



Программирование ПЛК

РУКОВОДСТВО ПОЛЬЗОВАТЕЛЯ

ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ «СКПРО»

Справочная и правовая информация

Информация о руководстве

Данное руководство содержит указания и положения, которые необходимо соблюдать для обеспечения персональной безопасности и предотвращения материального ущерба.

Опасность

Указывает что несоблюдение надлежащих мер безопасности приведет к существенному вреду здоровью или серьезному материальному ущербу

Предостережение

Указывает что несоблюдение надлежащих мер безопасности может привести к существенному вреду здоровью или серьезному материальному ущербу

Предупреждение

Указывает что несоблюдение надлежащих мер безопасности может привести к материальному ущербу

Указание

Указывает на информацию, которая обязательно должна быть принята к сведению.

Квалифицированный персонал

Работать с продуктом или системой, описываемой в данной документации, должен только квалифицированный персонал, допущенный для выполнения поставленных задач и соблюдающий указания документации, в частности, указания и предупреждения по технике безопасности. Квалифицированный персонал в силу своих знаний и опыта в состоянии распознать риски при обращении с данными системами и избежать возникающие угрозы.

Использования продукта или системы

Продукт или систему разрешается использовать только для целей, указанных в соответствующей технической документации.

Исходными условиями для безошибочной и надёжной работы продукта или системы являются надлежащие транспортировка, хранение, размещение, монтаж, оснащение, ввод в эксплуатацию, обслуживание и поддержание в исправном состоянии. Обязательно учитывайте указания в соответствующей документации.

Товарные знаки

Товарные знаки «СК-1000», «СК-4000», «СКПро» являются собственностью АО «СибКом». Все другие товарные знаки, используемые или упоминаемые в данном руководстве, являются собственностью их уважаемых владельцев. Никакая информация, содержащаяся в этом руководстве, не должна быть истолкована как предоставление тем или иным образом лицензии или права на использование любого товарного знака.

Исключение ответственности

Содержимое данного руководство проверено на соответствие с описанным программным и аппаратным обеспечением. Тем не менее, разночтения, либо отклонения не могут быть исключены, в связи с чем мы не гарантируем полное соответствие. Данные в данном руководстве подвергаются регулярной проверке и соответствующие изменения вносятся в последующие редакции данного руководства.

СОДЕРЖАНИЕ

1	0Б3О	Ρ	6
	1.1 Occ	БЕННОСТИ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ	6
	1.1.1	Совместимость с операционными системами	6
	1.1.2	Поддержка международного стандарта МЭК	6
	1.1.3	Управление проектами – древовидная структура управления	7
	1.1.4	Языки программирования	7
	1.1.5	Режим программирования — Перекрестный вызов	7
	1.1.6	Вычислительные возможности — широкий спектр управляющих вычислительн	ых
	функц	ий	8
	1.1.7	Функция отображения — интуитивно понятный онлайн-просмотр	8
	1.1.8	Функция модифицирования — изменения в онлайн-режиме	8
	1.1.9	Функция отладки — мощный онлайн-отладчик	8
	1.1.10	Функция мониторинга — онлайн-мониторинг в реальном времени	8
	1.1.11	Симулятор – моделирование без реального аппаратного обеспечения	9
	1.1.12	Средство диагностики	9
	1.1.13	Режим передачи	9
	1.1.14	Языковая поддержка	9
	1.1.15	Визуальный режим печати	9
	1.1.16	Дружественный пользовательский интерфейс	9
	1.2 Сис	ТЕМНЫЕ ТРЕБОВАНИЯ	9
	1.2.1	Требования к аппаратному обеспечению	9
	1.2.2	Требования к операционным системам	10
	1.3 Уст/	АНОВКА ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ	10
2	РАБО	ТА С ПРОГРАММНЫМ ОБЕСПЕЧЕНИЕМ «СКПРО»	11
	2.1 Раб		11
	2.1.1	Рабочий интерфейс	11
	2.1.2	Основной функционал каждого окна	11
	2.2 BBE	ДЕНИЕ В СТРУКТУРУ МЕНЮ	12
	2.2.1	Главное меню и выпадающее меню	12
	2.2.2	Подпункты меню	13
	2.2.3	Контекстное или всплывающее меню	13
	2.3 Фун	КЦИОНАЛ МЕНЮ	14
	2.3.1	Меню File	14
	2.3.	1.1 Команды меню File	14
	2.3.	1.2 Функционал меню File	15
	2.3.2	Меню Edit	20
	2.3.	2.1 Команды меню Edit	20
	2.3.	2.2 Функционал меню Edit («Редактирование»)	21
	2.3.3	Меню View	26
	2.3.	3.1 Команды меню View	26
	2.3.	3.2 Функционал меню View («Вид»)	27
	2.3.4	Меню LD	27
	2.3.	4.1 Команды меню LD	27

	2.3.4.2 Функционал меню LD	28
	2.3.5 SCC	
	2.3.5.1 Команды меню SCC	31
	2.3.5.2 Функционал меню SCC	32
	2.3.6 Online	34
	2.3.6.1 Команды меню Online	
	2.3.6.2 Функционал меню Online	35
	2.3.7 Меню Download	37
	2.3.8 Меню Window	37
	2.3.8.1 Команды меню Window («Окно»)	37
	2.3.9 Меню Help («Справка»)	38
	2.3.9.1 Команды меню Help («Справка»)	38
	2.3.9.2 Функционал меню Help («Справка»)	38
	2.4 Системная панель инструментов	39
	2.5 Панель инструментов LD	40
	2.6 Панель инструментов FBD	42
	2.7 Панель инструментов IL	43
	2.8 ПАНЕЛЬ ИНСТРУМЕНТОВ ST	44
	2.9 ПАНЕЛЬ ИНСТРУМЕНТОВ SCC	
	2.10 Окно вывода	
	2.11 СПРАВОЧНИК ПО ГОРЯЧИМ КЛАВИШАМ	47
3	З РАБОТА С ПРОЕКТАМИ	51
	3.1 Браузер проекта	51
	3.2 Создание нового проекта	52
	3.2.1 Создание проекта	
	3.2.1 Создание проекта 3.2.2 Конфигурация аппаратного обеспечения ПЛК	
	3.2.1 Создание проекта 3.2.2 Конфигурация аппаратного обеспечения ПЛК 3.3 Программа	
	 3.2.1 Создание проекта	
	 3.2.1 Создание проекта	
	 3.2.1 Создание проекта	
	 3.2.1 Создание проекта 3.2.2 Конфигурация аппаратного обеспечения ПЛК 3.3 ПРОГРАММА	
	 3.2.1 Создание проекта	
	 3.2.1 Создание проекта	52 53 53 61 62 62 63 63 63 63 64 65
	 3.2.1 Создание проекта	
	 3.2.1 Создание проекта 3.2.2 Конфигурация аппаратного обеспечения ПЛК. 3.3 ПРОГРАММА. 3.3.1 Добавление программы. 3.3.2 Удаление программы. 3.3.3 Изменение названия программы. 3.3.4 Добавление описания программы. 3.3.5 Защита программы. 3.4 ЗАДАЧА (ТАЅК). 3.5 ПРЕРЫВАНИЕ. 3.6 ЗАЩИТА ПРОЕКТА. 	52 53 53 61 62
	 3.2.1 Создание проекта 3.2.2 Конфигурация аппаратного обеспечения ПЛК. 3.3 ПРОГРАММА. 3.3.1 Добавление программы. 3.3.2 Удаление программы. 3.3.3 Изменение названия программы. 3.3.4 Добавление описания программы. 3.4 Задача (Таѕк). 3.5 Прерывание. 3.6 Защита проекта. 3.7 Подключение и отключение. 	52 53 53 61 62 62 62 63 63 63 63 63 64 65 65 67 68 69
	 3.2.1 Создание проекта 3.2.2 Конфигурация аппаратного обеспечения ПЛК	52 53 53 61 62 62 63 63 63 64 65 65 67 67 68 69
	 3.2.1 Создание проекта	52 53 53 61 62 62 62 63 63 63 63 63 64 65 65 67 68 69 69 69
4	 3.2.1 Создание проекта	
4	 3.2.1 Создание проекта	
4	 3.2.1 Создание проекта	
4	 3.2.1 Создание проекта 3.2.2 Конфигурация аппаратного обеспечения ПЛК. 3.3 ПРОГРАММА. 3.3.1 Добавление программы. 3.3.2 Удаление программы. 3.3.3 Изменение названия программы. 3.3.4 Добавление описания программы. 3.3.5 Защита программы. 3.4 Задача (Таѕк). 3.5 ПРЕРЫВАНИЕ. 3.6 Защита проекта. 3.7 Подключение и отключение 3.8 Загрузка и выгрузка файла проекта. 3.9 Загрузка и выгрузка программ. 4 УПРАВЛЕНИЕ ДАННЫМИ 4.1 ТИП ДАННЫХ 4.2 УПРАВЛЕНИЕ ДАННЫМИ 4.1 Вкладка "Data". 	
4	 3.2.1 Создание проекта	52 53 53 61 62 62 63 63 63 63 63 64 65 67 67 68 69 67 68 69 67 71 68 69 71 73 73 74 74
4	 3.2.1 Создание проекта	
4	 3.2.1 Создание проекта	

4.2.4 Таблица дополнительных точек	
4.3 Режим адресации	
5 БАЗОВЫЕ БЛОКИ ФУНКЦИЙ И ПРОГРАММИРОВА	НИЕ НА АЗРІКАХ
ПОДДЕРЖИВАЕМЫХ ПО «СКПРО»	
6 ЗАГРУЗКА И ВЫГРУЗКА ФАИЛА ПРОЕКТА «СКПРО»	90
6.1 Ручная загрузка	
6.2 ЗАГРУЗКА ПРОЕКТА	
6.3 Выгрузка проекта	
7 ОТЛАДКА ПРОГРАММЫ	94
	04
7.1 ОГЛАДКА LD7 FBD	
7.1.1 OHJUUH-MOOUYUKUUUN M DJK	
7.2 OTDADKU C 60109990001101K	9J 97
7.2 ОГЛАДКА ЭСС	, 97 97
7.2.7 Наблюдение за выполнением	97 97
7.2.2 Партовстие за соптолнение	98
7.2.4 Выполнение отладки	98
7.2.5 Блокировка и разблокировка	
7.3 Отладка IL	
7.4 Отладка ST	
7.5 Симулятор	
	105
о ПРИЛОЖЕНИЕ	
8.1 Сервис и поддержка	
8.1.1 Контакты службы технической поддержки	
8.1.2 Порядок оказания технической поддержки по изделию	
8.2 Лист изменений	

1 Обзор

Данное руководство описывает базовые принципы работы с программным обеспечением «СКПро» и предназначено для пользователей программного обеспечения «СКПро».

Программное обеспечение «СКПро» является важной составляющей экосистемы контроллеров СК-1000 и СК-4000 и представляет собой интегрированную среду программирования для ПЛК, включающую в себя редактор, компилятор, отладчик, эмулятор и инструмент графического пользовательского интерфейса, и используемую для выполнения конфигурирования оборудования, настройки каналов ввода-вывода, программирования прикладного ПО, моделирования, отладки и загрузки. Данное программное обеспечение предоставляет персоналу инженерного и технического подразделений простой и практичный инструмент для программирования и онлайнотладки программ ПЛК.

ПО «СКПро» предоставляет полный ряд функций, которые могут помочь достичь более высокой производительности и лучшей совместимости программного обеспечения. За счет снижения стоимости разработки и оптимизации рабочих параметров, ПО «СКПро» позволяет оптимизировать вложения заказчика в программное обеспечение, снижает затраты на обучение и предоставляет уникальный потенциал для развития и обеспечения совместимости.

ПО «СКПро» поддерживает такие языки, как язык релейно-контактных схем (LD), язык функциональных блоковых диаграмм (FBD), список инструкций (IL) и структурированный текст (ST) в соответствии со стандартом МЭК 61131-3, а также поддерживает специфический язык последовательных управляющих схем (SCC).

1.1 Особенности программного обеспечения

1.1.1 Совместимость с операционными системами

Сегодня графический пользовательский интерфейс является базовым функционалом для ряда задач. По этой причине ПО «СКПро» разработано, как программа для Microsoft Windows и совместимо со всеми современными версиями Windows. ПО «СКПро» разработано на языке VC++6.0 для Windows, потому его интерфейс полностью совместим с дизайном окон данной системы. Это включает в себя стандарты построения меню, использования горячих клавиш, мыши и инструментальной панели. Применение данного решения снижает время на обучение персонала и производственные затраты за счет простоты использования.

1.1.2 Поддержка международного стандарта МЭК

Из-за различий в системах инструкций у производителей ПЛК и различий в требованиях конечных пользователей к методам программирования, Международная электротехническая комиссия (МЭК) разработала стандарт языков программирования МЭК 61131-3 для систем в среде Windows (в 1993, МЭК обнародовала международный

стандарт на ПЛК – МЭК 61131), который определяет пять языков программирования: список инструкций (IL), структурированный текст (ST), релейно-контактные схемы (LD), функциональные блоковые диаграммы (FBD), последовательная функциональная схема (SFC). Стандарт включает в себя как текстовое (IL, ST), так и графическое (LD, FBD) программирование, и SFC, который можно отнести к обеим группам. Это стандартизованный список, с которым программируемый логический контроллер – устройство, созданное на основе цифровых технологий – работает на высоком уровне, что является определяющим трендом развития ПЛК.

Программное обеспечение «СКПро» обеспечивает унифицированную и эффективную среду для конфигурирования системы, в соответствии с международным стандартом МЭК 61131-3, позволяя специалисту «выучив раз, применять везде».

1.1.3 Управление проектами – древовидная структура управления

Программное обеспечение «СКПро» применяет концепции управления проектами, отображая их в виде древовидной структуры в интегрированной среде разработки и визуально отображает содержимое программы в виде нескольких документов. Таким образом, всё соответствующее содержимое, а также разработка или внесение изменений в программу становятся интуитивно понятными.

1.1.4 Языки программирования

Как решение для промышленной автоматизации, программное обеспечение «СКПро» поддерживает стандартные языки, совместимые со стандартом МЭК 61131-3: список инструкций (IL), язык релейно-контактных схем (LD), язык функциональных блоковых диаграмм (FBD) и структурированный текст (ST), и предоставляет новый тип языка программирования последовательных управляющих схем (SCC - Sequential Control на вышеперечисленных языках, Chart). Программа, созданная может быть более перекрестно вызвана, что делает программирование гибким И соответствующим требованиям различных сложных условий. При этом LD, FBD и SCC используют графический редактор и являются гибкими, удобными и быстрыми. Все языки поддерживают функции быстрого доступа Cut («Вырезать»), Copy («Копировать»), Paste («Вставить»), Delete («Удалить»), Undo («Отменить»), Redo («Повторить»), Find and Replace («Найти и заменить») и т.д.

1.1.5 Режим программирования — Перекрестный вызов

Управляющие программы состоят из программ с логической структурой. В рамках одной программы применим только один язык программирования. Все программы объединяются для создания целостного проекта управления процессами. В программах могут вызываться подпрограммы, написанные на различных языках программирования МЭК (LD, FBD, IL, ST) и SCC.

1.1.6 Вычислительные возможности — широкий спектр управляющих вычислительных функций

В программное обеспечение «СКПро» встроен ряд стандартных операторов, блоков управления функциями и стандартных функций. Среди этих функциональных блоков есть такие, как импульсный выход, коммутаторы master/slave, сетевые функции и функции последовательных коммуникаций и т.п., что позволяет инженерам с легкостью решать комплексные задачи по управлению процессом и сокращает время цикла разработки проекта.

1.1.7 Функция отображения — интуитивно понятный онлайн-просмотр.

Система позволяет в режиме онлайн отслеживать рабочий статус схем на языке LD. Красный сигнал означает, что схема замкнута, зеленый – разомкнута. Всё интуитивно понятно с первого взгляда.

Для языка SCC отслеживается не только рабочий статус, но и шаги выполнения. Так же можно внедрять установку и сброс таймера, сигнал тревоги и т.п., что позволяет специалистам-инженерам с легкостью использовать различные функции.

1.1.8 Функция модифицирования — изменения в онлайн-режиме

Функция позволяет модифицировать параметры функциональных блоков, добавлять, удалять и перемещать функциональные блоки в режиме онлайн. Также возможно вносить модификации в программу, исполняемую в ПЛК, не останавливая его работу.

1.1.9 Функция отладки — мощный онлайн-отладчик

Все языки — список инструкций (IL), релейно-контактные схемы (LD), функциональные блоковые диаграммы (FBD) и структурированный текст (ST), поддерживают функции отладки в режиме онлайн, такие как точка останова, пошаговое выполнение, и т.п.

Для языка SCC предусмотрено три режима: автоматическое выполнение, наблюдаемое выполнение и отладочное выполнение. Когда программа на языке SCC находится в режиме онлайн-отладки, выполнение каждой функционального бокса обозначается тремя различными цветами: серым для не выполненного, красным для выполняемого и синим для выполненного. Поддерживаются такие функции, как точка останова и пошаговое выполнение, а также завершение выполнения или перезапуск выполнения в любое время. Инженерный персонал может легко отлаживать программы и находить ошибки.

1.1.10 Функция мониторинга — онлайн-мониторинг в реальном времени

В режиме онлайн все точки данных могут управляться (принудительно устанавливаться, назначаться, наблюдаться), все значения переменных могут отслеживаться в таблице переменных; все события SOE могут просматриваться в таблице событий SOE; все сигналы тревоги могут запрашиваться в таблице сигналов тревоги; все сообщения об ошибках можно проверить на вкладке debug (отладка) в

окне вывода. Данные могут отображаться в трех форматах: десятичном, двоичном или шестнадцатеричном.

1.1.11 Симулятор – моделирование без реального аппаратного обеспечения

Программы могут быть разработаны и отлажены в симуляторе без необходимости использовать аппаратный ПЛК. «СКПро» может идеально моделировать аппаратные функции, точно воспроизводить поведение целевой программы и эффективно сокращать период разработки программы.

1.1.12 Средство диагностики

ПО «СКПро» обладает комплексными функциями диагностики приложений. В окне вывода могут отображаться все сбои системы и приложений. Двойное нажатие кнопки мыши в этом окне открывает доступ к редактору, что позволяет исправить недочёты программы.

1.1.13 Режим передачи

Все операции сохранения, загрузки и выгрузки результатов программирования применяются в файловом режиме, таким образом, они могут поддерживать все конфигурации программы в соответствии.

1.1.14Языковая поддержка

Для ввода имен переменных необходимо использовать стандартную латиницу. Для комментариев можно использовать языки, поддерживаемые операционной системой компьютера, на котором установлено ПО «СКПро».

1.1.15 Визуальный режим печати

Для задач документирования проекта «СКПро» поддерживает визуальный режим печати для распечатки конфигурации ПЛК, информации о точках данных, и программ на всех языках (LD, FBD, ST, IL и SCC) в том же виде, в каком пользователь видит их на экране.

1.1.16 Дружественный пользовательский интерфейс

«СКПро» в полной мере использует преимущества графических и контекстнозависимых интерфейсов Windows. Эргономика улучшена за счет оптимизации использования экранного пространства, прямого доступа к инструментам и информации, а также использования комментариев на различных языках и т.д.

1.2 Системные требования

1.2.1 Требования к аппаратному обеспечению

32-разрядный (x86) или 64-разрядный (x64) процессор с тактовой частотой 1 ГГц или выше.

2 ГБ (для 32-разрядного процессора) или 4 ГБ (для 64-разрядного процессора) ОЗУ.

2 ГБ свободного места на жестком диске. Руководство пользователя Программное обеспечение «СКПро» Графическое устройство с поддержкой разрешения SXGA (1280*1024).

Сетевая карта Ethernet 100 Мбит/с.

1.2.2 Требования к операционным системам

Поддерживается работа с операционными системами семейства Microsoft Windows, начиная с Windows 7.

1.3 Установка программного обеспечения

Запустите программу установки "setup.exe", и следуйте инструкциям, чтобы завершить установку.

2 Работа с программным обеспечением «СКПро»

ПО «СКПро» включает в себя законченную систему для конфигурирования ПЛК и разработки прикладных приложений, которая соответствует принципам работы в системе Windows и которая проста в использовании.

2.1 Рабочее окно

2.1.1 Рабочий интерфейс

Интерфейс программного обеспечения «СКПро» приведен на рисунке 2.1. Среда разработки включает в себя следующие элементы: меню, панель инструментов, браузер проекта, окно вывода, строку состояния и окно программы, положение каждого элемента показано на рисунке 2.1.



Рисунок 2.1 Рабочий интерфейс

2.1.2 Основной функционал каждого окна

Меню: Выполнение основных функций «СКПро».

Панель инструментов: Команды работы с файлами: "создать", "открыть" и "сохранить"; онлайн-операции: вход в систему, загрузка и выгрузка; а также вызов панели инструментов каждого языка программирования.

Строка состояния: Строка состояния находится в нижней части экрана. Справа от строки состояния находится информация о состоянии, такая как координаты

программы, онлайн / оффлайн, эмуляция, принудительная метка, процент свободной памяти и т.д. Содержимое каждой операции отображается слева от строки состояния.

Окно программы: Настройка системы, редактирование программы, отладка программы и т.д.

Браузер проекта: обеспечивает управление проектом.

Окно вывода: Отображение результатов операций поиска, компиляции и отладки программы.

2.2 Введение в структуру меню

2.2.1 Главное меню и выпадающее меню

Главное меню позволяет реализовывать основные функции ПО «СКПро», такие как **File** (Работа с файлами), **Edit** (Редактирование), **View** (Просмотр), **LD** (**FBD**, **SCC**, **IL**, **ST**, и т.д.), **Online** (Онлайн), **Load** (Загрузка), **Window** (Окно), и **Help** (Справка), а также другие, как показано на рисунке 2.2.



Рисунок 2.2 Главное меню

Чтобы вызвать выпадающее меню, щелкните левой кнопкой мыши по любому пункту главного меню, фон пункта меню визуально изменится, одновременно отобразится выпадающее меню. Наведите мышь на любой пункт выпадающего меню, пункт станет синим, это означает, что при нажатии кнопки мыши, выбрана будет эта операция; щелкните любое место любой клавишей мыши за пределами меню или нажмите клавишу **ESC**, чтобы отключить меню, как показано на рисунке 2.3:



Рисунок 2.3 Выпадающее меню

2.2.2 Подпункты меню

Каждый подпункт меню отображается в выпадающем меню. Наведите курсор мыши на любой подпункт меню, он станет синим, это означает, что при нажатии левой кнопки мыши, выбрана будет эта операция; щелкните любое место за пределами меню или нажмите клавишу **ESC**, чтобы отключить меню, как показано на рисунке 2.4.

1 SCPro - C:\SCPro-pr	oj\TE	ST\TEST.prj - [FP	Q2]			_
🐹 File Edit View	LD	Online Load	Window	Help	2	
D 🖬 🖹 🗞 🕼	~	Move				2 🗸
Control -		Block				
		Contact		>	L: → ≪ 🖍 🔎 🏣 🏲 😭 abc 🕀	h ())
SCPLC		Coil		>	-()- Coil	F6
🖻 🖷 TEST		Mathematics		>	-(/)- Negated Coil	F7
🖻 🗎 Progr		Statistics		>	-(P)- Positive Transition-sensing Coil	F8
		Logic		>	-(N)- Negative Transition-sensing Coil	F9
		Comparison		>	-(S)- Set Coil	F10
		Conversion		>	-(R)- Reset Coil	F11
😑 😐 User I		Data Move		>	8 %N0004	
⊞ ⊟ B		Timer		>		
		Counter		>	10 %N0005	
		Control		>		
		PLC		>	12 %N0008	
		Others		>		
		Link			14 %N0007	
		Negate	Alt+	F1		
		Label	Alt+	F2	18 %N0008	
		Goto	Alt+	-F3	5 00000 C	
		Return	Alt+	F4		
		Comment	Alt+	F5		
Program Data	~	Grid				
Trogram	~	Page Line				
0 error(s)	~	Execution Orde	r			
		Point Name				
		Insert	Ctr	I+J		
4		Remove	Ctrl	+K		
		Zoom	Ctrl	+D		
Compile Debug		Property				
Coil		Debug		>		

Рисунок 2.4 Подпункты меню

2.2.3 Контекстное или всплывающее меню

Щелкните объект правой кнопкой мыши, чтобы открыть контекстное меню; при выборе нескольких объектов также можно вызвать контекстное меню, но в этом случае меню будет содержать только те пункты, которые подходят для каждого из объектов. Щелкните в любом месте за пределами меню или нажмите клавишу ESC, чтобы отключить меню, как показано на рисунке 2.5.



Рисунок 2.5 Контекстное или всплывающее меню

2.3 Функционал меню

2.3.1 Меню File

2.3.1.1 Команды меню File

Меню File используется для управления файлами, включая следующие New (Новый), Open (Открыть), Save (Сохранить), User Management (Управление пользователями), Close (Закрыть), New Project (Новый проект), Save Project (Сохранить проект), New Program (Новая программа), Save Program (Сохранить программу), Save All Programs (Сохранить все программы), Compile Program (Компилировать программу), Compile All Programs (Компилировать все программы), Password (Пароль), Used Times (Количество обращений), Export Point (Экспорт точек данных), Import Point (Импорт точек данных), Print (Печать), Print Preview (Предварительный просмотр), Print Setup (Настройка), и Exit (Выход), как показано на рисунке 2.6.

R.	File	Edit	View	LD	Online	Load	Window	He
Ľ		New						h
С		Open.						
k		Save						
-		User N	/lanage	ment				1
Ë		Close						
		New F	roject				Ctrl+N	
		Save F	roject				Ctrl+S	
		New F	rogram	n			Ctrl+E	
		Save F	rogram	n			Ctrl+I	
		Save A	All Prog	rams				
		Comp	ile Prog	gram			Ctrl+R	
		Comp	ile All P	rogra	ims		Ctrl+B	
		File Pa	assword	1				
		Used 7	Times					
		Export	t Point				:	>
		Impor	t Point				:	>∥
		Print					Ctrl+P	
		Print P	review					
		Print S	Setup					
		1 Test.	wsp					
Рі		2 Ruk	wsp					ſ
		3 C:\S	CPro-p	roj\'	\400_test.	wsp		
U		4 C:\S	CPro-p	roj\'	\400_test.	wsp		
		Exit						

Рисунок 2.6 Меню File

2.3.1.2 <u>Функционал меню File</u>

[New] : Создать новый файл проекта, который будет содержать базу данных, программы LD, FBD, SCC, IL и ST и т.д.

(Open) : Открыть существующий файл проекта. Выберите **Open**, и программа **откроет** диалоговое окно "Открыть", тип файла - "SCPro Workspace Files", расширение - ".wsp". Если файл защищен паролем, появится диалоговое окно ввода **пароля**, как показано на рисунке 2.7.

👯 Open							\times
Папка:	П, Т	EST		•	🗢 🔁	➡ 🖩 🏷	
Имя — Test.v	vsp	^			Дата изи 30.09.20	менения 22 13:35	
<							>
Имя файла	a: [Открыть	
Тип файло	ов: [SCPro Workspace F	iles (*.wsp)		-	Отмена	
Pa	assw	vord				\times	
		Password	*				
		ОК		Ca	ncel		

Рисунок 2.7 Открыть проект

[Save] : Сохранить редактируемый файл проекта. Если редактируемый файл не является новым, то исходный файл будет перезаписан, как показано на рис.2.8:

🗱 Save As	×
Папка: 🚺 TEST 💌	. ⇔ 🗈 📸 🖽 ◄
Имя	Дата изменения
Нет элементов, удовлетворяющих ус	ловиям поиска.
<	>
Имя файла: TEST.wsp	Сохранить
Тип файла: SCPro Workspace Files (*.wsp)	• Отмена

Рисунок 2.8 Сохранить проект

※ Изменение одного файла проекта повлияет на каждую программу в нем, потому обязательно запускайте компиляцию всех программ через команду *Compile All Programs* перед загрузкой.

ī.

[New Program] : Создать новую программу. Программа может быть написана на языке LD, FBD, SCC, IL или ST и т.д. Новой программе необходимо дать название, также можно добавить описание, содержимое которого будет отображаться в области редактирования программы, как показано на рисунке 2.9.

New Program	×
Program Type C LD FBD IL ST C C/C++ SCC	Program Name Task/Interrupt Main Task Program Description
ОК	Cancel

Рисунок 2.9 Новая программа

Программа может быть основной задачей, нумерованной задачей, либо прерыванием. Более подробно - см. раздел 3.3 настоящего руководства.

[Save Program] : Сохранить редактируемую программу. При выборе команды **Сохранить программу**, «СКПро» автоматически сохраняет программу, переписывая существующий файл или создавая новый.

[Save All Programs] : Сохранить все открытые программы.

[Compile] : Скомпилировать текущую программу. Если программа изменена, «СКПро» предложит сохранить её. После этого «СКПро» автоматически проверит текущую программу на предмет ошибок, при наличии таких ошибок, она не сможет быть скомпилирована; вкладка **Compile** окна вывода демонстрирует место, тип и количество ошибок, как показано на рисунке 2.10:



MODULE_2 : (2,1) syntax error MODULE_2 : (2,1) DFB instance undeclared
MODULE 2: 2 error(s)
<
,
Compile Debug Find Alarm SOE

Рисунок 2.10 Скомпилировать программу

[Compile All Programs] : Скомпилировать все программы текущего проекта. Команда выполняет всё то же самое, что и команда Compile Program, «СКПро» предложит сохранить программу (если были внесены изменения). После этого «СКПро» автоматически проверит все программы на предмет ошибок, при наличии таких ошибок, они не смогут быть скомпилированы; также в отдельной вкладке отображаются место, тип и количество ошибок.

[Пароль] : Изменение пароля к файлу и пароля для входа в систему. Новым проектам не назначаются пароли к файлам, как показано на рис.2.11:

Password		Х
Now Password		
New Password		
Confirm New Password		
ОК	Cancel	

Рисунок 2.11 Пароль

[Used Times] : Статистический расчет применения каждой точки данных в программах показан на рисунке 2.12:

Nu	mber	1	Des	cription		Used	Times	1	/alue	
%N	0007						1			
%N	8000						1			
%N	0009						1			
%N	0010						0			
%N	0011						0			
%N0012							0			
%N0013							0			
I .	Q	IW	QW	м	MW	N	NW	S	SW	T

Рисунок 2.12 Количество применений

[Print] : Распечатать конфигурацию ПЛК, информацию о точках, LD, FBD, SCC, IL и ST. При выборе команды Print откроется стандартное диалоговое окно печати, в котором можно выбрать принтер, диапазон печати, копии и т.д. Как показано на рис.2.13: *Руководство пользователя* Программное обеспечение «СКПро» 18

Печать			×
Принтер			
Имя: Adobe	PDF	•	Свойства
Состояние: Готов Тип: Adobe Место: Docum Комментарий:	PDFConverter ents*.pdf		
Диапазон печати Все Страницы с: 1 С Въделенный фрагм	по: 1ент	Копии Число копий: ✓ Разобрать 123	1 то копиям 123
		ОК	Отмена

Рисунок 2.13 Печать

[Print Preview] : Предварительный просмотр результатов печати конфигурации ПЛК, сообщения о точках данных, LD, FBD, SCC, IL и ST.

(Print Setup) : Сбросить настройки печати. При выборе команды **Print Setup** откроется диалоговое окно **Print Setup**. Чтобы наилучшим образом отобразить конфигурацию на печати, для LD и FBD обычно устанавливается альбомная ориентация; для других программ ориентация – портретная. Как показано на рисунке 2.14:

Настройка печа	ти			>	<
Принтер					1
Имя:	Adobe PDF		•	Свойства	
Состояние:	Готов				
Тип:	Adobe PDF Converter				
Место:	Documents*.pdf				
Комментари	ă:				
Бумага			Ориентац	ия	1
Размер: А	4	-		Книжная	
Подача: 🛛	втовыбор	•	Α	С Альбомная	
Сеть			ОК	Отмена	

Рисунок 2.14 Настройка печати

	LEXIT	ΙС] : ВЫΙ	ити из	прог	раммы	«CKI IDO	».
--	-------	-----------	--------	------	-------	----------	----

Кроме того, в меню **File** отображаются несколько недавно открытых файлов проекта. Файлы могут быть открыты щелчком левой кнопки мыши. Как показано на рисунке 2.15:



Рисунок 2.15 Открыть последние открытые файлы

2.3.2 Меню Edit

2.3.2.1 Команды меню Edit

В состав меню *Edit* входят команды, часто применяемые при редактировании программы, такие как *Undo*, *Redo*, *Cut*, *Copy*, *Paste*, *Delete*, *Select All*, *Find*, *Replace* и *Global Find* и т.п., как показано на рисунке 2.16:

🗱 File	Edit	View	LD	Online	Load	Window	Help
		Undo		Ctrl	+Z	2 ⊆ ₩	🎭 🗖 🗖
All		Redo		Ctrl	+Y		
		Cut		Ctrl	+X	(S) (R) -	l / L:
📄 SC		Сору		Ctrl	+C	1	
<u></u>		Paste		Ctrl	+V	$(\text{Ver } 2\overline{0})$)22-10-0
-		Delete		[Del		
		Select A	AII	Ctrl	+A		
		Find		Ctrl	+F		
		Replace	e	Ctrl+	۰H		
÷ 🗎		Global	Find	Ctrl	+G		
II 📥 🔤	TEC	ידי					

Рисунок 2.16 Меню Edit

2.3.2.2 Функционал меню Edit («Редактирование»)

[Undo]: Отменить последнюю операцию. При помощи команды **Undo** можно отменить такие операции, как разместить / удалить функцию блок или функциональное поле, вставить / вырезать раздел программы, переместить функцию или функциональное поле и т.д. Изменение параметров функции или функционального поля не воспринимается программой в качестве операции, поэтому не может быть отменено.

[Redo] : Повторить последнюю операцию. Команда **Redo** используется только в отношении операции, которая была отменена командой **Undo**, любую другую операцию эта команда не отменяет.

【Cut】: Удалить выбранное содержимое и поместить его в буфер обмена, содержимое из буфера обмена можно затем вставить в другой участок. Вырезать командой *Cut* можно такое содержимое, как функцию (контакт, катушка и специальный функциональный блок) в LD и FBD, функциональное поле в SCC, инструкцию в IL или оператор в ST, а также содержимое области, выбранное с помощью операции блока.

[Сору] : Скопировать выбранное содержимое в буфер обмена, но не удалять его.

[Paste] : Вставить содержимое из буфера обмена в месте, где находится курсор.

[Delete] : Удалить выбранное содержимое, не перемещая его в буфер обмена.

Примечание: Операции *Cut*, *Copy*, *Paste* и *Delete* также можно выполнить, щелкнув по области редактирования правой кнопкой мыши. После выбора функции или функционального поля, могут быть выполнены такие операции, как *Cut*, *Copy* и *Delete*, тогда, как операция *Paste* может быть выполнена на пустом месте. Примеры приведены на рисунке 2.17:



Рисунок 2.17 Операции вырезания (Cut), копирования (Copy), вставки (Paste), и удаления (Delete)

Операции вырезания (*Cut*), копирования (*Copy*), вставки (*Paste*), и удаления (*Delete*)

[Select All] : Выбрать все содержимое в текущей области редактирования. Как показано на рисунке 2.18:



Рисунок 2.18 Операция Select all

(Find) : Найти подходящие функцию, функциональное поле, инструкцию или оператор. Операция поиска *Find* может быть выполнена только в текущей рабочей области.

Если текущая рабочая область – LD (или FBD), то поиск будет осуществлен только в текущем LD (или FBD). Например, если вы хотите найти функциональный блок с параметром "%M0036", выберите **[Edit] / [Find]**, после чего откроется диалоговое окно поиска *Find*. Введите параметр "%M0036" в строке "*Find What*" и "All" в строке "*Find Limit*", затем нажмите кнопку *Find Next*. Программа автоматически перейдет к первой подходящей функции, необходимо несколько раз нажать кнопку "*Find Next*", затем перейти к следующей в очереди функции с "%M0036", пока не будет найдено все необходимое. По завершении, «СКПро» откроет окно завершения *Finished*. Функция поиска может быть выполнена для всех функций, а также для функций указанного типа. Например, при установке значения "Нормально разомкнутый контакт" в параметре "*Find Limit*", поиск функций будет осуществляться только среди нормально разомкнутых контактов. Это показано на рисунке 2.19:

Find		>
Find What	%M 0036	•
Find Limit	All	-
🔽 Match Wh	ole Word Only	

Рисунок 2.19 Операция "Найти" в LD

После того, как найдено будет всё, «СКПро» откроет окно "Finished", как показано на рисунке 2.20:



Рисунок 2.20 Поиск окончен

Если текущая рабочая область – это SCC, то поиск будет осуществляться в текущем SCC. Функция поиска для SCC осуществляет поиск не только в операторах выполнения функциональных полей SCC, но и в описаниях. Так же, как и в LD, функция поиска может выполняться во всех типах функциональных полей, а также в функциональных полях указанного типа. Например, если изменить ограничение поиска "*Find Limit*" на *"Execution Box*", то поиск подходящих функциональных полей будет осуществляться только в полях выполнения. Это показано на рисунке 2.21:

Start	Find		×
	Find What	Alarm_1	
log("Alarm_1")	Find Limit	All	
End exit(0)	Find Next	Cancel	

Рисунок 2.21 Операция поиска в SCC

Если текущая рабочая область – IL, то поиск будет осуществляться в текущей программе IL. В диалоговом окне поиска IL (см. рисунок 2.22) есть такие опции, как "Match Whole Word Only" («Найти только полное слово») и "Match Case" («С учетом регистра»). Например, если необходимо найти "R1", но ввести "R1" в поле "Поиск" и не выбрать "С учетом регистра", то поиск выдаст значения как "R1", так и "r1".

1 2 3	GT St LD	R1 <mark>R3</mark> R4								
	Find		\times							
		Find What R3								
		Match Whole Word Only								
		🗖 Match Case								
		Find Next Cancel								

Рисунок 2.22 Операция поиска для IL

Если текущая рабочая область – ST, то поиск будет осуществляться в текущей программе ST. В диалоговом окне поиска ST (см. рисунок 2.23) есть такие опции, как "Match Whole Word Only" («Найти только полное слово») и "Match Case" («С учетом регистра»). Например, если необходимо найти "R1", но ввести "R1" в поле "Поиск" и не выбрать "С учетом регистра", то поиск выдаст значения как "R1", так и "r1".

$\frac{\mathbf{IF} \mathbf{R1} > 1}{\mathbf{R1} = 0}$	THEN Find		×
R2=2; ELSE R3:=2; END_IF;		Find What r1 Match Whole V	Yord Only
		Find Next	Cancel

Рисунок 2.23 Операция поиска для ST

[Replace] : Найти подходящий блок функции, функциональное поле, инструкцию или оператор в текущей программе и заменить их. Операция *Replace* аналогична операции *Find*, только имеет дополнительно функцию замены. Иными словами, операция *Replace* содержит в себе операцию *Find*. В рамках операции *Replace* может быть выполнена либо выборочная (частичная) замена, либо полная замена. Замена может быть выполнена только в текущей рабочей области.

[Global Find] : Глобальный поиск подходящего блока функции, функционального поля, инструкции или оператора. Эта функция выполняется во всех программах (LD, FBD, SCC, IL и ST). После завершения глобального поиска результат отображается во вкладке "**Find**" окна вывода; также там отображаются место и номер результата, как показано на рисунке 2.24:

TEST2 : (1,5) STEST : (1,4)		
2 occurrence(s) have been	n found.	
<		
Compile Debug Find	Alarm	SOE

Рисунок 2.24 Глобальный поиск

2.3.3 Меню View

2.3.3.1 Команды меню View

В меню *View* («Вид») содержатся: *System Toolbar* («Системная панель инструментов»), *FB Toolbar* («Панель инструментов FB»), *LD Toolbar* («Панель инструментов LD»), *FBD Toolbar* («Панель инструментов FBD»), *IL Toolbar* («Панель инструментов IL»), *ST Toolbar* («Панель инструментов ST»), *SCC Toolbar* («Панель инструментов SCC»), *Project Browser* («Браузер проекта»), *Output Window* («Окно вывода»), *Point Table* (Таблица точек данных), и *Status Bar* («Строка состояния») и т.п., как показано на рисунке 2.25:



Рисунок 2.25 Меню View

2.3.3.2 Функционал меню View («Вид»)

[System Toolbar] : Показать или скрыть системную панель инструментов.

[FB Toolbar] : Показать или скрыть панель инструментов функций.

(LD Toolbar) : Показать или скрыть панель инструментов LD.

[FBD Toolbar] : Показать или скрыть панель инструментов FBD.

(IL Toolbar) : Показать или скрыть панель инструментов IL.

[ST Toolbar] : Показать или скрыть панель инструментов ST.

[SCC Toolbar] : Показать или скрыть панель инструментов SCC.

[Project Browser] : Показать или скрыть браузер проекта.

[Output Window] : Показать или скрыть окно вывода.

[Point Table] : Показать или скрыть таблицу точек данных.

[Used Time Table] : Показать таблицу применения точек данных.

[Строка состояния] : Показать или скрыть строку состояния.

2.3.4 Меню LD

2.3.4.1 Команды меню LD

В меню *LD* содержатся такие команды, как: *Move* («Переместить»), *Block* («Блокировка»), *Contact* («Контакт»), *Coil* («Катушка»), *Mathematics* («Математика»), *Statistics* («Статистика»), *Logic* («Логика»), *Comparison* («Сравнение»), *Conversion* («Преобразование»), *Data Move* («Перемещение данных»), *Timer* («Таймеры»), *Counter* («Счетчики»), *Control* («Управление»), *PLC* («ПЛК»), *Others* («Прочие»), *Link* («Связать»), *Negate* («Присвоить отрицательное значение»), *Label* («Метка»), *Goto* («Перейти»), *Return* («Возврат»), *Comment* («Комментарий»), *Grid* («Сетка»), *Page Line* («Строка страницы»), *Execution Order* («Порядок исполнения»), *Point Name* («Название точки данных»), *Insert* («Вставка»), *Remove* («Удалить»), *Zoom* («Масштаб»), *Property* («Свойства») и *Debug* («Отладка»), и т.д., как показано на рисунке 2.26:



Рисунок 2.26 Меню LD

2.3.4.2 <u>Функционал меню LD</u>

(Move) : Если в меню *LD* не выбрана функция, то область редактирования по умолчанию находится в состоянии перемещения. Выбранную в данный момент позицию функции можно изменить движением мыши. Для перемещения функции, наведите на неё курсор мыши и, удерживая левую кнопку, переместите её в указанное место, затем отпустите кнопку.

[Block] : Операция **Block** используется для выбора всех функций и связующих элементов в одной области. Используйте мышь, чтобы выделить нужную область, все элементы в этой области будут выбраны. Операции перемещения, вырезания, копирования, удаления и другие для нескольких элементов должны быть завершены операцией **Block**.

[Link] : Операция *Link* используется для создания связи между двумя точкамиконнекторами двух функций. Наведите курсор мыши на первую точку-коннектор, которую необходимо связать, и щелкните левой кнопкой, чтобы выбрать её; затем наведите курсор мыши на вторую точку-коннектор и щелкните левой кнопкой на ней. Таким образом, между двумя точками-коннекторами функций возникает связь. Если точки-коннекторы не соответствуют принципу связи, то появится диалоговое окно с сообщением об этом, как показано на рисунке 2.27:

SCPro		×
	Type mismatch, please do again!	
	ОК	

Рисунок 2.27 Ошибка связи

[Negate] : Когда выбранная функция является контактом, с помощью операции *Negate* можно циклически переключаться между нормально разомкнутым контактом, нормально замкнутым контактом, контактом с положительным переходом, контактом с отрицательным переходом.

Когда выбранная функция является катушкой, с помощью операции **Negate** можно циклически переключаться между катушкой, инверсной катушкой, катушкой положительного перехода, катушкой отрицательного перехода, катушкой установки и катушкой сброса.

[Grid] : Если для редактирования требуются визуальные направляющие, на фоне области редактирования LD может быть установлена точечная сетка.

[Page Line] : Для удобства навигации на фоне области редактирования LD могут отображаться строка и номер страницы.

[Execution Order] : Выделение порядка выполнения всех функций в состоянии сканирования, как показано на рис.Рисунок 2.28:

[Point Name] : Выделение названия точек данных всех функций. Каждой точке данных может быть присвоено имя в таблице точек; при отображении названия точки данных, в нижней части функции также будет отображаться присвоенное имя, как показано на рисунке 2.28:



Рисунок 2.28 Порядок выполнения и название точки

[Insert] : Выберите место для вставки строки, щелкните левой кнопкой мыши в этом месте и выберите команду *Insert*, чтобы добавить одну строку. В основном эта команда используется, когда во время редактирования программы LD необходимо добавить программный сегмент между существующими сегментами.

(Remove) : Выберите строку, которую необходимо удалить, щелкните левой кнопкой мыши в этом месте и выберите команду **Remove**, чтобы удалить одну строку. В основном эта команда используется, когда во время редактирования программы LD необходимо удалить пустой сегмент программы, а на месте удаленной строки не должно быть никаких функциональных блоков или связок.

(Zoom) : Операция масштабирования Zoom может использоваться для изменения масштаба отображения области редактирования LD. После выбора команды **Zoom**, «СКПро» откроет диалоговое окно **Zoom**. Можно ввести в поле ввода желаемое значение масштабирования, нажать кнопки вверх и вниз рядом с полем ввода или перетащить курсор для изменения масштаба; заданный масштаб всегда отображается в поле ввода, как показано на рисунке 2.29:

Zoom		\times
	Zoom Ratio 100	· ÷ %
	ок	Cancel

Рисунок 2.29 Масштабирование с командой Zoom

[Property] : Отобразить свойства выбранного объекта.

Для функции LD отображаются такие свойства, как тип, название программы, порядок выполнения, положение, определение параметров ввода/вывода и т.д. Тип функции отображается в верхней части диалогового окна. В примере далее контакт отображается следующим образом: название программы LD, в которой находится этот функция, "FPQ2"; Порядок выполнения 1-ый; Позиция находится в первом столбце и 4-ой строке; Параметр "Display EN/ENO" в редакторе FBD может использоваться в качестве альтернативы; Параметр "Input Number" («Номер ввода») может быть изменен в некоторых функциональных блоках LD и FBD; Параметр каждой точкиконнектора указан последовательно, как показано на рис.2.30:

FB Prop	oerty			×
		- - Normally	open contact	
Pro	gram Name	FPQ2	Start Position	[1,4]
Inp	ut Number	1 -	Execution Order	1
Inst	tance Name		▼ Ø Dis	play EN/ENO
	No.	Input	Output	^
	1	T	B1	
	2	\$∎0001		
	3			
	4			
	5			
	6			
	7			
	 Q			-
		ок	Cancel	

Рисунок 2.30 Свойство функционального блока

[Debug] : Отладка программы LD; включает команды *Step* («Шаг»), *Continue* («Продолжение»), *Insert/Remove Breakpoint* («Вставить/удалить точку останова») и *Remove All Breakpoints* («Удалить все точки останова»), и т.п.

2.3.5 SCC

2.3.5.1 Команды меню SCC

В меню SCC содержатся такие команды, как: Move («Переместить»), Block («Блок»), Link («Связать»), Start Box («Начальное поле»), End Box («Конечное поле»), Execution Box («Поле выполнения»), Condition Box («Поле условий»), Time-limited Condition Box («Поле условий с ограничением по времени»), Connector 1 («Коннектор 1»), Connector 2 («Коннектор 2»), Comment («Комментарий»), Page Line («Строка страницы»), Execution Statement («Исполнение оператора»), Property («Свойство»), Debug («Отладка») и т.д., как показано на рис.2.31:

🐹 File	Edit	View	SCC	Online	Load	Window	Help							- 8 ×
	8	ù Ø	~	Move				2	T 001	VAR DEB IN	3 👂 🖴	• 🏜 🖴	2	🖶 🏠 🛛
Conve	rtion	-		Block										
N 🗆	Q	⇒‡		Link				Ļ	×B		h. 🌒	<u>ک</u> ()		
📄 S	CPLC			Start Box				1	0-1		N3)			^
📄 🖶 🚥 🧰	400	TEST	ſ	End Box				ľ	× 🗨	Start				
	TES	Т		Execution	Box									
1 9	···· 🗾	Progr		Condition	Box					1				
	<u> </u>			Time-limi	ted Co	ndition Bo	x							
				Connecto	r 1					_				
		·····		Connecto	r 2			1		-				
		llcer		Commen	t				N		0			
	····		~	Page Line										
÷	TES	Т2	~	Execution	Staten	nent			, Oms	TY				
				Droporty										~
-	D	- 4 -		SVG Eile										>
Progr	am U	ลเล		3001116				-1						
	Number	1		Debug				>		Used	Times	V	/alue	
	%10001													
	%10002 %10003													
1	%10004													_
	10005													
1	Q		w	QW	м	MM	V I	N	NW	S	SW	Т	С	
<														>
	_													
Comp	ile Do	ebug	Find	Alarr	n SC	DE								
									-1.0-11	00/				. استناس

Рисунок 2.31 Меню SCC

2.3.5.2 Функционал меню SCC

(Move) : Если в меню *SCC* не выбрано функциональное поле, то область редактирования по умолчанию находится в состоянии перемещения. Выбранную в данный момент позицию функционального поля или связующего элемента можно изменить движением мыши. Для перемещения функционального поля или связи, наведите на неё курсор мыши и, удерживая левую кнопку, переместите её в указанное место, затем отпустите кнопку.

(Block) : Операция **Block** используется для выбора всех функциональных полей и связующих элементов в одной области. Используйте мышь, чтобы выделить нужную область, все элементы в этой области будут выбраны. Операции перемещения, вырезания, копирования, удаления и другие для нескольких элементов должны быть завершены операцией **Block**.

[Link] : Операция *Link* используется для создания связи между двумя точкамиконнекторами двух функциональных полей. Наведите курсор мыши на первую точкуконнектор, которую необходимо связать, и щелкните левой кнопкой, чтобы выбрать её; затем наведите курсор мыши на вторую точку-коннектор и щелкните левой кнопкой на ней. Таким образом, между двумя точками-коннекторами функций возникает связь. Если точки-коннекторы не соответствуют принципу связи, то появится диалоговое окно с сообщением об этом, как показано на рис.2.32.

SCPro	>	<
	Type mismatch, please do again!	
	ОК	

Рисунок 2.32 Ошибка связи SCC

[Page Line] : Для удобства навигации на фоне области редактирования SCC могут отображаться строка и номер страницы.

[Execution Statement] : Показать или скрыть исполнение оператора SCC. Если вам необходимо ознакомиться лишь с процессом выполнения SCC, отображать исполнение оператора не обязательно, достаточно лишь прочитать описание функционального поля SCC. Если необходимо узнать подробности выполнения определенного шага последовательного управления, необходимо отобразить выполнение оператора. Символ √ перед данным элементом указывает на то, что исполнение оператора находится в состоянии "показать", в противном случае - в состоянии "скрыть". При отображении исполнения оператора, в каждом функциональном поле отображается серийный номер, который указывает порядок размещения функционального поля вместо порядка выполнения, как в LD (FBD). Пример показан на рисунке 2.33



Рисунок 2.33 Оператор выполнения

[Свойство] : Отобразить свойства выбранного объекта.

Свойства функционального поля SCC включают в себя тип, имя SCC, серийный номер, положение, описание, оператор выполнения, название операции и относительные параметры. Тип функционального поля отображается в верхней части диалогового окна. Окно выполнения отображается следующим образом: имя SCC — SSS, серийный номер - 1, позиция - (163,75)-(303,125), операция выполнения - это выражение. Отображаются только действующие параметры, так как параметры функциональных полей отличаются в зависимости от их типа. Как показано на рисунке 2.34:

SCC Property			×
	Exec	ution Bo×	
SCC Name	sc	Serial Number 1	
Position	(163,75) — (3	03,125)	
Description			
Statement			
Operation Name	Expression		
Parameter 1			
Parameter 2			
Parameter 3			
		ОК	

Рисунок 2.34 Свойство SCC

[Debug]: Отладка программы SCC; включает в себя следующие функции: Automatic («Автоматический»), Watching («Просмотр»), Debugging («Отладка»), Stop («Останов»), Lock («Блокировка»), Unlock («Разблокировка»), Restart («Перезапуск»), Step, («Шаг»), Continue («Продолжение»), Stop Debugging («Остановка отладки»), Insert/Remove Breakpoint («Вставка/Удаление точки останова»), и Remove All Breakpoints («Удаление всех точек останова») и т.д.

2.3.6 Online

2.3.6.1 Команды меню Online

В меню Online содержатся такие команды, как: Connect/PLC («Подключение/ПЛК»), Connect/Simulator («Подключение/Симулятор»), Disconnect («Отключение»), Display Format («Формат отображения»), Refresh Program («Обновление программы»), Unforce («Отмена принудительного ввода»), Reset («Сброс»), Set Time («Установить время»), Master/Slave Switch («Переключатель Master/Slave») и т.д., как показано на рисунке 2.35:

🗱 File Edit View SCC	Online Load Window Help – 🗗	×
D 🖬 🗈 🗞 🎒 🍪	Connect > PLC Ctrl+W 🖳 🔮 🖊 🕇	ì C
Convertion VINT_1	Disconnect Ctrl+Q Simulator Ctrl+U	
▶□ ♀ ᄒ ݨ ◊ ♥	Display Format > 🛃 🗃 🗊 🕀 🗈 🖑 🕱	
SCPLC ⊕ 400_TEST1 ⊕ 1 FST	Refresh Program Ctrl+M Unforce	^
Program	Reset Set Time Master/Slave Switch	
Licer Define	Backup Data > Refresh Data > 0	

Рисунок 2.35 Меню Online

2.3.6.2 <u>Функционал меню Online</u>

[Connect] / **[PLC]** : Подключить текущий компьютер для отладки к ПЛК. Перед подключением нужно обязательно убедиться, что сеть физически доступна. В противном случае «СКПро» **откроет** диалоговое окно с ошибкой, как показано на рисунке 2.36:

SCPro		×
Â	PLC is nonexistent, cannot connect!	
	ОК	

Рисунок 2.36 Сбой соединения с ПЛК

Условия необходимые для подключения к ПЛК

IP-адрес компьютера для отладки должен находиться в том же адресном пространстве, что и IP-адрес ПЛК, то есть, первые три сегмента IP-адреса должны совпадать. В противном случае соединение установить не удастся. Например: если IP-адрес ПЛК - 192.168.1.100, то IP-адрес компьютера для отладки должен начинаться с 192.168.1.***.

«СКПро» автоматически выполнит поиск ПЛК в сети в соответствии с IP-адресом Ethernet модуля ЦПУ в конфигурации ПЛК. Настраивать его заново необходимости нет.

Указание

ПЛК должен быть сконфигурирован согласно руководству по первому подключению.

Конфигурация после успешного подключения

После успешного подключения, цвет фона области редактирования программы становится бледно-лиловым. Информация о данных в ПЛК передается в «СКПро» по сети; параметры со значением 1 и проводящие связи отображаются красным цветом, а параметры со значением 0 и непроводящие связи отображаются зеленым цветом, как показано на рисунке 2.37:


Рисунок 2.37 LD в состоянии онлайн

[Connect] / **[Simulator]** : Функция имитирует рабочее состояние ПЛК, принудительно вводит фактические значения точек и переменных, а также отлаживает программы.

[Disconnect] : Разрыв соединения компьютера для отладки с ПЛК или симулятором.

(Display Format) : Функция в режиме реального времени указывает форматы отображения целочисленных данных (не для значений типа BOOL). Есть три формата отображения: десятичный, шестнадцатеричный и двоичный; выбранный формат обозначается символом √. Например, если текущий регистр "%МW0001" - десятичный 2002, то в таблице точек три вида форматов отображения будут показаны следующим образом. Все приведенные ниже значения представляют "2002" в различных форматах отображения, за шестнадцатеричными и двоичными данными соответственно следуют буквы "Н" и "В", которые представляют текущий формат отображения, как показано на рисунке 2.38:



Рисунок 2.38 Формат отображения

[Refresh Program] : Функция обеспечивает удобную отладку для изменения программ без необходимости отключения ПЛК. Если при включенном ПЛК удалить, переместить функцию или изменить параметр функции, справа от названия программы в браузере проекта появится метка "*"; например, "MAIN*". После внесения подобных изменений, необходимо выбрать команду **Refresh Program**, чтобы загрузить изменения в ПЛК. После внесения изменений и перед выходом из «СКПро», сохраните измененные программы, в противном случае это приведет к несогласованности программы. Перед обновлением, «СКПро» автоматически скомпилирует программу. Если возникнет какая-либо ошибка, то программа не обновится, а место ошибки будет указано. После обновления, ПЛК будет запущен с учетом новых параметров загруженных программ. Перезапускать систему не нужно. Поскольку команда **Refresh Program** изменяет только исполняемую программу, не изменяя исходную программу, сохраненную в ПЛК, программа, загруженная в этот момент, не обновится, поэтому рекомендуется сохранять программы после внесения изменений и выполнять полную загрузку после того, как программа определена в финальном виде.

(Unforce) : При включенном ПЛК в таблице точек данных есть команда *Force* для принудительного ввода. После запуска функции *Force* отсканированные состояния сигналов дискретных и аналоговых входов/выходов не будут отправлены в соответствующие области памяти, они могут быть установлены в соответствии с требованиями отладки без учета их фактического состояния. Команда *Unforce* отменяет принудительный ввод точек и возвращается к стандартному сканированию.

[Reset] : Сброс модуля центрального процессора ПЛК по сети для перезагрузки. При работе с резервированной системой, команда одновременно делает сброс обоих модулей ЦП. Если модуль ЦП ПЛК, подлежащий сбросу, не находится в одной сети с инженерной станцией, то «СКПро» выдаст сигнал *"Reset failure"* («Сбой сброса»).

[Set Time] : Установка времени для ПЛК по сети в режиме online. При работе с резервированной системой, команда одновременно устанавливает время для обоих модулей процессора. Однако устанавливаемое время является временем инженерной станции, а не стандартным временем.

[Master/Slave Switch] : Команда переключения на резервный ЦПУ действительна только для резервированных систем. В системах, оснащённых двумя ЦПУ один выполняет роль ведущего (master), а второй – ведомого (slave). Команда **Master/Slave Switch**, переключает текущий ведущий ЦПУ в состояние ведомого, а текущий ведомый – в состояние ведущего.

2.3.7 Меню Download

См. главу 3. «Управление проектами».

2.3.8 Меню Window

2.3.8.1 Команды меню Window («Окно»)

В меню *Window* содержатся такие команды, как *Close* («Закрыть»), *Close All* («Закрыть все»"), *Cascade* («Окна каскадом»), *Tile Horizontally* («Окна плиткой по горизонтали») и *Tile Vertically* («Окна плиткой по вертикали») и т.д., как показано на рисунке 2.39:



Рисунок 2.39 Меню Window

2.3.9 Меню Help («Справка»)

2.3.9.1 Команды меню Help («Справка»)

В меню *Help* («Справка») содержатся такие команды, как: *Contents* («Содержимое»), *Index* («Индекс»), *Search* («Поиск»), *About...* («О «СКПро») и т.д. Внешний вид приведён на рисунке 2.40:



Рисунок 2.40 Меню Help («Справка»)

2.3.9.2 Функционал меню Help («Справка»)

[Contents] : Отобразить вкладку *Contents* («Содержимое») из меню «Справка».

[Index] : Отобразить вкладку *Index* («Индекс») из меню «Справка».

[Search] : Отобразить вкладку Search («Поиск») из меню «Справка».

[About ...] Отобразить версию и авторские права на программное обеспечение. Как показано на рисунке – 2.41:

About SCPro		×
	SCPro V6.2.11 PLC Programming Software Copyright(C) 2021-2022 Joint Stock Company "SybCom"	ОК

Рисунок 2.41 Информация о версии «СКПро»

2.4 Системная панель инструментов

Для удобства, системная панель инструментов может размещать в верхней части области редактирования часто применяемые функции операций и процессов редактирования в виде иконок. Все функции могут быть запущены с помощью операций меню, поэтому здесь приведены соответствующие операции меню. Подробная информация о функциях приведена в описании команд меню.

- . New, соответствует команде меню [File] / [New].
- 🐸: Open, соответствует команде меню 【File】/【Open】.
- 🔜 : Save, соответствует команде меню 【File】/【Save】.
- 🗈 : New Program, соответствует команде меню 【File】 / 【New Program】.
- 🔊: Save Program, соответствует команде меню 【File】 / 【Save Program】.
- 🖾 : Save All Programs, соответствует команде меню 【File】 / 【Save All Programs】.
- 🖄: Compile Program, соответствует команде меню 【File】/【Compile Program】.
- 👗 : Cut, соответствует команде меню 【Edit】 / 【Cut】.
- 🖻: Сору, соответствует команде меню [Edit] / [Сору].
- 💼: Paste, соответствует команде меню 【Edit】 / 【Paste】.
- 🖴: Undo, соответствует команде меню 【Edit】 / 【Undo】.
- 🖴: Redo, соответствует команде меню [Edit] / [Redo] .
- 🐴: Find, соответствует команде меню 【Edit】 / 【Find】.
- 🍽 : Global Find, соответствует команде меню 【Edit】 / 【Global Find】 .
- 🖳: Project Browser, соответствует команде меню 【View】 / 【Project Browser】.
- 🔊: Output Window, соответствует команде меню 【View】/【Output Window】.

🕮: Point Table, соответствует команде меню 【View】 / 【Point Table】.

Login, соответствует команде меню [Online] / [Login].

🚔 : PLC Connect, соответствует команде меню 【Online】 / 【Connect】 / 【PLC】.

🚔 : Disconnect, соответствует команде меню 【Online】 / 【Disconnect】 .

Simulator Connect, соответствует команде меню [Online] / [Connect] / [Simulator].

🔁 : Refresh Program, соответствует команде меню 【Online】/【Refresh Program】.

🗣: Download All, соответствует команде меню 【Load】/【Download All】.

🖸: Download Project, соответствует команде меню 【Load】/【Download Project】.

🗳: Upload Project, соответствует команде меню 【Load】 / 【Upload Project】.

🗈: Download Program, соответствует команде меню 【Load】 / 【Download Program】

🖺: Upload Program, соответствует команде меню 【Load】 /【Upload Program】.

🗁: Print, соответствует команде меню 【File】/【Print】.

😰 : About, соответствует команде меню 【Help】 /【About SCPro】.

2.5 Панель инструментов LD

Для удобства редактирования программ LD, панель инструментов LD может размещать в верхней части меню **LD** операции и функциональные блоки в виде иконок.

📐 : Move, Быстрая клавиша 【LD】 /【Move】.

🛄: Block, Быстрая клавиша 【LD】 / 【Block】.

 Image: Normally Open Contact, соответствует команде меню [LD] / [Normally Open Contact].

 Contact].
 Быстрая клавиша: F2

III: Normally Close Contact, соответствует команде меню [LD] / [Normally Close Contact]. Быстрая клавиша: F3

IPI:Positive Transition-sensing Contact, соответствует команде меню [LD] / [Positive Transition-sensing Contact]. Быстрая клавиша: F4

 Image: Negative Transition-sensing Contact, соответствует команде меню [LD] / [

 Negative Transition-sensing Contact].
 Быстрая клавиша: F5

(): Coil, соответствует команде меню [LD] / [Coil]. Быстрая клавиша: F6

W: Negated Coil, соответствует команде меню 【LD】 / 【Negated Coil】. Быстрая клавиша: F7

 Positive Transition-sensing Coil, соответствует команде меню [LD] / [Positive Transition-sensing Coil].

 Быстрая клавиша: F8

We represent the set of the set of

(S): Set Coil, соответствует команде меню 【LD】/【Set Coil】. Быстрая клавиша: F10

🛞: Reset Coil, соответствует команде меню 【LD】 / 【Reset Coil】. Быстрая клавиша: F11

¹: Link, соответствует команде меню [LD] / [Link].

: Negate, соответствует команде меню [LD] / [Negate].

L: : Label, соответствует команде меню [LD] / [Label].

≫: Goto, соответствует команде меню 【LD】 / 【Goto】.

Return, соответствует команде меню 【LD】 / 【Return】.

🖪: Comment, соответствует команде меню 【LD】 / 【Comment】.

: Zoom, соответствует команде меню [LD] / [Zoom].

Insert, соответствует команде меню [LD] / [Insert].

🟁: Remove, соответствует команде меню 【LD】 / 【Remove】.

🖹: Property, соответствует команде меню 【LD】 / 【Property】.

🔁: Step, соответствует команде меню 【LD】 / 【Debug】 / 【Step】.

🗈 : Continue, соответствует команде меню 【LD】 / 【Debug】 / 【Continue】.

Insert/Remove Breakpoint, соответствует команде меню [LD] / [Debug] / [Insert/Remove Breakpoint].

1: Clear All Breakpoints, соответствует команде меню [LD] / [Debug] / [Clear All Breakpoints].

Существует большое количество типов базовых функций, потому необходимо выбирать одну функцию с помощью двух выпадающих списков. Первый выпадающий список используется для выбора группы, а второй - для выбора указанной функции. Например, текущая группа отображения – *Move*, что означает, что функции группы *Move* будут приведены во втором выпадающем списке. Выбрать базовую функцию можно, нажав на обозначающую её иконку. Таким образом, базовая функция будет размещена непосредственно в области редактирования, как показано на рис.2.42:

Mo	ve	-	MOVE	•	₽
			MOVE	^	
		6	BLKMOV BLKCLR ETHMOV	=	
			COMMOV READ WRITE	~	

Рисунок 2.42 Базовый функциональный блок

2.6 Панель инструментов FBD

Для удобства редактирования программ FBD, панель инструментов FBD может размещать в верхней части меню FBD операции и функциональные блоки в виде иконок.

- 📐 : Move, соответствует команде меню 【FBD】 / 【Move】.
- 🛄: Block, соответствует команде меню 【FBD】 / 【Block】.
- ¹: Link, соответствует команде меню **[FBD] / [Link]**.
- 🖻: Negate, соответствует команде меню 【FBD】 / 【Negate】.
- L:: Label, соответствует команде меню [FBD] / [Label].
- → : Goto, соответствует команде меню 【FBD】 / 【Goto】.
- Return, соответствует команде меню [FBD] / [Return].
- 🖪 : Comment, соответствует команде меню 【FBD】 / 【Comment】.
- I Zoom, соответствует команде меню [FBD] / [Zoom].
- Insert, соответствует команде меню [FBD] / [Insert].
- 🟁: Remove, соответствует команде меню 【FBD】 / 【Remove】.
- 🖆: Property, соответствует команде меню 【FBD】 / 【Property】.
- 🔁: Step, соответствует команде меню [FBD] / [Debug] / [Step] .
- 🗈 : Continue, соответствует команде меню 【FBD】 / 【Debug】 / 【Continue】.
- 🖑 : Insert/Remove Breakpoint, соответствует команде меню 【FBD】 / 【Debug】 /

[Insert/Remove Breakpoint] .

繁: Clear All Breakpoints, соответствует команде меню **[FBD] / [Debug] / [Clear All** Breakpoints].

2.7 Панель инструментов IL

Для удобства редактирования программ IL, панель инструментов IL может размещать в верхней части меню **IL** операции и функциональные блоки в виде иконок.

LDN :LDN инструкция, соответствует команде меню 【IL】 / 【Load】 / 【LDN】.

ят :ST инструкция, соответствует команде меню 【IL】 / 【Store】 / 【ST】.

STN :STN инструкция, соответствует команде меню **[IL] / [Store] / [STN]**.

s :S инструкция, соответствует команде меню 【IL】 / 【Store】 / 【S】.

R : R инструкция, соответствует команде меню [IL] / [Store] / [R].

AND АND инструкция, соответствует команде меню [IL] / [Logic] / [AND] .

ок : OR инструкция, соответствует команде меню [IL] / [Logic] / [OR].

хок XOR инструкция, соответствует команде меню [IL] / [Logic] / [XOR].

ст : GT инструкция, соответствует команде меню [IL] / [Comparison] / [GT].

GE инструкция, соответствует команде меню [IL] / [Comparison] / [GE] .

LT :LT инструкция, соответствует команде меню 【IL】 / 【Comparison】 / 【LT】.

LE :LE инструкция, соответствует команде меню 【IL】 / 【Comparison】 / 【LE】.

EQ :EQ инструкция, соответствует команде меню [IL] / [Comparison] / [EQ] .

№ :NE инструкция, соответствует команде меню 【IL】 / 【Comparison】 / 【NE】.

JMP инструкция, соответствует команде меню [IL] / [Jump] / [JMP].

САL САL инструкция, соответствует команде меню [IL] / [Call] / [CAL].

RET : **RET** инструкция, соответствует команде меню **[IL] / [Call] / [RET]**.

🔂 : Step, соответствует команде меню 【IL】 / 【Debug】 / 【Step】.

🗈 : Continue, соответствует команде меню 【IL】 / 【Debug】 / 【Continue】.

Insert/Remove Breakpoint, соответствует команде меню [IL] / [Debug] / [Insert/Remove Breakpoint].

ื Clear All Breakpoints, соответствует команде меню 【IL】 / 【Debug】 / 【Clear All **Breakpoints** .

2.8 Панель инструментов ST

Для удобства редактирования программ ST, панель инструментов ST может размещать в верхней части меню **ST** операции и функциональные блоки в виде иконок.

Assignment, соответствует команде меню [ST] / [Operator] / [Assignment]

IF ·IF оператор, соответствует команде меню [ST] / [IF].

CASE CASE оператор, соответствует команде меню [ST] / [CASE].

FOR .FOR OREPATOP, COOTBETCTBYET KOMAHAE MEHIO [ST] / [FOR] .

WHILE •WHILE оператор, соответствует команде меню [ST] / [WHILE] .

REPEAT . **REPEAT** оператор, соответствует команде меню **[ST] / [REPEAT]**.

EXIT • EXIT оператор, соответствует команде меню [ST] / [EXIT].

RETURN · **RETURN** OREPATOR, COOTBETCTBYET КОМАНДЕ МЕНЮ **(ST) / (RETURN)**.

- 🚯 : Step, соответствует команде меню [ST] / [Debug] / [Step] .
- **E** : Continue, соответствует команде меню **[ST] / [Debug] / [Continue]**.

🖑 : Insert/Remove Breakpoint, соответствует команде меню [ST] / [Debug] /

[Insert/Remove Breakpoint] .

🕺 : Clear All Breakpoints, соответствует команде меню 【ST】/【Debug】/【Clear All Breakpoints].

2.9 Панель инструментов SCC

Для удобства редактирования программ SCC, панель инструментов SCC может размещать в верхней части меню SCC операции и функциональные блоки в виде иконок.



I: Move, соответствует команде меню [SCC] / [Move].



. Block, соответствует команде меню [SCC] / [Block].

🖵 : Start Box, соответствует команде меню 【SCC】 / 【Start Box】.

👛 : End Box, соответствует команде меню 【SCC】 / 【End Box】.

🖵 : Execution Box, соответствует команде меню 【SCC】 / 【Execution Box】.

💠 : Condition Box, соответствует команде меню 【SCC】 / 【Condition Box】.

 : Time-limited Condition Box, соответствует команде меню [SCC] / [Time-limited

 Condition Box].

- Connector 1, соответствует команде меню [SCC] / [Connector 1].
- 오 : Connector 2, соответствует команде меню [SCC] / [Connector 2] .
- —: Link, соответствует команде меню [SCC] / [Link].
- 🗅 : Comment, соответствует команде меню [SCC] / [Comment] .
- **!** : Automatic, соответствует команде меню [SCC] / [Debug] / [Automatic].
- м!: Watching, соответствует команде меню [SCC] / [Debug] / [Watching] .
- 💷 : Debugging, соответствует команде меню [SCC] / [Debug] / [Debugging] .
- 🗙 : Stop, соответствует команде меню 【SCC】 / 【Debug】 / 【Stop】.
- 園 : Lock, соответствует команде меню 【SCC】 / 【Debug】 / 【Lock】.
- 🛅 : Unlock, соответствует команде меню [SCC] / [Debug] / [Unlock] .
- 🗊 : Restart, соответствует команде меню [SCC] / [Debug] / [Restart] .
- 🕀 : Step, соответствует команде меню [SCC] / [Debug] / [Step] .
- En : Continue, соответствует команде меню [SCC] / [Debug] / [Continue] .
- Stop Debugging, соответствует команде меню [SCC] / [Debug] / [Stop Debugging].
- 😬 : Insert/Remove Breakpoint, соответствует команде меню 【SCC】 / 【Debug】 /

[Insert/Remove Breakpoint] .

🗏 : Clear All Breakpoints, соответствует команде меню 【SCC】 / 【Debug】 / 【Clear All Breakpoints】.

2.10 Окно вывода

Для удобства компиляции, отладки программы и отслеживания статуса, в окне вывода отображаются результаты компиляции, статус отладки, результаты поиска, тревоги, события SOE и т.д.

[Compile] : Это окно вывода результатов компиляции программ.

Если дважды щелкнуть на сообщение об ошибке в этом окне, то система перейдет к соответствующему местоположению ошибки. Окно показано на рис.2.43:

MAIN : - - (2,2) input expected MAIN : - - (2,2) output expected TEST_P : (1,1) syntax error SC : start box expected					
4 error(s)					
Compile Debug Find Alarm SOE					

Рисунок 2.43 Окно вывода

[Debug] : Это окно вывода сообщений процесса отладки. В режиме онлайн-отладки, ошибки, возникающие в процессе выполнения программы, будут выводиться в этом окне.

【Find】: Это окно вывода результатов глобального поиска с помощью команды *Global Find*. При глобальном поиске с помощью функции 【Edit】 / 【Global Find】 или кнопки На панели инструментов, местоположение и общее количество найденного содержимого будут отображаться в этом окне, как показано на рис.2.44:

TEST2 : (1,5) STEST : (1,4)	
2 occurrence(s) have be	en found.
<	
Compile Debug Find	Alarm SOE

Рисунок 2.44 Результаты глобального поиска

[Alarm] : Это окно вывода сообщений о тревогах, возникающих в программе. В этом окне, в поле *Execute Box/Alarm* будут отображаться сообщение о тревогах, при редактировании программ с помощью редактора SCC в процессе выполнения программы, а также сообщения о состоянии запуска/завершения программы, как показано на рис.2.45:

Execution Box	×
✓ Description	Copy Statement
Statement log("Alarm")	Property
Use Basic Operation OK 06 – Alarm	Cancel
0001 2022-10-18 11:43:55 SCC SCJ-J-Startup (in debug 0002 2022-10-18 11:44:08 SCC SCJ-J-Alarm 0003 2022-10-18 11:44:09 SCC SCJ-J-Exit < Compile Debug Find Alarm SOE	ing mode)
Complie Debug Find Alarm SUE	

Рисунок 2.45 Сообщение о тревоге

(SOE) : Это окно вывода сообщений о событиях SOE.

Если ПЛК, оснащен модулем последовательности событий (SOE), в этом окне будут выводиться все сообщения о событиях SOE, включая номер точки данных, изменение 0->1 бит или 1->0 бит, сообщения о времени и т.д., как показано на рисунок 2.46:

	0001	%10017	0->1	2008-10-	08 09:33	3:41:504.0)
	0002	%10018	0->1	2008-10-	08 09:33	3:41:504.0) 🚽
	0003	%10019	0->1	2008-10-	08 09:33	3:41:504.0)
	0004	%10020	0->1	2008-10-	08 09:33	3:41:504.0)
	0005	%10021	0->1	2008-10-	08 09:33	3:41:504.0)
	0006	%10022	0->1	2008-10-	08 09:33	3:41:504.0)
	0007	%10023	0->1	2008-10-	08 09:33	3:41:504.0) 🚽
	•						<u>></u>
1							
Ĩ	Com	nile De	hua	Find	Alarm	SOF	
	~~		oug	- ma	Alum	JUL	

Рисунок 2.46 Сообщение о событии SOE

2.11 Справочник по горячим клавишам

	-		-
	Новый проект	Ctrl + N	
	Новая программа	Ctrl + E	
File	Сохранить проект	Ctrl + S	
	Сохранить программу	Ctrl + l	
	Скомпилировать программы	все	Ctrl + B

	Скомпилировать	Ctrl + R	
	Программу		
	Открыть проект	Ctrl + O	
	Печать	Ctrl + P	
	Отменить	Ctrl + Z	
	Повторить	Ctrl + Y	
	Вырезать	Ctrl + X	
	Копировать	Ctrl + C	
	Вставить	Ctrl + V	
	Удалить	Del	
Edit	Выбрать все	Ctrl + A	
	Поиск	Ctrl + F	
	Заменить	Ctrl + H	
	Глобальный поиск	Ctrl + G	
	Вставка	Ctrl + J	
	Удалить	Ctrl + K	
	Масштабировать	Ctrl + D	
	Подключение / ПЛК	Ctrl + W	
Online	Подключение / Симулятор	Ctrl + U	
	Отключение	Ctrl + Q	
	Загрузить все	Ctrl + L	
Load	Загрузить программу	Ctrl + T	
	Обновить программу	Ctrl + M	
	Шаг	Ctrl+F9	
	Продолжить	Ctrl+F10	
Debug	Вставка/Удаление точки останова	Ctrl+F11	
	Удаление всех точек останова	Ctrl+F12	

	Окно программы	Alt+0		
View	Браузер проекта	Alt+1		
VIEW	Окно вывода	Alt+2		
	Таблица точек данных	Alt+3		
	Нормально разомкнутый контакт	F2		
	Нормально замкнутый контакт	F3		
	Контакт с положительным переходом	F4		
	Контакт с отрицательным переходом	F5		
	Катушка	F6		
	Инверсная катушка	F7		
	Катушка положительного перехода	F8		
Program	Катушка отрицательного перехода	F9		
	Катушка установки	F10		
	Катушка сброса	F11		
	Присвоить отрицательное значение	Alt + F1		
	Метка (Label)	Alt + F2		
	Переход (Goto)	Alt + F3		
	Возврат (Return)	Alt + F4		
	Комментарий (Comment)	Alt + F5		
	Блок функции (Function block (FB))	First letter of FB		
Link	Левая вертикальная связь (Left vertical link)	Alt + F10		
	Правая вертикальная связь (Right vertical link)	Alt + F11		

	Горизонтальная связь (Horizontal link)	Alt + F12	
	Вверх	↑	
	Вниз	\downarrow	
	Влево	<i>←</i>	
	Вправо	\rightarrow	
Перемещени	Страница вверх	Page Up	
е курсора	Страница вниз	Page Down	
	Первая колонка	Home	
	Последняя колонка	End	
	Первая строка	Ctrl + Home	
	Последняя строка	Ctrl + End	
	Вверх	Ctrl +↑	
Переместить	Вниз	Ctrl +↓	
FB	Влево	Ctrl +←	
	Вправо	Ctrl +→	
	Помощь	F1	
Прочее	Свойства FB	Shift + Enter	
	Перейти в программу	Ctrl + Enter	

3 Работа с проектами

После входа в среду разработки «СКПро», основной задачей пользователя становится преобразование имеющегося у него проекта в код, который может быть исполнен в ПЛК. В среде разработки проект управляет всеми элементами, такими как программы, данные, ресурсы и т.д. С технической точки зрения, управление проектом осуществляется с помощью файла конфигурации проекта и файлов программ проекта. Файл конфигурации проекта отвечает за конфигурацию оборудования, управление проекта определяют программный набор процессов, которые должны быть выполнены, такие, как программа LD, программа FBD, программа IL, программа ST и программа SCC и т.д.

3.1 Браузер проекта

Используя браузер проекта, можно отобразить содержимое проекта «СКПро», а также переключаться между составляющими элементами проекта, такими, как программа, данные и ресурсы.

Браузер проекта отображается в виде дерева содержимого и предоставляет доступ к следующим командам:

- Program---LD, FBD, IL, ST, и SCC и т.д.;
- Data---Point Table, Variable Table, Optional Point Table;
- Resource---PLC Configuration, Task Configuration, Interrupt Configuration и т.д., как показано на рисунке 3.1:



Рисунок 3.1 Браузер проекта

По умолчанию программы отображается на третьем уровне вкладки "Program" дерева, дважды щелкните узел, чтобы открыть соответствующее содержимое.

3.2 Создание нового проекта

Создание нового проекта включает в себя несколько следующих шагов:

Шаг	Операция
1	Создание нового проекта. См. раздел 3.2.1 «Создание проекта».
2	Настройка ПЛК. См. раздел 3.2.2 «Конфигурация аппаратного обеспечения ПЛК».
3	Создание пользовательских программ. См. раздел 3.3 «Управление программой».
4	Сохранение и компиляция программы.
5	Выгрузка проекта и программ. Смотрите 3.8 «Загрузка и выгрузка файла проекта» и 3.9 «Загрузка и выгрузка программных файлов».

3.2.1 Создание проекта

Для начала, создайте пустой проект. Откройте программу «СКПро», нажмите [File] /

[New] в главном меню или значок **h**а системной панели инструментов, как показано на рисунке 3.2:

🇱 SCPro - C:\SCPro-proj\TEST\TEST.prj - [MAIN]										
灁	File	Edit	View	LD	Online	Load	Window	Help		
Ľ		New								
k		Open.								
		Save								
12		User N	User Management							
11		Close								
		New P	proiect.				Ctrl+	N		

Рисунок 3.2 Создание нового проекта

В открывшемся диалоговом окне **Save As** будет предложено сохранить проект. Сохраните проект, например, как "test.prj", как показано на рисунке 3.3. Этот файл можно открыть напрямую, если в дальнейшем понадобится внести какие-либо изменения.

🗱 Save As		×
Папка:	TEST	← 🗈 📸 🖬 -
Имя	^	Дата изменения
Test.wsp)	30.09.2022 13:35
<		>
Имя файла:	Test.wsp	Сохранить
Тип файла:	SCPro Workspace Files (*.wsp)	• Отмена

Рисунок 3.3 Сохранение существующего файла

«СКПро» выведет диалоговое окно с предупреждением: «Пожалуйста, настройте ПЛК!», как показано на рисунке 3.4, выберите **ОК**, а затем создайте конфигурацию оборудования.

SCPro		×
	Please configure PLC!	
	ОК	

Рисунок 3.4 Диалоговое окно предупреждения

3.2.2 Конфигурация аппаратного обеспечения ПЛК

В браузере проекта дважды щелкните **[Resource] / [PLC Configuration]** левой кнопкой мыши, после чего появится диалоговое окно **Hardware Configuration**, как показано на рисунке ниже. Чтобы сохранить исходный файл проекта и исходные файлы программы в ПЛК, необходимо поставить галочку рядом с пунктом "Allow to upload". Если галочка поставлена, то исполняемые и исходные файлы будут загружены в ПЛК с возможностью последующей выгрузки; если галочка не поставлена, то будут загружены только исполняемые файлы и последующая выгрузка с помощью «СКПро» завершится неудачей.

P ⊕ … ≘ ⊕ … ≘ In M	ırce LC Config ask Config terrupt Co ODBUS/T	uration guration onfiguration CP Master
<		>
Drogram	Data r	

	⊙ SC-40	00	ଁ ୨	C-1000	
Rack Redund	lancy				
⊙ No	one	СВ	lus	C 10	
5.14					
Rack 1	None	_	Rack 5	None	_
Rack 2	None	-	Rack 6	None	-
Rack 3	None	-	Rack 7	None	-
Rack 4	None	-	Rack 8	None	-
Allow to up	load 🗆 IC) Channel	Redundancy	🗆 🗆 Creat	e Debug Inf

Рисунок 3.5 Конфигурация оборудования

Стойки (Rack) должны быть сконфигурированы в соответствии с техническими требованиями. Стойка 1 является основной стойкой по умолчанию, а остальные – стойки расширения. Для каждой стойки можно выбрать монтажную плату с 6 слотами, 9 слотами, 12 слотами или 15 слотами в соответствии с техническими требованиями. Нажмите кнопку **ОК** после завершения настройки, как показано на рис.Рисунок 3.6:

Hardware	e Configuration					×
	Series	• SC-4000		୍ ଚେ	C-1000	
	-Rack Redund	lancy]
	• No	ine	○ Bus		© 10	
	Rack 1	BK-4000-1201	• R	ack 5	None 🔽	
	Rack 2	BK-4000-0601	• R	lack 6	None 💌	
	Rack 3	None	• R	ack 7	None 💌	
	Rack 4	None	• R	lack 8	None 💌	
I	Allow to upl	oad 🗔 IO Cl	nannel Redu	indancy	🗖 Create Debug Info	
		ОК			Cancel	

Рисунок 3.6 Конфигурация оборудования

После нажатия кнопки **ОК**, аппаратная структура внутри браузера проекта будет выглядеть, как показано на рисунке 3.7



Рисунок 3.7 Аппаратная структура

Далее необходимо выбрать модули для каждой стойки. Дважды щелкните нужную стойку в браузере проекта, конфигурация модулей отобразится в правой области редактирования, все слоты в это время будут пусты, как показано на рисунке 3.8

SC-4000 - Rack 2



Рисунок 3.8 Пустая стойка

Указание

Любой слот стойки ПЛК серии СК-4000 не предъявляет особых требований к типу модуля, то есть почти любой модуль может быть сконфигурирован в любом слоте, если его можно выбрать. Исключение составляют системы с резервированными ПЛК.

Дважды щелкните модуль или пустой слот, появится диалоговое окно **Module**, а затем выберите группу модулей и тип модуля, как показано на рисунке 3.9:

Module	>
	Rack 2 — Slot 1
Module Group	Module Type
None CPU Module Digital Input Module Digital Output Module Digital Input/Output Module Sequence of Event Module Analog Input Module Analog Output Module Analog Input/Output Module High Speed Counter Module Special Module	CP-4000-0221—SC-4000 Advanced CPU Module (2 RS232, 1 Ethernet, Program Space 2 MB) CP-4000-0331—SC-4000 Advanced CPU Module (2 RS485, 1 Ethernet, Program Space 16 MB) CP-4000-0431—SC-4000 Advanced CPU Module (2 RS485, 2 Ethernets, Program Space 16 MB) CP-4000-0531—SC-4000 Advanced Redundant CPU Module (2 RS485, 1 Ethernet, Program Spa CP-4000-0631—SC-4000 Advanced IO Redundant CPU Module (2 RS485, 1 Ethernet, Program S CP-4000-0631—SC-4000 Advanced IO Redundant CPU Module (2 RS485, 1 Ethernet, Program S
Power Module	< >>
0K	Property

Рисунок 3.9 Конфигурация модуля

Группы включают в себя следующие модули: модули центрального процессора (CPU), модули дискретного ввода (digital input), модули дискретного вывода (digital output), модули дискретного ввода / вывода (digital input/output), модули последовательности событий (sequence of event), модули аналогового ввода (analog input), модули аналогового ввода (analog input), модули аналогового ввода / вывода (analog output), модули аналогового ввода / вывода (analog input/output), модули высокоскоростного счетчика (high speed counter), модули связи (communication), специальные модули (special) и блоки питания (power). "None" означает, что слот пуст, модуля нет.

После выбора модуля нажмите кнопку "*Property*", чтобы настроить свойства модуля, как показано на рисунке 3.10:

odule	x
	Rack 2 — Slot 1
Module Group	Module Type
None CPU Module	CP-4000-0221—SC-4000 Advanced CPU Module (2 RS232, 1 Ethernet, Program Space 2 MB) CP-4000-0331—SC-4000 Advanced CPU Module (2 RS485, 1 Ethernet, Program Space 16 MB) CD 4000-0431—SC 4000 Advanced CPU Module (2 RS485, 2 Ethernets, Program Space 16 MB)
Digital Output Module Digital Input/Output Module Sequence of Event Module Analog Input Module Analog Output Module Analog Input/Output Module High Speed Counter Module Communication Module Special Module Power Module	CP-4000-0531—SC-4000 Advanced Redundant CPU Module (2 RS485, 1 Ethernet, Program Spa CP-4000-0631—SC-4000 Advanced IO Redundant CPU Module (2 RS485, 1 Ethernet, Program S CP-4000-0631—SC-4000 Advanced IO Redundant CPU Module (2 RS485, 1 Ethernet, Program S
ОК	Property Cancel

Рисунок 3.10 Свойства модуля

Модуль центрального процессорного устройства (CPU Module)

Модули ЦПУ включают в себя базовые, стандартные, продвинутые и резервируемые продвинутые модули. Для каждого такого типа модулей есть несколько вариантов, отличающихся количеством последовательных портов, интерфейсов Ethernet и памятью программы/данных. Резервируемые продвинутые модули ЦПУ могут быть использованы только в резервированных системах. Для различных модулей ЦПУ конфигурации настройки встроенных процедура последовательных портов одинакова: необходимо установить значения параметров «Скорость передачи данных» (Baudrate), «Бит данных» (Data Bit), «Стоповый бит» (Stop Bit), «Четность» (Parity) и «Протокол» (Protocol) из соответствующих раскрывающихся списков. Существуют требования к конфигурациям Ethernet-адреса двойных сетей. IP-адрес Ethernet состоит из четырех частей. Для двойной сети первая, вторая и четвертая части IP-адреса обеих сетей должны быть одинаковыми, а третья - отличаться. Например, двойные сетевые адреса могут быть настроены следующим образом 192.168.200.100 и 192.168.201.100, как показано на рис.3.11:

Для ПЛК с возможностью централизованной синхронизации времени по протоколу NTP доступно два поля "NTP Server 1 IP Address" и "NTP Server 1 IP Address". Синхронизация времени происходит раз в минуту, текущее время ПЛК далее можно проверить в онлайн-режиме в %SW1-%SW7.

CPU Module Parameter	×
COM1	СОМ2
Address 1	Address 1
Baudrate 9600 💌	Baudrate 9600 💌
Data Bit 8	Data Bit 8
Stop Bit 1	Stop Bit 1
Parity None 🔹	Parity None 🔹
Protocol None	Protocol None
Ethernet 1 IP Address Subnet 0.0.0.0 Mask Default 0.0.0.0 Gateway 0.0.0.0 NTP Server 1 IP Address NTP Server 2 IP Address Advanced >>	IP 0 0 0 0 Address 0 0 0 0 Subnet 0 0 0 0 Mask 0 0 0 0 Default 0 0 0 0 Gateway 0 0 0 0

Рисунок 3.11 Параметры модуля процессора

Модули дискретного ввода

Существует несколько различных модулей дискретного ввода, однако принцип их настройки идентичен – как минимум, необходимо указать только стартовый номер бита для первого входа. После установки стартового номера, последующие будут присвоены автоматически, последовательным возрастанием, как показано на рисунке 3.12.

Все модули дискретного ввода имеют возможность фильтрации сигнала от дребезга, время фильтрации каналов настраивается в параметрах модуля. При неверной настройке времени фильтрации появится предупреждение, содержащее пределы (минимальный - 0, максимальный - 250) и единицу измерения (10 мс).

Digital Input Module Parameter		×
Start Num	nber (%I) 1	
Channel	Filter	^
1	0	
2	0	
3	0	
4	0	
5	0	
6	0	
7	0	
8	0	
n	n	
ОК	Cancel	

Рисунок 3.12 Параметры модуля цифрового ввода

Модули дискретного вывода

Существует несколько различных модулей дискретного вывода, однако принцип их настройки идентичен – как минимум, необходимо указать только стартовый номер бита для первого выхода. После установки стартового номера, последующие будут присвоены автоматически, последовательным возрастанием, как показано на рисунке 3.13. Все модули дискретного вывода настраиваются на безопасное состояние при ошибках вывода (ошибка самодиагностики модуля, ошибка связи с ПЛК), в выпадающем поле "Fault Output" имеется три режима настройки: "Hold" – значение канала останется тем же, что было до ошибки; "1" – значение канала установится в "Канал отключен"; "0" – значение канала установится в "Канал отключен".

<u>g</u>	Start Number	(%Q) 1 ÷	
Γ	Channel	Fault Output	
-	1	Hold	
	2	Hold	
	3	Hold	
	4	Hold	
	5	Hold	
	6	Hold	
	7	Hold	
	8	Hold	
	n	Uald	
	ОК	Cance	

Рисунок 3.13 Параметры модуля цифрового вывода

Модули аналогового ввода

Существует несколько различных модулей аналогового ввода, однако принцип их настройки идентичен – как минимум, необходимо указать только стартовый номер слова для первого входного канала. После установки стартового номера, последующие будут присвоены автоматически, последовательным возрастанием, как показано на рисунке 3.14.

Кроме того, в зависимости от модуля, каждый канал модуля имеет настройку типа сигнала ("Signal Type"), смещение нуля ("Zero Offset"), а так же может иметь настройки верхнего предела ("Upper Limit") и нижнего предела ("Lower Limit"). Единицы измерения сигнала АЦП зависят от выбранного типа сигнала. Подробное описание предоставлено в соответствующем системном руководстве на линейку ПЛК или на модуль.

При настройке модулей Al-4000-0801, Al-4000-0802, Al-4000-1601, Al-4000-0804 можно также настроить цикл опроса. Для цикла опроса можно выбрать значение Normal, либо Fast, где Normal – опрос раз в 400 мс, Fast – опрос при каждом выполнении цикла сканирования (выбор данного значения увеличивает общее время цикла).

Указание

Настройки "Zero Offset", "Upper Limit" и "Lower Limit" не смещают измеряемый диапазон, а лишь пересчитывают выходное значение.

		Start Number (%IW) 17	÷		
		Cycle © Normal O F	ast		
Channel	Signal Type	Zero Offset	Upper Limit	Lower Limit	Т
1	4~20mA	0	0	0	
2	4~20mA	0	0	0	
3	4~20mA	0	0	0	
4	4~20mA	0	0	0	
5	4~20mA	0	0	0	
6	4~20mA	0	0	0	
7	4~20mA	0	0	0	
8	4~20mA	0	0	0	



Модули аналогового вывода

Существует несколько различных модулей аналогового вывода, однако принцип их настройки идентичен – как минимум, необходимо указать только стартовый номер слова для первого выходного канала. После установки стартового номера, последующие будут присвоены автоматически, последовательным возрастанием, как показано на рисунке 3.15.

Кроме того, каждый канал модуля имеет настройку типа сигнала ("Signal Type"). Все модули аналогового вывода настраиваются на безопасное состояние при ошибках вывода (ошибка самодиагностики модуля, ошибка связи с ПЛК), в выпадающем поле "Fault Output" имеется два режима настройки: "Hold" – значение канала останется тем же, что было до ошибки; "Preset" – значение канала установится в предустановленное "Preset Value". Единицы измерения сигнала ЦАП зависят от выбранного типа сигнала. Подробное описание предоставлено в соответствующем системном руководстве на линейку ПЛК или на модуль.

	Start Number	(%QW) 5	· · · ·	
Channel	Signal Type	Fault Output	Preset Value	
1	4~20mA	Hold		
2	4~20mA	Hold		
3	4~20mA	Hold		
4	4~20mA	Hold		
-				
	ОК	C	ancel	

Рисунок 3.15 Параметры модуля аналогового вывода

3.3 Программа

В программе должна быть как минимум одна основная программа "MAIN". Каждый цикл сканирования начинается с основной программы, а циклы сканирования других подпрограмм выполняются путем вызова подпрограммы из основной программы. Основная программа "MAIN" и все подпрограммы перечислены на вкладке **Program** браузера проекта. Дважды щелкните по названию программы, чтобы увидеть её содержимое. Программы разделены на типы по языкам, на которых они написаны: LD, FBD, IL, ST, SCC и т.д., как показано на рисунке 3.17:



Вместо сканирования SCC выполняется методом последовательного управления, то есть пошагово, останавливаясь после выполнения последнего шага. Программа SCC никогда не выполняется автоматически, она может быть запущена только программой "MAIN" или другими подпрограммами. Выполнение SCC не влияет на выполнение программ LD или других программ.

3.3.1 Добавление программы

Если вы хотите добавить новую программу, в главном меню нужно запустить команду

[File] / [New Program] или кнопку на панели инструментов. Затем выбрать тип программы и дать ей название. "Program Description" заполнять не обязательно, но данное поле может помочь пользователю понять предназначение программы. Программа может быть основной задачей (выполнение цикла), а также задачей 1 ~ 16 (они выполняются своевременно, приоритет от низкого до высокого), также может быть прерыванием (прерывание по таймеру 1 ~ 4, прерывание ввода-вывода, уровень прерывания от низкого до высокого, в любой из выполняемых программ необходимо включить обработку прерываний функциональным блоком "ENI"), как показано на рис.3.18:



Рисунок 3.17 Добавление программы

3.3.2 Удаление программы

Если вы хотите удалить программу, щелкните на программе правой кнопкой мыши и выберите **Delete** (основная программа "MAIN" создается по умолчанию, и не может быть удалена), как показано на рисунке 3.19:



Рисунок 3.18 Удаление программы

3.3.3 Изменение названия программы

Если вы хотите переименовать программу, щелкните на программе правой кнопкой мыши и выберите **Rename** (основная программа "MAIN" не может быть переименована), как показано на рисунке 3.20:





3.3.4 Добавление описания программы

Если вы хотите добавить описание программы, щелкните на программе правой кнопкой мыши и выберите *Description*. Содержимое описания отображается в области редактирования программы, как показано на рисунке 3.21:



Рисунок 3.20 Добавление описания программы

3.3.5 Защита программы

Если вы хотите защитить программу, щелкните на программе правой кнопкой мыши и выберите **Protect**, как показано на рисунке 3.22. Откроется диалоговое окно **Protect**, как показано на рисунке 3.23, «None» означает отсутствие защиты; «Read Only» означает, что после защиты программа может быть только просмотрена другими людьми, но не может быть изменена; «No Read or Write» означает, что после защиты программа не может быть просмотрена другими людьми, а также не может быть удалена или изменена. Если вы хотите изменить программу, вам необходимо ввести пароль и изменить параметр защиты программы на «None».

Предупреждение

После защиты запомните пароль, так как его невозможно восстановить без передачи копии разработанного ПО службе поддержки!



Рисунок 3.21 Защита программы

•
r Write

Рисунок 3.22 Свойства защиты

3.4 Задача (Task)

Проект может быть сконфигурирован таким образом, чтобы исполнять несколько задач. Вызов программ, в таком случае, будет зависеть от распределения задач.

В состоянии по умолчанию система выполняется как единая задача, «MAIN» — это основная программа, она выполняется автоматическим и самой первой. Затем она прямо или косвенно вызывает другие программы. В таком случае нет необходимости отдельно настраивать задачи. Как правило, ПЛК управляет средой выполнения одной задачи, что удовлетворяет требования к работе системы.

В рабочем окружении с несколькими задачами основная задача (автоматически выполняющая программу «MAIN») выполняется циклически, и её приоритет является самым высоким. В рабочем окружении поддерживается до 16 задач, и приоритет задачи № 16 является самым низким. Каждая программа должна относится к основной задаче или любой из задач № 1 - 16 (настроить можно при создании новой программы); программы, относящиеся к одной и той же задаче, могут вызываться между собой.

Все задачи и все программы каждой задачи перечислены в разделе **Task Configuration** вкладки **(Resource)** браузера проекта, как показано на рисунке 3.24:



Рисунок 3.23 Список задач

Программы LD2 и TASKDEALER относятся к задаче 2, щелкните правой кнопкой мыши по задаче, чтобы выбрать **Configuration**, а затем откройте диалоговое окно **Task Configuration**, установите для параметра "Automatically Execute Program" (Автоматически выполнять программу) значение LD2 и установите "Cycle" (Цикл) равный 200 мс, как показано на рисунке 3.25. Временной цикл не может быть установлен менее, чем на 1 миллисекунду. Программа автоматического выполнения (LD2) начинает выполняться и вызывает другие программы (TASKDEALER), относящиеся к этой задаче.

Если цикл вызова программы Task1-Task16 менее общего цикла выполнения программ %SW32 "Scan Time", то программа Task1-Task16 будет вызвана в цикле %SW32, но не чаще!

Пример 1: Task1 сконфигурирована на цикл 2 мс, и программист сделал таймер на цикл 2 мс, но в результате Task1 выполняется после вызова Main Task, и время между его вызовами равно %SW32, в результате таймер сильно занижает свое значение и не достоверен.

Пример 2: Task1 сконфигурирована на цикл 200 мс, и программист сделал таймер на цикл 200 мс, но в результате Task1 выполняется после вызова Main Task, и время между его вызовами равно 200 ± %SW32 мс, в результате таймер недостоверен.

Task Configuration		×			
Task 2					
Automatically Execute Program	LD2 -				
Cycle	200 (ms)				
ОК	Cancel				

Рисунок 3.24 Конфигурация задач

3.5 Прерывание

Прерывание — это определённый тип реагирования ПЛК на внешние события или внутренние события и их обработка. Данный процесс включает в себя три части: событие прерывания, программу прерывания и управление прерыванием. Прерывание осуществляется в результате появления события прерывания, таковым может быть прерывание ввода-вывода или прерывание по таймеру. При возникновении события прерывания, ПЛК прерывает сканирование текущей программы, и право управления ПЛК переходит программе прерывания, запущенной этим событием прерывания. После выполнения программы прерывания право управления автоматически возвращается прерванной программе ПЛК, и она продолжает выполняться.

ПЛК СК-х000 поддерживают пять видов прерываний, а именно, прерывание по таймеру 1, прерывание по таймеру 2, прерывание по таймеру 3, прерывание по таймеру 4 и прерывание ввода-вывода. Уровень прерывания таймера 1 является самым низким, а уровень прерывания ввода-вывода - самым высоким. ПЛК обрабатывает события прерывания согласно уровням.

Программа прерывания настраивается при создании новой программы, и чем она короче, тем лучше.

Управление прерыванием осуществляется функциональными блоками **ENI** (Включение прерывания) и **DISI** (Отключение прерывания). При запуске ПЛК прерывания отключены. Прерывание осуществляется только когда программа прерывания включена, что позволяет вызвать её. Когда программа прерывания отключена, аппаратное обеспечение продолжает принимать ответ на прерывание, но вызывать программу прерывания оно не может.

Все прерывания и все программы каждого прерывания перечислены в разделе *Interrupt Configuration* вкладки **(Resource)** браузера проекта. В примере программа

прерывания таймера 1 (Timer INT 1) - это программа LD "TMR1", программа прерывания таймера 2 (Timer INT 2) - программа FBD "TMR2", программа прерывания таймера 3 ((Timer INT 3) - программа IL "TMR3", программа прерывания таймера 4 (Timer INT 4) программа ST "TMR4", программа прерывания ввода-вывода (I/O INT)- - это программа LD "IO", как показано на рисунке:



Рисунок 3.25 Список прерываний

Для срабатывания прерывания таймера 1...4 необходимо настроить время таймера. Щелкните правой кнопкой мыши на прерывании, которое необходимо настроить, чтобы выбрать *Configuration*, затем появится диалоговое окно **Interrupt Configuration**, введите цикл, как показано на рисунке 3.27.

Interrupt Configuration		
	Timer INT 1	
Cycle	200	(ms)
ОК	С	ancel

Рисунок 3.26 Конфигурация таймера прерывания

3.6 Защита проекта

Чтобы защитить конфигурацию проекта и программы от просмотра или изменения, для проекта может быть установлен пароль. Существует два уровня паролей: для файлов и для входа в систему. Пароль к файлу используется для открытия файла проекта, а пароль для входа в систему используется для подключения, загрузки и выгрузки. По умолчанию, файлу нового проекта пароль не назначается.

3.7 Подключение и отключение

Подключение означает, что «СКПро» подключено к ПЛК по сети Ethernet и обменивается с ним пакетами данных. Чтобы «СКПро» начало процесс подключения, щелкните на значок **[Connect]** на системной панели инструментов. После успешного подключения можно просмотреть статус выполнения программы, напрямую запросить статус каждого регистра, а также принудительно вызвать некоторые регистры при помощи программ. При неудачном подключении будет выдан сигнал о сбое подключения.

Отключение означает, что «СКПро» отключено от ПЛК, и в отключенном состоянии может быть изменена каждая составляющая проекта, а также могут быть загружены файлы проекта и программы.

3.8 Загрузка и выгрузка файла проекта

После создания или изменения файла проекта он должен быть загружен в ПЛК, в противном случае при подключении к ПЛК появится сообщение, показанное на рисунке 3.28:



Рисунок 3.27 Предупреждение о подключении

Загрузка означает, что отредактированные программы (файлы проекта и программы) выгружаются в ПЛК.

Выгрузка означает, что файлы проекта и программы ПЛК сохраняются на компьютере (они могут быть загружены только при отметке галочкой пункта "Allow to upload".).

Download project (Загрузка проекта)

Загрузите файл проекта в ПЛК. При загрузке «СКПро» автоматически выполнит поиск узла в сети и продолжит загрузку в соответствии с адресом Ethernet в конфигурации оборудования. Во время процессов поиска и загрузки появится сообщение "Please wait while downloading project..." ("Пожалуйста, подождите пока идет выгрузка проекта..."). Если поиск узла ПЛК невозможен, об этом известит сигнал о сбое подключения. При работе с резервной системой, «СКПро» автоматически загрузится в оба ПЛК при наличии подключения к резервному ПЛК. После загрузки программ в ПЛК, модуль центрального процессора должен быть сброшен и перезапущен один раз, после чего выгруженные программы могут быть выполнены, в противном случае система выполнит программы, находившиеся в ПЛК до загрузки. Операция сброса может бытьвыполнена командой **Reset**.

Manually download (Ручная загрузка)

Ручная загрузка часто используется для первой загрузки файла проекта и всех программных файлов. IP-адрес установлен по умолчанию, в случае если ПЛК не требуется загружать файл проекта. После выбора команды *Manually Download* откроется диалоговое окно, как показано на рисунке 3.29:

Ethernet IP	×
Ethernet IP	<mark>192</mark> .168.1.66
ОК	Cancel

Рисунок 3.28 Ручная загрузка

Нажмите кнопку **ОК**, чтобы загрузить файлы проекта и программы, после чего система отобразит окно процесса, как показано на рисунке 3.30:

Please wait v	vhile down	loading m	annually	
riodoo maiti		iouuing iii	amaanym	

Рисунок 3.29 Процесс ручной загрузки

Если сетевое оборудование не подключено или адрес указан неверно, то «СКПро» откроет окно, как показано на рисунке 3.31:



Рисунок 3.30 Сбой загрузки

В случае сбоя необходимо проверить сетевые конфигурации.

Примечание: IP-адрес компьютера для отладки должен находиться в том же адресном пространстве, что и IP-адрес ПЛК, то есть, первые три сегмента IP-адреса должны совпадать. В противном случае соединение установить не удастся. Например: если IP-адрес ПЛК - 192.168.1.66, то IP-адрес компьютера для отладки должен начинаться с 192.168.1.***. Более подробно – см. системное руководство по подключению к ПЛК.

Upload project (Выгрузка проекта)

Данная функция требуется для выгрузки проекта с ПЛК на компьютер для отладки. Отличие от загрузки заключается в том, что при выгрузке появляется диалоговое окно, в котором будет предложено ввести IP-адрес Ethernet ПЛК, по которому требуется выгрузить файл. После ввода нажмите кнопку **ОК** и «СКПро» выполнит поиск узла в сети в соответствии с адресом, как показано на рисунке 3.32:

Ethernet IP	×
Ethernet IP 192.168.1.66	
OK Cancel	

Рисунок 3.31 Выгрузка проекта

После выгрузки «СКПро» предложит ввести название для сохраненяемого проекта. Можно перезаписать файл проекта или другие файлы с теми же именами на текущем компьютере либо ввести новое имя для сохранения в новом файле.

3.9 Загрузка и выгрузка программ

Download program (Загрузка программы)

Если файл проекта не изменен, а изменен только файл программы, то файл программы в ПЛК может быть обновлен с помощью функции *Download Program*.

Upload program (Выгрузка программы)

Выгрузите все программные файлы с ПЛК, включая LD, FBD, ST, IL, SCC и т.д. Выгрузка программ аналогична выгрузке файла проекта, для этого также необходимо ввести IPадрес ПЛК. После этого появится диалоговое окно, в котором необходимо выбрать название программы для выгрузки, также можно выбрать все программы, как показано на рисунке 3.33:
Program		×
All Programs		Selected Programs
FPQ2	Add ->	
TEST_P SC TEST_UD	<- Delete	
	Add All ==>	
	<== Delete All	
ОК]	Cancel



Download All (Загрузить все)

Загрузка файла проекта и всех программных файлов в ПЛК одновременно.

Download All Programs (Загрузить все программы)

Загрузка всех программных файлов в ПЛК.

4 Управление данными

Стандарт IEC61131-3 определил наиболее часто используемые типы данных при программировании ПЛК, поэтому в отношении ПЛК эти типы данных определяются и применяются единообразно. Это упрощает конфигурирование и обслуживание систем, оснащенных ПЛК производства различных компаний, для производителей машин и установок и инженерно-технического персонала: унифицированные типы данных повышают читаемость и портируемость программ ПЛК.

4.1 Тип данных

Ключевое слово	Тип	Бит	Допустимый диапазон	Описание	
BOOL	Boolean	1	0 или 1	Хранится в единицах бита только в двух состояниях: 1 или 0.	
BYTE	Byte	8	0~255	Используется 8 бит регистра данных, 8 бит данных могут быть независимыми и указывать только на состояние текущего бита 0 или 1; также они могут быть целым числом без знака в диапазоне от 0 до 255.	
WORD	Word	16	0~65535	Используется 16 бит регистра данных, 16 бит данных могут быть независимыми и указывать только на состояние текущего бита: 0 или 1; также они могут быть целым числом без знака в диапазоне 0~65535.	
DWORD	Double word	32	0~4294967295	Используется 32 бита регистра данных, 32 бита данных могут быть независимыми и указывать только на состояние текущего бита: 0 или 1; также они могут быть целым числом без знака в диапазоне 0~4294967295.	
SINT	Short integer	8	-128~+127	Используется 8 бит регистра данных, указывается целое число со знаком в диапазоне -128~+127.	
INT	Integer	16	-32768~+32767	Используется 16 бит регистра данных, указывается целое число со знаком в диапазоне -32768~+32767.	
DINT	Double integer	32	- 2147483648~+214 7483647	Используется 32 бита регистра данных, указывается целое число со знаком в диапазоне -2147483648 ~+2147483647.	
REAL	Real	32		Указывается значение с плавающей запятой.	

ПЛК СК-4000 работает со следующими типами данных:

	Unsigned		Используется 8 бит регистра данных,
USINT	short 8	0~255	указывается целое число без знака в
	integer		диапазоне 0~255.
	Unsigned		Используется 16 бит регистра данных,
UINT	integer 16	0~65535	указывается целое число без знака в
	integer		диапазоне 0~65535.
	Unsigned		Используется 32 бита регистра данных,
UDINT	double 32	0~4294967295	указывается целое число без знака в
	integer		диапазоне 0~4294967295.

4.2 Управление данными

4.2.1 Вкладка "Data"

В браузере проекта вкладка **[Data]** включает в себя следующие элементы: **Point Table**, **Variable Table**, **Optional Point Table** и т.д., как показано на рис.4.1:

🗎 Data

- DFB Type Table
- DFB Instance Table
- Point Table

 - Real-time Curve

Рис.4.1 Вкладка "Data"

4.2.2 Point table (Таблица точек данных)

Тип точки

Тип	Наименова ние	Тип данных	Описание			
I	Digital input (Дискретн ый вход)	BOOL, 0 или 1	Текущее состояние базовой точки дискретного входа.			
Q	Digital output (Дискретн ый выход)	BOOL, 0 или 1	Текущее состояние базовой точки дискретного выхода.			
IW	Analog input (Аналогов ый вход)	INT, напряжение и ток: 0~20000 Температурный сигнал: - 32768~+32767	Текущее значение базовой точки аналогового входа.			

QW	Analog output (Аналогов ый выход)	WORD, 0~20000	Текущее значение базовой точки аналогового выхода.		
М	Bit register (Битовый регистр)	BOOL, 0 или 1	Логическая переменная область хранения, предлагаемая пользователю системой.		
MW	Word register (Регистр слов)	WORD, 0~65535	Область хранения переменной Word, предлагаемая пользователю системой.		
N	Non- volatile bit register (Энерго- независи- мый битовый регистр)	BOOL, 0 или 1	Отличие от регистра М заключается в энергонезависимости. После отключения питания ПЛК данные в регистре N не будут утеряны.		
NW	Non- volatile word register (Энерго- независи- мый регистр слов)	WORD, 0~65535	Отличие от регистра МШ заключается в энергонезависимости. После отключения питания ПЛК данные в регистре NW не будут утеряны.		
S	System bit register (Регистр системных бит)	BOOL, 0 или 1	Логическая переменная, отражающая текущее состояние системы, определенное внутри системы. Каждый элемент имеет свое конкретное определение. Может быть прочитан, но не может быть переписан.		
SW	System word register (Регистр системных слов)	WORD, 0~65535	Переменная в формате word, отражающая текущее состояние системы, определенное внутри системы. Каждый элемент имеет свое конкретное определение. Может быть прочитан, но не может быть переписан.		

Т	Timer (Таймер)	DWORD, 0~60000	Таймер, предлагаемый пользователю системой.
С	Counter (Счетчик)	DWORD, 0~4294967295	Счетчик, предлагаемый пользователю системой.

ПЛК СК-4000 предоставляет большое количество системных регистров для хранения состояния работы системы. Пользователю предоставляется удобный интерфейс для считывания и контроля состояния работы ПЛК в режиме реального времени.

Определения системного регистра:

Номер	Наименование	Описание
System b	it register	
%S0001	FIRST_SCAN	Первое сканирование
%S0002	ALWAYS_ON	Всегда включен
%S0003	ALWAYS_OFF	Всегда выключен
%S0004	T_SECOND	Таймер на 1 секунду
%S0006	PRG_OVERRUN	Переполнение выполнения программы
%S0007	PRG_EXECERR	Ошибка выполнения программы
%S0010	BACKUP_OK	Резервное копирование устройства резервированной пары ЦПУ прошло успешно
%S0033	IO_COMERR	Ошибка связи модуля ввода-вывода
%S0034	IO_DIAGERR	Ошибка самодиагностики модуля ввода- вывода
%S0035	IO_CFGERR	Несоответствие типа модуля ввода- вывода
%S0036	COM_COMERR	Ошибка связи модуля связи
%S0037	COM_DIAGERR	Ошибка самодиагностики модуля связи
%S0038	COM_CFGERR	Несоответствие типа модуля связи
%S0039	VER_DISMATCH	Несоответствие версий ПО резервированной пары ЦПУ
%S0041	IO_DOWNLOAD	Инициализация модуля ввода-вывода прошла успешно

%S0042	COM_DOWNLOAD	Инициализация модуля связи прошла успешно		
Состояние работы процессора				
%S0097	CPU1_MASTER	Мастер СРU1		
%S0098	CPU1_FAULT	Ошибка CPU1		
%S0099	CPU1_GPSLOST	Ошибка GPS CPU1		
%S0100	CPU1_SECOND	Запуск СРU1		
%S0101	CPU1_CAN1FLT	Ошибка CAN1 CPU1		
%S0102	CPU1_CAN2FLT	Ошибка CAN2 CPU1		
%S0103	CPU1_ETH1FLT	Ошибка ЕТН1 СРU1		
%S0104	CPU1_ETH2FLT	Ошибка ETH2 CPU1		
%S0105	CPU1_TASKFLT	Ошибка задачи CPU1		
%S0106	CPU1_SELFON	СРU1 онлайн		
%S0107	CPU1_FATAL	Фатальная ошибка CPU1		
%S0114	CPU1_PEERON	CPU1 одноранговый узел онлайн		
%S0115	CPU1_PEERMST	Одноранговый мастер CPU1		
%S0116	CPU1_STOP	Состояние остановки CPU1		
%S0117	CPU1_DEBUG	Состояние отладки CPU1		
%S0118	CPU1_NVRAMFLT	Ошибка NVRAM CPU1		
%S0120	CPU1_FIRST	СРU1 первый		
%S0121	CPU2_MASTER	Мастер СРU2		
%S0122	CPU2_FAULT	Ошибка CPU2		
%S0123	CPU2_GPSLOST	Ошибка GPS CPU2		
%S0124	CPU2_SECOND	Запуск CPU2		
%S0125	CPU2_CAN1FLT	Ошибка CAN1 CPU2		
%S0126	CPU2_CAN2FLT	Ошибка CAN2 CPU2		
%S0127	CPU2_ETH1FLT	Ошибка ETH1 CPU2		
%S0128	CPU2_ETH2FLT	Ошибка ETH2 CPU2		
%S0129	CPU2_TASKFLT	Ошибка задачи CPU2		
%S0130	CPU2_SELFON	СРU2 онлайн		

%S0131	CPU2_FATAL	Фатальная ошибка CPU2
%S0138	CPU2_PEERON	CPU2 одноранговый узел онлайн
%S0139	CPU2_PEERMST	Одноранговый мастер CPU2
%S0140	CPU2_STOP	Состояние остановки CPU2
%S0141	CPU2_DEBUG	Состояние отладки CPU2
%S0142	CPU2_NVRAMFLT	Ошибка NVRAM CPU2
%S0144	CPU2_FIRST	CPU2 первый
Состоян	ие основной связи І	MODBUS/TCP (1: Ошибка, 0: ОК)
%S0145		Запрос 1
%S0146		Запрос 2
%S0208		Запрос 64
Состоян	ие работы модуля	
%S0513	MDU001_COMERR	Ошибка связи модуля (адрес 001)
%S0514	MDU001_DIAGERR	Ошибка самодиагностики модуля (адрес 001)
%S0515	MDU001_CFGERR	Несоответствие типа модуля (адрес 001)
%S0516	MDU001_RVS1	Резерв 1 модуля (адрес 001)
%S0517	MDU001_RVS2	Резерв 2 модуля (адрес 001)
%S0518	MDU001_RVS3	Резерв 3 модуля (адрес 001)
%S0519	MDU001_RVS4	Резерв 4 модуля (адрес 001)
%S0520	MDU001_RVS5	Резерв 5 модуля (адрес 001)
%S0521	MDU002_COMERR	Ошибка связи модуля (адрес 002)
%S0522	MDU002_DIAGERR	Ошибка самодиагностики модуля (адрес 002)
%S0523	MDU002_CFGERR	Несоответствие типа модуля (адрес 002)
%S0524	MDU002_RVS1	Резерв 1 модуля (адрес 002)
%S0525	MDU002_RVS2	Резерв 2 модуля (адрес 002)
%S0526	MDU002_RVS3	Резерв 3 модуля (адрес 002)

%S0527	MDU002_RVS4	Резерв 4 модуля (адрес 002)
%S0528	MDU002_RVS5	Резерв 5 модуля (адрес 002)
System v	vord register	
%SW0001	TIME_YEAR	Часы: год
%SW0002	TIME_MONTH	Часы: месяц
%SW0003	TIME_DAY	Часы: день
%SW0004	TIME_HOUR	Часы: час
%SW0005	TIME_MINUTE	Часы: минута
%SW0006	TIME_SECOND	Часы: секунда
%SW0007	TIME_MS	Часы: миллисекунда
%SW0008	TIME_Week	Часы: День недели
%SW0009	ALARM_PTR	Указатель тревоги
%SW0010	SOE_PTR	Указатель SOE
%SW0011	OVERRUN_INFO	Место переполнения программы
%SW0014	EXERERR_INFO	Место ошибки программы
%SW0021	COM1_SEND	Состояние отправки СОМ1
%SW0022	COM1_RECV	Состояние приема СОМ1
%SW0023	COM2_SEND	Состояние отправки СОМ2
%SW0024	COM2_RECV	Состояние приема СОМ2
%SW0025	STTM_YEAR	Время начала: год
%SW0026	STTM_MONTH	Время начала: месяц
%SW0027	STTM _DAY	Время начала: день
%SW0028	STTM_HOUR	Время начала: час
%SW0029	STTM _MINUTE	Время начала: минута
%SW0030	STTM_SECOND	Время начала: секунда
%SW0031	STTM_MS	Время начала: миллисекунда
%SW0032	SCAN_TIME	Время сканирования

Свойство точки данных

Все точки данных имеют два общих вида свойства, и особые свойства, которыми обладают только некоторые точки.

Общие свойства

[Number]: Номер используется для того, чтобы различать точки данных. Он генерируется автоматически и не может быть изменен. Номер может быть непосредственно использован в качестве объекта операции блока функции или инструкции. Номер состоит из двух частей: тип точки и номер точки, например %10001, %MW0100 и т.д. "Тип точки" указывает текущий тип точки данных, например, "%I" указывает точку дискретного входа, "%MW" указывает регистр слов. "Номер точки" указывает текущий последовательный номер точки, который не может превышать своего максимального значения. Для разных типов точек и разных типов процессоров максимальное значение отличается.

Тип	Наименование	CP-4000- 0221	CP-4000- 0331	CP-4000- 0431	CP-4000- 0531	CP-4000-0631
I	Дискретные входы	1024	4096	8192	16384	16384
Q	Дискретные выходы	1024	4096	8192	16384	16384
IW	Аналоговые входы	256	1024	4096	4096	4096
QW	Аналоговые выходы	256	1024	4096	4096	4096
М	Битовые регистры	4096	8192	16384	16384	16384
MW	Регистры слова	4096	16384	32768	16384	32768
N	Энергонезависимы е битовые регистры - реманентные (N)	1024	2048	4096	4096	4096
NW	Энергонезависимы е регистры слова - реманентные (NW)	1024	2048	4096	4096	4096
S	System bit register (Регистр системных бит)	1024	2048	4096	4096	4096
SW	System word register (Регистр системных слов)	1024	2048	4096	4096	4096
Т	Timer (Таймер)	256	512	1024	1024	1024
С	Counter (Счетчик)	256	512	1024	1024	1024

Каждой точке данных может быть присвоено имя, которое может быть непосредственно использовано в качестве объекта операции при программировании LD и SCC и т.д., а также может быть отображено непосредственно в программе. Например, точке "%10001" присвоено имя "DL_ON". В области редактирования LD поместите контакт и введите параметр "DL_ON", после компиляции «СКПро» автоматически его идентифицирует. Если имя точки изменено, то точка "DL_ON" заменяется на вторую точку цифрового ввода, нужно только присвоить точке %10002 имя "DL_ON", и никаких других изменений в программе не потребуется.

[Description] : Каждой точке данных также может быть присвоено описание, которое должно подробно описывать точку данных. Например, имя точки данных %10001 - "DL_ON", описание – «автоматический выключатель включен». Если при чтении программы LD вы не поймете значения точки "DL_ON" из ее названия, можете ознакомиться с описанием. Однако содержимое описания не отображается в LD.

【Used Times】: Эта функция отображает сколько раз текущая точка данных была применена с прямой адресацией в программах, данная информация доступна в 【File 】 / 【Used Times】, после сохранения нет необходимости пересчитывать в следующий раз. Количество применения точки особенно полезно для таймера и счетчика: его можно просмотреть, чтобы избежать повторного применения.

Особые свойства

[Module Address] : Адрес модуля действителен только для физических точек данных, таких как дискретный вход I, дискретный выход Q, аналоговый вход IW и аналоговый выход QW. Адрес модуля – это адрес модуля, в котором находится текущая точка. Он генерируется автоматически после того, как задана точка. Адрес не может быть изменен. Адрес модуля в стойке 1 - от 1 до 15, адрес модуля в стойке 2 - от 16 до 30 и т.д. Например, если модуль, на котором расположена текущая точка, установлен в третий слот второй стойки, то адрес модуля будет равен 18.

(Filter Time) : Время фильтрации применимо только к дискретному входу I. Каждому входящему цифровому сигналу может быть добавлено определенное время фильтрации для предотвращения дребезга сигналов. Единица измерения времени фильтрации - 10 мс, она может быть изменена в диапазоне 0 ~ 250. Как правило, время фильтрации по умолчанию составляет 0 мс. Если введённое значение не входит в указанный диапазон, «СКПро» отобразит это в соответствующем окне, как показано на рис.4.2:



Рис.4.2 Время фильтрации

[Force] : Принудительный ввод (форсирование) действителен только для физических точек данных, таких как дискретный вход I, дискретный выход Q, аналоговый вход IW и аналоговый выход QW. При помощи форсирования точки, вы можете установить значение точки в соответствии с вашими требованиями, независимо от текущего значения сигнала. Принудительный ввод может быть выполнен только в режиме онлайн работы ПЛК. Дважды щелкните на панели форсирования точки, для которой необходимо установить значение. На панели появится отметка √, что говорит о том, что эта точка была форсирована. Дважды щелкните еще раз, и метка пропадет. Это означает, что эта точка не форсирована. В состоянии принудительного ввода состояние сигнала, отсканированного системой, никогда не будет отправлено в область хранения точек данных.

[Value] : Функция значения показывает текущие значения фактических точек данных в оперативном состоянии для дискретного входа I, дискретный выхода Q, аналогового входа IW и аналоговый выхода QW; а также показывает текущие значения виртуальных точек данных в текущем состоянии для битового регистра M, регистра слов MW, энергонезависимого битового регистра N, энергонезависимого регистра слов NW, системного битового регистра S и системного регистра слов SW. Для системного регистра S и SW оно может быть только прочитано и не может быть записано.

[Signal Type] : Функция "Тип сигнала" действительна только для аналогового входа IW и аналогового выхода QW. Аналоговый модуль может иметь различные типы сигналов, такие как тип "Напряжение", "Ток" или "Термосопротивление".

(Preset Value) : Предустановленное значение действительно только для таймера Т и счетчика С (предустановленное значение в настройках аналоговых выходных каналов можно посмотреть в настройках соответствующего аналогового выходного модуля). В режиме онлайн можно просмотреть заданное время таймера или заданное значение счетчика.

[Current Value] : Текущее значение действительно только для таймера T и счетчика C. В режиме онлайн можно просмотреть текущее время таймера или текущее значение счетчика.

(Dimension) : Размерность применима только для переменной %V, доступной в Variable Table. Размерность переменной составляет не более 4096. Индекс переменной начинается с 0, например, имя %V0001 определяется как "status", его размерность равна 100, тогда можно использовать "status[0] ~status[99]". Если это одномерная переменная, нужно только ввести имя переменной без нижнего индекса при вводе, например, %V0001 (имя "status"), размерность одна, при использовании нужно только ввести "status" вместо "status[0]". Имя переменной должно быть определено при использовании переменной, например, имя %V0001 - "status", тогда вместо "%V[0]" следует использовать "status".

(Туре) : Тип действителен только для переменной %V. Выберите тип данных переменной с помощью выпадающего списка. Типом данных могут быть все типы в ПЛК, включая BOOL, BYTE, WORD, DWORD, SINT, INT, DINT, REAL, USINT, UINT и UDINT и т.д. (Для получения подробной информации о типе данных, пожалуйста, см. часть «Тип данных» этой главы), однако каждый массив переменных может быть определен только как данные одного и того же типа.

[Zero Offset] : Смещение нуля применимо только к аналоговому входу IW и используется для компенсации ошибок фактического сигнала.

4.2.3 Таблица переменных

В «СКПро» существует своего рода виртуальная точка данных, которая называется переменной. Переменная хранит данные в виде массива. Размерность массива и тип данных могут быть установлены в допустимом диапазоне. В режиме онлайн текущее значение данных можно просмотреть с помощью таблицы переменных, как показано на рисунке 4.3. Для получения подробной информации о просмотре и изменении переменной см. главу 11.6.

HSC_CPU HSC_DATA[] 2 PTO_CPU PTO_CPU PTO_DATA[] 4 MOT_CPU MOT_CPU MOT_DATA 1 D1_1 D_1 BOOL 1 SetPoint SetPoint REAL 1 TI T REAL 1 TD TD REAL 1 Ymax Ymax REAL 1 Ymax Ymax REAL 1 Man Man BOOL 1 Yman Yman REAL 1 Yv Y REAL 1	Name	Input	Data Type	Dimension	Value	Comment
PTO_CPU PTO_CPU PTO_DATA[] 4 MOT_CPU MOT_CPU MOT_DATA 1 Dl_1 Dl_1 BOOL 1 SetPoint SetPoint REAL 1 KP KP REAL 1 TI T REAL 1 Sample_time Sample_time REAL 1 Sample_time Sample_time REAL 1 Sample_time Sample_time REAL 1 Ymin Ymax REAL 1 Point DeadBand REAL 1 Man Man BOOL 1 Yman Yman REAL 1 Ynan Yman REAL 1 Ynan Yman REAL 1 Yman Yman REAL 1 Ynan Yman <td>HSC_CPU</td> <td>HSC_CPU</td> <td>HSC_DATA[]</td> <td>2</td> <td></td> <td></td>	HSC_CPU	HSC_CPU	HSC_DATA[]	2		
MOT_CPUMOT_CPUMOT_DATA1MOT_CPUDL1BOOL1Image: DL1DL1BOOL1Image: DL1SetPointSetPointREAL1Image: DL1TIREAL11Image: DL1TDREAL11Image: DL1TDREAL11Image: DL1TDREAL11Image: DL1TDREAL11Image: DL1TDREAL11Image: DL1TomanREAL11Image: DL1DeadBandREAL11Image: DL1ManBOOL11Image: DL1TomanREAL11Image: DL1TomanREAL		PTO_CPU	PTO_DATA[]	4		
Dl_1Dl_1BOOL1StPbintStPbintREAL1KPKPREAL1TIREAL1TDTDREAL1YmaxYmaxREAL1YmanYmanYman <t< td=""><td>🖶 🗗 MOT_CPU</td><td>MOT_CPU</td><td>MOT_DATA</td><td>1</td><td></td><td></td></t<>	🖶 🗗 MOT_CPU	MOT_CPU	MOT_DATA	1		
SetPoint SetPoint REAL 1 VP VP REAL 1 Th Th REAL 1 TD TD REAL 1 Man Man REAL 1 Man Man BOOL 1 Yman Yman REAL 1	DI_1	DI_1	BOOL	1		
KP KP REAL 1 Image: The state of the s	SetPoint	SetPoint	REAL	1		
III II REAL 1 ID TD REAL 1 Sample,time Sample,time REAL 1 Image:	📝 KP	KP	REAL	1		
TD TD TD REAL 1 Image: Sample_time Sample_time REAL 1 Image: Sample_time Ymax REAL 1 Image: Sample_time Ymax REAL 1 Image: Sample_time DeadBand REAL 1 Image: Sample_time DeadBand REAL 1 Image: Sample_time Man BOOL 1 Image: Sample_time Yman REAL 1 Image: Sample_time Yman REAL 1 Image: Sample_time REAL 1 1 Image: Sample_time Image: Sample_time 1 1	🖸 Π	п	REAL	1		
Sample_time Sample_time REAL 1 Ymax Ymax REAL 1 Ymin Ymin REAL 1 DeadBand DeadBand REAL 1 Man BOOL 1 Yman Yman REAL 1 Yman Yman REAL 1 Ytan Yman REAL 1 Ytan Yman REAL 1 Ytan Yman REAL 1	🔽 TD	TD	REAL	1		
Ymax Ymax REAL 1 Ymin Ymin REAL 1 DeadBand DeadBand REAL 1 Man BOOL 1 1 Yman Yman REAL 1 Yman Yman REAL 1 Yn Yman REAL 1 Yt Yman REAL 1 Yt Y REAL 1 Yt Y REAL 1 Yt Y REAL 1	Sample_time	Sample_time	REAL	1		
Ymin Ymin REAL 1 P DeadBand DeadBand REAL 1 Man Man BOOL 1 Man Yman REAL 1 Yman Yman REAL 1 Yt Y REAL 1 Yt Y REAL 1 Yt Y REAL 1		Ymax	REAL	1		
Image: Provide and the set of t	Vmin	Ymin	REAL	1		
Image: Man Man BOOL 1 Image: Man Yman REAL 1 Image: Man Y REAL 1 Image: Man Y REAL 1 Image: Man Image: Man Image: Man Image: Man Image: Man Image: Man Image: Man Image: Man Image: Man Image: Man Ima Image: Man	DeadBand	DeadBand	REAL	1		
Yman Yman REAL 1 Y Y REAL 1 Status INT 1	📝 Man	Man	BOOL	1		
☑ Y Y REAL 1 ☑ Status Status INT 1	📝 Yman	Yman	REAL	1		
Status Status INT 1	🖸 Y	Υ	REAL	1		
CARD INT CARD INT DINT 1	Status	Status	INT	1		
	temp_INT	tEMP_INT	DINT	1		
wq wq REAL 1	🗹 wq	wq	REAL	1		
- 🗋 i INT 1	i i	i	INT	1		
- 🖓 j INT 1	🚺 j	j	INT	1		
Connect Connect ETH_PARAM 1	🗈 🔽 connect	connect	ETH_PARAM	1		
Deconnect status INT 1	Connect status	connect status	INT	1		
	Vedeble Teble					



4.2.4 Таблица дополнительных точек

В режиме онлайн просмотреть / изменить значение точки можно с помощью таблицы дополнительных точек. Таблица дополнительных точек позволяет свободно

определять желаемые точки, причем таким образом, чтобы при отладке программы можно было наблюдать только за желаемыми точками. В браузере проекта выберите

[Data] / **[Optional Point Table]**, откроется таблица дополнительных точек. Если она применяется в первый раз, то эта таблица точек будет пустой, как показано на рисунке 4.4:

Enter	Value	Address	Name	Force	Туре	Comment	No.	

Рисунок 4.4 Пустая таблица дополнительных точек

Добавление точки

Щелкните правой кнопкой мыши на пустом месте, чтобы выбрать команду **Add**, откроется диалоговое окно **Point Name**, в котором нужно вести название точки, как показано на рисунке 4.5:

Enter	Valua	Address	Name	Force	Туре	Comment	No	
SetPoint	Value	%MW0601	Name	TUICE	REAL	comment	1	
SEPOINT		7510100001			NLAL			
		Add						
		Delete						
		Edit						
		Add fro	om Program					
		Clear	-					
•								
	P	oint Name				×		
		Point N	lame					
			ок	C	ancel			

Рисунок 4.5 Добавление точки

Добавление из программы

Щелкните правой кнопкой мыши в таблице дополнительных точек, чтобы выбрать команду *Add from Program*, откроется диалоговое окно *Select Program*, выберите название добавляемой программы и нажмите *OK*, затем все точки и переменные, используемые в этой программе, будут добавлены в открытую таблицу дополнительных точек, как показано на рисунке 4.6:

Enter	Value	Number	Name	Force	No.	
		Add				
		Delete				
		Edit				
		Add from Program				
		Clear				

Select Program	×
All Pr	ograms
	-
MAIN	
000	
ОК	Cancel

Рисунок 4.6 Добавление точки из программы

После добавления точек, информацию о значении точки, количестве, названии, принудительном вводе и т.д. можно просмотреть в таблице дополнительных точек. Как показано на рис.4.7:

Enter 🛆	Value	Address	Name	Force	Туре	Comment	No.
%M3003		%M3003	Valve_1_OpOpen		BOOL	Оператор - Отк	1
%M3004		%M3004	Valve_1_OpClose		BOOL	Оператор - зак	2
%M3008		%M3008	Valve_1_OpReset		BOOL	Сброс ошибки	3
%MW5000		%MW5000			WORD		30
%T0001		%T0001			WORD		31
- avr_work		%M6004	avr_work		BOOL		29
OK2024_1_4		%MW20111	OK2024_1_40003		WORD	Включение/вы	6
walve_1_ops		%M3005	Valve_1_OpStop		BOOL	оператор - стоп	12
		%V9110			BOOL	Закрыта	10
		%V9112			BOOL		8
		%V9109			BOOL	Открыта	9
		%V9111			BOOL		7
Vlv[0].InOut		%V9160			BOOL	оператор - стоп	11
Vlv[0].InOut		%V9172			BOOL		17
Vlv[0].InOut		%V9181			BOOL		23
Vlv[0].InOut		%V9178			BOOL		20
		%V9185			BOOL		27
		%V9184			BOOL		26
Vlv[0].InOut		%V9183			BOOL		25
Vlv[0].InOut		%V9186			BOOL		28
Vlv[0].InOut		%V9171			BOOL	Внутренняя ко	16
Vlv[0].InOut		%V9180			BOOL		22
Vlv[0].InOut		%V9177			BOOL		19
		0/1/0470	1		POO!		10
Uptional Point	lable						

Рисунок 4.7 Таблица дополнительных точек

Значение принудительного ввода точки

Используется для изменения значения точки в таблице дополнительных точек. Для точек %М, %МW, %N, %NM, %Q, %QW и переменных, значение точки можно изменить, дважды щелкнув значение точки, которое необходимо изменить в столбце *Value*; Для точек %I и %IW значения точек могут быть изменены в столбце *Value* только после того, как опция принудительного ввода будет отмечена двойным щелчком по столбцу *Force*.

Удаление и редактирование точки

Если некоторые точки не нужны, их можно удалить или изменить, чтобы они стали другими. Операцию можно выбрать, щелкнув правой кнопкой мыши по соответствующему столбцу, как показано на рисунке 4.8:

Руководство пользователя Программное обеспечение «СКПро»

Enter		Value	Address	Name
; %M3004	ļ		%M3004	Valve_1_OpClose
%M3008	3		0/1 (2000	Valve_1_OpReset
%MW50	00	Add		
%T0001		Delete		
- avr_worl	c	Edit		avr_work
- OK2024_	1_4		1	OK2024_1_40003
w valve_1_	ops	Add from Prog	ram	Valve_1_OpStop
	.FB.	Clear		
	.FB		%V9112	
E 1/1 101 L	FD		0/1/0100	ĺ

Рис.4.8 Удаление и редактирование точки

Добавление таблицы дополнительных точек

Пользователи могут создать не более 16 листов таблицы дополнительных точек. На вкладке **[Data]** браузера проекта щелкните правой кнопкой мыши, чтобы выбрать команду *Add Optional Point Table*, как показано на рисунке 4.9:



Рисунок 4.9 Добавление таблицы дополнительных точек

Удаление таблицы дополнительных точек

Если вы не хотите сохранять новую таблицу точек, а хотите удалить ее, вам нужно очистить таблицу точек, затем закрыть окно. Система автоматически удалит эту таблицу дополнительных точек, как показано на рисунке 4.10:

Enter		Value	Add
%M300	4		%N
%N	Add		%№
··· %I	Delet	te	ŝMi
%1	Edit		%Т
avı	Luit		%N
OK	Add	from Program	MV
val	Clear	r	%№
···· VIV			%V
 VP1011 	n ED		9/1/

Рисунок 4.10 Очистка точки

Таблица дополнительных точек сохраняется на текущей инженерной станции для отладки, а затем при следующем подключении, таблица дополнительных точек

сохранит статус, существовавший до последнего выхода. Таблица дополнительных точек никогда не загружается в ПЛК.

4.3 Режим адресации

Адресация — это метод доступа к точкам данных для языка LD и других языков программирования. Существует несколько типов режимов адресации:

Константа

Константа — это способ адресации неизменямых данных. Для следующего функционального блока функция выполнения **MOVE**, заключается в присвоении значения 100 точке %MW0001, в таком случае ввод значения 100 является режимом ввода константы.



Рисунок 4.11 Константа

Указание

В режиме ввода константы, при вводе данных после шестнадцатеричных данных должна быть буква "Н", если первое число шестнадцатеричных данных является буквой, то перед буквой добавляется "0". Например, если вы хотите ввести В021Н, то записать необходимо 0В021Н, в противном случае, при компиляции системы, возникает ошибка.

Прямая адресация

Тип точки, добавляющий номер точки, является объектом доступа прямой адресации. Как показано на рис.4.11, для операции **MOVE** %MW0001 является указанной точкой, это режим прямой адресации.

В качестве особого примера, использование имени точки также относится к прямой адресации, поскольку имя точки соответствует указанной точке. Например, присвойте точке %l0001 имя "GD_ON", в таком случае имя "GD_ON" может быть использован непосредственно в программе, так как оно ссылается на %l0001.

Косвенная адресация

Номер точки объекта доступа для косвенной адресации является не константой, а другой точкой, то есть в качестве номера точки используется текущее значение этой точки. Отличие от номера точки прямой адресации в следующем: точка косвенной адресации начинается с 0, то есть, в случае косвенной адресации точки %[%МW0001]

текущее значение %МW0001 равно 1, а точка доступа - %l0002, вместо %l0001. Как показано на puc.4.12, %Q[%MW0001] — это режим косвенной адресации. Тип точки - Q, номер точки не является константой, а другой точкой - %MW0001. Операция выполнения LD такова: %MW0001 присваивается как 0, катушка выдает 1 для выходной точки %Q0001.



Рисунок 4.12 Косвенная адресация

5 Базовые блоки функций и программирование на языках, поддерживаемых ПО «СКПро»

Базовые блоки функций и программирование на языках, поддерживаемых программным обеспечением «СКПро» подробно рассмотрены в системном руководстве по программному обеспечению «СКпро».

6 Загрузка и выгрузка файла проекта «СКПро»

6.1 Ручная загрузка

Ручная загрузка часто используется для загрузки файла проекта и всех программных файлов в первый раз, либо если неизвестен IP-адрес ПЛК в сети Ethernet. Ручная загрузка загрузит все файлы проекта и программные файлы в ПЛК.

Убедитесь, что ПЛК находится в выключенном состоянии. После этого сначала поверните переключатель режима работы ПЛК в положение "*Debug*", а затем включите питание ПЛК.

После успешного запуска, индикаторы "R/F" на панели ПЛК будет мигать некоторое время (около 1 секунды), указывая на то, что ПЛК работает в состоянии отладки. В этом состоянии IP-адрес ПЛК установлен как 192.168.1.66.

Затем измените IP-адрес в сетевом подключении компьютера, чтобы IP-адрес ПЛК находился в том же сегменте сети, как показано на рисунке 6.1:

Свойства: IP версии 4 (TCP/IPv4)	×
Общие	
Параметры IP можно назначать авто поддерживает эту возможность. В п параметры IP у сетевого администр	матически, если сеть ротивном случае узнайте атора.
Получить IP-адрес автоматиче	ски
Оспользовать следующий IP-ад	apec:
ІР-адрес:	192 . 168 . 1 . 254
Маска подсети:	255 . 255 . 255 . 0
Основной шлюз:	
🔿 Получить адрес DNS-сервера а	втоматически
• Использовать следующие адре	ca DNS-серверов:
Предпочитаемый DNS-сервер:	
Альтернативный DNS-сервер:	
Подтвердить параметры при в	ыходе Дополнительно
	ОК Отмена

Рисунок 6.1 Настройка ІР-адреса ПК

Затем запустите среду программирования «СКПро» и после загрузки выберите в меню **[Load]** пункт **[Manual Download]** (как показано на рисунке 6.2):



Рисунок 6.2 Пункт Manual Download в меню Load

Откроется следующее диалоговое окно, приведённое на рисунке 6.3:

Ethernet IP	×
Ethernet IP	192 .168.1.66
ОК	Cancel

Рисунок 6.3 Загрузка проекта вручную

Нажмите кнопку ОК, чтобы загрузить файл проекта. После этого откроется следующее окно, показанное на рисунке 6.4 и свидетельствующее о загрузке проекта:

Рисунок 6.4 Загрузка проекта

Если нет подключения ПЛК по сети, либо адрес указан неверно, то появится следующее окно запроса, приведённое рисунке 6.5:



Рисунок 6.5 Ошибка загрузки

В таком случае необходимо проверить сетевую конфигурацию ПК.

Вы можете проверить доступность ПЛК по сети используя интерфейс командной строки. Для этого запустите программу «Командная строка», либо выполните команду cmd в штатном поиске Windows. В появившемся окне введите команду "PING 192.168.1.66 –t".

Команда ping – это утилита командной строки, которая нужна для проверки подключения к другому устройству на уровне IP. Принцип работы очень простой: команда ping ip отправляет серию небольших пакетов данных на указанное устройство, а затем показывает время ответа. Параметр /t команды указывает, что команда ping продолжает отправлять пакеты запросов по месту назначения, пока не будет прервана. Чтобы прервать и отобразить статистику, нажмите сочетание клавиш CTRL + Break. Чтобы прервать выполнение и выйти из этой команды, нажмите клавиши CTRL + C.

Если во время выполнения команды ping связь с ПЛК имеется, но программа не загружается, пожалуйста, в этом случае проверьте настройки брандмауэра компьютера, чтобы узнать, не заблокирована ли передача файлов или не занесен ли программный пакет «СКПро» в черный список. В этом случае Вы можете отключить брандмауэр и повторить попытку.

После успешной загрузки переведите переключатель режима работы на ПЛК в состояние "Run", и ЦПУ снова будет включено

6.2 Загрузка проекта

Скомпилируйте и загрузите проект в ПЛК выбрав в меню **[Load]** пункт **[Download Project]**. При загрузке "СКПро" автоматически выполнит поиск ПЛК в сети в соответствии с адресом Ethernet в конфигурации оборудования и загрузит его. В процессе поиска и загрузки "СКПро" выведет "*Download*". Если поиск ПЛК не удается выполнить, будет сообщено об ошибке подключения. Для резервированной системы "СКПро" проект будет автоматически загружен на оба ПЛК.

После загрузки программы в ПЛК модуль ПЛК должен быть сброшен и перезапущен для выполнения загруженной программы. В противном случае система выполнит программу перед загрузкой. Сброс может быть завершен командой сброса "СКПро". *Руководство пользователя Программное обеспечение «СКПро»* 92

6.3 Выгрузка проекта

Для выгрузки проекта с ПЛК на компьютер используйте пункт **(Upload Project)** выбрав его в меню **(Load)**. В отличие от загрузки, при выгрузке "СКПро" отобразит диалоговое окно с запросом ввести Ethernet-адрес ПЛК, где находится проект, который Вы хотите выгрузить. Нажмите клавишу "ОК" после ввода, и "СКПро" выполнит поиск узла в сети в соответствии с адресом, как показано на рис. 6.6:

Ethernet IP	x
Ethernet IP 192 .168.1.66	
OK Cancel	

Рисунок 6.6 Выгрузка проекта

После выгрузки "СКПро" предложит вам ввести название проекта. Вы можете перезаписать файл с тем же именем или дать иное имя.

Примечание: если " *Allow to upload*" не выбрано в окне *Hardware Configuration* меню "СКПро" "Resource" / "PLC configuration" при конфигурировании аппаратного обеспечения проекта (см. рисунок 6.7), то файл проекта и файл программы не будут выгружены, и будет отображено предупреждение "Project file upload failed!".

File Edit View Online Load Wind	dow Help	
D 🖬 D 🖪 🖉 🕸 X 🖻 🖻	그 그 # 🙀 🔽 🖪 🖩 📾 📾 📾 🖉 🊔 ଅ 🖉 🖊 🏠 📭 🕼 🕅	1 8 8
Resource PLC Configuration Rack 1 (BK-4000-0 Rack 2 (BK-4000-0 I) O Channel Redun Task Configuration Interrupt Configuration MODBUS/TCP Master	Hardware Configuration Series © SC-4000 © SC-1000 Rack Redundancy © Bus © 10 Rack 1 BK-4000-0601 Rack 5 None Rack 2 BK-4000-0601 Rack 6 None Rack 3 None Rack 7 None Rack 4 None Rack 8 None ØK Cancel	×

Рисунок 6.7 Разрешение на выгрузку проекта

7 Отладка программы

ПО «СКПро» поддерживает множество языков программирования, таких как LD, FBD, IL, ST и SCC. В соответствии с характеристиками каждого языка для «СКПро» разработаны различные методы отладки, чтобы помочь специалистам модифицировать и дорабатывать программы в ходе проекта. Существует две среды отладки: отладка в режиме работающего ПЛК и отладка в симуляторе. Для отладки с ПЛК в онлайн-режиме, «СКПро» напрямую подключается к ПЛК, и изменения загружаются в него напрямую. Симулятор имитирует среду, в которой «СКПро» и ПЛК связаны виртуально. В нем выполняются предварительные тесты корректности выполнения программы посредством создания переменных, форсирования точек ввода-вывода и т.д., что обеспечивает успех дальнейшей отладки и безопасность эксплуатации оборудования. Ниже приведены иллюстрации шагов отладки для различных языков программирования.

7.1 Отладка LD / FBD

7.1.1 Онлайн-модификация ПЛК

Отладка LD/FBD интуитивно понятна. После включения все проводящие контуры становятся красными (%М0008), непроводящие - зелеными (%М0007), как показано на рисунке 7.1. Программа LD обычно используется для оценки состояния, поэтому в ней отображается непосредственно состояние точки ввода-вывода.



Рисунок 7.1 LD в режиме онлайн

После подключения, модификация LD/FBD может быть так же выполнена, как показано на рис.7.2.



Рисунок 7.2 Модификация параметров

После подключения и внесения изменений в программу к названию программы в браузере проекта будет добавлен знак «*», например «MAIN*», как показано на рисунке 6.3. Это указывает на то, что текущая программа была изменена, но еще не была загружена.



Рисунок 7.3 Программа изменена, но не загружена в ПЛК

После завершения модификации выберите значок в системной панели инструментов, чтобы сохранить измененный файл, затем выберите **[Online] / [Refresh Program]**, как показано на рисунок 6.4, выгрузите измененную программу в ПЛК.

<u>O</u> nline	Loa <u>d</u>	<u>W</u> indow	Hel				
Login							
Logou	Log <u>o</u> ut						
Conne	Connect •						
Di <u>s</u> connect							
Displ	Display <u>F</u> ormat 🕨 🕨						
Refresh Progra <u>m</u>							
<u>U</u> nforce							
<u>R</u> eset							
Set <u>1</u>	Set <u>T</u> ime						
Maste	er/Slav	ze S <u>w</u> itch	1				

Рисунок 7.4 Обновить программу

7.1.2 Отладка с включенным ПЛК

Функция онлайн-отладки обычно используется при одноэтапной отладке программы. Точка прерывания может быть установлена в любом функциональном блоке LD/FBD во время отладки. Когда программа доходит до точки прерывания, она останавливается и ожидает выполнения **шага (Step)** инструкции. Если отладка программы завершена, очистите все точки прерывания и нажмите кнопку *Continue*, и программа продолжит обычное сканирование.

• Шаг (Step) 🔂

При отладке программы каждый блок функции представляет собой один шаг, поэтому программа останавливается в исходном месте после завершения каждого блока функции и ожидает инструкции **Step**. Инструкции **Step** отправляются следующим образом: наведите указатель мыши на кнопку **Step** на панели инструментов LD/FBD, а затем нажмите левую кнопку, чтобы отправить эту инструкцию.

• Продолжить (Continue) 🗈

Когда выполнение программы остановится, нажмите кнопку **Continue** на панели инструментов LD/FBD, после чего программа продолжит выполнение. Если в программе есть точки останова, то программа остановится, дойдя до первой точки останова и будет ждать инструкции **Step**. Если в программе нет точек останова, то программа будет выполняться непрерывно.

• Вставка/Удаление точки останова 🖑

Точка останова может быть установлена в любом функциональном блоке программ. Устанавливается точка останова следующим образом: выберите функциональный блок, в котором необходимо установить точку останова (наведите мышь на функциональный блок и нажмите левую кнопку мыши, чтобы выбрать его), затем наведите курсор мыши на кнопку *Insert/Remove Breakpoint* на панели инструментов LD/FBD, нажмите левую кнопку мыши, чтобы в правом верхнем углу выбранного блока функции появилась зеленая точка, это указывает на то, что точка останова успешно установлена. Если снова нажать кнопку *Insert/Remove Breakpoint* на панели инструментов LD/FBD, то точка останова будет удалена. Если нажать кнопку в третий раз, то точка останова будет установлена снова. Программа может одновременно установить до 10 точек останова в разных местах, как показано на рис. 7.5:



Рисунок 7.5 Установка точки останова в LD

• Очистить все точки останова 🇏

Все точки останова в программе могут быть очищены одновременно. Чтобы запустить очистку, наведите курсор мыши на кнопку *Clear All Breakpoints* на панели инструментов LD/FBD и нажмите левую кнопку, после чего все точки останова будут удалены.

Указание

После выполнения все точки останова будут автоматически удалены.

7.2 Отладка SCC

После того, как программа SCC скомпилирована и загружена, ее необходимо отладить. Перед отладкой систему необходимо подключить к ПЛК или симулятору. В этом случае можно начать использовать панель инструментов SCC, и могут быть отправлены инструкции по отладке.

7.2.1 Автоматическое выполнение

Во время работы в режиме автоматического выполнения **П**ЛК не будет выгружать информацию в процессе выполнения программы, и процесс выполнения программы в данный момент не может быть отражен в режиме реального времени на компьютере для отладки. Чтобы запустить автоматический режим, нажмите кнопку **Automatic** на панели инструментов SCC.

7.2.2 Наблюдение за выполнением

В режиме наблюдения за выполнением ⁶⁴² ПЛК загружает результат выполнения программы в режиме реального времени и процесс выполнения программы можно наблюдать на компьютере для отладки (только на компьютере, который отправил инструкции по отладке). В этом режиме выполнения отладочный персонал может отслеживать и управлять процессом выполнения программы в режиме реального времени и анализировать условия работы различных устройств (это может быть отражено в программе). В процессе выполнения программы, запущенный отображается функциональный элемент красным цветом, выполненные функциональные элементы отображаются синим цветом, а невыполненные - серым цветом. Чтобы запустить режим наблюдения, наведите курсор мыши на кнопку *Watching* на панели инструментов SCC и нажмите левую кнопку. В случае вызова подпрограммы система автоматически переключится на окно подпрограммы и будет следить за выполнением подпрограммы. После завершения подпрограммы система автоматически вернется к окну текущей программы, как показано на рисунке 7.6:



Рисунок 7.6 Наблюдение за выполнением SCC

7.2.3 Остановить выполнение

Когда наблюдение за выполнением программы находится в определенной точке, вы можете отправить инструкцию **Stop**, если хотите остановить выполнение программы. Чтобы остановить наблюдение, наведите курсор мыши на кнопку **Stop** на панели инструментов SCC и нажмите левую кнопку, после чего будет отправлена инструкция **Stop**. ПЛК остановит выполнение программы после получения инструкции.

7.2.4 Выполнение отладки

В отличие от просмотра выполнения, режим отладки выполнения ^В больше подходит для отладки программ и поддерживает множество методов. Чтобы запустить отладку, выберите кнопку **Debugging** на панели инструментов SCC и нажмите левую кнопку мыши, затем отправляется инструкция **Debugging**, и программа выполняется. Однако, выполнение программы пошаговое, и после завершения первого шага программа остановится и будет ждать инструкции **Step**.

• Шаг (Step) 🔂

Когда программа находится в режиме выполнения одноступенчатой отладки, каждый функциональный элемент блок-схемы представляет собой один шаг. Программа останавливается в исходном положении и ждет инструкцию **Step** после завершения одного шага. Чтобы отправить инструкцию, наведите курсор мыши на кнопку **Step** и нажмите левую кнопку мыши, после чего инструкция будет отправлена.

•Вставка/Удаление точки останова (Insert/Remove Breakpoint) 🖑

Точка останова может быть установлена в любом функциональном блоке программ. Устанавливается точка останова следующим образом: выберите функциональный блок, в котором необходимо установить точку останова (наведите мышь на функциональный блок и нажмите левую кнопку мыши, чтобы выбрать его), затем наведите курсор мыши на кнопку *Insert/Remove Breakpoint* на панели инструментов LD/FBD, нажмите левую кнопку мыши, чтобы на выбранном функциональном блоке появилась зеленая точка, это указывает на то, что точка останова успешно установлена. Если снова нажать кнопку *Insert/Remove Breakpoint* на панели инструментов SCC, то точка останова будет удалена. Если нажать кнопку в третий раз, то точка останова будет установлена снова. Программа может одновременно установить до 10 точек останова в разных местах, как показано на рисунке 7.7:



Рисунок 7.7 Установка точки останова в SCC

• Продолжить (Continue) 🗈

Находясь в режиме пошагового выполнения, нажмите кнопку *Continue* на панели инструментов SCC, после чего программа продолжит выполнение. Если в программе есть точки останова, то программа остановится, дойдя до первой точки останова и

будет ждать инструкции **Step**. Если в программе нет точек останова, то программа будет выполняться до завершения.

• Очистить все точки останова 🌋

Все точки останова в программе могут быть очищены одновременно в режиме пошагового выполнения. Чтобы запустить очистку, наведите курсор мыши на кнопку *Clear All Breakpoints* на панели инструментов SCC и нажмите левую кнопку, после чего все точки останова будут удалены.

Указание

После выполнения все точки останова будут автоматически удалены.

• Перезапуск 🗊

Когда в режиме пошагового выполнения программа выполнена наполовину и ее требуется перезапустить, есть два способа сделать это:

(1) Нажмите кнопку **Restart** на панели инструментов SCC, и программа будет запущена заново.

(2) Сначала нажмите кнопку *Stop Debugging* на панели инструментов SCC, чтобы выполнить операцию **Stop debugging**. Затем, чтобы запустить программу и перезапустить выполнение с самого начала, выберите метод отладки.

• Прекратить отладку 🎽

В режиме выполнения отладки нажмите кнопку *Stop Debugging*, и выполнение отладки программы будет остановлено.

7.2.5 Блокировка и разблокировка

С помощью кнопок *Lock* lock kak показано на рисунке 7.8.



Рисунок 7.8 Блокировка SCC

7.3 Отладка IL

Функция отладки в реальном времени в основном используется при пошаговой отладке программы. В процессе отладки точка останова может быть установлена в любой допустимой строке IL. Программа остановится при достижении точки останова и будет ждать инструкции **Step**. Если отладка завершена, все точки останова будут

очищены. Нажмите на кнопку *Continue*, чтобы программа продолжила обычное сканирование.

• Шаг (Step) 🔂

При отладке программы каждая строка кода представляет собой один шаг. Программа останавливается в исходном положении и ждет инструкцию **Step** после завершения одного шага. Чтобы отправить инструкцию **Step**, наведите курсор мыши на кнопку *Step* на панели инструментов IL и нажмите левую кнопку мыши, после чего инструкция **Step** будет отправлена.

• Продолжить (Continue) 🗈

Находясь в режиме пошагового выполнения, нажмите кнопку *Continue* на панели инструментов IL, чтобы продолжить выполнение программы. Если в программе есть точки останова, то программа остановится, дойдя до первой точки останова и будет ждать инструкции **Step**. Если в программе нет точек останова, то программа будет выполняться постоянно.

• Вставка/Удаление точки останова 🖑

Точку останова можно установить в любой допустимой строке программы. Способ установки точки останова: наведите курсор на строку, где нужно установить точку останова, затем нажмите левой кнопкой мыши на кнопку *Insert/Remove Breakpoint* на панели инструментов IL. Строка загорится зеленым, что означает, что точка останова успешно установлена. Если снова нажать кнопку *Insert/Remove Breakpoint*, то точка останова будет удалена. Если нажать кнопку в третий раз, то точка останова будет установа. Программа может одновременно установить до 10 точек останова в разных местах, как показано на рисунке 7.9:

() DD		
LD	%IW0001	
AND	%MW0001	
ST	%QW0001	

Рисунок 7.9 Установка точки останова в IL

• Очистить все точки останова 🌋

Все точки останова в программе могут быть очищены одновременно. Чтобы запустить очистку, нажмите левой кнопкой мыши на кнопку *Clear All Breakpoints* на панели инструментов IL после чего все точки останова будут удалены.

Указание

После отключения все точки останова удаляются автоматически.

7.4 Отладка ST

Функция отладки в реальном времени в основном используется при пошаговой отладке программы. В процессе отладки точка останова может быть установлена в любой допустимой строке ST. Программа остановится при достижении точки останова и будет ждать инструкции **Step**. Если отладка завершена, все точки останова будут очищены. Нажмите на кнопку **Continue**, чтобы программа продолжила обычное сканирование.

• Шаг (Step) 🔂

При отладке программы каждая строка кода представляет собой один шаг. Программа останавливается в исходном положении и ждет инструкцию **Step** после завершения одного шага. Чтобы отправить инструкцию **Step**, наведите курсор мыши на кнопку *Step* на панели инструментов ST и нажмите левую кнопку мыши, после чего инструкция **Step** будет отправлена.

• Продолжить (Continue) 🗈

Когда выполнение программы остановится, нажмите кнопку **Continue** на панели инструментов ST, после чего программа продолжит выполнение. Если в программе есть точки останова, то программа остановится, дойдя до первой точки останова и будет ждать инструкции **Step**. Если в программе нет точек останова, то программа будет выполняться постоянно.

• Вставка/Удаление точки останова 🕙

Точку останова можно установить в любой допустимой строке программы. Способ установки точки останова: наведите курсор на строку, где нужно установить точку останова, затем нажмите левой кнопкой мыши на кнопку *Insert/Remove Breakpoint* на панели инструментов ST. Строка загорится зеленым, что означает, что точка останова успешно установлена. Как показано на рисунке 10.10. Если снова нажать кнопку *Insert/Remove Breakpoint*, то точка останова будет удалена. Если нажать кнопку в третий раз, то точка останова будет установлена снова. Программа может одновременно установить до 10 точек останова в разных местах. Как показано на рисунке 7.10:

FOR i := 1 TO II PCNT BY	1 D0	1 D0	8Y 1 D	T BY	PENT	II	TO	1		i	OR
IF II_BUF[i]=1 THEN			EN	THEN	TI	1	[i]=	JF [_BL	II	IF

Рисунок 7.10 Установка точки останова в ST

• Очистить все точки прерывания 🌋

Все точки останова в программе могут быть очищены одновременно. Чтобы запустить очистку, нажмите левой кнопкой мыши на кнопку **Clear All Breakpoints** на панели инструментов ST после чего все точки останова будут удалены.

Указание

После отключения все точки останова удаляются автоматически.

7.5 Симулятор

ПО «СКПро» предоставляет два режима отладки для тестирования готовых программ: в режиме подключенного онлайн ПЛК и в симуляторе в режиме реального времени.

При использовании симулятора система работает в виртуальном режиме, то есть «СКПро» не подключается к ПЛК и выполняет виртуальную отладку готовой программы ПЛК в режиме реального времени или выполняет симуляционный тест без реального ПЛК, а также моделирует производственный процесс на реальном испытательном участке и тестирует готовые программы, чтобы предоставить возможность просмотра результата выполнения текущих программ и их корректности.

При применении режима подключенного онлайн ПЛК «СКПро» подключается к самому ПЛК. Используя коммуникации и встроенные возможности мониторинга в режиме реального времени, на основе фактических или аналоговых сигналов, а также процессах управления, система может осуществлять мониторинг и отладку отредактированной программы в режиме реального времени, что позволяет просматривать результаты выполнения текущих программ и их корректность.

Различия между режимом подключенного онлайн ПЛК и симулятором заключаются в следующем:

В симуляторе в режиме реального времени «СКПро» не связывается с ПЛК, принудительный ввод и модификация всех сигналов происходят только на компьютере, где находится «СКПро». Результаты операции или данные выводятся только на компьютере, однако он не осуществляет реального управления процессом.

В режиме подключенного онлайн ПЛК, «СКПро» связывается с ПЛК, форсирование и модификация всех сигналов происходят в реальном ПЛК. Результаты работы или данные выводятся с ПЛК и могут управлять реальными процессами.

Симулятор можно запустить с помощью меню **[Online] / [Connect] / [Simulator]**, а режим подключенного онлайн ПЛК можно запустить с помощью меню **[Online] / [Connect] / [PLC]**. Выйти из обоих режимов можно с помощью меню **[Online] / [Disconnect]**.

Оба режима позволяют просматривать результаты выполнения текущих программ и их корректность. В целом, симулятор подходит для этапа предварительного

проектирования, в то время как режим подключенного онлайн ПЛК подходит для этапа отладки на заводе или в полевых условиях после продажи.

Основные функции симулятора приведены ниже:

Он может принудительно вводить все сигналы ввода и вывода и изменять их значение.

Он может полностью симулировать фактические результаты выполнения логических связей, инструкций и программ.

8 Приложение

8.1 Сервис и поддержка

8.1.1 Контакты службы технической поддержки

Запрос по телефону

Обращение по телефону службы технической поддержки: +7 (347) 223-99-22

Запрос по электронной почте

Обращение на адрес электронной почты службы технической поддержки: support@sybcom.ru

Запрос через интернет

Обращение через заполнение формы на сайте технической поддержки. www.sybcom.ru/support

8.1.2 Порядок оказания технической поддержки по изделию

Пользователь обращается в службу технической поддержки путем формирования обращения с указанием идентифицирующей пользователя информации и описанием возникшей проблемы.

При обращении пользователя в службу технической поддержки фиксируются его контактные данные (Фамилия, Имя, Отчество (при наличии), место работы и должность, адрес электронной почты в корпоративном домене, телефон), описываются причины обращения. В случае отказа Пользователя сообщить идентифицирующую его информацию, Сотрудник Службы технический поддержки имеет право не оказывать такому Пользователю услуг по технической поддержке.

Обращения в службу технической поддержки регистрируются в виде заявки с присвоением уникального номера. Подтверждением регистрации обращения для его инициатора служит номер заявки, передаваемый техническими средствами, входящими в инструментарий службы технической поддержки. Указание пользователем номера исходной заявки при повторных обращениях позволяет сотрудникам службы технической поддержки оперативно коммуницировать с пользователем.

Пользователь принимает на себя обязанность своевременного и квалифицированного взаимодействия со службой технической поддержки в соответствии с настоящим регламентом. При необходимости он самостоятельно информирует других пользователей, действующих в интересах того же юридического лица, о статусе обращения или делегирует им работу с обращением, уведомив об этом службу технической поддержки.

8.2 Лист изменений

Версия	Дата	Изменение			
V1.0	12/2022	Первое издание для версии «СКПро» 6.2.11			
V1.1	01/2023	Стилистические правки.			
V1.1.1		Исправлены ссылки на рисунки в разделах. Грамматические			
		правки.			
V1.1.2		Добавлена информация о настройке цикла опроса для модулей			
		AI-4000-0801, AI-4000-0802, AI-4000-1601, AI-4000-0804			