MICRO^3

Micro Programmable Logic Controller

idec

User's Manual

SAFETY PRECAUTIONS

- Read this user's manual to make sure of correct operation before starting installation, wiring, operation, maintenance, and inspection of the MICRO3.
- All MICRO3's are manufactured under IDEC's rigorous quality control system, but users must add a backup or failsafe provision to the control system using the MICRO3 in applications where heavy damage or personal injury may be caused in case the MICRO3 should fail.
- In this user's manual, safety precautions are categorized in order of importance to Warning and Caution:

⚠ Warning	**Warning notices are used to emphasize that improper operation may cause severe personal injury or death.**
⚠ Caution	**Caution notices are used where inattention might cause personal injury or damage to equipment.**

⚠ Warning

- Turn power off to the MICRO3 before starting installation, removal, wiring, maintenance, and inspection on the MICRO3. Failure to turn power off may cause electrical shocks or fire hazard.
- Special expertise is required to install, wire, program, and operate the MICRO3. People without such expertise must not use the MICRO3.
- Emergency and interlocking circuits must be configured outside the MICRO3. If such a circuit is configured inside the MICRO3, failure of the MICRO3 may cause disorder of the control system, damage, or accidents.

⚠ Caution

- Install the MICRO3 according to instructions described in this user's manual. Improper installation will result in falling, failure, or malfunction of the MICRO3.
- MICRO3 is designed for installation in equipment. Do not install the MICRO3 outside of equipment.
- Install the MICRO3 in environments described in this user's manual. If the MICRO3 is used in places where the MICRO3 is subjected to high-temperature, high-humidity, condensation, corrosive gases, excessive vibrations, and excessive shocks, then electrical shocks, fire hazard, or malfunction will result.
- The pollution degree of the MICRO3 is "Pollution degree 2." Use the MICRO3 in environments of pollution degree 2 (according to IEC664-1).
- All DC power type MICRO3 units are "PS2" type (according to EN61131).
- Prevent the MICRO3 from falling while moving or transporting the MICRO3, otherwise damage or malfunction of the MICRO3 will result.
- Prevent metal fragments and pieces of wire from dropping inside the MICRO3 housing. Put a cover on the MICRO3 during installation and wiring. Ingress of such fragments and chips may cause fire hazard, damage, or malfunction.
- Use a power supply of the rated value. Use of a wrong power supply may cause fire hazard.
- Use wires of a proper size to meet voltage and current requirements. Tighten M3 terminal screws to a proper tightening torque of 0.3 to 0.5 N-m.
- Use an IEC127-approved fuse on the power line outside the MICRO3. This is required when exporting equipment containing MICRO3 to Europe.
- Use an IEC127-approved fuse on the output circuit. This is required when exporting equipment containing MICRO3 to Europe.
- Use an EU-approved circuit breaker. This is required when exporting equipment containing MICRO3 to Europe.
- Make sure of safety before starting and stopping the MICRO3 or when operating the MICRO3 to force outputs on or off. Incorrect operation on the MICRO3 may cause machine damage or accidents.
- If relays or transistors in the MICRO3 output circuit fail, outputs may remain on or off. For output signals which may cause heavy accidents, provide a monitor circuit outside of the MICRO3.
- Do not connect to the ground directly from the MICRO3. Connect a protective ground to the equipment containing MICRO3 using an M4 or larger screw. This is required when exporting equipment containing MICRO3 to Europe.
- Do not disassemble, repair, or modify the MICRO3.
- Dispose of the battery in the MICRO3 when the battery is dead in accordance with pertaining regulations. When storing or disposing of the battery, use a proper container prepared for this purpose. This is required when exporting equipment containing MICRO3 to Europe.
- When disposing of the MICRO3, do so as an industrial waste.
- Dispose of the battery in the memory card when the battery is dead in accordance with pertaining regulations.

MICRO^3 USER'S MANUAL

This user's manual primarily describes MICRO^3's entire functions shared with the MICRO^{3}C programmable controllers, such as installation instructions, general specifications, basic and advanced instructions, allocation numbers, and FUN settings. For the MICRO^{3}C additional functions not included in the MICRO^3, see the MICRO^{3}C user's manual.

MICRO^3 and MICRO^{3}C Comparison

PLC			MICRO^3	MICRO^{3}C
Advanced Instructions			38	40 (TXD, RXD, CMP2 added; ANR1 deleted)
Data Registers	Standard Processing		100 points	500 points
	High-speed Processing		32 points	32 points
Analog Potentiometers			1 point (10-I/O type) 2 points (16/24-I/O types)	1 point
Communication Specifications	Loader Port	Standards	EIA RS485	EIA RS232C
	Data Link Terminal	Standards	EIA RS485	EIA RS485
		Baud Rate	Expansion/data link communication: 19,200 bps (fixed)	Expansion/data link communication: 19,200 bps (fixed) Loader protocol communication: 9,600 bps (fixed)
Weight (approx.)			290g (10-I/O type) 350g (16-I/O type) 390g (16-I/O AC input type) 400g (24-I/O type)	380g (16-I/O type) 430g (24-I/O type)
Standards			EN61131-1, EN61131-2, EN60204-1 IEC801-2, -3, -4 PrEN50082-2, EN55011 UL508, CSA C22.2, No. 142	EN55011 Group 1, Class A EN50082-2 UL508, CSA C22.2, No. 142 EN61131-1, EN61131-2, EN60204-1
Certification File No.			TÜV Product Service E9 95 09 13332 313 UL E102542 CSA LR66809	TÜV Product Service B950913332 UL E102542 CSA LR66809

Program Loader for MICRO^3

To edit user programs for the MICRO^3, read FUN11 (program capacity and PLC type selection) on the program loader, and set the fourth line in the FUN11 screen to 0 to select MICRO^3 as the PLC type, using the FUN11, ▼, ▼, ▼, 0, and ↵ keys. When FUN11 is set to 0, available data registers are limited to D0 through D99 for programming the MICRO^3.

Since the loader port on the MICRO^3 uses RS485 communication while the loader port on the MICRO^{3}C uses RS232C, a different loader cable is needed to connect the program loader to MICRO^3 or MICRO^{3}C. Use loader cable FC2A-KL1 or FC2A-KL2 to connect a program loader to the MICRO^3 loader port.

To use the expanded capabilities of the MICRO^{3}C such as new advanced instructions for communication and comparison and increased data registers, use an upgraded program loader of version 2.00 or later. To check the program loader version, read FUN31 (program loader version readout/hardware check) using the FUN31 and ▼ keys on the program loader.

To connect a program loader to the MICRO^{3}C loader port, use loader cable 3C (FC2A-KL3C). A program loader can also be connected to the data link terminals on the MICRO^{3}C using loader cable 4C (FC2A-KL4C). In either case, loader protocol must be selected for the loader port or data link terminals using the protocol selector switch. For selection of the protocol selector switch, see the MICRO^{3}C user's manual.

IMPORTANT INFORMATION

Under no circumstances shall IDEC Corporation be held liable or responsible for indirect or consequential damages resulting from the use of or the application of IDEC PLC components, individually or in combination with other equipment.

All persons using these components must be willing to accept responsibility for choosing the correct component to suit their application and for choosing an application appropriate for the component, individually or in combination with other equipment.

All diagrams and examples in this manual are for illustrative purposes only. In no way does including these diagrams and examples in this manual constitute a guarantee as to their suitability for any specific application. To test and approve all programs, prior to installation, is the responsibility of the end user.

CHAPTER 5: CPU CONFIGURATION (FUN)

CHAPTER 6: ALLOCATION NUMBERS

CHAPTER 7: BASIC INSTRUCTIONS

CHAPTER 8: ADVANCED INSTRUCTIONS

CHAPTER 9: MOVE INSTRUCTIONS

CHAPTER 10: COMPARISON INSTRUCTIONS

CHAPTER 11: BINARY ARITHMETIC INSTRUCTIONS

CHAPTER 12: BOOLEAN COMPUTATION INSTRUCTIONS

CHAPTER 13: BIT SHIFT / ROTATE INSTRUCTIONS

CHAPTER 14: CLOCK / CALENDAR INSTRUCTIONS

CHAPTER 15: INTERFACE INSTRUCTIONS

CHAPTER 16: PULSE, A/D CONVERSION INSTRUCTIONS

CHAPTER 17: HIGH-SPEED COUNTER INSTRUCTIONS

CHAPTER 18: TROUBLESHOOTING

APPENDIX

INDEX

Introduction

This chapter describes general information for understanding MICRO^3 functions and specifications.

Features

MICRO^3 is a space-saving micro programmable controller, yet has high-performance functions described below:

High-speed Processing Function

MICRO^3 operates in standard processing mode or high-speed processing mode. Standard mode has a program capacity of 1012 steps, minimum processing time of 1.2 µsec per basic instruction, and average scan time of 2.9 msec for 1,000 steps.

High-speed processing mode has a program capacity of 100 steps, minimum processing time of 0.2 µsec per basic instruction, and average scan time of 400 µsec for 100 steps. Data and expansion link cannot be used with high-speed processing.

Catch Input Function

The catch input function makes sure to receive short input pulses (40 µsec minimum at the rising edge) from sensors without regard to the scan time.

The 10-I/O type MICRO^3 base unit can receive short pulse inputs at 6 input terminals (I0 through I5). The 16- and 24-I/O type base units can use 8 input terminals (I0 through I7) for catch inputs.

Input Filter Function (DC Input Type Only)

The input filter can be adjusted for input signal durations. This function is useful for adjusting the input pulse width in sensor controller applications and for eliminating input noises and chatter in limit switches.

High-speed Counter Function

MICRO^3 has a built-in high-speed counter to make it possible to count up to 4,294,967,295 (FFFF FFFFh) high-speed pulses which cannot be counted by the normal user program processing. The maximum count input frequency is 10 kHz. This function can be used for simple positioning control and simple motor control.

Pulse Output Function

Pulse outputs can be generated using advanced instructions. The PULS (pulse output) instruction can vary the output frequency at a fixed duty ratio of 50%. This instruction can be used in combination with the high-speed counter function to control servo motors and stepper motors. The PWM (pulse width modulation) instruction can change the duty ratio at a fixed frequency. This instruction can be used for illumination control.

Computer Link Function

A personal computer can be connected to MICRO^3 in 1:1 peer-to-peer or 1:N network communication system to monitor the operating statuses and change data in MICRO^3. CUBIQ software is available for easy programming and monitoring.

Expansion Link Function

The I/O points can be expanded from 6 inputs and 4 outputs up to 28 inputs and 20 outputs by connecting another MICRO^3 in six combinations, maximizing flexibility, (see note).

Data Link Function

A maximum of seven MICRO^3 base units (1 master station and 6 slave stations) can be linked in the data link network for distributed control. FA-3S high-performance CPU can also be used as a master station in the data link system (see note).

Real-time Clock/Calendar Function (16- and 24-I/O Type Units Only)

The 16- and 24-I/O type MICRO^3 base units feature a real-time clock/calendar to program year, month, day, day of week, hour, minute, and second using advanced instructions. This function can be used for time-scheduled control of illumination, air conditioners, sprinklers, and many others.

External Analog Timer

MICRO^3 is equipped with one or two analog potentiometers to enter analog values. In addition, a separate analog timer unit can also be connected to MICRO^3, allowing for fine adjustment of timer preset values on the control panel.

User Program Read and/or Write Protection

User programs in the MICRO^3 base unit can be protected from reading and/or writing by setting a pass word. This function is ideal for the security of user programs and prevention of inadvertent rewriting of programs.

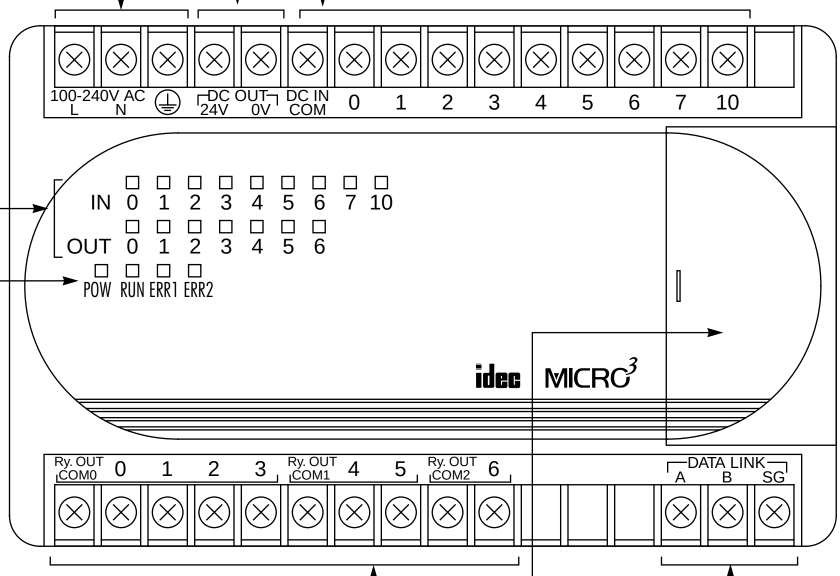
Analog I/O Function

A/D and D/A converter units are available for 8-bit conversion to process analog signals.

Note: Expansion link and data link cannot be used concurrently.

Parts Description

This section describes parts names and functions of the MICRO^3 base unit.



Sensor Power Terminals
For supplying power to sensors (24V DC, 150mA). These terminals have an overload sensing function.

Input Terminals
For connecting input signals from input devices such as sensors, pushbuttons, and limit switches. Available in DC input (sink/source) and AC input (100-120V AC) types.

Power Supply Terminals
Connect power supply to these terminals. Power voltage 100-240V AC or 24V DC.

I/O Indicators
Turn on when the corresponding inputs or outputs are on.

Power Indicator (POW)
Turns on when power is supplied to MICRO^3.

Operation Indicator (RUN)
Turns on when MICRO^3 is running.

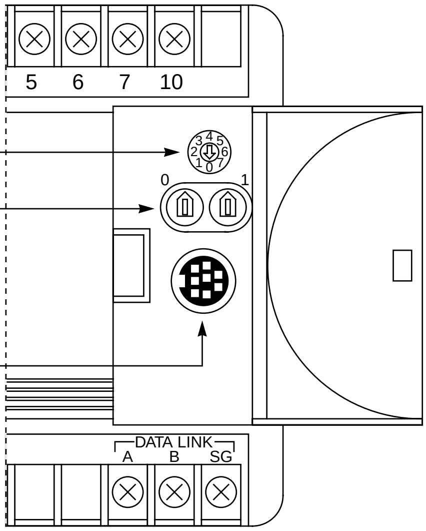
Error Indicator 1 (ERR1)
Turns on when an error occurs in MICRO^3.

Error Indicator 2 (ERR2)
Turns on when the sensor power is overloaded.

Output Terminals
For connecting output signals to output devices such as electromechanical relays and solenoid valves. Available in relay output (240V/2A, 30V DC/2A), transistor sink output (24V DC, 0.5A), transistor protect source output (24V DC, 0.5A) types.

Data Link Terminals
For connecting the data link line in the expansion link or data link system.

Pop-up Lid
Press the lid to open and gain access to the function selector switch, analog potentiometer, and loader port.



Function Selector Switch
Selects the station function in the expansion or data link system.

Analog Potentiometer
Sets the analog value for the analog timer, frequency or pulse width of pulse outputs. The 10-I/O type has one potentiometer. The 16- and 24-I/O types have two potentiometers; analog potentiometer 0 on the left and analog potentiometer 1 on the right.

Loader Port
For connecting the program loader or computer.

The figures above and at right illustrate the 16-I/O type MICRO^3 base unit with DC input and relay output.

◆ ***For the MICRO^{3}C, see the MICRO^{3}C User's Manual.*** ◆

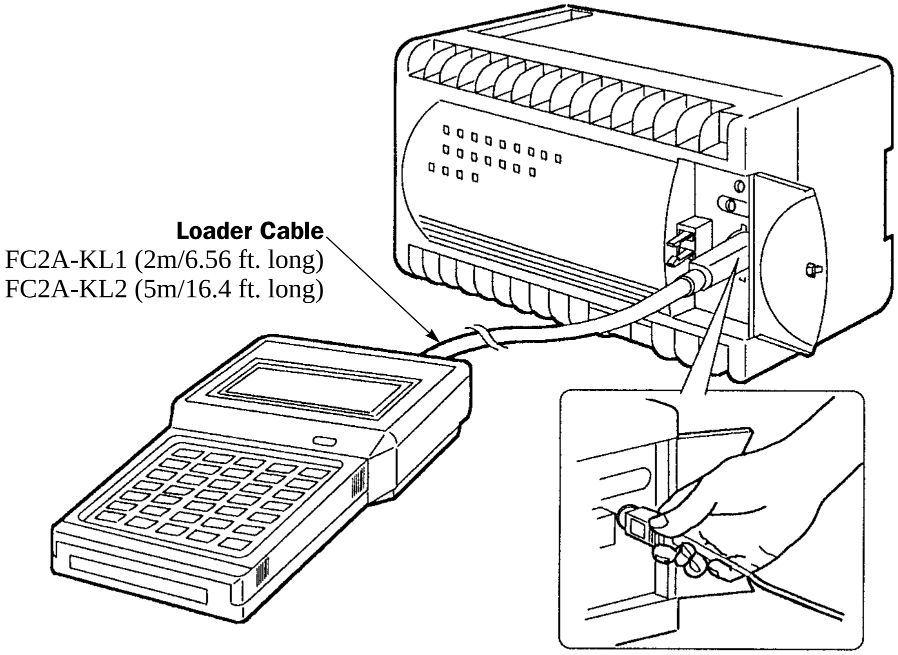
System Setup

This section describes settings and precautions for the basic system, expansion system, and various link systems consisting of MICRO^3.

Basic System

The basic system consists of the base unit and the program loader. This system is used to edit a user program on the program loader, transfer the user program to the base unit, start and stop the base unit operation, and monitor the operating status.

◆ ***For the MICRO^{3}C, see the MICRO^{3}C User's Manual.*** ◆

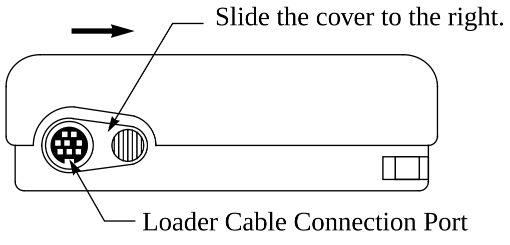


When connecting and disconnecting the loader cable, be sure to hold the connector. Since the connector has a latch, the cable cannot be removed by holding the cable.

Connecting the Cable

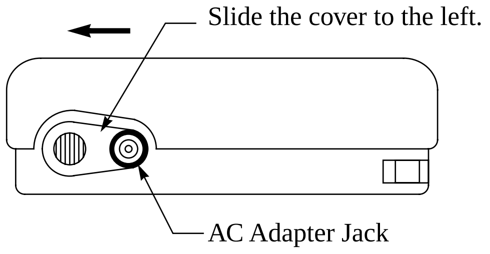
The program loader has a cover on the top to select the loader cable connection port or AC adapter jack. Slide the cover to the right to open the loader cable connector.

Connect the connector of the loader cable to the loader cable connection port on the program loader and the other connector of the cable to the loader port on the MICRO^3 base unit.



Off-line Programming

The program loader can be used off-line for remote programming. Slide the cover to the left to open the AC adapter jack of the program loader and connect an AC adapter to the AC adapter jack. For the power supply requirements and the plug dimensions, see page A-4.



Programming Tool

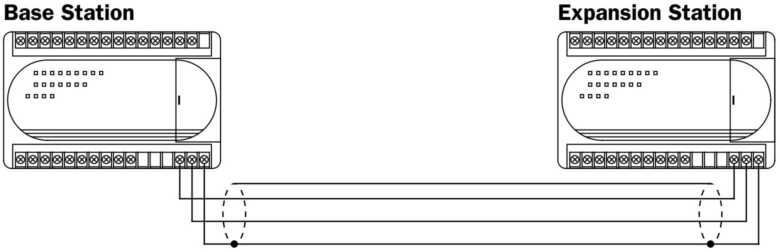
In addition to the program loader, optional software CUBIQ is available for editing user programs on a personal computer. See page 4-16.

Link Systems

MICRO^3 has three link functions; expansion link, data link, and computer link. When using a link function, the function selector switch may have to be set or the FUN settings may be required. For details of these settings, see Expansion Link Function on page 4-6, Data Link Function on page 4-8, and Computer Link 1:N Communication on page 4-17. The expansion link cannot be used in the data link system.

Expansion Link System

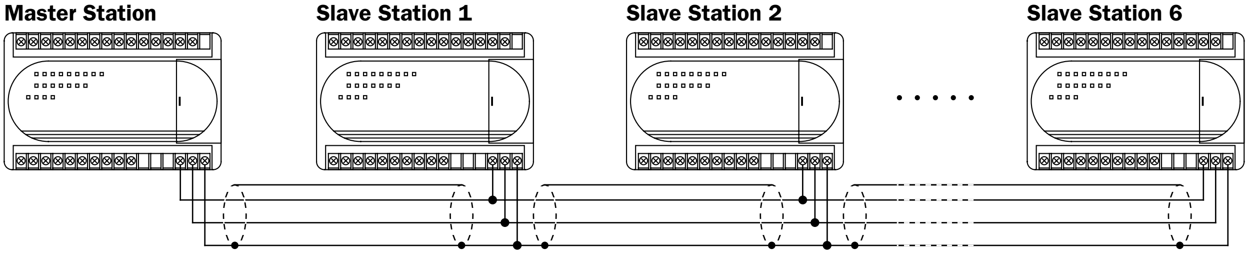
The expansion link system consists of two MICRO^3 base units connected through the data link terminals using the optional expansion cable FC2A-KE1 (250 mm/9.84" long) or a shielded twisted pair cable as shown below. The cable for the expansion link system can be extended up to 200 meters (656 feet). Every MICRO^3 base unit can be used as an expansion station.



The RUN indicator on the expansion station remains off whether the base station is running or stopped.

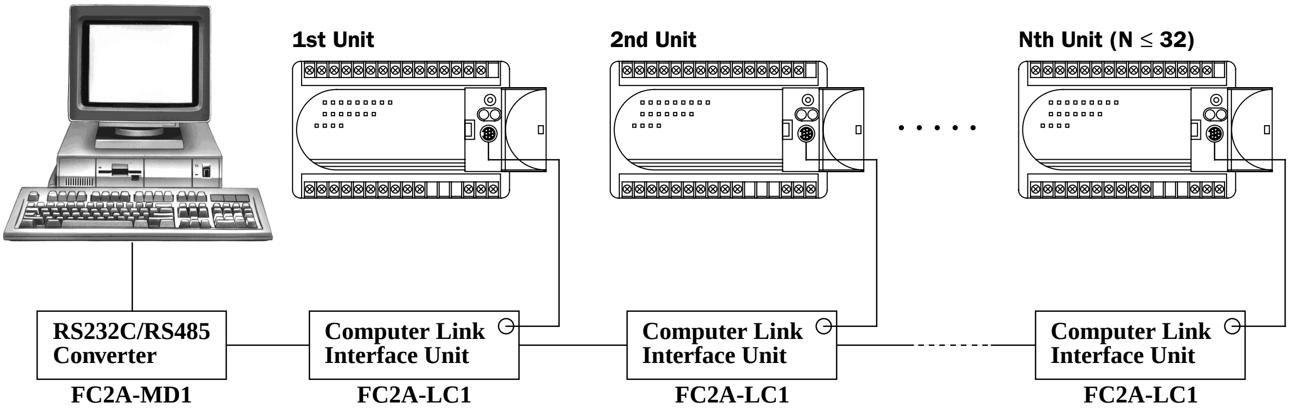
Data Link System

The data link system consists of one master station connected to a maximum of six slave stations to communicate control data for distributed control. Every MICRO^3 base unit can be used as a master or slave station. When a slave station performs communication at 19,200 bps through the loader port, multi-stage comparison instruction HSC1 cannot be used at the slave station.



Computer Link System

In the computer link system, a personal computer is connected to one or a maximum of 32 MICRO^3 base units to control the operation of all MICRO^3 base units. The 1:1 computer link system requires the computer link cable FC2A-KC2. The 1:N computer link system requires computer link interface unit FC2A-LC1 and RS232C/RS485 converter FC2A-MD1 in addition to three types of cables.



General Specifications

Type		AC Power	DC Power
Power Supply	Rated Power Voltage	100 to 240V AC	24V DC
	Allowable Voltage Range	85 to 264V AC	19 to 30V DC (including ripple)
	Dielectric Strength	Between power terminal and FG: 2000V AC, 1 minute Between I/O terminal and FG: 1500V AC, 1 minute	Between power terminal and FG: 1500V AC, 1 minute Between I/O terminal and FG: 1500V AC, 1 minute
	Repetitive Peak Current	Approx. 310 mA (maximum at 85V)	———
	Input Current	Approx. 220 mA (maximum at 85V)	Approx. 500 mA (maximum at 19V)
	Rated Frequency	50/60 Hz (47 to 63 Hz)	———
	Power Consumption	Approx. 30 VA (240V AC)	Approx. 14W (24V DC)
	Allowable Momentary Power Interruption	25 msec (100V)	25 msec (24V), Level PS-2
	Insulation Resistance	Between power terminal and FG: 10 MΩ minimum (500V DC megger) Between I/O terminal and FG: 10 MΩ minimum (500V DC megger)	
	Inrush Current	40A maximum	
	Ground	Grounding resistance: 100Ω maximum	
	Protective Ground	Allowable current 10A maximum, 10 sec	
	Grounding Wire	1.25 mm^2 (AWG16)	
Effect of Improper Power Supply Connection	Reverse Polarity	No trouble	No operation, no damage
	Improper Voltage or Frequency	Permanent damage may be caused	
	Improper Lead Connection	Connection failure may be caused	
Power Up/Down Order		AC or DC main power must be turned on not later than I/O power. AC or DC main power must be turned off not earlier than I/O power.	
Memory Backup	Backup Duration	Without clock/calendar (10-I/O type): Approx. 50 days at 25°C With clock/calendar (16/24-I/O types): Approx. 30 days at 25°C (after backup battery fully charged)	
	Battery	Lithium secondary battery	
	Charging Speed	Approx. 2 hours from 0% to 90% of full charge	
	Backup Subjects	Internal relays, shift registers, counters, data registers, clock/calendar	
	Replaceability	Impossible	
	User Program Storage	EEPROM	
Others		IEC1131-2 3.2.3.4) Non-standard power supply cannot be connected	

General Specifications, continued

Operating Temperature	0 to 60°C	
Storage Temperature	–25 to +70°C	
Relative Humidity	Relative humidity severity level RH1, 30 to 95% (non-condensing)	
Pollution Degree	2 (IEC 664)	
Corrosion Immunity	Free from corrosive gases	
Altitude	Operation: 0 to 2,000m (0 to 6,565 feet) Transport: 0 to 3,000m (0 to 9,840 feet)	
Vibration Resistance (IEC 68-2-6)	5 to 55 Hz, 60 m/sec^2, 2 hours each in 3 axes	
Shock Resistance (IEC 62-2-27)	300 m/sec^2, 11 msec, 3 shocks each in 3 axes	
I/O Duty Ratio	All specification values are determined at an I/O duty ratio of 100% for the 10-I/O type and at an I/O duty ratio of 80% for the 16- and 24-I/O types	
Wiring	Core wire 0.75 to 1.25 mm^2 (AWG18 to AWG16) Input lines must be separated from power, output, and motor lines M3 screw terminal	
Degree of Protection	IP40 (IEC 529), provided with finger protection covers	
Installation	35-mm-wide DIN rail and wall mount In either case, MICRO^3 must be mounted on a vertical plain	
Dimensions	105W × 85H × 60D mm — 4.134"W × 3.346"H × 2.362"D (10-I/O type) 135W × 85H × 60D mm — 5.315"W × 3.346"H × 2.362"D (16-I/O type) 165W × 85H × 60D mm — 6.496"W × 3.346"H × 2.362"D (16-I/O AC input type) 165W × 85H × 60D mm — 6.496"W × 3.346"H × 2.362"D (24-I/O type)	
Weight	**MICRO^3** Approx. 290g (10-I/O type) Approx. 350g (16-I/O type) Approx. 390g (16-I/O AC input type) Approx. 400g (24-I/O type)	**MICRO^{3}C** Approx. 380g (16-I/O type) Approx. 430g (24-I/O type)
Standards	**MICRO^3** EN61131-1, EN61131-2, EN60204-1 IEC801-2, -3, -4 PrEN50082-2, EN55011 UL508, CSA C22.2 No. 142	**MICRO^{3}C** EN55011 Group 1, Class A EN50082-2 UL508, CSA C22.2 No. 142 EN61131-1, EN61131-2, EN60204-1
Certification File No.	**MICRO^3** TÜV Product Service E9 95 09 13332 313 UL E102542 CSA LR66809	**MICRO^{3}C** TÜV Product Service B950913332 UL E102542 CSA LR66809

Function Specifications

Mode		Standard Processing	High-speed Processing
Program Capacity		1012 steps	Approx. 100 steps
User Program Memory		EEPROM, RAM (backed up by battery)	
Backup Function		A user program is transferred from the program loader through the CPU to RAM and EEPROM in the MICRO^3. The user program and data in the RAM are backed up by a lithium secondary battery. If the contents in the RAM are destroyed after a power failure longer than the specified value, the user program is transferred from the EEPROM to the RAM automatically at power up, and is not erased. However, since data is destroyed, the user is alerted with an error message (keep data sum check error, etc.).	
Control System		Stored program system (not in compliance with IEC1131-3)	
Programming Method		Logic symbol	
Instruction Words	**Basic Instruction**	28 basic instructions LOD, LODN, OUT, OUTN, SET, RST, AND, ANDN, OR, ORN, AND LOD, OR LOD, BPS, BRD, BPP, TIM, CNT, CC=, CC≥, SFR, SFRN, SOTU, SOTD, JMP, JEND, MCS, MCR, END	
	Advanced Instruction	**MICRO^3: 38 advanced instructions** NOP, MOV, MOVN, IMOV, IMOVN, CMP=, CMP<>, CMP<, CMP>, CMP<=, CMP>=, ADD, SUB, MUL, DIV, ANDW, ORW, XORW, SFTL, SFTR, ROTL, ROTR, CLS4, CALW, CLKR, CLKW, ADF, DISP, DGRD, ANR0, ANR1, PULS, PWM, A/D, HSC0, HSC1, HSC2, HSC3 **MICRO^{3}C: 40 advanced instructions** NOP, MOV, MOVN, IMOV, IMOVN, CMP=, CMP<>, CMP<, CMP>, CMP<=, CMP>=, ADD, SUB, MUL, DIV, ANDW, ORW, XORW, SFTL, SFTR, ROTL, ROTR, CLS4, CALW, CLKR, CLKW, ADF, DISP, DGRD, ANR0, PULS, PWM, A/D, HSC0, HSC1, HSC2, HSC3, TXD, RXD, CMP2	
I/O	**Input Points**	6 points (10-I/O type), 9 points (16-I/O type), 14 points (24-I/O type)	
	Output	4 points (10-I/O type), 7 points (16-I/O type), 10 points (24-I/O type)	
	Expansion I/O	One expansion station can be added. Maximum I/O is 48 points.	———
Data Link		Possible with 6 slave stations	———
Scan Time		2.9 msec average/1K steps	400 µsec average/100 steps
Processing Time (basic instruction)		2.2 µsec average	0.45 µsec average
Internal Relay		232 points	40 points
Data Register	**MICRO^3**	100 points	32 points
	MICRO^{3}C	500 points	
Control Data Register		10 points (designated from data registers)	———
Shift Register		64 points	32 points
Counter/Timer		32 points total	16 points total
Counter/ Timer Presets	**Adding Counter**	0 to 9999	
	Reversible Counter	0 to 9999	
	1-msec Timer	1 msec to 9.999 sec	
	10-msec Timer	10 msec to 99.99 sec	
	100-msec Timer	100 msec to 999.9 sec	
Catch Input Relay		8 points	
Special Internal Relay		16 points	
Catch Input	**Points**	8 points	
	Must Turn ON Pulse	40 µsec minimum (when hard filter is set to 10)	
	Must Turn OFF Pulse	150 µsec minimum (when hard filter is set to 10)	

Function Specifications, continued

Mode		Standard Processing	High-speed Processing
High-speed Counter	Points/Phase	1 point, single-phase	
	Preset Value	HSC0, HSC1, HSC2: 0 to 4,294,967,295 HSC3: 0 to 65535	
	Frequency Response	HSC0 and HSC3: 10 kHz HSC1 and HSC2: 5 kHz	
Pulse Output		1 channel (available on transistor output types only)	
Analog Potentiometer	MICRO^3	1 point (10-I/O type), 2 points (16/24-I/O types), Converted value: 0 to 249	
	MICRO^{3}C	1 point, Converted value: 0 to 249	
Real-time Clock/ Calendar	Accuracy	±30 sec/month at 25°C (typical)	
	Backup Duration	30 days at 25°C (typical)	
	Calendar Function	Year, month, day, day of week, hour, minute, second	
Sensor Power Supply	Output Voltage/Current	Available on AC power, DC input types only 24V ±3.6V DC, 150 mA maximum including input signal current	
	Overload Detection	Overload detection current: 190±40 mA	
	Isolation	Isolated from the internal circuit	
Self-diagnostic Function	System Initialization	Keep data sum check	
	Internal Processing A	Power failure check, WDT (watchdog timer) check, user program sum check, sensor power overload check, clock error check, LED indicator data update	
	(Read Inputs)	Update input data	
	Internal Processing B	Processed only once immediately after starting to run: User program CRC check, timer/counter preset value CRC check	
	Execute Program	Execute the user program	
	(Update Outputs)	Update outputs	
	Loader Communication	User program syntax check, user program writing check	
	Data Link Communication	Data link connection check	
Execution Time	Basic Instruction Execution time	2.2 µsec average	0.45 µsec average
	Basic Processing (Processing A/B + Determination)	200 µsec	220 µsec
	I/O Processing	130 µsec (update inputs and outputs)	
	Expansion Link	9 to 10 msec	
	Control Data Register Service	15 µsec when all control data registers are enabled	
	Clock/Calendar Processing	Processed at every 500 msec	
	Data Link Master Station Processing	12.5 to 13 msec when using data link function	
Start/Stop Method		Turning power on and off. Using the RUN/STOP switch on the program loader. Turning special internal relay M300 on and off. Turning designated stop or reset input off and on.	
Restart	Cold Restart	Possible to restart using program loader, power supply, or special internal relay (response time: 1 sec maximum)	
	Hot Restart	Impossible because timer data cannot be maintained	
	Warm Restart	Possible using a user program	
Stop/Reset Using External Signal		Possible using inputs I0 through I15 designated as a stop or reset input	

Communication and Noise Specifications

Loader Port Communication Specifications

◆ ***For the MICRO^{3}C specifications, see the MICRO^{3}C User's Manual.*** ◆

Standards		EIA RS485 (termination resistor is not required)
Connection to Program Loader		Using optional loader cable (FC2A-KL1 or FC2A-KL2)
Cable	**1:1 Communication**	Using optional computer link cable FC2A-KC2
	1:N Communication	Using ø0.9-mm shielded twisted pair cable
	Conductor Resistance	85 Ω/km maximum
	Shield Resistance	12 Ω/km maximum
Slave Stations		32 maximum in the 1:N network communication
Maximum Cable Length		200m (656 ft.) between RS232C/RS485 converter and most distant station
RS232C/RS485 Converter		FC2A-MD1
Computer Link Interface Unit		FC2A-LC1

Data Link Terminal Communication Specifications

◆ ***For the MICRO^{3}C specifications, see the MICRO^{3}C User's Manual.*** ◆

Standards	EIA RS485 (termination resistor is not required)
Recommended Cable	ø0.9 mm shielded twisted cable
Conductor Resistance	85 Ω/km maximum
Shield Resistance	12 Ω/km maximum
Maximum Cable Length	200m (656 ft.)
Isolation	Between data link terminals of multiple MICRO^{3}C units: Not isolated
Baud Rate	19200 bps (fixed)
Communication Delay	Expansion link: Master station normal scan time + approx. 9 to 10 msec Data link: Master station normal scan time + approx. 12.5 to 13 msec + Slave station scan time

Noise Immunity Specifications

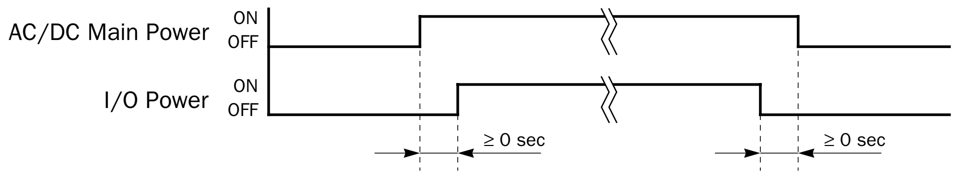
Electrostatic discharge (IEC 801-2)	
RH-1/ESD-3	Level 3 (8 kV)
Field withstandability (IEC 801-3)	
	Level 3 (10 V/m)
Damped oscillatory wave withstandability (IEC 801-4)	
Power supply	1 kV
Digital I/Os	1 kV
Fast transient withstandability (IEC 801-4)	
Power supply	Level 3 (2 kV)
Digital I/Os	Level 4 (2 kV) Analog I/Os, Communication I/Os: Level 3 (1 kV)

Noise Emission Specifications

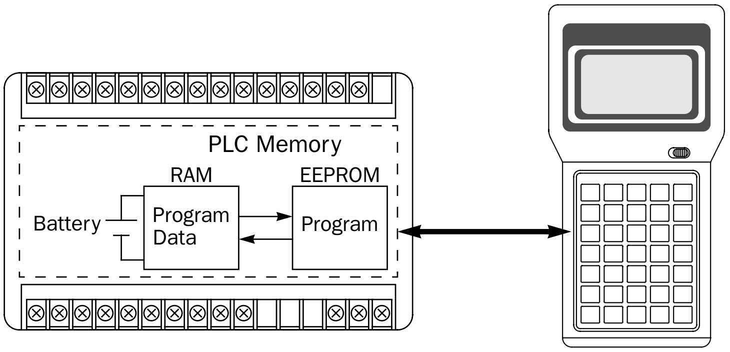
Radiated emission (EN55011)	
	Group 1 class A
Line conduction (EN55011)	
	Group 1 class A

Power Supply Timing Chart

Turn on AC or DC main power and I/O power at the same time, or turn on AC or DC main power first.
Turn off AC or DC main power and I/O power at the same time, or turn off I/O power first.



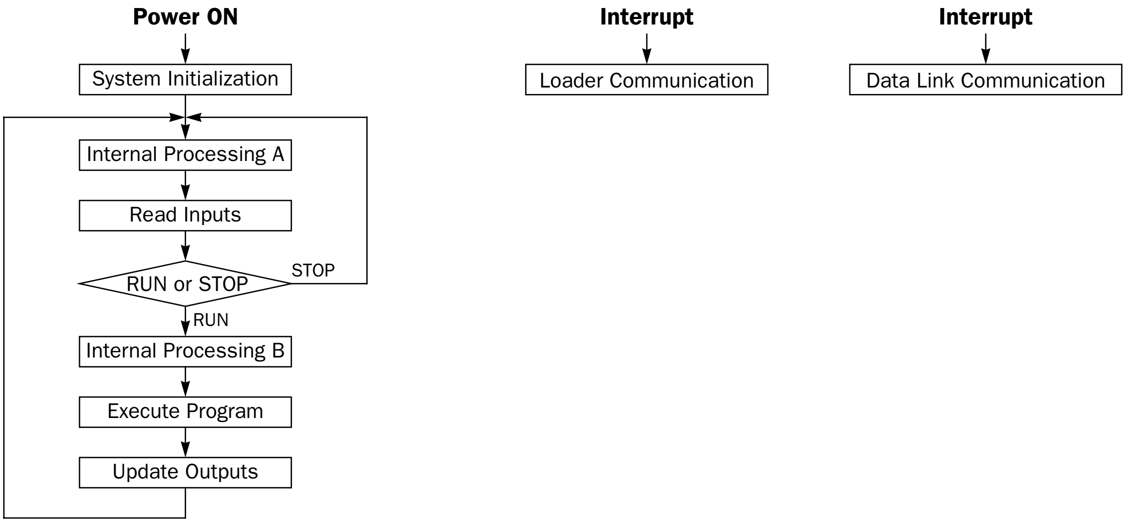
Memory Backup Function



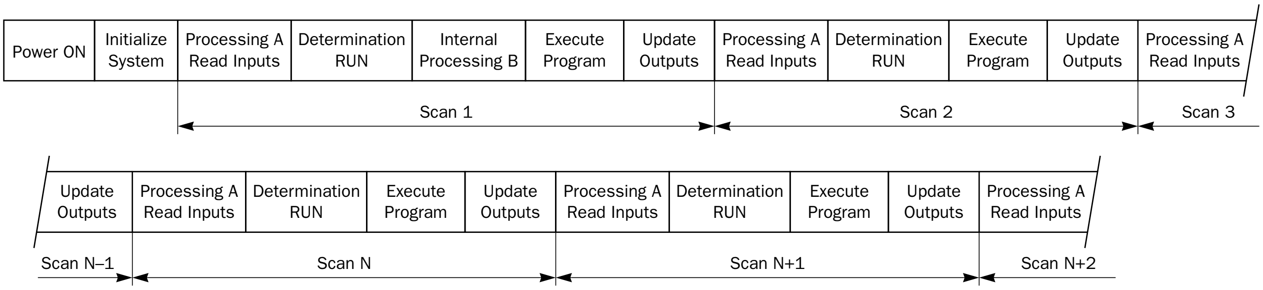
The user program and data stored in the RAM are backed up by a lithium secondary battery.

When the contents in the RAM are destroyed after a power failure longer than the specified value, the user program is transferred from the EEPROM to the RAM at power up automatically, so the user memory is not erased. Since the data is destroyed, an error message, such as keep data sum check error, is evoked to alert the user.

Self-diagnostics Flow Chart



Scanning Process and WDT (Watch Dog Timer)



When the scan time is longer than the WDT preset value (300 msec), error indicator ERR1 flashes and the PLC stops operation.

Digital DC Input Specifications

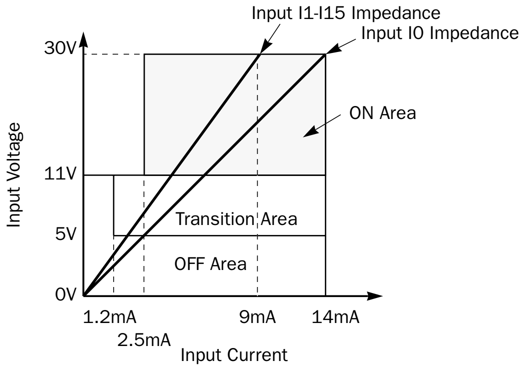
Rated Input Voltage		24V DC sink/source input signal
Input Voltage Range		19 to 30V DC
Input Impedance		I0 and I20: 2.1 kΩ I1 to I15, I21 to I35: 3.5 kΩ
Turn ON Time		I0: 4 µsec + filter preset I1 to I15: 20 µsec + filter preset I20 to I35: 3 msec
Turn OFF Time		I0: 6 µsec + filter preset I1 to I15: 120 µsec + filter preset I20 to I35: 3 msec
Common and Input Points		10-I/O type: 6 input points connected in 1 common line 16-I/O type: 9 input points connected in 1 common line 24-I/O type: 14 input points connected in 1 common line
Isolation		Between input terminals: Not isolated Internal circuit: Photocoupler isolated
Input Type		Type 1 (IEC 1131)
External Load for I/O Interconnection		Not needed
Signal Determination Method		Static
Input Filter	**Filter Function**	The meaning of the filter values shown below: Inputs accept signals of the pulse width shown.
	Soft Filter	0 msec, 3 msec (default), 7 msec, 10 msec [Setting] I0 and I1: Independently set I2 and I3: Set in combination I4 to I7: Set in combination Input reject pulse width: (Soft filter value) – 2 msec
	Hard Filter	I0: 4 to 616 µsec (ON pulse) I0: 6 to 618 µsec (OFF pulse) I1 to I7: 20 to 625 µsec (ON pulse) I1 to I7: 120 to 618 µsec (OFF pulse) [Setting] I0 to I7: Set in combination, preset 0 to 255 (default 10) I10 to I35: 3 msec (fixed) Input reject pulse width: (Hard filter value)/3 µsec
	Factory Initial Setting	Default value
Effect of Improper Input Connection		Both sinking and sourcing input signals can be connected. If any input exceeding the rated value is applied, permanent damage may be caused.
Cable Length		3m (9.84 ft.) in compliance with electromagnetic immunity
Others (IEC 1131-2 Information)		IEC 1131-2 3.3.1.4 8) Input part cannot be replaced because of non-modular structure

Digital AC Input Specifications

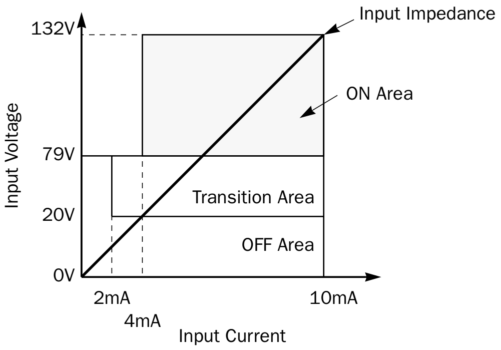
Rated Input Voltage	100 to 120V AC
Input Voltage Range	85 to 132V AC
Input Impedance	13 kΩ at 60 Hz
Turn ON Time	20 msec maximum
Turn OFF Time	20 msec maximum
Common and Input Points	16-I/O type: 9 input points connected in 1 common line
Isolation	Between input terminals: Not isolated Internal circuit: Photocoupler isolated
Input Type	Type 1 (IEC 1131)
External Load for I/O Interconnection	Not needed
Signal Determination Method	Operation using system program

Input Operating Range

Digital DC Input

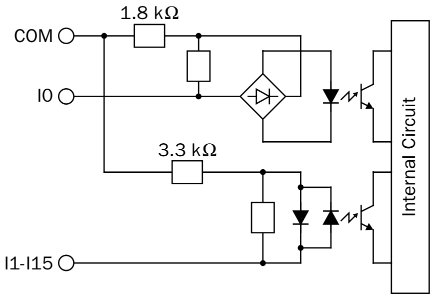


Digital AC Input

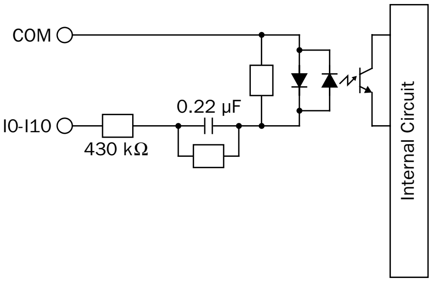


Input Internal Circuit

Digital DC Input



Digital AC Input



Digital AC/DC Output (Relay Output) Specifications

I/O Type		10-I/O Type	16-I/O Type	24-I/O Type
Output Protection		Without protection		
Protection Circuits Prepared by User		See page 1-17.		
Output Points		4 points	7 points	10 points
Output Points per Common Line	**COM0**	Common NO 3 points	Common NO 4 points	
	COM1	Independent NO 1 point	Common NO 2 points	Common NO 4 points
	COM2	—	Independent NO 1 point	Independent NO 1 point
	COM3	—	—	Independent NO 1 point
Output Terminal Ratings (Relay Contact Capacity)	**TÜV**	IEC 255-0-20 (DIN VDE 0435 part 120) IEC 255-1-00 (DIN VDE 0435 part 201) IEC 947-5-1 240V AC, 2A (RES) 240V AC, 1.5A (ø=0.4) 30V DC, 2A (RES) 240V AC, 1.5A (AC-15)		
	UL/CSA	UL508/C22.2 No. 14 240V AC, 2A (RES) 30V DC, 2A (RES)		
COM Current		Total of all contacts in a COM circuit: 8A maximum		
Minimum Switching Load		1 mA/5V DC (reference value)		
Initial Contact Resistance		30 mΩ maximum		
Electrical Life		100,000 operations minimum (rated load 1,800 operations/hour)		
Mechanical Life		20,000,000 operations minimum (no load 18,000 operations/hour)		
Isolation	**Between Output Terminal and FG**	1,500V AC		
	Between Output Terminal and Internal Circuit	1,500V AC		
	Between Output Terminals of Different COM Points	1,500V AC		
Effect of Improper Connection		When a current larger than the rated current flows, permanent damage such as contact welding may be caused.		
Output Status by MPU Operation	**Stop**	OFF		
	Power Interruption over 25 msec	OFF		
	Power Interruption 25 msec or less	ON/OFF status maintained		
	Power Up	OFF until MPU starts to run		
Others (IEC 1131-2 Information)		IEC 1131-2 3.3.2.3 3) Not zero-cross switching IEC 1131-2 3.3.2.3 9) Not tested for other categories IEC 1131-2 3.3.2.3 10) Not multichannel module IEC 1131-2 3.3.2.3 11) Suppressor networks are not incorporated into the relay output circuit IEC 1131-2 3.3.2.3 16) Output can not be replaced because of non-modular structure IEC 1131-2 3.3.2.3 18) Non-latching type output operation IEC 1131-2 3.3.2.3 19) Not multi-circuit module		

Digital DC Output (Transistor Sink Output) Specifications

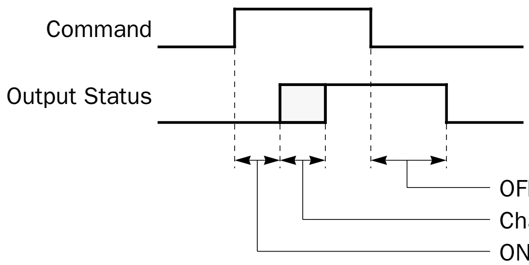
I/O Type		**10-I/O Type**	**16-I/O Type**	**24-I/O Type**
Output Protection		Without protection		
Protection Circuits Prepared by User		See page 1-23.		
Output Points		4 points	7 points	10 points
Output Points per Common Line	**COM0**	4 points	4 points	5 points
	COM1	—	3 points	5 points
Rated Load Voltage		24V DC		
Operating Load Voltage Range		19 to 30V DC		
Rated Load Current		0.5A per output point		
Maximum Load Current		0.625A per output point (at 30V DC)		
Voltage Drop (ON Voltage)		1.5V maximum (voltage between COM and output terminals when output is on)		
Inrush Current		5A maximum		
Leakage Current		0.1 mA maximum		
Clamping Voltage		39V±1V		
Maximum Clamping Load		10W		
Inductive Load		Continuous operation of $T_{0.95}$ = 60 msec (DC13) at 1 Hz, 24V DC		
External Current Draw		10 mA maximum, 24V DC (power supply to the +V terminal)		
Maximum Frequency Response **Test Condition:** **Load resistance 1 kΩ, 24V DC**	**Q0**	10 kHz minimum		
	Q1-Q31	1 kHz minimum (not including scan time)		
Isolation	**Between Output Terminal and FG**	1,500V AC		
	Between Output Terminal and Internal Circuit	Photocoupler isolated		
	Between Output Terminals of Different COM Lines	Not isolated		
Effect of Improper Connection		When a current larger than the rated current flows, permanent damage of output elements may be caused.		
Output Status by MPU Operation	**Stop**	OFF		
	Power Interruption over 25 msec	OFF		
	Power Interruption 25 msec or less	ON/OFF status maintained		
	Power Up	OFF until MPU starts to run		
Others (IEC 1131-2 Information)		IEC 1131-2 3.3.3.3 3) Not applicable IEC 1131-2 3.3.3.3 8) Not applicable IEC 1131-2 3.3.3.3 9) Not tested for other categories IEC 1131-2 3.3.3.3 10) Not multichannel module IEC 1131-2 3.3.3.3 16) Output can not be replaced because of non-modular structure IEC 1131-2 3.3.3.3 18) Non-latching type output operation IEC 1131-2 3.3.3.3 19) Not multi-circuit module		

Digital DC Output (Transistor Protect Source Output) Specifications

I/O Type			10-I/O Type	16-I/O Type	24-I/O Type
Output Protection			Protected output		
Output Points			4 points	7 points	10 points
Output Points per Common Line		COM0	4 points	4 points	5 points
		COM1	—	3 points	5 points
Rated Load Voltage			24V DC		
Operating Load Voltage Range			19 to 30V DC		
Rated Load Current			0.5A per output point		
Maximum Load Current			0.625A per output point (at 30V DC)		
Voltage Drop (ON Voltage)			1.5V maximum (voltage between COM and output terminals when output is on)		
Inrush Current			5A maximum		
Leakage Current			0.1 mA maximum		
Clamping Voltage			39V±1V		
Maximum Clamping Load			10W		
Inductive Load			Continuous operation of $T_{0.95}$ = 60 msec (DC13) at 0.5 Hz, 24V DC		
External Current Draw			100 mA maximum, 24V DC (power supply to the –V terminal)		
Protected Output	Protect Activation Current	Q0 & Q20	0.626 to 0.9A		
		Q1-Q11 Q21-Q31	0.7 to 1.5A		
	Restarting Method	Q0 & Q20	• Remove the cause of overload and turn outputs off for 5 sec using the program loader or CUBIQ on the computer. • Or, turn power off.		
		Q1-Q11 Q21-Q31	• Remove the cause of overload, then the output protection is reset automatically. Note: When using at a high temperature (45°C or above), it may take a long time before normal operation is restored. If this is the case, turn output power off.		
	Maximum Frequency Response Test Condition: Load resistance 1 kΩ, 24V	Q0	10 kHz minimum		
		Q1-Q31	1 kHz minimum (not including scan time)		
	PWM Setting		Protection function does not work.		
	Note		The protected output is not reset using FUN20 PLC error data readout and reset without removing the cause of the overload.		
Isolation	Between Output Terminal and FG		1,500V AC		
	Between Output Terminal and Internal Circuit		Photocoupler isolated		
	Between Output Terminals of Different COM Lines		Not isolated		
Effect of Improper Connection			When a current larger than the rated current flows, permanent damage of output elements may be caused.		
Output Status by MPU Operation	Stop		OFF		
	Power Interruption over 25 msec		OFF		
	Power Interruption 25 msec or less		ON/OFF status maintained		
	Power Up		OFF until MPU starts to run		
Others (IEC 1131-2 Information)			• See Isolation described above. • See Effect of Improper Connection described above.		

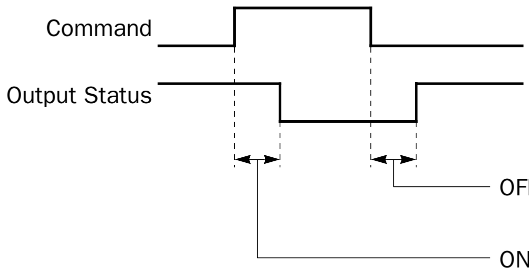
Output Delay

Digital AC/DC Output (Relay Output)



OFF delay: 10 msec maximum
Chatter: 6 msec maximum
ON delay: 6 msec maximum

Digital DC Output (Transistor Sink Output)

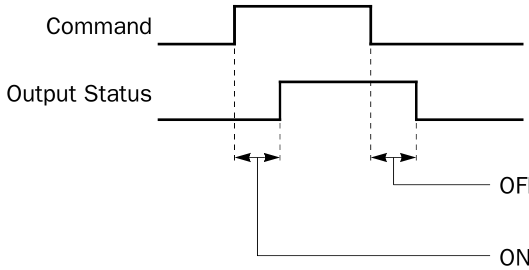


Test Condition: Load resistance 1 kΩ
24V DC

OFF delay Q0: 5 µsec maximum
Q1-Q31: 500 µsec maximum

ON delay Q0: 5 µsec maximum
Q1-Q31: 500 µsec maximum

Digital DC Output (Transistor Protect Source Output)



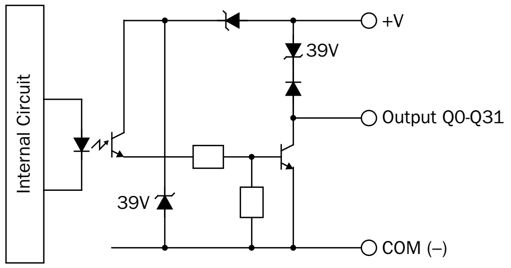
Test Condition: Load resistance 1 kΩ
24V DC

OFF delay Q0: 5 µsec maximum
Q1-Q31: 500 µsec maximum

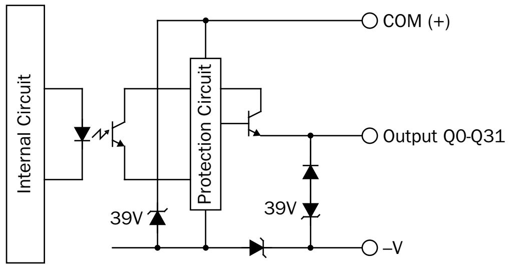
ON delay Q0: 5 µsec maximum
Q1-Q31: 500 µsec maximum

Output Internal Circuit

Digital DC Output (Transistor Sink Output)



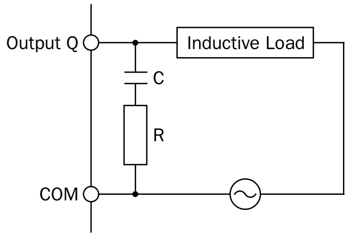
Digital DC Output (Transistor Protect Source Output)



Contact Protection Circuit for Relay Output

Depending on the load, a protection circuit may be needed for the relay output of the MICRO^3. Choose a protection circuit from A through D shown below according to the power supply and connect the protection circuit to the outside of the MICRO^3.

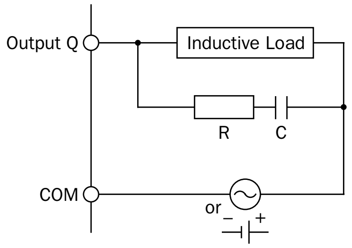
Protection Circuit A



This protection circuit can be used when the load impedance is smaller than the RC impedance in an AC load power circuit.

C: 0.1 to 1 µF
R: Resistor of approximately the same resistance value as the load

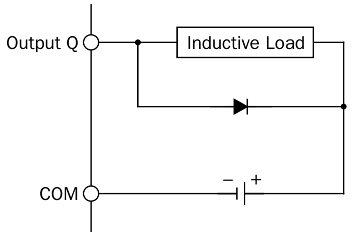
Protection Circuit B



This protection circuit can be used for both AC and DC load power circuits.

C: 0.1 to 1 µF
R: Resistor of approximately the same resistance value as the load

Protection Circuit C

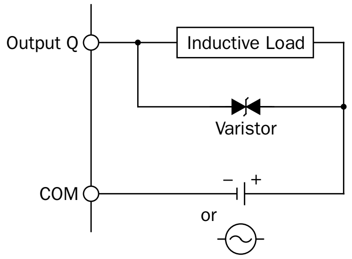


This protection circuit can be used for DC load power circuits.

Use a diode with the following ratings.

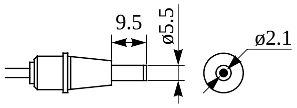
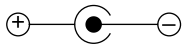
Reverse withstand voltage:	Power voltage of the load circuit × 10
Forward current:	More than the load current

Protection Circuit D



This protection circuit can be used for both AC and DC load power circuits.

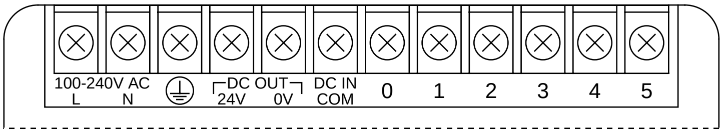
Program Loader Specifications

Power Supply	• Supplied by the MICRO^3 base unit through the loader cable. • Supplied by an AC adapter during off-line programming. Applicable AC adapter 5 to 6.5 V DC, 4W Output plug:  Dimensions in mm. **Polarity** 
Operating Temperature	0 to 50°C
Storage Temperature	–20 to +70°C
Relative Humidity	Relative humidity severity level RH1, 30 to 95% (non-condensing)
Pollution Degree	3 (IEC 664)
Vibration Resistance	5 to 55 Hz, 60 m/sec^2, 2 hours each in 3 axes
Shock Resistance	300 m/sec2, 11 msec, 3 shocks each in 3 axes
Power Consumption	Normal operation: Approx. 1.5W Writing to flash PROM: Approx. 2.5W
Mounting Method	The permanent magnet on the back of the program loader attaches to iron panels.
Dimensions	185H × 95W × 30D mm (7.283"H × 3.740"W × 1.181"D)
Weight	Approx. 300g
Noise Immunity	Withstands the noise same as the MICRO^3 noise immunity
Display	4 lines × 16 characters Back-lighted LCD with automatic turn off function
Program Key	35 keys, membrane switch key pad (key sheet replaceable)
Control Switch	RUN/STOP for MICRO^3 operation
Connection to MICRO^3	Using loader cable FC2A-KL1 (2m/6.56 ft.) or FC2A-KL2 (5m/16.4 ft.), round 8-pin DIN connector Baud rate: 9600 bps, using RS485 special protocol
Power Failure Protection	Approx. 1 hour at 25°C, using a super capacitor
Memory Card	Compliance with JEIDA Ver. 4.0/PCMCIA Rel. 1.0 Accessible capacity 256K bytes SRAM card (with battery) User program storage Read, write, and battery voltage drop detection Applicable cards: Fujitsu, Mitsubishi, Rohm, Fuji Electrochemical, Towa Electron PROM card Upgrade system program storage (128K bytes) Read only
User Program Edit Capacity	8K steps maximum

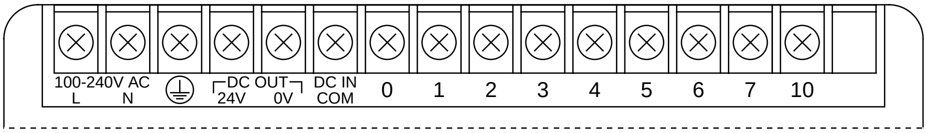
Input Terminal Arrangements

DC Input (AC Power Type)

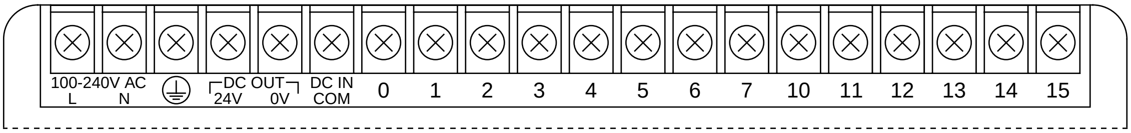
10-I/O Unit
(6 inputs)



16-I/O Unit
(9 inputs)

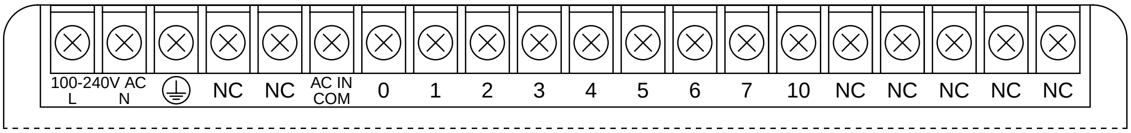


24-I/O Unit
(14 inputs)



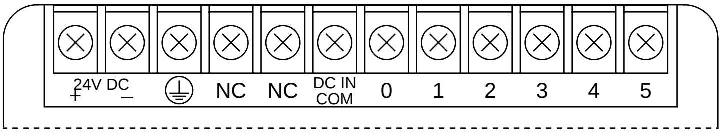
AC Input (AC Power Type)

16-I/O Unit
(9 inputs)

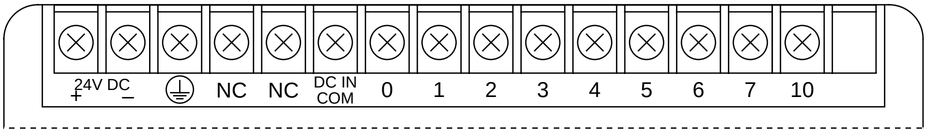


DC Input (DC Power Type)

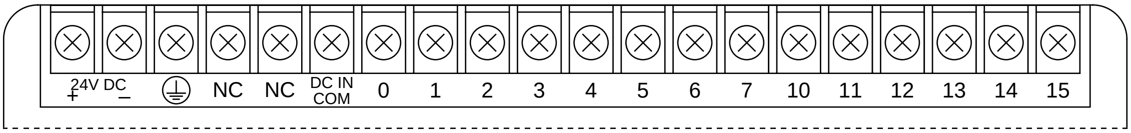
10-I/O Unit
(6 inputs)



16-I/O Unit
(9 inputs)



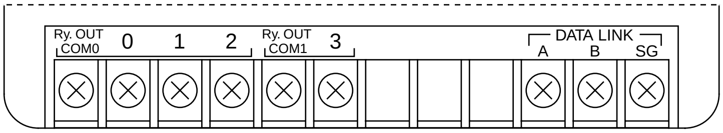
24-I/O Unit
(14 inputs)



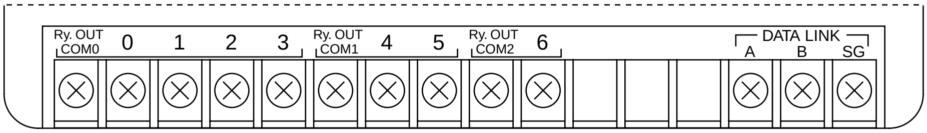
Output Terminal Arrangements

Relay Output (DC Input Type)

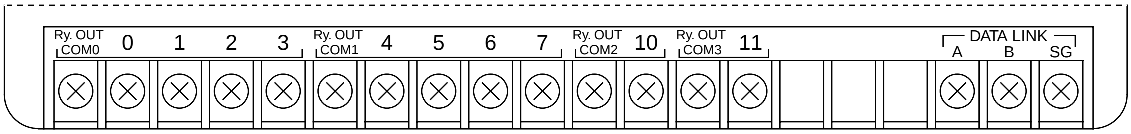
10-I/O Unit
(4 outputs)



16-I/O Unit
(7 outputs)

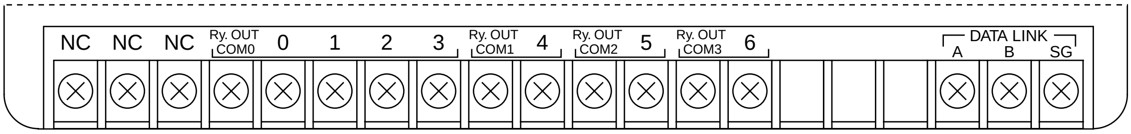


24-I/O Unit
(10 outputs)



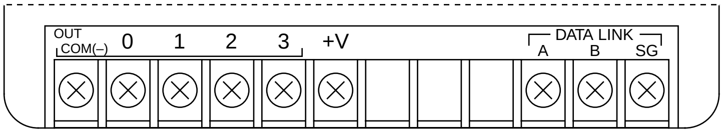
Relay Output (AC Input Type)

16-I/O Unit
(7 outputs)

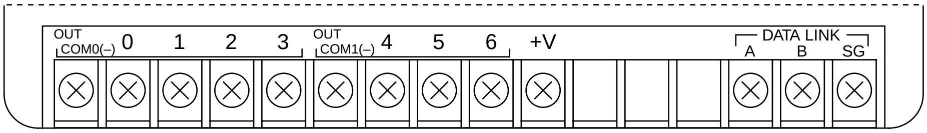


Transistor Sink Output

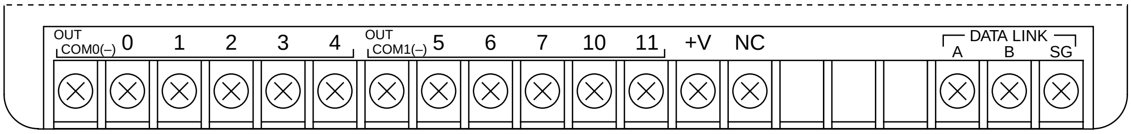
10-I/O Unit
(4 outputs)



16-I/O Unit
(7 outputs)

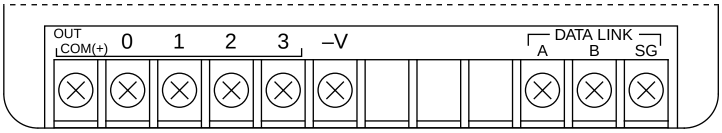


24-I/O Unit
(10 outputs)

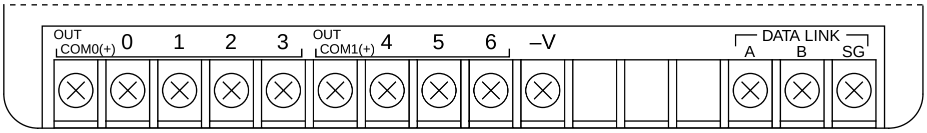


Transistor Protect Source Output

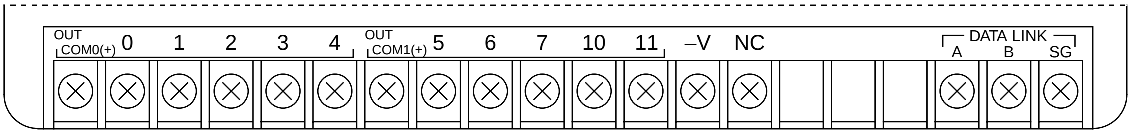
10-I/O Unit
(4 outputs)



16-I/O Unit
(7 outputs)



24-I/O Unit
(10 outputs)



Input Wiring Diagrams

⚠ Warning

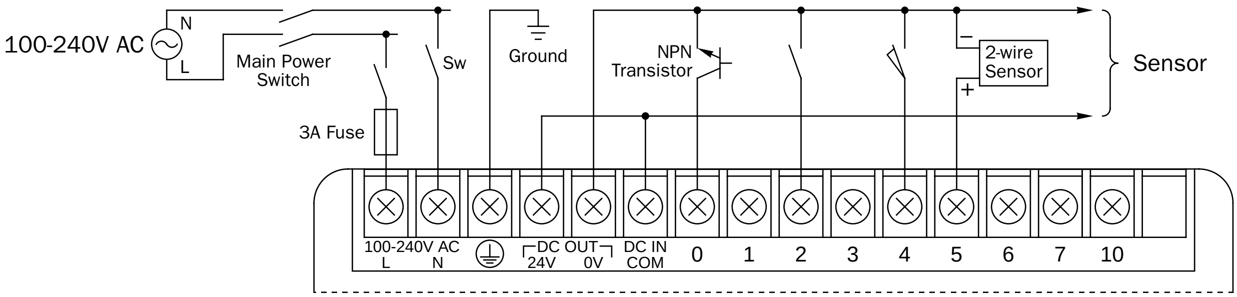
- Emergency and interlocking circuits must be configured outside the MICRO^3. If such a circuit is configured inside the MICRO^3, failure of the MICRO^3 may cause disorder of the control system, damage, or accidents.

⚠ Caution

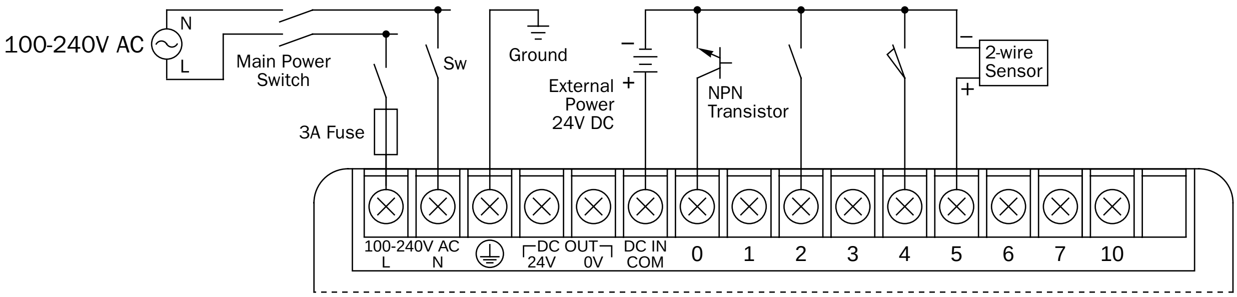
- Use a power supply of the rated value. Use of a wrong power supply may cause fire hazard.
- Use an IEC127-approved fuse on the power line outside the MICRO^3. This is required when exporting equipment containing MICRO^3 to Europe.
- Use an EU-approved circuit breaker. This is required when exporting equipment containing MICRO^3 to Europe.
- Do not connect to the ground directly from the MICRO^3. Connect a protective ground to the equipment containing MICRO^3 using an M4 or larger screw. This is required when exporting equipment containing MICRO^3 to Europe.
- If relays or transistors in the MICRO^3 output circuit fail, outputs may remain on or off. For output signals which may cause heavy accidents, provide a monitor circuit outside of the MICRO^3.
- Use an IEC127-approved fuse on the output circuit. This is required when exporting equipment containing MICRO^3 to Europe.

DC Source Input (AC Power Type)

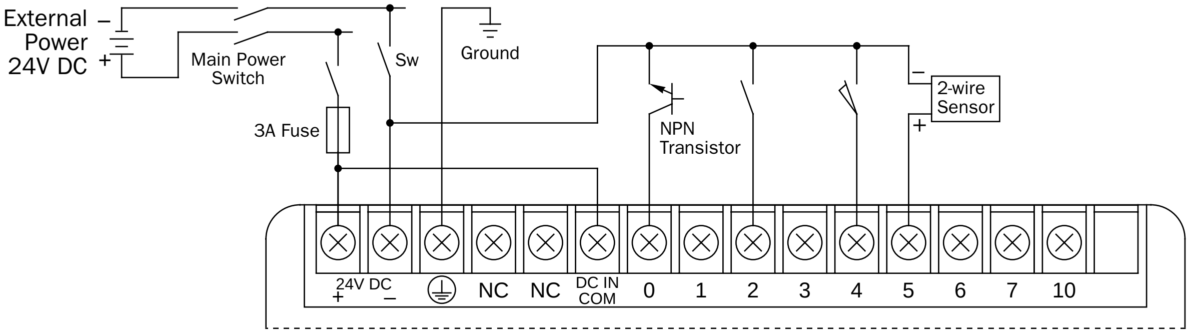
- **When using the sensor power supply from the DC OUT terminals**



- **When using an external power supply**



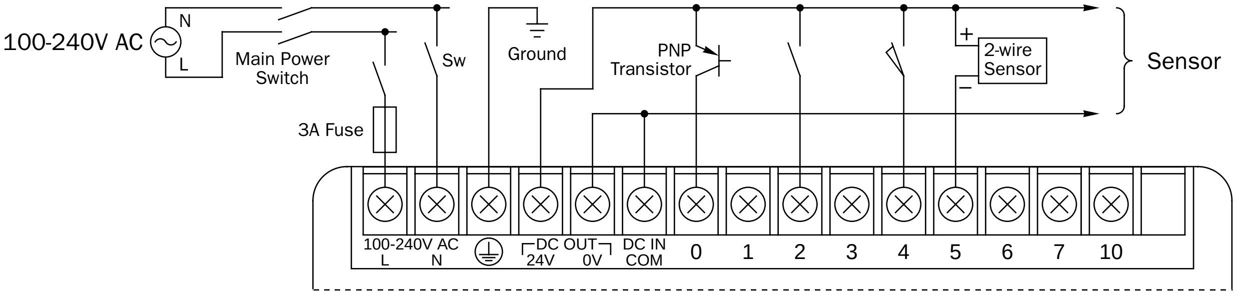
DC Source Input (DC Power Type)



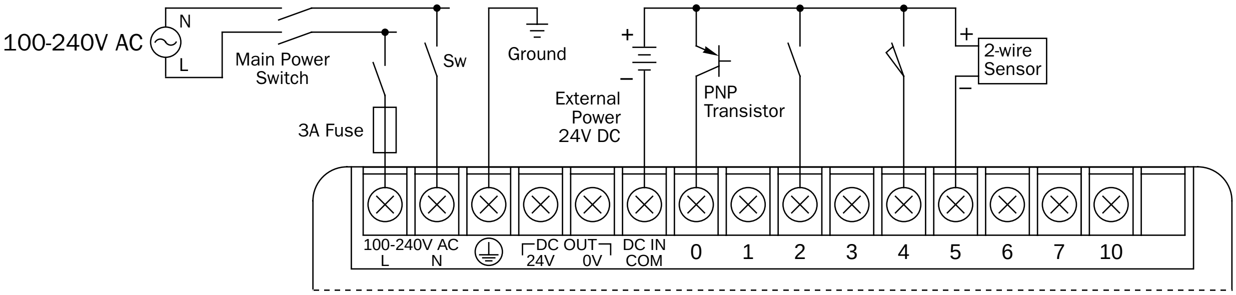
Input Wiring Diagrams, continued

DC Sink Input (AC Power Type)

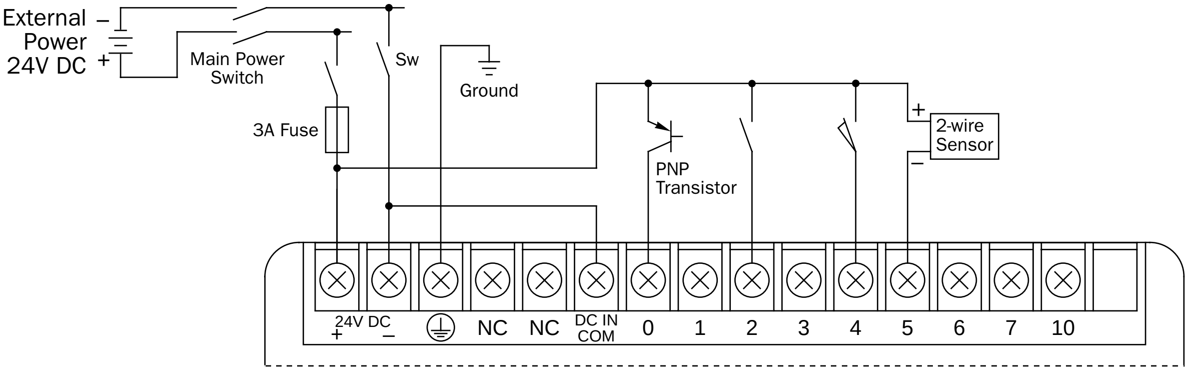
- **When using the sensor power supply from the DC OUT terminals**



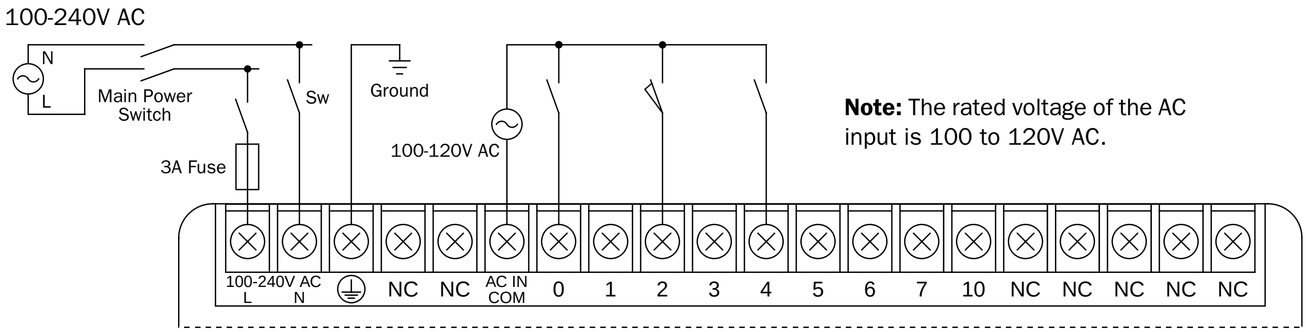
- **When using an external power supply**



DC Sink Input (DC Power Type)



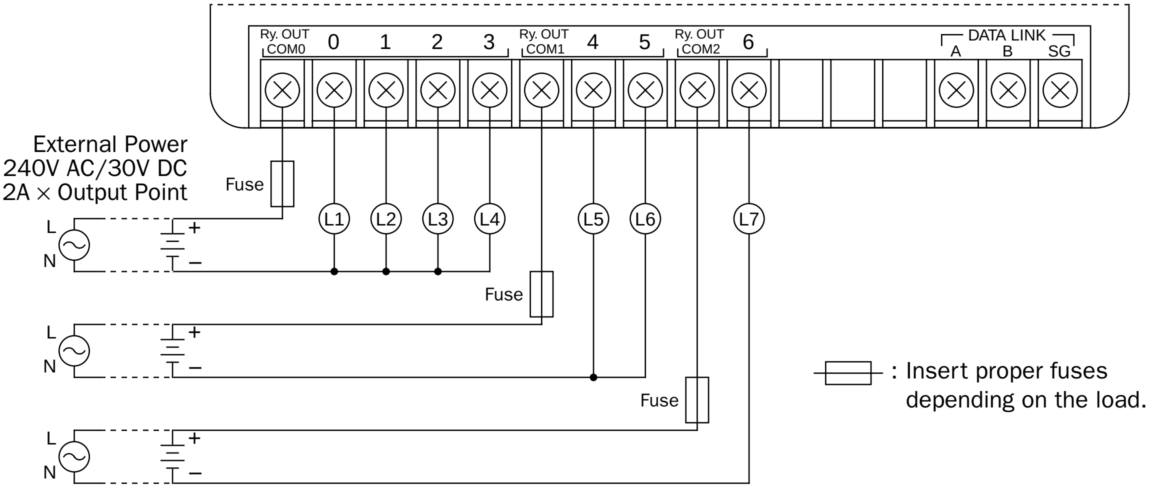
AC Input



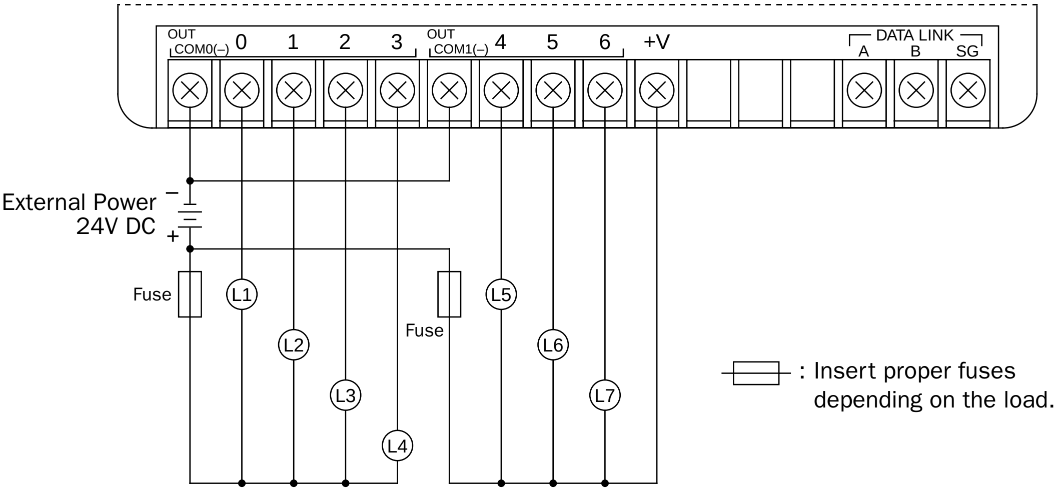
Note: The rated voltage of the AC input is 100 to 120V AC.

Output Wiring Diagrams

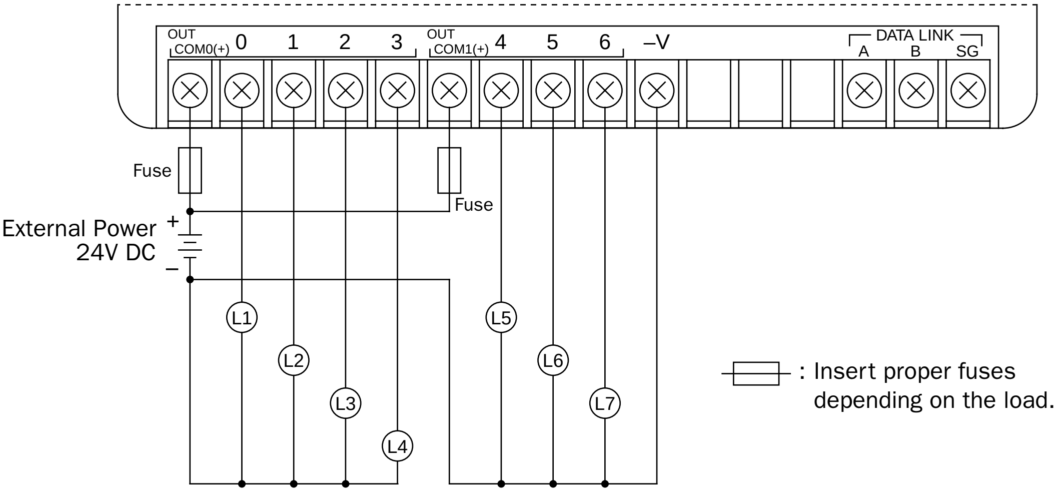
Relay Output



Transistor Sink Output

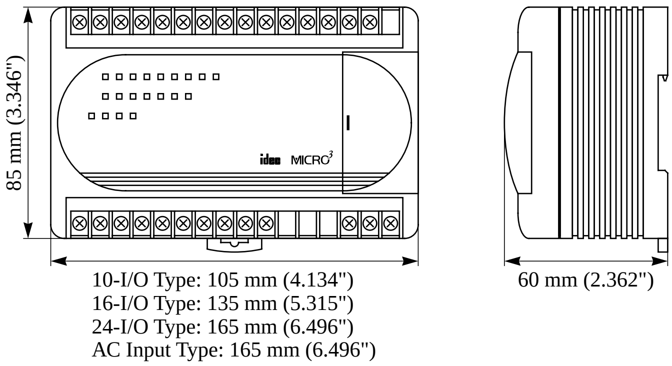


Transistor Protect Source Output

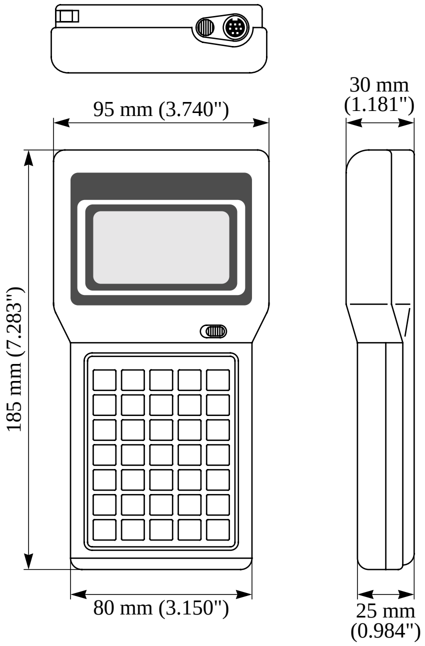


Dimensions

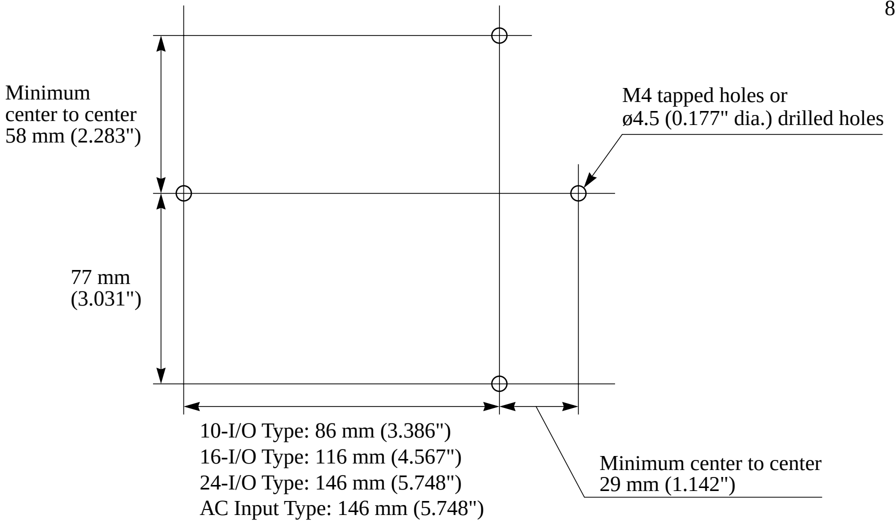
MICRO3 Base Unit


85 mm (3.346")
idec MICRO3
10-I/O Type: 105 mm (4.134")
16-I/O Type: 135 mm (5.315")
24-I/O Type: 165 mm (6.496")
AC Input Type: 165 mm (6.496")
60 mm (2.362")

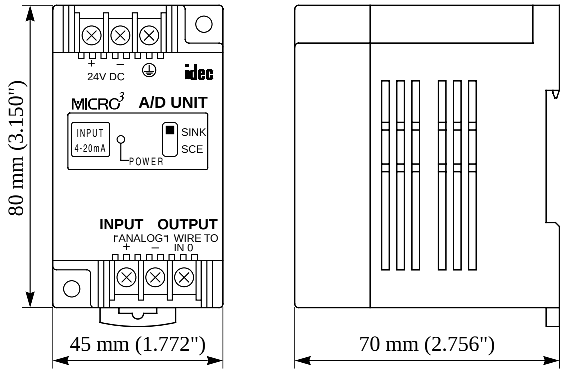
Program Loader



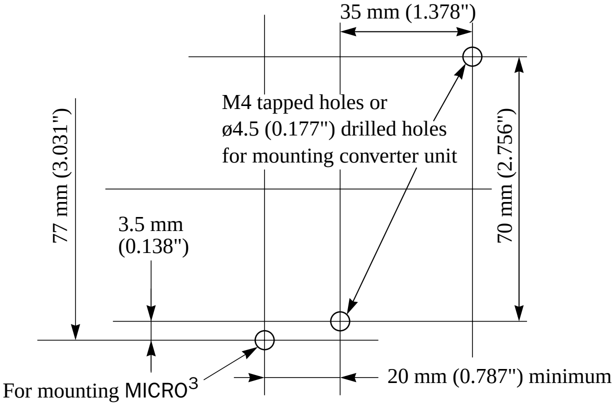
Mounting Hole Layout for MICRO3 Base Units


Minimum center to center 58 mm (2.283")
M4 tapped holes or ø4.5 (0.177" dia.) drilled holes
77 mm (3.031")
10-I/O Type: 86 mm (3.386")
16-I/O Type: 116 mm (4.567")
24-I/O Type: 146 mm (5.748")
AC Input Type: 146 mm (5.748")
Minimum center to center 29 mm (1.142")

A/D and D/A Converter Units



Mounting Hole Layout For A/D and D/A Converter Units



Installation

This section describes the methods and precautions for installing the MICRO^3.

⚠ **Warning**

- Turn power off to the MICRO^3 before starting installation, removal, wiring, maintenance, and inspection on the MICRO^3. Failure to turn power off may cause electrical shocks or fire hazard.
- Emergency and interlocking circuits must be configured outside the MICRO^3. If such a circuit is configured inside the MICRO^3, failure of the MICRO^3 may cause disorder of the control system, damage, or accidents.
- Special expertise is required to install, wire, program, and operate the MICRO^3. People without such expertise must not use the MICRO^3.

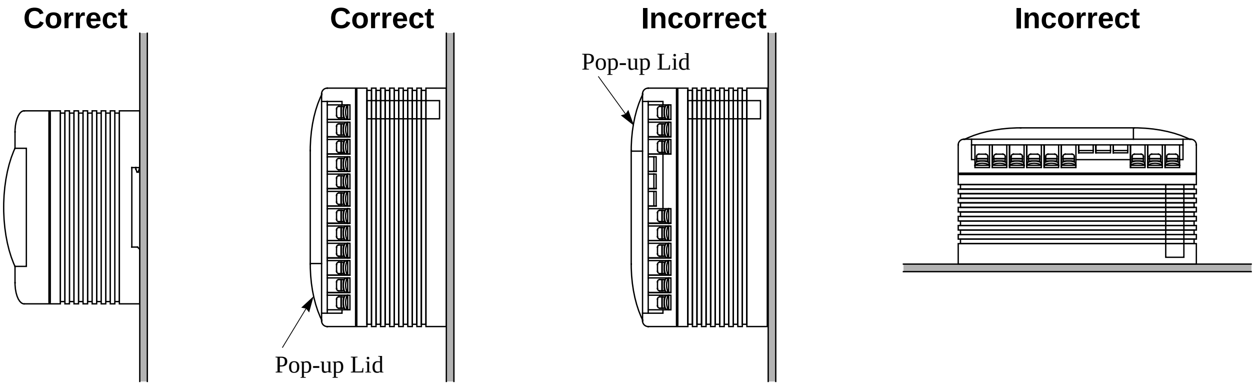
⚠ **Caution**

- Prevent metal fragments and pieces of wire from dropping inside the MICRO^3 housing. Put a cover on the MICRO^3 during installation and wiring. Ingress of such fragments and chips may cause fire hazard, damage, or malfunction.
- MICRO^3 is designed for installation in equipment. Do not install the MICRO^3 outside of equipment.
- The pollution degree of the MICRO^3 is “Pollution degree 2.” Use the MICRO^3 in environments of pollution degree 2 (according to IEC664-1).

Installation Location

The MICRO^3 programmable controller should be installed correctly for optimum performance.

Mount the MICRO^3 base unit on a vertical plane; not on a horizontal plane. When mounting the MICRO^3 base unit vertically, place the pop-up lid down to prevent heat build-up.



Make sure that the operating temperature does not drop below 0°C or exceed 60°C. If the temperature does exceed 60°C, use a fan or cooler.

To eliminate excessive temperature build-up, provide ample ventilation. Do not install MICRO^3 near, and especially above, any device which generates considerable heat, such as a heater, transformer, or large capacity resistor. The relative humidity should be above 45% and below 85%.

MICRO^3 should not be exposed to excessive dust, dirt, salt, direct sunlight, vibrations, or shocks. Do not use MICRO^3 in an area where corrosive chemicals or flammable gases are present. The unit should not be exposed to chemical, oil, or water splashes.

Installation Methods

⚠ **Caution**

- Install the MICRO^3 according to instructions described in this user’s manual and the MICRO^3 user’s manual. Improper installation will result in falling, failure, or malfunction of the MICRO^3.

MICRO^3 can be installed in two ways; direct mounting on a panel surface and mounting on a DIN rail.

Direct Mounting

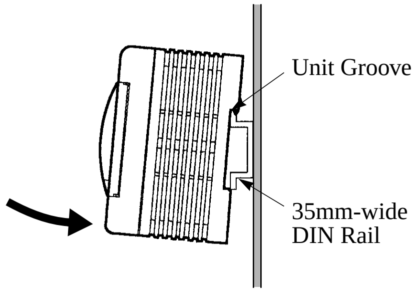
The MICRO^3 base unit can be mounted on a panel surface.

Drill mounting holes as shown on page 1-24. Use M4 screws (6 or 8 mm long) to mount the MICRO^3 base unit. Spring washers can be used with the screws.

DIN Rail Mounting

The MICRO^3 unit can be mounted on a 35-mm-wide DIN rail.

Applicable DIN rail: IDEC's BAA1000 (1000mm/39.4" long)



• Mounting on DIN Rail

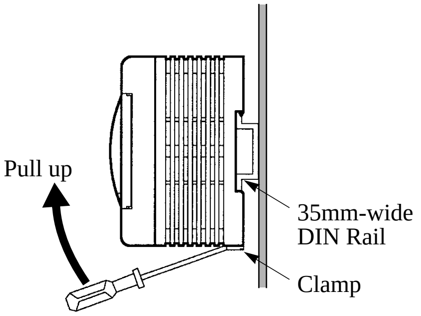
Fasten the DIN rail to a panel using screws firmly.

Put the groove of the MICRO^3 base unit on the DIN rail, with the input terminal side up, and press the unit to the panel as shown on the right.

Use BNL6 mounting clips on both sides of the MICRO^3 base unit to prevent moving sideways.

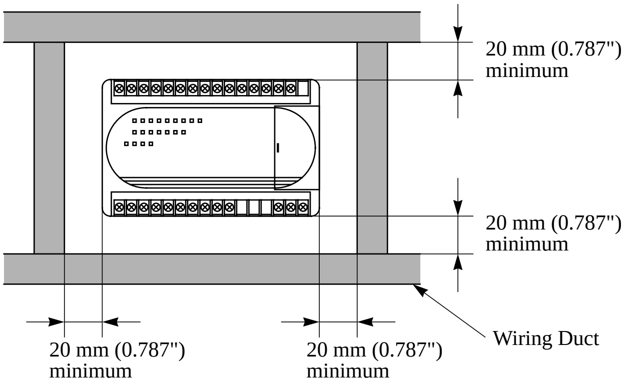
• Removing from DIN Rail

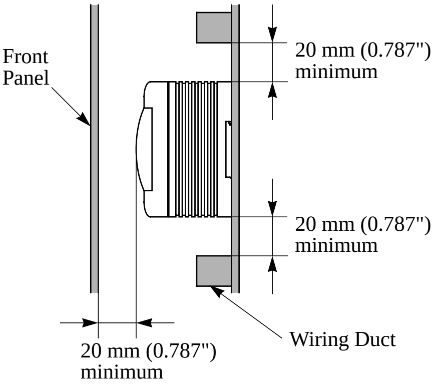
Insert a flat screwdriver into the slot in the clamp, pull the screwdriver up, and turn the MICRO^3 base unit bottom out.



Installation in Control Panel

When wiring input and output lines in ducts, keep a minimum space of 20 mm above and below the MICRO^3 base unit for maintenance. To prevent excessive heat built-up, keep a minimum space of 20 mm around the MICRO^3 unit for ventilation.





Disposing of the MICRO^3 Units

⚠ Caution

- When disposing of the MICRO^3 units, do so as an industrial waste.
- Dispose of the battery in the MICRO^3 when the battery is dead in accordance with pertaining regulations. When storing or disposing of the battery, use a proper container prepared for this purpose. This is required when exporting equipment containing MICRO^3 to Europe.
- Dispose of the battery in the memory card when the battery is dead in accordance with pertaining regulations.

Wiring

⚠ Caution

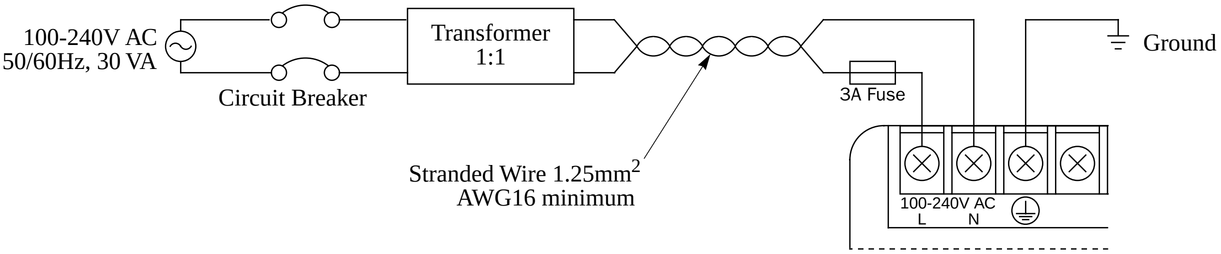
- Use wires of a proper size to meet voltage and current requirements. Tighten M3 screws for power and I/O terminals to a proper tightening torque of 0.3 to 0.5 N-m.
- Do not disassemble, repair, or modify the MICRO^3.

Power Supply Wiring

Use a stranded wire of 1.25 mm^2 cross section (AWG16) for power supply wiring. Make the power supply wiring as short as possible and run the wiring as far away as possible from motor lines.

To prevent electrical shocks or malfunctioning due to noise, connect the FG terminal to the ground using a grounding wire of 2 mm^2 cross section (AWG14) minimum (grounding resistance 100Ω maximum). Do not connect the grounding wire in common with the grounding wire of motor equipment.

When using MICRO^3 on AC power, noise can be greatly reduced by connecting a 1:1 transformer as shown below:



⚠ Caution

- Do not connect to the ground directly from the MICRO^3. Connect a protective ground to the equipment containing MICRO^3 using an M4 or larger screw. This is required when exporting equipment containing MICRO^3 to Europe.
- Use an EU-approved circuit breaker. This is required when exporting equipment containing MICRO^3 to Europe.

Input Wiring

Use wire between 0.75 and 1.25 mm^2 cross section (AWG18 and AWG16) for input wiring. Separate the input wiring from the output line, power line, and motor line. For input wiring diagrams, see pages 1-21 and 1-22.

Output Wiring

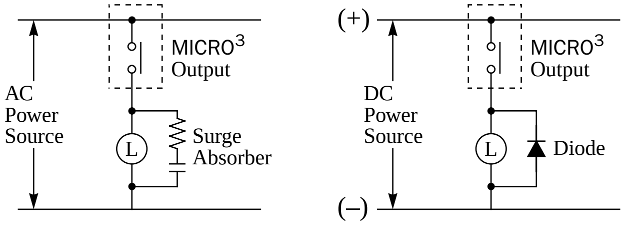
⚠ Caution

- If relays or transistors in the MICRO^3 output circuit fail, outputs may remain on or off. For output signals which may cause heavy accidents, provide a monitor circuit outside of the MICRO^3.
- Use an IEC127-approved fuse on the output circuit. This is required when exporting equipment containing MICRO^3 to Europe.

Use wire between 0.75 and 1.25 mm^2 cross section (AWG18 and AWG16) for output wiring.

When driving loads which generate noise, such as electromagnetic contactors and solenoid valves, use a surge absorber for AC power or a diode for DC power.

For output wiring diagrams, see page 1-23.



Data Link Wiring

For wiring the data link terminals in the expansion link or data link system, use a two-core twisted pair shielded cable with a minimum core wire diameter of 0.9 mm. Separate the data link wiring from the output line, power line, and motor line.

Power Supply Voltage

⚠ **Caution**

- Use a power supply of the rated value. Use of a wrong power supply may cause fire hazard.
- Use an IEC127-approved fuse on the power line outside the MICRO^3. This is required when exporting equipment containing MICRO^3 to Europe.

The applicable power range for MICRO^3 is 85 to 264V AC or 19 to 30V DC. When MICRO^3 is powered up, the inrush current flow is 40A maximum at 264V AC or 30V DC with the rated input and output.

Power failure voltage varies with the operating conditions of the program loader and the number of I/O points used. In most cases, power failure is detected when the power voltage drops below 85V AC or 19V DC. Operation is stopped at this point to prevent malfunctioning. Momentary power failures of 25 msec or less are not detected at the rated power voltage.

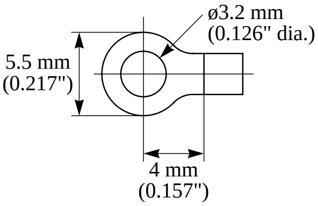
Other Precautions

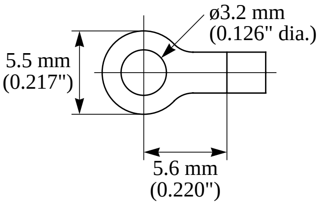
⚠ **Caution**

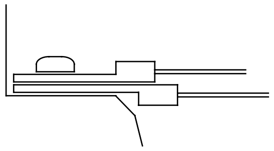
- Do not use the MICRO^3 in environments outside of the specification values.
- Connect the FG terminal to a proper ground; otherwise, electrical shocks may be caused.
- Do not touch all screw terminals while the MICRO^3 is powered up; otherwise, electrical shocks may be caused.
- Do not touch the input terminals immediately after inputs are turned off; otherwise, electrical shocks may be caused.

Crimping Terminal

When connecting one wire to one terminal, use a crimping terminal shown on the left below.
Only when connecting two wires to one terminal, use the longer crimping terminal shown in the middle below.







Introduction

This chapter describes general information for starting and stopping MICRO3 operation, and introduces simple operating procedures from creating a user program to monitoring the MICRO3 operation.

⚠ **Caution**

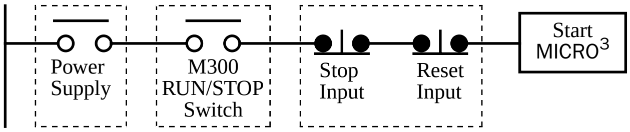
- Make sure of safety before starting and stopping the MICRO3 or when operating the MICRO3 to force outputs on or off. Incorrect operation on the MICRO3 may cause machine damage or accidents.

Start/Stop Operation

This section describes operations to start and stop MICRO3 and to use the stop and reset inputs.

Start/Stop Schematic

The start/stop circuit of MICRO3 consists of three blocks; power supply, M300 (start control special internal relay), and stop/reset inputs. Each block can be used to start and stop MICRO3 while the other two blocks are set to run MICRO3.



Start/Stop Operation using Program Loader

MICRO3 can be started and stopped using the program loader connected to the MICRO3 base unit. When the RUN/STOP switch on the program loader is set to RUN, start control special internal relay M300 is turned on to start MICRO3. When the RUN/STOP switch is set to STOP, M300 is turned off to stop MICRO3.

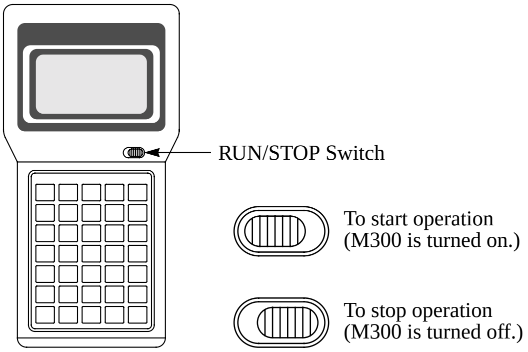
Connect the program loader to MICRO3 and power up MICRO3. See page 1-3.

Check that a stop input is not designated using FUN1. See 5-3.

To start or stop MICRO3 operation, set the RUN/STOP switch to RUN or STOP.

Note: When a stop input is designated using FUN1, MICRO3 cannot be started or stopped when start control special internal relay M300 is turned on or off.

The response time of the RUN/STOP switch operation is shown below.


RUN/STOP Switch
To start operation (M300 is turned on.)
To stop operation (M300 is turned off.)

When setting the RUN/STOP switch to STOP, MICRO3 stops operation and the program loader displays “PC-STOP” immediately. After approximately 1 second, the program loader restores the previous display.

When setting the RUN/STOP switch to RUN, MICRO3 starts operation and the program loader displays “PC-RUN.” After approximately 1 second, the program loader restores the previous display.

The status of start control special internal relay M300 can be monitored using the program loader. MICRO3 can also be started and stopped by turning M300 on and off using the program loader.

Monitor and set M300 to start MICRO3.

MON | C/SOT M | 3 BPP | 0 | 0 | SET I | ↵

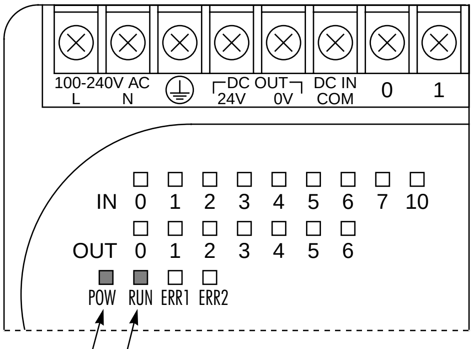
Monitor and reset M300 to stop MICRO3.

MON | C/SOT M | 3 BPP | 0 | 0 | F/RST Q | ↵

Note: Special internal relay M300 is a keep type internal relay and stores the status when power is turned off. M300 retains its previous status when power is turned on again. However, when the backup battery is dead, M300 loses the stored status and is turned on when MICRO3 is powered up. The backup time after lithium battery fully charged is:

Without clock/calendar (10-I/O type):	Approx. 50 days at 25°C (typical)
With clock/calendar (16/24-I/O types):	Approx. 30 days at 25°C (typical)

Start/Stop Operation using the Power Supply



Remains on during operation.

Remains on while power is on.

MICRO^3 can be started and stopped by turning power on and off.

Check that start control special internal relay M300 is on using the program loader. If M300 is off, turn it on. See above.

Turn power on to start operation. Turn power off to stop operation.

Note: If M300 is off, MICRO^3 does not to start operation when power is turned on even if the RUN/STOP switch on the program loader is set to RUN. To start operation, turn power on, and set the RUN/STOP switch to STOP, then back to RUN. If M300 is on, then MICRO^3 starts operation when power is turned on regardless of the RUN/STOP switch setting position.

Stop Input (FUN1) and Reset Input (FUN2)

Input I0 through I15 can be designated as a stop input using FUN1. Input I0 through I15 can also be configured as a reset input using FUN2. These functions are explained in detail on pages 5-3 and 5-4.

Note: When using a stop and/or reset input to start and stop MICRO^3 operation, make sure that start control special internal relay M300 is on. If M300 is off, then MICRO^3 does not start operation when the stop or reset input is turned off. M300 is not turned on or off when the stop and/or reset input is turned on or off.

When a stop or reset input is turned on during program operation, the RUN indicator is turned off, MICRO^3 stops operation, and all outputs are turned off.

The reset input has priority over the stop input.

System Statuses

The system statuses during running, stop, reset, restart after resetting, and restart after stopping are listed below:

Mode	Outputs	Internal Relays, Shift Registers		Timer Current Value	Counter Current Value	Data Register
		Keep Type	Clear Type			
Run	Operating	Operating	Operating	Operating	Operating	Operating
Reset	OFF	OFF	OFF	Reset to zero	Reset to zero	Reset to zero
Stop	OFF	Unchanged	Unchanged	Unchanged	Unchanged	Unchanged
Reset → Restart	OFF	OFF	OFF	Reset to preset	Reset to zero	Reset to zero
Stop → Restart	OFF	Unchanged	OFF	Reset to preset	Unchanged	Unchanged

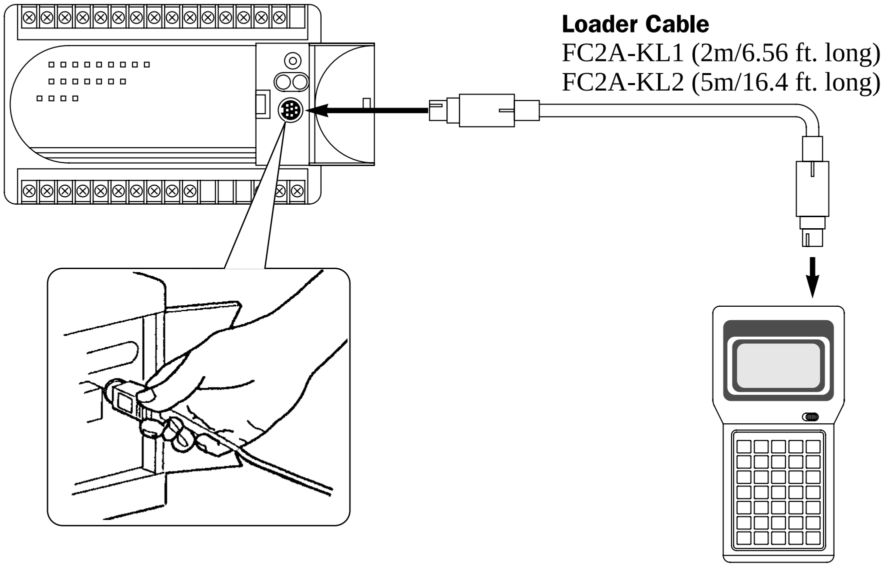
Simple Operation

This section describes how to edit a simple program using the program loader connected to MICRO^3, transfer the program to MICRO^3, run the program on MICRO^3, and monitor the operation on the program loader.

Connect Program Loader to MICRO^3

Connect the program loader to MICRO^3 using the loader cable. The program loader is powered by MICRO^3.

◆ ***For the MICRO^{3}C, see the MICRO^{3}C User's Manual.*** ◆



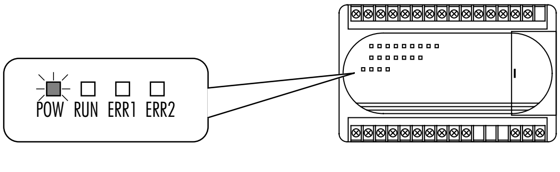
Plug the connector of the loader cable into the loader port on the MICRO^3 base unit until the connector clicks. Plug the connector on the other end of the loader cable into the loader cable connection port on the program loader.

⚠ **Caution**

- The connector has an orientation. Make sure of the correct orientation when plugging. To disconnect the cable, squeeze the connector, and pull it out.

Connect power supply and input switches to the MICRO^3 base unit. See pages 1-21 and 1-22.

Turn power on. The POW (power) indicator on the base unit goes on.



Program Loader Display

Check that the program loader displays the messages as shown on the right when the program loader is powered up.

```
*** Power on ***

Prg.Size  1Kstep
System Ver  1.02
```

Create a User Program

Create a simple program using the program loader. The sample program performs the following operation:

When input I0 is turned on, output Q0 is maintained.
When input I1 is turned on, output Q1 is maintained.
When input I2 is turned on, both outputs Q0 and Q1 are reset.

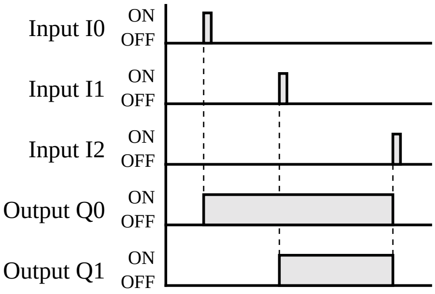
Set the RUN/STOP switch on the program loader to STOP. This will stop the MICRO^3 operation.

Delete the entire program from the program loader.

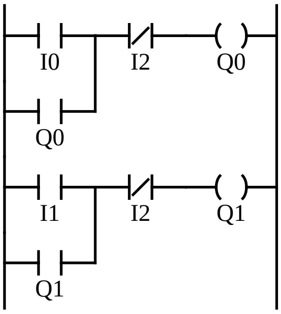
DEL END ↵

Make a time chart, relay diagram, and program list to perform the intended operation.

Timing Chart



Ladder Diagram



Program List

Prgm Adrs	Instruction	Data
0	LOD	I0
1	OR	Q0
2	AND NOT	I2
3	OUT	Q0
4	LOD	I1
5	OR	Q1
6	AND NOT	I2
7	OUT	Q1
8	END	

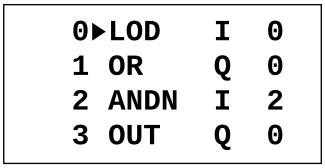
Enter the program by pressing the keys on the program loader. If you make a mistake in the key sequence, press the CLR key to begin the current line of programming (current address) again.

Address 0	LOD 10	SET I	0	↵	
Address 1	E OR D	F RST Q	0	↵	
Address 2	D AND	A NOT	SET I	2 BRD	↵
Address 3	OUT 16	F RST Q	0	↵	
Address 4	LOD 10	SET I	1 BPS	↵	
Address 5	E OR D	F RST Q	1 BPS	↵	
Address 6	D AND	A NOT	SET I	2 BRD	↵
Address 7	OUT 16	F RST Q	1 BPS	↵	

Check the program from address 0 to the end of the program using the program loader.

Press the CLR key three times to read address 0.

Press the ▼ key to verify the program up to the last address.



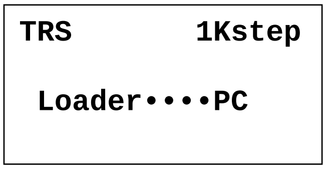
```
0▶LOD   I  0
1 OR    Q  0
2 ANDN  I  2
3 OUT   Q  0
```

Transfer Program and Monitor MICRO^3 Operation

Transfer the program to the MICRO^3 base unit, run the program and monitor the operation using the program loader using the following procedures.

Press the TRS key on the program loader to select the transfer mode. The program loader displays as shown on the right:

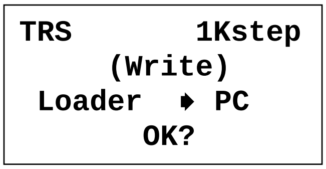
TRS



```
TRS        1Kstep

 Loader••••PC
```

Press the ↵ key. The display changes as shown on the right.

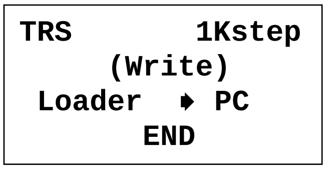
↵



```
TRS        1Kstep
     (Write)
 Loader   ➧ PC
       OK?
```

Press the ↵ key again to start program transfer. When program transfer is completed, the display changes as shown on the right.

↵



```
TRS        1Kstep
     (Write)
 Loader   ➧ PC
       END
```

To run the program on the MICRO^3 base unit, set the RUN/STOP switch on the program loader to RUN. See that the RUN indicator on the MICRO^3 base unit is turned on.

Monitor the input and output operation referring to the time chart on the preceding page.

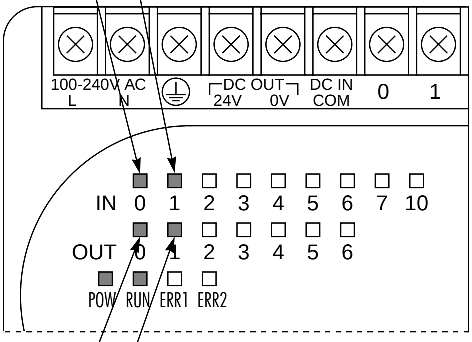
When input I0 is turned on (IN0 indicator on), output Q0 is turned on (OUT0 indicator on).
When input I0 is turned off (IN0 indicator off), output Q0 remains on.

When input I1 is turned on (IN1 indicator on), output Q1 is turned on (OUT1 indicator on).
When input I1 is turned off (IN1 indicator off), output Q1 remains on.

When input I2 is turned on (IN2 indicator on), both outputs Q0 and Q1 are turned off (OUT0 and OUT1 indicators off).

Remains on while input I0 is on.

Remains on while input I1 is on.



Remains on while output Q1 is on.

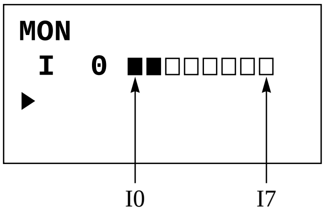
Remains on while output Q0 is on.

The I/O operation can also be monitored using the program loader.

Press the MON key on the program loader to select the monitor mode.

Then, enter the operand and number to monitor. To monitor input I0, press keys:

MON SET/I 0 ▼



```
MON
 I  0 ■■□□□□□□
 ▶
```

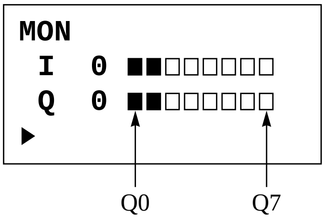
■ ON

□ OFF

Eight points are monitored starting with the selected number. The program loader displays changes as shown on the right.

To monitor 8 output points starting with output Q0 on the program loader, press keys:

MON F/RST/Q 0 ▼

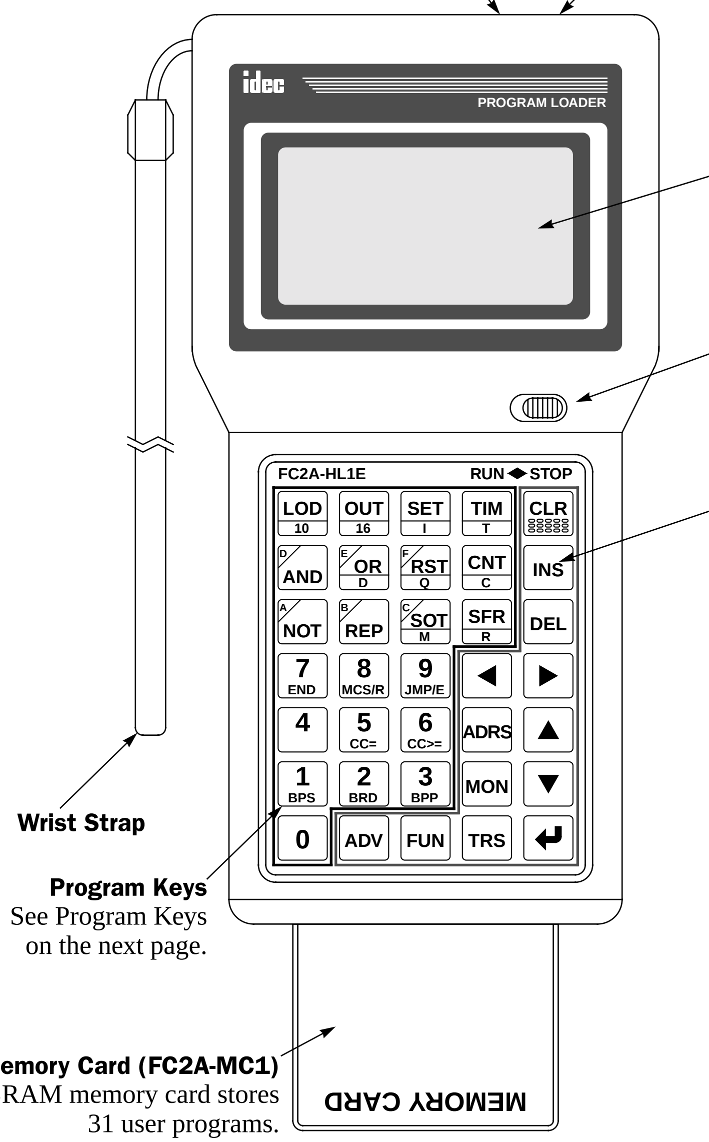


```
MON
 I  0 ■■□□□□□□
 Q  0 ■■□□□□□□
 ▶
```

Introduction

This chapter describes general information for understanding the functions and specifications of the FC2A-HL1E program loader. The program loader is used to edit user programs, transfer a user program to the MICRO^3 base unit, and monitor the MICRO^3 operation. Operating procedures for the editor, transfer, and monitor modes are described later in this chapter.

Parts Description



AC Adapter Jack
Connects an AC adapter to supply power to the program loader when using off line.

Loader Cable Connection Port
Connects the loader cable FC2A-KL1 (2m/6.56 ft. long) or FC2A-KL2 (5m/16.4 ft. long) to the MICRO^3 base unit.

◆ ***For the MICRO^{3}C, see the MICRO^{3}C User's Manual.*** ◆

Display
The back-lighted LCD shows programs and monitored data in 4 rows of 16 characters.

RUN/STOP Switch
Starts (RUN) or stops (STOP) the MICRO^3 operation.

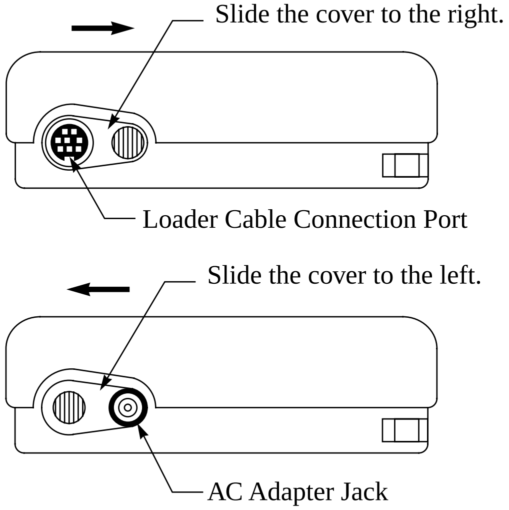
Function Keys
See Function Keys on the next page.

Wrist Strap

Program Keys
See Program Keys on the next page.

Memory Card (FC2A-MC1)
The SRAM memory card stores 31 user programs.
Memory capacity 64K bytes.

Top View of the Program Loader



Slide the cover to the right.

Loader Cable Connection Port

Slide the cover to the left.

AC Adapter Jack

Magnets on the Back
Magnet sheets are provided on the back of the program loader to attach to steel panels.

⚠ **Caution**

- Keep diskettes and magnetic cards away from the magnets; otherwise, data may be lost.

⚠ **Caution**

- Prevent the program loader from falling while inserting a memory card, connecting the loader cable, or plugging an AC adapter; otherwise damage or malfunction of the program loader, memory card, or MICRO^3 connector will result.
- Dispose of the battery in the memory card when the battery is dead in accordance with pertaining regulations.

Function Keys

CLR	Clear key used to return to the previous operation level or back to the editor mode.
INS	Insert key used to insert a program instruction.
DEL	Delete key used to delete program instructions.
ADRS	Address key used to select a program address.
MON	Monitor key used to monitor the MICRO^3 operation, to change timer/counter preset value, or to enter data into data register.
ADV	Advance instruction key used to program advanced instructions, to monitor high-speed counter, or to monitor double-word data.
FUN	Function key used to change FUN table settings.
TRS	Transfer key used to transfer and compare user programs between the program loader and the MICRO^3 base unit or memory card.
↵	Enter key used to write a program or FUN settings.
◀ ▶ ▲ ▼	Cursor move keys used to move the cursor or read a program

Program Keys

LOD 10	Programs the LOD instruction. Precedes entering a decimal value.
OUT 16	Programs the OUT instruction. Precedes entering a hexadecimal value.
SET I	Programs or executes the SET instruction. Specifies the input operand (I).
TIM T	Programs the TIM, TMH, or TMS instruction. Specifies the timer operand (T).
D/AND	Enters hexadecimal value D. Programs the AND instruction.
E/OR D	Enters hexadecimal value E. Programs the OR instruction. Specifies the data register operand (D).
F/RST Q	Enters hexadecimal value F. Programs or executes the RST instruction. Specifies the output operand (Q).
CNT C	Programs the counter instruction. Specifies the counter operand (C).
A/NOT	Enters hexadecimal value A. Programs the NOT instruction.
B/REP	Enters hexadecimal value B. Programs the repeat or selects an option.
C/SOT M	Enters hexadecimal value C. Programs the SOTU or SOTD instruction. Specifies the internal relay operand (M).
SFR R	Programs the shift register instruction. Specifies the shift register operand (R).
7 END	Enters decimal or hexadecimal value 7. Programs the END instruction.
8 MCS/R	Enters decimal or hexadecimal value 8. Programs the MCS or MCR instruction.
9 JMP/E	Enters decimal or hexadecimal value 9. Programs the JMP or JEND instruction.
4	Enters decimal or hexadecimal value 4.
5 CC=	Enters decimal or hexadecimal value 5. Programs the CC= instruction.
6 CC>=	Enters decimal or hexadecimal value 6. Programs the CC≥ instruction.
1 BPS	Enters decimal or hexadecimal value 1. Programs the BPS instruction.
2 BRD	Enters decimal or hexadecimal value 2. Programs the BRD instruction.
3 BPP	Enters decimal or hexadecimal value 3. Programs the BPP instruction.
0	Enters decimal or hexadecimal value 0.

Program keys have one or more legends or numbers on the key top. These keys select the operation automatically depending on the preceding key. For example, when the following keys are pressed in sequence:

LOD/10 SET/I 1/BPS ↵

The first key selects "LOD" to start key sequence.
The second key selects "Input" because "SET" does not follow "LOD."
The third key selects "1" because "BPS" does not follow "I."

Program Loader Operation Modes

The program loader has four operation modes and displays as shown below.

Editor Mode

```
0▶LOD   I  0
1 LOD   I  1
2 CNT 2   10
4 C 2=     5
```

The editor mode is used to edit a user program in the user program memory of the program loader. See page 3-5. From the normal editor mode, the operation mode can be changed to the address selection mode, insert mode, or delete mode.

Address selection mode: A program address is selected by pressing the ADRS key.

Insert mode: A program instruction is inserted by pressing the INS key.

Delete mode: Program instructions are deleted by pressing the DEL key.

Transfer Mode

```
TRS       1Kstep

 Loader••••PC
```

The transfer mode is used to transfer or compare user programs between the program loader and the MICRO^3 base unit or memory card. See page 3-9.

Monitor Mode

```
MON
▶X  0 □■□■□■□■
 Y  0 □■□■□■□■
 C  2 ■  10   10
```

The monitor mode is used to monitor input, output, internal relay, shift register statuses, preset and current values of timers and counters, and data of data registers on the program loader display. The monitor mode is also used to set or reset an input, output, internal relay, or shift register bit from the program loader. Timer/counter preset values and data register values are also changed using the monitor mode. See page 3-12.

FUN (Function) Mode

```
FUN▶ 1    STOP

Stop Input
         :I  5
```

The FUN mode is used to change the FUN table settings for the MICRO^3 base unit, program loader, and memory card. See page page 5-1.

Internal Memory and User Memory

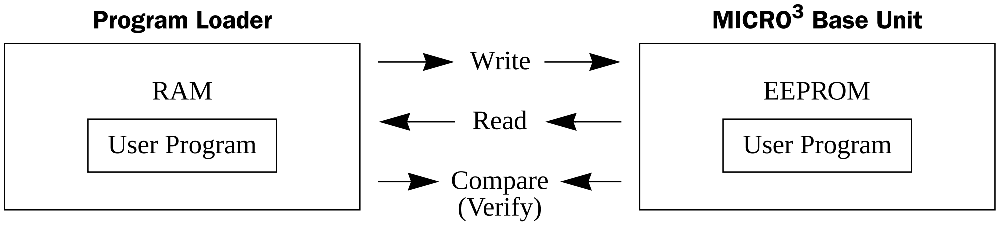
When a user program and FUN settings are edited using the program loader, the data is stored in the internal RAM of the program loader. When the user program is transferred to the MICRO^3 base unit, FUN1 through FUN11 settings are also transferred.

When program transfer and verification are performed, the user program moves as follows:

Write: The user program and FUN1 through FUN11 settings are written from the program loader to the EEPROM in the MICRO^3 base unit.

Read: The user program and FUN1 through FUN11 settings are read from the MICRO^3 base unit to the internal RAM of the program loader.

Verify: User programs and FUN1 through FUN11 settings are compared between the MICRO^3 base unit and the internal RAM of the program loader.



Programming Procedures and Precautions

This section describes the programming procedures using the program loader.

Supply Power

The program loader can be powered in two ways. When the program loader is connected to the MICRO^3 base unit using the loader cable, the program loader is powered by the base unit. Another way is to use an AC adaptor to power the program loader directly. For specifications of an applicable AC adapter, see page A-4.

When powered up, the program loader beeps and displays the initial screen.

While the initial screen is displayed, pressing any key except the FUN, MON, and TRS keys enables the editor mode and displays the user program stored in the program loader internal RAM. The FUN key calls the FUN setting mode, the MON key the monitor mode, and the TRS key the transfer mode.

```
*** Power on ***

Prg.Size  1Kstep
System Ver  1.02
```

Delete User Program from Program Loader

To delete the entire user program from the program loader, press the following keys on the program loader. When the user program is deleted, the FUN1 through FUN10 settings are also deleted.

DEL END ↵

```
   0▶END
   1 END
   2 END
   3 END
```

Change FUN Settings

See FUN Settings on page 5-1.

Create a User Program

See Using Editor Mode on the next page.

Check User Program

When programming is completed, check the user program by pressing the following keys.

FUN 3 BPP 0 ▼

When the program is correct:

```
FUN▶30    CHECK1

  Program  OK
```

When the program is incorrect:

```
FUN▶30    CHECK1
  FUN  Error   1  ◀── FUN1 error
  OP.  Error 100  ◀── Operand error at address 100
  MCR  Error 200  ◀── MCR error at address 200
```

Correct the program and check the program again.
For error details, see FUN30 on page 5-12.

Transfer the User Program from the Program Loader to the MICRO^3 Base Unit

To transfer the user program to the MICRO^3 base unit, press the following keys.

TRS ↵ ↵

```
TRS       1Kstep
    (Write)
 Loader  ➧ PC
      END
```

Note: When the MICRO^3 base unit is running, user programs cannot be transferred. To stop the MICRO^3 operation, set the RUN/STOP switch on the program loader to STOP.

When the user program in the MICRO^3 base unit is write- or read/write-protected, “Protected PC” is displayed. Before transferring the user program, cancel the program protection using FUN22. See page 5-10.

```
TRS       1Kstep
    (Write)
 Loader  ➧ PC
--Protected PC--
```

Using Editor Mode

The editor mode is used to create and revise user programs by writing, deleting, and inserting program instructions in the internal RAM (program memory) of the program loader. The display can be scrolled and a selected program instruction or address can be searched for in the editor mode.

Deleting Entire User Program

The entire user program can be deleted by clearing the user program memory. When the program memory is cleared, FUN1 through FUN10 settings are also cleared to the default values.

Before creating a new program, delete the entire program from the program memory in the program loader.

To delete all program instructions, press the keys:

DEL 7 END ↵

To delete only the user program without deleting FUN1 through FUN10 settings, see Deleting Program Instructions on page 3-6.

Selecting Program Addresses and Displaying Instructions

It is possible to select a program address and read out the instruction on the display. Selecting program addresses is possible whether MICRO3 is running or not. Press the ADRS key, enter an address to which to jump, and press the ▼ key to start. Selecting address is started in either direction by pressing the ▼ key.

Example: Jump to Address 50

Enter the address to jump to:

ADRS 5 CC= 0

```
ADRS▸  50
 101 OR    Q  1
 102 ANDN  I 10
 103 OUT   Q  1
```

To start the search, press the key:

▼ Searches through the program and displays the specified address.

```
  50▸AND   T  7
  51 LOD   M 10
  52 AND   M 11
  53 AND   M 12
```

CLR CLR CLR To display the first instruction at address zero.

7 END ▼ To display the address where an END statement first occurs.

The program loader usually displays 4 lines of the program in the editor mode. The display window can be scrolled to read the program. To scroll up or down through instructions surrounding a particular address, first select the desired address (shown above).

▲ To scroll up through preceding instructions.

▼ To scroll down through following instructions.

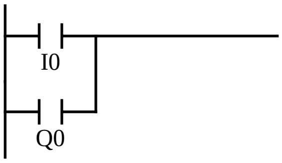
If the ▼ key is pressed when the program loader is displaying the last address, the program loader beeps twice to signal an error.

Entering Program Instructions

Program instructions are entered to the program memory at the selected address in the program loader. A new program instruction overwrites the existing program instruction at the selected address.

Move the cursor where you want to enter a program instruction using the ▼ and ▲ keys. To select an address, press the ADRS key, address number, and the ▼ key.
Enter an instruction and operand.
Press the ↵ key to enter the program instruction into the program memory.

Example: Enter an OR program shown below at address 101.



Prgm Adrs	Instruction	Data
101	LOD	I0
102	OR	Q0
103		

Move the cursor to address 101 by pressing keys:

ADRS | 1 BPS | 0 | 1 BPS | ▼

```
101▶END
102 END
103 END
104 END
```

Enter a load instruction and input I0 by pressing keys:

LOD 10 | SET I | 0 | ↵

```
101 LOD   I  0
102▶END
103 END
104 END
```

When an instruction and an operand are entered, the cursor moves to the next address.

The program loader checks the instruction word and operand when a key is pressed. When the instruction word and operand are correct, the program loader beeps once. When incorrect, the program loader beeps twice to signal an error. The beep sound can also be silenced using FUN34. See page 5-14.

Enter an OR instruction by pressing keys:

E/OR D | F/RST Q | 0 | ↵

```
101 LOD   I  0
102 OR    Q  0
103▶END
104 END
```

Note: When the same output or internal relay number is programmed for the OUT instruction, the program loader beeps and signals Double Out Error, but the output instruction is written in the program memory.

Deleting Program Instructions

One or more program instructions can be deleted from the user program.

To delete a program instruction, move the cursor to the address, and press the keys:

DEL | ↵

To delete several program instructions continuously, move the cursor to the first address to delete, and press the key:

DEL

Then, move the cursor to the last address to delete, and press the key:

▼ | ▼ | ▼ | • • • | ▼ | ↵

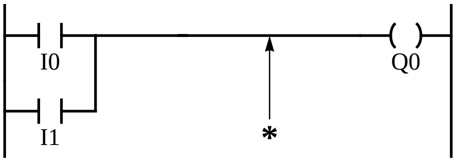
When the delete operation is completed, the remaining program is shifted up.

Inserting Program Instructions

A program instruction can be inserted at any address. When a program instruction is inserted, subsequent program instructions are shifted down.

Move the cursor to the address where you want to insert a program instruction.
Press the INS key.
Enter the program instruction to insert, and press the ↵ key.
To insert more instructions, continue to enter the instructions.
To return to the normal editor mode, press the CLR key.

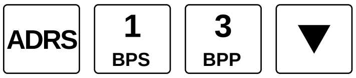
Example: Insert an NO contact of input I2 at the position * in the diagram below

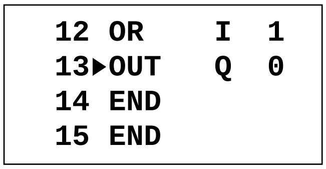


Prgm Adrs	Instruction	Data
11	LOD	I0
12	OR	I1
13	OUT	Q0

This insertion is done by inserting an AND instruction between addresses 12 and 13.

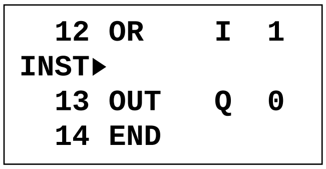
Move the cursor to address 13 by pressing keys:





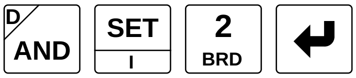
```
  12 OR    I  1
  13▶OUT   Q  0
  14 END
  15 END
```

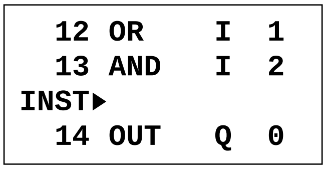
Press the insert key:



```
  12 OR    I  1
INST▶
  13 OUT   Q  0
  14 END
```

Enter the AND instruction and operand I2:

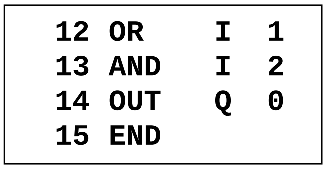




```
  12 OR    I  1
  13 AND   I  2
INST▶
  14 OUT   Q  0
```

The cursor moves to the next line.

To return the normal editor mode, press the clear key:



```
  12 OR    I  1
  13 AND   I  2
  14 OUT   Q  0
  15 END
```

Searching for a Program Instruction

A specified program instruction can be searched for through the user program.

Enter an instruction to search for and start the search by pressing the appropriate key.

▲ Searches to the smaller address starting at the cursor position.

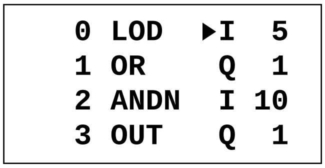
▼ Searches to the larger address starting at the cursor position.

B/REP ▲ or B/REP ▼ Repeats the search for the same instruction.

Example: Search for instruction LOD I5 to the larger address

Enter the instruction to search.

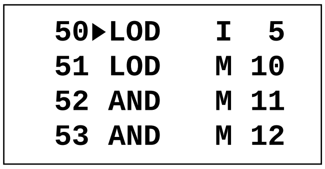
LOD/10 SET/I 5/CC=



```
 0 LOD  ▸I  5
 1 OR    Q  1
 2 ANDN  I 10
 3 OUT   Q  1
```

To start the search, press the key:

▼ Searches to the larger address starting at the cursor position and displays the instruction when found.



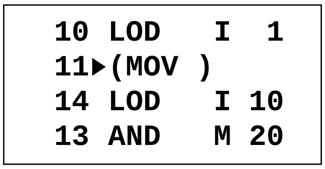
```
50▸LOD   I  5
51 LOD   M 10
52 AND   M 11
53 AND   M 12
```

When the specified instruction is not found, the program loader beeps twice.

Reading Advanced Instructions

To read and edit the S (source) and D (destination) operands of advanced instructions, press the following keys.

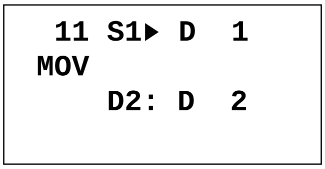
Move the cursor to the address of an advanced instruction to read.



```
10 LOD   I  1
11▸(MOV )
14 LOD   I 10
13 AND   M 20
```

To display the advanced instruction operands, press the key:

▶



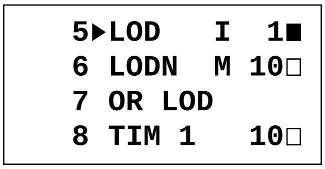
```
 11 S1▸ D  1
MOV
    D2: D  2
```

To return to the normal editor mode, press the CLR key.

Sequential Monitoring

The sequential monitoring can be enabled in the editor mode to monitor input, output, internal relay, timer, and counter statuses at 4 consecutive addresses on the program loader. For the operating procedure to enable the sequential monitoring, see FUN32 on page 5-13.

The sequential monitoring is possible at addresses of LOD, LODN, AND, ANDN, OR, ORN, OUT, OUTN, SET, RST, TIM, and CNT instructions.



```
 5▸LOD   I  1■
 6 LODN  M 10□
 7 OR LOD
 8 TIM 1   10□
```

Using Transfer Mode

User programs can be transferred between the program loader and the MICRO^3 base unit or the memory card installed in the program loader. The transfer mode includes writing, reading, and comparing of user programs.

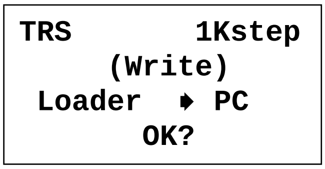
When the user program in the MICRO^3 base unit is protected from writing and/or reading, the program transfer operation cannot be performed. For user program protection, see FUN22 on page 5-10.

Writing Program from Program Loader to MICRO^3 Base Unit

A user program can be transferred from the program loader to the MICRO^3 base unit only when the MICRO^3 base unit is stopped. When a user program is transferred, FUN1 through FUN11 settings are also transferred.

Make sure that the MICRO^3 base unit is stopped and press the keys:

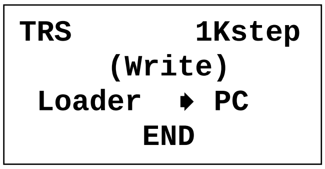
TRS ↵



```
TRS       1Kstep
    (Write)
 Loader  ▶ PC
      OK?
```

To start the program transfer, press the enter key:

↵



```
TRS       1Kstep
    (Write)
 Loader  ▶ PC
      END
```

When the transfer is completed, “END” is displayed.

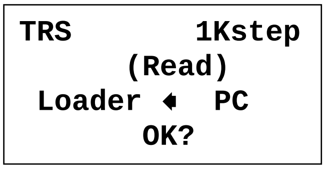
Note: When error occurs during program transfer, see “Error Messages for Program Loader Operation” on page 20.

Reading Program from MICRO^3 Base Unit to Program Loader

A user program can be transferred from the MICRO^3 base unit to the program loader whether the MICRO^3 base unit is running or not. When a user program is transferred, FUN1 through FUN11 settings are also transferred.

Press the keys:

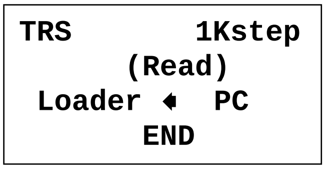
TRS ▼



```
TRS       1Kstep
     (Read)
 Loader ◀  PC
      OK?
```

To start the program transfer, press the enter key:

↵



```
TRS       1Kstep
     (Read)
 Loader ◀  PC
      END
```

When the transfer is completed, “END” is displayed.

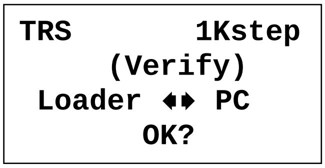
Note: When an error occurs during the program transfer, see “Error Messages for Program Loader Operation” on page 20.

Comparing Programs between Program Loader and MICRO^3 Base Unit

User programs can be compared between the program loader and the MICRO^3 base unit whether the MICRO^3 base unit is running or not. When user programs are compared, FUN1 through FUN11 settings are also compared.

Press the keys:

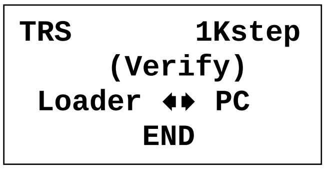
TRS ▲



```
TRS       1Kstep
    (Verify)
 Loader ◀▶ PC
      OK?
```

To start the program comparison, press the enter key:

↵



```
TRS       1Kstep
    (Verify)
 Loader ◀▶ PC
      END
```

When the programs match, “END” is displayed.

Note: When error occurs during program comparison, see “Error Messages for Program Loader Operation” on page 20.

Writing Program from Program Loader to Memory Card

A new memory card must be formatted before writing user programs using FUN41. See “FUN41: Memory Card Formatting” on page 15. Insert a formatted memory card into the program loader. Make sure that the write protect switch on the memory card is set to the write enable side.

Press the keys:

TRS TRS ↵

```
TRS        1Kstep
    (Write)
 Loader   ➧ Card
OK? ▶ 1:********
```

Select a storage address from 1 through 84 to store the user program in the memory card by pressing the key:

▼ or ▲

```
TRS        1Kstep
    (Write)
 Loader   ➧ Card
OK? ▶10:********
```

At storage addresses where no program is stored, 8 asterisks are displayed. At addresses where programs are stored without program names, the program name line is left blank. A new program can be overwritten at any storage address.

To enter a program name, move the cursor to the right by pressing the key:

▶

```
TRS        1Kstep
    (Write)
 Loader   ➧ Card
OK?  10▶*******
```

Select a character at each position to enter a program name of 8 characters maximum. When FUN35 display language selection is set to English, characters A to Z, 0 to 9, and a space can be used for the program name. To select a character, press the key:

▼ or ▲

```
TRS        1Kstep
    (Write)
 Loader   ➧ Card
OK? ▶10:IDEC003*
```

Move the cursor to the next or previous position by pressing the ▶ or ◀ key.

After entering a program name, start to transfer the program from the program loader to the memory card by pressing the key:

↵

```
TRS        1Kstep
    (Write)
 Loader   ➧ Card
END    :IDEC003
```

When the transfer is completed, “END” is displayed.

Note: When an error occurs during program transfer, see “Error Messages for Program Loader Operation” on page 20.

Reading a Program from Memory Card to Program Loader

Insert a memory card containing user programs into the program loader.

Press the keys:

TRS TRS ▼

```
TRS        1Kstep
     (Read)
 Loader ◀  Card
OK? ▶ 1:PROGRAM1
```

Select a program name in the memory card as described above and start the program transfer by pressing the keys:

▼ or ▲ ↵

```
TRS        1Kstep
     (Read)
 Loader ◀  Card
END    :IDEC003
```

When the transfer is completed, “END” is displayed.

Note: When an error occurs during program transfer, see “Error Messages for Program Loader Operation” on page 20.

Comparing Programs between Program Loader and Memory Card

Insert a memory card containing user programs into the program loader.

Press the keys:

TRS TRS ▲

```
TRS        1Kstep
     (Verify)
 Loader ◂▸ Card
OK? ▶ 1:PROGRAM1
```

Select a program name in the memory card as described on the preceding page and start the program comparison by pressing the keys:

▼ or ▲ ↵

```
TRS        1Kstep
     (Verify)
 Loader ◂▸ Card
END    :IDEC003
```

When the programs match, “END” is displayed. When the programs do not match, NG is displayed.

Note: When an error occurs during the program comparison, see “Error Messages for Program Loader Operation” on page 20.

Deleting a Program from the Memory Card

A user program can be deleted from the memory card. Insert a memory card containing user programs into the program loader.

Press the keys:

TRS TRS DEL

```
TRS        1Kstep
     (Delete)
           Card
OK? ▶ 1:PROGRAM1
```

Select a program name in the memory card as described on the preceding page and start the program deletion by pressing the key:

↵

```
TRS        1Kstep
     (Delete)
           Card
       END
```

When the program is deleted, “END” is displayed.

To delete all programs from the memory card, see “FUN41: Memory Card Formatting” on page 15.

Using Monitor Mode

The monitoring mode is enabled by pressing the MON key on the program loader. The statuses of inputs, outputs, internal relays, shift registers, the preset and current values of timers and counters, and the data of data registers can be monitored on the program loader. Changing timer/counter preset values, entering data into data registers, and setting/resetting are also enabled in the monitor mode. The program loader has 3 lines to display the monitor data.

Monitoring I/O, Internal Relays, and Shift Registers

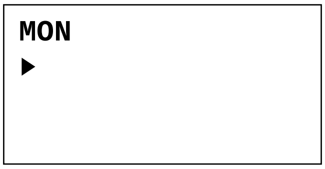
Bit operands such as inputs, outputs, internal relays, and shift register statuses are displayed in a group of 8 points starting at the designated number.

Press the MON key to enable the monitor mode.
Enter the operand and number to monitor.
Press the ▼ key to start monitoring.

Example: Monitor inputs I0 through I7

To enable the monitor mode, press the key:

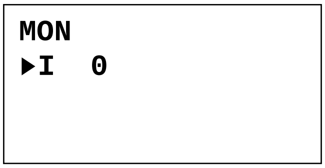
MON



```
MON
▶
```

Enter the operand and the first number to monitor:

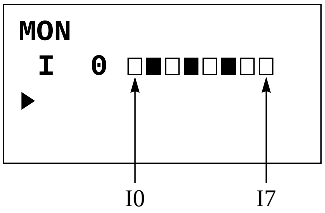
SET / I   0



```
MON
▶I  0
```

Start monitoring by pressing the key:

▼

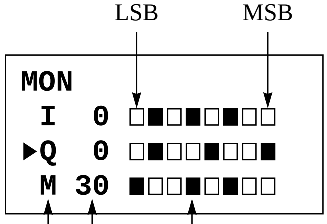


```
MON
 I  0 □■□■□■□□
▶
```

The ON/OFF statuses of inputs I0 through I7 are displayed.

To return to the editor mode, press the CLR key.

The monitor screen shows 3 lines to monitor different operands. For example, when input I0, output Q0, and internal relay M30 are specified as monitor data, the screen displays as shown below:



```
MON
 I  0 □■□■□■□□
▶Q  0 □■□□■□□■
 M 30 ■□□■□■□□
```

■ ON □ OFF

The second line shows I0 through I7.
The third line shows Q0 through Q7.
The fourth line shows M30 through M37.

Displays the statuses of 8 points starting at the first number.
The first operand number to monitor
The operand to monitor

To monitor the preceding or next 8 points of the same operand, move the cursor to the line and press the ◀ or ▶ key, respectively.

Monitoring Timers and Counters

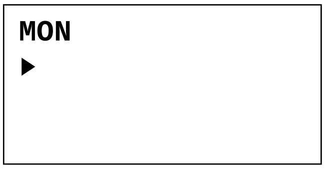
The preset and current values of timers and counters are displayed.

Press the MON key to enable the monitor mode.
Enter the operand and number to monitor.
Press the ▼ key to start monitoring.

Example: Monitor timer T10

To enable the monitor mode, press the key:

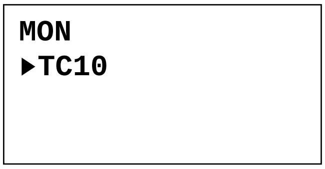
MON



```
MON
▶
```

Enter the operand and number to monitor:

TIM/T 1/BPS 0

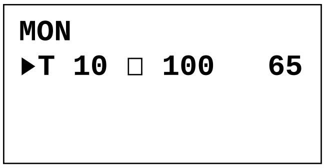


```
MON
▶TC10
```

When pressing the TIM or CNT key, “TC” is displayed until monitoring is started.

Start monitoring by pressing the key:

▼

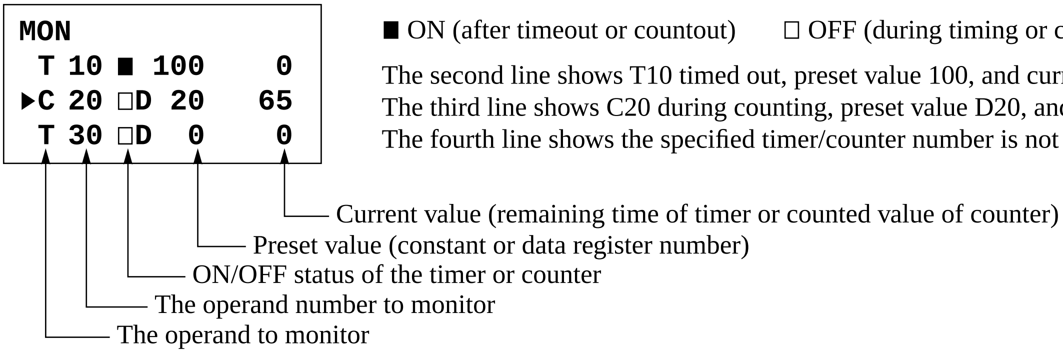


```
MON
▶T 10 □ 100   65
```

The preset and current values are displayed.

To return to the editor mode, press the CLR key.

The monitor screen shows 3 lines to monitor different operands. For example, when timer T10, counter C20, and counter C30, which is not programmed, are specified as monitor data, the screen displays as shown below:



```
MON
 T 10 ■ 100    0
▶C 20 □D 20   65
 T 30 □D  0    0
```

■ ON (after timeout or countout) □ OFF (during timing or counting)

The second line shows T10 timed out, preset value 100, and current value 0.
The third line shows C20 during counting, preset value D20, and current value 65.
The fourth line shows the specified timer/counter number is not programmed.

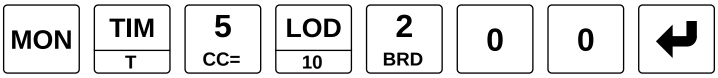
Current value (remaining time of timer or counted value of counter)
Preset value (constant or data register number)
ON/OFF status of the timer or counter
The operand number to monitor
The operand to monitor

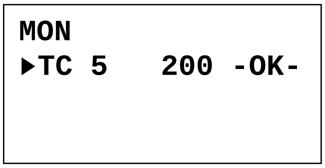
To monitor the preceding or next number of the same operand, move the cursor to the line and press the ◀ or ▶ key, respectively.

Changing Preset Values for Timers and Counters

Preset values for timers (TIM, TMH, and TMS) and counters can be changed by transferring a new value to the MICRO^3 base unit RAM. This is possible whether the base unit is running or not. Only preset values programmed with a constant value can be changed using a constant value by this operation. Preset values designated with a data register can also be changed by entering a new preset value to the data register. To change the preset value for a timer or counter, press the MON key, TIM or CNT key, the operand number, the LOD/10 key, the new preset value, and the ↵ key.

Example: Change timer TIM5 preset value to 200





```
MON
▸TC 5   200 -OK-
```

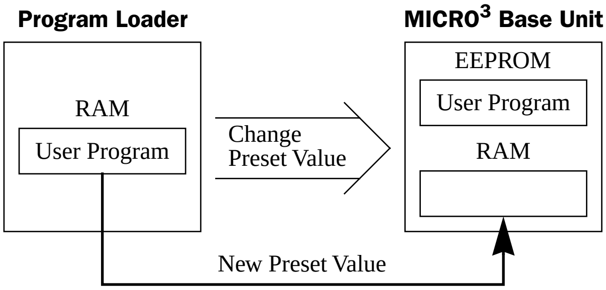
When the preset value is changed correctly, “OK” is displayed. If not, “NG” is displayed. To return to the editor mode, press the CLR key.

If the timer preset value is changed during timedown or after timeout, the timer remains unchanged for that cycle. The change becomes effective for the following timedown cycle. If the timer preset value is changed to zero, then the timer operation stops and the timer output is turned on immediately.

If the counter preset value is changed during counting, the new preset value becomes effective immediately. If the new preset value is smaller than or equal to the current value, the counter output goes on as soon as the new preset value is transferred. If the counter preset value is changed after countout, the counter output remains on until reset.

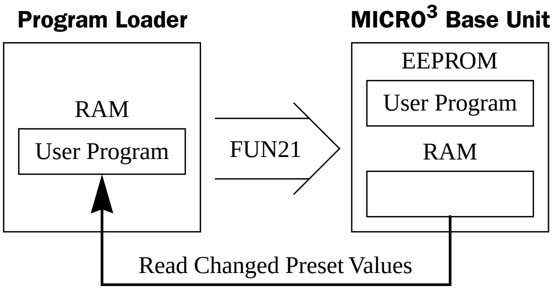
Data movement when changing a timer/counter preset value

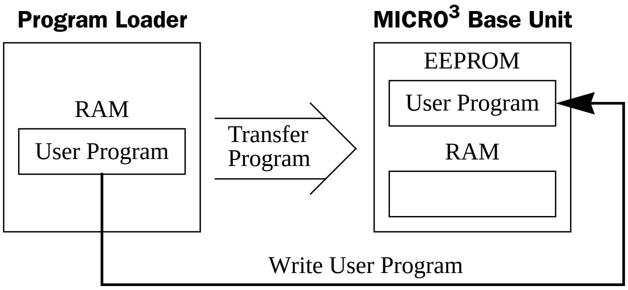
When changing a timer/counter preset value using the program loader, the new preset value is written into the MICRO^3 base unit RAM. The user program and preset values in the EEPROM are not changed.



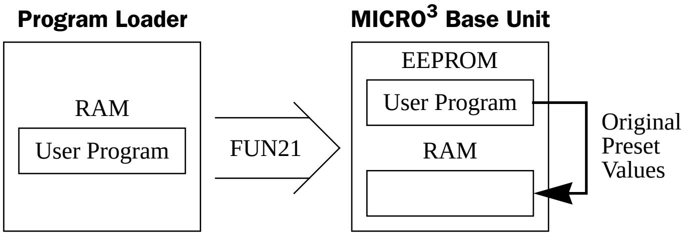
Moving data when writing changed preset values into user program

The changed timer/counter preset values can be read out from the MICRO^3 base unit RAM to the program loader using FUN21 (see page 5-9). To update the preset values for the user program in the EEPROM, transfer the user program from the program loader to the EEPROM.





Moving data when clearing changed preset values to restore original values



Changing preset values for timers and counters in the MICRO^3 base unit RAM does not automatically update preset values in the user memory, EEPROM. This is useful for restoring previous preset values using FUN21 (see page 5-9). When the program in the loader is rewritten to the EEPROM without using FUN21 to update preset values beforehand, existing values are transferred to the EEPROM and overwrite the modified values in the RAM, also. When the changed timer/counter preset values are cleared from the RAM using FUN21, the original preset values are written from the EEPROM to the RAM.

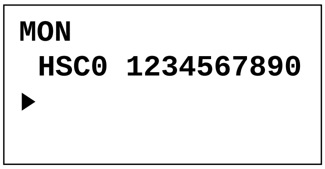
Monitoring High-speed Counters

The preset and current values of high-speed counters HSC0 through HSC3 are displayed in decimal or hexadecimal notation on the program loader.

Press the MON key to enable the monitor mode.
Enter the CNT and ADV keys, followed by the operand number 0 through 3 to monitor.
Press the LOD/10 or OUT/16 key to display the preset or current value in decimal or hexadecimal notation. If not pressed, the value is displayed in the data type selected by FUN36. See “FUN36: Display Data Type Selection” on page 14. The decimal or hexadecimal notation can also be switched after starting the monitor.
Press the ▼ key to start monitoring.
To return to the editor mode, press the CLR key.

Example: Monitor high-speed counter HSC0 in decimal notation (FUN36 set to select decimal notation)

MON | CNT C | ADV | 0 | ▼



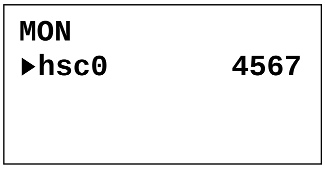
```
MON
 HSC0 1234567890
▶
```

When the ▼ key is pressed, the capital letters, HSC0, are displayed followed by the double-word preset value of the specified high-speed counter number, and the cursor moves down to the next line. When a data register is designated as source operand S1 for preset value, the data register number is displayed in place of the preset value.

To monitor the current value, press the ▲ key to move the cursor one line up, and press the ▶ key.

▲ | ▶

HSC: Preset value
hsc: Current value

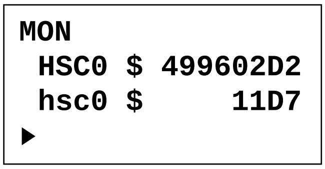


```
MON
▶hsc0       4567
```

To change the preset or current value notation between decimal and hexadecimal, place the cursor on the line, and press the LOD/10 or OUT/16 key followed by the ▼ key.

When the preset and current values of the above are monitored in hexadecimal notation, the display will be as shown on the right.

OUT 16 | ▼



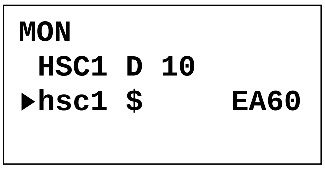
```
MON
 HSC0 $ 499602D2
 hsc0 $     11D7
▶
```

With the cursor placed at hsc0, pressing the ▶ key again will display the preset value of the next high-speed counter, HSC1 in this example. The preset value and current value of the next or preceding high-speed counter is displayed by pressing the ▶ or ◀ key, respectively.

When high-speed counter HSC1 for multi-stage comparison is monitored, the data register designated by source operand S1 is displayed as the preset value.

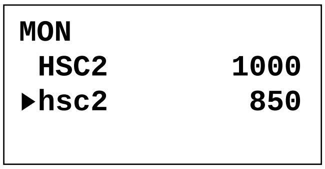
To monitor HSC1 preset value in hexadecimal notation, press the keys:

MON | CNT C | ADV | 1 BPS | OUT 16 | ▼ | ▲ | ▶



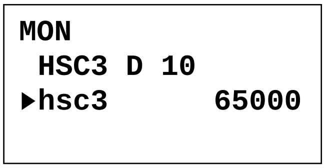
```
MON
 HSC1 D 10
▶hsc1 $     EA60
```

Similarly, when high-speed counter HSC2 for pulse output control is monitored, the preset value or data register designated by source operand S1 is displayed following HSC2, and the current value is displayed following hsc2.



```
MON
 HSC2       1000
▶hsc2        850
```

When high-speed counter HSC3 for gate control is monitored, which has no preset value, then the data register designated by destination operand D1 is displayed following HSC3. The current value is displayed following hsc3.



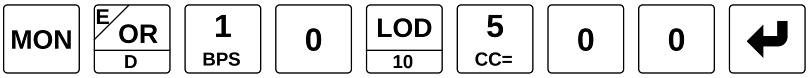
```
MON
 HSC3 D 10
▶hsc3      65000
```

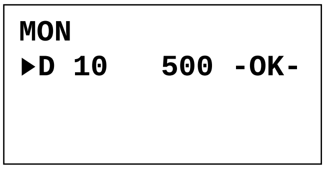
Entering Data into Data Registers

Data in data registers can be changed using the program loader whether the MICRO^3 base unit is running or not. Data register values can be entered in decimal or hexadecimal notation. In addition, a double-word value can also be entered into two consecutive data registers in decimal or hexadecimal notation. This function is particularly useful for entering a preset value for a high-speed counter when data registers are used for a preset value.

Press the MON key to enable the monitor mode.
Enter the D key, followed by the data register number to which to enter data.
Press the ADV key to enter a double-word value into two consecutive data registers. If not pressed, a one-word value is entered into the designated data register.
Press the LOD/10 or OUT/16 key to enter the value in decimal or hexadecimal notation, followed by the value to enter.
Press the ↵ key to enter the value into the data register. “OK” is displayed for approximately 1 second.
To return to the editor mode, press the CLR key.

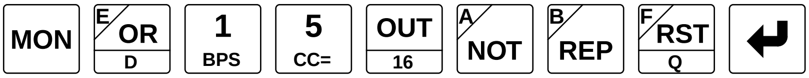
Example: Enter decimal value 500 into data register D10

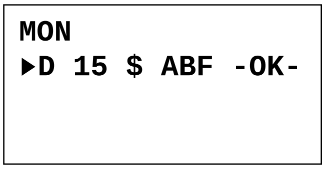




```
MON
▸D 10   500 -OK-
```

Example: Enter hexadecimal value ABF into data register D15

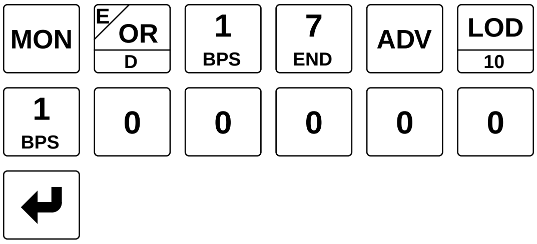


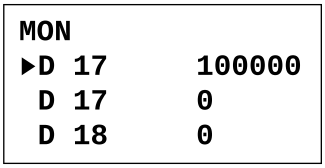


```
MON
▸D 15 $ ABF -OK-
```

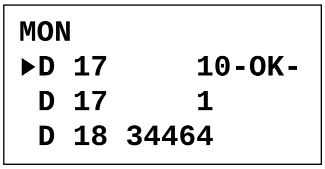
The $ symbol is displayed to show the hexadecimal data type.

Example: Enter double-word decimal value 100,000 into data registers D17 and D18





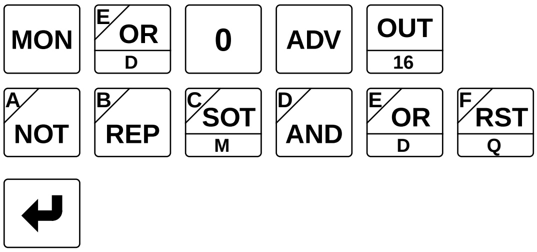
```
MON
▸D 17     100000
 D 17     0
 D 18     0
```

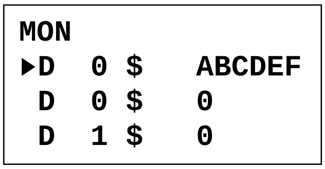


```
MON
▸D 17     10-OK-
 D 17     1
 D 18 34464
```

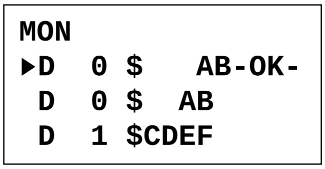
If FUN36 is set to select decimal data type, the LOD/10 key may be omitted to enter a decimal value. The upper word of the double-word decimal value enters into the specified data register, and the lower word enters into the next data register. The third and fourth lines in this example display the data of individual data registers D17 and D18, respectively.

Example: Enter double-word hexadecimal value ABCDEF into data registers D0 and D1





```
MON
▸D  0 $   ABCDEF
 D  0 $   0
 D  1 $   0
```



```
MON
▸D  0 $   AB-OK-
 D  0 $  AB
 D  1 $CDEF
```

The $ symbol is displayed to show the hexadecimal data type.

If FUN36 is set to select hexadecimal data type, the OUT/16 key may be omitted to enter a hexadecimal value. The upper word of the double-word hexadecimal value enters into the specified data register and the lower word enters into the next data register. The third and fourth lines in this example display the data of individual data registers D0 and D1, respectively.

Note: To clear data of all data registers to zero, use FUN26 Operand Data Clear. See page page 5-11. When the reset input is turned on, all data register values are also cleared to zero. See page 2-2.

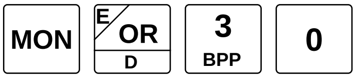
Monitoring Data Registers

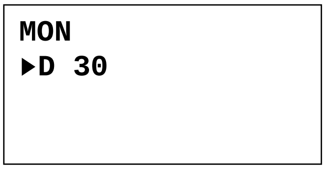
The data of data registers can be displayed in decimal or hexadecimal notation on the program loader. In addition, the double-word data of two consecutive data registers can also be displayed in decimal or hexadecimal notation. This function is particularly useful for confirming the preset value of high-speed counters when data registers are used for a preset value.

Press the MON key to enable the monitor mode.
Enter the D key, followed by the data register number to monitor.
Press the ADV key to display the double-word data of two consecutive data registers. If not pressed, the one-word value of the designated data register is displayed.
Press the LOD/10 or OUT/16 key to display the value in decimal or hexadecimal notation. If not pressed, the value is displayed in the data type selected by FUN36. See "FUN36: Display Data Type Selection" on page 14. The decimal or hexadecimal notation can also be switched after starting the monitor.
Press the ▼ key to start monitoring.
To return to the editor mode, press the CLR key.

Example: Monitor data register D30 in decimal notation (FUN36 set to select decimal notation)

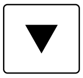
Enable the monitor mode, and enter the operand and number by pressing the keys:

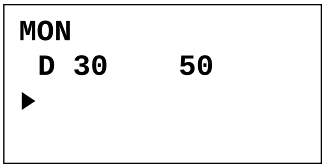




```
MON
▸D 30
```

Start monitoring by pressing the key:



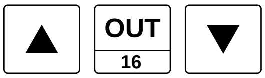


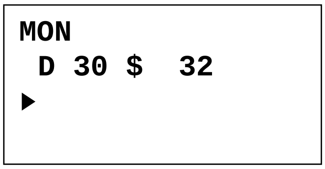
```
MON
 D 30    50
▸
```

The data of data register D30 is displayed.

To change the data register value notation between decimal and hexadecimal, place the cursor on the line, and press the LOD/10 or OUT/16 key followed by the ▼ key.

Example: Change decimal value of data register D30 to hexadecimal



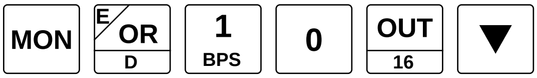


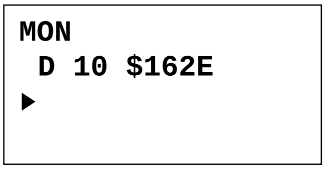
```
MON
 D 30 $  32
▸
```

The $ symbol is displayed to show the hexadecimal data type.

To return to the editor mode, press the CLR key.

Example: Monitor data register D10 in hexadecimal notation

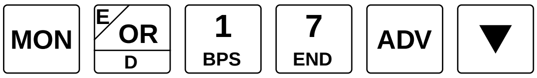


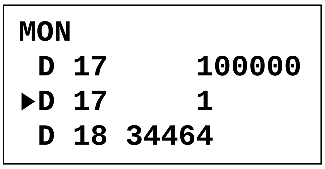


```
MON
 D 10 $162E
▸
```

The $ symbol is displayed to show the hexadecimal data type.

Example: Display double-word value of data registers D17 and D18 in decimal notation



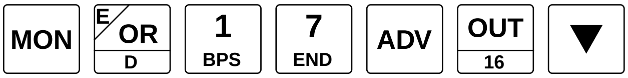


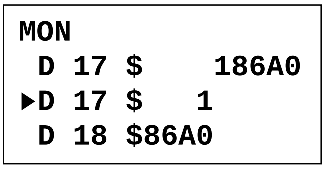
```
MON
 D 17      100000
▸D 17      1
 D 18 34464
```

The specified data register comprises the upper word and the next data register the lower word of the monitored double-word data.

The third and fourth lines in this example display the data of individual data registers D17 and D18, respectively.

Example: Display double-word value of data registers D17 and D18 in hexadecimal notation





```
MON
 D 17 $    186A0
▸D 17 $   1
 D 18 $86A0
```

The $ symbol is displayed to show the hexadecimal data type.

The third and fourth lines in this example display the hexadecimal data of data registers D17 and D18, respectively.

Setting and Resetting

⚠ Caution • Make sure of safety when operating the MICRO³ to force outputs on (SET) or off (RST). Incorrect operation on the MICRO³ may cause machine damage or accidents.

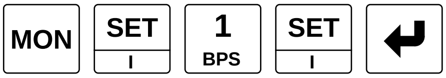
Inputs, outputs, internal relays, and shift register bits can be temporarily turned on (SET) or turned off (RST), using the program loader.

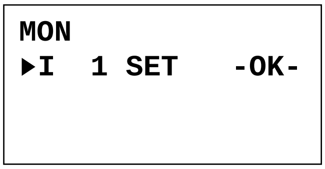
Inputs and outputs can be set or reset only while the MICRO³ base unit is running. The designated input or output is set or reset at the first execution of the END instruction after the ↵ key is pressed. After executing the END instruction, the input reflects the actual input, and the output is operated according to the existing program.

Internal relays and shift register bits can be set or reset whether the MICRO³ base unit is running or not. When setting or resetting internal relays or shift register bits, the on or off status becomes in effect as soon as the ↵ key is pressed. If the internal relay or shift register bit is designated with “keep” status, then the set or reset operation remains in effect after the MICRO³ base unit is tuned on. For “keep” designation, see FUN3 on page 5-4 and FUN4 on page 5-5.

To set or reset an operand, press the MON key, the operand and number, followed by the SET or RST key, and the ↵ key. When the operand is correctly set or reset, “OK” is displayed. If not, the program loader will beep.

Example: Set input I1

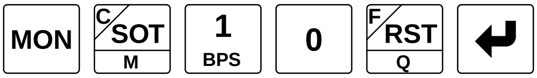


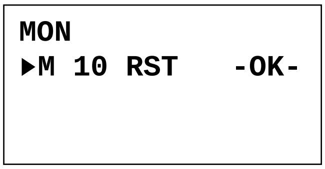


```
MON
▶I  1 SET   -OK-
```

When the input is set correctly, “OK” is displayed.

Example: Reset internal relay M10

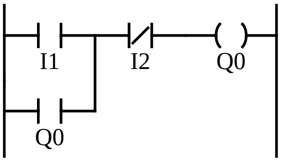




```
MON
▶M 10 RST   -OK-
```

When reset correctly, “OK” is displayed.

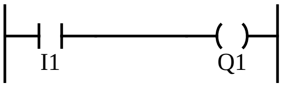
When input I1 is turned on (SET), the circuit on the right will be actuated to hold output Q0. If the NC input I2 is turned off using the SET operation, the circuit will return to its non-actuated status.



Timing for SET and RST Operation

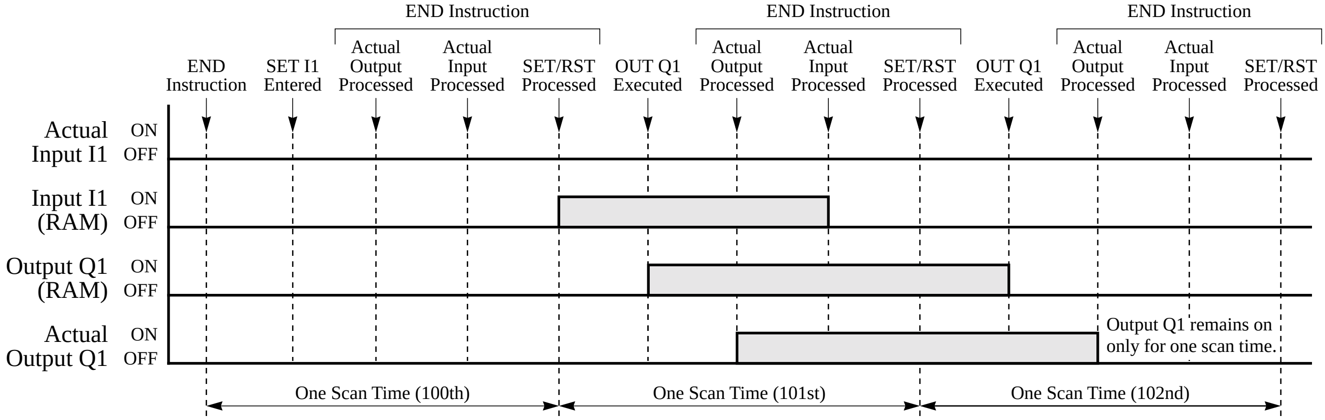
The SET or RST operation is set to the MICRO³ base unit RAM when the first END instruction is executed after pressing the ↵ key. The subsequent sequence is executed according to the user program. Inputs are updated depending on actual external inputs. Outputs, internal relays, and shift registers are updated according to the user program.

In the END execution, the processing occurs on actual output processing, actual input processing, and SET/RST processing in this order. When input I1 is turned on using the SET operation in the program on the right while the actual external input remains off, the result is reflected as follows.



If input I1 is set using the SET operation in the 100th scan, input I1 in the RAM is turned on in the 101st scan, which turns output Q1 in the RAM on. As a result, actual output Q1 is turned on when the END instruction is executed in the 101st scan. Since actual input I1 is off in the 102nd scan, I1 and Q1 in the RAM are turned off. Consequently, actual output Q1 is turned off when the END instruction is executed in the 102nd scan. See the timing chart on the next page.

Time Chart for SET and RST Operation



Error Messages for Program Loader Operation

When using the program loader for programming or transferring a user program, the following error messages may be displayed.

Error Message	Error Details
Calendar NG	Invalid calendar data.
CRC Code NG	The CRC code of the user program to be transferred is incorrect.
Data Clear NG	The designated data cannot be initialized.
Double Out Error	The same output operand is used repeatedly.
Expansion Unit	Program loader is connected to the MICRO^3 base unit at the expansion station.
No Connect	Memory card is not inserted.
Operand NG	Invalid operand.
Pass Word NG	Incorrect password.
PC Run Error	The MICRO^3 base unit is in the run state, and user program cannot be transferred.
Prg. NG–(A)	User programs are different between the program loader and the MICRO^3 base unit. Address is displayed in 4 digits.
Prg. NG–(FUN)	FUN settings are different between the program loader and the MICRO^3 base unit. FUN number is displayed in 2 digits.
Prg. Size NG	Invalid user program size selection (FUN11).
Program Over	User program exceeds the program size selected with FUN11.
Protected Card	The write protect switch on the memory card is set to protect.
Protected PC	The MICRO^3 base unit is read and/or write protected.
Receive Error	Line disconnection.
Receive Error 0	Invalid BCC code of the received data.
Receive Error 1	Data parity, framing, or overrun error.
Receive Error 2	Time over between data characters.
Receive Error 3	Invalid communication command.
Receive Error 4	Invalid communication procedure (protocol).
System Card ????	A system card is inserted. The ID number is displayed in 4 digits.
T/C Data NG	Invalid timer/counter data.
Unformat Card	The memory card is not formatted to store user programs.
Unrecognized One	Incompatible memory card.
Data Over	Transmit/receive data designation exceeds 200 bytes (MICRO^{3}C only).
Setting NG	The protocol selector switch is not set to 3 when using FUN50 user communication data monitor (MICRO^{3}C only).
Speed Mode Error	TXD/RXD is programmed in the high-speed mode (MICRO^{3}C only).

Introduction

MICRO^3 features special functions such as the high-speed processing mode, catch input function, input filter function, pulse output function, high-speed counter function, expansion and data link functions, external analog timer function, and analog I/O functions. This chapter describes these special functions.

High-speed Processing Mode

MICRO^3 can execute the user program in the standard processing mode and the high-speed processing mode. The high-speed processing mode is ideal for using MICRO^3 as a sensor controller or executing a user program when high-speed processing is required. The processing mode can be selected using FUN5 on the program loader. See page 5-5.

Using the high-speed processing mode, program capacity and available operand numbers are limited as shown below.

Processing Speed and Program Capacity

	High-speed Processing Mode	Standard Processing Mode
Processing Time: Basic Instructions	0.45 µsec average 0.2 µsec minimum	2.2 µsec average 1.2 µsec minimum
Scan Time	400 µsec/100 steps	2.9 msec/1k steps
Program Capacity	Approx. 100 steps	1012 steps

LOD, LODN, AND, ANDN, OR, ORN, OUT, OUTN, SET, RST, AND LOD, and OR LOD instructions are processed faster in the high-speed processing mode. Other instructions do not have to be processed faster in the high-speed processing mode.

The average scan time is not equal to the total of instruction processing times because processing of other than user program instructions is involved.

Available Operands and Allocation Numbers

	High-speed Processing Mode		Standard Processing Mode	
Input (see note)	14 points	I0 to I15	28 points	I0 to I35
Output (see note)	10 points	Q0 to Q11	20 points	Q0 to Q31
Internal Relay	40 points	M0 to M47	232 points	M0 to M287
Catch Input Relay	8 points	M290 to M297	8 points	M290 to M297
Special Internal Relay	16 points	M300 to M317	16 points	M300 to M317
Data Register	32 points	D0 to D31	100 points	D0 to D99
Timer	16 points total	T0 to T15	32 points total	T0 to T31
Counter		C0 to C15		C0 to C31
Shift Register	32 points	R0 to R31	64 points	R0 to R63

Note: Available input and output numbers depend on the MICRO^3 base unit used.

Limitations on High-speed Processing Mode

Using the high-speed processing mode, the following functions are limited:

- The program capacity is limited to approximately 100 steps.
- Expansion link and data link functions cannot be used.
- Available operand numbers are limited as shown above.
- Control data registers D90 through D99 cannot be used.

Catch Input Function

The catch input function is used to receive short pulses from sensor outputs regardless of the scan time. Since input signals to inputs I0 through I7 are always set to special internal relays M290 through M297, input signals are securely received even if short-pulse input signals turn on and off within one scan time.

Input terminals I0 through I7 are assigned to catch inputs and also used for normal inputs. All normal input signals are read when the END instruction is executed at the end of a scan.

Catch Input Terminals and Pulse Widths

10-I/O MICRO^3 base unit:	Catch inputs 6 points (I0 through I5)
16- and 24-I/O MICRO^3 base units:	Catch inputs 8 points (I0 through I7)

Minimum detectable pulse width (when hard filter is set to 10):
Input I0 ON pulse = 28 µsec, Input I0 OFF pulse = 30 µsec
Input I1 to I7 ON pulse = 37 µsec, Input I1 to I7 OFF pulse = 120 µsec

Catch Input Terminals and Special Internal Relays

Each catch input is assigned to a special internal relay to store the catch input signal. Catch input terminals are divided into four groups to select rising or falling edges for catch inputs.

Catch Group	Catch Input Number	Corresponding Special Internal Relay
G1	I0	M290
G2	I1	M291
G3	I2 I3	M292 M293
G4 **(see note)**	I4 I5 I6 I7	M294 M295 M296 M297

Note: The 10-I/O type MICRO^3 base unit has only inputs I4 and I5 in group G4.

Rising or Falling Edge Selection for Catch Inputs (FUN6)

FUN6 is used to select whether catch inputs are accepted at the rising edge (ON pulse) or falling edge (OFF pulse). Select the rising or falling edge for each group using FUN6 on the program loader. For setting FUN6, see page 5-6.

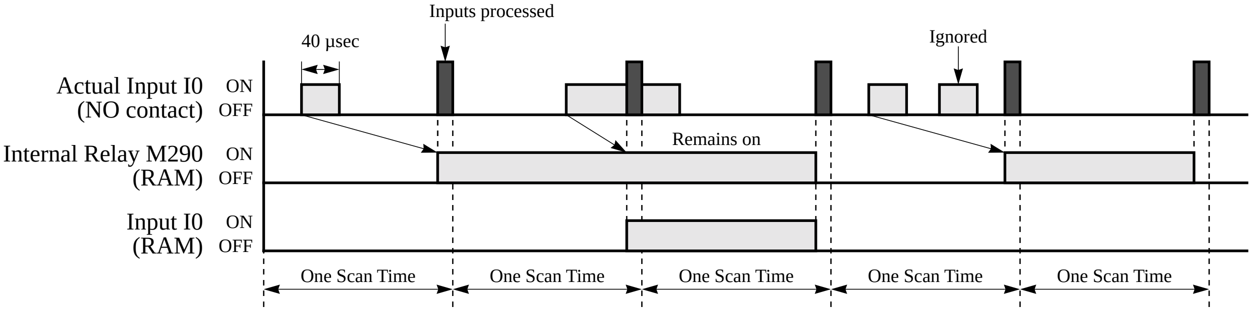
Input Filter Time Selection (FUN7)

To make sure of correct receiving of catch input signals, set the input filter time using FUN7 on the program loader. Only hard filter can be used for catch inputs. The hard filter can be set between 0 and 255 to select the detectable pulse width. For setting FUN7, see page 5-6. For details of the input filter function, see the following pages.

Catch Input vs Normal Input

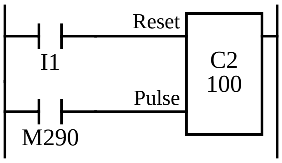
The figure below compares how ON-pulse catch inputs and normal inputs are processed by MICRO^3. In this example, FUN6 is set to select the rising edge to receive ON-pulse catch inputs.

When a short-pulse input enters, the corresponding catch input special internal relay is turned on for the next one scan time. When a catch input turns on in every scan, the corresponding catch input special internal relay remains on.



Example: Counting Catch Input Pulses

This example demonstrates a program to count short pulses using the catch input function.

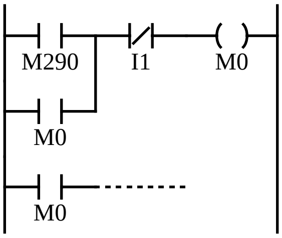


Input I1 is used as a reset input for adding counter C2.
Input I0 is assigned to catch input special internal relay M290.
Counter C2 counts short-pulse inputs to input I0.

Note: When catch inputs M290 through M297 are used as pulse inputs to a counter, the repeat cycle periods of the pulse inputs must be more than 2 scan times.

Example: Maintaining Catch Input

When a catch input is received, a special internal relay assigned to the catch input is turned on for only one scan. This example demonstrates a program to maintain a catch input status for more than one scan.



Input I0 is assigned to catch input special internal relay M290.
When input I0 is turned on, M290 is turned on, and M0 is maintained in the self-holding circuit.
When NC input I1 is turned on, the self-holding circuit is unlatched, and M0 is turned off.

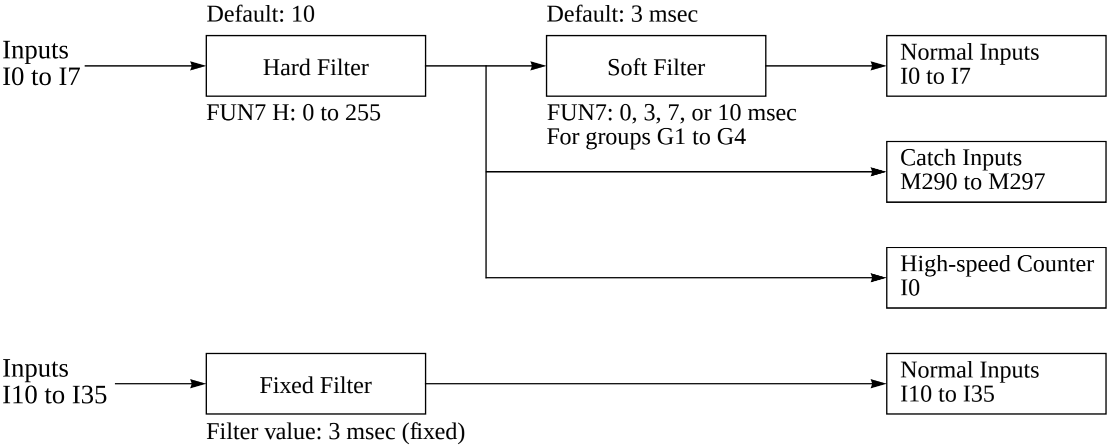
M0 is used as an input condition for the subsequent program instructions.

Input Filter Function

MICRO^3 features the input filter function to select the input pulse widths to read inputs I0 through I7. The input filter ignores pulse inputs shorter than the selected value to prevent malfunction caused by noises.

Input filters are available in hard filter and soft filter. Both filters are selected using FUN7. High-speed counters and catch inputs can use only the hard filter. Normal inputs I0 through I7 can use both hard and soft filters.

Filter Circuit Schematic



When hard filter is set at default value of 10, catch input and high-speed counter input values are shown below.

Minimum pulse width to accept catch input (ON pulse): Input I0 = 28 µsec, Inputs I1 to I7 = 37 µsec
Minimum pulse width to accept catch input (OFF pulse): Input I0 = 30 µsec, Inputs I1 to I7 = 120 µsec
High-speed counter input frequency: 10 kHz (HSC0 and HSC3), 5 kHz (HSC1 and HSC2)

Inputs I10 through I15 and all inputs I20 through I35 at the expansion station cannot use the hard filter and soft filter. Input signals to these inputs are filtered by fixed filter of 3.0 msec. Short-pulse inputs and noises shorter than 3.0 msec are ignored.

Note: Normal inputs I0 through I35 require 1 scan time in addition to the applicable hard, soft, or fixed filter value to accept input signals.

Setting Input Filter

Use FUN7 on the program loader to set the hard filter and soft filter values. See page 5-6.

Hard filter: 0 through 255 (default value is 10)
Input I0 ON pulse = 4 to 616 µsec, Input I0 OFF pulse = 6 to 618 µsec
Inputs I1 to I7 ON pulse = 20 to 625 µsec, Inputs I1 to I7 OFF pulse = 120 to 618 µsec

Soft filter: 0, 3, 7, or 10 msec (default value is 3 msec)

Filtering Operation

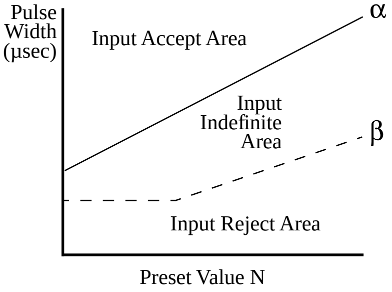
Depending on the selected values, the hard and soft filters have three response areas to receive or ignore input signals.

Input reject area:	Input signals are ignored and not received definitely.
Input indefinite area:	Input signals may be received or ignored.
Input accept area:	Input signals are received definitely.

Hard Filter

Three input response areas are calculated for preset value N from the following formula. Use these values for reference only.

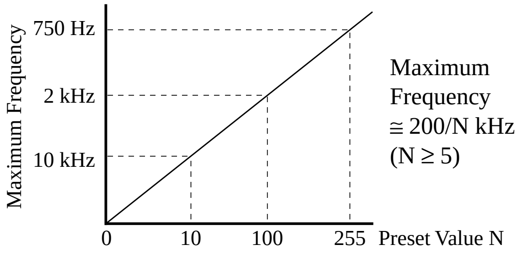
Input Signal		Input Accept Area α	Input Reject Area β
I0	ON Pulse	$\alpha > 2.4N + 4$ (Equation A)	$N \leq 1$: $\beta < 1$ $N > 1$: $\beta < 0.8N - 1$ (Equation B)
I1 to I7	ON Pulse	$N \leq 2$: $\alpha > 20$ $N > 2$: $\alpha > 2.4N + 13$	$N \leq 138$: $\beta < 8$ $N > 138$: $\beta < 0.8N - 103$
I0	OFF Pulse	$\alpha > 2.4N + 6$	$N \leq 2$: $\beta < 1$ $N > 2$: $\beta < 0.8N - 1$
I1 to I7	OFF Pulse	$N \leq 47$: $\alpha > 120$ $N > 47$: $\alpha > 2.4N + 6$	$\beta < 0.8N + 40$



Input response areas vary with input signals and hard filter preset values as listed below

Input Signal Example	Hard Filter Preset	Input Reject, Accept, or Indefinite Area
High-speed counter 10 kHz A/D conversion	10	I0 7 µsec 28 µsec Reject Indefinite Accept I1 to I7 8 µsec 37 µsec
Catch input 40-µsec ON pulse		
Catch input 110-µsec ON pulse	40	I0 31 µsec 100 µsec Reject Indefinite Accept I1 to I7 8 µsec 109 µsec
Catch input 260-µsec ON pulse	100	I0 79 µsec 244 µsec Reject Indefinite Accept I1 to I7 8 µsec 253 µsec
Catch input 500-µsec ON pulse	200	I0 159 µsec 484 µsec Reject Indefinite Accept I1 to I7 57 µsec 493 µsec
Catch input 200-µsec OFF pulse	80	I0 63 µsec 198 µsec Reject Indefinite Accept I1 to I7 104 µsec 198 µsec

If the hard filter is set to a value smaller than required, MICRO^3 becomes susceptible to noises and malfunctions may occur frequently. As the hard filter is set to a larger value, the maximum operating frequency of high-speed counters will decrease. The relationship between the hard filter setting and maximum operating frequency is shown on the right. When high-speed response is required in an environment where noise exists, use shielded wires for input signal lines.



Example: Receiving Minimum Pulse Width of 150 µsec

When input I0 is required to receive short pulses of 150 µsec minimum using the catch input function, the preset value N for the hard filter is calculated as follows. From Equation A on page 4-4,

$$150 = 2.4N + 4$$

$$N = 60.8$$

Set the hard filter preset value N to 60 or less to catch short input ON pulses of 150 µsec.

Example: Eliminating Input Signals of 150 µsec

When input I0 is required to eliminate noise signals of 150 µsec or less using the catch input function, the preset value N for the hard filter is calculated as follows. From Equation B on page 4-4,

$$150 = 0.8N – 1$$

$$N = 188.75$$

Set the hard filter preset value N to 189 or more to eliminate input pulses of 150 µsec. When N is set to 189, the minimum input signal width that can be received is calculated as follows. From Equation A on page 4-4,

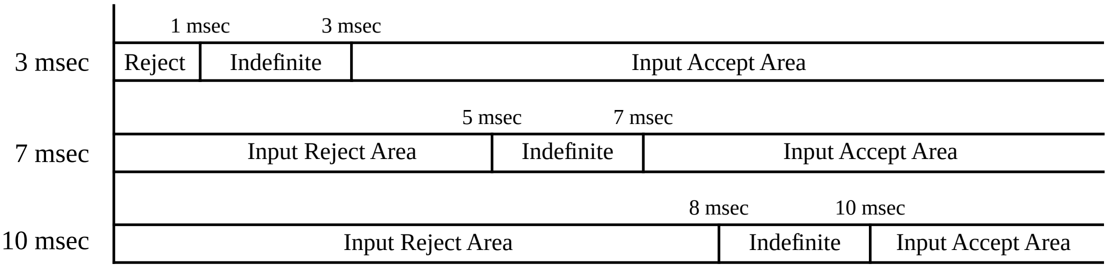
$$2.4 \times 189 + 4 = 457.6 \text{ µsec}$$

Soft Filter

The soft filter can be set to 0 msec, 3 msec, 7 msec, or 10 msec for normal inputs I0 through I7 in four groups, using FUN7 on the program loader. See page 5-6.

When the soft filter is set to 0 msec, the filtering function depends on the hard filter.

When the soft filter is set to 3 msec, 7 msec, or 10 msec, the soft filter is enabled to filter input signals. The input accept or reject areas for each setting are shown below.



Normal inputs require pulse widths of the above value plus 1 scan time to read the input signal.

The soft filter can be set in four groups of inputs.

Soft Group	Input Number
G1	I0
G2	I1
G3	I2, I3
G4	I4, I5, I6, I7

On the 10-I/O type MICRO^3 base unit, only inputs I4 and I5 are available for group G4.

Pulse Output Function

MICRO^3 features a pulse output function which can be used for illumination control and pulse-driven machines such as machine tools and conveyors. For details, see the PULS (pulse output) instruction on page 16-1 and PWM (pulse width modulation) instruction on page 16-3.

High-speed Counter Function

MICRO^3 features high-speed counter functions which can be used for position control by counting high-speed pulses or for simple motor control in combination with the pulse output. For details, see page 17-1.

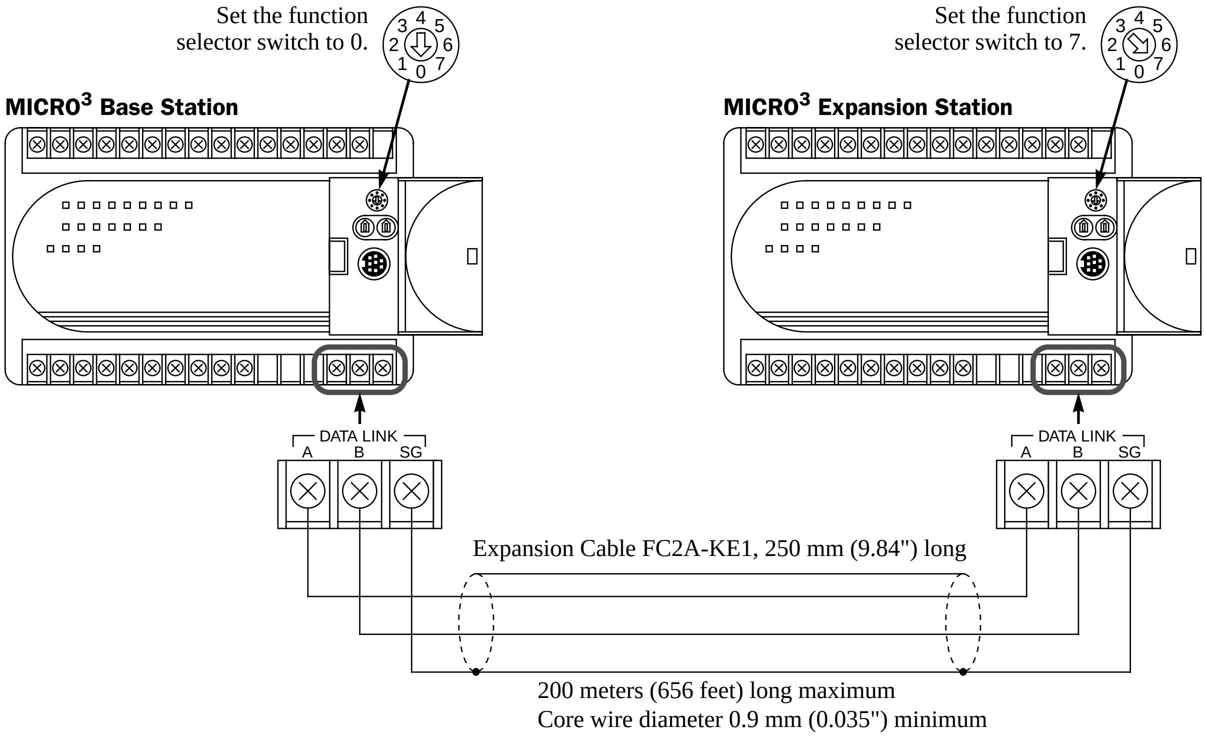
Expansion Link Function

I/O points can be expanded by connecting another MICRO^3 base unit using a shielded 2-core twisted cable. Only one unit can be added to expand I/O points from 10, 16, or 20 points up to 48 points. The expansion link function cannot be used with the data link function or in the high-speed processing mode.

Expansion Link System Setup

To set up an expansion link system, connect the data link terminals of both units using an expansion cable FC2A-KE1 (250mm or 9.84" long) or a shielded twisted pair cable with a minimum core wire diameter of 0.9 mm (0.035") as shown below. The cable for the expansion link system can be extended up to 200 meters (656 feet).

Set the function selector switch to 0 at the base station and to 7 at the expansion station.


Set the function selector switch to 0.
3 4 5 2 6 1 0 7
Set the function selector switch to 7.
3 4 5 2 6 1 0 7
MICRO3 Base Station
MICRO3 Expansion Station
DATA LINK
A B SG
DATA LINK
A B SG
Expansion Cable FC2A-KE1, 250 mm (9.84") long
200 meters (656 feet) long maximum
Core wire diameter 0.9 mm (0.035") minimum

Operating Procedure for Expansion Link System

Power up both MICRO^3 base units at the same time or power up the expansion station first. If the expansion station is powered up later than the base station, the base station does not recognize the expansion station. To recognize the expansion station in this case, execute FUN27 Link Formatting Sequence at the base station (see page 5-11) or turn on M307 Link Communication Initialize Flag at the master station (see page 6-3).

The scan time is extended by approximately 10 msec in the expansion link system.

If any communication error occurs in the expansion link system, communication error codes can be set to control data register D94. For details of link communication error codes, see page 18-5. To enable the control data register, use FUN10 Control Data Register Setting. See page 5-8. If a communication error occurs, the data is resent three times. If the error still exists after three attempts, the error code is set to data register D94.

The program loader can be connected to the base station only. If the program loader is connected to the expansion station, an error will result and error message “Expansion Unit” is displayed on the program loader.

The RUN indicator on the expansion station remains off whether the base station is running or stopped.

I/O Allocation Numbers for Expansion Link System

Input and output allocation numbers do not continue from the base station to the expansion station. At the expansion station, inputs start at I20 and outputs start at Q20. Inputs and outputs are allocated depending on the MICRO^{3} base units used in the expansion link system as shown below:

<table>
<tr><th colspan="2">I/O Points</th><th colspan="2" rowspan="2">MICRO^{3} Base Station
I/O Allocation Numbers</th><th colspan="2" rowspan="2">MICRO^{3} Expansion Station
I/O Allocation Numbers</th></tr>
<tr><th>Total</th><th>IN/OUT</th></tr>
<tr><td rowspan="2">10</td><td rowspan="2">6/4</td><td colspan="2">10-I/O Type</td><td colspan="2" rowspan="2">———</td></tr>
<tr><td>I0 - I5</td><td>Q0 - Q3</td></tr>
<tr><td rowspan="2">16</td><td rowspan="2">9/7</td><td colspan="2">16-I/O Type or AC input Type</td><td colspan="2" rowspan="2">———</td></tr>
<tr><td>I0 - I7
I10</td><td>Q0 - Q6</td></tr>
<tr><td rowspan="2">20</td><td rowspan="2">12/8</td><td colspan="2">10-I/O Type</td><td colspan="2">10-I/O Type</td></tr>
<tr><td>I0 - I5</td><td>Q0 - Q3</td><td>I20 - I25</td><td>Q20 - Q23</td></tr>
<tr><td rowspan="2">24</td><td rowspan="2">14/10</td><td colspan="2">24-I/O Type</td><td colspan="2" rowspan="2">———</td></tr>
<tr><td>I0 - I7
I10 - I15</td><td>Q0 - Q7
Q10 - Q11</td></tr>
<tr><td rowspan="4">26</td><td rowspan="4">15/11</td><td colspan="2">10-I/O Type</td><td colspan="2">16-I/O Type or AC input Type</td></tr>
<tr><td>I0 - I5</td><td>Q0 - Q3</td><td>I20 - I27
I30</td><td>Q20 - Q26</td></tr>
<tr><td colspan="2">16-I/O Type or AC input Type</td><td colspan="2">10-I/O Type</td></tr>
<tr><td>I0 - I7
I10</td><td>Q0 - Q6</td><td>I20 - I25</td><td>Q20 - Q23</td></tr>
<tr><td rowspan="2">32</td><td rowspan="2">18/14</td><td colspan="2">16-I/O Type or AC input Type</td><td colspan="2">16-I/O Type or AC input Type</td></tr>
<tr><td>I0 - I7
I10</td><td>Q0 - Q6</td><td>I20 - I27
I30</td><td>Q20 - Q26</td></tr>
<tr><td rowspan="4">34</td><td rowspan="4">20/14</td><td colspan="2">10-I/O Type</td><td colspan="2">24-I/O Type</td></tr>
<tr><td>I0 - I5</td><td>Q0 - Q3</td><td>I20 - I27
I30 - I35</td><td>Q20 - Q27
Q30 - Q31</td></tr>
<tr><td colspan="2">24-I/O Type</td><td colspan="2">10-I/O Type</td></tr>
<tr><td>I0 - I7
I10 - I15</td><td>Q0 - Q7
Q10 - Q11</td><td>I20 - I25</td><td>Q20 - Q23</td></tr>
<tr><td rowspan="4">40</td><td rowspan="4">23/17</td><td colspan="2">16-I/O Type or AC input Type</td><td colspan="2">24-I/O Type</td></tr>
<tr><td>I0 - I7
I10</td><td>Q0 - Q6</td><td>I20 - I27
I30 - I35</td><td>Q20 - Q27
Q30 - Q31</td></tr>
<tr><td colspan="2">24-I/O Type</td><td colspan="2">16-I/O Type or AC input Type</td></tr>
<tr><td>I0 - I7
I10 - I15</td><td>Q0 - Q7
Q10 - Q11</td><td>I20 - I27
I30</td><td>Q20 - Q26</td></tr>
<tr><td rowspan="2">48</td><td rowspan="2">28/20</td><td colspan="2">24-I/O Type</td><td colspan="2">24-I/O Type</td></tr>
<tr><td>I0 - I7
I10 - I15</td><td>Q0 - Q7
Q10 - Q11</td><td>I20 - I27
I30 - I35</td><td>Q20 - Q27
Q30 - Q31</td></tr>
</table>

Other allocation numbers for the expansion system are the same as the basic system. For other allocation numbers, see page 6-1.

Data Link Function

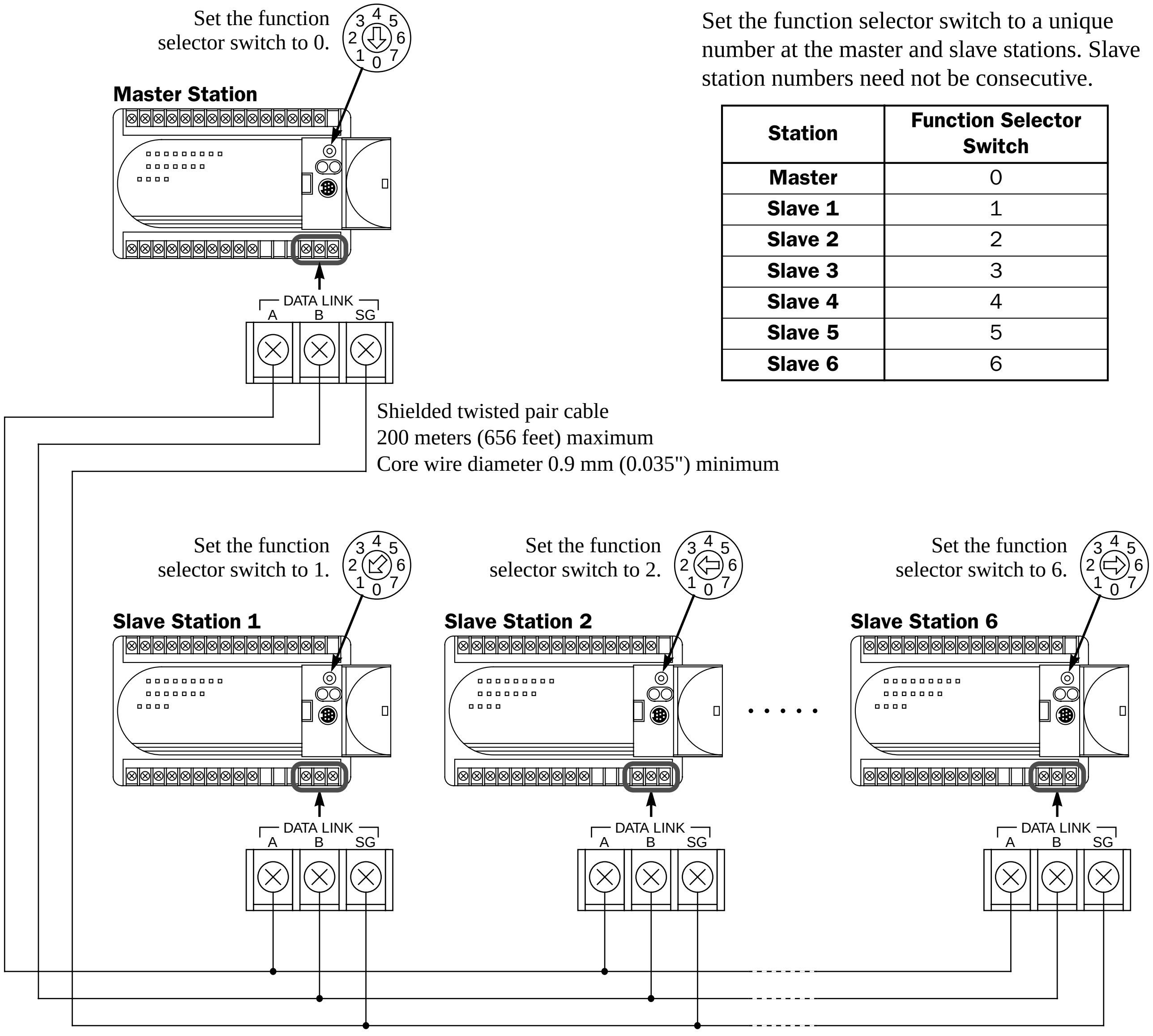
MICRO^3 features the data link function to set up a distributed control system. A maximum of six slave stations can be connected to the master station. Data of inputs, outputs, internal relays, timers, counters, shift registers, and data registers are communicated between the master and slave stations. The master station has five data registers assigned for each slave station. Each slave station has five data registers assigned for communication with the master station. When data is set in a data register at the master station assigned for data link communication, the data is sent to the corresponding data register at a slave station. When data is set in a data register at the slave station assigned for data link communication, the data is sent to the corresponding data register at the master station. Therefore, any particular program is not required for sending or receiving data in the data link communication system.

The data link function cannot be used with the expansion link function or in the high-speed processing mode. When a slave station performs communication at 19,200 bps through the loader port, multi-stage comparison instruction HSC1 cannot be used at the slave station.

MICRO^3 can also be connected to FA-3S series serial interface module PF3S-SIF4 mounted with high-performance CPU module PF3S-CP12 or PF3S-CP13. Since two serial interface modules can be mounted with one CPU, a maximum of 12 MICRO^3 base units can be connected to the FA-3S master station in the data link system. For details, see page 4-13.

Data Link System Setup

To set up a data link system, connect the data link terminals of every unit using a shielded twisted pair cable as shown below. The total length of the cable for the data link system can be extended up to 200 meters (656 feet).



Set the function selector switch to 0.

Set the function selector switch to 1.

Set the function selector switch to 2.

Set the function selector switch to 6.

Shielded twisted pair cable
200 meters (656 feet) maximum
Core wire diameter 0.9 mm (0.035") minimum

Set the function selector switch to a unique number at the master and slave stations. Slave station numbers need not be consecutive.

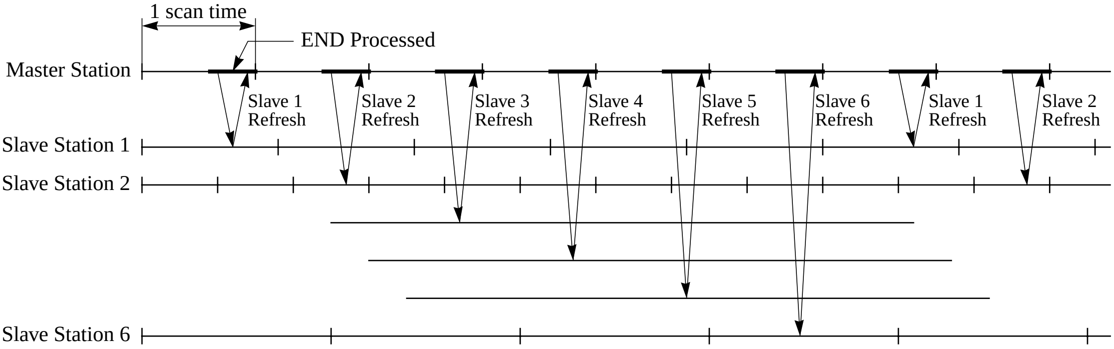
Station	Function Selector Switch
Master	0
Slave 1	1
Slave 2	2
Slave 3	3
Slave 4	4
Slave 5	5
Slave 6	6

Data Link Specifications

Electric Specifications	Compliance with EIA-RS485
Baud Rate	19200 bps (fixed)
Maximum Cable Length	200m (656 feet) total
Maximum Slave Stations	6 slave stations

Communication Sequence

Only one slave station can communicate with the master station in one scan. When a slave station receives a command from the master station, the slave station returns a response of processing results. When six slave stations are connected, six scans are required to communicate with all slave stations.



Data Register Allocation for Data Link System

Slave Station 1

D85	Link Communication Error
D86	Transmission Data H
D87	Transmission Data L
D88	Receive Data H
D89	Receive Data L

Master Station

D60	Link Communication Error
D61	Transmission Data H
D62	Transmission Data L
D63	Receive Data H
D64	Receive Data L
D65	Link Communication Error
D66	Transmission Data H
D67	Transmission Data L
D68	Receive Data H
D69	Receive Data L
D70	Link Communication Error
D71	Transmission Data H
D72	Transmission Data L
D73	Receive Data H
D74	Receive Data L
D75	Link Communication Error
D76	Transmission Data H
D77	Transmission Data L
D78	Receive Data H
D79	Receive Data L
D80	Link Communication Error
D81	Transmission Data H
D82	Transmission Data L
D83	Receive Data H
D84	Receive Data L
D85	Link Communication Error
D86	Transmission Data H
D87	Transmission Data L
D88	Receive Data H
D89	Receive Data L

Slave Station 2

D85	Link Communication Error
D86	Transmission Data H
D87	Transmission Data L
D88	Receive Data H
D89	Receive Data L

Slave Station 3

D85	Link Communication Error
D86	Transmission Data H
D87	Transmission Data L
D88	Receive Data H
D89	Receive Data L

Slave Station 4

D85	Link Communication Error
D86	Transmission Data H
D87	Transmission Data L
D88	Receive Data H
D89	Receive Data L

Slave Station 5

D85	Link Communication Error
D86	Transmission Data H
D87	Transmission Data L
D88	Receive Data H
D89	Receive Data L

Slave Station 6

D85	Link Communication Error
D86	Transmission Data H
D87	Transmission Data L
D88	Receive Data H
D89	Receive Data L

Slave station data registers D60 through D84 can be used as ordinary data registers.

If any slave stations are not connected, master station data registers which are assigned to the vacant slave stations can be used as ordinary data registers.

Operating Procedure for Data Link System

To set up and use a data link system, complete the following steps:

First determine the assignments for the master station and slave stations.
Connect MICRO^3 base units at the master station and all slave stations as illustrated on page 4-8.
Set the function selector switch to 0 on the MICRO^3 base unit at the master station and to 1 through 6 at slave stations.
Create user programs for the master and slave stations. Different programs are used for the master and slave stations.
Power up every MICRO^3 base unit at the same time, and transfer the user programs to the master and slave stations.
Monitor the data registers used for data link at the master and slave stations.

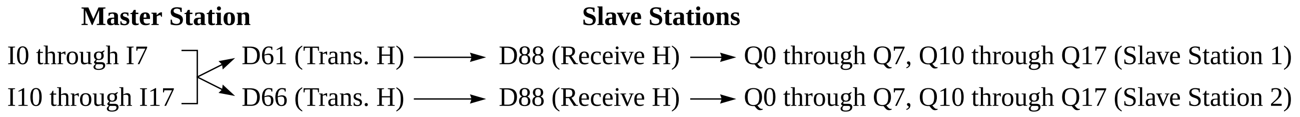
Note: To enable data link communication, power up every MICRO^3 base unit at the same time, or power up slave stations first. If a slave station is powered up later than the master station, the master station does not recognize the slave station. To recognize the slave station in this case, execute FUN27 Link Formatting Sequence at the master station (see page 5-11) or turn on M307 Link Communication Initialize Flag at the master station (see page 6-3).

The scan time is extended by approximately 12 msec in the data link system.

If any communication error occurs in the data link system, link communication error codes are set to data register D85 at the slave station and to a corresponding data register for link communication error at the master station. For details of link communication error codes, see page 18-5. To enable control data register D94 for link communication error code, use FUN10 Control Data Register Setting. See page 5-8. If a communication error occurs, the data is resent three times. If the error still exists after three attempts, then the error code is set to the data registers for link communication error. Since the error code is not communicated between the master and slave stations, error codes must be cleared individually.

Data Link Example 1: Data Transmission from Master Station

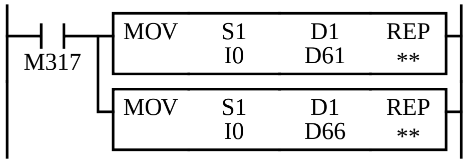
This example demonstrates data communication from the master station to two slave stations. Data of inputs I0 through I7 and I10 through I17 are set to data registers D61 (transmission data H for slave station 1) and D66 (transmission data H for slave station 2) at the master station. D61 data is sent to D88 (receive data H) of slave station 1, and D66 data is sent to D88 (receive data H) of slave station 2.



Function selector switch setting

Master station:	0
Slave station 1:	1
Slave station 2:	2

Master station program



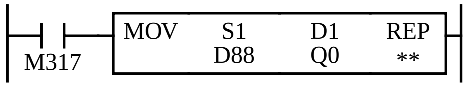
M317 is the in-operation output special internal relay which remains on during operation.

The first MOV (move) instruction sets 16 inputs I0 through I7 and I10 through I17 to data register D61 (transmission data H for slave station 1).

The second MOV (move) instruction sets 16 inputs I0 through I7 and I10 through I17 to data register D66 (transmission data H for slave station 2).

Slave station program

The same program is used for slave stations 1 and 2 in this example.

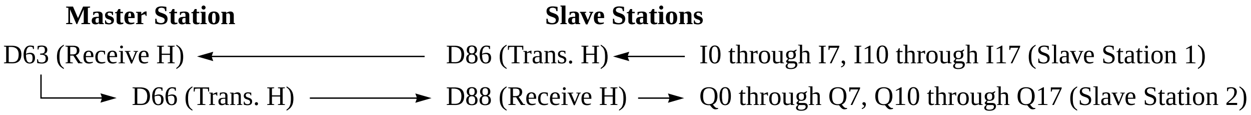


The MOV (move) instruction sets the data of data register D88 (receive data H) to 16 outputs Q0 through Q7 and Q10 through Q17.

Note: The MOV (move) instruction moves 16-bit word data from the source operand to the destination operand. Although 16-bit word data is processed internally, data cannot be read from or written to non-existent terminals When using the 24 I/O type MICRO^3 base unit which has 14 input terminals and 10 output terminals, data of only 14 input points I0 through I15 can be read to data register D61 and D66 at the master station and the upper two bits are set to zero in the data registers. Data of data register D88 can be taken out from only lower 10 output points Q0 through Q11 at the slave stations and the upper 6 outputs Q12 through Q17 cannot be taken out.

Data Link Example 2: Data Transmission from Slave Station

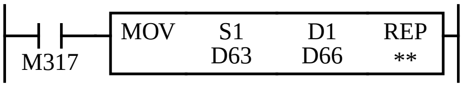
This sample program demonstrates data communication from slave station 1 to the master station, then to slave station 2. Data of inputs I0 through I7 and I10 through I17 are set to data register D86 (transmission data H) at slave station 1. The D86 data is sent to data register D63 (receive data H for slave station 1) of the master station. At the master station, D63 data is moved to data register D66 (transmission data H for slave station 2). The D66 data is sent to data register D88 (receive data H) of slave station 2, where the D88 data is set to outputs Q0 through Q7 and Q10 through Q17.


Master Station
Slave Stations
D63 (Receive H)
D86 (Trans. H)
I0 through I7, I10 through I17 (Slave Station 1)
D66 (Trans. H)
D88 (Receive H)
Q0 through Q7, Q10 through Q17 (Slave Station 2)

Function selector switch setting

Master station:	0
Slave station 1:	1
Slave station 2:	2

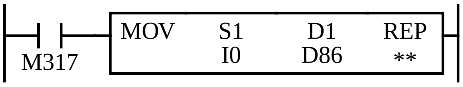
Master station program



M317 is the in-operation output special internal relay which remains on during operation.

The MOV (move) instruction sets the data of data register D63 (receive data H for slave station 1) to data register D66 (transmission data H for slave station 2).

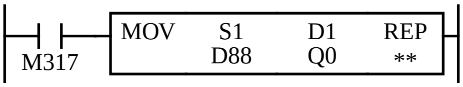
Slave station 1 program



M317 is the in-operation output special internal relay which remains on during operation.

The MOV (move) instruction sets 16 inputs I0 through I7 and I10 through I17 to data register D86 (transmission data H).

Slave station 2 program



M317 is the in-operation output special internal relay which remains on during operation.

The MOV (move) instruction sets the data of data register D88 (receive data H) to 16 outputs Q0 through Q7 and Q10 through Q17.

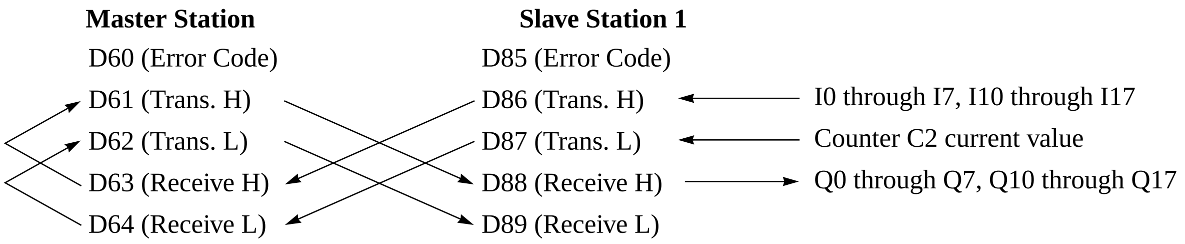
Data Link Example 3: Input and Counter Data Transmission

This sample program demonstrates a data link system to transmit input and counter data between the master station and 6 slave stations. At every slave station, data of inputs I0 through I7 and I10 through I17 are set to data register D86 (transmission data H). The D86 data from slave station 1 is sent to data register D63 (receive data H for slave station 1) of the master station. At the master station, the D63 data is moved to data register D61 (transmission data H for slave station 1). The D61 data is sent to data register D88 (receive data H) of slave station 1, where the D88 data is set to outputs Q0 through Q7 and Q10 through Q17.

In addition, counter C2 current value is set to data register D87 (transmission data L) at every slave station. The D87 data from slave station 1 is sent to data register D64 (transmission data L for slave station 1). At the master station, the D64 data is moved to data register D62 (transmission data L for slave station 1). The D62 data is sent to data register D89 (receive data L) of slave station 1.

Similarly, slave stations 2 through 6 also transmit and receive the same data to and from the corresponding data registers at the master station.

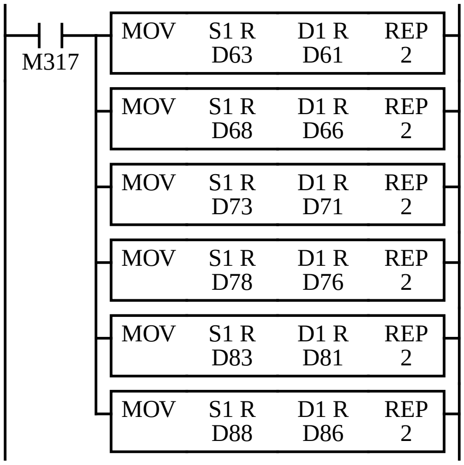
Data Link Example 3: Input and Counter Data Transmission, continued



Function selector switch setting

Master station:	0	Slave station 1:	1	Slave station 2:	2	Slave station 3:	3
Slave station 4:	4	Slave station 5:	5	Slave station 6:	6		

Master station program


M317
MOV S1 R D63 D1 R D61 REP 2
MOV S1 R D68 D1 R D66 REP 2
MOV S1 R D73 D1 R D71 REP 2
MOV S1 R D78 D1 R D76 REP 2
MOV S1 R D83 D1 R D81 REP 2
MOV S1 R D88 D1 R D86 REP 2

M317 is the in-operation output special internal relay which remains on during operation.

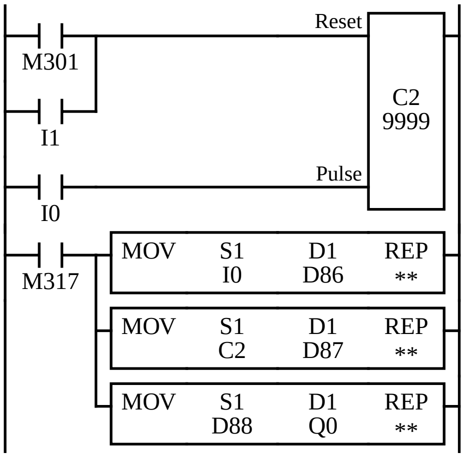
The first MOV (move) instruction with 2 repeat cycles sets the data of data registers D63 and D64 (receive data H and L for slave station 1) to data registers D61 and D62 (transmission data H and L for slave station 1), respectively.

Similarly, next 5 MOV instructions set data of 2 receive data registers to 2 transmission data registers for slave stations 2 through 6.

D63 and D64 → D61 and D62
D68 and D69 → D66 and D67
D73 and D74 → D71 and D72
D78 and D79 → D76 and D77
D83 and D84 → D81 and D82
D88 and D89 → D86 and D87

Slave station program

The same program is used for slave stations 1 through 6 in this example.



M301 is the initialize pulse special internal relay to reset counter C2 when starting operation.

Adding counter C2 counts input signals to input I0 and is reset when input I1 is turned on.

M317 is the in-operation output special internal relay which remains on during operation.

The first MOV (move) instruction sets 16 inputs I0 through I17 to data register D86 (transmission data H).

The second MOV instruction sets the counter C2 current value to data register D87 (transmission data L).

The last MOV instruction sets data of data register D88 (receive data H) to 16 outputs Q0 through Q17.

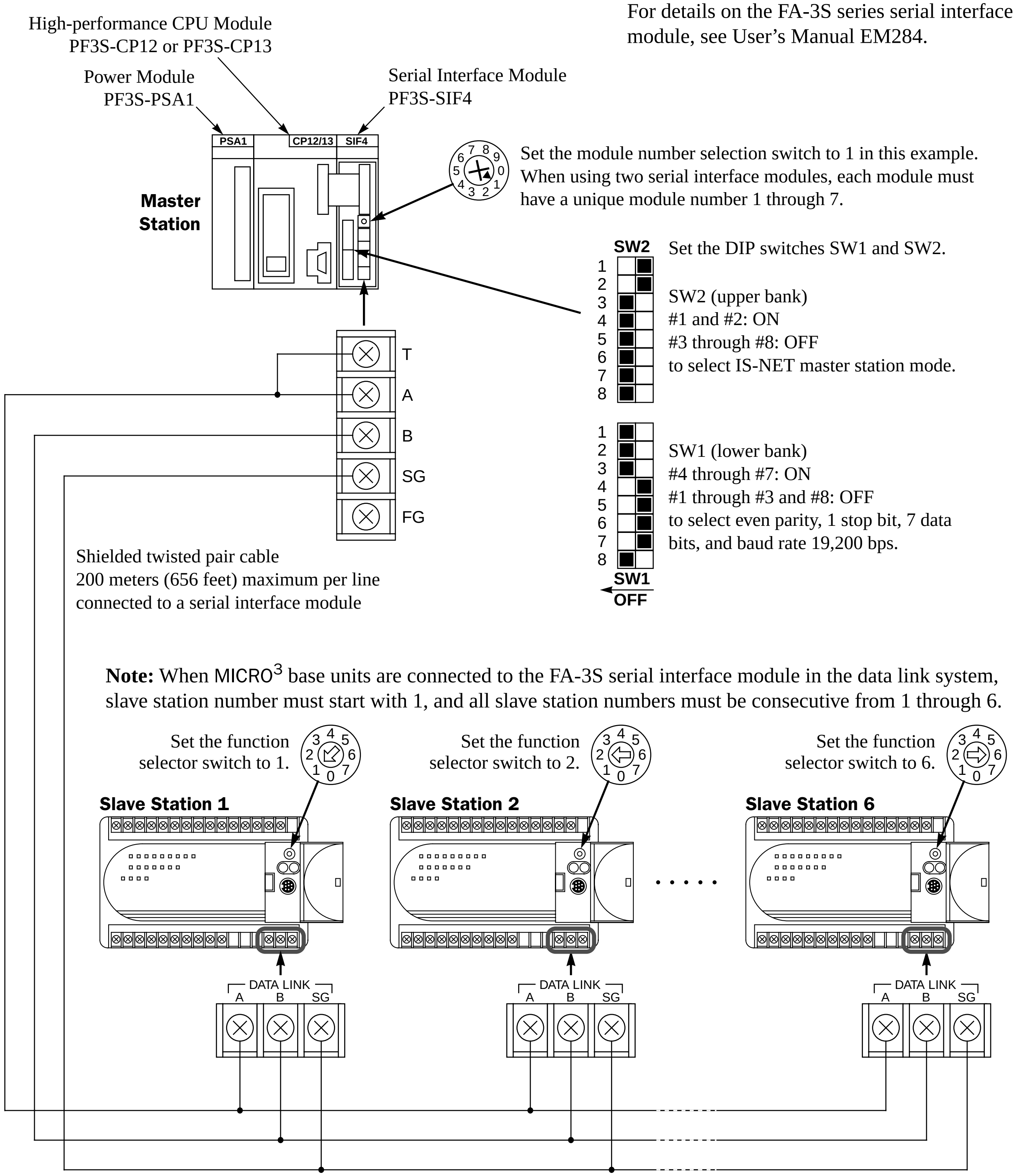
Data Link Example 4: Data Transmission through FA-3S Serial Interface Module

This sample program demonstrates data communication between the FA-3S master station and MICRO^3 slave stations using the PF3S-SIF4 serial interface module. FA-3S series high-performance CPU module PF3S-CP12 or PF3S-CP13 is used for the master station. The serial interface module mounted at the master station is set to operate in the IS-NET communication mode and the FA-3S series CPU module at the master station uses a universal mode master station program.

Six MICRO^3 slave stations can be connected to one serial interface module. Since two serial interface modules can be mounted with one FA-3S high-performance CPU, a maximum of 12 MICRO^3 base units can be connected to the FA-3S.

System Setup

For details on the FA-3S series serial interface module, see User's Manual EM284.



High-performance CPU Module
PF3S-CP12 or PF3S-CP13

Power Module
PF3S-PSA1

Serial Interface Module
PF3S-SIF4

Set the module number selection switch to 1 in this example. When using two serial interface modules, each module must have a unique module number 1 through 7.

Set the DIP switches SW1 and SW2.

SW2 (upper bank)
#1 and #2: ON
#3 through #8: OFF
to select IS-NET master station mode.

SW1 (lower bank)
#4 through #7: ON
#1 through #3 and #8: OFF
to select even parity, 1 stop bit, 7 data bits, and baud rate 19,200 bps.

Shielded twisted pair cable
200 meters (656 feet) maximum per line
connected to a serial interface module

Note: When MICRO^3 base units are connected to the FA-3S serial interface module in the data link system, slave station number must start with 1, and all slave station numbers must be consecutive from 1 through 6.

Set the function selector switch to 1.

Set the function selector switch to 2.

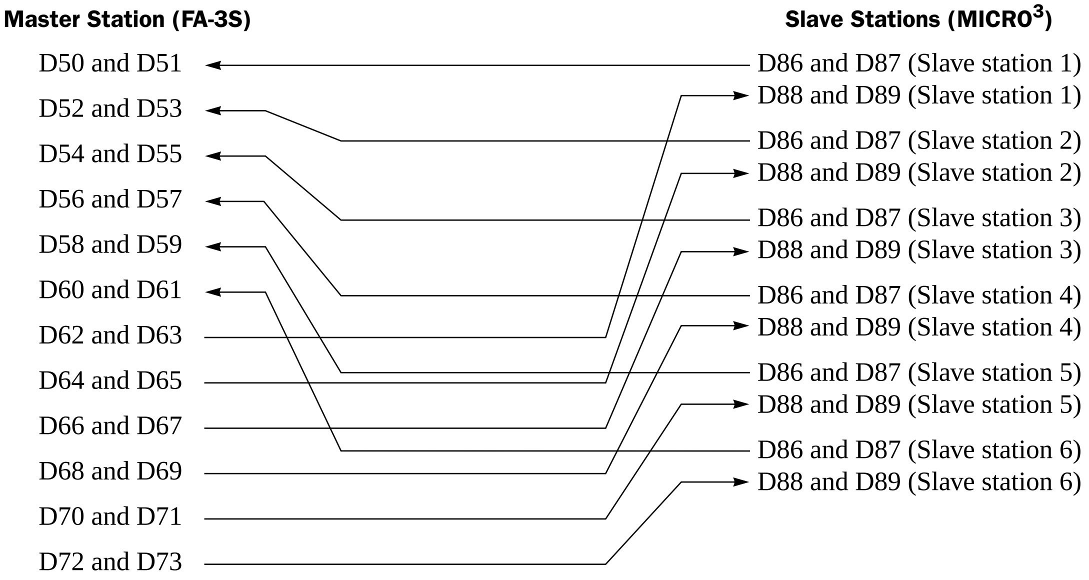
Set the function selector switch to 6.

Data Link Example 4: Data Transmission through FA-3S Serial Interface, continued

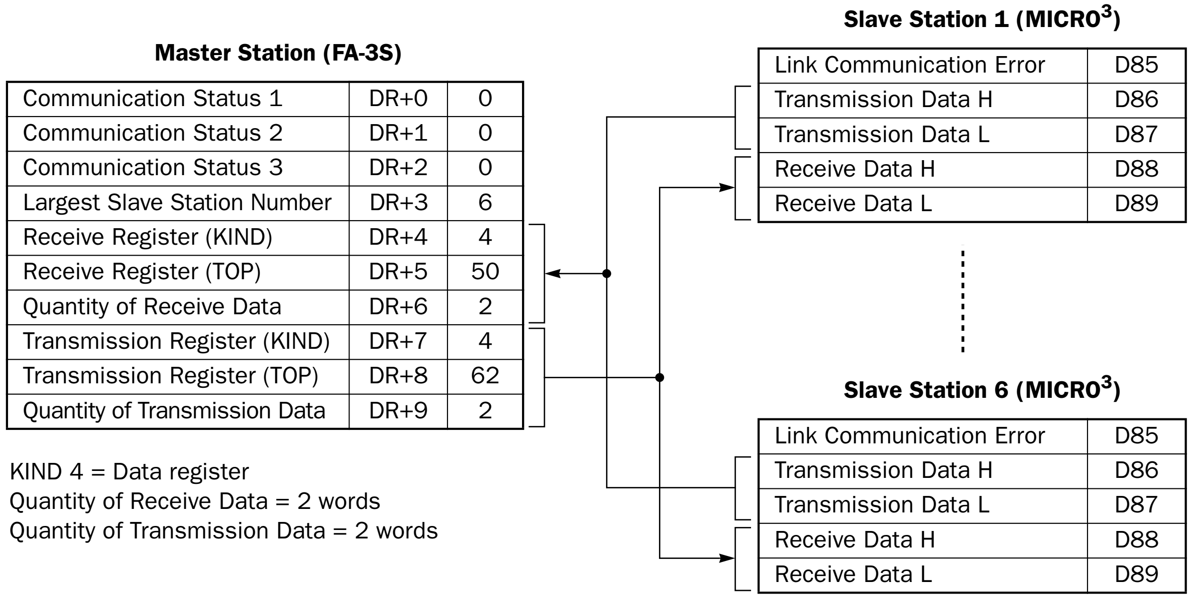
Data movement and LCOPR (local operand)

MICRO^3 has two data registers for transmission and two data registers for receiving in the data link system. So, each MICRO^3 slave station can communicate four words of data with the FA-3S master station. Since the quantity of data registers assigned to MICRO^3 is fixed, using the universal mode program is easier for the master station rather than using the individual mode program. MICRO^3 slave stations do not require any particular program for communication.

Using this sample program, data of data registers D86 and D87 (transmission data H and L) at slave station 1 is sent to data registers D50 and D51 at the master station. Data of the same data registers at slave station 2 is sent to the next two data registers at the master station, and so on. From the master station, data of D62 and D63 is sent to data registers D88 and D89 (receive data H and L) at slave station 1. Data of next two data registers is sent to the same data registers at slave station 2, and so on.


Master Station (FA-3S)
Slave Stations (MICRO3)
D50 and D51
D52 and D53
D54 and D55
D56 and D57
D58 and D59
D60 and D61
D62 and D63
D64 and D65
D66 and D67
D68 and D69
D70 and D71
D72 and D73
D86 and D87 (Slave station 1)
D88 and D89 (Slave station 1)
D86 and D87 (Slave station 2)
D88 and D89 (Slave station 2)
D86 and D87 (Slave station 3)
D88 and D89 (Slave station 3)
D86 and D87 (Slave station 4)
D88 and D89 (Slave station 4)
D86 and D87 (Slave station 5)
D88 and D89 (Slave station 5)
D86 and D87 (Slave station 6)
D88 and D89 (Slave station 6)

LCOPR (local operands) for the FNTWW instruction used at the master station are allocated as shown below:



Master Station (FA-3S)

Communication Status 1	DR+0	0
Communication Status 2	DR+1	0
Communication Status 3	DR+2	0
Largest Slave Station Number	DR+3	6
Receive Register (KIND)	DR+4	4
Receive Register (TOP)	DR+5	50
Quantity of Receive Data	DR+6	2
Transmission Register (KIND)	DR+7	4
Transmission Register (TOP)	DR+8	62
Quantity of Transmission Data	DR+9	2

KIND 4 = Data register
Quantity of Receive Data = 2 words
Quantity of Transmission Data = 2 words

Slave Station 1 (MICRO^3)

Link Communication Error	D85
Transmission Data H	D86
Transmission Data L	D87
Receive Data H	D88
Receive Data L	D89

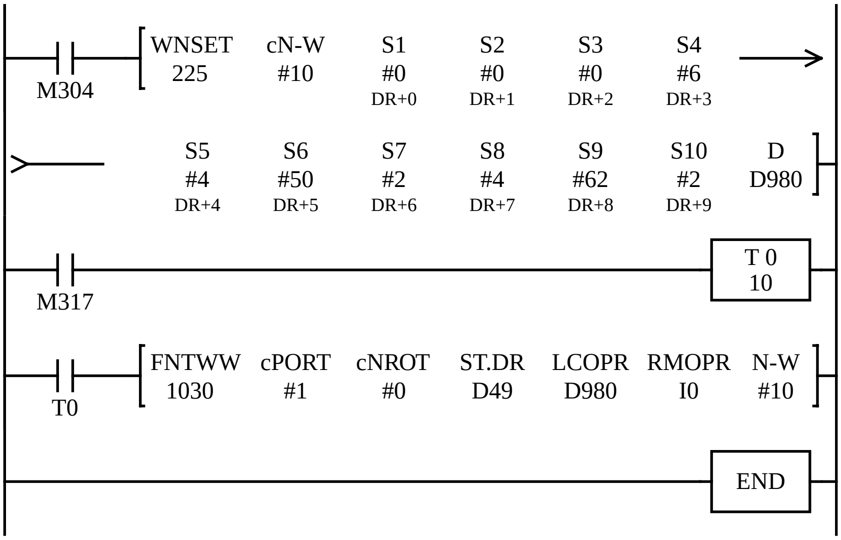
Slave Station 6 (MICRO^3)

Link Communication Error	D85
Transmission Data H	D86
Transmission Data L	D87
Receive Data H	D88
Receive Data L	D89

Note: At the master station, the first three data registers store communication status codes. At slave stations, data register D85 stores link communication error codes when any communication error occurs during data link communication. The data in these data registers are not communicated between the master and slave stations.

Data Link Example 4: Data Transmission through FA-3S Serial Interface, continued

Master station program for FA-3S high-performance CPU (IS-NET communication universal mode)



M304 is the initialize pulse special internal relay.

When the CPU starts, local operand data is set to 10 data registers starting with D980 (LCOPR).

M317 is the in-operation special internal relay which stays on while the CPU is running.

Timer T0 ensures 1-sec time delay before executing the format instruction for the master station.

cPORT #1 specifies module number 1. RMOPR I0 specifies the universal mode. N-W #10 specifies 10 words of data used for LCOPR in the universal mode.

Using this sample program, the FA-3S master station issues the formatting sequence 1 second after starting operation and confirms with which slave stations the master station can communicate. If any slave station is not powered up, then the slave station cannot be recognized. So, power up all slave stations at least 1 second before the master station starts to run. After slave stations are recognized, data is communicated between the master station and slave stations.

Computer Link Function

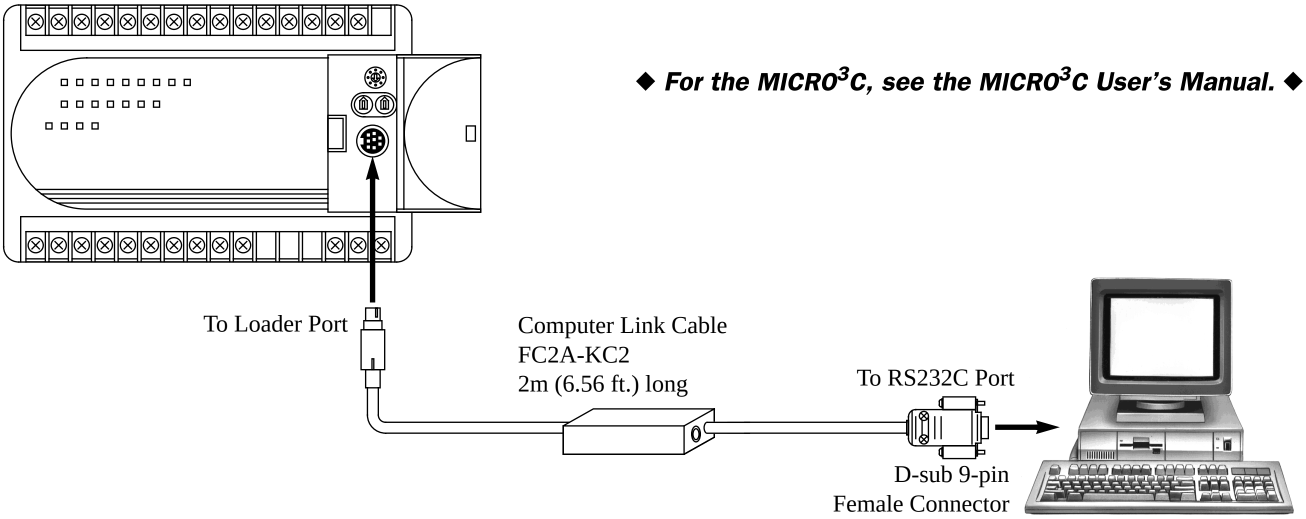
A personal computer can be connected to one MICRO^3 base unit in a peer-to-peer configuration (1:1 communication) or to a maximum of 32 MICRO^3 base units in a network configuration (1:N communication).

Using the optional software CUBIQ (FC9Y-LP1E314) on an IBM PC or compatible, user programs can be edited on the computer and transferred between the computer and MICRO^3. It is also possible to monitor from the computer the operation of the MICRO^3 system, current values of timers and counters, the data in data registers, and the statuses of inputs and outputs. MICRO^3 can be started or stopped from the computer. Preset values for timers and counters, and data in data registers can also be changed. Ladder diagrams, mnemonic lists, FUN tables, and labels can be printed out from the computer on a printer.

Computer Link 1:1 Communication

To set up a 1:1 computer link system, connect a computer to MICRO^3 using computer link cable FC2A-KC2.

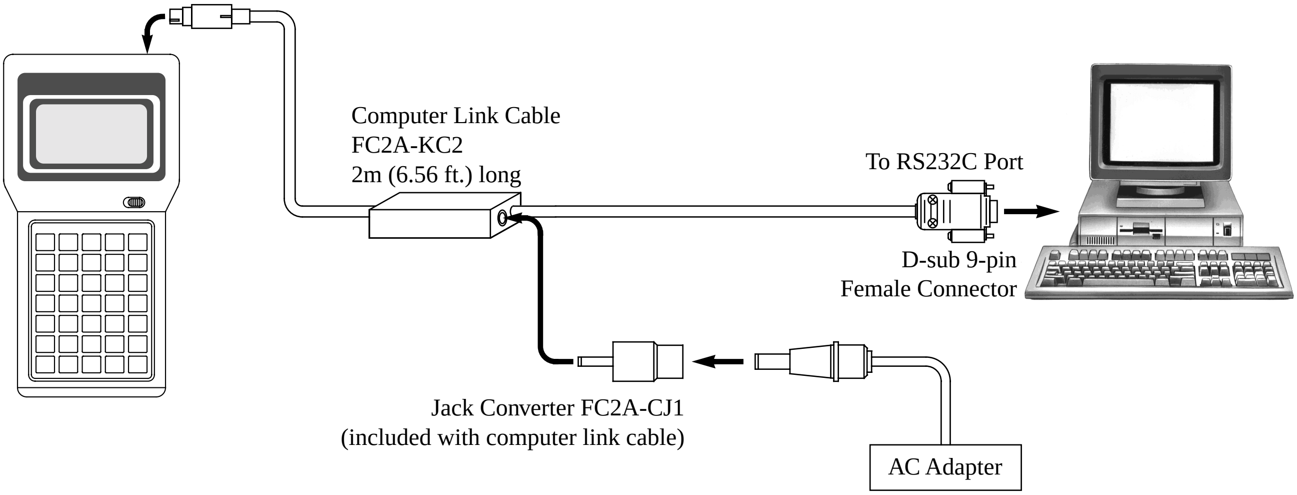
◆ For the MICRO^{3}C, see the MICRO^{3}C User's Manual. ◆



Use FUN8 Loader Port Communication Mode Setting to make sure that the communication parameters for the MICRO^3 loader port are the same as the computer connected. For FUN8, see page 5-7.

Communication between the program loader and computer

The program loader can also be connected to an IBM PC or compatible using computer link cable FC2A-KC2 for communication. An AC adapter is required to power the program loader. Connect the computer link cable to the loader cable connection port on the program loader. Plug the jack converter into the converter box on the computer link cable and plug the AC adapter into the jack converter. For specifications of an applicable AC adapter, see page A-4.

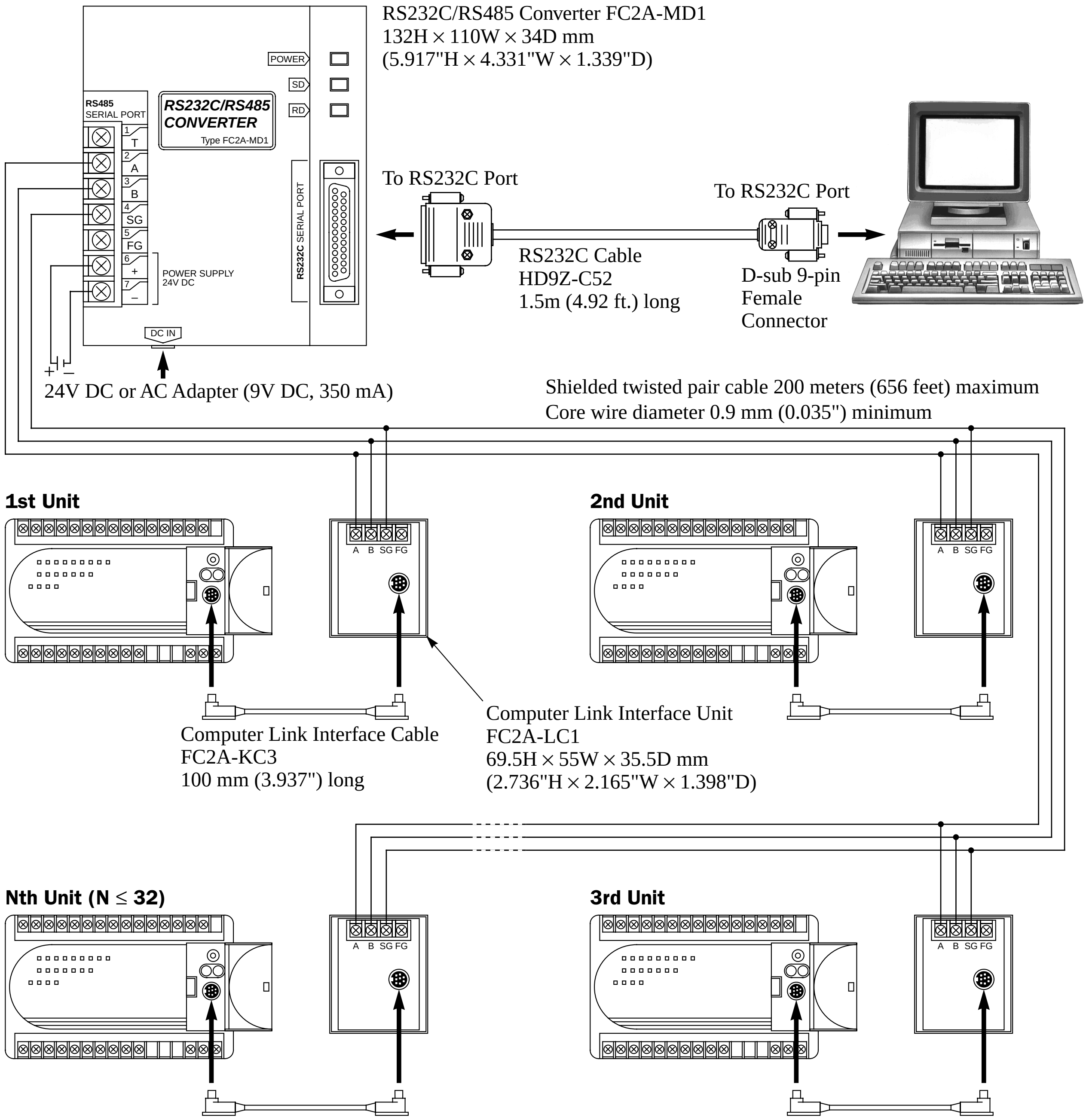


Computer Link 1:N Communication

To set up a 1:N computer link system, connect a computer to RS232C/RS485 converter using RS232C cable HD9Z-C52. Connect the RS232C/RS485 converter to computer link interface units FC2A-LC1 using shielded twisted pair cables. Connect MICRO^3 to each computer link interface unit using computer link interface cable FC2A-KC3.

Supply power to the RS232C/RS485 converter by connecting a 24V DC source to terminals 6 and 7 or by plugging an AC adapter to the DC IN jack. For specifications of the AC adapter, see page A-4.

◆ ***For the MICRO^{3}C, see the MICRO^{3}C User's Manual.*** ◆



RS232C/RS485 Converter FC2A-MD1
132H × 110W × 34D mm
(5.917"H × 4.331"W × 1.339"D)

To RS232C Port

To RS232C Port

RS232C Cable
HD9Z-C52
1.5m (4.92 ft.) long

D-sub 9-pin
Female
Connector

24V DC or AC Adapter (9V DC, 350 mA)

Shielded twisted pair cable 200 meters (656 feet) maximum
Core wire diameter 0.9 mm (0.035") minimum

Computer Link Interface Cable
FC2A-KC3
100 mm (3.937") long

Computer Link Interface Unit
FC2A-LC1
69.5H × 55W × 35.5D mm
(2.736"H × 2.165"W × 1.398"D)

Use FUN8 Loader Port Communication Mode Setting to make sure that the communication parameters for the MICRO^3 loader port are the same as the computer connected. For FUN8, see page 5-7.

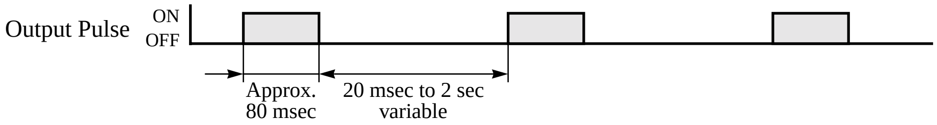
Select a unique PLC address number from 0 through 31 for each MICRO^3 using FUN9 PLC Address for Network Communication on the program loader and transfer the user program to MICRO^3. For FUN9, see page 5-7.

External Analog Timer

Analog timer unit PFA-1U11 can be connected to MICRO^3 to be used as an external analog timer and the preset value can be adjusted in very small increments from the panel front.

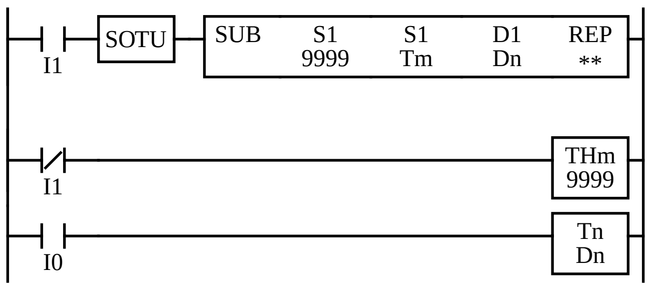
Analog Timer Unit PFA-1U11

The analog timer unit generates output pulses of approximately 80 msec and the interval can be changed between 20 msec and 2 sec using the knob on the analog timer unit.



Measuring Analog Timer OFF Duration

Connect the output of the analog timer unit to an input terminal of MICRO^3 and measure the OFF duration of the analog timer output using a timer instruction. The OFF duration can be varied using the knob on the analog timer unit. The resolution of the timer preset value depends on the first timer instruction TIM (100-msec timer), TMH (10-msec timer), or TMS (1-msec timer) used for measuring the OFF duration. Use the measured OFF duration as a preset value of another timer instruction.



I1: Pulse output from the external analog timer unit
Tm: Current value of timer TMHm
Dn: Preset value for the analog timer Tn

When the output pulse of the external analog timer unit is turned on (input I1 is turned on), timer current value Tm is subtracted from 9999 and the result is set to data register Dn. Dn is used as a preset value of timer instruction Tn (analog timer).

When the output pulse is off (input I1 is off), the 10-msec timer TMH times down from 9999 to measure the OFF duration of the external analog timer unit output.

When input I0 is on, the analog timer Tn starts to time down from the preset value Dn.

The preset value of the timer instruction for measuring the OFF duration must be the same as operand S1 of the SUB (subtraction) instruction. The first line and the second line of the program above must be in this order. If reversed, the measured duration cannot be set to the preset value for the analog timer Tn correctly.

The value of Dn is set to the preset value of the analog timer approximately 2 seconds after the program is started or the knob setting on the analog timer unit is changed.

Preset value Dn for the analog timer depends on the resolution of the timer instruction used for measuring the OFF duration. The maximum preset value also varies with the input filter preset value of MICRO^3 and ambient temperature.

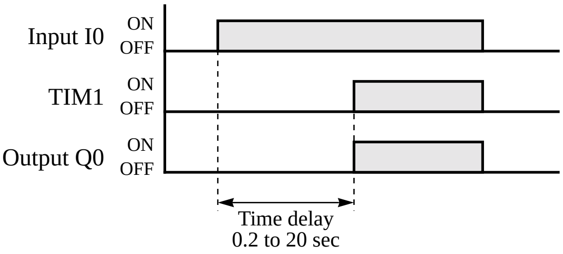
Depending on the combination of timer instructions used for measuring the OFF duration and for time-delay operation, available time delay ranges are shown below.

Timer instruction for measuring OFF duration	Measurement resolution	Preset value Dn for analog timer	Time delay range of analog timer		
			TIM	TMH	TMS
TIM	100 msec	1 to 20 (Note)	0.1 to 2 sec	10 to 200 msec	1 to 20 msec
TMH	10 msec	2 to 200	0.2 to 20 sec	20 msec to 2 sec	2 to 200 msec
TMS	1 msec	20 to 2000	2 to 200 sec	0.2 to 20 sec	20 msec to 2 sec

Note: When the TIM (100-msec timer) instruction is used for measuring the OFF duration, do not set the control knob of the analog timer unit to the minimum, because the TIM instruction cannot measure the minimum OFF duration of 20 msec. Set the control knob to make sure that the TIM instruction can measure the output pulse OFF duration.

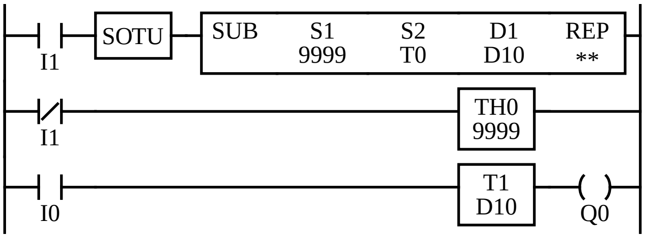
Example: ON-delay Analog Timer

This example demonstrates a program to vary the timer preset value for the TIM instruction between 0.2 and 20 sec using the TMH instruction for measuring the output pulse OFF duration of the external analog timer unit.



I0:	Start input for TIM1
I1:	Pulse output from the external analog timer unit
Q0:	Timer output
D10:	Preset value for TIM1
TMH0:	10-msec timer used for measuring the OFF duration of the pulse output from the external analog timer unit
TIM1:	100-msec instruction used for time-delay

Ladder Diagram



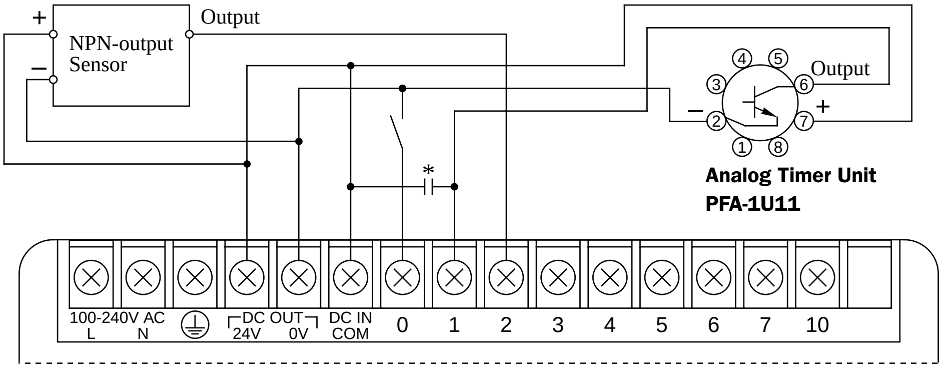
When the output pulse of the analog timer unit is turned on, the timer TMH0 current value is subtracted from 9999, and the result is set to data register D10, which is used as a preset value for 100-msec timer TIM1.

When the output pulse is off, 10-msec timer TMH0 times down from 9999 to measure the OFF duration of the external analog timer unit output.

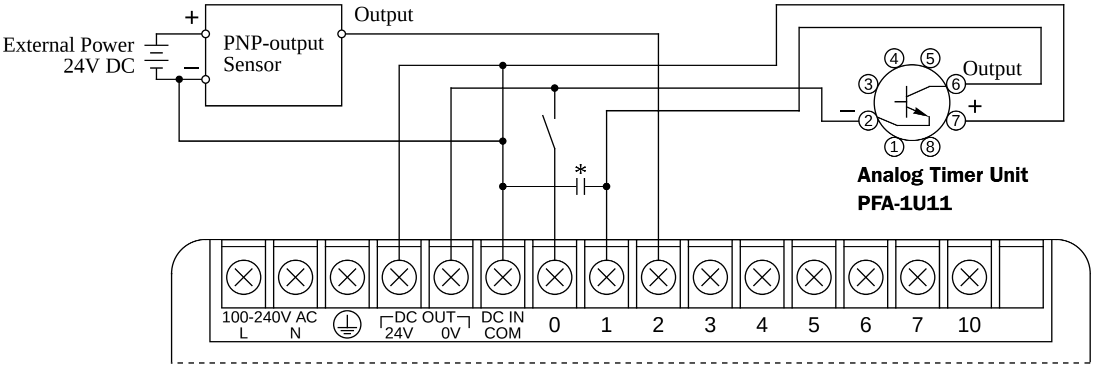
When I0 is turned on timer TIM1 starts to time down from preset value D10. When TIM1 times out, Q0 is turned on.

Wiring Diagram

- **When using with NPN-output sensors**



- **When using with PNP-output sensors**



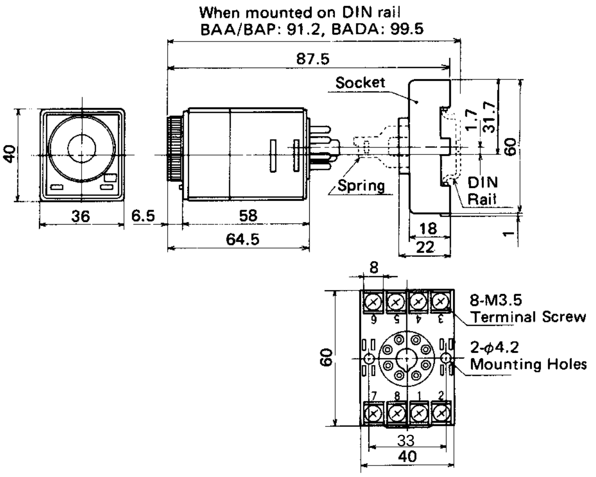
* When using analog timer in an environment subject to noise or when using long wires for connecting the analog timer, connect a capacitor of 1 µF/50V between the DC IN COM terminal and the input terminal connected to the analog timer output.

Analog Timer Unit and Accessories

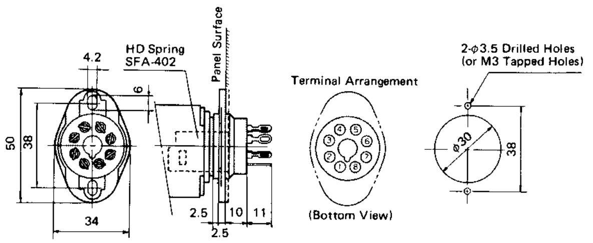
Name	Type No.	Remarks
Analog Timer Unit	PFA-1U11	For changing the preset value of timer instructions
DIN Rail Mount Socket	SR2P-06U	With screw terminals
Panel Mount Socket	SR2P-511	With solder terminals
	SR2P-70	With wire wrap terminals
Panel Mount Adapter	RTB-C01	Bluish gray
Wiring Socket Adapter	SR6P-S08	With solder terminals
	SR6P-M08G	With screw terminals
Hold-down Spring	SFA-202	For SR2P-06U
	SFA-402	For SR2P-511 and SR2P-70
DIN Rail	BAA1000	35mm-wide DIN rail, 1m long, made of aluminum
	BAP1000	35mm-wide DIN rail, 1m long, made of steel
	BADA1000	35mm-wide DIN rail, 1m long, made of aluminum

Analog Timer Unit Dimensions

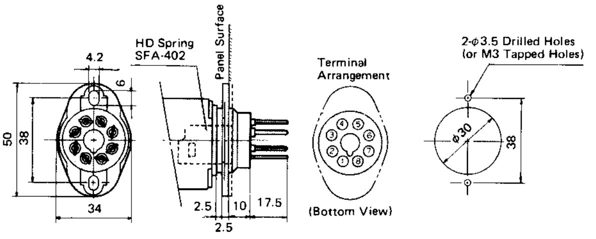
When using DIN Rail Mount Socket SR2P-06U



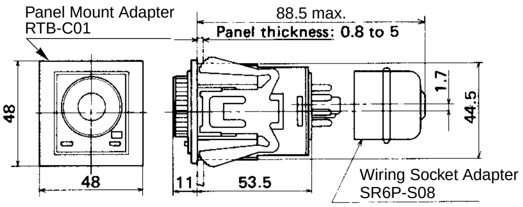
When using Panel Mount Socket SR2P-511



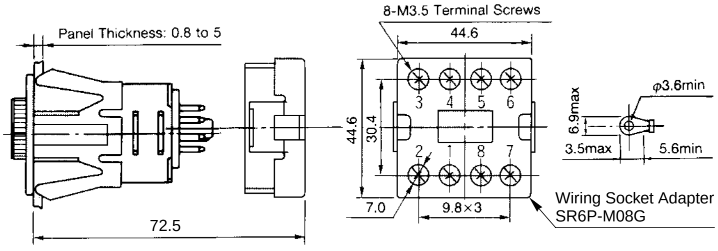
When using Panel Mount Socket SR2P-70



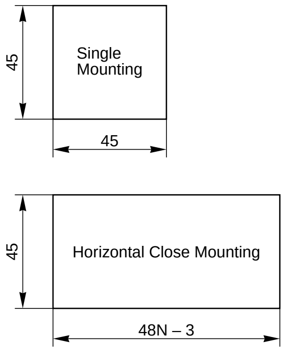
When using Panel Mount Adapter RTB-C01 and Wiring Socket Adapter SR6P-S08


Panel Mount Adapter RTB-C01
88.5 max.
Panel thickness: 0.8 to 5
48
1.7
44.5
48
11
53.5
Wiring Socket Adapter SR6P-S08

When using Panel Mount Adapter RTB-C01 and Wiring Socket Adapter SR6P-M08G



Panel Cut-out Dimensions When using Panel Mount Adapter RTB-C01



N = Quantity of analog timer units mounted
Tolerance: +0.5 to 0

All dimensions in mm.

Analog Input Function

The A/D converter unit is used with MICRO^3 to perform an 8-bit A/D conversion. The A/D converter unit reads analog input signals from an analog output device such as an analog distance sensor. The output from the A/D converter unit is entered to MICRO^3 input I0 and converted into a digital value 0 through 249 using the A/D (analog/digital conversion) instruction. If the input to the A/D converter unit exceeds the input range, an overflow occurs and 250 is set to the destination operand of the A/D instruction. Only one A/D converter unit can be connected to the MICRO^3 base unit.

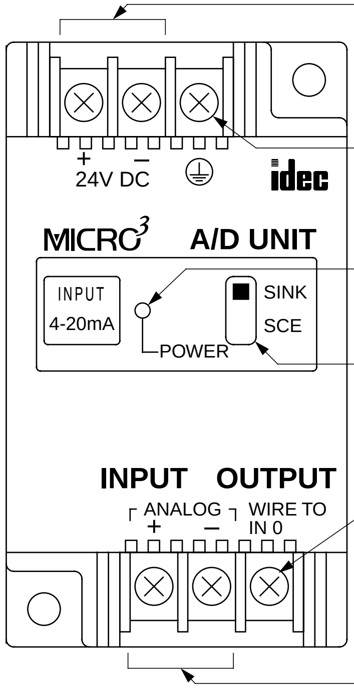
Note: When the A/D converter unit is connected to MICRO^3, the HSC (high-speed counter) function cannot be used.

A/D Converter Unit

Depending on the input signals, five A/D converter units are available:

Type No.	Input Signal Range	Remarks
FC2A-AD1	0 to 5V DC	
FC2A-AD2	0 to 10V DC	
FC2A-AD3	–5 to 5V DC	
FC2A-AD4	4 to 20mA DC	Input resistance 250Ω
FC2A-AD5	–10 to +10V DC	

Parts Description



Power Supply Terminals
Connect power supply 24V DC.

FG Terminal
Connect to the ground. (Grounding resistance 100Ω maximum)

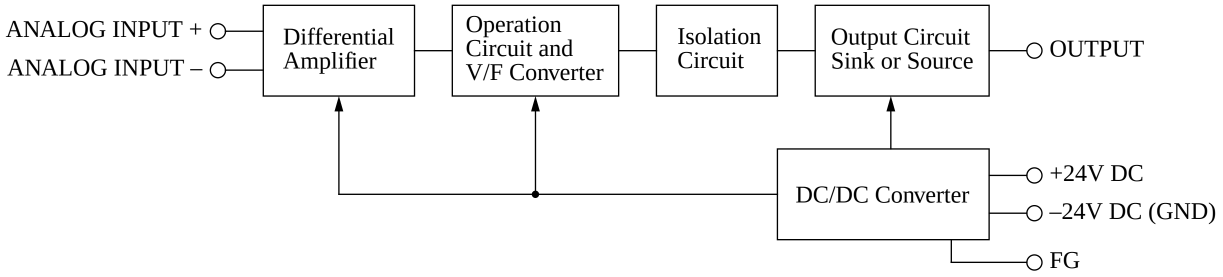
Power Indicator
Turns on when power is supplied.

Output Selector Switch
Select the sink or source output depending on the MICRO^3 input.
Set to SINK when connecting the output to the source MICRO^3 input.
Set to SCE when connecting the output to the sink MICRO^3 input.

Output Terminal
Connect to the input 0 terminal on the MICRO^3.

Analog Input Terminals
Connect analog input signal.

Internal Circuit



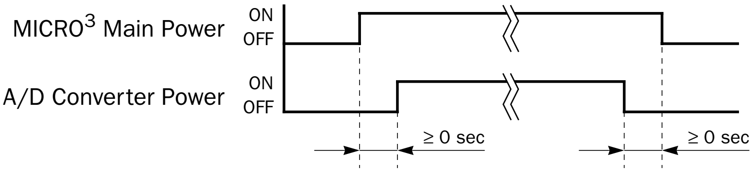
General Specifications (A/D Converter Unit)

Power Supply	**Rated Power Voltage**	24V DC
	Allowable Voltage Range	19 to 30V DC (including ripple)
	Dielectric Strength	Between input and output terminals: 500V AC Between I/O terminal and FG: 1500V AC Between power and output terminals: Not isolated
	Insulation Resistance (500V DC megger)	Between input and output terminals: 10 MΩ minimum Between I/O terminal and FG: 10 MΩ minimum Between power and output terminals: Not isolated
	Power Consumption	Approx. 2.5W (24V DC)
	Allowable Momentary Power Interruption	25 msec minimum (24V DC)
	Power Inrush Current	10A maximum
	Ground	Grounding resistance: 100 Ω maximum
	Protective Ground Allowable Current	10A maximum, 10 sec
	Grounding Wire	1.25 mm^2 (AWG16)
Effect of Improper Power Supply Connection	**Reverse Polarity**	No operation, no damage
	Improper Voltage Level	Permanent damage may be caused
	Improper Lead Connection	Connection failure may be caused
Power Up/Down Order		Power up the A/D converter unit and MICRO^3 at the same time, or power up the MICRO^3 first. Power down the A/D converter unit and MICRO^3 at the same time, or power down the A/D converter first.
Operating Temperature		0 to 60°C
Storage Temperature		–25 to +70°C
Relative Humidity		Relative humidity severity level RH1, 30 to 95% (non-condensing)
Pollution Degree		2 (IEC 664)
Corrosion Immunity		Free from corrosive gases
Altitude		Operation: 0 to 2,000m (0 to 6,565 feet) Transport: 0 to 3,000m (0 to 9,840 feet)
Vibration Resistance (IEC 68-2-6)		5 to 55Hz, 60 m/sec^2, 2 hours each in 3 axes
Shock Resistance (IEC 62-2-7)		300 m/sec^2, 11 msec, 3 shocks each in 3 axes
Wiring		Core wire 0.75 to 1.25 mm^2 (AWG18 to AWG16) Input lines must be separated from power, output, and motor lines. M3 screw terminal with finger protection cover
Input Wiring Length		50m (164 feet) maximum using 2-core shielded wire
Output Wiring Length		2m (6.56 feet) maximum using shielded wire
Dimensions		45W × 80H × 70D mm (1.772"W × 3.150"H × 2.756"D)
Weight		Approx. 120g
Standards		EN61131-1, EN61131-2, EN60204-1 PrEN50082-2, EN55011 UL 508, CSA C22.2 No. 142
Certification File No.		TÜV Product Service E9 95 09 13332 313 UL E102542 CSA LR66809
Others (IEC 1131-2 Information)		IEC1131-2 3.4.1.2.3 6) No common point because of 1 channel input IEC1131-2 3.4.1.2.4 2) No crosstalk because of 1 channel input IEC1131-2 3.4.1.2.4 5) No electromagnetic relay used

Function Specifications (A/D Converter Unit)

<table>
<tr><td colspan="2"><b>Load Impedance in Signal Range</b></td><td colspan="5">Voltage input unit: 1 MΩ minimum
Current input unit: 250Ω</td></tr>
<tr><td colspan="2"><b>Analog Input Error</b></td><td colspan="5">Maximum error at 25°C: ±0.6% of full scale
Temperature coefficient: +0.012% of full scale/°C (typ.)</td></tr>
<tr><td colspan="2"><b>Maximum Error over Full Temperature Range</b></td><td colspan="5">±1% of full scale</td></tr>
<tr><td colspan="2"><b>Digital Resolution</b></td><td colspan="5">250 increments</td></tr>
<tr><td colspan="2"><b>Data Format Returned to User Program</b></td><td colspan="5">BCD (0 to 249, 250: overflow)</td></tr>
<tr><td colspan="2"><b>Input Value of LSB (Least Significant Bit)</b></td><td>FC2A-AD1
40 mV</td><td>FC2A-AD2
64 µA</td><td>FC2A-AD3
20 mV</td><td>FC2A-AD4
40 mV</td><td>FC2A-AD5
80 mV</td></tr>
<tr><td colspan="2"><b>Maximum Permanent Allowed Overload (No Damage)</b></td><td>±16V</td><td>±64 mA</td><td>±16V</td><td>±16V</td><td>±16V</td></tr>
<tr><td colspan="2"><b>Digital Output Reading at Overload</b></td><td colspan="5">250</td></tr>
<tr><td colspan="2"><b>Type of Input</b></td><td colspan="5">Differential input</td></tr>
<tr><td colspan="2"><b>Common Mode Reject Ratio</b></td><td colspan="5">–50 dB</td></tr>
<tr><td colspan="2"><b>Common Mode Voltage</b></td><td colspan="5">16V DC</td></tr>
<tr><td colspan="2"><b>Other Inputs</b></td><td colspan="5">None</td></tr>
<tr><td colspan="2"><b>Total Input System Transfer Time</b></td><td colspan="5">1 msec maximum</td></tr>
<tr><td colspan="2"><b>Sample Duration Time</b></td><td colspan="5">125 msec</td></tr>
<tr><td colspan="2"><b>Sample Repetition Time</b></td><td colspan="5">1 msec maximum</td></tr>
<tr><td colspan="2"><b>Input Filter Characteristics</b></td><td colspan="5">None</td></tr>
<tr><td colspan="2"><b>Maximum Temporary Deviation during Electrical Noise Tests and Test Conditions</b></td><td colspan="5">3% maximum of full scale at impulse test 500V</td></tr>
<tr><td colspan="2"><b>Conversion Method</b></td><td colspan="5">V-F conversion</td></tr>
<tr><td colspan="2"><b>Operating Mode</b></td><td colspan="5">Self-scan</td></tr>
<tr><td colspan="2"><b>Type of Protection</b></td><td colspan="5">Resistor, diode, photocoupler</td></tr>
<tr><td colspan="2"><b>Output Short-circuit</b></td><td colspan="5">Damage will be caused</td></tr>
<tr><td colspan="2"><b>Maximum Allowed Output Voltage</b></td><td colspan="5">Voltage at the +24V DC terminal + 0.3V or – 0.3V
(between GND and OUTPUT)</td></tr>
<tr><td colspan="2"><b>Maximum Allowed Input Voltage</b></td><td colspan="5">Maximum permanent allowed overload (no damage)</td></tr>
<tr><td rowspan="4"><b>External Power Supply Data</b></td><td><b>Rated Power Voltage</b></td><td colspan="5">24V DC</td></tr>
<tr><td><b>Power Voltage Range</b></td><td colspan="5">19 to 30V DC</td></tr>
<tr><td><b>Resetting Method</b></td><td colspan="5">Use a power supply of self-reset type or with an overcurrent protection against 10A inrush current into the A/D converter unit</td></tr>
<tr><td><b>Output Power</b></td><td colspan="5">2.5W minimum</td></tr>
<tr><td colspan="2"><b>Calibration or Verification to Maintain Rated Accuracy</b></td><td colspan="5">Once every 6 months (recommended value)</td></tr>
<tr><td colspan="2"><b>Effect of Improper Input Terminal Connection</b></td><td colspan="5">If a signal over the maximum permanent allowed overload (no damage) is applied, permanent damage may be caused.</td></tr>
<tr><td colspan="2"><b>Monotonicity</b></td><td colspan="5">Yes</td></tr>
<tr><td colspan="2"><b>Non-linearity</b></td><td colspan="5">0.2% maximum of full scale</td></tr>
<tr><td colspan="2"><b>Repeatability after Stabilization Time</b></td><td colspan="5">0.5% maximum of full scale (more than 30 minutes after power up)</td></tr>
<tr><td colspan="2"><b>Life Time of Electromagnetic Relay Multiplexers</b></td><td colspan="5">None</td></tr>
<tr><td colspan="2"><b>MICRO^3 Setting</b></td><td colspan="5">Set hard filter value to 10 (default) using FUN7
Factory setting: 10 (default)</td></tr>
</table>

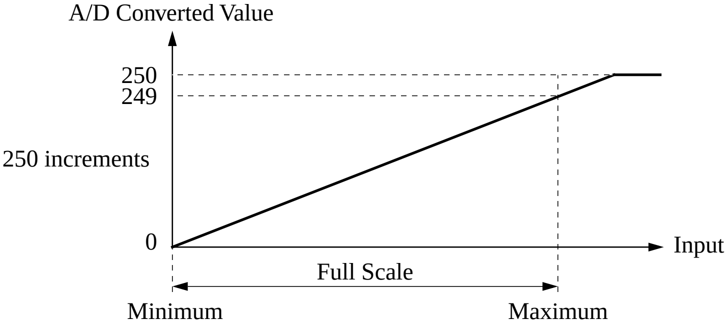
Power Supply Timing Chart



Power up the A/D converter unit and MICRO3 at the same time, or power up the MICRO3 first.

Power down the A/D converter unit and MICRO3 at the same time, or power down the A/D converter first.

Digital Resolution



If the input signal changes within the conversion time of 125 msec, an error in the converted value will result.

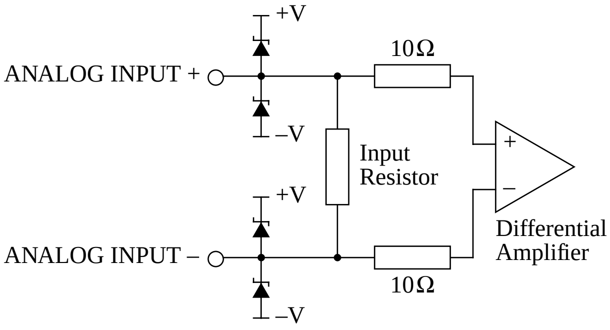
If the input to the A/D converter unit is below the minimum input, 0 is set to the destination operand of the A/D instruction.

If the input to the A/D converter unit is over the maximum input, an overflow occurs and 250 is set to the destination operand of the A/D instruction.

$$\text{A/D Converted Value} = \frac{\text{Analog Input Value} - \text{Minimum Input Value}}{\text{Maximum Input Value} - \text{Minimum Input Value}} \times 249$$

Type of Protection

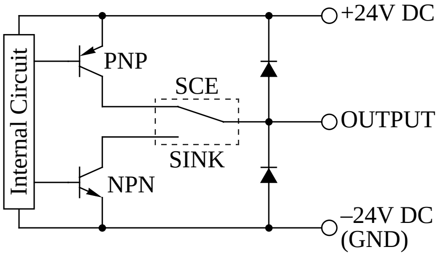
- **Input Circuit**



- **Output Circuit**

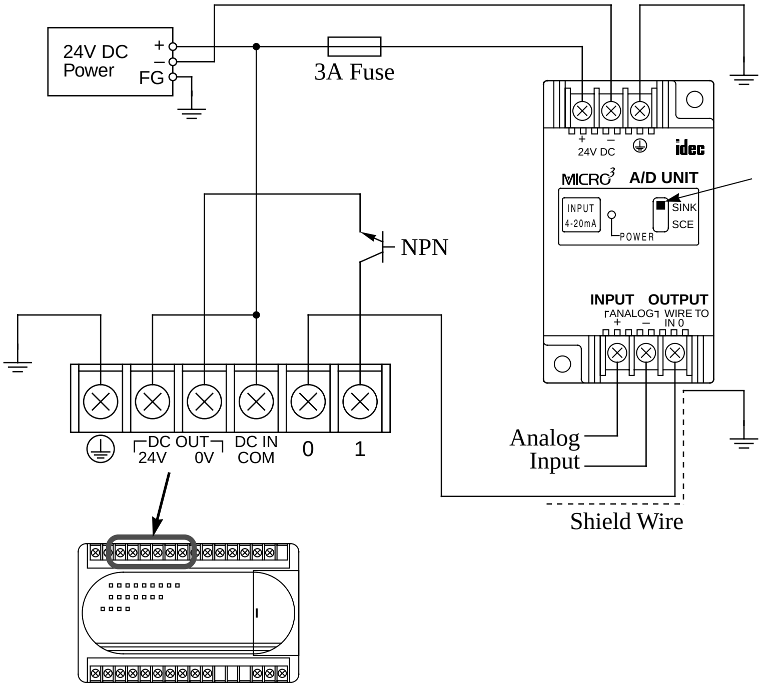
Open collector

The output selector switch is used to select sink or source output.



Wiring Diagram

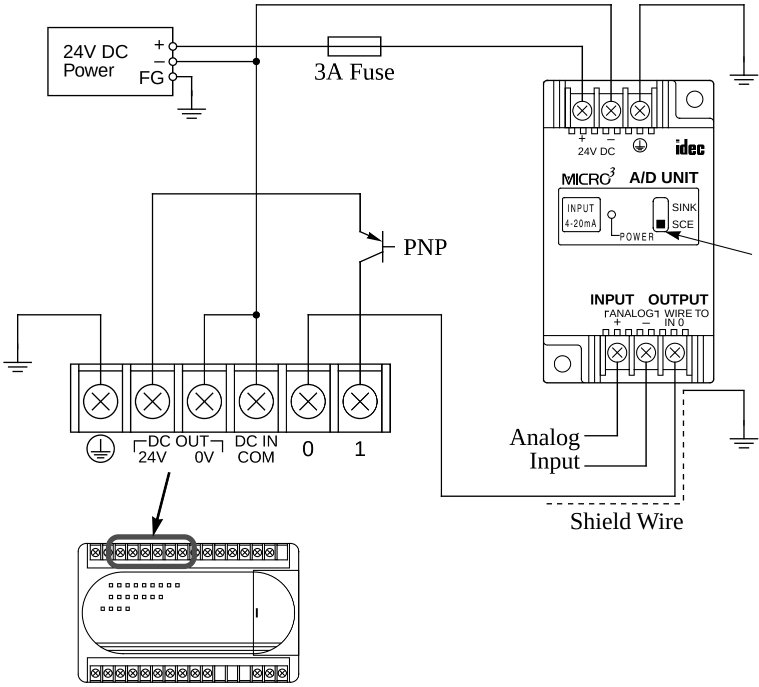
- **Source Input to MICRO^3**



When using the MICRO^3 in the source input connection, select the sink output from the A/D converter unit.

Set the output selector switch to SINK.

- **Sink Input to MICRO^3**



When using the MICRO^3 in the sink input connection, select the source output from the A/D converter unit.

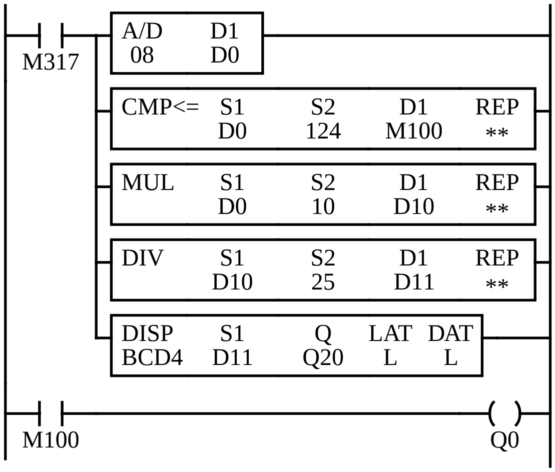
Set the output selector switch to SCE.

Example: A/D Conversion

The following example demonstrates a program to perform ON/OFF control on a heater using the A/D converter unit (4 to 20 mA). The temperature sensor generates an analog output of 4 through 20 mA while the temperature changes from 0°C through 100°C. The output from the temperature sensor is connected to the A/D converter unit. The output from the A/D converter unit is connected to input I0 of MICRO^3. When the temperature is 50°C or less, output Q0 is turned on to turn the heater on. When the temperature is above 50°C, output Q0 is turned off to turn the heater off. The temperature is also displayed on digital display units.

Temperature (°C)	Sensor Output (mA)	A/D Converted Value	Heater
0	4	0	ON
50	12	124	ON
51	12.064	125	OFF
100	20	249	OFF

Ladder Diagram


M317
A/D 08 D1 D0
CMP<= S1 D0 S2 124 D1 M100 REP **
MUL S1 D0 S2 10 D1 D10 REP **
DIV S1 D10 S2 25 D1 D11 REP **
DISP BCD4 S1 D11 Q Q20 LAT L DAT L
M100
Q0

M317 is the in-operation output special internal relay.

The analog data from the A/D converter unit is 8-bit converted to a digital value 0 through 249 and set to data register D0.

When the D0 value is less than or equal to 124, internal relay M100 is turned on.

The D0 value is multiplied by 10, and the result is set to data register D10.

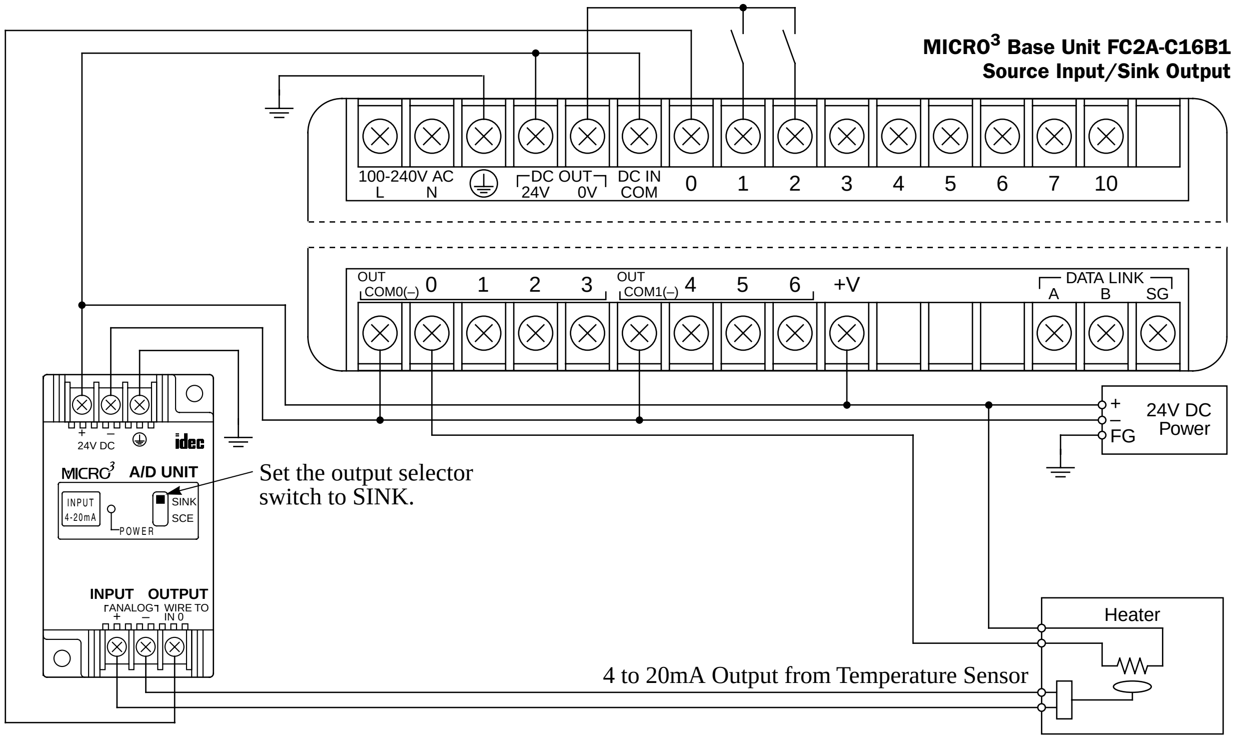
The D10 value is divided by 25, and the result is set to data register D11.

The 4-digit D11 value is displayed on display units connected to outputs Q20 through Q27.

When M100 is on, output Q0 is turned on.

I/O Wiring Diagram

This wiring example shows a source-input, sink-output connection for the MICRO^3 base unit. Digital display units are connected to outputs Q20 through Q27 at the expansion station (not shown).



Analog Output Function

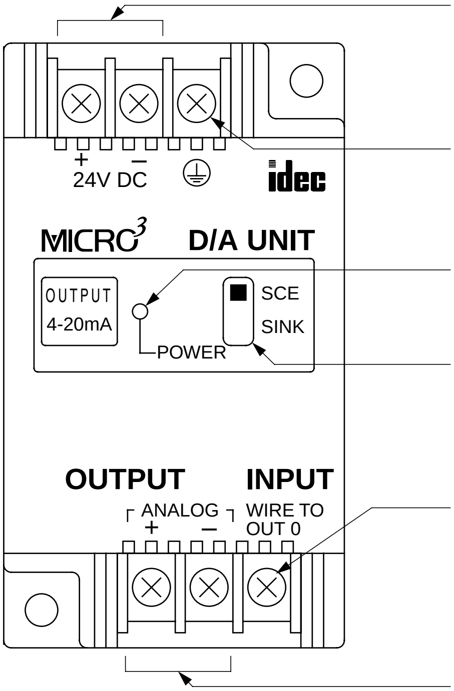
The D/A converter unit is used with the transistor output type MICRO^3 base unit to perform an 8-bit D/A conversion. The PWM (pulse width modulation) instruction is used to convert the digital value to a pulse output signal. The output from MICRO^3 output Q0 is entered to the D/A converter unit to generate an analog current or voltage output to control an inverter. Only one D/A converter unit can be connected to the MICRO^3 base unit.

D/A Converter Unit

Depending on the output signals, five D/A converter units are available:

Type No.	Output Signal Range	Remarks
FC2A-DA1	0 to 5V DC	
FC2A-DA2	0 to 10V DC	
FC2A-DA3	–5 to +5V DC	
FC2A-DA4	4 to 20mA DC	Load resistance 300Ω maximum
FC2A-DA5	–10 to +10V DC	

Parts Description



Power Supply Terminals
Connect power supply 24V DC.

FG Terminal
Connect to the ground. (Grounding resistance 100Ω maximum)

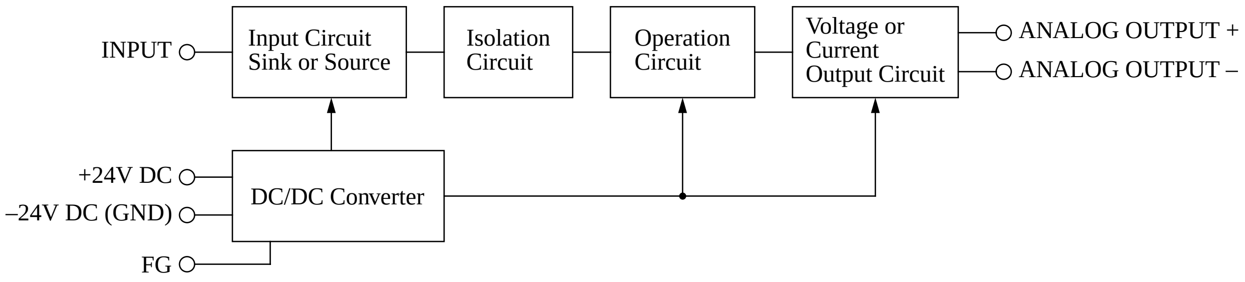
Power Indicator
Turns on when power is supplied.

Input Selector Switch
Select the sink or source input depending on the MICRO^3 output.
Set to SINK when connecting the input to the source MICRO^3 output.
Set to SCE when connecting the input to the sink MICRO^3 output.

Input Terminal
Connect to the output 0 terminal on the MICRO^3.

Analog Output Terminals
Send out analog current or voltage output signal.

Internal Circuit



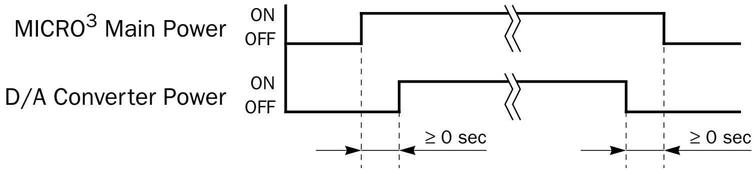
General Specifications (D/A Converter Unit)

Power Supply	**Rated Power Voltage**	24V DC
	Allowable Voltage Range	19 to 30V DC (including ripple)
	Dielectric Strength	Between input and output terminals: 500V AC Between I/O terminal and FG: 1500V AC Between power and output terminals: Not isolated
	Insulation Resistance (500V DC megger)	Between input and output terminals: 10 MΩ minimum Between I/O terminal and FG: 10 MΩ minimum Between power and output terminals: Not isolated
	Power Consumption	Approx. 2.5W (24V DC)
	Allowable Momentary Power Interruption	25 msec minimum (24V DC)
	Power Inrush Current	10A maximum
	Ground	Grounding resistance: 100 Ω maximum
	Protective Ground Allowable Current	10A maximum, 10 sec
	Grounding Wire	1.25 mm^2 (AWG16)
Effect of Improper Power Supply Connection	**Reverse Polarity**	No operation, no damage
	Improper Voltage Level	Permanent damage may be caused
	Improper Lead Connection	Connection failure may be caused
Power Up/Down Order		Power up the D/A converter unit and MICRO^3 at the same time, or power up the MICRO^3 first. Power down the D/A converter unit and MICRO^3 at the same time, or power down the D/A converter first.
Operating Temperature		0 to 60°C
Storage Temperature		–25 to +70°C
Relative Humidity		Relative humidity severity level RH1, 30 to 95% (non-condensing)
Pollution Degree		2 (IEC 664)
Corrosion Immunity		Free from corrosive gases
Altitude		Operation: 0 to 2,000m (0 to 6,565 feet) Transport: 0 to 3,000m (0 to 9,840 feet)
Vibration Resistance (IEC 68-2-6)		5 to 55Hz, 60 m/sec^2, 2 hours each in 3 axes
Shock Resistance (IEC 62-2-7)		300 m/sec^2, 11 msec, 3 shocks each in 3 axes
Wiring		Core wire 0.75 to 1.25 mm^2 (AWG18 to AWG16) Input lines must be separated from power, output, and motor lines. M3 screw terminal with finger protection cover
Input Wiring Length		2m (6.56 feet) maximum using shielded wire
Output Wiring Length		50m (164 feet) maximum using 2-core shielded wire
Dimensions		45W × 80H × 70D mm (1.772"W × 3.150"H × 2.756"D)
Weight		Approx. 120g
Standards		EN61131-1, EN61131-2, EN60204-1 PrEN50082-2, EN55011 UL 508, CSA C22.2 No. 142
Certification File No.		TÜV Product Service E9 95 09 13332 313 UL E102542 CSA LR66809
Others (IEC 1131-2 Information)		IEC1131-2 3.4.2.2.3 8) No common point because of 1 channel input

Function Specifications (D/A Converter Unit)

Load Impedance in Signal Range		Voltage output unit: 5 kΩ minimum Current output unit: 250Ω (300Ω maximum)
Analog Output Error		Maximum error at 25°C: ±0.7% of full scale Temperature coefficient: –0.005% of full scale/°C (typ.)
Maximum Error over Full Temperature Range		±1% of full scale
Digital Resolution		245 increments
Data Format in User Program		BCD (0 to 249)
Value of LSB (Least Significant Bit)		FC2A-DA1 FC2A-DA2 FC2A-DA3 FC2A-DA4 FC2A-DA5 41 mV 65 µA 20 mV 41 mV 82 mV
Total Output System Transfer Time		1 msec maximum
Settling Time after Maximum Range Change		0.5 sec maximum after changing from 0% to 95%
Overshoot		0%
Maximum Temporary Deviation during Electrical Noise Tests and Test Conditions		3% maximum of full scale at impulse test 500V
Output Short-circuit		No damage (between OUTPUT + and –)
Maximum Allowed Output Voltage		Voltage output type: ±12V DC (between OUTPUT + and –) Current output type: +12 or –0.6V DC (between OUTPUT + and –)
Maximum Allowed Input Voltage		Between INPUT and GND: +30V DC Between INPUT and +24V: –30V DC
Output Voltage Drop		1% maximum of full scale
External Power Supply Data	**Rated Power Voltage**	24V DC
	Power Voltage Range	19 to 30V DC
	Resetting Method	Use a power supply of self-reset type or with an overcurrent protection against 10A inrush current into the D/A converter unit
	Inrush Current	10A
	Output Power	2.5W minimum
Calibration or Verification to Maintain Rated Accuracy		Once every 6 months (recommended value)
Type of Applicable Load		Resistive load (5 kΩ minimum, voltage output type)
Effect of Improper Output Terminal Connection		Permanent damage may be caused
Monotonicity		Yes
Crosstalk		No crosstalk because of 1 channel output
Non-linearity		0.2% maximum of full scale
Repeatability after Stabilization Time		0.5% maximum of full scale (more than 30 minutes after power up)
Output Ripple		1% maximum of full scale
MICRO^3 Setting		PWM MODE3

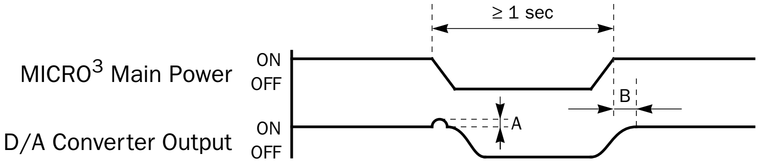
Power Supply Timing Chart



Power up the D/A converter unit and MICRO^3 at the same time, or power up the MICRO^3 first.

Power down the D/A converter unit and MICRO^3 at the same time, or power down the D/A converter first.

Output Response at Power Up and Down



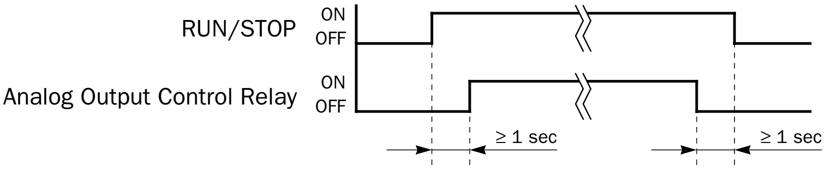
A ≤ 1% of full scale
B ≤ 1 sec

Caution

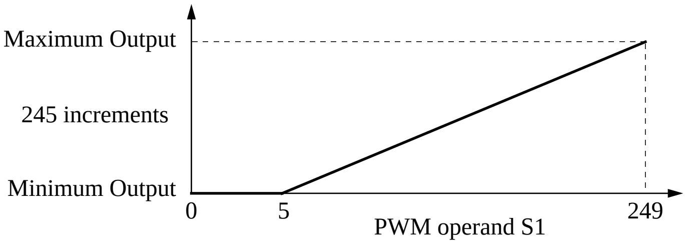
D/A converter units FC2A-DA3 and FC2A-DA5 generate a momentary voltage output when the MICRO^3 or the D/A converter unit is powered up or when the MICRO^3 is started or stopped.

- FC2A-DA3: –5V DC
- FC2A-DA5: –10V DC

When the analog voltage output from the FC2A-DA3 or FC2A-DA5 is used to control motor and a trouble may occur, use a relay to ensure a delay between the RUN/STOP signal and the analog output as shown below.



Resolution



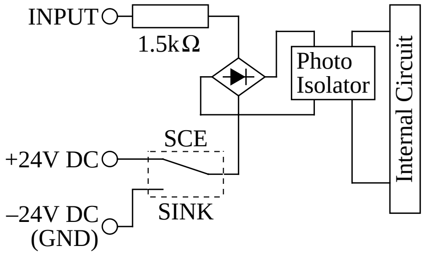
When the value of the data register designated as operand S1 (pulse width coefficient) for the PWM (pulse width modulation) instruction is between 0 and 4, S1 is designated as 5, and the minimum output is generated.

Do not designate constant 0 through 4 as S1. If a constant value between 0 and 4 is designated as S1, the output is not generated correctly.

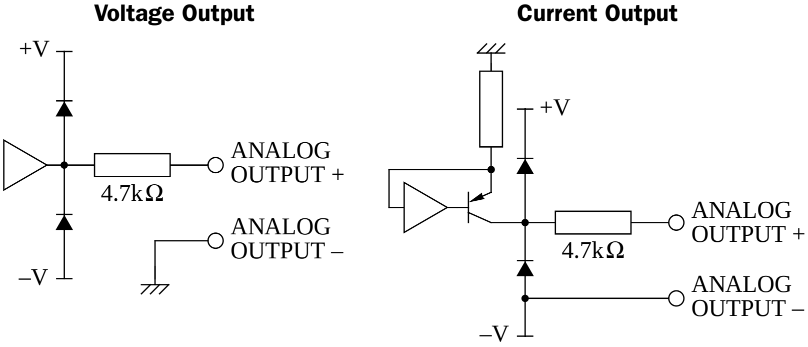
$$\text{Analog Output Value} = (\text{Maximum Output} - \text{Minimum Output}) \times \frac{S1 - 5}{244} + \text{Minimum Output}$$

Type of Protection

- **Input Circuit**
  The input selector switch is used to select sink or source input.

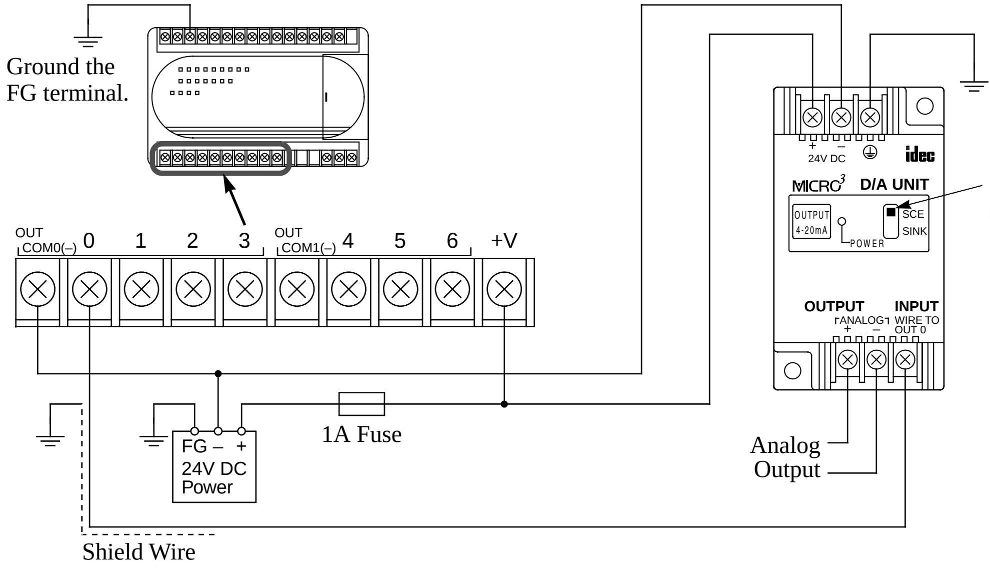


- **Output Circuit**



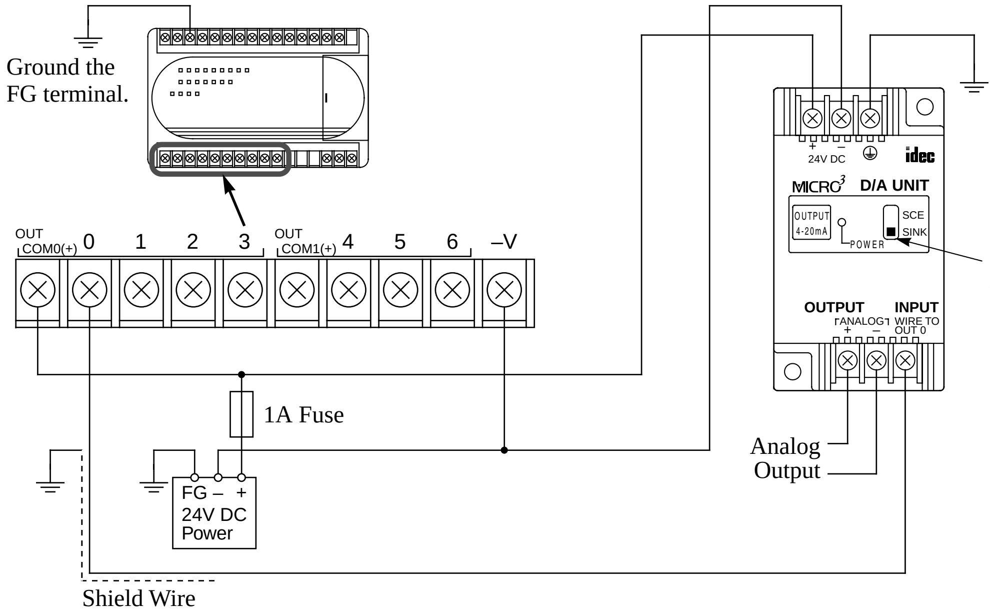
Wiring Diagram

- **Transistor Sink Output from MICRO3**



When using a transistor sink output type MICRO3 base unit, select the source input to the D/A converter unit.

- **Transistor Protect Source Output from MICRO3**

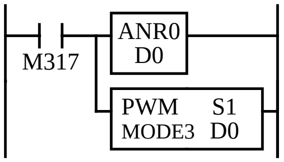


When using a transistor protect source output type MICRO3 base unit, select the sink input to the D/A converter unit.

Example: D/A Conversion

The following example demonstrates a program to control motor speed using the D/A converter unit. Analog potentiometer 0 on the MICRO3 base unit is used to change the digital value for operand S1 (pulse width coefficient) of the PWM (pulse width modulation) instruction. The PWM output from output Q0 is converted to an analog value by the D/A converter unit and the output from the D/A converter unit is entered to the inverter to control the motor speed.

Ladder Diagram



M317 is the in-operation output special internal relay.

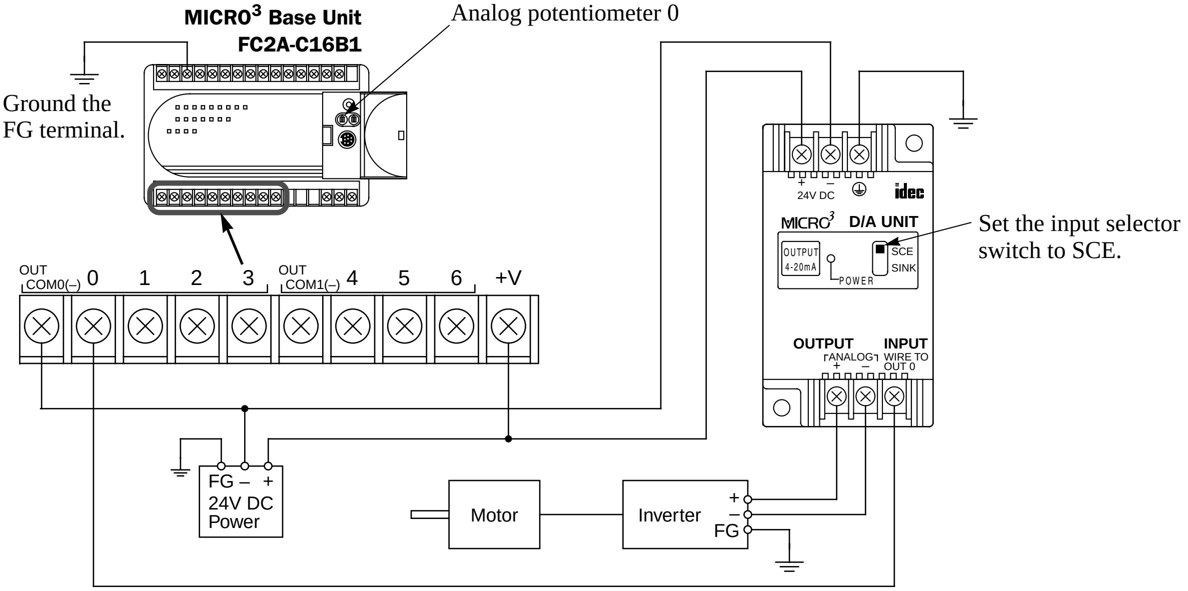
The ANR0 (analog read 0) instruction reads the analog potentiometer 0 setting and sets digital value 0 through 249 to data register D0.

The PWM (pulse width modulation) instruction converts the D0 value to a pulse output signal of variable pulse widths.

Note: When using the D/A converter unit, the MODE in the PWM instruction must be set to MODE3 to make sure of correct output from the D/A converter unit. When the value of the data register designated as S1 is between 0 and 4, the PWM instruction sets operand S1 to 5, and the minimum output is generated by the D/A converter unit. If the value of the data register designated as S1 exceeds 249 during operation, a user program execution error will occur, then error indicator ERR1 on the MICRO3 base unit is lit, and special internal relay M304 is also turned on. Do not designate constant 0 through 4 as S1. If a constant value between 0 and 4 is designated as S1, then the output is not generated correctly.

I/O Wiring Diagram

This wiring example shows a sink-output connection for the MICRO3 base unit.



Ground the FG terminal.

Set the input selector switch to SCE.

◆ ***For dimensions of A/D and D/A converter units, see page 1-24.*** ◆

Introduction

This chapter describes setting the FUN (function) table.

FUN1 through FUN11 are used to configure the user program and these settings must be designated before attempting to transfer the user program to the MICRO^3 base unit.

FUN20 through FUN28 are used to check the MICRO^3 base unit status and data.

FUN29 is used to read user communication status to the program loader (MICRO^{3}C only).

FUN30 through FUN36 are used to set the operation modes of the program loader.

FUN40 through FUN43 are used for the memory card installed in the program loader.

FUN50 is used to monitor user communication data on the program loader (MICRO^{3}C only).

FUN Settings (FUN1 through FUN11)

For	Number	Name	Function	Option	Default
User Program	**FUN1**	Stop input number selection	Selects any input terminal as a stop input.	0 to 15	None
	FUN2	Reset input number selection	Selects any input terminal as a reset input.	0 to 15	None
	FUN3	Internal relay “keep” designation	Designates a range of internal relays as keep type.	Standard processing: 0 to 287 High-speed processing: 0 to 47	All clear types
	FUN4	Shift register “keep” designation	Designates a range of shift register bits as keep type.	Standard processing: 0 to 63 High-speed processing: 0 to 31	All clear types
	FUN5	Processing mode selection	Selects standard or high-speed processing mode.	Standard or high-speed processing mode	Standard
	FUN6	Rising or falling edge selection for catch inputs	Selects rising (ON pulse) or falling edge (OFF pulse) to receive catch inputs.	Up or down	Up
	FUN7	Input filter time selection	Selects the input filter time.	Hard filter: 0 to 255 Soft filter: 0, 3, 7, 10	Hard: 10 Soft: 3
	FUN8	Loader port communication mode setting	Sets the communication format for MICRO^3 connected to computer or modem.	Baud rate Terminator code Data bits Parity check Stop bits Mode selection input Receive timeout	9600 bps 0D (CR) 7 bits Even 1 bit None 500 msec
	FUN9	PLC address for network communication	Sets the communication device number of MICRO^3 in 1:N computer link.	0 to 31	0
	FUN10	Control data register setting	Enables or disables control data register function.	Enable or disable	All disabled
	FUN11	Program capacity and PLC type selection	Selects the program capacity to write into the program loader and selects the PLC type.	Capacity: 244, 500, 1K PLC: MICRO^3, MICRO^{3}C	1K (initial) MICRO^3

Note: Since FUN1 through FUN11 settings relate to the user program, the user program must be transferred to the MICRO^3 after changing any of these settings. When the user program is cleared using the DEL, END, and ↵ keys, FUN1 through FUN10 settings are also reset to the default values. The FUN11 value is not changed by deleting the entire user program.

FUN Settings (FUN20 through FUN50)

For	Number	Name	Function
MICRO^3 Status	**FUN20**	PLC error data readout and reset	Displays the error code and data of the MICRO^3, and clears the error data.
	FUN21	Timer/counter preset value readout and restore	Reads changed timer/counter preset values from MICRO^3. Restores the original timer/counter preset values.
	FUN22	User program protection	Protects the user program in the MICRO^3 from reading and/or writing. Cancels the user program protection.
	FUN23	PLC system program version readout	Displays the MICRO^3 system program version.
	FUN24	PLC operating status readout	Displays the run/stop status of MICRO^3.
	FUN25	Scan time readout	Displays the current and maximum scan time values of the user program in operation.
	FUN26	Operand data clear	Clears all or selected operand data of the user program.
	FUN27	Link formatting sequence	Updates the data link terminal connection data.
	FUN28	Calendar/clock data readout and setting	Displays and changes the calendar/clock data in MICRO^3.
	FUN29	User communication status readout	Displays user communication error data, execution of transmit/ receive instructions, and communication parameters.
Program Loader	**FUN30**	Program check	Checks the user program in the program loader and displays program errors, if any.
	FUN31	Program loader version readout/hardware check	Displays the program loader version. Checks the display and internal RAM of the program loader.
	FUN32	Sequential monitoring	Enables or disables sequential monitoring in the editor mode.
	FUN33	Monitor screen holding	Enables or disables to hold monitor data when the program loader is turned off or when the monitor mode is exited.
	FUN34	Program loader beep sound	Turns on or off the program loader beep sound.
	FUN35	Display language selection	Selects the display language from English or Japanese.
	FUN36	Display data type selection	Selects decimal or hexadecimal notation of constant values during editing and monitoring.
Memory Card	**FUN40**	Memory card identification	Displays the memory card battery status, card name, and stored user program names.
	FUN41	Memory card formatting	Formats the memory card to store user programs.
	FUN42	Program loader system program installation	Installs another system program from the memory card into the program loader.
	FUN43	Program loader system program restore	Erases the additional system program from the program loader.
Communication Monitor	**FUN50**	User communication data monitor	Monitors transmit and receive data of user communication between the MICRO^{3}C and RS232C equipment.

Note 1: FUN32 through FUN36 settings are held when the program loader is turned off.

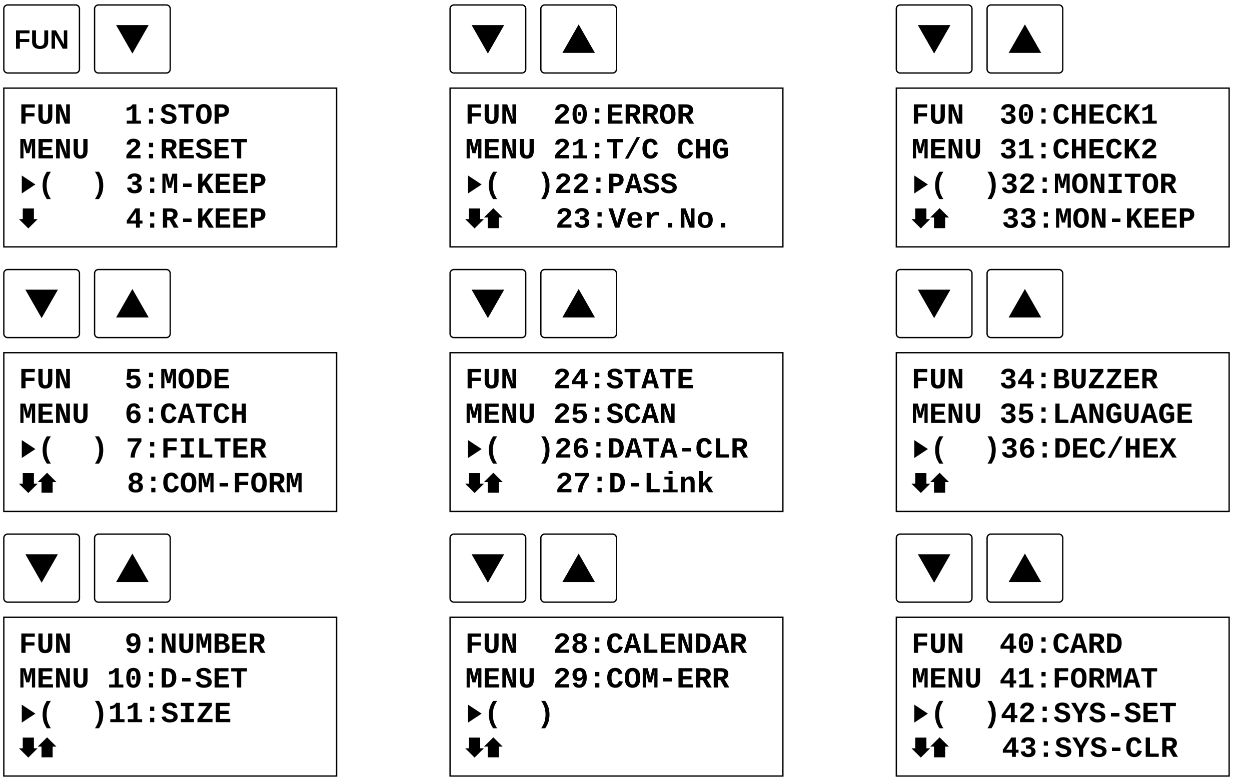
Note 2: FUN29 and FUN50 can be used on the MICRO^{3}C only.

Key Operation

Each FUN setting screen can be called directly or from a FUN menu.

To directly call an individual FUN setting screen, press the FUN key, then enter a FUN number, and press the ▼ key as described in detail in the following sections.

To call a FUN menu, press the FUN key, followed by the ▼ key. To display the next or previous page of the FUN menu, press the ▼ or ▲ key.



```
FUN   1:STOP
MENU  2:RESET
▶(  ) 3:M-KEEP
⬇     4:R-KEEP
```

```
FUN   5:MODE
MENU  6:CATCH
▶(  ) 7:FILTER
⬇⬆    8:COM-FORM
```

```
FUN   9:NUMBER
MENU 10:D-SET
▶(  )11:SIZE
⬇⬆
```

```
FUN  20:ERROR
MENU 21:T/C CHG
▶(  )22:PASS
⬇⬆   23:Ver.No.
```

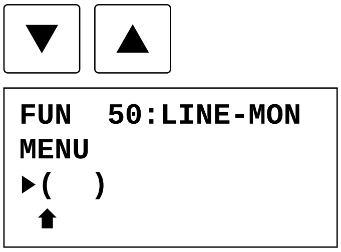
```
FUN  24:STATE
MENU 25:SCAN
▶(  )26:DATA-CLR
⬇⬆   27:D-Link
```

```
FUN  28:CALENDAR
MENU 29:COM-ERR
▶(  )
⬇⬆
```

```
FUN  30:CHECK1
MENU 31:CHECK2
▶(  )32:MONITOR
⬇⬆   33:MON-KEEP
```

```
FUN  34:BUZZER
MENU 35:LANGUAGE
▶(  )36:DEC/HEX
⬇⬆
```

```
FUN  40:CARD
MENU 41:FORMAT
▶(  )42:SYS-SET
⬇⬆   43:SYS-CLR
```



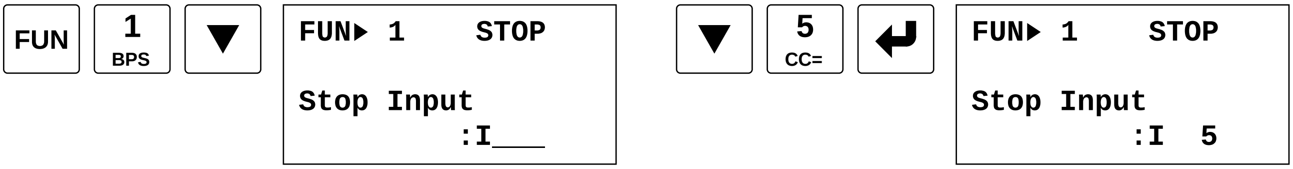
```
FUN  50:LINE-MON
MENU
▶(  )
 ⬆
```

When the required FUN number is displayed, enter the number, and press the ↵ key.

Individual FUN setting screens for FUN1 through FUN11 and for FUN32 through FUN36 are called successively by pressing the ▶ or ◀ key.

FUN1: Stop Input Number Selection

There are several ways to start and stop the MICRO^3 base unit. See page 2-1 for detailed information on starting and stopping operations. One method for stopping the MICRO^3 operation is to designate an input number as the stop terminal. When this input is turned on, the MICRO^3 operation is stopped:.



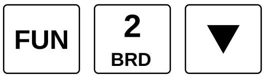
```
FUN▶ 1    STOP

Stop Input
          :I___
```

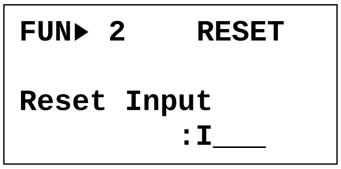
```
FUN▶ 1    STOP

Stop Input
          :I  5
```

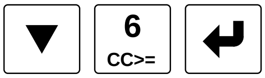
Applicable stop input numbers: I0 through I15
Default: No stop input
Input numbers at the expansion station cannot be designated as a stop input.
To cancel the stop input number selection, move the cursor down to the colon, and press the DEL and ↵ keys.
To return to the editor mode, press the CLR key.

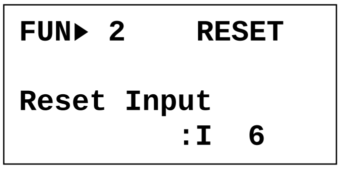
FUN2: Reset Input Number Selection

FUN2 is provided to designate an input number as a reset terminal. When this input is turned on, MICRO^3 stops operation and resets all statuses. See page 2-2 for detailed information on system status during reset. To set FUN2:




```
FUN▶ 2    RESET

Reset Input
          :I___
```



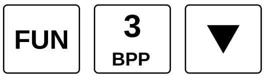

```
FUN▶ 2    RESET

Reset Input
          :I  6
```

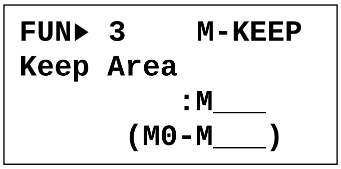
Applicable reset input numbers: I0 through I15
Default: No reset input
Input numbers at the expansion station cannot be designated as a reset input.
To cancel the reset input number selection, move the cursor to the colon, and press the DEL and ↵ keys.
To return to the editor mode, press the CLR key.

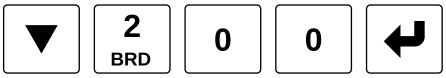
FUN3: Internal Relay “Keep” Designation

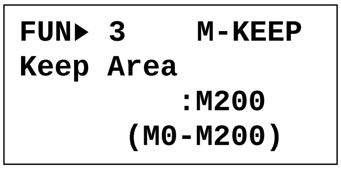
The status of any internal relay from M0 through M287 is normally cleared during a power failure. It is possible to maintain the status of an internal relay by using FUN3 to designate the internal relay as a “keep” type.

The “keep” designation can only be specified for a block of consecutive internal relay numbers, starting with M0. When FUN3 is set to specify an internal relay between M0 and M287, all those numbers M0 through the specified one will be designated as “keep” type. All internal relays above the specified number will be cleared.




```
FUN▶ 3    M-KEEP
Keep Area
          :M___
       (M0-M___)
```




```
FUN▶ 3    M-KEEP
Keep Area
          :M200
       (M0-M200)
```

Applicable “keep” designation internal relay numbers: M0 through M287 in the standard processing mode
M0 through M47 in the high-speed processing mode
Default: All “clear” type internal relays
To cancel the internal relay “keep” designation, move the cursor down to the colon, and press the DEL and ↵ keys.
To return to the editor mode, press the CLR key.

When M200 is assigned as shown above, internal relays M0 through M200 become “keep” type internal relays and M201 through M287, “clear” types.

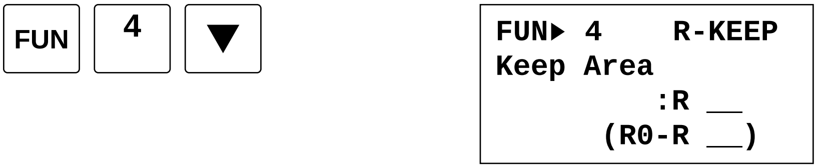
Special internal relays M290 through M297 assigned for catch inputs are always “clear” types.

For the status of special internal relays M300 through M317, see page 6-2.

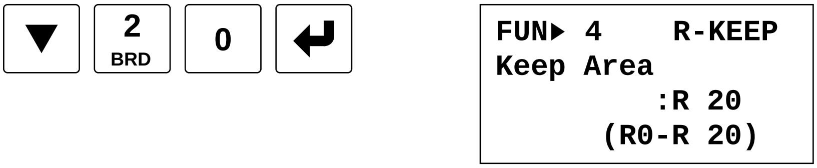
FUN4: Shift Register "Keep" Designation

The status of any shift register bit from R0 through R63 is normally cleared during a power failure. It is possible to maintain the status of a shift register bit by using FUN4 to designate the shift register bit as a "keep" type.

The "keep" designation can only be specified for a block of consecutive shift register bits, starting with R0. When FUN4 is set to specify a shift register bit between R0 and R63, all those numbers R0 through the specified one will be designated as "keep" type. All shift register bits above the specified number will be cleared.



```
FUN▶ 4    R-KEEP
Keep Area
         :R __
      (R0-R __)
```



```
FUN▶ 4    R-KEEP
Keep Area
         :R 20
      (R0-R 20)
```

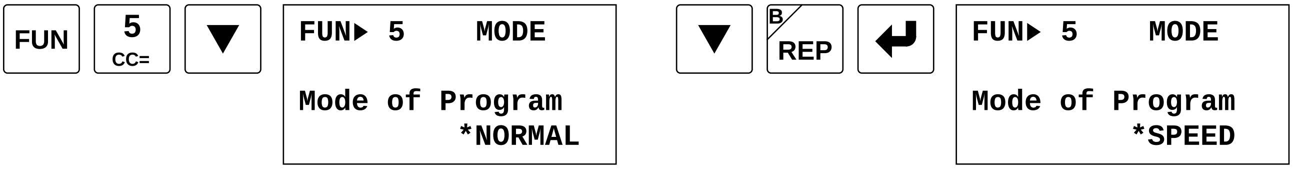
Applicable "keep" designation shift register bit numbers: R0 through R63 in the standard processing mode
R0 through R31 in the high-speed processing mode
Default: All "clear" type shift register bits
To cancel the shift register "keep" designation, move the cursor down to the colon, and press the DEL and ↵ keys.
To return to the editor mode, press the CLR key.

When R20 is assigned as shown above, shift register bits R0 through R20 become "keep" type shift register bits and R21 through R63, "clear" types.

FUN5: Processing Mode Selection

FUN5 is provided to select the standard or high-speed processing mode. The high-speed processing mode has average scan time of 400 µsec per 100 steps and can be used as a sensor controller in combination with the catch input function. For details of the high-speed processing mode, see page 4-1.

To select the high-speed processing mode, press the keys:



```
FUN▶ 5    MODE

Mode of Program
         *NORMAL
```

```
FUN▶ 5    MODE

Mode of Program
          *SPEED
```

Pressing the REP key toggles NORMAL and SPEED to select the standard or high-speed processing mode.
The default selection is the standard operation mode.
To return to the editor mode, press the CLR key

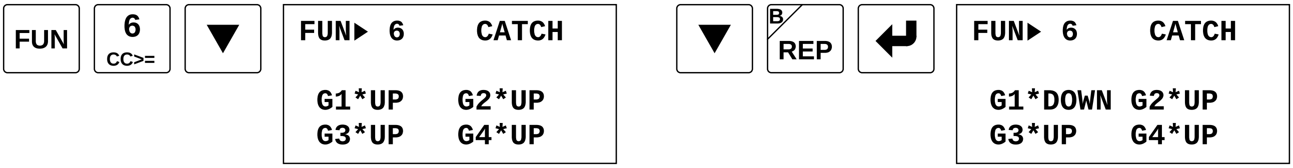
In the high-speed operation mode, the I/O expansion function and data link function cannot be used, and available operands are limited. The program capacity is approximately 100 steps. For available operands, see page 6-1.

User communication transmit (TXD) and receive (RXD) instructions for the MICRO^{3}C cannot be used in the high-speed processing mode.

FUN6: Rising or Falling Edge Selection for Catch Inputs

FUN6 is provided to select whether catch inputs are accepted at the rising (ON pulse) or falling edge (OFF pulse). The catch input signals entered to inputs I0 through I7 are stored to special internal relays M290 through M297. See page 4-2. The edge selection is done in 4 groups of catch inputs:

Catch Group	G1	G2	G3	G4
Input Number	I0	I1	I2 and I3	I4 through I7 (16- and 24-I/O MICRO^3 base units) I4 and I5 (10-I/O MICRO^3 base unit)



```
FUN▶ 6    CATCH

 G1*UP   G2*UP
 G3*UP   G4*UP
```

```
FUN▶ 6    CATCH

 G1*DOWN G2*UP
 G3*UP   G4*UP
```

Pressing the REP key toggles UP and DOWN to select the rising edge or the falling edge.
To move from a group to another, press the ▼ or ▲ key.
The default selection is the rising edge for all groups.
To return to the editor mode, press the CLR key.

FUN7: Input Filter Time Selection

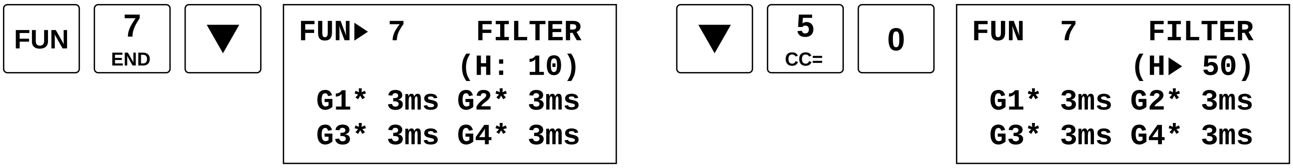
FUN7 is provided to select the input filter time to accept short-pulse inputs or to eliminate noise pulses. When catch input special internal relays M290 through M297 are not included in the user program, set the input filter time to a small value to make sure of receiving short-pulse input signals.

The input filter time for catch inputs and high-speed counter inputs can be selected using the hard filter setting. The hard filter is set to the same value for inputs I0 through I7. The hard filter preset value can be between 0 and 255. For details of the hard filter time calculated from the preset value, see page 4-4.

The soft filter can be selected from 0, 3, 7, and 10 msec and the selection is done in 4 groups of inputs:

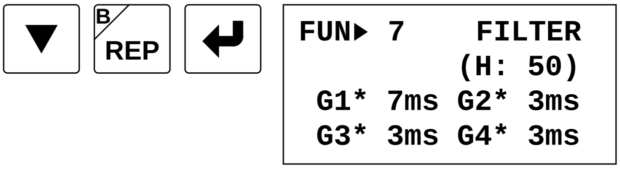
Soft Group	G1	G2	G3	G4
Input Number	I0	I1	I2 and I3	I4 through I7 (16- and 24-I/O MICRO^3 base units) I4 and I5 (10-I/O MICRO^3 base unit)

Inputs I10 through I35 are provided with a fixed filter of 3 msec. Hard and soft filters cannot be used for I10 through I35.



```
FUN▶ 7    FILTER
         (H: 10)
 G1* 3ms G2* 3ms
 G3* 3ms G4* 3ms
```

```
FUN  7    FILTER
         (H▶ 50)
 G1* 3ms G2* 3ms
 G3* 3ms G4* 3ms
```



```
FUN▶ 7    FILTER
         (H: 50)
 G1* 7ms G2* 3ms
 G3* 3ms G4* 3ms
```

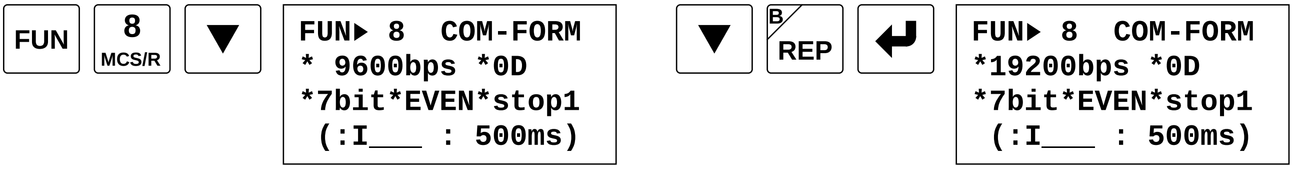
While the cursor is at G1 through G4, pressing the REP key toggles 0 ms, 3 ms, 7 ms, and 10 ms to select the soft filter preset value.
To move from a group to another, press the ▼ or ▲ key.

The default selection of the hard filter is 10.
The default selection of soft filter is 3 msec for all four groups.

To return to the editor mode, press the CLR key.

FUN8: Loader Port Communication Mode Setting

The MICRO^3 or MICRO^{3}C base unit can communicate with a personal computer or modem through the RS485 loader port (MICRO^3) or the RS232C loader port (MICRO^{3}C). For this, the communication format can be selected from the standard or optional mode. When the mode selection input selected by FUN8 is turned on, the optional communication mode is enabled. When using the program loader to communicate with the MICRO^3 or MICRO^{3}C base unit, use the standard communication mode of all default values.



```
FUN▶ 8  COM-FORM
* 9600bps *0D
*7bit*EVEN*stop1
 (:I___ : 500ms)
```

```
FUN▶ 8  COM-FORM
*19200bps *0D
*7bit*EVEN*stop1
 (:I___ : 500ms)
```

Pressing the REP key toggles the options for each parameter.
To move the cursor from a parameter to another, press the ▼ or ▲ key.

To restore the default value, move the cursor to the parameter, and press the DEL key.

Pressing the DEL key with the cursor placed at the mode input selection on the bottom line, the input selection is cleared, and the optional communication mode is canceled.

Communication Parameter	Option	Default (Standard Mode)
Baud Rate	1200, 2400, 4800, 9600, 19200 bps	9600 bps
Terminator Code	0D (CR), 0D 0A (CR LF)	0D (CR)
Data Bits	7, 8 bits	7 bits
Parity Check	None, Even, Odd	Even
Stop Bits	1, 2 bits	1 bit
Mode Selection Input (Note 1)	I0 to I15	None
Receive Timeout (Note 2)	10 to 2550 (10-msec increments)	500 msec

To return to the editor mode, press the CLR key.

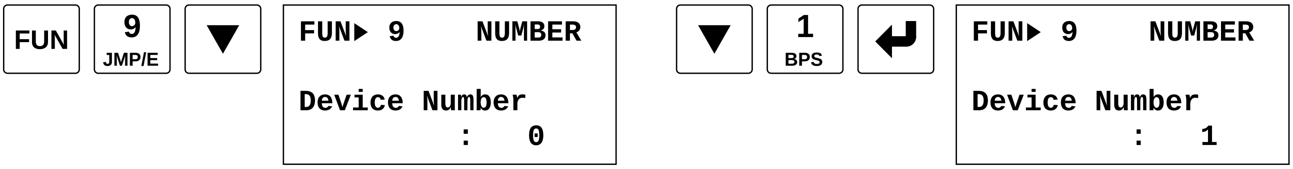
Note 1: When the protocol selector switch is set to 1 or 3 to select user protocol for the loader port on the MICRO^{3}C, the mode selection input is not used and need not be specified to enable the FUN8 value.

Note 2: When 2550 is selected, receive timeout is disabled in the user communication of the MICRO^{3}C.

FUN9: PLC Address for Network Communication

The MICRO^3 or MICRO^{3}C base unit can be connected to a personal computer through the RS485 loader port (MICRO^3) or the RS485 data link terminals (MICRO^{3}C) for a 1:N communication computer link network. When used in a network, the PC must have addresses to differentiate various MICRO^3 or MICRO^{3}C base units that it communicates with.

Each MICRO^3 or MICRO^{3}C in a network can be allocated to a unique number from 0 through 31, using FUN9. A PC will communicate with the MICRO^3 or MICRO^{3}C base units in a network, addressing to the values that have been allocated in FUN9:



```
FUN▶ 9    NUMBER

Device Number
          :   0
```

```
FUN▶ 9    NUMBER

Device Number
          :   1
```

The key sequence above will allocate address 1 to the MICRO^3 or MICRO^{3}C base unit being programmed.
The default selection is device number 0.

To return to the editor mode, press the CLR key.

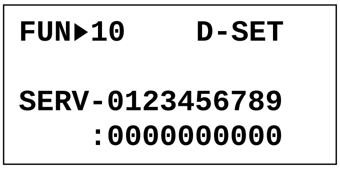
FUN10: Control Data Register Setting

FUN10 is used to select whether data registers D90 through D99 are enabled as a control data register or not. When the control data register function is enabled, the data register stores various data as shown below. Control data registers are for readout only and can be used in the user program or for monitoring.

The control data register function can be used in the standard processing mode only and cannot be used in the high-speed processing mode.

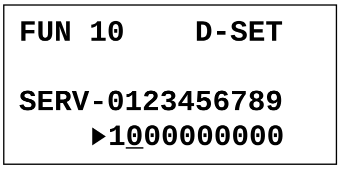
When the control data register function is disabled, the data register can be used as an ordinary data register.

FUN | 1 BPS | 0 | ▼



```
FUN▶10    D-SET

SERV-0123456789
    :0000000000
```

▼ | 1 BPS



```
FUN 10    D-SET

SERV-0123456789
    ▶1000000000
```

Control data registers D90 through D99 correspond to 0 through 9 on the right of SERV-. Press the 1 or 0 key to enable or disable the control data register function.

1: Enables the control data register function.
0: Disables the control data register function.
DEL: Sets 1 at all positions to enable all control data registers.
CLR: Clears all changes and restores the previous settings.
▶ or ◀: Moves the cursor.
↵: Ends the setting.

SERV Number	Data Register	Function	Data in Control Data Register
0	D90	Base Unit System Code	MICRO^3 base unit system code 1: 24-I/O type 2: 16-I/O type 4: 10-I/O type 8: AC input type 16: Protect source output type
1	D91	Base Unit Processing Mode Code	MICRO^3 base unit processing mode code 0: Standard processing mode 1: High-speed processing mode
2	D92	General Error Code	See General Error Code on page 18-2
3	D93	User Program Execution Error Code	See User Program Execution Error on page 18-5
4	D94	Link Communication Error Code	See Link Communication Error on page 18-5
5	D95 (Note)	Day of Week (Calendar)	0: Sunday, 1: Monday, 2: Tuesday, 3: Wednesday, 4: Thursday, 5: Friday, 6: Saturday
6	D96 (Note)	Hour (Clock)	0 to 23
7	D97 (Note)	Minute (Clock)	0 to 59
8	D98 (Note)	Second (Clock)	0 to 59
9	D99	Scan Time (Current Value)	Current scan time in msec See FUN25 on page page 5-11

The default setting is to disable all control data register functions.

Note: Since the 10-I/O type MICRO^3 base unit does not have the real-time calendar/clock function, data registers D95 through D98 cannot be designated as control data registers and are always used as ordinary data registers. Only 16-I/O and 24-I/O type MICRO^3 base units can designate D95 through D98 as control data registers.

To return to the editor mode, press the CLR key.

FUN11: Program Capacity and PLC Type Selection

The user program capacity can be selected from 244, 500, or 1K steps using FUN11:

FUN | 1 BPS | 1 BPS | ▼

```
FUN▶11    SIZE
Program Capacity
     *  1K(step)
           :  0
```

Program capacity (must be set to 244, 500, or 1K)

PLC type code (must be set to 0 or 1)
0: MICRO^3
1: MICRO^{3}C

▼ | B REP | B REP | B REP

▼ | 1 BPS | ↵

```
FUN▶11    SIZE
Program Capacity
     * 244(step)
           :  1
```

Pressing the REP key changes the program capacity from1K to 4K, 8K, 244, 500, and back to 1K. Do not select 4K and 8K because MICRO^3 and MICRO^{3}C can run user programs of 244, 500, and 1K steps only. The initial setting is 1K steps.

The fourth line in the FUN11 screen shows the PLC type code. Do not select other than 0 and 1. When the PLC type code is set to 0 to select MICRO^3, data registers D100 through D499 cannot be programmed.

To return to the editor mode, press the CLR key.

FUN20: PLC Error Data Readout and Reset

The error data stored in the MICRO^3 base unit can be read out and reset using FUN20:

FUN | 2 BRD | 0 | ▼

```
FUN▶20 ERROR

   Nothing
```

```
FUN▶20 ERROR 401
  Power Off
  24V Overload
```

When no error is detected, “Nothing” is displayed as shown above. If there is an error, then the error code and error message are shown as on the right above. For details of error messages, see page 18-1. To clear the error display:

DEL

```
FUN▶20 ERROR 401

   Delete?
```

↵

```
FUN▶20 ERROR

   Nothing
```

To return to the editor mode, press the CLR key.

FUN21: Timer/Counter Preset Value Readout and Restore

Timer and counter preset values in the MICRO^3 base unit can be changed from the monitor mode without changing the entire program. See page 3-14. FUN21 can be used to see if T/C preset values have been changed. When preset values are not changed, “Nothing” is displayed. When timer and counter preset values have been changed, the changed values can be read from the MICRO^3 base unit RAM to the program loader using FUN21. The changed preset values in the MICRO^3 base unit RAM can also be cleared to restore the original preset values using FUN21:

FUN | 2 BRD | 1 BPS | ▼

```
FUN▶21   T/C CHG

CHG data of T/C
       *READ OK?
```

▼

```
FUN 21   T/C CHG

CHG data of T/C
       ▶READ OK?
```

Press the REP key to select “READ OK?” or “CLR OK?”

READ: Reads the changed timer/counter preset values from the MICRO^3 base unit RAM to the program loader.

CLR: Clears the changed preset values and restores the original preset values in the MICRO^3 base unit RAM.

To execute the command, press the ↵ key.
To return to the editor mode, press the CLR key.

FUN22: User Program Protection

The user program in the MICRO^3 base unit can be protected from reading, writing, or both using FUN22.

⚠ **Warning**

- When proceeding with the following steps, make sure to note the password, which is needed to cancel the program protection. If the user program in the MICRO^3 base unit is write- or read/write-protected, then the user program cannot be changed without the password.

To protect a user program, first transfer the user program from the program loader to the MICRO^3 base unit (see page 3-9) and send a password to the MICRO^3 base unit by pressing the keys:

FUN | 2 BRD | 2 BRD | ▼

```
FUN▶22    PASS

Pass Word :____
 (Mode *    )
```

▼

```
FUN 22    PASS

Pass Word ▶____
 (Mode *    )
```

Enter a password of 1 through 4 digits using the 0 through 9 and A through F keys:

0 through 9 JMP/E   A NOT through F RST Q

```
FUN 22    PASS

Pass Word ▶91AF
 (Mode *    ) OK?
```

Move down the cursor to the asterisk, and press the REP key repeatedly to select a protection mode from read, write, or read/write protection.

▼ | B REP

```
FUN 22    PASS

Pass Word :91AF
 (Mode ▶R/ ) OK?
```

`R/:`	Read protect
`/W:`	Write protect
`R/W:`	Read and write protect

To execute the protect command, press the ↵ key. When the program protection is completed, “END” is displayed.

To cancel the program protection in the MICRO^3 base unit, first read FUN22 and move down the cursor:

FUN | 2 BRD | 2 BRD | ▼

```
FUN▶22    PASS

Pass Word :■■■■
 (Mode *R/W)
```

▼

```
FUN 22    PASS

Pass Word ▶■■■■
 (Mode *R/W)
```

Enter the password correctly and press the ↵ key. When the program protection is canceled successfully, “END” is displayed. If the password is incorrect, “Pass Word NG” is displayed, and the protection is not canceled.

To return to the editor mode, press the CLR key.

FUN23: PLC System Program Version Readout

The system program version in the MICRO^3 base unit can be read using FUN23. The system version is displayed in 4 digits.

To return to the editor mode, press the CLR key.

FUN | 2 BRD | 3 BPP | ▼

```
FUN▶23   Ver.No.

PC System
           Ver.0002
```

FUN24: PLC Operating Status Readout

MICRO^3 operating status can be read using FUN24.

`Run:`	MICRO^3 is running.
`Stop:`	MICRO^3 is stopped.

To return to the editor mode, press the CLR key.

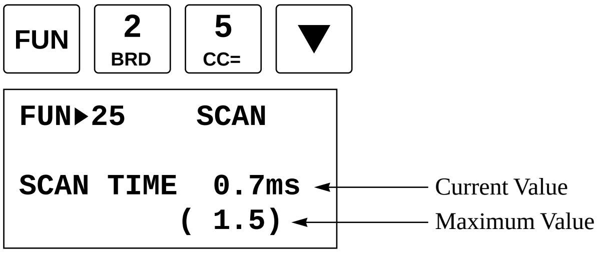
FUN | 2 BRD | 4 | ▼

```
FUN▶24    STATE

State     (Run)
```

FUN25: Scan Time Readout

The scan time of the user program in the MICRO^3 base unit can be read using FUN25.

The current value of the scan time is displayed in units of msec and updated periodically. The maximum value of the scan time is shown in parentheses on the bottom line.



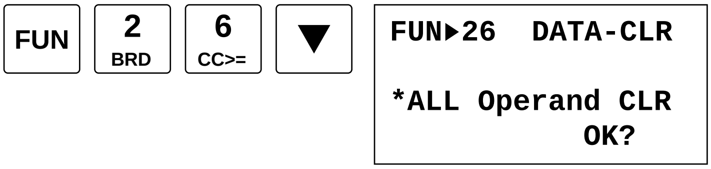
```
FUN▶25    SCAN

SCAN TIME  0.7ms
          ( 1.5)
```

The current and maximum values of the scan time are displayed in the decimal notation for the integer and in the octal notation for the fraction. In the example on the right, the actual current value is 0.7 × 1.25 = 0.875 msec, and the maximum value is 1 + 0.5 × 1.25 = 1.625 msec.

To return to the editor mode, press the CLR key.

The current value of the scan time is stored in control data register D99, if enabled using FUN10. See page 5-8.

FUN26: Operand Data Clear



```
FUN▶26  DATA-CLR

*ALL Operand CLR
           OK?
```

The data of all or selected operand can be cleared using FUN26.

To clear the data of all operands, press the ↵ key.

To select an operand, move the cursor down, and press the REP key. Press the ↵ key to clear the data of the selected operand.

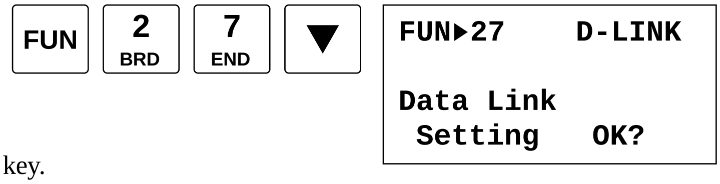
When the data is cleared, “END” is displayed.

When this command is executed the following data is cleared.

Selection	Operand	Data Cleared
ALL	All operands (Q, M, R, C, T, D)	All data shown below are cleared.
Q	Output	All outputs are turned off.
M	Internal relay	All internal relays are turned off.
R	Shift register	All shift register bits are turned off.
C	Counter	CNT0 and 1: Current values are cleared to the preset value. CNT2 to 31: Current values are cleared to zero.
T	Timer	All timer current values are cleared.
D	Data register	Data of all data registers are cleared to zero.

To return to the editor mode, press the CLR key.

FUN27: Link Formatting Sequence



```
FUN▶27    D-LINK

Data Link
 Setting   OK?
```

When the expansion link or data link configuration is changed, execute the link formatting sequence from the master station using FUN27 to initialize the link communication line.

To initialize the link communication line, press the ↵ key.

The MICRO^3 base unit with the function selector switch set to zero is used for the base station in the expansion link system or the master station in the data link system. When the base or master station is powered up, the base or master station sends the link formatting sequence to confirm whether the expansion station or slave stations can be communicated with. If the expansion station or any slave station is not powered up, the expansion or slave station is not recognized. To enable communication with this expansion or slave station, power up the station, and execute FUN27 from the base or master station to initialize the link communication line.

To return to the editor mode, press the CLR key.

FUN28: Calendar/Clock Data Readout and Setting

The calendar/clock data in the 16- and 24-I/O type MICRO^3 base units can be read and changed using FUN28.

FUN | 2 BRD | 8 MCS/R | ▼

```
FUN▶28  CALENDAR
 Y:’97 M: 3 D: 4
  H:17 M:15 S:34
   *(Tue)
```

The calendar/clock data is displayed.

Y: Year (the last 2 digits)
M: Month (1 through 12 correspond to January through December)
D: Day (1 through 31)
H: Hours (0 through 23)
M: Minutes (0 through 59)
S: Seconds (0 through 59)
Day of week in parentheses

To change the calendar/clock data, press the ▼ key to move the cursor and enter new data using the 0 through 9 keys. To change the day of week, move the cursor to the asterisk, and press the REP key repeatedly to display the required day.

To update the calendar/clock data, press the ↵ key. When completed, “END” is displayed.
To return to the editor mode, press the CLR key.

FUN29: User Communication Status Readout

◆ *MICRO^{3}C Only* ◆

User communication error data, execution of transmit/receive instructions, and communication parameters can be read using FUN29 on the program loader. For details on FUN29, see the MICRO^{3}C User’s Manual.

FUN30: Program Check

The user program in the program loader memory can be checked using FUN30. When the user program is correct, “Program OK” is displayed. When any program error is found, an error message and its address is displayed.

FUN | 3 BPP | 0 | ▼

```
FUN▶30    CHECK1

  Program  OK
```

```
FUN▶30    CHECK1
  FUN  Error   1
  OP.  Error 100
  MCR  Error 200
```

When more than 3 errors are found, scroll the display using the ▼ and ▲ keys.
To return to the editor mode, press the CLR key.

When executing FUN30, the following error message may be displayed, followed by the address where the error is found.

Error Message	Error Details
BPS Error + Address	BPS instruction is not followed by BPP instruction.
BPS Over ? + Address	More than 8 BPS instructions are programmed continuously, overflowing the bit stack registers.
BPP Error + Address	BPP instruction is not preceded by BPS instruction.
BRD Error + Address	BRD instruction is not preceded by BPS instruction.
END Error	END instruction is not found.
FUN Error + Number	Invalid FUN setting. The incorrect FUN number is displayed.
JEND Error + Address	JEND instruction is not preceded by JMP instruction.
JMP Error + Address	JMP instruction is not followed by JEND instruction. (Note)
MCR Error + Address	MCR instruction is not preceded by MCS instruction.
MCS Error + Address	MCS instruction is not followed by MCR instruction. (Note)
No Program	END instructions exist at all addresses.
OP. Error + Address	Invalid operand number.
PRG. Error + Address	User program is broken.

When any error was found, correct the program, and execute FUN30 again to make sure of no program error.

Note: These messages are displayed for warning only. The JMP or MCS instruction can also be used with the END instruction, instead of JEND or MCR. User programs run correctly even if the JEND or MCR is not programmed.

FUN31: Program Loader Version Readout/Hardware Check

At the beginning of the hardware check, using FUN31, the version of the program loader system program is displayed. Next, the program loader display is tested, and the internal memory is checked by testing the readout and writing functions of the entire RAM of the program loader. If all functions are normal, the message on the right below is displayed.

FUN | 3 BPP | 1 BPS | ▼

```
FUN▶31    CHECK2

Loader System
          Ver 2.00
```

```
FUN▶31    CHECK2

--TEST      OK--
(■■■■■■■■■■■■■■)
```

To abort the program loader hardware check and return to the editor mode, press the CLR key.

FUN32: Sequential Monitoring

When sequential monitoring is set to ON using FUN32, the ON/OFF statuses of operands can be monitored in the editor mode. Sequential monitoring is possible at addresses of LOD, LODN, AND, ANDN, OR, ORN, OUT, OUTN, SET, RST, TIM, and CNT instructions.

The ON/OFF status of the operand is displayed at the right end of the line.

The statuses of inputs, outputs, internal relays, and shift register bits are shown as follows:

■: ON
□: OFF

The statuses of timers and counters are shown as follows:

■: ON (timed out or counted out)
□: OFF (during timing or counting)

FUN | 3 BPP | 2 BRD | ▼

```
FUN▶32   MONITOR
  Sequential
     Monitoring
         *OFF
```

▼ | B REP | ↵

```
FUN▶32   MONITOR
  Sequential
     Monitoring
         *ON
```

CLR

```
   5▶LOD   I  1■
   6 LODN  M 10□
   7 OR LOD
   8 TIM 1   10□
```

At addresses where the NOT instruction is programmed, the ON/OFF display is reversed.

FUN33: Monitor Screen Holding

Once the monitor operands are selected in the monitor mode, the monitor screen is usually restored after returning from another mode. The monitor screen can also be set to be cleared when returning from another mode using FUN33.

KEEP: Retains the monitor screen selections.
CLR: Clears the monitor screen selections.

To return to the editor mode, press the CLR key.

FUN | 3 BPP | 3 BPP | ▼

```
FUN▶33  MON-KEEP

Monitor Setting
         *KEEP
```

▼ | B REP | ↵

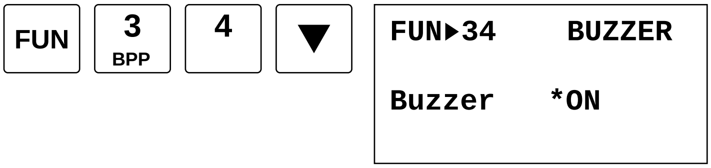
```
FUN▶33  MON-KEEP

Monitor Setting
         *CLR
```

FUN34: Program Loader Beep Sound

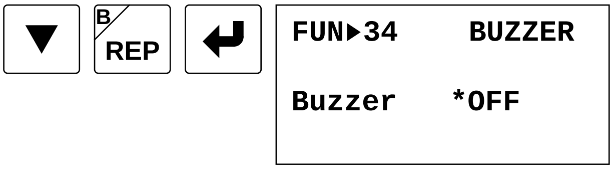
The program loader usually beeps to acknowledge each key input and signal an error. The buzzer can also be silenced using FUN34.

ON: Buzzer beeps
OFF: Buzzer silenced

To return to the editor mode, press the CLR key.



```
FUN▶34    BUZZER

Buzzer   *ON
```



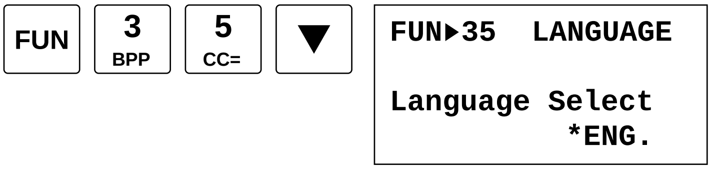
```
FUN▶34    BUZZER

Buzzer   *OFF
```

FUN35: Display Language Selection

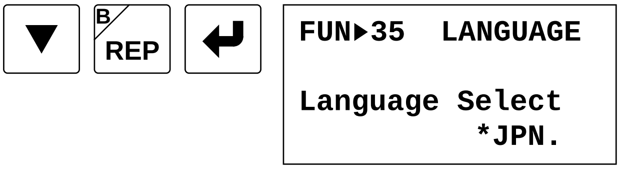
The program loader usually displays messages in English. The messages can also be displayed in Japanese using FUN35.

ENG.: English
JPN.: Japanese

To return to the editor mode, press the CLR key.



```
FUN▶35  LANGUAGE

Language Select
          *ENG.
```



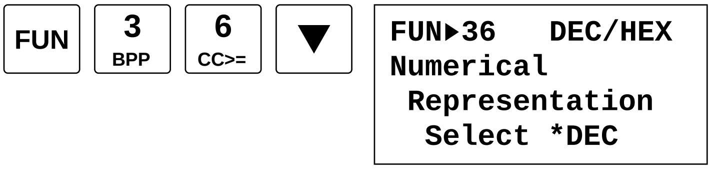
```
FUN▶35  LANGUAGE

Language Select
          *JPN.
```

FUN36: Display Data Type Selection

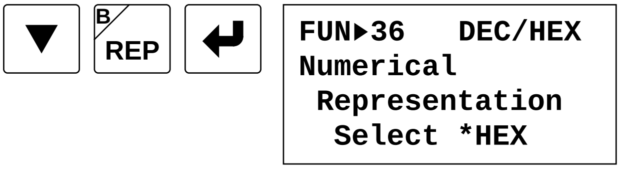
High-speed counter preset and current values can be displayed either in decimal or hexadecimal notation in the editor and monitor screens. Data of data registers can also be displayed in either decimal or hexadecimal notation in the monitor mode. The selection can be made using FUN36.

DEC: Decimal
HEX: Hexadecimal

To return to the editor mode, press the CLR key.



```
FUN▶36   DEC/HEX
Numerical
 Representation
  Select *DEC
```



```
FUN▶36   DEC/HEX
Numerical
 Representation
  Select *HEX
```

FUN40: Memory Card Identification

The memory card installed in the program loader can be identified using FUN40.

FUN 4 0 ▼

```
FUN▶40    CARD
  Battery  OK
  Name(IDEC    )
⬇ Free( 26Kstep)
```

The second line shows the condition of the battery in the memory card:

OK:	Battery works normally.
LOW:	Replacing the battery is recommended.
NG:	Data cannot be kept. Replace the battery.

▼

```
FUN▶40    CARD
⬆ 1K 1(PROGRAM1)
  1K 2(PROGRAM2)
⬇ 1K 3(PROGRAM3)
```

The memory in the FC2A-MC1 memory card is backed up for approximately 4 years.

The third line shows the name of the memory card for storing user programs.

The bottom line shows the available memory capacity of the user program memory card or the type of the memory card.

Free ( ):	The card is a user program memory card. Available memory capacity is shown in parentheses.
Unformat Card:	The card is not formatted. Format the card using FUN41.
System Card:	The card is a system memory card. The system number is displayed on the right.
Unrecognized One:	Not a MICRO³ memory card.

When a user program memory card is inserted, pressing the ▼ key shows the capacities, addresses, and names of user programs stored in the memory card. To scroll the pages, press the ▼ and ▲ keys.

To return to the editor mode, press the CLR key.

FUN41: Memory Card Formatting

A new memory card must be formatted using FUN41 to store user programs.

FUN 4 1 BPS ▼

```
FUN▶41    FORMAT

 Name:(        )
 Card Format OK?
```

Install a new memory card into the program loader. If the memory card stores user programs, formatting deletes all programs from the card.

After calling the FUN41 screen, move down the cursor to the colon and enter a card name of 8 characters maximum.

▼ ▲ or ▼

▶ or ◀

```
FUN 41    FORMAT

 Name▶(IDEC    )
 Card Format OK?
```

Applicable characters are A through Z, 0 through 9, and a space.

After entering the card name, press the ↵ key to start formatting. If no card name is entered, formatting is not started.

↵

```
FUN▶41    FORMAT

 Name:(IDEC    )
 Card Format END
```

When formatting is complete, “END” is displayed.

To return to the editor mode, press the CLR key.

FUN42: Program Loader System Program Installation

A new system program can be installed into a separate memory area in the program loader using FUN42.

FUN 4 2 BRD ▼

```
FUN▶42   SYS-SET
 Extension
  System  Read
       OK?
```

Install a system program card into the program loader. After calling the FUN42 screen, press the ↵ key to start.

When the program loader is powered up again, the new system program is booted.

To use the original system program, depress and hold the CLR key, and power up the program loader.

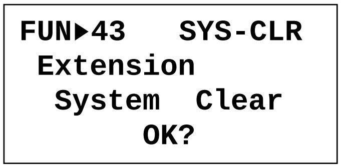
To return to the editor mode, press the CLR key.

Note: If the program loader is connected to the MICRO³ when installing a new system program, stop MICRO³ operation; otherwise, the program cannot be installed because of insufficient power supplied from the MICRO³.

FUN43: Program Loader System Program Restore

The new system program installed using FUN42 can be cleared, and the original system program of the program loader can be restored using FUN43.

FUN 4 3 BPP ▼



```
FUN▶43   SYS-CLR
 Extension
  System  Clear
       OK?
```

After calling the FUN43 screen, press the ↵ key to start.

To return to the editor mode, press the CLR key.

Note: If the program loader is connected to the MICRO^3 when clearing the new system program, stop MICRO^3 operation; otherwise, the program cannot be cleared because of insufficient power supplied from the MICRO^3.

FUN50: User Communication Data Monitor

◆ MICRO^{3}C Only ◆

While the MICRO^{3}C is communicating through the loader port using the user protocol, the transmit and receive data of user communication between the MICRO^{3}C and RS232C equipment can be monitored using FUN50 on the program loader connected to the data link terminals. The communication monitor functions are useful for debugging user communication programs.

For details on FUN50, see the MICRO^{3}C User's Manual.

Introduction

◆ *For the MICRO^{3}C, see the MICRO^{3}C User's Manual.* ◆

This chapter describes allocation numbers available for the MICRO^3 base unit to program basic and advanced instructions. Special internal relays are also described.

The MICRO^3 programmable controller is programmed using operands such as inputs, outputs, internal relays, timers, counters, shift registers, and data registers.

Inputs (I) are relays to receive input signals through the input terminals.
Outputs (Q) are relays to send the processed results of the user program to the output terminals.
Internal relays (M) are relays used in MICRO^3 and cannot be outputted to the output terminals.
Special internal relays (M) are internal relays dedicated to specific functions.
Timers (T) are relays used in the user program, available in 100-msec, 10-msec, and 1-msec timers.
Counters (C) are relays used in the user program, available in reversible counters and adding counters.
Shift registers (R) are registers to shift the data bits according to pulse inputs.
Data registers (D) are registers used to store numerical data. Some of the data registers are dedicated to special functions.

Allocation Numbers

Available I/O numbers depend on the type and combination of the MICRO^3 base units used in the expansion link system. For details of available I/O numbers in the expansion link system, see page 4-7.

Operand	Processing Mode	Allocation Number	Maximum Points
Input	Standard and High-speed	I0 - I7 I10 - I15	14 points (Base) + 14 points (Expansion)
	Standard only	I20 - I27 I30 - I35	
Output	Standard and High-speed	Q0 - Q7 Q10 - Q11	10 points (Base) + 10 points (Expansion)
	Standard only	Q20 - Q27 Q30 - Q31	
Internal Relay	Standard and High-speed	M0 - M7 M10 - M17 M20 - M27 M30 - M37 M40 - M47	232 points (40 points)
	Standard only	M50 - M57 M60 - M67 M70 - M77 M80 - M87 M90 - M97 M100 - M107 M110 - M117 M120 - M127 M130 - M137 M140 - M147 M150 - M157 M160 - M167 M170 - M177 M180 - M187 M190 - M197 M200 - M207 M210 - M217 M220 - M227 M230 - M237 M240 - M247 M250 - M257 M260 - M267 M270 - M277 M280 - M287	
Catch Input Relay	Standard and High-speed	M290 - M297	8 points (8 points)
Special Internal Relay	Standard and High-speed	M300 - M307 M310 - M317	16 points (16 points)
Timer	Standard and High-speed	T0 - T15	32 points total (16 points total)
	Standard only	T16 - T31	
Counter	Standard and High-speed	C0 - C15	
	Standard only	C16 - C31	
Shift Register	Standard and High-speed	R0 - R31	64 points (32 points)
	Standard only	R32 - R63	
Data Register	Standard and High-speed	D0 - D31	100 points (32 points)
	Standard only	D32 - D99	

Notes:

Input and output allocation numbers for the expansion station start with I20 and Q20. Note that input and output allocation numbers are not continuous between the base station and expansion station in the expansion link system.
The maximum points shown in ( ) are values for the high-speed processing mode.
The same number cannot be used for a counter and a timer in a user program.

Special Internal Relays

Internal relays M290 through M317 are special internal relays with the following functions:

Allocation Number	Description		CPU Stopped	Power OFF
M290	Catch Input Status Set (See Note below)	Input I0	Operating	Cleared
M291		Input I1	Operating	Cleared
M292		Input I2	Operating	Cleared
M293		Input I3	Operating	Cleared
M294		Input I4	Operating	Cleared
M295		Input I5	Operating	Cleared
M296		Input I6	Operating	Cleared
M297		Input I7	Operating	Cleared
M300	Start Control		Maintained	Maintained
M301	Initialize Pulse (See Note below)		Cleared	Cleared
M302	All Outputs OFF		Cleared	Cleared
M303	Carry (Cy) or Borrow (Bw)		Cleared	Cleared
M304	User Program Execution Error		Cleared	Cleared
M305	Link Communication Error (Expansion mode and data link mode)		Maintained	Cleared
M306	Link Communication Prohibit Flag (Expansion mode and data link mode)		Maintained	Maintained
M307 (Ver. 6 or later)	Link Communication Initialize Flag (Master Station) (Expansion mode and data link mode)		Cleared	Cleared
	Link Communication Stop Flag (Slave Station) (Data link mode)			
M307 (Ver. 5 or earlier)	Link Communication Stop Flag (Slave Station) (Data link mode)		Cleared	Cleared
M310	1-sec Clock Reset		Cleared	Cleared
M311	1-sec Clock		Operating	Cleared
M312	100-msec Clock		Operating	Cleared
M313	10-msec Clock		Operating	Cleared
M314	Timer/Counter Preset Value Changed		Maintained	Maintained
M315	High-speed Counter Soft Reset		Maintained	Cleared
M316	High-speed Counter (HSC3) Overflow		Cleared	Cleared
M317	In-operation Output		Cleared	Cleared

Note: M290 through M297 and M301 are used only for reading in the user program, but can be directly set or reset using the program loader or optional software CUBIQ on a computer.

M290 to M297 Catch Input Status Set

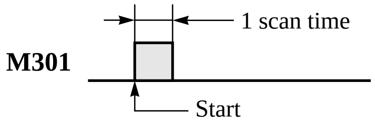
When a rising or falling input edge is detected during a scan, the input statuses of inputs I0 through I7 at the moment are set to M290 through M297 without regard to the scan status. Only one edge is detected in one scan. For the catch input function, see page 4-2.

M300 Start Control

MICRO^3 starts operation when M300 is turned on and stops operation when M300 is turned off. M300 can be turned on or off using the RUN/STOP switch on the program loader. When a stop or reset input is designated, M300 must remain on to control the MICRO^3 operation using the stop or reset input. For the start and stop operation, see page 2-1.

M301 Initialize Pulse

When MICRO^3 starts operation, M301 turns on for a period of one scan.



M302 All Outputs OFF

When M302 is turned on, all outputs (Q0 through Q31) go off until M302 is turned off. Self-maintained circuits using outputs also go off and are not restored when M302 is turned off.

M303 Carry (Cy) and Borrow (Bw)

When a carry or borrow results from executing an addition or subtraction instruction, M303 turns on. M303 is also used for the bit shift and rotate instructions. See pages 11-1 and 13-1.

M304 User Program Execution Error

When an error occurs during executing a user program, M304 turns on. The cause of the user program execution error can be checked using FUN20 on the program loader. See page 18-5.

M305 Link Communication Error

When an error occurs during communication in the expansion link or data link system, M305 turns on. The M305 status is maintained when the error is cleared and remains on until M305 is reset using the program loader or until MICRO^3 is turned off. The cause of the link communication error can be checked using FUN20 on the program loader. See page 18-5.

M306 Link Communication Prohibit Flag

When M306 at the base or master station is turned on in the expansion link or data link system, communication is stopped. The M306 status is maintained when MICRO^3 is turned off and remains on until M306 is reset using the program loader.

M307 Link Communication Initialize Flag (Master Station)/Link Communication Stop Flag (Slave Station)

Special internal relay M307 has different functions depending on the PLC system program version. With version 6 or later, M307 can be used at the base station in the expansion link system or at the master station in the data link system. With version 5 or earlier, M307 cannot be used at the base or master station. With either version of the system program, M307 has the same function at the slave station in the data link system and has no effect at the expansion station in the expansion link system. To check the PLC system program version, see FUN23 PLC System Program Version Readout on page 5-11.

Base or master station: Link communication initialize flag (Ver. 6 or later only)

When M307 at the base or master station is turned on during operation, the link configuration is checked to initialize the expansion or data link system. When an expansion station or slave station is powered up after the base or master station, turn M307 on to initialize the link system. After an expansion link or data link setup is changed, M307 must also be turned on to ensure correct communication.

Slave station: Link communication stop flag

When a slave station does not receive communication data from the master station for 800 msec or more in the data link system, M307 turns on. When the slave station receives correct communication data, M307 turns off.

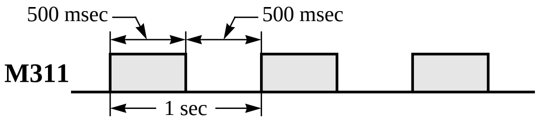
In the expansion station, M307 has no effect and cannot be monitored using the program loader.

M310 1-sec Clock Reset

When M310 is on, M311 (1-sec clock) is reset to zero.

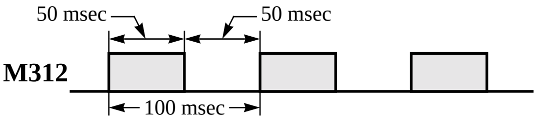
M311 1-sec Clock

When M310 is off, M311 generates clock pulses in 1-sec increments, with a duty ratio of 1:1 (500 msec on and 500 msec off).



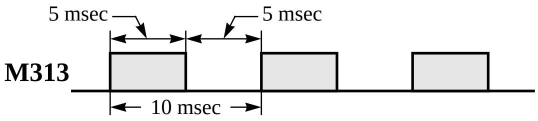
M312 100-msec Clock

M312 always generates clock pulses in 100-msec increments, whether M310 is on or off, with a duty ratio of 1:1 (50 msec on and 50 msec off).



M313 10-msec Clock

M313 always generates clock pulses in 10-msec increments, whether M310 is on or off, with a duty ratio of 1:1 (5 msec on and 5 msec off).



M314 Timer/Counter Preset Value Changed

When timer or counter preset values are changed in the MICRO^3 base unit RAM using the program loader, M314 turns on. When a user program is transferred to MICRO^3 from the program loader or when the changed timer/counter preset value is cleared using FUN21, M314 turns off.

M315 High-speed Counter Soft Reset

When M315 is turned on while a high-speed counter (HSC0, HSC1, HSC2, or HSC3) is used, the high-speed counter current value is reset to zero. When M315 is turned off, the high-speed counter restarts to count.

M316 High-speed Counter (HSC3) Overflow

When the current value of high-speed counter HSC3 exceeds the maximum value of 65535, M316 is turned on. M316 is turned off when the high-speed counter is reset using the hard or soft reset input.

M317 In-operation Output

M317 remains on while MICRO^3 is running.

Data Register Allocation Numbers

◆ For the MICRO^{3}C, see the MICRO^{3}C User's Manual. ◆

Available data registers are limited in the high-speed processing mode or in the data link system configuration. Some data registers are allocated to special functions in the data link system as shown below. For the data link function, see page 4-8.

Data Register Number	Standard Processing Mode			High-speed Processing Mode
	Other than Data Link	Data Link (Master Station)	Data Link (Slave Station)	
D0 to D31	Available	Available	Available	Available
D32 to D59				Not available
D60 to D84		For data link		
D85 to D89			For data link	
D90 to D99	Can be designated as control data registers using FUN10. See page 5-8.			

Introduction

This chapter describes programming of the basic instructions, available operands, and sample programs.

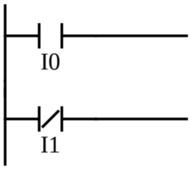
Basic Instruction List

Symbol	Name	Function	Addresses Required
AND	And	Series connection of NO contact	1
AND LOD	And Load	Series connection of circuit blocks	1
ANDN	And Not	Series connection of NC contact	1
BPP	Bit Pop	Restores the result of bit logical operation which was saved temporarily	1
BPS	Bit Push	Saves the result of bit logical operation temporarily	1
BRD	Bit Read	Reads the result of bit logical operation which was saved temporarily	1
CC=	Counter Comparison (=)	Equal to comparison of counter current value	2
CC≥	Counter Comparison (≥)	Greater than or equal to comparison of counter current value	2
CNT	Counter	Adding or reversible counter (0 to 9999)	2
END	End	Ends a program	1
JEND	Jump End	Ends a jump instruction	1
JMP	Jump	Jumps a designated program area	1
LOD	Load	Stores intermediate results, and reads contact status	1
LODN	Load Not	Stores intermediate results, and reads inverted contact status	1
MCS	Master Control Set	Starts a master control	1
MCR	Master Control Reset	Ends a master control	1
NOT	Not	Inversion	—
OR	Or	Parallel connection of NO contacts	1
OR LOD	Or Load	Parallel connection of circuit blocks	1
ORN	Or Not	Parallel connection of NC contacts	1
OUT	Output	Outputs the result of bit logical operation	1
OUTN	Output Not	Outputs the inverted result of bit logical operation	1
RST	Reset	Resets output, internal relay, or shift register bit	1
SET	Set	Sets output, internal relay, or shift register bit	1
SFR	Shift Register	Forward shift register	2
SFRN	Shift Register Not	Reverse shift register	2
SOTD	Single Output	Falling-edge differentiation output	1
SOTU		Rising-edge differentiation output	1
TIM	Timer	Subtracting 100-msec timer (0.1 to 999.9 sec)	2
TMH		Subtracting 10-msec timer (0.01 to 99.99 sec)	2
TMS		Subtracting 1-msec timer (0.001 to 9.999 sec)	2

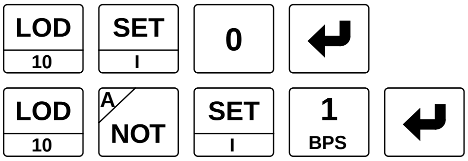
LOD (Load) and LODN (Load Not)

The LOD or LODN instruction is used before an operand starting at the left bus of the ladder diagram. The LOD instruction starts the logical operation with an NO (normally open) contact. The LODN instruction starts the logical operation with an NC (normally closed) contact. Eight LOD and LODN instructions can be used consecutively.

Ladder Diagram



Key Operation



Valid Operands (Standard Processing)

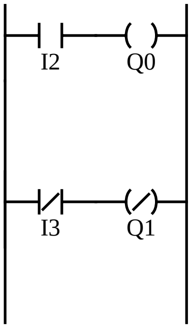
Instruction	I	Q	M	T	C	R
LOD	0-35	0-31	0-317	0-31	0-31	0-63
LODN	0-35	0-31	0-317	0-31	0-31	0-63

When using in the high-speed processing mode, operands are limited. See page 6-1.

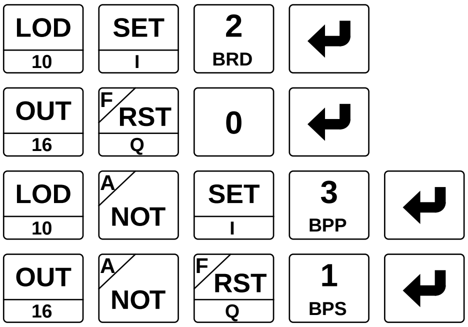
OUT (Output) and OUTN (Output Not)

The OUT instruction outputs the result of bit logical operation to the specified operand. The OUTN instruction outputs the inverted result of bit logical operation to the specified operand.

Ladder Diagram



Key Operation


LOD 10 | SET I | 2 BRD | ↵
OUT 16 | F RST Q | 0 | ↵
LOD 10 | A NOT | SET I | 3 BPP | ↵
OUT 16 | A NOT | F RST Q | 1 BPS | ↵

Valid Operands (Standard Processing)

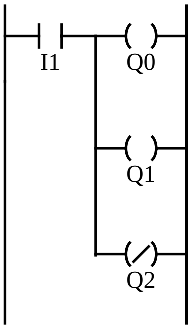
Instruction	I	Q	M	T	C	R
OUT	—	0-31	0-287	—	—	—
OUTN	—	0-31	0-287	—	—	—

When using in the high-speed processing mode, operands are limited. See page 6-1.

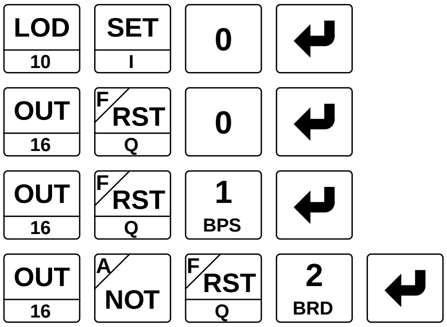
Multiple OUT and OUTN

There is no limit to the number of OUT and OUTN instructions that can be programmed into one rung.

Ladder Diagram



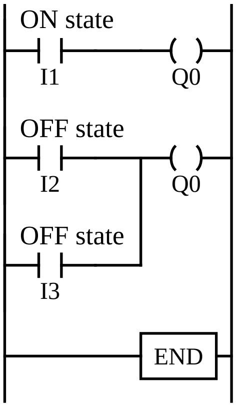
Key Operation


LOD 10 | SET I | 0 | ↵
OUT 16 | F RST Q | 0 | ↵
OUT 16 | F RST Q | 1 BPS | ↵
OUT 16 | A NOT | F RST Q | 2 BRD | ↵

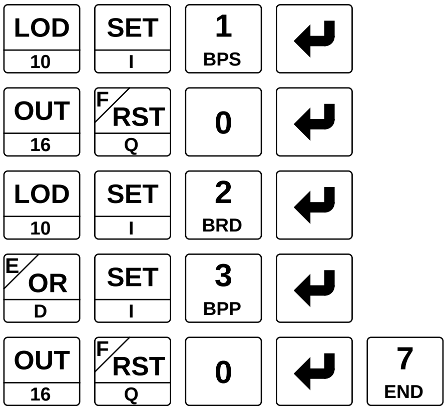
Programming multiple outputs of the same output number is not recommended. However, when doing so, it is good practice to separate the outputs with the JMP/JEND set of instructions, or the MCS/MCR set of instructions. These instructions are detailed later in this chapter.

When the same output address is programmed more than once with one scan, the output nearest to the END instruction is given priority for outputting. In the example on the right, output Q0 is off.

Ladder Diagram

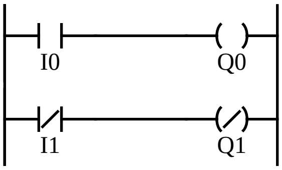


Key Operation


LOD 10 | SET I | 1 BPS | ↵
OUT 16 | F RST Q | 0 | ↵
LOD 10 | SET I | 2 BRD | ↵
E OR D | SET I | 3 BPP | ↵
OUT 16 | F RST Q | 0 | ↵ | 7 END

Examples: LOD (Load), NOT, and OUT (Output)

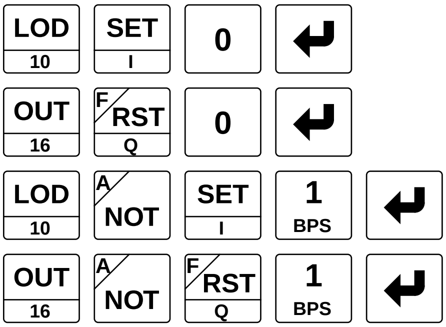
Ladder Diagram



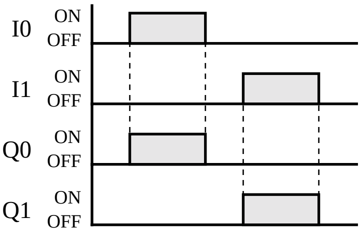
Program List

Prgm Adrs	Instruction	Data
0	LOD	I0
1	OUT	Q0
2	LOD NOT	I1
3	OUT NOT	Q1

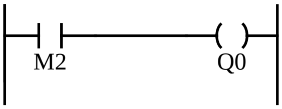
Key Operation



Timing Chart



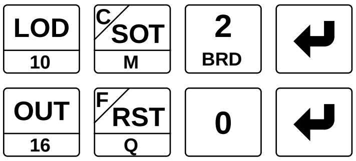
Ladder Diagram



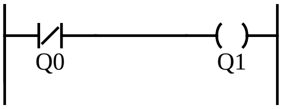
Program List

Prgm Adrs	Instruction	Data
0	LOD	M2
1	OUT	Q0

Key Operation



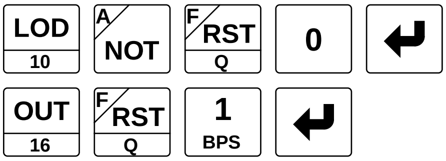
Ladder Diagram



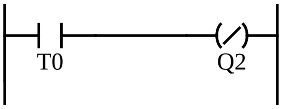
Program List

Prgm Adrs	Instruction	Data
2	LOD NOT	Q0
3	OUT	Q1

Key Operation



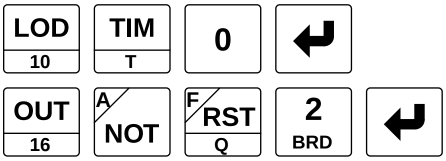
Ladder Diagram



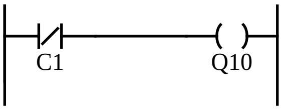
Program List

Prgm Adrs	Instruction	Data
4	LOD	T0
5	OUT NOT	Q2

Key Operation



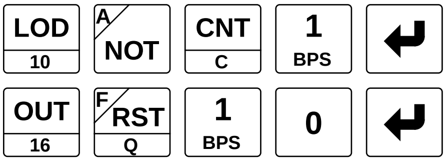
Ladder Diagram



Program List

Prgm Adrs	Instruction	Data
6	LOD NOT	C1
7	OUT	Q10

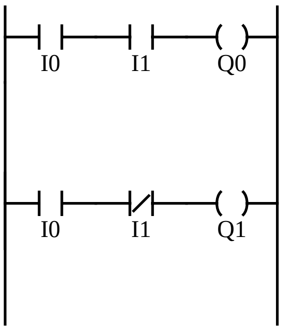
Key Operation



AND and ANDN (And Not)

The AND instruction is used for programming an NO contact in series. The ANDN instruction is used for programming an NC contact in series. The AND or ANDN instruction is entered after the first set of contacts.

Ladder Diagram



Program List

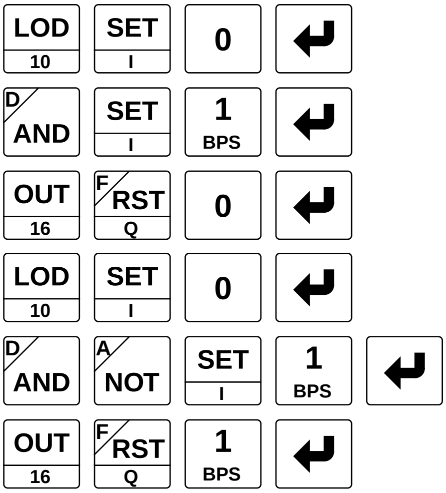
Prgm Adrs	Instruction	Data
0	LOD	I0
1	AND	I1
2	OUT	Q0
3	LOD	I0
4	AND NOT	I1
5	OUT	Q1

Valid Operands (Standard Processing)

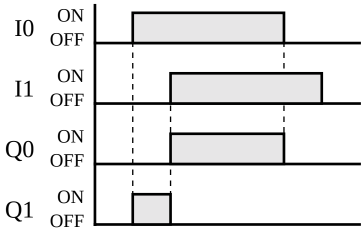
Instruction	I	Q	M	T	C	R
AND	0-35	0-31	0-317	0-31	0-31	0-63
ANDN	0-35	0-31	0-317	0-31	0-31	0-63

When using in the high-speed processing mode, operands are limited. See page 6-1.

Key Operation



Timing Chart



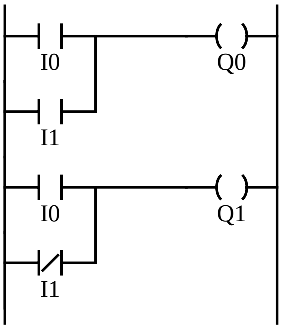
When both inputs I0 and I1 are on, output Q0 is on. When either input I0 or I1 is off, output Q0 is off.

When input I0 is on and input I1 is off, output Q1 is on. When either input I0 is off or input I1 is on, output Q1 is off.

OR and ORN (Or Not)

The OR instruction is used for programming an NO contact in parallel. The ORN instruction is used for programming an NC contact in parallel. The OR or ORN instruction is entered after the first set of contacts.

Ladder Diagram



Program List

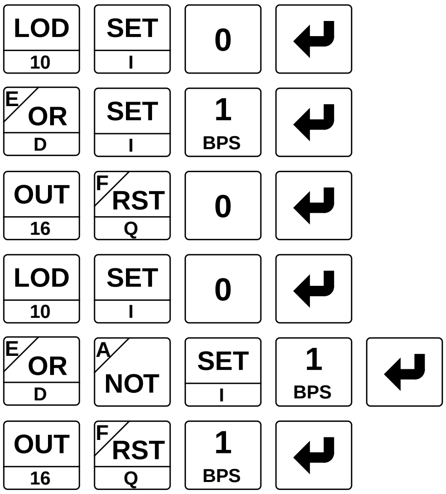
Prgm Adrs	Instruction	Data
0	LOD	I0
1	OR	I1
2	OUT	Q0
3	LOD	I0
4	OR NOT	I1
5	OUT	Q1

Valid Operands (Standard Processing)

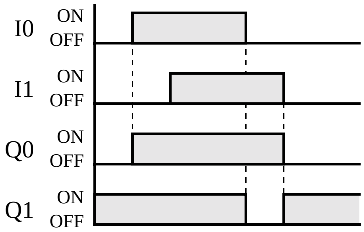
Instruction	I	Q	M	T	C	R
OR	0-35	0-31	0-317	0-31	0-31	0-63
ORN	0-35	0-31	0-317	0-31	0-31	0-63

When using in the high-speed processing mode, operands are limited. See page 6-1.

Key Operation



Timing Chart



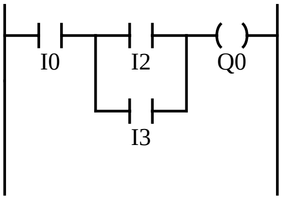
When either input I0 or I1 is on, output Q0 is on. When both inputs I0 and I1 are off, output Q0 is off.

When either input I0 is on or input I1 is off, output Q1 is on. When input I0 is off and input I1 is on, output Q1 is off.

AND LOD (Load)

The AND LOD instruction is used to connect, in series, two or more circuits starting with the LOD instruction. The AND LOD instruction is the equivalent of a “node” on a ladder diagram. The AND LOD instruction is keyed after entering those circuits to be connected.

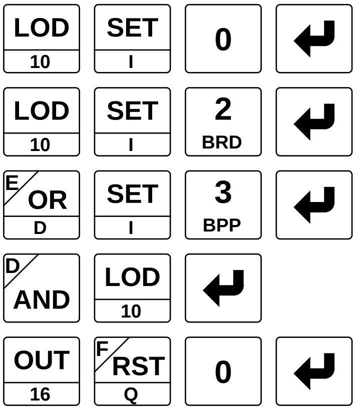
Ladder Diagram



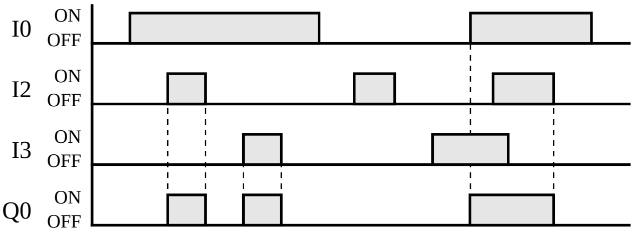
Program List

Prgm Adrs	Instruction	Data
0	LOD	I0
1	LOD	I2
2	OR	I3
3	AND LOD	
4	OUT	Q0

Key Operation

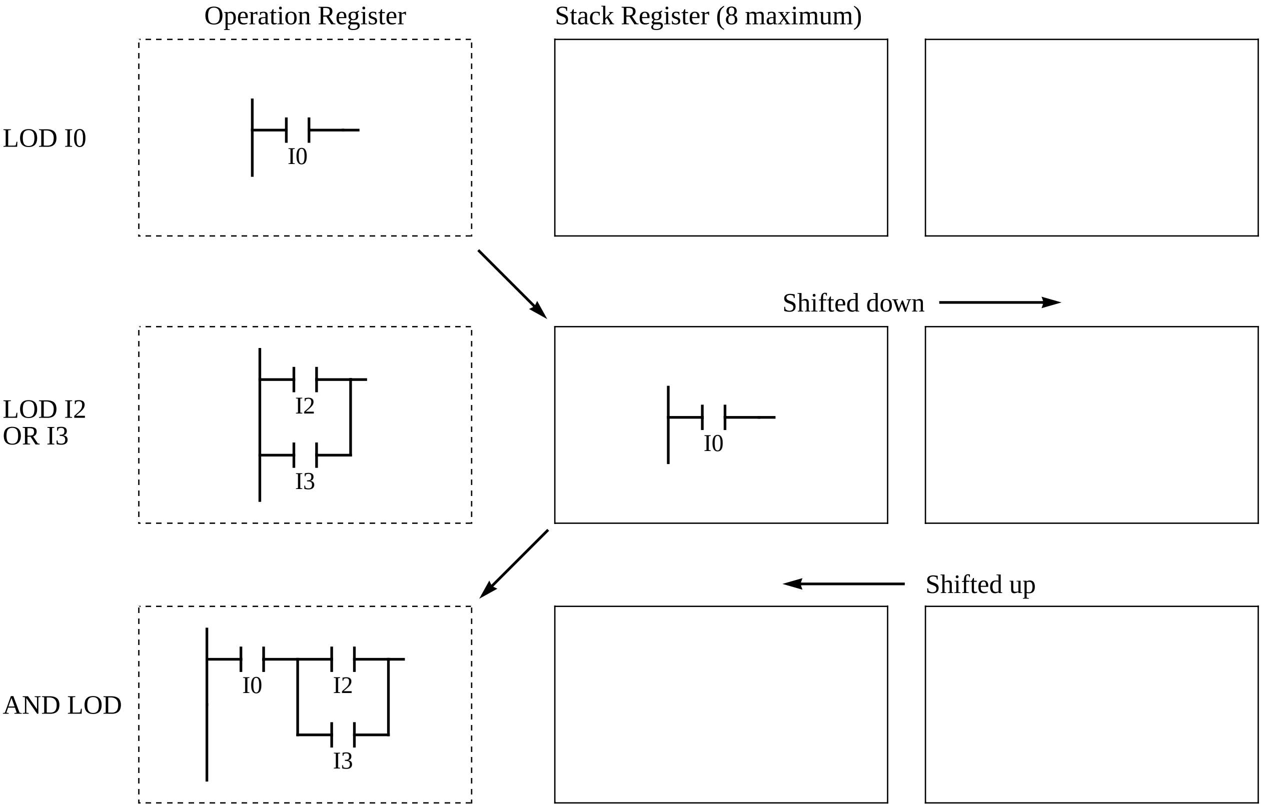


Timing Chart



When input I0 is on and either input I2 or I3 is on, output Q0 is on.
When input I0 is off or both inputs I2 and I3 are off, output Q0 is off.

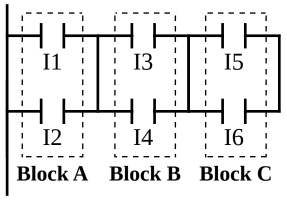
The AND LOD (load) instruction reads programs stored in the stack register by the LOD or LODN instruction and then AND’s them.



Example: AND LOD (Load)

For the following circuit, the AND LOD instruction can be used in two ways.

Ladder Diagram



First, the AND LOD instruction can be keyed following each of the sets of circuits that are to be connected in series.

Second, the AND LOD instruction can be entered twice at the end, after all the circuits to be connected in series have been keyed.

In either way, there is a relationship between the quantity of LOD instructions and the quantity of AND LOD instructions:

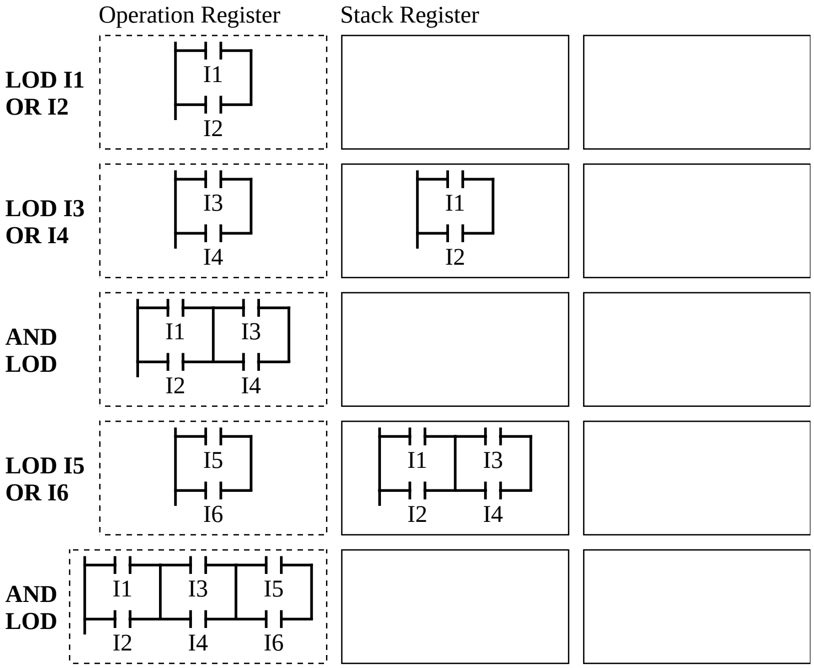
Quantity of AND LOD instructions = Quantity of LOD instructions – 1

Program List 1

Prgm Adrs	Instruction	Data
0	LOD	I1
1	OR	I2
2	LOD	I3
3	OR	I4
4	AND LOD	
5	LOD	I5
6	OR	I6
7	AND LOD	

The program represented by Block A is stored beginning with the LOD I1 instruction, and Block B is stored with LOD I3. Then, these two programs are read by the AND LOD instruction which is entered next, forming a circuit connected in series.

The program represented by Block C is stored beginning with the LOD I5 instruction. This is read by the final AND LOD instruction, so that it is connected in series with the two programs connected previously.



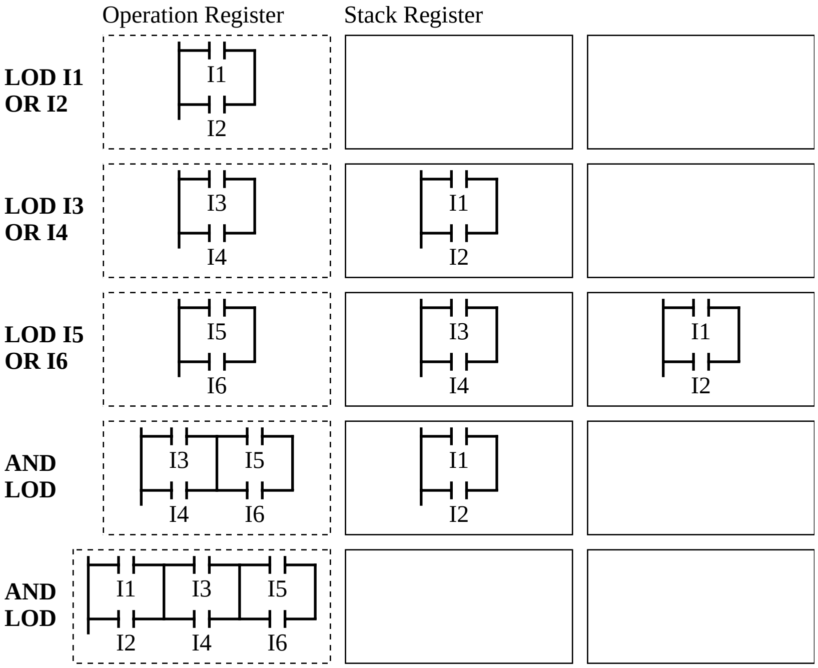
Program List 2

Prgm Adrs	Instruction	Data
0	LOD	I1
1	OR	I2
2	LOD	I3
3	OR	I4
4	LOD	I5
5	OR	I6
6	AND LOD	
7	AND LOD	

The program represented by Block A is stored beginning with the LOD I1 instruction, Block B is stored with LOD I3, and Block C is stored with LOD I5.

Then, the AND LOD instruction is entered twice consecutively, connecting the blocks in series, sequentially.

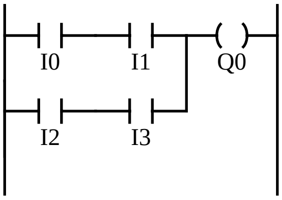
In this case, the number of stored circuits and read operations are increased.



OR LOD (Load)

The OR LOD instruction is used to connect, in parallel, two or more circuits starting with the LOD instruction. The OR LOD instruction is the equivalent of a “node” on a ladder diagram. The OR LOD instruction is keyed after entering those circuits to be connected.

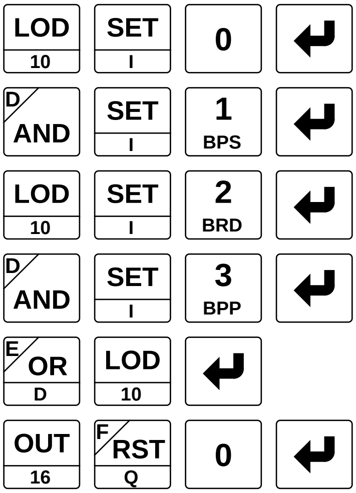
Ladder Diagram



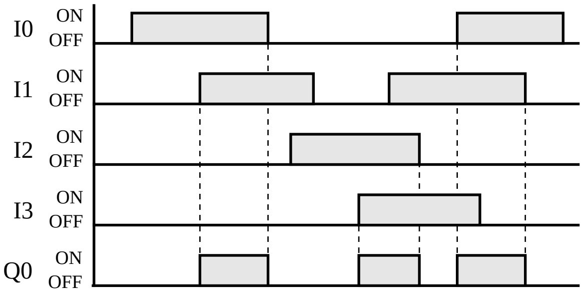
Program List

Prgm Adrs	Instruction	Data
0	LOD	I0
1	AND	I1
2	LOD	I2
3	AND	I3
4	OR LOD	
5	OUT	Q0

Key Operation

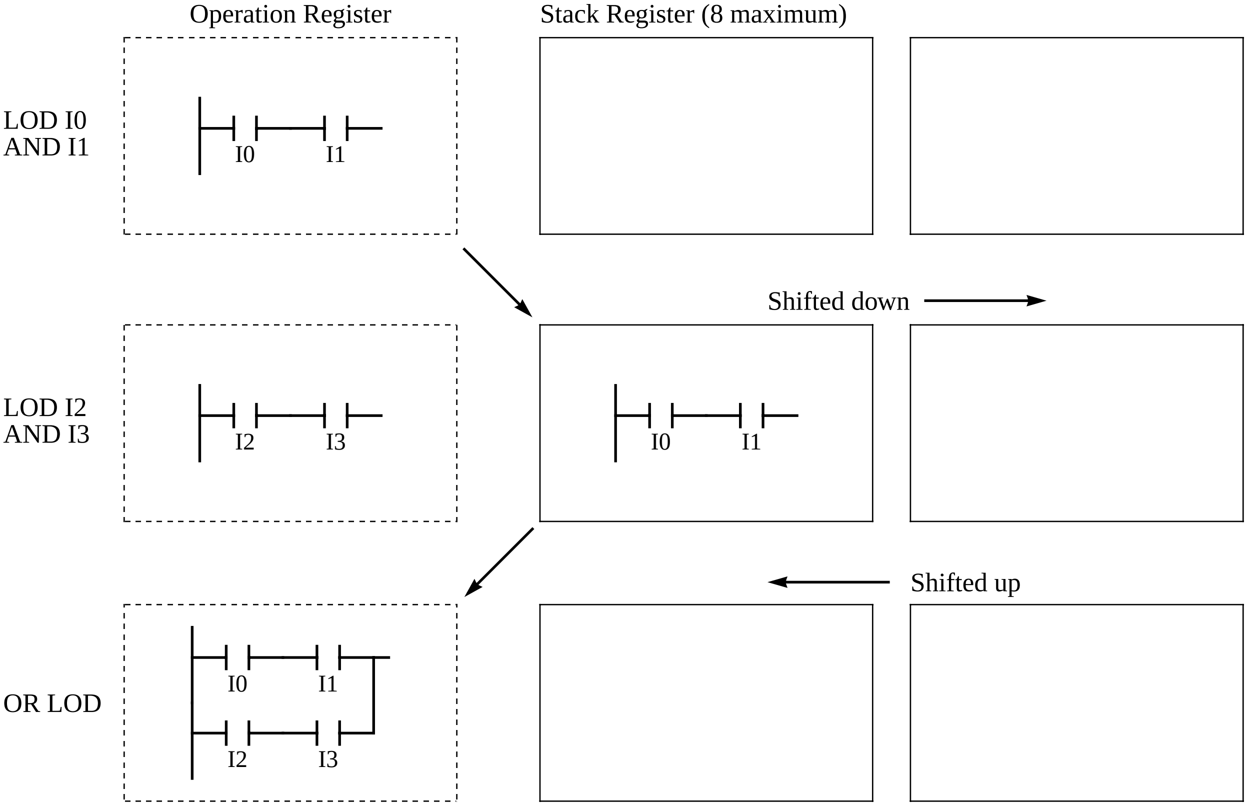


Timing Chart



When both inputs I0 and I1 are on or both inputs I2 and I3 are on, output Q0 is on.
When either input I0 or I1 is off and either input I2 or I3 is off, output Q0 is off.

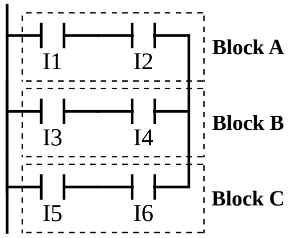
The OR LOD (load) instruction reads programs stored in the stack register by the LOD or LODN instruction and then OR’s them.



Example: OR LOD (Load)

For the following circuit, the OR LOD instruction can be used in two ways.

Ladder Diagram



First, the OR LOD instruction can be keyed following each of the sets of circuits that are to be connected in parallel.

Second, the OR LOD instruction can be entered twice at the end, after all the circuits to be connected in parallel have been keyed.

In either way, there is a relationship between the quantity of LOD instructions and the quantity of OR LOD instructions:

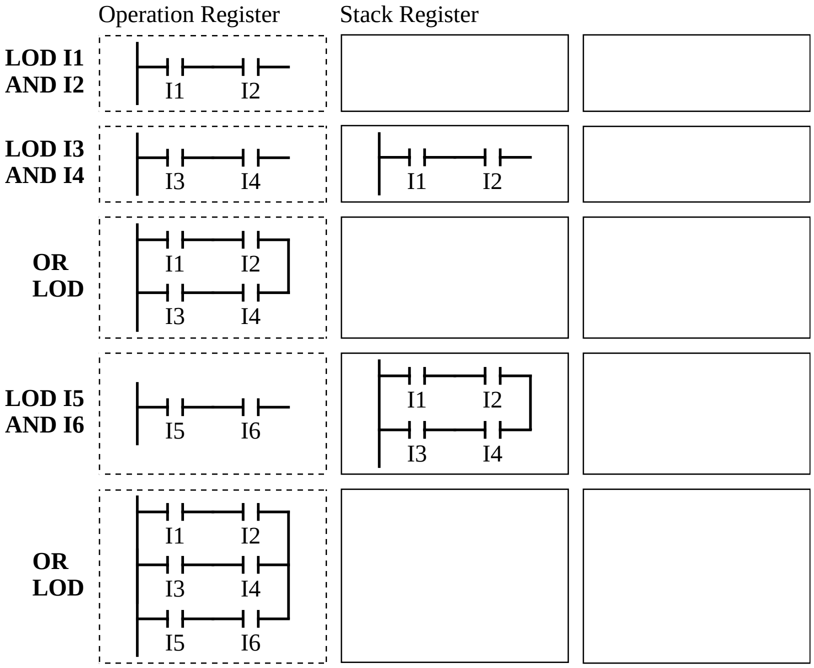
Quantity of OR LOD instructions = Quantity of LOD instructions – 1

Program List 1

Prgm Adrs	Instruction	Data
0	LOD	I1
1	AND	I2
2	LOD	I3
3	AND	I4
4	OR LOD	
5	LOD	I5
6	AND	I6
7	OR LOD	

The program represented by Block A is stored beginning with the LOD I1 instruction, and Block B is stored with LOD I3. Then, these two programs are read by the OR LOD instruction which is entered next, forming a circuit connected in parallel.

The program represented by Block C is stored beginning with the LOD I5 instruction. This is read by the final OR LOD instruction, so that it is connected in parallel with the two programs connected previously.



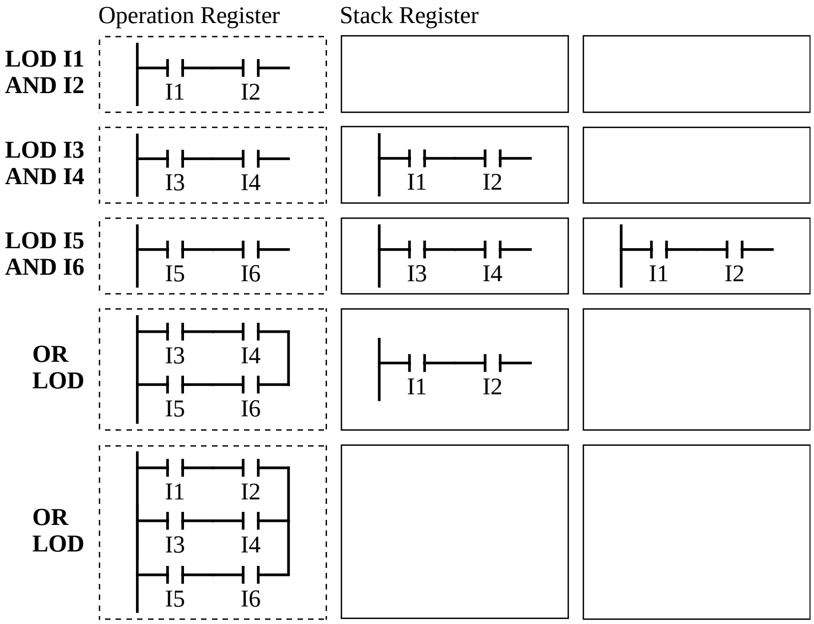
Program List 2

Prgm Adrs	Instruction	Data
0	LOD	I1
1	AND	I2
2	LOD	I3
3	AND	I4
4	LOD	I5
5	AND	I6
6	OR LOD	
7	OR LOD	

The program represented by Block A is stored beginning with the LOD I1 instruction, Block B is stored with LOD I3, and Block C is stored with LOD I5.

Then, the OR LOD instruction is entered twice consecutively, connecting the blocks in parallel, sequentially.

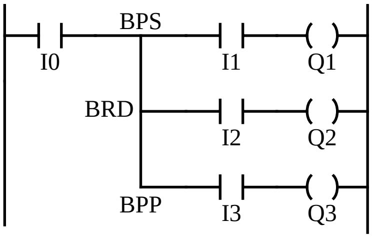
In this case, the number of stored circuits and read operations are increased.



BPS (Bit Push), BRD (Bit Read), and BPP (Bit Pop)

The BPS (bit push) instruction is used to save the result of bit logical operation temporarily.
The BRD (bit read) instruction is used to read the result of bit logical operation which was saved temporarily.
The BPP (bit pop) instruction is used to restore the result of bit logical operation which was saved temporarily.

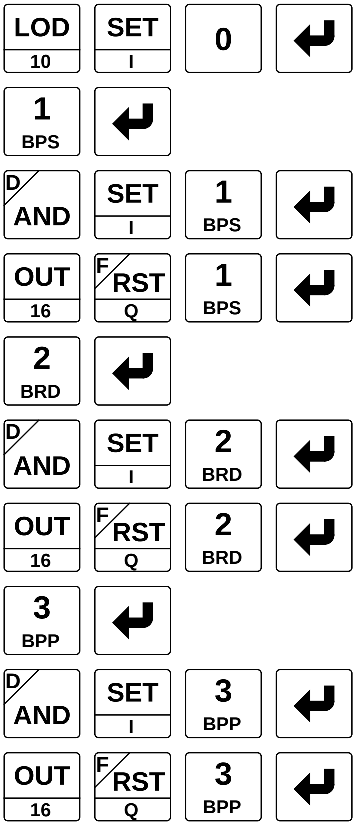
Ladder Diagram



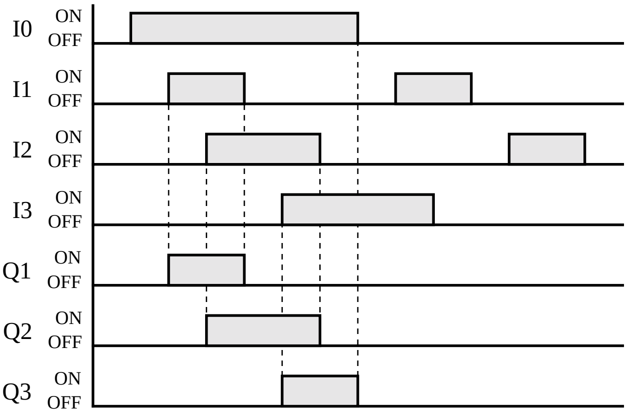
Program List

Prgm Adrs	Instruction	Data
0	LOD	I0
1	BPS	
2	AND	I1
3	OUT	Q1
4	BRD	
5	AND	I2
6	OUT	Q2
7	BPP	
8	AND	I3
9	OUT	Q3

Key Operation



Timing Chart



When both inputs I0 and I1 are on, output Q1 is turned on.
When both inputs I0 and I2 are on, output Q2 is turned on.
When both inputs I0 and I3 are on, output Q3 is turned on.

BPS (Bit Push), BRD (Bit Read), and BPP (Bit Pop), continued

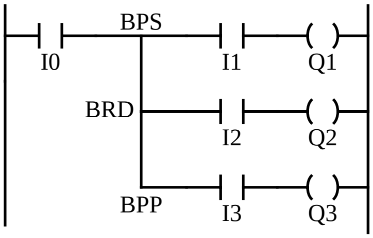
Data Movement in Operation Register and Bit Stack Register

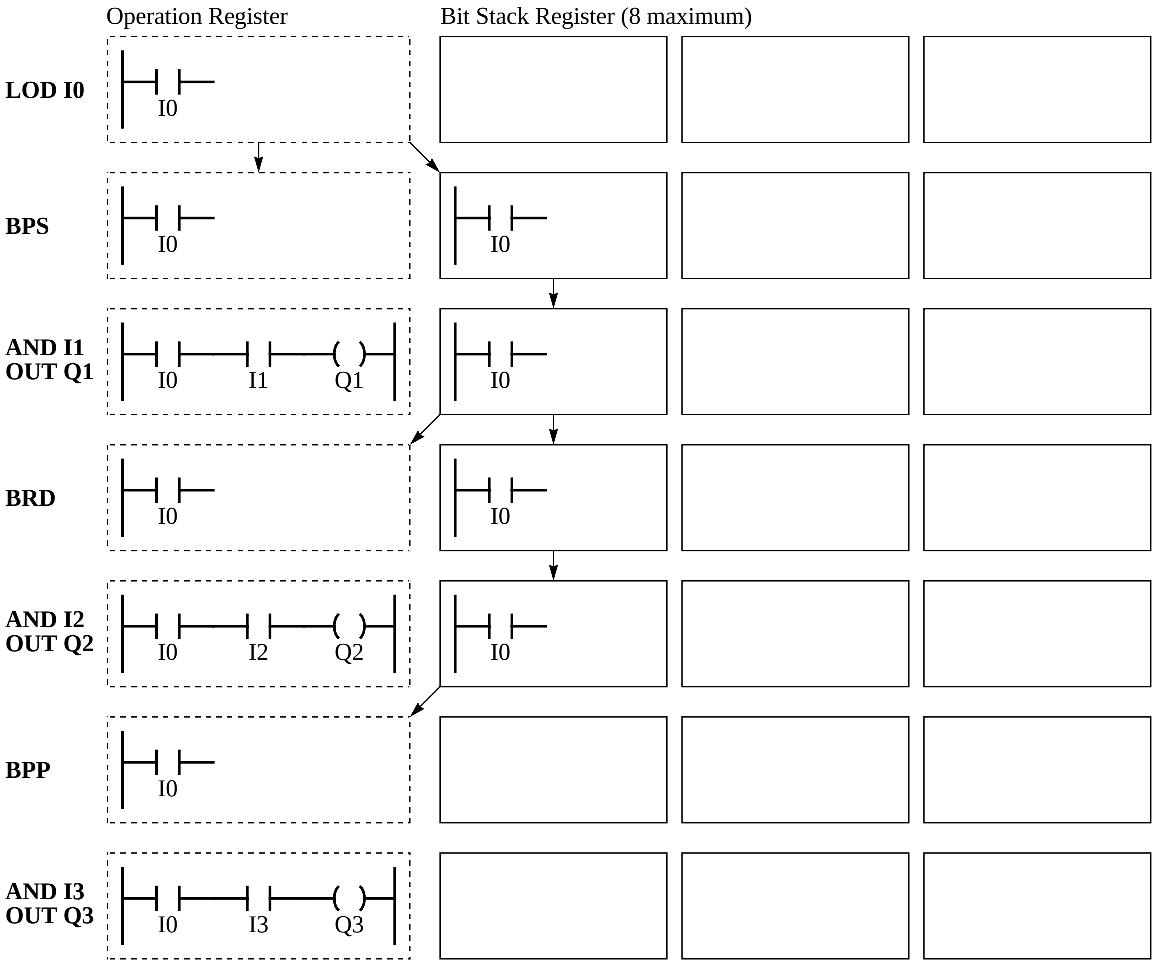
When the BPS (bit push) instruction is used, the program in the operation register is stored in the first bit stack register. When the BPS instruction is used again, the program in the first stack register is stored in the second bit stack register and the program in the operation register is stored in the first stack register. Each time the BPS instruction is used, the program is moved to the next bit stack register. Program blocks can be stored in a maximum of eight bit stack registers.

When the BRD (bit read) instruction is used, the program in the first bit stack register is read to the operation register. All program blocks stored in bit stack registers are not moved.

When the BPP (bit push) instruction is used, all program blocks in bit stack registers are shifted back by one place. The program in the first bit stack register is moved to the operation register.

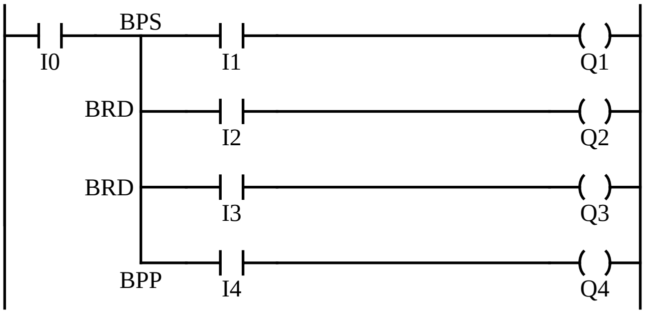
Ladder Diagram





Example: Using One-bit Stack Register

Ladder Diagram

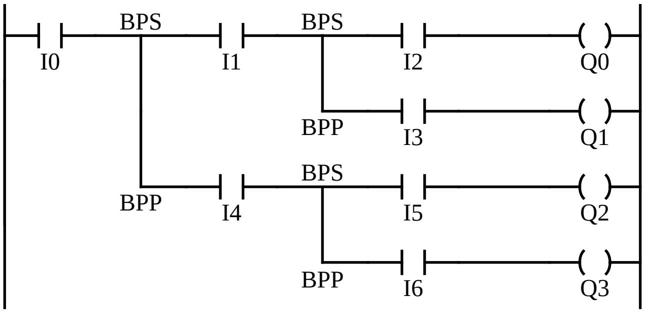


Program List

Prgm Adrs	Instruction	Data
0	LOD	I0
1	BPS	
2	AND	I1
3	OUT	Q1
4	BRD	
5	AND	I2
6	OUT	Q2
7	BRD	
8	AND	I3
9	OUT	Q3
10	BPP	
11	AND	I4
12	OUT	Q4

Example: Using Two-bit Stack Registers

Ladder Diagram

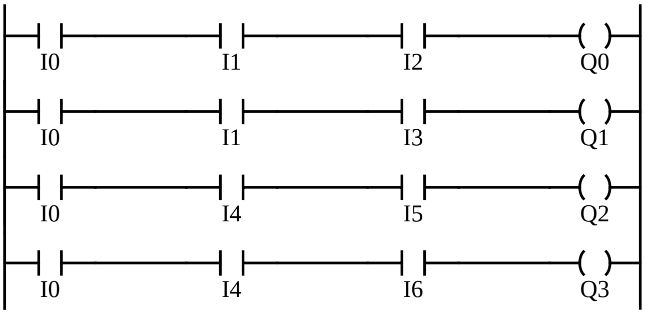


Program List

Prgm Adrs	Instruction	Data
0	LOD	I0
1	BPS	
2	AND	I1
3	BPS	
4	AND	I2
5	OUT	Q0
6	BPP	
7	AND	I3
8	OUT	Q1
9	BPP	
10	AND	I4
11	BPS	
12	AND	I5
13	OUT	Q2
14	BPP	
15	AND	I6
16	OUT	Q3

Equivalent to above – not using BPS and BPP

Ladder Diagram

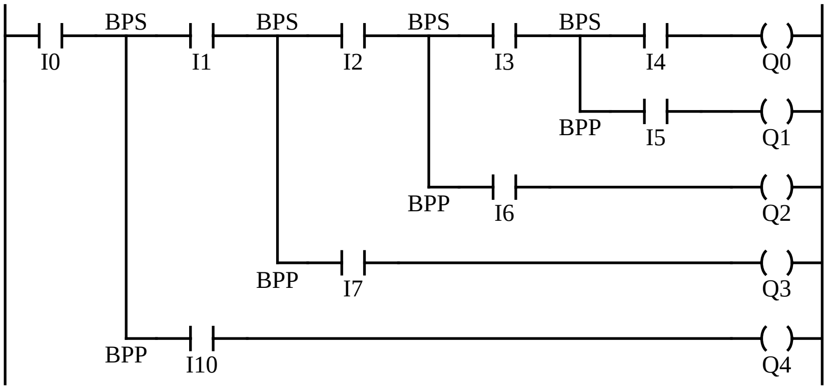


Program List

Prgm Adrs	Instruction	Data
0	LOD	I0
1	AND	I1
2	AND	I2
3	OUT	Q0
4	LOD	I0
5	AND	I1
6	AND	I3
7	OUT	Q1
8	LOD	I0
9	AND	I4
10	AND	I5
11	OUT	Q2
12	LOD	I0
13	AND	I4
14	AND	I6
15	OUT	Q3

Example: Using Four-bit Stack Registers

Ladder Diagram

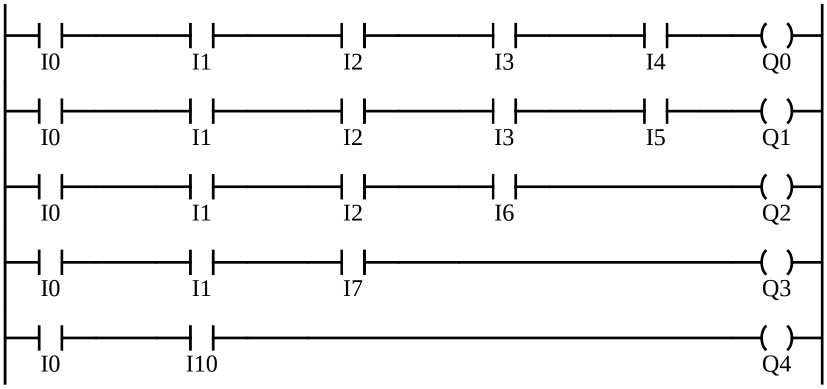


Program List

Prgm Adrs	Instruction	Data
0	LOD	I0
1	BPS	
2	AND	I1
3	BPS	
4	AND	I2
5	BPS	
6	AND	I3
7	BPS	
8	AND	I4
9	OUT	Q0
10	BPP	
11	AND	I5
12	OUT	Q1
13	BPP	
14	AND	I6
15	OUT	Q2
16	BPP	
17	AND	I7
18	OUT	Q3
19	BPP	
20	AND	I10
21	OUT	Q4

Equivalent to above – not using BPS and BPP

Ladder Diagram

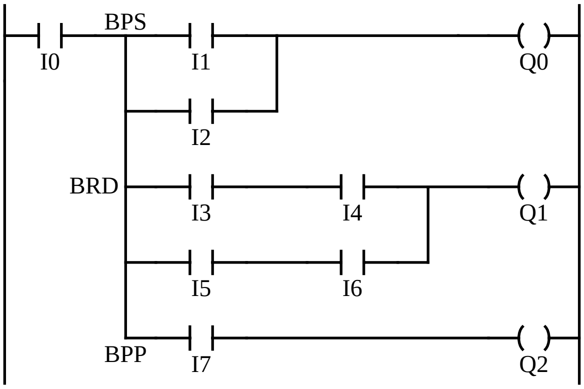


Program List

Prgm Adrs	Instruction	Data
0	LOD	I0
1	AND	I1
2	AND	I2
3	AND	I3
4	AND	I4
5	OUT	Q0
6	LOD	I0
7	AND	I1
8	AND	I2
9	AND	I3
10	AND	I5
11	OUT	Q1
12	LOD	I0
13	AND	I1
14	AND	I2
15	AND	I6
16	OUT	Q2
17	LOD	I0
18	AND	I1
19	AND	I7
20	OUT	Q3
21	LOD	I0
22	AND	I10
23	OUT	Q4

Example: BPS, BRD, and BPP with AND LOD and OR LOD

Ladder Diagram

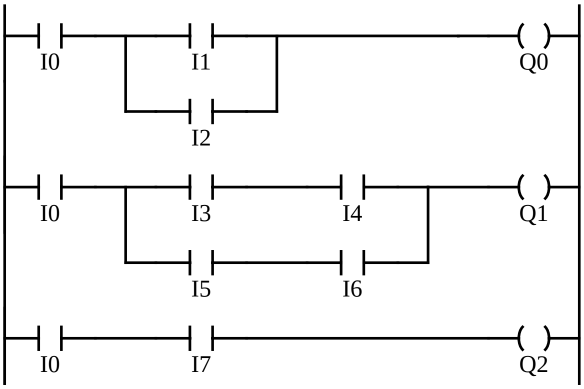


Program List

Prgm Adrs	Instruction	Data
0	LOD	I0
1	BPS	
2	LOD	I1
3	OR	I2
4	AND LOD	
5	OUT	Q0
6	BRD	
7	LOD	I3
8	AND	I4
9	LOD	I5
10	AND	I6
11	OR LOD	
12	AND LOD	
13	OUT	Q1
14	BPP	
15	AND	I7
16	OUT	Q2

Equivalent to above – not using BPS, BRD, and BPP

Ladder Diagram



Program List

Prgm Adrs	Instruction	Data
0	LOD	I0
1	LOD	I1
2	OR	I2
3	AND LOD	
4	OUT	Q0
5	LOD	I0
6	LOD	I3
7	AND	I4
8	LOD	I5
9	AND	I6
10	OR LOD	
11	AND LOD	
12	OUT	Q1
13	LOD	I0
14	AND	I7
15	OUT	Q2

TIM, TMH, and TMS (Timer)

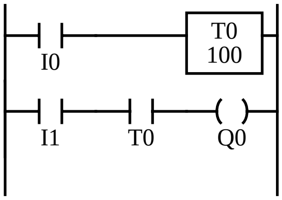
Three types of timers are available; 100-msec timedown timer TIM, 10-msec timedown timer TMH, and 1-msec timedown timer TMS. A total of 32 timers and counters can be programmed in the standard processing mode. Each timer must be allocated to a unique number 0 through 31, and the same number cannot be used for counters. In the high-speed processing mode, timer numbers 0 through 15 are available.

Timer	Allocation Number	Range	Increments	Preset Value
TIM (100-msec timer)	TIM0 to TIM31	0 to 999.9 sec	100 msec	Constant: 0 to 9999 Data registers: D0 to D99 (standard mode) D0 to D31 (high-speed mode)
TMH (10-msec timer)	TMH0 to TMH31	0 to 99.99 sec	10 msec	
TMS (1-msec timer)	TMS0 to TMS31	0 to 9.999 sec	1 msec	

The preset value can be 0 through 9999 and designated using a decimal constant or data register. Although a data register can hold a value up to 65535, a timer preset value can be 0 through 9999. If the data register designated as a timer preset value holds a value over 9999, a user program execution error will result, then error indicator ERR1 is lit and special internal relay M304 turns on. Data registers D0 through D99 are available in the standard processing mode and D0 through D31 in the high-speed processing mode.

TIM (100-msec Timer)

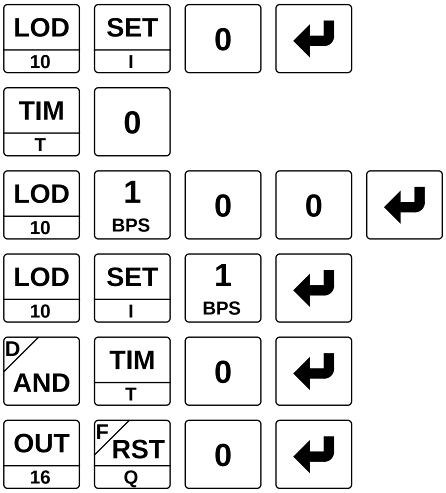
Ladder Diagram (TIM)



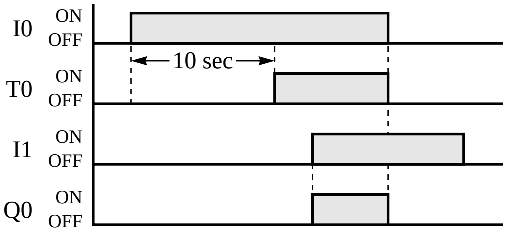
Program List

Prgm Adrs	Instruction	Data
0	LOD	I0
1	TIM	0
		100
3	LOD	I1
4	AND	T0
5	OUT	Q0

Key Operation



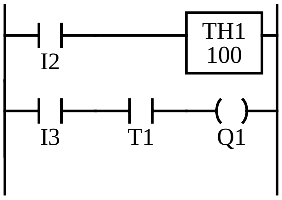
Timing Chart



Note: To enter a decimal constant as a preset value, press the LOD/10 key, followed by the preset value.

TMH (10-msec Timer)

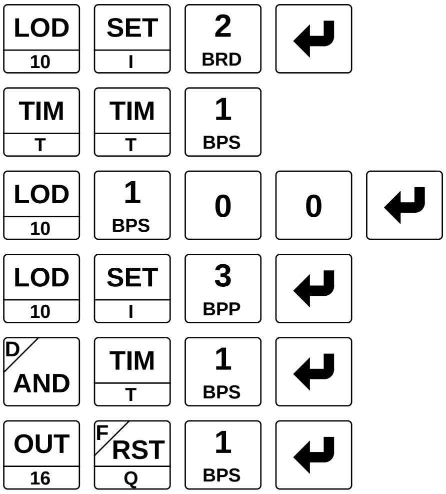
Ladder Diagram (TMH)



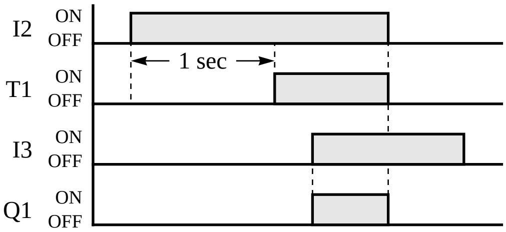
Program List

Prgm Adrs	Instruction	Data
0	LOD	I2
1	TMH	1
		100
3	LOD	I3
4	AND	T1
5	OUT	Q1

Key Operation


LOD 10 | SET I | 2 BRD | ↵
TIM T | TIM T | 1 BPS
LOD 10 | 1 BPS | 0 | 0 | ↵
LOD 10 | SET I | 3 BPP | ↵
D AND | TIM T | 1 BPS | ↵
OUT 16 | F RST Q | 1 BPS | ↵

Timing Chart



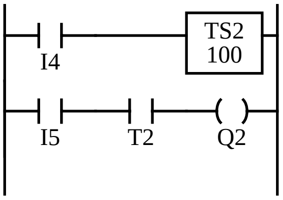
Note: To enter a decimal constant as a preset value, press the LOD/10 key, followed by the preset value.

Pressing the TIM key on the program loader programs the TIM, TMH, or TMS instruction alternately.

TIM, TMH, and TMS (Timer), continued

TMS (1-msec Timer)

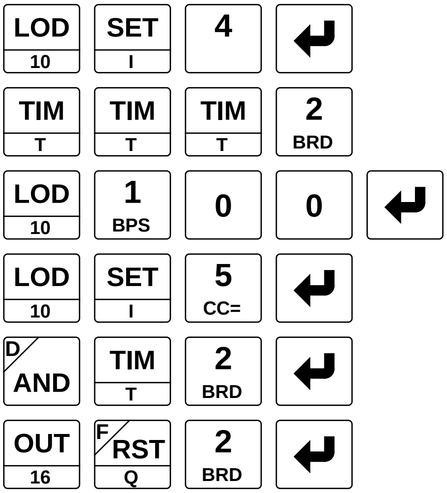
Ladder Diagram (TMS)



Program List

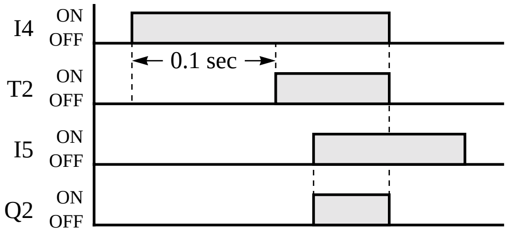
Prgm Adrs	Instruction	Data
0	LOD	I4
1	TMS	2 100
3	LOD	I5
4	AND	T2
5	OUT	Q2

Key Operation


LOD 10 | SET I | 4 | ↵
TIM T | TIM T | TIM T | 2 BRD
LOD 10 | 1 BPS | 0 | 0 | ↵
LOD 10 | SET I | 5 CC= | ↵
D AND | TIM T | 2 BRD | ↵
OUT 16 | F RST Q | 2 BRD | ↵

Note: Pressing the TIM key on the program loader programs the TIM, TMH, or TMS instruction alternately.

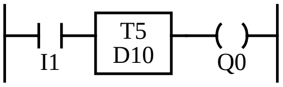
Timing Chart



Timer Circuit

The preset value 0 through 9999 can be designated using a data register D0 through D99 in the standard processing mode or D0 through D31 in the high-speed processing mode; then the data of the data register becomes the preset value. Directly after the TIM, TMH, or TMS instruction with two required addresses, the OUT (output) instruction can be keyed.

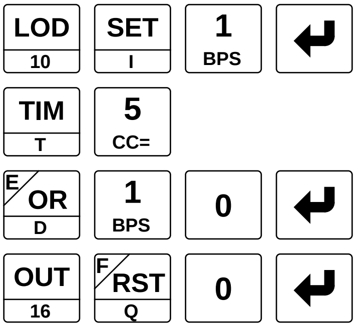
Ladder Diagram



Program List

Prgm Adrs	Instruction	Data
0	LOD	I1
1	TIM	5 D10
3	OUT	Q0

Key Operation



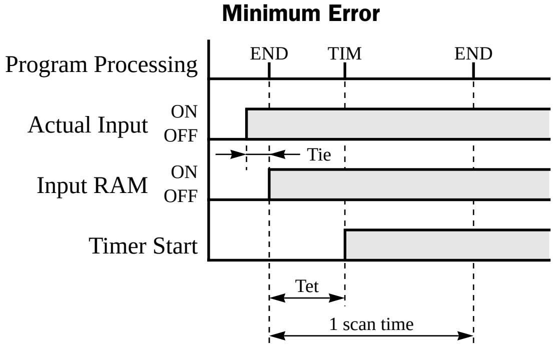
- Timedown from the preset value is initiated when the operation result directly before the timer input is on.
- The timer output turns on when the timed value reaches zero.
- The timed value returns to the preset value when the timer input is off.
- The same timer or counter number cannot be programmed more than once. If an attempt is made to do so, then an error message will result.
- Timer preset values can be changed without transferring the entire program to the MICRO^3 base unit again. See page 3-14. If the timer preset value is changed during timedown, then the timer remains unchanged for that cycle. The change will be reflected in the next time cycle.
- If the timer preset value is changed to zero, then the timer stops operation, and the timer output is turned on immediately.

Timer Accuracy

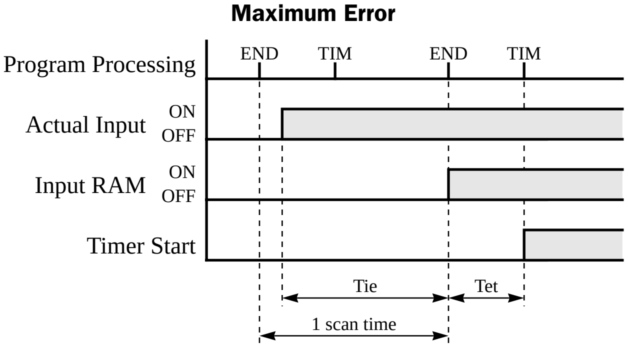
Timer accuracy due to software configuration depends on three factors: timer input error, timer counting error, and timeout output error. These errors are not constant but vary with the user program and other causes.

Timer Input Error

The input status is read at the END processing and stored to the input RAM. So, an error occurs depending on the timing when the timer input turns on in a scan cycle. The same error occurs on the normal input and the catch input. The timer input error shown below does not include input delay caused by the hardware.



When the input turns on immediately before the END processing, Tie is almost zero. Then the timer input error is only Tet (behind error) and is at its minimum.



When the input turns on immediately after the END processing, Tie is almost equal to one scan time. Then the timer input error is Tie + Tet = one scan time + Tet (behind error) and is at its maximum.

Tie: Time from input turning on to the END processing
Tet: Time from the END processing to the timer instruction execution

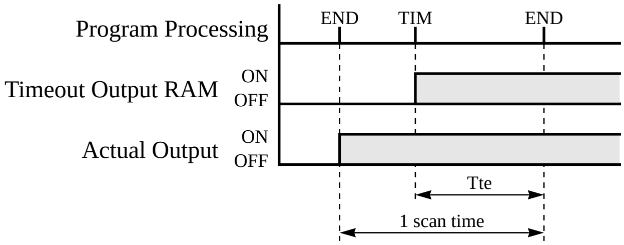
Timer Counting Error

Every timer instruction operation is individually based on asynchronous 16-bit reference timers. Therefore, an error occurs depending on the status of the asynchronous 16-bit timer when the timer instruction is executed.

Error		TIM (100-msec timer)	TMH (10-msec timer)	TMS (1-msec timer)
Minimum	Advance error	0 msec	0 msec	0 msec
	Behind error	0 msec	0 msec	0 msec
Maximum	Advance error	100 msec	10 msec	1 msec
	Behind error	1 scan time	1 scan time	1 scan time

Timeout Output Error

The output RAM status is set to the actual output when the END instruction is processed. So, an error occurs depending on the timing when the timeout output turns on in a scan cycle. The timeout output error shown on the right does not include output delay caused by the hardware.



Timeout output error is equal to Tte (behind error) and can be between zero and one scan time.

$0 < \text{Tte} < 1$ scan time

Tte: Time from the timer instruction execution to the END processing

Timer Accuracy, continued

Maximum and Minimum of Individual Errors

Error		Timer Input Error	Timer Counting Error	Timeout Output Error	Remarks
Minimum	Advance error	0 (Note)	0	0 (Note)	Note: Advance error does not occur at the timer input and timeout output. Increment is 100 msec (TIM), 10 msec (TMH), or 1 msec (TMS).
	Behind error	Tet	0	Tte	
Maximum	Advance error	0 (Note)	Increment	0 (Note)	
	Behind error	1 scan time + Tet	1 scan time	Tte	

Maximum and Minimum of Total Error

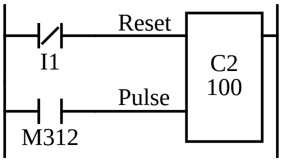
Error		Total Error	Remarks
Minimum	Advance error	0	
	Behind error	0	
Maximum	Advance error	Increment – (Tet + Tte)	The maximum advance error is: Increment – 1 scan time
	Behind error	2 scan times + (Tet + Tte)	The maximum behind error is: 3 scan times

Tet + Tte = 1 scan time

Power Failure Memory Protection

Timers TIM, TMH, and TMS do not have power failure protection. A timer with this protection can be devised using a counter instruction and special internal relay M311 (1-sec clock), M312 (100-msec clock), or M313 (10-msec clock).

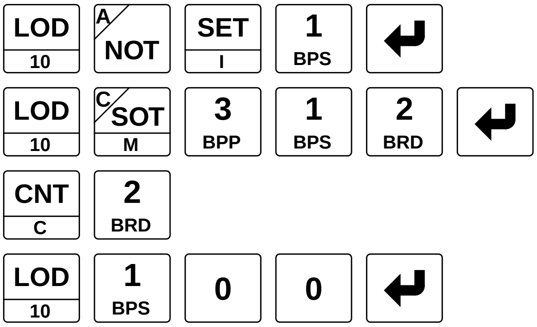
Ladder Diagram
(100-msec Timer)



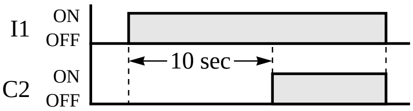
Program List

Prgm Adrs	Instruction	Data
0	LOD NOT	I1
1	LOD	M312
2	CNT	2 100

Key Operation



Timing Chart



Note: Counters CNT2 through CNT31 are adding counters, and all counter current values are maintained during a power failure.

CNT (Counter)

Three types of counters are available; dual-pulse reversible counter CNT0, up/down selection reversible counter CNT1, and adding (up) counters CNT2 through CNT31. A total of 32 timers and counters can be programmed in the standard processing mode. Each counter must be allocated to a unique number 0 through 31, and the same number cannot be used for timers. In the high-speed processing mode, counter numbers 0 through 15 are available. For the high-speed processing mode, see page 4-1.

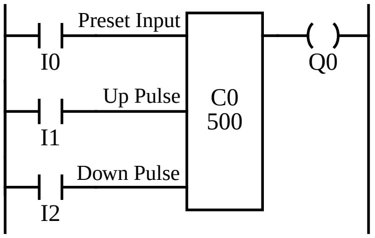
Counter	Allocation Number	Preset Value
Dual-pulse reversible counter	CNT0	Constant: 0 to 9999 Data registers: D0 to D99 (standard mode) D0 to D31 (high-speed mode)
Up/down selection reversible counter	CNT1	
Adding (up) counter	CNT2 to CNT31 (standard mode) CNT2 to CNT15 (high-speed mode)	

Dual-Pulse Reversible Counter CNT0

The dual-pulse reversible counter CNT0 has up and down pulse inputs, so that three inputs are required. The circuit for a dual-pulse reversible counter must be programmed in the following order: preset input, up-pulse input, down-pulse input, and the CNT0 instruction, followed by the counter preset value from 0 to 9999.

The preset value can be designated using a decimal constant or a data register. When a data register is used, the data of the data register becomes the preset value. If the data register designated as a counter preset value holds a value over 9999, a user program execution error will result, then the error indicator, ERR1, is lit, and the special internal relay, M304, turns on. Data registers D0 through D99 are available in the standard processing mode and D0 through D31 in the high-speed processing mode.

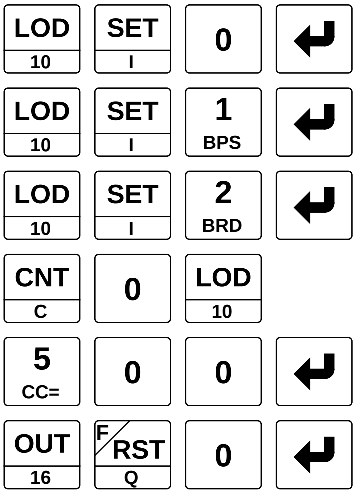
Ladder Diagram (CNT0)



Program List

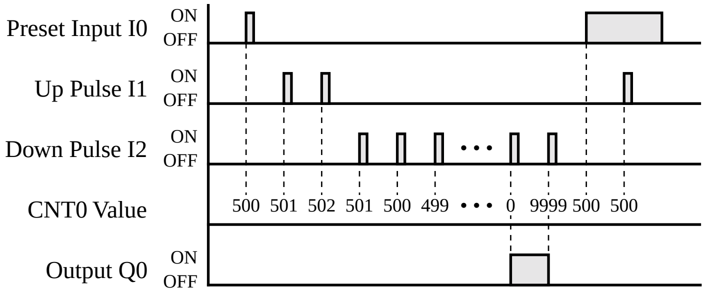
Prgm Adrs	Instruction	Data
0	LOD	I0
1	LOD	I1
2	LOD	I2
3	CNT	0 500
5	OUT	Q0

Key Operation



Note: To enter a decimal constant as a preset value, press the LOD/10 key followed by the preset value.

Timing Chart



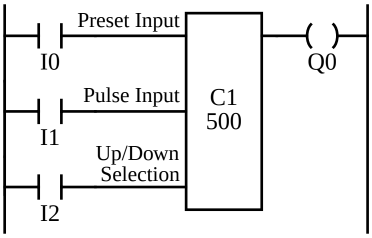
- The same counter or timer number cannot be programmed more than once.
- The preset input must be turned on initially so that the counted value returns to the preset value.
- The preset input must be turned off before counting may begin.
- When the up pulse and down pulse are on simultaneously, no pulse is counted.
- The counter output is on only when the counted value is zero.
- After the counted value reaches zero (counting down), it changes to 9999 on the next count down.
- After the counted value reaches 9999 (counting up), it changes to zero on the next count up.
- When power is off, the counter's counted value is held.
- Counter preset values can be changed without transferring the entire program to the MICRO^3 base unit (see page 3-14).
- When the preset value is changed during counter operation, the change becomes effective immediately.

Up/Down Selection Reversible Counter CNT1

The up/down selection reversible counter CNT1 has selection input to switch the up/down gate, so that three inputs are required. The circuit for an up/down selection reversible counter must be programmed in the following order: preset input, pulse input, up/down selection input, and the CNT1 instruction, followed by the counter preset value from 0 to 9999.

The preset value can be designated using a decimal constant or a data register. When a data register is used, the data of the data register becomes the preset value. If the data register designated as a counter preset value holds a value over 9999, a program execution error will result, then the error indicator, ERR1, is lit, and the special internal relay, M304, turns on. Data registers D0 through D99 are available in the standard processing mode and D0 through D31 in the high-speed processing mode.

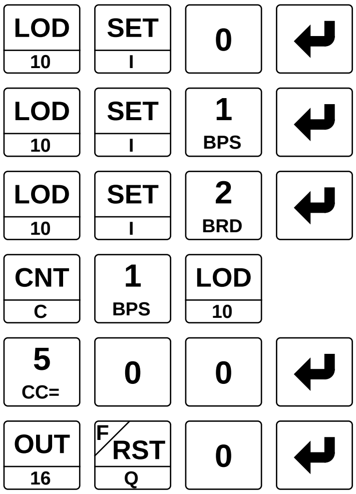
Ladder Diagram (CNT1)



Program List

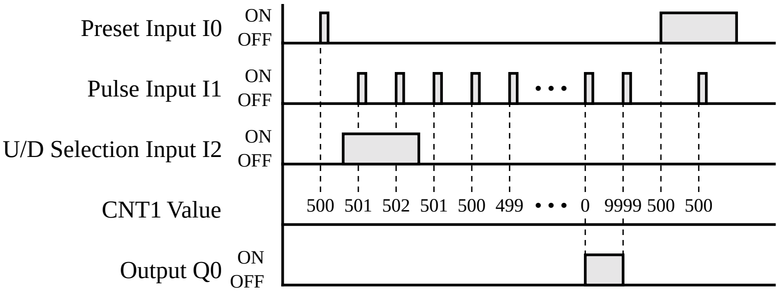
Prgm Adrs	Instruction	Data
0	LOD	I0
1	LOD	I1
2	LOD	I2
3	CNT	1 500
5	OUT	Q0

Key Operation



Note: To enter a decimal constant as a preset value, press the LOD/10 key followed by the preset value.

Timing Chart



- The same counter or timer number cannot be programmed more than once.
- The preset input must be turned on initially so that the counted value returns to the preset value.
- The preset input must be turned off before counting may begin.
- The up mode is selected when the up/down selection input is on.
- The down mode is selected when the up/down selection input is off.
- The counter output is on only when the counted value is zero.
- After the counted value reaches zero (counting down), it changes to 9999 on the next count down.
- After the counted value reaches 9999 (counting up), it changes to zero on the next count up.
- When power is off, the counter's counted value is held.
- Counter preset values can be changed without transferring the entire program to the MICRO^3 base unit (see page 3-14).
- When the preset value is changed during counter operation, the change becomes effective immediately.

Adding (Up) Counters CNT2 through CNT31

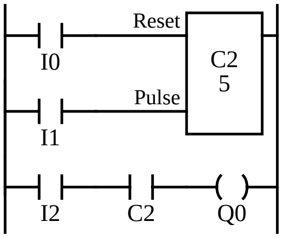
Standard counter circuits, using the CNT instruction, feature an adding (UP) counter. There are 30 adding counters CNT2 through CNT31 in the standard processing mode or 14 adding counters CNT2 through CNT15 in the high-speed processing mode.

When counter instructions are programmed, two addresses are required. The circuit for an adding (UP) counter must be programmed in the following order: reset input, pulse input, the CNT instruction, a counter number 2 through 31, followed by the counter preset value from 0 to 9999.

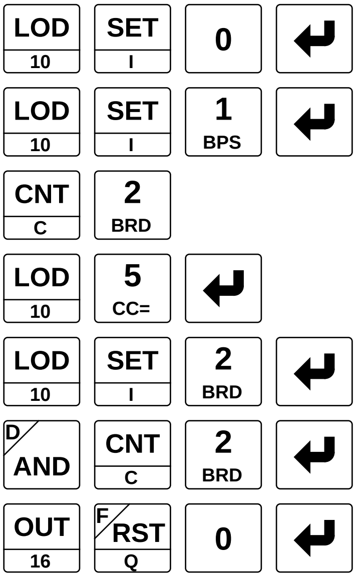
The preset value can be designated using a decimal constant or a data register. When a data register is used, the data of the data register becomes the preset value. If the data register designated as a counter preset value holds a value over 9999, a program execution error will result, then error indicator ERR1 is lit, and special internal relay M304 turns on. Data registers D0 through D99 are available in the standard processing mode and D0 through D31 in the high-speed processing mode.

Adding (Up) Counters CNT2 through CNT31, continued

Ladder Diagram (CNT2 to 31)



Key Operation

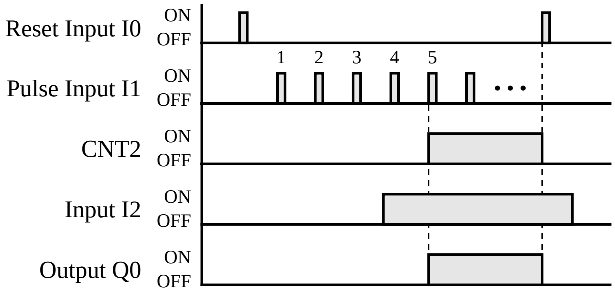

LOD 10 | SET I | 0 | ↵
LOD 10 | SET I | 1 BPS | ↵
CNT C | 2 BRD
LOD 10 | 5 CC= | ↵
LOD 10 | SET I | 2 BRD | ↵
D AND | CNT C | 2 BRD | ↵
OUT 16 | F RST Q | 0 | ↵

Program List

Prgm Adrs	Instruction	Data
0	LOD	I0
1	LOD	I1
2	CNT	2
		5
4	LOD	I2
5	AND	C2
6	OUT	Q0

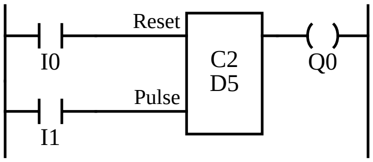
Note: To enter a decimal constant as a preset value, press the LOD/10 key followed by the preset value.

Timing Chart

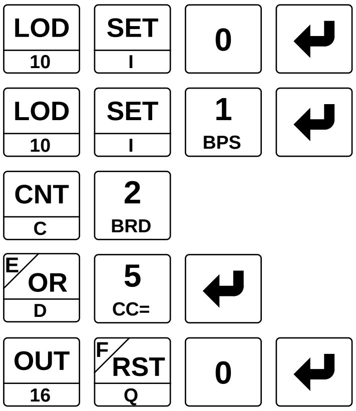


The preset value 0 through 9999 can be designated using a data register D0 through D99 in the standard processing mode or D0 through D31 in the high-speed processing mode, then the data of the data register becomes the preset value. Directly after the CNT instruction with two required addresses, the OUT (output) instruction can be keyed.

Ladder Diagram



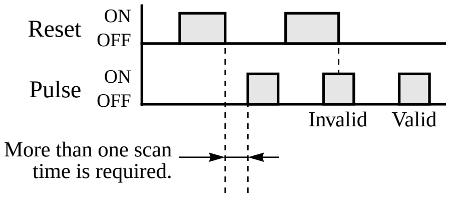
Key Operation



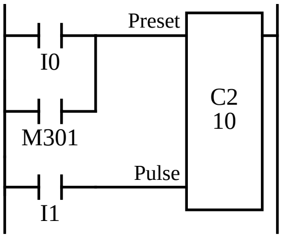
Program List

Prgm Adrs	Instruction	Data
0	LOD	I0
1	LOD	I1
2	CNT	2
		D5
4	OUT	Q0

- The same counter or timer number cannot be programmed more than once.
- While the reset input is off, the counter counts the leading edges of pulse inputs and compares them with the preset value.
- When the counted value reaches the preset value, the counter turns output on. The output stays on until the reset input is turned on.
- When the reset input changes from off to on, the counted value is reset.
- When the reset input is on, all pulse inputs are ignored.
- The reset input must be turned off before counting may begin.
- When power is off, the counter's counted value is held.
- Counter preset values can be changed without transferring the entire program to the MICRO^3 base unit (see page 3-14).
- When the preset value is changed during counter operation, the change becomes effective immediately.
- The reset input has priority over the pulse input. One scan after the reset input has changed from on to off, the counter starts counting the pulse inputs as they change from off to on.



- When MICRO^3 is turned off, counter current values are maintained. When resetting the counter current values is required at start up, include initialize pulse special internal relay M301 in an OR circuit with the reset input.



CC= and CC≥ (Counter Comparison)

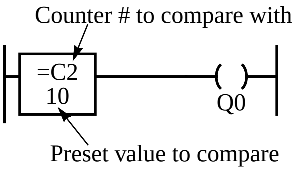
The CC= instruction is an equivalent comparison instruction for counted values. This instruction will constantly compare counted values to the value that has been programmed in. When the counter value equals the given value, the desired output will be initiated.

The CC≥ instruction is an equal to or greater than comparison instruction for counted values. This instruction will constantly compare counted values to the value that has been programmed in. When the counter value is equal to or greater than the given value, the desired output will be initiated.

When a counter comparison instruction is programmed, two addresses are required. The circuit for a counter comparison instruction must be programmed in the following order: the CC= or CC≥ instruction, a counter number 1 through 31, followed by the preset value to compare from 0 to 9999.

The preset value can be designated using a decimal constant or a data register. When a data register is used, the data of the data register becomes the preset value. If the data register designated as a counter comparison preset value holds a value over 9999, a program execution error will result, then the error indicator, ERR1, is lit, and the special internal relay, M304, turns on. Data registers D0 through D99 are available in the standard processing mode and D0 through D31 in the high-speed processing mode.

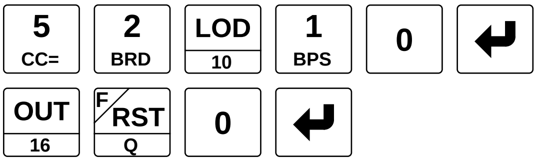
Ladder Diagram (CC=)



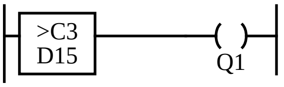
Program List

Prgm Adrs	Instruction	Data
0	CC=	2 10
2	OUT	Q0

Key Operation



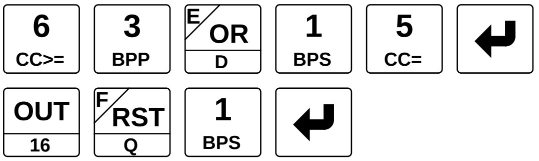
Ladder Diagram (CC≥)



Program List

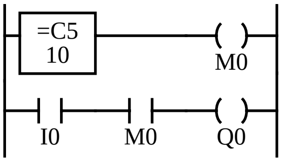
Prgm Adrs	Instruction	Data
0	CC≥	3 D15
2	OUT	Q1

Key Operation



- The CC= and CC≥ instructions can be used repeatedly for different preset values.
- The comparison instructions only compare the counted value. The status of the counter does not affect this function.
- The comparison instructions also serve as an implicit LOD instruction.
- The comparison instructions can be used with internal relays, which are AND'ed or OR'ed at a separate program address.
- Like the LOD instruction, the comparison instructions can be followed by the AND and OR instructions.
- Another way to accomplish the above is to use comparison instructions which are then followed by the AND LOD or OR LOD instructions.

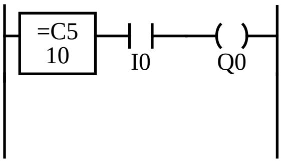
Ladder Diagram



Program List

Prgm Adrs	Instruction	Data
0	CC=	5 10
2	OUT	M0
3	LOD	I0
4	AND	M0
5	OUT	Q0

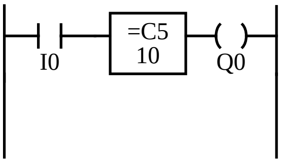
Ladder Diagram



Program List

Prgm Adrs	Instruction	Data
0	CC=	5 10
2	AND	I0
3	OUT	Q0

Ladder Diagram

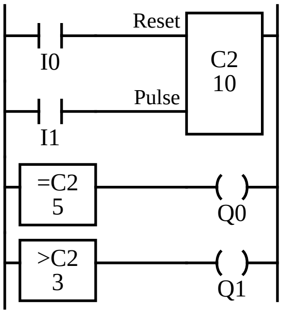


Program List

Prgm Adrs	Instruction	Data
0	LOD	I0
1	CC=	5 10
3	AND LOD	
4	OUT	Q0

Examples: CC= and CC≥ (Counter Comparison)

Ladder Diagram 1



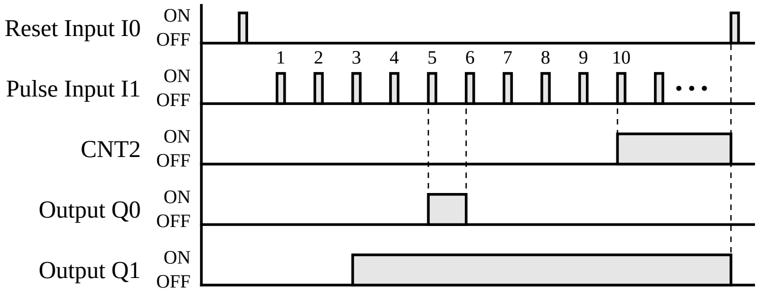
Program List

Prgm Adrs	Instruction	Data
0	LOD	I0
1	LOD	I1
2	CNT	2
		10
4	CC=	2
		5
6	OUT	Q0
7	CC≥	2
		3
9	OUT	Q1

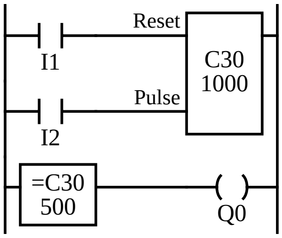
Key Operation

LOD 10	SET I	0	↵	
LOD 10	SET I	1 BPS	↵	
CNT C	2 BRD			
LOD 10	1 BPS	0	↵	
5 CC=	2 BRD	LOD 10	5 CC=	↵
OUT 16	F RST Q	0	↵	
6 CC>=	2 BRD	LOD 10	3 BPP	↵
OUT 16	F RST Q	1 BPS	↵	

Timing Chart



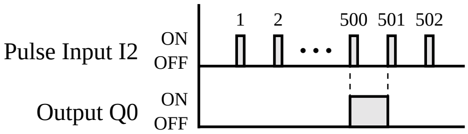
Ladder Diagram 2



Program List

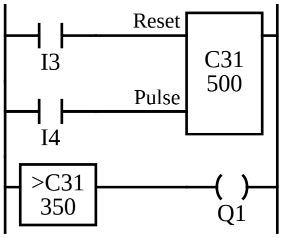
Prgm Adrs	Instruction	Data
0	LOD	I1
1	LOD	I2
2	CNT	30
		1000
4	CC=	30
		500
6	OUT	Q0

Timing Chart



Output Q0 is on when counter C30 current value is 500.

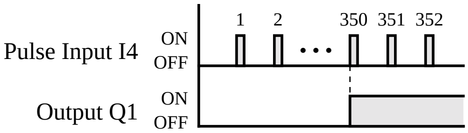
Ladder Diagram 3



Program List

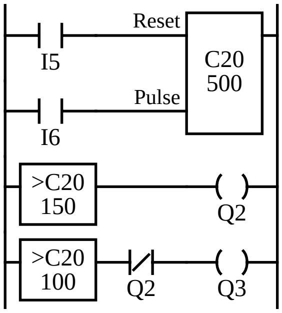
Prgm Adrs	Instruction	Data
0	LOD	I3
1	LOD	I4
2	CNT	31
		500
4	CC≥	31
		350
6	OUT	Q1

Timing Chart



Output Q1 is turned on when counter C31 current value reaches 350 and remain on until counter C31 is reset.

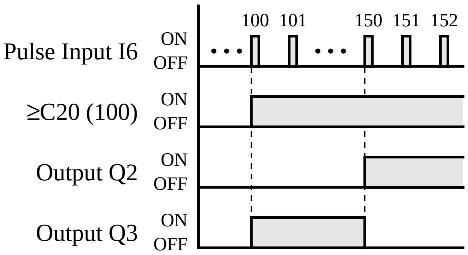
Ladder Diagram 4



Program List

Prgm Adrs	Instruction	Data
0	LOD	I5
1	LOD	I6
2	CNT	20
		500
4	CC≥	20
		150
6	OUT	Q2
7	CC≥	20
		100
9	AND NOT	Q2
10	OUT	Q3

Timing Chart



Output Q3 is on when counter C20 current value is between 100 and 150.

SFR and SFRN (Forward and Reverse Shift Register)

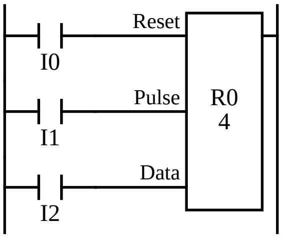
The shift register consists of a total of 64 bits which are allocated to R0 through R63 in the standard processing mode. In the high-speed processing mode, 32 bits are available for the shift register allocated to R0 through R31. Any number of available bits can be selected to form a train of bits which store on or off status. The on/off data of constituent bits is shifted in the forward direction (forward shift register) or in the reverse direction (reverse shift register) when a pulse input is turned on.

Forward Shift Register (SFR)

When SFR instructions are programmed, two addresses are always required. The SFR instruction is keyed, followed by a shift register number selected from appropriate operand numbers. The shift register number corresponds to the first, or head bit. The number of bits is the second required address after the SFR instruction.

The SFR instruction requires three inputs. The forward shift register circuit must be programmed in the following order: reset input, pulse input, data input, and the SFR instruction, followed by two required addresses.

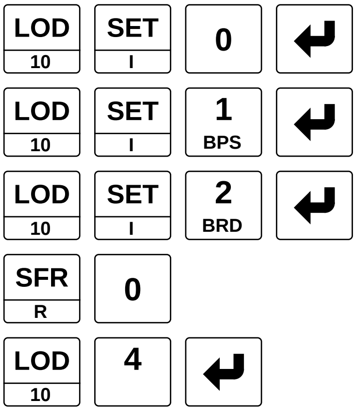
Ladder Diagram



First Bit: 0 to 63 (standard processing mode)
0 to 31 (high-speed processing mode)

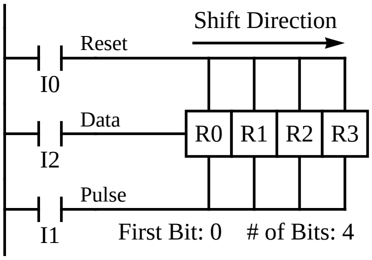
# of Bits: 1 to 64 (standard processing mode)
1 to 32 (high-speed processing mode)

Key Operation



Note: To enter a decimal constant as the number of bits, press the LOD/10 key followed by the number of bits.

Structural Diagram



Program List

Prgm Adrs	Instruction	Data
0	LOD	I0
1	LOD	I1
2	LOD	I2
3	SFR	0 4

Reset Input

The reset input will cause the value of each bit of the shift register to return to zero. Initialize pulse special internal relay, M301, may be used to initialize the shift register at start-up.

Pulse Input

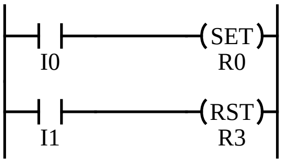
The pulse input triggers the data to shift. The shift is in the forward direction for a forward shift register and in reverse for a reverse shift register. A data shift will occur upon the leading edge of a pulse; that is, when the pulse *turns on*. If the pulse has been on and stays on, no data shift will occur.

Data Input

The data input is the information which is shifted into the first bit when a forward data shift occurs, or into the last bit when a reverse data shift occurs.

Note: When power is turned off, the statuses of all shift register bits are normally cleared. It is also possible to maintain the statuses of shift register bits by setting FUN4 as required. See page 5-5.

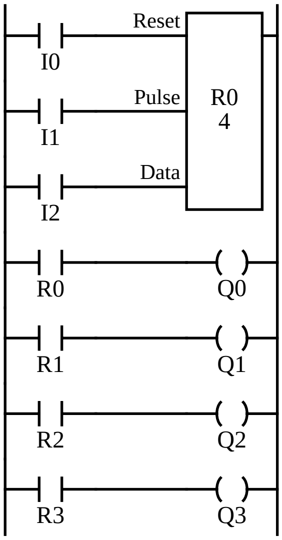
Setting and Resetting Shift Register Bits



- Any shift register bit can be turned on using the SET instruction.
- Any shift register bit can be turned off using the RST instruction.
- The SET or RST instruction is actuated by any input condition.

Forward Shift Register (SFR), continued

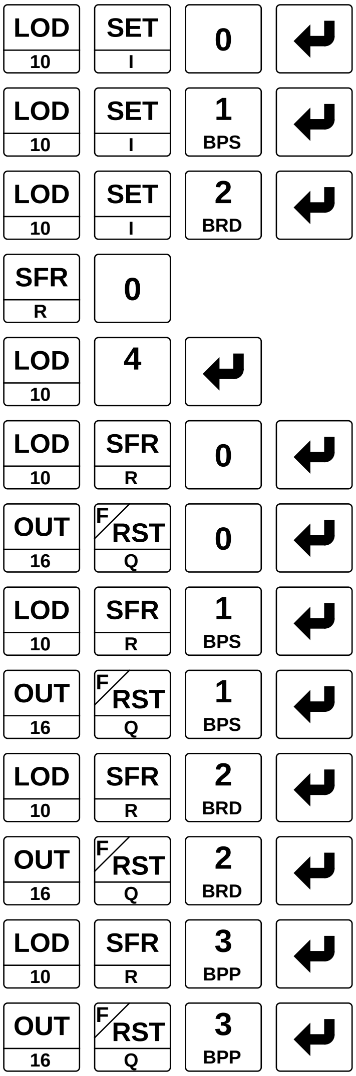
Ladder Diagram



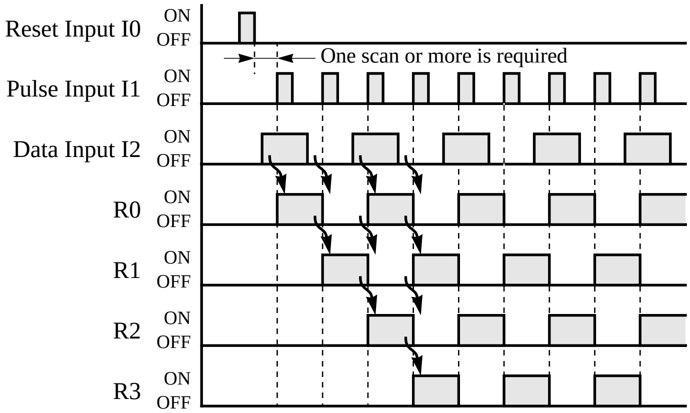
Program List

Prgm Adrs	Instruction	Data
0	LOD	I0
1	LOD	I1
2	LOD	I2
3	SFR	0
		4
5	LOD	R0
6	OUT	Q0
7	LOD	R1
8	OUT	Q1
9	LOD	R2
10	OUT	Q2
11	LOD	R3
12	OUT	Q3

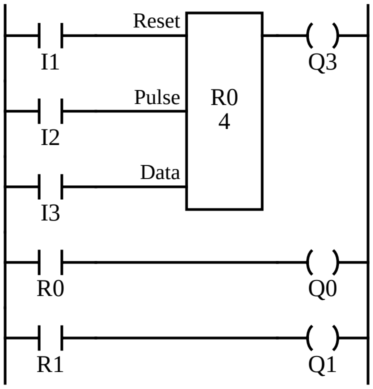
Key Operation


LOD 10 | SET I | 0 | ↵
LOD 10 | SET I | 1 BPS | ↵
LOD 10 | SET I | 2 BRD | ↵
SFR R | 0
LOD 10 | 4 | ↵
LOD 10 | SFR R | 0 | ↵
OUT 16 | F RST Q | 0 | ↵
LOD 10 | SFR R | 1 BPS | ↵
OUT 16 | F RST Q | 1 BPS | ↵
LOD 10 | SFR R | 2 BRD | ↵
OUT 16 | F RST Q | 2 BRD | ↵
LOD 10 | SFR R | 3 BPP | ↵
OUT 16 | F RST Q | 3 BPP | ↵

Timing Chart



Ladder Diagram



Program List

Prgm Adrs	Instruction	Data
0	LOD	I1
1	LOD	I2
2	LOD	I3
3	SFR	0
		4
5	OUT	Q3
6	LOD	R0
7	OUT	Q0
8	LOD	R1
9	OUT	Q1

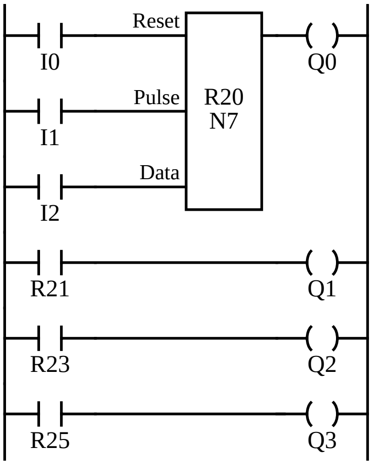
- The last bit status output can be programmed directly after the SFR instruction with two required addresses is keyed. In this example, the status of bit R3 is read to output Q3.
- Each bit can be loaded using the LOD SFR R# instructions.

Reverse Shift Register (SFRN)

For reverse shifting, use the SFR instruction followed by the NOT instruction. When SFRN instructions are programmed, two addresses are always required. The SFR and NOT instructions are keyed, followed by a shift register number selected from appropriate operand numbers. The shift register number corresponds to the lowest bit number in a string. The number of bits is the second required address after the SFR NOT instructions.

The SFRN instruction requires three inputs. The reverse shift register circuit must be programmed in the following order: reset input, pulse input, data input, and the SFR and NOT instructions, followed by two required addresses.

Ladder Diagram



Last Bit:
0 to 63 (standard processing mode)
0 to 31 (high-speed processing mode)

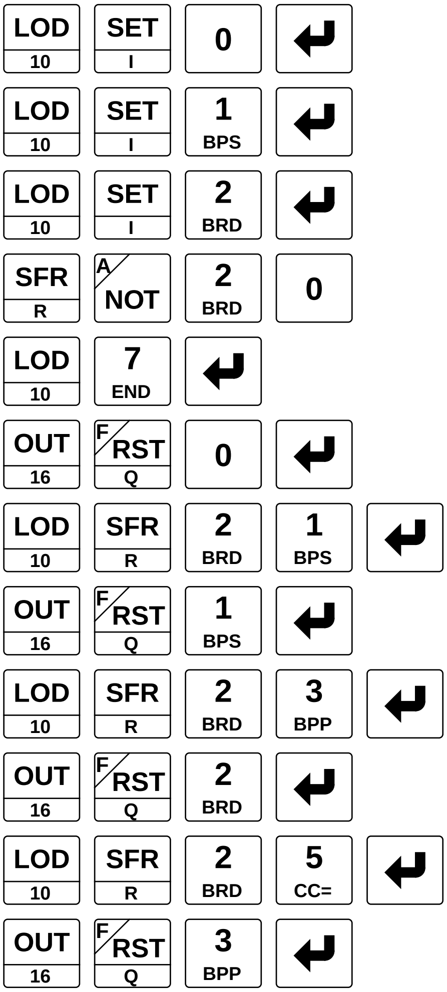
# of Bits:
1 to 64 (standard processing mode)
1 to 32 (high-speed processing mode)

- The last bit status output can be programmed directly after the SFRN instruction with two required addresses is keyed. In this example, the status of bit R20 is read to output Q0.
- Each bit can be loaded using the LOD SFR R# instructions.
- For details of reset, pulse,

Program List

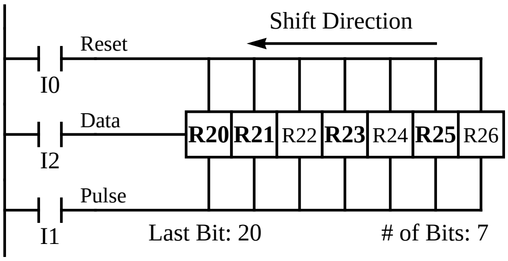
Prgm Adrs	Instruction	Data
0	LOD	I0
1	LOD	I1
2	LOD	I2
3	SFR NOT	20
		7
5	OUT	Q0
6	LOD	R21
7	OUT	Q1
8	LOD	R23
9	OUT	Q2
10	LOD	R25
11	OUT	Q3

Key Operation



Note: To enter a decimal constant as the number of bits, press the LOD/10 key followed by the number of bits.

Structural Diagram



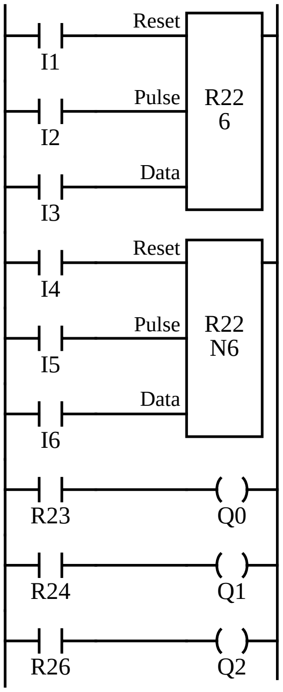
Note: Output is initiated only for those bits highlighted in bold print.

Note: When power is turned off, the statuses of all shift register bits are normally cleared. It is also possible to maintain the statuses of shift register bits by setting FUN4 as required. See page 5-5.

Bidirectional Shift Register

A bidirectional shift register can be created by first keying in the SFR instruction, complete with two addresses, as detailed in the Forward Shift Register section on page 7-23. Next, the SFR and NOT instructions are keyed in, complete with two addresses, as detailed in the Reverse Shift Register section on page 7-25.

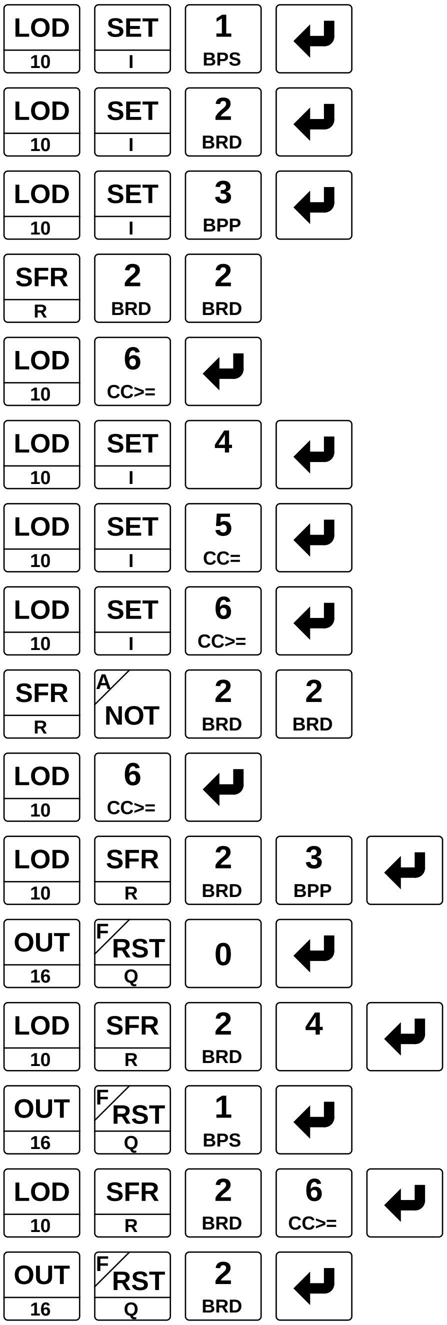
Ladder Diagram



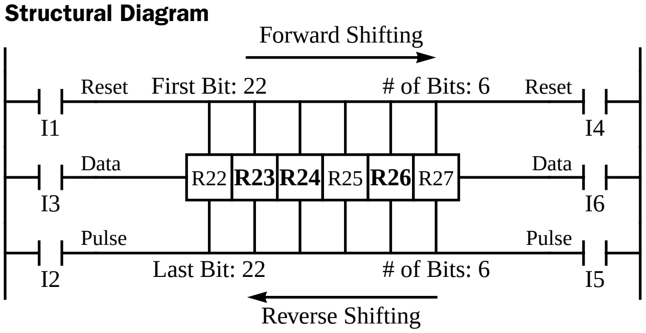
Program List

Prgm Adrs	Instruction	Data
0	LOD	I1
1	LOD	I2
2	LOD	I3
3	SFR	22 6
5	LOD	I4
6	LOD	I5
7	LOD	I6
8	SFR NOT	22 6
10	LOD	R23
11	OUT	Q0
12	LOD	R24
13	OUT	Q1
14	LOD	R26
15	OUT	Q2

Key Operation


LOD 10 | SET I | 1 BPS | ↵
LOD 10 | SET I | 2 BRD | ↵
LOD 10 | SET I | 3 BPP | ↵
SFR R | 2 BRD | 2 BRD
LOD 10 | 6 CC>= | ↵
LOD 10 | SET I | 4 | ↵
LOD 10 | SET I | 5 CC= | ↵
LOD 10 | SET I | 6 CC>= | ↵
SFR R | A NOT | 2 BRD | 2 BRD
LOD 10 | 6 CC>= | ↵
LOD 10 | SFR R | 2 BRD | 3 BPP | ↵
OUT 16 | F RST Q | 0 | ↵
LOD 10 | SFR R | 2 BRD | 4 | ↵
OUT 16 | F RST Q | 1 BPS | ↵
LOD 10 | SFR R | 2 BRD | 6 CC>= | ↵
OUT 16 | F RST Q | 2 BRD | ↵

Structural Diagram



Note: Output is initiated only for those bits highlighted in bold print.

SOTU and SOTD (Single Output Up and Down)

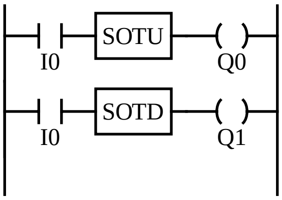
The SOTU instruction “looks for” the transition of a given input from off to on. The SOTD instruction looks for the transition of a given input from on to off. When this transition occurs, the desired output will turn on for one scan. The SOTU or SOTD instruction converts an input signal to a “one-shot” pulse signal. The SOTU or SOTD instruction is followed by one address.

SOTU and SOTD instructions can be used repeatedly.

If operation is started while the given input is already on, the SOTU output will not turn on. The transition from off to on is what triggers the SOTU instruction.

When a relay of the relay output type MICRO^3 base unit is defined as the SOTU or SOTD output, it may not operate if the scan time is not compatible with relay requirements.

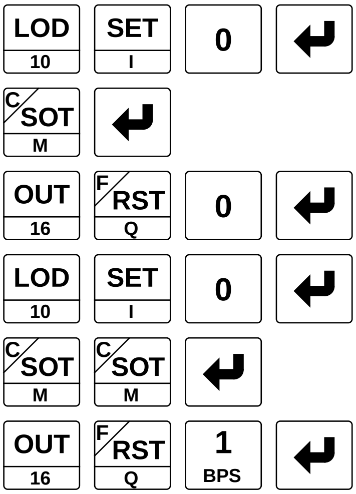
Ladder Diagram



Program List

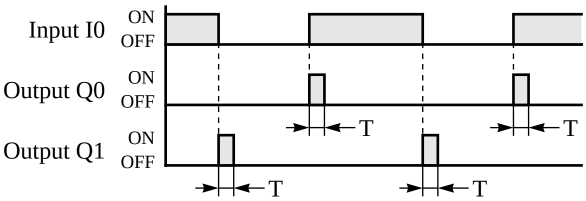
Prgm Adrs	Instruction	Data
0	LOD	I0
1	SOTU	
2	OUT	Q0
3	LOD	I0
4	SOTD	
5	OUT	Q1

Key Operation



Note: Pressing the SOT key on the program loader programs the SOTU or SOTD instruction alternately.

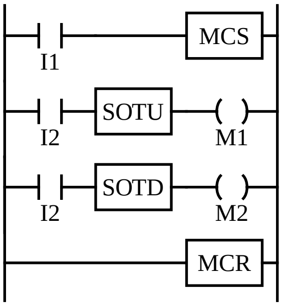
Timing Chart



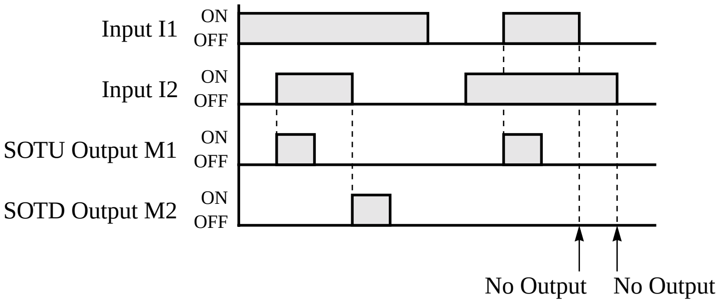
Note: “T” equals one scan time (one-shot pulse).

There is a special case when the SOTU and SOTD instructions are used between the MCS and MCR instructions (which are detailed on page 7-28). If input I2 to the SOTU instruction turns on while input I1 to the MCS instruction is on, then the SOTU output turns on. If input I2 to the SOTD instruction turns off while input I1 is on, then the SOTD output turns on. If input I1 turns on while input I2 is on, then the SOTU output turns on. However, if input I1 turns off while input I2 is on, then the SOTD output does not turn on as shown below.

Ladder Diagram



Timing Chart



MCS and MCR (Master Control Set and Reset)

The MCS (master control set) instruction is usually used in combination with the MCR (master control reset) instruction. The MCS instruction can also be used with the END instruction, instead of the MCR instruction.

When the input preceding the MCS instruction is off, the MCS is executed so that all inputs to the portion between the MCS and the MCR are forced off. When the input preceding the MCS instruction is on, the MCS is not executed so that the program following it is executed according to the actual input statuses.

When the input condition to the MCS instruction is off and the MCS is executed, other instructions between the MCS and MCR are executed as follows:

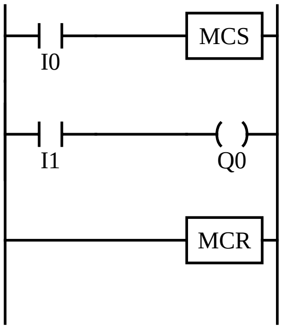
Instruction	Status
SOTU	Rising edges (ON pulses) are not detected.
SOTD	Falling edges (OFF pulses) are not detected.
OUT	All are turned off.
OUTN	All are turned on.
SET and RST	All are held in current status.
TIM, TMH, and TMS	Current values are reset to zero. Timeout statuses are turned off.
CNT	Current values are held. Pulse inputs are turned off. Countout statuses are turned off.
SFR	Shift register bit statuses are held. Pulse inputs are turned off. The output from the last bit is turned off.

Input conditions cannot be set for the MCR instruction.

More than one MCS instruction can be set by one MCR instruction.

Corresponding MCS/MCR instructions cannot be nested within another pair of corresponding MCS/MCR instructions.

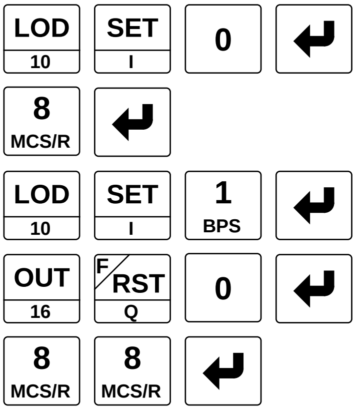
Ladder Diagram



Program List

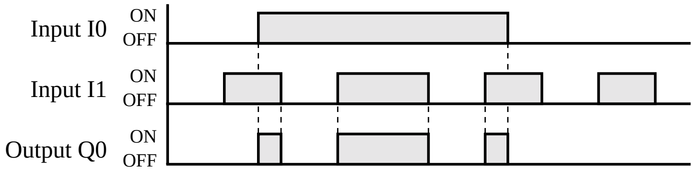
Prgm Adrs	Instruction	Data
0	LOD	I0
1	MCS	
2	LOD	I1
3	OUT	Q0
4	MCR	

Key Operation



Note: Pressing the MCS/R key on the program loader programs the MCS or MCR instruction alternately.

Timing Chart



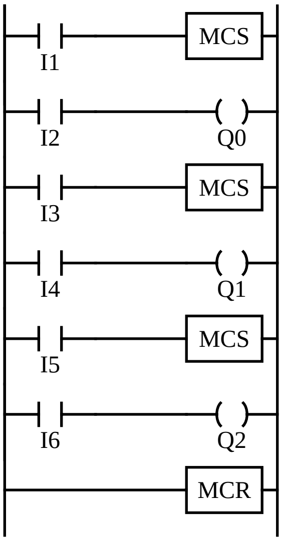
When input I0 is off, MCS is executed so that the subsequent input is forced off.

When input I0 is on, MCS is not executed so that the following program is executed according to the actual input statuses.

MCS and MCR (Master Control Set and Reset), continued

Multiple Usage of MCS instructions

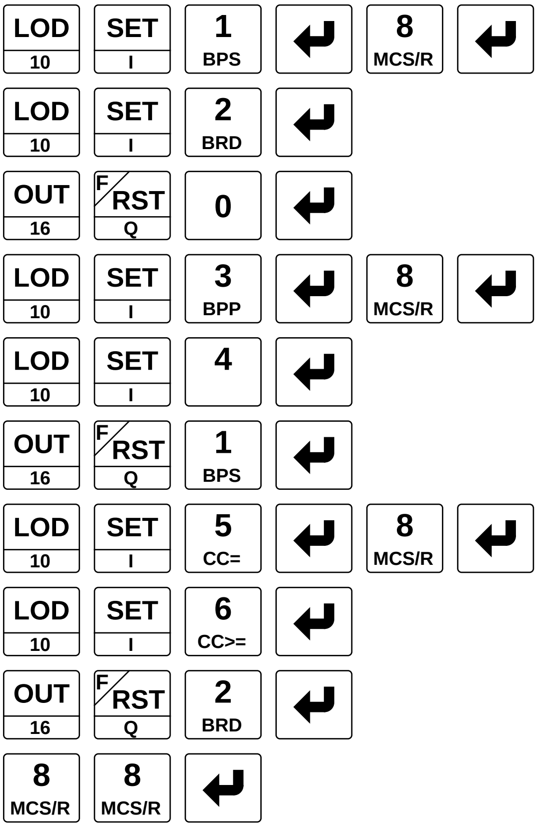
Ladder Diagram



Program List

Prgm Adrs	Instruction	Data
0	LOD	I1
1	MCS	
2	LOD	I2
3	OUT	Q0
4	LOD	I3
5	MCS	
6	LOD	I4
7	OUT	Q1
8	LOD	I5
9	MCS	
10	LOD	I6
11	OUT	Q2
12	MCR	

Key Operation


LOD 10 | SET I | 1 BPS | ↵ | 8 MCS/R | ↵
LOD 10 | SET I | 2 BRD | ↵
OUT 16 | F RST Q | 0 | ↵
LOD 10 | SET I | 3 BPP | ↵ | 8 MCS/R | ↵
LOD 10 | SET I | 4 | ↵
OUT 16 | F RST Q | 1 BPS | ↵
LOD 10 | SET I | 5 CC= | ↵ | 8 MCS/R | ↵
LOD 10 | SET I | 6 CC>= | ↵
OUT 16 | F RST Q | 2 BRD | ↵
8 MCS/R | 8 MCS/R | ↵

This master control circuit will give priority to I1, I3, and I5, in that order.

When input I1 is off, the first MCS is executed so that subsequent inputs I2 through I6 are forced off.

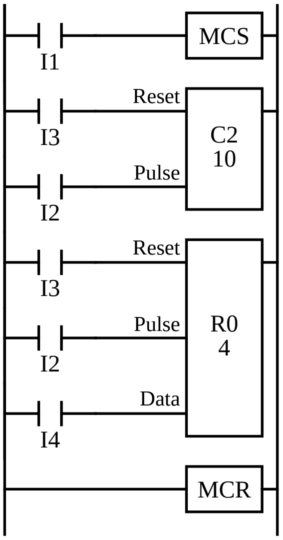
When input I1 is on, the first MCS is not executed so that the following program is executed according to the actual input statuses of I2 through I6.

When I1 is on and I3 is off, the second MCS is executed so that subsequent inputs I4 through I6 are forced off.

When both I1 and I3 are on, the first and second MCS's are not executed so that the following program is executed according to the actual input statuses of I4 through I6.

Counter and Shift Register in Master Control Circuit

Ladder Diagram

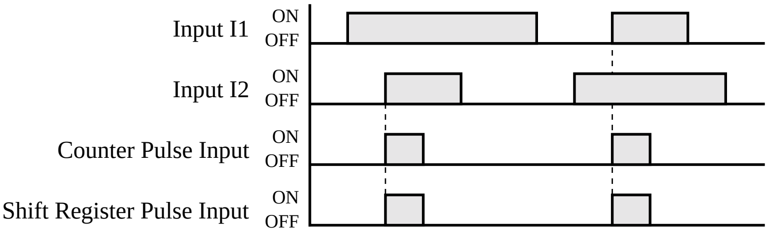


When input I1 is on, the MCS is not executed so that the counter and shift register are executed according to actual statuses of subsequent inputs I2 through I4.

When input I1 is off, the MCS is executed so that subsequent inputs I2 through I4 are forced off.

When input I1 is turned on while input I2 is on, the counter and shift register pulse inputs are turned on as shown below.

Timing Chart



JMP (Jump) and JEND (Jump End)

The JMP (jump) instruction is usually used in combination with the JEND (jump end) instruction. At the end of a program, the JMP instruction can also be used with the END instruction, instead of the JEND instruction.

These instructions are used to proceed through the portion of the program between the JMP and the JEND *without* processing. This is similar to the MCS/MCR instructions, except that the portion of the program between the MCS and MCR instruction *is* executed.

When the operation result immediately before the JMP instruction is on, the JMP is valid and the program is *not* executed. When the operation result immediately before the JMP instruction is off, the JMP is invalid and the program is executed.

When the input condition to the JMP instruction is on and the JMP is executed, other instructions between the JMP and JEND are executed as follows:

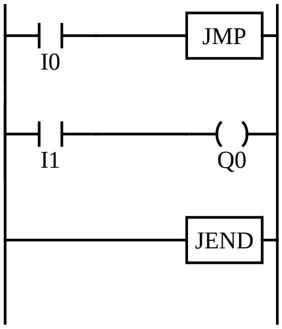
Instruction	Status
SOTU	Rising edges (ON pulses) are not detected.
SOTD	Falling edges (OFF pulses) are not detected.
OUT and OUTN	All are held in current status.
SET and RST	All are held in current status.
TIM, TMH, and TMS	Current values are held. Timeout statuses are held.
CNT	Current values are held. Pulse inputs are turned off. Countout statuses are held.
SFR	Shift register bit statuses are held. Pulse inputs are turned off. The output from the last bit is held.

Input conditions cannot be set for the JEND instruction.

More than one JMP instruction can be set by one JEND instruction.

Corresponding JMP/JEND instructions cannot be nested within another pair of corresponding JMP/JEND instructions.

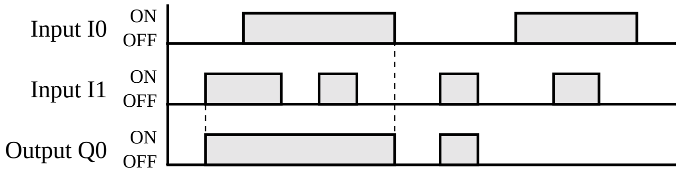
Ladder Diagram



Program List

Prgm Adrs	Instruction	Data
0	LOD	I0
1	JMP	
2	LOD	I1
3	OUT	Q0
4	JEND	

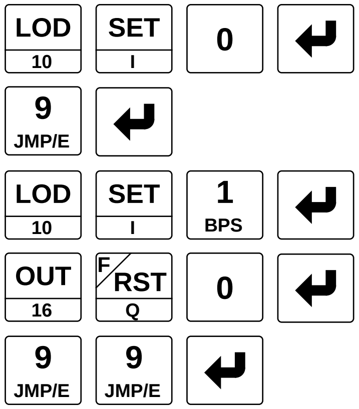
Timing Chart



When input I0 is on, JMP is executed so that the subsequent output status is held.

When input I0 is off, JMP is not executed so that the following program is executed according to the actual input statuses.

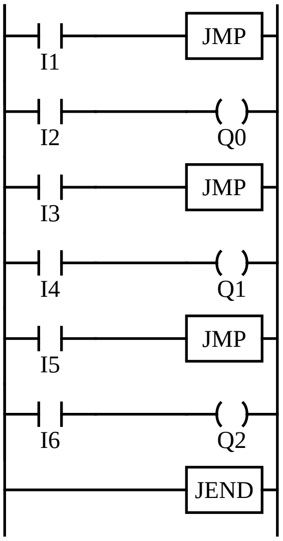
Key Operation



Note: Pressing the JMP/E key on the program loader programs the JMP or JEND instruction alternately.

JMP (Jump) and JEND (Jump End), continued

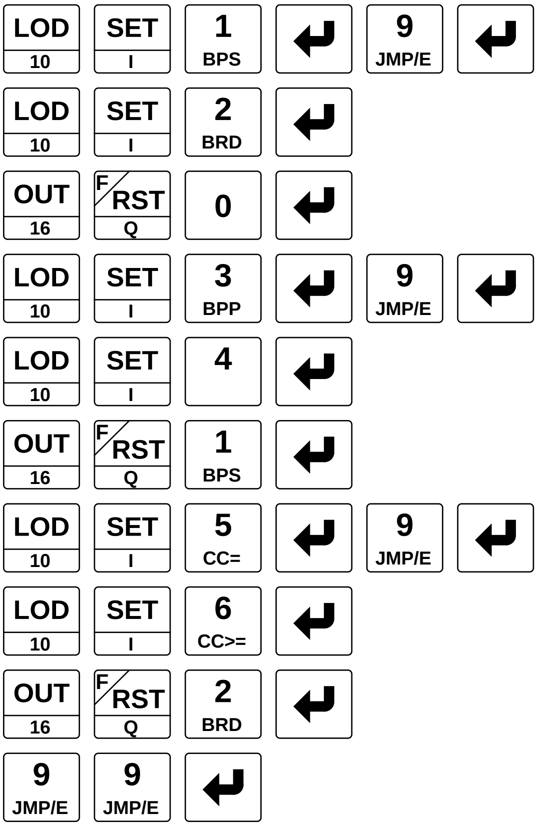
Ladder Diagram



Program List

Prgm Adrs	Instruction	Data
0	LOD	I1
1	JMP	
2	LOD	I2
3	OUT	Q0
4	LOD	I3
5	JMP	
6	LOD	I4
7	OUT	Q1
8	LOD	I5
9	JMP	
10	LOD	I6
11	OUT	Q2
12	JEND	

Key Operation


LOD 10, SET I, 1 BPS, ↵, 9 JMP/E, ↵
LOD 10, SET I, 2 BRD, ↵
OUT 16, F RST Q, 0, ↵
LOD 10, SET I, 3 BPP, ↵, 9 JMP/E, ↵
LOD 10, SET I, 4, ↵
OUT 16, F RST Q, 1 BPS, ↵
LOD 10, SET I, 5 CC=, ↵, 9 JMP/E, ↵
LOD 10, SET I, 6 CC>=, ↵
OUT 16, F RST Q, 2 BRD, ↵
9 JMP/E, 9 JMP/E, ↵

This jump circuit will give priority to I1, I3, and I5, in that order.

When input I1 is on, the first JMP is executed so that subsequent output statuses of Q0 through Q2 are held.

When input I1 is off, the first JMP is not executed so that the following program is executed according to the actual input statuses of I2 through I6.

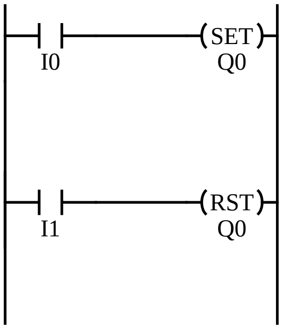
When I1 is off and I3 is on, the second JMP is executed so that subsequent output statuses of Q1 and Q2 are held.

When both I1 and I3 are off, the first and second JMP's are not executed so that the following program is executed according to the actual input statuses of I4 through I6.

SET and RST (Reset)

The SET and RST (reset) instructions are used to set (on) or reset (off) outputs, internal relays, and shift register bits. The SET and RST instructions require one address which must be selected from the appropriate operand numbers. The same output can be set and reset many times within a program. SET and RST instructions operate in every scan while the input is on.

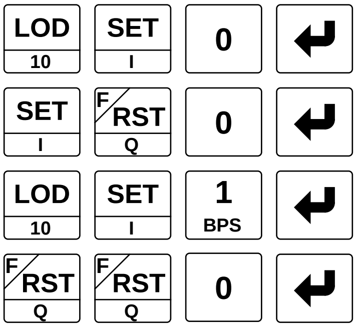
Ladder Diagram



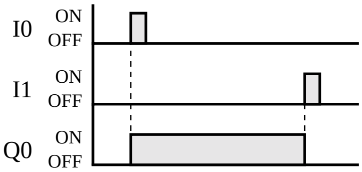
Program List

Prgm Adrs	Instruction	Data
0	LOD	I0
1	SET	Q0
2	LOD	I1
3	RST	Q0

Key Operation



Timing Chart



Valid Operands

Instruction	I	Q	M	T	C	R
SET	—	0-31	0-287	—	—	0-63
RST	—	0-31	0-287	—	—	0-63

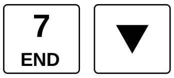
When using in the high-speed processing mode, operands are limited. See page 6-1.

END

The END instruction is always required at the end of a program; however, it is not necessary to program the END instruction after the last programmed instruction. The END instruction already exists at every unused address. (When an address is used for programming, the END instruction is removed.)

The END key is provided on the program loader for purposes other than programming an END instruction, (which is rarely necessary). A useful purpose for the END key is to find the program address for the end of a program:

Key Operation



A *scan* is the execution of all instructions from address zero to the END instruction. The time required for this execution is referred to as one *scan time*. The scan time varies with respect to program length, which corresponds to the address where the END instruction is found.

During the scan time, program instructions are processed sequentially. This is why the output instruction closest to the END instruction has priority over a previous instruction for the same output. No output is initiated until all logic within a scan is processed.

Output occurs simultaneously, and this is the first part of the END instruction execution. The second part of the END instruction execution is to monitor all inputs, also done simultaneously. Then program instructions are ready to be processed sequentially once again.

Introduction

This chapter describes the advanced instruction menus, terms, available operands, formats, and data types used for advanced instructions.

Advanced Instruction Menus

Advanced instructions are programmed using the ADV key of the program loader. To program an advanced instruction using the program loader, select the address where you want to program an advanced instruction, and press the ADV and ▼ keys to display the first menu of advanced instructions:

ADV ▼

```
[MENU1]
 0:NOP   3:+-*/
 1:MOV   4:LOGIC
⬇2:CMP   5:SFT
```

The MENU1 of advanced instructions include six headings of submenus. To show a submenu, press a number 0 through 5. When the submenu is displayed, press a number to program a required advanced instruction, then the program screen is displayed to enter operands.

The NOP instruction is entered at the selected address in the editor screen immediately when 0 is pressed while the MENU1 is displayed.

It is also possible to jump from the editor screen to the submenu by pressing the ADV key and a number, 0 through 9, A or B.

Press the 0 key.

```
   0 NOP
   1▶END
   2 END
   3 END
```

Press the 1 key.

```
[MOV MENU]
 1:MOV   4:IMOVN
 2:MOVN
 3:IMOV
```

Press the 2 key.

```
[CMP MENU]
 1:=     4:>
 2:<>    5:<=
 3:<     6:>=
```

Press the 3 key.

```
[+-*/ MENU]
 1:ADD+  4:DIV/
 2:SUB-
 3:MUL*
```

Press the 4 key.

```
[LOGIC MENU]
 1:ANDW
 2:ORW
 3:XORW
```

Press the 5 key.

```
[BIT SFT MENU]
 1:SFTL  4:ROTR
 2:SFTR
 3:ROTL
```

To show the MENU2 of advanced instructions, press the ▼ key while the MENU1 is displayed.

▼

```
[MENU2]
⬆7:CLK   A:HSC
 8:I/F   B:COMM
 9:EXT
```

The MENU2 includes five submenu headings. As described above, to go to a submenu, press a number key 7 through B. When the submenu is displayed, press a number to program a required advanced instruction, then the program screen is displayed to enter operands

To go back to MENU1 from MENU2, press the ▲ key.

Details of programming advanced instructions are described in the following chapters.

Press the 7 key.

```
[CLOCK MENU]
 1:CALR  4:CLKW
 2:CALW  5:ADJ
 3:CLKR
```

Press the 8 key.

```
[I/F MENU]
 1:DISP  4:ANR1
 2:DGRD
 3:ANR0
```

Press the 9 key.

```
[EXT MENU]
 1:PULS
 2:PWM
 3:A/D
```

Press the A key.

```
[HSC MENU]
 1:HSC0  4:HSC3
 2:HSC1
 3:HSC2
```

Press the B key.

```
[COMM MENU]
 1:TXD
 2:RXD
 3:CMP2
```

Advanced Instruction List

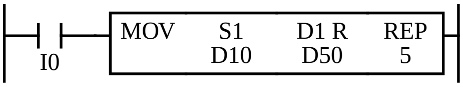
Group	Number	Symbol	Name	Addrs Req'd	Available on
NOP	0	NOP	No Operation	1	All MICRO^3 and MICRO^{3}C
Move	11	MOV	Move	3 or 4	All MICRO^3 and MICRO^{3}C
	12	MOVN	Move Not	3 or 4	All MICRO^3 and MICRO^{3}C
	13	IMOV	Indirect Move	5 or 6	All MICRO^3 and MICRO^{3}C
	14	IMOVN	Indirect Move Not	5 or 6	All MICRO^3 and MICRO^{3}C
Comparison	21	CMP=	Compare Equal To	4 or 5	All MICRO^3 and MICRO^{3}C
	22	CMP<>	Compare Unequal To	4 or 5	All MICRO^3 and MICRO^{3}C
	23	CMP<	Compare Less Than	4 or 5	All MICRO^3 and MICRO^{3}C
	24	CMP>	Compare Greater Than	4 or 5	All MICRO^3 and MICRO^{3}C
	25	CMP<=	Compare Less Than or Equal To	4 or 5	All MICRO^3 and MICRO^{3}C
	26	CMP>=	Compare Greater Than or Equal To	4 or 5	All MICRO^3 and MICRO^{3}C
Binary Arithmetic	31	ADD	Addition	4 or 5	All MICRO^3 and MICRO^{3}C
	32	SUB	Subtraction	4 or 5	All MICRO^3 and MICRO^{3}C
	33	MUL	Multiplication	4 or 5	All MICRO^3 and MICRO^{3}C
	34	DIV	Division	4 or 5	All MICRO^3 and MICRO^{3}C
Boolean Computation	41	ANDW	AND Word	4 or 5	All MICRO^3 and MICRO^{3}C
	42	ORW	OR Word	4 or 5	All MICRO^3 and MICRO^{3}C
	43	XORW	Exclusive OR Word	4 or 5	All MICRO^3 and MICRO^{3}C
Bit Shift and Rotate	51	SFTL	Shift Left	3	All MICRO^3 and MICRO^{3}C
	52	SFTR	Shift Right	3	All MICRO^3 and MICRO^{3}C
	53	ROTL	Rotate Left	3	All MICRO^3 and MICRO^{3}C
	54	ROTR	Rotate Right	3	All MICRO^3 and MICRO^{3}C
Real-time Clock/Calendar	71	CALR	Calendar Read	2	16- and 24-I/O types only
	72	CALW	Calendar Write	2	16- and 24-I/O types only
	73	CLKR	Clock Read	2	16- and 24-I/O types only
	74	CLKW	Clock Write	2	16- and 24-I/O types only
	75	ADJ	Adjust	1	16- and 24-I/O types only
Interface	81	DISP	Display	4	Transistor output type only
	82	DGRD	Digital Switch Read	4	Transistor output type only
	83	ANR0	Analog Read 0	2	All MICRO^3 and MICRO^{3}C
	84	ANR1	Analog Read 1	2	16- and 24-I/O MICRO^3 only
Pulse and A/D Conversion	91	PULS	Pulse Output	3	Transistor output type only
	92	PWM	Pulse Width Modulation	3	Transistor output type only
	93	A/D	Analog To Digital Conversion	2	All MICRO^3 and MICRO^{3}C
High-speed Counter	A1	HSC0	High-speed Counter 0 (32 bits)	4	24V DC input type only
	A2	HSC1	High-speed Counter 1 (32 bits)	2	24V DC input type only
	A3	HSC2	High-speed Counter 2 (32 bits)	4	Transistor output type only
	A4	HSC3	High-speed Counter 3 (16 bits)	2	24V DC input type only
User Communication	B1	TXD	Transmit	5 to 404	MICRO^{3}C only
	B2	RXD	Receive	5 to 404	MICRO^{3}C only
Comparison	B3	CMP2	Double-word Comparison	4 or 5	MICRO^{3}C only

Note: For details about the TXD, RXD, and CMP2 instructions, see the MICRO^{3}C User's Manual.

Programming Advanced Instructions Using Program Loader

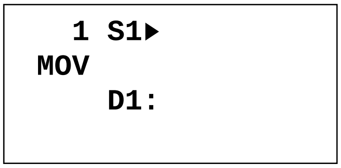
Repeat operation can be used with some advanced instructions. When repeat is designated for a source or destination operand, consecutive operands as many as the repeat cycles are processed starting with the designated operand. Details of repeat operation are described in following chapters. Repeat operation can be programmed using the program loader, shown below.

Example: Program the MOV (move) instruction with 5 repeat cycles for destination operand D1.



First, program the input condition for the advanced instruction pressing the LOD, I, 0, and ↵ keys. Then program the MOV (move) instruction pressing the keys shown below.

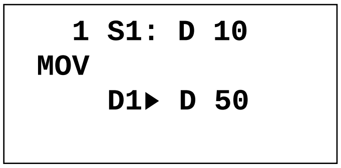
ADV 1 BPS 1 BPS



```
  1 S1▶
 MOV
    D1:
```

Program data registers D10 and D50 for source operand S1 and destination operand D1. The cursor can be moved using the ▼ and ▲ keys.

E/OR/D 1 BPS 0 ▼   E/OR/D 5 CC= 0

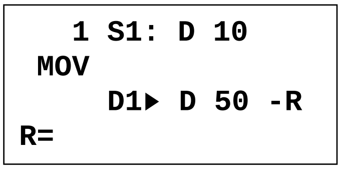


```
  1 S1: D 10
 MOV
    D1▶ D 50
```

Note: To enter a decimal or hexadecimal constant for a source operand, press the LOD/10 or OUT/16 key followed by the constant value.

With the cursor placed at the operand to designate repeat operation, press the REP (repeat) key. The display changes as shown on the right.

B/REP

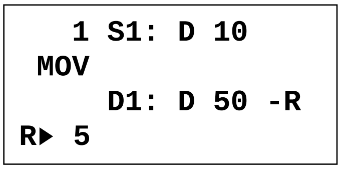


```
  1 S1: D 10
 MOV
    D1▶ D 50 -R
R=
```

On the right of the operand to repeat, "–R" is displayed.

Move down the cursor to the bottom line, and enter the quantity of repeat cycles, which can be 1 through 31.

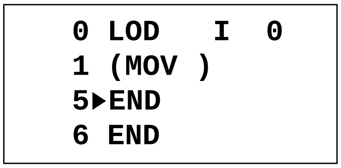
▼ 5 CC=



```
  1 S1: D 10
 MOV
    D1: D 50 -R
R▶ 5
```

When programming of all operands and repeat cycles is complete, return to the normal editor screen.

↵

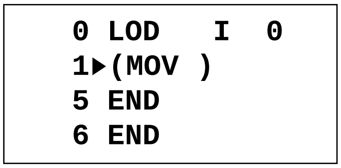


```
  0 LOD   I  0
  1 (MOV )
  5▶END
  6 END
```

Revising Advanced Instructions Using Program Loader

Place the cursor at the advanced instruction you want to revise using the ▼ and ▲ keys.

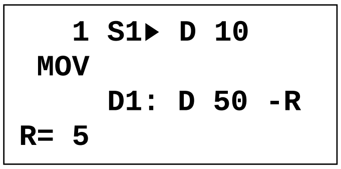
▲ ▼



```
  0 LOD   I  0
  1▶(MOV )
  5 END
  6 END
```

With the cursor placed at the advanced instruction, press the ▶ key.

▶

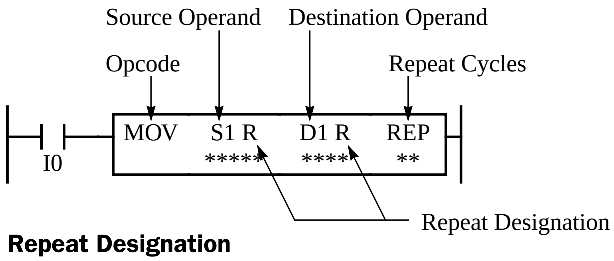


```
  1 S1▶ D 10
 MOV
    D1: D 50 -R
R= 5
```

The editor screen for the advanced instruction will be displayed with the programmed operands.

To exit after revising is complete, press the ↵ key.

Structure of an Advanced Instruction



Repeat Designation

Specifies whether repeat is used for the operand or not.

Repeat Cycles

Specifies the quantity of repeat cycles: 1 through 31.

Opcode

The opcode is a symbol to identify the advanced instruction.

Source Operand

The source operand specifies the 16-bit word data to be processed by the advanced instruction. Some advanced instructions require two source operands.

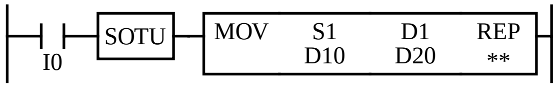
Destination Operand

The destination operand specifies the 16-bit word data to store the result of the advanced instruction. Some advanced instructions require two destination operands.

Input Condition for Advanced Instructions

Almost all advanced instructions must be preceded by a contact, except NOP (no operation) and HSC0 through HSC3 (high-speed counter) instructions. The input condition can be programmed using a bit operand such as input, output, internal relay, or shift register. Timer and counter can also be used as an input condition to turn on the contact when the timer times out or the counter counts out.

While the input condition is on, the advanced instruction is executed in each scan. To execute the advanced instruction only at the rising or falling edge of the input, use the SOTU or SOTD instruction.



Source and Destination Operands

The source and destination operands specify 16-bit word data. When a bit operand such as input, output, internal relay, or shift register is designated as a source or destination operand, 16 points starting with the designated number are processed as source or destination data. When a word operand such as timer or counter is designated as a source operand, the current value is read as a source data. When a timer or counter is designated as a destination operand, the result of the advanced instruction is set to the preset value for the timer or counter. When a data register is designated as a source or destination operand, the data is read from or written to the designated data register.

Using Timer or Counter as Source Operand

Since all timer instructions—TIM (100-msec timer), TMH (10-msec timer), and TMS (1-msec timer)—subtract from the preset value, the current value is decremented from the preset value and indicates the remaining time. As described above, when a timer is designated as a source operand of an advanced instruction, the current value, or the remaining time, of the timer is read as a source data. Reversible counters C0 and C1 start counting at the preset value and the current value is incremented or decremented from the preset value. Adding counters C2 through C31 start counting at 0, and the current value is incremented up to the preset value. When any counter is designated as a source operand of an advanced instruction, the current value is read as a source data.

Using Timer or Counter as Destination Operand

As described above, when a timer or counter is designated as a destination operand of an advanced instruction, the result of the advanced instruction is set to the preset value of the timer or counter. Since timer and counter preset values can be 0 through 9999, make sure that the result of the advanced instruction does not exceed 9999. If the result to be set to a timer or counter designated as destination exceeds 9999, a user program execution error will result, then error indicator ERR1 is lit and special internal relay M304 turns on.

When a timer or counter preset value is designated using a data register, the timer or counter cannot be designated as a destination of an advanced instruction. When executing such an advanced instruction, a user program execution error will result. If a timer or counter is designated as a destination of an advanced instruction and if the timer or counter is not programmed, then a user program execution error will also result. For details of user program execution error, see page 18-5.

Note: When a user program execution error occurs, the result is not set to the destination.

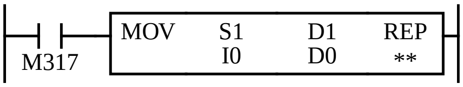
Using Input or Output as Source or Destination Operand

When an input or output is designated as a source or destination operand of an advanced instruction, 16 points starting with the designated number are used. Depending on the MICRO^3 base unit used alone or in the expansion link system, available input terminals are limited and special care is needed. In the standard processing mode, inputs I0 through I35 and outputs Q0 through Q31 are available. In the high-speed processing mode, available I/O numbers are limited to I0 through I15 and Q0 through Q11. Only these I/O numbers can be used for advanced instructions although the internal RAM has an I/O area of I0 through I37 and Q0 through Q37 for processing user programs.

Expansion Link Station	Input No.	10-I/O Base Units (6 input points)	16-I/O Base Units (9 input points)	24-I/O Base Units (14 input points)
Base Station	I0 to I5	Input terminals available	Input terminals available	Input terminals available
	I6 and I7	Always off (0)		
	I10			
	I11 to I15		Always off (0)	
	I16 and I17			Always off (0)
Expansion Station	I20 to I25	Input terminals available	Input terminals available	Input terminals available
	I26 and I27	Always off (0)		
	I30			
	I31 to I35		Always off (0)	
	I36 and I37			Always off (0)

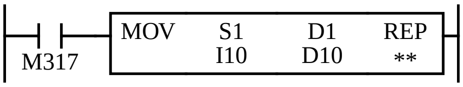
Expansion Link Station	Output No.	10-I/O Base Units (4 output points)	16-I/O Base Units (7 output points)	24-I/O Base Units (10 output points)
Base Station	Q0 to Q3	Output terminals available	Output terminals available	Output terminals available
	Q4 to Q6	Impossible to output		
	Q7		Impossible to output	
	Q10 and Q11			
	Q12 to Q17			Impossible to output
Expansion Station	Q20 to Q23	Output terminals available	Output terminals available	Output terminals available
	Q24 to Q26	Impossible to output		
	Q27		Impossible to output	
	Q30 and Q31			
	Q32 to Q37			Impossible to output

Input Source in the Move Instruction

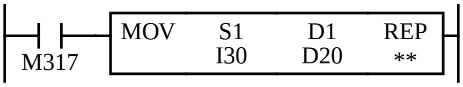


M317 is the in-operation special internal relay which remains on during operation.

The MOV (move) instruction sets data of 16 inputs I0 through I17 to data register D0. When using a 10-I/O MICRO^3 base unit, input terminals for I6 through I17 are not available, and these upper bits are always set to 0.



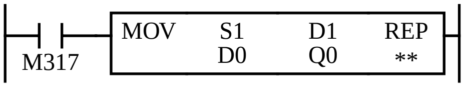
The MOV (move) instruction sets data of 16 inputs I10 through I27 to data register D10. When using two 16-I/O MICRO^3 base units in the expansion link system, input terminals for I11 through I17 are not available, and these intermediate bits are always set to 0.



The MOV (move) instruction sets data of 16 inputs starting with I30 to data register D20. When using two 24-I/O MICRO^3 base units in the expansion link system, input terminals for I36 through I47 are not available, and these upper bits are always set to 0.

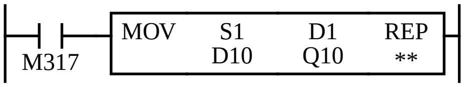
Using Input or Output as Source or Destination Operand, continued

Output Destination in the Move Instruction

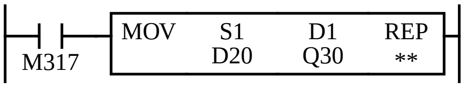


M317 is the in-operation special internal relay which remains on during operation.

The MOV (move) instruction sets data of data register D0 to 16 outputs Q0 through Q17. When using a 10-I/O MICRO^3 base unit, output terminals for Q4 through Q17 are not available, and these bits cannot be taken out.



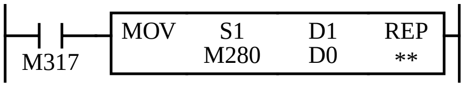
The MOV (move) instruction sets data of D10 to 16 outputs Q10 through Q27. When using two 16-I/O MICRO^3 base units in the expansion link system, output terminals for Q10 through Q17 and Q27 are not available, and these bits cannot be taken out.



The MOV (move) instruction sets data of D20 to 16 outputs starting with Q30. When using two 24-I/O MICRO^3 base units in the expansion link system, output terminals for Q32 through Q47 are not available, and these bits cannot be taken out.

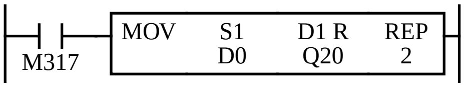
Discontinuity of Operand Areas

Each operand area is discrete and does not continue, for example, from input to output or from output to internal relay. In addition, special internal relays M290 through M317 are in a separate area from internal relays M0 through M287.



Since the internal relay ends at M287, the MOV (move) instruction sets only 8 internal relays M280 through M287 to data register D0. Upper 8 bits of D0 are set to 0.

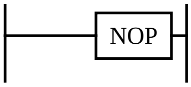
Advanced instructions execute operation only on the available operands in the valid area. If invalid operands are designated, a user program syntax error occurs when transferring the user program to the MICRO^3 base unit.



The MOV (move) instruction sets data of data register D0 to 16 outputs Q20 through Q37 in the first repeat cycle. The destination of the second cycle is the next 16 outputs Q40 through Q57, which are invalid, resulting in a user program syntax error, and error indicator ERR1 is lit.

For details of repeat operations, see the following chapters.

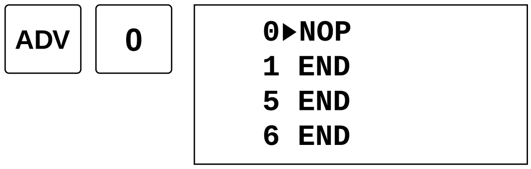
0 NOP (No Operation)



No operation is executed by the NOP instruction.

The NOP instruction may serve as a place holder. Another use would be to add a delay to the MICRO^3 scan time, in order to simulate communication with a machine or application, for debugging purposes.

Key Operation



```
0▶NOP
1 END
5 END
6 END
```

No operands can be programmed for the NOP instruction.

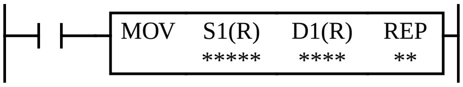
Details of all other advanced instructions are described in following chapters.

Introduction

Data can be moved using the MOV (move), MOVN (move not), IMOV (indirect move), or IMOVN (indirect move not) instruction. The moved data is 16-bit word data, and the repeat operation can also be used to increase the quantity of data moved. In the MOV or MOVN instruction, the source and destination operand are designated by S1 and D1 directly. In the IMOV or IMOVN instruction, the source and destination operand are determined by the offset values designated by S2 and D2 added to source operand S1 and destination operand D1.

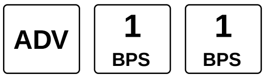
Since the move instructions are executed in each scan while input is on, a pulse input from an SOTU or SOTD instruction should be used as required.

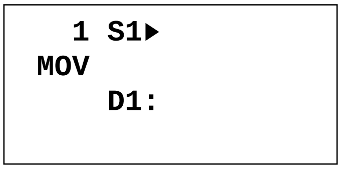
11 MOV (Move)



S1 → D1
When input is on, 16-bit word data from operand designated by S1 is moved to operand designated by D1.

Key Operation





```
   1 S1▶
 MOV
     D1:
```

Enter operands S1 and D1.

When repeat is required, press the REP key for the operand to repeat, and enter the number of repeat cycles.

To exit, press the ↵ key.

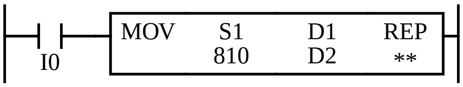
Valid Operands (Standard Processing)

Operand	Function	I	Q	M	T	C	R	D	Constant	Repeat
S1 (Source 1)	Operand to move	0-35	0-31	0-317	0-31	0-31	0-63	0-99	0-65535	1-31
D1 (Destination 1)	Operand to move to	—	0-31	0-287	0-31	0-31	0-63	0-99	—	1-31

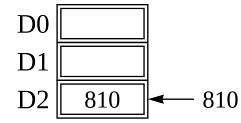
In the high-speed processing mode, operands for advanced instructions are limited. See page 6-1.
When T (timer) or C (counter) is used as S1, the timer/counter current value is read out. When T (timer) or C (counter) is used as D1, the data is written in as a preset value which can be 0 through 9999.

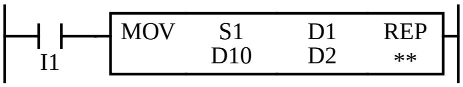
When a bit operand such as input, output, internal relay, or shift register is used as the source or destination, 16 points are used. When repeat is designated for a bit operand, the quantity of operand bits increases in 16-point increments.

Examples: MOV

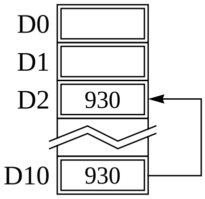


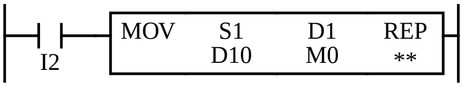
810 → D2
When input I0 is on, constant 810 designated by source operand S1 is moved to data register D2 designated by destination operand D1.





D10 → D2
When input I1 is on, the data in data register D10 designated by source operand S1 is moved to data register D2 designated by destination operand D1.

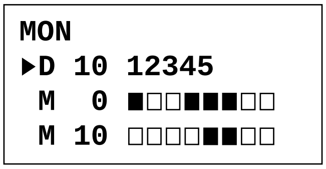




D10 → M0
When input I2 is on, the data in data register D10 designated by source operand S1 is moved to 16 internal relays starting with M0 designated by destination operand D1.

D10 12345 → M0 through M7, M10 through M17

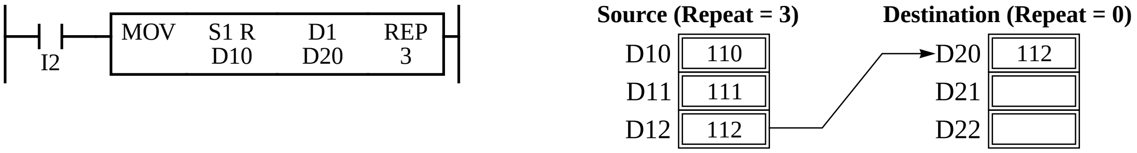
The data in the source data register is converted into 16-bit binary data, and the ON/OFF statuses of the 16 bits are moved to internal relays M0 through M7 and M10 through M17. M0 is the LSB (least significant bit). M17 is the MSB (most significant bit). When D0, M0, and M10 are monitored on the program loader, the data is displayed as shown on the right.



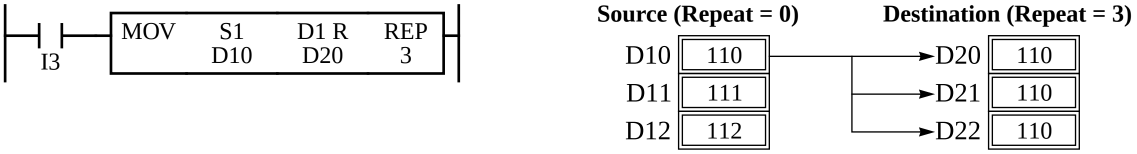
```
MON
▶D 10 12345
 M  0 ■□□■■■□□
 M 10 □□□□■■□□
```

Repeat Operation in the Move Instructions

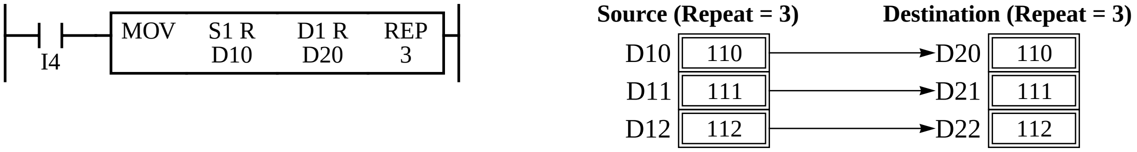
When the S1 (source) is designated with repeat, operands as many as the repeat cycles starting with the operand designated by S1 are moved to the destination. As a result, only the last of the source operands is moved to the destination.



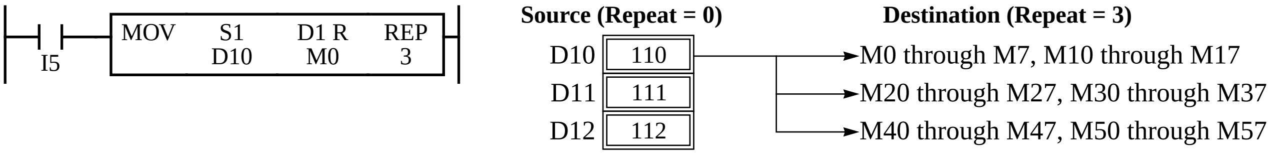
When the D1 (destination) is designated to repeat, the source operand designated by S1 is moved to all destination operands as many as the repeat cycles starting with the destination designated by D1.



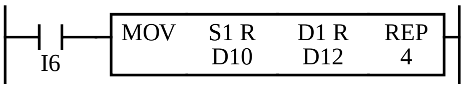
When both S1 (source) and D1 (destination) are designated to repeat, operands as many as the repeat cycles starting with the operand designated by S1 are moved to the same quantity of operands starting with the operand designated by D1.



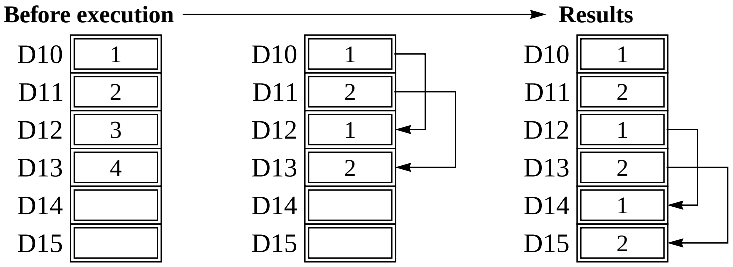
The MOV (move) instruction moves 16-bit word data. When a bit operand such as input, output, internal relay, or shift register is designated as the source or destination operand, 16 bits starting with the one designated by S1 or D1 are the target data. If a repeat operation is designated for a bit operand, the target data increases in 16-bit increments.


I5
MOV S1 D10 D1 R M0 REP 3
Source (Repeat = 0)
Destination (Repeat = 3)
D10 110
D11 111
D12 112
M0 through M7, M10 through M17
M20 through M27, M30 through M37
M40 through M47, M50 through M57

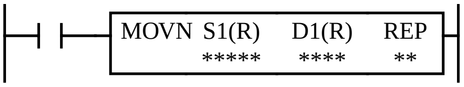
If the repeat operation is designated for both the source and destination and if a portion of the source and destination areas overlap each other, then the source data in the overlapped area is also changed.



Source: D10 through D13 (Repeat = 4)
Destination: D12 through D15 (Repeat = 4)



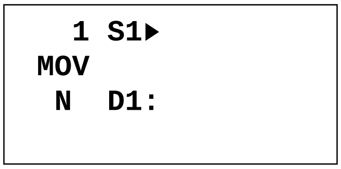
12 MOVN (Move Not)



S1 NOT → D1
When input is on, 16-bit word data from operand designated by S1 is inverted bit by bit and moved to operand designated by D1.

Key Operation

ADV | 1 BPS | 2 BRD



```
  1 S1▶
MOV
 N  D1:
```

Enter operands S1 and D1.

When repeat is required, press the REP key for the operand to repeat, and enter the number of repeat cycles.

To exit, press the ↵ key.

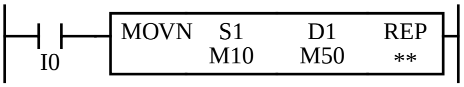
Valid Operands (Standard Processing)

Operand	Function	I	Q	M	T	C	R	D	Constant	Repeat
S1 (Source 1)	Operand to move	0-35	0-31	0-317	0-31	0-31	0-63	0-99	0-65535	1-31
D1 (Destination 1)	Operand to move to	—	0-31	0-287	0-31	0-31	0-63	0-99	—	1-31

In the high-speed processing mode, operands for advanced instructions are limited. See page 6-1.
When T (timer) or C (counter) is used as S1, the timer/counter current value is read out. When T (timer) or C (counter) is used as D1, the data is written in as a preset value which can be 0 through 9999.

When a bit operand such as input, output, internal relay, or shift register is used as the source or destination, 16 points are used. When repeat is designated for a bit operand, the quantity of operand bits increases in 16-point increments.

Examples: MOVN



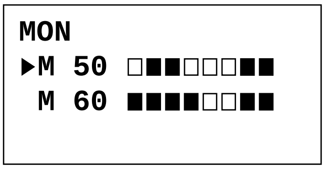
M10 NOT → M50
When input I0 is on, the 16 internal relays starting with M10 designated by source operand S1 are inverted bit by bit and moved to 16 internal relays starting with M50 designated by destination operand D1.

M10 through M17, M20 through M27 NOT ⟶ M50 through M57, M60 through M67

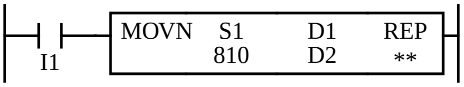
The ON/OFF statuses of the 16 internal relays M10 through M17 and M20 through M27 are inverted and moved to 16 internal relays M50 through M57 and M60 through M67. When M50 and M60 are monitored on the program loader, the data is displayed as shown on the right below. M50 is the LSB (least significant bit), and M67 is the MSB (most significant bit).

	MSB							S1								LSB
Before inversion (M27-M10):	0	0	1	1	0	0	0	0	0	0	1	1	1	0	0	1

	MSB							D1								LSB
After inversion (M67-M50):	1	1	0	0	1	1	1	1	1	1	0	0	0	1	1	0



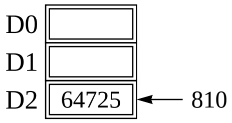
```
MON
▶M 50 □■■□□□■■
 M 60 ■■■■□□■■
```

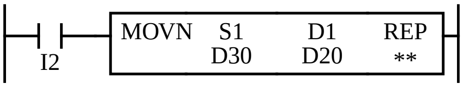


810 NOT → D2
When input I1 is on, decimal constant 810 designated by source operand S1 is converted into 16-bit binary data, and the ON/OFF statuses of the 16 bits are inverted and moved to data register D2 designated by destination operand D1.

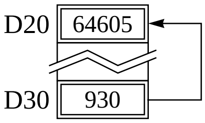
	MSB							S1								LSB
Before inversion (810):	0	0	0	0	0	0	1	1	0	0	1	0	1	0	1	0

	MSB							D1								LSB
After inversion (64725):	1	1	1	1	1	1	0	0	1	1	0	1	0	1	0	1

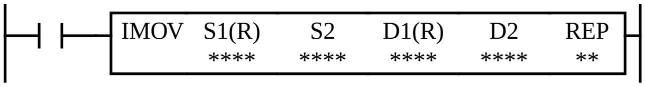




D10 NOT → D2
When input I2 is on, the data in data register D30 designated by S1 is inverted bit by bit and moved to data register D20 designated by D1.



13 IMOV (Indirect Move)



S1 + S2 → D1 + D2
When the input is on, the values contained in operands designated by S1 and S2 are added to determine the source of data. The 16-bit word data so determined is moved to destination, which is determined by the sum of values contained in operands designated by D1 and D2.

Key Operation

ADV | 1 BPS | 3 BPP

```
   1 S1▶
IMOV S2:
     D1:
     D2:
```

Enter operands S1, S2, D1, and D2.

When repeat is required, press the REP key for the operand to repeat, and enter the number of repeat cycles.

To exit, press the ↵ key.

Valid Operands (Standard Processing)

Operand	Function	I	Q	M	T	C	R	D	Constant	Repeat
S1 (Source 1)	Base address to move from	0-35	0-31	0-317	0-31	0-31	0-63	0-99	—	1-31
S2 (Source 2)	Offset for S1	0-35	0-31	0-287	0-31	0-31	0-63	0-99	—	—
D1 (Destination 1)	Base address to move to	—	0-31	0-287	0-31	0-31	0-63	0-99	—	1-31
D2 (Destination 2)	Offset for D1	0-35	0-31	0-287	0-31	0-31	0-63	0-99	—	—

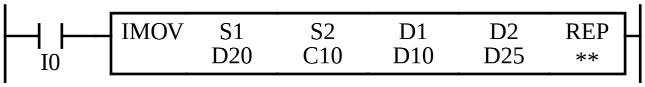
In the high-speed processing mode, operands for advanced instructions are limited. See page 6-1.
When T (timer) or C (counter) is designated as S1, S2, or D2, the operand data is the timer/counter current value. When T (timer) or C (counter) is designated as D1, the operand data is the timer/counter preset value which can be 0 through 9999.

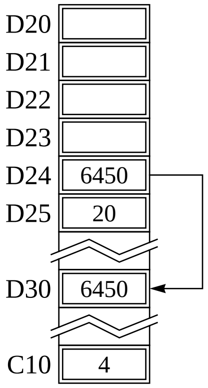
Make sure that the source data determined by S1 + S2 and the destination data determined by D1 + D2 are within the operand range. If the derived source or destination operand is out of the operand range, a user program execution error will result, turning on special internal relay M304 and error indicator ERR1.

When a bit operand such as input, output, internal relay, or shift register is used as the source or destination, 16 points are used. When repeat is designated for a bit operand, the quantity of operand bits increases in 16-point increments.

Source operand S2 and destination operand D2 do not have to be designated. If S2 or D2 is not designated, the source or destination operand is determined by S1 or D1 without offset.

Example: IMOV





D20 + C10 → D10 + D25
Source operand S1 and destination operand D1 determine the type of operand. Source operand S2 and destination operand D2 are the offset values to determine the source and destination operands.

If the current value of counter C10 designated by source operand S2 is 4, the source data is determined by adding the offset to data register D20 designated by source operand S1:

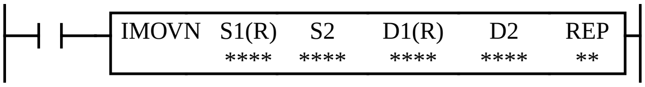
D(20 + 4) = D24

If data register D25 contains a value of 20, the destination is determined by adding the offset to data register D10 designated by destination operand D1:

D(10 + 20) = D30

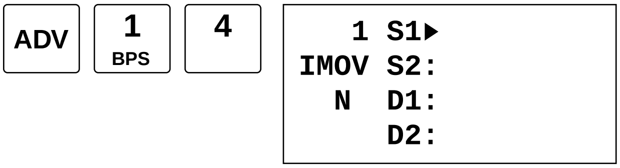
As a result, when input I0 is on, the data in data register D24 is moved to data register D30.

14 IMOVN (Indirect Move Not)



S1 + S2 NOT → D1 + D2
When input is on, the values contained in operands designated by S1 and S2 are added to determine the source of data. The 16-bit word data so determined is inverted and moved to destination, which is determined by the sum of values contained in operands designated by D1 and D2.

Key Operation



```
   1 S1▶
IMOV S2:
   N D1:
     D2:
```

Enter operands S1, S2, D1, and D2.

When repeat is required, press the REP key for the operand to repeat, and enter the number of repeat cycles.

To exit, press the ↵ key.

Valid Operands (Standard Processing)

Operand	Function	I	Q	M	T	C	R	D	Constant	Repeat
S1 (Source 1)	Base address to move from	0-35	0-31	0-317	0-31	0-31	0-63	0-99	—	1-31
S2 (Source 2)	Offset for S1	0-35	0-31	0-287	0-31	0-31	0-63	0-99	—	—
D1 (Destination 1)	Base address to move to	—	0-31	0-287	0-31	0-31	0-63	0-99	—	1-31
D2 (Destination 2)	Offset for D1	0-35	0-31	0-287	0-31	0-31	0-63	0-99	—	—

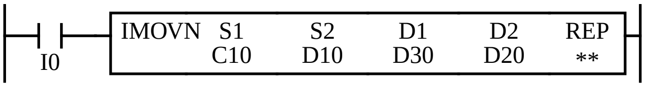
In the high-speed processing mode, operands for advanced instructions are limited. See page 6-1.
When T (timer) or C (counter) is designated as S1, S2, or D2, the operand data is the timer/counter current value. When T (timer) or C (counter) is designated as D1, the operand data is the timer/counter preset value which can be 0 through 9999.

Make sure that the source data determined by S1 + S2 and the destination data determined by D1 + D2 are within the operand range. If the derived source or destination operand is out of the operand range, then a user program execution error will result, turning on special internal relay M304 and error indicator ERR1.

When a bit operand such as input, output, internal relay, or shift register is used as the source or destination, 16 points are used. When repeat is designated for a bit operand, the quantity of operand bits increases in 16-point increments.

Source operand S2 and destination operand D2 do not have to be designated. If S2 or D2 is not designated, the source or destination operand is determined by S1 or D1 without offset.

Example: IMOVN



C10 + D10 NOT → D30 + D20
Source operand S1 and destination operand D1 determine the type of operand. Source operand S2 and destination operand D2 are the offset values to determine the source and destination operands.

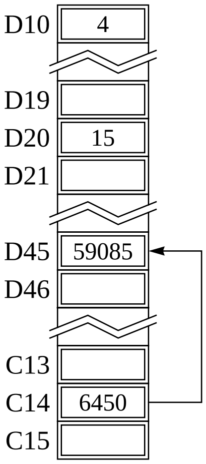
If the data of data register D10 designated by source operand S2 is 4, then the source data is determined by adding the offset to counter C10 designated by source operand S1:

C(10 + 4) = C14

If data register D20 designated by destination operand D2 contains a value of 15, then the destination is determined by adding the offset to data register D30 designated by destination operand D1:

D(30 + 15) = D45

As a result, when input I0 is on, the current value of counter C14 is inverted and moved to data register D45.

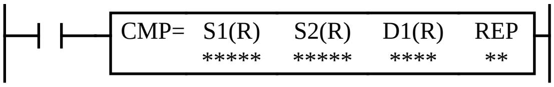


Introduction

Data can be compared using comparison instructions, such as equal to, unequal to, less than, greater than, less than or equal to, and greater than or equal to. When the comparison result is true, an output or internal relay is turned on. The repeat operation can also be used to compare more than one set of data.

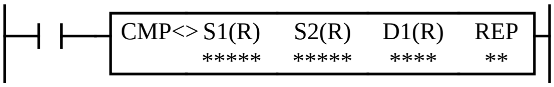
Since the comparison instructions are executed in each scan while input is on, a pulse input from an SOTU or SOTD instruction should be used as required.

21 CMP= (Compare Equal To)



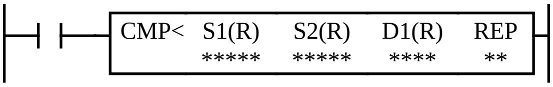
S1 = S2 → D1 on
When input is on, 16-bit word data designated by source operands S1 and S2 are compared. When S1 data is equal to S2 data, destination operand D1 is turned on. When the condition is not met, D1 is turned off.

22 CMP<> (Compare Unequal To)



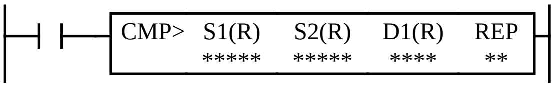
S1 ≠ S2 → D1 on
When input is on, 16-bit word data designated by source operands S1 and S2 are compared. When S1 data is not equal to S2 data, destination operand D1 is turned on. When the condition is not met, D1 is turned off.

23 CMP< (Compare Less Than)



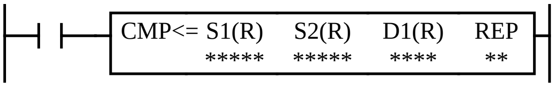
S1 < S2 → D1 on
When input is on, 16-bit word data designated by source operands S1 and S2 are compared. When S1 data is less than S2 data, destination operand D1 is turned on. When the condition is not met, D1 is turned off.

24 CMP> (Compare Greater Than)



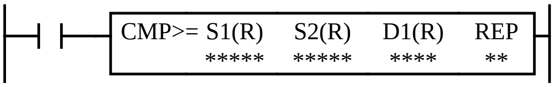
S1 > S2 → D1 on
When input is on, 16-bit word data designated by source operands S1 and S2 are compared. When S1 data is greater than S2 data, destination operand D1 is turned on. When the condition is not met, D1 is turned off.

25 CMP<= (Compare Less Than or Equal To)



S1 ≤ S2 → D1 on
When input is on, 16-bit word data designated by source operands S1 and S2 are compared. When S1 data is less than or equal to S2 data, destination operand D1 is turned on. When the condition is not met, D1 is turned off.

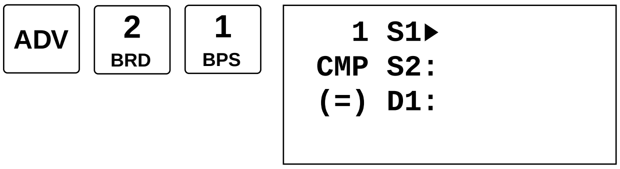
26 CMP>= (Compare Greater Than or Equal To)



S1 ≥ S2 → D1 on
When input is on, 16-bit word data designated by source operands S1 and S2 are compared. When S1 data is greater than or equal to S2 data, destination operand D1 is turned on. When the condition is not met, D1 is turned off.

Key Operation

Press the ADV key, followed by the advanced instruction number.



```
  1 S1▶
CMP S2:
(=) D1:
```

Enter operands S1, S2, and D1.

When repeat is required, press the REP key for the operand to repeat, and enter the number of repeat cycles.

To exit, press the ↵ key.

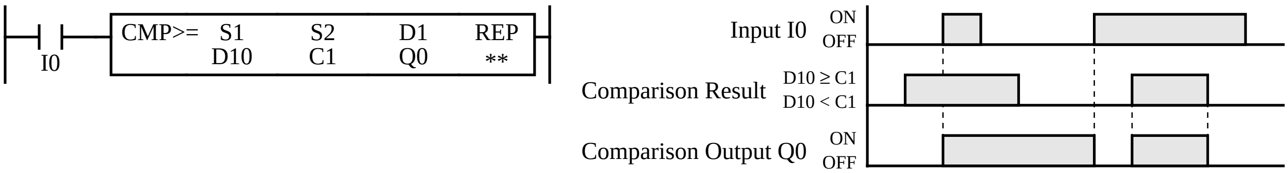
Valid Operands (Standard Processing)

Operand	Function	I	Q	M	T	C	R	D	Constant	Repeat
S1 (Source 1)	Data to compare	0-35	0-31	0-317	0-31	0-31	0-63	0-99	0-65535	1-31
S2 (Source 2)	Data to compare	0-35	0-31	0-317	0-31	0-31	0-63	0-99	0-65535	1-31
D1 (Destination 1)	Comparison output	—	0-31	0-287	—	—	—	—	—	1-31

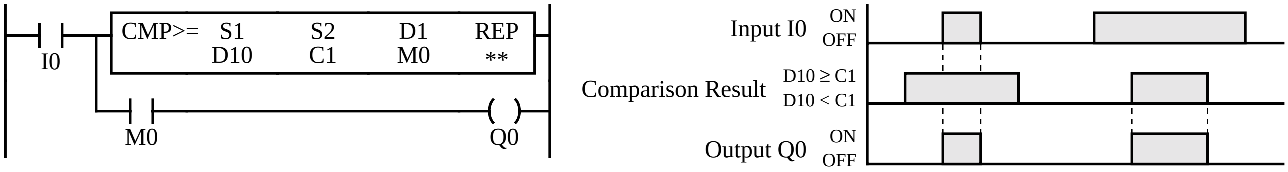
In the high-speed processing mode, operands for advanced instructions are limited. See page 6-1.
When T (timer) or C (counter) is used as S1 or S2, the timer/counter current value is read out. When T (timer) or C (counter) is used as D1, the data is written in as a preset value which can be 0 through 9999.

Examples: CMP>=

The comparison output is usually maintained while the input to the comparison instruction is off. If the comparison output is on, the on status is maintained when the input is turned off as demonstrated by this program.



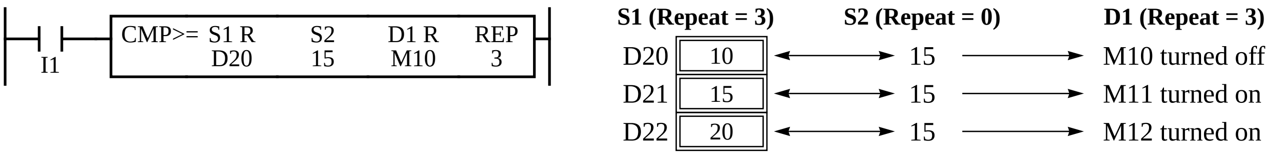
This program turns the output off when the input is off.



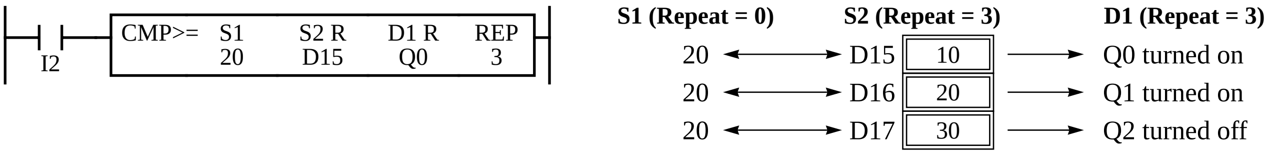
Repeat Operation in the Comparison Instructions

When S1 and/or S2 (source) is designated to repeat, D1 (destination) is normally required to be designated to repeat, otherwise only the result of the comparison in the last repeat cycle is set to one destination.

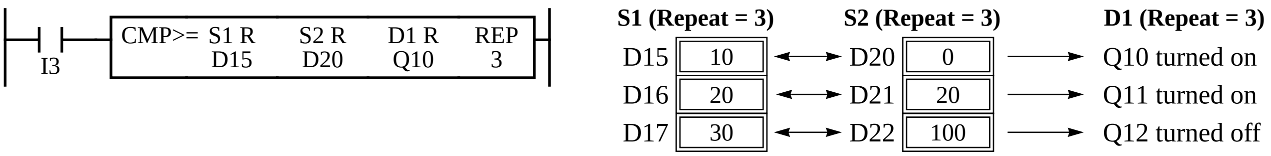
When S1 (source) and D1 (destination) are designated with repeat, operands as many as the repeat cycles starting with the operand designated by S1 are compared with the operand designated by S2. The comparison results are set to operands as many as the repeat cycles starting with the operand designated by D1.



When S2 (source) and D1 (destination) are designated to repeat, the operand designated by S1 is compared with operands as many as the repeat cycles starting with the operand designated by S2. The comparison results are set to operands as many as the repeat cycles starting with the operand designated by D1.



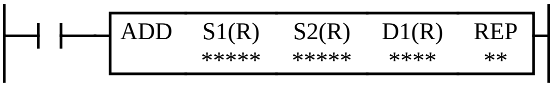
When S1, S2 (source), and D1 (destination) are designated to repeat, operands as many as the repeat cycles starting with the operands designated by S1 and S2 are compared with each other. The comparison results are set to operands as many as the repeat cycles starting with the operand designated by D1.



Introduction

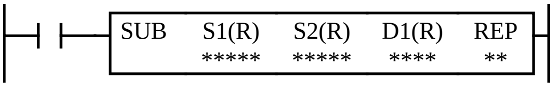
The binary arithmetic instructions make it possible for the user to program computations using addition, subtraction, multiplication, and division. For addition and subtraction operands, internal relay M303 is used to carry or to borrow.

31 ADD (Addition)



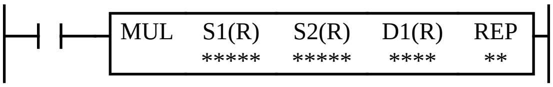
S1 + S2 → D1, CY
When input is on, 16-bit word data designated by source operands S1 and S2 are added. The result is set to destination operand D1 and carry (M303).

32 SUB (Subtraction)



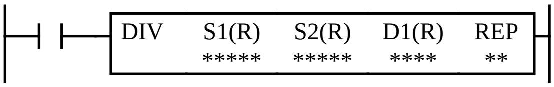
S1 – S2 → D1, BW
When input is on, 16-bit word data designated by source operand S2 is subtracted from 16-bit word data designated by source operand S1. The result is set to destination operand D1 and borrow (M303).

33 MUL (Multiplication)



S1 × S2 → D1
When input is on, 16-bit word data designated by source operand S1 is multiplied by 16-bit word data designated by source operand S2. The result is set to destination operand D1.
When the result exceeds 65535, error indicator ERR1 and special internal relay M304 (user program execution error) are turned on.

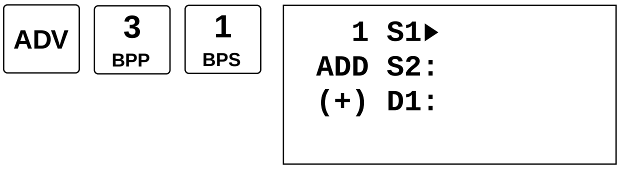
34 DIV (Division)



S1 ÷ S2 → D1 (quotient), D1+1 (remainder)
When input is on, 16-bit word data designated by source operand S1 is divided by 16-bit word data designated by source operand S2. The quotient is set to destination operand D1, and the remainder is set to D1+1.
When S2 is 0 (dividing by 0), error indicator ERR1 and special internal relay M304 (user program execution error) are turned on.

Key Operation

Press the ADV key, followed by the advanced instruction number.



```
  1 S1▶
ADD S2:
(+) D1:
```

Enter operands S1, S2, and D1.

When repeat is required, press the REP key for the operand to repeat, and enter the number of repeat cycles.

To exit, press the ↵ key.

Valid Operands (Standard Processing)

Operand	Function	I	Q	M	T	C	R	D	Constant	Repeat
S1 (Source 1)	Data for calculation	0-35	0-31	0-317	0-31	0-31	0-63	0-99	0-65535	1-31
S2 (Source 2)	Data for calculation	0-35	0-31	0-317	0-31	0-31	0-63	0-99	0-65535	1-31
D1 (Destination 1)	Destination to store results	—	0-31	0-287	0-31	0-31	0-63	0-99	—	1-31

In the high-speed processing mode, operands for advanced instructions are limited. See page 6-1.
When T (timer) or C (counter) is used as S1 or S2, the timer/counter current value is read out. When T (timer) or C (counter) is used as D1, the data is written in as a preset value which can be 0 through 9999.

Note: When using the timer or counter as destination, make sure that the data does not exceed the maximum preset value of 9999. When the preset value exceeds 9999, a user program execution error will result, turning on error indicator ERR1 and special internal relay M304. When a user program execution error occurs, the result is not set to the destination.

Since the binary arithmetic instructions are executed in each scan while input is on, a pulse input from an SOTU or SOTD instruction should be used as required.

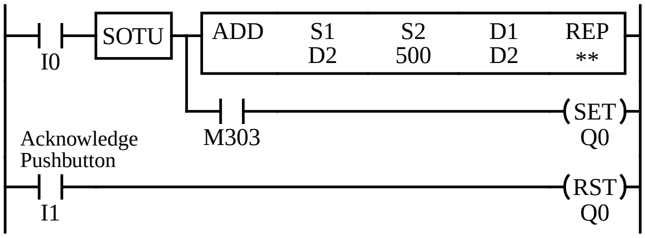
Using Carry or Borrow Signals

When the D1 (destination) data exceeds 65535 as a result of addition, a carry occurs, and special internal relay M303 is turned on. When the D1 (destination) data is less than zero as a result of subtraction, a borrow occurs, and special internal relay M303 is turned on.

There are three ways to program the carrying process (see examples below). If a carry never goes on, the program does not have to include internal relay M303 to process carrying. If a carry goes on unexpectedly, an output can be programmed to be set as a warning indicator. If a carry goes on, the number of times a carry occurs can be added to be used as one word data in a specified register.

Example: ADD

This example demonstrates the use of a carry signal from special internal relay M303 to set an alarm signal.



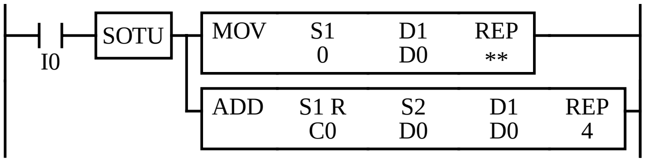
D2 + 500 → D2

When a carry occurs, output Q0 is set as a warning indicator.

When the acknowledge pushbutton (input I1) is pressed, the warning indicator is reset.

Example: Repeat Operation Using ADD

This example uses the repeat operation to total the current values of four counters using the ADD (addition) instruction.



When input I0 is turned on, the MOV (move) instruction sets 0 to data register D0.

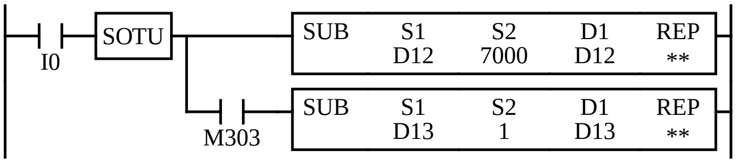
If the current values of counters C0 through C3 are 10, 20, 30, and 40, respectively, then the ADD instruction with 4 repeat cycles totalizes the current values as follows:

C0 (10) + D0 (0) → D0 (10)
C1 (20) + D0 (10) → D0 (30)
C2 (30) + D0 (30) → D0 (60)
C3 (40) + D0 (60) → D0 (100)

Data register D0 stores the final result of 100.

Example: SUB

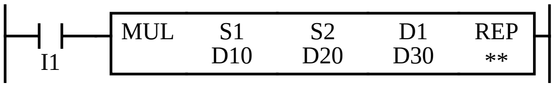
The following example demonstrates the use of special internal relay M303 to process a borrow.


SOTU
I0
SUB S1 D12 S2 7000 D1 D12 REP **
M303
SUB S1 D13 S2 1 D1 D13 REP **

D12 – 7000 → D12

To process borrowing so that the number of times a borrow occurs is subtracted from D13. When a borrow occurs, D13 is decremented by one.

Example: MUL

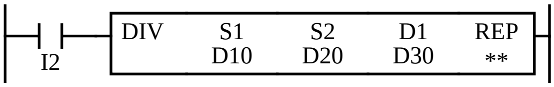


D10 × D20 → D30

When input I1 is on, data of D10 is multiplied by data of D20, and the result is set to D30.

Note: When the result exceeds 65535, a user program execution error will result, turning on error indicator ERR1 and special internal relay M304 (user program execution error). The result is not set to the destination operand.

Example: DIV



D10 ÷ D20 → D30 (quotient), D31 (remainder)

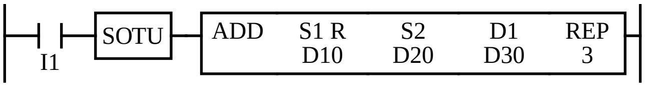
When input I2 is on, data of D10 is divided by data of D20. The quotient is set to D30, and the remainder is set to D31.

Note: Destination uses two word operands, so do not use data register D99 as destination operand D1, otherwise a user program syntax error occurs, and error indicator ERR1 is lit. When using a bit operand such as internal relay for destination, 32 internal relays are required; so do not use internal relay M251 or a larger number as destination operand D1.

Repeat Operation in the ADD, SUB, and MUL Instructions

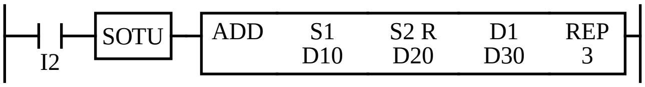
Source operands S1 and S2 and destination operand D1 can be designated to repeat individually or in combination. When destination operand D1 is not designated to repeat, the final result is set to destination operand D1. When repeat is designated, consecutive operands as many as the repeat cycles starting with the designated operand are used. Since the repeat operation works similarly on the ADD (addition), SUB (subtraction), and MUL (multiplication) instructions, the following examples are described using the ADD instruction.

When only S1 (source) is designated to repeat, the final result is set to destination operand D1.



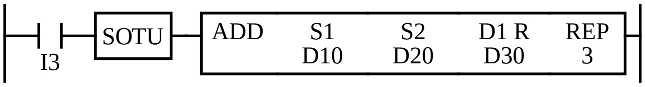
S1 (Repeat = 3)			S2 (Repeat = 0)			D1 (Repeat = 0)	
D10	10	+	D20	25	→	D30	(35)
D11	15	+	D20	25	→	D30	(40)
D12	20	+	D20	25	→	D30	45

When only S2 (source) is designated to repeat, the final result is set to destination operand D1.



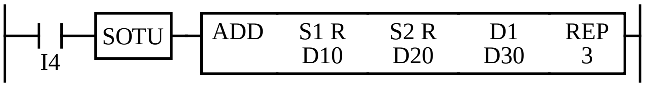
S1 (Repeat = 0)			S2 (Repeat = 3)			D1 (Repeat = 0)	
D10	10	+	D20	25	→	D30	(35)
D10	10	+	D21	35	→	D30	(45)
D10	10	+	D22	45	→	D30	55

When only D1 (destination) is designated to repeat, the same result is set to 3 operands starting with D1.



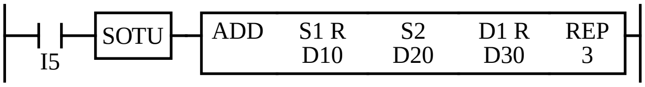
S1 (Repeat = 0)			S2 (Repeat = 0)			D1 (Repeat = 3)	
D10	10	+	D20	25	→	D30	35
D10	10	+	D20	25	→	D31	35
D10	10	+	D20	25	→	D32	35

When S1 and S2 (source) are designated to repeat, the final result is set to destination operand D1.



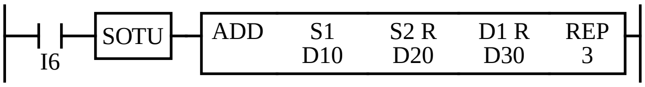
S1 (Repeat = 3)			S2 (Repeat = 3)			D1 (Repeat = 0)	
D10	10	+	D20	25	→	D30	(35)
D11	15	+	D21	35	→	D30	(50)
D12	20	+	D22	45	→	D30	65

When S1 (source) and D1 (dest.) are designated to repeat, different results are set to 3 operands starting with D1.



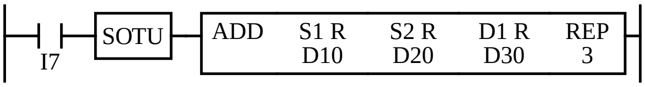
S1 (Repeat = 3)			S2 (Repeat = 0)			D1 (Repeat = 3)	
D10	10	+	D20	25	→	D30	35
D11	15	+	D20	25	→	D31	40
D12	20	+	D20	25	→	D32	45

When S2 (source) and D1 (dest.) are designated to repeat, different results are set to 3 operands starting with D1.



S1 (Repeat = 0)			S2 (Repeat = 3)			D1 (Repeat = 3)	
D10	10	+	D20	25	→	D30	35
D10	10	+	D21	35	→	D31	45
D10	10	+	D22	45	→	D32	55

When all operands are designated to repeat, different results are set to 3 operands starting with D1.



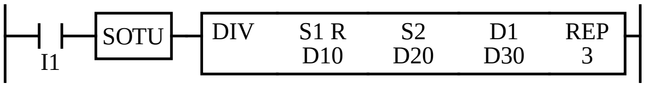
S1 (Repeat = 3)			S2 (Repeat = 3)			D1 (Repeat = 3)	
D10	10	+	D20	25	→	D30	35
D11	15	+	D21	35	→	D31	50
D12	20	+	D22	45	→	D32	65

Note: Special internal relay M308 (carry/borrow) is turned on when a carry or borrow occurs in the last repeat operation. When a user program error occurs in any repeat operation, error indicator ERR1 and special internal relay M304 (user program execution error) are turned on and maintained while operation for other instructions is continued. For the advanced instruction which has caused a user program execution error, results are not set to any destination.

Repeat Operation in the DIV Instruction

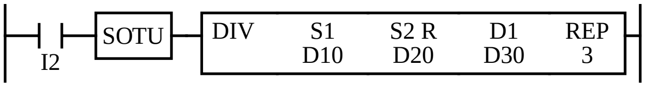
Since the DIV (division) instruction uses two destination operands, the quotient and remainder are stored as described below. Source operands S1 and S2 and destination operand D1 can be designated to repeat individually or in combination. When destination operand D1 is not designated to repeat, the final result is set to destination operand D1 (quotient) and D+1 (remainder). When repeat is designated, consecutive operands as many as the repeat cycles starting with the designated operand are used.

When only S1 (source) is designated to repeat, the final result is set to destination operands D1 and D1+1.



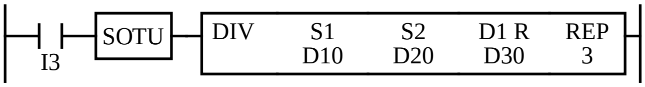
S1 (Repeat = 3)		S2 (Repeat = 0)		D1 (Repeat = 0)	
D10	÷	D20	→	(D30)	(D31)
D11	÷	D20	→	(D30)	(D31)
D12	÷	D20	→	D30	D31
				Quotient	Remainder

When only S2 (source) is designated to repeat, the final result is set to destination operands D1 and D1+1.



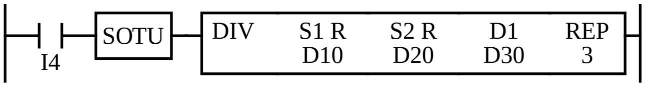
S1 (Repeat = 0)		S2 (Repeat = 3)		D1 (Repeat = 0)	
D10	÷	D20	→	(D30)	(D31)
D10	÷	D21	→	(D30)	(D31)
D10	÷	D22	→	D30	D31
				Quotient	Remainder

When only D1 (destination) is designated to repeat, the same result is set to 6 operands starting with D1.



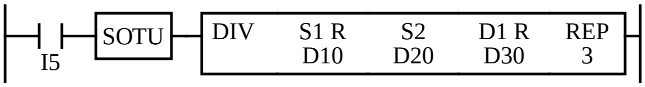
S1 (Repeat = 0)		S2 (Repeat = 0)		D1 (Repeat = 3)	
D10	÷	D20	→	D30	D33
D10	÷	D20	→	D31	D34
D10	÷	D20	→	D32	D35
				Quotient	Remainder

When S1 and S2 (source) are designated to repeat, the final result is set to destination operands D1 and D1+1.



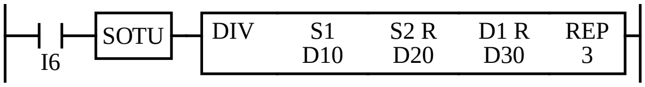
S1 (Repeat = 3)		S2 (Repeat = 3)		D1 (Repeat = 0)	
D10	÷	D20	→	(D30)	(D31)
D11	÷	D21	→	(D30)	(D31)
D12	÷	D22	→	D30	D31
				Quotient	Remainder

When S1 (source) and D1 (dest.) are designated to repeat, different results are set to 6 operands starting with D1.



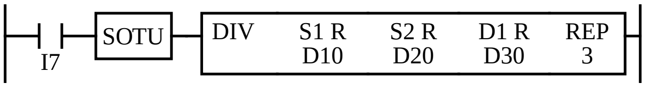
S1 (Repeat = 3)		S2 (Repeat = 0)		D1 (Repeat = 3)	
D10	÷	D20	→	D30	D33
D11	÷	D20	→	D31	D34
D12	÷	D20	→	D32	D35
				Quotient	Remainder

When S2 (source) and D1 (dest.) are designated to repeat, different results are set to 6 operands starting with D1.



S1 (Repeat = 0)		S2 (Repeat = 3)		D1 (Repeat = 3)	
D10	÷	D20	→	D30	D33
D10	÷	D21	→	D31	D34
D10	÷	D22	→	D32	D35
				Quotient	Remainder

When all operands are designated to repeat, different results are set to 6 operands starting with D1.



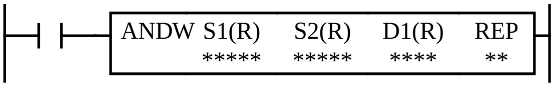
S1 (Repeat = 3)		S2 (Repeat = 3)		D1 (Repeat = 3)	
D10	÷	D20	→	D30	D33
D11	÷	D21	→	D31	D34
D12	÷	D22	→	D32	D35
				Quotient	Remainder

Note: When a user program error occurs in any repeat operation, error indicator ERR1 and special internal relay M304 (user program execution error) are turned on and maintained while operation for other instructions is continued. For the advanced instruction which has caused a user program execution error, results are not set to any destination.

Introduction

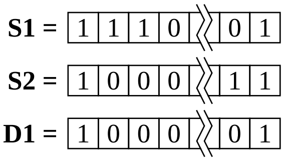
Boolean computations use the AND, OR, and exclusive OR statements as carried out by the ANDW, ORW, and XORW instructions, respectively. Since the Boolean computation is executed in each scan while the input is on, a level input or pulse input should be used as required.

41 ANDW (AND Word)



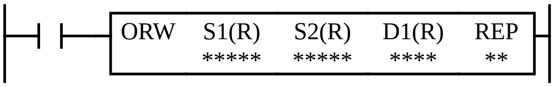
S1 · S2 → D1

When input is on, 16-bit word data designated by source operands S1 and S2 are ANDed, bit by bit. The result is set to destination operand D1.



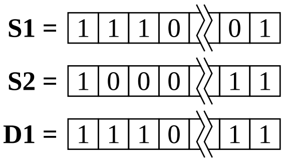
S1	S2	D1
0	0	0
0	1	0
1	0	0
1	1	1

42 ORW (OR Word)



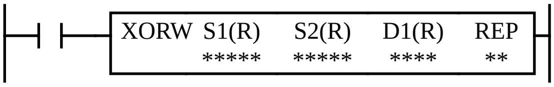
S1 + S2 → D1

When input is on, 16-bit word data designated by source operands S1 and S2 are ORed, bit by bit. The result is set to destination operand D1.



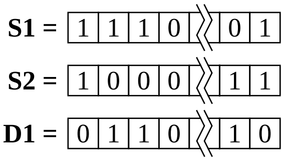
S1	S2	D1
0	0	0
0	1	1
1	0	1
1	1	1

43 XORW (Exclusive OR Word)



S1 ⊕ S2 → D1

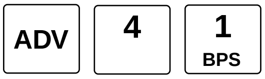
When input is on, 16-bit word data designated by source operands S1 and S2 are exclusive ORed, bit by bit. The result is set to destination operand D1.

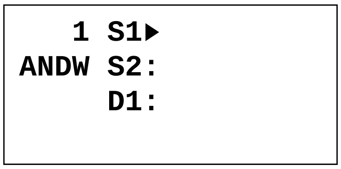


S1	S2	D1
0	0	0
0	1	1
1	0	1
1	1	0

Key Operation

Press the ADV key, followed by the advanced instruction number.





```
   1 S1▶
ANDW S2:
     D1:
```

Enter operands S1, S2, and D1.

When repeat is required, press the REP key for the operand to repeat, and enter the number of repeat cycles.

To exit, press the ↵ key.

Valid Operands (Standard Processing)

Operand	Function	I	Q	M	T	C	R	D	Constant	Repeat
S1 (Source 1)	Data for computation	0-35	0-31	0-317	0-31	0-31	0-63	0-99	0-65535	1-31
S2 (Source 2)	Data for computation	0-35	0-31	0-317	0-31	0-31	0-63	0-99	0-65535	1-31
D1 (Destination 1)	Destination to store results	—	0-31	0-287	0-31	0-31	0-63	0-99	—	1-31

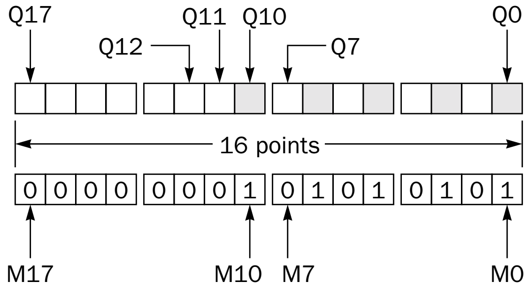
In the high-speed processing mode, operands for advanced instructions are limited. See page 6-1.
When T (timer) or C (counter) is used as S1 or S2, the timer/counter current value is read out. When T (timer) or C (counter) is used as D1, the data is written in as a preset value which can be 0 through 9999.

Note: When using the timer or counter as destination, make sure that the data does not exceed the maximum preset value of 9999. When the preset value exceeds 9999, a user program execution error will result, turning on error indicator ERR1 and special internal relay M304. When a user program execution error occurs, the result is not set to the destination.

Since these Boolean instructions are executed in each scan while input is on, a pulse input from an SOTU or SOTD instruction should be used as required.

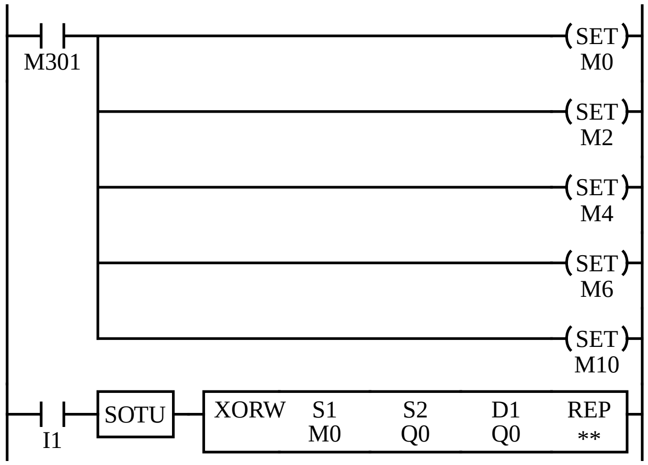
Example: XORW

To convert optional output status among a series of 10 output points, use the XORW instruction in combination with 10 internal relay points.



This program will invert the status of the shaded outputs at the left from on to off, and those not shaded from off to on.

This example uses the 24-I/O type MICRO^3 base unit, which has 10 output terminals Q0 through Q7, Q10, and Q11.



Ten points of outputs Q0 through Q11 are assigned to 10 points of internal relays M0 through M11.

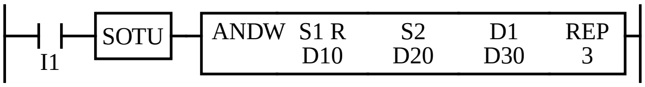
Five points of M0, M2, M4, M6, and M10 are set by initialize pulse special internal relay M301.

When input I1 is turned on, the XORW instruction is executed to invert the status of outputs Q0, Q2, Q4, Q6, and Q10.

Repeat Operation in the ANDW, ORW, and XORW Instructions

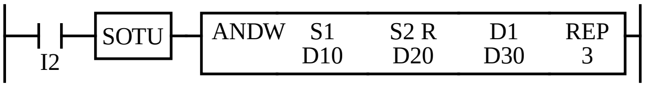
Source operands S1 and S2 and destination operand D1 can be designated to repeat individually or in combination. When destination operand D1 is not designated to repeat, the final result is set to destination operand D1. When repeat is designated, consecutive operands as many as the repeat cycles starting with the designated operand are used. Since the repeat operation works similarly on the ANDW (AND word), ORW (OR word), and XORW (exclusive OR word) instructions, the following examples are described using the ANDW instruction.

When only S1 (source) is designated to repeat, the final result is set to destination operand D1.



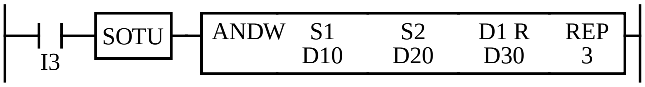
S1 (Repeat = 3)		S2 (Repeat = 0)		D1 (Repeat = 0)
D10	·	D20	→	(D30)
D11	·	D20	→	(D30)
D12	·	D20	→	D30

When only S2 (source) is designated to repeat, the final result is set to destination operand D1.



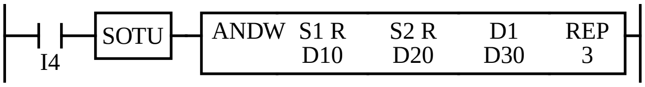
S1 (Repeat = 0)		S2 (Repeat = 3)		D1 (Repeat = 0)
D10	·	D20	→	(D30)
D10	·	D21	→	(D30)
D10	·	D22	→	D30

When only D1 (destination) is designated to repeat, the same result is set to 3 operands starting with D1.



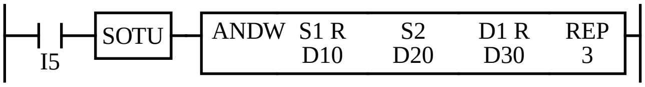
S1 (Repeat = 0)		S2 (Repeat = 0)		D1 (Repeat = 3)
D10	·	D20	→	D30
D10	·	D20	→	D31
D10	·	D20	→	D32

When S1 and S2 (source) are designated to repeat, the final result is set to destination operand D1.



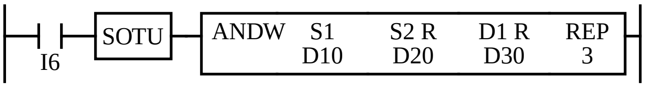
S1 (Repeat = 3)		S2 (Repeat = 3)		D1 (Repeat = 0)
D10	·	D20	→	(D30)
D11	·	D21	→	(D30)
D12	·	D22	→	D30

When S1 (source) and D1 (dest.) are designated to repeat, different results are set to 3 operands starting with D1.



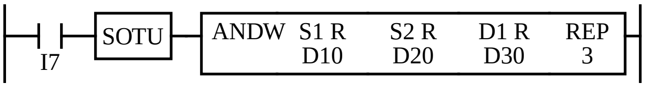
S1 (Repeat = 3)		S2 (Repeat = 0)		D1 (Repeat = 3)
D10	·	D20	→	D30
D11	·	D20	→	D31
D12	·	D20	→	D32

When S2 (source) and D1 (dest.) are designated to repeat, different results are set to 3 operands starting with D1.



S1 (Repeat = 0)		S2 (Repeat = 3)		D1 (Repeat = 3)
D10	·	D20	→	D30
D10	·	D21	→	D31
D10	·	D22	→	D32

When all operands are designated to repeat, different results are set to 3 operands starting with D1.



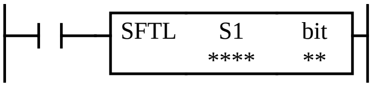
S1 (Repeat = 3)		S2 (Repeat = 3)		D1 (Repeat = 3)
D10	·	D20	→	D30
D11	·	D21	→	D31
D12	·	D22	→	D32

Note: When a user program error occurs in any repeat operation, error indicator ERR1 and special internal relay M304 (user program execution error) are turned on and maintained while operation for other instructions is continued. For the advanced instruction which has caused a user program execution error, results are not set to any destination.

Introduction

Bit shift and rotate instructions are used to shift the 16-bit data in the data register designated by source operand S1 to the left or right by the quantity of bits designated. The result is set to the data register designated by source operand S1 and a carry (special internal relay M303). Since the bit shift and rotate instructions are executed in each scan while the input is on, a level input or pulse input should be used as required.

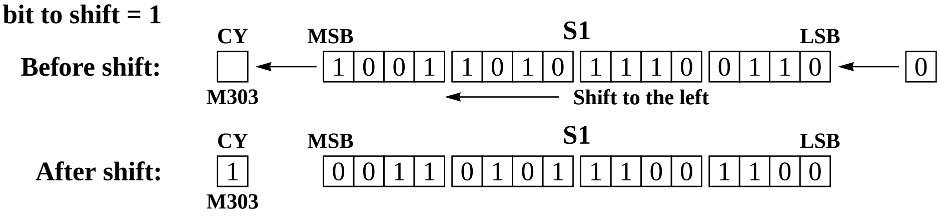
51 SFTL (Shift Left)



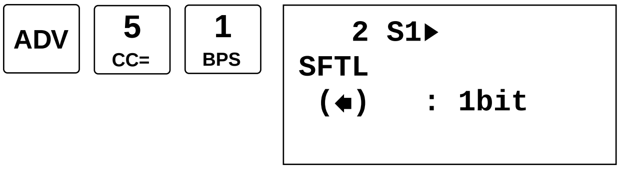
CY ← S1

When input is on, 16-bit data of the data register designated by source operand S1 is shifted to the left by the quantity of bits designated by operand bit. The result is set to the data register, and the last bit status shifted out is set to a carry (special internal relay M303). Zeros are set to the LSB.

When bit to shift = 1



Key Operation



```
   2 S1▶
SFTL
 (◆)   : 1bit
```

Enter operand S1 and the quantity of bits to shift.

Repeat cannot be designated for bit shift instructions.

To exit, press the ↵ key.

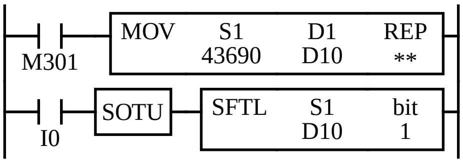
Valid Operands (Standard Processing)

Operand	Function	I	Q	M	T	C	R	D	Constant	Repeat
S1 (Source 1)	Data for bit shift	—	—	—	—	—	—	0-99	—	—
bit	Quantity of bits to shift	—	—	—	—	—	—	—	1-15	—

In the high-speed processing mode, data registers for this instruction are limited to D0 through D31.

Since the bit shift instructions are executed in each scan while input is on, a pulse input from an SOTU or SOTD instruction should be used as required.

Example: SFTL

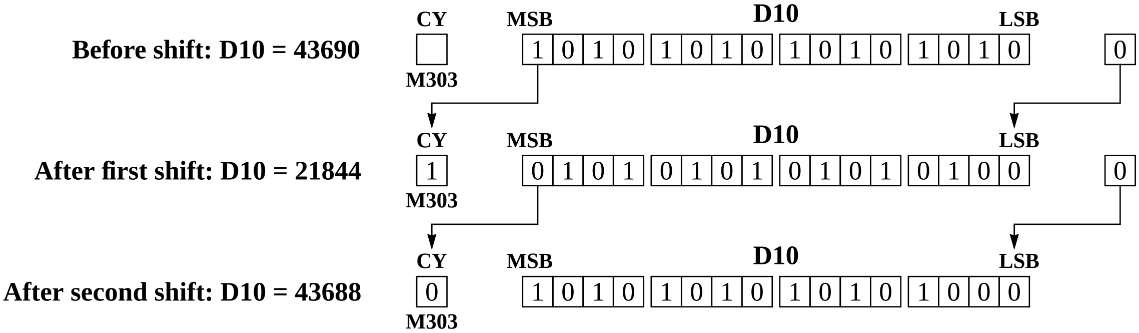


M301 is the initialize pulse special internal relay.

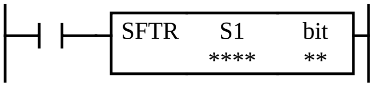
When MICRO^3 starts operation, the MOV (move) instruction sets 43690 to data register D10.

Each time input I0 is turned on, 16-bit data of data register D10 is shifted to the left by 1 bit as designated by operand bit. The bit status shifted out is set to a carry (special internal relay M303). Zeros are set to the LSB.

Bits to shift = 1



52 SFTR (Shift Right)

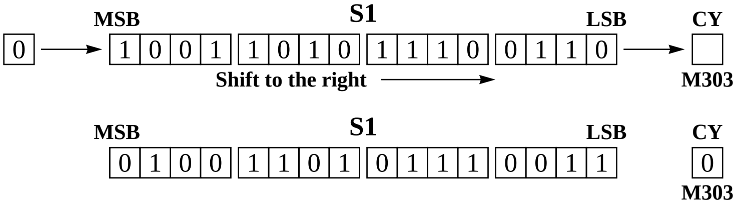


S1 → CY
When input is on, 16-bit data of the data register designated by source operand S1 is shifted to the right by the quantity of bits designated by operand bit. The result is set to the data register, and the last bit status shifted out is set to a carry (special internal relay M303). Zeros are set to the MSB.

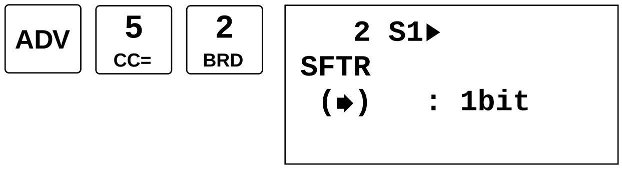
When bits to shift = 1

Before shift:

After shift:



Key Operation



Enter operand S1 and the quantity of bits to shift.

Repeat cannot be designated for bit shift instructions.

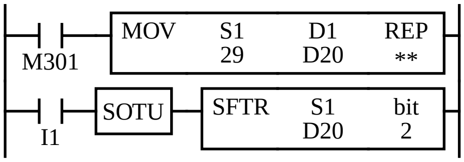
To exit, press the ↵ key.

Valid Operands (Standard Processing)

Operand	Function	I	Q	M	T	C	R	D	Constant	Repeat
S1 (Source 1)	Data for bit shift	—	—	—	—	—	—	0-99	—	—
bit	Quantity of bits to shift	—	—	—	—	—	—	—	1-15	—

In the high-speed processing mode, data registers for this instruction are limited to D0 through D31.
Since the bit shift instructions are executed in each scan while input is on, a pulse input from an SOTU or SOTD instruction should be used as required.

Example: SFTR



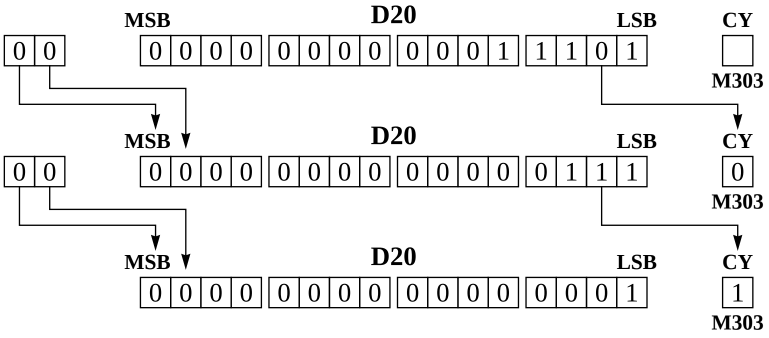
M301 is the initialize pulse special internal relay.
When MICRO^3 starts operation, the MOV (move) instruction sets 29 to data register D20.
Each time input I1 is turned on, 16-bit data of data register D20 is shifted to the right by 2 bits as designated by operand bit. The last bit status shifted out is set to a carry (special internal relay M303). Zeros are set to the MSB's.

Bits to shift = 2

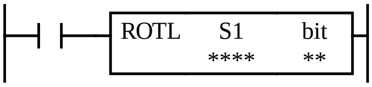
Before shift: D20 = 29

After first shift: D20 = 7

After second shift: D20 = 1



53 ROTL (Rotate Left)

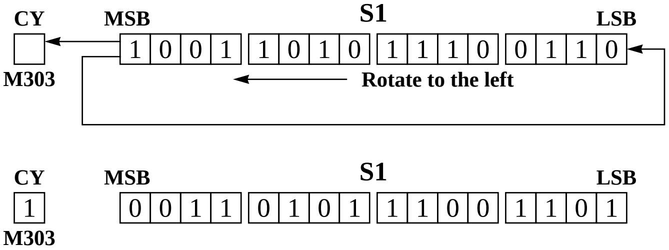


When input is on, 16-bit data of the data register designated by S1 is rotated to the left by the quantity of bits designated by operand bit. The last bit status rotated out of the data register is set to a carry (special internal relay M303).

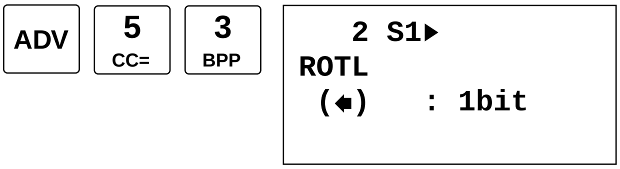
When bits to rotate = 1

Before rotation:

After rotation:



Key Operation



```
   2 S1▶
ROTL
 (◆)   : 1bit
```

Enter operand S1 and the quantity of bits to rotate.

Repeat cannot be designated for bit rotate instructions.

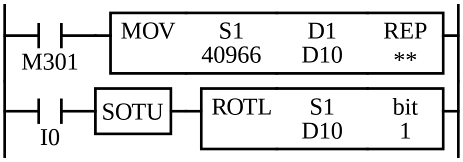
To exit, press the ↵ key.

Valid Operands (Standard Processing)

Operand	Function	I	Q	M	T	C	R	D	Constant	Repeat
S1 (Source 1)	Data for bit rotation	—	—	—	—	—	—	0-99	—	—
bit	Quantity of bits to rotate	—	—	—	—	—	—	—	1-15	—

In the high-speed processing mode, data registers for this instruction are limited to D0 through D31.
Since the bit rotate instructions are executed in each scan while input is on, a pulse input from an SOTU or SOTD instruction should be used as required.

Example: ROTL



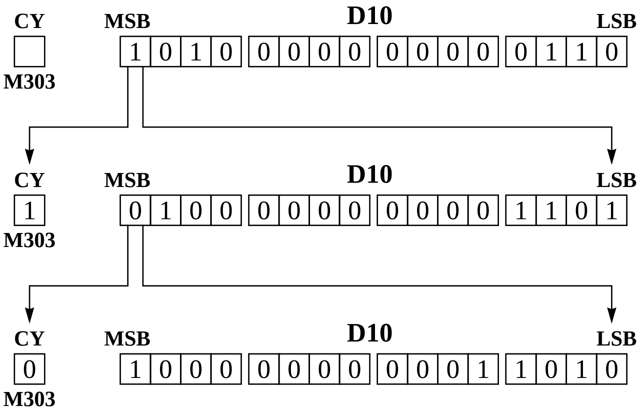
M301 is the initialize pulse special internal relay.
When MICRO^3 starts operation, the MOV (move) instruction sets 40966 to data register D10.
Each time input I0 is turned on, 16-bit data of data register D10 is rotated to the left by 1 bit as designated by operand bit. The status of the MSB is set to a carry (special internal relay M303).

Bits to rotate = 1

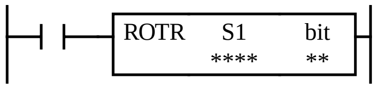
Before rotation: D10 = 40966

After first rotation: D10 = 16397

After second rotation: D10 = 32794



54 ROTR (Rotate Right)

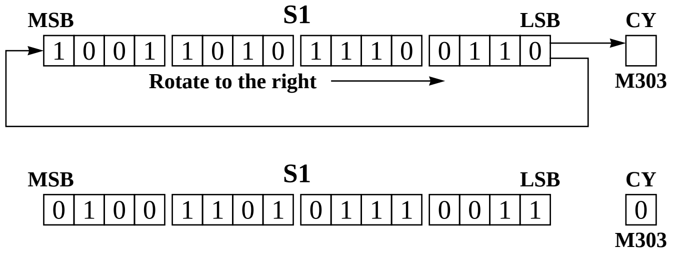


When input is on, 16-bit data of the data register designated by S1 is rotated to the right by the quantity of bits designated by operand bit. The last bit status rotated out of the data register is set to a carry (special internal relay M303).

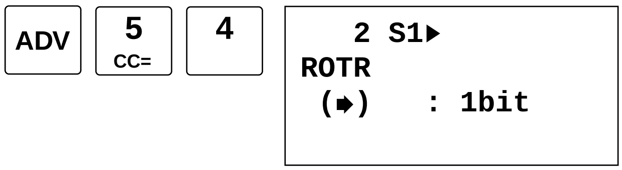
When bits to rotate = 1

Before rotation:

After rotation:



Key Operation



```
   2 S1▶
ROTR
 (➡)   : 1bit
```

Enter operand S1 and the quantity of bits to rotate.

Repeat cannot be designated for bit rotate instructions.

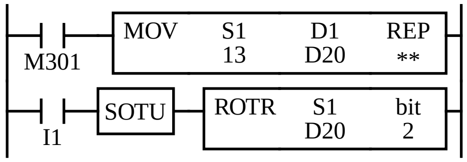
To exit, press the ↵ key.

Valid Operands (Standard Processing)

Operand	Function	I	Q	M	T	C	R	D	Constant	Repeat
S1 (Source 1)	Data for bit rotation	—	—	—	—	—	—	0-99	—	—
bit	Quantity of bits to rotate	—	—	—	—	—	—	—	1-15	—

In the high-speed processing mode, data registers for this instruction are limited to D0 through D31.
Since the bit rotate instructions are executed in each scan while input is on, a pulse input from an SOTU or SOTD instruction should be used as required.

Example: ROTR



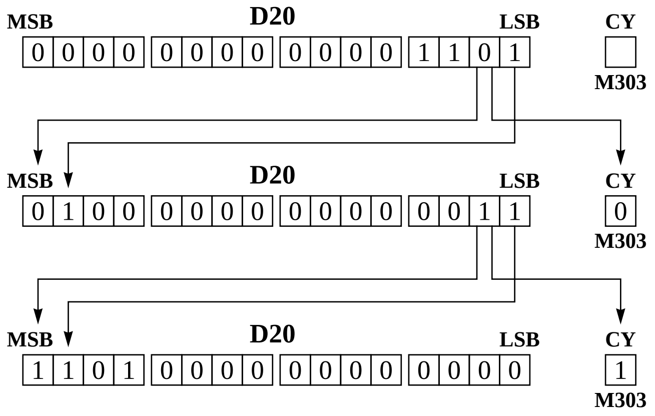
M301 is the initialize pulse special internal relay.
When MICRO^3 starts operation, the MOV (move) instruction sets 13 to data register D20.
Each time input I1 is turned on, 16-bit data of data register D20 is rotated to the right by 1 bit as designated by operand bit. The last bit status rotated out of the data register is set to a carry (special internal relay M303).

Bits to rotate = 2

Before rotation: D20 = 13

After first rotation: D20 = 16387

After second rotation: D20 = 53248

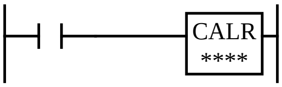


Introduction

The 16- and 24-I/O type MICRO^3 base units feature five real-time calendar and clock instructions used for programming the calendar and clock; CALR (calendar read), CALW (calendar write), CLKR (clock read), CLKW (clock write), and ADJ (adjust). These instructions cannot be used on the 10-I/O type MICRO^3 base unit. After initial setting of calendar and clock using FUN28, date and time are maintained. For FUN28, see page 5-12. If control data registers D95 through D98 are enabled using FUN10, day of week, hour, minute, and second data can be read out to these data registers when the MICRO^3 is running or stopped. For FUN10, see page 5-8.

Note: Each clock/calendar instruction can be used only once in a user program.

71 CALR (Calendar Read)

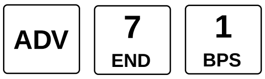


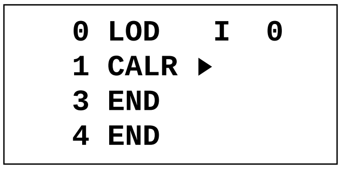
When input is on, calendar data (year, month, day, and day of week) is read to four data registers starting with the designated operand.

D = Year (0 to 99)
D+1 = Month (1 to 12)
D+2 = Day (1 to 31)
D+3 = Day of week (0 to 6) assigned as follows:

0	1	2	3	4	5	6
Sunday	**Monday**	**Tuesday**	**Wednesday**	**Thursday**	**Friday**	**Saturday**

Key Operation





```
0 LOD   I  0
1 CALR ▸
3 END
4 END
```

Enter operand for the first data register to read calendar data.

Four consecutive data registers are required to read data.

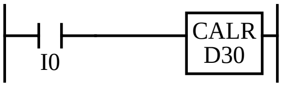
To enter the instruction, press the ↵ key.

Valid Operands (Standard Processing)

Operand Function	I	Q	M	T	C	R	D	Constant	Repeat
Destination to read calendar data	—	—	—	—	—	—	0-99	—	—

In the high-speed processing mode, data registers for this instruction are limited to D0 through D31.
Since the CALR instruction is executed in each scan while input is on, a pulse input from an SOTU or SOTD instruction should be used as required.

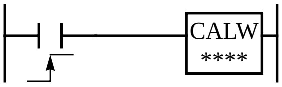
Example: CALR



When input I0 is on, calendar data is read to data registers D30 through D33.

D30 = Year
D31 = Month
D32 = Day
D33 = Day of week

72 CALW (Calendar Write)



When input is turned on, the calendar is set using data stored in four data registers starting with the designated operand.

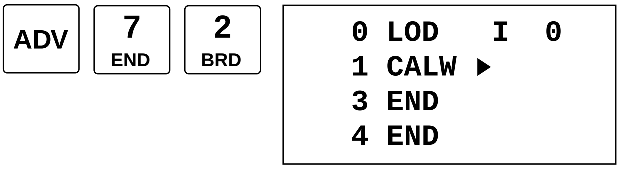
D = Year (0 to 99)
D+1 = Month (1 to 12)
D+2 = Day (1 to 31)
D+3 = Day of week (0 to 6) assigned as follows:

0	1	2	3	4	5	6
Sunday	**Monday**	**Tuesday**	**Wednesday**	**Thursday**	**Friday**	**Saturday**

Note: Only the months of January, March, May, July, August, October, and December can be programmed with a date of the 31st. The month of February can be programmed with a date of the 29th only for actual leap years. (Leap years are automatically adjusted for).

Note: If month, day, or day of week is assigned a value which is not within the range specified above or if a date is not assigned according to the preceding note, then invalid data will result in a user program execution error, internal relay M304 turns on, and the ERR1 indicator on the MICRO^3 base unit also turns on. The error code is stored in data register D93 when the control data register is enabled using FUN10. See page 5-8.

Key Operation



```
0 LOD   I  0
1 CALW ▶
3 END
4 END
```

Enter operand for the first data register to write calendar data.

To enter the instruction, press the ↵ key.

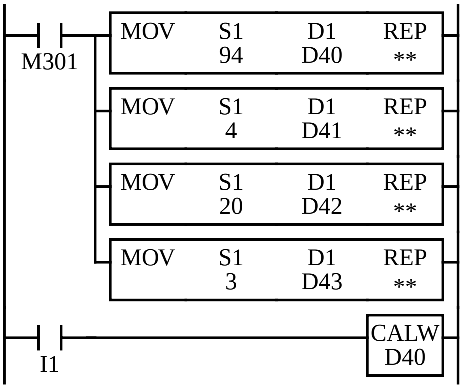
Store calendar data to four consecutive data registers starting with the designated operand.

Valid Operands (Standard Processing)

Operand Function	I	Q	M	T	C	R	D	Constant	Repeat
Source to write the calendar data	—	—	—	—	—	—	0-99	—	—

In the high-speed processing mode, data registers for this instruction are limited to D0 through D31.
Since the CALW instruction is executed at the rising edge of the input, a pulse input from an SOTU or SOTD instruction is not required.

Example: CALW



M301 is the initialize pulse special internal relay.

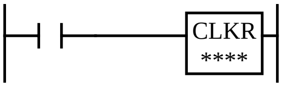
When MICRO^3 starts operation, the MOV instructions set calendar data to data registers D40 through D43.

D40 = 94 (Year 1994)
D41 = 4 (April)
D42 = 20 (Day)
D43 = 3 (Wednesday)

When input I1 is turned on, the calendar is set using data from data registers D40 through D43.

Note: Calendar data can also be set using FUN28. See page 5-12.

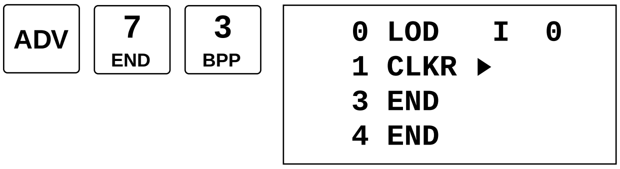
73 CLKR (Clock Read)



When input is on, clock data (hour, minute, and second) is read to three data registers starting with the designated operand.

D = Hour (0 to 23)
D+1 = Minute (0 to 59)
D+2 = Second (0 to 59)

Key Operation



```
0 LOD   I  0
1 CLKR ▶
3 END
4 END
```

Enter operand for the first data register to read clock data.

Three consecutive data registers are required to read the clock data.

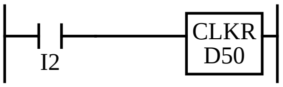
To enter the instruction, press the ↵ key.

Valid Operands (Standard Processing)

Operand Function	I	Q	M	T	C	R	D	Constant	Repeat
Destination to read the clock data	—	—	—	—	—	—	0-99	—	—

In the high-speed processing mode, data registers for this instruction are limited to D0 through D31.
Since the CLKR instruction is executed in each scan while input is on, a pulse input from an SOTU or SOTD instruction should be used as required.

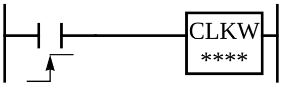
Example: CLKR



When input I2 is on, clock data is read to data registers D50 through D52.

D50 = Hour (0 to 23)
D51 = Minute (0 to 59)
D52 = Second (0 to 59)

74 CLKW (Clock Write)

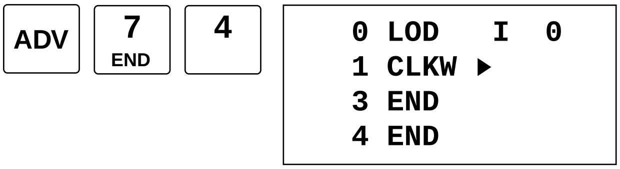


When input is turned on, the clock is set using data stored in three data registers starting with the designated operand.

D = Hour (0 to 23)
D+1 = Minute (0 to 59)
D+2 = Second (0 to 59)

Note: If time is assigned a value which is not within the range specified above, invalid data will result in a user program execution error, internal relay M304 turns on, and the ERR1 indicator on the MICRO3 base unit also turns on. The error code is stored in data register D93 when the control data register is enabled using FUN10. See page 5-8.

Key Operation



```
0 LOD   I  0
1 CLKW ▶
3 END
4 END
```

Enter operand for the first data register to write clock data.

To enter the instruction, press the ↵ key.

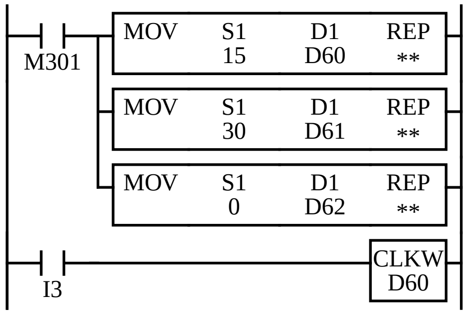
Store clock data to three consecutive data registers starting with the designated operand.

Valid Operands (Standard Processing)

Operand Function	I	Q	M	T	C	R	D	Constant	Repeat
Destination to write the clock data	—	—	—	—	—	—	0-99	—	—

In the high-speed processing mode, data registers for this instruction are limited to D0 through D31.
Since the CLKW instruction is executed at the rising edge of the input, a pulse input from an SOTU or SOTD instruction is not required.

Example: CLKW



M301 is the initialize pulse special internal relay.

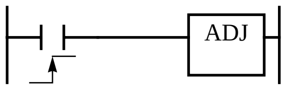
When MICRO^3 starts operation, the MOV instructions set clock data to data registers D60 through D62.

D60 = 15 (Hour)
D61 = 30 (Minute)
D62 = 0 (Second)

When input I3 is turned on, the clock is set using data from data registers D60 through D62.

Note: Clock data can also be set using FUN28. See page 5-12.

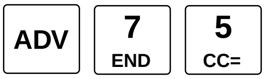
75 ADJ (Adjust)

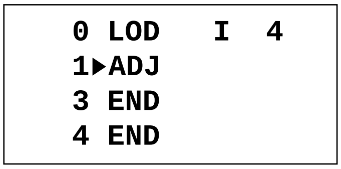


When input is turned on, the clock is adjusted with respect to seconds. If *seconds* are between 0 and 29 for current time, adjustment for *seconds* will be set to 0 and minutes remain the same. If *seconds* are between 30 and 59 for current time, adjustment for *seconds* will be set to 0 and *minutes* are incremented one. The ADJ instruction is useful for precise timing which starts at zero seconds.

Since the ADJ instruction is executed at the rising edge of the input, a pulse input from an SOTU or SOTD instruction is not required.

Key Operation



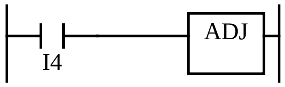


```
0 LOD   I  4
1▸ADJ
3 END
4 END
```

No operand is required for the ADJ instruction.

To enter the instruction, press the ↵ key.

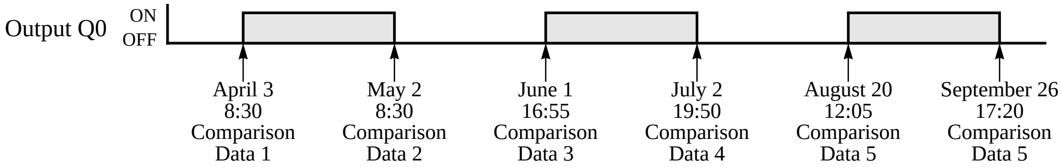
Example: ADJ



When input I4 is turned on, the clock is adjusted with respect to seconds.

Example: Time Scheduled Control

This example demonstrates a program to turn output Q0 on and off according to the chart below:



Internal relays are allocated as shown below.

Comparison Data	Month	Day	Day of Week	Hour	Minute
Comparison Data 1	M10	M11	M12	M13	M14
Comparison Data 2	M20	M21	M22	M23	M24
Comparison Data 3	M30	M31	M32	M33	M34
Comparison Data 4	M40	M41	M42	M43	M44
Comparison Data 5	M50	M51	M52	M53	M54
Comparison Data 6	M60	M61	M62	M63	M64

Data registers are allocated as shown below. Comparison data must be set to data registers D30 through D59 in advance using the program loader. For example, to enter decimal value “4” to data register D30, press the keys:

MON | E/OR D | 3 BPP | 0 | LOD 10 | 4 | ↵

Calendar/Clock Readout Data

Year D20	
Month D21	
Day D22	
Day of Week D23	
Hour D24	
Minute D25	
Second D26	

This program compares the data of month, day, hour, and minute and does not compare the data of day of week and second.

Comparison Data 1 (Turn output Q0 on)

Month D30	4	→ M10
Day D31	3	→ M11
Day of Week D32		→ M12
Hour D33	8	→ M13
Minute D34	30	→ M14

Comparison Data 2 (Turn output Q0 off)

Month D35	5	→ M20
Day D36	2	→ M21
Day of Week D37		→ M22
Hour D38	8	→ M23
Minute D39	30	→ M24

Comparison Data 3 (Turn output Q0 on)

Month D40	6	→ M30
Day D41	1	→ M31
Day of Week D42		→ M32
Hour D43	16	→ M33
Minute D44	55	→ M34

Comparison Data 4 (Turn output Q0 off)

Month D45	7	→ M40
Day D46	2	→ M41
Day of Week D47		→ M42
Hour D48	19	→ M43
Minute D49	50	→ M44

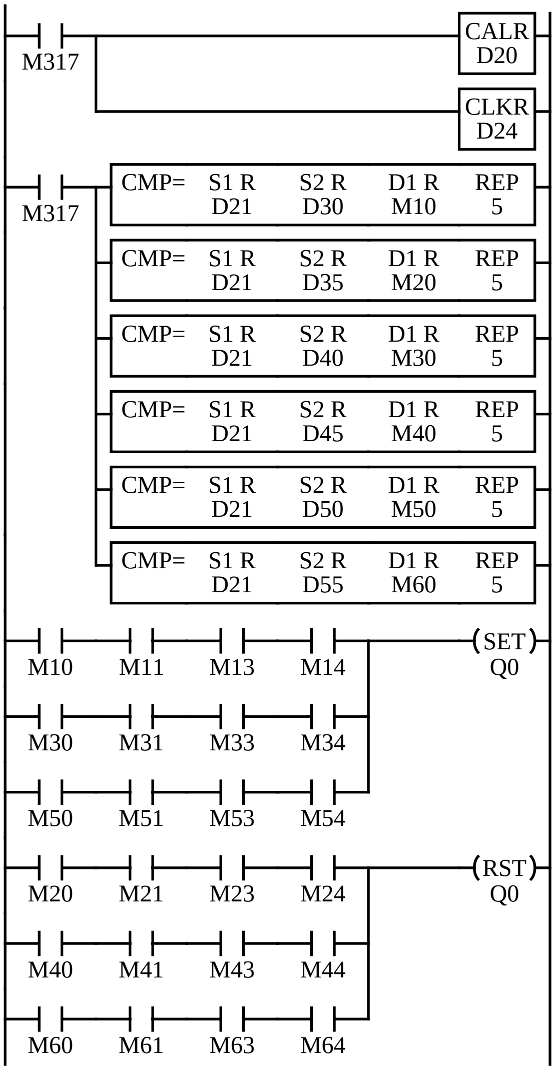
Comparison Data 5 (Turn output Q0 on)

Month D50	8	→ M50
Day D51	20	→ M51
Day of Week D52		→ M52
Hour D53	12	→ M53
Minute D54	5	→ M54

Comparison Data 6 (Turn output Q0 off)

Month D55	9	→ M60
Day D56	26	→ M61
Day of Week D57		→ M62
Hour D58	17	→ M63
Minute D59	20	→ M64

Example: Time Scheduled Control, continued


M317
CALR D20
CLKR D24
M317
CMP= S1 R D21 S2 R D30 D1 R M10 REP 5
CMP= S1 R D21 S2 R D35 D1 R M20 REP 5
CMP= S1 R D21 S2 R D40 D1 R M30 REP 5
CMP= S1 R D21 S2 R D45 D1 R M40 REP 5
CMP= S1 R D21 S2 R D50 D1 R M50 REP 5
CMP= S1 R D21 S2 R D55 D1 R M60 REP 5
M10 M11 M13 M14
SET Q0
M30 M31 M33 M34
M50 M51 M53 M54
M20 M21 M23 M24
RST Q0
M40 M41 M43 M44
M60 M61 M63 M64

M317 is the in-operation special internal relay which remains on during operation.

While the program is executed, the CALR (calendar read) and CLKR (clock read) instructions read the calendar and clock data to data registers D20 (year), D21 (month), D22 (day), D23 (day of week), D24 (hour), D25 (minute), and D26 (second).

The CMP= (compare equal to) instructions compare the current values of month, day, day of week, hour, and minute in data registers D21 through D25 with the comparison data in five consecutive data registers. When data matches, a corresponding internal relay is turned on.

The first CMP= makes the following comparison in five repeat cycles and turns internal relays on:

D21 ↔ D30 → M10 (month)
D22 ↔ D31 → M11 (day)
D23 ↔ D32 → M12 (day of week)
D24 ↔ D33 → M13 (hour)
D25 ↔ D34 → M14 (minute)

The next CMP= compares D21 through D25 with D35 through D39 and turns internal relays M20 through M24 on. Subsequent CMP= instructions make similar comparisons.

When the current time matches comparison data 1 of month, day, hour, and minute, then internal relays M10, M11, M13, and M14 are turned on, and output Q0 is turned on.

When the current time matches comparison data 3 or 5, output Q0 is also turned on.

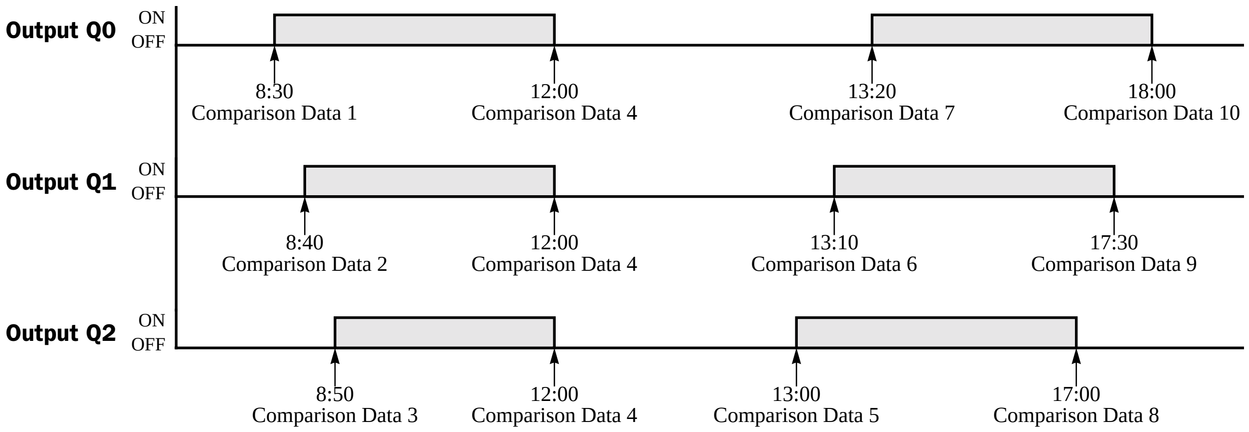
When the current time matches comparison data 2 of month, day, hour, and minute, internal relays M20, M21, M23, and M24 are turned on, and output Q0 is turned off.

When the current time matches comparison data 4 or 6, output Q0 is also turned off.

The day of week is not included in the comparison condition.

Example: Sequential Start

This example demonstrates a program to turn outputs Q0 through Q2 on and off in sequence according to the chart below:



Internal relays are allocated as shown below.

Comparison Data	Hour	Minute	Comparison Data	Hour	Minute
Comparison Data 1 (8:30)	M10	M11	Comparison Data 6 (13:10)	M22	M23
Comparison Data 2 (8:40)	M12	M13	Comparison Data 7 (13:20)	M24	M25
Comparison Data 3 (8:50)	M14	M15	Comparison Data 8 (17:00)	M26	M27
Comparison Data 4 (12:00)	M16	M17	Comparison Data 9 (17:30)	M30	M31
Comparison Data 5 (13:00)	M20	M21	Comparison Data 10 (18:00)	M32	M33

Data registers are allocated as shown below. Comparison data must be set to data registers D20 through D39 in advance using the program loader. For example, to enter decimal value "8" to data register D20, press the keys:

MON E/OR D 2 BRD 0 LOD 10 8 MCS/R ↵

Clock Readout Data

Hour D10	
Minute D11	
Second D12	

Comparison Data 1
(Turn output Q0 on)

Hour D20	8	→ M10
Minute D21	30	→ M11

Comparison Data 2
(Turn output Q1 on)

Hour D22	8	→ M12
Minute D23	40	→ M13

Comparison Data 3
(Turn output Q2 on)

Hour D24	8	→ M14
Minute D25	50	→ M15

Comparison Data 4
(Turn outputs Q0-Q2 off)

Hour D26	12	→ M16
Minute D27	0	→ M17

Comparison Data 5
(Turn output Q2 on)

Hour D28	13	→ M20
Minute D29	0	→ M21

Comparison Data 6
(Turn output Q1 on)

Hour D30	13	→ M22
Minute D31	10	→ M23

Comparison Data 7
(Turn output Q0 on)

Hour D32	13	→ M24
Minute D33	20	→ M25

Comparison Data 8
(Turn output Q2 off)

Hour D34	17	→ M26
Minute D35	0	→ M27

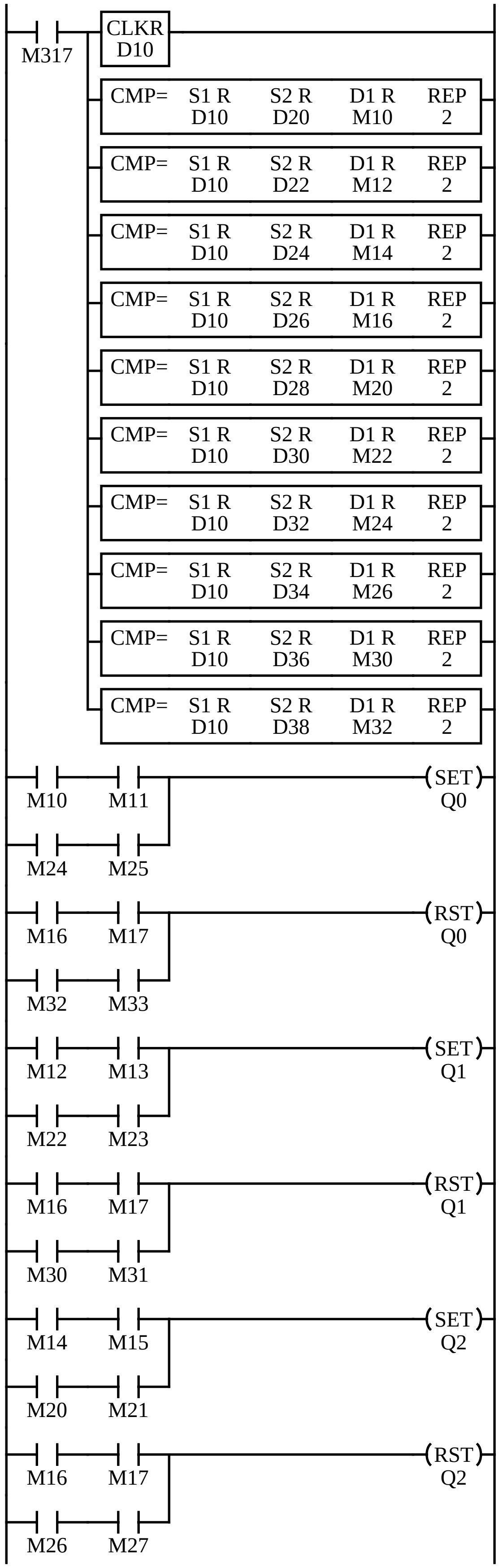
Comparison Data 9
(Turn output Q1 off)

Hour D36	17	→ M30
Minute D37	30	→ M31

Comparison Data 10
(Turn output Q0 off)

Hour D38	18	→ M32
Minute D39	0	→ M33

Example: Sequential Start, continued


M317
CLKR D10
CMP= S1 R D10 S2 R D20 D1 R M10 REP 2
CMP= S1 R D10 S2 R D22 D1 R M12 REP 2
CMP= S1 R D10 S2 R D24 D1 R M14 REP 2
CMP= S1 R D10 S2 R D26 D1 R M16 REP 2
CMP= S1 R D10 S2 R D28 D1 R M20 REP 2
CMP= S1 R D10 S2 R D30 D1 R M22 REP 2
CMP= S1 R D10 S2 R D32 D1 R M24 REP 2
CMP= S1 R D10 S2 R D34 D1 R M26 REP 2
CMP= S1 R D10 S2 R D36 D1 R M30 REP 2
CMP= S1 R D10 S2 R D38 D1 R M32 REP 2
M10 M11 M24 M25 SET Q0
M16 M17 M32 M33 RST Q0
M12 M13 M22 M23 SET Q1
M16 M17 M30 M31 RST Q1
M14 M15 M20 M21 SET Q2
M16 M17 M26 M27 RST Q2

M317 is the in-operation special internal relay which remains on during operation.

While the program is executed, the CLKR (clock read) instruction reads the clock data of hour, minute, and second to data registers D10, D11, and D12.

The CMP= (compare equal to) instructions compare the current values of hour and minute in data registers D10 and D11 with the comparison data in two consecutive data registers. When data matches, a corresponding internal relay is turned on.

The first CMP= makes comparison in two repeat cycles and turns internal relays on:

D10 ↔ D20 → M10 (hour)
D11 ↔ D21 → M11 (minute)

Similarly, subsequent CMP= instructions compare D10 and D11 with two consecutive data registers and turns internal relays M12 through M33 on.

When the current time matches comparison data 1 or 7 of hour and minute, internal relays M10 and M11 or M24 and M25 are turned on, and output Q0 is turned on.

When the current time matches comparison data 4 or 10 of hour and minute, internal relays M16 and M17 or M32 and M33 are turned on, and output Q0 is turned off.

When the current time matches comparison data 2 or 6 of hour and minute, internal relays M12 and M13 or M22 and M23 are turned on, and output Q1 is turned on.

When the current time matches comparison data 4 or 9 of hour and minute, internal relays M16 and M17 or M30 and M31 are turned on, and output Q1 is turned off.

When the current time matches comparison data 3 or 5 of hour and minute, internal relays M14 and M15 or M20 and M21 are turned on, and output Q2 is turned on.

When the current time matches comparison data 4 or 8 of hour and minute, internal relays M16 and M17 or M26 and M27 are turned on, and output Q2 is turned off.

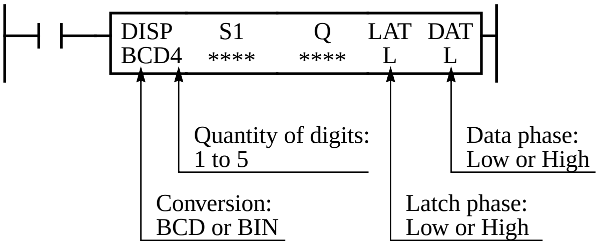
Introduction

The DISP (display) instruction is used to display 1 through 5 digits of timer/counter current values and data register data on 7-segment display units.

The DGRD (digital switch read) instruction is used to read 1 through 5 digits of digital switch settings to a data register or 16 internal relay points. This instruction is useful to change preset values for timers and counters using digital switches.

The ANR0 and ANR1 (analog read) instructions are used to read the analog value (0 through 255) set on the analog potentiometer on the MICRO^3 base unit to a data register.

81 DISP (Display)



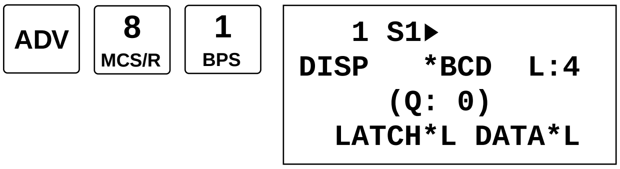
When input is on, data designated by source operand S1 is set to outputs designated by operand Q. This instruction is used to output 7-segment data to display units.

Note: The DISP instruction can be used on transistor output type MICRO^3 base units only.

The DISP instruction can be used only once in a user program.

Display data can be 0 through 65535 (FFFFh).

Key Operation



```
   1 S1▶
DISP   *BCD  L:4
     (Q: 0)
  LATCH*L DATA*L
```

Enter operands S1, Q, and quantity of digits (1 to 5).

To select the conversion, latch phase, and data phase, press the REP key.

To exit, press the ↵ key.

Valid Operands (Standard Processing)

Operand	Function	I	Q	M	T	C	R	D	Constant	Repeat
S1 (Source 1)	Data to display	—	—	—	0-31	0-31	—	0-99	—	—
Q	First output number to display data	—	0-25	—	—	—	—	—	—	—

In the high-speed processing mode, operands for advanced instructions are limited. See page 6-1.
When T (timer) or C (counter) is used as S1, the timer/counter current value is read out.

Conversion

BCD: To connect decimal display units
BIN: To connect hexadecimal display units

Latch Phase and Data Phase

Select the latch and data phases to match the phases of the display units in consideration of sink or source output of the MICRO^3 base unit.

Output Points

The quantity of required output points is 4 plus the quantity of digits to display. When displaying the maximum of 5 digits, 9 consecutive output points must be reserved starting with the first output number designated by operand Q. Make sure that actual output terminals are available for all output numbers. Do not let the output numbers straddle the base and expansion stations in the expansion link system.

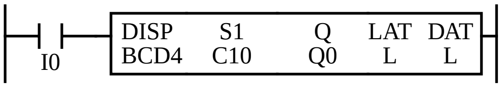
Display Processing Time

Displaying numerical data requires the following time after the input to the DISP instruction is turned on. Keep the input to the DISP instruction for the period of time shown below to process the display data.

Scan Time	Display Processing Time
Less than 5 msec	(10 msec + 1 scan time) × Quantity of digits
5 msec or more	3 scan times × Quantity of digits

Example: DISP

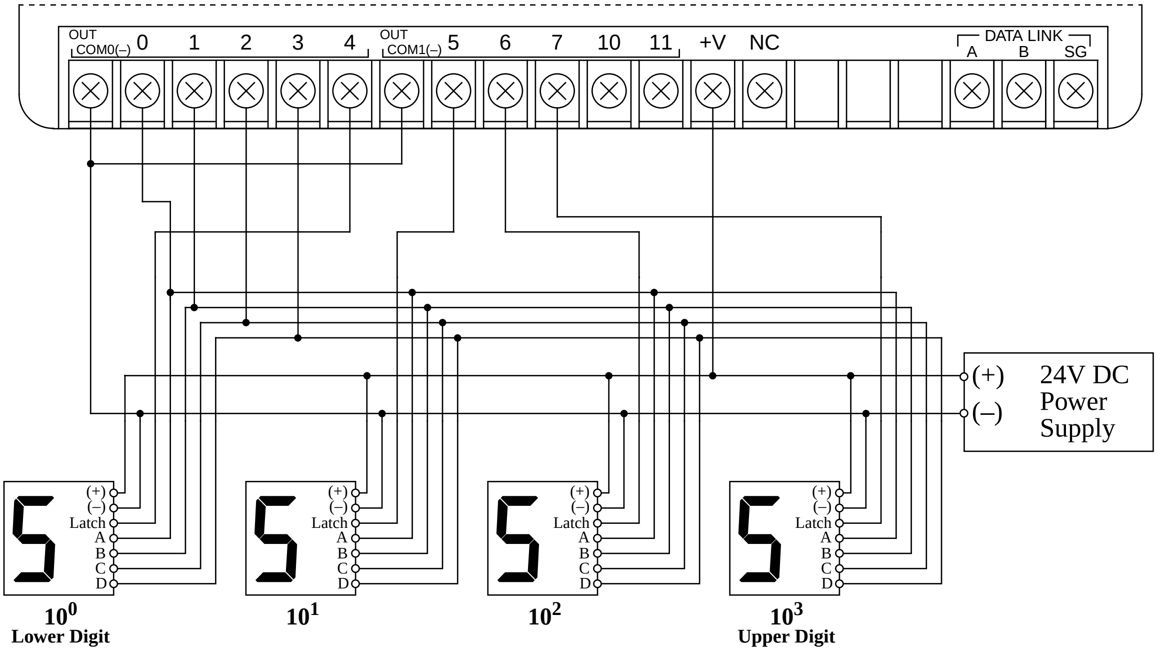
The following example demonstrates a program to display the 4-digit current value of counter CNT10 on display units connected to the 24-I/O transistor sink output type MICRO^3 base unit.



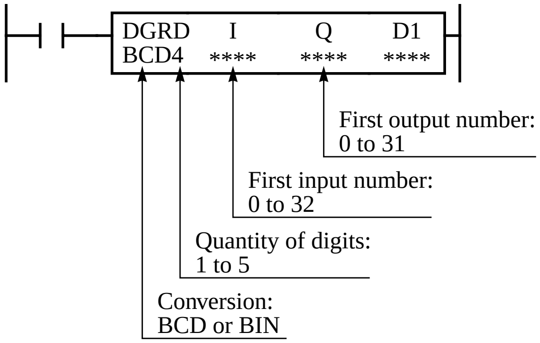
When input I0 is on, the 4-digit current value of counter C10 is displayed on 7-segment digital display units.

Output Wiring Diagram

MICRO^3 Base Unit FC2A-C24B1 (Transistor Sink Output)



82 DGRD (Digital Read)



When input is on, data designated by operands I and Q is set to 16 internal relays or a data register designated by destination operand D1.

This instruction can be used to set preset values for timer (TIM, TMH, and TMS), counter (CNT), and counter comparison instructions using digital switches. The data that can be read using this instruction is 0 through 65535 (5 digits), or FFFFh.

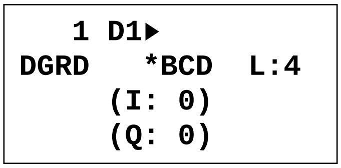
Note: The DGRD instruction can be used on transistor output type MICRO^3 base units only.

The DGRD instruction can be used only once in a user program.

Note: Do not use the DGRD instruction between JMP and JEND instructions or between MCS and MCR instructions.

Key Operation

ADV 8 MCS/R 2 BRD



```
    1 D1▶
DGRD   *BCD  L:4
      (I: 0)
      (Q: 0)
```

Enter operand D1, quantity of digits, first input number to read (I#), and first output number for digit selection (Q#).

To select the conversion, press the REP key.

To exit, press the ↵ key.

Valid Operands (Standard Processing)

Operand	Function	I	Q	M	T	C	R	D	Constant	Repeat
I	First input number to read	0-32	—	—	—	—	—	—	—	—
Q	First output number for digit selection	—	0-31	—	—	—	—	—	—	—
D1 (Destination 1)	Destination to store results	—	—	0-287	—	—	—	0-99	—	—

In the high-speed processing mode, operands for advanced instructions are limited. See page 6-1.

Conversion

BCD: To connect BCD digital switches
BIN: To connect hexadecimal digital switches

Input Points

The inputs are used to read the data from the digital switches. The quantity of required input points is always 4. Four input points must be reserved starting with the input number designated by operand I. Make sure that actual input terminals are available for all input numbers. Do not let the input numbers straddle the base and expansion stations in the expansion link system.

Output Points

Outputs are used to select the digits to read. The quantity of required output points is equal to the quantity of digits to read. When connecting the maximum of 5 digital switches, 5 output points must be reserved starting with the output number designated by operand Q. Make sure that actual output terminals are available for all output numbers. Do not let the output numbers straddle the base and expansion stations in the expansion link system.

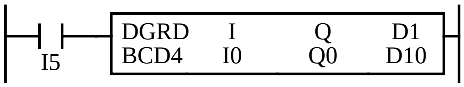
Digital Switch Data Reading Time

Reading digital switch data requires the following time after the input to the DGRD instruction is turned on. Keep the input to the DGRD instruction for the period of time shown below to read the digital switch data.

Scan Time	Digital Switch Data Reading Time
$\text{Scan time} > \dfrac{\text{Soft filter} + 3\text{ msec}}{2}$	2 scan times × (Quantity of digits + 1)
$\text{Scan time} \leq \dfrac{\text{Soft filter} + 3\text{ msec}}{2}$	(Soft filter + 3 msec) × (Quantity of digits + 1)

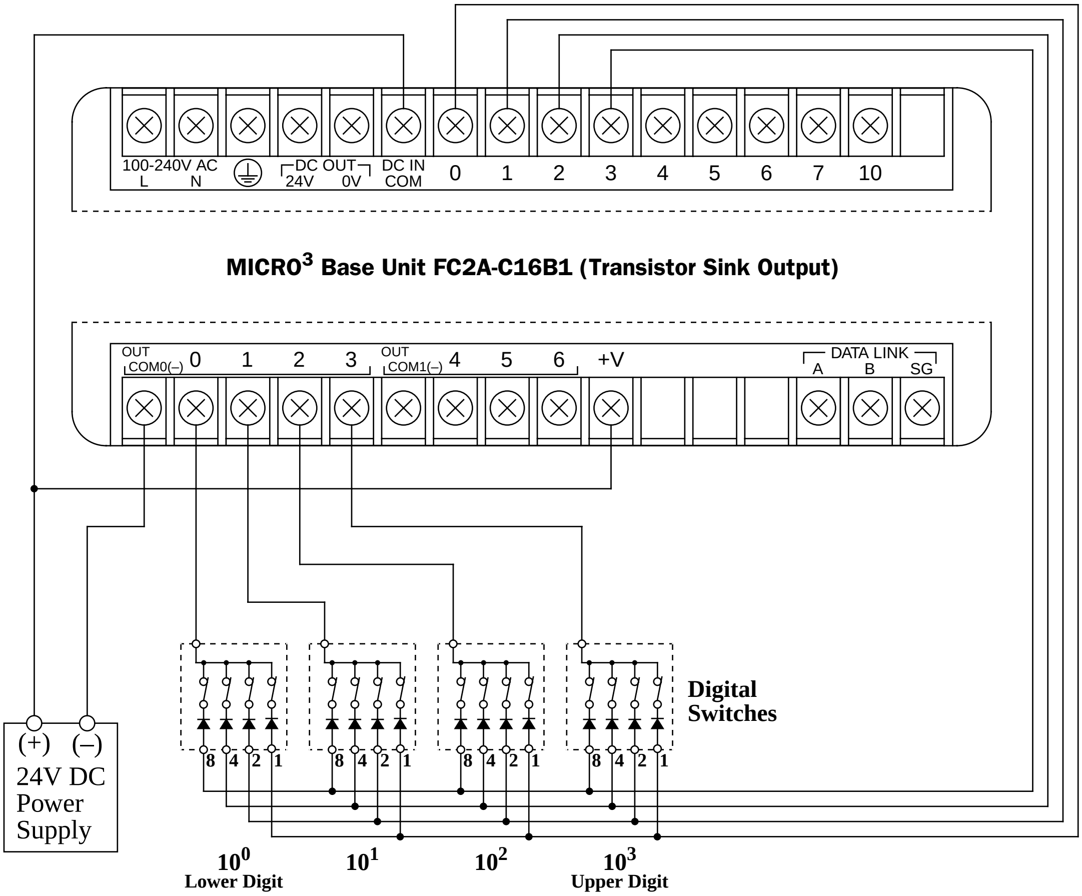
Example: DGRD

The following example demonstrates a program to read data from four digital switches to a data register in the 16-I/O transistor sink output type MICRO^3 base unit.

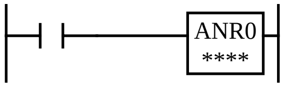


When input I5 is on, the 4-digit value from BCD digital switches is read to data register D10.

I/O Wiring Diagram

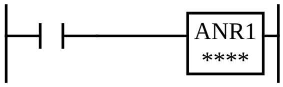


83 ANR0 (Analog Read 0)



When input is on, the value (0 through 249) set with analog potentiometer 0 is read to the data register designated as destination. This instruction is useful for adjusting preset values of timer (TIM, TMH, and TMS) and pulse (PULS and PWM) instructions.

84 ANR1 (Analog Read 1)

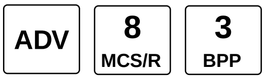


When input is on, the value (0 through 249) set with analog potentiometer 1 is read to the data register designated as destination. This instruction is useful for adjusting preset values of timer (TIM, TMH, and TMS) and pulse (PULS and PWM) instructions.

Note: Analog potentiometer 0 is provided on all models of MICRO^3 and MICRO^{3}C. Analog potentiometer 1 is provided on 16- and 24-I/O MICRO^3 base units. So, ANR0 can be used on all models. ANR1 can be used on 16- and 24-I/O MICRO^3 only; not on MICRO^{3}C and 10-I/O MICRO^3.

Key Operation

Press the ADV key, followed by the advanced instruction number.



```
0 LOD   I  0
1 ANR0 ▶
2 END
3 END
```

Enter a data register operand number for the ANR0 or ANR1 instructions to store data read from analog potentiometer 0 or 1.
To enter the instruction, press the ↵ key.

Note: The ANR0 and ANR1 instructions can be used only once each in a user program.

Valid Operands (Standard Processing)

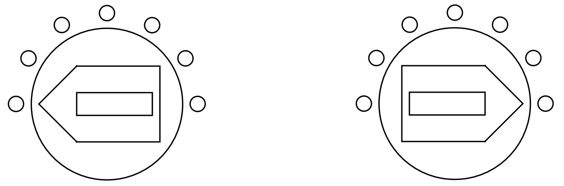
Operand Function	I	Q	M	T	C	R	D	Constant	Repeat
Destination to store the analog potentiometer value	—	—	—	—	—	—	0-99	—	—

In the high-speed processing mode, data registers for these instructions are limited to D0 through D31.
Since the ANR0 and ANR1 instructions are executed in each scan while input is on, a pulse input from an SOTU or SOTD instruction should be used as required.

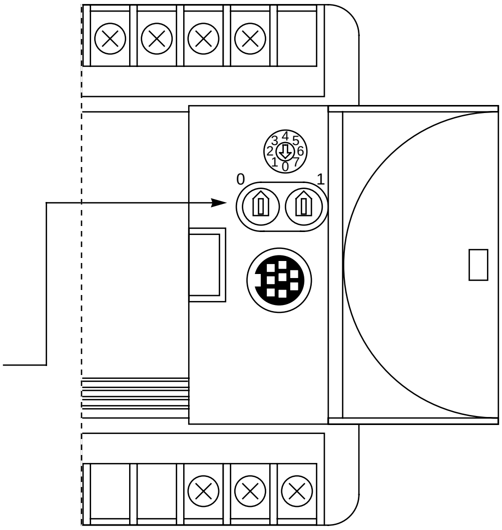
Analog Potentiometer Setting

The analog potentiometer positions and set values are shown below:

Turned Fully to the Left **Turned Fully to the Right**

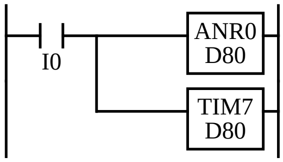


Minimum Value = 0 **Maximum Value = 249**



Analog Potentiometer
The 10-I/O type MICRO^3 and all MICRO^{3}C have one potentiometer.
The 16- and 24-I/O type MICRO^3 have two potentiometers: analog potentiometer 0 on the left and analog potentiometer 1 on the right.

Example: ANR0



When input I0 is on, the value from analog potentiometer 0 is read to data register D80 and is used as a preset value for timer TIM7.

Introduction

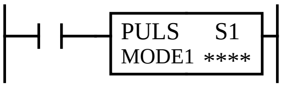
The PULS (pulse output) instruction is used to generate pulse outputs of 9.574 Hz through 13,020.8 Hz which can be used to control pulse motors for simple position control applications. The output pulse ratio is fixed at 50%.

The PWM (pulse width modulation) instruction is used to generate pulse outputs of a 51.2, 25.6, 3.2, or 1.6 msec period with a variable pulse width ratio between 0% and 100%, which can be used for illumination control.

The PULS and PWM instructions can be used on transistor output type MICRO^3 base units only.

The A/D (analog/digital conversion) instruction is used to convert an analog value from the A/D converter unit to a digital value and stores the result to a data register.

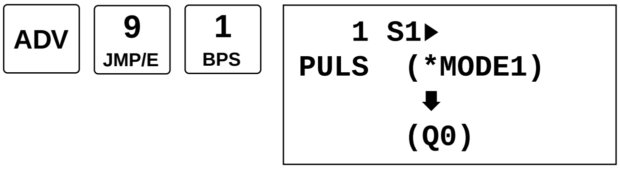
91 PULS (Pulse Output)



When input is on, output Q0 generates a pulse output. The output pulse frequency is determined by the MODE selection and source operand S1 according to the equation below. When input is off, output Q0 remains off.

Note: Either the PULS or PWM instruction can be used only once in a user program.

Key Operation



```
   1 S1▶
PULS  (*MODE1)
        ⬇
       (Q0)
```

Enter operand S1 using the LOD/10 key for a decimal constant, or designate a data register.

Select MODE1 through MODE4 using the REP key.
Do not choose MODE5 and MODE6.

To exit, press the ↵ key.

Valid Operands (Standard Processing)

Operand	Function	I	Q	M	T	C	R	D	Constant	Repeat
S1 (Source 1)	Pulse width coefficient	—	—	—	—	—	—	0-99	0-249	—

In the high-speed processing mode, data registers for this instruction are limited to D0 through D31.

While the MICRO^3 is running, the MODE selection cannot be changed. To change the output pulse frequency during operation, use a data register as source operand S1, and change the value of the data register. See page 3-16.

When a data register is designated as S1, make sure that the value of the data register does not exceed 249. If the value of the data register designated as S1 exceeds 249 during operation, a user program execution error will occur, then error indicator ERR1 on the MICRO^3 base unit is lit and special internal relay M304 is also turned on. Correct the program and transfer the corrected program to the base unit.

When a data register is designated as source operand S1, the data is read as the user program is scanned. When changing the value of the data register designated as S1, change the value slowly in comparison to the output frequency.

When output Q0 is monitored on the program loader while the PULS instruction is executed, Q0 remains on, and the output indicator also remains on. When input to the PULS instruction is turned off while the pulse output is on, the output is turned off after a complete pulse is generated.

Output Frequency

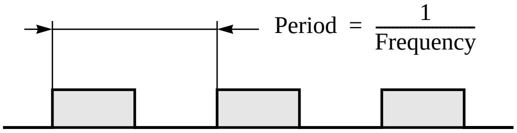
Select MODE1 through MODE4 to determine the base frequency. (Do not choose MODE5 and MODE6.)

MODE	Base Frequency	Output Frequency Range (Coefficient 249 through 0)
MODE1	4882.81 Hz	9.574 through 406.901 Hz
MODE2	9765.63 Hz	19.148 through 813.802 Hz
MODE3	78,125 Hz	153.186 through 6510.42 Hz
MODE4	156,250 Hz	306.373 through 13020.8 Hz

The output frequency is determined by the following equation:

$$\text{Output Frequency} = \frac{\text{Base Frequency (MODE)}}{(\text{Pulse Width Coefficient S1} + 6) \times 2}[\text{Hz}]$$

$$\text{Pulse Width Coefficient S1} = \frac{\text{Base Frequency (MODE)}}{\text{Output Frequency} \times 2} - 6$$



Example: PULS

This example explains how to set 1-kHz output pulses using the PULS instruction.

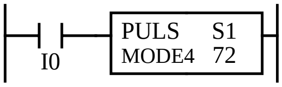
From the table on the preceding page, MODE 3 and MODE4 can be used to set 1 kHz. If MODE4 is selected, then

$$\text{Pulse Width Coefficient S1} = \frac{\text{Base Frequency}}{\text{Output Frequency} \times 2} - 6$$

$$= \frac{156250}{1000 \times 2} - 6$$

$$= 72.125$$

Pulse width coefficient 72 should be used as source operand S1.



When input I0 is on, output Q0 generates pulse outputs of 1001.6 Hz.

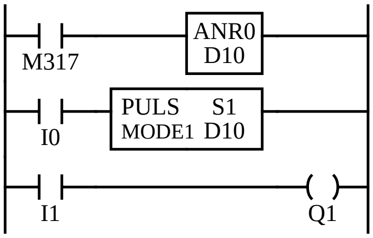
$$\text{Output Frequency} = \frac{156250}{(72 + 6) \times 2} = 1001.6 \text{ Hz}$$

Example: Pulse Motor Speed Control Using PULS Instruction

This example demonstrates a program to control the rotating speed of a pulse motor using the PULS instruction. Analog potentiometer 0 is used to change the pulse motor speed. When input I0 is on, the pulse output is generated to rotate the pulse motor. When input I1 is on, the pulse motor rotates in the reverse direction.

Operands

I0	Input to execute the PULS instruction and start the pulse motor
I1	Input to reverse the pulse motor rotation
Q0	Pulse output
Q1	Output to reverse the pulse motor rotation
D10	Pulse width coefficient
MODE1	9.574 through 406.901 Hz



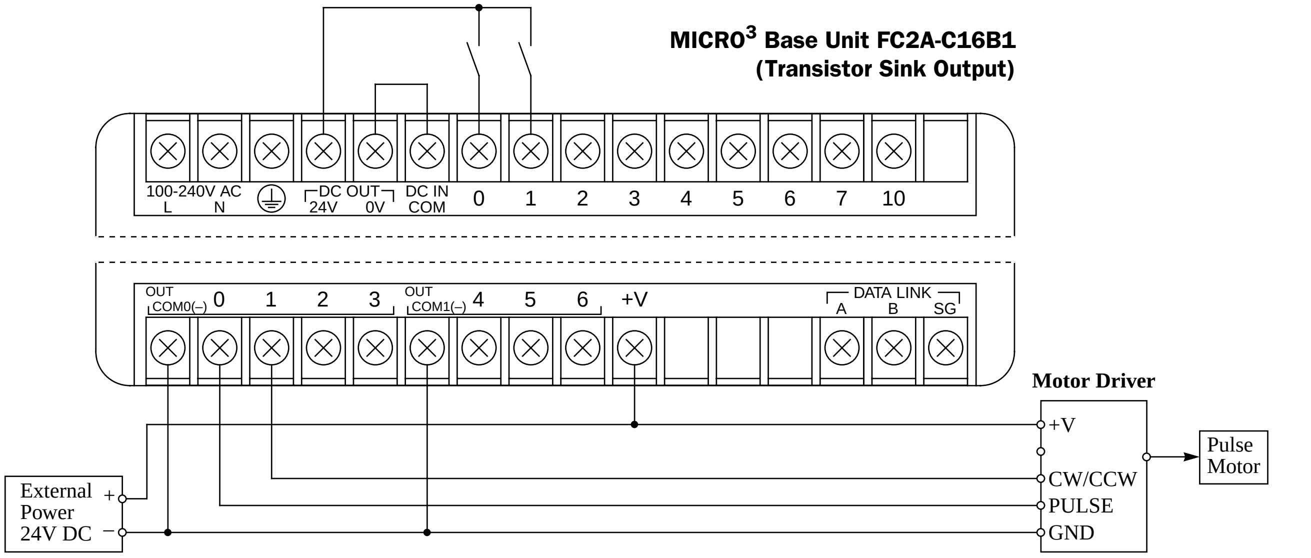
M317 is the in-operation output special internal relay which remains on while the program is executed.

The ANR0 (analog read 0) instruction sets the value of analog potentiometer 0 to data register D10.

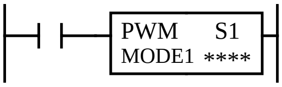
While input I0 is on, the PULS instruction is executed to generate output pulses determined by the value of D10. Output Q0 sends out the output pulses.

When input I1 is on, output Q1 is turned on to reverse the pulse motor.

I/O Wiring Diagram



92 PWM (Pulse Width Modulation)



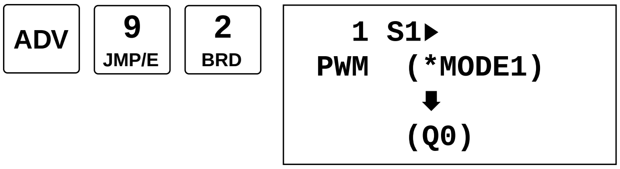
When input is on, output Q0 generates a pulse output. The period of the pulse output is selected from 51.2, 25.6, 3.2, or 1.6 msec. The output pulse width ratio is determined by source operand S1 according to the equation shown below.

When input is off, output Q0 remains off.

Note: Either the PULS or PWM instruction can be used only once in a user program.

Note: When the PWM instruction used in the protect source output type MICRO^3, the output protection function does not work on output Q0.

Key Operation



Enter operand S1 using the LOD/10 key for a decimal constant, or designate a data register.

Select MODE1 through MODE4 using the REP key.
Do not choose MODE5 and MODE6.

To exit, press the ↵ key.

Valid Operands (Standard Processing)

Operand	Function	I	Q	M	T	C	R	D	Constant	Repeat
S1 (Source 1)	Pulse width coefficient	—	—	—	—	—	—	0-99	0-249	—

In the high-speed processing mode, data registers for this instruction are limited to D0 through D31.

The pulse cycle period (MODE selection) cannot be changed during operation. To change the duty ratio during operation, use a data register as source operand S1, and change the value of the data register. See page 3-16. If the value of the data register designated as S1 is between 0 and 4, the pulse width coefficient is designated as 5.

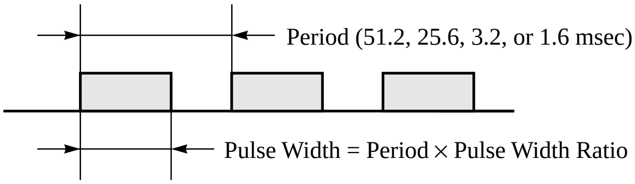
When a data register is designated as S1, make sure the value of the data register does not exceed 249. If the value of the data register designated as S1 exceeds 249 during operation, a user program execution error will occur, error indicator ERR1 on the MICRO^3 base unit is lit, and special internal relay M304 is turned on. Correct the program and transfer corrected program to the base unit. When a data register is designated as S1, the data is read as the user program is scanned. When changing the value of the data register designated as S1, change the value slowly in comparison to the output frequency.

When output Q0 is monitored on the program loader while the PWM instruction is executed, Q0 remains on, and the output indicator also remains on. When input to the PWM instruction is turned off while the pulse output is on, the output is turned off after a complete pulse is generated.

Output Pulse Width Ratio

Select MODE1 through MODE4 to determine the pulse cycle period. (Do not choose MODE5 and MODE6.)

MODE	Pulse Cycle Period
MODE1	51.2 msec
MODE2	25.6 msec
MODE3	3.2 msec
MODE4	1.6 msec


Period (51.2, 25.6, 3.2, or 1.6 msec)
Pulse Width = Period × Pulse Width Ratio

The output pulse width ratio is determined by the following equation:

$$\text{Pulse Width Ratio} = \frac{\text{Pulse Width Coefficient S1} + 1}{250}$$

$$\text{Pulse Width} = \text{Period} \times \frac{\text{Pulse Width Coefficient S1} + 1}{250} [\text{msec}]$$

Variable Range of Pulse Width Ratio

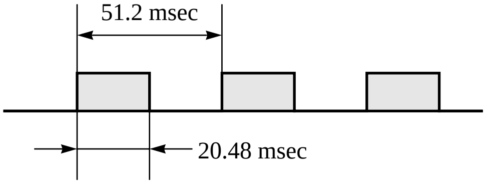
When S1 is a data register:	2.4% through 100% in 0.4% increments
When S1 is a constant:	0.4% through 100% in 0.4% increments

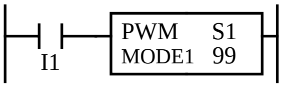
To turn the pulse output off, turn the input to the PWM instruction off.

Example: PWM

When MODE1 (pulse cycle period 51.2 msec) is selected and 99 is set to S1, the output pulse waveform is as follows.

$$
\begin{aligned}
\text{Pulse Width} &= \text{Period} \times \frac{\text{Pulse Width Coefficient S1} + 1}{250} \\
&= 51.2 \text{ msec} \times \frac{99 + 1}{250} \\
&= 20.48 \text{ msec}
\end{aligned}
$$





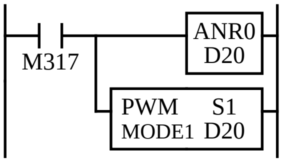
When input I1 is on, output Q0 generates a pulse output shown above.

Example: Illumination Control Using PWM Instruction

This example demonstrates a program to control incandescent lamp illumination using the PWM instruction. Analog potentiometer 0 is used to change the illumination intensity.

Operands

Q0	Pulse output
D20	Pulse width coefficient
MODE1	Pulse cycle period 51.2 msec



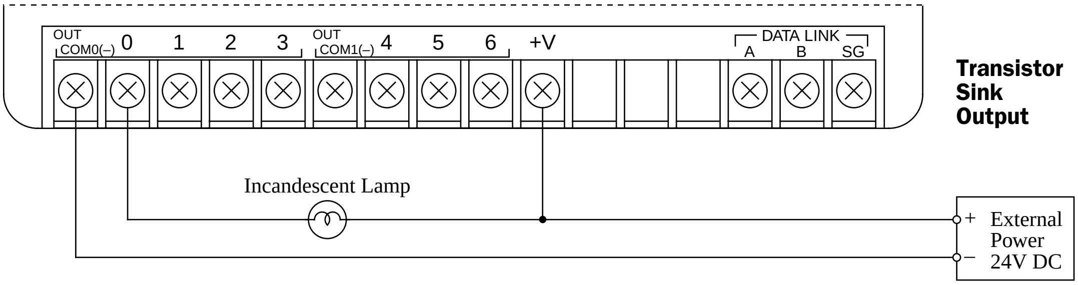
M317 is the in-operation output special internal relay which remains on while the program is executed.

The ANR0 (analog read 0) instruction sets the value of analog potentiometer 0 to data register D20.

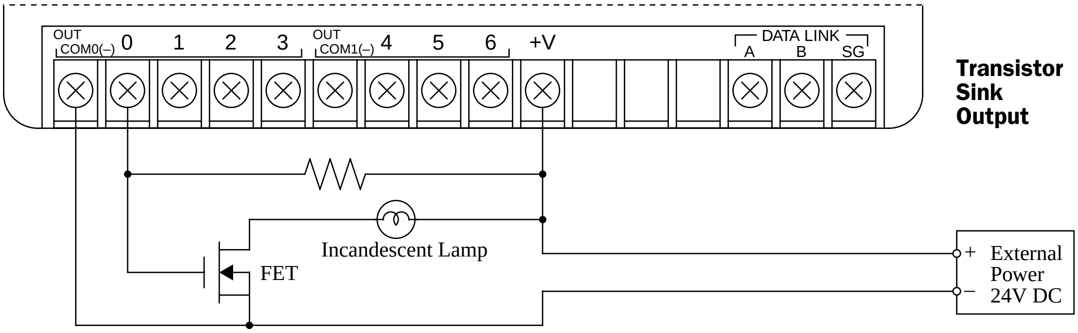
The PWM instruction is executed to generate output pulses. The pulse width ratio is determined by the value of data register D20. Output Q0 sends out the output pulses.

Output Wiring Diagram

- **When using an incandescent lamp of 0.5A or less**

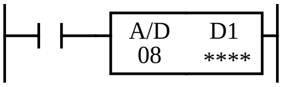


- **When using an incandescent lamp of 0.5A or more**



Note: Provide protection against a rush current depending on the load.

93 A/D (Analog/Digital Conversion)

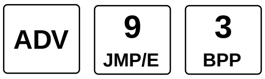


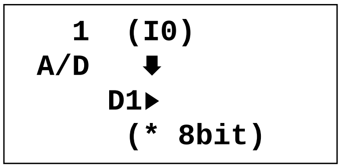
When input is on, the analog data from the A/D converter unit (FC2A-AD1, -AD2, -AD3, -AD4, or -AD5) connected to input I0 is converted to a digital value and set to a data register designated by destination operand D1.

Resolution of the A/D conversion is 8 bits. Conversion time is 125 msec.

Note: Either the A/D or HSC (high-speed counter) instruction can be used only once in a user program. The A/D converter unit can be connected to input I0 of MICRO^3 at the base station only, not at the expansion station.

Key Operation





```
   1  (I0)
A/D   ⬇
    D1▶
     (* 8bit)
```

Enter operand D1.

Select the resolution of 8 bits using the REP key.

To exit, press the ↵ key.

Valid Operands (Standard Processing)

Operand	Function	I	Q	M	T	C	R	D	Constant	Repeat
D1 (Destination 1)	Destination to store data	—	—	—	—	—	—	0-99	—	—

In the high-speed processing mode, data registers for this instruction are limited to D0 through D31.

Digital Data Range

Depending on the resolution of A/D conversion, the digital data stored in the data register is limited to the range shown below:

Resolution	Digital Data Range
8 bits	0 through 249, or 250

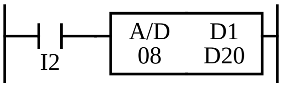
If the input to the A/D converter unit exceeds the input range, an overflow occurs and 250 is set.

Resolution

When the user program is cleared from the program loader memory, the resolution for the A/D instruction is set to the default value of 8 bits.

Although pressing the REP key on the program loader toggles the resolution between 8 bits and 12 bits, select the resolution of 8 bits for programming the MICRO^3 used with the A/D converter unit (FC2A-AD1, -AD2, -AD3, -AD4, or -AD5).

Example: A/D



When input I2 is on, the analog data from the A/D converter unit is converted to a digital value with an 8-bit resolution and set to data register D20.

Introduction

MICRO^3 features high-speed counter functions which can be used for position control by counting high-speed pulses or for simple motor control in combination with the pulse output. This function can also be used in combination with a pulse generator to measure lengths or widths of objects.

The ordinary counter instruction counts only one pulse in one scan, and the counting speed depends on the scan time. The high-speed counter can count many input pulses in one scan and make it possible to count high-speed pulses faster than the scan time. If the high-speed counter counts input pulses representing a position, the current position can be determined. This function is useful for position control.

The HSC0 is a high-speed counter with a single-stage comparison function. When the current value is equal to or greater than the preset value (4,294,967,295 maximum), a designated output or internal relay is turned on.

The HSC1 is a multi-stage comparison counter. The preset value and output data are programmed in data registers. When preset values are reached (4,294,967,295 maximum), designated outputs or internal relays are turned on in sequence.

The HSC2 is a pulse output control counter used with the PULS (pulse output) instruction. When a preset value is reached, a designated output or internal relay is turned on, and the pulse output at output Q0 is turned off.

The HSC3 is a gate-controlled counter without comparison function. When the gate input is turned off, the current value is moved to a designated data register.

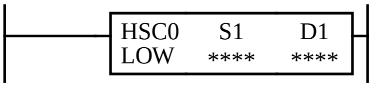
Note: The high-speed counter function can be used with the 24V DC input type MICRO^3 only, not with the AC input type.

High-speed Counter Specifications (Hard Filter Value: 10)

High-speed Counter	HSC0	HSC1	HSC2	HSC3
Counted Value Range	0 to 4,294,967,295 (FFFF FFFFh)	0 to 4,294,967,295 (FFFF FFFFh)	0 to 4,294,967,295 (FFFF FFFFh)	0 to 65,535 (FFFFh)
Points	1 point	1 point	1 point	1 point
Phase	Single phase	Single phase	Single phase	Single phase
Maximum Frequency	10 kHz	5 kHz	5 kHz (4 kHz when using program loader)	10 kHz

Note: The input response frequency of the high-speed counter depends on the hard filter setting. The soft filter does not affect the high-speed counter function. See Input Filter Function on page 4-3.

A1 HSC0 (Single-stage Comparison)

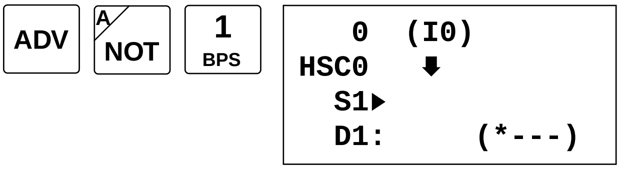


High-speed counter 0 counts input pulses to input I0. When the current value is equal to or greater than the preset value designated by source operand S1, the output or internal relay designated by destination operand D1 is turned on.

The high-speed counter current value is reset to 0 when MICRO^3 is powered up. The high-speed counter holds the current value while MICRO^3 is stopped and restarts counting input pulses starting with the existing current value. Include the hard reset or soft reset in the user program, if necessary.

Note: Only one of HSC0 through HSC3 and A/D instructions can be used only once in a user program.

Key Operation



```
   0  (I0)
HSC0   ⬇
   S1▶
   D1:     (*---)
```

Enter operands S1 and D1.

To select hard reset mode from LOW, HIGH, or unused, press the REP key.

To exit, press the ↵ key.

Valid Operands (Standard Processing)

Operand	Function	I	Q	M	T	C	R	D	Constant	Repeat
S1 (Source 1)	Preset value	—	—	—	—	—	—	0-99	1-4,294,967,295	—
D1 (Destination 1)	High-speed counter output	—	0-31	0-287	—	—	—	—	—	—

In the high-speed processing mode, operands for advanced instructions are limited. See page 6-1.

Allocation Numbers

The HSC0 instruction uses the following input and internal relay numbers:

Pulse input:	Input I0
Hard reset input:	Input I1
Soft reset special internal relay:	Internal relay M315 (When M315 is on, the current value is reset to 0.)

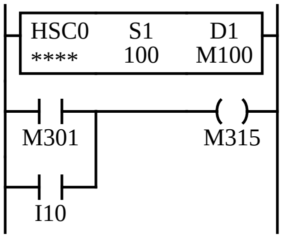
Hard Reset Selection

Input I1 can be used to reset the current value of high-speed counter HSC0.

LOW: Resets the current value when input I1 is turned off. HSC0 is enabled while I1 is on.
HIGH: Resets the current value when input I1 is turned on. HSC0 is enabled while I1 is off.
***---:** Disables hard reset. (Input I1 can be used as an ordinary input.)

Soft Reset Special Internal Relay M315

In addition to the hard reset using input I1, the high-speed counter current value can be reset by turning special internal relay M315 on using another input, output, or internal relay. M315, if used, must be programmed immediately after the HSC0 instruction as shown on the right.



M301 is the initialize pulse special internal relay.

Preset Value

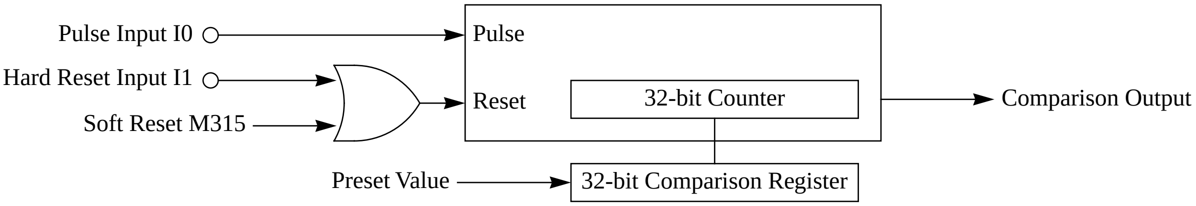
The preset value can be 1 through 4,294,967,295 (FFFF FFFFh), which is designated using a constant or two consecutive data registers. The first data register designated by source operand S1 stores the upper digits, and the next data register stores the lower digits. To enter a double-word value to two consecutive data registers using the program loader, from the editor mode press the MON, D, data register number, ADV, followed by the LOD/10 (decimal) or OUT/16 (hexadecimal), data register value, and ↵ keys. See page 3-16. If the preset value designated by a data register is changed during high-speed counter operation, the high-speed counter remains unchanged for that cycle. The change will be reflected in the next count cycle after resetting.

Input Filter and Input Frequency

MICRO^3 has hard and soft filter functions. Only the hard filter works on high-speed counter instructions. The hard filter settings affect the input response. See page 4-3. The maximum input frequency for the HSC0 instruction is 10 kHz.

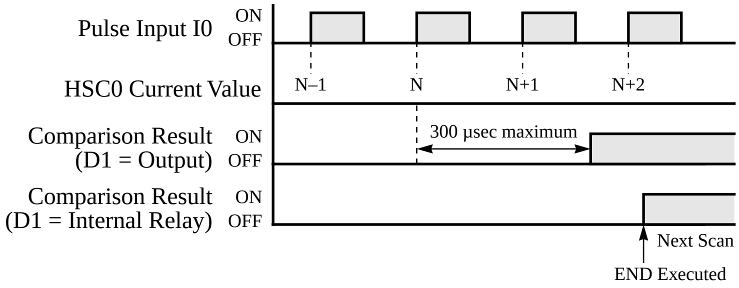
Block Diagram (HSC0: Single-stage Comparison)

HSC0 counts input pulses to input I0. When the preset value is reached, comparison output is turned on.



Output Delay (HSC0: Single-stage Comparison)

After the HSC0 has counted the Nth input pulse (the preset value), the output or internal relay designated by destination operand D1 is turned on with a delay shown below.

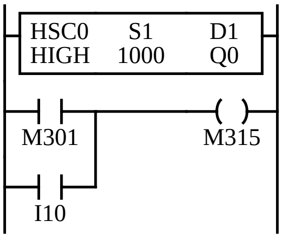


When an output is designated as destination operand D1, the maximum output delay can be 300 µsec, not including the delay in the hardware.

When an internal relay is designated as destination operand D1, the delay can be 1 scan time at the maximum.

Note: After the preset value has been reached, the HSC0 current value continues to increase.

Example: HSC0



I0: Pulse input
I1: Hard reset input (HSC0 is reset when I1 is on because the hard reset is set to HIGH.)
I10: Soft reset input to turn on soft reset special internal relay M315

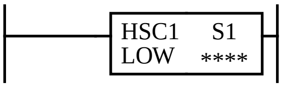
M301 is the initialize pulse special internal relay used to turn soft reset special internal relay M315 on at start up.

While hard reset input I1 is off, the HSC0 instruction counts input pulses to input I0. When the HSC0 current value reaches 1000, output Q0 is turned on.

When hard reset input I1 or soft reset input I10 is turned on, the HSC0 current value is reset to 0.

For monitoring high-speed counter preset and current values, see page 3-15.

A2 HSC1 (Multi-stage Comparison)



Multi-stage high-speed counter HSC1 counts input pulses to input I0 and compares the current value with multiple preset values. When the current value reaches the first preset value, the first comparison output is turned on. When the second preset value is reached, the first comparison output is turned off, the second comparison output is turned on, and so forth.

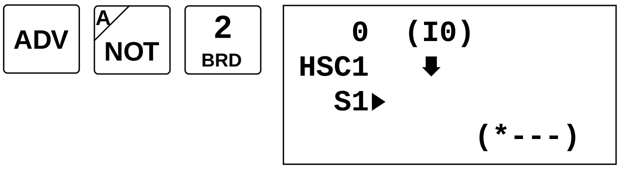
The high-speed counter current value is reset to 0 when MICRO³ is powered up. The high-speed counter holds the current value while MICRO³ is stopped and restarts counting input pulses starting with the existing current value. Include the hard reset or soft reset in the user program, if necessary.

Note: Only one of HSC0 through HSC3 and A/D instructions can be used only once in a user program.

⚠ **Caution**

When a slave station performs communication at 19,200 bps through the loader port in the data link system, multi-stage comparison instruction HSC1 cannot be used at the slave station.

Key Operation



```
   0  (I0)
HSC1   ⬇
  S1▶
           (*---)
```

Enter operand S1.

To select hard reset mode from LOW, HIGH, or unused, press the REP key.

To exit, press the ↵ key.

Valid Operands (Standard Processing)

Operand	Function	I	Q	M	T	C	R	D	Constant	Repeat
S1 (Source 1)	Multi-stage preset data	—	—	—	—	—	—	0-99	—	—

In the high-speed processing mode, data registers for this instruction are limited to D0 through D31.

Allocation Numbers

The HSC1 instruction uses the following input and internal relay numbers:

Pulse input:	Input I0
Hard reset input:	Input I1
Soft reset special internal relay:	Internal relay M315 (When M315 is on, the current value is reset to 0.)

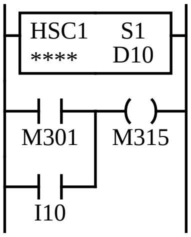
Hard Reset Selection

Input I1 can be used to reset the current value of high-speed counter HSC1.

LOW: Resets the current value when input I1 is turned off. HSC1 is enabled while I1 is on.
HIGH: Resets the current value when input I1 is turned on. HSC1 is enabled while I1 is off.
***---:** Disables hard reset. (Input I1 can be used as an ordinary input.)

Soft Reset Special Internal Relay M315

In addition to the hard reset using input I1, the high-speed counter current value can be reset by turning special internal relay M315 on using another input, output, or internal relay. M315, if used, must be programmed immediately after the HSC1 instruction as shown on the right.



M301 is the initialize pulse special internal relay.

Multi-stage Data Setting

The data of comparison stages, preset values, and comparison outputs are stored in consecutive data registers starting with the data register designated by source operand S1.

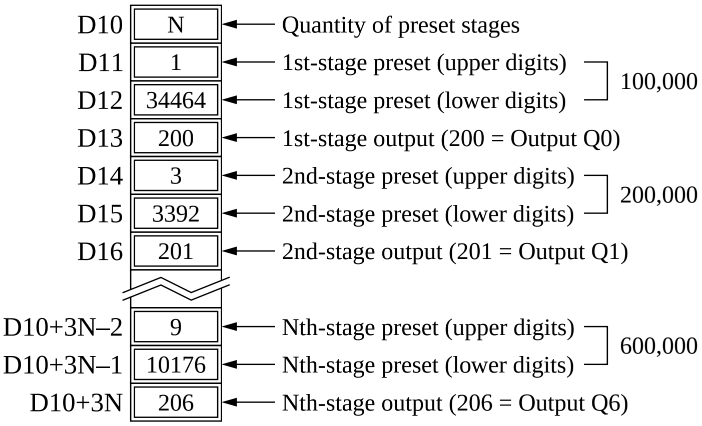
Store the quantity of preset stages in the first data register.

In the next two data registers, store the upper and lower digits of the preset value for the first stage.

In the fourth data register, store the destination of the first-stage comparison output, using a numeric allocation number of output or internal relay. See below.

Store data in these data registers before executing the HSC1 instruction.

When data register D10 is designated as source operand S1


D10 N Quantity of preset stages
D11 1 1st-stage preset (upper digits)
D12 34464 1st-stage preset (lower digits)
100,000
D13 200 1st-stage output (200 = Output Q0)
D14 3 2nd-stage preset (upper digits)
D15 3392 2nd-stage preset (lower digits)
200,000
D16 201 2nd-stage output (201 = Output Q1)
D10+3N–2 9 Nth-stage preset (upper digits)
D10+3N–1 10176 Nth-stage preset (lower digits)
600,000
D10+3N 206 Nth-stage output (206 = Output Q6)

Preset Value

The preset value can be 1 through 4,294,967,295 (FFFF FFFFh), which is stored in two consecutive data registers. The first data register stores the upper digits, and the next data register stores the lower digits. To enter a double-word value to two consecutive data registers using the program loader, from the editor mode press the MON, D, data register number, ADV, followed by the LOD/10 (decimal) or OUT/16 (hexadecimal), data register value, and ↵ keys. See page 3-16. If the preset value is changed during high-speed counter operation, the high-speed counter remains unchanged for that cycle. The change will be reflected in the next count cycle after resetting.

Allocation Numbers: Numeric and Symbolic

Use the numeric allocation numbers to specify the destination of the HSC1 comparison outputs.

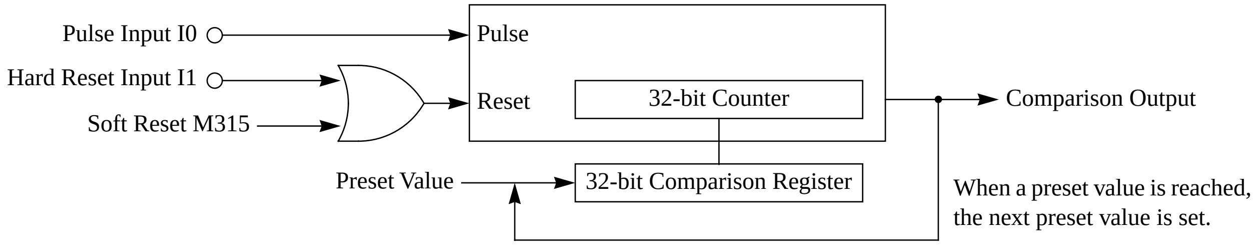
Operand	Symbolic	Numeric	Operand	Symbolic	Numeric	Operand	Symbolic	Numeric
Output	Q0 - Q7	200 - 207	Internal Relay	M70 - M77	470 - 477	Internal Relay	M180 - M187	580 - 587
	Q10 - Q11	210 - 211		M80 - M87	480 - 487		M190 - M197	590 - 597
	Q20 - Q27	220 - 227		M90 - M97	490 - 497		M200 - M207	600 - 607
	Q30 - Q31	230 - 231		M100 - M107	500 - 507		M210 - M217	610 - 617
Internal Relay	M0 - M7	400 - 407		M110 - M117	510 - 517		M220 - M227	620 - 627
	M10 - M17	410 - 417		M120 - M127	520 - 527		M230 - M237	630 - 637
	M20 - M27	420 - 427		M130 - M137	530 - 537		M240 - M247	640 - 647
	M30 - M37	430 - 437		M140 - M147	540 - 547		M250 - M257	650 - 657
	M40 - M47	440 - 447		M150 - M157	550 - 557		M260 - M267	660 - 667
	M50 - M57	450 - 457		M160 - M167	560 - 567		M270 - M277	670 - 677
	M60 - M67	460 - 467		M170 - M177	570 - 577		M280 - M287	680 - 687

Input Filter and Input Frequency

MICRO^3 has hard and soft filter functions. Only the hard filter works on high-speed counter instructions. The hard filter settings affect the input response. See page 4-3. The maximum input frequency for the HSC1 instruction is 5 kHz.

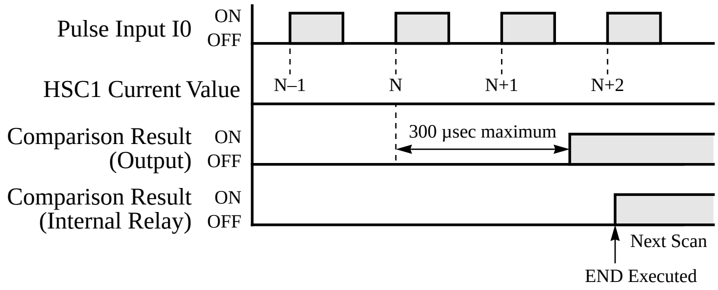
Block Diagram (HSC1: Multi-stage Comparison)

HSC1 counts input pulses to input I0. When the preset value is reached, comparison output is turned on. Multiple preset values and comparison outputs can be programmed.



Output Delay (HSC1: Multi-stage Comparison)

After the HSC1 has counted the Nth input pulse (the preset value), the output or internal relay designated as destination of comparison result is turned on with a delay shown below.



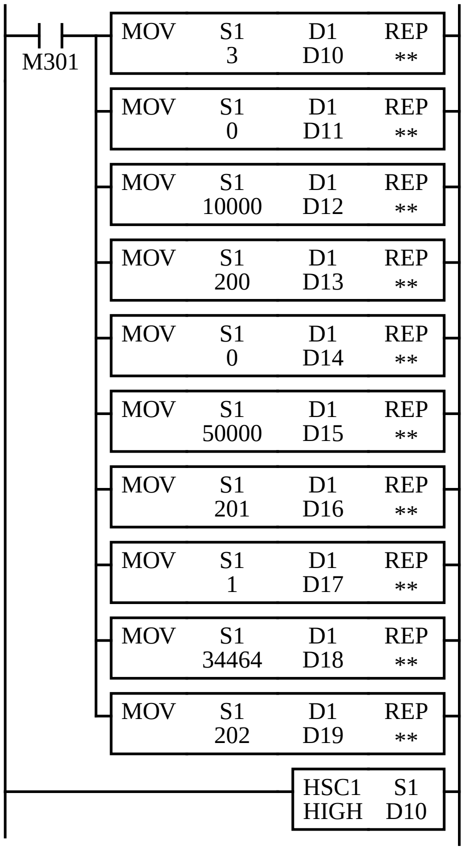
When an output is designated as destination operand, the maximum output delay can be 300 µsec, not including the delay in the hardware.

When an internal relay is designated as destination operand, the delay can be 1 scan time at the maximum.

Note: After the last preset value has been reached, the HSC1 current value continues to increase.

Example: HSC1

This example demonstrates a 3-stage high-speed counter operation using the HSC1 instruction.



M301 is the initialize pulse special internal relay used to execute the MOV (move) instructions at start up.

The MOV instructions set data to data registers D10 through D19.

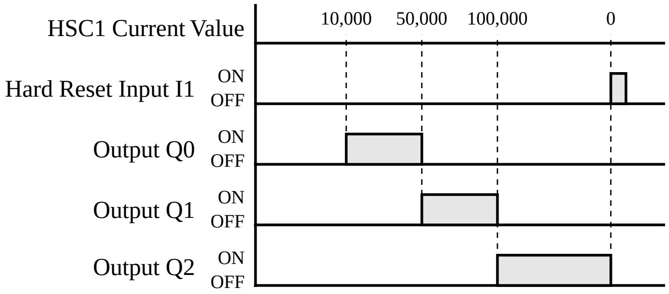
D10	3	Quantity of preset stages	
D11	0	1st-stage preset (upper digits)	10,000
D12	10000	1st-stage preset (lower digits)	
D13	200	1st-stage output (200 = Output Q0)	
D14	0	2nd-stage preset (upper digits)	50,000
D15	50000	2nd-stage preset (lower digits)	
D16	201	2nd-stage output (201 = Output Q1)	
D17	1	3rd-stage preset (upper digits)	100,000
D18	34464	3rd-stage preset (lower digits)	
D19	202	3rd-stage output (202 = Output Q2)	

Preset value 100,000 is set to two data registers D17 (upper digits) and D18 (lower digits). Values for the two data registers are calculated by dividing the preset value by 65,536 (10000h) as follows:

100,000 ÷ 65,536 = 1 and remainder 34,464

Upper Digits (D17) Lower Digits (D18)

Example: HSC1, continued

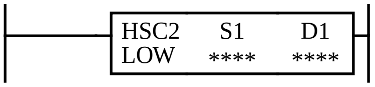


High-speed counter HSC1 counts input pulses to input I0. When the first preset value 10,000 is reached, output Q0 is turned on. When the second preset value 50,000 is reached, output Q0 is turned off, and output Q1 is turned on. When the last preset value 100,000 is reached, output Q1 is turned off, and output Q2 is turned on. Output Q2 remains on until hard reset input I1 is turned on to reset the high-speed counter (hard reset is set to HIGH).

Since this example does not include the soft reset special internal relay to reset at startup, the current value is held when MICRO^3 is stopped.

For monitoring high-speed counter preset and current values, see page 3-15.

A3 HSC2 (Pulse Output Control)



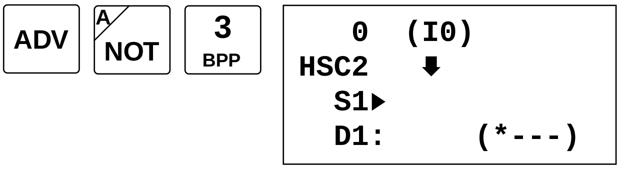
The HSC2 instruction is used with the PULS (pulse output) instruction to generate a predetermined number of pulse outputs. The PULS instruction generates high-frequency output pulses at output Q0. By hard-wiring output Q0 to input I0, HSC2 counts input pulses to input I0. (Input pulses can also be entered to input I0 from another source.) When the HSC2 current value is equal to or greater than the preset value designated by source operand S1, the output or internal relay designated by destination operand D1 is turned on, and the pulse output at output Q0 is stopped.

The high-speed counter current value is reset to 0 when MICRO^3 is powered up. The high-speed counter holds the current value while MICRO^3 is stopped and restarts counting input pulses starting with the existing current value. Include the hard reset or soft reset in the user program, if necessary.

Note: Since the PULS instruction can be used on transistor output type MICRO^3 base units only, the HSC2 instruction can also be used on transistor output type MICRO^3 base units only.

Note: Only one of HSC0 through HSC3 and A/D instructions can be used only once in a user program.

Key Operation



```
  0  (I0)
HSC2   ⬇
  S1▸
  D1:      (*---)
```

Enter operands S1 and D1.

To select hard reset mode from LOW, HIGH, or unused, press the REP key.

To exit, press the ↵ key.

Valid Operands (Standard Processing)

Operand	Function	I	Q	M	T	C	R	D	Constant	Repeat
S1 (Source 1)	Preset value	—	—	—	—	—	—	0-99	1-4,294,967,295	—
D1 (Destination 1)	High-speed counter output	—	0-31	0-287	—	—	—	—	—	—

In the high-speed processing mode, operands for advanced instructions are limited. See page 6-1.

Allocation Numbers

The HSC2 instruction uses the following input and internal relay numbers:

Pulse input:	Input I0
Hard reset input:	Input I1
Soft reset special internal relay:	Internal relay M315 (When M315 is on, the current value is reset to 0.)

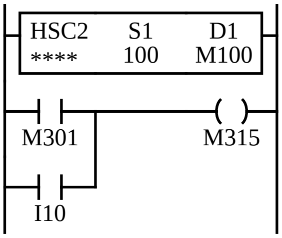
Hard Reset Selection

Input I1 can be used to reset the current value of high-speed counter HSC2.

LOW: Resets the current value when input I1 is turned off. HSC2 is enabled while I1 is on.
HIGH: Resets the current value when input I1 is turned on. HSC2 is enabled while I1 is off.
***---:** Disables hard reset. (Input I1 can be used as an ordinary input.)

Soft Reset Special Internal Relay M315

In addition to the hard reset using input I1, the high-speed counter current value can be reset by turning special internal relay M315 on using another input, output, or internal relay. M315, if used, must be programmed immediately after the HSC2 instruction as shown on the right.



M301 is the initialize pulse special internal relay.

Preset Value

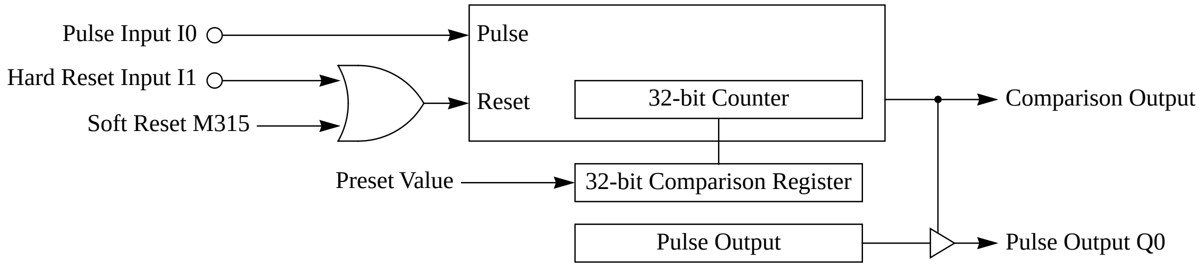
The preset value can be 1 through 4,294,967,295 (FFFF FFFFh), which is designated using a constant or two consecutive data registers. The first data register designated by source operand S1 stores the upper digits, and the next data register stores the lower digits. To enter a double-word value to two consecutive data registers using the program loader, from the editor mode press the MON, D, data register number, ADV, followed by the LOD/10 (decimal) or OUT/16 (hexadecimal), data register value, and ↵ keys. See page 3-16. If the preset value designated by a data register is changed during high-speed counter operation, the high-speed counter remains unchanged for that cycle. The change will be reflected in the next count cycle after resetting.

Input Filter and Input Frequency

MICRO3 has hard and soft filter functions. Only the hard filter works on high-speed counter instructions. The hard filter settings affect the input response. See page 4-3. The maximum input frequency for the HSC2 instruction is 5 kHz (4 kHz when using the program loader).

Block Diagram (HSC2: Pulse Output Control)

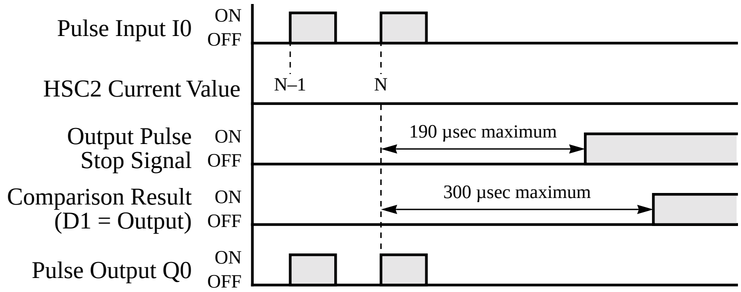
HSC2 counts input pulses to input I0. When the preset value is reached, comparison output is turned on, and pulse output Q0 is turned off. The pulse output frequency is determined by the PULS (pulse output) instruction.



When the preset value is reached, the pulse output is turned off.

Output Delay (HSC2: Pulse Output Control)

After the HSC2 has counted the Nth input pulse (the preset value), the output or internal relay designated by destination operand D1 is turned on with a delay shown below.



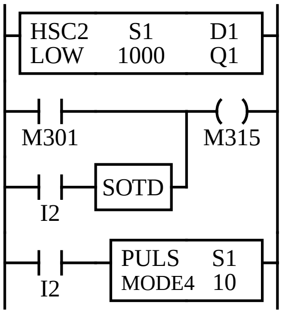
The output pulse stop signal is turned on 190 µsec at the maximum after the current value reaches the preset value.

The example on the left shows the operation when the PULS instruction generates 4882.81-Hz output pulses.

Note: When MICRO3 is restarted without resetting the HSC2 after the preset value has been reached, the comparison result output or internal relay is reset, and the HSC2 continues to count input pulses starting with the preset value.

Example: HSC2

The PULS (pulse output) instruction is used to generate output pulses. The output pulses are sent from output Q0 to input I0. The HSC2 instruction is used to count the pulse signals up to 1000. When the preset value is reached, HSC2 stops the pulse output at Q0 and turns output Q1 on.



I0: Pulse input
I1: Hard reset input (HSC2 is reset when I1 is off because the hard reset is set to LOW.)
I2: Input to execute the PULS (pulse output) instruction and to turn on soft reset special internal relay M315
M301: Initialize pulse special internal relay

While input I2 is on, the PULS instruction generates output pulses at output Q0.

$$\text{Output Frequency} = \frac{\text{Base Frequency (MODE)}}{(\text{Pulse Width Coefficient S1} + 6) \times 2} = \frac{156250}{(10 + 6) \times 2} \cong 4882.81\ \text{Hz}$$

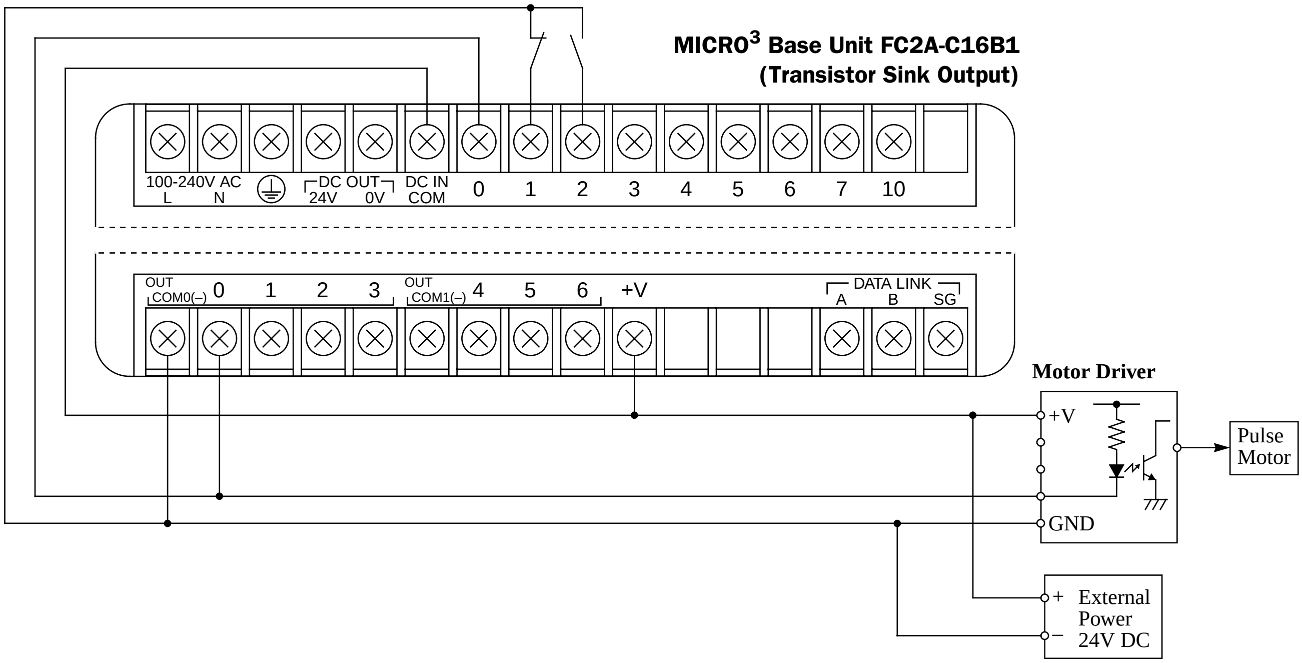
While hard reset input I1 is on, HSC2 counts input pulses to input I0. When the HSC2 current value reaches 1000, the pulse output at Q0 is stopped, and output Q1 is turned on.

When hard reset input I1 is turned off or soft reset input I2 is turned on, HSC2 is reset.

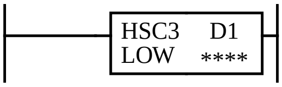
When input I2 is turned off, the PULS instruction turns off the pulse output at Q0.

For monitoring high-speed counter preset and current values, see page 3-15.

I/O Wiring Diagram



A4 HSC3 (Gate Control)



Gate controlled high-speed counter HSC3 counts input pulses to input I0 while gate control input I2 is on. When gate control input I2 is turned off, the current value is moved to a data register designated by destination operand D1. HSC3 does not compare the current value with a preset value.

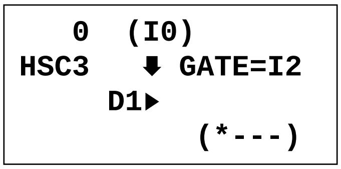
HSC3 can count up to 65535. When another input pulse enters at 65535, the current value becomes 0, and special internal relay M316 (HSC3 overflow) is turned on.

The high-speed counter current value is reset to 0 when MICRO^3 is powered up. The high-speed counter holds the current value while MICRO^3 is stopped and restarts counting input pulses starting with the existing current value. Include the hard reset or soft reset in the user program, if necessary.

Note: Only one of HSC0 through HSC3 and A/D instructions can be used only once in a user program.

Key Operation

ADV A/NOT 4



```
   0  (I0)
HSC3   ⬇ GATE=I2
     D1▶
          (*---)
```

Enter operand D1.

To select hard reset mode from LOW, HIGH, or unused, press the REP key.

To exit, press the ↵ key.

Valid Operands (Standard Processing)

Operand	Function	I	Q	M	T	C	R	D	Constant	Repeat
D1 (Destination 1)	Store the current value	—	—	—	—	—	—	0-99	—	—

In the high-speed processing mode, data registers for this instruction are limited to D0 through D31.

Allocation Numbers

The HSC3 instruction uses the following input and internal relay numbers:

Pulse input:	Input I0
Hard reset input:	Input I1
Gate input:	Input I2
Soft reset special internal relay:	Internal relay M315 (When M315 is on, the current value is reset to 0.)
HSC3 overflow special internal relay:	Internal relay M316

Hard Reset Selection

Input I1 can be used to reset the current value of high-speed counter HSC3.

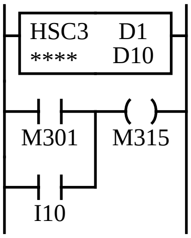
`LOW:` Resets the current value when input I1 is turned off. HSC3 is enabled while I1 is on.
`HIGH:` Resets the current value when input I1 is turned on. HSC3 is enabled while I1 is off.
`*---:` Disables hard reset. (Input I1 can be used as an ordinary input.)

Gate Input

Input I2 is allocated as a gate input for the HSC3 instruction. When I2 is on, HSC3 is enabled to count input pulses to input I0. When I2 is turned off, HSC3 is disabled, and the current value is moved to a data register designated by destination operand D1. When I2 is turned on again, HSC3 continues counting from the existing current value.

Soft Reset Special Internal Relay M315

In addition to the hard reset using input I1, the high-speed counter current value can be reset by turning special internal relay M315 on using another input, output, or internal relay. M315, if used, must be programmed immediately after the HSC3 instruction as shown on the right.



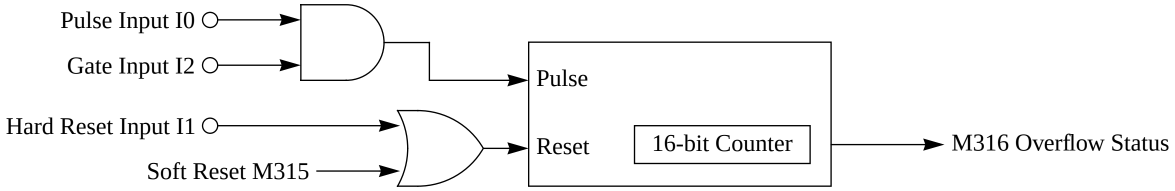
M301 is the initialize pulse special internal relay.

Input Filter and Input Frequency

MICRO^3 has hard and soft filter functions. Only the hard filter works on high-speed counter instructions. The hard filter settings affect the input response. See page 4-3. The maximum input frequency for the HSC3 instruction is 10 kHz.

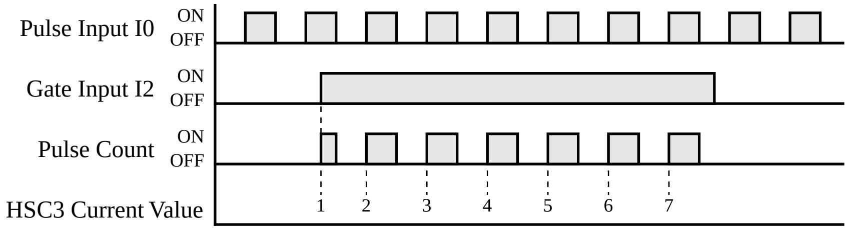
Block Diagram (HSC3: Gate Control)

HSC3 counts input pulses to input I0 while gate input I2 is on and stops counting when I2 is off.

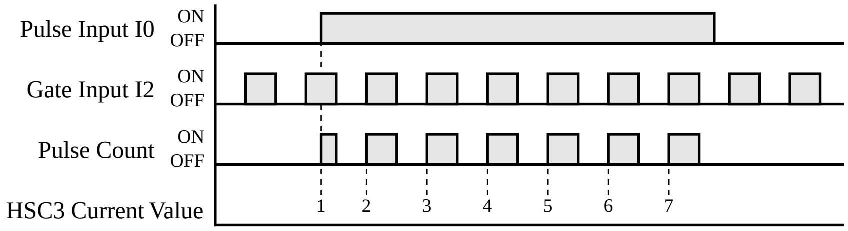


Gate Input (HSC3: Gate Control)

As shown in the figure above, pulse input I0 and gate input I2 are connected in an AND circuit. If gate input I2 is turned on and off while pulse input I0 is on, the gate pulses are counted as shown below.

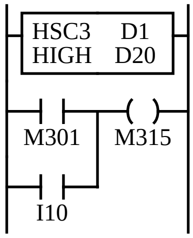


While gate input I2 is on, HSC3 counts ON pulses to input I0.



While pulse input I0 is on, HSC3 counts ON pulses to input I2.

Example: HSC3



I0: Pulse input
I1: Hard reset input (HSC3 is reset when I1 is on because the hard reset is set to HIGH.)
I2: Gate input
I10: Soft reset input to turn soft reset special internal relay M315 on

M301 is the initialize pulse special internal relay used to turn soft reset special internal relay M315 on at start up.

While hard reset input I1 is off and gate input I2 is on, the HSC3 instruction counts input pulses to input I0. When gate input I2 is turned off, HSC3 stops counting, and the current value is moved to data register D20. When gate input I2 is turned on again, HSC3 continues counting from the existing current value.

When hard reset input I1 or soft reset input I10 is turned on, the HSC3 current value is reset to 0.

When the current value exceeds 65535, HSC3 overflow special internal relay M316 is turned on to signal an overflow.

For monitoring high-speed counter preset and current values, see page 3-15.

Introduction

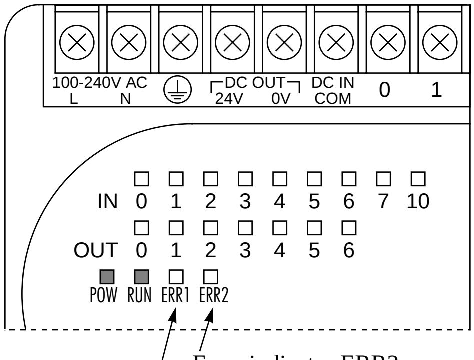
This chapter describes the procedures to determine the cause of trouble and actions to be taken when any trouble occurs during operating the MICRO3 programmable controller.

MICRO3 has self-diagnostic functions to prevent the spread of troubles if any trouble should occur. In case of any trouble, follow the troubleshooting procedures to determine the cause and to correct the error.

Error Indicators ERR1 and ERR2

The MICRO3 base unit has two error indicators: ERR1 and ERR2. When an error occurs in the MICRO3 base unit, error indicator ERR1 is lit. In addition, when the sensor power supply from the MICRO3 base unit is overloaded, error indicator ERR2 is lit. See the trouble shooting diagrams on pages 18-9 and 18-10.

For error causes to turn ERR1 and ERR2 on, see page 18-2.



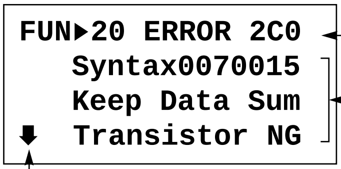
Error indicator ERR2

Error indicator ERR1

Reading Error Data

When any error occurs during the MICRO3 operation, the error codes and messages can be read out using FUN20 on the program loader.

FUN 2 BRD 0 ▼



```
FUN▶20 ERROR 2C0
   Syntax0070015
   Keep Data Sum
⬇  Transistor NG
```

Total of General Error Codes

Error Messages

⬇: Next page exists. To view the next page, press the ▼ key.

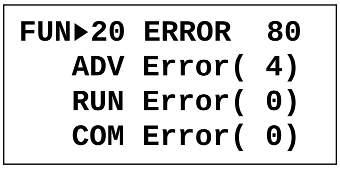
⬆: Preceding page exists. To view the preceding page, press the ▲ key.

Error Messages

When operating FUN20, the following error messages may be displayed.

Error Message	Error Code	Error Status
Power Off	1h	Power failure
Watch Dog Time	2h	Watch dog timer error
PC Connect NG	4h	Data link connection error
Users Prg. CRC	8h	User program CRC error
TIM/CNT CRC	10h	Timer/counter preset value CRC error
Prg. Sum Check	20h	User program sum check error
Keep Data Sum	40h	Keep data sum check error
Syntax	80h	User program syntax error
EEPROM NG	100h	User program writing error
Transistor NG	200h	Protect output overload error
24V Overload	400h	Sensor power overload error
Calendar NG	800h	Calendar/clock error
ADV Error ( )	——	Advanced instruction syntax error
RUN Error ( )	——	User program execution error
COM Error ( )	——	Link communication error

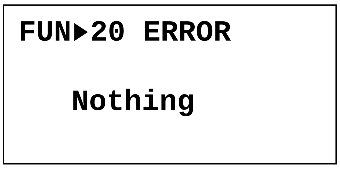
ADV Error, RUN Error, and COM Error are displayed on the same page.



```
FUN▶20 ERROR  80
   ADV Error( 4)
   RUN Error( 0)
   COM Error( 0)
```

After removing cause of the error, clear error code using the program loader.

DEL ↵



```
FUN▶20 ERROR

   Nothing
```

The error indicator remains on until the cause of the error is removed.

Note: When the error code is cleared using the program loader as described above, the error data on the user program syntax error (Syntax), advanced instruction syntax error (ADV Error), user program execution error (RUN Error), and link communication error (COM Error) is cleared from memory. To read the error code again before removing the cause of the error, power up the MICRO3 or transfer the user program to the MICRO3 again, then operate FUN20.

General Error Codes

When reading error data using FUN20, the error code is displayed using one, two, or three digits in hexadecimal notation (0 through F). Each digit of the error code indicates a different set of conditions requiring attention. If there are none of the conditions from the first chart (digit on the left), then this digit does not display on the program loader. If there are none from the first and second charts, then these two digits are not displayed.

For example, the error code may read out “21.” This indicates two conditions requiring attention, “User program sum check error” from the second chart and “Power failure” from the third chart. If the read-out displays “D,” this indicates three conditions exist from only the chart on the right.

When control data register D92 is enabled using FUN10, general error code is stored in D92. See page 5-8.

Error Code Display: Digit on the Left	Error Items			
	Calendar & clock error	Sensor power overload error	Protect output overload error	User program writing error
No display (No error)				
100h				●
200h			●	
300h			●	●
400h		●		
500h		●		●
600h		●	●	
700h		●	●	●
800h	●			
900h	●			●
A00h	●		●	
B00h	●		●	●
C00h	●	●		
D00h	●	●		●
E00h	●	●	●	
F00h	●	●	●	●

Error Code Display: Digit in the Middle	Error Items			
	User program syntax error	Keep data sum check error	User program sum check error	TIM/CNT preset value CRC error
No error 00h				
10h				●
20h			●	
30h			●	●
40h		●		
50h		●		●
60h		●	●	
70h		●	●	●
80h	●			
90h	●			●
A0h	●		●	
B0h	●		●	●
C0h	●	●		
D0h	●	●		●
E0h	●	●	●	
F0h	●	●	●	●

Error Code Display: Digit on the Right	Error Items			
	User program CRC error	Data link connec-tion error	Watch dog timer error	Power failure
No error 0h				
1h				●
2h			●	
3h			●	●
4h		●		
5h		●		●
6h		●	●	
7h		●	●	●
8h	●			
9h	●			●
Ah	●		●	
Bh	●		●	●
Ch	●	●		
Dh	●	●		●
Eh	●	●	●	
Fh	●	●	●	●

MICRO^3 Operating Status, Output, and Error Indicator during Errors

Error Items	Operating Status	Output	Error Indicator	Checked at
Power failure	Stop	OFF	ERR1 ON *1	Any time
Watchdog timer error	Stop	OFF	ERR1 ON	Any time
Data link connection error	Stop	OFF	ERR1 ON	Initializing data link
User program CRC error	Stop	OFF	ERR1 ON	Starting operation
TIM/CNT preset value CRC error	Maintained	Maintained	OFF	Starting operation
User program sum check error	Stop *2	OFF	ERR1 ON	During operation
Keep data sum check error	Maintained	Maintained	OFF	Turning power on
User program syntax error	Stop	OFF	ERR1 ON	Writing user program
User program writing error	Stop	OFF	ERR1 ON	Writing user program
Protect output overload error	Maintained	Maintained *3	ERR1 ON	During operation
Sensor power overload error	Stop	OFF	ERR1 & ERR2 ON	Any time
Calendar/clock error	Maintained	Maintained	ERR1 ON	Any time
Advanced instruction syntax error	Stop	OFF	ERR1 ON	Writing user program
User program execution error	Maintained	Maintained	ERR1 ON	During operation
Link communication error	Maintained	Maintained	OFF	Any time

*1: When the power voltage to the MICRO^3 base unit drops below the rated value, the ERR1 indicator is lit. While the power voltage remains below the rated value, the ERR1 indicator does not go on.

*2: When a program sum check error occurs, operation is stopped momentarily for recompiling the user program. After completing the recompilation, operation resumes.

*3: Outputs where error occurs are turned off.

Error Causes and Actions

1h: Power Failure (Power Off)

This error indicates when the power supply is lower than the specified voltage. This error is also recorded when the power is turned off. Clear the error code using FUN20 on the program loader.

2h: Watchdog Timer Error (Watch Dog Time)

The watchdog timer monitors the time required for one program cycle (scan time). When the time exceeds approximately 300 msec, the watchdog timer indicates an error. If this error occurs frequently, the MICRO^3 base unit has to be replaced. Clear the error code using FUN20 on the program loader.

4h: Data Link Connection Error (PC Connect NG)

This error indicates that both data link slave stations (function selector switch set to 1 through 6) and an expansion station (function selector switch set to 7) are connected to the data link master station (function selector switch set to 0). Make sure that the function selector switches at all slave stations are set to 1 through 6 in the data link system. The expansion link function cannot be used in the data link system.

To correct this error, change the function selector switch setting to 1 through 6 on slave station units. Turn power off and on again for the slave station unit. Then take one of the following method:

- Turn power off and on for the master station unit.
- Execute the link formatting sequence (FUN27) for the master station using the program loader. See page 5-11.
- Turn M307 on at the master station during operation to initialize the link communication. See page 6-3.

8h: User Program CRC Error (Users Prg. CRC)

The user program stored in the MICRO^3 base unit EERPOM is broken. Transfer a correct user program to MICRO^3, and clear the error code using FUN20 on the program loader.

10h: Timer/Counter Preset Value CRC Error (TIM/CNT CRC)

The execution data of timer/counter preset values is broken. Clear the error code using FUN20 on the program loader; then the timer/counter preset values are initialized to the values of the user program. Note that modified preset values are cleared and that the original values are restored when the error code is cleared.

20h: User Program Sum Check Error (Prg. Sum Check)

The data of the user program compile area in the MICRO^3 base unit RAM is broken. When this error occurs, the user program is recompiled automatically. Clear the error code using FUN20 on the program loader; then the timer/counter preset values are initialized to the values of the user program. Note that modified preset values are cleared and that the original values are restored when the error code is cleared.

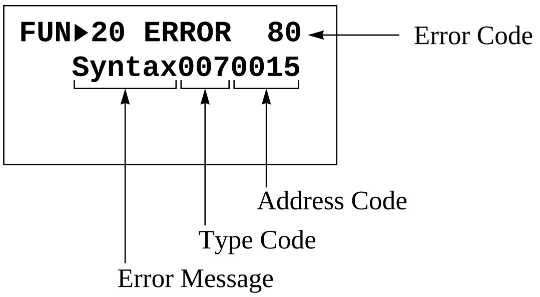
40h: Keep Data Sum Check Error (Keep Data Sum)

This error indicates that the data designated to be maintained during power failure is broken because of memory backup failure. Clear the error code using FUN20 on the program loader. Note that the “keep” data of internal relays and shift registers are cleared when the error code is cleared.

80h: User Program Syntax Error (Syntax)

This error indicates that the user program has a syntax error or that FUN1 through FUN10 is set incorrectly. Correct the user program or FUN settings, and transfer the corrected user program to MICRO^3. The error code is cleared when a correct user program is transferred.

When this error occurs, the error message is displayed with a type code and an address code of 7 digits total.



```
FUN▶20 ERROR  80
   Syntax0070015
```

For details of the type code and address code, see the next page.

User Program Syntax Error Type Code and Address Code

Type Code	Address Code	Error Details
1	0001	Stop input number selection error (FUN1)
	0002	Reset input number selection error (FUN2)
	0003	Internal relay “keep” designation error (FUN3)
	0004	Shift register “keep” designation error (FUN4)
	0005	Processing mode selection error (FUN5)
	0006	Catch input edge selection error (FUN6)
	0007	Input filter time selection error (FUN7)
	0008	Loader port communication mode setting error (FUN8)
	0009	PLC address error for network communication (FUN9)
	0010	Control data register setting error (FUN10)
2	0000 to 1012 Address of the incorrect program	Invalid opcode for basic instruction
3		Invalid operand for basic instruction
4		Invalid timer/counter preset value
5 (Note)		Invalid opcode for advanced instruction *TXD/RXD programmed in the high-speed processing mode*
6 (Note)		Invalid data for advanced instruction *Same data register designated as status DR for TXD and RXD*
7 (Note)		Invalid repeated usage of advanced instruction
8		User program capacity over error

Note: When type code 5, 6, or 7 is displayed, the details are shown by the error code of the ADV Error (advanced instruction syntax error). See the next page.

100h: User Program Writing Error (EEPROM NG)

This error indicates a failure of writing into the MICRO^3 base unit EEPROM when transferring a user program or when setting user program protection. The error code is cleared when writing into the EEPROM is completed correctly. If this error occurs frequently, the MICRO^3 base unit has to be replaced.

200h: Protect Output Overload Error (Transistor NG)

This error is issued when a protect transistor output is overloaded during operation, then only the overloaded output is forced off. When this error occurs at the base station in the expansion link system, error indicator ERR1 at the base station is lit. When the error is at the expansion station, error indicator ERR1 is lit at both the base and expansion stations.

If this error has occurred at output Q0 or Q20, then remove the cause of the overload, turn the output off (MON, Q, output number, RST, ↵), or turn the output power off. Clear the error code using FUN20 on the program loader.

If this error has occurred at other outputs, then remove the cause of the overload, and the output restores normal operation automatically. Clear the error code using FUN20 on the program loader.

400h: Sensor Power Overload Error (24V Overload)

This error indicates that the sensor power supply from the MICRO^3 base unit is overloaded. When this error occurs, both error indicators ERR1 and ERR2 are lit.

When this error occurs at the base station in the expansion link system, error indicators ERR1 and ERR2 are lit at the base station. When the error is at the expansion station, ERR1 and ERR2 are lit at both the base and expansion stations.

To correct this error, reduce the sensor power output load within the rated value, and either turn power to MICRO^3 off and on or clear the error code using FUN20 on the program loader.

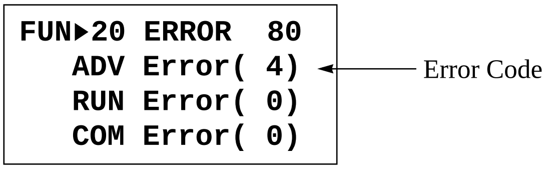
800h: Calendar/Clock Error (Calendar NG)

This error indicates that the real time calendar/clock in the MICRO^3 base unit has an error caused by invalid clock data due to voltage drop or by erroneous quartz oscillator operation.

Clear the error code using FUN20, and set the calendar/clock data using FUN28 on the program loader. Turn power off and on again. If the error continues, the MICRO^3 base unit has to be replaced. See Troubleshooting Diagram 14 on page 18-20.

Advanced Instruction Syntax Error (ADV Error)

When a user program syntax error (error code 80h—error related with advanced instruction) is indicated with type code 5, 6, or 7, the detailed information can be viewed from the error code indicated in the ADV Error line.



```
FUN▶20 ERROR  80
   ADV Error( 4)
   RUN Error( 0)
   COM Error( 0)
```

Error Code	Error Details
1	The internal allocation number of the operand is invalid.
2	Input or special internal relay is designated as a destination.
3	Quantity of repeat cycles is set to 32 (or shift bits to 16 for shift/rotate instructions) or more.
4	Advanced instruction which is not allowed for repeat usage is programmed more than once.

Correct the error in the user program and transfer the corrected user program to MICRO^3. The error code is cleared when a correct user program is transferred.

User Program Execution Error (RUN Error)

This error indicates that invalid data is found during execution of a user program. When this error occurs, special internal relay M304 (user program execution error) is also turned on. The detailed information of this error can be viewed from the error code indicated in the RUN Error line. This error code is stored in control data register D93, if enabled using FUN10. See page 5-8. When this error occurs, program operation and all output statuses are maintained.

Error Code	Error Details
1	As a result of an advanced instruction, a value exceeding 9999 is written into the timer/counter preset value. Data is written into the preset value of a timer/counter which is not included in the user program or which uses a data register as a preset value.
2	Data register used as a preset value for timer, counter, or counter comparison instruction exceeds 9999.
3	Indirect operand for the IMOV or IMOVN instruction is out of range.
4	Overflow or underflow has resulted from advanced instruction.
5	Division by 0.
6	Invalid data occurred during data conversion for the DISP or DGRD instruction.
7	Attempt was made to write invalid value to calender/clock data.
8	Data register used as an operand for the PULS or PWM instruction contains invalid data.
9	The quantity of multi-stage preset data for high-speed counter HSC1 exceeds available data registers. Invalid numeric allocation number is designated as a comparison output of the HSC1.

Remove the cause of the error, and clear the error code using FUN20 on the program loader. Special internal relay M304 is reset when restarting the MICRO^3 operation; M304 can also be reset using the program loader (MON, M304, RST, ↵).

Link Communication Error (COM Error)

This error indicates a communication error in the expansion link or data link system. When this error occurs, special internal relay M305 (link communication error) is also turned on. The detailed information of this error can be viewed from the error code indicated in the COM Error line. This error code is stored in control data register D94, if enabled using FUN10. See page 5-8. When this error occurs, program operation and all output statuses are maintained.

Error Code	Error Details
1h	Overrun error (Data is received when the receive data registers are full.)
2h	Framing error (Failure to detect start or stop bit.)
4h	Parity error (An error was found by the parity check.)
8h	Receive timeout (Line disconnection)
10h	BCC (block check character) error (Disparity with data received up to BCC.)
20h	Retry cycle over (Error occurred in all 3 trials of communication.)
40h	I/O definition quantity error (Error in the connection to the FA-3S series PF3S-SIF4 serial interface module)

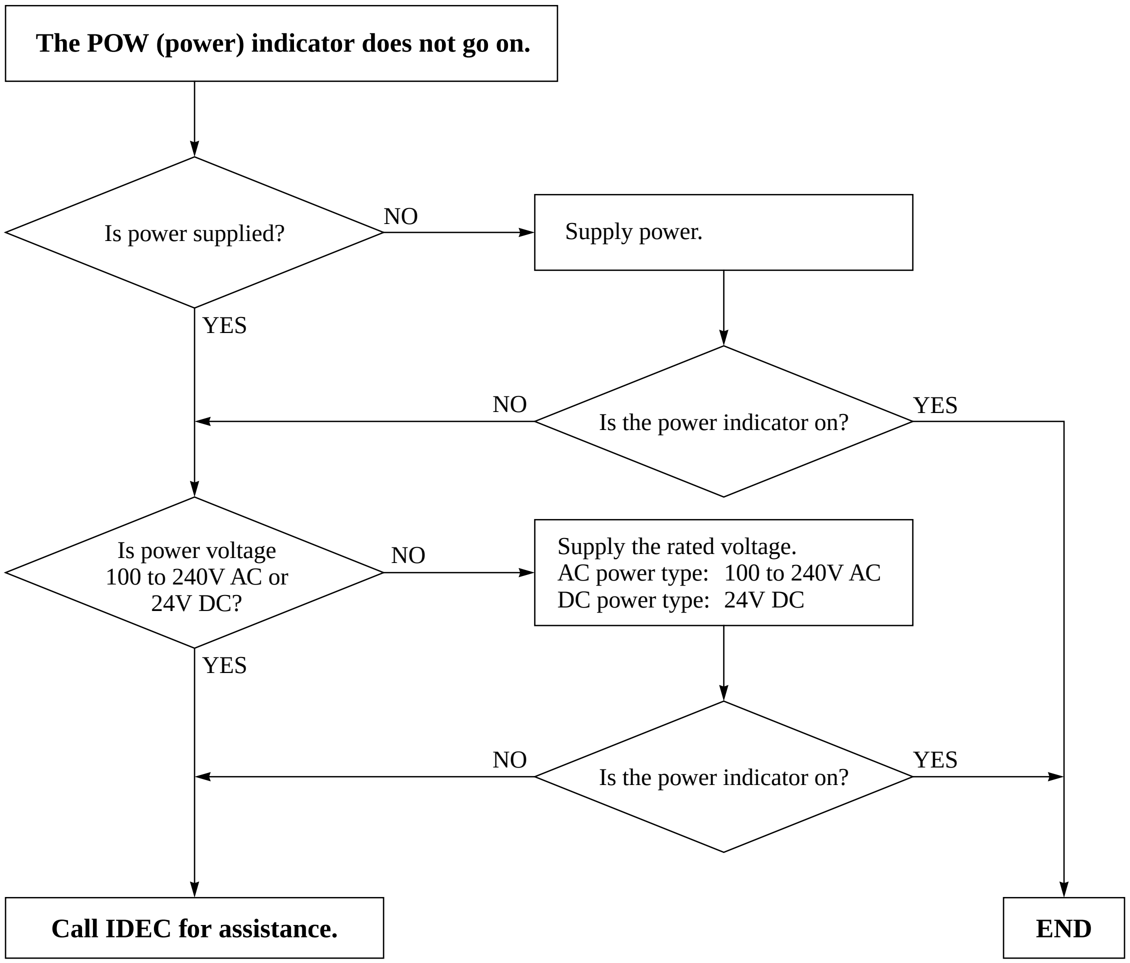
When more than one error is detected in the expansion link or data link system, the total of error codes is indicated. For example, when framing error (error code 2h) and BCC error (error code 10h) are found, error code 12 is displayed.

Troubleshooting Diagrams

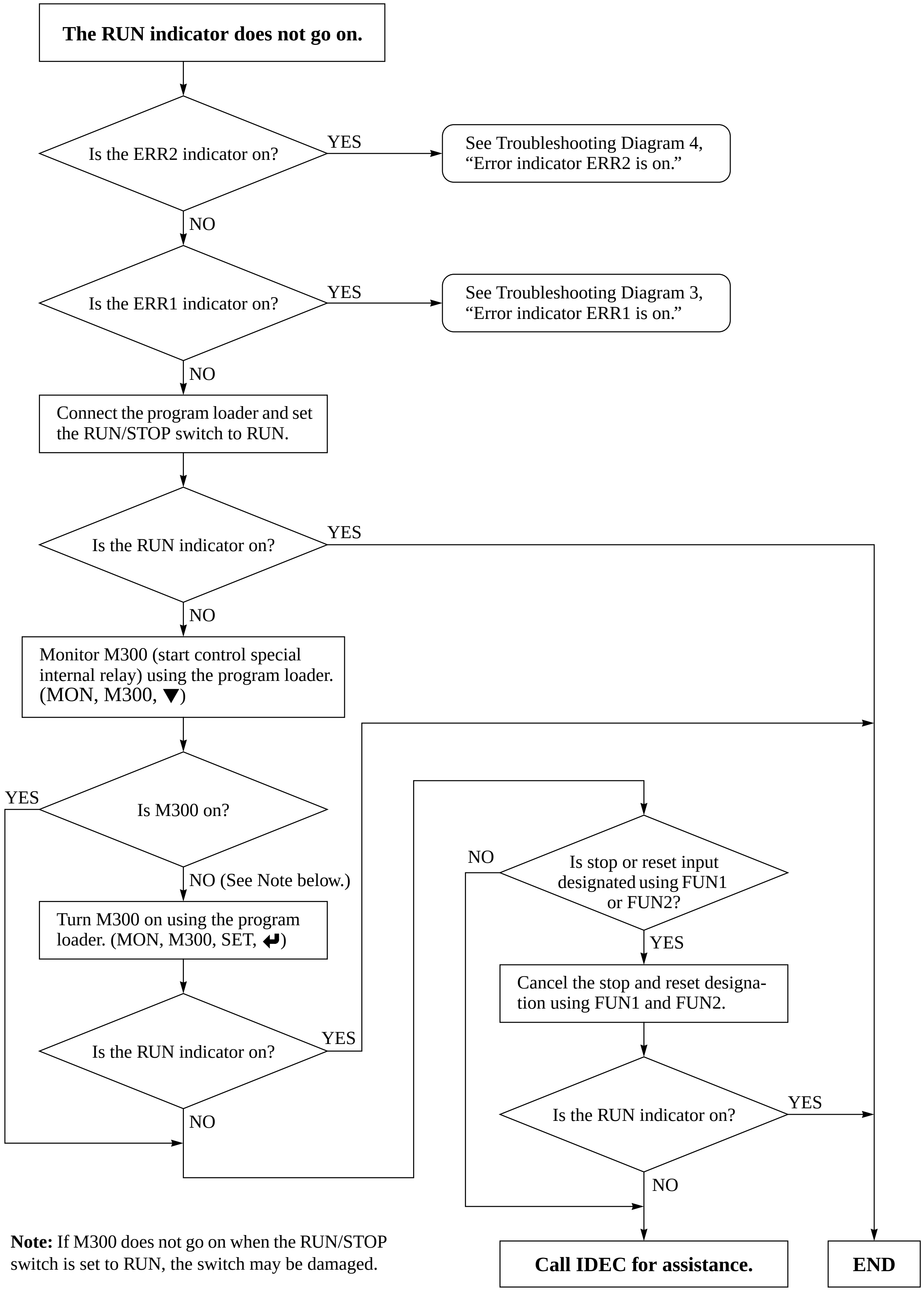
When one of the following problems is encountered, see the troubleshooting diagrams on the following pages. When using MICRO^{3}C, also see the MICRO^{3}C User's Manual for troubles particular to the MICRO^{3}C.

Problem	Troubleshooting Diagram
The POW (power) indicator does not go on.	Diagram 1
The RUN indicator does not go on.	Diagram 2
Error indicator ERR1 is on.	Diagram 3
Error indicator ERR2 is on.	Diagram 4
Inputs do not operate normally.	Diagram 5
Outputs do not operate normally.	Diagram 6
Communication between the program loader and the MICRO^3 base unit is not possible.	Diagram 7
Stop and reset operation cannot be performed.	Diagram 8
Normal voltage does not appear on sensor power terminals.	Diagram 9
Expansion link or data link is impossible.	Diagram 10
Output pulses are not generated at output Q0 when using the PULS or PWM instruction.	Diagram 11
High-speed counter does not work correctly.	Diagram 12
The catch input function cannot receive short pulses.	Diagram 13
The calendar/clock does not operate correctly.	Diagram 14
Transfer to and from the memory card is impossible.	Diagram 15

Troubleshooting Diagram 1

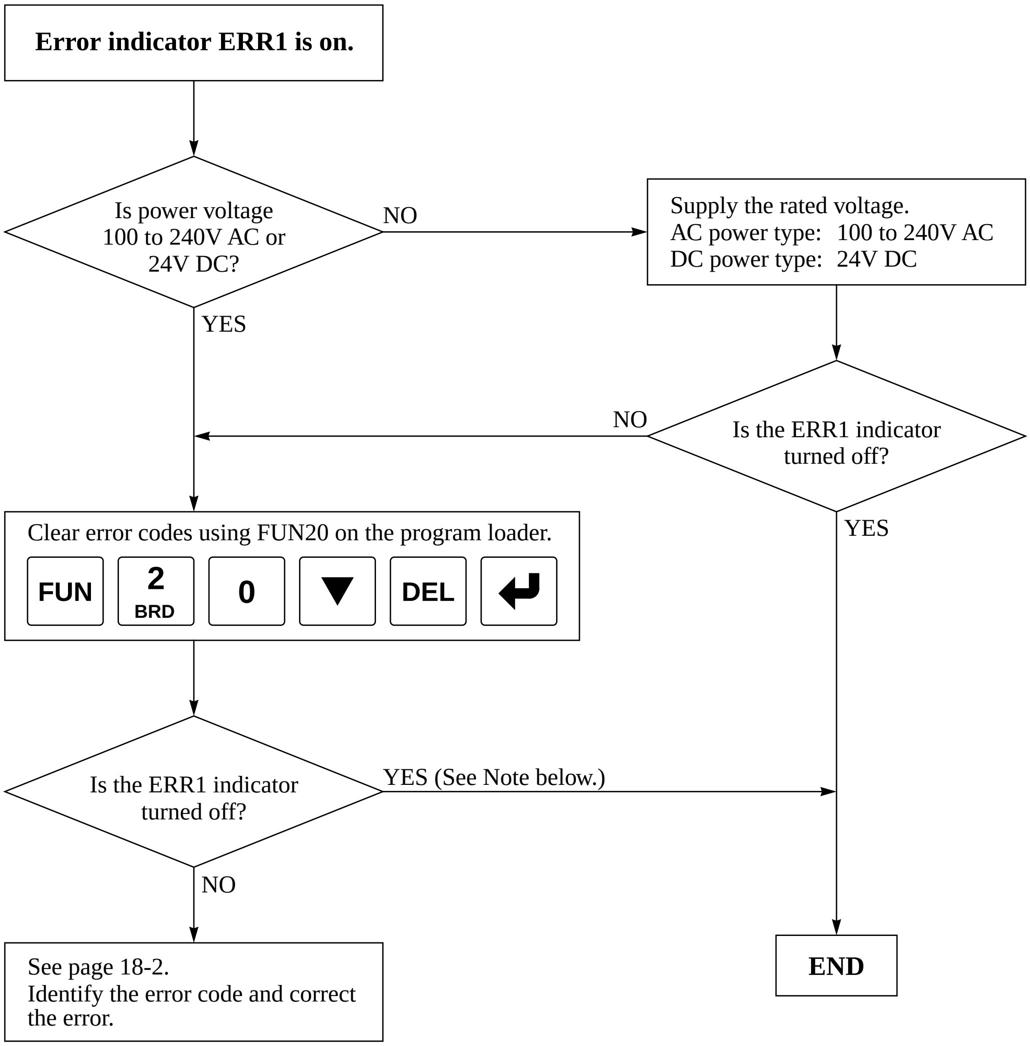

The POW (power) indicator does not go on.
Is power supplied?
NO
Supply power.
YES
NO
Is the power indicator on?
YES
Is power voltage 100 to 240V AC or 24V DC?
NO
Supply the rated voltage.
AC power type: 100 to 240V AC
DC power type: 24V DC
YES
NO
Is the power indicator on?
YES
Call IDEC for assistance.
END

Troubleshooting Diagram 2


The RUN indicator does not go on.
Is the ERR2 indicator on?
YES
See Troubleshooting Diagram 4, "Error indicator ERR2 is on."
NO
Is the ERR1 indicator on?
YES
See Troubleshooting Diagram 3, "Error indicator ERR1 is on."
NO
Connect the program loader and set the RUN/STOP switch to RUN.
Is the RUN indicator on?
YES
NO
Monitor M300 (start control special internal relay) using the program loader. (MON, M300, ▼)
YES
Is M300 on?
NO (See Note below.)
Turn M300 on using the program loader. (MON, M300, SET, ↵)
Is the RUN indicator on?
YES
NO
NO
Is stop or reset input designated using FUN1 or FUN2?
YES
Cancel the stop and reset designation using FUN1 and FUN2.
Is the RUN indicator on?
YES
NO
Call IDEC for assistance.
END

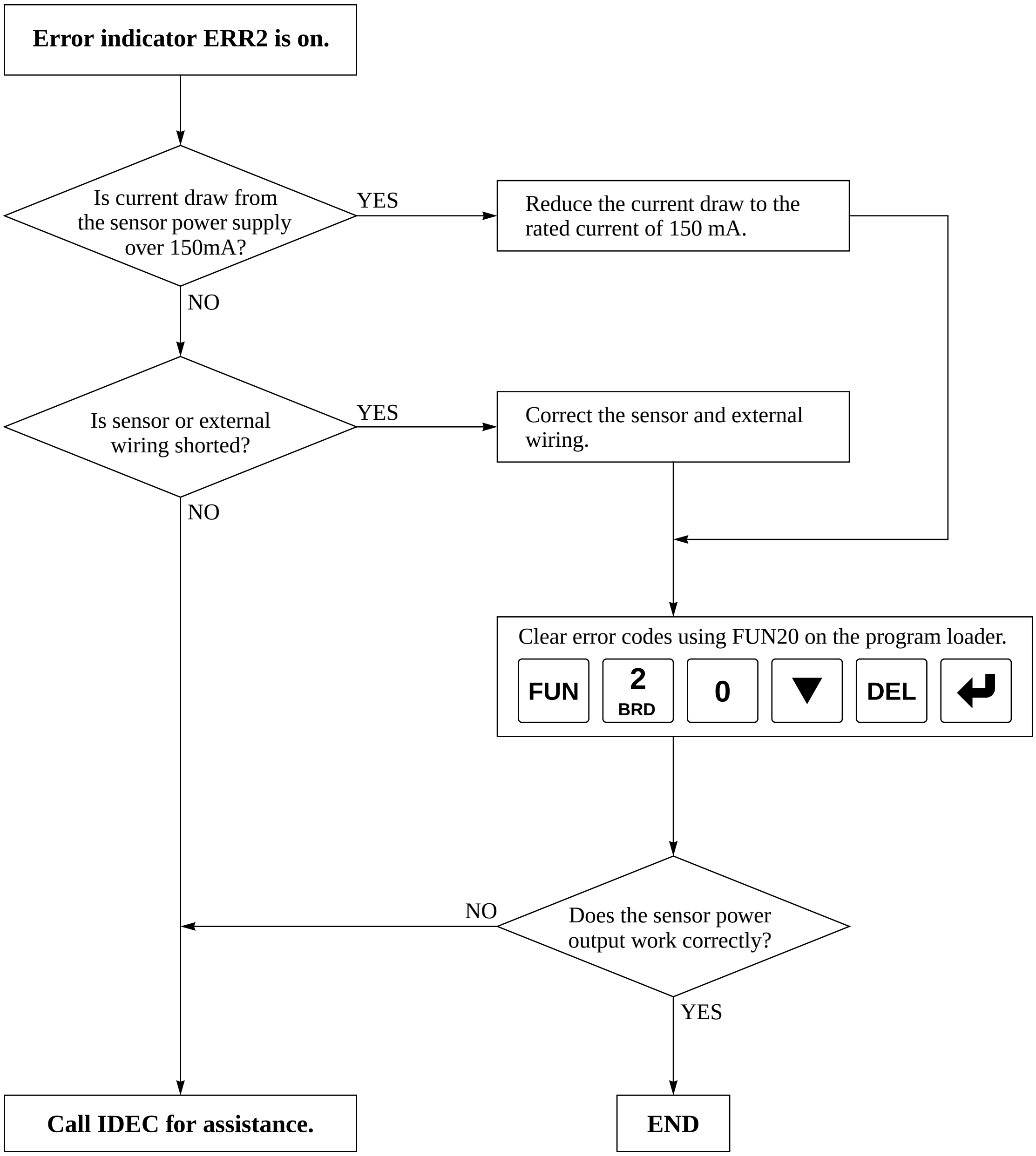
Note: If M300 does not go on when the RUN/STOP switch is set to RUN, the switch may be damaged.

Troubleshooting Diagram 3

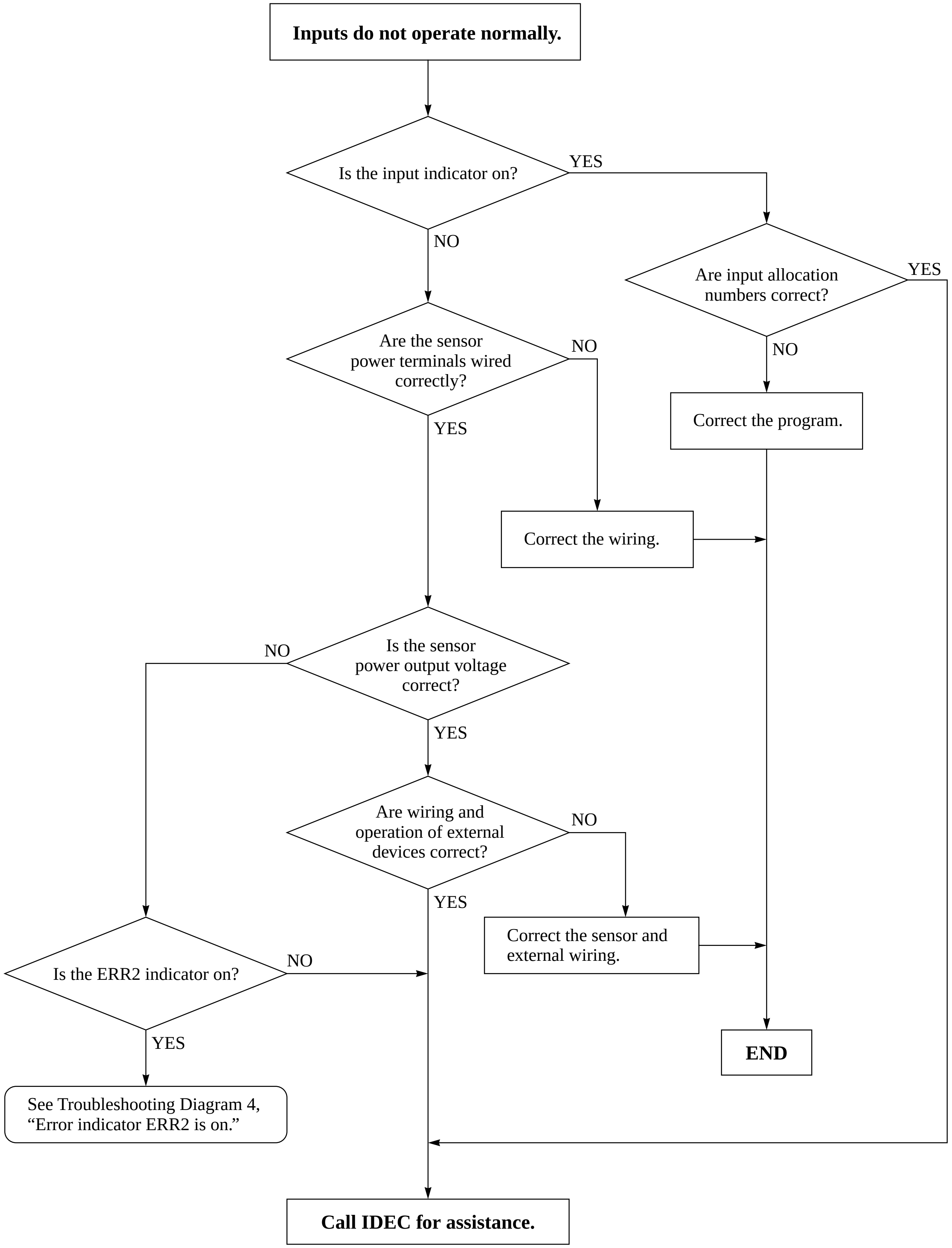

Error indicator ERR1 is on.
Is power voltage 100 to 240V AC or 24V DC?
NO
Supply the rated voltage.
AC power type: 100 to 240V AC
DC power type: 24V DC
YES
NO
Is the ERR1 indicator turned off?
YES
Clear error codes using FUN20 on the program loader.
FUN
2 BRD
0
▼
DEL
↵
Is the ERR1 indicator turned off?
YES (See Note below.)
NO
See page 18-2.
Identify the error code and correct the error.
END

Note: Temporary errors can be cleared to resume normal operation using FUN20.

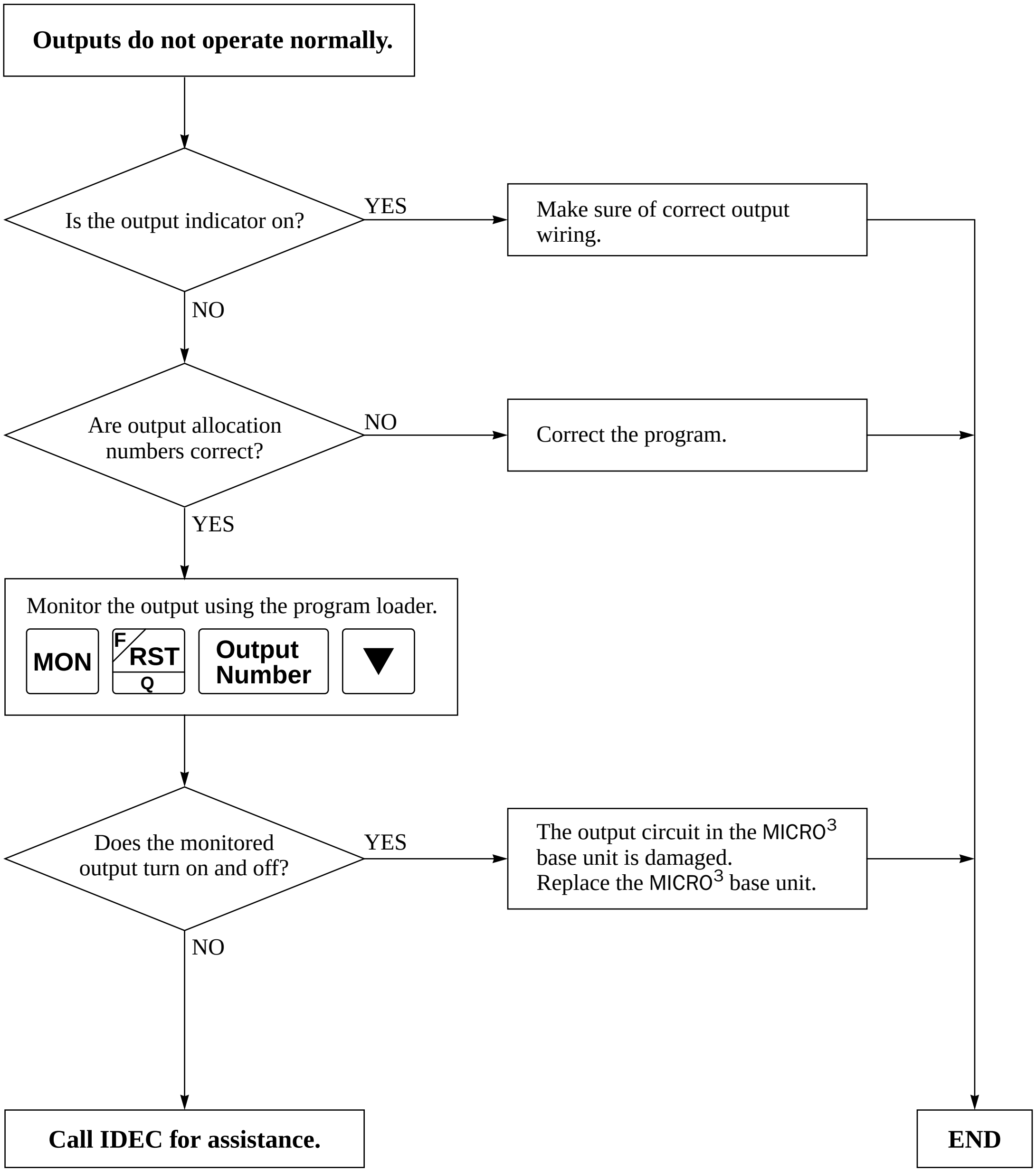
Troubleshooting Diagram 4


Error indicator ERR2 is on.
Is current draw from the sensor power supply over 150mA?
YES
Reduce the current draw to the rated current of 150 mA.
NO
Is sensor or external wiring shorted?
YES
Correct the sensor and external wiring.
NO
Clear error codes using FUN20 on the program loader.
FUN
2
BRD
0
▼
DEL
↵
NO
Does the sensor power output work correctly?
YES
Call IDEC for assistance.
END

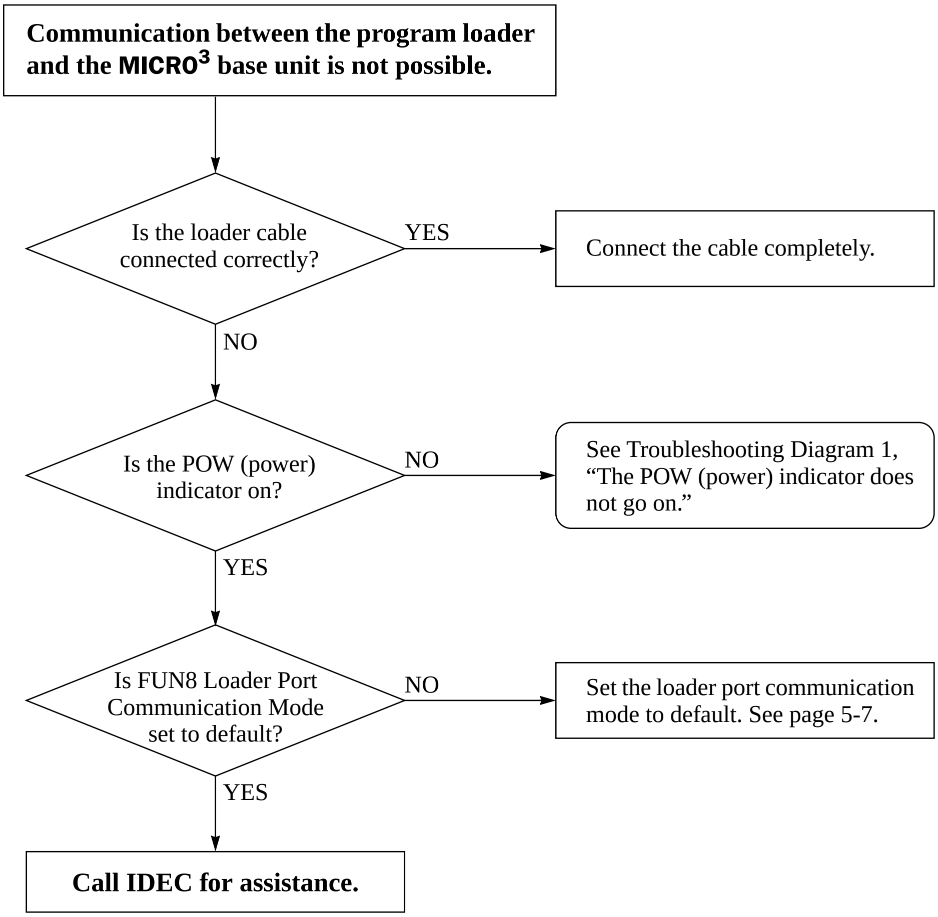
Troubleshooting Diagram 5


Inputs do not operate normally.
Is the input indicator on?
YES
NO
Are input allocation numbers correct?
YES
NO
Correct the program.
Are the sensor power terminals wired correctly?
NO
YES
Correct the wiring.
Is the sensor power output voltage correct?
NO
YES
Are wiring and operation of external devices correct?
NO
YES
Correct the sensor and external wiring.
Is the ERR2 indicator on?
NO
YES
See Troubleshooting Diagram 4, "Error indicator ERR2 is on."
END
Call IDEC for assistance.

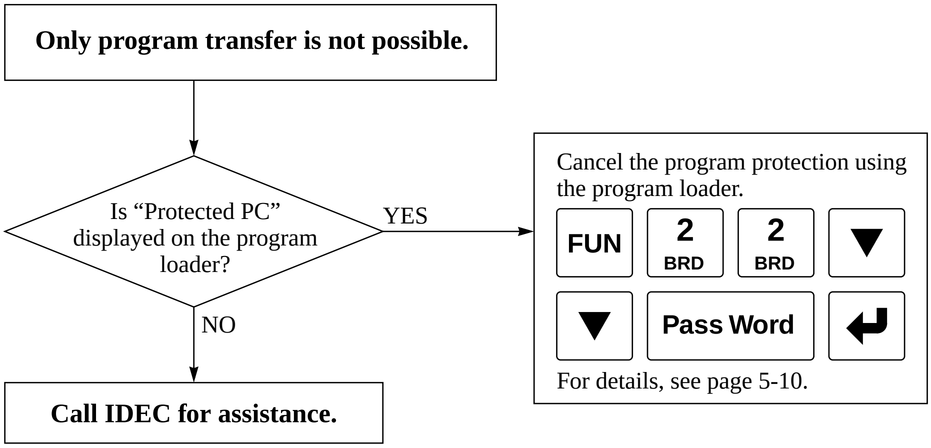
Troubleshooting Diagram 6


Outputs do not operate normally.
Is the output indicator on?
YES
Make sure of correct output wiring.
NO
Are output allocation numbers correct?
NO
Correct the program.
YES
Monitor the output using the program loader.
MON
F RST Q
Output Number
▼
Does the monitored output turn on and off?
YES
The output circuit in the MICRO3 base unit is damaged. Replace the MICRO3 base unit.
NO
Call IDEC for assistance.
END

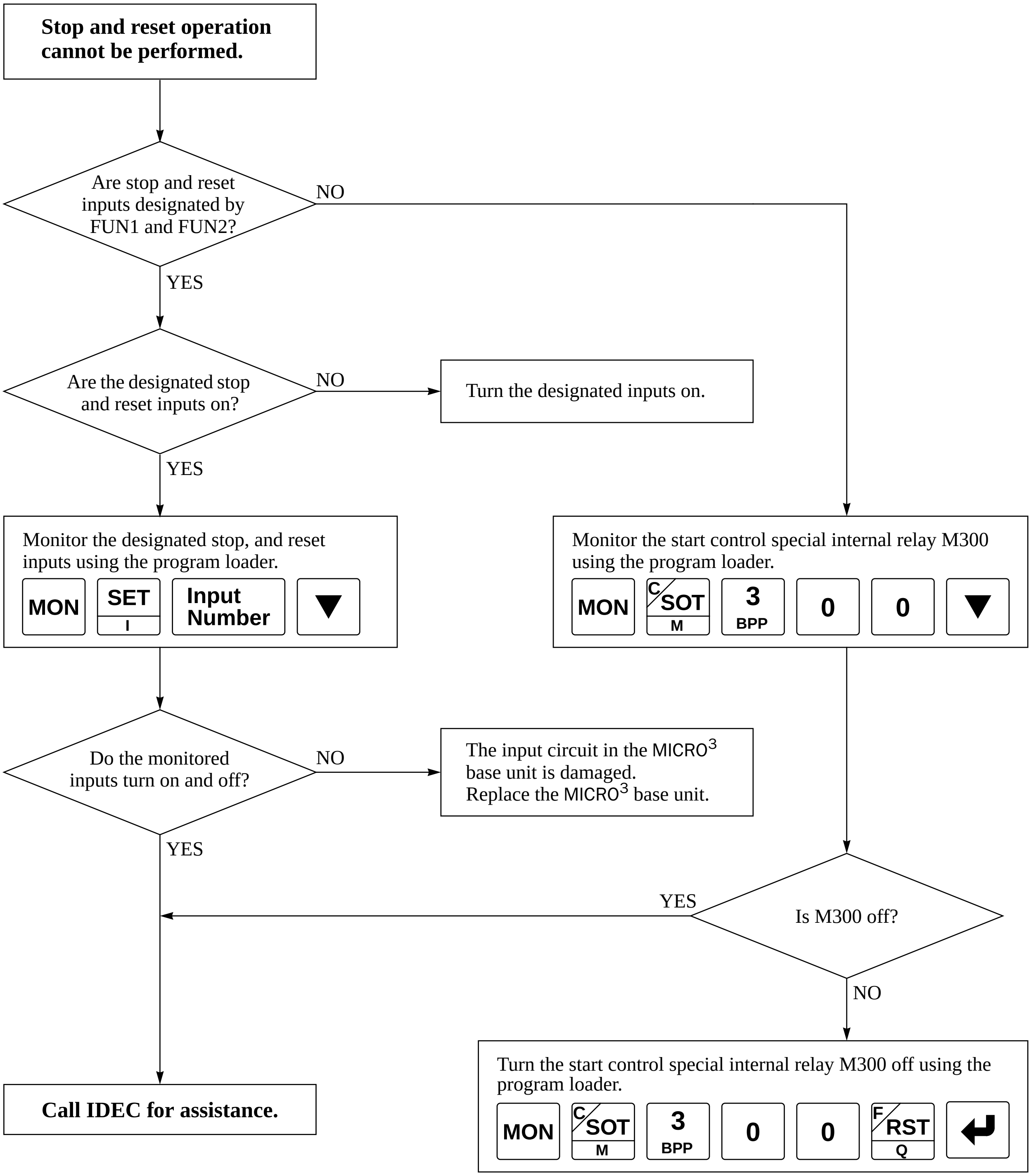
Troubleshooting Diagram 7


Communication between the program loader and the MICRO3 base unit is not possible.
Is the loader cable connected correctly?
YES
Connect the cable completely.
NO
Is the POW (power) indicator on?
NO
See Troubleshooting Diagram 1, "The POW (power) indicator does not go on."
YES
Is FUN8 Loader Port Communication Mode set to default?
NO
Set the loader port communication mode to default. See page 5-7.
YES
Call IDEC for assistance.

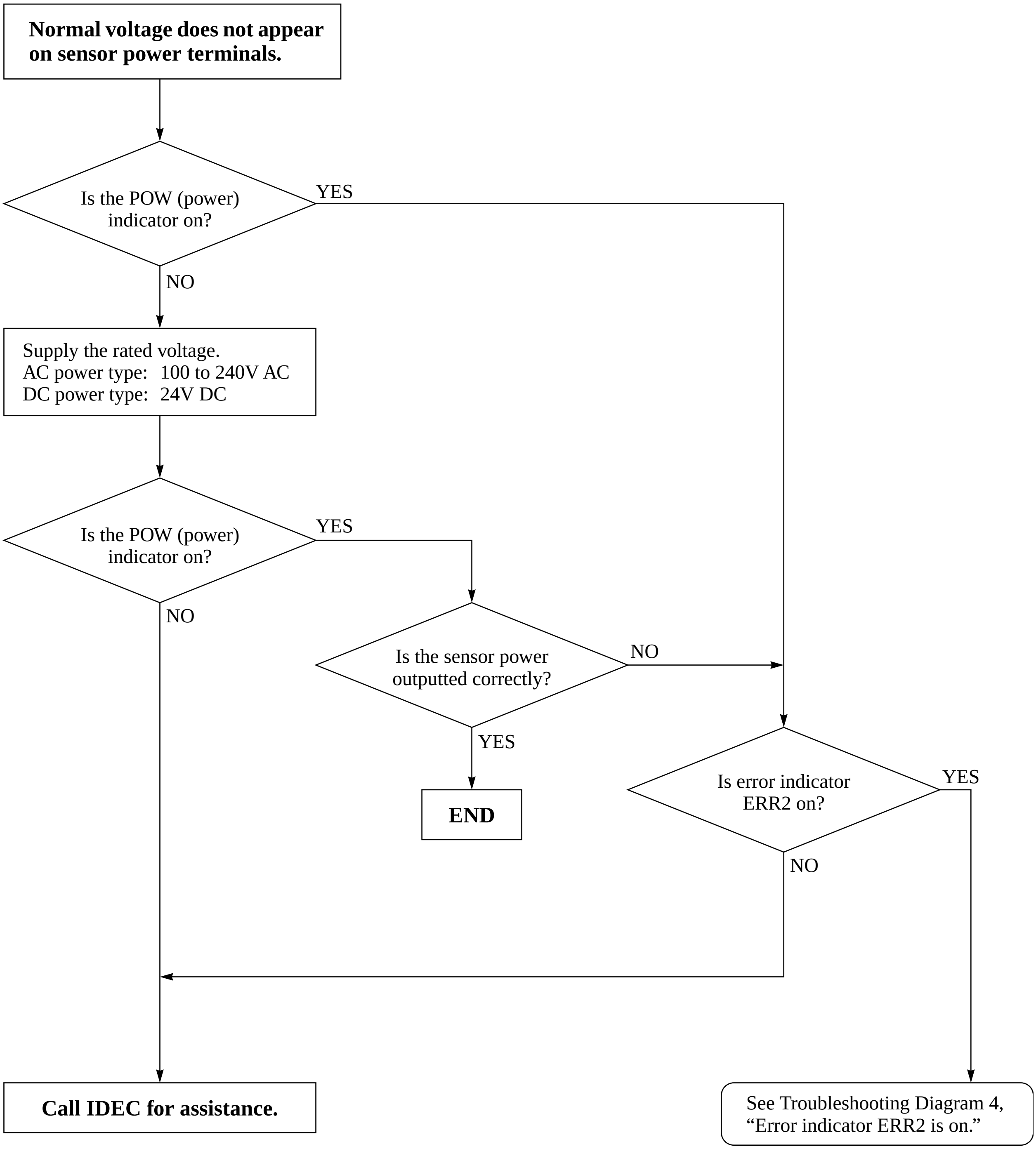
When only program transfer is not possible:


Only program transfer is not possible.
Is "Protected PC" displayed on the program loader?
YES
Cancel the program protection using the program loader.
FUN
2 BRD
2 BRD
▼
▼
Pass Word
↵
For details, see page 5-10.
NO
Call IDEC for assistance.

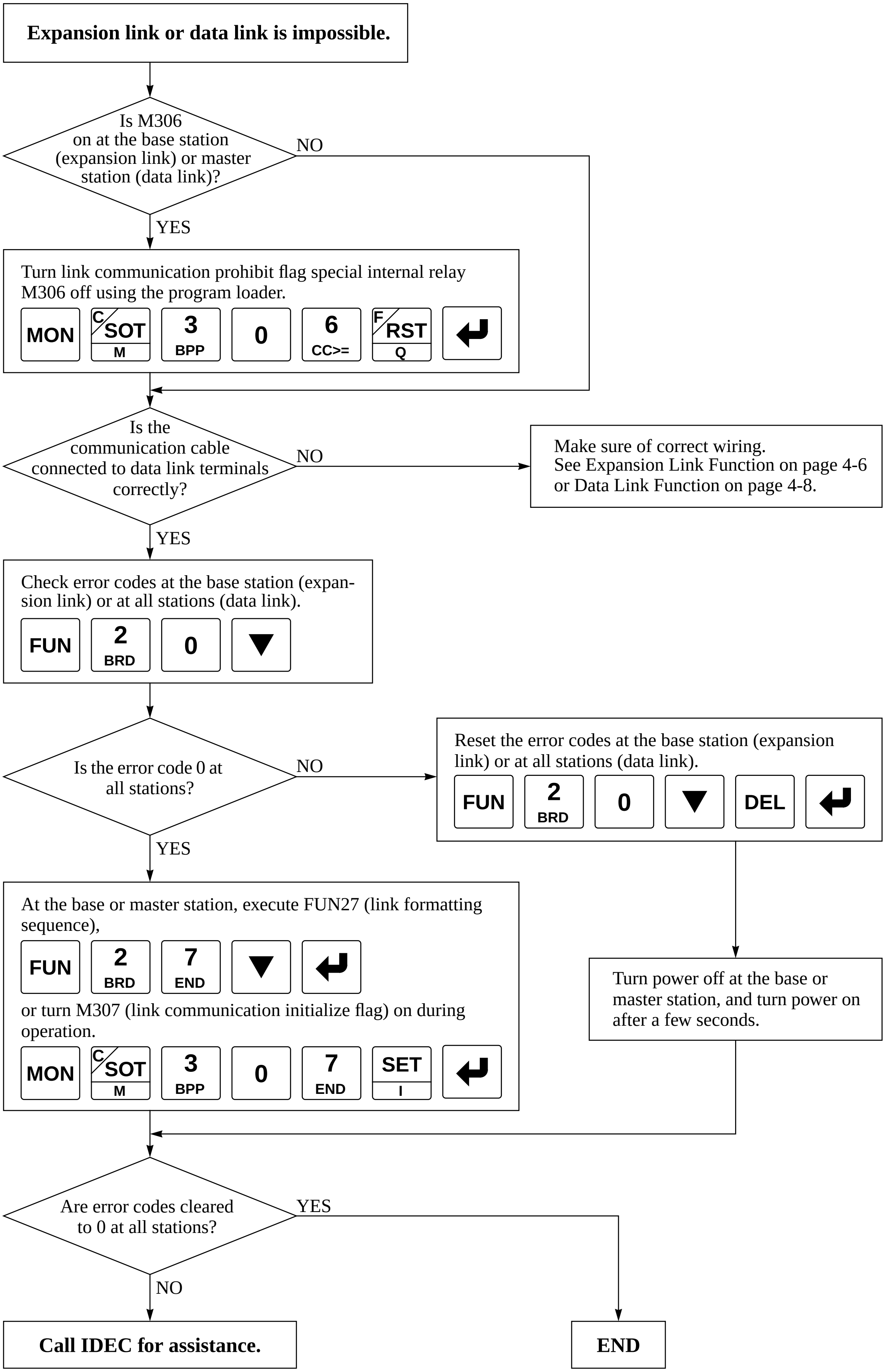
Troubleshooting Diagram 8


Stop and reset operation cannot be performed.
Are stop and reset inputs designated by FUN1 and FUN2?
NO
YES
Are the designated stop and reset inputs on?
NO
Turn the designated inputs on.
YES
Monitor the designated stop, and reset inputs using the program loader.
MON
SET
I
Input Number
▼
Monitor the start control special internal relay M300 using the program loader.
MON
C
SOT
M
3
BPP
0
0
▼
Do the monitored inputs turn on and off?
NO
The input circuit in the MICRO3 base unit is damaged. Replace the MICRO3 base unit.
YES
YES
Is M300 off?
NO
Call IDEC for assistance.
Turn the start control special internal relay M300 off using the program loader.
MON
C
SOT
M
3
BPP
0
0
F
RST
Q
↵

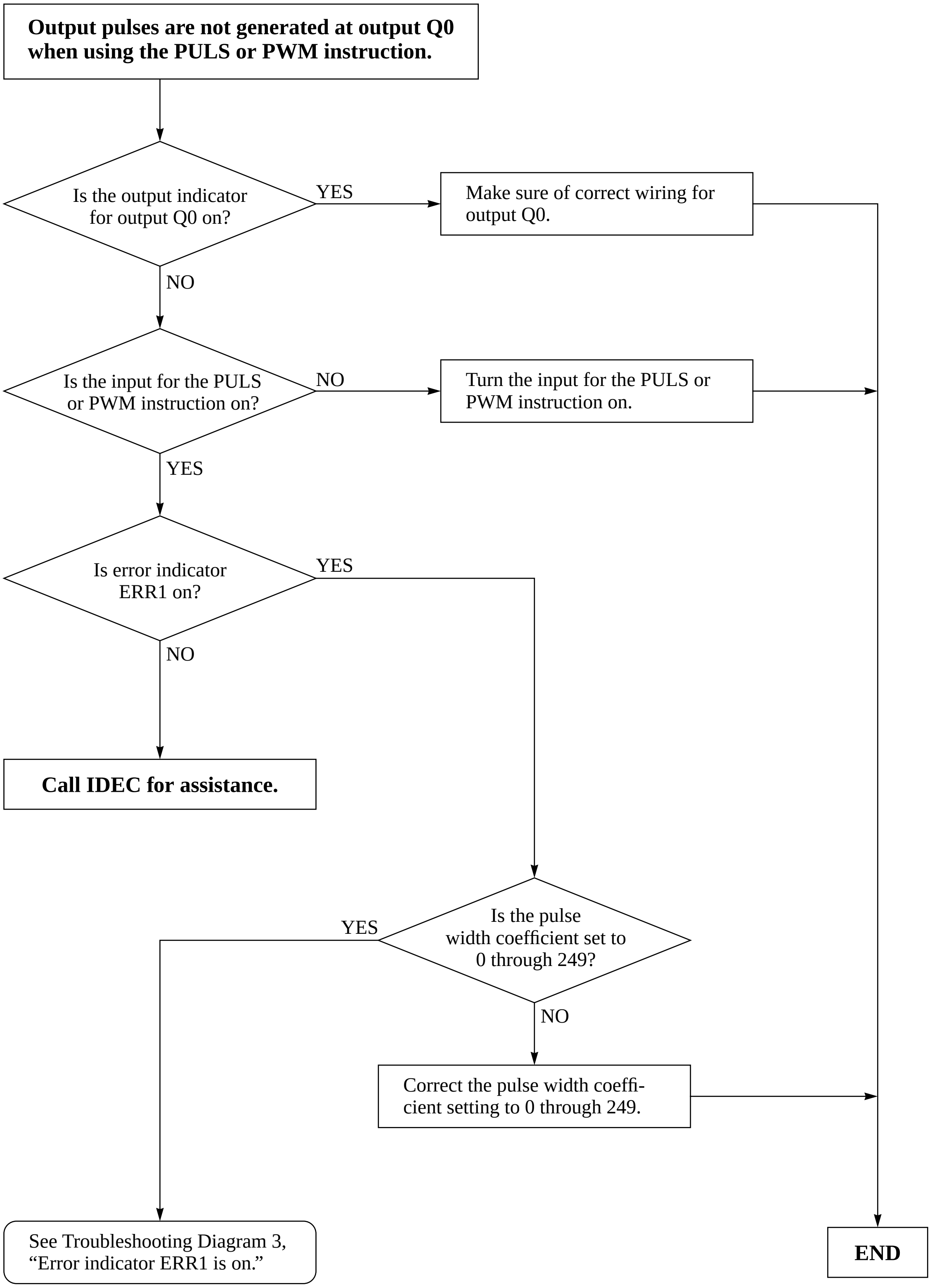
Troubleshooting Diagram 9


Normal voltage does not appear on sensor power terminals.
Is the POW (power) indicator on?
YES
NO
Supply the rated voltage.
AC power type: 100 to 240V AC
DC power type: 24V DC
Is the POW (power) indicator on?
YES
NO
Is the sensor power outputted correctly?
NO
YES
END
Is error indicator ERR2 on?
YES
NO
Call IDEC for assistance.
See Troubleshooting Diagram 4, "Error indicator ERR2 is on."

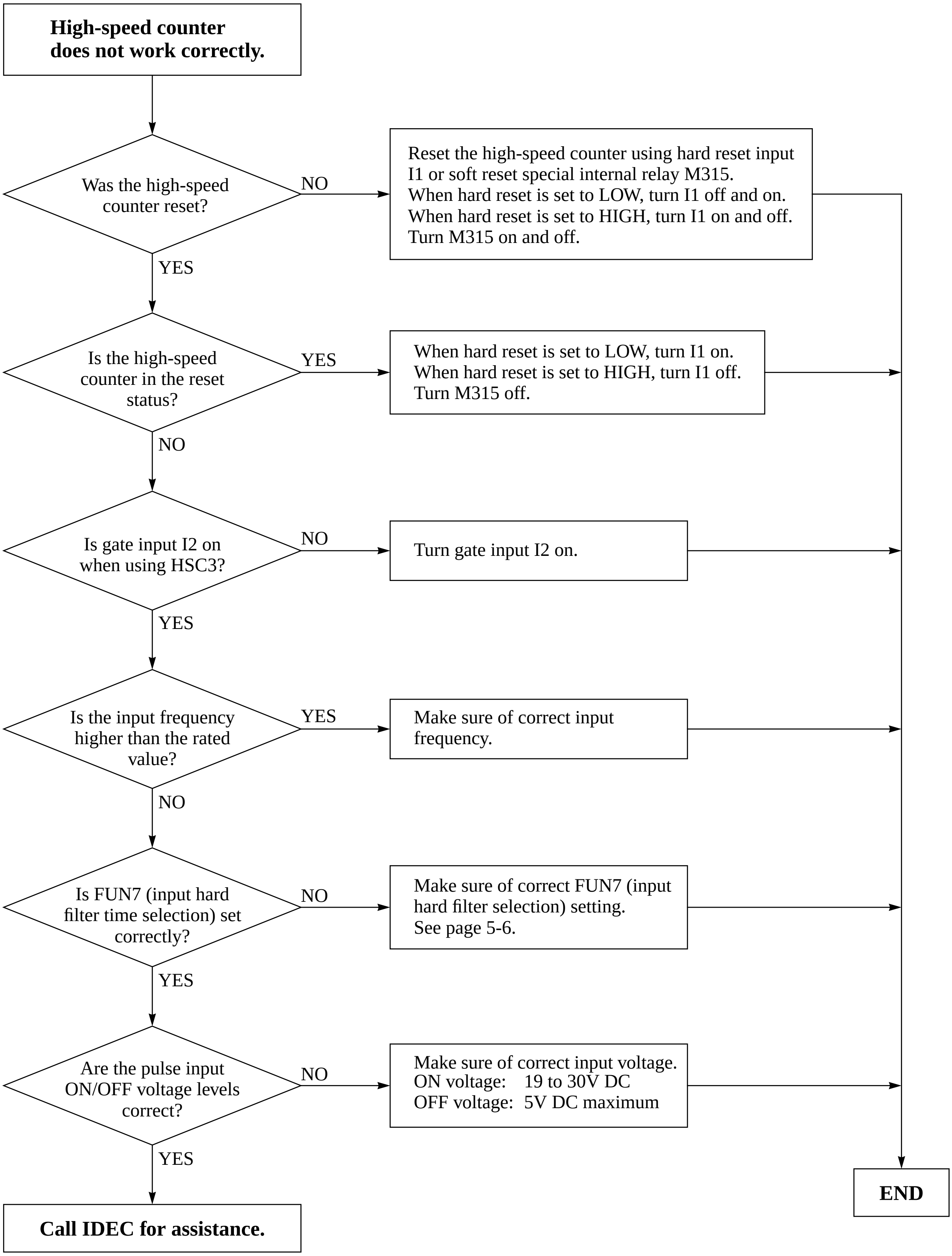
Troubleshooting Diagram 10


Expansion link or data link is impossible.
Is M306 on at the base station (expansion link) or master station (data link)?
NO
YES
Turn link communication prohibit flag special internal relay M306 off using the program loader.
MON
C SOT M
3 BPP
0
6 CC>=
F RST Q
↵
Is the communication cable connected to data link terminals correctly?
NO
Make sure of correct wiring. See Expansion Link Function on page 4-6 or Data Link Function on page 4-8.
YES
Check error codes at the base station (expansion link) or at all stations (data link).
FUN
2 BRD
0
▼
Is the error code 0 at all stations?
NO
Reset the error codes at the base station (expansion link) or at all stations (data link).
FUN
2 BRD
0
▼
DEL
↵
YES
At the base or master station, execute FUN27 (link formatting sequence),
FUN
2 BRD
7 END
▼
↵
or turn M307 (link communication initialize flag) on during operation.
MON
C SOT M
3 BPP
0
7 END
SET I
↵
Turn power off at the base or master station, and turn power on after a few seconds.
Are error codes cleared to 0 at all stations?
YES
NO
Call IDEC for assistance.
END

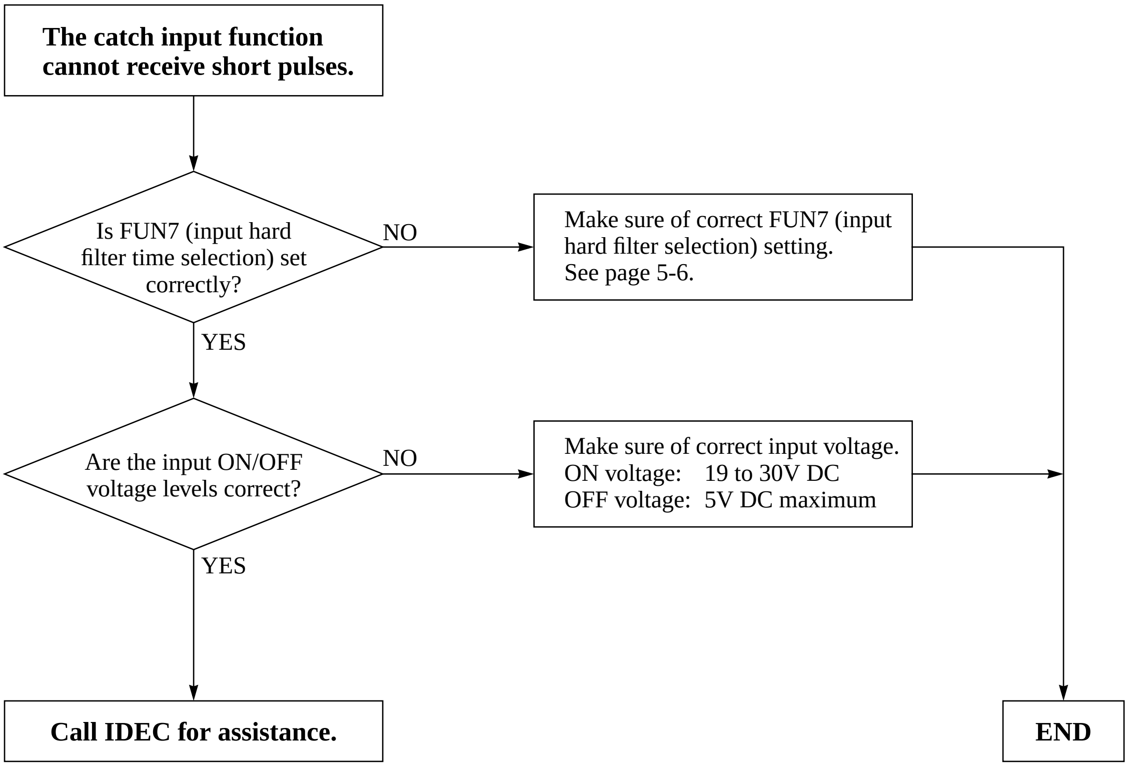
Troubleshooting Diagram 11


Output pulses are not generated at output Q0 when using the PULS or PWM instruction.
Is the output indicator for output Q0 on?
YES
Make sure of correct wiring for output Q0.
NO
Is the input for the PULS or PWM instruction on?
NO
Turn the input for the PULS or PWM instruction on.
YES
Is error indicator ERR1 on?
YES
NO
Call IDEC for assistance.
Is the pulse width coefficient set to 0 through 249?
YES
NO
Correct the pulse width coefficient setting to 0 through 249.
See Troubleshooting Diagram 3, "Error indicator ERR1 is on."
END

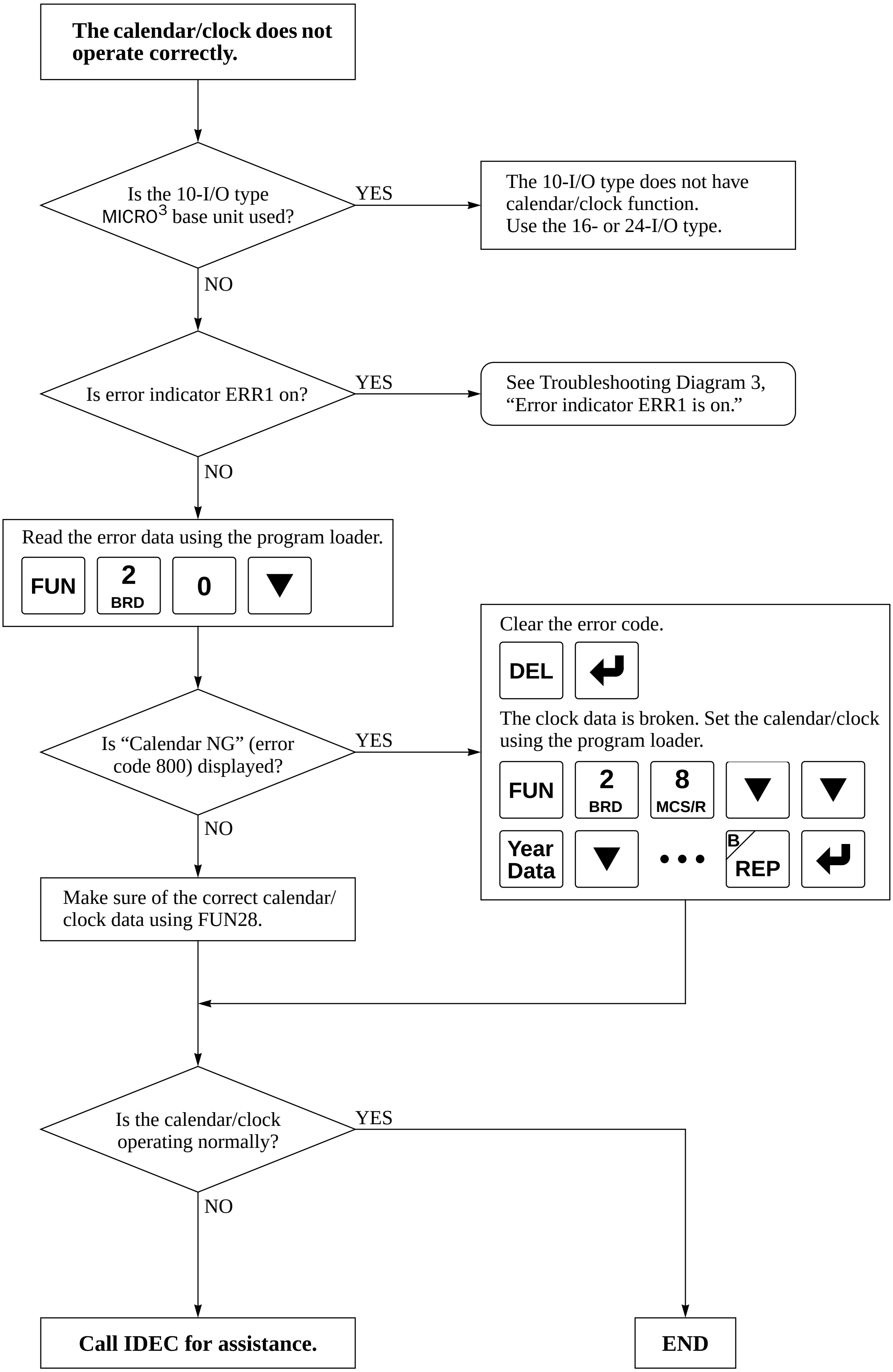
Troubleshooting Diagram 12


High-speed counter does not work correctly.
Was the high-speed counter reset?
NO
Reset the high-speed counter using hard reset input I1 or soft reset special internal relay M315.
When hard reset is set to LOW, turn I1 off and on.
When hard reset is set to HIGH, turn I1 on and off.
Turn M315 on and off.
YES
Is the high-speed counter in the reset status?
YES
When hard reset is set to LOW, turn I1 on.
When hard reset is set to HIGH, turn I1 off.
Turn M315 off.
NO
Is gate input I2 on when using HSC3?
NO
Turn gate input I2 on.
YES
Is the input frequency higher than the rated value?
YES
Make sure of correct input frequency.
NO
Is FUN7 (input hard filter time selection) set correctly?
NO
Make sure of correct FUN7 (input hard filter selection) setting.
See page 5-6.
YES
Are the pulse input ON/OFF voltage levels correct?
NO
Make sure of correct input voltage.
ON voltage: 19 to 30V DC
OFF voltage: 5V DC maximum
YES
Call IDEC for assistance.
END

Troubleshooting Diagram 13


The catch input function cannot receive short pulses.
Is FUN7 (input hard filter time selection) set correctly?
NO
Make sure of correct FUN7 (input hard filter selection) setting.
See page 5-6.
YES
Are the input ON/OFF voltage levels correct?
NO
Make sure of correct input voltage.
ON voltage: 19 to 30V DC
OFF voltage: 5V DC maximum
YES
Call IDEC for assistance.
END

Troubleshooting Diagram 14



The calendar/clock does not operate correctly.

Is the 10-I/O type MICRO3 base unit used?

The 10-I/O type does not have calendar/clock function. Use the 16- or 24-I/O type.

Is error indicator ERR1 on?

See Troubleshooting Diagram 3, "Error indicator ERR1 is on."

Read the error data using the program loader.

Is "Calendar NG" (error code 800) displayed?

Clear the error code.

The clock data is broken. Set the calendar/clock using the program loader.

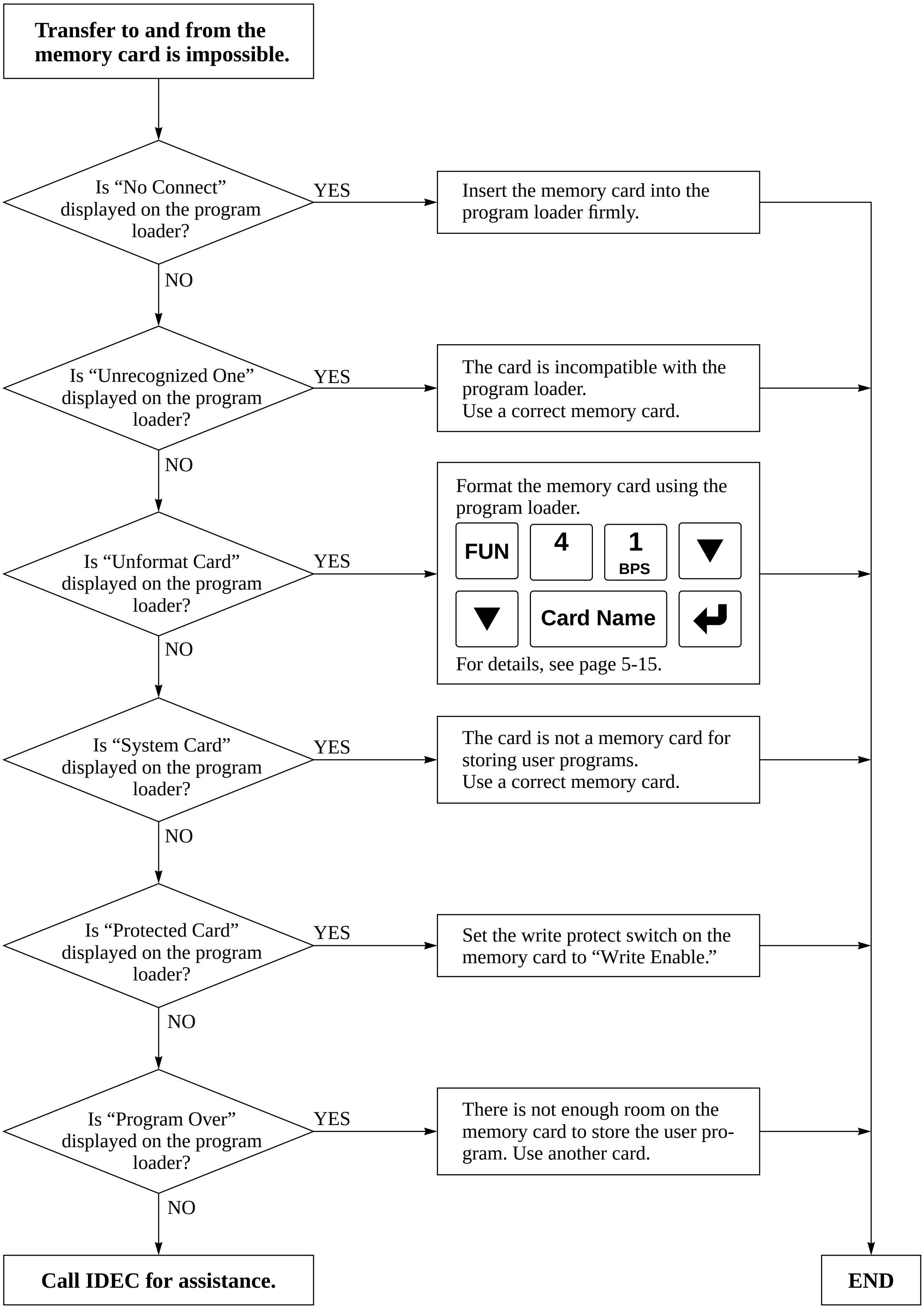
Make sure of the correct calendar/clock data using FUN28.

Is the calendar/clock operating normally?

Call IDEC for assistance.

END

Troubleshooting Diagram 15


Transfer to and from the memory card is impossible.
Is "No Connect" displayed on the program loader?
YES
Insert the memory card into the program loader firmly.
NO
Is "Unrecognized One" displayed on the program loader?
YES
The card is incompatible with the program loader. Use a correct memory card.
NO
Is "Unformat Card" displayed on the program loader?
YES
Format the memory card using the program loader.
FUN
4
1 BPS
▼
▼
Card Name
↵
For details, see page 5-15.
NO
Is "System Card" displayed on the program loader?
YES
The card is not a memory card for storing user programs. Use a correct memory card.
NO
Is "Protected Card" displayed on the program loader?
YES
Set the write protect switch on the memory card to "Write Enable."
NO
Is "Program Over" displayed on the program loader?
YES
There is not enough room on the memory card to store the user program. Use another card.
NO
Call IDEC for assistance.
END

Execution Times for Instructions

Instruction	Operand and Condition		Maximum Execution Time (μsec)	
			Standard Processing	High-speed Processing
LOD, LODN	I, Q, M		2.0	0.4
	R, T, C		2.0	0.8
OUT, OUTN			4.0	1.0
SET, RST	Q, M		3.4	0.8
	R		3.4	1.2
AND, ANDN, OR, ORN	I, Q, M		1.4	0.2
	R, T, C		1.4	0.6
AND LOD, OR LOD			1.2	0.4
BPS			7.0	5.0
BRD			1.4	0.6
BPP			6.0	6.0
TIM, TMH, TMS	When stopped		23	44
	While timing down		31	52
	After timeout		31	51
CNT	0	Preset input ON	20	41
		After countout	20	40
		Others	18	39
	1	Pulse input ON	16	37
		After countout	19	40
		Others	14	35
	2 to 31 (2 to 15)	Reset input ON	12	32
		After countout	22	43
		Others	21	42
CC=, CC≥			12	32
SFR, SFRN **(N bits)**	Reset input ON		0.4N + 27	0.4N + 48
	Pulse input ON		0.4N + 32	0.4N + 54
	Others		28	48
SOTU, SOTD			17	37
JMP, JEND, MCS, MCR			6.0	5.0
END	See the next page.			
MOV, MOVN	M → M		109	129
	D → D		61	80
CMP=, <>, <, >, <=, >=	M ↔ M → M		121	140
	D ↔ D → M		80	99
ADD	M + M → D		121	141
	D + D → D		80	99
SUB	M – M → D		121	141
	D – D → D		80	99
MUL	M × M → D		131	151
	D × D → D		90	109
DIV	M ÷ M → D		182	201
	D ÷ D → D		140	159

Breakdown of END Processing Time

The END processing time depends on the MICRO^3 settings and system configuration. The total of execution times for applicable conditions shown below is the actual END processing time.

Item	Condition	Execution Time
Housekeeping	Standard processing mode	200 μsec
	High-speed processing mode	220 μsec
I/O service	Base unit only	130 μsec
	Expansion link system	9 to 10 msec
Control data register service	All control data registers enabled	15 μsec
Calendar/clock function processing (see note1)		150 μsec
Data link master station processing (see note 2)	Data link system	12.5 to 13 msec

Note 1: Calendar/clock function is processed once every 500 msec in the 16- and 24-I/O MICRO^3 base units. The 10-I/O MICRO^3 base unit does not have the calendar/clock function.

Note 2: Data link slave stations are processed in interrupt processing asynchronous to the ordinary system processing.

Processing times of advanced instructions are generally approximately 20 μsec longer in the high-speed processing mode than in the standard processing mode.

In addition to processing user program instructions and END instruction, the MICRO^3 system processing includes interrupt processing of various functions.

Type List

◆ For the MICRO^{3}C, see the MICRO^{3}C User's Manual. ◆

MICRO^3 CPU Base Units / Expansion I/O and Program Loader

Name				Total I/O Points (Inputs/Outputs)	Clock and Calendar	Type No.
MICRO^3 CPU Base Unit, Expansion I/O	AC Power Power Voltage: 100-240V AC 50/60Hz	24V DC Input Sink/Source	Relay Output 240V AC, 2A 30V DC, 2A	10 points (6 in / 4 out)	Without	FC2A-C10A1
				16 points (9 in / 7 out)	With	FC2A-C16A1
				24 points (14 in / 10 out)	With	FC2A-C24A1
			Transistor Sink Output 24V DC, 0.5A	10 points (6 in / 4 out)	Without	FC2A-C10B1
				16 points (9 in / 7 out)	With	FC2A-C16B1
				24 points (14 in / 10 out)	With	FC2A-C24B1
		120V AC Input (85-132V AC) 50/60Hz	Relay Output 240V AC, 2A 30V DC, 2A	16 points (9 in / 7 out, 24-pt housing)	With	FC2A-CA16A1
	DC Power Power Voltage: 24V DC	24V DC Input Sink/Source	Relay Output 240V AC, 2A 30V DC, 2A	10 points (6 in / 4 out)	Without	FC2A-C10A4
				16 points (9 in / 7 out)	With	FC2A-C16A4
				24 points (14 in / 10 out)	With	FC2A-C24A4
			Transistor Sink Output 24V DC, 0.5A	10 points (6 in / 4 out)	Without	FC2A-C10B4
				16 points (9 in / 7 out)	With	FC2A-C16B4
				24 points (14 in / 10 out)	With	FC2A-C24B4
			Transistor Protect Source Output 24V DC, 0.5A	10 points (6 in / 4 out)	Without	FC2A-C10D4
				16 points (9 in / 7 out)	With	FC2A-C16D4
				24 points (14 in / 10 out)	With	FC2A-C24D4
Program Loader (loader cable not included)						FC2A-HL1E

Type List, continued

◆ *For the MICRO^{3}C, see the MICRO^{3}C User's Manual.* ◆

Cables and Accessories

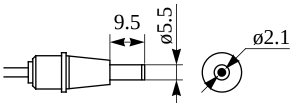
Name	Function	Type No.
Loader Cable 1 (2m/6.56 ft. long)	Used to connect the program loader to the MICRO^3 base unit (Loader cable is not included with program loader.)	FC2A-KL1
Loader Cable 2 (5m/16.4 ft. long)		FC2A-KL2
Computer Link Cable	Used to connect the MICRO^3 base unit or program loader to IBM PC (in the 1:1 computer link system), 2m (6.56 ft.) long, with D-sub 9-pin female connector to connect to computer	FC2A-KC2
Jack Converter (included with computer link cable)	Used on the computer link cable to connect an AC adapter to power the program loader connected to IBM PC	FC2A-CJ1
Memory Card	SRAM memory card to store 31 user programs max. (64K bytes)	FC2A-MC1
Expansion Cable	Used to connect MICRO^3 base units for close mounting in the expansion link system, 250mm (9.84") long	FC2A-KE1
Computer Link Interface Unit	Used to connect MICRO^3 base unit to the RS232C/RS485 converter in the 1:N computer link system	FC2A-LC1
Computer Link Interface Cable	Used to connect MICRO^3 base unit to the computer link interface unit in the 1:N computer link system, 100 mm (3.937") long	FC2A-KC3
RS232C/RS485 Converter	Used to connect the computer link interface unit to IBM PC in the 1:N computer link system	FC2A-MD1
RS232C Cable (4-wire)	Used to connect the RS232C/RS485 converter to IBM PC in the 1:N computer link system, 1.5m (4.92 ft.) long, with D-sub 9-pin female connector to connect to computer	HD9Z-C52
A/D Converter Unit 1 (0 to 5V)	Used to convert an analog signal to a digital signal and send it to input I0 of the MICRO^3 base unit (Resolution: 8 bits)	FC2A-AD1
A/D Converter Unit 2 (0 to 10V)		FC2A-AD2
A/D Converter Unit 3 (±5V)		FC2A-AD3
A/D Converter Unit 4 (4 to 20mA)		FC2A-AD4
A/D Converter Unit 5 (±10V)		FC2A-AD5
D/A Converter Unit 1 (0 to 5V)	Used to convert a digital signal (PWM signal) from output Q0 of the MICRO^3 base unit to an analog signal (Resolution: 8 bits)	FC2A-DA1
D/A Converter Unit 2 (0 to 10V)		FC2A-DA2
D/A Converter Unit 3 (±5V)		FC2A-DA3
D/A Converter Unit 4 (4 to 20mA)		FC2A-DA4
D/A Converter Unit 5 (±10V)		FC2A-DA5
Analog Timer Unit	To set timer preset value externally (See page 4-20 – accessories.)	PFA-1U11
DIN Rail	35-mm-wide DIN rail to mount MICRO^3 base unit, 1m (3.28 ft.) long	BAA1000
Mounting Clip	Used on DIN rail to fasten the MICRO^3 base unit	BNL6
CUBIQ	Programming and monitoring software used on PC (3.5" diskette)	FC9Y-LP1E314

AC Adapter

When using the program loader for off line programming or communication with a computer, an AC adapter is required to power the program loader. AC adapter output capacity: 5 to 6.5V DC, 4W

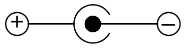
The RS232C/RS485 converter is powered by 24V DC source or an AC adapter with 9V DC, 350mA output capacity.

The output plug of the AC adapter applicable to both the program loader and RS232C/RS485 converter is shown on the right.



Dimensions in mm.

Polarity



UNITED STATES
IDEC CORPORATION
1213 Elko Drive
Sunnyvale, CA 94089-2240, USA
Tel (408) 747-0550
Toll Free (800) 262-IDEC
Fax (408) 744-9055
Fax (800) 635-6246
E-mail ideccorp@industry.net
www.industry.net/ideccorp

JAPAN
IDEC IZUMI CORPORATION
7-31, Nishi-Miyahara
1-Chome, Yodogawa-ku
Osaka 532, Japan
Tel (06) 398-2571
Fax (06) 392-9731

CANADA
IDEC CANADA LIMITED
Unit 22-151 Brunel Road
Mississauga, Ontario, L4Z 1X3, Canada
Tel (905) 890-8561
Fax (905) 890-8562

GERMANY
IDEC ELEKTROTECHNIK GmbH
Wendenstraße 331
D-20537 Hamburg, Germany
Tel (040) 25 11 91-93
Fax (040) 25 4 33 61

UNITED KINGDOM
IDEC ELECTRONICS LIMITED
Unit 12, Canbury Business Park
Elm Crescent
Kingston-Upon-Thames
Surrey KT2 6HJ, United Kingdom
Tel (0181) 549-0737
Fax (0181) 546-0963

HONG KONG
IDEC IZUMI (H.K.) CO., LTD.
Room No.1409, Tower 1 Silvercord
30 Canton Road, Tsimshatsui
Kowloon, Hong Kong
Tel (02) 376-2823
Fax (02) 376-0790

TAIWAN
IDEC TAIWAN CORPORATION
3F., No.75, Hsin Tai Wu Road, Sec. 1
Hsi-Chih, Taipei County, Taiwan
Republic of China
Tel (02) 698-2601
Fax (02) 698-2709

AUSTRALIA
IDEC AUSTRALIA PTY. LTD.
2/3 Macro Court
Rowville, Victoria 3178
Australia
Tel (03) 9763-3244
Fax (03) 9763-3255

Users Manual #EM317-0
Printed in the USA 5U 2/97