

Компьютерные системы **ELOP II**

Первые шаги



HIMA Paul Hildebrandt GmbH + Co KG
Системы промышленной автоматизации

HI 800 001 AVA

Все упомянутые в этом руководстве продукты фирмы HIMA защищены товарными знаками HIMA. Это действует и в отношении других упомянутых в данном руководстве производителей и их продуктов, если не оговорено иначе.

Запрещается копирование и размножение данного документа, использование и передача сведений о его содержании, кроме случаев, подтвержденных особым разрешением. Нарушение этих требований влечет за собой требование возмещения ущерба.

Все технические сведения и указания в настоящем руководстве приведены на основании тщательно выверенных данных, проверенных с соблюдением эффективных мер контроля. Тем не менее, не исключается возможность появления ошибок. В связи с этим, фирма HIMA считает необходимым подчеркнуть, что она не дает гарантию и не несет юридическую или иную ответственность за возможные последствия, возникшие в результате ошибочных сведений. Фирма HIMA с благодарностью примет все сообщения о возможных ошибках.

Фирма HIMA оставляет за собой право на технические изменения.

Более подробная информация содержится в документации на CD-ROM и на нашей интернет-странице www.hima.de.

С вопросами по предоставлению информации следует обращаться по адресу:

HIMA Paul Hildebrandt GmbH + Co KG
Postfach 12 61
68777 Brühl
Deutschland / Германия

Тел.: +49 (6202) 709 0
Факс: +49 (6202) 709 107

Электронная почта: info@hima.com

Содержание

1	КРАТКИЙ ОБЗОР	1
1.1	КОМПЛЕКТ ПОСТАВКИ	1
1.2	ИНФОРМАЦИЯ ПО ДАННОМУ РУКОВОДСТВУ	1
1.3	ТЕХНИЧЕСКАЯ ПОДДЕРЖКА	2
2	ИНСТАЛЛЯЦИЯ.....	3
2.1	ЧТО ТРЕБУЕТСЯ?	4
2.2	КУДА ИНСТАЛЛИРУЕТСЯ ПРОГРАММА?	4
2.3	КАК ЗАПУСТИТЬ ИНСТАЛЛЯЦИЮ?	5
2.4	СЕТЕВАЯ ИНСТАЛЛЯЦИЯ	6
2.5	ДЕИНСТАЛЛЯЦИЯ	6
3	ВВЕДЕНИЕ В ELOP II	7
3.1	ЗАПУСК ELOP II	7
3.2	РАСПРЕДЕЛЕНИЕ ЭКРАНА	8
3.3	СТРОКА НАЗВАНИЯ	9
3.4	СТРОКА МЕНЮ	9
3.5	ПАНЕЛЬ ИНСТРУМЕНТОВ	10
3.6	СТРОКА СОСТОЯНИЯ	10
3.7	ОКНО СТРУКТУРЫ ПРОЕКТА	11
3.7.1	Контекстное меню для объектов	12
3.8	РАБОЧАЯ ОБЛАСТЬ	13
3.8.1	Редактор диаграммы функционального блока (Редактор FBD)	13
3.8.2	Редактор документов	14
3.9	ОКНО СТАТУСА И ОШИБОК	15
3.10	ВСТРОЕННАЯ СПРАВКА	15
4	ОБЪЕКТЫ В ОКНЕ СТРУКТУРЫ КАТАЛОГОВ	17
4.1	ПРОЕКТ	17
4.2	БИБЛИОТЕКА	17
4.2.1	Тип программы	17
4.2.2	Тип функционального блока	18
4.2.3	Функция	18
4.3	КОНФИГУРАЦИЯ	19
4.3.1	Ресурс	19
4.4	ДОКУМЕНТАЦИЯ	20

5	КОНЦЕПЦИЯ БАЗОВОЙ СИСТЕМЫ	21
5.1	МЕНЕДЖЕР ПРОЕКТА.....	21
5.2	БИБЛИОТЕКИ ФУНКЦИОНАЛЬНЫХ БЛОКОВ	22
5.3	РЕДАКТОР ДИАГРАММЫ ФУНКЦИОНАЛЬНОГО БЛОКА.....	23
5.3.1	Развертка рабочей области.....	24
5.3.2	Развертка и восстановление	25
5.4	ФУНКЦИОНАЛЬНЫЕ ДИАГРАММЫ С ЦЕНТРАЛЬНОЙ НАЧАЛЬНОЙ СТРАНИЦЕЙ.....	26
5.4.1	Масштабирование	27
5.5	СОЗДАНИЕ ЛОГИКИ.....	28
5.5.1	Работа с переменными	28
5.5.2	Перенос переменных	30
5.5.3	Перенос функциональных блоков.....	31
5.5.4	Соединение элементов.....	32
5.6	СОЗДАНИЕ ДОКУМЕНТАЦИИ И УПРАВЛЕНИЕ ЕЮ	33
5.6.1	Шаблоны для печати документов	34
5.7	АВТОНОМНОЕ МОДЕЛИРОВАНИЕ ЛОГИКИ	36
6	РАБОТА С ТИПАМИ РЕСУРСОВ	37
6.1	ОПРЕДЕЛЕНИЕ РЕСУРСА.....	38
6.2	ПРИСВОЕНИЕ ТИПА ПРОГРАММЫ РЕСУРСУ	39
6.3	ПРИСВОЕНИЕ В/В (ВХОДА/ВЫХОДА)	40
6.4	ПРИСВОЕНИЕ СИСТЕМНЫХ ПЕРЕМЕННЫХ	42
6.5	ГЕНЕРАТОР КОДА	43
6.5.1	Программа сравнения кода	43
6.6	ПАНЕЛЬ УПРАВЛЕНИЯ.....	44
6.6.1	Загрузка/Перезагрузка	45
6.7	ОНЛАЙНОВЫЙ ТЕСТ.....	47
6.8	ДОКУМЕНТАЦИЯ ПО АППАРАТНОМУ ОБЕСПЕЧЕНИЮ	48
7	УПРАЖНЕНИЯ	49
7.1	СОЗДАНИЕ ПРОЕКТА.....	49
7.2	СОЗДАНИЕ ФУНКЦИОНАЛЬНОГО БЛОКА	54
7.2.1	Описание переменных	54
7.2.2	Схема размещения функционального блока.....	57
7.2.3	Создание логики в область диаграммы.....	61
7.3	СОЗДАНИЕ ПРОГРАММЫ	66
7.3.1	Описание переменных	66
7.3.2	Создание логики в область диаграммы.....	66
7.4	СОЗДАНИЕ КОНФИГУРАЦИИ И РЕСУРСА.....	67
7.5	АВТОНОМНОЕ МОДЕЛИРОВАНИЕ ЛОГИКИ	72

7.6	СХЕМА КОММУТИРУЮЩЕГО ШКАФА PLC И ПРИСВОЕНИЕ ПЕРЕМЕННЫХ.....	80
7.7	ЗАПУСК ПРОГРАММЫ НА PLC	86
7.7.1	Компиляция программы.....	86
7.7.2	Коммуникация между компьютером и контроллером	88
7.7.3	Загрузка и запуск.....	95
7.8	ОНЛАЙНОВЫЙ ТЕСТ	99
7.9	ИНИЦИИРОВАНИЕ ВХОДОВ И ВЫХОДОВ.....	102
7.10	ДОКУМЕНТАЦИЯ.....	108
7.11	РЕЗЕРВИРОВАНИЕ ПРОЕКТА	114
7.11.1	Архивация	114
7.11.2	Восстановление проекта	116
8	ПРИЛОЖЕНИЯ	119
8.1	УКАЗАТЕЛЬ	119
8.2	СПИСОК СОКРАЩЕНИЙ	125

1 Краткий обзор

1.1 Комплект поставки

В комплект поставки ELOP II входят следующие компоненты:

- Настоящее руководство
Руководство "Первые шаги" поможет Вам быстро и просто получить первые сведения по работе с ELOP II. Для этого, помимо общего обзора функций, в нем содержится подробная инструкция по созданию проекта.
- CD-ROM
Наряду с программным обеспечением ELOP II, CD-ROM содержит некоторые вспомогательные программы и полную документацию по актуальным системам фирмы HIMA.
- Аппаратный ключ (донгл)
Аппаратный электронный ключ применяется для контроля за лицензиями (защита от незаконного использования) защищенного программного обеспечения ELOP II.

1.2 Информация по данному руководству

Данное руководство содержит необходимые сведения, позволяющие пользователю освоить важнейшие функции ELOP II в рамках обучающего курса или самостоятельно.

В главе 2 описана инсталляция ELOP II. В главах с 3 по 6 в общих чертах описывается работа с программой ELOP II и ее использование. Не имеющие специальных знаний пользователи должны тщательно проработать эти разделы.

В главе 7 содержатся тренировочные уроки, благодаря которым пользователи, уже имеющие базовые знания по ELOP II, могут научиться создавать проекты или углубить свои познания.

В приложении в главе 8 находятся объяснения терминов и понятий, использующихся в руководстве, а также предметный указатель и список сокращений.

1.3 Техническая поддержка

Вы можете получить ответы на вопросы относительно работы с программой или сообщить об обнаруженных ошибках в программе и предложения по ее улучшению различными способами.

Часто возникающие вопросы	Глава руководства	Вопросы и ответы по основным темам
Новости, FAQ, обновления и документация для загрузки	Наша интернет-страница www.hima.com	Новости, часто возникающие вопросы, функциональные блоки
Вопросы и предложения	По электронной почте: Support@hima.com Телефон: +49-(0)6202-709 259 +49-(0)6202-709 261 Факс: +49-(0)6202-709 199	С 9:00 до 17:00 (среднеевропейское время)

Чтобы мы могли получить всю необходимую для правильного ответа по ELOP II информацию, используйте пожалуйста находящийся на компакт-диске формуляр отзыва (Product Feedback Report – Отзыв потребителя продукта).

Указание: данное руководство является частью рабочей документации для семинаров по программному обеспечению ELOP II, проводящихся фирмой HIMA. В связи с многообразием функций ELOP II здесь описываются только самые важные функции данной программы.

Для углубления знаний рекомендуется участие в одном из семинаров.

2 Инсталляция

В этой главе описаны следующие темы:

- Что требуется?
- Куда инсталлируется программа?
- Как запустить инсталляцию?
- Сетевая инсталляция
- Деинсталляция

ELOP II является программой, защищенной аппаратным электронным ключом. Модуль аппаратного ключа (донгл) необходимо подключить к параллельному порту.



Рис. 1 Аппаратный ключ

Для подключения модуля аппаратного ключа требуется установить на компьютер особый драйвер. Для инсталляции драйвера в Windows NT® пользователь должен иметь права администратора. В случае необходимости следует проконсультироваться с ответственным за работу системы.

2.1 Что требуется?

Помимо персонального компьютера необходим аппаратный ключ и установочный компакт-диск.

Требования к аппаратному обеспечению:

	Минимальные требования	Рекомендуется
Процессор Intel Pentium II®	133 МГц	400 МГц
ОЗУ	64 Мб	128 Мб
Видеокарта	2 Мб XGA (1024x768)	8 Мб XGA (1024x768)
Операционная система	Windows NT® Workstation (рекомендуется актуальный пакет обновления)	

Таблица 1 Требования к аппаратному обеспечению компьютера

Если к аппаратному ключу подключен принтер, он должен быть включен, так как некоторые принтеры в выключенном состоянии оказывают слишком малое замыкающее сопротивление нагрузки.

2.2 Куда устанавливается программа?

Вы можете выполнить инсталляцию на локальный жесткий диск или на сетевой диск (см. главу 2.4).

Если ELOP II, установленный на одном компьютере, будет использоваться несколькими пользователями (с различными именами пользователя), необходимо установить ELOP II для каждого из этих пользователей. Процедура многократной инсталляции описывается во встроенной справке ELOP II или в программе инсталляции.



2.3 Как запустить инсталляцию?

Инсталляция программы ELOP II и всех дополнительных компонентов выполняется из меню инсталляции CD-ROM.

Кроме того, на CD-ROM в виде файлов PDF содержится полная документация по данному программному обеспечению и семейству систем H41q/H51q. Для просмотра документации прилагается программа Adobe Acrobat Reader®.

Инсталляция запускается следующим образом:

1. Вставьте CD-ROM в дисковод для компакт-дисков. Автоматически запускается меню инсталляции.
Если этого не произошло, откройте из Windows Explorer (Проводник Windows) корневой каталог CD-ROM и дважды щелкните кнопкой мыши на файл START.EXE.

Выберите язык инсталляции –  немецкий или  английский.

2. Щелкните на кнопку "ELOP II".
3. Щелкните на кнопку "Hardlock" (Аппаратный ключ).
4. После инсталляции драйвера аппаратного ключа щелкните на кнопку "ELOP II" для установки ELOP II.
5. После успешной инсталляции появляется ELOP II Control Center (Центр управления ELOP II). Находясь в нем Вы можете запускать ELOP II и дополнительные программы. Позже Вы сможете вызвать Control Center (Центр управления) и Менеджер проекта, расположенные в папке ELOP II, из стартового меню Windows.

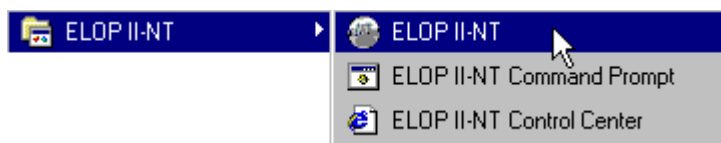


Рис. 2 Стартовое меню ELOP II

2.4 Сетевая инсталляция

1. Инсталляция производится так же, как описано в главе 2.3, но в качестве каталога для инсталляции выбирается один из каталогов на сетевом сервере.
2. Рекомендуется открыть на сетевом сервере доступ к каталогу для инсталляции также и другим пользователям.
3. Выполните инсталляцию на сетевом сервере для каждого рабочего места и каждого пользователя, выполнив для инсталляции программу WS_SETUP из каталога, находящегося на сервере.

2.5 Деинсталляция

Для удаления программы ELOP II запустите Control Center (Центр управления) и выберите пункт "Uninstall" (Удалить). Затем укажите выбранную программу в зависимости от настроек безопасности браузера.

3 Введение в ELOP II

В этой главе описаны следующие темы:

- Запуск программы
- Какие элементы входят в пользовательский интерфейс ELOP II?
 - Строка названия и меню
 - Панель инструментов и строка состояния
 - Окно структуры проекта и рабочее окно
 - Окно статуса и ошибок

ELOP II является многофункциональной программой. Доступ к различным функциям облегчается использованием удобного интуитивного пользовательского интерфейса.

3.1 Запуск ELOP II

Выберите в меню "Пуск" указатель "Программы" и там:

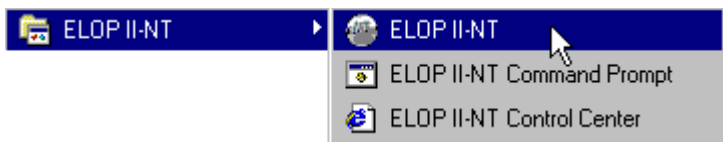


Рис. 3 Стартовое меню ELOP II

Другой способ запуска: Вы можете запустить менеджер проекта из ELOP II Control Center (Центр управления ELOP II).

3.2 Распределение экрана

После запуска ELOP II появляется стандартный экран пользовательского интерфейса (Рис. 4). Стандартный экран состоит из следующих основных элементов:

- 1 Строка названия
- 2 Окно структуры проекта
- 3 Строка меню
- 4 Панель инструментов для менеджера проекта
- 5 Рабочая область
- 6 Панель инструментов для редактора диаграммы функциональных блоков (Редактор FBD)
- 7 Окно статуса и ошибок
- 8 Строка состояния с координатами курсора в рабочей области или в окне менеджера проекта

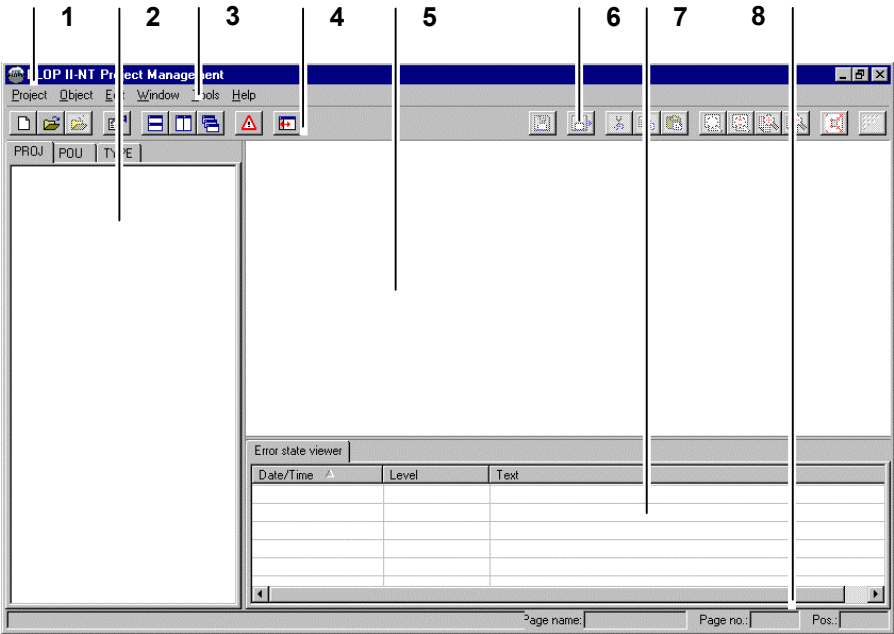


Рис. 4 Стандартный экран пользовательского интерфейса ELOP II

3.3 Строка названия

В строке названия содержится кнопка со стандартными функциями Windows (свернуть, развернуть, переместить, изменить размер и закрыть окно) и заголовок программы с информацией о проекте и обрабатываемом блоке.

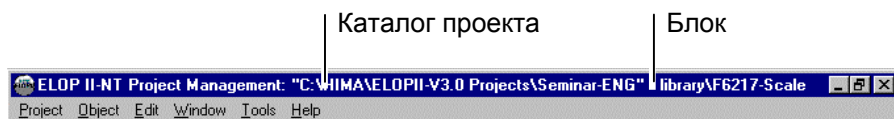


Рис. 5 Строка названия

3.4 Строка меню

С помощью строки меню осуществляется работа с ELOP II. Большинство функций ELOP II представлены в строке меню.

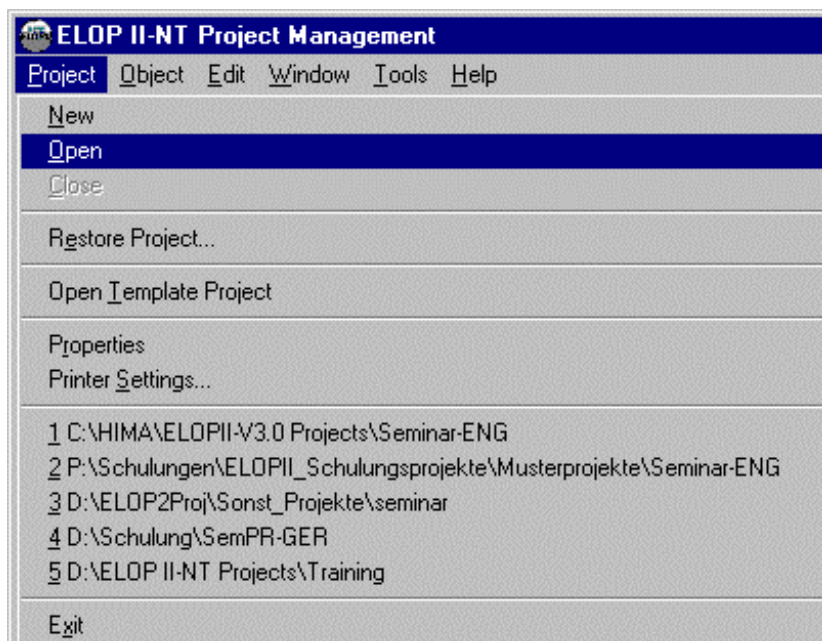


Рис. 6 Меню "Project" (Проект)

3.5 Панель инструментов

Панель инструментов находится под строкой меню и разделена на две части: "Менеджер проекта" и "Редактор диаграммы функциональных блоков".



Рис. 7 Панель инструментов проекта



Рис. 8 Панель инструментов редактора диаграммы функциональных блоков

Указание: если установить курсор на некоторое время на одну из кнопок, появится "Quick-Info" (Всплывающая подсказка).

3.6 Строка состояния

В строке состояния в нижней части окна отображается дополнительная информация и сообщения об использовании, относящиеся к менеджеру проекта и редактору диаграммы функционального блока, а также текущее положение курсора в пределах одной страницы диаграммы.

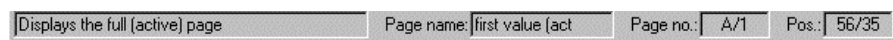
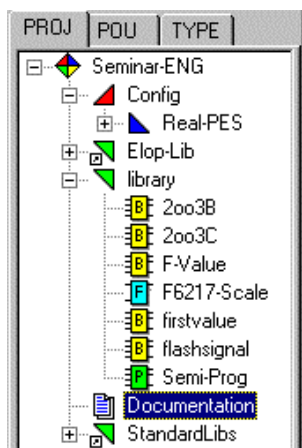


Рис. 9 Строка состояния

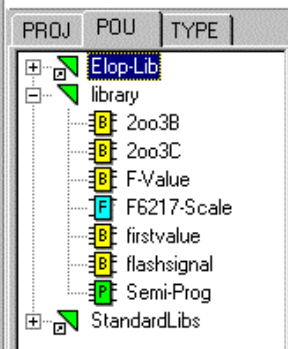
3.7 Окно структуры проекта

В окне структуры проекта показан проект со всей своей иерархической структурой. Вы можете выбрать три различных вида с различной степенью детализации.

Весь проект



Все программные
компоненты (блоки)



Все типы данных

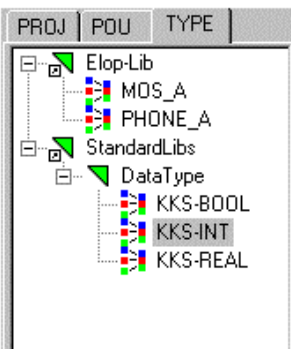


Рис. 10 Окно структуры проекта

3.7.1 Контекстное меню для объектов

Щелкнув правой кнопкой мыши на объект окна структуры проекта, Вы откроете относящееся к этому объекту контекстное меню. Команды меню выбираются, как обычно, щелчком левой кнопки мыши.



Рис. 11 Контекстное меню проекта

3.8 Рабочая область

В рабочей области (см. Рис. 4) обрабатываются объекты данных с помощью

- редактора диаграммы функционального блока и
- редактора документов.

3.8.1 Редактор диаграммы функционального блока (Редактор FBD)

С помощью редактора диаграммы функционального блока создаются функциональные диаграммы на языке функциональных блоков (FBD) или языке последовательных функциональных схем (SFC).

Редактор диаграммы функционального блока (Рис. 12) имеет следующие области:

- 1 поле обработки диаграммы
- 2 редактор описания переменных
- 3 окно общего обзора и навигации проекта
- 4 редактор деклараций интерфейсов

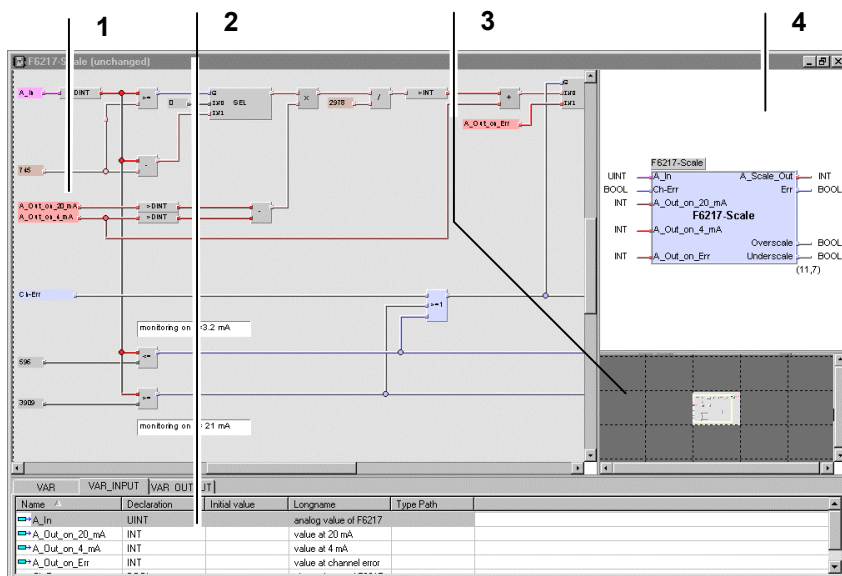
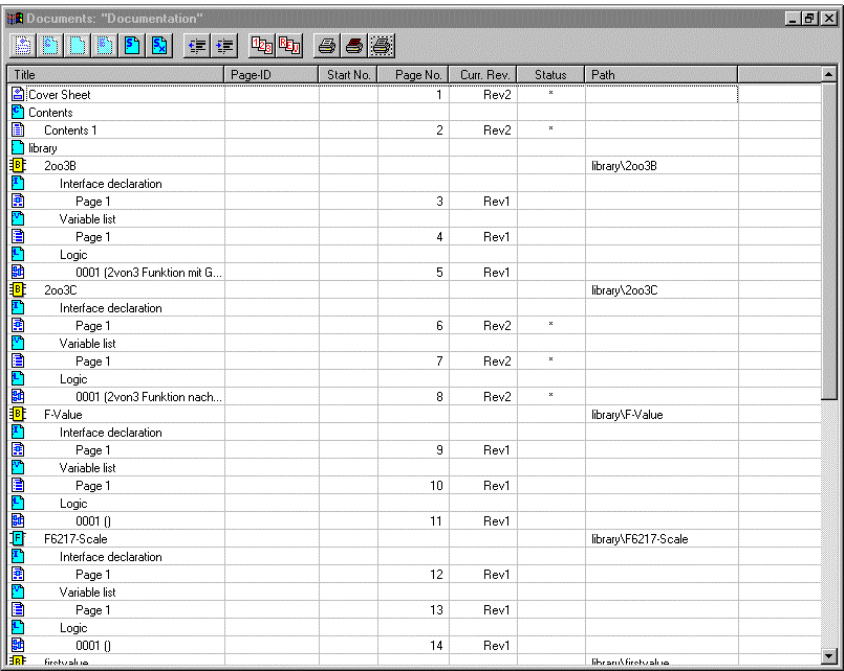


Рис. 12 Редактор диаграммы функционального блока

3.8.2 Редактор документов

С помощью редактора документов создается документация проекта из объектов, имеющихся в окне структуры проекта. Для всех документов можно использовать общее управление версиями.


Щелкните мышью на кнопке "1 2 3" на панели инструментов редактора документов, чтобы обновить оглавление.

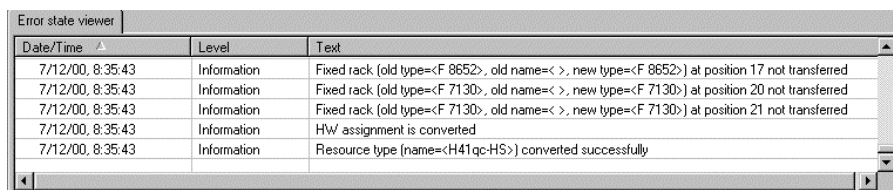


Title	Page-ID	Start No.	Page No.	Curr. Rev.	Status	Path
Cover Sheet			1	Rev2	*	
Contents						
Contents 1			2	Rev2	*	
library						
2oo3B						library\2oo3B
Interface declaration						
Page 1			3	Rev1		
Variable list						
Page 1			4	Rev1		
Logic						
0001 (2von3 Funktion mit G...			5	Rev1		
2oo3C						library\2oo3C
Interface declaration						
Page 1			6	Rev2	*	
Variable list						
Page 1			7	Rev2	*	
Logic						
0001 (2von3 Funktion nach...			8	Rev2	*	
F-Value						library\F-Value
Interface declaration						
Page 1			9	Rev1		
Variable list						
Page 1			10	Rev1		
Logic						
0001 ()			11	Rev1		
F6217-Scale						library\F6217-Scale
Interface declaration						
Page 1			12	Rev1		
Variable list						
Page 1			13	Rev1		
Logic						
0001 ()			14	Rev1		
firstvalue						library\firstvalue

Рис. 13 Редактор документов

3.9 Окно статуса и ошибок

В окне статуса и ошибок отображаются общая информация и сообщения об ошибках. О появлении нового сообщения об ошибке или завершение представления новой информации сигнализирует мигающий символ  в панели задач Windows.



Date/Time	Level	Text
7/12/00, 8:35:43	Information	Fixed rack (old type=<F 8652>, old name=< >, new type=<F 8652>) at position 17 not transferred
7/12/00, 8:35:43	Information	Fixed rack (old type=<F 7130>, old name=< >, new type=<F 7130>) at position 20 not transferred
7/12/00, 8:35:43	Information	Fixed rack (old type=<F 7130>, old name=< >, new type=<F 7130>) at position 21 not transferred
7/12/00, 8:35:43	Information	HW assignment is converted
7/12/00, 8:35:43	Information	Resource type (name=<H41qc-HS>) converted successfully

Рис. 14 Окно статуса и ошибок

3.10 Встроенная справка

Во встроенной справке Вы найдете подробные объяснения по всем функциям в ELOP II. С помощью указателя Вы можете быстро получить информацию по ключевым словам.

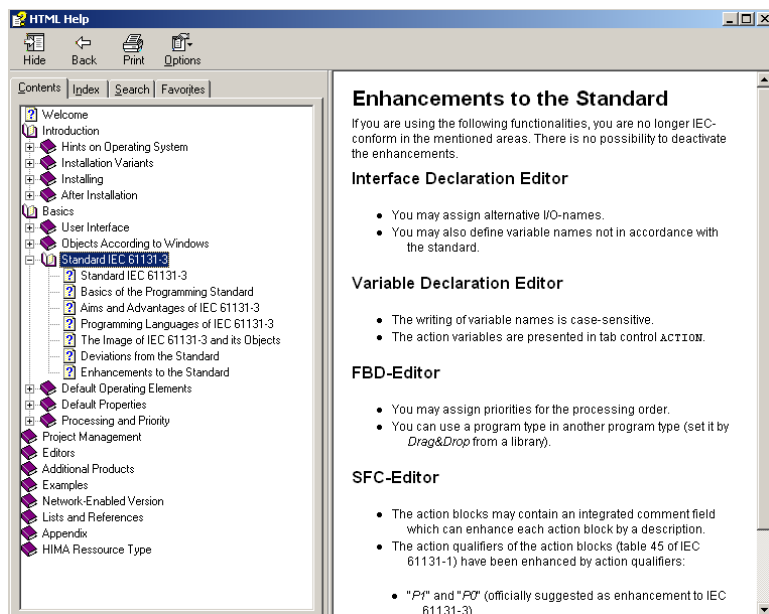


Рис. 15 Встроенная справка


4 Объекты в окне структуры проекта

В этой главе описаны следующие темы:


- Иерархическое расположение объектов в окне структуры проекта
- Значение объектов

В окне структуры проекта (см. также Рис. 10) можно просматривать и управлять всеми объектами проекта, представленными в иерархической последовательности. Управление проектом производится с помощью окна структуры проекта.


4.1 Проект

Проект (обозначен знаком ) – это объект, находящийся на высшей ступени иерархии. Все другие объекты создаются внутри проекта. В одно и то же время в ELOP II может быть открыт только один проект.


4.2 Библиотека

В библиотеке (обозначена знаком ) содержатся функции, функциональные блоки и программы. Согласно стандарту МЭК 61131-3 они также называются программными компонентами (сокращенно POU). В ELOP II содержатся стандартные библиотеки с готовым набором исходных функций. На их основе Вы можете создавать собственные, более сложные функции, функциональные блоки и программы (программные компоненты).

4.2.1 Тип программы


В типе программы (обозначен знаком ) содержатся все функции приложения. Тип программы может быть присвоен для выполнения нескольким контроллерам. При этом под контроллером понимается экземпляр (*инстанция* описанного в библиотеке *типа программы*).

4.2.2 Тип функционального блока


В типе функционального блока (обозначен знаком ) содержатся отдельные функции приложения, таким образом его можно сравнить с подпрограммой. Тип функционального блока может также использоваться для структурирования программы в соответствии со структурой управляемого оборудования. Тип функционального блока может временно сохранять значения в локальных переменных. Выходное значение зависит от входных и сохраненных значений (типичный пример: триггер, таймер).

Кроме того, с помощью типа функционального блока можно получить доступ к внешним переменным (глобальные переменные программы).


4.2.3 Функция

Под функцией (обозначена знаком ) подразумеваются базовые (элементарные и созданные пользователем) функции приложения. В отличие от типа функционального блока, функция не может сохранять текущие значения. Выходное значение зависит исключительно от входных значений (типичный пример: AND (И), OR (ИЛИ)).




4.3 Конфигурация

Конфигурация (обозначена знаком ) объединяет несколько ресурсов (контроллеров) в логическую единицу (систему контроллеров), которые могут быть связаны друг с другом для обмена информацией.


4.3.1 Ресурс

Ресурс (обозначен знаком ) – это установленное стандартом МЭК 61131-3 понятие для целевой системы, которая выполняет управляющую задачу, например, для контроллера HIMA.

4.3.1.1 Экземпляр программы


Экземпляр (инстанция) программы (обозначен знаком ) – это ссылка на уже имеющийся в библиотеке тип программы. Программа, содержащаяся в типе, выполняется в ресурсе (PES), в котором имеется ссылка на тип. Такие настройки контроллера, как задание коммуникативных и системных переменных, должны производиться в экземпляре программы , а не в типе программы .

4.3.1.2 Задача


Задача (обозначена знаком ) служит для определения порядка выполнения программ в ресурсе. ELOP II позволяет работу и с многозадачными системами, но контроллеры HIMA допускают выполнение только одной программы (= одного экземпляра программы).

Поэтому объект структуры "Задача" необходим только для автономного моделирования, так как в этом режиме можно работать с несколькими задачами. В этом случае экземпляру программы следует присвоить одну из имеющихся задач, в которой заранее задаются время цикла и другие параметры.

4.3.1.3 Дерево проекта

Дерево проекта (обозначено знаком ) необходимо для построения внутреннего алгоритма поиска. С помощью этого объекта можно повлиять на последовательность поиска; изменять его рекомендуется только опытным пользователям.

4.4 Документация

С помощью документации (обозначена знаком ) обеспечивается подбор подлежащих документированию объектов с помощью простого переноса. Существует возможность применения управления версиями для всех имеющихся документов.

5 Концепция базовой системы

В этой главе описаны следующие темы:

- Менеджер проекта
- Создание функций
- Управление документацией

В этой главе рассказывается об основных концепциях ELOP II. Практические примеры находятся в главе 7.

5.1 Менеджер проекта

Менеджер проекта является центральным инструментом для работы с ELOP II. В окне структуры проекта отображается папка проекта (см. Рис. 10, слева).

Вы можете

- открывать или закрывать проект (через строку меню или кнопки на панели инструментов)
- архивировать или восстанавливать проект (через контекстное меню)

С помощью шаблона проекта, который также можно открыть через строку меню, Вы можете воспользоваться многочисленными предварительными настройками для создания Вашего проекта.

Открытый проект показывается в окне структуры проекта.

5.2 Библиотеки функциональных блоков

Проект может состоять из любого количества библиотек, и, следовательно, из любого количества блоков.

Библиотеки могут создаваться в проекте, конфигурации, ресурсе или же в другой библиотеке.

Благодаря этому Вы можете выполнить соответствующее логике структурирование библиотек.

Основные функции МЭК 61131-3 содержатся в библиотеке "StandardLibs" (Стандартная библиотека), которая автоматически включается при создании нового проекта.

Библиотека "EOP-LIB" содержит специальные функциональные блоки HIMA, преимущественно для обработки сигналов В/В (ввода/вывода).

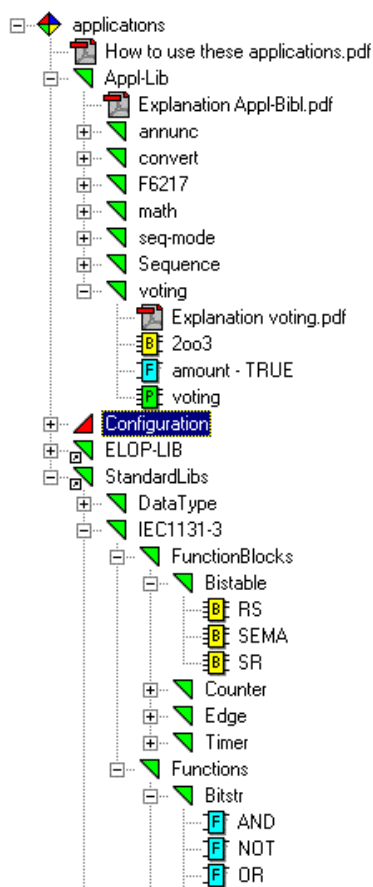


Рис. 16 Библиотеки в окне структуры проекта

5.3 Редактор диаграммы функционального блока

Редактор диаграммы функционального блока запускается при открытии любого программного компонента (функции, функционального блока или программы). Области

- область диаграммы
- редактор деклараций интерфейсов
- редактор описаний переменных
- окно обзора

представляются в одном окне в рабочей области менеджера проекта таким же образом, как в последнем сохраненном проекте.

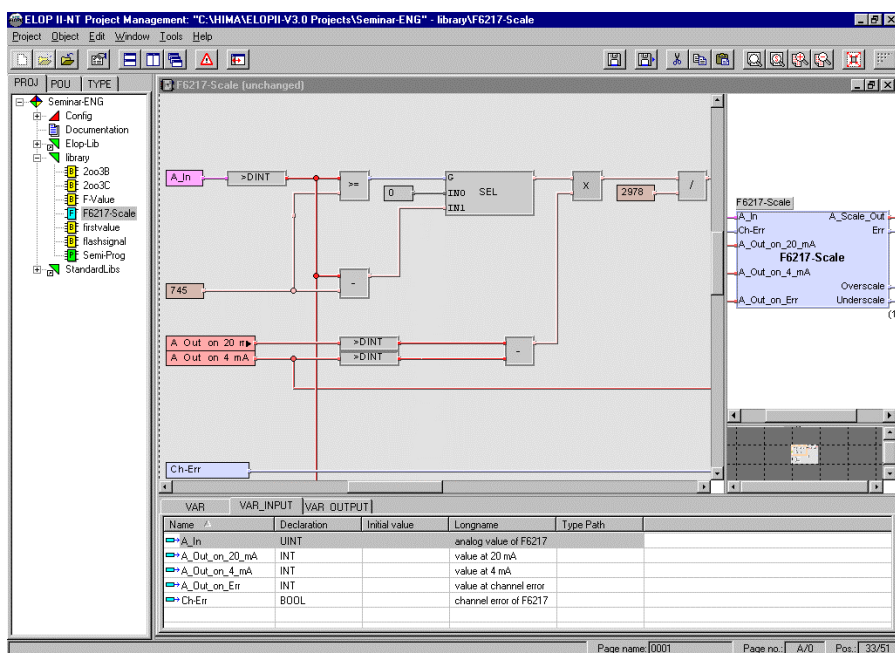




Рис. 17 Редактор диаграммы функционального блока в рабочей области

Для перехода между развернутой рабочей областью и общим обзором предусмотрены специальные функции.

5.3.1 Развертка рабочей области

Окно структуры, представленное на Рис. 17, Вы можете скрыть или показать с помощью кнопки , находящейся в левой части панели инструментов. То же самое относится к окну статуса и ошибок с кнопкой .

Благодаря этому можно увеличить рабочую область для редактора диаграммы функционального блока.

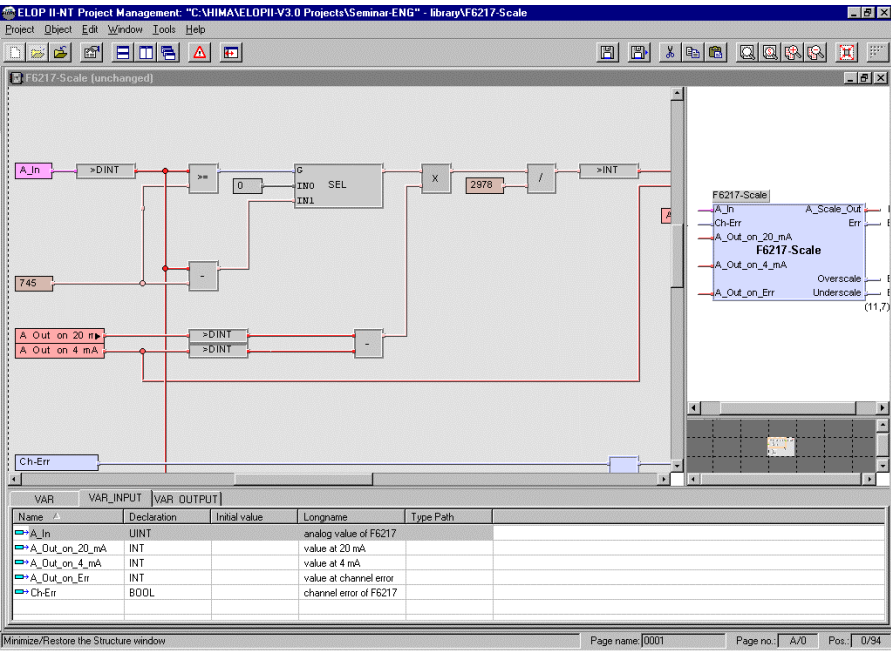



Рис. 18 Редактор FBD в развернутой рабочей области

Указание: размер рабочей области можно дополнительно настраивать переносом границы окна редактора описания переменных и окна редактора деклараций интерфейсов.

5.3.2 Развертка и восстановление

Щелкнув на одно из окон в рабочей области (Рис. 18), Вы активируете это окно. С помощью кнопки  можно развернуть активное окно.

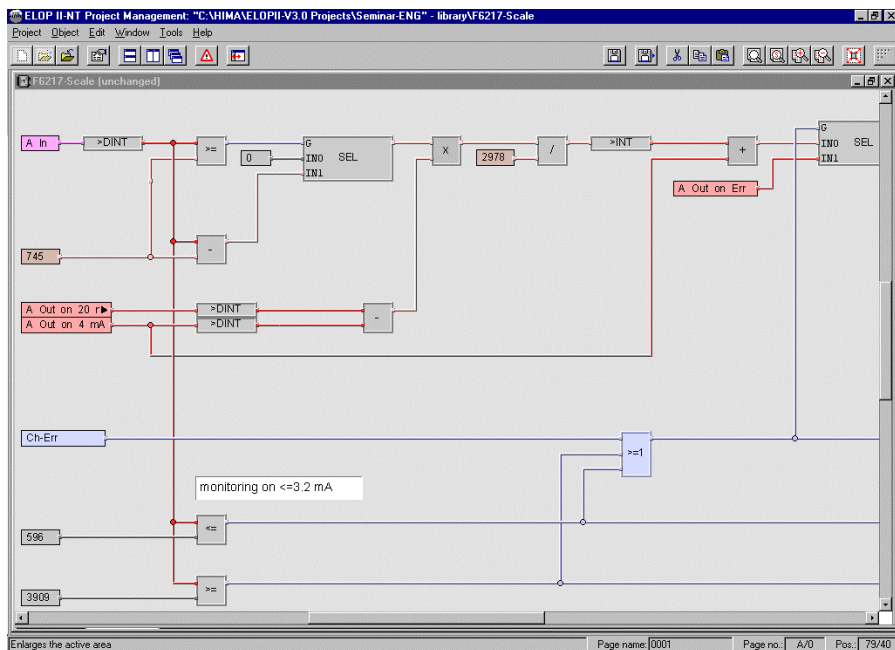


Рис. 19 Область диаграммы развернута

При повторном нажатии этой кнопки области редактора диаграммы функционального блока вернутся к своему прежнему размеру.

5.4 Функциональные диаграммы с центральной начальной страницей

Концепция ELOP II позволяет обойтись без дополнительной вставки отдельных страниц благодаря тому, что центральная начальная страница может быть расширена в любую сторону без какого-либо ограничения.

Позиция страницы указывается с помощью системы координат. Столбцы обозначаются прописными буквами, а строки – цифрами.

По умолчанию первой страницей является страница с координатами A/0. Как только на эту страницу помещается какой-либо элемент, она становится активной.

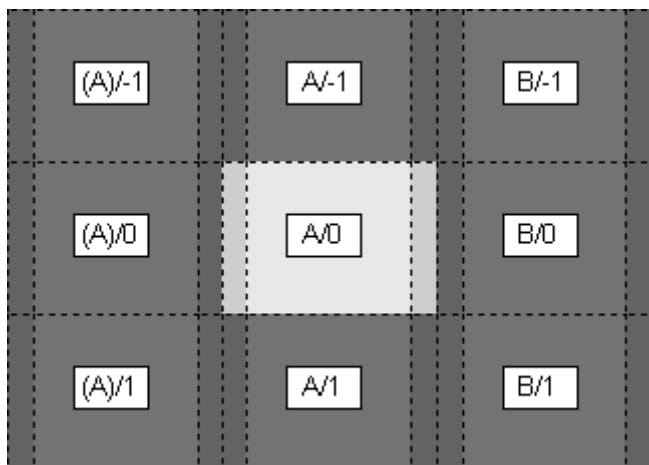


Рис. 20 Нумерация страниц в функциональной диаграмме

Активная страница выделена подсветкой. Как только элемент будет помещен на соседнюю страницу, она тоже становится активной и выделяется подсветкой. Таким образом функциональную диаграмму можно продолжать в любом направлении.

Указание: если Вам необходимо вставить страницу между уже имеющимися страницами в диаграмме, Вы можете передвинуть одну из страниц. Для этого в контекстном меню страницы под пунктом "Plug-Ins" (Подключаемые модули) следует выбрать пункт "Move page" (Переместить страницу).

Окно обзора отображает общий вид функциональной диаграммы. Вы можете перемещаться между страницами, выбирая их с помощью щелчка кнопки мыши.

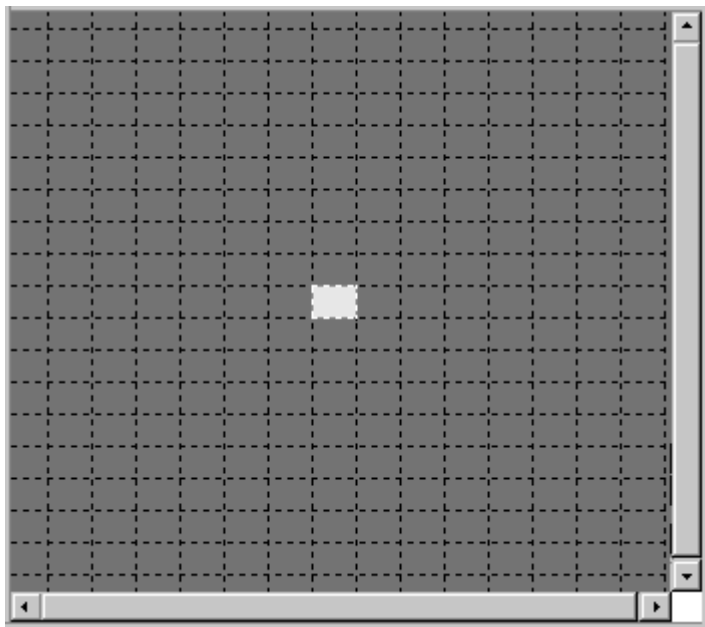



Рис. 21 Окно обзора

Указание: в контекстном меню страницы под пунктом "Plug-Ins" (Подключаемые модули) находится пункт "Page list" (Список страниц). С его помощью Вы также можете переходить к нужным страницам.

5.4.1 Масштабирование

Используя кнопки , расположенные на панели инструментов, Вы можете уменьшать или увеличивать содержимое области диаграммы и окна обзора.

5.5 Создание логики

С помощью этих основных функций Вы можете создавать логику в функциональной диаграмме.

5.5.1 Работа с переменными

Переменные создаются в редакторе описаний переменных.

Существуют следующие типы переменных:

Локальные переменные (VAR) определены в программе, функции или в функциональном блоке и могут быть использованы только в данном программном компоненте. Локальные переменные в программе могут быть связаны с аппаратным обеспечением (например, DI или AI, см. VARxx).

Переменные интерфейса (VAR_INPUT, VAR_OUTPUT) могут быть определены только в функции или в функциональном блоке. Переменные интерфейса непосредственно связывают окружающую (внешнюю) и внутреннюю логику функции или функционального блока. При вызове функции или функционального блока производится перенос внешних значений во внутреннюю логику с помощью переменных VAR_INPUT. Результаты внутренней логики записываются в вызывающем программном компоненте с помощью переменных VAR_OUTPUT (Рис. 22, FBX).

Глобальные переменные (VAR_GLOBAL) могут быть заданы только в программе и использованы в любом функциональном блоке (FB), где они определяются как **VAR_EXTERNAL** с точно таким же именем переменной.

Пример:

На Рис. 22 переменная GE2 создается в функциональном блоке FB Y, а также используется в функциональном блоке FB X. В связи с этим, эта переменная должна быть задана в обоих функциональных блоках как VAR_EXTERNAL.

Кроме того, переменная GE2 должна быть определена как VAR_GLOBAL в программе и может использоваться и там.

Указание:

Глобальные переменные могут быть связаны с аппаратным обеспечением (например, DI, AI, DO, AO).

Ввод начальных значений для VAR_GLOBAL возможен только в программе.

Вывод на принтер списка перекрестных ссылок, где указаны все использования VAR_GLOBAL / VAR_EXTERNAL, можно выполнить из контекстного меню ресурса.

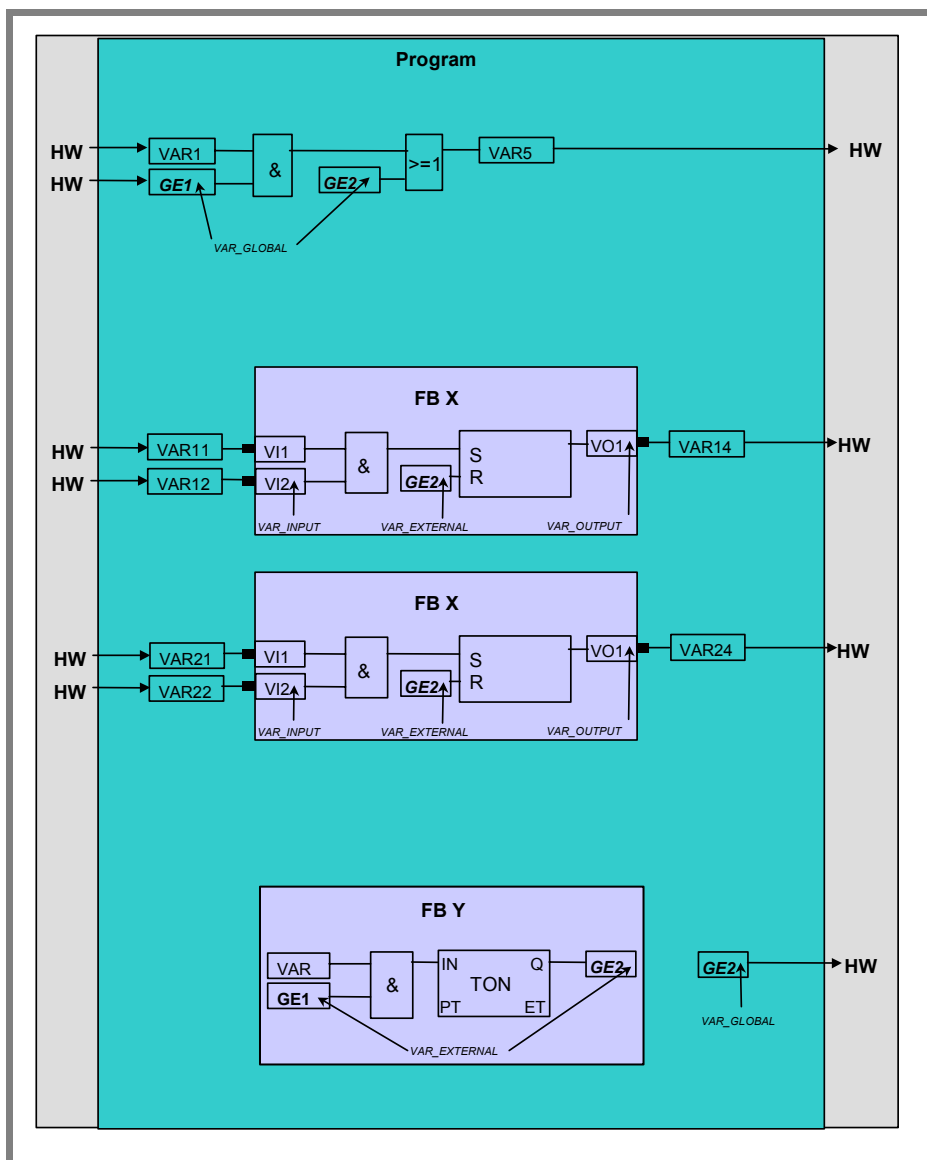


Рис. 22 Работа с переменными

5.5.2 Перенос переменных

Переменные создаются в редакторе описаний переменных, где можно выбрать разные типы переменных (локальные переменные, входные переменные, выходные переменные, внешние переменные). Чтобы использовать переменные в логике, щелкните на переменную в редакторе описаний переменных и перенесите ее в требуемую позицию области диаграммы, в ней будет создано поле значения с именем переменной. Пиктограмма в редакторе описаний переменных, расположенная слева от переменной, показывает вид использования переменной в области диаграммы.

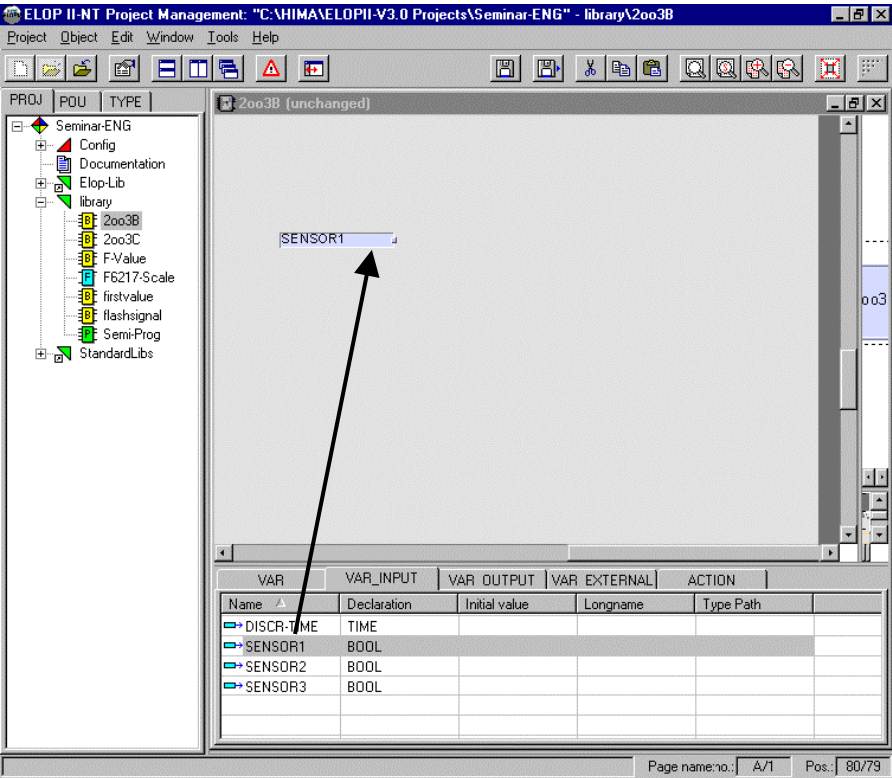


Рис. 23 Перенос переменных в область диаграммы

Указание: переменные можно импортировать также из внешнего источника данных. См. встроенную справку.

5.5.3 Перенос функциональных блоков

Чтобы использовать функциональный блок в логике, выберите функциональный блок в библиотеке, находящейся в окне структуры, и перенесите его в требуемую позицию области диаграммы.

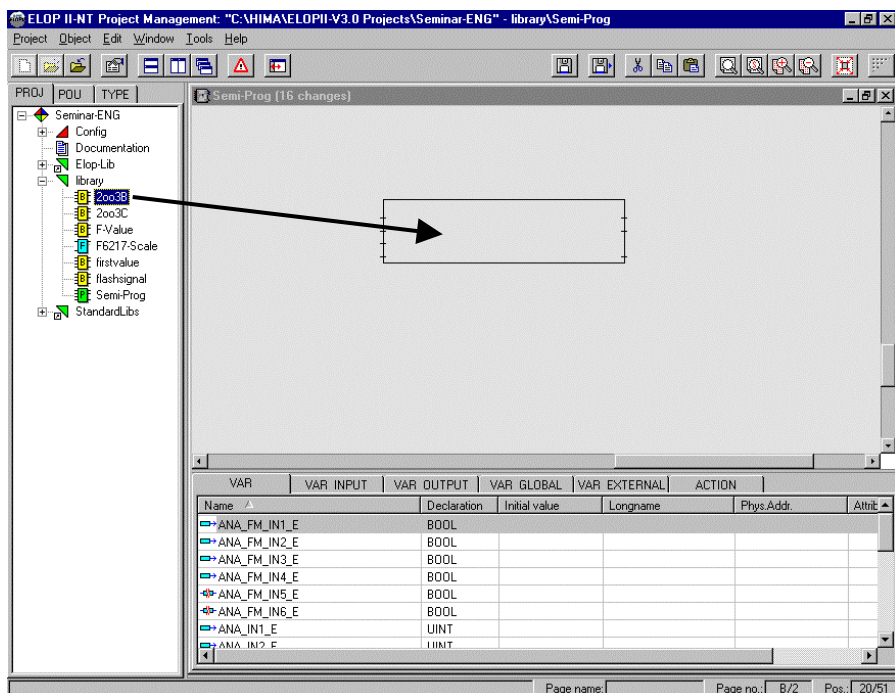


Рис. 24 Перенос программного компонента в область диаграммы

5.5.4 Соединение элементов

Элементы, расположенные в области диаграммы, можно соединить друг с другом с помощью линий между точками связи. Обратите внимание на то, что существуют точки связи для входа и выхода, которые не всегда можно соединить между собой любым образом.

Для создания соединения подведите курсор к тому месту, где должна начаться линия связи (например, выход функционального блока). Нажмите левую клавишу мыши и проведите линию связи до нужной Вам точки связи, не отпуская клавиши мыши. Теперь, если Вы отпустите клавишу мыши, возникнет линия, связывающая эти точки. Вы можете также создавать узловые точки, соединяя уже существующую линию с другим входом или выходом.

Указание: ELOP II автоматически распределяет линии, но при необходимости пользователь может изменить их вручную, переноса необходимый сегмент линии указателем мышки и удерживая при этом нажатой клавишу SHIFT.

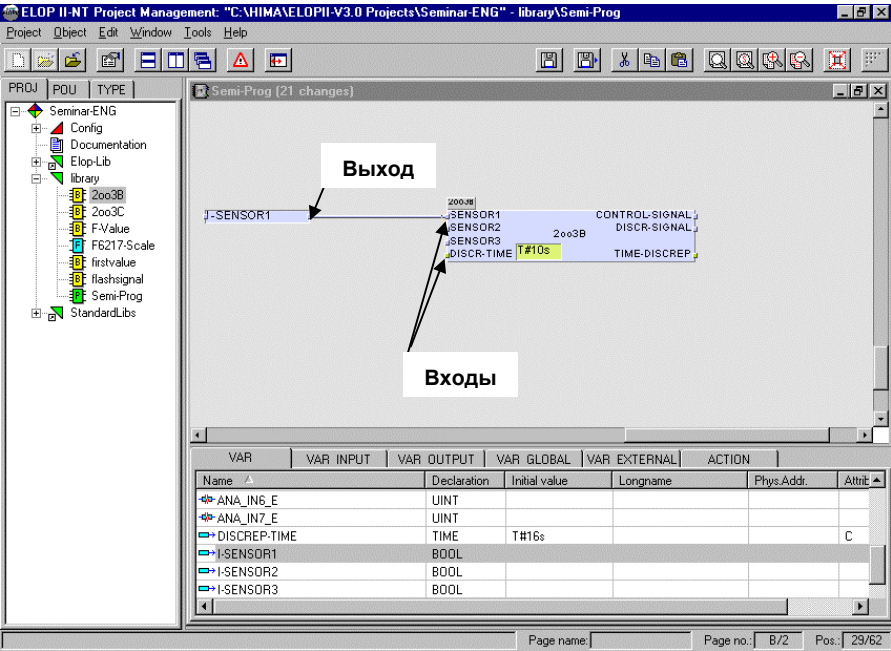


Рис. 25 Создание линии между точками связи

5.6 Создание документации и управление ею

ELOP II обеспечивает полное управление документами, предоставляя сервис по работе с версиями.

Благодаря этому Вы можете создавать различные версии документов.

Управление документами может не только полностью распечатать документы с помощью сервиса по работе с версиями, но и обнаружить изменения, сделанные на отдельных страницах Ваших документов, что поможет Вам распечатать только эти измененные страницы.

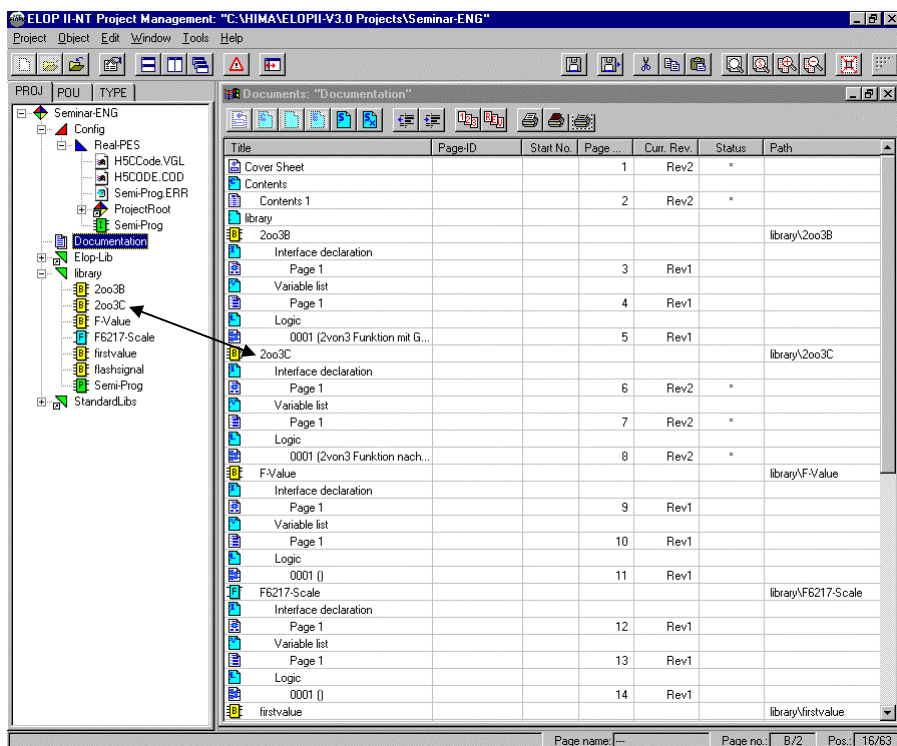


Рис. 26 Управление документами

5.6.1 Шаблоны для печати документов

При печати документов в качестве шаблонов используются формуляры формата DXF. В ELOP II содержится стандартный набор формуляров для печати всех объектов.

Weitergabe sowie Vervielfältigung dieses Dokuments, Verwertung und Mitteilung seines Inhalts sind verboten, soweit nicht ausdrücklich gestattet. Zuwiderhandlungen verpflichten zu Schadenersatz. Alle Rechte für den Fall des Patents-, Gebrauchsmuster- oder Geschmacksmustereintrags vorbehalten.

Deckblatt		D1	
ELOP II-NT Projektdokumentation			
Auftraggeber:	D2	Status / Inderung	Datum
	D3	D13	D14
Auftrags-Nr.:	ORDNOCHST	D16	D17
	D5	D19	D20
Projektbezeichnung:	D6	D22	D23
	D7	D25	D26
	D8	D28	D29
Endkunde:	D9	D31	D32
	D10	D34	D35
	D11	D37	D38
Projektbearbeiter:	D12	D40	D41
			D42

NAME: _____

24

ORNO

Рис. 27 Формуляр DXF для титульного листа

В свойствах объектов Вы можете изменить как стандартные записи для шаблонов, так и содержание отдельных полей в формулярах DXF.

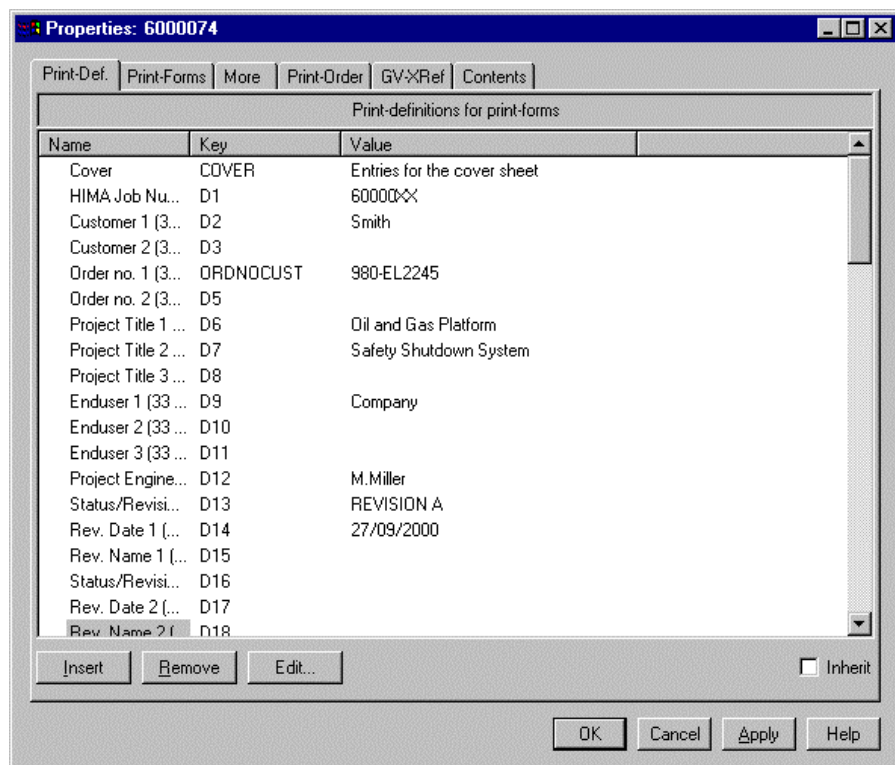


Рис. 28 Параметры настройки проекта для вывода на печать

5.7 Автономное моделирование логики

С помощью автономного моделирования Вы можете проверить на компьютере правильность логики функциональных диаграмм, не используя программируемую электронную систему (PES = PLC). Для этого функциональные диаграммы ELOP-II компилируются и выполняются компьютером в течение предварительно установленного времени цикла.

Автономное моделирование может выполняться только для экземпляров программы в пределах одного ресурса. Для этого в ресурсе необходимо создать объект задачи и присвоить его экземпляру программы.

Автономное моделирование позволяет проследить работу функциональной диаграммы в действии. С помощью диалоговых тестовых полей (полей OLT) Вы можете просмотреть отдельные значения в любом месте функциональной диаграммы. Для булевых значений линии дополнительно отображаются красным и синим цветом.

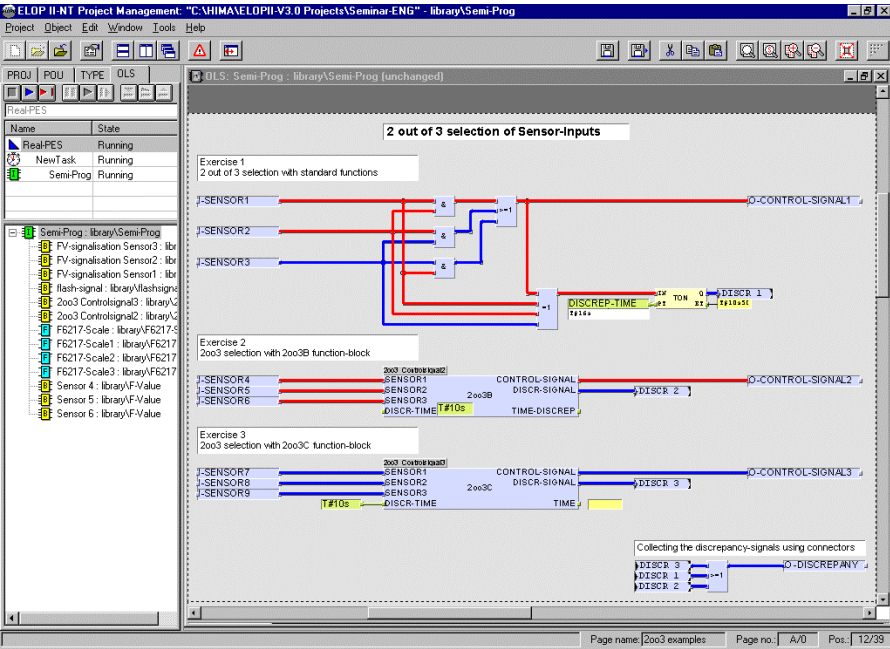


Рис. 29 Автономное моделирование функциональной диаграммы

6 Работа с типами ресурсов

В этой главе описывается:

- Присвоение типов контроллеров и В/В (Входов/Выходов)
- Генерация кодов
- Загрузка программы в контроллер
- Онлайн-тест
- Документация по аппаратному обеспечению

После того, как функциональная диаграмма была создана и протестирована независимо от аппаратного обеспечения, производится ее присвоение к определенному ресурсу.

Согласно стандарту МЭК 61131-3 ресурс – это система, выполняющая программу и обслуживающая уровень В/В.

Тип системы выбирается из списка всех имеющихся типов ресурсов.

6.1 Определение ресурса

Выберите в контекстном меню ресурса тип контроллера.

Доступны все типы контроллеров семейства систем H11, H41/H51 и H41q/H51q, а также A1. При необходимости позже можно изменить присвоение с помощью пункта меню "Hardware change" (Изменить аппаратное обеспечение).



Рис. 30 Выбор типа ресурса

Указание: при вводе имени ресурса следует учесть, что максимальная длина имени для контроллеров HIMA составляет восемь символов!

6.2 Присвоение типа программы ресурсу

Прежде чем ресурс может выполнять программу, следует присвоить ему тип программы. Выберите в диалоговом окне тип программы из библиотеки.

Для этого откройте контекстное меню этого ресурса и выберите тип программы в меню "New" (Новая).

После этого в окне структуры ресурса появляется в качестве экземпляра программы *тип* программы с тем же именем.

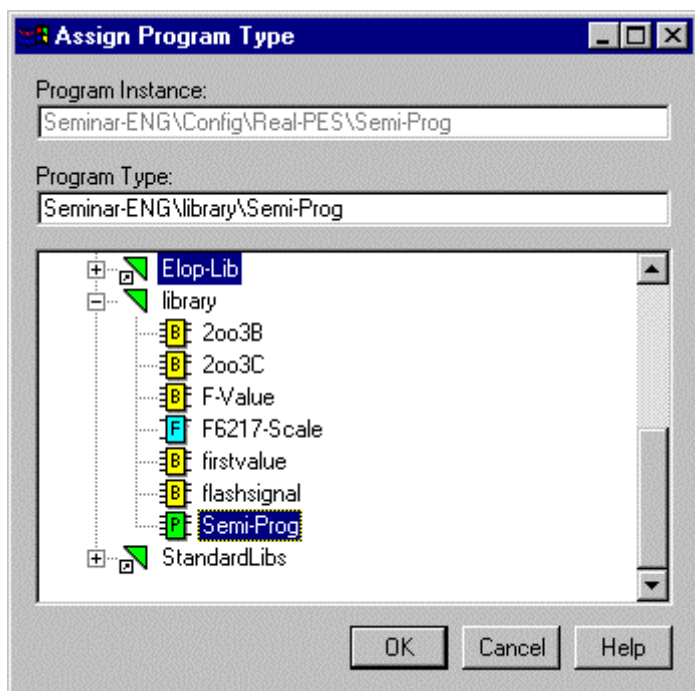


Рис. 31 Присвоение типа программы ресурсу

6.3 Присвоение В/В (Входа/Выхода)

Для того, чтобы присвоить переменные модулям В/В, сначала требуется расположить все типы модулей в коммутирующем шкафу (контекстное меню выбранного ресурса) в соответствии с построением аппаратного обеспечения.

Модули и каркасы В/В выбираются из списка, находящегося в нижней части диалогового окна (Рис. 32) и переносятся в соответствующую позицию схемы коммутирующего шкафа.

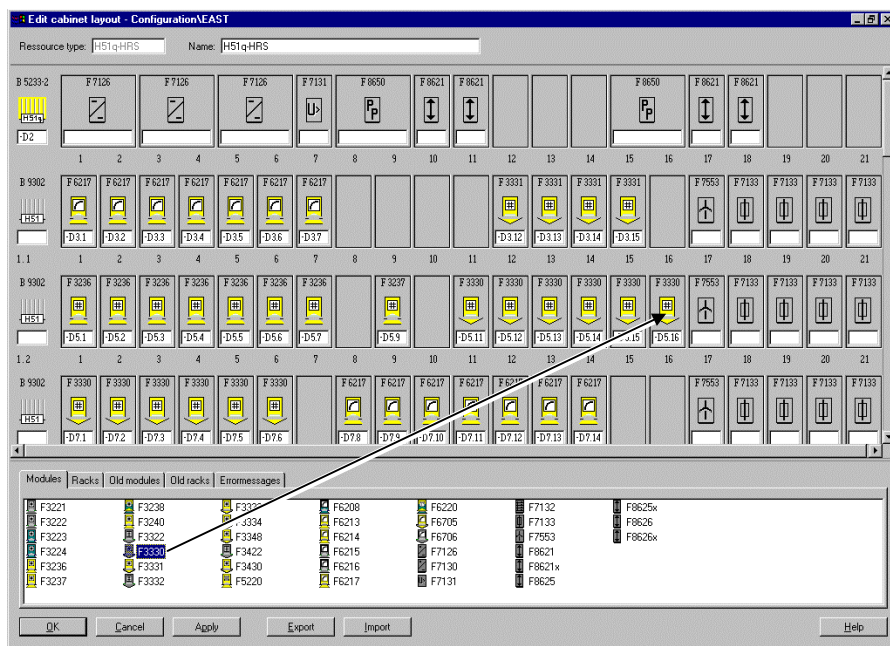


Рис. 32 Выбор и распределение В/В

После двойного щелчка по пиктограмме модуля в шкафу открывается диалоговое окно для присвоения переменных каналам В/В.

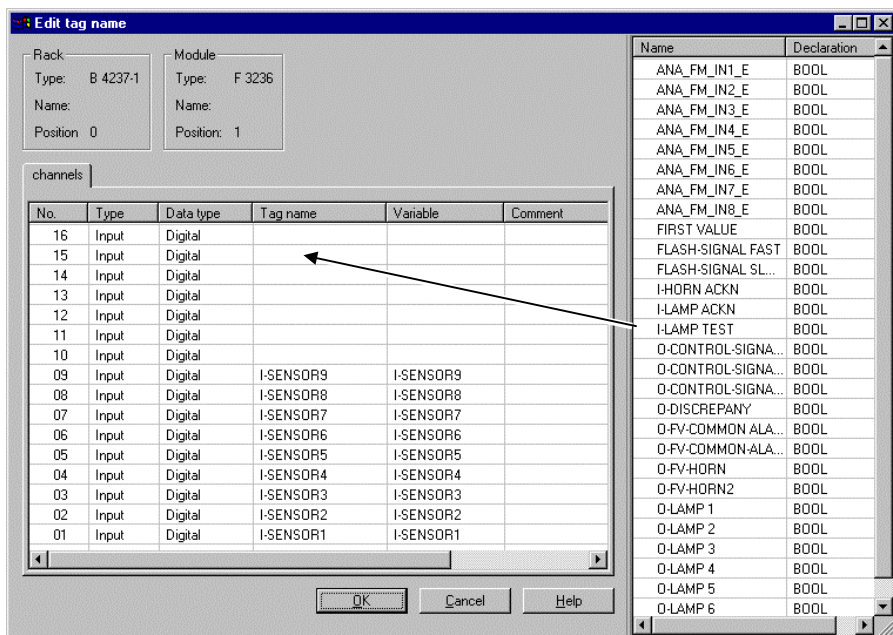


Рис. 33 Присвоение переменных каналам В/В

Выбрать можно только те переменные, тип данных которых подходит к имеющимся каналам В/В.

Модуль F3236, представленный на Рис. 33 является 16-канальным модулем входа для двоичных сигналов. Для 16 каналов входа (№ 01 ... 16), находящихся в левом списке, определен **тип данных "Digital"** (Цифровой). Булевы переменные, описанные в логике (ANA_FM_IN1_E, ANA_FM_IN2_E и т.д.) представлены в левом списке.

Присвоения выполняются с помощью переноса переменных из правого списка переменных в поле **Tag name (Имя признака)** левого списка.

6.4 Присвоение системных переменных

В дополнение к переменным, присвоенным к модулям В/В, в центральном процессоре имеются системные переменные с такой информацией о статусе, как "одноканальный режим" или "ошибка В/В" для использования в определенной пользователем логике. Эти системные переменные могут быть также переданы в системы DCS (DCS = Data Control System, система управления данными), системы сигнализации или в другие системы визуализации.

Присвоение системных переменных переменным, определенным в логике, выполняется с помощью редактора описаний переменных в экземпляре программы.

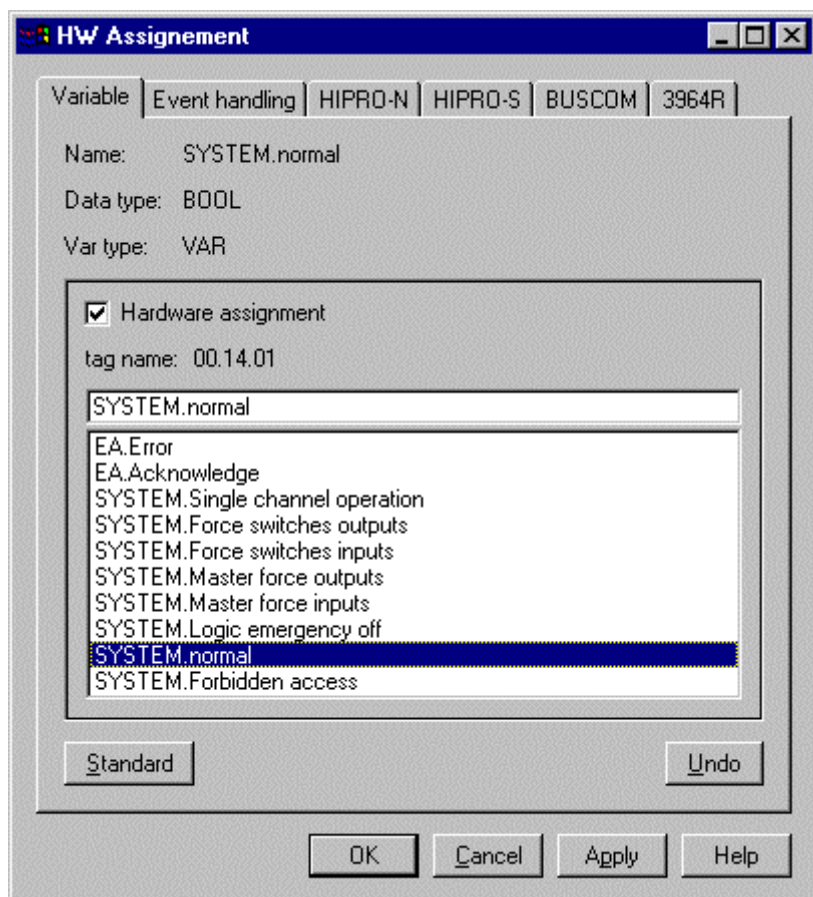


Рис. 34 Присвоение системных переменных

6.5 Генератор кода

Генератор кода переводит графическую информацию логики в код, выполняемый PES (программируемой электронной системой) и генерирует однозначную версию кода. Генератор кода запускается из контекстного меню ресурса.

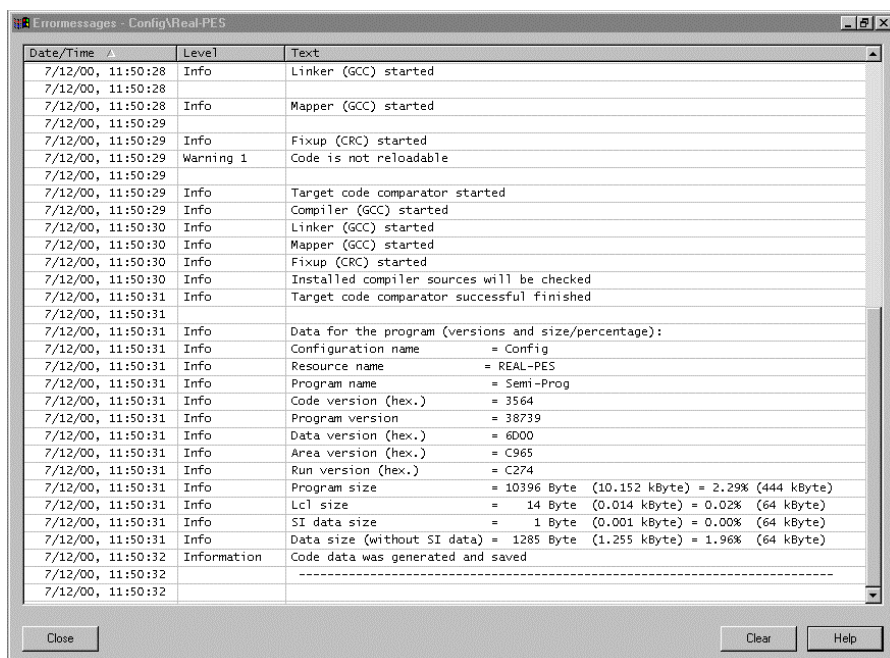


Рис. 35 Сообщения генератора кода

Сообщения генератора кода выводятся в отдельное окно и записываются в файл RESOURCE.ERR, который отображается в окне структуры под ресурсом.

6.5.1 Программа сравнения кода

Программа сравнения кода сравнивает два скомпилированных проекта на основании версий кода и выводит различия в форме списка. Результат программы сравнения кода проходит проверку на безопасность и позволяет в дальнейшем производить проверку изменений программного обеспечения без выполнения полных функциональных тестов.

6.6 Панель управления

Прежде чем загрузить в контроллер машинный код, составленный генератором кода, необходимо определить коммуникационную шину в объекте конфигурации окна структуры (в том числе и для соединений "точка-точка") и выбрать в ресурсе имеющуюся коммуникационную шину между компьютером и ресурсом.

После того, как коммуникационная шина была присвоена ресурсу, закладка панели управления CP появляется в окне структуры (Рис. 36).

Используйте панель управления для загрузки кода в контроллер и запуска выполнения.

Кроме того, такие дополнительные инструменты, как "Status display" (Просмотр статуса) и "Online change" (Изменение онлайн) позволяют запрашивать информацию о статусе контроллера или же инициировать вход и выход.

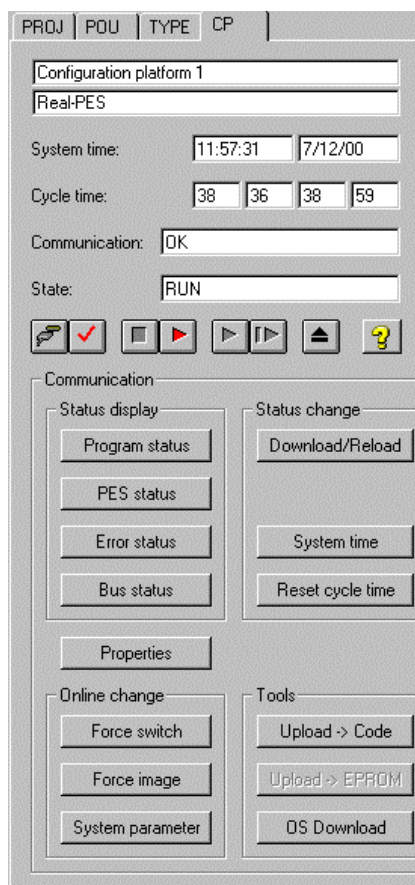


Рис. 36 Панель управления

6.6.1 Загрузка/Перезагрузка

Используйте кнопку "Download/Reload" (Загрузка/Перезагрузка) для загрузки компилированной программы в контроллер.

Загрузка имеет следующие свойства:

- Загрузка выполняется для загрузки программного кода в пустой центральный процессор.
- Загрузка необходима в том случае, если были выполнены серьезные изменения в текущем приложении.
- После старта загрузки работающий контроллер останавливается для приема программы и впоследствии запускается вручную в режиме холодного старта. Все обрабатываемые данные будут утеряны!

Перезагрузка имеет следующие свойства:

- Перезагрузка может быть выполнена в том случае, если необходимо произвести небольшие изменения в загружаемом приложении.
- Изменения в приложении могут быть выполнены без остановки контроллера.
- Контроллер должен иметь два центральных процессора. В этом случае в оба центральных процессора поочередно загружается программный код, что гарантирует работоспособность контроллера во время перезагрузки.
- Следует разрешить генерацию перезагружаемого кода в свойствах ресурса на закладке **Codegenerator (Генератор кода)**.

В диалоговом окне Вы можете задать дополнительные параметры загрузки.

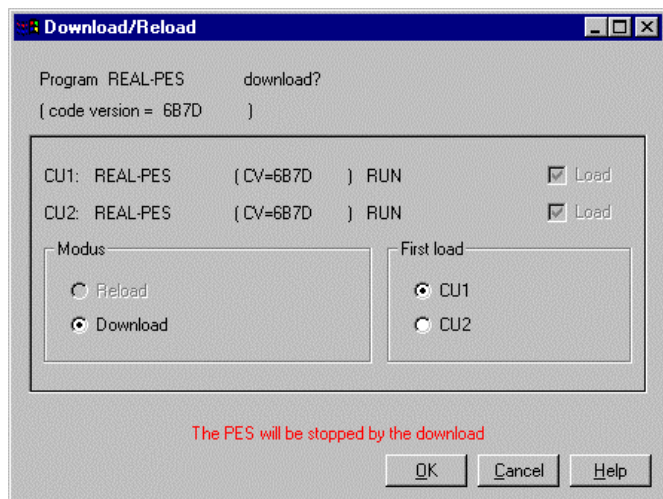


Рис. 37 Диалоговое окно для запуска загрузки/перезагрузки

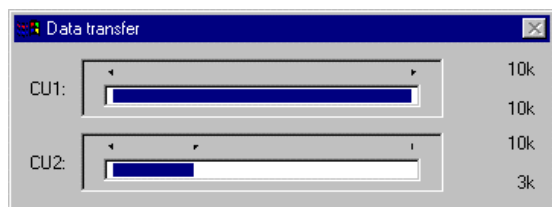



Рис. 38 Статус передачи данных во время загрузки

После успешной загрузки нового программного кода следует перезапустить контроллер, нажав кнопку  (за исключением перезагрузки).

Внимание: в отличие от перезагрузки, загрузка ведет к остановке контроллера, а тем самым – к остановке системы! Загрузка или перезагрузка должна выполняться только квалифицированными и опытными пользователями.

6.7 Онлайновый тест

Онлайновый тест (OLT) – это отдельная закладка в окне структуры. В отличие от независящего от аппаратного обеспечения автономного моделирования (см. главу 5.7), для выполнения которого нужен только компьютер, онлайновый тест считывает актуальные значения переменных с контроллера.

Функциональная диаграмма используется при этом для отображения значений и статусов состояния.

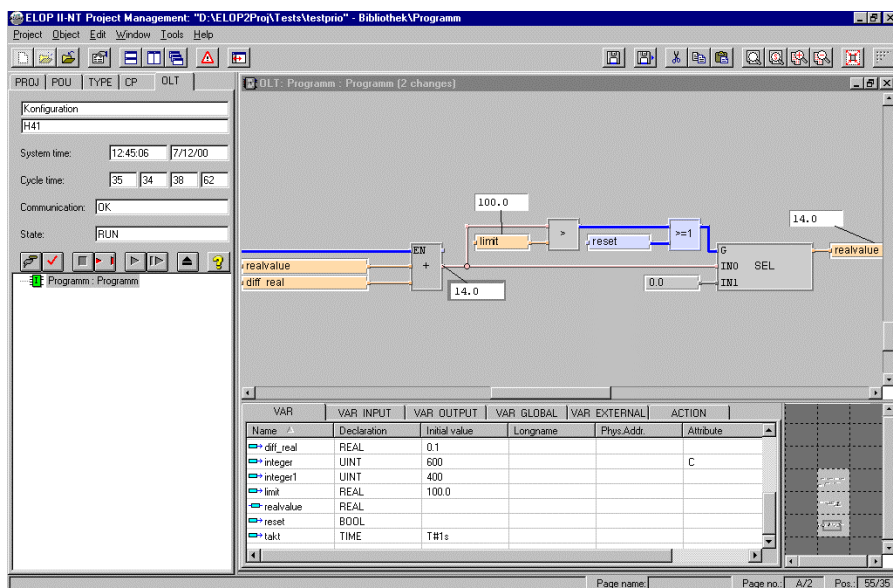


Рис. 39 Онлайновый тест функции

Указание: онлайновый тест можно выполнить только в том случае, если версия кода проекта, загруженного в компьютер, идентична версии кода контроллера. Если проект утерян, например, из-за поломки компьютера и т.д., и нет резервной копии проекта, то онлайновый тест этого проекта невозможен. Обратная загрузка проекта из PES в компьютер невозможна!

6.8 Документация по аппаратному обеспечению

Выберите документацию аппаратного обеспечения соответствующего ресурса. Вы можете экспортировать эту документацию в текстовый файл (формат ASCII) и вставить его в объект документа.

The screenshot shows a software window titled "RES docu generated - Konfiguration\H41". It features a tabbed interface with tabs for "Cabinet", "Rack", "Module", "Parameter", "BUSCOM", "3964R", "HIPRO-N/S", "Lcl", "CRF", and "CG error". The "Parameter" tab is active, displaying a table of configuration parameters. The table has two columns: "Parameter" and "Value". The parameters are grouped into sections: "Cabinet and central module", "Safety parameter", and "Test operation". The values for these parameters include "H41qc-HS", "2", "Mono", "Redundant", "52", "5", "1000", "6", "Not allowed", "Allowed", "Allowed", "Allowed", "Allowed", "Allowed", and "Allowed". At the bottom of the window, there are three buttons: "Close", "Export", and "Print", and a "Help" button.

Parameter	Value
Cabinet and central module	
Ressource type:	H41qc-HS
Name:	H41qc-HS
Amount of power supply units:	2
IO bus:	Mono
CU:	Redundant
CU at CU1:	
CU at CU2:	
CU type:	52
Safety parameter	
Safety time (s):	5
Watchdog (ms):	1000
Safety requirement class (AK):	6
Test operation:	Not allowed
Start:	Allowed
Constant change:	Allowed
Variable change:	Allowed
Force IO:	Allowed
Reload:	Allowed
Online change of parameters p...	Allowed

Рис. 40 Документация по аппаратному обеспечению


7 Упражнения

7.1 Создание проекта

Шаг 1: Запустите ELOP II:

- ☐ Щелкните на **ELOP II** в меню "Пуск" Windows и выберите **ELOP II**.

Шаг 2: Создайте новый проект:

- ☐ В меню **Project (Проект)** выберите опцию **New (Новый)** или щелкните клавишей мыши по символу  на панели инструментов.

Шаг 3: Задайте путь и имя проекта:

- ☐ Выберите в левой стороне Рис. 41 каталог, в котором должен быть создан новый проект, введите имя проекта в поле **Object name (Имя объекта)** и щелкните на **OK**.

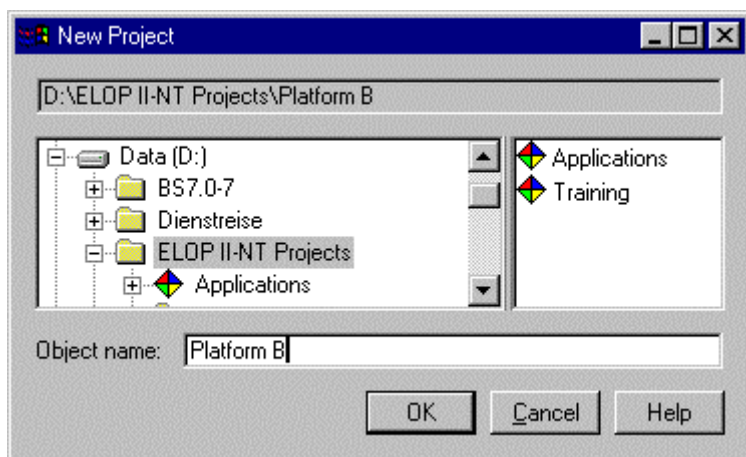


Рис. 41 Создание проекта

Новый проект появился в окне структуры. Все стандартные библиотеки автоматически были включены в новый проект.

- ☐ Щелкните на "+" перед именем проекта и откройте папку.

Шаг 4:

Создайте новую библиотеку в качестве первого элемента проекта. В этой библиотеке будут содержаться созданные пользователем функциональные блоки:

- ☐ Щелкните на проект в окне структуры.
- ☐ Откройте контекстное меню, щелкнув правой клавишей мыши.
- ☐ Выберите **New, Library (Новая, Библиотека)**.

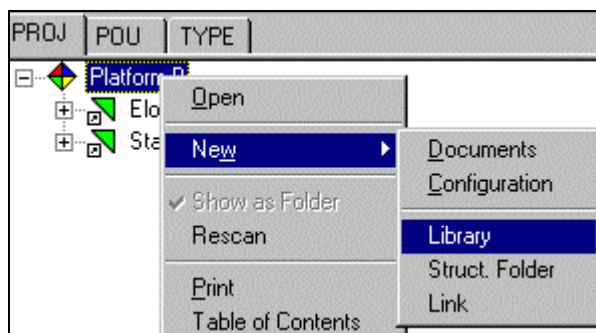


Рис. 42 Создание библиотеки

К проекту добавлена новая библиотека с именем "NewLib".

Шаг 5:

Измените имя библиотеки "NewLib" на "Prog_Library":

- ☐ Щелкните на "NewLib" в окне структуры.
- ☐ Откройте контекстное меню библиотеки "NewLib".
- ☐ Выберите **Rename (Переименовать)** и измените имя на "Prog_Library".

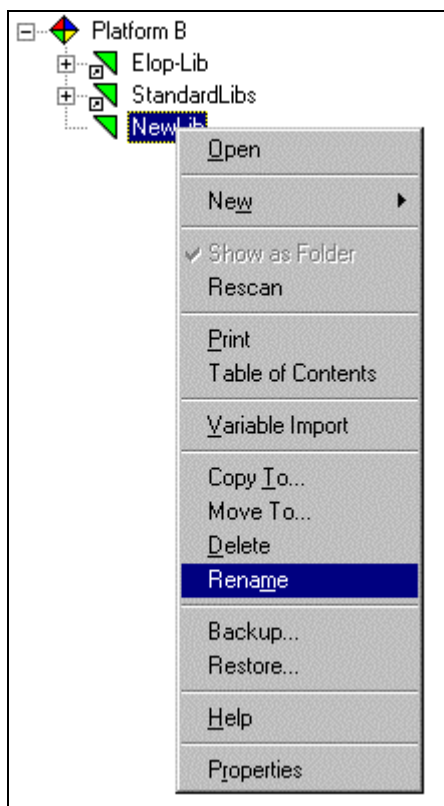


Рис. 43 Переименование библиотеки

Шаг 6:

Создайте Ваши собственные функциональные блоки в библиотеке "Prog_Library":

- ☐ Щелкните на "Prog_Library".
- ☐ Откройте контекстное меню.
- ☐ В меню **New (Новый)**, выберите **Program Type (Тип программы)**.

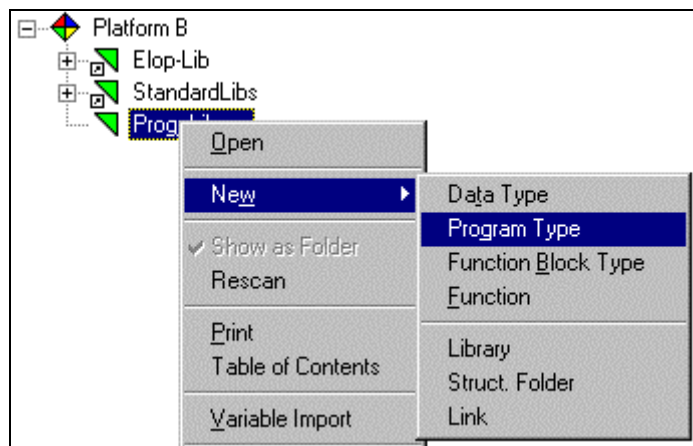



Рис. 44 Создание типа программы

- ☐ Щелкните на "+" перед библиотекой.

Новый функциональный блок **тип программы**  добавлен к структуре.

Указание: тип программы содержит логику, которая выполняется контроллером, кроме того, тип программы может содержать и другие функциональные блоки.

- ☐ Повторите шаг 6 и создайте **тип функционального блока**.
- ☐ Переименуйте тип программы и тип функционального блока, как описано в шаге 5.

Вид структуры проекта после этих шести шагов:

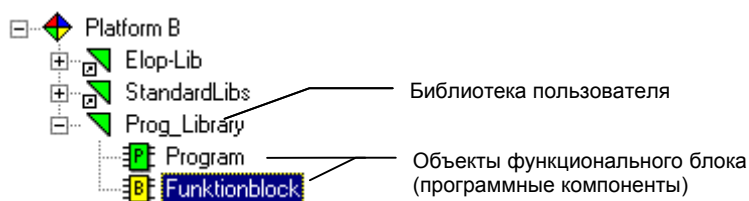



Рис. 45 Структура проекта после этого упражнения

7.2 Создание функционального блока

- ☐ Откройте функциональный блок  из Рис. 45 с помощью двойного щелчка по его символу:

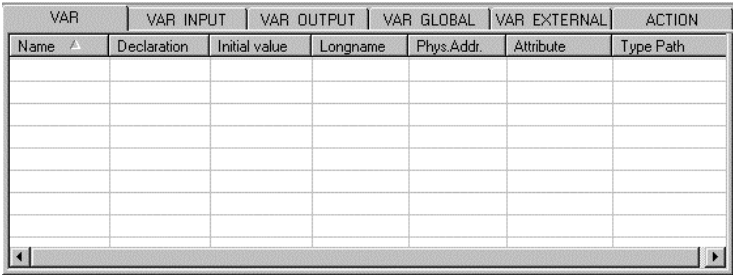
Открывается редактор FBD.

7.2.1 Описание переменных

Шаг 1:

Выберите в редакторе описаний переменных тип переменной:

- ☐ Щелкните на закладку с необходимым типом переменной.
Выберите VAR_INPUT (см. Рис. 46).



VAR						
VAR INPUT VAR OUTPUT VAR GLOBAL VAR EXTERNAL ACTION						
Name ▲	Declaration	Initial value	Longname	Phys.Addr.	Attribute	Type Path

Рис. 46 Редактор описаний переменных

Указание: для функций или функциональных блоков переменные VAR_INPUT и VAR_OUTPUT используются как входные и выходные переменные для вышестоящего программного компонента.

Шаг 2:

Введите переменную:

- ☐ Откройте диалог описания переменной двойным щелчком на свободное место в редакторе описаний переменных.
- ☐ В описании переменной замените стоящее по умолчанию значение "I1" в поле **Name (Имя)** на "Variable1" (Переменная1).
- ☐ В поле **Declaration (Описание)** выберите тип переменной "BOOL" (Булева).
- ☐ Определите **Position (Позиция)** переменной ("2" на Рис. 47) в модуле.
В примере на Рис. 47 "Variable1" определена как VAR_Input с левой стороны во второй позиции.

Рис. 47 Описание переменной

- В поле **Long name (Подробное имя)** введите ясное и понятное описание для "Variable1". Подробное имя поможет прояснить использование данной переменной.
- Добавьте введенные Вами данные в список переменных, нажав кнопку **Add (Добавить)**.

Диалоговое окно остается открытым, пока Вы его не **закроете**.

Указание: переменные VAR_INPUT и VAR_OUTPUT появляются в позиции, предписанной декларацией интерфейса.

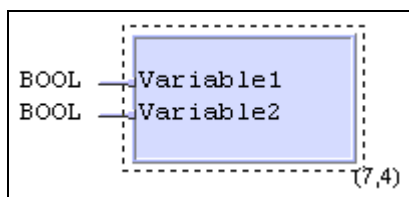


Рис. 48 Декларация интерфейса

7.2.2 Схема размещения функционального блока

Шаг 1:

Определите свойства функционального блока, используя декларацию интерфейса:

- ☐ Откройте контекстное меню функционального блока и выберите **Properties (Свойства)**.

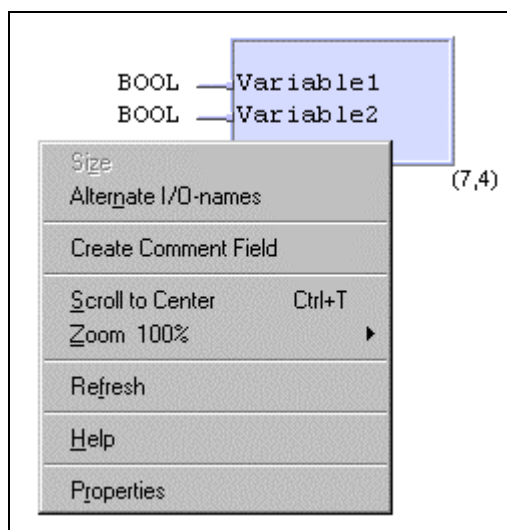


Рис. 49 Контекстное меню программного компонента

Шаг 2:

Введите имя программного компонента:

- В поле **Module text (Текст модуля)** введите имя данного программного компонента. При необходимости можно выбрать другой шрифт при помощи кнопки **Edit font...** (**Выбрать шрифт**).

Текст модуля появляется в центре программного компонента (см. Рис. 52).

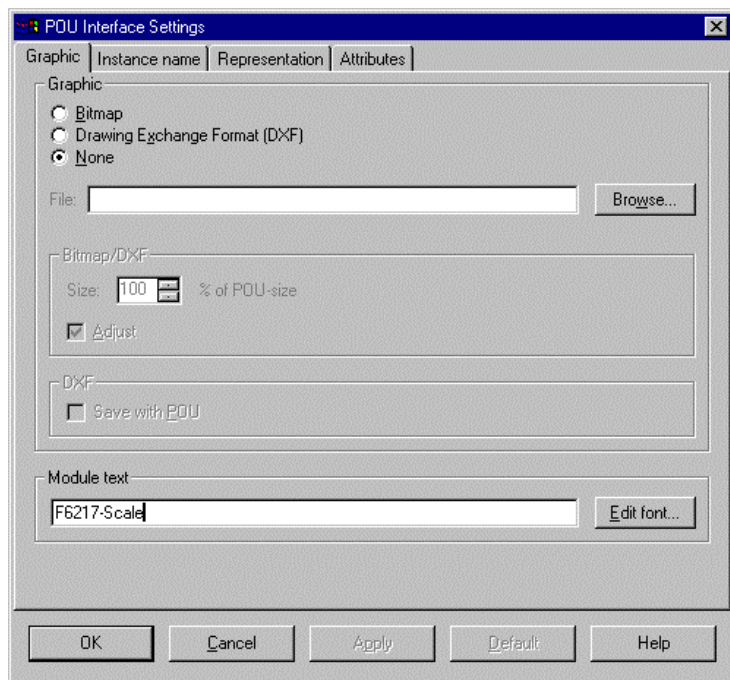


Рис. 50 Ввод имени программного компонента

Указание: текст модуля используется для описания функциональности программного компонента, например, "2-out-of-3".

Шаг 3:

Определите имя экземпляра:

- ☐ Выберите закладку **Instance name (Имя экземпляра)**.
- ☐ Введите в поле **Instance name (Имя экземпляра)** какое-либо имя и поставьте флаг в поле **Display (Просмотр)**.
- ☐ При необходимости можно выбрать другой шрифт и откорректировать выравнивание (поле **Alignment**).

Имя экземпляра появляется в верхнем левом углу программного компонента (см. Рис. 52).

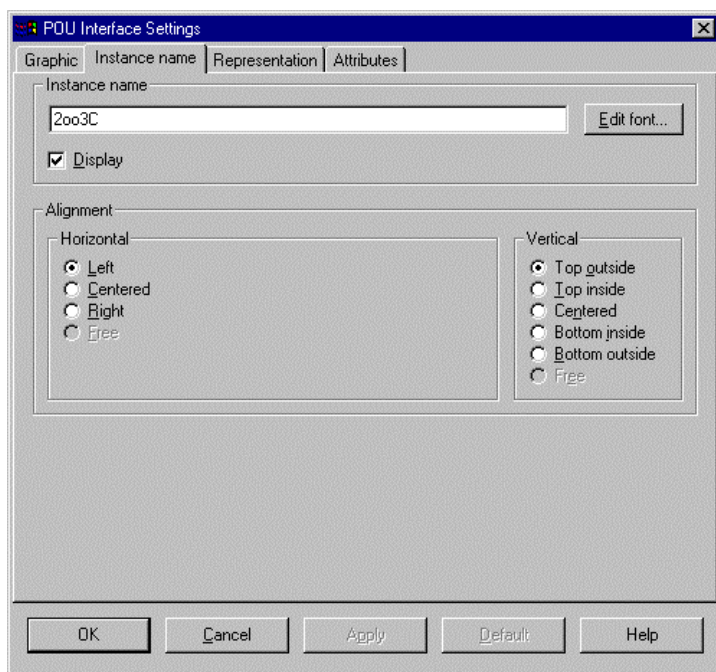


Рис. 51 Ввод и форматирование имени экземпляра

Указание: имя экземпляра служит для описания использования программного компонента в проекте, например, "2-out-of-3 temperature monitoring".

Шаг 4:

Измените размер программного компонента:

- Поместите курсор в правом нижнем углу программного компонента, нажмите левую клавишу мыши и измените размер программного компонента в соответствии с Вашими требованиями.

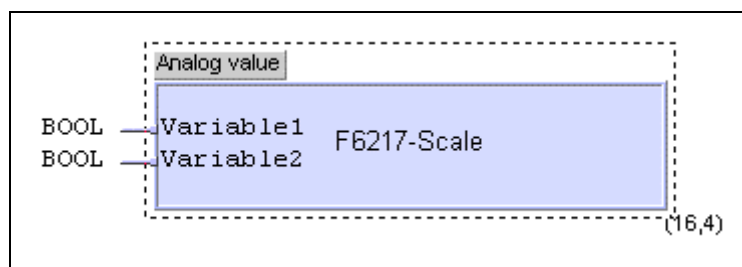


Рис. 52 Изменение размера программного компонента

7.2.3 Создание логики в область диаграммы

Шаг 1:

Используйте перенос для перемещения логических элементов из библиотек в область диаграммы:

- ☐ Щелкните в окне структуры на символ "+", чтобы открыть библиотеку **StandardLibs (Стандартные библиотеки)**.
- ☐ В пункте **StandardLibs** откройте библиотеку **IEC1131-3**, а там – **Functions, Bitstr (Функции, Битовая цепочка)**.
- ☐ Щелкните на функции **AND (И)** и перенесите этот объект из окна структуры в область диаграммы. Во время переноса включается предварительный просмотр объекта.
- ☐ После того, как Вы отпустили клавишу мыши объект становится на позицию курсора.

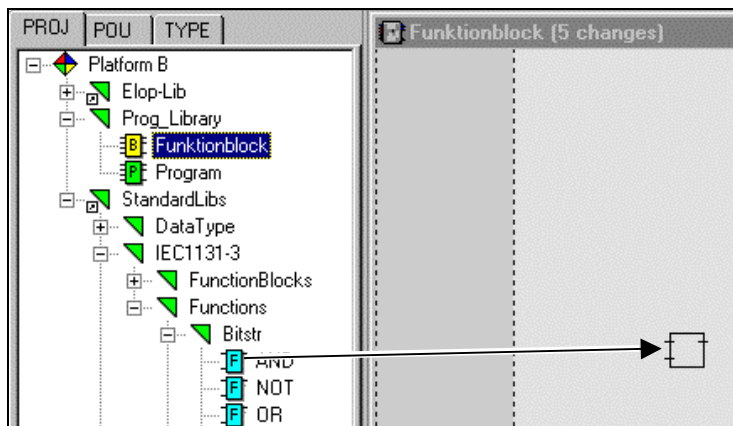


Рис. 53 Перенос программного компонента

Указание: процедура добавления стандартных объектов из библиотек в логику используется и для пользовательских объектов.

Шаг 2:

Дополните данные страницы:

Так как размещение объекта **AND (И)** является первым изменением содержимого этой страницы (см. также главу 5.4), автоматически открывается диалоговое окно "Edit Page Data" (Редактирование данных страницы).

В поле **Long name (Подробное имя)** введите ясное и понятное описание страницы.

Dialog box titled "Edit Page Data" with the following fields and controls:

- Page data**
 - Page no. **A.0**
 - Short name:
 - Long name:
 - created by (logon-ID): **SN2520** Date: **07/12/00**
 - tested by: Date:
- Labeling systems**
 - ULS:
 - PPLS:
 - ☒ Inherit labeling systems from page options
 - Page Id.: ☒ Inherit
- Buttons: OK, Cancel, Help

Рис. 54 Редактирование данных страницы

Шаг 3:

Дополните логику с помощью дополнительных модулей:

- ☐ Вставьте дополнительные объекты из библиотек в область диаграммы, как описано в [шаге 1](#).
- ☐ Создайте идентичные объекты, удерживая нажатой клавишу **Ctrl** во время переноса уже существующего объекта в другую позицию области диаграммы.

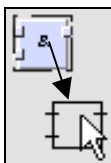


Рис. 55 Копирование объекта

Указание: объекты не могут находиться друг на друге. В этом случае ELOP II прерывает копирование и следует акустический сигнал.

Шаг 4:

Включите сетку и выполните ее масштабирование:

- ☐ Включите сетку раstra крайней правой кнопкой панели инструментов.
- ☐ Выполните масштабирование для той части экрана, которую Вы хотите увеличить.



Рис. 56 Кнопки масштабирования и сетки раstra

Указание: кнопки справа относятся к области диаграммы открытого объекта.

Шаг 5:

Добавьте переменные в Вашу логику:

Указание: переменные перемещаются в область диаграммы с помощью переноса.

- ❑ Щелкните в списке переменных на имя переменной и перенесите эту переменную в область поля значения (светло-серые поля с левой и правой стороны страницы) или в область диаграммы. Во время переноса включается предварительный просмотр поля значения.

После установки переменной в нужную позицию в поле значения появляется имя переменной.

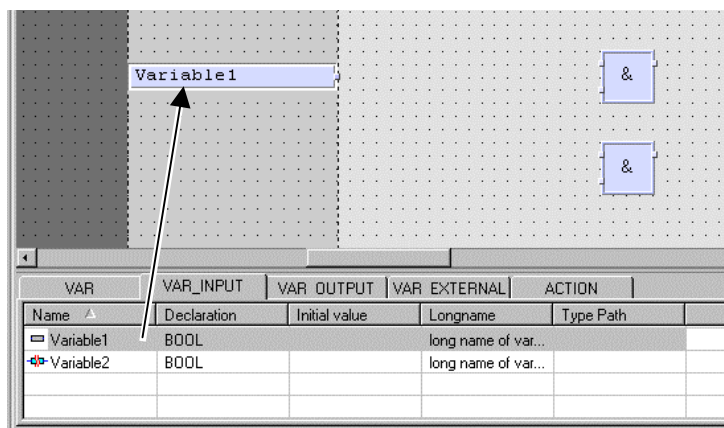


Рис. 57 Перенос переменных

Указание: невозможно установить переменные поверх других объектов в области диаграммы.

Шаг 6:

Проведите соединительные линии между переменными (поля значений) и функциональными блоками.

- ☐ Установите курсор на узел, с которого вы хотите начать линию (выход переменной).
- ☐ Нажмите левую клавишу мыши и проведите линию к тому узлу, с которым Вы хотите соединить переменную. Отпустите клавишу мыши.

В результате появилась соединительная линия между двумя узлами.

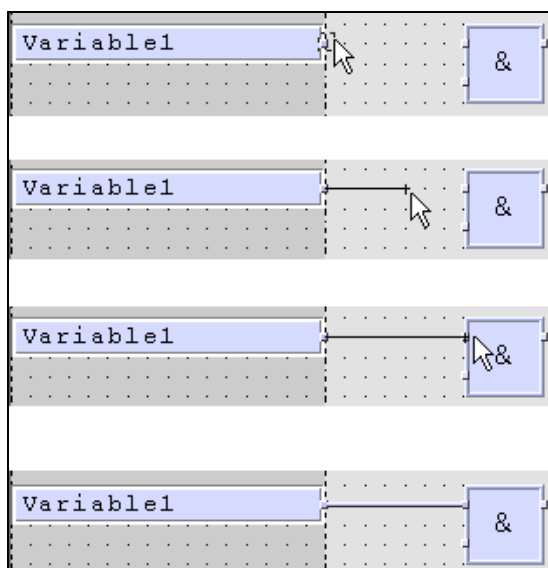



Рис. 58 Проведение соединительных линий

Указание: цвет линии зависит от типа данных (BOOL, Integer, Real (булево значение, целое число, вещественное число) и т.д.). Невозможно соединить узлы с различными типами данных!

7.3 Создание программы

- ☐ Двойным щелчком на пиктограмму  запускается редактирование программы.

Открывается редактор FBD.

7.3.1 Описание переменных

Процедура описания переменных идентична процедуре описания функциональных блоков (см. главу 7.2.1).

Указание: VAR и VAR_GLOBAL используются для программного компонента типа программы.

7.3.2 Создание логики в область диаграммы

Процедура создания логики типа программы идентична процедуре создания логики функционального блока. См. главу 7.2.3.

7.4 Создание конфигурации и ресурса

Шаг 1:

Создайте конфигурацию. В эту конфигурацию позже будут входить один или несколько ресурсов:

- ☐ Щелкните на проект в окне структуры.
- ☐ Откройте контекстное меню.
- ☐ Выберите **New, Configuration (Новая, Конфигурация)**.

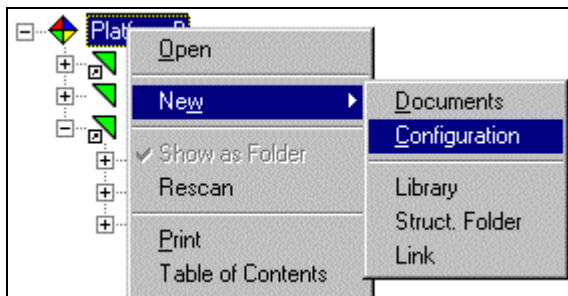


Рис. 59 Создание новой конфигурации

Шаг 2:

Переименуйте новую конфигурацию:

- ☐ Дважды с паузой щелкните в окне структуры клавишей мыши на имя конфигурации.

Открывается поле ввода, теперь имя можно изменить.

Шаг 3:

Создайте новый ресурс в конфигурации:

- ☐ Щелкните на конфигурацию в окне структуры.
- ☐ Откройте контекстное меню.
- ☐ Выберите **New, Resource (Новый, Ресурс)**.

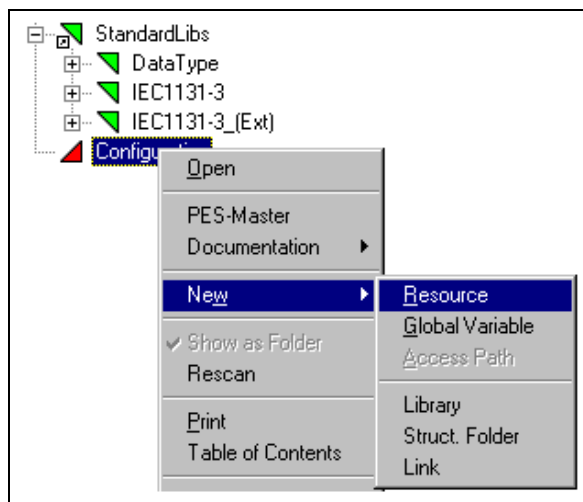


Рис. 60 Создание ресурса

Указание: если Вы переименовываете ресурс, помните, что имя ресурса не может быть длиннее восьми символов!

Шаг 4:

Присвойте ресурсу программу:

- ☐ Щелкните на ресурс в окне структуры.
- ☐ Откройте контекстное меню.
- ☐ Выберите **New, Program instance...** (Новая, Экземпляр программы).

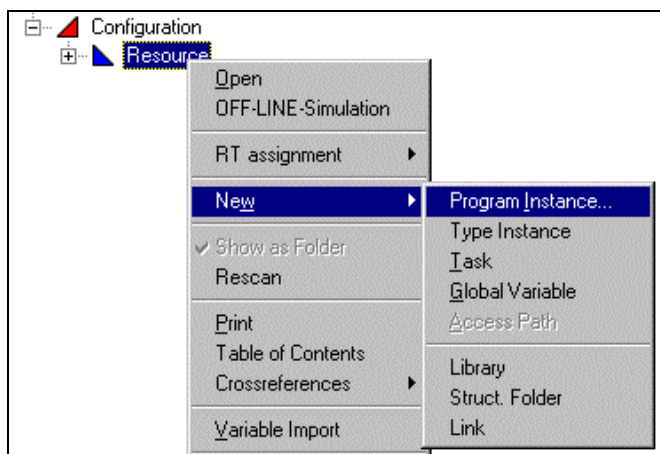


Рис. 61 Создание нового экземпляра программы

Шаг 5:

Выберите тип программы, которая должна выполняться в ресурсе:

- ☐ Выберите Вашу программу.

В ресурсе создается *экземпляр* программы, который является ссылкой на *тип* программы в Prog_Library.

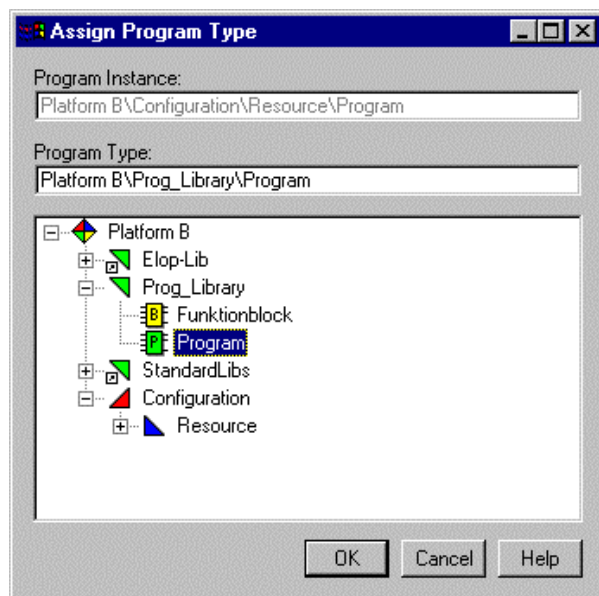


Рис. 62 Присвоение типа программы

Вид структуры проекта после этих пяти шагов:

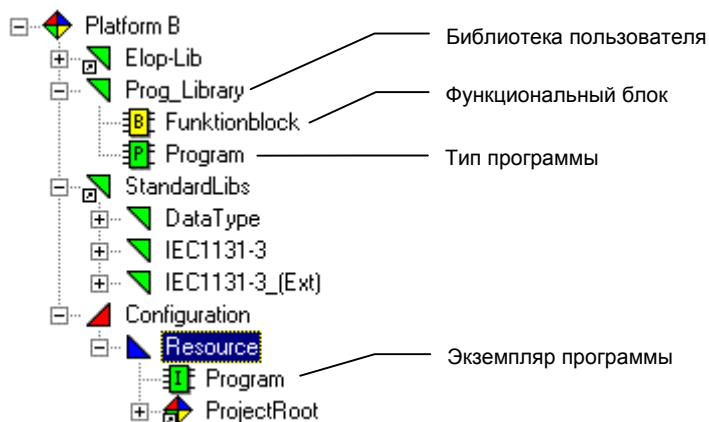


Рис. 63 Структура проекта

Указание: экземпляр программы в ресурсе – это только ссылка на тот тип программы из Prog_Library, который выполняется в контроллере. Тип программы в библиотеке содержит только логику и переменные. В экземпляре программы следует выполнить настройку в зависимости от аппаратного обеспечения (типа ресурса), например, адреса В/В или адреса шины.

7.5 Автономное моделирование логики

Шаг 1: Создайте задачу для моделирования на компьютере:

- ☐ Щелкните на ресурс в окне структуры.
- ☐ Откройте контекстное меню ресурса.
- ☐ Выберите **New, Task (Новая, Задача)**.

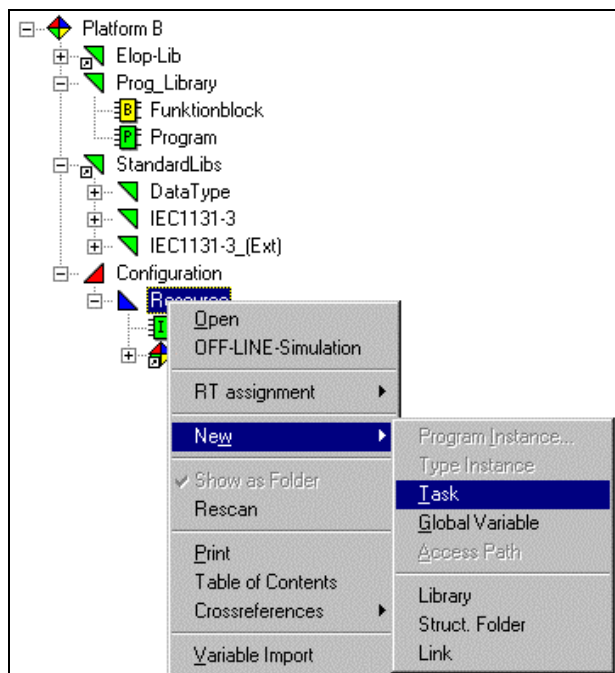


Рис. 64 Создание новой задачи

Шаг 2:

Присвойте задачу экземпляру программы:

- ☐ Щелкните на экземпляре программы в окне структуры.
- ☐ Откройте контекстное меню экземпляра программы.
- ☐ Выберите **Properties (Свойства)**.

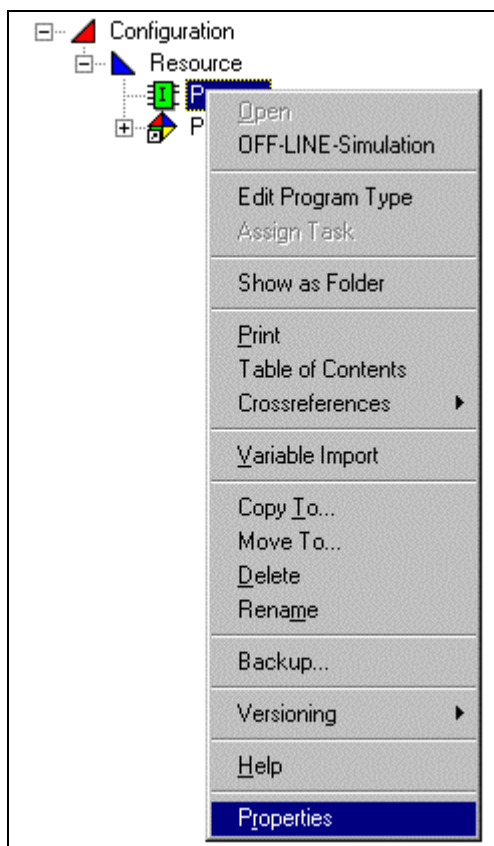


Рис. 65 Вызов меню "Свойства" экземпляра программы

- В окне "Properties: Program" (Свойства: программа) щелкните на кнопку **Browse (Обзор)** в части окна **Task Assignment (Присвоение задачи)**.

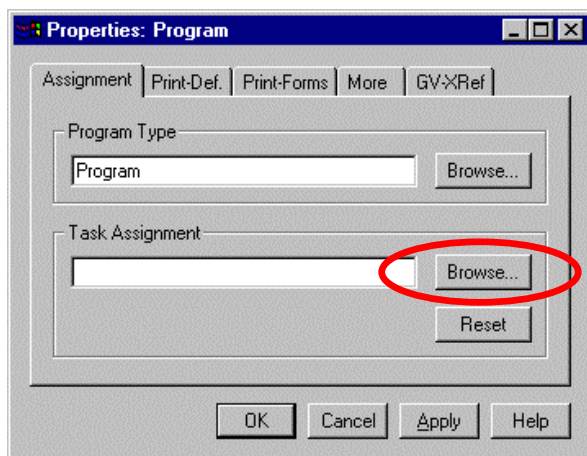


Рис. 66 Присвоение задачи

- Выберите задачу, созданную в [шаге 1](#).

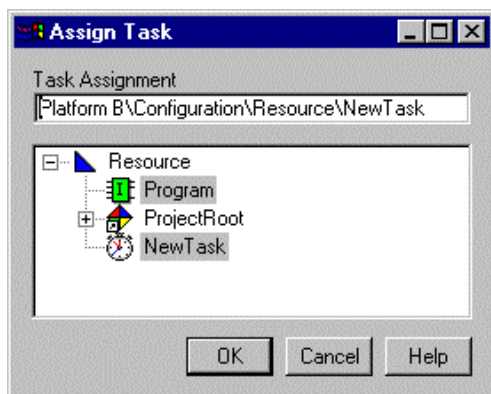


Рис. 67 Выбор задачи

Указание: данная задача управляет смоделированным выполнением экземпляра программы на компьютере. Она не влияет на экземпляры программы в PLC (программируемой электронной системе).

Шаг 3:

Откройте автономное моделирование:

- ☐ Щелкните на ресурс.
- ☐ Откройте контекстное меню ресурса.
- ☐ Выберите **OFF-LINE-Simulation**
(Автономное моделирование).

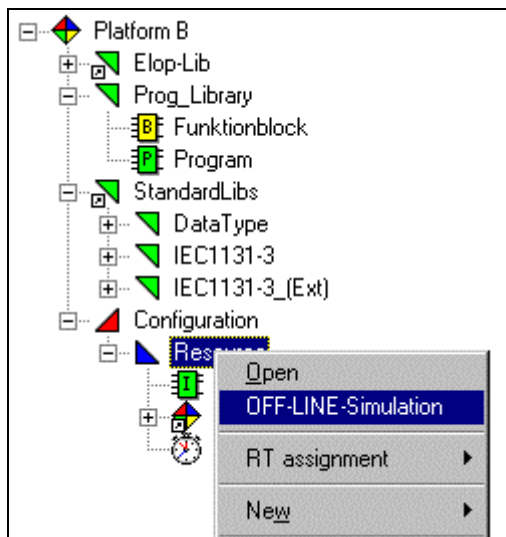


Рис. 68 Открытие автономного моделирования

Указание: в окне структуры открывается новая закладка OLS (Offline Simulation – автономное моделирование).

Шаг 4:

Запустите автономное моделирование:

- Щелкните на кнопку **Cold boot (Холодная загрузка)** (с голубой стрелкой).

После запуска статус изменяется из "Stopped" (Остановлено) к "Running" (Работает).

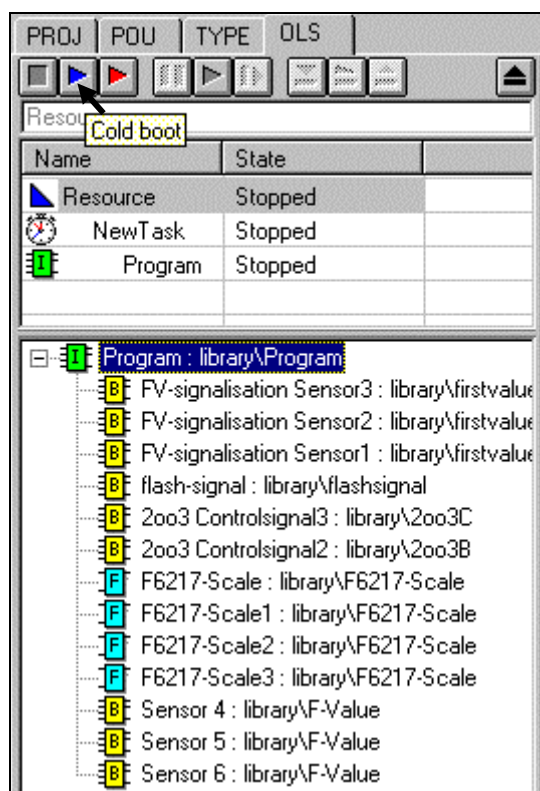


Рис. 69 Запуск автономного моделирования

Шаг 5:

Откройте функциональный блок в автономном моделировании:

- ☐ Откройте функциональный блок двойным щелчком по нему в окне структуры.

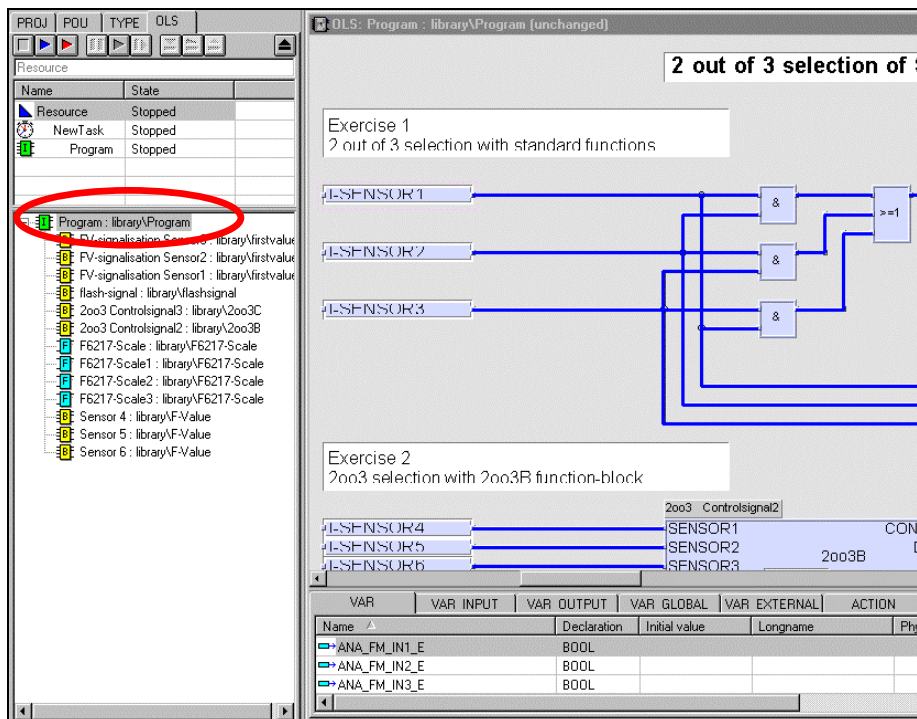


Рис. 70 Автономное моделирование программы

Указание: Вы можете открыть несколько функциональных блоков в одно и тоже время и проследить прохождение сигнала между ними.
Кроме того, используя двойной щелчок, функциональные блоки можно открывать непосредственно в логике.

Шаг 6:

Измените статус сигнала и протестируйте логику:

Изменение статуса сигнала с помощью поля онлайнного теста (поля OLT):

- ☐ Щелкните на поле значения и удерживайте кнопку мыши нажатой.
- ☐ Переместите курсор из поля значения и отпустите поле OLT, следующее за курсором, на свободном месте.
- ☐ При необходимости переместите поле OLT.
- ☐ Измените статус сигнала, дважды щелкнув по полю OLT.

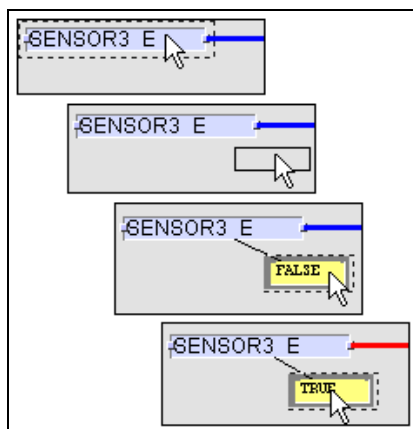


Рис. 71 Создание поля OLT

Указание: после того, как Вы вставили поля OLT при закрытии функционального блока в области диаграммы появляется вопрос "Save change?" (Сохранить изменения?).

Если Вы нажимаете **Yes (Да)**, поля OLT сохраняются. Если Вы нажимаете **No (Нет)**, созданные Вами поля OLT не будут сохранены.

Изменение статуса сигнала непосредственно в поле значения:

- ☐ Установите курсор на поле, значение которого Вы хотите изменить.
- ☐ Нажмите и удерживайте клавишу ALT. Вместо имени сигнала в поле значения отображается статус сигнала.
- ☐ Измените статус сигнала, щелкнув по полю значения.
- ☐ Отпустите клавишу ALT. Снова отображается имя сигнала.

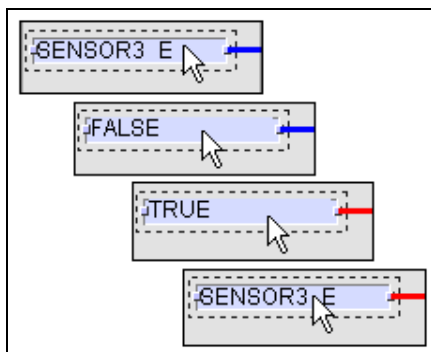


Рис. 72 Изменение поля значения с помощью клавиши ALT

Указание: Вы можете изменить только те значения, которые не созданы логикой.

Шаг 7:

Выйдите из автономного моделирования:

- ☐ Щелкните на кнопку **Close OLS (Закрывать автономное моделирование)** на панели инструментов.

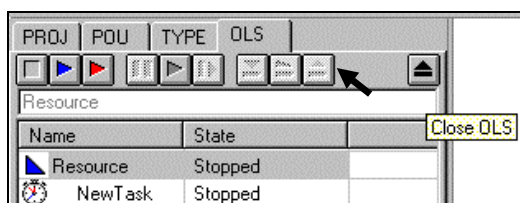


Рис. 73 Выход из автономного моделирования

7.6 Схема коммутирующего шкафа PLC и присвоение переменных

Шаг 1: Присвойте тип ресурса (RT):

- ☐ Щелкните на ресурс в окне структуры.
- ☐ Откройте контекстное меню ресурса.
- ☐ Щелкните на **RT assignment (Присвоение типа ресурса)**, и выберите необходимый тип **PLC** (программируемой электронной системы) **HIMA**.

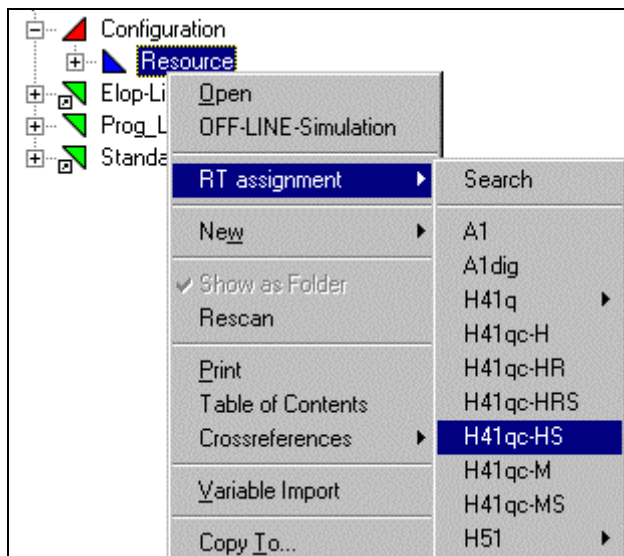


Рис. 74 Присвоение типа ресурса

Шаг 2:

Откройте схему коммутирующего шкафа:

- ☐ Откройте контекстное меню ресурса.
- ☐ Выберите **Edit cabinet layout (Редактировать схему коммутирующего шкафа)**.

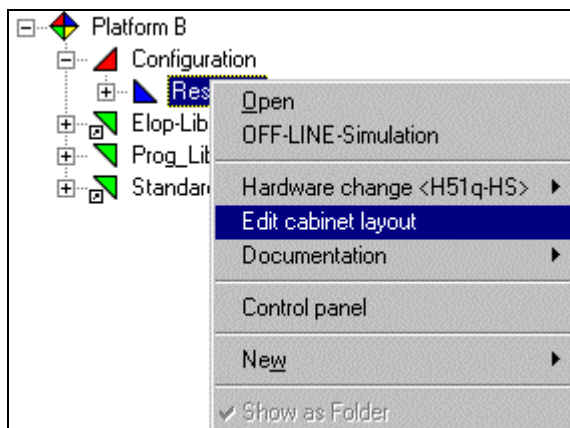


Рис. 75 Редактирование схемы коммутирующего шкафа

В окне "Edit cabinet layout" (Редактирование схемы коммутирующего шкафа) отображается незаполненный коммутирующий шкаф выбранной программируемой электронной системы (PLC) HIMA (см. Рис. 76).

Шаг 3:

Определите каркасы Вашего коммутирующего шкафа (в нашем случае H51):

- ☐ Выберите закладку **Racks (Каркасы)** внизу окна.
- ☐ Перенесите каркас из списка в коммутирующий шкаф и расположите ее в требуемом месте.

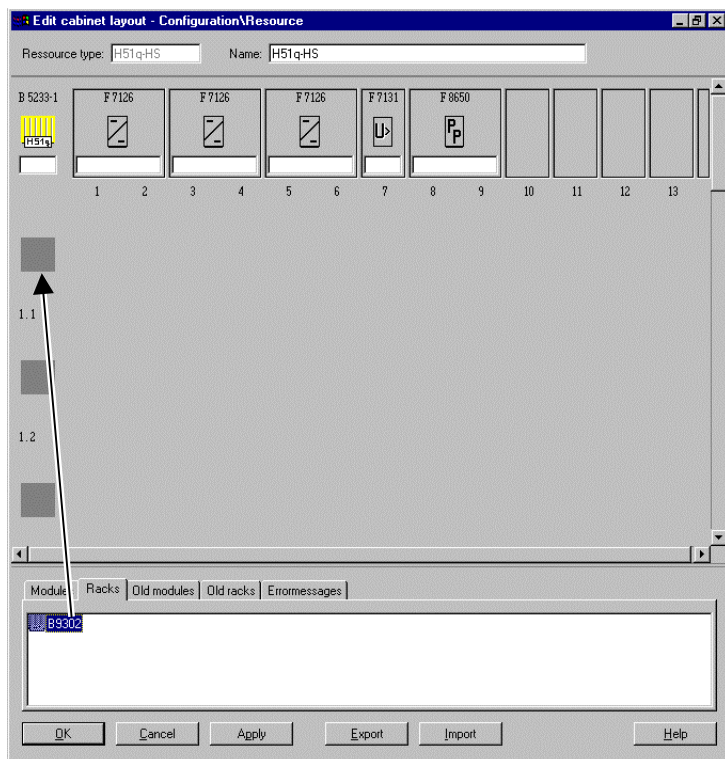


Рис. 76 Распределение каркасов

Указание: позиция каркаса должна совпадать с адресом каркаса, установленном в аппаратном обеспечении.

Шаг 5:

Откройте присвоение модулей V/V:

- Чтобы открыть окно "Edit Tag name" (Редактирование имени признака), дважды щелкните по пиктограмме модуля (в нашем случае F 3236).

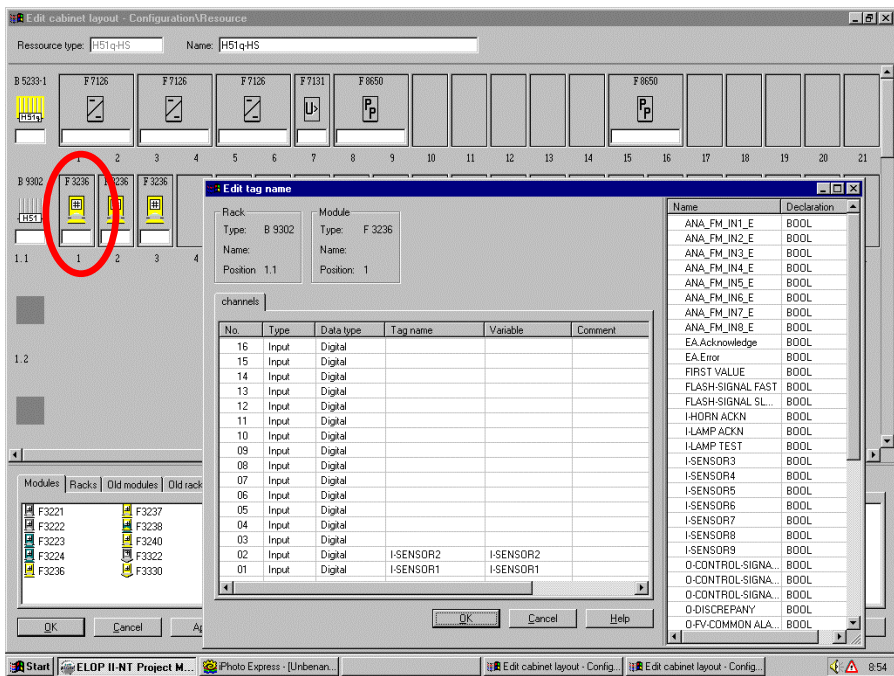


Рис. 78 Присвоение модулей

Шаг 6:

Присвойте переменные к входам и выходам модуля В/В:

Указание: на Рис. 79, слева, показаны 16 цифровых входных каналов модуля F 3236. На Рис. 79, справа, перечислены все переменные, тип данных которых (BOOL – булев) соответствует каналам модуля F 3236, и которые еще не присвоены к какому-либо каналу В/В.

- ❑ Щелкните на имя в списке переменных, перенесите переменную в список каналов В/В и установите ее там на нужную Вам позицию.

Присвоенные переменные отображаются в списке каналов В/В у своих каналов и удаляются из списка переменных.

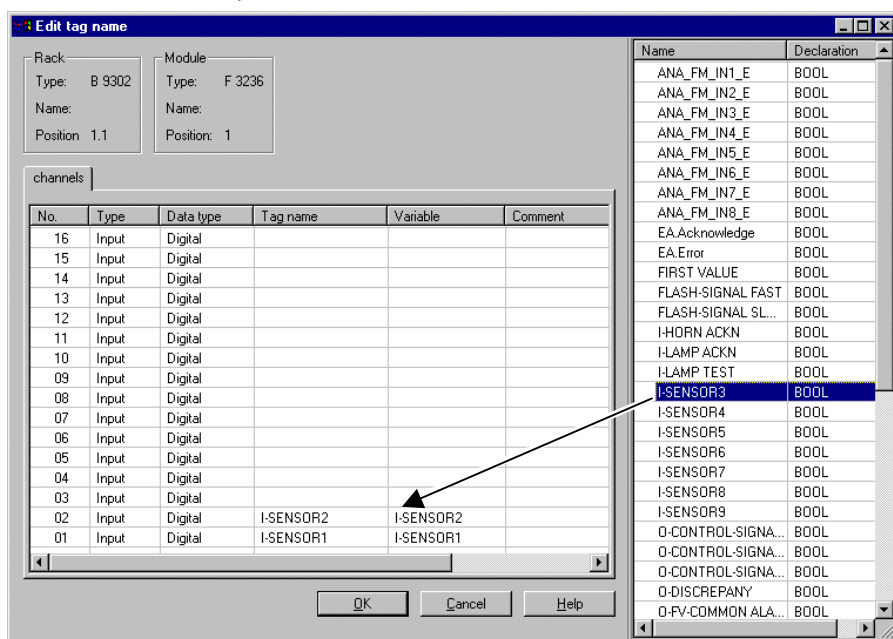


Рис. 79 Присвоение переменных каналам В/В

Указание: одна переменная может быть присвоена только одному каналу В/В.

Шаг 7:

Закройте окна, нажав **ОК**.

7.7 Запуск программы на PLC

7.7.1 Компиляция программы

Прежде чем созданная пользователем логика начнет работать на контроллере, она должна быть преобразована в код, который будет выполняться центральным процессором:

Шаг 1:

Запустите генератор кода.

- ☐ Щелкните на ресурс.
- ☐ Откройте контекстное меню ресурса.
- ☐ Выберите **Codegenerator (Генератор кода)**.

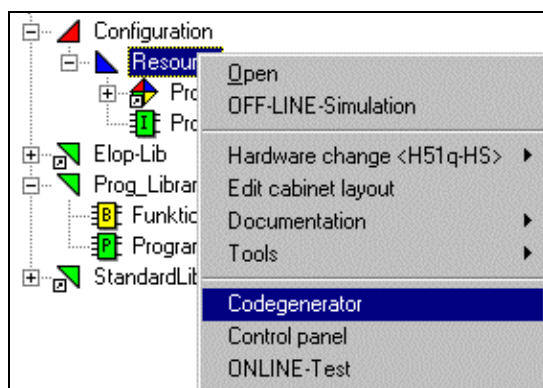


Рис. 80 Запуск генератора кода

Шаг 2:

Подтвердите запуск генератора кода:

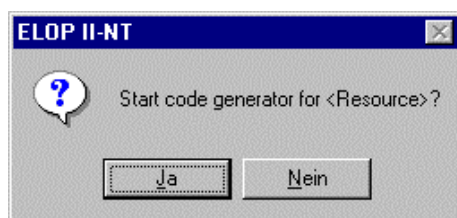


Рис. 81 Подтверждение запуска

Генератор кода сообщает о процессе компиляции в своем окне.

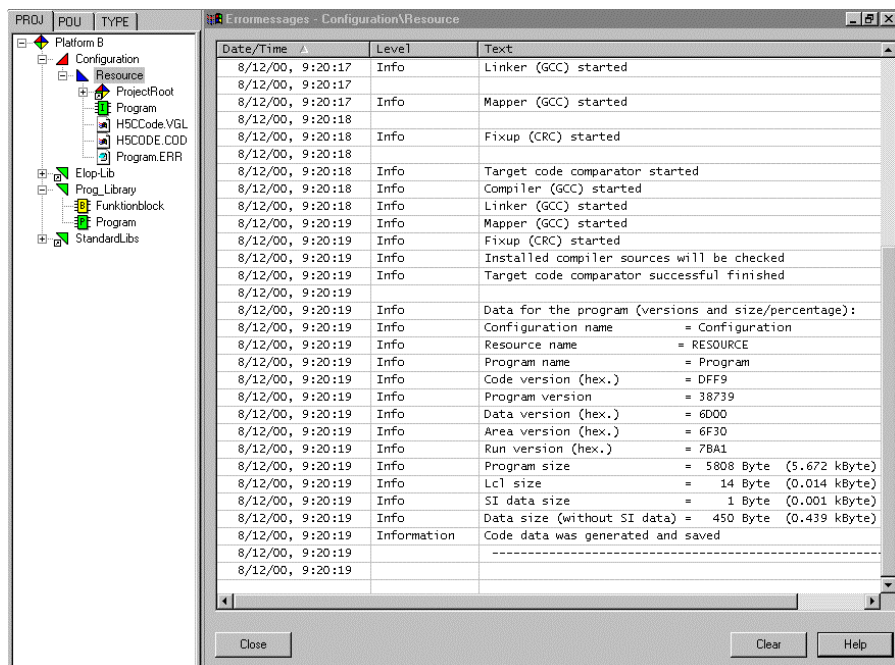



Рис. 82 Сообщения генератора кода

После успешного окончания работы генератора кода в окне структуры появляются следующие дополнительные элементы:

- NAME.VGL файл сравнения
- NAME.COD файл с преобразованным кодом
- NAME.ERR файл с сообщениями об ошибках

Указание: если во время генерации кода произошла ошибка, подробную информацию можно получить, открыв окно статуса и ошибок .

7.7.2 Коммуникация между компьютером и контроллером

Указание: для создания связи (коммуникации) между компьютером и контроллером необходимо определить коммуникационную шину и установить ее настройки.

Шаг 1:

Откройте свойства конфигурации:

- ☐ Щелкните на конфигурацию в окне структуры.
- ☐ Откройте контекстное меню конфигурации.
- ☐ Выберите **Properties (Свойства)**.

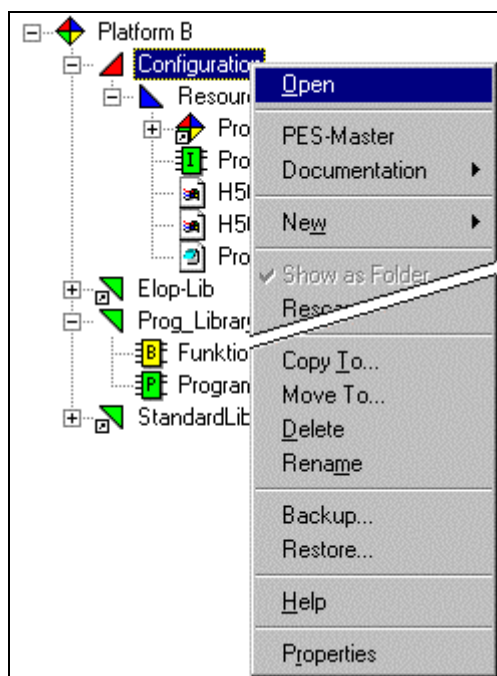


Рис. 83 Открытие свойств конфигурации

Шаг 2:

Добавьте новую шину:

- ☐ В окне "Properties: Configuration" (Свойства: конфигурация) выберите закладку **Buses (Шины)**.
- ☐ Щелкните на клавишу **Add (Добавить)**, чтобы определить новую шину.
- ☐ В диалоговом окне "Add communication system" (Добавить коммуникационную систему) введите имя новой шины в поле **Name (Имя)**.
- ☐ Убедитесь, что в поле **Type (Тип)** установлен тип шины *HIBUS*.
- ☐ Нажмите **ОК**, чтобы добавить введенные Вами данные.

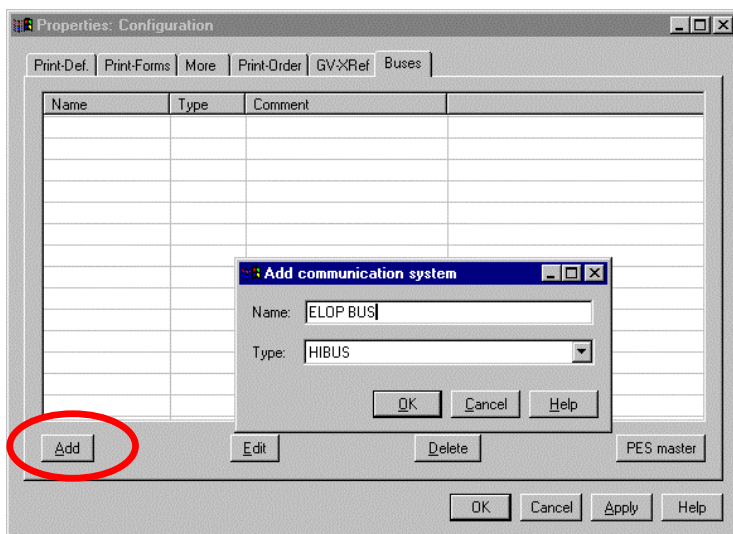


Рис. 84 Добавление шины

Шаг 3:

Добавьте ресурс к списку пользователей шины:

- ☐ В списке, показанном на Рис. 84, выделите только что созданную шину и щелкните на **Edit (Редактировать)**.
- ☐ Открылось окно "Edit HIBUS – Bus name" (Редактирование HIBUS – имя шины). Выберите закладку **Station (Станция)**.
- ☐ Щелкните на клавишу **Add (Добавить)**, чтобы определить участника шины.
- ☐ В окне "HIBUS station – Bus name" (Редактирование HIBUS – имя шины) (см. Рис. 85) выберите в выпадающем списке **Name (Имя)** ресурс (контроллер), который Вы хотите подключить к шине.

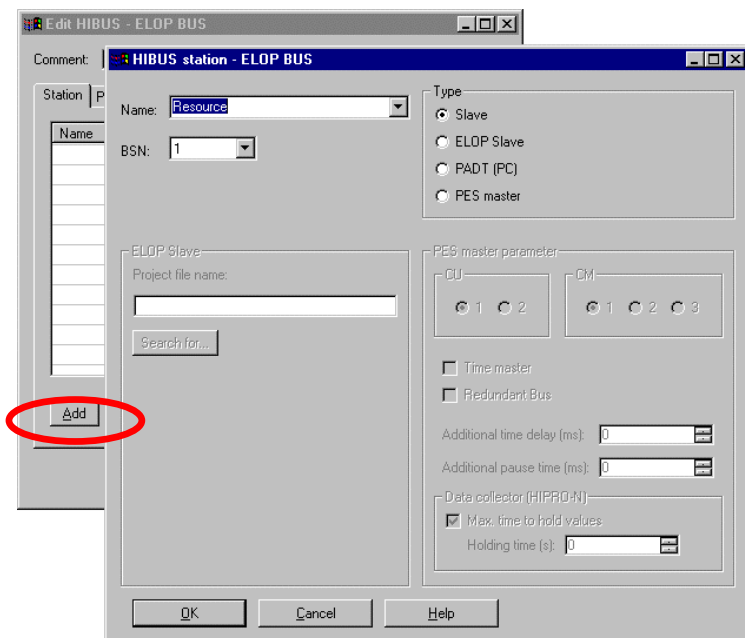


Рис. 85 Добавление ресурса к списку участников шины

- ☐ Определите **Bus Subscriber Number** (Абонентский номер шины – **BSN**) в соответствии с настройкой центрального модуля контроллера.
- ☐ В пункте **Типе (Тип)** выберите "Slave" (Подчиненный компонент) ресурса.

Шаг 4:

Добавьте компьютер к списку участников шины:

- ☐ В окне "Edit HIBUS – *Bus name*" (Редактирование HIBUS – *имя шины*) щелкните на кнопку **Add (Добавить)**, чтобы определить нового участника шины.
- ☐ В окне "HIBUS station – *Bus name*" (Редактирование HIBUS – *имя шины*) введите в поле **Name (Имя)** имя компьютера, который связан с ресурсом. Имя может быть произвольным.
- ☐ Определите **Bus Subscriber Number (Абонентский номер шины – BSN)** для компьютера. Обычно используется "31".
Важно: BSN компьютера не должен конфликтовать с BSN контроллера.
- ☐ В поле **Type (Тип)** выберите "PADT (PC)" для компьютера.

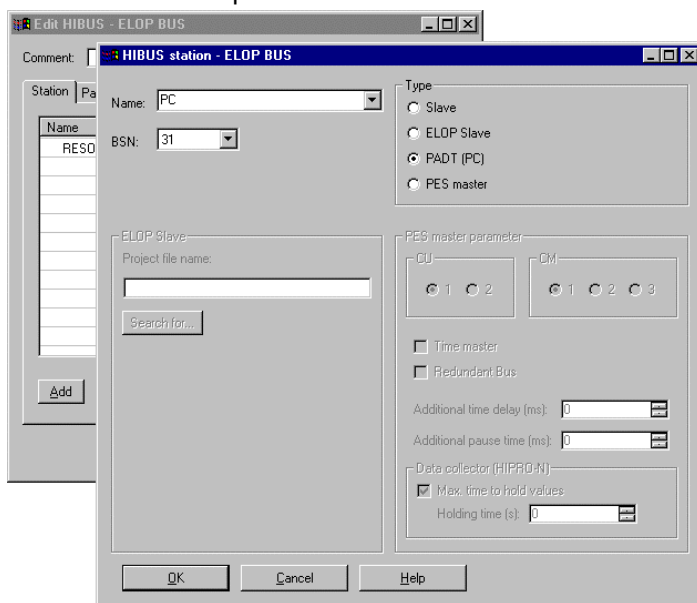


Рис. 86 Добавление компьютера к списку участников шины

Указание: в соответствии со стандартом МЭК 61131-3 компьютер с пользовательским программным обеспечением обычно называется "**Programming and Debugging Tool**" (Инструмент программирования и отладки) или сокращенно PADT.

Шаг 5:

Определите параметры шины:

- В окне "Edit HIBUS – Bus name" (Редактирование HIBUS – имя шины) выберите закладку **Parameter (Параметры)** для определения **Baud rate (Скорость передачи данных в бодах)**, количества **Stop bits (Стоповые биты)** и **Parity (Четность)** последовательной шины.

Указание: данные настройки должны соответствовать настройкам контроллера.

- Сохраните Ваши настройки шины, нажав **OK**.

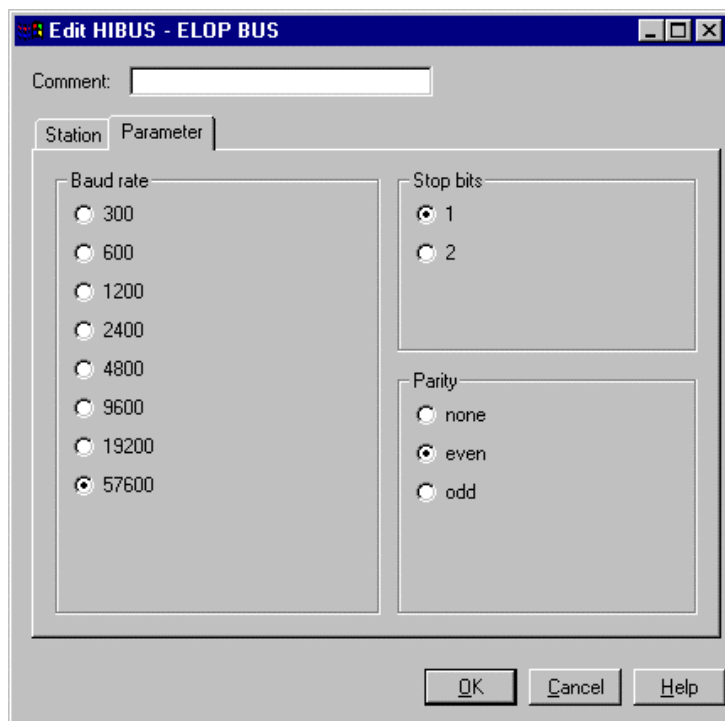


Рис. 87 Определение параметров шины

Указание: параметры, установленные для шины, используются компьютером для коммуникации. Скорость передачи данных порта контроллера может быть изменена переключением на центральном модуле.

Шаг 6:

Откройте свойства ресурса:

- ☐ Щелкните на ресурс в окне структуры.
- ☐ Откройте контекстное меню ресурса.
- ☐ Выберите **Properties (Свойства)**.

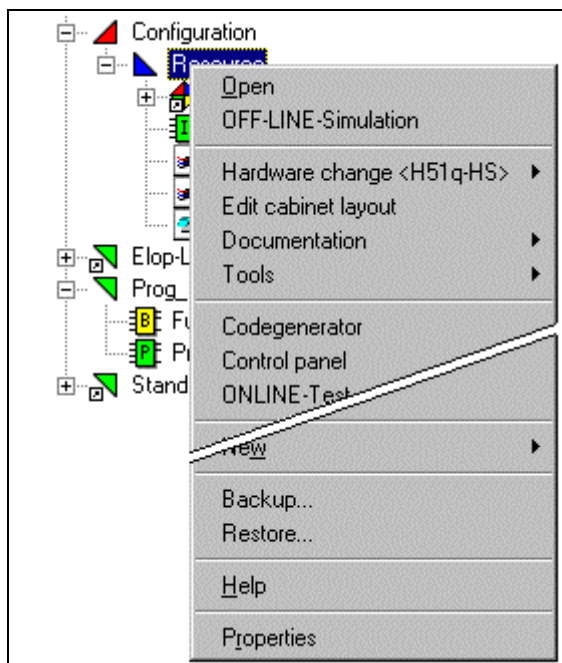


Рис. 88 Открытие свойств ресурса

Шаг 7:

Закладка PADT (PC)

- ☐ В окне "Properties: Resource" (Свойства: Ресурс) откройте закладку **PADT (PC)**.
- ☐ В поле **Bus (Шина)** выберите Вашу шину из списка сконфигурированных шин.
- ☐ В поле **PADT (PC)** выберите Ваш компьютер из списка сконфигурированных PADT шины.
- ☐ В поле **PC interface (Порт компьютера)** выберите последовательный порт, связанный с контроллером.

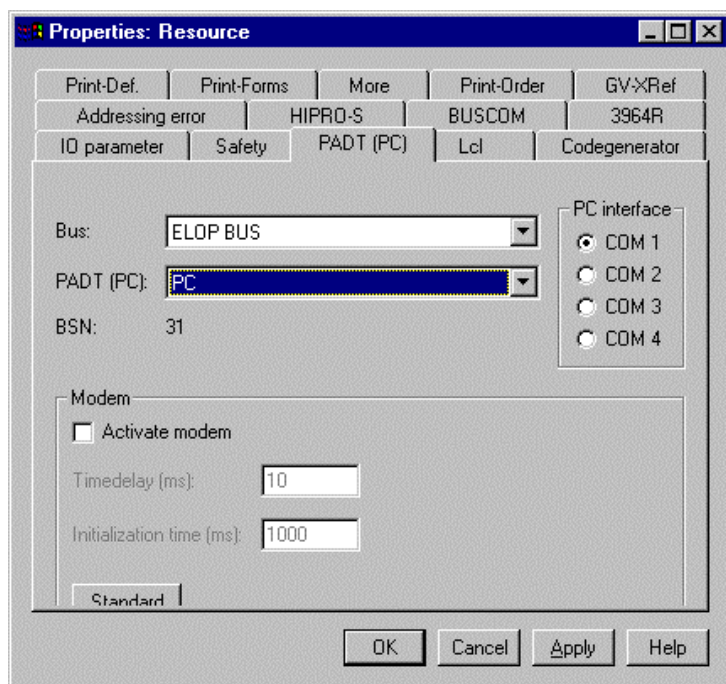


Рис. 89 Закладка PADT (PC) в свойствах ресурса

7.7.3 Загрузка и запуск

Если коммуникационная шина определена, можно загрузить контроллер и управлять им по шине.

Шаг 1: Откройте панель управления:

- ☐ Щелкните на ресурс.
- ☐ Откройте контекстное меню ресурса.
- ☐ Выберите **Control panel (Панель управления)**.

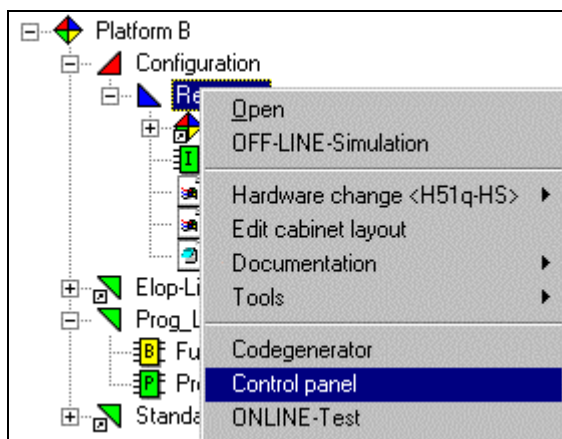


Рис. 90 Запуск панели управления

*Указание: Вы можете оставить панель управления открытой для других операций. Просто выберите закладку **CP** в окне структуры.*

Шаг 2:

Запустите процедуру загрузки:

- ❑ Щелкните в панели управления на кнопку **Download/Reload (Загрузка/Перезагрузка)**.
- ❑ В окне "Download/Reload" (Загрузка/Перезагрузка) щелкните на **ОК**, чтобы запустить загрузку.

Процесс загрузки отображается в окне "Data transfer" (Передача данных).

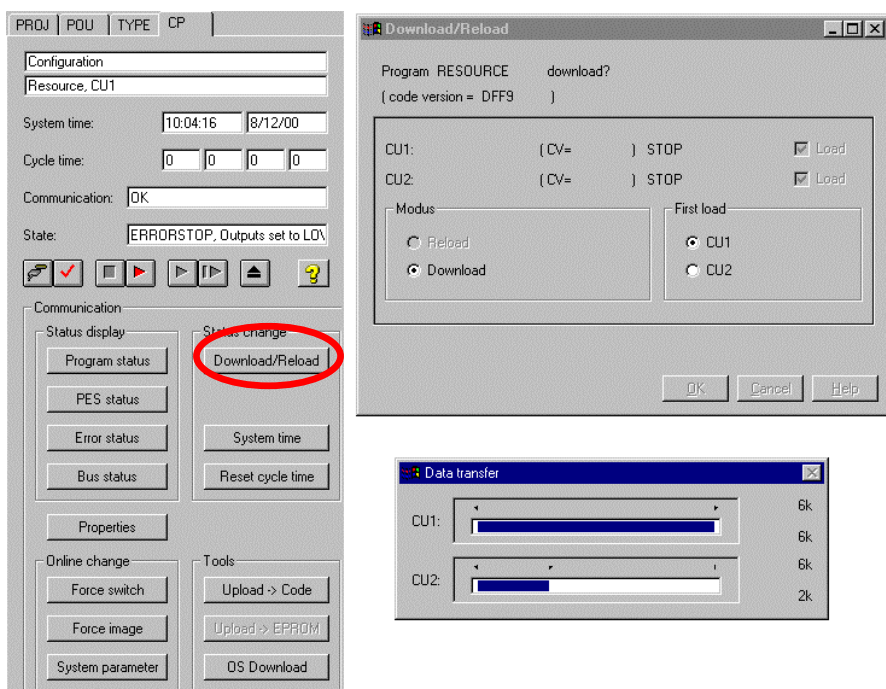


Рис. 91 Запуск загрузки

Указание: загрузка должна производиться только квалифицированными работниками и только с разрешения сотрудника, отвечающего за соответствующее оборудование, так как при загрузке происходит остановка оборудования.

Шаг 3:

Запустите контроллер:

- ☐ Щелкните в панели управления на кнопку **Start (Пуск)**.
- ☐ В случае нового проекта в качестве режима запуска выберите **Cold start (Холодный пуск)**.

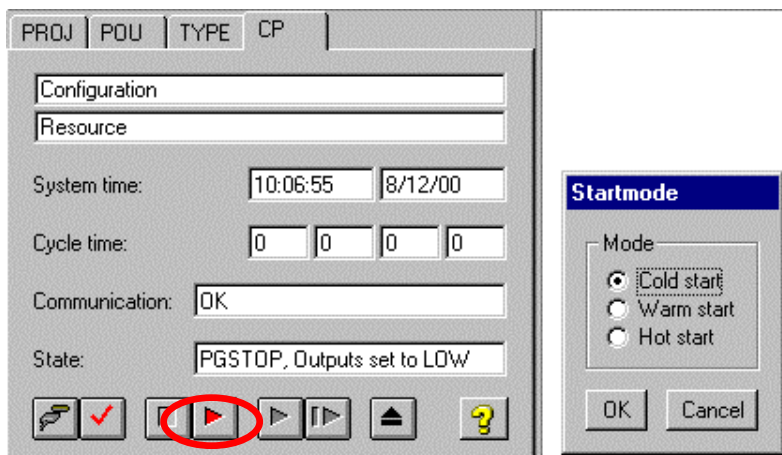


Рис. 92 Холодный пуск контроллера

- ☐ После нажатия кнопки **OK** появляется запрос подтверждения пуска контроллера. Изменение статуса контроллера (STOP -> RUN) (СТОП -> РАБОТА) отображается в отдельном окне.

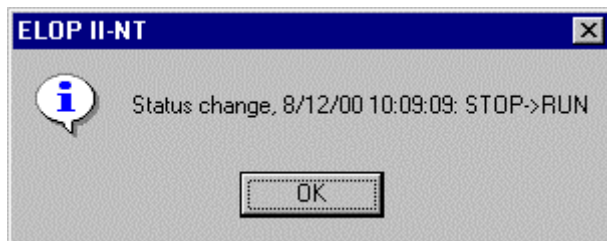


Рис. 93 Запуск контроллера

Статус контроллера виден на панели управления, при работающем контроллере пиктограмма кнопки пуска движется.



Рис. 94 Панель управления после запуска контроллера

7.8 Онлайн-тест

Указание: онлайн-тест можно запустить только в том случае, если версия кода приложения в контроллере идентична версии кода проекта в компьютере.

Шаг 1:

Откройте онлайн-тест:

- ☐ Щелкните на ресурс.
- ☐ Откройте контекстное меню ресурса.
- ☐ Выберите **ONLINE-Test (Онлайн-тест)**.

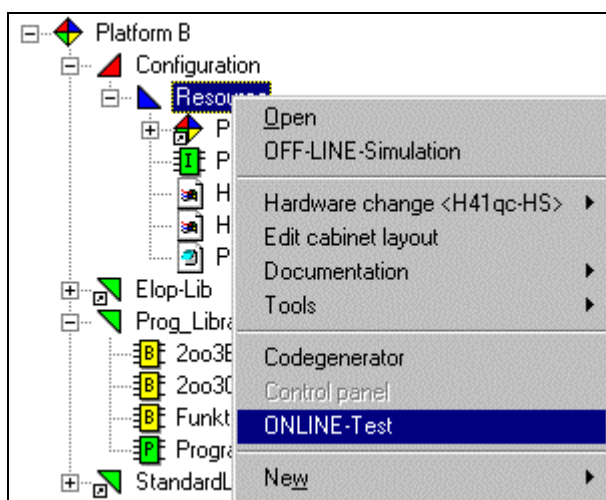


Рис. 95 Вызов онлайн-теста

В окне структуры открылась закладка **OLT (Online-Test)** (Онлайновый тест).

Под панелью с кнопками представлены все функциональные блоки проекта.

- ☐ Щелкните дважды в этом списке на тот функциональный блок, который Вы хотите протестировать.

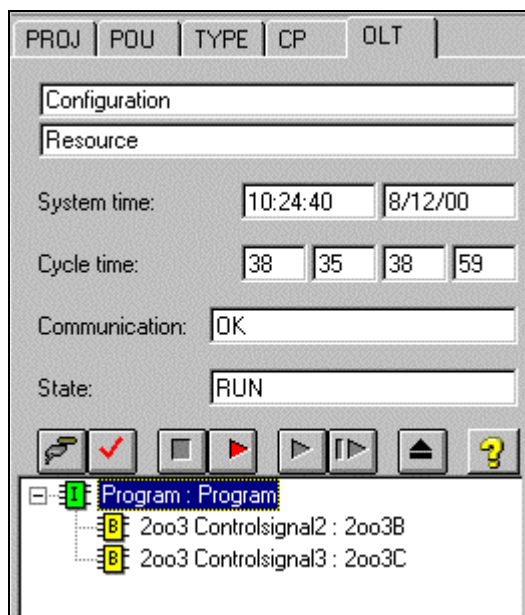


Рис. 96 Онлайновый тест в окне структуры

Шаг 2:

Откройте функциональные блоки в режиме OLT:

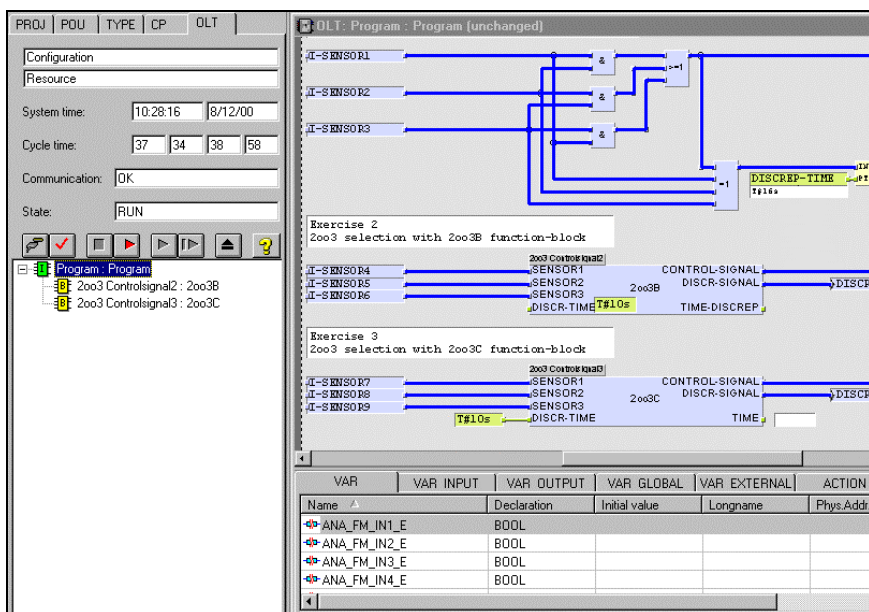


Рис. 97 Онлайнный тест программы

Шаг 3:

Закройте онлайнный тест:

- ☐ Щелкните на кнопку **Close (Заккрыть)**.



Рис. 98 Заккрытие онлайнного теста

Указание: прежде чем пытаться изменить логику программы следует закрыть онлайнный тест!

7.9 Инициирование входов и выходов

Шаг 1:

Откройте панель управления:

- ☐ Щелкните на ресурс.
- ☐ Откройте контекстное меню ресурса.
- ☐ Выберите **Control panel (Панель управления)**.

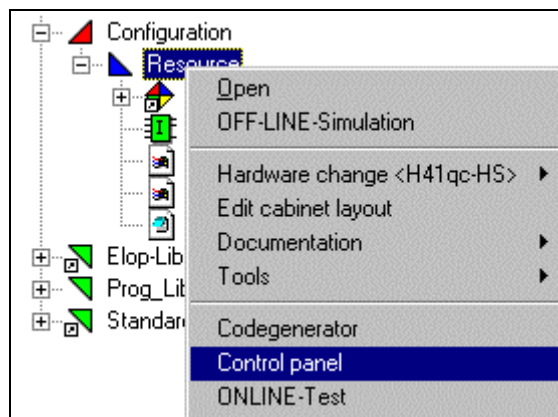


Рис. 99 Запуск панели управления

Шаг 2:

Проверьте настройки выключателя инициирования:

- Щелкните в панели управления на кнопку **Force switch** (**Выключатель инициирования**).

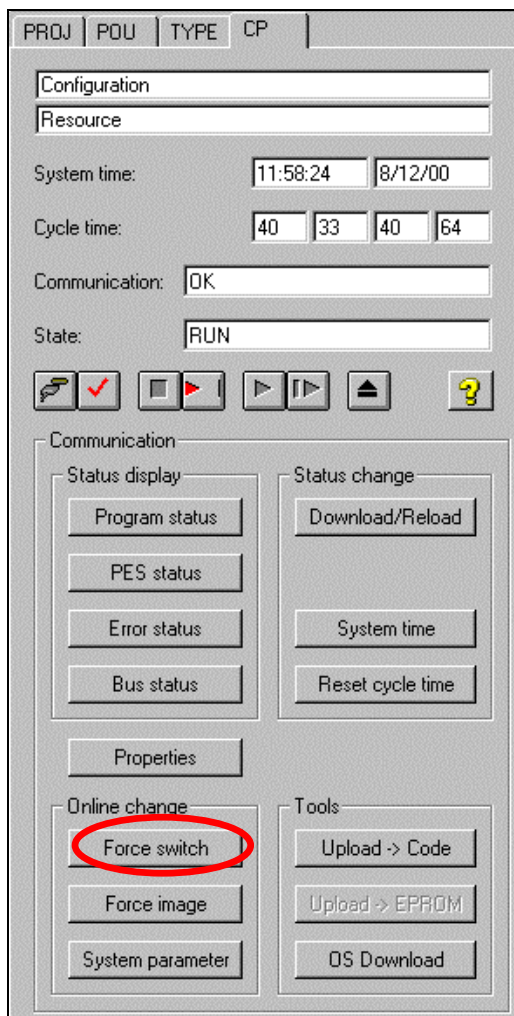


Рис. 100 Проверка настроек выключателя инициирования

В окне "Force switch" (Выключатель инициирования) отдельно для входов и выходов указан статус главного выключателя инициирования (**Master force**) и число активированных выключателей инициирования (**Activated force switches**).

- ☐ Установите флаг в поля **Set (Установить)** для входов и выходов (см. рис. ниже).

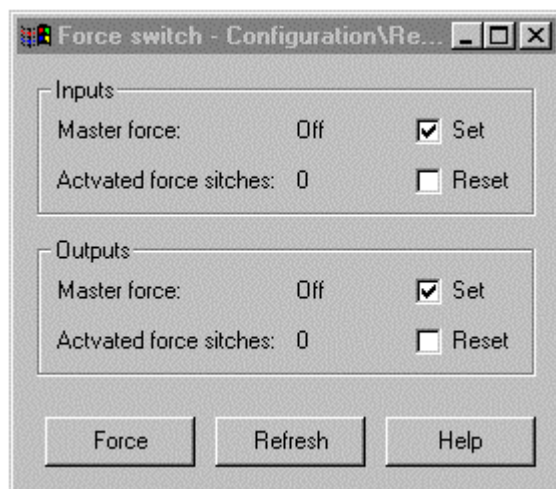


Рис. 101 Включение инициирования

Шаг 3:

Запустите онлайнный тест, как описано в главе 7.8.

Шаг 4:

Создайте поле онлайн-теста:

Указание: см. также главу 7.5, шаг 6.

- ☐ Щелкните по полю значения и держите клавишу нажатой.
- ☐ Перенесите поле OLT в необходимую Вам позицию.

Шаг 5:

Измените или активируйте значение инициирования.

Поле OLT разделено на три части:

1. Просмотр одиночного выключателя и статуса главного выключателя.
2. Просмотр статуса текущего сигнала входа или выхода.
3. Просмотр значения инициирования.



Рис. 102 Поле OLT булевой переменной

- ☐ Измените значение инициирования и выключатель инициирования, дважды щелкнув по соответствующей части поля OLT.

Выключатель инициирования и значение инициирования **изменяются непосредственно в контроллере**. Если главный выключатель инициирования еще не установлен, следует запрос ELOP II о необходимости его установки (см. рис. ниже).

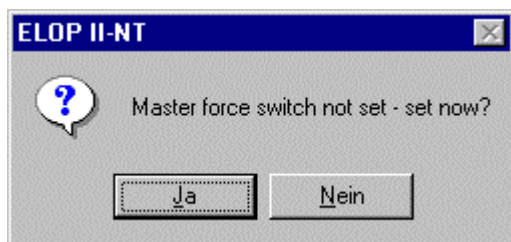


Рис. 103 Запрос на установку главного выключателя инициирования

Указание: не рекомендуется устанавливать главный выключатель инициирования через поле OLT, так как существует возможность ненужной активации уже установленного отдельного выключателя.

Возможные статусы поля OLT:

	Отдельный выключатель инициирования	Главный выключатель инициирования	Значение инициирования	Текущее значение	Выход
<input type="checkbox"/> TRUE <input type="checkbox"/> FALSE	Не активен	Не активен	FALSE (ЛОЖНО)	TRUE (ИСТИННО)	Текущее значение
<input type="checkbox"/> TRUE <input type="checkbox"/> FALSE	Активен	Не активен	FALSE (ЛОЖНО)	TRUE (ИСТИННО)	Текущее значение
<input checked="" type="checkbox"/> TRUE <input type="checkbox"/> FALSE	Не активен	Активен	FALSE (ЛОЖНО)	TRUE (ИСТИННО)	Текущее значение
<input type="checkbox"/> TRUE <input checked="" type="checkbox"/> FALSE	Активен	Активен	FALSE (ЛОЖНО)	TRUE (ИСТИННО)	Значение инициирования
<input type="checkbox"/> TRUE <input checked="" type="checkbox"/> TRUE	Активен	Активен	TRUE (ИСТИННО)	TRUE (ИСТИННО)	Значение инициирования

Указание: если поле OLT добавлено к логике, и Вы закрываете функциональный блок, появляется запрос о необходимости сохранения изменений. Сохранение полей OLT не влияет на версию кода проекта.

Указание: если входная переменная инициализирована, инициализированное значение используется во всей логике. Если инициализирована выходная переменная, инициализируется только физический выход. Если выходная переменная используется в логике, значение переменной командой на инициализацию не изменяется, это значение используется в логике и в дальнейшем.

7.10 Документация

В объекте документа может быть создан и структурирован вывод логики на печать. Все программные компоненты распечатываются с помощью объекта документации. Документирование аппаратного обеспечения производится отдельно, но может быть вставлено в виде файла в объект документа.

Шаг 1:

Создайте новый объект документа:

- ☐ Щелкните на проект.
- ☐ Откройте контекстное меню проекта.
- ☐ Выберите **New, Documents (Новые, Документы)**.

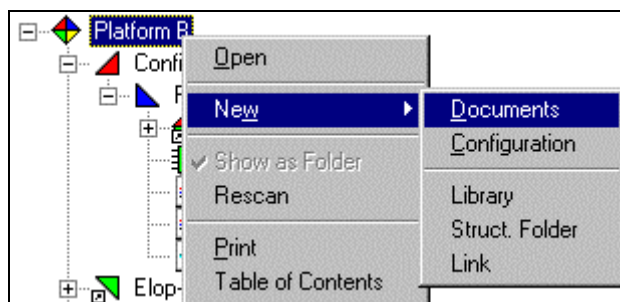


Рис. 104 Создание объекта документа

Шаг 2:

Переименуйте новый объект документа:

- ☐ Дважды с паузой щелкните на имя, которое Вы хотите изменить. Открывается поле ввода, куда Вы можете ввести новое имя.

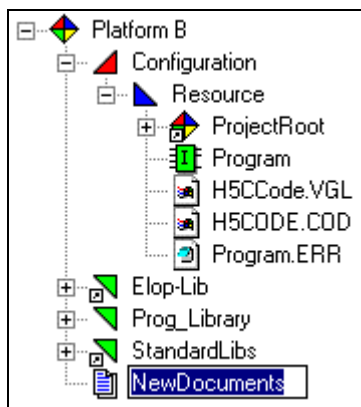


Рис. 105 Переименование объекта документа

Шаг 3:

Вставьте все данные Вашего проекта в объект документа:

- ☐ Откройте объект документа двойным щелчком.
- ☐ Щелкните на Ваше имя проекта и удерживайте клавишу нажатой.
- ☐ Перенесите проект в объект документа.

Теперь в объекте документа показаны все элементы Вашего проекта.

Указание: Вы можете вставить в документацию и отдельные объекты из Вашего проекта.

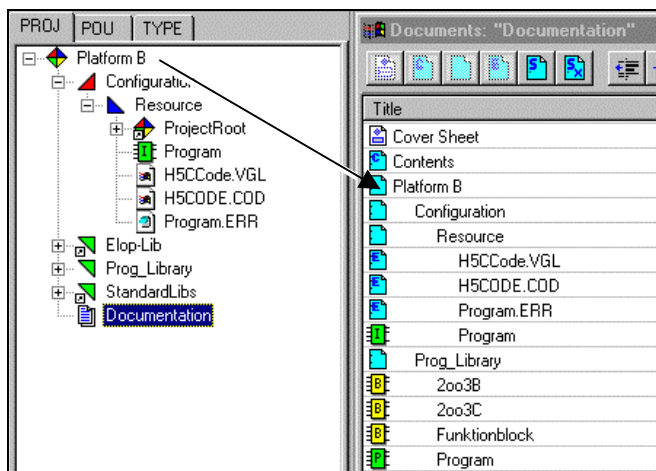


Рис. 106 Добавление проекта к документации

Шаг 4:

Обновите оглавление:

- ☐ Щелкните на кнопку **Update TOC** (Обновить оглавление).

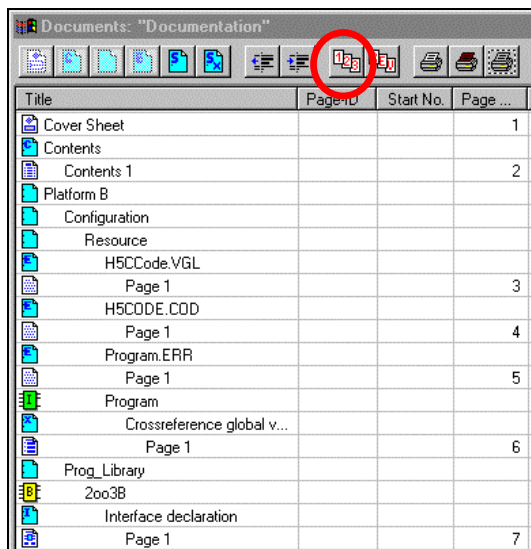


Рис. 107 Обновление оглавления

После обновления оглавления будет показана нумерация страниц.

Разделы в объекте документа соответствуют папке проекта и папкам библиотек. Отступы показывают их иерархию в окне структуры.

Указание: Вы можете изменить порядок следования элементов или удалить отдельные элементы. После этого не забудьте обновить оглавление!

Шаг 5:

Распечатайте документацию:

- ☐ Запустите печать всех, модифицированных или выбранных страниц через контекстное меню или с помощью кнопки на панели инструментов редактора документов.

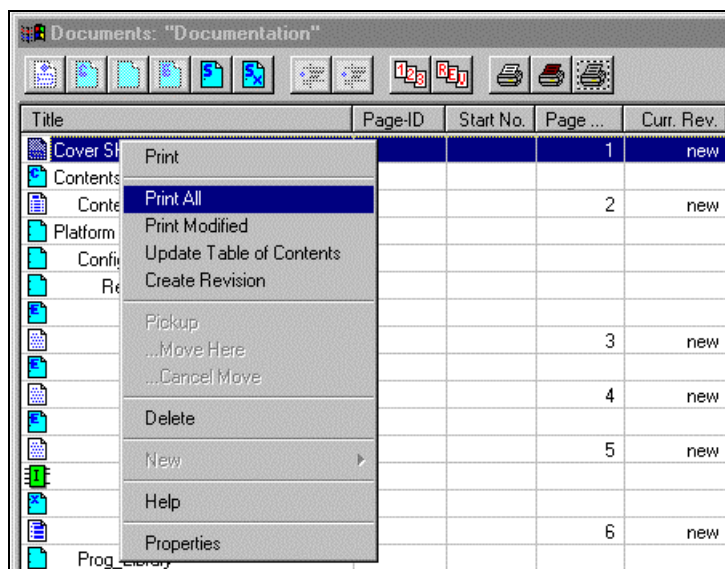


Рис. 108 Запуск печати

Шаг 6:

Откройте документацию ресурса:

Указание: в документации ресурса содержатся все данные, относящиеся к аппаратному обеспечению.

- ☐ Щелкните на ресурс.
- ☐ Откройте контекстное меню ресурса.
- ☐ Выберите **Documentation, RES-DOCU (generated)** (Документация, Документация ресурса (сгенерированная)).

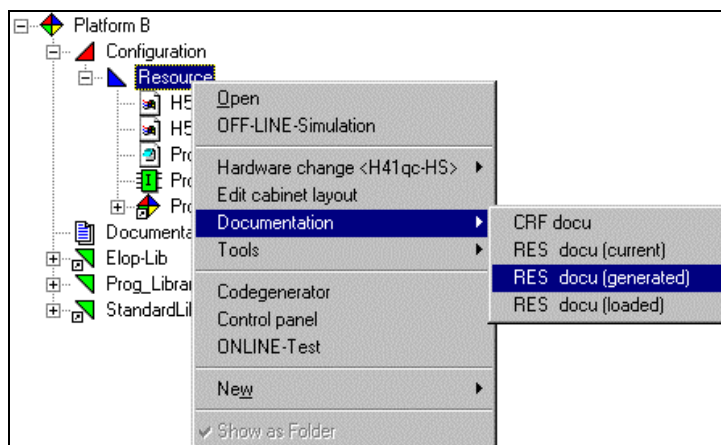


Рис. 109 Открытие документации ресурса

Шаг 7:

Запустите печать документации ресурса:

- ☐ В окне "RES docu generated ..." (Сгенерированная документация ресурса) выберите закладку **Rack** (**Каркас**).
- ☐ Щелкните на кнопку **Print (Печать)**.

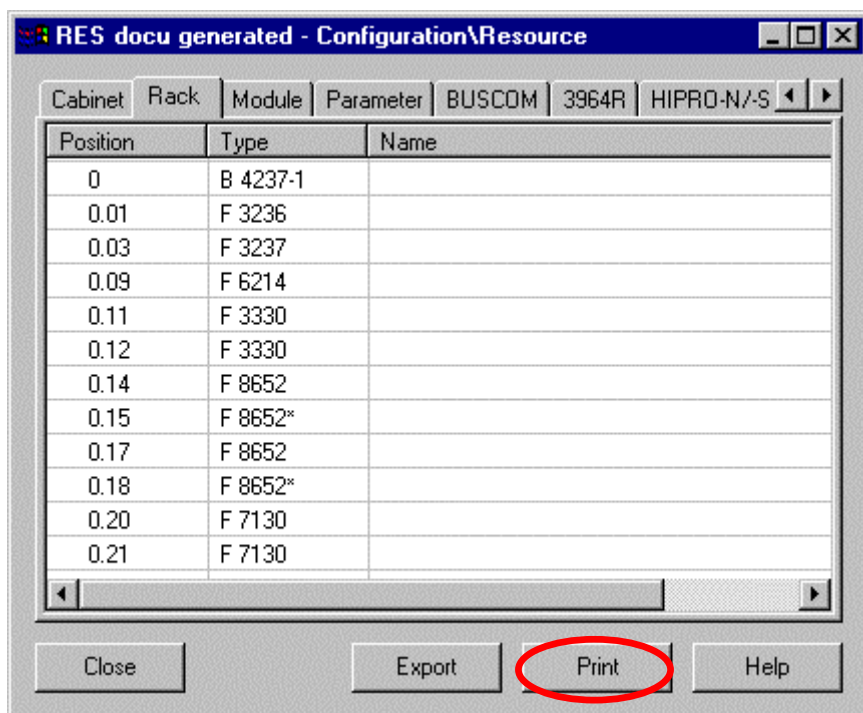


Рис. 110 Документация ресурса

Указание: кроме того, Вы можете экспортировать документацию ресурса в файл (создаваемый в том же каталоге, что и сам проект) и включить ее в объект документа.
Внимание: печатаются или экспортируются все поля!

7.11 Резервирование проекта

7.11.1 Архивация

Шаг 1: Выполните архивацию проекта:

- ☐ Щелкните на проект.
- ☐ Откройте контекстное меню проекта.
- ☐ Выберите **Backup... (Резервирование)**.

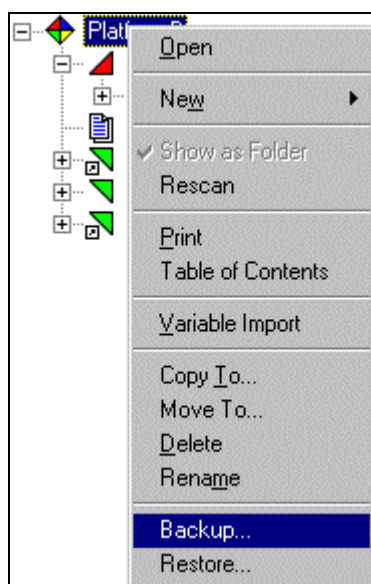


Рис. 111 Архивация проекта

Шаг 2:

Введите путь архива.

- ☐ В окне "Backup" (Резервирование) (Рис. 112) выберите путь архивации проекта. Введите каталог в поле **Target-file (Целевой файл)** или же нажмите на кнопку **Browse... (Обзор)** для выбора каталога (Рис. 113).
- ☐ В поле **Description (Описание)** введите ясное и понятное описание проекта.
- ☐ Щелкните на кнопку **Backup (Резервировать)**.

Проект архивируется в выбранный каталог. Архив проекта получает расширение ".L3P" и состоит из трех файлов (Рис. 114).

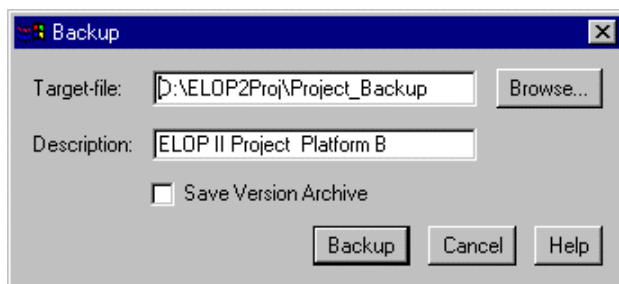


Рис. 112 Ввод имени архива

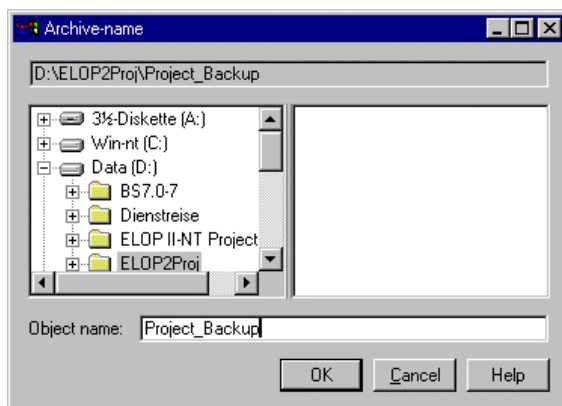


Рис. 113 Выбор пути архивации

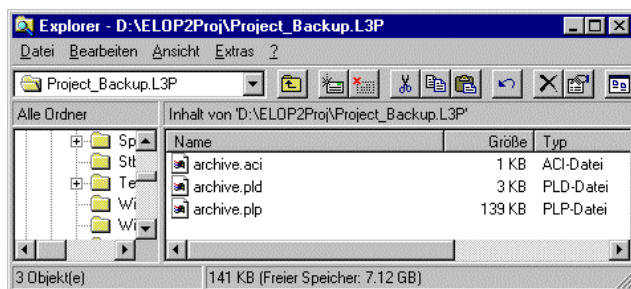


Рис. 114 Файлы архива проекта

7.11.2 Восстановление проекта

Указание: при восстановлении проекта из архива в ELOP II не должно быть открытого проекта.

Шаг 1: Закройте (Close) открытый проект.

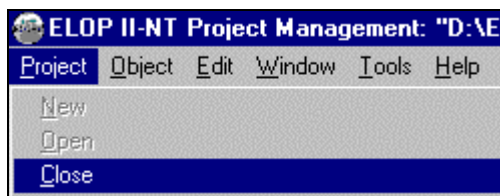


Рис. 115 Закрытие проекта

Шаг 2: Запустите восстановление проекта (Restore Project):

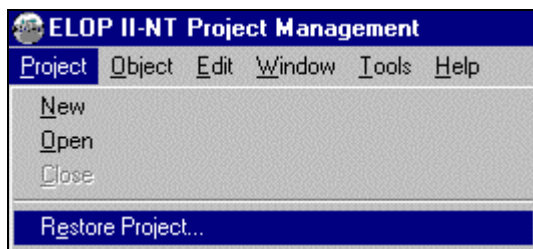


Рис. 116 Восстановление проекта

Шаг 3:

Выберите подлежащий восстановлению проект:

- В окне "Restore" (Восстановить) (Рис. 117) выберите тот целевой каталог в поле **Target-directory**, куда Вы хотите сохранить восстановленный проект. Используйте кнопку **Browse... (Обзор)** для выбора уже существующего каталога.

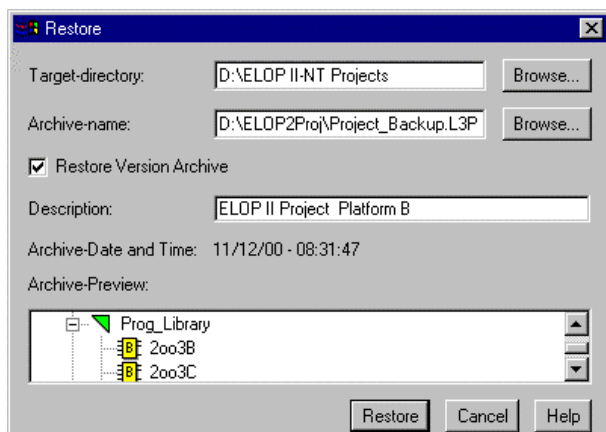


Рис. 117 Восстановление архива

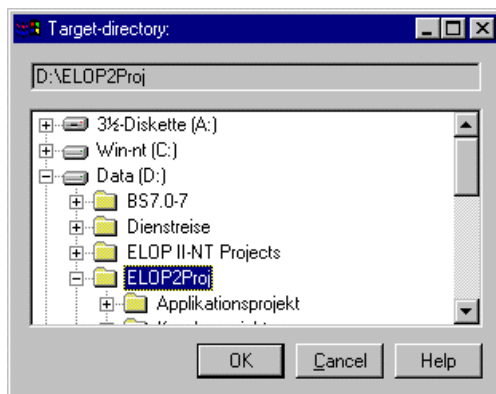


Рис. 118 Выбор целевого каталога

- ☐ В окне "Restore" (Восстановить) (Рис. 117) выберите имя архива в поле **Archive-name** и путь восстанавливаемого проекта. Используйте кнопку **Browse... (Обзор)** для выбора архива с помощью мыши.

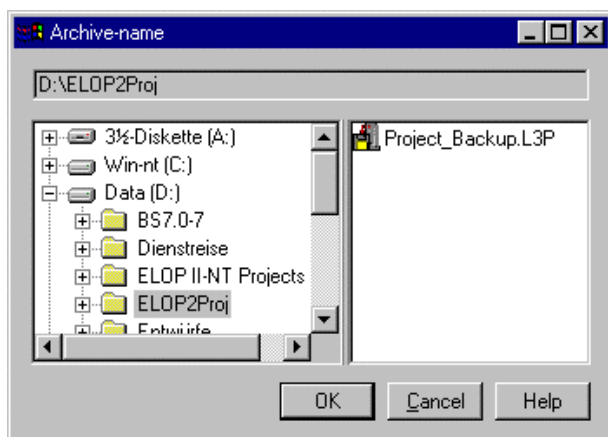


Рис. 119 Выбор архива

После восстановления проекта он автоматически открывается и выводится в окно структуры.

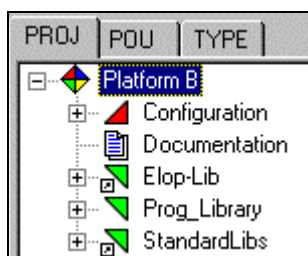


Рис. 120 Восстановленный проект

Указание: Вы можете архивировать отдельные объекты проекта и восстанавливать их в других проектах.

8 Приложения

8.1 Указатель

.COD	87
.ERR	87
.L3P	115
.VGL	87
Adobe Acrobat Reader©	5
Baud rate (Скорость передачи данных в бодах)	92
Bitstr (Битовая цепочка)	61
Control Center (Центр управления)	5, 7
Duplicate (Копирование)	63
DXF	34
Edit page data (Редактирование данных страницы)	62
Explorer (Проводник)	5
Hardware change (Изменить аппаратное обеспечение)	38
HIBUS	89
PADT (PC)	91, 94
Parity (Четность)	92
Plug-Ins	26
Programming and Debugging Tool (Инструмент программирования и отладки)	91
Quick-Info (Всплывающая подсказка)	10
StandardLibs (Стандартные библиотеки)	61
Stop bits (Стоповые биты)	92
Автономное моделирование	19, 36, 72
запуск ~	76
Администратор	3
Активная страница	26
Аппаратный ключ	1, 3, 4
Архивация	114
Библиотека	17, 19, 22, 31
новая ~	50
переименовать ~	51
Библиотеки в	
окне структуры каталогов	22
Блок	9
Булево значение	36
Версия кода	43, 47, 99, 106
Внешняя переменная	18

Восстановление	116
Время цикла	19
Встроенная справка	30
Генератор кода	43, 86
сообщения ~	87
Деинсталляция	6
Декларация интерфейса	57
Дерево проекта	20
Документация	2, 14, 20, 108
~ по аппаратному обеспечению	48
~ ресурса	112
Добавление проекта	110
новый объект ~	108
создание ~	33
Донгл	1
Драйвер аппаратного ключа	3
Загрузка	45, 95, 96
Задача	19, 36
новая ~	72
присвоить ~	74
Запуск инсталляции	5
Имя экземпляра	59
Инициирование	102
входной переменной	107
выходной переменной	107
главный выключатель инициирования	104
значение инициирования	105
настройки выключателя инициирования	103
отдельный выключатель инициирования	104
Инсталляция	3, 5
Каркас	82
Коммуникационная	
~ шина	44
добавить ~ систему	89
Коммуникация	88
Компьютер	4
Контекстное меню	12
Конфигурация	19, 22
новая ~	67
Копирование объекта	63
Локальная переменная	28
Масштабирование	27

Менеджер проекта	5, 7, 8, 21
Множественная установка	4
Модули	40
аналоговые ~ В/В	83
присвоение ~	84
расстановка ~	83
цифровые ~ В/В	83
Нумерация страниц	26
Область диаграммы	30
Объект документа	
переименовать ~	109
Объекты в окне структуры проекта	17
Окно обзора	13, 23, 24
Окно статуса и ошибок	7, 8, 15, 24
Окно структуры проекта	8, 11
Онлайн-тест	
поле ~	78, 105
Онлайн-тест (OLT)	47, 99, 100
Остановка	96
Панель инструментов	8, 10
Панель управления	44, 95
Папка проекта	21
Перезагрузка	45, 96
Переменная	
VAR_External	28
VAR_Global	28
VAR_Input	28, 56
VAR_Output	28, 56
добавить ~	64
описание ~	54
редактор ~	54
присвоение системных ~	42
список ~	64
Перенос	
~ переменных	30, 64
~ функциональных блоков	31
Подробное имя	56
Подчиненный компонент	90
Поле значения	30, 79
область ~	64
Принтер	4
Присвоение В/В (входа/выхода)	40

Проведение соединительных линий	65
Программа	
компилирование ~	86
создать ~	66
тип ~	39
новая ~	52
присвоить ~	70
экземпляр ~	36, 39, 71
новая ~	69
Программа сравнения кода	43
Программный компонент	
имя ~	58
размер ~	60
Программный компонент (POU)	17
Проект	9, 11, 17, 22
путь ~а	49
создание ~	49
Рабочая область	8, 13, 24, 25
Редактор деклараций интерфейсов	13, 23
Редактор диаграммы функционального блока	13, 23, 24
Редактор диаграммы функциональных блоков	8, 10
Редактор документов	13, 14
Редактор описаний переменных	13, 23, 24
Резервирование	114
Ресурс	19, 22
новый ~	68
определение ~а	38
переименовать ~	68
тип ~а	37
присвоение ~а	80
Сетевая инсталляция	3, 6
Сетевой диск	4
Сетка	63
Сеть	6
Система координат	26
Системная переменная	19
Соединение элементов	32
Создание логики	28, 61
Сообщение об ошибке	15
Справка	15
Стандартный экран	8
Стартовое меню	5

Столбцы системы координат	26
Строка меню	8, 9
Строка названия	8, 9
Строка состояния	8, 10
Строки системы координат	26
Схема коммутирующего шкафа	81
Техническая поддержка	2
Тип данных	41, 85
Тип программы	17, 19
Тип функционального блока	18
Управление версиями	14, 20
Управление документами	33
Упражнения	49
Функциональный блок	
создать ~	54
тип ~	53
Функция	18
Холодный пуск	97
Целевой файл	115
Центральный процессор	42
Шаблон проекта	21
Шаблоны	35
Шина	
добавить ~	89
добавить пользователя ~	90
параметры ~	92
параметры станции ~	90
тип ~	89
Экземпляр программы	17, 19
Язык последовательных функциональных схем	13
Язык функциональных блоков	13

8.2 Список сокращений

BSN	Bus Subscriber Number (Абонентский номер шины)
CB	Coprocessor module (Модуль сопроцессора)
CFG	Configuration (bus documentation) (Конфигурация (документация шины))
CG	Code Generator (Генератор кода)
CONST	Constant (Постоянная)
CRC	Cyclic redundancy check (Циклический избыточностный контроль)
CRF	Cross reference (Info on inputs and outputs) (Перекрестная ссылка (информация по входу или выходу))
CSV	Формат данных для функции импорта/экспорта, формат ASCII со значениями, разделенными запятыми (comma separated value)
CU	Central Unit (Центральный процессор)
DXF	Стандартный графический формат AutoCAD для печати
IO	Input/output (Вход/выход, В/В)
FB	Function block (Функциональный блок)
FBD	Function Block Diagram (Диаграмма функционального блока)
FUN	Объект типа функции
GV	Global variable (Глобальная переменная)
HIPRO-N / S	Коммуникация Hibus, не связанная / связанная с безопасностью
HW	Hardware (Аппаратное обеспечение)

LgP	Управляемая диаграммой логики запись в журнал
OLS	Offline simulation (Автономное моделирование)
OLT	Online Test (Онлайновый тест)
PADT	Programming and debugging Tool (PC) (Инструмент программирования и отладки – компьютер)
PES	Programmable Electronic System (Программируемая электронная система)
PFR	Product Feedback Report (Отчет потребителя продукта)
POU	Program Organization Unit (Программный компонент)
RES	Resource (Ресурс)
RETAIN	Сохранение производительности
RT	Resource Type (Тип ресурса)
RWP	Read/write parameter (Параметр считывания/записи)



HIMA Paul Hildebrandt GmbH + Co KG
Системы промышленной автоматизации