

# **CODESYS V3.5**

# Примеры настройки обмена с ПЛК110 [М02]



01.12.2018 версия 2.0

## Оглавление

1 L	<b>L</b> ель и структура документа	3
2 (	Основные особенности настройки обмена для CDS V3.5 и CDS V2.3	4
2.1	CODESYS V3.5: Соответствие нумерации СОМ-портов контроллера и CODESYS	4
2.2	CODESYS V3.5: Использование объединений (UNION)	4
2.3	CODESYS V3.5: области памяти slave-устройства	6
2.4	CoDeSys V2.3: выравнивание данных в Modbus Slave	6
3 N	/lodbus RTU. ПЛК (CDS V3.5) – master, ПЛК (CDS V2.3) – slave	7
3.1	Описание примера	7
3.2	Настройка ПЛК (slave)	8
3.3	Настройка СПК (master)	13
3.4	Работа с примером	19
4 N	/lodbus RTU. ПЛК (CDS V3.5) – slave, ПЛК (CDS V2.3) – master	21
4.1	Описание примера	
4.2	Настройка СПК (slave)	22
4.3	Настройка ПЛК (master)	
4.4	Работа с примером	34
5 N	/lodbus TCP. ПЛК (CDS V3.5) – master, ПЛК (CDS V2.3) – slave	36
5.1	Описание примера	
5.2	Настройка ПЛК (slave)	
5.3	Настройка СПК (master)	41
5.4	Работа с примером	47
6 N	/lodbus TCP. ПЛК (CDS V3.5) – slave,ПЛК (CDS V2.3) – master	49
6.1	Описание примера	49
6.2	Настройка СПК (slave)	50
6.3	Настройка ПЛК (master)	55
6.4	Работа с примером	61

### 1 Цель и структура документа

Настоящее руководство описывает настройку обмена данными между контроллерами, программируемыми в CODESYS V3.5 и CoDeSys V2.3 по протоколу Modbus. Подробные сведения о работе с Modbus приведены в других документах – CODESYS V3.5. Modbus и Руководство пользователя ПЛК, которые доступных на дисках с ПО из комплекта поставки и сайте компании <u>OBEH</u>.

Руководство предполагает, что пользователь уже имеет базовые навыки работы с соответствующими приборами.

Контроллеры могут работать как в режиме Master, так и в режиме Slave. Пользователь должен выбрать режим для приборов, основываясь на специфике конкретной задачи и используемого оборудования.

Документ содержит 4 примера:

- 1. <u>Протокол Modbus RTU. ПЛК (CDS V3.5) master, ПЛК (CDS V2.3) slave</u>.
- 2. <u>Протокол Modbus RTU. ПЛК (CDS V3.5) slave, ПЛК (CDS V2.3) master</u>.
- 3. <u>Протокол Modbus TCP. ПЛК (CDS V3.5) master, ПЛК (CDS V2.3) slave</u>.
- 4. <u>Протокол Modbus TCP. ПЛК (CDS V3.5) slave, ПЛК (CDS V2.3) master</u>.

# 2 Основные особенности настройки обмена для CDS V3.5 и CDS V2.3

# 2.1 CODESYS V3.5: Соответствие нумерации СОМ-портов контроллера и CODESYS

Во время настройки интерфейсов RS-232/RS-485 в **CODESYS V3.5** следует указывать номера портов. Номера портов перечислены в руководстве **CODESYS V3.5. FAQ.** 

### 2.2 CODESYS V3.5: Использование объединений (UNION)

Стандарт **Modbus** предусматривает только два типа данных, участвующих в обмене – **BOOL** и **WORD**. Достаточно часто возникает потребность передать данные других типов, например, **REAL** и **STRING**. В данном случае на устройстве, которое отправляет данные, следует преобразовать их в последовательность **WORD** регистров. Соответственно, на устройстве, получающем данные, должно быть выполнено обратное преобразование. Наиболее простой способ сделать это в **CODESYS V3.5** – использовать тип данных <u>объединение (Union)</u>. Во время настройки ПЛК в **CoDeSys V2.3** это не требуется, т. к. у пользователя уже есть готовые модули для каждого типа данных.

**Объединение (UNION)** представляет собой пользовательский тип данных, все переменные которого расположены в одной области памяти. Таким образом, переменные различных типов будут представлять различную интерпретацию одних и тех же данных. Для конвертации достаточно записать значение в одну из переменных объединения и считать его из другой.

Для конвертации значения с плавающей точкой, хранящегося в двух **WORD**, в переменную типа **REAL** следует:

**1.** Нажать **ПКМ** на приложение **Application** и добавить объект **DUT** типа **объединение** с названием **Real\_Word**:



Рисунок 2.1 – Добавление в проект объединения

2. В объединении объявить переменную rRealValue типа REAL и массив awModbusReal типа WORD, содержащий два элемента:



Рисунок 2.2 – Объявление переменных объединения

3. В программе объявить экземпляр объединения Real\_Word с названием \_2WORD\_TO\_REAL:



Рисунок 2.3 – Объявление экземпляра объединения в программе

Для использования переменных объединения в нужном месте программы следует ввести имя экземпляра объединения и нажать точку, после чего выбрать из списка нужную переменную:

 []
📦 awModbusReal
🖗 rRealValue
Wittealvalue

Рисунок 2.4 – Работа с переменными объединения в программе

4. Переменные массива **awModbusReal** будут привязаны к регистрам во время настройки **Modbus**, переменная **rRealValue** будет использоваться в программе для работы со значением с плавающей точкой.

На рисунке 2.5 приведен скриншот значения переменных объединения в процессе работы программы. Во время записи в элементы массива **awModbusReal** значений в HEX, в переменную **rReaLValue** будет записано соответствующее им значение с плавающей точкой.

Device.Application.PLC_PRG					
\$\Phi\$ \$\Phi\$ _2WORD_TO_REAL	REAL_WORD				
rRealValue	REAL	1.1			
🗏 < awModbusReal	ARRAY [01] OF WORD				
awModbusReal[0]	WORD	16#CCCD			
awModbusReal[1]	WORD	16#3F8C			

Рисунок 2.5 – Отображение значений переменных объединения

### 2.3 CODESYS V3.5: области памяти slave-устройства

Во время настройки ПЛК в **CODESYS V3.5** в режиме **Modbus Slave** следует обращать внимание на следующие особенности:

- **1.** Input Registers (регистры ввода) доступны только для чтения, Holding Registers (регистры временного хранения) для чтения и записи.
- 2. Области Coils и Holding Registers, Discrete Inputs и Input Registers наложены друг на друга.
- 3. Значения Holding регистров не могут быть изменены из программы контроллера.

### 2.4 CoDeSys V2.3: выравнивание данных в Modbus Slave

Во время настройки ПЛК в **CoDeSys V2.3** в режиме **Modbus Slave** следует выравнивать данные по регистрам (регистр занимает 2 байта, т. е. 16 бит). Следует учитывать, что адреса регистров размещаемых данных должны быть <u>кратны количеству байт</u> этих данных.

В качестве примера в slave-устройстве будет использоваться элемент **8 bits**, два элемента **2 byte** и элемент **float**. Размещать их последовательно <u>нельзя</u>. Выравненные данные в данном случае будет выглядеть следующим образом:

Элемент	Комментарий	Адреса регистров	Адреса битов (в абсолютной адресации)
8 bits			07
8 bits	Добавлен для выравнивания	0	815
2 byte		1	1631
2 byte		2	3247
2 byte	Добавлен для выравнивания	3	4863
float		4	6479

### Таблица 2.1 – Карта регистров slave-устройства

Более подробная информация о выравнивании содержится в документе **Руководство пользователя ПЛК**, доступном на диске с ПО из комплекта поставки и сайте компании <u>OBEH</u>.

# 3 Modbus RTU. ПЛК (CDS V3.5) – master, ПЛК (CDS V2.3) – slave

### 3.1 Описание примера

Пример посвящен настройке обмена данными между сенсорным панельным контроллером СПК1хх [M01] и контроллером ПЛК110 [M02] по протоколу Modbus RTU. В примере СПК выполняет функцию Master, а ПЛК – Slave.

Основные характеристики используемых устройств приведены в таблице 3.1. Используемые в примере переменные описаны в таблице 3.2.

Устройство	СПК1хх [М01]	ПЛК110 [М02]	
Функция	Master	Slave	
Используемый порт (нумерация на корпусе)	RS-485 (COM1)	RS-485 (1)	
Настройки обмена	115200, 8 бит, 1 стоп бит, без контроля четности		
Slave ID	_	1	
Таргет	3.5.11.50 (2)	PLC110.30-М v2 (версия 3.12)	
Версия прошивки	1.0.0814.1125	0.3.72	
Среда разработки проекта	CODESYS V3.5 SP11 Patch 5	CoDeSys V2.3.9.41	
Название файла проекта	ModbusRTUmaster.projectarchive	ModbusRTUslave.pro	

### Таблица 3.1 – Характеристики устройств

### Таблица 3.2 – Список переменных

СПК1хх[М01]		ПЛК110 [М	02] (Slave)	
Переменные, в которые считываются значения из Slave	Переменные, значения которых записываются в Slave	Тип данных	Переменные ПЛК	Адрес регистра/бита
xVarRead	xVarWrite	BOOL	xVar	0/0
wVarRead	wVarWrite	WORD	wVar	1
rVarRead	rVarWrite	REAL	rVar	2–3
sVarRead	sVarWrite	STRING(6)	sVar	4–6

Проекты примера доступны для скачивания: <u>Example\_SpkModbusRtuMaster.zip</u>

### 3.2 Настройка ПЛК (slave)

Для настройки ПЛК (slave) следует:

- 1. Создать новый проект CoDeSys V2.3 для ПЛК110 с программой PLC\_PRG на языке CFC.
- **2.** В компоненте **Конфигурация ПЛК** (вкладка **Ресурсы**) настроить регистры Modbus и привязать к ним переменные.

Нажать **ПКМ** на название контроллера (в данном примере – **PLC110\_30**) и добавить подэлемент **Modbus (Slave)**:

CoDeSvs - ModbusRTUslave.pro - I	Конфигурация ПЛКІ	and the second second			_	
Файл Правка Проект Встан	вка Дополнения Онла	айн Окно Справка				
		•				
Ресурсы Global_Variables Gl	PLC110_30     PLC110_30     PFast discr     m-Fast discr     m-Fas	Вставить Элемент Добавить Подэлемент Заменить элемент Вычислять адреса Вырезать Копировать Вставить Удалить	Ctrl+X Ctrl+C Ctrl+V Del	астройки Statis Exter Univ Mod DCO OWE OWE Owe Mod Arch	Параметры модуля stic nded setings ersal network module Bus (Master) N (Master) N (slave) N (spy) n (Master) Bus (slave) iver	

Рисунок 3.1 – Конфигурация ПЛК. Добавление Modbus (Slave)

В его настройках выбрать адрес (Slave ID), равный 1 (в соответствии с таблицей 3.1):

Базо	вые парам	етры Пар	раметры модуля	
	Индекс	Имя	Значение	По умолч.
	1 2	Name Address	ModBus (slave) 1	ModBus (slave) 1

Рисунок 3.2 – Конфигурация ПЛК. Настройка Modbus (Slave)

**3.** Выбрать порт ПЛК, который будет использоваться для связи с контроллером. Для выбора следует нажать **ПКМ** на элемент **Modbus (FIX)** и добавить подэлемент **RS-485-1** (согласно таблице 3.1).

 Вставить Элемент		
Добавить Подэлемент	+	Debug RS-232
Заменить элемент		RS-485-1
Вычислять адреса		RS-232
Вырезать	Ctrl+X	TCP
Копировать	Ctrl+C	Modem
Вставить	Ctrl+V	
Удалить	Del	

Рисунок 3.3 – Конфигурация ПЛК. Добавление подэлемента RS-485-1

В параметрах подэлемента указать сетевые настройки в соответствии с <u>таблицей 3.1</u> и используемый протокол – **RTU**.

Базон	вые парам	етры Параметры моду	ля		
	Индекс	Имя	Значение	По умолч.	Мин.
	1	Name	RS-485-1	RS-485-1	
	2	Communication speed	115200 🔄 🔽	115200	
	3	Parity	NO PARITY 🔽	NO PARITY C	
	4	Data bits	8 bits 📃 🔽	8 bits	
	5	Stop length	One stop bit 🔄 🔽	One stop bit	
	6	Interface Type	RS485 🚽 🚽	RS485	
	7	Frame oriented	RTU 🔽	ASCII	
	8	Framing time ms	0	0	0
	9	Visibility	No 💌	No	

Рисунок 3.4 – Конфигурация ПЛК. Настройки подэлемента RS-485-1

- 4. Нажать ПКМ на элемент Modbus (Slave) и добавить следующие подэлементы:
  - 8 bits (для BOOL);
  - 8 bits (для обеспечения выравнивания памяти);
  - 2 byte (для WORD);
  - Float (для REAL);
  - 3 элемента 2 byte (для STRING из 6 символов).

3. Modbus RTU. ПЛК (CDS V3.5) - master, ПЛК (CDS V2.3) - slave



Рисунок 3.5 – Конфигурация ПЛК. Добавление подэлементов

В результате **Конфигурация ПЛК** будет выглядеть следующим образом (см. рисунок 3.6). Затем следует объявить переменные (после ввода имен они автоматически будут добавлены в список глобальных переменных проекта). Для ввода имени переменной два раза нажмите на **AT**.

### ПРИМЕЧАНИЕ

i

К регистрам 4–6 переменные не привязываются. Соответствующая им переменная **sVar** будет объявлена в программе **PLC\_PRG** (в пп. 5).



Рисунок 3.6 – Внешний вид Modbus (Slave) с добавленными подэлементами

### ПРИМЕЧАНИЕ

Нумерация регистров в CODESYS всегда начинается с нуля, каждый регистр физически занимает два байта (16 бит). Поэтому переменная типа **REAL** займет два регистра (с адресами 2 и 3). Переменная типа **STRING**, которой соответствует три **2 byte** элемента, займет регистры с адресами 4–6. Это необходимо учитывать во время настройки masterустройства.

_		, - ( )		
5.	Программа <b>PLC_PRG</b> будет вы	глядеть следующим образом:		
0001	PROGRAM PLC_PRG			
0002	sVar AT %QW6.5.0: STRING(6);	(*собираем STRING переменную из трех WORD [т.е	STRING переменную из трех WORD [т.е. шести символов],	
0004	END_VAR	указывая адрес (см. конфигурация г лік) первого и:	зних^)	
0006	-			
0007				
0000	•			
			1	
	СІ ІК считывает/записыва	ет значения из ПЛК		
	Изменяйте значения перемен	ных и наблюдайте соответствующие изменения в СПК		
	Изменяите значения в СПК и н	наолюдаите за изменением переменных		
	xVar -			
	wVar			
	With			
	rVar —			
	sVar –			

Рисунок 3.7 – Код программы PLC\_PRG

Настройка ПЛК (slave) завершена.

# i

### ПРИМЕЧАНИЕ

Данный проект не содержит операций и используется только для отображения и ввода значений. Пользователь должен создать программу для реализации необходимых алгоритмов.

### 3.3 Настройка СПК (master)

Для настройки СПК (master) следует:

- 1. Создать новый проект CODESYS V3.5 для СПК1хх [M01] с программой PLC\_PRG на языке CFC.
- 2. Добавить в проект <u>объединение</u> с именем Real\_Word:



Рисунок 3.8 – Добавление в проект объединения

В объединении объявить переменную **rRealValue** типа **REAL** и массив **awModbusReal** типа **WORD**, содержащий два элемента:



Рисунок 3.9 – Объявление переменных объединения

3. Добавить в проект объединение с именем String\_Word.

В объединении объявить переменную sStringValue типа STRING и массив awModbusString типа WORD, содержащий три элемента (STRING сможет содержать до 6 символов, поскольку каждый WORD может содержать два символа):

/ <b>*</b> \$	Stri	ng_Word ×
	1	TYPE String_Word :
	2	UNION
	3	<pre>sStringValue: STRING(6);</pre>
	4	awModbusString: ARRAY [02] OF WORD;
	5	END_UNION
	6	END_TYPE

Рисунок 3.10 – Объявление переменных объединения

**4.** Объявить в программе **PLC\_PRG** девять переменных – 4 из них будут использоваться для отображения данных, считанных из ПЛК, еще 4 – для ввода данных, которые будут записаны в ПЛК. Последняя переменная будет являться триггером записи.

<u></u>	PLC_	PRG X	
	1	PROGRAM PLC_PRG	
-	2	VAR	
	3	(*данные, из счи	(тываемые ПЛК*)
	4	xVarRead:	BOOL;
	5	wVarRead:	WORD;
	6	rVarRead:	Real_Word;
	7	sVarRead:	String_Word;
	8		
	9	(*данные, запись	ваемые в ПЛК*)
1	10	xVarWrite:	BOOL;
1	1	wVarWrite:	WORD;
1	2	rVarWrite:	Real_Word;
1	.3	sVarWrite:	String_Word;
1	.4		
1	.5	xTrigger:	BOOL; // триггер записи
1	.6	END_VAR	

Рисунок 3.11 – Объявление переменных программы

5. Код программы будет выглядеть следующим образом:

Чтение данных из ПЛК Изменяйте значения в ПЛК и наблюдайте за изменением значений переменных	Запить данных в ПЛК Изменяйте значения переменных, активируйте триггер записи и наблюдайте за изменением значений в ПЛК
xVarRead -	xVarWrite
wVarRead	wVarWrite
rVarRead.rRealValue	rVarWrite.rRealValue
sVarRead.sStringValue	sVarWrite.sStringValue -
	xTrigger -

Рисунок 3.12 – Код программы на языке CFC

6. Добавить в проект устройство Modbus COM.



### ПРИМЕЧАНИЕ

Версия компонента не должна превышать версию таргет-файла контроллера. Подробнее см. в документе **CODESYS V3.5. Описание таргет-файлов**.

### 3. Modbus RTU. ПЛК (CDS V3.5) - master, ПЛК (CDS V2.3) - slave

Устройства 👻	џ,	< .		🗊 Добавить устройство
ModbusRTUmaster	•			
Master (SPK1xx[M01])			_	Имя: Modbus_COM
Plc Logic	*	Вырезать		Reference
Application	8	Копировать		
* Real_Word	Ē.	Вставить		Добавить устройство     Обновить устройство     Обновить устройство     Обновить устройство     Обновить устройство
String_Word	$\times$	Удалить		
🛍 Менеджер библиотек		Обзор 🔸		оведите строку для полнотекстового поиска во с производитель. <all vendors=""></all>
PLC_PRG (PRG)	e	Свойства		Имя Производитель Версия Описание: ^
🖹 🌃 Конфигурация задач	*::	Добавление объекта		₽Ш Разн.
🖻 🍪 MainTask	6	Добавить папку		🖻 🔟 Промышленные сети (fieldbus)
PLC_PRG		Добавить устройство	$\longrightarrow$	* 🔊 АS-интерфейс
· 🕓 RTC (RTC)		Обновить устройство		R CANbus
- 🕹 OwenCloud (OwenCloud)	Cĩ,	Редактировать объект		⊕ Brd Ethercat
- 💖 Buzzer (Buzzer)		Редактировать объект в		🖲 🎫 Ethernet-адаптер
Drives (Drives)		Изменить І/О-соотнесение		* 😝 Ethernet/IP
品 Network (Network)		Импорт соотнесений из CSV		= • • • • Modbus
C Screen (Screen)		Экспортировать соотнесения в CSV		= I II Flop Modbus Serial
Debug (Debug)	К	Режим онлайн-конфигурации		Modbus COM 3S - Smart Software Solutions GmbH 3.5.11.20 A serial COM Port
info (Info)		Сброс заводской устройства [Master]		
		Эмуляция		
	-			
				Группировать по категориям Отображать все версии (для экспертов) Показать устаревшие версии
				IMM9: Modbus COM     Iponsbogurrenb: 35 - Smart Software Solutions GmbH     Fpyrnha: Fopr Modbus Serial     Bepcra: 3.5.11.2     Homep Hogenit: -     Onicative: A serial COM Port on a Windows PC.
				дочавить выкранное устроиство как последнего потомка Master • (Можно выбрать другой таргет-узел, пока окно открыто.)
				Добавить устройство Закрыть

Рисунок 3.13 – Добавление устройства Modbus COM

В конфигурации СОМ-порта следует указать сетевые настройки в соответствии с <u>таблицей 3.1</u> и номер порта. Нумерация СОМ-портов приведена в документе **CODESYS V3.5. FAQ.** 

Modbus_COM ×		
Общее	Конфигурация последоват	гельного порта
SerialPort Конфигурация	СОМ-порт	1
Состояние	Скорость передачи	115200 ~
Информация	Четность	NONE ~
	Биты информации	8
	Стоповые биты	1

Рисунок 3.14 – Настройки СОМ-порта СОМ1

7. В СОМ-порт добавить компонент Modbus Master.



### ПРИМЕЧАНИЕ

Версия компонента не должна превышать версию таргет-файла контроллера. Подробнее см. в документе **CODESYS V3.5. Описание таргет-файлов**. 3. Modbus RTU. ПЛК (CDS V3.5) - master, ПЛК (CDS V2.3) - slave

Устройства	• # ×						
ModbusRTUmaster	•		a contraction of the second se				~
Master (SPK1xx[M01])			Шдо	равить устроиство			~
Pic Logic			Inc	Modbus Master COM Port			
Application			VIPO.	houbus_haster_com_roit			
- * Real_Word			Дейс	твие			
String_Word			. До	бавить устройство 🛛 🔘 Вставить	устройство ОПодключить	устройство 🛛 🔾	Обновить устройсте
前 Менеджер библиот	ек						
PLC_PRG (PRG)			Введи	те строку для полнотекстового пои	ска во в Производитель: <А	vendors>	$\sim$
🖻 🧱 Конфигурация зада	14		Имя		Производитель	Версия	Описание:
🖻 😻 MainTask			=-(**	Промышленные сети (fieldbus)			
● PLC_PRG				Modbus			
Modbus_COM (Modbus COI	M) X	Вырезать		Macrep Modbus Serial			
U RTC (RTC)		Копировать		Modbus Master, COM Port	3S - Smart Software Solutions Gr	nbH 3.5.11.20	A device that
OwenCloud (OwenCloud)	123	Вставить		• III Устройство Modbus Serial			
N Buzzer (Buzzer)	×	Удалить					
Dives (Drives)		06300					
The Network (Network)	6	Свойства					
B Dobug (Dobug)	200						
The second secon							
and (mo)	-						
		Вставить устройство					
		Отка устройство					
		Обновить устройство	<				>
	D <sup>2</sup>	Редактировать объект					
	-	Редактировать объект в					
		Изменить І/О-соотнесение					
		Импорт соотнесений из CSV			ражать все версии (для экспертов		аревшие версии
		Экспортировать соотнесения в CSV					
		Эмуляния	9	Имя: Modbus Master, COM Port	uara Calutions Cashil		
	-	strift map of		Группы: Mactep Modbus Serial	vare solutions GmbH		
				Версия: 3.5.11.20			<u></u>
				номер модели: - Описание: A device that works as a	Modbus Master on a serial COM Po	rt of a	× 1
				Windows PC.			
			Доба	вить выбранное устройство как г	юследнего потомка		
			Modt	ous_COM			
			•	(Можно выбрать другой таргет-узел,	пока окно открыто.)		
						Побавить устройство	Закоыть
						цооконтв устроиство	Jacporte

Рисунок 3.15 – Добавление компонента Modbus Master

В настройках компонента поставить галочку Автоперезапуск соединения.

2	Modbus_Master_COM_F	Port ×			
	Общее ModbusGenericSerialMaster	Modbus-RTU/ASCII		_	MODBUS
	Соотнесение входов/выход	Режим передачи	🔘 RTU	○ ASCII	
	Состояние	Таймаут ответа (мс)	1000		
	Информация	Время между фреймами (мс)	10		
		Автоперезапуск соединения	I		

Рисунок 3.16 – Настройка компонентов Modbus Master

8. B Modbus Master добавить компонент Modbus Slave.



### ПРИМЕЧАНИЕ

Версия компонента не должна превышать версию таргет-файла контроллера. Подробнее см. в документе **CODESYS V3.5. Описание таргет-файлов**.



Рисунок 3.17 – Добавление компонента Modbus Slave в проект

В настройках компонента на вкладке **Общее** следует указать адрес slave-устройства в соответствии с <u>таблицей 3.1</u>.

Modbus_Slave_COM_Po	rt x		
Общее	Modbus-RTU/ASCII		NODDIE
Канал Modbus Slave	Адрес слейва [1247]	1	MUDDO2
Modbus Slave Init	Таймаут ответа (мс)	1000	
ModbusGenericSerialSlave Соотнесение входов/выход	ravina yr orbera (ney	1000	
Состояние			
Информация			

Рисунок 3.18 – Настройки компонента Modbus Slave в проект

На вкладке **Канал Modbus Slave** следует создать 8 каналов – 4 из них будут использоваться для чтения переменных, 4 – для записи. Чтение будет осуществляться циклически, запись – по переднему фронту триггера (**RISING\_EDGE**). Используемые функции соответствуют типам данных, адреса регистров настроены согласно <u>таблице 3.2</u>.

Общее		Имя	Тип доступа	Триггер	Сдвиг READ	Длина	Обработка ошибок	Сдвиг WRITE	Длина	Комментарий
Канал Modbus Slave	0	Channel 8	Read Coils (Код функции 01)	Цикл., t#100ms	16#0000	1	Сохранить посл. значение			
	1	Channel 1	Read Holding Registers (Код функции 03)	Цикл., t#100ms	16#0001	1	Сохранить посл. значение			
Modbus Slave Init	2	Channel 2	Read Holding Registers (Код функции 03)	Цикл., t#100ms	16#0002	2	Сохранить посл. значение			
ModbusGenericSerialSlave	3	Channel 3	Read Holding Registers (Код функции 03)	Цикл., t#100ms	16#0004	3	Сохранить посл. значение			
соотнесение входов/выход	4	Channel 4	Write Single Coll (Код функции 05)	Передний фронт				16#0000	1	
Состояние	5	Channel 5	Write Single Register (Код функции 06)	Передний фронт				16#0001	1	
Информация	6	Channel 6	Write Multiple Registers (Код функции 16)	Передний фронт				16#0002	2	
	7	Channel 7	Write Multiple Registers (Код функции 16)	Передний фронт				16#0004	3	

Рисунок 3.19 – Настройка каналов Modbus Slave

На вкладке ModbusGenericSerialSlave Соотнесение входов/выходов следует привязать к каналам переменные программы в соответствии с <u>таблицей 3.2</u>. У параметра Всегда обновлять переменные должно быть выставлено значение Включено 2.

бщее	Найти переменную	Фильтр Показ	зать все	•				
Канал Modbus Slave	Переменная	Соотнесение	Канал	Адрес	Тип	Единица	Описание	
Modbus Slave Init			Channel 8	%IB0	ARRAY [00] OF BYTE			
ModbusGenericSerialSlave	8-1		Channel 8[0]	%IB0	BYTE		Read Colls	
Соотнесение входов/выход	Application.PLC_PRG.xVarRead	۹	BitO	%D(0.0	BOOL		бит 0	
Состояние	8-19		Channel 1	%IW1	ARRAY [00] OF WORD		Read Holding Registers	
tu de a su a s	Application.PLC_PRG.wVarRead	٠	Channel 1[0]	%IW1	WORD		регистр 1	
нформация	9-19		Channel 2	%IW2	ARRAY [01] OF WORD		Read Holding Registers	
	Application.PLC_PRG.rVarRead.awModbusReal[0]	۰.	Channel 2[0]	%IW2	WORD		регистр 2	
	Image: Barrier Bar	<b>*</b>	Channel 2[1]	%IW3	WORD		регистр 3	
	- <b>*</b>		Channel 3	%IW4	ARRAY [02] OF WORD		Read Holding Registers	
	Application.PLC_PRG.sVarRead.awModbusString[0]	۰.	Channel 3[0]	<del>%I₩4</del>	WORD		регистр 4	
	Application.PLC_PRG.sVarRead.awModbusString[1]	۰.	Channel 3[1]	<del>%IW5</del>	WORD		регистр 5	
	Image: Application.PLC_PRG.sVarRead.awModbusString[2]	۰	Channel 3[2]	<del>%I₩6</del>	WORD		регистр 6	
	Application.PLC_PRG.xTrigger	۰	Channel 4	%QX0.0	BIT		Trigger Variable	
	÷-*		Channel 4	%QB1	ARRAY [00] OF BYTE			
	<u>≘</u> _•		Channel 4[0]	%QB1	BYTE		Write Single Coil	
	Application.PLC_PRG.xVarWrite	<b>*</b>	BitO	%QX1.0	BOOL		бит 0	
	Application.PLC_PRG.xTrigger	۰.	Channel 5	<del>%QX2.0</del>	BIT		Trigger Variable	
	÷-•		Channel 5	%QW2	ARRAY [00] OF WORD		Write Single Register	
		<b>*</b>	Channel 5[0]	%QW2	WORD		регистр 1	
	Application.PLC_PRG.xTrigger	<b>*</b>	Channel 6	%QX6.0	BIT		Trigger Variable	
	÷-•		Channel 6	%QW4	ARRAY [01] OF WORD		Write Multiple Registers	
	Application.PLC_PRG.rVarWrite.awModbusReal[0]	۰.	Channel 6[0]	%QW4	WORD		регистр 2	
	Application.PLC_PRG.rVarWrite.awModbusReal[1]	<b>*</b>	Channel 6[1]	%QW5	WORD		регистр 3	
	Application.PLC_PRG.xTrigger	<b>*</b>	Channel 7	%QX12.0	BIT		Trigger Variable	
	<u>_</u>		Channel 7	%QW7	ARRAY [02] OF WORD		Write Multiple Registers	
	Application.PLC_PRG.sVarWrite.awModbusString[0]	۰.	Channel 7[0]	%QW7	WORD		регистр 4	
	Application.PLC_PRG.sVarWrite.awModbusString[1]	۰.	Channel 7[1]	%QW8	WORD		регистр 5	
	Application.PLC_PRG.sVarWrite.awModbusString[2]	۰,	Channel 7[2]	%QW9	WORD		регистр 6	
	· · · · · · · · · · · · · · · · · · ·							

Рисунок 3.20 – Привязка переменных к каналам

Настройка СПК (master) завершена.

### ПРИМЕЧАНИЕ

i

A Madhur Chur COM Bart u

Проект не содержит операций и используется только для отображения и ввода значений. Пользователь должен создать программу для реализации необходимых алгоритмов.

### 3.4 Работа с примером

Проекты следует загрузить проекты в оба устройства и запустить.

В случае изменения значения переменных в ПЛК, соответствующие изменения будут отображаться в программе СПК:



Рисунок 3.21 – СПК считывает данные из ПЛК

Затем следует изменить значения **write** переменных СПК и активировать триггер записи. Соответствующие изменения будут отображаться в программе ПЛК. Также новые значения будут считаны в **read** переменные программы СПК.



Рисунок 3.22 – СПК записывает данные в ПЛК

# 4 Modbus RTU. ПЛК (CDS V3.5) – slave, ПЛК (CDS V2.3) – master

### 4.1 Описание примера

Пример посвящен настройке обмена данными между сенсорным панельным контроллером СПК1хх [M01] и контроллером ПЛК110 [M02] по протоколу Modbus RTU. В примере СПК выполняет функцию Slave, а ПЛК – Master.

Основные характеристики используемых устройств приведены в таблице 4.1. Используемые в примере переменные описаны в таблице 4.2.

Устройство	СПК1хх [М01]	ПЛК110 [М02]		
Функция	Slave	Master		
Используемый порт (нумерация на корпусе)	RS-485 (COM1)	RS-485 (1)		
Настройки обмена	115200, 8 бит, 1 стоп бит, без контроля четности			
Slave ID	1	_		
Таргет	3.5.11.50 (2)	PLC110.30-М v2 (версия 3.11)		
Версия прошивки	1.0.0814.1125	0.3.72		
Среда разработки проекта	CODESYS V3.5 SP11 Patch 5	CoDeSys V2.3.9.41		
Название файла проекта	ModbusRTUslave.projectarchive	ModbusRTUmaster.pro		

### Таблица 4.1 – Характеристики устройств

### Таблица 4.2 – Список переменных

СПК1хх [М0	1] (Slave)		ПЛК110 [M02] (Master)
Переменные, которые считывает Master	Адрес регистра/бита	Тип данных	Переменные ПЛК
xVarFromSPK	0/0 (Discrete Inputs)	BOOL	xVarFromSPK
wVarFromSPK	1 (Input Registers)	WORD	wVarFromSPK
rVarFromSPK	2–3 (Input Registers)	REAL	rVarFromSPK
sVarFromSPK	4–6 (Input Registers)	STRING(6)	sVarFromSPK
Переменные, которые записывает Master	Адрес регистра/бита	Тип данных	Переменные ПЛК
xVarToSPK	0/0 (Coils)	BOOL	xVarToSPK
wVarToSPK	1 (Holding Registers)	WORD	wVarToSPK
rVarToSPK	2–3 (Holding Registers)	REAL	rVarToSPK
sVarToSPK	4–6 (Holding Registers)	STRING(6)	sVarToSPK

Проекты примера доступны для скачивания: Example\_SpkModbusRtuSlave.zip

### 4.2 Настройка СПК (slave)

Для настройки СПК (slave) следует:

- 1. Создать новый проект CODESYS V3.5 для СПК1хх [M01] с программой PLC\_PRG на языке CFC.
- 2. Добавить в проект <u>объединение</u> с именем Real\_Word:



Рисунок 4.1 – Добавление в проект объединения

В объединении объявить переменную **rRealValue** типа **REAL** и массив **awModbusReal** типа **WORD**, содержащий два элемента:

1	At Real_Word X						
	1	TYPE Real_Word :					
8	2	UNION					
	3	rRealValue: REAL;					
	4	awModbusReal: ARRAY [01] OF WORD;					
	5	END_UNION					
	6	END_TYPE					

Рисунок 4.2 – Объявление переменных объединения

3. Добавить в проект объединение с именем String\_Word.

В объединении следует объявить переменную sStringValue типа STRING и массив awModbusString типа WORD, содержащий три элемента (STRING сможет содержать до 6 символов, поскольку каждый WORD может содержать два символа):

Stri	ng_Word ×
1	TYPE String_Word :
2	UNION
3	<pre>sStringValue: STRING(6);</pre>
4	awModbusString: ARRAY [02] OF WORD;
5	END_UNION
6	END_TYPE

Рисунок 4.3 – Объявление переменных объединения

**4.** Объявить в программе **PLC\_PRG** 8 переменных – 4 из них будут считываться ПЛК, 4 – записываться ПЛК. *Следует обратить внимание* на <u>п. 2.3</u>.

DLC	C_PRG ×
1	PROGRAM PLC_PRG
₿ 2	VAR
3	(*данные, считываемые ПЛК*)
4	xVarFromSPK: BOOL;
5	wVarFromSPK: WORD;
6	rVarFromSPK: Real_Word;
7	sVarFromSPK: String_Word;
8	
9	(*данные, записываемые ПЛК*)
10	xVarToSPK: BOOL;
11	wVarToSPK: WORD;
12	rVarToSPK: Real_Word;
13	sVarToSPK: String_Word;
14	END_VAR

Рисунок 4.4 – Объявление переменных программы

5. Код программы будет выглядеть следующим образом:

ПЛК считывает данные из СПК	ПЛК записывает данные в СПК
Изменяйте значения переменных и наблюдайте за изменением значений в ПЛК	Изменяйте значения в ПЛК и наблюдайте за изменением значений переменных
xVarFromSPK -	xVarToSPK -
wVarFromSPK	wVarToSPK -
rVarFromSPK.rRealValue	rVarToSPK.rRealValue
sVarFromSPK.sStringValue -	sVarToSPK.sStringValue -

Рисунок 4.5 – Код программы на языке CFC

6. Добавить в проект устройство Modbus COM.

### ПРИМЕЧАНИЕ

i

Версия компонента не должна превышать версию таргет-файла контроллера. Подробнее см. в документе **CODESYS V3.5. Описание таргет-файлов**.



Рисунок 4.6 – Добавление устройства Modbus COM

В конфигурации СОМ-порта следует указать сетевые настройки в соответствии с <u>таблицей 4.1</u> и номер порта. Нумерация СОМ-портов приведена в документе **CODESYS V3.5. FAQ.** 

Modbus_COM ×		
Общее SerialPort Конфигурация Состояние	<ul> <li>Конфигурация последоват</li> <li>СОМ-порт</li> <li>Скорость передация</li> </ul>	115200 ×
Информация	Четность	NONE ~
	Биты информации Стоповые биты	8

Рисунок 4.7 – Настройки СОМ-порта СОМ1

7. В СОМ-порт добавить компонент Modbus Serial Device.



### ПРИМЕЧАНИЕ

Версия компонента не должна превышать версию таргет-файла контроллера. Подробнее см. в документе **CODESYS V3.5. Описание таргет-файлов**.

### 4. Modbus RTU. ПЛК (CDS V3.5) – slave, ПЛК (CDS V2.3) – master

ModbusRTUslave	•	Добавить устройство
Device (SPK1xx[M01])		
Plc Logic		Mage Modhus Serial Device
Application		
- *# Real_Word		Действие
- *\$ String_Word		Добавить устройство     Овставить устройство     Оподключить устройство     Осбновить у
前 Менеджер библиотек		
PLC_PRG (PRG)		Введите строку для полнотекстового поиска во є Производитель: «Al vendors» ····
😑 🧱 Конфигурация задач		Имя Производитель Версия Описание:
🖹 🥩 MainTask		S- 1 Промышленные сети (feldbus)
PLC_PRG		S Modbus
Modbus_COM (Modbus COM)		B Macrep Modbus Serial
S RTC (RTC)	", Вырезать	B-IN VCTDOĂCTRO MORBUS Serial
CovenCloud (OwenCloud)	Копировать	Modbus Serial Device 3S - Smart Software Solutions GmbH 3.5.11.0 A device that works a
🕸 Buzzer (Buzzer)	Вставить	
Drives (Drives)	Х Удалить	< >
品 Network (Network)	Обзор	
C Screen (Screen)	🗟 Свойства	
🙀 Debug (Debug)	Добавление объекта	
🕼 Info (Info)	Добавить папку	Группировать по категориям Отображать все версии (для экспертов) Показать устаревшие версии
	Добавить устройство	
	Вставить устройство	1 Имя: Modbus Serial Device
	Откл. устройство	Производитель: 35 - Smart Software Solutions GmbH
	Обновить устройство	Pynna: YCTPOVCTBO Modbus Senal
	ј Редактировать объект	Номер модели: -
	Редактировать объект в	Onucauuou A douico that works as a Modhus Corial standalono slavo
	Изменить I/O-соотнесение	Добавить выбранное устройство как последнего потомка
	Импорт соотнесений из CSV	Modbus_COM
	Экспортировать соотнесения в CSV	<ul> <li>(Можно выбрать другой таргет-узел, пока окно открыто.)</li> </ul>
	Эмуляция	
		Добавить устройство Закрыть

Рисунок 4.8 – Добавление компонента Modbus Serial Device

В настройках компонента на вкладке **Modbus Serial Device** следует указать адрес slave-устройства (1 в соответствии с <u>таблицей 4.1</u>).

Modbus_Serial_Device	×	
Modbus Serial Device		
Modbus Serial Device	ID:	1
Информация	Сторожевой таймер:	500
	Регистр хранения (%IW):	10
	Входные регистры (%QW):	10

Рисунок 4.9 – Настройки компонента Modbus Serial Device

На вкладке **Modbus Serial Device Соотнесение входов/выходов** следует привязать к регистрам переменные программы в соответствии с <u>таблицей 4.2</u>.



### ПРИМЕЧАНИЕ

Канал Inputs содержит <u>Holding регистры</u>, канал Outputs – <u>Input регистры</u>. *Следует обратить внимание* на порядок WORD для переменных типа REAL.

У параметра Всегда обновлять переменные должно быть выставлено значение Включено 2.

4. Modbus RTU. ПЛК (CDS V3.5) - slave, ПЛК (CDS V2.3) - master

Мodbus Serial Device Найти переменную Фильтр Показать все -									
Modbus Serial Device	Переменная	Соотнесение	ие Канал	Адрес	Тип	Единица	Описание		
Muhomusuus	= *		Inputs	96IW0	ARRAY [09] OF WORD		Регистры време	нного хранения Modbus	
информация	8.9		inputs[0]	96TW0	WORD				
	* Application.PLC_PRG.xVarToSPK		BitO	96D00.0	BOOL				
	- *		BR1	%D00.1	BOOL				
	- 9		Bit2	96D(0.2	BOOL				
			040	-	000				
			• •	•					
	-*		Bit14	96D(1.6	BOOL				
			Bit15	96D(1.7	BOOL				
	* * Application.PLC_PRG.wVarToSPK	~	Inputs[1]	967W1	WORD				
	# * Application.PLC_PRG.rVarToSPK.awModbu		Inputs[2]	%EW2	WORD				
	# * Application.PLC_PRG.rVarToSPK.awModbu	70	Inputs[3]	%IW3	WORD				
	* * Application.PLC_PRG.sVarToSPK.awModbu		Inputs[4]	96TW4	WORD				
	# * Application.PLC_PRG.sVarToSPK.awModbu		Inputs[5]	%7W5	WORD				
	* * Application.PLC_PRG.sVarToSPK.awModbu	7	Inputs[6]	967W6	WORD				
			Inputs[7]	%IW7	WORD				
	a *a		Inputs[8]	96TW8	WORD				
	H 7		Inputs[9]	%IW9	WORD				
	= **		Outputs	%OW0	ARRAY [0.,9] OF WORD		Входные регист	гры Modbus	
	8-14		Outputs[0]	%OW0	WORD				
	Application.PLC_PRG.xVarFromSPK	~	BILO	960X0-0	BOOL				
	-50		Bit1	%OX0.1	BOOL				
	-10		Bit2	%0X0.2	BOOL				
	1		B#12	96/DX1.4	BOOL				
			Bit13	%0X1.5	BOOL				
			B#14	960X1.6	BOOL				
	- 10		Bt15	%0X1.7	BOOL				
	* * Application.PLC_PRG.wVarFromSPK	3	Outputs[1]	960W1	WORD				
	* * Application.PLC_PRG.rVarFromSPK.awMod		Outputs[2]	960W2	WORD				
	Application_PLC_PRG_rVarFromSPK.awMod		Outputs[3]	%0W3	WORD				
	* * Application.PLC_PRG.sVarFromSPK.awMod		Outputs[4]	%GW4	WORD				
	* * Application_PLC_PRG_sVarFromSPK.awMod		Outputs[5]	960WS	WORD				
	* * Application PLC_PRG sVarFromSPK awMod		Outputs[6]	940446	WORD				
	* *		Outputs[7]	960W7	WORD				
			Outputs[8]	960W8	WORD				
			Outouts[9]	960049	WORD				
			- advantal	indiana.	Los alter.				

Рисунок 4.10 – Привязка переменных к регистрам slave-устройства

Настройка СПК (slave) завершена.

### 4.3 Настройка ПЛК (master)

Для настройки ПЛК (master) следует:

1. Создать новый проект CoDeSys V2.3 для ПЛК110 с программой PLC\_PRG на языке CFC. Нажать ПКМ на название контроллера (в данном примере – PLC110\_30) и добавить подэлемент Modbus (Master):



Рисунок 4.11 – Конфигурация ПЛК. Добавление Modbus (Master)

Данный элемент не нуждается в настройках.

2. Выбрать порт ПЛК, который будет использоваться для связи с СПК. Для этого в элементе Modbus (Master) следует нажать ПКМ на порт Debug RS-232 и в контекстном меню выбрать команду Заменить элемент. В данном примере используется порт RS-485-1.

PLC110_30  Sum PLC110_30  Sum Past discrete inputs[SLOT]  Bum Discrete inputs - 16[E[X]			Базол	вые параметры 🛛 Пар	аметры модуля
Borner in puts - rojrikg     Borner fast discrete outputs[SLOT]     Borner fast discrete outputs - 8[FIX]     Borner fast discrete output[FIX]     Borner	(* Last address *) [CHANNEL (Q)] * Last error *) [CHANNEL (Q)]			Идент, модул Идент, уз, Адрес входл Адрес выходл Адрес диагностин	ия: 203 па: 0 ов: %186.0 ов: %086.0
RS-232[SLOT]	Вставить Universal Modbus device Добавить Подэлемент			Комментар	ий:
	Заменить элемент	•	Debu	ıg RS-232	
	Вычислять адреса		RS-48	85-1	
	Вырезать	Ctrl+X	RS-23	32	
	Ctrl+C	Mode	em		
Вставить Ctr			RS-48	85-2	
	Удалить	Del			

Рисунок 4.12 – Конфигурация ПЛК. Выбор порта

Настройки порта по умолчанию соответствуют тем настройкам, которые были заданы СПК (согласно <u>таблице 4.1</u>): скорость – **115200**, бит данных – **8**, стоп бит – **1**, контроль четности – **нет**. Для параметра **Frame Oriented** следует выбрать значение **RTU**.



Рисунок 4.13 – Конфигурация ПЛК. Настройки порта

3. Нажать ПКМ на элемент Modbus (Master) и добавить два подэлемента Universal Modbus Device:

 Вставить Элемент	Вставить Элемент 🕨					
Добавить Universal Modbu	s device					
Заменить элемент		L				
Вычислять адреса		L				
Вырезать	Ctrl+X	L				
Копировать	Ctrl+C	L				
Вставить	Ctrl+V	L				
Удалить	Del	L				

Рисунок 4.14 – Конфигурация ПЛК. Добавление Universal Modbus Device

Один из подэлементов будет использоваться для чтения значений из СПК, второй – для записи.

⊡······ModBus (Master)[VAR]	
AT %QD6.0: DWORD; (* Last address *) [CHANNEL (Q)]	
AT %QW6.1: WORD; (* Last error *) [CHANNEL (Q)]	
RS-485-1[SLOT]	
🖽 Universal Modbus device (* Чтение из СПК *) [VAR]	
Universal Modbus device (* Запись в СПК *) [VAR]	

Рисунок 4.15 – Внешний вид Конфигурации ПЛК после добавления двух Universal Modbus Device

В настройках элемента **Чтение из СПК** следует указать тип связи (**Serial**), **Slave ID** СПК (в соответствии с <u>таблице 4.1</u> он равен **1**) и режим опроса (**By poll time**, т. е. циклически).

Базон	вые парам	етры Парамет	пры модуля		
	Индекс	Имя	Значение	По умолч.	M
	1	Name	Universal Modbus o	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	

Рисунок 4.16 – Настройки Universal Modbus Device (Чтение из СПК)

В настройках элемента **Запись в СПК** следует указать тип связи (**Serial**), **Slave ID** СПК (в соответствии с <u>таблицей 4.1</u> он равен 1) и режим опроса (**By value change**, т. е. спорадически).

Базов	вые параме	етры Парамет	ры модуля			
	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	

Рисунок 4.17 – Настройки Universal Modbus Device (Запись в СПК)

Нажать ПКМ на элемент Universal Modbus Device (Чтение из СПК) и добавить в него подэлементы Register Input Module, Real Input Module и String Input Module.



Рисунок 4.18 – Universal Modbus Device (Чтение из СПК) с добавленными Input модулями

Затем следует привязать к каждому из каналов переменную (после ввода ее имени она автоматически будет добавлена в список глобальных переменных проекта). Для ввода имени переменной следует два раза нажать на **AT**.



Рисунок 4.19 – Привязка переменных к каналам

Настройки модулей (используемые регистры СПК в соответствии с <u>таблицей 4.1</u> и функции Modbus) приведены ниже.



### ПРИМЕЧАНИЕ

Во время работы с переменными, занимающими несколько регистров СПК (тип **REAL** и **STRING**), указывается только первый из группы регистров.

Базовые параметры Параме		етры Парамет	ры модуля	
	Индекс	Имя	Значение	По умолч.
	1 2 3 8	Name Register address Command Visibility	8 bit input module 0 Read inputs status (0x02) No	8 bit input module 0 Read inputs status (0x02) No

Рисунок 4.20 – Параметры 8 bit Input Module

Базовые параметры Параметры		етры Параметр	ны модуля	
	Индекс	Имя	Значение	По умолч.
	1	Name	Register input module	Register input module
	2	Register Address	1	0
	3	Command	Read input registers (0x04)	Read holding Registers (0x03)
	8	Visibility	No	No

Рисунок 4.21 – Параметры Register Input Module

Базовые параметры Параметры модуля				
Индекс Имя Значение По умолч.				
1 Name float input module float input module	!			
2 Register Address 2 0				
3 Command Read input registers (0x04) 🖃 Read holding Reg	gisters (0x03)			
8 Visibility No 🔽 No				

Рисунок 4.22 – Параметры Real Input Module

Базо	Базовые параметры Па		ры модуля	
	Индекс	Имя	Значение	По умолч.
	1	Name	String input module	String input module
	2	Command	Read input registers (0x04)	Read bytes (0x70)
	3	Register address	4	0
	4	Amounth byte	6	80
	8	Visibility	No	Vo

Рисунок 4.23 – Параметры String Input Module

Затем следует нажать ПКМ на элемент Universal Modbus Device (Запись в СПК) и добавить в него подэлементы Register Output Module, Real Output Module и String Output Module.



Рисунок 4.24 – Universal Modbus Device (Запись в СПК) с добавленными Output модулями

Следует привязать к каждому из каналов переменную (после ввода ее имени она автоматически будет создана в проекте как глобальная). Для ввода имени переменной следует два раза нажать на **AT**.



Рисунок 4.25 – Привязка переменных к каналам

Настройки модулей (используемые регистры СПК в соответствии с <u>таблицей 4.1</u> и функции Modbus) приведены ниже.

Во время работы с переменными, занимающими несколько регистров СПК (тип **REAL** и **STRING**), указывается только первый из группы регистров.

Базовые параметры Параметры		тры Параметры	ы модуля	
	Индекс	Имя	Значение	По умолч.
	1	Name	8 bit output module	8 bit output module
	2	Register Address	0	0
	3	Command	Force multiply coils (0x0f)	Force multiply coils (0x0f)
	8	Visibility	No	▼ No

Рисунок 4.26 – Параметры 8 Bit Output Module

Базов	вые параме	тры Параметры	модуля	
	14	14	2	<b>D</b> = 10 = 10
	индекс	ИМЯ	значение	По умолч.
	1	Name	Register	Register
	2	Register Address	1	0
	3	Command	Preset singl register (0x06) 💌	Preset singl register (0x06)
	8	Visibility	No	No
		-	_	-

Рисунок 4.27 – Параметры Register Output Module

Базовые параметры Параметр		етры Параметр	ы модуля	
	Индекс	Имя	Значение	По умолч.
	1	Name	float output module	float output module
	2	Register Address	2	0
	3	Command	Preset multiple Registers (0x10) 토	Preset multiple Registers (0x10)
	8	Visibility	No	No
			-	

Рисунок 4.28 – Параметры Real Output Module

Базовые параметры Параметры		етры Параметры	і модуля	
	Индекс	Имя	Значение	По умолч.
	1 2 3 4 8	Name Command Register address Amounth byte Visibility	String output module Preset multiple Registers (0x10) [ 4 6 No [	String output module Preset singl register (0x06) 0 80 No

Рисунок 4.29 – Параметры String Output Module

4. Программа PLC\_PRG будет выглядеть следующим образом:

0001 PROGRAM PLC_PRG 0002 VAR 0003 0004 END_VAR 0005	
Чтение данных из СПК	Запись данных в СПК
Изменяйте значения в СПК и наблюдайте за изменением значений переменных	Изменяйте значения переменных и наблюдайте за изменением значений в СПК
xVarFromSPK	xVarToSPK -
wVarFromSPK -	wVarToSPK -
rVarFromSPK	rVarToSPK -
sVarFromSPK	sVarToSPK -

Рисунок 4.30 – Код программы PLC\_PRG

Настройка ПЛК (master) завершена.



### ПРИМЕЧАНИЕ

Проект не содержит операций и используется только для отображения и ввода значений. Пользователь должен создать программу для реализации необходимых алгоритмов.

#### 4.4 Работа с примером

Проекты следует загрузить в оба устройства и запустить.

В случае изменения значения ТоЅРК переменных в ПЛК, соответствующие изменения будут отображаться в программе СПК:



Рисунок 4.31 – ПЛК записывает данные в СПК

В случае изменения значения **FromSPK** переменных СПК, соответствующие изменения будут отображаться в программе ПЛК.

ПЛК считывает данные из СПК	ПЛК записывает данные в СПК
Изменяйте значения переменных и	Изменяйте значения в ППК и
наблюдайте за изменением значений в ПЛК	наблюдайте за изменением значений переменных
xVarFromSPK —	xVarToSPK —
22	11
wVarFromSPK	wVarToSPK
rVarFromSPK.rRealValue	rVarToSPK.rRealValue
v/orEromSPK oString/Joluo	о\/arToSPK aString\/alua
svarromork.soungvalue	svari oserk.ssungvalde
Чтение данных из СПК	Запись данных в СПК
Изменяйте значения в СПК и наблюдайте	Изменяйте значения переменных и наблюдайте
за изменением значений переменных	за изменением значений в СПК
v) /or From SDI/	v)/orToCDI/
xvarromser	XVariospin -
wVarFromSPK=33	wVarToSPK=11
rVarFromSPK=33.3	rVarToSPK=11.1
sVarEromSDK='hello'	s\/arToSDK='spuper'
Svan fomor (- neilo	

Рисунок 4.32 – ПЛК считывает данные из СПК

## 5 Modbus TCP. ПЛК (CDS V3.5) – master, ПЛК (CDS V2.3) – slave

### 5.1 Описание примера

Пример посвящен настройке обмена данными между сенсорным панельным контроллером СПК1хх [M01] и контроллером ПЛК110 [M02] по протоколу Modbus TCP. В этом примере СПК выполняет функцию Master, ПЛК – Slave.

Основные характеристики используемых устройств приведены в таблице 5.1. Используемые в примере переменные описаны в таблице 5.2.

Устройство	СПК1хх [М01] ПЛК110 [М02]		
Функция	Master	Slave	
IP адрес	10.2.11.20	10.2.11.10	
Маска подсети	255.255.0.0		
Порт ТСР	502		
Slave ID	_	1	
Таргет	3.5.11.50 (2)	PLC110.30-М v2 (версия 3.11)	
Версия прошивки	1.0.0814.1125	0.3.72	
Среда разработки проекта	CODESYS V3.5 SP11 Patch 5	CoDeSys V2.3.9.41	
Название файла проекта	ModbusTCPmaster.projectarchive	ModbusTCPslave.pro	

### Таблица 5.1 – Характеристики устройств

### Таблица 5.2 – Список переменных

СПК1xx [М01	] (Master)		ПЛК110 [M02] (Sla		
Переменные, в которые считываются значения из Slave	Переменные, значения которых записываются в Slave	Тип данных	Переменные ПЛК	Адрес регистра/бита	
xVarRead	xVarWrite	BOOL	xVar	0/0	
wVarRead	wVarWrite	WORD	wVar	1	
rVarRead	rVarWrite	REAL	rVar	2–3	
sVarRead	sVarWrite	STRING(6)	sVar	4–6	

Проекты примера доступны для скачивания: Example\_SpkModbusTcpMaster.zip

### 5.2 Настройка ПЛК (slave)

Для настройки ПЛК (slave) следует:

- 1. Создать новый проект CoDeSys V2.3 для ПЛК110 с программой PLC\_PRG на языке CFC.
- **2.** В компоненте **Конфигурация ПЛК** (вкладка **Ресурсы**) настроить регистры Modbus и привязать к ним переменные.

Нажать ПКМ на название контроллера (в примере – PLC110\_30) и добавить подэлемент Modbus (Slave):

😓 CoDeSys - ModbusRTUslave.pro -	[Конфигурация ПЛК]	and the second			
🎹 Файл Правка Проект Вста	вка Дополнения Онлай	йн Окно Справка			
` <b>```````````````````````````````````</b>					
Pecypcы     Profamble переменные     Profamble переменные     Profamble Configuration (V)     Profamble Configuration (V	PLC110_30  Fast discr  Fast d	Вставить Элемент Добавить Подэлемент Заменить элемент Вычислять адреса Вырезать Копировать Вставить Удалить	Ctrl+X Ctrl+C Ctrl+V Del	астройки Параметры модуля Statistic Extended setings Universal network module ModBus (Master) DCON (Master) OWEN (slave) OWEN (slave) Owen (Master) ModBus (slave) Archiver	

Рисунок 5.1 – Конфигурация ПЛК. Добавление Modbus (Slave)

В его настройках следует выбрать адрес (Slave ID), равный 1 (в соответствии с таблицей 5.1):

Базов	зые парам	етры Пар	оаметры модуля		
	Индекс	Имя	Значение	По умолч.	
	1	Name	ModBus (slave)	ModBus (slave)	
	2	Address	1	1	

Рисунок 5.2 – Конфигурация ПЛК. Настройка Modbus (Slave)

**3.** Выбрать порт ПЛК, который будет использоваться для связи с СПК. Для этого следует нажать **ПКМ** на элемент **Modbus (FIX)** и добавим подэлемент **TCP**.

TCPIVARI	Вставить Элемент		
	Добавить Подэлемент	•	Debug RS-232
	Заменить элемент Вычислять адреса		RS-485-1
			RS-232
	Вырезать	Ctrl+X	RS-485-2
	Копировать	Ctrl+C	TCP
	Вставить	Ctrl+V	Modem
	Удалить	Del	

Рисунок 5.3 – Конфигурация ПЛК. Добавление подэлемента ТСР

В параметрах подэлемента следует указать используемый TCP порт – в соответствии с <u>таблицей 5.1</u> это порт **502**.

Базон	вые парам	етры Парамет	ры модуля	
	Индекс	Имя	Значение	По умолч.
	1 2 3	Name RemotePort Visibility	TCP 502 No	TCP 502 No

Рисунок 5.4 – Конфигурация ПЛК. Настройки подэлемента ТСР

- 4. Нажать ПКМ на элемент Modbus (Slave) и добавить следующие подэлементы:
  - 8 bits (для BOOL);
  - 8 bits (для обеспечения выравнивания памяти);
  - 2 byte (для WORD);
  - Float (для REAL);
  - 3 элемента 2 byte (для STRING из 6 символов).



Рисунок 5.5 – Конфигурация ПЛК. Добавление подэлементов

В результате **Конфигурация ПЛК** будет выглядеть следующим образом (см. рисунок 5.6). Затем следует объявить переменные (после ввода имен они автоматически будут добавлены в список глобальных переменных проекта). Для ввода имени переменной следует два раза нажать на **AT**.



### ПРИМЕЧАНИЕ

К регистрам 4–6 не привязывается никаких переменных. Соответствующая им переменная **sVar** будет объявлена в программе **PLC\_PRG** (в пп. 5).



Рисунок 5.6 – Внешний вид Modbus (Slave) с добавленными подэлементами

### ПРИМЕЧАНИЕ

Нумерация регистров в среде CODESYS всегда начинается с нуля, каждый регистр физически занимает два байта (16 бит). Поэтому переменная типа **REAL** займет два регистра (с адресами 2 и 3). Переменная типа **STRING**, которой соответствует три **2 byte** элемента, займет регистры с адресами 4–6, что следует учитывать при настройке masterустройства.

Подробнее вопросы адресации рассмотрены в Руководстве пользователя ПЛК.

### 5. Программа PLC\_PRG будет выглядеть следующим образом: 0001 PROGRAM PLC\_PRG 0002 VAR 0003 sVar AT %QW6.5.0: STRING(6); (\*собираем STRING переменную из трех WORD [т.е. шести символов], 0004 указывая адрес [см. Конфигурация ПЛК] первого из них\*) 0005 END\_VAR 0006 0007 0008 0000 • СПК считывает/записывает значения из ПЛК Изменяйте значения переменных и наблюдайте соответствующие изменения в СПК Изменяйте значения в СПК и наблюдайте за изменением переменных xVar wVar rVar sVar

Рисунок 5.7 – Код программы PLC\_PRG

Настройка ПЛК (slave) завершена.



### ПРИМЕЧАНИЕ

Проект не содержит операций и используется только для отображения и ввода значений. Пользователь должен создать программу для реализации необходимых алгоритмов.

### 5.3 Настройка СПК (master)

Для настройки СПК (master) следует:

- 1. Создать новый проект CODESYS V3.5 для СПК1хх [M01] с программой PLC\_PRG на языке CFC.
- **2.** Добавить в проект <u>объединение</u> с именем **Real\_Word**:



Рисунок 5.8 – Добавление в проект объединения

В объединении следует объявить переменную **rRealValue** типа **REAL** и массив **awModbusReal** типа **WORD**, содержащий два элемента:



Рисунок 5.9 – Объявление переменных объединения

3. Добавить в проект объединение с именем String\_Word.

В объединении следует объявить переменную sStringValue типа STRING и массив awModbusString типа WORD, содержащий три элемента (STRING сможет содержать до 6 символов, поскольку каждый WORD может содержать два символа):

◆     String_Word ×							
1		TYPE String_Word :					
₿ 2		UNION					
3		<pre>sStringValue: STRING(6);</pre>					
4		awModbusString: ARRAY [02] OF WORD;					
5	i	END_UNION					
6	5	END_TYPE					

Рисунок 5.10 – Объявление переменных объединения

**4.** Объявить в программе **PLC\_PRG** девять переменных – 4 из них будут использоваться для отображения данных, считанных из ПЛК, еще 4 – для ввода данных, которые будут записаны в ПЛК. Последняя переменная будет являться триггером записи.

1	PLC	_PRG X	
	1	PROGRAM PLC_PRG	
-	2	VAR	
	3	(*данные, из	считываемые ПЛК*)
	4	xVarRead:	BOOL;
	5	wVarRead:	WORD;
	6	rVarRead:	Real_Word;
	7	sVarRead:	String_Word;
	8		
	9	(*данные, зап	исываемые в ПЛК*)
1	.0	xVarWrite:	BOOL;
1	.1	wVarWrite:	WORD;
1	2	rVarWrite:	Real_Word;
1	.3	sVarWrite:	String_Word;
1	.4		
1	.5	xTrigger:	BOOL; // триггер записи
1	.6	END_VAR	

Рисунок 5.11 – Объявление переменных программы

5. Код программы будет выглядеть следующим образом:

Чтение данных из ПЛК Изменяйте значения в ПЛК и наблюдайте за изменением значений переменных	Запить данных в ПЛК Изменяйте значения переменных, активируйте триггер записи и наблюдайте за изменением значений в ПЛК
xVarRead -	xVarWrite
wVarRead	wVarWrite
rVarRead.rRealValue	rVarWrite.rRealValue
sVarRead.sStringValue	sVarWrite.sStringValue
	xTrigger –

Рисунок 5.12 – Код программы на языке CFC

6. Добавить в проект компонент Ethernet.



### ПРИМЕЧАНИЕ

Версия компонента не должна превышать версию таргет-файла контроллера. Подробнее см. в документе **CODESYS V3.5. Описание таргет-файлов**.

### 5. Modbus TCP. ПЛК (CDS V3.5) - master, ПЛК (CDS V2.3) - slave

Устройства	<b>▼</b> ₽ :	<	
ModbusTCPmaster		-	
Device (SPK1xx[M01])			doornal Action 2
Pic Logic	ð	вырезать	Mwa: Ethernet
Application		копировать	
- * Real_Word		Вставить	Деиствие
tring_Word	^	здалить	Добавить устройство     Обновить устройство     Обновить устройство     Обновить устройство     Обновить устройство     Обновить устройство
👘 Менеджер библиотек		06300	
PLC_PRG (PRG)	6	Свойства	Введите строку для полнотекстового поиска во е Производитель: «Al vendors» ~
😑 🌃 Конфигурация задач	<b>*</b>	Добавление объекта	Имя Производитель Версия Описание:
😑 🥩 MainTask		Добавить папку	
- @] PLC_PRG		Добавить устройство	🖶 👘 Промышленные сети (fieldbus)
U RTC (RTC)		Обновить устройство	В АБ-ИНТЕРФЕЙС
- CovenCloud (OwenCloud)	Ľ	Редактировать объект	CAN CANbus
49 Buzzer (Buzzer)		Редактировать объект в	🕴 🔐 🔐 Ethercat
Drives (Drives)		Изменить І/О-соотнесение	Ethernet-aganrep
る Network (Network)		Импорт соотнесений из CSV	Ethernet 35 - Smart Software Solutions GmbH 3.5.11.0 Ethernet Link.
Screen (Screen)		Экспортировать соотнесения в CSV	
Debug (Debug)	1	Режим онлайн-конфигурации	
		Сброс заводской устройства [Device]	
		Эмуляция	Группировать по категориям Отображать все версии (для экспертов) Показать устаревшие версии
			Image: Ethernet:         Прокаводители: 35 - Smart Software Solutions GmbH         Прокаводители: 35 - Smart Software Solutions GmbH         Предокав: 55.1.0         Номер модели: -         Описание: Ethernet Link.
			Добавить выбранное устройство как последнего потомка           Device         (Можно выбрать другой таргет-узел, пока окно открыто.)           Добавить устройство         Закрыть

Рисунок 5.13 – Добавление компонента Ethernet

В конфигурации Ethernet следует указать сетевые настройки в соответствии с таблицей 5.1.

Ethernet ×			
Конфигурация ethernet Состояние Ethernet Device Соотнесение входов/выход Информация	Интерфейс: eth0 IP-адрес Маска подсети Gateway по умолчанию Adjust Operating Syst	10       2       .       11       .       20         255       .       255       .       0       .       0         10       .       2       .       1       .       1         rem Settings	]

Рисунок 5.14 – Настройки компонента Ethernet

7. В компонент Ethernet следует добавить компонент Modbus TCP Master.



### ПРИМЕЧАНИЕ

Версия компонента не должна превышать версию таргет-файла контроллера. Подробнее см. в документе **CODESYS V3.5. Описание таргет-файлов**.

= 🄄 ModbusTCPmaster	•	
Device (SPK1xx[M01])		П Дооавить устроиство Х
Pic Logic		Hum Modhur TCR Matter
Application		VIMA. Produce_ICF_Praster
*# Real_Word		Действие
*# String_Word		Добавить устройство Вставить устройство Подключить устройство Обновить устройство
👘 Менеджер библиотек		
PLC_PRG (PRG)		Введите строку для полнотекстового поиска во є Производитель: <ai vendors=""> ~</ai>
😑 🗱 Конфигурация задач		Имя Производитель Версия Описани ^
= 🍪 MainTask		Промышленные сети (fieldbus)
型 PLC_PRG		* Ethernet/IP
Ethernet (Ethernet)	V Rupparti	e • Modbus
- 🕚 RTC (RTC)		🗘 💷 Macrep Modbus TCP
- 2 OwenCloud (OwenCloud)	Вставить	Modbus TCP Master 35 - Smart Software Solutions GmbH 3.5.11.30 A device tl 🗸
- 49 Buzzer (Buzzer)	Х Улалить	< >> >>
Drives (Drives)	Ofeon	
品 Network (Network)	Determine v	
- Q: Screen (Screen)	Рефакторині	
Debug (Debug)	Свойства	У Группировать по категориям Отображать все версии (для экспертов) Показать устаревшие версии
一出象 Info (Info)	Добавление объекта	
	Добавить папку	Hing: Modhus TCP Master
	Добавить устройство	Производитель: 3S - Smart Software Solutions GmbH
	Вставить устройство	Fpynnis: Macrep Modbus TCP
	Откл. устройство	Номер модели: -
	Обновить устройство	Описание: A device that works as a Modbus Master on Ethernet.
	Редактировать объект	
	Редактировать объект в	Добавить выбранное устройство как последнего потомка
	Изменить I/О-соотнесение	Ethernet
	Импорт соотнесений из CSV	<ul> <li>(Можно выбрать другой таргет-узел, пока окно открыто.)</li> </ul>
	экспортировать соотнесения в CSV	
	Эмуляция	Добавить устройство Закрыть

Рисунок 5.15 – Добавление компонента Modbus TCP Master

В настройках компонента следует поставить галочку Автоподключение.

Modbus_TCP_Master ×			
Общее	Modbus-TCP		MODBIIS
ModbusTCPMaster Соотнесение входов/выход	Таймаут ответа (мс)	1000	MUDDUJ
ModbusTCPMaster Конфигурация	Таймаут сокета (мс)	10	
Состояние	🖂 автоподключение		
Информация			

Рисунок 5.16 – Настройка компонентов Modbus TCP Master

8. B Modbus TCP Master добавить компонент Modbus TCP Slave.



### ПРИМЕЧАНИЕ

Версия компонента не должна превышать версию таргет-файла контроллера. Подробнее см. в документе **CODESYS V3.5. Описание таргет-файлов**.



Рисунок 5.17 – Добавление компонента Modbus TCP Slave в проект

В настройках компонента на вкладке **Общее** следует указать IP-адрес, Unit ID и порт slave-устройства в соответствии с <u>таблицей 5.1</u>.

Modbus_TCP_Slave ×			
Общее Канал Modbus Slave Modbus Slave Init ModbusTCPSlave Конфигурация ModbusTCPSlave Соотнесение входов/выход Состояние Информация	Modbus-TCP IP-адрес слейва: Unit-ID [1247] Таймаут ответа (мс) Порт	10       2       . 11       . 10         1       .       .       .         1000       .       .       .         502       .       .       .	MODBUS

Рисунок 5.18 – Настройки компонента Modbus Slave в проект

Во вкладке **Канал Modbus Slave** следует создать 8 каналов – 4 из них будут использоваться для чтения переменных, 4 – для записи. Чтение будет осуществляться циклически, запись – по переднему фронту триггера (**RISING\_EDGE**). Используемые функции соответствуют типам данных, адреса регистров настроены согласно <u>таблице 5.2</u>.

5. Modbus TCP. ПЛК (CDS V3.5) - master, ПЛК (CDS V2.3) - slave

Канал Modbus Slave 0 1	Channel 0	Read Coils (Код функции 01)	Цикл., t#100ms	16#0000	1	C		
- 1	Channel 1				1	сохранить посл. значение		
	. Channel I	Read Holding Registers (Код функции 03)	Цикл., t#100ms	16#0001	1	Сохранить посл. значение		
lodbus Slave Init 2	Channel 2	Read Holding Registers (Код функции 03)	Цикл., t#100ms	16#0002	2	Сохранить посл. значение		
IodbusTCPSlave 3	Channel 3	Read Holding Registers (Код функции 03)	Цикл., t#100ms	16#0004	3	Сохранить посл. значение		
IodbusTCPSlave 4	Channel 4	Write Single Coil (Код функции 05)	Передний фронт				16#0000	1
оотнесение входов/выход 5	6 Channel 5	Write Single Register (Код функции 06)	Передний фронт				16#0001	1
остояние 6	6 Channel 6	Write Multiple Registers (Код функции 16)	Передний фронт				16#0002	2
7	Channel 7	Write Multiple Registers (Код функции 16)	Передний фронт				16#0004	3

Рисунок 5.19 – Настройка каналов Modbus Slave

Во вкладке **ModbusTCPSIave Соотнесение входов/выходов** следует привязать к каналам переменные программы в соответствии с <u>таблицей 5.2</u>. У параметра Всегда обновлять переменные должно быть выставлено значение Включено 2.

Han Modbus Slave	Переменная	6					
dbus Slave Init	- **	Соотнесение	Канал	Адрес	Тип	Единица	Описание
dbusTCPSlave			Channel 0	%IB0	AKRAY [00] OF BYTE		Buildet
doubler blave		2.	Channel U[U]	%IB0	BYIE		Read Colls
нфигурация	Application.PLC_PRG.xVarRead	•	Bitu Channel 1	%LX0.0	BOOL		БИТ U Dead Helding Desisters
отнесение входов/выход			Channel 1	%IVV1	AKRAY [UU] OF WORD		Read Holding Registers
стояние	Application.PLC_PRG.wVarRead	•	Channel 1[U]	%IV/1	WORD		Регистр I Dead Halding Desistant
		2.	Channel 2	%IW2	ARRAY [01] OF WORD		Read Holding Registers
формация	Application.PLC_PRG.rVarRead.awModbusReal[0]	2	Channel 2[0]	%IW2	WORD		Регистр 2
	Application.PLC_PRG.rVarRead.awModbusReal[1]		Channel 2[1]	%IW3	WORD		Регистр 3
	- 7 ·		Channel 3	%IW4	ARRAY [02] OF WORD		Read Holding Registers
	Application.PLC_PRG.sVarRead.awModbusString[0]		Channel 3[0]	<del>%IW4</del>	WORD		Регистр 4
	Provide the second s	•	Channel 3[1]	<del>%IW5</del>	WORD		Регистр 5
	Image: Application.PLC_PRG.sVarRead.awModbusString[2]	٠	Channel 3[2]	%IW6	WORD		Регистр 6
	Application.PLC_PRG.xTrigger	٠,	Channel 4	%QX0.0	BIT		Trigger Variable
5	- <b>•</b>		Channel 4	%QB1	ARRAY [00] OF BYTE		
	8- •		Channel 4[0]	%QB1	BYTE		Write Single Coil
	Application.PLC_PRG.xVarWrite	٠,	BitO	%QX1.0	BOOL		Бит 0
	Application.PLC_PRG.xTrigger	٠,	Channel 5	%QX2.0	BIT		Trigger Variable
	÷-**		Channel 5	%QW2	ARRAY [00] OF WORD		Write Single Register
	Application.PLC_PRG.wVarWrite	٠	Channel 5[0]	%QW2	WORD		Регистр 1
	Application.PLC_PRG.xTrigger	۰,	Channel 6	<del>%QX6.0</del>	BIT		Trigger Variable
e	÷**		Channel 6	%QW4	ARRAY [01] OF WORD		Write Multiple Registers
	#-* Application.PLC_PRG.rVarWrite.awModbusReal[0]	٠,	Channel 6[0]	%Q₩4	WORD		Регистр 2
	Application.PLC_PRG.rVarWrite.awModbusReal[1]	٠,	Channel 6[1]	%QW5	WORD		Регистр 3
	Application.PLC_PRG.xTrigger	٠,	Channel 7	%QX12.0	BIT		Trigger Variable
6	à. <b>~</b> ≱		Channel 7	%QW7	ARRAY [02] OF WORD		Write Multiple Registers
	Application.PLC_PRG.sVarWrite.awModbusString[0]	۰,	Channel 7[0]	%QW7	WORD		Регистр 4
	Application.PLC_PRG.sVarWrite.awModbusString[1]	۰,	Channel 7[1]	%QW8	WORD		Регистр 5
	Application.PLC_PRG.sVarWrite.awModbusString[2]	۰,	Channel 7[2]	%QW9	WORD		Регистр 6
		Сбро	соотнесения	Всегда обновлят	гь переменные: Вкл. 2 (	всегда в зада	че цикла шины)
N	МЭК-объекты						
1	Переменная Соотнесение Тип — Ф Modbus_TCP_Slave 🏠 ModbusTCP	PSlave					

Рисунок 5.20 – Привязка переменных к каналам



### ПРИМЕЧАНИЕ

Проект не содержит операций и используется только для отображения и ввода значений. Пользователь должен создать программу для реализации необходимых алгоритмов.

Настройка СПК (master) завершена.

### 5.4 Работа с примером

Проекты следует загрузить в оба устройства и запустить их.

В случае изменения значения переменных в ПЛК, соответствующие изменения будут отображаться в программе СПК:



Рисунок 5.21 – СПК считывает данные из ПЛК

Затем следует сохранить изменения **write** переменных СПК и активировать триггер записи. Соответствующие изменения будут отображаться в программе ПЛК. Также новые значения будут считаны в **read** переменные программы СПК.



Рисунок 5.22 – СПК записывает данные в ПЛК

## 6 Modbus TCP. ПЛК (CDS V3.5) – slave,ПЛК (CDS V2.3) – master

### 6.1 Описание примера

Пример посвящен настройке обмена данными между сенсорным панельным контроллером **СПК1хх** [01] и контроллером **ПЛК110** [M02] по протоколу **Modbus TCP**. В примере СПК выполняет функцию **Slave**, ПЛК – **Master**.

Основные характеристики используемых устройств приведены в таблице 6.1. Используемые в примере переменные описаны в таблице 6.2.

Устройство	СПК1хх [М01]	ПЛК110 [М02]
Функция	Slave	Master
IP адрес	10.2.11.20	10.2.11.10
Маска подсети	2	55.255.0.0
Порт ТСР		502
Slave ID	1	_
Таргет	3.5.11.50 (2)	PLC110.30-М v2 (версия 3.11)
Версия прошивки	1.0.0814.1125	0.3.72
Среда разработки проекта	CODESYS V3.5 SP11 Patch 5	CoDeSys V2.3.9.41
Название файла проекта	ModbusTCPslave.projectarchive	ModbusTCPmaster.pro

### Таблица 6.1 – Характеристики устройств

### Таблица 6.2 – Список переменных

СПК1хх [М	101] (Slave)		ПЛК110 [M02] (Master)
Переменные, которые считывает Master	Адрес регистра/бита	Тип данных	Переменные ПЛК
xVarFromSPK	0/0 (Discrete Inputs)	BOOL	xVarFromSPK
wVarFromSPK	1 (Input Registers)	WORD	wVarFromSPK
rVarFromSPK	2–3 (Input Registers)	REAL	rVarFromSPK
sVarFromSPK	4–5 (Input Registers)	STRING(6)	sVarFromSPK
Переменные, которые записывает Master	Адрес регистра/бита	Тип данных	Переменные ПЛК
xVarToSPK	0/0 (Coils)	BOOL	xVarToSPK
wVarToSPK	1 (Holding Registers)	WORD	wVarToSPK
rVarToSPK	2–3 (Holding Registers)	REAL	rVarToSPK
sVarToSPK	4–6 (Holding Registers)	STRING(6)	sVarToSPK

Проекты примера доступны для скачивания: <u>Example\_SpkModbusTcpSlave.zip</u>

### 6.2 Настройка СПК (slave)

Для настройки СПК (slave) следует:

- 1. Создать новый проект CODESYS V3.5 для СПК1хх [M01] с программой PLC\_PRG на языке CFC.
- 2. Добавить в проект объединение с именем Real\_Word:



Рисунок 6.1 – Добавление в проект объединения

В объединении следует объявить переменную **rRealValue** типа **REAL** и массив **awModbusReal** типа **WORD**, содержащий два элемента:

1	Rea	L_Word ×
	1	TYPE Real_Word :
8	2	UNION
	3	rRealValue: REAL;
	4	awModbusReal: ARRAY [01] OF WORD;
	5	END_UNION
	6	END_TYPE

Рисунок 6.2 – Объявление переменных объединения

3. Добавить в проект объединение с именем String\_Word.

В объединении следует объявить переменную sStringValue типа STRING и массив awModbusString типа WORD, содержащий три элемента (STRING сможет содержать до 6 символов, так как каждый WORD может содержать два символа):

🕏 Str	ing_Word ×
1	TYPE String_Word :
2	UNION
3	<pre>sStringValue: STRING(6);</pre>
4	awModbusString: ARRAY [02] OF WORD;
5	END_UNION
6	END_TYPE

Рисунок 6.3. – Объявление переменных объединения

**4.** Объявить в программе **PLC\_PRG** 8 переменных – 4 из них будут считываться ПЛК, 4 – записываться ПЛК. *Следует обратить внимание* на <u>п. 2.3</u>.

2	
🖉 🚺 PLC	_PRG ×
1	PROGRAM PLC_PRG
■ 2	VAR
3	(*данные, считываемые ПЛК*)
4	xVarFromSPK: BOOL;
5	wVarFromSPK: WORD;
6	rVarFromSPK: Real_Word;
7	sVarFromSPK: String_Word;
8	
9	(*данные, записываемые ПЛК*)
10	xVarToSPK: BOOL;
11	wVarToSPK: WORD;
12	rVarToSPK: Real_Word;
13	sVarToSPK: String_Word;
14	END_VAR

Рисунок 6.4 – Объявление переменных программы

5. Код программы будет выглядеть следующим образом:

ПЛК считывает данные из СПК	ПЛК записывает данные в СПК
Изменяйте значения переменных и наблюдайте за изменением значений в ПЛК	Изменяйте значения в ПЛК и наблюдайте за изменением значений переменных
xVarFromSPK -	xVarToSPK -
wVarFromSPK	wVarToSPK -
rVarFromSPK.rRealValue	rVarToSPK.rRealValue
sVarFromSPK.sStringValue	sVarToSPK.sStringValue

Рисунок 6.5 – Код программы на языке CFC

6. Добавить в проект компонент Ethernet.

# i B

### ПРИМЕЧАНИЕ

Версия компонента не должна превышать версию таргет-файла контроллера. Подробнее см. в документе **CODESYS V3.5. Описание таргет-файлов**.



Рисунок 6.6 – Добавление компонента Ethernet

В конфигурации Ethernet следует указать сетевые настройки в соответствии с таблицей 6.1.

Ethernet ×			
Конфигурация ethernet Состояние Ethernet Device Соотнесение входов/выход Информация	Интерфейс: eth0 IP-адрес Маска подсети Gateway по умолчанию Adjust Operating System	10       2       .       11       .       20         255       .       255       .       0       .       0         10       .       2       .       1       .       1         m Settings       .       .       .       .       .       .	

Рисунок 6.7 – Настройки компонента Ethernet

7. В компонент Ethernet добавить компонент Modbus TCP Slave Device.



### ПРИМЕЧАНИЕ

Версия компонента не должна превышать версию таргет-файла контроллера. Подробнее см. в документе **CODESYS V3.5. Описание таргет-файлов**.



Рисунок 6.8 – Добавление компонента Modbus TCP Slave Device

В настройках компонента на вкладке **Страницу конфигурации** следует указать порт и адрес slaveустройства (**502** и **1** в соответствии с <u>таблицей 6.1</u>).

ModbusTCP_Slave_Dev	ice ×	
Страница конфигурации	Заданные параметры	
Modbus TCP Slave Device Соотнесение входов/выход	Сторожевой таймер:	500 🚖 (MC)
Информация	Slave-порт:	502
	Unit ID:	1
	Регистры временного хранения (%IW):	10
	Входные регистры (%QW):	10

Рисунок 6.9 – Настройки компонента Modbus TCP Slave Device

Во вкладке Modbus Serial Device Соотнесение входов/выходов следует привязать к регистрам переменные программы в соответствии с <u>таблицей 6.2</u>. Следует обратить енимание на порядок WORD для переменных типа REAL.



### ПРИМЕЧАНИЕ

Канал Inputs содержит Holding регистры, канал Outputs – Input регистры.

У параметра Всегда обновлять переменные должно быть выставлено значение Включено 2.



Рисунок 6.10 – Привязка переменных к регистрам slave-устройства

Настройка СПК (slave) завершена.

### 6.3 Настройка ПЛК (master)

Для настройки ПЛК (master) следует:

1. Создать новый проект CoDeSys V2.3 для ПЛК110 с программой PLC\_PRG на языке CFC. Нажать ПКМ на название контроллера (в данном примере – PLC110\_30) и добавить подэлемент Modbus (Master):

🞭 CoDeSys - ModbusRTUmaster.pro* - [K	Сонфигурация ПЛК]	ALCON BALL	1.00	A POINT AND AND AND	1000 01.00	
🌐 Файл Правка Проект Вставка	Дополнения Онлайн Окно Спра	вка				
"∎ 🚅 🖬 📲 🚳 🛹 📲 🚔 🗳						
Pecypcы     Constant Sector Constant Sec	PLC110_30     B—Fast discrete inputs[SLOT]     B—Discrete inputs - 16[FIX]     B—Fast discrete outputs[SLOT     B—Sorete outputs - 8[FIX]     B—Special input[FIX]     B—Special output[FIX]	Вставить Элемент Добавить Подэлемент Заменить элемент Вычислять адреса Вырезать Копировать Вставить Удалить	Ctrl+X Ctrl+C Ctrl+V Del	Hacrpołku Dz     Statistic Extended setings Universal network module ModBus (Master) DCON (Master) OWEN (slave) OWEN (slave) Owen (Master) ModBus (slave) Archiver	араметры модуля ческое вычисление адресов: перекрытия адресов: ъ конф. файлы в проекте:	া র র

Рисунок 6.11 – Конфигурация ПЛК. Добавление Modbus (Master)

Элемент не нуждается в настройках.

2. Нажать ПКМ на элемент Modbus (Master) и добавить два подэлемента Universal Modbus Device:

Вставить Элемент	+		
Добавить Universal Modbus device			
Заменить элемент		L	
Вычислять адреса		L	
Вырезать	Ctrl+X	L	
Копировать	Ctrl+C	L	
Вставить	Ctrl+V	L	
Удалить	Del		

Рисунок 6.12 – Конфигурация ПЛК. Добавление Universal Modbus Device

Один из подэлементов будет использоваться для чтения значений из СПК, второй подэлемент – для записи.

⊡ModBus (Master)[VAR]
AT %QD6.0: DWORD; (* Last address *) [CHANNEL (Q)]
AT %QW6.1: WORD; (* Last error *) [CHANNEL (Q)]
RS-485-1[SLOT]
🗄 Universal Modbus device (* Чтение из СПК *) [VAR]
Universal Modbus device (* Запись в СПК *) [VAR]

Рисунок 6.13 – Внешний вид Конфигурации ПЛК после добавления двух Universal Modbus Device

В настройках элемента **Чтение из СПК** следует указать тип связи (**TCP**), IP-адрес, порт и **Slave ID** СПК (в соответствии с таблицей 6.1) и режим опроса (**By poll time**, т. е. циклически).

Базов	вые параме	етры Парамет	пры модуля		
	Индекс	Имя	Значение	По умолч.	Мин.
		Name	Universal Modbus (	d Universal Modbus d	
	2	ModuleIP	10:2:11:20	10:0:0:223	
	3	Max timeout	150	150	10
	4	TCPport	502	502	
	5	NetMode	TCP	🔽 Serial	
	6	ModuleSlave		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	

Рисунок 6.14 – Настройки Universal Modbus Device (Чтение из СПК)

В настройках элемента **Запись в СПК** следует указать тип связи (**TCP**), IP-адрес, порт и **Slave ID** СПК (в соответствии с таблицей 6.1) и режим опроса (**By value change**, т. е. спорадически).

Базов	Базовые параметры Параметры модуля					
	Индекс	Имя	Значение	По умолч.		
	1	Name	Universal Modbus d.	. Universal Modbus d		
	2	ModuleIP	10:2:11:20	10:0:0:223		
	3	Max timeout	150	150		
	4	TCPport	502	502		
	5	NetMode	TCP	<ul> <li>Serial</li> </ul>		
	6	ModuleSlave	1	_1		
	7	Work mode	By value change	<ul> <li>By poll time</li> </ul>		
	8	Polling time ms	100	_ 100		
	9	Visibility	No	<ul> <li>No</li> </ul>		
	10	Amount Rep	0 _	_0		
	11	Byte Sequen	Trace_mode	<ul> <li>Trace_mode</li> </ul>		

Рисунок 6.15 – Настройки Universal Modbus Device (Запись в СПК)

Затем следует нажать ПКМ на элемент Universal Modbus Device (Чтение из СПК) и добавить в него подэлементы Register Input Module, Real Input Module и String Input Module.



Рисунок 6.16 – Universal Modbus Device (Чтение из СПК) с добавленными Input модулями

К каждому из каналов следует привязать переменную (после ввода ее имени она автоматически будет добавлена в список глобальных переменных проекта). Для ввода имени переменной следует два раза нажать на **AT**.



### Рисунок 6.17 – Привязка переменных к каналам

Настройки модулей (используемые регистры СПК в соответствии с <u>таблицей 6.1</u> и функции Modbus) приведены ниже.



### ПРИМЕЧАНИЕ

Во время работы с переменными, занимающими несколько регистров СПК (тип **REAL** и **STRING**), указывается только первый из группы регистров.

Базовые параметры		етры Г	Параметр	ы модуля			
	Индекс	Имя		Значение		По умолч.	
	1 2	Name Registe	r address	8 bit input module 0		8 bit input module 0	
	3 8	Comma Visibility	nd	Read inputs status No	(0x02)	▼ Read inputs status ▼ No	(0x02)

Рисунок 6.18 – Параметры 8 bit Input Module

Базо	разовые параметры Параметры модуля					
	Индекс	Имя	Значение	По умолч.		
	1 2 3 8	Name Register Address Command Visibility	Register input module 1 Read input registers (0x04) No	Register input module 0 Read holding Registers (0x03) No		

Рисунок 6.19 – Параметры Register Input Module

Базо	вые параме	тры Параметрь	і модуля	
	Индекс	Имя	Значение	По умолч.
	1 2 3 8	Name Register Address Command Visibility	float input module 2 Read input registers (0x04) No	float input module 0 ▼ Read holding Registers (0x03) ▼ No

Рисунок 6.20 – Параметры Real Input Module

Базо	вые парам	етры Параметр	ы модуля	
	Индекс	Имя	Значение	По умолч.
	1	Name	String input module	String input module
	2	Command	Read input registers (0x04)	Read bytes (0x70)
	3	Register address	4	0
	4	Amounth byte	6	80
	8	Visibility	No	Vo

Рисунок 6.21 – Параметры String Input Module

Нажать ПКМ на элемент Universal Modbus Device (Запись в СПК) и добавить в него подэлементы Register Output Module, Real Output Module и String Output Module.

Universal Modbus device (* Запись в СПК *) [VAR]
AT %QB6.2.0: BYTE; (* Command (0xff - Start) *) [CHANNEL (Q)
ш8 bit output module (* запись в coil 0 *) [VAR]
Register output module (* запись в holding register 1 *) [VAR]
——Real output module (* запись в holding register 2-3 *) [VAR]
String Output module (* запись в holding register 4-6 *) [VAR]

Рисунок 6.22 – Universal Modbus Device (Запись в СПК) с добавленными Output модулями

К каждому из каналов следует привязать переменную (после ввода ее имени она автоматически будет создана в проекте как глобальная). Для ввода имени переменной два раза нажмите на **AT**.



Рисунок 6.23 – Привязка переменных к каналам

Настройки модулей (используемые регистры СПК в соответствии с <u>таблицей 6.1</u> и функции Modbus) приведены ниже.



### ПРИМЕЧАНИЕ

Во время работы с переменными, занимающими несколько регистров СПК (тип **REAL** и **STRING**), указывается только первый из группы регистров.

Базовые параметры		тры Параметры	ы модуля	
	Индекс	Има	Зиачение	По имоди
	1	Name	8 bit output module	8 bit output module
	2	Register Address	0	0
	3	Command	Force multiply coils (0x0f)	Force multiply coils (0x0f)
	8	Visibility	No	▼ No

Рисунок 6.24 – Параметры 8 Bit Output Module

Базов	вые параме	тры Параметры	модуля	
	Индекс	Имя	Значение	По умолч.
	1	Name	Register	Register
	2	Register Address	1	0
	3	Command	Preset singl register (0x06)	Preset singl register (0x06)
	8	Visibility	No	No

Рисунок 6.25 – Параметры Register Output Module

Базо	Базовые параметры Параметры модуля					
	14	14	2	<b>D</b>		
	индекс	ИМЯ	значение	По умолч.		
	1	Name	float output module	float output module		
	2	Register Address	2	0		
	3	Command	Preset multiple Registers (0x10)	Preset multiple Registers (0x10)		
	8	Visibility	No	No		

Рисунок 6.26 – Параметры Real Output Module

Базо	вые парам	иетры Парамет	ры модуля	
	Индекс	Имя	Значение	По умолч.
	1	Name	String output module	String output module
	2	Command	Preset multiple Registers (0x10)	Preset singl register (0x06)
	3	Register address	4	0
	4	Amounth byte	6	80
	8	Visibility	No	▼ No

Рисунок 6.27 – Параметры String Output Module

3. Программа PLC\_PRG будет выглядеть следующим образом:

0001 PROGRAM PLC_PRG 0002 VAR 0003 0004 END_VAR 0005 0006 0006 0007 <	
Чтение данных из СПК Изменяйте значения в СПК и наблюдайте	Запись данных в СПК Изменяйте значения переменных и наблюдайте
xVarFromSPK	xVarToSPK
wVarFromSPK	WVarToSPK
sVarFromSPK	sVarToSPK -

Рисунок 6.28 – Код программы PLC\_PRG

# i

### ПРИМЕЧАНИЕ

Проект не содержит операций и используется только для отображения и ввода значений. Пользователь должен создать программу для реализации необходимых алгоритмов.

Настройка ПЛК (master) завершена.

### 6.4 Работа с примером

Проекты следует загрузить в оба устройства и запустить.

В случае изменения значения **ToSPK** переменных в ПЛК, соответствующие изменения будут отображаться в программе СПК:



Рисунок 6.27 – ПЛК записывает данные в СПК

В случае изменения значения **FromSPK** переменных СПК, соответствующие изменения будут отображаться в программе ПЛК.

ПЛК считывает данные из СПК	ПЛК записывает данные в СПК
Изменайте значения переменных и	Изменайте значения в ППК и
наблюдайте за изменением значений в ППК	наблюдайте за изменением значений переменных
x\/arEromSPK	x\/arToSPK
XVan Ionior IX	XVIIIOUIX
22	11
wVarFromSPK	wVarToSPK
r\/arEramSBK rBaal\/alua	r\/arTaSPK_rPaal\/alua
IValFIOIIISFK.IRealValue	I Val I OSF K.I Keal Value
'hello'	'DMPAT'
sVarFromSPK.sStringValue	sVarToSPK.sStringValue
	<b>`</b>
Чтение данных из СПК	Запись данных в СПК
Изменяйте значения в СПК и наблюдайте	Изменяйте значения переменных и наблюдайте
	за измононимом значений в СПК
за изменением значении переменных	за изменением значении в отпа
xVarFromSPK	xVarToSPK
wVarFromSPK=22	wVarToSPK=11
rVarFromSPK=22.2	rVarToSPK=11.1
sVarFromSPK='hello'	sVarToSPK='привет'

Рисунок 6.28 – ПЛК считывает данные из СПК