SHARP®

Version 2.0
Produced in June 1997

Sharp Programmable Controller

New Satellite *JW10*

User's Manual

We thank you for your purchase of the SHARP programmable controller JW10.
Carefully read this user's manual and the JW10 instruction manual attached to the system module of JW10 so that you are able to operate JW10 properly, having thoroughly familiarized yourself with the functions of the system module and their operation method.
Keep this user's manual with you together with the JW10 instruction manual.
We are confident that these booklets will be helpful whenever you face problem.

Note

This book applies the following abbreviations for indicating numerical expression methods such as addresses and setting values.

Octal .... (8)	Example : 377(8)
Decimal............ (D) or no indication	Example : 255(D), 255
Hexadecimal..... (H)	Example : FF(H)

Note

- This module is made in accordance with Japanese domestic specifications. Its guarantee clauses are described in a separate guarantee card (packed together with the module). When this module is used outside Japan, these guarantee clauses are not applicable. In addition, the guarantee should be understood as a guarantee of the delivered product as a single unit and every other damages or losses due to damage or malfunction of the product will not be included in this guarantee.
- Should you have any questions and inquiries, please feel free to contact our dealers.
- The whole or partial photocopy of this booklet is prohibited.
- Contents of this booklet may be revised for improvement without notice.

Safety Precautions

Read this manual and attached documents carefully before installation, operation, maintenance and checking in order to use the machine correctly. Understand all of the machine knowledge, safety information, and cautions before starting to use. In this instruction manual, safety precautions are ranked into "danger" and "caution" as follows.

◇! Danger	: Wrong handling may possibly lead to death or heavy injury.
⚠ Caution	: Wrong handling may possibly lead to medium or light injury.

Even in the case of ⚠ Caution, a serious result may be experienced depending on the circumstances. Anyway, important points are mentioned. Be sure to observe them strictly.

The picture signs of prohibit and compel are explained below.

⃠	: It means don'ts. For example, prohibition of disassembly is indicated as (⃠).
●	: It means a must. For example, obligation of grounding is indicated as (⏚).

1) Installation

⚠ Caution

- Use in the environments specified in the catalog and instruction manual.
  Electric shock, fire or malfunction may be caused when used in the environments of high temperature, high humidity, dusty or corrosive atmosphere, vibration or impact.
- Install according to the manual.
  Wrong installation may cause drop, breakdown, or malfunction.
- Never admit wire chips or foreign matters.
  Or fire, breakdown or malfunction may be caused.

2) Wiring

⏚ Compel

- Be sure to ground.
  Unless grounded, electric shock or malfunction may be caused.

⚠ Caution

- Connect the rated power source.
  Connection of a wrong power source may cause a fire.
- Wiring should be done by qualified electrician.
  Wrong wiring may lead to fire, breakdown or electric shock.

3) Use

Danger

- Don’t touch the terminal while the power is being supplied or you may have an electric shock.
- Assemble the emergency stop circuit and interlock circuit outside of the programmable controller. Otherwise breakdown or accident damage of the machine may be caused by the trouble of the programmable controller.

Caution

- Change of program durung operation, or "Run" or "stop" during operation should be done with particular care by confirming safety. Misoperation may lead to damage or accident of the machine.
- Turn on the power source in the specified sequence. Turning ON with wrong sequence may lead to machine breakdown or accident.

4) Maintenance

Prohibit

- Don’t disassemble or modify the modules.
  Or fire, breakdown or malfunction may be caused.

Caution

- Turn OFF the power source before detaching or attaching the module.
  Or electric shock, malfunction or breakdown may be caused.

SHARP **Programmable Controller**

New Satellite *JW10*

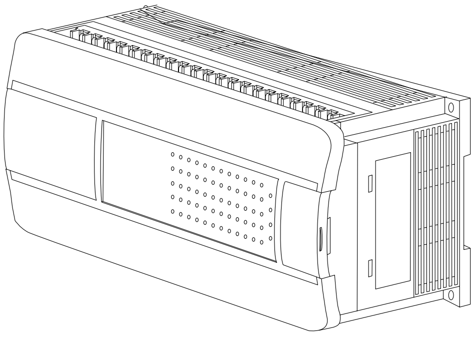


Table of contents

Chapter 1. Outline

The New Satellite JW10 is a programmable controller (hereafter referred to as "PC") for controlling small scale circuit systems up to 124 of input and output points.

Feature 1 **Selection of 13 modules**

JW10 is a compact and module type PC which has all components integrated one body. You can choose from six models of basic modules , five models of expansion module ,and analog input/output module in accordance with control scale and contents which are used.

Module name	Model name	Outline
Basic module	JW-1324K	DC input 16 points, relay output 12 points Unavailable for expansion
	JW-1342K	DC input 16 points(high speed response), transistor output 12 points Unavailable for expansion
	JW-1424K	DC input 24 points, relay output 16 points *1
	JW-1442K	DC input 24 points(high speed response), transistor output 16 points *1
	JW-1624K	DC input 36 points, relay output 24 points *1
	JW-1642K	DC input 36 points(high speed response), transistor output 24 points *1
Expansion module	JW-112N	DC input 16 points
	JW-112S	Transistor output 16 points
	JW-114S	Relay output 16 points
	JW-1124NS	DC input 8 points, relay output 8 points
	JW-1324NS	DC input 16 points, relay output 16 points
Analog input module	JW-14AD	4 channels 0 to 10V → Binary 12 bits *2 0 to 20mA → Binary 11 bits
Analog output module	JW-12DA	2 channels Binary 12 bits → 0 to 10V *2 Binary 11 bits → 0 to 20mA

*1 The basic modules, the JW-1424K, JW-1442K, JW-1624K, and JW-1642K, can be connected to 2 expansion modules, 1 analog input module, and 1 analog output module.

*2 The JW-14AD and JW-12DA can be connected to any basic module that is version 2.0 or later.

Feature 2 **Large memory capacity and variety of application instructions**

Though the JW10 is a compact PC, it is equipped with a large capacity data memory of 6,656 points of auxiliary relays and 2,048 bytes of registers. The JW10 has a variety of application instructions such as logical operation instruction, arithmetic operation instruction, comparison instruction, etc. so that it can be used for processing data.

Feature 3 **Equipped with three types of communication function as standard**

The JW10 is equipped with three types of communication functions: computer link, data link, and remote I/O, as standard. You can construct a network system without any additional modules. (Select one from the three communication functions for use.)

(1) Computer link

- Enables data communication between one host computer and maximum 63 modules of JW10.

(2) Data link

- Enables data communication between one JW10 master station and maximum seven modules of JW10 slave stations which can communicate 8 bytes of data for each transmission and reception per one station.

(3) Remote I/O

- Enables remote data input and output between one JW10 master station and maximum four modules JW10 slave stations and is able to transmit data through 36 points of input and 24 points of output per one station.

  Data link and remote I/O functions are available using upper model JW20H/30H as a master station and JW10 as slave stations.

Feature 4 **MMI port to be used for computer link**

Connection port (MMI port) of support tool can be used as computer link.

Feature 5 **High-speed counter is integrated**

The JW10 incorporates high-speed counter (two points of single-phase rising pulse input, or one point of 90-degree phase difference 2-phase signals) of maximum frequency 10 kHz so that it can easily receive pulses from rotary encoders.

Data converting software "Mitaro JW10" to export data to a table calculation software "Microsoft Excel*" is available. Using this software, JW10 data can be automatically exported to an Excel work sheet without need of any another software. For details, see the separate manual.

* Registered trade mark of Microsoft Corporation, USA.

■ Version of the basic module

The system ROM on the basic module has been upgraded by adding new functions. The relation between the additional functions and the corresponding ROM versions are listed in the following table.

Additional functions	ROM version	See page
Set system memory #055 (Set run/stop in error of program check)	1.4 or later	7-12
Connection of analog input module (JW-14AD) and analog output module (JW-12DA)	2.0 or later	15-1 to 15-10
Remote I/O slave station battery error flag	2.3 or later	13-32

Your system ROM version number can be checked at system memory address #041.
(See page 7-11)

Chapter 2. Safety Precautions

(1) Setup environment

Avoid keeping the JW10 in the following conditions:

- Direct sunlight.
- Relative humidity which exceeds the range of 5 to 90 %. No condensation due to rapid temperature variation.
- Corrosive and flammable gases.

(2) Emergency stop circuit

- Prepare an emergency stop circuit at an external relay circuit, and connect the halt output from the JW10. (The halt output is integrated in basic module JW-1424K/1442K/1624K/1642K.)
- Basic module JW-1324K/1342K does not have halt output. Provide a circuit to make output of JW-1324K/1342K normally ON, and connect this contact point into the emergency stop circuit. In this case, set system memory #206 to 00(H) (turn OFF output when the PC stops operation.)

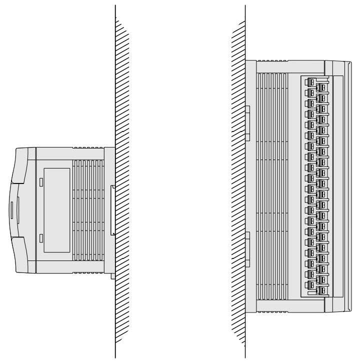
(3) Grounding

- Prepare a class-3 grounding of the JW10 separately. Never co-ground with high power equipment grounding lines.

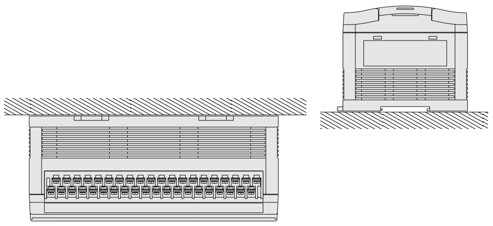
(4) Installation

- Securely fasten the fixing screws in each module, and confirm again that it is fastened prior to supplying power. Looseness of screws may cause malfunction.
- Firmly connect cable (expansion cable), connecting to the basic and expansion modules. Confirm connectors are fastened prior to supplying power. Looseness may cause malfunction.
- When JW-1424K/1442K/1624K/1642K is used as basic module, make sure to insert a termination connector on the end module.
- Each Module has a ventilation hole to allow for cooling. Do not block the holes.
- Install the JW10 vertically against a control panel.

Good; Vertical installation



No good; Horizontal installation



(5) Wiring

- Keep the input/output lines away from high voltage or strong current lines such as power lines.

(6) Cautions for static electricity

- Significant amounts of static electricity may build up on the human body in extremely dry conditions. Prior to touching the JW10, discharge the static electricity by touching grounded metals.

(7) Cleaning

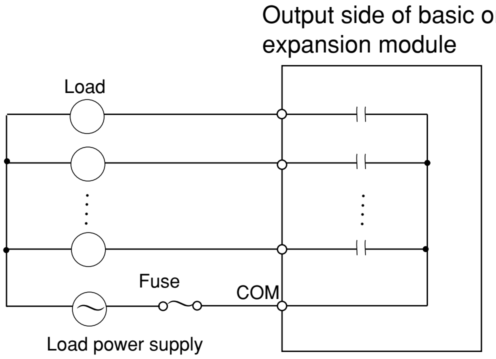
- Use the dry and soft cloths for cleaning. Volatile solvents (alcohol, paint thinner, freon etc.) and wet rags may cause deformation of a change of color.

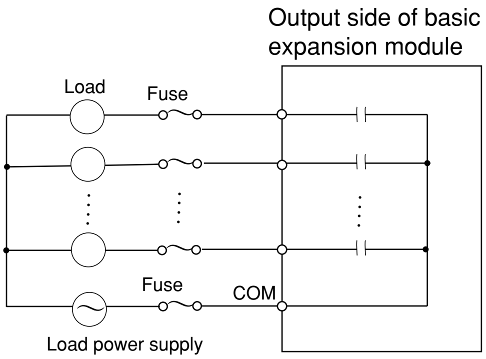
(8) Storage

- Keep the JW10 in cool and dry conditions as it is equipped with a battery for memory backup. High ambient temperature may shorten its battery life.
- Do not put other objects on the JW10.

(9) Overcurrent protection (fuse)

- Output circuit of the JW10 basic and expansion modules does not have fuse inside.
  If a load connected with the output terminals is shorted, it may cause burn out of an external wiring or the module. Therefore, insert protection fuses for outputs of each common line. These protection fuses are to prevent abnormal heat or burn out of modules due to overcurrent. These fuses do not protect output elements and loads.
- To secure safety, we recommend to provide fuses for each output point with enough capacity to meet each load.



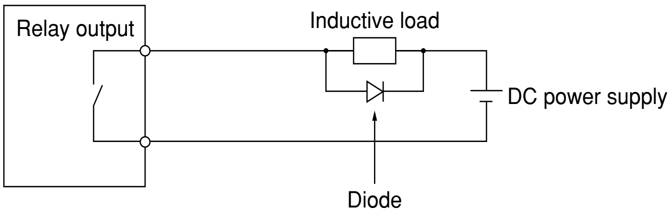


- When a fuse is blown out, solve the cause (short-circuit of an external wiring, use of load exceeding the rated output) and then replace the relative module.

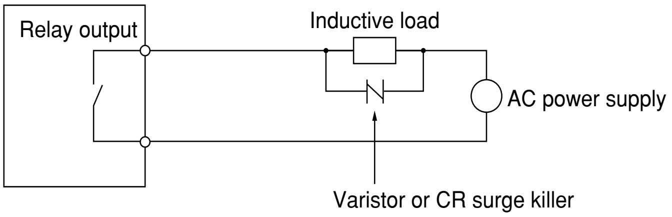
(10) Description of relay output [output port of JW-1324K/1424K/1624K/114S/1124NS/1324NS]

- A relay output is different from a transistor or other semiconductor output.Relays have a limited lifespan, determined by the number of times the relay contracts open and close.
  In addition, the actual life span of the relay output is greatly affected by the type of load connected, and whether or not there is a protection circuit for the relay.
- Generally, inductive loads such as relays and solenoid valves generate surge voltages when the output is turned OFF.This affects the lifespan of the relay.We recommend connecting a protective circuit, as shown below.(If the inductive loads already have integrated protection circuits, additional protective circuits are not necessary.)
  A relay output equipped with a protective circuit will generally extend its life to two to five times longer than that of relays without a protective circuit.
- For frequent open and close operations and for long continuous operation, we recommend using a transistor output instead.

In case of DC power supply



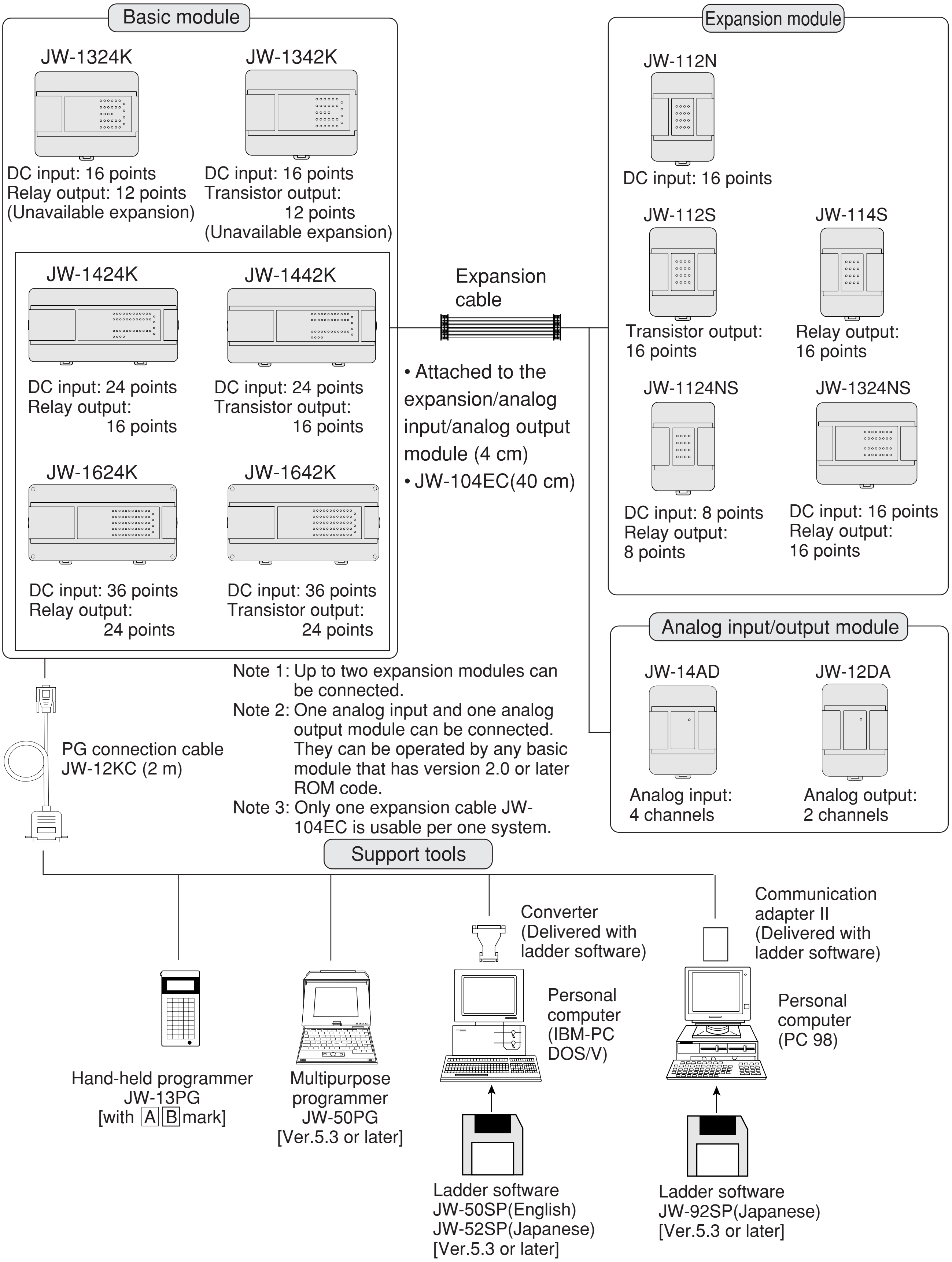
In case of AC power supply



Chapter 3. System Configuration

3-1 Basic system configuration

[1] System configuration



Note 1: Up to two expansion modules can be connected.

Note 2: One analog input and one analog output module can be connected. They can be operated by any basic module that has version 2.0 or later ROM code.

Note 3: Only one expansion cable JW-104EC is usable per one system.

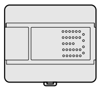
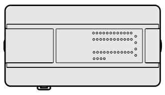
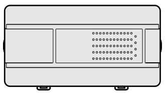
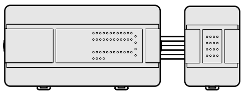
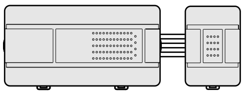
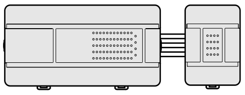
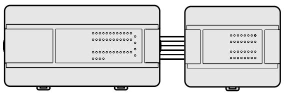
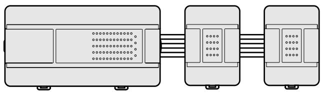
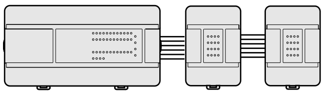
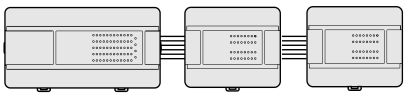
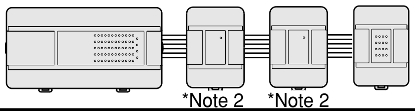
[2] Table of module

Module name	Model name	Outline	Accessories
Basic module	JW-1324K	Power voltage 85 to 250VAC 24 VDC input 16 points Relay output 12 points Max. No. of I/O points 28 points (unavailable expansion) Program capacity 1.5K words	Instruction manual 1
	JW-1342K	Power voltage 85 to 250VAC 24 VDC input (high speed response) 16 points Transistor output 12 points Max. No. of I/O points 28 points (unavailable expansion) Program capacity 1.5K words	Instruction manual 1
	JW-1424K	Power voltage 85 to 250VAC 24 VDC input 24 points Relay output 16 points Max. No. of I/O points 104 points Program capacity 4K words	Instruction manual 1
	JW-1442K	Power voltage 85 to 250VAC 24 VDC input 24 points Transistor output 16 points Max. No. of I/O points 104 points Program capacity 4K words	Instruction manual 1
	JW-1624K	Power voltage 85 to 250VAC 24 VDC input 36 points Relay output 24 points Max. No. of I/O points 124 points Program capacity 4K words	Instruction manual 1
	JW-1642K	Power voltage 85 to 250VAC 24 VDC input 36 points Transistor output 24 points Max. No. of I/O points 124 points Program capacity 4K words	Instruction manual 1
Expansion module	JW-112N	24 VDC input 16 points	Expansion cable (4cm) 1
	JW-112S	Transistor output 16 points	Expansion cable (4cm) 1
	JW-114S	Relay output 16 points	Expansion cable (4 cm) 1
	JW-1124NS	24 VDC input 8 points Relay output 8 points	Expansion cable (4 cm) 1
	JW-1324NS	24 VDC input 16 points Relay output 16 points	Expansion cable (4 cm) 1
Analog input module	JW-14AD	4 channels 0 to 10V→Binary 12 bits 0 to 12mA→Binary 11 bits	Instruction manual 1 Expansion cable (4cm) 1
Analog output module	JW-12DA	2 channels Binary 12 bits→0 to 10V Binary 11 bits→0 to 20mA	Instruction manual 1 Expansion cable (4cm) 1

Module name	Model name	Outline	Accessories
Expansion cable	JW-104EC	Expansion module connection cable (40 cm)	
PG connection cable	JW-12KC	A cable (2 m) to connect between basic module and support tool.	
Hand-held programmer	JW-13PG (with A ,B mark)	LCD dot matrix display Instruction words programmer (16 characters 4 lines)	Instruction manual 1 Lock spring 2 Programmer mounting bracket 1 Retention screw of programmer mounting bracket (M3x6) 1
Multipurpose programmer	JW-50PG (Ver. 5.3 or later)	LCD graphic display (640 × 480 dots) Built in 1 set of 3.5" floppy disk drive Built in 1 set of 2.5" hard disk drive (256 MB)	Instruction manual 5 Power supply cable 1 JW-52SP(FD) 1
Ladder software	JW-50SP (Ver. 5.3I or later)	Ladder software for IBM-PC	Instruction manual 1 Key label 1 Converter 1
	JW-52SP (Ver. 5.3 or later)	Ladder software for DOS/V personal computer, Japanese display	Instruction manual 1 Key label 1
	JW-92SP (Ver. 5.3 or later)	Ladder software for PC-98 series, Japanese display	Communication adapter II 1

[3] Example of system configuration

Figures in parenthesis are relay numbers (octal)

System configuration	Input / output	Basic module	Expansion module 1	Expansion module 2	Total
JW-1324K/1342K 	Input	16 points (00000 to 00017)	—	—	16 points
	Ouput	12 points (00400 to 00413)	—	—	12 points
JW-1424K/1442K 	Input	24 points (00000 to 00027)	—	—	24 points
	Ouput	16 points (00400 to 00417)	—	—	16 points
JW-1624K/1642K 	Input	36 points (00000 to 00043)	—	—	36 points
	Ouput	24 points (00400 to 00427)	—	—	24 points
JW-1424K JW-112N 	Input	24 points (00000 to 00027)	16 points (00030 to 00047)	—	40 points
	Output	16 points (00400 to 00417)	—	—	16 points
JW-1624K JW-114S 	Input	36 points (00000 to 00043)	—	—	36 points
	Output	24 points (00400 to 00427)	16 points (00430 to 00447)	—	40 points
JW-1624K JW-1124NS 	Input	36 points (00000 to 00043)	8 points (00050 to 00057)	—	44 points
	Output	24 points (00400 to 00427)	8 points (00430 to 00437)	—	32 points
JW-1424K JW-1324NS 	Input	24 points (00000 to 00027)	16 points (00030 to 00047)	—	40 points
	Output	16 points (00400 to 00417)	16 points (00420 to 00437)	—	32 points
JW-1624K JW-112N JW-1124NS 	Input	36 points (00000 to 00043)	16 points (00050 to 00067)	8 points (00070 to 00077)	60 points
	Output	24 points (00400 to 00427)	—	8 points (00430 to 00437)	32 points
JW-1424K JW-1124NS JW-112S 	Input	24 points (00000 to 00027)	8 points (00030 to 00037)	—	32 points
	Output	16 points (00400 to 00427)	8 points (00420 to 00427)	16 points (00430 to 00447)	40 points
JW-1624K JW-1324NS JW-1324NS 	Input	36 points (00000 to 00043)	16 points (00050 to 00067)	16 points (00070 to 00107)	68 points
	Output	24 points (00400 to 00427)	16 points (00430 to 00447)	16 points (00450 to 00467)	56 points
JW-1424K JW-14AD JW-12DA JW-1124NS  *Note 2 *Note 2	Input	24 points (00000 to 00027)	8 points (00030 to 00037)	—	32 points
	Output	16 points (00400 to 00417)	8 points (00420 to 00427)	—	24 points

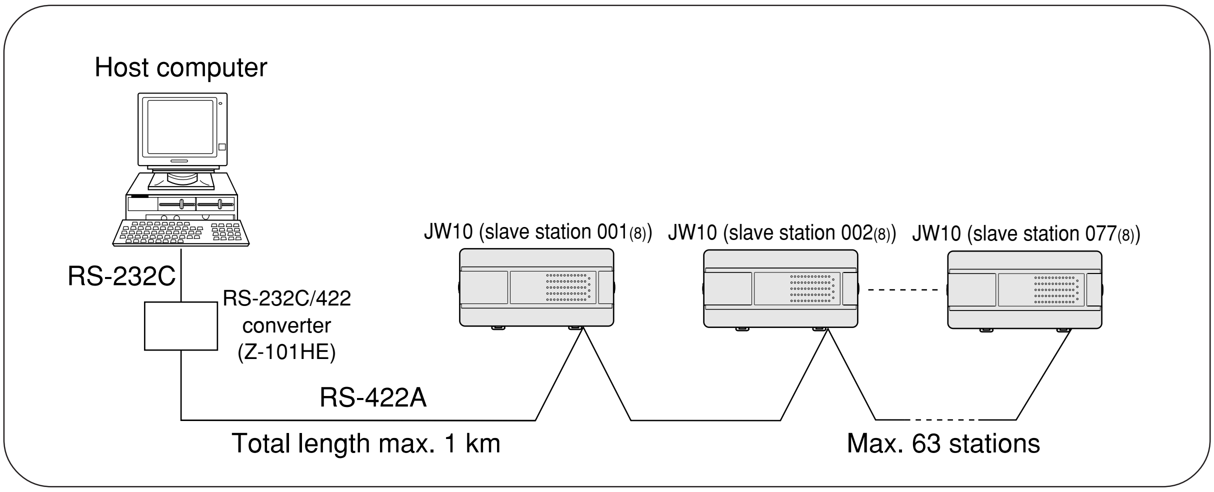
*Note 1 : For allocation of relay number, see page 10・3.

*Note 2 : JW-14AD and JW-12DA do not occupy I/O relay area.

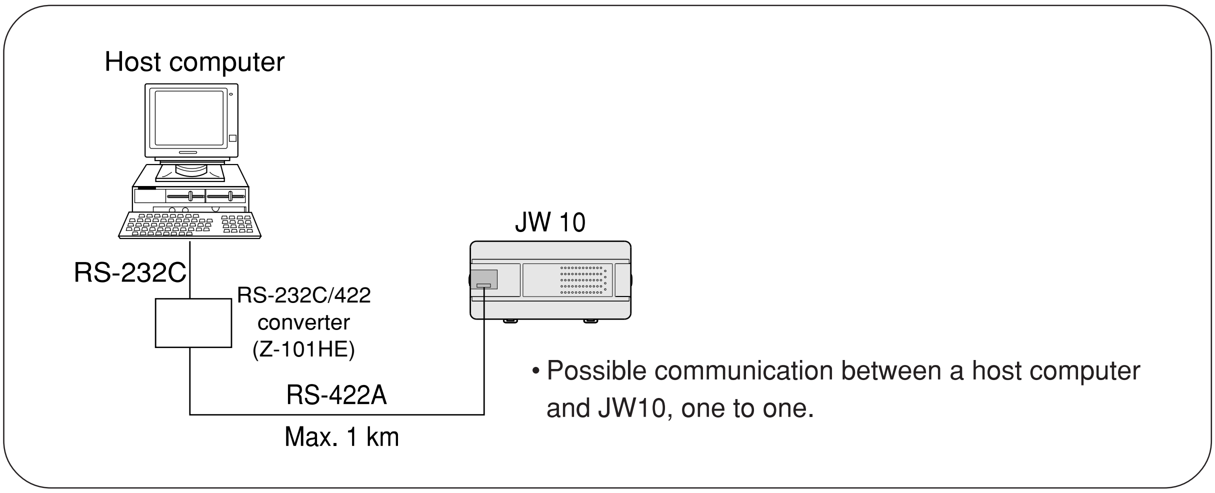
3-2 System configuration using communication

[1] Computer link system

(1) In case of using communication port


Host computer
RS-232C
RS-232C/422 converter (Z-101HE)
JW10 (slave station 001(8))
JW10 (slave station 002(8))
JW10 (slave station 077(8))
RS-422A
Total length max. 1 km
Max. 63 stations

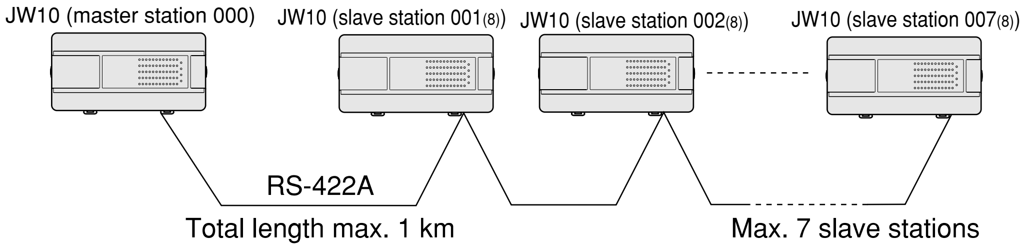
(2) In case of using MMI port



- Possible communication between a host computer and JW10, one to one.

[2] Data link system

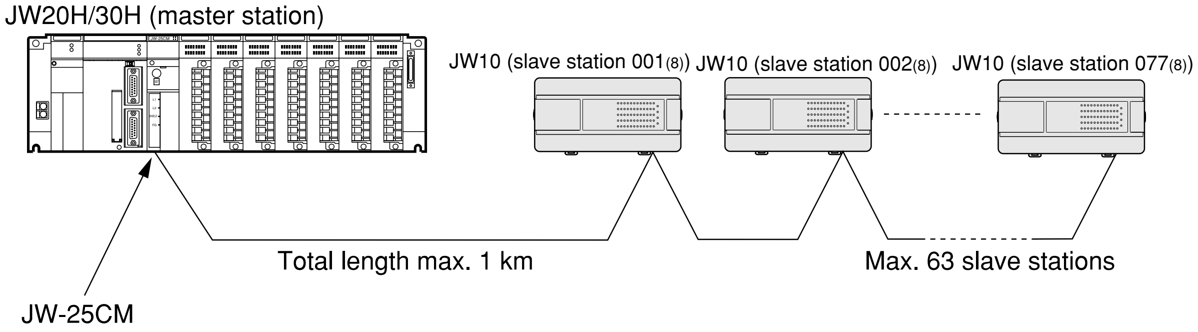
(1) In case of using the JW10 as master station


JW10 (master station 000)
JW10 (slave station $001_{(8)}$)
JW10 (slave station $002_{(8)}$)
JW10 (slave station $007_{(8)}$)
RS-422A
Total length max. 1 km
Max. 7 slave stations

- Possible communication between one JW10 master station and maximum seven JW10 slave stations, one to N modules communication. (Communication between slave stations is not available.)
- Transmitting and receiving data capacity per station is 8 bytes.
- Total extendible length is determined by setting communication speed.

Communication speed	Total length distance	No. of connected slave stations
76800 bits/s	500m	7 stations
38400 bits/s	1 km	7 stations

(2) In case of using the JW20H/30H as master station


JW20H/30H (master station)
JW10 (slave station $001_{(8)}$)
JW10 (slave station $002_{(8)}$)
JW10 (slave station $077_{(8)}$)
Total length max. 1 km
Max. 63 slave stations
JW-25CM

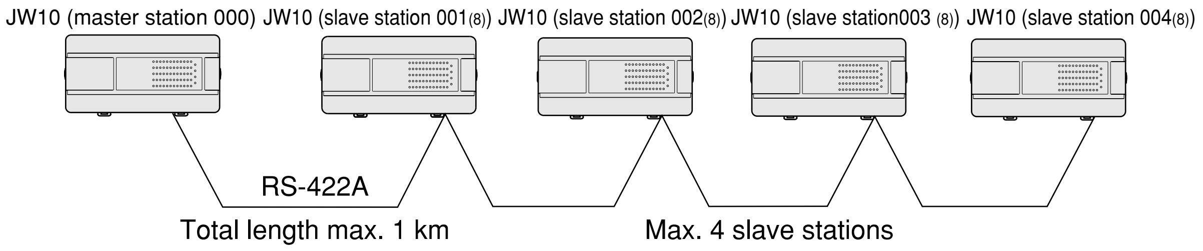
- Possible communication between one JW20H/30H master station and JW10 slave stations, one to N modules communication. (Communication between slave stations is not available.)
- Transmitting and receiving data capacity per station is 8 bytes.
- Total extendible length and maximum number of connectable slave stations are determined by setting communication speed.

Communication speed	Total length distance	No. of connected slave stations
76800 bits/s	500m	31 stations
38400 bits/s	1 km	63 stations

- See "JW-25CM instruction manual" in detail for this system.

[3] Remote I/O system

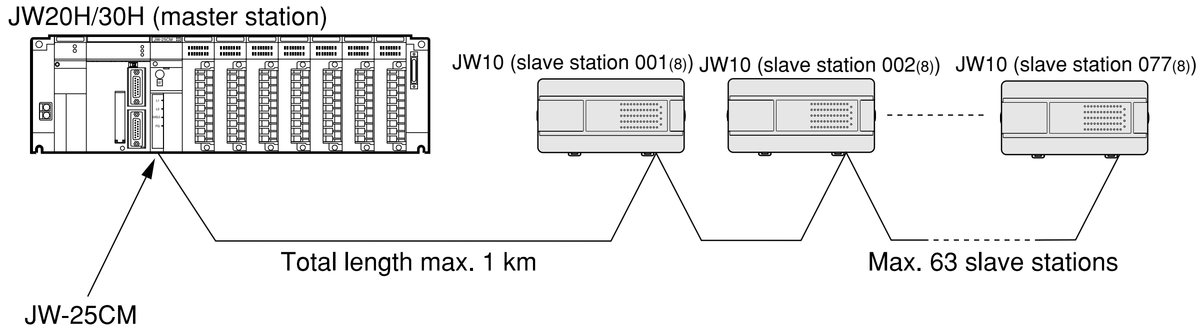
(1) In case of using the JW10 as master station


JW10 (master station 000)
JW10 (slave station 001(8))
JW10 (slave station 002(8))
JW10 (slave station003 (8))
JW10 (slave station 004(8))
RS-422A
Total length max. 1 km
Max. 4 slave stations

- No. of I/O points per station is max. 60 points (input 36 points, output 24 points)
- Connection of an expansion module to a slave station is not available
- Data exchage between a master station and slave stations synchronize with master station's operation speed.
- Total extendible length is determined by setting communication speed.

Communication speed	Total length distance	No. of connected slave stations
76800 bits/s	500m	4 stations
38400 bits/s	1 km	4 stations

(2) In case of using the JW20H/30H as master station


JW20H/30H (master station)
JW10 (slave station 001(8))
JW10 (slave station 002(8))
JW10 (slave station 077(8))
Total length max. 1 km
Max. 63 slave stations
JW-25CM

- No. of I/O points per station is max. 60 points (input 36 points, output 24 points)
- Connection of an expansion module to a slave station is not available.
- Data exchage between a master station and slave stations can be selected from synchronous or asynchronous with the operation.
- Total extendible length and maximum number of connectable slave stations are determined by setting communication speed and selection between synchronous and asynchronous.

Communication speed	Operation synchronous/ asynchronous	Total length distance	No. of connected slave stations
76800 bits/s	Synchronous/ Asynchronous	500 m	31 stations
38400 bits/s	Synchronous	1 km	16 stations
	Asynchronous	1 km	63 stations

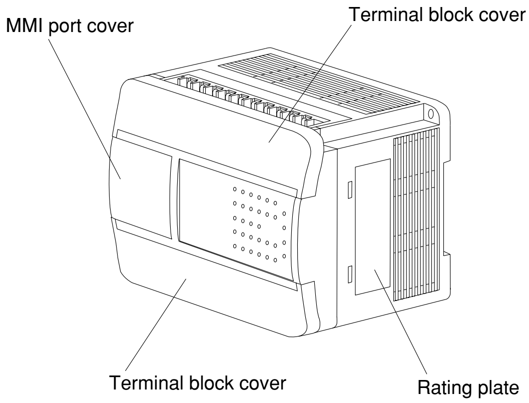
•See "JW-25CM instruction manual" in detail for this system.

Chapter 4. Name and Function of Each Part

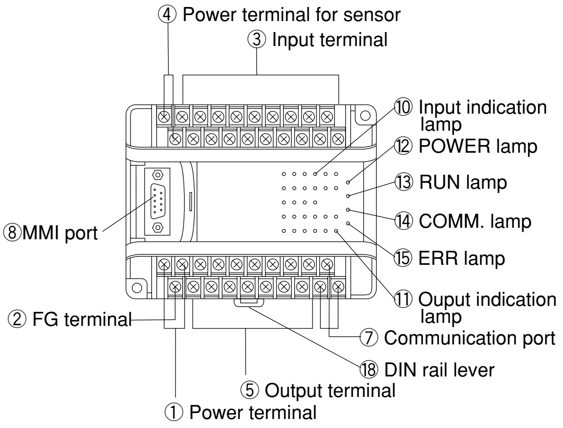
4-1 Basic module

Six models : JW-1324K, JW-1342K, JW-1424K, JW-1442K, JW-1624K, and JW-1642K, are available as basic module of JW10.

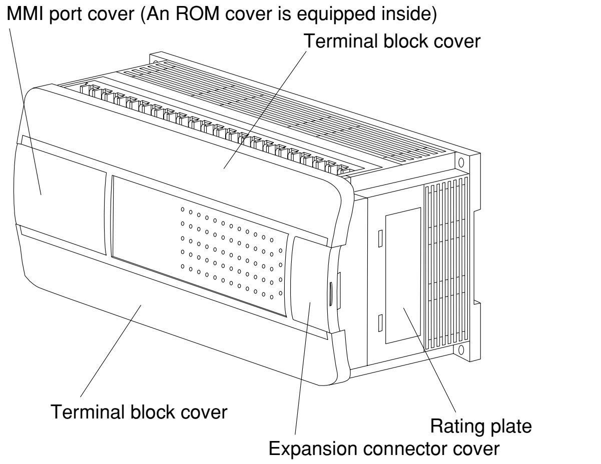
JW-1324K JW-1342K



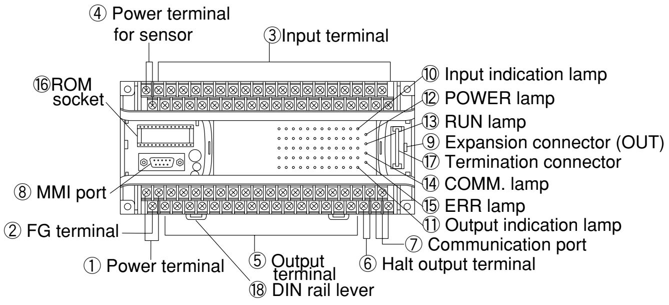
(Each cover opened condition)



JW-1424K JW-1442K JW-1624K JW-1642K


MMI port cover (An ROM cover is equipped inside)
Terminal block cover
Terminal block cover
Rating plate
Expansion connector cover

(Each cover opened condition)



① **Power terminal**

Supply 85 to 250 VAC power.

② **FG terminal**

Connect class-3 grounding.

③ **Input terminal**

Connect a cable from an input device.

Terminals 0 to 3 can be used for input of a high-speed counter.

④ **Power terminal for sensor**

Can be used for driving power of an external sensor (24 VDC).

(JW-1324K/1342K : 0.3A,
JW-1424K/1442K/1624K/1642K : 0.4A)

⑤ **Output terminal**

Connect a cable from an output device.

⑥ **Halt output terminal [JW-1424K/1442K/1624K/1642K only]**

A relay output terminal which will open when JW10 is stopped operation.

⑦ **Communication port**

Connect a communication cable when using computer link, data link, or remote I/O.

⑧ **MMI port**

Insert a PG connection cable (JW-12KC) to connect with a support tool such as JW-13PG.

Or, insert a communication cable when using computer link.

⑨ **Expansion connector (OUT) [JW-1424K/1442K/1624K/1642K only]**

Insert an expansion cable (4 cm) supplied with an expansion module or JW-104EC (40 cm) for connection with an expansion module for expansion of I/O.

In case of the expansion module is not connected, insert termination connector ⑰ in this connector.

⑩ **Input indication lamp (amber)**
Lights is when the connected input devices are functioning.
(JW-1324K/1342K: 16 pieces, JW-1424K/1442K: 24 pieces, JW-1624K/1642K: 32 pieces)

⑪ **Output indication lamp (amber)**
Lights when the outputs are functioning.
(JW-1324K/1342K: 12 pieces, JW-1424K/1442K: 16 pieces, JW-1624K/1642K: 24 pieces)

⑫ **POWER lamp (green)**
Lights when a power supply inside the basic module is supplying 5 V power.

⑬ **RUN lamp (green)**
- During normal operation: Lights ON
- During programming using support tool (stopped PC operation): Blinking
- Detecting an error with self diagnosis function: Goes OFF (Lights when battery error is occurred.)

⑭ **COMM. lamp**
Lights during communication through computer link, data link, or remote I/O using the communication port.

⑮ **ERR lamp (red)**
When an error is detected with self diagnosis function, this lamp lights and PC stops operation. (In case when battery error has occurred, the PC continues operation.)

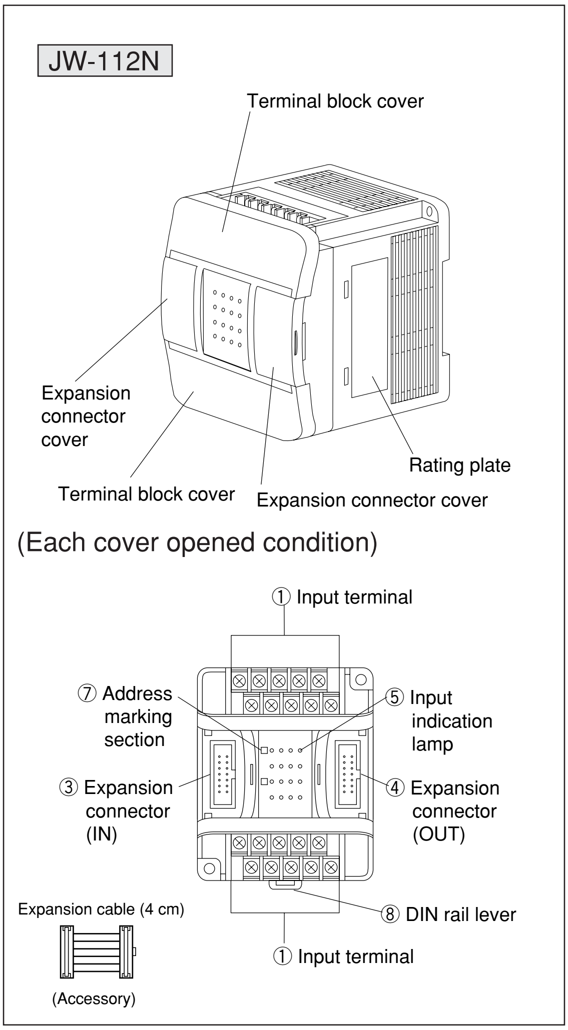
⑯ **ROM socket [JW-1424K/1442K/1624K/1642K only]**
Inserts a ROM (EPROM, EEPROM) for ROM operation.

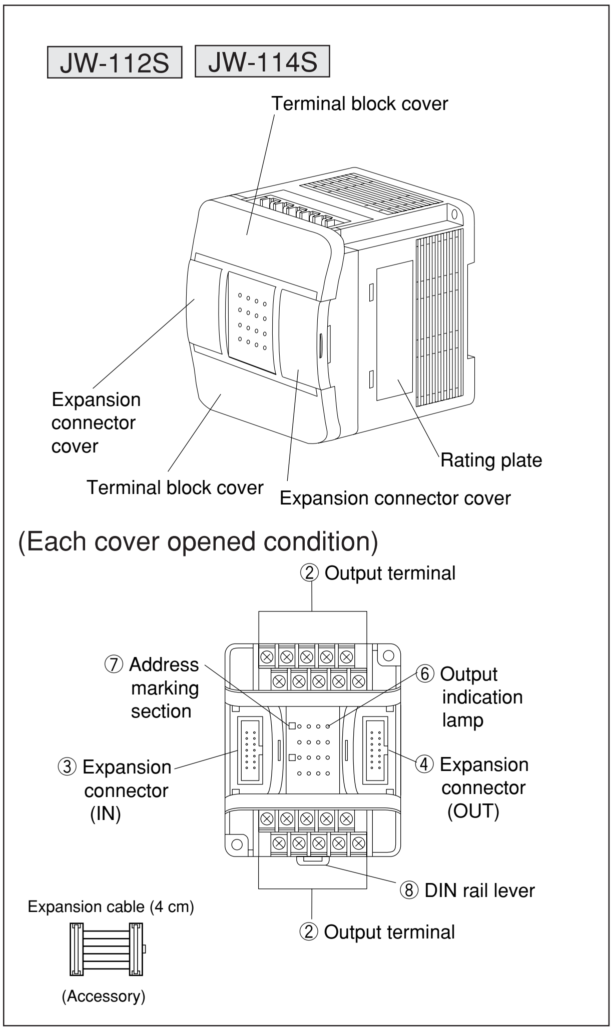
⑰ **Termination connector**
Makes sure to insert the termination connector on an expansion connector (OUT) of the end module.

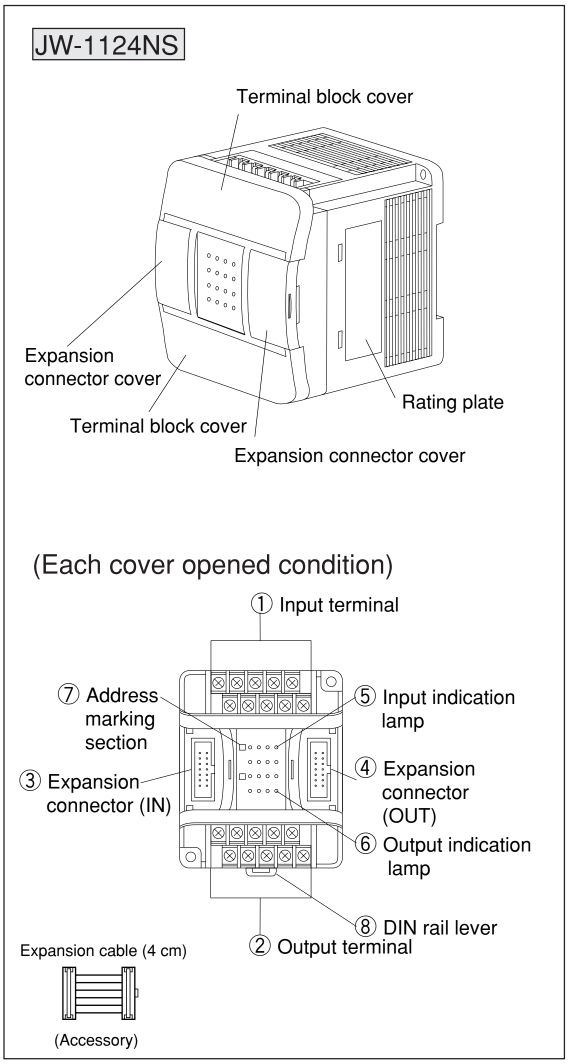
⑱ **DIN rail lever**
Moves up and down to install or detach the module on a DIN rail.

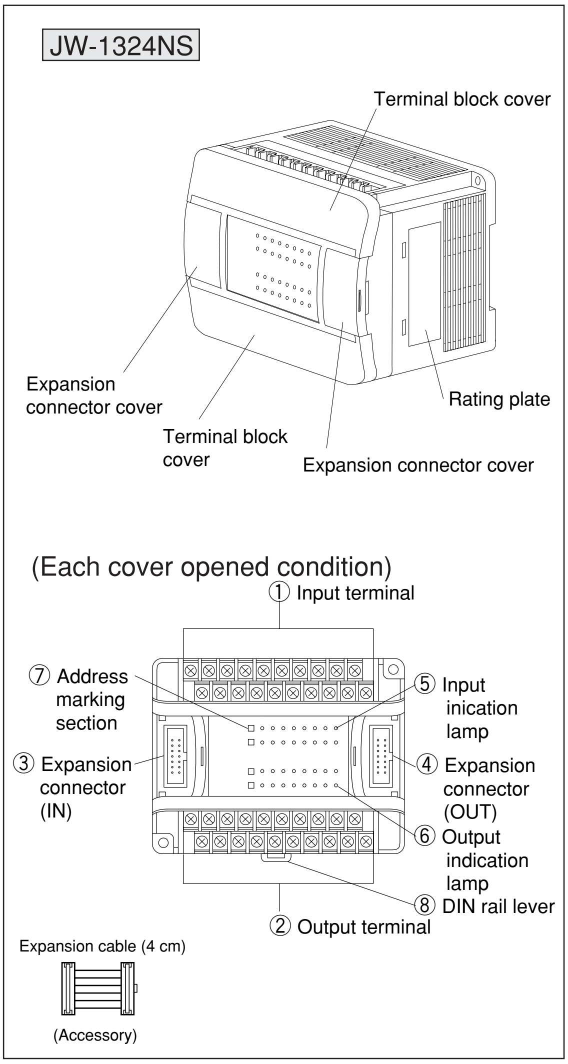
4-2 Expansion module

Five models: JW-112N, JW-112S, JW-114S, JW-1124NS, and JW-1324NS, are available as expansion module of JW10.









① **Input terminal**

Connects a cable from an input device.

② **Output terminal**

Connects a cable from an output device.

③ **Expansion connector (IN)**

Inserts an expansion cable (4 cm) supplied with an expansion module or JW-104EC (40 cm) for connection with an expansion connector of basic module. When this expansion module is used as the second expansion module, insert an expansion cable for connecting with the first expansion module's expansion connector (OUT).

④ **Expansion connector (OUT)**

Inserts the expansion cable (4 cm) supplied with an expansion module or JW-104EC (40 cm) for connection with the first expansion module (OUT) and second expansion module (IN).

When only one expansion module is used, insert a termination connector which is inserted in the expansion connector (OUT) of basic module (JW-1424K/1442K/1624K/1642K).

⑤ **Input indication lamp (amber)**

Lights when the connected input devices are functioning.

(JW-112N/1324NS: 16 pieces, JW-1124NS: 8 pieces)

⑥ **Output indication lamp (amber)**

Lights when the outputs are functioning.

(JW-112S/114S/1324NS: 16 pieces, JW-1124NS: 8 pieces)

⑦ **Address marking section**

Writes the 2nd and 3rd digits of input relay and output relay using an oil based ink.

Ex.: Write "43" for top address "00430."

⑧ **DIN rail lever**

Moves up and down to install or detach the module on a DIN rail.

Chapter 5. Installation

5-1 Setup environment

Avoid keeping the JW10 in the following conditions:

① Ambient temperature exceeds the range of 0 to 55˚C.

② The relative humidity exceeds the range of 5 to 90%.

③ Much dusts, salty and iron powders conditions.

④ Direct sunlight

⑤ Strong vibration and shock may be received.

⑥ Location where corrosive, combustible, or flammable gases are generated.

⑦ Location where organic solvent such as benzene, thinner, or alcohol, or strong base such as ammonia, or caustic soda may be adhered, and these atmospheric conditions.

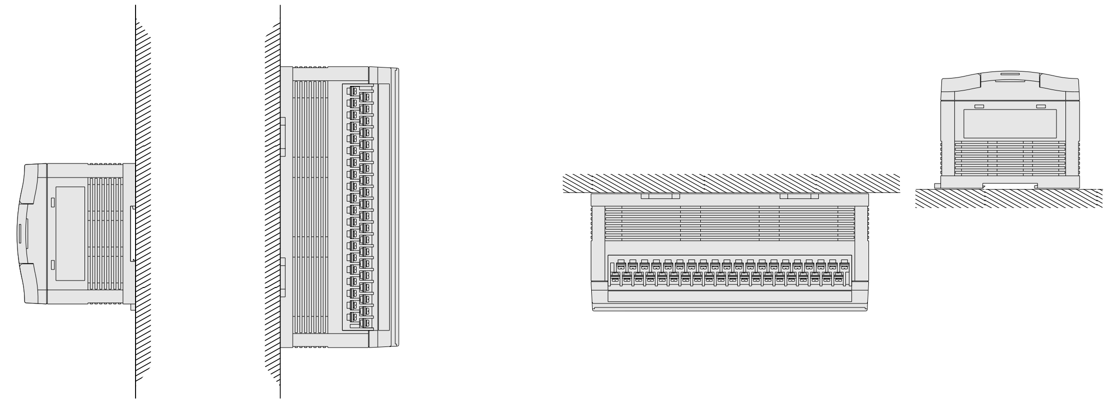
⑧ Location which is not well ventilated, and does not provide enough space from surrounding objects or heating devices.

5-2 Installation of basic module/expansion module

Install the module as far away as possible from high-voltage wires, high-voltage devices, power lines, power devices, radio equipment with transmitters, and other devices that generate large power surges.

Good : Vertical installation

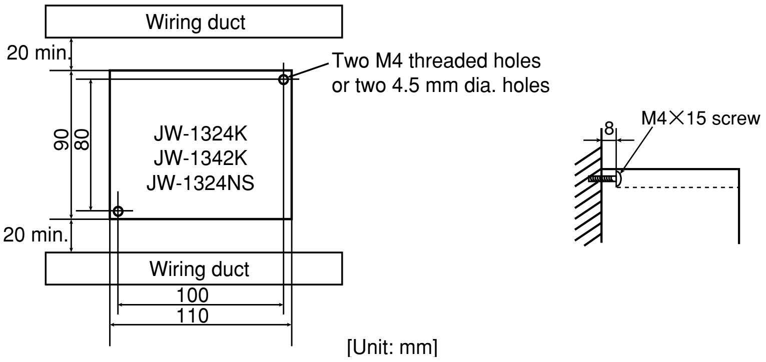
No good : Horizontal installation



[1] Direct installation

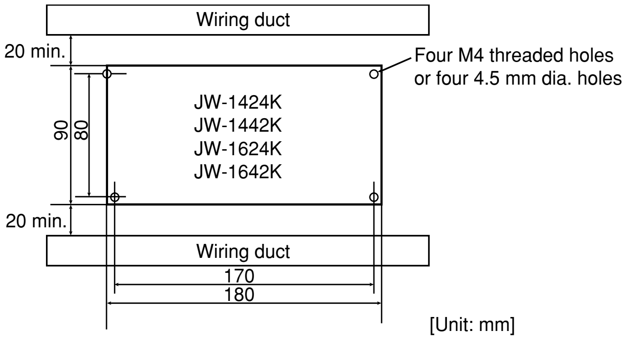
When the module is installed directly to a panel surface, use M4 screws with a length of 15 mm or more to secure the module.

JW-1324K, JW-1342K,
JW-1324NS

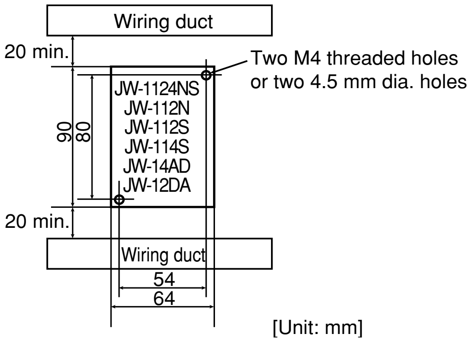


[Unit: mm]

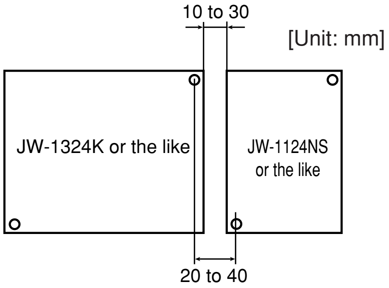
JW-1424K, JW-1442K
JW-1624K, JW-1642K



JW-1124NS, JW-112N,
JW-112S, JW-114S, JW-14AD, JW-12DA



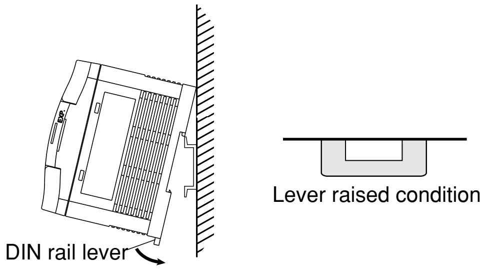
Installation dimensions between modules [using the cable (4 cm) supplied with the expansion module]



[2] Installation using DIN rail

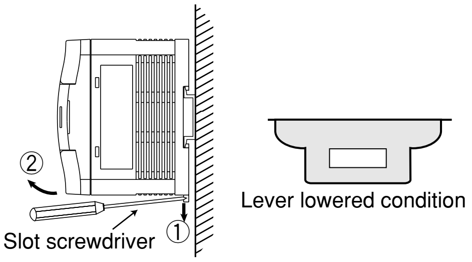
Use a DIN rail having rail width 35 mm and fix using the end plates.

Installation



Fix the grooved section of module at the rear side on the DIN rail, and press down in the direction of the arrow.
After the installation, push the lever of the DIN rail upward.

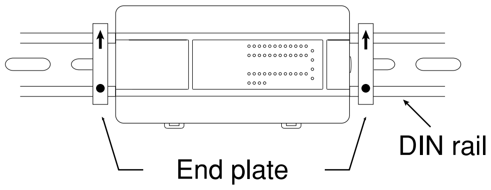
Removal

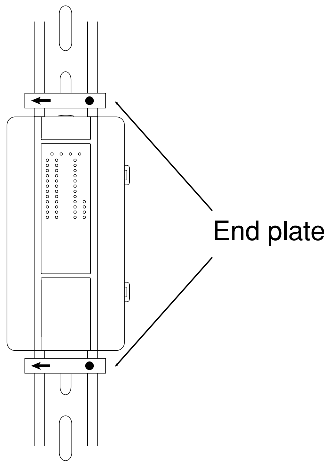


Lower the DIN rail and lever's groove using a slot screwdriver, and then hold up entire module to remove.

Make sure to use the end plate to fix the module.

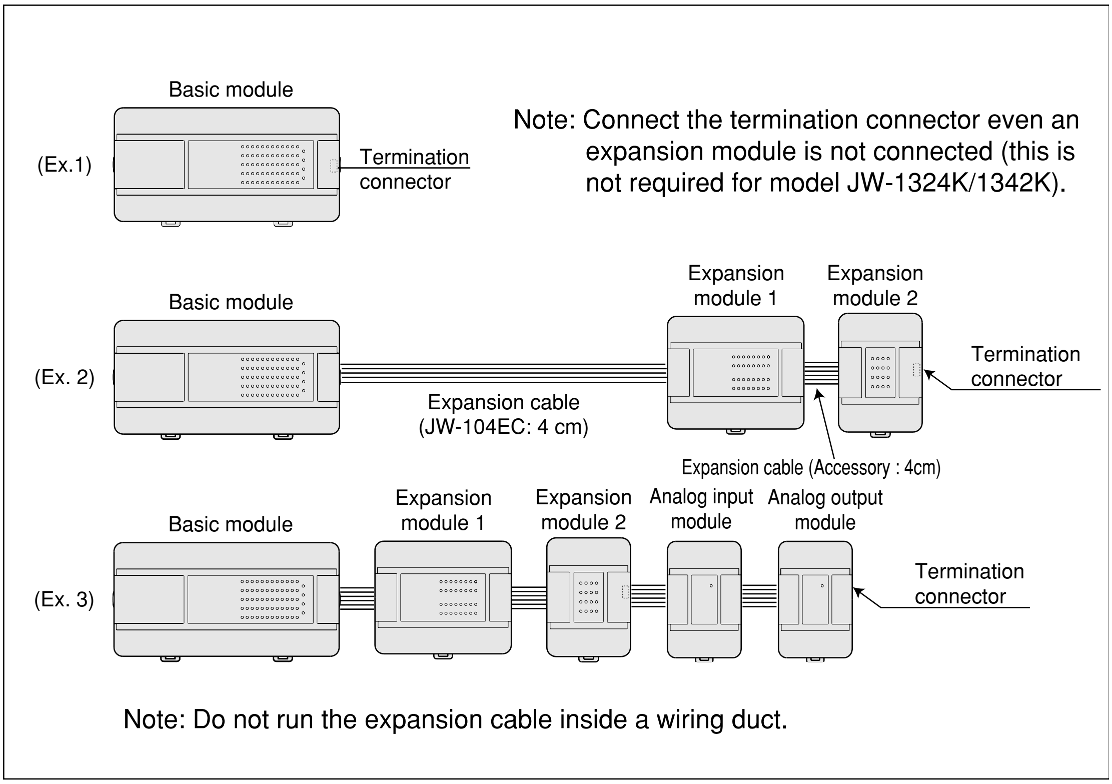
Fixing





5-3 Installation of expansion cable and termination connector

- Basic module JW-1424K, JW-1442K, JW-1624K, and JW-1642K can be connected two expansion modules at maximum, one analog input module JW-14AD, and analog output module JW-12DA.
- Use the cable (4 cm) supplied with the expansion/analog input/analog output module or JW-104EC (40 cm) for connection to the expansion cable.
- JW-104EC can be used only one for one system. (Ex. 2)
- Install a termination connector on the end module's expansion connector (OUT).



Note: Connect the termination connector even an expansion module is not connected (this is not required for model JW-1324K/1342K).

Note: Do not run the expansion cable inside a wiring duct.

■ Installation procedure of the expansion cable

① Open the expansion connector cover.

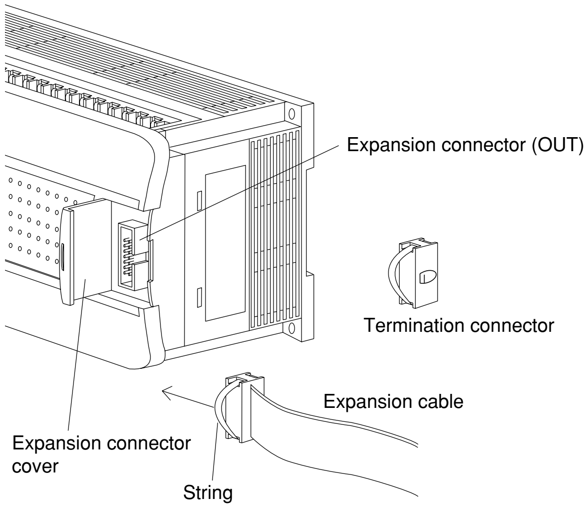
② Remove the termination connector which is inserted in the expansion connector (OUT) of the basic module (JW-1424K/1442K/1624K/1642K).

③ Insert the expansion cable in the expansion connector (OUT).

④ Insert the expansion cable in the expansion connector (IN) of the expansion module.

⑤ Insert the termination connector, which was removed in item ② above, in the expansion connector (OUT) of the end expansion module.

⑥ Close the expansion connector cover.



Note 1: To remove the expansion cable or termination connector, pull the strings of each connector.

Note 2: Do not connect between IN sides of the expansion connectors, or between OUT sides of the expansion connectors.

Chapter 6: Wiring

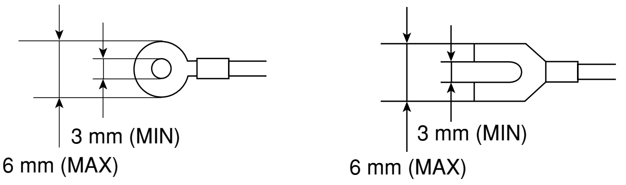
6-1 Precautions for wiring

(1) Separate signal lines and power line of the module from high voltage lines and power lines as far as possible. Do not run signal lines and power lines in parallel with high voltage or power lines.

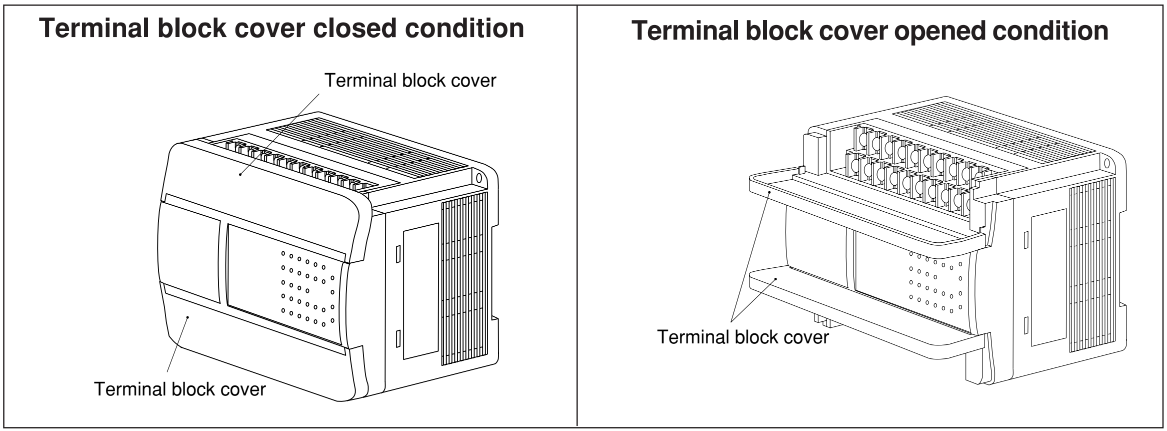
(2) Provide easy-to-detach wiring thoroughly considering operability at maintenance and repair.

(3) M3 terminal screws are applied for either of power terminals, input terminals, and output terminals. Use crimp-style terminals equivalent to JIS standard 1.25-3, and securely fix with tightening torque 4 to 8 kgf-cm.

[Recommended crimp-style terminals]
JAPAN SOLDERLESS TERMINAL MFG. CO., LTD
Model: 1.25-B3A 1.25-C3A
1.25-3 1.25-MS3



(4) Execute wiring work while opening the terminal block cover. Close this cover after the wiring work.


Terminal block cover closed condition

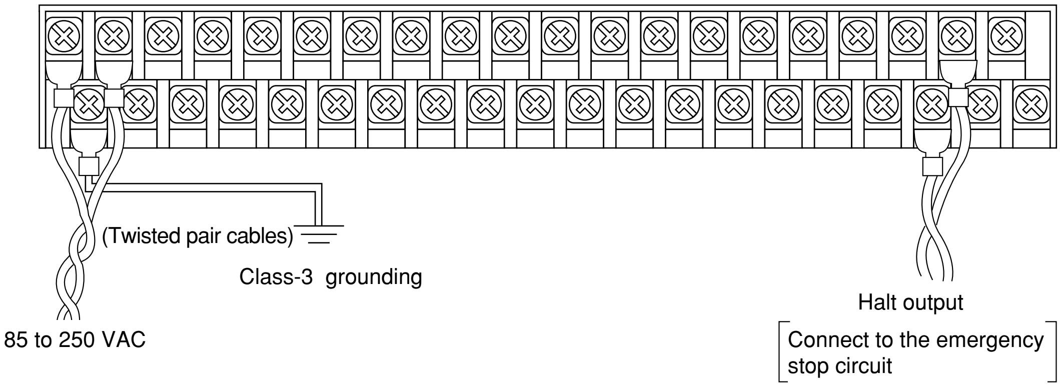
Terminal block cover opened condition

6-2 Wiring to power, FG terminals, and halt output terminals

- Use twisted-pair lines of 1.25 mm^2 sectional area for power lines.
- Connect the halt output to the emergency stop circuit. (See "10-2 Cautions on system design")

Terminal block's lower row of JW-1624K

AC-L	AC-N	COM.B	COM.C	COM.D	COM.E	405	407	410	412	COM.G	415	417	420	422	424	426	COM.I	HLT-C	L2	
	FG	400	401	402	403	404	406	COM.F	411	413	414	416	COM.H	421	423	425	427	HLT	L1	SHLD



Note 1: JW-1324K/1342K does not have halt output.

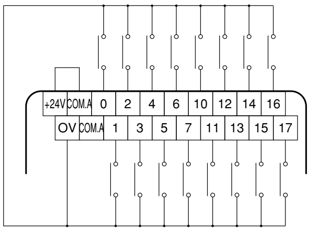
6-3 Wiring to input terminal

- Separate input lines from power lines, output lines, and power lines.
- Use cables of 0.75 to 1.25 mm^2 sectional area and connect using crimp-style terminals.

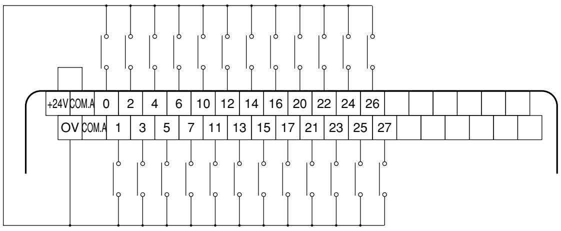
[1] Basic module

- Upper row of the terminal block of each module is input side.

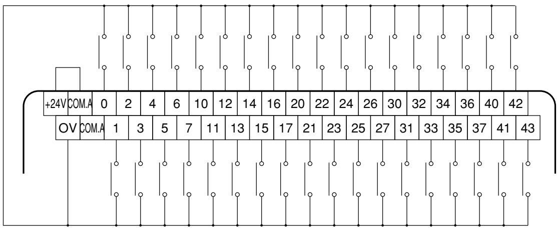
(1) JW-1324K/1342K ［DC input : 16 points］

		Terminal 0 to 3	Terminal 4 to 17	Terminal connection diagram
No. of input		4 points	12 points	
Rated input voltage		12/24 VDC	24 VDC	
Input voltage range		10 to 26.4 VDC	20 to 26.4 VDC	
Rated input current		3.6 mA TYP. (12 V) 7.6 mA TYP. (24 V)	4.8 mA TYP. (24 V)	
Input ON level		10 V (3 mA) max.	20 V (3.5 mA) max.	
Input OFF level		5 V (1.5 mA) min.	8 V (1.5 mA) min.	
Response time	OFF→ON	1 ms max.	10 ms max.(JW-1324K) 1 ms max.(JW-1342K)	
	ON→OFF	1 ms max.	10 ms max.(JW-1324K) 1 ms max.(JW-1342K)	
Common system		1 common line for 16 points (no polarity)		
Power capacity of 24 VDC for the integrated sensor.		24 VDC ± 10%, 300mA		

(2) JW-1424K/1442K ［DC input : 24 points］

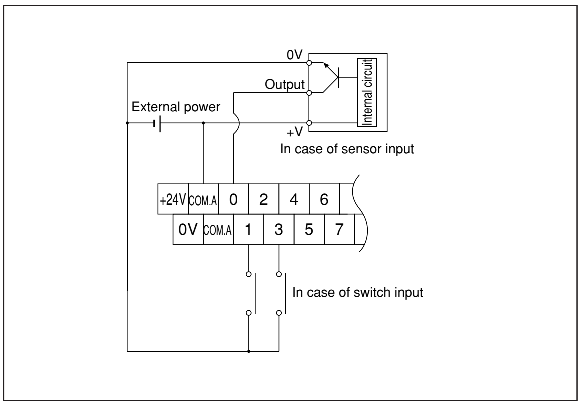
		Terminal 0 to 3	Terminal 4 to 27	Terminal connection diagram
No. of input		4 points	20 points	
Rated input voltage		12/24 VDC	24 VDC	
Input voltage range		10 to 26.4 VDC	20 to 26.4 VDC	
Rated input current		3.6 mA TYP. (12 V) 7.6 mA TYP. (24 V)	4.8 mA TYP. (24 V)	
Input ON level		10 V (3 mA) max.	20 V (3.5 mA) max.	
Input OFF level		5 V (1.5 mA) min.	8 V (1.5 mA) min.	
Response time	OFF→ON	1 ms max.	10 ms max.(JW-1424K) 1 ms max.(JW-1442K)	
	ON→OFF	1 ms max.	10 ms max.(JW-1424K) 1 ms max.(JW-1442K)	
Common system		1 common line for 24 points (no polarity)		
Power capacity of 24 VDC for the integrated sensor.		24 VDC ± 10%, 400mA		

(3) JW-1624K/1642K ［DC input : 36 points］

		Terminal 0 to 3	Terminal 4 to 43	Terminal connection diagram
No. of input		4 points	32 points	
Rated input voltage		12/24 VDC	24 VDC	
Input voltage range		10 to 26.4 VDC	20 to 26.4 VDC	
Rated input current		3.6 mA TYP. (12 V) 7.6 mA TYP. (24 V)	4.8 mA TYP. (24 V)	
Input ON level		10 V (3 mA) max.	20 V (3.5 mA) max.	
Input OFF level		5 V (1.5 mA) min.	8 V (1.5 mA) min.	
Response time	OFF→ON	1 ms max.	10 ms max.(JW-1624K) 1 ms max.(JW-1642K)	
	ON→OFF	1 ms max.	10 ms max.(JW-1624K) 1 ms max.(JW-1642K)	
Common system		1 common line for 36 points (no polarity)		
Power capacity of 24 VDC for the integrated sensor.		24 VDC ± 10%, 400mA		

■ In case of using an external power supply

- If 24 VDC power for the integrated sensor is insufficient capacity, supply power from outside.

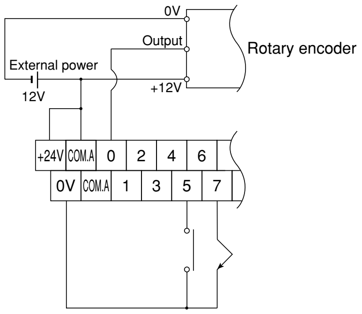


Basic module	Power capacity of 24 VDC for integrated sensor
JW-1324K JW-1342K	300 mA
JW-1424K JW-1442K JW-1624K JW-1642K	400 mA

■ In case of using 12 V power supply

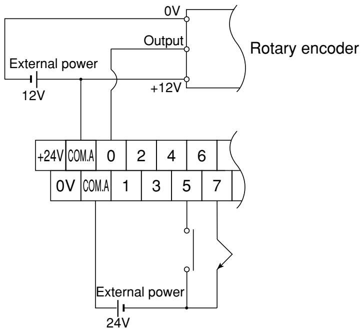
- Rated input voltage of input terminals 0 to 3 is 12/24 VDC. (Available for the high-speed counter input)
- In case of providing two power supplies (12 VDC and 24 VDC) separately, wire as shown below.

In case of common use with 24 VDC power for the integrated sensor



Note: In this case, the external 12 VDC power supply and the 24 VDC power for the integrated sensor are common for their positive (+) line side. Therefore, make sure to separate them at their negative (-) line side.

In case of common use with an external 24 VDC power



Note: In this case, the external 12 VDC power supply and the external 24 VDC power are common for their positive (+) line side. Therefore, make sure to separate them at their negative (-) line side.

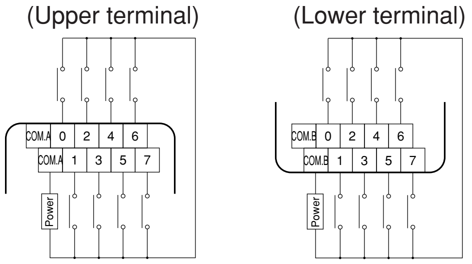
[2] Expansion module

- In JW-1124NS and JW-1324NS, upper row of the terminal block is input side. In JW-112N, upper and lower rows of the terminal block are input side.

(1) JW-112N ［DC input : 16 points］

		Terminal 0 to 7 × 2
No. of input		16 points
Rated input voltage		24 VDC
Input voltage range		20 to 26.4 VDC
Rated input current		4.8 mA TYP. (24 V)
Input ON level		20 V (3.5 mA) max.
Input OFF level		8 V (1.5 mA) min.
Response time	OFF→ON	10 ms max.
	ON→OFF	10 ms max.
Common system		1 common line for 8 points × 2

Terminal connection diagram

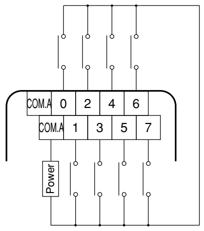


Relay numbers are allocated from 0 to 16 from upper to lower rows of the terminal block.

(2) JW-1124NS ［DC input : 8 points］

		Terminal 0 to 7
No. of input		8 points
Rated input voltage		24 VDC
Input voltage range		20 to 26.4 VDC
Rated input current		4.8 mA TYP. (24 V)
Input ON level		20 V (3.5 mA) max.
Input OFF level		8 V (1.5 mA) min.
Response time	OFF→ON	10 ms max.
	ON→OFF	10 ms max.
Common system		1 common line for 8 points

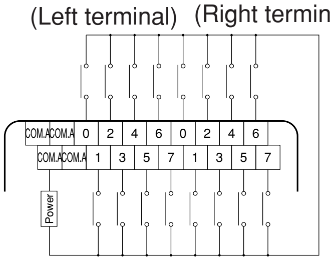
Terminal connection diagram



(3) JW-1324NS ［DC input : 16 points］

		Terminal 0 to 7 × 2
No. of input		16 points
Rated input voltage		24 VDC
Input voltage range		20 to 26.4 VDC
Rated input current		4.8 mA TYP. (24 V)
Input ON level		20 V (3.5 mA) max.
Input OFF level		8 V (1.5 mA) min.
Response time	OFF→ON	10 ms max.
	ON→OFF	10 ms max.
Common system		1 common line for 16 points

Terminal connection diagram



Relay numbers are allocated from 0 to 16 from left to right points of the terminal block.

[3] Precautions for connecting input device

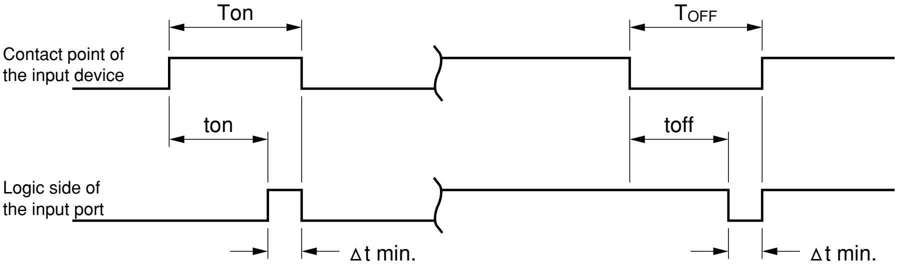
(1) ON/OFF time of the input signal

- In order to ensure ON/OFF condition of the input device correctly (limit switch etc.) on the operation of the JW10, ON or OFF time should meet the following conditions.

ON time of the input device (T_{ON}) Ton > Δt + ton

OFF time of the input device (T_{OFF}) Toff > Δt + toff

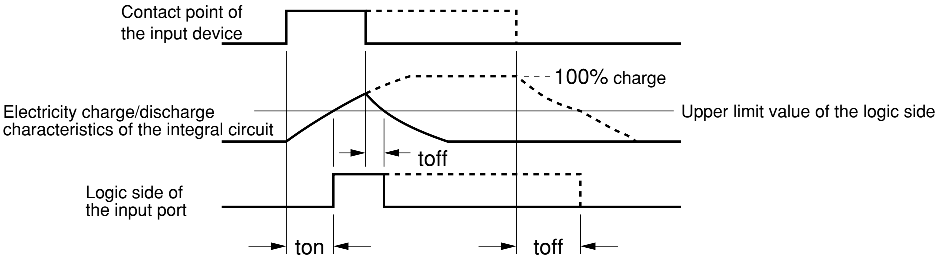
Δt	1 scanning time of PC
ton	OFF to ON response time of the input port
toff	ON to OFF response time of the input port



Ton
T_OFF
Contact point of the input device
ton
toff
Logic side of the input port
Δt min.
Δt min.

- In the input/output process at the beginning of each scanning cycle, ON/OFF state of the logic side of the input port is written in the data memory and used as input data for operation of the user's program within its scanning cycle. Therefore, if ON or OFF time of the logic side of the input port is less than 1 scanning time (Δt), ON/OFF data may not be included in the data memory.

Note 1: The response time of the input port is made by the electricity charge/discharge characteristics of the integral circuit of the input port, and it varies depending on the time of duration of ON or OFF.



Contact point of the input device
100% charge
Electricity charge/discharge characteristics of the integral circuit
Upper limit value of the logic side
toff
Logic side of the input port
ton
toff

- toff shows the difference, shown in the above, between the case when the ON time of the contact point of the input device is longer as shown by dotted lines and the case when the ON time is shorter as shown by solid line.

(Calculation example in case the JW-1324K is used as basic module)

If 1 scanning time is 5ms,

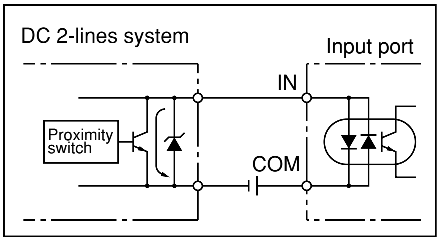
Ton > Δt + ton = 5 + 10 = 15 (ms)

Toff > Δt + toff = 5 + 10 = 15 (ms)

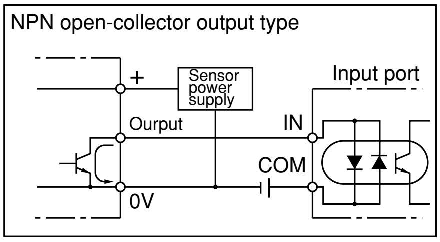
(2) Connectable input device

- The followings are sensors and switches which can be connected as input. See below for selection and connection of the input device.

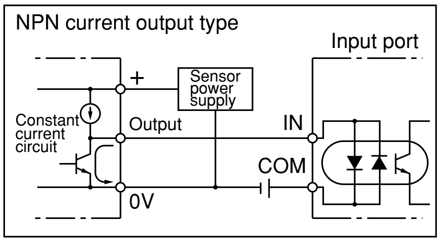
①



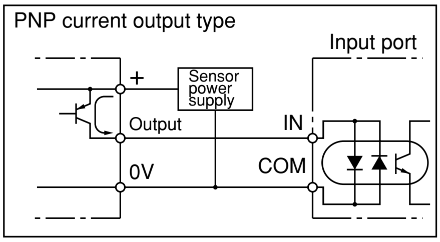
②



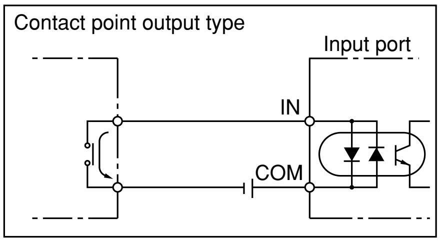
③



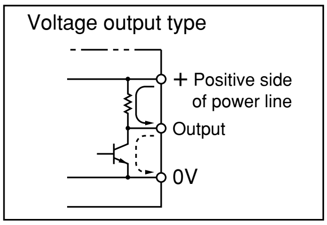
④



⑤



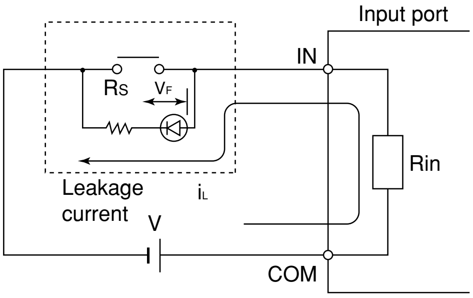
- Use a transistor having current driving capacity larger than that of the constant input current of the input port.
- In case of ①, pay attention to leakage current at OFF. (When leakage current is higher than the OFF input current level of the input port, the proximity switch may not turn OFF.)
- Be careful that voltage output type DC input device shown in the right may not be connected. (Driving capacity of an output transistor should be higher than the ON level of the input port).



(3) Countermeasure for leakage current on input device

In case of connecting with limit switch with LED or the like, there is also leakage current at OFF. If the leakage current is higher than the OFF level of the input port, the input port may not turn OFF, or noise margin at OFF state may drop.

Limit switch with LED



(Reference)

Calculation of leakage current i_L

$$i_L = \frac{V - V_F}{R_S + R_{in}}$$

V : Power supply voltage

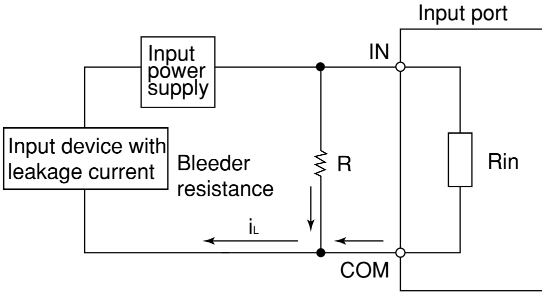
V_F : Voltage drop in the forward direction of LED

R_S : Current limit resistance

R_{in} : Input impedance of input port

Countermeasure **Connection of bleeder resistance**

As a countermeasure, a bleeder resistance can be inserted in the input side of the input port as shown below.



Choose the bleeder resistance value R to meet the following conditions:

$$i_L = \left(\frac{R_{in} \times R}{R_{in} + R}\right) < V_{in\ OFF}$$

Composite impedance of the bleeder resistance and the input impedance

$$R < \left(\frac{V_{in\ OFF} \times R_{in}}{R_{in} \times i_L - V_{in\ OFF}}\right) \times \underbrace{0.5}_{\text{Margin}}$$

i_L : Current leakage of the input device

$V_{in\ OFF}$: Input of the input port OFF level voltage

R_{in} : Input impedance of the input port

V : Input power supply voltage

In this case, the rating electric power W is,

$$W > \frac{V^2}{R} \times \underbrace{3}_{\text{Margin}}$$

[Example] In case that the JW-1324K is used as a basic module at the input power supply voltage of 24 V, and that the leakage current of the input device is 6 mA,

$i_L = 6$ mA

$V_{in\ OFF} = 8$ V

$R_{in} = 5$ kohm

$V = 24$ V

$$R < \frac{8 \times 5}{5 \times 6 - 8} \times 0.5 = 0.909 \text{ kohm}$$

If R is 0.9 kohm

$$W > \frac{24^2}{0.9 \times 10^3} \times 3 = 1.92 \text{ W}$$

W will be 2 W.

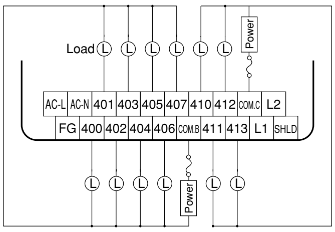
6-4 Wiring to output terminal

- Use cables of 0.75 to 1.25 mm^2 sectional area and connect using crimp-style terminals.
- Make sure to insert fuses which match each load in order to secure safety for every output circuit for common.

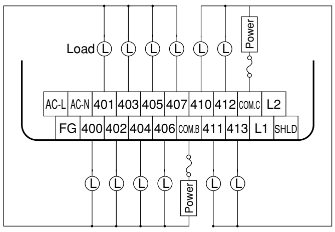
[1] Basic module

- Lower row of terminal block of each module is output side.

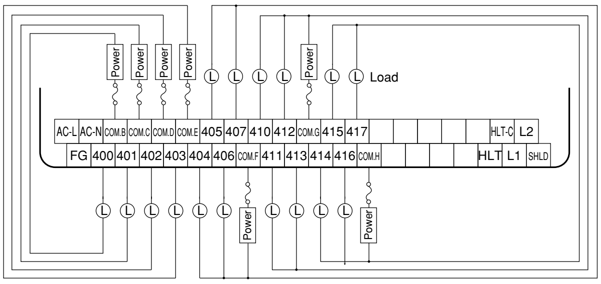
(1) JW-1324K ［Relay output : 12 points］

		Terminal 400 to 413	Terminal connection diagram
No. of output		12 points	 Load L Power AC-L AC-N 401 403 405 407 410 412 COM.C L2 FG 400 402 404 406 COM.B 411 413 L1 SHLD
Max. open-close voltage and current		250 VAC/30 VDC 2 A/point 2 A/common	
Min. load		5 VDC 10 mA	
Response time	OFF→ON	10 ms max.	
	ON→OFF	10 ms max.	
Common system		1 common line for 8 points ×1 (400 to 407) 1 common line for 4 points × 1 (404 to 413)	

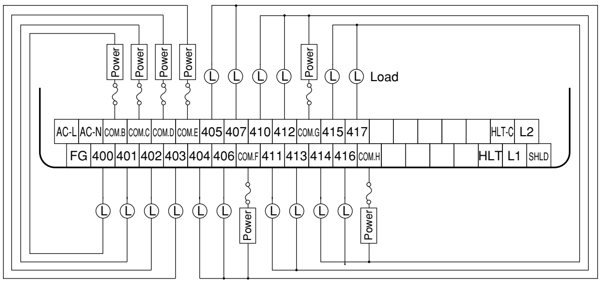
(2) JW-1342K ［Transistor output : 12 points］

		Terminal 400 to 413	Terminal connection diagram
No. of output		12 points	 Load L Power AC-L AC-N 401 403 405 407 410 412 COM.C L2 FG 400 402 404 406 COM.B 411 413 L1 SHLD
Rated load voltage		5/12/24 VDC	
Load voltage range		4.5 to 27 VDC	
Rated max. load current		0.3A/point 1.6A/8 points common(400 to 407) 0.8A/4 points common(410 to 413)	
Response time	OFF→ON	1 ms max.(resistance load)	
	ON→OFF	1 ms max.(resistance load)	
Common system		1 common line for 8 points ×1 (400 to 407) 1 common line for 4 points × 1 (404 to 413)	

(3) JW-1424K ［Relay output : 16 points］

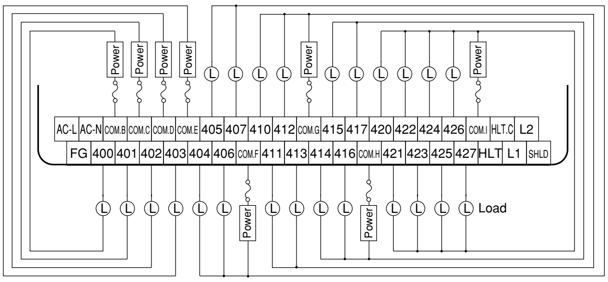
		Terminal 400 to 417	Terminal connection diagram
No. of output		16 points	 Power L Load AC-L AC-N COM.B COM.C COM.D COM.E 405 407 410 412 COM.G 415 417 HLT-C L2 FG 400 401 402 403 404 406 COM.F 411 413 414 416 COM.H HLT L1 SHLD
Max. open-close voltage and current		250 VAC/30 VDC 2 A/point 2 A/common	
Min. load		5 VDC 10 mA	
Response time	OFF→ON	10 ms max.	
	ON→OFF	10 ms max.	
Common system		1 common line for 1 point × 4 (400 to 403) 1 common line for 4 points × 3 (404 to 417)	

(4) JW-1442K ［Transistor output : 16 points］

		Terminal 400 to 417	Terminal connection diagram
No. of output		16 points	
Rated load voltage		5/12/24 VDC	
Load voltage range		4.5 to 27 VDC	
Rated max. load current		0.3A/point 0.8A/4 points common(404 to 417)	
Response time	OFF→ON	1 ms max.(resistance load)	
	ON→OFF	1 ms max.(resistance load)	
Common system		1 common line for 1 point ×4 (400 to 403) 1 common line for 4 points × 3 (404 to 417)	

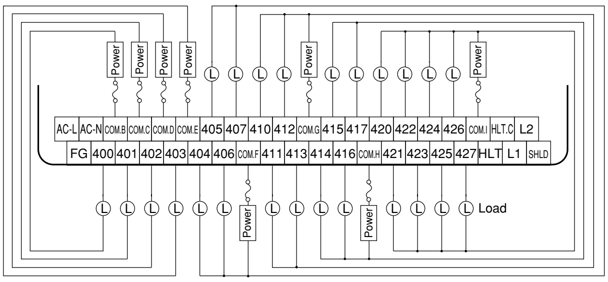
Power
L
Load
AC-L AC-N COM.B COM.C COM.D COM.E 405 407 410 412 COM.G 415 417 HLT-C L2
FG 400 401 402 403 404 406 COM.F 411 413 414 416 COM.H HLT L1 SHLD

(5) JW-1624K ［Relay output : 24 points］

		Terminal 400 to 427	Terminal connection diagram
No. of output		24 points	
Max. open-close voltage and current		250 VAC/30 VDC 2 A/point 2 A/common	
Min. load		5 VDC 10 mA	
Response time	OFF→ON	10 ms max.	
	ON→OFF	10 ms max.	
Common system		1 common line for 1 point × 4 (400 to 403) 1 common line for 4 points × 3 (404 to 417) 1 common line for 8 points × 1 (420 to 427)	

Power
L
Load
AC-L AC-N COM.B COM.C COM.D COM.E 405 407 410 412 COM.G 415 417 420 422 424 426 COM.I HLT.C L2
FG 400 401 402 403 404 406 COM.F 411 413 414 416 COM.H 421 423 425 427 HLT L1 SHLD

(6) JW-1642K ［Transistor output : 24 points］

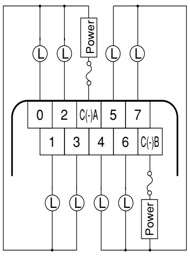
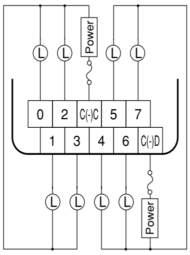
		Terminal 400 to 427	Terminal connection diagram
No. of output		24 points	
Rated load voltage		5/12/24 VDC	
Load voltage range		4.5 to 27 VDC	
Rated max. load current		0.3A/point 0.8A/4 points common(404 to 417) 1.6A/8 points common(420 to 427)	
Response time	OFF→ON	1 ms max.(resistance load)	
	ON→OFF	1 ms max.(resistance load)	
Common system		1 common line for 1 point ×4 (400 to 403) 1 common line for 4 points × 3 (404 to 417) 1 common line for 8 points × 1 (420 to 427)	

Power
L
Load
AC-L AC-N COM.B COM.C COM.D COM.E 405 407 410 412 COM.G 415 417 420 422 424 426 COM.I HLT.C L2
FG 400 401 402 403 404 406 COM.F 411 413 414 416 COM.H 421 423 425 427 HLT L1 SHLD

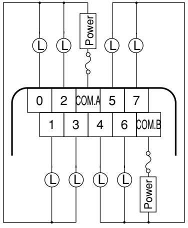
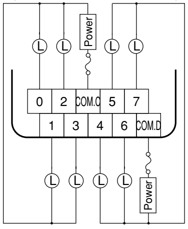
[2] Expansion module

- In JW-1124NS and JW-1324NS, lower row of the terminal block is output side. In JW-112S and JW-114S, upper and lower rows of the terminal block are output side.

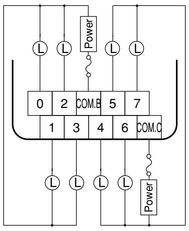
(1) JW-112S ［Transistor output : 16 points］

		Terminal 400 to 417	Terminal connection diagram
No. of output		16 points	(Upper terminal)  (Lower terminal) 
Rated load voltage		5/12/24 VDC	
Load voltage range		4.5 to 27 VDC	
Rated max. load current		0.3A/point 0.8A/4 points common	
Response time	OFF→ON	1 ms max.(resistance load)	
	ON→OFF	1 ms max.(resistance load)	
Common system		1 common line (-) for 4 points ×4	Relay numbers are allocated from 0 to 16 from upper to lower rows of the terminal block.

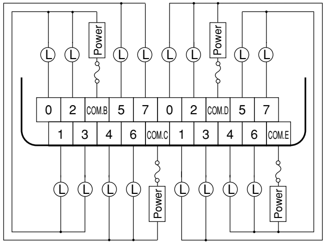
(2) JW-114S ［Relay output : 16 points］

		Terminal 0 to 7 × 2	Terminal connection diagram
No. of output		16 points	(Upper terminal)  (Lower terminal) 
Max. open-close voltage and current		250 VAC/30 VDC 2 A/point 2 A/common	
Min. load		5 VDC 10 mA	
Response time	OFF→ON	10 ms max.	
	ON→OFF	10 ms max.	
Common system		1 common line for 4 point × 4	Relay numbers are allocated from 0 to 16 from upper to lower rows of the terminal block.

(3) JW-1124NS ［Relay output : 8 points］

		Terminal 0 to 7	Terminal connection diagram
No. of output		8 points	
Max. open-close voltage and current		250 VAC/30 VDC 2 A/point 2 A/common	
Min. load		5 VDC 10 mA	
Response time	OFF→ON	10 ms max.	
	ON→OFF	10 ms max.	
Common system		1 common line for 4 point × 2	

(4) JW-1324NS ［Relay output : 16 points］

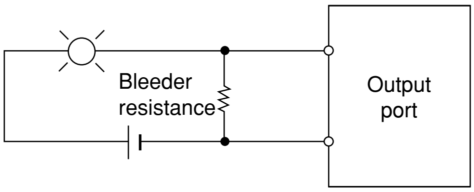
		Terminal 0 to 7 × 2	Terminal connection diagram
No. of output		16 points	(Left terminal) (Right terminal)  L Power 0 2 COM.B 5 7 0 2 COM.D 5 7 1 3 4 6 COM.C 1 3 4 6 COM.E
Max. open-close voltage and current		250 VAC/30 VDC 2 A/point 2 A/common	
Min. load		5 VDC 10 mA	
Response time	OFF→ON	10 ms max.	
	ON→OFF	10 ms max.	
Common system		1 common line for 4 point × 4	Relay numbers are allocated from 0 to 16 from left to right points of the terminal block.

[3] Precautions for connecting output device

(1) Countermeasure to surge current of lamp load

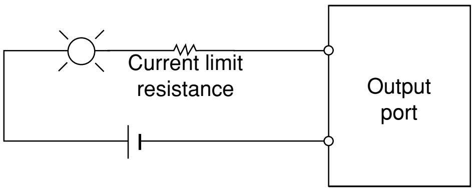
At turning on an incandescent lamp, there is surge current 10 to 20 times bigger than normal current for several 10 ms. For reduction of the surge current, insert either a bleeder resistance or an electric current limit resistance.

① To insert a bleeder resistance



During the OFF state of the output port, keep supplying dark current so small as to turn on the lamp dimly.

② To insert an electric current limit resistance



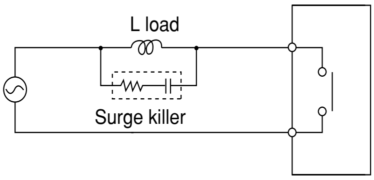
This limits the current within a value determined by the value of the current limit resistance. When the resistance value is high, the voltage on the lamp decreases. Determine the resistance value by the brightness needed when turning on the lamp.

(2) Countermeasure to surge voltage at opening/closing induced load

Some load generates surge voltage of several thousands volt when an induced load is opened or closed its circuit. As the module of relay output type does not have surge absorption circuit, surge countermeasure outside the module is indispensable in case the load generates high voltage. (This surge voltage countermeasure can extend the life time of the contact points of the relay.)
The transistor output module has an integrated internal circuit to absorb voltage surges.Howefver, if a load is located far from the relay output and a long wire is used to connect the load, the surge absorption circuit may not function well. In this case, you should provide a surge absorption circuit near the load.

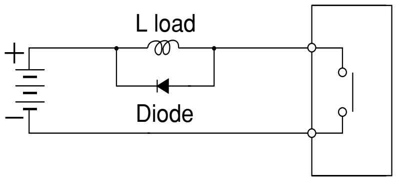
Surge voltage countermeasure

① Inserting CR surge killer

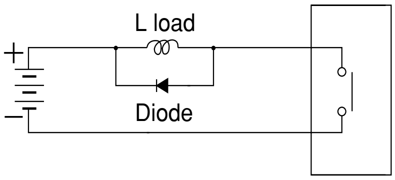


(Relay output)

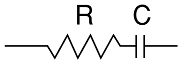
② Inserting a diode



(Relay output)



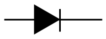
(Transistor output)

CR surge killer: 

C: 0.033 to 0.33 μF (Pressure resistance should be 250 VAC or higher)
R: 47 to 120 ohm

Example of CR surge killer

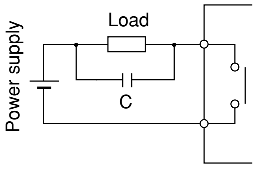
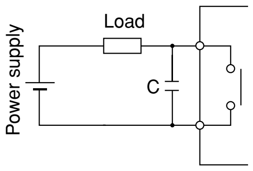
For 100 VAC	ECQ-J0186X(0.1 μ + 120 ohm) (made by Matsushita Electric Co., Ltd.)
For 200 VAC	ECQ-J0187X(0.033 μ + 120 ohm) (made by Matsushita Electric Co.,Ltd.)

Diode:  Peak inverse voltage (V_{RM}) should be more than 3 times of the load voltage.
Average rectified current (Io) should be more than load current.

In case of AC load, a varistor can be used in place of the CR surge killer for the same effect.
(Installation of both of the CR surge killer and the varistor increases the effect.)

For 85 to 132 VAC	TNR12G221K (made by Marcon Co., Ltd.), ERZV14D221 (made by Matsushita Electric Co., Ltd.)
For 170 to 264 VAC	TNR12G431K (made by Marcon Co., Ltd.), ERZV14D431 (made by Matsushita Electric Co., Ltd.)

Note 1: Avoid the use of a capacitor only as an arc killer, as shown below:

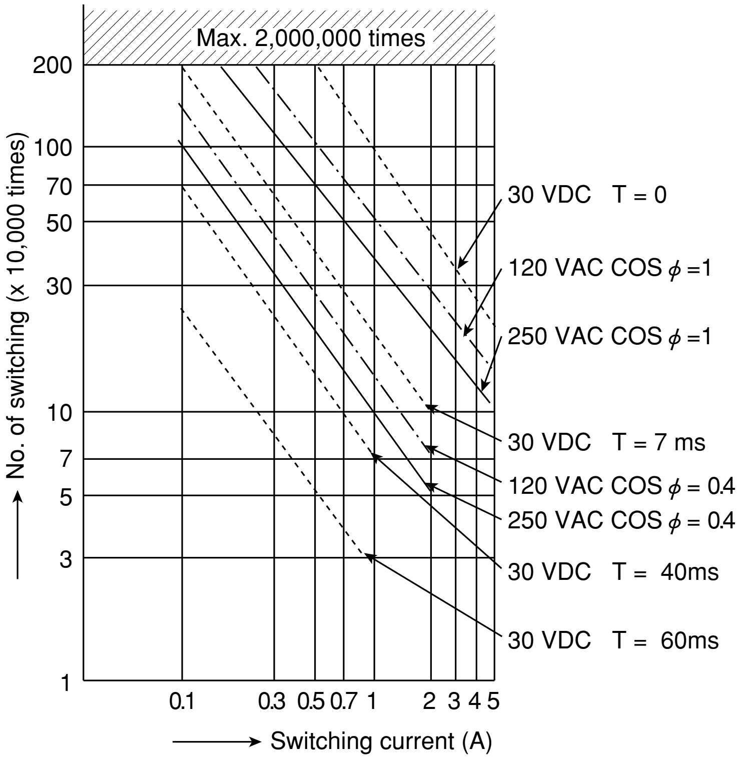
 Power supply Load C	Though a capacitor is very effective for the arc deletion at shut-off. But charged current to the capacitor may melt the contact point at turning ON a contact point.	 Power supply Load C	Though a capacitor is very effective for the arc deletion at shut-off. But at the opening of the circuit of a contact point, electricity is accumulated at the capacitor. Therefore, the short circuit current of the capacitor may melt the contact point at turning ON the contact.

(3) When driving miniature load

The relay used in the module of relay output type is suitable for power drive, and so with a low voltage and small current such as less than 10 mA, the contact reliability of the contact point drops. In such a case, we recommend to be used module of transistor output type.

(4) Life of relays

The life of relays used in output circuit, varies depending on the kind of loads. The following shows characteristic curves of the relay contact point.



Note 1: The above chart shows standard values.
Depending on the environment of usage (ambient temperature, etc.), different life may result.

Note 2: When the signal to the contact point is DC, the life of relays varies according to the load rise characteristics (time constant: T) of the load. The load rise characteristics of the load after the contact point is turned ON are determined by inductance: L and resistance: R

$$\left(T = \frac{R}{L} \right)$$

For the time constant of the load used, see below:

In case of resistance load: T < 1ms
In case of small size relay: T = 7 ms
In case of large current L load and magnet: T = 40 ms

In case of a L load with a diode for surge countermeasure, the life may be similar to the case of T < 1ms.

Note 3: Use the contact switching life of more than 100,000 times and within the switching current of less than 2A.

6-5 Wiring with noise countermeasures

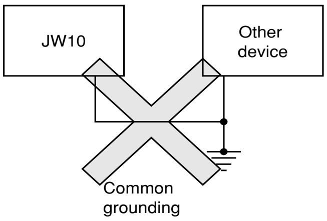
For your safe usage of the JW10, observe the "6-1 Precaution for wiring" carefully.
Wiring to prevent the JW10 from malfunction caused by noise is shown below. Besides, some malfunctions by noise come from complex causes or a cause which cannot be analyzed in quantity. Use the following noise countermeasures as your reference, when you take measures for each actual situation.

[1] Grounding

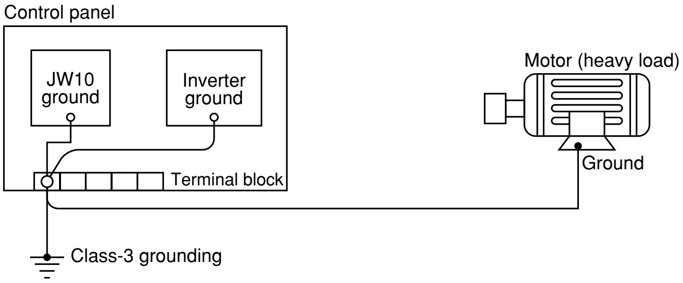
Grounding has two purposes; to protect operators from electric shock and to prevent malfunction by noise. The grounding for noise prevention is shown here.

Don't use a common ground for the JW10 and other device.

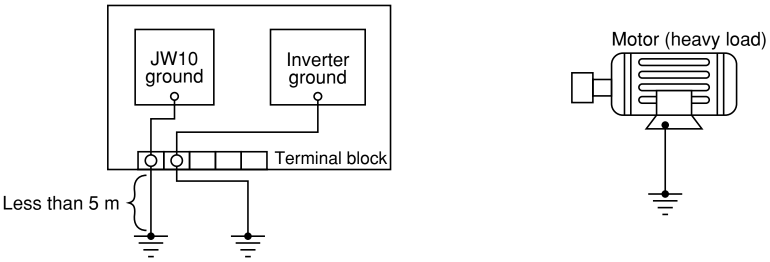
When the FG cable of the JW10 is also used for grounding for other device, noise might come into the JW10 from other device.



Bad example: Don't use the FG line of the JW10 for grounding of a motor or an inverter.



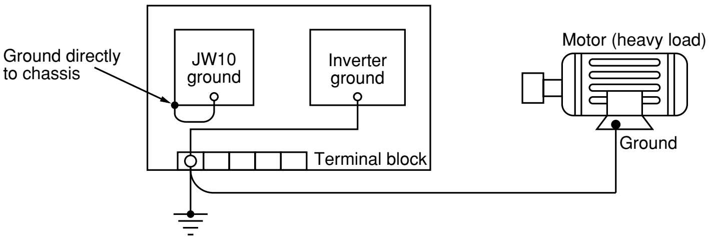
Countermeasure 1: Separate grounding for each of the JW10, the motor and the inverter.



Note 1: Use a twisted wire of over 2 mm^2 sectional area and less than 5 m long in grounding the JW10 for the noise prevention purpose.

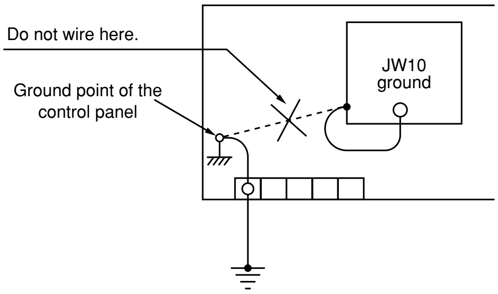
Countermeasure 2: When separate grounding is not available.

- If a separate grounding for the JW10 cannot be made, ground directly from the FG terminal of the JW10 to the chassis on which the JW10 is mounted.

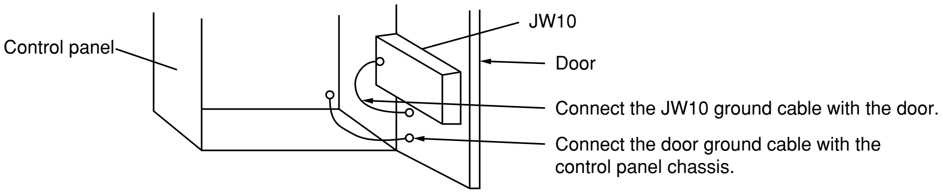


Note 1: Observe the following points for direct grounding of the FG cable of the JW10 to the chassis:

- Connect the grounding cable from the FG terminal of the JW10 to the chassis in the minimum distance.
- Where the control panel itself is grounded, do not wire between the grounding point and the FG terminal of the JW10.



Reference: Note for fitting the JW10 on the control panel door.
Ground from the FG terminal of the JW10 to the door.
Use a twisted wire of over 2 mm^2 sectional area for grounding cable of the control panel from its door (less than 50 cm.)

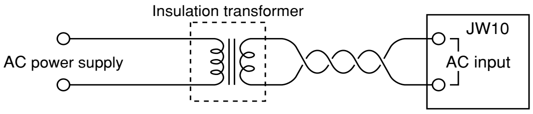

Control panel
JW10
Door
Connect the JW10 ground cable with the door.
Connect the door ground cable with the control panel chassis.

[2] Countermeasure of noise from power supply line

The AC power supply input noise resistance capacity of the JW10 is 1000 Vp-p. When any noise over this limit is possible to come through the power supply line, install an insulation transformer.

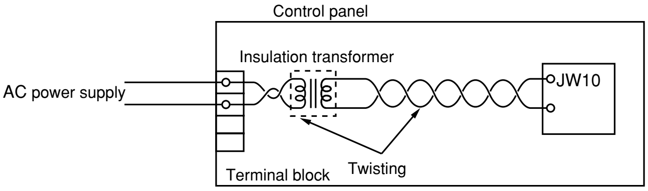
Countermeasure 1: Install an insulation transformer

Noise has a high frequency of 100 KHz to 2 MHz, which should be blocked by a transformer.



Note 1: When using an insulation transformer, note the following points:

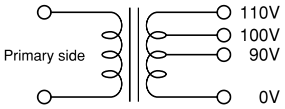
- An insulation transformer with static electricity shield can also prevent noise by static coupling.
- Install an insulation transformer near the power supply input of the control panel in order to block noise at the entrance of the control panel.



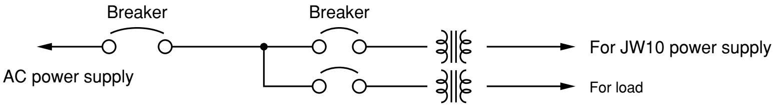
- Use two twisted wires in the primary and secondary sides of the transformer.
- Choose the insulation transformer of the capacity of more than 20% higher than that of the rated load. When a transformer of the same capacity as that of the rated load is used, a primary input voltage might exceed the transformer rated capacity and become dangerous state such as emitting smoke.

Basic module	Power consumption
JW-1324K/1342K	30 VA max.
JW-1424K/1442K	55 VA max.
JW-1624K/1642K	60 VA max.

- When a large-capacity transformer with higher voltage in the secondary side is chosen, we recommend to install a intermediate voltage tap.

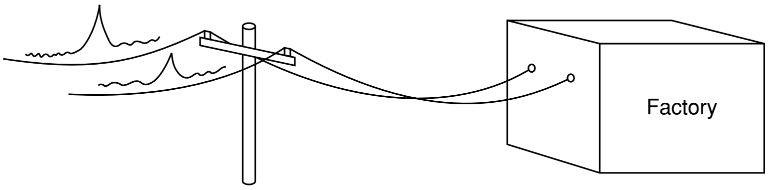


- With particularly large noise, several transformers can be installed, not only to the power supply input of the JW10 but also to the load.



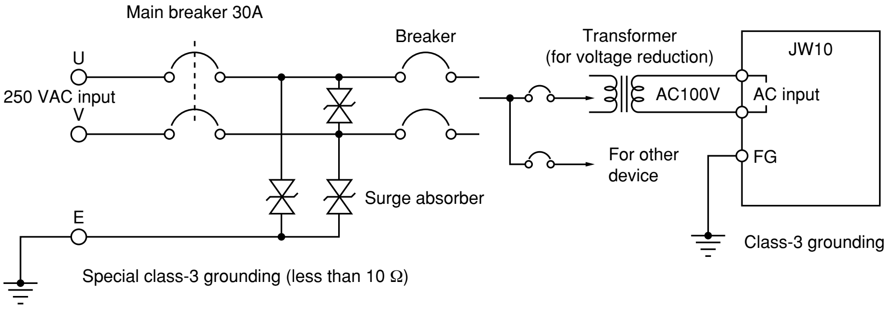
[3] Safeguard from lightning

Below are countermeasures in case when the factory facilities are located far from residential areas and that effects from induced lightning (induced voltage by lightning strikes) are expected. Note, however, that they are not the measures for direct strikes of lightning. In some cases, the voltage of the induced lightning may go beyond 4,000 KV. Therefore, the purpose of these countermeasures is just to minimize the damage on the device.



Countermeasure 1: Install a surge absorber on the receiver panel of commercial electric power as protection from induced lightning.

Different models should be chosen according to the facilities load and power supply voltage. For your reference, below is a wiring diagram of the outdoor type cubicle for 1.7 KVA.



Note 1: Note the following when wiring.

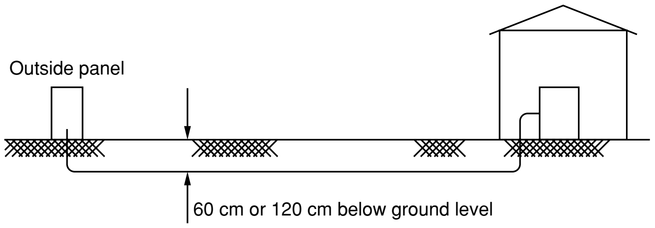
- The ground of the surge absorber is the special class-3 ground (less than 10 ohm ground resistance) and should be separated from the ground of the JW10. (class-3 grounding.)
- Install the main breaker before the surge absorber.
- The followings are known surge absorbers in the market. Different types for different power supply voltages.

Commercial power voltage	Model name	Specifications	Manufacturer
100 VAC	ERZ-A20PK251	Varistor voltage: 250 V ± 10% Surge resistant volume: 5,000 A (8/20 μs) Energy resistant volume: 90 Joule	Matsushita Electric Co.,Ltd.
200 VAC	ERZ-A20PK501	Varistor voltage: 500 V ± 10% Surge resistant volume: 5,000 A (8/20 μs) Energy resistant volume: 70 Joule	

- Use the ground wire of over 3.5 mm^2 section area for the surge absorber.

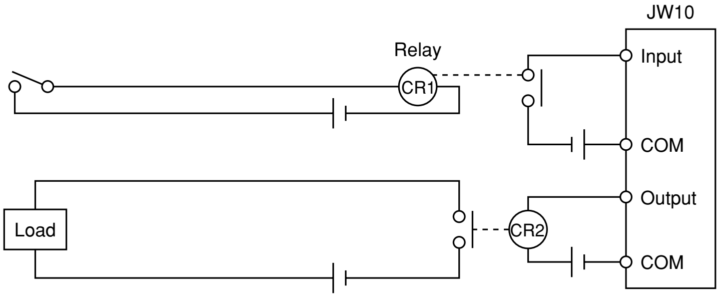
Countermeasure 2: Underground wiring as a countermeasure of lightning.
When communication cables and input cables of the JW10 go out of a building, place them underground. Provide junction for input/output signals using relays.

1) Underground cabling
In a lightning weather condition, the atmosphere is electrically charged and a wiring in the air induces a voltage of over 24 VDC. Therefore, place the wiring under the ground.



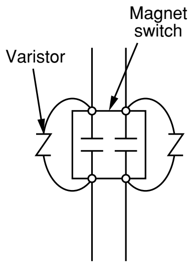
As for the depth of cable installation, refer to local regulations.

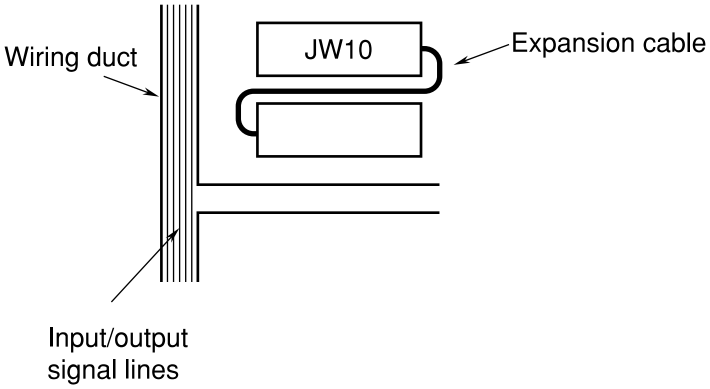
2) Relay connection for the input/output signals using relays.
The relay isolates the effects of lightning and minimizes the damage.



[4] Wire of expansion cable

When turning ON/OFF of the magnet switch installed near the JW10 and expansion cable, high noise and high voltage may occur to give bad effects on the operation of the JW10. Therefore, for prevention of the noise occurrence, insert a noise killer, such as a varistor, at the contact point of the magnet switch. Do not place the expansion cable, through which input/output signal lines and power lines are running.

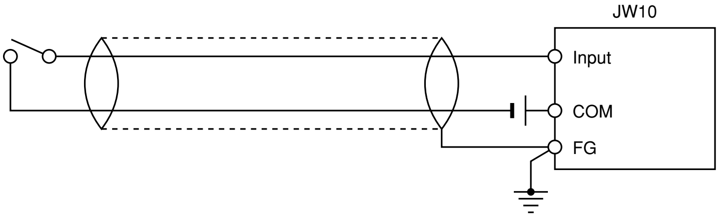




[5] Note for external wiring to I/O port

(1) Wiring to input port

When extending the external line to the input port for more than 100 m, use shielding wire. Even in case of less than 100 m extension, shielding wires should be used under certain conditions. Do not forget to connect the shield of the shielding wire with the FG terminal of the JW10.



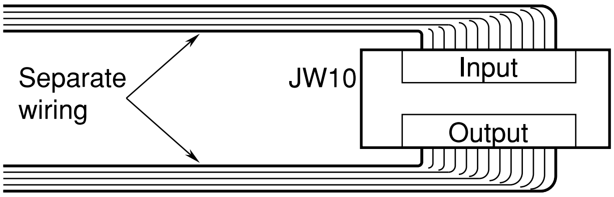
(2) Wiring to output port

Since the output port does not have a built-in surge absorbing circuit, do not forget to install a surge killer, such as a varistor, in the output side. Operation without a surge killer might give bad effects on other modules by spark noises from the relay. As for the surge killer, see 6・10, "Precaution for connecting output device."

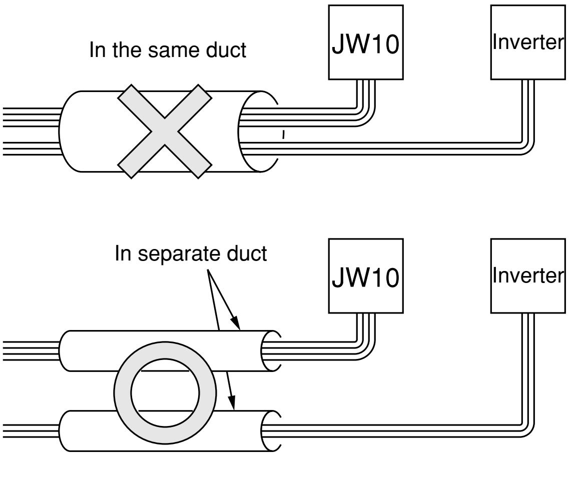
(3) Wiring with power line

Do not run the input signal, output signal and communication cables of the JW10 near and in parallel with the power line.

- When input and output signal cables are extended for long distance, make separate wiring for the input signal and the output signal of JW10.



- Make separate wiring for the input signal and the output signal of the JW10 from the power line. Particularly with the power line for the inverter and the servo driver, do not place signal wires inside the same duct or cable with the power line, even if the distance is short.



Chapter 7. Memory Configuration of the JW10

7-1 Data memory

[1] Kinds of data memory

Kind	Capacity	Relay number (Note 1)	Byte address (Note 1)	File address (Note 1)
Input relay	256 points (32 bytes)	00000 to 00377	コ0000 to コ0037	000000 to 000037
Output relay	256 points (32 bytes)	00400 to 00777	コ0040 to コ0077	000040 to 000077
Auxiliary relay (Note 2)	6656 points (832 bytes)	01000 to 15777	コ0100 to コ1577	000100 to 001577
Timer/counter contact	256 points (64 bytes)	T/C000 to T/C377	—	001600 to 001677
Timer/counter current value	512 bytes	—	b0000 to b0777	002000 to 002777
Register	512 bytes	—	09000 to 09777	004000 to 004777
	512 bytes	—	19000 to 19777	005000 to 005777
	512 bytes	—	29000 to 29777	006000 to 006777
	512 bytes	—	39000 to 39777	007000 to 007777

Note 1: All data memory is handled with octal (except "9" on the 4th digit of the register area). However, numerical figures displayed on support tools (such as JW-13PG and JW-50PG) can be selected from octal, decimal, or hexadecimal by setting system memory (#115).

Note 2: Auxiliary relay and register include special relays and special registers.

■ Special relay

Relay number	Capacity	Contents
07320 to 07337	16 points	Relay for high-speed counter
07340 to 07347	8 points	Relay for data link, remote I/O
07350 to 07377	24 points	Operation flag, error relay etc.

■ Special register

Byte address	Capacity	Contents
コ0100 to コ0104	5 bytes	Register for remote I/O
コ0110 to コ0114	5 bytes	
コ0120 to コ0124	5 bytes	
コ0130 to コ0134	5 bytes	
コ0140 to コ0142	3 bytes	
コ0150 to コ0152	3 bytes	
コ0160 to コ0162	3 bytes	
コ0170 to コ0172	3 bytes	
コ0200 to コ0207	8 bytes	Register for analog input module (JW-14AD)
コ0210 to コ0237	24 bytes	Reserved register (Note 3)
コ0240 to コ0243	4 bytes	Register for analog output module (JW-12DA)
コ0244 to コ0277	28 bytes	Reserved register (Note 3)
コ0740 to コ0767	24 bytes	Register for high-speed counter
コ1400 to コ1557	112 bytes	Register for data link
コ1570 to コ1577	8 bytes	Register for clock data

Note 3: For reserved register, do not use in user program.

[2] Relay area

(1) Input relay (00000 to 00377)

- ON/OFF information of the input port is read into this area during input/output processing at every scan and will be retained for a period of 1 scan cycle. (Note 1)
- Used for input information (contact, source of application instruction) by the user program.

  Note 1: Data memory will be overwritten by the results during the scan cycle, if input relay was used OUT instruction and destination of application instruction in the program.

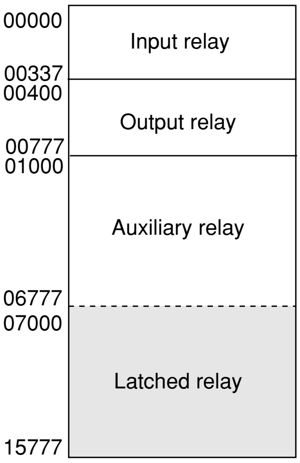
(2) Output relay (00400 to 00777)

- Operational result is stored into this area by the users program as OUT instruction or destination of application instruction.
- Operational result can be used for contact and source of application instruction in the program.
- ON/OFF information of the every scan cycle is transferred to the output port at the input/output processing.

(3) Auxiliary relay (01000 to 15777 except for special relay and special register)

- Operational result is stored into this area by the user program as OUT instruction or destination of application instruction.
- Used for a temporary storage of the operational result that may not be required to send outside.
- Operational result can be used for contact and source of application instruction in the program.
- Relays from 07000 to 15777 in the auxiliary relays are latched relays. The latched relays keep their condition even after power failure.

  The latched relay area can be increased and decreased by setting the system memory (#230, #231).



(4) Timer/counter contact (T000 to T377, C000 to C377)

- When the current value of timer/counter (TMR, CNT) is "0," the timer contact and counter contact are ON.
- TMR and CNT can not use same number.

(5) Relay area byte address (コ0000 to コ1577)

- The JW10 is the programmable controller capable of handling four math rules and data transfer in addition to the bit based operation such as AND and OR.
- A data processing is handled in terms of byte or word. To operate relay area, these areas are accessed in terms of byte address.
- The byte address is an address in terms of byte (8 bits) that corresponds to a relay number. To clearly indicate that it is the byte address, the address is prefixed with "コ" discarding the least significant digit of a 5 digits relay number. The term "コ" represents "code."

  [Example]

02017	02016	02015	02014	02013	02012	02011	02010

The byte address for the above is コ0201.

- Byte address is used by the application instruction to specify the relay area in terms of byte for source and destination.

(6) Special relay (07320 to 07377)

- The following special relays are provided.

■ Special relay for high-speed counter

07320	Count enable relay	[Mode1-1]	07330	Count enable relay	[Mode2]
07321	Preset relay	[Mode1-1]	07331	Preset relay	[Mode2]
07322	Preset release relay	[Mode1-1]	07332	Preset release relay	[Mode2]
07323	Preset status relay	[Mode1-1]	07333	Preset status relay	[Mode2]
07324	Count enable relay	[Mode1-2]	07334	Reserved	
07325	Preset relay	[Mode1-2]	07335	Reserved	
07326	Preset release relay	[Mode1-2]	07336	Reserved	
07327	Preset status relay	[Mode1-2]	07337	Reserved	

■ Special relay for data link and remote I/O

07340	Data link or remote I/O communication flag
07341	Data link or remote I/O communication flag
07342	Data link or remote I/O communication flag
07343	Data link or remote I/O communication flag
07344	Data link or remote I/O communication flag
07345	Data link communication flag
07346	Data link communication flag
07347	Data link communication flag

■ Special relay

07350	Reserved	07360	0.1 second clock	07370	Memory error
07351	Reserved	07361	Reserved	07371	CPU error
07352	Reserved	07362	Initialize pulse	07372	Battery error
07353	Reserved	07363	Reserved	07373	I/O error
07354	Non-carry flag	07364	1 second clock	07374	Communication error
07355	Error flag	07365	Reserved	07375	Reserved
07356	Carry flag	07366	Normally OFF	07376	Reserved
07357	Zero flag	07367	Reserved	07377	Power supply error

- These special relays (07340 to 07377) are areas written by the CPU and used as contacts or sources of application instruction in the user program.
- Do not use them for OUT instruction and destination of application instruction by the user program. Specific attention is required when using the instruction that uses data memory of more than 2 bytes.
- Do not use the reserved area with the user program.

① 07320 to 07337 (relays for high-speed counter)

- Use these relays when using the high-speed counter which is integrated inside the basic module.
- For details, see "Chapter 12: How to Use the High-Speed Counter."

② 07340 to 07347 (data link and remote I/O communication flag)

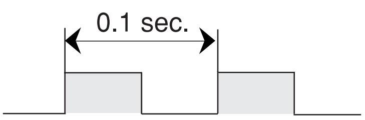
- These areas indicate communication status of each station while data link or the remote I/O is used.
- For details, see "13-3: Data link" and "13-4: Remote I/O."

③ 07354 to 07357 (operation flag)

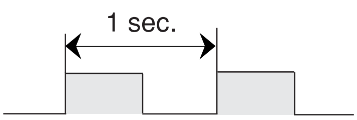
- Set according to the kind of operation, when an application instruction is executed that may affect the flag.
- Refer to page 9・24, "Operation flag" for details.

④ 07360 (0.1 second clock) and 07364 (1 second clock)

- Used for the clock of the CNT instruction and other application instruction.



07360 (0.1 second clock)

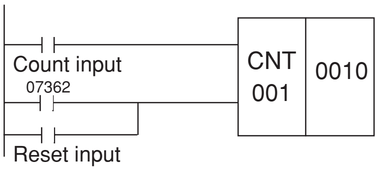


07364 (1 second clock)

⑤ 07362 (Initialize pulse)

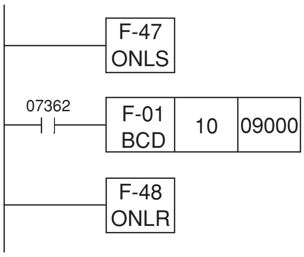
- Turns ON during 1 scan cycle immediately after the run mode of the PC is started.
- "Immediately after the run" indicates the following cases:
  a) The module power is input with the operation mode (monitor mode or change mode)
  b) Changed from stop mode (program mode) to operation mode.
- Used to initially reset (initialize) a CNT instruction or shift register.

Example:



When the module has started operation, reset input turns ON with one scan interval, and counted value of the CNT001 is cleared.

Note 1: If the initialize pulse is used as operation execution trigger instruction input when operation condition changes from ON to OFF, it must be used within the level operation conditions.



Setting of level operation conditions

At start of operation, transfer BCD constant 10 to register 09000.

Resetting of level operation conditions

⑥ 07366 (normally OFF contact)

- Used for the contact that programmed to be normally OFF (a-contact) or normally ON (b-contact).

⑦ 07370 to 07377 (self-diagnostic result)

- If an error was met in a course of diagnosis, the relevant relay will be set ON.
- For details, refer to, "8-3 Self diagnosis."

(7) Special register

- The following registers are provided as special register.

■ Special register for high-speed counter

⊐0740	Current value (lower bits) [Mode1-1]	⊐0750	Current value (lower bits) [Mode1-2]	⊐0760	Current value (lower bits) [Mode2]
⊐0741	Current value (upper bits) [Mode1-1]	⊐0751	Current value (upper bits) [Mode1-2]	⊐0761	Current value (upper bits) [Mode2]
⊐0742	Compare value (lower bits) [Mode1-1]	⊐0752	Compare value (lower bits) [Mode1-2]	⊐0762	Compare value (lower bits) [Mode2]
⊐0743	Compare value (upper bits)[Mode1-1]	⊐0753	Compare value (upper bits)[Mode1-2]	⊐0763	Compare value (upper bits)[Mode2]
⊐0744	Preset value (lower bits) [Mode1-1]	⊐0754	Preset value (lower bits) [Mode1-2]	⊐0764	Preset value (lower bits) [Mode2]
⊐0745	Preset value (upper bits) [Mode1-1]	⊐0755	Preset value (upper bits) [Mode1-2]	⊐0765	Preset value (upper bits) [Mode2]
⊐0746	Reserved	⊐0756	Reserved	⊐0766	Reserved
⊐0747	Reserved	⊐0757	Reserved	⊐0767	Reserved

■ Special register for remote I/O

⊐0100 to ⊐0104	Input relay of slave station 1	⊐0140 to ⊐0142	Output relay of slave station 1
⊐0110 to ⊐0114	Input relay of slave station 2	⊐0150 to ⊐0152	Output relay of slave station 2
⊐0120 to ⊐0124	Input relay of slave station 3	⊐0160 to ⊐0162	Output relay of slave station 3
⊐0130 to ⊐0134	Input relay of slave station 4	⊐0170 to ⊐0172	Output relay of slave station 4

■ Special register for data link

⊐1400 to ⊐1407	Master station to slave station 1 transmitting data	⊐1500 to ⊐1507	Master station to slave station 5 transmitting data
⊐1410 to ⊐1417	Slave station 1 to master station receiving data	⊐1510 to ⊐1517	Slave station 5 to master station receiving data
⊐1420 to ⊐1427	Master station to slave station 2 transmitting data	⊐1520 to ⊐1527	Master station to slave station 6 transmitting data
⊐1430 to ⊐1437	Slave station 2 to master station receiving data	⊐1530 to ⊐1537	Slave station 6 to master station receiving data
⊐1440 to ⊐1447	Master station to slave station 3 transmitting data	⊐1540 to ⊐1547	Master station to slave station 7 transmitting data
⊐1450 to ⊐1457	Slave station 3 to master station receiving data	⊐1550 to ⊐1557	Slave station 7 to master station receiving data
⊐1460 to ⊐1467	Master station to slave station 4 transmitting data		
⊐1470 to ⊐1477	Slave station 4 to master station receiving data		

■ Register for special I/O

⊐0200 to ⊐0207	Data for analog input
⊐0240 to ⊐0243	Data for analog output

■ Register for clock data

⊐1570 to ⊐1577	Clock data

① ⊐0740 to ⊐0767 (Register for high-speed counter)

- Current values etc. of the high-speed counter are stored.
- For details, see "Chapter 12. How to Use the High-Speed Counter."

② ⊐0100 to ⊐0172 (Register for remote I/O)

- Input/output data area of each slave station of the remote I/O.
- For details, see "13-4 Remote I/O."
- When these registers are not used as remote I/O, they can be used as auxiliary relays.

③ ⊐1400 to ⊐1557 (Register for data link)

- Transmitting/receiving data area between master station and each slave station of data link.
- For details, see "13-3 Data link."
- When these registers are not used as data link, they can be used as auxiliary relays.

④ ⊐0200 to ⊐0207 (Register for analog input), ⊐0240 to ⊐0243 (Register for analog output)

- The digital values for analog input module, JW-14AD, and the analog output module, JW-12DA, will be stored here.
- For details, see "Chapter 15. Analog input/output module"

⑤ コ1570 to コ1577 (Data register for clock JW-1424K/1442K/1624K/1642K only)

- JW-1424K, JW-1442K, JW-1624K, and JW-1642K have clock function. (JW-1324K/1342K do not have clock function.)
- 8 bytes of registers from コ1570 to コ1577 are used for reading clock data and time setting.

1) The JW10 automatically recognizes 30-day months, 31-day months and leap years.
2) Year is represented by the lower 2 digits of the Gregorian calendar year.
   Leap years are assumed to occur every four years ('92, '96, and 00 are identified as leap years).
3) Set the day of the week when adjusting the present time. It changes from 0 through 6 each time the day data is increased. Day-of-week data is not computed according to the year/month/day setting.

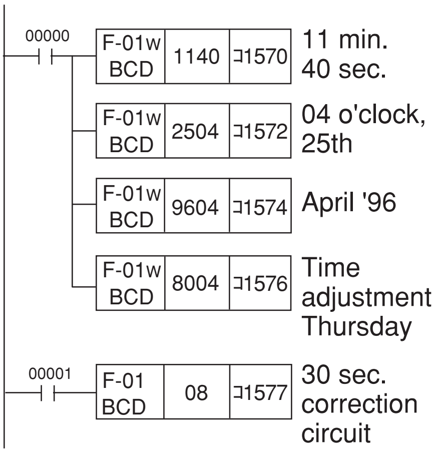
Register no.	Contents
コ1570	Second: 00 to 59 (BCD)
コ1571	Minute: 00 to 59 (BCD)
コ1572	Hour: 00 to 23 (BCD)
コ1573	Day: 01 to 31 (BCD)
コ1574	Month: 00 to 12 (BCD)
コ1575	Year: 00 to 99 (BCD)
コ1576	Day of week: 0 to 6 (BCD)
コ1577	Control

Day of week	SUN	MON	TUE	WED	THU	FRI	SAT
BCD value	00	01	02	03	04	05	06

4) The control byte is affected when register コ1577 bits are set as below :

- Bit D_0 is used to start/stop the clock. If it is set to ON, the clock stops.
- Bit D_3 is used for 30 sec. correction. Carry depends on the present second :
  0 to 29 sec.: Reset to "00" sec., with no carry to the minutes digit.
  30 to 59 sec.: Reset to "00" sec., with a carry to the minutes digit.
- If D_3 is set to "ON," the module automatically resets after completion of auxiliary function.
- Bit D_7, when setting to "ON," allows the time to be adjusted. When it is reset to "OFF," the time monitor mode is selected. Since time adjustment is done after I/O operation are completed, the sequence at right will allow the time to be adjusted without having to stop the clock (D_0: ON).
  If D_7 is set to "ON," the module automatically resets after completion of time adjustment.

コ1577		ON	OFF
D_0	15770	Stop clock	Start clock
D_1	15771	Not in use	
D_2	15772		
D_3	15773	30 sec correction	——
D_4	15774	Not in use	
D_5	15775		
D_6	15776		
D_7	15777	Time adjust	Time monitor



Note 1: Bits D_0 and D_7 of register コ1577 should not be left set to "ON" continuously with an application command, otherwise the clock will not function normally.

Note 2: Do not set illegal clock data (e. g. 30th of February), or the clock may not function normally.

Note 3: The clock has an accuracy of ±30 sec./month (25˚C) and ±60 sec./month (0 to 55˚C). The temperature refers to the clock device's ambient temperature.

- Indication time can easily be adjusted by using initial mode of the hand-held programmer JW-13PG.

[3] Timer/counter current value storage area

- The 512 bytes area of b0000 to b0777 is the area to store current value of timer/counter (TMR,CNT).
- For the total points is 256 for the TMR and CNT, 2 bytes are used per point. See the table 1 for relationship of the TMR and CNT number vs., b**** area.

(Table 1)

TMR, CNT number	Data storage area
000	b0000, b0001
001	b0002, b0003
002	b0004, b0005
003	b0006, b0007
:	:
277	b0576, b0577
300	b0600, b0601
:	:
376	b0774, b0775
377	b0776, b0777

(Table 2)

	7	6	5	4	3	2	1	0	
TMR 000 to TMR 277	($\times 10^{0}$)				($\times 10^{-1}$)				n
	8	4	2	1	8	4	2	1	
	0	0	0	($\times 10^{2}$) 1	($\times 10^{1}$)				n + 1
					8	4	2	1	
TMR 300 to TMR 377	($\times 10^{-1}$)				($\times 10^{-2}$)				n
	8	4	2	1	8	4	2	1	
	0	0	0	($\times 10^{1}$) 1	($\times 10^{0}$)				n + 1
					8	4	2	1	
CNT 000 to CNT 377	($\times 10^{1}$)				($\times 10^{0}$)				n
	8	4	2	1	8	4	2	1	
	0	0	0	($\times 10^{3}$) 1	($\times 10^{2}$)				n + 1
					8	4	2	1	

- The TMR or CNT current value can be used for operation when b0000 to b0777 are specified by the data processing instruction such as F-00.
- The data format of b0000 to b0777 indicate "table 2."

Note 1: Numerical data is handled in BCD for b0000 to b0777.

Note 2: n and n+1 represent the order of addresses. For example, in case of TMR is 001, n = b0002, n+1 = b0003.

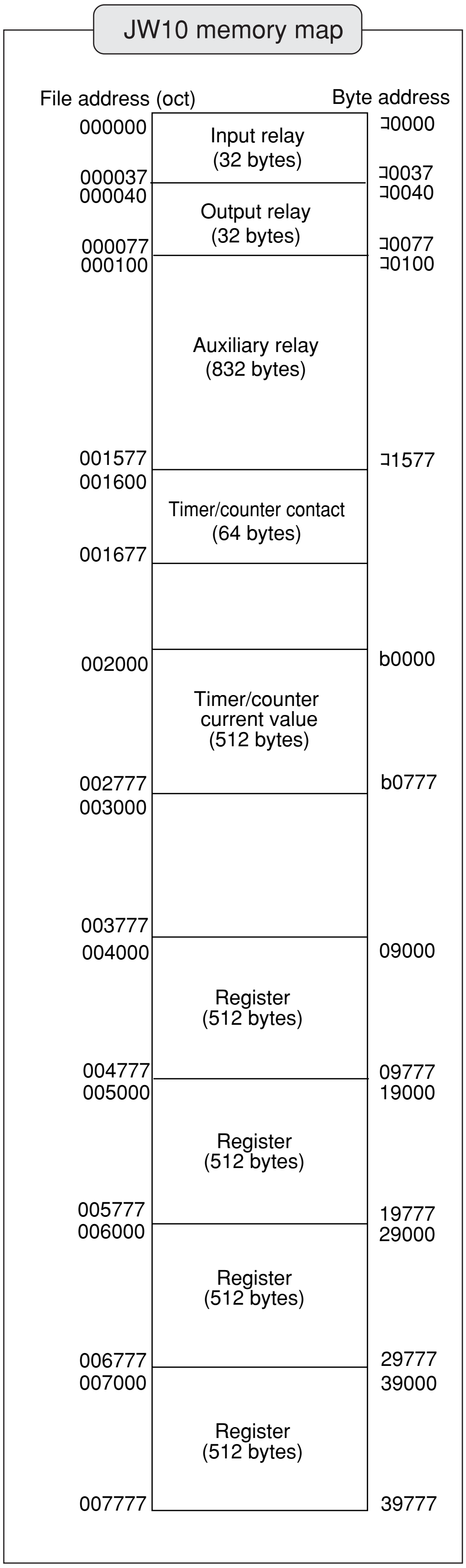
[4] Register area

- 512 bytes each from 09000 to 09777, 19000 to 19777, 29000 to 29777, and 39000 to 39777 are registers for byte unit (8 bits) used to temporarily store operational results.
- These can be used as source and destination area using application instruction of user program.
- Data of the register area are latched at power failure.
- Area from 39000 to 39777 can be used as ROM (EPROM, EEPROM). (JW-1424K, JW-1442K, JW-1624K, and JW-1642K only).

[5] File address

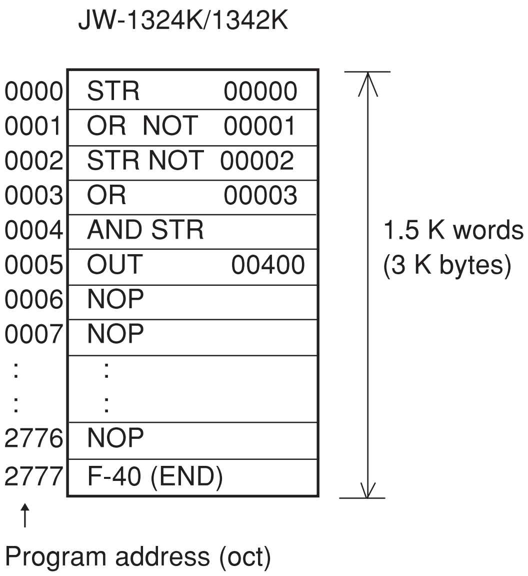
- Concerning byte address, serial numbers are allocated in each area: relay area (コ0000 to コ1577), timer/counter current value area (b0000 to b0777), register area (09000 to 09777, 19000 to 19777, 29000 to 29777, and 39000 to 39777). A group of these areas is referred to as "file."
- Serial numbers called file address are allocated for each file.
- Use file address to assign indirect address using packaged transfer instruction (F-70, F-70w). (See page 9・21.)

  Note 1: Do not write data using indirect address for timer/counter contact point area (file address 001600 to 001677) and CPU's internal processing area (file addresses 001700 to 001777, 003000 to 003777, and 010000 or up).

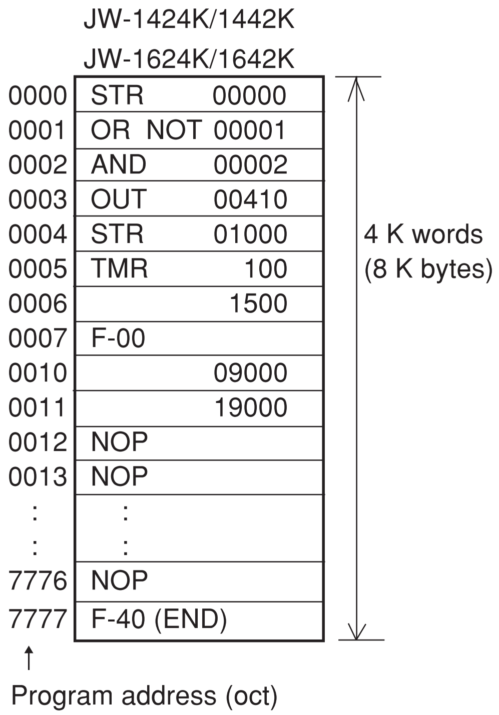


7-2 Program memory

- The program memory is the areas in which the user program is stored. As the programmable controller begins to operate, the program is stated to read from the top address to do operation according to the program.
- In the JW10, the program memory capacity varies with the kinds of basic module .



```
0000  STR          00000
0001  OR  NOT      00001
0002  STR NOT      00002
0003  OR           00003
0004  AND STR
0005  OUT          00400
0006  NOP
0007  NOP
 :      :
 :      :
2776  NOP
2777  F-40 (END)
```



```
0000  STR        00000
0001  OR  NOT    00001
0002  AND        00002
0003  OUT        00410
0004  STR        01000
0005  TMR          100
0006              1500
0007  F-00
0010             09000
0011             19000
0012  NOP
0013  NOP
 :      :
 :      :
7776  NOP
7777  F-40 (END)
```

- To represent the program memory capacity, the term "word" is used, instead of "byte." (1 K bytes expressed as 0.5 K words.)
- As 1 K represents 1024, 1.5 K words represents 1536 words.
- After program memory has been cleared, END instruction (F-40) and NOP instruction (instruction of not to operate) are written in final address and other addresses, respectively.
- Program address is expressed in octal notation. By setting system memory (#115), indication of program address on a support tool such as JW-13PG can be chosen from oct, dcml, or hex.

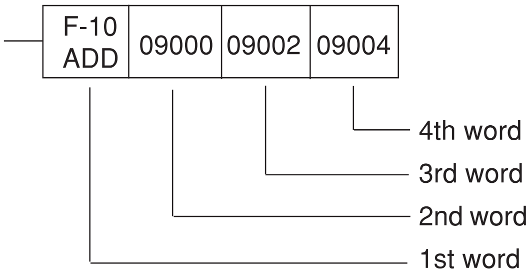
Basic module	Program memory capacity (Number of words)	Program address		
		OCT	DCML	HEX
JW-1324K/1342K	1.5 K words	0000 to 2777	0000 to 1535	0000 to 05FF
JW-1424K/1442K JW-1624K/1642K	4 K words	0000 to 7777	0000 to 4095	0000 to 0FFF

- Instructions are available with 1-word, 2-word, 3-word and 4-word. Each word consists of 2 bytes.

	Typical instruction	Bytes required
1-word	STR, AND etc.	2
2-word	TMR, CNT etc.	4
3-word	F-00, F-01 etc.	6
4-word	F-10, F-11 etc.	8

See the "9-1 Description of instructions" concerning no. of words for each instruction.

(Example of 4-word instruction)



7-3 System memory

The system memory is used to set all kinds of JW10 functions and monitor error contents and the like of JW10.

[1] Table of system memory

- The system memory has a capacity of 256 bytes which occupy an address area from #000 to #377.
- The following are memory numbers opened to the user, but the other memory numbers are reserved area, and data should not be written there.
- System memory numbers are expressed with octal. By setting system memory (#115), indication of program address on a support tool such as JW-13PG can be chosen from oct, dcml, or hex.

(1) System memories used for monitoring each status of JW10.

System memory number	Contents	See page
#030, #031	Monitoring scan time minimum value	7 • 11
#032, #033	Monitoring current value of scan time	7 • 11
#034, #035	Monitoring scan time maximum value	7 • 11
#041	System ROM version	7 • 11
#043	Model code of basic module	7 • 11
#052, #053	Monitor the user program's error address	7 • 12
#054	Monitor the system memory's error address	7 • 12
#160 to #167	Diagnostic error code	7 • 13
#257	BCC check code	7 • 18

(2) System memories used for setting each function of JW10.

System memory number	Contents	Initial value	See page
#055	Setting run/stop in program check error	00(H)	7 • 12
#114	Select notation of address and label numbers	00(H)	7 • 13
#115	Setting numeric figure expression system for address and label numbers	00(H)	7 • 13
#136	Setting the model of support tool	00(H)	7 • 13
#201	TMR reset condition	00(H)	7 • 14
#202	CNT reset condition	00(H)	7 • 14
#203	Mode setting of the high-speed counter	00(H)	7 • 14
#206	Setting output status at PC stopped operation	00(H)	7 • 14
#210	Setting mode of analog input module (JW-14AD)	00(H)	7 • 14
#211	Setting averaging function of analog input module (JW-14AD)	00(H)	7 • 15
#212	Setting mode of analog output module (JW-12DA)	00(H)	7 • 15
#226	Setting computer link transfer specifications of the MMI port	00(H)	7 • 15
#227	Setting computer link station number of the MMI port	000(8)	7 • 15
#230,#231	Setting the latched relay area	000700(8)	7 • 16
#234	Setting communication mode of communication port	00(H)	7 • 16
#235	Setting number of slave stations connected to data link, remote I/O	000(D)	7 • 16
#236	Setting transfer specifications of the communication port	00(H)	7 • 17
#237	Setting station number of communication port	000(8)	7 • 17
#244	Setting enable/disable 10 ms timer interruption	00(H)	7 • 17
#255	Setting ROM operation mode	00(H)	7 • 18

Note: If a value other than the setting range is attempted to be set in the system memory, the module emits system memory error. (See the "8-3 Self-diagnosis.")

[2] Description of system memory

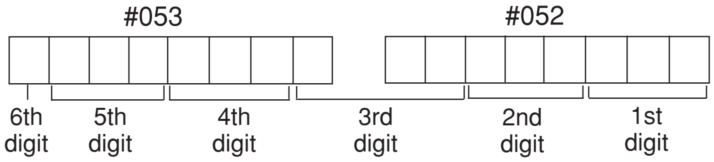
#030 **#031**	Monitoring scan time minimum value	• Scan time minimum value is stored in a BCD value. [Example] If the BCD value monitoring was 0020, the scan time minimum value is 20 ms. 0 0 2 0 Monitored with #030 (Lower digit) Monitored with #031 (Upper digit)
#032 **#033**	Monitoring current value of scan time	• Scan time current value is stored in a BCD value. [Example] If the BCD value monitoring was 0050, the current scan time is 50 ms. 0 0 5 0 Monitored with #032 (Lower digit) Monitored with #033 (Upper digit)
#034 **#035**	Monitoring scan time maximum value	• Scan time maximum value is stored in a BCD number. [Example] If the BCD value monitoring was 0100, the current scan time is 100 ms. 0 1 0 0 Monitored with #034 (Lower digit) Monitored with #035 (Upper digit)

- The scan time measurement starts when the power is turned ON.
- When the operation is changed from operation mode to stop mode (program mode), the latest scan time minimum and maximum values are stored. When the operation is changed from stop mode to operation mode, the existing minimum and maximum values are cleared and newly detected values are stored.
- A scan error allowance is ±1 ms.

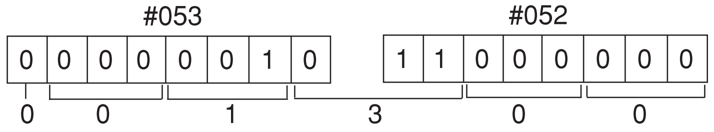
<table>
<tr><td><b>#041</b></td><td>System ROM version</td><td>• Version of system ROM is stored in hexadecimal notation.
[Example]
#041
0 0 0 1 0 0 1 0 The left figures mean version 1.2.
1 2
(Reference) Setting values at address #055 is effective only with modules that have version 1.4 or later ROM code.
The JW-14AD and/or JW-12DA can only be connected to modules that have version 2.0 or later ROM code.</td></tr>
<tr><td><b>#043</b></td><td>Model code</td><td>• Model name code of the basic module is stored in hexadecimal notation.
<table>
<tr><th>Basic module</th><th>Model code</th></tr>
<tr><td>JW-1324K/1342K</td><td>13(H)</td></tr>
<tr><td>JW-1424K/1442K</td><td>14(H)</td></tr>
<tr><td>JW-1624K/1642K</td><td>16(H)</td></tr>
</table></td></tr>
</table>

<table>
<tr>
<td>#052
#053</td>
<td>Monitor the user program's error address</td>
<td>

- If there was the error code "21(H)" (parity error) or "24(H)" (instruction code check error) stored in the system memory #160 after diagnosis, you will know the address in error in the user program when the system memory is monitored.
- The error address is represented by an octal notation, which will be as shown below.

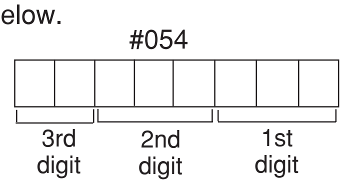


[Example] The bit pattern below indicates an error in the user program address 1300 (octal).

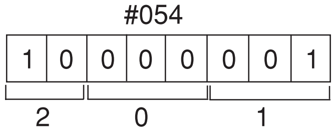


</td>
</tr>
<tr>
<td>#054</td>
<td>Monitor the system memory's error address</td>
<td>

- If there was the error code "23(H)" (system memory error) stored in the system memory #160 after diagnosis, you will know the address in error of system memory that user is setting when the system memory is monitored.
- The error address is represented by an octal notation, which will be as shown below.



[Example]



Set reset condition of TMR in #201. Set either of 00(H) or 01(H). If 02(H), out of the setting range, is set, 23(H) is stored in #160, and its address 201 is stored in #054.

</td>
</tr>
<tr>
<td>#055</td>
<td>Set run/stop in prorgam check error</td>
<td>

- If the PC detects an error in the program check (see page 9・121), you must select whether to continue or to execute an error stop when the PC switches from the stop mode (program mode) to the operation mode (monitor, change mode).

<table>
<tr><th>Setting value</th><th>Contents</th></tr>
<tr><td>00(H)</td><td>Stop in error</td></tr>
<tr><td>55(H)</td><td>Run in error</td></tr>
</table>

- Initial value is 00(H). (Stop in error)

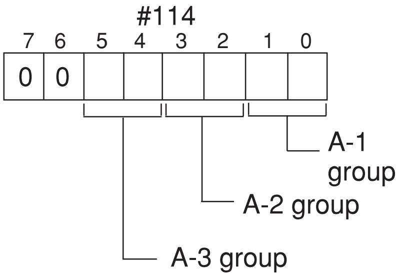
Note : This setting can only be performed on modules that have version 1.4 or later ROM code. Modules with version 1.3 or earlier ROM code will stop the PC operation when it detects an error, regardless of this setting.

</td>
</tr>
</table>

#114

Select notation of application instruction constants.

- Select which notation is used for expressing application instruction constant using a support tool such as JW-13PG.
- Set notation of instruction after classifying them into three.



Setting value of each 2 bits	Contents
00	Initial value (Note)
01	Octal display
10	Decimal display
11	Hexadecimal display

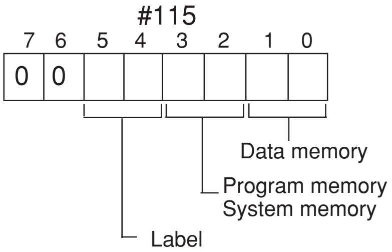
Note: Initial value is the notation described in "Description of each application instruction" in page 9・28 to 116.

A-1 group	Transfer/compare instructions with constants F-01, F-01w, F-07, F-07w, F-08, F-08w F-71, F-71w, Fc12, Fc12w
A-2 group	Instructions having constants in bit pattern specification Fc13, Fc13w, Fc14, Fc14w, Fc18, Fc18w
A-3 group	Instructions having constants in byte count specification/ frequency specification instruction F-70, F-70w, F-74, F-74w, F-144

#115

Select notation of address and label numbers

- Select which notation is used for expressing each address of data memory (relay, TMR/CNT, register number), program memory, and system memory using a support tool such as JW-13PG.



Setting value of each 2 bits	Contents
00	Initial value (Note)
01	Octal display
10	Decimal display
11	Hexadecimal display

Note: Initial value is octal.

[Example] Choose notation for addresses of program and system memory. (Data memory and label is initial value)

Notation	Value of #115	Example of program memory address	Example of system memory address
Octal	00(H), 04(H)	#115	2777
Decimal	08(H)	#077	1535
Hexadecimal	0C(H)	#04D	05FF

#136

Setting the model of support tool

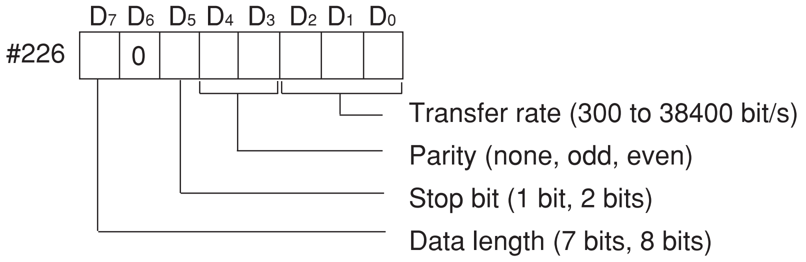
- Select the type of support tool to be connected.

Setting value	Contens
00(H)	Connected with other than JW-2PG
02(H)	Connected with JW-2PG

- Initial value is 00(H). (Connected with other than JW-2PG)

<table>
<tr><td><b>#160
to
#167</b></td><td>Diagnostic error code</td><td>• The error code will be stored when an error is encountered as a result of diagnosis.
• #160 to #167 function as a shift register which will be able to store 8 errors. For details, refer to "8-3 Self diagnosis."
• Error codes remain in system memory after the cause of the error is removed. To clear the error code, write "0" using support tool.</td></tr>
<tr><td><b>#201</b></td><td>TMR reset condition</td><td>• Used to program the state of the TMR instruction upon power recovery.
<table>
<tr><th>Setting value</th><th>Contents</th></tr>
<tr><td>00(H)</td><td>Reset at power recovery.</td></tr>
<tr><td>01(H)</td><td>Store the state at power recovery.</td></tr>
</table>
• Initial value is 00(H). (Reset at power recovery.)</td></tr>
<tr><td><b>#202</b></td><td>CNT reset condition</td><td>• Used to program reset input condition for CNT instruction, each application instruction of F-60, F-60w, F-62 and F-62w.
<table>
<tr><th>Setting value</th><th>Contents</th></tr>
<tr><td>00(H)</td><td>Reset when ON</td></tr>
<tr><td>01(H)</td><td>Reset when OFF</td></tr>
</table>
• Initial value is 00(H). (Reset when ON)</td></tr>
<tr><td><b>#203</b></td><td>Mode setting of the high-speed counter</td><td>• Set mode of the high-speed counter which is integrated inside of the basic module.
<table>
<tr><th>Setting value</th><th>Contents</th></tr>
<tr><td>00(H)</td><td>Not in use (00000 to 00003 are normal input)</td></tr>
<tr><td>01(H)</td><td>Mode 1 (Single-phase rising pulse input: 2 points)</td></tr>
<tr><td>02(H)</td><td>Mode 2 (90˚ phase difference 2-phase signal: 1 point)</td></tr>
</table>
• Initial value is 00(H). (Not in use)</td></tr>
<tr><td><b>#206</b></td><td>Setting output status at PC stopped operation</td><td>• Select output statues when the PC is in stop mode or stopped operation due to detection of an error using self-diagnosis function.
<table>
<tr><th>Setting value</th><th>Contents</th></tr>
<tr><td>00(H)</td><td>Reset (all output are OFF)</td></tr>
<tr><td>55(H)</td><td>Latch</td></tr>
</table>
• In case the module is used as a remote I/O slave station, the module sets output status when an error has occurred in its own station (remote I/O slave station).
• Initial value is 00(H). (Reset.)</td></tr>
<tr><td><b>#210</b></td><td>Setting mode of analog input module JW-14AD [JW-1424K/ 1442K/1624K/ 1642K only]</td><td>• Set operation mode of analog input module JW-14AD.
<table>
<tr><th rowspan="2">Setting value</th><th rowspan="2">Operation mode</th><th colspan="2">Analog input</th><th rowspan="2">Digital value</th></tr>
<tr><th>Voltage input</th><th>Current input</th></tr>
<tr><td>01(H)</td><td>Mode 1</td><td>0 to 12V</td><td>-</td><td>0 to 4000 (12 bits binary)</td></tr>
<tr><td>02(H)</td><td>Mode 2</td><td>0 to 5V</td><td>0 to 20mA</td><td>0 to 2000 (11 bits binary)</td></tr>
<tr><td>03(H)</td><td>Mode 3</td><td>1 to 5V</td><td>4 to 20mA</td><td>0 to 2000 (11 bits binary)</td></tr>
<tr><td>00(H)</td><td colspan="4">JW-14AD dose not work.(Cannot convert analog to digit)</td></tr>
</table>
• JW-14AD has 4 channels, but above setting is applied 4 channels all.
• Initial value is 00(H).(Cannot convert analog to digit.)</td></tr>
</table>

<table>
<tr>
<td><b>#211</b></td>
<td>Setting averaging function of analog input module JW-14AD</td>
<td>
• You can enable/disable the averaging function when the JW-14AD analog input module is used.
<table>
<tr><th>Setting value</th><th>Contents</th></tr>
<tr><td>00(H)</td><td>Averaging disable</td></tr>
<tr><td>01(H)</td><td>Averaging enable</td></tr>
</table>
• JW-14AD has 4 channels, but above setting applies all channels.
• Initial value is 00(H). (Averaging disable.)
</td>
</tr>
<tr>
<td><b>#212</b></td>
<td>Setting mode of analog output module JW-12DA [JW-1424K/ 1442K/1624K/ 1642K only]</td>
<td>
• Set operation mode of analog output module JW-12DA.
<table>
<tr><th rowspan="2">Setting value</th><th rowspan="2">Operation mode</th><th rowspan="2">Digital value</th><th colspan="2">Analog output</th></tr>
<tr><th>Voltage output</th><th>Current output</th></tr>
<tr><td>01(H)</td><td>Mode 1</td><td>0 to 4000 (12 bits binary)</td><td>0 to 10VDC</td><td>-</td></tr>
<tr><td>02(H)</td><td>Mode 2</td><td>0 to 2000 (11 bits binary)</td><td>0 to 5VDC</td><td>0 to 20mA</td></tr>
<tr><td>03(H)</td><td>Mode 3</td><td>0 to 2000 (11 bits binary)</td><td>1 to 5 VDC</td><td>4 to 20mA</td></tr>
<tr><td>00(H)</td><td colspan="4">JW-12DA dose not work.(Cannot convert digit to analog)</td></tr>
</table>
• JW-12DA has 2 channels, but above setting is applied 2 channels all.
• Initial value is 00(H). (Cannot convert digit to analog.)
</td>
</tr>
<tr>
<td><b>#226</b></td>
<td>Setting computer link transfer specifications of the MMI port</td>
<td>
• Set transfer specifications when the MMI port is used with computer link.
• Set bit D0 to D7 on #226.

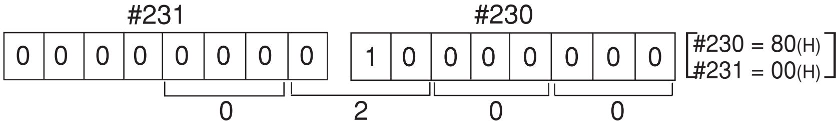


<table>
<tr><th>D7</th><th>Data length</th></tr>
<tr><td>0</td><td>7 bits</td></tr>
<tr><td>1</td><td>8 bits</td></tr>
</table>
<table>
<tr><th>D5</th><th>Stop bit</th></tr>
<tr><td>0</td><td>1 bit</td></tr>
<tr><td>1</td><td>2 bits</td></tr>
</table>
<table>
<tr><th>D4</th><th>D3</th><th>Parity</th></tr>
<tr><td>0</td><td>0</td><td>None</td></tr>
<tr><td>0</td><td>1</td><td>Odd</td></tr>
<tr><td>1</td><td>0</td><td>Even</td></tr>
<tr><td>1</td><td>1</td><td>Disable</td></tr>
</table>
<table>
<tr><th>D2</th><th>D1</th><th>D0</th><th>Transfer rate (bit/s)</th></tr>
<tr><td>1</td><td>1</td><td>1</td><td>38400</td></tr>
<tr><td>0</td><td>0</td><td>0</td><td>19200</td></tr>
<tr><td>0</td><td>0</td><td>1</td><td>9600</td></tr>
<tr><td>0</td><td>1</td><td>0</td><td>4800</td></tr>
<tr><td>0</td><td>1</td><td>1</td><td>2400</td></tr>
<tr><td>1</td><td>0</td><td>0</td><td>1200</td></tr>
<tr><td>1</td><td>0</td><td>1</td><td>600</td></tr>
<tr><td>1</td><td>1</td><td>0</td><td>300</td></tr>
</table>
• Initial value is 00(H).
(19200 bits/s, no parity bit, 1 stop bit, and 7 bits data length.)
</td>
</tr>
<tr>
<td><b>#227</b></td>
<td>Setting computer link station number of MMI port</td>
<td>
• When the MMI port is used in the computer link mode, you must set station number of this port.
• Since the connection of MMI port is performed by 1:1 basis, the station number of this port should be set to "001(8)".
• Initial value is 000(8).
</td>
</tr>
</table>

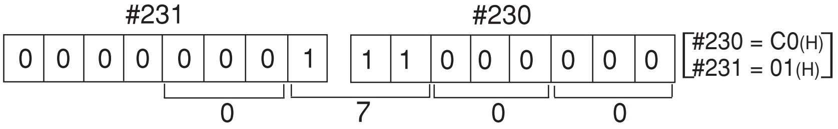
<table>
<tr>
<td>#230
#231</td>
<td>Setting the latched relay area</td>
<td>

- Used to increase/decrease latched relay area from the initial condition.
- Latched relays restore before power failure status at reinput of power from power failure.
- Set with 8 points as unit. Set numeric values by setting byte address with octal.
- Setting value is 0000 to 1577(8).

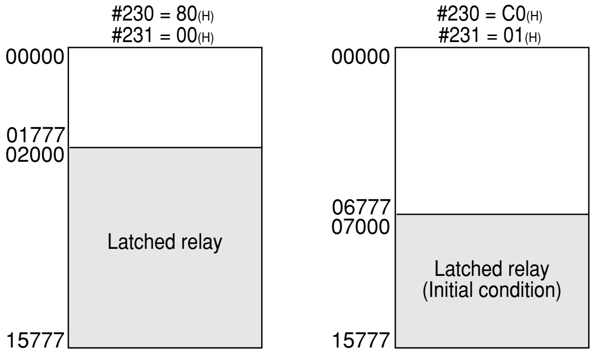
  Example: To assign it to the latched relay area from 02000 (⊐0200 to ⊐1577)



- Initial condition is that relays from 07000 (⊐0700 to ⊐1577) are set as latched relay area.



See page 7・2 "Byte address of relay area" for byte address.



</td>
</tr>
<tr>
<td>#234</td>
<td>Setting communication mode of communication port</td>
<td>

- Set communication mode of communication port.

<table>
<tr><th>Setting value</th><th>Communication mode</th></tr>
<tr><td>00(H)</td><td>Computer link</td></tr>
<tr><td>01(H)</td><td>Data link</td></tr>
<tr><td>02(H)</td><td>Remote I/O</td></tr>
</table>

- Initial value is 00(H). (Computer link)

  Note: When #234 = 02(H) and #237 ≠ 000(8), remote I/O slave station is selected and program operation is disabled.

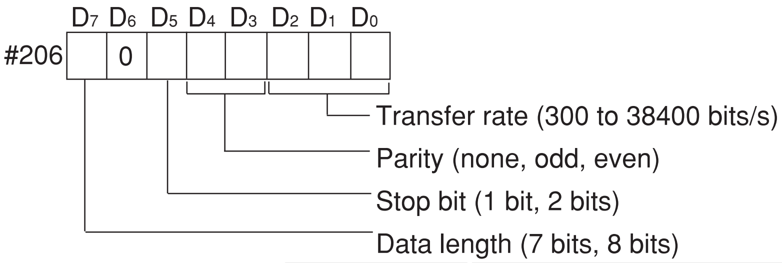
</td>
</tr>
<tr>
<td>#235</td>
<td>Setting number of slave stations connected to data link, remote I/O</td>
<td>

- When the communication port is used as data link or remote I/O (#234 = 01(H) or 02(H)), and the module is data link master station or remote I/O master station, set number of connected slave stations.

<table>
<tr><th>Function</th><th>Setting value</th></tr>
<tr><td>Data link master station</td><td>001 to 007(D)</td></tr>
<tr><td>Remote I/O master station</td><td>001 to 004(D)</td></tr>
</table>

- Initial value is 000(D).

</td>
</tr>
</table>

<table>
<tr>
<td><b>#236</b></td>
<td>Set transfer specifications of the communication port</td>
<td>
(1) In case of computer link
• Set transfer specifications in case of using the communication port for computer link (#234 = 00(H)).
• Set bit D0 to D7 on #236.


<table>
<tr><th>D7</th><th>Data length</th></tr>
<tr><td>0</td><td>7 bits</td></tr>
<tr><td>1</td><td>8 bits</td></tr>
</table>
<table>
<tr><th>D5</th><th>Stop bit</th></tr>
<tr><td>0</td><td>1 bit</td></tr>
<tr><td>1</td><td>2 bits</td></tr>
</table>
<table>
<tr><th>D4</th><th>D3</th><th>Parity</th></tr>
<tr><td>0</td><td>0</td><td>None</td></tr>
<tr><td>0</td><td>1</td><td>Odd</td></tr>
<tr><td>1</td><td>0</td><td>Even</td></tr>
<tr><td>1</td><td>1</td><td>Disable</td></tr>
</table>
<table>
<tr><th>D2</th><th>D1</th><th>D0</th><th>Transfer rate (bit/s)</th></tr>
<tr><td>1</td><td>1</td><td>1</td><td>38400</td></tr>
<tr><td>0</td><td>0</td><td>0</td><td>19200</td></tr>
<tr><td>0</td><td>0</td><td>1</td><td>9600</td></tr>
<tr><td>0</td><td>1</td><td>0</td><td>4800</td></tr>
<tr><td>0</td><td>1</td><td>1</td><td>2400</td></tr>
<tr><td>1</td><td>0</td><td>0</td><td>1200</td></tr>
<tr><td>1</td><td>0</td><td>1</td><td>600</td></tr>
<tr><td>1</td><td>1</td><td>0</td><td>300</td></tr>
</table>
• Initial value is 00(H).
(19200 bits/s, no parity bit, 1 stop bit, and 7 bits data length.)

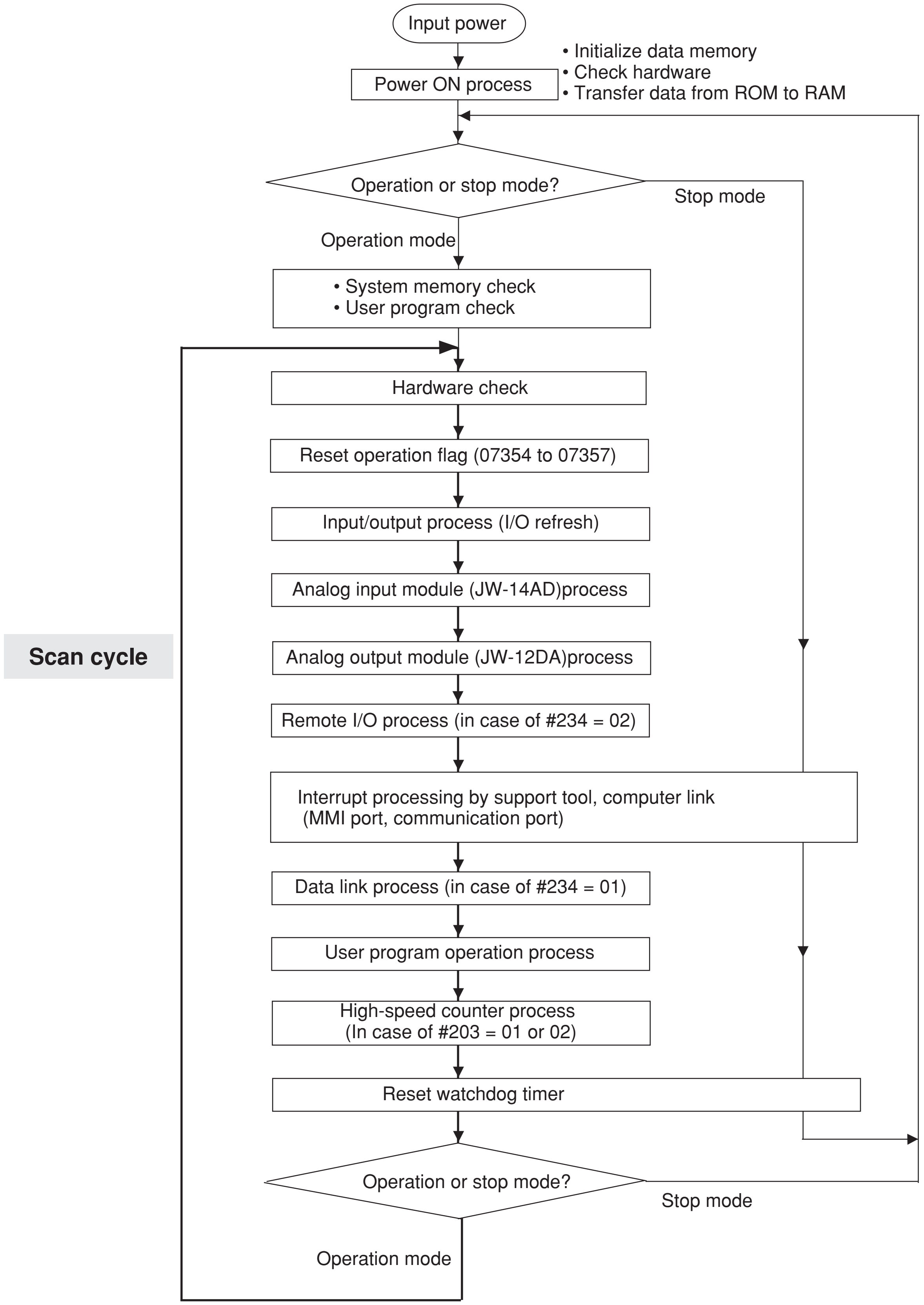
(2) In case of data link and remote I/O
• Set transfer rate in case of using the communication port for data link or remote I/O (#234 = 01(H), 02(H))
<table>
<tr><th>Setting value</th><th>Transfer rate</th><th>Total extension distance</th></tr>
<tr><td>00(H)</td><td>76800 bits/s</td><td>500 m max.</td></tr>
<tr><td>01(H)</td><td>38400 bits/s</td><td>1 km max.</td></tr>
</table>
• Initial value is 00(H). (76800 bits/s)
</td>
</tr>
<tr>
<td><b>#237</b></td>
<td>Setting station number of communication port</td>
<td>
• Set station number of the communication port.
<table>
<tr><th>Communication function</th><th>Master station</th><th>Slave station</th></tr>
<tr><td>Computer link</td><td>—</td><td>001(8) to 077(8)</td></tr>
<tr><td>Data link</td><td>000(8)</td><td>001(8) to 007(8)</td></tr>
<tr><td>Remote I/O</td><td>000(8)</td><td>001(8) to 004(8)</td></tr>
</table>
• Initial value is 000(8).
</td>
</tr>
<tr>
<td><b>#244</b></td>
<td>Set enable/disable 10 ms timer interruption</td>
<td>
• Set enable/disable 10 ms timer interruption.
• See page 8・6 "8-2 Interruption function" concerning timer interruption.
<table>
<tr><th>Setting value</th><th>Contents</th></tr>
<tr><td>00(H)</td><td>Timer interruption prohibited</td></tr>
<tr><td>01(H)</td><td>Timer interruption allowed</td></tr>
</table>
• Initial value is 00(H). (Timer interruption prohibited)
</td>
</tr>
</table>

<table>
<tr>
<td><b>#255</b></td>
<td>Setting ROM operation mode [JW-1424K/1442K/1624K/1642K only]</td>
<td>• Setting ROM area at ROM operation.
<table>
<tr><th>Setting value</th><th>Registerable area to ROM</th></tr>
<tr><td>00(H)</td><td>RAM operation (not ROM operation)</td></tr>
<tr><td>44(H)</td><td>Program memory, system memory (#200 to #377)</td></tr>
<tr><td>45(H)</td><td>Program memory, system memory (#200 to #377)
Register (39000 to 39777)</td></tr>
</table>
• Initial value is 00(H). (RAM operation)</td>
</tr>
<tr>
<td><b>#257</b></td>
<td>BCC check code</td>
<td>• The JW10 automatically computes BCC check code for the contents of system memory #200 to #256.</td>
</tr>
</table>

Chapter 8. Operation of the JW10

8-1 Operation cycle

[1] Operation flow chart



Input power
Power ON process
• Initialize data memory
• Check hardware
• Transfer data from ROM to RAM
Operation or stop mode?
Stop mode
Operation mode
• System memory check
• User program check
Hardware check
Reset operation flag (07354 to 07357)
Input/output process (I/O refresh)
Analog input module (JW-14AD)process
Scan cycle
Analog output module (JW-12DA)process
Remote I/O process (in case of #234 = 02)
Interrupt processing by support tool, computer link (MMI port, communication port)
Data link process (in case of #234 = 01)
User program operation process
High-speed counter process (In case of #203 = 01 or 02)
Reset watchdog timer
Operation or stop mode?
Stop mode
Operation mode

[2] Power ON processing

- When power is input, the JW10 executes the following processes:

(1) Initialize data memory

- The JW10 initializes its data memory. After the initialization, the data memory is arranged as follows:

Data memory	Address	Status after initialization
Input relay	00000 to 00377	Start address of latch function can be assigned by setting system memory #230 and #231. Addresses before the latch function assigned address: All OFF Addresses after the latch function assigned address: Latch ON or OFF status before power failure
Output relay	00400 to 00777	
Auxiliary relay	01000 to15777	
Timer (TMR)	000 to 377	Status at power input can be assigned by setting value of system memory #201. 00(H): The current value is applied with setting value. TMR contact is reset. 01(H): Status before the power failure is kept as current value. TMR contact is kept at ON or OFF status before the power failure.
Counter (CNT)	000 to 377	Keeps current value as the status before the power failure. CNT contact is kept at ON or OFF status before the power failure.
Register	09000 to 09777 19000 to 19777 29000 to 29777 39000 to 39777	Keeps current value as the status before the power failure.

Note 1: When the power is input, the data memory is initialized as shown above. However, input and output relay area of the data memory change as below by input/output process of the first scan cycle.

1) Input relay used area
Changes to ON or OFF in accordance with ON/OFF status of the input device (such as limit switches) connected to the input port.

2) Output relay area and input relay non-used area
These areas do not change from the initialized status until entering operation of user program.

(2) Hardware check

- The JW10 diagnoses its own hardware (such as system ROM, I/O bus, and power supply).
- For self diagnosis function, see "8-3 Self diagnosis."

(3) Transfer data from ROM to RAM

- If JW-1424K, JW-1442K, JW-1624K or JW-1642K is used as basic module, the JW10 can perform ROM operation.
- In case of ROM operation (set system memory #255 = 44(H) or 45 (H)) , the JW10 transfers data from ROM to RAM, and operates according to user program of the ROM.
- For ROM operation, see "Chapter 11: ROM Operation."

[3] Scan cycle

- After JW10 has completed power ON process, and if it is in operation mode, the JW10 checks its system memory and user program (see "8-3 Self diagnosis"), and enters scan cycle.
- Scan cycle consists of a sequence from hardware check to operation process of user program (execute until a step where F-40 END instruction is written). After processing operation of user program, the JW10 again returns to hardware check. It repeats this cycle.
  If the JW10 is using high-speed counter, it processes high-speed counter before returning to hardware check.
- Time taken for this one cycle is referred to "scan time."

(1) Hardware check

- The JW10 diagnoses its own hardware (such as memory, and I/O bus).
- For self diagnosis, see "8-3 Self diagnosis."

(2) Reset of operation flag (07354 to 07357)

- Some application instructions affect operation flags as results of operation. Therefore, flags are cleared before operation process of user program at each scan cycle.
- For operation flag, see "operation flag" on page 9・24.

(3) Input/output process (I/O refresh)

- The JW10 exchanges data between its input/output port and data memory. This is referred to as "I/O refresh."
- In case of the input port, ON/OFF data of input device (such as limit switches) connected to the input port are written into address of data memory corresponding to this input port.
- In case of output port, contents of address of data memory corresponding to the output port are written into the output port so that the output port changes to ON or OFF.

Note 1: At first cycle after inputting the power, contents of the data memory initialized by "power ON process" is written into the output port. In the following cycles, operation result of the previous cycle is written in the output port.

Note 2: Module-not-mounted area of input relay (00000 to 00377) and output relay (00400 to 00777) can be used as auxiliary relays. However, we recommend not to use these areas for future addition purpose of input/output devices.

(4) Analog input module (JW-14AD) process

- The JW10 will exchange data between its data memory (コ0200 to コ0207) and the JW-14AD.
- When the averaging function is enabled (#211 = $01_{(H)}$), the JW10 will also average the incoming data.
- For JW-14AD, see "15-5 How to use JW-14AD".

(5) Analog output module (JW-12DA) process

- The JW10 will exchange data between the JW-12DA and its data memory (コ0240 to コ0243).
- For JW-12DA, see "15-5 How to use JW-12DA".

(6) Remote I/O process

- When the communication port is used with the remote I/O (set system memory #234 = 02), the JW10 exchanges data between the input/output port of the remote I/O slave station and special register (コ0100 to コ0172) for remote I/O.
- For remote I/O, see "13-4 Remote I/O."

(7) Interrupt process by support tool or computer link

- The JW10 exchanges data between a support tool (JW-13PG, JW-50PG etc.) connected to the MMI port and connected devices via computer link.
- When the communication port is used with computer link (set system memory #234 = 00), the JW 10 exchanges data with the connected device.

(8) Data link process

- When the communication port is used as data link (set system memory #234 = 01), the JW10 exchanges data with communication opposite station through data link.
- For data link, see "13-3 Data link."Data link process

(9) Operation process of user program

- The JW10 reads user program in order from its top, and executes operation in accordance with programmed contents. This operation terminates with END instruction (F-40).
- Each instruction of STR, STR NOT, AND, AND NOT, OR, OR NOT, AND STR, and OR STR stores operational result to an accumulator or stack register.
- OUT, TMR, CNT, and most of application instructions (F-xx) write operation results to the data memory.
- For details of each instruction, see "Chapter 9: Description of Instructions."

(10) High-speed counter process

- When the high-speed counter is used (set system memory #203 = 01 or 02), the JW10 exchanges data between special relay for high-speed counter (07320 to 07337) or special register (コ0740 to コ0767) and internal high-speed counter processing hardware section.
- For high-speed counter, see "Chapter 12: How to Use the High-Speed Counter."

(11) Watchdog timer

- Using a hardware watchdog timer, the JW10 checks if its CPU functions normally in accordance with inside processing flow.
- If the JW10 processes normally the scan cycle, its CPU resets the watchdog timer so that the watchdog timer does not time up.
- If the CPU has an error or enters limitless loop program, the CPU cannot reset the watchdog timer so it times up and stops operation.
- This watchdog timer is set to 200 ms.

(12) Scan time

- Interval time of operation from hardware check to END instruction (F-40) is referred to as "scan time." Scan time is roughly given by the equations below:

One scan time (T) = $t_1 + t_2 + t_3 + t_4 + t_5 + t_6 + t_7$

t_1: Fixed processing time (hardware check, reset of watchdog timer etc.)

$t_1 = 560\ \mu s$

t_2: Input/output process time

$t_2 = 120 + 8 \times$ number of input points $+ 6 \times$ number of output points μs

t_3: Analog input module (JW-14AD) process time

In case of averaging disable (#211=00) $t_3 = 1170\ \mu s$

In case of averaging enable (#211=01) $t_3 = 1200\ \mu s$

t_4: Analog output module (JW-12DA) process time

$t_4 = 240\ \mu s$

t_5: Communication process time

In case where data link is applied,

$t_3 = 200\ \mu s$

In case where remote I/O is applied

$t_3 = 1200 + 2600 \times$ number of slave stations μs [in case of 76800 bits/s]

$t_3 = 2400 + 5200 \times$ number of slave stations μs [in case of 38400 bits/s]

t_6: User program processing time (total of all instruction processing time from program address 0000 to END instruction)
For processing time of each instruction, see "9-1 Table of instructions."

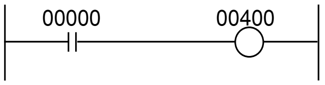
t_7: Interrupted processing time by support tool and computer link

$t_5 = 100\ \mu s$ each

Note 1: If the program memory is cleared, the JW10 writes NOP instruction to all the program memories and END instruction (F-40) on the last address. If instructions are written at intermediate of the program memory, processing time of NOP instruction should be added to scan time. (JW-1324K/1342K: 1.63 μs × number of NOP instructions, JW-1424K/1442K/1624K/1642K: 0.81 μs × number of NOP instructions)
Lastly, write END instruction next to end of program address, the JW10 terminates user program at this address so that scan time can be shortened.

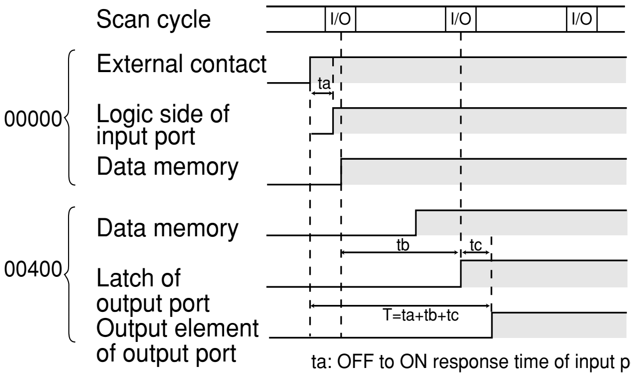
Response time of whole of PC

Response time of the whole range of PC modules including response time of input port and output port of basic and expansion modules are as follows:



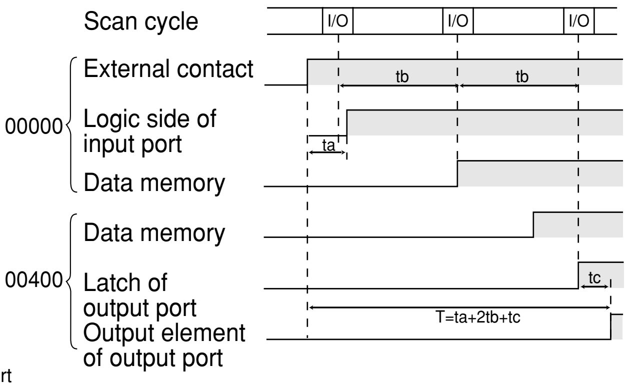
The below shows time from where external contact point 00000 is converted until output element (relay) of output relay 00400 changes in the above ladder program.

(a) In case of the shortest time



ta: OFF to ON response time of input port
tb: One scan time
te: OFF to ON response time of output port

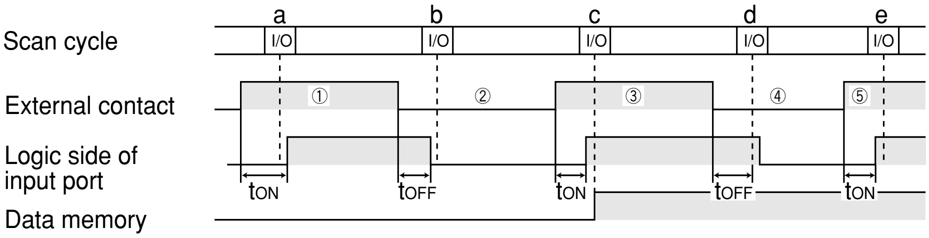
(b) In case of the longest time



In case of ON to OFF, delay by response time of input port and output port also affects total time.

ON/OFF time of input device

In order to securely store ON/OFF status of external contact points into the data memory, scan time of longer than one is required to turn ON or OFF logic side of the input port.



tON: OFF to ON response time of input port
tOFF: ON to OFF response time of input port

- In case of ON external contact point at ①, when logic side of the input port turns ON, input/output process of the corresponding input is already completed. Therefore, logic side of the input port turns OFF just before input/output process of "b" so that data memory is kept OFF.
- In case of ON external contact point at ③, logic side of the input port turns ON just before input/output process of "c" so that ON is written on the data memory.
- In case of OFF external contact point at ④, logic side of the input port is still ON at input/output process of "d," so that data memory is kept ON. In case of input/output processing of "e," input port again turns ON so that the data memory is kept ON.

As such, if ON/OFF time of input port logic side is shorter than one scan time, it may be taken or not be taken, sometimes.

In order to surely reflect ON/OFF status of input devices (such as limit switches), ON or OFF time of these devices should meet the following conditions.

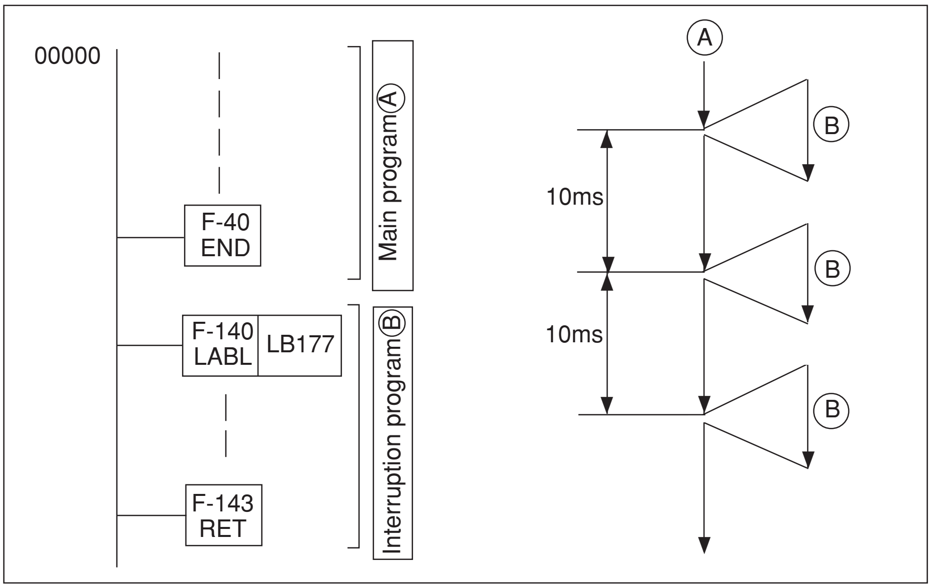
ON time of input device > One scan time + (OFF to ON response time of input port)
OFF time of input device > One scan time + (ON to OFF response time of input port)

8-2 Interruption function

- On the JW10, both timer interruption and high-speed counter interruption are available.
- Using the interruption function, the JW10 realizes high-speed process not restricted by scan cycle.

[1] Timer interruption

- If timer interruption is used, the JW10 shifts program execution to subroutine of LB177 label (F-140) 10 ms each, and returns the program to the address before the interruption by return instruction (F-143).



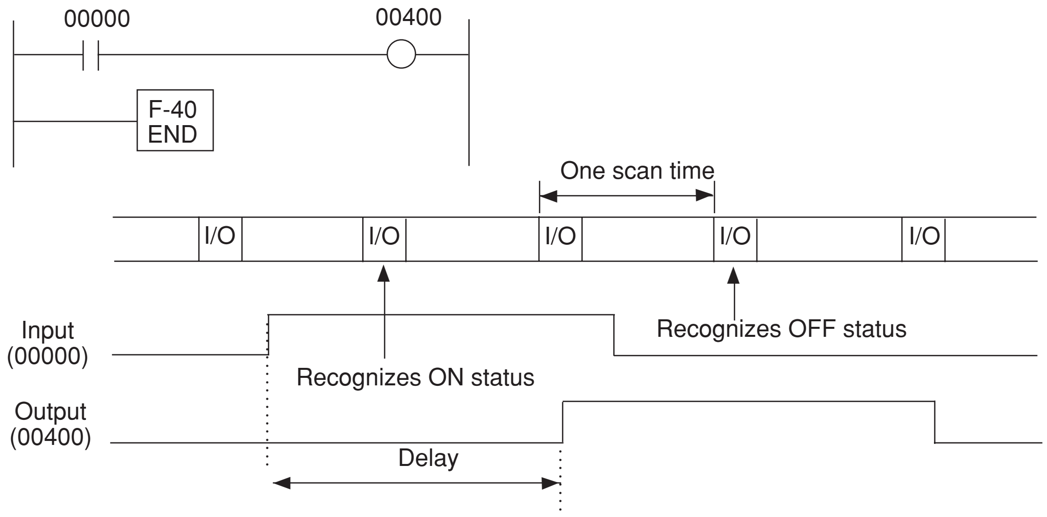
- Interruption can be carried out not only PC operation but also during input/output process.
- Set interruption program within 10 ms operation.
- To set timer interruption, use system memory #244.

Value of #244	Contents
00(H)	Disable timer interruption
01(H)	Enable timer interruption

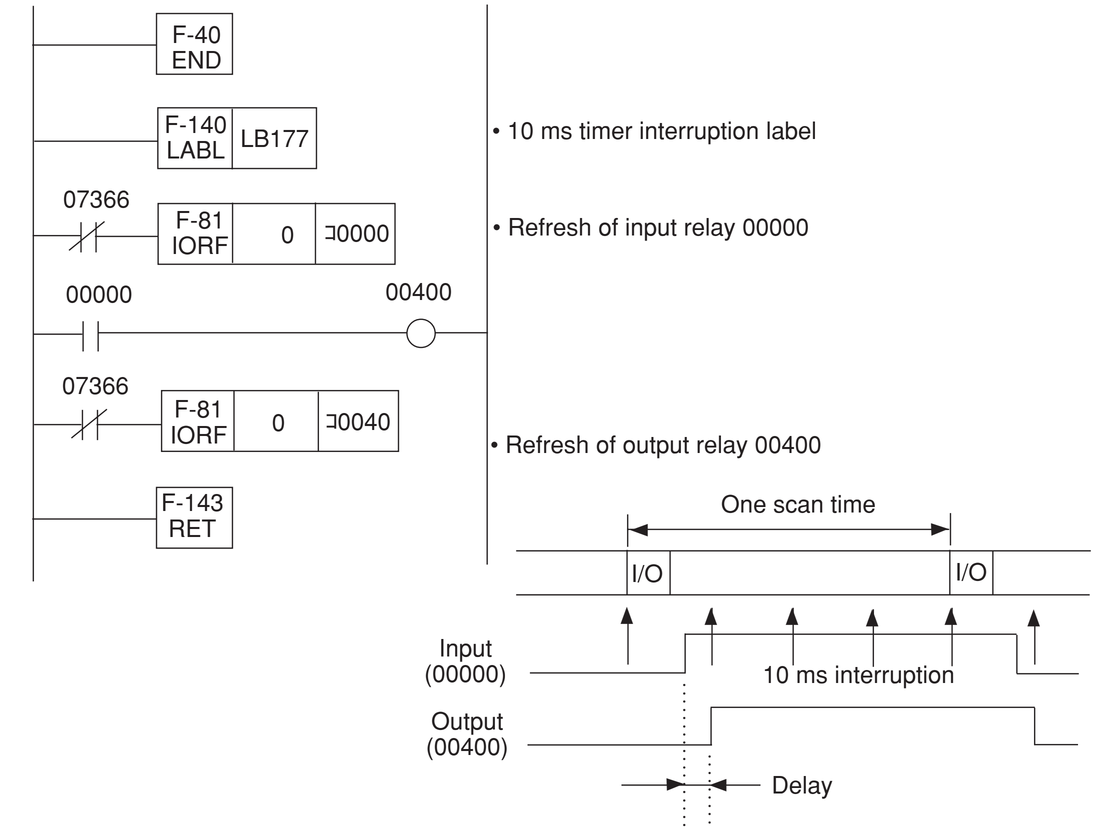
■ An example of timer interruption

- By combined use of timer interruption and I/O refresh instruction (F-81), high-speed response of input/output is possible.

(1) An example of not using timer interruption



(2) An example of using timer interruption



[2] Interruption of high-speed counter

- If count value of the high-speed counter (16 bit binary: 0 to 65535) matches with count comparison value, the JW10 executes interruption program.

		Mode 1		Mode 2
		CH1	CH2	
Count comparison value	Lower digits	⊐0742	⊐0752	⊐0762
	Upper digits	⊐0743	⊐0753	⊐0763
Interruption label		LB170	LB171	LB172

- For details, see “Chapter 12: How to Use High-Speed Counter.”

8-3 Self diagnosis

- JW10 diagnoses its own hardware. If an error occurs, the JW10 treats as the following status. Find the cause with the table below, and treat accordingly.
  For troubleshooting using self diagnosis, see "16-2 Trouble shooting."

Item		Contents	PC operation state	Halt output	LED indicater			Special relay	Error code (BCD) #160 to 167
					Power (green)	RUN (green)	ERR (red)		
Normal	Operation mode	Monitor/change mode	Run	Close (ON)	ON ●	ON ●	OFF ○	–	–
	Stop mode	Program mode	Stop	Open (OFF)	ON ●	Blink ◎	OFF ○	–	–
Error	Memory error	System ROM error	Stop	Open (OFF)	ON ●	OFF ○	ON ●	07370	20
		RAM error							27
		User ROM error 1							26
		User program error 1 (Instruction code check) (*1)							24
		User program error 2 (Parity check) (*1)							21
		User program error 3 (Endless program check) (*6)				Not fixed	Not fixed		25
		System memory error (*2)				OFF ○	ON ●		23
	CPU error	Watchdog timer error				Not fixed	Not fixed	07371	31
	I/O error	I/O bus error				OFF ○	ON ●	07373	44
	Communication error	Remote I/O error						07374	53
	Battery error	Battery voltage drop	Run	Close (ON)	ON ●	ON ●	ON ●	07372	22
	Power error	Power failure or voltage drop	Stop	Open (OFF)	OFF ○	OFF ○	OFF ○	07377	13 (*3)

Note:
*1 Program addresses where errors were detected are stored in #052 to #053.
*2 System memory addresses where errors were detected are stored in #054.
*3 Error code "13" is always stored when power is turned ON.(except JW10 detected WDT error or user program error 3)
*4 JW-1324K/1342K do not have halt output.
*5 In case of stop mode (program mode), RUN lamp blinks. In this case, ERR lamp goes OFF even an error is not recovered. (Only battery error keeps lighting.)
*6 In case of user program error 3, the table above shows the case of reinputting the power.

[1] Contents of self diagnosis

(1) System ROM error

- Checks sum of system ROM.
- This is checked at power input.

(2) RAM error

- Checks whether reading from and writing to RAM is possible.
- This is checked at each scan cycle.

(3) User ROM error

- Checks sum of ROM at transfer data from ROM to RAM with ROM operation.
- Verifies contents of ROM and RAM after transferring data from ROM to RAM and vice versa.

(4) User program error 1 (instruction code check)

- Checks instruction code of user program.
- If there is grammatical error on program (such as doubled use of OUT instruction), this error also occurs. For grammatical error, check with "program check" function of a support tool such as JW-13PG. (See page 9・121 "Program check.")

Note 1: If 55$_{(H)}$ is stored in register #055 in the system memory, the JW10 will continue operation without halting in a "user program error 1" status when it detects an error while checking the program.(The option is available in version 1.4 or later ROM code.)

- Program address of error instruction codes are stored in system memory #052 and #053.
- This is checked at changeover from stop to operation mode.

(5) User program error 2 (parity check)

- Checks parity of program memory.
- Program address having parity errors are stored in system memory #052 and #053.
- This is checked at changeover from stop to operation mode.

(6) User program error 3 (endless program check)

- If jump destination of jump instruction (F-141) is inadequate on user program, and operation enters limitless loop, or operation time of loop instruction (F-144/F-145) is too long, the watchdog timer is not reset so the JW10 stops operation.
- To recover from this error condition, reinput the power and enter stop mode (program mode) so that you can revise the program.

(7) System memory error

- If setting value of system memory is out of setting range, this error occurs.
- Error occurred system memory addresses are stored in system memory #054.
- JW10 calculates BCC codes from #200 to #256, and if this is different from value of #257, this error occurs.
- This is checked at changeover from stop to operation mode.

(8) Watchdog timer error

- If operation cycle is abnormal, the CPU does not reset and the watchdog timer times up. Then the CPU is reset.
- If the JW10 does not recover from this error after reinputting the power or loading program again, the module has hardware error and must be replaced.

(9) I/O bus error

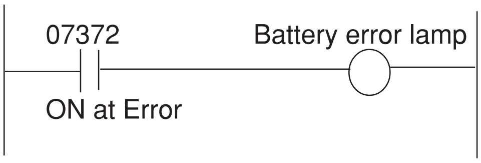
- Checks fault of I/O data bus.
- Also checks connection condition of an expansion module, analog input module, and analog output module.
- If a termination connector is not inserted, this error also occurs.
- This is checked at inputting the power, and each scan cycle.

(10) Remote I/O error

- Checks communication condition with slave stations when a remove I/O is used (system memory #234 = 02).
- If the JW10 cannot communicate normally with any slave station, this error occurs.
- This is checked at every scan cycle.

(11) Battery voltage drop

- If battery voltage for memory backup drops below 2.5 V, this error occurs.
- This is checked every scan cycle.
- Even if this error occurs, JW10 does not stop operation. Therefore, make a circuit to light a lamp or emit buzzer sound at battery error using special relay 07372.



Note 1: Battery error of remote I/O slave station can be checked by slave station battery error flag of master station. For details, see page 13・32.

(12) Power failure or voltage drop of power supply

- If JW-13PG is not connected, the JW10 does not respond to instantaneous power failure of less than 20 ms, and continues operation. In case of power failure longer than this interval, the CPU stops operation, and halt output is opened. (JW-1324K/1342K do not have halt output.)
- When the power failure is recovered, the JW10 automatically restores operation.
- This is checked at inputting power and each scan cycle.

[2] Halt output

- The halt outputs opens when the JW10 determines error by its self diagnosis function (relay output 250 VAC, 30 VDC 1A). It closes at normal operation.
- During the time that power is OFF and stop mode (program mode), this output is opened.
- If emergency stop circuit of a system is connected to halt output of JW10, the system can be stopped at PC error.

Note 1: JW-1324K/1342K do not have halt output.

Note 2: Do not use halt output of remote I/O slave station.

[3] Special relay

- If the JW10 determines error by its self diagnosis function, it writes self diagnosis results to special relay area (07370 to 07377) of the data memory.
- When the error is recovered, the special relays for self diagnoses are also reset. However, they are kept ON at first one scan time soon after the recovery.
- Only battery error (07372) can be taken out from the output port of PC operation among the special relays. Read out other relays through a support tool such as JW-13PG or computer link.
- Power error relay (07377) turns ON only one scan interval after inputting the power.

[4] Error code

- After self diagnosis, if JW10 determines an error has occurred, it writes error code to system memory (#160 to #167).
- #160 to #167 functions as shift register and stores eight errors.
  If number of errors exceeds eight, the first written error is erased.
- Error codes on the system memory are not cleared after recovery from the error.
  To clear the written error codes, write "00" to #160 to #167 using a support tool such as JW-13PG.
- If the same error occurs repeatedly, the JW10 does not write error code of this error.

	#167	#166	#165	#164	#163	#162	#161	#160	
Cleared status	00	00	00	00	00	00	00	00	
	00	00	00	00	00	00	00	27	<-- RAM error
	00	00	00	00	00	00	27	44	<-- I/O bus error
27<-- (Erase)	44	**	**	**	**	**	**	22	<-- Battery error

[5] ON/OFF status of the output port at error

- ON/OFF status of the output port when the JW10 stops operation by self diagnosis result is determined by setting of system memory #206.
- In some error contents, the output port may not be turned OFF. If you want to turn OFF some outputs when JW10 has an error, connect halt output in series. (JW-1324K/1342K do not have halt output function.)
- In case of CPU error or user program error 3, the output port turns OFF regardless of setting contents of #206.

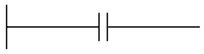
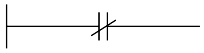
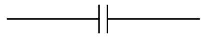
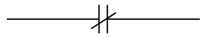
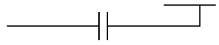
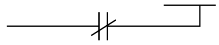
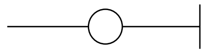
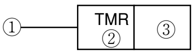
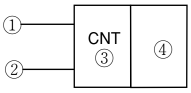
#206	Output port
00(H)	OFF
55(H)	Latch ON/OFF status just before stop

- When the JW10 stops operation, it latches the output value of the JW-12DA analog output module just before stopping. (Holds in case of #206=00).

Chapter 9. Description of Instructions

9-1 Table of instructions

[1] Basic instruction

Instruction	Symbol	Words	Function	Execution time (μs)				See page
				JW-1324K/1342K		JW-1424K/1442K JW-1624K/1642K		
				Executing	Not executing	Executing	Not executing	
STR		1	Starts at normally open contact and intermediate result is stored.	1.83		1.02		9·9
STR NOT		1	Starts at normally closed contact and intermediate result is stored.	1.83		1.02		10
AND		1	AND	1.83		1.02		11
AND NOT		1	AND NOT	1.83		1.02		11
OR		1	OR	1.83		1.02		12
OR NOT		1	OR NOT	1.83		1.02		12
AND STR		1	AND with the intermediate result	1.63		0.81		13
OR STR		1	OR with the intermediate result	1.63		0.81		14
OUT		1	Output result	6.9		5.9		15
TMR		2	Timer (decremental) ① Start input ② TMR number (000 to 377) ③ Setting value(1 to 1999) (0.1 to 199.9 sec. (TMR000 to 277)) (0.01 to 19.99 sec.(TMR300 to 377))	130	94	127	91	16
CNT		2	Counter (decremental) ① Calculation input ② Reset input ③ CNT number (000 to 377) ④ Setting value (1 to 1999)	136	102	133	99	18

[2] Application instruction (numeric order)

Instruction	Symbol	Words	Function	Execution time (μs)				See page
				JW-1324K/1342K		JW-1424K/1442K JW-1624K/1642K		
				Executing	Not executing	Executing	Not executing	
F-00	F-00 XFER S D	3	Transfer data register to data register (1 byte)	35	17	33	13	9·28
F-00w	F-00w XFER S D	3	Transfer data register to data register (1 word)	33	17	31	13	28
F-01	F-01 BCD n D	3	Transfer BCD constant (2 digits)	33	17	29	15	29
F-01w	F-01w BCD n D	3	Transfer BCD constant (4 digits)	31	17	29	15	29
F-02	F-02 XCHG D_1 D_2	3	Exchange registers (1 byte)	35	17	31	13	30
F-02w	F-02w XCHG D_1 D_2	3	Exchange registers (1 word)	35	17	33	13	30
F-03	F-03 →BIN S D	3	Convert 2 digits BCD to 8 bits binary	67	17	63	15	31
F-03w	F-03w →BIN S D	3	Convert 4 digits BCD to 16 bits binary	117	17	115	15	32
F-04	F-04 →BCD S D	3	Convert 8 bits binary to 2 digits BCD	133	17	129	15	33
F-04w	F-04w →BCD S D	3	Convert 16 bits binary to 6 digits BCD	329	17	311	15	33
F-07	F-07 DCML n D	3	Transfer 1 byte decimal constant	31	17	29	15	34
F-07w	F-07w DCML n D	3	Transfer 1 word decimal constant	31	17	31	15	34
F-08	F-08 OCT n D	3	Transfer 1 byte octal constant	33	19	31	17	35
F-08w	F-08w OCT n D	3	Transfer 1 word octal constant	49	19	47	17	35
F-09	F-09 INV S D	3	Complement 8 bits data	35	19	33	17	36
F-09w	F-09w INV S D	3	Complement 16 bits data	35	19	31	17	36
F-10	F-10 ADD S_1 S_2 D	4	Add register and register (BCD 2 digits)	105	21	101	19	37
F-10w	F-10w ADD S_1 S_2 D	4	Add register and register (BCD 4 digits)	193	21	187	19	38
Fc10	Fc10 ADD S_1 n D	4	Add register (BCD 2 digits) and constant (BCD 2 digits)	109	21	89	19	39
Fc10w	Fc10w ADD S_1 n D	4	Add register (BCD 4 digits) and constant (BCD 4 digits)	147	21	141	19	40
F-11	F-11 SUB S_1 S_2 D	4	Subtract register from register (BCD 2 digits)	103	21	99	17	41
F-11w	F-11w SUB S_1 S_2 D	4	Subtract register from register (BCD 4 digits)	195	21	185	17	42
Fc11	Fc11 SUB S_1 n D	4	Subtract constant (BCD 2 digits) from register (2 digits)	93	21	87	17	43
Fc11w	Fc11w SUB S_1 n D	4	Subtract constant (BCD 4 digits) from register (4 digits)	146	21	139	17	44
F-12	F-12 CMP S_1 S_2	3	Compare register with register (1 byte)	47	13	45	11	45
F-12w	F-12w CMP S_1 S_2	3	Compare register with register (1 word)	85	13	83	11	46
Fc12	Fc12 CMP S_1 n	3	Compare register with octal constant (1 byte)	47	13	45	11	47
Fc12w	Fc12w CMP S_1 n	3	Compare register with octal constant (1 word)	63	13	61	11	48
F-13	F-13 AND S D	3	AND register with register (1 byte)	43	19	39	19	49
F-13w	F-13w AND S D	3	AND register with register (1 word)	83	19	79	19	49
Fc13	Fc13 AND n D	3	AND register with octal constant (1 byte)	43	19	39	19	50
Fc13w	Fc13w AND n D	3	AND register with octal constant (1 word)	63	19	59	19	50

Instruction	Symbol	Words	Function	Execution time (μs)				See page
				JW-1324K/1342K		JW-1424K/1442K JW-1624K/1642K		
				Executing	Not executing	Executing	Not executing	
F-14	F-14 OR | S | D	3	OR register with register (1 byte)	43	19	41	17	9·51
F-14w	F-14w OR | S | D	3	OR register with register (1 word)	83	19	79	17	51
Fc14	Fc14 OR | n | D	3	OR register with octal constant (1 byte)	41	19	39	17	52
Fc14w	Fc14w OR | n | D	3	OR register with octal constant (1 word)	63	19	61	17	52
F-15	F-15 MUL | S_1 | S_2 | D	4	Multiply register by register (BCD 4 digits)	465	19	459	17	53
Fc15	Fc15 MUL | S_1 | n | D	4	Multiply register (BCD 4 digits) by constant (BCD 3 digits)	419	19	413	17	54
F-16	F-16 DIV | S_1 | S_2 | D	4	Divide register (BCD 4 digits) by register (BCD 2 digits)	383	19	377	17	55
Fc16	Fc16 DIV | S_1 | n | D	4	Divide register (BCD 4 digits) by constant (BCD 2 digits)	369	19	363	17	56
F-18	F-18 XOR | S | D	3	Exclusive OR register with register (1 byte)	39	17	35	15	57
F-18w	F-18w XOR | S | D	3	Exclusive OR register with register (1 word)	41	17	37	15	57
Fc18	Fc18 XOR | n | D	3	Exclusive OR register with octal constant (1 byte)	39	17	35	15	58
Fc18w	Fc18w XOR | n | D	3	Exclusive OR register with octal constant (1 word)	41	17	37	15	58
F-30	F-30 MCS ┐	1	Set master control	19	—	17	—	59
F-31	F-31 MCR	1	Reset master control	9	—	7	—	59
F-32	F-32 SET | R	2	Set coil	29	19	25	15	62
F-33	F-33 RST | R	2	Reset coil	29	19	25	15	62
F-34	F-34 TSET | n_1 | n_2 | BIT	4	Comparison with current value of clock (specified relay set)	—	—	33	25	64
F-35	F-35 TRST | n_1 | n_2 | BIT	4	Comparison with current value of clock (specified relay reset)	—	—	35	29	65
F-40	F-40 END	1	End instruction	30	—	30	—	66
F-41	F-41 JCS	1	Set jump control	19	—	17	—	67
F-42	F-42 JCR	1	Reset jump control	9	—	7	—	67
F-43	F-43 ──▷o──	1	Complement bit (ACC contents)	6	—	4	—	69
F-44	F-44 ──┤↑├──	1	Differentiate at ON	22	19	20	17	70
F-45	F-45 ──┤↓├──	1	Differentiate at OFF	23	19	21	17	71
F-47	F-47 ONLS	1	ON level set	6	—	4	—	72
F-48	F-48 ONLR	1	ON level reset	6	—	4	—	72
F-50	F-50 4→16 | S | D	3	Decode 4 to 16	33	15	29	13	73
F-51	F-51 16→ 4 | S | D	3	Encode 16 to 4	97	15	95	13	73
F-52	F-52 7SEG | S | D	3	Decode to 7 segments data	33	15	31	13	74
F-53	F-53 →BIN | S | D	3	Convert 4 digits BCD to 16 bits binary	117	17	115	15	75
F-54	F-54 →BCD | S | D	3	Convert 16 bits binary to 6 digits BCD	329	17	311	15	76
F-55	F-55 SWAP | S | D	3	Swap upper 4 bits with lower 4 bits	31	15	29	13	77

Instruction	Symbol	Words	Function	Execution time (μs)				See page
				JW-1324K/1342K		JW-1424K/1442K JW-1624K/1642K		
				Executing	Not executing	Executing	Not executing	
F-58	F-58 Σ BIT n S D	4	Total of ON bits	83	29	81	13	9･77
F-60	① ② ③ ④ F-60 SFR D	2	Shift register bidirectional (1 byte) ① Shift direction input ② Data input ③ Shift input ④ Reset input	114	32	113	31	78
F-60w	① ② ③ ④ F-60w SFR D	2	Shift register bidirectional (1 word) ① Shift direction input ② Data input ③ Shift input ④ Reset input	115	32	114	32	80
F-62	① ② ③ F-62 U/DC D	2	2 digits BCD up/down counter ① Up/down counter direction input ② Counter input ③ Reset input	59	51	87	47	81
F-62w	① ② ③ F-62w U/DC D	2	4 digits BCD up/down counter ① Up/down counter direction input ② Counter input ③ Reset input	75	51	125	47	82
F-63	F-63 INC D	2	Add binary counter (1 byte)	55	29	53	29	83
F-63w	F-63w INC D	2	Add binary counter (1 word)	57	29	55	29	83
F-64	F-64 DEC D	2	Subtract binary counter (1 byte)	57	31	55	29	84
F-64w	F-64w DEC D	2	Subtract binary counter (1 word)	59	31	57	29	84
F-70	F-70 FILE n S D	4	Transfer n bytes block	(Note)	19	(Note)	13	85
F-70w	F-70w FILE n S D	4	Transfer n words block	(Note)	19	(Note)	13	86
F-71	F-71 CONS n D_1 D_2	4	Transfer octal constant block (1 byte)	(Note)	57	(Note)	53	87
F-71w	F-71w CONS n D_1 D_2	4	Transfer octal constant block (1 word)	(Note)	57	(Note)	53	88
F-74	F-74 nXFR n S D	4	Transfer n bytes	(Note)	19	(Note)	15	89
F-74w	F-74w nXFR n S D	4	Transfer n words	(Note)	19	(Note)	15	90
F-80	F-80 IORF D	2	I/O refresh (1 byte)	140	16	138	14	91
F-81	F-81 IORF n D	3	I/O refresh (1 bit)	140	16	138	14	91
F-90	F-90 REM n	2	Remark n = 0000 to 3777	3.25	—	1.65	—	92
F-140	F-140 LABL LBn	2	Set label LB0000 to LB0177	0	—	0	—	93
F-141	F-141 JMP LBn	2	Jump to label	19	9	17	7	94
F-142	F-142 CALL LBn	2	Call labeled subroutine	33	20	31	18	96
F-143	F-143 RET	1	Call subroutine label	38	23	37	22	96
F-144	F-144 FOR n	2	Set Loop count	36	20	34	18	97
F-145	F-145 NEXT	1	END of Loop	15	8	14	7	97
F-210	F-210 ADD S_1 S_2 D	4	Add register and register in binary (8 bits + 8 bits)	85	23	77	17	98
F-210w	F-210w ADD S_1 S_2 D	4	Add register and register in binary (16 bits + 16 bits)	89	23	81	17	99
Fc210	Fc210 ADD S_1 n D	4	Add register and constant in binary (8 bits + 8 bits)	85	23	79	17	100
Fc210w	Fc210w ADD S_1 n D	4	Add register and constant in binary (16 bits + 16 bits)	87	23	81	17	101

Instruc-tion	Symbol	Words	Function	Execution time (μs)				See page
				JW-1324K/1342K		JW-1424K/1442K JW-1624K/1642K		
				Executing	Not executing	Executing	Not executing	
F-211	F-211 SUB S_1 S_2 D	4	Subtract register from register in binary (8 bits − 8 bits)	85	21	81	19	9·102
F-211w	F-211w SUB S_1 S_2 D	4	Subtract register from register in binary (16 bits − 16 bits)	87	21	83	19	103
Fc211	Fc211 SUB S_1 n D	4	Subtract constant from register in binary (8 bits − 8 bits)	83	21	79	19	104
Fc211w	Fc211w SUB S_1 n D	4	Subtract constant from register in binary (16 bits − 16 bits)	85	21	81	19	105
F-212	F-212 WNDW S_1 S_2 S_3	4	Window comparator (1 byte register)	83	19	73	15	106
F-212w	F-212w WNDW S_1 S_2 S_3	4	Window comparator (1 word register)	85	19	75	15	107
Fc212	Fc212 WNDW S_1 n_1 n_2	4	Window comparator (between 1 byte octal constants)	81	19	67	15	108
Fc212w	Fc212w WNDW S_1 n_1 n_2	4	Window comparator (between 1 word octal constant)	83	19	69	15	108
F-215	F-215 MUL S_1 S_2 D	4	Multiply register by register in binary (8 bits × 8 bits)	59	19	57	17	109
F-215w	F-215w MUL S_1 S_2 D	4	Multiply register by register in binary (16 bits × 16 bits)	61	19	59	17	110
Fc215	Fc215 MUL S_1 n D	4	Multiply register by constant in binary (8 bits × 8 bits)	57	19	55	17	111
Fc215w	Fc215w MUL S_1 n D	4	Multiply register by constant in binary (16 bits × 16 bits)	59	19	57	17	112
F-216	F-216 DIV S_1 S_2 D	4	Divide register by register in binary (8 bits ÷ 8 bits)	59	19	57	17	113
F-216w	F-216w DIV S_1 S_2 D	4	Divide register by register in binary (15 bits ÷ 15 bits)	63	19	77	17	114
Fc216	Fc216 DIV S_1 n D	4	Divide register by constant in binary (8 bits ÷ 8 bits)	59	19	53	17	115
Fc216w	Fc216w DIV S_1 n D	4	Divide register by constant in binary (15 bits ÷ 15 bits)	75	19	59	17	116
NOP		1	Non-operation instruction	1.63	—	0.81	—	—

Note: Processing time of instruction F-70, F-71, and F-74 are determined by number of transfer bytes. Processing times of instruction F-70w, F-71w, and F-74w are determined by number of transfer words.

Instruction	JW-1324K/1342K	JW-1424K/1442K JW-1624K/1642K	Remarks
F-70	46 + 6.5B	41 + 6.5B	B : No. of bytes (1 to 256) W : No. of words (1 to 256) Unit : μs
F-70w	47 + 11.6W	42 + 17.8W	
F-71	78 + 3.0B	62 + 3.5B	
F-71w	86 + 2.9W	66 + 2.9W	
F-74	40 + 4.9B	39 + 4.8B	
F-74w	49 + 5.0W	46 + 4.8W	

[3] Application instructions (functional order)

Type				Instruction	See page
Transfer instructions	Register to Register transfer		1 byte	F-00	9·28
			1 word	F-00w	28
			n bytes	F-70	85
			n words	F-70w	86
			n bytes (Same data)	F-74	89
			n words (Same data)	F-74w	90
	BCD constant transfer		2 digits	F-01	29
			4 digits	F-01w	29
	Decimal constant transfer		1 byte	F-07	34
			1 word	F-07w	34
	Octal constant transfer		1 byte	F-08	35
			1 word	F-08w	35
			n bytes	F-71	87
			n words	F-71w	88
Arithmetic operation instructions	BCD addition	Register and register	2digits + 2digits	F-10	37
			4digits + 4digits	F-10w	38
		Constant and register	2digits + 2digits	Fc10	39
			4digits + 4digits	Fc10w	40
	BCD subtraction	Register and register	2digits - 2digits	F-11	41
			4digits - 4digits	F-11w	42
		Constant and register	2digits - 2digits	Fc11	43
			4digits - 4digits	Fc11w	44
	BCD multiplication	Register and register	4digits×4digits	F-15	53
		Constant and register	4digits×3digits	Fc15	54
	BCD division	Register and register	4digits÷2digits	F-16	55
		Constant and register	4digits÷2digits	Fc16	56
	Binary addition	Register and register	8 bits + 8 bits	F-210	98
			16 bits + 16 bits	F-210w	99
		Constant and register	8 bits + 8 bits	Fc210	100
			16 bits + 16 bits	Fc210w	101
	Binary subtraction	Register and register	8 bits - 8 bits	F-211	102
			16 bits - 16 bits	F-211w	103
		Constant and register	8 bits - 8 bits	Fc211	104
			16 bits - 16 bits	Fc211w	105
	Binary Multiplication	Register and register	8 bits×8 bits	F-215	109
			16 bits×16 bits	F-215w	110
		Constant and register	8 bits×8 bits	Fc215	111
			16 bits×16 bits	Fc215w	112
	Binary division	Register and register	8 bits÷8 bits	F-216	113
			15 bits÷15 bits	F-216w	114
		Constant and register	8 bits÷8 bits	Fc216	115
			15 bits÷15 bits	Fc216w	116

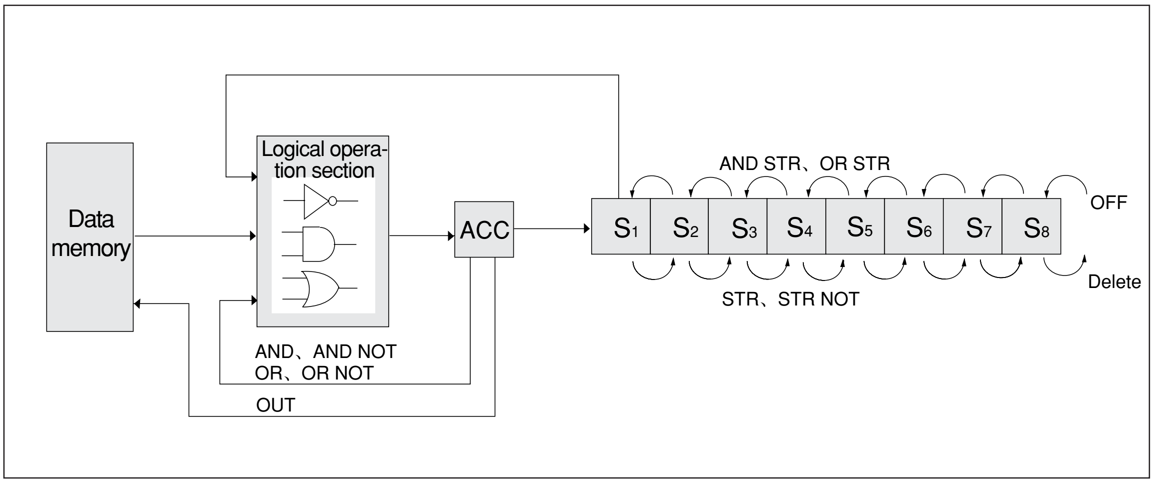
Type				Instruction	See page
Logical operation instructions	AND	Register and register	8 bits	F-13	9·49
			16 bits	F-13w	49
		Register with octal constant	8 bits	Fc13	50
			16 bits	Fc13w	50
	OR	Register and register	8 bits	F-14	51
			16 bits	F-14w	51
		Register with octal constant	8 bits	Fc14	52
			16 bits	Fc14w	52
	Exclusive OR	Register and register	8 bits	F-18	57
			16 bits	F-18w	57
		Register with octal constant	8 bits	Fc18	58
			16 bits	Fc18w	58
	Complement		8 bits	F-09	36
			16 bits	F-09w	36
Compare instructions	Compare	Register and register	1 byte	F-12	45
			1 word	F-12w	46
		Register with octal constant	1 byte	Fc12	47
			1 word	Fc12w	48
	Window comparator	Register and register	1 byte	F-212	106
			1 word	F-212w	107
		Register with octal constant	1 byte	Fc212	108
			1 word	Fc212w	108
Convert instructions	Convert BCD to BIN		2digits →8 bits	F-03	31
			4digits →16 bits	F-03w	32
			4digits →16 bits	F-53	75
	Convert BIN to BCD		8digits →2 bits	F-04	33
			16digits →6 bits	F-04w	33
			16digits →6 bits	F-54	76
	Decode from 4 to 16			F-50	73
	Encode from 16 to 4			F-51	73
	Decode 7 SEG			F-52	74
	Total of ON bit			F-58	77
Exchange instructions	Exchange data		1 byte	F-02	30
			1 word	F-02w	30
	Swap high order 4 bits with low order 4bits		1 byte	F-55	77
Bit processing instructions	Complement bit			F-43	69
	Differentiate at ON			F-44	70
	Differentiate at OFF			F-45	71
	Set coil			F-32	62
	Reset coil			F-33	62

Type			Instruction	See page
Counter instructions	BCD up/down counter	2 digits	F-62	9･81
		4 digits	F-62w	82
	Add binary counter	1 byte	F-63	83
		1 word	F-63w	83
	Subtract binary counter	1 byte	F-64	84
		1 word	F-64w	84
Shift instructions	Reversible shift register	8 bits	F-60	78
		16 bits	F-60w	80
Operational condition instructions	Set master control		F-30	59
	Reset master control		F-31	59
	Set jump control		F-41	67
	Reset jump control		F-42	67
	Set level operating condition		F-47	72
	Reset level operating condition		F-48	72
	End		F-40	66
Branch instructions	Label		F-140	93
	Jump		F-141	94
	Call subroutine		F-142	96
	Return from subroutine		F-143	96
	Set loop count		F-144	97
	End of loop		F-145	97
Clock instructions	Comparison with current value of clock (specified relay set)		F-34	64
	Comparison with current value of clock (specified relay reset)		F-35	65
Other instructions	Refresh I/O	1 byte	F-80	91
		1 bit	F-81	91
	Remark (instruction for comment identification)		F-90	92

9-2 Basic instruction

[1] Execution of basic instruction

- The JW10 uses the data memory section, logical operation section, accumulator (ACC), and stack registers (S_1 to S_8) for operating basic instructions.


Data memory
Logical operation section
ACC
AND STR、OR STR
S_1 S_2 S_3 S_4 S_5 S_6 S_7 S_8
OFF
Delete
STR、STR NOT
AND、AND NOT
OR、OR NOT
OUT

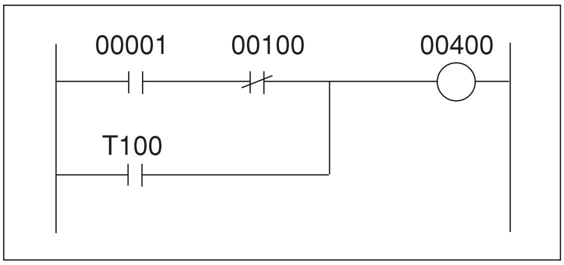
1) Data memory

- The data memory is assigned by relay numbers (input relay, output relay, auxiliary relay, and timer/counter contact points), and stores ON/OFF information.
- The ON/OFF information of the data memory is read by six basic instructions: STR, STR NOT, AND, AND NOT, OR, and OR NOT, and written by OUT instruction and timer/counter instruction.

(An example of writing instruction words)

```
STR           00001
AND NOT       00100
OR            T100
OUT           00400
↑               ↑
Instruction    Relay number
```

(An example of ladder chart)



2) Logical operation section

- Logical processing takes place according to the contents of the instruction.

3) Accumulator (ACC)

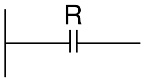
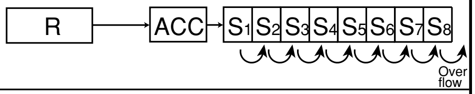
- This is a 1 bit register in which the result from the logical processing is stored.
- It changes with eight instructions: STR, STR NOT, AND, AND NOT, OR, OR NOT, AND STR, and OR STR.

4) Stack register (S_1, S_2, S_3, S_4, S_5, S_6, S_7, S_8)

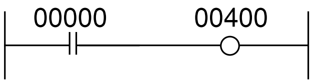
- This is an 8 bits register in which the intermediate result is stored during processing of the serial/parallel circuit or counter instruction, application instruction (F-60, F-60w, F-62, F62w) that has a plural number of input conditions.
- It changes with four instructions: STR, STR NOT, AND STR and OR STR.

[2] Description of each basic instruction

STR

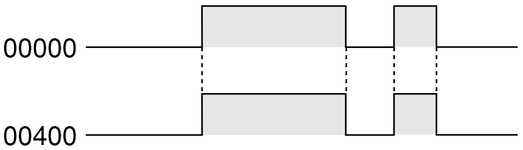
Symbol		
Function		Use when first contact point from bus line, or first contact point of circuit block is "a" contact.
Operation		Stores data memory contents (ON/OFF data) of relay number R into the accumulator (ACC). ON/OFF data previously registered in the ACC is transferred to S_1 of the stack register. S_1 data is transferred to S_2. Thereafter, S_2 to S_3, S_3 to S_4, S_4 to S_5, S_5 to S_6, S_6 to S_7, and S_7 to S_8, respectively. Data in S_8 is deleted. 
Range of R		00000 to 15777 T000 to T377 C000 to C377
After operation	Contents of R	Latch
	ACC	Contents of R
	S_1	Contents of ACC before operation
	S_2	Contents of S_1 before operation
	S_3	Contents of S_2 before operation
	S_4	Contents of S_3 before operation
	S_5	Contents of S_4 before operation
	S_6	Contents of S_5 before operation
	S_7	Contents of S_6 before operation
	S_8	Contents of S_7 before operation

[Example for use 1]

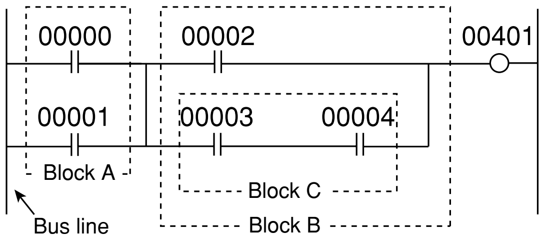


Instruction	
STR	00000
OUT	00400

When input relay 00000 is ON, output relay 00400 turns ON.

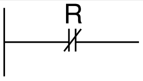
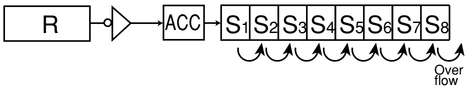


[Example for use 2]

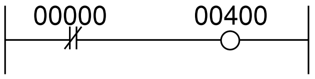


```
STR       00000 --------- First contact point of bus line
OR        00001           (block A)
STR       00002 --------- First contact point of block B
STR       00003 --------- First contact point of block C.
AND       00004
OR   STR
AND  STR
OUT       00401
```

STR NOT

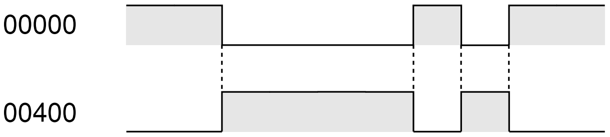
Symbol		
Function		Use when first contact point from bus line, or first contact point of circuit block is "b" contact.
Operation		Reverses data memory contents (ON/OFF data) of relay number R, and stores it in the accumulator (ACC). ON/OFF data previously registered in the ACC is transferred to S_1 of the stack register. S_1 data is transferred to S_2. Thereafter, S_2 to S_3, S_3 to S_4, S_4 to S_5, S_5 to S_6, S_6 to S_7, and S_7 to S_8, respectively. Data in S_8 is deleted. 
Range of R		00000 to 15777 T000 to T377 C000 to C377
After operation	Contents of R	Latch
	ACC	Value after reversed R contents.
	S_1	Contents of ACC before operation
	S_2	Contents of S_1 before operation
	S_3	Contents of S_2 before operation
	S_4	Contents of S_3 before operation
	S_5	Contents of S_4 before operation
	S_6	Contents of S_5 before operation
	S_7	Contents of S_6 before operation
	S_8	Contents of S_7 before operation

[Example for use 1]

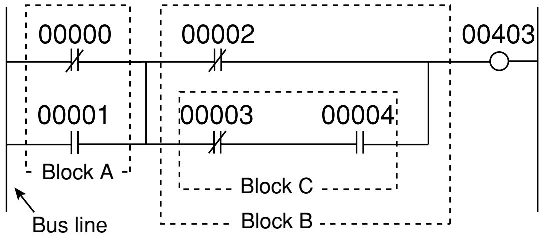


Instruction	
STR NOT	00000
OUT	00402

When input relay 00000 is OFF, output relay 00402 turns ON.

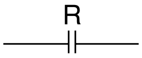
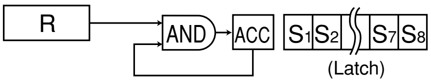


[Example for use 2]

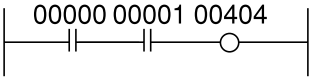


```
STR  NOT  00000········ First contact point of bus line
OR        00001         (block A)
STR  NOT  00002········ First contact point of block B
STR  NOT  00003········ First contact point of block C.
AND       00004
OR   STR
AND  STR
OUT       00403
```

AND

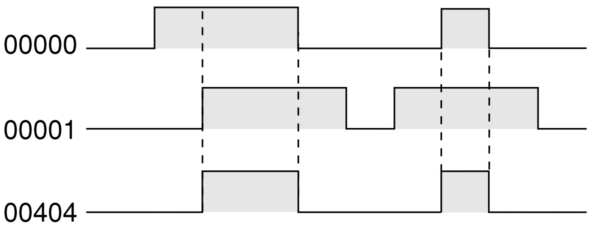
Symbol		
Function		Use when serial contact point is "a" contact.
Operation		AND operates data memory contents of relay number R (ON/OFF data) and contents of accumulator (ACC). Then, stores the result in the ACC. 
Range of R		00000 to 15777 T000 to T377 C000 to C377
After operation	Contents of R	Latch
	ACC	AND operated value of R contents and contents in the ACC before operation.
	S_1 to S_8	Latch

[Example for use]

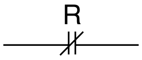
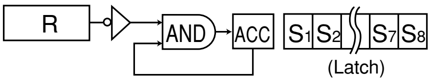


Instruction	
STR	00000
AND	00001
OUT	00404

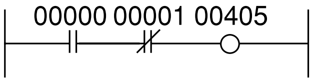
When input relay 00000 and 00001 are both ON, output relay 00404 turns ON.



AND NOT

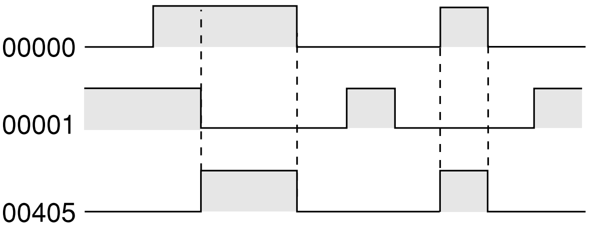
Symbol		
Function		Use when serial contact point is "b" contact.
Operation		Reverses data memory contents (ON/OFF data) of relay number R and AND operates with contents of accumulator (ACC). Then, stores the result in the ACC. 
Range of R		00000 to 15777 T000 to T377 C000 to C377
After operation	Contents of R	Latch
	ACC	AND operated reversed value of R contents and contents in the ACC before operation.
	S_1 to S_8	Latch

[Example for use]

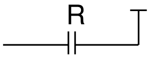
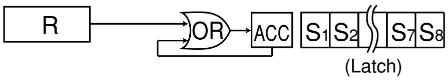


Instruction	
STR	00000
AND NOT	00001
OUT	00405

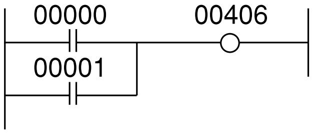
When input relay 00000 is ON and 00001 is OFF, output relay 00405 turns ON.



OR

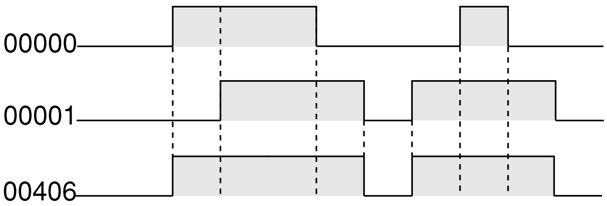
Symbol		
Function		Use when parallel contact point is "a" contact.
Operation		OR operates data memory contents of relay number R (ON/OFF data) and contents of accumulator (ACC). Then, stores the result in the ACC. 
Range of R		00000 to 15777 T000 to T377 C000 to C377
After operation	Contents of R	Latch
	ACC	OR operated value of R contents and contents in the ACC before operation.
	S1 to S8	Latch

[Example for use]

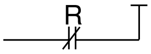
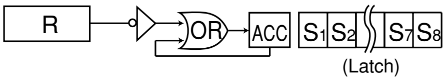


Instruction	
STR	00000
OR	00001
OUT	00406

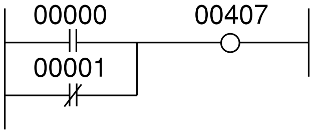
When input relay 00000 or 00001 is ON, output relay 00406 turns ON.



OR NOT

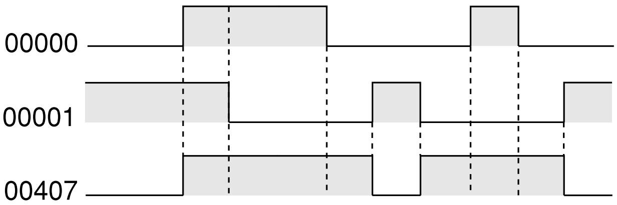
Symbol		
Function		Use when parallel contact point is "b" contact.
Operation		Reverses data memory contents of relay number R (ON/OFF data), and operates OR with contents of accumulator (ACC). Then, stores the result in the ACC. 
Range of R		00000 to 15777 T000 to T377 C000 to C377
After operation	Contents of R	Latch
	ACC	OR operated reversed value of R contents and contents in the before operation
	S1 to S8	Latch

[Example for use]

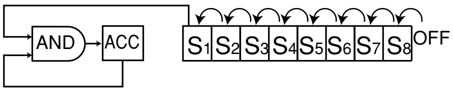


Instruction		
STR		00000
OR	NOT	00001
OUT		00407

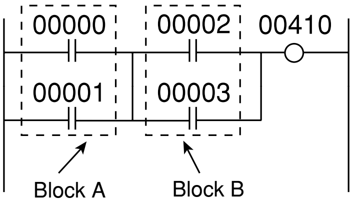
When input relay 00000 is ON, or 00001 is OFF, output relay 00407 turns ON.



AND STR

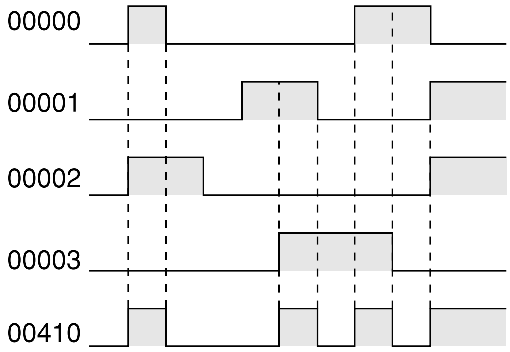
Function		Use to serially connect between circuit blocks.
Operation		AND operates stack register S_1 and accumulator (ACC) contents (ON/OFF data). Then, stores data in the ACC. Previously stored ON/OFF data in S_2 is transferred to S_1. Data in S_3 is transferred to S_2. Thereafter data is shifted to S_4 to S_3, S_5 to S_4, S_6 to S_5, S_7 to S_6, S_8 to S_7. In S_8, OFF data is stored. 
After operation	ACC	AND operated value of S_1 contents before operation and ACC contents.
	S_1	Contents of S_2 before operation
	S_2	Contents of S_3 before operation
	S_3	Contents of S_4 before operation
	S_4	Contents of S_5 before operation
	S_5	Contents of S_6 before operation
	S_6	Contents of S_7 before operation
	S_7	Contents of S_8 before operation
	S_8	OFF (0)

[Example for use 1]

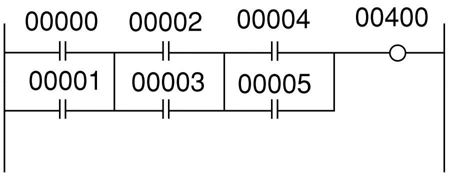


Instruction	
STR	00000
OR	00001
STR	00002
OR	00003
AND STR	
OUT	00410

Connect block A and block B in series.
If input relay 00000 or 00001 is ON, and input relay 00002 or 00003 is ON, output relay 00410 turns ON.



Note 1: The following two methods (a) and (b) are available to program the ladder chart below.



(a)

```
STR      00000
OR       00001
STR      00002
OR       00003
AND  STR
STR      00004
OR       00005
AND  STR
OUT      00400
```

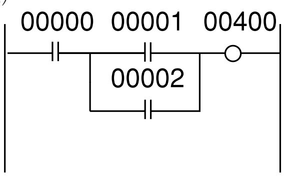
(b)

```
STR      00000
OR       00001
STR      00002
OR       00003
STR      00004
OR       00005
AND  STR
AND  STR
OUT      00400
```

Both (a) and (b) give the same result. However, (a) uses only S_1 stack, and (b) uses stacks S_1 and S_2. The JW10 has eight stacks so programming like (b) can only be connected at a maximum nine of blocks.

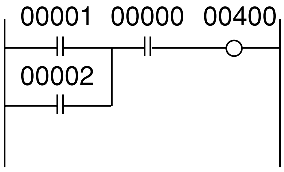
Note 2: Operation contents of both (a) and (b) below are the same.

(a)



```
STR      00000
STR      00001
OR       00002
AND  STR
OUT      00400
```

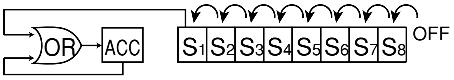
(b)



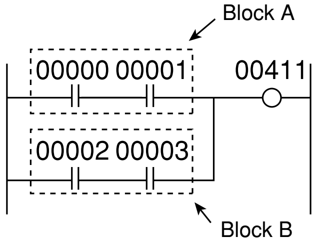
```
STR      00001
OR       00002
AND      00000
OUT      00400
```

Case (b) is one step smaller than case (a).

OR STR

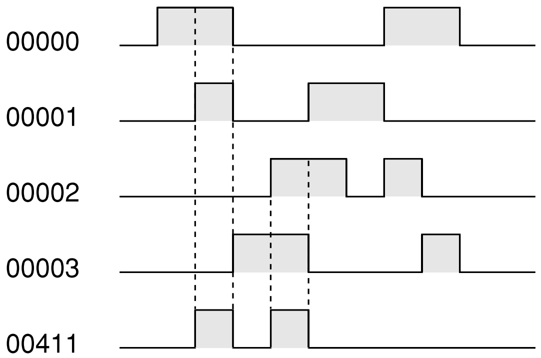
Function		Use to parallel connect between circuit blocks.
Operation		OR operates stack register S_1 and accumulator (ACC) contents (ON/OFF data). Then, stores data in the ACC. Previously stored ON/OFF data in S_2 is transferred to S_1. Data in S_3 is transferred to S_2. Thereafter data is shifted to S_4 to S_3, S_5 to S_4, S_6 to S_5, S_7 to S_6, S_8 to S_7. In S_8, OFF data is stored.  OR ACC S1 S2 S3 S4 S5 S6 S7 S8 OFF
After operation	ACC	OR operated value of S_1 contents before operation and ACC contents.
	S_1	Contents of S_2 before operation
	S_2	Contents of S_3 before operation
	S_3	Contents of S_4 before operation
	S_4	Contents of S_5 before operation
	S_5	Contents of S_6 before operation
	S_6	Contents of S_7 before operation
	S_7	Contents of S_8 before operation
	S_8	OFF (0)

[Example for use]

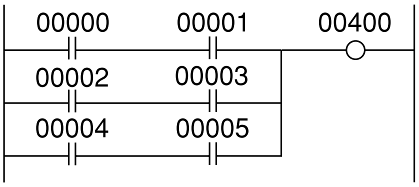


Instruction	
STR	00000
AND	00001
STR	00002
AND	00003
OR STR	
OUT	00411

Connect block A and block B in parallel.
When both input relays 00000 and 00001, or both input relays 00002 and 00003 are ON, output relay 00411 turns ON.



Note 1: The following two methods (a) and (b) are available to program the ladder chart below.



(a)

STR		00000
AND		00001
STR		00002
AND		00003
OR	STR	
STR		00004
AND		00005
OR	STR	
OUT		00400

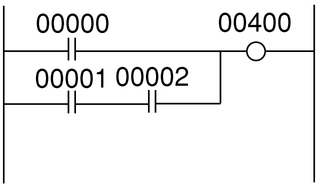
(b)

STR		00000
AND		00001
STR		00002
AND		00003
STR		00004
AND		00005
OR	STR	
OR	STR	
OUT		00400

Both (a) and (b) give the same result. However, (a) uses only S_1 stack, and (b) uses stacks S_1 and S_2. The JW10 has eight stacks so programming like (b) can only be connected at a maximum nine of blocks.

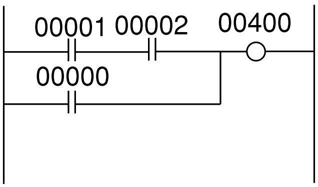
Note 2: Operation contents of both (a) and (b) below are the same.

(a)



STR		00000
STR		00001
AND		00002
OR	STR	
OUT		00400

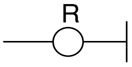
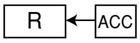
(b)



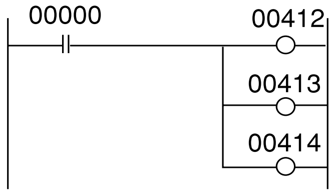
STR	00001
AND	00002
OR	00000
OUT	00400

Case (b) is one step smaller than case (a).

OUT

Symbol		
Function		Use for output of operation result.
Operation		Write contents of the accumulator (ACC) into data memory of relay number R. 
Range of R		00400 to 15777 (Note 1) (Note 2)
After operation	Contents of R	Contents ofACC
	ACC	Latch
	S_1 to S_8	Latch

[Example for use]



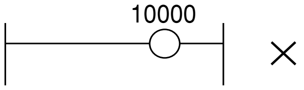
Instruction	
STR	00000
OUT	00412
OUT	00413
OUT	00414

When input relay 00000 is ON, the JW10 turns ON output relay 00412, 00413, and 00414.
(OUT instruction does not change ACC contents so that sequential use of OUT instruction is possible.)

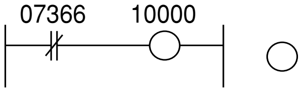
Note 1: Module-not-mouted input relay area (00000 to 00377) are also used for OUT instruction as auxiliary relay.

Note 2: Be careful because some special relays and special registers cannot be used for OUT instruction. (See "Special relay" on page 7・3 and "special register" on page 7・5.)

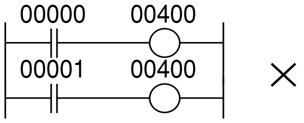
Note 3: OUT instruction cannot be started directly from



For relays are normally to be kept ON, use normally OFF contact (07366).



Note 4: If the same relay number OUT instruction is used twice, the connected support tool such as JW-13PG displays error at program check.

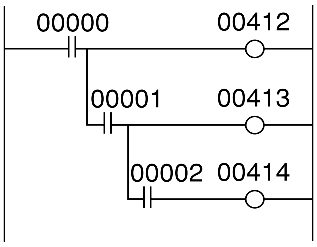


If the JW10 is changed to operation mode after inputting the program above, it detects "user program error 1" by self diagnosis function and cannot operate.

However, if a 55(H) is stored in register #055 in the system memory, the JW10 can continue the operation.(The option is available in version 1.4 or later ROM code.)

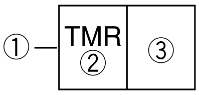
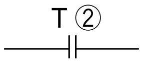
In this case, the result of the last step in the program ("b" in this example) will be sent to the output section.

Note 5: As the contents of the ACC do not change after operating OUT instruction, the following ladder program is effective.



STR	00000
OUT	00412
AND	00001
OUT	00413
AND	00002
OUT	00414

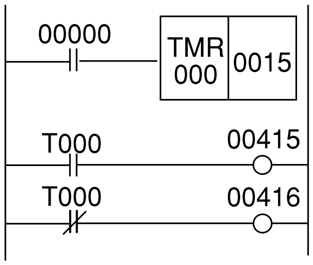
TMR

Symbol	Timer instruction 	Timer contact 
Function	When start input① is turned ON, current value is decrement by one from the setting value ③ every 0.1 second (TMR number ② = 000 to 277) or 0.01 second (TMR number ② = 300 to 377). When current value becomes 0, the timer contact turns	

Operation	Start input	Current value	TMR contact
	OFF	Setting value	OFF
	ON (Current value>0)	Decrement by 1 every 0.1 sec. or 0.01 sec.	OFF
	ON (Current value = 0)	0	ON

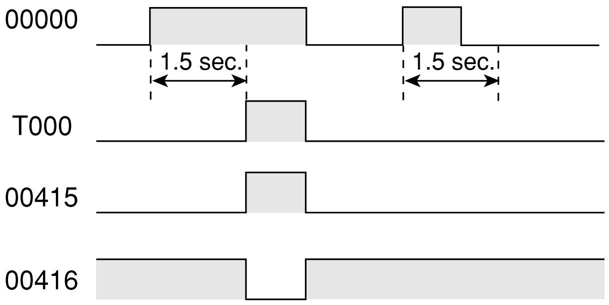
Condition	Start when stat input ① is ON
Range of TMR number ②	000 to 277 (8) [0.1 sec. timer] (Note 1) 300 to 377 (8) [0.01 sec. timer]
Range of setting value ③	0001 to 1999 (BCD) (0.1 to 199.9sec. : TMR000 to TMR277 0.01 to 19.99sec. : TMR300 to TMR377) コ0000 to コ1576, b0000 to b0376 09000 to 09776, 19000 to 19776 29000 to 29776, 39000 to 39776 (Note 4)
Precision (unit : sec.)	Setting value $^{+0}_{-0.01}$ + scan time (0.1 sec. timer) Setting value $^{+0}_{-0.01}$ + scan time (0.01 sec. timer)
After operation ACC	Latch (Start input ①)
S$_1$ to S$_8$	Latch

[Example for use]



Instruction	
STR	00000
TMR	000
	0015
STR	T000
OUT	00415
STR NOT	T000
OUT	00416

After start input 00000 turns from OFF to ON, timer contact T000 turns ON after 1.5 sec., and output relay 00415 turns ON and 00416 turns OFF.



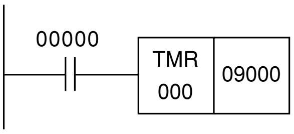
When start input 00000 is turned OFF, TMR000 is reset.

Note 1: TMR number is used commonly with CNT so that numbers used for CNT cannot be used for TMR. The same TMR number also cannot be used twice. If the same number is used, "user program error 1" occurs by self diagnosis and cannot run.
However, if a 55(H) is stored in register #055 in the system memory, the JW10 can continue the operation.(The option is available in version 1.4 or later ROM code.) To check assigned condition of the same number, use "program check" function of a support tool such as JW-13PG.

Note 2: Assign timer contact with the same number of timer numbers. Settable for any number of "a"

Note 4: When resister number is used for setting value, the setting value can be changed by program operation. Make sure to assign even number address for register.

(Example)



09001		09000		
0 0 0 1	0 1 0 0	0 1 0 0	0 1 1 0	Set value:1446
1	4	4	6	

Be careful not to enter other than BCD values or over 2000, as these may cause malfunction.

- **Timer's current value storage area of timer**

Current value of the timer is stored in 512 bytes from b0000 to b0777 as follow.

Using this area, current value of the timer can be output to an external device, and setting value can be changed from outside.

TMR number	Data storage area
000	b0000, b0001
001	b0002, b0003
002	b0004, b0005
003	b0006, b0007
⋮	⋮
376	b0774, b0755
377	b0776, b0777

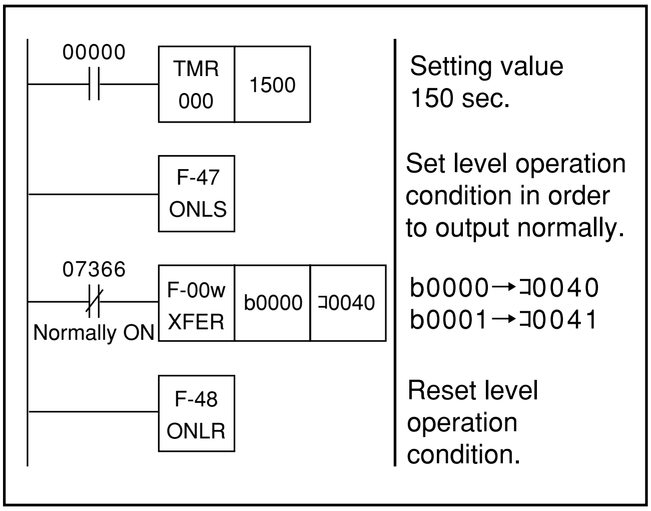
Note 5: Area from b0000 to b0777 are commonly used with the counter (CNT).

	7	6	5	4	3	2	1	0	
TMR000 to TMR277	8	($\times 10^0$) 4	2	1	8	($\times 10^{-1}$) 4	2	1	n
	0	0	0	($\times 10^2$) 1	8	($\times 10^1$) 4	2	1	n+1
TMR300 to TMR377	8	($\times 10^{-1}$) 4	2	1	8	($\times 10^{-2}$) 4	2	1	n
	0	0	0	($\times 10^1$) 1	8	($\times 10^0$) 4	2	1	n+1

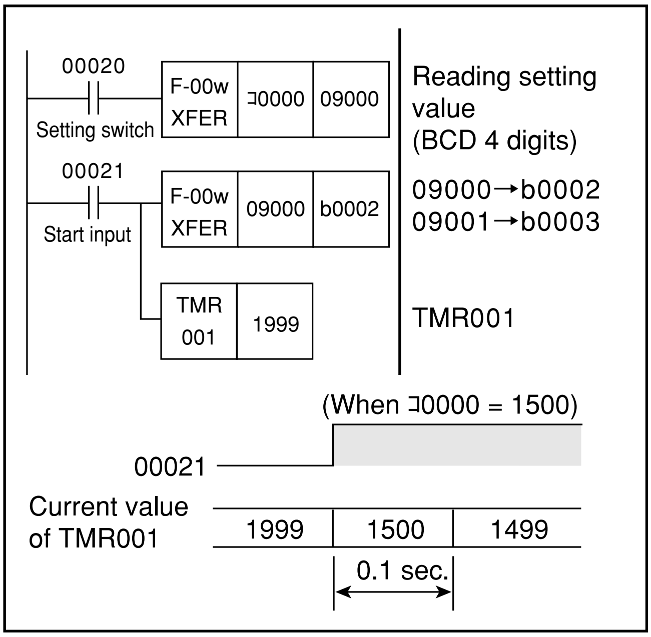
Note 6: Address b0000 to b0777 are handled numerical values with BCD.

Note 7: "n" and "n + 1" express address order.
For example, in case of TMR001, n = b0002 and n + 1 = b0003.

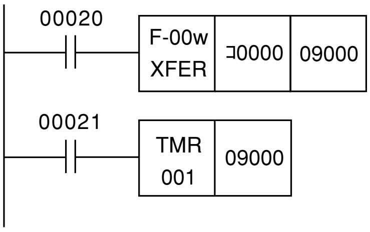
(1) An example of program to output timer current value to an external device.



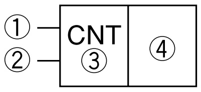
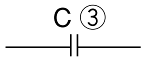
(2) An example of program to input timer setting value from an external device.



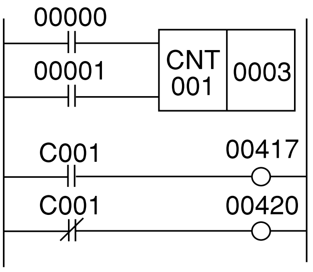
If register number is assigned to setting number, the JW10 runs with the same pattern of the program example above.



CNT

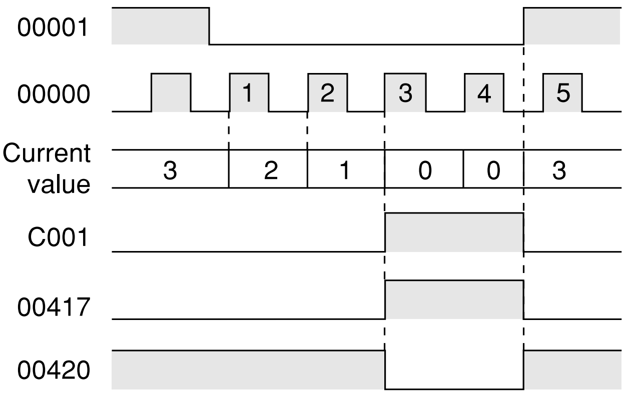
Symbol	Counter instruction  	Counter contact  	
Function	When reset input ② is OFF, current value is decrement by one from the setting value ④ at each change of calculation input ① from OFF to ON. When current value reaches 0, the counter contact turns ON. When reset input ② is ON, the JW10 does not calculate and keeps current value as setting value, and counter contact is		
Operation	Start input	Current value	TMR contact
	OFF	Setting value	OFF
	ON (Current value>0)	Decrement by 1 at each turn ON from OFF of calculation input	OFF
	ON (Current value = 0)	0	ON
Condition	When reset input ② is OFF (Note 6) and calculation input ① changes from OFF to ON.		
Range of CNT number ③	000 to 377 (8)		
Range of setting value ④	0001 to 1999 (BCD) コ0000 to コ1576, b0000 to b0376 09000 to 09776, 19000 to 19776 29000 to 29776, 39000 to 39776 (Note 7)		
After operation ACC	Latch (Reset input ②)		
S1	Latch (Calcultaion input ①)		
S2 to S8	Latch		

[Example for use]



Instruction	
STR	00000
STR	00001
CNT	001
	0003
STR	C001
OUT	00417
STR NOT	C001
OUT	00420

When the reset input 0001 is OFF and calculation input 00000 switches ON from OFF three times repeatedly, counter contact C001 turns ON and output relay 00417 turns ON and 00420 turns OFF.



Note 1: CNT number is used commonly with TMR so that numbers used for TMR cannot be used for CNT. The same CNT number also cannot be used twice. If the same number is used, "user program error 1" occurs by self diagnosis and cannot run.
However, if a 55(H) is stored in register #055 in the system memory, the JW10 can continue the operation.(The option is available in version 1.4 or later ROM code.)
To check assigned condition of the same number, use "program check" function of a support tool such as JW-13PG.

Note 2: Assign counter contact with the same number of counter numbers. Settable for any number of "a" contact and "b" contact.

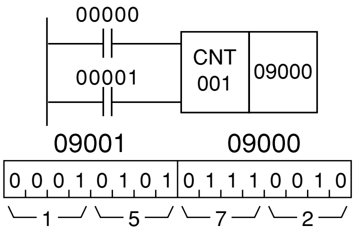
Note 3: When the JW10 counts up, it ignores further input. To restart calculation, turn reset input ON and then OFF. Or, forcibly reset using a support tool, and then start calculation.

Note 5: At power failure, the counter stores current value. However, if reset input turns ON at reinputting power, it resets current value. If you want to store current value, add reset input that turns OFF at inputting power.

Note 6: Reset input also can be set as "reset by OFF" by setting reset condition (001) on system memory #202.

Nore 7: When register number is used for setting value, the setting value can be changed by program operation. Make sure to assign even number address for register number.

(Example)



Set value:1572

Be careful not to enter other than BCD values or over 2000, as these may cause malfunction.

- **Counter's current value storage area of counter**

  Current value of the counter is stored in 512 bytes from b0000 to b0777 as follow.

  Using this area, current value of the counter can be output to an external device, and setting value can be changed from outside.

CNT number	Data storage area
000	b0000, b0001
001	b0002, b0003
002	b0004, b0005
003	b0006, b0007
⋮	⋮
376	b0774, b0775
377	b0776, b0777

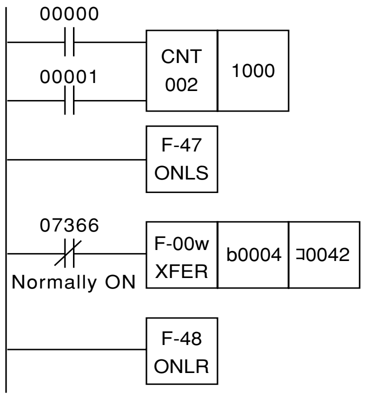
Note 8: Area from b0000 to b0777 are commonly used with the timer (TMR).

	7	6	5	4	3	2	1	0	
CNT000 to CNT377	(×10^1) 8	4	2	1	(×10^0) 8	4	2	1	n
	0	0	0	(×10^3) 1	(×10^2) 8	4	2	1	n+1

Note 9: Address b0000 to b0777 are handled numerical values with BCD.

Note 10: "n" and "n + 1" express address order. For example, in case of CNT001, n = b0002 and n + 1 = b0003.

(1) An example of program to output counter current value to an external device.



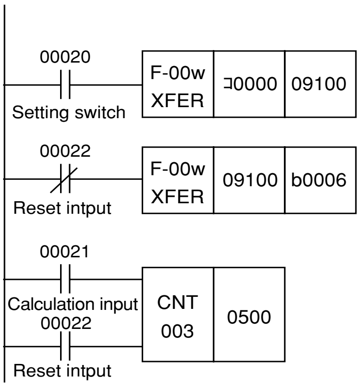
Setting value 1000

Set level operation condition in order to output normally.

b0004→コ0042
b0005→コ0043

Reset level operation condition.

(2) An example of program to input counter setting value from an external device.

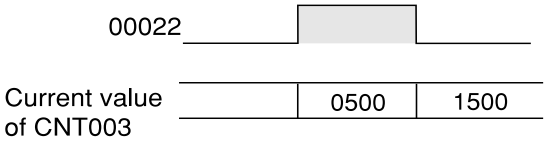


Reading setting value (BCD 4 digits)

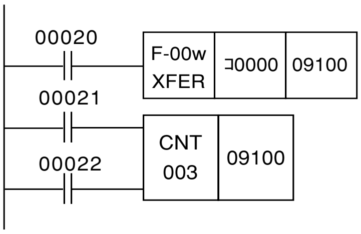
09100→b0006
09101→b0007

CNT003

When system memory #202 is 00(H) (reset by turning ON reset input), current value is changed at down edge of 00022. (when コ0000 = 1500)



If register number is assigned to set number, the JW10 runs with the same pattern of the program example above.



9-3 Application Instruction

[1] Function number

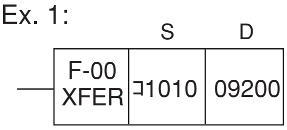
- All application instructions are expressed by function numbers.
- In addition, among transfer instruction, arithmetic operation instruction, logical operation instruction, comparison instruction etc., instructions for handling word unit or constants lower case "w" and "c" are added to each function number.

Instruction to operate between registers (byte unit)	F00, F-10, F-12, F-13 etc.
Instruction to operate between registers (word unit)	F-00w, F-10w, F-12w, F-13w etc.
Instruction to operate between register and constants (byte unit)	Fc10, Fc12, Fc13 etc.
Instruction to operate between register and constants (word unit)	Fc10w, Fc12w, Fc13w etc.

[2] Source and destination

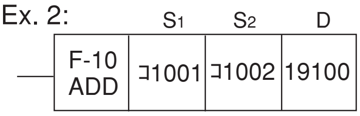
- Data processing instructions such as transfer instruction and arithmetic operation instruction handle data memory with byte unit or word unit.
- A register having data before operation is called source (S). A register to store operation result is referred to as destination (D).

Ex. 1:



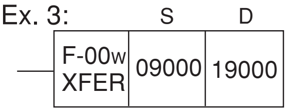
Transfer contents of コ1010(S) to 09200(D).

Ex. 2:



Adds contents of コ1001(S_1) and コ1002(S_2), and stores the result in 19100(D).

Ex. 3:

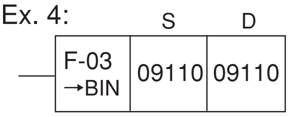


Store contents of 09000(S) and 09001(S+1) to 19000(D) and 19001 (D+1).

Note 1: Make sure to set even number address for source and destination of word processing instruction (instruction having "w").

Note 2: Register contents of source side do not change after operation. Use of the same register both for source and destination is available. However, in this case, source contents may change for some instruction (that is contents of destination).

Ex. 4:



Converts contents of 09110(S) (2 digits of BCD) into binary code and stores in 09110 (D).

Note 3: Among special relay and special register area, do not assign destination to area where CPU writes (such as コ0734 to コ0737).

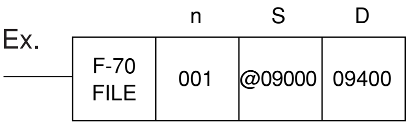
Note 4: If source and destination are assigned which cross over blocks of data memory for block process instruction (F-70, F-70w, F-71, F-71w, F-74, F-74w), these instructions will be as shown in the table at right. Especially, be careful not to write data in timer/counter contact area and CPU internal processing area.

S, D	S+1, D+1
コ1577	Timer/counter contact area
b0777	CPU internal processing area
09777	19000
19777	29000
29777	39000
39777	CPU internal processing area

[3] Indirect address assignment

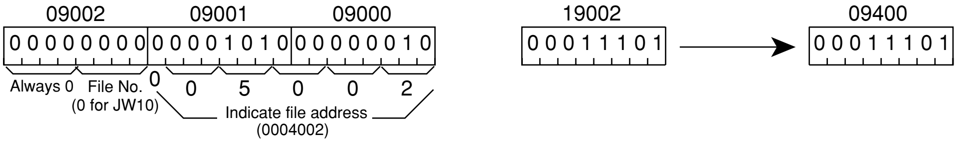
- F-70 and F-70w can assign indirect address for their source and destination.
- If source or destination is assigned with indirect address, the assigned register itself does not execute operation. Registers of file address assigned by contents of three bytes headed by its register executes operation. (For file address, see page 7・8 "File address")
- To assign indirect address, put @ (at mark) at the top of register.

Ex.



Transfer register contents assigned by 09000, 09001, and 09002 to 09400.

In the example above, file address 005002 is register 19002. As a result, @09000 indicates 19002.

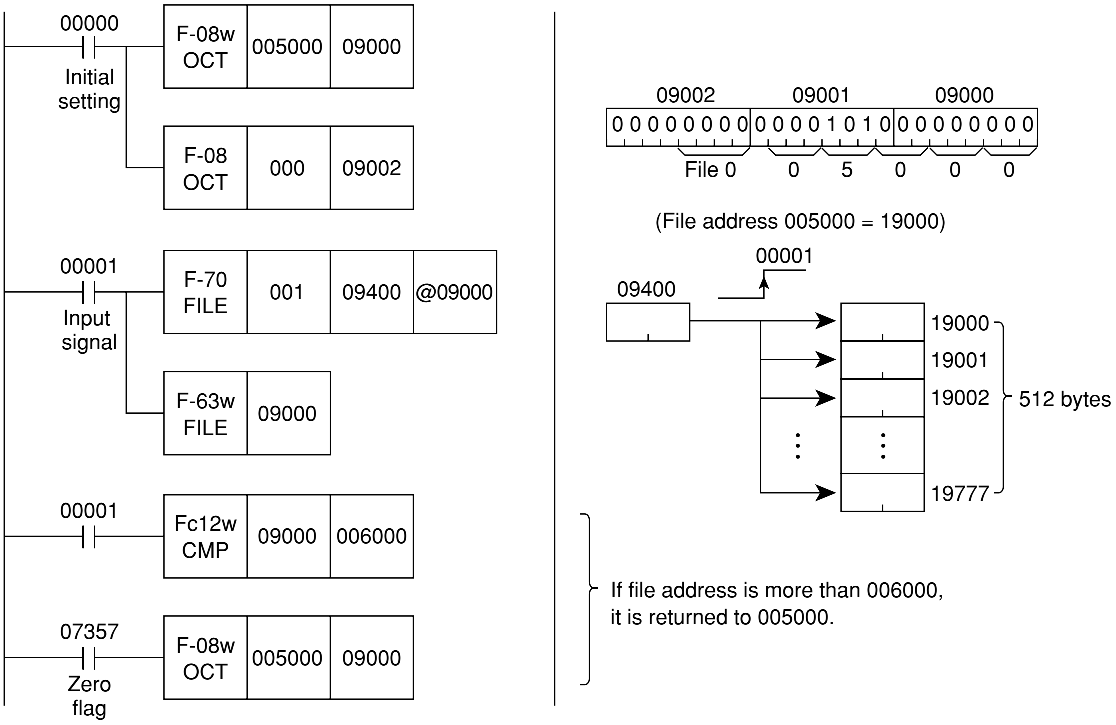


Note 1: For assigning indirect address, make sure to set even number address. One will be deducted if odd address number is set and changed to even address. (If @09003 is set, it is treated as @09002.)

Note 2: Timer/counter setting area (001600 to 001677) of file address cannot be assigned as indirect address. Further, addresses 001700 to 001777, 003000 to 003777, and 010000 and after are used inside the CPU. These areas cannot be used by users. Therefore they cannot be set as indirect address.

Reference: Examples of using indirect address

- Each time input signal changes from OFF to ON, the program stores data of register 09400 to 512 bytes from 19000 to 19777 in order.

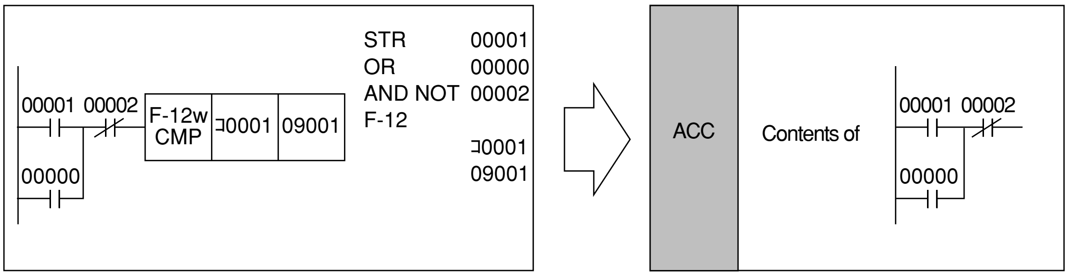


[4] Accumulator and stack register

Use accumulator (ACC) and stack register with basic instructions such as STR and STR NOT. (See “Operation of basic instruction” in page 9･8.) However, application instructions execute operations using these as operation conditions.

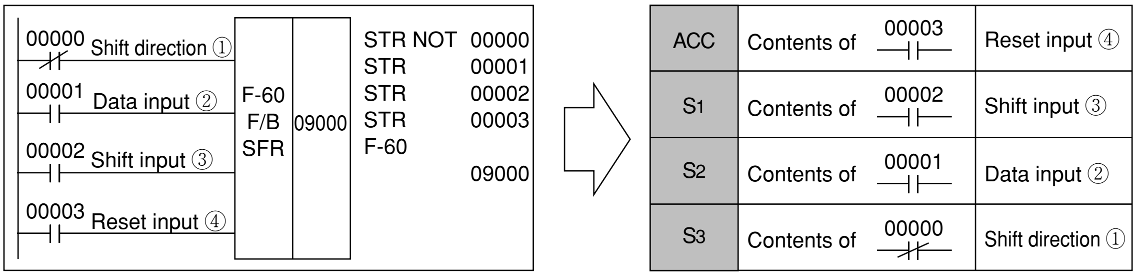
1) Single input instruction

- Execute only the contents of the accumulator (ACC) as operation condition.
- The contents of ACC and stack register are stored after operation.



2) Multiple input instruction

- F-62 and F-62w use the accumulator (ACC) and stack register S1 and S2 as they are three input instructions.
- F-60 and F-60w use the accumulator (ACC) and stack register S1, S2 and S3 as they are four input instructions.
- After operation, contents of ACC and stack register are latched.



3) Instruction without input

- Instructions F-31, F-40, F-42, F-47, F-48, F-90, F-140, F-143, and F-145 do not have input condition so that they do not use the accumulator (ACC) and stack register.
- After operation, contents of ACC and stack register are latched.

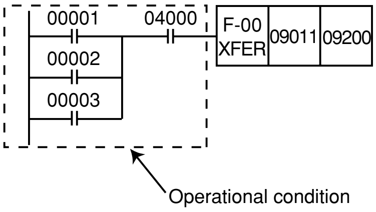
4) Instructions with which status of the accumulator (ACC) changes

- F-43 reverses ACC condition just before (OFF to ON, ON to OFF).
- F-44 turns ON the ACC for only one scan time just before the ACC changes from OFF to ON. The ACC turns OFF cases other than switching from OFF to ON.
- F-45 turns ON the ACC for only one scan time just before the ACC changes from ON to OFF. The ACC turns OFF cases other than switching from OFF to ON.
- After operation, contents of ACC and stack register are latched.

[5] Operational condition

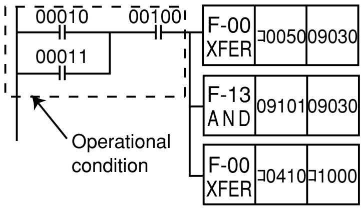
- Operational condition of an application instruction can be a complicated serial/parallel circuit, not limited to a single contact ON/OFF condition.
- If the operational condition is same, it may be programmed in succession, as shown in Example 2).

Example 1)



Instruction	
STR	00001
OR	00002
OR	00003
AND	04000
F-00	
	09011
	09200

Example 2)



Instruction	
STR	00010
OR	00011
AND	00100
F-00	
	コ0050
	09030
F-13	
	09101
	09030
F-00	
	コ0410
	コ1000

Note 1: In case where operational condition is the same, and some instructions vary operation results between programming continually and programming by separated circuits. See page 9・26, "Double-length operation."

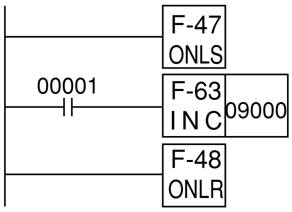
- There are following two kinds of execution modes for application instruction to be executed when the operational condition is met.

① Those which operation takes place only at the 1 scan cycle that the operational condition is met.
F-00, F-10 etc.

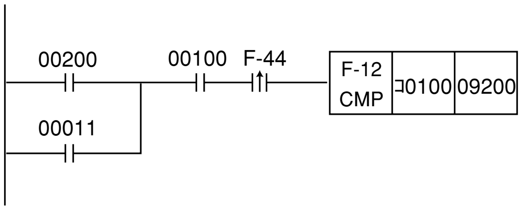
② Those which operation takes place at each scan cycle, so long as the operational condition is met.
F-12, F-80 etc.

- For the instruction of the group ① above, the ON/OFF state of the operational condition of the preceding scan cycle is compared with the ON/OFF state of the operational condition at the current scan cycle. If the preceding state is OFF and the current state is ON, the operation takes place as if the operational condition changed from OFF to ON.

- To execute at every scan cycle for the group of ①, use the F-47(ON level set) or F-48(ON level reset).



- In case the operation is to be executed only at an OFF to ON transition of the operational condition of the group of ②, the F-44 instruction (rise edge differential instruction) must be used.



- If the operational condition is not met, no operation takes place and the register on the destination side remains unaffected. In case of the instruction that affects the flag, the flag is cleared (See the next page for the flag.)

[6] Operation flag

1) Kind of flags

- Flag is used to reflect the operational result. There are 4 kinds of flags for the JW10 ; non-carry, error, carry, and zero flag. These are assigned to four bits of data memory, 07354 to 07357.

Non-carry flag	07354
Error flag	07355
Carry flag	07356
Zero flag	07357

2) Instruction that affects flag

Instruction such as F-03, F-12, F-63 etc. will set flag according to the result.

3) Transition of flag during scan cycle

- Prior to processing of the user program at each scan cycle, flags are cleared. (Refer to page 8･1 "Operation cycle".)

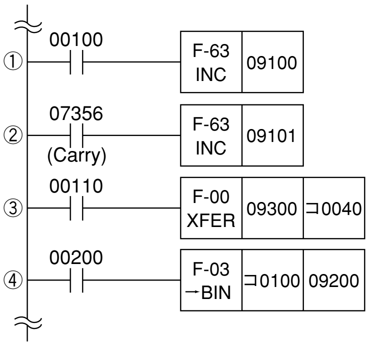
- When going into processing of a flag affecting instruction

a.The flag is set according to the result of the instruction, if the operational condition is met for the instruction.

b.Flag is cleared if the operational condition is not met for the instruction.

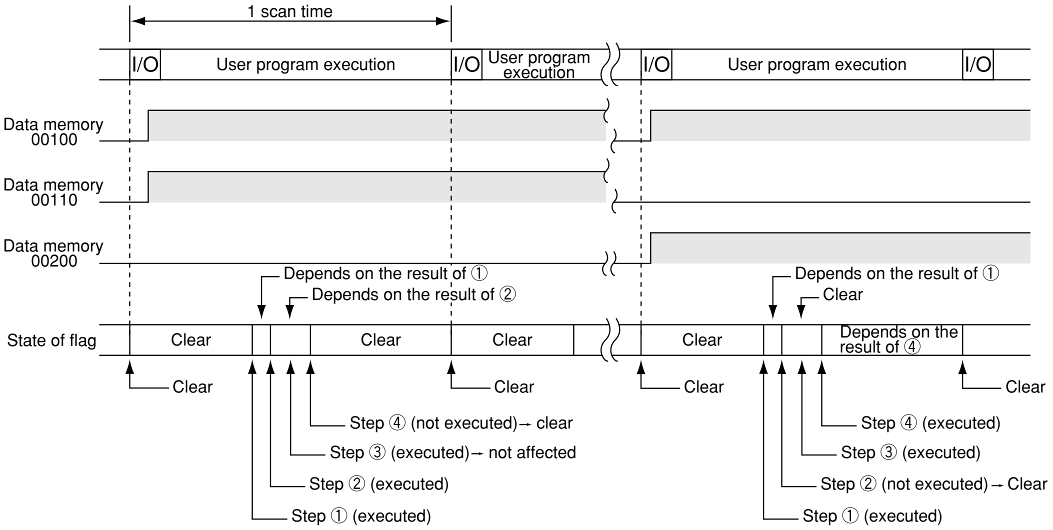
- As to the execution of no flag affecting instruction, the state of flag is not changed, whether the operation took place or not.

(If there is no flag affecting instructions in preceding steps.)



(If there is no flag affecting instructions in steps to follow.)

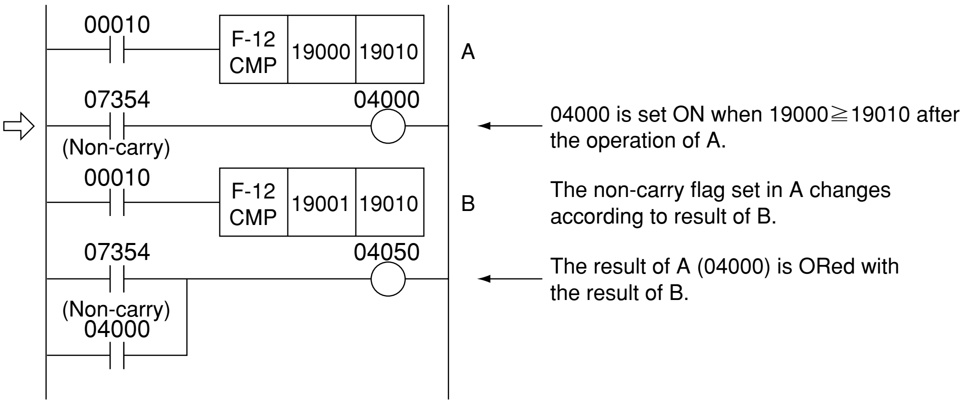
Address	Instruction		
01000	STR	00100	①
01001	F-63		
01002		09100	
01003	STR	07356	②
01004	F-63		
01005		09101	
01006	STR	00110	③
01007	F-00		
01010		09300	
01011		⊐0040	
01012	STR	00200	④
01013	F-03		
01014		⊐0100	
01015		09200	



4) How to retain flag

- Flag thus set according to the result may be changed or cleared after the operation of a flag affecting instruction in that scan cycle. When it goes into a next scan cycle, the flag is cleared before operation of the user program.
- In case the flag has to be retained of its current state, you must write the state of the flag into the coil (such as auxiliary relay) immediately following the respective instruction. This retains the state of the flag until the operation of the respective instruction in the next scan cycle.

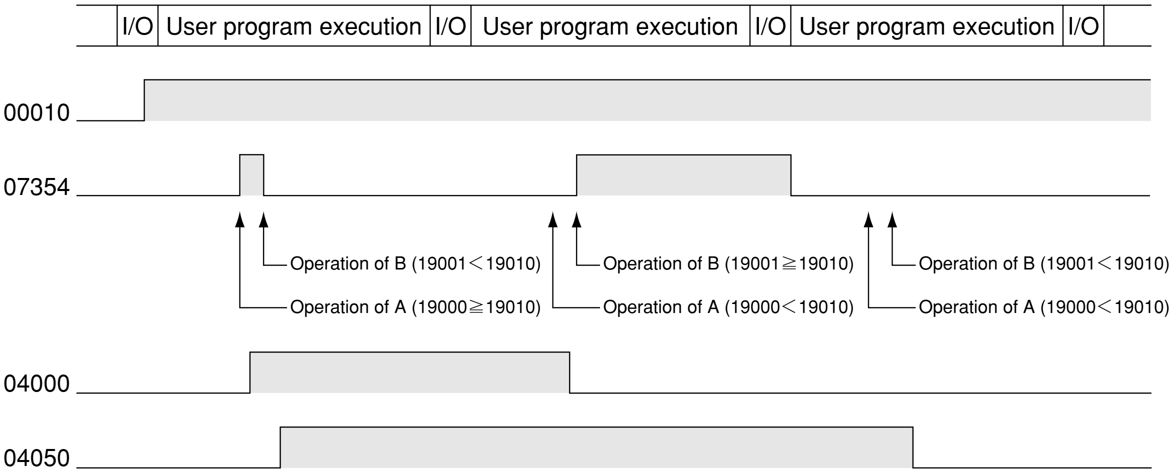
Example: Program that sets 04050 ON when 19000 ≧ 19010 or 19001 ≧ 19010.



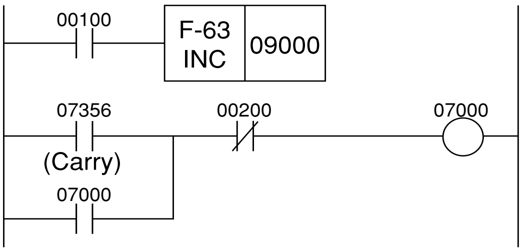
04000 is set ON when 19000≧19010 after the operation of A.

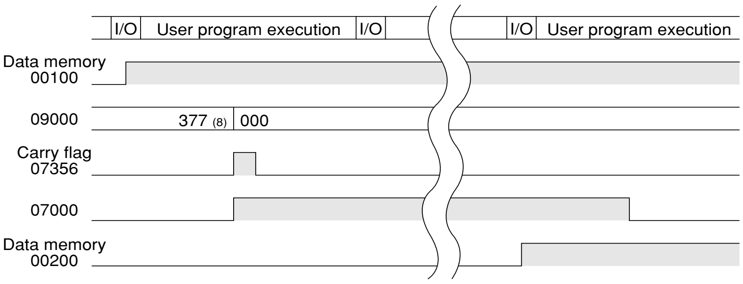
The non-carry flag set in A changes according to result of B.

The result of A (04000) is ORed with the result of B.


I/O
User program execution
I/O
User program execution
I/O
User program execution
I/O
00010
07354
Operation of B (19001<19010)
Operation of A (19000≧19010)
Operation of B (19001≧19010)
Operation of A (19000<19010)
Operation of B (19001<19010)
Operation of A (19000<19010)
04000
04050

- To monitor the state of flag by the support tool such as hand-held programmer JW-13PG etc. or to display it externally, it is not possible to visually check the state of the flag because the state of the flag can only be retained for a single scan cycle by simply writing the state of the flag in the coil as in example above.
- In such a case, it will be necessary to make the flag self-retained.





The state of the carry flag (07356) after the operation of A can be self-retained until 00200 is set ON.

[7] Double-length operation

1) Instructions that are capable of double-length operation

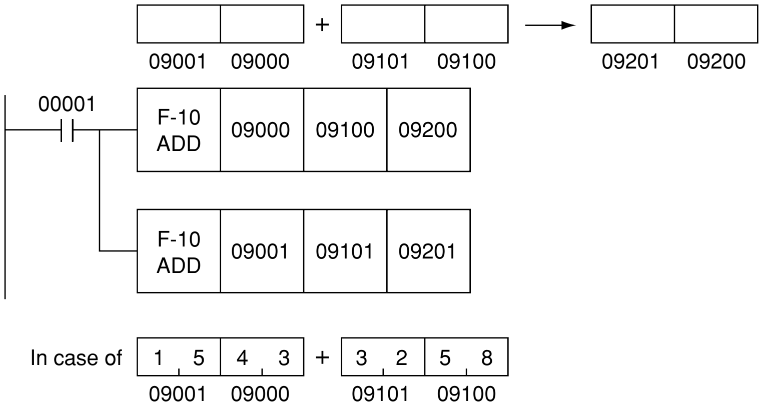
- The following 12 instructions have the double-length operation function to permit operation of more than 2 bytes (or 4 bytes for a word execution instruction).

① F-10, F-10w	: add register and register
② Fc10, Fc10w	: add register and BCD constant
③ F-11, F-11w	: subtract register from register
④ Fc11, Fc11w	: subtract BCD constant from register
⑤ F-12, F-12w	: compare register with register
⑥ Fc12, Fc12w	: compare register with constant

2) Programming double-length operation

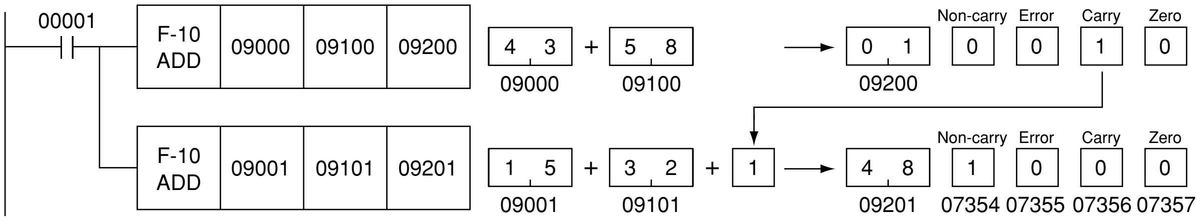
- Double-length operation is utilized to reflect a carry over or a carry down in a next digit to follow. It should be programmed from a low order digit following to the operational condition.

Example 1)



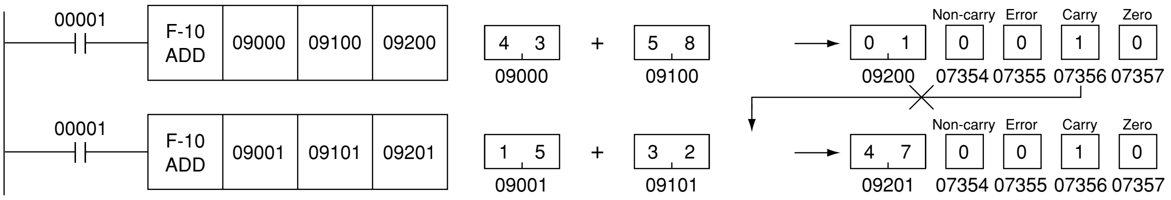
09001 09000 + 09101 09100 → 09201 09200
00001
F-10 ADD 09000 09100 09200
F-10 ADD 09001 09101 09201
In case of 1 5 4 3 + 3 2 5 8
09001 09000 09101 09100

Instruction	
STR	00001
F-10	
	09000
	09100
	09200
F-10	
	09001
	09101
	09201

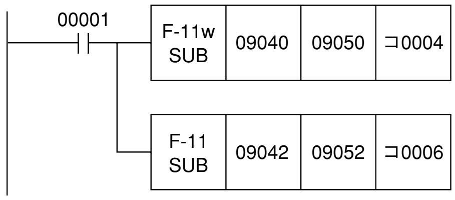


The carry flag in a lower digit is added to a next higher digit during the operation.

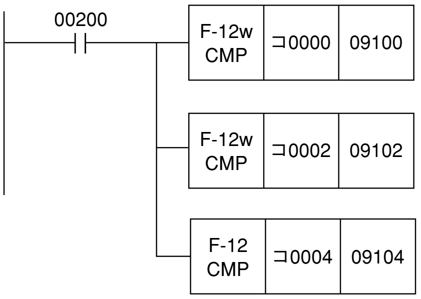
Reference: Programming in the following way would not perform the double-length operation.



Example 2) Double-length operation about 3 bytes is also possible in the same manner.



(Subtraction of BCD 6 digits)



(Comparison of 6 words data)

3) Internal processing during the double-length operation

- For operation of double-length operational instruction that appears first after the operational condition, the operation takes place without including the state of the flag before that.
- If there is double-length operational instruction encountered during execution of the common operational condition, the following operation takes place.

a. Execution takes place including the state of the carry flag immediately before.

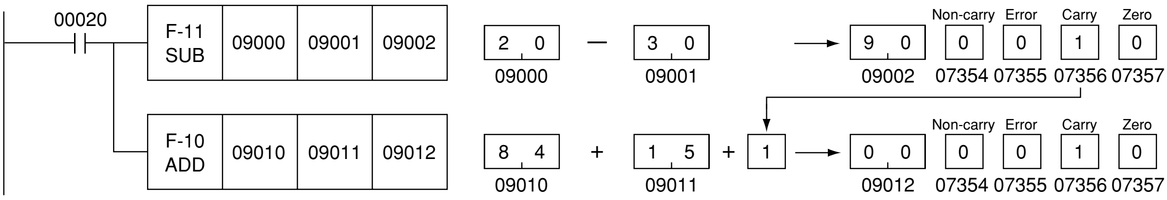
b. For the zero flag, the state of the zero flag immediately before is ANDed with the state ofthe zero flag after the operation of the respective instruction. If both are 1, the zero flag is then set.

F-10, F-10w Fc10, Fc10w	The state of the carry flag immediately before is added
F-11, F-11w Fc11, Fc11w	The state of the carry flag immediately before is subtracted
F-12, F-12w Fc12, Fc12w	The state of the carry flag immediately before is subtracted

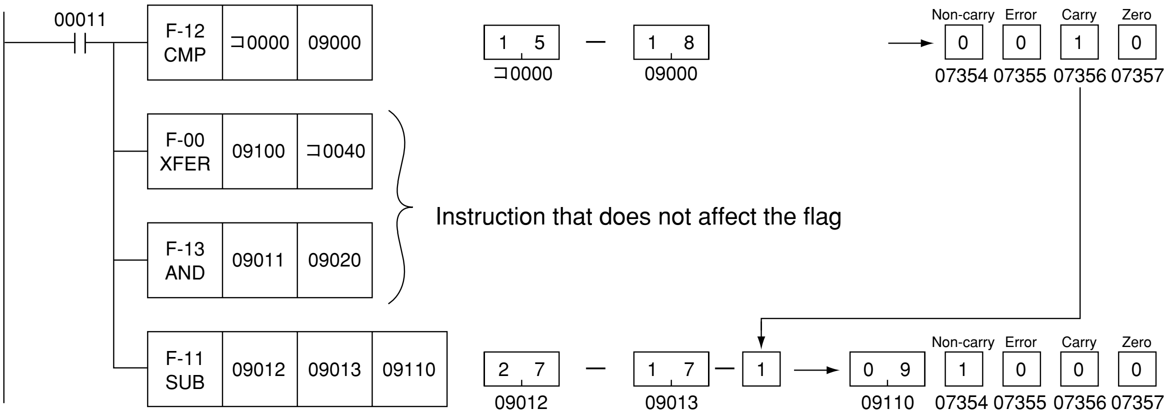
Note 1: For the F-12, F-12w, Fc12 and Fc12w instructions, operation of S1-S2 or S1-n is carried out and its result is stored in the flag.

4) Cautions concerning the double-length operation

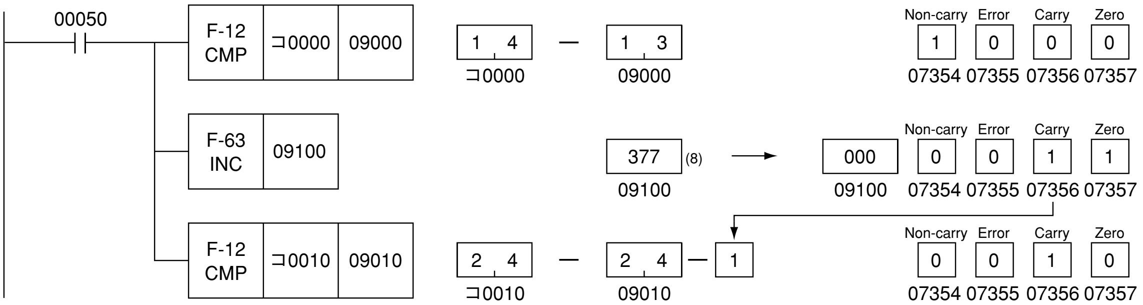
① Operation that includes the flag takes place for a different instruction, if double-length operational instruction has been programmed in a form of common operational condition.



② The double-length operation takes place even if there is no flag affecting instruction between the double-length operational instructions.



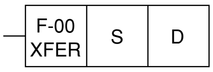
③ If there is a flag affecting instruction between the double-length operational instructions, operation takes place including the flag caused by the operation of that instruction.



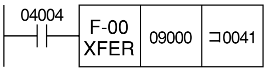
④ If other than BCD code is specified for the F-10, F-10w, Fc10, Fc10w, F-11, F-11w, Fc11, or Fc11w instruction, an error flag is evoked so that double-length operation will no more be executed thereafter.

[8] Description of each application instructions

F-00 XFER — Transfer 1 byte data

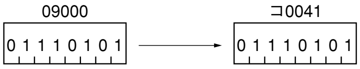
Symbol		
Function		The contents of the register (1 byte) are transferred to the register D.
Operation		S→D
Range of S		⊐0000 to ⊐1577 b0000 to b0777 09000 to 09777 19000 to 19777 29000 to 29777 39000 to 39777
Range of D		⊐0000 to ⊐1577 b0000 to b0777 09000 to 09777 19000 to 19777 29000 to 29777 39000 to 39777
Condition		Rising edge of input signal (OFF to ON)
After operation	Contents of S	Unchanged
	Contents of D	Contents of register S
	Flag	Unchanged

〔Example for use〕



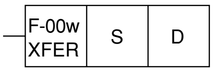
Instruction	
STR	04004
F-00	
	09000
	⊐0041

When the input condition 04004 changes from OFF to ON, the contents of the register 09000 are transferred to the register ⊐0041.

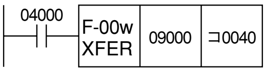


Similarity instructions: F-00w, F-70, F-70w, F-74, F-74w

F-00w XFER — Transfer 1 word data

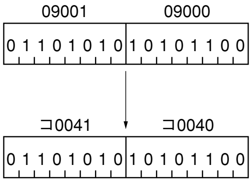
Symbol		
Function		The contents of the registers S, S+1 (1 word) are transferred to the register D, D+1.
Operation		S, S+1→D, D+1
Range of S		⊐0000 to ⊐1576 b0000 to b0776 09000 to 09776 19000 to 19776 29000 to 29776 39000 to 39776
Range of D		⊐0000 to ⊐1576 b0000 to b0776 09000 to 09776 19000 to 19776 29000 to 29776 39000 to 39776
Condition		Rising edge of input signal (OFF to ON)
After operation	Contents of S, S+1	Unchanged
	Contents of D	Contents of register S
	Contents of D+1	Contents of register S+1
	Flag	Unchanged

〔Example for use〕



Instruction	
STR	04000
F-00w	
	09000
	⊐0040

When the input condition 04000 changes from OFF to ON, the contents of the registers 09000 and 09001 (1 word) are transferred to the registers ⊐0040 and ⊐0041.



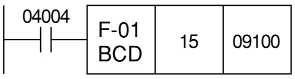
Note 1: Be sure to use even addresses for registers S and D.
Similarity instructions: F-00, F-70, F-70w, F-74, F-74w

F-01
BCD

Transfer BCD (2 digits) constant

Symbol	F-01 BCD · n · D	
Function	A 2 digits BCD constant "n" is transferred to the register D.	
Operation	n→D	
Range of "n"	00 to 99	
Range of D	コ0000 to コ1577 b0000 to b0777 09000 to 09777 19000 to 19777 29000 to 29777 39000 to 39777	
Condition	Rising edge of input signal (OFF to ON)	
After operation	Contents of D	n (00 to 99)
	Flag	Unchanged

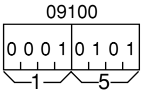
〔Example for use〕



Instruction	
STR	04004
F-01	
	15
	09100

When the input condition 04004 changes from OFF to ON, the BCD constant "15" is transferred to the register 09100.

The following value is contained in the register 09100 after the transfer.



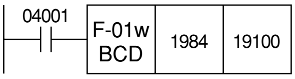
Similarity instructions: F-01w

F-01w
BCD

Transfer BCD (4 digits) constant

Symbol	F-01w BCD · n · D	
Function	A 4 digits BCD constant "n" is transferred to the registers D, D+1.	
Operation	n→D, D+1	
Range of "n"	0000 to 9999	
Range of D	コ0000 to コ1576 b0000 to b0776 09000 to 09776 19000 to 19776 29000 to 29776 39000 to 39776	
Condition	Rising edge of input signal (OFF to ON)	
After operation	Contents of D, D+1	n
	Flag	Unchanged

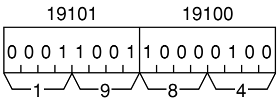
〔Example for use〕



Instruction	
STR	04001
F-01w	
	1984
	19100

When the input condition 04001 changes from OFF to ON, the BCD constant "1984" is transferred to the registers 19100 and 19101.

The following value is contained in the registers 19100 and 19101 after the transfer.

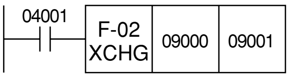


Note 1: Be sure to use even addresses for registers D.
Similarity instructions: F-01

F-02 XCHG Exchange 1 byte data between registers (eXCHanGe)

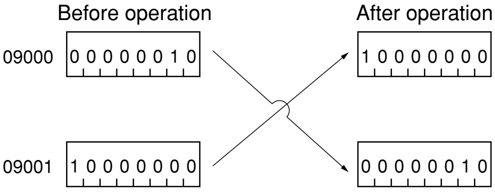
Symbol	F-02 XCHG D_1 D_2	
Function	The contents of the register D_1 are exchanged with the contents of the register D_2.	
Operation	$D_1 \longleftrightarrow D_2$	
Range of D_1	コ0000 to コ1577 b0000 to b0777 09000 to 09777 19000 to 19777 29000 to 29777 39000 to 39777	
Range of D_2	コ0000 to コ1577 b0000 to b0777 09000 to 09777 19000 to 19777 29000 to 29777 39000 to 39777	
Condition	Rising edge of input signal (OFF to ON)	
After operation	Contents of D_1	Contents of register D_2
	Contents of D_2	Contents of register D_1
	Flag	Unchanged

〔Example for use〕



Instruction	
STR	04001
F-02	
	09000
	09001

When the input condition 04001 changes from OFF to ON, the contents of the register 09000 are exchanged with the contents of the register 09001.

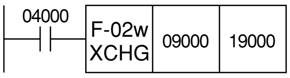


Similarity instrucions: F-02w

F-02w XCHG Exchange 1 word data between registers (eXCHanGe)

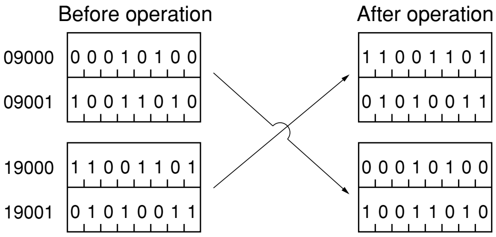
Symbol	F-02w XCHG D_1 D_2	
Function	The contents of the registers D_1, D_1+1 (1 word) are exchanged with the contents of the registers D_2, D_2+1 (1 word).	
Operation	D, D_1+1 $\longleftrightarrow$ D_2, D_2+1	
Range of D_1	コ0000 to コ1576 b0000 to b0776 09000 to 09776 19000 to 19776 29000 to 29776 39000 to 39776	
Range of D_2	コ0000 to コ1576 b0000 to b0776 09000 to 09776 19000 to 19776 29000 to 29776 39000 to 39776	
Condition	Rising edge of input signal (OFF to ON)	
After operation	Contents of D_1	Contents of the register D_2
	Contents of D_1+1	Contents of register D_2+1
	Contents of D_2	Contents of the register D_1
	Contents of D_2+1	Contents of register D_1+1
	Flag	Unchanged

〔Example for use〕



Instruction	
STR	04000
F-02w	
	09000
	19000

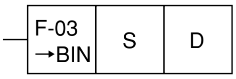
When the input condition 04000 changes from OFF to ON, the contents of the registers 09000 and 09001 (1 word) are exchanged with the contents of the registers 19000 and 19001 (1 word).



Note 1: Be sure to use even addresses for registers D_1 and D_2.
Similarity instructions: F-02

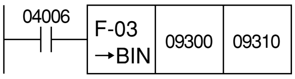
F-03
→BIN

Convert 2 digits BCD to 8 bits in binary

Symbol		
Function		The contents of the register S (8 bits) are assumed as BCD code, converted into binary, then the result is stored in the register D.
Operation		S→D
Range of S		コ0000 to コ1577 b0000 to b0777 09000 to 09777 19000 to 19777 29000 to 29777 39000 to 39777
Range of D		コ0000 to コ1577 b0000 to b0777 09000 to 09777 19000 to 19777 29000 to 29777 39000 to 39777
Condition		Rising edge of input signal (OFF to ON)
After operation	Contents of S	Unchanged
	Contents of D	・Result ・Unchanged when the contents of the register S are not BCD code.
	Flag	

Contents of register S	Zero 07357	Carry 07356	Error 07355	Non-carry 07354
BCD code	0	0	0	0
Not BCD code			1	

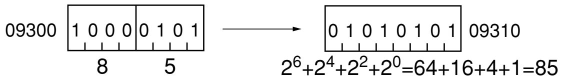
〔Example for use〕



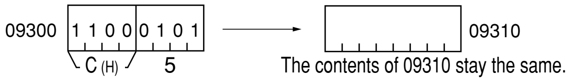
Instruction	
STR	04006
F-03	
	09300
	09310

When the input condition 04006 changes from OFF to ON, the contents of the register 09300 (8 bits) are assumed as BCD code, converted into binary, and its result is transferred to the register 09310. The contents of the register 09300 remain unchanged. If the contents of the register 09300 are not BCD code, the contents of the register 09310 are unchanged, and the error flag (07355) is set to "1".

・Transition of register contents and flags



Zero 07357	Carry 07356	Error 07355	Non-carry 07354
0	0	0	0



Zero 07357	Carry 07356	Error 07355	Non-carry 07354
0	0	1	0

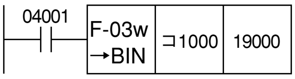
Similarity instructions: F-03w, F-53

F-03w →BIN

Convert 4 digits BCD to 16 bits in binary

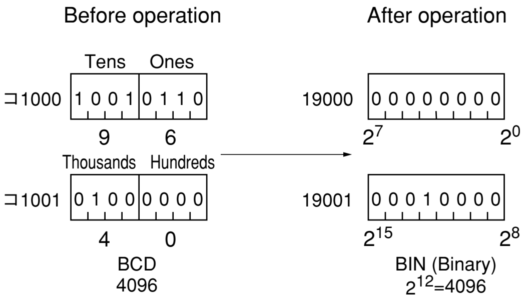
Symbol	F-03w →BIN S D					
Function	The 2 bytes BCD contents (4 digits data) of the registers S, S+1 are converted into binary, and the result is stored in the 2 bytes area of the registers D, D+1					
Operation	S_1, S+1→D, D+1					
Range of S	⊐0000 to ⊐1576 b0000 to b0776 09000 to 09776 19000 to 19776 29000 to 29776 39000 to 39776					
Range of D	⊐0000 to ⊐1576 b0000 to b0776 09000 to 09776 19000 to 19776 29000 to 29776 39000 to 39776					
Condition	Rising edge of input signal (OFF to ON)					
After operation	Contents of S,S+1	Unchanged				
	Contents of D	Result (0 to 255)	Unchanged when the contents of the registers S, S+1 are not BCD code.			
	Contents of D+1	Result (256 to 9999)				
	Flag	Contents of registers S,S+1	Zero 07357	Carry 07356	Error 07355	Non-carry 07354
		BCD code	0	0	0	0
		Not BCD code			1	

〔Example for use〕



Instruction	
STR	04001
F-03w	
	⊐1000
	19000

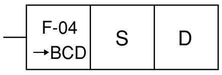
When the input condition 04001 changes from OFF to ON, the contents of 4 digits BCD data in registers ⊐1000 and ⊐1001 are converted into binary, and its result is transferred and stored in 2 bytes area of registers 19000 and 19001.



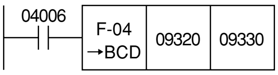
Note 1: If the F-53 insutruction is used for programming, the F-03w instruction displays the program during monitoring.
Similarity instructions: F-03, F-53

F-04
→BCD

Convert 8 bits binary to 2 digits BCD

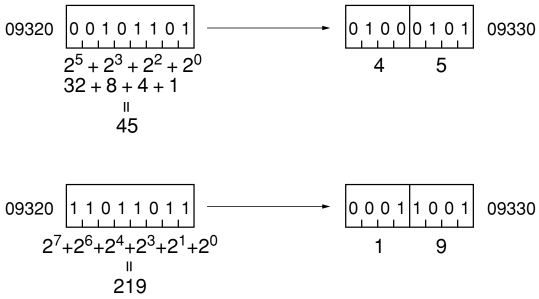
Symbol		
Function	The contents of the register S (8 bits) are assumed as a binary code, converted into BCD code, then the result is stored in the register D.	
Operation	S→D	
Range of S	⊐0000 to ⊐1577 b0000 to b0777 09000 to 09777 19000 to 19777 29000 to 29777 39000 to 39777	
Range of D	⊐0000 to ⊐1577 b0000 to b0777 09000 to 09777 19000 to 19777 29000 to 29777 39000 to 39777	
Condition	Rising edge of input signal (OFF to ON)	
After operation	Contents of S	Unchanged
	Contents of D	Result
	Flag	Unchanged

〔Example for use〕



Instruction	
STR	04006
F-04	
	09320
	09330

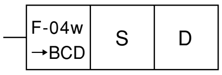
When the input condition 04006 changes from OFF to ON, the contents of the register 09320 (8 bits) are assumed as a binary code, converted into BCD code, and its result is transferred to the register 09330.
The contents of the register 09320 remain unchanged.
If the BCD number converted should exceed"100", the digit of hundreds will be disregarded.



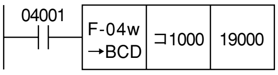
Similarity instructions: F-04w, F-54

F-04w
→BCD

Convert 16 bits binary to 6 digits BCD

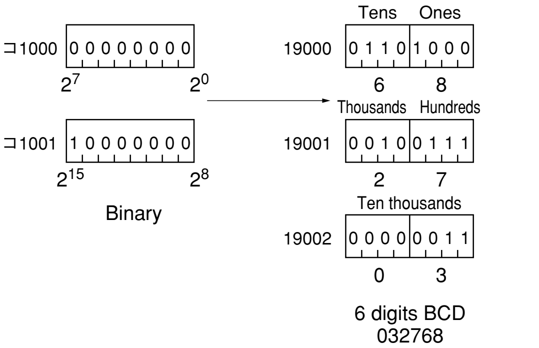
Symbol		
Function	The contents of 2 bytes binary in the registers S, S+1 converted into 6 digits BCD,and the result is stored in 3 bytes of the registers D, D+1, and D+2.	
Operation	S, S+1→D, D+1, D+2	
Range of S	⊐0000 to ⊐1576 b0000 to b0776 09000 to 09776 19000 to 19776 29000 to 29776 39000 to 39776	
Range of D	⊐0000 to ⊐1575 b0000 to b0775 09000 to 09775 19000 to 19775 29000 to 29775 39000 to 39775	
Condition	Rising edge of input signal (OFF to ON)	
After operation	Contents of S, S+1	Unchanged
	Contents of D	Result (ones and tens)
	Contents of D+1	Result (hundreds and thousands)
	Contents of D+2	Result (ten thousands)
	Flag	Unchanged

〔Example for use〕



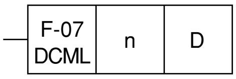
Instruction	
STR	04001
F-04w	
	⊐1000
	19000

When the input condition 04001 changes from OFF to ON, the contents of 2 bytes binary in registers ⊐1000 and ⊐1001 are converted into 6 digits BCD and stored in 3 bytes area of registers that begin from 19000.

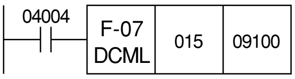


Note 1: If the F-54 instruction is used for programming, the F-04w instruction displays the program during monitoring.
Similarity instructions: F-04, F-54

F-07 DCML Transfer decimal (1 byte) constant (DeCiMaL)

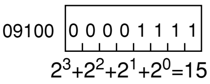
Symbol	
Function	A decimal constant "n" is transferred to the register D.
Operation	n → D
Range of "n"	000 to 255
Range of D	ㄱ0000 to ㄱ1577 b0000 to b0777 09000 to 09777 19000 to 19777 29000 to 29777 39000 to 39777
Condition	Rising edge of input signal (OFF to ON)
After operation: Contents of D	n (000 to 255)
After operation: Flag	Unchanged

〔Example for use〕



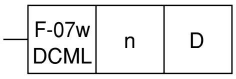
Instruction	
STR	04004
F-07	
	015
	09100

When the input condition 04004 changes from OFF to ON, the decimal constant "15" is transferred to the register 09100.
The register 09100 is in the following binary code representation.

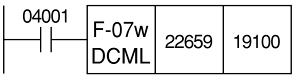


Similarity instructions: F-07w

F-07w DCML Transfer decimal (1 word) constant (DeCiMaL)

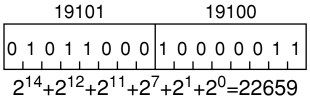
Symbol	
Function	A decimal constant "n" is transferred to the registers D, D+1.
Operation	n→D, D+1
Range of "n"	00000 to 65535
Range of D	ㄱ0000 to ㄱ1576 b0000 to b0776 09000 to 09776 19000 to 19776 29000 to 29776 39000 to 39776
Condition	Rising edge of input signal (OFF to ON)
After operation: Contents of D,D+1	n (0000 to 65535)
After operation: Flag	Unchanged

〔Example for use〕



Instruction	
STR	04001
F-07w	
	22659
	19100

When the input condition 04001 changes from OFF to ON, the decimal constant 22659 is transferred to the registers 19100 and 19101.
The following binary code is stored in the registers 19100 and 19101.



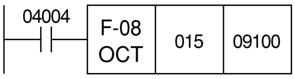
Note 1: Be sure to use even addresses for register D.
Similarity instructions: F-07w

F-08 OCT

Transfer octal (1 byte) constant (OCTal)

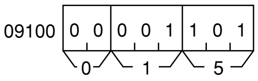
Symbol		F-08 OCT | n | D
Function		An octal constant "n" is transferred to the register D.
Operation		n → D
Range of "n"		000 to 377 (8)
Range of D		コ0000 to コ1577 b0000 to b0777 09000 to 09777 19000 to 19777 29000 to 29777 39000 to 39777
Condition		Rising edge of input signal (OFF to ON)
After operation	Contents of D	n (000 to 377)
	Flag	Unchanged

〔Example for use〕



Instruction	
STR	04004
F-08	
	015
	09100

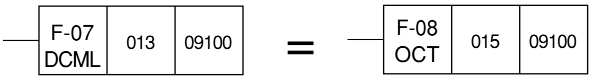
When the input condition 04004 changes from OFF to ON, the octal constant 015 is transferred to the register 09100.
The register 09100 is in the following code representation.

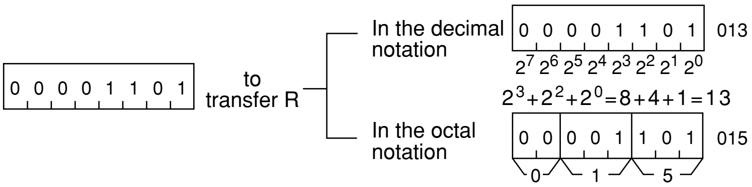


Similarity instructions: F-08w, F-71, F-71w

Reference

Though F-07 (transfer decimal constant) handles a decimal number and F-08 (transfer octal constant) an octal number, the contents of the register after the transfer are represented in the binary code for both instructions.



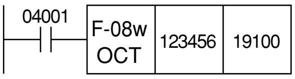


F-08w OCT

Transfer decimal (1 word) constant (OCTal)

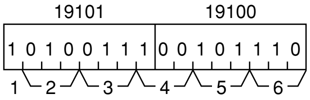
Symbol		F-08w OCT | n | D
Function		An octal constant "n" is transferred to the registers D, D+1.
Operation		n→D, D+1
Range of "n"		000000 to 177777 (8)
Range of D		コ0000 to コ1576 b0000 to b0776 09000 to 09776 19000 to 19776 29000 to 29776 39000 to 39776
Condition		Rising edge of input signal (OFF to ON)
After operation	Contents of D, D+1	n (000000 to 177777)
	Flag	Unchanged

〔Example for use〕



Instruction	
STR	04001
F-08w	
	123456
	19100

When the input condition 04001 changes from OFF to ON, the decimal constant 123456 is transferred to the registers 19100 and 19101.
The following code is stored in the registers 19100 and 19101.

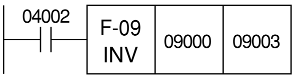


Note 1: Be sure to use even addresses for register D.
Similarity instructions: F-08, F-71, F-71w

F-09 INV Complement 8 bits data (INVerter)

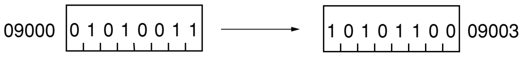
Symbol	F-09 INV | S | D	
Function	The contents of the register S are complemented and stored in the register D.	
Operation	$\overline{S} \rightarrow D$	
Range of S	⊐0000 to ⊐1577 b0000 to b0777 09000 to 09777 19000 to 19777 29000 to 29777 39000 to 39777	
Range of D	⊐0000 to ⊐1577 b0000 to b0777 09000 to 09777 19000 to 19777 29000 to 29777 39000 to 39777	
Condition	Rising edge of input signal (OFF to ON)	
After operation	Contents of S	Unchanged
	Contents of D	Complement of register S contents
	Flag	Unchanged

〔Example for use〕



Instruction	
STR	04002
F-09	
	09000
	09003

When the input condition 04002 changes from OFF to ON, the 8 bits contents of the register 09000 are complemented and its result is stored in the register 09003.
The contents of the register 09000 remain unchanged.

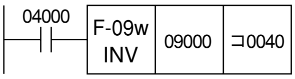


09000 | 0 1 0 1 0 0 1 1 | → | 1 0 1 0 1 1 0 0 | 09003

F-09w INV Complement 16 bits data (INVerter)

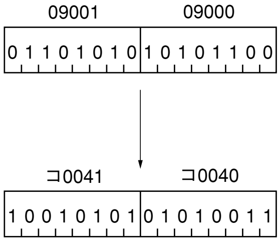
Symbol	F-09w INV | S | D	
Function	The contents of the registers S, S+1 (16 bits) are complemented and stored in the registers D, D+1.	
Operation	$\overline{S, S+1} \rightarrow D, D+1$	
Range of S	⊐0000 to ⊐1576 b0000 to b0776 09000 to 09776 19000 to 19776 29000 to 29776 39000 to 39776	
Range of D	⊐0000 to ⊐1576 b0000 to b0776 09000 to 09776 19000 to 19776 29000 to 29776 39000 to 39776	
Condition	Rising edge of input signal (OFF to ON)	
After operation	Contents of S, S+1	Unchanged
	Contents of D	Complement of register S contents
	Contents of D+1	Complement of register S+1
	Flag	Unchanged

〔Example for use〕



Instruction	
STR	04000
F-09w	
	09000
	⊐0040

When the input condition 04000 changes from OFF to ON, the contents of the registers 09000 and 09001 (16 bits) are complemented and its result is stored in the registers ⊐0040 and ⊐0041.
The contents of the registers 09000 and 09001 remain unchanged.



09001	09000
0 1 1 0 1 0 1 0	1 0 1 0 1 1 0 0

↓

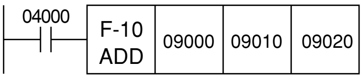
⊐0041	⊐0040
1 0 0 1 0 1 0 1	0 1 0 1 0 0 1 1

Note 1: Be sure to use even addresses for registers S and D.

F-10 ADD Add register and register (BCD 2 digits) (ADD)

<table>
<tr><td colspan="2">Symbol</td><td>F-10 ADD | S1 | S2 | D</td></tr>
<tr><td colspan="2">Function</td><td>The contents of the register S1 are added with the contents of the register S2 in BCD 2 digits and its result is stored in the register D.</td></tr>
<tr><td colspan="2">Operation</td><td>S1+S2→D</td></tr>
<tr><td colspan="2">Range of S1</td><td>コ0000 to コ1577
b0000 to b0777
09000 to 09777
19000 to 19777
29000 to 29777
39000 to 39777</td></tr>
<tr><td colspan="2">Range of S2</td><td>コ0000 to コ1577
b0000 to b0777
09000 to 09777
19000 to 19777
29000 to 29777
39000 to 39777</td></tr>
<tr><td colspan="2">Range of D</td><td>コ0000 to コ1577
b0000 to b0777
09000 to 09777
19000 to 19777
29000 to 29777
39000 to 39777</td></tr>
<tr><td colspan="2">Condition</td><td>Rising edge of input signal (OFF to ON)</td></tr>
<tr><td rowspan="4">After operation</td><td>Contents of S1</td><td>Unchanged</td></tr>
<tr><td>Contents of S2</td><td>Unchanged</td></tr>
<tr><td>Contents of D</td><td>・Lower 2 digits of the result.
・Unchanged when the contents of registers S1 and S2 are not BCD code.</td></tr>
<tr><td>Flag</td><td>
<table>
<tr><th>Result</th><th>Zero 07357</th><th>Carry 07356</th><th>Error 07355</th><th>Non-carry 07354</th></tr>
<tr><td>0</td><td>1</td><td>0</td><td>0</td><td>1</td></tr>
<tr><td>1 to 99</td><td>0</td><td>0</td><td>0</td><td>1</td></tr>
<tr><td>100</td><td>1</td><td>1</td><td>0</td><td>0</td></tr>
<tr><td>101 and above</td><td>0</td><td>1</td><td>0</td><td>0</td></tr>
<tr><td>S1 and S2 are not BCD code.</td><td>0</td><td>0</td><td>1</td><td>0</td></tr>
</table>
</td></tr>
</table>

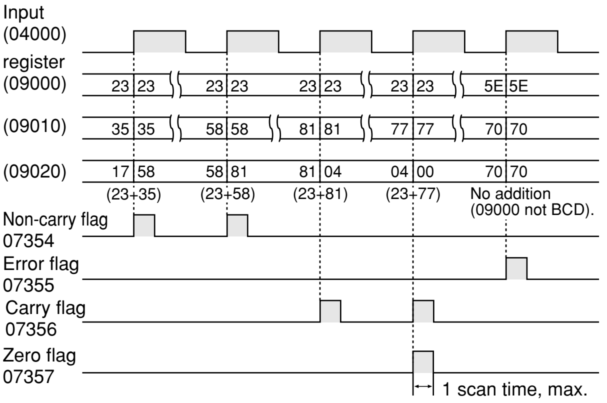
〔Example for use〕



Instruction	
STR	04000
F-10	09000
	09010
	09020

When the input condition 04000 changes from OFF to ON, the contents of the register 09000 are added with the contents of the register 09010 and its result is stored in the register 09020. The contents of registers 09000 and 09010 remain unchanged.

・Transition of result and flags

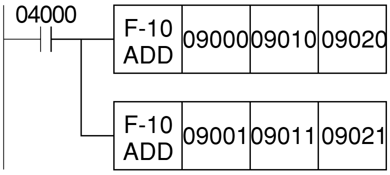


Valid until a flag affecting instruction is met in the program.

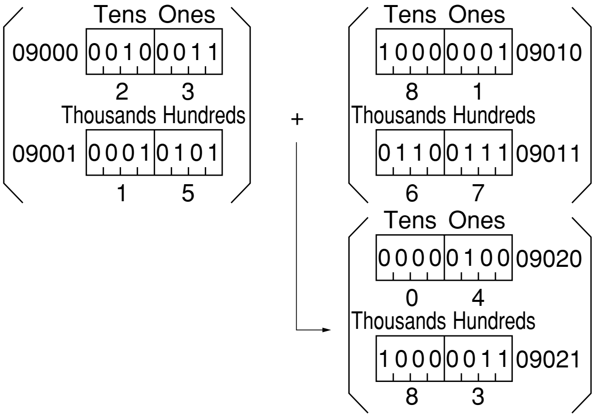
Similarity instructions: F-10w, Fc10, Fc10w

Reference

In case 3 BCD digits or more have to be added, the F-10 instruction must be provided successively.
When the F-10 instruction is programmed repeatedly, the contents of the carry flag (07356) are also added after the second instruction. For the F-10 instruction that appears first in succession to the STR instruction, the contents of the carry flag (07356) are not added.



Instruction	
STR	04000
F-10	09000
	09010
	09020
F-10	09001
	09011
	09021



・The above example shows the case of 1523+6781=8304.
・If programmed from lower digit, the carry information will be carried on to upper digit.

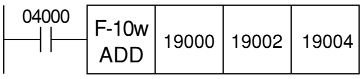
F-10w ADD

Add register and register (BCD 4 digits) (ADD)

Symbol	F-10w ADD | S_1 | S_2 | D
Function	The contents of the registers S_1, S_1+1 are added with the contents of the registers S_2, S_2+1 in BCD 4 digits and its result is stored in the registers D, D+1.
Operation	(S_1, S_1+1) + (S_2, S_2+1) →D, D+1
Range of S_1	コ0000 to コ1576 b0000 to b0776 09000 to 09776 19000 to 19776 29000 to 29776 39000 to 39776
Range of S_2	コ0000 to コ1576 b0000 to b0776 09000 to 09776 19000 to 19776 29000 to 29776 39000 to 39776
Range of D	コ0000 to コ1576 b0000 to b0776 09000 to 09776 19000 to 19776 29000 to 29776 39000 to 39776
Condition	Rising edge of input signal (OFF to ON)

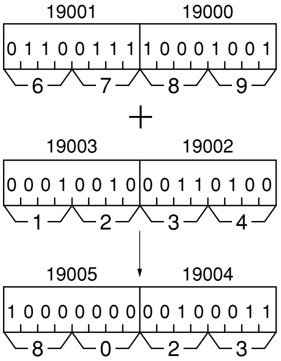
<table>
<tr><td rowspan="10">After operation</td><td>Contents of S1,S1+1</td><td colspan="5">Unchanged</td></tr>
<tr><td>Contents of S2,S1+2</td><td colspan="5">Unchanged</td></tr>
<tr><td>Contents of D</td><td>Lower 2 digits of the result</td><td colspan="4" rowspan="2">Unchanged when the contents of registers S1, S1+1, S2 and S2+1 are not BCD code.</td></tr>
<tr><td>Contents of D+1</td><td>Upper 2 digits of the result</td></tr>
<tr><td rowspan="6">Flag</td><td>Result</td><td>Zero 07357</td><td>Carry 07356</td><td>Error 07355</td><td>Non-carry 07354</td></tr>
<tr><td>0</td><td>1</td><td>0</td><td>0</td><td>1</td></tr>
<tr><td>1 to 9999</td><td>0</td><td>0</td><td>0</td><td>1</td></tr>
<tr><td>10000</td><td>1</td><td>1</td><td>0</td><td>0</td></tr>
<tr><td>10001 and above</td><td>0</td><td>1</td><td>0</td><td>0</td></tr>
<tr><td>Not BCD code</td><td>0</td><td>0</td><td>1</td><td>0</td></tr>
</table>

〔Example for use〕



Instruction	
STR	04000
F-10w	
	19000
	19002
	19004

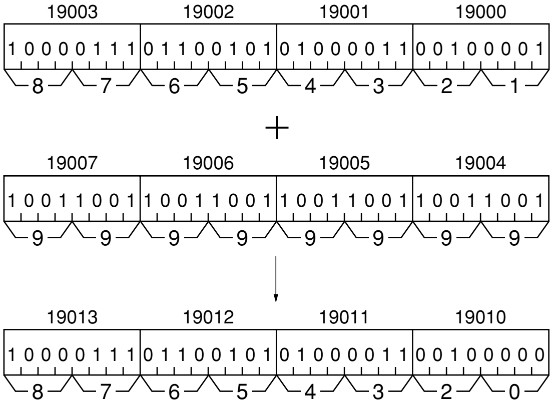
When the input condition 04000 changes from OFF to ON, the 4 digits BCD contents of the registers 19000 and 19001 are added with the contents of the registers 19002 and 19003 and its result is stored in the registers 19004 and 19005.

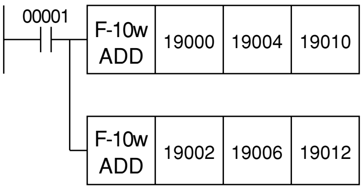


Note 1: Be sure to use even addresses for registers S and D.
Similarity instructions: F-10, Fc10, Fc10w

Reference

As double length operation is possible for the F-10w instruction, same as the F-10 instruction, write F-10w instruction in succession to add 5 digits or more BCD.





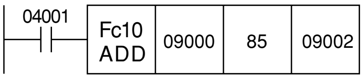
Instruction	
STR	00001
F-10w	
	19000
	19004
	19010
F-10w	
	19002
	19006
	19012

Fc10 ADD Add register (BCD 2 digits) and constant (BCD 2 digits) (ADD)

Symbol	Fc10 ADD | S_1 | n | D
Function	The contents of the register S_1 are added with the 2 digits BCD constant "n" and its result is stored in the register D.
Operation	S_1+n→D
Range of S_1	ヨ0000 to ヨ1577 b0000 to b0777 09000 to 09777 19000 to 19777 29000 to 29777 39000 to 39777
Range of "n"	00 to 99
Range of D	ヨ0000 to ヨ1577 b0000 to b0777 09000 to 09777 19000 to 19777 29000 to 29777 39000 to 39777
Condition	Rising edge of input signal (OFF to ON)

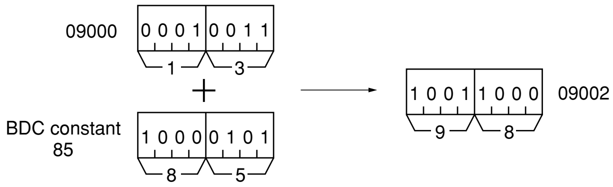
After operation	Contents of S1	Unchanged				
	Contents of D	・Lower 2 digits of the result ・Unchanged when the contents of the register S_1 are not BCD code.				
	Flag	Result	Zero 07357	Carry 07356	Error 07355	Non-carry 07354
		0	1	0	0	1
		1 to 99	0	0	0	1
		100	1	1	0	0
		101 and above	0	1	0	0
		S_1 is not BCD code	0	0	1	0

〔Example for use〕



Instruction	
STR	04001
Fc10	
	09000
	85
	09002

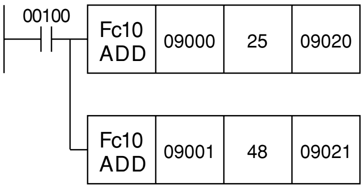
When the input condition 04001 changes from OFF to ON, the contents of the register 09000 are added with the BCD constant 85 and its result is stored in the register 09002.
It operates in the same timing as the F-10 instruction.



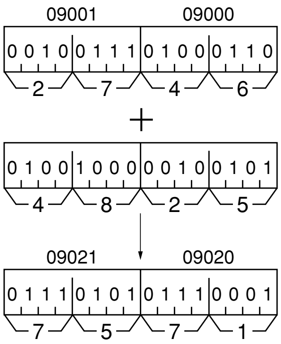
Similarity instructions: F-10, F-10w, Fc10w

Reference

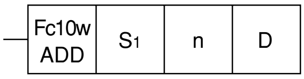
Similar to the F-10 instruction, it is possible to add 3 digits or more of BCD value.



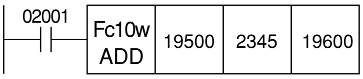
Instruction	
STR	00100
Fc10	
	09000
	25
Fc10	09020
	09001
	48
	09021



Fc10w ADD — Add register (BCD 4 digits) and constant (BCD 4 digits)

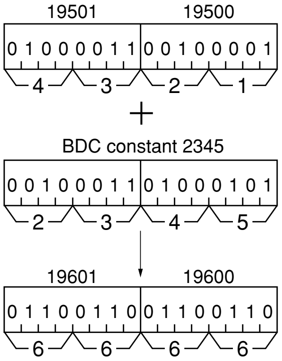
Symbol						
Function	The BCD 4 digits contents of the registers S_1, S_1+1 are added with the 4 digits BCD constant "n" and its result is stored in the registers D, D+1.					
Operation	(S_1, S_1+1) +n→D, D+1					
Range of S_1	コ0000 to コ1576 b0000 to b0776 09000 to 09776 19000 to 19776 29000 to 29776 39000 to 39776					
Range of "n"	0000 to 9999					
Range of D	コ0000 to コ1576 b0000 to b0776 09000 to 09776 19000 to 19776 29000 to 29776 39000 to 39776					
Condition	Rising edge of input signal (OFF to ON)					
After operation	Contents of S1,S1+1	Unchanged				
	Contents of D	Lower 2 digits of the result	Unchanged when the contents of registers S_1 and S_1+1 are not BCD code.			
	Contents of D+1	Upper 2 digits of the result				
	Flag	Result	Zero 07357	Carry 07356	Error 07355	Non-carry 07354
		0	1	0	0	1
		1 to 9999	0	0	0	1
		10000	1	1	0	0
		10000 and above	0	1	0	0
		Not BCD code	0	0	1	0

〔Example for use〕



Instruction	
STR	02001
Fc10w	
	19500
	2345
	19600

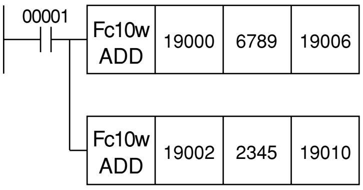
When the input condition 02001 changes from OFF to ON, the 4 digits BCD contents of the registers 19500 and 19501 are added with the BCD constant 2345 and its result is stored in the registers 19600 and 19601.



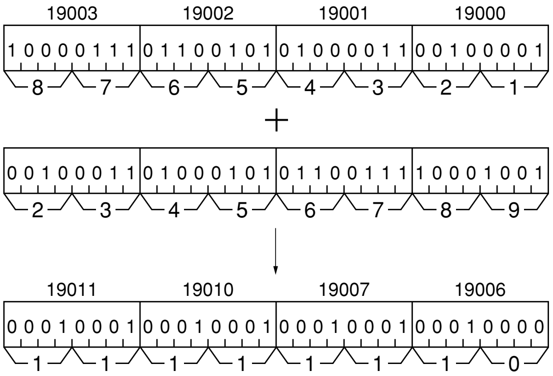
Note 1: Be sure to use even addresses for registers S_1 and D.
Similarity instructions: F-10, F-10w, Fc10

Reference

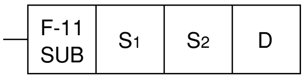
As double length operation is possible for the Fc10w instruction, same as the Fc10 instruction, write Fc10w instruction in succession to add 5 digits or more of BCD.



Instruction	
STR	00001
Fc10w	
	19000
	6789
	19006
Fc10w	
	19002
	2345
	19010

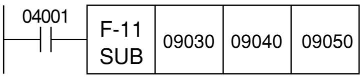


F-11 SUB Subtract register from register (BCD 2 digits) (SUBtract)

Symbol	
Function	The contents of the register S_1 are subtracted by the contents of the register S_2 in BCD 2 digits and its result is stored in the register D.
Operation	$S_1 - S_2 \rightarrow D$
Range of S_1	∃0000 to ∃1577 b0000 to b0777 09000 to 09777 19000 to 19777 29000 to 29777 39000 to 39777
Range of S_2	∃0000 to ∃1577 b0000 to b0777 09000 to 09777 19000 to 19777 29000 to 29777 39000 to 39777
Range of D	∃0000 to ∃1577 b0000 to b0777 09000 to 09777 19000 to 19777 29000 to 29777 39000 to 39777
Condition	Rising edge of input signal (OFF to ON)

After operation	Contents of S_1	Unchanged				
	Contents of S_2	Unchanged				
	Contents of D	・Result ・Unchanged when the contents of registers S_1 and S_2 are not BCD code.				
	Flag	Result	Zero 07357	Carry 07356	Error 07355	Non-carry 07354
		0	1	0	0	1
		1 to 99	0	0	0	1
		Negative value	0	1	0	0
		S_1 and S_2 are not BCD code	0	0	1	0

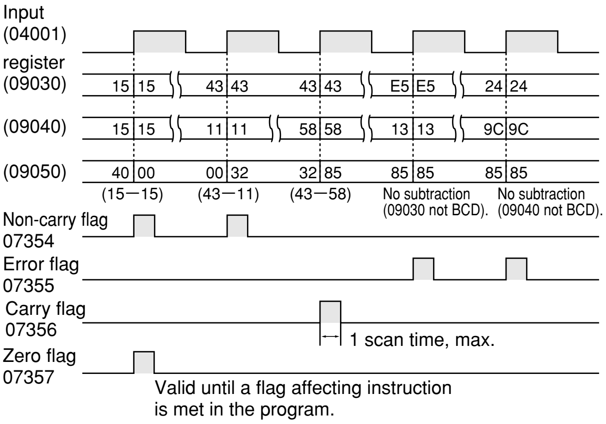
〔Example for use〕



Instruction	
STR	04001
F-11	
	09030
	09040
	09050

When the input condition 04001 changes from OFF to ON, the contents of the register 09030 are subtracted by the contents of the register 09040 and its result is stored in the register 09050. The contents of registers 09030 and 09040 remain unchanged.

・Transition of operational result and flags



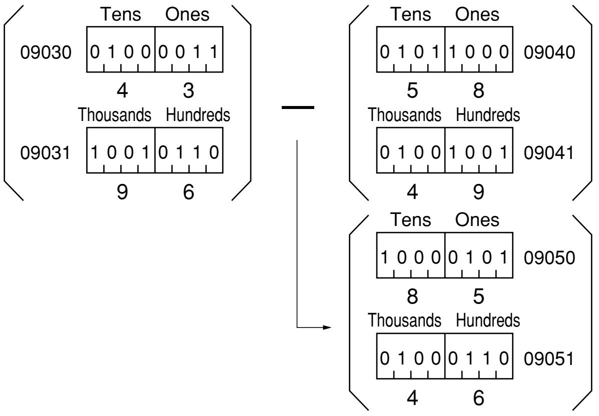
If (contents of S_1) < (contents of S_2) is calculated, the answer will be produced in the complement of 100.

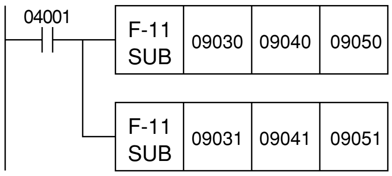
〔Example〕
23−85 = −62 will produce the answer of 38 which is the complement of 100 of 62.
(Assume it to be 123−85 = 38.)

Similarity instructions: F-11w, Fc11, Fc11w

Reference

In case 3 BCD digits or more have to be subtracted, the F-11 instruction must be provided successively.
When the F-11 instruction is programmed repeatedly, the contents of the carry flag (07356) is also subtracted after the second instruction. For the F-11 instruction that appears first in succession to the STR instruction, the contents of the carry flag (07356) is not subtracted.

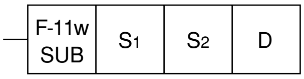




Instruction	
STR	04001
F-11	
	09030
	09040
	09050
F-11	
	09031
	09041
	09051

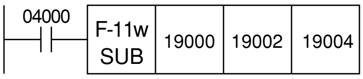
・The above example shows the case of 9643−4958=4685.
・If programmed from low order digit, the carry information will be carried on to a higher digit.

F-11w SUB Subtract register from register (BCD 4 digits) (SUBtract)

Symbol	
Function	The 4 digits BCD contents of the registers S_1 and S_1+1 are subtracted by the contents of the registers S_2, S_2+1 and its result is stored in the registers D, D+1.
Operation	(S_1, S_1+1) − (S_2, S_2+1) → D, D+1
Range of S_1	コ0000 to コ1576 b0000 to b0776 09000 to 09776 19000 to 19776 29000 to 29776 39000 to 39776
Range of S_2	コ0000 to コ1576 b0000 to b0776 09000 to 09776 19000 to 19776 29000 to 29776 39000 to 39776
Range of D	コ0000 to コ1576 b0000 to b0776 09000 to 09776 19000 to 19776 29000 to 29776 39000 to 39776
Condition	Rising edge of input signal (OFF to ON)

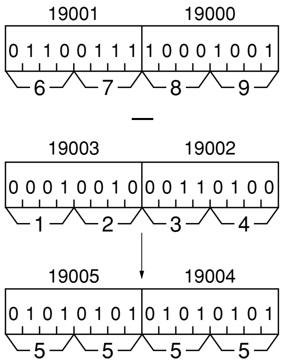
After operation	Contents of S_1, S_1+1	Unchanged				
	Contents of S_2, S_2+1	Unchanged				
	Contents of D	Lower 2 digits of the result	Unchanged when the contents of registers S_1, S_1+1, S_2 and S_2+1 are not BCD code.			
	Contents of D+1	Upper 2 digits of the result				
	Flag	Result	Zero 07357	Carry 07356	Error 07355	Non-carry 07354
		0	1	0	0	1
		1 to 9999	0	0	0	1
		Negative value	0	1	0	0
		Not BCD code	0	0	1	0

〔Example for use〕



Instruction	
STR	04000
F-11w	19000 19002 19004

When the input condition 04000 changes from OFF to ON, the 4 digits BCD contents of the registers 19000 and 19001 are subtracted by the contents of the registers 19002 and 19003 and its result is stored in the registers 19004 and 19005.



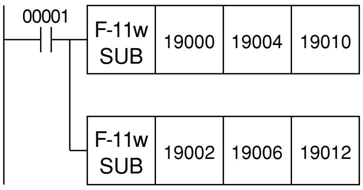
Result will be produced as a complement of 10000, if (contents of S_1, S_1+1) < (contents of S_2, S_2+1) is carried out.

〔Example〕
2578−7890 = −5312
will produce the answer of 4688
which is the complement of 10000 of 5312.
(Assume it to be 12578−7890=4688.)

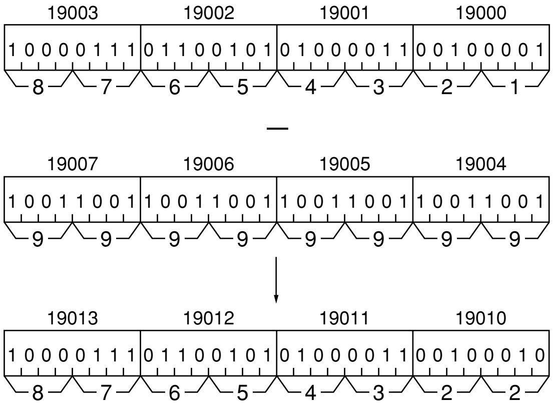
Note 1: Be sure to use even addresses for registers S_1, S_2 and D.
Similarity instructions: F-11, Fc11, Fc11w

Reference

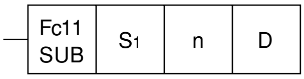
As double length operation is possible for the F-11w instruction, same as the F-11 instruction, write F-11w instruction in succession to subtract 5 digits or more of BCD.



Instruction	
STR	00001
F-11w	19000 19004 19010
F-11w	19002 19006 19012

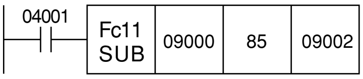


Fc11 SUB — Subtract constant (2 digits) from register (BCD 2 digits) (SUBtract)

Symbol	
Function	The contents of the register S_1 are subtracted by 2 digits BCD constant "n" and its result is stored in the register D.
Operation	S_1→n→D
Range of S_1	コ0000 to コ1577 b0000 to b0777 09000 to 09777 19000 to 19777 29000 to 29777 39000 to 39777
Range of "n"	00 to 99
Range of D	コ0000 to コ1577 b0000 to b0777 09000 to 09777 19000 to 19777 29000 to 29777 39000 to 39777
Condition	Rising edge of input signal (OFF to ON)

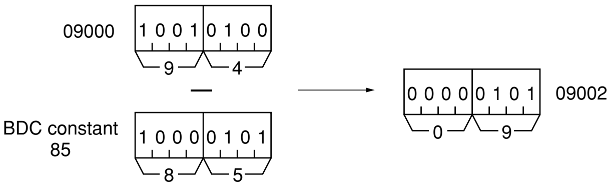
After operation	Contents of S1	Unchanged				
	Contents of D	・Result ・Unchanged when the contents of the register S_1 are not BCD code.				
	Flag	Result	Zero 07357	Carry 07356	Error 07355	Non-carry 07354
		0	1	0	0	1
		1 to 99	0	0	0	1
		Negative value	0	1	0	0
		S_1 is not BCD code	0	0	1	0

〔Example for use〕



Instruction	
STR	04001
Fc11	
	09000
	85
	09002

When the input condition 04001 changes from OFF to ON, the contents of the register 09000 are subtracted by the BCD constant 85 and its result is stored in the register 09002.
It operates in the same timing as the F-11 instruction.

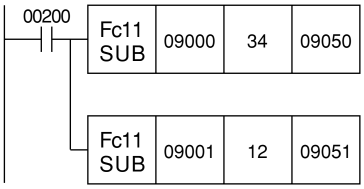


If (contents of S_1) < n is calculated, the answer will be produced in the complement of 100.
〔Example〕
23−85 = −62 will produce the answer of 38 which is the complement of 100 of 62.
(Assume it to be 123−85=38.)

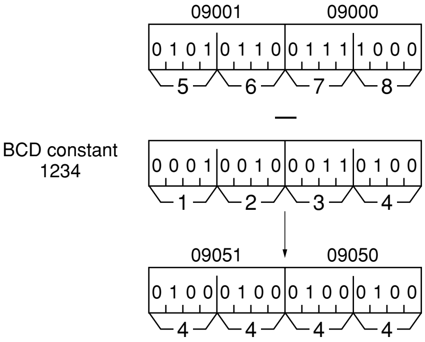
Similarity instructions: F-11, F-11w, Fc11w

Reference

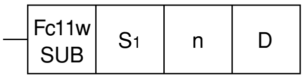
Similar to the F-11 instruction, it is possible to subtract 3 digits or more of BCD value.



Instruction	
STR	00200
Fc11	
	09000
	34
	09050
Fc11	
	09001
	12
	09051

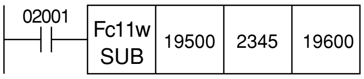


Fc11w SUB Subtract constant (4 digits) from register (BCD 4 digits) (SUBtract)

Symbol	
Function	The 4 digits BCD contents of the registers S1, S1+1 are subtracted by 4 digits BCD constant "n" and its result is stored in the registers D, D+1.
Operation	(S1, S1+1) −n→D, D+1
Range of S1	コ0000 to コ1576 b0000 to b0776 09000 to 09776 19000 to 19776 29000 to 29776 39000 to 39776
Range of "n"	0000 to 9999
Range of D	コ0000 to コ1576 b0000 to b0776 09000 to 09776 19000 to 19776 29000 to 29776 39000 to 39776
Condition	Rising edge of input signal (OFF to ON)

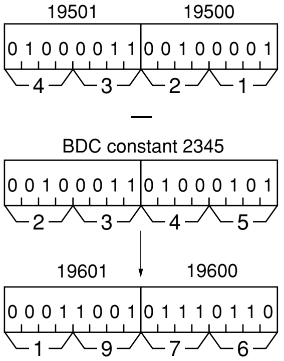
After operation	Contents of S1,S1+1	Unchanged				
	Contents of D	Lower 2 digits of the result	Unchanged when the contents of registers S1 and S1+1 are not BCD code.			
	Contents of D+1	Upper 2 digits of the result				
	Flag	Result	Zero 07357	Carry 07356	Error 07355	Non-carry 07354
		0	1	0	0	1
		1 to 9999	0	0	0	1
		Negative value	0	1	0	0
		Not BCD code	0	0	1	0

〔Example for use〕



Instruction	
STR	02001
Fc11w	
	19500
	2345
	19600

When the input condition 02001 changes from OFF to ON, the 4 digits BCD contents of the registers 19500 and 19501 are subtracted by the BCD constant 2345 and its result is stored in the registers 19600 and 19601.

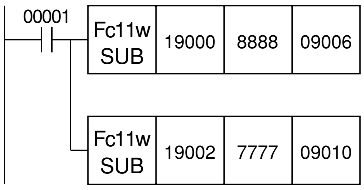


If (contents of S1, S1+1) < n is calculated, the answer will be produced in the complement of 10000.
〔Example〕
4568−7890 = −3322 will produce the answer of 6678 which the complement of 10000 of 3322.
(Assume it to be 14568−7890=6678.)

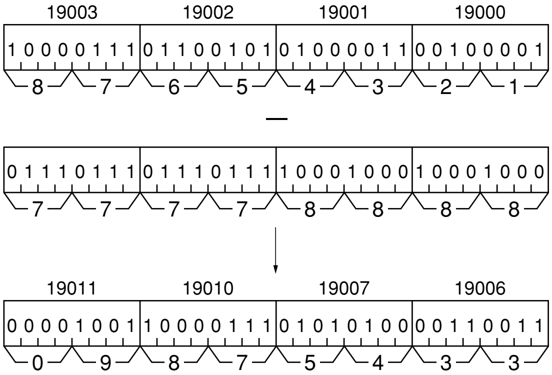
Note 1: Be sure to use even addresses for registers S1 + D.
Similarity instructions: F-11, F-11w, Fc11

Reference

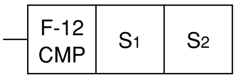
As double length operation is possible for the Fc11w instruction, same as the Fc11 instruction, write Fc11w instruction in succession to subtract 9 digits or more of BCD.



Instruction	
STR	00001
Fc11w	
	19000
	8888
	19006
Fc11w	
	19002
	7777
	19010

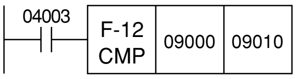


F-12 CMP Compare register with register (1 byte) (CoMPare)

Symbol	
Function	The contents of the register S_1 are compared with the contents of the register S_2.
Operation	$S_1 < = > S_2$→Flag
Range of S_1	ℷ0000 to ℷ1577 b0000 to b0777 09000 to 09777 19000 to 19777 29000 to 29777 39000 to 39777
Range of S_2	ℷ0000 to ℷ1577 b0000 to b0777 09000 to 09777 19000 to 19777 29000 to 29777 39000 to 39777
Condition	When the input signal is ON (not limited to OFF to ON change)

After operation	Contents of S1	Unchanged				
	Contents of S2	Unchanged				
	Flag	Contents of register	Zero 07357	Carry 07356	Error 07355	Non-carry 07354
		$S_1 > S_2$	0	0	0	1
		$S_1 = S_2$	1	0	0	1
		$S_1 < S_2$	0	1	0	0

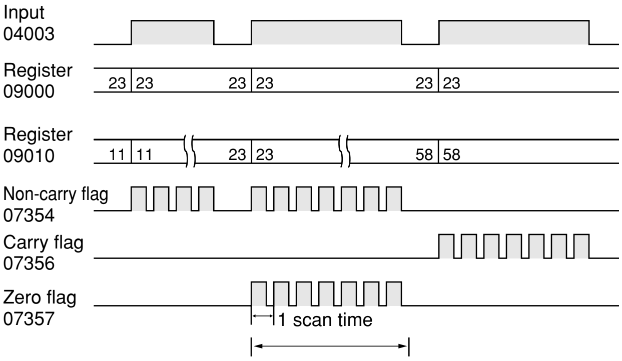
〔Example for use〕



Instruction	
STR	04003
F-12	
	09000
	09010

When the input condition 04003 is ON, the contents of the register 09000 are compared with the contents of the register 09010 and its results are set in the non-carry flag (07354), carry flag (07356), and the zero flag (07357). registers 09000 and 09010 remain unchanged after this operation.

・Transition of register contents and flags

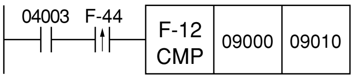


So long as the input condition is ON, comparison is done at every scan and the zero flag is tumed ON.

Similarity instructions: F-12w Fc12, Fc12w

Reference

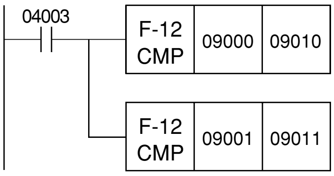
In case comparison is to be done only at an OFF to ON transition of the input condition, use the differentiate instruction in conjunction with the input condition.



Instruction	
STR	04003
F-44	
F-12	09000
	09010

Reference

To compare data of 2 bytes or more, it must be so programmed that comparision should start from a lower order digit, as in addition (F-10) and subtraction (F-11). If the F-12 were programmed in succession, the contents of the carry flag (07356) are also compared after the second F-12 instruction. For the F-12 instruction that first follows the STR instruction, the contents of the carry flag (07356) are exemption from comparison.



Instruction	
STR	04003
F-12	
	09000
	09010
F-12	
	09001
	09011

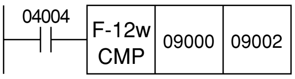
When programmed from a lower digit, the carry-down information is forwarded to an upper digit.

F-12w CMP Compare register with register (1 word) (CoMPare)

Symbol	F-12w CMP | S_1 | S_2
Function	The word contents of the registers S_1, S_1+1 are compared with the word contents of the registers S_2, S_2+1.
Operation	S_1, S_1+1< = >S_2, S_2+1→Flag
Range of S_1	コ0000 to コ1576 b0000 to b0776 09000 to 09776 19000 to 19776 29000 to 29776 39000 to 39776
Range of S_2	コ0000 to コ1576 b0000 to b0776 09000 to 09776 19000 to 19776 29000 to 29776 39000 to 39776
Condition	When the input signal is ON state (not limited to OFF to ON change)

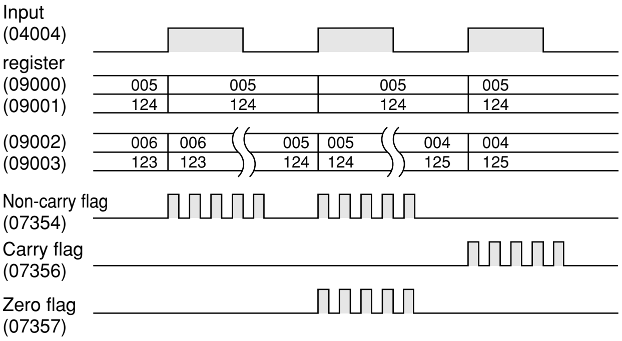
After operation	Contents of S_1, S_1+1	Unchanged				
	Contents of S_2, S_2+1	Unchanged				
	Flag	Contents of register	Zero 07357	Carry 07356	Error 07355	Non-carry 07354
		S_1, S_1+1>S_2, S_2+1	0	0	0	1
		S_1, S_1+1=S_2, S_2+1	1	0	0	1
		S_1, S_1+1<S_2, S_2+1	0	1	0	0

〔Example for use〕



Instruction	
STR	04004
F-12w	
	09000
	09002

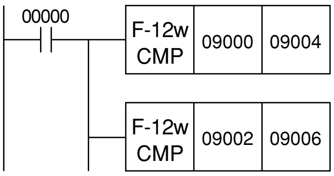
When the input condition 04004 is ON, the word contents of the registers 09000 and 09001 are compared with the word contents of the registers 09002 and 09003 and its results are set in the non-carry flag (07354), carry flag (07356), and the zero flag (07357). Registers 09000 to 09003 remain unchanged.



Note 1: Be sure to use even addresses for registers S_1 + S_2.
Similarity instructions: F-12, Fc12, Fc12w

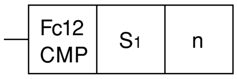
Reference

If the F-12w instruction were used in succession, 3 bytes or more data comparison may be done.



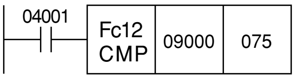
Instruction	
STR	00000
F-12w	
	09000
	09004
F-12w	
	09002
	09006

Fc12 CMP Compare register with constant (1 byte) (CoMPare)

Symbol	
Function	The contents of the register S_1 are compared with an octal constant "n".
Operation	$S_1 <=> n \rightarrow$ Flag
Range of S_1	コ0000 to コ1577 b0000 to b0777 09000 to 09777 19000 to 19777 29000 to 29777 39000 to 39777
Range of "n"	000 to $377_{(8)}$
Condition	When the input signal is ON (not limited to OFF to ON change)

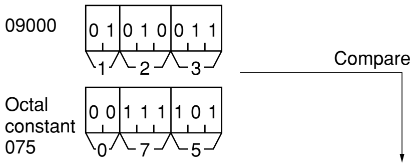
After operation						
	Contents of S1	Unchanged				
	Flag	Contents of register	Zero 07357	Carry 07356	Error 07355	Non-carry 07354
		$S_1 > n$	0	0	0	1
		$S_1 = n$	1	0	0	1
		$S_1 < n$	0	1	0	0

〔Example for use〕



```
Instruction
STR        04001
Fc12
           09000
             075
```

When the input condition 04001 is ON, the contents of the register 09000 are compared with the octal constant 075 and its results are set in the non-carry flag (07354), carry flag (07356), and zero flag (07357). The contents of the register 09000 remain unchanged after this operation.

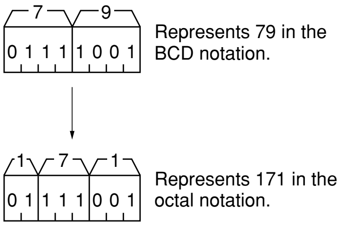


Zero	Carry	Error	Non-carry
07357	07356	07355	07354
0	0	0	1

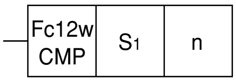
Similarity instructions: F-12, F-12w, Fc12w

Reference

Use an octal number in writing program with the Fc12 instruction. Octal number can express any bit pattern and it does not require annoying weight calculation. To compare with a BCD constant, convert it to an octal notation

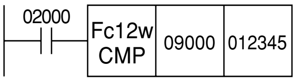

7
9
0 1 1 1 1 0 0 1
Represents 79 in the BCD notation.
1
7
1
0 1 1 1 1 0 0 1
Represents 171 in the octal notation.

Fc12w CMP Compare register with constant (1 word) (CoMPare)

Symbol	
Function	The word contents of the registers S_1 and S_1+1 are compared with an octal constant "n".
Operation	$S_1, S_1+1 <=> n \rightarrow$ Flag
Range of S_1	コ0000 to コ1576 b0000 to b0776 09000 to 09776 19000 to 19776 29000 to 29776 39000 to 39776
Range of "n"	000000 to 177777(8)
Condition	When the input signal is ON (not limited to OFF to ON change)

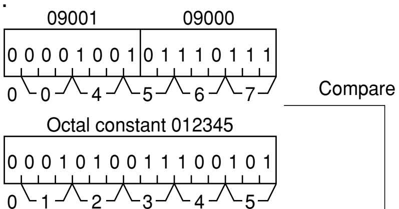
After operation						
	Contents of S1, S1+1	Unchanged				
	Flag	Contents of Register	Zero 07357	Carry 07356	Error 07355	Non-carry 07354
		$S_1, S_1+1 > n$	0	0	0	1
		$S_1, S_1+1 = n$	1	0	0	1
		$S_1, S_1+1 < n$	0	1	0	0

〔Example for use〕



Instruction	
STR	02000
F-12w	
	09000
	012345

When the input condition 02000 is ON, the word contents of the registers 09000 and 09001 are compared with the octal constant 012345 and its results are set in the non-carry flag (07354), carry flag (07356), and zero flag (07357). The contents of the registers 09000, 09001 remain unchanged. Operation takes place in the same timing as F-12w.

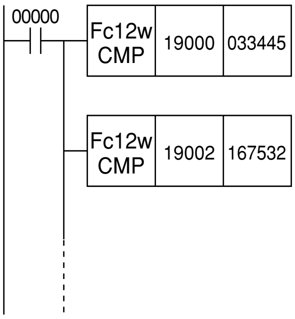


Zero 07357	Carry 07356	Error 07355	Non-carry 07354
0	1	0	0

Note 1: Be sure to use even addresses for register S_1.
Similarity instructions: F-12, F-12w, Fc12

Reference

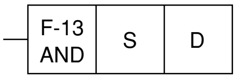
If the Fc12w instruction is used in succession, 4 bytes or more data comparison may be done.



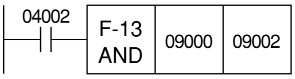
Instruction	
STR	00000
Fc12w	
	19000
	033445
Fc12w	
	19002
	167532

F-13 AND

AND register with register (1 byte) (AND)

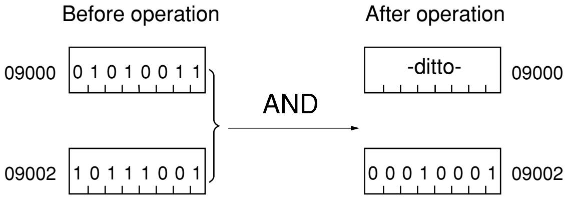
Symbol	 F-13 AND S D	
Function	The contents of the register S (8 bits) are ANDed with the contents of the register D (8 bits) and its result is stored in the register D.	
Operation	S∩D→D	
Range of S	Ↄ0000 to Ↄ1577 b0000 to b0777 09000 to 09777 19000 to 19777 29000 to 29777 39000 to 39777	
Range of D	Ↄ0000 to Ↄ1577 b0000 to b0777 09000 to 09777 19000 to 19777 29000 to 29777 39000 to 39777	
Condition	Rising edge of input signal (OFF to ON)	
After operation	Contents of S	Unchanged
	Contents of D	Result
	Flag	Unchanged

〔Example for use〕

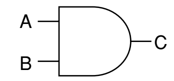


Instruction	
STR	04002
F-13	
	09000
	09002

When the input condition 04002 changes from OFF to ON, the 8 bits contents of the register 09000 are ANDed with the 8 bits contents of the register 09002 and its result is stored in the register 09002. The contents of the register 09000 remain unchanged.



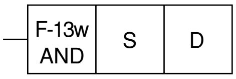
AND truth table

Symbol	A	B	C
 A B C	0	0	0
	1	0	0
	0	1	0
	1	1	1

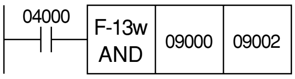
Similarity instructions: F-13w, Fc13, Fc13w

F-13w AND

AND register with register (1 word) (AND)

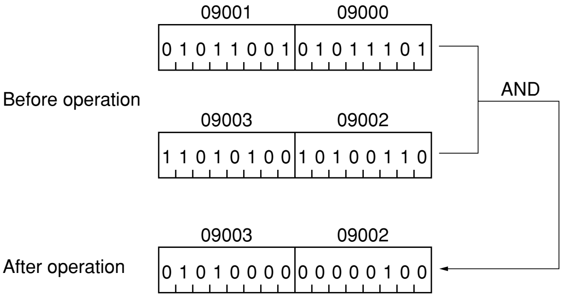
Symbol	 F-13w AND S D	
Function	The 16 bits contents of the registers S, S+1 are ANDed with the 16 bits contents of the registers D, D+1 and its result is stored in the registers D, D+1.	
Operation	S, S+1∩D, D+1→D, D+1	
Range of S	Ↄ0000 to Ↄ1576 b0000 to b0776 09000 to 09776 19000 to 19776 29000 to 29776 39000 to 39776	
Range of D	Ↄ0000 to Ↄ1576 b0000 to b0776 09000 to 09776 19000 to 19776 29000 to 29776 39000 to 39776	
Condition	Rising edge of input signal (OFF to ON)	
After operation	Contents of S, S+1	Unchanged
	Contents of D, D+1	Result
	Flag	Unchanged

〔Example for use〕



Instruction	
STR	04000
F-13w	
	09000
	09002

When the input condition 04000 changes from OFF to ON, the 16 bits contents of the registers 09000 and 09001 are ANDed with the 16 bits contents of the registers 09002 and 09003 and its result is stored in the registers 09002 and 09003. The contents of the registers 09000 and 09001 remain unchaged.



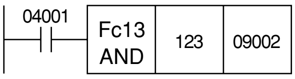
Note 1: Be sure to use even addresses for registers S and D.
Similarity instructions: F-13, Fc13, Fc13w

Fc13 AND

AND register with constant (1 byte) (AND)

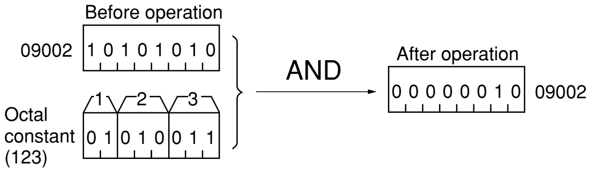
Symbol	Fc13 AND n D	
Function	An octal constant "n" is ANDed with the contents of the register D and its result is stored in the register D.	
Operation	n∩D→D	
Range of "n"	000 to 377$_{(8)}$	
Range of D	⊐0000 to ⊐1577 b0000 to b0777 09000 to 09777 19000 to 19777 29000 to 29777 39000 to 39777	
Condition	Rising edge of input signal (OFF to ON)	
After operation	Contents of D	Result
	Flag	Unchanged

〔Example for use〕

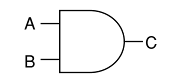


Instruction	
STR	04001
Fc13	
	123
	09002

When the input condition 04001 changes from OFF to ON, the octal constant 123 is ANDed with the contents of the register 09002 and its result is stored in the register 09002.



AND truth table

Symbol	A	B	C
	0	0	0
	1	0	0
	0	1	0
	1	1	1

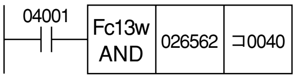
Similarity instructions: F-13, F-13w, Fc13w

Fc13w AND

AND register with constant (1 word) (AND)

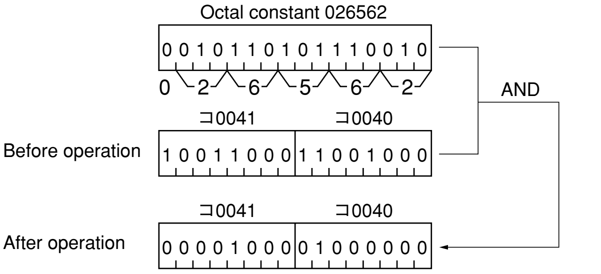
Symbol	Fc13w AND n D	
Function	An octal constant "n" is ANDed with the 16 bits contents of the registers D and D+1 and its result is stored in the registers D and D+1.	
Operation	n∩D, D+1→D, D+1	
Range of "n"	000000 to177777$_{(8)}$	
Range of D	⊐0000 to ⊐1576 b0000 to b0776 09000 to 09776 19000 to 19776 29000 to 29776 39000 to 39776	
Condition	Rising edge of input signal (OFF to ON)	
After operation	Contents of D, D+1	Result
	Flag	Unchanged

〔Example for use〕



Instruction	
STR	04001
Fc13w	
	026562
	⊐0040

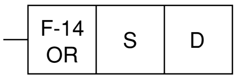
When the input condition 04001 changes from OFF to ON, the octal constant 026562 is ANDed with the16 bits contents of the registers ⊐0040 and ⊐0041 and its result is stored in the registers ⊐0040 and ⊐0041.



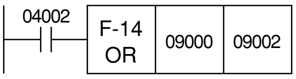
Note 1: Be sure to use even addresses for registers D.
Similarity instructions: F-13, F-13w, Fc13

F-14 OR

OR register with register (1 byte) (OR)

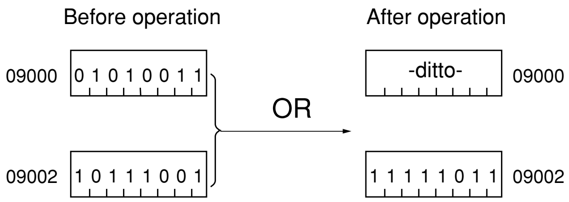
Symbol		
Function	The contents of the register S (8 bits) are ORed with the contents of the register D (8 bits) and its result is stored in the register D.	
Operation	S∪D→D	
Range of S	コ0000 to コ1577 b0000 to b0777 09000 to 09777 19000 to 19777 29000 to 29777 39000 to 39777	
Range of D	コ0000 to コ1577 b0000 to b0777 09000 to 09777 19000 to 19777 29000 to 29777 39000 to 39777	
Condition	Rising edge of input signal (OFF to ON)	
After operation	Contents of S	Unchanged
	Contents of D	Result
	Flag	Unchanged

〔Example for use〕

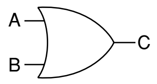


Instruction	
STR	04002
F-14	
	09000
	09002

When the input condition 04002 changes from OFF to ON, the 8 bits contents of the register 09000 are ORed with the 8 bits contents of the register 09002 and its result is stored in the register 09002. The contents of the register 09000 remain unchanged.



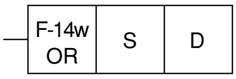
OR truth table

Symbol	A	B	C
	0	0	0
	1	0	1
	0	1	1
	1	1	1

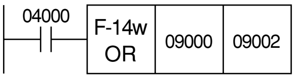
Similarity instructions: F-14w, Fc14, Fc14w

F-14w OR

OR register with register (1 word) (OR)

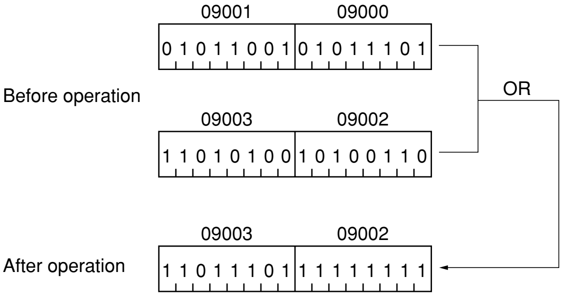
Symbol		
Function	The 16 bits contents of the registers S, S+1 are ORed with the 16 bits contents of the registers D, D+1 and its result is stored in the registers D, D+1.	
Operation	S, S+1∪D, D+1→D, D+1	
Range of S	コ0000 to コ1576 b0000 to b0776 09000 to 09776 19000 to 19776 29000 to 29776 39000 to 39776	
Range of D	コ0000 to コ1576 b0000 to b0776 09000 to 09776 19000 to 19776 29000 to 29776 39000 to 39776	
Condition	Rising edge of input signal (OFF to ON)	
After operation	Contents of S, S+1	Unchanged
	Contents of D, D+1	Result
	Flag	Unchanged

〔Example for use〕



Instruction	
STR	04000
F-14w	
	09000
	09002

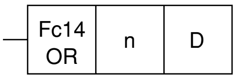
When the input condition 04000 changes from OFF to ON, the 16 bits contents of the registers 09000 and 09001 are ORed with the 16 bits contents of the registers 09002 and 09003 and its result is stored in the registers 09002 and 09003. The contents of the registers 09000 and 09001 remain unchaged.



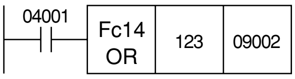
Note 1: Be sure to use even addresses for registers S and D.
Similarity instructions: F-14, Fc14, Fc14w

Fc14 OR

OR register with constant (1 byte) (OR)

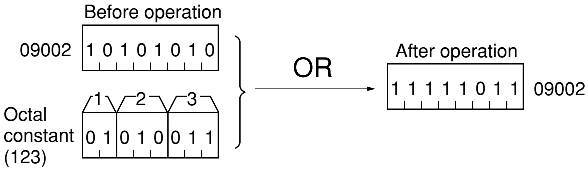
Symbol		
Function	An octal constant "n" is ORed with the contents of the register D and its result is stored in the register D.	
Operation	n∪D→D	
Range of "n"	000 to 377(8)	
Range of D	コ0000 to コ1577 b0000 to b0777 09000 to 09777 19000 to 19777 29000 to 29777 39000 to 39777	
Condition	Rising edge of input signal (OFF to ON)	
After operation	Contents of D	Result
	Flag	Unchanged

〔Example for use〕

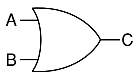


Instruction	
STR	04001
Fc14	
	123
	09002

When the input condition 04001 changes from OFF to ON, the octal constant 123 is ORed with the contents of the register 09002 and its result is stored in the register 09002.



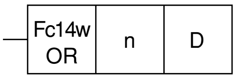
OR truth table

Symbol	A	B	C
	0	0	0
	1	0	1
	0	1	1
	1	1	1

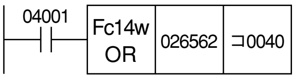
Similarity instructions: F-14, F-14w, Fc14w

Fc14w OR

OR register with constant (1 word) (OR)

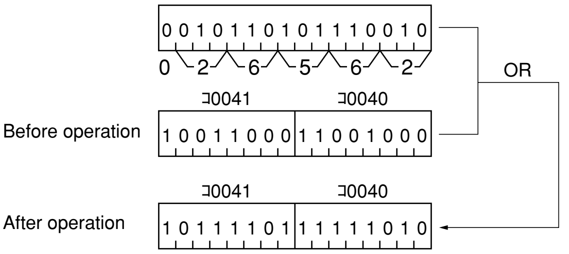
Symbol		
Function	An octal constant "n" is ORed with the 16 bits contents of the registers D and D+1 and its result is stored in the registers D and D+1.	
Operation	n∪D, D+1→D, D+1	
Range of "n"	000000 to 177777(8)	
Range of D	コ0000 to コ1576 b0000 to b0776 09000 to 09776 19000 to 19776 29000 to 29776 39000 to 39776	
Condition	Rising edge of input signal (OFF to ON)	
After operation	Contents of D, D+1	Result
	Flag	Unchanged

〔Example for use〕



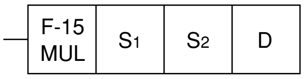
Instruction	
STR	04001
Fc14w	
	026562
	コ0040

When the input condition 04001 changes from OFF to ON, the octal constant 026562 is ORed with the16 bits contents of the registers コ0040 and コ0041 and its result is stored in the registers コ0040 and コ0041.



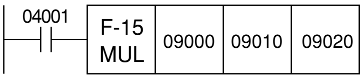
Note 1: Be sure to use even addresses for registers D.
Similarity instructions: F-14, F-14w, Fc14

F-15 MUL Multiply register by register (BCD 4 digits) (MULtiply)

Symbol	 F-15 MUL S_1 S_2 D					
Function	The contents of the registers S_1 and S_1+1 (BCD 4 digits) are multiplied by the contents of the registers S_2 and S_2+1 (BCD 4 digits) and its result is stored in the 4 bytes area starting from the register D.					
Operation	(S_1, S_1+1) × (S_2, S_2+1) →D, D+1, D+2, D+3					
Range of S_1	コ0000 to コ1576 b0000 to b0776 09000 to 09776 19000 to 19776 29000 to 29776 39000 to 39776					
Range of S_2	コ0000 to コ1576 b0000 to b0776 09000 to 09776 19000 to 19776 29000 to 29776 39000 to 39776					
Range of D	コ0000 to コ1574 b0000 to b0774 09000 to 09774 19000 to 19774 29000 to 29774 39000 to 39774					
Condition	Rising edge of input signal (OFF to ON)					
After operation	Contents of S_1, S_1+1	Unchanged				
	Contents of S_2, S_1+2	Unchanged				
	Contents of D	Result (in ones and tens)	Unchanged if the contents of registers S_1, S_1+1, S_2, and S_2+1 are not BCD code.			
	Contents of D+1	Result (in hundreds and thousands)				
	Contents of D+2	Result (in ten thousands and hundred thousands)				
	Contents of D+3	Result (in millions and ten millions)				
	Flag	S_1, S_1+1, S_2, S_2+1	Zero 07357	Carry 07356	Error 07355	Non-carry 07354
		BCD code	0	0	0	0
		Not BCD code			1	

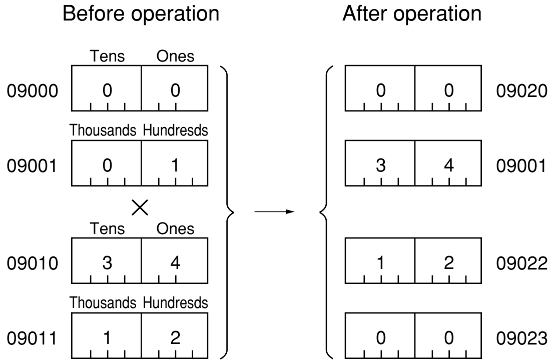
Similarity instructions: Fc15

〔Example for use〕



Instruction	
STR	04001
F-15	09000 09010 09020

When the input condition 04001 changes from OFF to ON, the BCD 4 digits in registers 09000 and 09001 are multiplied by the BCD 4 digits of registers 09010 and 09011 and its result is stored in the 4 bytes area starting from the registers 09020.



The above example shows the operation of 100×1234=123400.

Fc15
MUL

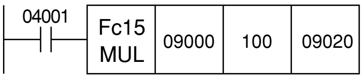
Multiply register (BCD 4 digits) by constant (BCD 3 digits) (MULtiply)

Symbol	Fc15 MUL | S_1 | n | D
Function	The contents of the registers S_1 and S_1+1 (BCD 4 digits) are multiplied by a 3 digits BCD constant "n" and its result is stored in the 4 bytes area starting from the register D.
Operation	(S_1, S_1+1) × n→D, D+1, D+2, D+3
Range of S_1	コ0000 to コ1576 b0000 to b0776 09000 to 09776 19000 to 19776 29000 to 29776 39000 to 39776
Range of "n"	000 to 999
Range of D	コ0000 to コ1574 b0000 to b0774 09000 to 09774 19000 to 19776 29000 to 29774 39000 to 39774
Condition	Rising edge of input signal (OFF to ON)

After operation	Contents of S_1, S_1+1	Unchanged	
	Contents of D	Result (in ones and tens)	Unchanged when the contents of registers S_1 and S_1+1 are not BCD code.
	Contents of D+1	Result (in hundreds and thousands)	
	Contents of D+2	Result (in ten thousands and hundred thousands)	
	Contents of D+3	Result (in millions and ten millions)	

Flag	S_1, S_1+1	Zero 07357	Carry 07356	Error 07355	Non-carry 07354
	BCD code	0	0	0	0
	Not BCD code			1	

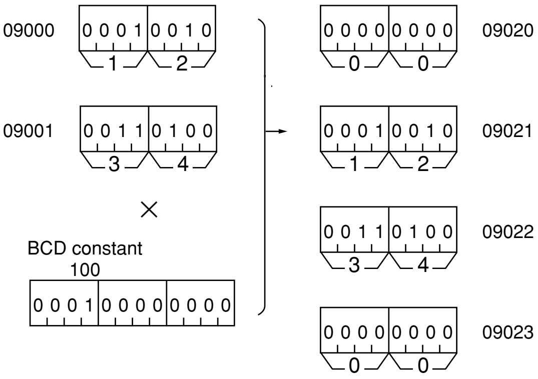
〔Example for use〕



Instruction	
STR	04001
Fc15	
	09000
	100
	09020

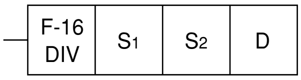
When the input condition 04001 changes from OFF to ON, the BCD 4 digits in registers 09000 and 09001 are multiplied by the 3 digits BCD constant 100 and its result is stored in the 4 bytes area starting from the registers 09020.

3412×100=341200

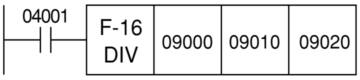


Similarity instructions: F-15

F-16 DIV Divide register (BCD 4 digits) by register (BCD 2 digits) (DIVide)

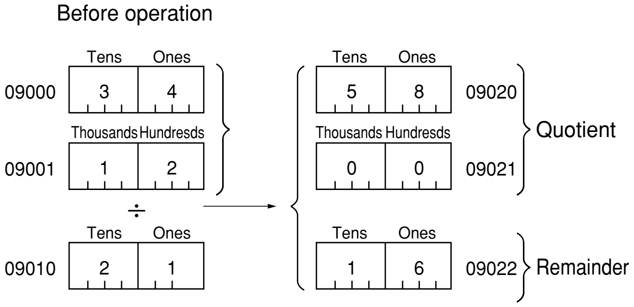
<table>
<tr><td colspan="2">Symbol</td><td colspan="5"> </td></tr>
<tr><td colspan="2">Function</td><td colspan="5">The contents of the registers S_1 and S_1+1 (BCD 4 digits) are divied by the contents of the register S_2 (BCD 2 digits) and the quotient is stored in the 2 bytes area starting from the register D and the remainder in the third byte area.</td></tr>
<tr><td colspan="2">Operation</td><td colspan="5">(S_1, S_1+1) ÷S_2→D, D+1, D+2</td></tr>
<tr><td colspan="2">Range of S_1</td><td colspan="5">コ0000 to コ1576
b0000 to b0776
09000 to 09776
19000 to 19776
29000 to 29776
39000 to 39776</td></tr>
<tr><td colspan="2">Range of S_2</td><td colspan="5">コ0000 to コ1577
b0000 to b0777
09000 to 09777
19000 to 19777
29000 to 29777
39000 to 39777</td></tr>
<tr><td colspan="2">Range of D</td><td colspan="5">コ0000 to コ1575
b0000 to b0775
09000 to 09775
19000 to 19775
29000 to 29775
39000 to 39775</td></tr>
<tr><td colspan="2">Condition</td><td colspan="5">Rising edge of input signal (OFF to ON)</td></tr>
<tr><td rowspan="8">After operation</td><td>Contents of S_1, S_1+1</td><td colspan="5">Unchanged</td></tr>
<tr><td>Contents of S_2</td><td colspan="5">Unchanged</td></tr>
<tr><td>Contents of D</td><td colspan="2">Quotient (in ones and tens)</td><td colspan="3" rowspan="3">Unchanged when the contents of registers, S_1, S_1+1, and S_2 are not BCD code or the contents of S_2 is 00.</td></tr>
<tr><td>Contents of D+1</td><td colspan="2">Quotient (in hundreds and thousands)</td></tr>
<tr><td>Contents of D+2</td><td colspan="2">Remainder</td></tr>
<tr><td rowspan="3">Flag</td><td>S_1, S_1+1, S_2</td><td>Zero
07357</td><td>Carry
07356</td><td>Error
07355</td><td>Non-carry
07354</td></tr>
<tr><td>BCD code</td><td rowspan="2">0</td><td rowspan="2">0</td><td>0</td><td rowspan="2">0</td></tr>
<tr><td>Not BCD code
If S_2 is 00</td><td>1</td></tr>
</table>

〔Example for use〕



Instruction	
STR	04001
F-16	09000 09010 09020

When the input condition 04001 changes from OFF to ON, the BCD 4 digits in registers 09000 and 09001 are divided by the BCD 2 digits in registers 09010 and the quotient is stored in the 2 bytes area starting from the registers 09020 with the remainder in the 3 bytes area.



The above example shows the operation of 1234 divided by 21 equals 58 with the remainder of 16.

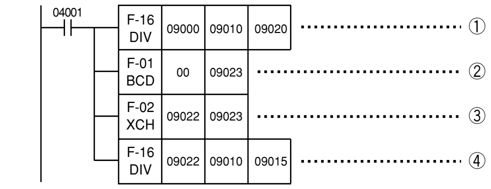
- If denominator is not greater than numerator (S_1<S_2, S_1+1=0), the quotient (contents of D, D+1) is 0 and remainder (contents of D+2) is numerator (contents of S_1). For instance, 20/30 will produce the result of 0 with a remainder of 20.

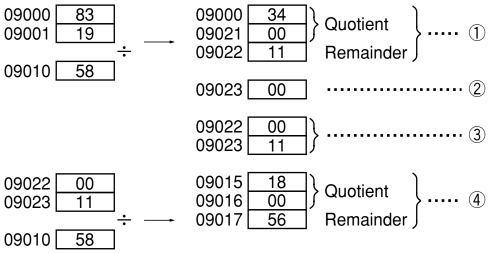
Similarity insturctions: Fc16

Reference

To obtain a result of 2 places under the decimal point, the following programming is suggested.

Example : 1983/58=34.18 ... remainder 0.56





Instruction	
S T R	04001
F -16	09000 09010 09020
F -01	00 09023
F -02	09022 09023
F -16	09022 09010 09015

① When the input condition 04001 changes from OFF to ON, the contents of registers 09000 and 09001 are divided by the contents of the register 09010 and the quotient is stored in 09020 and 09021 and remainder is stored in 09022.

② Enter data 00 in 09023.

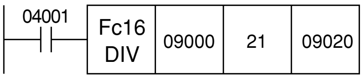
③ The contents of 09022 are exchanged with 09023 and a remainder is converted into thousands and hundreds.

④ The data in ③ is divided by the contents of 09010 again, and a quotient is stored in 09015 and 09016 with a remainder stored in 09017. The result under 2 digits of decimal place is then stored in 09015.

Fc16 DIV — Divide register (BCD 4 digits) by constant (BCD 2 digits) (DIVide)

<table>
<tr><td colspan="2">Symbol</td><td colspan="5">— Fc16 DIV | S_1 | n | D</td></tr>
<tr><td colspan="2">Function</td><td colspan="5">The contents of the registers S_1 and S_1+1 (BCD 4 digits) are divided by a 2 digits BCD contents "n" and the quotient is stored in the 2 bytes area starting from the register D and the remainder in the 3 bytes area.</td></tr>
<tr><td colspan="2">Operation</td><td colspan="5">$(S_1, S_1+1) \div n \rightarrow D, D+1, D+2$</td></tr>
<tr><td colspan="2">Range of S_1</td><td colspan="5">コ0000 to コ1576
b0000 to b0776
09000 to 09776
19000 to 19776
29000 to 29776
39000 to 39774</td></tr>
<tr><td colspan="2">Range of "n"</td><td colspan="5">00 to 99</td></tr>
<tr><td colspan="2">Range of D</td><td colspan="5">コ0000 to コ1575
b0000 to b0775
09000 to 09775
19000 to 19775
29000 to 29775
39000 to 39775</td></tr>
<tr><td colspan="2">Condition</td><td colspan="5">Rising edge of input signal (OFF to ON)</td></tr>
<tr><td rowspan="7">After operation</td><td>Contents of S_1, S_1+1</td><td colspan="5">Unchanged</td></tr>
<tr><td>Contents of D</td><td colspan="2">Quotient (in ones and tens)</td><td colspan="3" rowspan="3">Unchanged when the contents of registers S_1 and S_1+1 are not BCD code or the contents of "n" is 00.</td></tr>
<tr><td>Contents of D+1</td><td colspan="2">Quotient (in hundreds and thousands)</td></tr>
<tr><td>Contents of D+2</td><td colspan="2">Remainder</td></tr>
<tr><td rowspan="3">Flag</td><td>S_1, S_1+1, "n"</td><td>Zero
07357</td><td>Carry
07356</td><td>Error
07355</td><td>Non-carry
07354</td></tr>
<tr><td>BCD code</td><td rowspan="2">0</td><td>0</td><td rowspan="2">0</td><td rowspan="2">0</td></tr>
<tr><td>Not BCD code
If "n" is 00</td><td>1</td></tr>
</table>

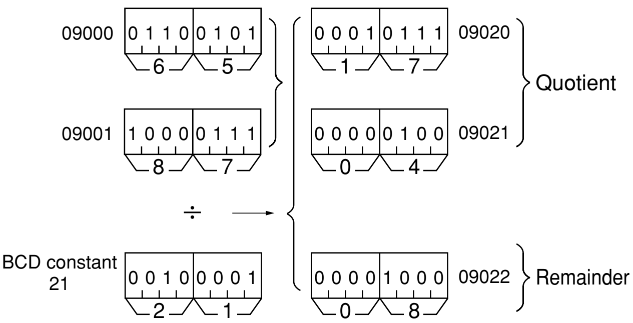
〔Example for use〕



Instruction	
STR	04001
Fc16	
	09000
	21
	09020

When the input condition 04001 changes from OFF to ON, the BCD 4 digits in registers 09000 and 09001 are divided by the BCD constant 21 and the quotient is stored in the 2 bytes area starting from the register 09020 with the remainder in the 3 bytes area.

8765÷21=417……8

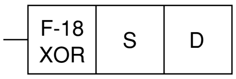


- If denominator is not greater than numerator (S_1<n, S_1+1=0), the quotient (contents of D, D+1) is 0 and remainder (contents of D+2) is numerator (contents of S_1). For instance, 20/30 will produce the result of 0 with a remainder of 20.

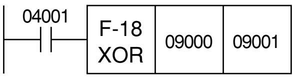
Similarity instructions: F-16

F-18 XOR

Exclusive OR register with register (1 byte) (eXclusive OR)

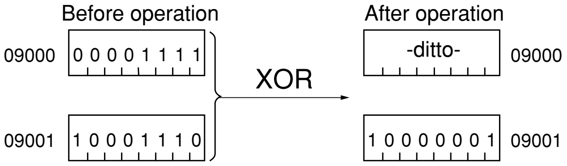
Symbol		
Function	The contents of the register S are XORed with the contents of the register D and its result is stored in the register D.	
Operation	S⊕D→D	
Range of S	⊐0000 to ⊐1577 b0000 to b0777 09000 to 09777 19000 to 19777 29000 to 29777 39000 to 39777	
Range of D	⊐0000 to ⊐1577 b0000 to b0777 09000 to 09777 19000 to 19777 29000 to 29777 39000 to 39777	
Condition	Rising edge of input signal (OFF to ON)	
After operation	Contents of S	Unchanged
	Contents of D	Result
	Flag	Unchanged

〔Example for use〕



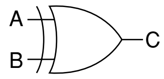
Instruction	
STR	04001
F-18	
	09000
	09001

When the input condition 04001 changes from OFF to ON, the contents of the register 09000 are XORed with the contents of the register 09001 and its result is stored in the register 09001. The contents of the register 09000 remain unchanged.



Bit matched in 09000 and 09001 (0 for 0, 1 for 1) is turned to 0 and unmatched bit (0 and 1) is turned to 1.

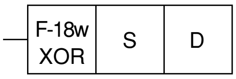
Exclusive OR truth table

Symbol	A	B	C
	0	0	0
	1	0	1
	0	1	1
	1	1	0

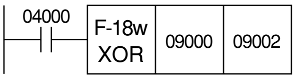
Similarity instructions: F-18w, Fc18, Fc18w

F-18w XOR

Exclusive OR register with register (1 word) (eXclusive OR)

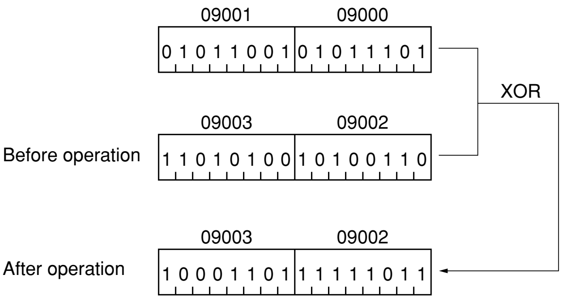
Symbol		
Function	The 16 bits contents of the registers S, S+1 are XORed with the 16 bits contents of the registers D, D+1 and its result is stored in the registers D, D+1.	
Operation	S, S+1⊕D, D+1→D, D+1	
Range of S	⊐0000 to ⊐1576 b0000 to b0776 09000 to 09776 19000 to 19776 29000 to 29776 39000 to 39776	
Range of D	⊐0000 to ⊐1576 b0000 to b0776 09000 to 09776 19000 to 19776 29000 to 29776 39000 to 39776	
Condition	Rising edge of input signal (OFF to ON)	
After operation	Contents of S, S+1	Unchanged
	Contents of D, D+1	Result
	Flag	Unchanged

〔Example for use〕



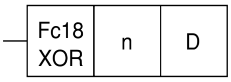
Instruction	
STR	04000
F-18w	
	09000
	09002

When the input condition 04000 changes from OFF to ON, the 16 bits contents of the registers 09000 and 09001 are XORed with the 16 bits contents of the registers 09002 and 09003 and its result is stored in the registers 09002 and 09003. The contents of the registers 09000 and 09001 remain unchanged.

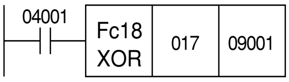


Note 1: Be sure to use even addresses for registers S and D.
Similarity instructions: F-18, Fc18, Fc18w

Fc18 XOR Exclusive OR register with constant (1 byte) (eXclusive OR)

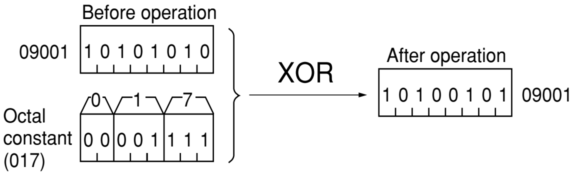
Symbol		
Function	An octal constant "n" is XORed with the contents of the register D and its result is stored in the register D.	
Operation	n⊕D→D	
Range of "n"	000 to 377(8)	
Range of D	⊐0000 to ⊐1577 b0000 to b0777 09000 to 09777 19000 to 19777 29000 to 29777 39000 to 39777	
Condition	Rising edge of input signal (OFF to ON)	
After operation	Contents of D	Result
	Flag	Unchanged

〔Example for use〕

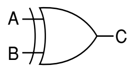


Instruction	
STR	04001
Fc18	
	017
	09001

When the input condition 04001 changes from OFF to ON, the octal constant 017 is XORed with the contents of the register 09001 and its result is stored in the register 09001.

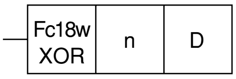


Exclusive OR truth table

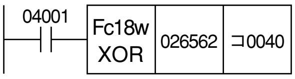
Symbol	A	B	C
	0	0	0
	1	0	1
	0	1	1
	1	1	0

Similarity instructions: F-18, F-18w, Fc18w

Fc18w XOR Exclusive OR register with constant (1 word) (eXclusive OR)

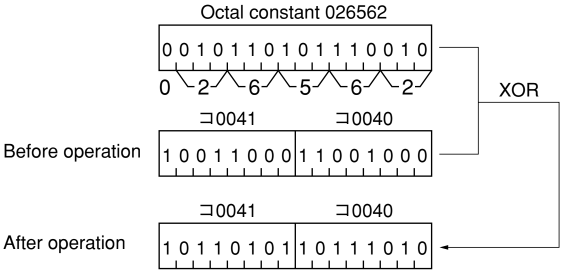
Symbol		
Function	An octal constant "n" is XORed with the 16 bits contents of the registers D and D+1 and its result is stored in the registers D and D+1.	
Operation	n⊕D, D+1→D, D+1	
Range of "n"	000000 to 177777(8)	
Range of D	⊐0000 to ⊐1576 b0000 to b0776 09000 to 09776 19000 to 19776 29000 to 29776 39000 to 39776	
Condition	Rising edge of input signal (OFF to ON)	
After operation	Contents of D, D+1	Result
	Flag	Unchanged

〔Example for use〕



Instruction	
STR	04001
Fc18w	
	026562
	⊐0040

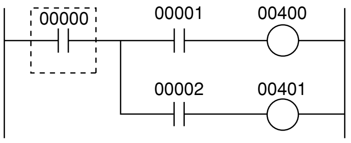
When the input condition 04001 changes from OFF to ON, the octal constant 026562 is XORed with the16 bits contents of the registers ⊐0040 and ⊐0041 and its result is stored in the registers ⊐0040 and ⊐0041.



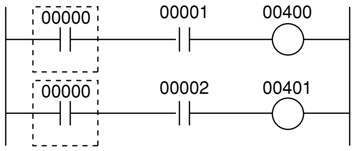
Note 1: Be sure to use even addresses for registers D.
Similarity instructions: F-18, F-18w, Fc18

F-30 MCS	**Set master control (Master Control Set)**
F-31 MCR	**Reset master control (Master Control Reset)**

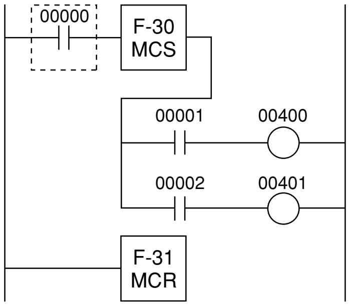
MCS and MCR can be used when the circuit is branched to a multiple number of outputs after common operational condition.



(1) In case of a relay board



(2) In case of MCS and MCR not in use



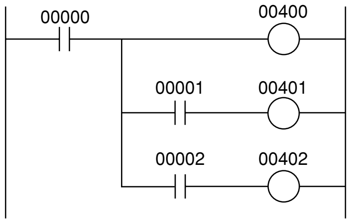
```
        S T R   00000
M C S → F -30
        S T R   00001
        O U T   00400
        S T R   00002
        O U T   00401
M C R → F -31
```

(3) In case of MCS and MCR in use

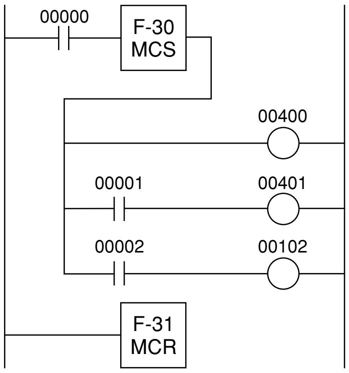
When the F-30 (MCS) instruction is programmed, the contents of the accumulator are stored in the CPU internal register, and succeeding instructions are ANDed with the contents of the CPU internal register until the F-31 (MCR) instruction is met. The F-31 (MCR) instruction indicates the end of AND operation. It will help to simplify the program when the operational condition indicated in a block is complicated or many branches are set after the common operational condition.

Note 1: Do not directly connect the F-30 (MCS) derived bus line with OUT, TMR, and CNT instructions or application instruction.

(1) In case of the relay board

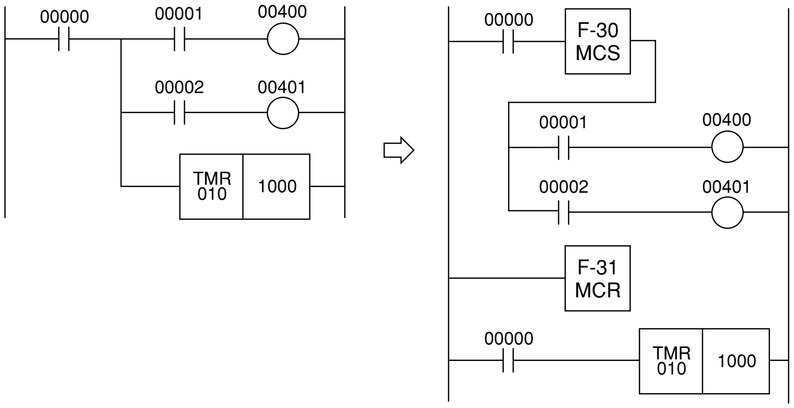


(2) MCS, MCR prohibited program

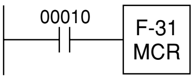


Instruction	
S T R	00000
F -30	
O U T	00400
S T R	00001
O U T	00401
S T R	00002
O U T	00402
F -31	

Must be programmed in the following manner.

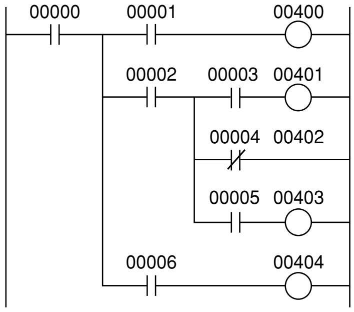


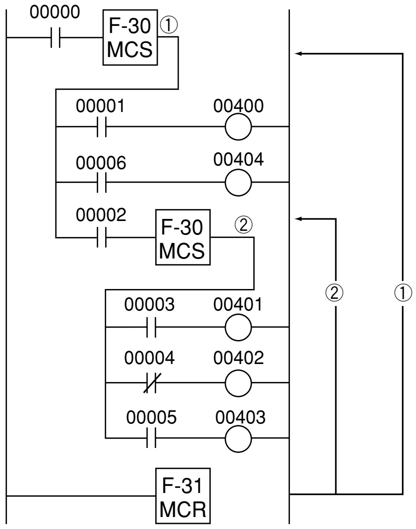
Note 2: The F-31 (MCR) instruction is an unconditional instruction.



It prohibits the program like shown above.

Another MCS may be used in between MCS and MCR.





The relay board ladder chart shown at left can be programmed in the following manner using MCS and MCR. However, there may be a need of changing the program sequence in the example(※).

Instruction		
S T R	00000	
F -30		
S T R	00001	
O U T	00400	
S T R	00006	※
O U T	00404	
S T R	00002	
F-30		
S T R	00003	
O U T	00401	
S T R N O T	00004	
O U T	00402	
S T R	00005	
O U T	00403	
F -31		

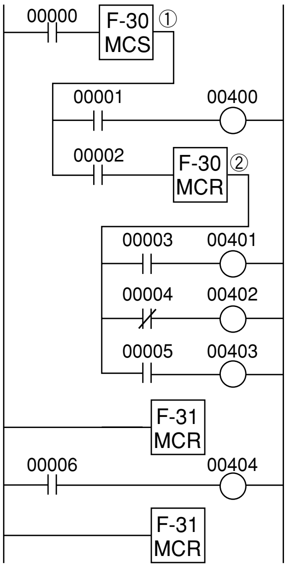
- The F-31 (MCR) instruction indicates the termination of the preceding F-30 (MCS) instruction ; ① and ② in the example.

Note 3: The desired circuit would not be established if programmed in the following way.

	Instruction		
	S T R	00000	
M C S→	F -30		
	S T R	00001	
	O U T	00400	
	S T R	00002	
M C S→	F-30		
	S T R	00003	
	O U T	00401	
	S T R N O T	00004	
	O U T	00402	
	S T R	00005	
	O U T	00403	
M C R→	F -31		
	S T R	00006	※
	O U T	00404	
M C R→	F -31		

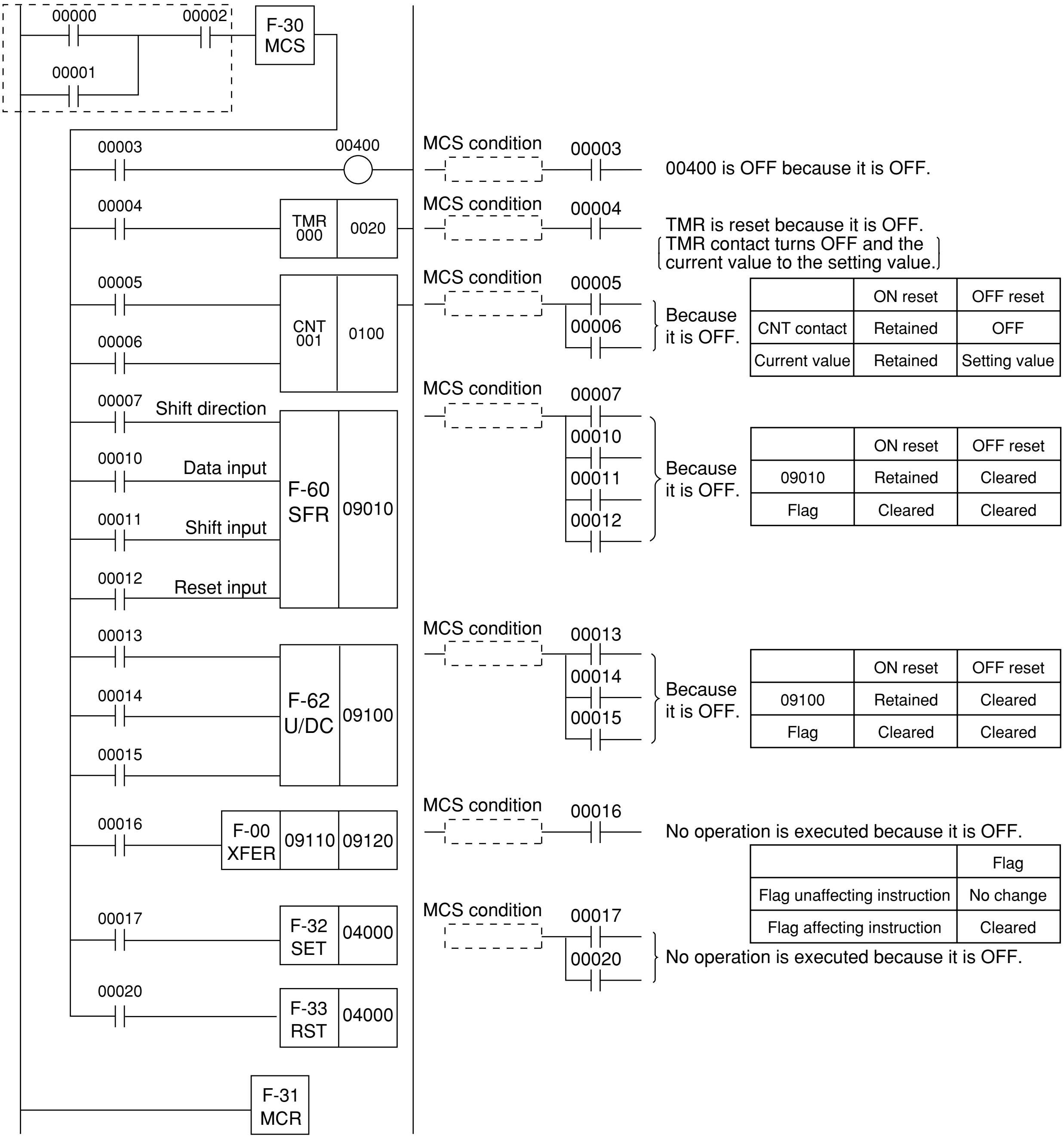
This MCR is insignificant

MCS of ① and ② has been terminated with this MCR.



- Although it is possible to insert MCS as many times as required between MCS and MCR (※), the range of MCS terminates with MCR marked with an asterisk (※).

When the MCS condition (enclosed with dotted line) is OFF, the instructions existing between MCS and MCR will be handled in the following manner.



00400 is OFF because it is OFF.

TMR is reset because it is OFF.
(TMR contact turns OFF and the current value to the setting value.)

Because it is OFF.

	ON reset	OFF reset
CNT contact	Retained	OFF
Current value	Retained	Setting value

Because it is OFF.

	ON reset	OFF reset
09010	Retained	Cleared
Flag	Cleared	Cleared

Because it is OFF.

	ON reset	OFF reset
09100	Retained	Cleared
Flag	Cleared	Cleared

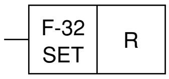
No operation is executed because it is OFF.

	Flag
Flag unaffecting instruction	No change
Flag affecting instruction	Cleared

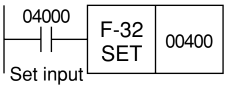
No operation is executed because it is OFF.

Note 4: With a CNT, F-60, or F-62 instruction, it is possible to ON reset or OFF reset the reset condition using the system memory #202. In case of the OFF reset, reset is done by MCS.

F-32 SET Set coil

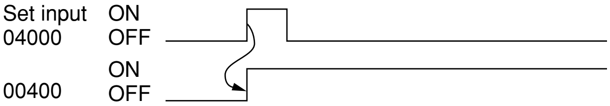
Symbol		
Function	Activate a coil specified by relay number R when the set input is turned ON.	
Operation	Activate the R specified by F-32.	
Range of OUT	00400 to 15777	
Condition	Set input is at ON (not limited to OFF to ON change)	
After operation	Contents of R	ON
	Flag	Unchanged

〔Example for use〕



Instruction	
STR	04000
F-32	
	00400

When the set input of 04000 is ON, this instruction sets ON 00400. Once sets ON, 00400 remains ON after the set input is set to OFF.
When the set input of 04000 is OFF, this instruction does not affect the state of 00400.



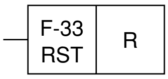
Note 1: If the relay specified by the F-32 (SET) instruction is within the latched relay area, it maintains the last state when the power is recovered from power failure. If the specified relay is outside the latched relay area, it is reset upon power recovery.

Note 2: The relay assigned by F-32 (SET) instruction is set to latch output when the JW10 stops operation (system memory #206 = 55(H)), it maintains status before stop operation. If the assigned relay is set to reset output when the JW10 stops operation system memory #206 = 00(H)), it resets at stoppage.

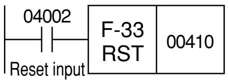
Note 3: It is recommended that the F-32 (SET) instruction be used in conjunction with the F-33 instruction described on the next item.

Note 4: The F-32 (SET) and F-33 (RST) instructions placed between the F-30 (MCS) and F-31 (MCR) instructions are not operative if the operational condition of the F-30 (MCS) is OFF.

F-33 RST Reset coil

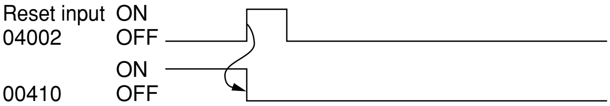
Symbol		
Function	Deactivate a coil specified by relay number R when the reset input is turned ON.	
Operation	Deactivate the R specified by F-33.	
Range of R	00400 to 15777	
Condition	Reset input is at ON (not limited to OFF to ON change)	
After operation	Contents of R	OFF
	Flag	Unchanged

〔Example for use〕



Instruction	
STR	04002
F-33	
	00410

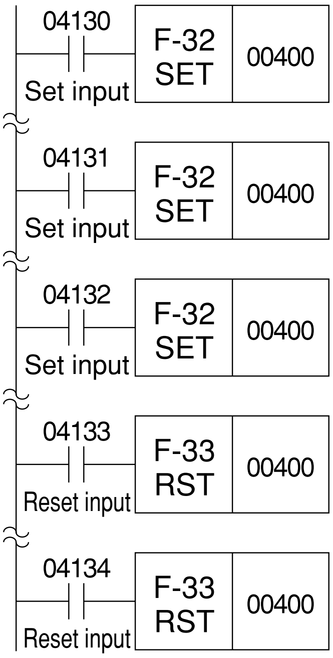
When the reset input of 04002 is ON, this instruction sets OFF 00410. Once sets OFF, 00410 remains OFF after the reset input is set to OFF.
When the reset input of 04002 is OFF, this instruction does not affect the state of 00410.



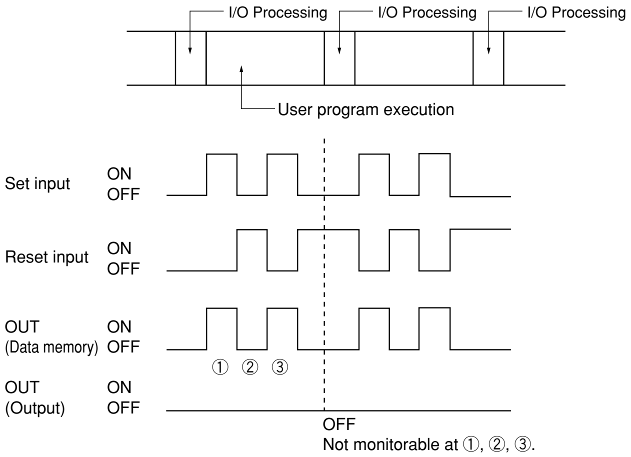
Note 1: If the relay specified by the F-33 (RST) instruction is within the latched relay area, the last state is retained when the power is recovered from power failure. If the specified relay is outside the latched relay area, it is reset upon power recovery.

Note 2: The relay assigned by F-33 (RST) instruction is set to latch output when the JW10 stops operation (system memory #206 = 55(H)), it maintains status before stop operation. If the assigned relay is set to reset output when the JW10 stops operation system memory #206 = 00(H)).

- The F-32 (SET) and F-33 (RST) instructions allow to output relay to be controlled under more than 1 condition.



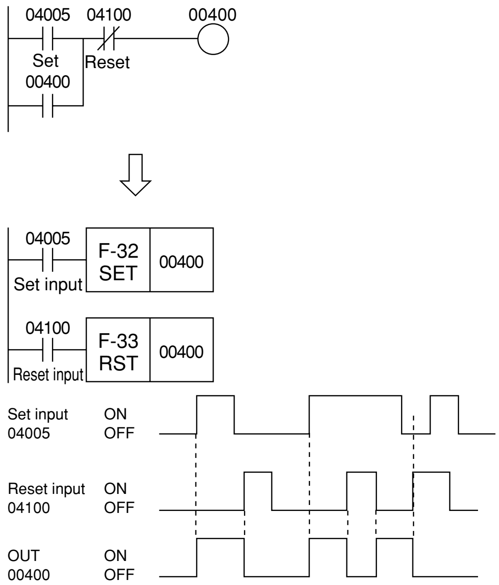
- If the set and reset inputs are turned ON or OFF more than 1 time in 1 scan cycle, the data memory used as the relay R is set and reset repeatedly in 1 scan cycle.
  The result of the relay R (ON or OFF) appearing immediately before I/O processing should be sent to the output terminal, however.



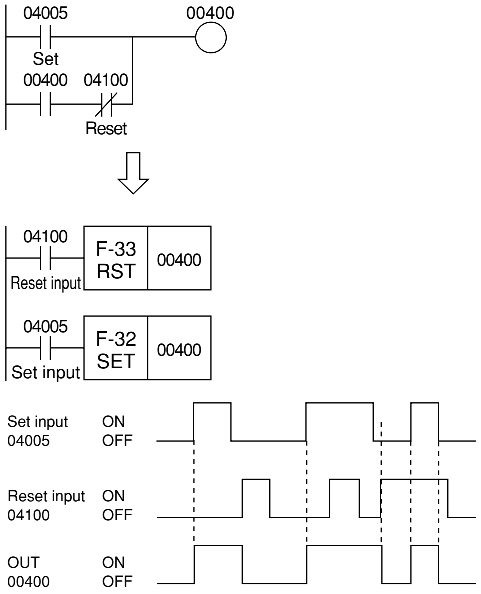
During execution of user program, even the data memory turns ON/OFF more than one time, only the result just before I/O processing can be monitored.

- It is recommended that the F-32 (SET) and F-33 (RST) instruction be used in pair. It will simplify the self holding circuit.

Reset priority self holding circuit

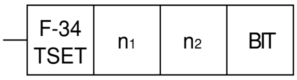


Set priority self holding circuit

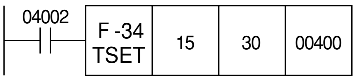


F-34 TSET Comparison with current value of clock (specified relay set)

(This instruction is able to program by JW-1424K/1442K/ 1624K/1642K and is not able to program by JW-1324K/1342K.)

Symbol		
Function	Compares the constants n_1 (hours) and n_2 (minutes) with the current value of the clock and sets (turns ON) the specified BIT (relay) if they match.	
Operation	Compares the current value of the clock with n_1 and n_2, and turns ON the relay if the comparison result shows a match.	
Range of n_1	00 to 23 (decimal)	
Range of n_2	00 to 59 (decimal)	
Range of BIT	00400 to 15777	
Condition	When the input signal is ON (not limited to OFF to ON change)	
After operation	Contents of n_1	Unchanged
	Contents of n_2	Unchanged
	Contents of BIT	When the assigned time does not match with the clock's current value, the bit condition does not change. When the assigned time matches with the clock's current value, this bit turns ON.
	Flag	Unchanged

〔Example for use〕



Instruction	
STR	04002
F-34	
	15
	30
	00400

When input condition 04002 is ON and at 15 hour 30 minutes, relay 00400 turns ON. The relay 00400 which was turned ON remains ON even if input condition 04002 turns OFF. If the current value of the clock is except at 15 hour 30 minutes, the state of relay 00400 does not change.

Note 1: If the relay specified by the F-34 (TSET) instruction is within the latch specified area, it retains its state that existed before the power failure even after power is restored. If the specified relay is outside the latch specified area, it is reset when power is restored.

Note 2: A relay assigned by F-34 (TSET) instruction is set to latch output when the JW10 stops operation (system memory #206 = 55(H)), it maintains status before stop operation. If the assigned relay is set to reset output when the JW10 stops operation system memory #206 = 00(H)), it resets at stoppage.

Note 3: Use the F-34 (TSET) instruction as a pair with the F-35 (TRST) instruction on the next page.

Note 4: Any F-34 (TSET) and F-35 (TRST) instructions between the F-30 (MCS) and F-31 (MCR) instructions do not function when the operational condition for the F-30 (MCS) instruction is OFF.

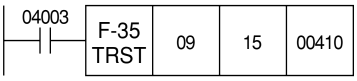
F-35
TRST

Comparison with current value of clock (specified relay reset)

(This instruction is able to program by JW-1424K/1442K/1624K/1642K and is not able to program by JW-1324K/1342K.)

Symbol	F-35 TRST | n_1 | n_2 | BIT	
Function	Compares the constants n_1 (hours) and n_2 (minutes) with the current value of the clock and resets (turns OFF) the specified BIT (relay) if they match.	
Operation	Compares the current value of the clock with n_1 and n_2, and turns OFF the relay if the comparison result shows a match.	
Range of n_1	00 to 23 (decimal)	
Range of n_2	00 to 59 (decimal)	
Range of BIT	00400 to 15777	
Condition	When the input signal is ON (not limited to OFF to ON change)	
After operation	Contents of n_1	Unchanged
	Contents of n_2	Unchanged
	Contents of BIT	When the assigned time does not match with the clock's current value, the bit condition does not change. When the assigned time matches with the clock's current value, this bit turns OFF.
	Flag	Unchanged

〔Example for use〕



Instruction	
STR	04003
F-35	
	09
	15
	00410

When input condition 04003 is ON and at 9 hours 15 minutes, relay 00410 turns OFF.
The relay which was turned OFF remains OFF even if input condition 04003 turns OFF.
If the current value of the clock is except at 9 hour 15 minutes, the state of relay 00410 does not change.

Note 1: If the relay specified by the F-35 (TRST) instruction is within the latch specified area, it retains its state that existed before the power failure even after power is restored. If the specified relay is outside the latch specified area, it is reset when power is restored.

Note 2: A relay assigned by F-35 (TRST) instruction is set to latch output when the JW10 stops operation (system memory #206 = 55(H)), it maintains status before stop operation. If the assigned relay is set to reset output when the JW10 stops operation system memory #206 = 00(H)), it resets at stoppage.

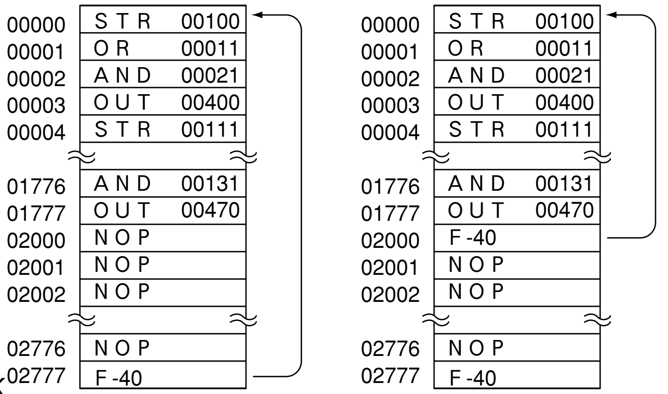
F-40
END

End instruction (END)

The F-40 instruction indicates the end of the program. There is no need of writing the END instruction, except for the following cases, because it will be automatically set in the last address of the program memory when the memory is cleared.

(1) Saving scan time

- The scan time is the I/O processing time added with the user program execution time. The user program execution time is the total time required to execute all instructions from the program address 00000 to the END instruction.
- The location of the END instruction automatically written after program is memory clear, for instance, 02777 (1535th word) in the case of basic module is JW-1324K/1342K and 07777(4096th word) in case of it is JW1424K/1442K/1624K/1642K.
- Assume now if the last address is 01777 (1024th word) when the ladder chart has written by the support tool such as JW-13PG, 02000 to 12776 are padded with NOP instruction with the END instruction in 02777, so that it requires an unwanted time in order to execute NOP instructions (JW-1324K/1342K is 1.63 μs, JW-1424K/1442K/1624K/1642K is 0.81 μs per word).
- If F-40 was written in 02000, it saves the processing time without executing those unwanted NOP instructions, and the control can proceed to a next scan cycle after termination of the user program.



00000	STR	00100
00001	OR	00011
00002	AND	00021
00003	OUT	00400
00004	STR	00111
≈		
01776	AND	00131
01777	OUT	00470
02000	NOP	
02001	NOP	
02002	NOP	
≈		
02776	NOP	
02777	F-40	

(a) END (02777) only by memory clear

00000	STR	00100
00001	OR	00011
00002	AND	00021
00003	OUT	00400
00004	STR	00111
≈		
01776	AND	00131
01777	OUT	00470
02000	F-40	
02001	NOP	
02002	NOP	
≈		
02776	NOP	
02777	F-40	

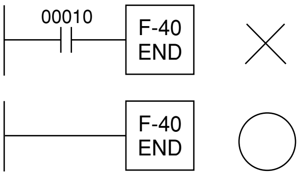
(b) F-40 (END)is written in 02000

(2) To perform a partial program execution during trial run

By inserting the F-40 instruction at the end of a sequence block, you will be able to execute only the required portion of the program. If the result was successful, the F-40 may then be deleted.

Note 1: A multiple number of F-40 may exist when the END instruction is written in (1) and (2). In this case, the userprogram terminates at the first F-40. So, it would be necessary to check the location of the END instruction before going into the actual operation.

Note 2: The F-40 (END) instruction is unconditional, and it prohibits the following kind of programming.

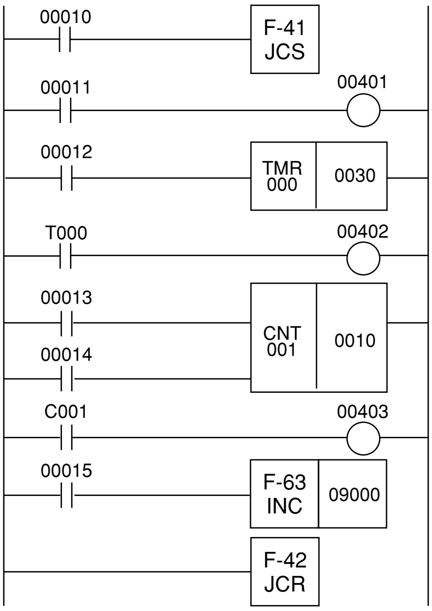


Note 3: Although the F-40 instruction has the highest priority, it will be ignored if an END instruction exists between F-141 (JMP) and F-140 (LABL) or between F-142 (CALL) and F-140 (LABL) and that F-141 or F-142 was executed.

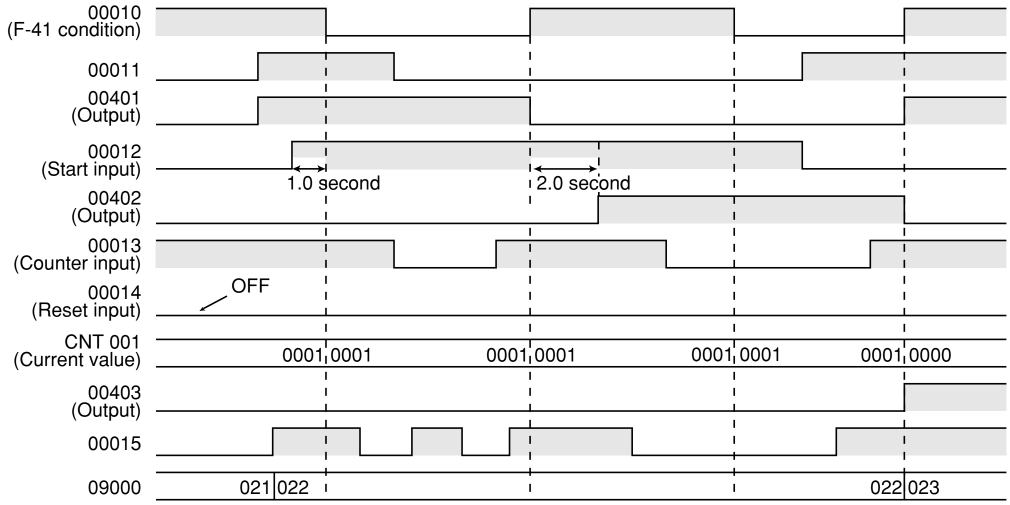
F-41 JCS — Set jump control (Jump Control Set)

F-42 JCR — Reset jump Control (Jump Control Reset)

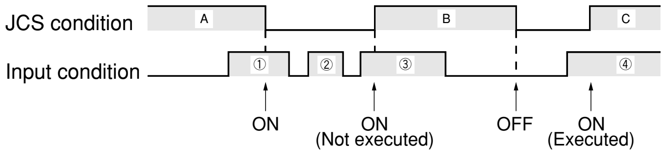
When the F-41 (JCS) condition is OFF, all instructions will not be executed, except the F-40 (END) instruction that exists before the F-42 (JCR) instruction. Therefore, it does affect the contents of the data memory, even if there was an instruction that writes the result in the data memory, such as OUT, TMR, CNT, and application instruction. And, it retains the state when JCS condition is ON.



```
   Instruction
→ S T R    00010
  F -41
  S T R    00011
  O U T    00401
  S T R    00012
  T M R      000
            0030
  S T R     T000
  O U T    00402
  S T R    00013
  S T R    00014
  C N T      001
            0010
  S T R     C001
  O U T    00403
  S T R    00015
  F -63
           09000
→ F -42
```



Note 1: Attention must be paid to the TMR internal clock (0.1 second clock), CNT counter input and application instruction input condition (which the operation takes place at an OFF to ON transition in the input condition), and F-41 (JCS) condition ON/OFF timing.

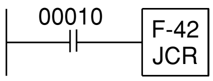


- Operation takes place at a rising of ① because the JCS condition is ON.
- Operation does not take place at a rising of ② because the JCS condition is OFF.
- Operation does not take place at a rising of ③ because the JCS condition is OFF.
- The JCS condition turns to ON while ③ is ON, but operation does not take place as it does not recognize that the input condition has changed from OFF to ON, because the input condition is ON with which theJCS condition of Ⓐchanges from ON to OFF and the input condition is ON with which the JCS condition of Ⓑchanges from OFF to ON.
- Operation does not take place at a rising of ④ because the JCS condition is OFF.
- The JCS condition becomes ON while ④ is ON. Operation takes place immediately after the JCS condition of Ⓒchanges from OFF to ON because there was no change in that the input condition is OFF with which the JCS condition of Ⓑ changes from ON to OFF and the input condition is ON with which the JCS condition of Ⓒ changes from OFF to ON.

Note 2: The END instruction will be executed regardless whether the JCS condition be ON or OFF when there was an F-40(END) instruction between F-41(JCS) and F-42(JCR). And the user program execution is terminated and the control proceeds to the next scan cycle.

Note 3: It is not possible to insert another F-41 and F-42 between F-41 (JCS) and F-42 (JCR). It will evoke "JCS ERROR" on the hand-held programmer during program check, if such a program was written.

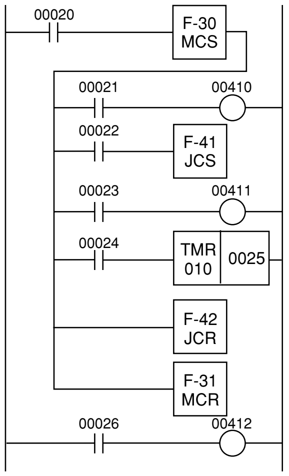
Note 4: F-42(JCR) is an unconditional instruction.



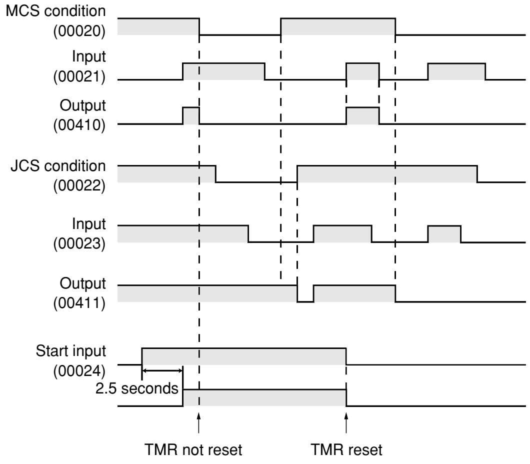
It prohibits the program like shown above.

Note 5: To use an application instruction that operates at a rising edge of F-41 (JCS) and F-42 (JCR), the input condition must be different from F-41 (JCS). No operation will take place if the condition is same.

Note 6: F-41 (JCS) and F-42 can be nested between F-30 (MCS) and F-31 (MCR). However, between JCS instruction and JCR instruction are not executed regardless whether the JCS condition be ON or OFF when the MCS condition is OFF.



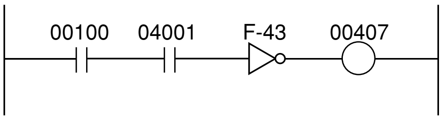
Instruction	
S T R	00020
F -30	
S T R	00021
O U T	00410
S T R	00022
F -41	
S T R	00023
O U T	00411
S T R	00024
T M R	010
	0025
F -42	
F -31	
S T R	00026
O U T	00412



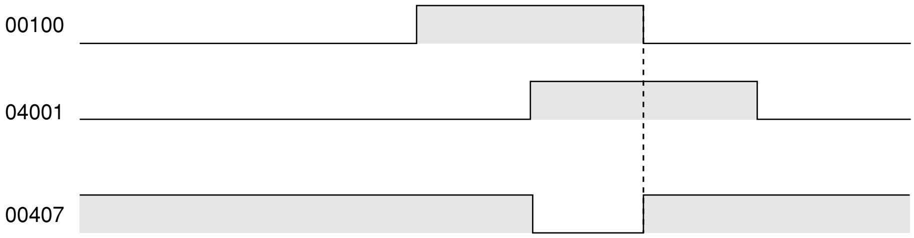
F-43
CPL

Complement bit (ComPLement)

The F-43 instruction complements the bit in the accumulator.

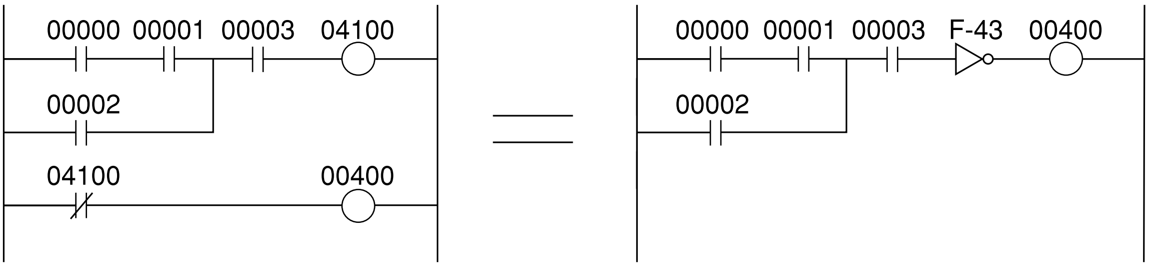


instruction	
STR	00100
AND	04001
F-43	
OUT	00407



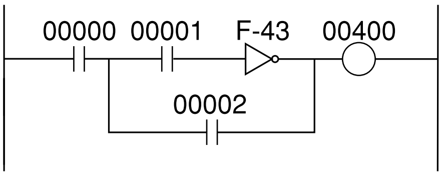
Results from the STR instruction to the F-43 instruction are complemented and sent to the output relay 00407.

Use of F-43 permits to obtain the complemented output without the use of auxiliary contact.

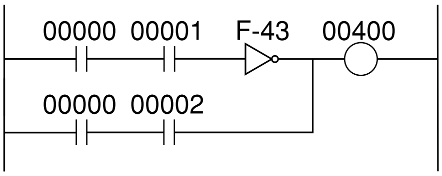


Note 1: The F-43 instruction may be used for a single or a multiple number of contacts.

Note 2: Pay attention that the following programs (a) and (b) will not produce the same result because the F-43 instruction is the instruction that complement the contents of the accumulator.



(a)

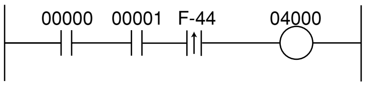


(b)

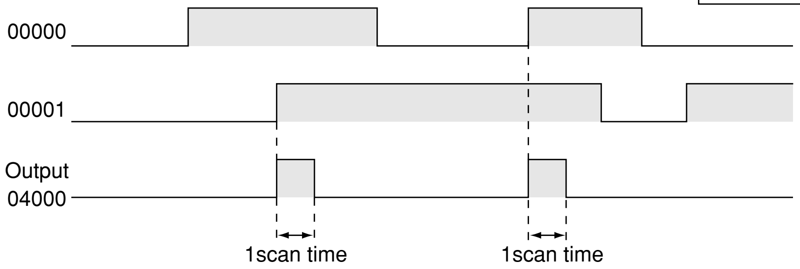
F-44
⊣↑⊢

Differentiate at ON

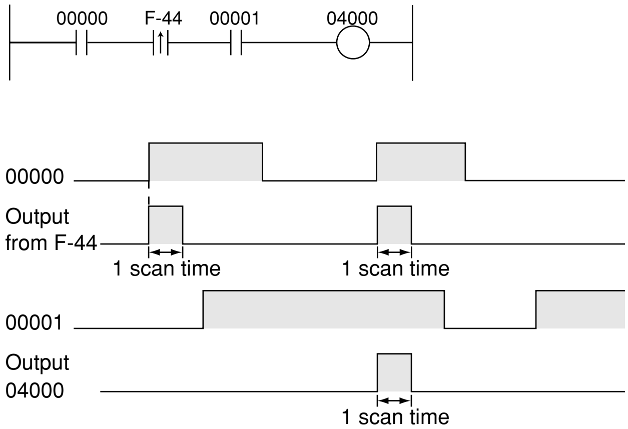
The F-44 instruction sends 1 scan time pulse when the immediate state of the accumulator changes from OFF to ON.



Instruction	
STR	00000
AND	00001
F-44	
OUT	04000

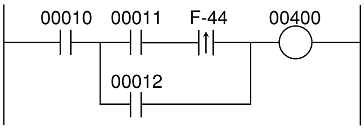


Note 1:Note that a different result is produced when the sequence of the F-44 program is changed in the above ladder chart. (Identical in case of F-45.)

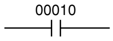
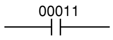
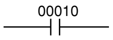
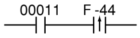
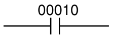
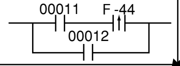
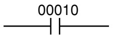
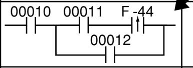
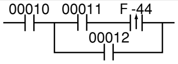


Note 2: The F-44 instruction scans only once even if inserted between an F-47 instruction (level operation condition set) and an F-48 instruction (level operation condition reset). (See F-47 and F-48.)

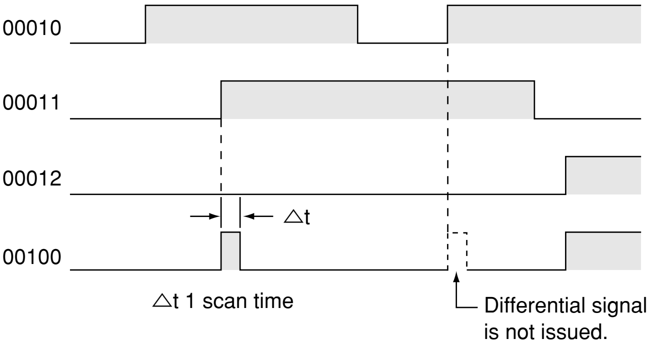
Note 3:



STR	00010
STR	00011
F-44	
OR	00012
AND	STR
OUT	00400

	Accumulator ACC	Stack register S_1
STR 00010		
STR 00011		
F-44		
OR 00012		
AND STR		
OUT 00400		

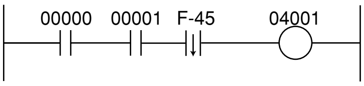
The accumulator turns ON for only a scan cycle that 00011 changed from OFF to ON.



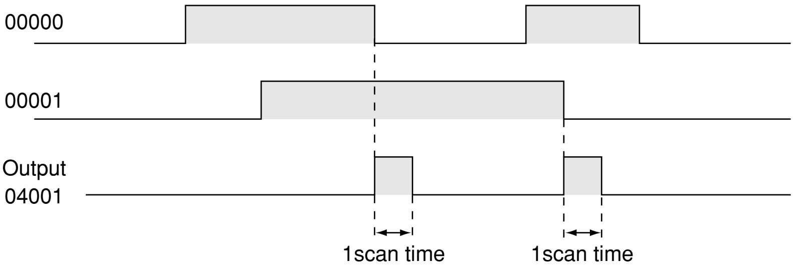
In the above example, a differential signal is not issued even if 00010 has changed from OFF to ON when 00011 is ON, because 00010 is ANDed by the AND/STR instruction.

F-45 ┤↑├ Differentiate at OFF

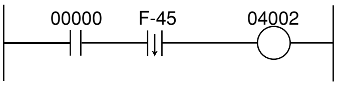
The F-45 instruction sends 1 scan time pulse when the immediate state of the accumulator changes from OFF to ON.

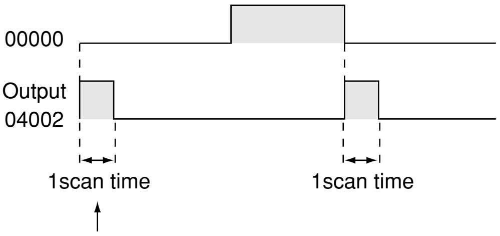


Instruction	
STR	00000
AND	00001
F-45	
OUT	04001



Note 1: Use of the F-45 instruction may sometimes generate 1 scan time pulse at the operation immediately following program writing, that is, when the address of F-45 instruction is changed because of program insertion or deletion.



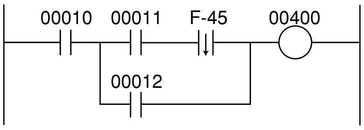


Output (04002) turns ON if input (00000) is OFF at the start of operation immediately after write of program.

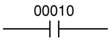
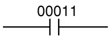
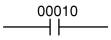
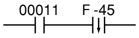
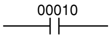
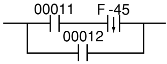
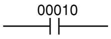
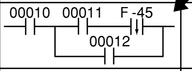
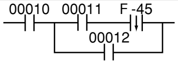
Note 2: The F-45 instruction scans only once even if inserted between an F-47 instruction (level operation condition set) and an F-48 instruction (level operation condition reset). (See page F-47 and F-48.)

Note 3: This instruction does not perform an operation during the 1 scan time when the power is turned ON or immediately after the start of the operation mode.

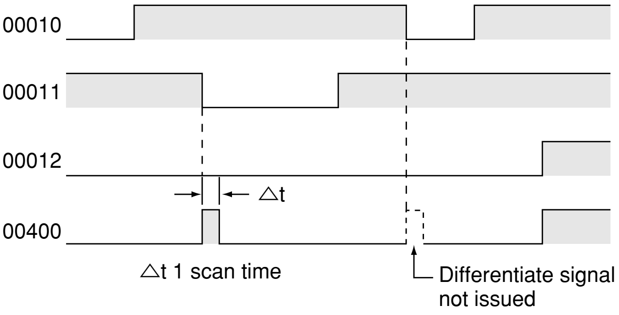
(Note 4)



STR	00010
STR	00011
F-45	
OR	00012
AND	STR
OUT	00400

	Accumulator ACC	Stack register S_1
STR 00010		
STR 00011		
F-45		
OR 00012		
AND STR		
OUT 00100		

The accumulator turns ON for only a scan cycle that 00011 changed from OFF to ON.

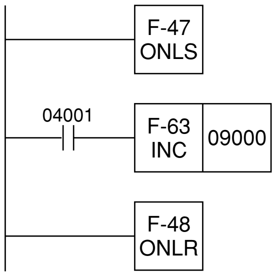


In the above example, a differential signal is not issued even if 00010 has changed from ON to OFF when 00011 is ON, because 00010 is ANDed by the AND/STR instruction.

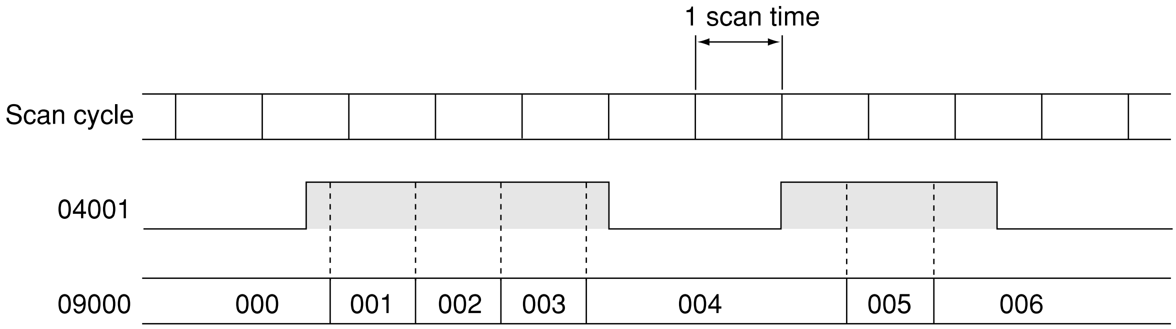
F-47 ONLS — ON level set (ON Level Set)

F-48 ONLR — ON level reset (ON Level Reset)

Set to execute operation conditions of the instruction, which shall be executed at rising edge (from OFF to ON) of the input signal, when the input signal is ON.



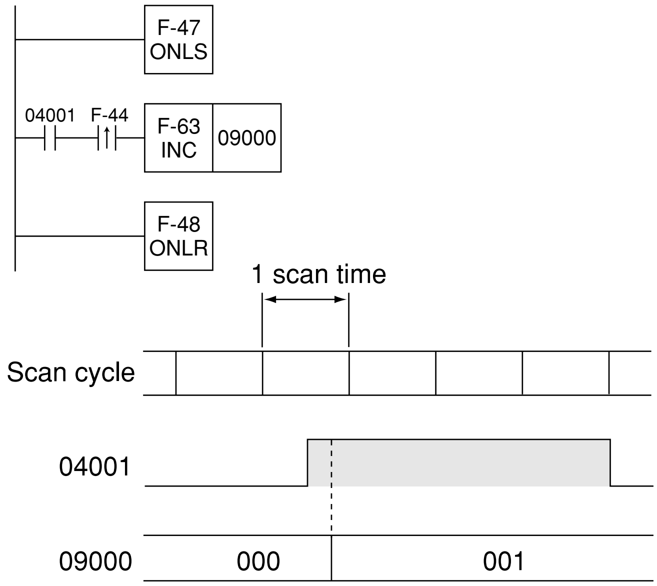
```
Instruction
F -47
S T R          04001
F-63
               09000
F -48
```



Note 1: Another F-47 may not be nested between already existing F-47 (ONLS) and F-48 (ONLR).

Note 2: Only 1 scan will take place at a rising edge of the operation condition, if it contains a differentiate instruction

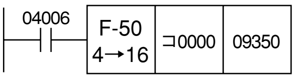
(F-44, F-45). (Example for F-44)



F-50 4→16 Decode 4 to 16

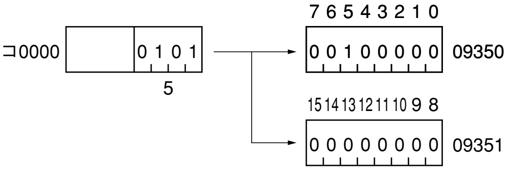
Symbol	F-50 4→16 S D	
Function	The lower 4 bits data in the register S are decoded and stored in the 2 bytes area of the registers D and D+1.	
Operation	S→D, D+1	
Range of S	⊐0000 to ⊐1577 b0000 to b0777 09000 to 09777 19000 to 19777 29000 to 29777 39000 to 39777	
Range of D	⊐0000 to ⊐1576 b0000 to b0776 09000 to 09776 19000 to 19776 29000 to 29776 39000 to 39776	
Condition	Rising edge of input signal (OFF to ON)	
After operation	Contents of S	Unchanged
	Contents of D	Result (0 to 7)
	Contents of D+1	Result (8 to 15)
	Flag	Unchanged

〔Example for use〕



Instruction	
S T R	04006
F -50	
	⊐0000
	09350

When the input condition 04006 changes from OFF to ON, the lower 4 bits data of ⊐0000 are decoded and stored as 16 bits data in the 2 bytes area consisting of registers 09350 and 09351.



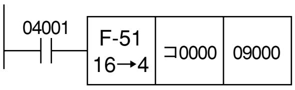
Only the bit position that corresponds to the low order 4 bits representing 0 to 15 is turned ON, and other bits are turned OFF.

Note 1: Upper 4 bits of S are disregarded.

F-51 16→4 Encode 16 to 4

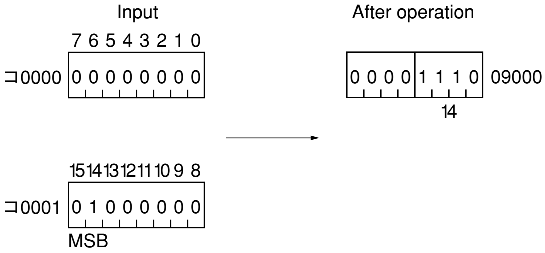
Symbol	F-51 16→4 S D	
Function	The 2 bytes data in registers S and S+1 are encoded and stored in the register D.	
Operation	S, S+1→D	
Range of S	⊐0000 to ⊐1576 b0000 to b0776 09000 to 09776 19000 to 19776 29000 to 29776 39000 to 39776	
Range of D	⊐0000 to ⊐1577 b0000 to b0777 09000 to 09777 19000 to 19777 29000 to 29777 39000 to 39777	
Condition	Rising edge of input signal (OFF to ON)	
After operation	Contents of S, S+1	Unchanged
	Contents of D	Result
	Flag	Unchanged

〔Example for use〕



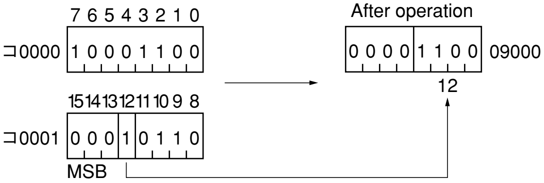
Instruction	
S T R	04001
F -51	
	⊐0000
	09000

When the input condition 04001 changes from OFF to ON, the 2 bytes data consisting of registers ⊐0000 and ⊐0001 are encoded and stored in the register 09000.



Note 1: After the operation, the upper 4 bits of D (09000 in the example) become 0 at all times.

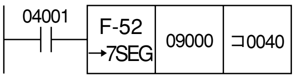
Note 2: The MSB of the encoder input takes the highest priority.



F-52 →7SEG Decode to 7 segments data

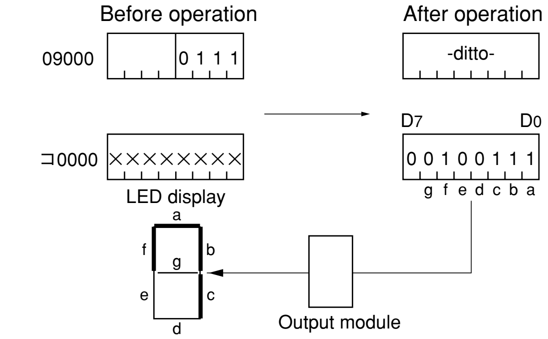
Symbol	F-52 →7SEG | S | D	
Function	The lower 4 bits data in register S is decoded into 7 segments display data.	
Operation	S→D	
Range of S	コ0000 to コ1577 b0000 to b0777 09000 to 09777 19000 to 19777 29000 to 29777 39000 to 39777	
Range of D	コ0000 to コ1577 b0000 to b0777 09000 to 09777 19000 to 19777 29000 to 29777 39000 to 39777	
Condition	Rising edge of input signal (OFF to ON)	
After operation	Contents of S	Unchanged
	Contents of D	Result (Refer to "7 segments decoder chart")
	Flag	Unchanged

〔Example for use〕



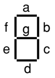
Instruction	
S T R	04001
F-52	
	09000
	コ0040

When the input condition 04001 changes from OFF to ON, the lower 4 bits data of the register 09000 are decoded into the 7 segments display data. See "7 segments decoder chart" relation between input data and display output.



Output data D_0 to D_6 correspond to a to g of the 7 segments display, respectively. The output of D_7 is always 0.

7 segments decoder chart

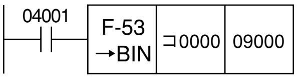


Input data	Output data	Display output
	g f e d c b a	
0 0 0 0 0 0 0 0	0 0 1 1 1 1 1 1	0
0 0 0 0 0 0 0 1	0 0 0 0 0 1 1 0	1
0 0 0 0 0 0 1 0	0 1 0 1 1 0 1 1	2
0 0 0 0 0 0 1 1	0 1 0 0 1 1 1 1	3
0 0 0 0 0 1 0 0	0 1 1 0 0 1 1 0	4
0 0 0 0 0 1 0 1	0 1 1 0 1 1 0 1	5
0 0 0 0 0 1 1 0	0 1 1 1 1 1 0 1	6
0 0 0 0 0 1 1 1	0 0 1 0 0 1 1 1	7
0 0 0 0 1 0 0 0	0 1 1 1 1 1 1 1	8
0 0 0 0 1 0 0 1	0 1 1 0 1 1 1 1	9
0 0 0 0 1 0 1 0	0 1 1 1 0 1 1 1	A
0 0 0 0 1 0 1 1	0 1 1 1 1 1 0 0	b
0 0 0 0 1 1 0 0	0 0 1 1 1 0 0 1	C
0 0 0 0 1 1 0 1	0 1 0 1 1 1 1 0	d
0 0 0 0 1 1 1 0	0 1 1 1 1 0 0 1	E
0 0 0 0 1 1 1 1	0 1 1 1 0 0 0 1	F

F-53 →BIN Convert 4 digits BCD to 16 bits binary

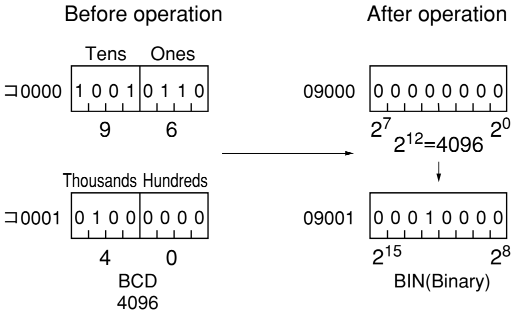
<table>
<tr><td colspan="2">Symbol</td><td colspan="5">— F-53 →BIN | S | D</td></tr>
<tr><td colspan="2">Function</td><td colspan="5">The 2 bytes BCD contents (4 digits data) of the registers S, S+1 are converted into binary, and the result is stored in the 2 bytes area of the registers D, D+1.</td></tr>
<tr><td colspan="2">Operation</td><td colspan="5">S, S+1→D, D+1</td></tr>
<tr><td colspan="2">Range of S</td><td colspan="5">⊐0000 to ⊐1576
b0000 to b0776
09000 to 09776
19000 to 19776
29000 to 29776
39000 to 39776</td></tr>
<tr><td colspan="2">Range of D</td><td colspan="5">⊐0000 to ⊐1576
b0000 to b0776
09000 to 09776
19000 to 19776
29000 to 29776
39000 to 39776</td></tr>
<tr><td colspan="2">Condition</td><td colspan="5">Rising edge of input signal (OFF to ON)</td></tr>
<tr><td rowspan="6">After operation</td><td>Contents of S, S+1</td><td colspan="5">Unchanged</td></tr>
<tr><td>Contents of D</td><td colspan="2">Result (0 to 255)</td><td colspan="3" rowspan="2">Unchanged when the contents of the registers S and S+1 are not BCD code.</td></tr>
<tr><td>Contents of D+1</td><td colspan="2">Result (256 to 9999)</td></tr>
<tr><td rowspan="3">Flag</td><td>Contents of registers S,S+1</td><td>Zero 07357</td><td>Carry 07356</td><td>Error 07355</td><td>Non-carry 07354</td></tr>
<tr><td>BCD code</td><td rowspan="2">0</td><td rowspan="2">0</td><td>0</td><td rowspan="2">0</td></tr>
<tr><td>Not BCD code</td><td>1</td></tr>
</table>

〔Example for use〕



Instruction	
STR	04001
F-53	
	⊐0000
	09000

When the input condition 04001 changes from OFF to ON, the contents of the 4 digits BCD data in registers ⊐0000 and ⊐0001 are converted into binary, and its result is transferred and stored in 2 bytes area of registers 09000 and 09001.



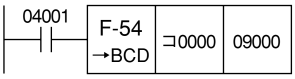
Note 1: If the F-53 instruction is used for programming, the F-03w instruction displays the program during monitoring.

Similarity instructions: F-03, F-03w

F-54 →BCD Convert 16 bits binary to 6 digits BCD

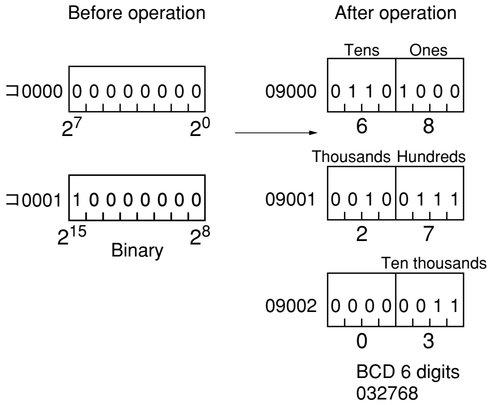
Symbol	F-54 →BCD | S | D	
Function	The contents of 2 bytes binary in the registers S, S+1 converted into 6 digits BCD, and the result is stored in 3 bytes of the registers D, D+1, and D+2.	
Operation	S, S+1→D, D+1, D+2	
Range of S	⊐0000 to ⊐1576 b0000 to b0776 09000 to 09776 19000 to 19776 29000 to 29776 39000 to 39776	
Range of D	⊐0000 to ⊐1575 b0000 to b0775 09000 to 09775 19000 to 19775 29000 to 29775 39000 to 39775	
Condition	Rising edge of input signal (OFF to ON)	
After operation	Contents of S, S+1	Unchanged
	Contents of D	Result (ones and tens)
	Contents of D+1	Result (hundreds and thousands)
	Contents of D+2	Result (ten thousands)
	Flag	Unchanged

〔Example for use〕



Instruction	
S T R	04001
F -54	
	⊐0000
	09000

When the input condition 04001 changes from OFF to ON, the contents of 2 bytes binary in registers ⊐0000 and ⊐0001 are converted into 6 digits BCD and stored in 3 bytes area of registers that begin from 09000.

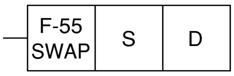


Note 1: If the F-54 instruction is used for programming, the F-04w instruction displays the program during monitoring.

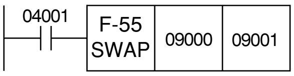
Similariry instructions: F-04, F-04w

F-55 SWAP

Swap upper 4 bits with lower 4 bits

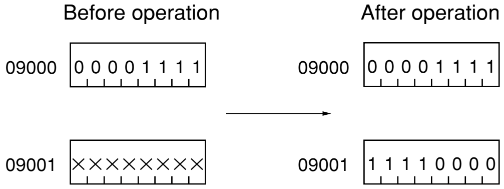
Symbol		
Function	Upper 4 bits are swapped with lower 4 bits of the register S and stored in the register D.	
Operation	S→D	
Range of S	コ0000 to コ1577 b0000 to b0777 09000 to 09777 19000 to 19777 29000 to 29777 39000 to 39777	
Range of D	コ0000 to コ1577 b0000 to b0777 09000 to 09777 19000 to 19777 29000 to 29777 39000 to 39777	
Condition	Rising edge of input signal (OFF to ON)	
After operation	Contents of S	Unchanged
	Contents of D	Result
	Flag	Unchanged

〔Example for use〕



Instruction	
S T R	04001
F -55	
	09000
	09001

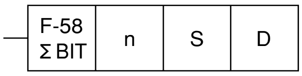
When the input condition 04001 changes from OFF to ON, upper 4 bits are swapped with lower 4 bits of the register 09000 and its result is stored in the register 09001. The contents of the register 09000, however, remains unchanged.



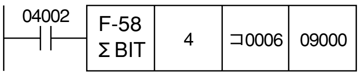
Reference F-55 becomes valid in the following case.
With the F-52 (7 SEG decode) instruction, the lower 4 bits are decoded into the 7 segments display data. To display a multiple number of digits on the display, the upper 4 bits need to be swapped with the lower 4 bits, then F-52 should be executed.

F-58 Σ BIT

Total ON bits

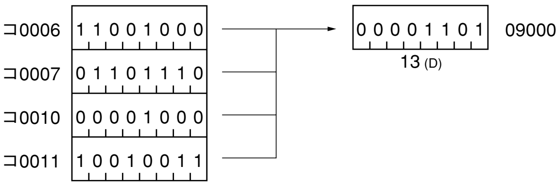
Symbol		
Function	All active bits in the "n" byte register having the register S at its top are stored in the register D.	
Operation	ON bits→D	
Range of "n"	0 to 7 (8 bytes for 0)	
Range of S	コ0000 to コ1577 b0000 to b0777 09000 to 09777 19000 to 19777 29000 to 29777 39000 to 39777	
Range of D	コ0000 to コ1577 b0000 to b0777 09000 to 09777 19000 to 19777 29000 to 29777 39000 to 39777	
Condition	Rising edge of input signal (OFF to ON)	
After operation	Contents of S, S+1,··· S+n-1	Unchanged
	Contents of D	Result
	Flag	Unchanged

〔Example for use〕



Instruction	
S T R	04002
F -58	
	4
	コ0006
	09000

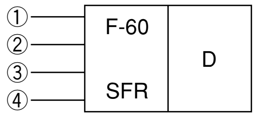
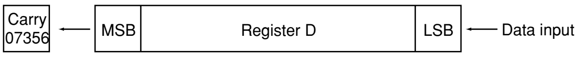
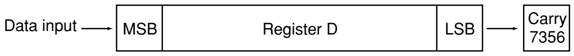
When the input condition 04002 changed from OFF to ON, the total number of active bits in a 4 bytes register headed by the register コ0006 is stored in the register 09000.



Among 32 bits in コ0006 to コ0011, 13 bits are ON.

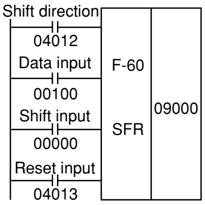
F-60
SFR

Shift register bidirectional (1 byte) (forward/backward ShiFt Register)

Symbol	 ① Shift direction input ② Data input ③ Shift input ④ Reset input
Function	The 8 bits data in the register S is shifted to upper or lower bit positions according to the direction input of ①.
Operation	• When the shift direction input is ON :  • When the shift direction input is OFF : 
Range of Register D	Ǝ0000 to Ǝ1577 b0000 to b0777 09000 to 09777 19000 to 19777 29000 to 29777 39000 to 39777
Condition	When the reset input ④ is OFF, bits are shifted at a rising edge (OFF to ON) of the shift input ③.
After operation	Contents of D: • Result is produced when the reset input ④ is OFF. • All bits are turned OFF when the reset input ④ is ON.

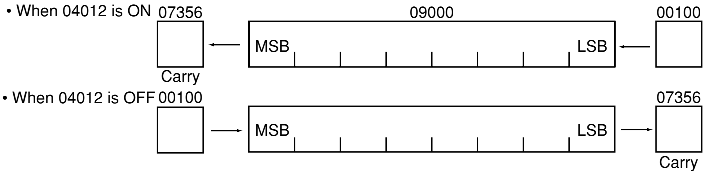
	Reset input ④	Zero 07357	Carry 07356	Error 07355	Non-carry 07354
Flag	OFF	0 or 1	0 or 1	0	1 or 0
	ON	0	0		0

〔Example for use〕



Instruction	
STR	04012
STR	00100
STR	00000
STR	04013
F-60	09000

When the shift input 00000 changes from OFF to ON, shift takes place in the following way according to the state of the shift direction input 04012.



Input condition		09000 (before operation)								09000 (after operation)								Zero 07357	Carry 07356	Non-carry 07354
		7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0			
04012	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	●	○	●
00100	○	●	○	○	○	●	○	●	○	○	●	○	○	○	●	○	●	○	○	●
00000	ʃ	○	●	○	○	●	○	○	●	○	○	●	○	○	●	○	○	○	●	○
04013	○	○	○	○	○	○	○	○	●	○	○	○	○	○	○	○	○	●	●	○
04012	●	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	●	○	●
00100	○	○	●	○	●	○	○	○	○	●	○	●	○	○	○	○	○	○	○	●
00000	ʃ	●	○	○	○	●	○	○	○	○	○	○	●	○	○	○	○	○	●	○
04013	○	●	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	●	●	○
04012 00100	○ ●	●	○	○	○	○	●	○	○	●	●	○	○	○	○	●	○	○	○	●
00000 04013	ʃ ○	○	○	○	●	○	○	○	●	●	○	○	○	●	○	○	○	○	●	○
04012 00100	● ●	○	●	○	○	○	○	○	○	●	○	○	○	○	○	○	●	○	○	●
00000 04013	ʃ ○	●	○	○	○	●	○	○	○	○	○	○	●	○	○	○	●	○	●	○
4013	●	○	○	●	○	●	○	○	○	○	○	○	○	○	○	○	○	○	○	○

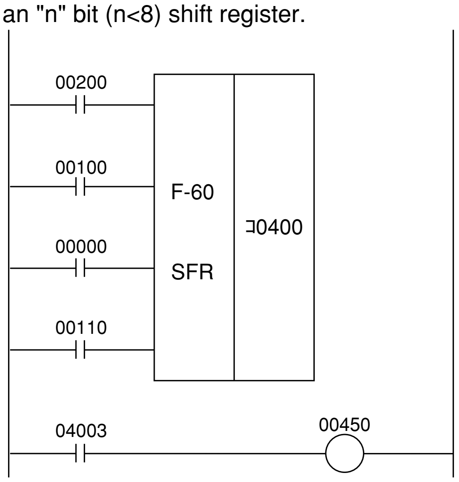
·Error flag (07355) is OFF at all times. ○ OFF ● ON

Note 1: By setting the reset condition in the system memory for reset input ④ (#202), it permits to "reset with OFF."

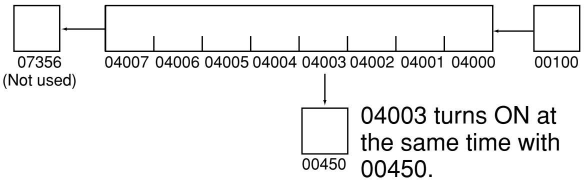
Similarity instructions: F-60w

Reference

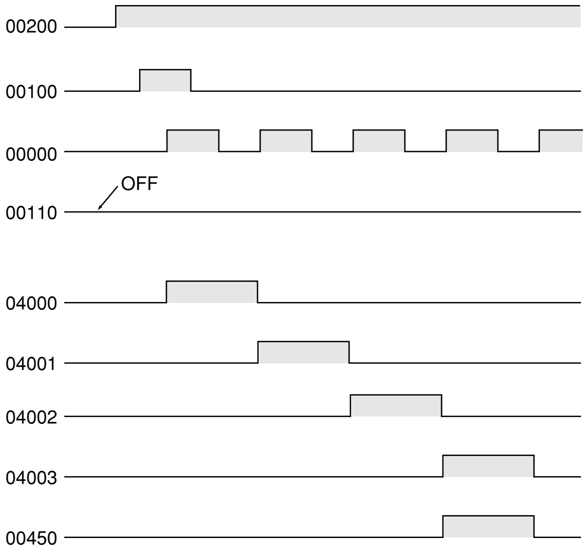
When ⊐×××× is assigned for D, it allows to constitute an "n" bit (n<8) shift register.



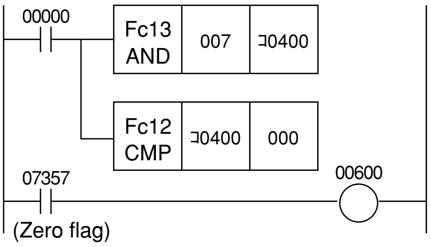
(When 00200 is ON)



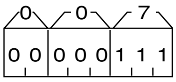
04003 turns ON at the same time with 00450.



- Data are shifted in 04004 to 04007.
- When all of 04000 to 04007 are 0, the zero flag turns to 1. The following program must be used to check that 04000 to 04002 are 0.



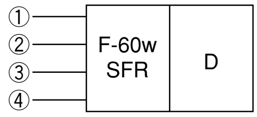
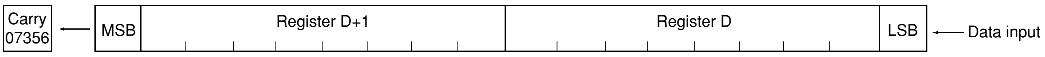
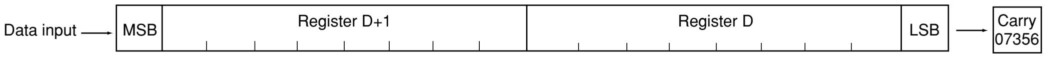
(Zero flag)



By ANDing this way, 04003 to 04007 are masked (all 0).

F-60w
SFR

Shift register bidirectional (1 word) (forward/backward ShiFt Register)

Symbol	 ① Shift direction input ② Data input ③ Shift input ④ Reset input
Function	The 16 bits contents of the registers D, D+1 are shifted to upper or lower bit positions according to the direction input of ①.
Operation	• When the shift direction input ① is ON:  Carry 07356 MSB Register D+1 Register D LSB Data input • When the shift direction input ① is OFF:  Data input MSB Register D+1 Register D LSB Carry 07356
Range of D	コ0000 to コ1576 b0000 to b0776 09000 to 09776 19000 to 19776 29000 to 29776 39000 to 39776
Condition	When the reset input ④ is OFF, bits are shifted at a rising edge (OFF to ON) of the shift input ③.

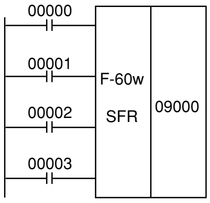
After operation						
	Contents of D, D+1	• Result is produced when the reset input ④ is OFF. • All bits are turned OFF when the reset input ④ is ON.				
	Flag	Reset input ④	Zero 07357	Carry 07356	Error 07355	Non-carry 07354
		OFF	0 or 1	0 or 1	0	1 or 0
		ON	0	0		0

Note 1: Be sure to use even address for register D.

Note 2: By setting the reset input ④ in the system memory (#202), it permits to "reset with OFF."

Similarity instructions: F-60

〔Example for use〕



Instruction	
S T R	00000
S T R	00001
S T R	00002
S T R	00003
F -60w	
	09000

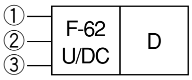
00000 ① ON ------------------ Shifts towards MSB
00001 ② ON ------------------ Data input ON
00002 ③ OFF→ON ----------- Shift direction
00003 ④ OFF ----------------- No reset function

The following result will be produced for the above input condition.

	09001	09000
Before operation	1 0 0 1 0 1 1 0	1 0 1 0 0 1 0 0
After operation	0 0 1 0 1 1 0 1	0 1 0 0 1 0 0 1

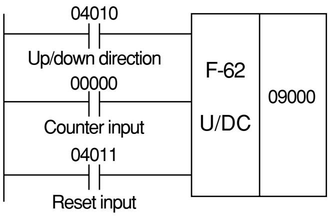
Zero 07357	Carry 07356	Error 07355	Non-carry 07354
0	1	0	0

F-62 U/DC — 2 digits BCD up/down counter (Up/Down Counter)

Symbol	 ① Up/down direction input ② Counter input ③ Reset input
Function	The contents of the register D (BCD 2 digits) are added (① ON) or subtracted (① OFF), according to the up/down counter direction input.
Operation	When the up/down direction input ① is ON: 〈D〉+1→D When the up/down direction input ① is OFF: 〈D〉−1→D
Range of D	コ0000 to コ1577 b0000 to b0777 09000 to 09777 19000 to 19777 29000 to 29777 39000 to 39777
Condition	At a rising of the counter input ② (OFF to ON) when the reset input ③ is OFF

After operation	Contents of D	• Result (BCD code) is contained when the reset input ③ is OFF. • All bits are OFF when the reset input ③ is ON.					
	Flag	Up/down direction input ①	Result	Zero 07357	Carry 07356	Error 07355	Non-carry 07354
		ON	99+1 →00	1	1	0	0
			00 to 98+1 →01 to 99	0	0	0	1
			Not BCD code	0	0	1	0
		OFF	00−1 →99	0	1	0	0
			01−1 →00	1	0	0	1
			02 to 99−1 →01 to 98	0	0	0	1
			Not BCD code	0	0	1	0
		Reset input ③ ON		0	0	0	0

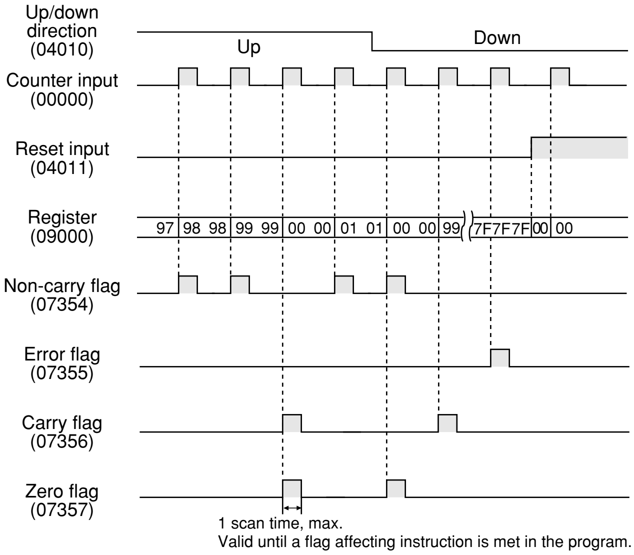
〔Example for use〕



Instruction	
S T R	04010
S T R	00000
S T R	04011
F -62	
	09000

When the reset input 04011 is OFF, the counter is enabled, if set to ON reset mode.

When the up/down counter direction input 04010 is ON, it acts as an incremental counter. When it is OFF, it acts as a decremental counter. If the contents of 09000 were other than the BCD code, the error flag (07355) is turned ON and no counter operation takes place. (7F in the example)



Note 1: By setting the reset input③ in the system memory (#202), it permits to "reset with OFF."

Similarity instructions: F-62w

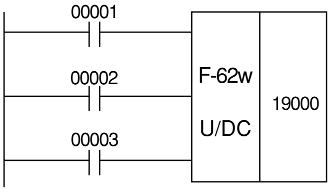
F-62w U/DC

4 digits BCD up/down counter (Up/Down Counter)

Symbol	① ② ③ F-62w U/DC | D ① Up/down direction input ② Counter input ③ Reset input
Function	The 4 digits BCD contents of the register D, D+1 are added (① ON) or subtracted (① OFF), according to the up/down counter direction input ①.
Operation	When the up/down direction input ① is ON : 〈D, D+1〉+1→D, D+1 When the up/down direction input ① is OFF : 〈D, D+1〉−1→D, D+1
Range of D	コ0000 to コ1577 b0000 to b0777 09000 to 09777 19000 to 19777 29000 to 29777 39000 to 39777
Condition	At a rising of the counter input ② (OFF to ON) when the reset input ③ is OFF.

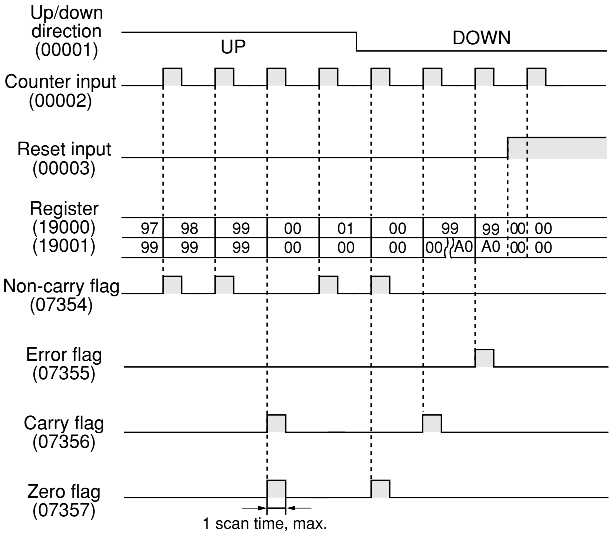
After operation	Contents of D	Lower 2 digits of result		All bits are OFF when the reset input ③ is OFF.				
	Contents of D+1	Upper 2 digits of result						
	Flag	Up/down direction input ①	Result	Zero 07357	Carry 07356	Error 07355	Non-carry 07354	
		ON	9999+1	1	1	0	0	
			0000 to 9998+1	0	0	0	1	
			Not BCD code	0	0	1	0	
		OFF	0000−1	0	1	0	0	
			0001−1	1	0	0	1	
			0002 to 9999−1	0	0	0	1	
			Not BCD code	0	0	1	0	
		Reset input ③ ON		0	0	0	0	

〔Example for use〕



Instruction	
S T R	00001
S T R	00002
S T R	00003
F -62w	19000

When the reset input 00003 is OFF, the counter is enabled, if set to ON reset mode.
When the up/down counter direction input 00001 is ON, it acts as an incremental counter. When it is OFF, it acts as a decremental counter. If the contents of the register 19000 or 19001 were other than the BCD code, the error flag (07355) is turned ON and no counter operation takes place.

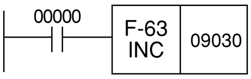


Note 1: Be sure to use even address for register D.
Note 2: By setting the reset input③ in the system memory (#202), it permits to "reset with OFF."
Similarity instructions: F-62

F-63 INC — Increment counter (1 byte) (INCrement)

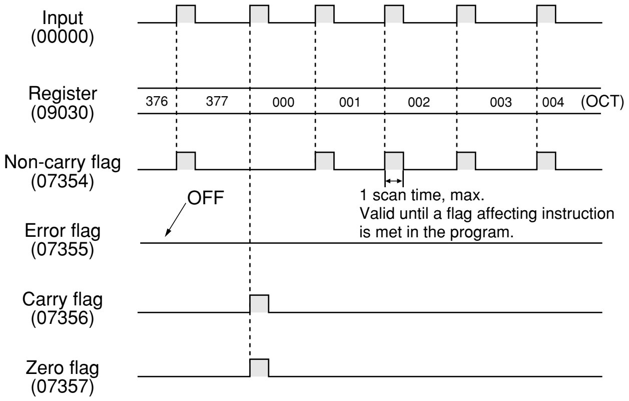
Symbol	F-63 INC | D					
Function	The contents of the register D (binary data) are incremented.					
Operation	〈D〉+1→D					
Range of D	⊐0000 to ⊐1577 b0000 to b0777 09000 to 09777 19000 to 19777 29000 to 29777 39000 to 39777					
Condition	Rising edge of input signal (OFF to ON)					
After operation	Contents of D	Result (binary code)				
	Flag	Result (oct)	Zero 07357	Carry 07356	Error 07355	Non-carry 07354
		377→000	1	1	0	0
		Other than above	0	0	0	1

〔Example for use〕



Instruction	
S T R	00000
F -63	
	09030

When the input condition 00000 changes from OFF to ON, the data are incremented.



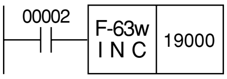
Note 1: The contents of D are represented by a binary number, which can assume a number of 000 to 255 in the decimal notation or 000 to 377(8) in the octal notation.

Similarity instructions: F-63w

F-63w INC — Increment counter (1 word) (INCrement)

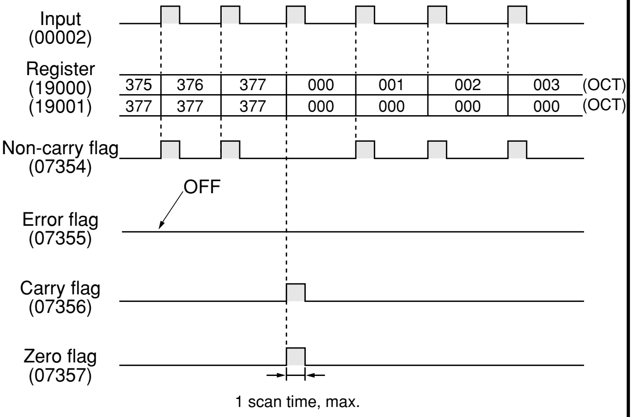
Symbol	F-63w INC | D					
Function	The binary contents of the registers D and D+1 are incremented.					
Operation	〈D, D+1〉+1→D, D+1					
Range of D	⊐0000 to ⊐1576 b0000 to b0776 09000 to 09776 19000 to 19776 29000 to 29776 39000 to 39776					
Condition	Rising edge of input signal (OFF to ON)					
After operation	Contents of D	Lower digits of result				
	Contents of D+1	Upper digits of result				
	Flag	Result (oct)	Zero 07357	Carry 07356	Error 07355	Non-carry 07354
		177777 →000000	1	1	0	0
		Other than above	0	0	0	1

〔Example for use〕



Instruction	
S T R	00002
F -63w	
	19000

When the input condition 00002 changes from OFF to ON, the data are incremented.

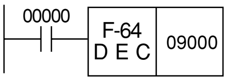


Note 1: Be sure to use even address for register D.
Similarity instructions: F-63

F-64 DEC Decrement counter (1 byte) (DECrement)

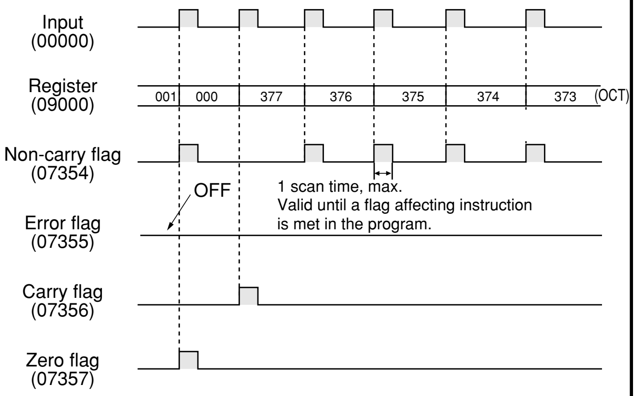
Symbol	F-64 DEC │ D					
Function	The contents of the register D (binary data) are incremented.					
Operation	〈D〉−1→D					
Range of D	⊐0000 to ⊐1577 b0000 to b0777 09000 to 09777 19000 to 19777 29000 to 29777 39000 to 39777					
Condition	Rising edge of input signal (OFF to ON)					
After operation	Contents of D	Result (binary code)				
	Flag	Result (oct)	Zero 07357	Carry 07356	Error 07355	Non-carry 07354
		001 → 000	1	0	0	1
		000 → 377	0	1	0	0
		Other than above	0	0	0	1

〔Example for use〕



Instruction	
S T R	00000
F -64	
	09000

When the input condition 00000 changes from OFF to ON, the data are decremented.



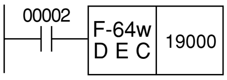
Note 1: The contents of D are represented by a binary number, which can assume a number of 000 to 255 in the decimal notation or 000 to 377 (8) in the octal notation.

Similarity instructions: F-64w

F-64w DEC Decrement counter (1 word) (DECrement)

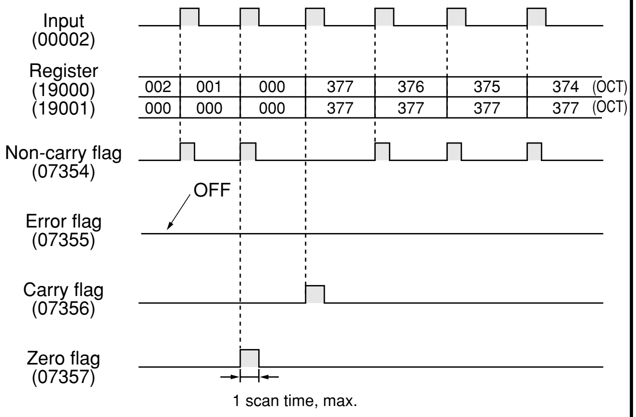
Symbol	F-64w DEC │ D					
Function	The binary contents of the registers D, D+1 are decremented.					
Operation	〈D, D+1〉−1→D, D+1					
Range of D	⊐0000 to ⊐1576 b0000 to b0776 09000 to 09776 19000 to 19776 29000 to 29776 39000 to 39776					
Condition	Rising edge of input signal (OFF to ON)					
After operation	Contents of D	Lower digits of result				
	Contents of D+1	Upper digits of result				
	Flag	Result (octal)	Zero 07357	Carry 07356	Error 07355	Non-carry 07354
		000001 →000000	1	0	0	1
		000000 →177777	0	1	0	0
		Other than above	0	0	0	1

〔Example for use〕



Instruction	
S T R	00002
F -64w	
	19000

When the input condition 00002 changes from OFF to ON, the data are decremented.



Note 1: Be sure to use even address for register D.

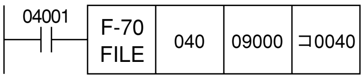
Similarity instructions: F-64w

F-70
FILE

Transfer "n" bytes block (FILE)

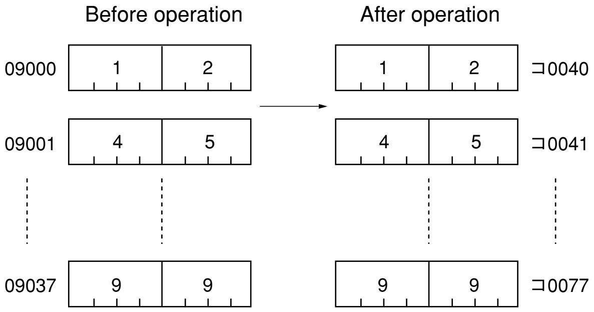
Symbol	F-70 FILE | n | S | D	
Function	"n" bytes data through the register S to the register S+n−1 are transferred in batch to the "n" bytes area beginning from the register D to D+n−1.	
Operation	S, …S+n−1→D, …D+n−1	
Range of "n"	000～377(8) (256 bytes for 000)	
Range of S	⊐0000 to ⊐1577 b0000 to b0777 09000 to 09777 19000 to 19777 29000 to 29777 39000 to 39777	@⊐0000 to @⊐1574 @b0000 to @b0774 @09000 to @09774 @19000 to @19774 @29000 to @29774 @39000 to @39774
Range of D	⊐0000 to ⊐1577 b0000 to b0777 09000 to 09777 19000 to 19777 29000 to 29777 39000 to 39777	@⊐0000 to @⊐1574 @b0000 to @b0774 @09000 to @09774 @19000 to @19774 @29000 to @29774 @39000 to @39774
Condition	Rising edge of input signal (OFF to ON)	
After operation	Contents of S, S+n−1	Unchanged
	Contents of D Contents of D+1 ⋮ Contents of D+n-1	Contents of the register S Contents of the register S+1 ⋮ Contents of the register S+n-1
	Flag	Unchanged

〔Example for use〕



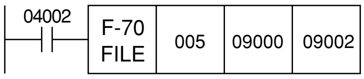
Instruction	
S T R	04001
F -70	
	040
	09000
	⊐0040

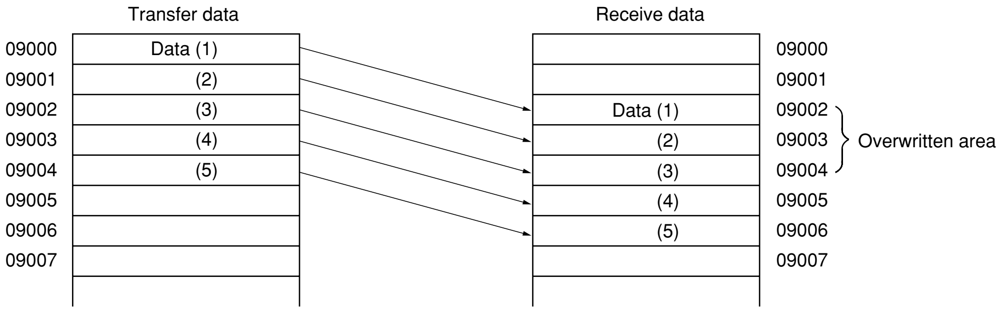
When the input condition 04001 changes from OFF to ON, 040 (8) bytes data (32 bytes in decimal) in register 09000 to 09037 are transferred in batch to the 32 bytes area of ⊐0040 to ⊐0077.
The contents of registers, 09000 to 09037, remain unaffected.



Note 1: F-70 can use indirect address for assigning S and D such as @⊐0000 or @⊐09000.
For indirect address, see page 9・21 "Indirect address assignment."

Note 2: It would be possible to program "n", S, and D that the source may overwrite the destination.





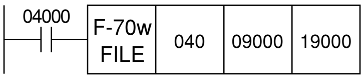
Similarity instructions: F-00, F-00w, F-70w, F-74, F-74w

F-70w FILE

Transfer "n" words block (FILE)

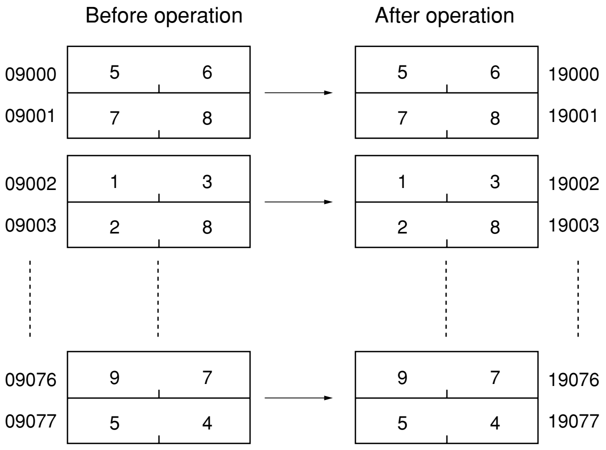
Symbol	F-70w FILE | n | S | D	
Function	"n" words data through the register S to the register S+2n−1 are transferred in batch to the "n" words area in the registers D to D+2n−1.	
Operation	S, S+1, ···S+2n−1→ D, D+1, ···D+2n−1	
Range of "n"	000 to 377 (8) (256 words for 000)	
Range of S	コ0000 to コ1577 b0000 to b0777 09000 to 09777 19000 to 19777 29000 to 29777 39000 to 39777	@コ0000 to @コ1574 @b0000 to @b0774 @09000 to @09774 @19000 to @19774 @29000 to @29774 @39000 to @39774
Range of D	コ0000 to コ1577 b0000 to b0777 09000 to 09777 19000 to 19777 29000 to 29777 39000 to 39777	@コ0000 to @コ1574 @b0000 to @b0774 @09000 to @09774 @19000 to @19774 @29000 to @29774 @39000 to @39774
Condition	Rising edge of input signal (OFF to ON)	
After operation	Contents of S, ···S+2n−1	Unchanged
	Contents of D, D+1 ⋮ D+2n-1	Contents of the register S Contents of the register S+1 ⋮ Contents of the register S+2n-1
	Flag	Unchanged

〔Example for use〕



Instruction	
S T R	04000
F -70w	
	040
	09000
	19000

When the input condition 04000 changes from OFF to ON, 040 (8) words data (32 words in decimal) in registers 09000 to 09077 are transferred in batch to the 32 words area of 19000 to 19007.
The contents of registers, 09000 to 09077, remain unaffected.



	Before operation			After operation		
09000	5	6	→	5	6	19000
09001	7	8		7	8	19001
09002	1	3	→	1	3	19002
09003	2	8		2	8	19003
⋮	⋮			⋮		⋮
09076	9	7	→	9	7	19076
09077	5	4		5	4	19077

Note 1: F-70w can use indirect address for assigning S and D such as @コ0000 or @コ09000.
For indirect address, see page 9・21 "Indirect address assignment."

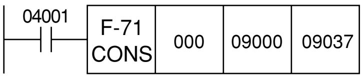
Note 2: Be sure to use even addresses for registers S and D.

Similarity instructions: F-00, F-00w, F-70, F-74, F-74w

F-71 CONS Transfer octal constant block (1 byte) (CONStant)

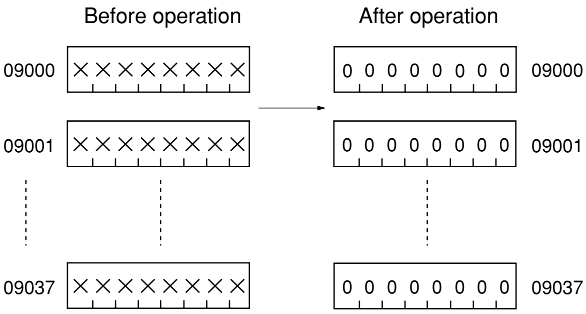
Symbol	F-71 CONS | n | D_1 | D_2	
Function	An octal constant "n" is transferred in batch through the register D_1 to the register D_2.	
Operation	n→D_1, ······, D_2	
Range of "n"	000 to 377 (8)	
Range of D_1	コ0000 to コ1577 b0000 to b0777 09000 to 09777 19000 to 19777 29000 to 29777 39000 to 39777	
Range of D_2	コ0000 to コ1577 b0000 to b0777 09000 to 09777 19000 to 19777 29000 to 29777 39000 to 39777	
Condition	Rising edge of input signal (OFF to ON)	
After operation	Contents of D_1 Contents of D_1+1 ⋮ Contents of D_2−1 Contents of D_2	Constant "n"
	Flag	Unchanged

〔Example for use〕



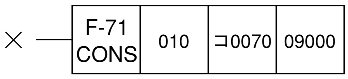
Instruction	
S T R	04001
F -71	
	000
	09000
	09037

When the input condition 04001 changes from OFF to ON, the octal constant 000 is transferred in batch from registers 09000 to 09037.

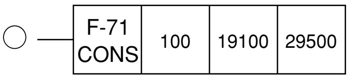


Note 1: No operation takes place if D_1 or D_2 is programmed that may override ① to ③, as shown below.

Block	Range
①	コ0000 to コ1577
②	b0000 to b0777
③	09000 to 09777 19000 to 19777 29000 to 29777 39000 to 39777

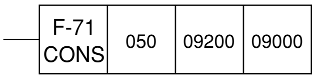


The end address of the block that contains コ0070 is コ1577.



Both D_1 and D_2 are stored in block ③.

Note 2: No operation takes place if the address D_1 is greater than D_2.



D_1(09200)>D_2(09000)

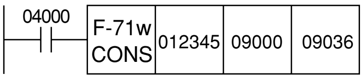
Similarity instructions: F-08, F-08w, F-71w

F-71w CONS

Transfer octal constant block (1 word) (CONStant)

Symbol	F-71w CONS n D_1 D_2	
Function	An octal constant "n" is transferred in batch in registers D_1, D_1+1, through D_2, D_2+1.	
Operation	n→ (D_1, D_1+1), ······ (D_2, D_2+1)	
Range of "n"	000000 to 177777 (8)	
Range of D_1	⊐0000 to ⊐1576 b0000 to b0776 09000 to 09776 19000 to 19776 29000 to 29776 39000 to 39776	
Range of D_2	⊐0000 to ⊐1576 b0000 to b0776 09000 to 09776 19000 to 19776 29000 to 29776 39000 to 39776	
Condition	Rising edge of input signal (OFF to ON)	
After operation	D_1, D_1+1 D_1+2, D_1+3 ⋮ D_2-2, D_2+1 D_2, D_2+1	Constant "n"
	Flag	Unchanged

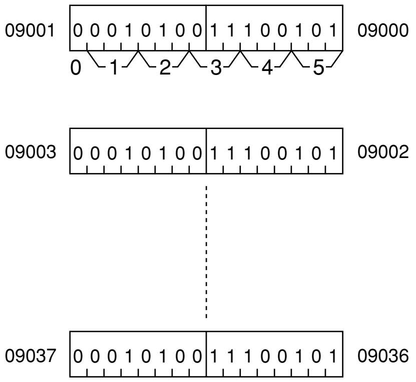
〔Example for use〕



Instruction	
S T R	04000
F -71w	
	012345
	09000
	09036

When the input condition 04000 changes from OFF to ON, the octal constant 012345 is transferred in batch from registers 09000, 09001 to 09036, 09037.

After operation



Note 1: Be sure to use even addresses for registers D_1 and D_2.

Note 2: No operation takes place if D_1 or D_2 is programmed that may override ① to ③, as shown below.

Block	Range
①	⊐0000 to ⊐1576
②	b0000 to b0776
③	09000 to 09776 19000 to 19776 29000 to 29776 39000 to 39776

×	F-71w CONS	010	⊐0070	09000	The end address of the block that contains ⊐0070 is ⊐1576.
○	F-71w CONS	100	19100	29500	Both D_1 and D_2 are stored in block ③.

Note 3: No operation takes place if the address D_1 is greater than D_2.

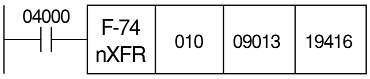
F-71w CONS	050	09200	09000	D_1(09200)＞D_2(09000)

Similarity instructions: F-08, F-08w, F-71

F-74 nXFR Transfer "n" bytes

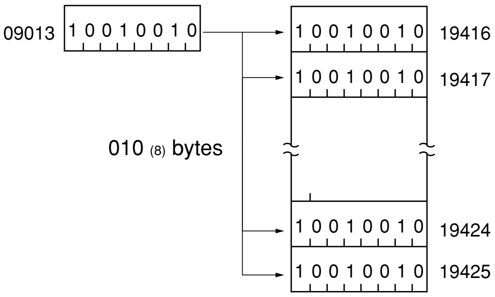
Symbol	F-74 nXFR n S D	
Function	The contents of the register S are transferred to "n" bytes registers headed by the register D.	
Operation	S→D, ···D+n−1	
Range of "n"	000 to 377 (8) (256 bytes for 000)	
Range of S	コ0000 to コ1577 b0000 to b0777 09000 to 09777 19000 to 19777 29000 to 29777 39000 to 39777	
Range of D	コ0000 to コ1577 b0000 to b0777 09000 to 09777 19000 to 19777 29000 to 29777 39000 to 39777	
Condition	Rising edge of input signal (OFF to ON)	
After operation	Contents of S	Unchanged
	Contents of D Contents of D+1 ⋮ Contents of D+n -1	Contents of the register S
	Flag	Unchanged

〔Example for use〕



Instruction	
S T R	04000
F -74	
	010
	09013
	19416

When the input condition 04000 changes from OFF to ON, the contents of the register 09013 are transferred to a 010 (8) bytes registers headed by the register 19416.



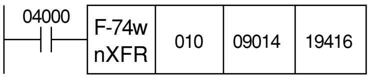
Similarity instructions: F-00, F-00w, F-70, F-70w, F-74w

F-74w nXFR

Transfer "n" words

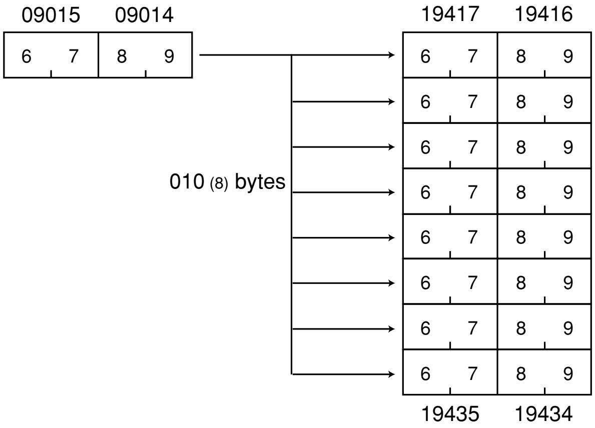
Symbol	F-74w nXFR n S D	
Function	The contents of the registers S, S+1 are transferred to "n" words registers headed by the registers D, D+1.	
Operation	S, S+1→D, D+1···D+2n-2, D+2n-1	
Range of "n"	000 to 377 (8) (256 words for 000)	
Range of S	コ0000 to コ1576 b0000 to b0776 09000 to 09776 19000 to 19776 29000 to 29776 39000 to 39776	
Range of D	コ0000 to コ1576 b0000 to b0776 09000 to 09776 19000 to 19776 29000 to 29776 39000 to 39776	
Condition	Rising edge of input signal (OFF to ON)	
After operation	Contents of S, S+1	Unchanged
	Contents of D Contents of D+1 ⋮ D+2n−2 D+2n−1	Contents of the register S Contents of the register S+1 ⋮ Contents of the register S Contents of the register S+1
	Flag	Unchanged

〔Example for use〕



Instruction	
S T R	04000
F-74w	
	010
	09014
	19416

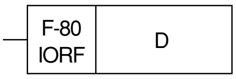
When the input condition 04000 changes from OFF to ON, the contents of the registers 09014 and 09015 are transferred to a 010 (8) words registers headed by the registers 19416 and 19417.



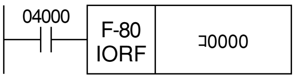
Note 1: Be sure to use even addresses for registers S and D.
Similarity instructions: F-00, F-00w, F-70, F-70w, F-74

F-80 IORF

I/O refresh (1 byte) (I/O ReFresh)

Symbol		
Function	Refreshes input/output port byte data assigned by D.	
Operation	Input port → D Output port ← D	
Range of D	コ0000 to コ0077	
Condition	When the input signal is ON (not limited to an OFF to ON change)	
After operation	Input port	Update of data memory
	Output port	Update of output state
	Flag	Unchanged

〔Example for use〕



Instruction	
S T R	04000
F -80	
	コ0000

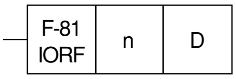
When input condition 04000 is ON, the JW10 refreshes one byte data of コ0000.

Note 1: This instruction can be used any number of times during program operation.

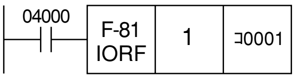
Note 2: The I/O port which is data refreshed with this instruction performs the data refresh operation also during an ordinary I/O processing of scan cycle.

F-81 IORF

I/O refresh (1 bit) (I/O ReFresh)

Symbol		
Function	Refreshes input/output port nth bit data assigned by D.	
Operation	Input port → nth bit of D Output port ← nth bit of D	
Range of n	0 to 7	
Range of D	コ0000 to コ0077	
Condition	When the input signal is ON (not limited to an OFF to ON change)	
After operation	Input port	Update of data memory
	Output port	Update of output state
	Flag	Unchanged

〔Example for use〕



Instruction	
S T R	04000
F -81	
	1
	コ0001

When input condition 04000 is ON, the JW10 refreshes first bit (00011) data of コ0001.

Note 1: This instruction can be used any number of times during program operation.

Note 2: The I/O port which is data refreshed with this instruction performs the data refresh operation also during an ordinary I/O processing of scan cycle.

F-90 REM

Remark (REMark)

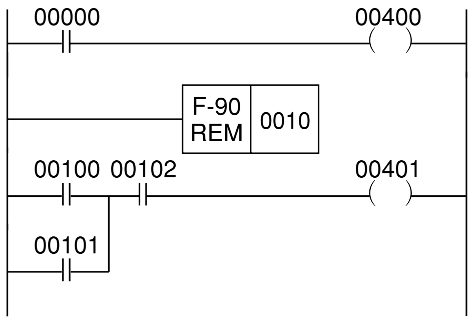
Symbol	F-90 REM | n
Function	When printing ladder chart and instruction words, executed printing line comment on the multipurpose programmer (JW-50PG etc.) and ladder software (JW-50SP/52SP/92SP).
Operation	NOP (This instruction does not cause the PC to perform an operation)
Range of "n"	0000 to 3777 (8)
After operation	Data memory of flag etc. is unchanged.

〔Description〕

- Symbols and comments for F-90 are registrered with "symbol/comment setting" by multipurpose programmer or ladder software.
  (Symbol: 16 half characters, Comments: 28 half characters)
- When printing ladder chart, the multipurpose programmer or ladder chart prints symbol and comment contents. It does not print F-90 instruction in this case.
  If @ is registered at first character of the symbol, it is paged and symbol comment contents are not printed.
- When printing instructions, the multipurpose programmer or ladder chart prints each of F-90 instruction and symbol/comment contents.
  Even if @ is registered at first character of the symbol contents, paging is not executed and the registered contents are printed.

〔Example for use〕

(1) Ladder chart programming



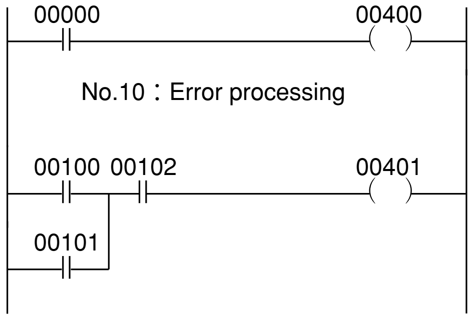
(2) Symbol/comment setting

	Address	Symbol	Comment
F-90	0010	No.10	Error processing
F-90	0011	@	

↑ Paging when printing ladder chart

⇩

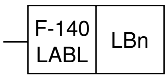
(3) Ladder chart printing



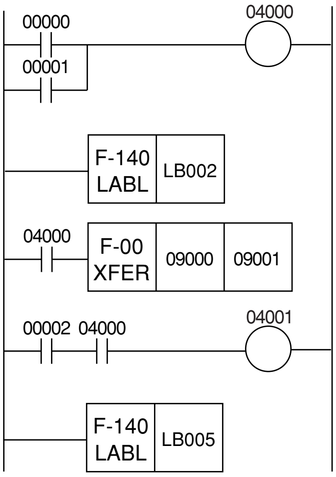
(4) Instruction printing

```
STR          00000
OUT          00400
F-90[REM]
               0010   No.10：Error processing
STR          00100
OR           00101
AND          00102
OUT          00401
```

F-140 LABL Set label (LABeL)

Symbol	
Function	Use to set Jump destination for the F-141 (JMP) instruction. Branch destination for the F-142 (CALL) instruction. Branch destination for timer interrupt. Branch destination for high-speed counter interrupt.
Range of LBn	LB 000 to LB167 (8) : For F-141, F-142 LB170 : For high-speed counter (mode 1) LB171 : For high-speed counter (mode 1-2) LB172 : For high-speed counter (mode 2) LB173 to LB176 : Reserved LB177 : 10 ms timer interruption

〔Example for use〕



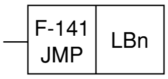
Adrs	Instruction	
00000	S T R	00000
00001	O R	00001
00002	O U T	04000
00003	F -140	
00004		LB002
00005	S T R	04000
00006	F -00	
00007		09000
00010		09001
00011	S T R	00002
00012	A N D	04000
00013	O U T	04001
00014	F -140	
00015		LB005

F-140 is a jump or subroutine label which is not to execute the program. Therefore, the data memory is retained after execution of an F-140 instruction.

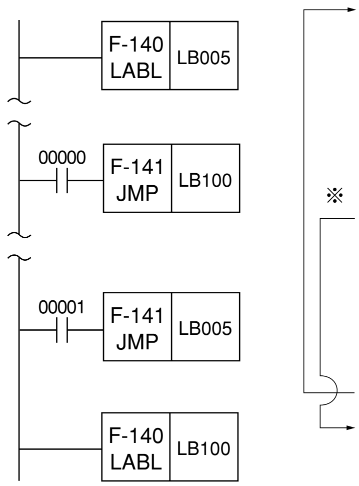
Note 1: The same label number should not be used again within the program.

F-141
JMP

Jump to label (JuMP)

Symbol	
Function	The control jumps to the label address LBn (F-140).
Range of LBn	LB000 to LB167 (8)
Condition	When the input signal is ON (not to limited to OFF to ON change)

〔Example for use〕



Adrs	Instruction
00005	F -140
00006	LB005
⋮	⋮
00777	S T R 00000
01000	F -141
01001	LB100
⋮	⋮
02002	S T R 00001
02003	F -141
02004	LB0005
02005	F -140
02006	LB100

When the input condition 00000 is ON, the control jumps to the program address 02005 of the label LB100 and instructions after the address 02007 are executed.

When the input condition 00001 is ON, the control jumps to the program address 00005 of the label LB0005 and instructions after the address 00007 are executed.

Note 1: The contents of the data memory are not affected after execution of an F-141 instruction.

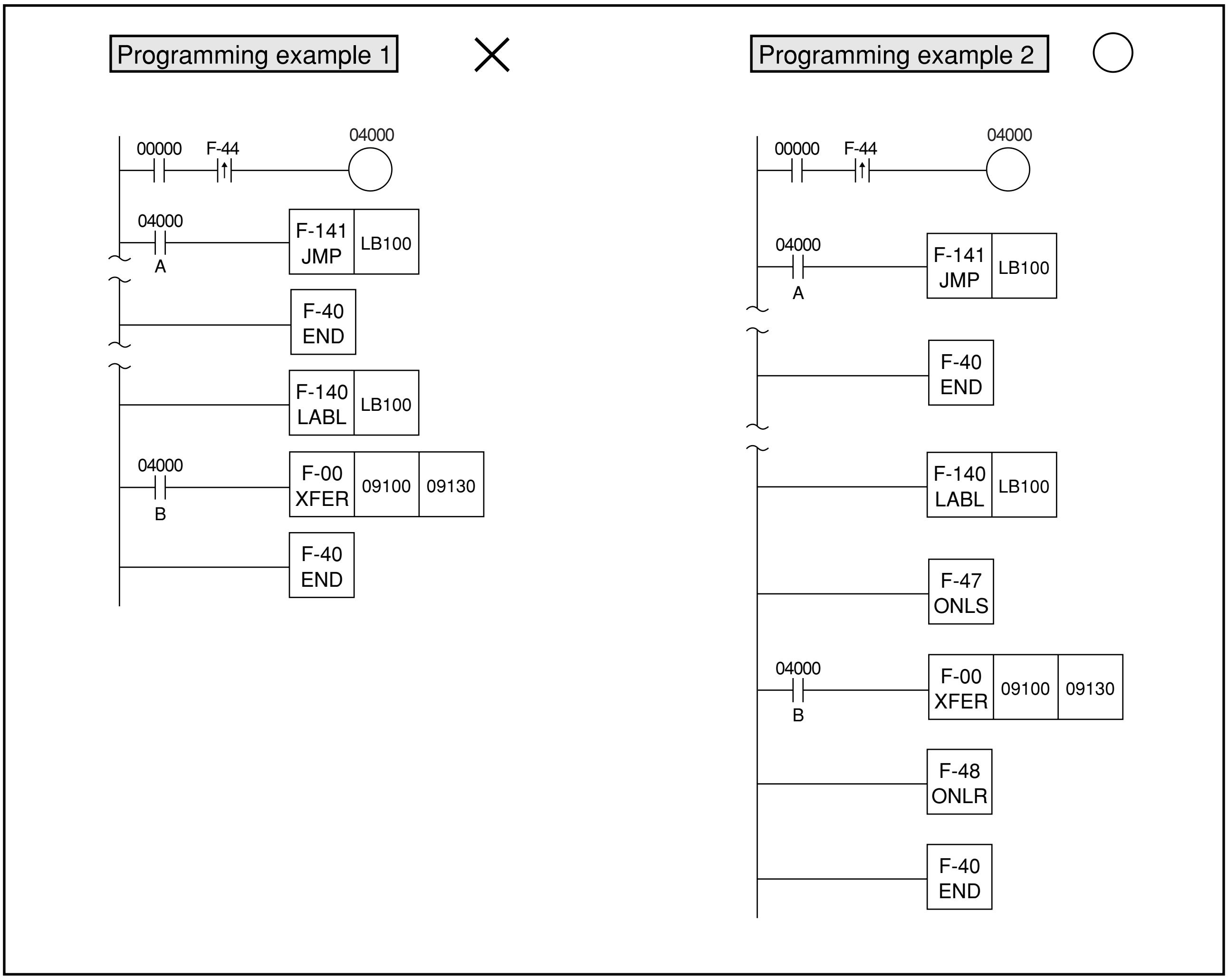
Note 2: Execution may be done with F-41 (JCS) and F-42 (JCR) for the location indicated with an ※ mark, the execution time may be saved if an F-141 (JMP) is used as it does not execute instruction down to F-140 (LABL).

Note 3: Same number may be used for the label of an F-141 instruction at any time.

Note 4: Because the control does not execute down to the jump address with an F-141 instruction, an F-40 (END) will be disregarded even if there is an F-40 before the jump address.

Note 5: The jump address label (F-140) should be written in the program, in order to avoid malfunction that may occur if there is not a jump address.

Note 6: Though the same operation takes place for the program examples 1 and 2 that had been programmed using an F-140 (LABL) or F-141 (JMP), the F-00 (XFER) instruction may not be valid for the example 1. Operation wise, the F-141 is executed and the control jumps to the destination specified by the F-141 instruction, when the contact 04000 (A contact) turns ON for both examples. And, the F-00 is executed when a next contact 04000 (B contact) is ON.



- Regarding the programming example 1, the F-141 is executed in the first cycle the contact 04000 (A contact) had turned ON and the control jumps to the destination specified by the F-141.
  For the next contact 04000 (B contact) has been ON after the jump was made, the F-00 is executed. However, the F-00 is not executed in the second cycle after the contact 04000 (A contact) has turned ON even if it is ON. Because the F-00 does not recognize the rising edge of the signal, as both contents of the accumulator before one scan cycle and the current contents of the accumulator is ON.
- Regarding the programming example 2, the F-00 after the jump is executed in the second cycle after the contact 04000 (A contact) has turned ON. This is because the level operation condition (F-47, F-48) is provided so as to execute the instruction after the jump at the time of ON.

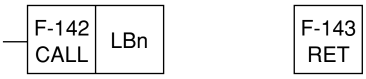
Reference

The F-00 instruction compares the contents of the previous contents of the accumulator with the current contents and executes the program when a rising edge is recognized.

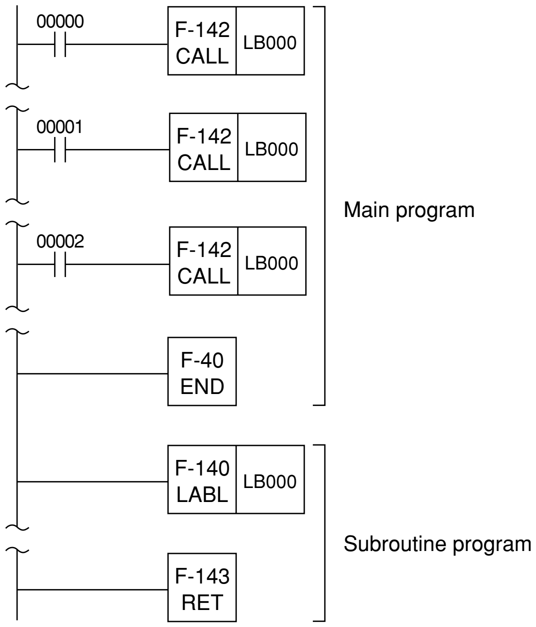
Therefore, it has to be programmed like the example 2 in order to execute the program at every operational cycle after a jump.

F-142 CALL — Call subroutine label (CALL)

F-143 RET — Return from subroutine (Return)

Symbol	
Function	The control jumps to the subroutine of LBn (F-140) and returns with an F-143 instruction.
Range of LBn	LB000 to LB167 (8)
Condition	Rising edge of input signal (OFF to ON)

〔Example for use〕



Adrs	Instruction	
00010	S T R	00000
00011	F -142	
00012		LB000
⋮		⋮
00100	S T R	00001
00101	F -142	
00102		LB000
⋮		⋮
00200	S T R	00002
00201	F -142	
00202		LB000
⋮		⋮
00300	F -40	
00301	F -140	
00302		LB000
⋮		⋮
00315	F -143	

Use of an F-142 (CALL) and F-143 (RET) in the program not only eliminates need of writing the same program repeatedly within the program and saves the program size, but it will also help structuring program.

In the above example, when the input condition 00000 changes from OFF to ON, the control jumps from the main program to the subroutine to execute the addresses 00303 through 00314, and returns to resume executing the main program at the address 00013 when an F-143 instruction is met.

Note 1: The F-143 (RET) instruction is also used to program a subroutine for high speed counter interrupt (set with #203) or timer interrupts (set with #244).

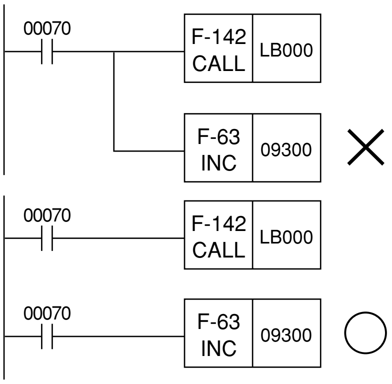
Note 2: When a subroutine is called, instructions within the subroutine are set to level operation with an ON state of signal.

Note 3: The following instructions may not be used within a subroutine.
TMR, CNT, F-30 (MCS), F-31 (MCR), F-40 (END), F-41 (JCS), F-42 (JCR), F-44 (┤↑├), F-45 (┤↓├), F-47 (ONLS), F-48 (ONLR)

Note 4: Main program must be created first before subroutine is created and it must be affixed with an F-40 (END instruction) at the end address.

Note 5: Nesting is not allowed for a subroutine.

Note 6: Execution conditions of F-142 (CALL) cannot be used commonly with instructions after this F-142 (CALL).

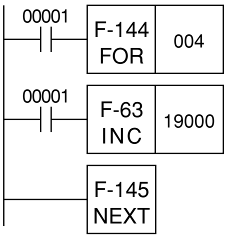


F-144 FOR — Set loop count (FOR)

F-145 NEXT — End of loop (NEXT)

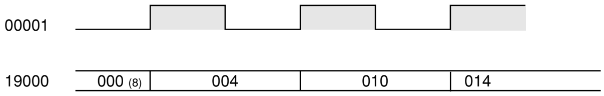
Symbol	— F-144 FOR | n F-145 NEXT
Function	Repeats to execute the program "n" times between F-144 (FOR) and F-145 (NEXT).
Range of "n"	000 to 377 (8) (256 times if 000)
Condition	Rising edge of input signal (OFF to ON)

〔Example for use〕



Instruction	
S T R	00001
F-144	
	004
S T R	00001
F-63	
	19000
F-145	

When the input condition 00001 changes from OFF to ON, the F-63 (INC) instruction is repeated for four times.



Note 1: The F-144 (FOR) must be used in conjunction with the F-145 (NEXT).

Note 2: When the F-144 (FOR) is executed, instructions between the F-144 (FOR) and F-145 (NEXT) are set ON level active.

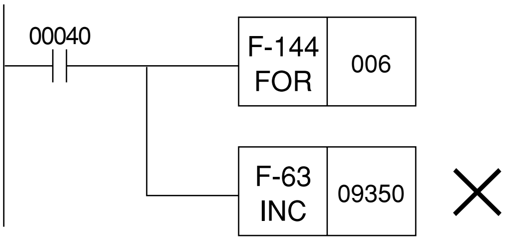
Note 3: When the instruction is not execution, the contents of the data memory between the F-144 (FOR) and F-145 (NEXT) do not change.

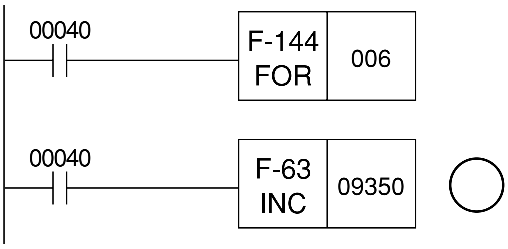
Note 4: Execution conditions of F-144 (FOR) cannot be commonly used with execution conditions of instruction after F-144 (FOR).

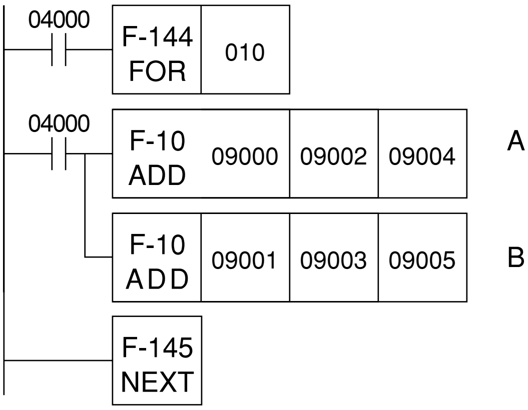
Note 5: The follwing instructions cannot be inserted between F-144 (FOR) and F-145 (NEXT). TMR, CNT, F-30 (MCS), F-31 (MCR), F-40 (END), F-41 (JCS), F-42 (JCR), F-44 (-|↑|-), F-45 (-|↓|-), F-47 (ONLS), F-48 (ONLR), F-141 (JMP), F-144 (FOR), F-145 (NEXT)

Note 6: Number of instruction between F-144 (FOR) and F-145 (NEXT) should be small as possible and consider operation time.

Note 7: Double-length operation is possible for the F-10 instruction, but the operation flag of the instruction of B as following program will not affect the instruction of A.







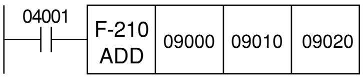
F-210 ADD

Add register and register in binary (8 bits ＋ 8 bits) (ADD)

Symbol	—[F-210 ADD] [S_1] [S_2] [D]
Function	The contents of the register S_1 are added in binary with the contents of the register S_2 and its result is stored in the register D.
Operation	$S_1+S_2 \rightarrow D$
Range of S_1	コ0000 to コ1577 b0000 to b0777 09000 to 09777 19000 to 19777 29000 to 29777 39000 to 39777
Range of S_2	コ0000 to コ1577 b0000 to b0777 09000 to 09777 19000 to 19777 29000 to 29777 39000 to 39777
Range of D	コ0000 to コ1577 b0000 to b0777 09000 to 09777 19000 to 19777 29000 to 29777 39000 to 39777
Condition	Rising edge of input signal (OFF to ON)

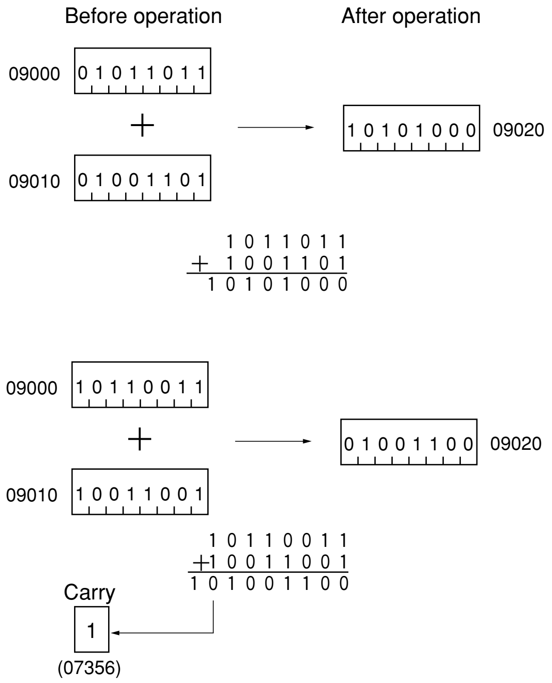
After operation	Contents of S1	Unchanged				
	Contents of S2	Unchanged				
	Contents of D	Result				
	Flag	Result	Zero 07357	Carry 07356	Error 07355	Non-carry 07354
		0	1	0	0	1
		001 to 377 (8)	0	0	0	1
		400 (8)	1	1	0	0
		Above 401 (8)	0	1	0	0

〔Example for use〕



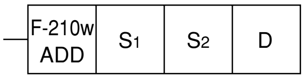
Instruction	
S T R	04001
F -210	
	09000
	09010
	09020

When the input condition 04001 changes from OFF to ON, the contents of the register 09000 and the register 09010 are added in binary and stored in the register 09020.



(Similarity instructions) F-210w, Fc210, Fc210w

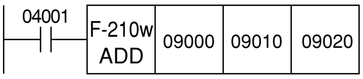
F-210w ADD — Add register and register in binary (16 bits ＋ 16 bits) (ADD)

Symbol	
Function	The contents of the registers S_1 and S_1+1 are added in binary with the contents of the registers S_2 and S_2+1 and its results are stored in the registers D and D+1.
Operation	$(S_1, S_1+1) + (S_2, S_1+1) \rightarrow D, D+1$
Range of S_1	ℶ0000 to ℶ1576 b0000 to b0776 09000 to 09776 19000 to 19776 29000 to 29776 39000 to 39776
Range of S_2	ℶ0000 to ℶ1576 b0000 to b0776 09000 to 09776 19000 to 19776 29000 to 29776 39000 to 39776
Range of D	ℶ0000 to ℶ1576 b0000 to b0776 09000 to 09776 19000 to 19776 29000 to 29776 39000 to 39776
Condition	Rising edge of input signal (OFF to ON)

After operation	Contents of S_1, S_1+1	Unchanged
	Contents of S_2, S_2+1	Unchanged
	Contents of D	Lower digits of result
	Contents of D+1	Upper digits of result
	Flag	

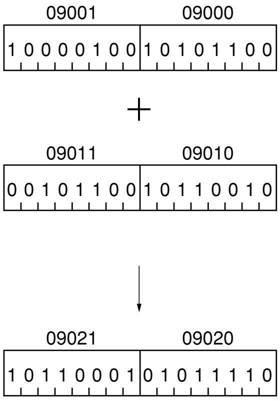
Result	Zero 07357	Carry 07356	Error 07355	Non-carry 07354
0	1	0	0	1
000001 to 177777	0	0	0	1
200000	1	1	0	0
Above 200001	0	1	0	0

〔Example for use〕



Instruction	
S T R	04001
F -210w	
	09000
	09010
	09020

When the input condition 04001 changes from OFF to ON, the contents of the registers 09000 and 09001 are added in binary with the registers 09010 and 09011 and its results are stored in the registers 09020 and 09021.



Note 1: Be sure to use even addresses for registers S_1, S_2, and D.

Similarity instructions: F-210, Fc210, Fc210w

Fc210
ADD

Add register and constant in binary (8 bits ＋ 8 bits) (ADD)

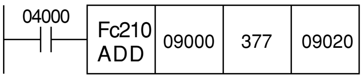
Symbol	Fc210 ADD │ S_1 │ n │ D
Function	Contents of the register S_1 are added in binary with the contents of an octal constant "n" and its result is stored in the register D.
Operation	S_1+n→D
Range of S_1	コ0000 to コ1577 b0000 to b0777 09000 to 09777 19000 to 19777 29000 to 29777 39000 to 39777
Range of "n"	000 to 377 (8)
Range of D	コ0000 to コ1577 b0000 to b0777 09000 to 09777 19000 to 19777 29000 to 29777 39000 to 39777
Condition	Rising edge of input signal (OFF to ON)

After operation		
	Contents of S1	Unchanged
	Contents of D	Result
	Flag	

Result	Zero 07357	Carry 07356	Error 07355	Non-carry 07354
0	1	0	0	1
001 to 377 (8)	0	0	0	1
400 (8)	1	1	0	0
Above 401 (8)	0	1	0	0

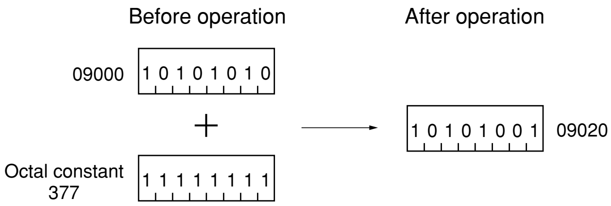
Similarity instructions: F-210, F-210w, Fc210w

〔Example for use〕



Instruction	
S T R	04000
F c210	
	09000
	377
	09020

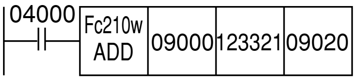
When the input condition 04000 changes from OFF to ON, the contents of the register 09000 are added in binary with the octal constant 377 and its result is stored in the register 09020.



Fc210w ADD — Add register and constant in binary (16 bits ＋ 16 bits) (ADD)

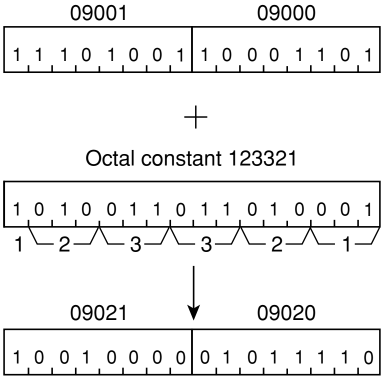
<table>
<tr><td>Symbol</td><td colspan="6"><table><tr><td>Fc210w
ADD</td><td>S_1</td><td>n</td><td>D</td></tr></table></td></tr>
<tr><td>Function</td><td colspan="6">Contents of the register S_1 and S_1+1 are added in binary with an octal constant "n" and its result are stored in the registers D and D+1.</td></tr>
<tr><td>Operation</td><td colspan="6">(S_1, S_1+1)+n→D, D+1</td></tr>
<tr><td>Range of S_1</td><td colspan="6">コ0000 to コ0006
b0000 to b0776
09000 to 09776
19000 to 19776
29000 to 29776
39000 to 39776</td></tr>
<tr><td>Range of "n"</td><td colspan="6">000000 to 177777 (8)</td></tr>
<tr><td>Range of D</td><td colspan="6">コ0000 to コ0006
b0000 to b0776
09000 to 09776
19000 to 19776
29000 to 29776
39000 to 39776</td></tr>
<tr><td>Condition</td><td colspan="6">Rising edge of input signal (OFF to ON)</td></tr>
<tr><td rowspan="8">After operation</td><td>Contents of S, S_1,</td><td colspan="5">Unchanged</td></tr>
<tr><td>Contents of D</td><td colspan="5">Lower digits of result</td></tr>
<tr><td>Contents of D+1</td><td colspan="5">Upper digits of result</td></tr>
<tr><td rowspan="5">Flag</td><td>Result (8)</td><td>Zefo
07357</td><td>Carry
07356</td><td>Error
07355</td><td>Non-carry
07354</td></tr>
<tr><td>0</td><td>1</td><td>0</td><td>0</td><td>1</td></tr>
<tr><td>000001 to 177777</td><td>0</td><td>0</td><td>0</td><td>1</td></tr>
<tr><td>200000</td><td>1</td><td>1</td><td>0</td><td>0</td></tr>
<tr><td>Above 200001</td><td>0</td><td>1</td><td>0</td><td>0</td></tr>
</table>

〔Example for use〕



Instruction	
STR	04000
Fc210w	
	09000
	123321
	09020

When the input condition 04000 changes from OFF to ON, the contents of the registers 09000 and 09001 are added in binary with the octal constant 123321 and its results are stored in the registers 09020 and 09021.



Note 1: Be sure to use even addresses for registers S_1 and D.

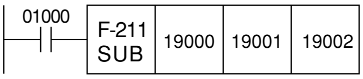
Similarity instrucitons: F210, F210w, Fc210

F-211 SUB Subtract register from register in binary (8 bits − 8 bits) (SUBtract)

Symbol		—F-211 SUB | S1 | S2 | D	
Function		Contents of the register S1 are subtracted by the contents of the register S2 in binary and its result is stored in the register D.	
Operation		$S_1 - S_2 \rightarrow D$	
Range of S1		コ0000 to コ1577 b0000 to b0777 09000 to 09777 19000 to 19777 29000 to 29777 39000 to 39777	
Range of S2		コ0000 to コ1577 b0000 to b0777 09000 to 09777 19000 to 19777 29000 to 29777 39000 to 39777	
Range of D		コ0000 to コ1577 b0000 to b0777 09000 to 09777 19000 to 19777 29000 to 29777 39000 to 39777	
Condition		Rising edge of input signal (OFF to ON)	
After operation	Contents of S1	Unchanged	
	Contents of S2	Unchanged	
	Contents of D	Result	
	Flag		

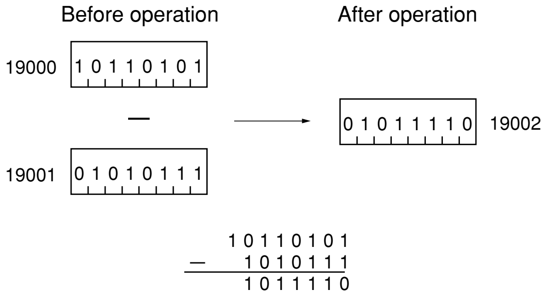
Result	Zero 07357	Carry 07356	Error 07355	Non-carry 07354
0	1	0	0	1
1 to 377 (8)	0	0	0	1
Negative value	0	1	0	0

〔Example for use〕

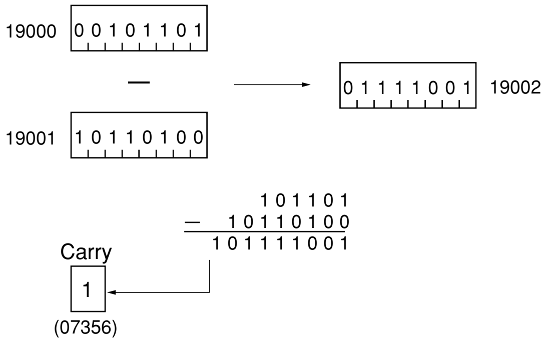


Instruction	
STR	01000
F-211	
	19000
	19001
	19002

When the input condition 01000 changes from OFF to ON, the contents of the register 19000 are subtracted by the contents of the register 19001 and its result is stored in the register 19002.



```
  1 0 1 1 0 1 0 1
−   1 0 1 0 1 1 1
    1 0 1 1 1 1 0
```

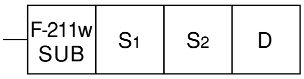


```
      1 0 1 1 0 1
−   1 0 1 1 0 1 0 0
  1 0 1 1 1 1 0 0 1
```

Similarity insturctions: F-211w, Fc211, Fc211w

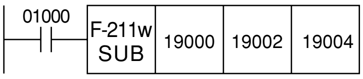
F-211w
SUB

Subtract register from register in binary (16 bits — 16 bits) (SUBtract)

Symbol	 F-211w SUB S_1 S_2 D
Function	Contents of the registers S_1 and S_1+1 are subtracted by the contents of the registers S_2 and S_2+1 in binary and its results are stored in the registers D and D+1.
Operation	(S_1, S_1+1) — (S_2, S_2+1) →D, D+1
Range of S_1	コ0000 to コ1576 b0000 to b0776 09000 to 09776 19000 to 19776 29000 to 29776 39000 to 39776
Range of S_2	コ0000 to コ1576 b0000 to b0776 09000 to 09776 19000 to 19776 29000 to 29776 39000 to 39776
Range of D	コ0000 to コ1576 b0000 to b0776 09000 to 09776 19000 to 19776 29000 to 29776 39000 to 39776
Condition	Rising edge of input signal (OFF to ON)

After operation	Contents of S_1, S_1+1	Unchanged				
	Contents of S_2, S_2+1	Unchanged				
	Contents of D	Lower digits of result				
	Contents of D+1	Upper digits of result				
	Flag	Result (8)	Zero 07357	Carry 07356	Error 07355	Non-carry 07354
		0	1	0	0	1
		1 to 177777 (8)	0	0	0	1
		Negative value	0	1	0	0

〔Example for use〕



Instruction	
STR	01000
F-211	
	19000
	19002
	19004

When the input condition 01000 changes from OFF to ON, the contents of the registers 19000 and 19001 are subtracted by the contents of the registers 19002 and 19003 and its results are stored in the registers 19004 and 19005.

19001	19000
1 0 1 0 0 0 0 1	0 1 0 0 0 0 1 0

—

19003	19002
0 1 1 1 0 0 0 0	1 0 1 1 1 0 0 1

↓

19005	19004
0 0 1 1 0 0 0 0	1 0 0 0 1 0 0 1

Note 1: Be sure to use even addresses for registers S_1, S_2, and D.

Similarity insturctions: F-211, Fc211, Fc211w

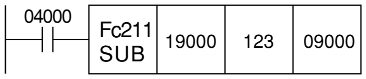
Fc211 SUB

Subtract constant from register in binary (8 bits − 8 bits) (SUBtract)

Symbol	Fc211 SUB | S_1 | n | D
Function	Contents of the register S_1 are subtracted by an octal constant "n" in binary and its result is stored in the register D.
Operation	$S_1 - n \rightarrow D$
Range of S_1	コ0000 to コ1577 b0000 to b0777 09000 to 09777 19000 to 19777 29000 to 29777 39000 to 39777
Range of "n"	000 to 377 (8)
Range of D	コ0000 to コ1577 b0000 to b0777 09000 to 09777 19000 to 19777 29000 to 29777 39000 to 39777
Condition	Rising edge of input signal (OFF to ON)

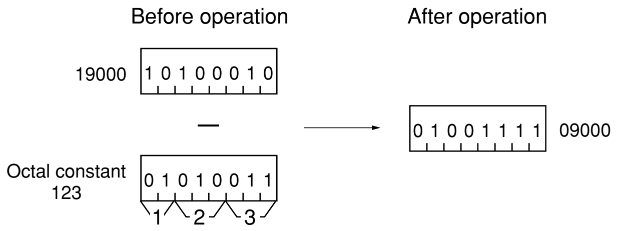
After operation						
	Contents of S_1	Unchanged				
	Contents of D	Result				
	Flag	Result (8)	Zero 07357	Carry 07356	Error 07355	Non-carry 07354
		0	1	0	0	1
		1 to 377 (8)	0	0	0	1
		Negative value	0	1	0	0

〔Example for use〕



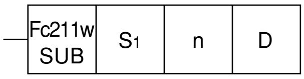
Instruction	
STR	04000
Fc211	19000 123 09000

When the input condition 04000 changes from OFF to ON, the contents of the register 19000 are subtracted by the octal constant 123 in binary and its result is stored in the register 09000.



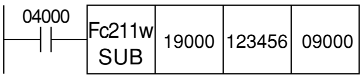
Similarity insturctions: F-211, F-211w, Fc211w

Fc211w SUB Subtract constant from register in binary (16 bits — 16 bits) (SUBtract)

Symbol			
Function	Contents of the registers S_1 and S_1+1 are subtracted by the octal constant "n" in binary and its results are stored in the registers D and D+1.		
Operation	(S_1, S_1+1) − n→D, D+1		
Range of S_1	コ0000 to コ1576 b0000 to b0776 09000 to 09776 19000 to 19776 29000 to 29776 39000 to 39776		
Range of "n"	000000 to 177777 (8)		
Range of D	コ0000 to コ1576 b0000 to b0776 09000 to 09776 19000 to 19776 29000 to 29776 39000 to 39776		
Condition	Rising edge of input signal (OFF to ON)		
After operation	Contents of S_1, S_1+1	Unchanged	
	Contents of D	Lower digits of result	
	Contents of D+1	Upper digits of result	

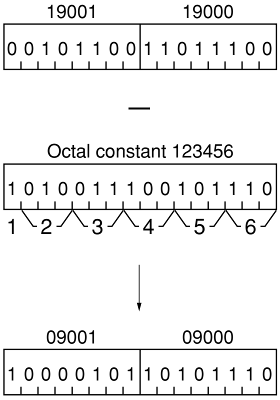
Flag: Result (8)	Zero 07357	Carry 07356	Error 07355	Non-carry 07354
0	1	0	0	1
1 to 177777 (8)	0	0	0	1
Negative value	0	1	0	0

〔Example for use〕



Instruction	
STR	04000
Fc211w	19000 123456 09000

When the input condition 04000 changes from OFF to ON, the contents of the registers 19000 and 19001 are subtracted by the octal constant 123456 and its results are stored in the registers 09000 and 09001.



Note 1: Be sure to use even addresses for registers S_1, and D.

Similarity insturctions: F-211, F-211w, Fc211

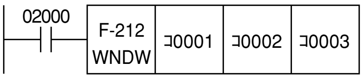
F-212
WNDW

Window comparator (between 1 byte register)

Symbol	F-212 WNDW | S_1 | S_2 | S_3
Function	Contents of the registers S_1, S_2 and S_3 are compared and its results are stored in the flags.
Operation	Compare result→flag
Range of S_1	コ0000 to コ1577 b0000 to b0777 09000 to 09777 19000 to 19777 29000 to 29777 39000 to 39777
Range of S_2	コ0000 to コ1577 b0000 to b0777 09000 to 09777 19000 to 19777 29000 to 29777 39000 to 39777
Range of S_3	コ0000 to コ1577 b0000 to b0777 09000 to 09777 19000 to 19777 29000 to 29777 39000 to 39777
Condition	When the input signal is ON (not limited to an OFF to ON change)

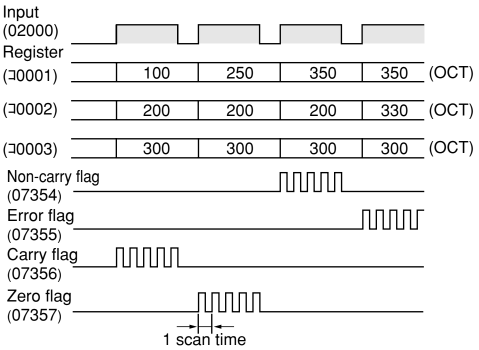
After operation	Contents of S_1	Unchanged				
	Contents of S_2	Unchanged				
	Contents of S_3	Unchanged				
	Flag	Contents of register	Zero 07357	Carry 07356	Error 07355	Non-carry 07354
		$S_1<S_2$	0	1	0	0
		$S_2 \leqq S_1 \leqq S_3$	1	0	0	0
		$S_3<S_1$	0	0	0	1
		$S_3<S_2$	0	0	1	0

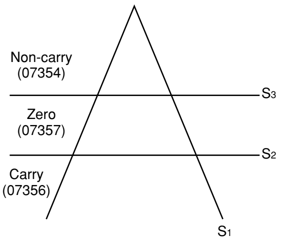
〔Example for use〕



Instruction	
STR	02000
F-212	コ0001 コ0002 コ0003

When the input condition 02000 changes from OFF to ON, the contents of the register コ0001 are checked if コ0001＜コ0002, コ0002≦コ0001≦コ0003, and コ0003＜コ0001, its results are stored in the carry flag (07356), zero flag (07357) and non-carry flag (07354). Operation takes place only when the condition is コ0002≦コ0003. Operation does not take place and the error flag (07355) is set active, if コ0002＞コ0003.



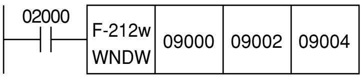


Similarity insturctions: F-212w, Fc212, Fc212w

F-212w WNDW Window comparator (between 1 word registers)

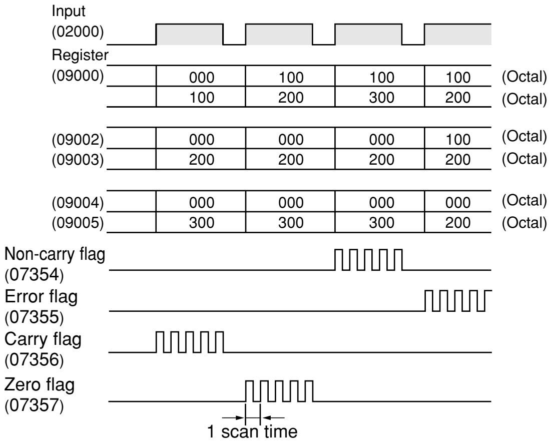
Symbol		F-212w WNDW │ S_1 │ S_2 │ S_3				
Function		Contents of the registers S_1, S_1+1, S_2, S_2+1, S_3, S_3+1 are compared and its results are stored in the flags.				
Operation		Result→flag				
Range of S_1		コ0000 to コ1576 b0000 to b0776 09000 to 09776 19000 to 19776 29000 to 29776 39000 to 39776				
Range of S_2		コ0000 to コ1576 b0000 to b0776 09000 to 09776 19000 to 19776 29000 to 29776 39000 to 39776				
Range of S_3		コ0000 to コ1576 b0000 to b0776 09000 to 09776 19000 to 19776 29000 to 29776 39000 to 39776				
Condition		Rising edge of input signal (OFF to ON)				
After operation	Contents of S_1, S_1+1	Unchanged				
	Contents of S_2, S_2+1	Unchanged				
	Contents of S_3, S_3+1	Unchanged				
	Flag	Result	Zero 07357	Carry 07356	Error 07355	Non-carry 07354
		$S_1, S_1+1 < S_2, S_2+1$	0	1	0	0
		$S_2, S_2+1 \leqq S_1, S_1+1 \leqq S_3, S_3+1$	1	0	0	0
		$S_3, S_3+1 < S_1, S_1+1$	0	0	0	1
		$S_3, S_3+1 < S_2, S_2+1$	0	0	1	0

〔Example for use〕



Instruction	
STR	02000
F-212w	09000 09002 09004

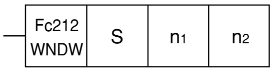
When the input condition 02000 changes from OFF to ON, the word contents of the registers 09000 and 09001, are checked if 09000, 09001＜09002, 09003 and 09002, 09003≦09000, 09001≦09000, 09001≦09004, 09005, and 09004, 09005＜09000, 09001, and its results are stored in the carry flag (07356), zero flag (07357) and non-carry flag (07354).
Operation takes place only when 09002, 09003≦ 09004, 09005 is established and operation stop with the error flag (07355) in activation if 09004, 09005＜09002, 09003.



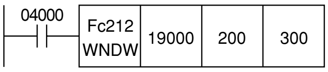
Note 1: Be sure to use even addresses for registers S_1, S_2 and S_3.
Similarity insturctions: F-212, Fc212, Fc212w

Fc212 WNDW

Window comparator (between 1 byte octal constants)

<table>
<tr><td colspan="2">Symbol</td><td colspan="5"> </td></tr>
<tr><td colspan="2">Function</td><td colspan="5">Contents of the register S_1 are compared with the octal constants n_1 and n_2 and its results are stored in the flags.</td></tr>
<tr><td colspan="2">Operation</td><td colspan="5">Result→flag</td></tr>
<tr><td colspan="2">Range of S_1</td><td colspan="5">コ0000 to コ1577
b0000 to b0777
09000 to 09777
19000 to 19777
29000 to 29777
39000 to 39777</td></tr>
<tr><td colspan="2">Range of n_1</td><td colspan="5">000 to 377 (8)</td></tr>
<tr><td colspan="2">Range of n_2</td><td colspan="5">000 to 377 (8)</td></tr>
<tr><td colspan="2">Condition</td><td colspan="5">When the input signal is ON
(not limited to an OFF to ON change)</td></tr>
<tr><td rowspan="6">After operation</td><td>Contents of S_1</td><td colspan="5">Unchanged</td></tr>
<tr><td rowspan="5">Flag</td><td>Contents of register</td><td>Zero 07357</td><td>Carry 07356</td><td>Error 07355</td><td>Non-carry 07354</td></tr>
<tr><td>$S_1<n_1$</td><td>0</td><td>1</td><td>0</td><td>0</td></tr>
<tr><td>$n_1≦S_1≦n_2$</td><td>1</td><td>0</td><td>0</td><td>0</td></tr>
<tr><td>$n_2<S_1$</td><td>0</td><td>0</td><td>0</td><td>1</td></tr>
<tr><td>$n_2<n_1$</td><td>0</td><td>0</td><td>1</td><td>0</td></tr>
</table>

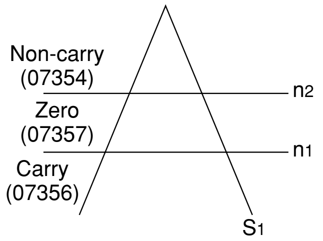
〔Example for use〕



Instruction	
STR	04000
Fc212	
	19000
	200
	300

When the input condition 04000 changes from OFF to ON, the contents of the register 19000 are checked if 19000<200, 200≦19000≦300, and 300<19000, and its results are stored in the carry flag (07356), zero flag (07357) and non-carry flag (07354).

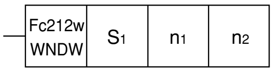
Contents of 19000	Zero 07357	Carry 07356	Error 07355	Non-carry 07354
150	0	1	0	0
250	1	0	0	0
350	0	0	0	1



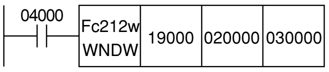
Similarity insturctions: F-212, F-212w, Fc212w

Fc212w WNDW

Window comparator (between 1 word octal constants)

<table>
<tr><td colspan="2">Symbol</td><td colspan="5"> </td></tr>
<tr><td colspan="2">Function</td><td colspan="5">Contents of the registers S_1 and S_1+1 are compared with the octal constants n_1 and n_2 and its results are stored in the flags.</td></tr>
<tr><td colspan="2">Operation</td><td colspan="5">Result→flag</td></tr>
<tr><td colspan="2">Range of S_1</td><td colspan="5">コ0000 to コ1576
b0000 to b0776
09000 to 09776
19000 to 19776
29000 to 29776
39000 to 39776</td></tr>
<tr><td colspan="2">Range of n_1</td><td colspan="5">000000 to 177777 (8)</td></tr>
<tr><td colspan="2">Range of n_2</td><td colspan="5">000000 to 177777 (8)</td></tr>
<tr><td colspan="2">Condition</td><td colspan="5">When the input signal is ON
(not limited to an OFF to ON change)</td></tr>
<tr><td rowspan="6">After operation</td><td>Contents of S_1, S_1+1</td><td colspan="5">Unchanged</td></tr>
<tr><td rowspan="5">Flag</td><td>Contents of register</td><td>Zero 07357</td><td>Carry 07356</td><td>Error 07355</td><td>Non-carry 07354</td></tr>
<tr><td>$S_1, S_1+1<n_1$</td><td>0</td><td>1</td><td>0</td><td>0</td></tr>
<tr><td>$n_1≦S_1, S_1+1≦n_2$</td><td>1</td><td>0</td><td>0</td><td>0</td></tr>
<tr><td>$n_2<S_1, S_1+1$</td><td>0</td><td>0</td><td>0</td><td>1</td></tr>
<tr><td>$n_2<n_1$</td><td>0</td><td>0</td><td>1</td><td>0</td></tr>
</table>

〔Example for use〕



Instruction	
STR	04000
Fc212w	
	19000
	020000
	030000

When the input condition 04000 changes from OFF to ON, the word contents of the registers 19000 and 19001 are checked if 19000, 19001<020000, 020000≦19000, 19001≦030000, and 030000<19000, 19001 and its results are stored in the carry flag (07356), zero flag (07357) and non-carry flag (07354).

Contents of 19000,19001	Zero 07357	Carry 07356	Error 07355	Non-carry 07354
015000	0	1	0	0
025000	1	0	0	0
035000	0	0	0	1

Note 1: Be sure to use even addresses for register S_1.

Similarity insturctions: F-212, F-212w, Fc212

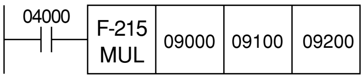
F-215
MUL

Multiply register by register in binary (8 bits × 8 bits) (MULtiply)

<table>
<tr><td colspan="2">Symbol</td><td colspan="4">F-215 MUL │ S1 │ S2 │ D</td></tr>
<tr><td colspan="2">Function</td><td colspan="4">The contents of the register S1 are multiplied by the contents of the register S2 in binary and its results are stored in the registers D and D+1.</td></tr>
<tr><td colspan="2">Operation</td><td colspan="4">S1×S2→D, D+1</td></tr>
<tr><td colspan="2">Range of S1</td><td colspan="4">コ0000 to コ1577
b0000 to b0777
09000 to 09777
19000 to 19777
29000 to 29777
39000 to 39777</td></tr>
<tr><td colspan="2">Range of S2</td><td colspan="4">コ0000 to コ1577
b0000 to b0777
09000 to 09777
19000 to 19777
29000 to 29777
39000 to 39777</td></tr>
<tr><td colspan="2">Range of D</td><td colspan="4">コ0000 to コ1576
b0000 to b0776
09000 to 09776
19000 to 19776
29000 to 29776
39000 to 39776</td></tr>
<tr><td colspan="2">Condition</td><td colspan="4">Rising edge of input signal (OFF to ON)</td></tr>
<tr><td rowspan="6">After operation</td><td>Contents of S1</td><td colspan="4">Unchanged</td></tr>
<tr><td>Contents of S2</td><td colspan="4">Unchanged</td></tr>
<tr><td>Contents of D</td><td colspan="4">Lower digits of result</td></tr>
<tr><td>Contents of D+1</td><td colspan="4">Upper digits of result</td></tr>
<tr><td rowspan="2">Flag</td><td>Zero
07357</td><td>Carry
07356</td><td>Error
07355</td><td>Non-carry
07354</td></tr>
<tr><td>0</td><td>0</td><td>0</td><td>0</td></tr>
</table>

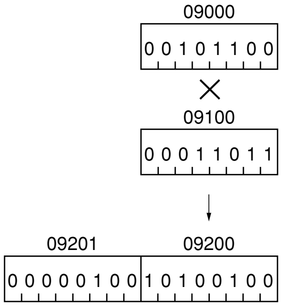
Similarity insturctions: F-215w, Fc215, Fc215w

〔Example for use〕



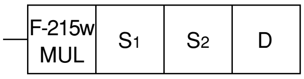
Instruction	
STR	04000
F-215	
	09000
	09100
	09200

When the input condition 04000 changes from OFF to ON, the contents of 09000 are multiplied by the contents of the register 09100 in binary and its results are stored in the registers 09200 and 09201.



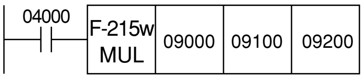
```
         1 0 1 1 0 0
 X         1 1 0 1 1
         1 0 1 1 0 0
       1 0 1 1 0 0
     1 0 1 1 0 0
   1 0 1 1 0 0
 1 0 0 1 0 1 0 0 1 0 0
```

F-215w MUL Multiply register by register in binary (16 bits × 16 bits) (MULtiply)

Symbol	 F-215w MUL S_1 S_2 D
Function	The contents of the registers S_1 and S_1+1 are multiplied by the contents of the registers S_2 and S_2+1 in binary and its results are stored in the registers D, D+1, D+2 and D+3.
Operation	(S_1, S_1+1)×(S_2, S_2+1)→D, D+1, D+2, D+3
Range of S_1	コ0000 to コ1576 b0000 to b0776 09000 to 09776 19000 to 19776 29000 to 29776 39000 to 39776
Range of S_2	コ0000 to コ1576 b0000 to b0776 09000 to 09776 19000 to 19776 29000 to 29776 39000 to 39776
Range of D	コ0000 to コ1574 b0000 to b0774 09000 to 09774 19000 to 19774 29000 to 29774 39000 to 39774
Condition	Rising edge of input signal (OFF to ON)

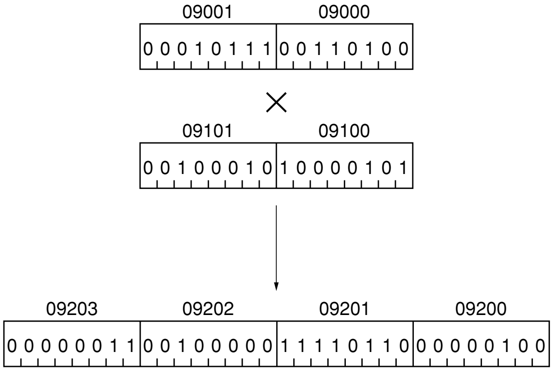
After operation	Contents of S_1, S_1+1	Unchanged			
	Contents of S_2, S_2+1	Unchanged			
	Contents of D	Lower digits of result			
	Contents of D+1	Result			
	Contents of D+2	Result			
	Contents of D+1	Upper digits of result			
	Flag	Zero 07357	Carry 07356	Error 07355	Non-carry 07354
		0	0	0	0

〔Example for use〕



Instruction	
STR	04000
F-215	
	09000
	09100
	09200

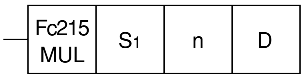
When the input condition 04000 changes from OFF to ON, the 16 bits contents of the registers 09000 and 09001 are multiplied by the contents of the registers 09100 and 09101 in binary and its results are stored in the registers 09200 to 09203.



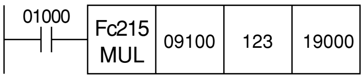
Note 1: Be sure to use even addresses for registers S_1, S_2, and D.

Similarity insturctions: F-215, Fc215, Fc215w

Fc215 MUL Multiply register by constant in binary (8 bits × 8 bits) (MULtiply)

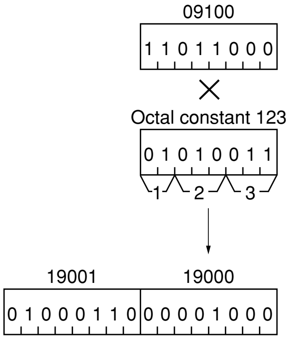
<table>
<tr><td colspan="2">Symbol</td><td colspan="4"> </td></tr>
<tr><td colspan="2">Function</td><td colspan="4">The contents of the register S are multiplied by the octal constant "n" in binary and its results are stored in the registers D and D+1.</td></tr>
<tr><td colspan="2">Operation</td><td colspan="4">$S_1 \times n \rightarrow D, D+1$</td></tr>
<tr><td colspan="2">Range of S_1</td><td colspan="4">コ0000 to コ1577
b0000 to b0777
09000 to 09777
19000 to 19777
29000 to 29777
39000 to 39777</td></tr>
<tr><td colspan="2">Range of "n"</td><td colspan="4">000 to 377 (8)</td></tr>
<tr><td colspan="2">Range of D</td><td colspan="4">コ0000 to コ1576
b0000 to b0776
09000 to 09776
19000 to 19776
29000 to 29776
39000 to 39776</td></tr>
<tr><td colspan="2">Condition</td><td colspan="4">Rising edge of input signal (OFF to ON)</td></tr>
<tr><td rowspan="5">After operation</td><td>Contents of S_1</td><td colspan="4">Unchanged</td></tr>
<tr><td>Contents of D</td><td colspan="4">Lower digits of result</td></tr>
<tr><td>Contents of D+1</td><td colspan="4">Upper digits of result</td></tr>
<tr><td rowspan="2">Flag</td><td>Zero
07357</td><td>Carry
07356</td><td>Error
07355</td><td>Non-carry
07354</td></tr>
<tr><td>0</td><td>0</td><td>0</td><td>0</td></tr>
</table>

〔Example for use〕



Instruction	
STR	01000
Fc215	
	09100
	123
	19000

When the input condition 01000 changes from OFF to ON, the contents of the register 09100 are multiplied by the octal constant 123 in binary and its results are stored in the registers 19000 and 19001.



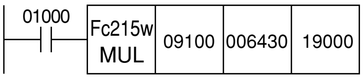
Similarity insturctions: F-215, F-215w, Fc215w

Fc215w
MUL

Multiply register by constant in binary (16 bits × 16 bits) (MULtiply)

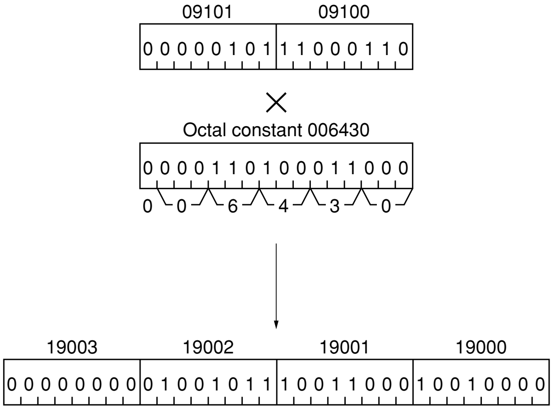
Symbol		Fc215w MUL　S_1　n　D			
Function		The 16 bits contents of the registers S_1 and S_1 +1 are multiplied by an octal content "n" in binary and its results are stored in the registers D, D+1, D+2 and D+3.			
Operation		(S_1, S_1+1)×n→D, D+1, D+2, D+3			
Range of S_1		コ0000 to コ1576 b0000 to b0776 09000 to 09776 19000 to 19776 29000 to 29776 39000 to 39776			
Range of "n"		000000 to 177777 (8)			
Range of D		コ0000 to コ1574 b0000 to b0774 09000 to 09774 19000 to 19776 29000 to 29774 39000 to 39774			
Condition		Rising edge of input signal (OFF to ON)			
After operation	Contents of S_1, S_1+1	Unchanged			
	Contents of D	Lower digits of result			
	Contents of D+1	Result			
	Contents of D+2	Result			
	Contents of D+3	Upper digits of result			
	Flag	Zero 07357	Carry 07356	Error 07355	Non-carry 07354
		0	0	0	0

〔Example for use〕



Instruction	
STR	01000
Fc215w	09100 006430 19000

When the input condition 01000 changes from OFF to ON, the 16 bits contents of the registers 09100 and 09101 are multiplied by the octal constant 006430 in binary and its results are stored in the registers 19000 to 19003.



Note 1: Be sure to use even addresses for registers S_1 and D.

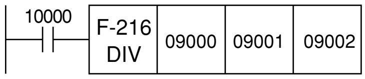
Similarity insturctions: F-215, F-215w, Fc215

F-216 DIV Divide register by register in binary (8 bits ÷ 8 bits) (DIVide)

Symbol		F-216 DIV	S_1	S_2	D	
Function		The 8 bits contents of the register S_1 are divided by the 8 bits contents of the register S_2 in binary and its quotient is stored in the register D and remainder in the register D+1.				
Operation		$S_1 \div S_2 \rightarrow D, D+1$				
Range of S_1		コ0000 to コ1577 b0000 to b0777 09000 to 09777 19000 to 19777 29000 to 29777 39000 to 39777				
Range of S_2		コ0000 to コ1577 b0000 to b0777 09000 to 09777 19000 to 19777 29000 to 29777 39000 to 39777				
Range of D		コ0000 to コ1576 b0000 to b0776 09000 to 09776 19000 to 19776 29000 to 29776 39000 to 39776				
Condition		Rising edge of input signal (OFF to ON)				
After operation	Contents of S_1	Unchanged				
	Contents of S_2	Unchanged				
	Contents of D	Quotient	Unchanged if the contents of the registers, S_2 is 000 (8).			
	Contents of D+1	Remainder				
	Flag	Contents of S_2	Zero 07357	Carry 07356	Error 07355	Non-carry 07354
		000 (8)	0	0	1	0
		Other than above			0	

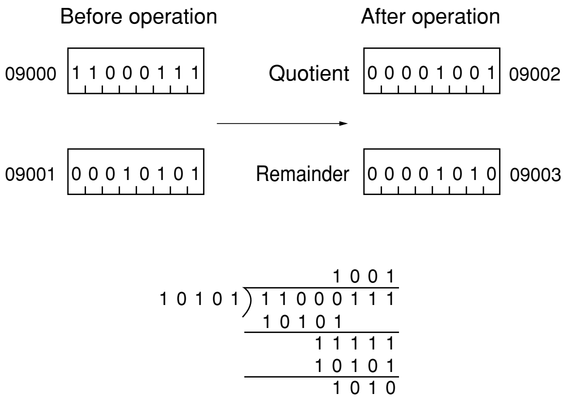
Similarity insturctions: F-216w, Fc216, Fc216w

〔Example for use〕



Instruction	
STR	10000
F-216	
	09000
	09001
	09002

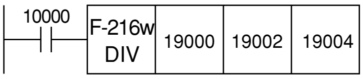
When the input condition 10000 changes from OFF to ON, the contents of the register 09000 are divided by the contents of the register 09001 in binary and its quotient is stored in the registers are 09002 and remainder in the register 09003.



F-216w DIV Divide register by register in binary (15 bits ÷ 15 bits) (DIVide)

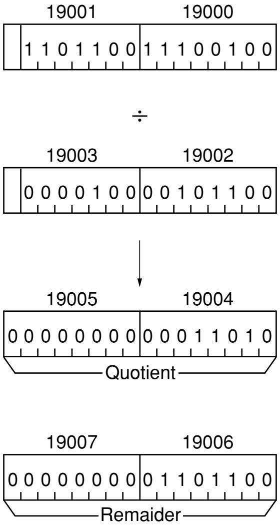
<table>
<tr><td colspan="2">Symbol</td><td colspan="5">F-216w DIV | S_1 | S_2 | D</td></tr>
<tr><td colspan="2">Function</td><td colspan="5">The 15 bits contents of the registers S_1 and S_1+1 are divided by the 15 bits contents of the registers S_2 and S_2+1 in binary and its quotient is stored in the registers D and D+1 and the remainder in the registers D+2 and D+3.</td></tr>
<tr><td colspan="2">Operation</td><td colspan="5">(S_1, S_1+1) ÷ (S_2, S_2+1)
→D, D+1, D+2, D+3</td></tr>
<tr><td colspan="2">Range of S_1</td><td colspan="5">コ0000 to コ1576
b0000 to b0776
09000 to 09776
19000 to 19776
29000 to 29776
39000 to 39776</td></tr>
<tr><td colspan="2">Range of S_2</td><td colspan="5">コ0000 to コ1576
b0000 to b0776
09000 to 09776
19000 to 19776
29000 to 29776
39000 to 39776</td></tr>
<tr><td colspan="2">Range of D</td><td colspan="5">コ0000 to コ1574
b0000 to b0774
09000 to 09774
19000 to 19774
29000 to 29774
39000 to 39774</td></tr>
<tr><td colspan="2">Condition</td><td colspan="5">Rising edge of input signal (OFF to ON)</td></tr>
<tr><td rowspan="9">After operation</td><td>Contents of S_1, S_1+1</td><td colspan="5">Unchanged</td></tr>
<tr><td>Contents of S_2, S_2+1</td><td colspan="5">Unchanged</td></tr>
<tr><td>Contents of D</td><td colspan="2">Lower digits of quotient</td><td colspan="3" rowspan="4">Unchanged if the contents of the registers S_2 and S_2+1 are 000000 (8).</td></tr>
<tr><td>Contents of D+1</td><td colspan="2">Upper digits of quotient</td></tr>
<tr><td>Contents of D+2</td><td colspan="2">Lower digits of remainder</td></tr>
<tr><td>Contents of D+3</td><td colspan="2">Upper digits of remainder</td></tr>
<tr><td rowspan="3">Flag</td><td>Contents of S_2, S_2+1</td><td>Zero
07357</td><td>Carry
07356</td><td>Error
07355</td><td>Non-carry
07354</td></tr>
<tr><td>000000 (8)</td><td rowspan="2">0</td><td rowspan="2">0</td><td>1</td><td rowspan="2">0</td></tr>
<tr><td>Other than above</td><td>0</td></tr>
</table>

〔Example for use〕



Instruction	
STR	10000
F-216w	
	19000
	19002
	19004

When the input condition 10000 changes from OFF to ON, the 15 bits contents of the registers 19000 and 19001 are divided by the 15 bits contents of the registers 19002 and 19003 in binary and its quotient is stored in the registers 19004 and 19005 and the remainder in 19006 and 19007.



MSB (bit 7) of the registers 19001 and 19003 will be disregarded.

Note 1: Be sure to use even addresses for registers S_1, S_2 and D.

Similarity instructions: F-216, Fc216, Fc216w

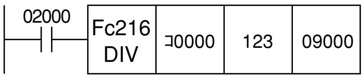
Fc216
DIV

Divide register by constant in binary (8 bits ÷ 8 bits) (DIVide)

Symbol	Fc216 DIV | S_1 | n | D
Function	Contents of the register S_1 are divided by an octal constant "n" in binary and its quotient is stored in the register D and the remainder in D+1.
Operation	$S_1 \div n \rightarrow D, D+1$
Range of S_1	コ0000 to コ1577 b0000 to b0777 09000 to 09777 19000 to 19777 29000 to 29777 39000 to 39777
Range of "n"	000 to 377 (8)
Range of D	コ0000 to コ1576 b0000 to b0776 09000 to 09776 19000 to 19776 29000 to 29776 39000 to 39776
Condition	Rising edge of input signal (OFF to ON)

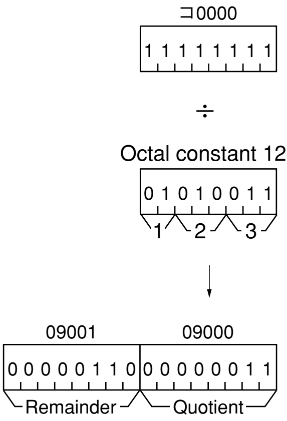
After operation	Contents of S	Unchanged				
	Contents of D	Quotient		Unchanged if "n" =000.		
	Contents of D+1	Remainder				
	Flag	Contents of S_2	Zero 07357	Carry 07356	Error 07355	Non-carry 07354
		000 (8)	0	0	1	0
		Other than above			0	

〔Example for use〕



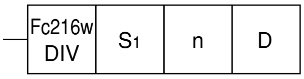
Instruction	
STR	02000
Fc216	コ0000
	123
	09000

When the input condition 02000 changes from OFF to ON, the contents of the register コ0000 are divided by the octal constant 123 and its quotient is stored in the register 09000 and the remainder in 09001.

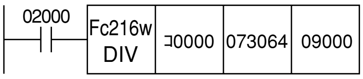


Similarity insturctions: F-216, F-216w, Fc216w

Fc216w DIV — Divide register by constant in binary (15 bits ÷ 15 bits) (DIVide)

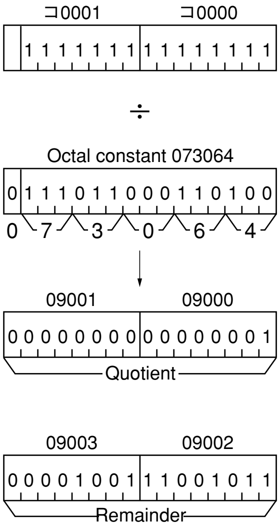
Symbol						
Function	The 15 bits contents of the registers S_1 and S_1+1 are divided by an octal constant "n" in binary and its quotient is stored in the registers D and D+1 and the remainder in D+2 and D+3.					
Operation	$(S_1, S_1+1) \div n \rightarrow D, D+1, D+2, D+3$					
Range of S_1	コ0000 to コ1576 b0000 to b0776 09000 to 09776 19000 to 19776 29000 to 29776 39000 to 39776					
Range of "n"	000000 to 077777 (8)					
Range of D	コ0000 to コ1574 b0000 to b0774 09000 to 09774 19000 to 19774 29000 to 29774 39000 to 39774					
Condition	Rising edge of input signal (OFF to ON)					
After operation	Contents of S_1, S_1+1	Unchanged				
	Contents of D	Lower digits of quotient	Unchanged if "n" =000000.			
	Contents of D+1	Upper digits of quotient				
	Contents of D+2	Lower digits of remainder				
	Contents of D+3	Upper digits of remainder				
	Flag	Octal constant "n"	Zero 07357	Carry 07356	Error 07355	Non-carry 07354
		000000	0	0	1	0
		Other than above			0	

〔Example for use〕



Instruction	
STR	02000
Fc216w	コ0000 073064 09000

When the input condition 02000 changes from OFF to ON, the 15 bits contents of the registers コ0000 and コ0001 are divided by the octal constant 073064 in binary and its quotient is stored in the registers 09000 and 09001 and the remainder in 09002 and 09003.



MSB of the register コ0001 will be disregarded.

Note 1: Be sure to use even addresses for registers S_1 and D.

Similarity insturctions: F-216, F-216w, Fc216

9-4 Ladder design precautions

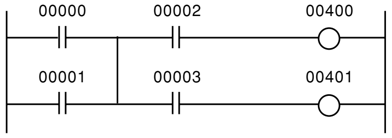
Since the programmable controller operates in the serial sequence, the ladder chart designed for the relay board may not be directly executed by the programmable controller. Also, it may not require the use of the relay board one-way control diode for the programmable controller, but it allows the use of unlimited number of auxiliary contact points.

Difference in ladder design between the relay board and the programmable controller should be well understood to comprise more effective ladder chart.

[1] Relay circuit that needs revision

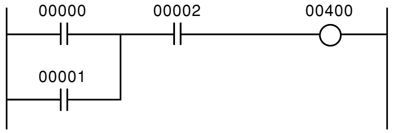
Example 1:

(a)



The programmable controller can not execute the ladder chart (a) without revising it.

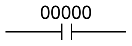
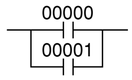
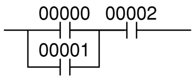
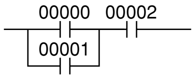
(b)

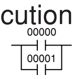


Instruction	
STR	00000
OR	00001
AND	00002
OUT	00400

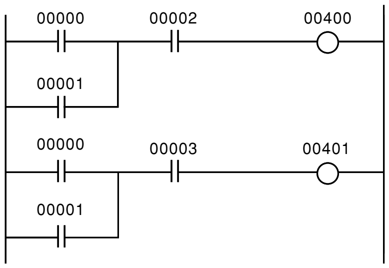
The part (b) can be executed with the program shown.

Figure below shows transition in the accumulator contents, when the program (b) is executed.

Instruction	Accumulator contents
STR 00000	
OR 00001	Result of 
AND 00002	Result of 
OUT 00100	Result of 

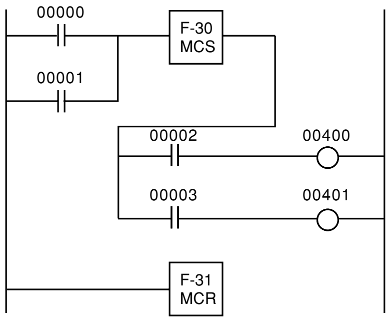
As the result represented by 0 or 1 is stored every time one programmable step is executed in the accumulator, execution up to AND 00002 deletes the result of  so that it could not be reflected on 00003. So, it has to be revised in the following was for a programmable controller ladder chart.

(a)—1



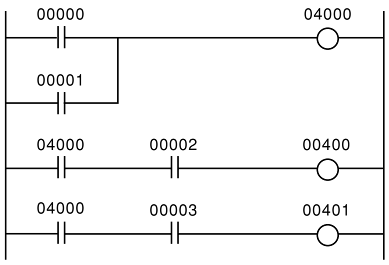
STR	00000
OR	00001
AND	00002
OUT	00400
STR	00000
OR	00001
AND	00003
OUT	00401

(a)—2



STR	00000
OR	00001
F-30	
STR	00002
OUT	00400
STR	00003
OUT	00401
F-31	

(a)—3

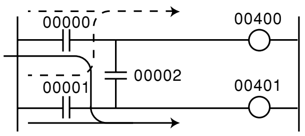


(Use as auxiliary relay for 04000.)

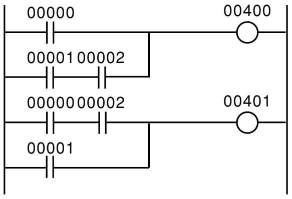
STR	00000
OR	00001
OUT	04000
STR	04000
AND	00002
OUT	00400
STR	04000
AND	00003
OUT	00401

Example 2:
As current flows to 00002 from both 00000 and 00001 with the relay board ladder chart of (a), it does the action identical to the ladder chart that revised for the programmable controller program (b).

(a)



(b)

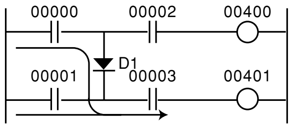


Instruction	
STR	00000
STR	00001
AND	00002
OR STR	
OUT	00400
STR	00000
AND	00002
OR	00001
OUT	00401

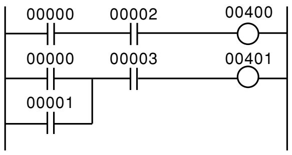
There is no concept applicable for the programmable controller that the current flows to one contact symbol from both lines as in 00002 of (a) above. Because execution takes place serial from the address 00000 to the END instruction for the programmable controller, it is not possible to go through the same contact symbol once again on the ladder chart.

Example 3:
On account of the one-way control diode D1 in the relay board circuit (a), the current does not flow from 00001 through 00002, but it performs the same action as the programmable controller ladder chart that revised for the programmable controller program (b).

(a)



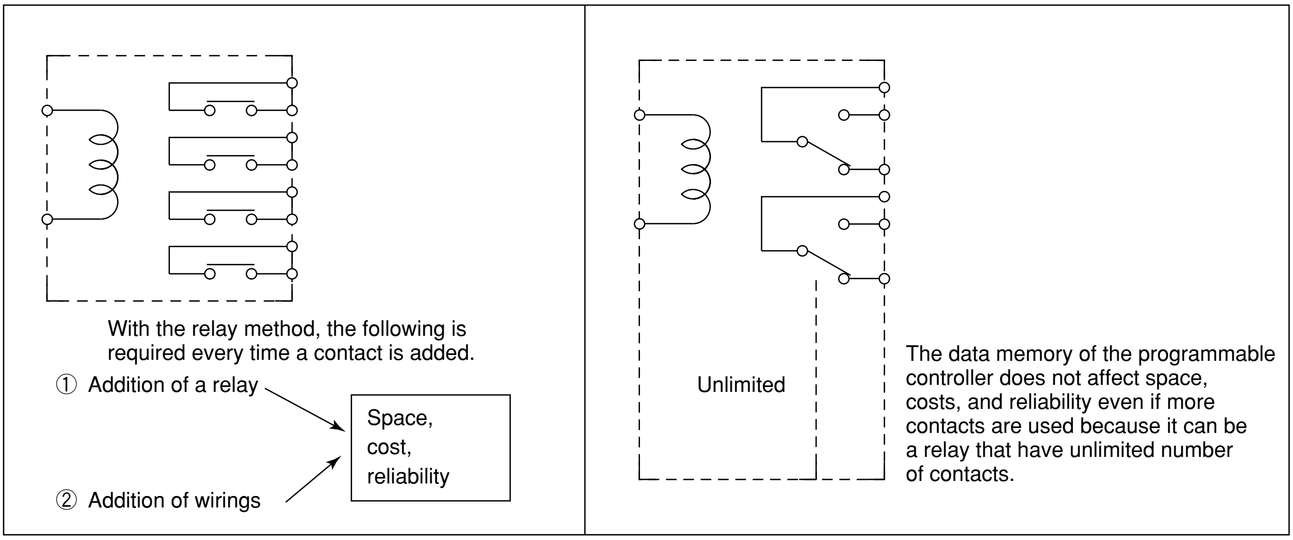
(b)



Instruction	
STR	00000
AND	00002
OUT	00400
STR	00000
OR	00001
AND	00003
OUT	00401

It is not possible for the programmable controller to program the one-way control diode D1 as in (a).

Three examples given are often used for the relay board to permit use of the relay with less contacts and simple wiring. On the other hand, with the programmable controller, there is no need of paying attention for number of contacts used as it has the data memory that offers unlimited number of contacts. It is preferable to have the ladder chart designed for recognition at a glance.



With the relay method, the following is required every time a contact is added.

The data memory of the programmable controller does not affect space, costs, and reliability even if more contacts are used because it can be a relay that have unlimited number of contacts.

[2] Input and output batch processing

- As already discussed in "8-1 Operational cycle," the JW10 performs data exchange between the data memory and the I/O port at every scan cycle.
- The I/O processing is scanned from a younger number first of relay number.

① For the input port, the ON/OFF state of the external contact connected to the input port is read into the data memory.

② For the output port, the ON/OFF state of the respective data memory is transferred to the output port.

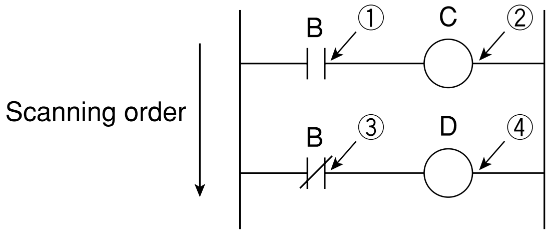
- After completing the above operation to the all I/O module, it enters the user program execution.
- Because the programmable controller executed I/O processing in the batch mode, it will be necessary to pay attention to the following points designing the ladder.

(1) As ON/OFF transition of the external contact is read into the data memory one at a time of every scan cycle during the I/O processing, a change in the ON/OFF state of the external device during that scan cycle may not affect the data memory contents (assigned to the input).
Therefore, the "Input racing phenomenon" discussed in the following passage does not occur.

(2) For the resultant ON/OFF state is written from the data memory to the output port once in each scan cycle of the I/O processing, the result is therefore sent to the output port during the I/O processing of a next scan cycle.

Input racing phenomenon

The following phenomenon may occur in the program controller that the input port ON/OFF state is read every time an instruction is executed.



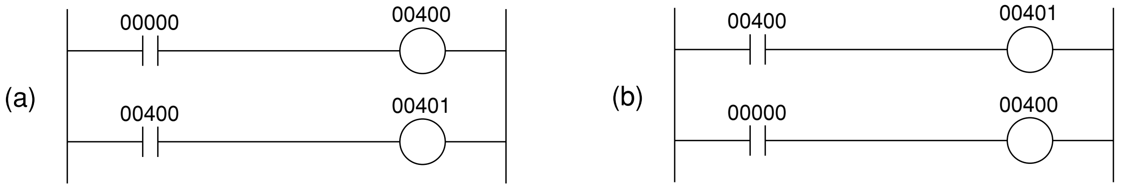
(The program that sets the coil C ON when the input B is ON and the coil D ON when the input B is OFF.)

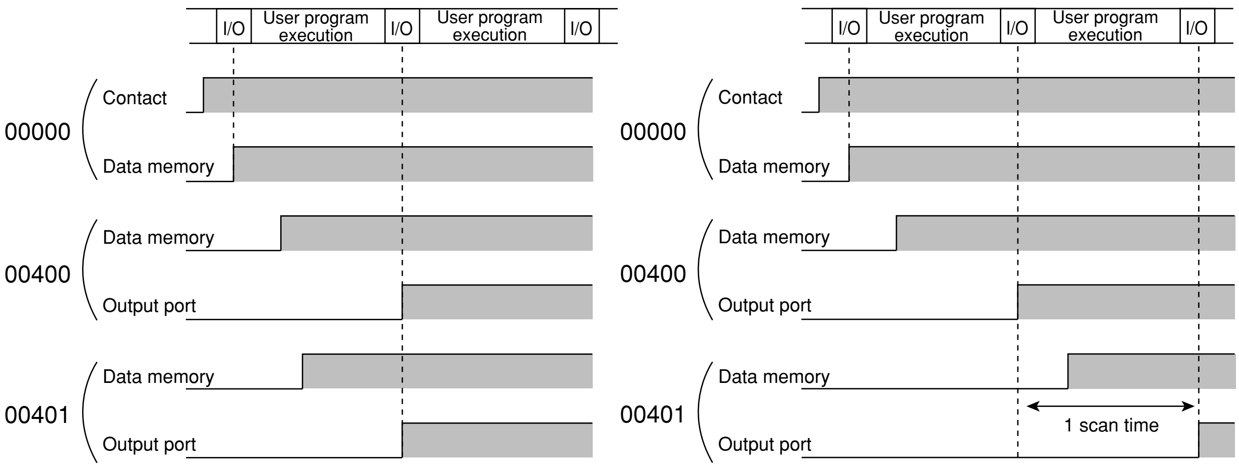
In the above program, C=$\overline{D}$ should be established. But, assume now that B was ON (C=ON) when the state of the input B is read from the input port to the accumulator at step ①. If the state of the input B has changed to OFF before the operation at ③, B is handled OFF during the operation at ③. This will establish a contradiction that the coil D is ON, while both C and D are ON.
It may result in the trouble whose cause is unknown as malfunction may or may not be evoked depending on the input timing. However, such a problem may not occur with the batch I/O processing programmable controller.

[3] Influence by programmed sequence

The programmable controller performs operation in serial sequence from the top to the end of the program step, and the same operation is repeated (cyclic scan method).

(1) Swapping orders programmed





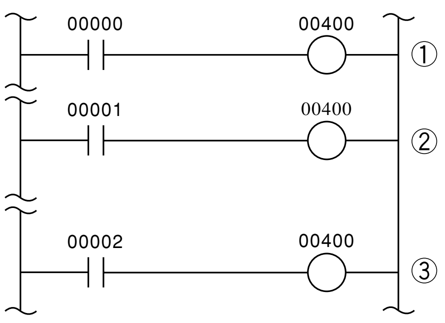
When the input 00000 turns ON for the program of (a), the outputs 00400 and 00401 turn ON in the same scan cycle, but 00401 turns ON with a delay of 1 scan cycle for with (b).

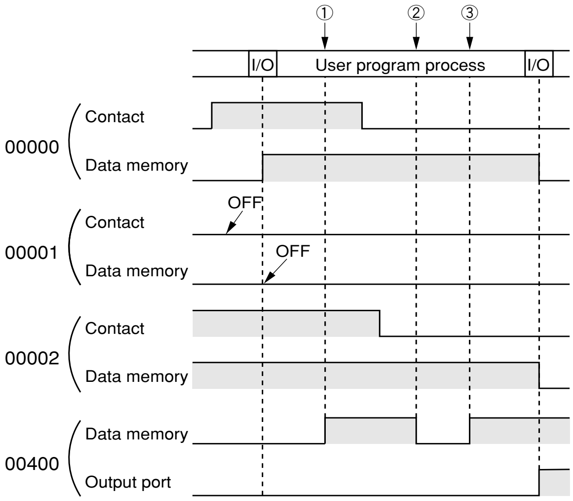
To use a coil auxiliary contact, consideration must be given when programming that "a change in the state of the auxiliary contact before the coil, comes effective in a next scan cycle that the change took place for the coil."

(2) Multiple use of the coil

If a 55(H) is stored in the register #055 in the system memory, multiple use of the coil (OUT instruction) will be possible. (The option is available in version 1.4 or later ROM code.)

If the same relay number is used more than once as a coil, the contents of data memory will change according to the program's contents. In this case, the JW10 will read the operation results of the program step that was written from data memory to the output section.





[4] Program check

- If there is any grammatical errors on a ladder program, "user program error 1" occurs due to self-diagnosis function (error code: 24(H)) and the JW10 cannot start operation. (See page 8・8 "8-3 Self diagnosis".)
- Error detected program address can be seen in system memory #052 and #053. (See page 7・12).
- A support tool such as the hand-held programmer JW-13PG may be possible to check grammatical error on the ladder program. This function is referred to "program check."
- The table below shows error messages and their program check contents using the hand-held programmer JW-13PG. (See page appendix・15 for operation procedure of JW-13PG).

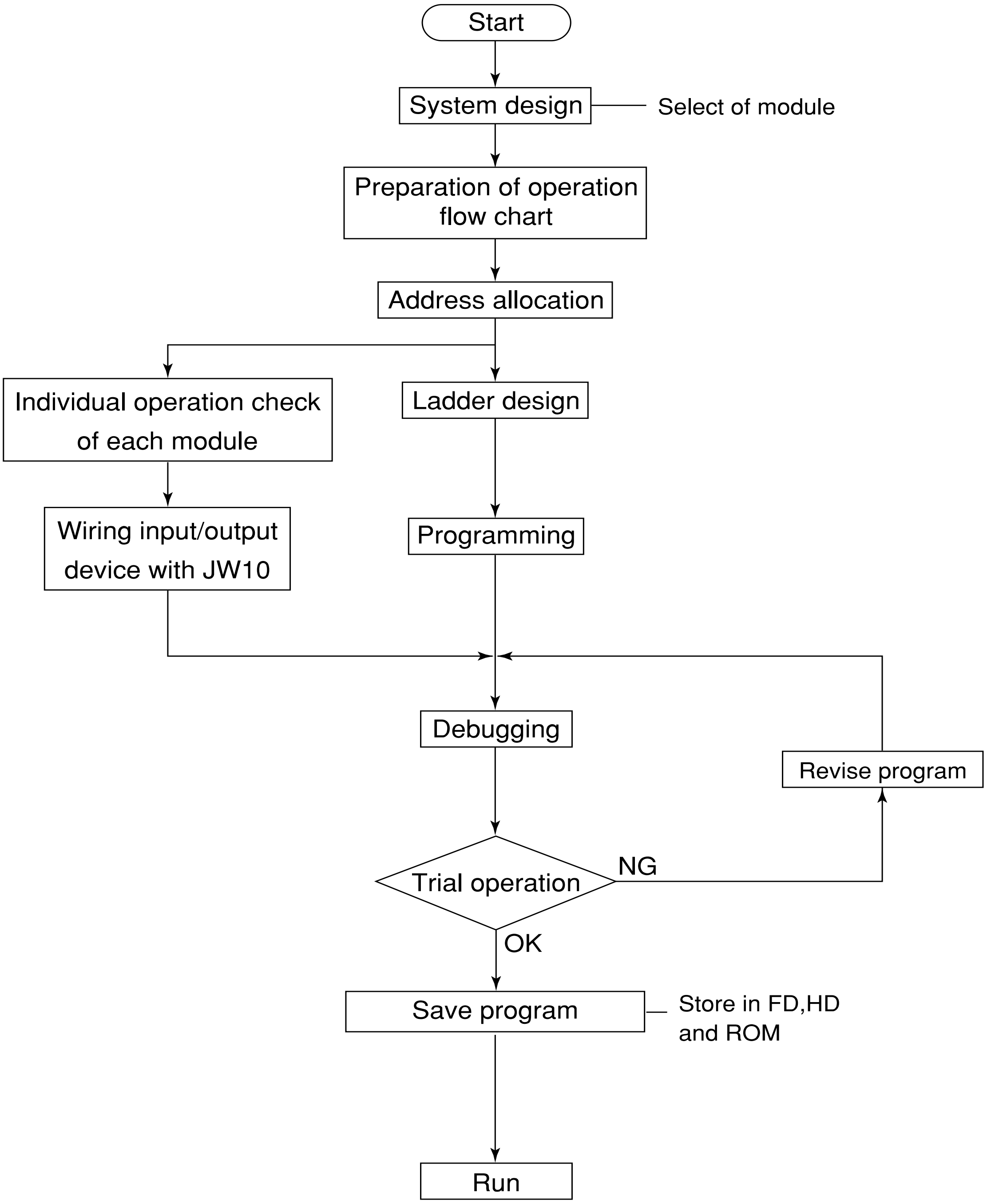
Error message	Error address	Cause	Hints for countermeasure
STACK OVER	Stack over occurred address.	STR (NOT) instruction was used too often.	Delete STR (NOT) instruction or insert AND (OR) STR instruction.
STACK UNDER	Stack under occurred address.	Lack of STR (NOT) instruction or too much use of AND (OR) STR instruction.	Insert STR (NOT) instruction or delete AND (OR) STR instruction.
MCR ERROR	MCR error occurred address.	F-31 (MCR) is used on non-F-30 (MCS) program.	Delete F-31 (MCR) or insert F-30 (MCS).
JCS ERROR	F-41(JCS) double used address.	Use of another F-41 (JCS) within the range of previous F-41 (JCS).	Delete F-41 (JCS).
JCR ERROR	JCR error occurred address.	F-42 (JCR) is used on non-F-41 (JCS) program.	Delete F-42 (JCR) or insert F-41 (JCS).
DOUBLE OUT	The same OUT instruction detected address.	Used the same relay number of OUT instruction twice.	Change assigned relay number of OUT instruction.
DOUBLE NUMBER	The same TMR/ CNT number double used address.	Used the same TMR/CNT number twice.	Change TMR/CNT number.
NO END ERROR	End address.	F-40 (END) instruction does not exist on the program.	Write F-40 (END) instruction.
LEVEL ERROR	Level error occurred address.	F-47 (ONLS) is used in the range of F-47 (ONLS).	Deleted F-47 (ONLS).
		F-48 (ONLR) is used in the non-F-47 (ONLS).	Delete F-48 (ONLR) or insert F-47 (ONLS).
NO LABEL	No label F-141 (JMP) or F-142 (CALL) address.	There is no label on jump destination of F-141 (JMP) or subroutine destination of F-142 (CALL).	Insert F-140 (LABL).
DOUBLE LABEL	2nd same label detected address.	The same label number I is used as F-140 (LABL).	Correct label number of F-140 (LABL).
FOR/NEXT ERROR	FOR - NEXT error occurred address	F-144 (FOR) is used in the F-144 (FOR) range.	Delete F-144 (FOR).
		F-145 (NEXT) is used in the non-F-144 (FOR) range.	Delete F-145 (NEXT) or insert F-144 (FOR).

Note 1 : If 55(H) is stored in the register #055 in the system memory, the JW10 can continue operation, even if it detects an error while checking the program.(The option is available in version 1.4 or later ROM code.)
However, we recommend setting register #055 to 00(H) which is "stop operation on error" because the JW10 may not operate as intended after an error is encountered.

Chapter 10. System Design

10-1 System design procedure

The system design procedure is virtually as the same as those of the other conventional relay controllers. The following is an example of the system design procedure of the JW10.

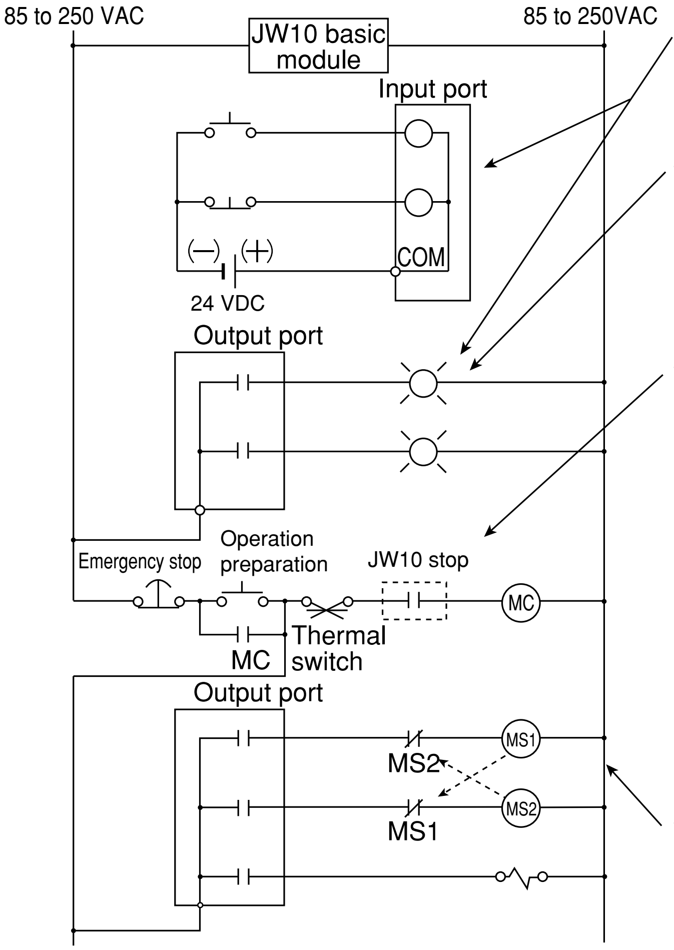


10-2 Cautions on system design

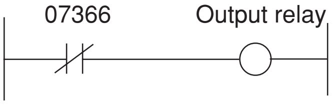
A principle difference between a programmable controller (PC) and a conventional relay circuit is that a PC controls each operation cyclically (in series), whereas relay circuit controls it in parallel.
Therefore, relay circuits limit the effect of an abnormal operation to a block.
However, a PC allows abnormal operations of the whole system when an abnormal condition occur.
In order to create a fail-safe system, we recommend preparing independent external protective circuits for following functions, which may cause a breakdown of machine or injury to workers.

- Emergency stop circuit,
- Protection circuit,
- Operating circuit of high voltage device.

Also, be aware of the operation response time, as a PC operates using cyclic control.



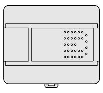
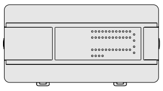
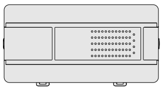
- Connecting the input port and output port for lighting lamps in front of emergency stop circuit makes it possible to grasp the stop condition of devices.
- Output status while the JW10 stops operation is determined by setting value of system memory #206.
  #206 = 00(H): Reset all outputs (initial value)
  #206 = 55(H): Latch the status just before the stoppage.
- Emergency stop circuit
  1) In case of JW-1424K/1442K/1624K/1642K
  Connect halt output of the JW10
  (Relay output 250 VAC/30 VDC, 1A)
  2) In case of JW-1324K/1342K
  Normally turns ON output relay and connects it with an emergency stop circuit.
  In this case, set system memory #206 as 00(H)



- Interlock circuit
  Prepare external interlock circuit to prevent reverse operation, damage of machines and injury of workers.

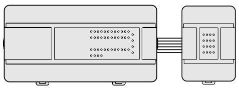
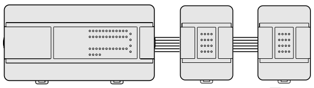
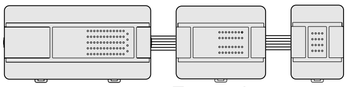
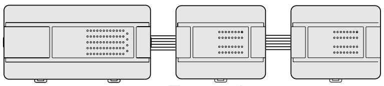
10-3 Allocation of relay number

- Relay numbers are allocated with sequential allocation taking 00000 and 00400 as top number for input relay and output relay, respectively.
- The below shows relay number allocation when the system is configured to maximum for each basic module.

Basic module	Input/output relay address (upper section: relay number, lower section: byte address)		
		Basic module	Expansion module (At maximum configuration)
JW-1324K/1342K 	Input relay	00000 to 00017	Not connectable
		コ0000 to コ0001	
	Output relay	00400 to 00413 ※1	Not connectable
		コ0040 to コ0041 ※1	
JW-1424K/1442K 	Input relay	00000 to 00027	00030 to 00067
		コ0000 to コ0002	コ0003 to コ0006
	Output relay	00400 to 00417	00420 to 00457
		コ0040 to コ0041	コ0042 to コ0045
JW-1624K/1642K 	Input relay	00000 to 00043 ※2	00050 to 00107
		コ0000 to コ0004 ※2	コ0005 to コ0010
	Output relay	00400 to 00427	00430 to 00467
		コ0040 to コ0042	コ0043 to コ0046

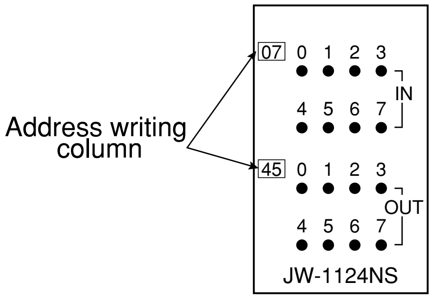
※ 1 In case of JW-1324K/1342K, output relays 00414 to 00417 are dummy areas (auxiliary relay).
※ 2 In case of JW-1624K/1642K, input relays 00044 to 00047 are dummy areas (auxiliary relay).

■ Allocation example of relay number

	Configuration of module	Input/output relay address (upper section: relay number, lower section: byte address)			
			Basic module	Expansion module (1)	Expansion module (2)
Example 1	JW-1424K JW-114S  Basic module Expansion module (1)	Input relay	00000 to 00027		
			コ0000 to コ0002		
		Output relay	00400 to 00417	00420 to 00437	
			コ0040 to コ0041	コ0042 to コ0043	
Example 2	JW-1624K JW-112N JW-1124NS  Basic module Expansion module (1) Expansion module (2)	Input relay	00000 to 00043	00050 to 00067	00070 to 00077
			コ0000 to コ0004	コ0005 to コ0006	コ0007
		Output relay	00400 to 00427		00430 to 00437
			コ0040 to コ0042		コ0043
Example 3	JW-1624K JW-1324NS JW-1124NS  Basic module Expansion module (1) Expansion module (2)	Input relay	00000 to 00043	00050 to 00067	00070 to 00077
			コ0000 to コ0004	コ0005 to コ0006	コ0007
		Output relay	00400 to 00427	00430 to 00447	00450 to 00457
			コ0040 to コ0042	コ0043 to コ0044	コ0045
Example 4	JW-1624K JW-1324NS JW-1324NS  Basic module Expansion module (1) Expansion module (2)	Input relay	00000 to 00043	00050 to 00067	00070 to 00107
			コ0000 to コ0004	コ0005 to コ0006	コ0007 to コ0010
		Output relay	00400 to 00427	00430 to 00447	00450 to 00467
			コ0040 to コ0042	コ0043 to コ0044	コ0045 to コ0046

Note 1: Second and third digit figures of input and output relays can be marked using oil-based ink on the address writing column of the expansion module.

Note 2: Analog input module (JW-14AD) and analog output module (JW-12DA) are not occupied input/output relays.



Chapter 11. ROM Operation

11-1 ROM operation

- Basic modules JW-1424K, JW-1442K, JW-1624K, and 1642K are available with ROM operation. (JW-1324K/1342K are not available ROM operation.)
- The ROM operation is a method of operating the JW10 by the content of the ROM, by storing the system memory or user program in the ROM (EPROM, EEPROM), and transferring from ROM to RAM.
- Saving of a program and more in the ROM is available; as the contents of the ROM do not disappear.
- ROM operation is convenient for changing a program without using any support tool. (Such as in a case where there is no special maintenance man in local facility.)

[1] Types of ROM

- Types of ROMs available the ROM operation are as follows (ROMs must be prepared by users.)

ROM type	Conditions of ROM	Recommended maker model
EPROM	27C512 Access time : Less than 200 ns Package : 28P DIP	NM27C512Q (NS) M27C512-20F1 (SGS-THOMSON)
EEPROM	28C256 Access time : Less than 200 ns Package : 28P DIP (Having 64 byte page write function.)	AT28C256 (ATMEL) HN58C256P-20 (Hitachi)

[2] Registrable contents to ROM

- Areas that can be registered ROM is decided by setting value of system memory #255.

#255	Registrable contents to ROM
44(H)	Program memory, system memory (#200 to #377)
45(H)	Program memory, system memory (#200 to #377), data memory (39000 to 39777)

- Whenever turning ON the power, the JW10 enters operation mode.
- When the power is ON, the JW10 latches area starting from latched relay top address set by system memory (#230, #231).

[3] Data transfer from ROM to RAM and from RAM to ROM

- Data transfer from ROM to RAM is carried out when the power is input or by using a support tool.

Value of #255 in ROM	ROM type	Data transfer from ROM to RAM	
		When the power is input	Operation by support tool
44(H), 45(H)	EPROM	Execute	Available
	EEPROM	Execute	Available
00(H)	EPROM	Not execute	Available
	EEPROM	Not execute	Available
Other than 00(H), 44(H), 45(H)	EPROM	Not execute	Not available
	EEPROM	Not execute	Not available

- During transferring data from ROM to RAM, the JW10 executes sum check and verifies contents of ROM and RAM.
- The JW10 can transfer data from RAM to ROM in all areas to be used as ROM (same as the case of #255 = 45(H)) using a support tool only if an EEPROM is used.
- If an EPROM is used, the following two methods are available for data transfer.
  ① Transfer programs etc. of the JW10 into a support tool such as multipurpose programmer JW-50PG. Then write them into the EPROM using a PROM writer's transfer function. (Details are on the next page.)
  ② Transfer programs etc. of the JW10 into the EEPROM (data transfer from RAM to ROM). Then copy contents of the EEPROM into the EPROM using a PROM writer.

11-2 Writing into ROM

[1] Procedure when using EPROM

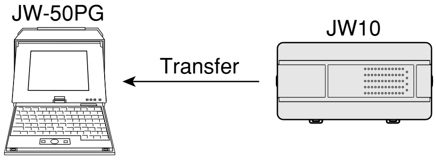
- Use the multipurpose programmer JW-50PG, ladder software JW-50SP/52SP/92SP, and PROM programmer to write program memory, system memory, and data memory into the EPROM.

■ Recommended PROM writers

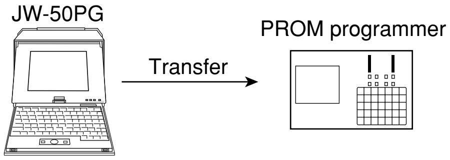
Manufacturer	Model
Ando Electric Corp.	AF-9703/9704
Minato Electronics Corp.	MODEL-1866A/1890A
Advantest Corp.	TR4943/4944A

- This section shows the writing procedure using the multipurpose programmer JW-50PG.
- For detailed operations of the multipurpose programmer and PROM programmer, see the instructions for their use.

1 Use the multipurpose programmer to create programs, system memory, and data memory to be contained in the ROM. In case of storing the programs and other data resident in the JW10 in the ROM, transfer them to the multipurpose programmer.



2 Set the system memory #255 at 44(H) or 45(H) using multipurpose programmer.



3 Transfer the programs etc. from the multipurpose programmer to the PROM programmer.

4 Set the EPROM (27C512) onto the PROM programmer, and write the programs.

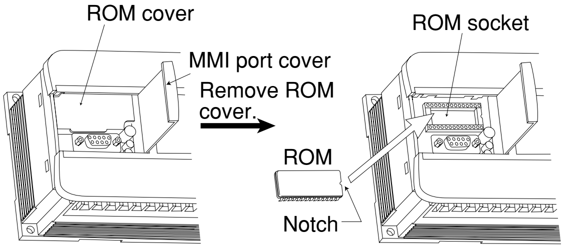
[2] Procedure when using EEPROM

- Use the multipurpose programmer JW-50PG, ladder software JW-50SP/52SP/92SP, and hand-held programmer JW-13PG to write program memory, system memory and data memory into the EEPROM.
- For detailed operations of hand-held programmer and other device, see their instruction manuals for their use.

1 Power OFF the JW10.

↓

2 Get the EPROM (28C256) seated onto the ROM socket.



- Set the direction (notch) of the ROM the same as that of the ROM socket.
- Use the dedicated tool when inserting/removing the ROM.

↓

3 Power ON the JW10.

↓

4 Use the JW-13PG etc. to create programs, system memory, and data memory in the RAM of JW10.

↓

5 Set the system memory #255 at 44(H) or 45(H).

↓

6 Operate the JW13PG etc. to write the contents of the RAM into the EEPROM.

- Use stop mode (or program mode) when writing into the EEPROM.

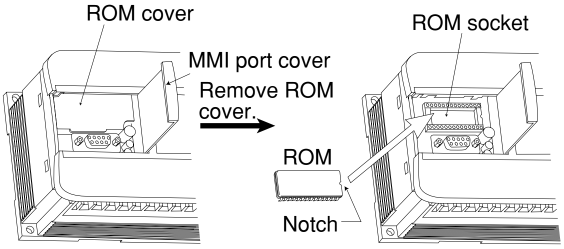
<Procedure for writing with JW-13PG>

クリア CLR → 編集 EDIT → A 0 → A 0 → B 1 (Write) → SET 8
→ A 0 (Interrupt)

11-3 ROM operation procedure

1 Power OFF the JW10.

2 Get the EPROM/EEPROM containing programs etc. seated onto the ROM socket.



- Set the direction (notch) of the ROM the same as of the ROM socket.
- Use the dedicated tool when inserting/removing the ROM.

3 Power ON the JW10.

- Data are transferred from the ROM to the RAM is conducted, and the JW10 will operate based on the contents of the ROM.

Note

Be careful that if any EPROM or EEPROM in use contain old programs written in the system memory #255 = 44(H) or 45(H), powering OFF followed by powering ON will rewrite the contents of the RAM completely to the old programs in the EPROM/EEPROM.

Chapter 12. How to Use the High-Speed Counter

12-1 High-speed counter

- The JW10 high-speed counter has two modes for different methods of counting signals.

Mode	Function
Mode 1	Single-phase rising pulse signal: 2 points Maximum frequency: 10 kHz Counter range: 0 to 65535 (16-bit binary counter) If the count value becomes equal to the comparison value, the interrupt program (LB170/LB171) will be executed.
Mode 2	90-degree phase difference two-phase signal: 1 point Maximum frequency: 10 kHz Counter range: 0 to 65535 (16-bit binary counter) If the count value becomes equal to the comparison value, the interrupt program (LB172) will be performed.

- The mode is selected by the value stored in system memory #203.

Value of #203	Contents
00(H)	High-speed counter is not used (00000 to 00003 are normal input)
01(H)	Mode 1 (Single-phase rising pulse signal: 2 points)
02(H)	Mode 2 (90-degree phase difference two-phase signal: 1 point)

- When the high-speed counter is enabled, use input terminals 0 - 3 for the high-speed counter.

Mode		Signal name	Input terminal
Mode 1	CH1	Input pulse signal	0
		Preset signal	1
	CH2	Input pulse signal	2
		Preset signal	3
Mode 2		Phase B input pulse signal	0
		Phase A input pulse signal	1
		Phase Z (marker) signal	2
		HLS (home position limit switch) signal	3

- When the high-speed counter is enabled, use the following special relays, special registers and labels.

		Mode 1		Mode 2
		CH1	CH2	
Current count value (16 bits binary)	Lower bits	コ0740	コ0750	コ0760
	Upper bits	コ0741	コ0751	コ0761
Count comparison value (16 bits binary)	Lower bits	コ0742	コ0752	コ0762
	Upper bits	コ0743	コ0753	コ0763
Preset value (16 bits binary)	Lower bits	コ0744	コ0754	コ0764
	Upper bits	コ0745	コ0755	コ0765
Count enable relay		07320	07324	07364
Preset relay		07321	07325	07365
Preset release relay		07322	07326	07366
Preset status relay		07323	07327	07367
Interrupt label		LB170	LB171	LB172

12-2 Mode 1 (single-phase rising pulse input)

- Mode 1 counts single-phase rising pulses and it has two channel counters (CH1, CH2).
- The maximum frequency is 10 kHz, and the counter range is 0 to 65535 (16-bit binary counter).
- Select mode 1 by setting system memory address #203 to 01(H).

[1] CH1 operation

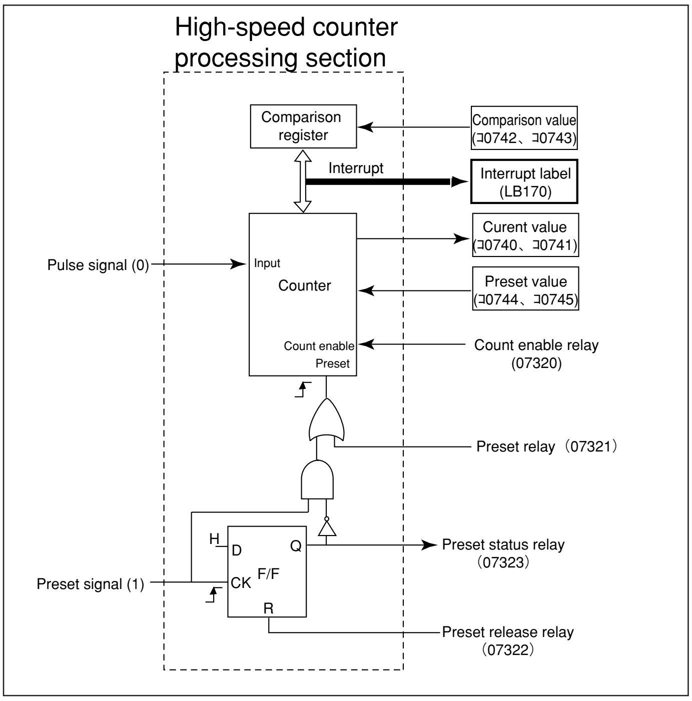
- Input the pulse signal to input terminal 0.
- Counting continues while the count enable relay (07320) is ON.
- The count value is stored in the current count value register (コ0740, コ0741).
- When the count value becomes equal to the comparison value (コ0742, コ0743), subroutine LB170 will be executed.
- The counter can be preset by an external signal or an internal relay.

① Preset by an external signal (preset signal)

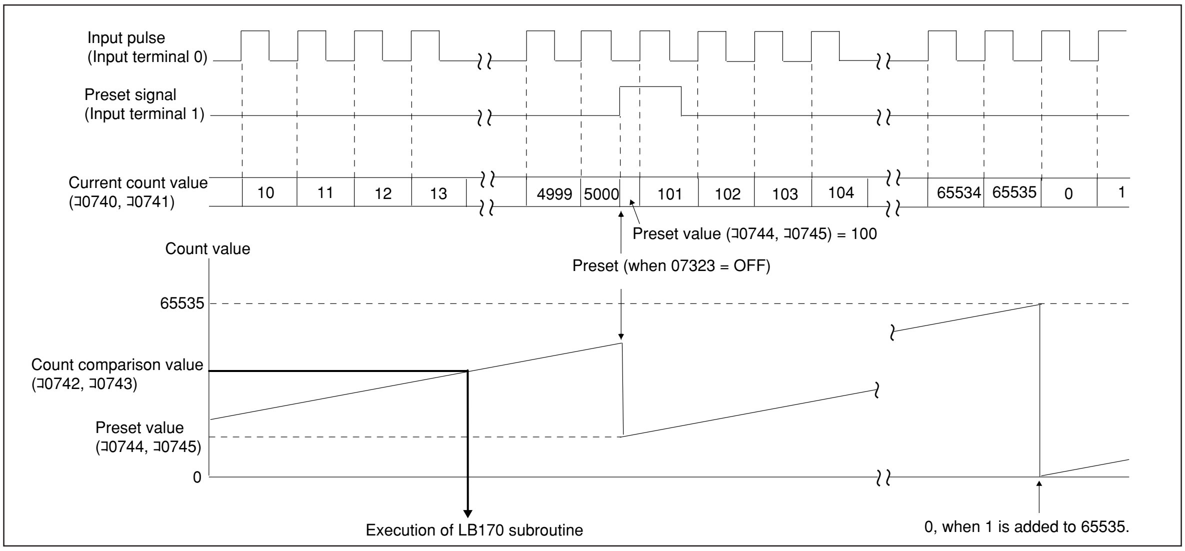
- The preset signal is detected at input terminal 1.
- When the preset status relay (07323) is OFF, and if the preset signal turns ON from OFF, the current count value will be set to the preset value (コ0744, コ0745).
- When preset is executed, the preset status relay (07323) will turn ON. When the preset release relay (07322) is turned ON, the preset status relay (07323) will turn OFF.

② Preset by an internal relay (preset relay: 07321)

- When the preset relay (07321) turns ON from OFF, the current count value will be set to the preset value (コ0744, コ0745).


High-speed counter processing section
Comparison register
Comparison value (コ0742、コ0743)
Interrupt
Interrupt label (LB170)
Curent value (コ0740、コ0741)
Pulse signal (0)
Input
Counter
Preset value (コ0744、コ0745)
Count enable
Preset
Count enable relay (07320)
Preset relay（07321）
H
D
Q
F/F
CK
R
Preset signal (1)
Preset status relay（07323）
Preset release relay（07322）

- Every time a scan cycle is executed, the count enable relay (07320), preset relay (07321), preset release relay (07322), preset status relay (07323), current value (コ0740, コ0741), comparison value (コ0742, コ0743), and preset value (コ0744, コ0745) will be refreshed before the next round of input/output processing.



[2] CH2 operation

- The pulse signal is detected on input terminal 2.
- Counting continues while the count enable relay (07324) is ON.
- The count value is stored in the current count value register (コ0750, コ0751).
- When the count value becomes equal to the comparison value (コ0752, コ0753), subroutine LB171 will be executed.
- The counter can be preset by an external signal or an internal relay.

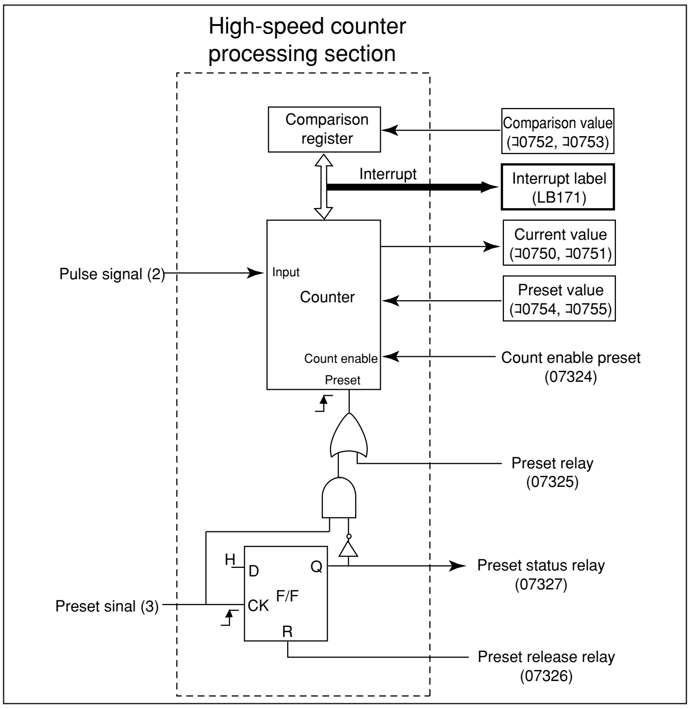
① **Preset by an external signal (preset signal)**

- The preset signal is detected at input terminal 3.
- When the preset status relay (07327) is OFF, and if the preset signal turns ON from OFF, the current count value will be set to the preset value (コ0754, コ0755).
- When preset is executed, the preset status relay (07327) will turn ON. When the preset release relay (07326) is turned ON, the preset status relay (07327) will turn OFF.

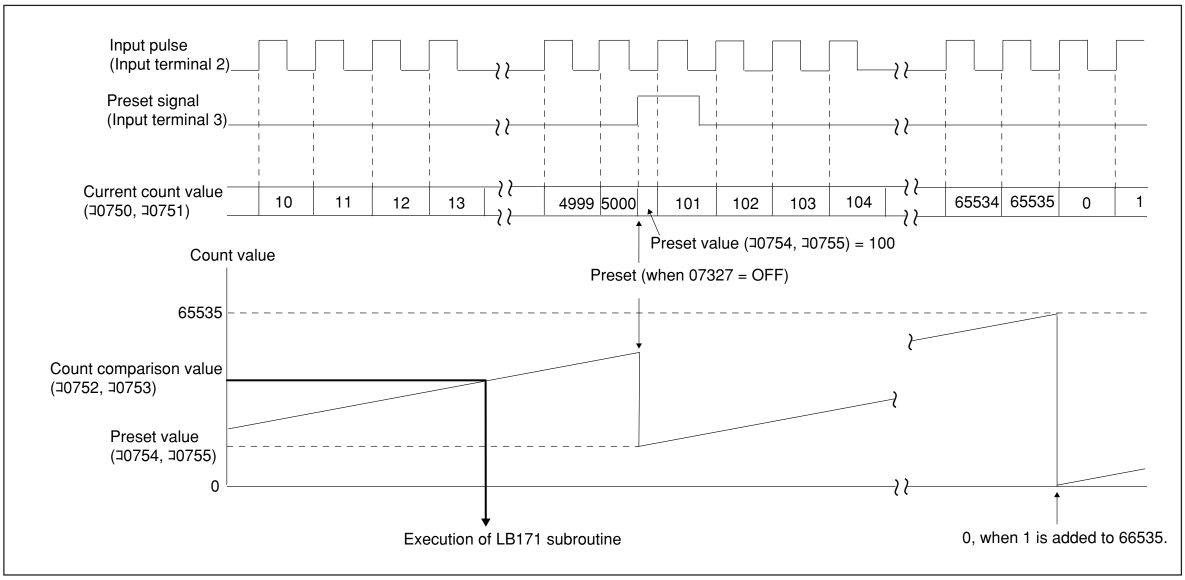
② **Preset by an internal relay (preset relay: 07325)**

- When the preset relay (07325) turns ON from OFF, the current count value will be set to the preset value (コ0754, コ0755).

- Every time a scan cycle is executed, the count enable relay (07324), preset relay (07325), preset release relay (07326), preset status relay (07327), current value (コ0750, コ0751), comparison value (コ0752, コ0753), and preset value (コ0754, コ0755) will be refreshed before the next round of input/output processing.

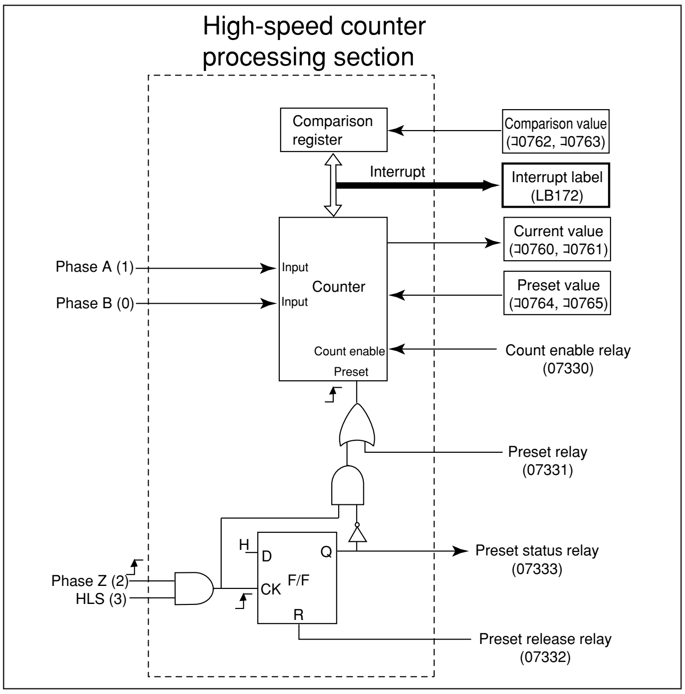


High-speed counter processing section
Comparison register
Comparison value (コ0752, コ0753)
Interrupt
Interrupt label (LB171)
Current value (コ0750, コ0751)
Pulse signal (2)
Input
Counter
Preset value (コ0754, コ0755)
Count enable
Preset
Count enable preset (07324)
Preset relay (07325)
H
D
Q
F/F
CK
R
Preset status relay (07327)
Preset sinal (3)
Preset release relay (07326)



12-3 Mode 2 (90-degree phase difference two-phase signal input)

- Mode 2 counts 90-degree phase difference two-phase signals and it has one channel.
- The maximum frequency is 10 kHz, and the counter range is 0 to 65535 (16-bit binary counter).
- Select mode 2 setting system memory address #203 to 02(H).
- Input phase A, phase B, phase Z, HLS (home position limit switch) on input terminal 1, input terminal 0, input terminal 2, and input terminal 3, respectively.
- Counting continues while the count enable relay (07330) is ON.
- The count value is stored in the current count value register (コ0760, コ0761).
- When the count value becomes equal to the comparison value (コ0762, コ0763), subroutine LB172 will be executed.
- The counter can be preset by an external signal or an internal relay.


High-speed counter processing section
Comparison register
Comparison value (コ0762, コ0763)
Interrupt
Interrupt label (LB172)
Current value (コ0760, コ0761)
Phase A (1)
Phase B (0)
Input
Input
Counter
Preset value (コ0764, コ0765)
Count enable
Preset
Count enable relay (07330)
Preset relay (07331)
H
D
Q
Preset status relay (07333)
Phase Z (2)
HLS (3)
CK
F/F
R
Preset release relay (07332)

① **Preset by an external signal (Z-phase signal or HLS signal)**

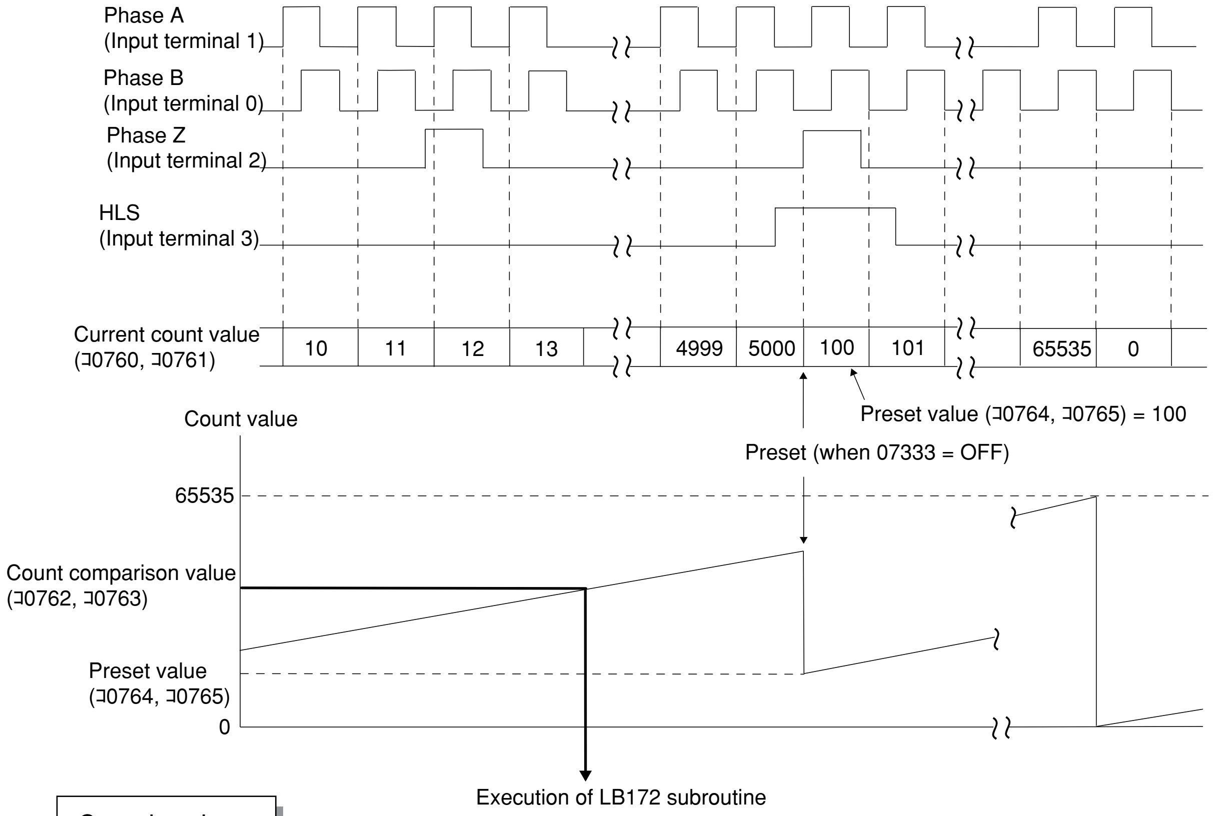
- When the preset status relay (07333) is OFF and the HLS signal is ON, if the Z-phase signal turns ON from OFF, the current count value will be set to the preset value (コ0764, コ0765).
- When preset is executed, the preset status relay (07333) will turn ON. When the preset release relay (07332) is turned ON, the preset status relay (07333) will turn OFF.

② **Preset by an internal relay (preset relay: 07331)**

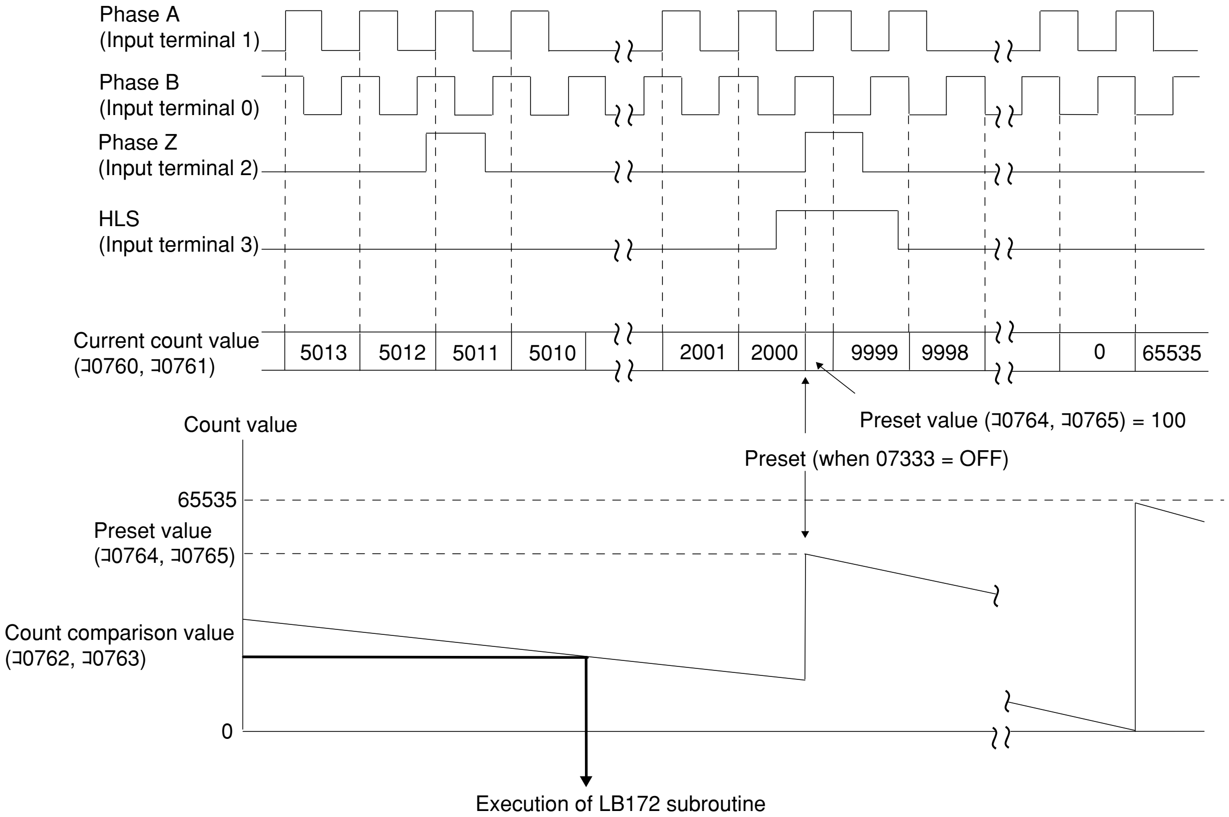
- When the preset relay (07331) turns ON from OFF, the current count value will be set to the preset value (コ0764, コ0765).

- Every time a scan cycle is executed, the count enable relay (07330), preset relay (07331), preset release relay (07332), preset status relay (07333), current value (コ0760, コ0761), comparison value (コ0762, コ0763), and preset value (コ0764, コ0765) will be refreshed before the next round of input/output processing.

Counting up



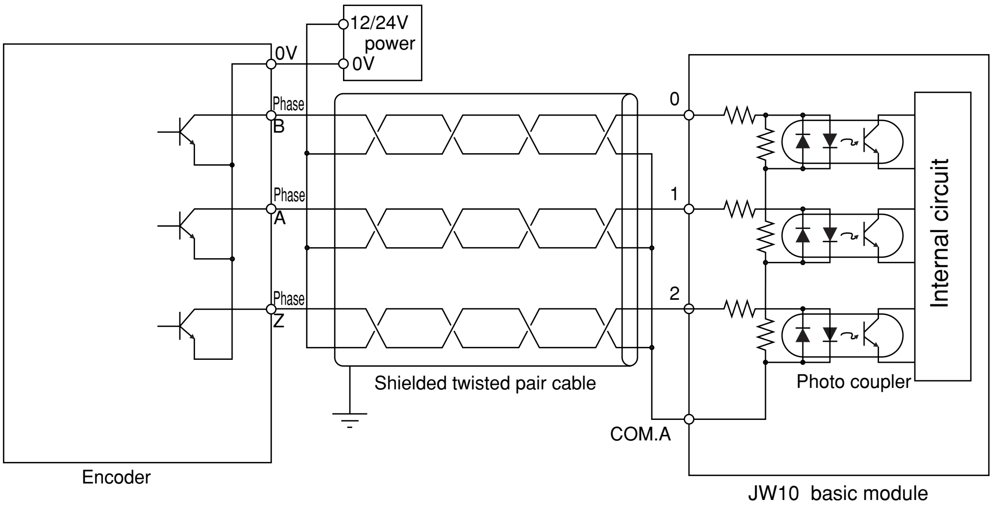
Counting down



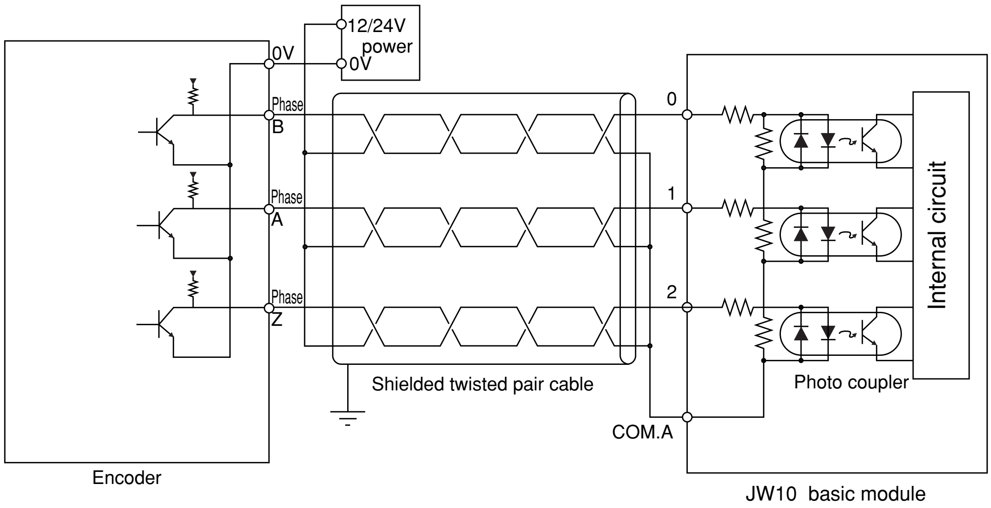
Encoder connection examples

- The high-speed counter can be connected to open-collector outputs or voltage-output encoders.
- The high-speed counter cannot be connected to differential output encoders.

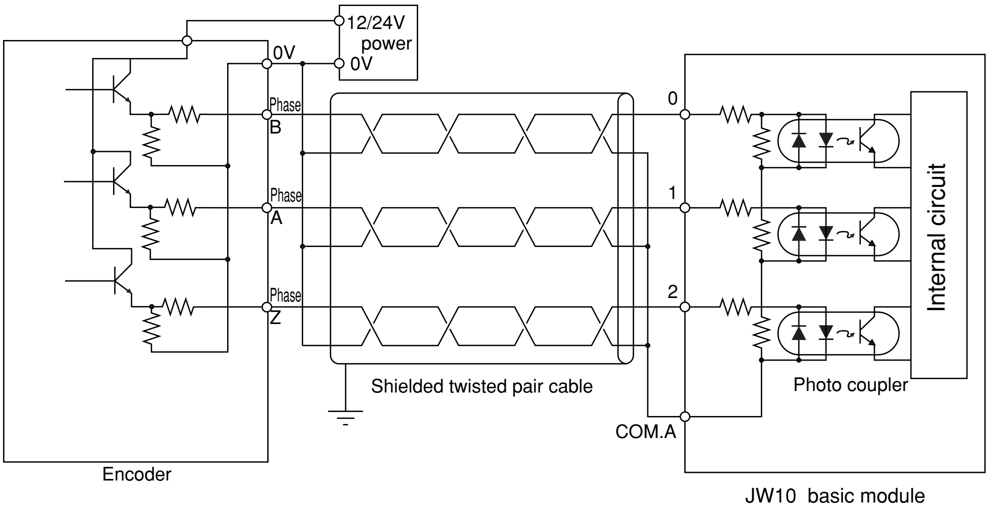
Example 1: An open collector type encoder



Example 2: A voltage output type encoder (load sinking)



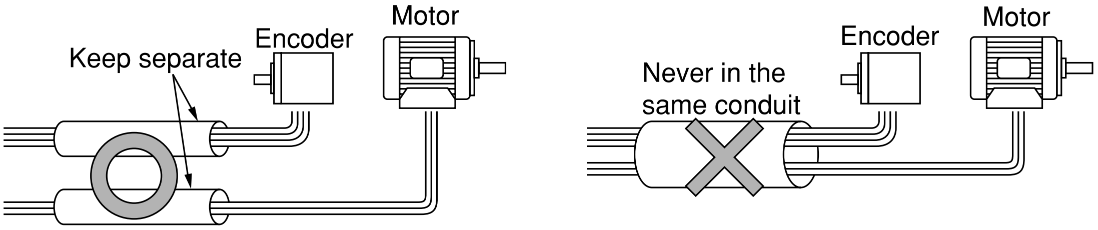
Example 3: A voltage output type encoder (load sourcing)



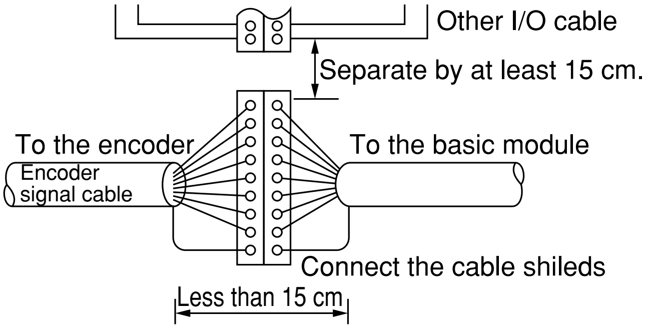
Note

★ If noise interferes with the encoder signals, miscounts are likely and the current counter value may be wrong. Pay special attention when wiring.

- Never run the motor output cable and the encoder cable in the same conduit.



- Use separate conduits to run the motor output and encoder cables from the connection box in the machine to the encoder. Never bundle them.
- Do not run the motor output cable and the encoder cable in parallel, even inside the control panel. They must never be in the same conduit.
- Do not run the encoder cable near device that generates noise. Never run the encoder cable in parallel with the wires of such device.
- Connect the encoder cable directly to the terminal block in the main module's control panel. If at all possible, do not connect it to the terminal strip in the control panel. If it must be connected to a terminal connection strip, keep the length of the exposed, unshielded wires as short as possible. Be sure to connect the shield from the cables on both sides. Keep this connection terminal away from any other connection terminals that generate noise.

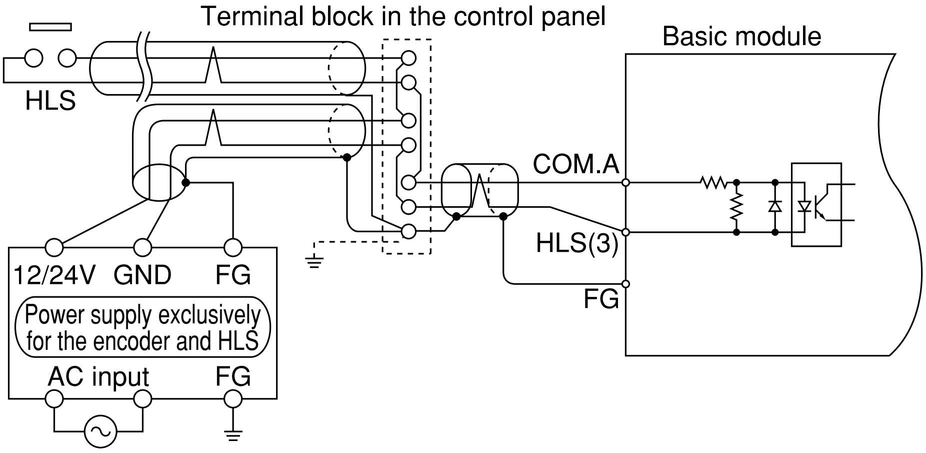


- Do not use the encoder's DC power supply for any other device (excepting HLS). Do not connect the encoder's DC power supply GND to the GND of any other DC power supply.
- Connect the encoder cable shield to the FG terminals in the basic module or to the ground terminals in the control panel.

Note

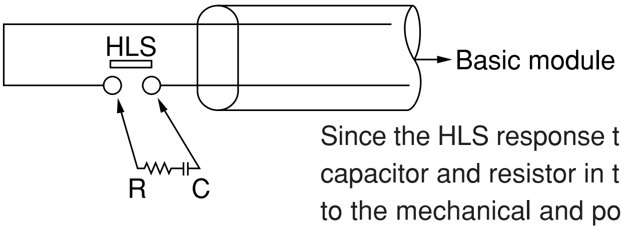
★ Do not make the wires leading to the HLS (home limit switch) unnecessarily long. If they become too long, use relays.

- In order to improve resistance to noise, use shielded twisted pair cables.
- Even when shielded twisted pair cables are used, do not run them in the same conduit as the motor output cable or any other PC control cables. Do not run them in parallel with other wiring.


Terminal block in the control panel
Basic module
HLS
COM.A
HLS(3)
FG
12/24V
GND
FG
Power supply exclusively for the encoder and HLS
AC input
FG

- If an external power is used for the HLS input, install a connection terminal block in the control panel. Keep the connection terminal block away from the connection terminal block for PC I/O control wires.
- Install a surge absorber for the HLS contacts in parallel.

(Since the origin signal (HLS) is installed on the machine side, the wiring may be quite long. In addition to that, the contacts of the origin signal are open when the encoder is not at the origin, so that the cable may function as an antenna and it may easily pick up electrical noise.)



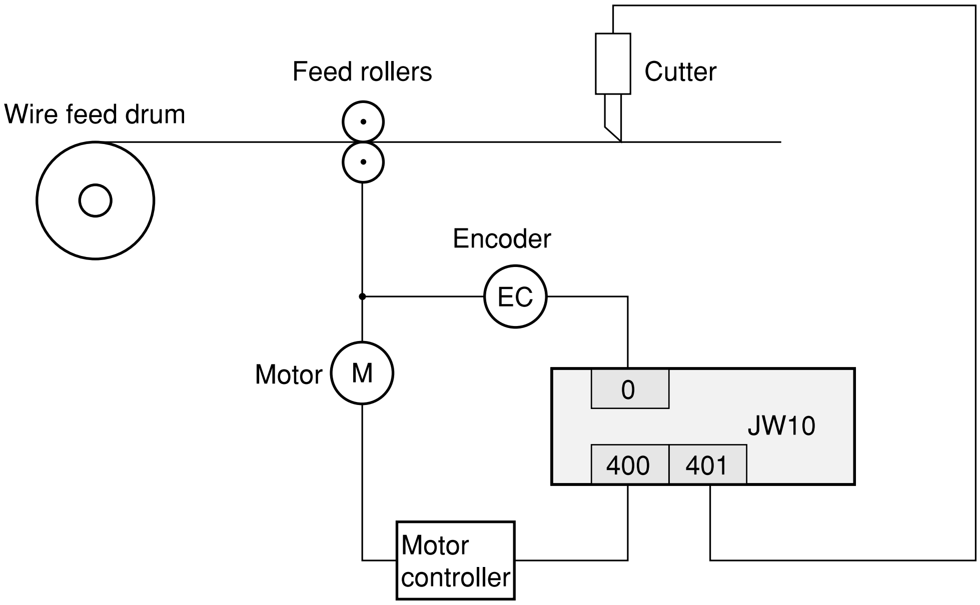
Since the HLS response time will be lengthened due to the capacitor and resistor in the surge absorber, pay attention to the mechanical and positional relationship between the HLS and phase Z.

- Connect the HLS signal cable shield to the basic module's FG terminals or to the control panel's ground terminals.

12-4 Application examples

[1] An example of use on a wire cutter

(1) System configuration

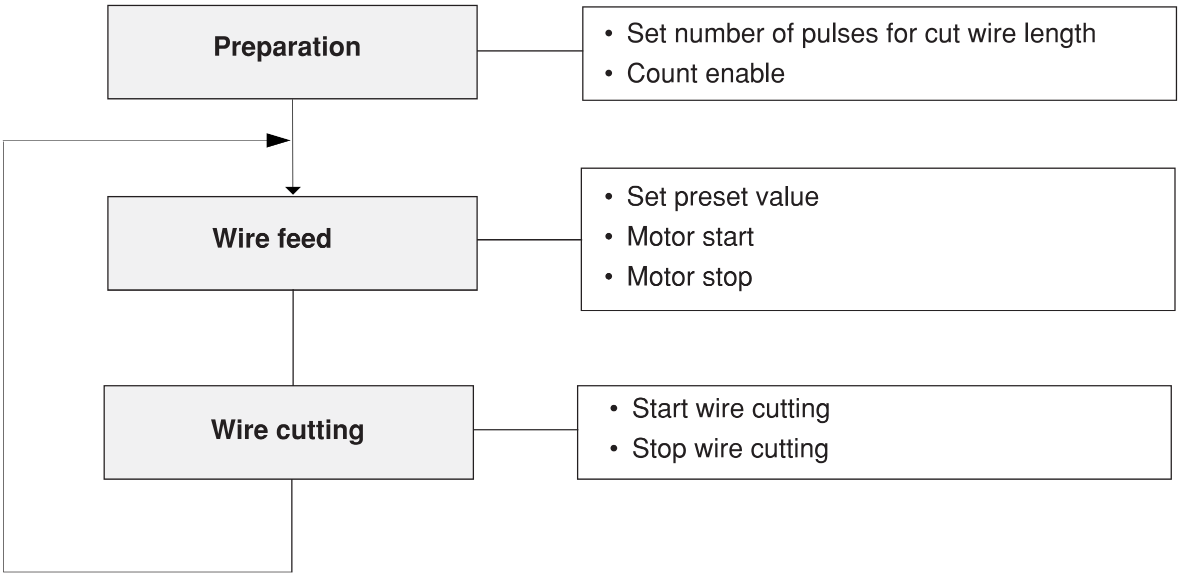


- The amount of wire unwound is determined by the pulses from the encoder installed on the feed roller, which are captured by the JW10 high-speed counter.
- When the JW10 determines that the number of pulses counted equal the preset length, it sends instructions to cut to the cutter.

(2) System memory setting

#203 = 01(H) ... Mode 1 (single-phase rising pulse input)

(3) Operation flow

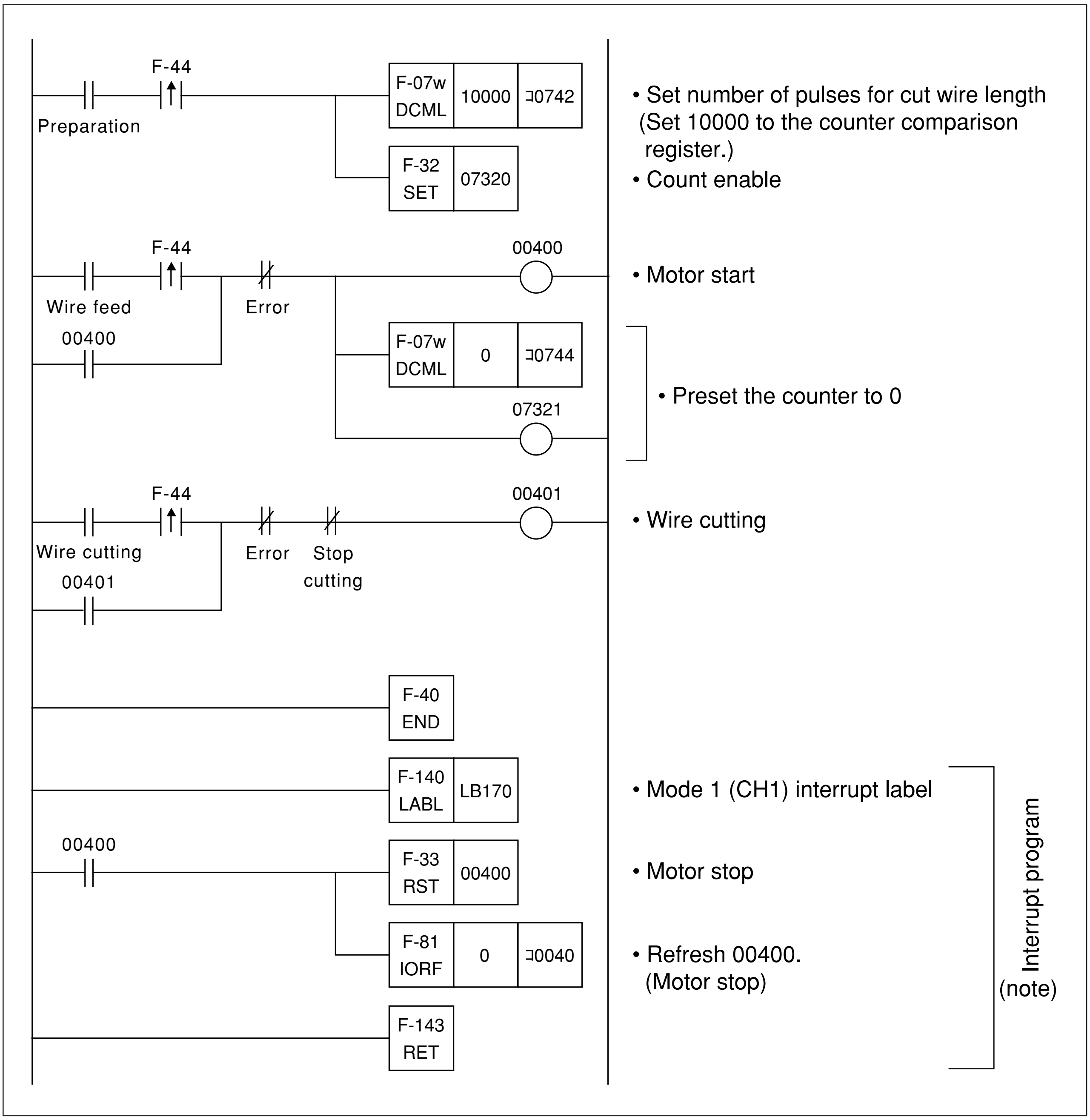


- Set number of pulses for cut wire length
- Count enable

- Set preset value
- Motor start
- Motor stop

- Start wire cutting
- Stop wire cutting

(4) Program

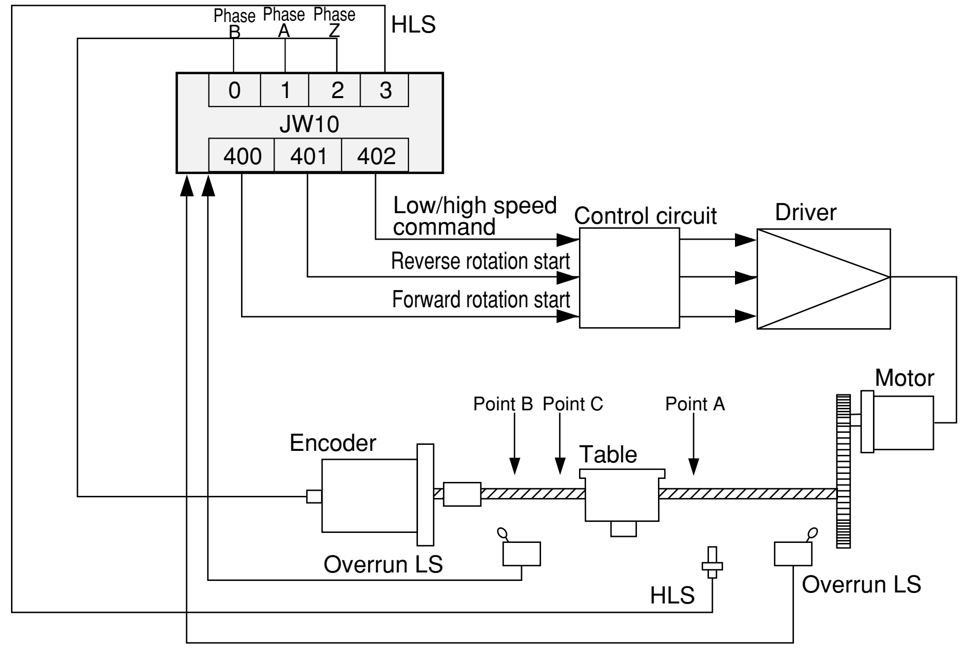


- Set number of pulses for cut wire length (Set 10000 to the counter comparison register.)
- Count enable
- Motor start
- Preset the counter to 0
- Wire cutting
- Mode 1 (CH1) interrupt label
- Motor stop
- Refresh 00400. (Motor stop)

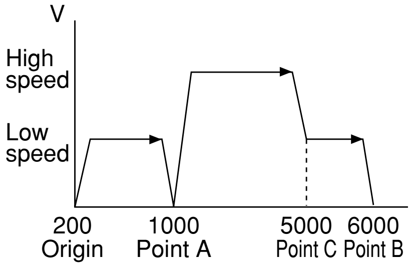
Note: The use of an interrupt program allows the motor to be stopped more quickly.

[2] An example of 2-point positioning of a single-axis table

(1) System configuration



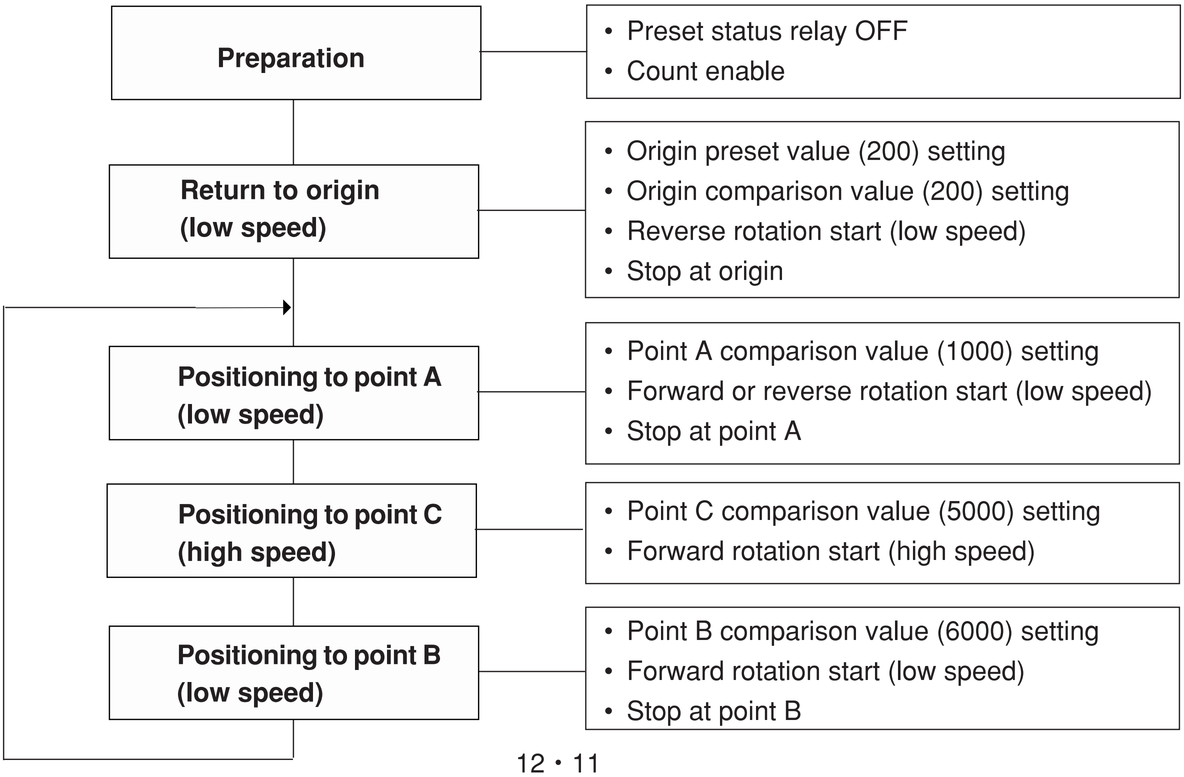
- Position on points A and B.
- The table moves to point A at low speed. When it moves to point B, it first moves to point C at high speed and then on to point B at low speed.
- The table moves to the home position in reverse at low speed.



(2) System memory setting

#203 = 02(H) ... Mode 2 (90-degree phase difference two-phase signal input)

(3) Operation flow



Preparation

- Preset status relay OFF
- Count enable

Return to origin (low speed)

- Origin preset value (200) setting
- Origin comparison value (200) setting
- Reverse rotation start (low speed)
- Stop at origin

Positioning to point A (low speed)

- Point A comparison value (1000) setting
- Forward or reverse rotation start (low speed)
- Stop at point A

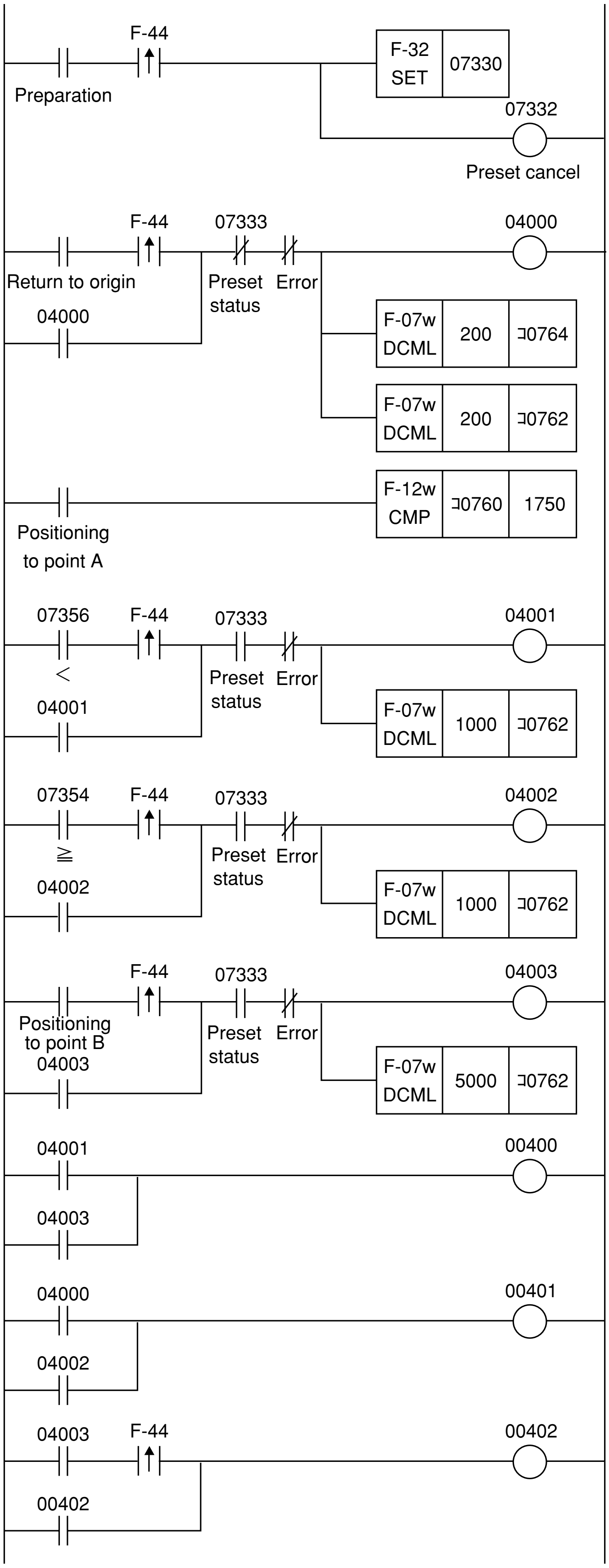
Positioning to point C (high speed)

- Point C comparison value (5000) setting
- Forward rotation start (high speed)

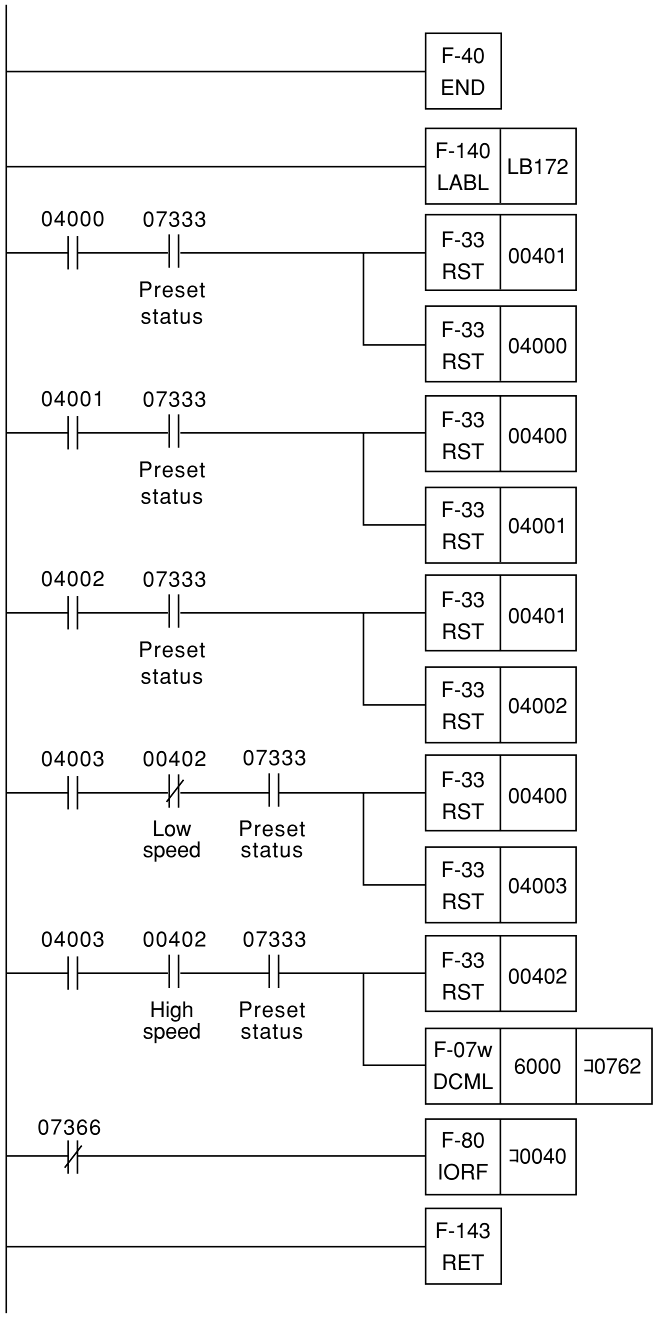
Positioning to point B (low speed)

- Point B comparison value (6000) setting
- Forward rotation start (low speed)
- Stop at point B

(4) Program



- Count enable
- Preset status relay (07333) OFF
- Auxiliary relay for starting reverse rotation to the origin
- Origin preset value setting
- Origin comparison value setting
- Direction check of point A [1750(8) = 1000(D)] (A < 1000 or A ≧ 1000)
- Auxiliary relay for starting forward rotation to point A
- Point A comparison value setting
- Auxiliary relay for starting reverse rotation to point A
- Point A comparison value setting
- Auxiliary relay for starting forward rotation to point B
- Point C comparison value setting
- Forward rotation start
- Reverse roation start
- High speed command



- Mode 2 interrupt label
- Reverse rotation stop
- Start reverse rotation to the origin
- Forward rotation stop
- Reset auxiliary relay for starting forward rotation to point A
- Reverse rotation stop
- Reset auxiliary relay for starting reverse roation to point A
- Forward rotation stop
- Reset auxiliary relay for starting reverse rotation to point B
- Low speed command
- Point B comparison value setting
- I/O refresh

Interrupt program

Chapter 13. How to Use Communication Port

13-1 Communication port

[1] Function of communication port

- The JW10 has a communication port for using any of computer link, data link or remote I/O.
- Each function is determined by values set on system memory (#234).

#234	Communication mode
00(H)	Computer link mode
01(H)	Data link mode
02(H)	Remote I/O mode

(1) Computer link mode

- The JW10 can communicate to a host computer or an LCD control terminal.
  (The JW10 can receive host computer's commands and response to the host computer.)
- The JW10 can communicate even in stop mode or error mode.
  (Except communication functional failure.)

(2) Data link mode

- The JW10 (master) can communicate to the JW10 (salve). (Programless communication)
- Even one slave station of the JW10 is not connected, or is in stop mode or error, a master station of the JW10 communicates with other normal slave stations of JW10.

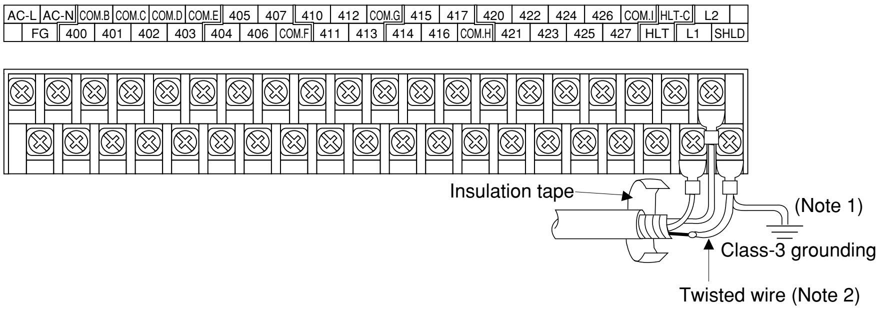
(3) Remote I/O mode

- The JW10 (master) can communicate to JW10 (slave) as remote I/O.
- If one slave station of JW10 is not connected or is error, the JW10 master station stops operation and communication with all the other slave stations.

[2] Wiring

- Communication port of the JW10 is L1, L2, and SHLD terminals of the basic module.

(Lower terminals of JW-1624K)



- See "6-1 Precautions for wiring."
- Make sure to use our recommended shielded twist pair wires for communication lines.
  Recommended cable: S-IREV-SB 2 × 0.5 (Hitachi Cable Ltd.)

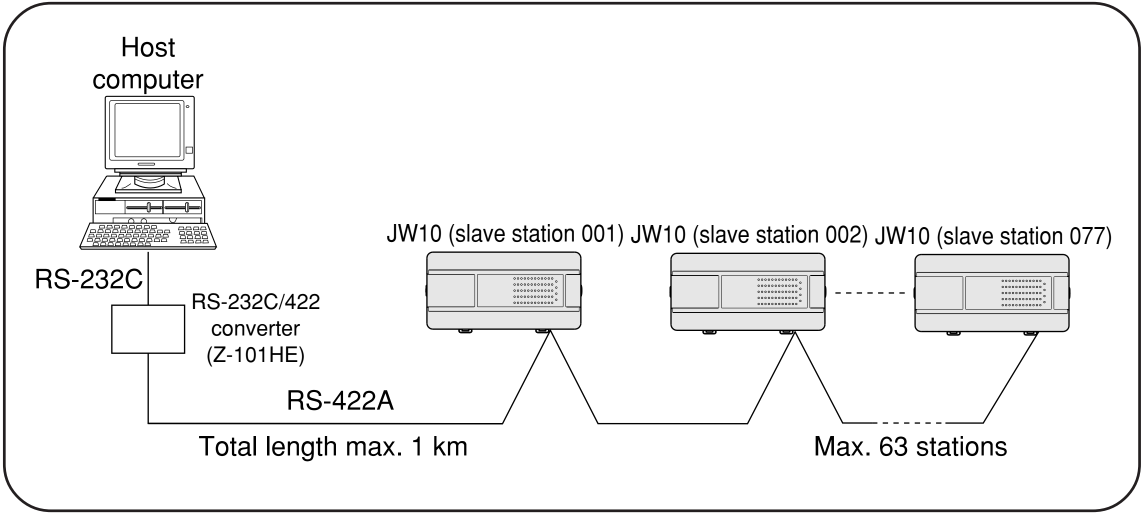
Note 1: FG terminal and SHLD terminal are not connected inside.

Note 2: Connecting in relay shield of shield cables externally using 0.5 mm^2 sectional area or so twisted wires (shorter than 30 mm) offers easy wiring to terminals.

13-2 Computer link

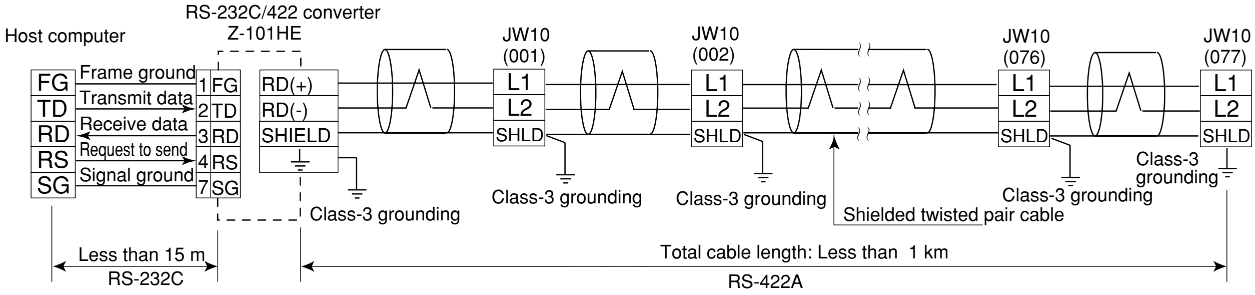
[1] Communication specifications

- A host computer can communicate with maximum 63 modules of JW10.
- With this communication, you can monitor operation, collect data, instruction operation, and change setting value of the connected JW10 modules in personal computer or the like.



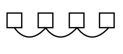
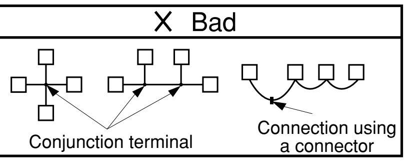
Item	Specifications
Data transfer standard	EIA RS-422A or equivalent, Start-stop sychronous system
Transfer rate	38400, 19200, 9600, 4800, 2400, 1200, 600, 300 bits/s
Data style	Start bit : 1 bit Data length : 7, 8 bits Parity bit : 1 bit (odd, even, none) Stop bit : 1, 2 bits
Character used	ASCII, alphanumeric characters
Error check	Parity check, sum check
No. of stations connected	Max. 63 sets
Communication line	Shielded twisted pair cable. Cable total length: 1 km (party line connection) 2-wire system

[2] Wiring



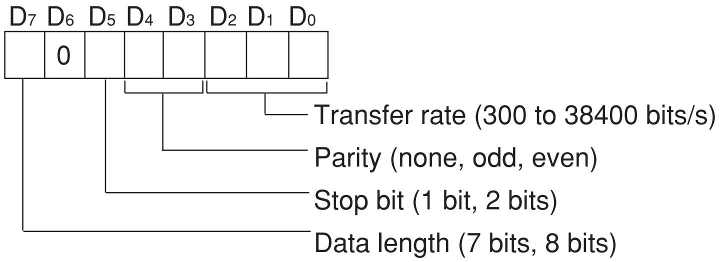
Note 1: Do class-3 grounding for SHLD terminal. The operation without class-3 grounding may cause malfunction due to noise.

Note 2: Never branch a communication cable into more than 3 lines.

○ Good	X Bad
	

[3] Setting system memory

- When using computer link mode, set system memory as follows.

#234	Communication mode	00(H): Computer link mode
#236	Transfer specifications	 D7 D6 D5 D4 D3 D2 D1 D0 0 Transfer rate (300 to 38400 bits/s) Parity (none, odd, even) Stop bit (1 bit, 2 bits) Data length (7 bits, 8 bits)
#237	Local number	001 to 077(8)

D7	Data length
0	7 bits
1	8 bits

D5	Stop bit
0	1 bit
1	2 bits

D4	D3	Parity
0	0	None
0	1	Odd
1	0	Even
1	1	Disable

D2	D1	D0	Transfer rate (bit/s)
1	1	1	38400
0	0	0	19200
0	0	1	9600
0	1	0	4800
0	1	1	2400
1	0	0	1200
1	0	1	600
1	1	0	300

Note 1: Communication mode and station number are set on system memory. We recommend to mark these settings on a label and stick it on each case for easily readable the set contents.

[Example of label]

Computer link #001

[4] Communication format

- When the JW10 receives "command" from a host computer, the JW10 executes processing in accordance with its command contents and sends "response" to the host computer.
- If an error occurs in the processing contents, the JW10 sends "error response."

① Command (host computer to JW10)

|◄── Sum check area ──►|

· ·	· ·	A D (H)	A D (L)	?	R I	Command content	S C (H)	S C (L)	C R

② Response (JW10 to host computer)

|◄── Sum check area ──►|

· ·	· ·	A D (H)	A D (L)	#	R I	Response content	S C (H)	S C (L)	C R

Error response where an error occurs

|◄Sum check area►|

· ·	· ·	A D (H)	A D (L)	%	R I	E C (H)	E C (L)	S C (H)	S C (L)	C R

(1) Identification symbol

ASCII character	ASCII code	Contents
:	3A(H)	Header (indicates start of command or response)
?	3F(H)	Indicates “command”
#	23(H)	Indicates “response” at normal operation
%	25(H)	Indicates “error response”
CR	0D(H)	Termination symbol (indicates termination of command or response)

(2) AD(H), AD(L): Slave station number — ASCII character 00 to 77(8)

- Inside "command," set ID number of a JW10 slave station attempting to control with 00 to 77(8).
- Inside "response," set ID number 01 to 77(8) of a JW10 slave station which is attempted to transmit data to the host computer.
- When slave station number is assigned as "00" on writing command such as SRR (set and reset of relays) or WRG (writing to register), all the stations can be simultaneously controlled. This slave station number "00" is called "**global address**."
- For global address usable commands, see "kinds of command" on page 13・7.
- A "command" having global address is not given "response."
- If global address is assigned to a command which is not available global address, the JW10 does nothing. (It does not return response, either.)

(3) RI: Response time — ASCII character 0 to F(H)

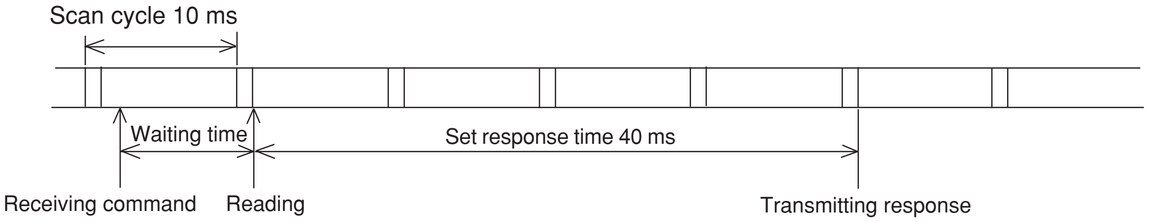
- Set interval from processing of "command" contents from a host computer to transmitting "response."
- Set this time in accordance with processing performance of the host computer.

RI(H)	Response time (ms)	RI(H)	Response time (ms)	RI(H)	Response time (ms)	RI(H)	Response time (ms)
0	0	4	40	8	80	C	300
1	10	5	50	9	90	D	400
2	20	6	60	A	100	E	500
3	30	7	70	B	200	F	600

- Response time in actual use is total of the above setting time and waiting time of PC scan cycle. Ex.: Monitoring relay 02000 of PC01 (in case where scan cycle is 10 ms and response time is 40 ms)

Command

:	:	0	1	?	4	M	R	L	0	2	0	0	0	4	F	CR



Note 1: For actual design, you may face the problem of how long a response time should be set. It is not as simply as using an optimum setting time which varies with type of the host computer, kind of programming language, and programming method. Therefore, we recommend to first set a rather long response time. Then, shorten it gradually by testing.

Note 2: Number of processing bytes is restricted with some commands. Communication buffer is also restricted with the type of host computer used. Be careful for number of processing bytes.

(4) Sum check codes SC(H), SC(L)

- The communication port detects error using sum check as well as parity check in order to increase the reliability.
- The contents of the sum check are as follows:
  ① Add data from slave station No. (AD(H)) to the last data of the command contents or response contents (just before sum check code) that are summed up in ASCII code.
  ② Convert the sum check code (2 digits hex.) to 8 bits data and add ① to the result.
  When the grand total is "00(H)" (disregarded figure up), the message is regarded as correct, when the grand total is not "00(H)," the message is regarded as an error.

Ex.: In case of setting relay 07030 of PC02 (0 as reset, 1 as set).

Command which should be sent to a host computer (in case of response time 50 ms)

```
0 2   ?  5  S R R 0 7 0 3 0 1
───      ─  ─────────────────
 │       │      └── Command contents
 │       └───────── Response time (50 ms)
 └───────────────── Address (slave station number)
```

Add from top address (slave station number) to end of command contents at their ASCII code format, and take complement of this total. This is sum check code.

		ASCII code
0	···	30(H)
2	···	32
?	···	3F
5	···	35
S	···	53
R	···	52
R	···	52
0	···	30
7	···	37
0	···	30
3	···	33
0	···	30
+) 1	···	31
		2F8(H)
		↓
		F8(H)
Complement of 2		↓
		08(H)

Complement of 2

Complement of 2 can be obtained by reversing all of data expressed by binary (0 to 1, 1 to 0) and incrementing by one.

F8(H)→11111000
↓ Reversing bit
00000111
↓ Add by one
00001000 → 08(H)

Therefore, "command" will be as follows:

```
: : 0 2 ? 5 S R R 0 7 0 3 0 1 0 8 CR
                              ───
                               └ Sum check code
```

When this "command" is received, PC02 sends the following "response."

```
: : 0 2 # 5 S R R 0 7 0 3 0 5 5 CR
                            ───
                             │
                      Sum check code
```

		ASCII code
0	···	30(H)
2	···	32
#	···	23
5	···	35
S	···	53
R	···	52
R	···	52
0	···	30
7	···	37
0	···	30
3	···	33
+) 0	···	30
		2AB(H)
+) Sum check code		55(H)
		300(H)

After receiving this response, the host computer adds from the top of slave station number to the last of the response contents in their ASCII code format, and adds sum check code "55." If lower 2 digits of the total become "00(H)" (300(H)), the JW10 judges that this response is correct.

- If sum check is not needed, set two @ (at sing : 40(H)) on SC(H) and SC(L) positions on the command line. The JW10 does not execute sum check.
  Even if this is set, the JW10 adds sum check code on response. You may make program to ignore these codes at the host computer.

(5) EC(H), EC(L): Error code

- When an error occurs, the JW10 sends the following codes as error response.

Error code (EC(H), EC(L))	Contents
01	Format error.
02	Designated address is not TMR/CNT setting value.
05	Number of transfer bytes is not correct.
06	JW10 does not stop by HLT (stop PC processing).
07	Writing to JW10 memory is not executed correctly.
0A	Parity error.
0B	Framing error.
0C	Overrun error.
0D	Sum check error.
0E	Prohibit program memory write.
10	Not match write mode.
11	Not program area.
12	Tried to write in ROM.
1B	System memory error.
30	Password is not registered.
31	Secret function is not released.
32	Wrong password is input.
33	Password error (attempted to register characters other than alphabetical and numeric figures).

Note 1: In any of the following cases, the JW10 will do nothing and transmit no response.

1) Assigned slave station number in the command and own station number do not match.
2) If the JW10 cannot find " : " " ? " or "$^{C}_{R}$" in the command.

In order to recover from any of these errors, create a program to check by time out with the host computer.

(6) Contents of command and response

- See next item "[5] Description of command."

[5] Description of command

(1) Kinds of command

- Commands are largely classified into read-out command, write command, and control command.

	Function	Command name	Global address	See page
Read-out command	Relay monitor	MRL	×	13•9
	Current value monitor of timer/counter	MTC	×	13•10
	Register current value monitor	MRG	×	13•11
	Read out program memory	RPM	×	13•13
	Read out system memory	RSM	×	13•15
	Read out date	MDY	×	13•16
	Read out time	MTM	×	13•17
Write command	Set/reset relay	SRR	○	13•9
	Set/reset timer/counter	SRT	○	13•11
	Write in register	WRG	○	13•12
	Write the same data in register	FRG	○	13•12
	Write in program memory	WPM	○	13•13
	Write in system memory	WSM	○	13•15
	Change setting value of timer/counter	CTC	○	13•14
	Set date	SDY	○	13•16
	Set time	STM	○	13•17
	Correct, run, and stop clock	ACL	○	13•18
Control command	Monitor operation conditions	MPC	×	13•18
	Stop PC operation	HLT	○	13•19
	Restart PC operation	RUN	○	13•19
	Read out write mode status	SWE	×	13•20
	Set write mode	EWR	○	13•20
	Turn back the message	TST	×	13•21
	Release secret/password registration	PAS	○	13•21
	Set secret function	SES	○	13•22
	Check secret function	SEI	×	13•22

Note: For details of "global address" see page 13-4

(2) Write mode

- Write mode of JW10 is "mode 0" (write prohibited for all memories) at power ON. Prior to writing program from the host computer, change write mode to "mode 1" or "mode 2" using EWR command (setting of write mode). SWE command read out current status of writing mode.
- Set write mode to "mode 0" as much as possible, except when writing data into JW10.
- Each mode has restrictions as follow:

Mode 0	Write prohibited for all memories
Mode 1	Write enable only for data memory
Mode 2	Write enable for all memories

(3) Address expression system

- In each command, the setting value in the following table is set in the address module of communication format.

		Address (octal)	Setting value (octal)	Using command
Relay number		00000 to 15777	00000 to 15777	MRL, SRR
Timer/counter point of contact number		T000 to T377	T0000 to T0377	MRL
		C000 to C377		
Timer/counter number		000 to 377	0000 to 0377	MTC, SRT
Register address		コ0000 to コ1577	A0000 to A1577	MRG, WRG, FRG
		b0000 to b0777	B0000 to B0777	
		09000 to 09777	09000 to 09777	
		19000 to 19777	19000 to 19777	
		29000 to 29777	29000 to 29777	
		39000 to 39777	39000 to 39777	
Program address	JW-1324K/1342K	0000 to 2777	000000 to 002777	RPM, WPM, CTC
	JW-1424K/1442K JW-1624K/1642K	0000 to 7777	000000 to 007777	
System memory address		#000 to #377	0000 to 0377	RSM, WSM

Note: Set with octal notation regardless of setting value on system memory #115.

(4) Data expression system

- Data are expressed by hexadecimal. Program contents are also expressed by hexadecimal based on machine language.
- Contents of bit configuration of program memory is not available.

[6] Description of each command

M R L	Relay monitor
Function	Monitor ON/OFF state of the designated relay.
Communication format Command	: : AD(H) AD(L) ? RI M R L Relay number SC(H) SC(L) CR
Communication format Response	: : AD(H) AD(L) # RI M R L Relay number Data SC(H) SC(L) CR
Relay number	00000 to 15777 T0000 to T0377 (Octal)
Data	1 : ON 0 : OFF
Execution condition	Write mode: Mode 0, Mode 1, Mode 2 PC operation state: Stop, Run
Example for use	Monitor auxiliary relay 04033 of PC01. (Response time: 40 ms) Command $::\underbrace{0\ 1}_{\text{Station number}}\ ?\ \underbrace{4}_{\text{Response time}}\ M\ R\ L\ \underbrace{0\ 4\ 0\ 3\ 3}_{\text{Relay number}}\ \underbrace{4\ 7}_{\text{Sum check code}}\ {}^{C}_{R}$ Response $::\underbrace{0\ 1}_{\text{Station number}}\ \#\ \underbrace{4}_{\text{Response time}}\ M\ R\ L\ \underbrace{0\ 4\ 0\ 3\ 3}_{\text{Relay number}}\ \underbrace{1}_{\text{ON}}\ \underbrace{3\ 2}_{\text{Sum check code}}\ {}^{C}_{R}$

S R R	Set/reset relay
Function	Set/reset the relay.
Communication format Command	: : AD(H) AD(L) ? RI S R R Relay number Data SC(H) SC(L) CR
Communication format Response	: : AD(H) AD(L) # RI S R R Relay number SC(H) SC(L) CR
Relay number	00000 to 15777 (Octal)
Data	1 : Set 0 : Reset
Execution condition	Write mode: Mode 1, Mode 2 PC operation state: Stop, Run
Example for use	Reset auxiliary relay 07001 of PC03. (Response time: 40 ms) Command $::\underbrace{0\ 3}_{\text{Station number}}\ ?\ \underbrace{4}_{\text{Response time}}\ S\ R\ R\ \underbrace{0\ 7\ 0\ 0\ 1}_{\text{Relay number}}\ \underbrace{0}_{\text{Reset}}\ \underbrace{0\ B}_{\text{Sum check code}}\ {}^{C}_{R}$ Response $::\underbrace{0\ 3}_{\text{Station number}}\ \#\ \underbrace{4}_{\text{Response time}}\ S\ R\ R\ \underbrace{0\ 7\ 0\ 0\ 1}_{\text{Relay number}}\ \underbrace{5\ 7}_{\text{Sum check code}}\ {}^{C}_{R}$

Note 1: Relays which are used for input relay, special relay, special register and link system (data link, remote I/O) cannot be set/reset.

M T C — Current value monitor of timer/counter

Function	Read out current value of TMR/CNT number 1 to 2. Sequential read out current value of timer/counter up to 256.

Communication format

Command

:	:	A D (H)	A D (L)	?	R I	M	T	C	TMR/CNT number 1	TMR/CNT number 2	S C (H)	S C (L)	C R

Response

:	:	A D (H)	A D (L)	#	R I	M	T	C	TMR/CNT number 1	TMR/CNT number 2	Data 1	· · · ·	Data n	Attributed data 1	. .	Attributed data n	S C (H)	S C (L)	C R

TMR/CNT number	0000 to 0377 (Octal)	
Data	4 characters. See note 1	n: Max.256
Attributed data	2 characters, 00: PC program is not in use 02: CNT 04: TMR	
Execution condition	Write mode	Mode 0, Mode 1, Mode 2
	PC operation state	Stop, Run

Example for use

Read out TMR/CNT current value 000 to 002 of PC 03. (Response time: 10 ms)

Command

$$::\underbrace{0\ 3}_{\text{Station number}}\ ?\ \underbrace{1}_{\text{Response time}}\ M\ T\ C\ \underbrace{0\ 0\ 0\ 0}_{\text{TMR/CNT number 1}}\ \underbrace{0\ 0\ 0\ 2}_{\text{TMR/CNT number 2}}\ \underbrace{C\ 7}_{\text{Sum check code}}\ \begin{smallmatrix}C\\R\end{smallmatrix}$$

Response

$$::\underbrace{0\ 3}_{\text{Station number}}\ \#\ \underbrace{1}_{\text{Response time}}\ M\ T\ C\ \underbrace{0\ 0\ 0\ 0}_{\text{TMR/CNT number 1}}\ \underbrace{0\ 0\ 0\ 2}_{\text{TMR/CNT number 2}}\ \underbrace{6\ 5\ 1\ 8}_{\text{TMR000 current value 1865}}$$

$$\underbrace{3\ 2\ 0\ 0}_{\substack{\text{CNT 001}\\\text{current value 0032}}}\ \underbrace{1\ 4\ 1\ 3}_{\substack{\text{CNT 002}\\\text{current value 1314}}}\ \underbrace{0\ 4}_{\text{TMR}}\ \underbrace{0\ 2}_{\text{CNT}}\ \underbrace{0\ 2}_{\text{CNT}}\ \underbrace{5\ 9}_{\text{Sum check code}}\ \begin{smallmatrix}C\\R\end{smallmatrix}$$

Note 1: Current value data of timer/counter is configured with two bytes as shown below:

	2nd byte								1st byte							
	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
TMR000 to TMR277	0	0	0	10^2	10^1				10^0				10^{-1}			
				1	8	4	2	1	8	4	2	1	8	4	2	1
TMR300 to TMR377	0	0	0	10^1	10^0				10^{-1}				10^{-2}			
				1	8	4	2	1	8	4	2	1	8	4	2	1
CNT000 to CNT377	0	0	0	10^3	10^2				10^1				10^0			
				1	8	4	2	1	8	4	2	1	8	4	2	1

SRT Set/reset timer/counter

Function	Set timer/counter (time-up, count-up) or reset (return to setting value).
Communication format — Command	: : AD(H) AD(L) ? RI S R T TMR/CNT number Data SC(H) SC(L) CR
Communication format — Response	: : AD(H) AD(L) # RI S R T TMR/CNT number SC(H) SC(L) CR
TMR/CNT number	0000 to 0377 (Octal)
Data	1: set 0: reset
Execution condition	Write mode: Mode 1, Mode 2 PC operation state: Run
Example for use	Set TMR 002 of PC06. (Response time: 70 ms) Command : : 0 6 ? 7 S R T 0 0 0 2 1 3 8 CR (0 6: Station number; 7: Response time; 0 0 0 2: TMR number; 1: Set; 3 8: Sum check code) Response : : 0 6 # 7 S R T 0 0 0 2 8 5 CR (0 6: Station number; 7: Response time; 0 0 0 2: TMR number; 8 5: Sum check code)

MRG Register current value monitor

Function	Read out current value of register address 1 to 2. Sequential read out up to 512 bytes.
Communication format — Command	: : AD(H) AD(L) ? RI M R G Register address 1 Register address 2 SC(H) SC(L) CR
Communication format — Response	: : AD(H) AD(L) # RI M R G Register address 1 Register address 2 Data 1 .... Data n SC(H) SC(L) CR
Register address (Note1)	A0000 to A1577, 09000 to 09777, 29000 to 29777 B0000 to B0777, 19000 to 19777, 39000 to 39777 (Octal)
Data	2 characters (hexadecimal) n: Max. 512
Execution condition	Write mode: Mode 1, Mode 2, Mode 3 PC operation state: Stop, Run
Example for use	Read the data (hexadecimal) 09000 to 09003 of PC06. (Response time : 100ms) Command : : 0 6 ? A M R G 0 9 0 0 0 0 9 0 0 3 3 F CR (0 6: Station number; A: Response time; 0 9 0 0 0: Register address 1; 0 9 0 0 3: Register address 2; 3 F: Sum check code) Response : : 0 6 # A M R G 0 9 0 0 0 0 9 0 0 3 (0 6: Station number; A: Response time; 0 9 0 0 0: Register address 1; 0 9 0 0 3: Register address 2) 0 0 4 F 3 2 0 1 E B CR (0 0: 09000; 4 F: 09001; 3 2: 09002; 0 1: 09003; E B: Sum check code)

Note 1: A0000 to A1577 and B0000 to B0777 indicate ⊐0000 to ⊐1577 and b0000 to b0777, repectively.

W R G — Write in register

Function	Write required data from the register address 1 to 2. Sequentially writable up to 512 bytes.	
Register address (note1)	A0000 to A1577, 09000 to09777, 29000 to 29777 B0000 to B0777, 19000 to 19777, 39000 to 39777 (Octal)	
Data	2 characters (hexadecimal) n: Max. 512	
Execution condition	Write mode	Mode 1, Mode 2
	PC operation state	Stop, Run

Communication format

Command	:	:	AD (H)	AD (L)	?	RI	W	R	G	Register address 1	Register address 2	Data 1	· · · ·	Data n	SC (H)	SC (L)	CR
Response	:	:	AD (H)	AD (L)	#	RI	W	R	G	Register address 1	Register address 2	SC (H)	SC (L)	CR			

Example for use

Write 14, 00, 32, 56 (hexadecimal) in ᆿ0400 to ᆿ0403 of PC30, respectively.
(Response time: 30 ms)

Command

$$: \; : \; \underbrace{3\;0}_{\text{Station number}} \; ? \; \underbrace{3}_{\text{Response time}} \; W\;R\;G\; \underbrace{A\;0\;4\;0\;0}_{\text{Register address 1}} \; \underbrace{A\;0\;4\;0\;3}_{\text{Register address 2}}$$

$$\underbrace{1\;4}_{\text{ᆿ0400}} \; \underbrace{0\;0}_{\text{ᆿ0401}} \; \underbrace{3\;2}_{\text{ᆿ0402}} \; \underbrace{5\;6}_{\text{ᆿ0403}} \; \underbrace{9\;9}_{\text{Sum check code}} \; {}^{C}_{R}$$

Response

$$: \; : \; \underbrace{3\;0}_{\text{Station number}} \; \# \; \underbrace{3}_{\text{Response time}} \; W\;R\;G\; \underbrace{A\;0\;4\;0\;0}_{\text{Register address 1}} \; \underbrace{A\;0\;4\;0\;3}_{\text{Register address 2}} \; \underbrace{4\;A}_{\text{Sum check code}} \; {}^{C}_{R}$$

Note 1: A0000 to A1577 and B0000 to B0777 indicate ᆿ0000 to ᆿ1577 and b0000 to b0777, repectively.

F R G — Write the same data in register

Function	Write the same data in the register address 1 to 2. Sequential write up to 512 bytes.	
Register address (note1)	A0000 to A1577, 09000 to 09777, 29000 to 29777 B0000 to B0777, 19000 to 19777, 39000 to 39777	
Data	2 characters (hexadecimal)	
Execution condition	Write mode	Mode 1, Mode 2
	PC operation state	Stop, Run

Communication format

Command	:	:	AD (H)	AD (L)	?	RI	F	R	G	Register address 1	Register address 2	Data	SC (H)	SC (L)	CR
Response	:	:	AD (H)	AD (L)	#	RI	F	R	G	Register address 1	Register address 2	SC (H)	SC (L)	CR	

Example for use

Write data 40 (hexadecimal) in register 09000 to 09077 of PC11. (Response time: 100 ms)

Command

$$: \; : \; \underbrace{1\;1}_{\text{Station number}} \; ? \; \underbrace{A}_{\text{Response time}} \; F\;R\;G\; \underbrace{0\;9\;0\;0\;0}_{\text{Register address 1}} \; \underbrace{0\;9\;0\;7\;7}_{\text{Register address 2}} \; \underbrace{4\;0}_{\text{Data}} \; \underbrace{D\;B}_{\text{Sum check code}} \; {}^{C}_{R}$$

Response

$$: \; : \; \underbrace{1\;1}_{\text{Station number}} \; \# \; \underbrace{A}_{\text{Response time}} \; F\;R\;G\; \underbrace{0\;9\;0\;0\;0}_{\text{Register address 1}} \; \underbrace{0\;9\;0\;7\;7}_{\text{Register address 2}} \; \underbrace{5\;B}_{\text{Sum check code}} \; {}^{C}_{R}$$

Note 1: A0000 to A1577 and B0000 to B0777 indicate ᆿ0000 to ᆿ1577 and b0000 to b0777, repectively.

R P M	Read out program memory

Function		Read out contents of the program address 1 to 2. This function can read sequentially up to 256 steps. It reads instruction as machine language, not a format of "STRXXXXX."
Communication format	Command	: | : | A D (H) | A D (L) | ? | R I | R | P | M | Program address 1 | Program address 2 | S C (H) | S C (L) | C R
	Response	: | : | A D (H) | A D (L) | # | R I | R | P | M | Program address 1 | Program address 2 | Command 1 | · · · · | Command n | S C (H) | S C (L) | C R
Program address		000000 to 002777 [JW-1324K/1342K] 000000 to 007777 [JW-1424K/1442K, JW-1624K/1642K] (Octal)
Data		4 characters (hexadecimal) n: Max. 256
Execution condition	Write mode	Mode 0, Mode 1, Mode 2
	PC operation state	Stop, Run

Example for use

Read address 000000 to 000002 contents of PC01. (Response time: 10 ms)

Command

$$: \; : \; \underbrace{0\ 1}_{\text{Station number}}\ ?\ \underbrace{1}_{\text{Response time}}\ R\ P\ M\ \underbrace{0\ 0\ 0\ 0\ 0\ 0}_{\text{Program address 1}}\ \underbrace{0\ 0\ 0\ 0\ 0\ 2}_{\text{Program address 2}}\ \underbrace{F\ E}_{\text{Sum check code}}\ {}^{C}_{R}$$

Response

$$: \; : \; \underbrace{0\ 1}_{\text{Station number}}\ \#\ \underbrace{1}_{\text{Response time}}\ R\ P\ M\ \underbrace{0\ 0\ 0\ 0\ 0\ 0}_{\text{Program address 1}}\ \underbrace{0\ 0\ 0\ 0\ 0\ 2}_{\text{Program address 2}}$$

$$\underbrace{0\ 0\ 8\ 0}_{\text{000000 contents of address}}\ \underbrace{0\ 0\ 9\ 1}_{\text{000001 contents of address}}\ \underbrace{0\ 8\ B\ 8}_{\text{000002 contents of address}}\ \underbrace{A\ 6}_{\text{Sum check code}}\ {}^{C}_{R}$$

Note 1: When you store program memory which is read by this command, be sure to store system memory with RSM command as well.

W P M	Write in program memory

Function		Write instruction in the program address 1 to 2. This function can write sequentially up to 256 steps. It writes instruction as machine language, not a format of "STRXXXXX."
Communication format	Command	: | : | A D (H) | A D (L) | ? | R I | W | P | M | Program address 1 | Program address 2 | Command 1 | · · · · | Command n | S C (H) | S C (L) | C R
	Response	: | : | A D (H) | A D (L) | # | R I | W | P | M | Program address 1 | Program address 2 | S C (H) | S C (L) | C R
Program address		000000 to 002777 [JW-1324K/1342K] 000000 to 007777 [JW-1424K/1442K, JW-1624K/1642K] (Octal)
Data		4 characters (hexadecimal) n: Max. 256
Execution condition	Write mode	Mode 2
	PC operation state	Stop by HLT command

Example for use

Write the below contents in address 000000 to 000002 of PC02. (Response time : 20 ms)

Command

$$: \; : \; \underbrace{0\ 2}_{\text{Station number}}\ ?\ \underbrace{2}_{\text{Response time}}\ W\ P\ M\ \underbrace{0\ 0\ 0\ 0\ 0\ 0}_{\text{Program address 1}}\ \underbrace{0\ 0\ 0\ 0\ 0\ 2}_{\text{Program address 2}}$$

$$\underbrace{0\ 0\ 8\ 0}_{\text{000000 contents of address}}\ \underbrace{0\ 0\ 9\ 1}_{\text{000001 contents of address}}\ \underbrace{0\ 8\ B\ 8}_{\text{000002 contents of address}}\ \underbrace{8\ 3}_{\text{Sum check code}}\ {}^{C}_{R}$$

Response

$$: \; : \; \underbrace{0\ 2}_{\text{Station number}}\ \#\ \underbrace{2}_{\text{Response time}}\ W\ P\ M\ \underbrace{0\ 0\ 0\ 0\ 0\ 0}_{\text{Program address 1}}\ \underbrace{0\ 0\ 0\ 0\ 0\ 2}_{\text{Program address 2}}\ \underbrace{1\ 3}_{\text{Sum check code}}\ {}^{C}_{R}$$

Note 1: Prior to loading the program which was stored by this command, be sure to load system memory contents.

CTC	Change setting value of timer/counter

Function		Change timer/counter setting value in the designated program address.
Communication format	Command	: : AD (H) AD (L) ? RI C T C Program address Setting value SC (H) SC (L) CR
	Response	: : AD (H) AD (L) # RI C T C Program address SC (H) SC (L) CR
Program address		000000 to 002777 [JW-1324K/1342K] 000000 to 007777 [JW-1424K/1442K, JW-1624K/1642K] (Octal)
Setting value		0000 to 1999 (BCD)
Execution condition	Write mode	Mode 2
	PC operation state	Stop, Run
Example for use		Change setting value of address 000024 to 0100 of PC04. (Response time: 20 ms) Command $: : \underbrace{0\ 4}_{\text{Station number}}\ ?\ \underbrace{2}_{\text{Response time}}\ C\ T\ C\ \underbrace{0\ 0\ 0\ 0\ 2\ 4}_{\text{Program address}}\ \underbrace{0\ 1\ 0\ 0}_{\text{Setting value}}\ \underbrace{6\ A}_{\text{Sum check code}}\ CR$ Response $: : \underbrace{0\ 4}_{\text{Station number}}\ \#\ \underbrace{2}_{\text{Response time}}\ C\ T\ C\ \underbrace{0\ 0\ 0\ 0\ 2\ 4}_{\text{Program address}}\ \underbrace{4\ 7}_{\text{Sum check code}}\ CR$

13

R S M — Read out system memory

Function	Read out contents of the system memory address 1 to 2. Sequential read up to 256 bytes.

Communication format																	
Command	:	:	A D (H)	A D (L)	?	R I	R	S	M	System memory addres 1	System memory addres 2	S C (H)	S C (L)	C R			
Response	:	:	A D (H)	A D (L)	#	R I	R	S	M	System memory addres 1	System memory addres 2	Data 1	....	Data n	S C (H)	S C (L)	C R

System memory address	0000 to 0377 (Octal)	
Data	2 characters (hexadecimal) n: Max. 512	
Execution condition	Write mode	Mode 0, Mode 1, Mode 2
	PC operation state	Stop, Run
Example for use	Read out data (hexadecimal) of system memory #201, #202 of PC10. (Response time : 20ms)	

Command $::\underbrace{1\ 0}_{\text{Station number}}\ ?\ \underbrace{2}_{\text{Response time}}\ R\ S\ M\ \underbrace{0\ 2\ 0\ 1}_{\text{System memory address 1}}\ \underbrace{0\ 2\ 0\ 2}_{\text{System memory address 2}}\ \underbrace{B\ 5}_{\text{Sum check code}}\ {}^{C}_{R}$

Response $::\underbrace{1\ 0}_{\text{Station number}}\ \#\ \underbrace{2}_{\text{Response time}}\ R\ S\ M\ \underbrace{0\ 2\ 0\ 1}_{\text{System memory address 1}}\ \underbrace{0\ 2\ 0\ 2}_{\text{System memory address 2}}\ \underbrace{0\ 1}_{\#201}\ \underbrace{0\ 0}_{\#202}\ \underbrace{D\ 1}_{\text{Sum check code}}\ {}^{C}_{R}$

W S M — Write in system memory

Function	Write data in the system memory address 1 to 2. Sequential write up to 256 bytes. (note 1)

Communication format																	
Command	:	:	A D (H)	A D (L)	#	R I	W	S	M	System memory addres 1	System memory addres 2	Data 1	....	Data n	S C (H)	S C (L)	C R
Response	:	:	A D (H)	A D (L)	#	R I	W	S	M	System memory addres 1	System memory addres 2	S C (H)	S C (L)	C R			

System memory address	0000 to 0377 (Octal)	
Data	2 characters (hexadecimal)	
Execution condition	Write mode	Mode 2
	PC operation state	Stop by HLT command
Example for use	Write 01, 01 (both hexadecimal) in system memory #201, #202 of PC22. (Response time: 10 ms)	

Command $::\underbrace{2\ 2}_{\text{Station number}}\ ?\ \underbrace{1}_{\text{Response time}}\ W\ S\ M\ \underbrace{0\ 2\ 0\ 1}_{\text{System memory address 1}}\ \underbrace{0\ 2\ 0\ 2}_{\text{System memory address 2}}\ \underbrace{0\ 1}_{\#0201}\ \underbrace{0\ 1}_{\#0202}\ \underbrace{E\ C}_{\text{Sum check code}}\ {}^{C}_{R}$

Response $::\underbrace{2\ 2}_{\text{Station number}}\ \#\ \underbrace{1}_{\text{Response time}}\ W\ S\ M\ \underbrace{0\ 2\ 0\ 1}_{\text{System memory address 1}}\ \underbrace{0\ 2\ 0\ 2}_{\text{System memory address 2}}\ \underbrace{C\ A}_{\text{Sum check code}}\ {}^{C}_{R}$

Note 1: Do not write in unreleased address in system memory address, or the PC may malfunction.

M D Y — Read out date

[JW-1424K, JW-1442K, JW-1624K ,and JW-1642K only]

Function: Read out date (year, month, date, and day) of clock.

Communication format

Command

:	:	A D (H)	A D (L)	?	R I	M	D	Y	S C (H)	S C (L)	C R

Response

:	:	A D (H)	A D (L)	#	R I	M	D	Y	Year (H)	Year (L)	Month (H)	Month (L)	Date (H)	Date (L)	Day (H)	Day (L)	S C (H)	S C (L)	C R

Year/month/date/day

Year (BCD)	Month (BCD)	Day (BCD)	Day of the week (BCD)						
			SUN	MON	TUE	WED	THU	FRI	SAT
00 to 99 [Lower 2digits of A.C. Example: 96 as 1996]	01 to 12	01 to 31	0 0	0 1	0 2	0 3	0 4	0 5	0 6

Execution condition

Write mode	Mode 0, Mode 1, Mode 2
PC operation state	Stop, Run

Example for use

Read out date of PC06. (Response time: 20 ms)

Command

$$: :\ \underbrace{0\ 6}_{\text{Station number}}\ ?\ \underbrace{2}_{\text{Response time}}\ M\ D\ Y\ \underbrace{3\ F}_{\text{Sum check code}}\ {}^{C}_{R}$$

Response

$$: :\ \underbrace{0\ 6}_{\text{Station number}}\ \#\ \underbrace{2}_{\text{Response time}}\ M\ D\ Y\ \underbrace{9\ 6\ 0\ 4\ 2\ 5\ 0\ 4}_{\text{April 25th, 1996 (Thu.)}}\ \underbrace{B\ D}_{\text{Sum check code}}\ {}^{C}_{R}$$

S D Y — Set date

[JW-1424K, JW-1442K, JW-1624K ,and JW-1642K only]

Function: Set date (year, month, date and day) of clock.

Communication format

Command

:	:	A D (H)	A D (L)	?	R I	S	D	Y	Year (H)	Year (L)	Month (H)	Month (L)	Date (H)	Date (L)	Day (H)	Day (L)	S C (H)	S C (L)	C R

Response

:	:	A D (H)	A D (L)	#	R I	S	D	Y	S C (H)	S C (L)	C R

Year/month/date/day

Year (BCD)	Month (BCD)	Day (BCD)	Day of the week (BCD)						
			SUN	MON	TUE	WED	THU	FRI	SAT
00 to 99 [Lower 2digits of A.C. Example: 96 as 1996]	01 to 12	01 to 31	0 0	0 1	0 2	0 3	0 4	0 5	0 6

Execution condition

Write mode	Mode1, Mode 2
PC operation state	Stop, Run

Example for use

Set date of PC07 as Friday, April 26th, 1996. (Response time: 30 ms)

Command

$$: :\ \underbrace{0\ 7}_{\text{Station number}}\ ?\ \underbrace{3}_{\text{Response time}}\ S\ D\ Y\ \underbrace{9\ 6\ 0\ 4\ 2\ 6\ 0\ 5}_{\text{April 26th, 1996 (Fri.)}}\ \underbrace{9\ 7}_{\text{Sum check code}}\ {}^{C}_{R}$$

Response

$$: :\ \underbrace{0\ 7}_{\text{Station number}}\ \#\ \underbrace{3}_{\text{Response time}}\ S\ D\ Y\ \underbrace{5\ 3}_{\text{Sum check code}}\ {}^{C}_{R}$$

MTM Read out time

[JW-1424K, JW-1442K, JW-1624K, and JW-1642K only]

Function	Read out time (hour, minute, second) of clock.
Communication format: Command	: : AD(H) AD(L) ? RI M T M SC(H) SC(L) CR
Communication format: Response	: : AD(H) AD(L) # RI M T M Hour(H) Hour(L) Min.(H) Min.(L) Sec.(H) Sec.(L) SC(H) SC(L) CR

Hour/minute/second	Hour (BCD)	Minute (BCD)	Second (BCD)
	00 to 23	00 to 59	00 to 59

Execution condition	
Write mode	Mode 0, Mode 1, Mode 2
PC operation state	Stop, Run

Example for use

Read out time of PC06. (Response time: 10 ms)

Command
$$: :\ \underbrace{0\ 6}_{\text{Station number}}\ ?\ \underbrace{1}_{\text{Response time}}\ M\ T\ M\ \underbrace{3\ C}_{\text{Sum check code}}\ \substack{C\\R}$$

Response
$$: :\ \underbrace{0\ 6}_{\text{Station number}}\ \#\ \underbrace{1}_{\text{Response time}}\ M\ T\ M\ \underbrace{0\ 8\ 3\ 0\ 3\ 0}_{\text{8:30:30 AM}}\ \underbrace{2\ A}_{\text{Sum check code}}\ \substack{C\\R}$$

STM Set time

[JW-1424K, JW-1442K, JW-1624K, and JW-1642K only]

Function	Set time (hour, minute, second) of clock.
Communication format: Command	: : AD(H) AD(L) ? RI S T M Hour(H) Hour(L) Min.(H) Min.(L) Sec.(H) Sec.(L) SC(H) SC(L) CR
Communication format: Response	: : AD(H) AD(L) # RI S T M SC(H) SC(L) CR

Hour/minute/second	Hour (BCD)	Minute (BCD)	Second (BCD)
	00 to 23	00 to 59	00 to 59

Execution condition	
Write mode	Mode 1, Mode 2
PC operation state	Stop, Run

Example for use

Set 13:30:00 in the clock of PC07. (Response time: 20 ms)

Command
$$: :\ \underbrace{0\ 7}_{\text{Station number}}\ ?\ \underbrace{2}_{\text{Response time}}\ S\ T\ M\ \underbrace{1\ 3\ 3\ 0\ 0\ 0}_{\text{13:30:30}}\ \underbrace{0\ D}_{\text{Sum check code}}\ \substack{C\\R}$$

Response
$$: :\ \underbrace{0\ 7}_{\text{Station number}}\ \#\ \underbrace{2}_{\text{Response time}}\ S\ T\ M\ \underbrace{5\ 0}_{\text{Sum check code}}\ \substack{C\\R}$$

ACL Correct, run, and stop clock [JW-1424K, JW-1442K, JW-1624K, and JW-1642K only]

Function		Correct time, run, and stop of clock.
Communication format	Command	: : AD(H) AD(L) ? RI A C L Data SC(H) SC(L) CR
	Response	: : AD(H) AD(L) # RI A C L SC(H) SC(L) CR
Data		00: Start clock 01: Stop clock (BCD) 8:30 sec. correction 0 to 29 sec.·······Becomes "00" sec. without addition of 1 minute. 30 to 59 sec.·····Becomes "00" sec. with addition of 1 minute.
Execution condition	Write mode	Mode 1, Mode 2
	PC operation state	Stop, Run
Example for use		Stop clock of PC07. (Response time: 20 ms) Command $: \, : \underbrace{0\ 7}_{\text{Station number}} ? \underbrace{3}_{\text{Response time}} A\ C\ L \underbrace{0\ 1}_{\text{Stop clock}} \underbrace{F\ 6}_{\text{Sum check code}}$ CR Response $: \, : \underbrace{0\ 7}_{\text{Station number}} \# \underbrace{3}_{\text{Response time}} A\ C\ L \underbrace{7\ 3}_{\text{Sum check code}}$ CR

MPC Monitor operational condition

Function		Monitor PC is running or stops.
Communication format	Command	: : AD(H) AD(L) ? RI M P C SC(H) SC(L) CR
	Response	: : AD(H) AD(L) # RI M P C Data SC(H) SC(L) CR
Data		0: Run 1: Stop by other optional device 2: Stop by HLT command
Execution condition	Write mode	Mode 0, Mode 1, Mode 2
	PC operation state	Stop, Run
Example for use		Monitor operational condition of PC01. (Response time: 20 ms) Command $: \, : \underbrace{0\ 1}_{\text{Station number}} ? \underbrace{2}_{\text{Response time}} M\ P\ C \underbrace{4\ E}_{\text{Sum check code}}$ CR Response $: \, : \underbrace{0\ 1}_{\text{Station number}} \# \underbrace{2}_{\text{Response time}} M\ P\ C \underbrace{0}_{\text{At operation}} \underbrace{3\ A}_{\text{Sum check code}}$ CR

HLT　Stop PC operation

Function		Stop PC operation.											
Communication format	Command	:	:	AD(H)	AD(L)	?	RI	H	L	T	SC(H)	SC(L)	CR
	Response	:	:	AD(H)	AD(L)	#	RI	H	L	T	SC(H)	SC(L)	CR
Execution condition	Write mode	Mode 0, Mode 1, Mode 2											
	PC operation state	Stop, Run											

Example for use

Stop operation of PC03. (Response time: 10 ms)

Command　$: :\ \underbrace{0\ 3}_{\text{Station number}}\ ?\ \underbrace{1}_{\text{Response time}}\ H\ L\ T\ \underbrace{4\ 5}_{\text{Sum check code}}\ \text{CR}$

Response　$: :\ \underbrace{0\ 3}_{\text{Station number}}\ \#\ \underbrace{1}_{\text{Response time}}\ H\ L\ T\ \underbrace{6\ 1}_{\text{Sum check code}}\ \text{CR}$

Note 1: A PC which has been stopped by HLT command cannot start operation again by support tool such as JW-13PG.

RUN　Restart PC operation

Function		Release HLT (stop PC operation) command, restart PC operation.											
Communication format	Command	:	:	AD(H)	AD(L)	?	RI	R	U	N	SC(H)	SC(L)	CR
	Response	:	:	AD(H)	AD(L)	#	RI	R	U	N	SC(H)	SC(L)	CR
Execution condition	Write mode	Mode 0, Mode 1, Mode 2											
	PC operation state	Stop, Run											

Example for use

Restart operation of PC03. (Response time: 10 ms)

Command　$: :\ \underbrace{0\ 3}_{\text{Station number}}\ ?\ \underbrace{1}_{\text{Response time}}\ R\ U\ N\ \underbrace{3\ 8}_{\text{Sum check code}}\ \text{CR}$

Response　$: :\ \underbrace{0\ 3}_{\text{Station number}}\ \#\ \underbrace{1}_{\text{Response time}}\ R\ U\ N\ \underbrace{5\ 4}_{\text{Sum check code}}\ \text{CR}$

Note 1: RUN command restarts a PC which has been stopped by HLT command. When the PC has stopped by other causes such as set to program mode by JW-13PG, it cannot restart operation even RUN command is executed.
In these cases, response returns normally.

SWE Read out write mode status

Function		Read out current write mode status.
Communication format	Command	: : A D (H) A D (L) ? R I S W E S C (H) S C (L) C R
	Response	: : A D (H) A D (L) # R I S W E Data S C (H) S C (L) C R
Data		0: Mode 0 .... Write prohibited for all memories 1: Mode 1 .... Write enable only for data memory 2: Mode 2 .... Write enable for all memories
Execution condition	Write mode	Mode 0, Mode 1, Mode 2
	PC operation state	Stop, Run
Example for use		Read out write mode status of PC06. (Response time: 10 ms) Command : : $\underbrace{0\ 6}_{\text{Station number}}\ ?\ \underbrace{1}_{\text{Response time}}\ S\ W\ E\ \underbrace{3\ B}_{\text{Sum check code}}$ C R Response : : $\underbrace{0\ 6}_{\text{Station number}}\ \#\ \underbrace{1}_{\text{Response time}}\ S\ W\ E\ \underbrace{0}_{\text{Mode 0}}\ \underbrace{3\ B}_{\text{Sum check code}}$ C R

Note 1: Be mode 0 (write prohibited), at power ON.

EWR Set write mode

Function		Set PC write mode.
Communication format	Command	: : A D (H) A D (L) ? R I E W R Data S C (H) S C (L) C R
	Response	: : A D (H) A D (L) # R I E W R S C (H) S C (L) C R
Data		0: Mode 0 .... Write prohibited for all memories 1: Mode 1 .... Write enable only for data memory 2: Mode 2 .... Write enable for all memories
Execution condition	Write mode	Mode 1, Mode 2
	PC operation state	Stop, Run
Example for use		Set write mode of PC22 to mode 2 (write allowed for all memories). (Response time: 40 ms) Command : : $\underbrace{2\ 2}_{\text{Station number}}\ ?\ \underbrace{4}_{\text{Response time}}\ E\ W\ R\ \underbrace{2}_{\text{Mode 2}}\ \underbrace{0\ 9}_{\text{Sum check code}}$ C R Response : : $\underbrace{2\ 2}_{\text{Station number}}\ \#\ \underbrace{4}_{\text{Response time}}\ E\ W\ R\ \underbrace{5\ 7}_{\text{Sum check code}}$ C R

Note 1: In order to prevent inadvertent accident, set the mode to "mode 0" (write prohibited) while not writing data.

TST	Turn back the message

Function	Send back the received command as it is.

Communication format

Command

:	:	A D (H)	A D (L)	?	R I	T	S	T	M_1	....	M_n	S C (H)	S C (L)	C R

Response

:	:	A D (H)	A D (L)	#	R I	T	S	T	M_1	....	M_n	S C (H)	S C (L)	C R

Data	Message visible character strings (20H to 7EH of ASCII code) Max. 1024 characters	
Execution condition	Write mode	Mode 0, Mode 1, Mode 2
	PC operation state	Stop, Run

Example for use

Test sending and returning of a message toward PC01. (Response time: 50 ms)

Command
$$::\underbrace{0\ 1}_{\text{Station number}}\ ?\ \underbrace{5}_{\text{Response time}}\ T\ S\ T\ \underbrace{T\ E\ S\ T\ C\ O\ M\ M\ A\ N\ D}_{\text{Message}}\ \underbrace{F\ 1}_{\text{Sum check code}}\ \begin{matrix}C\\R\end{matrix}$$

Response
$$::\underbrace{0\ 1}_{\text{Station number}}\ \#\ \underbrace{5}_{\text{Response time}}\ T\ S\ T\ \underbrace{T\ E\ S\ T\ C\ O\ M\ M\ A\ N\ D}_{\text{Message}}\ \underbrace{0\ D}_{\text{Sum check code}}\ \begin{matrix}C\\R\end{matrix}$$

PAS	Release secret/password registration

Function	Release secret function, register password.

Communication format

Command

:	:	A D (H)	A D (L)	?	R I	P	A	S	Data	Password	S C (H)	S C (L)	C R

Response

:	:	A D (H)	A D (L)	#	R I	P	A	S	Data	S C (H)	S C (L)	C R

Data	0: Release····Release the secret function. 1: Temporary registration····Set before regular registration 2: Regular registration····Set after temporary registration		
Pass word	Alphanumeric 4 characters		
Execution condition		Data = 0	Data = 1, 2
	Write mode	Mode 0, Mode 1, Mode 2	Mode 2
	PC operation state	Stop, Run	Stop by HLT command

Example for use

Regularly register password 15AE to PC05. (Response time: 20 ms)

Command
$$::\underbrace{0\ 5}_{\text{Station number}}\ ?\ \underbrace{2}_{\text{Response time}}\ P\ A\ S\ \underbrace{2}_{\text{Regular registration}}\ \underbrace{1\ 5\ A\ E}_{\text{Password}}\ \underbrace{2\ 8}_{\text{Sum check code}}\ \begin{matrix}C\\R\end{matrix}$$

Response
$$::\underbrace{0\ 5}_{\text{Station number}}\ \#\ \underbrace{2}_{\text{Response time}}\ P\ A\ S\ \underbrace{6\ 2}_{\text{Sum check code}}\ \begin{matrix}C\\R\end{matrix}$$

Note 1: If secret function is set, the following commands cannot be used:
RPM, WPM, WSM, CTC, HLT, RUN, SES.
In addition, setting of secret function on the JW10 requires a support tool such as JW-13PG to set its password which is connected to the JW10.

S E S Set secret function

Function	Set secret function.

Communication format

Command

:	:	A D (H)	A D (L)	?	R I	S	E	S	Data	S C (H)	S C (L)	C R

Response

:	:	A D (H)	A D (L)	#	R I	S	E	S	S C (H)	S C (L)	C R

Data

1: Enable secret function · · · · Enable secret function by the registered password.
F: Delete · · · · Delete registered password of JW10.

Execution condition

	Data = 1	Data = F
Write mode	Mode 0, Mode 1, Mode 2	Mode 2
PC operation state	Stop, Run	Stop by HLT command

Example for use

Enable secret function of PC07. (Response time: 10 ms)

Command $::\underbrace{0\ 7}_{\text{Station number}}\ ?\ \underbrace{1}_{\text{Response time}}\ S\ E\ S\ \underbrace{1}_{\text{Enable secret function}}\ \underbrace{0\ D}_{\text{Sum check code}}\ {}^{C}_{R}$

Response $::\underbrace{0\ 7}_{\text{Station number}}\ \#\ \underbrace{1}_{\text{Response time}}\ S\ E\ S\ \underbrace{5\ A}_{\text{Sum check code}}\ {}^{C}_{R}$

Note 1: If secret function is set, the following commands cannot be used:
RPM, WPM, WSM, CTC, HLT, RUN, SES.
In addition, setting of secret function on the JW10 requires a support tool such as JW-13PG to set its password which is connected to the JW10.

S E I Check secret function

Function	Check secret function (enable/disable).

Communication format

Command

:	:	A D (H)	A D (L)	?	R I	S	E	I	S C (H)	S C (L)	C R

Response

:	:	A D (H)	A D (L)	#	R I	S	E	I	Data	S C (H)	S C (L)	C R

Data

0: Disable secret function
1: Enable secret function

Execution condition

Write mode	Mode 0, Mode 1, Mode 2
PC operation state	Stop, Run

Example for use

Check secret function (enable/disable) of PC03. (Response time: 30ms)

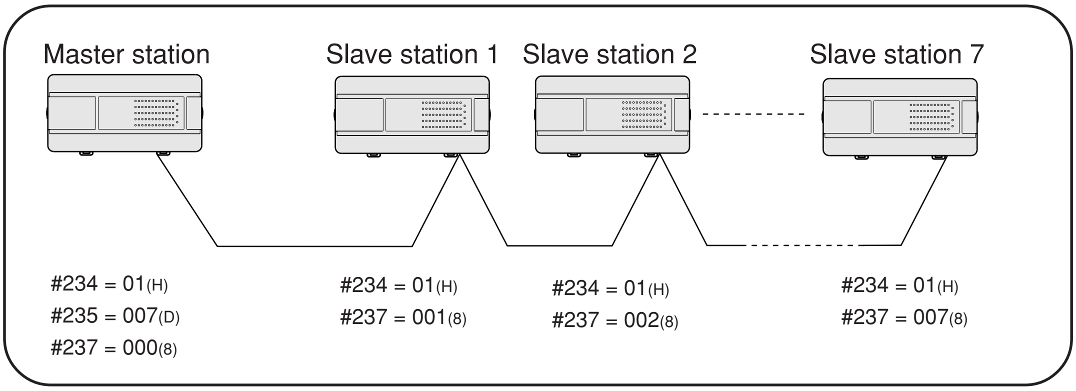
Command $::\underbrace{0\ 3}_{\text{Station number}}\ ?\ \underbrace{3}_{\text{Response time}}\ S\ E\ I\ \underbrace{4\ A}_{\text{Sum check code}}\ {}^{C}_{R}$

Response $::\underbrace{0\ 3}_{\text{Station number}}\ \#\ \underbrace{3}_{\text{Response time}}\ S\ E\ I\ \underbrace{1}_{\text{Enable secret function}}\ \underbrace{3\ 5}_{\text{Sum check code}}\ {}^{C}_{R}$

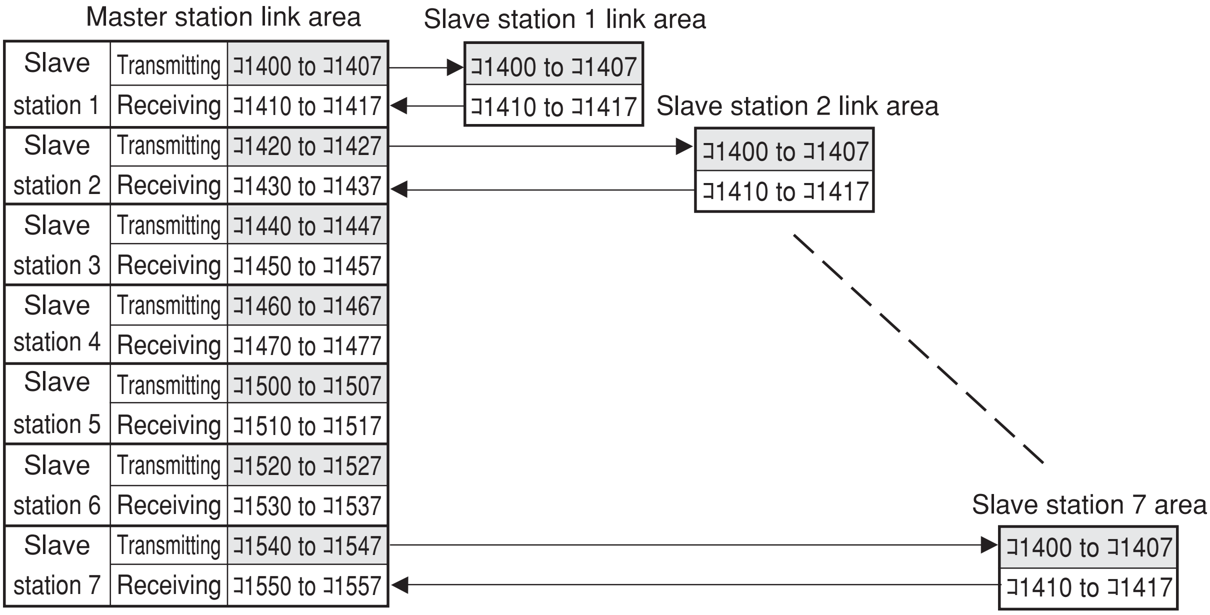
13-3 Data link

[1] Communication specifications

- Data link can communicate between one JW10 master station and maximum of seven JW10 slave stations.



- Data link area of master station and slave station are allocated as shown below.



Master station link area

Slave station 1	Transmitting	コ1400 to コ1407
	Receiving	コ1410 to コ1417
Slave station 2	Transmitting	コ1420 to コ1427
	Receiving	コ1430 to コ1437
Slave station 3	Transmitting	コ1440 to コ1447
	Receiving	コ1450 to コ1457
Slave station 4	Transmitting	コ1460 to コ1467
	Receiving	コ1470 to コ1477
Slave station 5	Transmitting	コ1500 to コ1507
	Receiving	コ1510 to コ1517
Slave station 6	Transmitting	コ1520 to コ1527
	Receiving	コ1530 to コ1537
Slave station 7	Transmitting	コ1540 to コ1547
	Receiving	コ1550 to コ1557

Slave station 1 link area

コ1400 to コ1407
コ1410 to コ1417

Slave station 2 link area

コ1400 to コ1407
コ1410 to コ1417

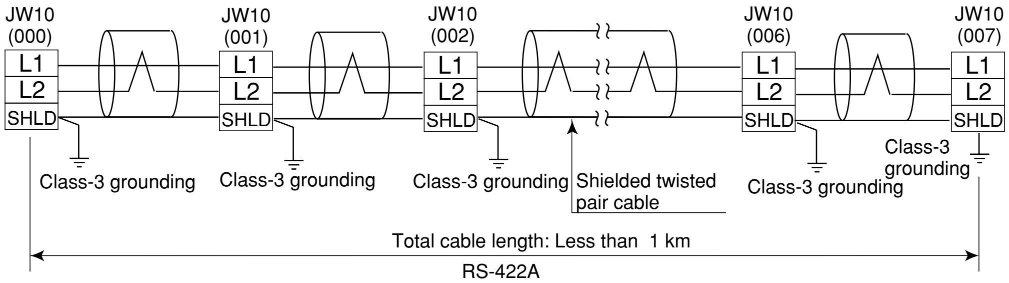
Slave station 7 area

コ1400 to コ1407
コ1410 to コ1417

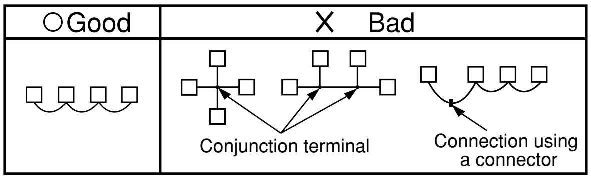
Note: Data are transferred between master station and slave station. They cannot be transferred between slave stations.

Item	Specifications
Data transfer standard	EIA RS-422A or equivalent
Transfer rate	76800, 38400 bits/s
No. of slave stations connected	Max. 7 sets
Link area	Master station: コ1400 to コ1557, slave station: コ1400 to コ1417
No. of link bytes per station	Master station to slave station: 8 bytes Slave station to master station: 8 bytes
Communication line	Shielded twisted pair cable. Party line connection. 2-wire system. Total length: 500 m (76800 bits/s), 1 km (38400 bits/s)

[2] Wiring



Note 1 : Do class-3 grounding for SHLD terminal. The operation without class-3 grounding may cause malfunction due to noise.
Note 2 : Never branch a communication cable into more than 3 lines.



[3] Setting system memory

- When data link function is used, set the following system memory in master and slave stations.

(1) System memory of master station

#234	Communication mode	01(H): Data link mode *Note 1
#235	No. of slave stations	001 to 007(D)
#236	Transfer rate	00(H): 76800 bits/s, 01(H): 38400 bits/s *Note 2
#237	Own station number	000(8)

(2) System memory of slave station

#234	Communication mode	01(H): Data link mode *Note 1
#236	Transfer rate	00(H): 76800 bits/s, 01(H): 38400 bits/s *Note 2
#237	Own station number	001 to 007(8) *Note 3

*Note 1 : To quit data link communication, set #234 as 00(H). (Computer link mode)

*Note 2 : Set identical communication speed for the master station and slave stations.

*Note 3 : Allocate slave station ID numbers for each with sequential numbers from 001.

*Note 4 : Communication mode and station number are set on system memory. We recommend to mark these settings on a label and stick it on each case for easily readable the set contents.

[Example of label]

Data link
#001

[4] Communication flag

- Communication condition of a master station and each slave station can be confirmed by communication flag.

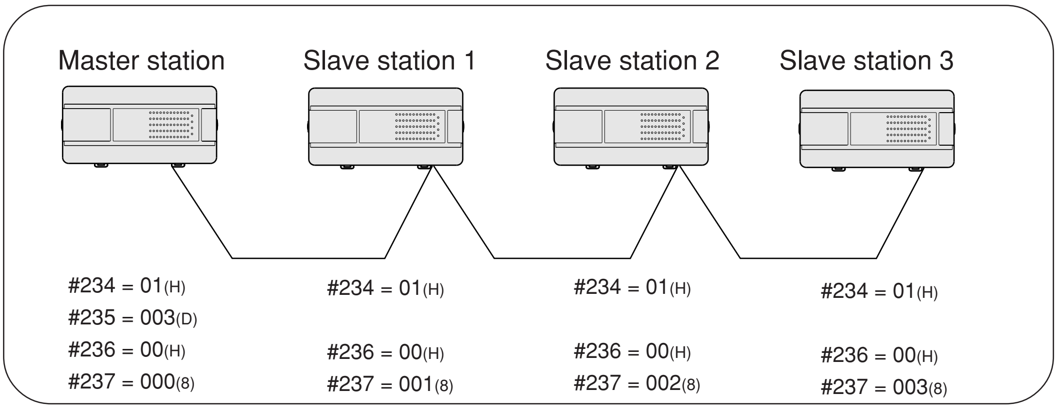
(1) Communication flag of master station

Communi-cation flag	Contents
07340	While normally communicating with all slave stations: ON
07341	While normally communicating with slave station 1: ON
07342	While normally communicating with slave station 2: ON
07343	While normally communicating with slave station 3: ON
07344	While normally communicating with slave station 4: ON
07345	While normally communicating with slave station 5: ON
07346	While normally communicating with slave station 6: ON
07347	While normally communicating with slave station 7: ON

(2) Communication flag of slave station

Communi-cation flag	Contents
07340	While normally communicating with the master station: ON
07341	Not in use: OFF
07342	Not in use: OFF
07343	Not in use: OFF
07344	Not in use: OFF
07345	Not in use: OFF
07346	Not in use: OFF
07347	Not in use: OFF

[Example of communication flag state]



Master station	Slave station 1	Slave station 2	Slave station 3
#234 = 01(H)	#234 = 01(H)	#234 = 01(H)	#234 = 01(H)
#235 = 003(D)			
#236 = 00(H)	#236 = 00(H)	#236 = 00(H)	#236 = 00(H)
#237 = 000(8)	#237 = 001(8)	#237 = 002(8)	#237 = 003(8)

1) In case of normal communication (Master station communicates to all save stations normally.)

	Master station		Slave station 1		Slave station 2		Slave station 3
07340	**ON**	07340	**ON**	07340	**ON**	07340	**ON**
07341	**ON**	07341	OFF *	07341	OFF *	07341	OFF *
07342	**ON**	07342	OFF *	07342	OFF *	07342	OFF *
07343	**ON**	07343	OFF *	07343	OFF *	07343	OFF *
07344	OFF *	07344	OFF *	07344	OFF *	07344	OFF *
07345	OFF *	07345	OFF *	07345	OFF *	07345	OFF *
07346	OFF *	07346	OFF *	07346	OFF *	07346	OFF *
07347	OFF *	07347	OFF *	07347	OFF *	07347	OFF *

* Not in use

2) When master station is error (Power OFF, disconnect, stop mode, module error)

	Master station		Slave station 1		Slave station 2		Slave station 3
07340	**OFF**	07340	**OFF**	07340	**OFF**	07340	**OFF**
07341	**OFF**	07341	OFF *	07341	OFF *	07341	OFF *
07342	**OFF**	07342	OFF *	07342	OFF *	07342	OFF *
07343	**OFF**	07343	OFF *	07343	OFF *	07343	OFF *
07344	OFF *	07344	OFF *	07344	OFF *	07344	OFF *
07345	OFF *	07345	OFF *	07345	OFF *	07345	OFF *
07346	OFF *	07346	OFF *	07346	OFF *	07346	OFF *
07347	OFF *	07347	OFF *	07347	OFF *	07347	OFF *

* Not in use

3) When slave station 1 is error (Power OFF, disconnect, stop mode, module error)

	Master station		Slave station 1		Slave station 2		Slave station 3
07340	**OFF**	07340	**OFF**	07340	**ON**	07340	**ON**
07341	**OFF**	07341	OFF *	07341	OFF *	07341	OFF *
07342	**ON**	07342	OFF *	07342	OFF *	07342	OFF *
07343	**ON**	07343	OFF *	07343	OFF *	07343	OFF *
07344	OFF *	07344	OFF *	07344	OFF *	07344	OFF *
07345	OFF *	07345	OFF *	07345	OFF *	07345	OFF *
07346	OFF *	07346	OFF *	07346	OFF *	07346	OFF *
07347	OFF *	07347	OFF *	07347	OFF *	07347	OFF *

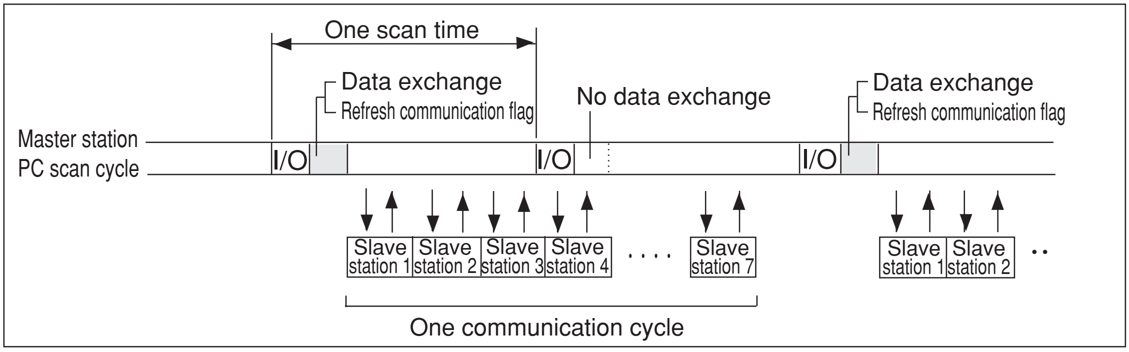
* Not in use

Note : When slave station 1 has an error (such as power OFF, disconnection, stop mode, or module error), the master station continues communication with other normal slave stations.
When slave station 1 recovers from error condition, the master station restart communication with slave station 1.

[5] Communication timing and transmission time required

(1) Communication timing of master station

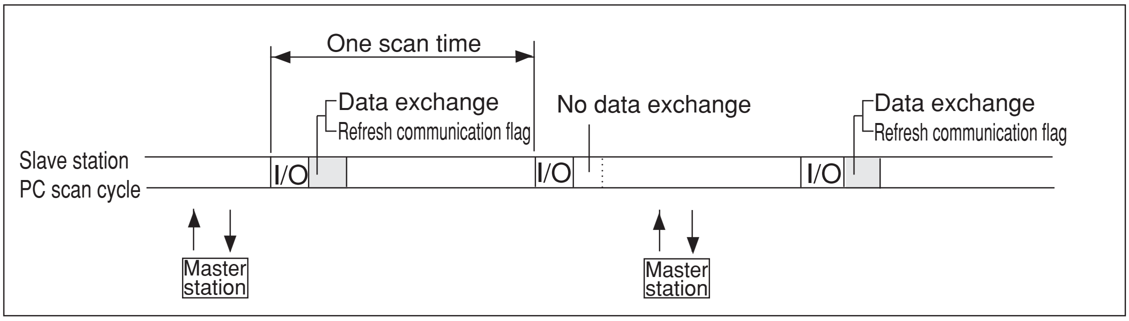
- The master station communicates with each slave station asynchronous with master station's scan cycle.
- The master station exchanges data for data link and refreshes communication flag with scan cycle after completion of communication with slave stations.
  Scan time of the master station increases with the interval required for these procedures (approximately 0.2 ms).



- When an error occurs during communication between the master station and slave stations, the master station does not exchange data for data link with the error occurred slave station, and turns OFF this slave station's communication flag. However, the master station exchanges data with other normal slave stations.
  Possible causes of errors are as follows:
  ① Occurrence of sum check error.
  ② Slave station is in stop mode or abnormal condition.
  ③ Slave station is not connected or there is disconnection of a cable.
- When the master station is in stop mode or is in abnormal condition, it does not communicate with any slave station.

(2) Communication timing of slave station

- A slave station communicates with the master station asynchronous with slave station's scan cycle.
- A slave station exchanges data for data link and refreshes communication flag with scan cycle after completion of communication with the master stations.
  Scan time of a slave station increases with the interval required for these procedures (approximately 0.2 ms).



- When an error occurs during communication between the master station, the slave station does not exchange data for data link, and turns OFF its own communication flag.
  Possible causes of errors are as follows:
  ① Occurrence of sum check error.
  ② Slave station is in stop mode or abnormal condition.
  ③ Master station is in stop mode or abnormal condition.
  ④ Slave station is not connected or there is disconnection of a cable.

(3) Transmission time required

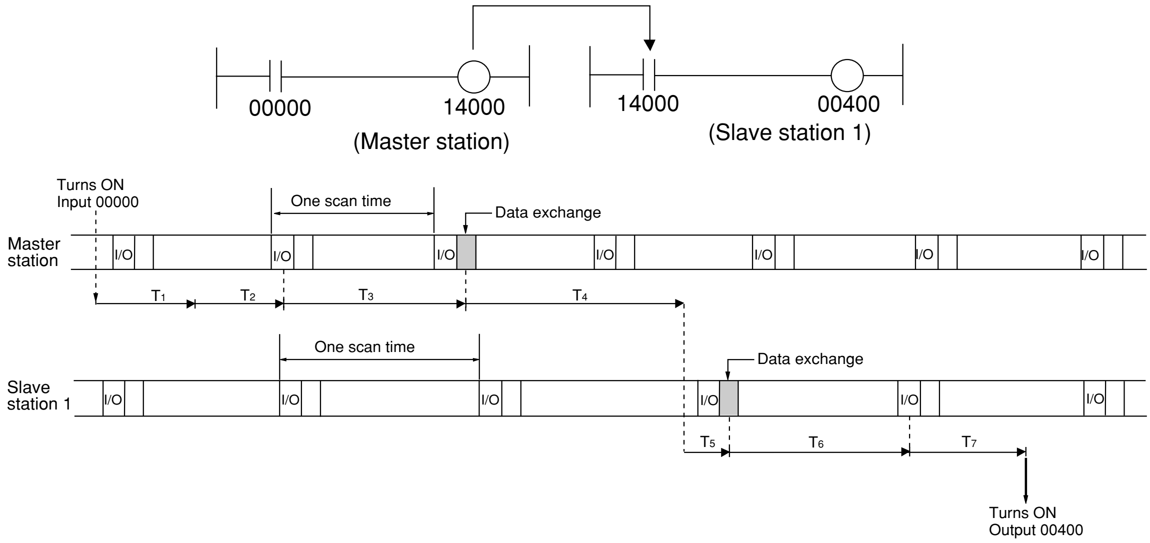
- Interval that a data linked master station needs to communicate with all of slave stations "T" (one communication cycle time) is as follows:

Number of slave stations	76800 bits/s (#236 = 00$_{(H)}$)		38400 bits/s (#236 = 01$_{(H)}$)	
1	3.6 ms	3.6 × number of slave stations [ms]	7.6 ms	7.6 × number of slave stations [ms]
2	7.2 ms		15.2 ms	
3	10.8 ms		22.8 ms	
4	14.4 ms		30.4 ms	
5	18.0 ms		38.0 ms	
6	21.6 ms		45.6 ms	
7	25.2 ms		53.2 ms	

(4) Communication delay time

- To receive data with data link, the following delays will occur.

a) Communication from master station to slave station



T_1: Delay on input port

T_2: Time until a PC detects input condition (maximum one scan time)

T_3: Operation time (one scan time) of data transmitting side PC (master station)

T_4: Time to complete transmitting operated result (maximum [one communication cycle time + one scan time]).

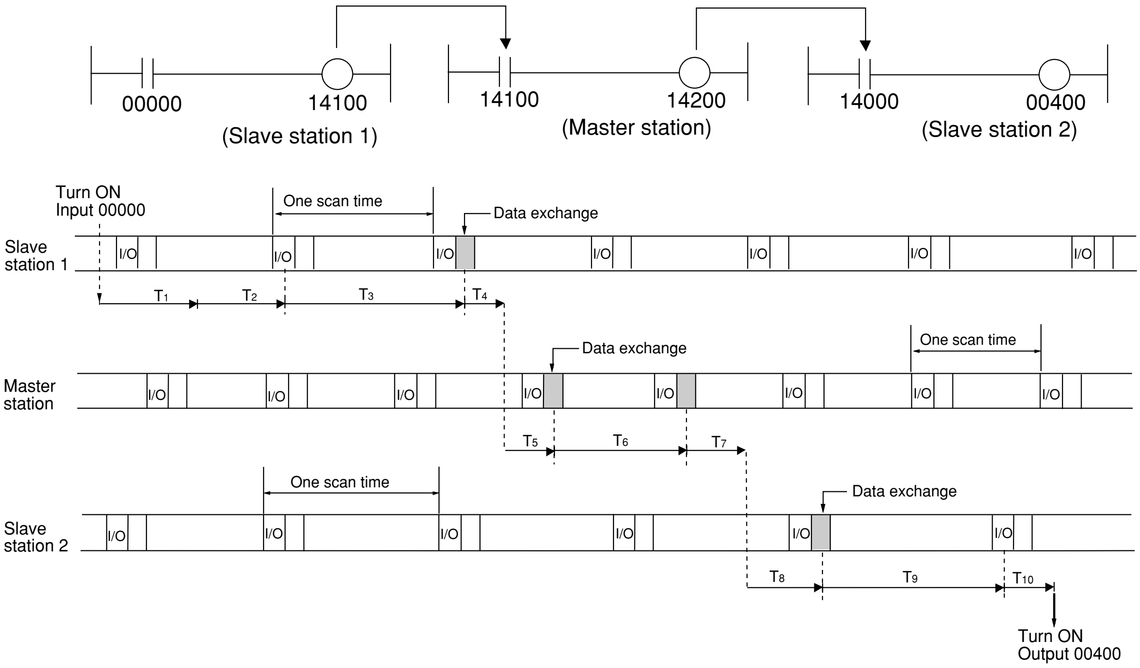
T_5: Time that data receiving side PC (slave station 1) completes writing received data onto PC's data memory (maximum one scan time)

T_6: Operation time of data receiving side PC (slave station 1) (one scan time)

T_7: Delay of output port

Total delay time = $T_1 + T_2 + T_3 + T_4 + T_5 + T_6 + T_7$

b) Communication between slave stations (slave station→master station→slave station)



T1: Delay on input port

T2: Time until the PC detects input condition (maximum one scan time)

T3: Operation time (one scan time) of slave station 01 (one scan time)

T4: Time that slave station 1 takes to complete transmitting operation result (maximum [one communication cycle time + one scan time]).

T5: Time that the master station takes to complete writing received data onto PC's data memory (maximum one scan time)

T6: Operation time of the master station (one scan time)

T7: Time that the master station takes to complete transmitting operated data (maximum [one communication cycle time + one scan time]).

T8: Time that slave station 2 takes to complete writing received data onto PC's data memory (maximum one scan time)

T9: Operation time of slave station 2 (one scan time)

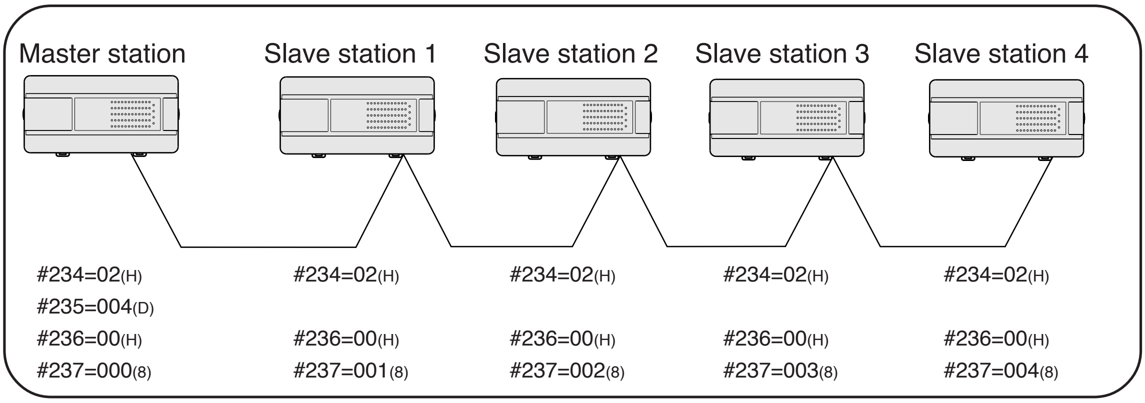
T10: Delay of output port

Total delay time = $T_1 + T_2 + T_3 + T_4 + T_5 + T_6 + T_7 + T_8 + T_9 + T_{10}$

13-4 Remote I/O

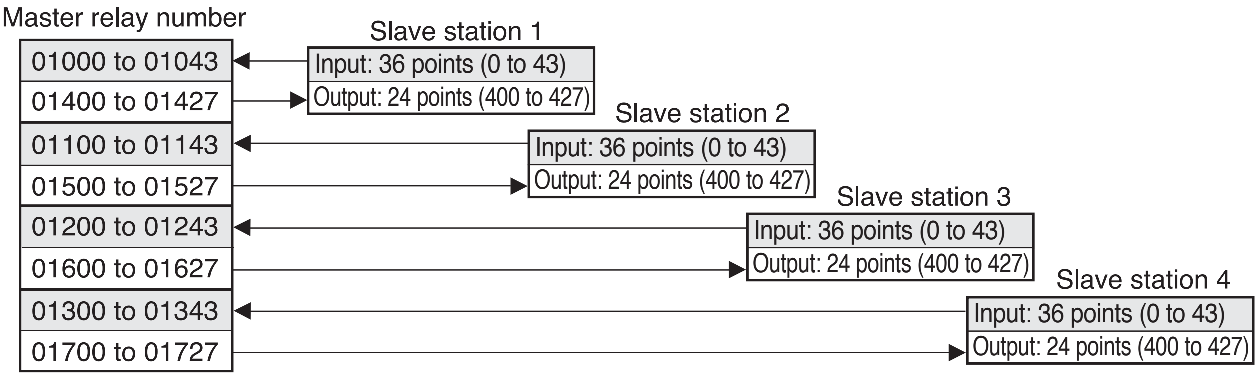
[1] Communication specifications

- Remote I/O can communicate between one JW10 master station and maximum of four JW10 slave stations.



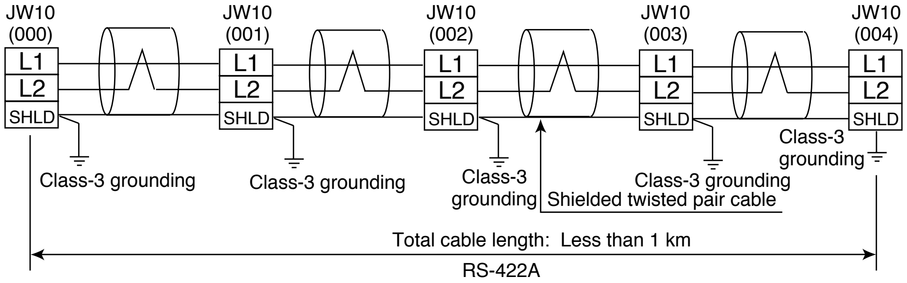
Note: Use JW10 basic module for remote slave station. The expansion module, analog input module, and analog output module are not usable as remote slave station.

- Remote I/O area of master station is allocated as follows:


Master relay number
01000 to 01043
01400 to 01427
01100 to 01143
01500 to 01527
01200 to 01243
01600 to 01627
01300 to 01343
01700 to 01727
Slave station 1
Input: 36 points (0 to 43)
Output: 24 points (400 to 427)
Slave station 2
Input: 36 points (0 to 43)
Output: 24 points (400 to 427)
Slave station 3
Input: 36 points (0 to 43)
Output: 24 points (400 to 427)
Slave station 4
Input: 36 points (0 to 43)
Output: 24 points (400 to 427)

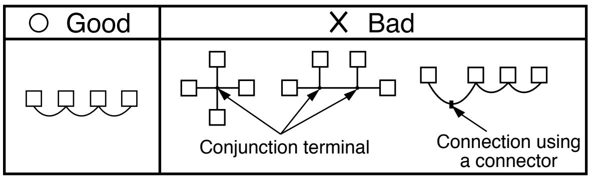
Item	Specifications	
Data transfer standard	EIA RS-422A or equivalent	
Transfer rate	76800, 38400 bits/s	
No. of slave stations connected	Max. 4 sets	
Remote I/O area (Allocation of master station)	Slave station 1	Input: 36 points (01000 to 01043) Output: 24 points (01400 to 01427) Power error input (01047)
	Slave station 2	Input: 36 points (01100 to 01143) Output: 24 points (01500 to 01527) Power error input (01147)
	Slave station 3	Input: 36 points (01200 to 01243) Output: 24 points (01600 to 01627) Power error input (01247)
	Slave station 4	Input: 36 points (01300 to 01343) Output: 24 points (01700 to 01727) Power error input (01347)
Communication line	Shielded twisted pair cable. Party line connection. 2-wire system Total length: 500 m (76800 bits/s), 1 km (38400 bits/s)	

[2] Wiring



Note 1: Do class-3 grounding for SHLD terminal. The operation without class-3 grounding may cause malfunction due to noise.

Note 2: Never branch a communication cable into more than 3 lines.



[3] Setting system memory

- When remote I/O is used, set the following system memory in master and slave stations.

(1) System memory of master station

#234	Communication mode	02(H): Remote I/O mode (Note 1)
#235	No. of slave stations	001 to 004(D)
#236	Transfer rate	00(H): 76800 bits/s, 01(H): 38400 bits/s (Note 2)
#237	Own station number	000(8)

(2) System memory of slave station

#234	Communication mode	02(H): Remote I/O mode
#236	Transfer rate	00(H): 76800 bits/s, 01(H): 38400 bits/s (Note 2)
#237	Own station number	001 to 004(8) (Note 3)
#206	Output status at error of own station	00(H): Reset (All of own outputs are OFF) 55(H): Hold the status just before error occurrence

Note 1 : To quit remote I/O communication, set #234 as 00(H). (Computer link mode)

Note 2 : Set identical communication speed for the master station and slave stations.

Note 3 : Allocate slave station ID numbers for each with sequential numbers from 001.

Note 4 : Communication mode and station number are set on system memory. We recommend to mark these settings on a label and stick it on each case for easily readable the set contents.

[Example of label]

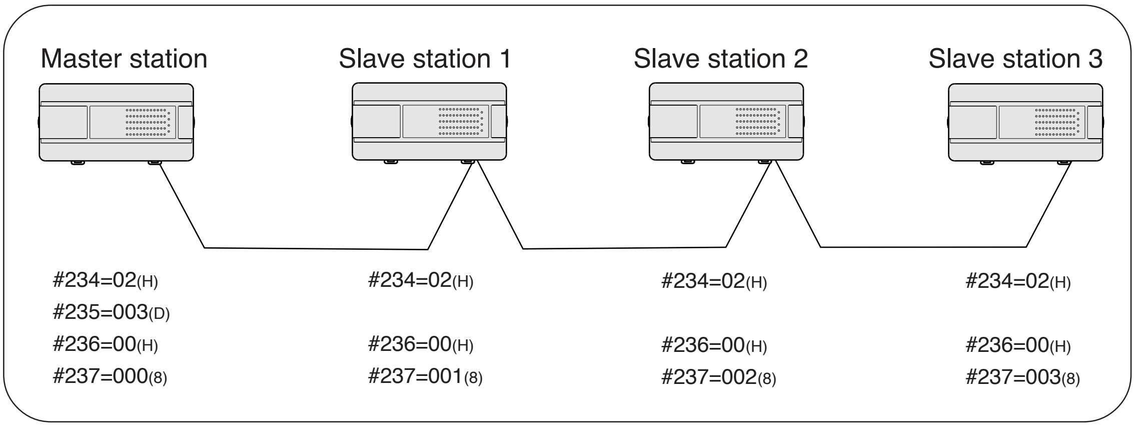
Data link
#001

[4] Communication flag

- Communication condition of a master station and each slave station can be confirmed by communication flag of master station.

Communication flag	Contents
07340	While normally communicating with all slave stations: ON
07341	While normally communicating with slave station 1 : ON
07342	While normally communicating with slave station 2 : ON
07343	While normally communicating with slave station 3 : ON
07344	While normally communicating with slave station 4 : ON

[Example of communication flag state]



Master station	Slave station 1	Slave station 2	Slave station 3
#234=02(H)	#234=02(H)	#234=02(H)	#234=02(H)
#235=003(D)			
#236=00(H)	#236=00(H)	#236=00(H)	#236=00(H)
#237=000(8)	#237=001(8)	#237=002(8)	#237=003(8)

1) In case of normal communication (Master station communicates to all save stations normally.)

	Master station
07340	**ON**
07341	**ON**
07342	**ON**
07343	**ON**
07344	OFF *

* Not in use

2) When master station is error (Power OFF, disconect, stop mode, module error)

	Master station
07340	**OFF**
07341	**OFF**
07342	**OFF**
07343	**OFF**
07344	OFF *

* Not in use

3) When slave station 1 is error (Power OFF, disconnect, stop mode, module error)

	Master station
07340	**OFF**
07341	**OFF**
07342	**ON**
07343	**ON**
07344	OFF *

* Not in use

Note: If slave station 1 has an error (power OFF, not connected, or module error), the master station stops operation and communication with all the slave stations.
When slave station 1 recovers from error condition, the master station resumes operation and communication with all the slave stations.

[5] Operation condition of remote I/O slave station

- The table below shows statue of remote I/O slave station at normal and error occurred condition.

System condition		Indication lamp				Status of output port		Halt output Note 1:	Error code(BCD) #160 to 167	
		POWER (green)	RUN (green)	ERR (red)	COMM (green)	#206 = 00(H) (set by slave station)	#206 = 55(H) (set by slave station)		Master station	Slave station
Normal	Master station is operating (monitor, change mode)	ON ●	ON ●	OFF ○	Blink ◎	—	—	Close (ON)	—	—
Error *Note 3	Master station has stopped operation (program mode)	ON ●	OFF ○	ON ●	OFF ○	OFF	Latch	Open (OFF)	—	53
	Master station error	ON ●	OFF ○	ON ●	OFF ○	OFF	Latch	Open (OFF)	Codes other than 53 and 22	53
	Slave station is not connected	ON ●	OFF ○	ON ●	OFF ○	OFF	Latch	Open (OFF)	53	53
	*Note 2 Slave station error (except battery error)	ON ●	OFF ○	ON ●	OFF ○	OFF	Latch	Open (OFF)	53	*Note 3
	*Note 4 Slave station battery error	ON ●	ON ●	ON ●	Blink ◎	—	—	Close (ON)	—	22
	Slave station power OFF	OFF ○	OFF ○	OFF ○	OFF ○	OFF	OFF	Open (OFF)	53	—

*Note 1 : JW-1324K/1342K do not have halt output.

*Note 2 : In case of CPU error or input/output error, status different from above may occur.

*Note 3 : When slave station error occurs, its error code is stored into slave station's system memory #160 to #167 at power input of slave station.
Error codes of slave station are the same as error codes of master station . See "8-3 Self diagnosis" for details.

*Note 4 : A slave station battery error can be monitored with the slave station battery error flag on a remote I/O master station.

(ROM version : version 2.3 or later)

Slave station battery error flag	Contents
01047	Slave station 1 battery error
01147	Slave station 2 battery error
01247	Slave station 3 battery error
01347	Slave station 4 battery error

The slave station battery error flag will be turned ON in the a remote I/O area input relay on the master station when the battery voltage of any slave station drops below 2.5V.

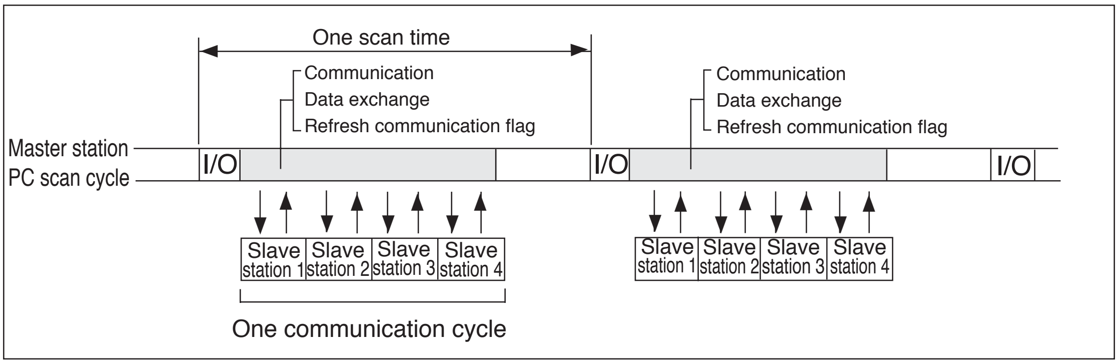
Notes

- When system memory of the JW10 is set to #234 = 02(H) and #237≠000(8), the JW10 functions as remote I/O slave station.
  In this case, the JW10 performs only I/O refresh and you can set system memory.
- Do not use halt output of slave station.

[6] Communication timing and transmission time required

(1) Communication timing

- The master station communicates, exchanges remote I/O data, and refresh communication flag synchronous with master station's scan cycle.
  Scan time of the master station increases with these intervals (one communication cycle time) only.



- When an error occurs during communication between the master station and slave stations, it stops remote I/O communication and PC operation, and enters into error condition. The master station also turns OFF communication flag of its own station and error occurred slave station.
  Possible causes of errors are as follows:
  ① Occurrence of sum check error.
  ② Slave station is in stop mode or abnormal condition.
  ③ Slave station is not connected or there is disconnection of a cable.
- When the master station is in stop mode or is in abnormal condition, it does not communicate with any slave station.

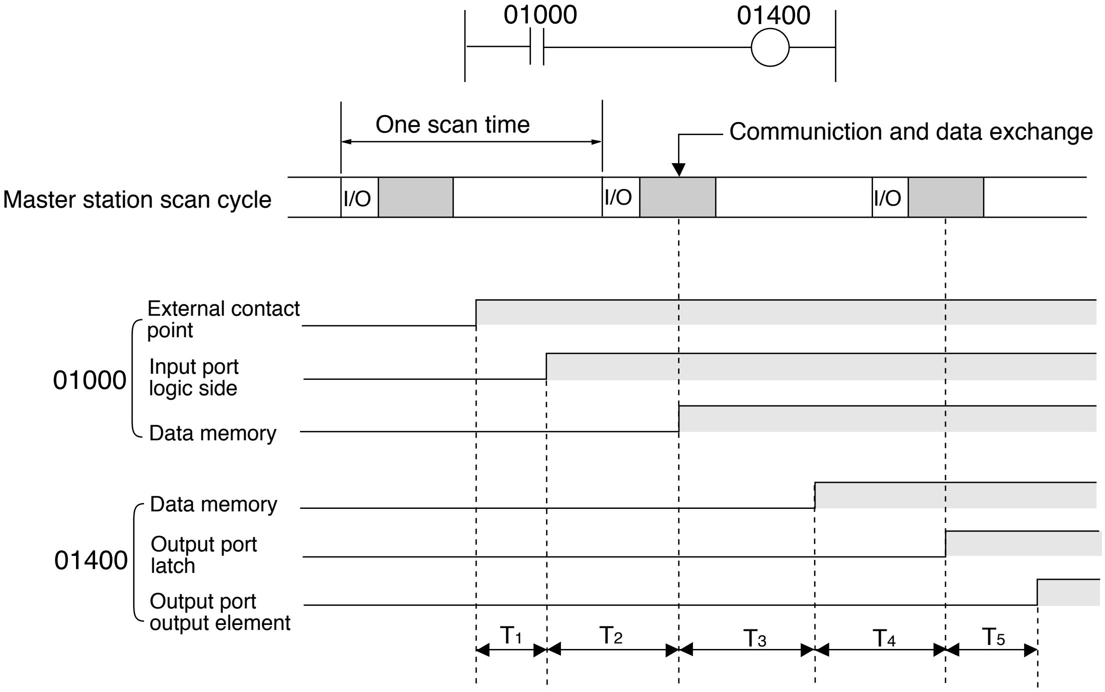
(2) Transmission time required

- Required interval time T for the remote I/O master station to communicate with all the slave stations (one communication cycle time) is as follows:

Number of slave stations	78600 bits/sec. (#236 = 00(H))		38400 bits/sec. (#236 = 01(H))	
1	3.8 ms	1.2 + 2.6 × number of slave stations (ms)	7.6 ms	2.4 + 5.2 × number of slave stations (ms)
2	6.4 ms		12.8 ms	
3	9.0 ms		18.0 ms	
4	11.6 ms		23.2 ms	

(3) Communiction delay time

- When receiving data with remote I/O, the following delay will occur.



T_1: Delay on input port (response time from OFF to ON)

T_2: Time until completion of writing data onto master station data memory (maximum one scan time)

T_3: Operation time (one scan time)

T_4: Time until completion of outputting operation result to output port (one scan time)

T_5: Delay of output port (response time from OFF to ON)

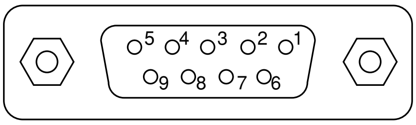
Chapter 14 How to Use the MMI Port

14-1 MMI port

- The JW10 MMI port has two modes: PG mode and computer link mode.
- The mode is selected by the PG/COM signal voltage on the MMI port.

PG/COM signal	Mode
High (open)	PG mode
Low (GND)	Computer link mode

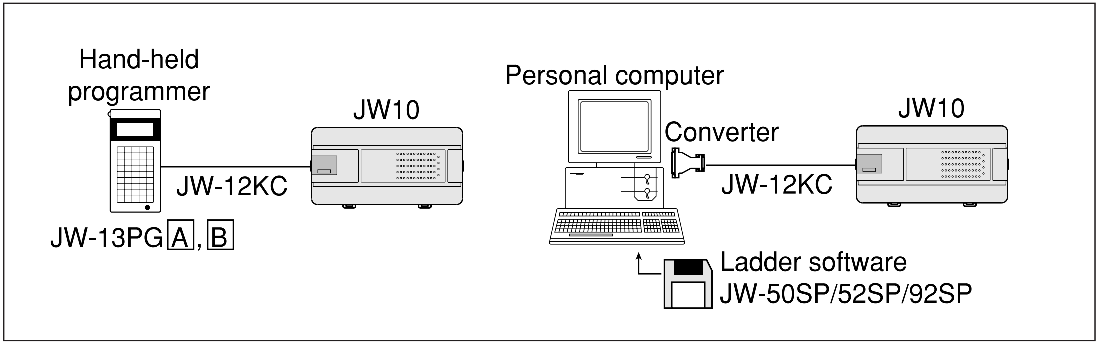
MMI port



Pin no.	Signal name
1	5 V (Vcc)
2	RX
3	TX
4	**PG/COM**
5	GND
6	5 V (Vcc)
7	/RX
8	/TX
9	GND

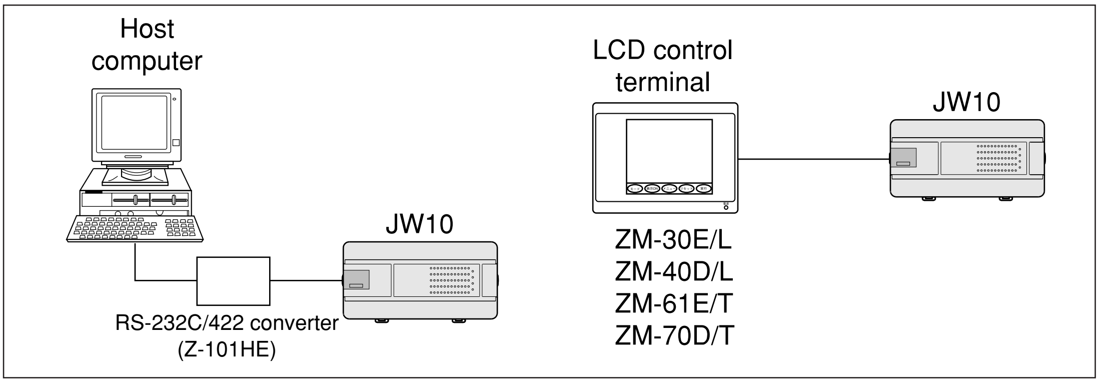
(1) PG mode

- A support tool, like the JW-13PG, is connected to perform JW10 programming and monitoring.
- The support tool and the MMI port must be connected using a special connecting cable (JW-12KC).



(2) Computer link mode

- The JW10 can communicate to a host computer or an LCD control terminal.
  (The JW10 can receive host computer's commands and response to the host computer.)
- Even if the JW10 is stopped or something is wrong with it, communication is still possible. (With the exception of a problem in the communication facility itself)
- The functions are the same as those of a computer link through a communication port. However, when the communication port is used, it allows you to communicate with up to 63 JW10s. But, if the MMI port is used, the host computer can only be connected to one JW10.
- The MMI port computer link and various other functions (computer link, data link, remote I/O) from the communication port can be used simultaneously.



14-2 PG mode

- In the PG mode, a connection between various types of support tools and JW10 allows you to perform programming and monitoring.
- Pay attention to support tool models used because the JW10's functions may be limited, depending on the tool's version.

[1] Types of support tools

(1) Models that allow you to use all of the JW10 functions

Part name	Model name	Summary
Hand-held programmer	JW-13PG (with $\boxed{A}$, $\boxed{B}$ mark)	• 16-character, 4-line, dot-matrix LCD display • Instruction word program, monitor
Multipurpose programmer	JW-50PG (Ver. 5.3 or later)	• 640 × 480 dot LCD • One unit of 3.5" FDD, one unit of 2.5" HDD (256 MB) • Ladder/instruction word program, monitor
Ladder software	JW-50SP (Ver. 5.3I or later)	• IBM-PC ladder software • Comes with an RS232C/RS422 converter • Ladder/instruction word program, monitor
	JW-52SP (Ver. 5.3 or later)	• DOS/V personal computer ladder software • Comes with an RS232C/RS422 converter • Ladder/instruction word program, monitor
	JW-92SP (Ver. 5.3 or later)	• PC-98 personal computer ladder software • Comes with a communication adapter • Ladder/instruction word program, monitor

- For more details about the operation methods of any model, see the instruction manual for a specific model.

(2) Models that allow you to use limited JW10 functions

Part name	Model name	Limited description
Hand-held programmer	JW-2PG JW-11PG JW-12PG JW-13PG (Without $\boxed{A}$, $\boxed{B}$ mark)	• Usable for the full range of JW20 functions. • The following JW10 functions and instructions (which the JW20 does not have) cannot be used. ① TMR/CNT instruction specified by register ② F-80 (byte specified I/O refresh) instruction ③ F-81 (bit specified I/O refresh) instruction
Ladder processor II	Z-100LP2F + Z-3LP2EM (Ver. 5.2 or later)	• Usable for the full range of JW20 functions. • The following JW10 functions and instructions (which the JW20 does not have) cannot be used. ① TMR/CNT instruction specified by register ② F-80 (byte specified I/O refresh) instruction ③ F-81 (bit specified I/O refresh) instruction • Model setting: JW22CU
Multipurpose programmer	JW-50PG (Ver. 5.2 or earlier)	
Ladder software	JW-50SP (Ver. 5.2I or earlier) JW-52SP (Ver. 5.2 or earlier) JW-92SP (Ver. 5.2 or earlier)	

Precautions for models with limited JW10 functions

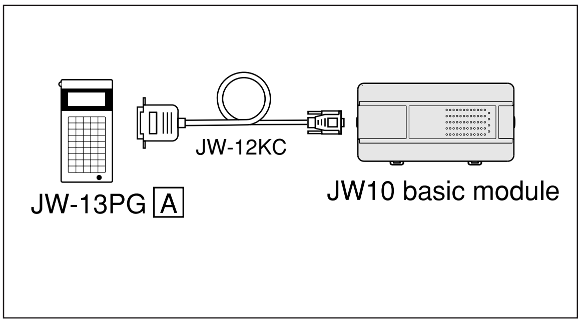
- When using a model with limited JW10 functions, care should be taken to note the following restrictions.

Model name	Precautions
JW-2PG JW-11PG JW-12PG JW-13PG (without $\boxed{A}$, $\boxed{B}$ mark)	• **Since the JW10 looks upon 7.5K words of the JW22's (JW-22CU) memory, pay attention to the following.** ① During programming, be sure to write the F-40 (END) instruction in the final program address (JW-1324K/1342K: 02777, JW-1424K/1442K/1624K/1642K: 07777). ② If data addresses (49000, E0000, etc.) which the JW22 has but the JW10 does not have are included in the program, a memory error will occur, and no operation will be executed. ③ The data memory which the JW22 has but the JW10 does not have will always be read as 0. The current value of those addresses cannot be changed. ④ When instructions are used (F-05, F-06, etc.) which the JW22 has but the JW10 does not have, UNDEFIND will be displayed. If the system is operated in this state, memory errors will occur, and no operation will be executed. ⑤ When the system is connected to the JW10, an error message will be displayed, however, the contents of the error message may vary. (If an abnormality is encountered, check the contents of the error message using the error code found at system memory #160.) ⑥ When all are initialized, ignore the □ shown on the right side of the LCD display. **Note: In order to use the JW-2PG with the JW10, set the JW10 system memory #136 to 02(H).**
Z-100LP2F + Z-3LP2EM (Ver. 5.2 or later)	• **Select the model type as 3.5K-words of JW22 (in the case of JW-1324K/1342K), or 7.5K-words of JW22(in this case of JW-1424K/1442K/1642K). Also, pay attention to the following.** ① When a program is written by PC transfer to the JW10 module, be sure that you have written an F-40 (END) instruction at the final program address (JW-1324K/1342K: 02777, JW-1424K/1442K/1624K/1642K: 07777). ② The timer/counter settings and the constants in the application instructions cannot be changed. ③ If an instruction exists in the ladder software after the final program address in the JW10, even though a program is written by PC transfer, it cannot be written to the JW10 module. Then, when the program is checked, a check error will occur. However, it is not related to the JW10 module's operation. ④ When an instruction or data memory address which the JW10 does not have is written by PC transfer, a memory error will occur. ⑤ The data memory which the JW22 has but the JW10 does not have will always be read as 0. The current value of those addresses cannot be changed. ⑥ It is not possible to read or write to the EEPROM. ⑦ ROM which is written to using the PROM programmer transfer function cannot be used. ⑧ A comment will be displayed in the system memory setting screen. Since it is a comment intended for the JW22, it may be different from the JW10 contents. ⑨ Do not execute "the CU memory clear."
JW-50PG (Ver. 5.2 or earlier)	
JW-50SP (Ver. 5.2I or earlier) JW-52SP (Ver. 5.2 or earlier) JW-92SP (Ver. 5.2 or earlier)	

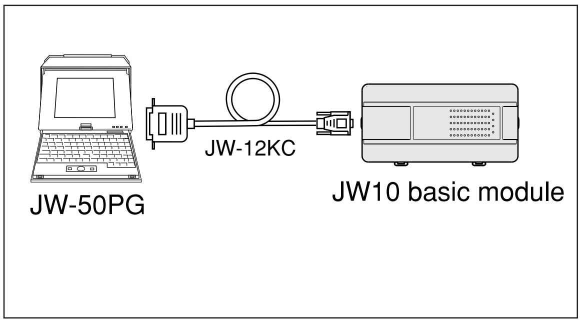
[2] Support tool connections

- Connect support tools to the MMI port on the JW10 module using the PG connecting cable (JW-12KC).

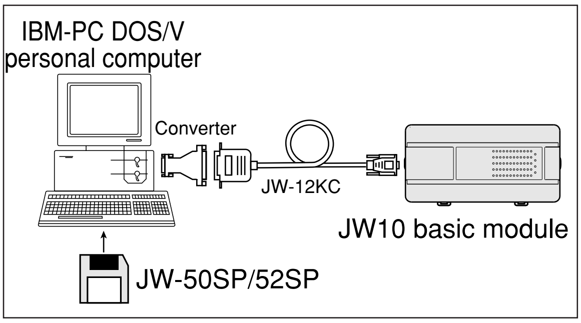
① Hand-held programmer (JW-13PG)



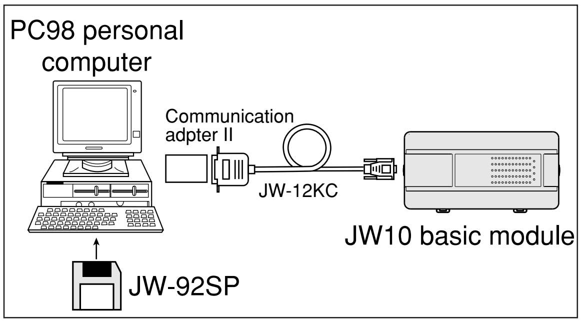
② Multipurpose programmer (JW-50PG)



③ Ladder software (JW-50SP/52SP)



④ Ladder software (JW-92SP)



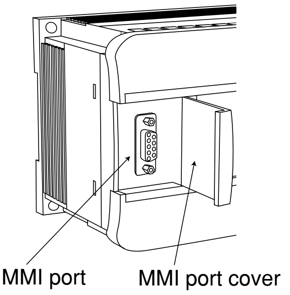
■ Procedure for connecting the PG connection cable (JW-12KC) and the JW10

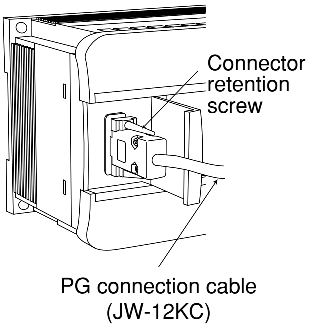
①Open the MMI port cover on the basic module.

②Connect the 9-pin connector of the JW-12KC cable to the MMI port on the basic module.

③Secure it with the connector screws.

JW-1324K



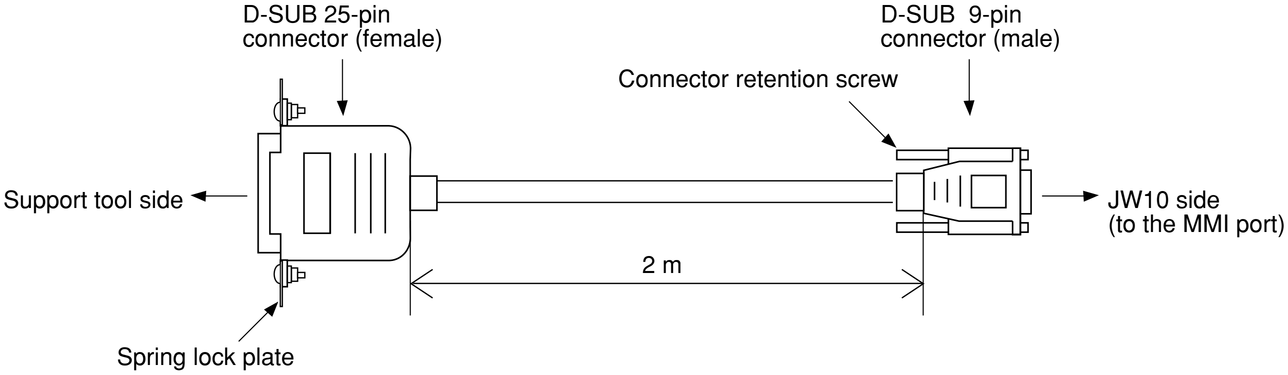


■ Communication specifications

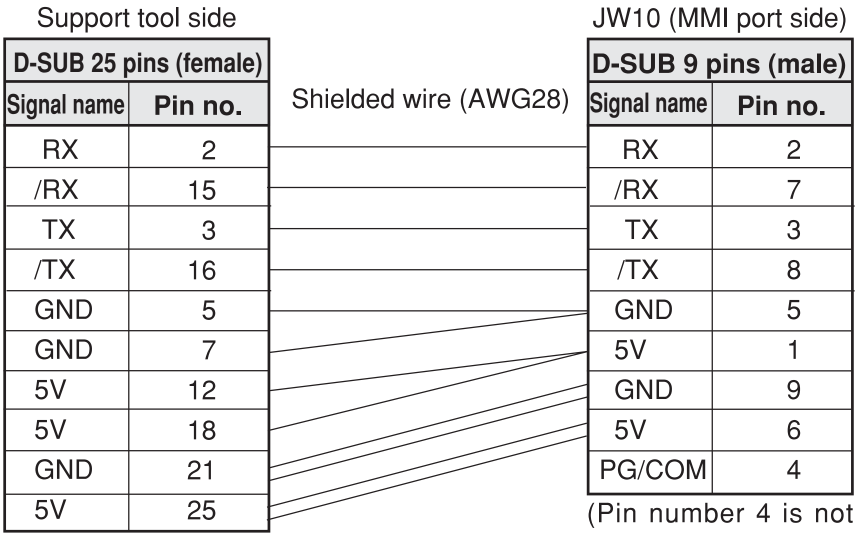
Item	Specifications
Data transfer standard	EIA RS-422A or equivalent
Transfer rate	19200 bits/s

■ PG connection cable (JW-12KC) specifications

1) External view


D-SUB 25-pin connector (female)
D-SUB 9-pin connector (male)
Connector retention screw
Support tool side
JW10 side (to the MMI port)
2 m
Spring lock plate

2) Wiring diagram



Support tool side

D-SUB 25 pins (female)	
Signal name	Pin no.
RX	2
/RX	15
TX	3
/TX	16
GND	5
GND	7
5V	12
5V	18
GND	21
5V	25

(The other pins are not connected.)

JW10 (MMI port side)

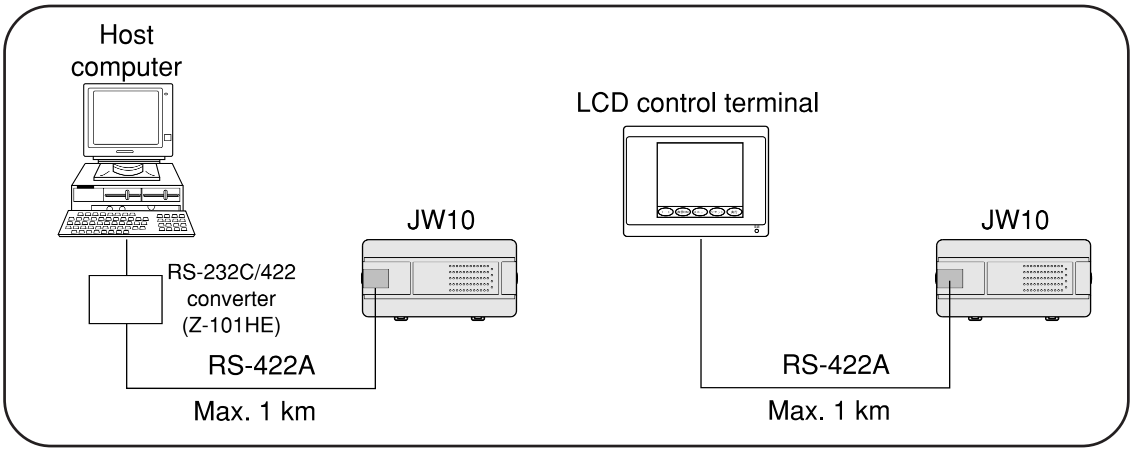
D-SUB 9 pins (male)	
Signal name	Pin no.
RX	2
/RX	7
TX	3
/TX	8
GND	5
5V	1
GND	9
5V	6
PG/COM	4

(Pin number 4 is not connected.)

14-3 Computer link mode

[1] Communication specifications

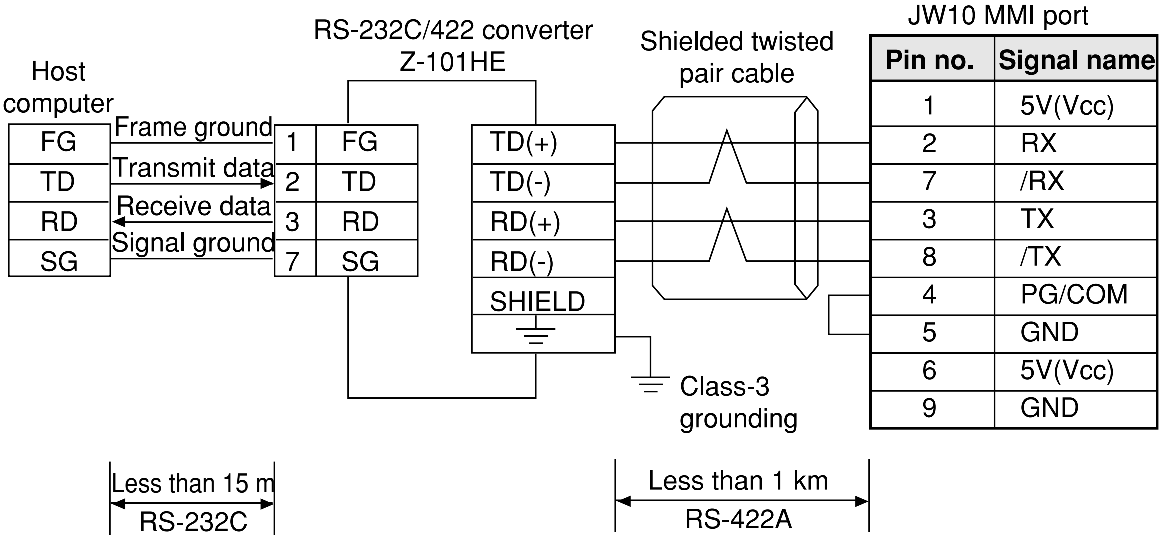
- The JW10 can communicate with a single host computer or a single LCD control terminal.



Item	Specifications
Data transfer standard	EIA RS-422A or equivalent. Start-stop synchronous system
Transfer rate	38400, 19200, 9600, 4800, 2400, 1200, 600, 300 bits/s
Data style	Start bit : 1 bit Data length : 7 or 8 bits Parity bit : 1 bit (odd, even, none) Stop bit : 1 or 2 bits
Characters used	ASCII alphanumeric characters
Error check	Parity check, sum check
Number of stations connected	1 module [The station number is fixed at 001(8).]
Communication line	Shielded twisted pair cable. Cable total length: 1 km 4-wire system

[2] Wiring

(1) When using an RS-232C/422 converter



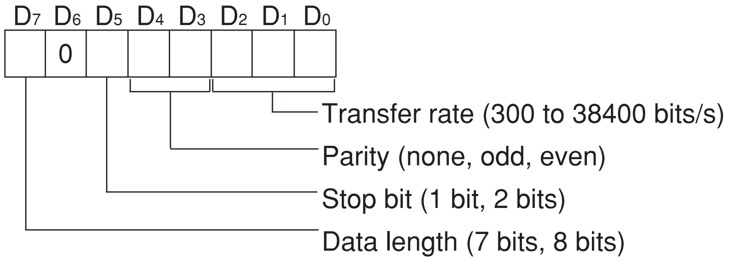
Pin no.	Signal name
1	5V(Vcc)
2	RX
7	/RX
3	TX
8	/TX
4	PG/COM
5	GND
6	5V(Vcc)
9	GND

(2) When connecting the JW10 to an LCD control terminal

See "Appendix-5. Connection with an LCD control terminal."

[3] Setting system memory

- When using the computer link mode, use the following system memory settings.

<table>
<tr><td>#226</td><td>Transfer specifications</td><td>


<table>
<tr><th>D7</th><th>Data length</th></tr>
<tr><td>0</td><td>7 bits</td></tr>
<tr><td>1</td><td>8 bits</td></tr>
</table>
<table>
<tr><th>D5</th><th>Stop bit</th></tr>
<tr><td>0</td><td>1 bit</td></tr>
<tr><td>1</td><td>2 bits</td></tr>
</table>
<table>
<tr><th>D4</th><th>D3</th><th>Parity</th></tr>
<tr><td>0</td><td>0</td><td>None</td></tr>
<tr><td>0</td><td>1</td><td>Odd</td></tr>
<tr><td>1</td><td>0</td><td>Even</td></tr>
<tr><td>1</td><td>1</td><td>Disable</td></tr>
</table>
<table>
<tr><th>D2</th><th>D1</th><th>D0</th><th>Transfer rate (bit/s)</th></tr>
<tr><td>1</td><td>1</td><td>1</td><td>38400</td></tr>
<tr><td>0</td><td>0</td><td>0</td><td>19200</td></tr>
<tr><td>0</td><td>0</td><td>1</td><td>9600</td></tr>
<tr><td>0</td><td>1</td><td>0</td><td>4800</td></tr>
<tr><td>0</td><td>1</td><td>1</td><td>2400</td></tr>
<tr><td>1</td><td>0</td><td>0</td><td>1200</td></tr>
<tr><td>1</td><td>0</td><td>1</td><td>600</td></tr>
<tr><td>1</td><td>1</td><td>0</td><td>300</td></tr>
</table>
</td></tr>
<tr><td>#227</td><td>Setting computer link station number for MMI port</td><td>• Set own station's number of the MMI port into 001(8).
• As the MMI port connection is only available as 1 vs 1, station number of own station should only be "001(8)."
• Initial value is 000(8).</td></tr>
</table>

[4] Communication requirements

- They are the same as for the computer link on the communication port. See "13-2 Computer link."

Note 1: Since only one-by-one connection is applied, the AD(H) and AD(L) (slave station address) in the communication format must be "01."

Chapter.15 Analog input/output module

15-1 Outline

[1] JW-14AD

- The analog input module JW-14AD converts an external analog input signal (voltage/current) to 11-bit/12-bit binary data.
- 4 channels can be input.

[2] JW-12DA

- The analog output module JW-12DA converts a 11-bit/12-bit binary data to an external analog output signal (voltage/current).
- 2 channels can be output.

Notes

★When you use JW-14AD/JW-12DA, please use the JW10 basic module(JW-1424K/1442K/1624K/1642K) of which version is 2.0 or more.

★A JW-14AD and a JW-12DA can be connected to above mentioned basic module respectively.

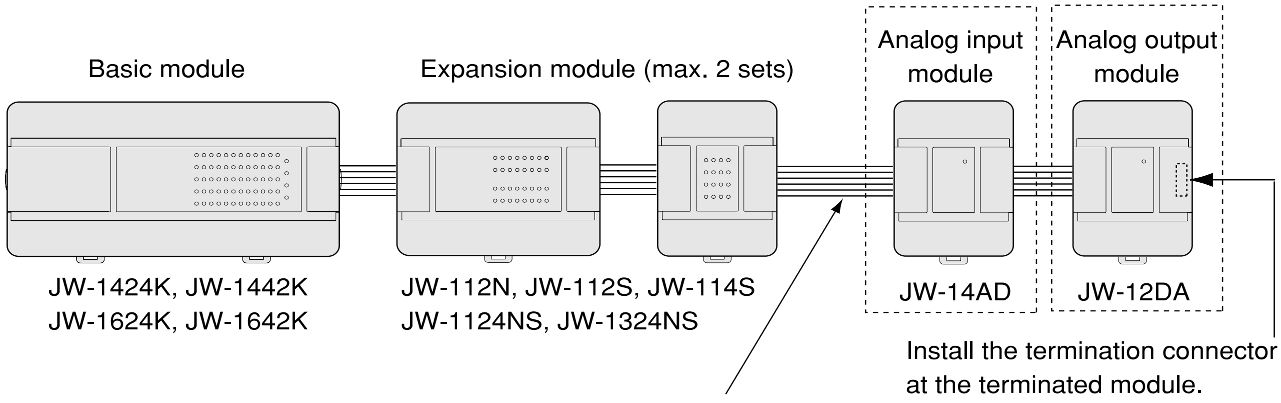
★External power supply (24VDC) is required.

★Set system memory for analog modules. (JW-14AD at #210,#211, JW-12DA at #212)

★The analog modules occupy special I/O register (JW-14AD at コ0200 to コ0207, JW-12DA at コ0240 to コ0243), but input/output relays are not occupied.

★The offset and gain adjustment of analog modules (JW-14AD/JW-12DA) are adjusted at shipping and no need to adjust at customer's side. When overall adjustment considered wiring resistance is required, adjust them at external circuit.

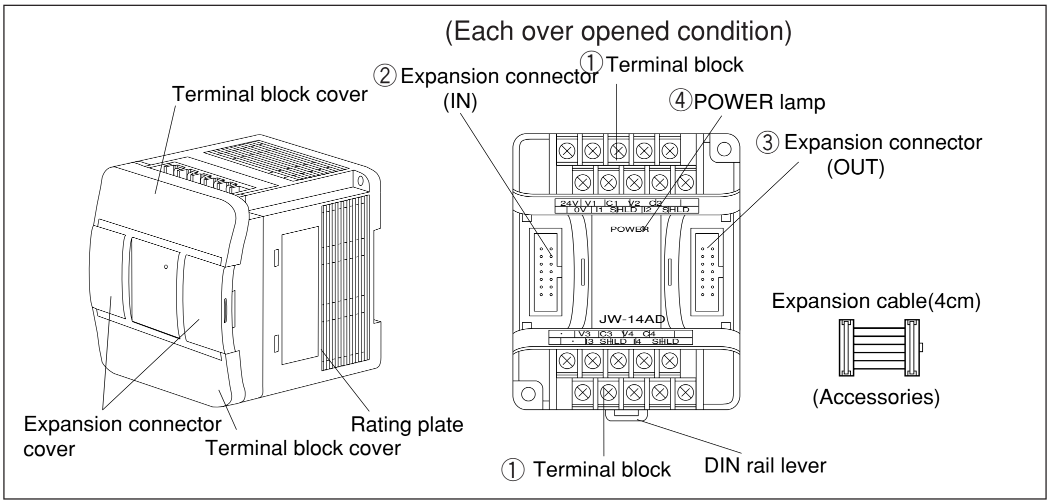
■ System configuration



One expansion cable JW-104EC(40cm) can be used at one system.
For others, use the cable (4cm) attached each expansion/analog input/analog output modules.

15-2 Name and function of each part

[1] Analog input module (JW-14AD)



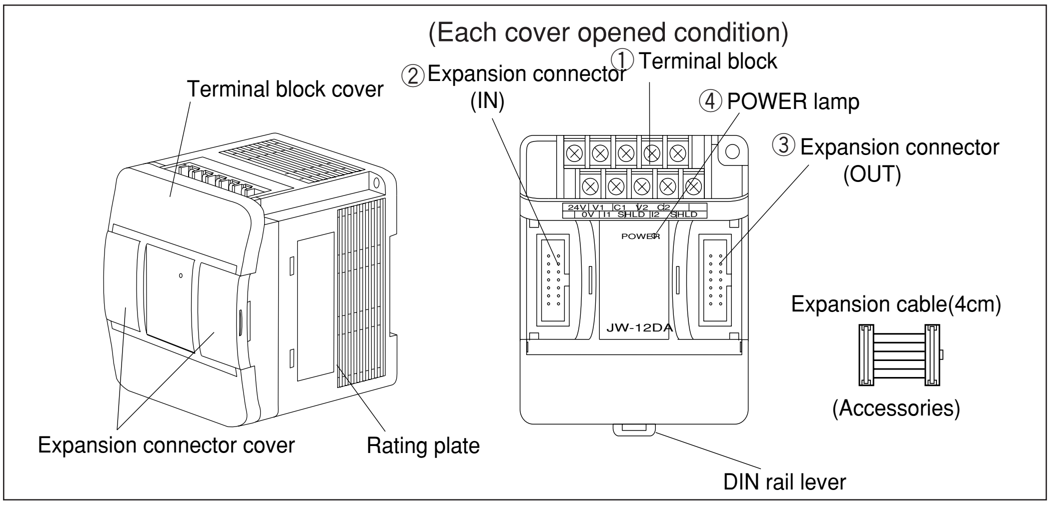
① Terminal block : Connects analog input signal line and power line.

② Expansion connector (IN) : Connect with expansion connector (OUT) at previous module(Basic /expansion /analog output module) by using expansion cable (use accessory or JW-104EC).

③ Expansion connector(OUT) : Connect with expansion connector (IN) at next module (Expansion/analog output module) by using expansion cable (use accessory or JW-104EC).Install the terminating connector when the analog input modules is the terminated module.

④ POWER lamp : Green LED is turned on when internal 5VDC is supplied normally.

[2] Analog output module (JW-12DA)



① Terminal block : Connects analog output signal line and power line.

② Expansion connector (IN) : Connect with expansion connector (OUT) at previous module(basic /expansion /analog input module) by using expansion cable (use accessory or JW-104EC).

③ Expansion connector(OUT) : Connect with expansion connector (IN) at next module (expansion/analog input module) by using expansion cable (use accessory or JW- 104EC).Install the terminating connector when the analog input modules is the terminated module.

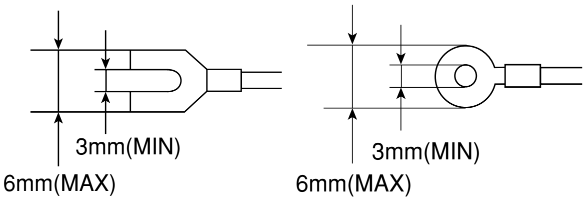
④ POWER lamp : Green LED is turned on when internal 5VDC is supplied normally.

15-3 Wiring

(1) Separate signal lines and power line of the module from high voltage lines and power lines as far as possible. Do not run signal lines and power lines in parallel with high voltage or power lines.

(2) M3 terminal screws are applied for either of terminal screws.Use crimp-style terminals equiva lent to JIS standard 1.25-3, and securely fix with tightening torque 4 to 8 kgf-cm.

【Recommended crimp-style terminals】
JAPAN SOLDERLESS TERMINAL
MFG. CO., LTD
Model name :1.25-B3A 1.25-C3A
1.25-3 1.25-MS3



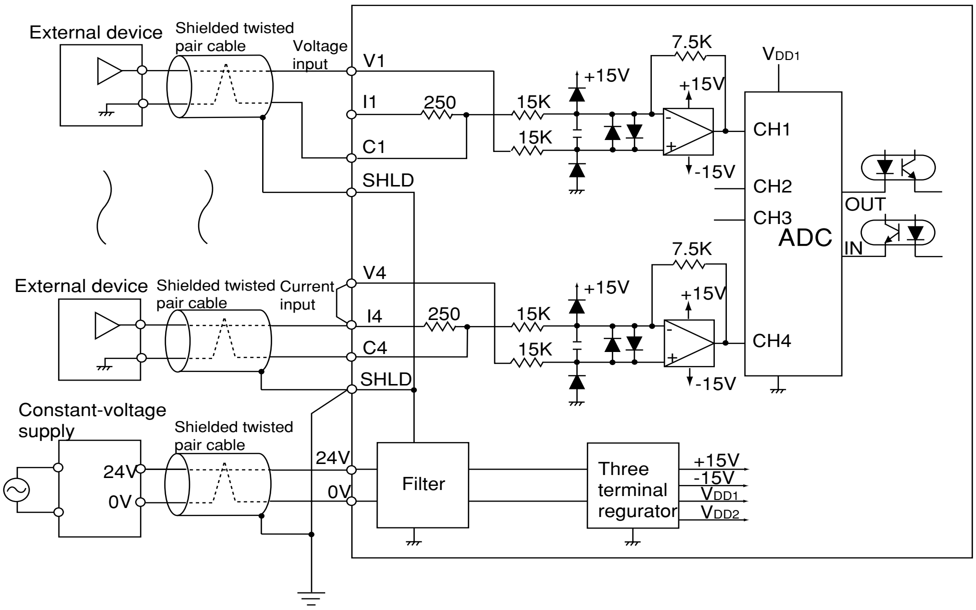
(3) Use the shieled twisted pair when wiring to the each terminal.
Use the cable given below or its equivalent for the shielded twisted pair cable.

HITACHI cable : CO-SPEV-SB (A) 0.5mm^2

[1] JW-14AD

Upper terminal	24V	V1	C1	V2	C2		
		0V	I1	SHLD	I2	SHLD	
Lower terminal	•	V3	C3	V4	C4		
		•	I3	SHLD	I4	SHLD	

Terminal				Contents
CH1	CH2	CH3	CH4	
V1	V2	V3	V4	Voltage input terminal Input voltage signal to terminal V and C for each channels (CH1 to 4) respectively.
C1	C2	C3	C4	Common terminal
I1	I2	I3	I4	Electric current terminal Input current signal to terminal I and C for each channels (CH1 to 4) and connect between terminal V and I respectively.
SHLD	SHLD	SHLD	SHLD	Shield terminal (SHLD of each channel are connected each other at internal circuit) Connects to flame ground (FG).
24V				24 VDC power supply input terminal (+)
0V				24 VDC power supply input terminal (−)



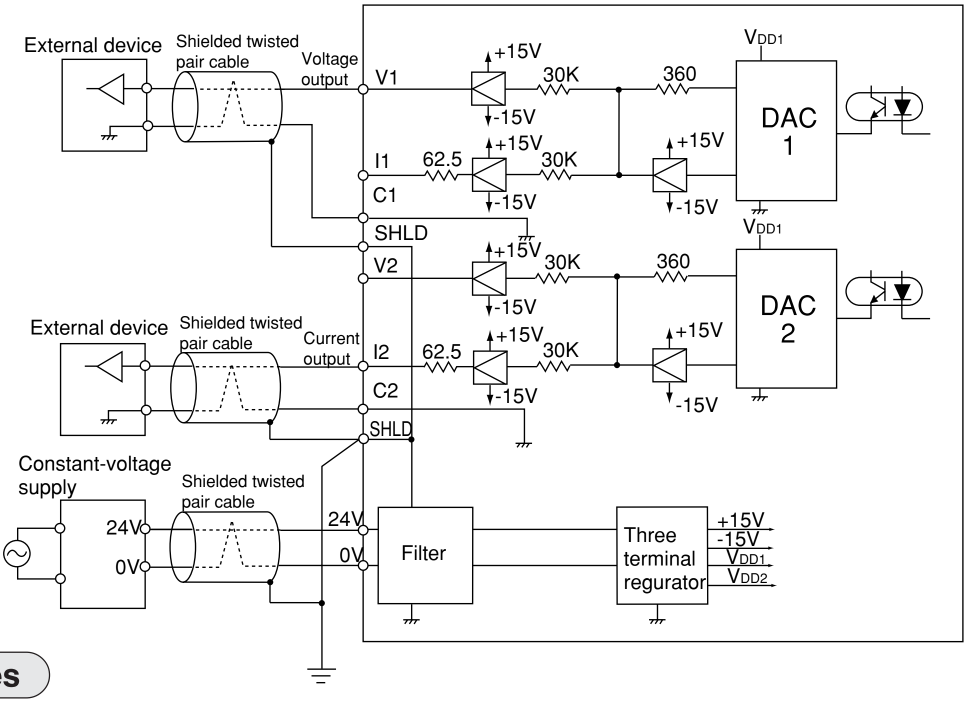
Notes

- Keep the wire from the shield as short as possible (30mm or less).
- Connect shield line to SHLD terminal and flame ground (FG) using twisted wire of apx. 1.25mm^2 via SHLD terminal.
- Use this module dedicated constant-voltage supply or 24VDC power supply of basic module for the 24VDC input power supply.
- Short the I terminal and V terminal for current input.

[2] JW-12DA

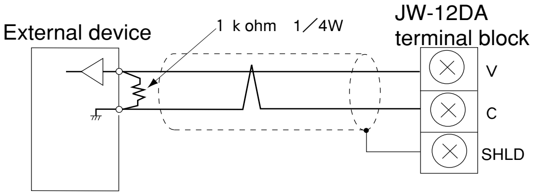
Upper terminal	24V	V1	C1	V2	C2	
		0V	I1	SHLD	I2	SHLD

Terminal		Contents
CH1	CH2	
V1	V2	Voltage output terminal Output voltage signal to terminal V and C for each channels (CH1 to 4) respectively.
C1	C2	Common terminal (C1 and C2 are connected each other at internal circuit)
I1	I2	Electric current terminal Output current signal to terminal I and C for each channels (CH1 to 4) respectively.
SHLD	SHLD	Shield terminal (SHLD of SH1 and SH2 are connected each other at internal circuit) Connects to flame ground (FG).
24V		24 VDC power supply input terminal (+)
0V		24 VDC power supply input terminal (−)

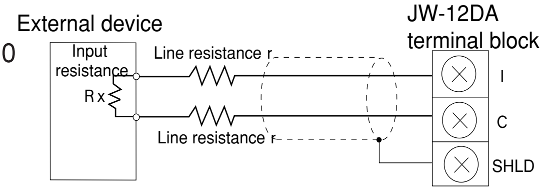


Notes

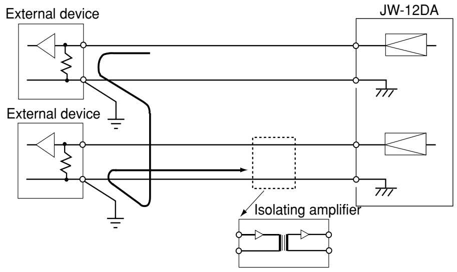
- Keep the wire from the shield as short as possible (30mm or less).
- Connect shield line to SHLD terminal and flame ground (FG) using twisted wire of apx. 1.25mm^2 via SHLD terminal.
- Use this module dedicated constant-voltage supply or 24VDC power supply of basic module for the 24VDC input power supply.
- If the input impedance of the external device is high, the effect of AC induction becomes larger for voltage inputs. In this case, connect a load resistor of about 1k ohm across the input terminals of the external device.



- The maximum load resistance of the current output is 500 ohm. If the sum of the input resistance of the external device and the line resistance exceeds 500 ohm, the linear accuracy of the output current deteriorates.



- If the 0V terminals of external devices are connected, a circuit will form through the outputs for CH1 and CH2. If the external devices are affected as a result, either provide a commercially available isolating amplifier for one channel.



15-4 How to use JW-14AD

[1] Operation mode

Select one of the below three modes by setting system memory #210.

#210(HEX)		Operation mode	Analog input		Digital value
Set value	Initial value		Voltage input	Current input	
01		Mode 1	0 to10VDC	—	0 to 4000 (12 bits binary)
02		Mode 2	0 to 5VDC	0 to 20mADC	0 to 2000 (11 bits binary)
03		Mode 3	1 to 5VDC	4 to 20mADC	0 to 2000 (11 bits binary)
00	○	JW-14AD does not work. (can not convert analog to digital)			

Note : JW-14AD has 4 channels, but above setting is applied 4 channels all.

[2] Allocation of data memory

The converted digital data from analog data are stored at special register for JW-14AD.

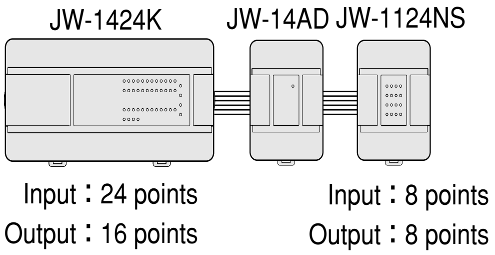
Channel	Special register for JW-14AD								
	Byte address	D7	D6	D5	D4	D3	D2	D1	D0
CH1	コ0200	2^{7}	2^{6}	2^{5}	2^{4}	2^{3}	2^{2}	2^{1}	2^{0}
	コ0201	0	0	0	0	2^{11}	2^{10}	2^{9}	2^{8}
CH2	コ0202	2^{7}	2^{6}	2^{5}	2^{4}	2^{3}	2^{2}	2^{1}	2^{0}
	コ0203	0	0	0	0	2^{11}	2^{10}	2^{9}	2^{8}
CH3	コ0204	2^{7}	2^{6}	2^{5}	2^{4}	2^{3}	2^{2}	2^{1}	2^{0}
	コ0205	0	0	0	0	2^{11}	2^{10}	2^{9}	2^{8}
CH4	コ0206	2^{7}	2^{6}	2^{5}	2^{4}	2^{3}	2^{2}	2^{1}	2^{0}
	コ0207	0	0	0	0	2^{11}	2^{10}	2^{9}	2^{8}

The stored data is binary value, but you can handle them as decimal data by adding each bit's weight as blow shown. (0 to 4095)

2^{11}	2^{10}	2^{9}	2^{8}	2^{7}	2^{6}	2^{5}	2^{4}	2^{3}	2^{2}	2^{1}	2^{0}
2048	1024	512	256	128	64	32	16	8	4	2	1

JW-14AD does not occupy I/O relay area (00000 to 00777).

(Example)



Allocation of input/output relay number

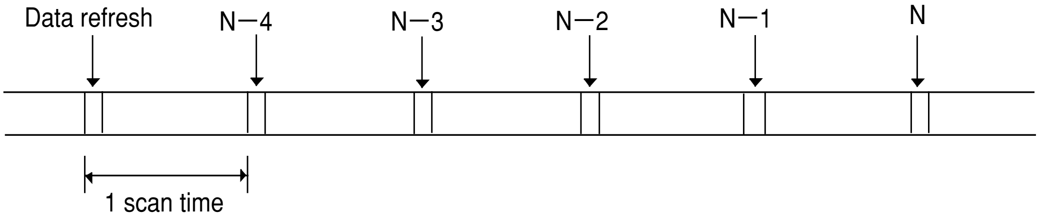
	Input relay	Output relay
JW-1424K	00000 to 00027	00400 to 00417
JW-14AD	—	—
JW-1124NS	00030 to 00037	00420 to 00427

[3] Averaging function

When input value change rapidly, averaging function average input values and outputs an average value in specific period.

It average prior five analog input data and get an average value following below procedure.

> Average data = Delete maximum and minimum value in prior five data and
> average remained three data.



(Example) N-4th. data = 161

N-3rd. data = 120 ———Minimum value

N-2nd. data = 154

N-1st. data = 160

N th. data = 190-———Maximum value

In case of above data

Average value at Nth. = (161+154+160)/3=158   (rounds to integral number)

The first four scanned data after operation start are output without averaging.

Set system memory #211 for averaging function as below

#211(HEX)		Contents
Set value	Initial value	
00	○	Averaging disable
01		Averaging enable

Note : JW-14AD has four channels, but above setting applies all channels.

[4] Module status in error

	PC status	Digital value (at コ0200 to コ0207)
External 24VDC power supply is disconnected	RUN	Maximum value(4095 at mode1, 2047 at mode 2 or 3)
PC stop	STOP	Holds prior data before PC stop
#210＝00	RUN	HOLD:JW-14AD does not execute (not convert analog to digital)
Except for #210＝00、01、02、03	STOP *Note	

*Note : Store error code 23(H) at system memory #160 to #167 and PC stop.

(23(H) : System memory setting error)

15-5 How to use JW-12DA

[1] Operation mode

Select one of the below three modes by setting system memory #212.

#212 (HEX)		Operation mode	Digital value	Analog output	
Set value	Initial value			Voltage output	Current output
01		Mode 1	0 to 4000 (12 bits binary)	0 to10VDC	—
02		Mode 2	0 to 2000 (11 bits binary)	0 to 5VDC	0 to 20mADC
03		Mode 3	0 to 2000 (11 bits binary)	1 to 5VDC	4 to 20mADC
00	○	JW-12DA does not work. (can not convert digital to analog)			

Note : JW-12DA has 2 channels, but above setting is applied 2 channels all.

[2] Allocation of data memory

The converted analog data from digital data are stored at special register for JW-12DA.

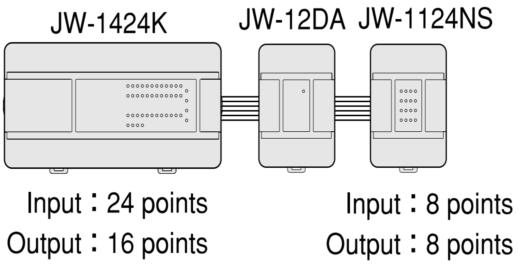
Channel	Special register for JW-12DA								
	Byte address	D7	D6	D5	D4	D3	D2	D1	D0
CH1	コ0240	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
	コ0241	0	0	0	0	2^{11}	2^{10}	2^9	2^8
CH2	コ0242	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
	コ0243	0	0	0	0	2^{11}	2^{10}	2^9	2^8

The stored data is binary value, but you can handle them as decimal data by adding each bit's weight as blow shown.(0 to 4095)

2^{11}	2^{10}	2^9	2^8	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
2048	1024	512	256	128	64	32	16	8	4	2	1

JW-14AD does not occupy I/O relay area (00000 to 00777).

(Example)



Allocation of input/output relay number

	Input relay	Output relay
JW-1424K	00000 to 00027	00400 to 00417
JW-12DA	—	—
JW-1124NS	00030 to 00037	00420 to 00427

[3] Module status in error

	PC status	Analog output value(between terminal V and C or I and C)
External 24VDC power supply is disconnected	RUN	0V or 0mA
PC stop	STOP	Holds prior output value before PC stop *Note 1
#212＝00	RUN	HOLD:JW-12DA does not execute (not convert digital to analog)
Except for #212＝00、01、02、03	STOP *Note 2	

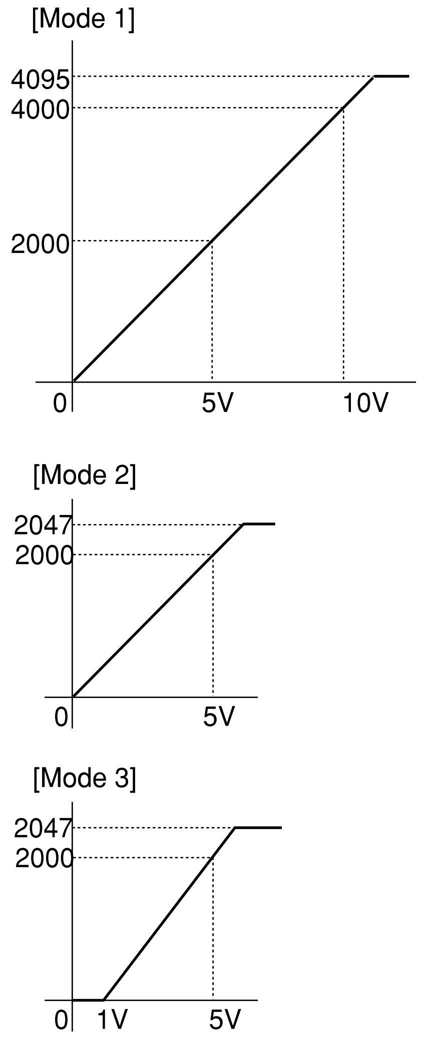
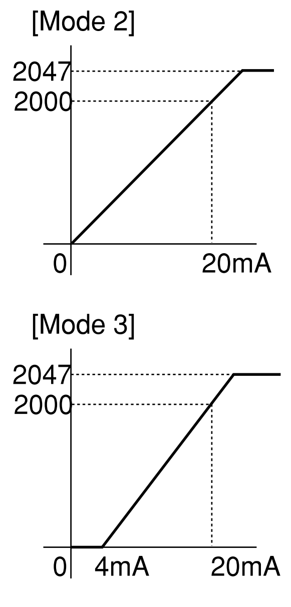
*Note 1 : Holds output value when system memory #206=0 (reset output when PC stop).

*Note 2 : Store error code 23(H) at system memory #160 to #167 and PC stop.

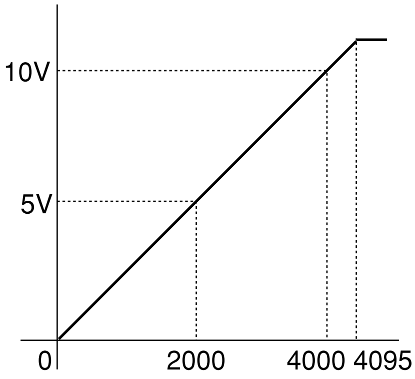
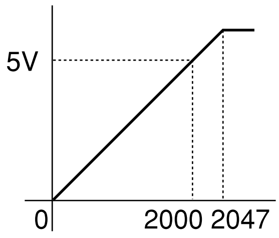
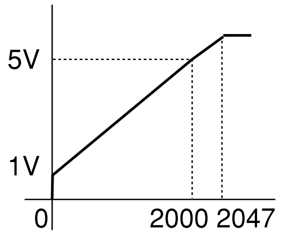
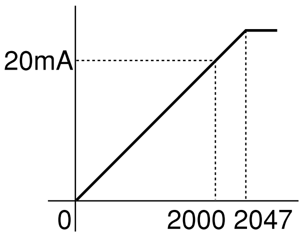
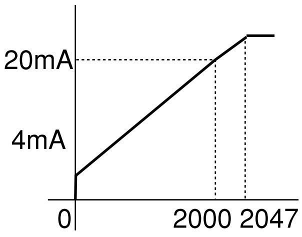
(23(H) : System memory setting error)

15-6 Specifications

[1] JW-14AD Performance specifications

Item	Specifications	
	Volatge input	Current input
No. of input channel	4 channels	
Analog input range	0 to 10VDC	0 to 20mADC
Absolute maximum input signal	−1V, +15V	0mA, +25mA
Input impedance	30k ohm	250 ohm
Digital output	0 to 4000(12 bits binary value)	0 to 2000(11 bits binary value)
Resolution	2.5mV (10V/4000)	10 μ A (20mA/2000)
Overall accuracy	±0.5% or less (full scale at 25℃) ±1% or less (full scale at 0 to 55℃)	
A/D conversion time	1 scan time/ 4 channels	
I/O characteristics		
Opearaion display	POWER lamp(Green LED is turned on when internal 5VDC is supplied normally.)	
External power supply	24VDC±10%(includes ripple) Spike noise 50mVP-P or less (max.100mA)	
Insulation system	Photocoupler isolation (Each channels are not isolated.)	
Insulation resistance	500VDC for 10M ohm or more (between input terminal and secondary circuit)	
Dielectrical strength	500VAC for one minute (between input terminal and secondary circuit)	

[2] JW-12DA Performance specifications

Item	Specifications	
	Voltage output	Current output
No.of output channels	2 channels	
Analog output range	0 to 10VDC	0 to 20mADC
External load resistance	500 ohm or more	500 ohm or less
Digital input	0 to 4000 (12 bits binary value)	0 to 2000 (11 bits binary value)
Resolution	2.5mV (10V/4000)	10 μA (20mA/2000)
Overall accuracy	±0.5% or less (full scale at 25℃) ±1% or less (full scale at 0 to 55℃)	
D/A conversion time	1 scan time/ 2 channels	
I/O characteristics	[Mode 1]  [Mode 2]  [Mode 3]  	[Mode 2]  [Mode 3] 
Operation display	POWER lamp(Green LED is turned on when internal 5VDC is supplied normally.)	
External power supply	24VDC±10%(includes ripple) Spike noise 50mVP-P or less (max.200mA)	
Insulation system	Photocoupler isolation (Each channels are not isolated.)	
Insulation resistance	500VDC for 10M ohm or more (between output terminal and secondary circuit)	
Dielectrical strength	500VAC for one minute (between output terminal and secondary circuit)	

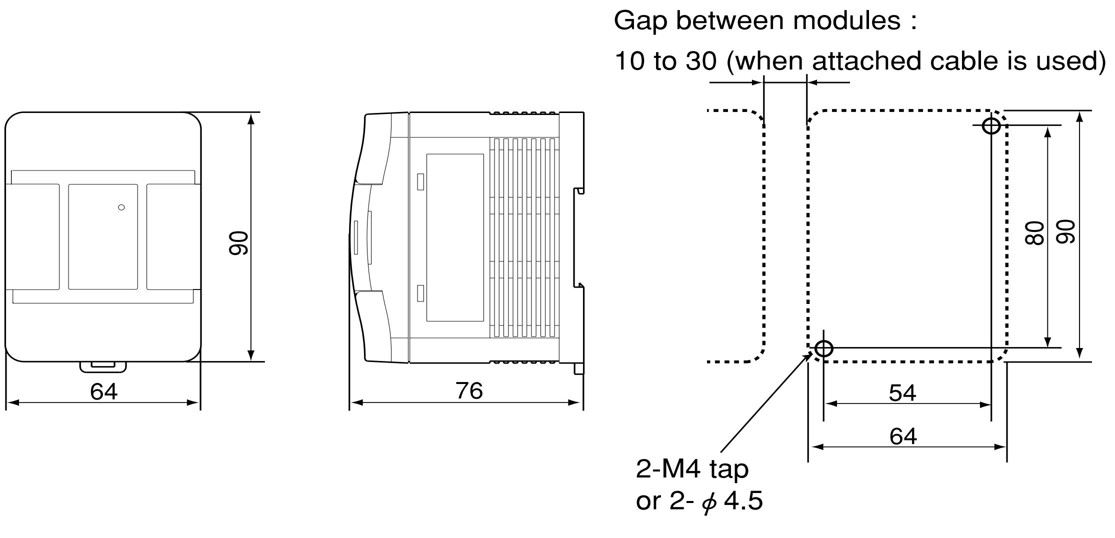
[3] General specifications

Item	Specifications	
	JW-14AD	JW-12DA
Storage temperature	-25 to 70℃	
Ambient temperature	0 to 55℃	
Ambient humidity	5 to 90％RH(non-condensing)	
Vibration resistance	JIS C 0911 or equivalent , Amplitude 0.15mm(10 to 58Hz), 1G(58 to 150Hz) (2 hours in each of X,Y and Z axis)	
Shock resistance	JIS C 0912 or equivalent 15G (3 times in each X, Y and Z axis)	
External wire grounding system	Connect to a terminal block(M3×7mm self lockup screw) Applied with crimp-style terminal : JIS 1.25-3 or equivalent	
Installation	Direct installation or using DIN rail (35mm width)	
Outline dimension	64mm(W)×90mm(H)×76mm(D)	
Weight	Approx. 180g	Approx. 195g
Grounding	Class-3 grounding	
Accessories	Instruction manual × 1 Expansion cable(4cm) × 1	

[4] Outline dimension drawings (JW-14AD and JW-12DA in common)

[Unit : mm]

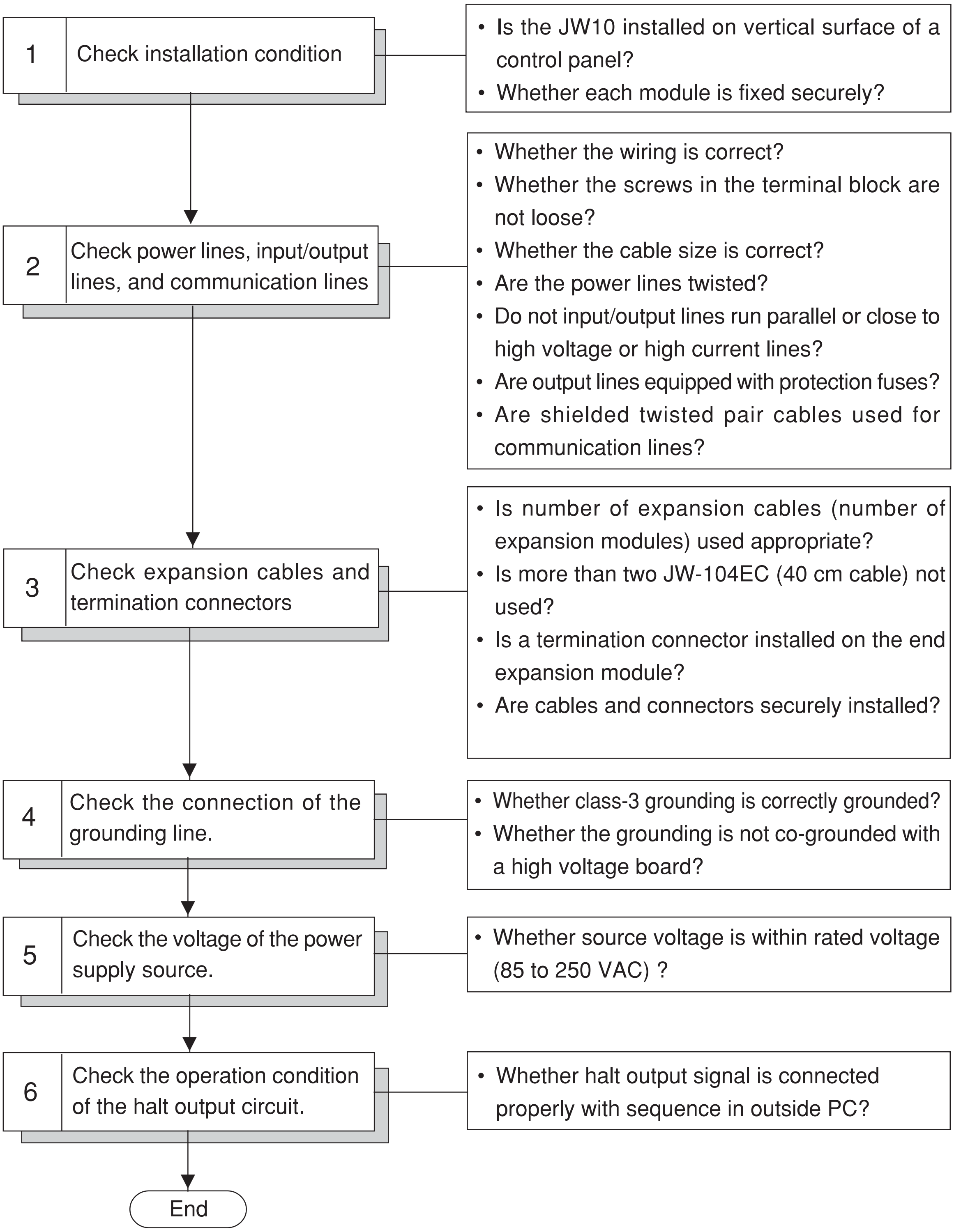
(Mounting dimensions to panel)



Chapter 16. Trial Run

16-1 Check points prior to trial run

When installing and wiring are completed, check the following points, prior to supplying power to the JW10.



1 Check installation condition

- Is the JW10 installed on vertical surface of a control panel?
- Whether each module is fixed securely?

2 Check power lines, input/output lines, and communication lines

- Whether the wiring is correct?
- Whether the screws in the terminal block are not loose?
- Whether the cable size is correct?
- Are the power lines twisted?
- Do not input/output lines run parallel or close to high voltage or high current lines?
- Are output lines equipped with protection fuses?
- Are shielded twisted pair cables used for communication lines?

3 Check expansion cables and termination connectors

- Is number of expansion cables (number of expansion modules) used appropriate?
- Is more than two JW-104EC (40 cm cable) not used?
- Is a termination connector installed on the end expansion module?
- Are cables and connectors securely installed?

4 Check the connection of the grounding line.

- Whether class-3 grounding is correctly grounded?
- Whether the grounding is not co-grounded with a high voltage board?

5 Check the voltage of the power supply source.

- Whether source voltage is within rated voltage (85 to 250 VAC) ?

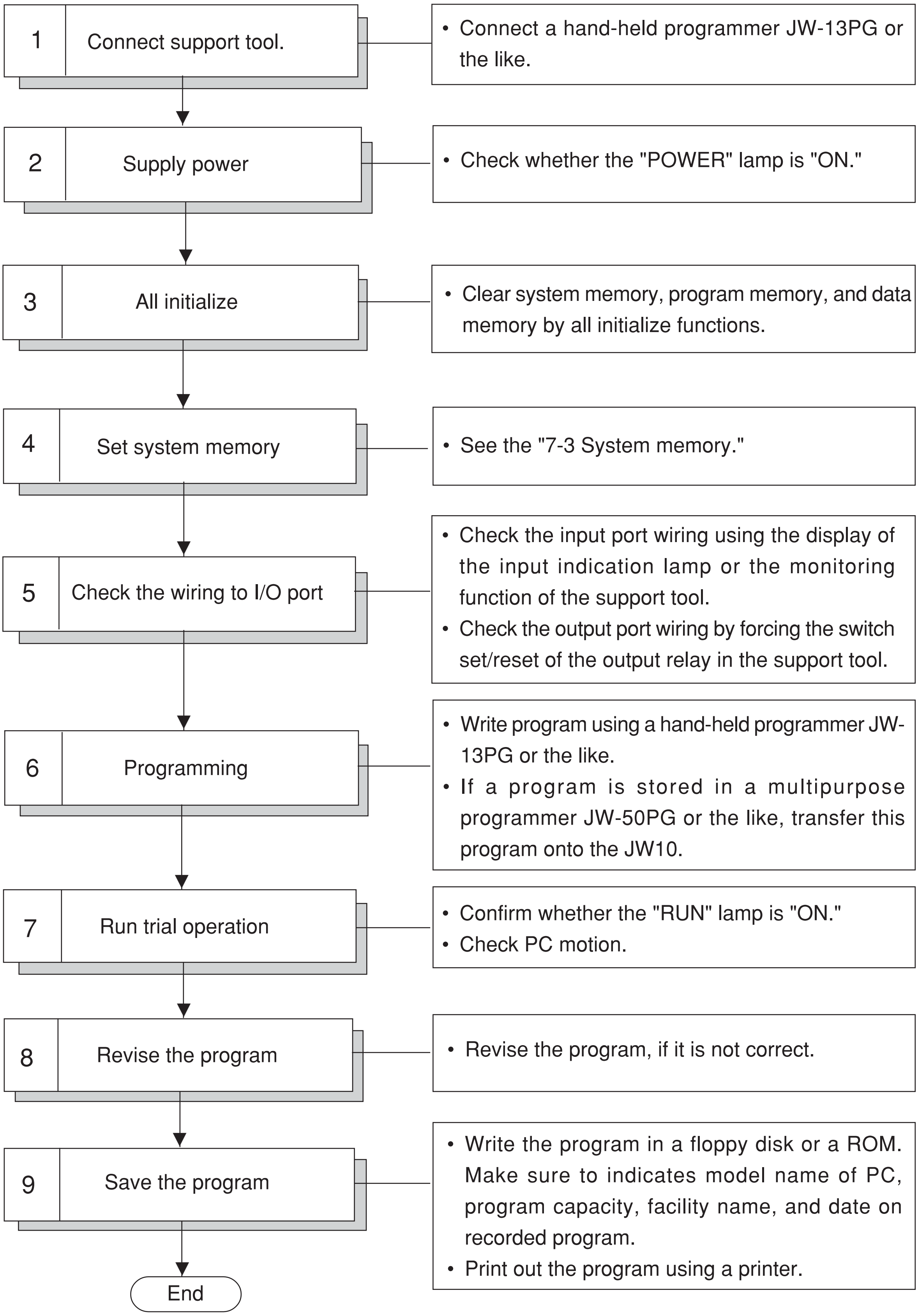
6 Check the operation condition of the halt output circuit.

- Whether halt output signal is connected properly with sequence in outside PC?

End

16-2 Operation method for trial run

Operate the JW10 according to the following process after precautions for operation is checked.



1 Connect support tool.

- Connect a hand-held programmer JW-13PG or the like.

2 Supply power

- Check whether the "POWER" lamp is "ON."

3 All initialize

- Clear system memory, program memory, and data memory by all initialize functions.

4 Set system memory

- See the "7-3 System memory."

5 Check the wiring to I/O port

- Check the input port wiring using the display of the input indication lamp or the monitoring function of the support tool.
- Check the output port wiring by forcing the switch set/reset of the output relay in the support tool.

6 Programming

- Write program using a hand-held programmer JW-13PG or the like.
- If a program is stored in a multipurpose programmer JW-50PG or the like, transfer this program onto the JW10.

7 Run trial operation

- Confirm whether the "RUN" lamp is "ON."
- Check PC motion.

8 Revise the program

- Revise the program, if it is not correct.

9 Save the program

- Write the program in a floppy disk or a ROM. Make sure to indicates model name of PC, program capacity, facility name, and date on recorded program.
- Print out the program using a printer.

Chapter 17. Maintenance and Check

17-1 Periodical check

The following table is the recommended periodical check of used items, so as to keep the JW10 operating normally and in the best condition:

■ General items

Check items	Check contents	Standard	Remarks
Ambient temperature	Within the specifications (Temperature in the control box becomes the ambient temperature when the JW10 is installed in a control box.)	0 to 55˚C	
Ambient humidity		5 to 90% RH	No condensation
Atmosphere		No corrosive gas, etc.	
Vibration		No	
Shock		No	

■ Basic module

Check items	Check contents	Standard	Remarks
Power voltage	Measure input voltage at the terminal block and check that it is within the specifications.	85 to 250 VAC	
Power voltage 24 VDC for sensors	Measure output voltage on power terminals for sensors, and check whether it is within the standard.	24 VDC ±10%	
Input port power	Voltage supplied at cable to input port is with the specifications.	10 to 26.4 VDC	Terminal 0 to 3
		20 to 26.4 VDC	Terminal 4 to
Output port power	Voltage supplied at cable to output port is with the specifications.	250 VAC max. 30 VDC max.	JW-1324K JW-1424K JW-1624K
		4.5 to 27VDC	JW-1342K JW-1442K JW-1642K
Error lamp of basic module	Visually check error lamp (ERR)	Light OFF	
Installed condition	The basic module is fixed firmly.	No looseness	
	Terminal block screws have not been loosened.	No looseness	
	Are expansion cables and a termination connector securely installed on expansion connector?	Should be securely installed	JW-1324K/1342K are unnecessary

■ Expansion module

Check items	Check contents	Standard	Remarks
Input port power	Voltage supplied at cable to input port is with the specifications.	20 to 26.4 VDC	JW-112N JW-1124NS JW-1324NS
Output port power	Voltage supplied at cable to output port is with the specifications.	250 VAC max. 30 VDC max.	JW-114S JW-1124NS JW-1324NS
		4.5 to 27VDC	JW-112S
Installed condition	The expansion module is fixed firmly.	No looseness	
	Terminal block screws have not been loosened.	No looseness	
	Are expansion cables and a termination connector securely installed on expansion connector?	Should be securely installed	

■ Analog input module/ analog output module

Check items	Check contents	Standard	Remarks
24 VDC power voltage	Measure output voltage on 24 VDC power terminal, and check whether it is within the standard.	24 VDC ±10%	
Installed condition	The analog input module, analog output module are fixed firmly.	No looseness	
	Terminal block screws have not been loosened.	No looseness	
	Are expansion cables and a termination connector securely installed on expansion connector?	Should be securely installed	

■ Others

Verify that the program stored in a ROM (EPROM or EEPROM) or a floppy diskette (FD) is the same as the currently operating program.

17-2 Troubleshooting

In the event of abnormality, check the lamp (RUN, ERR) of the basic module, and remedy according to the check flow depending on the state.

[1] State of LED

RUN	ERR	Remarks	
OFF ○	ON ●	Detectable error for self-diagnosis	→ Check flow 1
OFF ○	OFF ○	Power supply OFF	→ Check flow 2
Blink ◎	OFF ○	Halt mode	→ Check flow 3
ON ●	OFF ○	Disable detection error by self-diagnosis (input relation)	→ Check flow 4
		Disable detection error by self-diagnosis (output relation)	→ Check flow 5
ON ●	ON ●	Others	→ Check flow 1

[2] Precondition of check flow

This check flow describes the countermeasure method (replacement of defective module and subsequent restoring method) in the event the system running normally so far suddenly breaks down. Therefore, the following cases are excluded.

① Momentary failure due to transient abnormality due to noise or other effect (irreproducible trouble).

② Trouble due to effect of ladder program (customer's application).

[3] Prepare for causing trouble

1. Back-up for program memory and system memory

When the CPU board is abnormal, the current program memory and the like may not be saved by the support tool, or the saved data may be incorrect. Therefore, store the latest program memory and back-up of system memory always in the floppy disk (FD).

In the case of ROM operation, store the back-up in the FD with spare ROM, too.

2. Prepare for support tool

Prepare the support tool that can load/save of hand-held programmer or program.

3. Prepare for spare parts

Prepare always a spare of each module to be ready for abnormality.

4. Prepare for setting system memory table, I/O relay allocation table

For prompt trouble shooting, prepare the "system memory setting table" and "I/O relay allocation table."

[4] Check flow

Check flow 1

Monitor system memory #160 using a hand-held programmer JW-13PG

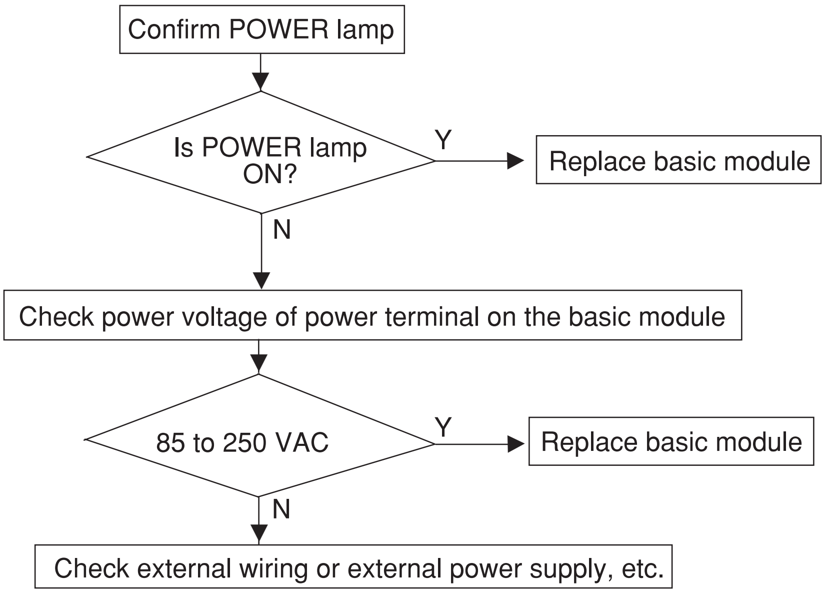
<Operation procedure of JW-13PG>

クリア CLR → システム SYS → B 1 → 6 → A 0 → モニタ MNTR

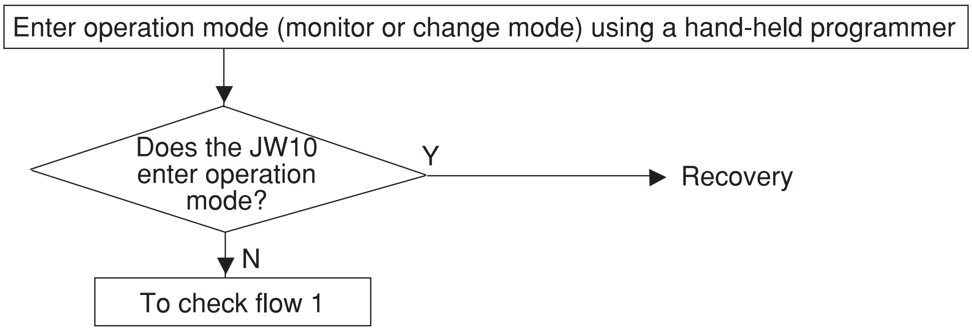
Value of #160	Contents	Treatment
20	System ROM error	Reinput the power and if this error occurs, replace the basic module.
22	Battery error	
27	RAM error	Reinput the power and if this error occurs, initialize all memory. Then load the program and system memory again. If the error is still not cleared, replace the basic module.
31	Watchdog timer error	
26	User ROM error	Reinput the power and if this error occurs, initialize all memory. Then load the program and system memory again, and rewrite onto a ROM. If the error is still not cleared, replace the basic module.
21	User program error 1 (instruction code check)	Reinput the power and if this error occurs, check user program address having the error with #052 and #053. Then, rewrite the program. If the error is not still cleared, initialize all memory and load the program and system memory again. If the situation is still not changed, replace the basic module. Note 1: If JW10 has grammatical error on the program (doubled use of OUT instruction or the like), be careful as "user program error 1" occurs and it cannot operate. After inputting the program, check the program using a hand-held programmer, and check grammatical error of the program. 《Operation of JW-13PG》 (Program mode) クリア CLR → ∗ → ∗ → 検索 SRCH If 55(H) is stored in register #055 in the system memory, the JW10 will continue operation without halting in a "user program error 1" status when it detects grammatical error.(The option is available in version 1.4 or later ROM code.) Note 2: Some support tools can be used with the JW10 by partly limiting functions of the JW10 (see page 14・2 for details). If any of these support tools is used with the JW10, instructions and data memory which are not available with the JW10 may be input to the JW10. Be careful that if these instructions or data memory are input, "user program error 1" occurs at start operation and the JW10 cannot start operation.
24	User program error 2 (parity check)	
25	User program error 3 (endless program check)	Reinput the power and if this error occurs, check that there is no endless program or a program having long operation time (more than 200 ms). If the error cannot be cleared, initialize all memory and load the program and system memory again. If the problem is not solved, replace the basic module.
23	System memory error	Reinput the power and if this error occurs, check system memory address, having the error with #054. Then, revise the value of the system memory. If the error cannot be cleared, initialize all memory and load the program and system memory again. If the problem is still not solved, replace the basic module. Note 1: Be careful that if value exceeding the specified range is set on a system memory, "system memory error" occurs at startup of the JW10 and the JW10 cannot be operated.
44	I/O bus error	Reinput the power and if this error occurs, check installation condition of termination connector and expansion cable. If the error cannot be cleared, replace the termination connector or expansion cable. If the problem still exists, replace the basic module, and then if not solved, replace the expansion module, too.

Value of #160(H)	Contents	Treatment
53	Communication error	• In case of #234 = 02 (remote I/O) Reinput the power and if this error occurs, check wiring condition of the communication cable which is lead to a slave station having turned OFF its communication flag. If the error cannot be cleared, replace the slave station with the communication flag OFF. If the problem still exists, replace the basic module. • In case of #234 ≠ 02 Reinput the power and if this error occurs, initialize all memory, and load the program and system memory, again. If the error cannot be cleared, replace the basic module.
Other code		Reinput the power and if this error occurs, initialize all memory. Then load the program and system memory again. If the error is still not cleared, replace the basic module. Note 1: Power error (error code 13) is always written at inputting the power. This is not abnormal.
If the JW10 cannot communicate with a hand-held programmer ("PC KIND?" is displayed)		Reinput the power and if this error occurs, replace the basic module.

Check flow 2


Confirm POWER lamp
Is POWER lamp ON?
Y
Replace basic module
N
Check power voltage of power terminal on the basic module
85 to 250 VAC
Y
Replace basic module
N
Check external wiring or external power supply, etc.

Check flow 3


Enter operation mode (monitor or change mode) using a hand-held programmer
Does the JW10 enter operation mode?
Y
Recovery
N
To check flow 1

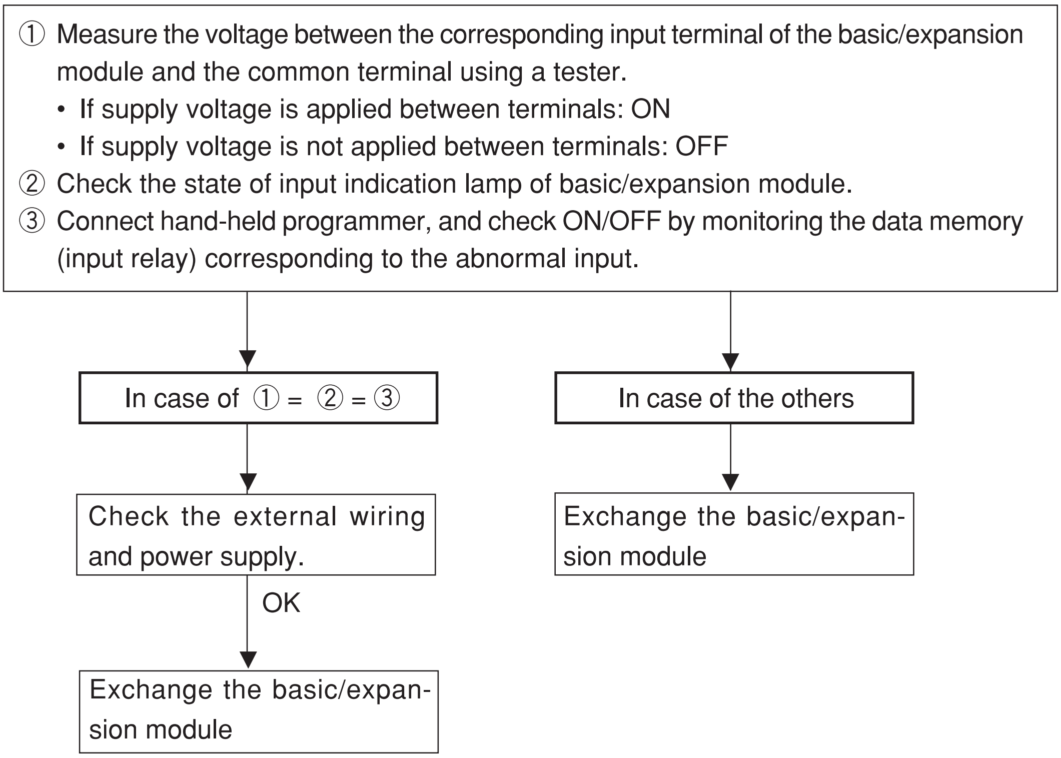
Check flow 4

This flow shows the checking procedure in the event of abnormality of input signal not detected by the self-diagnosis of the CPU.

Example of the error

- All inputs of specific basic/expansion module fail to be turned ON.
- Specific input fails to be turned ON (OFF).
- Among input signals of a same basic/expansion module, operation of a certain input signal affects other input signal.

Countermeasure　　Error input signal



① Measure the voltage between the corresponding input terminal of the basic/expansion module and the common terminal using a tester.
- If supply voltage is applied between terminals: ON
- If supply voltage is not applied between terminals: OFF

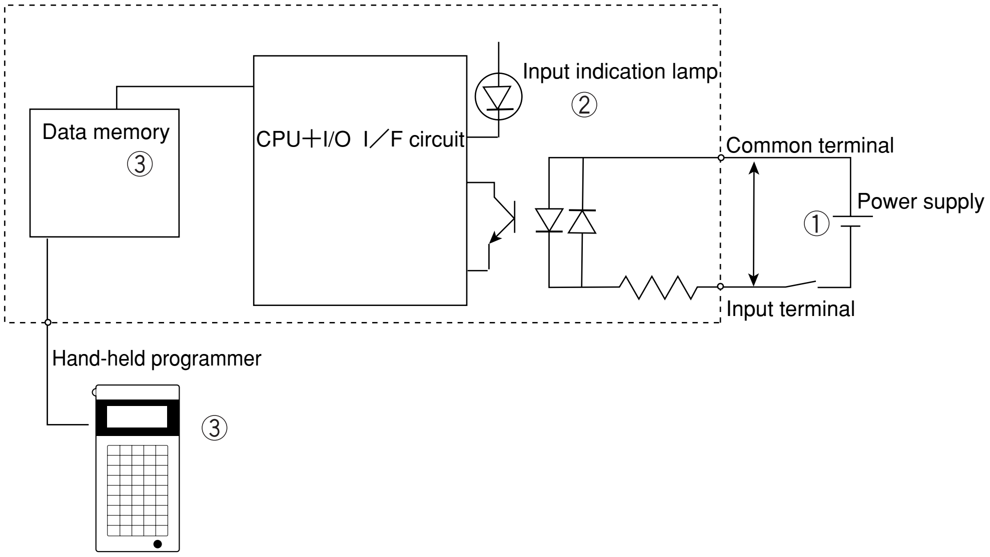
② Check the state of input indication lamp of basic/expansion module.

③ Connect hand-held programmer, and check ON/OFF by monitoring the data memory (input relay) corresponding to the abnormal input.

In case of ① = ② = ③ → Check the external wiring and power supply. → OK → Exchange the basic/expansion module

In case of the others → Exchange the basic/expansion module

[The flow of input signal]



Check flow 5

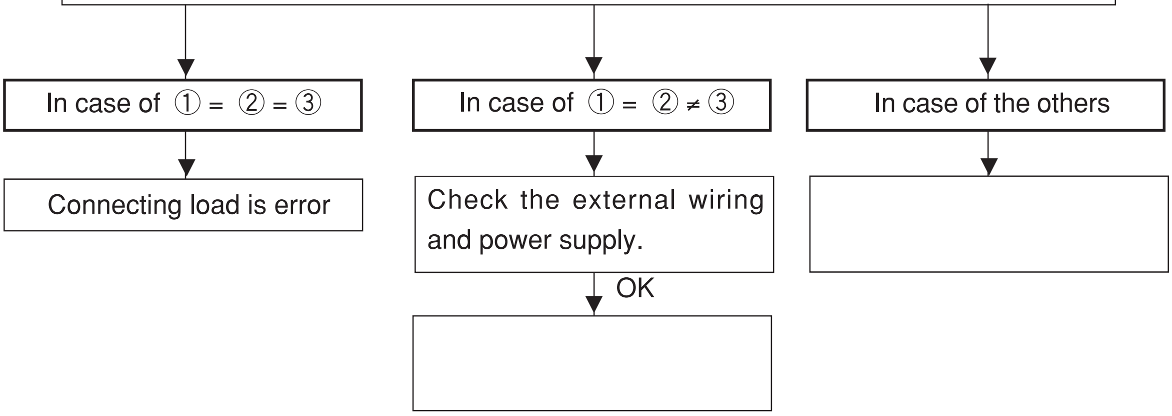
This flow shows the checking procedure in the event of abnormality of output signal not detected by the self-diagnosis of the CPU.

Example of the error

- All outputs of specific basic/expansion module fail to be turned ON.
  (In this case, it is highly possible that the fuse of load power output is melted down.)
- Specific output fails to be turned ON (OFF).
- Among output signals of a same basic/expansion module, operation of a certain output signal affects other output signal.

Countermeasure　　Error output signal

① Connect hand-held programmer, and check ON/OFF by monitoring the data memory (output relay) corresponding to the abnormal output.
② Check the state of output indication lamp of basic/expansion module.
③ Measure the voltage between the corresponding output terminal of the basic/expansion module and the common terminal using a tester.
When the inter-terminal voltage is output ON voltage (about 1 V or less): ON
When the inter-terminal voltage is load supply voltage: OFF
Note 1: When the load power source is OFF and wiring to the load is disconnected, it is abnormal if the output is normal.



In case of ① = ② = ③

Connecting load is error

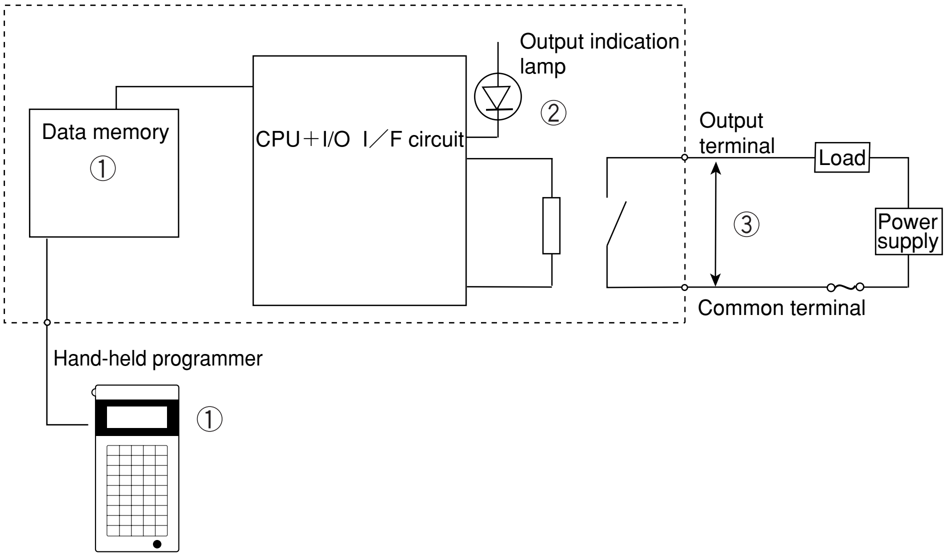
In case of ① = ② ≠ ③

Check the external wiring and power supply.

OK

In case of the others

[The flow of output signal]



Chapter 18. Specifications

18-1 General specifications

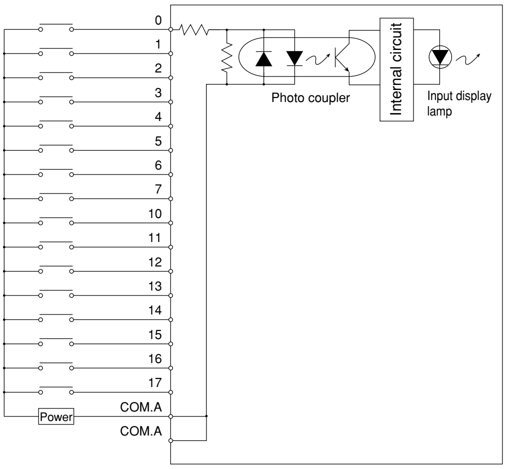
Items	Specifications					
	JW-1324K	JW-1342K	JW-1424K	JW-1442K	JW-1624K	JW-1642K
Power voltage	85 to 250 VAC, 47 to 63Hz					
Guaranteed voltage interruption time	Available voltage interruption time of 20 ms max. (In case a support tool is not connected)					
Insulation resistance	10 Mohm min. at 500 VDC megger (between AC input terminal and FG terminal)					
Dielectric strength	1500 VAC, 50/60Hz for 1 minute (between AC input terminal and FG terminal)					
Noise immunity	1000Vp-p 1 µs width impulse (by noise simulator between the power line and FG terminal)					
Storage temperature	−25 to 70 °C					
Ambient temperature	0 to 55 °C					
Ambient humidity	5 to 90 % RH (non-condensing)					
Atmosphere	Free from corrosive gas					
Vibration resistance	JIS C 0911 or equivalent amplitude 0.15 mm (10 to 58 Hz), 1G (58 to 150 Hz) (2 hours in each of X, Y and Z axis)					
Shock resistance	JIS C 0912 or equivalent 15G (3 times in each X, Y and Z axis)					
Power consumption	30 VA		55 VA		60 VA	
Power capacity for sensor	300 mA (24 VDC ±10%)		400 mA (24 VDC±10%)			
External wire grounding system	Connect to a terminal block (M3 × 7 mm self lockup). Applied with crimp-style terminal: JIS 1.25-3 or equivalent.					
Installation	Direct installation or using DIN rail (35 mm width)					
Weight	460g	580g	700g	860g	750g	890g
Grounding	Class-3 grounding					

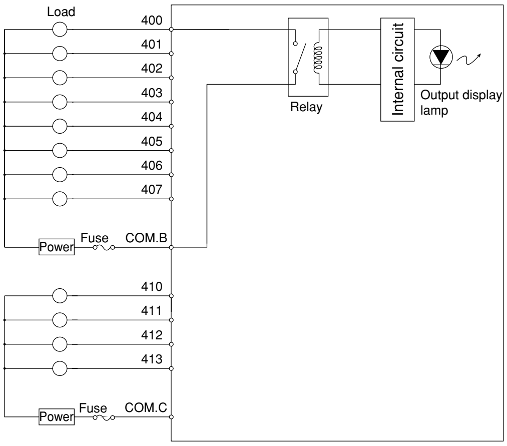
18-2 Performance specifications

Items		Specifications					
		JW-1324K	JW-1342K	JW-1424K	JW-1442K	JW-1624K	JW-1642K
Program system		Stored program system					
Control system		Compatible cyclic calculation and interrupt dealing system					
Processing speed		Basic instruction: 1.63 to 1.83µs (except OUT, TMR, CNT)		Basic instruction: 0.81 to 1.02µs (except OUT, TMR, CNT)			
Type and numbers of instruction		Basic instruction: 11 Application instruction: 69		Basic instruction: 11 Application instruction: 71			
Program size		1.5 k words (RAM)		4 k words (RAM)			
ROM operation		No available		EPROM (27C512) or EEPROM (28C256) [ROM will be supplied by user.]			
Memory back-up		By build-in lithium battery. (Battery life: 10 years)					
I/O control system		Both block refresh system and refresh system by instruction are applied.					
Basic numbers of I/O points	Input	16 points		24 points		36 points	
		DC input	DC input (high speed)	DC input	DC input (high speed)	DC input	DC input (high speed)
	Output	12 points		16 points		24 points	
		Relay output	Transistor output	Relay output	Transistor output	Relay output	Transistor output
Max. numbers of I/O points	Input	16 points		56 points		68 points	
	Output	12 points		48 points		56 points	
Data memory	Input relay	256 points (00000 to 00377) [⊐0000 to ⊐0037]					
	Output relay	256 points (00400 to 00777) [⊐0040 to ⊐0077]					
	Auxiliary relay	6656 points (01000 to 15777) [⊐0100 to ⊐1577] (Includes special relay, special register)					
	Timer / counter	Total 256 points (TMR 000 to TMR 377, CNT 000 to CNT 377) • Timer set time: 0.1 to 199.9 seconds (TMR 000 to TMR 277) 0.01 to 19.99 seconds（TMR 300 to TMR 377） • Counter setting value: 1 to 1999 Timer/counter current value storage area: 512 bytes [b0000 to b0777]					
	Register	2048 bytes [09000 to 09777, 19000 to 19777, 29000 to 29777, 39000 to 39777]					
System memory		256 bytes [#000 to #377]					
High-speed counter input		Single phase rising pulse input (2 points) or 90° differential two-phase signal (1 point). Maximum frequency: 10 kHz. Counter measuring range: 0 to 65535 (When the high-speed counter is used, input 00000 to 00003 becomes high-speed counter input.)					
Interrupt program		10 ms timer interruption (Execute subroutine having LB0177 label every 10 ms.)					
Password function		Yes					
Clock feature		None		Yes			
Halt output		None		1 point (Relay output 250 VAC/30VDC, 1A) OFF at error or stop mode, ON at normal operation			
Expansion module		Unconnectable		Max.2 expansion module and 1 analog input/output module			
Communication port		Selectable from computer link, data link, or remote I/O. 1) Computer link: SHARP's computer link protocol 38400/19200/9600/4800/2400/1200/600/300 bits/s. Maximum 63 stations. 1 km. 2) Data link: JW10 data link protocol 76800/38400 bits/s. Maximum number of slave stations: 7. 16 bytes/station. 500 m/1 km. 3) Remove I/O: JW10 remote I/O protocol 76800/38400 bits/s. Maximum number of slave stations: 4. 60 points/station. 500 m/1 km.					
MMI port		Selectable from PG mode and computer link mode 1) PG mode: Connection of support tool • Hand-held programmer: JW13PG: multipurpose programmer: JW-50PG • Ladder software: JW-92SP, JW-50SP, JW-52SP. 2) Computer link: SHARP's computer link protocol 38400/19200/9600/4800/2400/1200/600/300 bits/s. One by one connection. 1 km.					

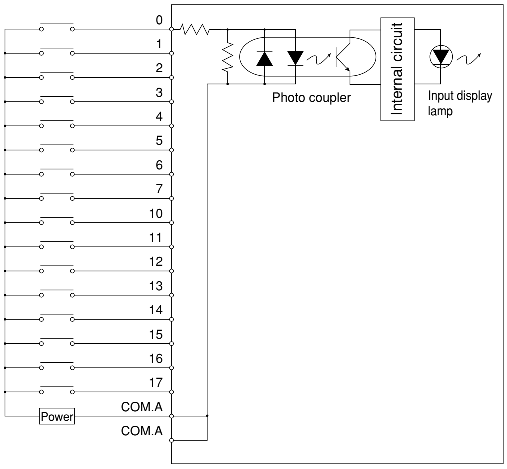
18-3 Specifications of I/O port

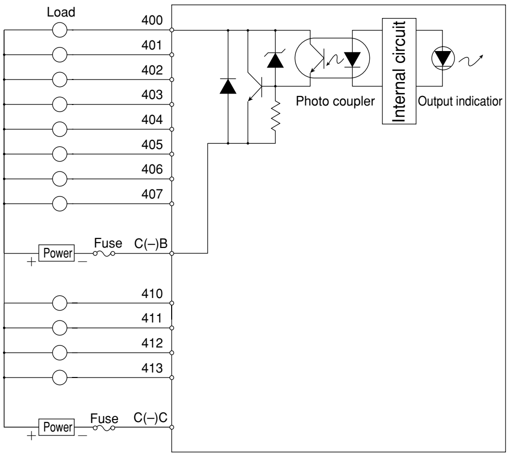
[1] JW-1324K (DC input : 16 points, relay output : 12 points)

			Terminal 0 to 3	Terminal 4 to 17	Circuit diagram
Input port	No. of input		4 points	12 points	 0 1 2 3 4 5 6 7 10 11 12 13 14 15 16 17 COM.A COM.A Power Photo coupler Internal circuit Input display lamp
	Rated input voltage		12/24 VDC	24 VDC	
	Input voltage range		10 to 26.4VDC	20 to 26.4VDC	
	Rated input current		3.6 mA TYP.(12V) 7.6 mA TYP.(24V)	4.8 mA TYP.(24 V)	
	Input impedance		3.2 kohm TYP.	5 kohm TYP.	
	Input ON level		10 V (3mA) max.	20 V (3.5mA) max.	
	Input OFF level		5 V (1.5mA) min.	8 V (1.5mA) min.	
	Response time	OFF→ON	1 ms max.	10 ms max.	
		ON→OFF	1 ms max.	10 ms max.	
	Operation indication		LED lights at ON condition		
	Insulation system		By photo coupler		
	Insulation resistance		500 VDC, 10 Mohm min. (between input terminal and secondary circuit)		
	Dielectrical strength		500 VAC for 1 minute (between input terminal and secondary circuit)		
	Common system		1 common line for 16 points (no polarity)		

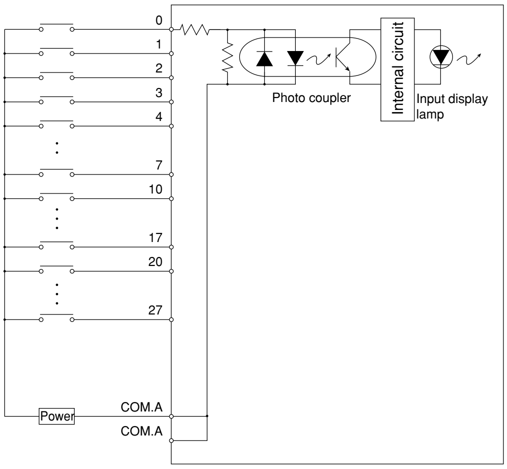
			Terminal 400 to 413	Circuit diagram
Output port	No. of output		12 points	 Load 400 401 402 403 404 405 406 407 Power Fuse COM.B 410 411 412 413 Power Fuse COM.C Relay Internal circuit Output display lamp
	Output system		Relay	
	Max. open-close voltage and current		250 VAC/30 VDC 2 A/point 2 A/common	
	Min. load		5 VDC, 10 mA	
	Operation	Mechanical	20,000,000 times min.	
		Electrical	1. Max. open-close voltage/current load: 100,000 times min. 2. Inductive load (250 VAC, 0.5 A (COS ø = 0.4)): 200,000 times min. 3. Inductive load (30 VDC, 0.5 A (T = 7 ms): 200,000 times min.	
	Response time	OFF→ON	10ms max.	
		ON→OFF	10ms max.	
	Operation indication		LED lights at ON condition	
	Insulation system		By relay	
	Insulation resistance		500 VDC, 10 Mohm min. (between output terminal and secondary circuit)	
	Dielectrical strength		1500 VAC for 1 minute (between output terminal and secondary circuit)	
	Common system		1 common line for 8 points (400 to 407) 1 common line for 4 points (410 to 413)	

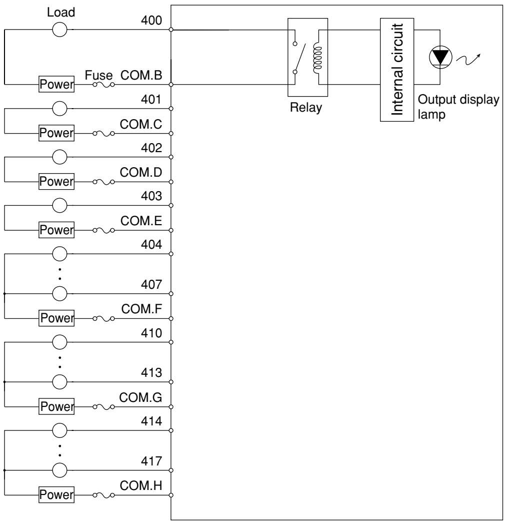
[2] JW-1342K (DC input : 16 points, transistor output : 12 points)

			Terminal 0 to 3	Terminal 4 to 17	Circuit diagram
Input port	No. of input		4 points	12 points	 0 1 2 3 4 5 6 7 10 11 12 13 14 15 16 17 COM.A COM.A Power Photo coupler Internal circuit Input display lamp
	Rated input voltage		12/24 VDC	24 VDC	
	Input voltage range		10 to 26.4VDC	20 to 26.4VDC	
	Rated input current		3.6 mA TYP.(12V) 7.6 mA TYP.(24V)	4.8 mA TYP.(24 V)	
	Input impedance		3.2 kohm TYP.	5 kohm TYP.	
	Input ON level		10 V (3mA) max.	20 V (3.5mA) max.	
	Input OFF level		5 V (1.5mA) min.	8 V (1.5mA) min.	
	Response time	OFF→ON	1 ms max.	10 ms max.	
		ON→OFF	1 ms max.	10 ms max.	
	Operation indication		LED lights at ON condition		
	Insulation system		By photo coupler		
	Insulation resistance		500 VDC, 10 Mohm min. (between input terminal and secondary circuit)		
	Dielectrical strength		500 VAC for 1 minute (between input terminal and secondary circuit)		
	Common system		1 common line for 16 points (no polarity)		

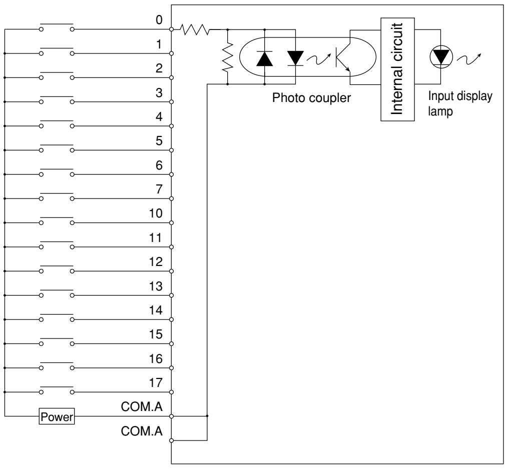
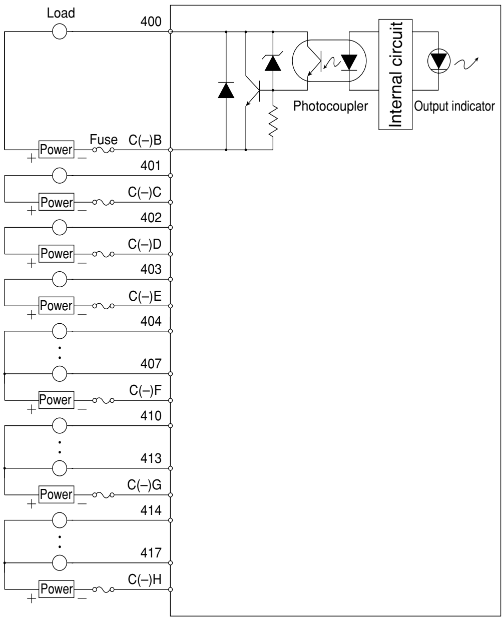
			Terminal 400 to 413	Circuit diagram
Output port	No. of output		12 points	 Load 400 401 402 403 404 405 406 407 Power Fuse C(−)B 410 411 412 413 Power Fuse C(−)C Photo coupler Internal circuit Output indicatior
	Output system		NPN transistor output (sink output)	
	Rated load voltage		5/12/24 VDC	
	Load voltage range		4.5 to 27VDC	
	Rated max. load current		0.3A/1 point 1.6A/8 points common (400 to 407) 0.8A/4 points common (410 to 413)	
	Leak current at OFF		0.2mA max.	
	Voltage breakdown at ON		1.2V max.	
	Surge killer		Zener diode	
	Response time	OFF→ON	1ms max. (resistance load)	
		ON→OFF	1ms max. (resistance load)	
	Operation indication		LED lights at ON condition	
	Insulation system		By photo coupler	
	Insulation resistance		500 VDC, 10 Mohm min. (between output terminal and secondary circuit)	
	Dielectrical strength		500 VAC for 1 minute (between output terminal and secondary circuit)	
	Common system		1 common line(-) for 8 points (400 to 407) 1 common line(-) for 4 points (410 to 413)	

[3] JW-1424K (DC input : 24 points, relay output : 16 points)

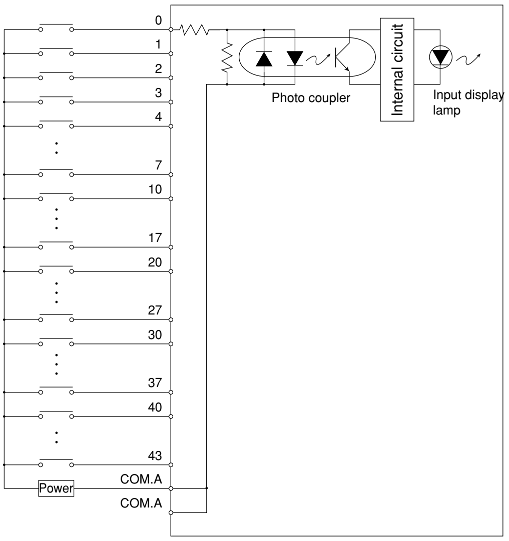
			Terminal 0 to 3	Terminal 4 to 27	Circuit diagram
Input port	No. of input		4 points	20 points	 0 1 2 3 4 7 10 17 20 27 Photo coupler Internal circuit Input display lamp Power COM.A COM.A
	Rated input voltage		12/24 VDC	24 VDC	
	Input voltage range		10 to 26.4VDC	20 to 26.4 VDC	
	Rated input current		3.6 mA TYP.(12V) 7.6 mA TYP.(24V)	4.8 mA TYP.(24 V)	
	Input impedance		3.2 kohm TYP.	5 kohm TYP.	
	Input ON level		10 V (3mA) max.	20 V (3.5mA) max.	
	Input OFF level		5 V (1.5mA) min.	8 V (1.5mA) min.	
	Response time	OFF→ON	1 ms max.	10 ms max.	
		ON→OFF	1 ms max.	10 ms max.	
	Operation indication		LED lights at ON condition		
	Insulation system		By photo coupler		
	Insulation resistance		500 VDC, 10 Mohm min. (between input terminal and secondary circuit)		
	Dielectrical strength		500 VAC for 1 minute (between input terminal and secondary circuit)		
	Common system		1 common line for 24 points (no polarity)		

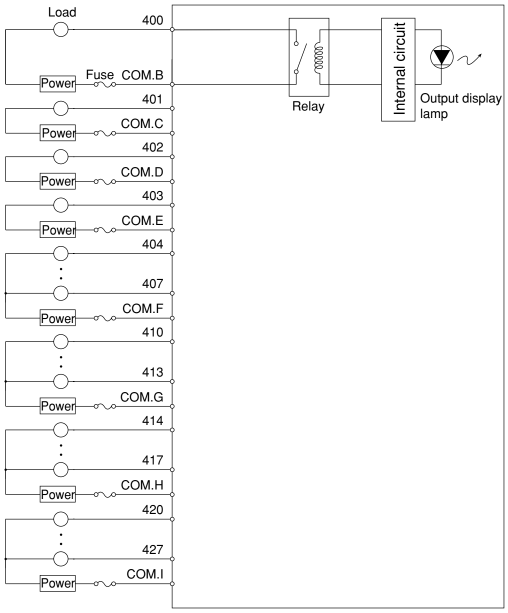
			Terminal 400 to 417	Circuit diagram
Output port	No. of output		16 points	 Load 400 Power Fuse COM.B 401 Power COM.C 402 Power COM.D 403 Power COM.E 404 407 Power COM.F 410 413 Power COM.G 414 417 Power COM.H Relay Internal circuit Output display lamp
	Output system		Relay	
	Max. open-close voltage and current		250 VAC/30 VDC 2 A/point 2 A/common	
	Min. load		5 VDC, 10 mA	
	Operation	Mechanical	20,000,000 times min.	
		Electrical	1. Max. open-close voltage/current load: 100,000 times min. 2. Inductive load (250 VAC, 0.5 A (COS ø = 0.4)):200,000 times min. 3. Inductive load (30 VDC, 0.5 A (T = 7 ms): 200,000 times min.	
	Response time	OFF→ON	10ms max.	
		ON→OFF	10ms max.	
	Operation indication		LED lights at ON condition	
	Insulation system		By relay	
	Insulation resistance		500 VDC, 10 Mohm min. (between output terminal and secondary circuit)	
	Dielectrical strength		1500 VAC for 1 minute (between output terminal and secondary circuit)	
	Common system		1 common line for 1 point × 4 (400 to 403) 1 common line for 4 points × 3 (404 to 417)	

[4] JW-1442K (DC input : 24 points, transistor output : 16 points)

Input port			Terminal 0 to 3	Terminal 4 to 17	Circuit diagram
	No. of input		4 points	12 points	 0 1 2 3 4 5 6 7 10 11 12 13 14 15 16 17 COM.A COM.A Power Photo coupler Internal circuit Input display lamp
	Rated input voltage		12/24 VDC	24 VDC	
	Input voltage range		10 to 26.4VDC	20 to 26.4VDC	
	Rated input current		3.6 mA TYP.(12V) 7.6 mA TYP.(24V)	4.8 mA TYP.(24 V)	
	Input impedance		3.2 kohm TYP.	5 kohm TYP.	
	Input ON level		10 V (3mA) max.	20 V (3.5mA) max.	
	Input OFF level		5 V (1.5mA) min.	8 V (1.5mA) min.	
	Response time	OFF→ON	1 ms max.	10 ms max.	
		ON→OFF	1 ms max.	10 ms max.	
	Operation indication		LED lights at ON condition		
	Insulation system		By photo coupler		
	Insulation resistance		500 VDC, 10 Mohm min. (between input terminal and secondary circuit)		
	Dielectrical strength		500 VAC for 1 minute (between input terminal and secondary circuit)		
	Common system		1 common line for 24 points (no polarity)		
Output port			Terminal 400 to 417		Circuit diagram
	No. of output		16 points		 Load 400 Photocoupler Internal circuit Output indicator Power Fuse C(–)B 401 C(–)C 402 C(–)D 403 C(–)E 404 407 C(–)F 410 413 C(–)G 414 417 C(–)H
	Output system		NPN transistor output (sink output)		
	Rated load voltage		5/12/24 VDC		
	Load voltage range		4.5 to 27VDC		
	Rated max. load current		0.3A/1 point 0.8A/4 points common (404 to 417)		
	Leak current at OFF		0.2mA max.		
	Voltage breakdown at ON		1.2V max.		
	Surge killer		Zener diode		
	Response time	OFF→ON	1ms max. (resistance load)		
		ON→OFF	1ms max. (resistance load)		
	Operation indication		LED lights at ON condition		
	Insulation system		By photo coupler		
	Insulation resistance		500 VDC, 10 Mohm min. (between output terminal and secondary circuit)		
	Dielectrical strength		500 VAC for 1 minute (between output terminal and secondary circuit)		
	Common system		1 common line(-) for 1 point × 4(400 to 403) 1 common line(-) for 4 points × 4(404 to 417)		

[5] JW-1624K (DC input : 36 points, relay output : 24 points)

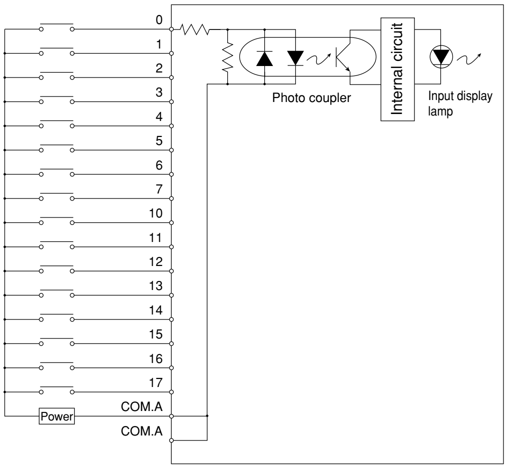
			Terminal 0 to 3	Terminal 4 to 43	Circuit diagram
Input port	No. of input		4 points	32 points	 0 1 2 3 4 7 10 17 20 27 30 37 40 43 Power COM.A COM.A Photo coupler Internal circuit Input display lamp
	Rated input voltage		12/24 VDC	24 VDC	
	Input voltage range		10 to 26.4VDC	20 to 26.4 VDC	
	Rated input current		3.6 mA TYP.(12V) 7.6 mA TYP.(24V)	4.8 mA TYP.(24 V)	
	Input impedance		3.2 kohm TYP.	5 kohm TYP.	
	Input ON level		10 V (3mA) max.	20 V (3.5mA) max.	
	Input OFF level		5 V (1.5mA) min.	8 V (1.5mA) min.	
	Response time	OFF→ON	1 ms max.	10 ms max.	
		ON→OFF	1 ms max.	10 ms max.	
	Operation indication		LED lights at ON condition		
	Insulation system		By photo coupler		
	Insulation resistance		500 VDC, 10 Mohm min. (between input terminal and secondary circuit)		
	Dielectrical strength		500 VAC for 1 minute (between input terminal and secondary circuit)		
	Common system		1 common line for 36 points (no polarity)		

			Terminal 400 to 427	Circuit diagram
Output port	No. of output		24 points	 Load 400 Power Fuse COM.B 401 COM.C 402 COM.D 403 COM.E 404 407 COM.F 410 413 COM.G 414 417 COM.H 420 427 COM.I Relay Internal circuit Output display lamp
	Output system		Relay	
	Max. open-close voltage and current		250 VAC/30 VDC 2 A/point 2 A/common	
	Min. load		5 VDC, 10 mA	
	Operation	Mechanical	20,000,000 times min.	
		Electrical	1. Max. open-close voltage/current load: 100,000 times min. 2. Inductive load (250 VAC, 0.5 A (COS ø = 0.4)):200,000 times min. 3. Inductive load (30 VDC, 0.5 A (T = 7 ms): 200,000 times min.	
	Response time	OFF→ON	10ms max.	
		ON→OFF	10ms max.	
	Operation indication		LED lights at ON condition	
	Insulation system		By relay	
	Insulation resistance		500 VDC, 10 Mohm min. (between output terminal and secondary circuit)	
	Dielectrical strength		1500 VAC for 1 minute (between output terminal and secondary circuit)	
	Common system		1 common line for 1 point × 4(400 to 403) 1 common line for 4 points × 3 (404 to 417) 1 common line for 8 points × 1 (420 to 427)	

[6] JW-1642K (DC input : 36 points, transistor output : 24 points)

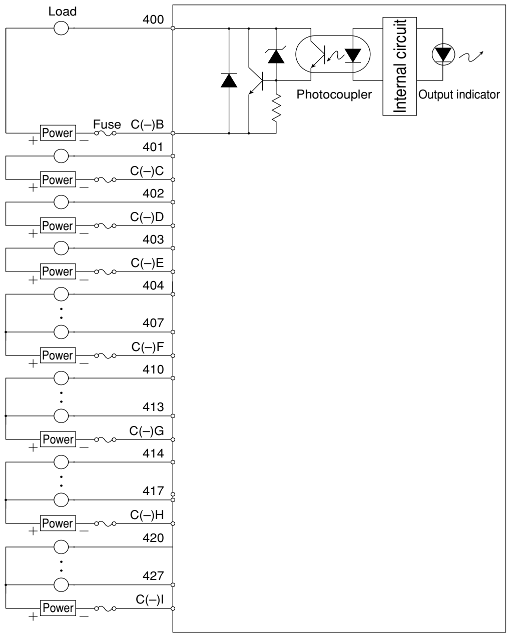
			Terminal 0 to 3	Terminal 4 to 17
Input port	No. of input		4 points	12 points
	Rated input voltage		12/24 VDC	24 VDC
	Input voltage range		10 to 26.4VDC	20 to 26.4VDC
	Rated input current		3.6 mA TYP.(12V) 7.6 mA TYP.(24V)	4.8 mA TYP.(24 V)
	Input impedance		3.2 kohm TYP.	5 kohm TYP.
	Input ON level		10 V (3mA) max.	20 V (3.5mA) max.
	Input OFF level		5 V (1.5mA) min.	8 V (1.5mA) min.
	Response time	OFF→ON	1 ms max.	10 ms max.
		ON→OFF	1 ms max.	10 ms max.
	Operation indication		LED lights at ON condition	
	Insulation system		By photo coupler	
	Insulation resistance		500 VDC, 10 Mohm min. (between input terminal and secondary circuit)	
	Dielectrical strength		500 VAC for 1 minute (between input terminal and secondary circuit)	
	Common system		1 common line for 36 points (no polarity)	

Circuit diagram

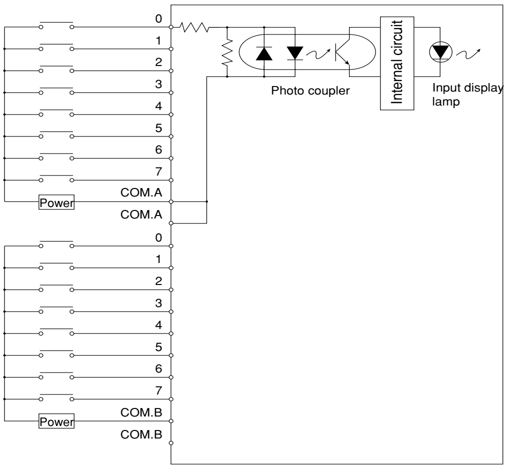


			Terminal 400 to 427
Output port	No. of output		24 points
	Output system		NPN transistor output (sink output)
	Rated load voltage		5/12/24 VDC
	Load voltage range		4.5 to 27VDC
	Rated max. load current		0.3A/1 point 0.8A/4 points common (404 to 417) 1.6A/8 points common (420 to 427)
	Leak current at OFF		0.2mA max.
	Voltage breakdown at ON		1.2V max.
	Surge killer		Zener diode
	Response time	OFF→ON	1ms max. (resistance load)
		ON→OFF	1ms max. (resistance load)
	Operation indication		LED lights at ON condition
	Insulation system		By photo coupler
	Insulation resistance		500 VDC, 10 Mohm min. (between output terminal and secondary circuit)
	Dielectrical strength		500 VAC for 1 minute (between output terminal and secondary circuit)
	Common system		1 common line(-) for 1 point × 4(400 to 403) 1 common line(-) for 4 points × 3(404 to 417) 1 common line(-) for 8 points × 1(420 to 427)

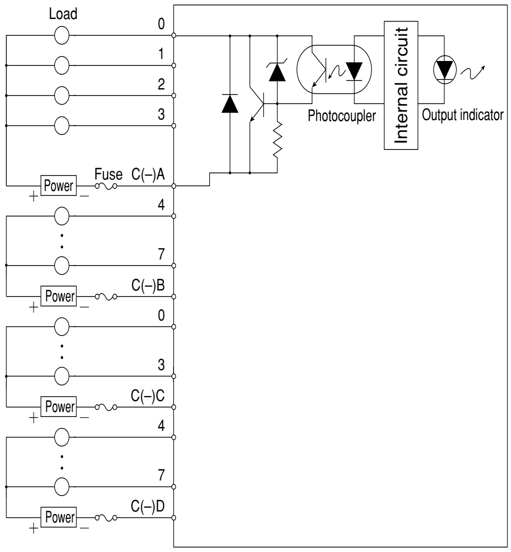
Circuit diagram



[7] JW-112N (DC input : 16 points)

			Terminal 0 to 7 × 2	Circuit diagram
Input port	No. of input		16 points	 0 1 2 3 4 5 6 7 Power COM.A COM.A 0 1 2 3 4 5 6 7 Power COM.B COM.B Photo coupler Internal circuit Input display lamp
	Rated input voltage		24 VDC	
	Input voltage range		20 to 26.4 VDC	
	Rated input current		4.8 mA TYP. (24 V)	
	Input impedance		5 Kohm TYP.	
	Input ON level		20 V (3.5 mA) max.	
	Input OFF level		8 V (1.5 mA) min.	
	Response time	OFF→ON	10 ms max.	
		ON→OFF	10 ms max.	
	Operation indication		LED lights at ON condition	
	Insulation system		By photo coupler	
	Insulation resistance		500 VDC, 10 Mohm min. (between input terminal and secondary circuit)	
	Dielectrical strength		500 VAC for 1 minute (between input terminal and secondary circuit)	
	Common system		1 common line for 8 points × 2 (no polarity)	
Weight			160 g	
Accessories			Expansion cable (4 cm) × 1	

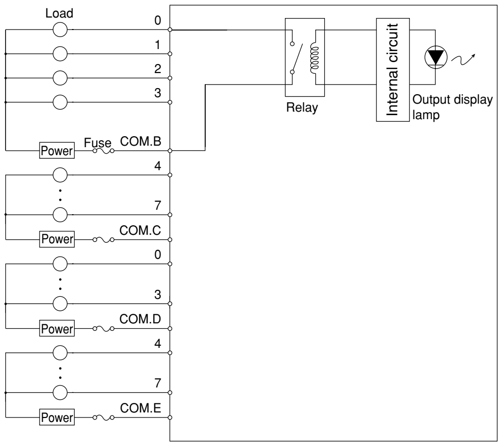
[8] JW-112S (Transistor output : 16 points)

			Terminal 0 to 7 × 2	Circuit diagram
Output port	No. of output		16 points	 Load 0 1 2 3 Power Fuse C(–)A 4 7 Power C(–)B 0 3 Power C(–)C 4 7 Power C(–)D Photocoupler Internal circuit Output indicator
	Output system		NPN transistor output (sink output)	
	Rated load voltage		5/12/24 VDC	
	Load voltage range		4.5 to 27VDC	
	Rated max. load current		0.3A/1 point 0.8A/4 points common	
	Leak current at OFF		0.2mA max.	
	Voltage breakdown at ON		1.2V max.	
	Surge killer		Zener diode	
	Response time	OFF→ON	1ms max. (resistance load)	
		ON→OFF	1ms max. (resistance load)	
	Operation indication		LED lights at ON condition	
	Insulation system		By photo coupler	
	Insulation resistance		500 VDC, 10 Mohm min. (between output terminal and secondary circuit)	
	Dielectrical strength		500 VAC for 1 minute (between output terminal and secondary circuit)	
	Common system		1 common line(-) for 4 points × 4	
Weight			230g	
Accessories			Expansion cable (4 cm) × 1	

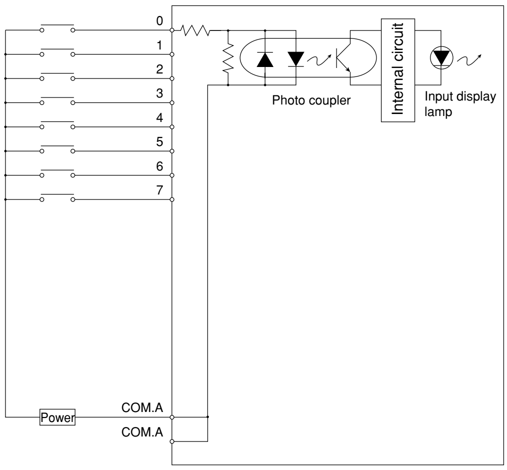
[9] JW-114S (Relay output : 16 points)

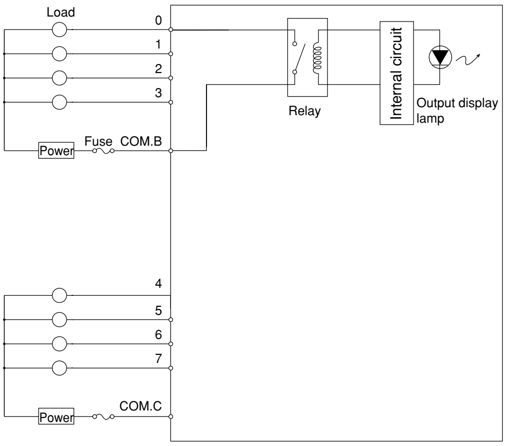
		Terminal 0 to 7 × 2
No. of output		16 points
Output system		Relay
Max. open-close voltage and current		250 VAC/30 VDC 2 A/point 2 A/common
Min. load		5 VDC, 10 mA
Operation	Mechanical	20,000,000 times min.
	Electrical	1. Max. open-close voltage/current load: 100,000 times min. 2. Inductive load (250 VAC, 0.5 A (COS ø = 0.4)):200,000 times min. 3. Inductive load (30 VDC, 0.5 A (T = 7 ms): 200,000 times min.
Response time	OFF→ON	10ms max.
	ON→OFF	10ms max.
Operation indication		LED lights at ON condition
Insulation system		By relay
Insulation resistance		500 VDC, 10 Mohm min. (between output terminal and secondary circuit)
Dielectrical strength		1500 VAC, 1 minute (between output terminal and secondary circuit)
Common system		1 common line for 4 points × 4 (no polarity)
		220 g
		Expansion cable (4 cm) × 1

Circuit diagram



[10] JW-1124NS (DC input : 8 points, relay output : 8 points)

			Terminal 0 to 7	Circuit diagram
Input port	No. of input		8 points	
	Rated input voltage		24 VDC	
	Input voltage range		20 to 26.4 VDC	
	Rated input current		4.8 mA TYP. (24 V)	
	Input impedance		5 Kohm TYP.	
	Input ON level		20 V (3.5 mA) max.	
	Input OFF level		8 V (1.5 mA) min.	
	Response time	OFF→ON	10 ms max.	
		ON→OFF	10 ms max.	
	Operation indication		LED lights at ON condition	
	Insulation system		By photo coupler	
	Insulation resistance		500 VDC, 10 Mohm min. (between input terminal and secondary circuit)	
	Dielectrical strength		500 VAC for 1 minute (between input terminal and secondary circuit)	
	Common system		1 common line for 8 points (no polarity)	

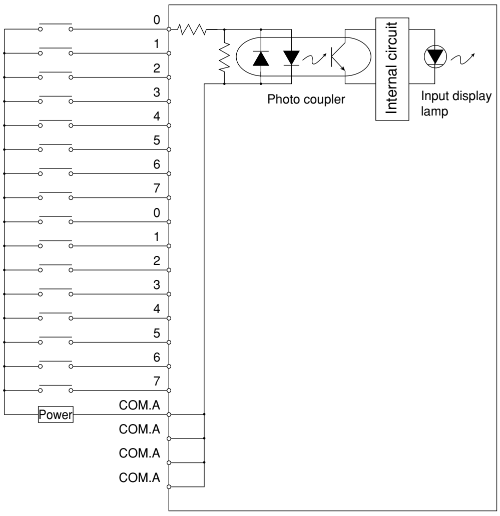
			Terminal 0 to 7	Circuit diagram
Output port	No. of output		8 points	
	Output system		Relay	
	Max. open-close voltage and current		250 VAC/30 VDC 2 A/point 2 A/common	
	Min. load		5 VDC, 10 mA	
	Operation	Mechanical	20,000,000 times min.	
		Electrical	1. Max. open-close voltage/current load: 100,000 times min. 2. Inductive load (250 VAC, 0.5 A (COS ø = 0.4)):200,000 times min. 3. Inductive load (30 VDC, 0.5 A (T = 7 ms): 200,000 times min.	
	Response time	OFF→ON	10ms max.	
		ON→OFF	10ms max.	
	Operation indication		LED lights at ON condition	
	Insulation system		By relay	
	Insulation resistance		500 VDC, 10 Mohm min. (between output terminal and secondary circuit)	
	Dielectrical strength		1500 VAC for 1 minute (between output terminal and secondary circuit)	
	Common system		1 common line for 4 points X 2 (no polarity)	
Weight			190 g	
Accessories			Expansion cable (4 cm) X 1	

18

[11] JW-1324NS (DC input : 16 points, relay output : 16 points)

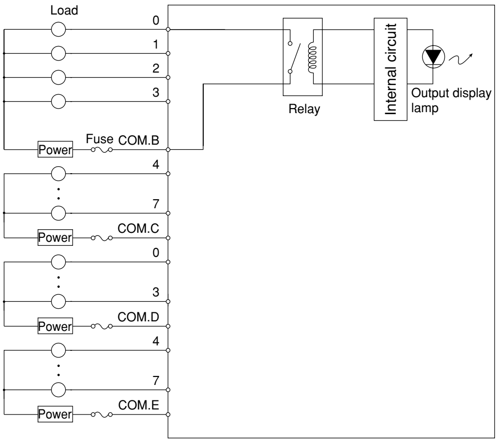
			Terminal 0 to 7 × 2
Input port	No. of input		16 points
	Rated input voltage		24 VDC
	Input voltage range		20 to 26.4 VDC
	Rated input current		4.8 mA TYP. (24 V)
	Input impedance		5 Kohm TYP.
	Input ON level		20 V (3.5 mA) max.
	Input OFF level		8 V (1.5 mA) min.
	Response time	OFF→ON	10 ms max.
		ON→OFF	10 ms max.
	Operation indication		LED lights at ON condition
	Insulation system		By photo coupler
	Insulation resistance		500 VDC, 10 Mohm min. (between input terminal and secondary circuit)
	Dielectrical strength		500 VAC for 1 minute (between input terminal and secondary circuit)
	Common system		1 common line for 16 points (no polarity)

Circuit diagram



			Terminal 0 to 7 × 2
Output port	No. of output		16 points
	Output system		Relay
	Max. open-close voltage and current		250 VAC/30 VDC 2 A/point 2 A/common
	Min. load		5 VDC, 10 mA
	Operation	Mechanical	20,000,000 times min.
		Electrical	1. Max. open-close voltage/current load: 100,000 times min. 2. Inductive load (250 VAC, 0.5 A (COS ø = 0.4)):200,000 times min. 3. Inductive load (30 VDC, 0.5 A (T = 7 ms): 200,000 times min.
	Response time	OFF→ON	10ms max.
		ON→OFF	10ms max.
	Operation indication		LED lights at ON condition
	Insulation system		By relay
	Insulation resistance		500 VDC, 10 Mohm min. (between output terminal and secondary circuit)
	Dielectrical strength		1500 VAC for 1 minute (between output terminal and secondary circuit)
	Common system		1 common line for 4 points × 4 (no polarity)
Weight			320 g
Accessories			Expansion cable (4 cm) × 1

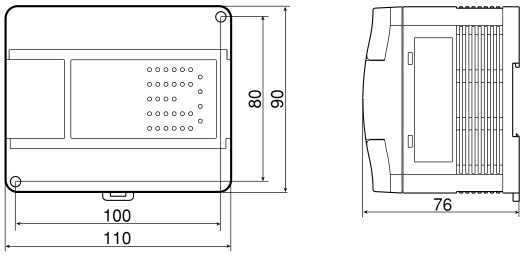
Circuit diagram



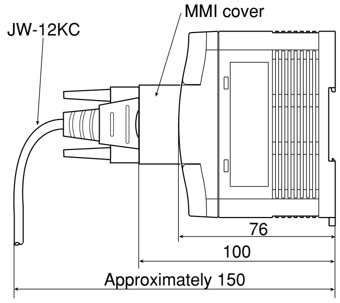
18-4 Outline dimension drawings

[1] Basic module

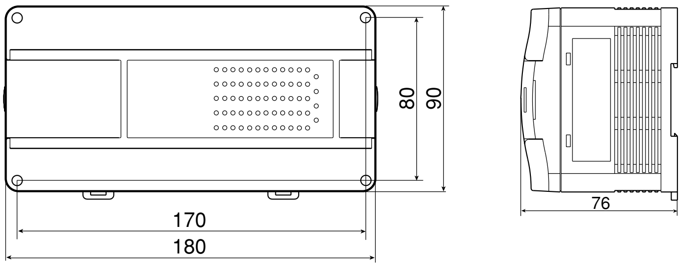
(1) JW-1324K, JW-1342K



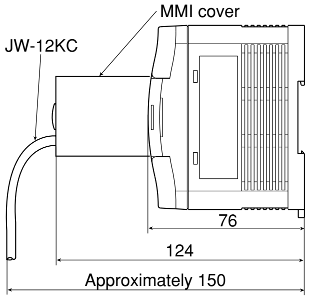
(At connecting with JW-12KC)



(2) JW-1424K, JW-1442K, JW-1624K, JW-1642K

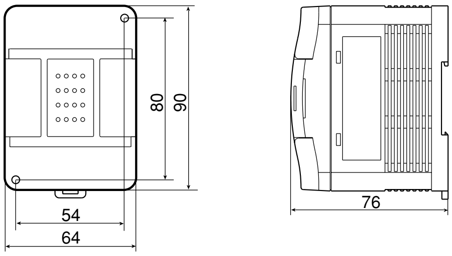


(At connecting with JW-12KC)

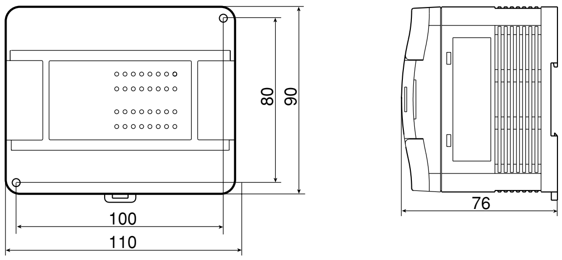


[2] Expansion module

(1) JW-112N, JW-112S, JW-114S, JW-1124NS



(2) JW-1324NS



[Unit: mm]

Appendix

Appendix-1 Address map of data memory

Relay number, timer/counter number, register number, byte address and file address are all octal notation.

[1] Input relay

Relay number								Byte address	File address
00007	00006	00005	00004	00003	00002	00001	00000	コ0000	000000
00017	00016	00015	00014	00013	00012	00011	00010	コ0001	000001
00027	00026	00025	00024	00023	00022	00021	00020	コ0002	000002
00037	00036	00035	00034	00033	00032	00031	00030	コ0003	000003
00047	00046	00045	00044	00043	00042	00041	00040	コ0004	000004
00057	00056	00055	00054	00053	00052	00051	00050	コ0005	000005
00067	00066	00065	00064	00063	00062	00061	00060	コ0006	000006
00077	00076	00075	00074	00073	00072	00071	00070	コ0007	000007
00107	00106	00105	00104	00103	00102	00101	00100	コ0010	000010
00117	00116	00115	00114	00113	00112	00111	00110	コ0011	000011
00127	00126	00125	00124	00123	00122	00121	00120	コ0012	000012
00137	00136	00135	00134	00133	00132	00131	00130	コ0013	000013
00147	00146	00145	00144	00143	00142	00141	00140	コ0014	000014
00157	00156	00155	00154	00153	00152	00151	00150	コ0015	000015
00167	00166	00165	00164	00163	00162	00161	00160	コ0016	000016
00177	00176	00175	00174	00173	00172	00171	00170	コ0017	000017
00207	00206	00205	00204	00203	00202	00201	00200	コ0020	000020
00217	00216	00215	00214	00213	00212	00211	00210	コ0021	000021
00227	00226	00225	00224	00223	00222	00221	00220	コ0022	000022
00237	00236	00235	00234	00233	00232	00231	00230	コ0023	000023
00247	00246	00245	00244	00243	00242	00241	00240	コ0024	000024
00257	00256	00255	00254	00253	00252	00251	00250	コ0025	000025
00267	00266	00265	00264	00263	00262	00261	00260	コ0026	000026
00277	00276	00275	00274	00273	00272	00271	00270	コ0027	000027
00307	00306	00305	00304	00303	00302	00301	00300	コ0030	000030
00317	00316	00315	00314	00313	00312	00311	00310	コ0031	000031
00327	00326	00325	00324	00323	00322	00321	00320	コ0032	000032
00337	00336	00335	00334	00333	00332	00331	00330	コ0033	000033
00347	00346	00345	00344	00343	00342	00341	00340	コ0034	000034
00357	00356	00355	00354	00353	00352	00351	00350	コ0035	000035
00367	00366	00365	00364	00363	00362	00361	00360	コ0036	000036
00377	00376	00375	00374	00373	00372	00371	00370	コ0037	000037

[2] Output relay

Relay number								Byte address	File address
00407	00406	00405	00404	00403	00402	00401	00400	コ0040	000040
00417	00416	00415	00414	00413	00412	00411	00410	コ0041	000041
00427	00426	00425	00424	00423	00422	00421	00420	コ0042	000042
00437	00436	00435	00434	00433	00432	00431	00430	コ0043	000043
00447	00446	00445	00444	00443	00442	00441	00440	コ0044	000044
00457	00456	00455	00454	00453	00452	00451	00450	コ0045	000045
00467	00466	00465	00464	00463	00462	00461	00460	コ0046	000046
00477	00476	00475	00474	00473	00472	00471	00470	コ0047	000047
00507	00506	00505	00504	00503	00502	00501	00500	コ0050	000050
00517	00516	00515	00514	00513	00512	00511	00510	コ0051	000051
00527	00526	00525	00524	00523	00522	00521	00520	コ0052	000052
00537	00536	00535	00534	00533	00532	00531	00530	コ0053	000053
00547	00546	00545	00544	00543	00542	00541	00540	コ0054	000054
00557	00556	00555	00554	00553	00552	00551	00550	コ0055	000055
00567	00566	00565	00564	00563	00562	00561	00560	コ0056	000056
00577	00576	00575	00574	00573	00572	00571	00570	コ0057	000057
00607	00606	00605	00604	00603	00602	00601	00600	コ0060	000060
00617	00616	00615	00614	00613	00612	00611	00610	コ0061	000061
00627	00626	00625	00624	00623	00622	00621	00620	コ0062	000062
00637	00636	00635	00634	00633	00632	00631	00630	コ0063	000063
00647	00646	00645	00644	00643	00642	00641	00640	コ0064	000064
00657	00656	00655	00654	00653	00652	00651	00650	コ0065	000065
00667	00666	00665	00664	00663	00662	00661	00660	コ0066	000066
00677	00676	00675	00674	00673	00672	00671	00670	コ0067	000067
00707	00706	00705	00704	00703	00702	00701	00700	コ0070	000070
00717	00716	00715	00714	00713	00712	00711	00710	コ0071	000071
00727	00726	00725	00724	00723	00722	00721	00720	コ0072	000072
00737	00736	00735	00734	00733	00732	00731	00730	コ0073	000073
00747	00746	00745	00744	00743	00742	00741	00740	コ0074	000074
00757	00756	00755	00754	00753	00752	00751	00750	コ0075	000075
00767	00766	00765	00764	00763	00762	00761	00760	コ0076	000076
00777	00776	00775	00774	00773	00772	00771	00770	コ0077	000077

[3] Auxiliary relay

Relay number								Byte address	File address
01007	01006	01005	01004	01003	01002	01001	01000	コ0100	000100
01017	01016	01015	01014	01013	01012	01011	01010	コ0101	000101
01027	01026	01025	01024	01023	01022	01021	01020	コ0102	000102
01037	01036	01035	01034	01033	01032	01031	01030	コ0103	000103
01047	01046	01045	01044	01043	01042	01041	01040	コ0104	000104
01057	01056	01055	01054	01053	01052	01051	01050	コ0105	000105
01067	01066	01065	01064	01063	01062	01061	01060	コ0106	000106
01077	01076	01075	01074	01073	01072	01071	01070	コ0107	000107
01107	01106	01105	01104	01103	01102	01101	01100	コ0110	000110
01117	01116	01115	01114	01113	01112	01111	01110	コ0111	000111
01127	01126	01125	01124	01123	01122	01121	01120	コ0112	000112
01137	01136	01135	01134	01133	01132	01131	01130	コ0113	000113
01147	01146	01145	01144	01143	01142	01141	01140	コ0114	000114
01157	01156	01155	01154	01153	01152	01151	01150	コ0115	000115
01167	01166	01165	01164	01163	01162	01161	01160	コ0116	000116
01177	01176	01175	01174	01173	01172	01171	01170	コ0117	000117
01207	01206	01205	01204	01203	01202	01201	01200	コ0120	000120
01217	01216	01215	01214	01213	01212	01211	01210	コ0121	000121
01227	01226	01225	01224	01223	01222	01221	01220	コ0122	000122
01237	01236	01235	01234	01233	01232	01231	01230	コ0123	000123
01247	01246	01245	01244	01243	01242	01241	01240	コ0124	000124
01257	01256	01255	01254	01253	01252	01251	01250	コ0125	000125
01267	01266	01265	01264	01263	01262	01261	01260	コ0126	000126
01277	01276	01275	01274	01273	01272	01271	01270	コ0127	000127
01307	01306	01305	01304	01303	01302	01301	01300	コ0130	000130
01317	01316	01315	01314	01313	01312	01311	01310	コ0131	000131
01327	01326	01325	01324	01323	01322	01321	01320	コ0132	000132
01337	01336	01335	01334	01333	01332	01331	01330	コ0133	000133
01347	01346	01345	01344	01343	01342	01341	01340	コ0134	000134
01357	01356	01355	01354	01353	01352	01351	01350	コ0135	000135
01367	01366	01365	01364	01363	01362	01361	01360	コ0136	000136
01377	01376	01375	01374	01373	01372	01371	01370	コ0137	000137
01407	01406	01405	01404	01403	01402	01401	01400	コ0140	000140
01417	01416	01415	01414	01413	01412	01411	01410	コ0141	000141
01427	01426	01425	01424	01423	01422	01421	01420	コ0142	000142
01437	01436	01435	01434	01433	01432	01431	01430	コ0143	000143
01447	01446	01445	01444	01443	01442	01441	01440	コ0144	000144
01457	01456	01455	01454	01453	01452	01451	01450	コ0145	000145
01467	01466	01465	01464	01463	01462	01461	01460	コ0146	000146
01477	01476	01475	01474	01473	01472	01471	01470	コ0147	000147
01507	01506	01505	01504	01503	01502	01501	01500	コ0150	000150
01517	01516	01515	01514	01513	01512	01511	01510	コ0151	000151
01527	01526	01525	01524	01523	01522	01521	01520	コ0152	000152
01537	01536	01535	01534	01533	01532	01531	01530	コ0153	000153
01547	01546	01545	01544	01543	01542	01541	01540	コ0154	000154
01557	01556	01555	01554	01553	01552	01551	01550	コ0155	000155
01567	01566	01565	01564	01563	01562	01561	01560	コ0156	000156
01577	01576	01575	01574	01573	01572	01571	01570	コ0157	000157
01607	01606	01605	01604	01603	01602	01601	01600	コ0160	000160
01617	01616	01615	01614	01613	01612	01611	01610	コ0161	000161
01627	01626	01625	01624	01623	01622	01621	01620	コ0162	000162
01637	01636	01635	01634	01633	01632	01631	01630	コ0163	000163
01647	01646	01645	01644	01643	01642	01641	01640	コ0164	000164
01657	01656	01655	01654	01653	01652	01651	01650	コ0165	000165
01667	01666	01665	01664	01663	01662	01661	01660	コ0166	000166
01677	01676	01675	01674	01673	01672	01671	01670	コ0167	000167
01707	01706	01705	01704	01703	01702	01701	01700	コ0170	000170
01717	01716	01715	01714	01713	01712	01711	01710	コ0171	000171
01727	01726	01725	01724	01723	01722	01721	01720	コ0172	000172
01737	01736	01735	01734	01733	01732	01731	01730	コ0173	000173
01747	01746	01745	01744	01743	01742	01741	01740	コ0174	000174
01757	01756	01755	01754	01753	01752	01751	01750	コ0175	000175
01767	01766	01765	01764	01763	01762	01761	01760	コ0176	000176
01777	01776	01775	01774	01773	01772	01771	01770	コ0177	000177

▨ 's address is special register.

Relay number								Byte address	File address
02007	02006	02005	02004	02003	02002	02001	02000	コ0200	000200
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
02777	02776	02775	02774	02773	02772	02771	02770	コ0277	000277
03007	03006	03005	03004	03003	03002	03001	03000	コ0300	000300
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
03777	03776	03775	03774	03773	03772	03771	03770	コ0377	000377
04007	04006	04005	04004	04003	04002	04001	04000	コ0400	000400
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
04777	04776	04775	04774	04773	04772	04771	04770	コ0477	000477
05007	05006	05005	05004	05003	05002	05001	05000	コ0500	000500
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
05777	05776	05775	05774	05773	05772	05771	05770	コ0577	000577
06007	06006	06005	06004	06003	06002	06001	06000	コ0600	000600
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
06777	06776	06775	06774	06773	06772	06771	06770	コ0677	000677
07007	07006	07005	07004	07003	07002	07001	07000	コ0700	000700
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
07317	07316	07315	07314	07313	07312	07311	07310	コ0731	000731
07327	07326	07325	07324	07323	07322	07321	07320	コ0732	000732
07337	07336	07335	07334	07333	07332	07331	07330	コ0733	000733
07347	07346	07345	07344	07343	07342	07341	07340	コ0734	000734
07357	07356	07355	07354	07353	07352	07351	07350	コ0735	000735
07367	07366	07365	07364	07363	07362	07361	07360	コ0736	000736
07377	07376	07375	07374	07373	07372	07371	07370	コ0737	000737
07407	07406	07405	07404	07403	07402	07401	07400	コ0740	000740
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
07767	07766	07765	07764	07763	07762	07761	07760	コ0776	000776
07777	07776	07775	07774	07773	07772	07771	07770	コ0777	000777
10007	10006	10005	10004	10003	10002	10001	10000	コ1000	001000
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
10777	10776	10775	10774	10773	10772	10771	10770	コ1077	001077
11007	11006	11005	11004	11003	11002	11001	11000	コ1100	001100
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
11777	11776	11775	11774	11773	11772	11771	11770	コ1177	001177
12007	12006	12005	12004	12003	12002	12001	12000	コ1200	001200
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
12777	12776	12775	12774	12773	12772	12771	12770	コ1277	001277
13007	13006	13005	13004	13003	13002	13001	13000	コ1300	001300
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
13777	13776	13775	13774	13773	13772	13771	13770	コ1377	001377
14007	14006	14005	14004	14003	14002	14001	14000	コ1400	001400
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
14777	14776	14775	14774	14773	14772	14771	14770	コ1477	001477
15007	15006	15005	15004	15003	15002	15001	15000	コ1500	001500
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
15577	15576	15575	15574	15573	15572	15571	15570	コ1557	001557
15607	15606	15605	15604	15603	15602	15601	15600	コ1560	001560
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
15677	15676	15675	15674	15673	15672	15671	15670	コ1567	001567
15707	15706	15705	15704	15703	15702	15701	15700	コ1570	001570
15717	15716	15715	15714	15713	15712	15711	15710	コ1571	001571
15727	15726	15725	15724	15723	15722	15721	15720	コ1572	001572
15737	15736	15735	15734	15733	15732	15731	15730	コ1573	001573
15747	15746	15745	15744	15743	15742	15741	15740	コ1574	001574
15757	15756	15755	15754	15753	15752	15751	15750	コ1575	001575
15767	15766	15765	15764	15763	15762	15761	15760	コ1576	001576
15777	15776	15775	15774	15773	15772	15771	15770	コ1577	001577

▨ 's address is special relay, special register.

[4] Timer/counter current value

TMR/CNT number	Byte address	File address
000	b0000	002000
	b0001	002001
001	b0002	002002
	b0003	002003
002	b0004	002004
	b0005	002005
003	b0006	002006
	b0007	002007
004	b0010	002010
	b0011	002011
005	b0012	002012
	b0013	002013
006	b0014	002014
	b0015	002015
007	b0016	002016
	b0017	002017
010	b0020	002020
	b0021	002021
011	b0022	002022
	b0023	002023
012	b0024	002024
	b0025	002025
013	b0026	002026
	b0027	002027
014	b0030	002030
	b0031	002031
015	b0032	002032
	b0033	002033
016	b0034	002034
	b0035	002035
017	b0036	002036
	b0037	002037
020	b0040	002040
	b0041	002041
021	b0042	002042
	b0043	002043
022	b0044	002044
	b0045	002045
023	b0046	002046
	b0047	002047
024	b0050	002050
	b0051	002051
025	b0052	002052
	b0053	002053
026	b0054	002054
	b0055	002055
027	b0056	002056
	b0057	002057
030	b0060	002060
	b0061	002061
031	b0062	002062
	b0063	002063
032	b0064	002064
	b0065	002065
033	b0066	002066
	b0067	002067
034	b0070	002070
	b0071	002071
035	b0072	002072
	b0073	002073
036	b0074	002074
	b0075	002075
037	b0076	002076
	b0077	002077

TMR/CNT number	Byte address	File address
040	b0100	002100
	b0101	002101
⋮	⋮	⋮
077	b0176	002176
	b0177	002177
100	b0200	002200
	b0201	002201
⋮	⋮	⋮
137	b0276	002276
	b0277	002277
140	b0300	002300
	b0301	002301
⋮	⋮	⋮
177	b0376	002376
	b0377	002377
200	b0400	002400
	b0401	002401
⋮	⋮	⋮
237	b0476	002476
	b0477	002477
240	b0500	002500
	b0501	002501
⋮	⋮	⋮
277	b0576	002576
	b0577	002577
300	b0600	002600
	b0601	002601
⋮	⋮	⋮
337	b0676	002676
	b0677	002677
340	b0700	002700
	b0701	002701
⋮	⋮	⋮
377	b0776	002776
	b0777	002777

[5] Register

Register	File address	Register	File address	Register	File address	Register	File address
09000	004000	09100	004100	09200	004200	09300	004300
09001	004001	09101	004101	09201	004201	09301	004301
09002	004002	09102	004102	09202	004202	09302	004302
09003	004003	09103	004103	09203	004203	09303	004303
09004	004004	09104	004104	09204	004204	09304	004304
09005	004005	09105	004105	09205	004205	09305	004305
09006	004006	09106	004106	09206	004206	09306	004306
09007	004007	09107	004107	09207	004207	09307	004307
09010	004010	09110	004110	09210	004210	09310	004310
09011	004011	09111	004111	09211	004211	09311	004311
09012	004012	09112	004112	09212	004212	09312	004312
09013	004013	09113	004113	09213	004213	09313	004313
09014	004014	09114	004114	09214	004214	09314	004314
09015	004015	09115	004115	09215	004215	09315	004315
09016	004016	09116	004116	09216	004216	09316	004316
09017	004017	09117	004117	09217	004217	09317	004317
09020	004020	09120	004120	09220	004220	09320	004320
09021	004021	09121	004121	09221	004221	09321	004321
09022	004022	09122	004122	09222	004222	09322	004322
09023	004023	09123	004123	09223	004223	09323	004323
09024	004024	09124	004124	09224	004224	09324	004324
09025	004025	09125	004125	09225	004225	09325	004325
09026	004026	09126	004126	09226	004226	09326	004326
09027	004027	09127	004127	09227	004227	09327	004327
09030	004030	09130	004130	09230	004230	09330	004330
09031	004031	09131	004131	09231	004231	09331	004331
09032	004032	09132	004132	09232	004232	09332	004332
09033	004033	09133	004133	09233	004233	09333	004333
09034	004034	09134	004134	09234	004234	09334	004334
09035	004035	09135	004135	09235	004235	09335	004335
09036	004036	09136	004136	09236	004236	09336	004336
09037	004037	09137	004137	09237	004237	09337	004337
09040	004040	09140	004140	09240	004240	09340	004340
09041	004041	09141	004141	09241	004241	09341	004341
09042	004042	09142	004142	09242	004242	09342	004342
09043	004043	09143	004143	09243	004243	09343	004343
09044	004044	09144	004144	09244	004244	09344	004344
09045	004045	09145	004145	09245	004245	09345	004345
09046	004046	09146	004146	09246	004246	09346	004346
09047	004047	09147	004147	09247	004247	09347	004347
09050	004050	09150	004150	09250	004250	09350	004350
09051	004051	09151	004151	09251	004251	09351	004351
09052	004052	09152	004152	09252	004252	09352	004352
09053	004053	09153	004153	09253	004253	09353	004353
09054	004054	09154	004154	09254	004254	09354	004354
09055	004055	09155	004155	09255	004255	09355	004355
09056	004056	09156	004156	09256	004256	09356	004356
09057	004057	09157	004157	09257	004257	09357	004357
09060	004060	09160	004160	09260	004260	09360	004360
09061	004061	09161	004161	09261	004261	09361	004361
09062	004062	09162	004162	09262	004262	09362	004362
09063	004063	09163	004163	09263	004263	09363	004363
09064	004064	09164	004164	09264	004264	09364	004364
09065	004065	09165	004165	09265	004265	09365	004365
09066	004066	09166	004166	09266	004266	09366	004366
09067	004067	09167	004167	09267	004267	09367	004367
09070	004070	09170	004170	09270	004270	09370	004370
09071	004071	09171	004171	09271	004271	09371	004371
09072	004072	09172	004172	09272	004272	09372	004372
09073	004073	09173	004173	09273	004273	09373	004373
09074	004074	09174	004174	09274	004274	09374	004374
09075	004075	09175	004175	09275	004275	09375	004375
09076	004076	09176	004176	09276	004276	09376	004376
09077	004077	09177	004177	09277	004277	09377	004377

Register	File address	Register	File address	Register	File address	Register	File address
09400	004400	09500	004500	09600	004600	09700	004700
09401	004401	09501	004501	09601	004601	09701	004701
09402	004402	09502	004502	09602	004602	09702	004702
09403	004403	09503	004503	09603	004603	09703	004703
09404	004404	09504	004504	09604	004604	09704	004704
09405	004405	09505	004505	09605	004605	09705	004705
09406	004406	09506	004506	09606	004606	09706	004706
09407	004407	09507	004507	09607	004607	09707	004707
09410	004410	09510	004510	09610	004610	09710	004710
09411	004411	09511	004511	09611	004611	09711	004711
09412	004412	09512	004512	09612	004612	09712	004712
09413	004413	09513	004513	09613	004613	09713	004713
09414	004414	09514	004514	09614	004614	09714	004714
09415	004415	09515	004515	09615	004615	09715	004715
09416	004416	09516	004516	09616	004616	09716	004716
09417	004417	09517	004517	09617	004617	09717	004717
09420	004420	09520	004520	09620	004620	09720	004720
09421	004421	09521	004521	09621	004621	09721	004721
09422	004422	09522	004522	09622	004622	09722	004722
09423	004423	09523	004523	09623	004623	09723	004723
09424	004424	09524	004524	09624	004624	09724	004724
09425	004425	09525	004525	09625	004625	09725	004725
09426	004426	09526	004526	09626	004626	09726	004726
09427	004427	09527	004527	09627	004627	09727	004727
09430	004430	09530	004530	09630	004630	09730	004730
09431	004431	09531	004531	09631	004631	09731	004731
09432	004432	09532	004532	09632	004632	09732	004732
09433	004433	09533	004533	09633	004633	09733	004733
09434	004434	09534	004534	09634	004634	09734	004734
09435	004435	09535	004535	09635	004635	09735	004735
09436	004436	09536	004536	09636	004636	09736	004736
09437	004437	09537	004537	09637	004637	09737	004737
09440	004440	09540	004540	09640	004640	09740	004740
09441	004441	09541	004541	09641	004641	09741	004741
09442	004442	09542	004542	09642	004642	09742	004742
09443	004443	09543	004543	09643	004643	09743	004743
09444	004444	09144	004544	09644	004644	09744	004744
09445	004445	09545	004545	09645	004645	09745	004745
09446	004446	09546	004546	09646	004646	09746	004746
09447	004447	09547	004547	09647	004647	09747	004747
09450	004450	09550	004550	09650	004650	09750	004750
09451	004451	09551	004551	09651	004651	09751	004751
09452	004452	09552	004552	09652	004652	09752	004752
09453	004453	09553	004553	09653	004653	09753	004753
09454	004454	09554	004554	09654	004654	09754	004754
09455	004455	09555	004555	09655	004655	09755	004755
09456	004456	09556	004556	09656	004656	09756	004756
09457	004457	09557	004557	09657	004657	09757	004757
09460	004460	09560	004560	09660	004660	09760	004760
09461	004461	09561	004561	09661	004661	09761	004761
09462	004462	09562	004562	09662	004662	09762	004762
09463	004463	09563	004563	09663	004663	09763	004763
09464	004464	09564	004564	09664	004664	09764	004764
09465	004465	09565	004565	09665	004665	09765	004765
09466	004466	09566	004566	09666	004666	09766	004766
09467	004467	09567	004567	09667	004667	09767	004767
09470	004470	09570	004570	09670	004670	09770	004770
09471	004471	09571	004571	09671	004671	09771	004771
09472	004472	09572	004572	09672	004672	09772	004772
09473	004473	09573	004573	09673	004673	09773	004773
09474	004474	09574	004574	09674	004674	09774	004774
09475	004475	09575	004575	09675	004675	09775	004775
09476	004476	09576	004576	09676	004676	09776	004776
09477	004477	09577	004577	09677	004677	09777	004777

Register	File address	Register	File address	Register	File address
19000	005000	29000	006000	39000	007000
⋮	⋮	⋮	⋮	⋮	⋮
19077	005077	29077	006077	39077	007077
19100	005100	29100	006100	39100	007100
⋮	⋮	⋮	⋮	⋮	⋮
19177	005177	29177	006177	39177	007177
19200	005200	29200	006200	39200	007200
⋮	⋮	⋮	⋮	⋮	⋮
19277	005277	29277	006277	39277	007277
19300	005300	29300	006300	39300	007300
⋮	⋮	⋮	⋮	⋮	⋮
19377	005377	29377	006377	39377	007377
19400	005400	29400	006400	39400	007400
⋮	⋮	⋮	⋮	⋮	⋮
19477	005477	29477	006477	39477	007477
19500	005500	29500	006500	39500	007500
⋮	⋮	⋮	⋮	⋮	⋮
19577	005577	29577	006577	39577	007577
19600	005600	29600	006600	39600	007600
⋮	⋮	⋮	⋮	⋮	⋮
19677	005677	29677	006677	39677	007677
19700	005700	29700	006700	39700	007700
⋮	⋮	⋮	⋮	⋮	⋮
19777	005777	29777	006777	39777	007777

Appendix-2 ASCII code table

(1) For binary/hexadecimal

- How to use ASCII code table
  Capital "A" is positioned in "4" of upper bit and "1" of lower bit. Therefore, ASCII code of A is "41(H)".

		Upper bit					
		0	1	2	3	4	5
Lower bit	0						
	1					A	
	2						
	3						

			Upper bit															
		Hexa-decimal	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
	Hexa-decimal	Decimal Binary	0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111
Lower bit	0	0000	NUL	DLE	SP	0	@	P	`	p			SP	ー	タ	ミ		
	1	0001	SOH	DC1	!	1	A	Q	a	q			。	ア	チ	ム		
	2	0010	STX	DC2	″	2	B	R	b	r			「	イ	ツ	メ		
	3	0011	ETX	DC3	#	3	C	S	c	s			」	ウ	テ	モ		
	4	0100	EOT	DC4	$	4	D	T	d	t			、	エ	ト	ヤ		
	5	0101	ENQ	NAK	%	5	E	U	e	u			・	オ	ナ	ユ		
	6	0110	ACK	SYN	&	6	F	V	f	v			ヲ	カ	ニ	ヨ		
	7	0111	BLE	ETB	′	7	G	W	g	w			ァ	キ	ヌ	ラ		
	8	1000	BS	CAN	(	8	H	X	h	x			ィ	ク	ネ	リ		
	9	1001	HT	EM	)	9	I	Y	i	y			ゥ	ケ	ノ	ル		
	A	1010	LF	SUB	*	:	J	Z	j	z			ェ	コ	ハ	レ		
	B	1011	VT	ESC	+	;	K	[	k	{			ォ	サ	ヒ	ロ		
	C	1100	FF	FS	,	<	L	¥	l	|			ャ	シ	フ	ワ		
	D	1101	CR	GS	−	=	M	]	m	}			ュ	ス	ヘ	ン		
	E	1110	SO	RS	.	>	N	^	n	‾			ョ	セ	ホ	゛		
	F	1111	SI	US	/	?	O	_	o	DEL			ッ	ソ	マ	゜		

- This code table is JIS standard table and undefined parts are deleted.

(2) For octal

- How to use ASCII code table
  Capital "A" is positioned in "10" of upper 2 digits and "1" of lower 1 digit. Therefore, ASCII code of "A" is "101(8)" in octal.

		Upper							
		04	05	06	07	08	10	11	12
Lower	0								
	1						A		
	2								

		Upper 2 digits															
	Octal	00	01	02	03	04	05	06	07	10	11	12	13	14	15	16	17
Lower 1 digit	0	NUL	BS	DLE	CAN	SP	(	0	8	@	H	P	X	`	h	p	x
	1	SOH	HT	DC1	EM	!	)	1	9	A	I	Q	Y	a	i	q	y
	2	STX	LF	DC2	SUB	"	*	2	:	B	J	R	Z	b	j	r	z
	3	ETX	VT	DC3	ESC	#	+	3	;	C	K	S	[	c	k	s	{
	4	EOT	FF	DC4	FS	$	,	4	<	D	L	T	¥	d	l	t	|
	5	ENQ	CR	NAK	GS	%	−	5	=	E	M	U	]	e	m	u	}
	6	ACK	SO	SYN	RS	&	.	6	>	F	N	V	^	f	n	v	‾
	7	BLE	SI	ETB	US	′	/	7	?	G	O	W	_	g	o	w	DEL

		Upper 2 digits															
	Octal	20	21	22	23	24	25	26	27	30	31	32	33	34	35	36	37
Lower 1 digit	0					SP	ィ	ー	ク	タ	ネ	ミ	リ				
	1					。	ゥ	ア	ケ	チ	ノ	ム	ル				
	2					「	ェ	イ	コ	ツ	ハ	メ	レ				
	3					」	ォ	ウ	サ	テ	ヒ	モ	ロ				
	4					、	ャ	エ	シ	ト	フ	ヤ	ワ				
	5					・	ュ	オ	ス	ナ	ヘ	ユ	ン				
	6					ヲ	ョ	カ	セ	ニ	ホ	ヨ	゛				
	7					ァ	ッ	キ	ソ	ヌ	マ	ラ	゜				

- This code table is JIS standard table and undefined parts are deleted.

Appendix-3 Binary/octal/decimal/hexadecimal/BCD code correspondence table

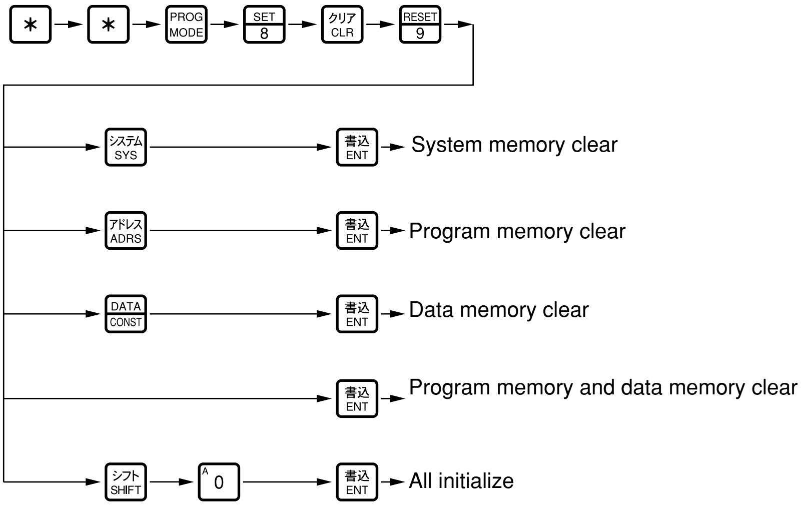
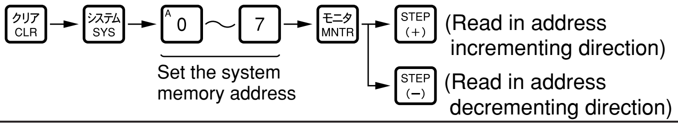
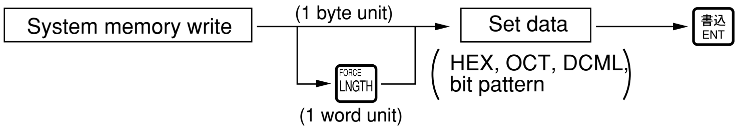
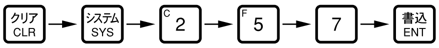
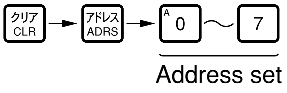
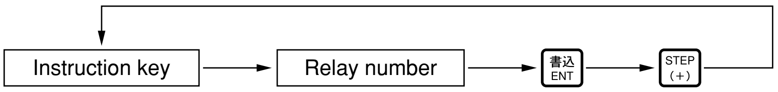
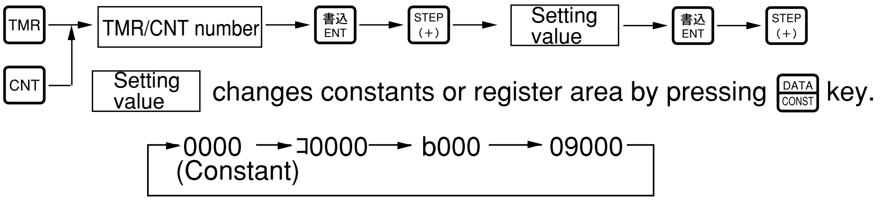
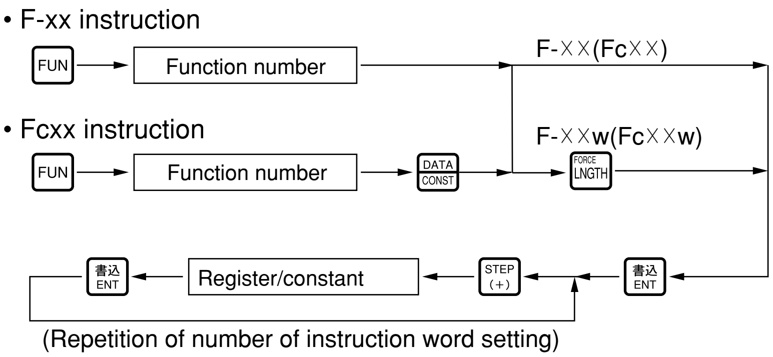
Decimal	Binary	Octal	Hexadecimal	Binary Coded Decimal (4 digits)
0	00000000 00000000	0	0000	0000 0000 0000 0000
1	00000000 00000001	1	0001	0000 0000 0000 0001
2	00000000 00000010	2	0002	0000 0000 0000 0010
3	00000000 00000011	3	0003	0000 0000 0000 0011
4	00000000 00000100	4	0004	0000 0000 0000 0100
5	00000000 00000101	5	0005	0000 0000 0000 0101
6	00000000 00000110	6	0006	0000 0000 0000 0110
7	00000000 00000111	7	0007	0000 0000 0000 0111
8	00000000 00001000	10	0008	0000 0000 0000 1000
9	00000000 00001001	11	0009	0000 0000 0000 1001
10	00000000 00001010	12	000A	0000 0000 0001 0000
11	00000000 00001011	13	000B	0000 0000 0001 0001
12	00000000 00001100	14	000C	0000 0000 0001 0010
13	00000000 00001101	15	000D	0000 0000 0001 0011
14	00000000 00001110	16	000E	0000 0000 0001 0100
15	00000000 00001111	17	000F	0000 0000 0001 0101
16	00000000 00010000	20	0010	0000 0000 0001 0110
17	00000000 00010001	21	0011	0000 0000 0001 0111
18	00000000 00010010	22	0012	0000 0000 0001 1000
19	00000000 00010011	23	0013	0000 0000 0001 1001
20	00000000 00010100	24	0014	0000 0000 0010 0000
21	00000000 00010101	25	0015	0000 0000 0010 0001
22	00000000 00010110	26	0016	0000 0000 0010 0010
23	00000000 00010111	27	0017	0000 0000 0010 0011
24	00000000 00011000	30	0018	0000 0000 0010 0100
25	00000000 00011001	31	0019	0000 0000 0010 0101
26	00000000 00011010	32	001A	0000 0000 0010 0110
27	00000000 00011011	33	001B	0000 0000 0010 0111
28	00000000 00011100	34	001C	0000 0000 0010 1000
29	00000000 00011101	35	001D	0000 0000 0010 1001
30	00000000 00011110	36	001E	0000 0000 0011 0000
31	00000000 00011111	37	001F	0000 0000 0011 0001
63	00000000 00111111	77	003F	0000 0000 0110 0011
255	00000000 11111111	377	00FF	0000 0010 0101 0101
9999	00100111 00001111	23417	270F	1001 1001 1001 1001
65535	11111111 11111111	177777	FFFF	—

Appendix-4 JW-13PG key operation

Below shows operation procedure to use JW-13PG for the JW10.

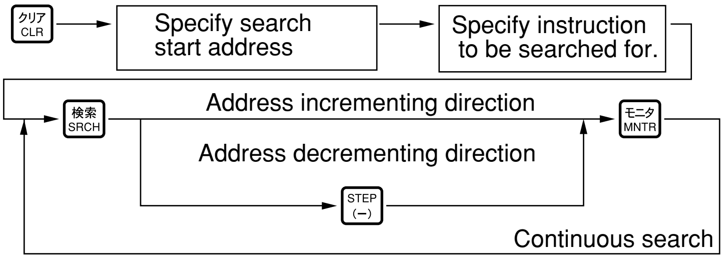
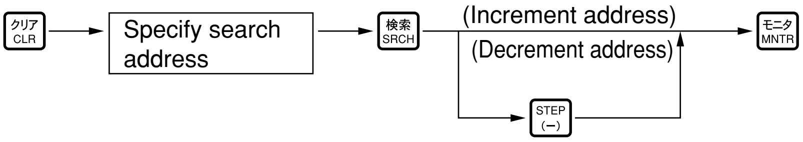
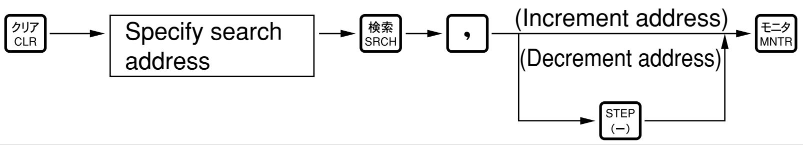
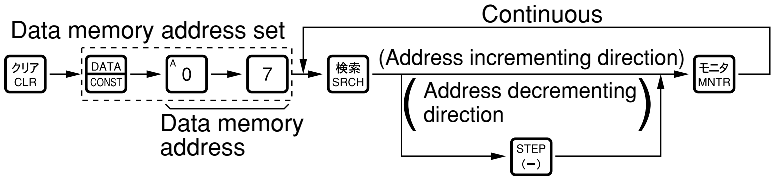
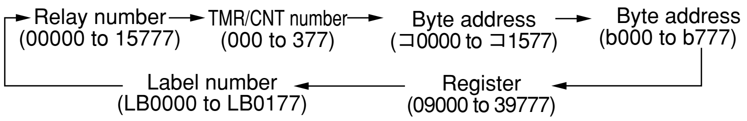
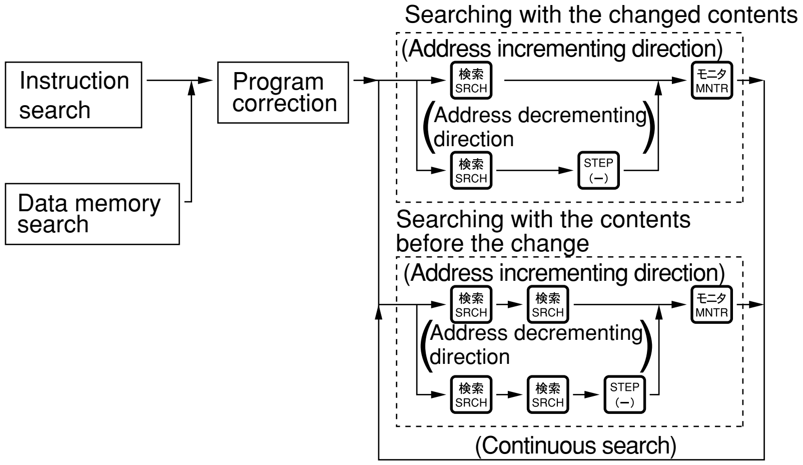
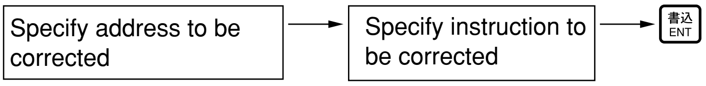
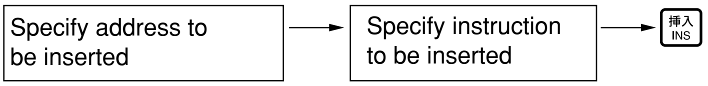
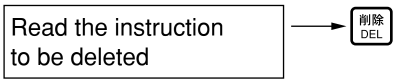
Mode [P: Program mode
M: Monitor mode
C: Change mode
T: Terminal mode
I: Initial mode]

Function	Operation procedure	Mode				
		P	M	C	T	I
Buzzer ON/ OFF select	∗ → ∗ → B 1 → SET 8 (ON) → RESET 9 (OFF)	○	○	○	×	○
EL backlight ON/OFF select	∗ → ∗ → A 0 → SET 8 (ON) → RESET 9 (OFF)	○	○	○	×	○
Contrast adjustment	∗ → ∗ → A 0 → STEP (+) (Darker) → STEP (–) (Brighter)	○	○	○	×	○
Auto repeat function	Monitoring of program address or data memory address → STEP (+) 1 second or longer ON ---- Repetition of address +1. → STEP (–) 1 second or longer ON ---- Repetition of address -1.	○	○	○	×	×
Display mode selection	1PG mode → ∗ ∗ INTL DISP → 2PG mode → ∗ ∗ INTL DISP → 1PG mode	○	○	○	×	×
Operation mode setting	∗ → ∗ → PROG MODE → SET 8 (Program mode) → MNTR MODE → SET 8 (Monitor mode) → CHNG MODE → SET 8 (Change mode) Program mode → ∗ → ∗ → シフト SHIFT → INTL DISP → SET 8 (Initial mode)	○	○	○	×	○
Operation screen selection	Key operation: クリア CLR → ∗ 1st → ∗ 2nd → 3rd ∗ → A 0 → EL backlight ON/OFF	○	○	○	×	○
	3rd ∗ → B 1 → Buzzer ON/OFF	○	○	○	×	○
	3rd ∗ → E 4 → Monitor clock (JW-1424K/1624K)	○	○	○	×	×
	4th ∗ → シフト SHIFT → E 4 → Error monitor	×	○	○	×	×
	5th ∗ → INTL DISP → Display mode selection	○	○	○	×	×
	5th ∗ → シフト SHIFT → INTL DISP → Initial mode	○	×	×	×	×
	6th ∗ → MNTR MODE → Monitor mode	○	○	○	×	○
	6th ∗ → CHNG MODE → Change mode	○	○	○	×	○
	6th ∗ → PROG MODE → Program mode	○	○	○	×	○
	7th ∗ → 検索 SRCH → Program check	○	×	×	×	×

Function		Operation procedure	Mode				
			P	M	C	T	I
Memory clear		 ＊ ＊ PROG MODE SET 8 クリア CLR RESET 9 システム SYS 書込 ENT System memory clear アドレス ADRS 書込 ENT Program memory clear DATA CONST 書込 ENT Data memory clear 書込 ENT Program memory and data memory clear シフト SHIFT A 0 書込 ENT All initialize	○	×	×	×	×
System memory read		 クリア CLR システム SYS A 0 7 モニタ MNTR STEP (+) (Read in address incrementing direction) STEP (−) (Read in address decrementing direction) Set the system memory address	○	○	○	×	×
System memory write		 System memory write (1 byte unit) FORCE LNGTH (1 word unit) Set data (HEX, OCT, DCML, bit pattern) 書込 ENT	○	×	×	×	×
System memory check code write			○	×	×	×	×
Program address set			○	○	○	×	×
Basic and application instruction entry	Entry method for basic instructions		○	×	×	×	×
	TMR/CNT instruction entry	 TMR CNT TMR/CNT number 書込 ENT STEP (+) Setting value 書込 ENT STEP (+) Setting value changes constants or register area by pressing DATA CONST key. 0000 (Constant) コ0000 b000 09000					
	Application instruction entry	 • F-xx instruction FUN Function number F-☓☓(Fc☓☓) • Fcxx instruction FUN Function number DATA CONST F-☓☓w(Fc☓☓w) FORCE LNGTH 書込 ENT Register/constant STEP (+) 書込 ENT (Repetition of number of instruction word setting)					

Function		Operation procedure	Mode				
			P	M	C	T	I
Basic and application instruction entry	Register area selection	• Press the DATA CONST key to change the register area. ⊐0000 → b000 → 09000 → (back to ⊐0000)	○	×	×	×	×
	Indirect address designation	Register number → 変換 CONV → @Register number → 変換 CONV → (back to Register number)					
	Setting the register address	Display ⊐0000 → Address Display b000 → Address Display 09000 → Register number set (0 to 3) → , → Address → DATA CONST → (back to Register number set)					
Program write		• Write from address 00000 クリア CLR → Instruction word → 書込 ENT → STEP (+) → (back to Instruction word) • Write from a specified address Program address set → モニタ MNTR → Instruction word → 書込 ENT → STEP (+) → (back to Instruction word) • Write from an address where no program is written クリア CLR → 検索 SRCH → モニタ MNTR (Search for NOP instruction) → Instruction word → 書込 ENT → STEP (+) → (back to Instruction word)	○	×	×	×	×
Program read		• Read by specifying an address Program address setting → モニタ MNTR → STEP (+) (Read in address incrementing direction) → STEP (−) (Read in address decrementing direction) • Read by searching for an instruction Instruction search → STEP (+) (Read in address incrementing direction) → STEP (−) (Read in address decrementing direction) • Read by searching data memory Data memory search → STEP (+) (Read in address incrementing direction) → STEP (−) (Read in address decrementing direction)	○	○	○	×	×

APP

	Function	Operation procedure	Mode P	M	C	T	I
Program search	Instruction search	 クリア CLR Specify search start address Specify instruction to be searched for. 検索 SRCH Address incrementing direction Address decrementing direction STEP (−) モニタ MNTR Continuous search	○	○	○	×	×
	NOP instruction search	 クリア CLR Specify search address 検索 SRCH (Increment address) (Decrement address) STEP (−) モニタ MNTR	○	○	○	×	×
	Non-NOP instruction search	 クリア CLR Specify search address 検索 SRCH , (Increment address) (Decrement address) STEP (−) モニタ MNTR	○	○	○	×	×
	Data memory search	 Data memory address set Continuous クリア CLR DATA CONST A 0 7 Data memory address 検索 SRCH (Address incrementing direction) (Address decrementing direction) STEP (−) モニタ MNTR • Press the DATA CONST key to change the data memory area.  Relay number (00000 to 15777) TMR/CNT number (000 to 377) Byte address (⊐0000 to ⊐1577) Byte address (b000 to b777) Register (09000 to 39777) Label number (LB0000 to LB0177)	○	○	○	×	×
	Re-execution of search	 Instruction search Data memory search Program correction Searching with the changed contents (Address incrementing direction) 検索 SRCH モニタ MNTR (Address decrementing direction) 検索 SRCH STEP (−) Searching with the contents before the change (Address incrementing direction) 検索 SRCH 検索 SRCH モニタ MNTR (Address decrementing direction) 検索 SRCH 検索 SRCH STEP (−) (Continuous search)	○	○	○	×	×
Program correction	Instruction change	 Specify address to be corrected Specify instruction to be corrected 書込 ENT	○	×	×	×	×
	Instruction insert	 Specify address to be inserted Specify instruction to be inserted 挿入 INS	○	×	×	×	×
	Instruction delete		○	×	×	×	×

Function		Operation procedure	Mode				
			P	M	C	T	I
Program correction	TMR/CNT setting value change	Read setting value in program → Setting value change → 書込 ENT	○	×	○	×	×
	Application instruction constant change	Program address search → Setting value change → 書込 ENT	○	×	○	×	×
Program check		Continuous check クリア CLR → ＊ → ＊ → 検索 SRCH	○	×	×	×	×
Program monitor		Setting program address → モニタ MNTR → STEP (+) (Address incrementing direction) → STEP (−) (Address decrementing direction) Instruction search → Data memory search →	×	○	○	×	×
Data memory monitor		Continuous monitor クリア CLR → Setting data memory address → モニタ MNTR → → STEP (+) (Address incrementing direction) → → STEP (−) (Address decrementing direction) → Press the DATA CONST key to change the data memory area. Relay number (00000 to15777) → TMR/CNT number (000 to 377) → Byte address (⊐0000 to ⊐1577) → Byte address (b000 to b777) → Register (09000 to 39777) → Relay number Display ⊐0000 → Address Display b000 → Address Display 09000 → Register number set (0 to 3) → , → Address → DATA CONST →	○	○	○	×	×
Data memory change	Relay set or reset	Relay monitor → SET 8 (Set) → RESET 9 (Reset)	○	×	○	×	×
	TMR/CNT set or reset	TMR or CNT monitor → SET 8 (Set) → RESET 9 (Reset)	×	×	○	×	×
	Current register value change	Register monitor → Current value → 書込 ENT	○	×	○	×	×

Function	Operation procedure	Mode				
		P	M	C	T	I
Setting the time (JW-1424K/ 1624K)	Change to program mode → Change to initial mode → [A 0] (Select clock) → Set the "year" (Lower 2 digits of A.C.) → [,] → Set the "month" → [,] → Set the "day" → [,] → Set the "day-of-the-week" → [,] → Set the "hours" → [,] → Set the "minutes" → [,] → Set the "seconds" → [,] → [SET 8] Day of the week: SUN, MON, TUE, WED, THU, FRI, SAT Setting value: 0, 1, 2, 3, 4, 5, 6	×	×	×	×	○
Time monitor	[＊] → [＊] → [E 4] → [SET 8] (Monitor) Clear the monitor with the [クリア CLR] key.	○	○	○	×	×
Writing a program to an EEPROM	[クリア CLR] → [編集 EDIT] → [A 0] → [A 0] → [B 1] (Write) → [SET 8] → [A 0] (Stop)	○	×	×	×	×
Reading a program from ROM	[クリア CLR] → [編集 EDIT] → [A 0] → [B 1] → [B 1] (Read) → [SET 8] → [A 0] (Stop)	○	×	×	×	×
Error monitor	[＊] → [＊] → [シフト SHIFT] → [E 4] → [SET 8] → [STEP (+)] (Step incrementing direction monitor) → [STEP (−)] (Step decrementing direction monitor)	×	○	○	×	×
Password register	[クリア CLR] → [編集 EDIT] → [B 1] → [B 1] → Password register (4 digits) (0 to F) → [書込 ENT] → [A 0] (Register) → [B 1] (Stop)	○	×	×	×	×
Password delete	[クリア CLR] → [編集 EDIT] → [B 1] → [C 2] → [A 0] (Delete) → [B 1] (Stop)	○	×	×	×	×
Secret ON	[クリア CLR] → [編集 EDIT] → [B 1] → [A 0] → [A 0] (ON) → [B 1] (Stop)	○	×	×	×	×
Secret OFF	JW-13PG is connected with JW10's MMI port → Password (4 digits) (0 to F) (Secret OFF) → All initialize (Clear the system memory, program memory, and data memory)	○	○	○	×	×

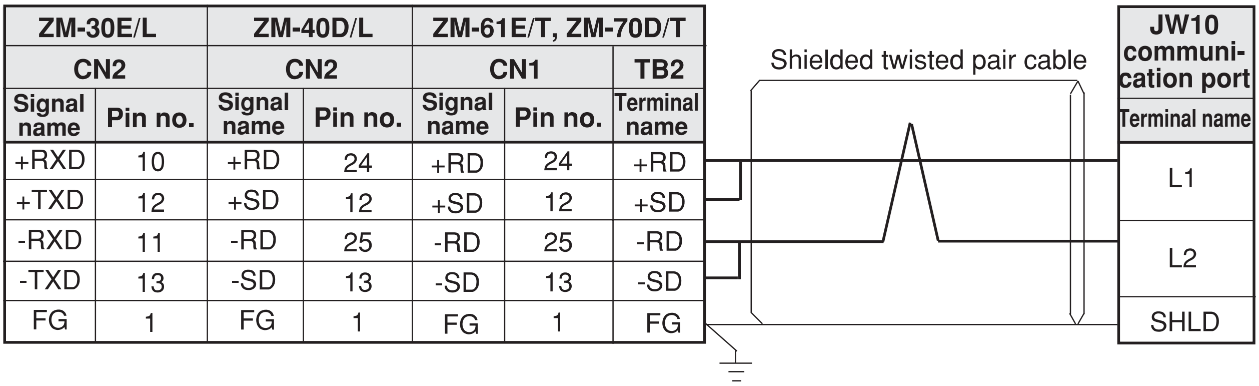
APP

Appendix-5 Connection with an LCD control terminal

Below shows wiring method and setting value of the system memory when a SHARP's LCD control terminal (ZM-30E/L, ZM-40D/L, ZM-61E/T, ZM-70D/T) is connected to the JW10.

[1] In case of connecting communication port

(1) Wiring



ZM-30E/L		ZM-40D/L		ZM-61E/T, ZM-70D/T		
CN2		CN2		CN1		TB2
Signal name	Pin no.	Signal name	Pin no.	Signal name	Pin no.	Terminal name
+RXD	10	+RD	24	+RD	24	+RD
+TXD	12	+SD	12	+SD	12	+SD
-RXD	11	-RD	25	-RD	25	-RD
-TXD	13	-SD	13	-SD	13	-SD
FG	1	FG	1	FG	1	FG

Note 1: Connection to a ZM-30E/L, ZM-40D/L, or ZM-61E/T can be done on models with version 1.13 or later ROM code.

Note 2: The termination resistance switches on the ZM-40D/L, ZM-61E/T, or ZM-70D/T must be set to "No termination resistance." If they are set to " termination resistance present," the JW10 may not communicate correctly.

(2) Setting of JW10 system memory

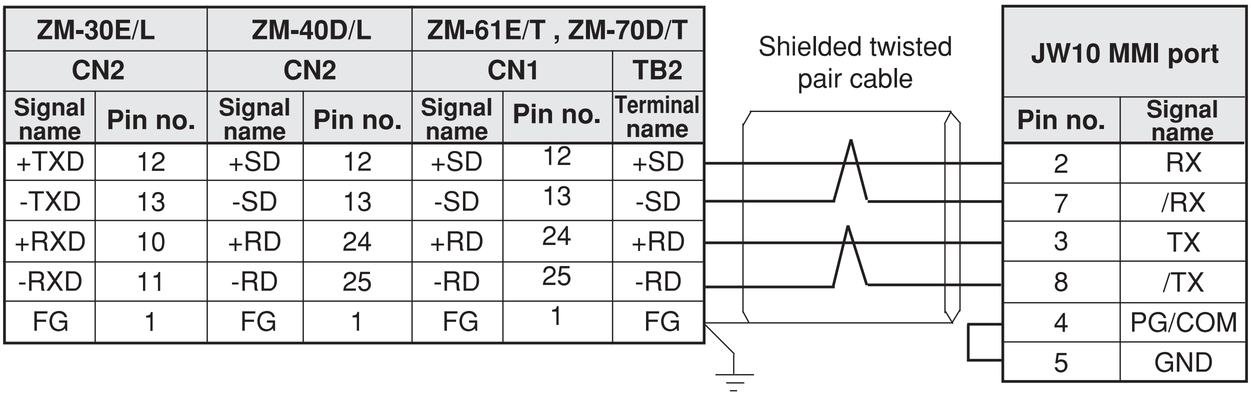
System memory no.	Setting value	Contents
#234	00(H)	Computer link mode
#236	30(H)	19200 bits/s, even parity, stop bit 2 bits, data length 7 bits
#237	001(8)	Station No. 001

(3) Setting of screen image creation software (ZM-31SE)

Item	Setting contents
Setting PLC model	SHARP (JW series)
Communication parameter	Baud rate [19200] bps., signal level [RS422]

[2] In case of connecting with MMI port

(1) Wiring



ZM-30E/L		ZM-40D/L		ZM-61E/T , ZM-70D/T		
CN2		CN2		CN1		TB2
Signal name	Pin no.	Signal name	Pin no.	Signal name	Pin no.	Terminal name
+TXD	12	+SD	12	+SD	12	+SD
-TXD	13	-SD	13	-SD	13	-SD
+RXD	10	+RD	24	+RD	24	+RD
-RXD	11	-RD	25	-RD	25	-RD
FG	1	FG	1	FG	1	FG

JW10 MMI port	
Pin no.	Signal name
2	RX
7	/RX
3	TX
8	/TX
4	PG/COM
5	GND

Note 1: The termination resistance switches on the ZM-40D/L, ZM-61E/T, or ZM-70D/T must be set to "No termination resistance." If they are set to " termination resistance present," the JW10 may not communicate correctly.

(2) Setting of JW10 system memory

System memory no.	Setting value	Contents
#226	30(H)	19200 bits/s, even parity, stop bit 2 bits, data length 7 bits
#227	001(8)	Station number 001(8)

(3) Setting of screen image creation software (ZM-31SE)

Item	Setting contents
Setting PLC model	SHARP (JW series)
Communication parameter	Baud rate [19200] bps., signal level [RS422]