

Протокол передачи данных EIB/KNX



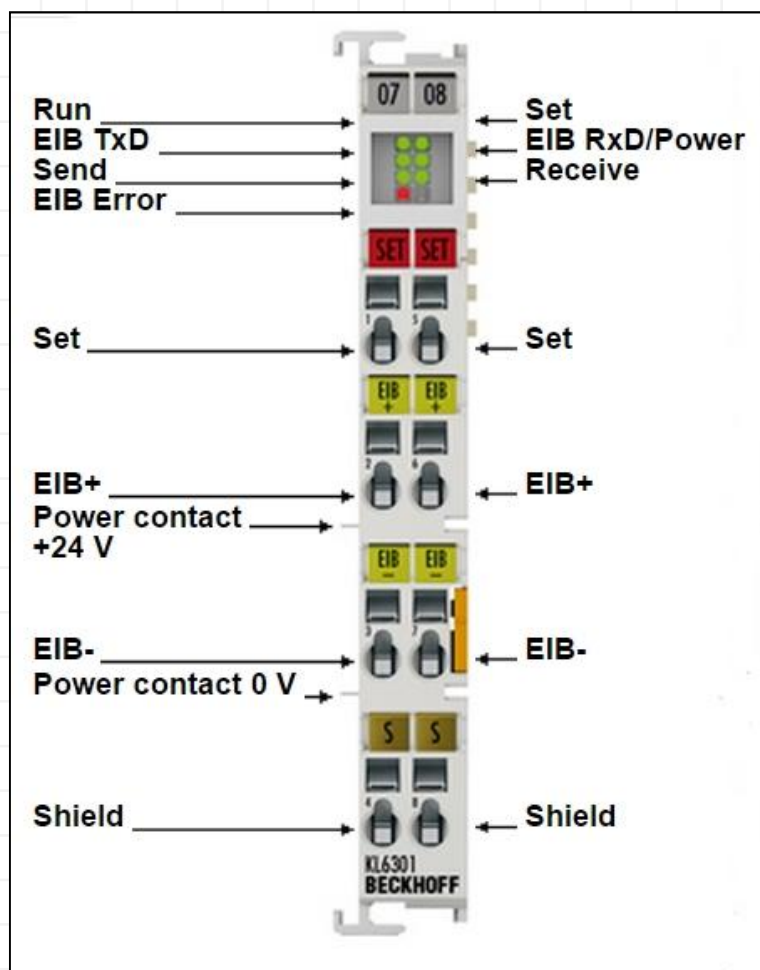
Содержание:

1. Продукты и библиотеки Beckhoff для работы с EIB/KNX
2. Описание функциональных блоков для коммуникации
3. Пошаговая инструкция по реализации проекта

1. Продукты и библиотеки Beckhoff для работы с EIB/KNX

KL6301 - модуль для работы с EIB/KNX

ТсEIB - библиотеки для всех типов контроллеров

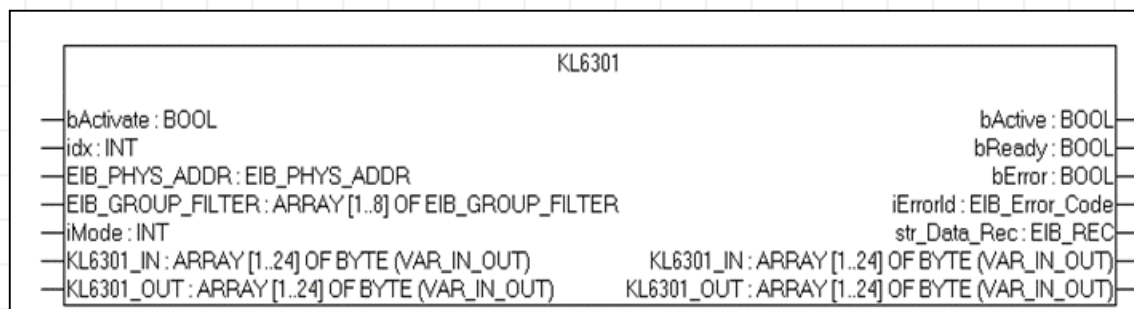


- Function Blocks for receiving
 - EIB_20CTET_FLOAT_REC
 - EIB_20CTET_SIGN_REC
 - EIB_20CTET_UNSIGN_REC
 - EIB_3BIT_CONTROL_REC
 - EIB_40CTET_FLOAT_REC
 - EIB_40CTET_SIGN_REC
 - EIB_40CTET_UNSIGN_REC
 - EIB_8BIT_SIGN_REC
 - EIB_8BIT_UNSIGN_REC
 - EIB_ALL_DATA_TYPES_REC
 - EIB_BIT_CONTROL_REC
 - EIB_BIT_REC
 - EIB_TIME_REC
 - EIB_DATE_REC
- Function Blocks for sending
 - EIB_20CTET_FLOAT_SEND
 - EIB_20CTET_SIGN_SEND
 - EIB_20CTET_UNSIGN_SEND
 - EIB_3BIT_CONTROL_SEND
 - EIB_40CTET_FLOAT_SEND
 - EIB_40CTET_SIGN_SEND
 - EIB_40CTET_UNSIGN_SEND
 - EIB_8BIT_SIGN_SEND
 - EIB_8BIT_SIGN_SEND_EX
 - EIB_8BIT_UNSIGN_SEND
 - EIB_8BIT_UNSIGN_SEND_EX
 - EIB_ALL_DATA_TYPES_SEND
 - EIB_BIT_CONTROL_REC
 - EIB_BIT_SEND
 - EIB_BIT_SEND_MANUAL
 - EIB_DATE_SEND
 - EIB_READ_SEND
 - EIB_TIME_SEND

Для PC систем (x86) и Embedded-PCs (CXxxxx):*Standard.lib**TcBase.lib**TcSystem.lib**TcEIB.lib***Для контроллеров серии BCxx00:***Standard.lib6**PlcHelperBC.lib6**TcEIB.lib6***Для контроллеров серий BCxx50, BCxx20 и BC9191:***Standard.libx**TcBaseBCxx50.libx**TcSystemBCxx50.libx**TcEIB.libx***Для контроллеров серии BXxx00:***Standard.libx**TcBaseBX.libx**TcSystemBX.libx**TcEIB.libx***2. Описание функциональных блоков для коммуникации**

Связь EIB/KNX осуществляется посредством функциональных блоков.

Основным является функциональный блок KL6301



Используется для конфигурирования KL6301 и для обмена данными по шине EIB

Параметры INPUT:

bActivate: Активирует блок в режим обмена данными

idx: Если в PLC программе используется более одного модуля, то каждый KL6301 должен иметь уникальный индекс от 1 до 64.

EIB_PHYS_ADDR: Физический EIB адрес. Адрес по умолчанию 1.2.3. Должен быть уникальным в EIB сети

EIB_GROUP_FILTER: Фильтр адреса. Максимум 4 фильтра.*

iMode:

0 – для прошивки B0 и выше - 4 фильтра, каждый с 64 записями.

1 – для прошивки B1 и выше - 8 фильтров, каждый с 32 записями.

100 – для прошивки B1 и выше – функция мониторинга, могут быть получены телеграммы всем групповым адресам. При мониторинге передача невозможна.

Описание функциональных блоков для коммуникации

EIB_GROUP_FILTER должен быть сконфигурирован до того, как функциональный блок KL6301 будет активирован.

Фильтры применяются ко всем данным, направленным в KL6301. Каждая телеграмма, которая проходит фильтр, становится видимой в функциональном блоке и соответственно ее данные могут быть обработаны. KL6301 отбрасывает EIB телеграммы с адресов, которые не включены в фильтр.

Пример:

Filter 1 group address 1/2/0 length: 10

Все EIB телеграммы с адресами 1/2/0 - 1/2/9 пройдут через фильтр

Параметры OUTPUT:

bActive: блок был активирован

bReady: блок готов для отправки и получения данных

bError: бит ошибки

iErrorId: номер ошибки

str_Data_Rec: связан с передающими и принимающими блоками

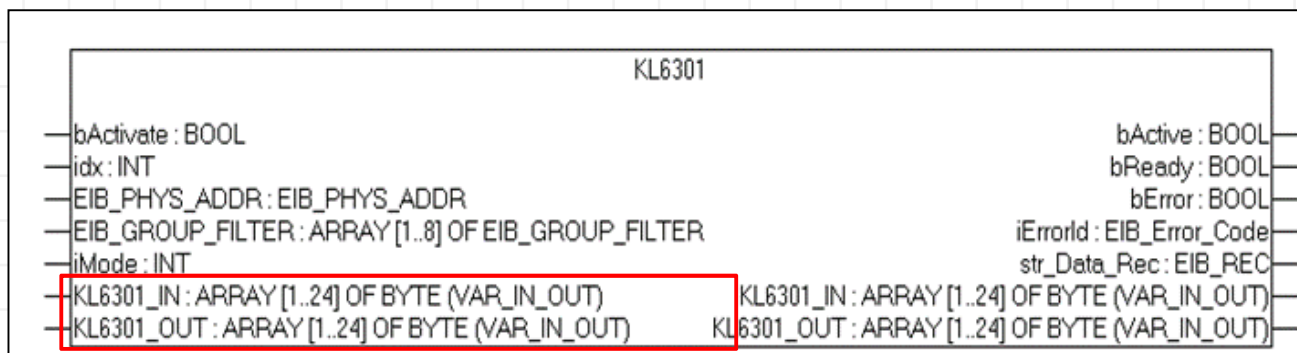
KL6301_IN: Связывается с входящими адресами KL6301

KL6301_OUT: Связывается с исходящими адресами KL6301

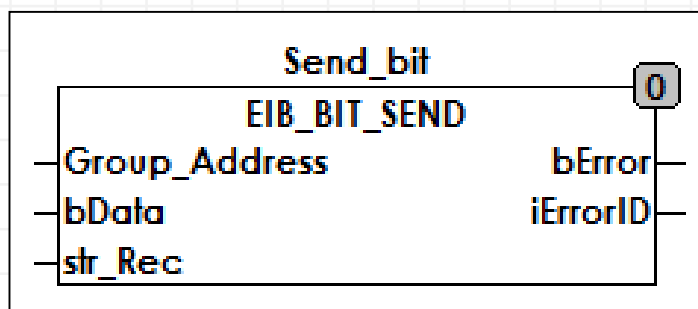
TwinCAT PLC Control – вызов и параметрирование функциональных блоков

3. Пошаговая инструкция по реализации проекта

1. Запускаем *TwinCAT PLC Control*
2. Создаем новый проект
3. Выбираем тип контроллера, под который будем писать программу
4. Указываем язык, на котором будем писать программу, нажимаем *OK*
5. Сохраняем проект
6. Через *Library Manager* добавляем в проект библиотеку TcEib.lib
7. Добавляем при помощи input assistant(клавиша F2 при нахождении курсора на рабочей области) функциональный блок KL6301
8. В соответствии с указанным ранее описанием функционального блока KL6301, ему при вызове необходимо передать ряд параметров: указать уникальный **idx**, и указать переменные, соответствующие входным и выходным данным. Также для настройки модуля KL6301 необходимо указать физический адрес (должен отличаться от физических адресов других устройств в сети EIB), а также групповые фильтры (минимум 1) – сделать это нужно до начала запуска программы.



9. Для отправки битовых значений вызовем функциональный блок *EIB_BIT_SEND*



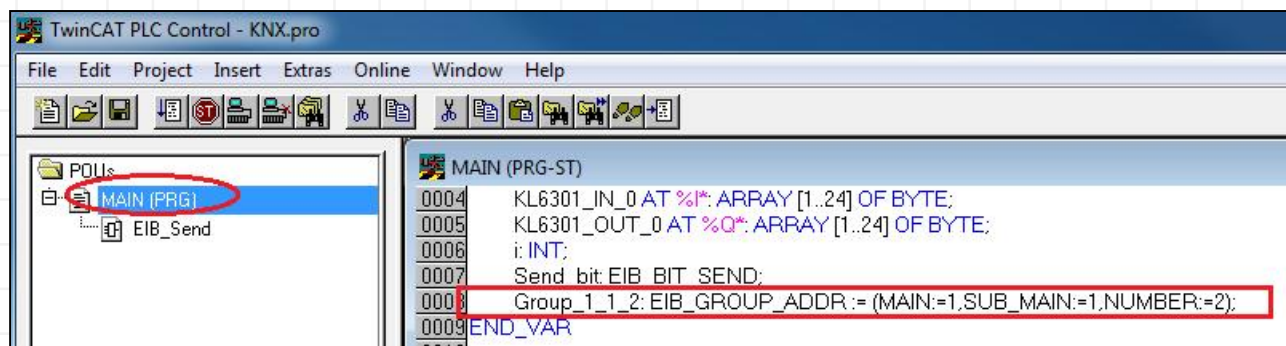
Для того, чтобы экземпляр *EIB_BIT_SEND* работал правильно, ему на вход необходимо подать следующие данные:

Group_Address- адрес

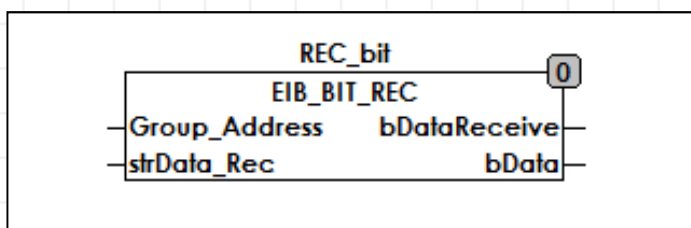
bData – битовые данные(можно указать переменную битовую или сразу значение)

str_Rec – структура данных из блока KL6301

Стоит заметить, что переменная *Group_Address* должна иметь тип *EIB_GROUP_ADDR* и ее еще нужно в явном виде указать.

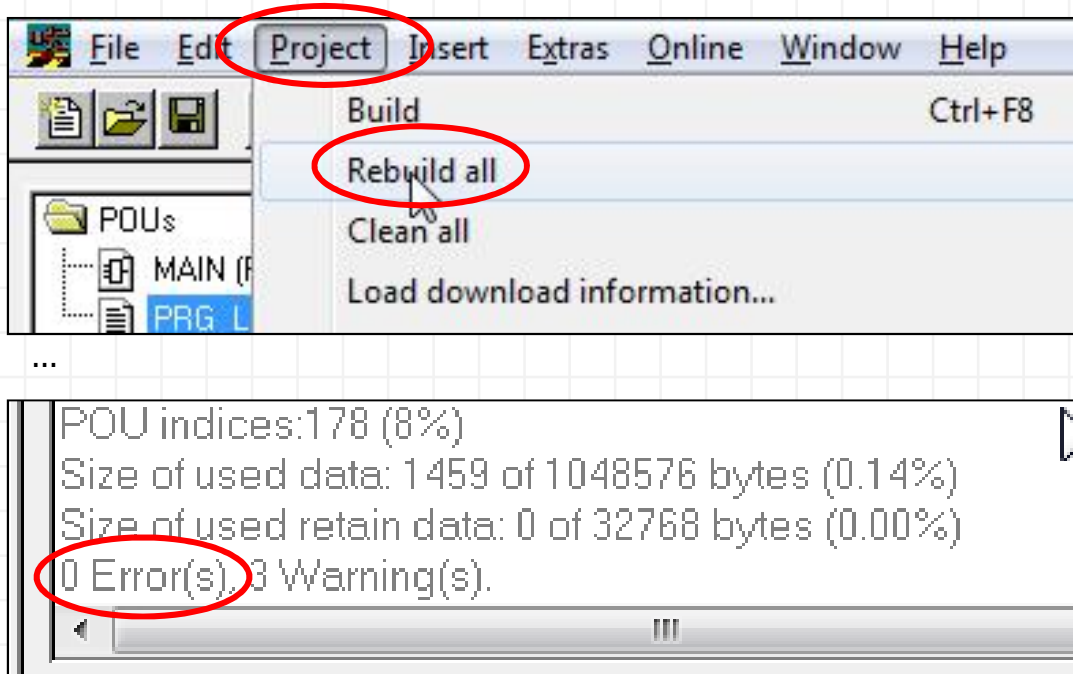


- Аналогичным образом вызываем функциональный блок *EIB_BIT_REC*.
Параметрируем его также, как и в предыдущем случае,.



Мы реализовали механизм получения/посылки битовых данных по KNX/EIB через модуль KL6301. Для пересылки других типов данных необходимо использовать соответствующие функциональные блоки. Также мы указали в PLC программе физический адрес для модуля KL6301 и привели примеры фильтров.

11. Скомпилируйте проект ¹⁾



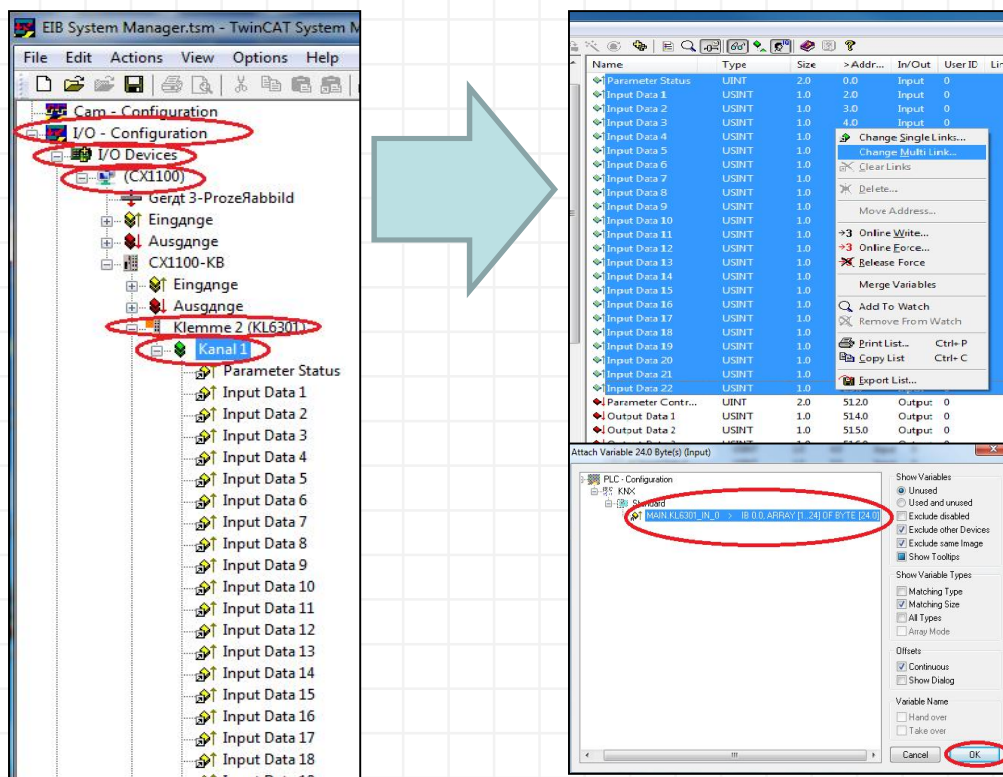
Ошибок быть не должно

12. Запустите TwinCAT System Manager ¹⁾
13. Создайте новый пустой проект ¹⁾
14. Выполните сохранение проекта ¹⁾
15. Путь к проекту не должен содержать кириллицу ¹⁾
16. Имя проекта не должно содержать кириллицу ¹⁾
17. Выполните поиск устройства ¹⁾
18. Добавьте устройство в роутер ПК ¹⁾
19. Выберите добавленный контроллер как целевое устройство ¹⁾

¹⁾ Подробное описание см. в руководстве "TwinCAT 2: Первые шаги программирования контроллера CX"

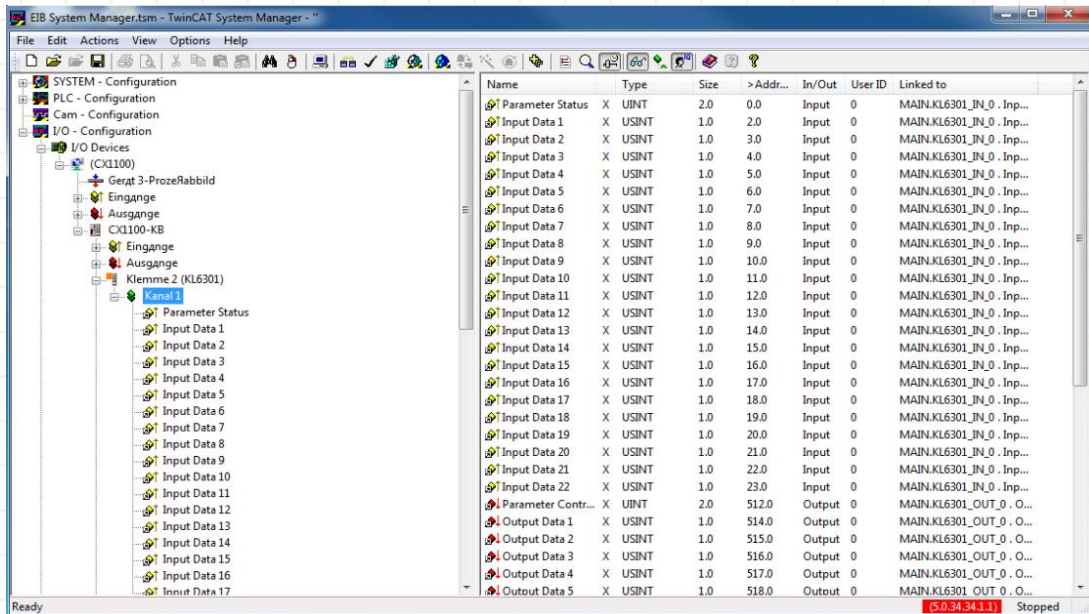
TwinCAT System Manager – выбор целевого устройства

20. Убедитесь, что Вы выбрали правильное целевое устройство ¹⁾
21. Убедитесь, что у Вас есть связь с выбранным целевым устройством ¹⁾
22. Переведите целевое устройство в режим Config Mode ¹⁾
23. Выполните сканирование модулей ввода/вывода ¹⁾
24. Выберите вкладку PLC-Configuration ¹⁾
25. Найдите файл *.tpy, который создается автоматически после компиляции проекта ПЛК без ошибок ¹⁾
26. Убедитесь, что во вкладке PLC-Configuration появился раздел, одноименный с именем файла *.tpy ¹⁾
27. Раскройте модуль KL6301 и встаньте курсором на Channel 1, выделите все входы, щелкните правой кнопкой мыши и выберите пункт Change multilink. В открывшемся окне выберите нужный массив



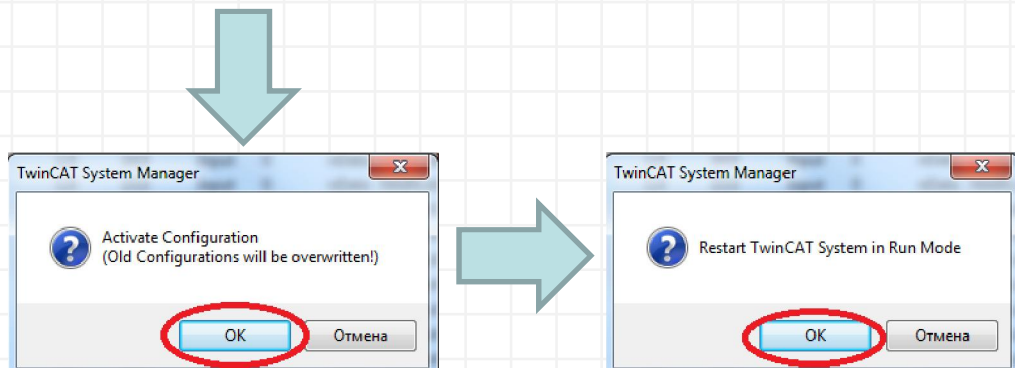
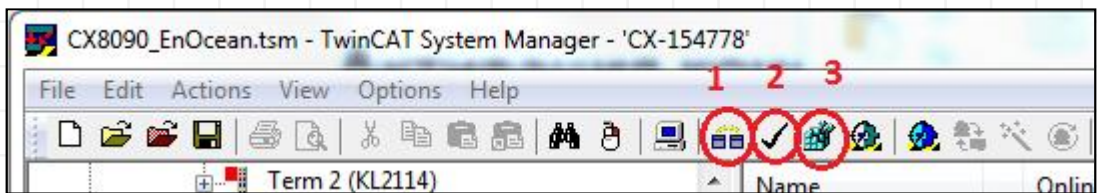
¹⁾ Подробное описание см. в руководстве "TwinCAT 2: Первые шаги программирования контроллера CX"

28. Аналогично нужно сделать с выходными переменными



29. После привязки всех переменных необходимо нажать

- 1) "Generate mappings"
- 2) "Check Configuration"
- 3) "Activate Configuration"



30. Вернитесь в TwinCAT PLC Control ¹⁾
31. Выберите Choose Run-Time System ¹⁾
32. Выберите контроллер и Порт ¹⁾
33. Выполните подключение к ПЛК и загрузите программу ¹⁾
34. Запустите выполнение программы ¹⁾
35. Программирование завершено

В электронную версию инструкции вложены результирующие файлы проектов TwinCAT PLC Control и TwinCAT System Manager, на примере которых рассматривалась данная инструкция.



KNX.tsm



KNX.pro

Результирующие файлы в архиве доступны для скачивания с сайта по этому QR-коду



Руководство "TwinCAT 2: Первые шаги программирования контроллера CX" доступно для скачивания с сайта по этому QR-коду



Данная инструкция является частью общего документа "Практическое пособие по программированию контроллеров Beckhoff", электронную версию которого можно скачать с сайта по этому QR-коду

¹⁾ Подробное описание см. в руководстве "TwinCAT 2: Первые шаги программирования контроллера CX"