

# PLC

## Základy softwaru GX Works2

Tento výukový kurz (e-learning) je navržen pro uživatele, kteří software GX Works2 poprvé používají k tvorbě sekvenčních programů.

Tento kurz poskytuje základní znalosti používání softwaru GX Works2 k programování, ladění a kontrole provozu programovatelného kontroléru (PLC). Kurz je určen pro uživatele vytvářející sekvenční programy pro kontroléry řad MELSEC-Q, MELSEC-L a MELSEC-F.

Obsah tohoto kurzu je následující.  
Doporučujeme, abyste začali 1. kapitolou.

### **1. kapitola – Metoda řízení systému PLC**

Představuje programovací jazyk a software sloužící k programování.

### **2. kapitola – Návrh programů**

Zde se naučíte navrhovat programy na základě položek řízení a konfigurace hardwaru.

### **3. kapitola – Programování**

Zde se naučíte programovat pomocí specializovaného softwaru GX Works2.

### **4. kapitola – Ladění**

Zde se naučíte zapisovat sekvenční programy do modulu CPU a ladit je.

### **5. kapitola – Závěrečný test**

Požadované skóre: 60% a vyšší.

## Úvod Používání tohoto elektronického výukového nástroje



Přejdete na následující stránku		Zobrazí následující stránku.
Zpět na předchozí stránku		Zobrazí předchozí stránku.
Přesunutí na požadovanou stránku		Zobrazí se „Obsah“, pomocí kterého můžete přejít na požadovanou stránku.
Ukončit výuku		Ukončí výuku. Zavřou se všechna okna, včetně výukového okna a okna „Obsah“.

**Bezpečnostní opatření**

Pokud se učíte používáním aktuálních produktů, pozorně si prosím přečtěte bezpečnostní opatření v odpovídajících návodech.

**Preventivní opatření v tomto kurzu**

- Zobrazené obrazovky vámi používané verze softwaru se mohou lišit od těch, které jsou vyobrazeny v tomto kurzu.

# 1. kapitola Metoda řízení systému PLC

Tento kurz je určen pro osoby pracující s technickým softwarem. Zabývá se některými základními koncepty správy systémů řady MELSEC-Q, L a F.

Software GX Works2 (GXW2) využívá mezinárodně standardizované programovací jazyky, jako je sekvenční funkční schéma (SFC), seznam instrukcí (IL)\*1, žebříkový diagram, diagram funkčních bloků (FBD)\*2 a strukturovaný text (ST).

Programy jsou vyvíjeny pomocí osobního počítače používajícího „technický software“, GX Works2, a jsou obvykle zapisovány do CPU programovatelného kontroléru prostřednictvím USB, ethernetového\*3 nebo sériového kabelu. Programování modulu CPU lze přepisovat neomezeně tolikrát, kolikrát je zapotřebí k provedení požadovaných změn.

\*1 Budoucí plán pro software GX Works2.

\*2 Aktuálně strukturovaný žebříkový diagram v softwaru GX Works2, plánuje se soulad s normou IEC.

\*3 Ethernet je registrovanou ochrannou známkou společnosti Xerox Corp.

Osobní počítač  
(programovací software)

Systém PLC

Automobilová výrobní linka



V tomto kurzu je pro příklad programování použita žebříková logika (jeden z nejpůlárnějších programovacích jazyků pro kontroléry PLC). Ačkoli je v příkladu použit kontrolér PLC řady L, obsah tohoto kurzu je stejně platný také pro systémy řady Q.

Základní metoda řízení je stejná jako u řady MELSEC-F, některé operace a funkce jsou však odlišné.

Tento elektronický výukový kurz popisuje kroky návrhu softwaru (znázorněné zeleně) nezbytné k implementaci systému programovatelného kontroléru.

### Návrh hardwaru

(1) Návrh systému ..... Kurz základů MELSEC-Q/MELSEC-L

(2) Výběr produktu ..... Kurz základů MELSEC-Q/MELSEC-L

(3) Předběžná příprava ..... Kurz základů MELSEC-Q/MELSEC-L

(4) Instalace a zapojení ..... Kurz základů MELSEC-Q/MELSEC-L

(5) Kontrola zapojení ..... Kurz základů MELSEC-Q/MELSEC-L

### Návrh softwaru

(6) Návrh programu ..... 2. kapitola

(7) Programování ..... 3. kapitola

(8) Ladění ..... 4. kapitola

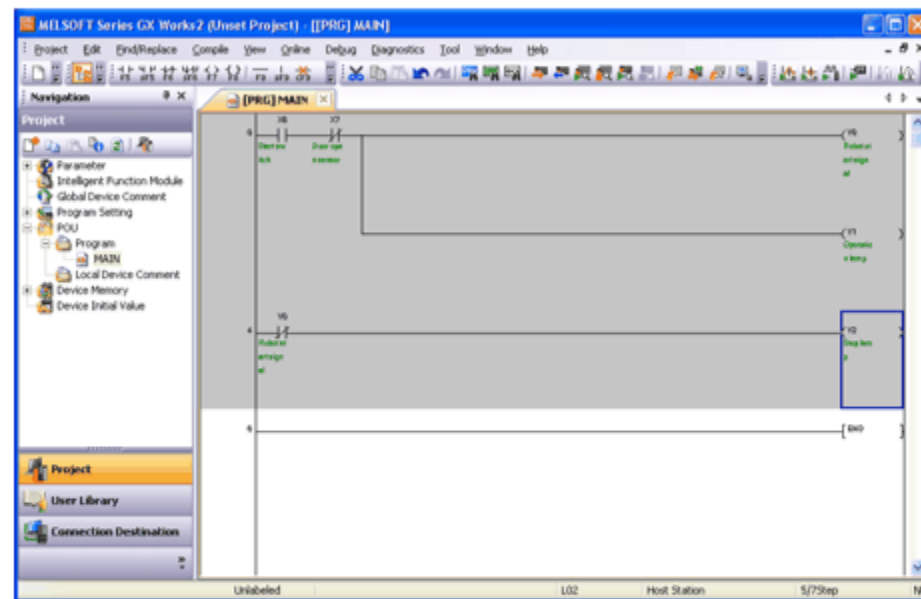
(9) Provoz

**Obsah tohoto kurzu**

Tento kurz se zaměřuje na použití technického softwaru pro programovatelné kontroléry GX Works2 k vývoji ukázkového systémového programu.

Níže je uvedeno několik hlavních funkcí softwaru GX Works2.

- Správa paměti a souborů
- Vývoj programů pro programovatelný kontrolér
- Správa dokumentace programu (komentáře apod.)
- Čtení a zápis dat (obzvláště programů) z/do modulu CPU
- Ověřování funkčnosti programu
  - Softwarová simulace hardwaru PLC
  - Vynucení zapnutí nebo vypnutí I/O
  - Monitorování stavu I/O a adresy paměti
- Provádění údržby a odstraňování problémů





## 1.3

## Konfigurace obrazovky softwaru GX Works2

Níže je uvedena konfigurace obrazovky GX Works2.

Umístěním kurzoru myši do červeného rámečku zobrazíte odpovídající funkci.

The screenshot displays the MELSOFT Series GX Works2 software interface. The main window shows a ladder logic diagram for a program named [PRG] SUB. The diagram includes the following components:

- Inputs:** X6 (Start Switch), X7 (Door open sensor).
- Outputs:** Y0 (Robot start signal), Y1 (Operation lamp), Y2 (Stop lamp).
- Logic:** A normally open contact for X6 and a normally closed contact for X7 are connected in series to output Y0. A normally open contact for Y0 is connected to output Y2.

A red rectangular box highlights a specific component in the diagram, which is a normally open contact labeled Y0 (Robot start signal). This component is also visible in a separate window titled [PRG] MAIN, which is overlaid on the main window. The [PRG] MAIN window shows a similar ladder logic diagram with the same components. The status bar at the bottom of the software interface indicates "Unlabeled", "L02", "Host Station", "5/7Step", and "M.S".

## 1.4

## Postup tvorby sekvenčního programu



Vytvořte sekvenční program podle následujícího postupu.

(1) Návrh programu ..... 2. kapitola



(2) Vytváření projektů ..... Oddíl 3.1



(3) Programování ..... Oddíl 3.2



(4) Konverze programů ..... Oddíl 3.4



(5) Zápis do modulu CPU ..... Oddíl 4.2



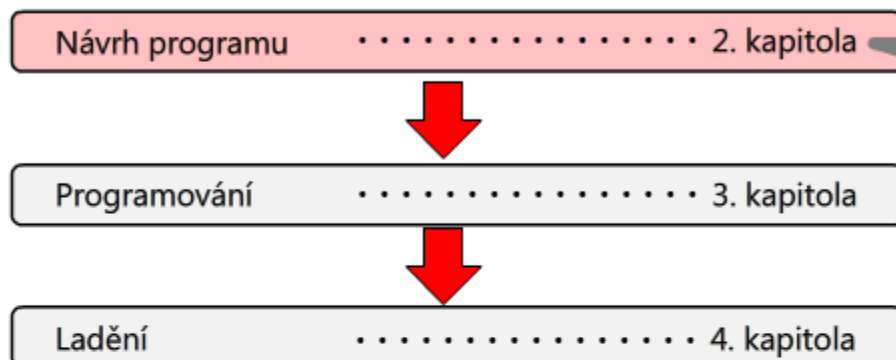
(6) Resetování a spouštění modulu CPU ..... Oddíl 4.3, 4.4



(7) Ladění ..... Oddíl 4.5

## 2. kapitola Návrh programu

V kapitole 2 se dozvíte, jak navrhovat programy, včetně definování obsahu řízení a konverze na program.

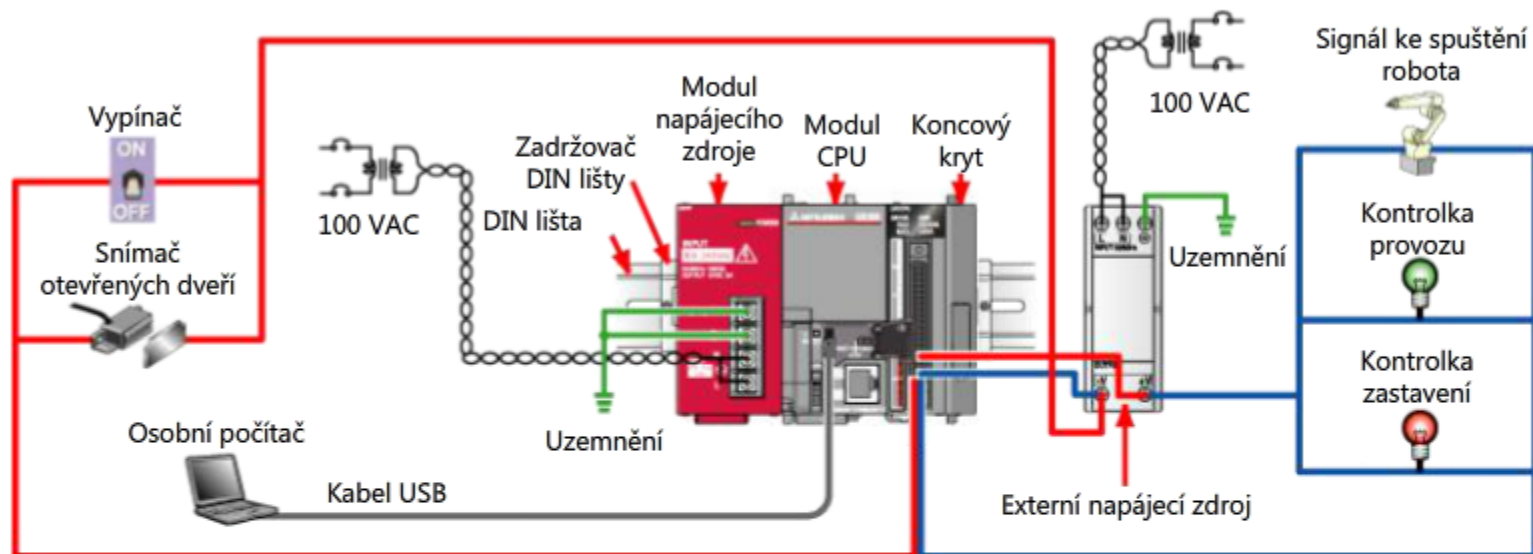


### Postup výuky ve 2. kapitole

- 2.1 Konfigurace hardwaru ukázkového systému použitého k výuce
- 2.2 Definice položek řízení
- 2.3 Tvorba korespondenční tabulky proměnných I/O a čísel proměnných
- 2.4 Návrh programu

## 2.1 Konfigurace hardwaru ukázkového systému použitého k výuce


V rámci tohoto kurzu vytvoříte systém PLC (dále nazývaný „ukázkový systém“), který spustí robota dle postupu. Diagram konfigurace hardwaru ukázkového systému je uveden níže se seznamem hardwarových komponent.



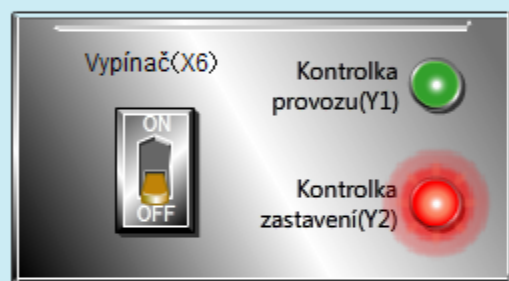
Položka	Komponenta	Model	Popis
Systém PLC	Modul napájecího zdroje	L61P	Zajišťuje napájení modulů, včetně modulu CPU a I/O.
	Modul CPU	L02CPU	Řídí systém PLC.
	Koncový kryt	L6EC	Je připojen k pravé straně bloku systému.
	Kabel USB	MR-J3USBCBL3M	Připojuje osobní počítač, na kterém je nainstalován software GX Works2, k modulu CPU.
	Osobní počítač	–	Používá nainstalovaný software GX Works2.
Externí napájecí zdroj	–	–	Zajišťuje napájení externích zařízení I/O.
Externí zařízení I/O	Vypínač	–	Zapnutím spustíte řízení.
	Snímač	–	Detekuje, zda jsou dveře otevřeny či zavřeny.
	Robot	–	Pracuje podle signálů řízení.
	Dvě kontrolky	–	Svítil podle provozního stavu.

Prvním krokem k návrhu programu je identifikace zařízení, která budou ovládána, a proměnných I/O nezbytných k danému ovládání. V ukázkovém systému je předvedeno řízení operace spuštění a zastavení provozu robota. Robot se nebude moci spustit, pokud budou dveře bezpečnostní klece otevřeny, nebo bude zastaven, když se dveře otevřou během provozu. Jak bude ukázkový systém fungovat, lépe pochopíte z níže uvedené animace.

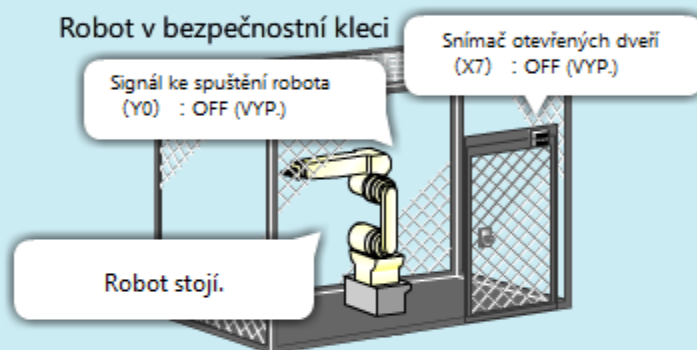
### Provoz ukázkového systému

 Klikněte do červeného kroužku

Ovládací panel robota



Robot v bezpečnostní kleci



Když přepnete **vypínač (X6)** do polohy OFF (VYP), **signál spuštění robota (Y0)** se deaktivuje a zastaví činnost robota. Současně zhasne **kontrolka provozu (Y1)** na ovládacím panelu a rozsvítí se **kontrolka zastavení (Y2)**.

Znovu



Předchozí

## 2.3 Tvorba korespondenční tabulky proměnných I/O a čísel proměnných

Je vhodné vytvořit tabulku zahrnující všechny proměnné I/O a registry používané v kontroléru PLC a jejich odpovídající informace pro jakýkoli vytvářený program. Lze tak omezit riziko vzniku chyb během procesu návrhu a programování a také zvýšit efektivitu programování. Pokud již korespondenční tabulka pro daný systém existuje, například vytvořená osobou, která hardware konfigurovala, využijte ji.

Níže je uvedena korespondenční tabulka pro ukázkový systém používaný v tomto kurzu.

Název proměnné I/O	Č. proměnné	Typ I/O	Typ proměnné	Popis
Vypínač	X6	Vstup	Bit	Tento vypínač spouští a zastavuje provoz robota.
Snímač otevřených dveří	X7	Vstup	Bit	Tento snímač kontroluje, zda jsou dveře bezpečnostní klece robota otevřeny. Když se dveře otevřou, snímač se aktivuje. Když se dveře zavřou, snímač se deaktivuje.
Signál ke spuštění robota	Y0	Výstup	Bit	Když se signál aktivuje, robot zahájí provoz.
Kontrolka provozu	Y1	Výstup	Bit	Tato kontrolka svítí, když je robot v provozu.
Kontrolka zastavení	Y2	Výstup	Bit	Tato kontrolka svítí, když je robot zastaven.

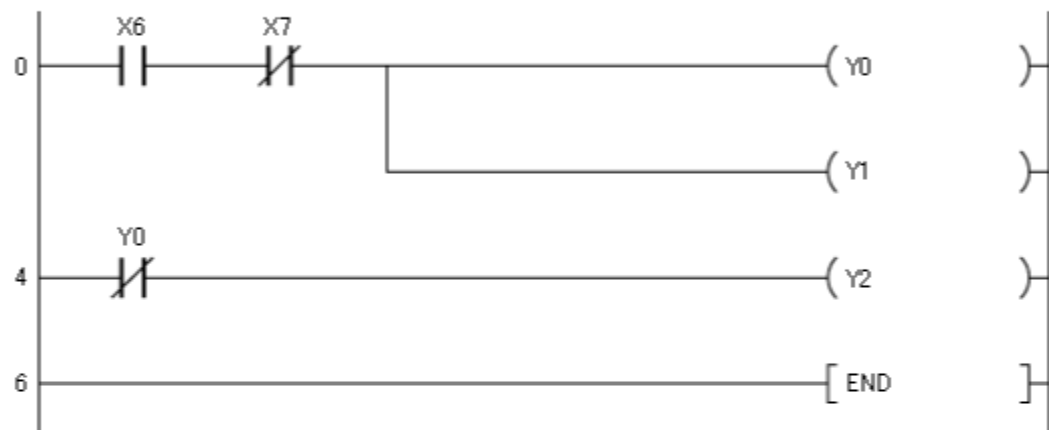
\* Jsou-li použita data typu slovo, v tabulce by měly být zahrnuty úvodní hodnota, rozsah nastavení (horní a dolní limity), typ dat (se znaménkem, reálný apod.) a komentář.  
Tyto informace budou užitečné při návrhu a úpravách programů.

## 2.4

## Návrh programů

Navrhňte program pomocí jazyka žebříkového diagramu na základě položek řízení a korespondenční tabulky I/O. Žebříkový program a korespondenční tabulka I/O navržené pro ukázkový systém jsou znázorněny níže.

## Žebříkový program

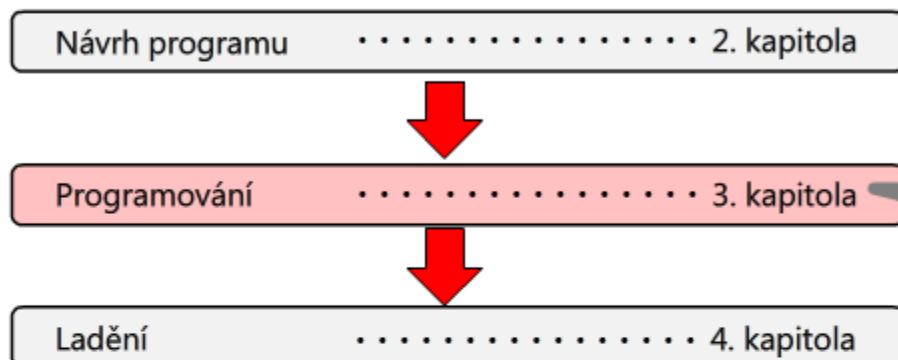


## Korespondenční tabulka I/O

Název proměnné I/O	Typ	Č. proměnné
Vypínač	Vstup	X6
Snímač otevřených dveří	Vstup	X7
Signál ke spuštění robota	Výstup	Y0
Kontrolka provozu	Výstup	Y1
Kontrolka zastavení	Výstup	Y2

## 3. kapitola Programování

Ve 3. kapitole se dozvíte, jak navržený program naprogramovat pomocí softwaru GX Works2.



### Postup výuky ve 3. kapitole

- 3.1 Vytváření projektů
- 3.2 Vytváření programů
- 3.3 Tvorba snadno pochopitelných programů
- 3.4 Konverze programů do spustitelné formy
- 3.5 Ukládání projektů



Prvním krokem při psaní programu je vytvoření projektu. Projekt je sbírkou dat, která software GX Works2 používá ke správě programů. Následující tabulka uvádí hlavní komponenty projektu.

Typ dat	Popis
Program	Zdrojový kód a zkompilovaný kód pro sekvenční operace CPU.
Komentář	Typ dokumentace zobrazené uvnitř programu. Podrobnosti najdete v oddílu 3.3 „Tvorba snadno pochopitelných programů“.
Parametr	Obsahuje většinu nebo všechny informace o nastavení a konfiguraci systému.
Nastavení přenosu	Informace o směrování zapojení nezbytné k navázání komunikace mezi systémem používajícím software GX Works2 a modulem CPU.

### Žebříkový program

Software GX Works2 umožňuje vybírat z následujících dvou typů projektů. Ukázkový program v tomto kurzu používá typ „**simple project**“ (jednoduchý projekt).

Typ projektu	Popis
Simple project	Tento typ projektu je zpětně kompatibilní s projekty softwaru GX Developer. Jednoduché projekty lze později převádět na strukturované projekty, v opačném směru však převod není možný.
Structured project	Tyto projekty umožňují využívat další programovací jazyk – strukturovaný žebříkový diagram. Navíc lze programy rozdělovat na více menších částí, takže je možné často používané kusy kódu snadno modularizovat a opakovaně používat prostřednictvím uživatelské knihovny. Symbolické proměnné lze podobným způsobem modularizovat a opakovaně používat také. To umožňuje zvýšit efektivitu programování a ladění, obzvláště u velmi velkých projektů.

### Symbolické proměnné

Symbolické proměnné jsou uživateli vytvořené názvy, které se stanou aliasy adres proměnných. Lze je používat globálně, lokálně či v rámci celého systému při implementaci společně se softwarem MELSOFT Navigator. Jednoduché projekty lze vytvářet s možností používat symbolické proměnné nebo bez ní. V případě ukázkového projektu symbolické proměnné nebudou použity.

## 3.1 Vytváření projektů

Chcete-li začít vytvářet ukázkový projekt, použijte následující nastavení.

Před vytvořením projektu je třeba znát sérii a model programovatelného kontroléru a také typ projektu, který má být použit.

Položka	Popis
Project type	Typ projektu určuje, jaké funkce budou k dispozici při psaní programů. V tomto případě vyberte možnost „simple project“ (jednoduchý projekt).
Use label	Pokud potřebujete možnost psát programy za použití symbolických proměnných, zaškrtněte tuto položku. Ukázkový program symbolické proměnné nepoužívá. Proto v tomto případě pole nezaškrťávejte.
PLC series	Řada kontroléru PLC určuje, které modely bude možné vybrat v rozevíracím seznamu PLC type (Typ PLC). V tomto případě vyberte možnost „LCPU“.
PLC type	Typ kontroléru PLC určuje, jak bude kompilátor převádět uživatelské programy do strojového kódu. Vyberte model PLC, který budete programovat, v tomto případě „L02“.
Programming language	Programovací jazyk určuje typ prvního automaticky vytvořeného programu (MAIN). Později je možné přidat další programy používající jiné programovací jazyky. V tomto případě vyberte možnost „Ladder“ (Žebříkový diagram).

Podívejte se na další stranu, kde je simulován postup vytvoření nového projektu.

## 3.1

## Vytváření projektů

MELSOFT Series GX Works2

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

Navigation

Project

Project Type: Simple Project

Use Label

PLC Series: LCPU

PLC Type: L02

Language: Ladder

OK

Cancel

Nový projekt bude vytvořen.  
Pokračujte kliknutím na tlačítko [OK]

## 3.2

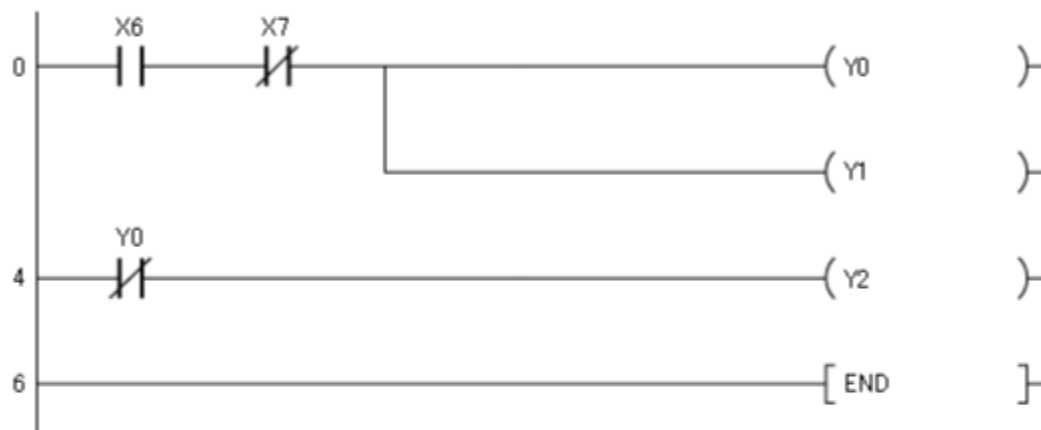
## Vytváření programů

Po vytvoření projektu můžeme vytvořit program.

Vytvořte následující program a osvojte si přitom základní operace (zadáání instrukcí, změna, odstranění, kopírování a vkládání a vkládání/odstraňování řádků).

Níže je znázorněn program navržený pro ukázkový systém z 2. kapitoly.

### Program pro ukázkový systém



Na další stránce zkuste tento program vytvořit v simulovaném okně.

## 3.2

## Vytváření programů



MELSOFT Series GX Works2 (Unset Project) - [[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

Unlabeled

L02

Host Station

0/15step

NL

Y0

Y1

Y2

END

0

X8

X7

Y0

Diagram description: The image shows a ladder logic program in MELSOFT GX Works2. The main window displays a single ladder logic network. The left rail is connected to a normally open contact labeled X8. This is followed by a normally closed contact labeled X7. The right rail of this network is connected to three output coils: Y0, Y1, and Y2. Below this network, there is a normally open contact labeled Y0, which is connected to a coil labeled END. A blue box highlights the END coil. The status bar at the bottom indicates 'Unlabeled', 'L02', 'Host Station', and '0/15step'. A message box in the bottom right corner contains the text: 'Nyní je program žebříkového obvodu dokončen. Pokračujte kliknutím na tlačítko [play icon]'.

Nyní je program žebříkového obvodu dokončen.

Pokračujte kliknutím na tlačítko  .

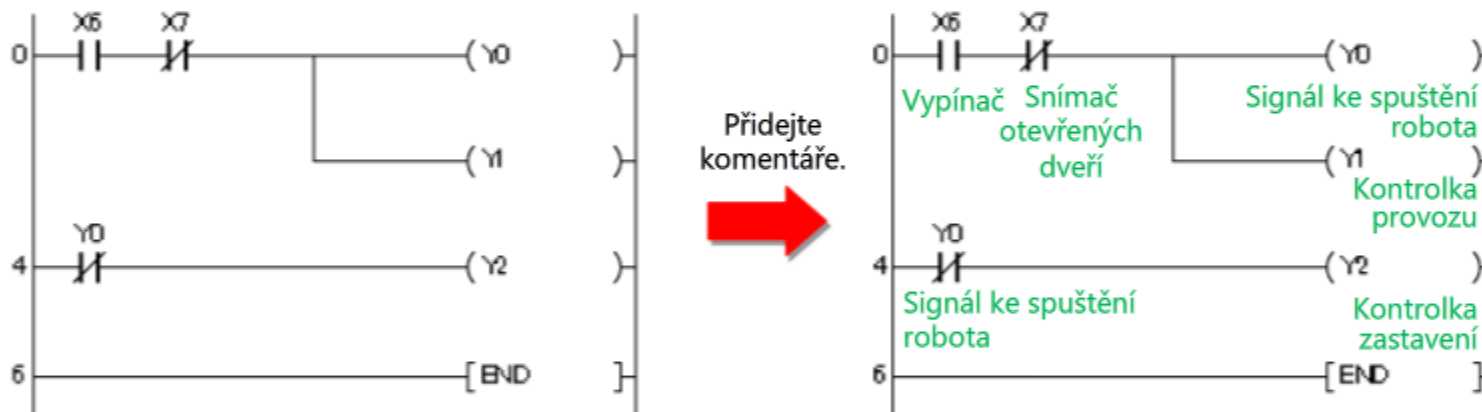
V současném stavu obsahuje vizuální reprezentace programu pouze proměnné, instrukce, řádky a čísla kroků. Při pohledu na složitý program může být náročné rozpoznat, k čemu slouží.

- Je náročné najít chyby v programování, jako jsou například nesprávná čísla proměnných nebo instrukce.
- Celkově je složité provádět provozní analýzu, ladění a rozšiřování programu.
- Pokud se již o údržbu programu nemůže starat původní vývojář, pochopit principy funkce programu může být pro jinou osobu velmi zdoluhavé, nebo dokonce nemožné.

### Protiopatření

Zahrnujte do programů **dokumentaci**, aby mohl kdokoli rychle pochopit, jakým způsobem program pracuje.

V rámci dobrých postupů by měl každý programátor zahrnovat ve svých programech podrobné komentáře, aby měl on sám i jiní lepší přehled o funkci programu.



Software GX Works2 umožňuje použití tří různých typů komentářů.

Další informace najdete v příručce pro jednoduché projekty softwaru GX Works2.

Typ komentáře	Obsah komentáře
Device comment	Do komentáře proměnné lze zadat až 32 znaků, které se zobrazí pod danou proměnnou (I/O nebo jiná adresa paměti).
Statement	Popisek může obsahovat až 64 znaků, které budou přidány nad vybraný blok žebříkového diagramu (nad číslo kroku). Každý blok žebříkového diagramu může mít několik popiseků.
Note	Poznámku o maximálně 32 znacích lze zobrazit nad vybranou cívku či aplikační instrukcí.

Následující stránka simuluje postup přidávání komentářů proměnné do ukázkového programu.

## 3.3

## Tvorba snadno pochopitelných programů

MELSOFT Series GX Works2 (Unset Project) - [[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

Unlabeled L02 Host Station 5/75Step

```

graph TD
    subgraph Rung_0 [0]
        X6[Start sw  
X6  
řich] --- AND(( ))
        X7[Door open  
sensor  
X7] --- AND
        AND --- Y0[Robot start  
signal  
Y0]
    end
    subgraph Rung_4 [4]
        Y0_4[Robot start  
signal  
Y0] --- Y1[Operation  
lamp  
Y1]
    end
    subgraph Rung_6 [6]
        END[END]
    end
  
```

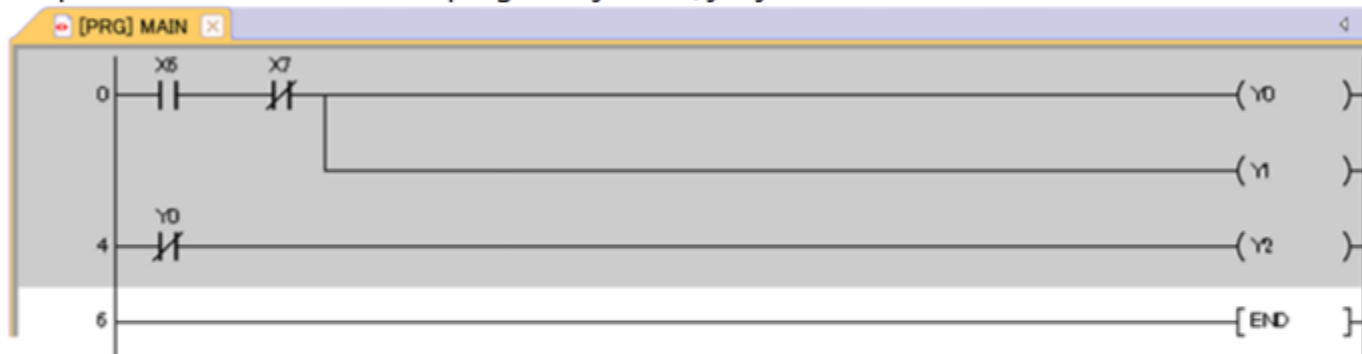
Zadávání komentářů proměnných bude dokončeno.  
Pokračujte kliknutím na tlačítko .

## 3.4

## Konverze programů do spustitelné formy

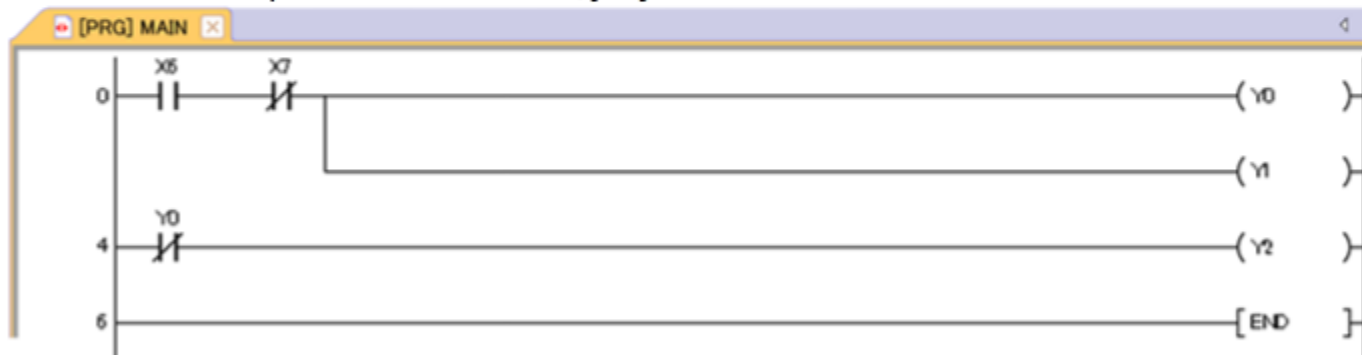
Po dokončení programu je třeba ho převést do formátu, který je možné spustit v modulu CPU. Nezkonvertované programy není možné spouštět ani ukládat.

Barva pozadí nezkonvertovaného programu je šedá, jak je znázorněno níže.



Konverze

Po konverzi se barva pozadí změní na bílou, jak je znázorněno níže.



Na další stránce zkuste program převést v simulovaném okně.



## 3.4

## Konverze programů do spustitelné formy



MELSOFT Series GX Works2 (Unset Project) - [[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

Unlabeled

L02

Host Station

5/75Step

NL


0 X6 X7 Y0 Y1 Y2

Start switch Door open sensor Robot start signal Operation lamp

4 Y0 Robot start signal

6 [END]

Jakmile bude program zkonvertován, barva pozadí se změní z šedé na bílou.

Program je nyní zkonvertován.  
Pokračujte kliknutím na tlačítko .

Po dokončení konverze programů uložte projekt spolu s programy. Dojde-li k ukončení softwaru GX Works2 bez uložení projektu, související programy budou ztraceny. Je tedy důležité projekt často ukládat.

Při ukládání nového projektu určete následující typy informací o projektu. (Při opakovaném ukládání to již není nutné.) Je vhodné zahrnout informace, které jiným osobám usnadní pochopit obsah řízení programu, název systému atd.

Položka	Povinné	Popis
Save destination path	✓	Určete složku, ke které má být přiřazen pracovní prostor.
Workspace/project list		Pokud již ve složce určené polem „Save destination path“ (Cesta k cíli ukládání) existuje jeden nebo více pracovních prostorů, budou uvedeny.
Workspace name	✓	Zadejte název pracovního prostoru o maximálně 128 znacích.
Project name	✓	Zadejte název projektu o maximálně 128 znacích.
Title		Zadejte nadpis projektu o maximálně 128 znacích. Tento parametr je užitečný, pokud chcete zadat dlouhý název, který se nevejde do pole „Project name“ (Název projektu).

**Pracovní prostor** je složka ke správě více projektů.

Příklad používání pracovního prostoru je uveden níže. (Projekty jsou spravovány pro každý typ vozidla automobilové výrobní linky.)

Název pracovního prostoru	Název projektu	Nadpis
Automobilová výrobní linka	Výrobní linka typu A	Normální provozní program k řízení výrobní linky typu A
	Výrobní linka typu B	Normální provozní program k řízení výrobní linky typu B
	Výrobní linka typu C	Normální provozní program k řízení výrobní linky typu C

### Poznámky:

- Pokud uložíte projekt obsahující nezkonvertovaný program, bude ztracen pouze tento nezkonvertovaný program. Před uložením projektu vždy zkonvertujte programy, jak jste se naučili v oddílu 3.4.
  - Zadejte cestu k cíli ukládání, název pracovního prostoru a název projektu tak, aby celkový počet znaků nepřesáhl 150.
- Na další stránce zkuste projekt uložit v simulovaném okně.

# 3.5 Ukládání projektů



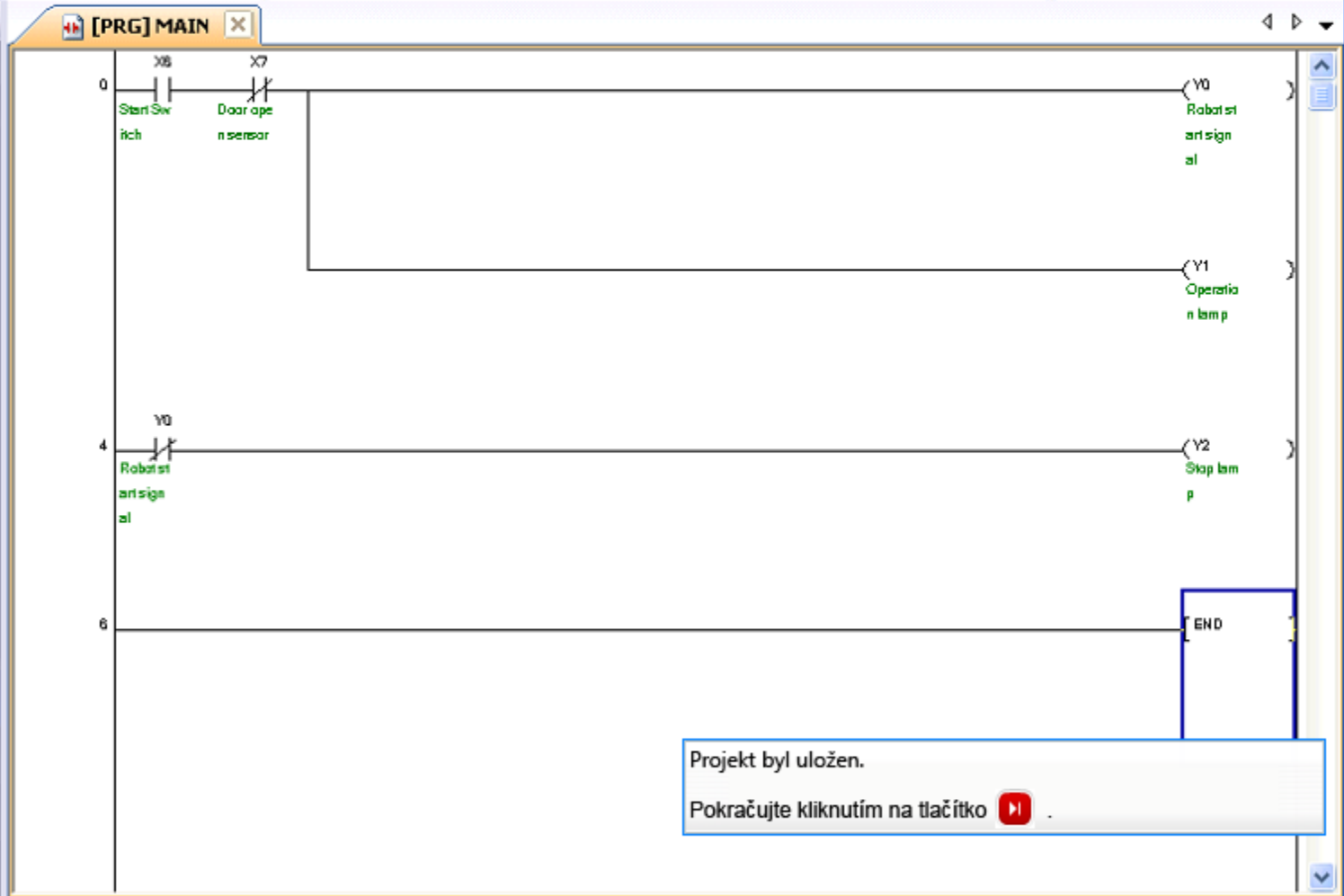
MELSOFT Series GX Works2 C:\SequenceProgram\Learning\Robot\_Control - [[PRG] MAIN]

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

Navigation [PRG] MAIN

Project

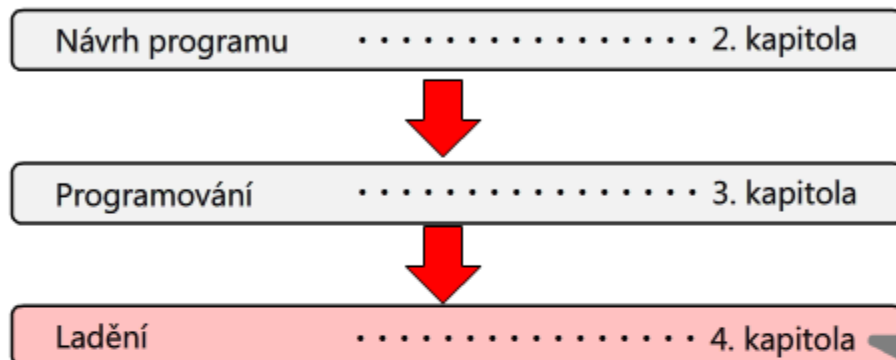
- Parameter
- Intelligent Function Module
- Global Device Comment
- Program Setting
- POU
  - Program
    - MAIN
  - Local Device Comment
- Device Memory
- Device Initial Value



Projekt byl uložen.  
Pokračujte kliknutím na tlačítko .

## 4. kapitola Ladění

Ve 4. kapitole se naučíte zapisovat sekvenční programy do modulu CPU a ladit je.



### Postup výuky ve 4. kapitole

#### 4.1 Ladění

- 4.1.1 Ladění programu bez použití modulu CPU
- 4.1.2 Změna stavu proměnné I/O
- 4.1.3 Monitorování stavu proměnné
- 4.2 Zápis programů do modulu CPU
- 4.3 Aktivace zapsaných programů
- 4.4 Spouštění programů
- 4.5 Ladění programů
- 4.6 Kontrola provozu systému PLC
- 4.7 Provozování systému PLC

Po zapsání programu nebo části programu je nutné kód otestovat, aby byla ověřena jeho funkce dle očekávání.

Nedostatky v softwaru (kdy napsaný kód nefunguje tak, jak byl zamýšlen) se nazývají „**programátorské chyby**“ a postupům hledání příčiny nesprávného chování a jeho nápravy se říká „**ladění**“.

Testování a ladění jsou klíčovými kroky při vytváření programů.

To platí u programovatelných kontrolérů dvojnásob, protože přítomnost chyb může způsobit selhání systému, poškození zařízení a další nehody.

Následující tabulka uvádí seznam některých funkcí softwaru GX Works2, které vám při procesu ladění pomohou.

Název funkce	Popis
Simulator	Tato funkce slouží k simulaci spuštění programu i bez modulu CPU. Lze ji použít k ladění v prostředí, ve kterém modul CPU není k dispozici.
Monitor	Tato funkce umožňuje monitorování stavu výkonu programu a stavu každé proměnné během výkonu programu v modulu CPU. K dispozici je více funkcí monitoru v závislosti na aplikaci. Například monitorování žebříkového diagramu, monitorování pouze registrovaných proměnných a monitorování všech proměnných v dávce.
Change current value	Tato funkce umožňuje nuceně změnit stav proměnné (bit; ON ↔ OFF (ZAP./VYP.)), slovo: aktuální hodnota) během výkonu programu v modulu CPU. Tato funkce je užitečná ke změně aktuální hodnoty proměnné typu slovo nebo stavu interního relé.
Forced input output registration/cancellation	Tato funkce umožňuje nuceně změnit stav (ON ↔ OFF (ZAP./VYP.)) registrované proměnné I/O během výkonu programu v modulu CPU. Pro účely ladění či ověřování provozu pomocí samotného modulu CPU lze tuto funkci použít jako náhradu za vypínač.

Tyto funkce jsou s ohledem na proces ladění podrobněji vysvětleny dále v této kapitole.

### Poznámky k ladění

Neprovádějte úkony ladění, když je programovatelný kontrolér připojen k fyzickým zařízením I/O.

Chyby v programu, vynucení proměnných I/O či změny slovních hodnot by mohly způsobit poškození externího zařízení nebo i hůře.

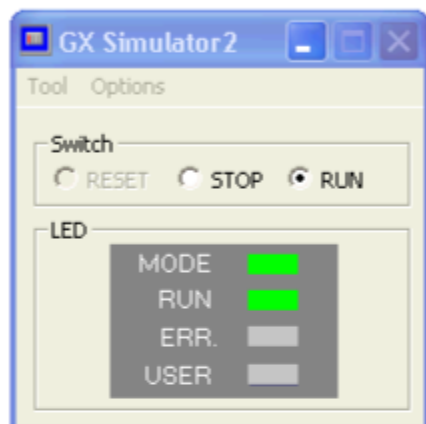
Pokud nemáte k dispozici odpojený systém PLC, použijte funkci simulátoru.

## 4.1.1


## Ladění programu bez použití modulu CPU

Pokud pro ladění není k dispozici modul CPU, použijte **funkci simulátoru**.

Program je možné spustit ve virtuálním modulu CPU, který software poskytuje bez použití skutečného modulu CPU.



 Svítí

 Nesvítí

Položka	Stav	Popis
Switch (Vypínač)	RUN (BĚH)	Spouští virtuální modul CPU.
	STOP (ZASTAVENÍ)	Zastavuje virtuální modul CPU.
	RESET (RESET)	Resetuje virtuální modul CPU. (K dispozici pouze ve stavu STOP (ZASTAVENÍ))
LED (LED)	MODE (REŽIM)	Indikuje stav MODE (REŽIM) virtuálního CPU.
	RUN (BĚH)	Indikuje stav provozu virtuálního CPU. • Svítí: Stav RUN (BĚH) • Nesvítí: Stav STOP (ZASTAVENÍ)
	ERR (CHYBA)	Indikuje chybový stav provozu virtuálního modulu CPU. Pokud bude zjištěna chyba, kontrolka se rozsvítí nebo rozbliká.
	USER (UŽIVATEL)	Informuje, zda došlo ve virtuálním modulu CPU k uživatelské chybě. Rozsvítí se nebo rozbliká, když nastanou problémy.

### Poznámky k používání funkce simulátoru

- Ladění za použití funkce simulátoru nezaručuje, že bude sekvenční program po ladění správně fungovat.
- Funkce simulátoru vykonává vstup/výstup dat pomocí modulů I/O za použití simulační paměti. Funkce nepodporuje některé instrukce, funkce a paměť proměnných. Z toho důvodu se mohou výsledky provozu s funkcí simulátoru lišit od těch se skutečným modulem CPU.

Na další stránce zkuste funkci simulátoru použít v simulovaném okně.

# 4.1.1 Ladění programu bez použití modulu CPU

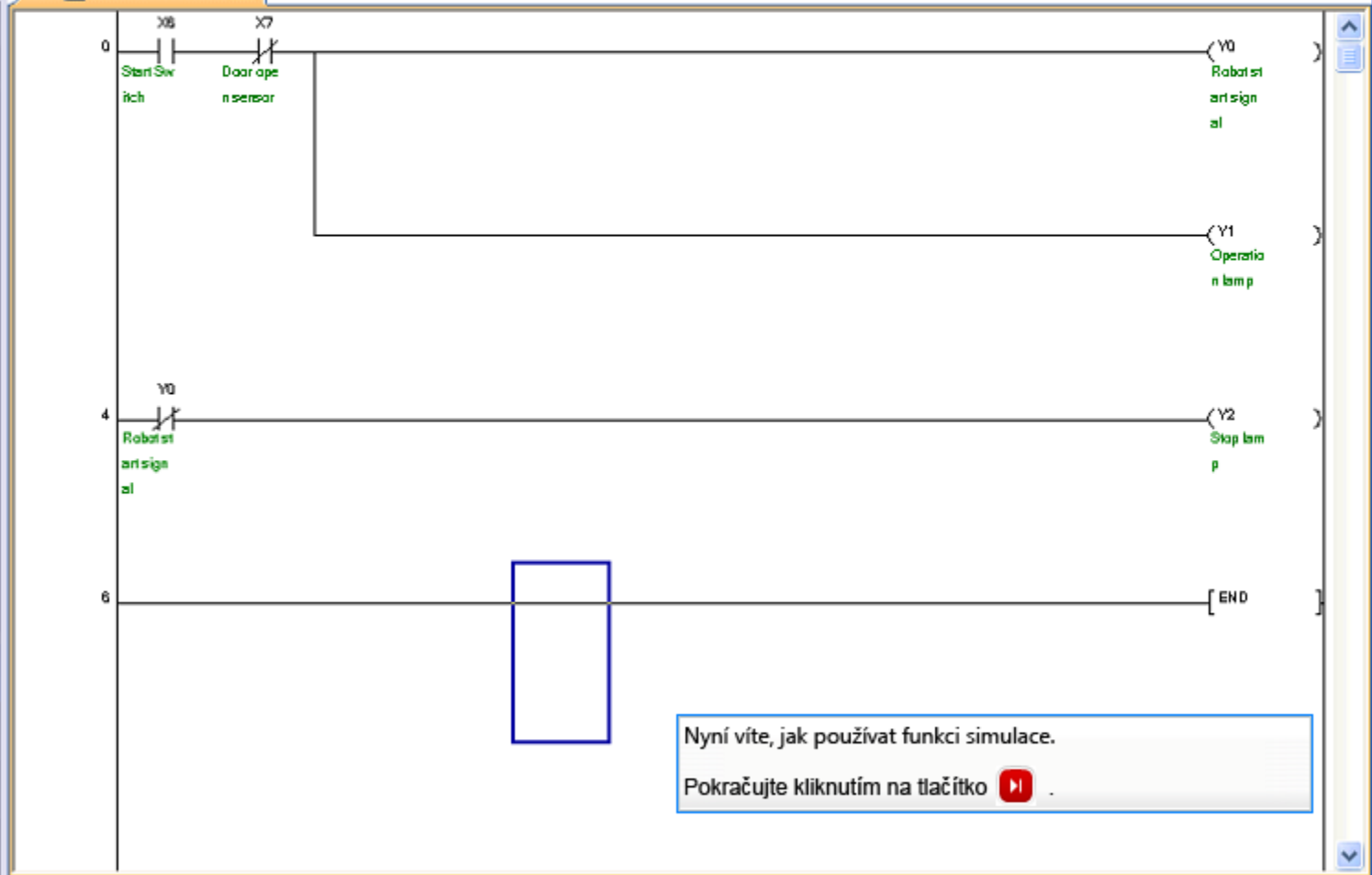
MELSOFT Series GX Works2 C:\SequenceProgram\Learning\Robot\_Control - [[PRG] MAIN]

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



Nyní víte, jak používat funkci simulace.  
 Pokračujte kliknutím na tlačítko .

## 4.1.2 Změna stavu proměnné I/O

Při ladění sekvenčního programu pomocí modulu CPU, ke kterému nejsou připojena žádná zařízení I/O, nebo pomocí simulator function (funkce simulátoru) použijte funkci **Forced Input Output Registration/Cancellation** (Vynucená registrace/zrušení vstupu a výstupu) ke změně stavu ON/OFF (ZAP./VYP.) proměnné I/O.

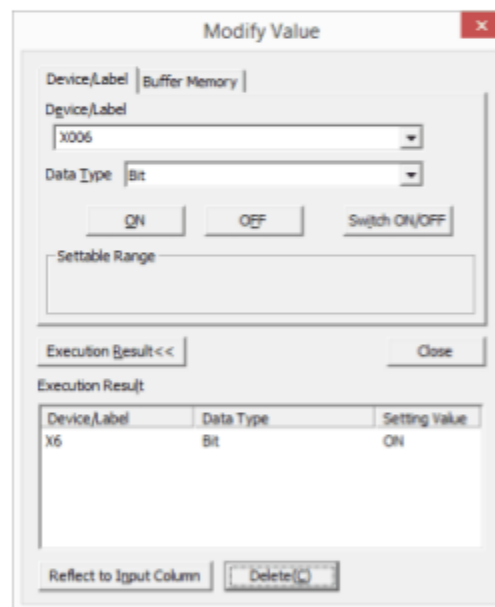
Stav zaregistrovaných proměnných I/O lze pomocí softwaru nuceně změnit na ON (ZAP.) nebo OFF (VYP.).

(Řady MELSEC-Q a MELSEC-L): z obrazovky „Forced Input Output Registration/Cancellation“ (Vynucená registrace/zrušení vstupu a výstupu)

(Řada MELSEC-F): z obrazovky „Modify Value“ (Upravit hodnotu)



Obrazovka Forced Input Output Registration/Cancellation (Vynucená registrace/zrušení vstupu a výstupu) (řady MELSEC-Q a MELSEC-L)



Obrazovka Modify Value (Upravit hodnotu) (řada MELSEC-F)

### Změny stavů jiných proměnných

Chcete-li změnit aktuální hodnotu proměnné typu slovo nebo stav ON/OFF (ZAP./VYP.) interního relé, použijte funkci **změny aktuální hodnoty**.

Podrobnosti najdete v příručce.



## 4.1.3 Monitorování stavu proměnné

Po spuštění simulace automaticky započne monitorování. Pokud chcete použít monitorovací režim při připojení ke skutečnému CPU programovatelného kontroléru, jednoduše klikněte na položky Online (Online), Monitor (Monitor) a poté Start Monitoring (Zahájit monitorování). Také můžete použít klávesovou zkratku F3.

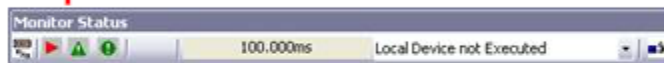
V režimu monitorování budou hodnoty a stavy všech proměnných používaných v programu zobrazeny ve vrstvě nad kódem programu. To umožňuje uživateli sledovat změny hodnot včetně efektů používání funkce „Forced input output registration/cancellation“ (Vynucená registrace/zrušení vstupu a výstupu).

Navíc se zobrazí panel **Monitor Status** (Monitorování stavu) se základními informacemi ohledně stavu modulu CPU nebo virtuálního modulu CPU. Vysvětlení informací panelu **Monitor Status** (Monitorování stavu) najdete v níže uvedené tabulce.

### Při připojení k modulu CPU



### Při používání funkce simulátoru



Stav	Ikona/indikátor	Popis
Stav připojení	Při připojení k modulu CPU	Zobrazuje stav připojení k modulu CPU nebo funkci simulátoru.
	Při používání funkce simulátoru	
Stav RUN/STOP (BĚH/ZASTAVENÍ)	RUN	Zobrazuje stav provozu modulu CPU (RUN nebo STOP).
	STOP	
Stav ERR. (CHYBA)	ERR. nesvítí	Indikuje chybový stav provozu modulu CPU.
	ERR. svítí	
	ERR bliká	
Stav USER (UŽIVATEL)	USER nesvítí	Indikuje chybový stav uživatele při provozu modulu CPU.
	USER svítí	
	USER bliká	
Doba scanu		Zobrazuje maximální dobu scanu monitorovaného modulu CPU.
Stav přítomnosti/absence nepodporovaných instrukcí	Existuje nepodporovaná instrukce.	Při spuštění funkce simulátoru uvádí, zda existují nepodporované instrukce.
	Neexistuje žádná nepodporovaná instrukce.	Kliknutím na ikonu otevřete okno Unsupported Instruction/Device (Nepodporované instrukce/proměnné).

## 4.1.3 Monitorování stavu proměnné

V monitorovacím režimu je viditelný aktuální stav všech proměnných v programu.

### Zobrazení stavu bitové proměnné (ON/OFF (ZAP./VYP.))

Stav ON/OFF (ZAP./VYP.) je během monitorování zobrazen následovně.

Stav OFF  
(VYP.)



Stav ON  
(ZAP.)



\* Tento typ zobrazení platí pouze pro instrukce SET, RST, PLS, PLF, SFT, SFTP, MC a porovnávací instrukce typu kontakt. Berte na vědomí, že pro instrukci RST se zobrazuje pouze stav ON/OFF (ZAP./VYP.).

### Zobrazení aktuální hodnoty proměnné typu slovo (zobrazení decimálního/hexadecimálního čísla)

Aktuální hodnota během monitorování je zobrazena následovně.



Aktuální hodnota T0 (v případě zobrazení decimálního čísla)

### Monitorování pouze určitých proměnných

Při monitorování velmi rozsáhlého nebo složitého programu může být výhodné monitorovat pouze určité proměnné. K tomuto účelu zahrnuje software GX Works2 sledovací okna, která umožňují snadné přidávání požadovaných proměnných, sledování jejich stavu a upravování jejich hodnot během monitorování. Podrobnosti najdete v provozní (obecné) příručce softwaru GX Works2.

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

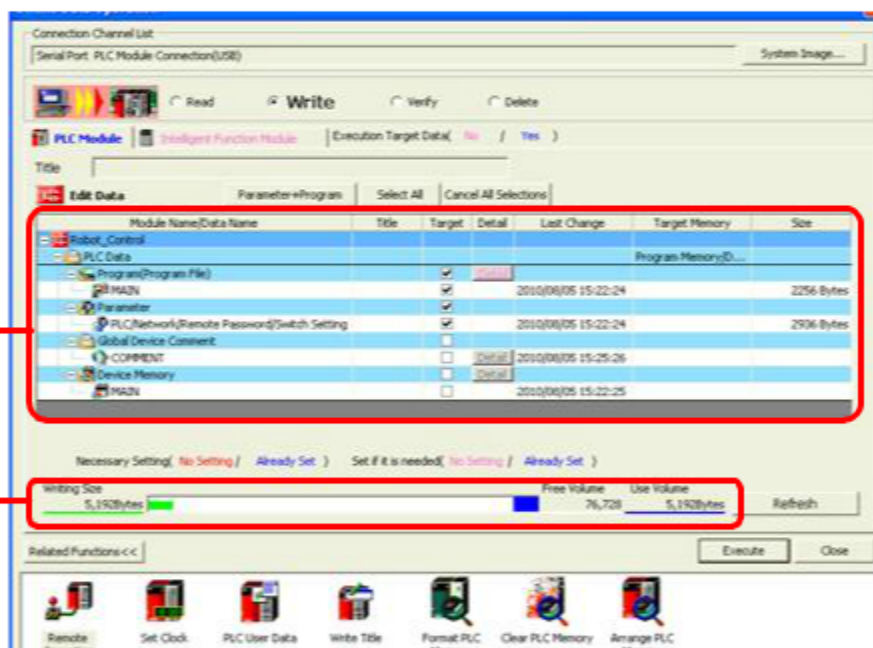
## Zápis programů do modulu CPU

Před zahájením jakéhokoli ladění za použití skutečného modulu CPU přepněte CPU do režimu STOP (ZASTAVENÍ), ověřte navázání připojení s CPU a zapište programy a parametry do programové paměti.

Jak je zřejmé z níže uvedeného obrázku, hlavní funkce okna Write to PLC (Zapsat do PLC) umožňují uživateli vybrat požadované soubory k zápisu, zvolit jejich umístění a potvrdit kapacitu paměti modulu CPU. Tři tlačítka nad seznamem souborů umožňují uživateli rychle vybrat požadované soubory k zápisu. Nejobvyklejší, který je používán v následující simulaci, je „Parameter+Program“ (Parametr + program).

Seznam souborů

Kapacita paměti



Na další stránce zkuste provést zápis do modulu CPU v simulovaném okně.

# 4.2 Zázpis programů do modulu CPU

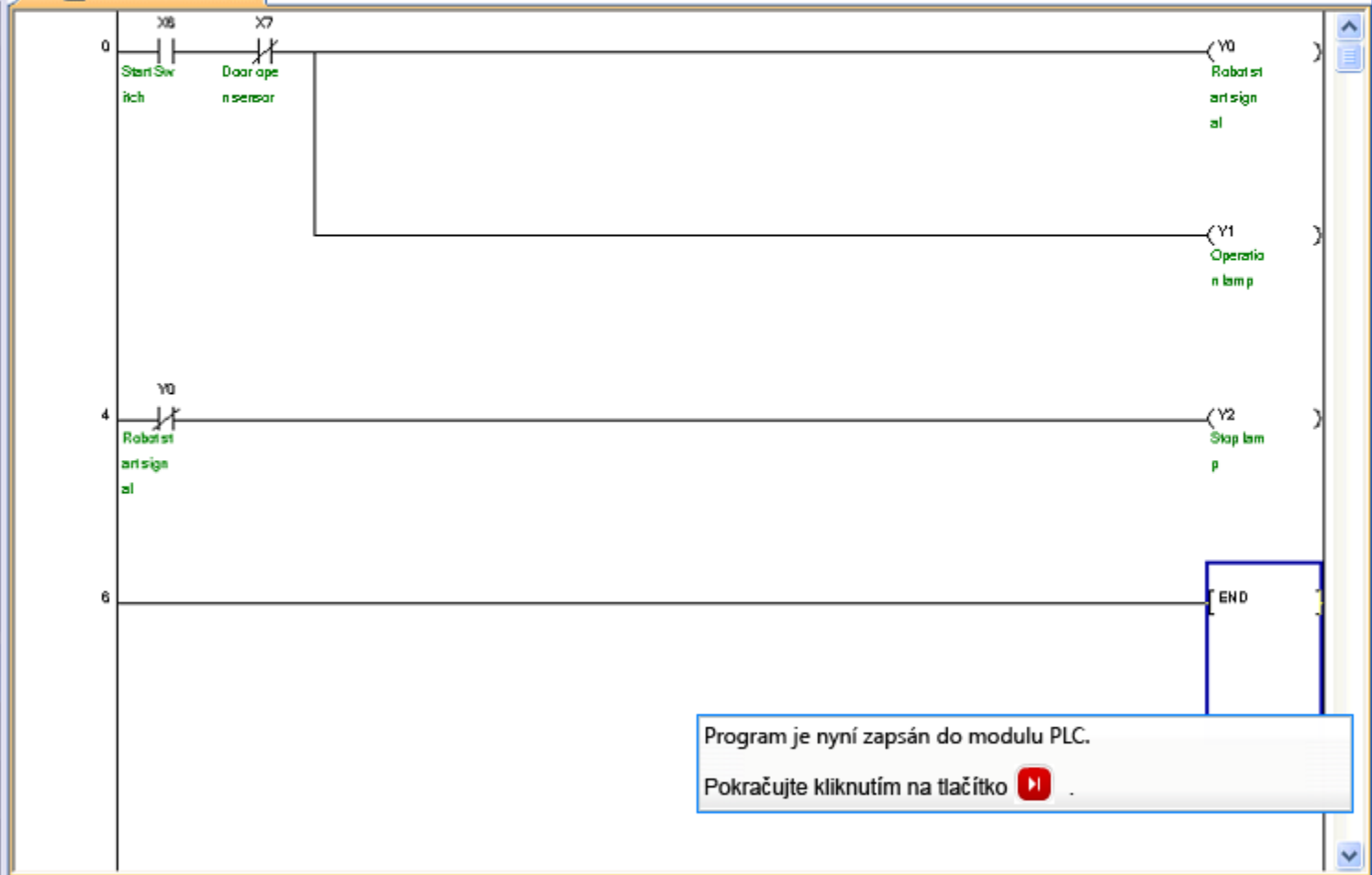
MELSOFT Series GX Works2 C:\SequenceProgram\Learning\Robot\_Control - [[PRG] MAIN]

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

Program je nyní zapsán do modulu PLC.  
 Pokračujte kliknutím na tlačítko  .

## 4.3

## Aktivace zapsaných programů

(Řada MELSEC-F): Následující operace není nutná.

(Řady MELSEC-Q a MELSEC-L): Následující operace je nutná.

Po zápisu programu do modulu CPU **resetujte** modul CPU.

Zapsané programy nebudou aktivní, dokud modul CPU neresetujete.

\* Tato operace není vyžadována, pokud k ladění používáte simulator function (funkci simulátoru).

Resetujte modul CPU následovně:

(1) Přepněte a podržte přepínač RESET/STOP/RUN (RESET/ZASTAVENÍ/BĚH) na předním panelu modulu CPU v poloze RESET (po dobu alespoň 1 sekundy).

[Probíhá reset]

L02CPU		MODE (REŽIM): Svítí zeleně
MODE ■	ERR.	RUN (BĚH) : Nesvítí
RUN	I/OERR.	ERR. (CHYBA) : Bliká
BAT.	USER	

Zatlačte na dobu alespoň 1 sekundy.



(2) Uvolněte přepínač poté, co svítící kontrolka MODE (REŽIM) a blikající kontrolka ERR. (CHYBA) zhasnou.

[Reset byl dokončen]

L02CPU		MODE (REŽIM): Svítí zeleně
MODE ■	ERR.	RUN (BĚH) : Nesvítí
RUN	I/OERR.	ERR. (CHYBA) : Nesvítí
BAT.	USER	

Přepínač RESET/  
STOP/RUN  
(RESET/ZASTAVENÍ/BĚH)

(3) Přepínač se pro dokončení resetu vrátí do polohy STOP (ZASTAVENÍ).

### Řady MELSEC-Q a MELSEC-L

Po dokončení resetu spusťte program.

Následujícím způsobem přepněte modul CPU do **stavu RUN (BĚH)**, aby spustil program.

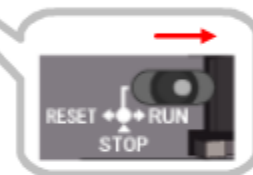
\* Tato operace není vyžadována, pokud k ladění používáte simulator function (funkci simulátoru).



- (1) Přepněte přepínač RESET/STOP/RUN (RESET/ZASTAVENÍ/BĚH) na předním panelu modulu CPU do polohy RUN (BĚH).

Kontrolky ve stavu STOP (ZASTAVENÍ)

L02CPU	MODE (REŽIM): Svítí zeleně
MODE	ERR. I/OERR. USER
RUN	RUN (BĚH) : Nesvítí
BAT.	



- (2) Pokud kontrolka RUN (BĚH) svítí zeleně, program funguje normálně.

Kontrolky ve stavu RUN (BĚH)

L02CPU	MODE (REŽIM): Svítí zeleně
MODE	ERR. I/OERR. USER
RUN	RUN (BĚH) : Svítí zeleně
BAT.	

### Řada MELSEC-F

Po zápisu programu do hlavní jednotky následujícím způsobem přepněte hlavní jednotku do stavu RUN (SPUSTIT) kvůli spuštění programu. (Operace resetování není nutná.)

- (1) Přepněte přepínač RUN/STOP (BĚH/ZASTAVENÍ) na předním panelu hlavní jednotky do polohy RUN (BĚH).



Kontrolky ve stavu STOP (ZASTAVENÍ)

- (2) Pokud se rozsvítí kontrolka RUN (BĚH), program funguje normálně.



Kontrolky ve stavu RUN (BĚH)

## 4.5

## Ladění programů

Po spuštění modulu CPU použijte funkci Forced input output registration/cancellation (Vynucená registrace/zrušení vstupu a výstupu) ke změně stavu jednotlivých proměnných a monitorování výsledků (výstupu) v žebříkovém diagramu.

(Příklad obrazovky pro řady MELSEC-Q a MELSEC-L)

The screenshot shows the MELSOFT GX Works2 interface. A dialog box titled "Forced Input Output Registration/Cancellation" is open, displaying a table for configuring forced I/O states. The table has columns for No., Device, and ON/OFF. The first two rows are pre-filled: No. 1 with Device X05 and ON/OFF ON; No. 2 with Device X07 and ON/OFF OFF. The rest of the table is empty. Buttons for "Register FORCE ON", "Register FORCE OFF", "Cancel Registration", "Update Status", "Batch Cancel Registration", and "Close" are visible. The background shows a ladder logic diagram with a normally open contact labeled X05 and a coil labeled Y05.

No.	Device	ON/OFF	No.	Device	ON/OFF
1	X05	ON	17		
2	X07	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		

Na další stránce si vyzkoušejte ladění programu v simulovaném okně.

## 4.5

## Ladění programů



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

Unlabeled L02 Host Station 6/75Step

Diagram description: The ladder logic shows a sequence of steps. Step 0 has two normally open contacts: 'Start Switch' (X6) and 'Door open sensor' (X7). This step outputs to two coils: 'Robot start signal' (Y0) and 'Operation lamp' (Y1). Step 4 has one normally open contact: 'Robot start signal' (Y0), which outputs to a coil: 'Stop lamp' (Y2). Step 6 contains an 'END' instruction.

Ladění programu bylo dokončeno.  
Pokračujte kliknutím na tlačítko .



## 4.6


## Kontrola provozu systému PLC

Po dokončení ladění programu jej zapište do reálného systému PLC za účelem finální kontroly provozu.

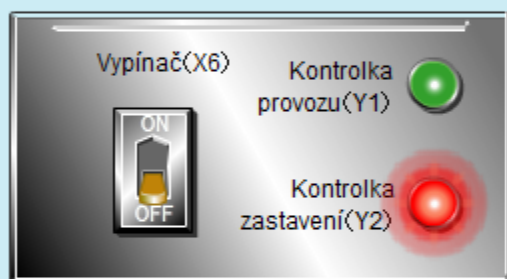
Ovládejte reálná zařízení I/O, abyste potvrdili funkčnost dle návrhu.

I při ovládání zařízení I/O je možné stav jednotlivých proměnných kontrolovat pomocí funkce monitoru softwaru GX Works2.

## Provoz ukázkového systému

 Klikněte do červeného kroužku

Ovládací panel robota



Robot v bezpečnostní kleci

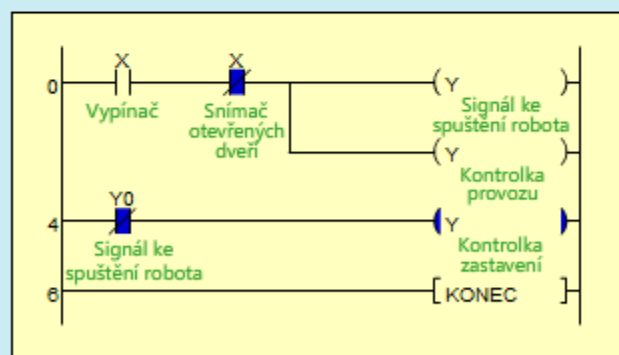


Když přepnete **vypínač (X6)** do polohy OFF (VYP), **signál spuštění robota (Y0)** se deaktivuje a zastaví činnost robota. Současně zhasne **kontrolka provozu (Y1)** na ovládacím panelu a rozsvítí se **kontrolka zastavení (Y2)**.

Znovu



Předchozí



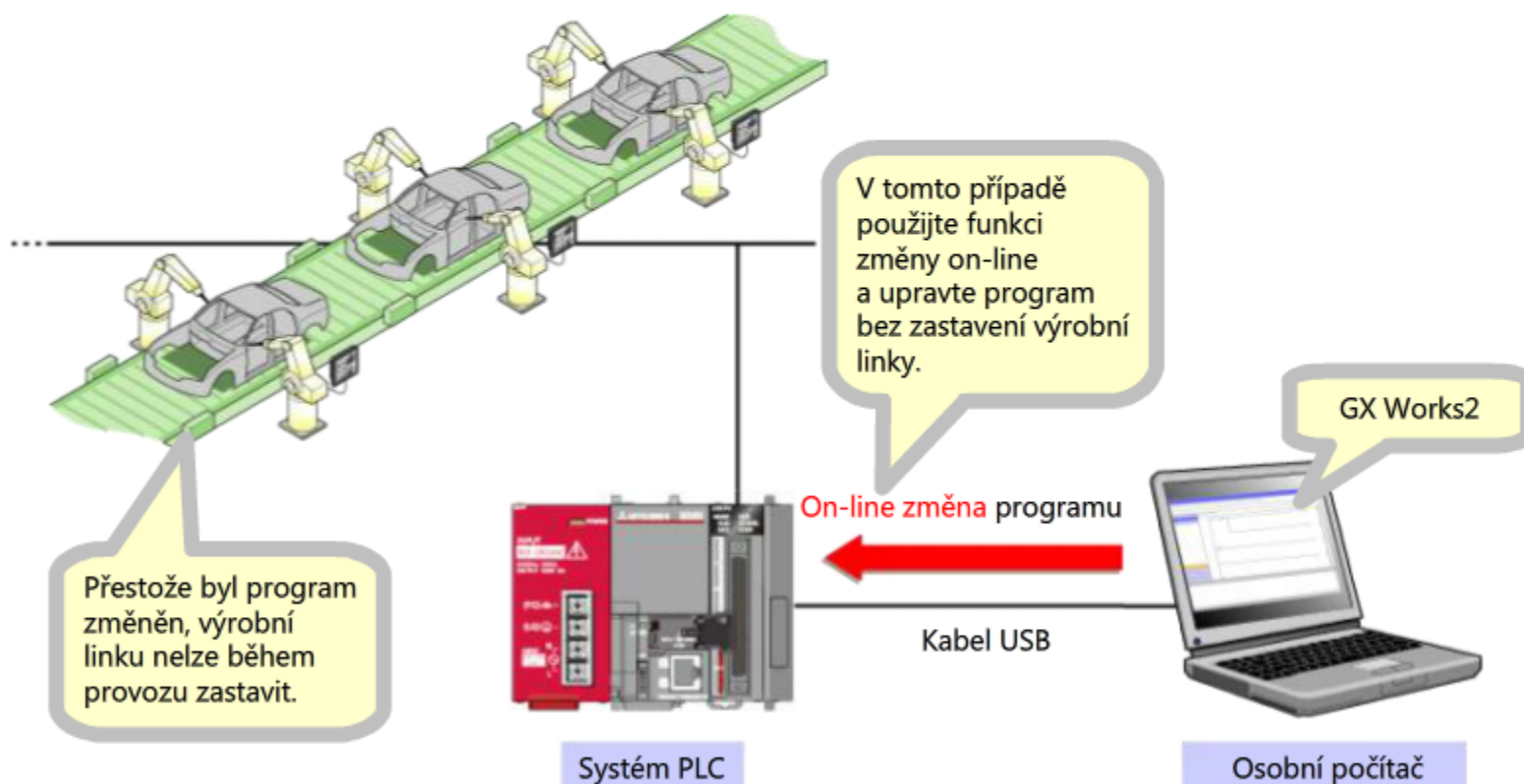
## 4.7 Provozování systému PLC

Po dokončení ověřování provozu spusťte systém PLC a zahajte provoz.

### **Pokud je nutné program upravit v běžícím systému**

Po zahájení provozu systému může být nezbytné provádět úpravy programu, například opravy chyb nebo rozšiřování systému. Normálně je třeba systém (modul CPU) zastavit, aby bylo možné zapsat upravený program, to však není vždy možné. K odstranění tohoto problému používá software GX Works2 funkci změn on-line, která slouží k zápisu programů bez zastavení běžícího modulu CPU.

### **Příklad: nepřetržitě provozovaná automobilová výrobní linka**



Na další stránce zkuste funkci změny on-line použít v simulovaném okně.

## 4.7

## Provozování systému PLC



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

Unlabeled L02 Host Station 2/75Step

Diagram description: The ladder logic shows three rungs. Rung 0: A normally open contact labeled 'Start Switch' (X0) is connected to a coil for output Y0, labeled 'Robot start signal'. Rung 4: A normally open contact labeled 'Robot start signal' (Y0) is connected to a coil for output Y2, labeled 'Stop lamp'. Rung 6: The program ends with an 'END' instruction.

Změna programu on-line byla dokončena.  
Pokračujte kliknutím na tlačítko .

**4.8****Závěr**

Tímto končí základní vysvětlení návrhu softwaru pro programovatelný kontrolér.

V tomto kurzu jste se naučili:

- nezbytné prvky k programování systému PLC;
- některé základní směrnice navrhování programů, včetně používání komentářů;
- jak používat software GX Works2 k provádění základních úkonů počítačového programování;
- několik technik ladění programů pro PLC kontroléry.

Nyní jste dokončili všechny lekce kurzu **PLC Základy softwaru GX Works2** a můžete podstoupit závěrečný test. Pokud si nejste jisti ohledně nějakého tématu, máte nyní možnost si jednotlivá témata zopakovat. **Tento závěrečný test obsahuje celkem 5 otázek (15 položek).** Závěrečný test můžete podstoupit kolikrát chcete.

### Způsob provedení testu

Po vybrání odpovědi nezapomeňte kliknout na tlačítko **Odpověď**. Pokud nekliknete na tlačítko Odpověď, bude vaše odpověď ztracena. (Otázka bude tedy považována za nezodpovězenou.)

### Hodnocení výsledků

Na stránce hodnocení se zobrazí počet správných odpovědí, počet otázek, procento správných odpovědí a výsledek úspěšný/neúspěšný.

Počet správných odpovědí: 4

Celkový počet odpovědí: 4

Procento: 100%

Pokračovat

Zkontrolovat

Abyste úspěšně složili tento test, musíte správně odpovědět na **60 %** otázek.

- Test můžete ukončit kliknutím na tlačítko **Pokračovat**.
- Test si můžete zkontrolovat kliknutím na tlačítko **Zkontrolovat**. (Kontrola správnosti odpovědí)
- Test si můžete zopakovat kliknutím na tlačítko **Znovu**.

Program, který jste měli na starosti, byl převzat jinou osobou, která měla potíže porozumět položkám řízení programu. Jakým vhodným opatřením lze takovému problému předejít?

- Použitím správného názvu a popisu programu pomocí funkce komentářů softwaru GX Works2.
- Ústním vysvětlením položek řízení nové osobě.
- Vyhnout se předávání složitých a rozsáhlých programů.
- Předávat spolu s programem korespondenční tabulku pro proměnné I/O a čísla proměnných.

Odpověď

Zpět

Dokončete správný postup programování.

Krok 1 Návrh programů

Krok 2 ( Q1  )

Krok 3 ( Q2  )

Krok 4 Konverze programů

Krok 5 Ukládání projektů

Krok 6 ( Q3  )

Krok 7 ( Q4  )

Krok 8 Running the CPU module (RUN)

Krok 9 ( Q5  )

Krok 10 Kontrola provozu systému PLC

Odpověď

Zpět

Vyplněním vynechaných míst dokončete vysvětlení toho, co je třeba udělat po dokončení programu.

Po zapsání programu je nutné ho otestovat, aby byla ověřena jeho funkce dle očekávání.

(  ) (kdy napsaný kód nefunguje tak, jak byl zamýšlen) se nazývá

(  ) a postupům hledání příčiny nesprávného chování a jeho nápravy se říká

(  ).

Tento krok je klíčovým krokem při vytváření programů.



Vyberte vhodné použití jednotlivých funkcí softwaru GX Works2.

Funkce	Aplikace
Simulation	--Select--
Forced input output registration/cancellation	--Select--
Change current value	--Select--
Ladder monitor	--Select--
Watch	--Select--

Odpověď

Zpět

Vyberte správný popis funkce změny on-line.

- Funkce automaticky zastaví modul CPU, zapíše program do CPU a poté CPU automaticky znovu spustí.
- Funkce porovná program spuštěný v modulu CPU s programem otevřeném v softwaru GX Works2.
- Funkce umožňuje zapsat program do modulu CPU po bezpečném zastavení běžícího modulu CPU.
- Funkce umožňuje zapsat program do běžícího modulu CPU bez zastavení.

Odpověď

Zpět

Dokončili jste závěrečný test. Vaše výsledky jsou následující.  
Závěrečný test ukončíte přechodem na další stránku.

Počet správných odpovědí:	5
Celkový počet odpovědí:	5
Procento:	100%

**Gratulujeme. Úspěšně jste prošli v testu.**

Dokončili jste kurz **PLC Základy softwaru GX Works2**.

Děkujeme za vaši účast v tomto kurzu.

Doufáme, že se vám lekce líbily a že informace získané v průběhu tohoto kurzu vám budou užitečné.

Celý kurz si můžete projít kolikrát chcete.

**Zkontrolovat**

**Zavřít**