



ЕАС

Контроллер программируемый ЭЛСИ-ТМК Построение подсистемы резервирования

РУКОВОДСТВО ПО ПРИМЕНЕНИЮ

часть 2

Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

2017

Литера

СОДЕРЖАНИЕ

1 ХАРАКТЕРИСТИКИ И УСТРОЙСТВО ПОДСИСТЕМЫ РЕЗЕРВИРОВАНИЯ ЭЛСИ-ТМК.....	6
1.1 НАЗНАЧЕНИЕ.....	6
1.2 ОСОБЕННОСТИ РЕЗЕРВИРОВАНИЯ В КОНТРОЛЛЕРЕ ЭЛСИ-ТМК	6
1.3 СТРУКТУРНАЯ СХЕМА ПОСТРОЕНИЯ РЕЗЕРВИРОВАННОГО КОНТРОЛЛЕРА	6
1.4 ТЕХНИЧЕСКИЕ ХАРАКТЕРИСТИКИ РЕЗЕРВИРОВАНИЯ В ЭЛСИ-ТМК	7
2 ОПИСАНИЕ РАБОТЫ ПОДСИСТЕМЫ РЕЗЕРВИРОВАНИЯ	9
2.1 ОСНОВНЫЕ ПОНЯТИЯ И ФУНКЦИИ ПОДСИСТЕМЫ РЕЗЕРВИРОВАНИЯ	9
2.2 АЛГОРИТМ РАБОТЫ ПОДСИСТЕМЫ РЕЗЕРВИРОВАНИЯ	9
2.3 ИНДИКАЦИЯ РАБОТЫ ПОДСИСТЕМЫ РЕЗЕРВИРОВАНИЯ	11
2.4 НАЗНАЧЕНИЕ ТРЕХПОЗИЦИОННОГО ПЕРЕКЛЮЧАТЕЛЯ SWITCH «123»	11
2.5 СОЗДАНИЕ ПРОЕКТА С ПОДДЕРЖКОЙ РЕЗЕРВИРОВАНИЯ В КОНТРОЛЛЕРЕ ЭЛСИ-ТМК.....	11
2.6 ИНФОРМАЦИОННЫЕ ПАРАМЕТРЫ УСТРОЙСТВА «REDUNDANCY».....	16
2.7 НАСТРОЙКА СИСТЕМНЫХ ПАРАМЕТРОВ УСТРОЙСТВА «REDUNDANCY».....	19
2.7.1 Параметры установки устройства «Redundancy».....	20
2.7.2 Соотнесение входов устройства «Redundancy».....	22
2.8 РАБОТА С ФБ «FBREDUNDANCY»	24
2.9 ПОДГОТОВКА К РАБОТЕ РЕЗЕРВИРУЕМЫХ КОНТРОЛЛЕРОВ	27
2.10 ОТЛАДКА ПРОЕКТА С ПОДДЕРЖКОЙ РЕЗЕРВИРОВАНИЯ В КОНТРОЛЛЕРЕ ЭЛСИ-ТМК	28
2.10.1 Первый этап отладки.....	28
2.10.2 Второй этап отладки.....	29
2.10.3 Третий этап отладки.....	33
2.10.4 Рекомендации по отладке проектов	33
3 ТЕХНИЧЕСКОЕ ОБСЛУЖИВАНИЕ	34
4 РЕШЕНИЕ ПРОБЛЕМ.....	35
СПИСОК ЛИТЕРАТУРЫ.....	36
ПРИЛОЖЕНИЕ А (СПРАВОЧНОЕ) СТРУКТУРНЫЕ СХЕМЫ ПОСТРОЕНИЯ РЕЗЕРВИРОВАННЫХ СИСТЕМ	37

Список терминов и сокращений

Термин/сокращение	Описание
ВВ	Ввод/Вывод
Крейт	Панель коммутационная
Модуль УВВ	Модуль удаленного ввода-вывода
ПЛК	Программируемый логический контроллер
ПО	Программное обеспечение
ПР	Подсистема резервирования
РЭ	Руководство по эксплуатации
ФБ	Функциональный блок
ЦП	Центральный процессор

Информация о документе

В настоящем руководстве по применению (далее – РП) содержится информация, необходимая пользователю для построения подсистемы резервирования контроллера ЭЛСИ-ТМК.

Данные, предоставленные в документе, проверены на соответствие аппаратному и программному обеспечению на момент поставки подсистемы резервирования контроллера ЭЛСИ-ТМК.

В связи с текущим совершенствованием продукции и документации, пользователю целесообразно следить за проводимыми обновлениями через сайт производителя.

Авторские права на настоящий документ принадлежат компании АО "ЭлеСи". Копирование и распространение настоящего документа без письменного разрешения владельца авторских прав запрещено.

Контактная информация:

- почтовый адрес: АО "ЭлеСи", 634021, г. Томск, ул. Алтайская, 161а;
- тел. (3822) 601-000, факс (3822) 601-001;
- официальный сайт компании: www.elesy.ru.

1 Характеристики и устройство подсистемы резервирования ЭЛСИ-ТМК

1.1 Назначение

Подсистема резервирования предназначена для резервирования работы контроллеров ЭЛСИ-ТМК.

1.2 Особенности резервирования в контроллере ЭЛСИ-ТМК

Под резервированием контроллера ЭЛСИ-ТМК следует понимать:

- **аппаратное резервирование источников питания**, при котором питание на крейт подается с одного из двух источников:
 - модуль питания TP 711 (TP 712);
 - резервное питание +24В, подаваемое на разъем питания коммутационной панели.

Выбор источника питания зависит от величины подаваемого на крейт напряжения, т.е. питание осуществляется от того источника, напряжение на котором выше.

Структурная схема подключения питания в системе с резервированием приведена на рисунке А.1 (Приложение А).

- **программное резервирование функциональности ЦП**. Структурная схема подключения модулей ЦП в системе с резервированием приведена на рисунке А.2 (Приложение А);
- **резервирование ввода/вывода** осуществляется дублированием модулей ввода/вывода в симметрично установленных двух одинаковых крейтах.

Структурная схема подключения модулей дискретного ввода в системе с резервированием приведена на рисунке А.3 (Приложение А).

Структурная схема подключения модулей дискретного вывода в системе с резервированием приведена на рисунке А.4 (Приложение А).

Подсистема резервирования (далее – ПР) является программным обеспечением, работающим в составе двух ПЛК, связанных между собой линиями резервирования по принципу «Основной-Резервный».

ПР служит для управления процессом резервирования, обеспечивая, тем самым, реализацию следующих возможностей:

- выполнение разных задач и конфигурирование разных проектов;
- резервирование данных задачи, указанных пользователем;
- переход на резервный контроллер по команде пользователя;
- автоматический переход на резервный контроллер при обнаружении отказа основного контроллера.

1.3 Структурная схема построения резервированного контроллера

На рисунке 1 приведена схема подключения модулей УВВ на базе ЦП без резервирования линии связи. На рисунке 2 приведена схема подключения модулей УВВ на базе ЦП с резервированием линии связи.

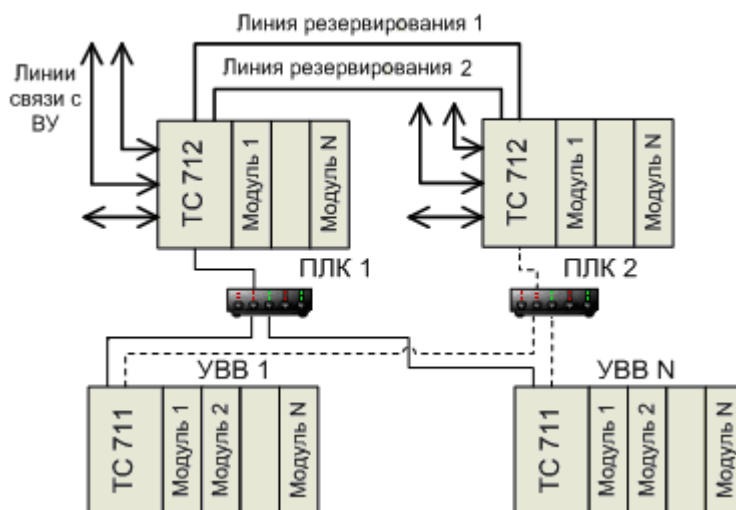


Рисунок 1 - Схема подключения резервируемого контроллера с УВВ без резервирования линии связи

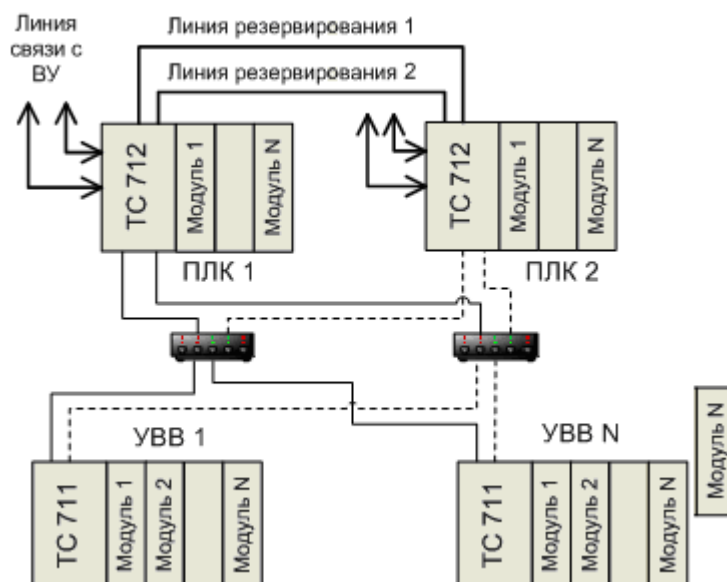


Рисунок 2 - Схема подключения резервируемого контроллера с УВВ с резервированием линий связи

1.4 Технические характеристики резервирования в ЭЛСИ-ТМК

ЭЛСИ_ТМК с поддержкой резервирования может работать со следующим списком оборудования:

- модуль ЦП TC712;
- состав функциональных модулей определяется потребностями конкретного проекта.

Временные характеристики резервирования представлены в Таблица 1 .

Таблица 1 – Временные характеристики резервирования

Описание	Значение
Максимально допустимое время автоматического перехода на резервный контроллер при отказе основного ПЛК, мс	50
Максимально допустимое время для перехода на резервный контроллер по команде пользователя с момента завершения выполнения задачи на основном контроллере, мс	80

Минимальный интервал запуска задачи, задаваемый пользователем, мс	100
Максимальный интервал запуска задачи, задаваемый пользователем, мс	10000
Максимально допустимое время на резервирование данных, указанных пользователем, с момента завершения выполнения задачи на основном контроллере, мс	80
Минимально допустимое время работы модуля ЦП при отказе двух источников питания либо пропадании питающего напряжения, мс	50
Максимально допустимое время продолжительности исполнения задачи соответствует разности интервала запуска задачи и максимального времени резервирования данных, указанных пользователем.	

Информационные ограничения резервирования (размеры контекстов и т.п.):

- максимальный размер данных, резервируемых на основном ПЛК, составляет 16 килобайт;
- максимальный размер данных, резервируемых на резервном ПЛК, составляет 1 килобайт;
- максимальное количество переменных, резервируемых в основном контексте, составляет 16 элементов;
- максимальное количество переменных, резервируемых в резервном контексте, составляет 16 элементов;
- максимальное количество задач резервирования соответствует 1 элементу.

2 Описание работы подсистемы резервирования

2.1 Основные понятия и функции подсистемы резервирования

К основным понятиям подсистемы резервирования относятся:

- **None** - состояние ПЛК, в котором ПР не запущена.
- **Active** - состояние ПЛК, в котором он управляет процессом, а второй контроллер находится в резерве.
- **Standby** - состояние ПЛК, в котором он находится в резерве, а второй контроллер управляет процессом.
- **Standalone** - состояние ПЛК, в котором он управляет процессом, а второй контроллер не работает.
- **Standbyblock** - состояние ПЛК, в котором он находится в резерве и заблокирован его переход в состояния **Active** и **Standalone**.
- **StandbyblockNoActive** - состояние ПЛК, в котором он находится в резерве и заблокирован его переход в состояние **Active**, а второй контроллер не работает.
- **Основной контекст** - резервируемые данные пользователя на ПЛК в роли **Active** или **StandAlone**.
- **Резервный контекст** - резервируемые данные пользователя на ПЛК в роли **Standby** или **StandbyBlock**.

Подсистема резервирования обеспечивает выполнение следующих функций:

- автоматическое распределение ролей между двумя ПЛК;
- запуск задачи пользователя;
- резервирование данных задачи пользователя;
- смена роли ПЛК по команде пользователя из конфигурации проекта;
- смена роли ПЛК по команде пользователя из задачи;
- управление индикаторами «Основной» и «Резервный»;
- формирование диагностической информации.

2.2 Алгоритм работы подсистемы резервирования

ПР запускается по старту приложения. При запуске ПР определяет, какой из двух ПЛК будет активным (Active), а какой - ожидающим (Standby).

Активный ПЛК запускает задачу пользователя с заданным интервалом времени, по завершении которой передает зарегистрированный контекст (Основной контекст) данных ожидающему контролеру. Резервный контроллер, получив контекст данных от основного контроллера, запускает задачу пользователя, по завершению которой передает свой зарегистрированный контекст (Резервный контекст) данных обратно активному ПЛК, как это показано на Рисунок 3 –Принцип работы подсистемы резервирования

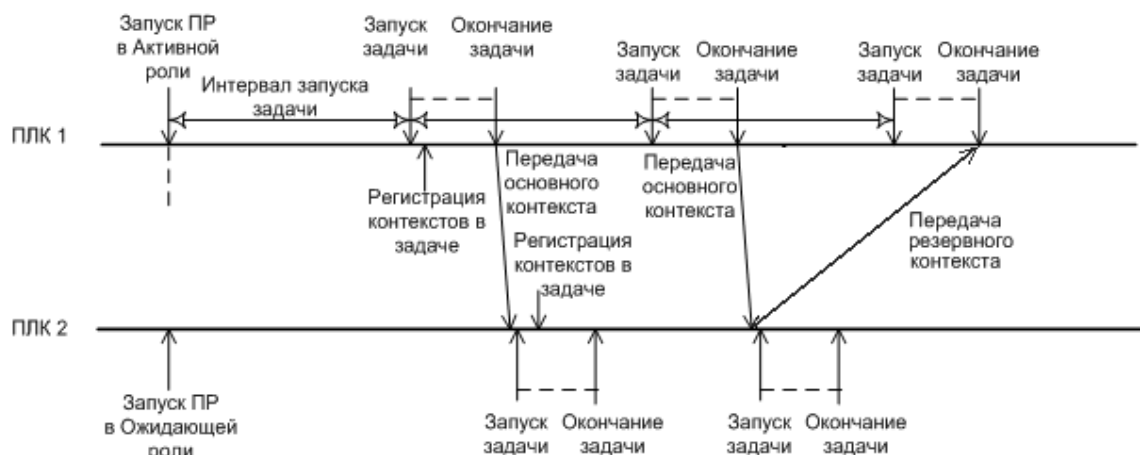


Рисунок 3 –Принцип работы подсистемы резервирования

Ниже представлены состояния и условия перехода подсистемы резервирования:

None → *Active* - только по старту приложения.

None → *Standby* - только по старту приложения.

None → *Standalone* - только по старту приложения.

Active → *Standby* - из задачи и из конфигурации по команде пользователя.

Active → *Standalone* - при потере связи с удаленным ПЛК.

Active → *None* - после сброса.

Standby → *Active* - из задачи и из конфигурации по команде пользователя.

Standby → *Standalone* - при потере связи с удаленным ПЛК.

Standby → *Standbyblock* - из задачи и из конфигурации по команде пользователя.

Standby → *None* - после сброса.

Standalone → *Active* - при восстановлении связи с удаленным ПЛК.

Standalone → *Standby* - при восстановлении связи с удаленным ПЛК.

Standalone → *None* - после сброса.

Standbyblock → *Standby* - из задачи и из конфигурации по команде пользователя.

Standbyblock → *StandbyblockNoActive* - при потере связи с удаленным ПЛК.

Standbyblock → *None* - после сброса.

StandbyblockNoActive → *Standbyblock* - при восстановлении связи с удаленным ПЛК.

StandbyblockNoActive → *Standalone* - из конфигурации по команде пользователя.

StandbyblockNoActive → *None* - после сброса.

2.3 Индикация работы подсистемы резервирования

Индикация процесса резервирования представлена в Таблица 2.

Таблица 2 – Работа индикаторов "Основной" и "Резервный"

Режим работы	Состояние индикатора «Основной»	Состояние индикатора «Резервный»	Описание режима работы
None	потушен	потушен	ПР не запущена
Active	непрерывно	мигание с периодом 1сек	Модуль находится в состоянии <i>Active</i> . Резервный контроллер находится на связи
StandAlone	непрерывно	Потушен	Модуль находится в состоянии <i>StandAlone</i> . Нет связи с резервным контроллером (выключен или неисправен)
Standby	мигание с периодом 1 сек	непрерывно	Модуль находится в состоянии <i>Standby</i> . Основной контроллер находится на связи и осуществляет обновление контекста резервируемых данных
StandbyBlock	мигание с периодом 1 сек	мигание с периодом 1сек	Заблокирован переход в состояние <i>Active</i> . Модуль находится в состоянии <i>Standby</i> , есть связь со вторым контроллером
StandbyBlockNoActive	потушен	мигание с периодом 1сек	Заблокирован переход в состояние <i>Active</i> . Модуль находится в состоянии <i>Standby</i> , нет связи со вторым контроллером

2.4 Назначение трехпозиционного переключателя Switch «123»

Трехпозиционный переключатель задает режимы работы основного и резервного ПЛК. Соотношения позиций и режимов:

- Позиция «1» – работа контроллера в режиме *Main (Основной)*.
- Позиция «2» – работа контроллера в режиме *Reserv (Резервный)*.
- Позиция «3» – работа контроллера в режиме *Maintance* (режим программирования и обслуживания). В данном режиме можно не только загружать ПО и значение ключа, но и использовать его как рабочий режим, при этом ПР считывается из файла проекта.

2.5 Создание проекта с поддержкой резервирования в контроллере ЭЛСИ-ТМК

Для создания проекта с поддержкой ПР необходимо:

1. Запустить систему программирования *CoDeSys* двойным щелчком левой кнопки мыши по ярлыку «*EleSy CODESYS V3.5 SP6 Patch 1*».

Примечание - Предполагается, что система программирования *CoDeSys* уже установлена.

2. В меню **Файл** выбрать команду «Новый проект ...».
3. В диалоговом окне «Новый проект...» в списке шаблонов выбрать «Стандартный проект»;
4. Ввести имя проекта и указать путь к каталогу, в котором файл проекта будет сохранен;
5. Нажать кнопке «ОК»;
6. В появившемся окне «Стандартный проект» в списке «Устройство» выбрать контроллер «*ELSYTMK_A8_2ETH (Elesy Company)*» (рисунок 4);

7. В списке «**PLC_PRG на:**» выбрать язык реализации основного программного компонента – «*Структурированный текст (ST)*» (рисунок 4);
8. Нажать кнопку «OK».

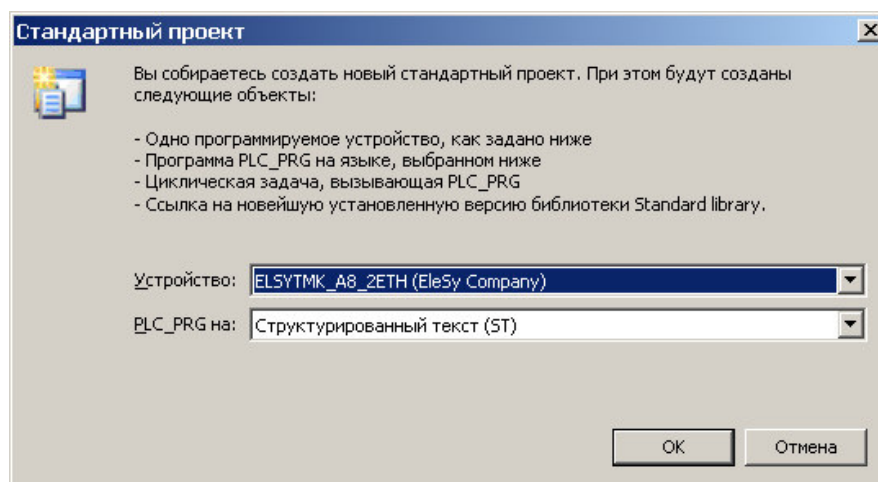


Рисунок 4 – Диалоговое окно «Стандартный проект»

9. В дереве устройств установить курсор мыши на **Device**, щелчком правой кнопки мыши вызвать контекстное меню и выбрать команду «*Добавить устройство...*». Далее выбрать крейт «TK711_10» либо «TK711_6», в зависимости от исполнения коммутационной панели (в приведенном примере – это «TK711_6», Рисунок 5), и нажать кнопку «Добавить устройство».

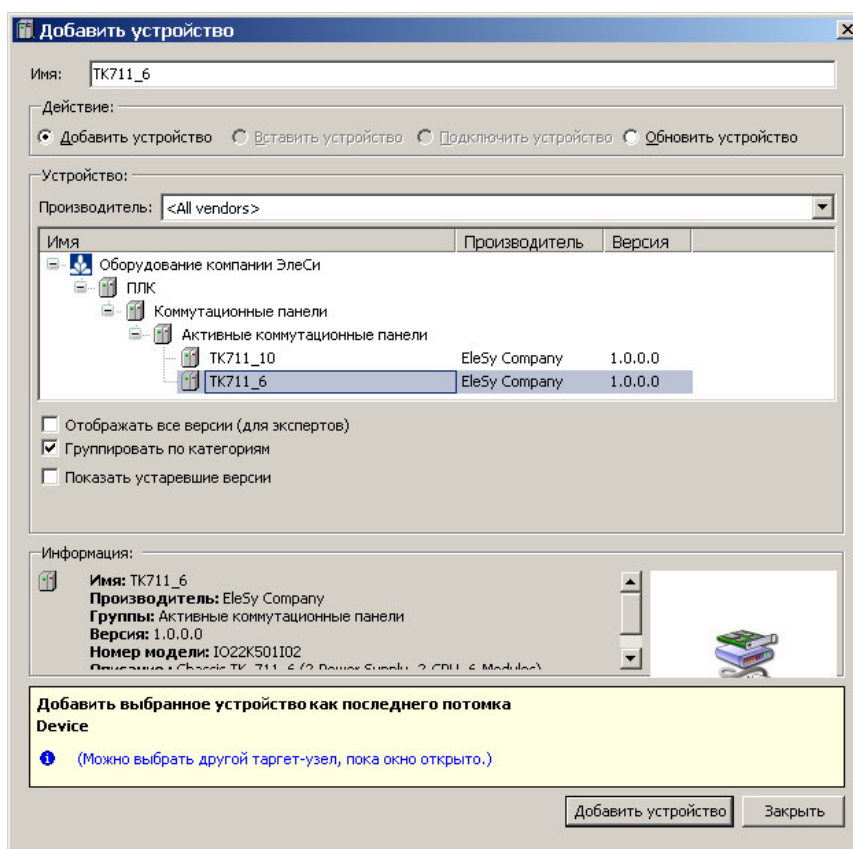


Рисунок 5 - Окно «Добавить устройство»

Примечание – При необходимости, можно, не закрывая диалоговое окно, последовательно добавить еще несколько устройств.

10. В дереве устройств установить курсор мыши на элемент **Power Supplies**, щелчком правой кнопки мыши вызвать контекстное меню и выбрать команду «Добавить устройство ...». Далее выбрать модуль источника питания «TP711» либо «TP712», в зависимости от исполнения (в приведенном примере – это «TP711») и нажать кнопку «Добавить устройство».
11. В дереве устройств установить курсор мыши на элемент «ModuleCP», щелчком правой кнопки мыши вызвать контекстное меню и выбрать команду «Добавить устройство...». Далее выбрать модуль ЦП «TC712» и нажать кнопку «Добавить устройство».
12. В дереве устройств установить курсор мыши на элемент «TC712», щелчком правой кнопки мыши вызвать контекстное меню и выбрать команду «Добавить устройство...». Далее выбрать устройство «Redundancy» и нажать кнопку «Добавить устройство». Дерево устройств примет вид в соответствии с Рисунок 6.

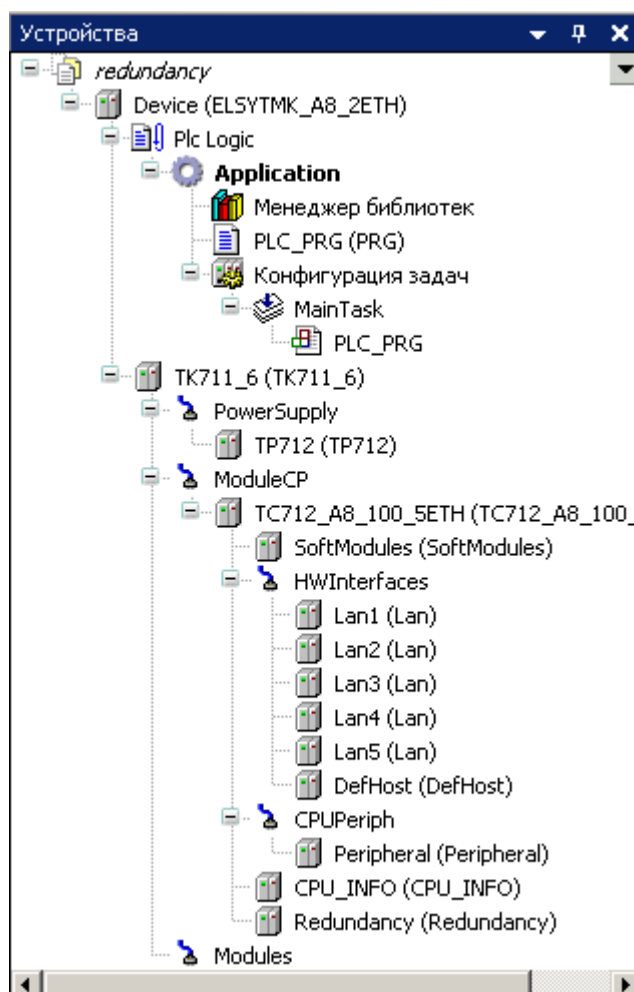


Рисунок 6 – Дерево устройств

Выполнить настройку конфигурации задачи («**Main Task**»), для этого:

- двойным щелчком по элементу «**Main Task**» перейти в режим его просмотра и редактирования;
- в списке возможных значений типа конфигурации выбрать **Внешняя**, при этом, значение в поле «Внешнее событие» должно смениться на **RedundancyCyclicInterrupt** (если нет, необходимо сделать это вручную!).

Примечание – В случае задания неверного параметра конфигурации задачи пользователем, при компиляции система выдаст соответствующее сообщение об ошибке.

13. В дереве устройств установить курсор мыши на элемент **PLC_PRG** и нажать двойным щелчком левой кнопки мыши. В появившемся окне редактора программ ввести следующий фрагмент программного кода:

Пример использования ФБ Redundancy на языке ST

В верхнем окне редактора:

```
PROGRAM PLC_PRG
VAR
    (* ФБ FBRedundancy*)
    redundancy : ElesyRedundancy.FBREDUNDANCY;
    (* переменная, соотнесенная со входом statePLC*)
    statePLC : EnumStatePLC;
    (* переменная для определения достоверности основного контекста*)
    mvalid : ElesyRedundancy.RDVALID;
    (* переменная для определения достоверности резервного контекста*)
    rvalid : ElesyRedundancy.RDVALID;
    (* Данные основного контекста *)
    MainContext : ARRAY [1..8192] OF INT;
    (* Данные резервного контекста *)
    ReservContext : ARRAY [1..512] OF INT;
    (* Переменная задачи пользователя *)
    mycmd : INT;
    (* Переменная задачи пользователя *)
    remotecmd : INT;
END_VAR
```

В нижнем окне редактора:

```
(* Для работы с функциональным блоком резервирования *)
redundancy();

(* Регистрация контекстов *)
IF redundancy.isnotaddcontext THEN

    (* Регистрация основного контекста *)
    redundancy.addmaincontext (ADR(MainContext), SIZEOF(MainContext));

    (*Регистрация основного контекста ОБЯЗАТЕЛЬНО должна заканчиваться вызовом
метода с 0-ми значениями *)
    redundancy.addmaincontext (0, 0);

    (* Регистрация резервного контекста *)
    redundancy.addreservcontext (ADR(ReservContext), SIZEOF(ReservContext));

    (*Регистрация резервного ОБЯЗАТЕЛЬНО должна заканчиваться вызовом метода с 0-ми
значениями *)
    redundancy.addreservcontext (0, 0);
END_IF

(* Проверка наличия ошибок после исполнения методов ФБ *)
IF redundancy.error <> 0 THEN //
(*ранее IF redundancy.error <> ElesyRedundancy.RDERROR.NO_ERROR*)
    (* Выходим из задачи *)
    RETURN;
END_IF

(* Если задача выполняется в состоянии Active или Standalone *)
IF statePLC = EnumStatePLC.ID_Active OR statePLC = EnumStatePLC.ID_StandAlone THEN
    (* Сохраняем данные в основном контексте для передачи резервному ПЛК *)
    MainContext[1] := mycmd;
    (* Получим признак достоверности резервного контекста *)
    rvalid := redundancy.getvalidreservcontext();
    (* Извлекаем данные из резервного контекста если контекст достоверен*)
```

```

        IF rvalid = ElesyRedundancy.RDVALID.OK_VALID THEN
            remotecmd := ReservContext[1];
            (* Передаем команду по окончании задачи о смене роли ПР *)
            IF remotecmd = 5 THEN
                redundancy.switchstate();
                (* Выходим из задачи *)
                RETURN;
            END_IF
        END_IF
    END_IF
END_IF

(* Если задача выполняется в состоянии Standby или StandbyBlock *)
IF statePLC = EnumStatePLC.ID_Standby OR statePLC = EnumStatePLC.ID_StandbyBlock
THEN
    (* Сохраняем данные в резервном контексте для передачи основному ПЛК *)
    ReservContext[1] := mycmd;
    (* Получим признак достоверности основного контекста *)
    mvalid := redundancy.getvalidmaincontext();
    (* Извлекаем данные из основного контекста если контекст достоверен*)
    IF mvalid = ElesyRedundancy.RDVALID.OK_VALID THEN
        remotecmd := MainContext[1];
        (* Передаем команду по окончании задачи о смене роли ПР *)
        IF remotecmd = 10 THEN
            redundancy.switchstate();
            (* Выходим из задачи *)
            RETURN;
        END_IF
    END_IF
END_IF
END_IF

```

ВНИМАНИЕ! Выход ФБ «error» может установиться только после вызова метода ФБ!

ВНИМАНИЕ! Ошибки *MAIN_SIGNATURE_ERROR* и *RESERV_SIGNATURE_ERROR* могут быть обнаружены только после получения контекста пользовательских данных от удаленного ПЛК!

ВНИМАНИЕ! Если ПР находится в «Ожидающей роли» («Standby», «StandbyBlock»), то в зарегистрированный основной контекст будут приходить данные от «Активного» ПЛК («Active»), а из зарегистрированного резервного контекста данные будут передаваться «Активному» ПЛК!

ВНИМАНИЕ! Если ПР находится в «Активной роли» («Active»), то в зарегистрированный резервный контекст будут приходить данные от «Ожидающего» ПЛК («Standby», «StandbyBlock»), а из зарегистрированного основного контекста данные будут передаваться «Ожидающему» ПЛК!

ВНИМАНИЕ! Данные, необходимые для резервирования, прописываются в секции *Redundancy*. Данные, находящиеся вне секции, не гарантированы!

Отладка данного проекта представлена в разделе «2.10 Отладка проекта с поддержкой резервирования в контроллере ЭЛСИ-ТМК».

Регистрация контекстов включает:

- добавление ФБ «FBRedundancy» в задачу пользователя «MainTask»;
- объявление переменных для резервирования в основном контексте и переменных для резервирования в резервном контексте;
- вызов методов регистрации резервируемых данных в основном и резервном контексте – *addmaincontext* и *addreservcontext*;
- построение логики программы;

- соотнесение переменных в задаче с входами устройства «Redundancy»;
- построение первого блока логики работы задачи в роли «*Active*» и «*StandAlone*», с использованием соотнесенной с входом «*StatePLC*» переменной;
- построение второго блока логики работы задачи в роли «*Standby*» и «*StandbyBlock*», с использованием соотнесенной с входом «*StatePLC*» переменной;
- построение третьего блока логики для обнаружения неисправностей в первом блоке логики при значении «*Active*» входа «*StatePLC*».

Примечание – При обнаружении неисправностей в третьем блоке логики следует вызвать метод *switchstate* и переключить логику работы текущего ПЛК в режим работы «*StandBy*», а удаленного ПЛК – в режим работы «*Active*».

2.6 Информационные параметры устройства «Redundancy»

Для чтения информационных параметров подсистемы резервирования необходимо в дереве устройств нажать двойным щелчком левой кнопки мыши на устройство «Redundancy» и затем перейти на вкладку «Редактор параметров». Перечень информационных параметров модуля приведен в Таблица 3.

Таблица 3 – Устройство Redundancy. Информация модуля

Имя	Значение по умолчанию	Описание
<i>CfgVersion</i>	<i>no data</i>	Конфигурационная версия ПР Redundancy
<i>RealVersion</i>	<i>no data</i>	Реальная версия ПР Redundancy
<i>SWState</i>	<i>no data</i>	Номер текущего ПЛК
<i>IpFirstLinePLC1</i>	<i>no data</i>	IP первой линии резервирования текущего ПЛК
<i>IpSecondLinePLC1</i>	<i>no data</i>	IP второй линии резервирования текущего ПЛК
<i>IpFirstLinePLC2</i>	<i>no data</i>	IP первой линии резервирования удаленного ПЛК
<i>IpSecondLinePLC2</i>	<i>no data</i>	IP второй линии резервирования удаленного ПЛК
<i>SilenceTime</i>	<i>no data</i>	Время обнаружения активности удаленного ПЛК
<i>StatePLCCur</i>	<i>None</i>	Состояние (роль) ПР текущего ПЛК
<i>ChStateReason</i>	<i>NotChangeState</i>	Причина последнего изменения состояния ПР
<i>ErrorPLCCur</i>		Код ошибки службы резервирования
<i>ErrorPLCPeer</i>		Код ошибки службы резервирования второго ПЛК
<i>Line1State</i>	<i>no data</i>	Состояние первой линии резервирования
<i>Line2State</i>	<i>no data</i>	Состояние второй линии резервирования
<i>StatePLCPeer</i>	<i>no data</i>	Состояние (роль) ПР удаленного ПЛК
<i>Jitter</i>	<i>no data</i>	Текущая задержка запуска цикла задачи, мкс
<i>MinJitter</i>	<i>no data</i>	Минимальная задержка запуска цикла задачи, мкс
<i>MaxJitter</i>	<i>no data</i>	Максимальная задержка запуска цикла задачи, мкс
<i>CntErr_StartTask</i>	<i>no data</i>	Счётчик ошибок запуска задачи при запущенной задаче
<i>CntErr_OverRuntimeTask</i>	<i>no data</i>	Счётчик ошибок превышения максимального времени исполнения задачи
<i>CntErr_RecvNotValidCadr</i>	<i>no data</i>	Счётчик ошибок полученных недостоверных кадров
<i>CntErr_OvertimeDisconnect</i>	<i>no data</i>	Счётчик ошибок превышения максимального времени разрыва связи с удаленным ПЛК

Информационные параметры *CfgVersion* и *RealVersion* необходимы для визуальной проверки соответствия версии устройства «Redundancy» в проекте пользователя и версии ПР «Redundancy» ПО модуля центрального процессора. При совпадении версий, устройство «Redundancy» проекта совместимо с ПО ПР модуля ЦП.

Информационный параметр *SWState* необходим для отображения текущего положения переключателя «Switch» (переключатель в положении «1» – ПЛК принимает роль *Активного (Main)*, переключатель в положении «2» – ПЛК принимает роль *Ожидającego (Reserv)*). При

неправильной установке переключателей значения параметров на обоих ПЛК будут содержать одинаковые значения.

Информационные параметры *IpFirstLinePLC1*, *IpSecondLinePLC1*, *IpFirstLinePLC2* и *IpSecondLinePLC2* служат для визуальной проверки адресов линий резервирования.

Информационный параметр *SilenceTime* и *AddSilenceTime* (см. Настройка системных параметров устройства «Redundancy») необходимы для обеспечения запуска ПР в обоих ПЛК в predetermined ролях – Активной (Active) или Ожидающей (Standby).

Информационный параметр *StatePLCCur* необходим для визуальной проверки состояния ПР текущего ПЛК.

Возможные значения состояния (роли) ПР текущего ПЛК (параметр *StatePLCCur*):

- **None** – ПР не запущена;
- **Active** – Активная роль. Связь с удаленным ПЛК есть;
- **Standby** – Ожидающая роль. Связь с удаленным ПЛК есть;
- **StandAlone** – Активная роль. Связи с удаленным ПЛК нет;
- **StandbyBlock** – Ожидающая роль. Связь с удаленным ПЛК есть. Переход в Активную роль заблокирован;
- **StandbyBlockNoActive** – Ожидающая роль. Связи с удаленным ПЛК нет. Переход в Активную роль заблокирован.

Информационный параметр *ChStateReason* необходим для визуальной проверки причины последнего изменения состояния ПР. Возможные значения параметра *ChStateReason*:

- **NotChangeState** – состояние ни разу не менялось;
- **FirstStart** – запуск ПР;
- **MainTask** – вызов метода *switchstate* функционального блока «FBRedundancy» из задачи пользователя (см. «Параметры установки устройства «Redundancy»);
- **Button** – нажатие кнопки в устройстве «Redundancy» (режим конфигурации проекта, вкладка «Редактор параметров» - «Параметры установки»);
- **Disconnect** – потеря связи с удаленным ПЛК;
- **Connect** – восстановление связи с удаленным ПЛК.

Информационные параметры *ErrorPLCCur* и *ErrorPLCPeer* необходимы для визуальной идентификации кода ошибки ПР. Ниже приведены возможные значения параметров *ErrorPLCCur* и *ErrorPLCPeer*:

- **START_EXEVENT_ERROR** - ошибка привязки внешнего события к задаче «MainTask»;
- **START_PARAMS_ERROR** - ошибка параметров в конфигурации «Redundancy»;
- **START_SYSTEM_ERROR** - системная ошибка запуска подсистемы резервирования;
- **RUN_MAINCTX_OVERCNT_ERROR** - ошибка превышения максимального количества регистрируемых областей основного контекста;
- **RUN_MAINCTX_PMEM_ERROR** - ошибка входного адреса регистрируемой области основного контекста;

- **RUN_MAINCTX_SIZE_ERROR** - ошибка входного размера регистрируемой области основного контекста;
- **RUN_MAINCTX_OVERSIZE_ERROR** - ошибка превышения максимального размера основного контекста;
- **RUN_MAINCTX_SIGNATURE_ERROR** - ошибка сигнатур, зарегистрированных областей основного контекста на обоих ПЛК;
- **RUN_RESCTX_OVERCNT_ERROR** - ошибка превышения максимального количества регистрируемых областей резервного контекста;
- **RUN_RESCTX_PMEM_ERROR** - ошибка входного адреса регистрируемой области резервного контекста;
- **RUN_RESCTX_SIZE_ERROR** - ошибка входного размера регистрируемой области резервного контекста;
- **RUN_RESCTX_OVERSIZE_ERROR** - ошибка превышения максимального размера резервного контекста;
- **RUN_RESCTX_SIGNATURE_ERROR** - ошибка сигнатур, зарегистрированных областей резервного контекста на обоих ПЛК;
- **RUN_SWSTATE_ERROR** - ошибка одинакового значения *SWState* в конфигурации;
- **RUN_HWSWITCH_ERROR** - ошибка одинакового положения аппаратного переключателя «Switch» в режиме «Production»;
- **RUN_SWITCHCONTROL_ERROR** - ошибка передачи управления удаленному ПЛК в состоянии «Standbyblock»;
- **STOP_SYSTEM_ERROR** - системная ошибка остановки подсистемы резервирования.

Информационные параметры *Line1State* и *Line2State* необходимы для визуальной проверки состояний линий резервирования. При наличии активности параметр принимает значение «1», в отсутствии – «0»;

Информационный параметр *StatePLCPeer* служит для визуальной проверки состояния (роли) ПР удаленного ПЛК. Возможные значения состояния соответствуют значениям параметра *StatePLCCur* (см. выше).

Информационные параметры *Jitter*, *MinJitter*, *MaxJitter* необходимы для визуальной проверки задержки запуска цикла задачи, правильной настройки системного параметра *IntervalTask* и задачи пользователя (см. «Настройка системных параметров устройства «Redundancy»). Данные параметры актуальны только для ПЛК в «Активной» роли.

Информационный параметр *CntErr_StartTask* служит для визуальной проверки ошибок запуска задачи при запущенной задаче. Ненулевое значение свидетельствует о неправильной настройке системного параметра *IntervalTask* или задачи пользователя (см. «Настройка системных параметров устройства «Redundancy»).

Информационный параметр *CntErr_OverRuntimeTask* используется для визуальной проверки ошибок превышения максимального времени исполнения задачи. Ненулевое значение свидетельствует о неправильной настройке системного параметра *IntervalTask* или задачи пользователя (см. «Настройка системных параметров устройства «Redundancy»).

Информационный параметр *CntErr_RecvNotValidCadr* используется для визуальной проверки ошибок полученных недостоверных кадров. Ненулевое значение свидетельствует о наличии промежуточных устройств на выделенной линии резервирования.

Информационный параметр *CntErr_OvertimeDisconnect* служит для визуальной проверки ошибок превышения максимального времени обнаружения разрыва связи с удаленным ПЛК, которое равняется 50 мс для ПЛК в «Ожидающей» роли и 80 мс для ПЛК в «Активной» роли. Ненулевое значение свидетельствует о наличии технических проблем в модуле центрального процессора.

2.7 Настройка системных параметров устройства «Redundancy»

Настройка системных параметров осуществляется до загрузки проекта в ПЛК следующим образом:

1. Двойным щелчком левой кнопки мыши по устройству «Redundancy» в дереве устройств перейти в просмотра.
2. Перейти во вкладку «Редактор параметров», затем щелчком по кнопке  «Системные Параметры Модуля» открыть таблицу системных параметров в соответствии с Рисунок 7. Описание системных параметров приведено в Таблица 4;

Таблица 4 – Устройство «Redundancy». Системные параметры

Имя	Значение по умолчанию	Описание
<i>IntervalTask</i>	<i>100</i>	Интервал запуска резервируемой задачи пользователя, мс.
<i>AddSilenceTime</i>	<i>0</i>	Добавленное время тишины, мс.

Значение параметра *IntervalTask* должно превышать сумму значений параметра *MaxJitter* (см. «Информационные параметры устройства «Redundancy»») и параметра *Max. Cycle Time* (максимальное время исполнения задачи).

- $IntervalTask > MaxJitter + Max. Cycle Time$.

Значение параметра *Max. Cycle Time* должно быть меньше разности значений параметра *IntervalTask* и максимального времени на передачу зарегистрированного контекста ожидающему ПЛК, равного 80 мс.

- $Max. Cycle Time < IntervalTask - 80ms$.

Параметр *AddSilenceTime* необходим при использовании различных проектов в обоих ПЛК. Время, указанное в параметре *AddSilenceTime*, добавляется к значению параметра *SilenceTime* (см. «Информационные параметры устройства «Redundancy»»).

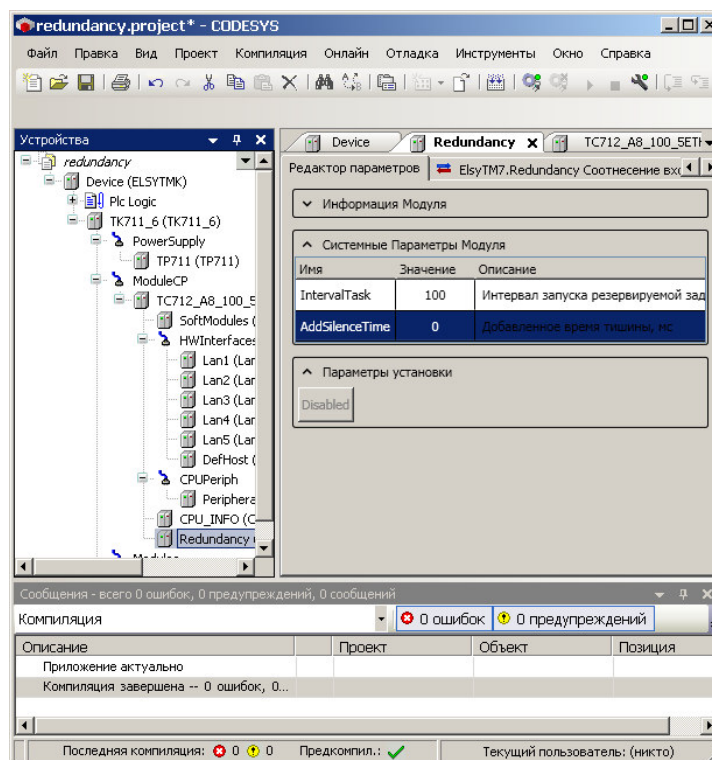


Рисунок 7 – Устройство Redundancy. Вкладка «Редактор параметров». Системные параметры модуля

Изменение значения параметра *IntervalTask* выполняется следующим образом:

- двойным щелчком мыши нажать на поле столбца «Value» и перейти в режим его редактирования (Рисунок 8);
- ввести требуемое значение параметра.

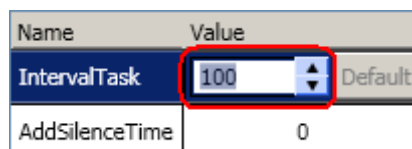


Рисунок 8 – Изменение системного параметра *IntervalTask*

Изменение значения параметра *AddSilenceTime* выполняется следующим образом:

- двойным щелчком мыши нажать на поле столбца «Value» и перейти в режим редактирования (Рисунок 9);
- ввести требуемое значение параметра.

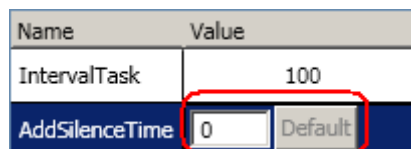


Рисунок 9 – Изменение системного параметра *AddSilenceTime*

2.7.1 Параметры установки устройства «Redundancy»

Состояния (роли) ПЛК в ПР могут меняться не только в соответствии с алгоритмом, заложенным в программном обеспечении модулей, но и в соответствии с командами непосредственно из задачи пользователя в программе *CoDeSys* (метод *switchstate*).

На Рисунок 10 приведена графическая схема возможных состояний ПЛК в ПР.

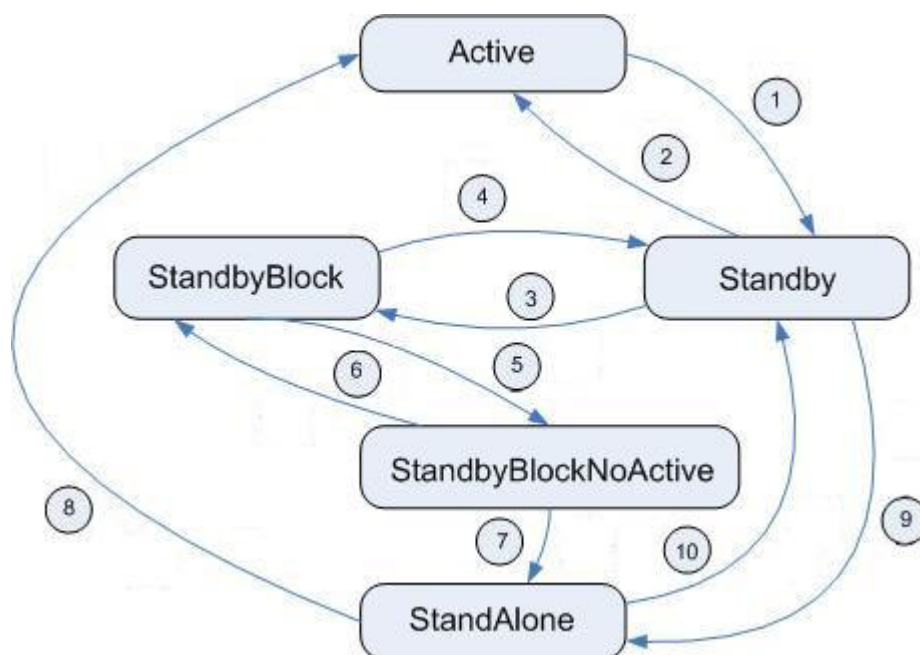


Рисунок 10 - Графическая схема изменения состояний ПЛК в ПР

Описание изменения возможных состояний ПЛК:

- 1) Переход из состояния «*Active*» в состояние «*Standby*» выполняется по команде пользователя (вызов метода *switchstate*);
- 2) Переход из состояния «*Standby*» в состояние «*Active*» выполняется по команде пользователя (вызов метода *switchstate*);
- 3) Переход из состояния «*Standby*» в состояние «*StandbyBlock*» выполняется по команде пользователя (вызов метода *switchstate*);
- 4) Переход из состояния «*StandbyBlock*» в состояние «*Standby*» выполняется по команде пользователя (вызов метода *switchstate*);
- 5) Переход из состояния «*StandbyBlock*» в состояние «*StandbyBlockNoActive*» выполняется согласно алгоритму ПО при отсутствии ПЛК в состоянии «*Active*»;
- 6) Переход из состояния «*StandbyBlockNoActive*» в состояние «*StandbyBlock*» выполняется согласно алгоритму ПО при восстановлении связи с ПЛК в состоянии «*Active*»;
- 7) Переход из состояния «*StandbyBlockNoActive*» в состояние «*StandAlone*» выполняется по команде пользователя;
- 8) Переход из состояния «*StandAlone*» в состояние «*Active*» выполняется согласно алгоритму ПО при восстановлении связи с резервным ПЛК;
- 9) Переход из состояния «*Standby*» в состояние «*StandAlone*» выполняется согласно алгоритму ПО при потере связи с ПЛК в состоянии «*Active*»;
- 10) Переход из состояния «*StandAlone*» в состояние «*Standby*» выполняется согласно алгоритму ПО при восстановлении связи с ПЛК в состоянии «*Active*»;

Вызов метода *switchstate* реализован в программе *CoDeSys* через графический интерфейс. При этом, программа автоматически предлагает допустимый вариант перехода в другое состояние из текущего (Рисунок 11).

«Параметры установки» доступны во время онлайн соединения во вкладке редактора параметров устройства «*Redundancy*». Для выполнения операции следует:

- в дереве устройств двойным щелчком левой кнопки мыши по устройству «Redundancy» перейти в режим просмотра и настройки.
- перейти во вкладку «Редактор параметров», затем щелчком по кнопке  «Параметры установки» получить доступ к кнопке, позволяющей сменить текущее состояние ПЛК (Рисунок 11);

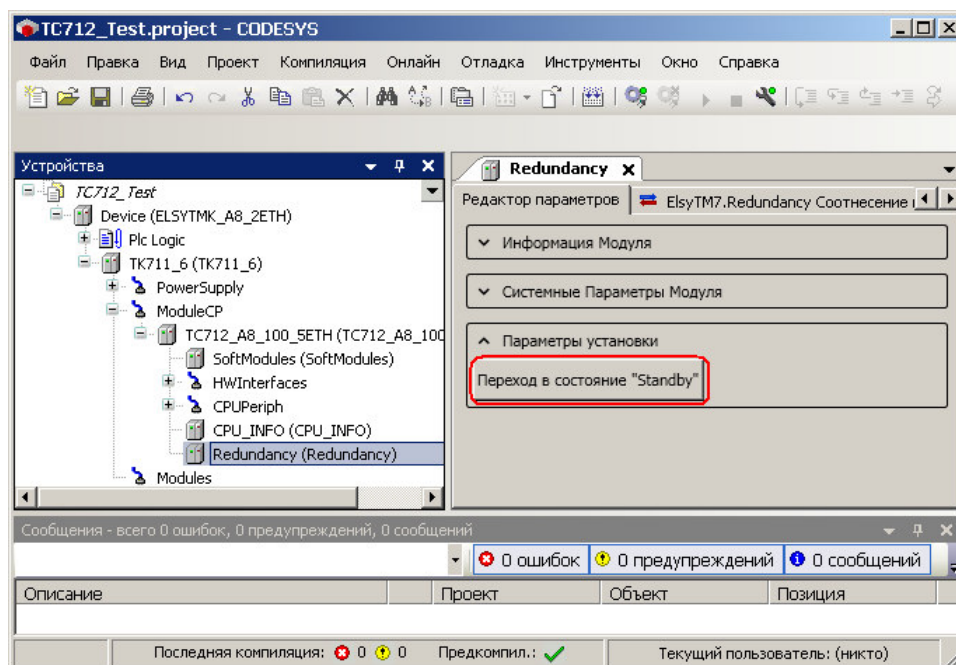


Рисунок 11 –Вкладка «Редактор параметров» - «Параметры установки»

Возможные смены состояния в зависимости от значений информационных параметров *StatePLCCur* и *StatePLCPeer* (см. «Информационные параметры устройства «Redundancy») во время онлайн-соединения приведены в Таблица 5.

Таблица 5 – Устройство Redundancy. Параметры установки

Значение <i>StatePLCCur</i>	Значение <i>StatePLCPeer</i>	Кнопка	Описание действия при нажатии
<i>Active</i>	<i>Standby</i>	<i>Standby</i>	Передача роли «Active» удаленному ПЛК и переход текущего ПЛК в роль «Standby»
<i>Standby</i>	<i>Active</i>	<i>Standbyblock</i>	Переход текущего ПЛК в роль «Standbyblock»
<i>Standbyblock</i>	<i>Active</i>	<i>Standby</i>	Переход текущего ПЛК в роль «Standby»
<i>Standbyblockna</i>	<i>None</i>	<i>Standalone</i>	Переход текущего ПЛК в роль «Standalone»

2.7.2 Соотнесение входов устройства «Redundancy»

Входные диагностические сигналы доступны при переходе во вкладку «ElsyTM7.Redundancy Соотнесение входов/выходов» устройства «Redundancy».

Соотнесения входов выполняется следующим образом:

1. Двойным щелчком левой кнопки мыши по устройству «Redundancy» в дереве устройств перейти в режим просмотра и настройки;
2. Выбрать вкладку **Соотнесение входов/выходов** (Рисунок 12);

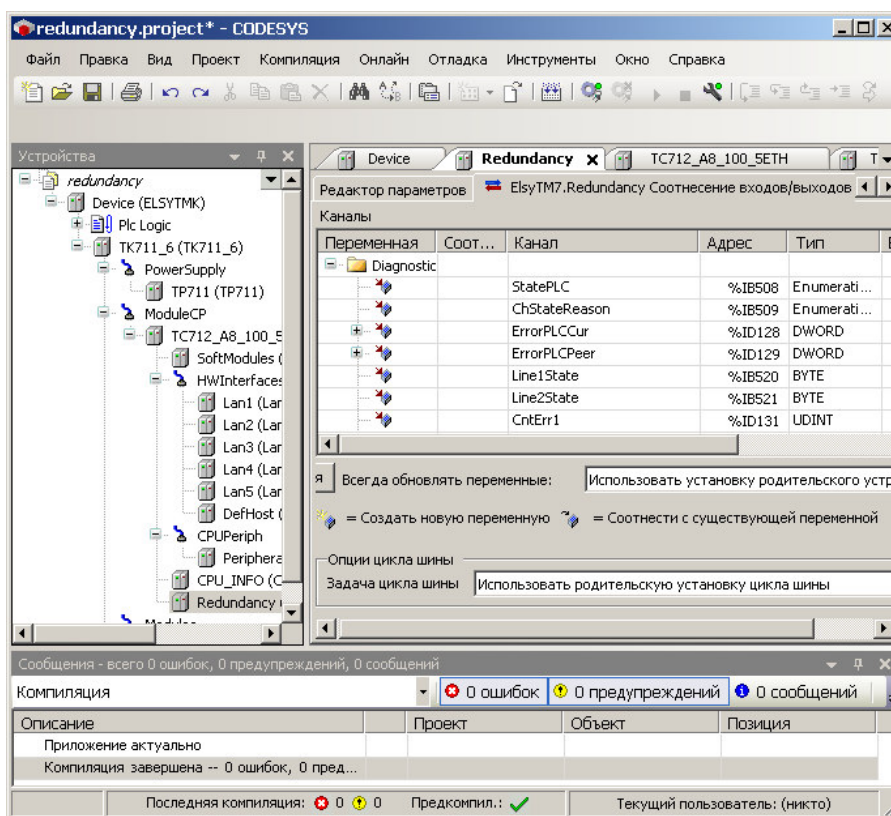


Рисунок 12 - Устройство Redundancy - Вкладка «ElsyTM7.Redundancy Соотнесение входов/выходов»

3. Выполнить соотнесение соответствующих входных сигналов с переменными задачи пользователя (Таблица 6).

Таблица 6 – Устройство Redundancy. Входные сигналы

Имя	Тип данных	Описание
<i>StatePLC*</i>	<i>EnumStatePLC</i>	Состояние (роль) ПР текущего ПЛК
<i>ChStateReason</i>	<i>EnumChStateReason</i>	Причина последнего изменения состояния ПР
<i>ErrorPLCCur</i>	<i>EnumRedundancyERROR</i>	Код ошибки ПР
<i>ErrorPLCPeer</i>	<i>EnumRedundancyERROR</i>	Код ошибки ПР
<i>Line1State</i>	<i>BYTE</i>	Состояние первой линии резервирования
<i>Line2State</i>	<i>BYTE</i>	Состояние второй линии резервирования
<i>CntErr1</i>	<i>UDINT</i>	Счетчик сбоя нумерации входящих кадров по первой линии резервирования текущего ПЛК
<i>CntErr2</i>	<i>UDINT</i>	Счетчик сбоя нумерации входящих кадров по второй линии резервирования текущего ПЛК
<i>CntErr_StartTask</i>	<i>UDINT</i>	Счётчик ошибок запуска задачи при запущенной задаче
<i>CntErr_OverRuntimeTask</i>	<i>UDINT</i>	Счётчик ошибок превышения максимального времени исполнения задачи
<i>CntErr_RecvNotValidCadr</i>	<i>UDINT</i>	Счётчик ошибок полученных недостоверных кадров
<i>CntErr_OvertimeDisconnect</i>	<i>UDINT</i>	Счётчик ошибок превышения максимального времени разрыва связи с удаленным ПЛК

* П р и м е ч а н и е : необходима привязка переменной *StatePLC* к каналу модуля «Redundancy».

Входной сигнал *StatePLC* необходим для проверки состояния (роли) ПР текущего ПЛК. Ниже приведены возможные значения роли:

- **ID_None (0)** – ПР не запущена.
- **ID_Active (1)** – «Активная роль», связь с удаленным ПЛК есть.
- **ID_Standby (2)** – «Ожидающая роль», связь с удаленным ПЛК есть.
- **ID_StandAlone (3)** – «Активная роль», связи с удаленным ПЛК нет.

- **ID_StandbyBlock (4)** – «Ожидающая роль», связь с удаленным ПЛК есть. Переход в «Активную роль» заблокирован.
- **ID_StandbybBockNoActive (5)** – «Ожидающая роль», связи с удаленным ПЛК нет. Переход в «Активную роль» заблокирован.

Входной сигнал **ChStateReason** служит для проверки причины последнего изменения состояния ПР. Ниже приведен список возможных значений входа **ChStateReason**:

- **NotChangeState (0)** – состояние ещё ни разу не менялось.
- **FirstStart (1)** – запуск ПР.
- **MainTask (2)** – вызов метода *switchstate* ФБ «FBREDUNDANCY» из задачи пользователя (см. ниже «2.8 Работа с ФБ «FBREDUNDANCY»).
- **Button (3)** – нажатие кнопки в устройстве «Redundancy» в режиме конфигурации проекта.
- **Disconnect (4)** – потеря связи с удаленным ПЛК.
- **Connect (5)** – восстановление связи с удаленным ПЛК.

Входные сигналы **ErrorPLCCur** и **ErrorPLCPeer** необходимы для идентификации кода ошибки ПР. Список возможных значений **ErrorPLCCur** и **ErrorPLCPeer** представлен в п. 2.6.

Входные сигналы **Line1State** и **Line2State** необходимы для проверки состояний линий резервирования. При наличии активности значение параметра будет установлено в **1**, при отсутствии – в **0**.

Входной сигнал **CntErr1** и **CntErr2** необходимы для проверки сбоя нумерации входящих кадров по линиям резервирования текущего ПЛК. Ненулевое значение свидетельствует либо о наличии источника помех на линии либо о разрыве и восстановлении линии связи, либо о перегрузке модуля ЦП.

Входной сигнал **CntErr_StartTask** служит для проверки ошибок запуска задачи при запущенной задаче. Ненулевое значение свидетельствует о неправильной настройке системного параметра **IntervalTask** или задачи пользователя (см. «2.7 Настройка системных параметров устройства «Redundancy»).

Входной сигнал **CntErr_OverRuntimeTask** служит для проверки ошибок превышения максимального времени исполнения задачи. Ненулевое значение свидетельствует о неправильной настройке системного параметра **IntervalTask** или задачи пользователя (см. Настройка системных параметров устройства «Redundancy»).

Входной сигнал **CntErr_RecvNotValidCadr** используется для проверки ошибок полученных недостоверных кадров. Ненулевое значение свидетельствует о наличии промежуточных устройств на выделенной линии резервирования.

Входной сигнал **CntErr_OvertimeDisconnect** необходим для проверки ошибок превышения максимального времени обнаружения разрыва связи с удаленным ПЛК, которое равняется 50 мс для ПЛК в «Ожидающей роли» и 80 мс для ПЛК в «Активной роли». Ненулевое значение свидетельствует о наличии технических проблем в модуле центрального процессора.

2.8 Работа с ФБ «FBREDUNDANCY»

Для ПР необходимо зарегистрировать контексты пользовательских данных, которыми будут обмениваться два ПЛК. Также имеется возможность смены ролей ПР из задачи

пользователя.

Для реализации этого функционала используется ФБ «FBREDUNDANCY».

ФБ «FBREDUNDANCY» находится в библиотеке **ElesyRedundancy**, которая автоматически создается в «Менеджере библиотек» («Library Manager») при добавлении устройства «Redundancy» в дерево устройств «Devices».

Доступ к списку библиотек осуществляется двойным щелчком левой кнопки по элементу «Менеджер библиотек». Пример списка библиотек приведен на Рисунок 13.

Для получения доступа к описанию любого из компонентов выбранной библиотеки следует перейти во вкладку «Документация».

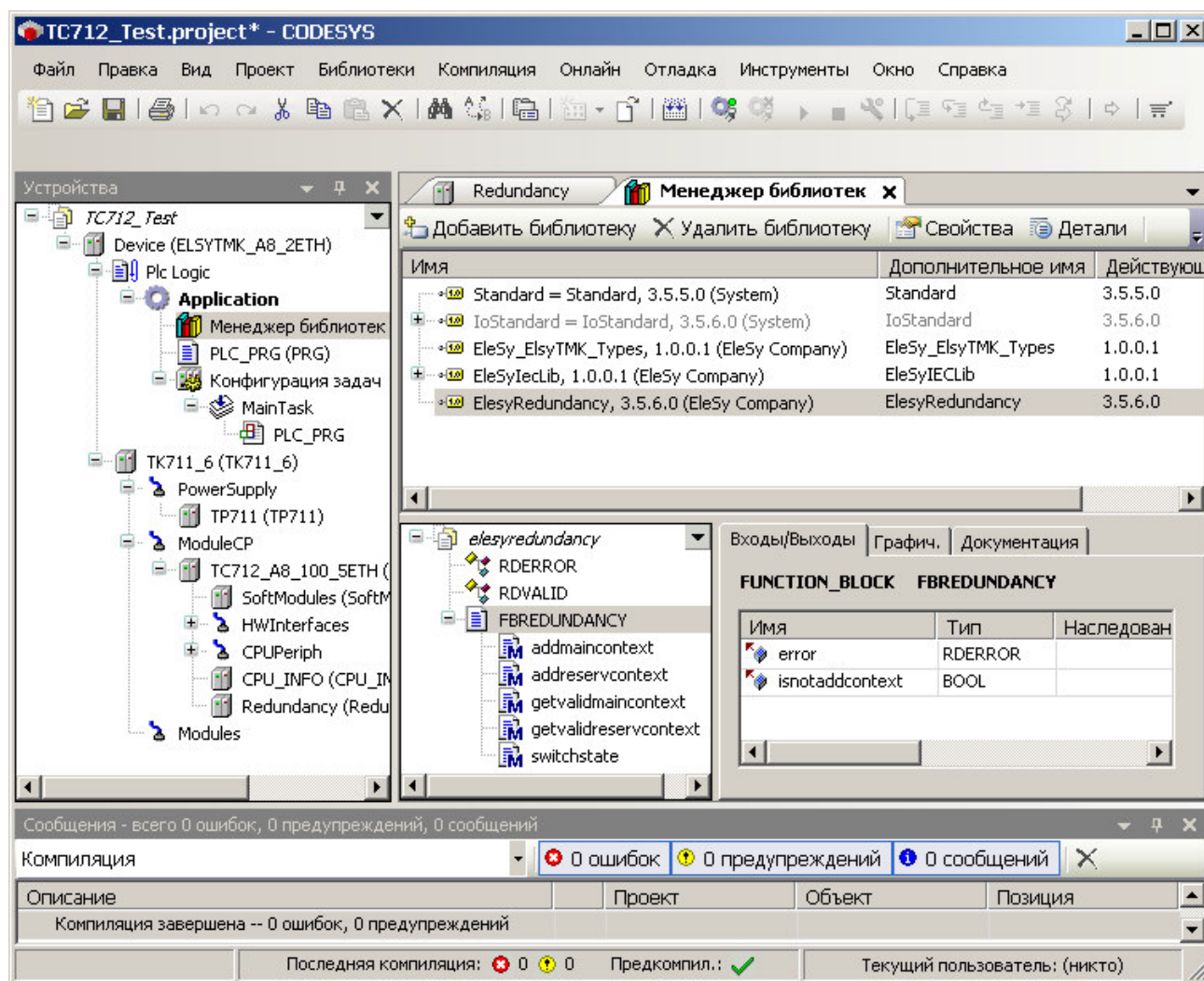


Рисунок 13 – Менеджер библиотек. Библиотека **ElesyRedundancy**

В состав библиотеки входят следующие компоненты:

- RDERROR (ENUM) – диагностика ФБ, см. Таблица 7;
- RDVALID (ENUM) – достоверность контекста, см. Таблица 8 ;
- FBREDUNDACY (FB) – ФБ FBREDUNDANCY, см. Таблица 9 :
 - **addmaincontext (МЕТН)** – регистрация элемента данных в основном контексте;
 - **addreservcontext (МЕТН)** – регистрация элемента данных в резервном контексте;

- **getvalidmaincontext (МЕТН)** – получить достоверность основного контекста;
- **getvalidreservcontext (МЕТН)** – получить достоверность резервного контекста;
- **switchstate (МЕТН)** – смена роли ПР.

Таблица 7 – ФБ FBREDUNDANCY. Диагностика ФБ FBREDUNDANCY (RDERROR)

Имя	Тип	Значение	Описание
START_EXEVENT_ERROR	INT	1	Ошибка привязки внешнего события к задаче резервирования
START_PARAMS_ERROR		2	Ошибка параметров в конфигурации Redundancy
START_SYSTEM_ERROR		4	Системная ошибка запуска подсистемы резервирования
RUN_MAINCTX_OVERCNT_ERROR		8	Ошибка превышения максимального количества регистрируемых областей основного контекста
RUN_MAINCTX_PMEM_ERROR		16	Ошибка входного адреса регистрируемой области основного контекста
RUN_MAINCTX_SIZE_ERROR		32	Ошибка входного размера регистрируемой области основного контекста
RUN_MAINCTX_OVERSIZE_ERROR		64	Ошибка превышения максимального размера основного контекста
RUN_MAINCTX_SIGNATURE_ERROR		128	Ошибка сигнатур, зарегистрированных областей основного контекста на обоих ПЛК
RUN_RESCTX_OVERCNT_ERROR		256	Ошибка превышения максимального количества регистрируемых областей резервного контекста
RUN_RESCTX_PMEM_ERROR		512	Ошибка входного адреса регистрируемой области резервного контекста
RUN_RESCTX_SIZE_ERROR		1024	Ошибка входного размера регистрируемой области резервного контекста
RUN_RESCTX_OVERSIZE_ERROR		2048	Ошибка превышения максимального размера резервного контекста
RUN_RESCTX_SIGNATURE_ERROR		4096	Ошибка сигнатур зарезервированных областей резервного контекста на обоих ПЛК
RUN_SWSTATE_ERROR		8192	Ошибка одинакового значения SWState в конфигурации
RUN_HWSWITCH_ERROR		16384	Ошибка одинакового положения аппаратного переключателя SWITCH в режиме PRODUCTION
RUN_SWITCHCONTROL_ERROR		32768	Ошибка передачи управления удаленному ПЛК в состоянии Standbyblock
STOP_SYSTEM_ERROR		65536	Системная ошибка остановки подсистемы резервирования

Таблица 8 – ФБ FBREDUNDANCY. Признак достоверности контекста (RDVALID)

Имя	Тип	Значение	Описание
OK_VALID	INT	0	Контекст достоверен
ERROR_SIGNATURE_VALID		1	Недостоверна сигнатура зарегистрированных элементов в контексте
ERROR_RECVCONTEXT_VALID		2	Еще не получал или не регистрировал контекст

Таблица 9 – ФБ FBREDUNDANCY. Состав ФБ

Область	Имя	Тип	Значение	Описание
Выхода	error	RDERROR	NO_ERROR	Диагностика ФБ
	isnotaddcontext	BOOL	TRUE	Признак регистрации контекстов
Методы	addmaincontext	-	-	Регистрация данных в основном контексте
	addreservcontext			Регистрация данных в резервном контексте

Область	Имя	Тип	Значение	Описание
	<i>getvalidmaincontext</i>			Получение признака достоверности основного контекста
	<i>getvalidreservcontext</i>			Получение признака достоверности резервного контекста
	<i>switchstate</i>			Смена состояния ПЛК

Выход ФБ *isnotaddcontext* необходим для однократного вызова методов регистрации контекстов пользовательских данных. Если выход установлен в **TRUE**, можно вызывать методы регистрации контекстов *addmaincontext* и *addreservcontext*.

Вызов метода *addmaincontext* регистрирует один элемент данных в основном контексте. На вход метода передается адрес и размер памяти регистрируемого элемента. Адрес элемента вычисляется с помощью оператора **ADR**, а размер с помощью оператора **SIZEOF**.

Вызов метода *addreservcontext* регистрирует один элемент данных в резервном контексте. На вход метода передается адрес и размер памяти регистрируемого элемента. Адрес элемента вычисляется с помощью оператора **ADR**, а размер с помощью оператора **SIZEOF**.

Вызов метода *getvalidmaincontext* возвращает признак достоверности основного контекста (см. описание компонента **RDVALID**, Таблица 8).

Вызов метода *getvalidreservcontext* возвращает признак достоверности резервного контекста (см. описание компонента **RDVALID**, Таблица 8).

Вызов метода *switchstate*, в зависимости от роли ПП текущего и удаленного ПЛК (см. «Информационные параметры устройства «Redundancy»), после завершения задачи выполнит одно из приведенных в Таблица 10 действий.

Таблица 10 – ПП Redundancy. Вызов метода switchstate

Роль ПП текущего ПЛК	Роль ПП удаленного ПЛК	Описание действия при вызове метода <i>switchstate</i>
<i>Active</i>	<i>Standby</i>	Передача «Активной роли» удаленному ПЛК и переход текущего ПЛК в роль «Ожидающего»
<i>Standby</i>	<i>Active</i>	Переход текущего ПЛК в роль «Standbyblock»
<i>Standbyblock</i>	<i>Active</i>	Переход текущего ПЛК в роль «Standby»

В случае возникновения ошибок, выход ФБ «**error**» примет соответствующее значение (см. описание компонента **RDERROR**, Таблица 7).

2.9 Подготовка к работе резервируемых контроллеров

Последовательность действий при подготовке к работе резервируемых контроллеров:

1. Выключить питание на обоих ПЛК.
2. Соединить два ПЛК с помощью выделенных линий резервирования (без участия промежуточных устройств).
3. Установить переключатель «Switch» в положение **1** («Основной» - «Main») на одном ПЛК и установить переключатель «Switch» в положение **2** («Резервный» – «Reserv») на втором ПЛК.
4. Включить питание на обоих ПЛК.
5. Выключить питание после установки лампочки «Р» в зеленый цвет.
6. Установить переключатель в положение **3** (режим **Maintance**) на обоих ПЛК.
7. Включить питание на обоих ПЛК.

8. Создать проект с поддержкой ПР согласно «Создание проекта с поддержкой резервирования в контроллере ЭЛСИ-ТМК».
9. Загрузить проект с поддержкой ПР на оба ПЛК.
10. Выполнить первый и второй этап отладки проекта с поддержкой ПР согласно «2.10 Отладка проекта с поддержкой резервирования в контроллере ЭЛСИ-ТМК».
11. Выключить питание на обоих ПЛК
12. Установить переключатель «Switch» в положение **1** («Основной» - «Main») на одном ПЛК и установить переключатель «Switch» в положение **2** («Резервный» – «Reserv») на втором ПЛК.
13. Одновременно включить питание на обоих ПЛК.
14. Выполнить третий этап отладки проекта с поддержкой ПР согласно «2.10 Отладка проекта с поддержкой резервирования в контроллере ЭЛСИ-ТМК».

Выполнение операций **1-5** обеспечивает связь по линиям резервирования между двумя ПЛК. Данные операции необходимо выполнить как минимум один раз для обеспечения правильной работы ПР в режиме *Maintance*.

Выполнение операций **6-10** необходимо для создания, загрузки и отладки проекта пользователя.

Выполнение операций **11-13** необходимо для запуска обоих ПЛК в режиме «Production». Операция **12** необходима для обеспечения запуска ПЛК в необходимой роли ПР при одновременном включении питания на обоих ПЛК

ВНИМАНИЕ! УСТАНОВКА ПЕРЕКЛЮЧАТЕЛЯ «SWITCH» ДОЛЖНА ВЫПОЛНЯТЬСЯ ТОЛЬКО ПРИ ВЫКЛЮЧЕННОМ ПИТАНИИ ПЛК. УСТАНАВЛИВАТЬ ПЕРЕКЛЮЧАТЕЛИ «SWITCH» НА ОБОИХ ПЛК В ОДИНАКОВЫЕ ПОЛОЖЕНИЯ 1 (MAIN) ИЛИ 2 (RESERV) ЗАПРЕЩАЕТСЯ!

2.10 Отладка проекта с поддержкой резервирования в контроллере ЭЛСИ-ТМК

Целью отладки проекта является:

- проверка ранее выполненной подготовки к работе на наличие ошибок и их устранение;
- проверка правильности выполненной настройки (см. «2.7 Настройка системных параметров устройства «Redundancy»»).

Проверка выполняется в три этапа.

2.10.1 Первый этап отладки

Первый этап отладки осуществляется *после загрузки проекта на оба ПЛК*. На данном этапе необходимо выполнить следующие действия:

1. Проверить на равенство значения информационных параметров *CfgVersion* и *RealVersion* на обоих ПЛК. Неравенство значений свидетельствует о несовместимости ПО системы программирования с ПО модуля ЦП. Для устранения проблемы необходимо установить ПО системы программирования, совместимое с ПО модуля ЦП.

2. Проверить на равенство значение информационного параметра *SWState* на одном ПЛК со значением одноименного параметра на другом ПЛК. Равенство значений свидетельствует о неправильном выполнении действий 1-5 при подготовке к работе резервируемых контроллеров (см. «2.9 Подготовка к работе резервируемых контроллеров»).

Для устранения данной проблемы необходимо заново выполнить действия 1-5 подготовки к работе резервируемых контроллеров.

3. Проверить на равенство значение информационного параметра ***IpFirstLinePLC1*** одного ПЛК со значением информационного параметра ***IpFirstLinePLC2*** второго ПЛК. Неравенство значений свидетельствует о неправильном выполнении действий 1-5 при подготовке к работе резервируемых контроллеров (см. «2.9 Подготовка к работе резервируемых контроллеров»). Для устранения данной проблемы необходимо заново выполнить действия 1-5 подготовки к работе резервируемых контроллеров.

4. Проверить на равенство значение информационного параметра ***IpSecondLinePLC1*** одного ПЛК со значением информационного параметра ***IpSecondLinePLC2*** второго ПЛК. Неравенство значений свидетельствует о неправильном выполнении действий 1-5 при подготовке к работе резервируемых контроллеров (см. «2.9 Подготовка к работе резервируемых контроллеров»). Для устранения данной проблемы необходимо заново выполнить действия 1-5 подготовки к работе резервируемых контроллеров.

5. Проверить на равенство значение информационного параметра ***SilenceTime*** одного ПЛК со значением одноименного параметра второго ПЛК. Равенство значений свидетельствует о неправильном выполнении действий 1-5 при подготовке к работе резервируемых контроллеров (см. «2.9 Подготовка к работе резервируемых контроллеров»). Для устранения данной проблемы необходимо заново выполнить шаги 1-5 подготовки к работе резервируемых контроллеров.

6. Проверить значение информационных параметров ***ErrorPLCCur*** и ***ErrorPLCPeer*** на обоих ПЛК. Ожидаемое значение ***No_Error***.

Значение ***START_EXEVENT_ERROR*** свидетельствует о неправильной настройке задачи «**MainTask**». Для решения этой проблемы необходимо:

- а) тип задачи установить во «**Внешняя**»;
- б) внешнее событие установить в ***RedundancyCyclicInterrupt***.

Примечание – Пользователь может переименовать задачу «MainTask», задав для нее произвольное имя.

Значение ***START_PARAMS_ERROR*** свидетельствует о несовместимости ПО системы программирования с ПО модуля ЦП. Для устранения данной проблемы необходимо установить совместимое ПО системы программирования с ПО модуля ЦП.

Значение ***START_SYSTEM_ERROR*** свидетельствует о системных проблемах ПО модуля ЦП. Для устранения данной проблемы см. «2.10.4 Рекомендации по отладке проектов».

2.10.2 Второй этап отладки

В случае отсутствия проблем на первом этапе, выполняется второй этап отладки. На этом этапе необходимо выполнить следующие действия:

1. Закомментировать вызов метода ***switchstate*** ФБ «FBREDUNDANCY» в задаче пользователя на обоих ПЛК и загрузить проект.

2. Запустить проект на ПЛК2 со значением ***Reserv*** информационного параметра ***SWState***. Затем в течение времени, равного сумме значений информационного параметра ***SilenceTime*** и системного параметра ***AddSilenceTime***, запустить приложение на ПЛК1 со значением ***Main*** информационного параметра ***SWState***.

3. Подождать 10 секунд с момента старта приложения на ПЛК1.

4. На ПЛК1 проверить значение информационного параметра *StatePLCCur*. Ожидаемое значение *Active* (Рисунок 11). Значение *None* свидетельствует о неправильном выполнении действия 2 текущего этапа отладки. Значение *Standby* свидетельствует о неправильном выполнении действия 1 или 2 текущего этапа отладки. Значение *Standalone* свидетельствует о разрыве двух линий резервирования. Значение *Standbyblock* и *StandbyblockNoActive* свидетельствует о неправильном выполнении действия 1 текущего этапа отладки. Для устранения данной проблемы необходимо заново выполнить действия 1-3 текущего этапа отладки.

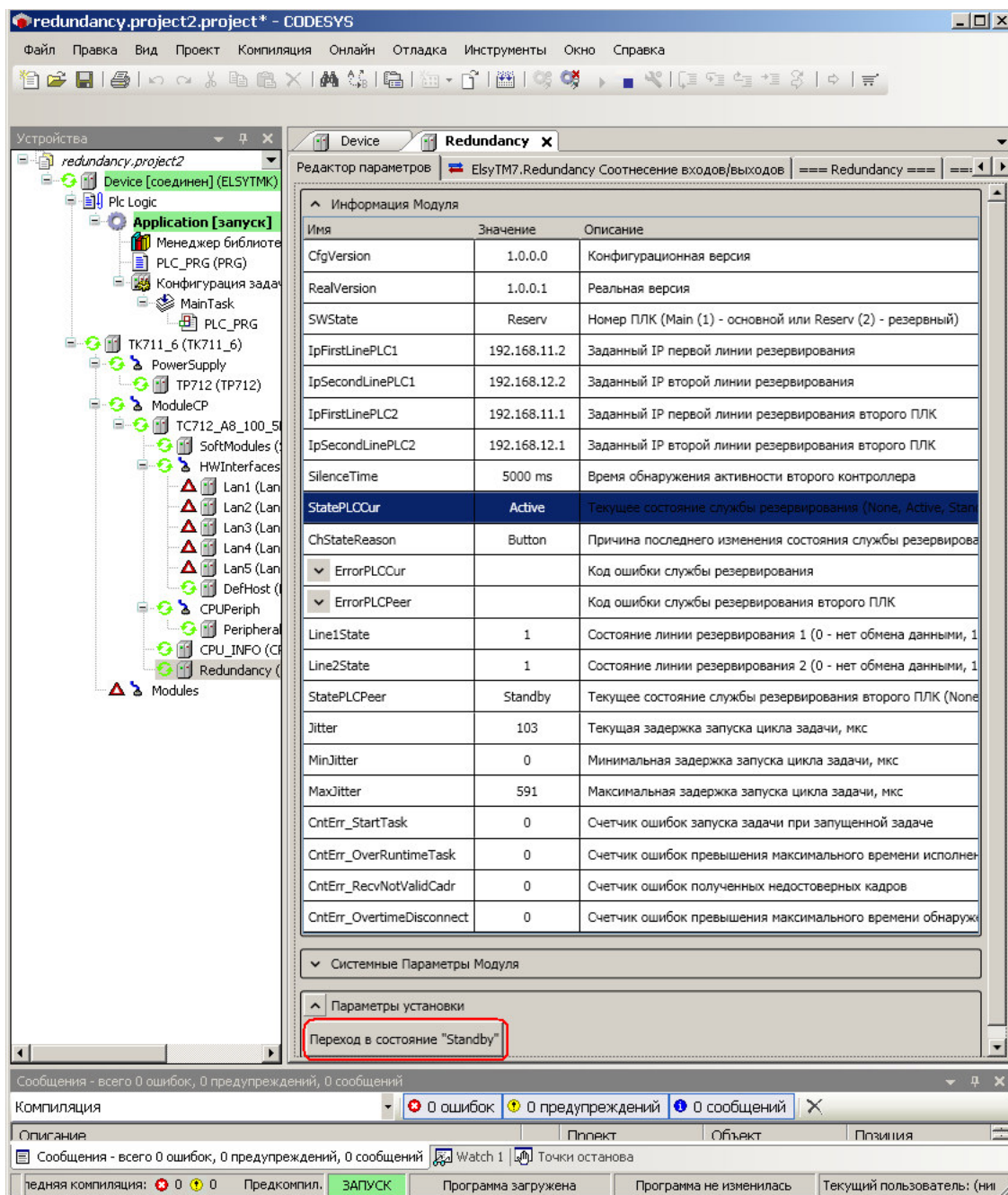


Рисунок 11 – Устройство «Redundancy». Вкладка «Информация Модуля»

ВНИМАНИЕ! После запуска проекта на ПЛК при закрытии вкладки REDUNDANCY (рисунок 11) и повторном ее открытии в среде CoDeSys может возникнуть ошибка, в связи с которой параметры принимают значения, не соответствующие истинным! Для устранения ошибки и проверки необходимо закрыть

вкладку **REDUNDANCY**, перезапустить проект на ПЛК и заново открыть вкладку с параметрами!

5. На ПЛК2 проверить значение информационного параметра **StatePLCCur**. Ожидаемое значение **Standby**. Значение **None** свидетельствует о неправильном выполнении действия 2 текущего этапа отладки. Значение **Active** свидетельствует о неправильном выполнении действия 1 или 2 текущего этапа отладки. Значение **Standalone** свидетельствует о разрыве двух линий резервирования. Значение **Standbyblock** и **StandbyblockNoActive** свидетельствует о неправильном выполнении действия 1 текущего этапа отладки. Для устранения данной проблемы необходимо заново выполнить действия 1-3 текущего этапа отладки.

6. На обоих ПЛК проверить значение информационного параметра **ChStateReason**. Ожидаемое значение **FirstStart**. Значение **NotChangeState** свидетельствует о неправильном выполнении действия 2 текущего этапа отладки. Значение **MainTask** свидетельствует о неправильном выполнении действия 1 текущего этапа отладки. Значение **Button** свидетельствует о том, что изменение роли ПЛК выполнялось по нажатию кнопки в разделе «Параметры установки» устройства «Redundancy». Значение **Disconnect** свидетельствует о разрыве двух линий резервирования. Значение **Connect** свидетельствует о разрыве и последующем восстановлении двух линий резервирования. Для устранения данной проблемы необходимо заново выполнить действия 1-3 текущего этапа отладки, но, если значение установлено в **Disconnect**, то в первую очередь необходимо восстановить линии резервирования.

7. На обоих ПЛК проверить значение информационного параметра **Line1State**. Ожидаемое значение **1**. Значение **0** свидетельствует о разрыве первой линии резервирования. Для устранения данной проблемы необходимо восстановить линию резервирования.

8. На обоих ПЛК проверить значение информационного параметра **Line2State**. Ожидаемое значение **1**. Значение **0** свидетельствует о разрыве второй линии резервирования. Для устранения данной проблемы необходимо восстановить линию резервирования.

9. На ПЛК1 проверить значение информационного параметра **StatePLCPeer**. Ожидаемое значение **Standby**. Значение **None** свидетельствует о неправильном выполнении действия 2 текущего этапа отладки. Значение **Active** свидетельствует о неправильном выполнении действия 1 или 2 текущего этапа отладки. Значение **Standalone** свидетельствует о системной ошибке ПО модуля ЦП. Значение **Standbyblock** и **StandbyblockNoActive** свидетельствует о неправильном выполнении действия 1 текущего этапа. Для устранения данной проблемы необходимо заново выполнить действия 1-3 текущего этапа отладки.

10. На ПЛК2 проверить значение информационного параметра **StatePLCPeer**. Ожидаемое значение **Active**. Значение **None** свидетельствует о неправильном выполнении действия 2 текущего этапа отладки. Значение **Standby** свидетельствует о неправильном выполнении действия 1 или 2 текущего этапа отладки. Значение **Standalone** свидетельствует о системной ошибке ПО модуля ЦП. Значение **Standbyblock** и **StandbyblockNoActive** свидетельствует о неправильном выполнении действия 1 текущего этапа. Для устранения данной проблемы необходимо заново выполнить действия 1-3 текущего этапа отладки.

11. На обоих ПЛК проверить значение информационных параметров **ErrorPLCCur** и **ErrorPLCPeer**. Ожидаемое значение – пустое поле (ранее **No_Error**).

а) значение **RUN_MAINCTX_OVERCNT_ERROR** свидетельствует о превышении максимального количества переменных, добавляемых в основной контекст. Для устранения данной проблемы необходимо устранить лишние вызовы методов **addmaincontext** в задаче пользователя;

- б) значение ***RUN_MAINCTX_PMEM_ERROR*** свидетельствует о неправильной передаче методу ***addmaincontext*** адреса переменной, добавляемой в основной контекст. Для устранения данной проблемы необходимо передать правильный адрес переменной в вызове метода ***addmaincontext*** в задаче пользователя;
- в) значение ***RUN_MAINCTX_SIZE_ERROR*** свидетельствует о неправильной передаче методу ***addmaincontext*** размера переменной, добавляемой в основной контекст. Для устранения данной проблемы необходимо передать правильный размер переменной в вызове метода ***addmaincontext*** в задаче пользователя;
- г) значение ***RUN_MAINCTX_OVERSIZE_ERROR*** свидетельствует о превышении максимального размера основного контекста. Для устранения данной проблемы необходимо ограничить размер переменных, регистрируемых в основном контексте в задаче пользователя;
- д) значение ***RUN_MAINCTX_SIGNATURE_ERROR*** свидетельствует о несовместимости зарегистрированных переменных в основном контексте на ПЛК1 с зарегистрированными переменными в основном контексте на ПЛК2. Для устранения этой проблемы необходимо соблюсти порядок регистрации переменных с одинаковыми типами данных на обоих ПЛК в задаче пользователя;
- е) значение ***RUN_RESERVCTX_OVERCNT_ERROR*** свидетельствует о превышении максимального количества переменных, добавляемых в резервный контекст. Для устранения данной проблемы необходимо устранить лишние вызовы методов ***addreservcontext*** в задаче пользователя;
- ж) значение ***RUN_RESERVCTX_PMEM_ERROR*** свидетельствует о неправильной передаче методу ***addreservcontext*** адреса переменной, добавляемой в резервный контекст. Для устранения данной проблемы необходимо передать правильный адрес переменной в вызове метода ***addreservcontext*** в задаче пользователя;
- з) значение ***RUN_RESERVCTX_SIZE_ERROR*** свидетельствует о неправильной передаче методу ***addreservcontext*** размера переменной, добавляемой в резервный контекст. Для устранения данной проблемы необходимо передать правильный размер переменной в вызове метода ***addreservcontext*** в задаче пользователя;
- и) значение ***RUN_RESERVCTX_OVERSIZE_ERROR*** свидетельствует о превышении максимального размера резервного контекста. Для устранения данной проблемы необходимо ограничить размер переменных регистрируемых в резервном контексте в задаче пользователя;
- к) значение ***RUN_RESERVCTX_SIGNATURE_ERROR*** свидетельствует о несовместимости зарегистрированных переменных в резервном контексте на ПЛК1 с зарегистрированными переменными в резервном контексте на ПЛК2. Для устранения этой проблемы необходимо соблюсти порядок регистрации переменных с одинаковыми типами данных на обоих ПЛК в задаче пользователя. Ошибки, связанные с добавлением переменных в контекст, могут быть установлены только после вызова любого из методов ФБ ***FBREDUNDANCY***;
- л) значение ***RUN_SWSTATE_ERROR***, ***RUN_HWSWITCH_ERROR*** и ***STOP_SYSTEM_ERROR*** свидетельствует о наличии системной ошибки ПО модуля ЦП. Для устранения данной проблемы см. «2.10.4 Рекомендации по отладке проектов»;
- м) значение ***RUN_SWITCHCONTROL_ERROR*** свидетельствует о неправильном выполнении действия 1 или о нажатии кнопки во вкладке «параметры установки» в

устройстве «Redundancy». Для устранения данной проблемы необходимо заново выполнить действия 1-3 текущего этапа отладки.

12. на обоих ПЛК проверить значение информационных параметров **CntErr_StartTask** и **CntErr_OverRuntimeTask**. Ожидаемое значение 0. Ненулевое значение свидетельствует о неправильной настройке системного параметра **IntervalTask** или задачи пользователя. Для устранения данной проблемы см. «2.7 Настройка системных параметров устройства «Redundancy».

13. На обоих ПЛК проверить значение информационного параметра **CntErr_RecvNotValidCadr**. Ожидаемое значение 0. Ненулевое значение свидетельствует о наличии промежуточных устройств на выделенной линии резервирования. Для устранения данной проблемы необходимо устранить промежуточные устройства на линиях резервирования.

14. На обоих ПЛК проверить значение информационного параметра **CntErr_OvertimeDisconnect**. Ожидаемое значение 0. Ненулевое значение свидетельствует о наличии технических проблем в модуле центрального процессора. Для устранения данной проблемы см. «2.10.4 Рекомендации по отладке проектов».

15. Раскомментировать вызов метода **switchstate** ФБ «FBREDUNDANCY» в задаче пользователя на обоих ПЛК и загрузить проект.

16. Отладить пользовательскую логику задачи, а именно: проверить все ветки программы на ожидаемые результаты.

2.10.3 Третий этап отладки

Третий этап отладки выполняется в случае отсутствия проблем на втором этапе, после выполнения действия 13 подготовки к работе резервируемых контроллеров (см. «2.9 Подготовка к работе резервируемых контроллеров»). На данном этапе необходимо выполнить следующие действия:

1. Дождаться мигания лампочки на обоих ПЛК.

2. Подключиться системой программирования к обоим ПЛК.

3. Выполнить действие 11 второго этапа отладки. Если во время выполнения значение информационного параметра **Error** установится в **RUN_HWSWITCH_ERROR**, это будет свидетельствовать об одинаковом положении аппаратного переключателя «Switch» в режиме «PRODUCTION», что может привести к ошибке распределения ролей ПП после перезагрузки ПЛК. Для устранения проблемы необходимо установить переключатели «Switch» в режиме «PRODUCTION» на обоих ПЛК в разные положения (см. «2.4 Назначение трехпозиционного переключателя Switch «123»).

2.10.4 Рекомендации по отладке проектов

При возникновении проблем во время выполнения трёх этапов отладки, описание которых, включая способы их решения, отсутствуют, необходимо обратиться в отдел поддержки, предоставив подробное описание проблемной ситуации, а также проекты, бортовые журналы и информацию о текущем оборудовании обоих ПЛК.

3 Техническое обслуживание

С целью обеспечения постоянной исправности и готовности контроллера к эксплуатации необходимо не реже, чем один раз в год проводить техническое обслуживание.

Порядок технического обслуживания:

1. Отключить питание контроллера, например основного. При этом, резервный контроллер, находящийся в режиме «Standby», переходит в режим «StandAlone».
2. Отстыковать от контроллера все подключенные кабели.
3. Отстыковать модули контроллера от коммутационной панели.
4. Промыть контакты разъемов составных частей контроллера этиловым ректифицированным техническим спиртом по ГОСТ 18300-87. При промывке контакты разъемов должны находиться в вертикальном положении. Норма расхода спирта – 0,05 л на 100 контактов.
5. Просушить на воздухе не менее 30 минут.
6. Установить все модули на исходные места, подключить кабели, затянуть винты крепления модулей и разъемов.
7. Подать питание на контроллер. В результате, подключаемый контроллер должен перейти в режим «Standby», т.е. станет резервным, а контроллер, прежде находившийся в режиме «StandAlone», перейдет в режим «Active», т.е. станет основным.

Примечание - После проведения технического обслуживания в формуляре контроллера в разделе «Учет технического обслуживания» должна быть сделана соответствующая отметка.

4 Решение проблем

В случае возникновения проблем при работе с подсистемой резервирования контроллера ЭЛСИ-ТМК, необходимо обратиться к документации.

Если же проблему самостоятельно устранить не удастся, следует обратиться к [поставщику оборудования](#).

Список литературы

Приложение А (справочное) Структурные схемы построения резервированных систем

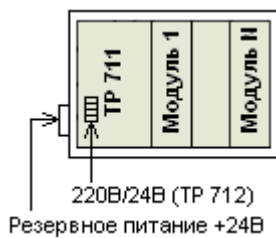
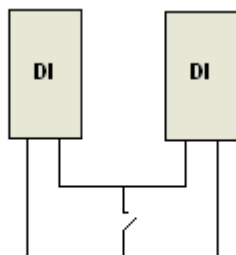


Рисунок А.1 – Структурная схема подключения питания в системе с резервированием

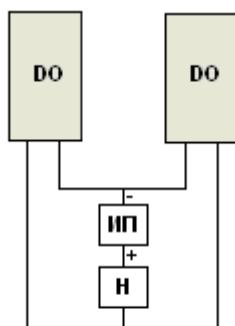


Рисунок А.2 – Структурная схема подключения модулей ЦП в системе с резервированием



DI - TD 711/TD 711m, TD 714
Максимальное сопротивление линий связи - 250Ом

Рисунок А.3 – Структурная схема подключения модулей дискретного ввода в системе с резервированием



DO - TD712, TD 714
ИП - источник питания от 5 до 30В
Н - нагрузка (например, реле)

Рисунок А.4 – Структурная схема подключения модулей дискретного вывода в системе с резервированием

Контактная информация

По всем вопросам, связанным с эксплуатацией контроллера ЭЛСИ-ТМК, обращаться в сервисный центр АО "ЭлеСи":

тел.: +7 (3822) 49-94-94

E-mail: service@elesy.ru

Сервисный центр располагается в г. Томске (часовой пояс +4 МСК).

При обращениях просим сообщать следующие данные:

- полное наименование изделия (указано на изделии или в паспорте);
- проект *CoDeSys*, в котором возникает проблема;
- версия установленного на компьютере пакета *EleSy ELSYTMK TSP (Target Support Package)*;
- подробное описание проблемы (постарайтесь наиболее полно пояснить суть проблемы и обстоятельства или условия, которые привели к ней).

Лист регистрации изменений