

Основы работы с ПО для ПЛК GX Works2

Настоящий учебный курс (курс электронного обучения) предназначен для тех, кто впервые использует программное обеспечение GX Works2 для создания программ ПЛК

Введение**Цель курса**

В этом курсе приводятся основные сведения по использованию программного обеспечения GX Works2 для программирования, отладки и проверки работы программируемого контроллера (ПЛК). Курс предназначен для пользователей, создающих программы для контроллеров серии MELSEC-Q, MELSEC-L и MELSEC-F.

Данный курс имеет следующее содержание.
Рекомендуем вам начать с Главы 1.

Глава 1. Методика управления системой ПЛК

Здесь рассматриваются языки программирования и соответствующее программное обеспечение.

Глава 2. Проектирование программ

Вы узнаете, как спроектировать программу исходя из параметров ПЛК и конфигурации оборудования.

Глава 3. Программирование

Из этой главы вы узнаете, как писать программы с помощью специализированного программного обеспечения GX Works2.

Глава 4. Отладка

Вы узнаете, как записывать программы в модуль ЦП и выполнять их отладку.

Глава 5. Заключительный тест

Проходной балл: 60% или выше.

Введение**Как использовать этот инструмент электронного обучения**

Переход к следующей странице		Переход к следующей странице.
Возврат к предыдущей странице		Возврат к предыдущей странице.
Переход к требуемой странице		Появится экран «Содержание», на котором вы сможете перейти к требуемой странице.
Завершение обучения.		Завершение обучения. Окно (например, «Содержание») будет закрыто, а обучение — завершено.

Меры безопасности

Если при обучении используются реальные продукты, внимательно прочтите меры безопасности в соответствующих инструкциях к ним.

Меры предосторожности относительно данного курса

- Отображаемые экраны с версией используемого программного обеспечения могут отличаться от тех, которые используются в данном курсе.

Глава 1

Методика управления системой ПЛК

Этот курс предназначен для технических специалистов, работающих с программным обеспечением. Он охватывает некоторые фундаментальные принципы управления системами серии MELSEC-Q, L и F.

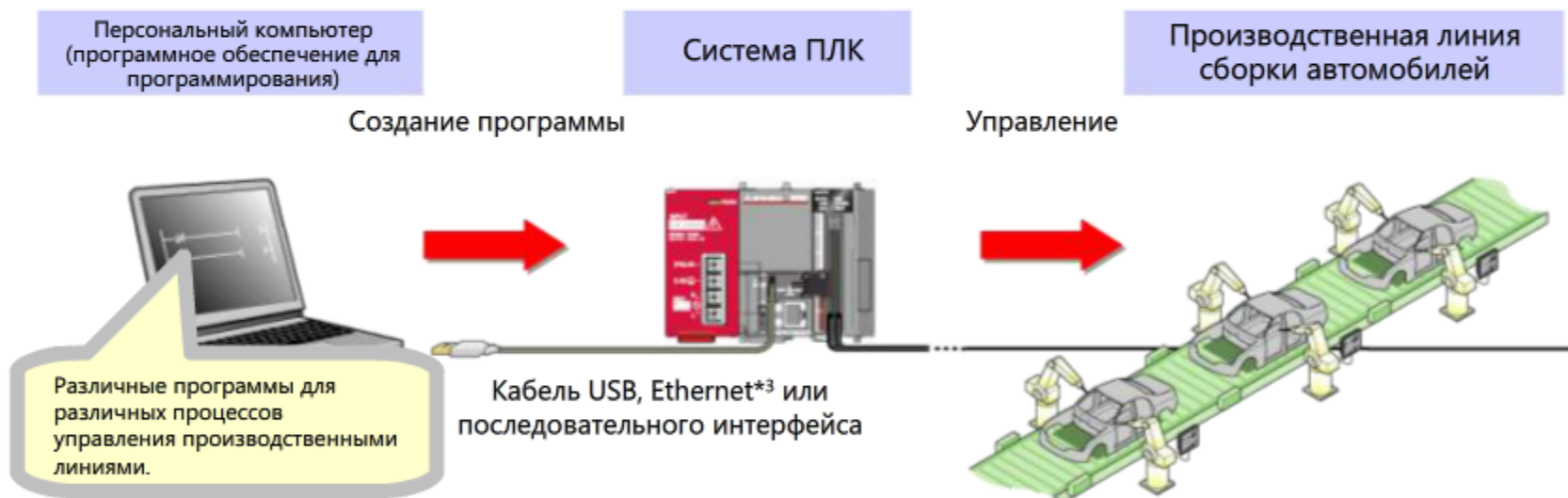
В ПО GX Works 2 (GXW2) используются языки программирования, соответствующие международным стандартам. Это могут быть следующие языки: язык последовательных функциональных схем (SFC) язык, список инструкций (IL)*1, язык релейно-лестничной логики (LD), язык диаграмм функциональных блоков (FBD)*2 и структурированный текст (ST).

Созданные программы записываются в ЦП программируемого контроллера по USB, кабелю Ethernet*3 или кабелю последовательного интерфейса. Модуль ЦП может быть перепрограммирован любое количество раз для внесения необходимых изменений в нужные части программ, параметров ПЛК и т.д.

*1 Планируется реализовать в будущих версиях GX Works2.

*2 В настоящее время в GX Works2 носит название Structured Ladder (Структурированная лестница), планируется подтверждение соответствия со стандартами МЭК.

*3 Ethernet является зарегистрированной торговой маркой Xerox Corp.



В этом курсе в качестве примера используется язык релейной логики (один из самых популярных языков программирования для ПЛК). Несмотря на то, что примеры приведены для ПЛК серии L, содержимое курса в равной степени относится и к системам серии Q.

В целом методика управления подходит и для контроллеров серии MELSEC-F, но некоторые операции и функции отличаются.

1.1 Порядок создания программ для ПЛК

Этот курс электронного обучения охватывает этапы подготовки системы программируемого контроллера, связанные с программным обеспечением (выделены зеленым на схеме).

Аппаратная часть



Программная часть



**Предмет
настоящего
курса**

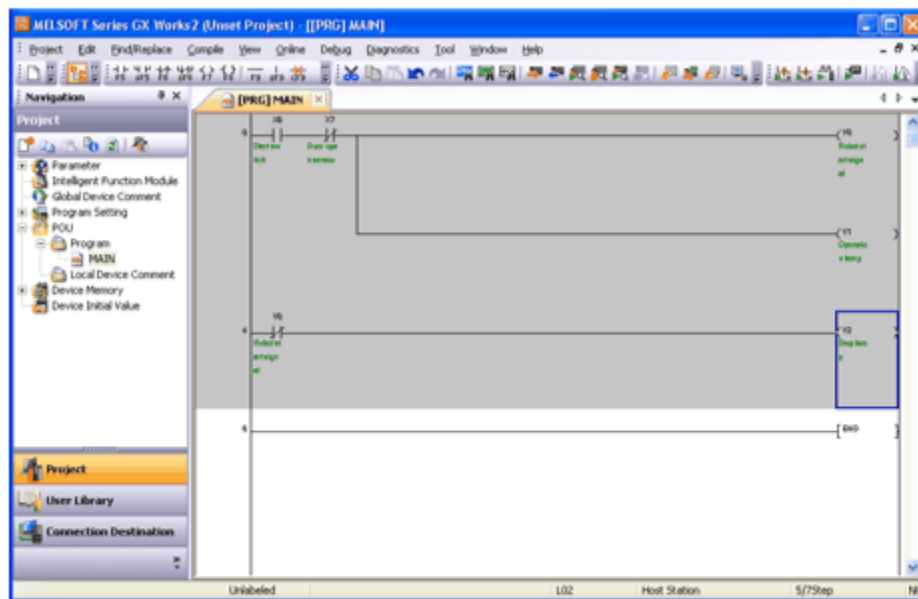
1.2

Требования к программированию

В этом курсе основное внимание уделяется вопросу использования специализированного программного обеспечения для программируемых контроллеров GX Works2 для разработки примера программы.

Ниже перечислены несколько главных функций ПО GX Works2.

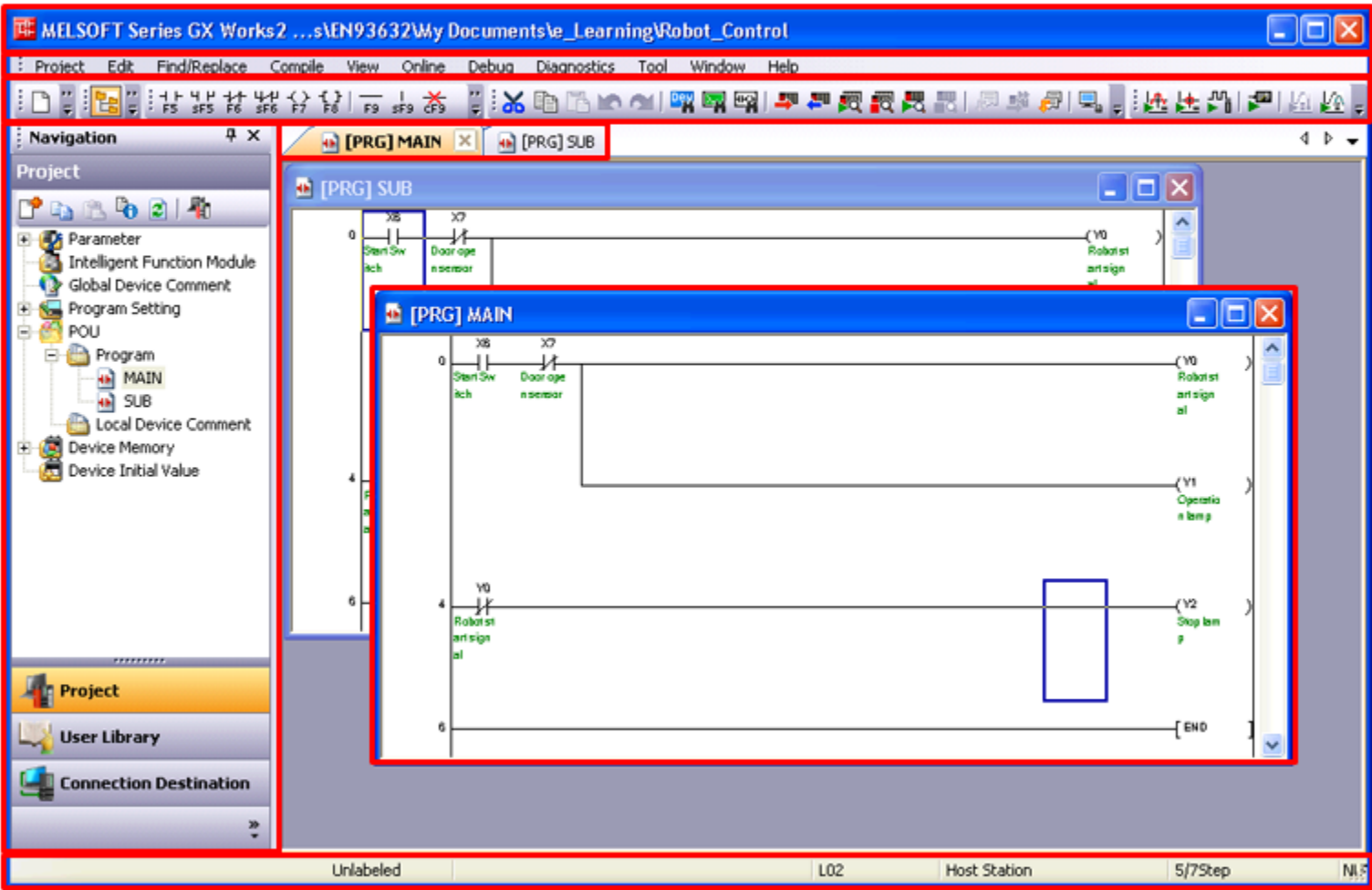
- Управление памятью и файлами
- Разработка программ для программируемых контроллеров
- Управление программной документацией (комментариями и т. д.)
- Чтение данных из модуля ЦП и запись данных в модуль ЦП
- Проверка работы программ
 - Имитация аппаратного обеспечения ПЛК программными средствами
 - Принудительное включение и выключение устройств ввода/вывода
 - Мониторинг состояния ввода/вывода и адресов памяти
- Обслуживание и устранение неисправностей



1.3 Интерфейс ПО GX Works2

Ниже представлен интерфейс ПО GX Works2.

Переместите курсор мыши в красную рамку, чтобы увидеть описание соответствующей функции.



1.4 Процедура создания программы

Ниже представлен порядок создания программы.



Глава 2 Проектирование программ

Из Главы 2 вы узнаете о том, как проектировать программы, включая определение элементов управления и их преобразование в программу.



Этапы обучения, предусмотренные в Главе 2

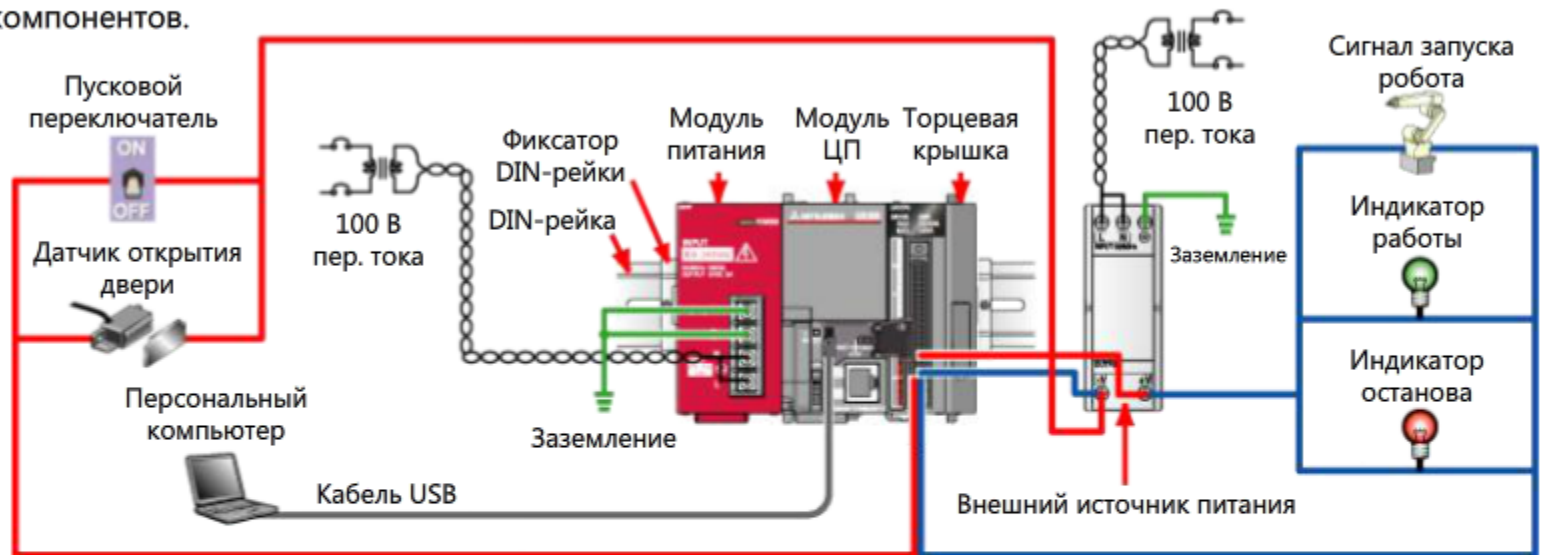
- 2.1 Конфигурация оборудования системы, используемой для обучения
- 2.2 Определение элементов управления
- 2.3 Создание таблицы соответствия устройств ввода/вывода адресам операндов
- 2.4 Проектирование программ

2.1

Конфигурация оборудования системы, используемой для обучения

В этом курсе вы создадите систему ПЛК (далее — «пример системы»), которая запускает робота в соответствии с установленной процедурой.

Ниже представлена схема конфигурации оборудования примера системы с перечнем аппаратных компонентов.



Наименование	Компонент	Модель	Описание
Система ПЛК	Модуль питания	L61P	Обеспечивает питанием другие модули, включая модуль ЦП и модуль входов/выходов.
	Модуль ЦП	L02CPU	Управляет системой ПЛК.
	Торцевая крышка	L6EC	Закреплена с правой стороны аппаратного модуля, крайнего справа
	Кабель USB	MR-J3USBCBL3M	Соединяет персональный компьютер, на котором установлено ПО GX Works2, с модулем ЦП.
	Персональный компьютер	—	Обеспечивает работу установленного на нем ПО GX Works2.
Внешний источник питания	—	—	Обеспечивает питанием внешние устройства ввода/вывода.
Внешнее оборудование ввода/вывода	Переключатель	—	Для начала управления установите в положение ВКЛ.
	Датчик	—	Определяет, открыта или закрыта дверь.
	Робот	—	Работает в соответствии с сигналами управления.
	Две лампы	—	Включаются в зависимости от состояния работы.

2.2

Определение элементов управления

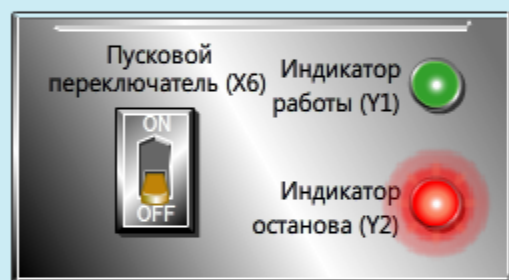
Первый шаг проектирования программы состоит в определении устройств, которые будут управляться, и устройств ввода/вывода, необходимых для осуществления такого управления. В нашем примере выполняется управление запуском и остановкой работы робота. Робот не сможет работать, если дверь защитного ограждения открыта, и будет останавливаться, если ее откроют во время его работы.

Чтобы лучше понять, как работает пример системы, см. приведенную ниже анимационную схему.

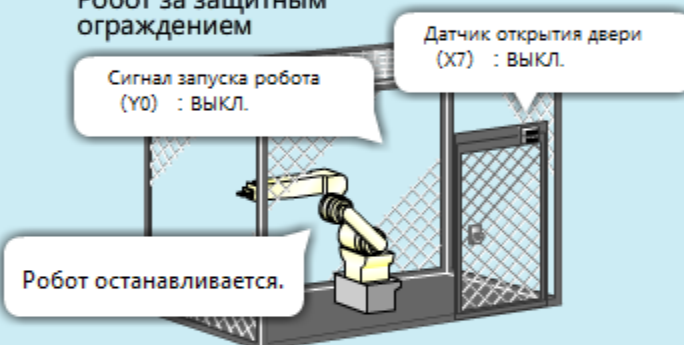
Работа системы, использующейся в качестве примера

Щелкните на красном кружке

Панель управления роботом



Робот за защитным ограждением



При установке **пускового переключателя (X6)** в положение OFF (ВЫКЛ.) **сигнал запуска робота (Y0)** выключается, вследствие чего робот прекращает работу. При этом **индикатор работы (Y1)** на панели управления выключается, а **индикатор останова (Y2)** включается.

Повторить



Назад

2.3 Создание таблицы соответствия устройств ввода/вывода и адресов операндов

Хорошей идеей будет для любой создаваемой программы составить таблицу, включающую все устройства ввода/вывода и регистры, которые используются в ПЛК, с добавлением поясняющих комментариев. Это уменьшает вероятность ошибок на этапе проектирования и программирования и повышает эффективность программирования. Если таблица соответствия для системы уже создана (например, лицом, выполнявшим конфигурацию оборудования), воспользуйтесь ей.

Приведенная ниже таблица представляет собой таблицу соответствия для примера системы, используемого в этом курсе.

Название устройства ввода/вывода	Номер операнда	Тип ввода/вывода	Тип операнда	Описание
Пусковой переключатель	X6	Ввод	Битовый	Этот переключатель запускает или останавливает работу робота.
Датчик открытия двери	X7	Ввод	Битовый	Этот датчик проверяет, открыта ли дверь защитного ограждения робота. Когда дверь открывается, датчик включается. Когда дверь закрывается, датчик выключается.
Сигнал запуска робота	Y0	Вывод	Битовый	Когда включается этот сигнал, робот начинает работать.
Индикатор работы	Y1	Вывод	Битовый	Эта лампа горит, когда робот работает.
Индикатор останова	Y2	Вывод	Битовый	Эта лампа горит, когда робот не работает.

* В случае использования словных данных в таблицу должны быть включены начальное значение, диапазон установок (верхний и нижний пределы), тип данных (со знаком, вещественный и т. д.) и комментарий. Эта информация будет полезной для отладки и изменения программ.

2.4

Проектирование программ

Спроектируем программу на языке релейной логики, используя элементы управления и таблицу соответствия ввода/вывода. Программа на языке релейной логики и таблица соответствия для примера системы приведены ниже.

Программа на языке релейной логики

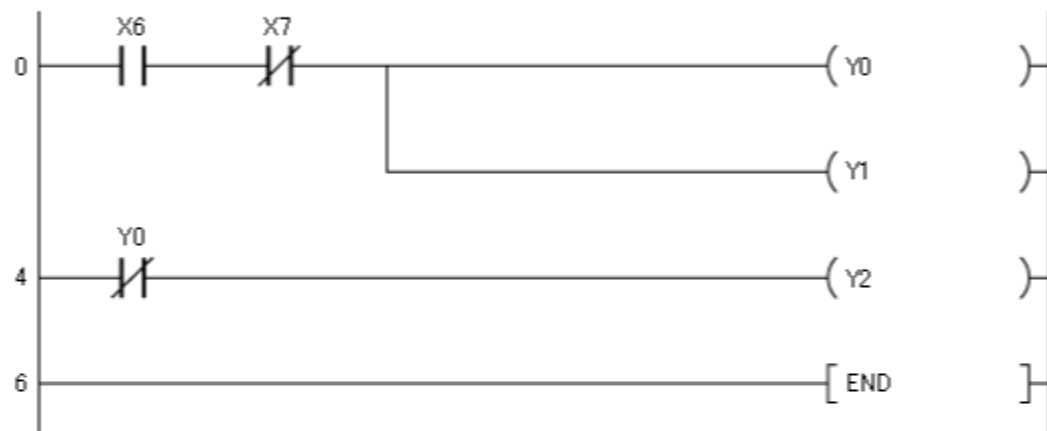


Таблица соответствия ввода/вывода

Название устройства ввода/вывода	Тип	Номер операнда
Пусковой переключатель	Ввод	X6
Датчик открытия двери	Ввод	X7
Сигнал запуска робота	Вывод	Y0
Индикатор работы	Вывод	Y1
Индикатор останова	Вывод	Y2

Глава 3 Программирование

Из главы 3 вы узнаете, как создавать программы, используя ПО GX Works2.

Проектирование программ Глава 2



Программирование Глава 3



Отладка Глава 4

Этапы обучения, предусмотренные в Главе 3

- 3.1 Создание проектов
- 3.2 Создание программ
- 3.3 Облегчение программ для понимания
- 3.4 Компиляция программ
- 3.5 Сохранение проектов

3.1

Создание проектов

Первым шагом для написания программы является создание проекта. Проект представляет собой набор данных, которые используются в GX Works2 для управления программами. В приведенной ниже таблице перечислены основные компоненты проекта.

Тип данных	Описание
Программа	Исходный и преобразованный коды для последовательности операций ЦП.
Комментарий	Текстовые пояснения, которые отображаются внутри программы. Подробнее см. Раздел 3.3. «Облегчение программ для понимания».
Параметры	Содержат информацию о настройках и параметрах системы.
Настройка передачи данных	Информация о способе подключения, необходимая для организации связи между системой с ПО GX Works2 и модулем ЦП.

Программа на языке релейной логики

GX Works2 позволяет создавать проекты двух типов.

Для примера программы, используемого в этом курсе, используется проект типа «Простой проект»

Тип проекта	Описание
Простой проект	Этот тип проекта имеет обратную совместимость с проектами GX Developer. Впоследствии такие проекты можно преобразовывать в структурированные, но не наоборот.
Структурированный проект	В этих проектах можно использовать дополнительный язык программирования, который называется Structured Ladder (Структурированная лестница). Кроме того, программы могут состоять из подпрограмм, а часто используемые фрагменты кода можно легко исполнять в виде готовых блоков и использовать повторно с помощью библиотеки пользователя. Также в структурированном проекте могут использоваться переменные. Это может повысить эффективность программирования и отладки, особенно для очень больших проектов.

Переменные

Переменные представляют собой создаваемые пользователями имена, присваиваемые операндам. Они могут быть глобальными, локальными или общими для всей системы при использовании программной надстройки MELSOFT Navigator. Проекты простого типа могут быть созданы как с возможностью использования переменных, так и без нее. В нашем примере переменные использоваться не будут.

3.1

Создание проектов

Чтобы приступить к созданию проекта, выполните следующие настройки.
 Перед созданием проекта необходимо узнать серию и название модели программируемого контроллера, а также тип проекта, который будет использоваться.

Наименование	Описание
Project type (Тип проекта)	Тип проекта определяет функции, доступные при написании программ. Для этого примера выберите «simple project» (линейный проект).
Use label (Использовать переменные)	Установите флажок напротив этого параметра, если для написания программ требуется использование переменных. В нашем примере программы переменные не используются, поэтому устанавливать флажок не нужно.
PLC series (Серия ПЛК)	Параметр «Серия ПЛК» определяет серию ПЛК из доступных для выбора в раскрывающемся списке PLC type (Тип ПЛК). Для этого примера выберите «LCPU» (LCPU).
PLC type (Тип ПЛК)	Параметр «Тип ПЛК» определяет модель ПЛК Выберите модель программируемого ПЛК, в данном случае, «L02» (L02).
Programming language (Язык программирования)	Параметр «Язык программирования» определяет тип первой автоматически созданной программы (MAIN). Впоследствии можно добавлять дополнительные программы на других языках. Для этого примера выберите «Ladder» (Язык релейной логики).

На следующей странице будет рассмотрен процесс создания нового проекта.

3.1

Создание проектов

MELSOFT Series GX Works2

Project Edit Find/Replace Compile View Online Debug Diagnostics Tool Window Help

Navigation

Project

Project

User Library

Connection Destination

New Project

Project Type: Simple Project

☐ Use Label


PLC Series: LCPU

PLC Type: L02

Language: Ladder

OK Cancel

Это приведет к созданию нового проекта.

Для продолжения нажмите кнопку .

3.2

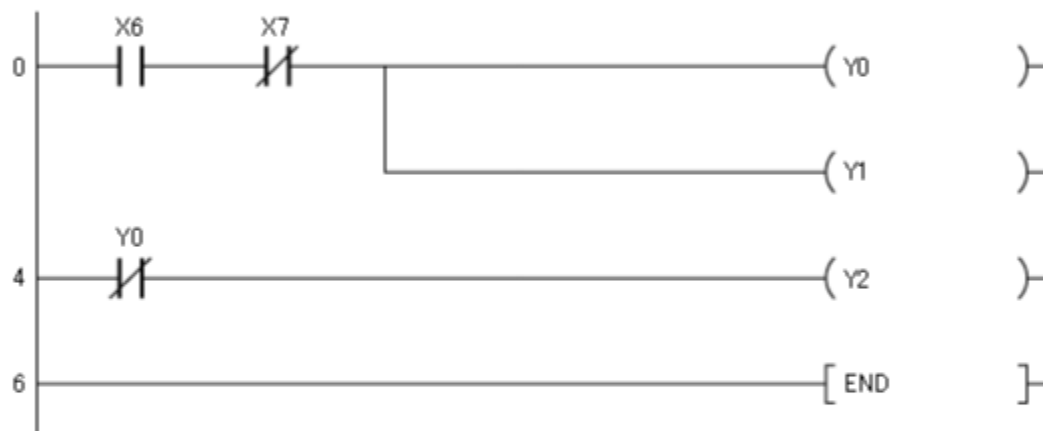
Создание программ

Мы уже создали проект. Давайте перейдем к созданию программы.

Создайте следующую программу и изучите основные операции (ввод, изменение, удаление, копирование и вставка команд, ввод/удаление линий).

Ниже приводится программа для примера системы, приведенного в Главе 2.

Программа для примера системы



На следующей странице вы попробуете создать эту программу с помощью окна имитации.

3.2

Создание программ

MELSOFT Series GX Works2 (Unset Project) - [[PRG] MAIN]

Project Edit Find/Replace Compile View Online Debug Diagnostics Tool Window Help

Navigation

Project

- Parameter
- Intelligent Function Module
- Global Device Comment
- Program Setting
- POU
 - Program
 - MAIN
 - Local Device Comment
- Device Memory
- Device Initial Value

Project

User Library

Connection Destination

[PRG] MAIN

```
graph LR
    X6[X6] -- NO --> X7[X7]
    X7 -- NC --> Y0[Y0]
    Y0 -- NC --> Y1[Y1]
    Y1 -- NO --> Y2[Y2]
    Y2 -- NO --> END[END]
```

Unlabeled


L02

Host Station

0/15Step

NI

Программа на языке релейной логики готова.

Для продолжения нажмите кнопку .

3.3

Облегчение программ для понимания

В своем нынешнем состоянии визуальное представление программы содержит только операнды, команды, линии и номера шагов. При взгляде на сложную программу может быть трудно определить, что она делает.

- Ошибки программирования, например использование неправильных адресов операндов или инструкций, обнаружить нелегко.
- В целом трудно выполнить анализ операций, отладку и изменение программы.
- Понимание принципа работы программы может оказаться сложной, если не невозможной задачей для тех, кто не является разработчиком программы

Способ упростить задачу

Включите в программу **пояснения (комментарии)**, что позволит другим быстро понять принцип ее работы, а также сделает ее удобной для Вашего восприятия в последующем.



В ПО GX Works2 можно использовать комментарии трех типов.
Подробную информацию см. в руководстве по созданию простых проектов в GX Works2.

Тип комментария	Предмет комментария
Device comment (Комментарий к операнду)	Введите до 32 символов, которые будут отображаться под выбранным операндом (устройством ввода/вывода или другим адресом памяти).
Statement (Комментарий к части программы)	Введите до 64 символов, которые будут добавлены в верхней части выбранного блока лестничной программы (над номером шага). Каждый блок лестничной программы может иметь несколько таких комментариев.
Note (Примечание)	Введите до 32 символов для отображения над выбранной катушкой реле или командой.

На следующей странице представлена имитация процесса добавления комментариев к операнду для примера программы.

3.3

Облегчение программ для понимания

MELSOFT Series GX Works2 (Unset Project) - [[PRG] MAIN]

Project Edit Find/Replace Compile View Online Debug Diagnostics Tool Window Help

Navigation

Project

- Parameter
- Intelligent Function Module
- Global Device Comment
- Program Setting
- POU
 - Program
 - MAIN
- Local Device Comment
- Device Memory
- Device Initial Value

Project

User Library

Connection Destination

[[PRG] MAIN

```
graph LR
    subgraph Rung_0 [0]
        X6[X6 Start switch] -- NO --> X7[X7 Door open sensor]
        X7 -- NC --> Y0[Y0 Robot start signal]
    end
    subgraph Rung_4 [4]
        Y0[Y0 Robot start signal] -- NO --> Y2[Y2 Stop lamp]
    end
    subgraph Rung_6 [6]
        END[END]
    end
```

Unlabeled L02 Host Station 5/75Step

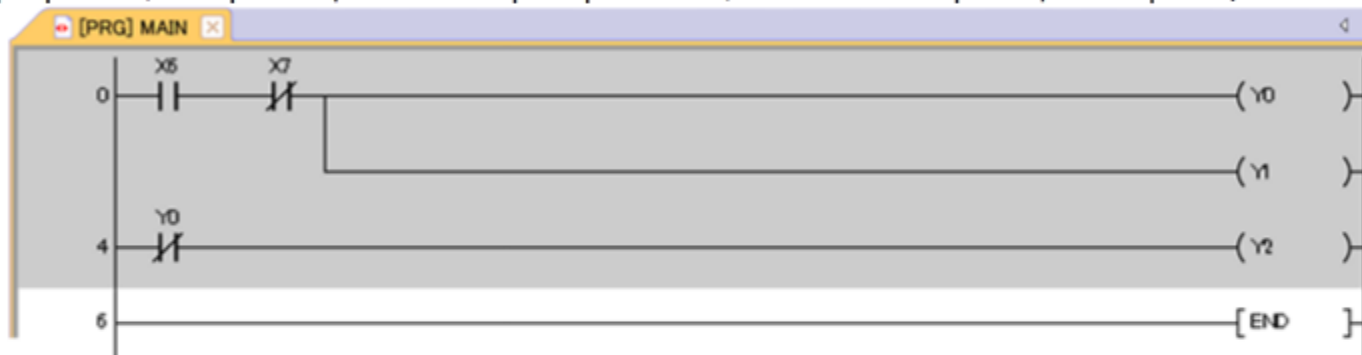
Ввод комментариев к операндам завершен.
Для продолжения нажмите кнопку

3.4

Компиляция программ

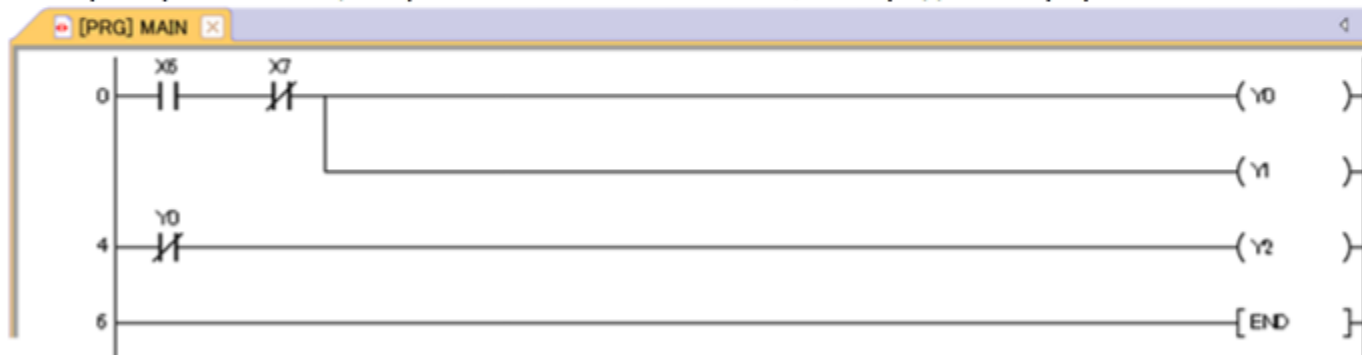
После завершения программы нужно преобразовать ее в формат, «понятный» для модуля ЦП. Непреобразованные программы не могут обрабатываться в ПЛК или быть сохранены на компьютере.

Программы, которые еще не были преобразованы, отличаются серым цветом фона (см. иллюстрацию ниже).



Преобразовать

После преобразования цвет фона изменяется на белый, как продемонстрировано ниже.



На следующей странице вы попытаете преобразовать программу с помощью окна имитации.

3.4 Компиляция программ

MELSOFT Series GX Works2 (Unset Project) - [[PRG] MAIN]

Project Edit Find/Replace Compile View Online Debug Diagnostics Tool Window Help

Navigation

Project

- Parameter
- Intelligent Function Module
- Global Device Comment
- Program Setting
- POU
 - Program
 - MAIN
- Local Device Comment
- Device Memory
- Device Initial Value

Project

User Library

Connection Destination

[[PRG] MAIN

0 X8 X7 Y0

Start switch Door open sensor Robot start signal

4 Y0 Robot start signal

6 Y2 Stop lamp

[END]

После преобразования программы цвет фона изменится с серого на белый.

Программа преобразована.
Для продолжения нажмите кнопку .

Unlabeled L02 Host Station 5/75Step

3.5

Сохранение проектов

После завершения преобразования программы сохраните проект с программами. При закрытии GX Works2 без сохранения проекта связанные программы будут удалены, поэтому необходимо регулярно сохранять проект. При сохранении нового проекта укажите следующие типы информации о проекте. (Это не требуется для сохранения с перезаписью.)

Следует включить информацию, облегчающую понимание содержимого элементов управления программы, имя системы и т. д.

Наименование	Обязательно	Описание
Save destination path (Сохранить конечный путь)	✓	Укажите папку, в которой будет размещено рабочее место.
Workspace/project list (Перечень рабочих мест/проектов)		Если в папке, указанной в области «Save destination path» (Сохранить конечный путь), уже существует одно или несколько рабочих мест, здесь будут перечислены существующие рабочие места.
Workspace name (Имя рабочего места)	✓	Укажите имя рабочего места длиной до 128 символов.
Project name (Имя проекта)	✓	Укажите название проекта длиной до 128 символов.
Title (Заголовок)		Укажите заголовок проекта длиной до 128 символов. Этот параметр полезен, когда требуется назначить длинное имя, которое не вписывается в параметр «Project name» (Имя проекта).

Рабочее место представляет собой папку для работы с несколькими проектами.

Ниже приведен пример использования рабочего места. (Представлены проекты для каждого типа транспортного средства на производственной линии сборки автомобилей.)

Имя рабочего места	Имя проекта	Заголовок
Производственная линия сборки автомобилей	Производственная линия типа А	Программа нормальной эксплуатации для управления производственной линией типа А
	Производственная линия типа В	Программа нормальной эксплуатации для управления производственной линией типа В
	Производственная линия типа С	Программа нормальной эксплуатации для управления производственной линией типа С

Примечания

- В случае сохранения проекта с непреобразованной программой не будет сохранена только такая программа. Перед сохранением проекта выполните преобразование программы, как описано в Разделе 3.4.
- Укажите конечный путь сохранения, имя рабочего места и имя проекта, при этом общее количество символов не должно превышать 150.

На следующей странице вы попытаетесь сохранить проект с помощью окна имитации.

3.5 Сохранение проектов

MELSOFT Series GX Works2 C:\SequenceProgram\Learning\Robot_Control - [[PRG] MAIN]

Project Edit Find/Replace Compile View Online Debug Diagnostics Tool Window Help

Navigation

Project

- Parameter
- Intelligent Function Module
- Global Device Comment
- Program Setting
- POU
 - Program
 - MAIN
 - Local Device Comment
- Device Memory
- Device Initial Value

Project

User Library

Connection Destination

[PRG] MAIN

0

X6 Start Switch

X7 Door open sensor

Y0 Robot start signal

Y1 Operation lamp

4

Y0 Robot start signal

Y2 Stop lamp

6

END

Unlabeled

L02

Host Station

6/75Step

NU

Проект сохранен.

Для продолжения нажмите кнопку

Глава 4 Отладка

Из Главы 4 вы узнаете, как записывать программы в модуль ЦП и выполнять их отладку.



Этапы обучения, предусмотренные в Главе 4

- 4.1 Отладка
 - 4.1.1 Отладка программы без использования модуля ЦП
 - 4.1.2 Изменение состояния устройства ввода/вывода
 - 4.1.3 Мониторинг состояния устройства
- 4.2 Запись программ в модуль ЦП
- 4.3 Активация записанных программ
- 4.4 Запуск программ
- 4.5 Отладка программ
- 4.6 Проверка работы системы ПЛК
- 4.7 Работа системы ПЛК

4.1

Что такое отладка?

После записи программы или сегмента программы необходимо проверить ее код, чтобы убедиться, что она работает правильно.

Дефекты программного обеспечения (когда написанный код не работает так, как ожидается) называются **ошибками**, а процесс поиска таких ошибок и их устранение известен как **отладка**.

Тестирование и отладка — неотъемлемые этапы создания программ.

Ошибки могут привести к остановке системы, повреждению оборудования или другим авариям.

В таблице ниже перечислены несколько функций в GX Works2, упрощающие процесс отладки.

Наименование функции	Описание
Simulator (Имитатор)	Эта функция используется для имитации выполнения программы даже без модуля ЦП. Она может использоваться для отладки в такой среде, в которой модуль ЦП недоступен.
Monitor (Мониторинг)	Эта функция позволяет осуществлять мониторинг состояний/значений всех операндов ПЛК во время выполнения программы модулем ЦП. В зависимости от условий применения доступны различные функции мониторинга, такие как мониторинг в окне программы, табличный мониторинг выбранных операндов и мониторинг всех операндов выбранного типа
Change current value (Изменение текущего значения)	С помощью этой функции можно принудительно изменить состояние операнда (битовый: ВКЛ. ↔ ВЫКЛ., словный: текущее значение) во время выполнения программы модулем ЦП. Эта функция полезна для изменения текущего значения словного операнда или состояния внутреннего маркера.
Forced input output registration/cancellation (Принудительная регистрация/отмена входного/выходного сигнала)	С помощью этой функции можно принудительно изменить состояние (ВКЛ. ↔ ВЫКЛ.) зарегистрированных устройств ввода/вывода во время выполнения программы модулем ЦП. Для отладки или проверки работы только с помощью модуля ЦП эта функция может заменять физический тумблер, подключенный на вход ПЛК

Использование этих функций в процессе отладки более подробно описано далее в этой главе.

Примечания по отладке

Не выполняйте отладку, когда программируемый контроллер подключен к физическим устройствам ввода/вывода. Ошибки в программе, принудительно запущенные устройства ввода/вывода и изменения словных величин могут привести к повреждению внешнего оборудования или худшим последствиям. Если отключенная система ПЛК недоступна, воспользуйтесь функцией имитации.

4.1.1

Отладка программы без использования модуля ЦП

Если модуль ЦП для отладки не доступен, воспользуйтесь **функцией имитации**.

Программа может работать на виртуальном модуле ЦП, реализованном программными средствами без использования настоящего модуля ЦП.



 Вкл.

 Выкл.

Наименование	Состояние	Описание
Switch (Переключатель)	RUN (РАБОТА)	Запускает виртуальный модуль ЦП.
	STOP (ОСТАНОВ)	Останавливает виртуальный модуль ЦП.
	RESET (СБРОС)	Сбрасывает виртуальный модуль ЦП. (Активно только в состоянии останова.)
LED (СВЕТОДИОД)	MODE (РЕЖИМ)	Показывает состояние режима виртуального ЦП.
	RUN (РАБОТА)	Показывает состояние работы виртуального ЦП. •Вкл.: состояние RUN (РАБОТА) •Выкл.: состояние STOP (ОСТАНОВ)
	ERR (ОШИБКА)	Показывает состояние ошибки виртуального модуля ЦП. В случае обнаружения ошибки в программе светодиодный индикатор включится или начнет мигать.
	USER (ОШИБКА ПОЛЬЗОВАТЕЛЯ)	Показывает наличие ошибок пользователя в виртуальном ЦП. Включается или мигает при возникновении ошибки.

Примечания по использованию функции имитации

- Отладка с помощью функции имитации не гарантирует правильную работу программы после отладки.
- Функция имитации выполняет обмен данными с модулями ввода/вывода с помощью памяти имитации. Функция не поддерживает некоторые команды, функции и память операндов, поэтому результаты работы с функцией имитации могут отличаться от результатов работы с настоящим модулем ЦП.

На следующей странице вы попробуете использовать функцию имитации с помощью окна имитации.

4.1.1

Отладка программы без использования модуля ЦП

MELSOFT Series GX Works2 C:\SequenceProgram\Learning\Robot_Control - [[PRG] MAIN]

Project Edit Find/Replace Compile View Online Debug Diagnostics Tool Window Help

Navigation

Project

- Parameter
- Intelligent Function Module
- Global Device Comment
- Program Setting
- POU
 - Program
 - MAIN
 - Local Device Comment
- Device Memory
- Device Initial Value

Project

User Library

Connection Destination

[PRG] MAIN

0

X6 Start Switch

X7 Door open sensor

Y0 Robot start signal

Y1 Operation lamp

4

Y0 Robot start signal

Y2 Stop lamp

6

END


Unlabeled

L02

Host Station

6/75Step

NU

Вы научились пользоваться функцией имитации.
Для продолжения нажмите кнопку  .

4.1.2

Изменение состояния устройства ввода/вывода

Выполняя отладку последовательной программы с использованием модуля ЦП, к которому не подключены устройства ввода/вывода, или используя simulator function (функцию имитации), задействуйте функцию **Forced Input Output Registration/Cancellation** (принудительной регистрации/отмены входного/выходного сигнала), чтобы изменить состояние ВКЛ./ВЫКЛ. устройства ввода/вывода.

Состояние зарегистрированных устройств ввода/вывода можно принудительно изменить на ВКЛ. или ВЫКЛ. с помощью программного обеспечения.

(ПЛК серии MELSEC-Q и MELSEC L): На экране «Forced Input Output Registration/Cancellation» (Принудительная регистрация/отмена входного/выходного сигнала)

(ПЛК серии MELSEC-F): На экране «Modify Value» (Изменить значение)

No.	Device	ON/OFF	No.	Device	ON/OFF
1	X6	ON	17		
2	X7	OFF	18		
3			19		
4			20		
5			21		
6			22		
7			23		
8			24		
9			25		
10			26		
11			27		
12			28		
13			29		
14			30		
15			31		
16			32		

Экран Forced Input Output Registration/Cancellation
(Принудительная регистрация/отмена
входного/выходного сигнала)
(ПЛК серии MELSEC-Q и MELSEC L)

Device/Label	Data Type	Setting Value
X6	Bit	ON

Экран Modify Value (Изменить значение) (ПЛК
серии MELSEC-F)

Изменение состояний других устройств

Чтобы изменить текущее значение словного операнда или состояние ВКЛ./ВЫКЛ. внутреннего маркера, воспользуйтесь **функцией изменения текущего значения**.

Подробную информацию см. в руководстве.

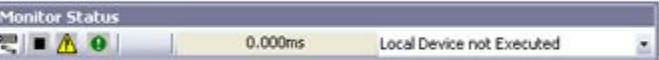
4.1.3 Мониторинг состояния устройства

При запуске имитации автоматически начинается мониторинг состояния операндов. Чтобы войти в режим мониторинга при подключении к физическому ЦП программируемого контроллера, просто выберите пункт меню Online (Подключение), Monitor (Мониторинг), а затем Start Monitoring (Начать мониторинг), или нажмите клавишу F3.

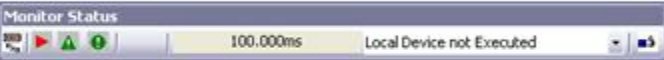
В режиме мониторинга можно увидеть значения и состояния всех использованных в программе операндов, непосредственно в программном коде. Это позволяет пользователю видеть изменения значений, включая результаты использования функции «Принудительная регистрация/отмена входного/выходного сигнала».

Кроме того, на экране отображается панель инструментов **Monitor Status** (Мониторинг состояния), которая содержит основную информацию, необходимую для определения состояния ЦП или виртуального ЦП. Приведенная ниже таблица поможет разобраться в информации, которая отображается на панели **Monitor Status** (Мониторинг состояния).

При подключении к модулю ЦП



При использовании функции имитации



Состояние	Пиктограмма/индикация	Описание
Состояние подключения	При подключении к модулю ЦП	Отображает состояние подключения к модулю ЦП или использования функции имитации.
	При использовании функции имитации	
Состояние RUN/STOP	RUN (РАБОТА)	Отображает состояние работы ЦП (RUN или STOP).
	STOP (ОСТАНОВ)	
Состояние ошибки	ERR. off (Индикатор ошибки не активен)	Отображает состояние ошибки модуля ЦП.
	ERR. on (Индикатор ошибки активен)	
	↔ ERR blinking (Индикатор ошибки мигает)	
Состояние пользователя	USER off (Индикатор ошибки пользователя не активен)	Отображает состояние ошибки пользователя модуля ЦП.
	USER on (Индикатор ошибки пользователя активен)	
	↔ USER blinking (Индикатор ошибки пользователя мигает)	
Время цикла	0.000ms	Отображает максимальное время цикла отслеживаемого модуля ЦП.
Наличие/отсутствие неподдерживаемых команд	Наличие неподдерживаемой команды.	Отображает наличие или отсутствие неподдерживаемой команды при использовании функции имитации. Если щелкнуть на пиктограмме, откроется окно Unsupported Instruction/Device (Неподдерживаемая команда/устройство).
	Отсутствие неподдерживаемой команды.	

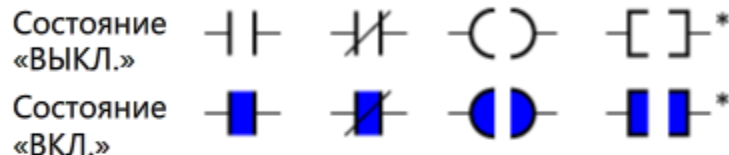
4.1.3

Мониторинг состояния устройства

В режиме мониторинга отображается текущее состояние всех операндов программы.

Отображение состояния битовых операндов (ВКЛ./ВЫКЛ.)

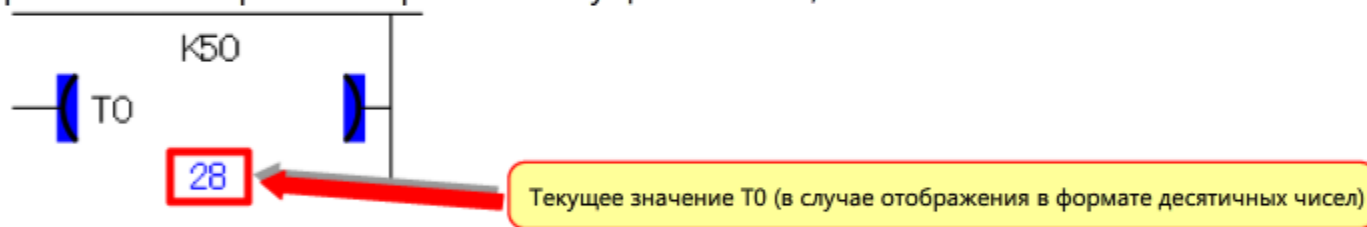
В режиме мониторинга отображается состояние ВКЛ./ВЫКЛ., как показано ниже.



* Такое отображение относится только к командам SET, RST, PLS, PLF, SFT, SFTP, MC и команде сравнения типов контактов. Обратите внимание, что для команды RST отображается только состояние ВКЛ./ВЫКЛ.

Отображение текущего значения словного операнда (отображение числа в десятичной/шестнадцатеричной системе)

В режиме мониторинга отображается текущее значение, как показано ниже.



Мониторинг только определенных операндов

В случае мониторинга очень больших или сложных программ может оказаться целесообразным отслеживать только определенные операнды, представляющие интерес. Для этого в ПЛ GX Works2 предусмотрена возможность табличного мониторинга, позволяющая пользователям легко добавлять интересующие их операнды, просматривать текущее состояние этих операндов и изменять их значения во время мониторинга. Подробную информацию см. в руководстве по эксплуатации GX Works2 (общем).

Watch 1					
Device/Label	Current Value	Data Type	Class	Device	Comment
X7	--	Bit		X7	Door open sensor
Y0	--	Bit		Y0	Robot start signal
Y1	--	Bit		Y1	Operation lamp
Y0	--	Bit		Y0	Robot start signal
Y2	--	Bit		Y2	Stop lamp
Y0	--	Bit		Y0	Robot start signal

4.2

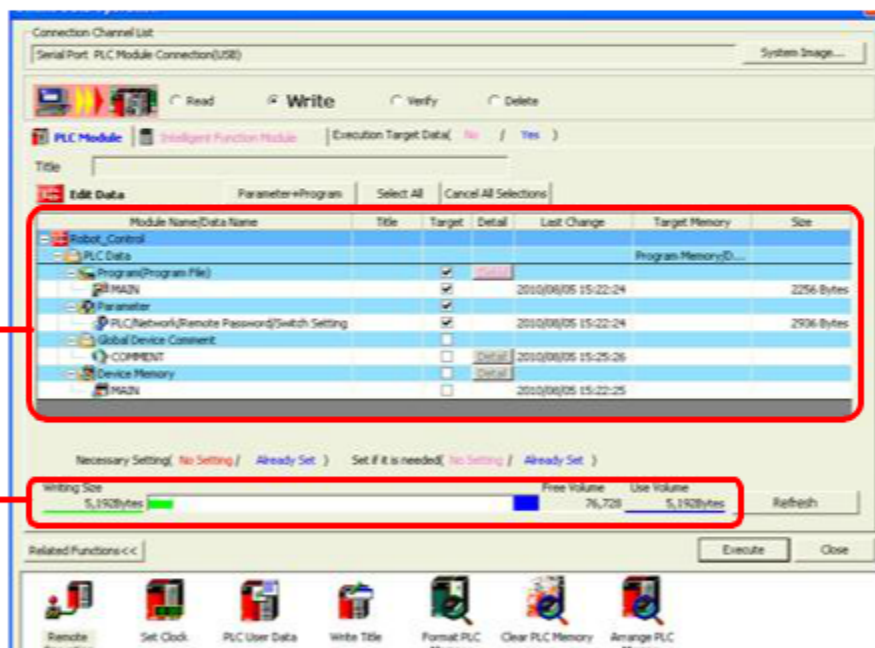
Запись программ в модуль ЦП

Прежде чем выполнять отладку с использованием физического модуля ЦП, переведите ЦП в **режим останова**, убедитесь, что соединение с ЦП установлено, и запишите программы и параметры в область памяти программы (Program memory) ЦП

Как видно из размещенного ниже снимка экрана, основные функции окна **Write to PLC** (Запись в ПЛК) позволяют пользователю выбрать файлы для записи, выбрать их расположение и подтвердить объем памяти ЦП. Три кнопки над перечнем файлов позволяют пользователю быстро выбрать нужные файлы для записи. Чаще всего используется функция **«Parameter+Program»** (Параметры + программа), которая будет рассмотрена в следующей имитации.

Перечень файлов

Объем памяти



На следующей странице вы попытаете записать программу в модуль ЦП с помощью окна имитации.

4.2

Запись программ в модуль ЦП

MELSOFT Series GX Works2 C:\SequenceProgram\Learning\Robot_Control - [[PRG] MAIN]

Project Edit Find/Replace Compile View Online Debug Diagnostics Tool Window Help

Navigation

Project

- Parameter
- Intelligent Function Module
- Global Device Comment
- Program Setting
- POU
 - Program
 - MAIN
- Local Device Comment
- Device Memory
- Device Initial Value

Project

User Library

Connection Destination

[[PRG] MAIN

0 X6 X7 Y0 Y1 Y2

Start Switch Door open sensor Robot start signal Operation lamp Stop lamp

4 Y0 Y2

Robot start signal Operation lamp Stop lamp

6 END

Unlabeled

L02

Host Station

6/75Step

NU

Программа записана в модуль ПЛК.

Для продолжения нажмите кнопку



4.3

Активация записанных программ

(ПЛК серии MELSEC-F): следующая операция необязательна.

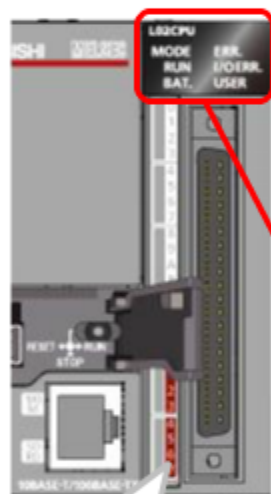
(ПЛК серии MELSEC-Q и MELSEC L): следующая операция обязательна.

Записав программу в модуль ЦП, **выполните его сброс**.

Записанные программы не будут работать, пока не будет выполнен сброс модуля ЦП.

* Выполнять эту операцию не нужно, если для отладки используется simulator function (функция имитации).

Выполните сброс модуля ЦП:



Переключатель
RESET/STOP/RUN
(СБРОС/ОСТАНОВ/
РАБОТА)

- (1) Нажмите и удерживайте переключатель RESET/STOP/RUN (СБРОС/ОСТАНОВ/РАБОТА) на передней панели модуля ЦП в положении RESET (СБРОС) (в течение 1 секунды или дольше).

[Выполняется сброс]

L02CPU	
MODE ■	ERR.
RUN	I/OERR.
BAT.	USER

MODE (РЕЖИМ): вкл., светится зеленым
RUN (РАБОТА): выкл.
ERR (ОШИБКА) : мигает

Нажмите и удерживайте в течение 1 секунды или дольше.



- (2) Отпустите выключатель после выключения горящего светодиодного индикатора MODE (РЕЖИМ) и мигающего индикатора ERR. (ОШИБКА).

[Сброс завершен]

L02CPU	
MODE ■	ERR.
RUN	I/OERR.
BAT.	USER

MODE (РЕЖИМ): вкл., светится зеленым
RUN (РАБОТА): выкл.
ERR (ОШИБКА) : выкл.

- (3) Переключатель возвращается в положение STOP (ОСТАНОВ), после чего процесс сброса завершен.

4.4

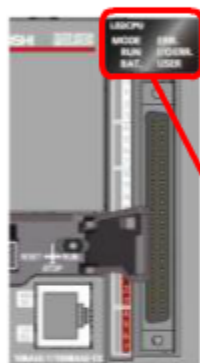
Запуск программ

ПЛК серии MELSEC-Q и MELSEC L

Запустите программу после выполнения сброса.

Для этого установите модуль ЦП в **состояние RUN (РАБОТА)**, выполнив следующие действия.

* Выполнять эту операцию не нужно, если для отладки используется simulator function (функция имитации).



- (1) Установите переключатель RESET/STOP/RUN (СБРОС/ОСТАНОВ/РАБОТА) на передней панели модуля процессора в положение RUN (РАБОТА).

Светодиодный дисплей в состоянии STOP (ОСТАНОВ)

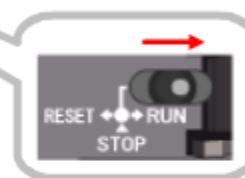
L02CPU	MODE	ERR.
MODE	RUN	I/OERR.
RUN	BAT.	USER
BAT.		



- (2) Если светодиодный индикатор RUN (РАБОТА) горит зеленым, это означает, что программа работает нормально.

Светодиодный дисплей в состоянии RUN (РАБОТА)

L02CPU	MODE	ERR.
MODE	RUN	I/OERR.
RUN	BAT.	USER
BAT.		



ПЛК серии MELSEC-F

Чтобы запустить программу после ее записи в основной блок, установите основной блок в состояние RUN (РАБОТА), выполнив следующие действия. (Операция сброса необязательна.)

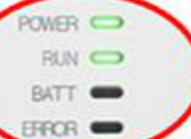
- (1) Установите переключатель RUN/STOP (РАБОТА/ОСТАНОВ) на передней панели основного блока в положение RUN (РАБОТА).



Светодиодный дисплей в состоянии STOP (ОСТАНОВ)



- (2) Если светодиодный индикатор RUN (РАБОТА) включен, это означает, что программа работает нормально.

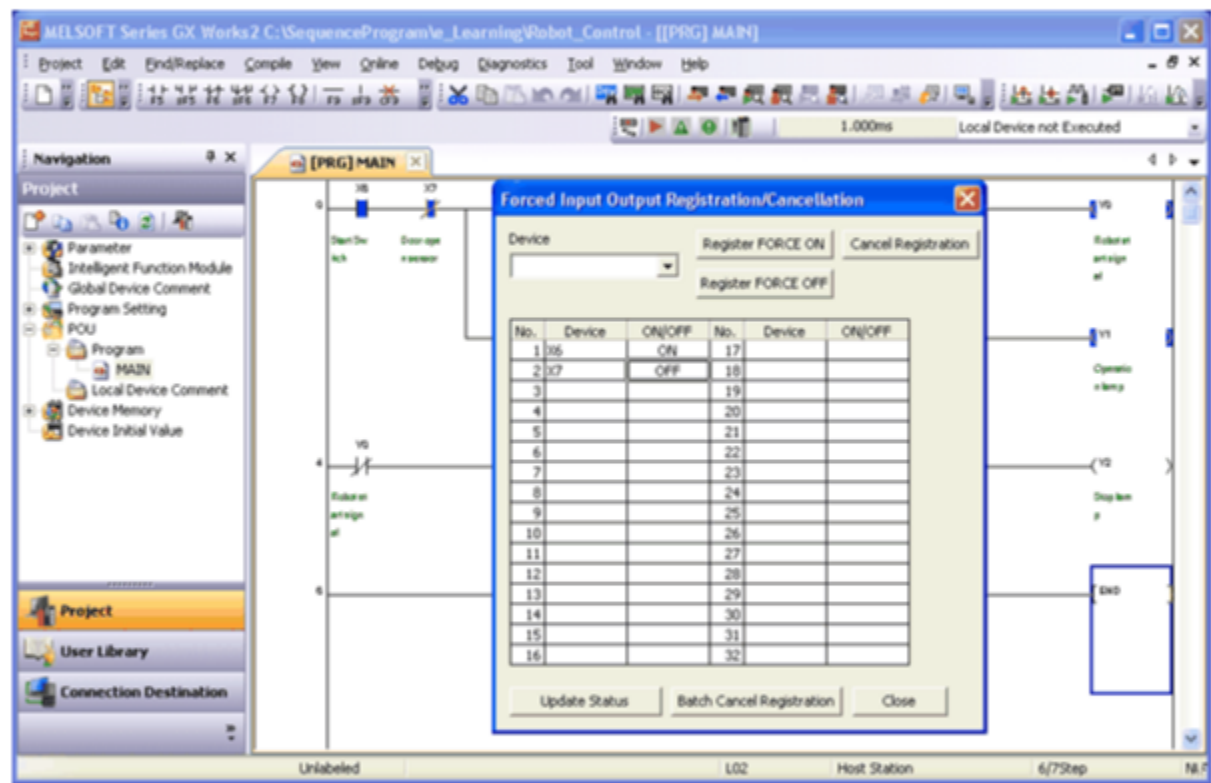


Светодиодный дисплей в состоянии RUN (РАБОТА)

4.5 Отладка программ

После запуска модуля процессора используйте функцию принудительной регистрации/отмены входного/выходного сигнала, чтобы изменить состояние всех устройств и отслеживать результат (выход) на лестнице.

(Пример экрана для ПЛК серии MELSEC-Q и MELSEC-L)



На следующей странице вы попробуете выполнить отладку программы с помощью окна имитации.

4.5 Отладка программ

MELSOFT Series GX Works2 C:\SequenceProgram\Learning\Robot_Control - [[PRG] MAIN]

Project Edit Find/Replace Compile View Online Debug Diagnostics Tool Window Help

F5 SF5 F6 SF6 F7 F8 F9 SF9 CF9

Dev

1.000ms

Local Device not Executed

Navigation

Project

- Parameter
- Intelligent Function Module
- Global Device Comment
- Program Setting
- POU
 - Program
 - MAIN- Local Device Comment
- Device Memory
- Device Initial Value

Project

User Library

Connection Destination

0

X6 Start Switch

X7 Door open sensor

Y0 Robot start signal

Y1 Operation lamp

4

Y0 Robot start signal

Y2 Stop lamp

6

END

Отладка программы завершена.

Для продолжения нажмите кнопку

Unlabeled

L02

Host Station

6/75Step

NL

4.6

Проверка работы системы ПЛК

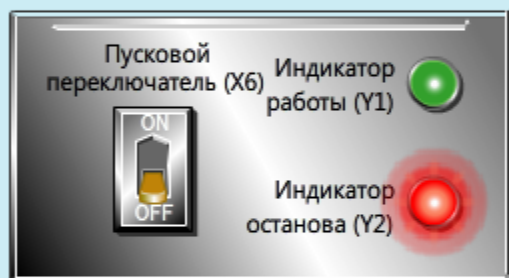
После завершения отладки программы запишите ее в физическую систему ПЛК, чтобы приступить к проверке ее работы. Проверьте управление физическим оборудованием ввода/вывода и удостоверьтесь, что все работает надлежащим образом.

Состояние всех устройств можно проверить с помощью функции монитора GX Works2 даже во время работы оборудования ввода/вывода.

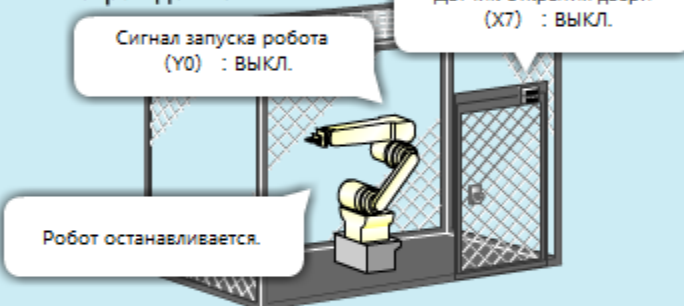
Работа системы, использующейся в качестве примера

Щелкните на красном кружке

Панель управления роботом



Робот за защитным ограждением

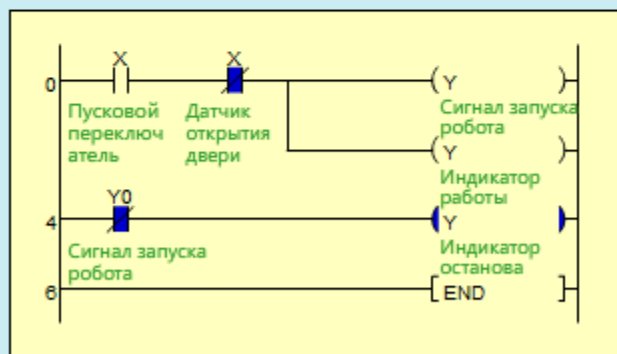


При установке **пускового переключателя (X6)** в положение OFF (ВЫКЛ.) **сигнал запуска робота (Y0)** выключается, вследствие чего робот прекращает работу. При этом **индикатор работы (Y1)** на панели управления выключается, а **индикатор останова (Y2)** включается.

Повторить



Назад



4.7

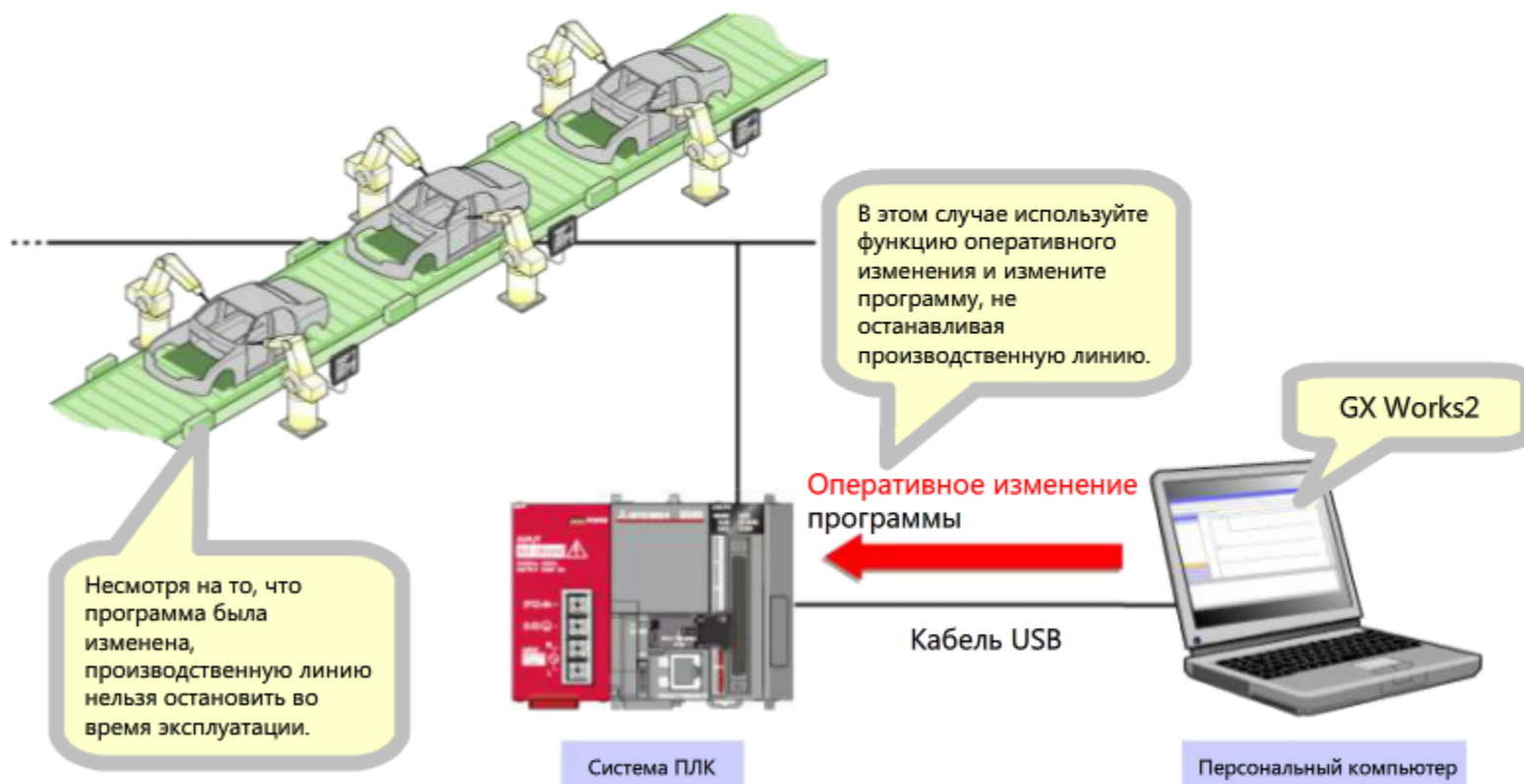
Работа системы ПЛК

После завершения проверки работы запустите систему ПЛК, чтобы приступить к эксплуатации.

Если необходимо изменить программу, не выключая систему

После начала работы системы может возникнуть необходимость внесения изменений в программу, например для устранения ошибок или расширения системы. Как правило, для записи измененной программы систему (модуль ЦП) нужно остановить, но это не всегда возможно. Для решения этой проблемы в GX Works предусмотрена функция оперативного изменения, которая используется для записи программ без остановки модуля процессора.

Пример. Производственная линия для сборки автомобилей, работающая 24 часа в сутки



На следующей странице вы попробуете использовать функцию оперативного изменения с помощью окна имитации.

4.7

Работа системы ПЛК

MELSOFT Series GX Works2 C:\SequenceProgram\Learning\Robot_Control - [[PRG] MAIN]

Project Edit Find/Replace Compile View Online Debug Diagnostics Tool Window Help

F5 SF5 F6 SF6 F7 F8 F9 SF9 CF9

Dev

1.000ms

Local Device not Executed

Navigation

Project

- Parameter
- Intelligent Function Module
- Global Device Comment
- Program Setting
- POU
 - Program
 - MAIN- Local Device Comment
- Device Memory
- Device Initial Value

Project

User Library

Connection Destination

[PRG] MAIN

0

X0

X0

Y0

Robot start signal

4

Y0

Robot start signal

Y2

Stop lamp

6

[END]

Unlabeled

L02

Host Station

2/75Step

NL

Оперативное изменение программы завершено.

Для продолжения нажмите кнопку

4.8

Заключение

На этом завершаются объяснения основных принципов создания программного обеспечения для программируемых контроллеров.

В этом курсе вы узнали следующее:

- элементы, необходимые для программирования системы ПЛК;
- некоторые основные принципы разработки программ, включая использование комментариев;
- как использовать ПО GX works2 для выполнения основных задач программирования с использованием ПЛК;
- несколько методов отладки программ для ПЛК.

Тест**Заключительный тест**

Вы завершили все уроки курса **Основы работы с ПО для ПЛК GX Works2** и готовы к прохождению итогового теста. Если вам неясны какие-либо из рассмотренных тем, воспользуйтесь возможностью еще раз просмотреть информацию по этим темам прямо сейчас.

Данный итоговый тест содержит всего 5 вопросов (15 пунктов).

Вы можете проходить заключительный тест любое количество раз.

Порядок подсчета баллов за тест

После выбора ответа обязательно щелкните кнопку **Ответить**. Если вы продолжите, не нажав кнопку «Ответить», ваш ответ будет потерян. (Будет считаться, что вы не ответили на вопрос.)

Результаты теста

Количество правильных ответов, количество вопросов, процент правильных ответов и результат (успешно ли пройден тест) будут отображаться на странице результатов.

Правильные ответы: **4**

Всего вопросов: **4**

Процент: **100%**

Продолжить

Просмотреть

Для успешного прохождения теста вы должны правильно ответить на **60%** вопросов.

- Щелкните кнопку **Продолжить**, чтобы завершить тест.
- Для просмотра теста нажмите кнопку **Просмотреть**. (Правильные ответы будут отмечены.)
- Щелкните кнопку **Повторить** попытку, чтобы пройти тест еще раз.

Тест**Заключительный тест 1**

Программа, за которую вы отвечали, перешла под руководство другого человека, которому сложно понять элементы ее управления. Укажите надлежащие меры по предотвращению подобных ситуаций.

- ☐ Использование функции комментирования в ПО GX Works2, надлежащего заголовка
- ☐ Устное объяснение элементов управления новому человеку.
- ☐ Работа над большими и сложными программами не должна передаваться другим.
- ☐ Передача таблицы соответствия устройств ввода/вывода адресам операндов вместе с

Тест

Заключительный тест 2



Выберите правильные варианты для отображения порядка создания программы.

Шаг 1 Проектирование программы

Шаг 2 (B1)

Шаг 3 (B2)

Шаг 4 Преобразование программ

Шаг 5 Сохранение проектов

Шаг 6 (B3)

Шаг 7 (B4)

Шаг 8 Запуск модуля ЦП (RUN)

Шаг 9 (B5)

Шаг 10 Проверка работы системы ПЛК

Ответить

Назад

Тест**Заключительный тест 3**

Заполните пропуски, чтобы воссоздать объяснение процесса, который необходимо выполнить после завершения работы над программой.

После записи программы ее необходимо проверить, чтобы убедиться, что она работает правильно.

А () (когда написанный код не работает так, как ожидается) носит название

() , а процесс поиска причин и их исправление —

().

Этот процесс является неотъемлемым этапом создания программ.

Тест

Заключительный тест 4



Выберите правильное применение для каждой из функций GX Works2.

Функция	Применение
Имитация	--Select--
Принудительная регистрация/отмена входного/выходного сигнала	--Select--
Изменение текущего значения	--Select--
Мониторинг «лестницы»	--Select--
Отслеживание	--Select--

[Ответить](#)[Назад](#)

Тест**Заключительный тест 5**

Выберите правильное определение функции оперативного изменения.

- ☐ Эта функция автоматически останавливает ЦП, записывает программу в ЦП, после чего автоматически
- ☐ Эта функция сравнивает программу в работающем модуле ЦП с программой, открытой в ПО GX Works2.
- ☐ Эта функция может записать программу в модуль ЦП после его безопасной остановки.
- ☐ Эта функция может записать программу в работающий модуль ЦП без его остановки.

Ответить

Назад

Тест**Результат теста**

Вы завершили заключительный тест. Ваша область результатов является следующей.

Правильные
ответы: 5

Всего вопросов: 5

Процент: 100%

Продолжить

Просмотреть

**П о з д р а в л я е м ! В ы п р о ш л и
т е с т .**

Вы завершили курс **Основы работы с ПО для ПЛК GX Works2.**

Благодарим за прохождение этого курса.

Надеемся, что вам понравились уроки, а информация, полученная
в рамках этого курса, окажется полезной в будущем.

Вы можете проходить данный курс любое количество раз.

Просмотреть

Заккрыть