SAMSUNG | Rockwell Automation

NX700 Series Controller
NX-CPU700p
User Manual

Important User Information

Solid state equipment has operational characteristics differing from those of electromechanical equipment. Because of these differences, and also because of the wide variety of uses for solid state equipment, all persons responsible for applying this equipment must satisfy themselves that each intended application of this equipment is acceptable.

In no event will Rockwell Samsung Automation be responsible or liable for indirect or consequential damages resulting from the use or application of this equipment.

The examples and diagrams in this manual are included solely for illustrative purposes. Because of the many variables and requirements associated with any particular installation, Rockwell Samsung Automation cannot assume responsibility or liability for actual use based on the examples and diagrams.

No patent liability is assumed by Rockwell Samsung Automation. with respect to use of information, circuits, equipment, or software described in this manual.

Reproduction of the contents of this manual, in whole or in part, without written permission of Rockwell Samsung Automation. is prohibited.

Throughout this manual we use notes to make you aware of safety considerations.

WARNING 	Identifies information about practices or circumstances which may lead to serious personal injury or death, property damage, or economic loss.
IMPORTANT	Identifies information that is critical for successful application and understanding of the product.
ATTENTION 	Identifies information about practices or circumstances that can lead to minor personal injury, property damage, economic loss, or product malfunction. However, depending on circumstances, failure to follow the directions accompanying this symbol may also lead to serious consequences.

Contents

Safety Instructions

Please read this manual and the related documentation thoroughly and familiarize yourself with product information, safety instructions and other directions before installing, operating, performing inspection and preventive maintenance. Make sure to follow the directions correctly to ensure normal operation of the product and your safety.

WARNING

- If this product is used in a situation that may cause personal injury and/or significant product damage, implement safe measures such as use of fault-safe equipment.
- Do not use this product under the conditions exposed to explosive gases. It may cause an explosion.

ATTENTION

- Make sure to use an external device when configuring the protective circuit breakers for emergencies or interlock circuits.
- Fasten the terminal screws tightly to ensure that the cable connection is secure. Incorrect cable connection may cause overheating and product malfunction.
- Operate and keep the product under the allowed conditions directed in product specifications. Otherwise it may cause overheating and product malfunction.
- Do not disassemble or remodel the product. Otherwise it may cause an electric shock or malfunction.
- Do not touch the terminals when the power is on. Otherwise it may cause an electric shock.

Installation Environment

ATTENTION

Do not install your PLC system if any of the following conditions are present:

- Ambient temperature outside the range of 0 to 55 °C (32 to 131 °F)
- Direct sunlight
- Humidity outside the range of 30% to 85% (non-condensing)
- Chemicals that may affect electronic parts
- Excessive or conductive dust, or salinity
- High voltage, strong magnetic fields, or strong electromagnetic influences
- Direct impact and excessive vibration

ATTENTION

Electrostatic Discharges

Under dry condition, excessive electrostatic discharges may occur. Make sure to remove electrostatic discharges by touching a grounded metal piece before touching your controller system modules.

ATTENTION

Cleaning

Never use chemicals such as thinner because they melt, deform or discolor PCB boards.

ATTENTION

Precautions for use of power

- Run your PLC system only after the I/O devices and motor devices have started. (For example, first power on in the PROG mode, then change the operation mode to RUN.)
- Make sure to power off I/O devices after ensuring PLC operation is stopped.
- If you power on/off I/O devices when the PLC system is in operation, the system may malfunction because input signal noises may be recognized as normal inputs.

ATTENTION

Before powering on

Make sure to follow these directions before powering on your PLC system.

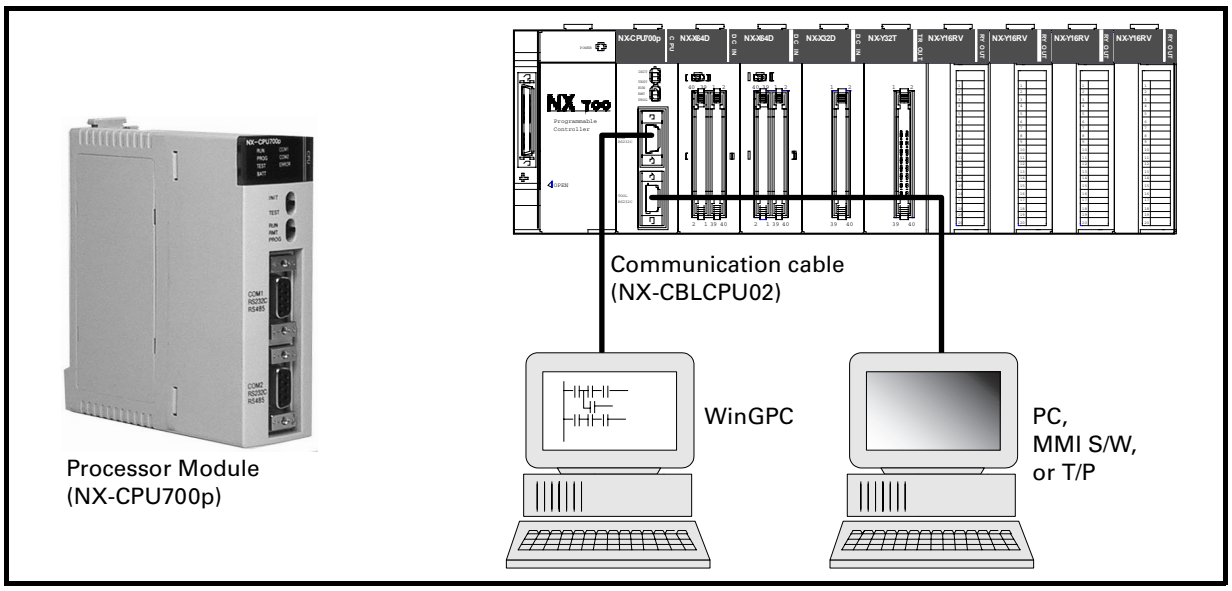
- When installing the system, ensure that there are no metal chips or conductive fragments that stick to wiring cables.
- Ensure that power supply and I/O wirings and power supply voltage are all correct.
- Securely fasten installation and terminal screws.
- Set the operation mode switch to PROG mode.

1

System Configuration

NX-CPU700p PLC Features and System Configuration

System configuration



Features

- **High-speed processing**
  With the high-speed ASIC, the NX-CPU700p processes basic instructions at a speed of 0.2 μs per step.

- **Runtime Editing**
  The NX-CPU700p processor module allows you to modify instructions while operating.

- **Built-In Real Time Clock (RTC)**
  Built-in real time clock supports programming by time and date.

- **High-capacity programming and memory backup**

  The NX-CPU700p allows you to program up to 20K words. Built-in flash EEPROM allows you to save programs separately.

- **Self-diagnostics**

  Self-diagnostics allows you to minimize system errors and maximize diagnostic efficiency.

- **Maximum 1600 I/O points**

  With a 12-slot base backplane and a 12-slot expansion backplane, you can use up to 1600 I/O points (when all I/Os are configured with 64-point digital I/O modules).

- **Supports various I/O and specialty modules**

  The NX-CPU700p processor module supports 24V dc input (16, 32, and 64 points), 110V ac input, 220V ac input, relay output, transistor output (16, 32, and 64 points), SSR output, A/D, D/A, RTD, TC, high-speed counter, CCU+, and SCU.

- **WinGPC as programming tool**

  WinGPC, which is the programming tool for NX-CPU700p processor module, lets you create, edit, download, and upload your control program, and allows you to peform processor status monitoring, force I/O, and I/O status montioring as well. It is a powerful, easy-to-use programming tool.

- **Various I/O backplane options (up to 12 slots)**

  The NX-CPU700p PLC allows you to choose one from 3-, 5-, 8-, 10-, or 12-slot backplanes for maximum system configuration flexibility.

- **Built-in RS232C/RS485, 2 ports**

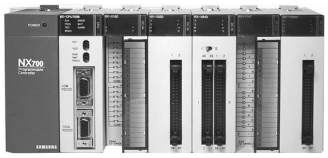
  With two built-in communication ports, the NX-CPU700p processor module allows you to connect directly to computers or touch panels and exchange a high volume of data at high speed. The COM2 port supports a simple user-defined communication, and allows you to connect to barcode readers, inverters, or servo motors. (Binary communication is available.)

I/O Backplanes and I/O Points

3-Slot Type

NX-BASE03
48 Points: 16-point I/O
192 Points: 32-point I/O

5-Slot Type



NX-BASE0)
80 Points: 16-point I/O
320 Points: 64-point I/O

8-Slot Type

NX-BASE08
128 Points: 16-point I/O
512 Points: 64-point I/O

10-Slot Type

NX-BASE10
160 Points: 16-point I/O
640 Points: 64-point I/O

12-Slot Type

NX-BASE12
192 Points: 16-point I/O
768 Points: 64-point I/O

- **Flexible system configuration: 5 types of backplane (3-, 5-, 8-, 10- and 12-slot)**

  The NX-CPU700p PLC has 5 types of backplane (3-, 5-, 8-, 10 and 12-slot type), providing you with very flexible I/O configuration.

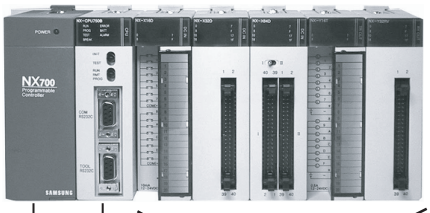
  All backplanes, I/O modules, power supply, and specialty modules are available to other NX700 series PLC systems, for example, NX-CPU700 or NX-CPU750A system.

- **Maximum 1600 I/O points**

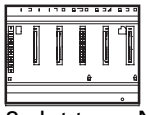
  With a 12-slot base backplane and a 12-slot expansion backplane, you can use up to 1600 I/O points by configuring all I/Os with 64-point digital I/O modules.

Module Types and Combinations

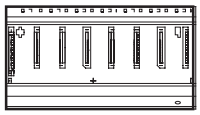
Combinations of backplanes and modules



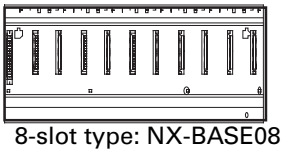
Backplanea



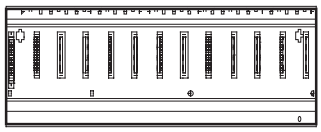
3-slot type: NX-BASE03
(common for base and
expansion backplanes)



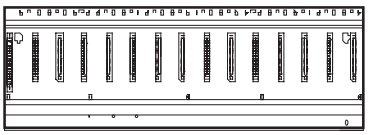
5-slot type: NX-BASE05
(common for base and
expansion backplanes)



8-slot type: NX-BASE08
(common for base and
expansion backplanes)

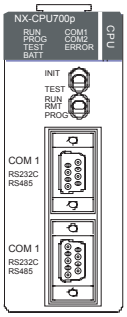


10-slot type: NX-BASE10
(common for base and
expansion backplanes)



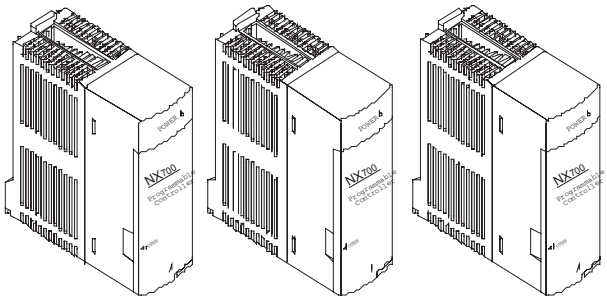
12-slot type: NX-BASE12
(common for base and
expansion backplanes)

Processor module



NX-CPU700p
20k step base memory
2 ports (RS232/RS485)

Power supply modules



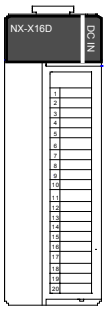
NX-POWER
110 to 220V ac input
(free voltage)
4A at 5V,
0.5A at 24V

NX-PWR220
220V ac input
6A at 5V

NX-PWRDC
24V dc input
5A at 5V

*: Available with NX-CPU700p v2.0 or higher

I/O modules



- **16-point input**

  24V dc In:
  NX-X16D

  110V ac In:
  NX-X16A110

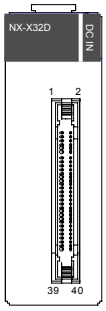
  220V ac In:
  NX-X16A220)

- **16-point output**

  Relay Out:
  NX-Y16R
  NX-Y16RV

  Transistor Out:
  NX-Y16T

  SSR Out:
  NX-Y16SSR



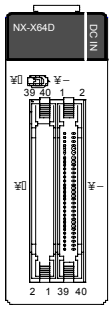
- **32-point input**

  24V dc In:
  NX-X32D

- **32-point output**

  Relay Out:
  NX-Y32RV

  Transistor Out:
  NX-Y32T



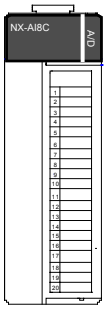
- **64-point input**

  24V dc In:
  NX-X64D

- **64-point output**

  Transistor Out:
  NX-Y64T

Specialty modules



- **Analog input**

  8-channel current input:
  NX-AI8C

  8-channel voltage input:
  NX-AI8V

- **Analog output**

  4-channel current output:
  NX-AO4C

  4-channel voltage output:
  NX70-AO4V

- **RTD input**

  8-channel:
  NX-RTD8

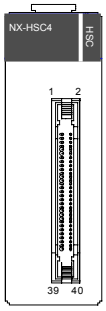
  4-channel:
  NX-RTD4

- **TC input**

  8-channel:
  NX-TC8

  4-channel:
  NX-TC4

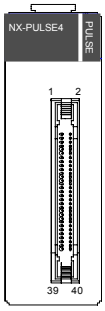
*



- **High-performance high-speed counter**

  4-channel:
  NX-HSC4

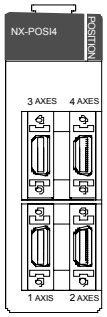
*



- **Pulse output module**

  4-channel:
  NX-PULSE4

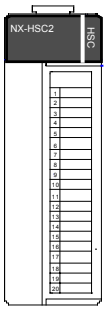
*



- **Positioning module**

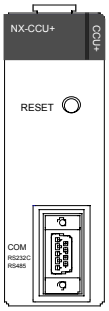
  (supporting open collector or line drive output)

  1-axis: NX-POSI1
  2-axes: NX-POSI2
  3-axes: NX-POSI3
  4-axes: NX-POSI4



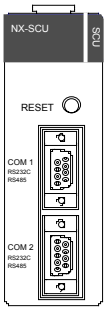
- **High-speed counter**

  2-channel:
  NX-HSC2



- **CCU+**

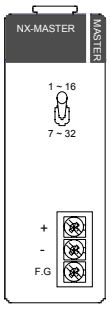
  NX-CCU+



- **SCU**

  NX-SCU

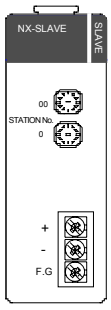
*



- **Remote I/O Master**

  NX-MASTER

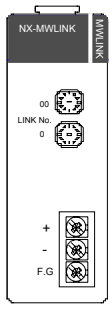
*



- **Remote I/O Slave**

  NX-SLAVE

*



- **MW-LINK Module**

  NX-MWLINK (Multi W-Link and W-Link functions)

Processor Module

NX-CPU700p

20k step, 0.2 μs
2 comm. ports
WinGPC S/W

Software & cables

Programming tool:
WinGPC S/W

Comm. cables to PC:
NX-CPUCBL02,
NX-CPUCBL05

Power Supply Module

AC Type

1) NX-POWER
110/220V ac, free voltage
4A at 5V, 0.5A at 24V

2) NX-PWR220
110/220V ac, free voltage
6A at 5V

DC Type

NX-PWRDC
24V dc input
5A at 5V

Specialty Module (1)

A/D, D/A, RTD, TC

CCU, SCU

High-speed counter
(1, 2CH)

DevicedNet System

Specialty Module (2)

High-performance,
high-speed counter (4CH)

Positioning module
(1, 2, 3, 4 axes)

Pulse output module
(4CH)

Remote I/O system

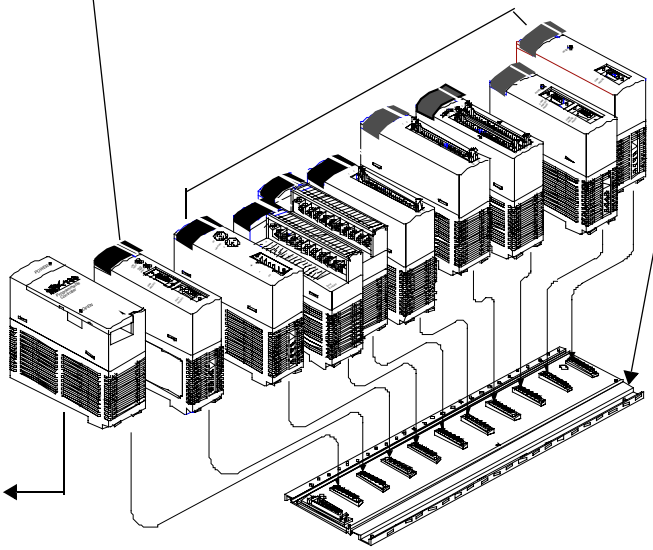
MW-LINK system

I/O Module

16-point type

32-point type

64-point type



Backplane

3-slot type

5-slot type

8-slot type

10-slot type

12-slot type

Available Combinations and Restrictions

Restrictions by Module

○ : Available, △ : Version 2.0 or higher required, ③: Not available

Configuration \ Module		Basic module					Specialty module								Network module					
		Processor module	Backplane (3, 5, 8, 10, and 12-slot)	Power supply module	Input module	Output module	A/D, D/A, RTD, TC modules	Positioning module (1,2,3 and 4 axes)	High-speed counter (1, 2CH)	High-performance high-speed counter (4CH)	Pulse I/O module (4CH)	SCU (2CH)	CCU+ 1CH) (1)	Remote I/O master module (1)	Remote I/O slave module (1)	MW-Link module (1)	EtherNet module (1)	DeviceNet master module	Profi-DP master module
Base backplane	Mount the modules in the following order, starting from the leftmost slot of the backplane. • Power supply module • Processor module • I/O module or specialty module 	○	○	○	○	○	○	△	○	△	△	○	○	△	△	△	③	△	△
Expansion backplane	Mount the modules in the following order, starting from the leftmost slot of the backplane. • Power supply module • I/O module or specialty module 	③	○	○	○	○	○	③	○	③	③	○	③	③	③	③	③	③	③

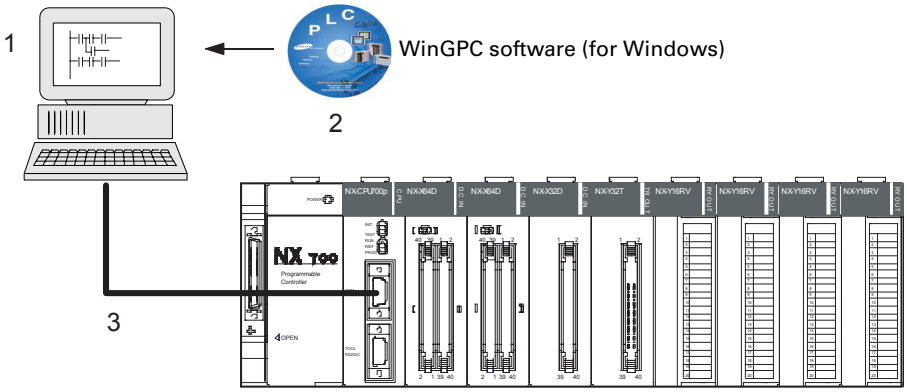
(1) Restrictions on available link modules

Module		Availability with NX-CPU700p
CCU+		Only one unit available
MW-Link module	W-Link mode	Available with NX-CPU700p version 2.0 or higher
	W-Link 2 mode	Unavailable
EtherNet module		Unavailable
Remote I/O Master and Slave modules		Available with NX-CPU700p version 2.0 or higher
Positioning, High-performance high-speed counter (4CH), and Pulse I/O modules		Available with NX-CPU700p version 2.0 or higher

Programming Tools

Tools required for programming

The following tools are required when programming with WinGPC.

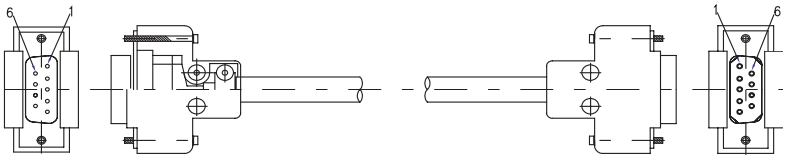


1. **Personal computer**
2. **WnGPC (for Windows)**

   WinGPC is the programming tool for editing and debugging PLC programs for NX series controllers. WinGPC supports Microsoft Windows operating system.

3. **Cable for programming**

   For connection to the communication port of the NX70 processor module, you can either purchase and use NX-CBLCPU02 or NX-CBLCPU05 cable or construct the needed cables as follows:



Processor module 9-pin connector

Signal	Pin No.
TXD	2
RXD	3
-	4
S.G	5
485(-)	6
485(+)	7
-	8
VCC	9

Computer 9-pin connector

Pin No.	Signal
2	RXD
3	TXD
4	DTR
5	S.G
6	DSR
7	RTS
8	CTS
9	-

2

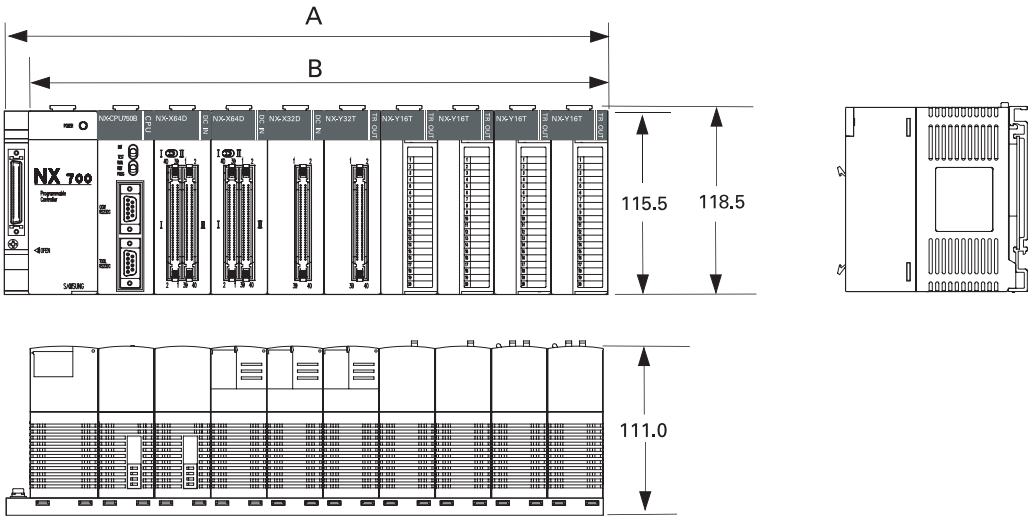
Hardware Features and Specifications

Overall Specifications

General specifications

Item		Specifications
Temperature	Operating	0 °C to +55 °C (32 °F to 131 °F)
	Storage	-25 °C to +70 °C (-13 °F to 158 °F)
Humidity	Operating	30 to 85% RH (non-condensing)
	Storage	30 to 85% RH (non-condensing)
Withstand voltage		1500V ac for 1 minute between external terminal (ac) and frame ground (FG)
		500V ac for 1 minute between external terminal (dc) and frame ground (FG)
Insulation resistance		100 MΩ or more at 500 mega V dc between external terminal and frame ground (FG)
Vibration immunity		10 to 55 Hz 1 sweep per minute, 0.75 mm peak to peak, 10 minutes per axis (X, Y, Z)
Shock immunity		10G 4 times for each X, Y, Z direction
Noise immunity		1500 Vp-p with 50 ns to 1 μs pulse width (generated by noise simulator)
Environment		IP 20

Dimensions (unit: mm)



unit (mm)

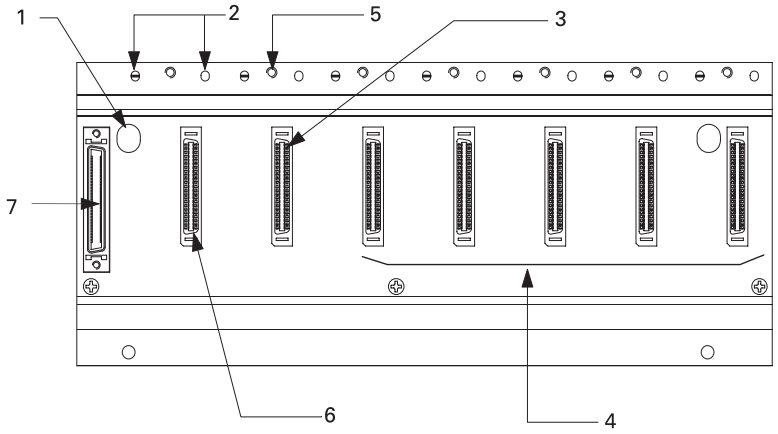
	3-slot type	5-slot type	8-slot type	10-slot type	12-slot type
A (mm)	205.0	276.0	381.0	452.0	522.0
B (mm)	183.8	254.2	359.8	430.2	500.6

Performance specifications

Processor		NX-CPU700p
Control method		Stored program, cyclic operation
External I/O		1,600 points
Instructions	Basic	28 types
	Advanced	150 types
Processing speed	Basic instructions	0.2 to 0.4 μs per step
	Advanced instructions	0.4 to several tens of μs per step
Max. program size		20K words
Memory map	I/O contact (R)	R0.0 to R127.15 (2,048 points, 128 words)
	Link contact (L)	L0.0 to L63.15 (1,024 points, 64 words)
	Internal contact (M)	M0.0 to M127.15 (2,048 points, 128 words) (Where, M0.0 to M63.15 can be used as link contact.)
	Keep contact (K)	K0.0 to M127.15 (2,048 points, 128 words)
	Special contact (F)	F0.0 to F15.15 (256 points)
	Timer/Counter contacts (TC or TIM)	256 channels (Timer + Counter), Set value range: 0 to 65535 Timer: 0.01 second: CH000 to CH063 (64 channels), 0.1 second: CH064 to CH255 (192 channels) Counter: CH000 to CH255 (256 channels)
	Data register (W)	W0000 to W2047, W3072 to W5119 (4,096 words)
	Special register (W, SR)	W256 (=SR000) to W3071 (=SR511) (512 words)
Real time clock (RTC)		Year, Month, Day, Hour, Minute, Second, Day of the week
Communication	Port 1	Supports both RS232 and RS485, 9600/19200/38400 bps
	Port 2	Supports both RS232 and RS485, 4800/9600/19200/38400 bps Supports a user-defined protocol (which allows connecting a barcode reader, an inverter, or a servo driver) Supports binary communication
Backup using flash ROM		Built-in user program backup capability

Backplane and Expansion Cable

Backplane (NX-BASE03, NX-BASE05, NX-BASE08, NX-BASE10, NX-BASE12)



Description

1. **Backplane mounting hole**
   This hole is for mounting a backplane on a control panel (control box). Use M5 screw for mounting.
2. **Module mounting guide holes**
   When mounting a module on the backplane, align the mounting clamp of the module with this hole.

   When using as a base backplane (processor side backplane), mount modules, starting from the leftmost slot, in the order of power supply module, processor module, and I/O (specialty) modules.

   When using as an expansion backplane, mount modules, starting from the leftmost slot, in the order of power supply module and I/O (specialty) modules.
3. **Connector for processor module**
   Mount the processor module onto this slot. Make sure to mount the processor module next to the power supply module. When using as an expansion backplane, I/O (specialty) modules can also be mounted onto this slot.
4. **Connectors for I/O modules (specialty modules)**
   Mount I/O (specialty) modules onto these slots.
5. **Module fixing hole**
   These holes are for holding a mounted module in place on a backplane. Fasten the screw on the module through this hole.
6. **Connector for power supply module**
7. **Connector for expansion cable**
   When using as an exansion backplane, use this connector to connect an expansion cable.

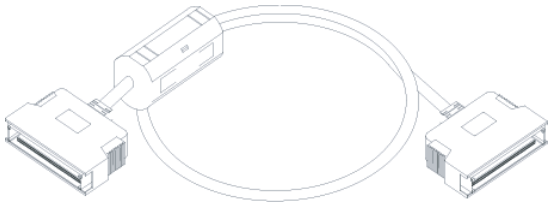
Backplane types

Type	Number of slots	Catalog number	Weight (g)	Remarks
Common for base and expansion backplanes	3	NX-BASE03	250g	
	5	NX-BASE05	330g	
	8	NX-BASE08	460g	
	10	NX-BASE10	570g	
	12	NX-BASE12	660g	

Expansion cable (NX-EXPCBL)

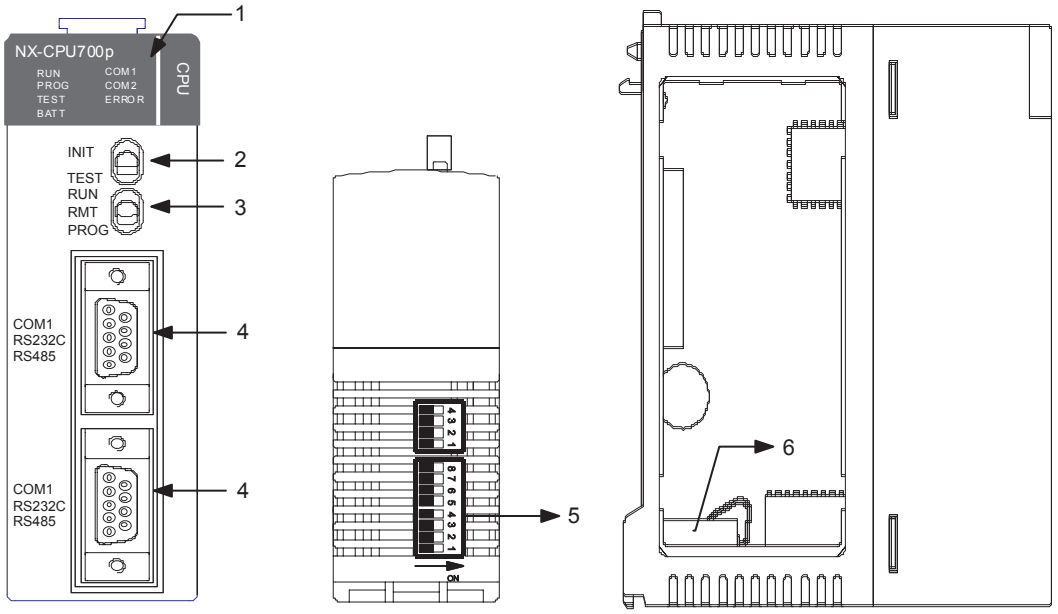
Expansion cable

Cable length	Catalog number	Remarks
1.5m	NX-EXPCBL15	Common for NX700 PLCs
0.8m	NX-EXPCBL08	



Processor Module

NX-CPU700p processor module



(Front) **(Bottom)** **(Side)**

- **Hardware features**

1. **Status LED**
   Indicates the operation status of PLC, such as run, stop, error, and alarm.
2. **Initialize/Test switch**
   Test switch is unused at present.
3. **Mode switch**
   Used to change the operation mode of the controller.
4. **RS232/RS485 communication ports (9-pin COM1 and COM2 ports)**
   Used to connect to the programming tool (WinGPC), touch panel, or MMI. Allows user-defined communication (COM2 port).
5. **TOOL connector for programming (TOOL, RS232C), 9 pins**
   Connector for connection with programming tool (WinFPST S/W).
6. **Operation condition setting switches**
   DIP switch 1 (4 poles, termination resistance setting)

   DIP switch 2 (8 poles, communication and program booting method setting)
7. **Memory backup battery**
   Backup battery for internal memory (RAM).
   Shipped with the battery connector disconnected.

- **Status LEDs**

LED	Color	Description
RUN	Green	On when the processor is running.
PROG	Green	On when the program can be edited.
BATT	Red	On when the battery is not mounted or is low.
TEST	Green	Not used
COM1	Green	Flashing when the processor is communicating via COM1.
COM2	Green	Flashing when the processor is communicating via COM2.
ERROR	Red	On when a processor error occurs.

- **Mode switch**

Mode	Description
RUN	Sets the processor operation mode to RUN mode.
RMT	Sets the processor operation mode to RUN or PROG mode.
PROG	Stops the processor, i.e, sets the processor operation mode to PROG mode.

- **Initialize switch**
  When set to the INIT position, all CPU errors are cleared. This operation is valid only when the Mode switch is set to PROG mode.

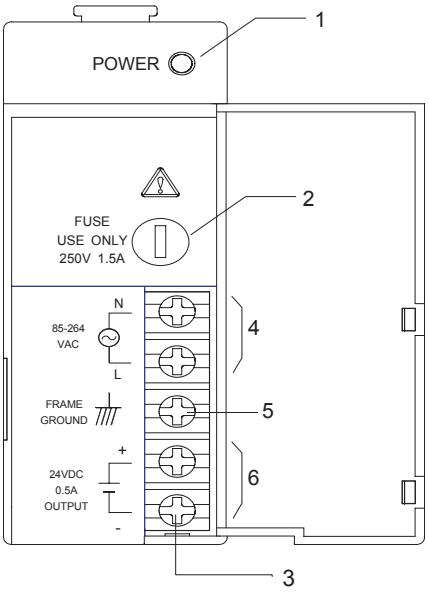
- **Operating condition setting switches**
- Switch for termination resistance setting (DIP switch 1)

Pin No.		Pin setting		Description	DIP Switch 1
4	3	ON	ON	For RS-485 communication, set both pins 3 and 4 to On if the system is a termination station. (Enables termination for COM1 terminal)	
		OFF	OFF	For RS-485 communication, set both pins 3 and 4 to Off if the system is not a termination station. (Disables termination for COM1 terminal)	
2	1	ON	ON	For RS-485 communication, set both pins 1 and 2 to On if the system is a termination station. (Enables termination for COM2 terminal)	
		OFF	OFF	For RS-485 communication, set both pins 1 and 2 to Off if the system is not a termination station. (Disables termination for COM2 terminal)	

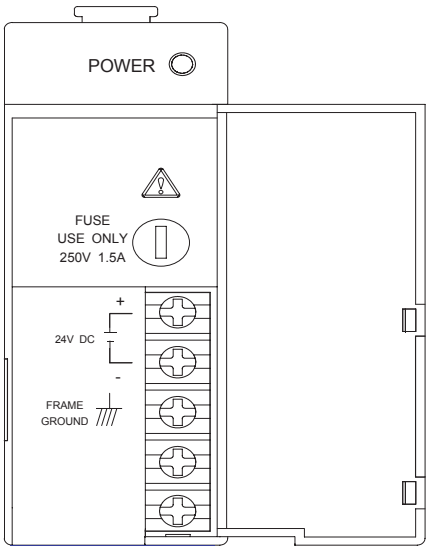
- Switch for communication and program booting method setting (DIP switch 2)

Pin No.		Pin setting		Description	DIP Switch 2
8	7	OFF	OFF	Sets the communication speed on COM2 terminal to 9600 bps	
		ON	OFF	Sets the communication speed on COM2 terminal to 19200 bps	
		OFF	ON	Sets the communication speed on COM2 terminal to 38400 bps	
		ON	ON	Sets the communication speed on COM2 terminal to 4800 bps	
6	5	OFF	OFF	Sets the communication speed on COM1 terminal to 9600 bps	
		ON	OFF	Sets the communication speed on COM1 terminal to 19200 bps	
		OFF	ON	Sets the communication speed of COM1 terminal to 38400 bps	
		ON	ON	Sets the communication speed on COM1 terminal to 4800 bps	
4		ON		Selects RS-485 communications for COM1	
		OFF		Selects RS-232C communications for COM1	
3		ON		Selects RS-485 communications for COM2	
		OFF		Selects RS-232C communications for COM2	
2		OFF		Always set to Off. (Reserved for system setting)	
1		ON		Loads the program from EEPROM (flash ROM) at power-on.	
		OFF		Operates the system with the program in RAM at power-on.	

Power Supply Module



NX-POWER



NX-PWRDC

- **Hardware features**

1. **Power status LED**
   Turns on when power is on.
2. **Power fuse holder**
3. **Terminal block**
   Terminal block for power wiring. Crimped terminal for M3.5 can be used. Detachable.
4. **Power input terminal**
   110-240V AC Free Voltage power input terminal (NX-PWR220 is only for 220V ac).
5. **Frame ground (FG) terminal**
   As a grounding terminal, it is connected to the metal part of the backplane. Use triple grounding to prevent electric shock.
6. **Service power terminal (24V DC)**
   DC power supply to input and output unit.

ATTENTION

This service power terminal should not be connected to any commercial power supplies in parallel because it causes an error.

- **Specifications**

Input type	AC input power		DC input power
Catalog number	NX-POWER	NX-PWR220	NX-PWRDC
Input rated voltage	110 to 220V AC Free Voltage	220V AC	24V DC
Allowed voltage range	85 to 264V AC	176 to 264V AC	24V DC ± 10%
Input power frequency	47 to 63Hz		
Inrush current	20A or less		
Rated output current	5V 4.0A, 24V 0.5A	5V 6.0A	5V 3.0A
Fuse	250V 1.5A		
Weight	350g	300g	320g

ATTENTION

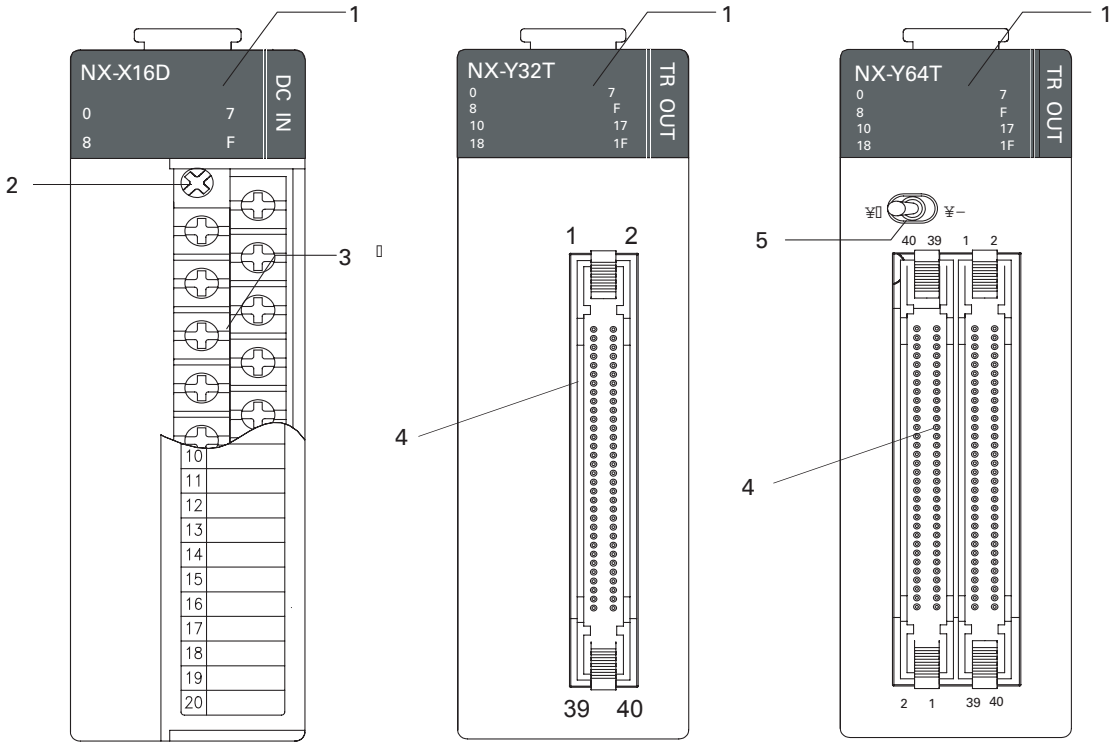
NX-POWER module does not guarantee protection against momentary power failure at 110V ac.

ATTENTION

Make sure that total current consumption of all modules mounted on the backplane does not exceed the rated current capacity of the power supply module.

- 5V power is supplied from the power supply module mounted on the same backplane.
- The 24V control power is used for supplying a dc power to I/O module.
- Do not connect this 24V control power with other commercial power supplies in parallel. It may cause error or product malfunction.
- A 250V 1A fuse is used for AC power. (The fuse is built into the power supply module.)
- Be careful that the 24V side on the external terminal does not exceed the current capacity. (It may cause system errors.)

I/O Modules



16-point type
(A Type)

32-point type
(B Type)

64-point type
(C Type)

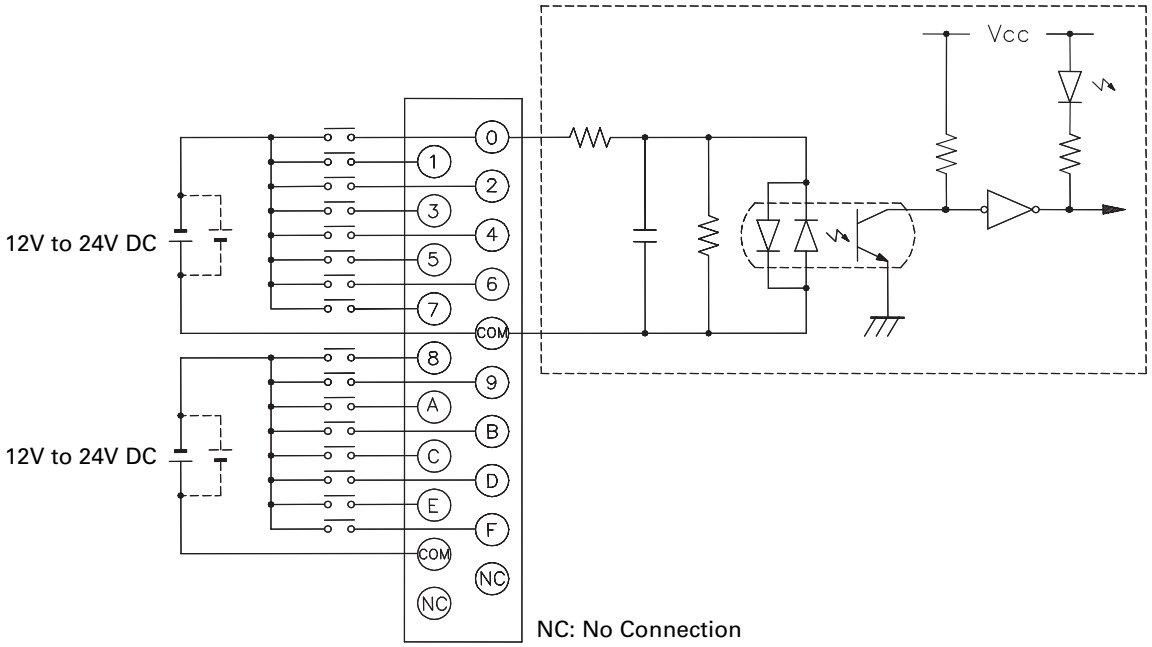
- **Hardware features**

1. **I/O Status LED**
   Shows I/O ON/OFF status.
2. **Terminal block fixing screw**
   Fixes detachable terminal block on the unit.
3. **Terminal block (20P, detachable terminal block)**
   Terminal block for I/O and power wiring. Crimped terminal for M3.5 can be used. See *"Terminal Block Type Module Wiring" in Chapter 4*.
4. **Connector 32 points (40P x 1), Connector 64 points (40P x 2)**
   Connector for I/O contact and power wiring. Use harness for PIN type or flat cable. See *"Connector Type Module Wiring" in Chapter 4* for details.
5. **Indication shift switch (64 points unit)**
   Convert the 64-point unit LED into front 32 points and rear 32 points.

Input unit specifications

Product name		DC input module
Catalog number		**NX-X16D**
Number of input points		16 points
Insulation method		Photocoupler
Rated input voltage		12 to 24V
Operating voltage range		10.2 to 26.4V
Max. input current		10mA or less (for 24V)
ON voltage/current		Min. 9.6V
OFF voltage/current		Max. 2.5V
Input impedance:		Approx. 3 KΩ
Response time	OFF → ON	10ms or less
	ON → OFF	10ms or less
Internal current consumption (5V)		65mA or less
Common method		8 points/1COM (Common for polarity +, -)
Operation indicator		LED
External cable connection type		Terminal block connection (M3.5), Detachable terminal block (20 pins)
Recommended cable size		0.5 to 1.25 mm^2
Capacity		170g
Types		(A) Type

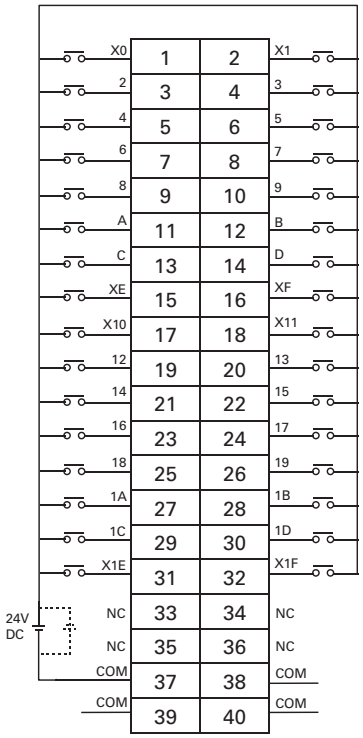
Internal circuit and wiring diagram



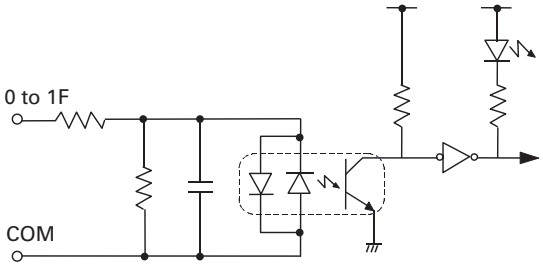
NC: No Connection

Product name		DC input module
Catalog number		**NX-X32D**
Number of input points		32 points
Insulation method		Photocoupler
Rated input voltage		12 to 24V
Use voltage range		10.2 to 26.4V
Max. input current		10mA or less (for 24V)
ON voltage/current		Min. 9.6V
OFF voltage/current		Max. 2.5V
Input impedance		Approx. 3 KΩ
Response time	OFF → ON	10ms or less
	ON → OFF	10ms or less
Internal current consumption (5V)		130mA or less
Common method		32 points/1COM (Common for polarity +, -)
Operation indicator		LED
External cable connection type		Connector (40 pin x 1 EA)
Recommended cable size		0.2 mm^2
Capacity		140g
Types		(B) Type

Wiring diagram



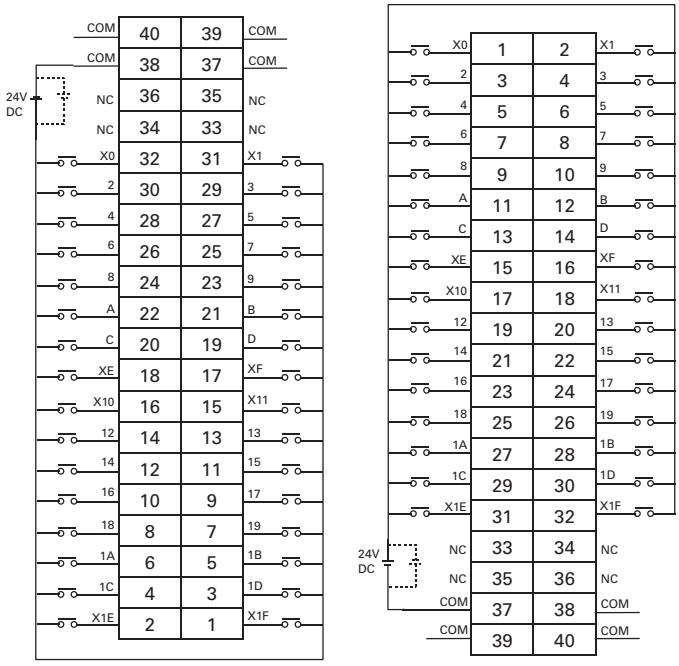
Internal circuit



- The COM terminal of a connector has a short inside of the unit.
- For wiring method, See *"Terminal Block Type Module Wiring" in Chapter 4*.
- Use separate commercial cable harness (NX-CBLDC) and Pin Type Ass'y (CPL8890) for external connection.
- NC: No Connection

Product name		DC input module
Catalog number		**NX-X64D**
Number of input points		64 points
Insulation method		Photocoupler
Rated input voltage		12 to 24V
Operating voltage range		10.2 to 26.4V
Max. input current		10mA or less (for 24V)
ON voltage/current		Min. 9.6V
OFF voltage/current		Max. 2.5V
Input impedance		Approx. 3 KΩ
Response time	OFF → ON	10ms or less
	ON → OFF	10ms or less
Internal current consumption (5V)		250mA or less
Common method		32 points/1COM (Common for polarity +, -)
Operation indicator		LED indication (display by 32 points conversion)
External cable connection type		Connector (40 pin x 2 EA)
Recommended cable size		0.2 mm^2
Capacity		200g
Types		(C) Type

Wiring diagram



(I) (II)

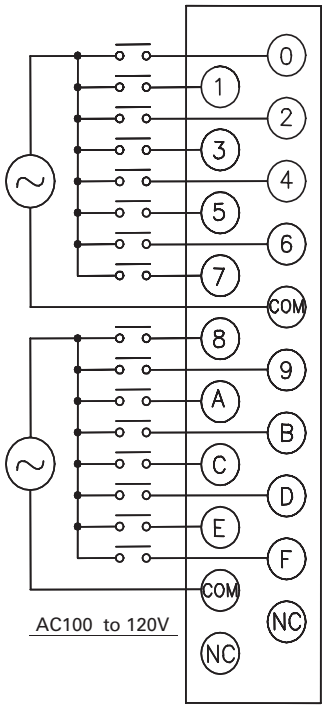
Internal circuit

Same as NX-X32D.

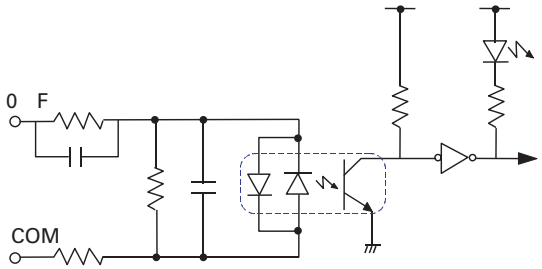
- The connectors [I] and [II] have opposite orientation each other, so be careful with wiring.
- The COM terminal of the connector has a short inside of the unit.
- For wiring method, See *"Terminal Block Type Module Wiring" in Chapter 4*.
- Use separate commercial cable harness (NX-CBLDC) and Pin Type Ass'y (CPL8890) for external connection.
- NC: No Connection

Product name		AC input module	
Catalog number		NX-X16A110	NX-X16A220
Number of input points		16 points	
Insulation method		Photocoupler	
Rated input voltage		AC 100 to 120V	AC 200 to 240V
Operating voltage range		AC 85 to 132V	AC 170 to 264V
Max. input current		20mA or less	
ON voltage/current		Min. 80V/6mA	Min. 160V/6mA
OFF voltage/current		Max. 30V/3mA	Max. 50V/3mA
Input impedance		Approx. 15KΩ	Approx. 20KΩ
Response time	OFF → ON	15ms or less	
	ON → OFF	20ms or less	
Internal current consumption (5V)		60mA or less	
Common method		8 points/1COM	
Operation indicator		LED	
External cable connection type		Terminal block connection (M3.5), Detachable terminal block (20 pins)	
Recommended cable size		0.5 to 1.25 mm^2	
Capacity		200g	
Types		(A) Type	

Wiring diagram



Internal circuit



NC: No Connection

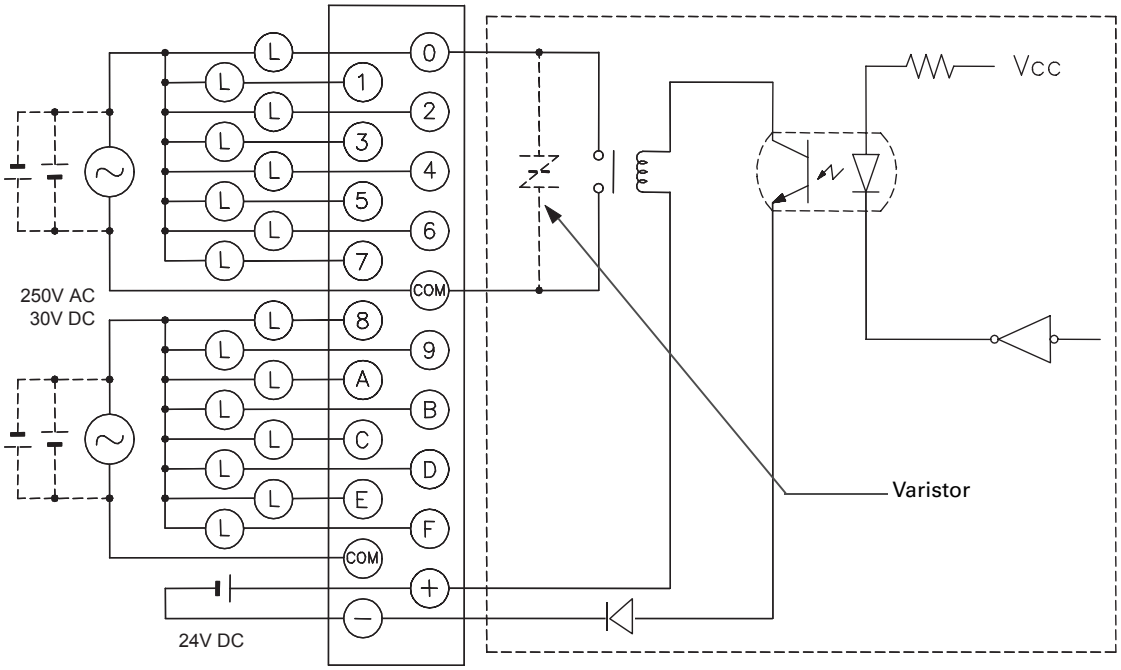
NOTE NX-X16A220: 200-240V AC

Output module specifications

Product name		RELAY output module	
Catalog number		**NX-Y16R**	**NX-Y16RV**
Number of output points		16 points	
Insulation method		Photocoupler	
Rated load voltage		2A 250V AC, 2A 30V DC	
Response time	OFF → ON	10ms or less	
	ON → OFF	10ms or less	
Life time	Mechanical	30 million times	
	Electric	200 thousand times	
External power supply		24V 150mA or less	
Surge protection		N/A	Varistor
Internal current consumption (5V)		120mA or less	
Common method		8 points/1COM	
Operation indicator		LED	
External cable connection type		Terminal block connection (M3.5), Detachable terminal block (20 pins)	
Recommended cable size		0.5 to 1.25 mm^2	
Capacity		225g	235g
Types		(A) Type	

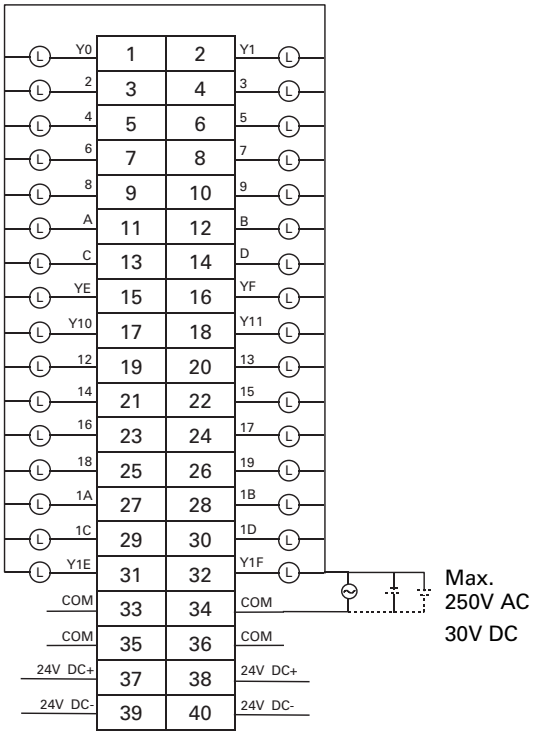
Internal circuit and wiring diagram

NX-Y16R: No varistor
NX-Y16RV: Varistor



Product name		RELAY output module
Catalog number		**NX-Y32RV**
Number of output points		32 points
Insulation method		Photocoupler
Rated load voltage		1A 250V AC, 1A 30V DC
Response time	OFF → ON	10ms or less
	ON → OFF	10ms or less
Life time	Mechanical	30 million times
	Electric	200 thousand times
External power supply		24V 150mA or less
Surge protection circuit		Varistor
Internal current consumption (5V)		180mA or less
Common method		32 points/1COM
Operation indicator		LED
External cable connection type		Connector (40 pin x 1 EA)
Recommended cable size		0.2 mm^2
Capacity		300g
Types		(B) Type

Wiring diagram



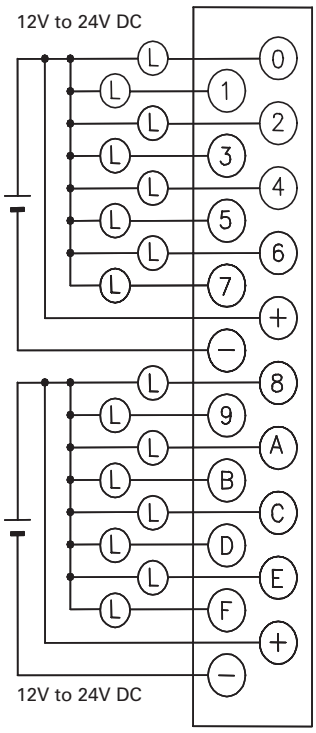
Internal circuit

Same as NX-Y16RV.

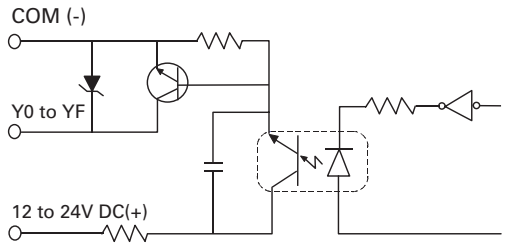
- The COM terminal of the connector has a short inside of the unit.
- For wiring method, See *"Terminal Block Type Module Wiring" in Chapter 4*.
- Use separate commercial cable harness (NX-CBLRY) and Pin Type Ass'y (CPL8890) for external connection.

Product name		TR output module
Catalog number		**NX-Y16T**
Number of input points		16 points
Insulation method		Photocoupler
Rated load voltage		12 to 24V DC
Operating load voltage range		10 to 30V DC
Max. load current		0.6A/point
OFF state leakage current		100 μA or less
Response time	OFF → ON	1ms or less
	ON → OFF	1ms or less
Internal current consumption (5V)		120mA or less
Surge absorber		Zener diode
Common method		8 points/1COM
Operation indicator		LED
External cable connection type		Terminal block connection (M3.5), Detachable terminal block (20 pins)
Recommended cable size		0.5 to 1.25 mm^2
Capacity		170g
Types		(A) Type

Wiring diagram

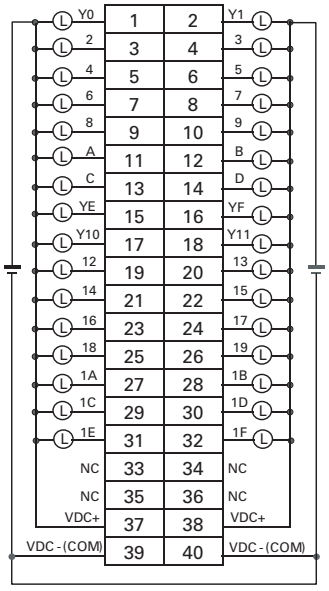


Internal circuit

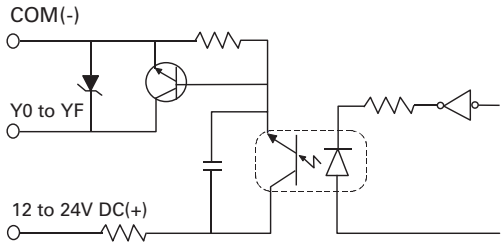


Product name		TR output module
Catalog number		**NX-Y32T**
Number of input points		32 points
Insulation method		Photocoupler
Rated load voltage		12 to 24V DC
Operating load voltage range		10 to 30V AC
Max. load current		0.4A/point
OFF state leakage current		100 μA or less
Response time	OFF $\rightarrow$ ON	1ms or less
	ON $\rightarrow$ OFF	1ms or less
Internal current consumption (5V)		180mA or less
Surge absorber		Zener diode
Common method		32 points/1COM
Operation indicator		LED
External cable connection type		Connector (40 pin x 1 EA)
Recommended cable size		0.2 mm^2
Capacity		140g
Types		(B) Type

Wiring diagram



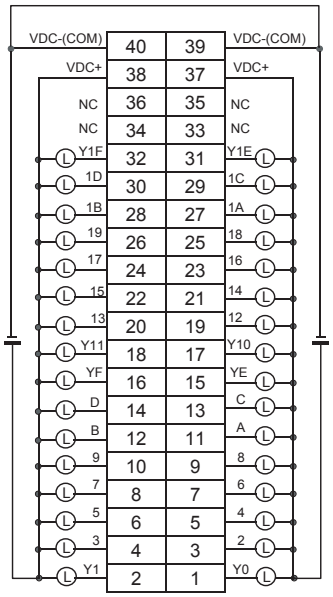
Internal circuit



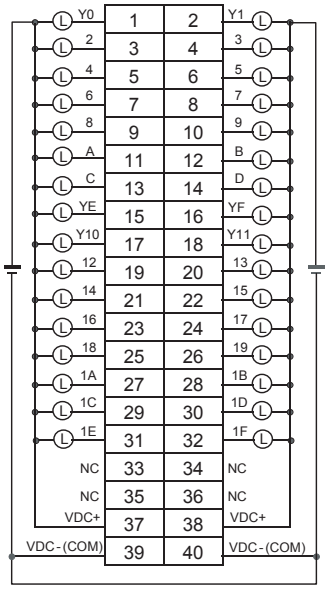
- COM (VDC-) and VDC+ (12 to 24V) terminals of the connector has a short inside of the unit.
- For wiring method, See *"Terminal Block Type Module Wiring" in Chapter 4*.
- Use separate commercial cable harness (NX-CBLTR) and Pin Type Ass'y (CPL8890) for external connection.
- NC: No Connection

Product name		TR output module
Catalog number		**NX-Y64T**
Number of input points		64 points
Insulation method		Photocoupler
Rated load voltage		12 to 24V DC
Operating load voltage range		10 to 30V AC
Max. load current		0.2A/point
OFF state leakage current		100 μA or less
Response time	OFF → ON	1ms or less
	ON → OFF	1ms or less
Internal current consumption (5V)		250mA or less
Surge absorber		Zener diode
Common method		32 points/1COM
Operation indicator		LED indication (display by 32 points conversion)
External cable connection type		Connector (40 pin x 2 EA)
Recommended cable size		0.2 mm^2
Capacity		205g
Types		(C) Type

Wiring diagram



(I)



(II)

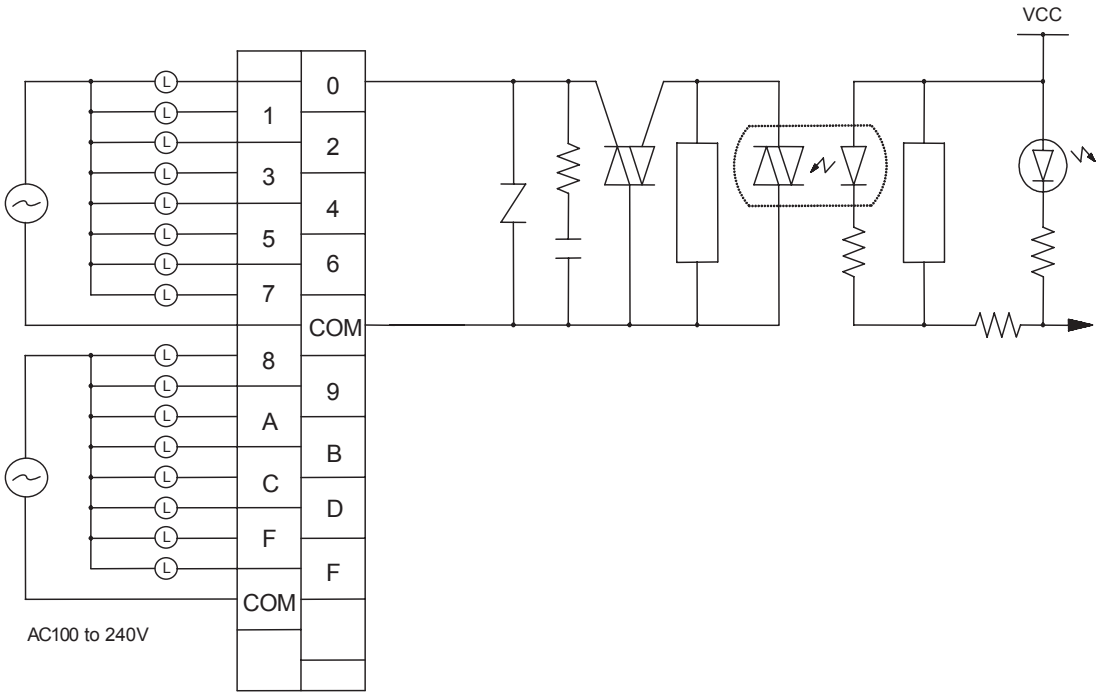
Internal circuit

Same as NX-Y32T.

- COM (VDC-) and VDC+ (12 to 24V) terminals of the connector has a short inside of the unit.
- For wiring method, See *"Terminal Block Type Module Wiring" in Chapter 4*.
- Use separate commercial cable harness (NX-CBLTR) and Pin Type Ass'y (CPL8890) for external connection.
- NC: No Connection

Product name		SSR output module
Catalog number		**NX-Y16SSR**
Number of input points		16 points
Insulation method		SSR
Rated load voltage		100 to 240V AC
Operating load voltage range		85 to 264V AC
Max. load current		0.5A/point
OFF state leakage current		100 μA or less
Response time	OFF → ON	1ms or less
	ON → OFF	0.5 CYCLE + 1ms or less
Internal current consumption (5V)		250mA or less
Fuse rating		3A
Common method		8 points/1COM
Operation indicator		LED
External cable connection type		Terminal block connection (M3.5), Detachable terminal block (20 pins)
Recommended cable size		0.5 to 1.25 mm^2
Weight		Approx. 240g
Types		(A) Type

Internal circuit and wiring diagram



3

Addressing Overview

Addressing Overview

All the memory used for external I/O processing and internal data processing has always both address and data (the content).

Addressing space is classified as R, L, M, K, F, TC, and W. These letters are used to designate a specific area in memory as shown in the following table.

Memory areas	Addresses	Description
External I/O area (R)	R000.0 to R127.15	• Local I/O memory area that can be set when configuring I/O module. • 2048 points, 128 words
	R640.0 to R127.15	• Remote I/O memory area (user defined area) • 1024 points, 64 words
Link contact (L)	L000.0 to L063.15	• Link contact shared memory area, 1024 points, Loop 0 • Can be used as internal contacts when not used as link contacts.
Internal contact (M)	M000.0 to M127.15	• Internal auxiliary contact memory area • 2048 points, 128 words
	M000.0 to M063.15	• When link loop 1 is used, this area can be used shared memory area for link contacts
Keep contact (K)	K000.0 to K127.15	• Retentive internal auxiliary contact memory area • 2048 points, 128 words
Special contact (F)	F000.0 to F015.15	• Special internal contact memory area • 256 points, 16 words
Timer/Counter (TC)	Contacts: TC000 to TC255 Set value: SV000 to SV255 Present value: PV000 to PV255	• The same 256 channels are used for contacts, set values, or preset values. • TC indicates "contact". • SV is Set Value, PV is Present Value. • The data value range of SV and PV is from 0 to 65535.
Data register (W)	W000 to W2047	• Area that retains the data in case of power failure • Bit addressing is not possible. • Cleared when downloading program
	W0000 to W0127	• Can be configured as shared memory area for loop 0 link contacts
	W0128 to W2557	• Can be configured as shared memory area for loop1 link contacts
Special register (SR)	SR000 to SR511	• Special internal data area for processor status and RTC • Make sure to use available areas only described in this manual.

- The R, L, M, K, F, and TC areas can be used for both bit and word addressing.
- The W area can be used to process word data only.
- The L area can be used as internal contacts.
- Keep contact (K), data register (W), and counter's preset value register retain their last values before power was removed. Cleared when a new program is downloaded.

Bit and Word Addressing

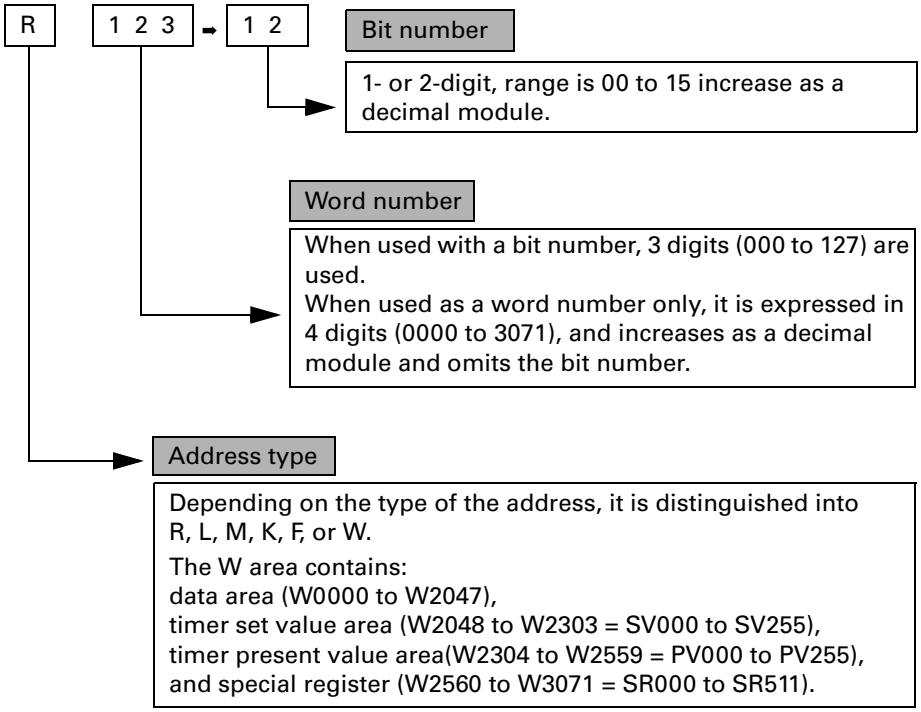
A bit address is composed of a character (R, L, M, K, F) that identifies its type, a five digit word address (0.0 to 127.15, increases by 0 to 15). The timer/counter contact is represented by the TC label followed by three digits. The three digits indicate the channel number of the timer/counter (TC000 to TC255). TIM000 indicates a contact instead of TC000 for PGM10 and PGM-500.

A word address is composed of a character (R, L, M, K, W) and a four digit number (0000 to 2047). Special registers have alternative address representation. Special registers SR000 to SR511 can be also represented as W2560 to W3071.

Both of bit and word addresses can be used to address the memory areas of R, L, M, K. However, be cautious that instructions use a specific type of address, either bit or word address, and the used addresses are resolved automatically depending on the type of instruction.

A bit can have the content of either On (1) or Off (0) state. A word is composed of 16 bits and holds a data value from 0 to 65,535. A double word is composed of 32 bits and holds a data value from 0 to 4,294,967,295.

Expression example

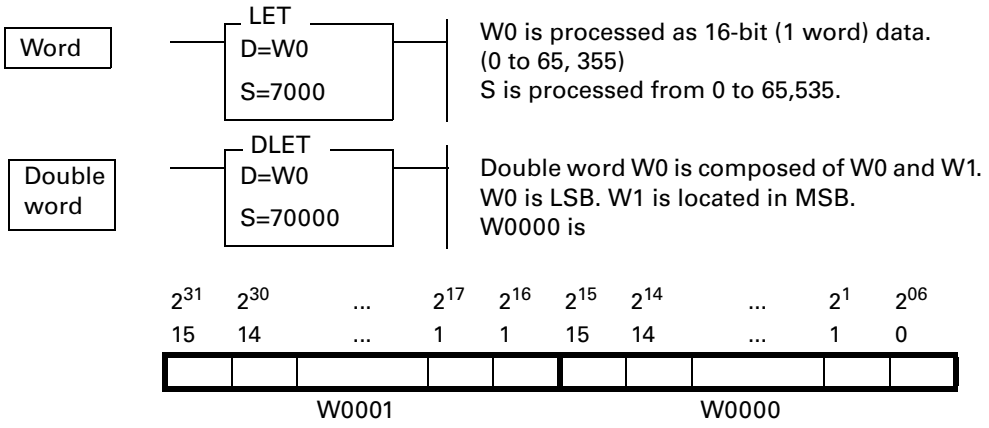


Double Word Addressing

Double word addressing is same with word addressing, except that 32-bit data is referenced by the specified address and its next address.

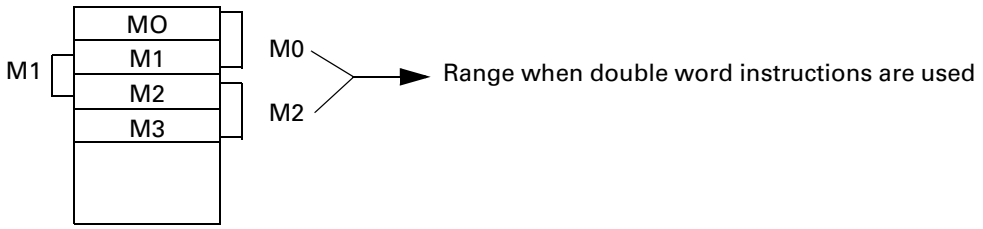
The type of instruction used determines which addressing, word or double word addressing, is applied. For comparison instructions, the programmer must be in “Double Mode” to enter a double-word comparison. For other instructions, those instructions that start with a D in front of the related word instruction are double word instructions, and the data is processed as double words

Example 1:



The value of D or S of the instruction can process data from 0 to 4,294,967,295.

Example 2:



Absolute Addressing

In LDR, DLDR, STO, and DSTO instructions, an absolute address is used to indirectly reference a register or to utilize the built-in communication port.

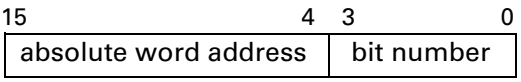
Classification	Register address	Absolute address	
		Dec.	Hex.
External I/O	R0	0	$0000
	R1	1	$0001
	R2	2	$0002
	...	...	...
	R126	126	$007E
	R127	127	$007F
Link area	L0	128	$0080
	L1	129	$0081
	L2	130	$0082
	...	...	...
	L62	190	$00BE
	L63	191	$00BF
Internal contact	M0	192	$00C0
	M1	192	$00C1
	M2	194	$00C2
	M3	195	$00C3
	...	...	...
	M126	318	$013E
	M127	319	$013F
Internal Keep contact	K0	320	$0140
	K1	321	$0141
	K2	322	$0142
	K3	323	$0143
	...	...	...
	K126	446	$01BE
	K127	447	$01BF

Classification	Register address	Absolute address	
		Dec.	Hex.
Special Internal contact	F0	448	$01C0
	F1	449	$01C1
	F2	450	$01C2
	...	...	...
	F126	462	$01CE
	F127	462	$01CF
Data area	W0	512	$0200
	W1	513	$0201
	W2	514	$0202
	...	...	...
	W2046	2558	$09FE
	W2047	2559	$09FF
T/C set value	SV000	2560	$0A00
	SV001	2561	$0A01
	...	...	...
	SV255	2815	$0AFF
T/C present value	PV000	2816	$0B00
	PV001	2817	$0B01
	...	...	...
	PV255	3071	$0BFF
Status	SR000	3072	$0C00
	SR001	3073	$0C01
	...	...	...
	SR511	3583	$0DFF
Expansion data area(1)	W3072	3584	$0E00
	...	...	...
	W5119	5631	$15FF

(1) Expansion data area is the memory space of 2048 words specially designed for NX-CPU700p

An absolute bit address, which is often used in communication, consists of an absolute word address and a bit number (0 to 15, represented as $0 to $F) as shown below.

absolute bit address

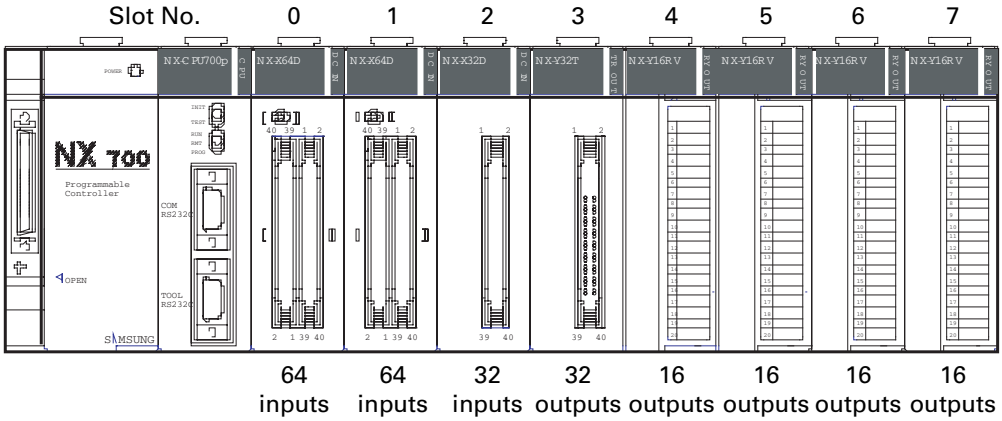


For example, the absolute bit address for internal contact K127.12 is $1BFC (hex). ($1BFC = absolute word address $01BF + bit number $C)

I/O Addressing

Addressing is based on the location of the module.

Example 1: 8-slot system



Example of addressing for the system shown above

Slot number		00	01	02	03	04	05	06	07
Word address	Processor module	R0 to R3	R4 to R7	R8 to R9	R10 to R11	R12	R13	R14	R15
Bit address		R0.0 R0.1 R0.2 ~ R0.15	R4.0 R4.1 R4.2 ~ R7.15	R8.0 R8.1 R8.2 ~ R9.15	R10.0 R10.1 R10.2 ~ R11.15	R12.0 R12.1 R12.2 ~ R12.15	R13.0 R13.1 R13.2 ~ R13.15	R14.0 R14.1 R14.2 ~ R14.15	R15.0 R15.1 R15.2 ~ R15.15

I/O addressing guidelines

The processor automatically distinguishes input address from output address for each module mounted.

A word address is allocated to a 16-point input or output module. For a 32-point mixed I/O module, 2 word addresses are allocated, a word address for inputs and another for outputs. For the 16-point mixed I/O module, 2 word addresses are allocated but only the lower 8 bits (0 to 7) of each word address can be used.

The modules mounted into the slots that are closer to the processor module have lower addresses.

An empty slot means no module is mounted and a blank module (NX-DUMMY) may be installed.

Occupied I/O points by module (unit: word)

The table below shows the occupied I/O points for each module when the memory space is allocated automatically by the system.

Catalog number[a]	I/O occupied points [b]	Input[b]	Output
NX-X16D	1	1	
NX-X16A110	1	1	
NX-X16A220	1	1	
NX-X32D	2	2	
NX-X64D	4	4	
NX-Y16R	1		1
NX-Y16RV	1		1
NX-Y16T	1		1
NX-Y16SSR	1		1
NX-Y32RV	2		2
NX-Y32T	2		2
NX-64T	4		4
NX-AI8C	8 (1)	8 (1)	
NX-AI8V	8 (1)	8 (1)	
NX-AO4C	4 (1)	(1)	4
NX-AO4V	4 (1)	(1)	4
NX-AO8C	8 (1)	(1)	8
NX-AO8V	8 (1)	(1)	8
NX-RTD8	8 (1)	8 (1)	
NX-RTD4	4 (1)	4 (1)	
NX-TC8	8 (1)	8 (1)	
NX-TC4	4 (1)	4 (1)	
NX-HSC1	2	1	1
NX-HSC2	2	1	1
NX-HSC4	4	2	2
NX-PULSE4	4	2	2
NX-POSI1	4	2	2
NX-POSI2	4	2	2
NX-POSI3	8	4	4
NX-POSI4	8	4	4
NX-CCU+	0	Not Applicable	Not Applicable
NX-SCU	2	1	1
NX-MASTER	0	Not Applicable	Not Applicable
NX-MWLINK	0	Not Applicable	Not Applicable
NX-DEVICE	2	1	1

(a) For digital I/O and analog modules, manual addressing is possible and the size of allocated space can be larger than the I/O occuplied word presented in the table above.

(b) The number in parenthesis indicates the number of words occupied when the shared memory method is used for addressing.

Special Registers

Word registers F000 to F015

Address	Function	Description
F000 register	System check/control	System self-check/program checking, operation control
F001 register	System check/clock	0.01/0.02/0.1 ms timer output, calculation results, carry flag
F002 register	Link control	Link installation and operation mode setting
F003 register	Link control	Link installation and operation mode setting
F004 register	Link status flag	Link participating station information
F005 register	Link status flag	Link participating station information
F006 register	Link status flag	Link data receiving information flag
F007 register	Link status flag	Link data receiving information flag
F008 register	Remote control flag	Remote operation control flag
F009 register	Remote control flag	Remote operation control flag
F010 register	Remote control flag	Remote operation control flag
F011 register	Remote control flag	Remote operation control flag
F012 register	User-defined communication flag	For port COM2 User-defined communication control flag
F013 register	System reservation	
F014 register	PID control	PID operation mode and operation/stop control flag
F015 register	PID control	PID operation mode and operation/stop control flag

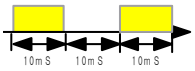
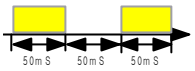
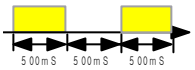
Word register F000 (F0.0 to F0.15)

Only a bit process is available.

Address	Function	Description	Normal status
F0.00	System check	When the power is applied, the system self-checks the ROM. Should any fault exist, the error lamp is turned on. Output and operation are halted.	OFF
F0.01	CPU ROM check	When the power is applied, the system self-checks the ROM. Should any fault exist, the error lamp is turned on. Output and operation are halted.	OFF
F0.02	CPU RAM check	When the power is applied, the system self-checks the RAM. Should any fault exist, the error lamp is turned on. Output and operation are halted	OFF
F0.03	User program memory error	If the user program memory is damaged or the program is faulty, the error lamp is turned on. Output and operation are halted.	OFF
F0.04	Program syntax error	The CPU initially runs and checks the user program's syntax. In the case of an error, the error lamp is turned on. Output and operation are halted.	OFF
F0.05	Module range error	Indicates an invalid R word (>64) used.	OFF
F0.06	Module change error	On when an I/O module is removed/added/fails while the system is running. The error lamp is on and the CPU keeps running. Turned off when the error is corrected.	OFF
F0.07	Module type error	On when the I/O module information that is stored in the CPU and module that is installed are different types, the error lamp is turned on. Operation stops.	OFF
F0.08	Input data control	Off when the running CPU input module's data is not updated. (Input update is turned Off.)	ON
F0.09	Output data control	Off to suspend updating of the output modules while the CPU is in the Run state. (Output update is turned Off). The outputs are maintained in their last valid state prior to update being disabled.	ON
F0.10	All output off	Turns all outputs off while CPU is in the Run state. (Outputs are disabled)	ON
F0.11	Constant cycle interrupt	On when the constant cycle interrupt instructions are used. (Refer to the INT instructions.) (The cycle time is defined by the user.)	OFF
F0.12	Watchdog error	On when a scan time exceeds the watchdog set time.	OFF
F0.13	Disable module type checking	On when the CPU starts the initial Run and the program is checked without performing I/O module type verification.	
F0.14	Program changes during running	On when a user corrects the program while in Run mode. If there are any syntax errors, the CPU is stopped.	
F0.15	Run state control	On when the CPU is in the Run state. Off when stopped or paused.	ON

Word register F1 (F1.0 to F1.15)

Only a bit process is available.

Address	Function	Description	Remarks
F1.0	First single scan	Maintain On state for first single-scan period, when the CPU shanges its status stop to Run.	
F1.1	Scan clock	Cycle On/Off state for each scan during the program. (1Scan On, 1Scan Off)	
F1.2	0.02-second clock	10 ms: On, 10 ms: Off 	
F1.3	0.1-second clock	50 ms: On, 50 ms: Off 	
F1.4	1-second clock	500 ms: On, 500 ms: Off 	
F1.5	Instantaneous interrupt	On when power is off for 20 ms or more.	Maintained
F1.6	CPU running status	On when the CPU is in the run state.	
F1.7	Keep area error display	On when the K retentive data is destroyed and/or changed.	
F1.8	Carry flag	On in the event of carry when performing math instruction.	
F1.9	Division by zero error	On when the denominator of division commands is zero.	
F1.10	Range designation error	On when the absolute address exceeds the specified range.	
F1.11	Reserved	System use area	Do not use
F1.12	W area error indication	System use area	
F1.13	Reserved	System use area	Do not use
F1.14	Reserved	System use area	Do not use
F1.15	Reserved	System use area	Do not use

NOTE The 16 bits in the F1 address provide the CPU's special function and self-diagnosis result. They are used for status contacts only, and are not used to modify or control the PLC. Only the F1.5 instantaneous interrupt display contact should be used as an output contact by the user, to be turned off after power loss indication.

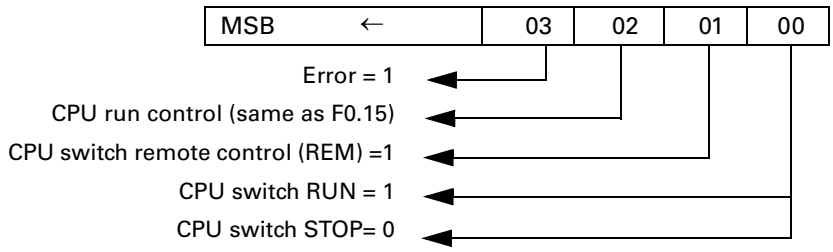
Word register F12 (F12.0 to F12.15)

Only a bit process is available.

Address	Function	Description	Remarks
F12.0	RTC check	On when the RTC is enabled.	-
F12.2	Flash	On when the 9.6 KW of flash memory is installed.	-
F12.3	Flash	On when the 16 KW of flash memory is installed.	-
F12.5	Battery error	On when the battery is not connected or the voltage is lower than the backup voltage.	-
F12.7	Periodical scan error	On when any error exists in the periodical program.	
F12.10	RTC set error	On when there is an error setting the RTC.	Output
F12.11	Saving Flash and checking boot	Off when the program is normally saved into Flash by setting F12.15 On.	Output
F12.12	EEPROM backup check	On when the program in EEPROM is the same with that in RAM.	
F12.13	RTC Set 1	On when you change the year, month, day, or date. Off when the data set is normal.	I/O
F12.14	RTC Set 2	On when you change the hour, minute, or second. Off when the data set is normal.	I/O
F12.15	Saving Flash	On when you save the current program into the flash. Off when the data set is normal.	I/O

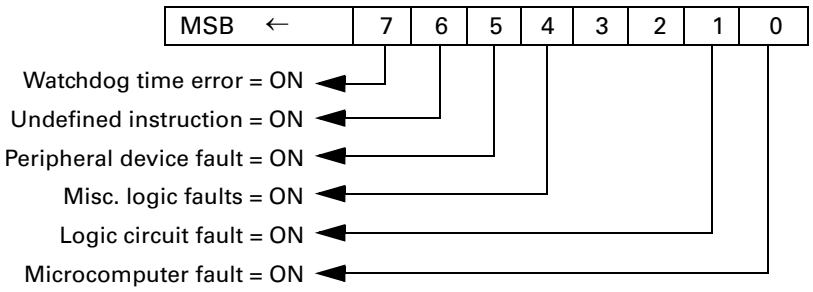
Special registers SR0 to SR016 (W2560 to W2576)

May be modified - each is composed of 1 word.

Address	Function	Description
SR000	CPU ID number	Indicates the CPU ID number in the lower 8 bits. 0 to 223 are the valid user-defined values, 255 is the default value.
SR001	CPU state	Indicates current CPU information state. (run/stop/remote control/error) 
SR2	Watchdog time	Indicates the user program watchdog time (unit: msec)
SR3	Scan time	Indicates the scan time when executing a program. Updated at every scan (unit: msec)
SR4	Max. scan time	Indicates maximum value of scan time when executing a program.
SR5 to SR7	Link module number	Link module number as set by the link # 1, 2, 3
SR8	PID table	PID register block start address
SR9 to SR10	Reserved	System use area
SR11 to SR16	Remote I/O information	Remote setup area

Special registers SR017 to SR511 (W2577 to W3071)

May be changed - each is composed of 1 word.

Address	Function	Description
SR017	System error information	Gives result of self-diagnosis by CPU. Indicates error content . 
SR018	Location of undefined instruction	Indicates the location of the instruction (the step number) that caused an undefined instruction error during program execution.
SR019	Reserved	System use area
SR020	Multiplication	Stores high order bit values upon executing 16 bit multiplication instructions.
SR021	Lower remainder	Stores the low order bit values of remainder after a division instruction has been executed.
SR022	Higher remainder	Stores the high order bit values of the remainder after a division instruction has been executed.
SR023 to SR027	Reserved	System use area
SR028 to SR029	Defective slot information	Location of defective slots mounted onto the basic slot
SR030 to SR48	Reserved	System use area (syntax information and system information)
SR49 to SR79	Slot information	Stores slot information for installed I/O modules.
SR261 to SR279 *	Remote control domain	Contains remote I/O configuration data.
SR289 to SR297	RTC	Contains real time clock information (year, month, day, hour, minute, second, date).
SR298 to SR373 *	User-defined communication area	For port COM2 User-defined communication area
SR374 to SR379 *	Link error information	Link error information data.
SR380 to SR511	Reserved	System use area

* Applied when using wire link modules

Program syntax error status register SR30 (W2590)

Indicates the result of the automatic check on the user program syntax when the programmer or GPC executes a syntax check, and when the operation mode is switched from the Stop state to the Run state. If the value of W2590 is not zero, F004 bit turns On. The error lamp also turns On.

Error correction method:

Find the error in the CPU online mode and then correct the program.

Word	Bit	Description
SR30	0	On if the I/O number range of bit process instruction is beyond the specified range or designates an external contact/output module which is not installed.
	1	On if the channel number of the timer or the counter exceeds 255 or is duplicated.
	2	On if the bit or word number in the advanced instruction is beyond the specified range or if it designates a module which is not installed.
	3	On if a word number in the refresh instruction (INPR, OUTR) is beyond the specified range, or if it designates a module which is not installed.
	4	On if an undefined instruction exists.
	5	On in event of a user program memory writing error.
	6	On in event of miscellaneous errors.
	7	On if the user program memory is abnormal.
	8	On if an error on external I/O address and bit/word/double word numbers used occurs. For example, the first slot is set with an input module and OUT R00001 is designated.
	9	On if the label numbers of the JMP or CALL instructions exceed 63, the corresponding instruction LBL or SBR does not exist, and/or the corresponding LBL/SBR instructions exist prior to JMP/CALL instructions.
	10	On if the label number of the LBL instruction exceeds 63 and/or is duplicated.
	11	On if the JMPS/JMP instructions are mistakenly combined and/or used.
	12	On if the FOR/NEXT instructions are mistakenly combined and/or used more than four times. (Loop)
	13	On if SBR/RET instructions are not combined and/or used and/or the SBR instructions overlap or exceed 63.
	14	On if NT/RETI instructions are not combined and/or used and/or more than two sets of INT instructions are used.
	15	On if no END instruction inserted automatically.

Real-time clock registers SR289 to SR297 (W2849 to W2857)

Sets the time of the built-in clock (RTC) and stores and displays the present time. Data is stored in BCD format.

(○ : bit = 0; × : bit change)

Classification	Address	Bit address / Adjustment/ indication	Details															
			15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Current time	SR289	Year (4-digit BCD)																
	SR290	Date: day	○	○	×	×	×	×	×	×	○	○	○	○	○	×	×	×
	SR291	Year: month	○	×	×	×	×	×	×	×	○	○	○	×	×	×	×	×
	SR292	Second: 00	○	○	×	×	×	×	×	×	○	○	○	○	○	○	○	○
	SR293	Hour: minute	○	○	×	×	×	×	×	×	○	○	○	○	○	○	○	○
Time setting	SR294	Date: day	○	○	×	×	×	×	×	×	○	○	○	○	○	×	×	×
	SR295	Year: month	○	×	×	×	×	×	×	×	○	○	○	×	×	×	×	×
	SR296	Second: 00	○	○	×	×	×	×	×	×	○	○	○	○	○	○	○	○
	SR297	Hour: minute	○	○	×	×	×	×	×	×	○	○	○	○	○	○	○	○

* Usable for NX70-CPU70p2 module only

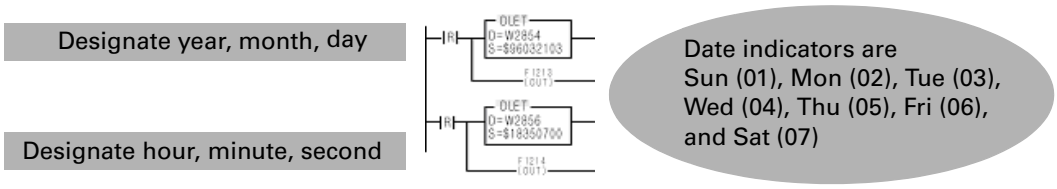
In SR289, the year can be read in a 4-digit BCD.

Ex) $1998=1998, $2000=2000

SR289 to SR297 for clock functions are shown in BCD, so it is convenient to confirm in HEX.

NOTE Set the range as follows:
Year: 00 to 99, Month: 01 to 12, Day: 01 to 31, Day of Week: 01 to 07 (Sun. to Sat.)
Hour: 00 to 23, Minute: 00 to 59, Second: 00 to 59

1. Ladder setting method:
   For example, current date and time are: Tuesday, March 21, 1996, 18:35:07



2. When changing the year, month, date, or day, new data is input in SR295, SR294, then the F12.14 bit is turned On. The F12.10 bit is kept Off.
3. When changing the hour, minute, and second, new data is input in SR297, SR296, then the F12.14 bit is turned on. If the new data is not set correctly, the F12.10 bit turns on.
4. The display date and set value are expressed in BCD so it is convenient to input as hex ($).
5. The year, month, and day are automatically changed.

Timer/Counter Area

Timer/counter set value and present value addresses

Channel	Set value (SV)	Present value (PV)
0	W2048	W2304
1	W2049	W2305
2	W2050	W2306
3	W2051	W2307
4	W2052	W2308
5	W2053	W2309
6	W2054	W2310
7	W2055	W2311
8	W2056	W2312
9	W2057	W2313
10	W2058	W2314
11	W2059	W2315
12	W2060	W2316
13	W2061	W2317
14	W2062	W2318
15	W2063	W2319
16	W2064	W2320
17	W2065	W2321
18	W2066	W2322
19	W2067	W2323
20	W2068	W2324
21	W2069	W2325
22	W2070	W2326
23	W2071	W2327
24	W2072	W2328
25	W2073	W2329
26	W2074	W2330
27	W2075	W2331
28	W2076	W2332
29	W2077	W2333
30	W2078	W2334
31	W2079	W2335
32	W2080	W2336
33	W2081	W2337
34	W2082	W2338
35	W2083	W2339
36	W2084	W2340
37	W2085	W2341
38	W2086	W2342
39	W2087	W2343

Channel	Set value (SV)	Present value (PV)
40	W2088	W2344
41	W2089	W2345
42	W2090	W2346
43	W2091	W2347
44	W2092	W2348
45	W2093	W2349
46	W2094	W2350
47	W2095	W2351
48	W2096	W2352
49	W2097	W2353
50	W2098	W2354
51	W2099	W2355
52	W2100	W2356
53	W2101	W2357
54	W2102	W2358
55	W2103	W2359
56	W2104	W2360
57	W2105	W2361
58	W2106	W2362
59	W2107	W2363
60	W2108	W2364
61	W2109	W2365
62	W2110	W2366
63	W2111	W2367
64	W2112	W2368
65	W2113	W2369
66	W2114	W2370
67	W2115	W2371
68	W2116	W2372
69	W2117	W2373
70	W2118	W2374
71	W2119	W2375
72	W2120	W2376
73	W2121	W2377
74	W2122	W2378
75	W2123	W2379
76	W2124	W2380
77	W2125	W2381
78	W2126	W2382
79	W2127	W2383

Channel	Set value (SV)	Present value (PV)
80	W2128	W2384
81	W2129	W2385
82	W2130	W2386
83	W2131	W2387
84	W2132	W2388
85	W2133	W2389
86	W2134	W2390
87	W2135	W2391
88	W2136	W2392
89	W2137	W2393
90	W2138	W2394
91	W2139	W2395
92	W2140	W2396
93	W2141	W2397
94	W2142	W2398
95	W2143	W2399
96	W2144	W2400
97	W2145	W2401
98	W2146	W2402
99	W2147	W2403
100	W2148	W2404
101	W2149	W2405
102	W2150	W2406
103	W2151	W2407
104	W2152	W2408
105	W2153	W2409
106	W2154	W2410
107	W2155	W2411
108	W2156	W2412
109	W2157	W2413
110	W2158	W2414
111	W2159	W2415
112	W2160	W2416
113	W2161	W2417
114	W2162	W2418
115	W2163	W2419
116	W2164	W2420
117	W2165	W2421
118	W2166	W2422
119	W2167	W2423

NOTE If you change the above registers while the program is running or program them incorrectly, errors or damage may occur. Be sure you understand the programming procedures of the timer/counter thoroughly.

Channel	Set value (SV)	Present value (PV)
120	W2168	W2424
121	W2169	W2425
122	W2170	W2426
123	W2171	W2427
124	W2172	W2428
125	W2173	W2429
126	W2174	W2430
127	W2175	W2431
128	W2176	W2432
129	W2177	W2433
130	W2178	W2434
131	W2179	W2435
132	W2180	W2436
133	W2181	W2437
134	W2182	W2438
135	W2183	W2439
136	W2184	W2440
137	W2185	W2441
138	W2186	W2442
139	W2187	W2443
140	W2188	W2444
141	W2189	W2445
142	W2190	W2446
143	W2191	W2447
144	W2192	W2448
145	W2193	W2449
146	W2194	W2450
147	W2195	W2451
148	W2196	W2452
149	W2197	W2453
150	W2198	W2454
151	W2199	W2455
152	W2200	W2456
153	W2201	W2457
154	W2202	W2458
155	W2203	W2459
156	W2204	W2460
157	W2205	W2461
158	W2206	W2462
159	W2207	W2463
160	W2208	W2464
161	W2209	W2465
162	W2210	W2466
163	W2211	W2467
164	W2212	W2468
165	W2213	W2469

Channel	Set value (SV)	Present value (PV)
166	W2214	W2470
167	W2215	W2471
168	W2216	W2472
169	W2217	W2473
170	W2218	W2474
171	W2219	W2475
172	W2220	W2476
173	W2221	W2477
174	W2222	W2478
175	W2223	W2479
176	W2224	W2480
177	W2225	W2481
178	W2226	W2482
179	W2227	W2483
180	W2228	W2484
181	W2229	W2485
182	W2230	W2486
183	W2231	W2487
184	W2232	W2488
185	W2233	W2489
186	W2234	W2490
187	W2235	W2491
188	W2236	W2492
189	W2237	W2493
190	W2238	W2494
191	W2239	W2495
192	W2240	W2496
193	W2241	W2497
194	W2242	W2498
195	W2243	W2499
196	W2244	W2500
197	W2245	W2501
198	W2246	W2502
199	W2247	W2503
200	W2248	W2504
201	W2249	W2505
202	W2250	W2506
203	W2251	W2507
204	W2252	W2508
205	W2253	W2509
206	W2254	W2510
207	W2255	W2511
208	W2256	W2512
209	W2257	W2513
210	W2258	W2514
211	W2259	W2515

Channel	Set value (SV)	Present value (PV)
212	W2260	W2516
213	W2261	W2517
214	W2262	W2518
215	W2263	W2519
216	W2264	W2520
217	W2265	W2521
218	W2266	W2522
219	W2267	W2523
220	W2268	W2524
221	W2269	W2525
222	W2270	W2526
223	W2271	W2527
224	W2272	W2528
225	W2273	W2529
226	W2274	W2530
227	W2275	W2531
228	W2276	W2532
229	W2277	W2533
230	W2278	W2534
231	W2279	W2535
232	W2280	W2536
233	W2281	W2537
234	W2282	W2538
235	W2283	W2539
236	W2284	W2540
237	W2285	W2541
238	W2286	W2542
239	W2287	W2543
240	W2288	W2544
241	W2289	W2545
242	W2290	W2546
243	W2291	W2747
244	W2292	W2548
245	W2293	W2549
246	W2294	W2550
247	W2295	W2551
248	W2296	W2552
249	W2297	W2553
250	W2298	W2554
251	W2299	W2555
252	W2300	W2556
253	W2301	W2557
254	W2302	W2558
255	W2303	W2559

NOTE
- Channel: The inherent number of the timer and the counter (numbers that are equivalent to 000 of TC000).
- Set Value (SV): The designated value for the timer (to turn On) and the counter (number of times On) to start operation.
- Present Value (PV): Current processing value of the timer (elapsed time) and the counter (number of counts).

Address (register)

Address refers to the location of memory being used. It can refer to the external I/O module and internal memory.

An address is categorized into 1 bit, 16 bits (word), or 32 bits (double word).

Bit

A bit is the minimum module required for calculation. It can be either On (1) or Off (0).

Byte

A byte is made up of 8 bits. It can hold data values from 0 to 255. In base 16, or hexadecimal, a byte can be expressed as 0 to FF. You cannot have a value greater than 255 when using one byte.

Word

A word is made of 16 bits. It can hold data values from 0 to 65,535. In base 16 a word can be expressed as 0 to FFFF.

NX PLCs set R, M, K, F, and W areas into word areas and can be processed without any separate measures.

Double Word

A double word is made of 32 bits. It can hold data values from 0 to 4,294,976,295. In base 16, a double word can be expressed as 0 to FFFFFFFF. In the D32LT, a double word is made up of two consecutive word addresses as follows:
Double word address = Start word address + Next word address.

Example: When using W003,
W003 (double word address) = W003 (start word address) + W004 (next word address)

Scan Time

The CPU follows a procedure in which it 1) reads the inputs, 2) processes the ladder program, and 3) updates the outputs. It continually repeats this process. This 3-step process is called a "scan," and the time it takes to complete this process is the "scan time." In a typical PLC application, most of the scan time is used to process the program. When programming, keep in mind that the scan time will increase as you increase the number of inputs and outputs and/or the size of the program

Edge

An edge is defined as the point when an input changes state. For example, a rising edge occurs during the very first scan after the input has changed from Off to On. A falling edge occurs after the input has changed from On to Off.

BCD (Binary Coded Decimal)

BCD is used to express a decimal digit (0 to 9) using 4 bits. Conversion of BCD values can be done in hexadecimal calculations.

Example: 59 (BCD) = 59 (HEX), 32 (BCD) = 32 (HEX)

Flash ROM

It refers to a ROM (EEPROM) that stores programs. Since its contents can be deleted periodically, it is frequently used for equipments that deals with programs such as a PLC.

How to use a register

- **R (Relay) register (Can be bit, word or double word)**
  Indicates the external I/O register that directly connects to the general I/O module
- **M (Memory) register (Can be bit, word or double word)**
  An internal bit memory address which supports the relay of logical operations. It is used as a word or double-word variable for general calculations and programs.

  When the power of the PLC is turned off or the CPU has stopped, the register value is reset to 0.
- **W (Word) register (Can be bit, word or double word)**
  Same usage as M registers. The value is preserved after the power is turned off, but can be cleared by program downloads or special command words.

NOTE Not used as a bit (register only for word use)

- **K (Keep) register (Can be bit, word or double word)**
  Used for general calculation programs. The value is preserved after the power is turned off.
- **F (Flag) register (Only process bit)**
  These bit registers provide special application specific functions to the programmer of the PLC. They are also used as system control bits, providing Run/Stop control of the PLC.
- **When a calculation or input value exceeds 65,535 (FFFF)**
  Use a double word instruction that can store and calculate values over 65,535 in the K, M, R, and W registers. When a double word instruction is used, it can represent values up to 4,294,967,295 (2^{32}).
- **When you want to reload the contents or values that you worked today**
  Use the K and W area. These are preserved unless specifically erased. The W area is erased by special instructions or program downloads.
- **When you need numerical expressions such as A+B=C, 34×45=D, A1>C1.**
  Use the R, M, W, and K area. If you use the R area, you can refer to and output calculated values in I/O modules.

- **For bit operations, such as setting, resetting, shifting, or rotating**
  Use the M, K, and R area. You cannot perform bit operations in the W area.
- **When you want to refer to or modify the set value of the timer or counter.**
  Refer to or modify W2048 to W2303 or SV0 to SV255.
- **When you want to refer to or modify the present value of the timer or counter.**
  Refer to or modify the address area from W2304 to W2559 or from PV0 to PV255. The value holds true in STOP (PROG.) state in this area.
  The present value (PV) of the counter is maintained even after the power is off.

Processor Operation Mode

What is the Processor operation mode?

The processor has an external RUN/REMOTE/PROG switch. The PLC performs a system check that determines the position of the switch. The switch position determines which operating mode the PLC is in. It can set to RUN, STOP, REMOTE, or ERROR mode.

RUN Mode (operating)

The PLC reads the external contact signals in Run mode and executes the user program stored in RAM. The external outputs are updated every scan according to program results.

STOP Mode

The user program is stopped and the external outputs are turned Off. In the Stop mode, you may correct, delete or transfer the program.(This is the only mode in which you can save a program in flash memory.)

PAUSE Mode

A user program is operated at every scan and the I/O and result value is maintained. This mode is used when checking and debugging a program at every scan. This mode is similar to the Stop mode, but it does not initialize data.

ERROR Mode

It occurs when the CPU module finds internal defects after running the self-diagnoses. Functions are not executed normally in this mode. When an error occurs, the CPU stops all programs and turns all output off. When an Error mode occurs, the user should check the error code and take appropriate measures. Then turn the power on or put the mode conversion switch in Prog and press the Initialize Key to clear the error.

Allowed Functions in Operation Modes

☼=On, ◑ =Flashing, ● =Off

Operation mode selector switch	Operation mode	LED status RUN LED	LED status PROG LED	Program change	Data change	Initialize switch is enabled	Mode at power off-on
RUN	RUN	☼	●	Disallowed	Allowed	○	Run
	STOP	●	☼	Allowed	Allowed	○	Run
REMOTE	RUN	☼	☼	Allowed	Allowed	○	Run
	PAUSE	◑	☼	Allowed	Allowed	○	Pause
PROG	STOP	●	☼	Allowed	Allowed	○	Stop

When the PROG.LED is on, you can change the user program.

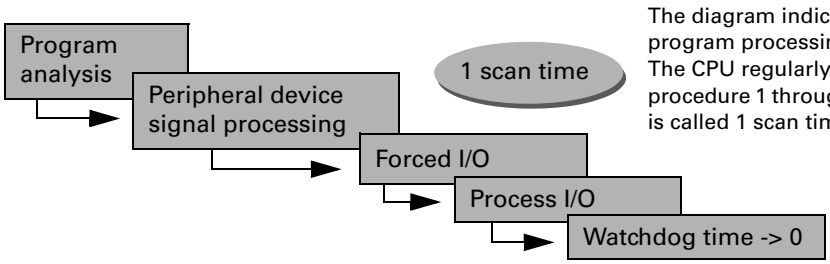
The INITIALIZE switch clears errors when the mode switch is set to PROG.

When the mode switch is set to REMOTE and power is switched from Off to On, the previous mode of operation is restored.

When debugging the user program, the mode switch should be set to REMOTE.

Processor Processing Procedure

Program processing procedure

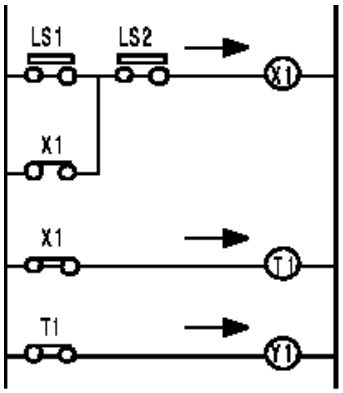


The diagram indicates the PLC program processing procedure. The CPU regularly repeats procedure 1 through 5. This cycle is called 1 scan time.

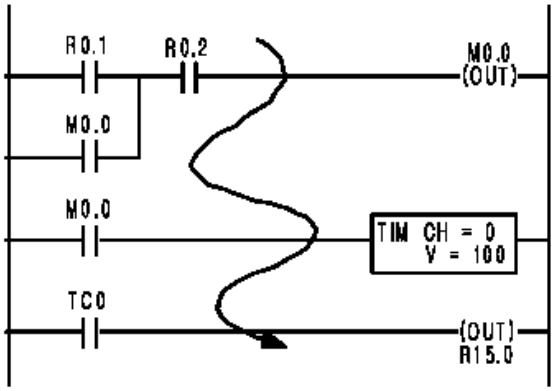
- **Program analysis**
  Executes the program from its first step to its final step and stores the internal/external output in the working RAM.
- **Peripheral device signal processing**
  Stores data from the communication module or peripheral device to the internal memory.
- **Forced I/O processing**
  Turns on/off forced I/O bits, if any.
- **I/O processing**
  Preserves the On/Off state of the external I/O and uses it as an input in the next scan. (For accurate processing, input should continue for more than 1 scan time.)
  The processed program outputs are sent from internal memory to the external memory.

- **Watchdog time initialization**
  The watchdog elapsed time value is set to 0. (This value is the watchdog calculation point until the next scan.)

The following illustration shows the difference between the relay board and PLC sequence processing. The relay carries out all sequences simultaneously while the PLC processes sequentially throughout the program.



Processing of relay sequence
(Parallel process)



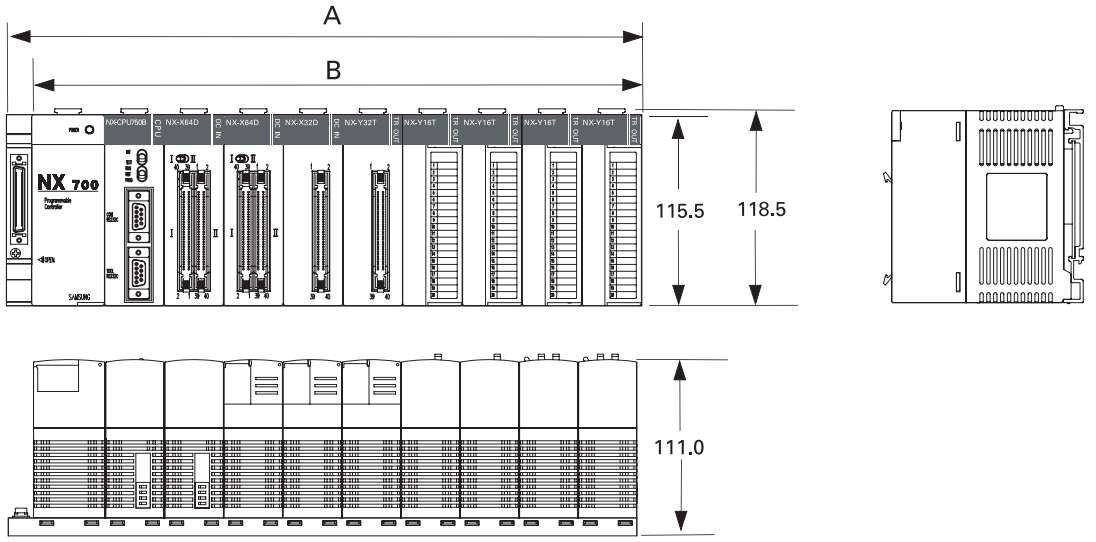
Processing of PLC program
(Serial process)

Installation and Wiring

Installation

Installation space and environment

External dimensions (mm)



unit (mm)

	3-slot type	5-slot type	8-slot type	10-slot type	12-slot type
A (mm)	205.0	276.0	381.0	452.0	522.0
B (mm)	183.8	254.2	359.8	430.2	500.6

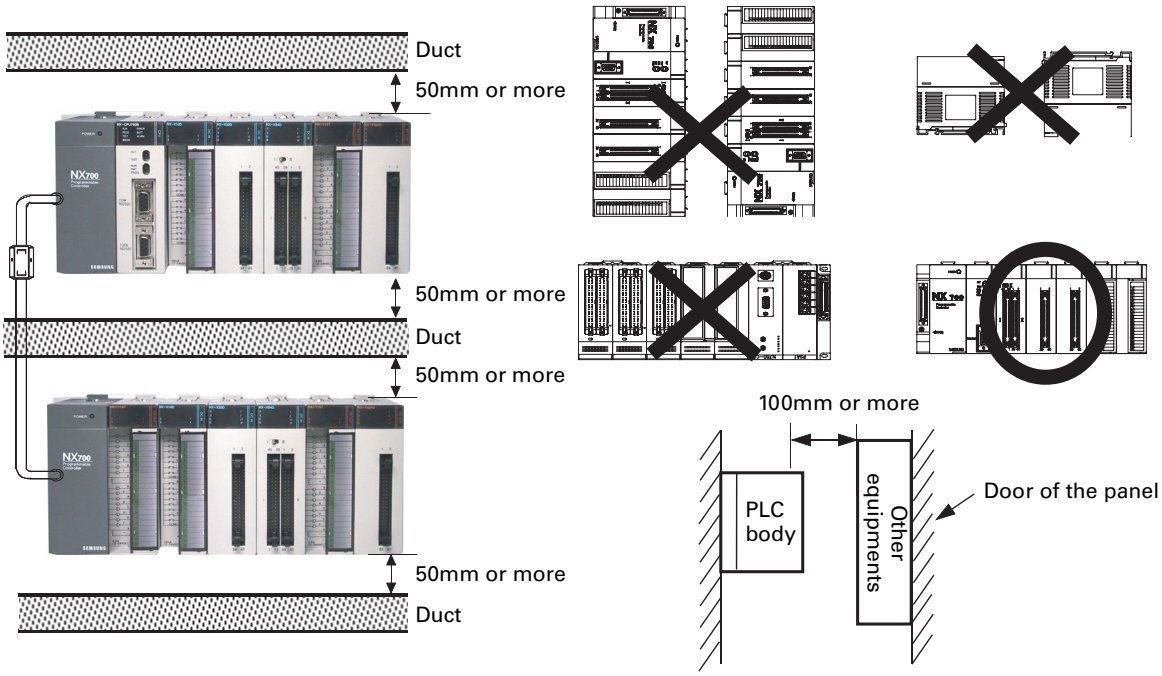
Installation location

Be sure to maintain a sufficient distance from wiring ducts, and other machines below and above the module for proper ventilation.

Do not install the modules stacked up or horizontally. Doing so will prevent proper cooling of the module and cause overheating inside the PLC (programmable controller).

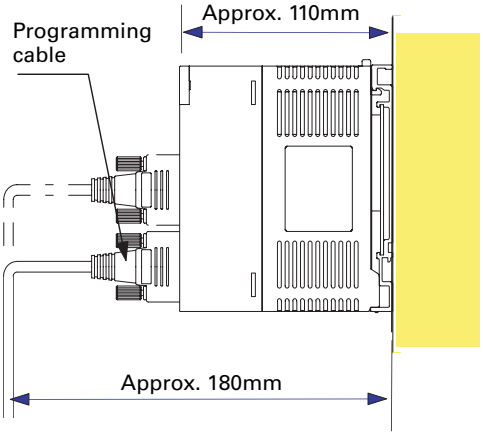
Do not install the module above devices which generate heat such as heaters, transistors or large scale resistors.

In order to eliminate any effects from noise emission, power wires and electromagnetic devices should be kept at least 100 mm away from the surfaces of the module. When installing the module behind the doors of the operation panel, be especially careful to maintain these distances.



Space for programming tool connection

Leave a space of at least 180 mm from the mounting surface for programming tool connections and wiring.

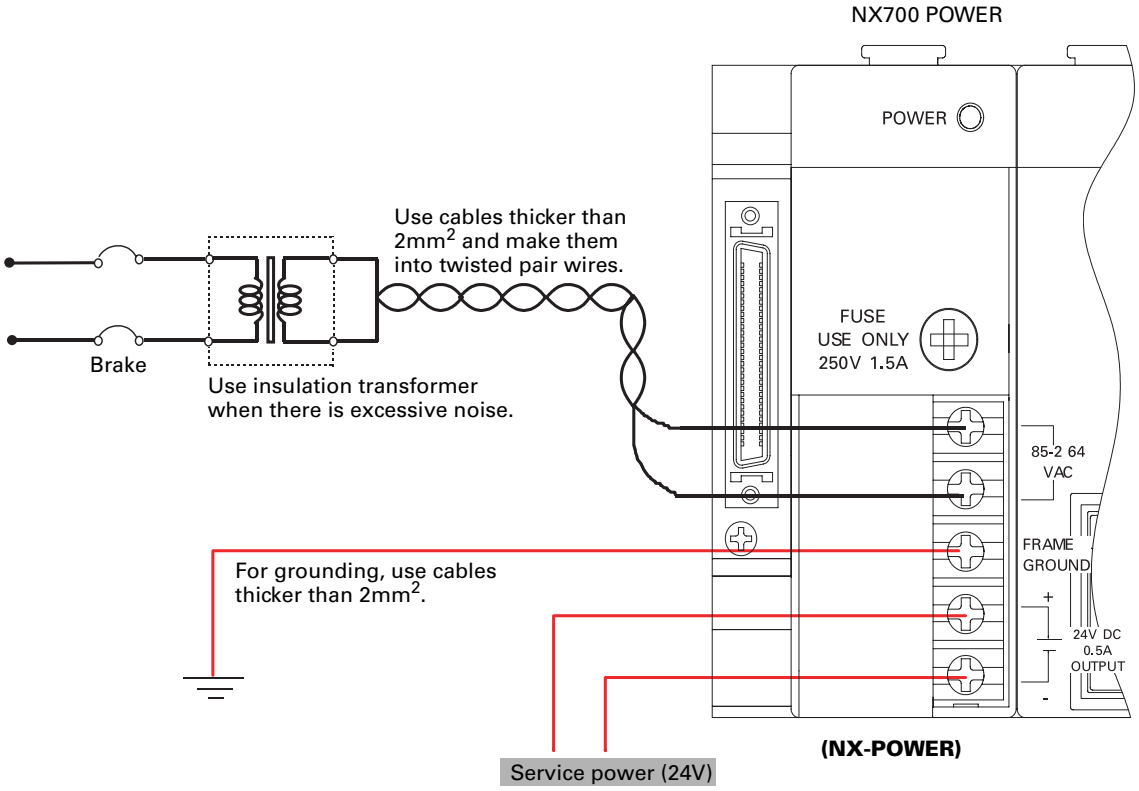


Avoid installing the module in the following conditions

- Ambient temperature outside the range of 0 to 55 °C
- Ambient humidity outside the range of 30 to 85% RH
- Sudden temperature changes causing condensation
- Inflammable or corrosive gases
- Excessive airborne dust, metal particles, salinity
- Benzene, thinner, alcohol, other organic solvents or strong alkaline solutions such as ammonia or caustic soda
- Excessive vibration or shock
- Direct sunlight
- Location near high-tension wires, high-voltage devices, power cables, power devices, or other devices with generate large power surges or electronic fields when starting and stopping (esp. if within 100 mm)

Power Supply Module Wiring

Power supply module wiring



ATTENTION

Supplying 24V to I/O units. However, avoid parallel connection with another 24V source or other different

The output voltage of a power supply should be within the allowed voltage

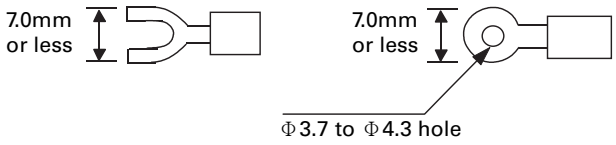
Product name	Rated input voltage	Allowed voltage range
NX-POWER	110–220V AC	85-264V AC

Power supply unit terminal screw

➡ Use M3.5 screw for terminal.

➡ It is recommended that you use crimped terminal for wiring.

➡ Open type terminal ➡ Circular terminal



Use 2mm² twisted pair cable or larger

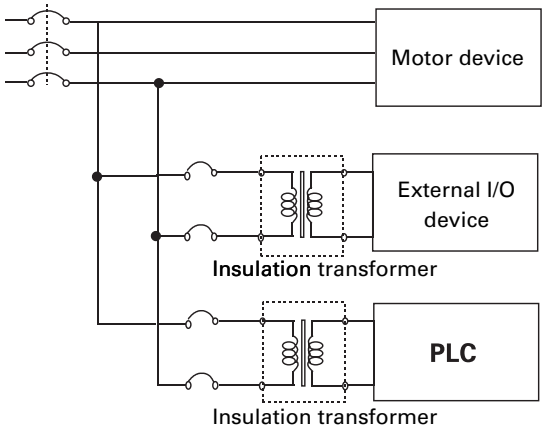
- Use power supply wire that is thicker than 2mm^2 to minimize voltage drops.
- Use twisted pair cable to minimize noise effects.

Proper compressed connection terminal

- Circular type terminal (O type Lug)
- Circular type terminal with insulation resistance
- Open type terminal (Y type Lug)

Power supply system

Use separate wiring systems for the PLC module, I/O module, and power supply module as shown in the following diagram.



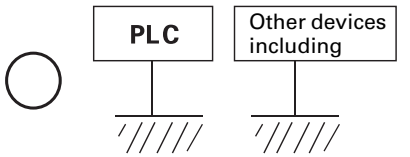
Use isolation transformer for noisy environments

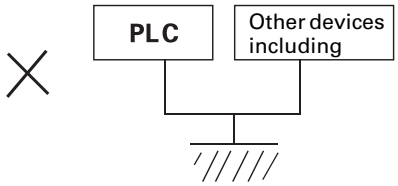
- Use a low noise power supply.
- Use an isolation transformer to reduce the noise as illustrated above.

Grounding

Ground the PLC for noisy environments

- Connected to the metal part of backplane, the frame ground terminal is connected to a solid earth ground.
- Use ground wires with a minimum of $2mm^2$ and the triple grounding connection which has a resistance of less than 100 Ω.
- The point of grounding should be as close to the PLC as possible and the ground wire should be as short as possible.
- If two devices share a single ground point, it may produce an adverse effect. Always use an exclusive ground for each device.





Input and Output Wiring

Input wiring

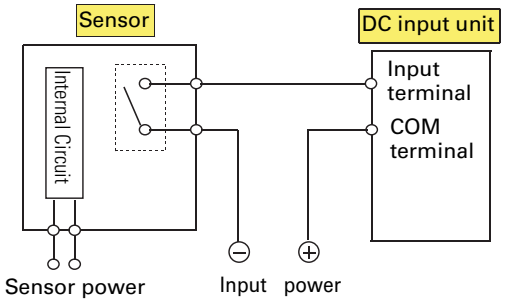
Checkpoints for input module wiring

There can be limits on the number of points that can be simultaneously turned on, based on the module type.
Check such limits in the specifications of each input module. In particular, be careful when using in high ambient temperature.

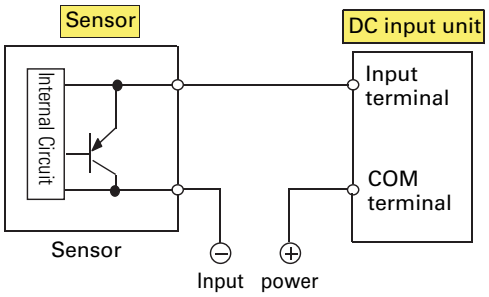
The connection method for each type of input device including sensor is described below.

Photoelectric sensors and proximity sensors

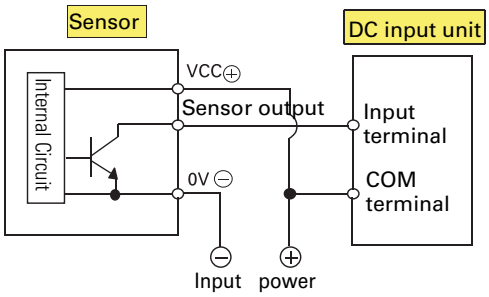
Relay output type



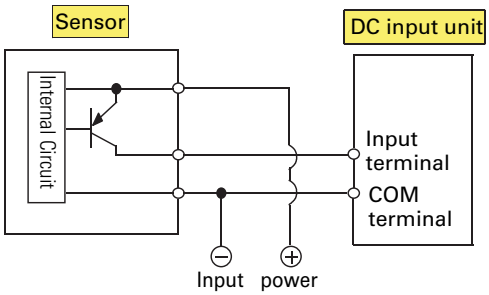
2 wired type



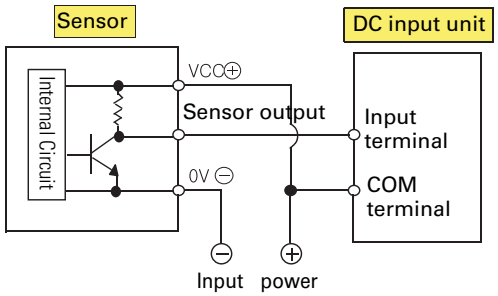
NPN open collector output type



PNP open collector output type

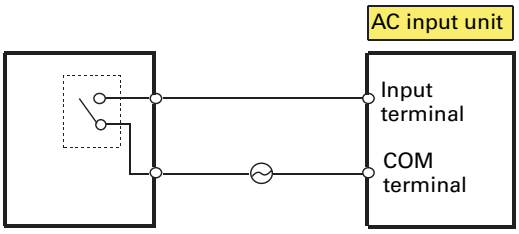


Voltage output type

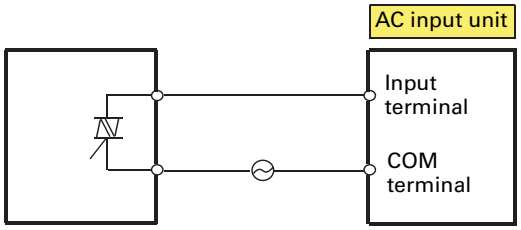


Connection example with AC input device

Contact output type

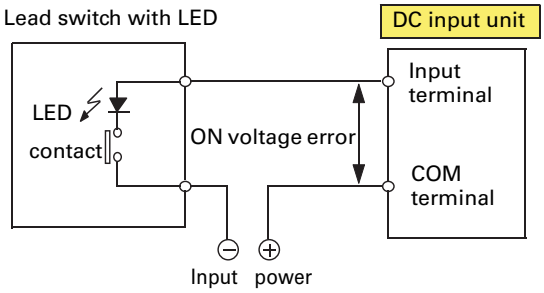


Non-contact output type



Cautions when using lead switch with LED

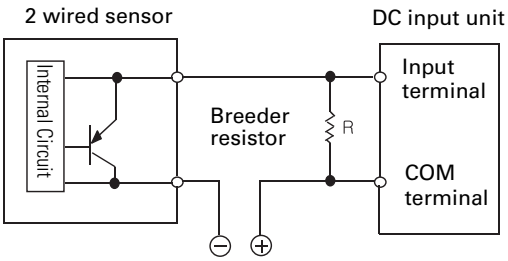
Even when using input contact with embedded serial LED such as lead switch with LED, make sure to supply voltage higher than ON voltage to the PLC input terminal. Pay extra attention, in particular, when connecting multiple switches in serial.



Cautions when using 2 wired type

Use a breeder resistor as below when the leakage current from 2 wired type photoelectric switch or proximity switch keeps flowing into the PLC.

DC 12-24V type input unit
(OFF voltage 2.5V, input impedance 3 KΩ)



I: Leakage current (mA)
R: Breeder resistance (KΩ)

Since the OFF voltage of input is 2.5V, determine the R value to set the voltage between COM terminal and input terminal to be 2.5V or less.
Input impedance is 3 KΩ

$$I \times \frac{3R}{3+R} \leq 2.5, \quad R \leq \frac{7.5}{3I-2.5} \ (K\Omega)$$

For the resistor, W can be calculated as follows:

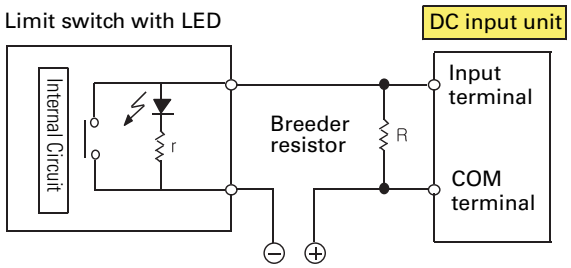
$$W = \frac{(\text{Power supply voltage})^2}{R}$$

Use 3 to 5 times larger value.

Cautions when using limit switch with LED

Use a breeder resistor as below when the leakage current from limit switch with LED keeps flowing into the PLC or LED is accidentally turned on.

DC 12-24V type input unit
(OFF voltage 2.5V, input impedance 3 KΩ)



r: Internal resistance of limit switch (KΩ)
R: Breeder resistance (KΩ)

Since the OFF voltage of the input is 2.5V, set R to get the following current,

$$I = \frac{24 - 2.5}{R} \text{ or more}$$

when the power supply voltage is 24V. Once calculating I, follow the same steps as those for 2 wired sensors described on the previous page.

$$R \leq \frac{7.5}{3I - 2.5} \ (K\Omega) \qquad W = \frac{(\text{Power supply voltage})^2}{R} \times (3 \text{ to } 5 \text{ times})$$

Output wiring

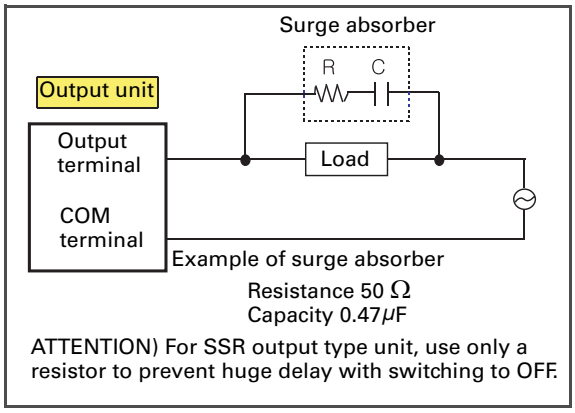
Checkpoints for output unit wiring

- There can be limits on the number of points that can be simultaneously turned on or load current, based on the unit type. Refer to the specification of each unit. Pay extra attention, in particular, when ambient temperature is high.
- Connect protection circuit to inductive load or capacity load, etc. as below.
- There can be current limitation per Common for output units, so make sure not to exceed the allowed range.

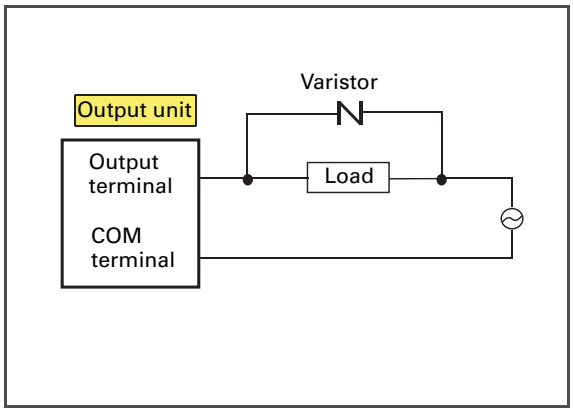
Protection circuit for inductive load

- Connect protection circuit in parallel with the inductive load.
- In particular, when switching the DC inductive load with relay output type, the protection circuit dramatically affects the lifetime of a system. Make sure to connect diodes at both sides of the load.

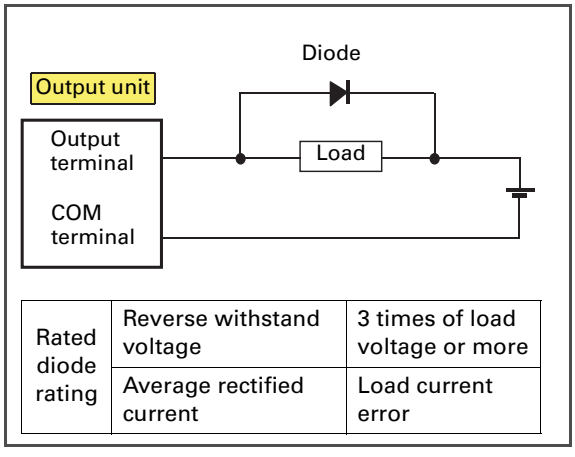
AC load (Relay output type)



ATTENTION) For SSR output type unit, use only a resistor to prevent huge delay with switching to OFF.



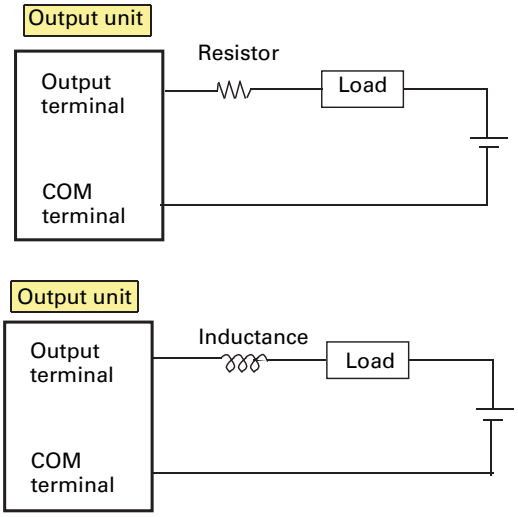
DC load



Rated diode rating	Reverse withstand voltage	3 times of load voltage or more
	Average rectified current	Load current error

Cautions when using capacity load

Use protection circuit as below to minimize the effect of a load with high inrush current.

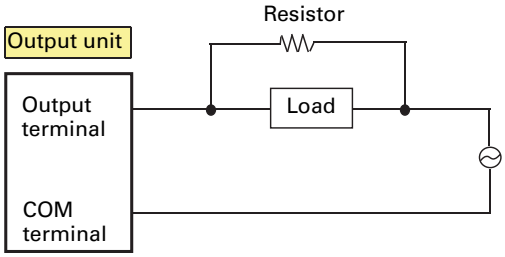


Use an external fuse to protect against overload

- Units embedded with fuse can prevent damages in case of output shortage.
- Since the unit with embedded fuse cannot protect all the components against overload, it is recommended that you use an external fuse for every point.
- The components of output unit may not be protected in case of shortage.

Cautions for leakage current

A low-current load may not be OFF due to leakage current in the SSR output unit. In such a case, connect a resistor in parallel with the load.



Common cautions for I/O module

Input/output/motor cables should be separated from each other.

Separate input/output wirings from power wirings as far as possible. Do not let them pass the same duct and do not bind them together.

Separate I/O and motor cables from high-voltage cables at least 100mm.

Terminal Block Type Module Wiring

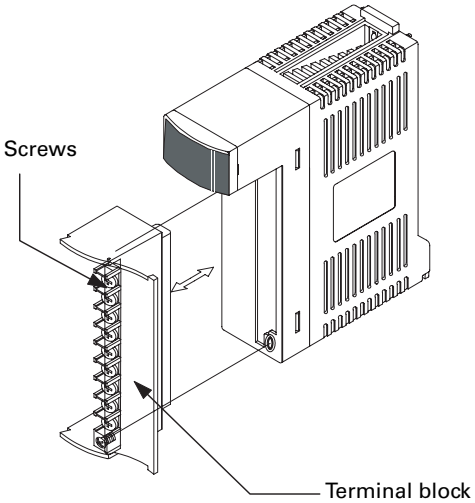
Compressed terminal, M3.0

The terminal base for the NX70 PLC I/O modules (Terminal Type) uses M3.0 terminal screw. Use the following compressed terminals for terminal wiring.

Open terminal Circular terminal

Removable terminal block

- Terminal block of this type of I/O unit can be separated from the unit with wires connected, by unfastening the screws on both ends.
- Make sure to refasten the screws after wiring.



Connector Type Module Wiring

Wiring

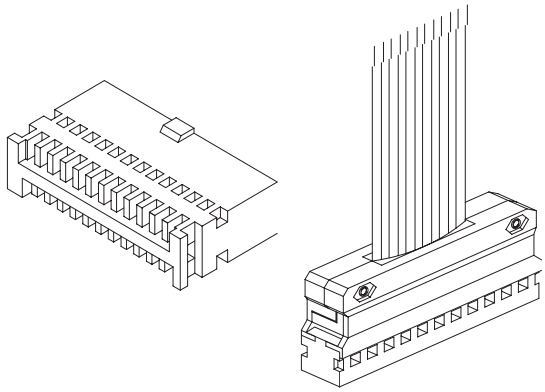
Wiring method

Both NX700 32-point I/O module (NX-X32D, NX-Y32RV, NX-Y32T) and 64-point I/O module (NX-X64D, NX-Y64T) use 40-pin MIL type connectors. To connect to external devices,

1. Insert each pin into socket
2. or, use harness with flat cables. (Available on the market)

Connecting PIN type

PIN type connection. Insert each PIN into a socket. Detail wirings are shown on the next page.

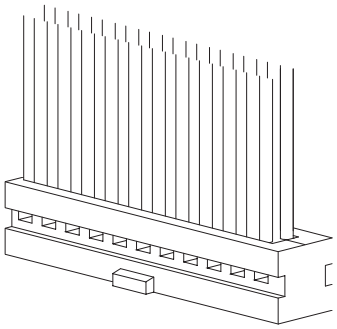


Product name	Catalog number	Product specifications
I/O connector ASS'Y (Pin type)	NX-PIN40 (CPL8890)	40 pins • 40 Pins • Connector hood • Sockets are included.

Applied products	
32 points	NX-X32D (DC IN 32 points) NX-Y32T (TR OUT 32 points) NX-Y32RV (Relay OUT 32 points)
64 points	NX-X64D (DC IN 64 points) NX-Y64T (TR OUT 64 points)

Connecting with harness (Using flat cable connector)

40-pin flat cable connector is used, and 20 crimped terminals are connected at the end. Its total length is 1.5m. Direct connection to the unit.



Product name	Catalog number	Product specifications
Flat cable ASS'Y	NX-CBLDC	DC IN 32 points, 64 points Connectors harness cable 1.5m
	NX-CBLTR	TR OUT 32 points, 64 points Connectors harness cable 1.5m
	NX-CBLRY	Relay OUT 32 points Connectors harness cable 1.5m

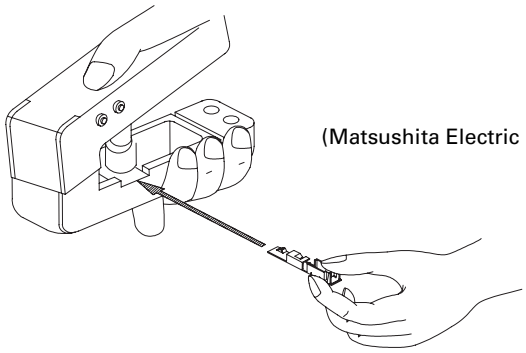
Applied products	
32 points	NX-X32D (DC IN 32 points) NX-Y32T (TR OUT 32 points) NX-Y32RV (Relay OUT 32 points)
64 points	NX-X64D (DC IN 64 points) NX-Y64T (TR OUT 64 points)

Pressure welding socket for PIN type connection

Direct pressure welding with clothings on saves wiring time and efforts.

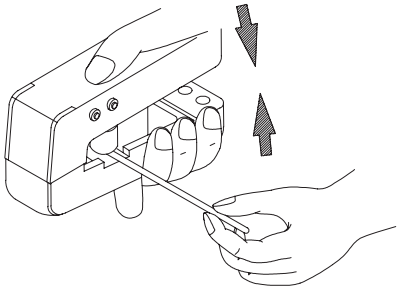
Wiring

1. Bend the contact part at a carrier, and set it into the pressure welding device.

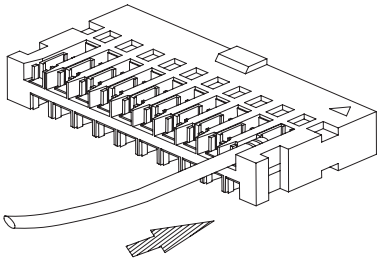


(Matsushita Electric Industrial Co.: AXY52000)

2. Insert both cables with case on until touching each other, and hold the pressure welder slightly.

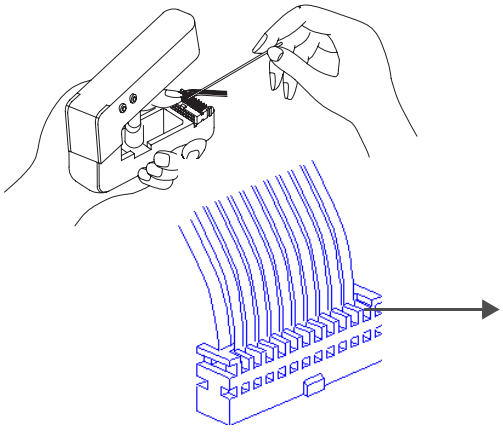


3. Insert the cables into the housing once welding is done.



Contact removal pin for wiring failure.

In case of wiring failure or cable pressure welding failure, use the contact removal pin with the welder to remove the contact

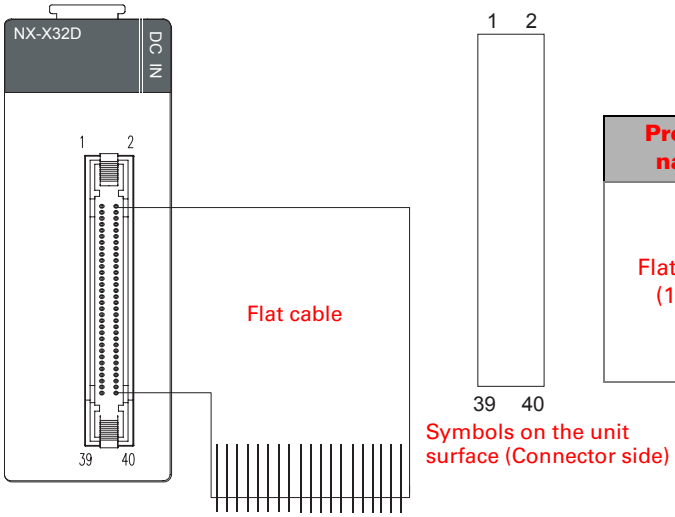


Wiring flat cable connector

Cautions when using flat cable connector

When a flat cable connector is used for direct connection, the mapping between cable No. and I/O No. is as follows:

Connecting 32-point unit (NX-X32D, NX-Y32T, NX-Y32RV)



Product name	Catalog number	Specifications
Flat cable (1.5m)	NX-CBLDC	DC IN 32 points (NX-X32D)
	NX-CBLTR	TR OUT 32 points (NX-Y32T)
	NX-CBLRY	Relay OUT 32 points (NX-32RV)

Mapping table between flat cable No. and I/O (32 points)

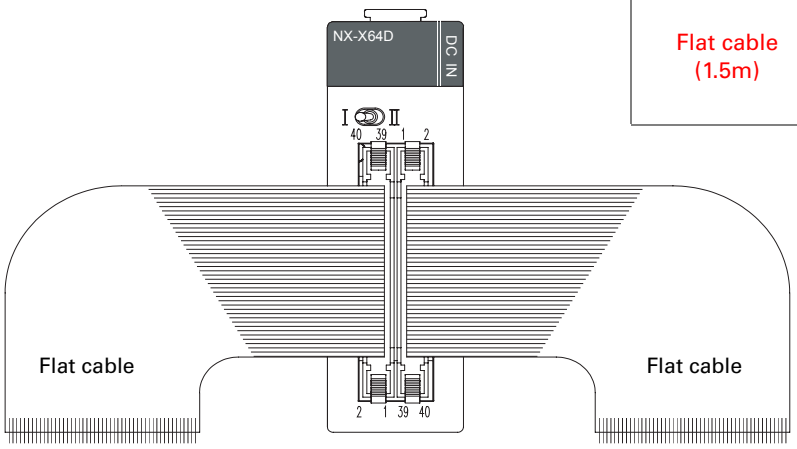
Symbols on the unit surface	NX-X32D	NX-Y32T	NX-Y32RV
1	X 0	Y 0	Y 0
2	X 1	Y 1	Y 1
3	X 2	Y 2	Y 2
4	X 3	Y 3	Y 3
5	X 4	Y 4	Y 4
6	X 5	Y 5	Y 5
7	X 6	Y 6	Y 6
8	X 7	Y 7	Y 7
9	X 8	Y 8	Y 8
10	X 9	Y 9	Y 9
11	X A	Y A	Y A
12	X B	Y B	Y B
13	X C	Y C	Y C
14	X D	Y D	Y D
15	X E	Y E	Y E
16	X F	Y F	Y F
17	X 10	Y 10	Y 10
18	X 11	Y 11	Y 11
19	X 12	Y 12	Y 12
20	X 13	Y 13	Y 13

Symbols on the unit surface	NX-X32D	NX-Y32T	NX-Y32RV
21	X 14	Y 14	Y 14
22	X 15	Y 15	Y 15
23	X 16	Y 16	Y 16
24	X 17	Y 17	Y 17
25	X 18	Y 18	Y 18
26	X 19	Y 19	Y 19
27	X 1A	Y 1A	Y 1A
28	X 1B	Y 1B	Y 1B
29	X 1C	Y 1C	Y 1C
30	X 1D	Y 1D	Y 1D
31	X 1E	Y 1E	Y 1E
32	X 1F	Y 1F	Y 1F
33	NC	NC	COM
34	NC	NC	COM
35	NC	NC	COM
36	NC	NC	COM
37	COM	V DC+	24V DC+
38	COM	V DC+	24V DC+
39	COM	V DC−	24V DC−
40	COM	V DC−	24V DC−

NC: No Connection

Connecting 64-point unit (NX-X64D, NX-Y64T)

Product name	Catalog number	Specifications
Flat cable (1.5m)	NX-CBLDC	DC IN 64 points (NX-X64D)
	NX-CBLTR	TR OUT 64 points (NX-Y64T)



Mapping table between flat cable No. and I/O (64 points)

Symbols on the unit surface	NX-X64D	NX-Y64T
1	X 0	Y 0
2	X 1	Y 1
3	X 2	Y 2
4	X 3	Y 3
5	X 4	Y 4
6	X 5	Y 5
7	X 6	Y 6
8	X 7	Y 7
9	X 8	Y 8
10	X 9	Y 9
11	X A	Y A
12	X B	Y B
13	X C	Y C
14	X D	Y D
15	X E	Y E
16	X F	Y F
17	X 10	Y 10
18	X 11	Y 11
19	X 12	Y 12
20	X 13	Y 13

Symbols on the unit surface	NX-X64D	NX-Y64T
21	X 14	Y 14
22	X 15	Y 15
23	X 16	Y 16
24	X 17	Y 17
25	X 18	Y 18
26	X 19	Y 19
27	X 1A	Y 1A
28	X 1B	Y 1B
29	X 1C	Y 1C
30	X 1D	Y 1D
31	X 1E	Y 1E
32	X 1F	Y 1F
33	NC	NC
34	NC	NC
35	NC	NC
36	NC	NC
37	COM	V DC+
38	COM	V DC+
39	COM	V DC−
40	COM	V DC−

NC: No Connection

ATTENTION

Be careful that Connector (I) and connector (II) are opposite.

Safety Measures

Precautions regarding system design

In certain applications, malfunction may occur for the following reasons:

- The timing difference between opening and closing of the PLC power supply, the I/O modules and power equipment
- An operation time lag when a momentary power failure occurs
- Abnormality in the PLC, external power supply, or other devices

In order to prevent a malfunction resulting in system shutdown choose the adequate safety measures listed in the following:

Interlock circuits on the outside of PLC

When a motor clockwise/counter-clockwise operation is controlled, provide an interlock circuit that prevents clockwise and counter-clockwise signals from inputting into the motor at the same time.

Emergency stop circuits on the outside of PLC

Install the emergency stop circuits outside the PLC to stop the power supply of the output device.

Start PLC after other devices (Start up sequence)

The PLC should be operated after all of the I/O devices and the power equipments are energized.

- Switch to RUN mode after the start of PLC.
- Use a timer circuit to delay the start of PLC.

Watchdog timer

The watchdog timer is a program error and hardware error detection timer. It goes On when the scan time exceeds 640 ms.

When the watchdog timer is activated, at the same time the ALARM LED lights, the ALARM contacts on the power supply module turn to On, all output modules are turned Off and the module is put in halted state. (The system is in a non-processing state that includes communications with programming tools as well.)

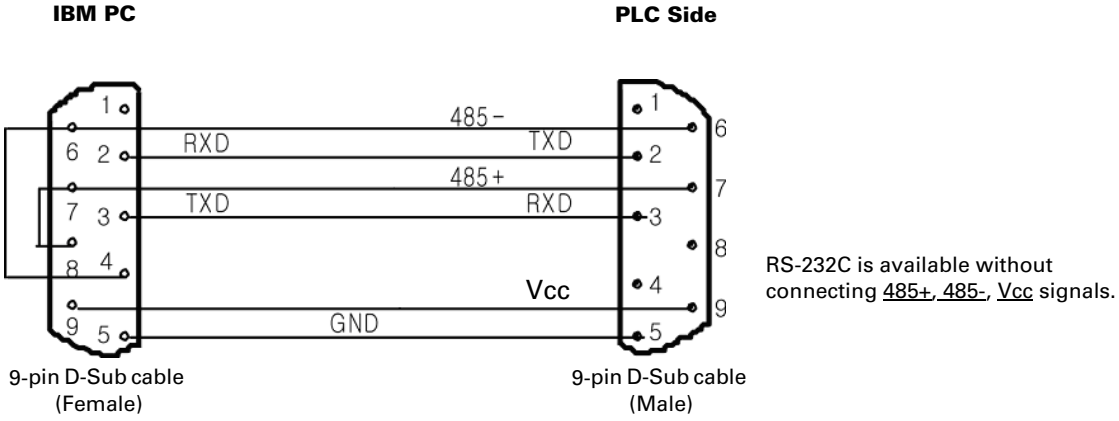
NX-CPU700p Processor Module
Communications Specifications

Connection specification	RS485	RS232C	Remarks
Transfer distance (Max)	1.2 Km	15 m	
Transmission speed	38,400, 19,200, 9,600, 4,800 bps		DIP switch setting
Protocol	Half duplex asynchronous polling		
Parity	No parity		
Stop bit	1 stop bit		
Cable type	Twisted pair cable		Use shield cable

RS232C/RS485 cable wiring diagram: NX-CBLCPU02 (2m), NX-CBLCPU05 (5m)

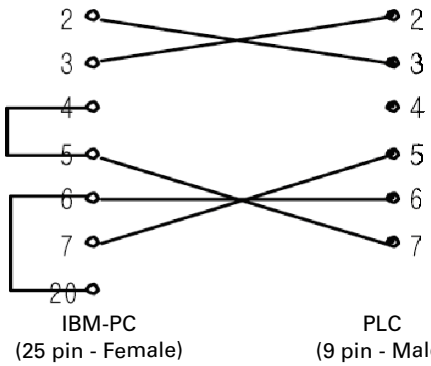
NX-CBLCPU02 (2m)

NX-CBLCPU05 (5m)



RS-232C is available without connecting 485+, 485-, Vcc signals.

Reference (25 pin to 9 pin) wiring diagram



EEPROM Backup

What's EEPROM backup?

EEPROM (Electric Erase Programmable Read Only Memory) can retain the data when the power is turned off, and erase or record data when the power is turned on. This function allows you to retain the PLC program when the power is turned off. And it also erases an existing program and records a new one when correcting or storing a program after turning on the power.

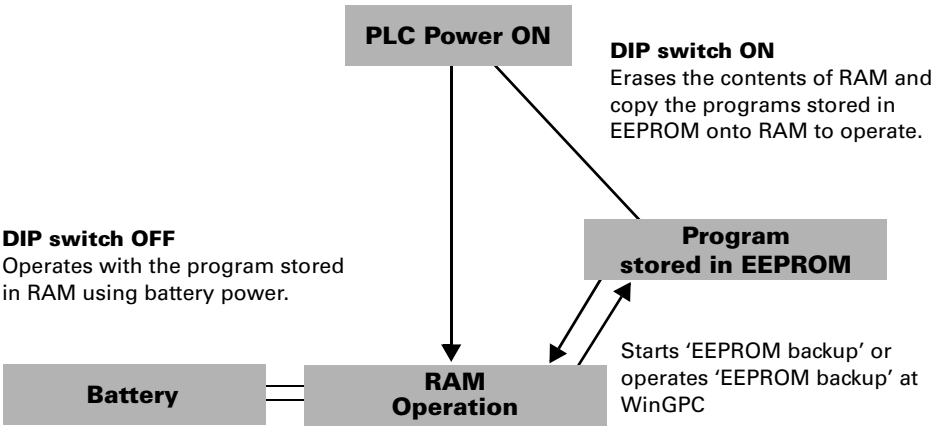
Applicable models

The types of EEPROM can be defined by its characteristics.
The CPUs for OE MAX NX70 (NX70-CPU70p1 and NX70-CPU70p2) PLC use the flash memory.

This function is widely used since it is easy to use, store, and transfer data, and is built in the NX70 PLC (NX70-CPU70p2).

EEPROM uses 29EE512 and has a minimum of 3000 times to write to flash memory. It should be noted that the memory is to be changed when exceeding the use of over 3,000 times.

Procedure



Backup using WinGPC

How to backup using WinGPC

Connect Online.

Save the completed program onto the PLC (Download: WinGPC => PLC).

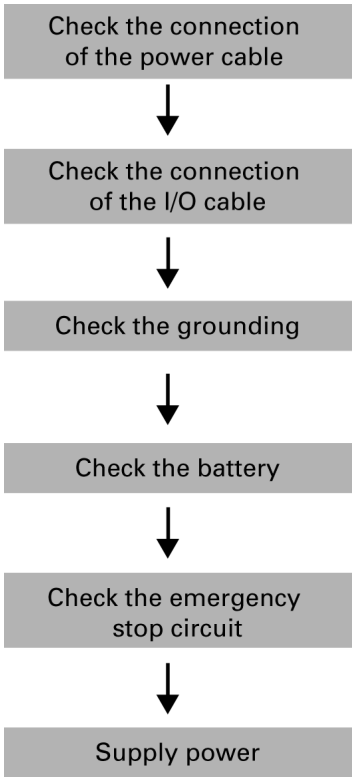
Select the 'EEPROM Backup (E)' in the 'Online' menu.

5

Test Run and Troubleshooting

Test Run Precautions

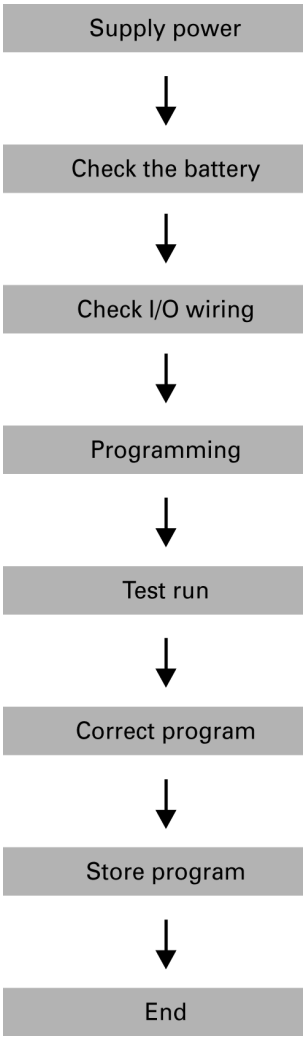
Before installing the I/O wiring of the PLC and supplying power, check the following items.



Item	What to check
The connection of the power cable and the I/O cables	• Check if the wiring is secured. • Check if the terminal screws are tightened. • Check if the parts of connectors are properly joined. • Check if the I/O modules are firmly fixed. • Check if the power cable is securely connected. • Check if the cable size is correct.
Grounding	• Check if the grounding is triple grounded and separate from other device grounds.
Battery	• Check if the battery is installed into holder on the CPU module. • Check if the battery connector is connected to the CPU board
Emergency stop circuit	• Check if the emergency stop circuit for problems external to the PLC is wired accurately, and will immediately disconnect power on demand.
Power source	• Check if the power and voltage sources are within specifications. - For 110V ac (90 to 132V ac) - For 220V ac (180 to 264V ac) • Check if the power to the AC input module is within specifications.

Test Run Procedure

When the PLC has been installed and wired, begin test run in the following order.

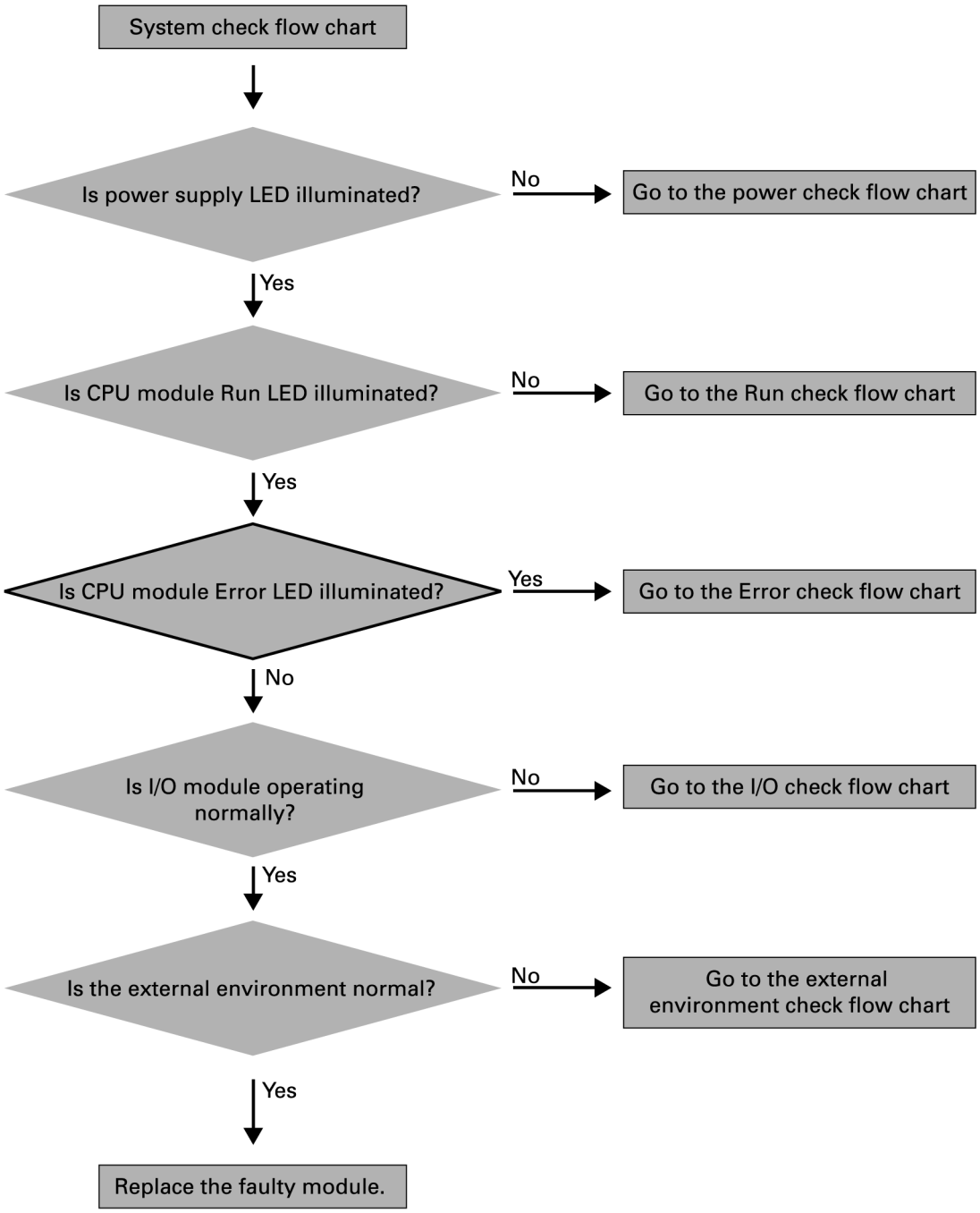


Item	What to check/do
Supply power	• Check if the input voltage of the power supply module is within specifications. • Check if the power voltage for the I/O modules is within specifications. • Connect WinGPC to the CPU module. (Set the CPU module to the PROG mode.) • Turn on the power source. • Check the LED display of the power supply module.
Initialize memory	• Initialize the PLC using WinGPC. (This clears the program on the PLC.)
Check I/O wiring	• Check the LED of the input modules and use the monitor function of WinGPC or HHP after test run the input device. • Check the wiring of the output by turning the output On/Off using the monitor mode of WinGPC. (set CPU module to Run mode.)
Programming	• Input the program instructions using WinGPC. • Download the program from WinGPC into the CPU module, if any.
Test run	• Set the mode switch of the CPU module to run. • Check if the Run LED is illuminated. • Check the sequence operation.
Correct program	• Check and correct any program errors.
Store program	• Store the program onto a floppy disk or similar storage device such as HDD. • Print the program (ladder, mnemonic) and store it in a secure place.
NOTE It is recommended to record the PLC types, program capacity, name of installation, and date for the recorded program.	

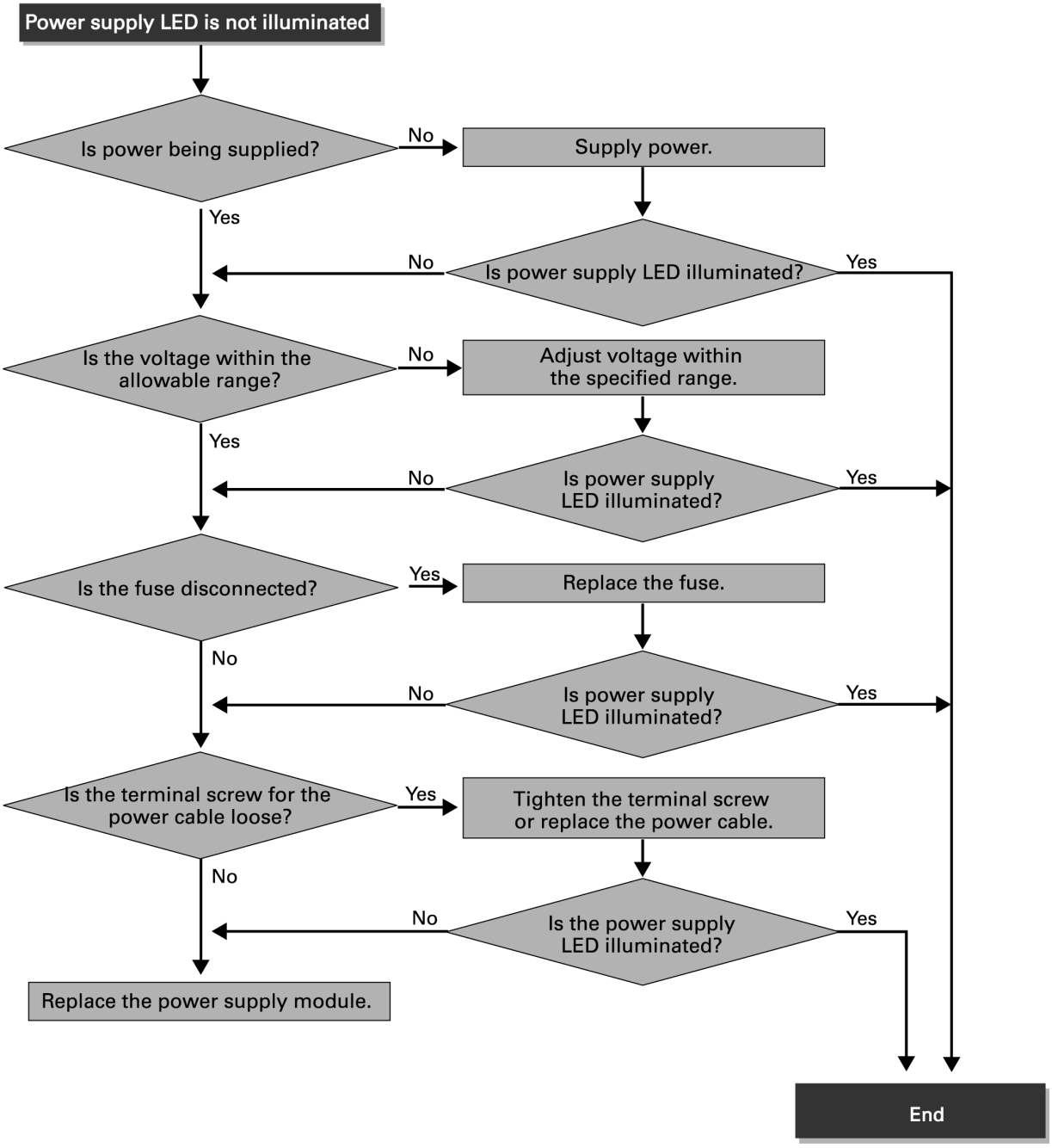
Test Run Flow Charts

System check flow chart

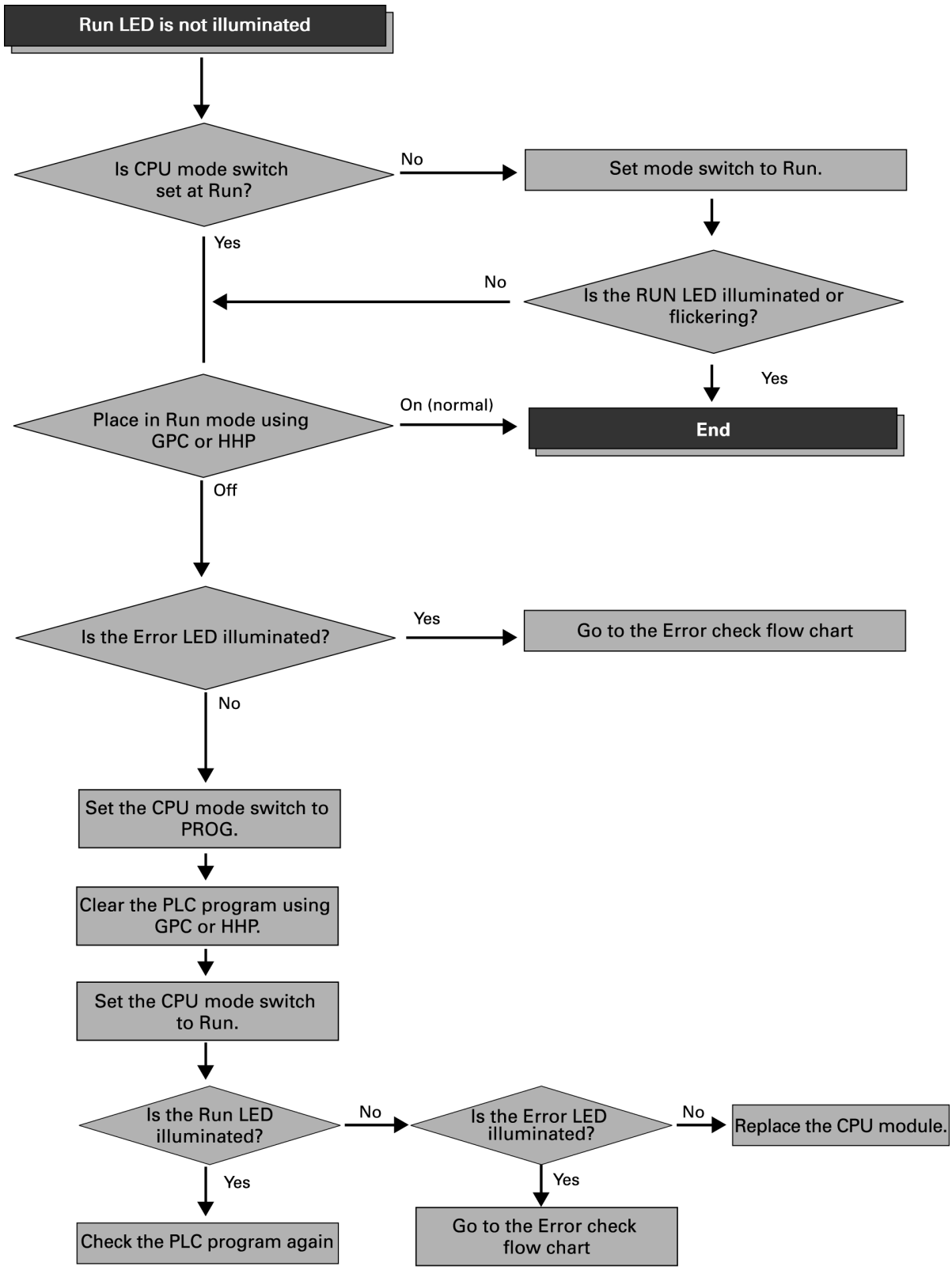
When you encounter problems during startup or test run, first of all, figure out the problems thoroughly. Check if the problems can be reproduced, and analyze the relevance to other devices. Then refer to the system check flow chart.



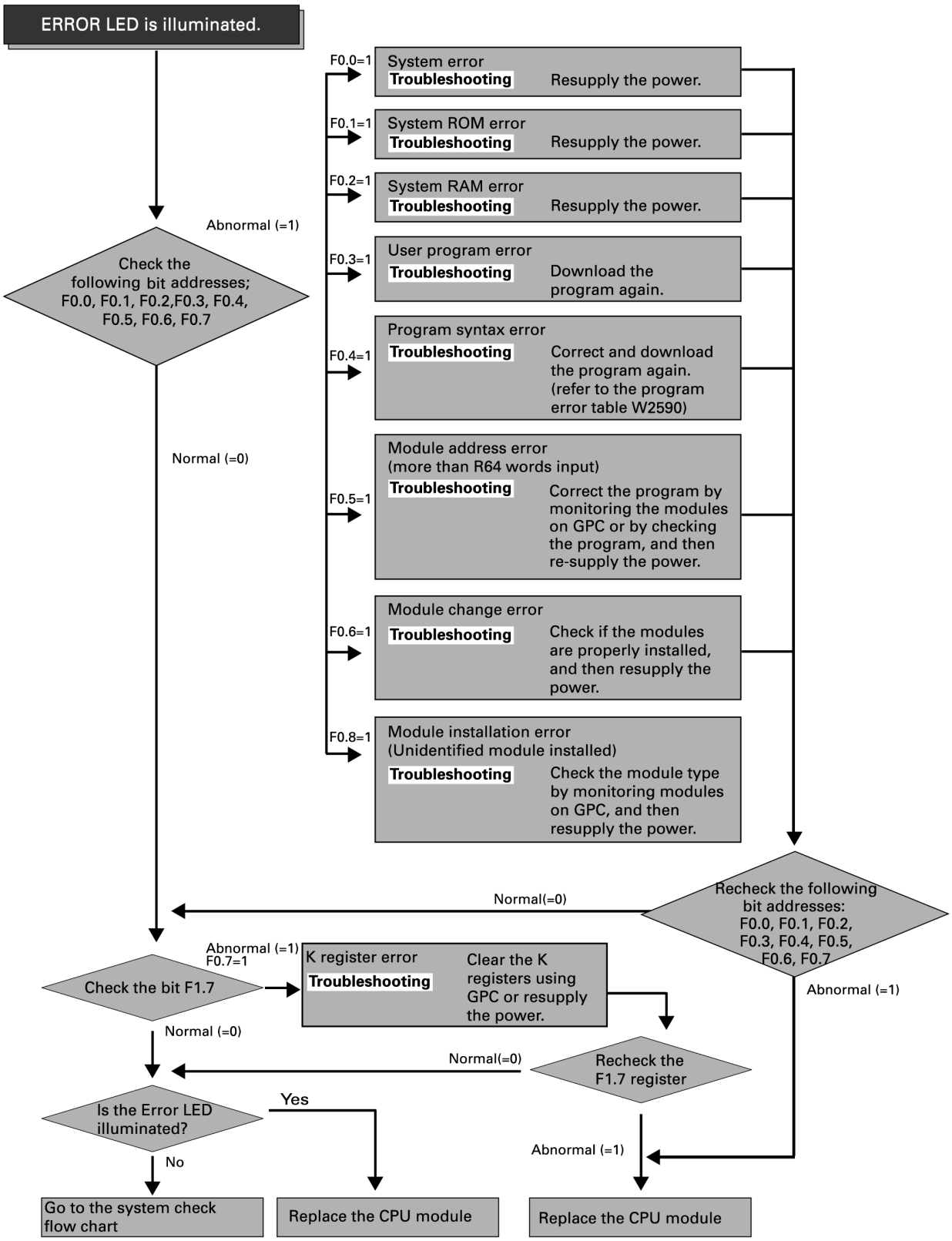
Power check flow chart



Run check flow chart

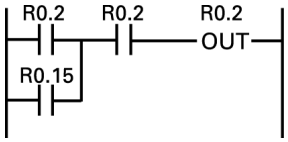


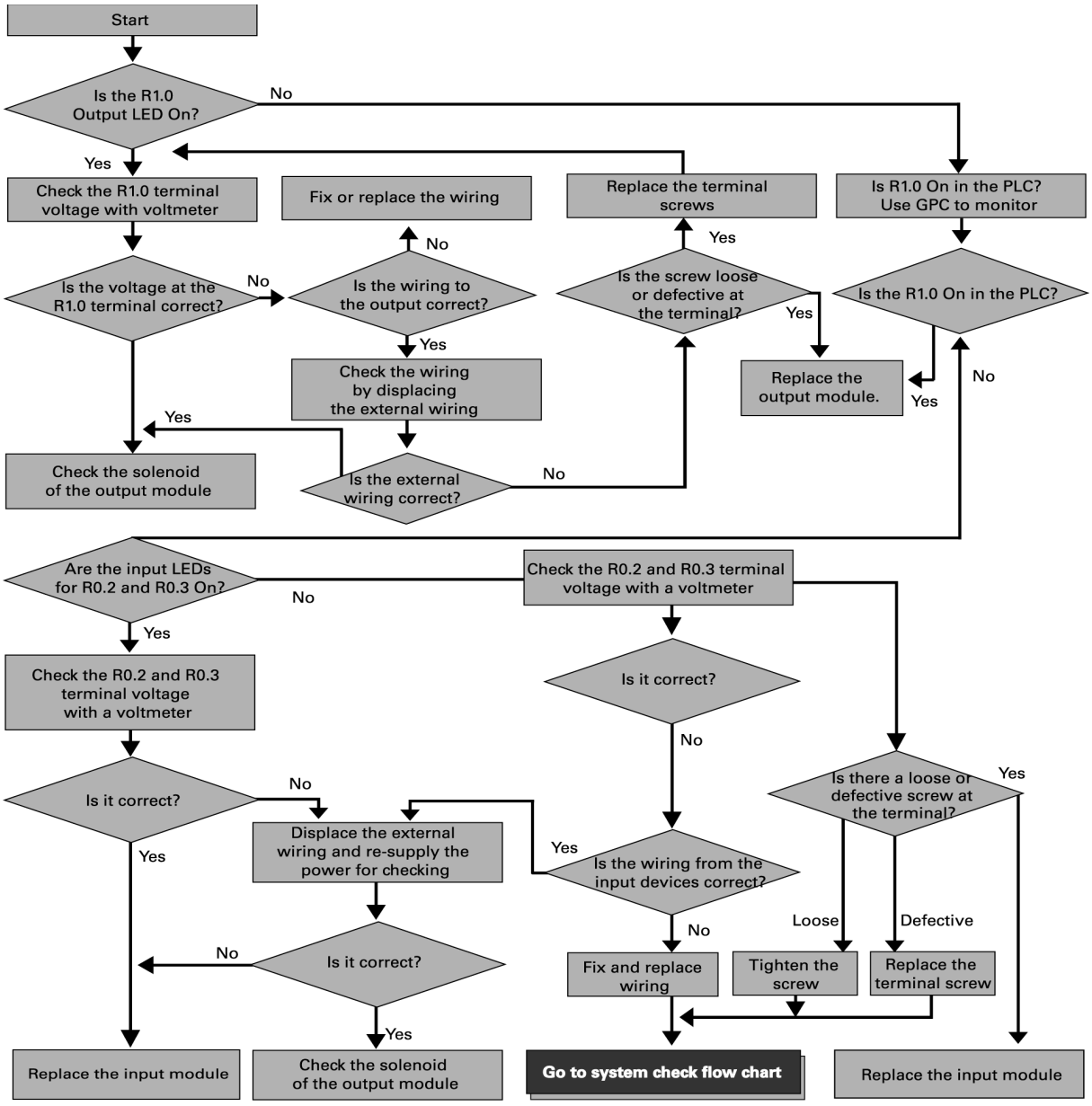
Error check flow chart



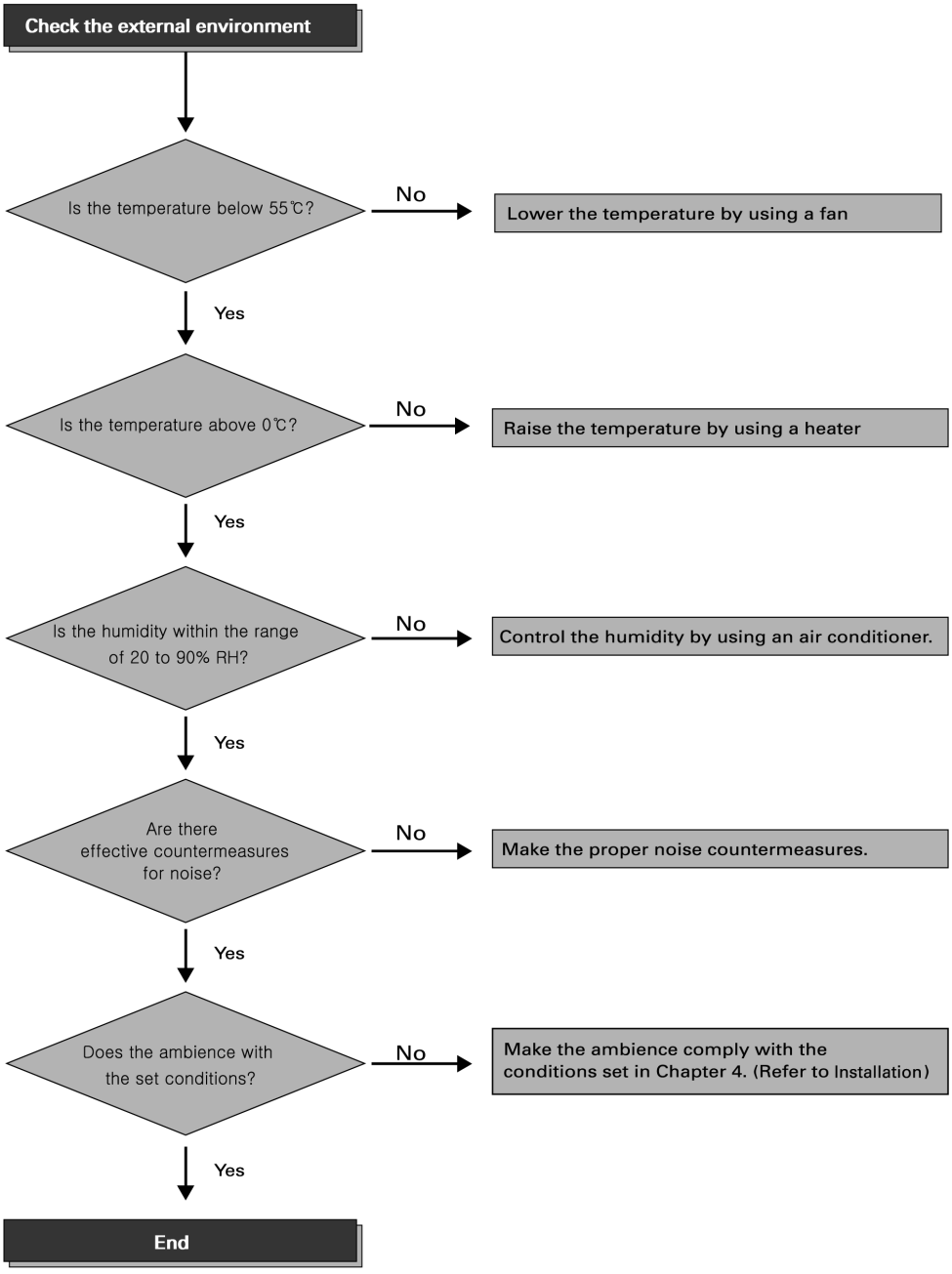
I/O check flow chart

This page presents an example of a troubleshooting procedure based on the right circuit.





External environment check flow chart



Inspection and Maintenance

Inspection and maintenance

Symptom	Possible cause	Action
Power supply LED will not illuminate.	Fuse blows	Replace the fuse
Fuse blows frequently.	Short circuit or defective part	Replace the power supply or the CPU module
Run LED will not illuminate.	Program errors	Correct the program
	Power line defect	Replace the CPU module
Output will not turn to On state during Run.	Short or open circuit	Replace the CPU module
I/O modules above a certain address will not operate.	I/O bus error	Replace the backplane module
Input or output module of only certain address will turn to On state.	I/O bus error	Replace the backplane module
Not all contacts on an output module operate properly.	I/O bus error	Replace the backplane module

Input module

Symptom	Possible cause	Action
No inputs on an input module will turn On (LEDs are not illuminated).	No external input power	Supply power
	Low external input voltage	Supply rated load power
	Loose terminal screw or defective contact	Tighten the screw and reconnect the module
Inputs will not turn to On state (LEDs are illuminated).	Defective input circuit	Replace the input module
Inputs will not turn to Off state.	Defective input circuit	Replace the input module
One or more inputs on an I/O module will not turn On.	Device connected to the input module is defective.	Replace the input device
	Loose input wiring	Reconnect the input wiring
	External input time is too short.	Adjust the input module
	Loose terminal screw or defective contact	Tighten the screw and reconnect the module
One or more inputs on an I/O module will not turn Off.	Defective input circuit	Replace the input module
Input changes On/Off state erratically.	Low external input voltage	Supply rated input voltage
	Noise error	Troubleshoot for noise
	Loose terminal screw or defective contact	Tighten the screw and reconnect the module
Input display LED will not illuminate (input is On in PLC).	LED error	Replace the input module

Output module

Symptom	Possible cause	Action
No outputs on an output module will turn On.	No external input power	Supply power
	Low external input voltage	Supply rated load power
	Loose terminal screw or defective contact	Tighten the screw and reconnect the module
	Defective I/O connector contacts	Replace the output module
	Defective output circuit	Reconnect the output module
No outputs on an output module will turn Off.	Defective output circuit	Replace the output module
One or more inputs on an I/O module will not change to On state (LED is illuminated).	Output time too short	Correct the program
	Defective output circuit	Replace the output module
One or more inputs on an I/O module will not change to On state (LED is not illuminated).	Incorrect output load	Replace the output load
	Short output wiring	Reconnect the output wiring
	Loose terminal screw or defective contact	Tighten screw and reconnect the module
	Defective output contact	Replace the output module or relay
	Defective output circuit	Replace the output module
One or more outputs on an I/O module will not change to On state (LED is illuminated).	Defective output circuit	Replace the output module or relay
	Error caused by leak or residual current	Replace the external load
One or more outputs on an I/O module will not turn Off (LED is not illuminated).	Defective output circuit	Replace the output module
Output changes On/Off state erratically.	Low external input voltage	Supply rated load power
	Noise error	Countermeasure against noise
	Loose terminal screw or defective contact	Tighten the screw and reconnect the module
A set of 8 points on an I/O module operate incorrectly or identically.	Loose common terminal screw	Tighten the screw and reconnect the module
	Defective terminal connector	Tighten the screw and reconnect the module
	CPU module error	Replace the CPU module
Output display LED is not illuminated.	LED error	Replace the output module

Periodic inspection and maintenance items

The NX-CPU700p controller requires periodic inspection and maintenance for proper operation. The following items should be checked every six months, but the period can be shortened according to the operational environment.

Item	Check item	Requirement	Remarks
Supplied power	Does the voltage measured within the power terminal fall within the specified range?	Voltage must be within the power module input voltage specifications.	Voltmeter
Environment	Does the temperature fall within the specified range?	0 to 55 °C	Thermometer
	Does the humidity fall within the specified range?	35 to 85%RH	Hygrometer
	Is there any dust present?	No dust	Visual
I/O power	Does the control voltage supplied to the I/O modules fall within the specified limit?	Control voltage must be within the input and output modules specifications.	Voltmeter
Module mounting and wiring	Are all modules firmly mounted?	All should be firmly secured.	Screwdriver
	Is the connection cable firmly wired?		
	Is the external wiring screw tight?		
Life expectancy of parts	Contact relay	Electric lifetime: 100,000 to 300,000 operations	
	Battery	3 years at 25 °C	

Precautions when troubleshooting

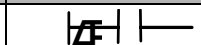
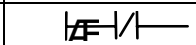
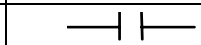
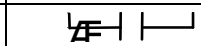
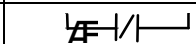
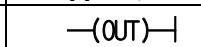
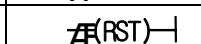
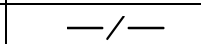
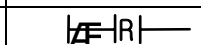
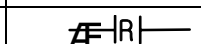
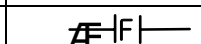
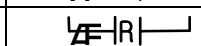
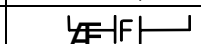
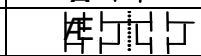
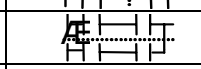
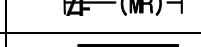
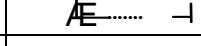
- *Always turn off the power whenever installing or removing modules.*
- *Check the module once more before replacing the defective part.*
- *Return the defective module for repair with any detailed information about its problems.*
- *When a contact is defective, clean the contact with a clean cotton and alcohol and then retest the module.*
- *Do not use thinner to clean since it might cause discoloration on the module's case.*

6

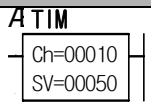
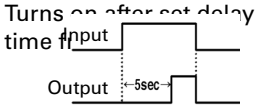
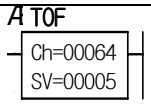
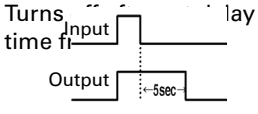
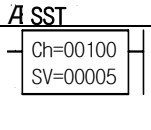
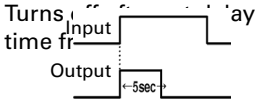
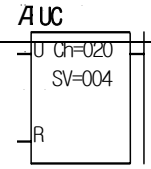
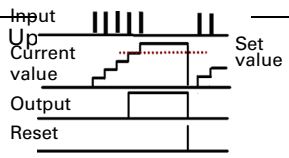
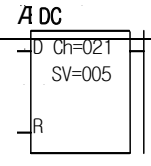
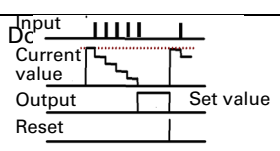
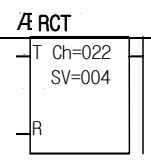
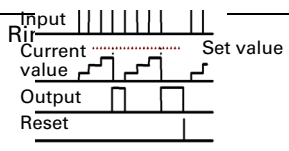
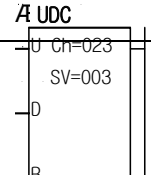
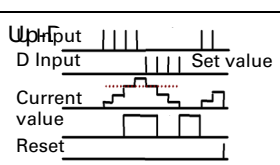
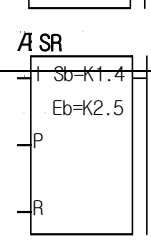
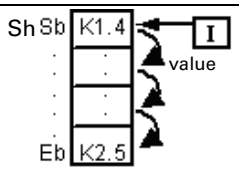
Programming Instructions

IMPORTANT Refer to the NX7/NX70/NX700 Instruction Set Reference Manual for detailed information on the NX7, NX70, and NX700 instruction set and for application examples to show the instruction set in use.

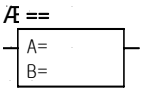
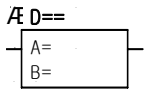
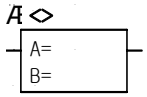
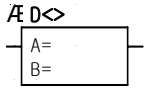
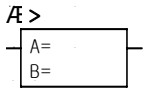
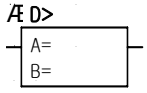
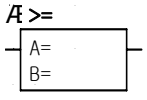
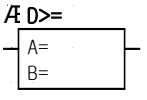
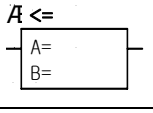
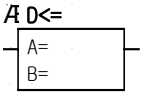
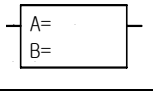
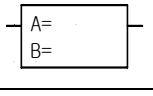
Basic Sequence Instructions

Mnemonic	Name	Ladder Symbol	Description
STR	Start		Starts contact A.
STN	Start Not		Starts contact B.
AND	And		Contact A series circuit
ANN	And Not		Contact B series circuit
OR	Or		Contact A parallel circuit
ORN	Or Not		Contact B parallel circuit
OUT	Out		Arithmetic result output
SET	Set		Sets output and retains On.
RST	Reset		Resets output and retains Off.
NOT	Not		Inverts circuit.
STR DIF	Start Differential		Starts rising edge contact  .
STR DFN	Start Dif. Not		Starts falling edge contact  .
AND DIF	And Dif.		Rising edge series connection 
AND DFN	And Dif. Not		Falling edge series connection 
OR DIF	Or Dif		Rising edge parallel connection 
OR DFN	Or Dif. Not		Falling edge parallel connection 
ANB	And Block		Circuit blocks series connection.
ORB	Or Block		Circuit blocks parallel connection.
MS	Master block Set		Starts master block. (for processor version 2.0 or higher)
MR	Master block Reset		Ends master block. (for processor version 2.0 or higher)
MCS	Master Control Set		Starts circuit branch.
MCR	Master Control Reset		Ends circuit branch.
-	Extension		Extension (Used in pairs with AND condition when extending. Exclusive for WinGPC, GPC5, etc)

Timer, Counter and Shift Register Instructions

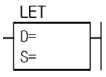
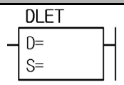
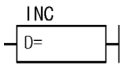
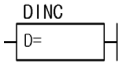
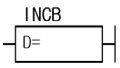
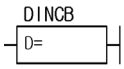
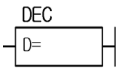
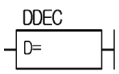
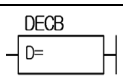
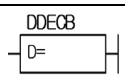
Mnemonic	Name	Ladder Symbol	Description	Remarks
TIM	On Delay Timer			Time base: Ch 0 to 63 = 0.01s Ch 64 to 255 = 0.1s Setting range: SV = 0 to 65535 Contact indicator: TC + channel number
TOF	Off Delay Timer			Time base: Ch 0 to 63 = 0.01s Ch 64 to 255 = 0.1s Setting range: SV = 0 to 65535 Contact indicator: TC + channel number
SST	Single Shot Timer			Time base: Ch 0 to 63 = 0.01s Ch 64 to 255 = 0.1s Setting range: SV = 0 to 65535 Contact indicator: TC + channel number
UC	Up Counter			Channel range: Ch 0 to 255 (Shared with timer) Setting range: SV = 0 to 65535 Contact indicator: TC + channel number
DC	Down Counter			Channel range: Ch 0 to 255 (Shared with timer) Setting range: SV = 0 to 65535 Contact indicator: TC + channel number
RCT	Ring Counter			Channel range: Ch 0 to 255 (Shared with timer) Setting range: SV = 0 to 65535 Contact indicator: TC + channel number
UDC	Up-Down Counter			Channel range: Ch 0 to 255 (Shared with timer) Setting range: SV = 0 to 65535 Contact indicator: TC + channel number
SR	Shift Register			Usable address areas for Sb and Eb : M, K 1 bit shift on each p input. Stores the status value I in Sb for every P input. Max. number of instructions: 256

Comparison Instructions

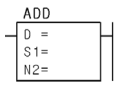
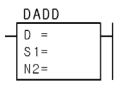
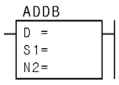
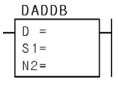
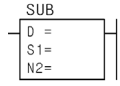
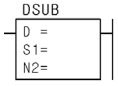
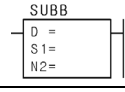
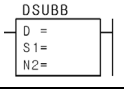
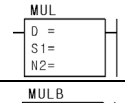
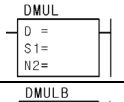
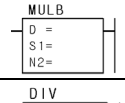
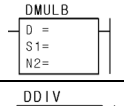
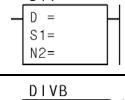
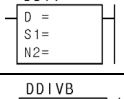
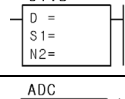
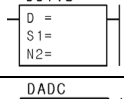
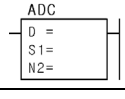
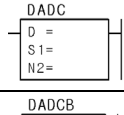
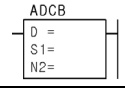
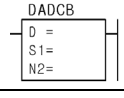
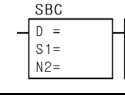
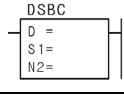
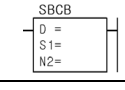
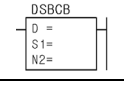
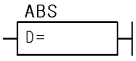
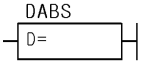
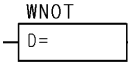
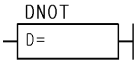
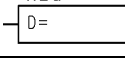
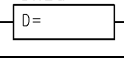
Mnemonic	Name	Word ladder symbol	Double word ladder symbol	Description
STR == AND == OR ==	START == AND == OR ==			On if A is equal to B. A and B are word/double word or data value.
STR <> AND <> OR <>	START <> AND <> OR <>			On if A is not equal to B. <> is same with ≠ . A and B are word/double word or data value.
STR > AND > OR >	START > AND > OR >			On if A is greater than B.
STR >= AND >= OR >=	START >= AND >= OR >=			On if A is equal to or greater than B.
STR <= AND <= OR <=	START <= AND <= OR <=			On if A is equal to or less than B.
STR < AND < OR <	START < AND < OR <			On if A is less than B.

NOTE For double word comparison instructions, the letter D should precede the word comparison instructions in the Mnemonic program.

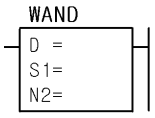
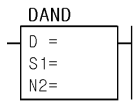
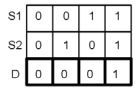
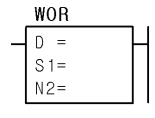
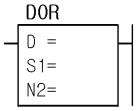
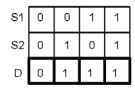
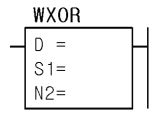
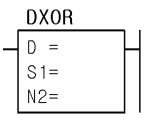
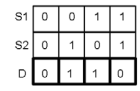
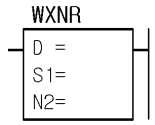
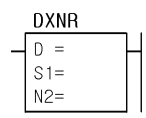
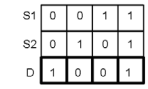
Substitution, Increment and Decrement Instructions

Mnemonic	Name	Word ladder symbol	Double word ladder symbol	Description
LET (DLET)	Let (Substitution)			Store the value of S into D.
INC (DINC)	Decimal increment			Increment D by 1 whenever input goes on.
INCB (DINCB)	BCD increment			Increment D by 1 in BCD mode whenever input goes on.
DEC (DDEC)	Decimal decrement			Decrement D by 1 whenever input goes on.
DECB (DDECB)	BCD decrement			Decrement D by 1 in BCD mode whenever input goes on.

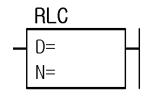
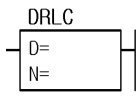
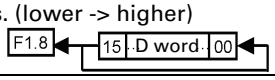
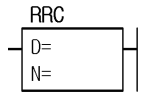
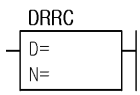
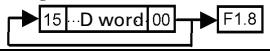
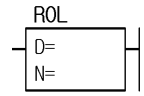
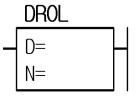
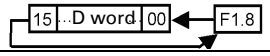
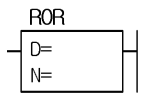
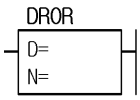
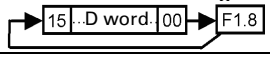
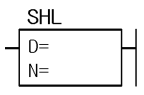
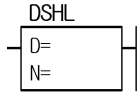
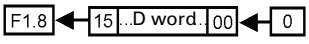
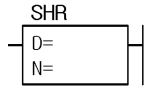
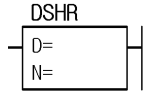
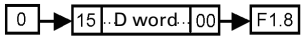
Arithmetic Instructions

Mnemonic	Name	Word ladder symbol	Double word ladder symbol	Description
ADD (DADD)	Decimal addition			D = S1 + S2 (Decimal operation)
ADDB (DADDB)	BCD addition			D = S1 + S2 (BCD operation)
SUB (DSUB)	Decimal subtraction			D = S1 - S2 (Decimal operation)
SUBB (DSUBB)	BCD subtraction			D = S1 - S2 (BCD operation)
MUL (DMUL)	Decimal multiplication			D = S1 x S2 (Decimal operation)
MULB (DMULB)	BCD multiplication			D = S1 x S2 (BCD operation)
DIV (DDIV)	Decimal division			D = S1/S2 (Decimal operation), Error when S2= 0
DIVB (DDIVB)	BCD division			D = S1/S2 (BCD operation) Error when S2 = 0
ADC (DADC)	Decimal addition with carry			D = S1 + S2 + CY (Decimal operation, include carry)
ADCB (DADCB)	BCD addition with carry			D = S1 + S2 + CY (BCD operation, include carry)
SBC (DSBC)	Decimal subtraction with carry			D = S1 - S2 – CY (Decimal operation, include carry)
SBCB (DSBCB)	BCD subtraction with carry			D = S1 - S2 - CY (BCD operation, include carry)
ABS (DABS)	Absolute value			D = |D| (Absolute value operation)
WNOT (DNOT)	NOT (1's complement)			Store 1's complement of D in D
NEG (DNEG)	Negative (2's complement)			Store 2's complement of D in D (1's complement + 1) (- Result)

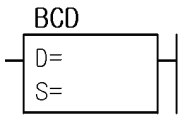
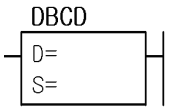
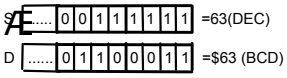
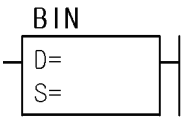
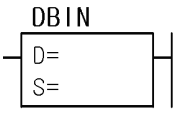
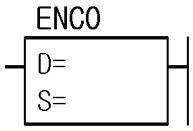
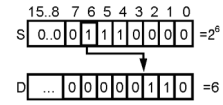
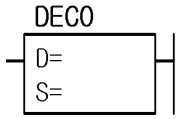
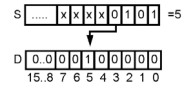
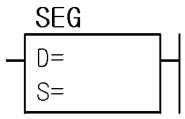
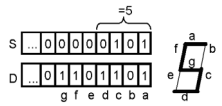
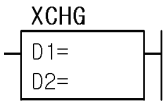
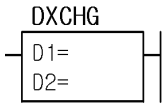
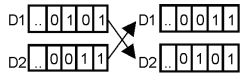
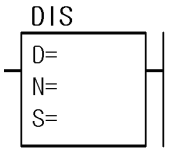
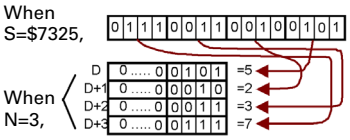
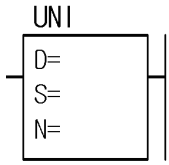
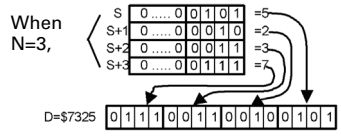
Logical Instructions

Mnemonic	Name	Word ladder symbol	Double word ladder symbol	Description
WAND (DAND)	AND (logical multiply)			Store AND of S1 and S2 in D 
WOR (DOR)	OR (logical sum)			Store OR of S1 and S2 in D 
WXOR (DXOR)	Exclusive OR (exclusive logical sum)			Store exclusive OR of S1 and S2 in D 
WXNR (DXNR)	Exclusive OR NOT (equivalence)			Store exclusive OR NOT of S1 and S2 in D 1 (ON if they are equal) 

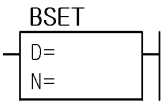
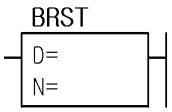
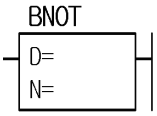
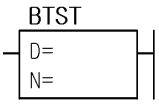
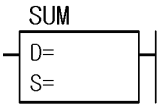
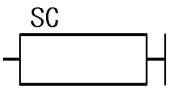
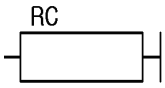
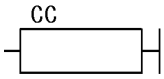
Rotation Instructions

Mnemonic	Instruction	Word ladder symbol	Double word ladder symbol	Description
RLC (DRLC)	Rotate left without carry			Rotate the content of D to the left N times. (lower -> higher) 
RRC (DRRC)	Rotates right without carry			Rotate the content of D to the right N times (higher -> lower) 
ROL (DROL)	Rotate left with carry			Rotate (shift) to the left N times (Input F1.8 value to the lowest bit) 
ROR (DROR)	Rotate right with carry			Rotate (shift) to the right N times (higher -> lower) (Input F1.8 value to the highest bit) 
SHL (DSHL)	Shift left			Shift the content of D to the left N times (input 0 to the lowest bit) 
SHR (DSHR)	Shift right			Shift the content of D to the right N times (input 0 to the highest bit) 

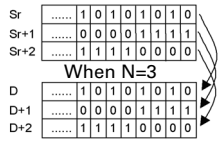
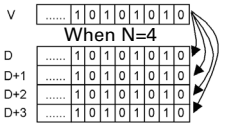
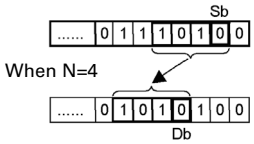
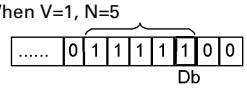
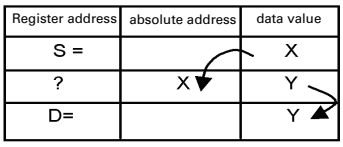
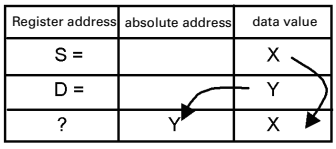
Word Conversion Instructions

Mnemonic	Name	Word ladder symbol	Double word ladder symbol	Description
BCD (DBCD)	BCD Conversion			Convert binary value of S to BCD and store it in D. 
BIN (DBIN)	Binary Conversion			Convert BCD of S to binary number and store it in D. 
ENCO	Encode		———	Store the location of the highest set bit in S in D.  Note) It is different than function of SPC series (2^n + 1)
DECO	Decode		———	Convert the low-order 4-bit value of S to a power of 2 (2^S) and store it in D. 
SEG	7-Segment		———	Converts the low-order 4-bit value of S to 7-segment display pattern and store them in D. 
XCHG (DXCHG)	Exchange			Exchange D1 and D2 values. 
DIS	Dissemble		———	Separate S into N+1 units, 4 bits each, and store them in the low 4 bits of words starting at D. 
UNI	Unify		———	Combine the low 4 bits of S+1 words starting at S, and store them in D (N= 0 to 3). 

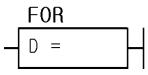
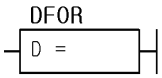
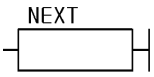
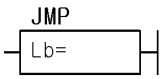
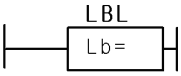
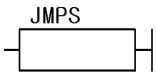
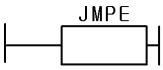
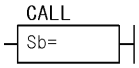
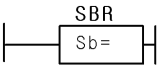
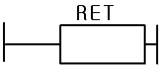
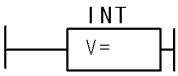
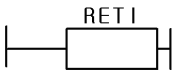
Bit Conversion Instructions

Mnemonic	Name	Word ladder symbol	Double word ladder symbol	Description
BSET	Bit Set		———	Set Nth bit of D to 1. D ..... 0 0 1 0 0 1 0 0 When N=15 ↑ 1 (N=0~15)
BRST	Bit Reset		———	Reset Nth bit of D to 0. D ..... 0 1 0 1 0 1 0 0 When N=3 ↑ 0
BNOT	Bit Not		———	Invert Nth bit of D. D ..... 0 1 1 1 0 1 0 0 When N=4 ↓ D 0 1 1 0 0 1 0 0
BTST	Bit Test		———	Store the value of Nth bit of D to F1.8. D ..... 0 1 1 1 0 1 0 0 When N=6 → F1.8
SUM	Sum		———	Store the number of bits in S that are 1 to D. S 0 0 0 1 1 0 1 0 0 1 1 0 1 0 0 1 No of 1=7 D 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 D=7
SC	Set Carry		———	Set carry bit (F1.8) to 1. 1 → F1.8
RC	Reset Carry		———	Reset carry bit (F1.8) to 0. 0 → F1.8
CC	Complement Carry		———	Invert carry bit (F1.8). F1.8 F1.8 1 → 0 0 → 1

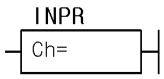
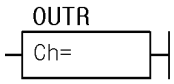
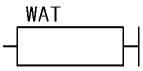
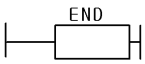
Move Instructions

Mnemonic	Name	Word ladder symbol	Double word ladder symbol	Description
MOV	Move	MOV D= S= N=	———	Copy Ns words from Sr to D. 
FMOV	Fill Move	FMOV D= N= V=	———	Repeatedly copy the value V to the Ns words starting from D. 
BMOV	Bit Move	BMOV Db= Sb= N =	———	Move Ns bits from the bit address Sb to the bit address Db. 
BFMV	Bit Fill Move	BFMV Db= N = V =	———	Repeatedly copy the bit value V to the N bits staring from the bit address Db. (V=0, 1)(N=1…256) (Db is bit address) 
LDR (DLDR)	Load D ← (S)	LDR D= S=	DLDR D= S=	Store to D the value of the register whose absolute address is the value of S. (Refer to the manual for information about absolute address.) 
STO (DSTO)	Store (D) ← S	STO S= D=	DSTO S= D=	Store the value of S to the register whose absolute address is the value of D. (Refer to the manual for information about absolute address.) 

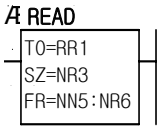
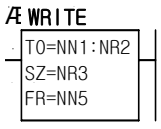
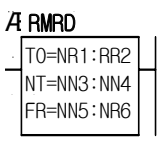
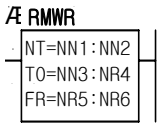
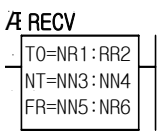
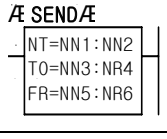
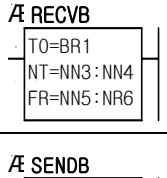
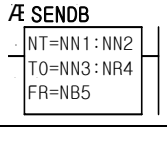
Program Control Instructions

Mnemonic	Name	Word ladder symbol	Double word ladder symbol	Description
FOR (DFOR)	For Loop			Execute instructions in the block between FOR and corresponding NEXT. Repeat execution D times.
NEXT	Next		———	Decrement D of FOR instruction by 1. If it is not zero, repeat execution from FOR instruction.
JMP	Jump		———	Jump to the position marked LBL L (label number). (L: 0 to 63)
LBL	Label		———	Position jumped to by the corresponding JMP instruction. (L:0 to 63)
JMPS	Jump Start		———	Jump to the JMPE instruction.
JMPE	Jump End		———	Position jumped to by the corresponding JMPS instruction.
CALL	Call Subroutine		———	Call subroutine Sb. (Sb = 0 to 63)
SBR	Subroutine Start		———	Start subroutine Sb. (Sb = 0 to 63)
RET	Subroutine Return		———	End of subroutine. Return execution to the instruction after CALL.
INT	Begin Interrupt		———	Begin the block of constant cycle scan instructions. Ni = 1 to 999 (20 msec to 10 sec) Constant cycle time = (Ni+1) x 0.01 sec
RETI	Return Interrupt		———	End the block of constant cycle scan instructions.

System Control Instructions

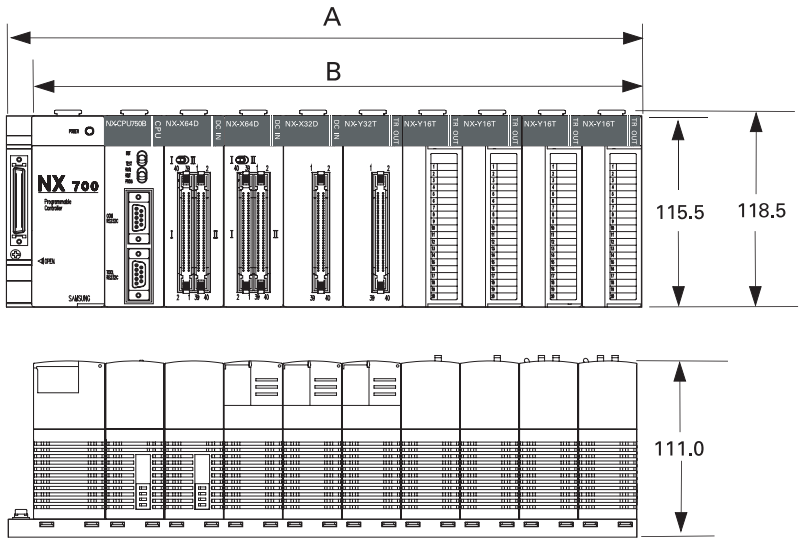
Mnemonic	Name	Word ladder symbol	Double word ladder symbol	Description
INPR	Input Refresh		———	Refresh external input (Receive input signal during program execution). Ch is external input word address.
OUTR	Output Refresh		———	Refresh external output (Send output signal during program execution). Ch is external output word address.
WAT	Watchdog Timer		———	Clear watchdog scan time.
END	END		———	End of program. This instruction is automatically added by WinGPC.

Communications Control Instructions

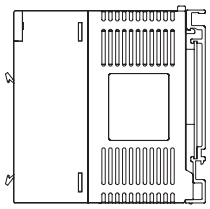
Mnemonic	Name	Word ladder symbol	Double word ladder symbol	Description
READ	Read Data (from shared memory of high performance module)		———	Read NR3 words from the module memory address NR6 of the slot NN5, and write them to the words starting from RR1.
WRITE	Write Data (to shared memory of high performance module)		———	Read NR3 words from NR5, and write them to the module memory address NR2 of slot NN1.
RMRD	Read Remote Slave Data (from shared memory of high performance module)		———	Read NR1 words from the module memory address NR6 of the slot NN5 for the station NN4 on the remote network loop NN3, and write them in words starting at RR2.
RMWR	Write Remote Slave Data (to shared memory of high performance module)		———	Read NR5 words from the module memory address NR6, and write them to the words starting from NR4 of the slot NN3 in the station NN2 on the remote network loop NN1.
RECV	Receive Link Data Word		———	Read NR1 words from the module memory NR6 of the slot NN5 in the station NN4 on the link network loop NN3, and write them to the words starting from RR2.
SEND	Send Link Data Word		———	Read NR5 words from the module memory NR6, and write them to the module memory starting from NR4 of the slot NN3 in the station NN2 on the link network loop NN1.
RECVB	Receive Link Data Bit		———	Read the bit NR6 of the slot NN5 in the station NN4 on the link network loop NN3, and write it the bit register BR1.
SENDB	Send Link Data Bit		———	Write the content of bit NB5 to the bit NR4 of the slot NN3 in the station NN2 on the link network loop NN1.

NX-CPU700p System Product Dimensions

System Dimensions



unit (mm)

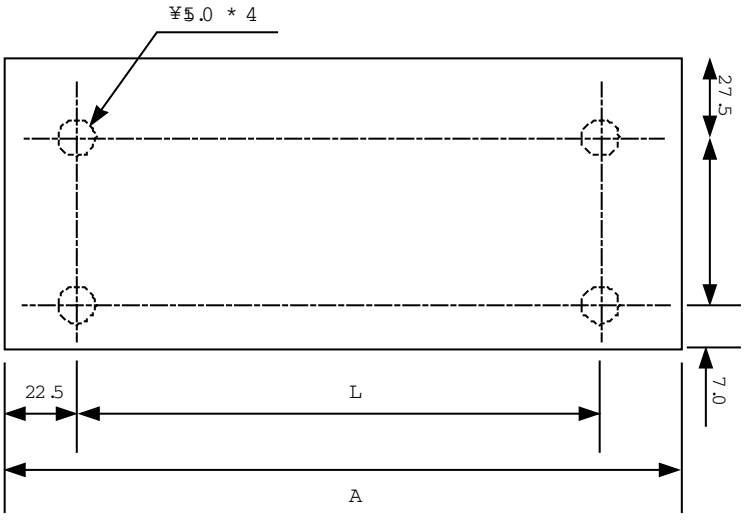


unit (mm)

	3-slot type	5-slot type	8-slot type	10-slot type	12-slot type
A (mm)	205.0	276.0	381.0	452.0	522.0
B (mm)	183.8	254.2	359.8	430.2	500.6

Backplane Dimensions

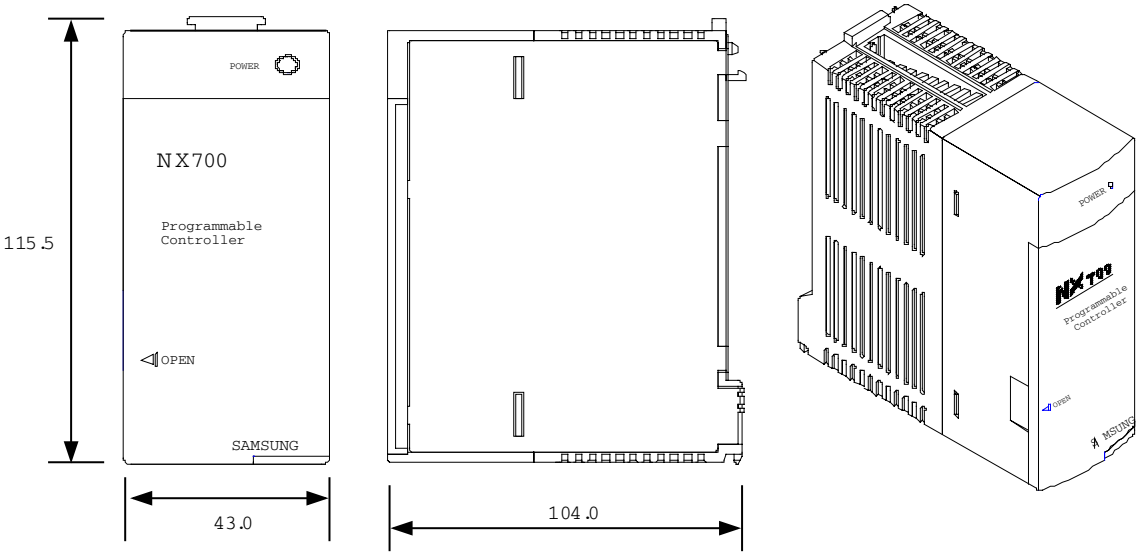
unit (mm)



Slots	A	L
3	205.0	153.8
5	276.0	224.2
8	381.0	329.8
10	452.0	400.2
12	522.0	470.6

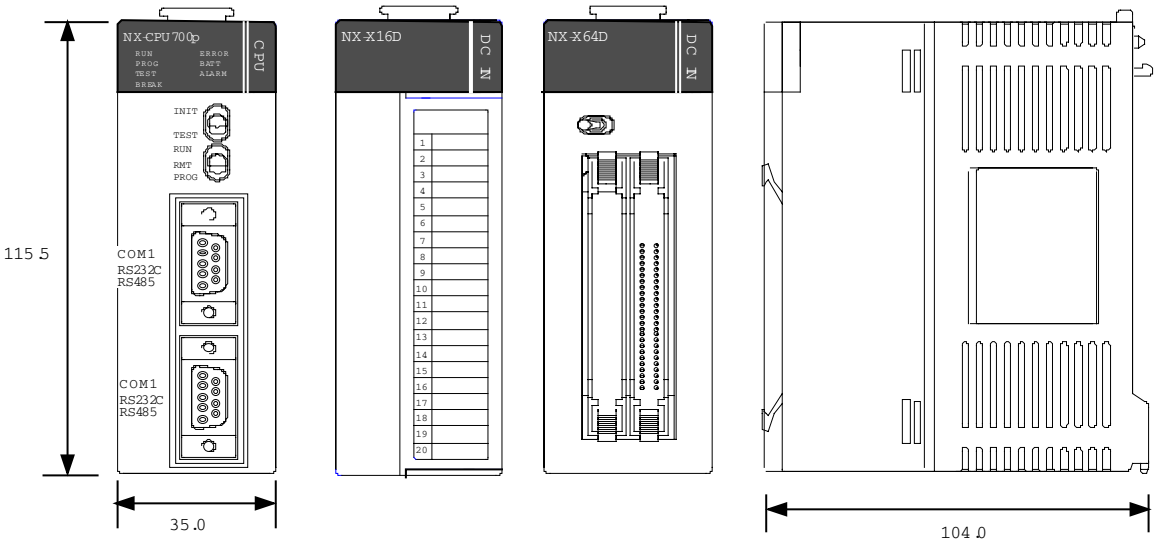
Power Supply Module Dimensions)

unit (mm)



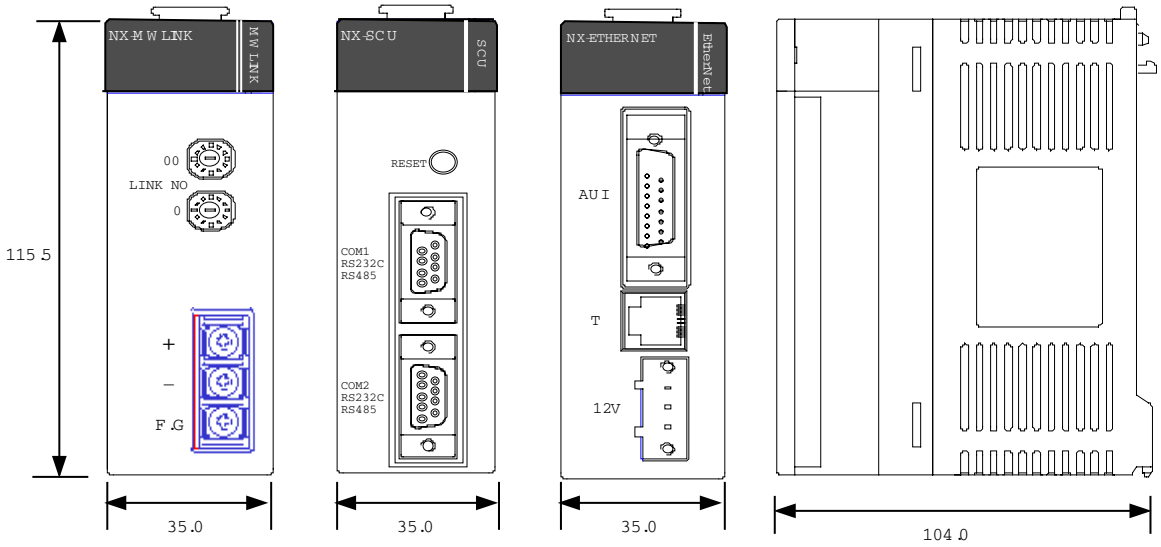
Processor and I/O Modules Dimensions

unit (mm)



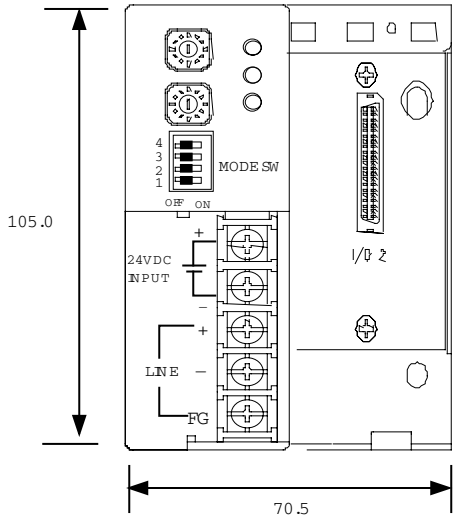
Specialty Module Dimensions

unit (mm)



NX-IOLINK Module Dimensions

unit (mm)



Decimal, Bin, Hex, BCD, Gray Code Cross-reference Table

Decimal	Hexadecimal	Binary	Binary coded decimal	Gray code
0	0000	00000000 00000000	0000 0000 0000 0000	0000 0000 0000 0000
1	0001	00000000 00000001	0000 0000 0000 0001	0000 0000 0000 0001
2	0002	00000000 00000010	0000 0000 0000 0010	0000 0000 0000 0011
3	0003	00000000 00000011	0000 0000 0000 0011	0000 0000 0000 0010
4	0004	00000000 00000100	0000 0000 0000 0100	0000 0000 0000 0110
5	0005	00000000 00000101	0000 0000 0000 0101	0000 0000 0000 0111
6	0006	00000000 00000110	0000 0000 0000 0110	0000 0000 0000 0101
7	0007	00000000 00000111	0000 0000 0000 0111	0000 0000 0000 0100
8	0008	00000000 00001000	0000 0000 0000 1000	0000 0000 0000 1100
9	0009	00000000 00001001	0000 0000 0000 1001	0000 0000 0000 1101
10	000A	00000000 00001010	0000 0000 0001 0000	0000 0000 0000 1111
11	000B	00000000 00001011	0000 0000 0001 0001	0000 0000 0000 1110
12	000C	00000000 00001100	0000 0000 0001 0010	0000 0000 0000 1010
13	000D	00000000 00001101	0000 0000 0001 0011	0000 0000 0000 1011
14	000E	00000000 00001110	0000 0000 0001 0100	0000 0000 0000 1001
15	000F	00000000 00001111	0000 0000 0001 0101	0000 0000 0000 1000
16	0010	00000000 00010000	0000 0000 0001 0110	0000 0000 0001 1000
17	0011	00000000 00010001	0000 0000 0001 0111	0000 0000 0001 1001
18	0012	00000000 00010010	0000 0000 0001 1000	0000 0000 0001 1011
19	0013	00000000 00010011	0000 0000 0001 1001	0000 0000 0001 1010
20	0014	00000000 00010100	0000 0000 0010 0000	0000 0000 0001 1110
21	0015	00000000 00010101	0000 0000 0010 0001	0000 0000 0001 1111
22	0016	00000000 00010110	0000 0000 0010 0010	0000 0000 0001 1101
23	0017	00000000 00010111	0000 0000 0010 0011	0000 0000 0001 1100
24	0018	00000000 00011000	0000 0000 0010 0100	0000 0000 0001 0100
25	0019	00000000 00011001	0000 0000 0010 0101	0000 0000 0001 0101
26	001A	00000000 00011010	0000 0000 0010 0110	0000 0000 0001 0111
27	001B	00000000 00011011	0000 0000 0010 0111	0000 0000 0001 0110
28	001C	00000000 00011100	0000 0000 0010 1000	0000 0000 0001 0010
29	001D	00000000 00011101	0000 0000 0010 1001	0000 0000 0001 0011
30	001E	00000000 00011110	0000 0000 0011 0000	0000 0000 0001 0001
31	001F	00000000 00011111	0000 0000 0011 0001	0000 0000 0001 0000
32	0020	00000000 00100000	0000 0000 0011 0010	0000 0000 0011 0000
63	003F	00000000 00111111	0000 0000 0110 0011	0000 0000 0010 0000
64	0040	00000000 01000000	0000 0000 0110 0011	0000 0000 0110 0000
255	00FF	00000000 11111111	0000 0010 0101 0101	0000 0000 0110 0000

ASCII Code Table

b8	b7	b6	b5	b4	b3	b2	b1	C / R	0	1	2	3	4	5	6	7
				0	0	0	0	0	NUL	DEL	SPACE	0	@	P	`	p
				0	0	0	1	1	SOH	DC_1	!	1	A	Q	a	q
				0	0	1	0	2	STX	DC_2	"	2	B	R	b	r
				0	0	1	1	3	ETX	DC_3	#	3	C	S	c	s
				0	1	0	0	4	EOT	DC_4	$	4	D	T	d	t
				0	1	0	1	5	ENQ	NAK	%	5	E	U	e	u
				0	1	1	0	6	ACK	SYN	&	6	F	V	f	v
				0	1	1	1	7	BEL	ETB	'	7	G	W	g	w
				1	0	0	0	8	BS	CAN	(	8	H	X	h	x
				1	0	0	1	9	HT	EM	)	9	I	Y	i	y
				1	0	1	0	A	LF	SUB	*	:	J	Z	j	z
				1	0	1	1	B	VT	ESC	+	;	K	[	k	{
				1	1	0	0	C	FF	FS	‘	<	L	₩	l	|
				1	1	0	1	D	CR	GS	-	=	M	]	m	}
				1	1	1	0	E	SO	RS	.	>	N	^	n	~
				1	1	1	1	F	SI	US	/	?	O	_	o	DEL

Appendix

Communication Protocols

The communication protocol of NX-CPU700p PLC provides a complete method of communications between the graphic consol programmers (WinGPC) and the PLC by controlling programs, CPU status, and I/O at user's convenience. The user can easily expand the capabilities of the overall PLC system by communicating to the PLC using a variety of peripheral communications equipment in accordance with the following communication protocols and procedures. Additionally, the communications protocol allows for the PLCs to communicate to a central computer on a single network using RS485, at a distance of up to 1.2 km (RS232C, 15 m).

Communication Protocols for NX-CPU700p

Communication environment

- Parity: No parity
- Stop bit: 1 stop bit
- Communication method: RS232 or RS485 (optional)
- Communication speed: 4800/9600/19200/38400 bps (optional)
- Communication cable: refer to the cable wiring diagram
- Number of PLCs on a single network: Maximum of 64 (communicating 1:N using RS485)
- Maximum communication delay time: 3 seconds

Communication Protocols

Step 1-Q **Query**

Q (Query) is a signal sent from the peripheral devices to the PLC after setting the network ID number and the function code for the PLC to communicate with.

Step 2-QA **Query Acknowledge**

QA (Query Acknowledge) is a signal sent from the PLC to the peripheral devices, indicating that the Q signal from the peripheral device was received.

Step 3-RR **Response Request**

RR (Response Request) is a signal sent from the peripheral device to the PLC, indicating that the QA signal from the PLC was received. This signal is sent when Q→QA is normal.

Step 4-R **Response**

When the PLC receives the RR from the peripheral device, it determines that the communication with peripherals is successful and sends R (Response) signal to the peripherals. This R signal contains how the original Q signal from the peripheral device handled its function code. The communication cycle for one function code ends when the PLC sends the R.

Communications delay

The PLC will return a signal after receiving a Q or an RR within a specific time. However, due to errors in the communications network, CRC values, and communication speed flux, there are occasions when the PLC will not receive the signal from the peripheral device. The peripheral device should allow up to three seconds for a response from the PLC. If there are no responses to the Q or the RR message, the communication is considered to have failed, and the Q or RR should be sent again.

CPU ID

All devices connected to the network need a network ID number for communication. There is an available range of 0 to 191 network ID numbers for the NX series.

Redundancy is not permitted. When a single PLC and a peripheral device are connected, usually 0, 1, or 255 is assigned as the network ID number to the PLC. When the peripheral device wants to communicate to a connected PLC regardless of its programmed network ID number, it can use global network ID number 255, to which any PLC will respond. However, the NX series can not be used to communicate with more than two CPU modules at one time, so if you assign ID 225 as an ID of more than two CPU modules at once, it will cause communication errors.

When several CPU modules are connected to one communication network, they must use individual ID numbers. The PLC's network ID number is configured using the WinGPC.

Communication steps

The NX CPU can support 2-step or 4-step communication methods. The communication methods are easily distinguished each other by selecting and sending the function code of the Q frame. Even for the 4-step method, the 2-step method can be used for the repeated function. This function sends and receives the only RR repeatedly when you want to redo the frame you sent with query, allowing users to quickly monitor data.

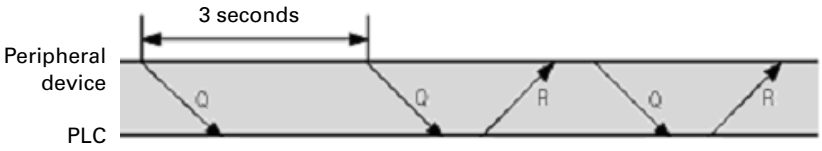
- **2-step communication method**
  This method allows users to easily and directly program communication since it only uses the simple Q → R steps.
  2-step configuration:
  Q(step 1) → R(step 2)
  Repeated function code:
  Q(step 1) → R(step 2) → RR(step 1) → R(step 2) → RR(step 1) → R(step 2)....

- **4-step communication method**
  Q → QA → RR → R.
  2-step method can be used for the response to the repeated function code.
  4-step configuration:
  Q(step 1) → QA(step 2) → RR(step 3) → R(step 4)
  Repeated function code:
  Q(step 1) → QA(step 2) → RR(step 3) → R(step 4) → RR(step 1) → R(step 2)...

2-step communication method

No communication error

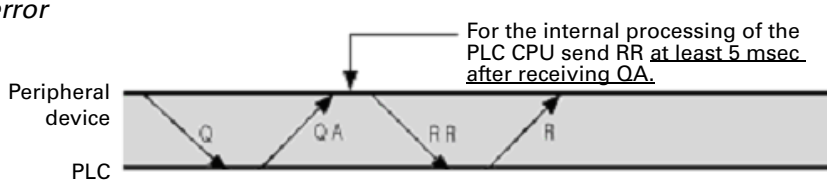
When R is not received



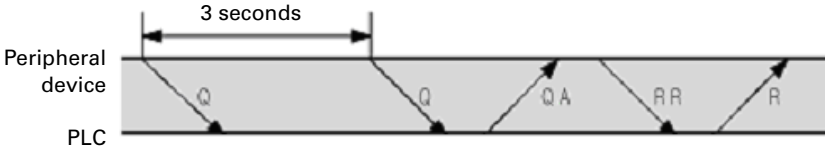
Response to repeated function code

4-step communication method

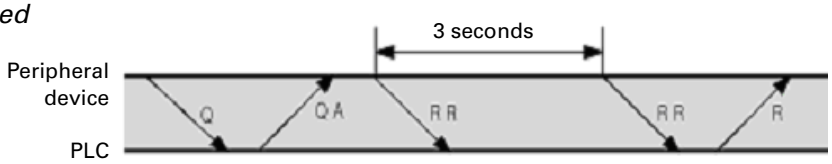
No communication error



When QA is not received



When R is not received



Response to repeated function code

Function codes included in the query

Each function code is 1 byte. When the PLC receives a query (Q), the function code of the final response (R) is formed by adding $80 (hex) to the function code sent by the query. The value added to the function code sent by the query differs for 2-step and 4-step by $20 (hex).

The function code of the R message can be used by the peripheral device to verify that the correct Q message has been received by the PLC.

Communication function codes

$ notes hexadecimal notations

Communication function	Query (Q) function code		Response (R) function code		Remarks
	2-step	4-step	2-step	4-step	
Read bits	$21	$01	$A1	$81	Detailed description
Write bits	$22	$02	$A2	$82	"
Read words	$23	$03	$A3	$83	"
Write words	$24	$04	$A4	$84	"
Read bits and words	$25	$05	$A5	$85	"
Write bits and words	$26	$06	$A6	$86	"
Read program	$27	$07	$A7	$87	No detailed description
Write program	$28	$08	$A8	$88	"
Read instruction	$29	$09	$A9	$89	"
Change instruction	$2A	$0A	$AA	$8A	"
Change operand	$2B	$0B	$AB	$8B	"
Insert instruction	$2C	$0C	$AC	$8C	"
Delete instruction	$2D	$0D	$AD	$8D	"
Search instruction	$2E	$0E	$AE	$8E	"
Search operand	$2F	$0F	$AF	$8F	"
Delete all/parts of program	$20	$10	$A0	$90	"
No service	$00	$00	$00 (hex)	$00 (hex)	"

The bit/word address assignment uses the absolute address method for reading memory locations. (Refer to *Absolute Addressing on Chapter 3*)

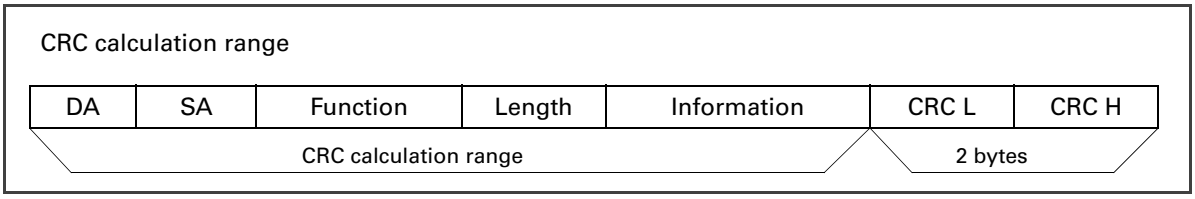
Please contact our technical support for more information about reading/writing program or other function codes.

Query, which dictionary meaning is `question', `ask', or `question mark', means that a user or an application program requests any specific information to a corresponding part when it is used as a communication term.

Cyclic Redundancy Checking (CRC)

The CRC is a 2-byte checksum code attached to the end of the message by the sender to check if the communication frame is transmitted without error.

The sender calculates the CRC when it sends one-byte message, and the receiver should also calculate the CRC from the data of the message. Since this CRC calculation takes a long time when writing a communication program, you should find any ways to increase the speed of this part to avoid errors and improve the communication speed.



CRC-16 calculation subroutine written in BASIC

```
        CRC_Sum: CRC-16 reserve code after the calculation (CRC content to be sent at the
        end of message)
        Data: CRC-16 data input to be calculated (byte data from message)

1000    CRC_Sum = CRC_Sum XOR Data
1010    FOR I=1 to 8
1020    CARRY=CRC_Sum AND 1
1030    RC_Sum=CRC_Sum SHR 1
1040    IF CARRY=1 THEN CRC_Sum XOR 0A001H
1050    NEXT I
1060    RETURN
```

CRC-16 calculation subroutine written in PASCAL

```
Procedure CRC16 (Data: Byte)
   Var i : Byte;
Begin
   CRC_Sum := CRC_Sum xor Data;
   for i : 1 to 8 do
Begin
   if((CRC_Sum and 1)=1) then CRC_Sum := (CRC_Sum shr 1) xor $A001;
                         Else CRC_Sum: = CRC_Sum shr 1;
   End;
End;
```

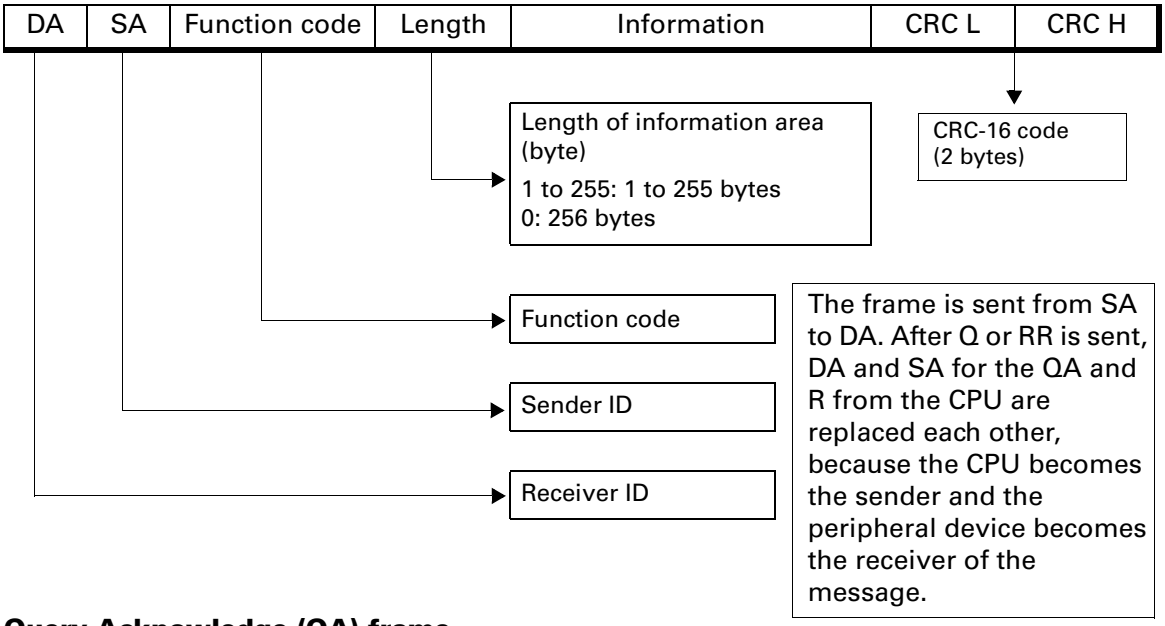
CRC-16 calculation subroutine written in C

```
Void Crc16 (unsigned int Data) {
        Unsigned int i;
        Crc=Crc^(Data & 0x00FF);
        for(i=0;i<=7;I++) {
                if((Crc & 0x0001) == 0x0001) Crc=(Crc>>1)^0xA001;
                else Crc=Crc>>1;
                }
        }
```

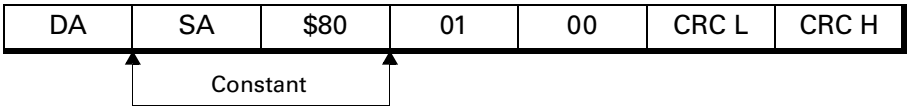
Structure of Communication Frames

The function code is explained with the example of Query and Response frame based on the 2-step communication.

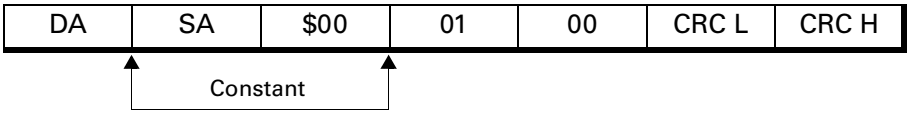
Query (Q) and Response (R) frame



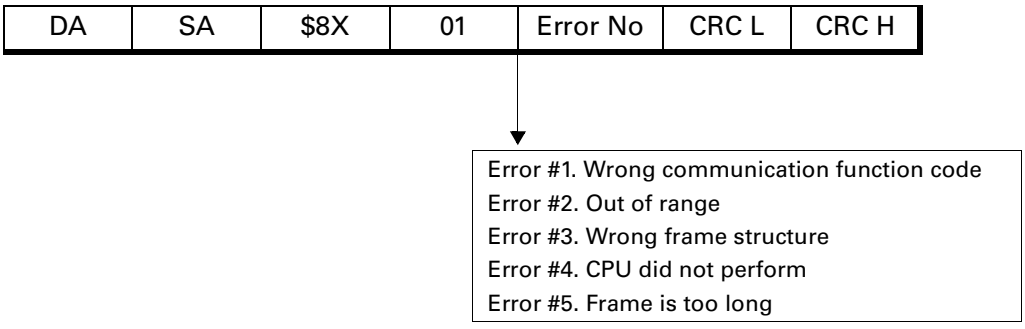
Query Acknowledge (QA) frame



Response Request (RR) frame



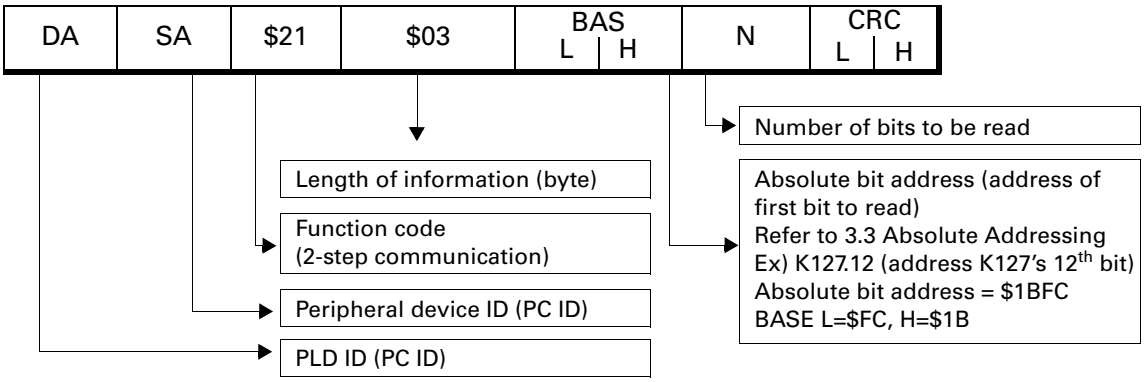
Response (R) frame for an error



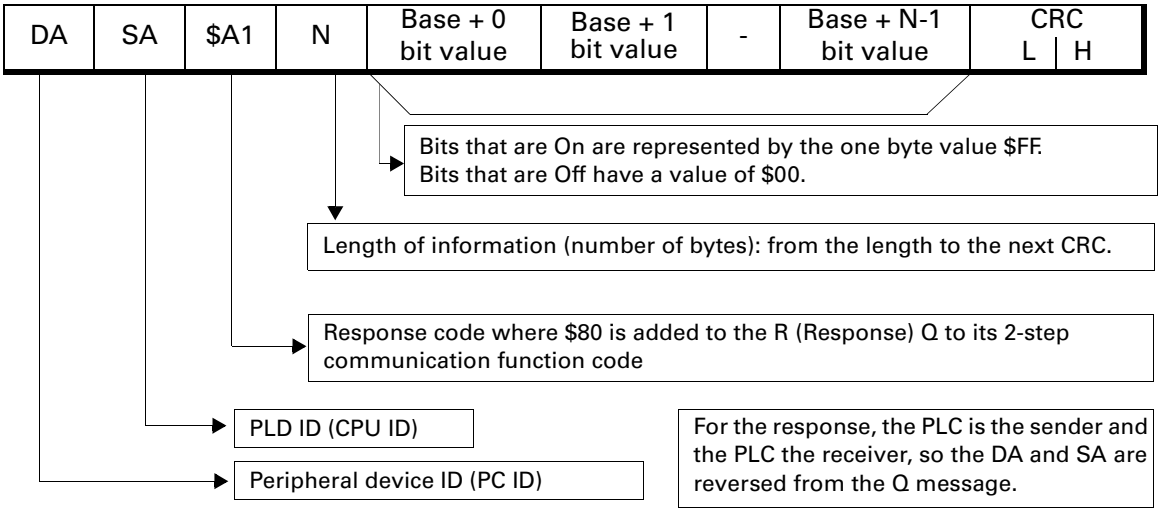
Read bits

Read the content of the bits (R, L, M, K, F, or TC) assigned to the absolute address. Can read n consecutive bits (On/Off).

Query (Q) frame



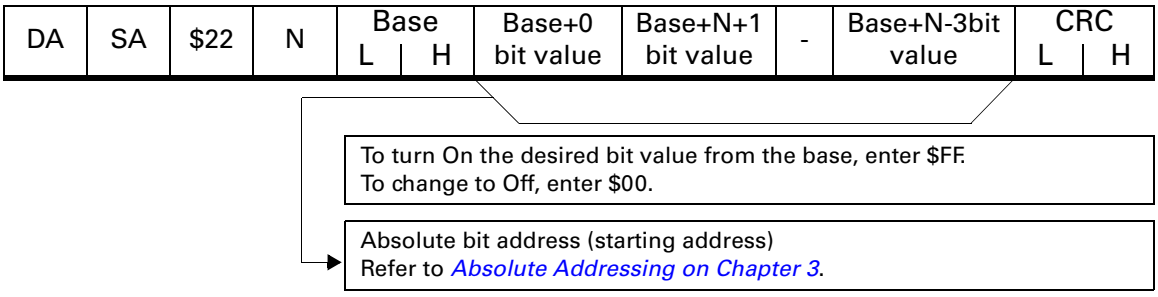
Response (R) frame



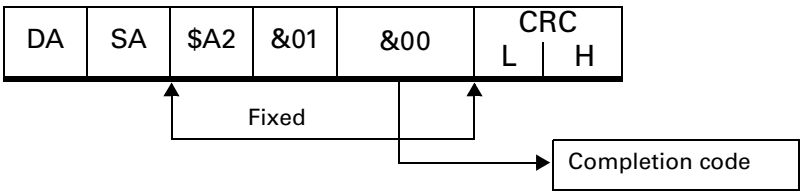
Write bits

Modify the contents of the bits stored in the absolute address (R, L, M, K, F, or TC).
Change the bit state between On/Off.
Can change multiple consecutive bytes.

Query (Q) frame



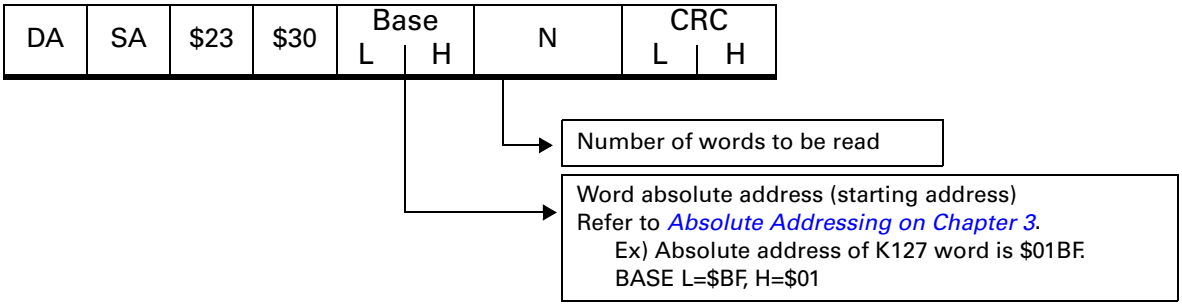
Response (R) frame



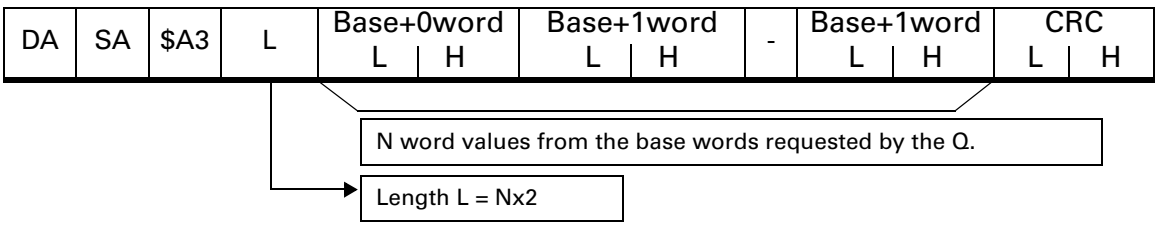
Read words

Read the content of the words (R, L, M, K, F, or W) assigned to the absolute address.
Can read n consecutive words.

Query (Q) frame



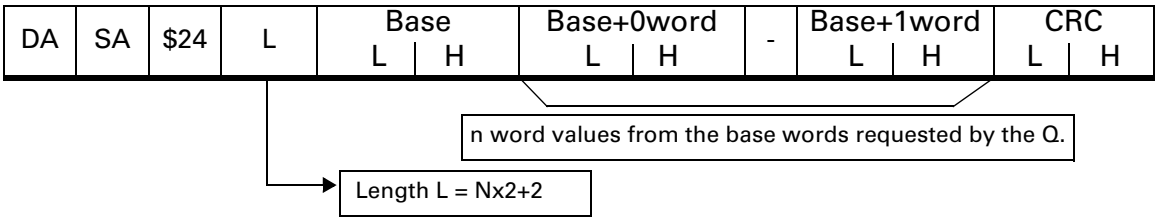
Response (R) frame



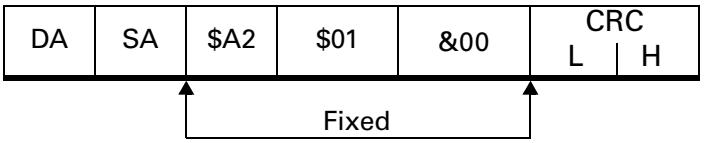
Write words

Change the content of the words (R, L, M, K, F, or W) assigned to the absolute address.
Can read n consecutive words.

Query (Q) frame



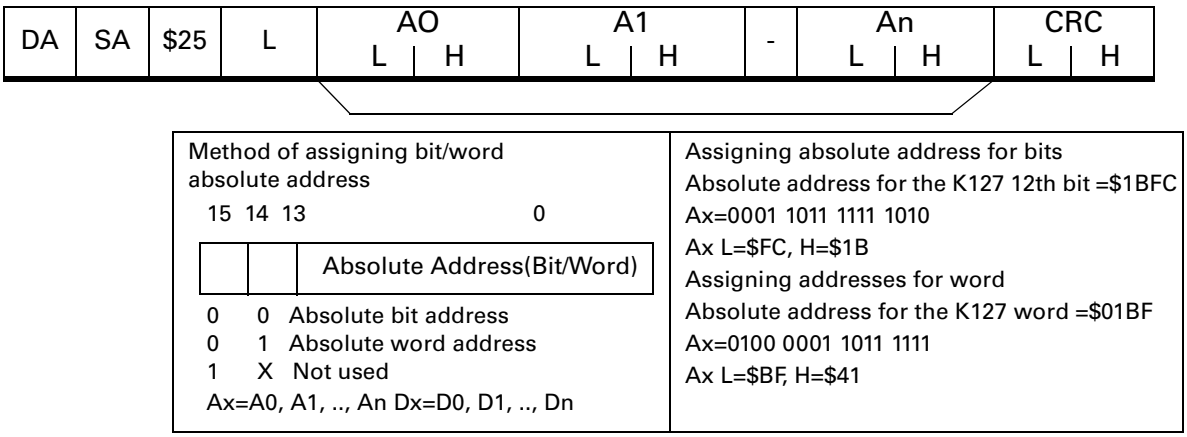
Response (R) frame



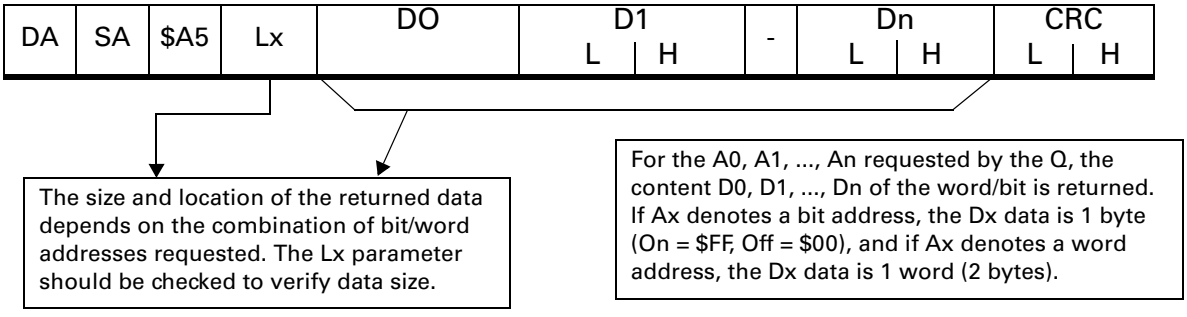
Read bits and words

Read the bits and/or word contents of the assigned absolute addresses.
Can read bits and words regardless of their order and location in memory.

Query (Q) frame



Response (R) frame

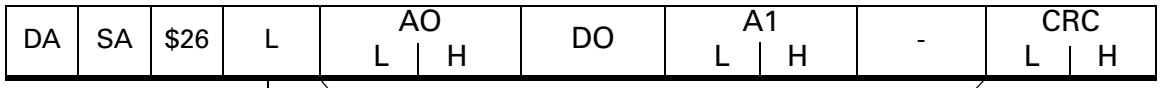


Write bits and words

Read the bits and/or word contents of the assigned absolute addresses.

Can read bits and words regardless of their order and location in memory.

Query (Q) frame



Method of assigning bit/word absolute address

15 14 13 0

		Absolute Address(Bit/Word)

0 0 Absoulte bit address
0 1 Absoulte word address
1 X Not used

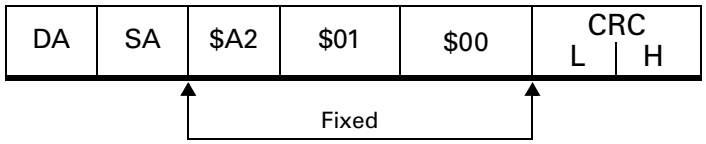
Ax=A0, A1, .., An Dx=D0, D1, .., Dn

Assigning absolute addresses for bits
Absolute address for the K127.12 bit = $1BFC
Ax=0001 1011 1111 1010
Assigning absolute addresses for word
The absolute address for the K127 word = $01BF
AX=0100 0001 1011 1111

When structuring the outgoing frame, be aware that the Dx of the Q changes according to the bit/word Ax type, and the L (information length) changes as well. The Dx will be either 1 or 2 bytes.

If Ax denotes a bit address, the Dx data is 1 byte (On=$FF, Off=$00), and if Ax denotes a word address, the Dx is 1 word (2 bytes).

Response (R) frame



Communication Program Examples

Users can write a communication program by using the following example. For more information, contact the sales or technical department.

Program	Notes
<PLC communication sample code> #include <stdio.h> #include <dos.h> #include <conio.h>	This program was written in Borland C++. It uses the peripheral devices such as PC to read M000 to M127 words, stores them in the K000 to K127, and then compares the two registry values and indicates the results on the screen using the OK or the FAIL notation. The user may read or manipulate the various communication function codes and the sent/ received information to control the PLC in various ways.
#define PC_ID 0xE2 #define time_limit 28 #define retrial_limit 2 #define TRUE 1 #define FALSE 0 #define lower_byte(x) (unsigned int) ((x)& 0x00FF) #define upper_byte (x) (unsigned int) (((x)& 0xFF00)>>8)	This program consists of a header, the main program, and various functions. The buffers and variables needed to store the communication data are set as global variables, so that the main and various other functions may reference them. By using the COM1 and COM2 ports of the computer, serial communication is possible. By using the GPU-300 card, parallel communication is also enabled.
typedef int BOOL; unsigned int PORTADD,DIVISOR,sending_delay, receiving_delay; unsigned int sending_frame[262],receiving_frame[262]; unsigned int Crc; unsigned int card,i,ix,iy,smode; unsigned int port_number; unsigned int PlcID,OldID; BOOL Success; unsigned int data,JobID,retrialC; unsigned int Old,New,receiving_Index_max,sending_Index_max, index,watchdog; unsigned int M[128],K[128]; /* Example Register */	The Qs, QAs, RRs, and Rs are handled in the job function. If there is any communication delay or frame breakdown, retry 3 times, then issue a communication error. The procedure of the communication, according to the JobID is: 1.Q sending 2.QA receiving 3.RR sending 4.R receiving When an error occurs in a frame, a retransmission should be made.

void RR_occurring(void);
void Trsport(unsigned int);
unsigned int Recport(void);
BOOL sending_occuring(void);
BOOL receiving_occuring (void);
void Crc16(unsigned int);
void Job(void);
unsigned int communication(void);
void Mword_reading(void);
void Kword_writing(void);
	<Main operations of the program>
	1. Adjusts the initial communication port and the board rate for communication. Then initializes the variables.
	2. Using the communication function codes, reads the data of the M field, reads the word values of the M0 to M127 word area. The K registers are the retentive registers.
	3. Uses the communication code to read the data of the K area.
	4. Compares the values of the M area and the values of the K area, and indicates OK when the values are the same.

void main(void)
{
unsigned int i;
/* Selection of communication port */
clrscr();
printf("PORT : COM1[1]/ COM[2]/ GPC-232[3]/GPC-485[4]/GPC-Parallel[5] = ");
scanf("%d",&port_number);
if ((port_number < 1) || (port_number > 5)) port_number=5;
	Beginning of the main program
	Select the port of the peripheral device for the communication:
	Serial 9 pins, 25 pins
	Parallel GPU-300 parallel port

/* Selection of Baudrate for Serial communication */
sending_delay=10;
if (port_number != 5)
{
printf("GPC card BAUD-RATE : 9600[1]/ 4800[2]/ 2400[3] = ");
scanf("%d",&i);
if ((i < 1) || (i > 3)) i=1;
if (i == 3) i=4;
	Select board rate:
	9600 bps (max):
	4800 bps
	2400 bps

if ((port_number == 1) || (port number == 2)) DIVISOR=12 * i;
else DIVISOR=40 * i;
receiving_delay=3 * i + 1;
}
	Set the communication environment (delay time) for the selected ports.
	Note: GPC-300 card port address = 0x0300

Program	Notes

* Initialization of GPC card */
if(port_number == 1) PORTADD=0x3F0;
if(port_number == 2) PORTADD=0x2F0;
if ((port_number >= 3) && (port_number <=5))
{
PORTADD=0x300;
outportb(0x303,0xC0);/* Mode=2 of 8255 */
outportb(0x303,0x05);/* PC2=1 of 8255 :Disable IRQ2 */
outportb(0x301,0xFF);/* PB0=1 of 8255 :Sending Enable RS-485*/
outportb(0x303,0x01);/* PC0=1 of 8255 :Serial Input Enable*/
if(port_number == 3) outportb(0x303,0x02);/* PC1=0 of 8255 :Select RS-232 */
if(port_number == 4) outportb(0x303,0x03);/* PC1=1 of 8255 :Select RS-485 */
if(port_number == 5) outportb(0x303,0x00);/* PC0=0 of 8255 :Disable SerialInput*/
}
else
outportb(PORTADD+0x09,(inportb(PORTADD+0x09)&0xF0));/ *Disable Interrupt*/
/* Initialization of USART-Chip : 8250 */
if (port_number != 5)
{
outportb(PORTADD+0x0B,0x80);/* Set of DLAB=1 */
outportb(PORTADD+0x09,0x00);/* Set of High Byte DIVISOR */
outportb(PORTADD+0x08,DIVISOR);/* Set of Low Byte DIVISOR */
outportb(PORTADD+0x0B,0x03); /* Parity=None/Stop=1/ Length=8 */
}
/* Processing communication of Read & Write */
for(; ;)
{
	GPC-300 card Setting (8255chip setting)
	Uses the communication card that is connected, and sets the environment according to the PLC communication specifications, so that communication is possible.

printf("----------\nPLC-ID (CPU ID) :");
scanf("%d",&PlcID);
Mword_reading();
Kword_writing();
}
}
	CPU-ID: Input PLC ID (0 to 255)
	
	Reads the register value for the M area (M0 to M127)
	Stores the value for the M area in the K area. (K0 to K127)

void RR_occuring(void)
{
receiving_frame[2]=0;
receiving_frame[3]=1;
receiving_frame[4]=0;
}
	RR (Request Response) request function

void Trsport(unsigned int data)
{
if (port_number == 5) outportb(PORTADD,data);
else outportb(PORTADD+0x08,data);
}
	Sends data to the communication port.

unsigned int Recport(void)
{
unsigned int dt;
if (port_number == 5) dt=inportb(PORTADD);
else dt=inportb(PORTADD+0x08);
return(dt);
}
	Reads the received data from the communication port.

BOOL sending_occuring(void)
{
BOOL tf;
if (port_number == 5) tf=((inportb(PORTADD+0x02) & 0x80)==0x80);
else tf=((inportb(PORTADD+0x0D) & 0x20)==0x20);
return(tf);
}
	Outputs the data when a Send event occurs..

BOOL receiving_occuring(void)
{
BOOL rf;
if (port_number == 5) rf=((inportb(PORTADD+0x02) & 0x20)==0x20);
else rf=((inportb(PORTADD+0x0D) & 0x01)==0x01);
return(rf);
}
	Inputs the data when a Receiver event occurs.

void Crc16(unsigned int data)
{
unsigned int i;
Crc=Crc^(data & 0x00FF);
for(i=0;i<=7;i++)
{
if((Crc & 0x0001) == 0x0001) Crc=(Crc>>1)^0xA001; /* 0x0001 : multi-nominal expression */
else Crc=Crc>>1;
}
}
	CRC calculation
	Encodes the communication data in the byte stream. Any completed communication function will be attached to the latest frame or will be compared with the attached CRC to check for data errors.
	(Note: The CRC method can be implemented in several ways within the rule specified as shown in the left code.)

Program	Notes
void Job(void) { /* JobID=0 : Change to sending-mode for serial port */ /* JobID=1 : Transmit sending-frame */ /* JobID=2 : Change to receiving-mode for serial port */ /* JobID=3 : Address polling of ACK from CPU */ /* JobID=4 : Receive ACK from CPU */ /* JobID=5 : Change to sending-mode for serial port */ /* JobID=6 : Transmit RR-Frame */ /* JobID=7 : Change to receiving-mode for serial port */ /* JobID=8 : Address polling of RES from CPU */ /* JobID=9 : Receive RES from CPU */ /* JobID=10 : Success communication processing */ switch(JobID) {	**Communication sequence function** JobID=0 to 4: handle Q and QA frames JobID=5 to 9: handle RA and R frames

case 0: case 5:if (port_number != 5)
{
if (port_number == 4) outportb(0x301,0xFF);
else outportb(PORTADD+0x0C,(inportb(PORTADD+0x0C) | 0x02));
delay(sending_delay);
}
if (JobID == 5) RR_occuring();
watchdog=0; index=0; sending_Index_max=5; Crc=0xFFFF; JobID++;
break;
	JobID 0,5:
	A frame that sends the data from the peripheral device to the PLC. It resets the watchdog and the CRC. Use a delay after the send to avoid errors due to communications delays.

case 1: case 6:if (receiving_occuring()) data=Recport();
if (sending_occuring())
{
if (index<sending_Index_max-1)
{
Trsport(receiving_frame[index]);
Crc16(receiving_frame[index]);
if (index==3)
{
if (receiving_frame[3]==0) sending_Index_max=256+5;
else sending_Index_max=receiving_frame[3]+5;
}
}
else if (index==sending_Index_max-1)
{
receiving_frame[index]=lower_byte(Crc);
Trsport(receiving_frame[index]);
}
else if (index==sending_Index_max)
{
receiving_frame[index]=lower_byte(Crc);
Trsport(receiving_frame[index]);
}
else if (index==sending_Index_max)
{
receiving_frame[index]=upper_byte(Crc);
Trsport(receiving_frame[index]); watchdog=0; JobID++;
}; index++;
}
break;
	JobID 1,6:
	Sends the Q and RR data. When there is no error, it resets the watchdog and proceeds on to the next sequence.

case 2: case 7:if (port_number != 5)
{
delay(receiving_delay);
if (port_number ==4) outportb(0x301,0x00);
else outportb(PORTADD+0x0C,(inportb(PORTADD+0x0C) & 0xFD));
}
JobID++;
break;
	JobID=2,7:
	A sequence that senses the sending of the QA and R data to the peripheral device after the completion of the functions that are received by the PLC from the previous frame.

case 3:
case 8:if (receiving_occuring())
{
data=Recport();
if(data==PC_ID)
{
Crc=0xFFFF; index=1; receivingIndexmax=5;
receiving_frame[0]=data; Crc16(data); JobID++;
}
}
	JobID=3,8:
	Handles the received data, and calculates the CRC of the received data.

Program	Notes

break;
case 4:
case 9:if(receiving_occuring())
{
if(index<receiving_Index_max-1)
{
receiving_frame[index]=Recport();
Crc16(receiving_frame[index]);
if(index==3)
{
if(receiving_frame[3]==0) receiving_Index_max=256+5;
else receiving_Index_max=receiving_frame[3]+5;
}
}
else if(index==receiving_Index_max-1)
{
 	
receiving_frame[index]=Recport();
if(receiving_frame[index]!=lower_byte(Crc)) JobID=(JobID & 0x05);
}
else if(index==receiving_Index_max)
{
receiving_frame[index]=Recport();
if(receiving_frame[index]==upper_byte(Crc)) JobID++;
else JobID=(JobID & 0x05);
}; index++;
}
break;
case 10:Success=TRUE;
}
}
	JobID=4,9:
	Stores the received data in the internal receive buffer and compares the CRC value sent by the PLC to the calculated CRC value. It notifies the system that a successful communication is made when the two values match, and proceeds on to the next sequence.
	
	JobID=10:
	Notifies the successful sending and receiving

unsigned int communication(void)
{
struct time t;
unsigned far *tm;
int ret;
Success=FALSE;
receiving_frame[0]=PlcID; receiving_frame[1]=PC_ID;
retrialC=retrial_limit;
watchdog=0; JobID=0; index=0; sending_Index_max=5; Crc=0xFFFF;
do
{
tm=(unsigned far *) 0x046C;
New=*tm;
Job();
if(watchdog>Time_limit)
{
watchdog=0; retrialC--;
JobID=(JobID & 0x05);
}
if(!(((Old^New) & 0x02)==0))
{
watchdog=watchdog+1;
Old=New;
}
}while((retrialC!=0) && (Success==FALSE));
if(retrialC==0) ret=1;
else ret=0;
return(ret);
}
	If the frames that were sent have no response within 3 seconds, assumes it failed communication, and retransfers the data.
	The time from the sending and receiving is counted using the watchdog timer.
	Resets the watchdog time when a retransfer is being made. No response after 3 transmissions indicates a communication error. (Normal return value = 0, Abnormal return value = 1)

void Mword_reading(void)
{
/* Example of Read-Register */
int i;
receiving_frame[2]=3;/* EXAMPLE READ WORD(M000-M0127) */
receiving_frame[3]=3;/* Number Of Byte For Information = 3 */
receiving_frame[4]=0xC0;/* BASE(M000=$00c0) */
receiving_frame[5]=0;/* BASE HIGH */
receiving_frame[6]=128;/* Number Of Byte M000-M127 */
if(communication() == 0)
{
printf("READ M0000-M0127 OK\n");
for(i=0;i<=127;i++) M[i]= receiving_frame[i*2+4] + receiving_frame[i*2+5]*256;
}
else printf("communication error\n");
}
	Reading function of the M register
	Uses the communication function code 3 (reading N consecutive words) to read the M area.
	Note:
	Sending frame [4] = The lower byte of the absolute address of the words to be read.
	Sending frame [5] = The upper byte of the absolute address of the word to be read.
	Absolute address of M0 = 0x0C0
	Note: Sending frame [6] = The number of words to be read.
	Sends a function code requesting to read the M area, and stores the received data in the buffer.

void Kword_writing(void)
{
/* Example of Write-Register */
int i;
receiving_frame[2]=4; /* EXAMPLE write WORD(K000-K063) */
receiving_frame[3]=130; /* Number Of Byte For Information */
receiving_frame[4]=0x40; /* BASE(K000=$0140) LOW */
receiving_frame[5]=1; /* BASE HIGH */
for(i=0;i<=63;i++)
{
receiving_frame[i*2 +6]= lower_byte(K[i]);
receiving_frame[i*2 +7]= upper_byte(K[i]);
}
	Writing Function of the K Register
	Uses the communication function code 4 (writing N consecutive words) to store the specified value in the K000 to K063 word.
	Note:
	Absolute address of K0 = 0x0140

Program	Notes

if(communication() == 0) printf("WRITE K0000-K0063 OK\n");
else printf("communication error\n");
receiving_frame[2]=4; /* EXAMPLE write WORD(K064-K0127) */
receiving_frame[3]=130; /* Number Of Byte For Information */
receiving_frame[4]=0x80; /* BASE(K000=$0180) LOW */
receiving_frame[5]=1; /* BASE HIGH */
for(i=0;i<=63;i++)
{
receiving_frame[i*2 +6]= lower_byte(K[i+64]);
receiving_frame[i*2 +7]= upper_byte(K[i+64]);
}
if(communication() == 0) printf("WRITE K0064-K0127 OK\n");
else printf("communication error\n");
}
	Writing function of the K Register
	Uses the communication function code 4 (writing N consecutive words) to store the specified value in the K064 to K127 word.
	Note:
	Absolute address of K64 = 0x0180

NX700 Series Controller NX-CPU700p User Manual

www.samsungautomation.co.kr

Rockwell Samsung Automation

447-6, Gongse-Ri, Giheung-Eup, Youngin-City,
Gyeonggi-Do, South Korea, 449-902
Tel: 82-31-280-4700 Fax: 82-31-280-4900

Technical Support

Export Sales Team
Tel: 82-31-280-4768 Fax: 82-31-280-4900

Publication RSA-NX700-UM002B-EN-P - July 2005