

450081, Республика Башкортостан, г. Уфа ул. Шота Руставели, д. 51/1, оф. 104 Тел.: 8 (800) 775-74-70 E-mail: support@a-t-tech.ru custom-eng.ru



Модуль процессорный

K15.CPU.LX1

Руководство по программированию ЕСЛТ.421457.002 РП





СОДЕРЖАНИЕ

1	Обі	цие сведения	4
	1.1	Создание проекта	4
2	По	цключение к модулю	6
	2.1	Подключение к модулю	6
	2.2	Изменение IP адресов портов ЕТН1 и ЕТН2	7
3	По	цключение модулей K15	9
	3.1	Добавление модулей К15	9
	3.2	Присваивание адресов DIP	12
4	Mo	dbus RTU/ASCII	13
	4.1	Создание соединения Modbus RTU/ASCII	13
	4.2	Slave RTU/ASCII	13
	4.3	Master RTU/ASCII	14
5	Mo	dbus TCP	17
	5.1	Создание соединения Modbus TCP	17
	5.2	Server TCP	18
	5.3	Client TCP	19
6	Pa	бота с ОРС UA	21
7	Pa	бота с библиотеками	24
	7.1	Подключение библиотек	24
	7.2	Библиотека K15_LX1_gpio	24
	7.3	Библиотека K15_LX1_retain	25
	7.4	Библиотека K15_LX1_iec104	27
	7.5	Библиотека K15_LX1_service	29
8	Про	ограммированиес помощью API	34
	ПРИЈ	ЮЖЕНИЕ А	38



Настоящее руководство по программированию (в дальнейшем – РП) предназначено для программирования и настройки модулей процессорных К15.CPU.LX1 (в дальнейшем – модули).

Эксплуатация модулей должна осуществляться специально обученным персоналом, изучившим настоящее РП.

В связи с постоянной работой по совершенствованию модулей, в их программное обеспечение могут быть внесены незначительные изменения, не отраженные в настоящем РП.



1 ОБЩИЕ СВЕДЕНИЯ

К15.CPU.LX1 – процессорный модуль общего применения для решения различных задач автоматизации. Для создания программной логики его работы используется среда разработки (IDE) Codesys 3.5 SP16 Patch4, которая позволяет программировать модуль, используя стандартные языки МЭК 61131-3. Это облегчает реализацию проекта и существенно сокращает сроки разработки. Подробнее о среде разработки, а также об основных приемах программирования в этой среде можно узнать в справке в составе среды либо на сайте производителя <u>www.codesys.com</u>.

1.1 Создание проекта

1.1.1 Для создания проекта необходимо установить среду разработки Codesys 3.5 SP16 Patch4 с установками по умолчанию. После этого установить таргетфайл модуля. Для этого выбрать «Инструменты > Репозиторий устройств» и нажать «Установить...» . В открывшемся окне выбрать на локальном диске таргет-файл К15.LX1.devdesc. Устройство должно появиться в репозитории в разделе «ПЛК».

1.1.2 Далее необходимо добавить EDS-файлы модулей, которые будут установлены в корзине. Это делается аналогичным способом через репозиторий устройств. Но при указании файлов на локальном диске необходимо выбрать тип файлов «EDS и DCF файлы» в нижнем правом углу окна Проводника. Устройства должны появиться в репозитории в разделе «Промышленные сети (fieldbus) / CANopen / Удаленное устройство».

Фай	іл Правка	Вид	Проект	Компиляция	Онлайн	Отладка
1	Новый прое	кт			CTRL+1	V 🕅 🛛
2	Открыть про	рект			CTRL+(D
	Закрыть про	ект				
	Сохранить п	роект			CTRL+	s •
	Сохранить п	роект к	ак			
	Архив проек	ста				
	Выгрузка ис	ходного	о кода			
	Загрузка исх	одного	кода			
5	Печать					
	Предварите	льный г	росмотр			
D	Параметры	страни	цы			
	Недавние пр	оекты				•
	Выход				ALT+F	4

Рисунок 1 - Создание проекта



1.1.3 Теперь можно создавать новый проект. Для этого необходимо выбрать «Файл > Новый проект» (см. рисунок 1). В открывшемся окне «Новый проект» выбрать «Стандартный проект», имя проекта и путь, по которому будут созданы файлы проекта (рисунок 2). В следующем окне «Стандартный проект» выбрать устройство K15.LX1 (A and T), а также язык программирования базового объектного блока (POU) PLC_PRG.

🛅 Новый проек	г				×
Категории	еки	Шаблоны Проекты НМІ	(Пустой проект	Стандартный проект	Стандартный проект Арр
Проект, содержа	щий одно устройсте	< с	ие и пустук	о реализацию дл	> PLC_PRG
Имя	Проект1				
Расположение	C: \Users \User \Des	ktop\Tect-проект			~
			[ОК	Отмена

Рисунок 2 - Стандартный проект



2 ПОДКЛЮЧЕНИЕ К МОДУЛЮ

2.1 Подключение к модулю

2.1.1 Необходимо собрать корзину с модулями ввода-вывода, подключить порт ETH1 либо ETH2 процессорного модуля к ПК либо локальной сети и подать питание согласно Руководству по эксплуатации. Убедится в установлении соединения с портом ETH1 либо ETH2 по индикации на портах, а также в том, что запустилось встроенное ПО модуля. Об этом свидетельствует короткий сигнал (1-2 сек) через несколько секунд после подачи питания.

2.1.2 Во вновь созданном проекте Codesys в дереве проекта слева необходимо дважды щелкнуть мышью на Device (K15.LX1). В открывшемся окне в правом поле ввода, соответствующем адресу процессорного модуля, ввести IP адрес порта, к которому подключен ПК.

Внимание! Процессорный модуль имеет статические базовые адреса портов 192.168.1.251 (ETH1) и 192.168.1.252 (ETH2). Для их смены см. «Изменение IP адресов портов ETH1 и ETH2».

Device X					
Установки соединения	Сканировать сеть Gateway -	Устройство 👻			
Приложения					
Резервное копирование и восстановление					
Файлы	•			•••	
Журнал		Gat	eway	102 169 1 251	
Установки ПЛК		IP-Address:	Ŷ	Имя устройства:	~
Оболочка ПЛК		localhost Port:		firefly Аллес устройства:	
Пользователи и группы		1217		0301.9000.2DDC.C0A8.01FB	
Права доступа				IP-адрес устройства: 192.168.1.251	
Символьные права				ID таргета: 10C0 0101	
МЭК-объектов				Тип таргета: 4096	
Размещение задачи				Производитель таргета: Prolog	
Состояние				Версия таргета: 3.5.16.40	
Информация					
	Raus vers sterrs verse saure				

Рисунок 3 – Подключение к процессорному модулю

2.1.3 После ввода IP адреса нажать Enter. Если подключение установлено, в появившемся окне «Вход в систему» ввести имя пользователя «admin» и пароль «admin». После этого можно подключиться в режим Онлайн, нажав «Онлайн > Логин». В этом режиме можно загрузить проект, создать загрузочное приложение, сбросить,



запустить/остановить исполнение проекта и тд. Для корректировки проекта необходимо снова перейти в режим Оффлайн, нажав «Онлайн > Отключение».

2.2 Изменение IP адресов портов ETH1 и ETH2

2.2.1 Для смены базовых IP адресов предусмотрено подключение через порт Debug (Туре-С) на лицевой стороне модуля. Для этого понадобится:

- кабель USB A USB Type-C;
- терминальная утилита MobaXterm.

2.2.2 Необходимо подключить кабель USB к ПК, установить при необходимости драйвер CH либо FTDI виртуального COM-порта.

2.2.3 Запустить утилиту MobaXterm, выбрать «Session > Serial». В открывшейся вкладке указать виртуальный СОМ-порт модуля и указать скорость обмена.

COM Com	16 (USB-Er Sessions	hanced-S	SERIAL CH X server	19101 (CO Tools	M6)) Games S	ettinas	Macros	Help										—		\times
Session	Servers	🐧 Tools	Market Games	t Sessions	View	Split	Y MultiExe	ec Tunne	ling Packa	ges Setti	¢ (ngs F	? lelp							X X server	(U) Exit
Quick o	connect						<u>^</u>	12. COM	6 (USB-En	hanced-SE	erial of	× 💿	1				_			0
🔶 🚺	ession sett	ings														×				1
*	SSH	Telnet	₽ Rsh	Xdmcp	💻 RDP	VNC	🌏 FTP	e SFTP	ي Serial	9 File	Shell	Browser	Mosh	💖 Aws S3	II WSL					
	💉 Bas	ic Serial s	settings																	
	Serial port * COM6 (USB-Enhanced-SERIAL CH9101 (CON ~ Speed (bps) * 1500000 ~																			
	🖋 Advanced Serial settings 📩 💼 Terminal settings																			
						Ļ														
			T A	erminal for	nt settings				Terminal c	olor settin	igs									
			_																	
		Back	isnace se	nds AH		Z Lise Wir	adows PA	гн	Termina	l type: vt	erm	~			<u>,</u>					
			terminal o	utput to:			aono m		P	aste delay	r: Auto	~		•	∽ _					
	Syntax highlighting: Standard keywords (OK/warning/error/)																			
							OK		🙁 Cai	ncel										
							- on		• • • •											

Рисунок 4 – Утилита MobaXtrem

Внимание! В списке доступных скоростей отсутствует нужная скорость. Поэтому её необходимо ввести вручную в поле Speed (bps) скорость, равную 1500000 (1,5 Mbps).

2.2.4 После этого перейти во вкладку «Terminal settings» и снять флажок «Backspace sends ^H». Затем нажать ОК. Если соединение установлено, появляется черное окно терминала справа. Далее нажать Enter, после чего появится приглашение командной строки в каталог «/root».



2.2.5 В этом каталоге находятся два скрипта изменения IP портов ETH1 и ETH2. Чтобы изменить параметры порта ETH1, необходимо ввести в поле командной строки «./eth1.sh» и нажать Enter. При наличии соединения порта с сетью будет выведен текущий IP адрес, затем будет предложено ввести новый. Далее необходимо будет ввести даteway IP.



Рисунок 5 – Изменение параметров порта

2.2.6 При корректном вводе в командной строке появится сообщение «done!» и новые параметры применятся сразу без перезагрузки ПЛК. Аналогично настраиваются параметры порта ETH2 через скрипт eth2.sh.



3 ПОДКЛЮЧЕНИЕ МОДУЛЕЙ К15

3.1 Добавление модулей К15

3.1.1 Для работы с модулями К15 необходимо нажать правой кнопкой мыши на объект «Device (K15.LX1)» в дереве проекта и выбрать «Добавить устройство...». В директории «Промышленные сети > CANbus» выбрать объект CANbus и двойным щелчком добавить его в дерево проекта. Закрыть окно добавления устройств.

CANbus_1					
йствие					
Добавить устройство 🕜 Встави	пь устрой	тво 🔘 Подключ	ить устройств	о 🔘 Обно	вить устройств
		1			
рока для полнотекстового поиска		Поставщик	<all th="" vendors<=""><th>></th><th>~</th></all>	>	~
мя	Пос	тавщик		Версия	Описание:
1 Разн.					
Промышленные сети (fieldbus)				
CANbus					
CANbus	3S - 1	Smart Software Sol	utions GmbH	3.5.17.0	Needed for a
NetX CANbus	3S - 1	Smart Software Sol	utions GmbH	3.5.17.0	CANbus on a
🖲 🔐 🔐 Ethercat					
🗉 🕮 Ethernet-адаптер					
🐵 👄 Ethernet/IP					
🗉 🚮 Home&Building Automation					
🗄 📲 💷 Modbus					
Profibus					
🗷 🎹 Profinet IO					
• S sercos					
					>
					2
	_			_	
Группировать по категориям] Отобража	ать все версии (дл	я экспертов)	Показат	гь устаревшие в
MMg: CANbus					
Производитель: 3S - Smart	Software So	olutions GmbH			
Fpynna: CANbus Bencus: 3 5 17 0					
Номер модели:					S
Onucanne: Needed for all field	busses which	h communicate ove	r the CANBus, e	.a.	

Рисунок 6 – Добавление CANbus

3.1.2 В дереве проекта двойным щелчком по объекту «CANbus (CANbus)» открыть вкладку «Общее» и установить скорость передачи 500 кбит/с.

3.1.3 Далее нажать правой кнопкой мыши на объект «CANbus (CANbus)» и, выбрав «Добавить устройство...», перейти в категорию «Промышленные сети (fieldbus) > CANopen > CANopenManager», выбрать объект «CANopen_Manager» и двойным щелчком добавить его в дерево проекта. Закрыть окно добавления устройств.



🗊 Добавить устройство							
Имя: CANopen_Manager_1 Действие							
Добавить устройство ОВставить устройст	тво 🔿 Подключи	ть устройство 🛛 Об	новить устрой	іство			
Строка для полнотекстового поиска	Поставщик	<all vendors=""></all>		~			
Имя — Ш Промышленные сети (fieldbus) — cia CANopen — cia CANopen	Поставщик		Версия	C			
	3S - Smart So	ftware Solutions GmbH	3.5.17.0	c			
CANopen_Manager_SIL2	3S - Smart So	ftware Solutions GmbH	3.5.17.0	c			
 Сій Локальное устройство № 31939 							
< >>Группировать по категориям Отображать все версии (для экспертов) Показать устаревшие верс							
Имя: CANopen_Manager Производитель: 35 - Smart Software Sol Группы: CANopen_Manager Версия: 3.5. 17.0 Номер модели: Описание: CANopen Manager	lutions GmbH		S				

Рисунок 7 – Добавление CANopen_Manager

3.1.4 В дереве проекта двойным щелчком по объекту «CANopen_Manager (CANopen_Manager)» открыть вкладку «CANopen Соотнесение входов/выходов» и выбрать в опциях цикла шины «MainTask».

3.1.5 Теперь можно добавлять модули К15. Для этого необходимо нажать правой кнопкой мыши на объект «CANopen_Manager (CANopen_Manager)» и, выбрав «Добавить устройство...», перейти в категорию «Промышленные сети (fieldbus) > CANopen > Удаленное устройство», в которой выбрать те модули, которые установлены в корзину. Модули можно выбрать последовательно по одному, не закрывая окна выбора устройств, двойным щелчком добавляя каждый модуль в дерево проекта



🗊 Добавить устройство								
Имя: K15_AI8_1 Действие]					
Добавить устройство О Вставить устройство Подключить устройство Обновить устройство								
Строка для полнотекстового поиска р	Поставщик	<all vendors=""></all>	•	\sim				
Image: Second								
✓ Группировать по категориям Отображать все версии (для экспертов) Показать устаревшие верс Имя: К15_АІ8 Производитель: А&Т Группы: Уаленное устройство Версия: Revision = 16 #00000001, File Version = 100.0								
Описание: Импортировано из K15_AI8.eds			2					

Рисунок 8 – Добавление модулей

3.1.6 После того, как модули выбраны, нужно настроить каждый из них в дереве проекта. Для этого необходимо двойным щелчком перейти во вкладку «Общее» модуля и убедиться в совпадении назначенного ID узла и фактически установленного адреса модуля с помощью DIP переключателей (см. п. 4.1. Присваивание адресов DIP). Затем во вкладке «CANopen Coorнесение входов/выходов» установить параметр обновления переменных в значение «Вкл.2 (всегда в задаче цикла шины)». Произвести данные действия для каждого из модулей.

3.1.7 После этого можно привязывать переменные проекта к сигналам модулей. Для этого необходимо перейти во вкладку «CANopen Coorнecenue входов/выходов» модуля и в поле нужного канала модуля произвести соотнесение, дважды кликнув на пустую область первого столбца. Затем, нажав кнопку «…», указать переменную из нужного объекта, к которой будет привязан сигнал модуля.



бщее	Найти	Φν	льтр Пока	азать все			 Добавить ФБ для І 	О-канала 🔭 Перейти к экземпл
DO	Переменная	Соотнесение	Канал	Адрес	Тип	Единица	Описание	
	Application.GVL.test1	~	CH1_IN	%ID0	UDINT			
0	Application.GVL.test2	~	CH2_IN	%ID1	UDINT			
			CH3_IN	%ID2	UDINT			
рнал	🛞 - 🏘		CH4_IN	%ID3	UDINT			
			CH5_IN	%ID4	UDINT			
ореп соотнесение 108/выходов			CH6_IN	%ID5	UDINT			
	🕀 – 🏘		CH7_IN	%ID6	UDINT			
lopen МЭК-объектов	🚊 - 🍫		CH8_IN	%ID7	UDINT			
						Bo	сегда обновлять переменные	Вкл. 2 (всегда в задаче цикла шин

Рисунок 9 – Привязка переменных к модулю

Внимание! При первом подключении к модулю и загрузке проекта исполнение остановлено (Стоп), следовательно, шина CANbus также не запущена. Для запуска опроса модулей необходимо запустить исполнение (Отладка > Старт).

3.2 Присваивание адресов DIP

3.2.1 Адреса модулей на шине должны быть уникальными для каждого из модулей и находиться в диапазоне от 1 до 126. Для задания адреса используются переключатели DIP на боковой стенке.

3.2.2 В случае, если используются модули с 4 DIP переключателями, где 3 из них задают адрес, для увеличения количества модулей на шине используется механизм шифтинга адресов. Это позволяет использовать до 7 модулей каждого типа в одной корзине. Для интерпретации установленного адреса DIP и присвоенного ID адреса необходимо руководствоваться таблицей ниже.

Положение DIP	AI8	DI16	AO2	DO16	DI4 NAMUR
1	1	11	21	31	41
2	2	12	22	32	42
3	3	13	23	33	43
4	4	14	24	34	44
5	5	15	25	35	45
6	6	16	26	36	46
7	7	17	27	37	47

3.2.3 В случае применения модулей с 8 DIP переключателями шифтинг не применяется, и адреса DIP соответствуют адресам ID.



4 MODBUS RTU/ASCII

4.1 Создание соединения Modbus RTU/ASCII

4.1.1 Для создания соединения необходимо в дереве проекта добавить устройство «Modbus_COM», которое обеспечивает обмен по выбранному порту RS485/RS232. Нажать правой кнопкой мыши на объект «Device (K15.LX1)» в дереве проекта и выбрать «Добавить устройство...». В директории «Промышленные сети (fieldbus) > Modbus > Порт Modbus Serial» выбрать объект «Modbus COM» и двойным щелчком добавить его в дерево проекта. Закрыть окно добавления устройств.

вие Бавить устройство О Вставить	устройство 🔿 Полклю	чить устройств	О Обно	вить устройст
а для полнотекстового поиска	Поставщик	<all th="" vendors<=""><th>\$></th><th>~</th></all>	\$>	~
	Поставщик		Версия	Описані ^
Controls Control Cont				
Modbus COM	3S - Smart Software So	lutions GmbH	3.5.16.0	A serial C
Profibus Profinet IO				~
	Modbus_COM вие Saвить устройство Всгавить з для полнотекстового поиска Разн. Пронышленные сети (fieldbus) - CM CANbus - CM CANbus - Ethernet-ддаптер - Ethernet/IP - Ethernet/IP - Ethernet/IP - Ethernet/IP - Ethernet/IP - Modbus Serial - Modbus Serial - Modbus COM - Modbus COM - Modbus COM	Modbus_COM вие Завить устройство Вставить устройство Подслю з для полнотекстового поиска Поставщик Поставщик Разн. Проньшленные сети (fieldbus) См CANbus См CANbus Ethernet-ддаптер Ethernet/IP Home&Building Automation III Modbus Final Modbus Serial Modbus COM 3S - Smart Software Sol III Profileus	Modbus_COM вие Завить устройство Всгавить устройство Подилючить устройств а для полнотекстового поиска Поставщик <ali vendors<br="">Поставщик <ali vendors<br="">Поставщик Разн. Проньшленные сети (fieldbus) - См CANbus См CANbus Еthernet-ддаптер Ethernet/IP Home&Building Automation Modbus COM 35 - Smart Software Solutions GmbH </ali></ali>	Модbus_СОМ вие Залля полнотекстового поиска Поставщик All vendors> Поставщик All vendors> Поставщик Версия Разн. Поставщик Версия См. САНьиз См. САНьиз См. САНьиз Версия Версия Версия Фанненные сети (fieldbus) См. САНьиз Версия См. САНьиз Версия Версия Версия Версия Версия

Рисунок 10 – Добавление Modbus COM

4.1.2 В дереве проекта двойным щелчком по объекту «Modbus_COM (Modbus COM)» открыть вкладку «Общее» и установить номер выбранного порта согласно таблице:

Порт на ПЛК	СОМ-порт в проекте
RS485-1	1
RS485-2	2
RS485-3	3
RS232	4

4.1.3 Далее выбрать требуемую скорость передачи, четность, и количество бит.

Внимание! Максимальное число объектов «Modbus_COM» не должно быть больше числа физических портов, то есть больше четырех. Каждый из четырех портов может выступать Master- либо Slave- устройством в любом сочетании.

4.2 Slave RTU/ASCII

4.2.1 Для создания Slave соединения (когда модуль принимает запросы мастерузла) необходимо в дереве проекта добавить устройство «Modbus Serial Device». Нажать



правой кнопкой мыши на объект «Modbus_COM (Modbus COM)» в дереве проекта и выбрать «Добавить устройство...». В директории «Промышленные сети (fieldbus) > Modbus > Устройство Modbus Serial» выбрать объект «Modbus Serial Device» и двойным щелчком добавить его в дерево проекта. Закрыть окно добавления устройств.

A: Modbus_Senal_Device				
цействие)) Добавить устройство ○ Вставить ус	стройство О Подклю	чить устройство	О Обнови	ть устройств
трока для полнотекстового поиска	Поставщик	<all vendors=""></all>		~
Лмя ☐ Промышленные сети (fieldbus) ☐ ■ ■ Modbus ☐ ■ ■ Macrep Modbus Serial ☐ ■ ■ Macrep Modbus Serial	Поставщик		Версия	Описани
	35 - Smart Sortware	Solutions GmbH	3.5.17.0	A device t

Рисунок 11 – Добавление Modbus Serial Device

4.2.2 В дереве проекта двойным щелчком по объекту «Modbus_Serial_device (Modbus Serial Device)» открыть вкладку «Modbus Serial Device» и установить необходимые параметры устройства: ID (сетевой адрес Modbus), количество регистров хранения, входных регистров, дискретных битовых областей, а также смещения их адресов.

4.2.3 Во вкладке «Modbus Serial Device Соотнесение входов/выходов» установить параметр обновления переменных в значение «Вкл.2 (всегда в задаче цикла шины)». Затем выбрать в опциях цикла шины «MainTask».

4.2.4 После этого можно в этой же вкладке осуществить привязку переменных проекта к регистрам Modbus аналогично привязке переменных к сигналам модулей К15.

4.3 Master RTU/ASCII

4.3.1 Для создания Master соединения (когда модуль передает запросы слейвузлам) необходимо в дереве проекта добавить устройство «Modbus Master, COM Port». Нажать правой кнопкой мыши на объект «Modbus_COM (Modbus COM)» в дереве проекта



и выбрать «Добавить устройство...». В директории «Промышленные сети (fieldbus) > Modbus > Master Modbus Serial» выбрать объект «Modbus Master, COM Port» и двойным щелчком добавить его в дерево проекта. Закрыть окно добавления устройств.

Добавить устройство				2
Имя: Modbus_Master_COM_Port				
Действие				
 Добавить устройство Вставить устрой 	ство 🔿 Подключи	пь устройство 🤇) Обновить у	стройство
Строка для полнотекстового поиска	Поставщик	<all vendors=""></all>		\sim
Имя	Поставщик		Версия	Опис
Промышленные сети (fieldbus) Image: Modbus Image: Macrop Modbus Serial				
Modbus Master, COM Port	3S - Smart Softwar	e Solutions GmbH	3.5.17.0	A devi
ы⊢∎ы Устройство Modbus Serial				
<				>
Группировать по категориям Отображ	ать все версии (для	экспертов) 🔤 Г	іоказать уста	ревшие вер

Рисунок 12 – Добавление Modbus Master

4.3.2 В дереве проекта двойным щелчком по объекту «Modbus_Master_COM_Port (Modbus Master, COM Port)» открыть вкладку «Общее» и установить параметры работы мастера. Также в этой вкладке необходимо установить флажок «Автоперезапуск соединения». Во вкладке «ModbusGenericSerialMaster Соотнесение входов/выходов» установить в опциях цикла шины «MainTask».

4.3.3 Затем нужно добавить слейв-узлы, которые предполагается опрашивать. Нажать правой кнопкой мыши на объект «Modbus_Master_COM_Port (Modbus Master, COM Port)» в дереве проекта и выбрать «Добавить устройство…». В директории «Промышленные сети (fieldbus) > Modbus > Слейв Modbus Serial» выбрать объект «Modbus Slave, COM Port» и двойным щелчком добавить его в дерево проекта. Закрыть окно добавления устройств.



🕤 Добавить устройство					×
Имя: Modbus_Slave_COM_Port_1					
Действие					
Добавить устройство Вставить устрой	ство 🔘 Подключи	пь устройство 🌔	Обновить	устройство	
Строка для полнотекстового поиска	Поставщик	<all vendors=""></all>		\sim	
Имя	Поставщик		Версия	Описа	
🗏 🔟 Промышленные сети (fieldbus)					
🗎 модрия Modpus					
Modbus Slave, COM Port	3S - Smart Software	Solutions GmbH	3.5.16.0	A gener	
<				>	
🔽 Группировать по категориям 🗌 Отображ	ать все версии (для	экспертов) 🗌 Г	Токазать уст	аревшие ве	po

Рисунок 13 – Добавление слейв-узлов

4.3.4 После добавления узлов необходимо настроить каждый из них. В дереве проекта двойным щелчком по объекту «Modbus_Slave_COM_Port (Modbus Slave, COM Port)» открыть вкладку «Общее» и установить параметры работы узла: адрес и таймаут ответа. Затем во вкладках «Канал Modbus Slave» и «Modbus Slave Init» добавить необходимые каналы чтения/записи регистров узла.

Канал	
Имя	Channel 0
Тип доступа	Read Holding Registers (Код функции 3) — — — — — — — — — — — — — — — — — —
Триггер	Цикл. 🗸 Время цикла (мс) 100
Комментарий	
Регистр READ	
Сдвиг	0x0000 ~
Длина	1
Обработка оц	вибок Сохранить посл. значен 🗸
Perистр WRITE	E
Регистр WRITE Сдвиг	E 0x0000 V

Рисунок 14 – Добавление каналов чтения/записи регистров узла



5 MODBUS TCP

5.1 Создание соединения Modbus TCP

5.1.1 Для создания соединения необходимо в дереве проекта добавить устройство «Ethernet (Ethernet)», которое обеспечивает обмен по выбранному порту ETH1/ETH2. Для этого необходимо нажать правой кнопкой мыши на объект «Device (K15.LX1)» в дереве проекта и выбрать «Добавить устройство...». В директории «Промышленные сети (fieldbus) > Ethernet-адаптер» выбрать объект «Ethernet» и двойным щелчком добавить его в дерево проекта. Закрыть окно добавления устройств.

добавить устроиство ня: Ethernet Действие Добавить устройство Вставить	устройс	ство 🔿 Подкля	очить устройств	о Обна	овить устройство
Строка для полнотекстового поиска		Поставщик	<all th="" vendors<=""><th>\$></th><th>~</th></all>	\$>	~
Имя	Пос	тавщик		Версия	Описані ^
Burge Ethercat Ethercat Ethernet-agantep Different Ethernet/IP Different/IP Different/IP Different/IP Different/IP Different/IP Different/IP Different/IP	3S - 1	Smart Software S	olutions GmbH	3.5.17.0	Ethernet
Forofibus Forofinet IO					
in S sercos					>
🛛 Группировать по категориям 🔲 О	тобража	ать все версии (д	іля экспертов)	Показат	гь устаревшие в

Рисунок 15 – Добавление Ethernet

5.1.2 В дереве проекта двойным щелчком по объекту «Ethernet (Ethernet)» открыть вкладку «Конфигурация ethernet» и установить интерфейс сети, соответствующий выбранному порту, согласно таблице:

Порт на ПЛК	Интерфейс в проекте
ETH1	eth0
ETH2	eth1

5.1.3 Для этого нажать на кнопку «…» и выбрать нужный интерфейс в открывшемся окне «Сетевые адаптеры». Поля IP адреса, маски сети и шлюза по умолчанию будут прочитаны из ПЛК и вставлены в конфигурацию Ethernet.

Внимание! Для того, чтобы данные об интерфейсах загрузились, необходимо предварительно убедиться в успешном соединении с модулем. Затем отключиться от модуля. При нажатии кнопки «…» подключение установится автоматически.





5.2 Server TCP

5.2.1 Создание сервера Modbus TCP похоже на создание Slave RTU. Для создания Server TCP (когда модуль принимает запросы клиентов TCP) необходимо в дереве проекта добавить устройство «ModbusTCP Slave Device». Для чего необходимо нажать правой кнопкой мыши на объект «Ethernet (Ethernet)» в дереве проекта и выбрать «Добавить устройство...». В директории «Промышленные сети (fieldbus) > Modbus > Слейв-устройство ModbusTCP slave Device» и двойным щелчком добавить его в дерево проекта. Закрыть окно добавления устройств.

🌃 Добавить устройство				×
Имя: ModbusTCP_Slave_Device Действие				
Добавить устройство О Вставить устройс	тво 🔘 Подключ	ить устройство	Обновить ус	тройство
Имя В- Промышленные сети (fieldbus)	Поставщик	<ail vendors=""></ail>	Версия	Опи
⊕- Ethernet/IP ⊡- IIIII Modbus ⊕- IIIII Macrep Modbus TCP				
Слейв-устройство ModbusTCP ModbusTCP Slave Device Profinet IO	3S - Smart Softv	vare Solutions GmbH	3.5.17.0	A de
<				>
Пруппировать по категориям П Отобража	ть все версии (лл	я экспертов) 🗍 По	казать устар	евшие веро
	in the papers (p)			to and bepe

Рисунок 16 – Добавление Modbus TCP Slave Device

5.2.2 В дереве проекта двойным щелчком по объекту «ModbusTCP_Slave_Device (ModbusTCP Slave Device)» открыть вкладку «Страница конфигурации» и установить необходимые параметры устройства: порт, количество регистров хранения, входных регистров, дискретных битовых областей, а также смещения их адресов.

5.2.3 Во вкладке «Modbus TCP Slave Device Соотнесение входов/выходов» установить параметр обновления переменных в значение «Вкл.2 (всегда в задаче цикла шины)». Затем выбрать в опциях цикла шины «MainTask».

5.2.4 После этого можно в этой же вкладке осуществить привязку переменных проекта к регистрам Modbus аналогично привязке переменных к сигналам модулей К15.



5.3 Client TCP

5.3.1 Создание клиента Modbus TCP похоже на создание Master RTU. Для создания клиент-соединения (когда модуль передает запросы серверу TCP) необходимо в дереве проекта добавить устройство «Modbus TCP Master». Для этого необходимо нажать правой кнопкой мыши на объект «Ethernet (Ethernet)» в дереве проекта и выбрать «Добавить устройство...». В директории «Промышленные сети (fieldbus) > Modbus > Master Modbus TCP » выбрать объект «Modbus TCP Master» и двойным щелчком добавить его в дерево проекта. Закрыть окно добавления устройств.

Modbus_ICP_Master_1				
цействие)) Добавить устройство	тройство 🔿 Подклю	чить устройство 🛛	Обновить ус	тройст
трока для полнотекстового поиска	Поставщик	<all vendors=""></all>		~
Имя	Поставщик		Версия	Опи
🗏 🕤 Промышленные сети (fieldbus)				
thernet/IP				
Houses				
Macrep Moubus TCP				
Modbus TCP Master	3S - Smart Sof	tware Solutions GmbH	3.5.17.0	A de
Modbus TCP Master Моdbus TCP Master Киф Слейв-устройство ModbusT	3S - Smart Sof	tware Solutions GmbH	3.5.17.0	A de
Модрия ТСР Modbus TCP Master Подрия Слейв-устройство ModbusT Profinet IO	3S - Smart Sof	tware Solutions GmbH	3.5.17.0	A de
Ш Machep Housus TCP Modbus TCP Master Ш Ш Слейв-устройство ModbusT Ш Profinet IO	3S - Smart Sof	tware Solutions GmbH	3.5.17.0	A de
Modbus TCP Master	3S - Smart Sof	tware Solutions GmbH	3.5.17.0	A de
Macrep Modbus TCP Modbus TCP Master Эник Слейв-устройство ModbusT Эник Слейв-устройство ModbusT	3S - Smart Sof	tware Solutions GmbH	3.5.17.0	A de
Macrep Modbus TCP Master	3S - Smart Sof	tware Solutions GmbH	3.5.17.0	A der

Рисунок 17 – Добавление Modbus TCP Master

5.3.2 В дереве проекта двойным щелчком по объекту «Modbus_TCP_Master (Modbus TCP Master)» открыть вкладку «Общее» и установить параметры работы мастера. Также в этой вкладке необходимо установить флажок «Автоподключение». Во вкладке «ModbusTCPMaster Соотнесение входов/выходов» установить в опциях цикла шины «MainTask».

5.3.3 Затем нужно добавить слейв-узлы, которые предполагается опрашивать. Для чего необходимо нажать правой кнопкой мыши на объект «Modbus_TCP_Master (Modbus TCP Master)» в дереве проекта и выбрать «Добавить устройство...». В директории «Промышленные сети (fieldbus) > Modbus > Слейв Modbus TCP» выбрать объект «Modbus TCP Slave» и двойным щелчком добавить его в дерево проекта. Закрыть окно добавления устройств.



Modbus_ICP_Slave_2				
иствие Добавить устройство 🔵 Вставить	устройство 🔿 Подклю	чить устройство	Обно	вить у <mark>стройст</mark>
рока для полнотекстового поиска	Поставщик	<all th="" vendors:<=""><th>></th><th>~</th></all>	>	~
Iмя Промышленные сети (fieldbus)	Поставщик		Версия	Описание:
Слейв Modbus TCP Modbus TCP Slave	35 - Smart Software So	lutions GmbH	3 5 16 0	A generic Mo
				>
] Группировать по категориям 🔲 О Имя: Modbus TCP Slave	тображать все версии (д	ля экспертов)	Показат	» љ устаревшие

Рисунок 18 – Добавление слейв-узлов

5.3.4 После добавления узлов необходимо настроить каждый из них. В дереве проекта двойным щелчком по объекту «Modbus_TCP_Slave_(Modbus TCP Slave)» открыть вкладку «Общее» и установить параметры работы узла: IP-адрес, порт и таймаут ответа. Затем во вкладках «Канал Modbus Slave» и «Modbus Slave Init» добавить необходимые каналы чтения/записи регистров узла.

(ana/i	
Имя	Channel 0
Тип доступа	Read Holding Registers (Код функции 3) 🛛 🗸
Триггер	Цикл. — Время цикла (мс) 100
Комментарий	
Регистр READ	
Сдвиг	0x0000 ~
Длина	1
Обработка ош	ибок Сохранить посл. значен ∨
Регистр WRITE	E
Сдвиг	0x0000 ~

Рисунок 19 – Добавление каналов чтения/записи регистров узла



6 РАБОТА С ОРС **UA**

6.1 ПЛК имеет встроенный ОРС UA сервер, позволяющий организовать обмен данными с удаленными клиентами ОРС UA.

6.2 Для того, чтобы осуществить подключение к серверу, сначала необходимо перевести ПЛК в анонимный режим работы (рисунок 20). При этом подключение к ПЛК не потребует ввода логина и пароля. Для это необходимо подключиться к ПЛК, используя логин/пароль firefly/firefly. Затем в дереве проекта перейти в меню «Сброс заводской устройства [Device]».



Рисунок 20 – Перевод ЛК в анонимный режим

6.3 В открывшемся окне необходимо снять все галочки, кроме «User Management». После этого произойдет отключение от ПЛК и переход в анонимный режим.

6.4 Далее необходимо в дереве проекта добавить элемент «Символьная конфигурация» (рисунок 21).





Рисунок 21 - Создание символьной конфигурации

6.5 В открывшемся окне символьной конфигурации можно выбрать переменные, отображаемые в структуру OPC UA, и доступные для чтения/записи клиентом (рисунок 22). Диагностические и служебные переменные будут добавлены автоматически.

¢ ∯ ≣⁺	.
+≣ ₿ ¢	1
	pane (11
ия. Удали	ть
эмментарий	
RN	. Удали

Рисунок 22 - Привязка переменных



Внимание! При привязке структур можно выбрать те элементы структуры, которые необходимо отобразить. Это делается с помощью кнопки справа от структуры. По умолчанию будут отображены все элементы.

6.6 После компиляции и загрузки проекта в ПЛК можно осуществлять подключение к серверу ОРС UA. Адресная строка подключения имеет вид:

opc.tcp://[IP адрес]:4840

где **IP адрес** – IP адрес ПЛК, по которому будет осуществляться подключение.

По умолчанию для портов ETH1 и ETH2 анонимные подключения OPC UA должны осуществляться по адресам opc.tcp://192.168.1.251:4840 и opc.tcp://192.168.1.252:4840 соответственно.

При подключении клиент отобразит две структуры: DeviceSet и Server. Во вложенных элементах этих структур можно найти привязанные переменные (рисунок 23).



Рисунок 23 - Пример отображения переменных в клиенте UaExpert



7 РАБОТА С БИБЛИОТЕКАМИ

7.1 Подключение библиотек

7.1.1 Библиотеки предназначены для расширения функционала модуля, а также для упрощения реализации тех или иных частей проекта. Чтобы использовать библиотеку, необходимо ее подключить в свой проект. Для этого сначала библиотека добавляется в репозиторий библиотек Codesys.

7.1.2 Для этого необходимо перейти в «Инструменты > Репозиторий библиотек» и щелкнуть «Установить…». Указать расположение файла компилированной библиотеки. Закрыть окно «Репозиторий библиотек».

7.1.3 Чтобы подключить библиотеку, в дереве проекта дважды нажать на объект «Менеджер библиотек». В открывшемся окне нажать «Добавить библиотеку» и выбрать нужную библиотеку из репозитория.

Библиотека	×	
Введите строку для полнотекстового поиска во всех библиотеках		
Библиотека Application Intern System System	Компания	
Use Cases Смешан.) АbakPLCUtils	Incomsystem	
[CmpSysExec [IoDrvModem	Production association OWEN Production association OWEN	
- L. IVKAbakUtils - L. K15_LX1_gpio	Incomsystem A&T	
K15_LX1_retain Modbus TCP Slave Library	A&T OWEN	
With the second se	Production association OWEN Production association OWEN Production association OWEN	
- U OwenWatchdog	Production association OWEN Production association OWEN	
Дополнительно	Production association OWEN V OK OTMEHa	

Рисунок 24 – Подключение библиотеки в проект

7.2 Библиотека К15_LX1_gpio

7.2.1 Данная библиотека реализует работу встроенных дискретных входов DI и дискретных выходов DO. Для использования в проекте доступны два функциональных блока библиотеки:

GET_DI – возвращает состояние выбранного DI в формате BOOL;

SET_DO – записывает состояние выбранного DO в формате BOOL.



7.2.2 Для использования библиотеки необходимо разместить нужные

функциональные блоки в любом из программных модулей и привязать к соответствующим переменным.



Рисунок 25 – Пример использования библиотекм K15_LX1_gpio

- 7.2.3 Параметры функционального блока GET_DI:
 - DI_num номер канала DI (1..3),
 - val состояние канала (TRUE/FALSE),
 - ERR ошибка получения состояния (TRUE/FALSE),
 - Параметры функционального блока SET_DO:
 - DO_num номер канала DO (1..2),
 - val состояние канала (TRUE/FALSE),
 - ERR ошибка записи состояния (TRUE/FALSE).

7.3 Библиотека K15_LX1_retain

7.3.1 Данная библиотека позволяет создавать retain/persistent переменные в файловой памяти модуля объемом 512 КБ. Для использования в проекте предусмотрен функциональный блок MAP_RETAIN.

7.3.2 Параметры MAP_RETAIN:

• variable – переменная (массив, структура и т.д.), которой необходимо присвоить тип retain/persistent. Размер массива или структуры не должен превышать 255 байт

• index – смещение в памяти ПЛК, по которому хранится значение переменной

• set_def – устанавливает значение переменной равным значению по умолчанию. TRUE – установить значение



• mapped – выводит размер переменной (структуры) в байтах, размещенной в памяти

• next_ind – выводит индекс, следующий за блоком сохраненной переменной (структуры)

• error – код ошибки блока. 0 – ошибок нет, 1 – ошибка чтения файловой памяти ПЛК, 2 – размер переменной (структуры) превышает 255 байт, 3 – выход за пределы файловой памяти.

Device.Application.PLC_PRG				
Выражение	Тип	Значение		
MAP_RETAIN_0	MAP_RETAIN			
MAP_RETAIN_1	MAP_RETAIN			
MAP_RETAIN_2	MAP_RETAIN			
MAP_RETAIN_3	MAP_RETAIN			
🗄 < MAP_RETAIN_4	MAP_RETAIN			
test_BOOL	BOOL	TRUE		
<pre> test_INT </pre>	INT	-150		
test_REAL	REAL	15.4		
🗉 🧳 test_ARRAY	ARRAY [1255] OF BYTE			
Image:	test_STR			
def_flag	BOOL	FALSE		





Рисунок 27 – Пример использования библиотеки К15 – LX1_retain



7.3.3 Рекомендации по использованию библиотеки:

- Не допускать наложения областей сохранения переменных (структур), индекс следующего блока MAP_RETAIN не должен быть меньше значения next_ind предыдущего.
- Обращать внимание на размер mapped сохраняемой структуры. Он может быть больше суммы элементов структуры ввиду механизма выравнивания данных.
- Для экономии оперативной памяти и увеличения быстродействия стараться уменьшать количество блоков MAP_RETAIN. Для этого группировать переменные retain в массивы или структуры.
- Значения по умолчанию указываются обычным способом при объявлении переменных как начальные значения. При выставлении входа set_def = TRUE блока MAP_RETAIN происходит принудительная перезапись на эти значения. Для возможности дальнейшей записи в переменные необходимо вернуть значение set_def в состояние FALSE.

• Обратить внимание, что все переменные и структуры, переданные блокам MAP_RETAIN, являются PERSISTENT. Поэтому при перезаписи проекта те значения, индексы которых не изменились, сохранятся. Для возврата значений к умолчанию использовать вход set_def.

7.4 Библиотека К15_LX1_iec104

Библиотека реализует возможность обмена с сервером по протоколу МЭК 60870-5-104.

Для подключения к удаленному узлу необходимо добавить ФБ CLIENT_104 (рисунок 28).

Параметры CLIENT_104:

enable – запускает/останавливает попытки соединения

IP1..IP4 – адрес TCP удаленного узла

port – порт TCP удаленного узла

СА – адрес точки удаленного узла

recon_tout – таймаут попыток переподключения к удаленному узлу

tag_put – флаг отправки команды записи (телеуправления)

tag_index – начальный индекс регистра для записи

tag_len – количество регистров для записи

connected – флаг подключения к узлу

error – ошибка ФБ.



Device.Application.PLC_PRG			
Выражение	Тип	Значение	П
	CLIENT_104		
🚸 test1	BOOL	TRUE	
test2	BOOL	FALSE	
test3	BOOL	TRUE	
1 CLIENT_104_0	▲ ▼		
CLIENT_104			
192 TP1 error NO EPDOD			
1 — IP3			
163 — IP4			
2404 port			
153 — CA			
5000 reconn_tout			
4096 tag index			
7 tag len			
2			
pool[4096].SIQ.0 TRUE test1 TRUE			
3			
poor(4097).510.0 34635 Cest2 64635			
4			
pool[4098].SIQ.0 TRUE test3 TRUE			

Рисунок 28 - Добавление ФБ клиента

При подключении библиотеки добавляется пул регистров pool[] размером 65536 элементов.

Структура каждого элемента имеет вид как на рисунке 29.

Имя	Тип	Наследовано от	Адрес	Начальн.	Комментарий
typeID	BYTE				идентификатор типа
cause	BYTE				причина передачи
SIQ	BYTE				одноэлементные данные
DIQ	BYTE				двухэлементные данные
value_R	REAL				данные с плавающей точкой
value_W	WORD				данные с плавающей точкой приведенные к word
QDS	BYTE				качество данных с плавающей точкой
time_ms	UINT				метка времени мс
time_min	BYTE				метка времени мин
time_hour	BYTE				метка времени час
time_day	BYTE				метка времени день
time_month	BYTE				метка времени месяц
time_year	BYTE				метка времени год

Рисунок 29 - Структура элемента пула

Пул заполняется автоматически согласно принятым пакетам от удаленного узла. Индекс пула соответствует параметру IOA принятого пакета, а структура элемента заполняется в зависимости от идентификатора типа. Поддерживаются следующие идентификаторы:

type_M_SP_TB_1: //одноэлементные с меткой времени



type_M_DP_TB_1: //двухэлементные с меткой времени type_M_ME_TF_1: //с плавающей точкой с меткой времени typeID_t.type_C_SC_NA_1: //одноэлементная команда typeID_t.type_C_SC_TA_1: //одноэлементная команда с меткой времени typeID_t.type_C_DC_NA_1: //двухэлементная команда typeID_t.type_C_DC_TA_1: //двухэлементная команда при передаче команды записи возможна отправка блока элементов пула размером

от 1 до 7.

7.5 Библиотека K15_LX1_service

Данная библиотека предоставляет расширенные информационные и диагностические данные о ПЛК и корзине модулей ввода-вывода.



Рисунок 30 - Использование информационных функций

Библиотека состоит из следующих информационных функций (рисунок 30):

GETCPUID_SERVICE – возвращает уникальный идентификационный номер процессора ПЛК. Возвращаемый тип: STRING.

GETFWVER_SERVICE – возвращает версию установленного рантайма Codesys. Возвращаемый тип: STRING.



GETCPULOAD_SERVICE – возвращает текущую загрузку процессора ПЛК в процентах. Возвращаемый тип: REAL.

Внимание! Ввиду особенностей работы функции GETCPULOAD_SERVICE при периодическом вызове рекомендуется размещать ее в отдельном таске, так как она блокирует его выполнение на 2-3 сек.

GETCRC_SERVICE – возвращает контрольную сумму прикладного ПО, загруженного в ПЛК. Возвращаемый тип: STRING.



Рисунок 31 - Использование функций диагностики корзины

Для диагностики корзины модулей ввода-вывода предусмотрены функции (рисунок 31):

GETSTATECAN_SERVICE – возвращает статус работы менеджера CANopen. Для корректной работы функции необходимо на входе CANopen_Man указать используемый менеджер (по умолчанию CANopen_Manager). Возвращаемое значение соответствует перечислению BUS_STATE и принимает значения DISABLED, ERROR, RUNNING, STOPPED, UNKNOWN.

GETSTATEMODULE_SERVICE – возвращает статус работы модуля с указанным node ID. Для корректной работы функции необходимо на входе CANopen_Man указать используемый менеджер (по умолчанию CANopen_Manager). Возвращаемое значение соответствует перечислению DEVICE_STATE и принимает значения CONFIGURED, DISABLED, ERROR, NOT_CONFIGURED, NOT_FOUND, RUNNING, STOPPED, UNKNOWN. В случае, если node_ID меньше 1 или больше 126, а также если модуль с таким ID не представлен в дереве проекта, функция возвращает UNKNOWN.





Рисунок 32 - Функциональный блок состояния портов Ethernet

Для получения информации о состоянии портов Ethernet существует функциональный блок GETINFOETH_SERVICE. При создании его экземпляра необходимо задать входной параметр port. Он принимает значения перечисления ETH_t и соответствует портам ETH1 и ETH2 ПЛК.

В случае, если в порт вставлен патч-корд и установлен линк с подсетью, функциональный блок вернет значение state равное ETH_UP, а также параметры порта: MAC adpec, IP adpec и маску подсети. В противном случае значение state будет равно ETH_DOWN и будет возвращен только MAC adpec порта. В случае, если указанный порт недоступен, будет возвращено значение UNKNOWN. Значения соответствуют перечислению state_t.

GETINFOMEMORY_SERVICE_0	
GETINFOMEMORY_SERVICE	
TRUE enable RAM_AVAIL_KB	3 3650068
RAM_FREE_KB	3 - 3295328
RAM_USED_KB	3 - 354740
RAM_USED_PERC	9.72 +
DISK_AVAIL_KB	3 - 26446852
DISK_FREE_KB	3 - 25953160
DISK_USED_KB	493692
DISK_USED_PERC	1.87 +
USB_AVAIL_KB	3 - 0
USB_FREE_KB	3 - 0
USB_USED_KB	3 — 0
USB_USED_PERC	0
SD_AVAIL_KB	3 - 0
SD_FREE_KB	3 - 0
SD_USED_KB	3 - 0
SD_USED_PERC	0

Рисунок 33 - Функциональный блок состояния памяти ПЛК



Данный функциональный блок позволяет оценить использование памяти ПЛК и возвращает следующие параметры (рисунок 33):

RAM_AVAIL_KB – доступная оперативная память в КБ. Возвращаемый тип: UDINT.

RAM_FREE_KB – свободная оперативная память в КБ. Возвращаемый тип: UDINT.

RAM_USED_KB – используемая оперативная память в КБ. Возвращаемый тип: UDINT.

RAM_USED_PERC – используемая оперативная память в процентах. Возвращаемый тип: REAL.

DISK_AVAIL_KB – доступная память внутреннего хранилища в КБ. Возвращаемый тип: UDINT.

DISK_FREE_KB – свободная память внутреннего хранилища в КБ. Возвращаемый тип: UDINT.

DISK_USED_KB – используемая память внутреннего хранилища в КБ. Возвращаемый тип: UDINT.

DISK_USED_PERC – используемая память внутреннего хранилища в процентах. Возвращаемый тип: REAL.

USB_AVAIL_KB – доступная память внешнего флеш-накопителя в КБ. Возвращаемый тип: UDINT.

USB_FREE_KB – свободная память внешнего флеш-накопителя в КБ. Возвращаемый тип: UDINT.

USB_USED_KB – используемая память внешнего флеш-накопителя в КБ. Возвращаемый тип: UDINT.

USB_USED_PERC – используемая память внешнего флеш-накопителя в процентах. Возвращаемый тип: REAL.

Внимание! Ввиду особенностей работы ОС, используемой в ПЛК, не рекомендуется устанавливать более одного флеш-накопителя.

SD_AVAIL_KB – доступная память внешней карты microSD в KБ. Возвращаемый тип: UDINT.

SD_FREE_KB – свободная память внешней карты microSD в KБ. Возвращаемый тип: UDINT.

SD_USED_KB – используемая память внешней карты microSD накопителя в КБ. Возвращаемый тип: UDINT.

SD_USED_PERC – используемая память внешней карты microSD в процентах. Возвращаемый тип: REAL.





Рисунок 34 - Функциональный блок чтения часов реального времени

Для контроля за текущей датой и временем предусмотрен функциональный блок GETTIME_SERVICE (рисунок 34). Он возвращает параметры:

DT_current – текущая дата и время ПЛК. Возвращаемый тип: DATE_AND_TIME.

DT_start – дата и время начала работы (первого скана) прикладного ПО. Возвращаемый тип: DATE_AND_TIME.

T_elapsed – время, прошедшее с начала работы прикладного ПО. Возвращаемый тип: TIME.



8 ПРОГРАММИРОВАНИЕС ПОМОЩЬЮ АРІ

8.1 Структура проекта

Makefile – служит для сборки исходных файлов в исполняемое приложение.

Autorun.sh – скрипт для автозагрузки приложения при включении устройства.

Canlib – директория с библиотекой Canopen.

Canopen.service – настройки сервиса при автозагрузке приложения.

Canscripts – директория со скриптами для запуска или остановки интерфейса Can0.

Devicelib – директория с библиотекой модулей.

EedsFiles – директория с файлами eds модулей.

LogFiles – директория с журналом ошибок.

Main.c – главный исходный файл программы.

Servicelib – директория с сервисной библиотекой.

Userlib – директория для пользовательских исходный файлов.

8.2 Функции для работы с модулями К15_DO16

8.2.1 OutState_t DO16_GetChannelState (uint8_t nodeId, numberChannel_t numberChannel) предназначена для считывания состояния канала. Возвращает значение:

STATE_LOW – канал в низком логическом уровне;

STATE_HIGH – канал в высоком логическом уровне.

Принимает первым параметром идентификационный номер модуля, вторым – номер канала от *CH1* до *CH16.*

8.2.2 Void DO16_SetChannelState (uint8_t nodeld, numberChannel_t numberChannel, outState_t outState) предназначена для изменения состояния канала. Принимает первым параметром идентификационный номер модуля, вторым – номер канала от *CH1* до *CH16*, третьим – состояние канала *STATE_LOW* или *STATE_HIGH*.

8.2.3 Float DO16_GetChannelFrequency (uint8_t nodeld, numberChannel_t numberChannel) предназначена для считывания текущей частоты с выхода канала в Гц. Принимает первым параметром идентификационный номер модуля, вторым – номер канала от CH1 до CH16.



8.2.4 Void DO16_SetChannelFrequency (uint8_t nodeld, numberChannel_t numberChannel, float frequency) предназначена для установки частоты на выходе канала в Гц. Принимает первым параметром идентификационный номер модуля, вторым – номер канала от CH1 до CH16, третьим – заданную частоту.

8.3 Функции для работы с модулями K15_DI16

8.3.1 OutState_t DI16_GetChannelState (uint8_t nodeId, numberChannel_t numberChannel) предназначена для считывания состояния канала. Возвращает значение:

STATE_LOW – канал в низком логическом уровне;

STATE_HIGH – канал в высоком логическом уровне.

Принимает первым параметром идентификационный номер модуля, вторым – номер канала от *CH1* до *CH16.*

8.3.2 Uint32_t DI16_GetChannelCounter (uint8_t nodeld, numberChannel_t numberChannel) предназначена для считывания количества импульсов на входе канала. Принимает первым параметром идентификационный номер модуля, вторым – номер канала от CH1 до CH16.

8.3.3 Uint32_t DI16_GetChannelFrequency (uint8_t nodeld, numberChannel_t numberChannel) предназначена для считывания текущей частоты на входе канала. Принимает первым параметром идентификационный номер модуля, вторым – номер канала от CH1 до CH16.

8.4 Функции для работы с модулями К15_АО2

8.4.1 OutType_t AO2_GetChannelType (uint8_t nodeId, numberChannel_t numberChannel) предназначена для считывания типа канала. Возвращает значение:

ТҮРЕ_U – канал в режиме подачи напряжения;

ТҮРЕ_I – канал в режиме подачи тока;

Принимает первым параметром идентификационный номер модуля, вторым – номер канала от *CH1* до *CH2*.

8.4.2 Void AO2_SetChannelType (uint8_t nodeld, numberChannel_t numberChannel, outType_t outType) предназначена для установки типа канала. Принимает первым параметром идентификационный номер модуля, вторым – номер канала от CH1 до CH2, третьим – тип канала TYPE_U или TYPE_I.

8.4.3 Uint16_t AO2_GetChannelValue (uint8_t nodeld, numberChannel_t numberChannel) предназначена для считывания текущего значения на выходе канала (ток в мкА, напряжение в мВ).



Принимает первым параметром идентификационный номер модуля, вторым – номер канала от *CH1* до *CH2*.

8.4.4 Void AO2_SetChannelValue (uint8_t nodeld, numberChannel_t numberChannel, uint16_t value) предназначена для установки значения на выходе канала (ток в мкА, напряжение в мВ). Принимает первым параметром идентификационный номер модуля, вторым – номер канала от *CH1* до *CH2*, третьим – значение на выходе канала.

8.5 Функции для работы с модулями K15_Al8

8.5.1 Uint32_t Al8_GetChannelCode (uint8_t nodeld, numberChannel_t numberChannel) предназначена для считывания кода канала. Принимает первым параметром идентификационный номер модуля, вторым – номер канала от CH1 до CH8.

8.5.2 Uint32_t AI8_GetChannelValue (uint8_t nodeld, numberChannel_t numberChannel) предназначена для считывания значения тока на входе канала, мкА. Принимает первым параметром идентификационный номер модуля, вторым – номер канала от CH1 до CH8.

8.6 Проверка связи с модулями

Для проверки связи с модулями используется функция *uint*32_*t* getDeviceTypeOnModule (uint16_t nodeld), которая возвращает значение:

0-если нет связи с модулем;

Число – связь есть.

Принимает в качестве параметра идентификационный номер модуля.

8.7 Начало работы

Перед началом работы с API необходимо убедиться, что не работает рантайм. Если он запущен, выключить его через *script_OFF.* Более подробно о процедуре отключения рантайма в Руководстве по обновлению API.

Для начала работы с API необходимо выполнить следующую последовательность действий:

- 1. Перейти в директорию cd /home/firefly/userProgram.
- 2. Ввести команду *sh canscripts/upcan.sh* для запуска интерфейса *can0* (выполняется один раз при запуске системы).
- Установить gcc (sudo apt install gcc).
 Если не работает apt, curl -I http://mirrors.aliyun.com sudo apt update



sudo apt clean

- 4. Ввести команду *таке* для сборки приложения.
- 5. Ввести команду ./main для запуска приложения.

8.8 Автозагрузка приложения при старте системы

Для обеспечения автозагрузки приложения при старте системы необходимо:

- 1. Перейти в директорию cd /home/firefly/userProgram.
- 2. Сделать скрипт *autorun.sh* исполняемым, если он таковым не является *sudo chmod* "+x" *autorun.sh*
- 3. Ввести команду sudo cp canopen.service /etc/systemd/system.
- 4. Ввести команду sudo systemctl daemon-reload.
- 5. Ввести команду sudo systemctl enable canopen.service.
- 6. Перезагрузить устройство.

Если требуется исправить программу, необходимо ввести команду sudo systemctl disable canopen.service и перезагрузить устройство. Для запуска сервиса вновь ввести команду sudo systemctl enable canopen.service.



ПРИЛОЖЕНИЕ А

(Справочное)

Карта регистров Modbus RTU

RTU Slave

FUNCTION_BLOCK MBSlave_fb

Входные переменные	Тип данных	Описание	
INIT	BOOL	Com Порт инициализирован (для загрузки Retain, не использовать для остановки)	
Restart	BOOL	Перезапуск СОМ порта	
Port	BYTE	Номер физического СОМ порта 1 – 4	
SlaveID	BYTE	SlaveID ПЛК	
BaudRate_Num	BYTE	Номер битрейта: 0 - 4800 1 - 9600 2 - 19200 3 - 38400 4 - 57600 5 - 115200 6 - 2400 7 - 1200 8 - 600	
BaudRate	UDINT	Может принимать значения: 600 1200 2400 4800 9600 19200 38400 57600 115200	
Parity	BYTE	Паритет: 0 – NONE 1 – ODD 2 – EVEN	
StopBit	BYTE	Стоп бит: 1 – 1 2 - 1.5 3 – 2	
StatReset	BOOL	Сброс статистики	
SlaveStop	BOOL	Остановка опроса	
Send_interrupt	TIME	Прерывание перед записью байт	
Read_interrupt	TIME	Прерывание перед чтением байт	
Send_Timeout	UDINT	Таймаут отправки байт	
Read_Timeout	UDINT	Таймаут чтения байт	
Resp_Timeout	UDINT	Таймаут чтения записи	
Error	BOOL	Ошибка инициализации СОМ порта	
Online	BOOL	Ведется опрос ПЛК	



Входные переменные	Тип данных	Описание
Stat	<u>STATISTIC</u>	Статистика СОМ порта

RTU Master

FUNCTION AddRequest: BOOL

Возвращаемое значение	Тип данных	Описание
AddRequest	BOOL	Результат выполнения загрузки запроса в очередь: TRUE – загружен, FALSE – не загружен, запрос содержит ошибку.

Входные переменные	Тип данных	Описание
SlaveID	USINT	SlaveID Устройства
MB_Func	USINT	Код функции запроса
RegAddr	UINT	Номер регистра
DataLen	UINT	Размер в регистрах
pData	POINTER TO ARRAY [0124] OF WORD	Указатель на место записи ответа или записываемые данные
pStatistic	POINTER TO <u>STATISTIC</u>	Указатель на статистику опроса устройства
pStatus	POINTER TO <u>Master Status</u>	Указатель на результат выполнения опроса
SWAPByte	BOOL	Поменять байты местами
SWAPReg	BOOL	Поменять регистры местами

Входные выходные переменные	Тип данных	Описание
Master	MBMaster_fb	ФБ Мастер порта куда грузить запрос

FUNCTION_BLOCK MBMaster_fb

Входные переменные	Тип данных	Описание
Port	BYTE	Номер физического мастер порта 1 – 4
BaudRate_Num	BYTE	Номер битрейта: 0 – 4800 1 – 9600 2 – 19200 3 – 38400 4 – 57600 5 – 115200 6 – 2400 7 – 1200 8 – 600
BaudRate	UDINT	Может принимать значения: 600 1200 2400



Входные переменные	Тип данных	Описание
		4800 9600 19200 38400 57600 115200
Parity	BYTE	Паритет: 0 – NONE 1 – ODD 2 – EVEN
StopBit	BYTE	Стоп бит: 1 – 1 2 - 1.5 3 – 2
Restart	BOOL	Перезапуск Мастер порта
StatReset	BOOL	Сброс статистики
SlaveStop	BOOL	Остановка опроса
Send_interrupt	TIME	Прерывание перед отправкой запроса
Read_interrupt	TIME	Прерывание перед получением ответа
Read_timeout	TIME	Таймаут получения ответа
Resp_Timeout	UDINT	Таймаут чтения записи
RequestCnt	UINT	Количество запросов в очереди
RequestList	ARRAY [165535] OF <u>RequestItem_typ</u>	Список запросов в Мастер Порт

Выходные переменные	Тип данных	Описание
Error	BOOL	Ошибка открытия Мастер порта
Status	Master Status	Код результата опроса
Online	ARRAY [165535] OF BOOL	Список результата выполнения запросов
SlaveStat	STATISTIC	Статистика Мастер порта
ScanCycleDone	BOOL	Список запросов завершен

Пример описания переменных Мастер порта RTU

F	PROGRAM PLC_PRG				
١	/AR				
	RS1	: ModbusLibrary.MBMaster_fb; // 4	ФБ Мастер порта RTU		
	RS1_C_BIN	: ARRAY [0255] OF BOOL;	// Массив считанных Coil регистров		
	RS1_D_BIN	: ARRAY [0255] OF BOOL;	// Массив считанных Discret регистров		
	RS1_H_DAT	: ARRAY [0255] OF WORD;	// Массив считанных Holding регистров		
	RS1_I_DAT	: ARRAY [0255] OF WORD;	// Массив считанных Input регистров		
	// Массив ста	тусов выполнения запроса			
	RTU_Request_Status : ARRAY [14] OF ModbusLibrary.Master_Status;				
	// Массив статусов загрузки в очередь опроса				
	RTU_Request_Loaded : ARRAY [14] OF BOOL;				
	RS1_Stat	: ModbusLibrary.STATISTIC; //	[/] Статистика опроса устройства		



END_VAR

Пример программы Мастер порта RTU

RS1.Port := 1;
RS1.BaudRate := 9600;
RS1.Parity := 0;
RS1.StopBit := 1;
RS1();
IF RS1.ScanCycleDone THEN
RTU_Request_Loaded[1] := AddRequest(1, 1, 0, 1,
ADR(RS1_C_BIN[0]), ADR(RS1_Stat), ADR(RTU_Request_Status[1]), FALSE, FALSE, RS1);
RTU_Request_Loaded[2] := AddRequest(1, 2, 0, 1,
ADR(RS1_D_BIN[0]), ADR(RS1_Stat), ADR(RTU_Request_Status[2]), FALSE, FALSE, RS1);
RTU_Request_Loaded[3] := AddRequest(1, 3, 0, 8, ADR(RS1_H_DAT[0]), ADR(RS1_Stat), ADR(RTU_Request_Status[3]), TRUE,
FALSE,RS1);
RTU_Request_Loaded[4] := AddRequest(1, 4, 0, 8,
ADR(RS1_I_DAT[0]), ADR(RS1_Stat), ADR(RTU_Request_Status[4]), FALSE, TRUE, RS1);
END_IF

Карта регистров Modbus TCP

TCP Slave

FUNCTION_BLOCK MBSlaveTCP_fb

Входные переменные	Тип данных	Описание
INIT	BOOL	TCP инициализирован (для загрузки Retain, не использовать для остановки)
Restart	BOOL	Перезапуск
ipAdress	NBS.IP_ADDR	IP адрес устройства
Port	UINT	Порт устройства
SlaveID	BYTE	Slave ID
StatReset	BOOL	Сброс статистики
SlaveStop	BOOL	Остановка опроса

Выходные переменные	Тип данных	Описание
Error	BOOL	Ошибка открытия соединения
Online	BOOL	Ведется опрос устройства
Stat	STATISTIC	Статистика СОМ порта
ClientCnt	USINT	Число подключенных клиентов
TCP Master		

FUNCTION AddRequestTCP: BOOL

Возвращаемое значение	Тип данных	Описание
AddRequestTCP	BOOL	Результат выполнения загрузки запроса в очередь: TRUE – загружен, FALSE – не загружен, запрос содержит ошибку.

Входные переменные	Тип данных	Описание
SlaveID	USINT	SlaveID Устройства



Входные переменные	Тип данных	Описание
MB_Func	USINT	Код функции запроса
RegAddr	UINT	Номер регистра
DataLen	UINT	Размер в регистрах
pData	POINTER TO ARRAY [0124] OF WORD	Указатель на место записи ответа или записываемые данные
pStatistic	POINTER TO <u>STATISTIC</u>	Указатель на статистику опроса устройства
pStatus	POINTER TO <u>Master_Status</u>	Указатель на результат выполнения опроса
SWAPByte	BOOL	Поменять байты местами
SWAPReg	BOOL	Поменять регистры местами

Входные выходные переменные	Тип данных	Описание
Master	MBMasterTCP_fb	ФБ Мастер порта куда грузить запрос
FUNCTION_BLO	CK MBMasterTCF	P_fb
Входные переменные	Тип данных	Описание
IP	NBS.IP_ADDR	IP Устройства
Port	UINT	Порт устройства
Restart	BOOL	Пере подключение к устройству
StatReset	BOOL	Сброс статистики
SlaveStop	BOOL	Остановка опроса
Send_interrupt	TIME	Прерывание перед отправкой запроса
Read_interrupt	TIME	Прерывание перед получением ответа
Read_timeout	TIME	Таймаут получения ответа
RequestCnt	UINT	Количество запросов в очереди
RequestList	ARRAY [165535] OF RequestItem_typ	Список запросов в очереди

Выходные переменные	Тип данных	Описание
Error	BOOL	Ошибка соединения
Status	Master_Status	Код результата опроса
Online	ARRAY [165535] OF BOOL	Результаты выполнения запросов (наличие связи)
SlaveStat	STATISTIC	Статистика устройства
ScanCycleDone	BOOL	Очередь запросов выполнена





Пример описания переменных Мастер порта ТСР

PROGRAM PLC_PRG					
VAR					
RSTCP1	: ModbusLibrary.MBMasterTCP_fb	; // ФБ Мастер порта ТСР			
TCP_H_DAT	: ARRAY [0255] OF BOOL;	// Массив считанных Coil регистров			
TCP_I_DAT	: ARRAY [0255] OF BOOL;	// Массив считанных Discret регистров			
TCP_C_BIN	: ARRAY [0255] OF WORD;	// Массив считанных Holding регистров			
TCP_D_BIN	: ARRAY [0255] OF WORD;	// Массив считанных Input регистров			
// Массив статусов выполнения запроса					
TCP_Request_Status : ARRAY [14] OF ModbusLibrary.Master_Status;					
// Массив статусов загрузки в очередь опроса					
TCP_Request_Loaded : ARRAY [14] OF BOOL;					
TCP_Stat	: ModbusLibrary.STATISTIC;	// Статистика опроса устройства			
END_VAR					
	_				

Пример программы Мастер порта ТСР

RSTCP1.IP.sAddr := '192.168.16.74';
RSTCP1.Port := 502;
RSTCP1();
IF RSTCP1.ScanCycleDone THEN
<pre>TCP_Request_Loaded[1] := AddRequestTCP(1, 1, 0, 8,</pre>
ADR(TCP_C_BIN[0]), ADR(TCP_Stat), ADR(TCP_Request_Status[1]), FALSE, FALSE, RSTCP1);
TCP_Request_Loaded[2] := AddRequestTCP(1, 2, 0, 8,
ADR(TCP_D_BIN[0]), ADR(TCP_Stat), ADR(TCP_Request_Status[2]), FALSE, FALSE, RSTCP1);
<pre>TCP_Request_Loaded[3] := AddRequestTCP(1, 3, 1, 8,</pre>
ADR(TCP_H_DAT[0]), ADR(TCP_Stat), ADR(TCP_Request_Status[3]), TRUE, FALSE, RSTCP1);
TCP_Request_Loaded[4] := AddRequestTCP(1, 4, 1, 8,
ADR(TCP_I_DAT[0]), ADR(TCP_Stat), ADR(TCP_Request_Status[4]), FALSE, TRUE, RSTCP1);

END_IF

Описание структур

STATISTIC

Переменная	Тип данных	Описание	
Total	UINT	Отправлено / Получено запросов	
Done	UINT	Успешно обработано	
ErrCode	UINT	Код ошибки: 0 – выполняется 2 – Ошибок нет 1 – неверный SlaveID 3 – неверный CRC 4 – Таймаут	
Vis_RX	BOOL	Индикатор получения запроса	
Vis_TX	BOOL	Индикатор посылки запроса	
Rx	UDINT	всего считано ответов на запрос или получено запросов	
Тх	UDINT	всего обработано запросов или получено ответов	





Master_Status

Значение	Описание
BUSY	Выполняется
ERR_ID	Неверный SlaveID
ERR_OK	Ошибок нет
ERR_CRC	Ошибка CRC
ERR_TIMEOUT	Таймаут
ERR_FUNC	Ошибка функции
ERR_CONNECTION	Ошибка соединения
ERR_REQUEST	Ошибочный запрос в очереди

RequestItem_typ

Переменная	Тип данных	Описание
SlaveID	BYTE	SlaveID Устройства
MB_Func	BYTE	Код функции запроса
RegAddr	WORD	Номер регистра
DataLen	WORD	Размер в регистрах
pData	POINTER TO ARRAY [0124] OF WORD	Указатель на место записи ответа или записываемые данные
pStatistic	POINTER TO <u>STATISTIC</u>	Указатель на статистику опроса устройства
pStatus	POINTER TO <u>Master_Status</u>	Указатель на результат выполнения опроса
SWAPByte	BOOL	Поменять байты местами
SWAPReg	BOOL	Поменять регистры местами