

PLC GX Works2 Temel Bilgileri

Bu eğitim kursu (e-eğitim) sekans programları oluşturmak amacıyla GX Works2 yazılımını ilk kez kullananlar için tasarlanmıştır.

Bu kurs GX Works2 yazılımının programlanabilir denetleyicinin/kontrolörün (PLC) programlanması, hatalarının ayıklanması ve çalışma durumunun kontrol edilmesi için kullanılmasına ilişkin temel bilgiler sağlamaktadır. Kursun hedef kitlesi MELSEC-Q serisi, MELSEC-L serisi ve MELSEC-F serisi denetleyiciler için sekans programları oluşturan kişilerdir.

Bu kursun içeriği aşağıdaki gibidir.
Bölüm 1'den başlamanızı tavsiye ederiz.

Bölüm 1 - PLC Sistem Kontrol Yöntemi

Burada programlama için kullanılan programlama dili ve yazılımı tanıtılmaktadır.

Bölüm 2 - Program Tasarımı

Kontrol öğeleri ve donanım konfigürasyonuna bağlı olarak bir programın nasıl tasarlandığını öğreneceksiniz.

Bölüm 3 - Programlama

Özel GX Works2 yazılımı kullanılarak programlama yapmayı öğreneceksiniz.

Bölüm 4 - Hata Ayıklama

CPU modülüne sekans programları yazmayı ve bunların hatalarını ayıklamayı öğreneceksiniz.

Bölüm 5 - Son Test

Geçer not: %60 veya üzeri.

Sonraki sayfaya git		Sonraki sayfaya gidin.
Önceki sayfaya dön		Önceki sayfaya dönün.
İstenen sayfaya ulaş		"İçindekiler Tablosu" görüntülenerek istediğiniz sayfaya ulaşabilmenizi sağlar.
Eğitimden çık		Eğitimden çıkın. "İçindekiler" ekranı gibi pencereler ve eğitim kapatılacaktır.

Güvenlik önlemleri

Gerçek ürünleri kullanmayı öğrendiğinizde, lütfen ilgili kılavuzlardaki güvenlik önlemlerini dikkatlice okuyun.

Bu kurstaki önlemler

- Kullandığınız yazılım sürümünde görüntülenen ekranlar bu kurstakilerden farklı olabilir.

Bölüm 1 PLC Sistem Kontrol Yöntemi

Bu kursun hedef kitlesi mühendislik yazılımlarıyla çalışan kişilerdir. MELSEC-Q, L ve F serisi sistemlerin temel yönetim konseptlerinin bir kısmı ele alınmaktadır.

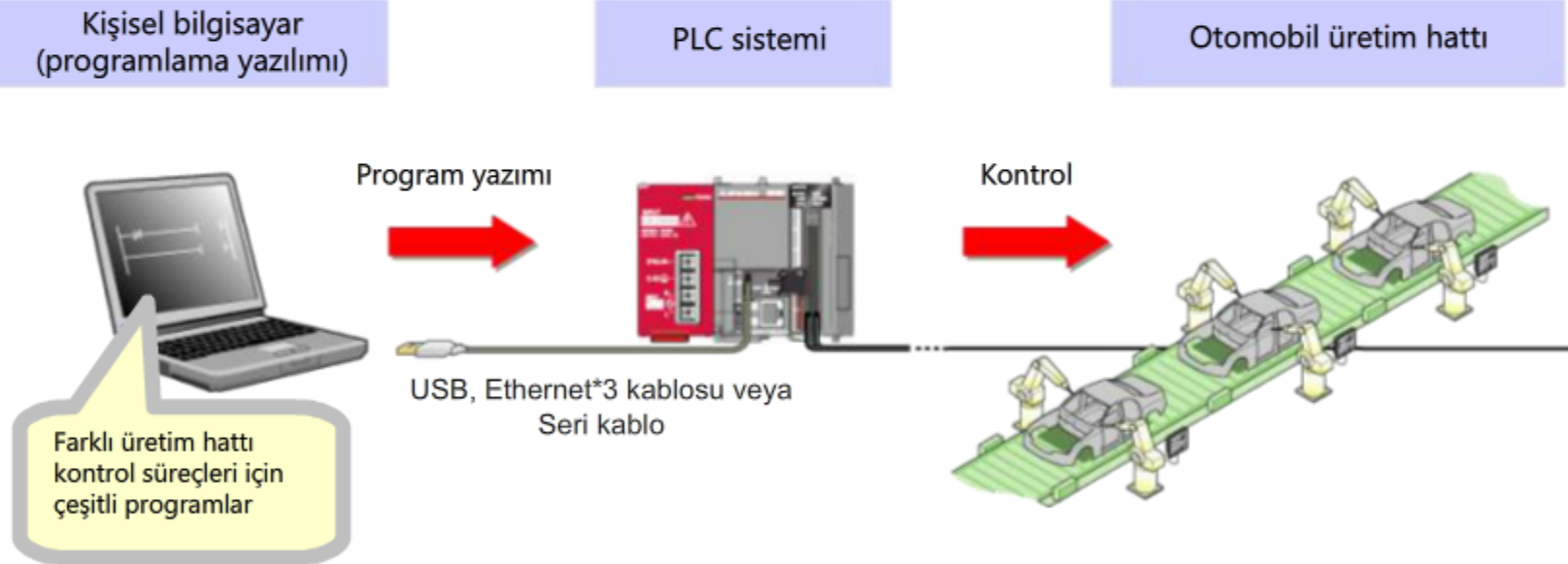
GX Works 2'de (GXW2) Sequential Function Chart (SFC) dili, Instruction List (IL)*1, Ladder Logic, Function Block Diagram (FBD)*2 ve Structured Text (ST) dâhil Uluslararası standart programlama dilleri kullanılmaktadır.

Programlar "mühendislik yazılımı" GX works2'yi çalıştıran bir kişisel bilgisayar kullanılarak geliştirilir ve programlanabilir denetleyici CPU modülü genellikle USB, Ethernet*3 veya Seri kablo aracılığıyla yazılır. CPU modülü, istenen kontrolde değişiklikler gerekli oldukça, istendiği kadar çok sayıda yeniden programlanabilir.

*1 GX works2 için gelecekte planlanmaktadır.

*2 Şu anda GX works2'de Structured Ladder olarak adlandırılmaktadır; IEC uyumu planlanmaktadır.

*3 Ethernet, Xerox Corp. firmasının tescilli ticari markasıdır.



Bu kursta, örnek programda ladder logic (en popüler PLC programlama dillerinden biri) kullanılmaktadır. Örnekte L Serisi PLC kullanılıyor olsa da, bu kursun içeriği aynı şekilde Q Serisi sistemler için de geçerlidir.

Temel kontrol yöntemi MELSEC-F serisi ile aynıdır; ancak bazı operasyonlar ve işlevler farklılık gösterir.

1.1

PLC Sistemi Yapım Prosedürü

Bu e-eğitim kursunda, programlanabilir denetleyici/kontrolör sistemlerinin uygulanması için gereken yazılım tasarım adımları (yeşil renkle gösterilmektedir) ele alınmaktadır.

Donanım tasarımı

(1) Sistem tasarımı MELSEC-Q/MELSEC-L Temel Kursu



(2) Ürün seçimi MELSEC-Q/MELSEC-L Temel Kursu



(3) Gelişmiş hazırlık MELSEC-Q/MELSEC-L Temel Kursu



(4) Kurulum ve kablo tesisatı MELSEC-Q/MELSEC-L Temel Kursu



(5) Kablo tesisatı kontrolü MELSEC-Q/MELSEC-L Temel Kursu

**Yazılım tasarımı**

(6) Program tasarımı Bölüm 2



(7) Programlama Bölüm 3



(8) Hata Ayıklama Bölüm 4



(9) Çalıştırma

**Bu kursun
kapsamı**

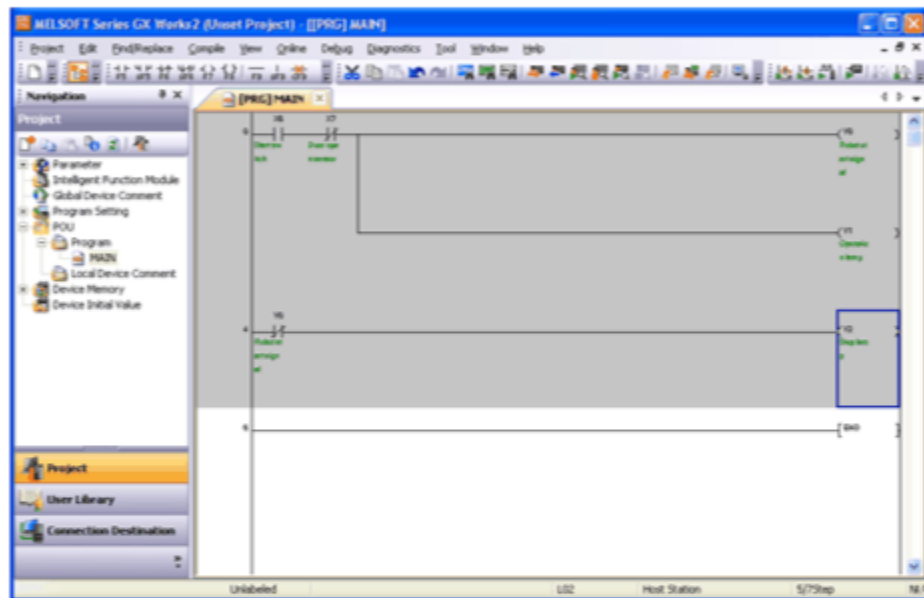
1.2

Programlama Gereksinimleri

Bu kursta, örnek sistem programını geliştirmek için programlanabilir denetleyici mühendislik yazılımı GX Works2'nin nasıl kullanılması gerektiğine yoğunlaşılmaktadır.

GX Works2'nin birkaç önemli işlevi aşağıda sıralanmaktadır.

- Bellek ve dosya yönetimi
- Programlanabilir denetleyici programlarının geliştirilmesi
- Program belgelerinin (yorumlar gibi) yönetimi
- CPU modülüne verilerin (özellikle programların) yazılması/okunması
- Programın çalıştığının doğrulanması
 - PLC donanımının yazılım simülasyonu
 - G/Ç açılmaya veya kapanmaya zorlanması
 - G/Ç ve bellek adres durumunun izlenmesi
- Bakım ve sorun giderme görevlerinin yapılması

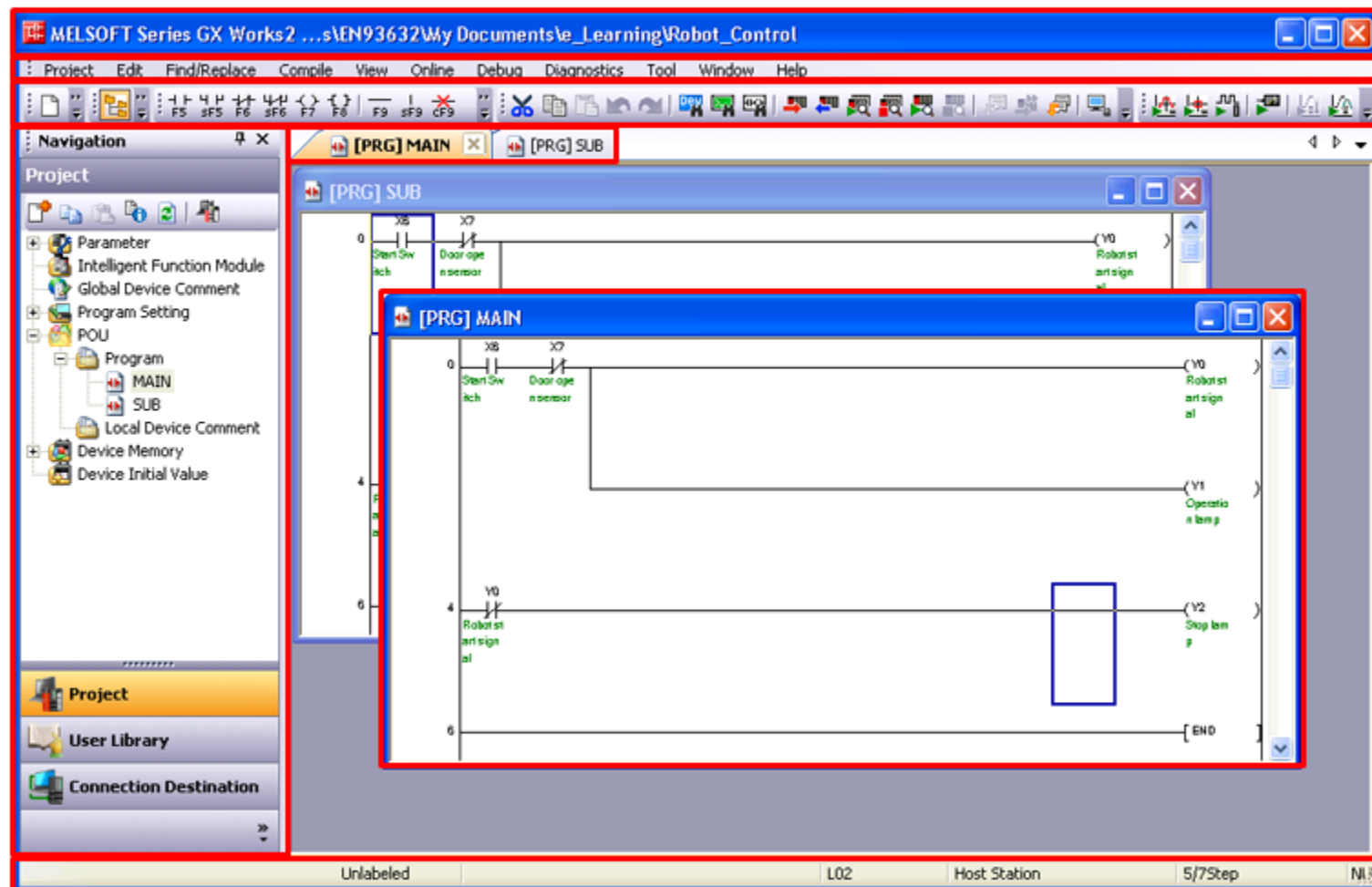


1.3

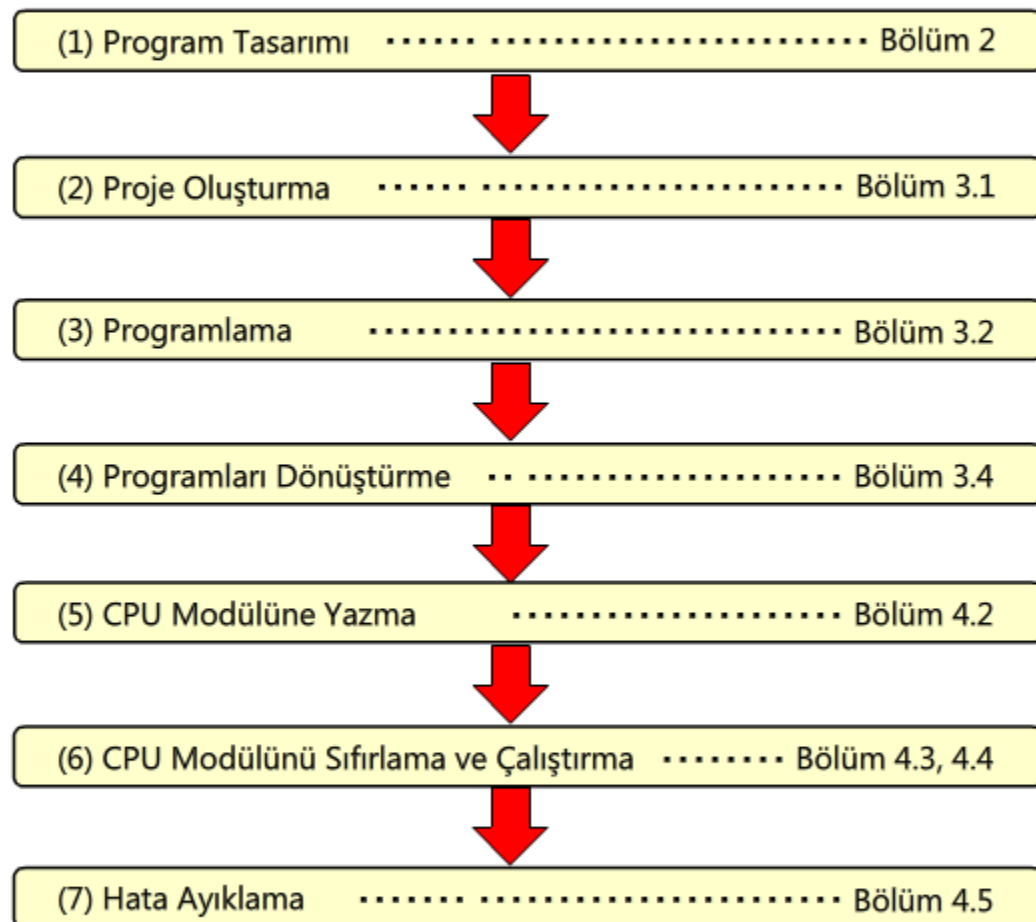
GX Works2 Ekran Konfigürasyonu

GX Works2 ekran konfigürasyonu aşağıda gösterilmektedir.

İlgili işlevi görüntülemek için fare imlecini kırmızı bir çerçevenin içine yerleştirin.

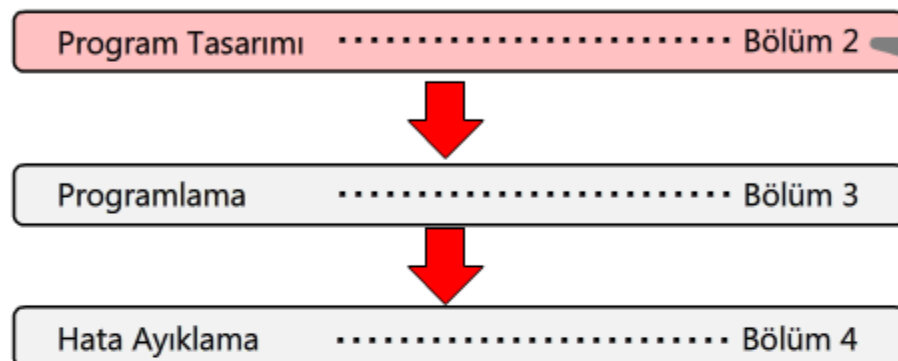


Aşağıdaki prosedüre göre bir sekans programı oluşturun.



Bölüm 2 Ekran Verilerinin Oluşturulması

Bölüm 2'de, kontrolün içeriğinin belirlenmesi ve bunların bir programa dönüştürülmesi dâhil olmak üzere programların nasıl tasarlandığını öğreneceksiniz.



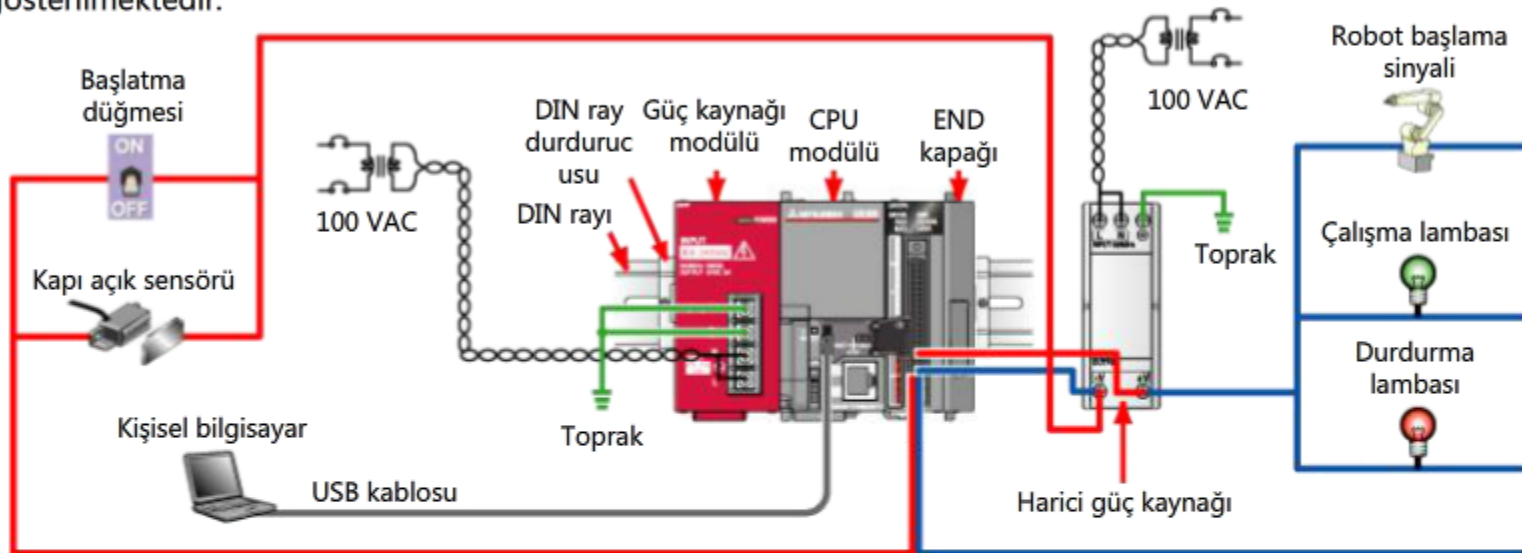
Bölüm 2'deki öğrenme adımları

- 2.1 Örnek Donanım Konfigürasyonu
Eğitim için Kullanılan Sistem
- 2.2 Kontrol Öğelerinin Tanımlanması
- 2.3 Bir G/Ç Uyum Tablosunun Oluşturulması
Cihazlar ve Cihaz Numaraları
- 2.4 Programın Tasarlanması

2.1 Eğitim için Kullanılan Örnek Sistemin Donanım Konfigürasyonu

Bu kursta, robotu bir prosedüre göre başlatan bir PLC sistemi (bundan sonra "örnek sistem" olarak adlandırılacaktır) oluşturacaksınız.

Örnek sistemin donanım konfigürasyonunu gösteren bir şema ile donanım bileşenlerinin bir listesi aşağıda gösterilmektedir.



Öge	Bileşen	Model	Açıklama
PLC sistemi	Güç kaynağı modülü	L61P	CPU modülü ve G/Ç modülü dâhil modüllere güç temin eder.
	CPU modülü	L02CPU	PLC sistemini kontrol eder.
	END kapağı	L6EC	Sistem bloğunun sağ tarafına takılır.
	USB kablosu	MR-J3USBCBL3M	GX Works2'nin kurulu olduğu kişisel bilgisayarı CPU modülüne bağlar.
	Kişisel bilgisayar	—	Kurulan GX Works2 ile çalışır.
Harici güç kaynağı	—	—	Harici G/Ç cihazlarına güç temin eder.
Harici G/Ç ekipmanı	Düğme	—	Kontrolü başlatmak için AÇIK konumuna getirilir.
	Sensör	—	Kapının açık mı yoksa kapalı mı olduğunu tespit eder.
	Robot	—	Kontrol sinyallerine uygun olarak çalışır.
	İki lamba	—	Çalışma durumuna göre yanar.

2.2

Kontrol Öğelerinin Tanımlanması

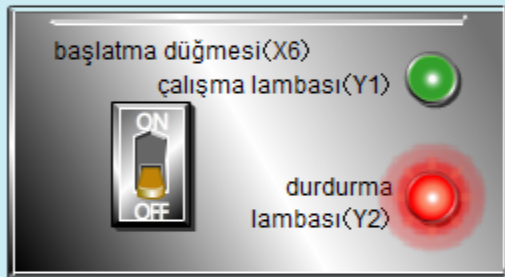
Program tasarlanmanın ilk adımı kontrol edilecek cihazların ve istenen kontrol için gerekli G/Ç cihazlarının tanımlanmasıdır. Örnek sistemde, bir robotun başlatılma ve durdurulma işleminin kontrolü gerçekleştirilmektedir. Güvenlik bariyerine açılan kapının açık olması durumunda robotun çalışması engellenecek, çalışma sırasında kapının açılması durumunda ise robot durdurulacaktır.

Örnek sistemin nasıl çalışacağını daha iyi anlamak için aşağıdaki animasyonu izleyin.

Örnek sistem çalışması

☐ Kırmızı dairenin içine tıklayın

Robot kontrol paneli



Güvenlik bariyerindeki robot



Başlatma düğmesi (X6) KAPALI olarak ayarlandığında, robot başlatma sinyali (Y0) kapanarak robotun çalışmasını durdurur. Aynı anda, kontrol panelindeki çalışma lambası (Y1) kapanır ve durdurma lambası (Y2) açılır.

Yeniden Oynat



Geri

2.3 G/Ç Cihazlarını ve Cihaz Numaralarını İçeren bir Uyum Tablosunun Oluşturulması

PLC'de kullanılan tüm G/Ç cihazlarını ve kayıtlarını ve bunların oluşturulan herhangi bir programa yönelik ilgili bilgilerini içeren bir tablonun oluşturulması iyi bir fikirdir. Bu tablo tasarım ve programlama sürecinde hata riskini azaltır ve programlama verimliliğinin artırılmasına olanak tanır. Donanımı yapılandıran kişi tarafından oluşturulanlar gibi, sistem için daha önceden oluşturulmuş bir uyum tablosu mevcut ise, bunu kullanın.

Aşağıdaki tablo, bu kursta kullanılan örnek sisteme ait uyum tablosudur.

G/Ç cihaz adı	Cihaz No.	G/Ç tipi	Cihaz tipi	Açıklama
Başlatma düğmesi	X6	Giriş	Bit	Bu düğme robotun çalışmasını başlatır veya durdurur.
Kapı açık sensörü	X7	Giriş	Bit	Bu sensör robotun güvenlik bariyerinin kapısının açık olup olmadığını kontrol eder. Kapı açıldığında, sensör devreye girer. Kapı kapatıldığında, sensör kapanır.
Robot başlama sinyali	Y0	Çıkış	Bit	Sinyal açıldığında, robot çalışmaya başlar.
Çalışma lambası	Y1	Çıkış	Bit	Bu lamba robot çalışırken yanar.
Durdurma lambası	Y2	Çıkış	Bit	Bu lamba robot durdurulduğunda yanar.

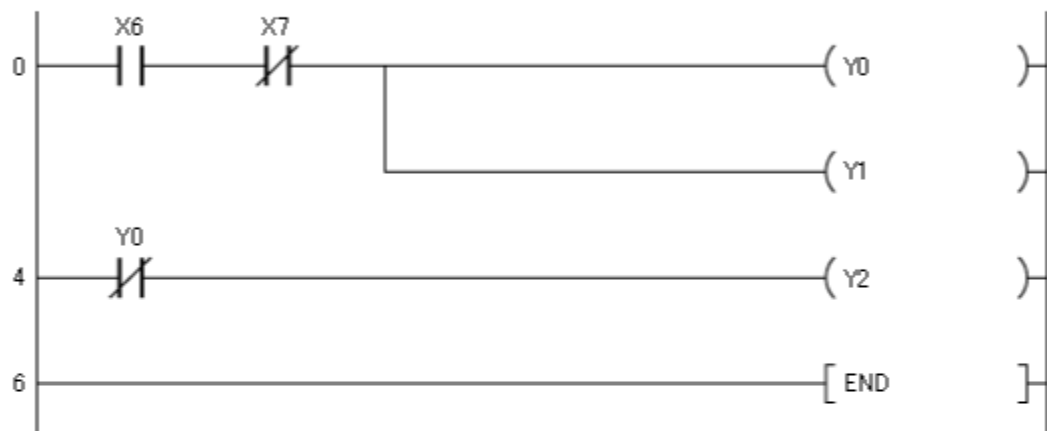
* Sözcük verileri kullanıldığı takdirde, tabloya ilk değer, ayarlama aralığı (üst ve alt limitler), veri tipi (imzalı, gerçek vb.) ve yorum eklenmelidir. Bu bilgi, programların tasarlanması ve değiştirilmesi için faydalı olacaktır.

2.4

Programların Tasarlanması

Kontrol öğeleri ve G/Ç uyum tablosuna göre Ladder logic dilini kullanarak bir program tasarlayın. Ladder programı ve örnek sistem için tasarlanan G/Ç uyum tablosu aşağıda gösterilmektedir.

Ladder programı



G/Ç uyum tablosu

G/Ç cihaz adı	Tip	Cihaz No. .
Başlatma düğmesi	Giriş	X6
Kapı açık sensörü	Giriş	X7
Robot başlama sinyali	Çıkış	Y0
Çalışma lambası	Çıkış	Y1
Durdurma lambası	Çıkış	Y2

Bölüm 3 Programlama

Bölüm 3'te, GX Works2 kullanarak tasarlanmış programı nasıl programlamanız gerektiğini öğreneceksiniz.

Program Tasarımı Bölüm 2



Programlama Bölüm 3



Hata Ayıklama Bölüm 4

Bölüm 3'deki öğrenme adımları

- 3.1 Proje Oluşturma
- 3.2 Program Oluşturma
- 3.3 Programları Kolay Anlaşılır Hale Getirme
- 3.4 Programları Yürütülebilir Şekle Dönüştürme
- 3.5 Projeleri Kaydetme

3.1

Proje Oluşturma

Program yazmanın ilk adımı bir proje oluşturmaktır.

Proje, GX Works2'nin programları yönetmek için kullandığı verilerden oluşan bir gruptur.

Aşağıdaki tabloda bir projenin temel bileşenleri sıralanmaktadır.

Veri türü	Açıklama
Program	CPU sekans işlemleri için kaynak kodu ve derlenmiş kod.
Yorum	Programın içinde görüntülenen belge tipi. Ayrıntılar için bkz. Bölüm 3.3 "Programları Kolay Anlaşılır Hale Getirme".
Parametre	Bir sisteme ait ayar ve konfigürasyon bilgilerinin çoğunu veya tümünü içerir.
Aktarım yapılandırması	GX Works2'yi çalıştıran sistem ile CPU modülü arasında iletişim kurulması için gerekli bağlantı yolu bilgisi.

Ladder programı

GX Works2 aşağıdaki proje tiplerini seçmenize olanak tanır.

Bu kurstaki örnek programda "simple project" tipi kullanılmaktadır.

Proje tipi	Açıklama
Basit proje	Bu proje tipi GX Developer projeleriyle geriye doğru uyumludur. Basit projeler daha sonra Yapılandırılmış projelere dönüştürülebilir, ancak bunun tersi mümkün değildir.
Yapılandırılmış projeler	Bu projeler Structured Ladder adlı ek bir programlama dilini kullanabilmektedir. Ayrıca, programlar çok sayıda küçük parçaya ayrılabilen ve kodun sık kullanılan parçaları modüllere ayrılıp bir kullanıcı kitaplığı yardımıyla yeniden kullanılabilir. Etiketler de benzer şekilde kolayca yeniden kullanılabilir. Bu, özellikle çok büyük projeler için programlama ve hata ayıklama verimliliğini artırabilmektedir.

Etiketler

Etiketler, kullanıcılar tarafından oluşturulup cihazların diğer adları haline gelen adlardır. Bunlar MELSOFT Navigator ile birlikte uygulandığında sistem yapında, global olarak veya yerel olarak kullanılabilir. Basit projeler, etiket kullanma yeteneği ile veya olmadan oluşturulabilir. Örnek projede, etiketler kullanılmayacaktır.

3.1

Proje Oluşturma

Örnek proje oluşturmaya başlamak için, aşağıdaki ayarları yapın.
Program oluşturmadan önce, programlanabilir denetleyici serisi, model adı ve kullanılacak proje tipi bilinmelidir.

Öge	Açıklama
Project type	Proje tipi program yazılırken hangi özelliklerin kullanılabileceğini belirler. Bu örnekte, lütfen "simple project" seçin.
Use label	Programların etiket kullanılarak yazılabilmesi gerekiyorsa, bu öğeyi seçin. Örnek programda etiket kullanılmamaktadır. Bu nedenle, bu kutuyu işaretlemeyin.
PLC series	PLC series, PLC type aşağı açılır listesinde seçime uygun olan modelleri belirler. Bu örnekte, lütfen "LCPU" öğesini seçin.
PLC type	PLC type, derleyicinin kullanıcı programlarını nasıl makine koduna dönüştürdüğünü belirler. Programlanacak PLC modelini seçin; örneğimizde "L02" seçilmelidir.
Programming language	Programlama dili otomatik olarak oluşturulan ilk programın (MAIN) program tipini belirler. Daha sonra, farklı diller kullanan ek programlar eklenebilir. Bu örnekte, lütfen "Ladder" öğesini seçin.

Lütfen yeni proje oluşturma süreçlerinin simüle edildiği sıradaki sayfayı inceleyin.

3.1

Proje Oluřturma

MELSOFT Series GX Works2

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

Navigation

Project

Project Type: Simple Project

PLC Series: LCPU

PLC Type: L02

Language: Ladder

OK Cancel

Use Label

Artık yeni bir proje oluşturulmuřtur.

İlerlemek için  düğmesini tıklayın

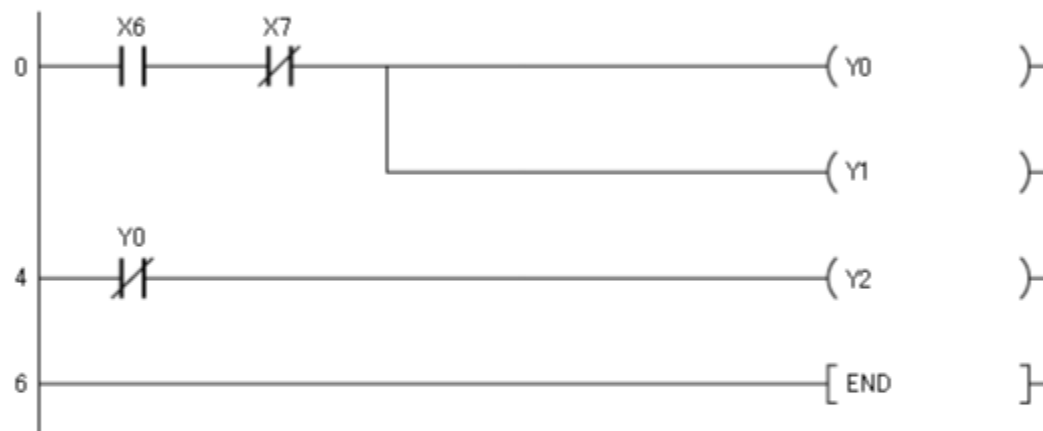
3.2

Program Oluşturma

Proje oluşturduktan sonra, bir program oluşturalım.

Aşağıdaki programı oluşturun ve temel işlemleri öğrenin (komut girişi, değiştirme, silme, kopyalama-yapıştırma ve çizilmiş çizgi girme/silme).

Bölüm 2'de örnek sistem için tasarlanan program aşağıda gösterilmektedir.

Örnek sisteme ait program

Sıradaki sayfada, simülasyon penceresini kullanarak bu programı oluşturmayı deneyin.

3.2

Program Oluşturma

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

Unlabeled

L02

Host Station

0/15Step

Artık Ladder devre programı tamamlanmıştır.
İlerlemek için  düğmesini tıklayın.

 düğmesini tıklayın.'"/>

3.3

Programları Kolay Anlaşılır Hale Getirme

Şu anki durumunda, programın görsel yapısında sadece cihazlar, komutlar, çizgiler ve adım numaraları bulunmaktadır. Karmaşık bir programa bakıldığında, programın ne yaptığının belirlenmesi zor olabilir.

- Hatalı cihaz numaraları veya komutlar gibi programlama yanlışlıklarının bulunması zor olabilir.
- Genel olarak, operasyonel analiz, hata ayıklama ve program genişletme işlemlerinin yapılması zor olabilir.
- Orijinal program geliştiricisinin artık programa bakım verememesi durumunda, programın nasıl çalıştığını öğrenmek herkes için göz korkutucu ve belki de imkansız olabilir.

Karşı önlemler

Herkesin programın nasıl çalıştığını hızlıca anlayabilmesi için programa **belgeler** ekleyin.

İyi bir uygulama örneği olarak, tüm programlayıcılar kendilerinin ve diğerlerinin programı daha iyi anlaması için programlarına ayrıntılı açıklamalar eklemelidir.



GX Works2 üç farklı açıklama türünün kullanılmasına imkan sağlar.

Daha ayrıntılı bilgi için, GX Works2 Basit Projeler kılavuzuna başvurun.

Açıklama türü	Açıklamanın kapsamı
Cihaz açıklaması	Seçilen cihazın altında (G/Ç veya diğer bellek adresi) görüntülenmek üzere maksimum 32 karakter girin.
Bildirim	Seçilen ladder bloğunun üstüne (adım numarasının üzerinde) görüntülenmek üzere bildirim başına maksimum 64 karakter girin. Her ladder bloğu birden fazla bildirime sahip olabilir.
Not	Seçilen bobinin veya uygulama komutunun üzerinde görüntülenmek üzere maksimum 32 karakter girin.

Sıradaki sayfada örnek programa yorum ekleme işlemi simüle edilmektedir.

3.3

Programları Kolay Anlaşılır Hale Getirme

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

[[PRG] MAIN

0 X6 X7 Y0 Y1 Y2

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

4 Y0

6 [END]

Cihaz açıklaması girme işlemi tamamlanmıştır.
İlerlemek için düğmesini tıklayın.

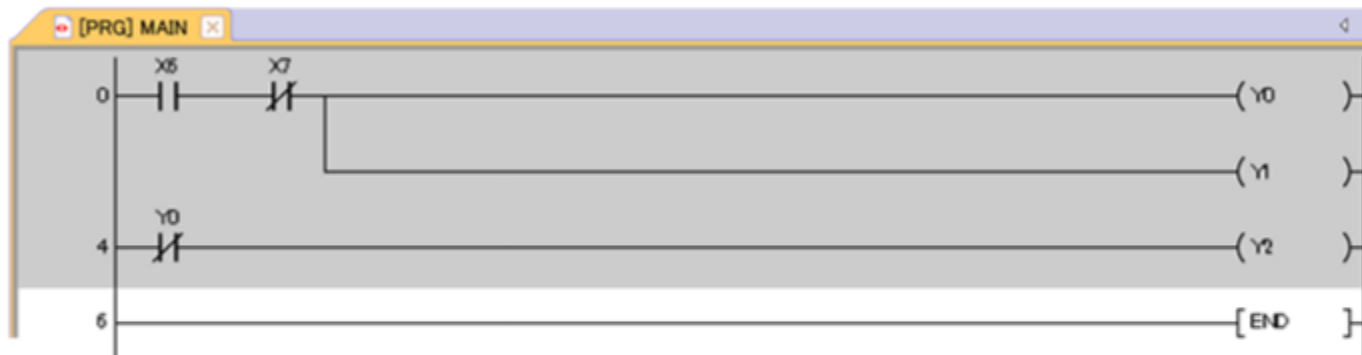
Unlabeled L02 Host Station 5/75Step

3.4

Programları Yürütülebilir Şekle Dönüştürme

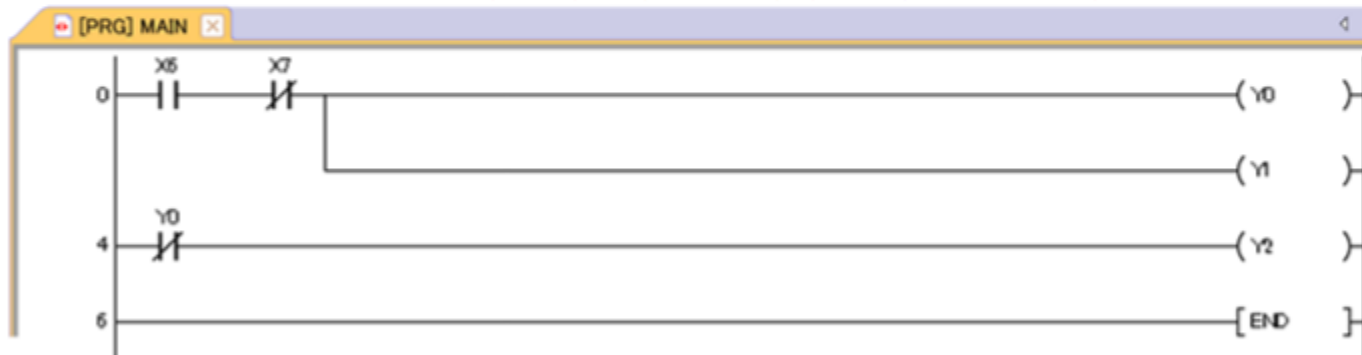
Program tamamlandıktan sonra, bunları CPU modülünde yürütülebilir bir şekle dönüştürmeniz gerekir. Dönüştürülmeyen programlar yürütülemez veya kaydedilemez.

Dönüştürülmemiş programların arka plan rengi, aşağıda gösterildiği gibi gridir.



Dönüştürün

Dönüştürmeden sonra, arka plan rengi aşağıda gösterildiği gibi beyaz olarak değişir.



Sıradaki sayfada, simülasyon penceresini kullanarak programı dönüştürmeyi deneyin.

3.4

Programları Yürütülebilir Şekle Dönüştürme

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

[[PRG] MAIN

0 X6 Start switch X7 Door open sensor Y0 Robot start signal Y1 Operation lamp Y2 Stop lamp

4 Y0 Robot start signal

6 [END]

Program dönüştürüldüğünde, gri olan arka plan rengi beyaza döner.

Program dönüştürülmüştür.
İlerlemek için  düğmesini tıklayın.

Unlabeled L02 Host Station 5/75Step

3.5

Projeleri Kaydetme

Program dönüştürme işlemi bittiğinde, programları içeren projeyi kaydedin. GX Works2'nin proje kaydedilmeden sonlandırılması durumunda, bağlantılı programlar silineceğinden, projenizi düzenli olarak kaydetmeniz gerekir. Yeni bir projeyi kaydederken, aşağıdaki proje bilgisi türlerini belirtin. (Üzerine yazma-kaydetme için bu gerekli değildir.) Diğer kişilerin programın kontrol içeriğini, sistem adını vb. anlamasını kolaylaştıran bilgilere yer vermeniz gerekir.

Öge	Gerekli	Açıklama
Save destination path	✓	Çalışma alanı ayrılacak klasörü belirleyin.
Workspace/project list		"Save destination path" bölümünde belirtilen klasörde bir ya da birkaç çalışma alanı zaten mevcut ise, mevcut çalışma alanları listelenir.
Workspace name	✓	Maksimum 128 karakter uzunluğunda bir çalışma alanı adı belirtin.
Project name	✓	Maksimum 128 karakter uzunluğunda bir proje adı belirtin.
Title		Maksimum 128 karakter uzunluğunda bir proje başlığı belirtin. Bu parametre, "Project name" bölümüne sığmayan uzun bir ad belirlemek istediğinizde faydalıdır.

Çalışma alanı birden fazla projeyi yönetmek için kullanılan bir klasördür.

Çalışma alanının bir kullanım örneği aşağıda gösterilmektedir. (Otomobil üretim hattında projeler her araç tipi için yönetilir.)

Workspace name	Project name	Title
Otomobil üretim hattı	A tipi üretim hattı	A tipi üretim hattının kontrolü için normal çalışma programı
	B tipi üretim hattı	B tipi üretim hattının kontrolü için normal çalışma programı
	C tipi üretim hattı	C tipi üretim hattının kontrolü için normal çalışma programı

Notlar:

- Dönüştürülmemiş program içeren bir proje kaydedilirse, sadece dönüştürülmemiş program silinir. Projeyi kaydetmeden önce, Kısım 3.4'te öğrendiğiniz gibi program dönüştürme işlemini gerçekleştirin.
- Toplam karakter sayısı 150'yi geçmeyecek şekilde save destination path, workspace name ve project name alanlarını doldurun.

Sıradaki sayfada, simülasyon penceresini kullanarak projeyi kaydetmeyi deneyin.

3.5

Projeleri Kaydetme

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

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

Navigation [PRG] MAIN

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

Unlabeled L02 Host Station 6/75Step

Diagram showing a ladder logic program with the following components:

- Step 0: Start Switch (X6) and Door open sensor (X7) in series, connected to Y0 (Robot start signal).
- Step 4: Y0 (Robot start signal) connected to Y2 (Stop lamp).
- Step 6: END.

Proje kaydedilir.
İlerlemek için  düğmesini tıklayın.

Bölüm 4 Hata Ayıklama

Bölüm 4'te, CPU modülüne sekans programları yazmayı ve bunların hatalarını ayıklamayı öğreneceksiniz.

Program Tasarımı Bölüm 2



Programlama Bölüm 3



Hata Ayıklama Bölüm 4

Bölüm 4'teki öğrenme adımları

4.1 Hata Ayıklama

4.1.1 CPU Modülünü Kullanmadan bir Programın Hatalarını Ayıklama

4.1.2 Bir G/Ç Cihazının Durumunu Değiştirme

4.1.3 Cihaz Durumunu İzleme

4.2 CPU Modülüne Program Yazma

4.3 Yazılmış Programları Etkinleştirme

4.4 Programları Çalıştırma

4.5 Programların Hatalarını Ayıklama

4.6 PLC Sisteminin Çalışmasını Kontrol Etme

4.7 PLC Sistemini Çalıştırma

4.8 Sonuç

4.1

Hata Ayıklama Nedir?

Bir program veya program bölümü yazıldıktan sonra, beklenen şekilde çalıştığından emin olmak için kodun test edilmesi gereklidir.

Yazılım kusurlarına (yazılan kodun amaçlanan şekilde işlev göstermemesine) "**yazılım hatası**" adı verilir ve amaçlanmayan davranışın sebebinin öğrenme ve bunu düzeltme süreci de "**hata ayıklama**" olarak bilinir.

Test ve hata ayıklama program oluşturmada temel adımlardır.

Yazılım hatalarının bulunması sistemin durmasına, ekipmanın zarar görmesine veya diğer kazalara neden olabileceğinden, bu adımlar programlanabilir denetleyicilerde özellikle önemlidir.

Aşağıdaki tabloda, GX Works2'de bulunan ve hata ayıklama sürecine yardımcı olabilecek birkaç işlev belirtilmektedir.

İşlevin adı	Açıklama
Simülatör	Bu işlev CPU modülü olmadan bile programın yürütülmesini simüle etmek için kullanılır. Bu işlev, CPU modülünün bulunmadığı bir ortamda hata ayıklama için kullanılabilir.
Monitör	Bu işlev, CPU modülünün yürütülmesi sırasında yürütme durumunun ve her bir cihazın durumunun izlenebilmesini sağlar. Ladder üzerinde izleme, sadece kayıtlı cihazların izlenmesi ve tüm cihazların toplu halde izlenmesi gibi, uygulamaya bağlı olarak çeşitli izleme işlevleri mevcuttur.
Geçerli değeri değiştirme	Bu işlev CPU modülünün yürütülmesi sırasında cihaz durumunu zorla değiştirebilir (bit; AÇIK ↔ KAPALI, sözcük: geçerli değer). Bu işlev bir sözcük cihazının geçerli değerinin veya dâhili bir rölenin durumunun değiştirilmesi için faydalıdır.
Zorlamalı giriş-çıkış kaydı/iptali	Bu işlev CPU modülünün yürütülmesi sırasında kayıtlı bir G/Ç cihazının durumunu (AÇIK ↔ KAPALI) zorla değiştirebilir. Tek başına CPU modülüyle hata ayıklama veya çalışma doğrulama için, bu işlev bir düğmenin yerine kullanılabilir.

Bu işlevler, bu bölümün kalanında hata ayıklama süreci bakımından daha ayrıntılı olarak açıklanmaktadır.

Hata ayıklama notları

Programlanabilir denetleyici fiziksel G/Ç cihazlarına bağlıyken hata ayıklama görevleri gerçekleştirmeyin.

Programdaki yazılım hataları, zorlanmış G/Ç cihazları veya sözcük değeri değişiklikleri harici ekipmanın zarar görmesine veya daha olumsuz durumlara neden olabilir.

Bağlantısı kesilmiş bir PLC sisteminin kullanılamaması durumunda, simülatör işlevini kullanın.

4.1.1

CPU Modülünü Kullanmadan bir Programın Hatalarını Ayıklama

Hata ayıklama için bir CPU modülü kullanılamıyorsa, **simülatör işlevini** kullanın.

Programlar, gerçek bir CPU modülü kullanılmadan yazılım tarafından sağlanan sanal CPU modülü üzerinde çalışabilir.



 Açık

 Kapalı

Öge	Durum	Açıklama
Düğme	RUN	Sanal CPU modülünü çalıştırır.
	STOP	Sanal CPU modülünü durdurur.
	RESET	Sanal CPU modülünün sıfırlar. (Sadece STOP durumunda etkinleşir)
LED	MODE	Sanal CPU'nun MODE durumunu gösterir.
	RUN	Sanal CPU'nun çalışma durumunu gösterir. •Açık: RUN durumu •Kapalı: STOP durumu
	ERR	Sanal CPU modülünün hata durumunu gösterir. Bir hata mevcut ise, LED açılır veya yanıp söner.
	USER	Sanal CPU'da bir kullanıcı hatasının olup olmadığını gösterir. Bir hata oluştuğunda açılır veya yanıp söner.

Simülatör işlevinin kullanımıyla ilgili notlar

- Simülatör işlevinin kullanıldığı hata ayıklama işleminde, hata ayıklandıktan sonra sekans programının doğru şekilde çalışacağı garanti edilmez.
- Simülatör işlevi simülasyon belleğini kullanarak G/Ç modülleriyle veri girişini/çıkışını yürütür. Bu işlev bazı komutları, işlevleri ve cihaz belleğini desteklemez. Bu nedenle, simülatör işleviyle elde edilen çalışma sonuçları gerçek CPU modülüyle elde edilenlerden farklı olabilir.

Sıradaki sayfada, simülatör işlevini simülasyon penceresiyle kullanmayı deneyin.

4.1.1

CPU Modülünü Kullanmadan bir Programın Hatalarını Ayıklama

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

4 Y0 Y2

Robot start signal Stop lamp

6 [END]

Unlabeled L02 Host Station 6/75Step

Artık simülasyon özelliğini kullanmayı öğrendiniz.
İlerlemek için  düğmesini tıklayın.

4.1.2

Bir G/Ç Cihazının Durumunu Değiştirme

G/Ç cihazının bağlandığı bir CPU modülüyle veya simülatör işlevini kullanarak bir sekans programının hatalarını ayıklarken, G/Ç cihazının AÇIK/KAPALI durumunu değiştirmek için **Forced Input Output Registration/Cancellation** işlevini kullanın. G/Ç cihazlarının durumu, yazılım ile zorla AÇIK veya KAPALI olarak değiştirilebilir.

(MELSEC-Q ve MELSEC-L serisi): "Forced Input Output Registration/Cancellation" (Zorunlu Giriş Çıkış Kaydı/İptali) ekranından
(MELSEC-F serisi): "Modify Value" (Değeri Değiştir) ekranından

No.	Device	ON/OFF
1	X05	ON
2	X07	OFF
3		
4		
5		
6		
7		
8		
9		
10		
11		
12		
13		
14		
15		
16		

Device/Label	Data Type	Setting Value
X06	Bit	ON

Forced Input Output Registration/Cancellation
(Zorunlu Giriş Çıkış Kaydı/İptali) ekranı (MELSEC-Q ve MELSEC-L serisi)

Modify Value (Değeri Değiştir) ekranı
(MELSEC-F serisi)

Diğer cihazların durumlarını değiştirmek için

Bir sözcük cihazının mevcut cihazını veya dâhili bir rölenin AÇIK/KAPALI durumunu değiştirmek için **geçerli değeri değiştirme işlevini** kullanın.
Ayrıntılar için, kılavuza başvurun.

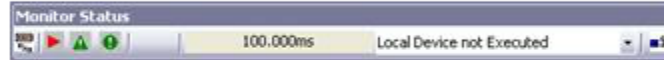
4.1.3

Cihaz Durumunu İzleme

Simülasyon başlatıldığında, izleme otomatik olarak başlar. Gerçek bir programlanabilir denetleyici CPU'suna bağlıken izleme moduna girmek için, Online, Monitor ve daha sonra Start Monitoring öğelerine tıklamanız yeterlidir. Veya klavye kısayolu F3 kullanılabilir.

İzleme modundayken, programda kullanılan tüm cihazların değerleri ve durumu program kodunun üstünde görülebilir. Bu sayede kullanıcı "forced input output registration/cancellation" işlevinin etkileri dâhil olmak üzere değerlerdeki değişimleri görebilir.

Ek olarak, **Monitör Durumu** çubuğu görüntülenir ve CPU veya sanal CPU durumunu belirlemek üzere temel bilgileri içerir. **Monitör Durumu** çubuğunun sunduğu bilgileri daha iyi anlamak için aşağıdaki tabloya başvurun.

CPU modülüne bağlıken**Simülasyon işlevi kullanılırken**

Durum	Simge/gösterge	Açıklama
Bağlantı durumu	CPU modülüne bağlıken	CPU modülüyle bağlantı durumunu veya simülasyon işlevini görüntüler.
	Simülasyon işlevi kullanılırken	
RUN/STOP durumu	RUN	CPU'nun çalışma durumunu görüntüler (RUN veya STOP).
	STOP	
ERR. durumu	ERR. kapalı	CPU modülünün hata durumunu görüntüler.
	ERR. açık	
	↔ ERR yanıp sönüyor	
USER durumu	USER kapalı	CPU modülünün kullanıcı hata durumunu görüntüler.
	USER açık	
	↔ USER yanıp sönüyor	
Tarama süresi	0.000ms	CPU modülü izlenirken maksimum tarama süresini görüntüler.
Desteklenmeyen komut varlık/yokluk durumu	Desteklenmeyen komut mevcut.	Simülasyon işlevi yürütülürken desteklenmeyen bir komutun var olup olmadığını görüntüler. Simgeye tıklandığında Unsupported Instruction/Device penceresi açılır.
	Desteklenmeyen komut mevcut değil.	

4.1.3

Cihaz Durumunu İzleme

İzleme modundayken, programdaki tüm cihazların geçerli durumu görünür hale gelir.

Bit cihaz durumunun gösterimi (AÇIK/KAPALI)

İzleme sırasında AÇIK/KAPALI durumu aşağıda gösterildiği gibi görüntülenir.

KAPALI
durumu



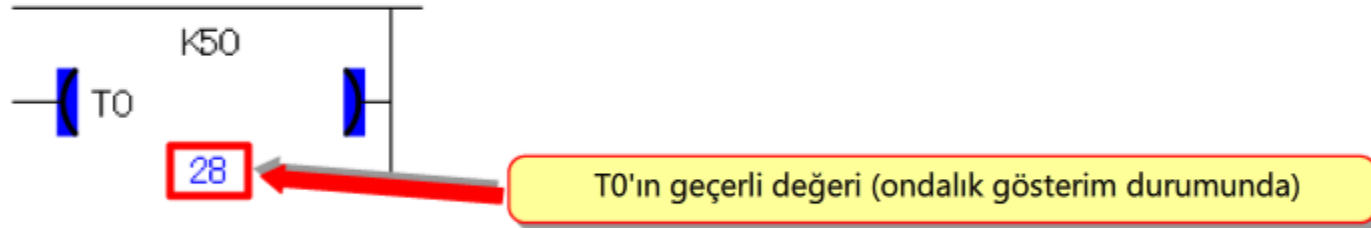
AÇIK
durumu



* Bu gösterim şekli sadece SET, RST, PLS, PLF, SFT, SFTP, MC ve kontak tipi karşılaştırma komutları için geçerlidir. RST komutu için sadece AÇIK/KAPALI durumunun görüntülediğine dikkat edin.

Sözcük cihazının geçerli değer gösterimi (ondalık/onalılık düzende gösterim)

İzleme sırasındaki geçerli değer aşağıda gösterildiği gibi görüntülenir.



Sadece belirli cihazların izlenmesi

Çok büyük veya karmaşık bir program izlenirken, sadece belli ilgili cihazların izlenmesi faydalı olabilir. Bunun gerçekleşmesi için, GX Works2'de kullanıcının ilgilendiği cihazları kolayca ekleyebilmesini, bunların durumu görmesini ve izleme sırasında değerlerini değiştirmesini sağlayan izleme pencereleri bulunur. Ayrıntılar için, GX Works2 Kullanım Kılavuzuna (Ortak) başvurun.

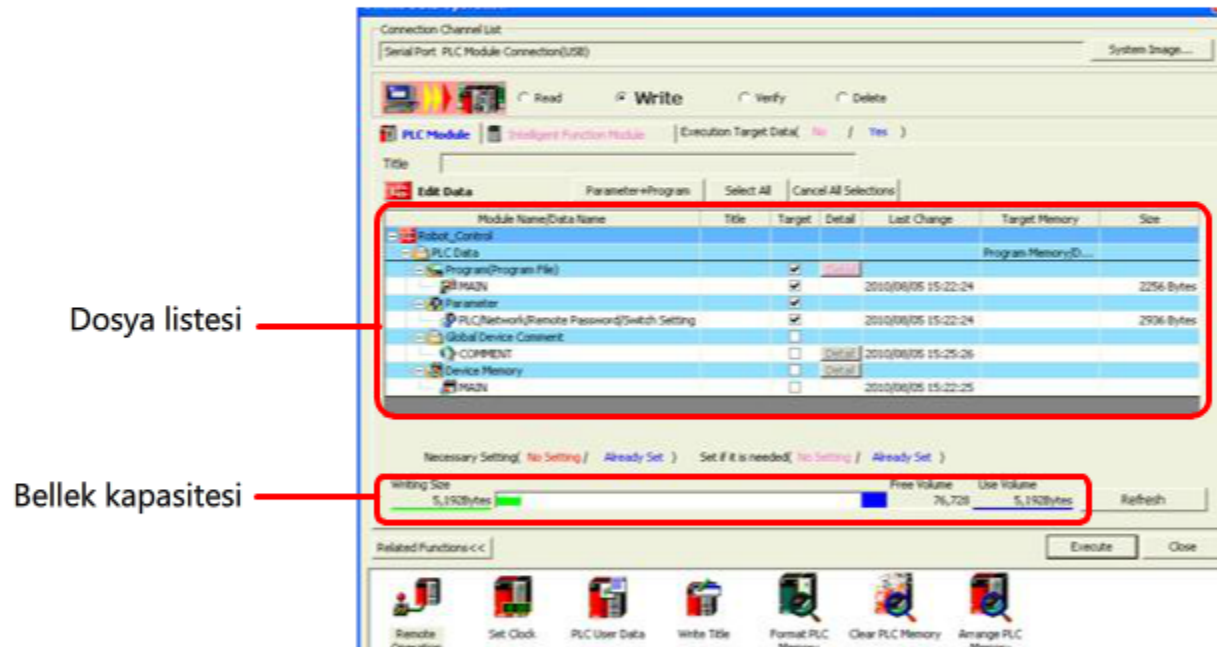
Watch 1					
Device/Label	Current Value	Data Type	Class	Device	Comment
X7	--	Bit		X7	Door open sensor
Y0	--	Bit		Y0	Robot start signal
Y1	--	Bit		Y1	Operation lamp
Y0	--	Bit		Y0	Robot start signal
Y2	--	Bit		Y2	Stop lamp
Y0	--	Bit		Y0	Robot start signal

4.2

CPU Modülüne Program Yazma

Gerçek bir CPU modülünü kullanarak hata ayıklama yapmadan önce, CPU'yu **STOP moduna** getirin, CPU ile bağlantı kurulduğundan emin olun ve program ve parametreleri program belleğine yazın.

Aşağıdaki ekran resminde görüldüğü üzere, **Write to PLC** penceresinin ana işlevleri kullanıcının yazılması istenen dosyaları seçmesini, bunların konumunu belirlemesini ve CPU'nun bellek kapasitesini onaylamasını sağlar. Dosya listesinin üzerindeki üç düğme kullanıcının yazılması istenen dosyaları hızlıca seçmesine imkan sağlar. Aşağıdaki simülasyonda kullanılan en yaygın düğme "**Parameter+Program**"dır.



Sıradaki sayfada, simülasyon penceresini kullanarak CPU modülüne yazmayı deneyin.

4.2

CPU Modülüne Program Yazma

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 opening sensor Y0 Robot start signal Y1 Operation lamp

4 Y0 Robot start signal Y2 Stop lamp

6 END

Unlabeled L02 Host Station 6/75Step

Program artık PLC modülüne yazılmıştır.
İlerlemek için  düğmesini tıklayın.

4.3

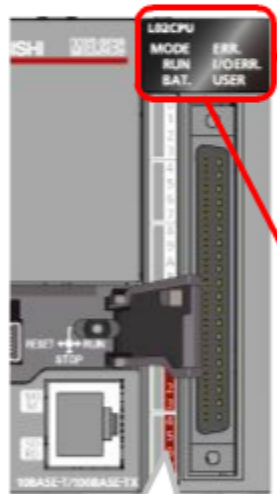
Yazılmış Programları Etkinleştirme

(MELSEC-F serisi): Aşağıdaki işleme gerek yoktur.

(MELSEC-Q ve MELSEC-L serisi): Aşağıdaki işlem gereklidir.
CPU modülüne bir program yazdıktan sonra, CPU modülünü **sıfırlayın**.
CPU modülü sıfırlanmadığı takdirde yazılan programlar etkinleşmez.

* Hata ayıklama için simülatör işlevinin kullanılması durumunda bu işleme gerek yoktur.

CPU modülünü aşağıdaki gibi sıfırlayın:



RESET/STOP/
RUN düğmesi

- (1) CPU modülünün ön panelindeki RESET/STOP/RUN düğmesine basıp RESET konumunda tutun (1 saniye veya daha fazla süreyle).
[Sıfırlama devam ediyor]

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

MODE : Yeşil renkli Açık
RUN : Kapalı
ERR. : Yanıp sönüyor

1 saniye veya daha
uzun süre basılı tutun.



- (2) Yanan MODE LED'i ve yanıp sönen ERR. LED'i birlikte kapandıktan sonra düğmeyi bırakın.
[Sıfırlama tamamlanmış]

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

MODE : Yeşil renkli Açık
RUN : Kapalı
ERR. : Kapalı

- (3) Düğme sıfırlama işlemini tamamlamak üzere STOP konumuna geri döner.

4.4

Programları Çalıştırma

MELSEC-Q ve MELSEC-L serisi

Sıfırlama tamamlandıktan sonra, programı çalıştırın.

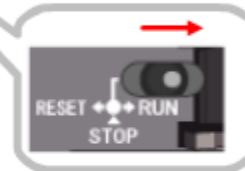
Programı çalıştırmak için CPU modülünü aşağıdaki gibi **RUN durumuna** getirin.

* Hata ayıklama için simülatör işlevinin kullanılması durumunda bu işleme gerek yoktur.



- (1) CPU modülünün ön panelindeki RESET/STOP/RUN düğmesini RUN konumuna getirin.
LED gösterimi STOP durumunda

L02CPU	MODE	ERR.	
RUN	I/OERR.	MODE	: Yeşil renkli Açık
BAT.	USER	RUN	: Kapalı



- (2) RUN LED'i yeşil renkli yanıyorsa, program normal çalışmaktadır.
LED gösterimi RUN durumunda

L02CPU	MODE	ERR.	
RUN	I/OERR.	MODE	: Yeşil renkli Açık
BAT.	USER	RUN	: Yeşil renkli Açık

MELSEC-F serisi

Ana üniteye bir program yazdıktan sonra, programı çalıştırmak için aşağıda gösterildiği şekilde ana üniteyi RUN durumuna getirin. (Sıfırlama işlemine gerek yoktur.)

- (1) Ana ünitenin ön panelinde yer alan RUN/STOP düğmesini RUN konumuna getirin.



LED gösterimi STOP durumunda

- (2) RUN LED yanıyorsa, program normal çalışmaktadır.



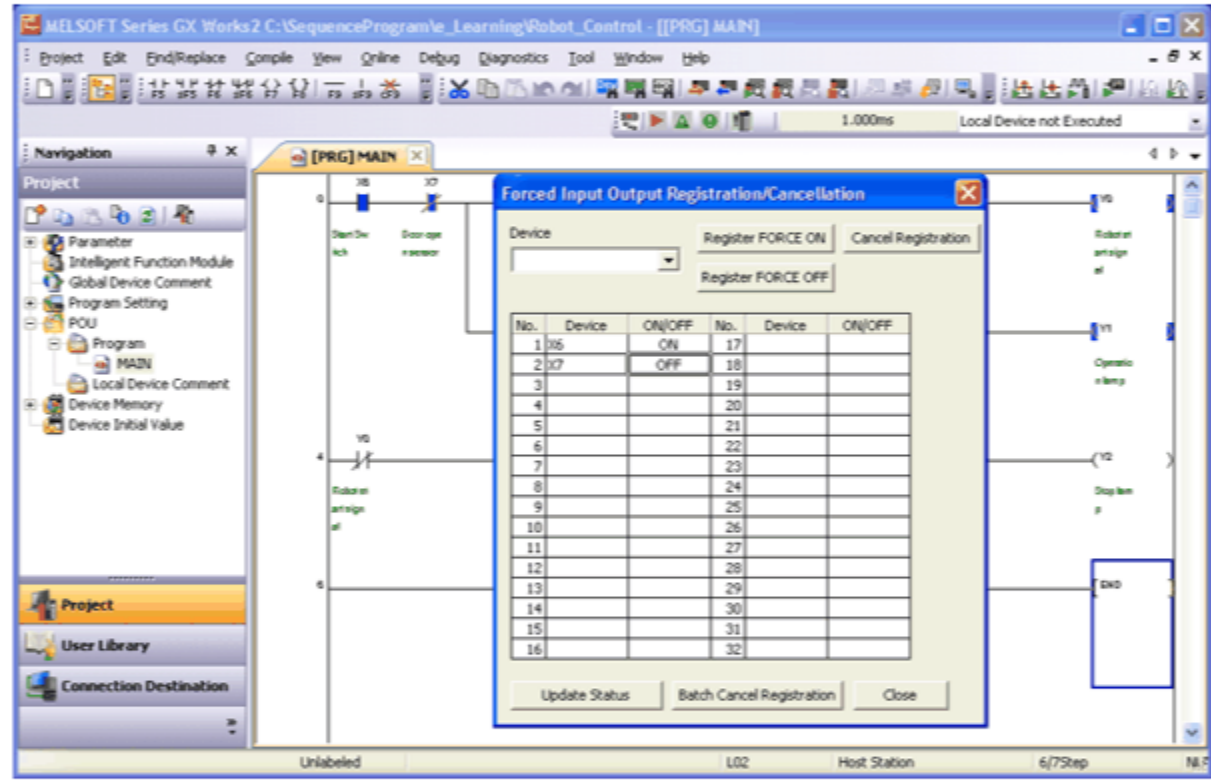
LED gösterimi RUN durumunda

4.5

การแก้ไขความผิดพลาดโปรแกรม

หลังจากที่รันโมดูล CPU แล้ว ใช้ฟังก์ชันบังคับการลงทะเบียน/การยกเลิกอินพุทเอาต์พุทเพื่อเปลี่ยนสถานะของอุปกรณ์แต่ละรายการและตรวจสอบผลลัพธ์ (เอาต์พุท) ของ Ladder

(ตัวอย่างหน้าจอของ MELSEC-Q and MELSEC-L ซีรีส์)



ในหน้าต่อไป ให้ลองแก้ไขความผิดพลาดโดยใช้หน้าต่างจำลอง

4.5

Programların Hatalarını Ayıklama

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

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

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 X6 X7 Y0 Y1 Y2

Start Switch Door open sensor Robot start signal Operation lamp

4 Y0 Y2

Robot start signal Stop lamp

6 END

Unlabeled L02 Host Station 6/75Step

Programın hatalarını ayıklama işlemi tamamlanmıştır.
İlerlemek için  düğmesini tıklayın.

4.6

PLC Sisteminin Çalışmasını Kontrol Etme

Program hatalarını ayıklama işlemi tamamlandıktan sonra, son olarak çalışmasını kontrol etmek için programı gerçek PLC sistemine yazın.

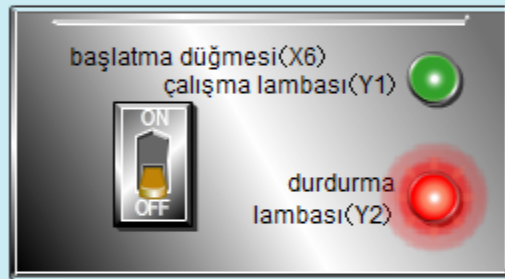
Gerçek G/Ç ekipmanının tasarlanan şekilde çalıştığını doğrulamak için ekipmanı çalıştırın.

G/Ç ekipmanını çalıştırırken bile, her cihazın durumu GX Works2'nin izleme işlevi kullanılarak kontrol edilebilir.

Örnek sistem çalışması

Kırmızı dairenin içine tıklayın

Robot kontrol paneli



Güvenlik bariyerindeki robot

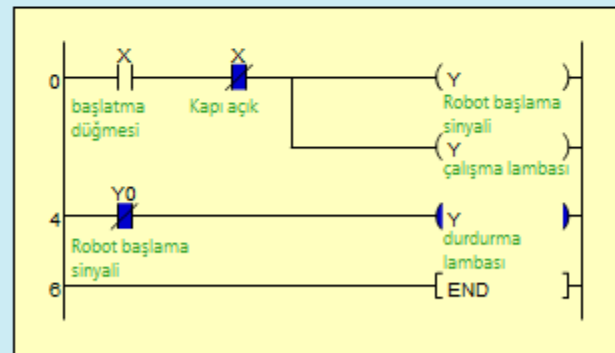


Başlatma düğmesi (X6) KAPALI olarak ayarlandığında, robot başlatma sinyali (Y0) kapanarak robotun çalışmasını durdurur. Aynı anda, kontrol panelindeki çalışma lambası (Y1) kapanır ve durdurma lambası (Y2) açılır.

Yeniden Oynat



Geri



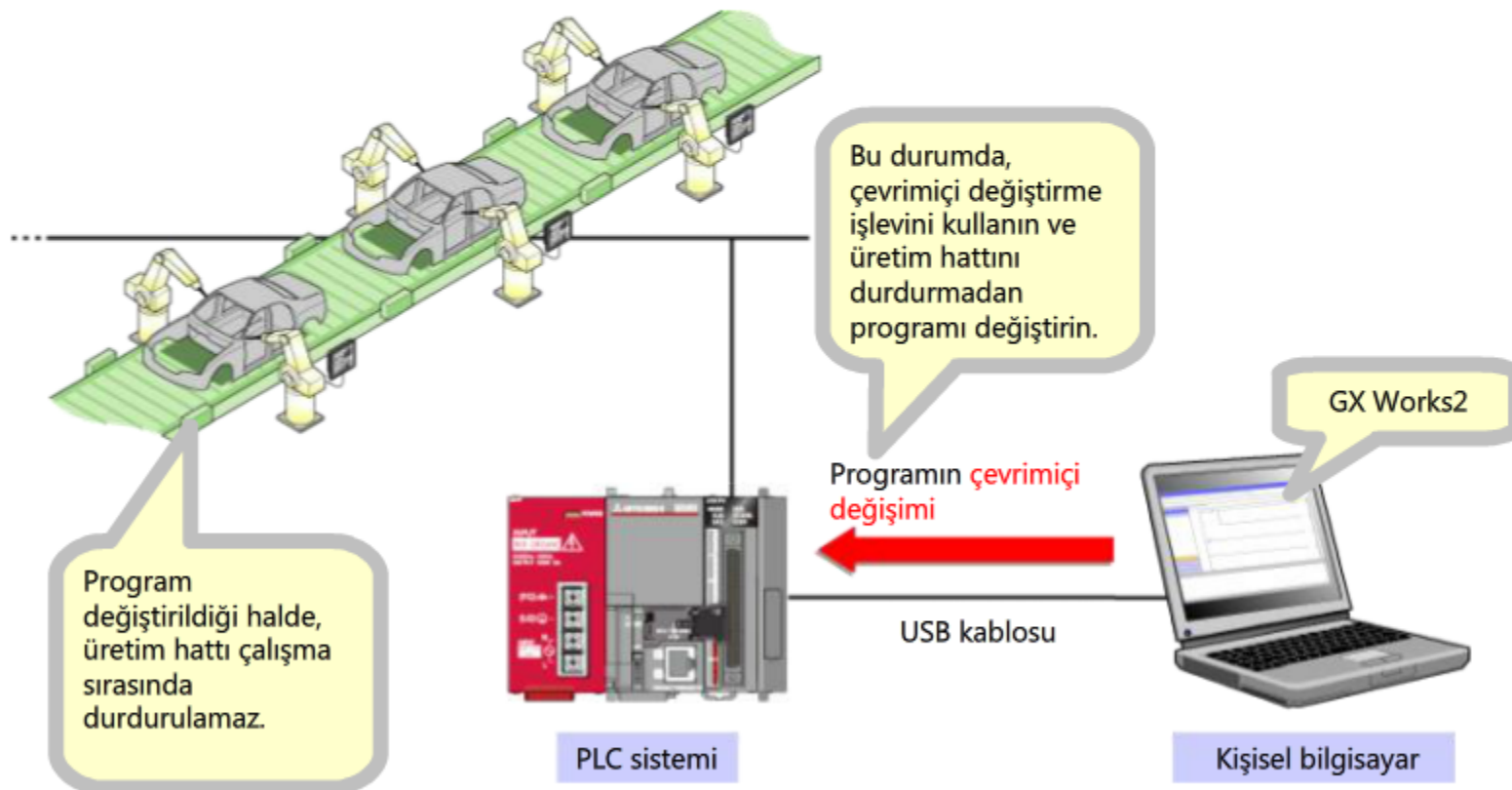
4.7

PLC Sistemini Çalıştırma

Çalışma doğrulama işlemi tamamlandığında, çalışmayı başlatmak için PLC sistemini çalıştırın.

Programın çalışan sistemde değiştirilmesi gerekiyorsa

Sistem çalıştırılmaya başlandıktan sonra yazılım hatalarını düzeltme veya sistemi genişletme gibi program değişikliklerinin yapılması gerekebilir. Normalde, değiştirilmiş bir programı yazmak için sistemin (CPU modülü) durdurulması gerekir. Bu problemi çözmek için, GX Works çalışan CPU modülünü durdurmadan program yazmak için kullanılan çevrimiçi bir değiştirme işlevi sunar.

Örnek: 24 saat çalışan otomobil üretim hattı

Sıradaki sayfada, çevrimiçi değiştirme işlevini simülasyon penceresini kullanarak deneyin.

4.7

PLC Sistemini Çalıştırma

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

Unlabeled

Y0 Robot start signal

Y1 Operation lamp

Y2 Stop lamp

END

Değiştirilen programın çevrimiçi değişimi tamamlanmıştır.
İlerlemek için [Play] düğmesini tıklayın.

Unlabeled L02 Host Station 2/75Step

4.8**Sonuç**

Programlanabilir denetleyici yazılım tasarımına yönelik temel açıklamalar burada tamamlandı.

Bu kursta şunları öğrendiniz:

- PLC sisteminin programlanması için gerekli öğeler
- Açıklamaların kullanımı dâhil program tasarımına yönelik bazı temel kılavuz ilkeler
- GX works2'nin temel PC programlama görevlerinin gerçekleştirilmesi için kullanımı
- PLC programlarının hatalarını ayıklamada kullanılan birkaç teknik

Test**Son Test**

Artık **PLC GX Works2 Temel Bilgileri** kursundaki tüm dersleri tamamladığınızdan, son teste girmeye hazırsınız. Ele alınan konulardan herhangi birini tam anlamadıysanız, lütfen bu konuları gözden geçirmek için bu fırsatı değerlendirin.

Bu Son Testte toplam 5 soru (15 madde) yer almaktadır.

Son testi istediğiniz sayıda uygulayabilirsiniz.

Testin puanlanması

Cevabı seçtikten sonra, **Cevapla** düğmesini tıkladığınızdan emin olun. Cevapla düğmesini tıklamadan ilerlemeniz durumunda cevabınız kaybolur. (Cevaplanmamış soru olarak değerlendirilir.)

Puan sonuçları

Doğru cevap sayısı, soru sayısı, doğru cevapların yüzdesi ve başarılı/başarısız sonucu puan sayfasında görüntülenir.

Doğru cevaplar : 2

Toplam soru : 9

Yüzde : 22%

Testi geçebilmek için, soruların %60'ını doğru cevaplamanız gerekir.

[Devam Et](#)[İncele](#)[Tekrar Dene](#)

- Testten çıkmak için **Devam Et** düğmesini tıklayın.
- Testi incelemek için **İncele** düğmesini tıklayın. (Doğru cevap kontrolü)
- Testi tekrar yapmak için **Tekrar Dene** düğmesini tıklayın.

Test**Son Test 1**

Sorumlu olduğunuz program başka bir kişi tarafından devralınmıştır ve bu kişi programa ait kontrol öğelerini anlamakta zorlanmıştır. Bu sorunu önlemek için alınması gereken karşı önlem nedir?

- ☐ GX Works2'nin açıklama işlevini kullanmak, programa uygun bir başlık ve açıklama vermek.
- ☐ Kontrol öğelerinin yeni kişiye sözlü olarak açıklamak.
- ☐ Karmaşık, büyük programları devralmaktan kaçınmak.
- ☐ Program ile birlikte G/Ç cihazları ve cihaz numaraları için uyum tablosunu devretmek.

[Puan](#)[Geri](#)

Test**Son Test 2**

Doğru programlama prosedürünü tamamlayın.

Adım 1 Program tasarımı

Adım 2 (Q1)

Adım 3 (Q2)

Adım 4 Programları dönüştürme

Adım 5 Projeleri kaydetme

Adım 6 (Q3)

Adım 7 (Q4)

Adım 8 CPU modülünü çalıştırma (RUN)

Adım 9 (Q5)

Adım 10 PLC Sisteminin Çalışmasını Kontrol Etme

Puan

Geri

Test**Son Test 3**

Bir program tamamlandıktan sonra yapılması gerekenlere ait açıklamayı tamamlamak için boşlukları doldurun.

Bir program yazıldıktan sonra, beklenen şekilde çalıştığından emin olmak için test edilmelidir.

A () (yazılan kodun amaçlanan şekilde işlev göstermemesine)

a () denir ve sebebini tespit edip düzeltme sürecine

() denir.

Bu süreç, program oluşturma sürecinde temel bir adımdır.

Puan

Geri

Test**Son Test 4**

Her GX Works2 işlevinin ilgili uygulamasını seçin.

İş lev	Uygulama
Simülasyon	--Select--
Zorlamalı giriş-çıkış kaydı/iptali	--Select--
Geçerli değeri değiştirme	--Select--
Ladder monitörü	--Select--
İzleme	--Select--

[Puan](#)[Geri](#)

Test**Son Test 5**

Çevrimiçi deęiřtirme iřleviyle ilgili doęru aıklamayı sein.

- ☐ İřlev CPU'yu otomatik olarak durdurur, programı CPU'ya yazar ve daha sonra CPU'yu otomatik olarak alıřtırır.
- ☐ İřlev, alıřan CPU modlndeki programı GX Works2 tarafından aılan program ile karřılařtırır.
- ☐ İřlev, alıřan CPU modln gvenli řekilde durdurduktan sonra CPU modlne bir program yazabilir.
- ☐ İřlev, alıřan CPU modln durdurmadan modle bir program yazabilir.

Puan

Geri

Test

TEST PUANI



Son Testi tamamladınız. Sonuçlarınız aşağıdaki alanda gösterilmektedir.
Son Testi sonlandırmak için, sonraki sayfaya geçin.

Doğru cevaplar : 0

Toplam soru : 5

Yüzde : 0%

[Devam Et](#)[İncele](#)[Tekrar Dene](#)

Testte başarısız oldunuz.

PLC GX Works2 Temel Bilgileri kursunu tamamladınız.

Bu kursa katıldığınız için teşekkür ederiz.

Derslerden keyif almış olmanızı ve bu kursta edindiğiniz bilgilerin
gelecekte faydalı olmasını umarız.

Kursu istediğiniz kadar çok gözden geçirebilirsiniz.

İncele

Kapat