

Конфигурирование корзины submodule ПЛК-40



Содержание

[Создание нового проекта в CODESYS V3.5 и настройка параметров устройства](#)

[Добавление нового устройства в дерево проекта](#)

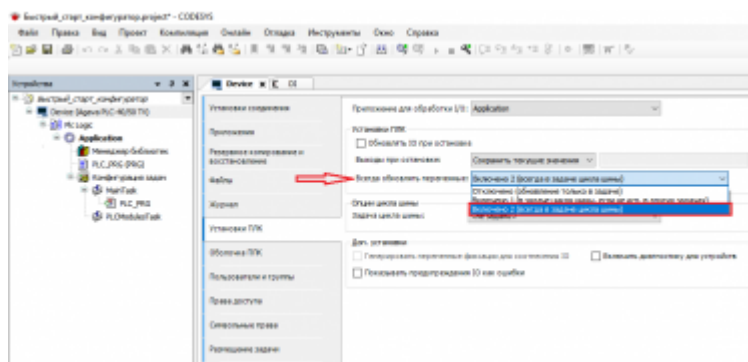
[Подключение submodule к слотам корзины ПЛК-40](#)

[Соотнесение входов и выходов с каналами submodule](#)

1 Создание нового проекта в CODESYS V3.5 и настройка параметров устройства

Рассмотрим пример создания проекта с использованием ПЛК-40, содержащего submodule дискретных входов, релейных выходов, аналоговых входов и входов термосопротивлений. В качестве примера конфигурации возьмём submodule **DI, R, AI, TMP**.

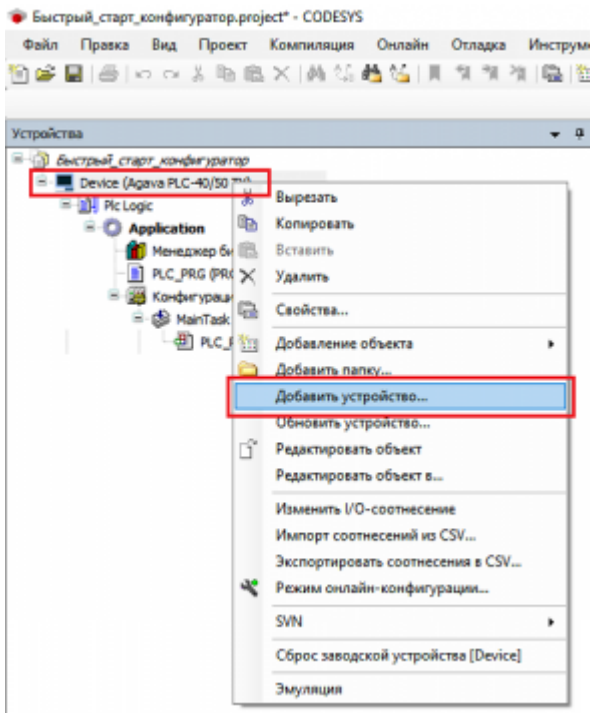
Создадим стандартный проект, в среде CODESYS 3.5 и подключим его к контроллеру. Двойным кликом мыши по **Device(Agava PLC-40/50 TV)** откроем настройки устройства и выберем пункт **«Установки ПЛК»**, опции **«Всегда обновлять переменные:»** зададим параметр **«Включено 2(всегда в задаче цикла шины)»**.



Настройка параметров устройства

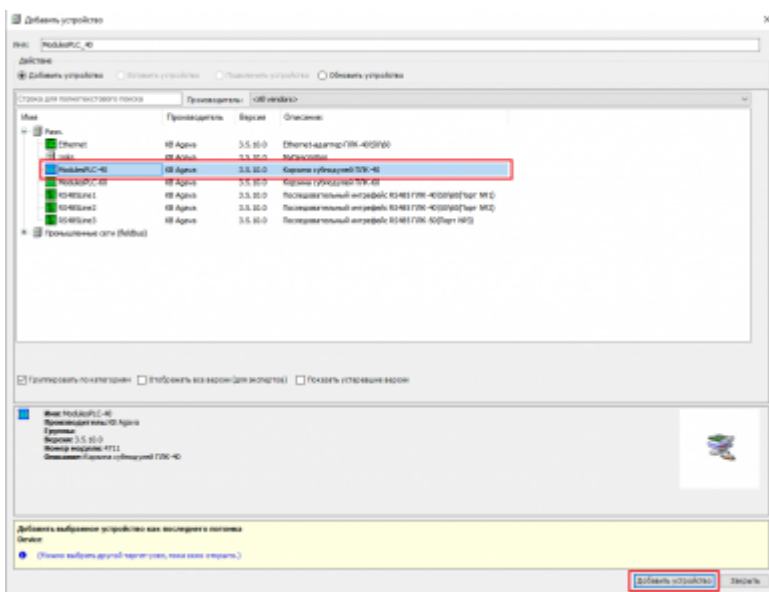
2 Добавление нового устройства в дерево проекта

Для добавления в проект корзины submodule ПЛК-40, кликнем правой клавишей мыши по устройству **Device(Agava PLC-40/50 TV)** и в открывшемся контекстном меню выберем пункт **«Добавить устройство»**.

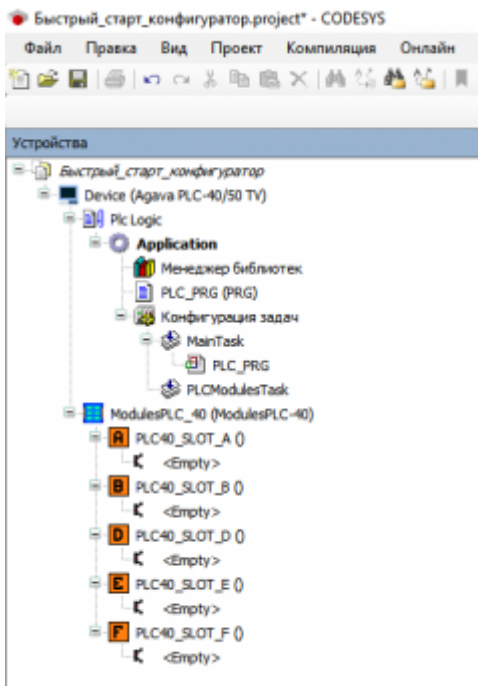


Добавление нового устройства

В открывшемся окне выбираем группу «Разн.», далее выбираем пункт из списка **ModulesPLC-40**, нажимаем кнопку «Добавить устройство». После того как устройство добавлено в дерево проекта, нажимаем кнопку «Заккрыть».



Выбор нового устройства



Добавленное новое устройство в
дерево проекта

3 Подключение submodule к слотам корзины ПЛК-40

После того как корзина submodule ПЛК-40 добавлена в дерево проекта, можно приступить к конфигурации submodule. Текущее расположение submodule в корзине ПЛК-40, в тестовой конфигурации имеет следующий порядок:

СЛОТ А - Submodule **DI**

СЛОТ В - Submodule **R**

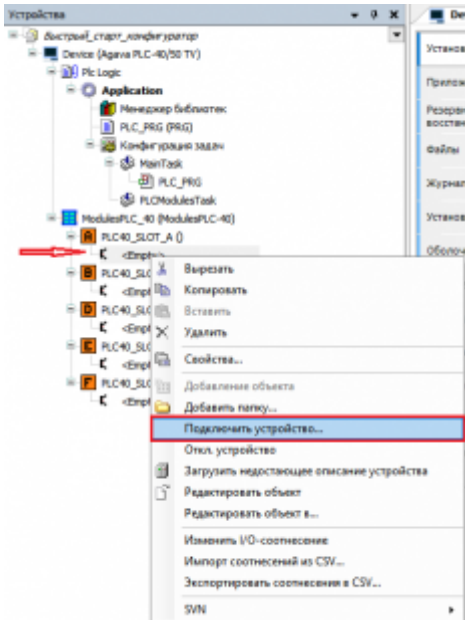
СЛОТ С - Блок питания

СЛОТ D - Пустой слот

СЛОТ E - Submodule **TMP**

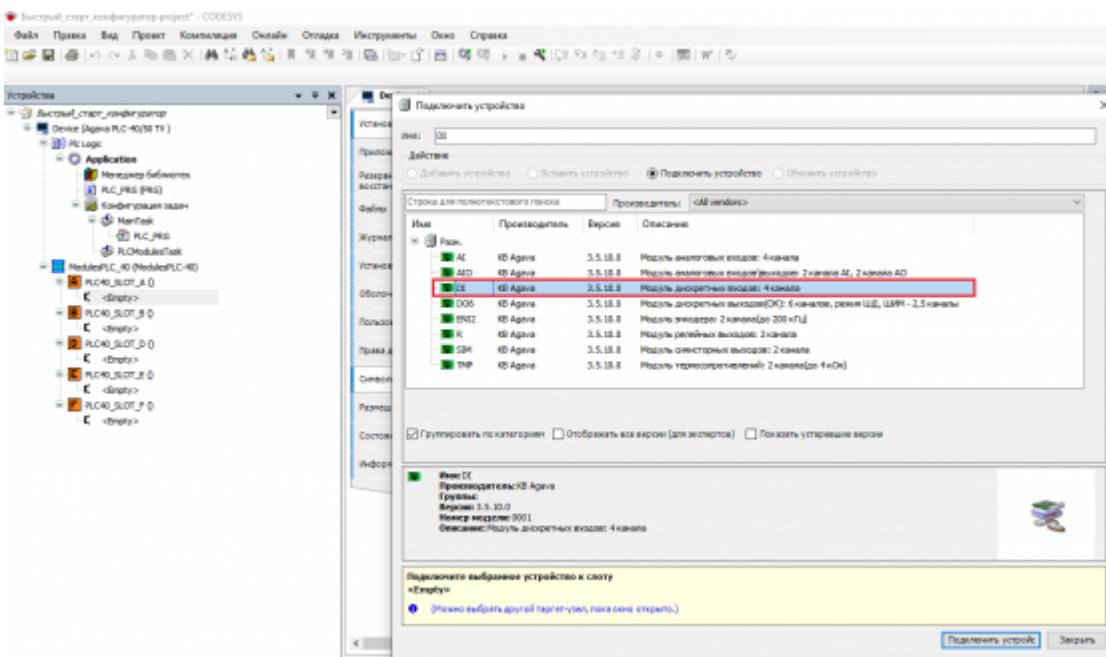
СЛОТ F - Submodule **AI**

Для подключения submodule к слоту в корзине ПЛК-40 кликнем правой клавишей мыши по слоту, в открывшемся контекстном меню выберем пункт «**Подключить устройство**»

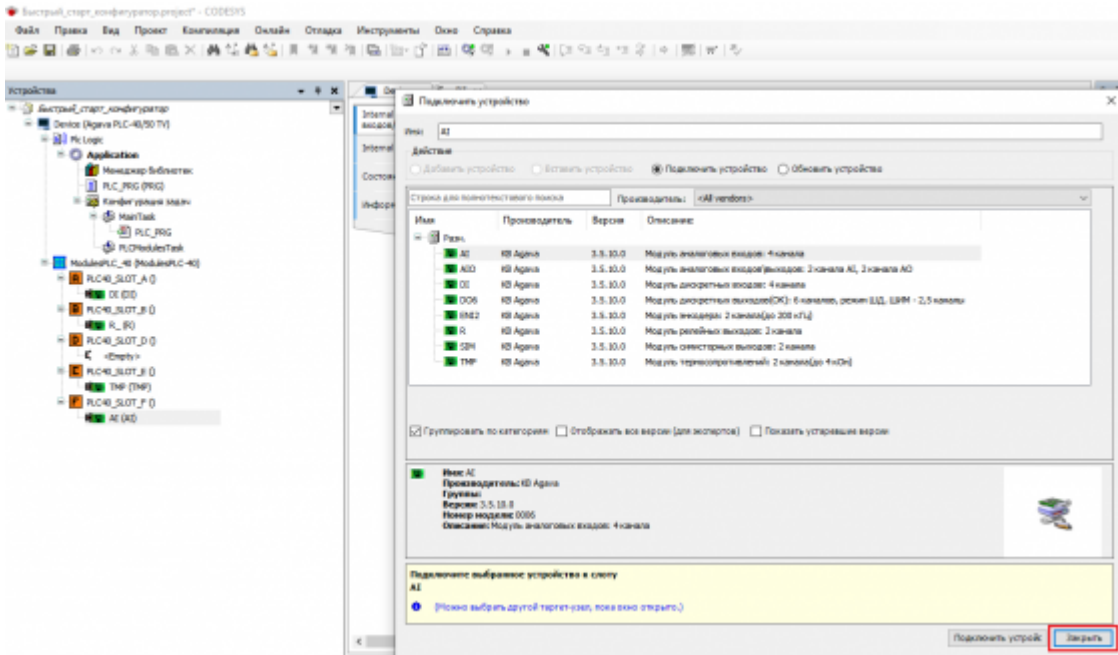


Подключение submodule

In the opened window, we will specify the required submodule type, in our case this is **DI**, then by a double mouse click we will connect the submodule. In an analogous way, without closing the window, we will specify the required submodules and connect them to the slots.



Выбор submodule дискретных входов

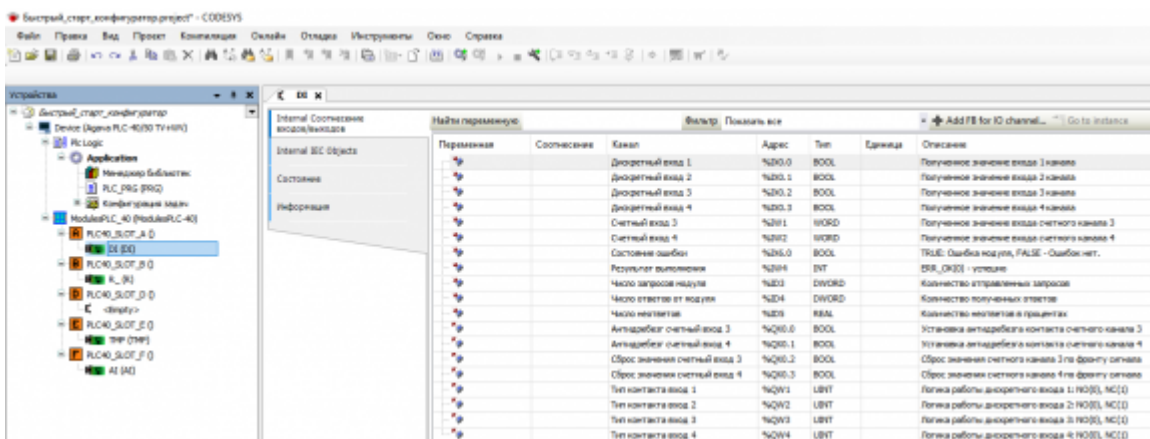


Подключенные к слотам корзины submodule

После завершения конфигурации закроем окно по нажатию кнопки «**Закрыть**».

4 Соотнесение входов и выходов с каналами submodule

После того как формирование корзины ПЛК-40 завершено, можно приступить к настройке submodule. Для настройки submodule **DI** дважды кликнем по нему левой клавишей мыши и выберем пункт «**Internal Соотнесение входов/выходов**».



Настройка submodule DI

Столбец «**Переменная**» позволяет создать соотнесение переменной проекта с каналом submodule. В качестве теста можно создать переменную нужного типа и сделать соотнесение с каналом submodule или использовать готовую структуру сигнала **TSensorsStruct**. В данном примере будет рассмотрено использование структуры **TSensorsStruct**.

```

// Структура описания аналогового/дискретного датчика
type TSensorsStruct :
struct

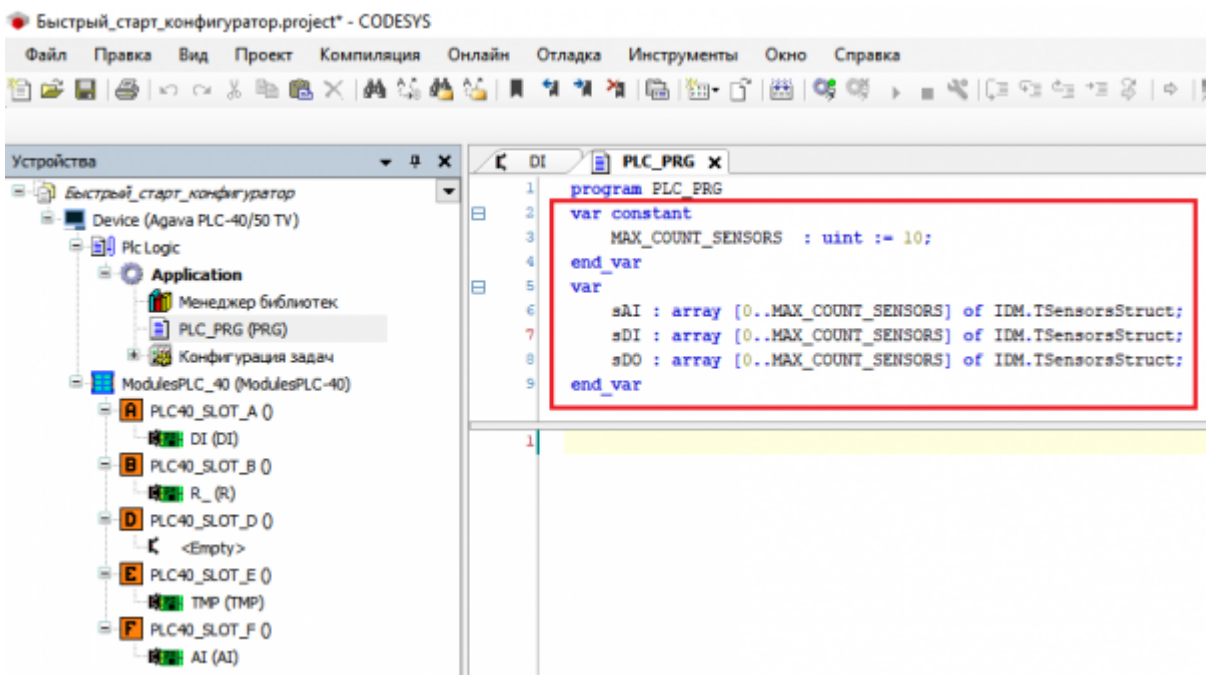
    id:          uint:          // Идентификатор сигнала
    Name:        wstring:       // Имя сигнала
    ShortName :  wstring:       // Краткое обозначение
    Unit:        wstring:       // Размерность
    bError:      bool:          // Флаг наличия ошибки объекта
    channel:    wstring:        // Описание канала. Например (DI10.A.X.1.1)
    AiType:      TSensTypeAI:   // Тип сигнала AI
    TmpType:     TSensTypeTMP:  // Тип сигнала TMP
    LogicType:   TSensTypeDI:   // Тип сигнала DI/DO
    MaxLimValue: real:          // Верхний предел измерения сигнала
    MinLimValue: real:          // Нижний предел измерения сигнала
    ErrorId:     uint:          // Код ошибки сигнала
    rValue:      real:          // Значение сигнала (float)
    bValue:      bool:          // Значение сигнала (bool)
    tImp:        int:           // Время импульса #ИМ(мс)
    Cnt3 :       uint:          // Светлый код канала #3 субмодуля DI
    Cnt4 :       uint:          // Светлый код канала #4 субмодуля DI
    Deb3 :       bool:          // #флаг установки антидребезга светлого канала #3 субмодуля DI
    Deb4 :       bool:          // #флаг установки антидребезга светлого канала #4 субмодуля DI
    ResCnt3 :    bool:          // Сброс светлого светлого канала #3 субмодуля DI
    ResCnt4 :    bool:          // Сброс светлого светлого канала #4 субмодуля DI

end_struct
end_type

```

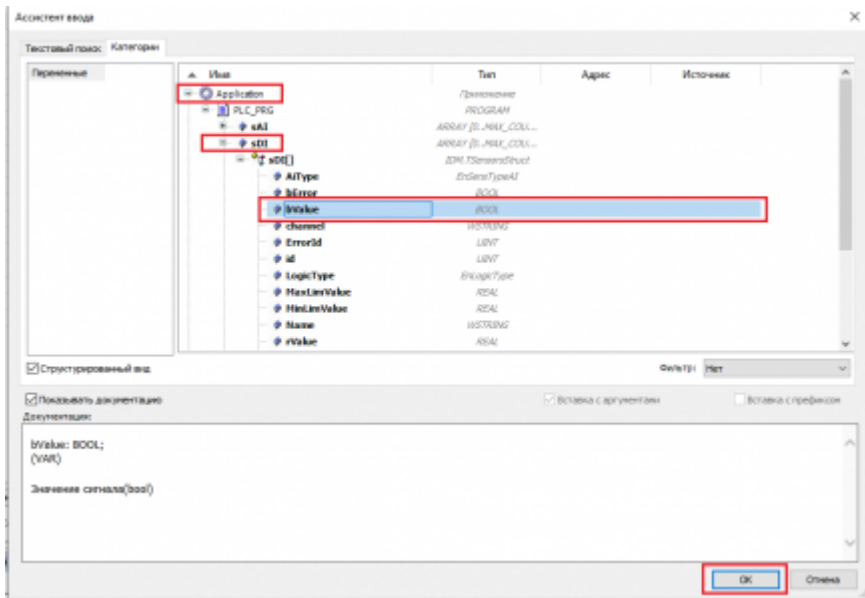
Описание структуры TSensorsStruct

Объявим необходимые экземпляры структур **TSensorsStruct** в программе **PLC_PRG**.



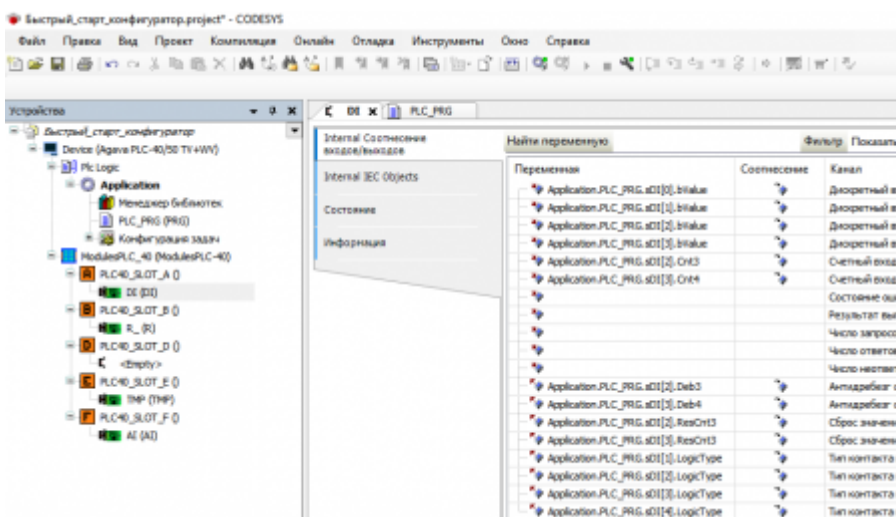
Объявление экземпляров структуры TSensorsStruct в PLC_PRG

В окне настроек сумодуля **DI** выполним соотнесение структуры с каналами. Для этого кликнем левой клавишей мыши в поле столбца «**Переменная**» и нажмём кнопку с тремя точками, в открывшемся окне выберем **Application -> sDI -> bValue** и нажмем **ОК**.



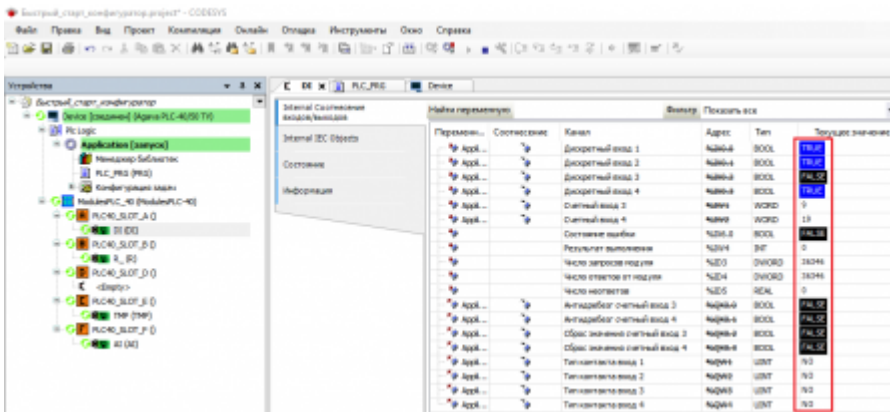
Соотнесение переменной структуры с каналом модуля

Так как объявленные переменные представлены массивом, указываем соответствующий индекс нужного элемента массива.



Завершение соотнесения переменных с каналами субмодуля

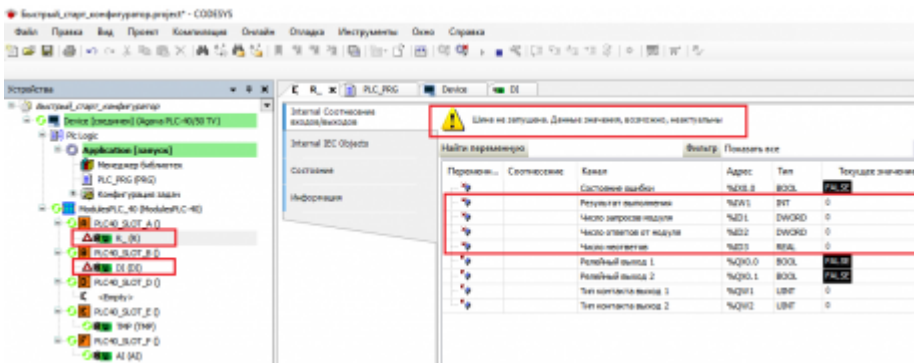
Компилируем (F11), загружаем (Alt+F8) и выполняем проект (F5), зелёные индикаторы на против субмодуля сигнализируют об успешном обмене, статистика обмена отображает текущее значение отправленных и полученных данных, а также количество ошибок(число неответов в процентном соотношении) . При замыкании дискретного входа мы видим, что сигнал получен, а соотнесение передаст значение сигнала переменной в проекте.



Проверка состояния субмодуля в режиме отладки

Индикация обмена на против каждого субмодуля позволяет оперативно определить проблему работы субмодуля, например если пользователь перепутал расположение субмодулей или установил в корзину не существующий субмодуль, то напротив проблемного субмодуля будет изображён красный треугольник.

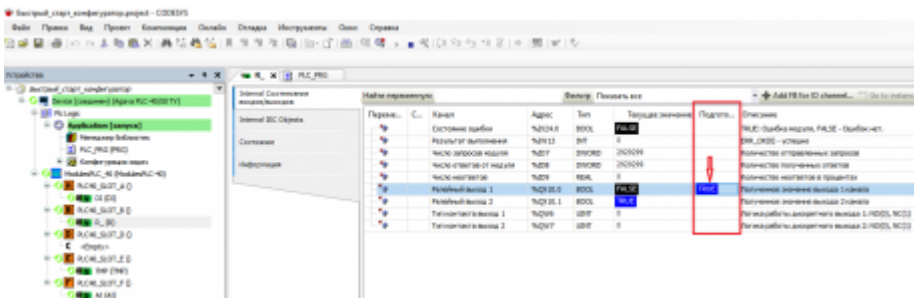
Ниже показан пример ошибочной установки субмодулей в слоты А и В.



Некорректная установка модуля в слот

Для управления релейными выходами достаточно реализовать соотнесение переменной с каналами субмодуля **R**.

Для проверки исправности работы выходов без соотнесения переменной, в столбец «Подготовленное значение» требуется установить значение **TRUE**, для этого необходимо кликнуть левой клавишей мыши в поле столбца на против канала модуля и нажать (Ctrl+F7).

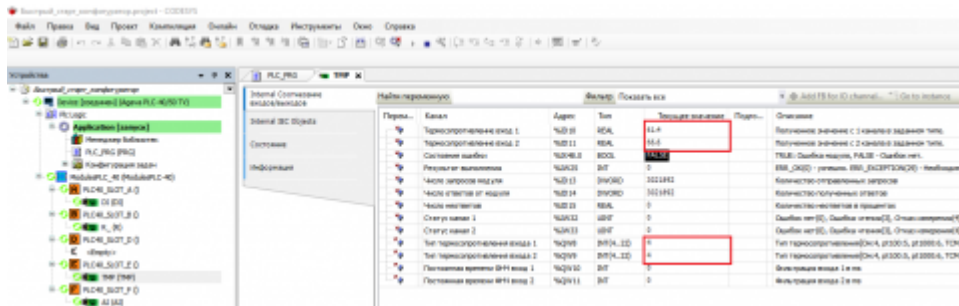


Проверка срабатывания релейного выхода

При необходимости можно реализовать инверсную логику срабатывания выхода, для этого в поле «Тип контакта выхода 1» нужно установить значение 1(NC) , тогда

релейный выход будет работать инверсно и по умолчанию будет иметь замкнутый контакт. Аналогичная настройка также имеется у submodule **DI**.

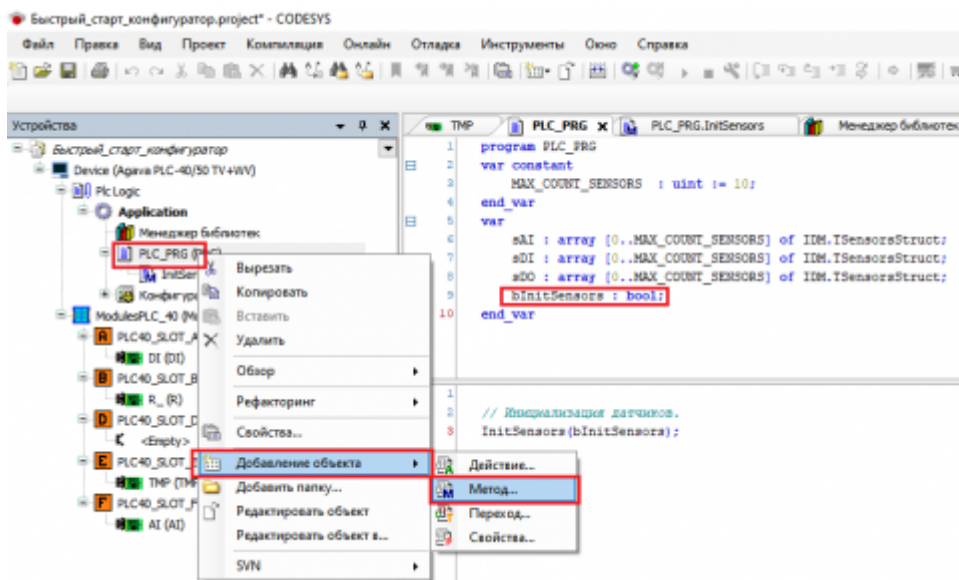
Submodule **TMP** по умолчанию отображает значение в **Om**, в качестве теста к каналам модуля подключены сопротивления номиналом 61.4 и 66.6 **Om**, в полях «**Тип термосопротивления входа 1**» и «**Тип термосопротивления входа 2**» указывается тип датчика. После установки нужного типа датчика, значение канала будет отображать температуру в градусах Цельсия.



Проверка чтения каналов submodule TMP

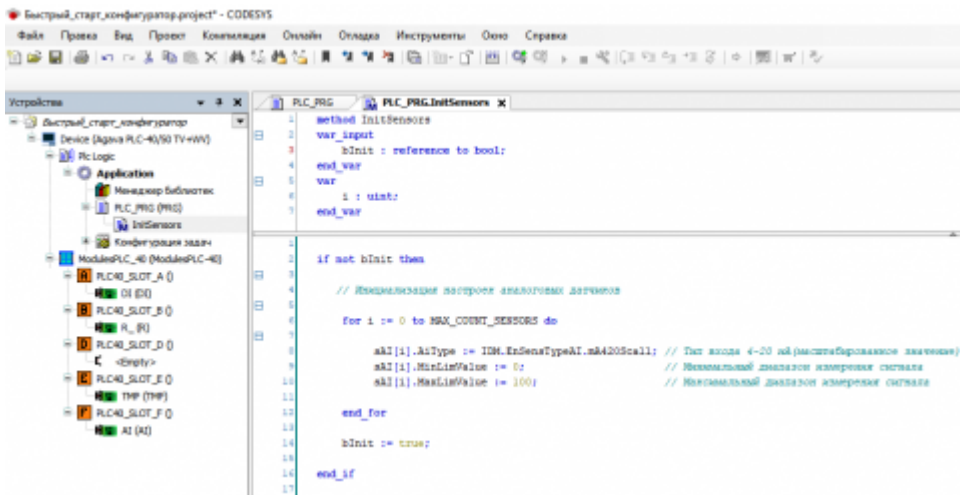
Submodule **AI** имеет четыре универсальных конфигурируемых канала, рассмотрим вариант настройки датчиков помощью структуры **TSensorsStruct**.

В программу **PLC_PRG** добавим метод инициализации датчиков, для этого кликнем **ПКМ** по программе **PLC_PRG** из контекстного меню выберем пункт «**Добавление объекта**», далее «**Метод**», зададим название метода **InitSensors**, возвращаемое значение оставим пустым и нажмём кнопку «**Добавить**». В программе объявим новую переменную **bInitSensors**.



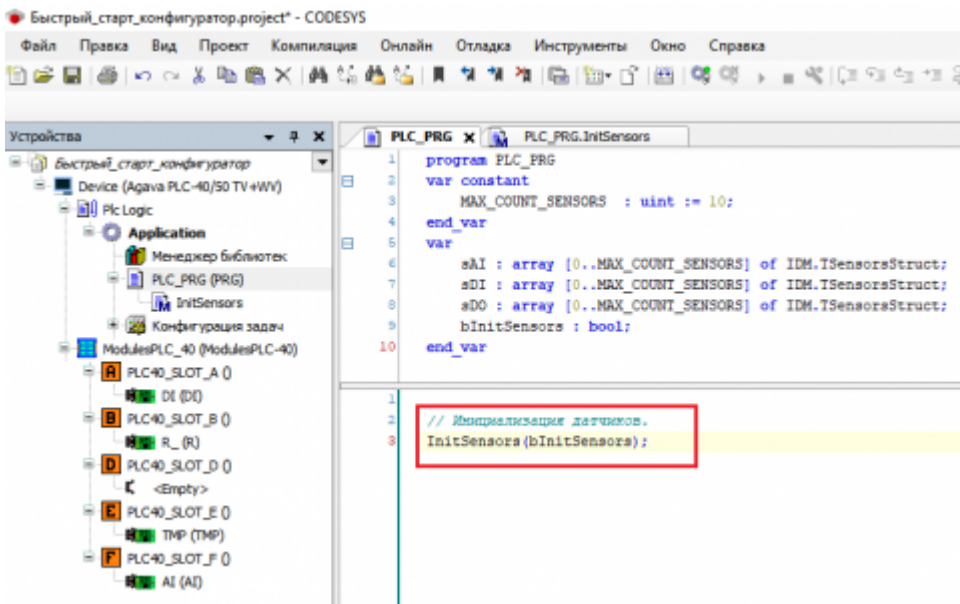
Добавление метода

Опишем реализацию метода как показано ниже.



Описание реализации метода

Добавим вызов метода в основной программе **PLC_PRG**.



Добавление вызова метода в основной программе

Далее производим соотнесение переменных структуры с каналами субмодуля **AI**, компилируем (F11), загружаем (Alt+F8) и выполняем проект (F5).

Первый канал субмодуля считывает тестовое значение задатчика тока равное **20 мА**, тип значения автоматически пересчитывается по установленному диапазону датчика, мы можем увидеть полученное значение **100**.

Статус первого канала отображает код ошибки равный нулю, что свидетельствует об успешном чтении значения сигнала. Расшифровка кодов ошибок каналов представлена в столбце «**Описание**».

Быстрый_старт_конфигуратор.проект - CODESYS

Быстрый_старт_конфигуратор

PLC_PRG

PLC_PRG.InstSensors

Имя переменной

Соединение

Канал

Адрес

Тип

Текущее значение

Единица

Описание

Имя переменной	Соединение	Канал	Адрес	Тип	Текущее значение	Единица	Описание
Application.PLC_PRG.sAI[0].rValue		Аналоговый вход 1	%DI17	REAL	100.250008		Получен
Application.PLC_PRG.sAI[1].rValue		Аналоговый вход 2	%DI18	REAL	-24.9275		Получен
Application.PLC_PRG.sAI[2].rValue		Аналоговый вход 3	%DI19	REAL	-25		Получен
Application.PLC_PRG.sAI[3].rValue		Аналоговый вход 4	%DI20	REAL	-25		Получен
		Состояние ошибки	%DI14.0	BOOL	FALSE		TRUE: 0
		Результат выполнения	%DI14.1	INT	0		ERR_OK
		Число запросов модуля	%DI22	DWORD	125991		Квантис
		Число ответов от модуля	%DI23	DWORD	125991		Квантис
		Число неответов	%DI24	REAL	0		Квантис
Application.PLC_PRG.sAI[0].ErrorId		Статус канал 1(AI)	%DI19	UINT	0		Ошибки
Application.PLC_PRG.sAI[1].ErrorId		Статус канал 2(AI)	%DI19	UINT	1		Ошибки
Application.PLC_PRG.sAI[2].ErrorId		Статус канал 3(AI)	%DI19	UINT	1		Ошибки
Application.PLC_PRG.sAI[3].ErrorId		Статус канал 4(AI)	%DI19	UINT	1		Ошибки
		Типы аналоговых входов	%Q06	REAL	0		Настрой
Application.PLC_PRG.sAI[0].AIType		Тип аналогового входа 1	%Q06.0	INT	AI420Scal		Тип входа
Application.PLC_PRG.sAI[0].MinLimitValue		Нижний предел	%Q07	REAL	0		Нижний л
Application.PLC_PRG.sAI[0].MaxLimitValue		Верхний предел	%Q08	REAL	100		Верхний л
Application.PLC_PRG.sAI[1].AIType		Тип аналогового входа 2	%Q06.1	INT	AI420Scal		Тип входа
Application.PLC_PRG.sAI[1].MinLimitValue		Нижний предел	%Q07.1	REAL	0		Нижний л
Application.PLC_PRG.sAI[1].MaxLimitValue		Верхний предел	%Q08.1	REAL	100		Верхний л
Application.PLC_PRG.sAI[2].AIType		Тип аналогового входа 3	%Q06.2	INT	AI420Scal		Тип входа
Application.PLC_PRG.sAI[2].MinLimitValue		Нижний предел	%Q07.2	REAL	0		Нижний л
Application.PLC_PRG.sAI[2].MaxLimitValue		Верхний предел	%Q08.2	REAL	100		Верхний л
Application.PLC_PRG.sAI[3].AIType		Тип аналогового входа 4	%Q06.3	INT	AI420Scal		Тип входа
Application.PLC_PRG.sAI[3].MinLimitValue		Нижний предел	%Q07.3	REAL	0		Нижний л
Application.PLC_PRG.sAI[3].MaxLimitValue		Верхний предел	%Q08.3	REAL	100		Верхний л
		Вычитание входов(0FF-0)	%Q036				Вычита

Проверка чтения каналов submodule AI

Источник —

http://docs.kb-agava.ru/index.php?title=Конфигурирование_корзины_субмодулей_ПЛК-40&oldid=1700

Эта страница в последний раз была отредактирована 4 мая 2022 в 09:13.