Описание библиотеки OwenModbuSlave

(Codesys 2.3)

Руководство по применению

Версия: 2.3.9.4

Введение

Библиотека OwenModbusSlave применяется для настройки OBEH ПЛК в режиме Modbus RTU Slave. В отличие от настройки прибора через *Конфигурацию ПЛК* данный способ предоставляет более широкий функционал и является удобным при работе с большим количеством переменных.

1. Установка дополнительных библиотек для проекта

В системе CoDeSys все файлы библиотек функциональных блоков имеют расширения *.lib и находятся в папке Library – она расположена по месту размещения основной программы на диске компьютера (чаще всего в папке C:\Program Files\3S Software\CoDeSys V2.3\Library). Для подключения новых библиотек к проекту сначала соответствующие файлы переписываются пользователем в ту же папку, где находятся все остальные библиотеки.

Чтобы увидеть какие библиотеки уже были раньше подключены к проекту и установить дополнительные, используется Менеджер библиотек – его можно открыть из главного меню CoDeSys командами Окно ► Менеджер библиотек (или в менеджере проекта на закладке «Ресурсы» открывается папка Менеджер библиотек), рис. 1.1 (в окне вверху показан список установленных библиотек).



Рис. 1.1. Окно открытой папки «Менеджер библиотек»

Установка дополнительных библиотек выполняется из главного меню последовательным выбором команд: Вставка > Добавить библиотеку (рис. 1.2), в открывшемся окне папки Library (рис. 1.3) выделяется файл с именем нужной библиотеки и выполняется команда Открыть.

ScoDeSys - (Untitled)*	RERE 2 3-3-11 Villand Viron, Bronn, Bronn, Manner Namer, Salita, Aanten Siller, Carrol print) Barren, Salita, San	
Файл Правка Проект Вставка Доп	толнения Онлайн Окно Справка	
``````````````````````````````````````		
Pecipical     Constraints represense     Goldmarker represense     Goldmarker apparense     Goldmarker application     Goldmarker     Goldmarker application     Goldmarker applic	Image: Control Use and Control	
× m +	Размер кода: 10 байт	-
🖹 POU 🔩 Тип 🖾 Виз 🔛 Рес		4
Добавляет еще одну библиотеку в проект		ОНЛАЙН ЗАМ ПРОСМОТР

Рис. 1.2 – Команда «Добавить библиотеку»

CoDeSys - (Untited)*	
Файл Правка Проект Вставка Дополнения Онлайн Окно Справка	
nai Honardada Xanaga	
Process       Processes         Processes       Processes         Processes	
	Þ
	ИЛАЙН ЗАМ ПРОСМОТР

Рис. 1.3 – Окно для выбора подключаемой дополнительной библиотеки к проекту

Теперь в перечне доступных для проекта библиотек появится установленная библиотека – **OwenModbusSlave_v2.3.9.4.lib**. Для просмотра состава и свойств функциональных блоков, выбирается нужная библиотека – при этом появится папка с блоками (на рис. 1.1 справа дана краткая справочная информация по использованию компонентов).

Подобны образом можно добавлять любую из поддерживаемых библиотек.

### ПРИМЕЧАНИЕ:

- 1. Для функционирования библиотеки **OwenModbusSlave_v2.3.9.4.lib** также необходима дополнительная библиотека **BufferTools_v2.3.9.01.lib**.
- 2. По умолчанию доступен только стандартный набор библиотек. Для каждого нового проекта проводится присоединение новых библиотек индивидуально, по мере необходимости.
- 3. У функциональных блоков, разработанных для контроллеров фирмы OBEH, режим симуляции (Simulation Mode) не предусмотрен. Отладка программы возможна только при подключенном контроллере.

### 2. Описание функциональных блоков библиотеки

Библиотека содержит только один пользовательский функциональный блок – MB_RTU_SLAVE.

	MB_RTU_SLAVE	
xEnable : BOOL stComSettings : COMSETTINGS timResponseDelay : TIME usiSlaveAddress : USINT pSlaveBuffer : POINTER TO BYTE uiSlaveBufferSize : UINT xForbidRemoteWrite : BOOL xLowerByteForward : BOOL	xDone : BOOL xBusy : BOOL xError : BOOL eSlaveError : OMB_Slave_SLAVE_BLOCK_ERROR	

Блок используется для настройки ПЛК в режиме Slave по протоколу Modbus RTU. Пока вход xEnable имеет значение TRUE, slave активен и отвечает на запросы. По заднему фронту входа xEnable блок прекращает работу.

ФБ содержит интегрированный блок СОМ-портом, поэтому в качестве аргументов принимает только сетевые настройки (открывать порт другими блоками для получения дескриптора не требуется).

Поддерживаются следующие функции Modbus: 0x01, 0x02, 0x03, 0x04, 0x05, 0x06, 0xF, 0x10.

Все области памяти наложены друг на друга (Data Model with only 1 block).

Поддерживается до 125 регистров в запросе.

Название	Тип данных	Описание	
Входы			
xEnable	BOOL	Пока вход имеет значение TRUE, блок находится в работе. По заднему фронту происходит прекращение работы slave- устройства.	
stComSettings	COMSETTINGS (см. библиотеку SysLibCom)	Настройки COM-порта, на котором функционирует slave.	
timResponseDelay	TIME	Задержка перед отправкой ответа master- устройству (значение по умолчанию – 20 мс.	
usiSlaveAddress	USINT	Адрес данного устройства (1247).	
pSlaveBuffer	POINTER TO BYTE	Указатель на буфер данных slave-устройства.	
uiSlaveBufferSize	UINT	Размер буфера в байтах.	
xForbidRemoteWrite	BOOL	Запрет на изменение данных slave- устройства со стороны master-устройства. Если вход имеет значение TRUE, то данные slave-устройства будут доступны только для чтения.	
xLowerByteForward	BOOL	Определяет порядок следования байт в буфере Slave-устройства. Значение FALSE (по умолчанию) - little endian, TRUE - big endian.	
Выходы			
xDone	BOOL	Флаг «работа блока была завершена».	
xBusy	BOOL	Флаг "блок находится в работе".	
xError	BOOL	Флаг "произошла ошибка".	

### 3. Создание проекта с использованием библиотеки

- Добавьте в проект следующие библиотеки:
   1.1 OwenModbusSlave_v2.3.9.4.lib;
   1.2 BufferTools_v2.3.9.01.lib.
- 2. Объявите в программе структуру SLAVE_DATA, которая будет представлять собой карту регистров slave-устройства. При наполнении структуры данными требуется соблюдать выравнивание (см. п. 4).

🞭 CoDeSys - (Untitled)*	#### > 3-2-			
Файл Правка Проект Вставить Доп	юлнения Онлайн Окно Справка			
12 - I I I I I I I I I I I I I I I I I I				
Типы данных Lave_Data (Struct)	SLAVE_DATA          0001       TYPE SLAVE_DATA:         0002       STRUCT         0003       wVar1: WORD;         0004       wVar2: WORD;         0005       rVar:         0006       END_STRUCT         0007       END_TYPE         0008			

Рис. 3.1 – Создание структуры с картой регистров slave-устройства

Данная структура соответствует следующей карте регистров:

Переменная	Номер регистра	Комментарий
wVar1	WORD	Поддерживаются функции
wVar2	WORD	как для Input, так и для
rVar	REAL	Holding регистров.

3. Объявите в программе экземпляры структур SLAVE_DATA и COMSETTINGS, а также экземпляр ФБ MB_RTU_SLAVE:

🎭 PLC_PRG (PRG-CFC)			
0001 PROGRAM PLC_PRG			
0002VAR			
0003 stComSettings: COMSETTINGS;			
0004 stSlaveData:	SLAVE_DATA;		
0005 fbRtuSlave:	MB_RTU_SLAVE;		
0006 END_VAR			
0007			

Рис. 3.2 – Объявление переменных программы

#### 4. Код программы будет выглядить следующем образом:



Рис. 3.3 – Код программы

В рамках примера используется ПЛК110 [M02]. Идентификатор его порта RS485-1 имеет значение «0». Информация об идентификаторах портов других ПЛК приведена в документации на ПЛК. Используемые сетевые настройки: 115200-8-N-1. Расшифровка значений для переменных byParity и byStopBits структуры COMSETTINGS приведены в документации на библиотеку SysLibCom.

5. Подключитесь к ПЛК и загрузите проект. Опросите его нужным вам master-устройством.



Рис. 3.4 – Запуск проекта

<ul> <li>Image: Second se</li></ul>		123 - ОВЕН ОРС-сервер		_ <i>8</i> :
Файл Проект				
Остановить опрос Вставить Хопировать Переместить вниз Доб Удалить	Добавить Добавить из Добавить устройство библиотеки	ъ из Добавить Добавить Сохранит а группу тег библиот	ть в Импорт Экспорт Обновить ? Сп Обновить программу ? О	правка программе
🔺 🚆 Сервер	Теги Устройства			
и 🔿 Узел1	Имя Адрес	Значение Тип данных Кач	нество Комментарий	
🔺 💶 Устройство1	Устройство1.Ter1 Holding Regist	ters [0] 6 Word GO	OD	^
Ter1	Устройство1.Ter2 Holding Regist	ters [1] 5 Word GO	OD	
Ter2	Устройство1.Ter3 Holding Regist	ters [2] 22,11 Float GO	OD	
Tor?				
				v
	NK O C			,
	журнал Ошиоки			
	№ 🔹 Метка времени	Устройство Порт	Формат посылки С	Сервисное сообщение
	0000001055 08-06-2018 13:46	26.938 Узел1.Устройство1 Rx	01 03 08 00 06 00 05 E1 48 41 B0 B8 19	^
	0000001054 08-06-2018 13:46	26.913 Узел1.Устройство1 Тх	01 03 00 00 00 04 44 09	
	0000001053 08-06-2018 13:46	25.935 Узел1.Устройство1 Rx	01 03 08 00 06 00 05 E1 48 41 B0 B8 19	
	0000001052 08-06-2018 13:46	25.910 Узел1.Устройство1 Тх	01 03 00 00 00 04 44 09	
	0000001051 08-06-2018 13:46	24.927 Узел1.Устройство1 Rx	01 03 08 00 06 00 05 E1 48 41 B0 B8 19	
	0000001050 08-06-2018 13:46	24.901 Узел1.Устройство1 Тх	01 03 00 00 00 04 44 09	
	0000001049 08-06-2018 13:46	23.917 Узел1.Устройство1 Rx	01 03 08 00 06 00 05 E1 48 41 B0 B8 19	
	0000001048 08-06-2018 13:46:	23.891 Узел1.Устройство1 Тх	01 03 00 00 00 04 44 09	
	0000001047 08-06-2018 13:46:	22.913 Узел1.Устройство1 Rx	01 03 08 00 06 00 05 E1 48 41 B0 B8 19	~ ~
	<			>

Рис. 3.5 – Опрос ПЛК с помощью ОРС-сервера

#### 4. Выравнивание данных в структурах

При работе с библиотекой удобно формирмировать карту регистров ПЛК с помощью создания структуры. Для корректного размещения элементов структуры требуется соблюдать выравнивание. Для этого необходимо следовать описанным ниже правилам:

- Не использовать переменные типа BOOL, BYTE, SINT, USINT (при необходимости можно работать с логическими значениями с помощью переменных типа WORD/DWORD по принципу битовой маски);
- Переменные DWORD/DINT/UDINT/REAL должны объявляться либо после переменной одного из таких же типов, либо после двух последовательно обяъявленных переменных типа WORD/UINT/INT.
- Переменные типа STRING должны иметь количество символов, кратное 4.

🎭 SLAVE_DATA				
0001 T	YPE SLAVE_D	ATA:		
0002 S	TRUCT			
0003	wVar1:	WORD;	(*регистр 0*)	
0004	wVar2:	WORD;	(*регистр 1*)	
0005	dwVar1:	DWORD:	(*регистры 2-3*)	
0006	dwVar2 :	DWORD;	(*регистры 4-5*)	
0007	rVar:	REAL;	(*регистры 6-7*)	
0008	dwVar3:	DWORD;	(*регистры 8-9*)	
0009	sVar:	STRING(12);	(*регистры 10-14*)	
0010 END_STRUCT				
0011 END_TYPE				

Рис. 4.1 – Пример правильного объявления структуры



Рис. 4.2 – Пример неправильного объявления структуры – перед переменными типа DWORD объявлена только одна переменна типа WORD (их должно быть как минимум две)