

Programação eficiente do **CLP**

Este curso descreve o uso da programação estruturada para aumentar a eficiência da programação. O curso é voltado para aqueles que programam usando diagramas ladder simples.

Ao longo do curso, você vai aprender sobre as vantagens da programação estruturada. Esperamos que esse curso ajude você a programar de forma mais eficiente.

O curso a seguir é um pré-requisito antes da realização deste curso:

- Programming Basic (Princípios básicos de programação)

Introdução**Estrutura do curso**

O conteúdo do curso é explicado a seguir.

Capítulo 1 - Por que a programação estruturada é necessária?

Este capítulo descreve a necessidade da programação estruturada para maior eficiência.

Capítulo 2 - Por que programas estruturados são tão eficientes?

Este capítulo descreve os benefícios da programação estruturada.

Teste Final

Pontuação para aprovação: no mínimo 60%

Introdução**Como utilizar esta ferramenta de e-Learning**

Ir para a próxima página		Ir para a próxima página.
Voltar para a página anterior		Voltar para a página anterior.
Mover-se para a página desejada		O "Índice" será exibido, permitindo que você navegue até a página desejada.
Sair do curso		Sair do curso.

Precauções de segurança

Quando você estiver aprendendo a operar os produtos reais, leia cuidadosamente as precauções de segurança dos respectivos manuais.

Precauções neste curso

As telas exibidas da versão de software que você utiliza podem ser diferentes das apresentadas neste curso. Esse curso usa a seguinte versão de software:

- GX Works3 Versão 1.032J

Capítulo 1 Por que a programação estruturada é necessária?

Este capítulo descreve a necessidade da programação estruturada para engenharia mais eficiente.

- 1.1 Estado atual da indústria de FA
- 1.2 A necessidade da eficiência de programação
- 1.3 Programação eficiente por meio de programas estruturados
- 1.4 Programação estruturada com MELSOFT GX Works3

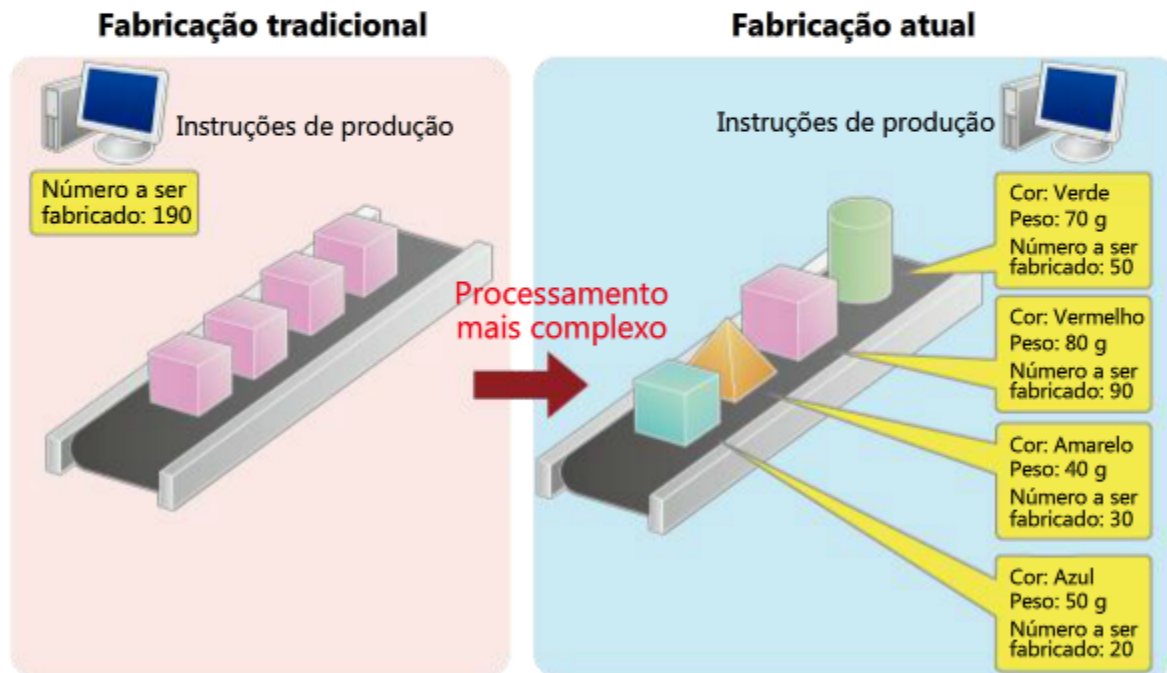
1.1

Estado atual da indústria de FA

Na indústria atual de FA, os dados processados por equipamentos de fabricação estão aumentando a medida que o processamento se torna mais avançado e complexo. Os detalhes são descritos abaixo.

1. Processamento mais complexo

Anteriormente, um único produto era fabricado em uma linha de produção de acordo com uma instrução de produção. No entanto, a fabricação de múltiplos produtos em uma linha de produção está se tornando mais difundida como resultado da diversificação e dinamização do sistema de produção. Por essa razão, o processamento se tornou mais e mais complexo.



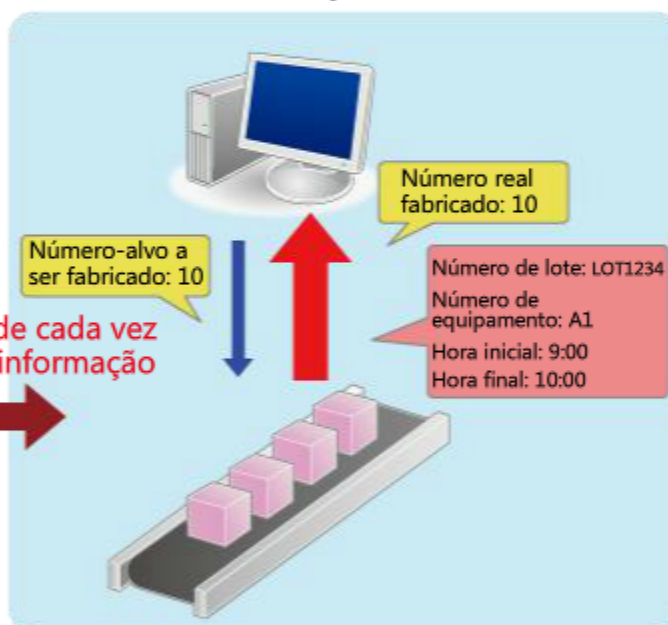
2. Quantidades cada vez maiores de informação

Anteriormente, as informações trocadas entre computadores de nível empresarial e equipamentos do chão de fábrica eram limitadas a instruções de trabalho simples e resultados de produção. Atualmente, registros de produção são incluídos nessas informações de controle de produção a fim de garantir a rastreabilidade. Dessa forma, aumentou a quantidade de informações processada em unidades de fabricação.

Fabricação tradicional



Fabricação atual

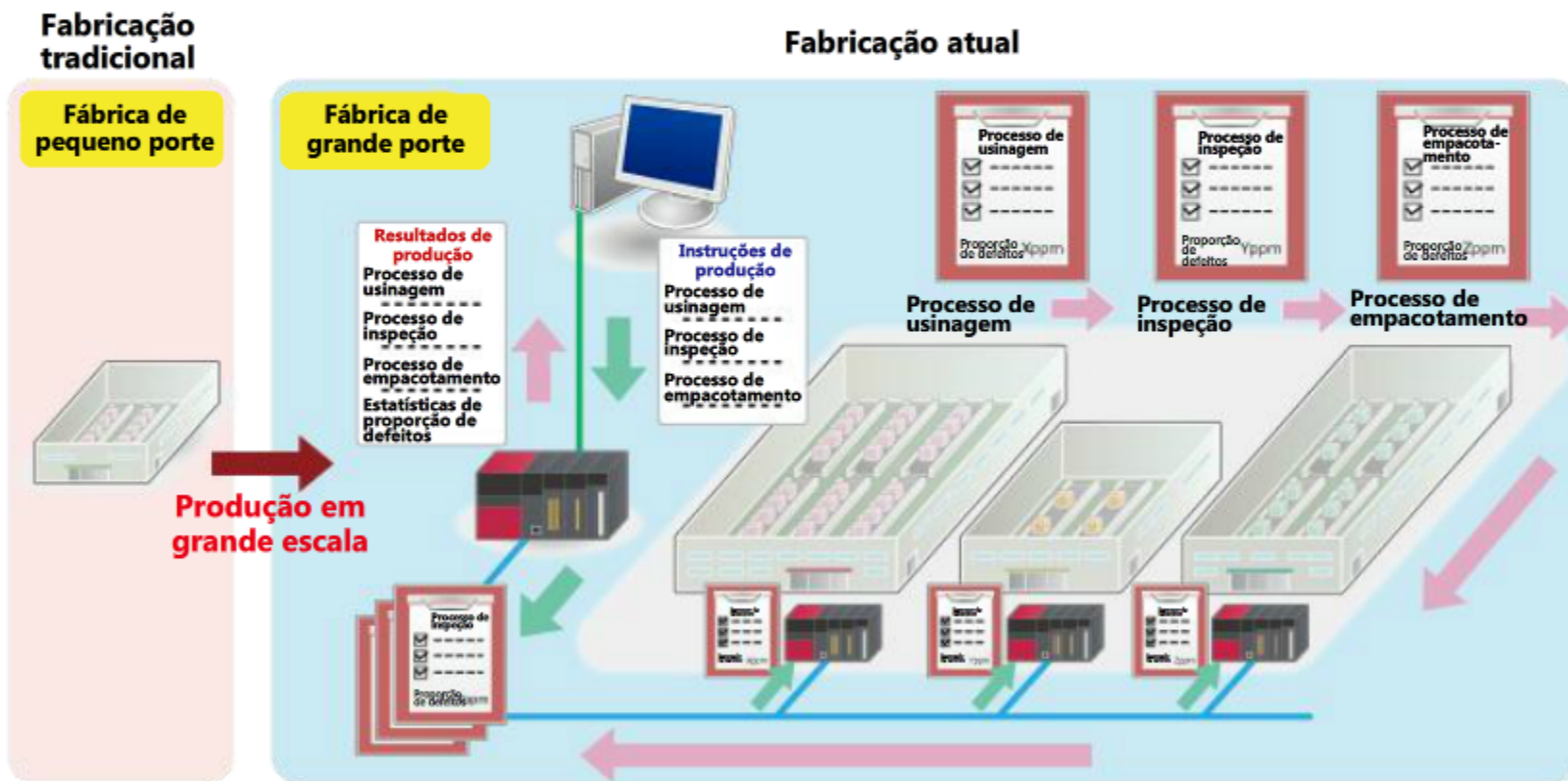


Quantidade cada vez maior de informação

1.1 Estado atual da indústria de FA

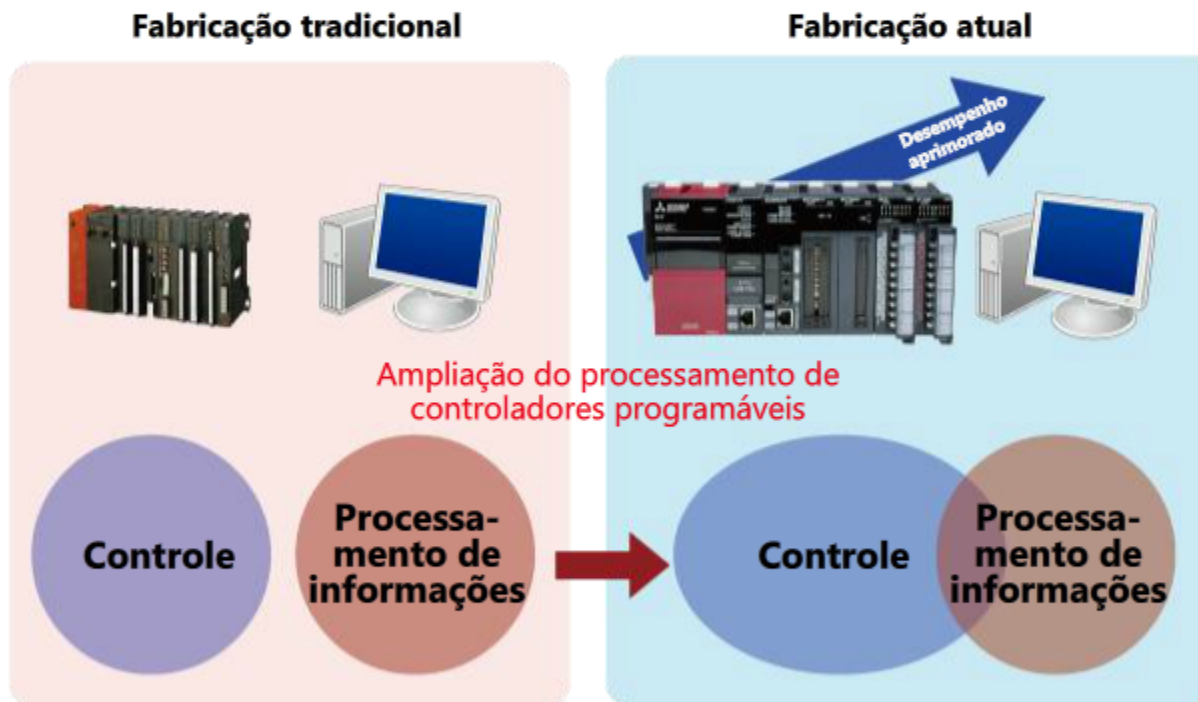
3. Configurações de sistema de grande escala

A construção e o desenvolvimento de fábricas de grande escala se tornaram cada vez mais comuns. Nessas fábricas de grande porte, redes são configuradas para cada processo de produção e as informações de cada processo são coletadas por computadores de nível empresarial. A quantidade de informações processada nesses ambientes é significativamente superior àquela em fábricas convencionais de pequeno porte.



Uma vez que as informações empregadas em ambientes FA aumentaram, os tipos de processamentos realizados por controladores programáveis também sofreram mudanças.

Anteriormente, controladores programáveis eram usados principalmente para controle, enquanto computadores pessoais eram usados para processamento de informações. Atualmente, a capacidade de armazenamento e a velocidade do processamento dos controladores programáveis subiram. Os controladores programáveis foram projetados para serem usados em ambientes FA e, por isso, são mais confiáveis e robustos do que computadores pessoais. Por essa razão, a tendência é começar a contar com eles para o processamento de informações.



Conforme o processamento de controladores programáveis é ampliado, programas de controle são agora usados para processamento mais complexo além de controle sequencial e o número de etapas ter aumentado. Alguns exemplos disso incluem processamento de dados de texto, realização de operações de bancos de dados e realização de controle de rede.

Em programas não convencionais criados usando diagrama ladder simples, todas as funções de processamento são descritas sem distinção clara. Programas complexos com muitas etapas possuem as seguintes desvantagens quando são editados ou quando novos recursos são adicionados posteriormente.

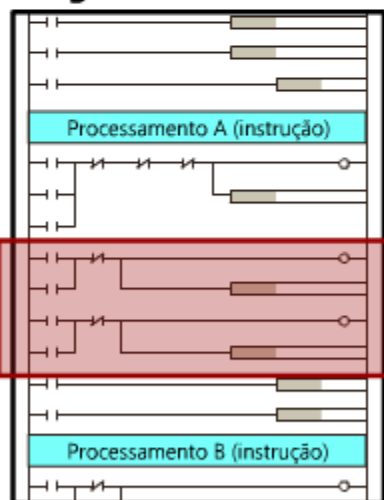
Usar programas existentes não é fácil



Determinar o início ou o fim de um processamento é difícil.

Seções utilizáveis devem ser pesquisadas através dos comentários e instruções.

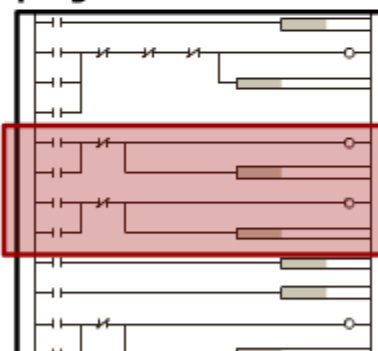
Programa existente




Início do processo

Fim do processo

Novo programa utilizando programas existentes



Clique em  na parte superior direita para iniciar a animação novamente do início.

Nos locais de desenvolvimento de programas, é necessária a programação eficiente tais como a utilização de programas similares e a padronização/compartilhamento de programas frequentes, a fim de reduzir o tempo de engenharia para desenvolver novos programas.

1.3 Programação eficiente por meio de programas estruturados

Os pontos negativos associados com programas ladder simples e convencionais são a difícil compreensão e utilização. Programas estruturados podem melhorar a eficiência da programação e eliminar tais problemas.

A animação a seguir descreve o conceito da divisão de programas em partes de programas.

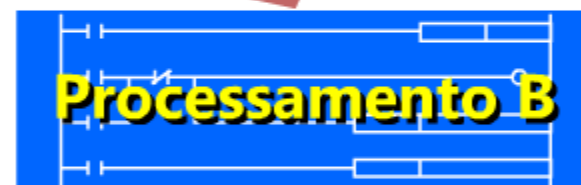
Programas podem ser utilizados selecionando apenas partes individuais.




Programa existente



Novo programa utilizando programas existentes



Só é necessário selecionar a parte do programa

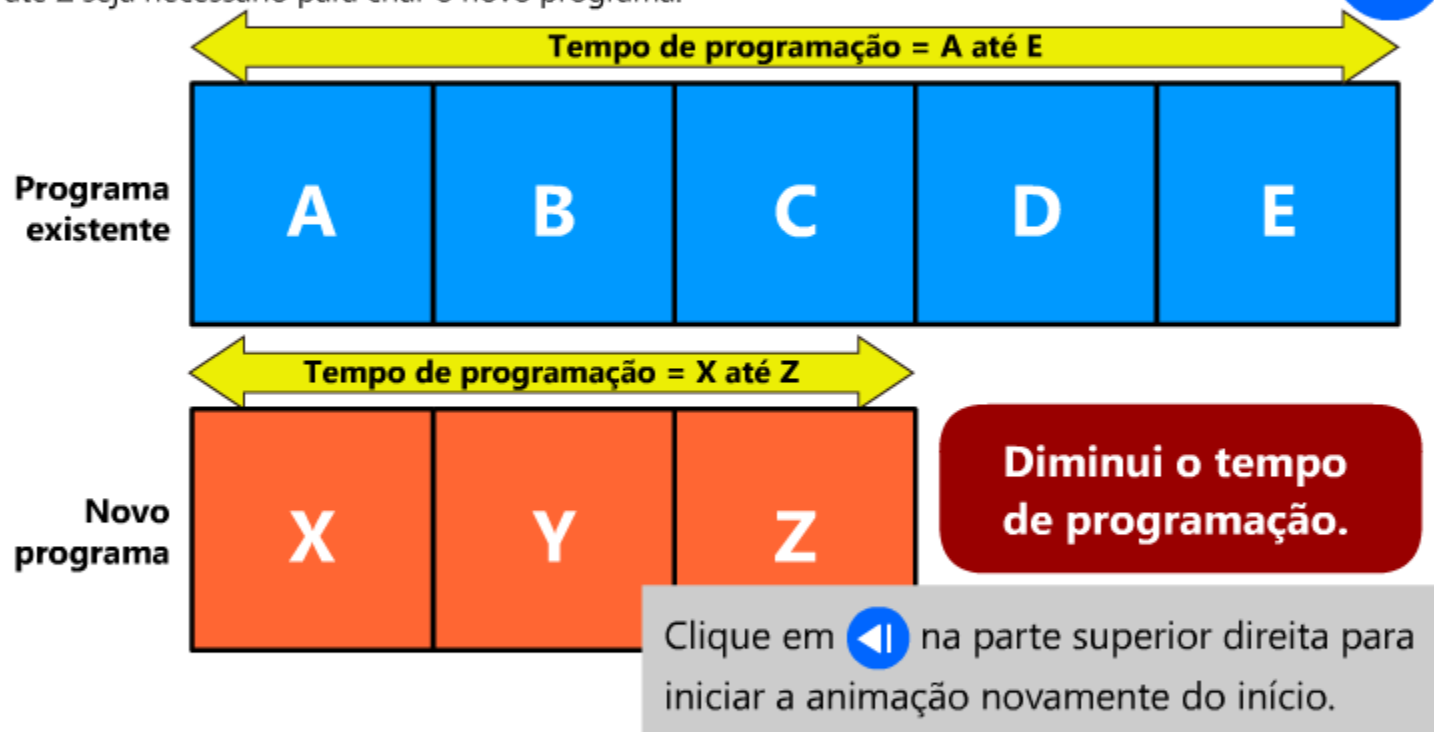
Clique em  na parte superior direita para iniciar a animação novamente do início.

A utilização do programa melhora a eficiência da programação. Os detalhes são descritos nas próximas seções.

1.3.1 Benefícios do uso de programas

A utilização máxima de programas existentes reduz o tempo de programação.

O novo programa inclui processamento representado como B até E no programa existente. Esse processamento pode ser utilizado de forma que somente o tempo de programação de X até Z seja necessário para criar o novo programa.



Dessa forma, o uso de programas existentes vai eliminar a criação de novos programas, reduzindo o tempo geral de programação. Maximizar a utilização de programas existentes aumenta significativamente a eficiência da programação.

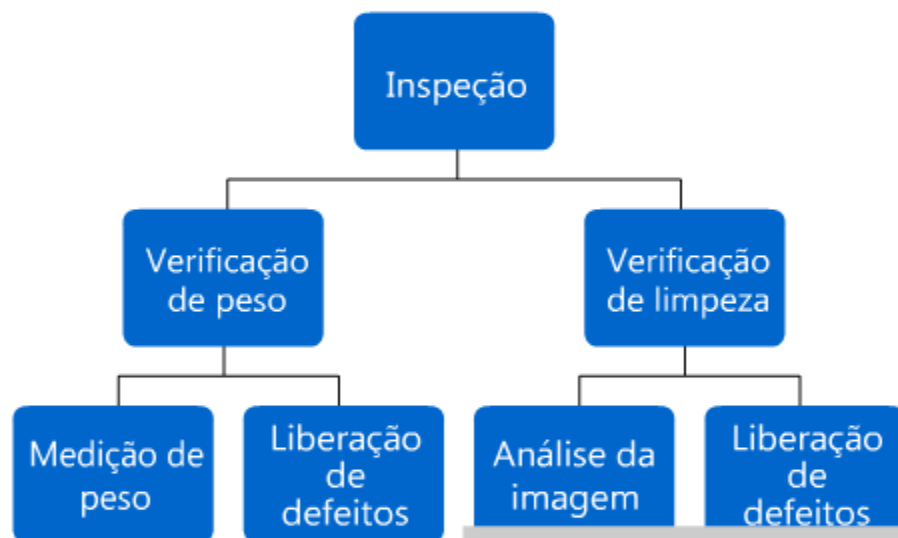
O conceito básico por trás da criação de programas estruturados é descrito nas próximas seções.


1.3.2

Seleção eficiente de seções utilizáveis

A animação a seguir descreve o conceito de organização hierárquica. O uso da organização hierárquica possibilita que seções utilizáveis sejam selecionadas de forma mais eficiente.

As camadas de POU dessa forma possibilitam que programas sejam vistos em partes menores e em um nível maior ao mesmo tempo, de forma que se pode selecionar as seções utilizáveis de forma mais eficiente.



Clique em  na parte superior direita para iniciar a animação novamente do início.

1.3.2

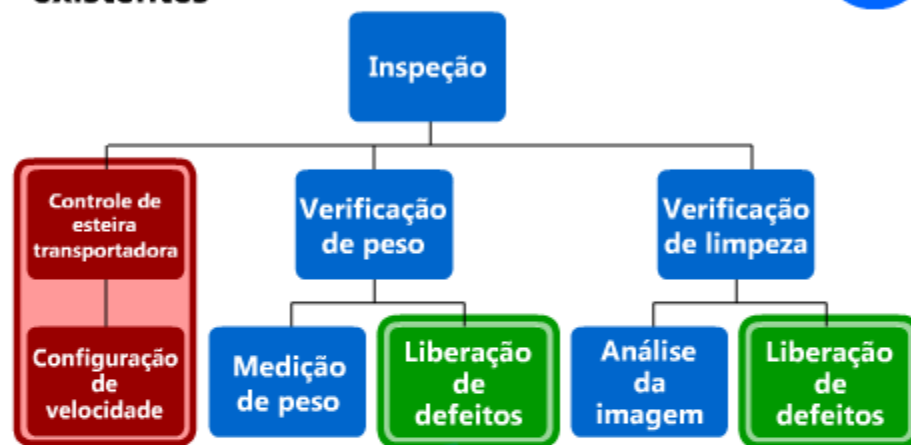
Seleção eficiente de seções utilizáveis

Esse exemplo ilustra a eficácia de usar seções de um programa organizado de forma hierárquica.


Programa existente



Novo programa utilizando programas existentes



Uso de múltiplas seções

Clique em  na parte superior direita para iniciar a animação novamente do início.

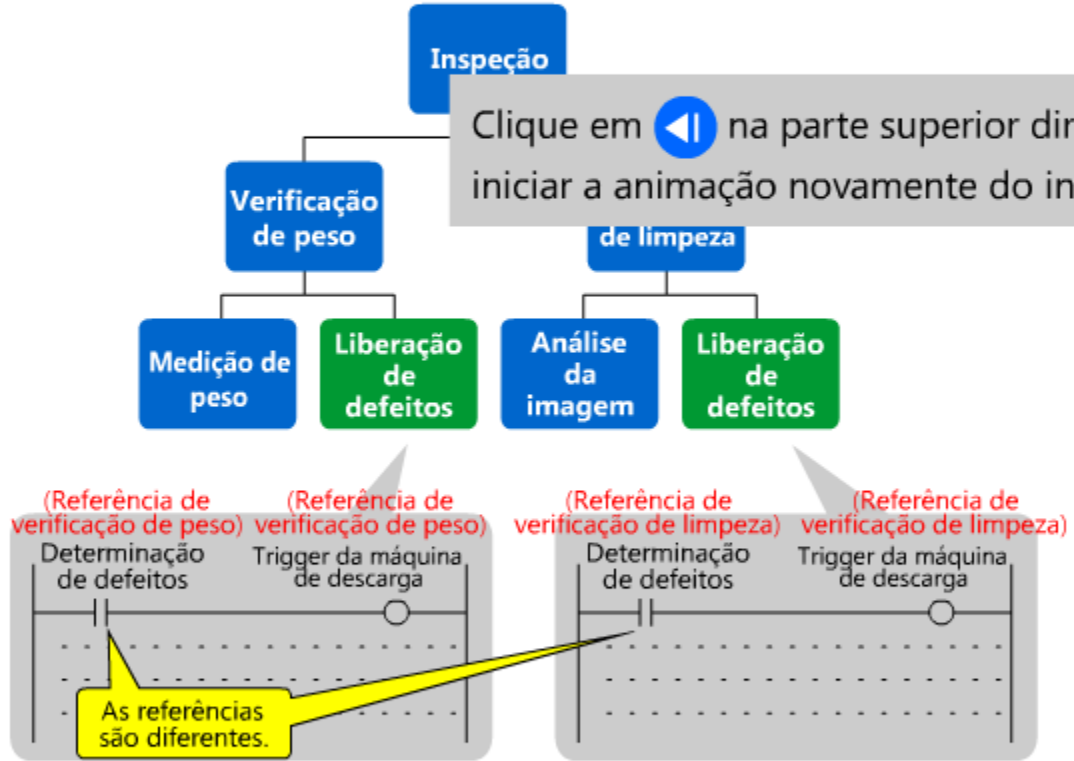
1.3.3 Utilização de programas sem precisar reatribuir dispositivos

Você aprendeu como programas estruturados facilitam a utilização de programas. Com a programação estruturada, labels são usadas no lugar de dispositivos. O uso de labels permite que programadores criem programas sem se preocupar com conflitos de dispositivo e o novo programa.

As verificações de duplicação dos dispositivos não são necessárias.



Clique em ◀ na parte superior direita para iniciar a animação novamente do início.



1.3.4

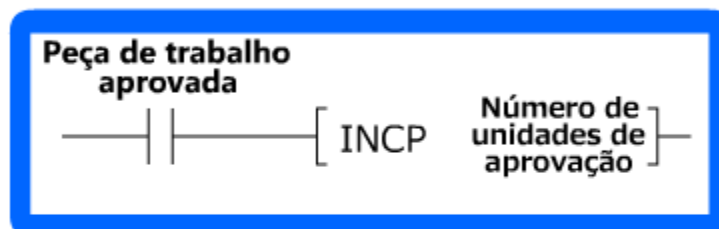
Programação usando labels para processamento específico

O uso de labels permite que sejam criados programas sem precisar lidar com comentários e dispositivos.


O uso de labels permite que programas sejam criados com os nomes de processamento sem precisar se preocupar com dispositivos.



Uso de labels



Intuitivo e de fácil compreensão

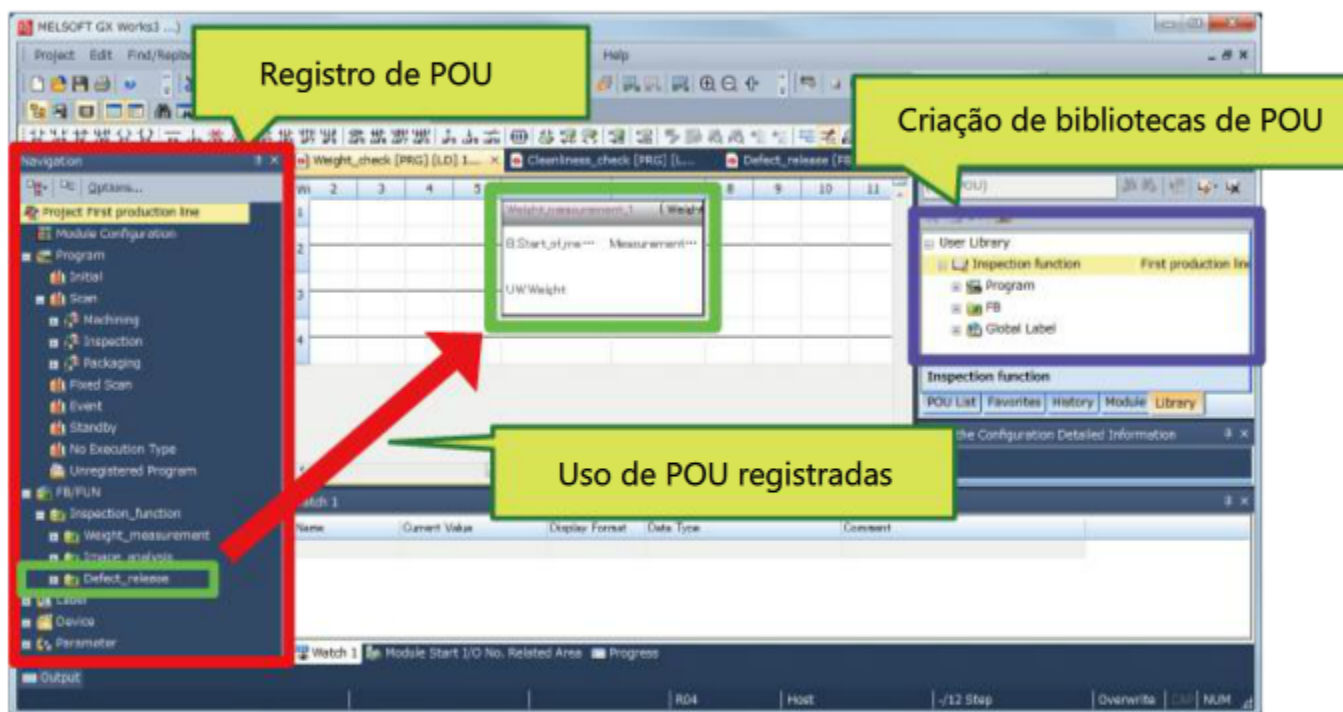
Clique em  na parte superior direita para iniciar a animação novamente do início.

1.4

Programação estruturada com MELSOFT GX Works3

O software de engenharia MELSOFT GX Works3 oferece suporte para a criação de programas estruturados.

O MELSOFT GX Works3 facilita a utilização de programas exibindo as unidades de organização do programa (POU) disponíveis para fácil seleção.



MELSOFT GX Works3

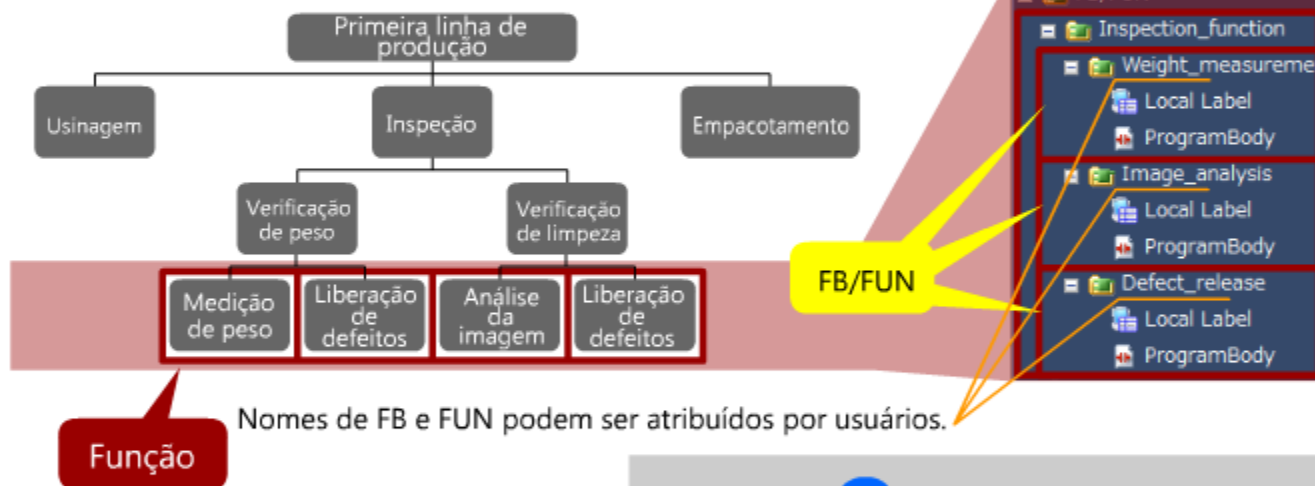
1.4.1


Como a organização hierárquica de programas é representada no GX Works3

A animação a seguir descreve como o GX Works3 representa a organização hierárquica de programas, usando um exemplo de linha de produção.

O processamento de verificação de peso inclui medição de peso e funções de liberação de defeitos.

A medição de peso, análise de imagem e funções de liberação de defeitos incluídas no processamento são representadas como níveis menores no nível FB/FUN.



Clique em  na parte superior direita para iniciar a animação novamente do início.

1.4.2

Uso de unidades de organização do programa (POU)

Essa seção explica o procedimento para usar POU no GX Works3. FB/FUN são selecionados na janela de projeto e organizados no programa.

The image shows the GX Works3 interface with a project tree on the left, a ladder logic diagram at the top, and a program code editor at the bottom. A blue callout box highlights the 'Weight_check' unit in the project tree. A red callout box highlights the 'Weight_measurement' unit in the project tree. A red callout box highlights the 'Weight_measurement' unit in the ladder logic diagram. A red callout box highlights the 'Weight_measurement' unit in the program code editor. A blue callout box highlights the 'Weight_measurement' unit in the program code editor.

Project Structure:

- Project First production line
 - Module Configuration
 - Program
 - Initial
 - Scan
 - Machining
 - Inspection
 - Weight_check (highlighted)
 - Local Label
 - ProgramBody
 - Cleanliness_check
 - Packaging
 - Fixed Scan
 - Event
 - Standby
 - No Execution Type
 - Unregistered Program
 - FB/FUN
 - Inspection_function
 - Weight_measurement (highlighted)
 - Local Label
 - ProgramBody
 - Image_analysis
 - Defect_release

Ladder Logic Diagram:

```

graph TD
    Insp[Inspeção] --> VerP[Verificação de peso]
    Insp --> VerL[Verificação de limpeza]
    VerP --> MedP[Medição de peso]
    VerP --> LibP[Liberação de defeitos]
    VerL --> AnaI[Análise da imagem]
    VerL --> LibL[Liberação de defeitos]
  
```

Program Code:


```

1 2 3 4 5 6 7 8 9 10 11 12
  Weight_measurement_I (Weight_measurement)
  B Start_of_measurement Measurement_result UW
  UW:Weight
  END
  
```

Program Code (Expanded):

```

1 2 3 4 5 6 7 8 9 10 11 12
< Weight K100 MOV K1 Measurement_result
= Weight K100
> Weight K100
  
```

Clique em  na parte superior direita para iniciar a animação novamente do início.

1.4.3 Uso de bibliotecas

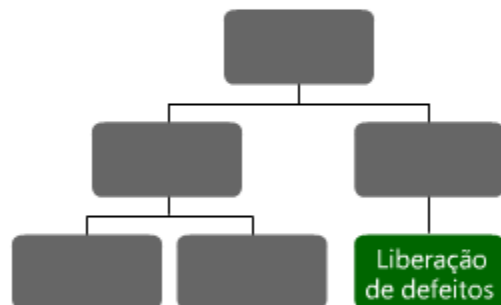
A POU pode ser compartilhada entre os projetos usando a função de biblioteca. A criação de bibliotecas de POU validadas ajuda a garantir uma qualidade consistente de programas.


POU podem ser usadas simplesmente arrastando-as e soltando no programa.



ProgPou [PRG] [LD] 12Step						
2	3	4	5	6	7	
				Defect_release 2	(Defect_release)	

User Library	
Inspection function	First production line
Program	
FB	
Defect_release	
Image_analysis	
Weight_measurement	
Global Label	



Clique em  na parte superior direita para iniciar a animação novamente do início.


1.4.3 Uso efetivo de bibliotecas

É de grande efetividade criar bibliotecas de POU usadas com frequência. Nos programas de inspeção a seguir, a FB/FUN de "liberação de defeitos" é a mais adequada para registro na biblioteca.

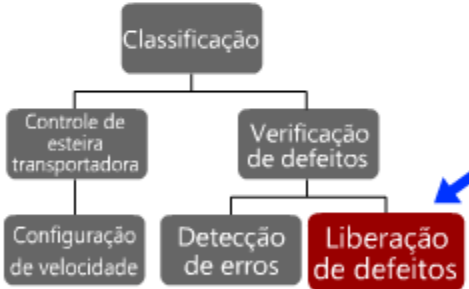


POU usadas com frequência são registradas.



Clique em  na parte superior direita para iniciar a animação novamente do início.

Equipamento A



Equipamento B



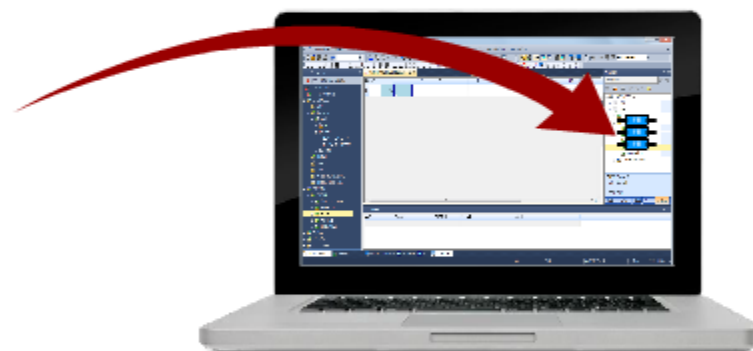
1.4.4 Uso de POU pré-instaladas


O GX Works3 é pré-configurado com POU para muitos programas típicos. A POU pré-instalada inclui FB de módulo registrada automaticamente de acordo com a configuração de módulo e a MELSOFT Library (biblioteca MELSOFT).



A MELSOFT Library fornecida pela Mitsubishi Electric possui diversas FBs úteis disponíveis para os usuários.

As FBs ajudam você a usar os módulos fabricados pela Mitsubishi Electric ou produtos de outros fabricantes.



Clique em  na parte superior direita para iniciar a animação novamente do início.

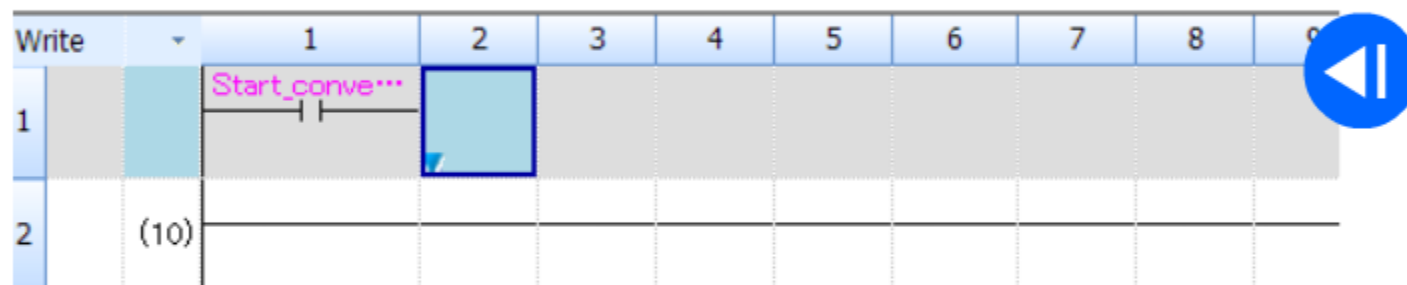
Para acessar a MELSOFT Library, entre em contato com um representante Mitsubishi Electric local.

1.4.5 Cadastro eficiente de labels

O recurso de texto preditivo ajuda você a cadastrar labels de forma mais eficiente.

Ao começar a cadastrar uma label, uma lista de labels registradas que correspondam aos primeiros caracteres inseridos aparece automaticamente.

Em seguida, a label desejada é cadastrada por seleção. Isso possibilita que você insira a mesma label em múltiplos locais sem nenhum erro de entrada.



Clique em ◀ na parte superior direita para iniciar a animação novamente do início.

Programação usando labels registradas

O conteúdo desse capítulo é:

- Estado atual da indústria FA
- Pontos negativos dos programas que usam Ladder simples
- Conceitos e vantagens de programas estruturados
- Programação estruturada com MELSOFT GX Works3

Pontos importantes a serem levados em conta:

Benefícios do uso de programas	A utilização de programas reduz a quantidade de novo desenvolvimento necessário. Maximizar a utilização de programas existentes aumenta significativamente a eficiência da programação.
Divisão de programas em partes de programas	Seções utilizáveis podem ser percebidas ao dividir programas em partes de programa e atribuir nomes significativos a essas partes.
Organização hierárquica	Além de dividir programas em partes de programas, o uso de uma organização hierárquica para programas simplifica e torna mais eficiente a seleção de seções utilizáveis dos programas.
Vantagens do uso de labels	<ul style="list-style-type: none">• O uso de labels elimina a necessidade de verificações de conflito de dispositivo e reatribuição de dispositivos, o que torna a utilização de programas mais eficiente• A programação é mais intuitiva quando nomes significativos são usados para representar o processamento específico
Programas estruturados no MELSOFT	<ul style="list-style-type: none">• O MELSOFT possibilita a criação de programas estruturados em um editor visual de fácil compreensão• O MELSOFT melhora a eficiência de programação

Capítulo 2 Por que programas estruturados são tão eficientes?

Este capítulo descreve os benefícios da programação estruturada.

O processo de programação será descrito conforme dividido nas fases a seguir.

- Edição de programa: Criação e edição do programa
- Testes de avaliação: Confirmar que o programa funciona corretamente
- Pesquisar e corrigir: Identificar e corrigir falhas

2.1 Diminuir o tempo de programação utilizando programas

2.2 Remover o processo de reatribuição de dispositivo

2.3 Evitar mudanças acidentais

2.1 Diminuir o tempo de programação utilizando programas


O uso de programas existentes diminui o tempo de reatribuição e melhora a eficiência de programação.

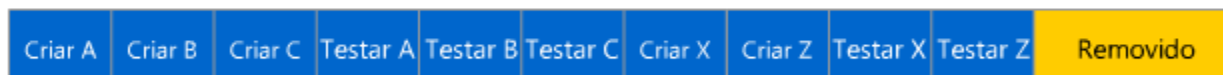
Programação eficiente usando programas estruturados



Processo de programação



Clique em  na parte superior direita para iniciar a animação novamente do início.

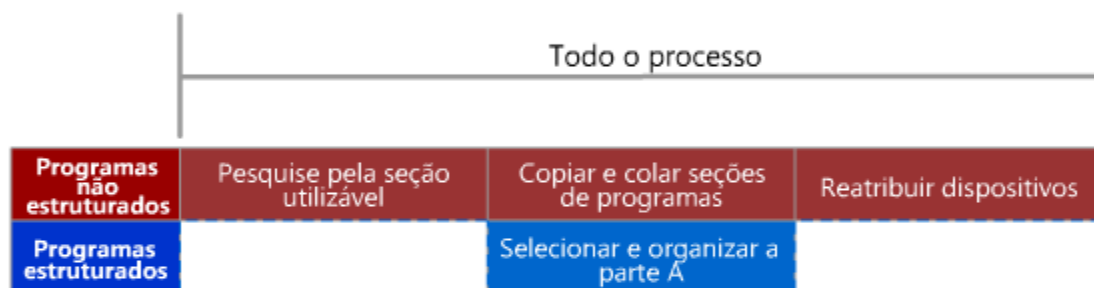



O uso de programas estruturados elimina a necessidade de reatribuir dispositivos em um novo programa.

Programação eficiente usando programas estruturados



Processo de programação



Clique em  na parte superior direita para iniciar a animação novamente do início.

Selecionar e organizar a parte A

Removido

2.3 Impedir mudanças acidentais



Mudanças acidentais que possam ocorrer com a adição de recursos aos programas podem ser evitadas através do uso de programas estruturados.

Programação eficiente usando programas estruturados



Processo de programação




 Clique em  na parte superior direita para iniciar a animação novamente do início.

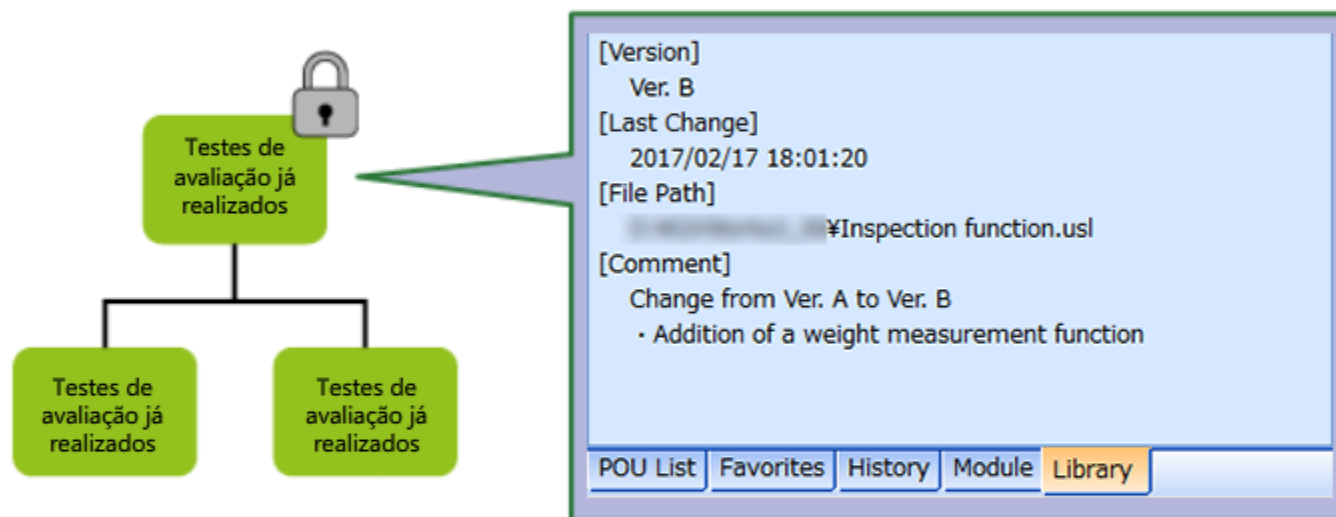


2.3.1

Gestão segura de POU

A POU pode ser gerenciada de forma segura usando o MELSOFT GX Works3.

Mudanças acidentais podem ser totalmente eliminadas ao bloquear cada POU individualmente. Cada POU pode ter suas próprias informações, incluindo a versão, a data de atualização e os comentários. O histórico de mudanças pode ser registrado nos comentários.



O conteúdo desse capítulo é:

- Benefícios do uso da POU
- Benefícios do uso de labels
- Benefícios do uso de programas estruturados

Pontos importantes a serem levados em conta:

Redução nos testes de avaliação	O uso de programas estruturados reduz não apenas o tempo necessário para criar programas, mas também o necessário para os testes de avaliação.
Impedir mudanças acidentais	Com programas estruturados, mudanças são feitas à POU individual, que evita que mudanças acidentais sejam feitas em outras partes do programa.
Bloqueio da POU	O bloqueio da POU após a conclusão do teste de avaliação garante a gestão segura da POU.

Este curso descreveu o uso da programação estruturada para aumentar a eficiência da programação. Você concluiu o curso. Aqui está um resumo do que foi tratado nesse curso.

- A programação estruturada é um método eficiente de criar programas que se tornaram cada vez mais complexos juntamente com tendências de sistemas de grande escala para a produção e de quantidade elevada de dados.
- O uso de programas estruturados é mais fácil, reduzindo a quantidade de novo desenvolvimento e melhorando a eficiência da programação
- A divisão de programas em partes de programa possibilita que eles sejam divididos por processamento e função. O uso de uma organização hierárquica e nomes significativos para partes de programa esclarece as seções dos programas que podem ser utilizadas.
- O uso da organização hierárquica possibilita que sejam selecionadas seções utilizáveis de forma mais eficiente
- O uso de etiquetas permite que a POU seja utilizável sem precisar se preocupar com conflitos entre o dispositivo e o novo programa
- O MELSOFT GX Works3 oferece suporte para programação estruturada e melhora a eficiência da programação
- A criação de bibliotecas permite o compartilhamento/padronização de partes de programas usados com frequência, alcançando programas de qualidade consistente
- Além de utilizar programas, as bibliotecas existentes podem ser utilizadas para reduzir a quantidade de novo desenvolvimento

Ao implementar a programação estruturada de fato, consulte os seguintes recursos.

Programação com MELSOFT GX Works3

- Curso de e-Learning "GX Works3 (Ladder)"
- Manuais de operações para MELSOFT GX Works3

Estrutura de programa e sintaxe

- Manuais de programação da Série MELSEC iQ-R

Agora que você concluiu todas as lições do curso **Programação eficiente do CLP**, está pronto para fazer o teste final. Se tiver qualquer dúvida sobre os tópicos abrangidos, aproveite esta oportunidade para revê-los.

O Teste Final é composto por 9 perguntas (10 itens).

Você pode fazer o teste final quantas vezes desejar.

Como é feita a pontuação do teste

Depois de selecionar a resposta, não se esqueça de clicar no botão **Resposta**. Sua resposta será perdida se você continuar sem clicar nesse botão. (O sistema assumirá que essa pergunta não foi respondida.)

Resultados da pontuação

O número de respostas corretas, o número de perguntas, a porcentagem de respostas corretas e o resultado (aprovado/reprovado) aparecem na página de pontuação.

Respostas corretas: 5

Total de perguntas: 5

Porcentagem: 100%

Para passar no teste, você precisa responder corretamente a **60%** das perguntas.

Continuar

Rever

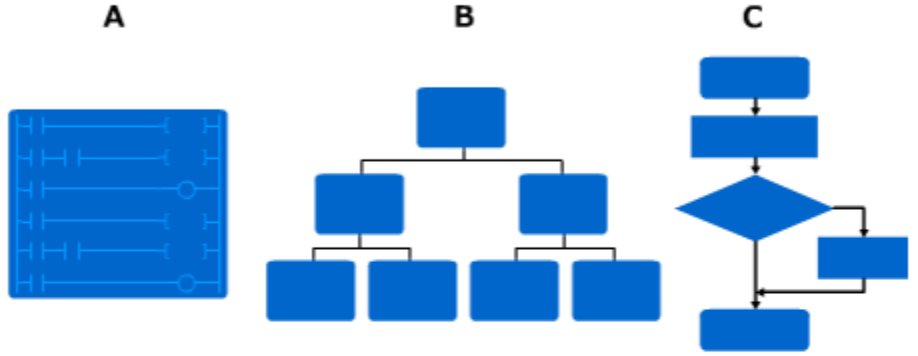
- Clique no botão **Continuar** para sair do teste.
- Clique no botão **Rever** para rever o teste. (Verificar a resposta correta)
- Clique no botão **Repetir** para refazer o teste.

Teste Final 1

Programas estruturados

Selecione a ilustração correta de programas estruturados.

- A
- B
- C



Resposta

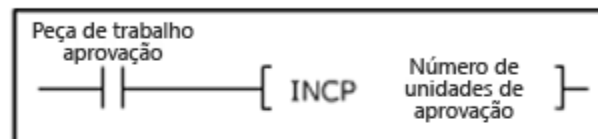
Voltar

Uso de labels

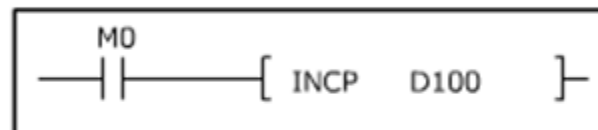
Selecione o programa criado usando labels.

- A
- B
- C

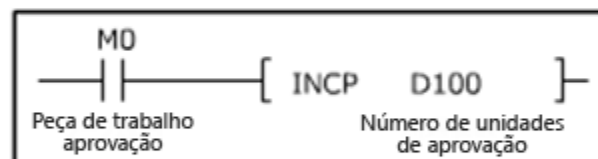
A



B



C



Resposta

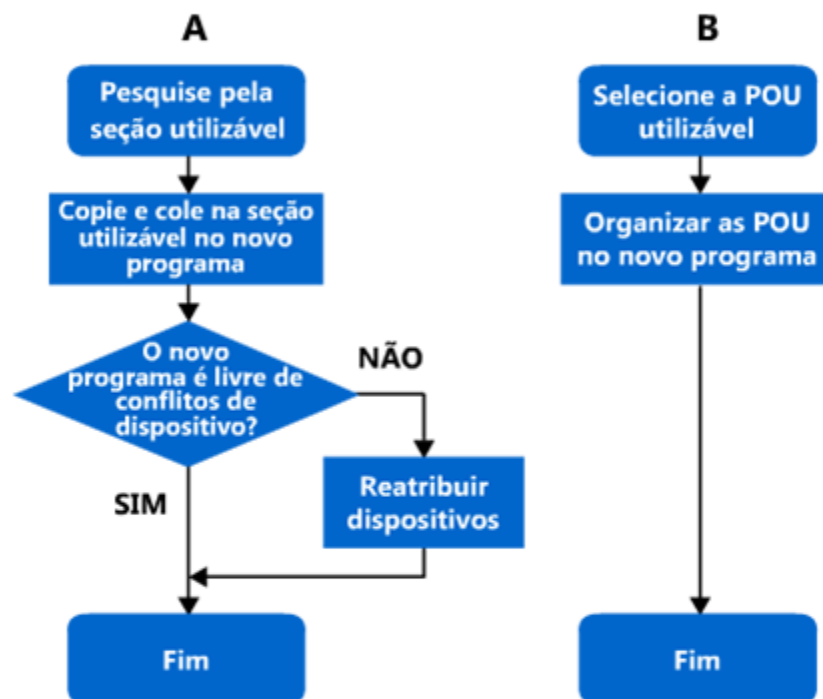
Voltar

Processo do uso de programas

Selecione o fluxograma que ilustra corretamente o processo de usar programas estruturados criados com labels.

A

B



Resposta

Voltar

Finalidade de programas estruturados

Selecione a descrição correta sobre a finalidade dos programas estruturados. (Múltiplas respostas)

- Melhorar a velocidade de processamento do programa
- Melhorar a eficiência da programação
- Reduzir o tempo de compilação do programa
- Melhorar a visualização de programas

Resposta

Voltar

Programação estruturada no GX Works3

Selecione a descrição correta sobre a criação de programas estruturados no GX Works3.

- Programas estruturados podem ser criados imediatamente depois da simples instalação do software.
- A criação de programas estruturados no GX Works3 exige a compra de um plug-in.

Resposta

Voltar

Divisão de programas em partes de programas

Selecione todas as afirmações que descrevem a divisão das partes do programa. (Múltiplas respostas)

- Dividir programas em um número igual e predeterminado de etapas
- Dividir programas em funções e processamento individuais
- Atribuir nomes significativos a processos
- Usar cursores para acessar diferentes seções
- As unidades de organização do programas (POU) são o resultado da divisão do programa em partes

Resposta

Voltar

Vantagens de bibliotecas

Selecione a descrição correta sobre as vantagens de registrar programas em bibliotecas. (Múltiplas respostas)

- O registro dum POU usada com frequência possibilita a utilização eficiente de programas
- Impede a utilização da POU
- A POU pode ser compartilhada entre muitas pessoas
- O registro e a reutilização de programas padronizados garantem a qualidade consistente dos programas

Resposta

Voltar

Utilização de bibliotecas de FB de módulo e a MELSOFT Library

Selecione a descrição correta sobre a utilização de bibliotecas como a function block do módulo e a MELSOFT Library.

- A operação interna não precisa ser verificada quando são usadas bibliotecas pré-instaladas
- FB de módulo deve ser criada de acordo com o modelo do módulo

Resposta

Voltar

Bibliotecas pré-configuradas

Selecione a opção que representa a biblioteca fornecida pela Mitsubishi Electric.

P1 FB de módulo

P2 MELSOFT Library

P1

P2

Resposta

Voltar

Você concluiu o Teste Final. Seus resultados são os seguintes.
Para terminar o Teste Final, vá para a próxima página.

Respostas corretas: **9**

Total de perguntas: **9**

Porcentagem: **100%**

Parabéns. Você passou no teste.

Você concluiu o curso **Programação eficiente do CLP**.

Muito obrigado por fazer este curso.

Esperamos que tenha gostado das lições e que as informações adquiridas sejam úteis no futuro.

Você pode rever o curso quantas vezes quiser.

Rever

Fechar