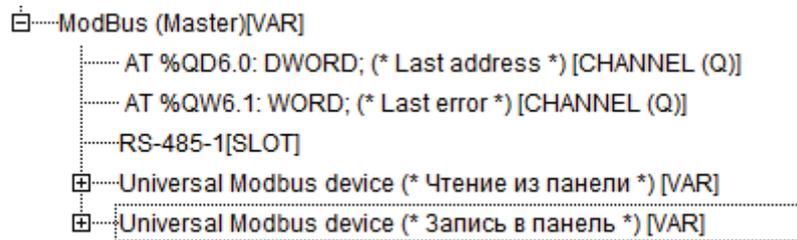


Рис. 18. Внешний вид **Конфигурации ПЛК** после добавления двух **Universal Modbus Device**

В настройках элемента **Чтение из панели** укажем тип связи (**Serial**), **Slave ID** панели (**1**) и режим опроса (**By poll time**, т.е. циклически).

Индекс	Имя	Значение	По умолч.	Мин
1	Name	Universal Modbus d...	Universal Modbus d...	
2	ModuleIP	10:0:6:20	10:0:0:223	
3	Max timeout	150	150	10
4	TCPport	502	502	
5	NetMode	Serial	Serial	
6	ModuleSlave...	1	1	0
7	Work mode	By poll time	By poll time	
8	Polling time ms	100	100	10
9	Visibility	No	No	
10	Amount Rep...	0	0	0
11	Byte Sequen...	Trace_mode	Trace_mode	

Рис. 19. Настройки **Universal Modbus Device (Чтение из панели)**

В настройках элемента **Запись в панель** укажем тип связи (**Serial**), **Slave ID** панели (**1**) и режим опроса и режим опроса (**By value change**, т.е. спорадически).

Индекс	Имя	Значение	По умолч.	Мин
1	Name	Universal Modbus d...	Universal Modbus d...	
2	ModuleIP	10:0:6:20	10:0:0:223	
3	Max timeout	150	150	10
4	TCPport	502	502	
5	NetMode	Serial	Serial	
6	ModuleSlave...	1	1	0
7	Work mode	By value change	By poll time	
8	Polling time ms	100	100	10
9	Visibility	No	No	
10	Amount Rep...	3	0	0
11	Byte Sequen...	Trace_mode	Trace_mode	

Рис. 20. Настройки **Universal Modbus Device (Запись в панель)**

Нажмем ПКМ на элемент **Universal Modbus Device (Чтение из панели)** и добавим в него подэлементы **8 bit Input Module**, **Register Input Module**, **Real Input Module** и **String Input Module**.

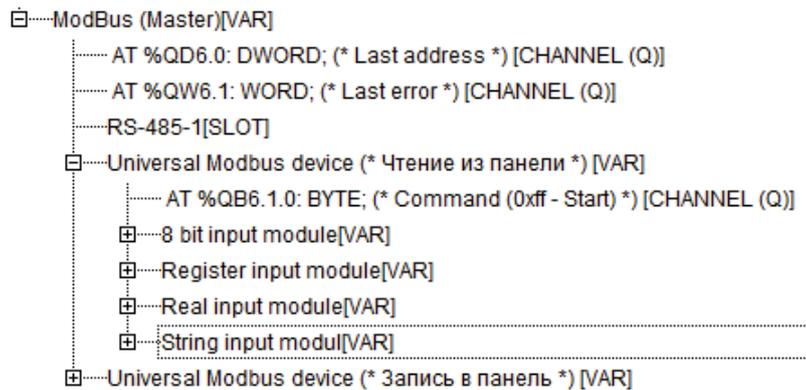


Рис. 21. Universal Modbus Device (Чтение из панели) с добавленными Input модулями

Привяжем к каждому из каналов переменную (после ввода ее имени она автоматически будет добавлена в список глобальных переменных проекта). Для ввода имени переменной два раза нажмите на **АТ**.

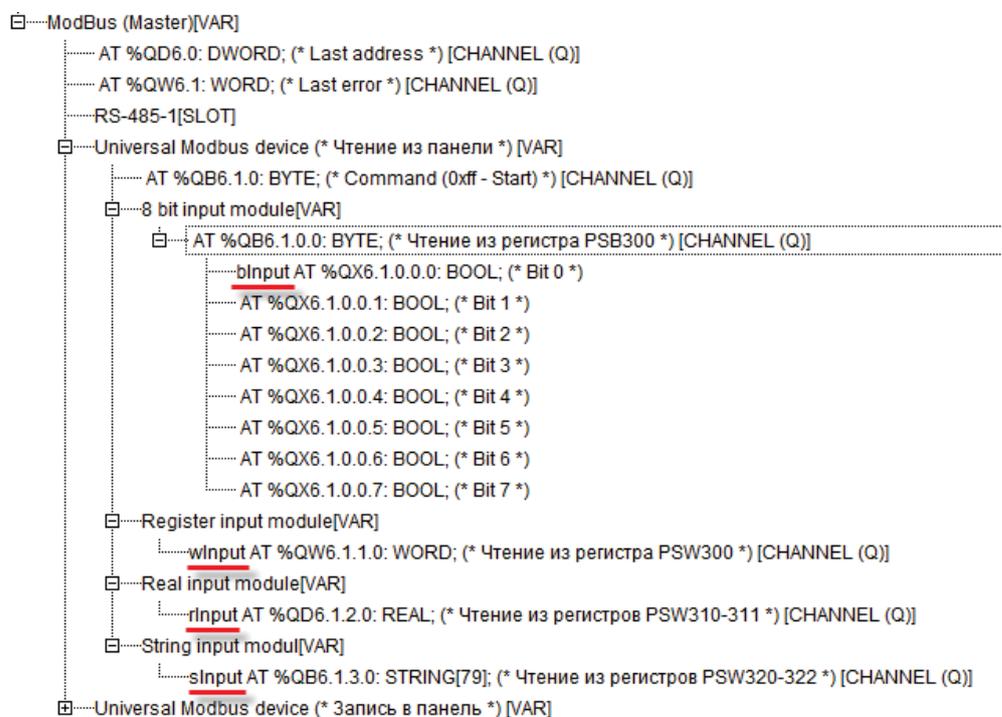


Рис. 22. Привязка переменных к каналам

Настройки модулей (используемые регистры панели и функции Modbus) приведены ниже.

Обратите внимание, что работа с переменными типа **BOOL** осуществляется через **8 Bit Input Module**. При этом нулевому биту соответствует заданный регистр модуля (см. рис. 22), первому – заданный регистр+1 (PSB301), второму – PSB302 и т.д.

Обратите внимание, что при работе с переменными, занимающими несколько регистров панели (тип **REAL** и **STRING**), указывается только первый из группы регистров.

Инде...	Имя	Значение	По умолч.
1	Name	8 bit input module	8 bit input module
2	Register a...	300	0
3	Command	Read coils status (...)	Read inputs status (0x02)
8	Visibility	No	No

Рис. 23. Параметры **8 Bit Input Module**

Инде...	Имя	Значение	По умолч.
1	Name	Register input module	Register input module
2	Regist...	300	0
3	Comma...	Read holding Register...	Read holding Registers ...
8	Visibility	No	No

Рис. 24. Параметры **Register Input Module**

Инде...	Имя	Значение	По умолч.
1	Name	float input module	float input module
2	Regist...	310	0
3	Comma...	Read holding Register...	Read holding Registers ...
8	Visibility	No	No

Рис. 25. Параметры **Real Input Module**

Базовые параметры		Параметры модуля	
Индекс	Имя	Значение	По умолч.
1	Name	String input module	String input m...
2	Command	Read holding Registers (0...	Read bytes (0...
3	Register a...	320	0
4	Amount...	6	80
8	Visibility	No	No

Рис. 26. Параметры **String Input Module**

Нажмем ПКМ на элемент **Запись в панель** и добавим в него подэлементы **8 bit Output Module**, **Register Output Module**, **Real Output Module** и **String Output Module**.

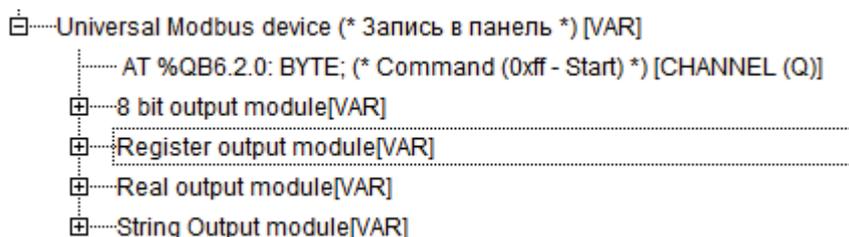


Рис. 27. **Universal Modbus Device (Запись в панель)** с добавленными **Output** модулями

Привяжем к каждому из каналов переменную (после ввода ее имени она автоматически будет создана в проекте как глобальная). Для ввода имени переменной два раза нажмите на **АТ**.

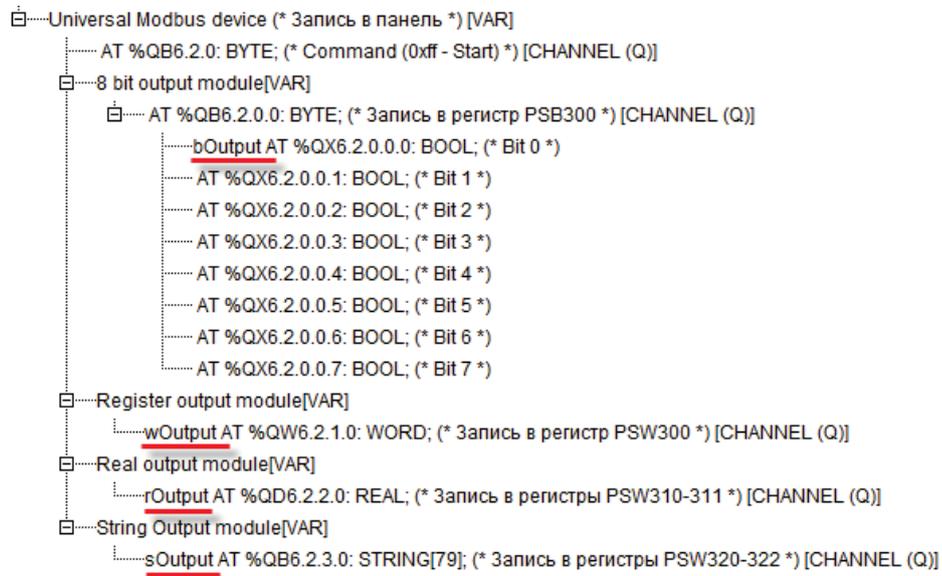


Рис. 28. Привязка переменных к каналам

Настройки модулей (используемые регистры панели и функции Modbus) приведены ниже.

Обратите внимание, что работа с переменными типа **BOOL** осуществляется через **8 Bit Output Module**. При этом нулевому биту соответствует заданный регистр модуля (см. рис. 28), первому – заданный регистр+1 (PSB301), второму – PSB302 и т.д.

Обратите внимание, что при работе с переменными, занимающими несколько регистров панели (тип **REAL** и **STRING**), указывается только первый из группы регистров.

Инде...	Имя	Значение	По умолч.
1	Name	8 bit output module	8 bit output mod
2	Register ...	300	0
3	Command	Force multiply coils (0...	Force multiply c
8	Visibility	No	No

Рис. 29. Параметры **8 Bit Output Module**

Инде...	Имя	Значение	По умолч.
1	Name	Register	Register
2	Register...	300	0
3	Command	Preset singl register (...)	Preset singl regi
8	Visibility	No	No

Рис. 30. Параметры **Register Output Module**

Инде...	Имя	Значение	По умолч.
1	Name	float output module	float output mod
2	Regist...	310	0
3	Comm...	Preset multiple Registe...	Preset multiple F
8	Visibility	No	No

Рис. 31. Параметры **Real Output Module**

Базовые параметры		Параметры модуля	
Индекс	Имя	Значение	По умолч.
1	Name	String output module	String output
2	Comma...	Preset multiple Register...	Preset singl re
3	Registe...	320	0
4	Amount...	6	80
8	Visibility	No	No

Рис. 32. Параметры **String Output Module**

3.2. Программа PLC_PRG

Данные, считанные через **Конфигурацию ПЛК**, присваиваются локальным переменным программы **PLC_PRG**. В свою очередь, значения, заданные в программе, передаются в **Конфигурацию ПЛК** и записываются в панель. Этот подход использован только на уровне примера – в реальных проектах зачастую удобнее пользоваться в программе переменными из **Конфигурации ПЛК**.

Объявим в программе **PLC_PRG** следующие переменные:

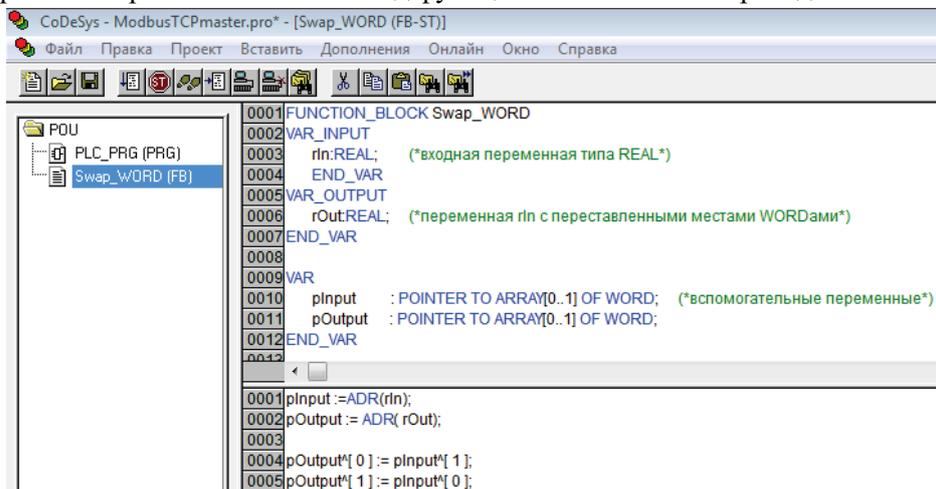
```

0001 PROGRAM PLC_PRG
0002 VAR
0003   bBoolRead   :BOOL;           (*чтение BOOL из регистра панели PSB300*)
0004   bBoolWrite  :BOOL;           (*запись BOOL в регистр панели PSB300*)
0005
0006   wWordRead   :WORD;           (*чтение WORD из регистра панели PSW300*)
0007   wWordWrite  :WORD;           (*запись WORD в регистр панели PSW300*)
0008
0009   rRealRead   :REAL;           (*чтение REAL из регистров панели PSW310-311*)
0010   rRealWrite  :REAL;           (*запись REAL в регистры панели PSW310-311*)
0011
0012   Swap_WORD_panel:Swap_WORD;   (*экземпляр ФБ для переворота WORD в REAL*)
0013
0014   sStringRead  :STRING(6);      (*чтение STRING из регистра панели PSW320-322*)
0015   sStringWrite :STRING(6);      (*запись STRING в регистры панели PSW320-322*)
0016 END_VAR

```

Рис. 33. Объявление переменных в программе **PLC_PRG**

Восемь переменных (**Read** и **Write**) соответствуют восьми переменным (**Input** и **Output**) из **Конфигурации ПЛК**; функциональный блок **SWAP_WORD** используется для перестановки **WORD** в переменной типа **REAL**, поскольку порядок **WORD** в переменных с плавающей точкой в панели инвертирован по сравнению с **ПЛК**. Код функционального блока приведен ниже:



```

0001 FUNCTION_BLOCK Swap_WORD
0002 VAR_INPUT
0003   rin:REAL;   (*входная переменная типа REAL*)
0004   END_VAR
0005 VAR_OUTPUT
0006   rOut:REAL;  (*переменная rIn с переставленными местами WORDами*)
0007 END_VAR
0008
0009 VAR
0010   pInput   : POINTER TO ARRAY[0..1] OF WORD; (*вспомогательные переменные*)
0011   pOutput  : POINTER TO ARRAY[0..1] OF WORD;
0012 END_VAR
0013
0014 pInput := ADR(rin);
0015 pOutput := ADR( rOut);
0016
0017
0018 pOutput[ 0 ] := pInput[ 1 ];
0019 pOutput[ 1 ] := pInput[ 0 ];

```

Рис. 34. Содержимое функционального блока **Swap_WORD**

Код программы **PLC_PRG**, написанной на языке **CFC**, выглядит следующим образом:

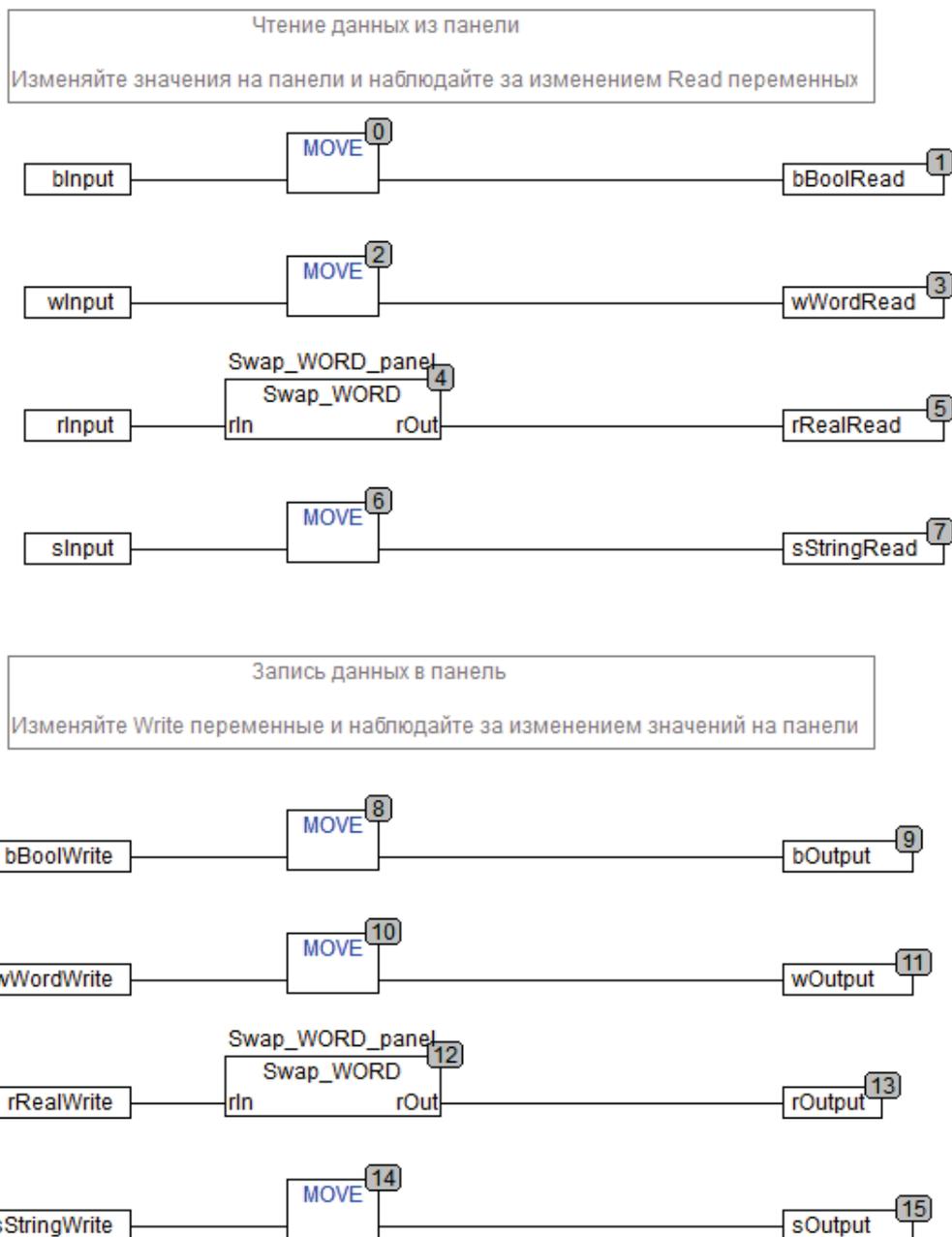


Рис. 35. Код программы **PLC_PRG**

Программа **PLC_PRG** каждый цикл выполняет следующие действия:

1. значение переменной **bInput** присваивается переменной **bBoolRead**;
2. значение переменной **wInput** присваивается переменной **wWordRead**;
3. значение переменной **rInput** обрабатывается с помощью функционального блока **Swap_WORD** (**WORD** в **REAL** меняются местами для корректного отображения в ПЛК; это связано с тем, что порядок **WORD** в панели инвертирован по сравнению с ПЛК) и записывается в переменную **rRealRead**;
4. значение переменной **sInput** присваивается переменной **sStringRead**;
5. значение переменной **bBoolWrite** присваивается переменной **bOutput**;
6. значение переменной **wWordWrite** присваивается переменной **wOutput**;
7. значение переменной **rRealWrite** обрабатывается с помощью функционального блока **Swap_WORD** (**WORD** в **REAL** меняются местами для корректного отображения в панели; это связано с тем, что порядок **WORD** в панели инвертирован по сравнению с ПЛК) и записывается в переменную **rOutput**;
8. значение переменной **sStringWrite** присваивается переменной **sOutput**.

В процессе работы проекта пользователь может изменять значения **Write** переменных (непосредственно в **CoDeSys**), наблюдая за соответствующими изменениями на дисплее панели, а также изменять данные на панели (с помощью сенсорного ввода) и наблюдать изменения у соответствующих **Read** переменных.

3.3. Конфигурация задач

Обратите внимание, что для работы программы **PLC_PRG**, она должна быть привязана к одной из задач (вкладка **Ресурсы**, компонент **Конфигурация задач**).

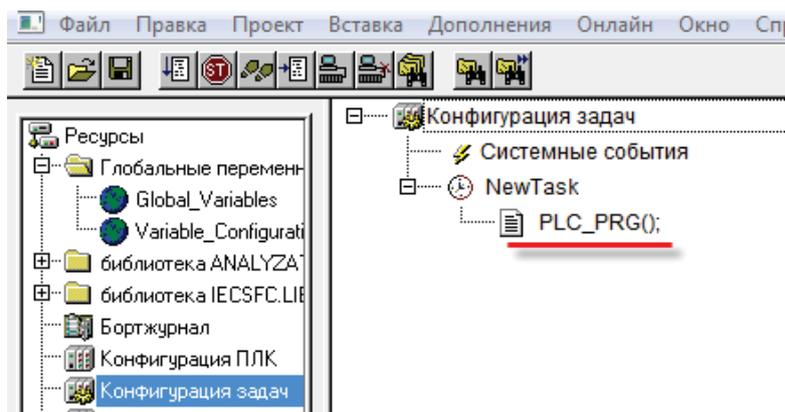


Рис. 36. Привязка программы **PLC_PRG** к задаче

4. Работа с примером

1. Откройте проект **ModbusRTUmaster.pro** в **CoDeSys**, подключитесь к ПЛК, загрузите проект и запустите его:

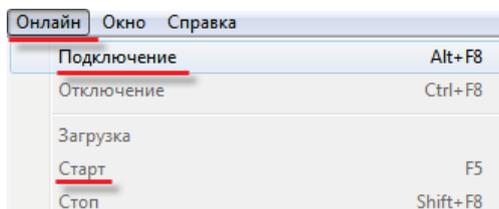


Рис. 37. Загрузка и запуск проекта для ПЛК в **CoDeSys**

2. Откройте **ModbusRTUslave.txp** в программе **Конфигуратор СП300** и загрузите его в панель:

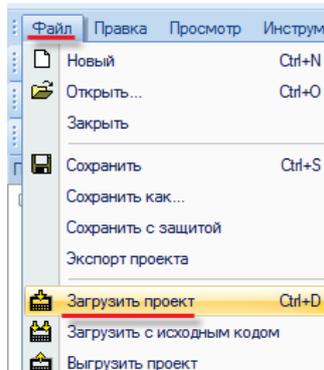


Рис. 38. Загрузка проекта в панель

3. Соедините панель и ПЛК с помощью кабеля (схемы соединительных кабелей приведены в **РЭ** панели СП3хх):

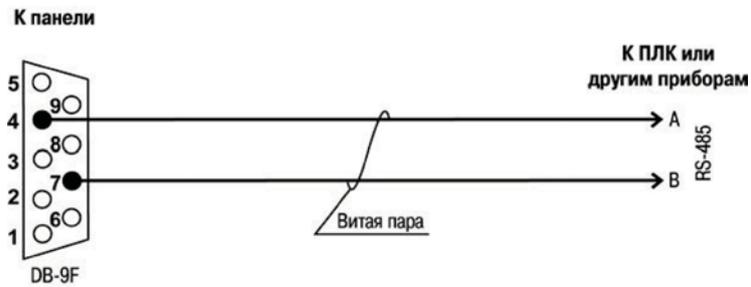


Рис. 39. Соединительный кабель для связи панели с ПЛК по интерфейсу **RS-485**

4. введите данные с помощью сенсорного дисплея панели и наблюдайте, как они будут считаны в **Read** переменные в программе **PLC_PRG**:



Рис. 40. Ввод значений с помощью сенсорного дисплея панели

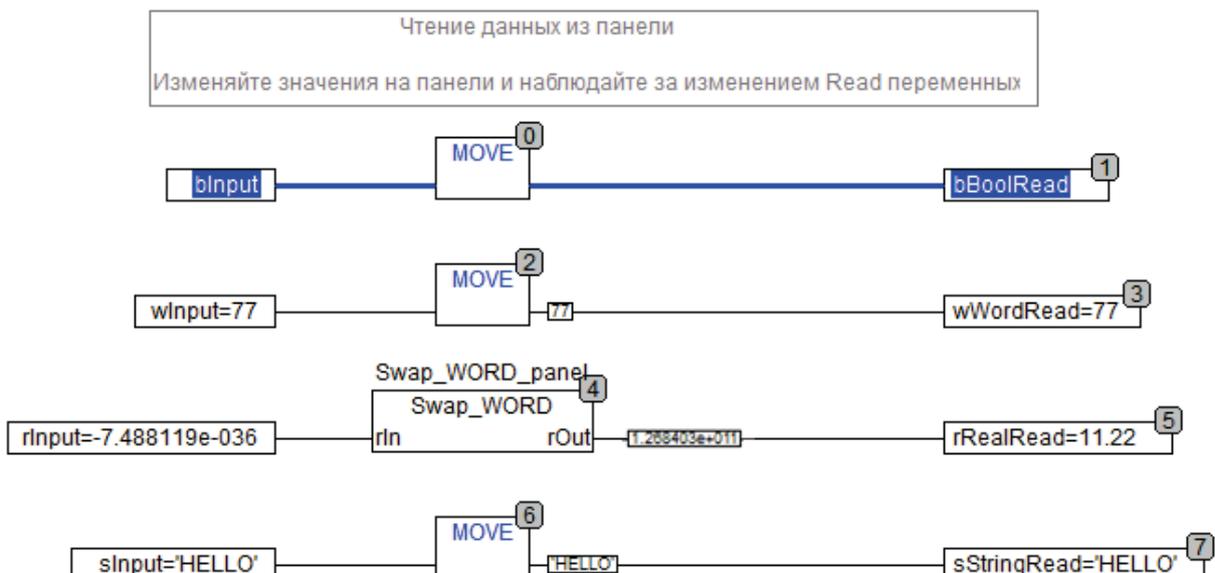


Рис. 41. Отображение введенных на панели значений в **CoDeSys**

5. введите новые значения **Write** переменных в программе **PLC_PRG**, запишите их с помощью комбинации клавиш **Ctrl+F7** и наблюдайте, как они будут записаны в панель и отображены на ее дисплее:

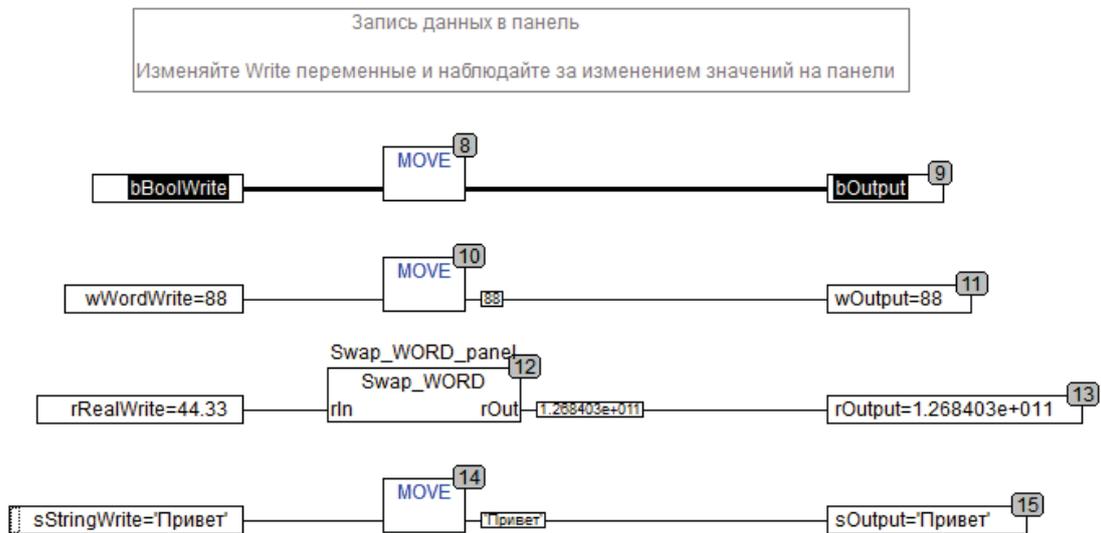


Рис. 42. Изменение значений переменных в CoDeSys

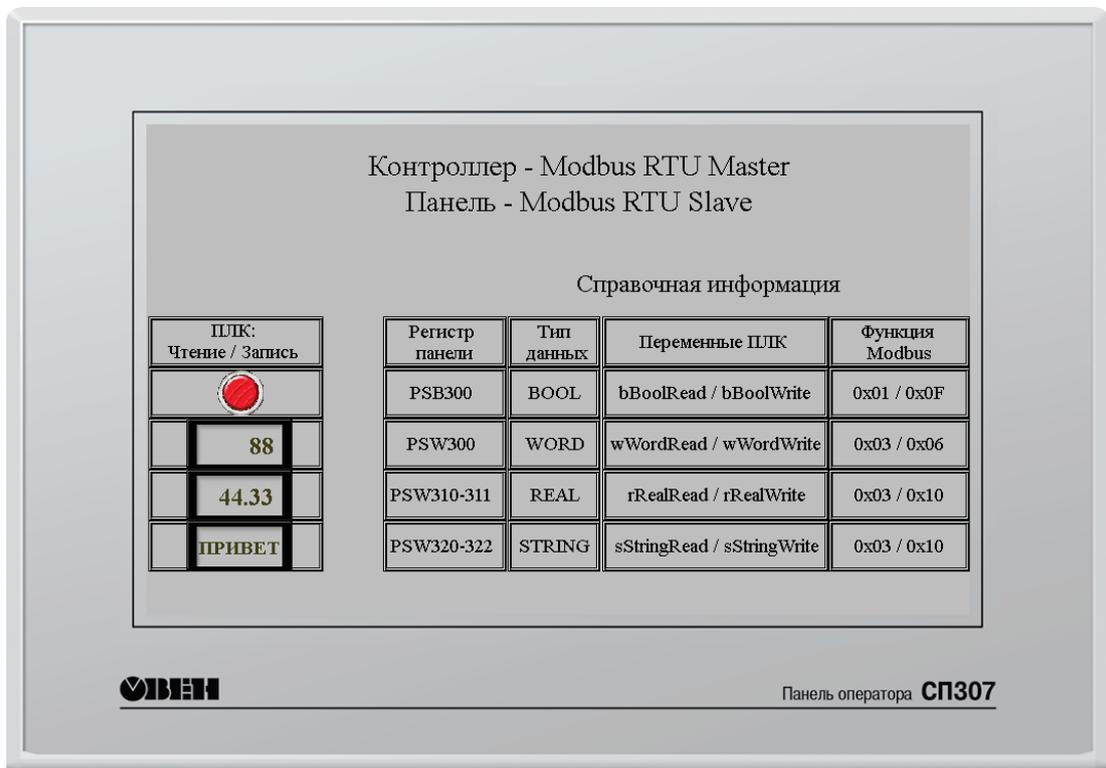


Рис. 43. Отображение введенных в CoDeSys значений на панели

8. Учебно-методическое и информационное обеспечение дисциплины

8.1. Основная литература

1. Деменков Н.П. Программирование и конфигурирование промышленных сетей: учебное пособие / Деменков Н.П. — Москва: Московский государственный технический университет имени Н.Э. Баумана, 2010. — 116 с. — ISBN 2227-8397. — Текст : электронный // Электронно-библиотечная система IPR BOOKS : [сайт]. — URL: <http://www.iprbookshop.ru/31176.html>

2. Скляр О.К. Волоконно-оптические сети и системы связи / Скляр О.К. — Москва : СОЛОН-Пресс, 2016. — 266 с. — ISBN 5-98003-147-2. — Текст : электронный // Электронно-библиотечная система IPR BOOKS : [сайт]. — URL: <http://www.iprbookshop.ru/90258.html>

3. Ившин В.П. Беспроводная сеть сбора и передачи измерительной информации в АСУТП : учебное пособие / Ившин В.П. — Казань : Казанский национальный исследовательский технологический университет, 2016. — 240 с. — ISBN 978-5-7882-1848-9. — Текст : электронный // Электронно-библиотечная система IPR BOOKS : [сайт]. — URL: <http://www.iprbookshop.ru/61960.html>

4. Федоров Ю.Н. Справочник инженера по АСУТП. Проектирование и разработка : учебно-практическое пособие / Федоров Ю.Н. — Вологда : Инфра-Инженерия, 2016. — 928 с. — ISBN 978-5-9729-0019-0. — Текст : электронный // Электронно-библиотечная система IPR BOOKS : [сайт]. — URL: <http://www.iprbookshop.ru/5060.html>

5. Симаков, Г. М. Микропроцессорные системы управления электроприводами и технологическими комплексами : учебное пособие / Симаков Г. М. , Бородин А. М. , Котин Д. А. , Панкрац Ю. В. - Новосибирск : Изд-во НГТУ, 2016. - 116 с. - ISBN 978-5-7782-2989-1. - Текст : электронный // ЭБС "Консультант студента" : [сайт]. - URL : <https://www.studentlibrary.ru/book/ISBN9785778229891.html>

6. Хиврин, М. В. Программирование ПЛК и промышленные сети. Программное обеспечение управления технологическими процессами : лаб. практикум / М. В. Хиврин, С. В. Данильченко. - Москва : МИСиС, 2020. - 139 с. - Текст : электронный // ЭБС "Консультант студента" : [сайт]. - URL : https://www.studentlibrary.ru/book/Misis_488.html

7. Тугов, В. В. Проектирование автоматизированных систем управления в TRACE MODE : учебное пособие / Тугов В. В. - Оренбург : ОГУ, 2017. - ISBN 978-5-7410-1857-6. - Текст : электронный // ЭБС "Консультант студента" : [сайт]. - URL : <https://www.studentlibrary.ru/book/ISBN9785741018576.html>

8.2. Дополнительная литература

1. Герасимов, А. В. Проектирование автоматизированных систем управления технологическими процессами : учебное пособие / Герасимов А. В. - Казань : Издательство КНИТУ, 2016. - 124 с. - ISBN 978-5-7882-1987-5. - Текст : электронный // ЭБС "Консультант студента" : [сайт]. - URL : <https://www.studentlibrary.ru/book/ISBN9785788219875.html>

2. Хиврин, М. В. Аппаратное и программное обеспечение управления технологическими процессами : учебно-методическое пособие / М. В. Хиврин - Москва : МИСиС, 2015. - 95 с. - Текст : электронный // ЭБС "Консультант студента" : [сайт]. - URL : https://www.studentlibrary.ru/book/Misis_246.html

3. Белоус, А. И. Кибербезопасность объектов топливно-энергетического комплекса. Концепции, методы и средства обеспечения / А. И. Белоус. - Москва : Инфра-Инженерия, 2020. - 644 с. - ISBN 978-5-9729-0512-6. - Текст : электронный // ЭБС "Консультант студента" : [сайт]. - URL : <https://www.studentlibrary.ru/book/ISBN9785972905126.html>

9. Материально-техническое обеспечение дисциплины

При выполнении студентами практических заданий используются технические средства обучения. Технические средства обучения – сосредоточены в компьютерных спец. лабораториях кафедры (ауд. 4-29, 4-35, 4-37).

Студенты полностью обеспечены учебными и методическими материалами для организации их обучения и контроля результатов.

Разработчик:

Доцент каф. «АТПП»



/Шухин В.В./

Согласовано:

И.о. зав. кафедрой «АТПП»



/Хакимов З.Л./

Директор ДУМР



/Магомаева М.А./

