



PLC GX Works2 Lanjutan

Kursus ini memberikan pengetahuan tentang fungsi-fungsi untuk meningkatkan lingkungan pengembangan situs desain yang mengalami masalah "produktivitas," "kualitas," "manajemen proyek," dan "langkah pengamanan." Kursus ini dirancang untuk pengembang program sekuens yang sudah menggunakan MELSOFT GX Works2.

Pendahuluan Tujuan Kursus

Kursus ini memberikan pengetahuan tentang fungsi-fungsi untuk meningkatkan lingkungan pengembangan situs desain yang mengalami masalah "produktivitas," "kualitas," "manajemen proyek," dan "langkah pengamanan." Kursus ini dirancang untuk pengembang program sekuens yang sudah menggunakan MELSOFT GX Works2.

Pendahuluan **Struktur Kursus**

Berikut adalah daftar isi kursus.
Sebaiknya Anda mulai dari Bab 1.

Bab 1 - Meningkatkan Lingkungan Pengembangan Menggunakan GX Works2

Anda akan mempelajari masalah yang dihadapi situs desain dan lingkungan pengembangan yang diperlukan untuk mengatasinya.

Bab 2 - Pemrograman

Anda akan mempelajari fungsi-fungsi yang digunakan untuk pemrograman.

Bab 3 - Debugging

Anda akan mempelajari fungsi-fungsi yang digunakan untuk debugging.

Bab 4 - Manajemen Proyek dan Langkah Pengamanan

Anda akan mempelajari fungsi-fungsi untuk manajemen proyek dan langkah pengamanan pada tahap pengembangan dan perawatan setelah dimulainya pengoperasian sistem.

Tes Akhir

Nilai lulus: 60% dan lebih tinggi

Pendahuluan Cara Menggunakan Alat e-Learning Ini

Buka halaman berikutnya		Membuka halaman berikutnya.
Kembali ke halaman sebelumnya		Kembali ke halaman sebelumnya.
Beralih ke halaman yang diinginkan		"Daftar Isi" akan ditampilkan, memungkinkan Anda untuk menavigasi ke halaman yang diinginkan.
Keluar dari kursus		Keluar dari kursus. Jendela seperti layar "Daftar Isi" dan kursus akan ditutup.

Pendahuluan **Perhatian Selama Penggunaan**

Petunjuk keselamatan

Saat Anda belajar dengan memakai produk sebenarnya, bacalah dengan cermat petunjuk keselamatan pada panduan yang sesuai.

Petunjuk keselamatan dalam kursus ini

- Layar yang ditampilkan pada versi perangkat lunak yang Anda gunakan mungkin berbeda dengan yang ada di dalam kursus ini.

Bab 1 Meningkatkan Lingkungan Pengembangan Menggunakan GX Works2

Langkah-langkah pembelajaran pada Bab 1

Kursus ini dirancang bagi programmer yang menangani pengembangan program sekuens. Anda akan mempelajari cara menggunakan fungsi-fungsi istimewa pada **MELSOFT GX Works2** untuk menyelesaikan masalah situs desain yang terkait dengan "**produktivitas,**" "**kualitas,**" "**manajemen proyek,**" dan **keamanan.**"

Pada Bab 1, Anda akan mempelajari masalah yang dihadapi situs desain dan lingkungan pengembangan yang diperlukan untuk mengatasinya.

- 1.1 Masalah yang Dihadapi Situs Desain
 - 1.1.1 Meningkatkan Produktivitas
 - 1.1.2 Meningkatkan Kualitas
 - 1.1.3 Manajemen Proyek
 - 1.1.4 Langkah Pengamanan
- 1.2 Prosedur Pembelajaran

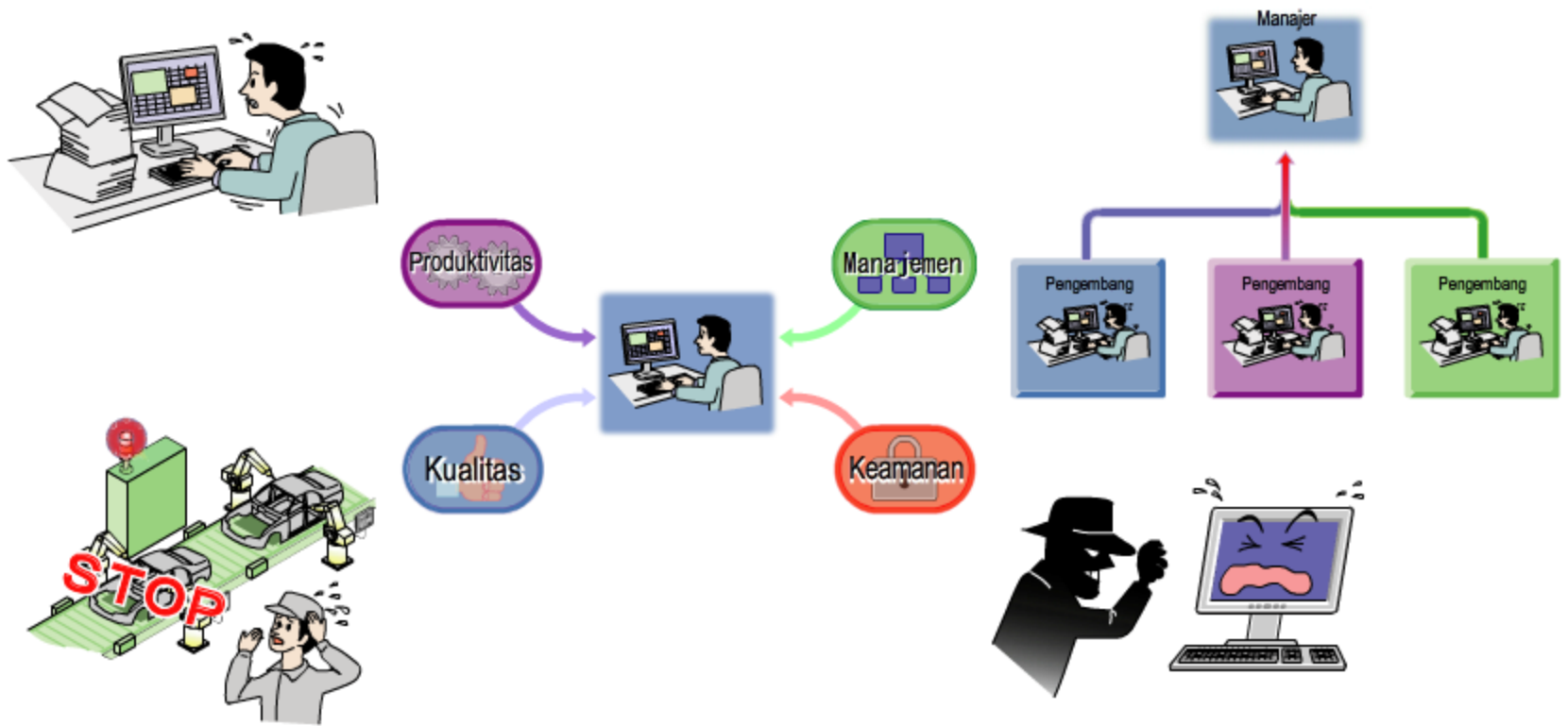


1.1 Masalah yang Dihadapi Situs Desain

Meskipun diperlukan untuk meningkatkan produktivitas guna mengurangi biaya pengembangan, situs desain juga harus memastikan bahwa program yang dikembangkan berkualitas tinggi.

Situs desain juga harus menjalankan manajemen proyek yang memungkinkan pengembangan tim dan memastikan pemulihan yang cepat bila terjadi gangguan.

Langkah pengamanan juga diperlukan karena program sekuens mencakup pengetahuan dan data penting.



1.1.1 Meningkatkan Produktivitas

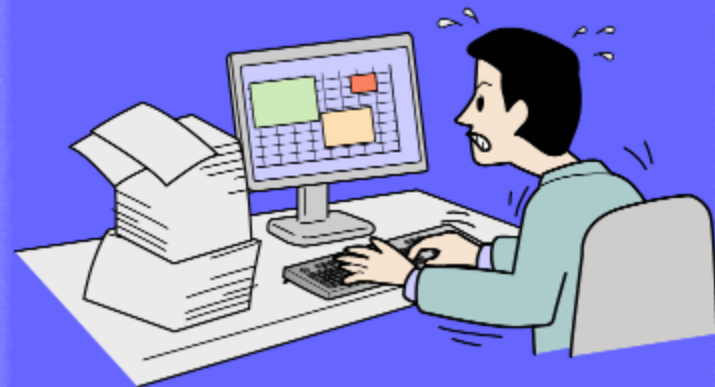
Situs desain diperlukan untuk mengembangkan program sekuens untuk menangani pemrosesan skala besar dan kompleks. Namun, biaya pengembangan meningkat seiring dengan lamanya periode pengembangan. Jadi, Anda diminta meningkatkan produktivitas pengembangan program guna mengurangi biaya pengembangan.

GX Works2 menyediakan fungsi-fungsi berikut untuk menyelesaikan masalah ini:

- Label
- Blok fungsi
- Nilai awal perangkat dan memori perangkat*
- Teks berstruktur sebaris
- Impor dari komentar contoh

* MELSEC seri-F tidak dapat mengonfigurasi nilai awal perangkat.

Produktivitas



1.1.2 Meningkatkan Kualitas

Bug pada program sekuens dapat menyebabkan sistem berhenti, gagal, atau mengalami kecelakaan, sehingga menghambat produksi dan mengakibatkan hilangnya keuntungan dan reliabilitas. Anda diminta mengembangkan program berkualitas yang bebas dari bug apa pun.

GX Works2 menyediakan fungsi-fungsi berikut untuk menyelesaikan masalah ini:

- Label
- Blok fungsi
- Nilai awal perangkat dan memori perangkat*
- Komentar
- Impor dari komentar contoh
- Pengawasan
- Referensi silang
- Pelacakan sampling*
- Uji perangkat terkondisi-eksekusi*
- Fungsi eksekusi langkah*
- Pengaturan sistem I/O



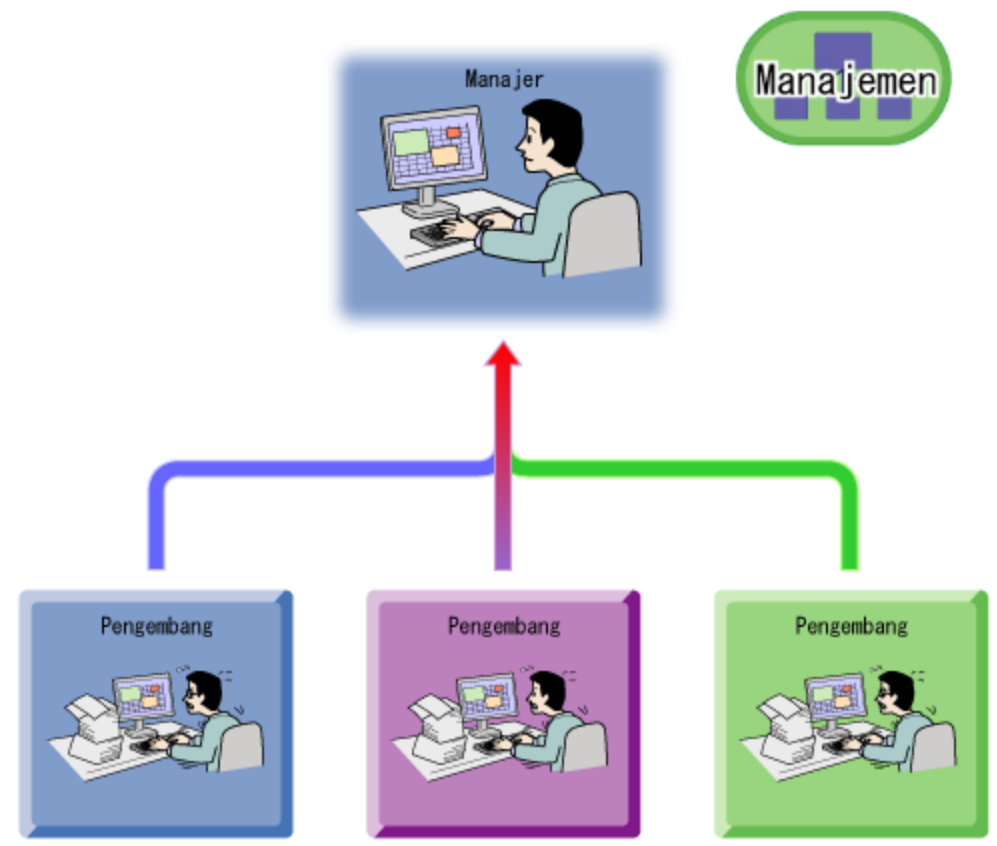
* MELSEC seri-F tidak dapat menggunakan pengaturan nilai awal perangkat, pelacakan sampling, uji perangkat terkondisi-eksekusi, dan fungsi eksekusi langkah.

1.1.3 Manajemen Proyek

Program sekuens skala-besar sering dikembangkan oleh tim programmer.
 Dari sudut pandang otoritas manajemen individu dan kerahasiaan data, harus ada batas-batas yang ditetapkan untuk rentang data yang dapat diakses dan fungsi yang dapat digunakan.
 Selain itu, untuk memastikan pemulihan cepat jika program hilang akibat kegagalan PLC, Anda diminta menerapkan manajemen versi program dan pencadangan program berkala.

GX Works2 menyediakan fungsi-fungsi berikut untuk menyelesaikan masalah tersebut:

- Keamanan
- Riwayat revisi
- Verifikasi dengan PC



1.1.4 Langkah Pengamanan

Program sekuen mengandung pengetahuan dan data yang penting secara strategis.

Kebocoran pengetahuan teknis dan data dari program ke luar dapat memberikan dampak merugikan pada bisnis.

Modifikasi program tanpa wewenang dapat mengakibatkan masalah produksi seperti berhentinya sistem.

Langkah pengamanan yang sesuai harus diambil untuk mencegah masalah itu.

GX Works2 menyediakan fungsi berikut untuk menyelesaikan masalah tersebut:

- Keamanan



1.2 Prosedur Pembelajaran

Dalam kursus ini, Anda akan mempelajari fungsi-fungsi GX Works2 dengan mengikuti prosedur pengembangan sistem sebenarnya.

(1) Pemrograman Bab 2



(2) Debugging Bab 3



(3) Manajemen proyek dan langkah pengamanan Bab 4

<Penjelasan ikon>

Ikon yang ditampilkan pada halaman-halaman di Bab 2, 3, dan 4 berhubungan dengan setiap fungsi peningkatan, sebagai berikut.



Fungsi yang digunakan untuk meningkatkan produktivitas program



Fungsi yang digunakan untuk meningkatkan kualitas program



Fungsi yang digunakan untuk manajemen proyek



Fungsi yang digunakan untuk langkah pengamanan

Bab 2 Pemrograman

Langkah-langkah pembelajaran pada Bab 2

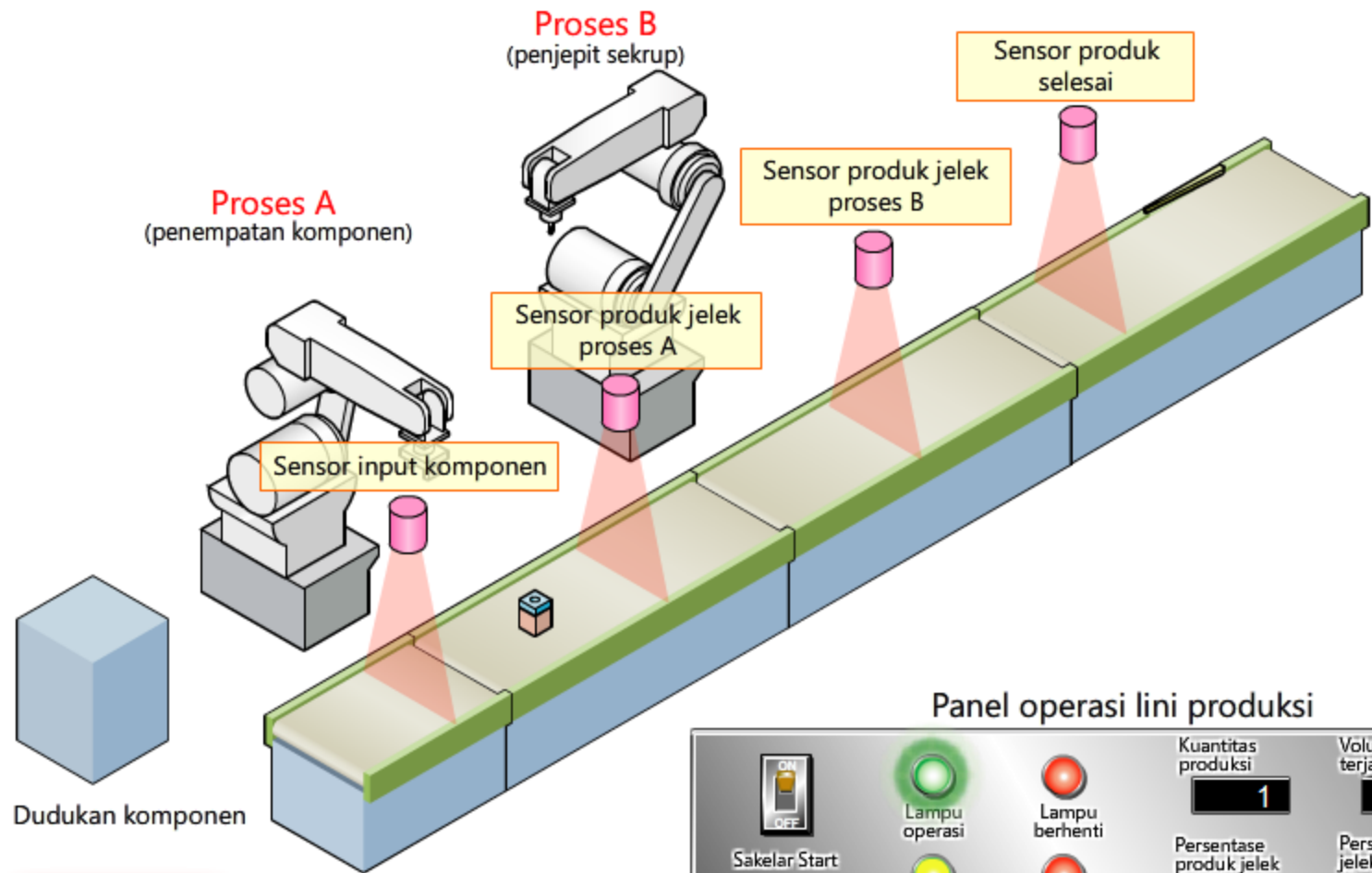
Pada Bab 2, Anda akan mempelajari fungsi-fungsi yang digunakan untuk pemrograman. GX Works2 menyediakan banyak fungsi berbeda untuk memastikan pemrograman yang efisien dan meningkatkan kualitas program.

- 2.1 Mengganti Nama Perangkat dengan Nama yang Dikaitkan dengan Aplikasi
 - 2.1.1 Tipe Label
 - 2.1.2 Tipe Aplikasi Label dan Nilai yang Disimpan
 - 2.1.3 Pendaftaran Label
 - 2.1.4 Penetapan Label ke Perangkat secara Otomatis
- 2.2 Mengatur Blok Ladder yang Digunakan Berulang sebagai Blok Fungsi untuk Diversi
 - 2.2.1 Membuat dan Menempatkan Blok Fungsi
 - 2.2.2 Menggunakan Perpustakaan Blok Fungsi
- 2.3 Mengubah Nilai Awal Perangkat tanpa Mengoreksi Program
- 2.4 Menyederhanakan Program Ladder
- 2.5 Membuat Program yang Mudah Dipahami dan Dibaca
 - 2.5.1 Menulis Komentar untuk Setiap Blok Ladder
 - 2.5.2 Menulis Komentar untuk Setiap Output (Kumpulan, Instruksi Aplikasi)
- 2.6 Memudahkan Membaca Program yang Mengandung Relai/Register Khusus

Bab 2 Mengenal Sistem yang Digunakan untuk Pembelajaran dalam Kursus Ini



Kursus ini menggunakan sistem hipotesis berikut untuk membantu Anda mempelajari cara menggunakan fungsi GX Works2:



- ▶ **Produksi normal**
- ▶ Kesalahan persentase produk jelek proses A
- ▶ Kesalahan persentase produk jelek proses B

Menekan tombol "Kembali ke Atas" mengembalikan ke pilihan operasi pertama.

Kembali ke Atas

Panel operasi lini produksi

			Kuantitas produksi	Volume produksi terjadwal
Sakelar Start	Lampu operasi	Lampu berhenti	1	20
			Persentase produk jelek proses A	Persentase produk jelek proses A yang ditoleransi
Reset kuantitas produksi	Kuantitas produksi diperoleh	Lampu kesalahan persentase produk jelek proses A	0	10
			Persentase produk jelek proses B	Persentase produk jelek proses B yang ditoleransi
		Lampu kesalahan persentase produk jelek proses B	0	5

2.1

Mengganti Nama Perangkat dengan Nama yang Dikaitkan dengan Aplikasi

Perangkat yang digunakan pada program sekuens diberi nama yang terdiri atas huruf diikuti dengan angka, misalnya "M0" atau "D5".

Jadi, nama perangkat tidak memberikan petunjuk apa pun terkait aplikasi perangkat tersebut.

Program skala-besar menggunakan banyak tipe perangkat, yang berarti bahwa selama pemrograman, Anda harus terus-menerus memeriksa dokumen desain sistem untuk mengetahui aplikasi perangkat. Cara ini mengurangi efisiensi kerja dan mempengaruhi kualitas program secara negatif karena kesalahan dalam memilih perangkat.

"Label" dapat digunakan untuk menyelesaikan masalah ini.

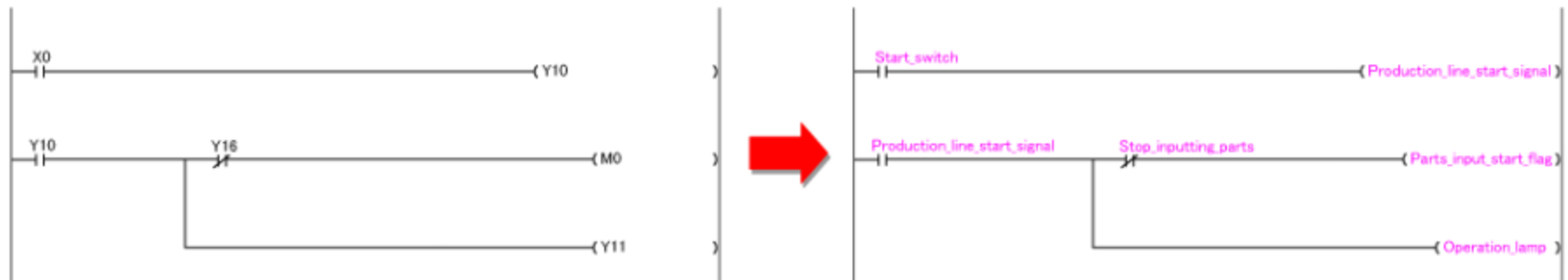
Alih-alih menggunakan nama perangkat, Anda dapat menggunakan nama (label) yang menunjukkan aplikasi sebenarnya, misalnya **"Sinyal mulai lini produksi"**.

Untuk nama tipe ini, karakter Jepang (hiragana, katakana, dan kanji) dapat digunakan sebagai tambahan karakter alfanumerik.

<Pernyataan untuk mengatur "M0" yang memuat sinyal mulai lini produksi ke ON melalui instruksi SET>

Perangkat	—[SET M0]—
Label	—[SET Production line start signal]—

Penggunaan label tipe ini adalah cara efektif untuk membuat program yang mudah dibaca, sehingga meningkatkan efisiensi pengembangan program dan mencegah kesalahan input perangkat.



2.1.1 Tipe Label

Produktivitas

Kualitas

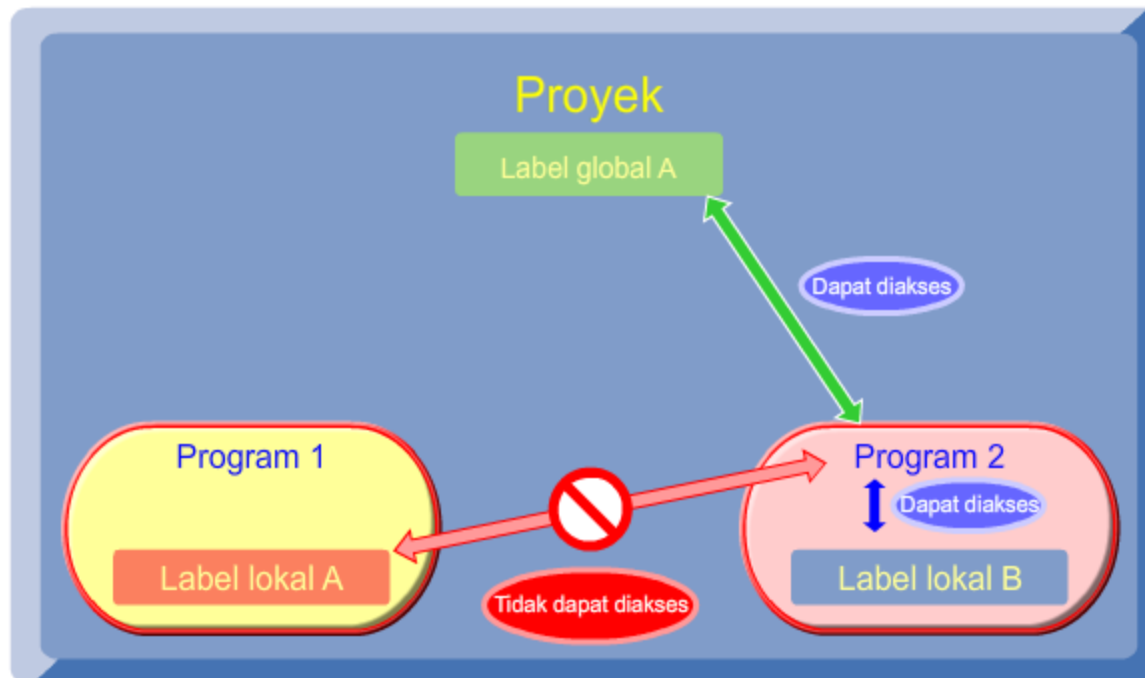
Ada dua tipe label: "**label global**" dan "**label lokal**."

<Label global>

Label global digunakan untuk seluruh proyek dan dapat diakses oleh sembarang program dalam proyek itu.

<Label lokal>

Label lokal digunakan dalam program tertentu dan hanya dapat diakses oleh program di mana label itu didaftarkan.



"Program 1" dan "Program 2" adalah tombol. Tekan salah satu program untuk melihat apakah program dapat mengakses kedua tipe label.

2.1.2 Tipe Aplikasi Label dan Nilai yang Disimpan

Produktivitas

Kualitas

Ketika mendaftarkan label, tentukan aplikasi label dan tipe nilai yang dapat disimpan menggunakan **"Kelas"** dan **"Tipe Data."**

<Kelas>

Kelas menunjukkan rentang penggunaan dan aplikasi dari sebuah label.

Kelas yang dapat dipilih bervariasi sesuai dengan tipe labelnya.

Kelas	Kegunaan	Area pengaturan label		
		Label global	lokal program Label	Blok fungsi label lokal
VAR_GLOBAL	Label umum yang dapat digunakan oleh program dan blok fungsi pada sebuah proyek	○	×	×
VAR_GLOBAL_CONSTANT	Label umum dengan konstanta yang dapat digunakan oleh program dan blok fungsi pada sebuah proyek	○	×	×
VAR	Label yang dapat digunakan oleh program dan blok fungsi yang dimaksudkan untuk label tersebut	×	○	○
VAR_CONSTANT	Label dengan konstanta yang dapat digunakan oleh program dan blok fungsi yang dimaksudkan untuk label tersebut	×	○	○
VAR_RETAIN	Label (jenis selot) yang dapat digunakan oleh program dan blok fungsi yang dimaksudkan untuk label tersebut	×	○	○
VAR_INPUT	Label yang digunakan untuk input blok fungsi yang dimaksudkan untuk label tersebut * Nilai ini tidak dapat diubah pada komponen program.	×	×	○
VAR_OUTPUT	Label yang digunakan untuk output blok fungsi yang dimaksudkan untuk label tersebut	×	×	○
VAR_IN_OUT	Label yang digunakan untuk input dan output blok fungsi yang dimaksudkan untuk label tersebut * Nilai ini dapat diubah pada komponen program.	×	×	○

2.1.2

Tipe Aplikasi Label dan Nilai yang Disimpan

<Tipe data>

Tipe data mengacu pada tipe nilai yang disimpan dalam label.

Tipe data yang ditetapkan ke label menunjukkan tipe dan rentang nilai yang dapat disimpan di label dan perangkat yang sesuai.

Tipe data yang dapat digunakan dengan program ladder dicantumkan di bawah ini.

Tipe data	Deskripsi	Panjang bit	Rentang nilai disimpan
Bit	ON atau OFF disimpan. Sesuai dengan perangkat "M"	1 bit	1: ON, 0: OFF
Kata (ditandai)	Bilangan bulat tanpa pecahan disimpan.	16 bit	-32768 hingga 32767
Kata ganda (ditandai)	Sesuai dengan perangkat "D"	32 bit	-2147483648 hingga 2147483647
FLOAT (presisi tunggal)	Bilangan asli yang mencakup pecahan disimpan.	16 bit	-2^{128} hingga -2^{-126} , 0, 2^{-126} hingga 2^{128}
FLOAT (presisi ganda)	Sesuai dengan perangkat "D"	32 bit	-2^{1024} hingga -2^{-1022} , 0, 2^{-1022} hingga 2^{1024}
String	String karakter disimpan. Sesuai dengan perangkat "D"	Variabel	Hingga 255 karakter
Pengatur waktu	NYALA bila waktu yang ditentukan tercapai. Sesuai dengan perangkat pengatur waktu "T"	-	-
Pengatur waktu retentif	NYALA bila waktu yang ditentukan tercapai. Sesuai dengan perangkat pengatur waktu retentif "ST"	-	-
Penghitung	NYALA bila jumlah yang ditentukan tercapai. Sesuai dengan perangkat penghitung "C"	-	-
Penunjuk	Posisi mulai sub-rutin disimpan. Sesuai dengan perangkat pointer "P"	-	-

2.1.3 Pendaftaran Label



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

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

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

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				...

Selesaikan pengaturan untuk pendaftaran label.
 Klik untuk melanjutkan.

Navigation icons: Home, Play, Previous, Next, Volume

2.1.4

Penetapan Label ke Perangkat secara Otomatis

Produktivitas

Kualitas

Mengonversi program secara otomatis akan menetapkan perangkat yang tepat ke label sesuai dengan kelas dan tipe datanya.

Ketika menggunakan label, tidaklah perlu untuk mengetahui perangkat mana yang ditetapkan.

Gunakan **"Device/Label Automatic-Assign Setting"** (Pengaturan Penetapan Perangkat/Label secara Otomatis) untuk mengubah rentang perangkat yang ditetapkan ke label.

* Layar ini adalah jendela pengaturan penetapan perangkat secara otomatis untuk MELSEC seri-Q dan MELSEC seri-L. Layar ini mungkin berbeda untuk MELSEC seri-F.

<Memulai "Device/Label Automatic-Assign Setting" (Pengaturan Penetapan Perangkat/Label secara Otomatis)>

Dari menu GX Works2, pilih **[Tool]** (Alat) – **[Device/Label Automatic-Assign Setting]** (Pengaturan Penetapan Perangkat/Label secara Otomatis).

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

- 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.
- 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 Mengatur Blok Ladder yang Digunakan Berulang sebagai Blok Fungsi untuk Diversi

Produktivitas

Kualitas

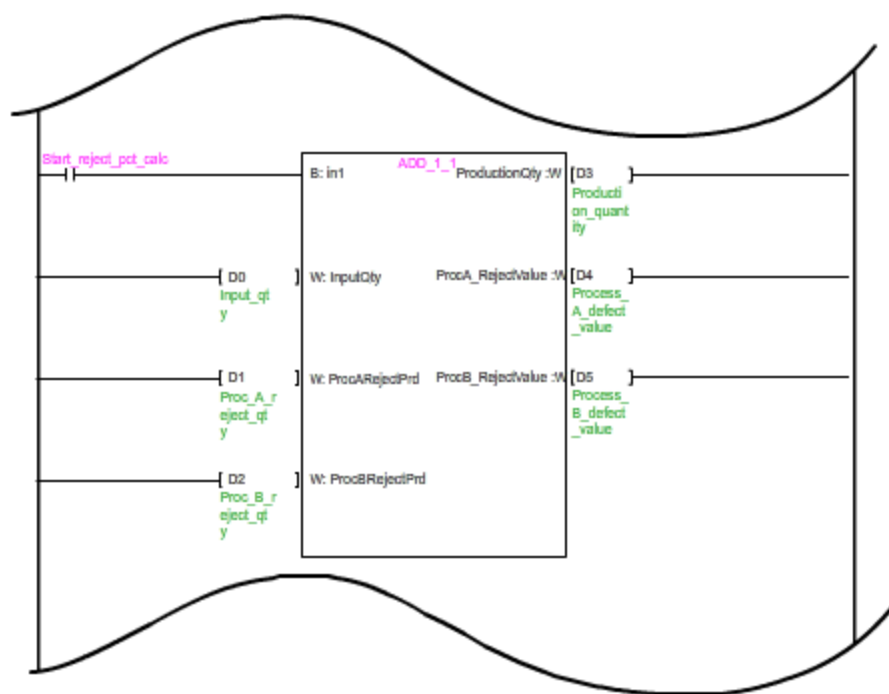
Pada program skala besar, beberapa blok ladder mungkin digunakan secara berulang. Atau, sebagian dari blok ladder yang sama mungkin digunakan pada program berbeda. Efisiensi kerja tidak dapat ditingkatkan jika Anda harus memasukkan blok ladder yang sama setiap kali diperlukan. Selain itu, jika ditemukan cacat pada blok ladder umum, setiap blok harus ditemukan dan dikoreksi. Hal ini mengakibatkan hilangnya waktu secara signifikan.

Gunakan "**Function Block (FB)**" (Blok Fungsi) untuk menyelesaikan masalah ini.

Blok ladder yang digunakan berulang dapat **diatur sebagai blok fungsi**, yang dapat digunakan di setiap program. Hal ini meningkatkan efisiensi kerja secara signifikan.

Jika ditemukan cacat pada blok umum, Anda hanya perlu mengoreksi blok fungsinya.

Hal ini tidak hanya menghemat waktu, namun juga mencegah risiko tidak mengoreksi salah satu blok.



Tekan tombol untuk melihat animasi dari awal.



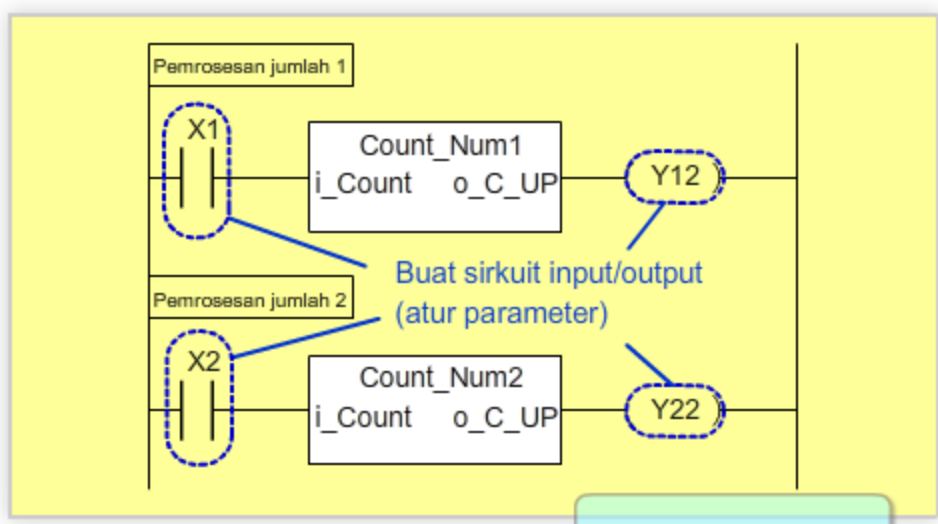
2.2.1

Membuat dan Menempatkan Blok Fungsi



Karena blok fungsi akan digunakan pada program berbeda, perangkat sebenarnya (seperti X, Y, dan D) tidak dapat digunakan pada blok fungsi. Perangkat ini harus diganti dengan **label** seperti yang telah Anda pelajari di Bagian 2.1. Prosedur untuk menata blok ladder ke dalam blok fungsi ditunjukkan pada animasi di bawah ini.

1. Siapkan program yang akan ditata ke dalam blok fungsi.
2. Bagi program menjadi input dan output, dan ganti perangkat internal dengan label internal untuk membuat blok fungsi.
3. Tempatkan blok fungsi yang telah dibuat pada program dengan tarik dan taruh.
4. Tetapkan parameter ke label input dan output untuk blok fungsi yang Anda tempatkan pada program.
5. Buat sirkuit input yang meneruskan parameter ke label input dan sirkuit output yang menerima parameter dari label output, sebelum dan sesudah blok fungsi.



Tekan tombol untuk kembali ke alur pertama.



2.2.1 Membuat dan Menempatkan Blok Fungsi



MELSOFT Series GX Works2 C:\Sequential Programs\le-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

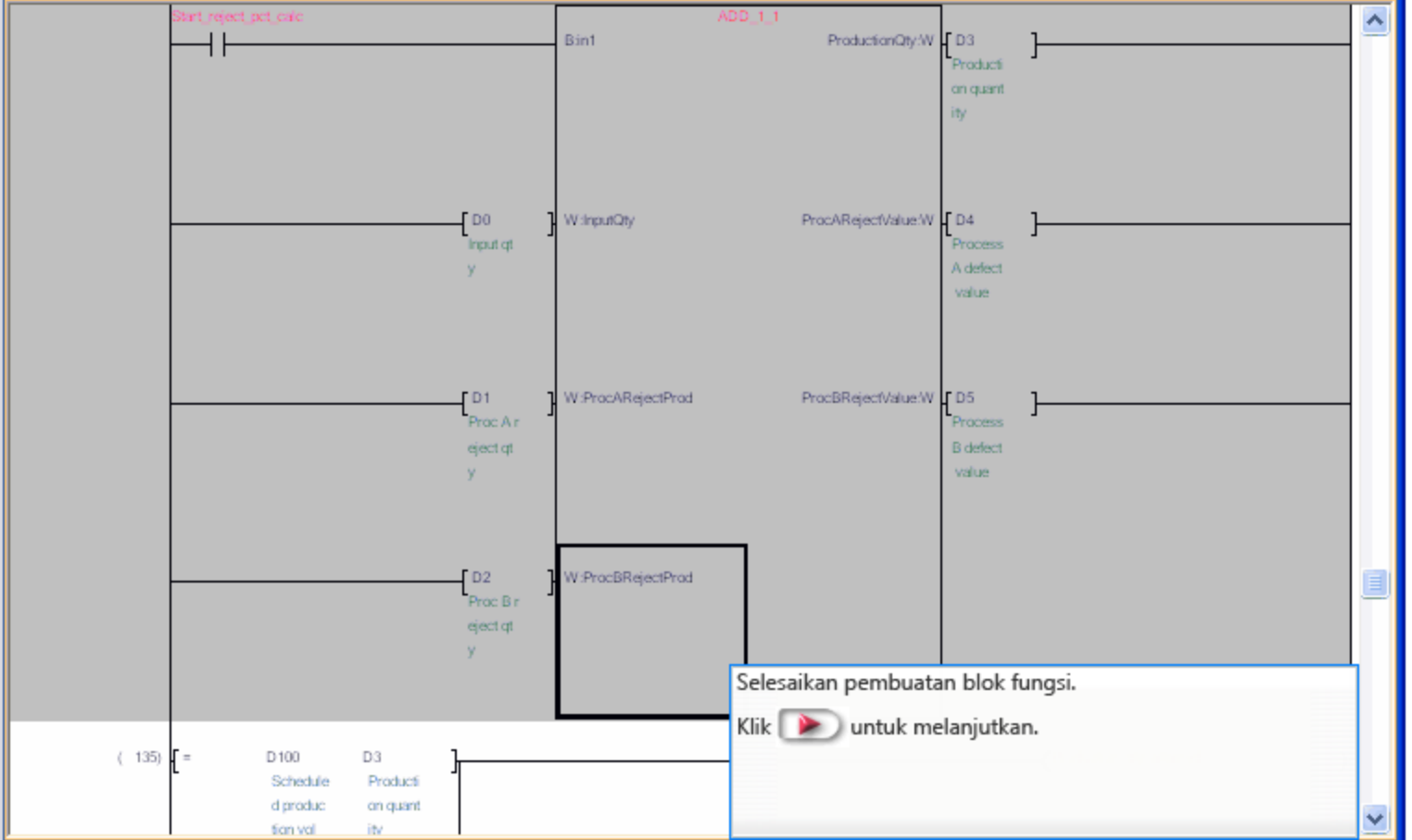
Project

- 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



Selesaikan pembuatan blok fungsi.
 Klik untuk melanjutkan.



2.2.2 Menggunakan Perpustakaan Blok Fungsi

Produktivitas

Kualitas

Sekarang Anda telah mempelajari cara membuat blok fungsi.

Mitsubishi Electric telah menyiapkan program kontrol untuk berbagai modul seperti CPU, modul input/output analog, jaringan, dan modul pemosisian sebagai blok fungsi, yang tersedia secara gratis sebagai **"Perpustakaan FB."**

Penggunaan perpustakaan FB akan meniadakan perlunya mengembangkan program kontrol modul, yang secara tradisional menjadi tanggung jawab pengguna. Bahkan modul yang tidak Anda kenal dengan baik pun dapat dimasukkan dengan mudah.

Perpustakaan FB dapat diunduh dari halaman **"MELSOFT Library Download"** (Pengunduhan Perpustakaan MELSOFT) di **MITSUBISHI ELECTRIC FA Website**.

<Sebelum>



<Jika Anda menggunakan perpustakaan FB>



* Perpustakaan FB tidak tersedia untuk MELSEC seri-F.

Model-model yang didukung dicantumkan di halaman unduhan. Harap periksa model yang didukung sebelum mengunduh.

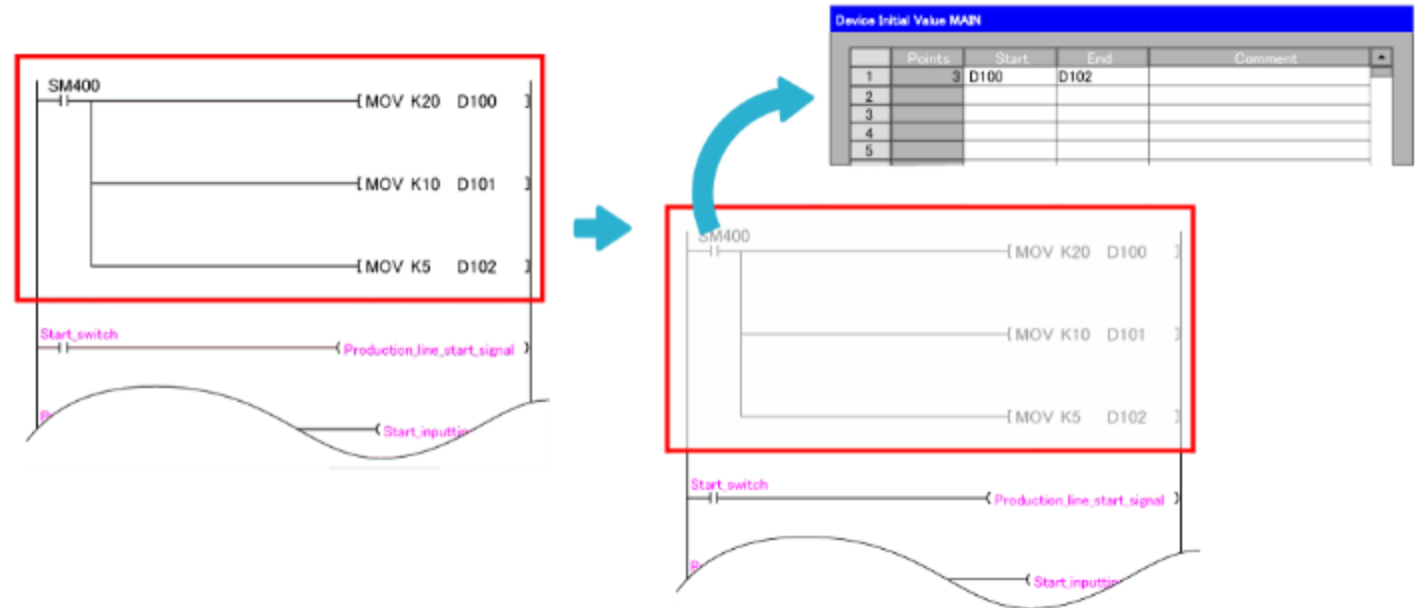
2.3 Mengubah Nilai Awal Perangkat tanpa Mengoreksi Program



Nilai awal atau konstanta perangkat biasanya diatur menggunakan instruksi MOV sebelum pemrosesan program utama. Dalam hal ini, program harus dikoreksi secara langsung setiap kali operasi program diubah sesuai dengan aplikasi sistem. Metode ini tidak hanya memboroskan waktu, tetapi juga berisiko menimbulkan kesalahan koreksi atau kegagalan membuat koreksi.

Gunakan **"Nilai Awal Perangkat"** untuk menyelesaikan masalah ini. Penggunaan fungsi GX Works2 memastikan pengelolaan nilai awal perangkat dengan tepat dan meniadakan perlunya membuat koreksi program, sehingga Anda dapat membuat program secara jauh lebih efisien. Selain itu, fungsi ini meniadakan perlunya program pengaturan nilai perangkat awal, yang akan mengurangi volume program (penggunaan memori) dan mengurangi risiko kegagalan.

Untuk nilai awal perangkat, tentukan rentang perangkat untuk mengatur nilai awal. Nilai awal aktual disimpan di **memori perangkat** dan ditetapkan ke rentang perangkat yang ditentukan. Setelah area memori perangkat disiapkan sesuai dengan aplikasi sistem, nilai awal perangkat dapat diubah hanya dengan mengubah area memori perangkat yang akan ditetapkan.



2.3 Mengubah Nilai Awal Perangkat tanpa Mengoreksi Program



MELSOFT Series GX Works2 C:\Sequential Programs\le-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]

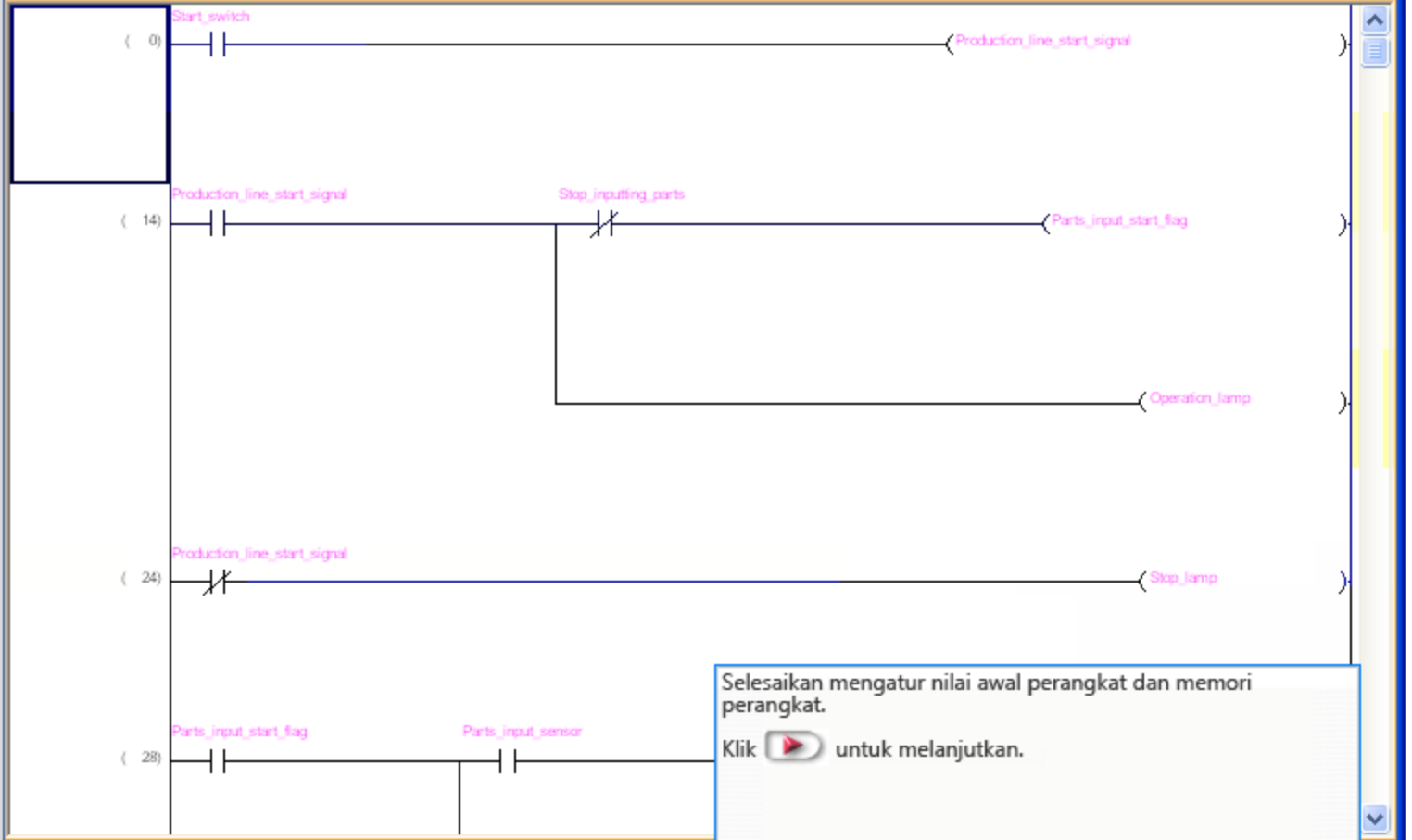
Project


- 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



Selesaikan mengatur nilai awal perangkat dan memori perangkat.
 Klik  untuk melanjutkan.

2.4 Menyederhanakan Program Ladder

Produktivitas

Kualitas

Pada program ladder skala besar, koneksi perangkat, instruksi, dan jalur yang diatur dapat menjadi sangat rumit, sehingga sulit dipahami pemrosesan apa yang dijalankan program.

Secara khusus, kalkulasi numerik, seperti rumus yang diselesaikan pada satu jalur, harus diprogram menggunakan kombinasi beberapa instruksi.

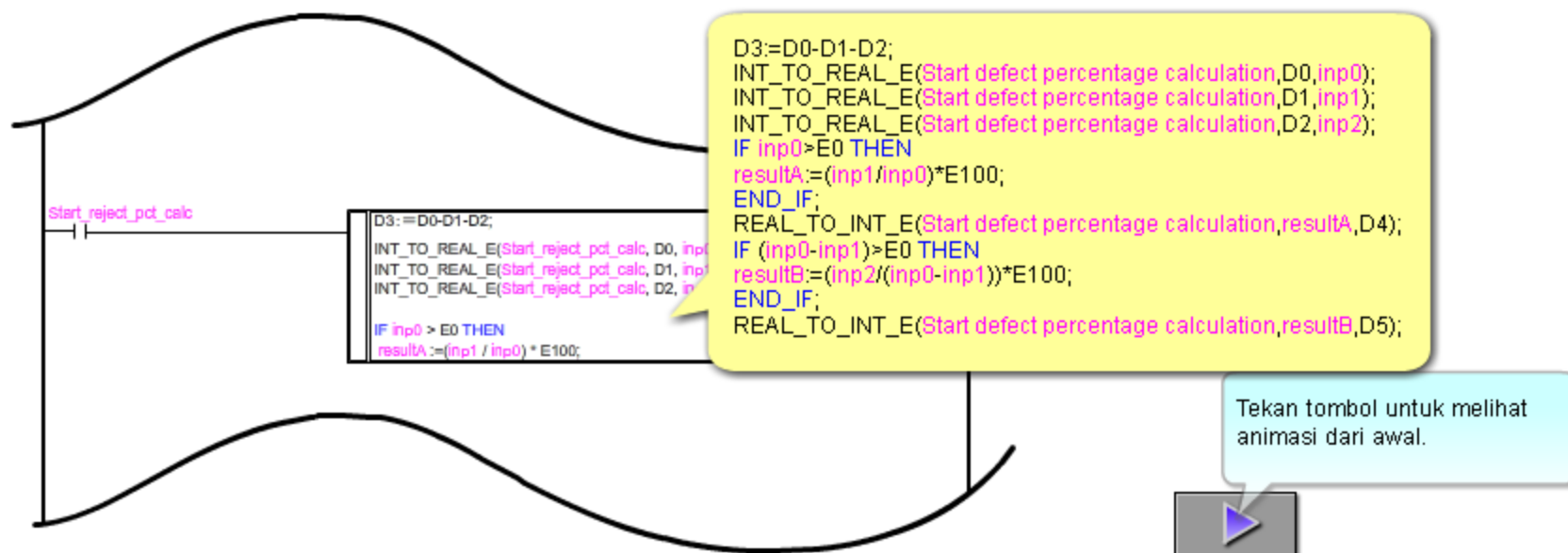
Gunakan **"Teks Berstruktur Sebaris"** untuk menyelesaikan masalah ini.

Program ladder diganti oleh program yang sebagian ditulis dalam **structure text (ST)** (bahasa teks struktur).

Bahasa ST adalah bahasa pemrograman kontrol sekuens yang mirip dengan bahasa C yang digunakan untuk pemrograman perangkat lunak komputer.

Kalkulasi numerik dapat ditulis menggunakan rumus, sehingga programmer yang tidak familier dengan C pun dapat menggunakan teks berstruktur sebaris.

Gambar di bawah ini menunjukkan contoh bagian program ladder untuk sistem yang diganti dengan teks berstruktur sebaris. Anda dapat melihat bahwa program ladder yang rumit sekarang mudah dipahami.



2.4 Menyederhanakan Program Ladder



MELSOFT Series GX Works2 C:\Sequential Programs\le-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 Global1, Local Label Setting MAIN [PRG]

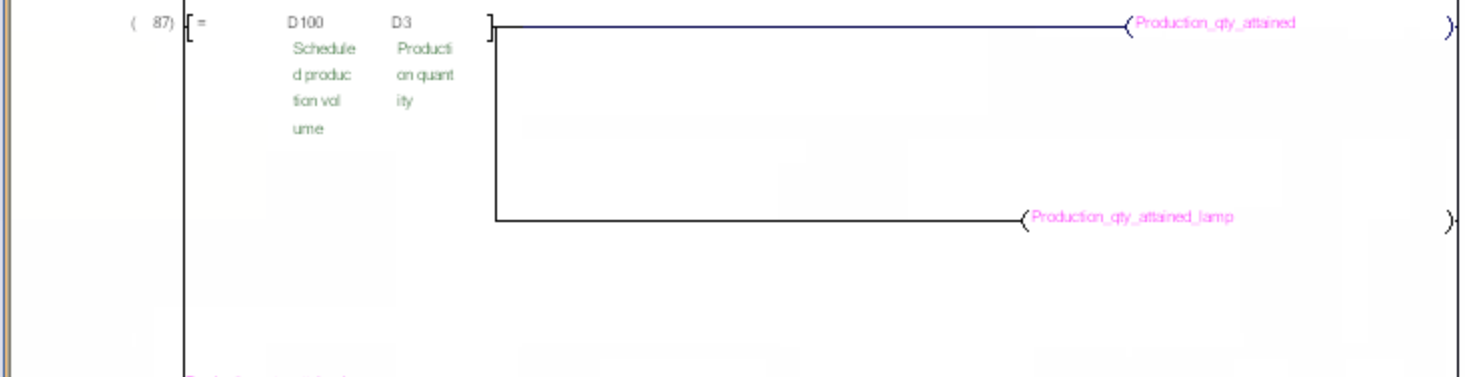
Project

- 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

```

( 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)

```



(94) Production_qty_ attained

Selesaikan pengaturan teks berstruktur sebaris.
Klik untuk melanjutkan.

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

2.5 Membuat Program yang Mudah Dipahami dan Dibaca

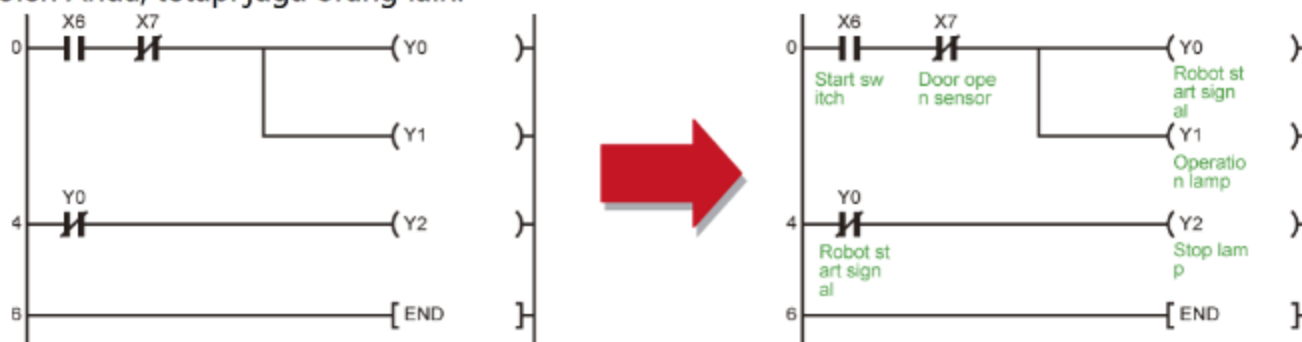
Anda mungkin kesulitan memahami detail kontrol pada program skala besar hanya dengan melihat program itu. Sebagai akibatnya, masalah berikut ini dapat terjadi:

- Anda membuat kesalahan program (seperti memasukkan instruksi atau perangkat yang salah).
- Anda tidak dapat menemukan penyebab kesalahan program.
- Seseorang yang mengambil alih pemrograman tidak dapat memahami detail kontrol.

Gunakan "**Komentar**" untuk menyelesaikan masalah ini.

Memo seperti informasi kontrol dan nama perangkat dapat ditambahkan ke program sehingga detail kontrol dapat dipahami dengan mudah.

Komentar ini harus dimasukkan di mana pun memungkinkan untuk membuat program yang mudah dipahami tidak hanya oleh Anda, tetapi juga orang lain.



GX Works2 memungkinkan komentar berikut ini dimasukkan.

Tipe komentar	Rentang komentar
Komentar perangkat	Komentar dapat ditambahkan ke perangkat. Komentar ini menunjukkan aplikasi setiap perangkat dan tipe perangkat I/O yang terhubung.
Pernyataan	Komentar dapat ditambahkan ke blok ladder. Komentar ini menjadikan alur pemrosesan mudah dipahami.
Catatan	Komentar dapat ditambahkan ke kumpulan/instruksi aplikasi pada program. Komentar ini membuat isi kumpulan (output) dan instruksi aplikasi mudah dipahami.

2.5.1

Menulis Komentar untuk Setiap Blok Ladder

Produktivitas

Kualitas

MELSOFT Series GX Works2 C:\Sequential Programs\le-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

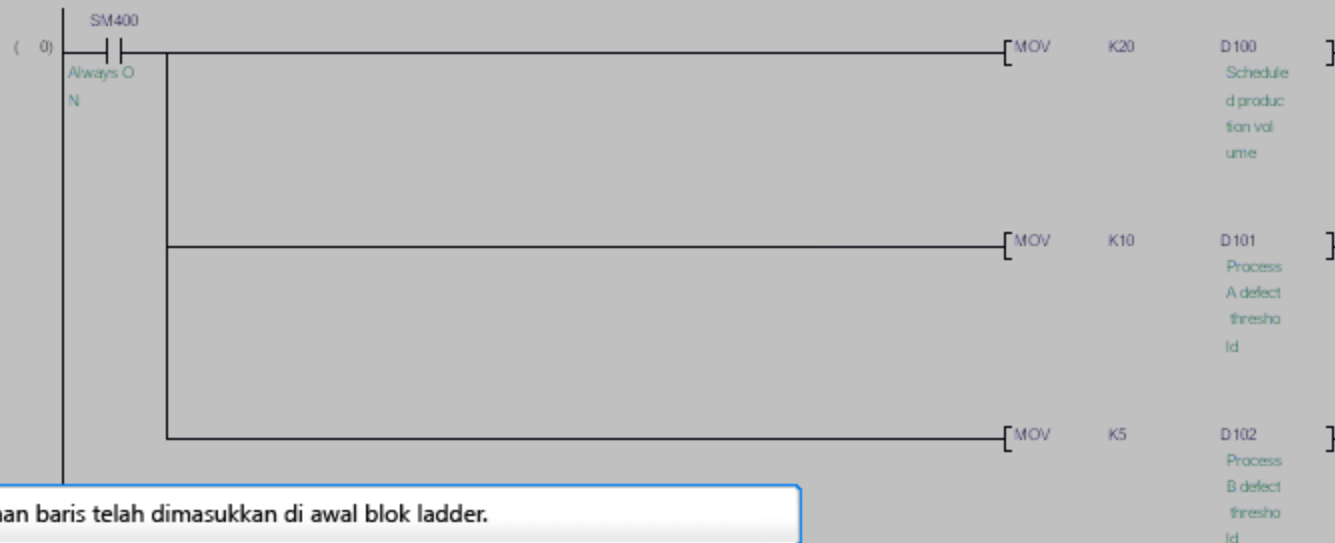
Project

User Library

Connection Destination

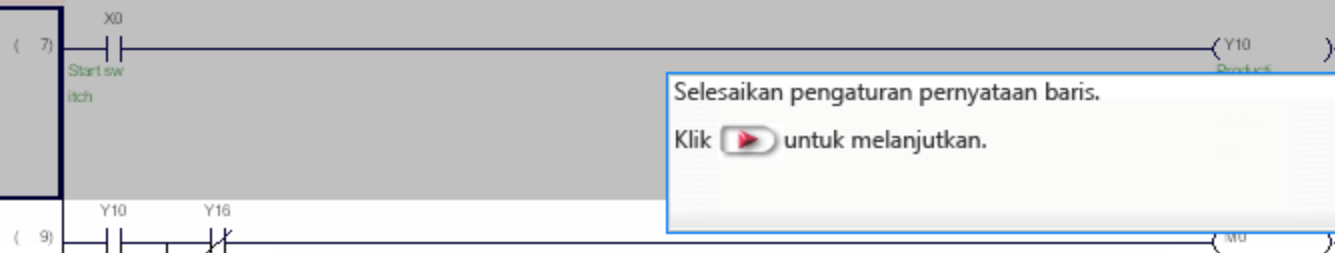
[PRG]Write MAIN 145 Step

Scheduled prod qty setting



Pernyataan baris telah dimasukkan di awal blok ladder.

Prod line start



Selesaikan pengaturan pernyataan baris.

Klik  untuk melanjutkan.

English

Simple

Q03UDE

Host Station

(7/ 145)

2.5.2 Menulis Komentar untuk Setiap Output (Kumparan, Instruksi Aplikasi)



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

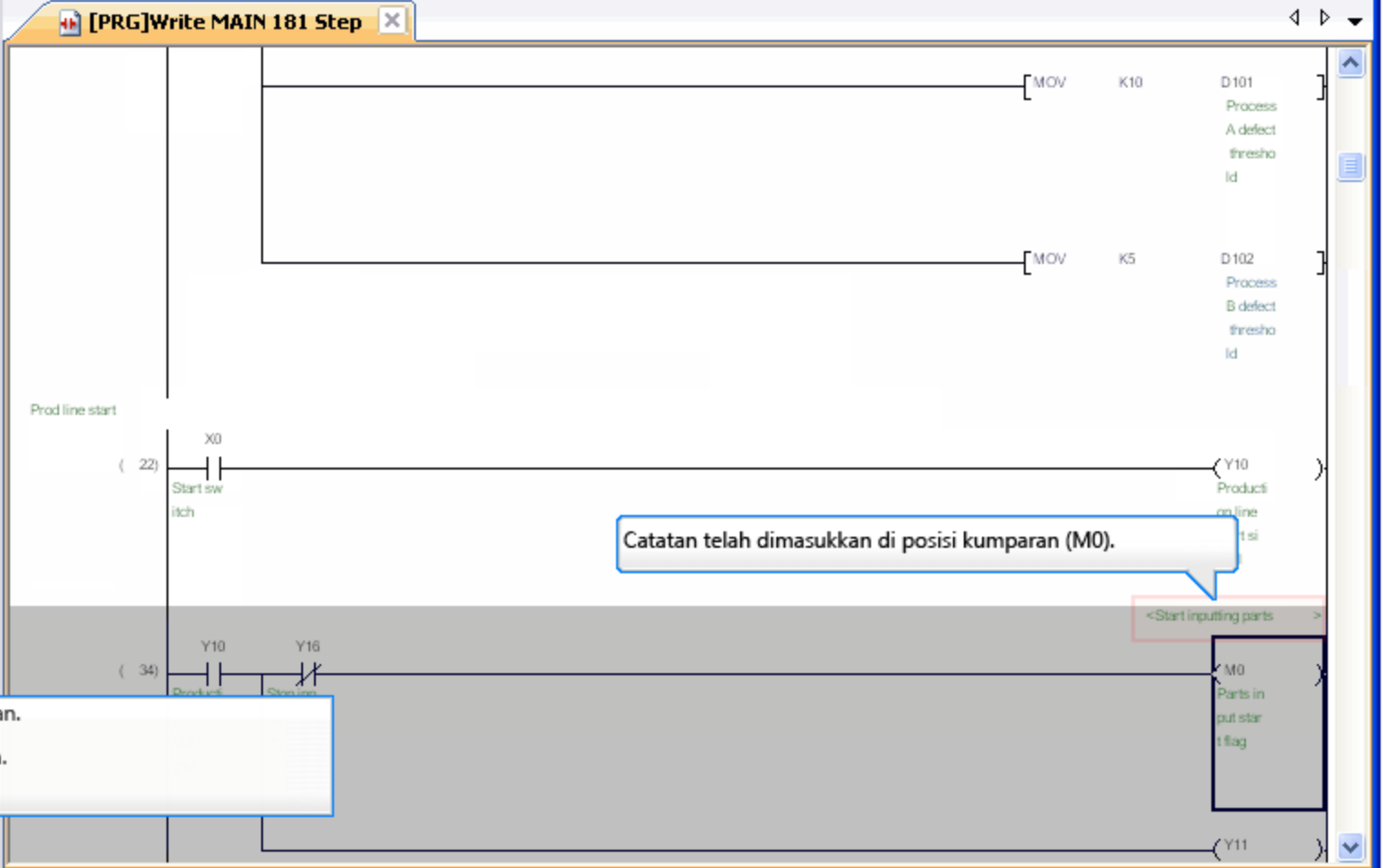
F5 SF5 F6 SF6 F7 F8 F9 SF9 CF9 CF10 SF7 SF8 AF7 AF8 SAF5 SAF6 SAF7 SAF8 AF5 CAF5 CAF10 F10 AF9

Navigation [PRG]Write MAIN 181 Step

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



Catatan telah dimasukkan di posisi kumparan (M0).

Selesaikan pengaturan catatan.
Klik [Play] untuk melanjutkan.

Navigation icons: Home, Play, Stop, Previous, Next

2.6

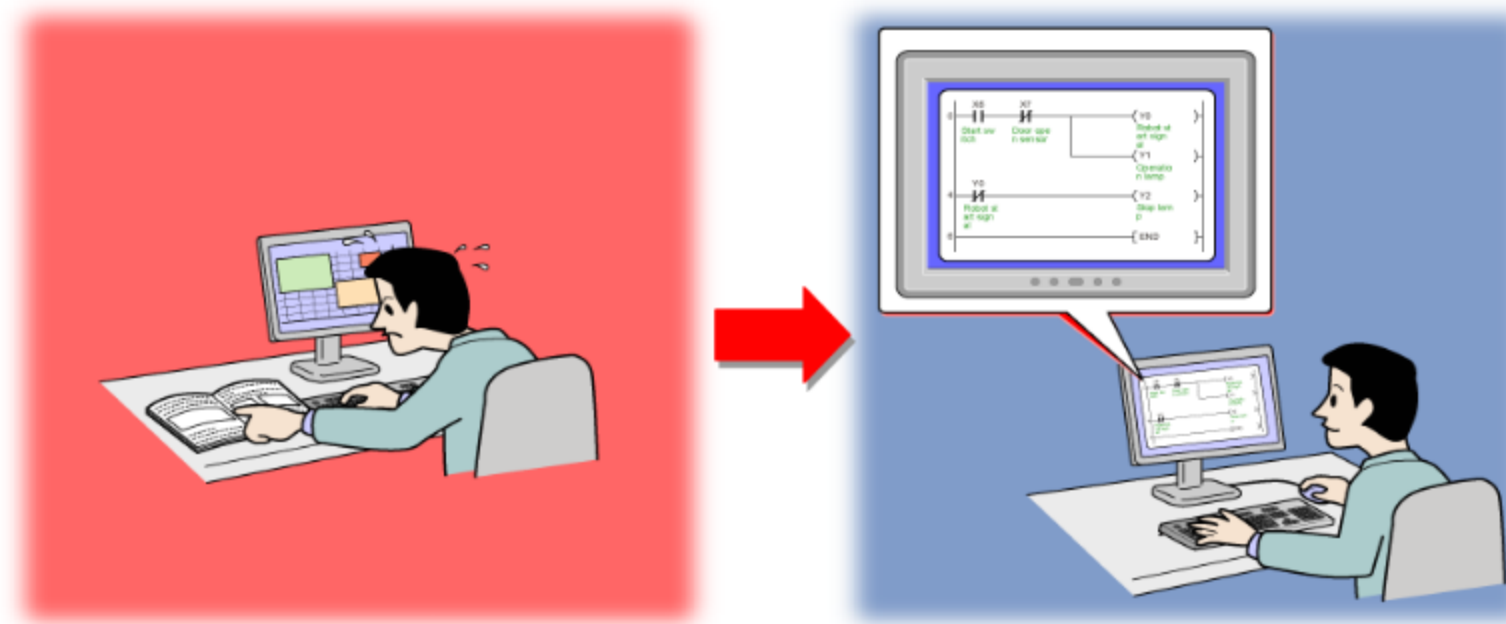
Memudahkan Membaca Program yang Mengandung Relai/Register Khusus

Jika relai khusus, register khusus, dan/atau perangkat modul fungsi cerdas digunakan dalam program, semua aplikasi dan fungsi perangkat tersebut terkadang sulit dipahami. Anda harus membaca program sambil merujuk panduan. Meskipun program akan lebih mudah dipahami jika komentar ditambahkan ke setiap perangkat, perlu waktu dan upaya lumayan untuk menambahkan komentar jika banyak perangkat digunakan.

Gunakan "**Komentar Contoh**" untuk menyelesaikan masalah ini.

GX Works2 menyediakan komentar contoh yang menguraikan aplikasi dan fungsi relai khusus, register khusus, dan perangkat modul fungsi cerdas.

Dengan komentar sampel ini, komentar akan lebih mudah ditambahkan ke perangkat sehingga program mudah dipahami. Komentar sampel dapat dimodifikasi bila diperlukan.



2.6 Memudahkan Membaca Program yang Mengandung Relai/Register Khusus

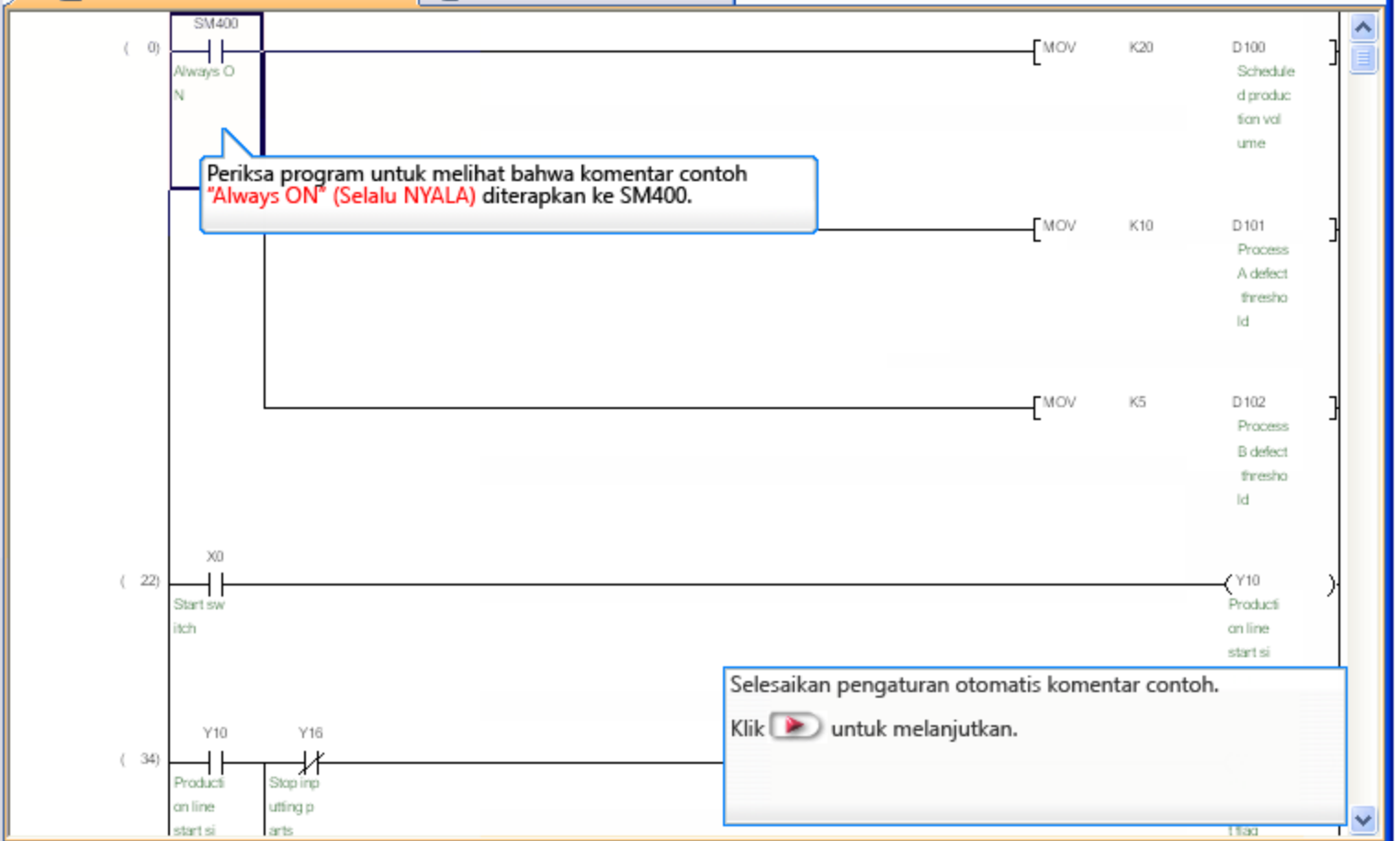
MELSOFT Series GX Works2 C:\Sequential Programs\learning\Robot control - [[PRG]Write MAIN 194 Step]

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

Navigation [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



Periksa program untuk melihat bahwa komentar contoh "Always ON" (Selalu NYALA) diterapkan ke SM400.

Selamatkan pengaturan otomatis komentar contoh. Klik untuk melanjutkan.

Bab 3 Debugging

Langkah-langkah pembelajaran pada Bab 3

Pada Bab 3, Anda akan mempelajari fungsi-fungsi yang digunakan untuk debugging. GX Works2 menyediakan banyak alat pemantauan dan debugging untuk mengoreksi kesalahan (bug). Buatlah program yang berkualitas dan bebas kesalahan dengan alat pemantauan dan debugging ini.

- 3.1 Memantau Perangkat dan Label Target Saja
- 3.2 Memeriksa Status Penggunaan Perangkat dan Label
- 3.3 Mengumpulkan Informasi Perubahan Nilai Perangkat dari Waktu ke Waktu
- 3.4 Mengubah Nilai Perangkat tanpa Mengoreksi Program
- 3.5 Operasi Program Debugging Langkah demi Langkah
 - 3.5.1 Menggunakan Fungsi Eksekusi Langkah
- 3.6 Mensimulasikan Operasi Perangkat Eksternal
 - 3.6.1 Memasukkan Nilai Perangkat untuk Mengatur Sistem I/o
 - 3.6.2 Menggunakan Format Diagram Waktu untuk Mengatur Sistem I/o

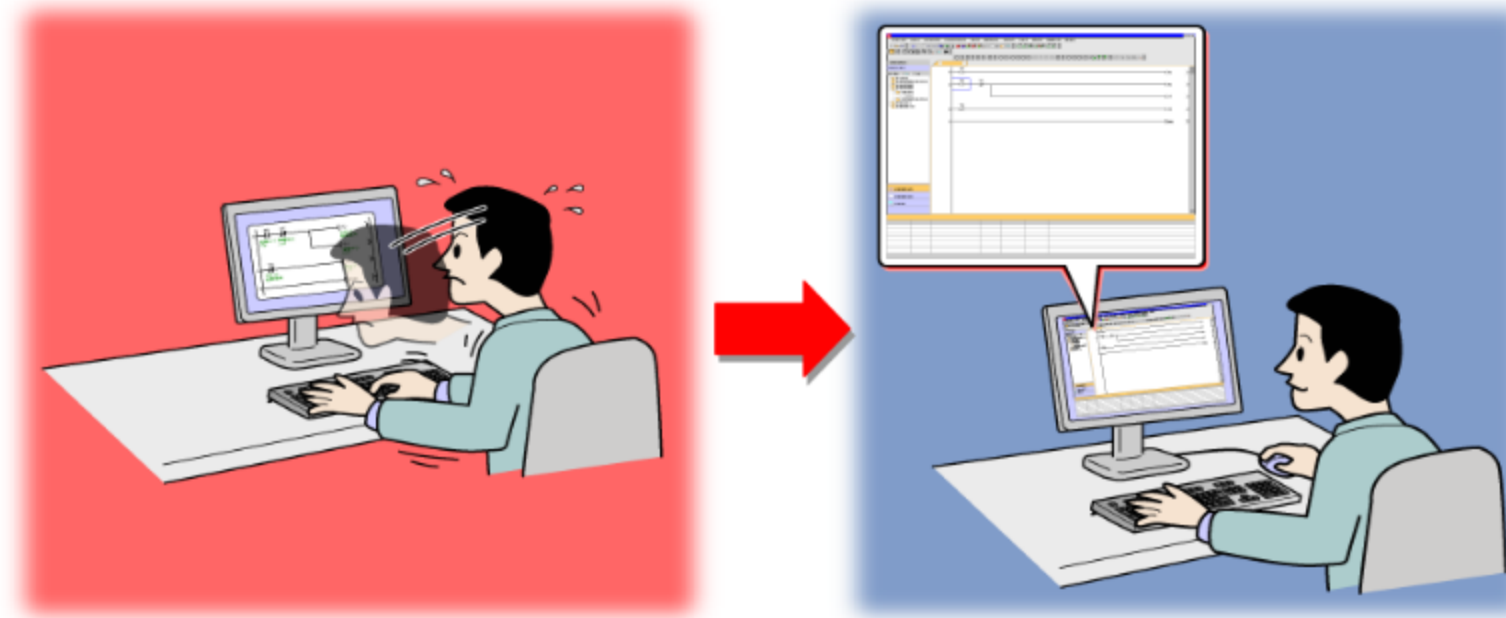
3.1

Memantau Perangkat dan Label Target Saja

Kualitas

Sebuah program menggunakan banyak instruksi dan perangkat. Selain itu, program yang panjang membuat layar monitor PC hanya dapat menampilkan sebagian program pada satu waktu. Karena itu, monitor ladder saja tidak dapat memantau seluruh operasi.

Gunakan "**Pengawasan**" untuk menyelesaikan masalah ini. Fungsi ini dapat digunakan untuk memantau hanya perangkat dan label yang telah Anda tetapkan di muka. Dua Pengawasan atau lebih dapat dibuat untuk memastikan bahwa setiap rentang dipantau.



3.1 Memantau Perangkat dan Label Target Saja



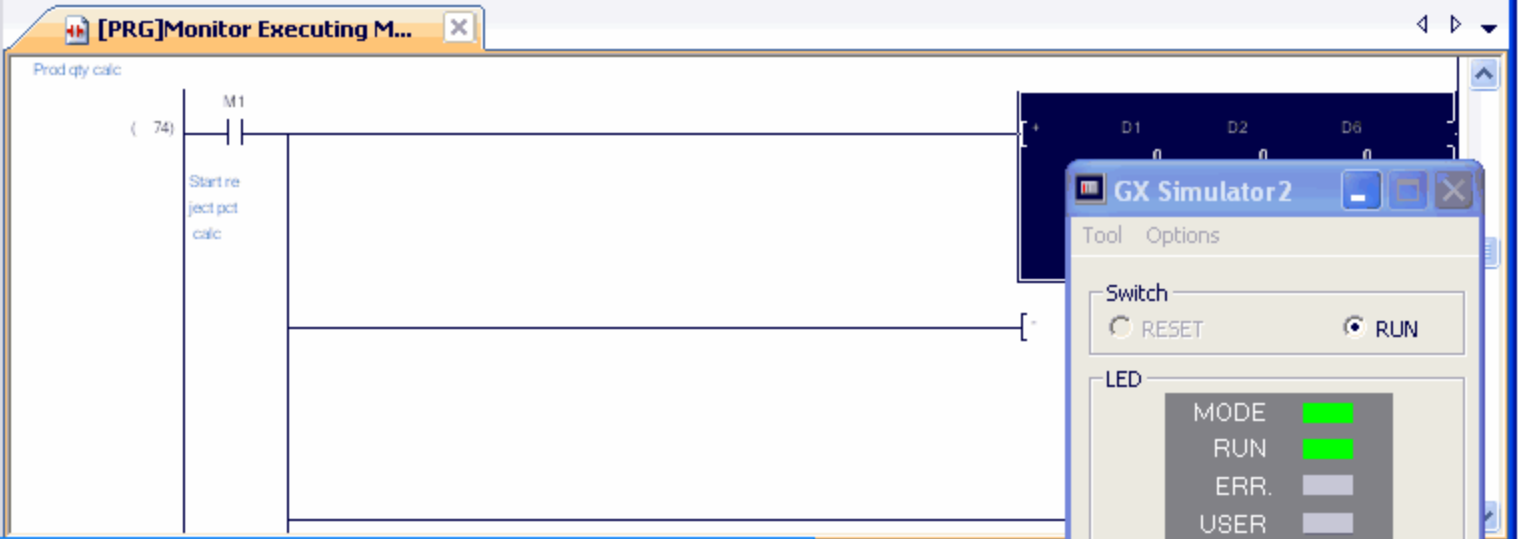
MELSOFT Series GX Works2 C:\Sequential Programs\le-learning\Robot control - [[PRG]Monitor Executing MAIN (Read Only) 194 Step

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

Navigation

Project

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



GX Simulator 2

Tool Options

Switch

RESET RUN

LED

- MODE
- RUN
- ERR.
- USER

Watch 1 Nilai yang ada untuk perangkat yang terdaftar di Watch 1 (Pengawasan 1) dipantau.

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	

Selesaikan pendaftaran dan pemantauan Watch (Pengawasan).
Klik untuk melanjutkan.

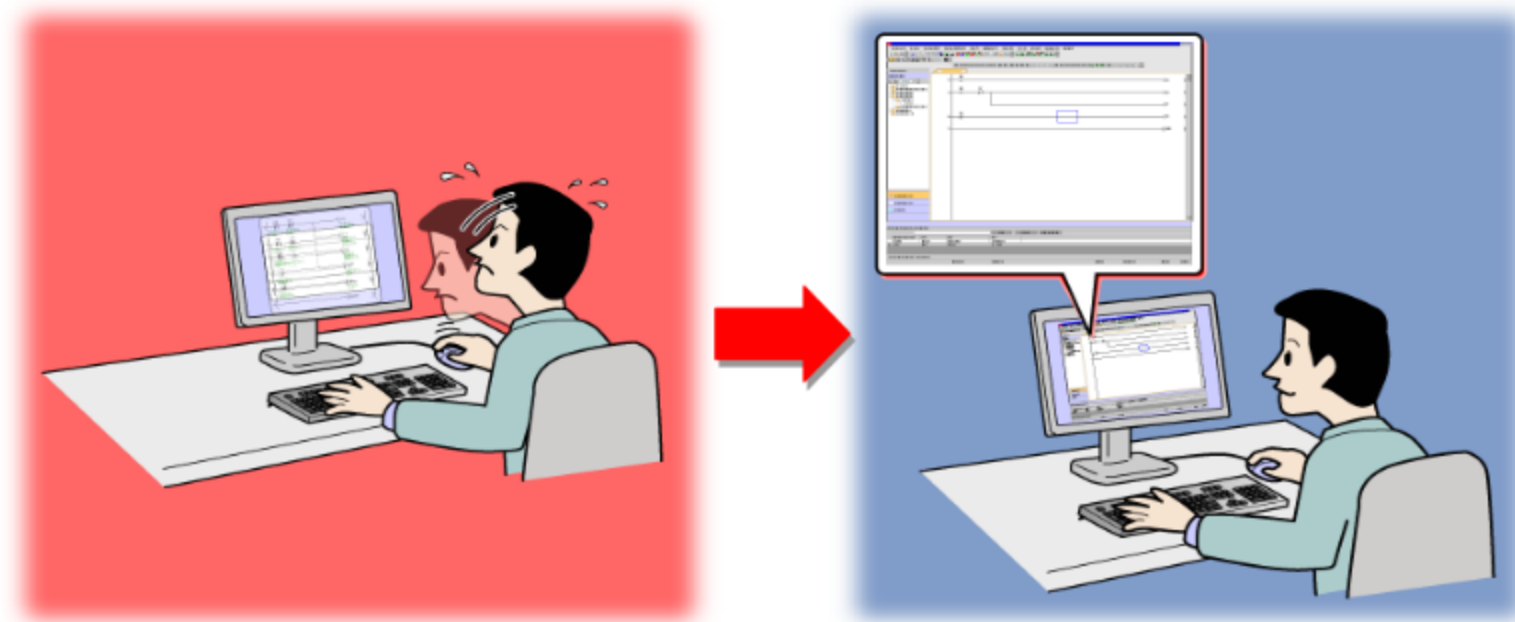
3.2

Memeriksa Status Penggunaan Perangkat dan Label

Kualitas

Sebuah program menggunakan perangkat dan label yang sama di lokasi berbeda. Anda mungkin ingin memeriksa status penggunaan perangkat dan label tersebut dengan membandingkannya di beberapa lokasi.

Gunakan "**Referensi silang**" untuk melakukannya. Fungsi ini mencantumkan lokasi perangkat dan label yang cocok dengan kondisi pencarian sehingga Anda dapat membandingkannya dan memeriksa apakah ada penggunaan yang salah.



3.2

Memeriksa Status Penggunaan Perangkat dan Label

Kualitas

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

Project

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"

Selesaikan penggunaan Cross Reference (Referensi Silang).

Klik  untuk melanjutkan.

English

Simple

Q03UDE

Host Station

(84 M...

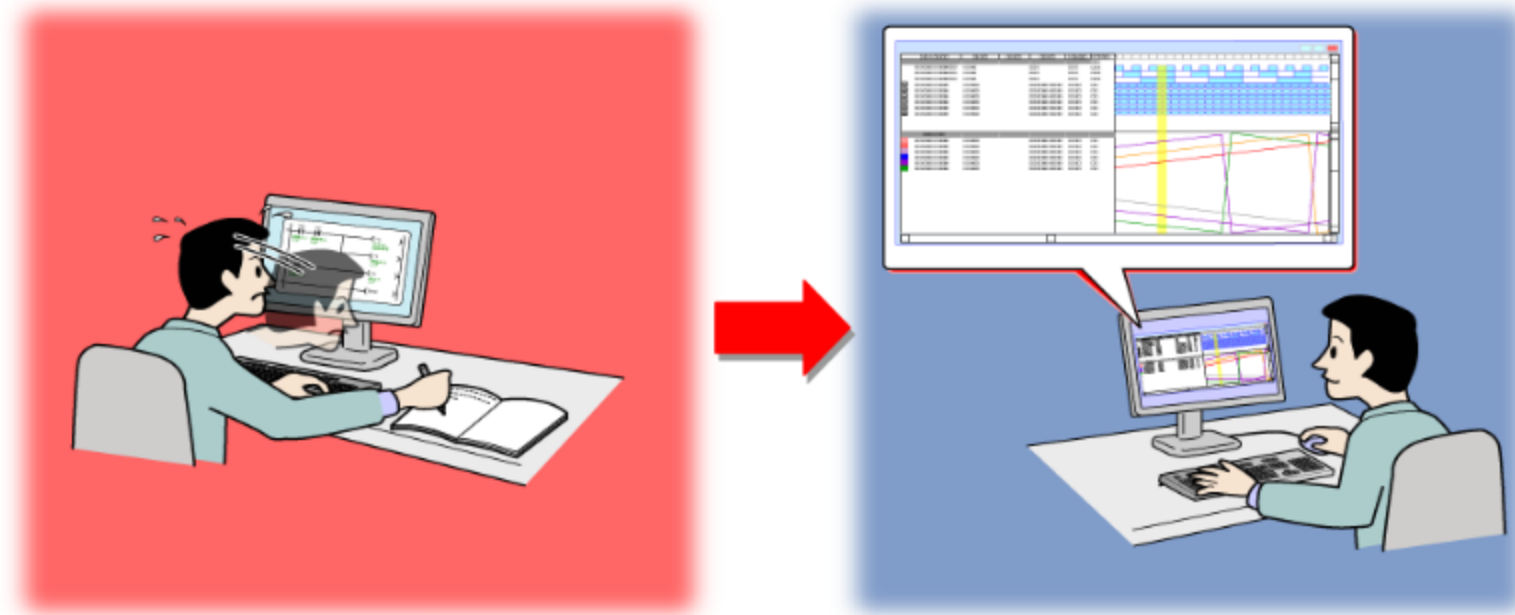
3.3

Mengumpulkan Informasi Perubahan Nilai Perangkat dari Waktu ke Waktu

Kualitas

Anda mungkin ingin mengonfirmasi bahwa perubahan pada nilai perangkat dan label masih berada dalam rentang desain atau menjalankan pemecahan masalah dengan memeriksa perubahan yang terjadi ketika ada kegagalan.

Dalam kasus ini, gunakan "**Pelacakan Sampling.**" (Hanya untuk MELSEC seri-Q dan MELSEC seri-L)
Fungsi ini dapat digunakan untuk memantau dan merekam perubahan pada nilai perangkat dan label dari waktu ke waktu. Rekaman tersebut dapat disimpan sebagai file.



3.4

Mengubah Nilai Perangkat tanpa Mengoreksi Program

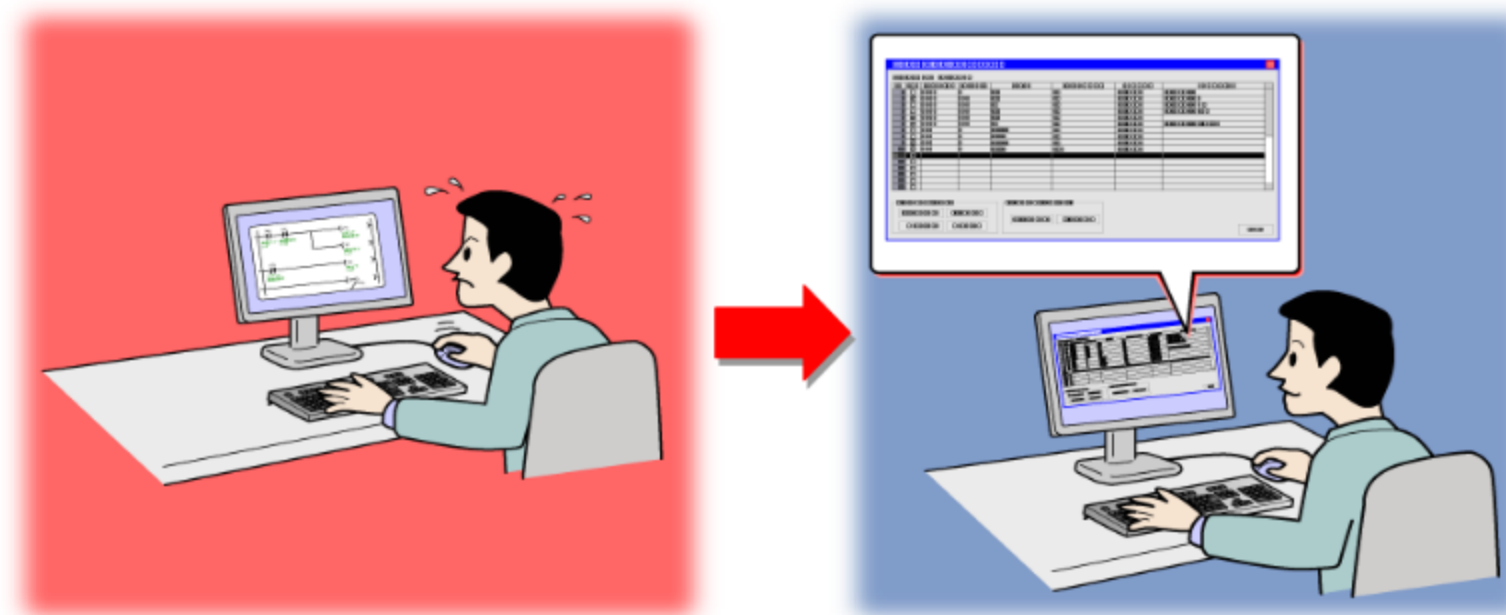
Kualitas

Ketika menjalankan debugging, Anda mungkin ingin mengubah nilai perangkat secara paksa untuk memeriksa perbedaan pada operasi program.

Namun, itu berarti Anda harus memodifikasi program setiap kali nilai perangkat diubah, yang memerlukan waktu dan upaya besar.

Selain itu, jika Anda lupa mengembalikan program yang telah dimodifikasi ke pengaturan aslinya, kegagalan lain dapat terjadi.

Dalam kasus ini, gunakan **"Uji Perangkat Terkondisi Eksekusi."** (Hanya untuk MELSEC seri-Q dan MELSEC seri-L) Fungsi ini mengubah nilai perangkat setelah nomor langkah yang ditentukan di muka dieksekusi, tanpa harus memodifikasi program.



3.4 Mengubah Nilai Perangkat tanpa Mengoreksi Program



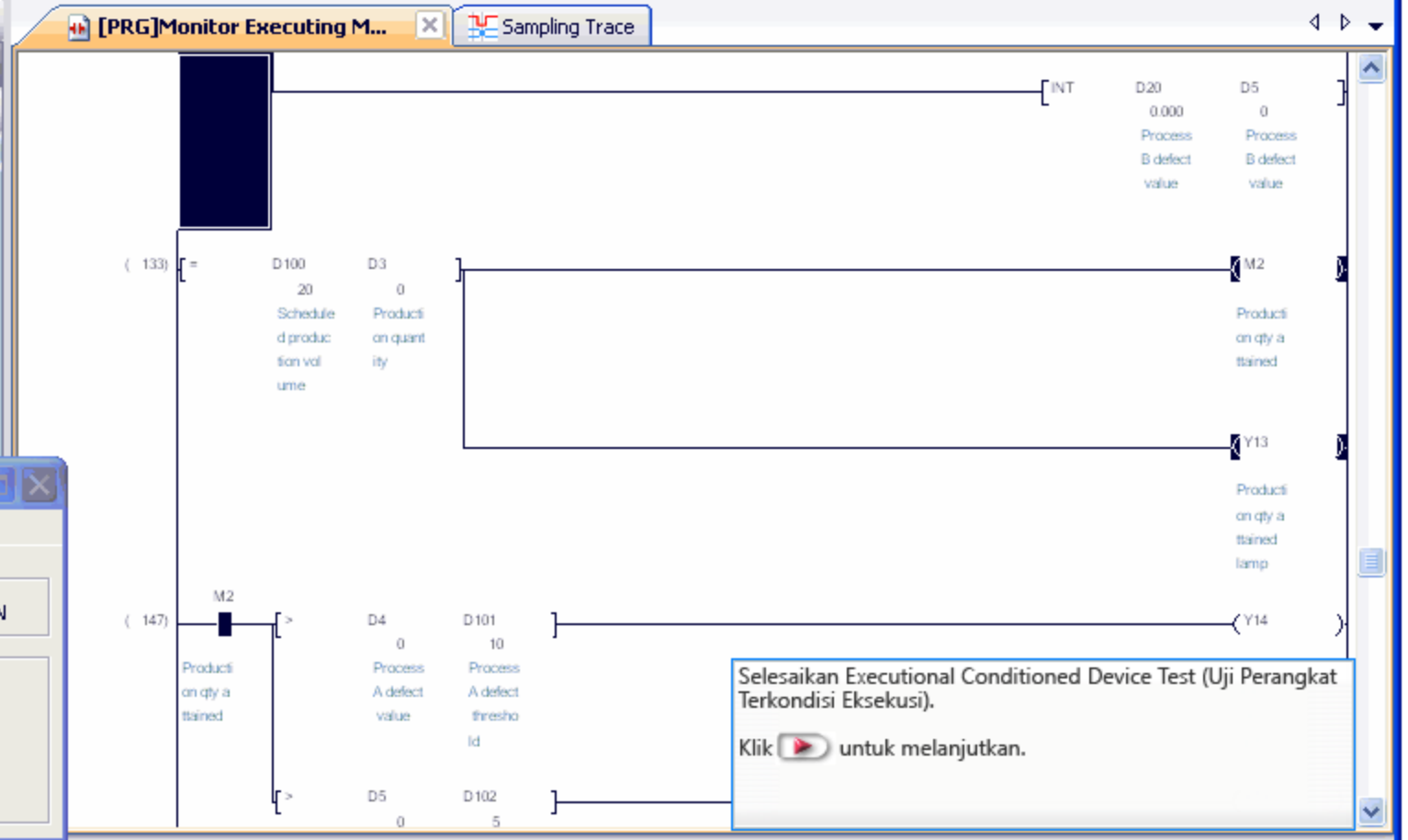
MELSOFT Series GX Works2 C:\Sequential Programs\le-learning\Robot control - [[PRG]Monitor Executing MAIN (Read Only) 194 Step

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

Toolbar: File operations, navigation, and execution controls.

Navigation 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:

Selesaikan Executional Conditioned Device Test (Uji Perangkat Terkondisi Eksekusi).
 Klik untuk melanjutkan.

3.5 Operasi Program Debugging Langkah demi Langkah

Kualitas

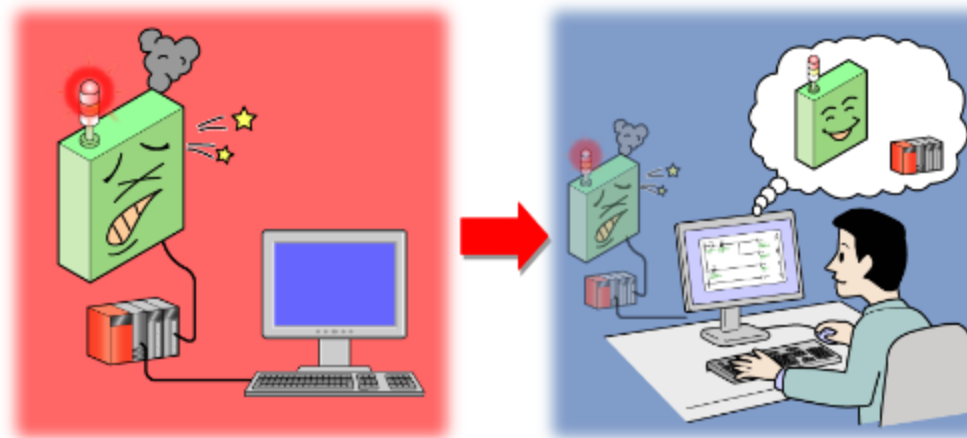
Selama debugging, Anda mungkin ingin mengonfirmasi eksekusi instruksi pada setiap langkah atau memeriksa apakah ada perubahan nilai perangkat.

Namun, debugging langkah demi langkah dapat sulit dilakukan karena cepatnya eksekusi program (waktu pindai).

Gunakan "**Fungsi Eksekusi Langkah**" untuk menyelesaikan masalah ini. (Hanya untuk MELSEC seri-Q dan MELSEC seri-L) Fungsi ini mengaktifkan eksekusi program satu per satu, yang berarti menerapkan debugging langkah demi langkah. Gunakan fungsi eksekusi langkah bersama dengan **fungsi simulasi**. (Fungsi eksekusi langkah tidak dapat digunakan untuk debugging pada PLC sebenarnya.)

Fungsi berikut ini dapat digunakan oleh fungsi eksekusi langkah.

Fungsi	Deskripsi
Eksekusi jeda	Fungsi ini mengeksekusi program hingga kondisi jeda yang ditentukan dipenuhi. Eksekusi program dihentikan setelah kondisi jeda terpenuhi. Gunakan titik jeda dan perangkat jeda untuk menentukan kondisi jeda.
Eksekusi langkah	Fungsi ini mengeksekusi program langkah demi langkah.
Eksekusi parsial	Fungsi ini mengeksekusi program hanya dari lokasi yang ditentukan.



3.5.1 Menggunakan Fungsi Eksekusi Langkah

Tentukan **break point** dan **break device** berturut-turut sebagai lokasi awal debugging dan kondisi start. Anda juga dapat menentukan **rentang lompat yang** di dalam rentang itu eksekusi program dihindari untuk sementara. (Hanya untuk MELSEC seri-Q dan MELSEC seri-L)

Bila kondisi jeda terpenuhi setelah **eksekusi jeda** dimulai, eksekusi program akan disela. Setelah itu, sambil mengeksekusi operasi program langkah demi langkah dengan **fungsi eksekusi langkah**, periksa apakah ada perubahan nilai perangkat untuk menemukan kegagalan.

<Titik jeda>

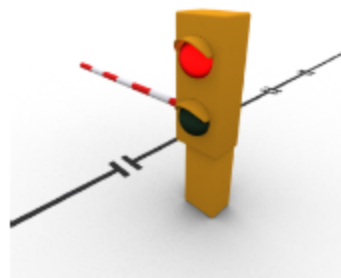
Atur titik jeda di mana Anda ingin menyela eksekusi program. Tentukan dalam satuan langkah. Hingga 64 titik jeda dapat diatur di seluruh proyek.

<Perangkat jeda>

Atur perangkat jeda berdasarkan eksekusi program mana yang disela ketika nilai perangkat atau label memenuhi kondisi yang ditentukan. Hingga 16 perangkat bit dan/atau kata dapat diatur.

<Rentang lompat>

Atur rentang yang di dalamnya program tidak boleh dieksekusi, dalam satuan blok ladder, selama eksekusi langkah. Hingga 16 rentang dapat ditentukan di seluruh proyek.



<Aplikasi Rentang lompat>

Titik kegagalan dapat dipersempit menggunakan **Rentang lompat**.

Jalankan eksekusi jeda dengan dan tanpa mengatur Rentang lompat.

Jika kegagalan terjadi hanya ketika pengaturan skip range tidak diterapkan, berarti rentang itu mencakup kegagalan.

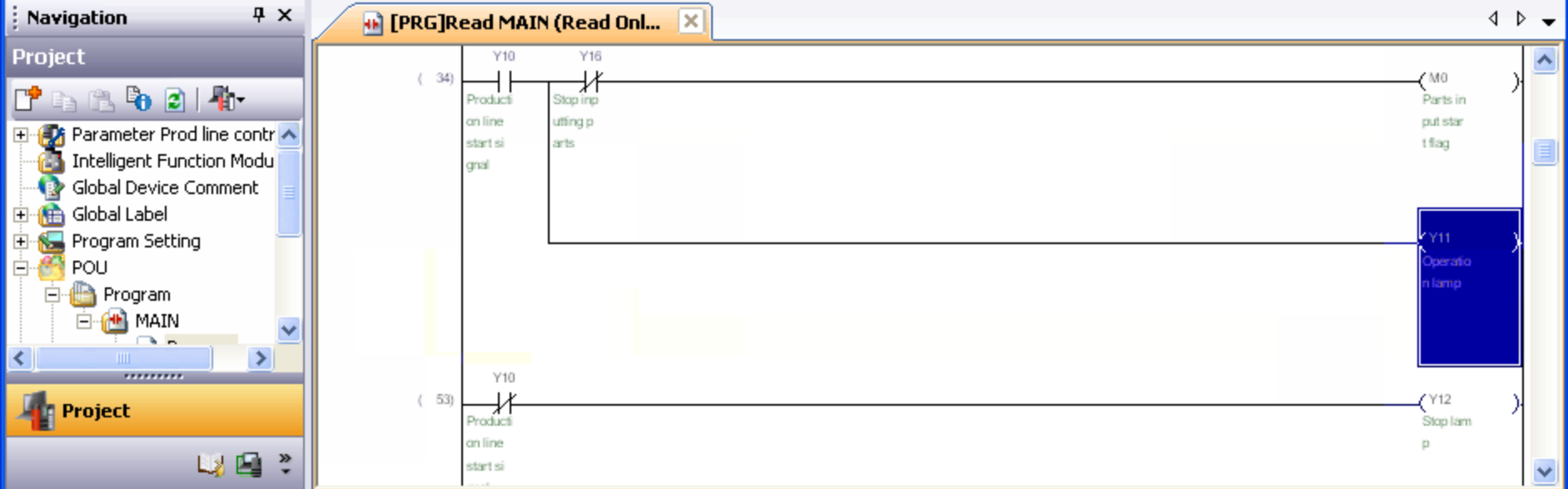
3.5.1

Menggunakan Fungsi Eksekusi Langkah

Kualitas

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

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



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

Debugging selesai. Selesaikan fungsi eksekusi langkah.
Klik untuk melanjutkan.

Skip Range Break Point Break Device

English Simple Q03UDE Host Station (52 N...



3.6 Mensimulasikan Operasi Perangkat Eksternal

Kualitas

Debugging bersamaan dengan pengoperasian perangkat eksternal tidak dimungkinkan dalam lingkungan pengembangan yang melarang penggunaan perangkat eksternal, misalnya fungsi simulasi.

Untuk mengatasi masalah ini, program debugging yang mensimulasikan pengoperasian perangkat eksternal ditambahkan secara konvensional.

Namun, selain diperlukan waktu dan upaya yang besar untuk membuat program simulasi, diperlukan juga modifikasi program ketika mengubah pengoperasian.

Gunakan "**Pengaturan Sistem I/O**" untuk menyelesaikan masalah ini.

Fungsi ini dapat mensimulasikan operasi perangkat eksternal tanpa menggunakan program debugging.

Operasi perangkat eksternal dapat diatur atau diubah dengan mudah di jendela pengaturan. Karena itu, pembuatan/modifikasi program secara konvensional tidak diperlukan.

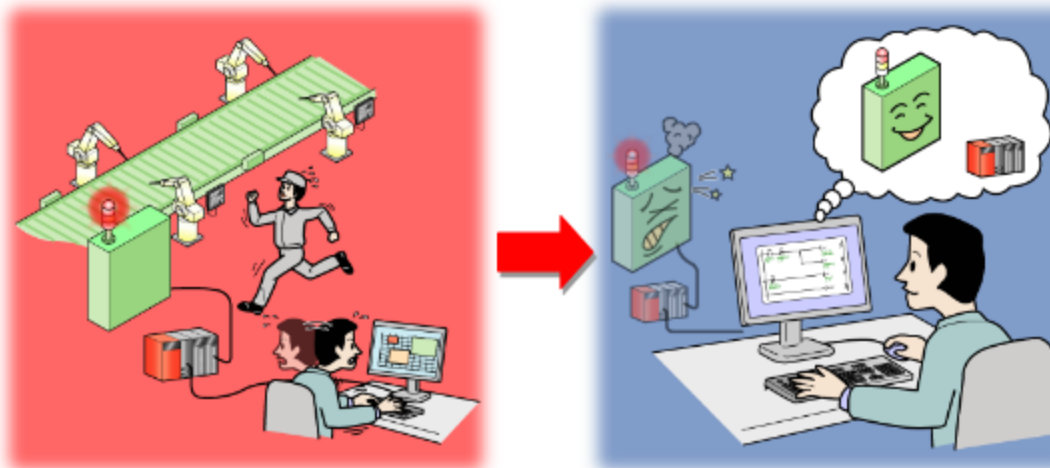
Operasi perangkat eksternal dapat diatur dalam dua cara sebagai berikut:

<Mengatur nilai perangkat>

Nilai perangkat yang ditentukan dapat diubah pada waktu yang ditetapkan oleh pengatur waktu setelah kondisi yang ditentukan terpenuhi.

<Mengatur dalam format diagram waktu>

Perubahan perangkat yang telah diatur dalam format diagram waktu yang ditentukan dapat dipicu ketika kondisi yang ditentukan terpenuhi.



3.6.1 Memasukkan Nilai Perangkat untuk Mengatur Sistem I/O



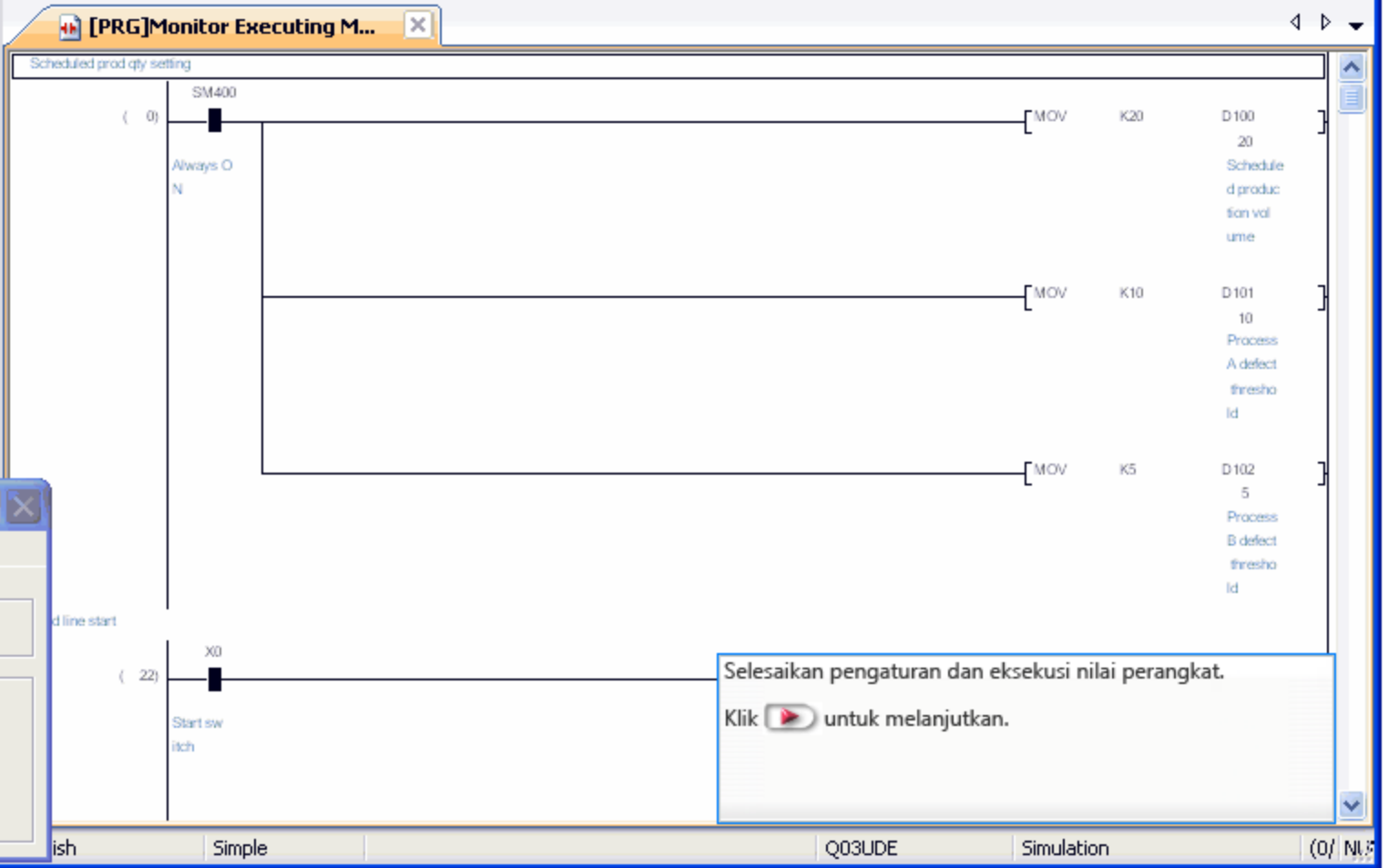
MELSOFT Series GX Works2 C:\Sequential Programs\le-learning\Robot control - [[PRG]Monitor Executing MAIN (Read Only) 194 Step

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

Toolbar: File operations, navigation, and execution controls.

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



GX Simulator 2

Tool Options

Switch

RESET STOP RUN

LED

MODE	■
RUN	■
ERR.	□
USER	□

Selesaikan pengaturan dan eksekusi nilai perangkat.
 Klik untuk melanjutkan.

3.6.2 Menggunakan Format Diagram Waktu untuk Mengatur Sistem I/O



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

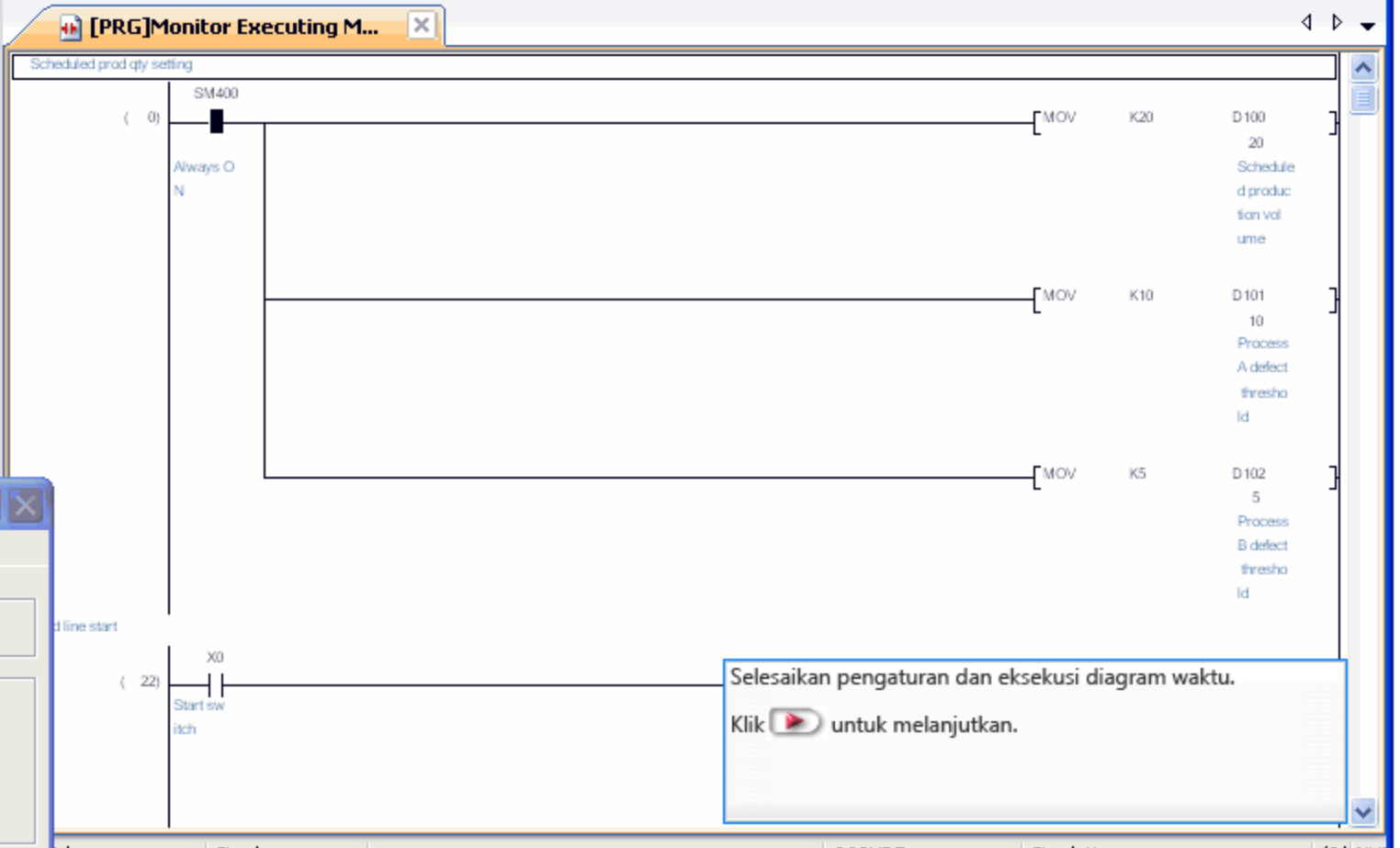
Project Edit Find/Replace Compile View Online Debug Diagnostics Tool Window Help

File Edit View Tools Simulation

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 RUN

LED

MODE	■
RUN	■
ERR.	■
USER	■

Selesaikan pengaturan dan eksekusi diagram waktu.
Klik untuk melanjutkan.

Simulation controls: Play, Stop, Home, etc.

Bab 4 Manajemen Proyek dan Langkah Pengamanan

Langkah-langkah pembelajaran pada Bab 4

Pada Bab 4, Anda akan mempelajari fungsi-fungsi yang digunakan untuk manajemen proyek dan langkah pengamanan.

- 4.1 Mencegah Kebocoran Pengetahuan Teknis dan Modifikasi Program Tanpa Wewenang
 - 4.1.1 Membatasi Data yang Dapat Diakses oleh Setiap Pengguna
- 4.2 Pencadangan Proyek dan Manajemen Versi
- 4.3 Membandingkan Program yang Disimpan ke PLC dan PC

4.1 Mencegah Kebocoran Pengetahuan Teknis dan Modifikasi Program Tanpa Wewenang



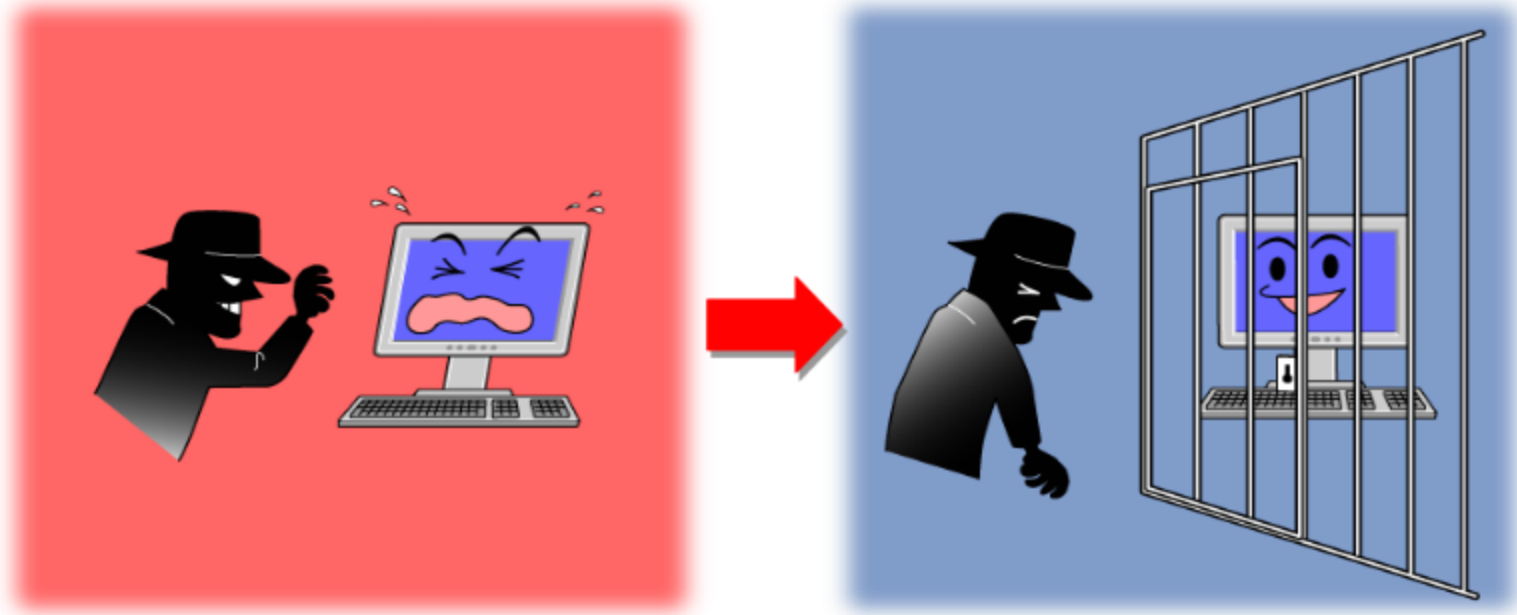
Program sekuens mengandung pengetahuan teknis dan data yang penting secara strategis. Kebocoran pengetahuan teknis dan data dari program ke luar dapat berdampak merugikan pada bisnis. Modifikasi program tanpa wewenang dapat mengakibatkan masalah produksi seperti berhentinya sistem.

Gunakan "**Keamanan**" untuk menyelesaikan masalah ini.

Fungsi ini membatasi pengguna yang dapat mengakses setiap proyek melalui perlindungan dengan kata sandi.

Fungsi ini juga dapat membatasi rentang data atau fungsi yang dapat diakses atau dioperasikan oleh setiap pengguna.

Jadi, fungsi ini mencegah pengguna yang tidak berwenang menjelajahi atau mengedit program.



4.1.1

Membatasi Data yang Dapat Diakses oleh Setiap Pengguna



Program sekuens skala besar sering dikembangkan oleh dua programmer atau lebih yang berbagi pekerjaan. Dalam kasus pengembangan oleh tim, rentang data yang dapat diakses dan fungsi yang tersedia harus dikelola dengan tepat sesuai rentang kerja dan tingkat keahlian setiap programmer serta kerahasiaan data yang ditangani oleh setiap programmer. Manajemen akses ini dapat diimplementasikan dengan mengatur **tingkat akses** keamanan.

<Tingkat akses>

Hak pengoperasian untuk data yang dicakup dalam proyek dapat diatur untuk setiap pengguna. Lima tingkat akses berikut dapat diatur.

Tingkat akses		Otoritas pengoperasian	
Tinggi Rendah	Administrator	Tingkat administrator	Berwenang menggunakan semua fungsi.
	Pengembang (Tingkat 3)	Tingkat pengembang	Pengaturan keamanan, akses data, dan beberapa operasi dibatasi.
	Pengembang (Tingkat 2)		
	Pengembang (Tingkat 1)		
	Pengguna	Tingkat operator	Hanya penjelajahan data proyek tersedia. Tidak dapat membaca dari CPU PLC.

4.1.1 Membatasi Data yang Dapat Diakses oleh Setiap Pengguna

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

Undo Ctrl+Z
 Redo Ctrl+Y
 Cut Ctrl+X
 Copy Ctrl+C
 Paste Ctrl+V
 Delete Del
 Restore After Ladder Conversion
 Insert Row Shift+Ins
 Delete Row Shift+Del
 Insert Column Ctrl+Ins
 Delete Column Ctrl+Del
 NOP Batch Insert...
 NOP Batch Delete
 Edit Line F10
 Delete Line Alt+F9
 Change TC Setting...
 Ladder Edit Mode(Z)
 Ladder Symbol
 Inline Structured Text
 Edit FB Instance...
 Documentation
 Easy Edit

English Simple User Q03UDE Host Station (15 N...

4.2

Pencadangan Proyek dan Manajemen Versi



Program penting dapat hilang akibat kegagalan CPU PLC.

Jika program cadangan tidak tersedia, harapannya kecil untuk melakukan pemulihan cepat.

Bahkan sekalipun program cadangan tersedia, akan sulit menentukan apakah versinya sama dengan program yang hilang kecuali jika ada manajemen versi.

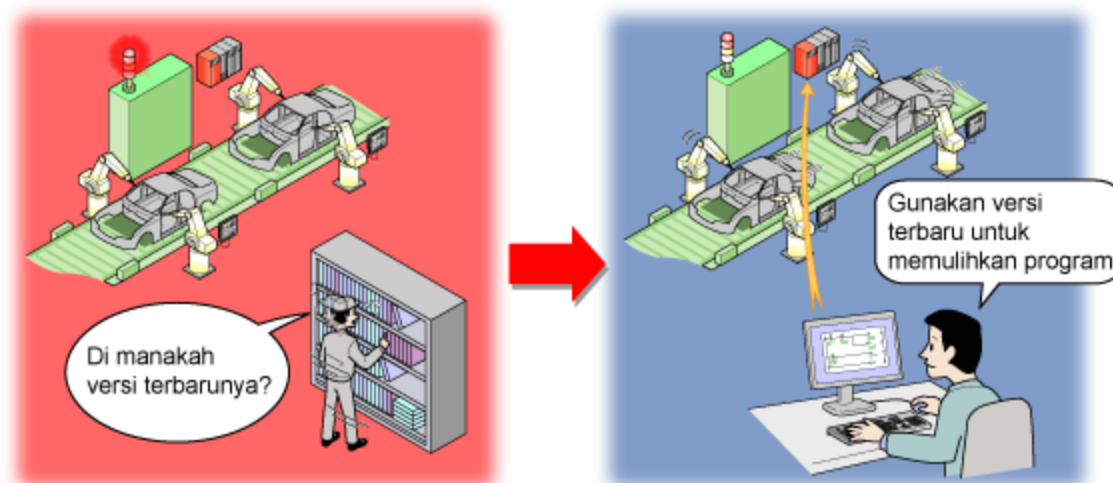
Untuk menghadapi situasi yang tidak diharapkan, penting untuk menjalankan pencadangan berkala dan memastikan manajemen versi.

Gunakan **"Riwayat perubahan"** untuk menyelesaikan masalah ini.

Fungsi ini dapat merekam hingga 100 set riwayat perubahan (nomor riwayat, tanggal/waktu, pengguna, judul, komentar) proyek.

Data proyek pada waktu perekaman juga dicadangkan di saat bersamaan.

Manajemen versi yang dipastikan dengan fungsi riwayat perubahan memungkinkan Anda memulihkan program yang hilang, memverifikasi versi program, dan menerapkan pemulihan cepat jika terjadi kondisi yang tidak diharapkan.



4.2 Pencadangan Proyek dan Manajemen Versi



MELSOFT Series GX Works2 C:\Sequential Programs\le-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

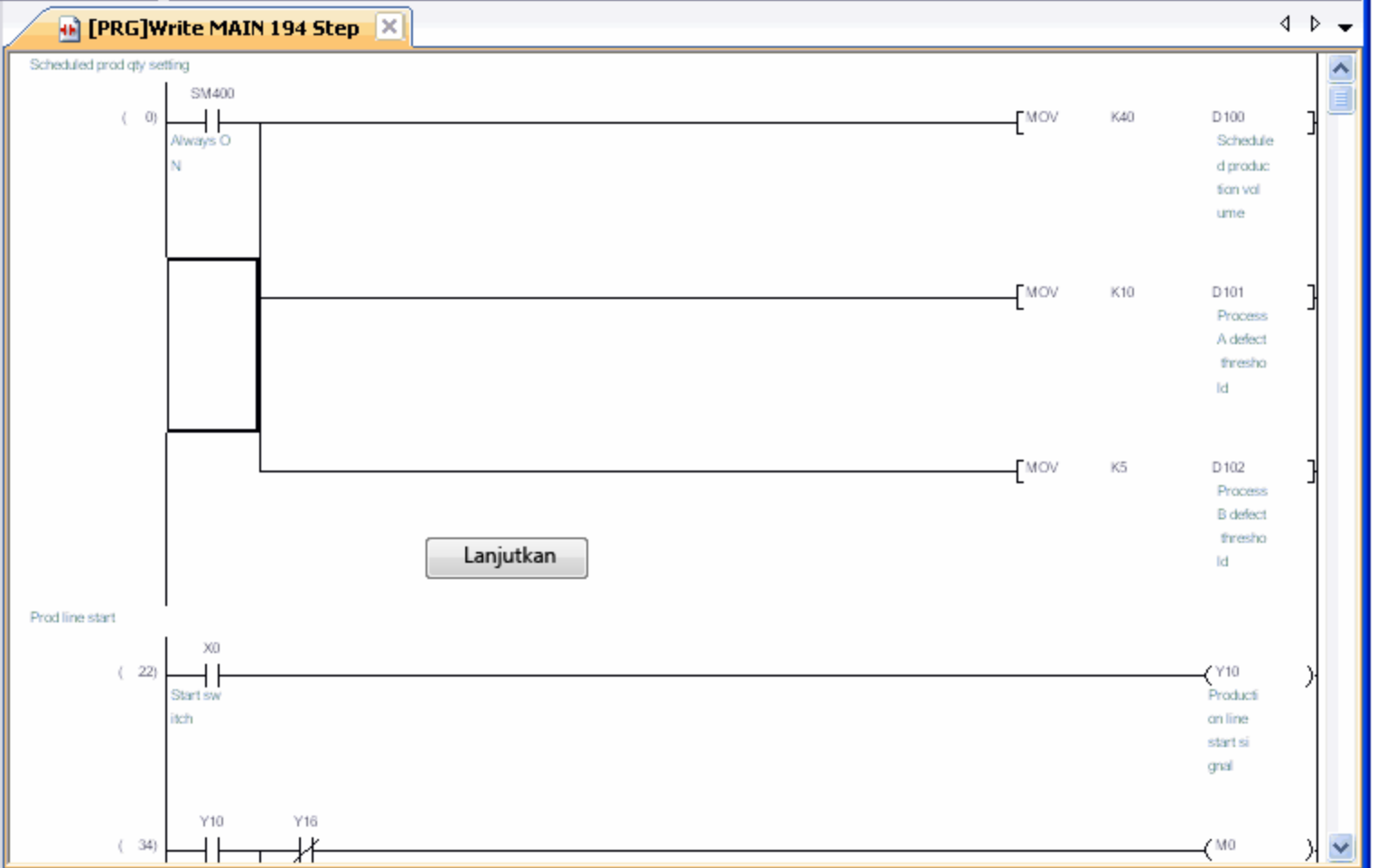
Project

- 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



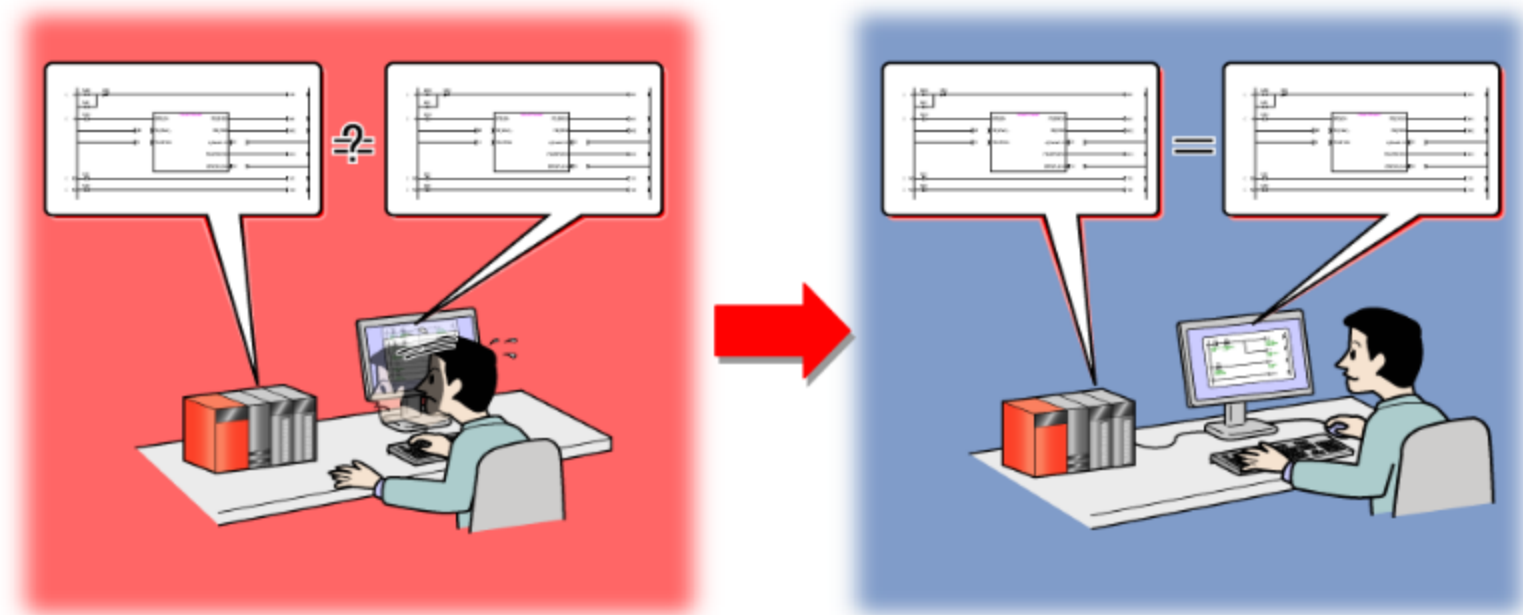
4.3

Membandingkan Program yang Disimpan ke PLC dan PC



Dalam kondisi normal, program disimpan ke PC di lingkungan pengembangan dan juga ditulis ke PLC. Kedua program ini tidak selalu sama. Melakukan pemeriksaan visual saja untuk melihat apakah keduanya sama dapat berujung pada kesalahan.

Gunakan **"Verifikasi dengan PLC"** untuk menyelesaikan masalah ini. Fungsi ini dapat memverifikasi bahwa program yang dibuka oleh GX Works2 cocok dengan program yang ditulis ke PLC.



4.3 Membandingkan Program yang Disimpan ke PLC dan PC

MELSOFT Series GX Works2 C:\Sequential Programs\le-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

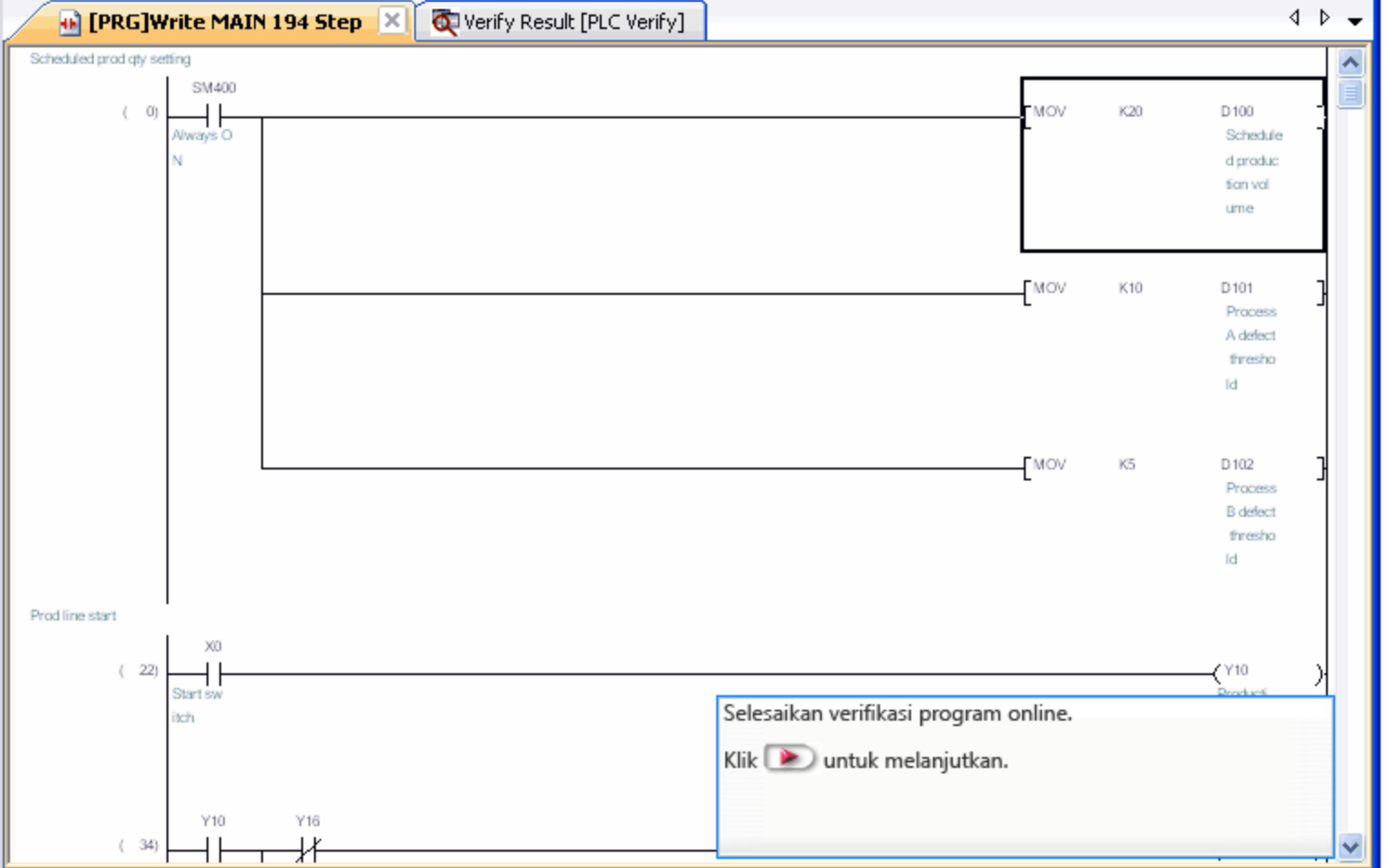
F5 SF5 F6 SF6 F7 F8 F9 SF9 CF9 CF10 SF7 SF8 AF7 AF8 SAF5 SAF6 SAF7 SAF8 AF5 CAF5 CA10 F10 AF9

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



Sesuaikan verifikasi program online.
 Klik untuk melanjutkan.

Tes Tes Akhir

Setelah menyelesaikan semua pelajaran dari **PLC GX Works2 Lanjutan**, kini Anda siap mengikuti tes akhir. Jika Anda masih kurang memahami salah satu topik yang dibahas, gunakan kesempatan ini untuk mengulas topik tersebut.

Total terdapat **8 pertanyaan (8 pilihan)** dalam Tes Akhir ini.

Anda dapat mengikuti tes akhir sesering mungkin.

Cara menilai tes

Setelah memilih jawaban, pastikan untuk mengeklik tombol **Jawab**. Jawaban akan hilang jika Anda melanjutkan tanpa mengeklik tombol Jawab. (Dianggap sebagai pertanyaan belum dijawab.)

Hasil penilaian

Jumlah jawaban yang benar, jumlah pertanyaan, persentase jawaban yang benar, dan hasil lulus/gagal akan ditampilkan pada halaman nilai.

Jawaban yang benar : 2
Jumlah total pertanyaan : 9
Persentase : 22%

Agar lulus tes, Anda harus menjawab **60%** pertanyaan dengan benar.

Lanjutkan

Tinjau

Coba lagi

- Klik tombol **Lanjutkan** untuk keluar dari tes.
- Klik tombol **Tinjau** untuk meninjau tes. (Jawaban yang benar dicentang)
- Klik tombol **Coba** lagi lagi untuk mengulang tes.

Tes**Tes Akhir 1**

Manakah dari fungsi berikut yang mengaktifkan pemrograman yang efisien dengan menggunakan blok ladder yang digunakan berulang sebagai komponen yang dapat dipakai bersama? (Pilih satu.)

- Inline structured text (Teks berstruktur sebaris)
- Label (label)
- Blok fungsi (Function block)

Jawab

Kembali

Tes

Tes Akhir 2

Manakah dari fungsi berikut yang dapat membuat program yang mudah dibaca dengan mengubah nama perangkat menjadi nama yang dikaitkan dengan aplikasinya? (Pilih satu.)

- Device comment (Komentar perangkat)
- Label (label)
- Note (Catatan)

Jawab

Kembali

Tes**Tes Akhir 3**

Manakah dari fungsi berikut yang dapat membuat program yang mudah dibaca dengan menyediakan informasi tentang pemrosesan untuk setiap blok ladder? (Pilih satu.)

- Device comment (Komentar perangkat)
- Line statement (Pernyataan baris)
- Note (Catatan)

Jawab

Kembali

Tes

Tes Akhir 4

Manakah dari berikut ini yang merupakan penjelasan yang benar untuk fungsi "Verify with PLC" (Verifikasi dengan PLC)? (Pilih satu.)

- Membandingkan program yang sedang diedit dengan program yang direkam pada riwayat perubahan.
- Membandingkan program yang sedang diedit dengan program yang dipilih yang disimpan ke PC.
- Membandingkan program yang sedang diedit dengan program yang ditulis ke CPU PLC.

Jawab

Kembali

Tes Tes Akhir 5

Manakah dari berikut ini yang merupakan penjelasan yang benar untuk fungsi "I/O System Setting" (Pengaturan Sistem I/O)? (Pilih satu.)

- Mensimulasikan pengoperasian peralatan I/O eksternal pada PC selama debugging.
- Mengontrol dari jauh pengoperasian peralatan I/O eksternal dari PC selama debugging.
- Mensimulasikan pengoperasian CPU PLC pada PC selama debugging.

Jawab

Kembali

Tes**Tes Akhir 6**

Manakah dari berikut ini yang merupakan penjelasan yang benar untuk fungsi "Change history" (Riwayat perubahan)? (Pilih satu.)

- Merekam operasi GX Works2 langkah demi langkah sehingga dapat dipulihkan dengan bebas nanti.
- Merekam informasi riwayat dan pencadangan proyek agar dapat diverifikasi dan dipulihkan nanti.

Jawab

Kembali

Tes

Tes Akhir 7

Manakah dari fungsi berikut yang dapat digunakan selama debugging untuk mengubah nilai perangkat saja tanpa memodifikasi program? (Pilih satu.)

- Break execution (Eksekusi jeda)
- I/O system setting (Pengaturan sistem I/O)
- Executional conditioned device test (Uji perangkat terkondisi eksekusi)

[Jawab](#)[Kembali](#)

Tes**Tes Akhir 8**

Misalkan proyek mencakup dua program, A dan B, dan Anda menggunakan fungsi "label". Manakah tipe label berikut yang dapat diakses oleh program B? (Pilih dua.)

- Label global
- Label lokal program A
- Label lokal program B

Jawab

Kembali

Tes**Skor Tes**

Anda telah menyelesaikan Tes Akhir. Hasil Anda adalah sebagai berikut.
Untuk mengakhiri Tes Akhir, lanjutkan ke halaman berikutnya.

Jawaban yang benar: **0**

Jumlah total pertanyaan: **8**

Persentase: **0%**

[Lanjutkan](#)[Tinjau](#)[Coba lagi](#)

Tes Anda gagal.

Anda telah menyelesaikan Kursus **PLC GX Works2 Lanjutan**.

Terima kasih telah mengikuti kursus ini.

Kami harap Anda menikmati pelajaran, dan kami harap informasi yang diperoleh dalam kursus ini dapat bermanfaat di masa mendatang.

Anda dapat mengulas kursus ini sesering yang Anda inginkan.

Tinjau

Tutup