



# PLC

## Conceptos Avanzados de GX Works2

Este curso brinda conocimientos sobre las funciones para mejorar el ambiente de desarrollo del sitio de diseño que presente problemas con la "productividad", "calidad", "administración de proyecto" y "medidas de seguridad". El curso está dirigido a programadores secuenciales que ya utilicen MELSOFT GX Works2.



## **Introducción** **Objetivo del curso**

Este curso brinda conocimientos sobre las funciones para mejorar el ambiente de desarrollo del sitio de diseño que presente problemas con la "productividad", "calidad", "administración de proyecto" y "medidas de seguridad". El curso está dirigido a programadores secuenciales que ya utilicen MELSOFT GX Works2.

## Introducción Estructura del curso

El contenido de este curso es el siguiente.  
Recomendamos comenzar desde el Capítulo 1.

### Capítulo 1 – Mejora del ambiente de desarrollo con GX Works2

Aprenderá sobre los problemas que enfrenta el sitio de diseño y el ambiente de desarrollo necesario para resolverlos.

### Capítulo 2 - Programación

Aprenderá sobre las funciones utilizadas para programar.

### Capítulo 3 - Depuración

Aprenderá sobre las funciones utilizadas para depurar.

### Capítulo 4 – Administración de proyecto y medidas de seguridad

Aprenderá sobre las funciones para la administración de proyecto y las medidas de seguridad en la etapa de desarrollo y mantenimiento después del inicio de la operación del sistema.

### Prueba final

Calificación para aprobar: 60 % o superior

## Introducción **Cómo usar esta herramienta de aprendizaje en línea**

Ir a la página siguiente		Ir a la página siguiente.
Regresar a la página anterior		Regresar a la página anterior.
Ir a la página deseada		Se visualizará el "Índice", lo que le permitirá navegar a la página deseada.
Salir del aprendizaje		Salir del aprendizaje. El aprendizaje y las ventanas como la pantalla de "Contenidos" se cerrarán.



## Introducción Precauciones del uso

### Precauciones de seguridad

Cuando aprenda usando productos reales, lea con cuidado las precauciones de seguridad ubicadas en los manuales correspondientes.

### Precauciones en este curso

- Es posible que las pantallas visualizadas de la versión del software que use sean diferentes a las de este curso.

# Capítulo 1 Mejora del ambiente de desarrollo con GX Works2

## Pasos para el aprendizaje en el Capítulo 1

Este curso está dirigido a los programadores que trabajan en el desarrollo de programas secuenciales. Aprenderá cómo usar las excelentes funciones de **MELSOFT GX Works2** para resolver los problemas del sitio de diseño relacionados con la **"productividad", "calidad", "administración de proyecto"** y **"seguridad"**.

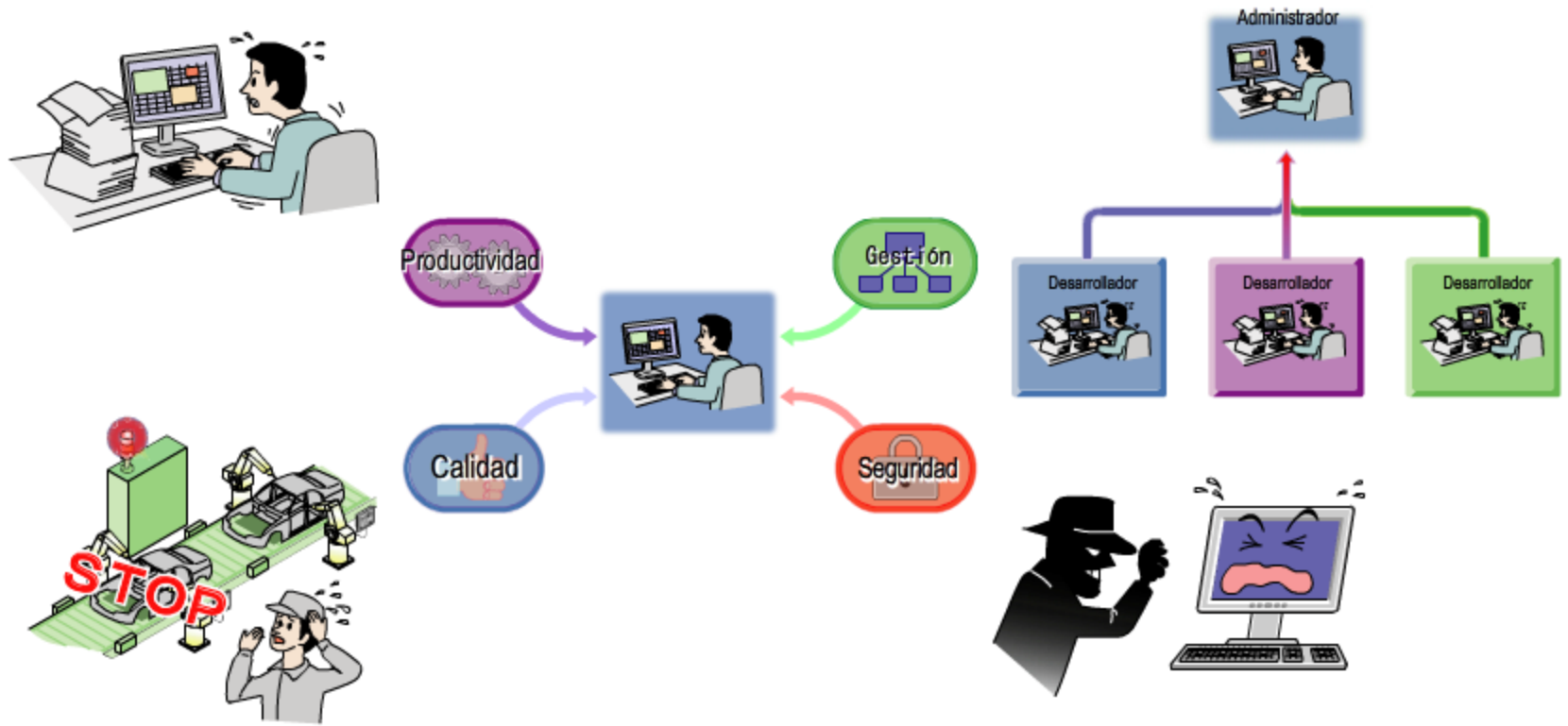
En el Capítulo 1, aprenderá sobre los problemas que enfrenta el sitio de diseño y el ambiente de desarrollo necesario para resolverlos.

- 1.1 Problemas que enfrenta el sitio de diseño
  - 1.1.1 Mejora de la productividad
  - 1.1.2 Mejora de la calidad
  - 1.1.3 Administración de proyecto
  - 1.1.4 Medidas de seguridad
- 1.2 Procedimiento de aprendizaje



# 1.1 Problemas que enfrenta el sitio de diseño

Si bien el sitio de diseño debe mejorar la productividad para reducir los costos de desarrollo, también debe garantizar que los programas desarrollados sean de alta calidad.  
 El sitio de diseño debe también realizar una administración de proyecto que permita el desarrollo en equipo y garantice una recuperación rápida en caso de errores.  
 También son necesarias las medidas de seguridad ya que los programas secuenciales incluyen conocimiento y datos importantes.



## 1.1.1 Mejora de la productividad

Es necesario que el sitio de diseño desarrolle programas secuenciales para manejar el procesamiento complejo a gran escala. Sin embargo, el costo de desarrollo aumenta a la par de la duración del período de desarrollo. Por lo tanto, se le pide a usted mejorar la productividad del desarrollo del programa a fin de reducir el costo de desarrollo.

GX Works2 brinda las siguientes funciones para resolver este problema:

- Etiqueta
- Bloque de función
- Valor inicial de dispositivo y memoria de dispositivo\*
- Texto estructurado en línea
- Importación desde comentario de muestra

\* La serie MELSEC-F no puede configurar los valores iniciales del dispositivo.

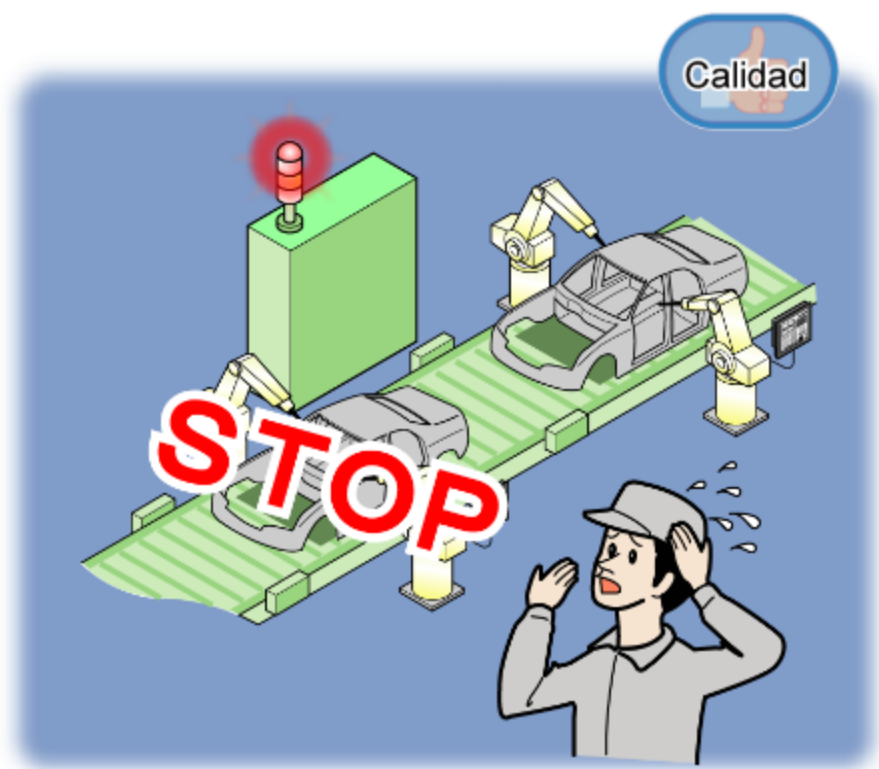


## 1.1.2 Mejora de la calidad

Un error en el programa secuencial podría causar una parada, falla o accidente del sistema, lo que detendría la producción y ocasionaría la pérdida de ganancias y fiabilidad. Se le pide desarrollar programas de calidad que estén libres de errores.

GX Works2 brinda las siguientes funciones para resolver este problema:

- Etiqueta
- Bloque de función
- Valor inicial de dispositivo y memoria de dispositivo\*
- Comentario
- Importación desde comentario de muestra
- Observación
- Referencia cruzada
- Seguimiento de muestreo\*
- Prueba de dispositivo de ejecución condicionada\*
- Función de ejecución escalonada\*
- Ajuste del sistema de E/S



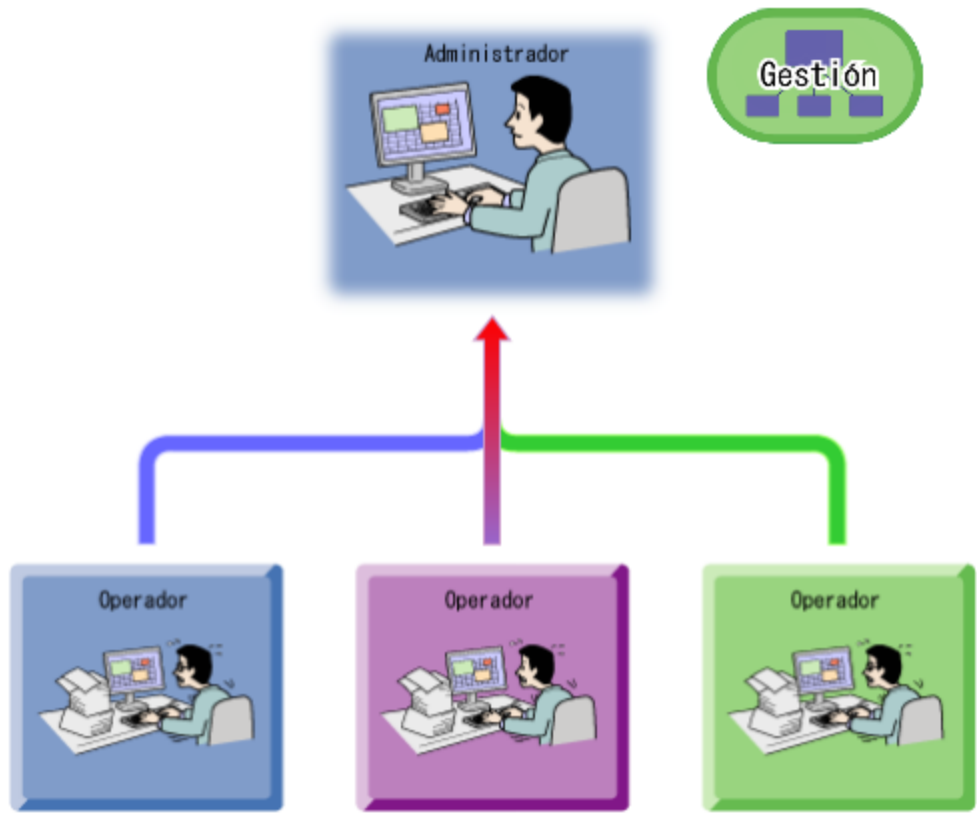
\* La serie MELSEC-F no puede usar los ajustes de valor inicial de dispositivo, el seguimiento de muestreo, la prueba de dispositivo de ejecución condicionada ni la función de ejecución escalonada.

# 1.1.3 Administración de proyecto

Por lo general, el desarrollo de programas secuenciales a gran escala está a cargo de un equipo de programadores. Desde el punto de vista de la autoridad de gestión individual y la confidencialidad de los datos, deben establecerse los límites para el rango de datos accesibles y funciones utilizables. Además, para garantizar una recuperación rápida en caso de pérdida del programa debido a falla del controlador programable, se le pide implementar la gestión de versiones de programa y una copia de respaldo periódica del programa.

GX Works2 brinda las siguientes funciones para resolver estos problemas:

- Seguridad
- Historial de revisión
- Verificar con PC



## 1.1.4 Medidas de seguridad

El programa secuencial incluye conocimiento y datos de importancia estratégica.

La **fuga** de know-how y datos del programa al exterior podría tener un efecto devastador en el negocio.

La **modificación no autorizada** del programa podría acarrear problemas de producción, como la parada del sistema. Será preciso tomar las medidas de seguridad pertinentes para evitar estos problemas.

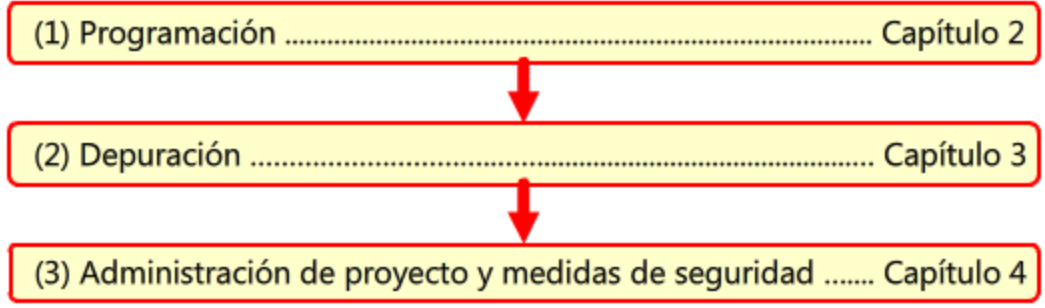
GX Works2 brinda las siguientes funciones para resolver estos problemas:

- Seguridad



## 1.2 Procedimiento de aprendizaje

En este curso, aprenderá sobre las funciones de GX Works2 siguiendo el procedimiento para el desarrollo del sistema real.



### <Explicación de los íconos>

Los íconos que aparecen en las páginas de los capítulos 2, 3 y 4 corresponden a las respectivas funciones de mejora, tal y como se muestra a continuación.



Función usada para mejorar la productividad del programa



Función usada para mejorar la calidad del programa



Función usada para la administración de proyecto



Función usada para las medidas de seguridad



# Capítulo 2 Programación

## Pasos para el aprendizaje en el Capítulo 2

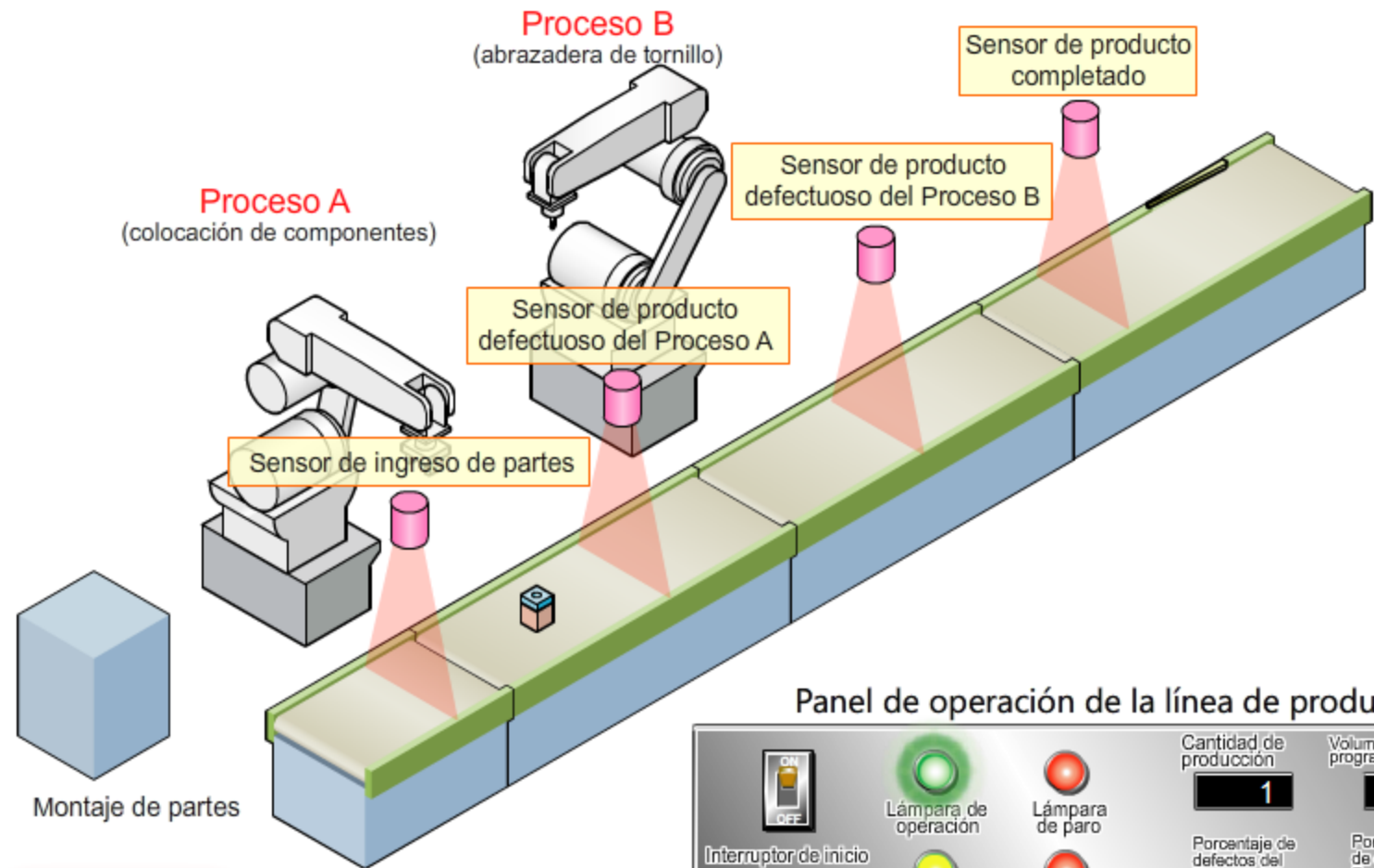
En el Capítulo 2, aprenderá sobre las funciones usadas para programar. GX Works2 brinda muchas funciones distintas para garantizar la programación eficiente y mejorar la calidad de los programas.

- 2.1 Cambiar el nombre del dispositivo por un nombre asociado a la aplicación
  - 2.1.1 Tipos de etiqueta
  - 2.1.2 Tipos de aplicaciones de etiquetas y valores guardados
  - 2.1.3 Registro de etiqueta
  - 2.1.4 Asignación automática de etiquetas a los dispositivos
- 2.2 Disponer bloques en escalera de uso reiterado como bloques de función para desviación
  - 2.2.1 Crear y colocar bloques de función
  - 2.2.2 Usar la biblioteca de bloques de función
- 2.3 Cambiar los valores iniciales del dispositivo sin corregir el programa
- 2.4 Simplificar los programas en escalera
- 2.5 Crear programas que sean fáciles de entender y leer
  - 2.5.1 Escribir un comentario para cada bloque en escalera
  - 2.5.2 Escribir un comentario para cada salida (bobina, instrucción de aplicación)
- 2.6 Hacer fácil la lectura de programas que contengan relés/registros especiales

# Capítulo 2 Introducción del sistema usado para el aprendizaje en este curso



Este curso emplea el siguiente sistema hipotético para ayudarlo a aprender cómo utilizar las funciones de GX Works2:



- ▶ Producción normal
- ▶ Error de porcentaje de defectos del Proceso A
- ▶ Error de porcentaje de defectos del Proceso B

Al pulsar el botón "Volver al inicio" se regresa a la primera selección de operación.

**Volver al inicio**

## Panel de operación de la línea de producción

			Cantidad de producción	Volumen de producción programado
Interruptor de inicio	Lámpara de operación	Lámpara de paro	1	20
			Porcentaje de defectos del Proceso A	Porcentaje permitido de defectos del Proceso A
Reinicio de la cantidad de producción	Cantidad de producción obtenida	Lámpara de error de porcentaje de defectos del Proceso A	0	10
			Porcentaje de defectos del Proceso B	Porcentaje permitido de defectos del Proceso B
		Lámpara de error de porcentaje de defectos del Proceso B	0	5

## 2.1

## Cambiar el nombre del dispositivo por un nombre asociado a la aplicación

Productividad

Calidad

A los **dispositivos** usados en el programa secuencial se les asigna un nombre que se compone de una letra seguida por un número, por ejemplo "M0" o "D5".

De esta forma, el nombre del dispositivo no ofrece ninguna pista sobre la aplicación del dispositivo.

Un programa a gran escala usa muchos tipos distintos de dispositivo, lo que significa que durante la programación, usted debe verificar constantemente los documentos de diseño del sistema para determinar la aplicación de los dispositivos. Esto reduce la eficiencia del trabajo y afecta negativamente la calidad del programa debido a los eventuales errores en la selección de dispositivos.

Es posible usar "**etiquetas**" para resolver estos problemas.

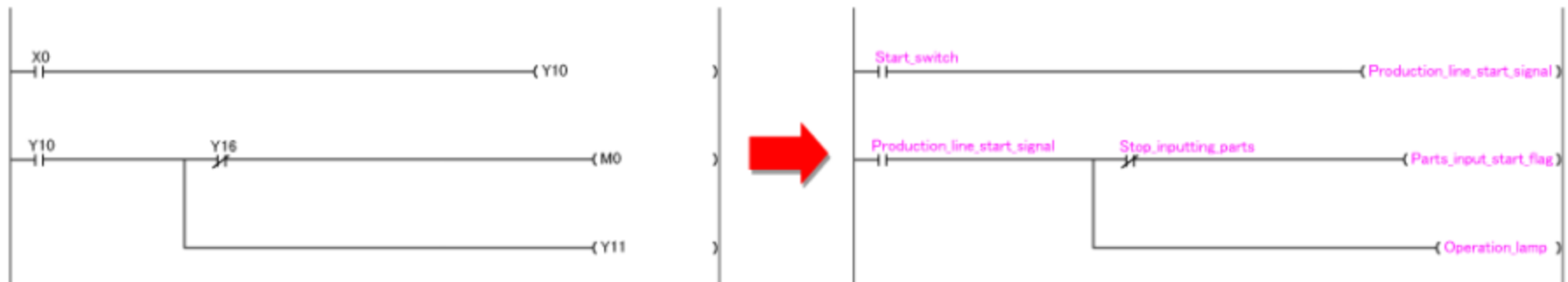
En lugar de usar un nombre de dispositivo, puede usar un nombre (etiqueta) que indique la aplicación real, por ejemplo "**Señal de inicio de línea de producción**".

Para este tipo de nombre, se puede utilizar caracteres japoneses (hiragana, katakana y kanji) además de los caracteres alfanuméricos.

**<Declaración para ajustar "M0" que contenga la señal de inicio de línea de producción a ENCENDIDO mediante instrucción AJUSTE>**

Dispositivo	—[ SET M0 ]
Etiqueta	—[ SET <b>Production line start signal</b> ]

El uso de este tipo de etiqueta es efectivo para crear un programa de fácil lectura, mejorando la eficiencia del desarrollo del programa y evitando los errores de entrada del dispositivo.



# 2.1.1 Tipos de etiquetas



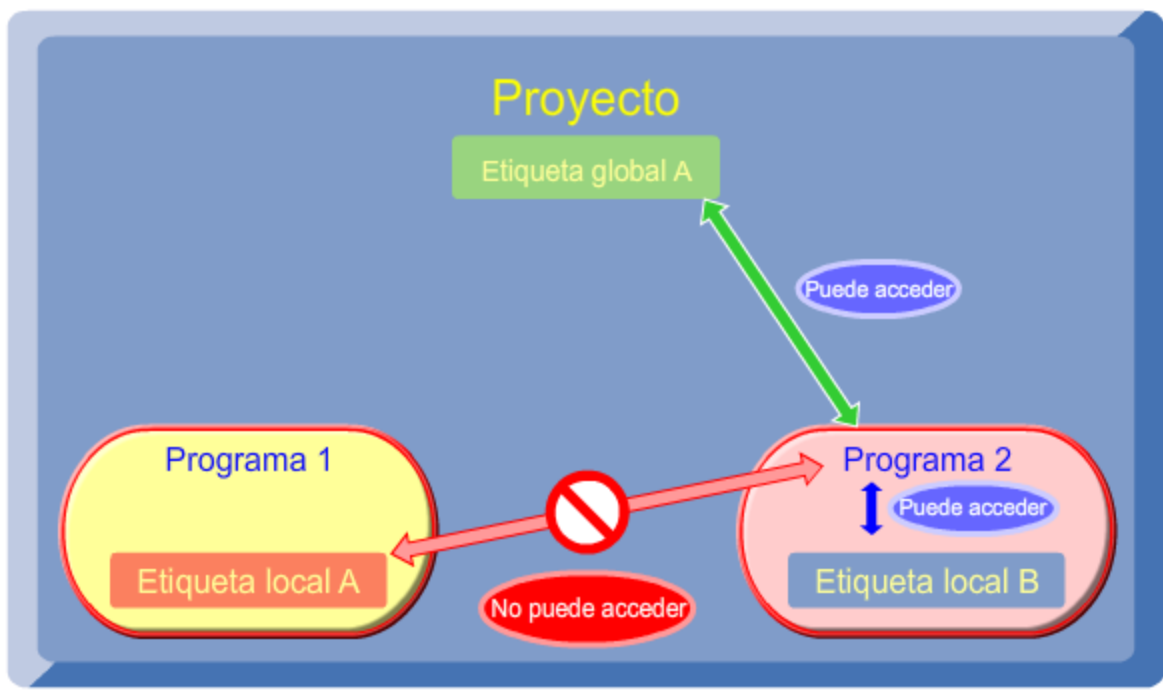
Hay dos tipos de etiquetas: "etiqueta global" y "etiqueta local".

### <Etiqueta global>

Las etiquetas globales se usan para un proyecto completo y cualquier programa de ese proyecto puede acceder a ellas.

### <Etiqueta local>

Las etiquetas locales se usan en un programa específico y solo el programa en el que esté registrada la etiqueta puede acceder a ellas.



"Programa 1" y "Programa 2" son botones. Pulse cualquiera de los programas para ver si puede acceder a los dos tipos de etiqueta.

## 2.1.2 Tipos de aplicaciones de etiquetas y valores guardados



Al registrar una etiqueta, especifique la aplicación de la misma y el tipo de valor que puede almacenarse utilizando **"Clase"** y **"Tipo de datos"**.

### <Clase>

La clase indica el rango de uso y la aplicación de una etiqueta.  
Las clases que pueden seleccionarse varían según el tipo de etiqueta.

Clase	Uso previsto	Área de ajuste de etiqueta		
		Etiqueta global	Programa local etiqueta	Bloque de función <Etiqueta local>
VAR_GLOBAL	Etiqueta común que puede ser usada por programas y bloques de función en un proyecto	○	✗	✗
VAR_GLOBAL_CONSTANT	Etiqueta común con una constante que puede ser usada por programas y bloques de función en un proyecto	○	✗	✗
VAR	Etiqueta que puede ser usada por programas y bloques de función para los cuales está prevista la etiqueta	✗	○	○
VAR_CONSTANT	Etiqueta con una constante que puede ser usada por programas y bloques de función para los que está prevista la etiqueta	✗	○	○
VAR_RETAIN	Etiqueta (tipo Latch) que se puede utilizar en los programas y bloques de función para los que está prevista la etiqueta	✗	○	○
VAR_INPUT	Etiqueta usada para la entrada de un bloque de función para el que está prevista la etiqueta <i>* El valor no puede cambiarse en un componente de programa.</i>	✗	✗	○
VAR_OUTPUT	Etiqueta usada para la salida de un bloque de función para el que está prevista la etiqueta	✗	✗	○
VAR_IN_OUT	Etiqueta usada para la entrada y salida de un bloque de función para el que está prevista la etiqueta <i>* El valor puede cambiarse en un componente de programa.</i>	✗	✗	○

## 2.1.2

## Tipos de aplicaciones de etiquetas y valores guardados

Productividad

Calidad

### <Tipo de datos>

El tipo de datos refiere al tipo de valor almacenado en la etiqueta.

El tipo de datos asignado a una etiqueta indica el tipo y rango de valores que pueden almacenarse en la etiqueta y el dispositivo correspondiente.

Los tipos de datos que pueden utilizarse con programas en escalera se mencionan a continuación.

Tipo de dato	Descripción	Longitud en bits	Rango de valores almacenados
Bit	Se almacena ON (encendido) u OFF (apagado). Corresponde al dispositivo "M"	1 bit	1: ON (Encendido), 0: OFF (Apagado)
Palabra (con signo)	Se almacena un número entero sin fracciones.	16 bits	-32768 a 32767
Palabra doble (con signo)	Corresponde al dispositivo "D"	32 bits	-2147483648 a 2147483647
FLOAT (precisión simple)	Se almacena un número real, incluso fracciones.	16 bits	$-2^{128}$ a $-2^{-126}$ , 0, $2^{-126}$ a $2^{128}$
FLOAT (precisión doble)	Corresponde al dispositivo "D"	32 bits	$-2^{1024}$ a $-2^{-1022}$ , 0, $2^{-1022}$ a $2^{1024}$
Cadena	Se almacena una cadena de caracteres. Corresponde al dispositivo "D"	Variable	Hasta 255 caracteres
Temporizador	Se enciende cuando se alcanza el tiempo especificado. Corresponde al dispositivo temporizador "T"	-	-
Temporizador retentivo	Se enciende cuando se alcanza el tiempo especificado. Corresponde al dispositivo temporizador retentivo "ST"	-	-
Contador	Se enciende cuando se alcanza el conteo especificado. Corresponde al dispositivo contador "C"	-	-
Puntero	Se almacena una posición de inicio de subrutina. Corresponde al dispositivo puntero "P"	-	-



# 2.1.3 Registro de etiqueta



MELSOFT Series GX Works2 C:\Sequential Programs\learning\Robot control - [Local Label Setting MAIN [PRG] ]

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

Navigation

[PRG]write MAIN (194)Step \* Global Label Setting Global1 Local Label Setting MAIN [P...]

**Project**

- Parameter Prod line contr
- Intelligent Function Modu
- Global Device Comment
- Global Label
  - Global1
- Program Setting
- POU
  - Program
    - MAIN
      - Program
        - Local Label
- FB\_Pool
- Structured Data Type
- Local Device Commen

Device Memory

Device Initial Value

	Class	Label Name	Data Type	Constant
1	VAR	Production_qty_attained	Bit	...
2	VAR	Parts_input_start_flag	Bit	...
3	VAR	Start_reject_pct_calc	Bit	...
4				...
5				...
6				...
7				...
8				...
9				...
10				...
11				...
12				...
13				...
14				...
15				...
16				...
17				...
18				...
19				...
20				...
21				...
22				...
23				...
24				...
25				...
26				...

Finalice los ajustes para el registro de etiquetas.  
 Haga clic en para continuar.

## 2.1.4 Asignación automática de etiquetas a dispositivos



La conversión de un programa asigna automáticamente el dispositivo adecuado a la etiqueta según la clase y el tipo de datos. Cuando se use una etiqueta, no es necesario saber qué dispositivo es asignado.

Use **"Device/Label Automatic-Assign Setting" (Ajuste de asignación automática de dispositivo/etiqueta)** para cambiar el rango de dispositivos asignados a la etiqueta.

\* Esta pantalla es la ventana de ajuste de asignación automática de dispositivo para la serie MELSEC-Q y la serie MELSEC-L. La pantalla puede ser distinta para la serie MELSEC-F

**<Iniciar "Device/Label Automatic-Assign Setting" (Ajuste de asignación automática de dispositivo/etiqueta)>**  
 En el menú GX Works2, seleccione **[Tool]** (Herramienta) – **[Device/Label Automatic-Assign Setting]** (Ajuste de asignación automática de dispositivo/etiqueta).

**Device/Label Automatic-Assign Setting**

Set a device range to automatically assign to labels.  
 Labels will be assigned from its way down the displayed device list when multiple devices are selected.

	Device	Digit	Assign Selection	Assignment Range		Total Points	PLC Parameter Device Setting Range
				Start	End		
<b>Word Device</b>							
VAR Range	D	10	<input checked="" type="checkbox"/>	6144	12287	6144	0 -- 12287
	W	16	<input checked="" type="checkbox"/>				0 -- 1FFF
	R	10	<input type="checkbox"/>				
<b>VAR_RETAIN Range Latch(1)</b>							
	D Latch	10	<input type="checkbox"/>			0	
	W Latch	16	<input type="checkbox"/>				
	ZR Latch	10	<input type="checkbox"/>				
<b>Bit Device</b>							
VAR Range	M	10	<input checked="" type="checkbox"/>	4096	8191	4096	0 -- 8191
	B	16	<input type="checkbox"/>				0 -- 1FFF
<b>VAR_RETAIN Range Latch(1)</b>							
	L Latch	10	<input type="checkbox"/>			0	
	B Latch	16	<input type="checkbox"/>				
<b>Pointer</b>							
VAR Range	P	10	<input checked="" type="checkbox"/>	2048	4095	2048	2048 -- 4095
<b>Timer</b>							
VAR Range	T	10	<input checked="" type="checkbox"/>	64	2047	1984	0 -- 2047
<b>VAR_RETAIN Range Latch(1)</b>							
	T Latch	10	<input type="checkbox"/>			0	
<b>Retentive Timer</b>							
VAR Range	ST	10	<input type="checkbox"/>			0	
<b>VAR_RETAIN Range Latch(1)</b>							
	ST Latch	10	<input type="checkbox"/>			0	
<b>Counter</b>							
VAR Range	C	10	<input checked="" type="checkbox"/>	512	1023	512	0 -- 1023
<b>VAR_RETAIN Range Latch(1)</b>							
	C Latch	10	<input type="checkbox"/>			0	

Latch(1) : Able to clear the value by using a latch clear.  
 Latch(2) : Unable to clear the value by using a latch clear. Clearing will be executed by remote operation or program.

(Caution)  
 1. Label-nonassigned devices, of the automatically assigned ones while compiling, will be allotted the device that displayed at the lowest of the selected ones. Ex) Device will be assigned to ZR when D and ZR are selected.  
 2. Changing the assignment target device may also change the processing speed since the arithmetic processing speed for R and ZR is difference from other devices.

OK Cancel

**Robot control - [[PRG]Write MAIN 194 Step]**

Tool Window Help

- IC Memory Card
- Check Parameter...
- Options...
- Key Customize...
- Device/Label Automatic-Assign Setting...**
- Block Password...
- Confirm Memory Size...
- Set TEL Data/Connect via Modem
- LCPU Logging Configuration Tool...
- Ethernet Adapter Module Configuration Tool...
- Built-in I/O Module Tool
- Check Intelligent Function Module Parameter
- Intelligent Function Module Tool
- Language Selection...

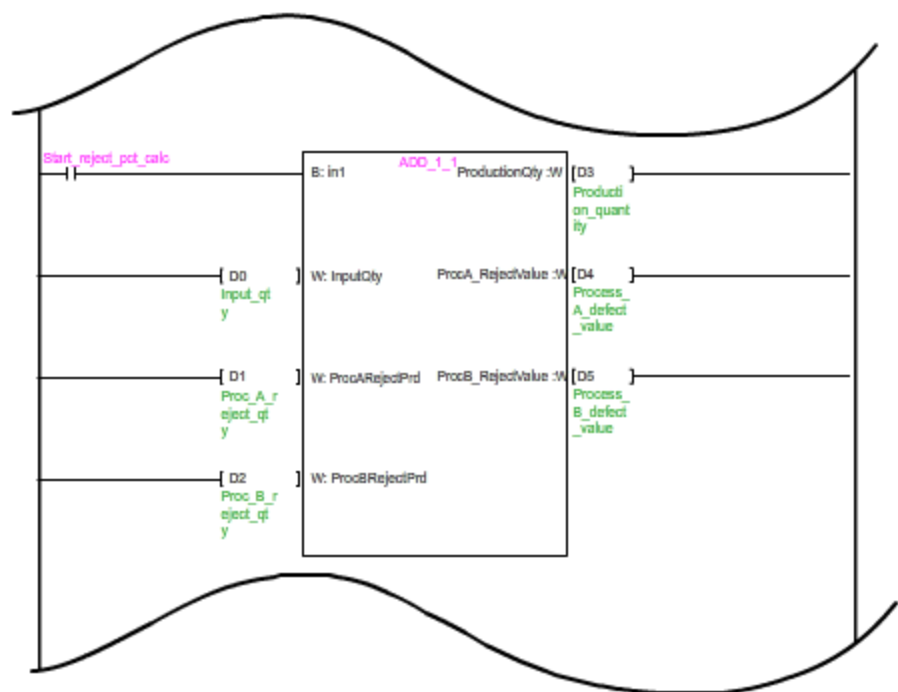


## 2.2 Disponer bloques en escalera de uso reiterado como bloques de función para desviación



En un programa a gran escala, algunos bloques en escalera pueden usarse repetidas veces. O, algunos de los mismos bloques en escalera pueden utilizarse en diferentes programas. No será posible mejorar la eficiencia del trabajo si usted debe introducir el mismo bloque en escalera cada vez que sea necesario. Además, si se identifica un defecto en un bloque en escalera común, se debe localizar y corregir cada bloque. Esto ocasiona una pérdida de tiempo significativa.

Use **"Bloque de Función (FB)"** para resolver estos problemas. Un bloque en escalera que se usa repetidas veces puede **disponerse como un bloque de función**, el cual puede usarse en todos los programas. Esto mejora enormemente la eficiencia del trabajo. Si se identifica un defecto en un bloque común, todo lo que necesita hacer es corregir el bloque de función. Esto no solo ahorra tiempo, sino que además evita el riesgo de no corregir uno de los bloques.



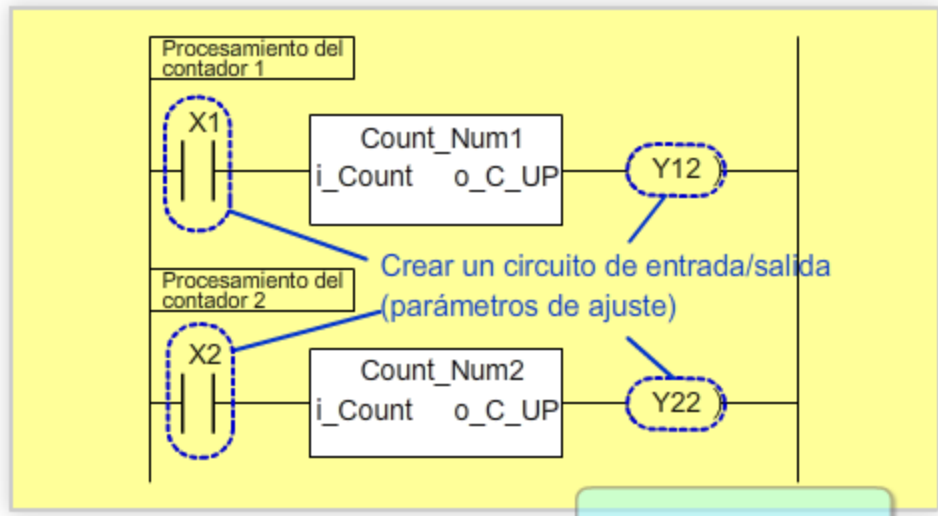
Pulse el botón para ver la animación desde el principio.



# 2.2.1 Crear y colocar bloques de función

Puesto que el bloque de función se utilizará en diferentes programas, no puede utilizarse dispositivos reales (como X, Y y D) en el bloque de función. Estos dispositivos deben ser reemplazados con **etiquetas**, tal y como aprendió en la Sección 2.1. El procedimiento para la disposición de bloques en escalera en un bloque de función se muestra a través de una animación.

1. Prepare un programa para su organización en un bloque de función.
2. Divida el programa en entrada y salida y reemplace los dispositivos internos por etiquetas internas para crear un bloque de función.
3. Coloque el bloque de función creado en el programa con la función arrastrar y soltar.
4. Asigne parámetros a las etiquetas de entrada y salida en el bloque de función que ha colocado en el programa.
5. Cree un circuito de entrada que envíe el parámetro a la etiqueta de entrada y un circuito de salida que reciba el parámetro desde la etiqueta de salida, antes y después del bloque de función.



Pulse el botón para regresar al primer flujo.

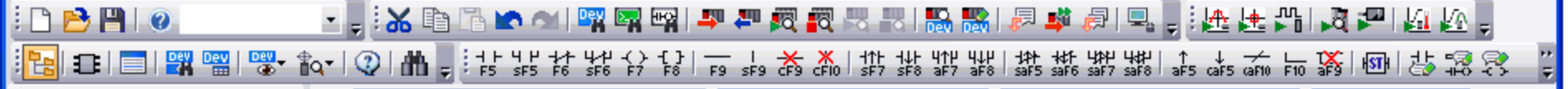


# 2.2.1 Crear y colocar bloques de función



MELSOFT Series GX Works2 C:\Sequential Programs\learning\Robot control - [[PRG]Write MAIN (188)Step \*

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



**Navigation**

[PRG]Write MAIN (188)Ste... Global Label Setting Global1 Local Label Setting MAIN [PRG] Function/FB Lat

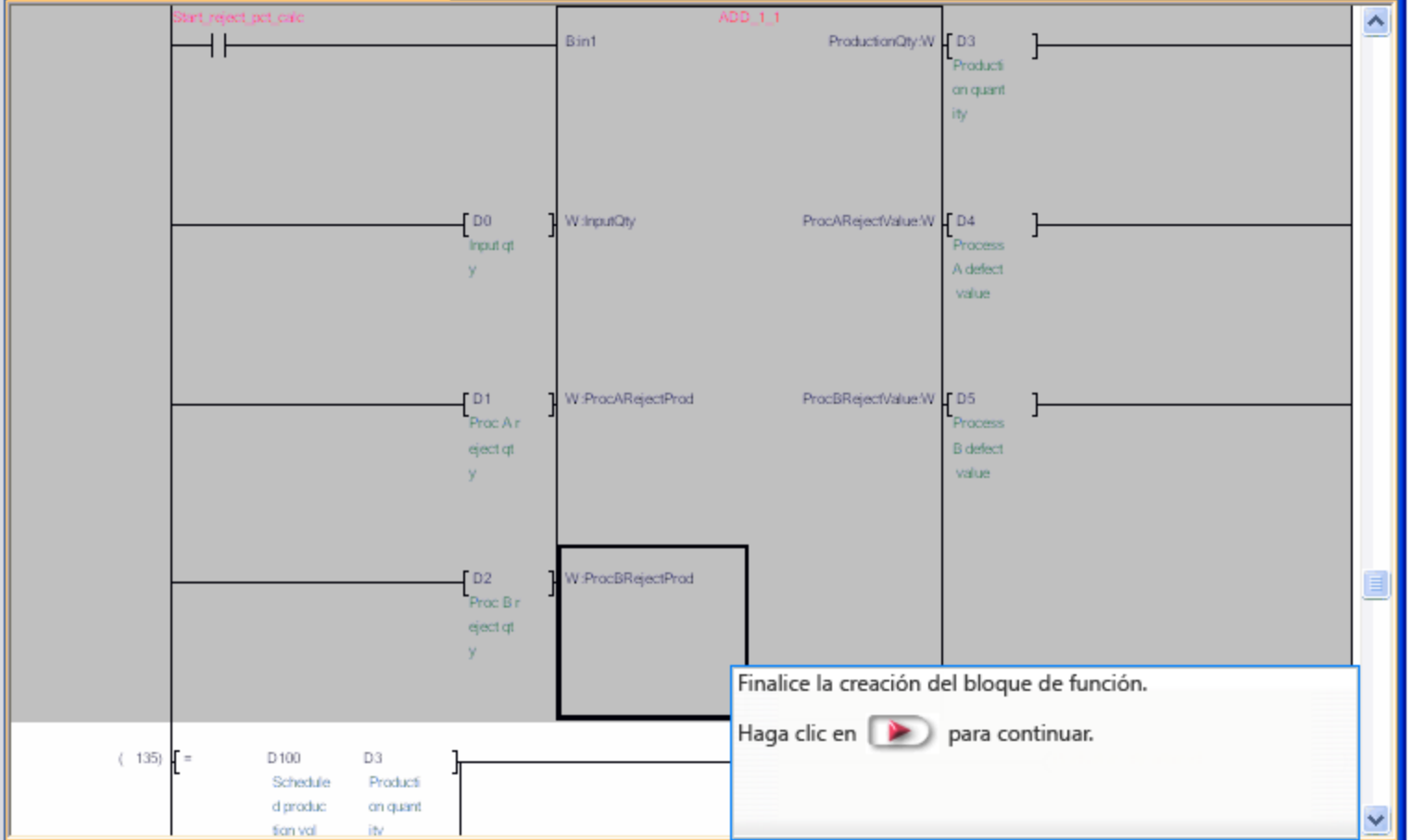
**Project**

- Parameter Prod line contr
- Intelligent Function Modu
- Global Device Comment
- Global Label
  - Global1
- Program Setting
- POU
  - Program
    - MAIN
      - Program
      - Local Label
  - FB\_Pool
    - ADD\_1
      - Program
      - Local Label
- Structured Data Type

**Project**

**User Library**

**Connection Destination**



Finalice la creación del bloque de función.  
 Haga clic en para continuar.



## 2.2.2 Usar la biblioteca de bloques de función

Productividad

Calidad

Ahora sabe cómo crear un bloque de función.

Mitsubishi Electric ha dispuesto los programas de control de diversos módulos como CPU, módulos de entrada/salida analógica, redes y módulos de posicionamiento como bloques de función, que están disponibles, sin costo alguno, como la **"Biblioteca FB"**.

Usar la biblioteca FB elimina la necesidad de desarrollar programas de control de módulos, que solía ser responsabilidad de los usuarios. Incluso los módulos con los que no esté familiarizado pueden introducirse fácilmente.

La biblioteca FB puede descargarse de la página **"MELSOFT Library Download" (Descargar Biblioteca MELSOFT)** del **MITSUBISHI ELECTRIC FA Website**.

<Antes>



<Si usa la biblioteca FB>



\* No se ha provisto una biblioteca FB para la serie MELSEC-F.

Los modelos compatibles se mencionan en la página de descarga. Verifique el modelo compatible antes de realizar la descarga.

## 2.3 Cambiar los valores iniciales del dispositivo sin corregir el programa

Productividad

Calidad

Por lo general, el valor inicial o constante de un dispositivo se ajusta usando la instrucción MOV antes del procesamiento del programa principal.

En este caso, debe corregirse directamente el programa cada vez que cambie la operación del programa según la aplicación del sistema. Este método no solo lleva mucho tiempo, sino que además existe el riesgo de errores de corrección o de que no se haga una corrección.

Use **"Valor inicial de dispositivo"** para resolver estos problemas.

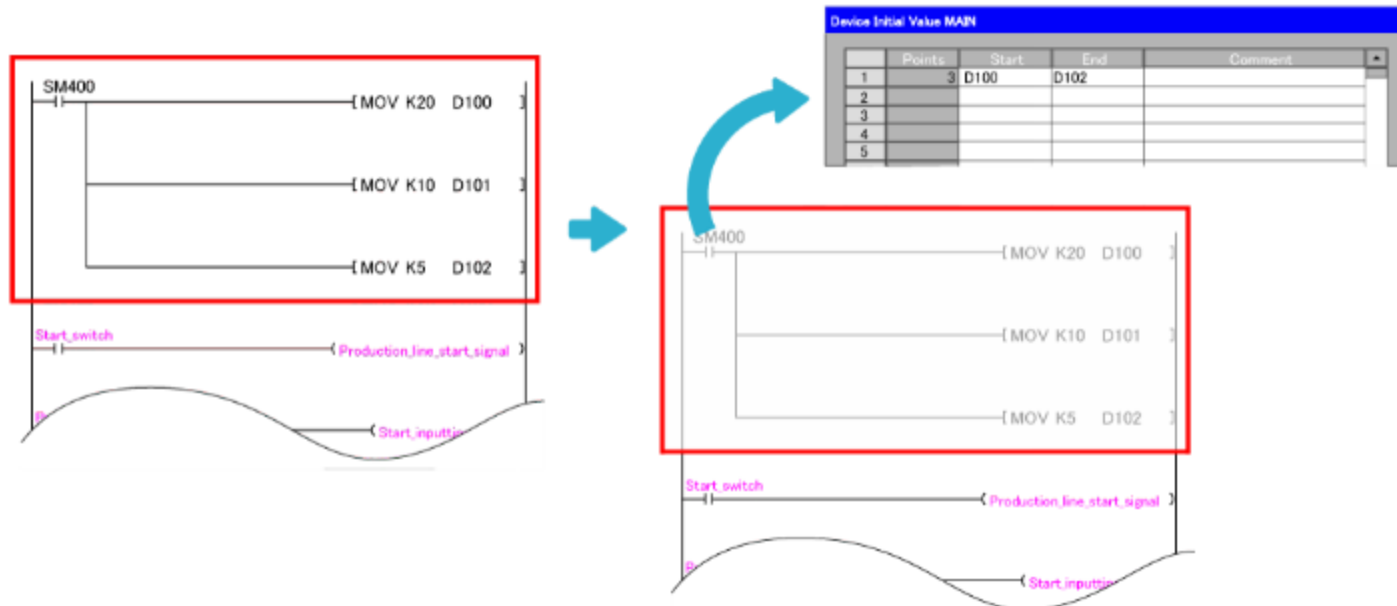
El uso de la función GX Works2 garantiza el manejo adecuado de los valores iniciales del dispositivo y elimina la necesidad de hacer correcciones al programa, permitiéndole así crear programas con mucha más eficiencia.

Además, esta función elimina la necesidad de un programa de ajuste de valores iniciales de dispositivo, lo que reduciría el volumen del programa (uso de memoria) y disminuiría el riesgo de fallo.

Para el valor inicial de dispositivo, especifique el rango de dispositivos para ajustar los valores iniciales.

Los valores iniciales reales se almacenan en la **memoria de dispositivo** y se asignan al rango de dispositivos especificado.

Con áreas de memoria de dispositivo preparadas según las aplicaciones del sistema, es posible cambiar los valores iniciales del dispositivo con solo cambiar el área de memoria de dispositivo a ser asignada.



# 2.3 Cambiar los valores iniciales del dispositivo sin corregir el programa



MELSOFT Series GX Works2 C:\Sequential Programs\learning\Robot control - [[PRG]Write MAIN (166)Step \*]

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

Navigation [PRG]Write MAIN (166)Ste... Global Label Setting Global1 Local Label Setting MAIN [PRG]

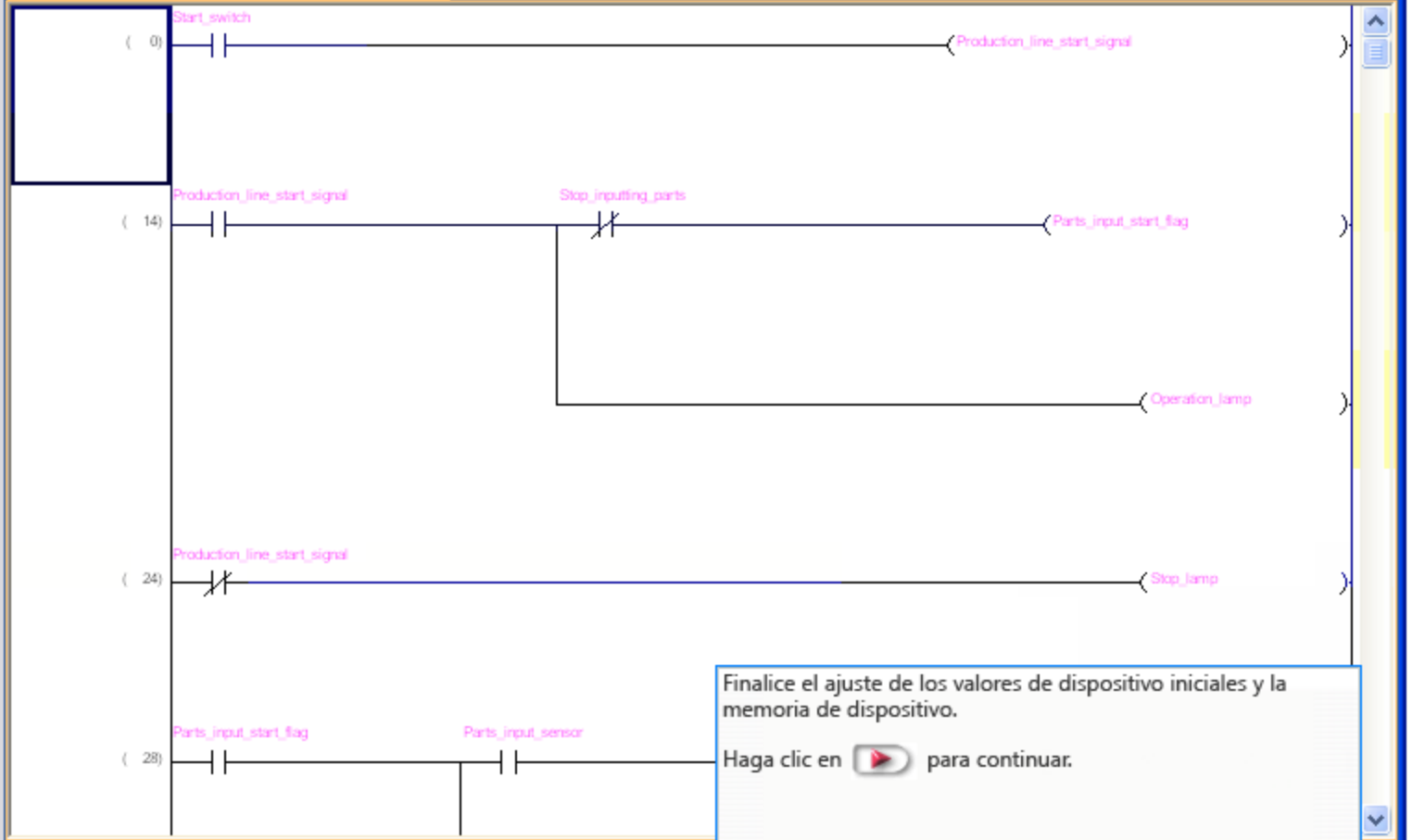
**Project**

- Global1
  - Program Setting
  - POU
    - Program
      - MAIN
        - Program
        - Local Label
      - FB\_Pool
      - Structured Data Type
      - Local Device Comment
    - Device Memory
      - MAIN
      - ProdQty1
    - Device Initial Value
      - MAIN

**Project**

**User Library**

**Connection Destination**



Finalice el ajuste de los valores de dispositivo iniciales y la memoria de dispositivo.  
 Haga clic en para continuar.

Navigation icons: Home, Play, Stop, Previous, Next, and a slider.

## 2.4

## Simplificar los programas en escalera

Productividad

Calidad

En un programa en escalera a gran escala, la conexión de dispositivos, instrucciones y líneas de regla puede volverse bastante complicada, haciendo difícil captar qué procesamiento realiza el programa.

En particular, los cálculos numéricos, como una fórmula completada en una sola línea, deben programarse usando una combinación de instrucciones.

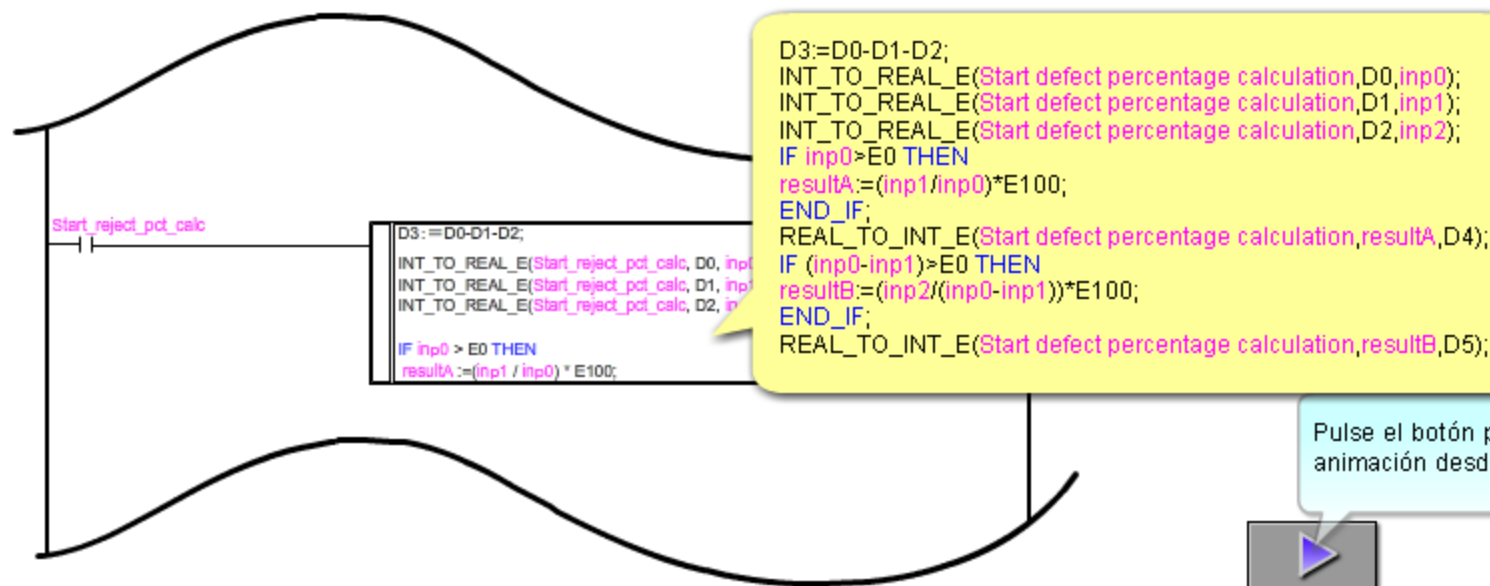
Use **"Inline Structured Text" (Texto estructurado en línea)** para resolver estos problemas.

Un programa en escalera se sustituye con un programa escrito parcialmente en **lenguaje de texto estructurado (ST)**.

El lenguaje ST es un lenguaje de programación de control secuencial similar al lenguaje C usado para la programación del software de computadora.

Los cálculos numéricos pueden escribirse usando fórmulas, de modo que incluso los programadores que no estén familiarizados con el lenguaje C pueden usar el texto estructurado en línea.

La siguiente figura muestra un ejemplo en el que parte del programa en escalera para el sistema es sustituido por texto estructurado en línea. Puede ver que el programa que antes era complicado ahora es fácil de entender.





# 2.4 Simplificar los programas en escalera



MELSOFT Series GX Works2 C:\Sequential Programs\learning\Robot control - [[PRG]Write MAIN 114 Step]

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

Navigation [PRG]Write MAIN 114 Step Global Label Setting Global Local Label Setting MAIN [PRG]

Project

- Parameter Prod line contr
- Intelligent Function Modu
- Global Device Comment
- Global Label
- Global1
- Program Setting
- POU
  - Program
    - MAIN
      - Program
        - Local Label
  - FB\_Pool
  - Structured Data Type
  - Local Device Commen
- Device Memory
- Device Initial Value

Project

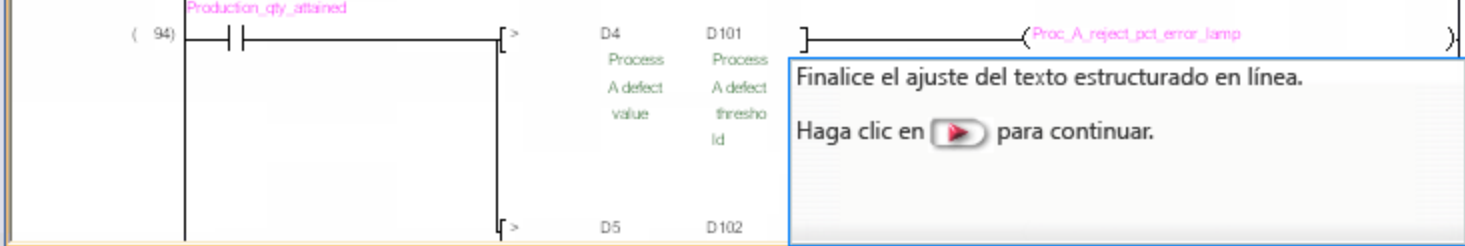
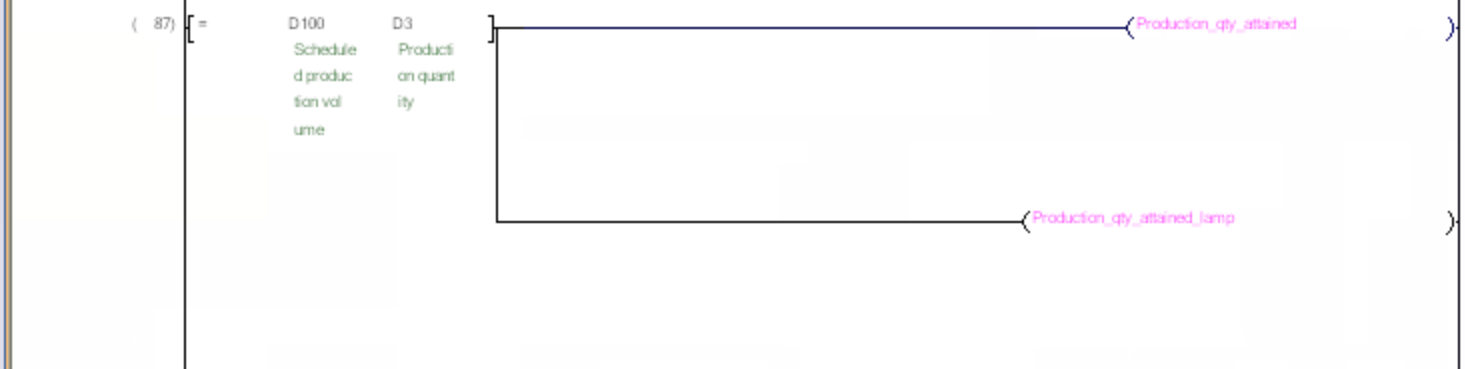
User Library

Connection Destination

```

( 75) Start reject pct calc
-----
INT_TO_REAL_E(Start_reject_pct_calc,D2,inp2);
IF inp0>E0 THEN
resultA:=(inp1/inp0)*E100;
END_IF;
REAL_TO_INT_E(Start_reject_pct_calc,resultA,D4);
IF (inp0-inp1)>E0 THEN
resultB:=(inp2/(inp0-inp1))*E100;
END_IF;
REAL_TO_INT_E(Start_reject_pct_calc,resultB,D5);

```



Navigation icons: back, forward, home, search, etc.



## 2.5 Crear programas que sean fáciles de entender y leer



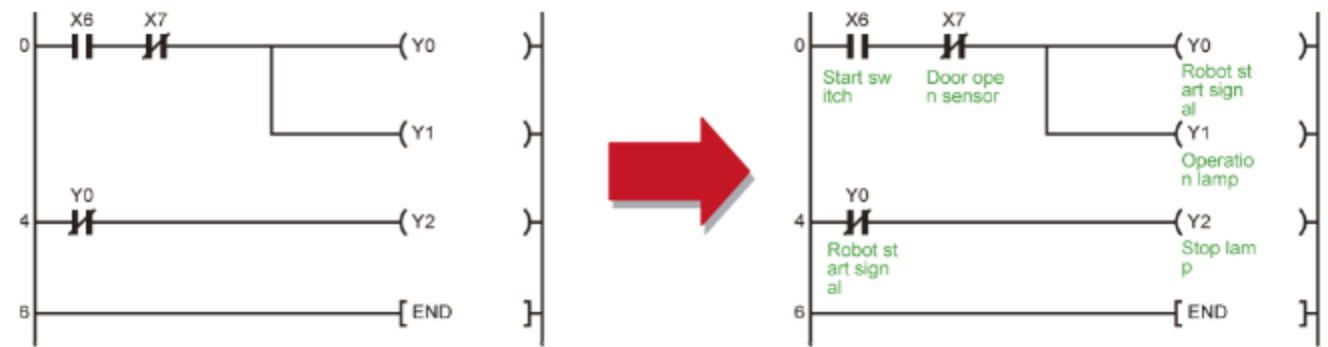
Es posible que le resulte difícil entender los detalles de control de un programa a gran escala con solo revisar el programa. Los siguientes problemas pueden ocurrir como consecuencia de lo siguiente:

- Comete errores de programa (como ingresar dispositivos o instrucciones incorrectas).
- Usted no puede encontrar las causas de los errores del programa.
- Alguien que se hace cargo de la programación no puede entender los detalles de control.

Use **"Comentario"** para resolver problemas.

Algunas notas, como por ejemplo información de control y nombres de dispositivos, pueden adjuntarse al programa, de modo que los detalles de control puedan entenderse fácilmente.

Estos comentarios deben ingresarse donde sea posible para crear programas que sean fáciles de entender no solo para usted, sino también para los demás.



GX Works2 permite ingresar los siguientes comentarios.

Tipo de comentario	Rango de comentario
Comentario de dispositivo	Puede adjuntarse un comentario a un dispositivo. Este comentario indica la aplicación de cada dispositivo y el tipo de dispositivo de E/S conectado.
Declaración	Puede adjuntarse un comentario a un bloque en escalera. Este comentario hace que el flujo de procesamiento sea fácil de entender.
Nota	Puede adjuntarse un comentario a una bobina/instrucción de aplicación en el programa. Este comentario hace que el contenido de una bobina (salida) e instrucción de aplicación sea fácil de entender.

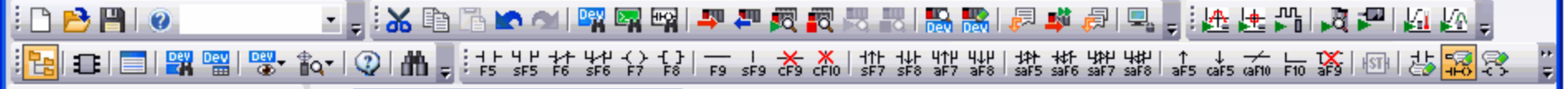
# 2.5.1

## Escribir un comentario para cada bloque en escalera



MELSOFT Series GX Works2 C:\Sequential Programs\learning\Robot control - [[PRG]Write MAIN 145 Step]

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



**Navigation**

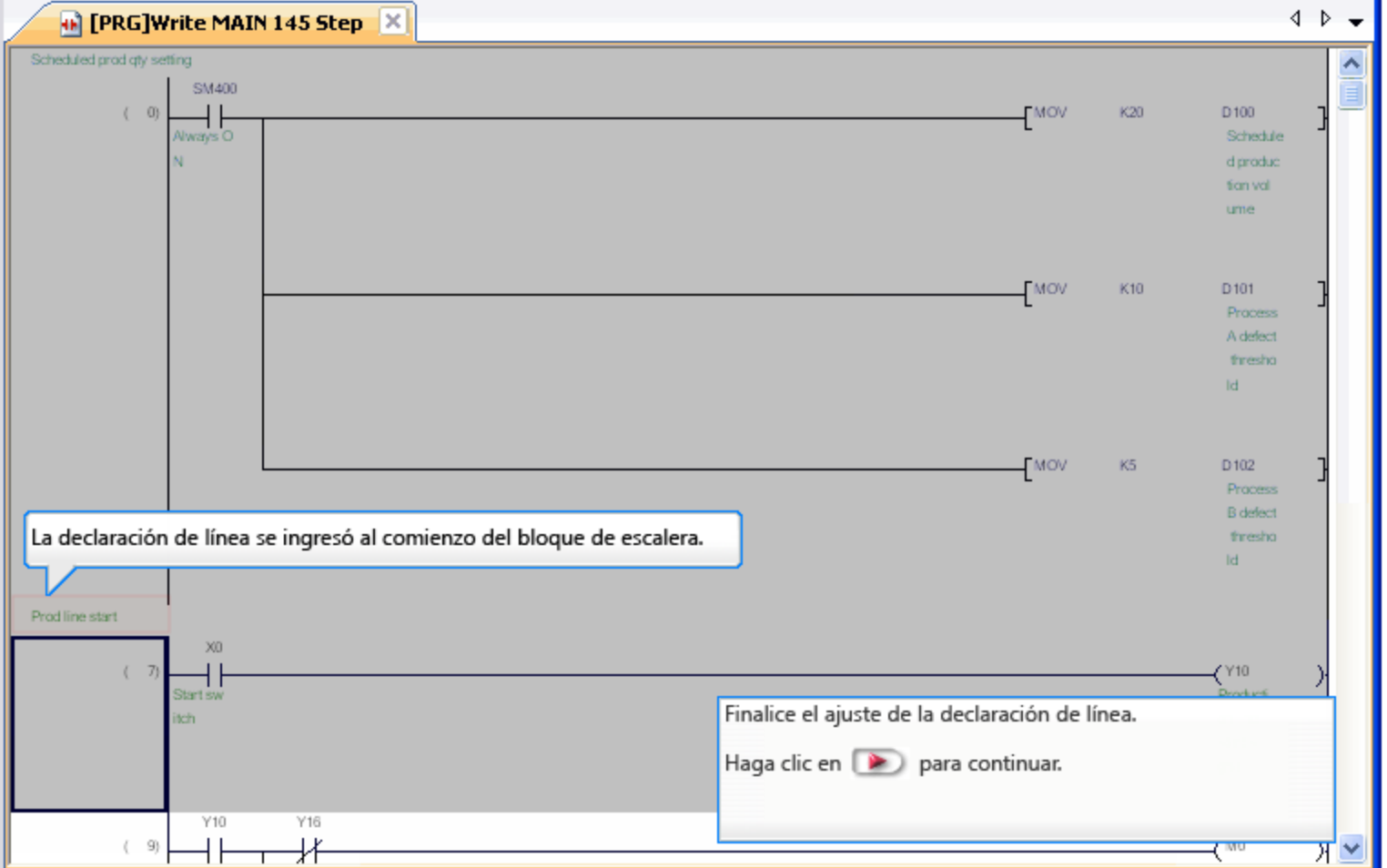
**Project**

- Parameter Prod line control s
- Intelligent Function Module
- Global Device Comment
- Global Label
- Program Setting
- POU
  - Program
    - MAIN
      - Program
      - Local Label
  - FB\_Pool
  - Structured Data Types
  - Local Device Comment
- Device Memory
- Device Initial Value

**Project**

**User Library**

**Connection Destination**



La declaración de línea se ingresó al comienzo del bloque de escalera.

Finalice el ajuste de la declaración de línea.  
Haga clic en para continuar.



# 2.5.2 Escribir un comentario para cada salida (bobina, instrucción de aplicación)

Productividad Calidad

MELSOFT Series GX Works2 C:\Sequential Programs\learning\Robot control - [[PRG]Write MAIN 181 Step]

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

File Edit View Tools Windows Help

Navigation icons: Undo, Redo, Copy, Paste, etc.

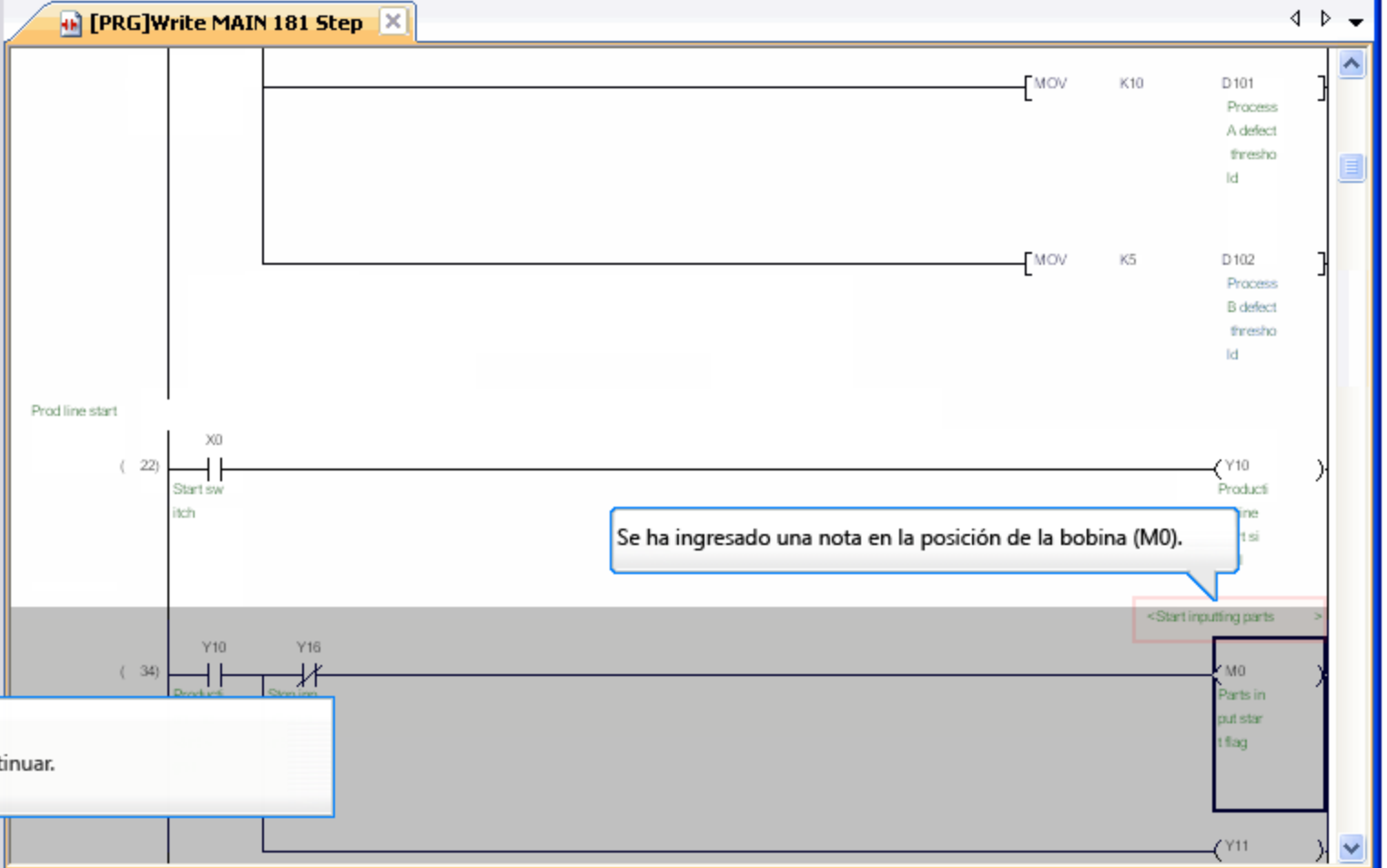
Navigation

Project

- Parameter Prod line control s
- Intelligent Function Module
- Global Device Comment
- Global Label
- Program Setting
- POU
  - Program
    - MAIN
      - Program
        - Local Label
- FB\_Pool
- Structured Data Types
- Local Device Comment

Device Memory

Device Initial Value



Se ha ingresado una nota en la posición de la bobina (M0).

Finalice el ajuste de la nota.  
Haga clic en para continuar.

Navigation icons: Home, Play, Stop, etc.

## 2.6 Hacer fácil la lectura de programas que contengan relés/registros especiales

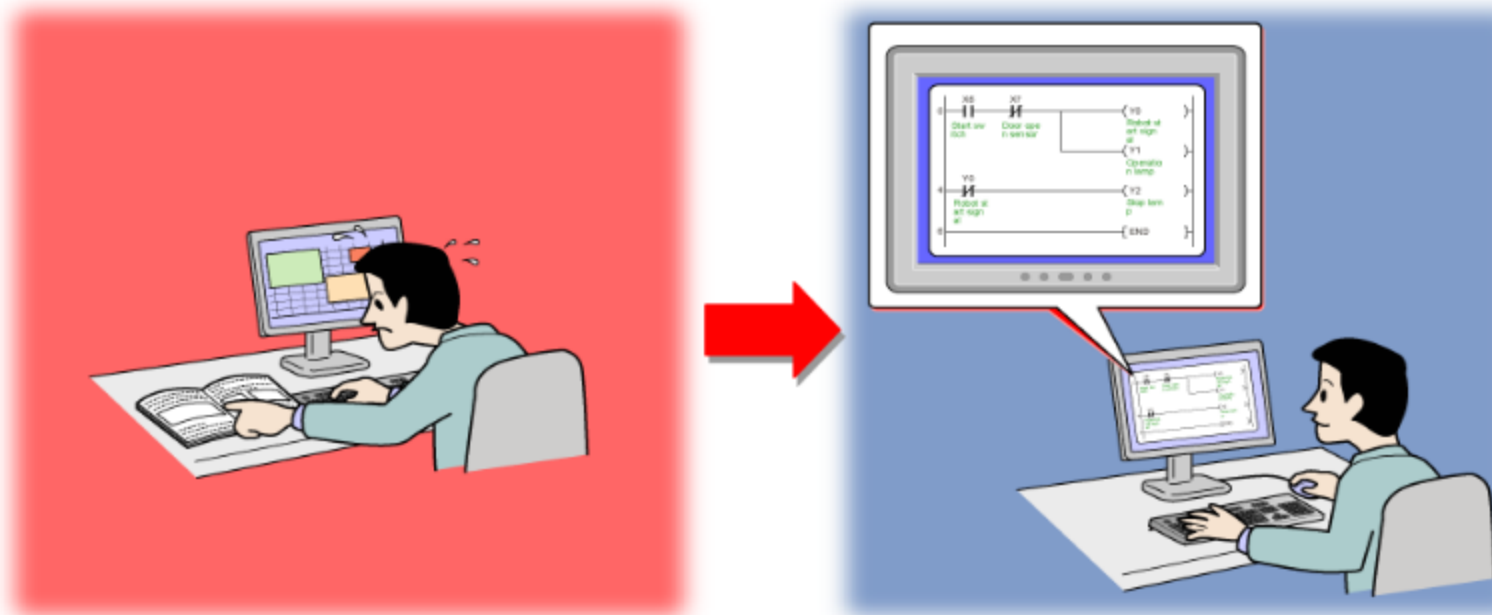
Si en un programa se utilizan relés especiales, registros especiales y/o dispositivos de módulo de función inteligente, podría ser difícil entender todas las aplicaciones y funciones de estos dispositivos. Usted deberá leer el programa con el manual en mano. Aunque el programa sería más fácil de leer si se adjuntara un comentario a cada dispositivo, se necesitaría invertir mucho tiempo y esfuerzo para adjuntar comentarios si se usaran varios dispositivos.

Use **"Comentario de muestra"** para resolver estos problemas.

GX Works2 brinda comentarios de muestra que describen las aplicaciones y funciones de relés especiales, registros especiales y dispositivos de módulo de función inteligente.

El uso de estos comentarios de muestra hace mucho más fácil adjuntar comentarios a los dispositivos para hacer que el programa sea fácil de leer.

Los comentarios de muestra pueden modificarse según sea necesario.



# 2.6 Hacer fácil la lectura de programas que contengan relés/registros especiales



MELSOFT Series GX Works2 C:\Sequential Programs\learning\Robot control - [[PRG]Write MAIN 194 Step]

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

Navigation icons: File, Edit, View, Run, Stop, etc.

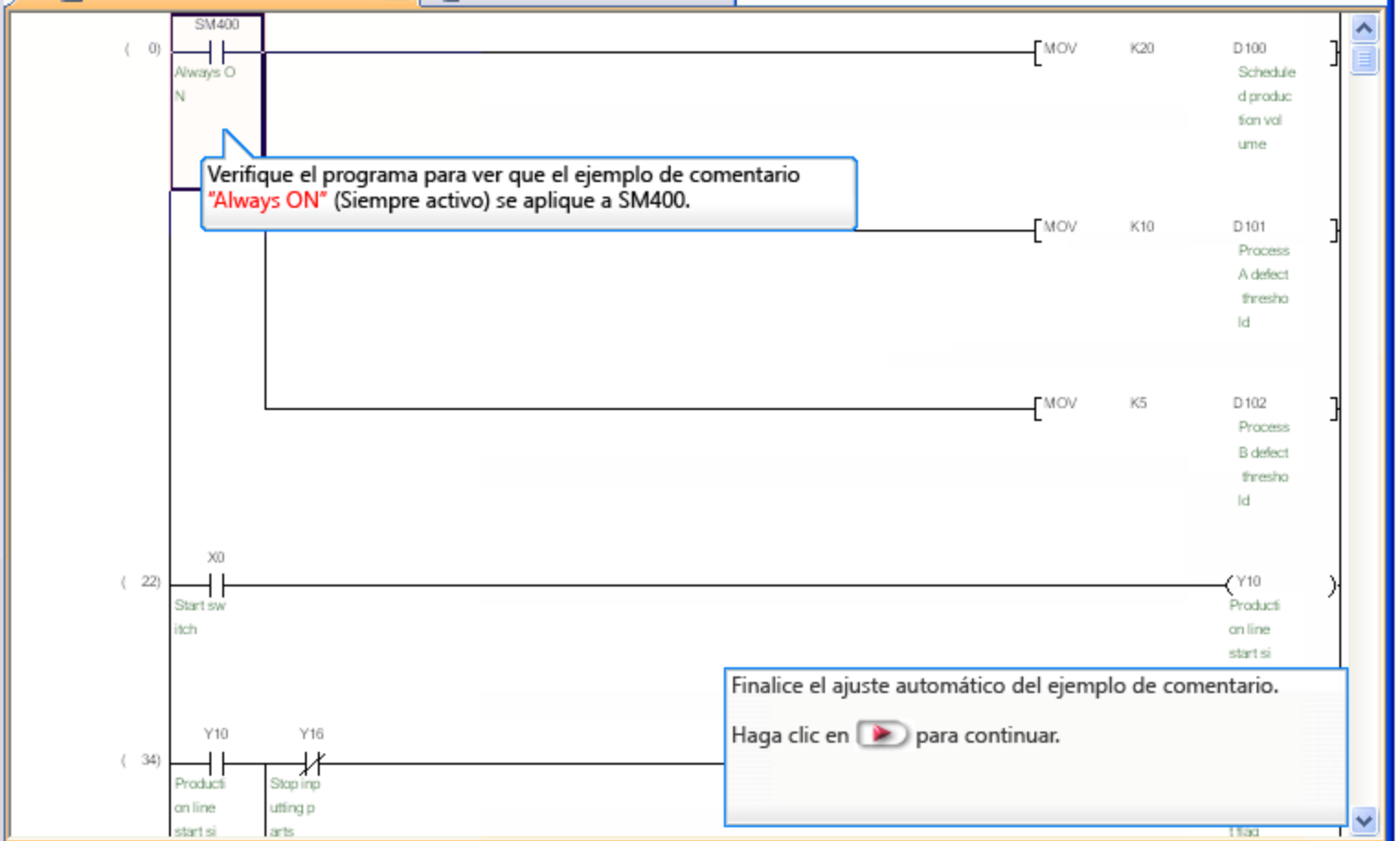
Navigation

[PRG]Write MAIN 194 Step

Device Comment COMMENT

Project

- Parameter Prod line control s
- Intelligent Function Module
- Global Device Comment
- Global Label
- Program Setting
- POU
  - Program
    - MAIN
      - Program
      - Local Label
  - FB\_Pool
  - Structured Data Types
  - Local Device Comment
- Device Memory
- Device Initial Value



Verifique el programa para ver que el ejemplo de comentario "Always ON" (Siempre activo) se aplique a SM400.

Finalice el ajuste automático del ejemplo de comentario. Haga clic en para continuar.

Navigation icons: back, forward, home, stop

## Capítulo 3 Depuración

### Pasos para el aprendizaje en el Capítulo 3

En el capítulo 3, aprenderá sobre las funciones usadas para la depuración. GX Works2 brinda muchas herramientas distintas de monitorización y depuración para corregir errores (bugs). Cree programas de calidad, libres de errores, usando estas herramientas de monitorización y depuración.

- 3.1 Monitorizar solo dispositivos y etiquetas objetivo
- 3.2 Verificar el estado de uso de dispositivos y etiquetas
- 3.3 Recopilar información sobre el cambio de valor de dispositivo en el tiempo
- 3.4 Cambiar los valores del dispositivo sin corregir el programa
- 3.5 Depurar la operación del programa paso a paso
  - 3.5.1 Usar la función de ejecución escalonada
- 3.6 Simular la operación de un dispositivo externo
  - 3.6.1 Ingresar los valores del dispositivo para ajustar el sistema de E/S
  - 3.6.2 Usar el formato de diagrama de sincronización para ajustar el sistema de E/S

## 3.1

## Monitorizar solo dispositivos y etiquetas objetivo

Calidad

Un programa usa muchas instrucciones y dispositivos.

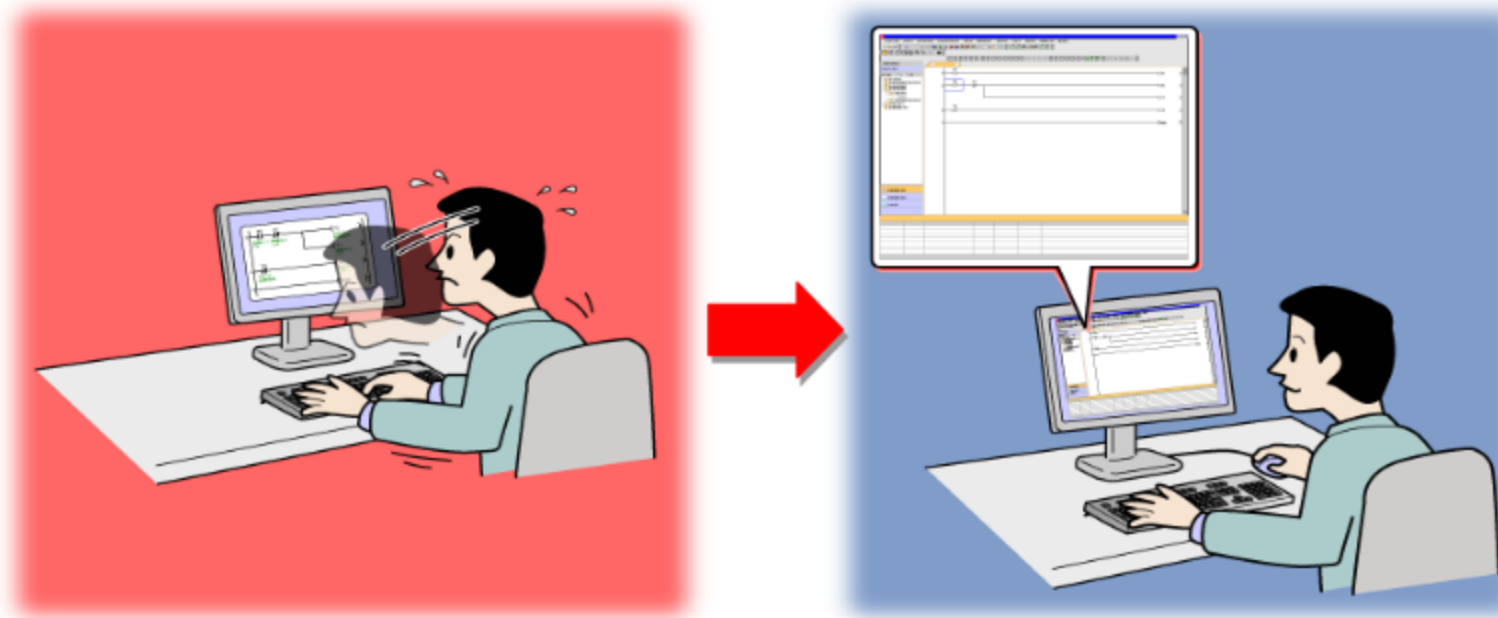
Además, la longitud vertical de un programa significa que solo parte de éste se visualiza por vez en la pantalla del monitor de la PC.

Por ello, el monitor en escalera no puede supervisar solo toda la operación.

Use "**Observación**" para resolver estos problemas.

Esta función puede utilizarse para supervisar solo los dispositivos y etiquetas que usted haya especificado con anticipación.

Es posible crear dos o más Observaciones para garantizar la monitorización de cada rango.



# 3.1 Monitorizar solo dispositivos y etiquetas objetivo

MELSOFT Series GX Works2 C:\Sequential Programs\learning\Robot control - [[PRG]Monitor Executing MAIN (Read Only) 194 Step

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

Navigation [PRG]Monitor Executing M...

Project

- Parameter Prod line contr
- Intelligent Function Modu
- Global Device Comment
- Global Label
- Program Setting
- POU
- Program
  - MAIN
  - Program

Prod qty calc

( 74) M1

Start re ject pct calc

D1 D2 D6

GX Simulator 2

Tool Options

Switch

RESET  RUN


LED

- MODE
- RUN
- ERR.
- USER

Watch 1

Se supervisa los valores actuales de los dispositivos registrados con Watch 1 (Observación 1).

Device/Label	Current Value	Data Type	Class	Device	Comment
X0	0	Bit		X0	Start switch
X1	0	Bit		X1	Parts input sensor
X2	0	Bit		X2	
X3	0	Bit		X3	

Finalice el registro y monitorización de Watch.  
Haga clic en  para continuar.

English Simple Q03UDE Simulation (84 N...



## 3.2

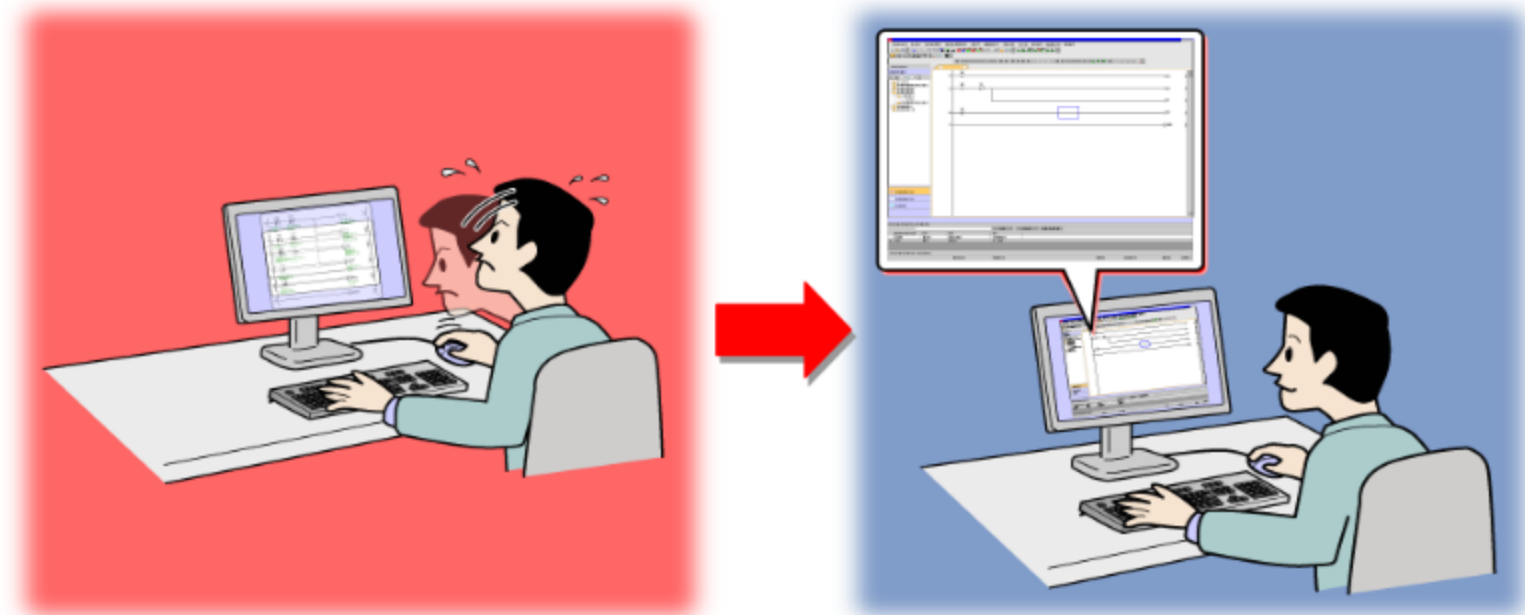
## Verificar el estado de uso de dispositivos y etiquetas

Calidad

Un programa usa los mismos dispositivos y etiquetas en distintas ubicaciones. Probablemente usted desee verificar el estado de uso de estos dispositivos y etiquetas comparándolos entre ubicaciones.

Use **"Referencia cruzada"** para hacerlo.

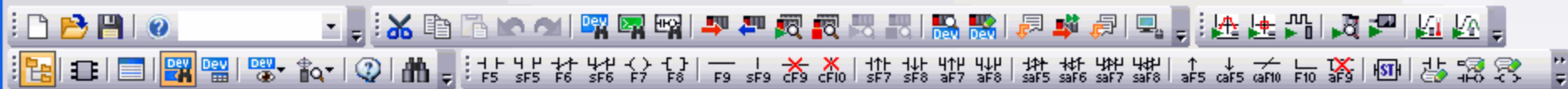
Esta función menciona las ubicaciones de los dispositivos y etiquetas que coinciden con las condiciones de búsqueda, de manera que pueda compararlas y verificar si hay un uso incorrecto.



# 3.2 Verificar el estado de uso de dispositivos y etiquetas

MELSOFT Series GX Works2 C:\Sequential Programs\learning\Robot control - [[PRG]Write MAIN 194 Step]

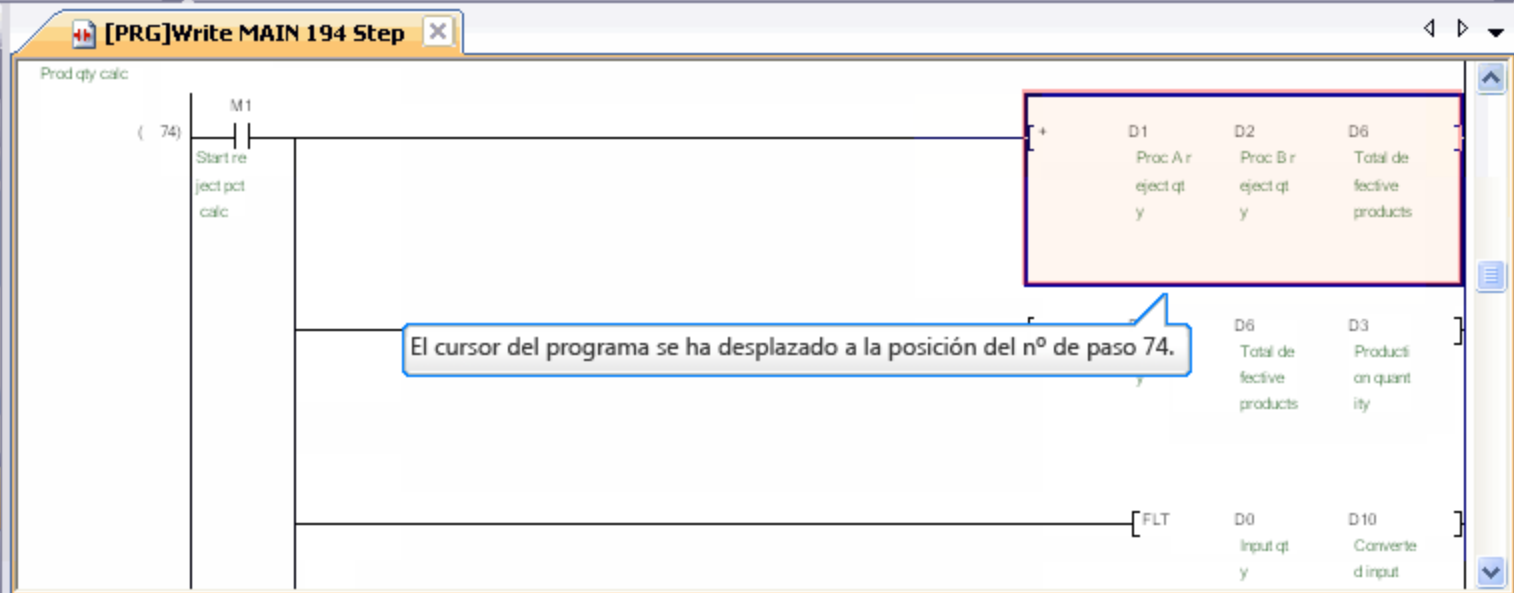
Project Edit Find/Replace Compile View Online Debug Diagnostics Tool Window Help



**Navigation**

**Project**

- Parameter Prod line contr
- Intelligent Function Modu
- Global Device Comment
- Global Label
- Program Setting
- POU
- Program
  - MAIN



El cursor del programa se ha desplazado a la posición del nº de paso 74.

**Cross Reference**

Cross Reference Information Condition Setting

Device/Label D1 Find Print... Print Preview...

Device/Label	Device	Instruction	Ladder Symbol	Position	Data Name
Filtering Condition	Filtering Con...	Filtering Condit...		Filtering Condition	
D1	D1	+P	-[ ]-	Step No.63	
D1	D1	+	-[ ]-	Step No.84	
D1	D1	FLT	-[ ]-	Step No.92	

5: device/cross reference information of label "D1"

Finalice el uso de la Referencia cruzada.

Haga clic en para continuar.



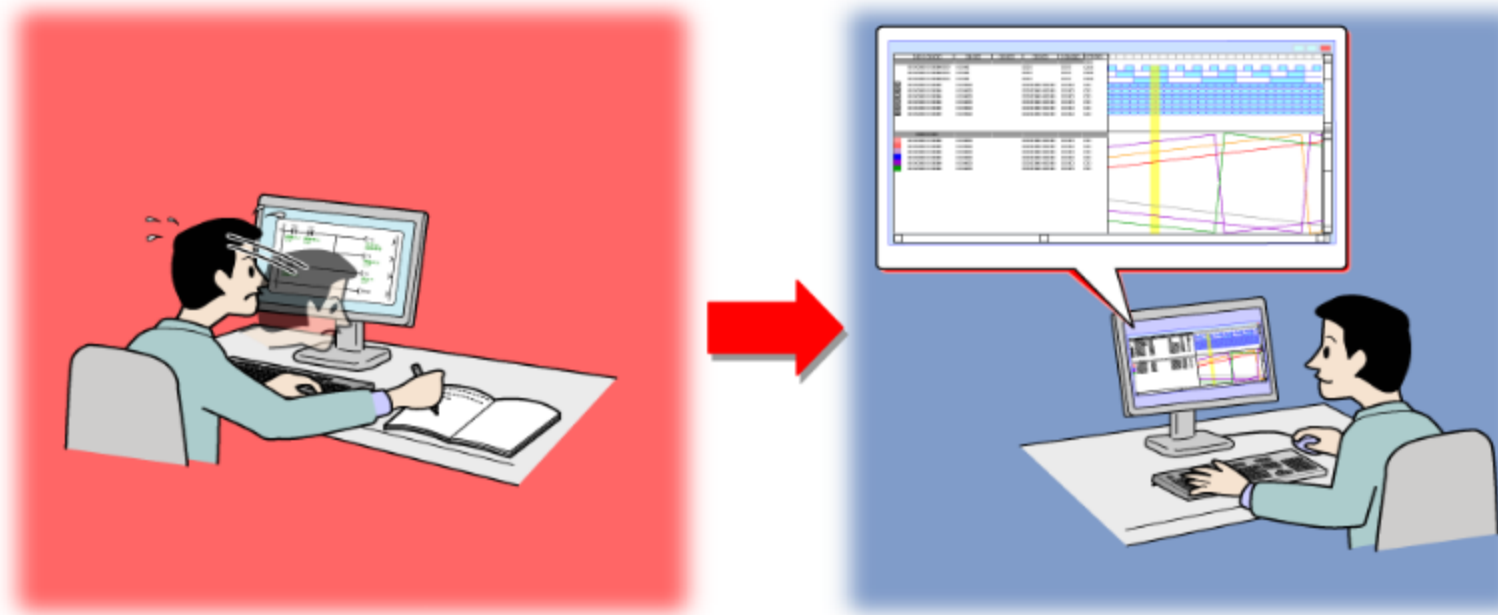
### 3.3 Recopilar información sobre el cambio de valor de dispositivo en el tiempo

Calidad

Tal vez desee confirmar que los cambios en los valores de etiqueta y dispositivo se encuentran dentro del rango de diseño, o proceder a una resolución de errores verificando los cambios que se hayan producido en caso de una falla.

Use **"Seguimiento de muestreo"** en este caso. (Solo para la serie MELSEC-Q y la serie MELSEC-L)

Esta función puede utilizarse para supervisar y registrar los cambios en los valores de etiqueta y dispositivo en el tiempo. El registro puede guardarse en los archivos.



# 3.3 Recopilar información sobre el cambio de valor de dispositivo en el tiempo



Navigation

Project

- Parameter Prod line contr
- Intelligent Function Modu
- Global Device Comment
- Global Label
- Program Setting
- POU
  - Program
    - MAIN
      - Program
        - Local Label
- FB\_Pool
- Structured Data Type
- Local Device Commen

- Device Memory

Project

User Library

Connection Destination

[PRG]Monitor Executing MAIN (Rea...)

Sampling Trace

Device/Label	Device	Comment	Data Type
M0	M0	Parts input st	Bit
M1	M1	Start reject pc	Bit
M2	M2	Production qt	Bit
<input checked="" type="checkbox"/> D3	D3	Production qu	Word(Signed)
<input checked="" type="checkbox"/> D4	D4	Process A de	Word(Signed)
<input checked="" type="checkbox"/> D5	D5	Process B de	Word(Signed)
X0	X0	Start switch	Bit
X1	X1	Parts input se	Bit
X2	X2	Proc A defect	Bit
X3	X3	Proc B defect	Bit

Timeline view showing bit signals (M0, M1, M2, X0, X1, X2, X3) and word signals (D3, D4, D5) over time. A vertical yellow bar highlights a specific time point between 240 and 250.

Trend Graph

	D3	D3	Production qu	Word(Signed)	D 1
	D4	D4	Process A de	Word(Signed)	D 1
	D5	D5	Process B de	Word(Signed)	D 1

Graph showing the trend of word signals D3, D4, and D5 over time. The highlighted time point shows a change in the values of these signals.

Finalice el ajuste del seguimiento de muestreo.  
 Haga clic en para continuar.

## 3.4 Cambiar los valores del dispositivo sin corregir el programa

Calidad

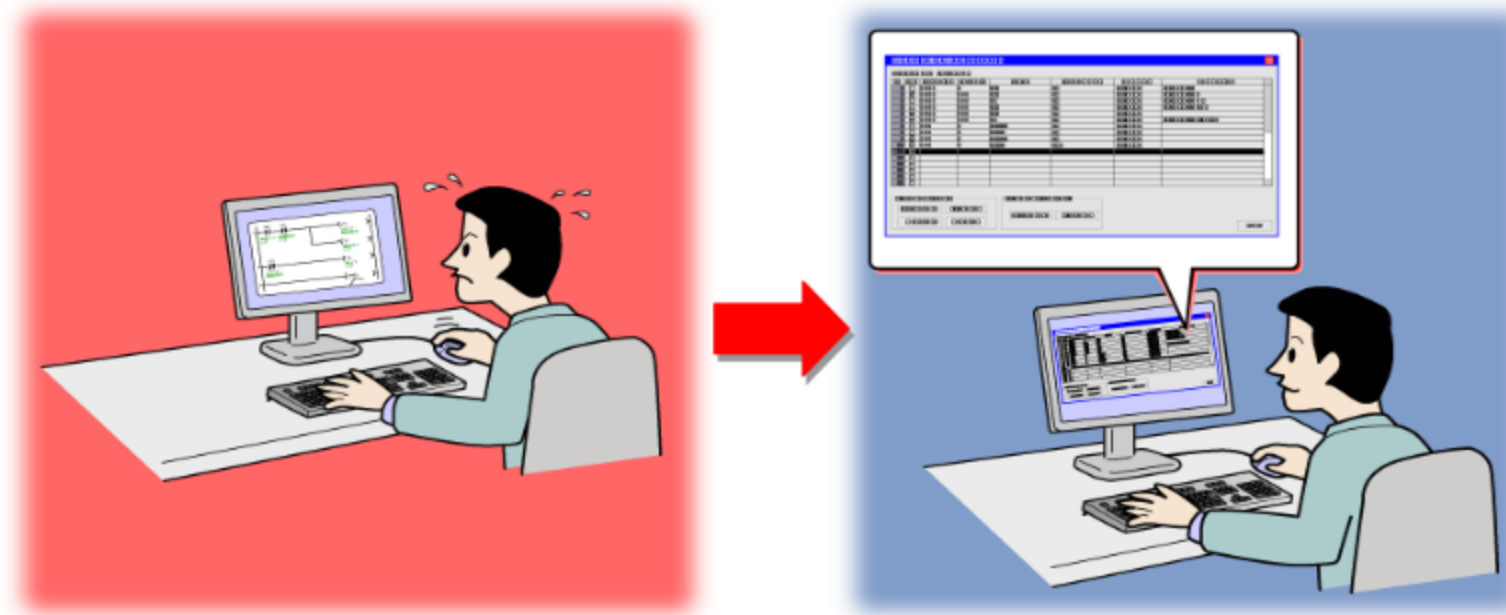
Cuando realice una depuración, es posible que desee cambiar forzosamente los valores del dispositivo para verificar la diferencia en la operación del programa.

Sin embargo, esto significa que debe modificar el programa cada vez que se cambie el valor de un dispositivo, lo que exige considerable tiempo y esfuerzo.

Además, si olvida restaurar los ajustes originales del programa modificado, ello podría causar otra falla.

Use **"Prueba de dispositivo de ejecución condicionada"** en este caso. (Solo para la serie MELSEC-Q y la serie MELSEC-L)

Esta función cambia los valores del dispositivo al ejecutarse el número de paso especificado por anticipado, sin que usted tenga que modificar el programa.



# 3.4 Cambiar los valores del dispositivo sin corregir el programa



MELSOFT Series GX Works2 C:\Sequential Programs\le-learning\Robot control - [[PRG]Monitor Executing MAIN (Read Only) 194 Step

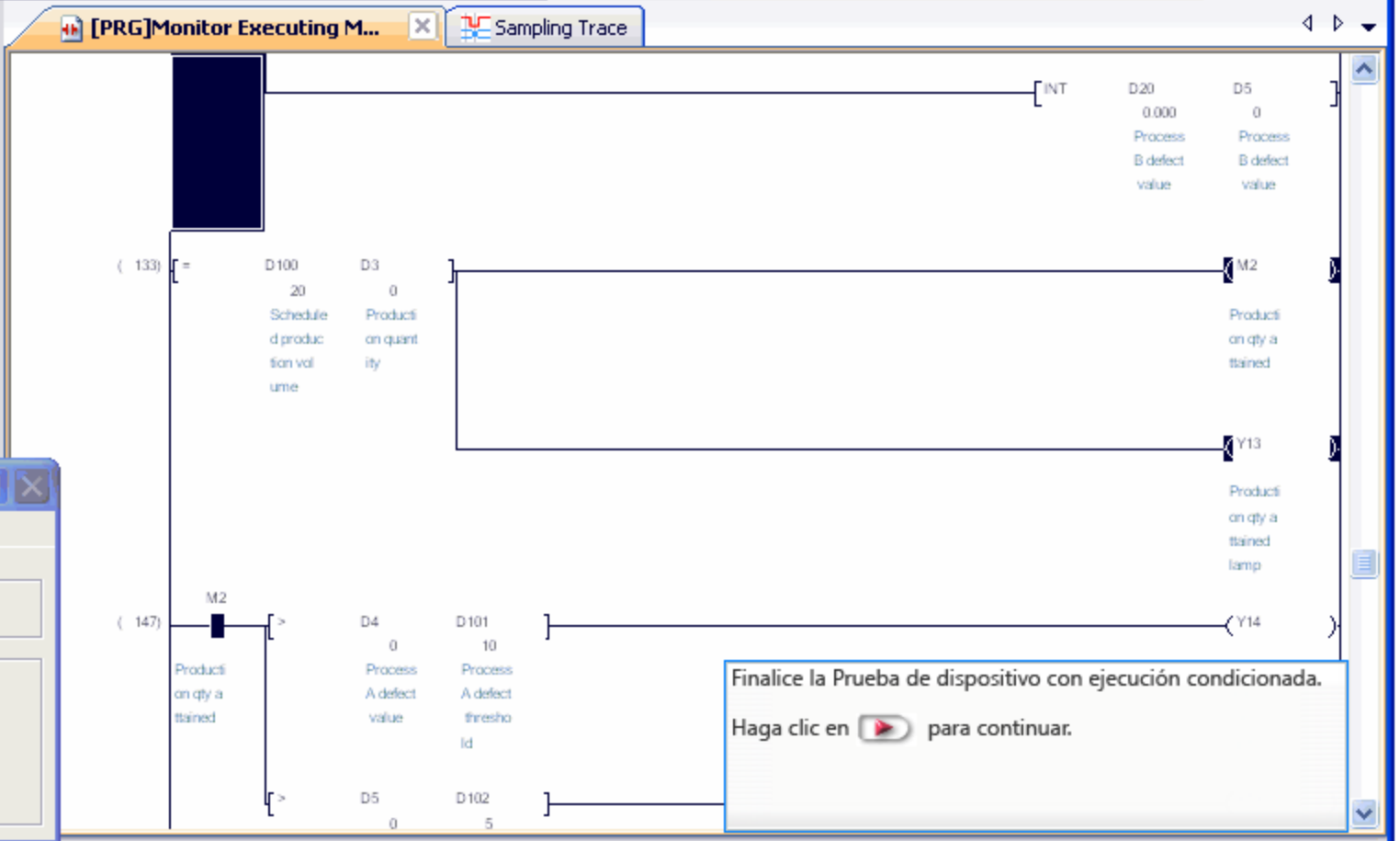
Project Edit Find/Replace Compile View Online Debug Diagnostics Tool Window Help

File Edit View Tools Windows Help

Navigation [PRG]Monitor Executing M... Sampling Trace

Project

- Parameter Prod line contr
- Intelligent Function Modu
- Global Device Comment
- Global Label
- Program Setting
- POU
  - Program
    - MAIN
      - Program
        - Local Label
- FB\_Pool



GX Simulator 2

Tool Options

Switch

RESET  STOP  RUN

LED

MODE

RUN

ERR.

USER

Finalice la Prueba de dispositivo con ejecución condicionada.  
 Haga clic en para continuar.

## 3.5 Depurar la operación del programa paso a paso

Calidad

Durante la depuración, es posible que desee confirmar la ejecución de instrucción en cada paso o verificar si hay cambios en el valor de un dispositivo.

Sin embargo, la depuración paso a paso puede resultar difícil debido a la rápida velocidad de ejecución del programa (tiempo de escaneo).

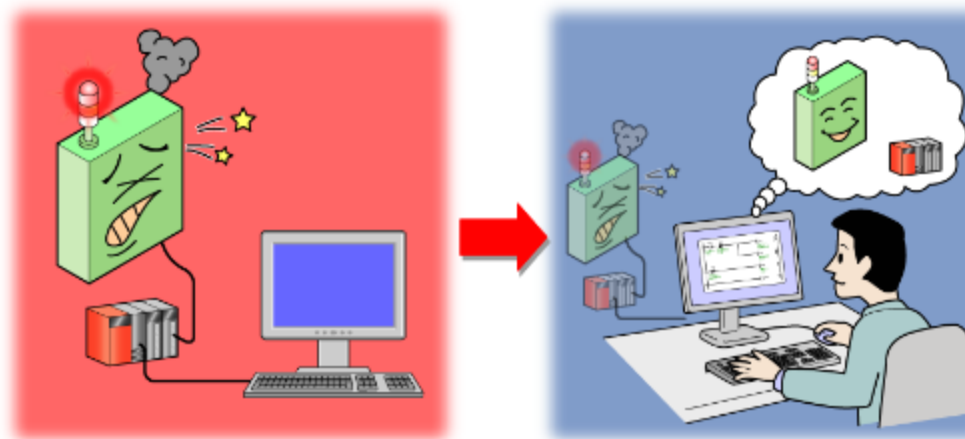
Use la **"Función de ejecución escalonada"** para resolver este problema. (Solo para la serie MELSEC-Q y la serie MELSEC-L)

Esta función permite la ejecución del programa un paso por vez e implementa así la depuración paso a paso.

Use la función de ejecución escalonada junto con la **función de simulación**. (La función de ejecución escalonada no puede usarse para depurar un PLC real.)

Las siguientes funciones pueden ser usadas por la función de ejecución escalonada.

Función	Descripción
Ejecución de interrupción	Esta función ejecuta el programa hasta que se cumplen las condiciones de interrupción especificadas. La ejecución del programa se detiene cuando se cumplen las condiciones de interrupción. Use un punto de interrupción y un dispositivo de interrupción para especificar las condiciones de interrupción.
Ejecución escalonada	Esta función ejecuta el programa paso a paso.
Ejecución parcial	Esta función ejecuta el programa solo desde la ubicación especificada.





## 3.5.1

# Usar la función de ejecución escalonada

Calidad

Especifique un **punto de interrupción** y un **dispositivo de interrupción** como la ubicación de inicio de la depuración y la condición de inicio, respectivamente.

También puede especificar un **rango de salto** dentro del cual usted desee evitar temporalmente la ejecución de programa. (Solo para la serie MELSEC-Q y la serie MELSEC-L)

Cuando se cumplan las condiciones de interrupción después de que se inicie la **ejecución de interrupción**, se interrumpe la ejecución del programa.

Por consiguiente, mientras se ejecute la operación del programa paso a paso con la **función de ejecución escalonada**, verifique si hay cambios en el valor del dispositivo para localizar una falla.

### <Punto de interrupción>

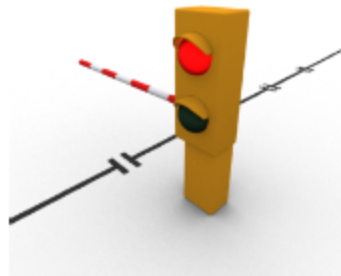
Ajuste un punto de interrupción donde desee interrumpir la ejecución del programa.

Especifique esto en unidades de pasos. Es posible ajustar hasta 64 puntos de interrupción en todo el proyecto.



### <Dispositivo de interrupción>

Ajuste un dispositivo de interrupción en base al cual la ejecución de programa se interrumpe cuando el valor de etiqueta o dispositivo cumple la condición especificada. Es posible ajustar dispositivos de hasta 16 bits y/o palabras.



### <Rango de salto>

Ajuste un rango, en unidades de bloques en escalera, dentro del cual no debería ejecutarse el programa durante la ejecución escalonada.

Es posible especificar hasta 16 rangos en todo el proyecto.



### <Aplicación de rango de salto>

Es posible reducir un punto de falla usando el **rango de salto**.

Realice una ejecución de interrupción con y sin un ajuste de rango de salto.

Si se produce una falla solo cuando se libera el ajuste de rango de salto, esto indica que el rango incluye una falla.

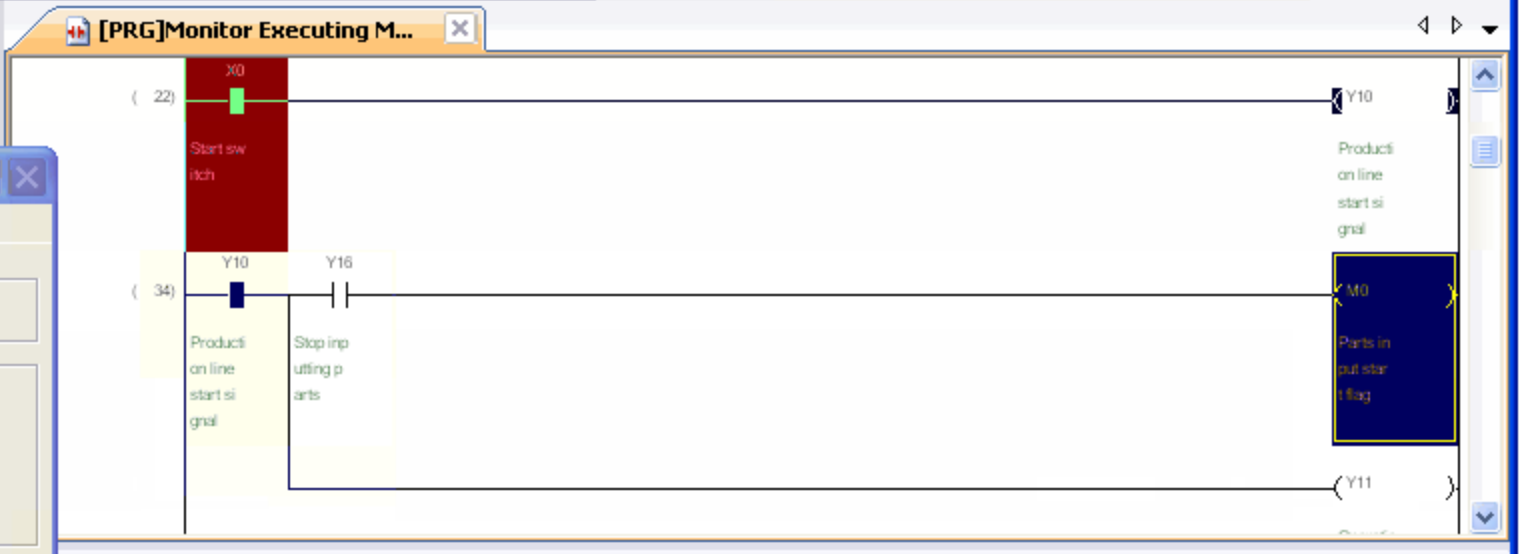
# 3.5.1 Usar la función de ejecución escalonada

MELSOFT Series GX Works2 C:\Sequential Programs\learning\Robot control - [[PRG]Monitor Executing MAIN (Read Only) 194 Step

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

Navigation

Project



GX Simulator2

Tool Options

Switch

RESET STOP RUN

LED

MODE RUN ERR. USER

Break Device

New Condition Cancel Cancel All

Combination

Judge each break device (OR condition) Judge all break devices (AND condition)

	Enable/Disable	Comparative Source (Device/Label)	Condition	Comparative Target (Value/Device/Label)	Comparative Type
1	<input checked="" type="checkbox"/>	Y10	Rising	-	Bit

Continuar

Skip Range Break Point Break Device

English Simple Q03UDE Simulation(Step executing) (37 N)

## 3.6

# Simular la operación de un dispositivo externo

Calidad

La depuración en conjunción con la operación de un dispositivo externo no es posible en un ambiente de desarrollo en el que no pueda utilizarse un dispositivo externo, como la función de simulación.

Para resolver este problema, se agrega de forma convencional un programa de depuración que simula la operación de un dispositivo externo.

Sin embargo, crear un programa de simulación no solo demanda una inversión considerable de tiempo y esfuerzo, sino que también es necesario modificar el programa al cambiar la operación.

Use **"Ajuste del sistema de E/S"** para resolver este problema.

Esta función puede simular la operación de un dispositivo externo sin usar un programa de depuración.

La operación de un dispositivo externo puede ajustarse o cambiarse fácilmente en la ventana de ajuste. Por ello, no es necesaria la creación/modificación de un programa convencional.

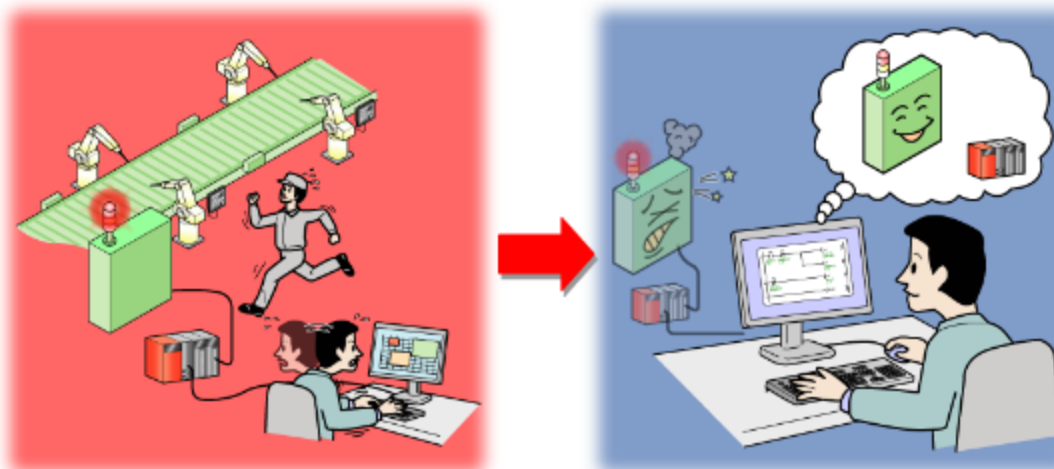
Es posible ajustar la operación de un dispositivo externo de las dos formas siguientes:

### <Ajustar los valores de dispositivo>

El valor del dispositivo especificado puede cambiarse al momento de realizar el ajuste del temporizador después de haber cumplido las condiciones especificadas.

### <Ajuste en formato de diagrama de sincronización>

Puede efectuarse un cambio de dispositivo que se haya ajustado en el formato de diagrama de sincronización especificado cuando se cumplan las condiciones especificadas.



# 3.6.1 Ingresar los valores del dispositivo para ajustar el sistema de E/S

MELSOFT Series GX Works2 C:\Sequential Programs\learning\Robot control - [[PRG]Monitor Executing MAIN (Read Only) 194 Step]

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

Navigation [PRG]Monitor Executing M...

Project

- Parameter Prod line contr
- Intelligent Function Modu
- Global Device Comment
- Global Label
- Program Setting
- POU
  - Program
    - MAIN
      - Program
      - Local Label
  - FB\_Pool
  - Structured Data Type



GX Simulator 2

Tool Options

Switch

RESET  STOP  RUN

LED

MODE	■
RUN	■
ERR.	■
USER	■

Finalice el ajuste y la ejecución del valor de dispositivo.  
 Haga clic en para continuar.

# 3.6.2 Usar el formato de diagrama de sincronización para ajustar el sistema de E/S

MELSOFT Series GX Works2 C:\Sequential Programs\learning\Robot control - [[PRG]Monitor Executing MAIN (Read Only) 194 Step

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

Navigation [PRG]Monitor Executing M...

Project

- Parameter Prod line contr
- Intelligent Function Modu
- Global Device Comment
- Global Label
- Program Setting
- POU
  - Program
    - MAIN
    - Program
    - Local Label
- FB\_Pool
- Structured Data Type

Scheduled prod qty setting

( 0) SM400


Always ON

[MOV K20 D100 20 Schedule d produc tion vol ume]

[MOV K10 D101 10 Process A defect thresho ld]

[MOV K5 D102 5 Process B defect thresho ld]

( 22) X0 Start sw itch

Finalice el ajuste y la ejecución del diagrama de sincronización.  
Haga clic en  para continuar.

Simple Q03UDE Simulation (0/ N:3

GX Simulator 2

Tool Options

Switch

RESET  RUN

LED

MODE	<span style="color: green;">■</span>
RUN	<span style="color: green;">■</span>
ERR.	<span style="color: gray;">■</span>
USER	<span style="color: gray;">■</span>

## Capítulo 4 Administración de proyecto y medidas de seguridad

### Pasos para el aprendizaje en el Capítulo 4

En el Capítulo 4, aprenderá sobre la funciones usadas para la administración de proyecto y las medidas de seguridad.

- 4.1 Evitar la fuga de know-how y la modificación no autorizada de programas
  - 4.1.1 Limitar los datos accesibles por cada usuario
- 4.2 Copia de respaldo del proyecto y gestión de versiones
- 4.3 Comparar los programas guardados en el controlador programable y la computadora personal

## 4.1

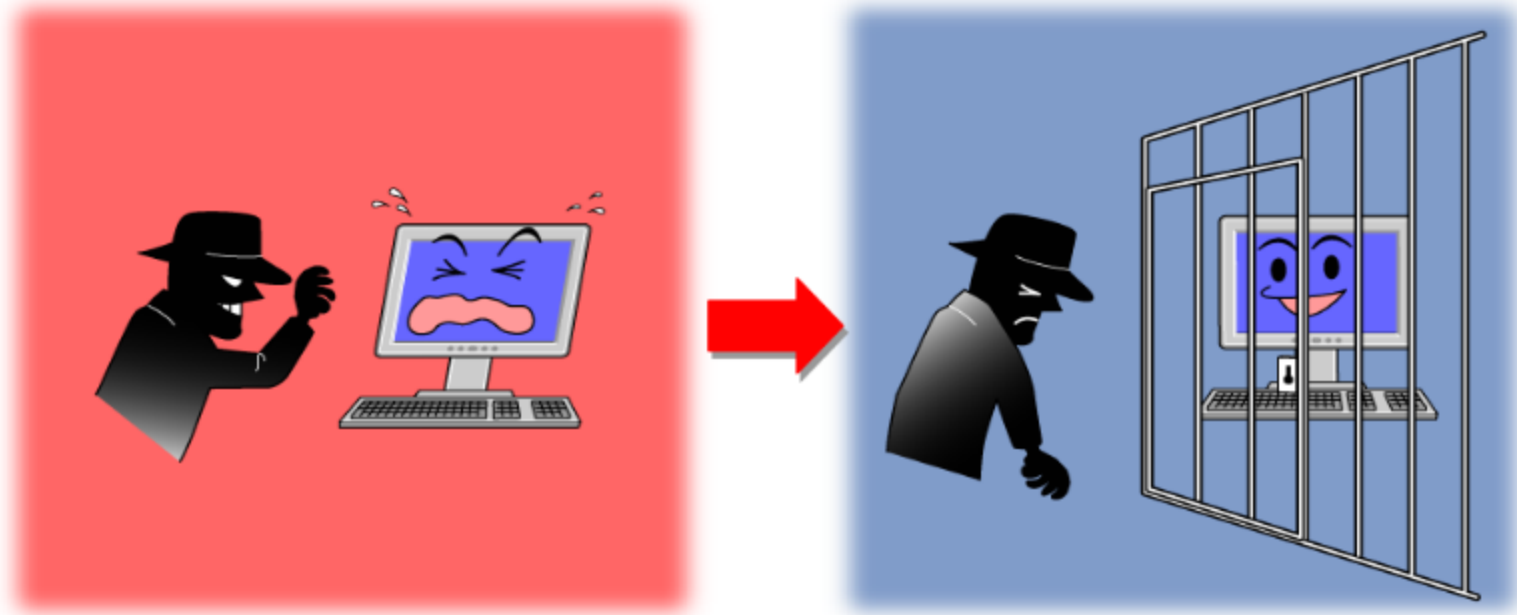
## Evitar la fuga de know-how y la modificación no autorizada de programas

Seguridad

Gestión

El programa secuencial incluye know-how y datos de importancia estratégica.  
La fuga de know-how y datos del programa al exterior podría tener un efecto devastador en el negocio.  
La modificación no autorizada del programa podría acarrear problemas de producción, como la parada del sistema.

Use **"Seguridad"** para resolver estos problemas.  
Esta función limita a los usuarios que pueden acceder a cada proyecto bajo protección mediante contraseña.  
También puede limitar el rango de datos o funciones al que cada usuario puede acceder u operar.  
La función evita así que usuarios no autorizados exploren o editen programas.







# 4.1.1 Limitar los datos accesibles por cada usuario



El desarrollo de un programa secuencial a gran escala está a cargo de dos o más programadores que comparten el trabajo. En caso de desarrollo en equipo, el rango de datos accesibles y funciones disponibles debe manejarse adecuadamente según el rango de trabajo y el nivel de destrezas de cada programador y la confidencialidad de los datos manejados por cada programador.

Este manejo del acceso puede implementarse ajustando **niveles de acceso** de seguridad.

### <Nivel de acceso>

Los privilegios de operación para los datos incluidos en el proyecto pueden ajustarse para cada usuario. Es posible ajustar los siguientes cinco niveles de acceso.

Nivel de acceso		Autoridad de operación	
<b>Alta</b>  <b>Baja</b>	Administradores	Nivel de administrador	Autorizados a usar todas las funciones.
	Desarrolladores (Nivel3)	Nivel de desarrollador	Los ajustes de seguridad, el acceso a los datos y algunas operaciones están restringidas.
	Desarrolladores (Nivel2)		
	Desarrolladores (Nivel1)		
	Usuarios	Nivel de operador	Solo está disponible la exploración de datos del proyecto. No pueden leer desde el CPU del PLC.

# 4.1.1 Limitar los datos accesibles por cada usuario

MELSOFT Series GX Works2 C:\Sequential Programs\learning\Robot control - [[PRG]Read MAIN (Read Only) 194 Step]

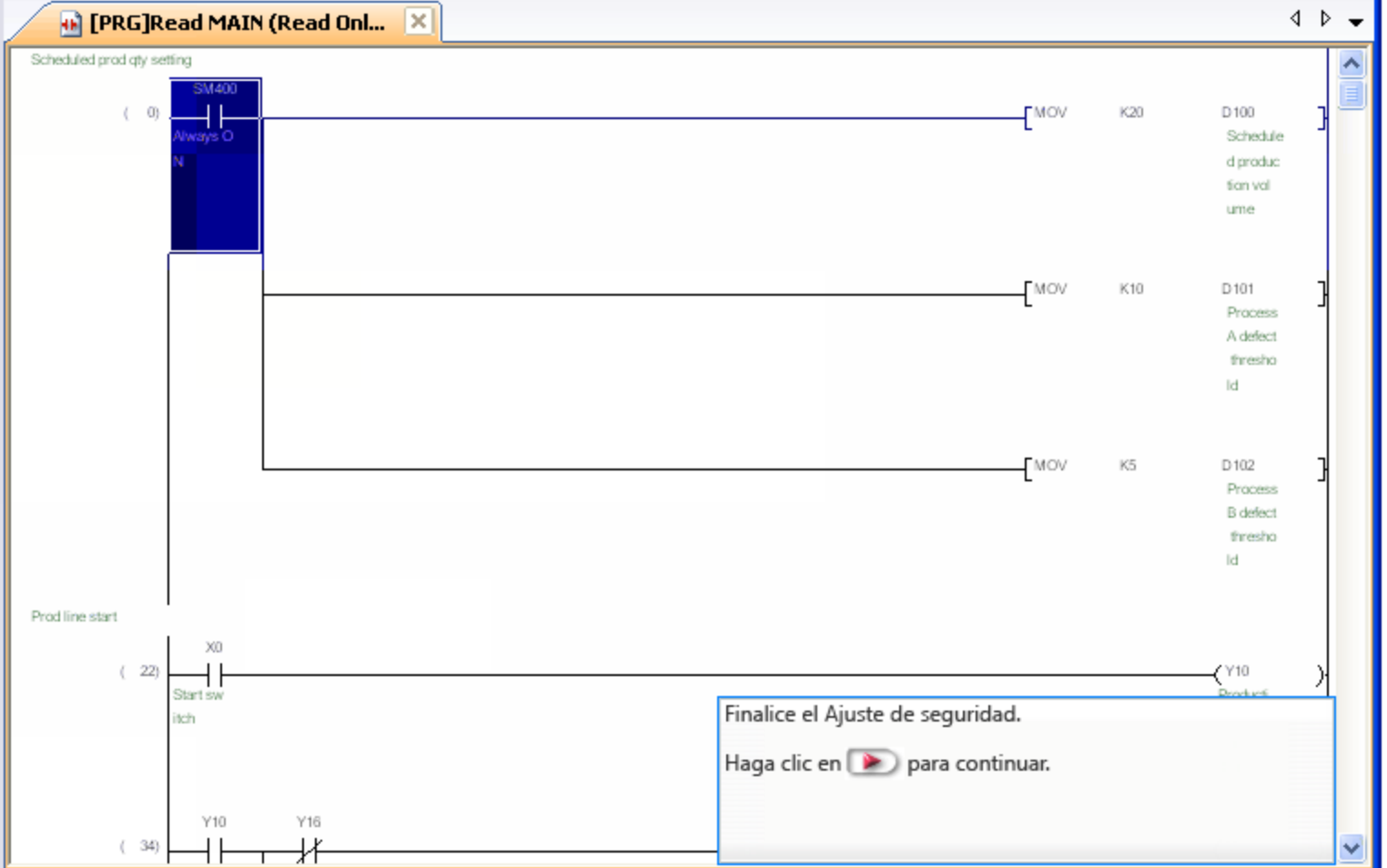
Project Edit Find/Replace Compile View Online Debug Diagnostics Tool Window Help



**Navigation**

**Project**

- Parameter Prod line control s
- Intelligent Function Module
- Global Device Comment
- Global Label
- Program Setting
- POU
  - Program
    - MAIN
      - Program
        - Local Label
  - FB\_Pool
  - Structured Data Types
  - Local Device Comment
- Device Memory
- Device Initial Value



Finalice el Ajuste de seguridad.  
 Haga clic en para continuar.



## 4.2

## Copia de respaldo del proyecto y gestión de versiones



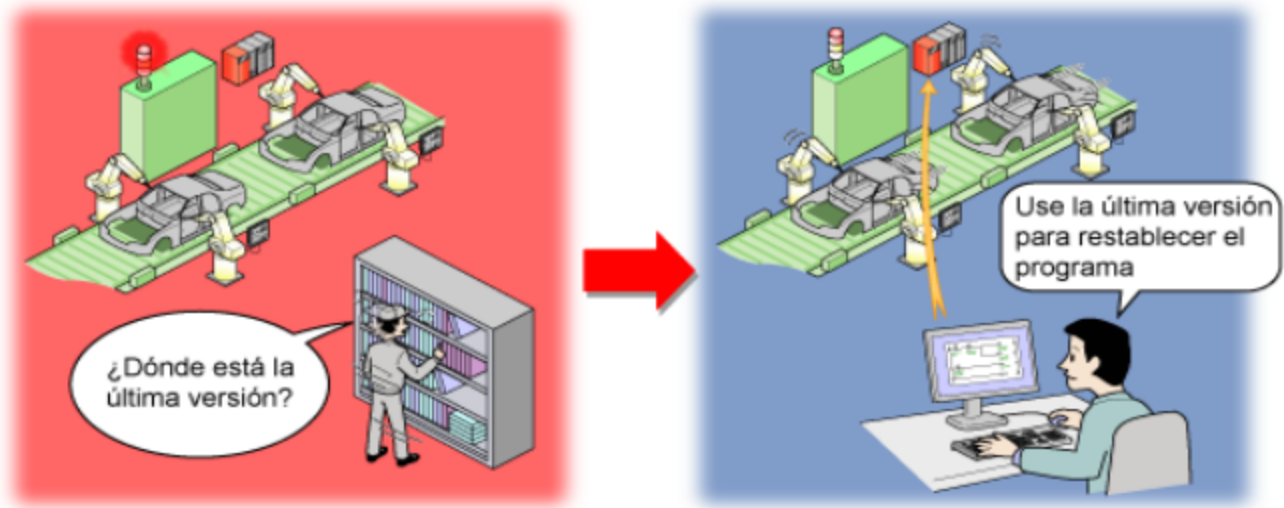
Un programa importante podría perderse debido a una falla del CPU del controlador programable. Si no estuviera disponible un programa de respaldo, hay pocas posibilidades de una recuperación rápida. Incluso si estuviera disponible un programa de respaldo, sería difícil determinar si la versión es la misma que el programa perdido, a menos que se garantice una gestión de versiones. Con el fin de estar preparados para circunstancias inesperadas, es necesario realizar copias de respaldo periódicas y garantizar la gestión de versiones.

Use **"Historial de cambios"** para resolver estos problemas.

Esta función puede registrar hasta 100 conjuntos de historiales de cambios (número de historial, fecha/hora, usuario, título, comentario) del proyecto.

También se hace, al mismo tiempo, una copia de respaldo de los datos del proyecto al momento del registro.

La gestión de versiones garantizada por la función de historial de cambios le permite restaurar los programas perdidos, verificar las versiones de programas e implementar así la recuperación rápida en caso de circunstancias inesperadas.



# 4.2 Copia de respaldo del proyecto y gestión de versiones

MELSOFT Series GX Works2 C:\Sequential Programs\learning\Robot control (Revision No.3 : Revision B) - [[PRG]Write MAIN 194 Step

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



**Navigation**

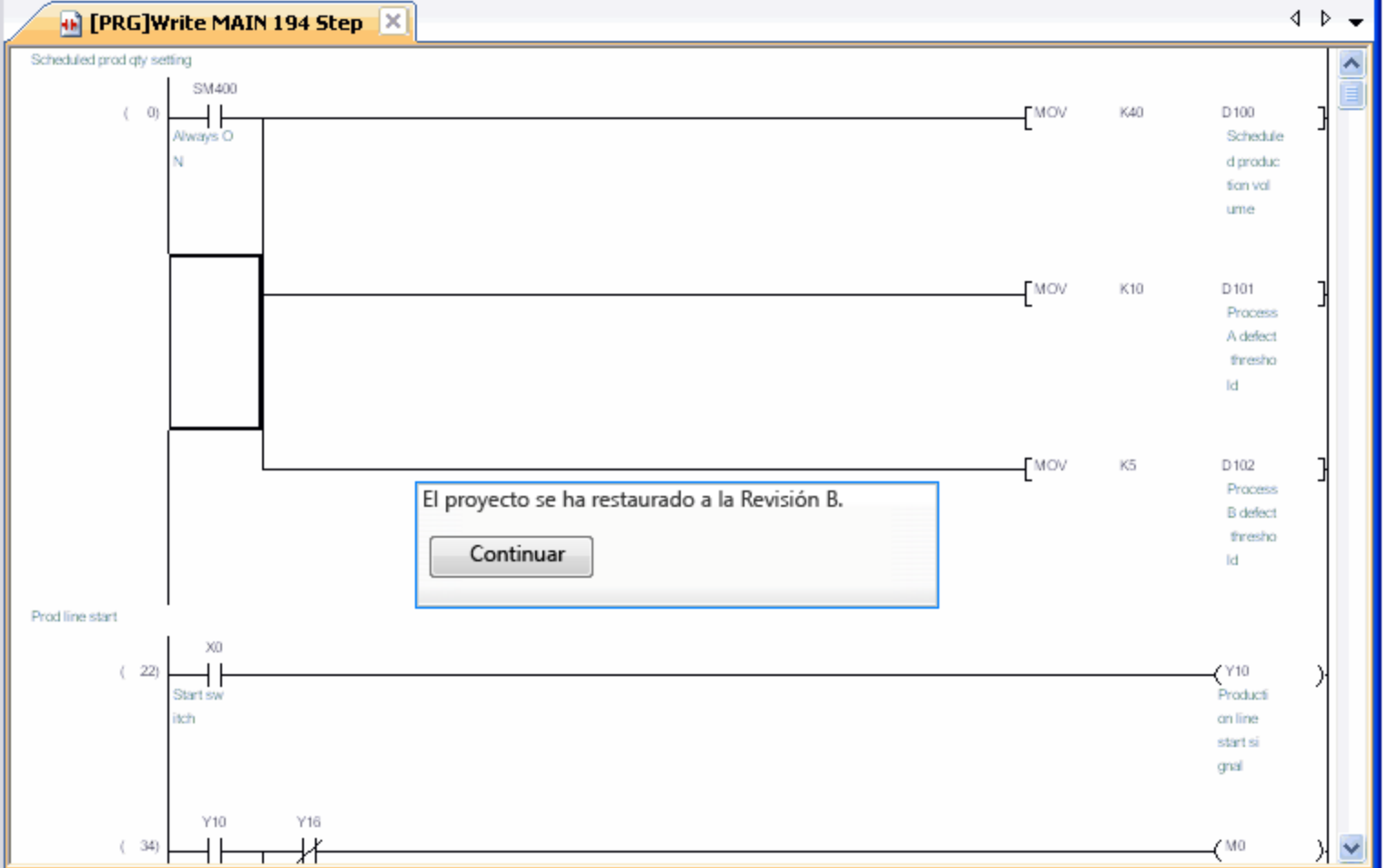
**Project**

- Parameter Prod line control s
- Intelligent Function Module
- Global Device Comment
- Global Label
- Program Setting
- POU
  - Program
    - MAIN
      - Program
        - Local Label
- FB\_Pool
- Structured Data Types
- Local Device Comment
- Device Memory
- Device Initial Value

**Project**

**User Library**

**Connection Destination**



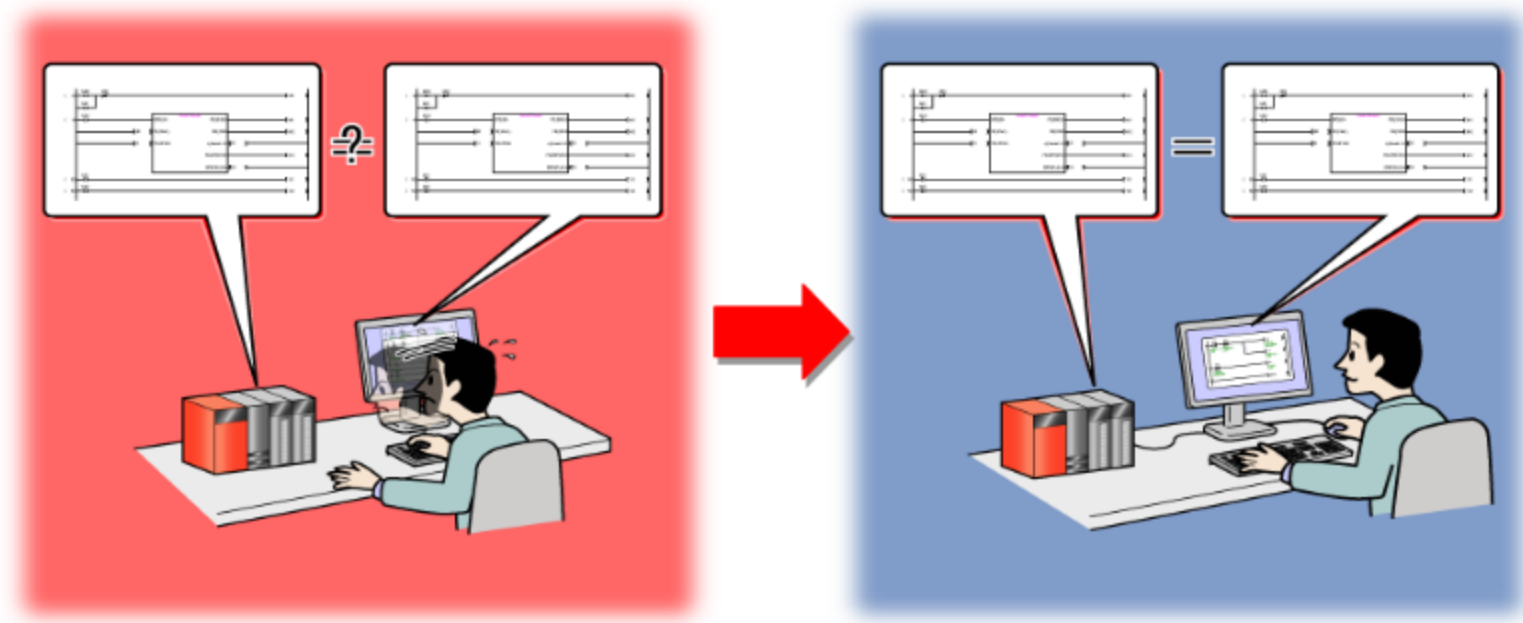
El proyecto se ha restaurado a la Revisión B.

### 4.3 Comparar programas guardados en el controlador programable y la computadora personal



Normalmente, los programas se guardan en una PC en el ambiente de desarrollo, aunque también se escriben al PLC. Estos dos programas no siempre son lo mismo. Realizar solo una verificación visual para comprobar que son lo mismo podría acarrear errores.

Use **"Verificar con PLC"** para resolver este problema. Esta función puede verificar que el programa abierto por GX Works2 coincide con el programa escrito al PLC.



# 4.3 Comparar programas guardados en el controlador programable y la computadora personal

MELSOFT Series GX Works2 C:\Sequential Programs\learning\Robot control - [[PRG]Write MAIN 194 Step]

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

File Edit View Tools PLC I/O Monitor Simulation

F5 SF5 F6 SF6 F7 F8 F9 SF9 CF9 CF10 SF7 SF8 AF7 AF8 SAF5 SAF6 SAF7 SAF8 AF5 AF6 AF7 AF8 F10 AF9

**Navigation**

[PRG]Write MAIN 194 Step [Verify Result [PLC Verify]]

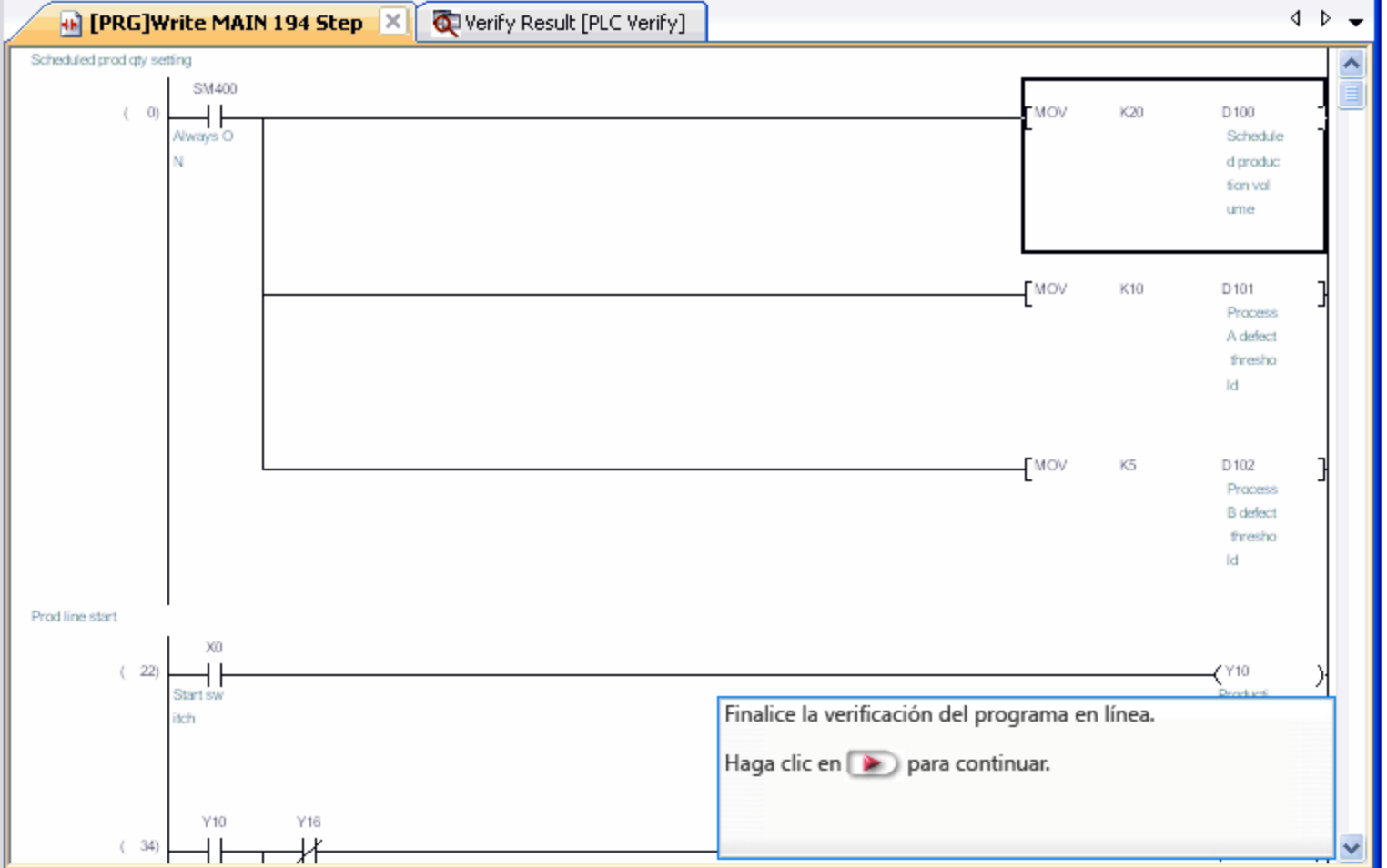
**Project**

- Parameter Prod line control s
- Intelligent Function Module
- Global Device Comment
- Global Label
- Program Setting
- POU
  - Program
    - MAIN
      - Program
      - Local Label
  - FB\_Pool
  - Structured Data Types
  - Local Device Comment
- Device Memory
- Device Initial Value

**Project**

**User Library**

**Connection Destination**



Finalice la verificación del programa en línea.  
 Haga clic en para continuar.

## Prueba Prueba final

Ahora que ha completado todas las lecciones del Curso **PLC Conceptos Avanzados de GX Works2**, está listo para tomar la prueba final. Si no tiene claro alguno de los temas cubiertos, tome esta oportunidad para revisar esos temas.

**Hay un total de 8 preguntas (8 áreas) en esta Prueba final.**

Puede tomar la prueba final las veces que desee.

### Cómo calificar la prueba

Luego de seleccionar la respuesta, asegúrese de hacer clic en el botón **Respuesta**. Su respuesta se perderá si no hace clic en el botón Respuesta. (Se considerará como pregunta sin respuesta.)

### Resultados de la calificación

El número de respuestas correctas, el número de preguntas, el porcentaje de respuestas correctas y el resultado sobre si aprobó o no aparecerá en la página de calificación.

Respuestas correctas: 2

Total de preguntas: 9

Porcentaje: 22%

Para aprobar la prueba, debe responder correctamente el **60 %** de las preguntas.

Continuar

Revisar

Volver a intentar

- Haga clic en el botón **Continuar** para salir de la prueba.
- Haga clic en el botón **Revisar** para revisar la prueba. (Verificar la respuesta correcta)
- Haga clic en el botón **Volver a intentar** para tomar la prueba nuevamente.



## Prueba Prueba final 1

¿Cuál de las siguientes funciones permite una programación eficiente mediante el uso de bloques de escalera de uso reiterado como componentes que pueden ser compartidos? (Elegir una.)

- Texto estructurado en línea
- Etiqueta
- Bloque de función

Respuesta

Volver

## Prueba Prueba final 2

¿Cuál de las siguientes funciones puede crear programas fáciles de leer al cambiar los nombres de dispositivo por nombres asociados con su aplicación? (Elegir una.)

- Comentario de dispositivo
- Etiqueta
- Nota

Respuesta

Volver

## Prueba Prueba final 3

¿Cuál de las siguientes funciones puede crear programas fáciles de leer al suministrar información sobre el procesamiento para cada bloque de escalera? (Elegir una.)

- Comentario de dispositivo
- Declaración de línea
- Nota

Respuesta

Volver

## Prueba Prueba final 4

¿Cuál de las siguientes es la explicación correcta de la función "Verificar con PLC"? (Elegir una.)

- Compara el programa que está editándose con un programa registrado en el historial de cambios.
- Compara el programa que está editándose con un programa seleccionado guardado en la PC.
- Compara el programa que está editándose con un programa escrito en la CPU del PLC.

Respuesta

Volver

**Prueba Prueba final 5**

¿Cuál de las siguientes es la explicación correcta para la función de "Ajuste del Sistema de E/S"? (Elegir una.)

- Simula la operación de equipos de E/S externos en la computadora personal durante la depuración.
- Controla en forma remota la operación de los equipos de E/S externos desde la computadora personal durante la depuración.
- Simula la operación de la CPU del PLC en la computadora personal durante la depuración.

Respuesta

Volver

**Prueba Prueba final 6**

¿Cuál de las siguientes es la explicación correcta de la función "Historial de cambios"? (Elegir una.)

- Registra la operación de GX Works2 paso por paso de forma tal que pueda restaurarse libremente más adelante.
- Registra información de historial y copias de respaldo del proyecto para permitir la verificación y restauración más adelante.

Respuesta

Volver

## Prueba Prueba final 7

¿Cuál de las siguientes funciones puede ser utilizada durante la depuración para cambiar únicamente los valores de dispositivo sin modificar el programa? (Elegir una.)

- Ejecución de interrupción
- Ajuste del sistema de E/S
- Prueba de dispositivo de ejecución condicionada

Respuesta

Volver



## Prueba Prueba final 8

Asuma que el proyecto incluye dos programas, A y B, y usted usa la función "etiqueta".  
¿A cuál de los siguientes tipos de etiqueta puede accederse mediante el programa B? (Elija dos.)

- Etiqueta global
- Etiqueta local del programa A
- Etiqueta local del programa B

Respuesta

Volver

## Prueba **Calificación de la prueba**

Ha completado la prueba final. Sus resultados son los siguientes.  
Para terminar con la Prueba Final, vaya a la página siguiente.

Respuestas correctas : **0**

Total de preguntas : **8**

Porcentaje : **0%**

Continuar

Revisar

Volver a intentar

**No ha aprobado la prueba.**

Ha completado el Curso **PLC Conceptos Avanzados de GX Works2.**

Gracias por tomar este curso.

Esperamos que haya disfrutado las lecciones y que la información recibida en este curso le sea útil en el futuro.

Puede revisar el curso las veces que desee.

**Revisar**

**Cerrar**