



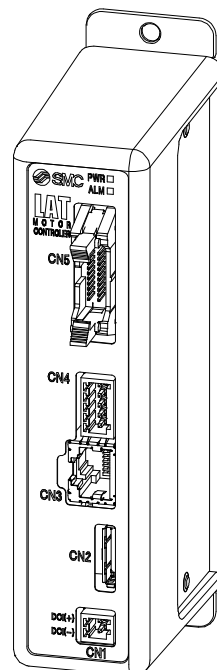
# Operation Manual

Product Name

**Card Motor Controller**  
(Serial Communication Modbus Protocol edition)

Model/ Series/ Product Number

**LATCA Series**



**SMC Corporation**

## **About this operation manual**

This operation manual summarizes how to operate the Card Motor Controller (LATCA-\*, Ver2.5) using the Modbus serial communication protocol.

Refer to the "Card Motor Controller (Step Data input edition)" operation manual for information relating to all control modes.

When using the LATCA controller in Step Data input mode, refer to the "Card Motor Controller (Step Data input edition)" operation manual, and when using it in pulse input mode, refer to the "Card Motor Controller (Pulse input edition)" operation manual.

When communicating using SMC's original protocol, refer to the "Card Motor Controller (Serial Communication Original Protocol edition)" operation manual.

## **Controller history**

Use the latest version of the controller.

Function	Controller version (the time in the bracket indicates the schedule for switching)		
	Ver 2.0 (August 2014)	Ver2.1 to Ver2.4 (Sept. 2015)	Ver2.5 (June 2020)
Corresponding card motor	6 types	LAT3M-50 LAT3F-50	LAT3M-50 LAT3F-50
Alarm history count	4 points	4 points	20 points
Communication speed	19200bps	2,400bps 9,600bps 19,200bps 38,400bps 57,600bps	2,400bps 9,600bps 19,200bps 38,400bps 57,600bps

# Contents

<b>1. Safety Instructions .....</b>	<b>4</b>
<b>2. Summary .....</b>	<b>6</b>
2.1 Purpose of the Operation Manual.....	6
2.2 Notation .....	6
2.3 Abbreviations.....	6
<b>3. Scope .....</b>	<b>7</b>
3.1 Scope.....	7
<b>4. Hardware Specifications.....</b>	<b>9</b>
4.1 Input Specifications .....	9
4.2 Communication Connector Pin Assignment.....	9
4.3 Communication Circuit.....	9
4.4 Wiring Examples.....	10
<b>5. Software Specifications .....</b>	<b>11</b>
5.1 Serial Communication Specification.....	11
5.2 Memory Outline .....	12
5.3 Message Frame Formats .....	13
5.4 Details of Function Codes supported by LATCA.....	17
(1) 01h Read the status of output signals .....	17
(2) 02h Read the status of input signals.....	19
(3) 03h Read data contents of holding register addresses .....	21
(4) 05h Write (force) a single output signal.....	23
(5) 06h Write data to a single holding register address.....	25
(6) 08h Diagnostics of communication using echo back .....	27
(7) 0Fh Write (force) multiple output signals .....	29
(8) 10h Write data to multiple holding register addresses .....	31
5.5 Memory Map.....	33
(1) Internal flags.....	33
(2) Step Data Operation .....	35
(3) Direct operation command .....	39
(4) Status data.....	41
5.6 List of Request Functions .....	43
5.7 Details of Functions .....	44
(1) Step Data Setting / Reading, Direct Operation Step Data Setting / Reading ...	44
(2) Read Input Signal Status .....	50
(3) Read Output Signal Status and Internal Flags .....	51
(4) Read Internal parameters .....	52

(5) Reading and Selection of Operation Mode .....	54
(6) Commands for Virtual I/O Operation by Serial Communication .....	55
(7) Direct Operation Command .....	57
(8) Read / Clear Alarm History .....	58
5.8 Error Codes.....	60
<b>6. Card Motor Controller Operation Examples .....</b>	<b>61</b>
6.1 Basic Settings, I/O Configuration .....	61
6.2 Step Data configuration .....	61
6.3 Operation Data Acquisition .....	62
6.4 Operation Instruction Method .....	62
6.5 Program Examples .....	63
(1) Basic settings .....	63
(2) Communication check .....	63
(3) Step Data configuration examples .....	64
(4) Return to Origin .....	64
(5) Positioning operation (Step Data Operation) example.....	65
(6) Positioning operation (Direct Operation) example .....	66
(7) Operation data acquisition example.....	67
(8) Alarm history acquisition example .....	67
(9) Alarm history clearance example .....	67
<b>7. Option (Separately Sold Products) .....</b>	<b>69</b>
7.1 Communication cable .....	69
7.2 Branch communication cable .....	70
<b>8. Reference Information .....</b>	<b>72</b>
8.1 LRC Checksum Calculation Procedure .....	72
8.2 CRC Check Sum Calculation Procedure.....	72
8.3 Communication Response Time Guides .....	74
8.4 ASCII Code List.....	76



# LATCA Series Controller

## 1. Safety Instructions

These safety instructions are intended to prevent hazardous situations and/or equipment damage. These instructions indicate the level of potential hazard with the labels of “**Caution**,” “**Warning**” or “**Danger**.” They are all important notes for safety and must be followed in addition to International Standards (ISO/IEC)\*<sup>1)</sup>, and other safety regulations.

\*1) ISO 4414: Pneumatic fluid power - General rules and safety requirements for systems and their components  
ISO 4413: Hydraulic fluid power - General rules and safety requirements for systems and their components  
IEC 60204-1: Safety of machinery - Electrical equipment of machines - Part 1: General requirements  
ISO 10218-1: Robots and robotic devices - Safety requirements for industrial robots - Part 1: Robots  
etc.



### **Danger**

**Danger** indicates a hazard with a high level of risk which, if not avoided, will result in death or serious injury.



### **Warning**

**Warning** indicates a hazard with a medium level of risk which, if not avoided, could result in death or serious injury.



### **Caution**

**Caution** indicates a hazard with a low level of risk which, if not avoided, could result in minor or moderate injury.

### **Warning**

#### **1. The compatibility of the product is the responsibility of the person who designs the equipment or decides its specifications.**

Since the product specified here is used under various operating conditions, its compatibility with specific equipment must be decided by the person who designs the equipment or decides its specifications based on necessary analysis and test results. The expected performance and safety assurance of the equipment will be the responsibility of the person who has determined its compatibility with the product. This person should also continuously review all specifications of the product referring to its latest catalog information, with a view to giving due consideration to any possibility of equipment failure when configuring the equipment.

#### **2. Only personnel with appropriate training should operate machinery and equipment.**

The product specified here may become unsafe if handled incorrectly. The assembly, operation and maintenance of machines or equipment including our products must be performed by an operator who is appropriately trained and experienced.

#### **3. Do not service or attempt to remove product and machinery/equipment until safety is confirmed.**

1. The inspection and maintenance of machinery/equipment should only be performed after measures to prevent falling or runaway of the driven objects have been confirmed.
2. When the product is to be removed, confirm that the safety measures as mentioned above are implemented and the power from any appropriate source is cut, and read and understand the specific product precautions of all relevant products carefully.
3. Before machinery/equipment is restarted, take measures to prevent unexpected operation and malfunction.

#### **4. Our products cannot be used beyond their specifications. Our products are not developed, designed, and manufactured to be used under the following conditions or environments. Use under such conditions or environments is not covered.**

1. Conditions and environments outside of the given specifications, or use outdoors or in a place exposed to direct sunlight.
2. Use for nuclear power, railways, aviation, space equipment, ships, vehicles, military application, equipment affecting human life, body, and property, fuel equipment, entertainment equipment, emergency shut-off circuits, press clutches, brake circuits, safety equipment, etc., and use for applications that do not conform to standard specifications such as catalogs and operation manuals.
3. Use for interlock circuits, except for use with double interlock such as installing a mechanical protection function in case of failure. Please periodically inspect the product to confirm that the product is operating properly.



# LATCA Series Controller

## 1. Safety Instructions



### Caution

We develop, design, and manufacture our products to be used for automatic control equipment, and provide them for peaceful use in manufacturing industries.

**Use in non-manufacturing industries is not covered.**

Products we manufacture and sell cannot be used for the purpose of transactions or certification specified in the Measurement Act.

The new Measurement Act prohibits use of any unit other than SI units in Japan.

### Limited warranty and Disclaimer/Compliance Requirements

The product used is subject to the following “Limited warranty and Disclaimer” and “Compliance Requirements”.

Read and accept them before using the product.

#### Limited warranty and Disclaimer

1. The warranty period of the product is 1 year in service or 1.5 years after the product is delivered, whichever is first. \*2)

Also, the product may have specified durability, running distance or replacement parts. Please consult your nearest sales branch.

2. For any failure or damage reported within the warranty period which is clearly our responsibility, a replacement product or necessary parts will be provided.

This limited warranty applies only to our product independently, and not to any other damage incurred due to the failure of the product.

3. Prior to using SMC products, please read and understand the warranty terms and disclaimers noted in the specified catalog for the particular products.

**\*2) Vacuum pads are excluded from this 1 year warranty.**

A vacuum pad is a consumable part, so it is warranted for a year after it is delivered.

Also, even within the warranty period, the wear of a product due to the use of the vacuum pad or failure due to the deterioration of rubber material are not covered by the limited warranty

#### Compliance Requirements

1. The use of SMC products with production equipment for the manufacture of weapons of mass destruction (WMD) or any other weapon is strictly prohibited.

2. The exports of SMC products or technology from one country to another are governed by the relevant security laws and regulations of the countries involved in the transaction. Prior to the shipment of a SMC product to another country, assure that all local rules governing that export are known and followed.

## 2. Summary

### 2.1 Purpose of the Operation Manual

This operation manual describes the Modbus communication specifications for the Card Motor controller (LATCA-\*, Ver.2.5).

### 2.2 Notation

Unless stated otherwise, this operation manual follows the notation detailed below.

- (1) Values are written in big-endian byte order.
- (2) Values are generally written in decimal, however those ending with “h” are in hexadecimal, and those ending in “b” in binary.
- (3) ASCII characters representing a hexadecimal numerical value.

e.g.) The decimal number 37 = 25h, and would be stored as ASCII “25” i.e. 32h, 35h.

### 2.3 Abbreviations

This manual uses the following abbreviations.

GUI	: Graphical User Interface
I/O	: Input/Output
I/F	: Interface
MSB	: Most Significant Bit
ASCII	: American Standard Code for Information Interchange
RTU	: Remote Terminal Unit
BCD	: Binary-coded decimal
MFC	: Microsoft Foundation Class
API	: Application Programming Interface
PLC	: Programmable Logic Controller
PC	: Personal computer
COM	: Computer on Module
RS-485	: Recommended Standard 485
USB	: Universal Serial Bus
LRC	: Longitudinal Redundancy Check
CRC	: Cyclical Redundancy Check

## 3. Scope

### 3.1 Scope

This operation manual applies only for communication between up to 16 Card Motor controllers (LATCA-\*, Version 2.5) and a communication device such as a PLC other than the Card Motor Controller Configuration Software, using the functions as described below.

#### (1) Step Data configuration

The following parameters may be configured.

- Operation mode selection (positioning/pushing)
- Movement mode (ABS/REL)
- Target position
- Positioning time
- Speed
- Acceleration
- Deceleration
- Thrust setting value
- Load mass
- Pushing speed
- Positioning width
- Threshold
- Area range

#### (2) Operation data acquisition

The following internal Card Motor controller operation data may be acquired.

- I/O data
- Positioning data
- Speed data
- Equivalent thrust value data
- Target position
- Step Data No. executed

#### (3) Step Data Operation

By employing pre-set Step Data, operation instructions will be given from the upper-level device such as a PLC without using parallel I/O signals input.

- \* When setting Step Data, turn the power supply to the Card Motor off and ensure a Return to Origin is performed after the Step Data configuration is complete.

#### (4) Direct Operation

By employing Direct Operation Step Data, operation instructions will be given from the upper-level device such as a PLC without using parallel I/O signals input.

- \* Direct Operation Step Data cannot be saved internally to the Card Motor controller.

A Return to Origin is not required after setting Direct Operation Step Data.

- Differences between Step Data Operation and Direct Operation:

Configuration of Step Data in Step Data Operation and Direct Operation have the following differences.

Criteria	Step Data Operation	Direct Operation
Power supply to Card Motor must be turned OFF when settings have been changed.	Yes	No
Step Data stored/processed after setting changes	Yes	No
Return to Origin to be executed after setting changes	Yes	No
Configuration data maintained even if the power supply is turned OFF	Saved	Not saved



(5) Acquisition and clearance of alarm history

The alarm history saved in the controller may be cleared / acquired.

 **Caution**

Use the "LATC Configurator" controller configuration software to pre-set the following basic controller settings.

1. Control signal input type (Step Data Input Type / Pulse Input Type)
2. Card Motor part number
3. Card Motor mounting orientation
4. Return to Origin method
5. Step Data Input Method
6. Controller ID configuration (factory default setting = 1)
7. Output signal functions
8. Communication speed

When the controller is set to "Pulse Input Type" control signal mode, operation cannot be performed via the serial communication.

Some Step Data settings, acquisition of operation information and alarm history only are available.

Refer to section **"5.6 List of Request Functions"** for the functions which are available with pulse input type control signal.

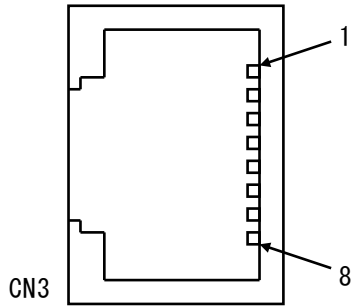
## 4. Hardware Specifications

### 4.1 Input Specifications

Based on the RS-485 (2-wire type)

### 4.2 Communication Connector Pin Assignment

Connector used: Hirose Electronics TM11R-5M2-88



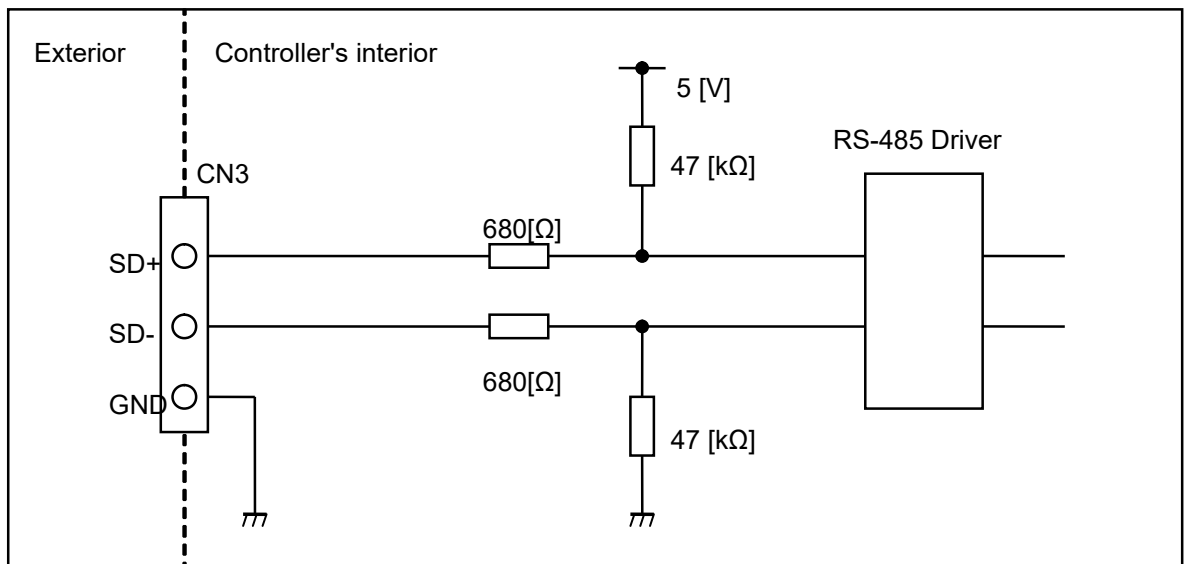
Terminal No.	Function	Description
1	N C	Not connected
2	N C	Not connected
3	SD +	Connect + signal lead *1)
4	SD -	Connect - signal lead *1)
5	N C	Not connected
6	N C	Not connected
7	N C	Not connected
8	N C	Not connected

Note 1) Refer to the operation manual for the controller module to be connected before wiring.

If the functions are indicated as A / B, connect the signals as + / -.

As this product is a 2-wire type, TXD and RXD are written together as "SD"

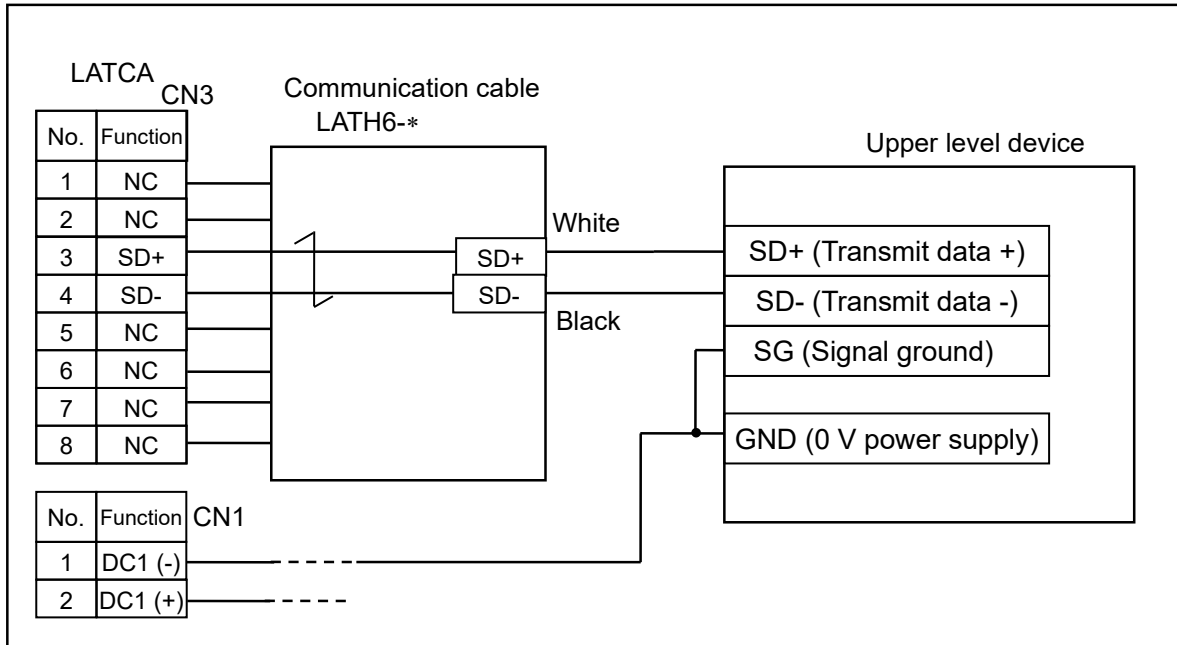
### 4.3 Communication Circuit



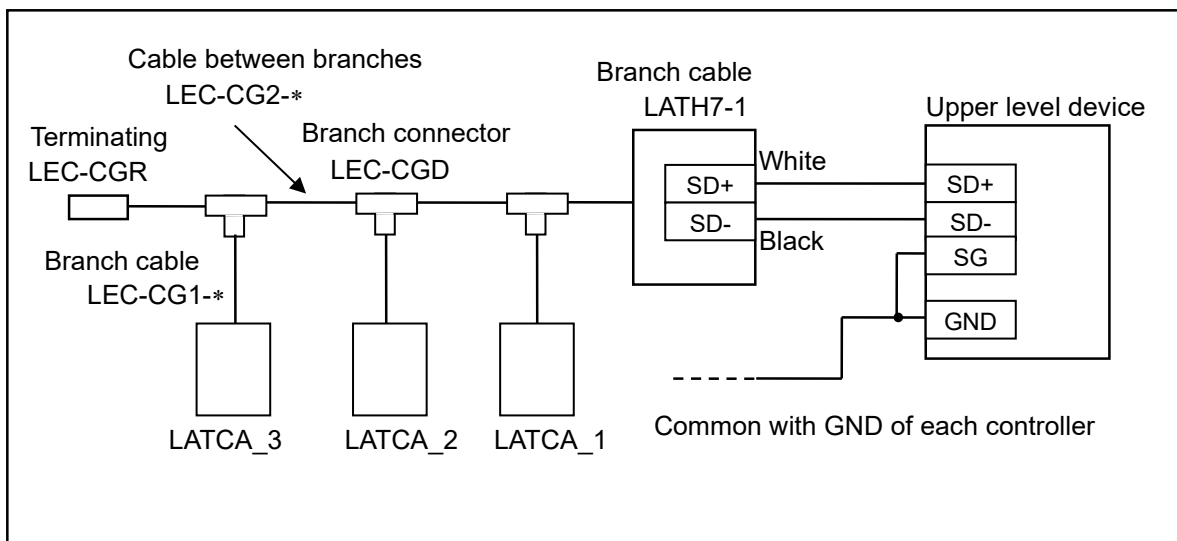
Note 1) There is no terminating resistor built into the controller communication circuit.

## 4.4 Wiring Examples

A wiring example of one controller connected to an upper level device is shown below.



A wiring example of multiple controllers (3 pcs.) connected to an upper level device is shown below.



### **Warning**

**The power supply 0 V of all the controllers used for serial communication should be at the same potential as that of the master communication device.**

**Turn OFF the power supply before connecting and removing any serial connector.**

Otherwise, the controller may be damaged.

## 5. Software Specifications

### 5.1 Serial Communication Specification

The serial communication method with the upper level device uses the single Master-Slave Modbus protocol. The upper level device is a master, and the LATCA controller is a slave.

When the master (upper level device) submits a request, the slave (LATCA controller) receives the request and responds (the slave does not submit requests). However, the slave does not respond to broadcast requests.

This controller is compatible with both Modbus ASCII and RTU transmission modes. The controller recognizes the received data type automatically.

Criteria	Description	
Protocol	Modbus	
Transmission modes	ASCII / RTU (recognized by LATCA automatically)	
Node type	Slave (controller)	
Error check	None	
Frame size	Variable length; 128 bytes maximum	
Communication method	RS-485; asynchronous mode	
	Communication speed	2,400bps 9,600bps 19,200bps (Default value) 38,400bps 57,600bps
	Data bits	8 bits
	Parity	Even parity
	Stop bit	1 bit
	Flow control	None

## 5.2 Memory Outline

The Controller data can be confirmed and changed and operation commands executed by accessing the internal data memory, which can be operated using serial communication.

The table below shows an outline of the memory areas which can be accessed using serial communication.

Item	Address	Details
Internal flag area (Coils and Discrete inputs)	X40 to X4F Y10 to Y1F, Y30	Area containing flags indicating the input and output status of the parallel I/O signal and the internal status of the controller. The controller can be operated by writing to and reading these flags or from parallel I/O.
Step Data area	D0410h to D05EF	Memory area where the contents of the Step Data parameters are stored. The contents of the Step Data parameters are changed by writing data to this area.
Direct Operation command area	D9100 to D9111	Memory area used to execute direct operation of the Card Motor by specifying the position and speed. The specified operation is executed by writing data such as a target position and positioning time (or speed) and then executed by changing the value of the D9100 and D9101 address from "0000h" to "0100h".
Status data area	D9000 to D9006	Memory area where I/O information, position, speed, equivalent thrust and target position are stored. Information such as current position and speed can be confirmed by reading this data.
Alarm data area	D0600 to D0613	Memory area where the alarm history of the controller is stored. The alarm history can be checked by reading this data. The contents of this memory area can also be cleared.

### Caution

**The Step Data area and alarm data area access the EEPROM.**

The write limit of the EEPROM is approximately 4 million cycles. Please avoid frequent writing.

**The internal coil area, status data area and Direct Operation command area are RAM memory access.**

The contents will be reset when the power supply is turned off.

## 5.3 Message Frame Formats

The message frame format for ASCII and RTU transmission modes used in serial communication are described below.

### (1) Message frame format

#### (i) ASCII mode

##### A) Request (upper-level device such as PLC → Card Motor controller)

Start code	ID <sup>Note 1)</sup>	Function code	Data	Check sum	End code
1 byte	2 bytes	2 bytes	up to 56 bytes	2 bytes	2 bytes
:	"01"-“FF”	Function code	Depends on function	LRC	CR, LF

##### B) Response (Card Motor controller → upper-level device such as PLC)

#### ● Normal response

Start code	ID <sup>Note 1)</sup>	Function code <sup>Note 2)</sup>	Data	Check sum	End code
1 byte	2 bytes	2 bytes	n byte	2 bytes	2 bytes
:	"01"-“FF”	Function code	Depends on function	LRC	CR,LF

#### ● Irregular response

Start code	ID <sup>Note 1)</sup>	Error function code	Details of error	Check sum	End code
1 byte	2 bytes	2 bytes	2 bytes	2 bytes	2 bytes
:	"01"-“FF”	The MSB of the function code is changed to 1. <sup>Note 3)</sup>	Problems code <sup>Note 4)</sup>	LRC	CR, LF

Note 1: This is the controller ID set in the Card Motor controller (default setting = 1).

ID configuration example)

ID 1: “01h”

ID16: “10h”

Note 2: Refer to sections “5.3 Message Frame Formats (4) Function Codes” and “5.4 Details of Function Codes Supported by LATCA” for details.

Note 3: The MSB (Most Significant Bit) of the received function code in the request is changed to 1 in the response.

Example)

Received function code: “03” (0000 0011) ASCII hex code 30h, 33h

Incorrect function code: “83” (1000 0011) ASCII hex code 38h, 33h

Note 4: Refer to section “5.8 Error Codes” for details.

(ii) RTU mode

A) Request (upper-level device such as PLC → Card Motor controller)

Start	ID <sup>Note 2)</sup>	Function code	Data	Check sum	End
T1-T2-T3-T4	1 byte	1 byte	up to 28 byte	2 bytes	T1-T2-T3-T4
3.5 character times <sup>Note 1)</sup>	01 – FF	Function code	Depends on function	CRC	3.5 character times <sup>Note 1)</sup>

B) Response (Card Motor controller → upper-level device such as PLC)

● Normal response

Start	ID <sup>Note 2)</sup>	Function code <sup>Note 3)</sup>	Data	Check sum	End
T1-T2-T3-T4	1 byte	1 byte	n byte	2 bytes	T1-T2-T3-T4
3.5 character times <sup>Note 1)</sup>	01 – FF	Function code	Depends on function	CRC	3.5 character times <sup>Note 1)</sup>

● Irregular response

Start	ID <sup>Note 2)</sup>	Error function code	Details of error	Check sum	End
T1-T2-T3-T4	1 byte	1 byte	1 byte	2 bytes	T1-T2-T3-T4
3.5 character times <sup>Note 1)</sup>	01 – FF	The MSB of the function code is changed to 1. <sup>Note 4)</sup>	Problems code <sup>Note 5)</sup>	CRC	3.5 character times <sup>Note 1)</sup>

Note 1: In Modbus RTU mode, message frames are separated by a silent interval (non-communication time). Provide a silent interval of at least 3.5 characters at the beginning and the end of the communication frame.

Note 2: This is the controller ID set in the Card Motor controller (default setting = 1).

ID configuration example)

ID 1: "01h"

ID16: "10h"

Note 3: Refer to sections "5.3 Message Frame Formats (4) Function Codes" and "5.4 Details of Function Codes Supported by LATCA" for details

Note 4: The MSB (Most Significant Bit) of the received function code in the request is changed to 1 in the response.

Example)

Received function code: 03 (0000 0011)

Incorrect function code: 83 (1000 0011)

Note 5: Refer to section "5.7 Error Codes" for details.

(2) Guard processing of received frames (applies to ASCII transmission mode only)

If the ASCII code is broken as a result of noise in the frame received and inaccurate data is mixed, an "NG" will be sent as response.

If only inaccurate data is received, by discarding of the received data, the frame received can be protected.

Inaccurate data refers to data other than the ASCII data below.

- (a) Alphanumeric characters (capital / small letters)
- (b) Special characters (BS, space, TAB, comma, full stop, hyphen)
- (c) Line-break code (CR + LF)

(3) ID

When communication is established between the Card Motor controller and upper level device such as a PLC, a request will be received and a response will be sent only when the pre-set Card Motor controller ID matches with the controller ID in the requested message frame.

When the controller ID in the requested message frame is "0", it will be received as a broadcast request<sup>Note 1</sup>). In this case, no response will be made.

When the controller ID in the requested message frame is not "0" and is different from the controller ID set in the Card Motor controller, the requested message frame will be deleted and no response will be made.

Note 1: Broadcast requests

Requests can be sent simultaneously to all Card Motor controllers connected to an upper level device such as a PLC. This is used for Step Data setting, SVON signal ON/OFF and Return to Origin.

 **Caution**

**Use unique controller IDs for each controller within the same serial communication network.**

Duplication of the same IDs may result in interference in response data.



#### (4) Function codes

Function codes and functions supported by this controller are summarized in the table below.

Refer to section "5.4 Details of Function Codes Supported by LATCA" for further details of each code.

Function code	Item	Function	Note
01h	Read status of output signals (Y)	Read Y contacts <sup>Note 1)</sup>	Broadcast is not available.
02h	Read status of input signals (X)	Read X contacts <sup>Note 2)</sup>	Broadcast is not available.
03h	Read data contents of holding register addresses (D)	Read Step Data parameters <sup>Note 3)</sup> and status data <sup>Note 4)</sup>	Broadcast is not available. Reading of X and Y contacts is not available.
05h	Write to (force) a single output signal (Y)	Write a single Y contact <sup>Note 1)</sup>	Broadcast is available. (No response is made.)
06h	Write data to a single holding register address (D)	Writing of a single parameter <sup>Note 3)</sup> and data <sup>Note 4)</sup>	Broadcast is available. (No response is made.) Y contact wiring is not available.
08h	Diagnostics	Communication test using echo back	Broadcast is not available.
0Fh	Write to (force) multiple output signals (Y)	Writing of multiple Y contacts <sup>Note 1)</sup>	Broadcast is available. (No response is made.)
10h	Write data to multiple holding register addresses (D)	Writing of multiple parameters <sup>Note 3)</sup> and data <sup>Note 4)</sup>	Broadcast is available. (No response is made.) Y contact wiring is not available.

Note 1: Y contacts indicate the status of the internal flags which can be read or written.

Refer to section "5.5 Memory Map (1) Internal Flags" for details of the internal flags.

Note 2: X contacts indicate the status of the internal flags which can be read.

Refer to section "5.5 Memory Map (1) Internal Flags" for details of the internal flags.

Note 3: Both Step Data parameters and Direct Operation parameters can be read.

Refer to section "5.5 Memory Map (2) Step Data" and "(3) Direct Operation Command" for details of the parameters.

Note 4: Each data shows the status data.

Refer to section "5.5 Memory Map (4) Status Data" for details of the status data.

#### (5) Data

Data corresponding to each function code (Maximum: 56 bytes).

#### (6) Check sum

Error-check code for the message frame. As the communication protocol of the controller is Modbus, the checksum calculation methods are also compliant with the Modbus protocol. The checksum will be calculated using the LRC method (for ASCII mode) or CRC method (for RTU mode).

When transmitting a message frame, the checksum value will be calculated and added to the transmitted message frame. When receiving the message frame, the checksum value of the received message frame will be calculated and compared with the checksum value added to the received message frame. If the checksum of the transmitted message frame does not match with that of the received message frame, no response will be made.

## 5.4 Details of Function Codes supported by LATCA

The message frame format described in this Operation Manual is indicated in Modbus ASCII. When the protocol is used in Modbus RTU, input the silent interval instead of the start and end codes. The check sum should be calculated in CRC. Refer to section "5.3 Message Frame Formats" for details.

### (1) 01h Read the status of output signals

Function code used to read the status of the output signals (internal flags / Y contacts) from the master to the slave.

(e.g.) Read 16 bits of Y10 to Y1F from a slave controller with controller ID: 1.

Controller ID: 1

Function code: 01h

Start address: Y10

Number of read outputs: 16 bits

Output signal status: Y1F to Y10 = 0000 0110 0000 0000 (SVON and DRIVE = 1, Others = 0)

### (i) Request

		Example	
1	Start code		" : "
2	ID		"0"
3			"1"
4	Function code		"0"
5			"1"
6	Data	Start reading from address No.	"0"
7			"0"
8			"1"
9		Number of outputs to be read <sup>Note 1)</sup> (Number of bits)	"0"
10			"0"
11			"0"
12	"1"		
13	"0"		
14	Checksum(LRC)		Calculated value
15			Calculated value
16	End code		"CR"
17			"LF"

Controller ID: 1 (rows 2-3)  
 Function code: 01h (rows 4-5)  
 Start address: Y10 (rows 6-8)  
 Number of read outputs: 16 bits (10h) (rows 9-13)

Note 1: The maximum value is 456 bits. Value exceeding the maximum number will lead to a data error.

(ii) Normal response

		Example	
1	Start code		" : "
2	ID		"0"
3			"1"
4	Function code		"0"
5			"1"
6	Data	Number of bytes in response data	"0"
7			"2"
8		Data 1	"0"
9			"0"
10	Data 2	"0"	
11		"6"	
12	Checksum(LRC)		Calculated value
13			Calculated value
14	End code		"CR"
15			"LF"

Controller ID: 1  
 Function code: 01h  
 Number of response data bytes = 2 bytes  
 Y contact status  
 Data 1 = "0000 0000"  
 Data 2 = "0000 0110"

Details of response data

Data 1	Y17	Y16	Y15	Y14	Y13	Y12	Y11	Y10
	0	0	0	0	0	0	0	0
Data 2	Y1F	Y1E	Y1D	Y1C	Y1B	Y1A	Y19	Y18
	0	0	0	0	0	1	1	0

(iii) Irregular response

		Example	
1	Start code		" : "
2	ID		"0"
3			"1"
4	Function code		"8"
5			"1"
6	Error code		"0"
7			"1"
8	Checksum(LRC)		Calculated value
9			Calculated value
10	End code		"CR"
11			"LF"

Controller ID: 1  
 Function code & MSB + 1 = "81h"  
 Error code = "01h"

(2) 02h Read the status of input signals

Function code used to read the status of the input signals (internal flags / X contacts) from the slave to the master.

(e.g.) Read 16 bits of X40 to X4F from a slave controller with controller ID: 1.

Controller ID: 1

Function code: 02h

Start address: X40

Number of read outputs: 16 bits

Input signal status: X4F to X40 = 1000 1110 0000 0100

(Active Step Data No. = 4, SVRE, SETON, INP, ALARM = 1, Others = 0)

(i) Request

		Example	
1	Start code		"."
2	ID		"0"
3			"1"
4	Function code		"0"
5			"2"
6	Data	Start reading from address No.	"0"
7			"0"
8			"4"
9		Number of inputs to be read <sup>Note 1)</sup> (Number of bits)	"0"
10			"0"
11			"0"
12	"1"		
13	"0"		
14	Checksum(LRC)		Calculated value
15			Calculated value
16	End code		"CR"
17			"LF"

Controller ID: 1 (rows 2-3)

Function code: 02h (rows 4-5)

Start address: X40 (rows 6-8)

Number of read outputs: 16 bits (10h) (rows 9-13)

Rows 2-3: Controller ID: 1

Rows 4-5: Function code: 02h

Rows 6-8: Start address: X40

Rows 9-13: Number of read outputs: 16 bits (10h)

Note 1: The maximum value is 456 bits. Value exceeding the maximum number will lead to a data error.

(ii) Normal response

		Example	
1	Start code		" :
2	ID		"0"
3			"1"
4	Function code		"0"
5			"2"
6	Data	Number of bytes in response data	"0"
7			"2"
8		Data 1	"0"
9			"4"
10		Data 2	"8"
11	"E"		
12	Checksum(LRC)		Calculated value
13			Calculated value
14	End code		"CR"
15			"LF"

Controller ID: 1

Function code: 02h

Number of response data bytes = 2 bytes

X contact status

Data 1 = "0000 0100"

Data 2 = "1000 1110"

Details of response data

Data 1	X47	X46	X45	X44	X43	X42	X41	X40
	0	0	0	0	0	1	0	0
Data 2	X4F	X4E	X4D	X4C	X4B	X4A	X49	X48
	1	0	0	0	1	1	1	0

(iii) Irregular response

		Example	
1	Start code		" :
2	ID		"0"
3			"1"
4	Function code		"8"
5			"2"
6	Error code		"0"
7			"1"
8	Checksum(LRC)		Calculated value
9			Calculated value
10	End code		"CR"
11			"LF"

Controller ID: 1

Function code & MSB + 1 = "82h"

Error code = "01h"

(3) 03h Read data contents of holding register addresses

Function code used to read Step Data and internal parameters and alarm history.

(e.g.) Read data contained in holding register addresses D0600 to D0603 from a slave controller with controller ID: 1 (reading 4 alarm histories)

Controller ID: 1

Function code: 03h

Start address: D0600

Number of read addresses: 4 addresses (1 address: 2 bytes = 4 words)

Alarm history 1 = Temperature error

Alarm history 2 = Return to Origin position non-execution error

Alarm history 3 = Origin parameter error

Alarm history 4 = No alarm

(i) Request

		Example	
1	Start code		" :
2	ID		"0"
3			"1"
4	Function code		"0"
5			"3"
6	Data	Start reading from address No.	"0"
7			"6"
8			"0"
9			"0"
10		Number of addresses to be read <sup>Note 1)</sup> (Number of words)	"0"
11			"0"
12			"0"
13			"4"
14	Checksum(LRC)		Calculated value
15			Calculated value
16	End code		"CR"
17			"LF"

Controller ID: 1

Function code: "03h"

Start address: D0600

Number of read addresses: 4 words

Note 1: The maximum value is 28 words. Value exceeding the maximum number will lead to a data error.

(ii) Normal response

		Example		
1	Start code		" : "	
2	ID		"0"	
3			"1"	
4	Function code		"0"	
5			"3"	
6	Data	Number of bytes in response data	"0"	
7			"8"	
8		Alarm 1 (Temperature error: 0003)	"0"	
9			"0"	
10			"0"	
11			"3"	
12		Alarm 2 (Return to Origin position non-execution error: 000B)	"0"	
13			"0"	
14			"0"	
15			"B"	
16		Alarm 3 (Origin parameter error: 0006)	"0"	
17			"0"	
18			"0"	
19			"6"	
20		Alarm 4 (No alarm: 0000)	"0"	
21			"0"	
22			"0"	
23			"0"	
24		Checksum(LRC)		Calculated value
25				Calculated value
26		End code		"CR"
27				"LF"

Controller ID: 1

Function code: 03h

Byte count of response data = 8 bytes

Alarm history 1

Alarm history 2

Alarm history 3

Alarm history 4

(iii) Irregular response

		Example	
1	Start code		" : "
2	ID		"0"
3			"1"
4	Function code		"8"
5			"3"
6	Error code		"0"
7			"1"
8	Checksum(LRC)		Calculated value
9			Calculated value
10	End code		"CR"
11			"LF"

Controller ID: 1

Function code & MSB + 1 = "83h"

Error code = "01h"

(4) 05h Write (force) a single output signal

Function code used to write (force) a single Y contact (output signal) to either ON or OFF. A value of "FF00h" requests the output to be ON. A value of "0000h" requests it to be OFF. (When writing all data simultaneously to multiple Y contacts, use Function code 0Fh.)

(e.g.) Turn ON Y30 in the controller with controller ID: 1

(Command to change the operation mode to serial communication).

Controller ID: 1

Function code: 05h

Start address: Y30

Data value to be written: "FF00h"

(i) Request

		Example		
1	Start code		" : "	
2	ID		"0"	
3			"1"	
4	Function code		"0"	
5			"5"	
6	Data		Address No. of contact to be written	"0"
7				"0"
8				"3"
9				"0"
10	Data value to be written <sup>Note 1)</sup>			"F"
11				"F"
12				"0"
13				"0"
14	Checksum(LRC)		Calculated value	
15			Calculated value	
16	End code		"CR"	
17			"LF"	

Controller ID: 1 (rows 2-3)  
 Function code: 05h (rows 4-5)  
 Start address: Y30 (rows 6-9)  
 Data value to be written: "FF00h" (rows 10-13)

Note 1) Set the data to either ON: FF00h or OFF: 0000h.



(ii) Normal response

		Example	
1	Start code		" : "
2	ID		"0"
3			"1"
4	Function code		"0"
5			"5"
6	Data	Response address No.	"0"
7			"0"
8			"3"
9		Written data	"0"
10			"F"
11	"F"		
12	"0"		
13		"0"	
14	Checksum(LRC)		Calculated value
15			Calculated value
16	End code		"CR"
17			"LF"

Controller ID: 1

Function code: 05h

Response address: Y30

Written data = "FF00h"

(iii) Irregular response

		Example	
1	Start code		" : "
2	ID		"0"
3			"1"
4	Function code		"8"
5			"5"
6	Error code		"0"
7			"1"
8	Checksum(LRC)		Calculated value
9			Calculated value
10	End code		"CR"
11			"LF"

Controller ID: 1

Function code & MSB + 1 = "85h"

Error code = "01h"

(5) 06h Write data to a single holding register address

Function code used to set a single Step Data parameter.

(When setting multiple Step Data parameters, use Function code 10h.)

(e.g.) Write "000Ah" to holding register address D0412 in the controller with controller ID: 1

(Set the positioning time for Step Data 1 to 0.1 s).

Controller ID: 1

Function code: 06h

Start address: D0412

Data value to be written: "000Ah"

(i) Request

		Example	
1	Start code		"."
2	ID		"0"
3			"1"
4	Function code		"0"
5			"6"
6	Data	Address No. of holding register to be written	"0"
7			"4"
8			"1"
9		"2"	
10	Data	Data to be written	"0"
11			"0"
12			"0"
13			"A"
14	Checksum(LRC)		Calculated value
15			Calculated value
16	End code		"CR"
17			"LF"

Controller ID: 1

Function code: 06h

Start address: D0412

Data value to be written: "000Ah"

Controller ID: 1

Function code: 06h

Start address: D0412

Data value to be written: "000Ah"

(ii) Normal response

		Example	
1	Start code		" : "
2	ID		"0"
3			"1"
4	Function code		"0"
5			"6"
6	Data	Address No. of response holding register	"0"
7			"4"
8			"1"
9		Written data	"0"
10			"0"
11			"0"
12	"0"		
13		"A"	
14	Checksum(LRC)		Calculated value
15			Calculated value
16	End code		"CR"
17			"LF"

Controller ID: 1

Function code: 06h

Response address: D0410

Written data: "000Ah"

(iii) Irregular response

		Example	
1	Start code		" : "
2	ID		"0"
3			"1"
4	Function code		"8"
5			"6"
6	Error code		"0"
7			"1"
8	Checksum(LRC)		Calculated value
9			Calculated value
10	End code		"CR"
11			"LF"

Controller ID: 1

Function code & MSB + 1 = "86h"

Error code: "01h"

(6) 08h Diagnostics of communication using echo back

Function code used to test proper communication by echo back

(e.g.) Perform an echo back test for the controller with controller ID:1.

Controller ID: 1

Function code: 08h

Test code: "0000h"

Test data value: "1234h"

(i) Request

			Example	
1	Start code		" :	
2	ID		"0"	} Controller ID: 1
3			"1"	
4	Function code		"0"	} Function code: 08h
5			"8"	
6	Data	Test code <sup>Note 1)</sup>	"0"	} Test code: "0000h"
7			"0"	
8			"0"	
9		"0"		
10	Data	Test data <sup>Note 2)</sup>	"1"	} Test data value: "1234h"
11			"2"	
12			"3"	
13			"4"	
14	Check sum(LRC)		Calculated value	
15			Calculated value	
16	End code		"CR"	
17			"LF"	

Note 1: Specified test code is Modbus sub-function: 0000h.

Note 2: Test data is the value requested to be returned (looped back) in the response. The entire response message should be identical to the request.

(ii) Normal response

		Example	
1	Start code		" : "
2	ID		"0"
3			"1"
4	Function code		"0"
5			"8"
6	Data	Returned test code	"0"
7			"0"
8			"0"
9			"0"
10		Received data	"1"
11			"2"
12			"3"
13			"4"
14	Checksum(LRC)		Calculated value
15			Calculated value
16	End code		"CR"
17			"LF"

Controller ID: 1

Function code: 08h

Test code: "0000h"

Response data = "1234h"

(iii) Irregular response

		Example	
1	Start code		" : "
2	ID		"0"
3			"1"
4	Function code		"8"
5			"8"
6	Error code		"0"
7			"1"
8	Checksum(LRC)		Calculated value
9			Calculated value
10	End code		"CR"
11			"LF"

Controller ID: 1

Function code & MSB + 1 = "88h"

Error code: "01h"

(7) 0Fh Write (force) multiple output signals

Function code used to write (force) multiple Y contacts (output signals) to either ON or OFF.

(e.g.) Write the following data to Y10 to Y1F of the controller with controller ID: 1

(Turn ON Y contacts that instigate an operation command for Step Data 1).

Controller ID: 1

Function code: 0Fh

Start address: Y10

Data to be written: 0000 0110 0000 0001 (IN0, SVON, DRIVE = 1, others = 0)

Number of outputs to be written: 16 bits (2 bytes)

(i) Request

		Example	
1	Start code		"."
2	ID		"0"
3			"1"
4	Function code		"0"
5			"F"
6	Data	Start writing from address No.	"0"
7			"0"
8			"1"
9			"0"
10		Number of outputs to be written <sup>Note 1)</sup> (Number of bits)	"0"
11			"0"
12			"1"
13		Number of written data (bytes)	"0"
14			"0"
15			"2"
16	Data to be written 1	"0"	
17		"1"	
18		"0"	
19	Data to be written 2	"6"	
20	Checksum(LRC)		Calculated value
21			Calculated value
22	End code		"CR"
23			"LF"

} Controller ID: 1

} Function code: 0Fh

} Start address: Y10

} Number of written outputs = 16 bits (10h)

} Number of written data = 2 bytes

} Data to be written  
Data 1 = "0000 0001"  
Data 2 = "0000 0110"

Note 1: The maximum value is 184 bits (23 bytes). Value exceeding the maximum number will lead to a data error.

Details of the written data

Data 1	Y17	Y16	Y15	Y14	Y13	Y12	Y11	Y10
	0	0	0	0	0	0	0	1
Data 2	Y1F	Y1E	Y1D	Y1C	Y1B	Y1A	Y19	Y18
	0	0	0	0	0	1	1	0

(ii) Normal response

		Example	
1	Start code		" : "
2	ID		"0"
3			"1"
4	Function code		"0"
5			"F"
6	Data	Writing start address No.	"0"
7			"0"
8			"1"
9		Number of written outputs (bits)	"0"
10			"0"
11			"0"
12		"1"	
13		"0"	
14	Checksum(LRC)		Calculated value
15			Calculated value
16	End code		"CR"
17			"LF"

Controller ID: 1

Function code: "0Fh"

Writing start address = Y10

Number of written outputs: 16

(iii) Irregular response

		Example	
1	Start code		" : "
2	ID		"0"
3			"1"
4	Function code		"8"
5			"F"
6	Error code		"0"
7			"1"
8	Checksum(LRC)		Calculated value
9			Calculated value
10	End code		"CR"
11			"LF"

Controller ID: 1

Function code & MSB + 1 = "8Fh"

Error code: "01h"

(8) 10h Write data to multiple holding register addresses

Function code used to set multiple Step Data parameters, for changing to serial operation mode and to clear alarm history.

(e.g.) Controller ID=1 Address D0410 to D0411 Write "0000h" , "2710h" (set the target position of the step data No.1 to 10 mm).

Function code: 10h

Start address: D0410

Data value to be written: "0000h", "2710h"

Number of written addresses: 2 addresses (2 words), 4 bytes

(i) Request

			Example	
1	Start code		" : "	
2	ID		"0"	Controller ID: 1
3			"1"	
4	Function code		"1"	Function code: "10h"
5			"0"	
6	Data	Start writing from address No.	"0"	Start address: D0410
7			"4"	
8			"1"	
9			"0"	
10	Number of addresses to be written <sup>Note 1)</sup> (Number of words)	"0"	Number of written addresses = 2 words	
11		"0"		
12		"0"		
13	Number of written data (bytes)	"2"	Number of written data = 4 bytes	
14		"0"		
15		"4"		
16	Data to be written 1	"0"	Data to be written to first register Target position of Step Data No.1 (upper 2 bytes) = "0000h"	
17		"0"		
18		"0"		
19		"0"		
20	Data to be written 2	"2"	Data to be written to fourth register Target position of Step Data No.1 (lower 2 bytes) = "2710h"	
21		"7"		
22		"1"		
23		"0"		
24	Checksum(LRC)		Calculated value	
25			Calculated value	
26	End code		"CR"	
27			"LF"	

Note 1: The maximum value is 11 words. Value exceeding the maximum number will lead to a data error.

(ii) Normal response



		Example	
1	Start code		" : "
2	ID		"0"
3			"1"
4	Function code		"1"
5			"0"
6	Data	Writing start address No.	"0"
7			"4"
8			"1"
9		"0"	
10		Number of written points addresses (words)	"0"
11			"0"
12	"0"		
13	"2"		
14	Checksum(LRC)		Calculated value
15			Calculated value
16	End code		"CR"
17			"LF"

Controller ID: 1

Function code: "10h"

Writing start address = D0410

Number of written addresses = 2

(iii) Irregular response

		Example	
1	Start code		" : "
2	ID		"0"
3			"1"
4	Function code		"9"
5			"0"
6	Error code		"0"
7			"1"
8	Checksum(LRC)		Calculated value
9			Calculated value
10	End code		"CR"
11			"LF"

Controller ID: 1

Function code & MSB + 1 = "90h"

Error code: "01h"

## 5.5 Memory Map

The memory map for the Card Motor controller is summarized in the tables below.

Refer to section “5.6. List of Request Functions” for more detailed information.

### **Caution**

**Use active addresses and flags only. Do not use non-defined or unavailable addresses or flags.**

Otherwise, the Card Motor may make an unexpected movement, leading to damage.

If an address or flag has been written in error or the Card Motor makes an unexpected movement, initialize the LATCA controller using the controller configuration software to reset the controller to factory default settings.

#### (1) Internal flags

##### (i) Internal status flags (X contacts / Master inputs)

The status of the internal process of the controller is indicated as follows: When a flag is ON, "1" will be read, and when a flag is OFF, "0" will be read.

Flag name		Read	Write	Description
X40	OUT0	Available	Unavailable	Step Data number currently being executed
X41	OUT1	Available	Unavailable	
X42	OUT2	Available	Unavailable	
X43	OUT3	Available	Unavailable	
X44	-	Available	Unavailable	Cannot be used.
X45	-	Available	Unavailable	Cannot be used.
X46	INF	Available	Unavailable	“INF” signal output flag
X47	INFP	Available	Unavailable	“INFP” signal output flag
X48	BUSY	Available	Unavailable	“BUSY” signal output flag
X49	SVON	Available	Unavailable	Servo ON status flag
X4A	SETON	Available	Unavailable	Return to Origin completion flag
X4B	INP	Available	Unavailable	“INP” signal output flag
X4C	AREA_A	Available	Unavailable	“AREA_A” signal output flag
X4D	AREA_B	Available	Unavailable	“AREA_B” signal output flag
X4E	-	Available	Unavailable	Cannot be used.
X4F	ALARM	Available	Unavailable	“ALARM” signal output flag
X50	OVC	Available	Unavailable	“OVC” signal output flag
X51	OVT	Available	Unavailable	“OVT” signal output flag

(ii) Internal editable status flags (Y contacts / Master outputs)

In serial operation mode, these flags can be set and read.

In parallel I/O operation mode, parallel I/O signal input status can only be read.

"1" indicates signal status ON, and "0" indicates signal status OFF.

Flag name		Read	Write	Description
Y10	IN 0	Available	Available	In serial operation mode - When reading: Command status is read. - When writing: Command is sent to the controller.
Y11	IN 1	Available	Available	
Y12	IN 2 <sup>Note 2)</sup>	Available	Available	In parallel I/O operation mode - When reading: Signal input status is read. - When writing: Invalid.
Y13	IN 3 <sup>Note 2)</sup>	Available	Available	
Y14	-	Available	Available	Cannot be used. <sup>Note 4)</sup>
Y15	-	Available	Available	
Y16	-	Available	Available	
Y17	-	Available	Available	
Y18	-	Available	Available	
Y19	SVON	Available	Available	In serial operation mode - When reading: Command status is read. - When writing: Command is sent to the controller.
Y1A	DRIVE <sup>Note 2)</sup>	Available	Available	In parallel I/O operation mode - When reading: Signal input status is read. - When writing: Invalid.
Y1B	-	Available	Available	Cannot be used. <sup>Note 4)</sup>
Y1C	SETUP <sup>Note 1)</sup>	Available	Available	In serial operation mode - When reading: Invalid. - When writing: Invalid.
Y1D	CLR <sup>Note 1)</sup>	Available	Available	
Y1E	TL <sup>Note 1)</sup>	Available	Available	In parallel I/O operation mode - When reading: Signal input status is read. - When writing: Invalid.
Y1F	-	Available	Available	Cannot be used. <sup>Note 4)</sup>
Y30	Operation mode flag <sup>Note 3)</sup>	Available	Available	0: Parallel I/O operation mode 1: Serial operation mode

Note 1: The value is always "0" when the LATCA controller is in Step Data Input Type control signal mode. The controller will not execute these operations even if "1" is written (turned ON) when the LATCA controller is in Pulse Input Type control signal mode.

Note 2: The value is always "0" when the LATCA controller is in Pulse Input Type control signal mode.

Note 3: The operation mode can be switched between parallel I/O / serial with Y30.

When Y30 is commanded from "0" to "1", all the status change flags will be reset to "0".

When Y30 is commanded from "1" to "0", the status of the parallel input terminals will be reflected immediately.

Note 4: When data is written to multiple flags (contacts) simultaneously, write "0" to the addresses which cannot be used.

## (2) Step Data Operation

## (i) Holding register addresses and contents of parameters for Step Data Operation

Step Data No.	Address	Number of bytes	Description	Configuration unit	Value range	For Pulse Input Mode
No. 1 (No. 0) <sup>Note 5)</sup>	D0410	4	Target position [ $\mu\text{m}$ ] <sup>Note 1)</sup>	1	Note 1)	No
	D0412	2	Positioning time [0.01s]	1	0 to 6000	No
	D0413	2	Speed [mm/s]	1	0 to 400	No
	D0414	2	Acceleration [ $\text{mm/s}^2$ ]	1	0 to 60000	No
	D0415	2	Deceleration [ $\text{mm/s}^2$ ]	1	0 to 60000	No
	D0416	2	Pushing speed [mm/s] <sup>Note2)</sup>	1	1 to 20, 32768 to 32788	No
	D0417	2	Thrust setting value <sup>Note 3)</sup> [0.1]	1	More than 10 <sup>Note 3)</sup>	Yes
	D0418	2	Load mass [10g]	5	0 , 5 , 10 to 100	Yes
	D0419	2	Movement mode (0: ABS, 1: REL)	1	0 , 1	No
	D041A	2	Threshold force value [0.1]	1	1 to 50	Yes
	D041B	2	Positioning width [ $\mu\text{m}$ ]	1	0 to 30000	Yes
	D041C	4	AREA A position 1 [ $\mu\text{m}$ ] <sup>Note1)</sup>	1	More than 0 <sup>Note 1)</sup>	Yes
	D041E	4	AREA A position 2 [ $\mu\text{m}$ ] <sup>Note1)</sup>	1	More than 0 <sup>Note 1)</sup>	Yes
	D0420	4	AREA B position 1 [ $\mu\text{m}$ ] <sup>Note1)</sup>	1	More than 0 <sup>Note 1)</sup>	Yes
	D0422	4	AREA B position 2 [ $\mu\text{m}$ ] <sup>Note1)</sup>	1	More than 0 <sup>Note 1)</sup>	Yes
D0424 to D042F	-	Writing is not available.	-	-	-	
No. 2 (No. 1) <sup>Note 5)</sup>	D0430 to D044F	Same as above				
- - -	- - -					
No. 15	D05D0 to D05EF	Same as above				No

Note 1: The maximum value is the same as for the Card Motor maximum stroke [ $\mu\text{m}$ ].

(e.g. Maximum value for the LAT3-10: 10000 [ $\mu\text{m}$ ])

The minimum values vary depending on the movement mode as described below.

Minimum value in REL operation: Card Motor stroke [ $\mu\text{m}$ ] x -1

(e.g. Minimum value for the LAT3-10: -10000  $\mu\text{m}$ )

Minimum value in ABS operation: 0

Note 2: Refer to the following examples for how to set the pushing speed.

For "Pushing Speed" use 32768 as base value, and add speed in [mm/s]. If no value or a value equal to or less than 32768 is input, the pushing operation will not be performed, even if a thrust setting value has been input.

- Step Data No. 1 (in Pulse Input Mode, corresponding Step Data is No. 0):

The controller uses the pushing speed value set in Step Data No. 1 (Step Data No. 0) as the speed value for the Return to Origin operation.

(Example) When Step Data No. 1 is used for a positioning operation:

Recommended set value

Set value for pushing speed = **6** (Thus, positioning speed of Return to Origin: 6 mm/s,  
Pushing speed for Step Data No. 1 positioning operation: 0 mm/s)

(Example) When Step Data No. 1 (Step Data No. 0) is used for a pushing operation and the speed is set to 6 mm/s:

Recommended set value

Set value for pushing speed = **32774** (Thus, Return to Origin speed and Pushing speed for Step Data No.1 (Step Data No. 0) pushing operation: 6 mm/s)

### **Caution**

**Do not set pushing speed value for Step Data No. 1 (Step Data No. 0 in Pulse Input Mode) to "0" or "32768".**

If those values are input, the Return to Origin operation cannot be performed as the speed becomes 0 mm/s.

- Step Data No. 2 and higher (in Pulse Input Mode, corresponding Step Data is No. 1 and higher):

(Example) For positioning operation:

Recommended set value

Set value for pushing speed = **32768** (Thus, pushing speed: 0 mm/s)

(Example) Pushing operation with set speed of 6 mm/s:

Recommended set value

Set value for pushing speed = **32774** (Pushing speed: 6 mm/s)

Note 3: Maximum values differ depending on the Card Motor model.

(LAT3\*-10: 5.0, LAT3\*-20: 4.8, LAT3\*-30: 3.9, LAT3\*-50: 2.0)

Note 4: If the target position exceeds the stroke range, the Card Motor operation speed is set to above 400 mm/s or any other impossible value is set, "Step Data error" will be generated from the Card Motor controller during the Step Data operation. Please set values as appropriate.

Note 5: Step Data numbers in brackets are valid in Pulse Input Mode.

Step Data No. 3 is the highest valid Step Data No. in Pulse Input Mode.

## (ii) Holding register address list for each Step Data No. and parameter.

Item	STEP 1 (STEP 0)	STEP 2 (STEP 1)	STEP 3 (STEP 2)	STEP 4 (STEP 3)	STEP 5	STEP 6	STEP 7	STEP 8
Target position	D0410	D0430	D0450	D0470	D0490	D04B0	D04D0	D04F0
	D0411	D0431	D0451	D0471	D0491	D04B1	D04D1	D04F1
Positioning time	D0412	D0432	D0452	D0472	D0492	D04B2	D04D2	D04F2
Speed	D0413	D0433	D0453	D0473	D0493	D04B3	D04D3	D04F3
Acceleration	D0414	D0434	D0454	D0474	D0494	D04B4	D04D4	D04F4
Deceleration	D0415	D0435	D0455	D0475	D0495	D04B5	D04D5	D04F5
Pushing speed	D0416	D0436	D0456	D0476	D0496	D04B6	D04D6	D04F6
Thrust setting value	D0417	D0437	D0457	D0477	D0497	D04B7	D04D7	D04F7
Load Mass	D0418	D0438	D0458	D0478	D0498	D04B8	D04D8	D04F8
Movement mode	D0419	D0439	D0459	D0479	D0499	D04B9	D04D9	D04F9
Threshold force value	D041A	D043A	D045A	D047A	D049A	D04BA	D04DA	D04FA
Positioning width	D041B	D043B	D045B	D047B	D049B	D04BB	D04DB	D04FB
AREA A position 1	D041C	D043C	D045C	D047C	D049C	D04BC	D04DC	D04FC
	D041D	D043D	D045D	D047D	D049D	D04BD	D04DD	D04FD
AREA A position 2	D041E	D043E	D045E	D047E	D049E	D04BE	D04DE	D04FE
	D041F	D043F	D045F	D047F	D049F	D04BF	D04DF	D04FF
AREA B position 1	D0420	D0440	D0460	D0480	D04A0	D04C0	D04E0	D0500
	D0421	D0441	D0461	D0481	D04A1	D04C1	D04E1	D0501
AREA B position 2	D0422	D0442	D0462	D0482	D04A2	D04C2	D04E2	D0502
	D0423	D0443	D0463	D0483	D04A3	D04C3	D04E3	D0503
Writing is not available.	D0424	D0444	D0464	D0484	D04A4	D04C4	D04E4	D0504
	to	to	to	to	to	to	to	to
	D042F	D044F	D046F	D048F	D04AF	D04CF	D04EF	D050F

Item	STEP 9	STEP 10	STEP 11	STEP 12	STEP 13	STEP 14	STEP 15
Target position	D0510	D0530	D0550	D0570	D0590	D05B0	D05D0
	D0511	D0531	D0551	D0571	D0591	D05B1	D05D1
Positioning time	D0512	D0532	D0552	D0572	D0592	D05B2	D05D2
Speed	D0513	D0533	D0553	D0573	D0593	D05B3	D05D3
Acceleration	D0514	D0534	D0554	D0574	D0594	D05B4	D05D4
Deceleration	D0515	D0535	D0555	D0575	D0595	D05B5	D05D5
Pushing speed	D0516	D0536	D0556	D0576	D0596	D05B6	D05D6
Thrust setting value	D0517	D0537	D0557	D0577	D0597	D05B7	D05D7
Load mass	D0518	D0538	D0558	D0578	D0598	D05B8	D05D8
Movement mode	D0519	D0539	D0559	D0579	D0599	D05B9	D05D9
Threshold force value	D051A	D053A	D055A	D057A	D059A	D05BA	D05DA
Positioning width	D051B	D053B	D055B	D057B	D059B	D05BB	D05DB
AREA A position 1	D051C	D053C	D055C	D057C	D059C	D05BC	D05DC
	D051D	D053D	D055D	D057D	D059D	D05BD	D05DD
AREA A position 2	D051E	D053E	D055E	D057E	D059E	D05BE	D05DE
	D051F	D053F	D055F	D057F	D059F	D05BF	D05DF
AREA B position 1	D0520	D0540	D0560	D0580	D05A0	D05C0	D05E0
	D0521	D0541	D0561	D0581	D05A1	D05C1	D05E1
AREA B position 2	D0522	D0542	D0562	D0582	D05A2	D05C2	D05E2
	D0523	D0543	D0563	D0583	D05A3	D05C3	D05E3
Writing is not available.	D0524	D0544	D0564	D0584	D05A4	D05C4	D05E4
	to	to	to	to	to	to	to
	D052F	D054F	D056F	D058F	D05AF	D04CF	D05EF

(3) Direct operation command

(i) Command for executing Direct Operation

Address	Function	Number of registers (Bytes)	Description
D9100 (D9101)	Direct operation command	2	Starts the operation according to parameters set in "D9103" to "D9115". (The register value will be forced to "0000h" when the DRIVE flag Y1A status is "1".) "0000h": Position holding status "0100h": Starts Direct Operation
	Undefined	2	
D9102	Spare	2	-

(ii) Holding register addresses and contents of Step Data Parameters for Direct Operation

Step Data No.	Address	Number of registers (Bytes)	Description	Configuration unit	Value range	For Pulse Input Mode
No.20	D9103	4	Target position [ $\mu\text{m}$ ] <sup>Note 1)</sup>	1	Note 1)	No
	D9105	2	Positioning time [0.01s]	1	0 to 6000	No
	D9106	2	Speed [mm/s]	1	0 to 400	No
	D9107	2	Acceleration [mm/s <sup>2</sup> ]	1	0 to 60000	No
	D9108	2	Deceleration [mm/s <sup>2</sup> ]	1	0 to 60000	No
	D9109	2	Pushing speed [mm/s] <sup>Note2)</sup>	1	0 to 20 32768 to 32788	No
	D910A	2	Thrust setting value <sup>Note 3)</sup> [0.1]	1 to 1	More than 10 <sup>Note 3)</sup>	No
	D910B	2	Load mass [10g]	5	0, 5, 10 to 100	No
	D910C	2	Movement mode(0: ABS, 1: REL)	1	0, 1	No
	D910D	2	Threshold force value [0.1]	1	1 to 50	No
	D910E	2	Positioning width [ $\mu\text{m}$ ]	1	More than 0 <sup>Note 1)</sup>	No
	D910F	4	AREA A position 1 [ $\mu\text{m}$ ] <sup>Note1)</sup>	1	More than 0 <sup>Note 1)</sup>	No
	D9111	4	AREA A position 2 [ $\mu\text{m}$ ] <sup>Note1)</sup>	1	More than 0 <sup>Note 1)</sup>	No
	D9113	4	AREA B position 1 [ $\mu\text{m}$ ] <sup>Note1)</sup>	1	More than 0 <sup>Note 1)</sup>	No
	D9115	4	AREA B position 2 [ $\mu\text{m}$ ] <sup>Note1)</sup>	1	More than 0 <sup>Note 1)</sup>	No
D09117 to D0911F	-	-	Writing is not available.	-	-	-

Note 1: The maximum value is the same as for the Card Motor maximum stroke [ $\mu\text{m}$ ].

(e.g. Maximum value for the LAT3-10: 10000 [ $\mu\text{m}$ ])

The minimum values vary depending on the movement mode as described below.

Minimum value in REL operation: Card Motor stroke [ $\mu\text{m}$ ] x -1

(e.g. Minimum value for the LAT3-10: -10000  $\mu\text{m}$ )

Minimum value in ABS operation: 0

Note 2: Refer to the following examples for how to set the pushing speed.



For "Pushing Speed" use 32768 as base value, and add speed in [mm/s]. If no value or a value equal to or less than 32768 is input, the pushing operation will not be performed even if a "Thrust Setting Value" has been input.

- Step Data No. 1 (in Pulse Input Mode, corresponding Step Data is No. 0):

The controller uses the pushing speed value set in Step Data No. 1 (Step Data No. 0) as the speed value for the Return to Origin operation.

(Example) When Step Data No. 1 is used for a positioning operation:

Recommended set value

Set value for pushing speed = **6** (Thus, positioning speed of Return to Origin: 6 mm/s,  
Pushing speed for Step Data No. 1 positioning operation: 0 mm/s)

(Example) When Step Data No. 1 (Step Data No. 0) is used for a pushing operation and the speed is set to 6 mm/s:

Recommended set value

Set value for pushing speed = **32774** (Thus, Return to Origin speed and Pushing speed for Step Data No.1 (Step Data No. 0) pushing operation: 6 mm/s)

### **Caution**

**Do not set pushing speed value for Step Data No. 1 (Step Data No. 0 in Pulse Input Mode) to "0" or "32768".**

If those values are input, the Return to Origin operation cannot be performed as the speed becomes 0 mm/s.

- Step Data No. 2 and higher (in Pulse Input Mode, corresponding Step Data is No. 1 and higher):

(Example) For positioning operation:

Recommended set value

Set value for pushing speed = **32768** (Thus, pushing speed: 0 mm/s)

(Example) Pushing operation with set speed of 6 mm/s:

Recommended set value

Set value for pushing speed = **32774** (Pushing speed: 6 mm/s)

Note 3: Maximum values differ depending on the Card Motor model.

(LAT3\*-10: 5.0, LAT3\*-20: 4.8, LAT3\*-30: 3.9, LAT3\*-50: 2.0)

Note 4: If the target position exceeds the stroke range, the Card Motor operation speed is set to above 400 mm/s or any other impossible value is set, "Step Data error" will be generated from the Card Motor controller during the Step Data operation. Please set values as appropriate.

#### (4) Status data

The actuator condition (position, speed, value equivalent to thrust, target position, currently executed Step Data No.) are stored in addresses D9000 to D 9006.

Address	Function	Number of registers (Bytes)	Details
D9000	Positioning data	4	The current position data is sent as an encoder count value.
D9002	Speed data [mm/s]	2	Data (absolute value) of the current Card Motor positioning speed is sent. Direction is not taken into consideration.
D9003	Equivalent thrust value data	2	A value 10x that of the equivalent thrust force value is sent.
D9004	Target position	4	The current target position data is sent as an encoder count value.
D9006	Step Data number executed	2	Step Data number being executed is sent.

##### (a) Positioning data (encoder count value)

The current position data is sent as an encoder count value.

Count value 1,000,000 has been fixed for 0 mm as a reference point. The count decreases each time the position moves from the 0 mm to the opposite side to the connector. The encoder count value is converted into units [mm] by multiplying the count value by the encoder resolution of the used Card Motor model.

Binary digit (decimal value)	900,000 to 1,100,000
ASCII hex	"000DBBA0" to "0010C8E0"

Formula example) LAT3-10 (encoder resolution: 0.03 mm)

Data response: "000F418C" ⇒ 999,820 count

Card Motor table position: (1,000,000-999,820) x 0.03 mm = 5.4 mm

##### (b) Speed data [mm/s]

Data (absolute value) of the current Card Motor positioning speed is sent. Direction is not taken into consideration.

Binary digit (decimal value)	0 to 1000
ASCII hex	"0000" to "03E8"

##### (c) Equivalent thrust value data

A value 10x that of the equivalent thrust force value is sent.

Binary digit (decimal value)	0 to 50
ASCII hex	"00" to "32"

(d) Target position (encoder count value): This item is valid for Pulse Input Type control signal mode

only.

The target position data is sent as an encoder count value.

Count value 1,000,000 has been fixed for 0 mm as a reference point. The count decreases each time the position is moved from 0 mm position toward the opposite side of the connector. The encoder count value is converted into units [mm] by multiplying the count value by the encoder resolution of the used Card Motor model

Binary digit (decimal value)	900,000 to 1,100,000
ASCII hex	"000DBBA0" to "0010C8E0"

Formula example) LAT3-10 (encoder resolution: 0.03 mm)

Data response: "000F418C" ⇒ 999,820 count

Card Motor table position:  $(1,000,000 - 999,820) \times 0.03 \text{ mm} = 5.4 \text{ mm}$

(e) Step Data number executed

Step Data number being executed is sent.

Applicable numbers are as follows.

- For Step Data Input Type control signal mode

0: No Step Data executed

1-15: Step Data number currently executed

20: Direct operation

99: Return to Origin position

- For Pulse Input Type control signal mode

0-3: Step Data number currently executed

99: Return to Origin position

Binary digit (decimal value)	0 to 99
ASCII hex	"00" to "63"

## 5.6 List of Request Functions

Function of each function code supported by LATCA are described in the table below.

When the controller is set to 'Pulse Input Type' control signal mode, the available functions are limited.

○: Available, △: Partially available <sup>Note 1)</sup>, ×: Not available

Function	Function code to be used	Summary of function	For pulse input
Step Data reading	03h	Reading of Step Data parameters.	△
Step Data setting <small>Note 2, 3)</small>	06h / 10h	Setting of a single or all Step Data parameter(s)	△
Direct Operation Step Data reading	03h	Reading of Direct Operation Step Data parameters.	×
Direct Operation Step Data setting <small>Note 2, 3)</small>	06h / 10h	Setting of a single or all Direct Operation Step Data parameter(s).	×
Internal parameter reading	03h	Reading of Card Motor current position, speed, value equivalent to thrust, target position and Step Data number currently being executed.	○
Input signal status reading	01h	Reading of signal input status and operation mode flags.	○
Output signal status and internal flag reading	02h	Reading of Step Data number currently being executed and status of signal output flags	○
Operation mode selection <small>Note 4)</small>	05h / 0Fh	Operation mode can be switched between serial communication and parallel I/O.	○
Commands for virtual I/O operation by serial communication <small>Note 2) Note 4)</small>	05h / 0Fh	By writing to the Y contacts, the Step Data number, Card Motor energization ON/OFF and holding of current position / operation start can be performed.	×
Alarm history reading	03h	Reading of the alarm history stored in the controller.	○
Alarm history clearance	10h	Clearing of the alarm history stored in the controller.	○
Confirmation of communication connection	08 h	When communication is working correctly the data in the request is to returned (looped back) identically in the response.	○

Note 1: △: The number of Step Data and contents are partially limited in Pulse Input mode.

Refer to section "5.7 Details of Each Function, Step Data Setting / Reading, Direct Operation Step Data Setting / Reading" for details.

Note 2: As the Step Data parameters set using the Step Data setting function are stored in a non-volatile EEPROM, data is stored even when power is cycled. However, the Direct Operation Step Data set using the Direct Operation Step Data setting function need to be reset every time as the data is stored in volatile memory.

Note 3: Make sure to turn OFF the power to the Card Motor (SVON OFF) when setting Step Data. Otherwise, the Card Motor may perform unexpected operation. It is not required to turn off the power to the Card Motor (SVON OFF) when setting Direct Operation Step Data.

Note 4: After changing the contact Y30 to "1" (for serial operation) using the operation mode selection function, send virtual I/O operation commands using serial communication. When the contact Y30 is "0" (for parallel I/O operation), the controller does not operate even if a virtual I/O operation command is sent using serial communication.

## 5.7 Details of Functions

### (1) Step Data Setting / Reading, Direct Operation Step Data Setting / Reading

(i) Function

Setting (writing) and reading of Step Data Operation and Direct Operation Step Data parameters. Single or multiple data can be set and read.

(ii) Function codes to be used

03h / 06h / 10h

(iii) Memory Map

Refer to section "5.5 Memory Map" for a complete list of addresses for all Step Data numbers.

#### Step Data parameters

Step Data No.	Address	Number of registers (Bytes)	Description	Configuration unit	Value range	For pulse input mode
No. 1 (No. 0) Note 6)	D0410	4	Target position [ $\mu\text{m}$ ] <sup>Note 1)</sup>	1	Note 1)	No
	D0412	2	Positioning time [0.01s]	1	0 to 6000	No
	D0413	2	Speed [mm/s]	1	0 to 400	No
	D0414	2	Acceleration [ $\text{mm/s}^2$ ]	1	0 to 60000	No
	D0415	2	Deceleration [ $\text{mm/s}^2$ ]	1	0 to 60000	No
	D0416	2	Pushing speed [mm/s] <sup>Note2)</sup>	1	0 to 20 32768 to 32788	No
	D0417	2	Thrust setting value [0.1] <sup>Note 3)</sup>	1	More than 10 <sup>Note 3)</sup>	Yes
	D0418	2	Load mass [10g]	5	0, 5, 10 to 100	Yes
	D0419	2	Movement mode(0: ABS, 1: REL)	1	0, 1	No
	D041A	2	Threshold force value [0.1]	1	1 to 50	Yes
	D041B	2	Positioning width [ $\mu\text{m}$ ]	1	More than 0 <sup>Note 1)</sup>	Yes
	D041C	4	AREA A position 1 [ $\mu\text{m}$ ] <sup>Note1)</sup>	1	More than 0 <sup>Note 1)</sup>	Yes
	D041E	4	AREA A position 2 [ $\mu\text{m}$ ] <sup>Note1)</sup>	1	More than 0 <sup>Note 1)</sup>	Yes
	D0420	4	AREA B position 1 [ $\mu\text{m}$ ] <sup>Note1)</sup>	1	More than 0 <sup>Note 1)</sup>	Yes
	D0422	4	AREA B position 2 [ $\mu\text{m}$ ] <sup>Note1)</sup>	1	More than 0 <sup>Note 1)</sup>	Yes
	D0424 to D042F	-	-	Writing is not available.	-	-
No. 2 (No. 1) Note 6)	D0430 to D044F	Same as above	Same as above			
...	...					
No. 15	D05D0 to D05EF	Same as above	Same as above			No
No.16 to No.19	Writing is not available.					

## Direct operation parameters

Step Data No.	Address	Number of registers (Bytes)	Description	Configuration unit	Value range	For pulse input mode
No. 20	D9103	4	Target position [ $\mu\text{m}$ ] <sup>Note 1)</sup>	1	Note 1)	No
	D9105	2	Positioning time [0.01 s]	1	0 to 6000	No
	D9106	2	Speed [mm/s]	1	0 to 400	No
	D9107	2	Acceleration [ $\text{mm/s}^2$ ]	1	0 to 60000	No
	D9108	2	Deceleration [ $\text{mm/s}^2$ ]	1	0 to 60000	No
	D9109	2	Pushing speed [mm/s] <sup>Note2)</sup>	1	0 to 20 32768 to 32788	No
	D910A	2	Thrust setting value <sup>Note 3)</sup> [0.1]	1	More than 10 <sup>Note 3)</sup>	No
	D910B	2	Load mass [10g]	5	0, 5, 10 to 100	No
	D910C	2	Movement mode(0: ABS, 1: REL)	1	0, 1	No
	D910D	2	Threshold force value [0.1]	1	1 to 50	No
	D910E	2	Positioning width [ $\mu\text{m}$ ]	1	More than 0 <sup>Note 1)</sup>	No
	D910F	4	AREA A position 1 [ $\mu\text{m}$ ] <sup>Note1)</sup>	1	More than 0 <sup>Note 1)</sup>	No
	D9111	4	AREA A position 2 [ $\mu\text{m}$ ] <sup>Note1)</sup>	1	More than 0 <sup>Note 1)</sup>	No
	D9113	4	AREA B position 1 [ $\mu\text{m}$ ] <sup>Note1)</sup>	1	More than 0 <sup>Note 1)</sup>	No
	D9115	4	AREA B position 2 [ $\mu\text{m}$ ] <sup>Note1)</sup>	1	More than 0 <sup>Note 1)</sup>	No
	D9117 to D911F	-	-	Writing is not available.	-	-

Note 1: The maximum value is the same as for the Card Motor maximum stroke [ $\mu\text{m}$ ].

(e.g. Maximum value for the LAT3-10: 10000 [ $\mu\text{m}$ ])

The minimum values vary depending on the movement mode as described below.

Minimum value in REL operation: Card Motor stroke [ $\mu\text{m}$ ] x -1

(e.g. Minimum value for the LAT3-10: -10000  $\mu\text{m}$ )

Minimum value in ABS operation: 0

Note 2: Refer to the following examples for how to set the pushing speed.

For "Pushing Speed" use 32768 as base value, and add speed in [mm/s]. If no value or a value equal to or less than 32768 is input, the pushing operation will not be performed even if a "Thrust Setting Value" has been input.

- Step Data No. 1 (in Pulse Input Mode, corresponding Step Data is No. 0):

The controller uses the pushing speed value set in Step Data No. 1 (Step Data No. 0) as the speed value for the Return to Origin operation.

(Example) When Step Data No. 1 is used for a positioning operation:

Recommended set value


Set value for pushing speed = **6** (Thus, positioning speed of Return to Origin: 6 mm/s, Pushing speed for Step Data No. 1 positioning operation: 0 mm/s)

(Example) When Step Data No. 1 (Step Data No. 0) is used for a pushing operation and the speed

is set to 6 mm/s:

Recommended set value

Set value for pushing speed = **32774** (Thus, Return to Origin speed and Pushing speed for Step Data No.1 (Step Data No. 0) pushing operation: 6 mm/s)

 **Caution**

**Do not set pushing speed value for Step Data No. 1 (Step Data No. 0 in Pulse Input Mode) to "0" or "32768".**

If those values are input, the Return to Origin operation cannot be performed as the speed becomes 0 mm/s.

- Step Data No. 2 and higher (in Pulse Input Mode, corresponding Step Data is No. 1 and higher):

(Example) For positioning operation:

Recommended set value

Set value for pushing speed = **32768** (Thus, pushing speed: 0 mm/s)

(Example) Pushing operation with set speed of 6 mm/s:

Recommended set value

Set value for pushing speed = **32774** (Pushing speed: 6 mm/s)

Note 3: Maximum values differ depending on the Card Motor model.

(LAT3\*-10: 5.0, LAT3\*-20: 4.8, LAT3\*-30: 3.9, LAT3\*-50: 2.0)

Note 4: Step Data No. 20 is used for Direct Operation.

Note 5: If the target position exceeds the stroke range, the Card Motor operation speed is set to above 400 mm/s or any other impossible value is set, "Step Data error" will be generated from the Card Motor controller during the Step Data operation. Please set values as appropriate.




Note 6: Step Data numbers in brackets are valid in Pulse Input Mode.

Step Data No. 3 is the highest valid Step Data No. in Pulse Input Mode.

(iv) Parameter description

Parameter name	Description
Step Data	Specifies the Step Data number to be read or set (written).
Target position	Specifies the Card Motor target position for a Positioning operation or the position to start pushing for a Pushing operation.
Positioning time	<p>Specifies the movement time from the start of a movement until the Card Motor has to reach the target position.</p> <p>Positioning time is a configurable parameter only when the LATCA controller is in "Step Data Input Type" control signal mode and "Cycle Time Entry Method" has been selected using the "LATC Configurator" software.</p> <div data-bbox="403 557 1418 775" style="border: 1px solid black; padding: 5px;"> <p style="text-align: center;"><b>⚠ Caution</b></p> <p><b>The condition may not execute the operation depending on the order of data calling.</b></p> <p>In this case an alarm will be generated, and the operation will not be completed. Please set according to the customer's operating condition.</p> </div>
Positioning speed	<p>Specifies the Card Motor movement speed to the target position or to the position where to start pushing.</p> <p>Positioning speed is a configurable parameter only when the LATCA controller is in "Step Data Input Type" control signal mode and "Speed Entry Method" has been selected using the "LATC Configurator" software.</p> <div data-bbox="403 1021 1418 1238" style="border: 1px solid black; padding: 5px;"> <p style="text-align: center;"><b>⚠ Caution</b></p> <p><b>If moving distance, speed, acceleration and deceleration parameters which cannot be executed are set, operation may not be able to be completed according to the settings. Acceleration and deceleration settings are detailed below.</b></p> </div>
Acceleration	<p>Specifies the Card Motor acceleration to reach the set positioning speed.</p> <p>Acceleration is a configurable parameter only when the LATCA controller is in "Step Data Input Type" control signal mode and "Speed Entry Method" has been selected using the "LATC Configurator" software.</p>
Deceleration	<p>Specifies the Card Motor deceleration from the set positioning speed to stop.</p> <p>Positioning speed is a configurable parameter only when the LATCA controller is in "Step Data Input Type" control signal mode and "Speed Entry Method" has been selected using the "LATC Configurator" software.</p>
Pushing speed	<p>Specifies the pushing speed for pushing operations.</p> <p>Return to Origin is performed according to the set pushing speed of Step Data No. 1 (Step Data No. 0 in Pulse Input Mode).</p>
Thrust setting value	<p>Specifies the maximum force for pushing operations.</p> <div data-bbox="403 1839 1418 2016" style="border: 1px solid black; padding: 5px;"> <p style="text-align: center;"><b>⚠ Caution</b></p> <p><b>The thrust setting value is for reference only.</b></p> <p>Be sure to check the operation sufficiently before setting and using the card motor.</p> </div>



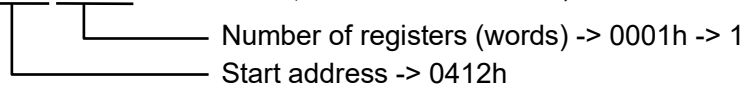
Parameter name	Description						
Load mass	<p>Specifies the weight of the work-piece to be applied to the Card Motor. Load mass configuration in Step Data No. 1 will be used for the Return to Origin operation.</p> <table border="1" data-bbox="405 273 1423 452"> <thead> <tr> <th colspan="2" data-bbox="405 273 1423 331" style="text-align: center;"> <b>Caution</b></th> </tr> </thead> <tbody> <tr> <td colspan="2" data-bbox="405 331 1423 376"><b>The selected load mass is for reference only.</b></td> </tr> <tr> <td colspan="2" data-bbox="405 376 1423 452">Be sure to check the operation sufficiently before setting and using the card motor.</td> </tr> </tbody> </table>	 <b>Caution</b>		<b>The selected load mass is for reference only.</b>		Be sure to check the operation sufficiently before setting and using the card motor.	
 <b>Caution</b>							
<b>The selected load mass is for reference only.</b>							
Be sure to check the operation sufficiently before setting and using the card motor.							
Movement mode	<p>Specifies the co-ordinate system for the target position.</p> <table border="1" data-bbox="414 515 1417 766"> <thead> <tr> <th data-bbox="414 515 660 564">Movement mode</th> <th data-bbox="660 515 1417 564">Details</th> </tr> </thead> <tbody> <tr> <td data-bbox="414 564 660 663">ABS</td> <td data-bbox="660 564 1417 663">The target position will be set as an absolute coordinate with the Card Motor origin point used as a reference point.</td> </tr> <tr> <td data-bbox="414 663 660 766">REL</td> <td data-bbox="660 663 1417 766">The target position will be set as a relative coordinate with the current position used as a reference point.</td> </tr> </tbody> </table>	Movement mode	Details	ABS	The target position will be set as an absolute coordinate with the Card Motor origin point used as a reference point.	REL	The target position will be set as a relative coordinate with the current position used as a reference point.
Movement mode	Details						
ABS	The target position will be set as an absolute coordinate with the Card Motor origin point used as a reference point.						
REL	The target position will be set as a relative coordinate with the current position used as a reference point.						
Threshold force value	<p>Specifies the condition that outputs the INF signal. The INF signal turns ON when the pushing force generated by the Card Motor exceeds the set value. The threshold force value can be configured independent of the thrust setting value, meaning that if a threshold force value higher than the thrust setting value is entered, the INF signal will not turn ON.</p>						
Positioning width	Configures the distance from the target position which outputs the INP signal.						
AREA A position 1	<p>Configures the table position range within which the AREA signal will be output. The AREA signal is output commonly for the ranges set for both the AREA A and AREA B. Set as <math>AREA\ 1 \leq AREA\ 2</math>.</p>						
AREA A position 2							
AREA B position 1							
AREA B position 2							

- Data (Saved data)

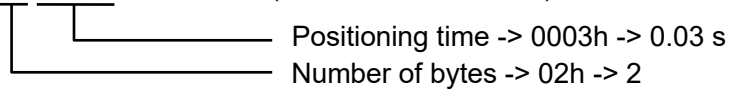
Data is sent as hexadecimal numbers.

Example 1) When reading the positioning time (setting: 0.03 s) for Step Data No. 1 of the controller with controller ID: 1

Request to be sent `":010304120001**CRLF"` (\*\* are the checksum.)

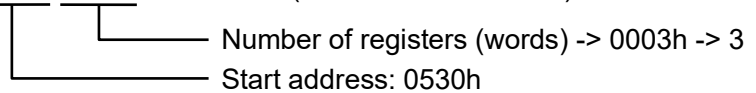


Response data `":0103020003**CRLF"` (\*\* are the checksum.)

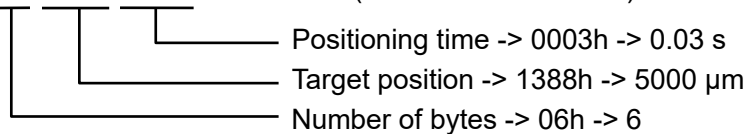


Example 2) When reading the target position (setting: 5000  $\mu\text{m}$ ) and the positioning time (setting 0.03 s) for Step Data No. 10 of the controller with controller ID: 1

Request to be sent `":010305300003**CRLF"` (\*\* are the checksum.)



Response data `":01030613880003**CRLF"` (\*\* are the checksum.)



**! Caution**

**Configure the contents of Step Data No.20 before using Direct Operation.**

The content is reset when it is turned off.

## (2) Read Input Signal Status

### (i) Function

The status of IN0 to IN3, SVON, DRIVE, SETUP, CLR, TL input signals and operation mode flags can be read. Refer to section "5.7(5) Reading and Selection of Operation Mode" for details of the operation modes.

### (ii) Function codes to be used

01h

### (iii) Memory map

In serial operation mode, these flags can be set and read.

In parallel I/O operation mode, parallel I/O signal input status can only be read.

"1" indicates signal status ON, and "0" indicates signal status OFF.

Flag name		Description
Y10	IN 0	In serial operation mode - Command status is read.
Y11	IN 1	
Y12	IN 2 <sup>Note 2)</sup>	
Y13	IN 3 <sup>Note 2)</sup>	In parallel I/O operation mode - Signal input status is read.
Y14	-	Cannot be used.
Y15	-	
Y16	-	
Y17	-	
Y18	-	
Y19	SVON	In serial operation mode - Command status is read.
Y1A	DRIVE <sup>Note 2)</sup>	In parallel I/O operation mode - Signal input status is read.
Y1B	-	Cannot be used.
Y1C	SETUP <sup>Note 1)</sup>	In serial operation mode - Invalid In parallel I/O operation mode - Signal input status is read.
Y1D	CLR <sup>Note 1)</sup>	
Y1E	TL <sup>Note 1)</sup>	
Y1F	-	Cannot be used.
Y30	Operation mode flag	0: Parallel I/O operation mode 1: Serial operation mode

Note 1: The value is always "0" when the LATCA controller is in Step Data Input Type control signal mode.

Note 2: The value is always "0" when the LATCA controller is in Pulse Input Type control signal mode.

### (3) Read Output Signal Status and Internal Flags

(i) Function

Step Data number currently being executed, the status of INF, INFP, BUSY, SVRE, SETON, INP, AREA\_A, AREA\_B, ALARM, OVC and OVT signal output flags can be read.

(ii) Function codes to be used

02h

(iii) Memory map

The status of the internal process of the controller is indicated as follows: When a flag is ON, "1" will be read, and when a flag is OFF, "0" will be read.

Flag name		Description
X40	OUT0	Step Data number currently being executed
X41	OUT1	
X42	OUT2	
X43	OUT3	
X44	-	Cannot be used.
X45	-	Cannot be used.
X46	INF	"INF" signal output flag
X47	INFP	"INFP" signal output flag
X48	BUSY	"BUSY" signal output flag
X49	SVON	Servo ON status flag
X4A	SETON	Return to Origin completion flag
X4B	INP	"INP" signal output flag
X4C	AREA_A	"AREA_A" signal output flag
X4D	AREA_B	"AREA_B" signal output flag
X4E	-	Cannot be used.
X4F	ALARM	"ALARM" signal output flag
X50	OVC	"OVC" signal output flag
X51	OVT	"OVT" signal output flag

#### (4) Read Internal parameters

##### (i) Function

Card Motor current position, speed, value equivalent to thrust, target position and Step Data No. currently being executed can be read.

Execute the Return to Origin operation before reading these parameters. If read before Return to Origin it will not be possible to obtain the correct information.

##### (ii) Function codes to be used

03h

##### (iii) Memory map

Address	Number of registers (Bytes)	Description	Setting unit	Value range
D9000	4	Positioning data	1	900,000 to 1,100,000
D9002	2	Speed data [mm/s]	1	0 to 1000
D9003	2	Value equivalent to thrust	1	0 to 50
D9004	4	Target position <sup>Note 1)</sup>	1	900,000 to 1,100,000 <sup>Note 1)</sup>
D9006	2	Step Data No. currently being executed	1	0 to 99

Note 1: The target position can be read only when the LATCA controller is in Pulse Input Type control signal mode.

##### (iv) Details of contents that can be monitored

###### (a) Positioning data (encoder count value)

The current position data is sent as an encoder count value.

Count value 1,000,000 has been fixed for 0 mm as a reference point.

The count decreases each time the position is moved from 0 mm position toward the opposite side to the connector.

Binary digit (decimal value)	900,000 to 1,100,000
ASCII hex	"000DBBA0" to "0010C8E0"

Formula example: LAT3-10 (encoder resolution: 0.03 mm)

Data response: "000F418C" ⇒ 999,820 count

Card Motor table position: (1,000,000-999,820) x 0.03 mm = 5.4 mm

###### (b) Speed data [mm/s]

Data (absolute value) of the current Card Motor positioning speed is sent. Direction is not taken into consideration.

Binary digit (decimal value)	0 to 1000
ASCII hex	"0000" to "03E8"

(c) Equivalent thrust value data

A value 10x that of the equivalent thrust force value will be sent.

Binary digit (decimal value)	0 to 50
ASCII hex	"00" to "32"

(d) Target position (encoder count value): This item is valid in Pulse Input Type control signal mode only.

The target position data is sent as an encoder count value.

Count value 1,000,000 has been fixed for 0 mm as a reference point. The count decreases each time the position is moved from 0 mm position toward the opposite side to the connector.

The encoder count value is converted into units [mm] by multiplying the count value by the encoder resolution of the used Card Motor model

Binary digit (decimal value)	900,000 to 1,100,000
ASCII hex	"000DBBA0" to "0010C8E0"

Formula example) LAT3-10 (encoder resolution: 0.03 mm)

Data response: "000F418C" ⇒ 999,820 count

Card Motor table position:  $(1,000,000 - 999,820) \times 0.03 \text{ mm} = 5.4 \text{ mm}$

(e) Step Data number executed

Step Data number being executed is sent.

Applicable numbers are as follows.

- For Step Data Input Type control signal mode

0: No Step Data executed

1-15: Step Data number currently executed

20: Direct operation

99: Return to Origin position

- For Pulse Input Type control signal mode

0-3: Step Data number currently executed

99: Return to Origin position

Binary digit (decimal value)	0 to 99
ASCII hex	"00" to "63"

## (5) Reading and Selection of Operation Mode

### (i) Function

The operation mode can be selected from either Parallel or Serial I/O (Parallel I/O operation mode is activated by default when the power supply is turned on).

- Parallel I/O operation mode: Operation of the Card Motor is controlled using parallel I/O signals (when power is supplied) via the LATCA parallel I/O connector (CN5).
- Serial operation mode: Parallel I/O input signals via the LATCA parallel I/O connector (CN5) will not be accepted by the LATCA controller in serial operation mode.

After changing the operation mode to serial operation using this function, the Card Motor is operated with virtual I/O signals transmitted via the LATCA serial connector (CN3) using serial communication. The Parallel I/O functions are similar to those in Parallel I/O operation mode.

### (ii) Function codes to be used

05h / 0Fh

### (iii) Memory map

Flag name		Read	Write	Details
Y30	Operation mode flag	Available	Available	1 <sup>Note 1)</sup> : The operation mode is changed to serial operation mode. 0: The operation mode is changed to parallel I/O operation mode.

Note 1: Use function code 05h, and set the data to FF00h.

### (iv) Cautions for setting

- When power is supplied to the controller, it is by default set to parallel I/O operation mode. Every time the power to the controller is turned off and on again, change the mode to the serial operation mode.
- After changing the operation mode, always perform the Return to Origin operation before executing any other Card Motor operations.
- When switching from serial operation mode to parallel I/O operation mode, the currently active parallel I/O signals will be disabled. When switching to operation by parallel I/O signals, turn off all the parallel I/O signals and input them again after changing to the parallel I/O operation mode.
- If the controller is configured to Pulse Input mode, serial operation cannot be performed.  
Use the "LATC Configurator" configuration software to switch between Step Data Input Mode and Pulse Input Mode.
- Make sure that the Card Motor power is turned OFF (SVON OFF) before using this function.

## (6) Commands for Virtual I/O Operation by Serial Communication

### (i) Function

By writing to the Y contacts, the Step Data number, Card Motor energization ON/OFF and holding of current position / operation start can be performed.

### (ii) Function codes to be used

05h / 0Fh

### (iii) Memory map

Address	Name	Details																																																																																					
Y10	IN 0	<p>Step Data instruction bit No.</p> <p>Specifies the Step Data number as a binary combination of IN0 to IN3.</p> <p>When all of IN0 to IN3 are turned OFF (Step Data No.0), Return to Origin position is performed.</p>																																																																																					
Y11	IN 1	<table border="1"> <thead> <tr> <th>Setting</th> <th>Y13 (IN3)</th> <th>Y12 (IN2)</th> <th>Y11 (IN1)</th> <th>Y10 (IN 0)</th> </tr> </thead> <tbody> <tr> <td>Return to Origin</td> <td>OFF</td> <td>OFF</td> <td>OFF</td> <td>OFF</td> </tr> <tr> <td>Step data 1</td> <td>OFF</td> <td>OFF</td> <td>OFF</td> <td>ON</td> </tr> <tr> <td>Step data 2</td> <td>OFF</td> <td>OFF</td> <td>ON</td> <td>OFF</td> </tr> <tr> <td>Step data 3</td> <td>OFF</td> <td>OFF</td> <td>ON</td> <td>ON</td> </tr> <tr> <td>Step data 4</td> <td>OFF</td> <td>ON</td> <td>OFF</td> <td>OFF</td> </tr> <tr> <td>Step data 5</td> <td>OFF</td> <td>ON</td> <td>OFF</td> <td>ON</td> </tr> <tr> <td>Step data 6</td> <td>OFF</td> <td>ON</td> <td>ON</td> <td>OFF</td> </tr> <tr> <td>Step data 7</td> <td>ON</td> <td>ON</td> <td>ON</td> <td>ON</td> </tr> <tr> <td>Step data 8</td> <td>ON</td> <td>OFF</td> <td>OFF</td> <td>OFF</td> </tr> <tr> <td>Step data 9</td> <td>ON</td> <td>OFF</td> <td>OFF</td> <td>ON</td> </tr> <tr> <td>Step data 10</td> <td>ON</td> <td>OFF</td> <td>ON</td> <td>OFF</td> </tr> <tr> <td>Step data 11</td> <td>ON</td> <td>OFF</td> <td>ON</td> <td>ON</td> </tr> <tr> <td>Step data 12</td> <td>ON</td> <td>ON</td> <td>OFF</td> <td>OFF</td> </tr> <tr> <td>Step data 13</td> <td>ON</td> <td>ON</td> <td>OFF</td> <td>ON</td> </tr> <tr> <td>Step data 14</td> <td>ON</td> <td>ON</td> <td>ON</td> <td>OFF</td> </tr> <tr> <td>Step data 15</td> <td>ON</td> <td>ON</td> <td>ON</td> <td>ON</td> </tr> </tbody> </table>	Setting	Y13 (IN3)	Y12 (IN2)	Y11 (IN1)	Y10 (IN 0)	Return to Origin	OFF	OFF	OFF	OFF	Step data 1	OFF	OFF	OFF	ON	Step data 2	OFF	OFF	ON	OFF	Step data 3	OFF	OFF	ON	ON	Step data 4	OFF	ON	OFF	OFF	Step data 5	OFF	ON	OFF	ON	Step data 6	OFF	ON	ON	OFF	Step data 7	ON	ON	ON	ON	Step data 8	ON	OFF	OFF	OFF	Step data 9	ON	OFF	OFF	ON	Step data 10	ON	OFF	ON	OFF	Step data 11	ON	OFF	ON	ON	Step data 12	ON	ON	OFF	OFF	Step data 13	ON	ON	OFF	ON	Step data 14	ON	ON	ON	OFF	Step data 15	ON	ON	ON	ON
Setting	Y13 (IN3)	Y12 (IN2)	Y11 (IN1)	Y10 (IN 0)																																																																																			
Return to Origin	OFF	OFF	OFF	OFF																																																																																			
Step data 1	OFF	OFF	OFF	ON																																																																																			
Step data 2	OFF	OFF	ON	OFF																																																																																			
Step data 3	OFF	OFF	ON	ON																																																																																			
Step data 4	OFF	ON	OFF	OFF																																																																																			
Step data 5	OFF	ON	OFF	ON																																																																																			
Step data 6	OFF	ON	ON	OFF																																																																																			
Step data 7	ON	ON	ON	ON																																																																																			
Step data 8	ON	OFF	OFF	OFF																																																																																			
Step data 9	ON	OFF	OFF	ON																																																																																			
Step data 10	ON	OFF	ON	OFF																																																																																			
Step data 11	ON	OFF	ON	ON																																																																																			
Step data 12	ON	ON	OFF	OFF																																																																																			
Step data 13	ON	ON	OFF	ON																																																																																			
Step data 14	ON	ON	ON	OFF																																																																																			
Step data 15	ON	ON	ON	ON																																																																																			
Y12	IN 2																																																																																						
Y13	IN 3																																																																																						
Y19	SVON	<p>The SVON signal turns the servo motor ON. "Servo ON" is a command to energize the motor.</p> <p>If the servo ON signal is turned OFF, the motor can be moved freely.</p> <p>It is possible to clear alarms by resetting the SVON flag Y19 (ON -&gt; OFF -&gt; ON).</p>																																																																																					
Y1A	DRIVE	<p>When the DRIVE flag Y1A is changed from OFF to ON, the system scans Y10 to Y13 (IN0 to IN3) and starts the operation according to corresponding Step Data parameters.</p> <p>The operation is stopped when the Y1A is changed from ON to OFF, and the position where this register value is changed will be held.</p>																																																																																					

Write "1" to switch the signal status to ON or "0" to switch it OFF.

When using function code 05h, write "FF00h" to switch the signal status to ON.




(iv) Cautions for setting

- (a) Before using this function, set the controller to serial operation mode using the operation mode selection function. This function can be used for both Step Data Operation or Direct Operation.
- (b) Mode switching process
  - Parallel I/O operation -> Serial operation: All Y10 to Y1F will be reset to "0".
  - Serial operation -> Parallel I/O operation: Input signal status will be reflected immediately.
- (c) When all Y contacts are written simultaneously using function code 0Fh, write "0" to the addresses that cannot be used.

(v) Examples of operation and parameters

Operation	Resister status		
	IN3 to IN0	SVON	DRIVE
Power supply to Card Motor is turned ON (servo ON)	0000	1	0
Execution of Return to Origin	0000	1	1
Position maintained after completion of the Return to Origin operation	0000	1	0
Execution of Step Data	#	1	1
Maintain current position after Step Data execution	#	1	0
Card Motor power OFF	0000	0	0

#: Step Data No. to be executed (1-15, 20)

 **Caution**

**Execute the next consecutive Step Data only after the Card Motor has completed the current operation.**

Otherwise, the Card Motor may execute an unexpected operation.

**If the SVON and DRIVE signals are turned ON when changing from serial operation to parallel I/O operation, the input signals will be invalid. Make sure to turn OFF the input signals when changing the mode.**

## (7) Direct Operation Command

### (i) Function

Command for executing Direct Operation

### (ii) Function codes to be used

10h

### (iii) Memory map

Address	Function	Number of registers (Bytes)	Details
D9100 (D9101)	Direct operation command	2	Starts the operation according to parameters set in "D9103" to "D 9111". (The register value will be forced to "0000h" when the DRIVE flag Y1A status is "1".) "0000h": Position holding status "0100h": Starts Direct Operation
	Undefined	2	

### (iv) Cautions for setting

- (a) Before performing the Direct Operation command, set the Direct Operation Step Data parameters for Step Data No. 20 (D9103 to D9115) using the Direct Operation Step Data setting function (function codes 06h or 10h). The Direct Operation is performed according to the parameters set in Step Data No. 20.
- (b) The Direct Operation starts when the register value of address D9100 is changed from "0000h" to "0100h".
- (c) If a Step Data positioning operation is performed by setting the DRIVE flag Y1A to "1", the Direct Operation command becomes invalid, and the register value of address D9100 will automatically be forced to "0000h".
- (d) If the register value of address D9100 is changed to "0000h" when the Card Motor is moving during a Direct Operation, the condition will be changed over to "Position holding status" and the Card Motor movement will stop before reaching the target position and the position will be held with the motor energized.
- (e) If DRIVE flag Y1A is set to "1" during a Direct Operation, the condition will be changed to "Step Data Operation", and the register value for address D9100 will automatically be rewritten to "0000h".

## (8) Read / Clear Alarm History

### (i) Function

The alarm history stored in the controller can be read. The alarm history can be cleared by writing "0000h" to D0600 to D0613 (All histories are set to Alarm No. 0 "No alarm").

Details of the alarm can be checked using this function when an alarm has been generated.

### (ii) Function codes to be used

03h / 10h

### (iii) Memory map

Address	Number of registers(Bytes)	Details
D0600	2	Alarm number of alarm entry 1
D0601	2	Alarm number of alarm entry 2
D0602	2	Alarm number of alarm entry 3
D0603	2	Alarm number of alarm entry 4
D0604	2	Alarm number of alarm history 5
D0605	2	Alarm number of alarm history 6
D0606	2	Alarm number of alarm history 7
D0607	2	Alarm number of alarm history 8
D0608	2	Alarm number of alarm history 9
D0609	2	Alarm number of alarm history 10
D060A	2	Alarm number of alarm history 11
D060B	2	Alarm number of alarm history 12
D060C	2	Alarm number of alarm history 13
D060D	2	Alarm number of alarm history 14
D060E	2	Alarm number of alarm history 15
D060F	2	Alarm number of alarm history 16
D0610	2	Alarm number of alarm history 17
D0611	2	Alarm number of alarm history 18
D0612	2	Alarm number of alarm history 19
D0613	2	Alarm number of alarm history 20

### (iv) Alarm numbers and names

Alarm No.	Hexadecimal value	Alarm name	Details
0	0000h	No alarm	-
1	0001h	Memory error	-
2	0002h	Actuator cable disconnected	-
3	0003h	Temperature error	-
4	0004h	Over current error	Motor overload error
5	0005h		I/O signal overcurrent error
6	0006h	Parameter error	Origin parameter error
7	0007h		Step Data parameter error
8	0008h		Invalid parameter error
9	0009h	Pulse input error	Pulse speed error
10	000Ah		Pulse overflow error
11	000Bh	Return to Origin position non-execution error	-

- The alarm number that is generated when the alarm occurred is saved in History 1. The previous history moves from 1 to 2, 2 to 3 and so on.

For details and solutions to alarms, see section "16. Alarm Detection" (p.81) in operation manual "Card Motor Controller (Step Data Input edition)".

### **Caution**

**Ensure the alarm history is cleared only after taking corrective action in response to the alarms.**

When the alarm history is cleared, any currently occurring alarms will also be cleared.

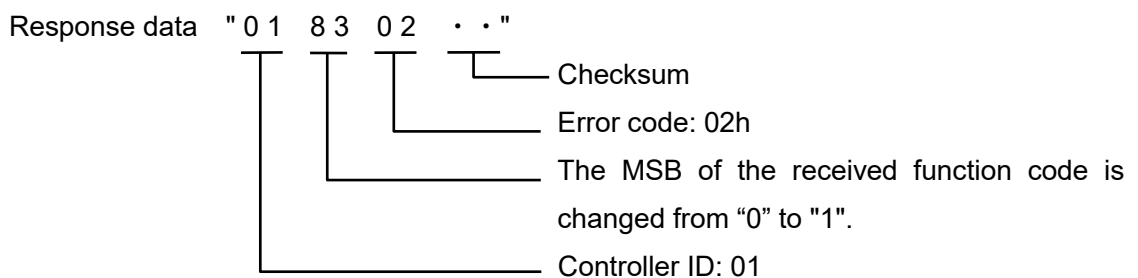
## 5.8 Error Codes

If an error is found in the received data, the following error codes are sent back.  
The MSB of the received function code in the request data is set to "1" in the response.

Error code	Name	Description · Countermeasures
01h	ILLEGAL FUNCTION	<p>&lt; Description &gt; An undefined function has been specified.</p> <p>&lt; Countermeasures &gt; Send the function code in <b><u>5.3 Message Frame Formats (4) Function codes.</u></b></p>
02h	ILLEGAL DATA ADDRESS	<p>&lt; Description &gt; Specified register address is incorrect.</p> <p>&lt; Countermeasures &gt; Specify the correct address referring to <b><u>5.5 Memory Map.</u></b></p>
03h	ILLEGAL DATA VALUE	<p>&lt; Description &gt; Content of the sent data is incorrect.</p> <p>(1) Frame format is incorrect or the data exceeds the setting range.</p> <p>(2) During the alarm, an instruction to turn on the SVON signal was sent before turning off the SVON signal.</p> <p>&lt; Countermeasures &gt; (1) referring to <b><u>5.4 Details of Function Codes</u></b> supported by LATCA.</p> <p>(2) If an alarm occurs, turn off the SVON signal to release the alarm and then turn on (SVON reset).</p>
06h	SLAVE DEVICE BUSY	<p>&lt; Description &gt; The request cannot be processed due to a controller busy status.</p> <p>(1) The following command was sent during a reply after receiving the communication.</p> <p>(2) The next write instruction was sent during the data write process.</p> <p>&lt; Countermeasures &gt; (1) After the controller has successfully replied, send the following command.</p> <p>(2) Perform a transmission retry on the upper level device.</p>

Note 1: If a communication error (framing error, over-run error, parity error, check sum error) occurs, the received data will be discarded without generating the error codes shown above as no response can be made.

Example) When an error with code 02h is found in the received data with function code 03h.



## 6. Card Motor Controller Operation Examples

### 6.1 Basic Settings, I/O Configuration

Set the following parameters in the controller using the “LATC Configurator” controller configuration software.

(See operation manual “Card Motor Controller (Step Data Input edition)” for details.)

- Control signal input type (select Step Data Input Type as control mode)
- Card Motor mounting orientation
- Return to Origin method (Retracted / Extended / Sensor)
- Step Data Input Method (Cycle Time Entry / Speed Entry)
- Output signal functions
- Area ranges
- Threshold force value

### 6.2 Step Data configuration

Summary of the Step Data parameter configuration procedure.

- (1) When setting Step Data Operation parameters, de-energize the Card Motor using “virtual I/O operation commands by serial communication”. It is not required to de-energize the Card Motor when setting Direct Operation Step Data No. 20 parameters.
- (2) Set the Step Data parameters using the “Step Data Operation” / “Direct Operation” Step Data setting functions.

#### \* Cautions for Step Data configuration

- (1) If a communication error (framing error, overrun error, parity error, checksum error) occurs, a response cannot be made, resulting in received data being discarded and no irregular response with error code sent. Therefore prepare retry/time-out processing in the master transmission device.
- (2) Use the “LATC Configurator” controller configuration software to pre-set the basic controller settings.
- (3) The Step Data overwrite target frequency for Step Data numbers 1-15 is 4 million. Please avoid excessive overwriting.
- (4) Do not cut the power while data is being saved.  
It is recommended that the controller be initialized using the “LATC Configurator” controller configuration software in the case that the power is cut accidentally.
- (5) Always ensure that Step Data parameter configuration for Step Data Operation is made while the power supply to the Card Motor is off. If Step Data configuration is done while the Card Motor is energized, malfunction may occur.
- (6) For Step Data Operation ensure that the Return to Origin operation is completed before operating the Card Motor after any Step Data parameter changes have been made.

## 6.3 Operation Data Acquisition

Summary of the Card Motor operation data acquisition procedure.

- (1) Acquire the operating information using the "Internal parameter reading" function.

## 6.4 Operation Instruction Method

Summary of the procedure for operation instructions to the Card Motor using serial communication.

- (1) Ensure the SVON signal input to the controller is turned OFF to disconnect the power from the Card Motor.
- (2) Change the operation mode of the controller to serial operation mode using the "Operation mode selection" function.
- (3) Set and operate the Step Data according to the operation mode as shown below, and command operation using the "virtual I/O operation commands by serial communication".

In Step Data Operation: Step Data 1-15 will be defined.

In Direct Operation: Step Data 20 will be defined.

### **Caution**

**When operating the controller using the "virtual I/O operation commands by serial communication", write "0" (OFF) to the DRIVE flag (Y1A) before driving, and then, write DRIVE "1" (ON) to start the movement.**

When the DRIVE is changed from "0" to "1" (from OFF to ON), the operation of the Step Data selected in IN0 to IN3 (Y10 to Y13) starts.

## 6.5 Program Examples

Program examples for how to perform Step Data setting, Return to Origin, positioning and acquisition of operating information by serial communication are listed below.

The request and response data is indicated in hexadecimal numbers; the monitored contents of "Reading of internal parameters" and "Reading of internal signals" functions are indicated as "\*\*"; and the start code, check sum and end code are omitted. The example has spaces, but the actual data does not contain spaces. Do not enter spaces when sending data.

### (1) Basic settings

Use the dedicated "LATC Configurator" controller configuration software. The controller configuration software can be downloaded from SMC website.

[https://www.smcworld.com/products/pickup/en-jp/electric\\_actuator/download.html](https://www.smcworld.com/products/pickup/en-jp/electric_actuator/download.html)

### (2) Communication check

Send an echo back (08h) to check if a successful response is returned.

#### i. Modbus ASCII protocol

(ASCII notation)

Request data : 01 08 0000 1234 B1 [CR,LF]

Response data (normal response) : 01 08 0000 1234 B1 [CR,LF]

(ASCII hexadecimal notation)

Request data 3Ah 30h 31h 30h 38h 30h 30h 30h 30h 31h 32h 33h 34h 42h  
31h 0Dh 0Ah

Response data (normal response) 3Ah 30h 31h 30h 38h 30h 30h 30h 30h 31h 32h 33h 34h 42h  
31h 0Dh 0Ah

#### ii. Modbus RTU protocol

(Hexadecimal notation)

Request data 01 08 0000 1234 ED7C

Response data (normal response) 01 08 0000 1234 ED7C

(Binary notation)

Response data 0001 1000 0000 0000 0001 0010 0011 0100 1110 1101 0111  
1100

Response data (normal response) 0001 1000 0000 0000 0001 0010 0011 0100 1110 1101 0111  
1100

For correspondence between ASCII notation and ASCII hexadecimal notation, refer to the [8.4 ASCII Code List](#).

[CR, LF] in ASCII notation indicates the exit code.

In Modbus RTU, provide a silent interval of at least 3.5 characters at the beginning and the end.



### (3) Step Data configuration examples

Instructions issued to set Step Data No. 1 Positioning time to 0.1 sec. and Target position to 10 mm in a Card Motor controller with controller ID1.

Step	Request data	Response data	Operation details
1	01 06 0412 000A	01 06 0412 000A	Sets Step Data No. 1 Positioning time to 0.1 sec.
2	01 10 0410 0002 04 0000 2710	01 10 0410 0002	Sets Step Data No. 1 Target position to 10 mm.

### (4) Return to Origin

Change the Card Motor controller (controller ID1) operation mode to serial operation, and send a Return to Origin instruction.

Step	Request data	Response data	Operation details
1	01 0F 0010 000B 02 0000	01 0F 0010 000B	Instruction to turn Card Motor power OFF sent.
2	01 05 0030 FF00	01 05 0030 FF00	Instruction to change the Card Motor controller operation mode to "serial operation" sent.
3	01 05 0019 FF00	01 05 0019 FF00	Instruction to turn Card Motor power ON sent.
4	01 05 001A FF00	01 05 001A FF00	Instruction to perform the Return to Origin operation sent.
5	01 02 004B 0001	01 02 01 **	Acquires the INP signal information.
6	Repeat steps 5-6		Acquires the information repeatedly until the received information shows that the INP is "1". When INP becomes "1", the Return to Origin operation is completed.
7	01 05 001A 0000	01 05 001A 0000	Transit to position retention and ends the return to origin operation.

### (5) Positioning operation (Step Data Operation) example

Change the Card Motor controller (controller ID1) operation mode to serial operation, send an instruction so that the Step Data No. 1 and No. 2 are continuously repeated after Return to Origin.

\* If returning to origin has already been completed, omit No.1 to No.6 and start from No.7.

Step	Request data	Response data	Operation details
1	01 0F 0010 000B 02 0000	01 0F 0010 000B	Instruction to turn Card Motor power OFF sent.
2	01 05 0030 FF00	01 05 0030 FF00	Instruction to change the Card Motor controller operation mode to "serial operation" sent.
3	01 05 0019 FF00	01 05 0019 FF00	Instruction to turn Card Motor power ON sent.
4	01 05 001A FF00	01 05 001A FF00	Instruction to perform the Return to Origin operation sent.
5	01 02 004B 0001	01 02 01 **	Acquires the INP signal information.
6	Repeat steps 5-6		Acquires the information repeatedly until the received information shows that the INP is "1". When INP becomes "1", the Return to Origin operation is completed.
7	01 0F 0010 000B 02 0102	01 0F 0010 000B	Instruction to hold current position sent. (Preparation for start of Step Data No. 1 operation)
8	01 0F 0010 000B 02 0106	01 0F 0010 000B	Instruction to execute the operation of Step Data No.1 sent.
9	01 02 004B 0001	01 02 01 **	Acquires the INP signal information.
10	Repeat steps 9-10		Acquires the information repeatedly until the received information shows that the INP is "1". INP = 1 indicates that the target location has been reached.
11	—		After settling time, the operation of step data No.1 is completed.
12	01 0F 0010 000B 02 0202	01 0F 0010 000B	Instruction to hold current position sent. (Preparation for start of Step Data No. 2 operation)
13	01 0F 0010 000B 02 0206	01 0F 0010 000B	Instruction to execute the operation of Step Data No. 2 sent.
14	01 02 004B 0001	01 02 01 **	Acquires the INP signal information.
15	Repeat steps 14-15		Acquires the information repeatedly until the received information shows that the INP is "1". INP = 1 indicates that the target location has been reached.
16	—		After settling time, the operation of step data No.2 is completed.
17	Repeat steps 7-16		Step Data No. 1 and No .2 will be repeated.

## (6) Positioning operation (Direct Operation) example

Change the pre-configured Card Motor controller (controller ID: 1, Cycle Time Entry Method) operation mode to serial operation, and the Card Motor is moved from 5 mm to 10 mm position by Direct Operation after Return to Origin.

\*If returning to origin has already been completed, omit No.1 to No.6 and start from No7.

Step	Request data	Response data	Operation details
1	01 0F 0010 000B 02 0000	01 0F 0010 000B	Instruction to turn Card Motor power OFF sent.
2	01 05 0030 FF00	01 05 0030 FF00	Instruction to change the Card Motor controller operation mode to "Serial operation" sent.
3	01 05 0019 FF00	01 05 0019 FF00	Instruction to turn Card Motor power ON sent.
4	01 05 001A FF00	01 05 001A FF00	Instruction to perform the Return to Origin operation sent.
5	01 02 004B 0001	01 02 01 **	Acquires the INP signal information.
6	Repeat steps 5-6		Acquires the information repeatedly until the received information shows that the INP is "1".
7	01 10 9103 0002 04 0000 1388	01 10 9103 0002	Direct Operation Step Data No. 20 parameters are configured for: Target position 5 mm, Positioning time 0.1 s.
8	01 06 9105 000A	01 06 9105 000A	
9	01 10 9100 0001 02 0000	01 10 9100 0000	Instruction to hold current position sent. (Preparation for start of Step Data No. 20 operation)
10	01 10 9100 0001 02 0100	01 10 9100 0001	Instruction to execute the operation of Step Data No. 20 sent.
11	01 02 004B 0001	01 02 01 **	Acquires the INP signal information.
12	Repeat steps 9-10		Acquires the information repeatedly until the received information shows that the INP is "1". INP = 1 indicates that the target location has been reached.
13	—		After settling time, the operation of step data No.20 is completed.
14	01 10 9103 0002 04 0000 2710	01 10 9103 0002	Step Data No. 20 target position changed to 10 mm.
15	01 10 9100 0001 02 0000	01 10 9100 0000	Instruction to hold current position sent. (Preparation for start of Step Data No. 20 operation)
16	01 10 9100 0001 02 0100	01 10 9100 0001	Instruction to execute the operation of Step Data No. 20 sent.
17	01 02 004B 0001	01 02 01 **	Acquires the INP signal information.
18	Repeat steps 17-18		Acquires the information repeatedly until the received information shows that the INP is "1". INP = 1 indicates that the target location has been reached.
19	-		After settling time, the operation of step data No.20 is completed.

## ⚠ Caution

**If the direct operation parameters such as target position (Step Data No. 20) for D9103 to D9115 are changed using the "Direct Operation Step Data setting" function during direct operation, the changed parameters will not be reflected to the operation being executed.**

The operation starts at the parameters set to the D9103 to D9115 only after the D9100 data is changed from "0000h" to "0100h" using the "virtual I/O operation commands by serial communication" .

**Configure the contents of Step Data No.20 before using Direct Operation.**

The content is reset when it is turned off.

### (7) Operation data acquisition example

E.g.) Card Motor and controller operation data is acquired from the pre-configured Card Motor controller (controller ID1, Cycle Time Entry Method).

Step	Request data	Response data	Operation details
1	01 03 9000 0007	01 03 0E **	Acquires the Card Motor operation data.
2	Operation Data Judgment		The Card Motor operation state can be determined based on the acquired operation data.

### (8) Alarm history acquisition example

E.g) Four (4) alarm histories stored in the Card Motor controller (Controller ID1) are acquired.

Step	Request data	Response data	Operation details
1	01 03 0600 0004	01 03 08 **	Acquires the alarm history.
2	Alarm history confirmation		Confirms the acquired alarm history.

### (9) Alarm history clearance example

E.g) The Alarm History saved in the Card Motor controller (controller ID1) is cleared.

Step	Request data	Response data	Operation details
1	01 10 0600 000A 14 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000	01 10 0600 000A	Alarm history command to clear alarm history sent. (0 to 10 entries)
2	01 10 060A 000A 14 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000	01 10 060A 000A	Alarm history command to clear alarm history sent. (11 to 20 entries)

 **Caution**

**Ensure the alarm history is cleared only after taking corrective action in response to the alarms.**

When the alarm history is cleared, any currently occurring alarms will also be cleared.

**Alarm Reset**

When an alarm occurs, most of the alarms may be reset using the SVON reset function (turn the SVON signal OFF and back ON again) after taking corrective measures.

## 7. Option (Separately Sold Products)

### 7.1 Communication cable

The cable used to connect one controller to an upper level device (PLC).

#### (1) Communication cable

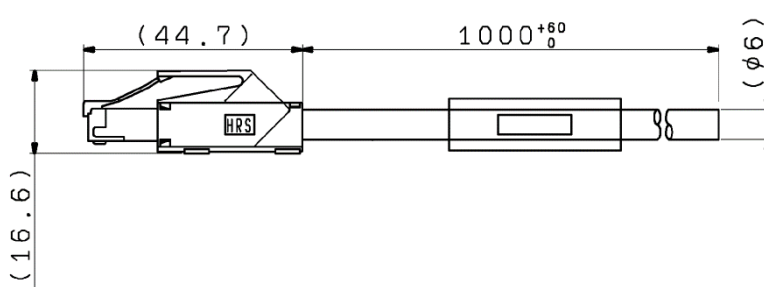
##### i. How to Order

**LATH 6 - 1**

● Cable length

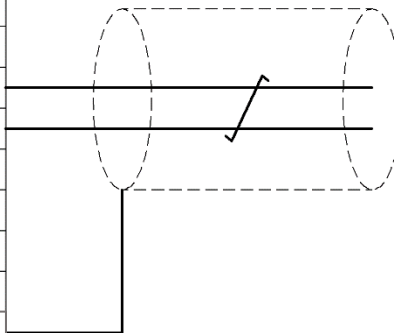
1	1m
---	----

##### ii. Dimensions



##### iii. Wiring diagram

Terminal No.	Insulation color	Function
1	—	NC
2	—	NC
3	White	SD+
4	Black	SD-
5	—	NC
6	—	NC
7	—	NC
8	—	NC
Connector case	Shield	FG



## 7.2 Branch communication cable

The cable used to connect multiple controllers to an upper level device (PLC).

(1) Branch communication cable (between an upper level device and a branch connector)

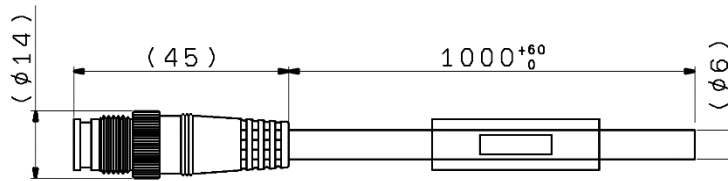
i. How to order

**LATH7-1**

● Cable length

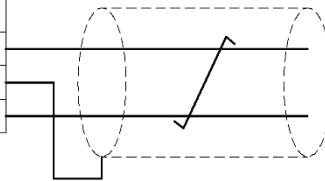
1	1m
---	----

ii. Dimensions



iii. Wiring diagram

Terminal	No.Insulation color	Function
1	—	NC
2	White	SD+
3	Shield	FG
4	Black	SD-



(2) Communication cable (between the branch connector and controller)

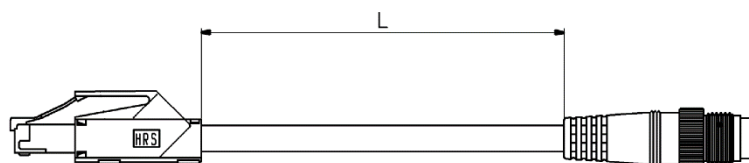
i. How to order

**LEC-CG1-L**

● Cable length(L)

K	0.3m
L	0.5m
1	1m

ii. Dimensions



(3) Cable between branches (between the branch connectors)

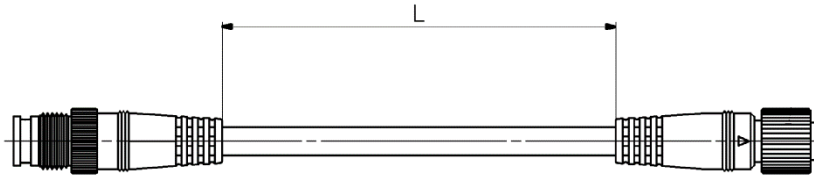
i. How to order

**LEC-CG2-L**

● Cable length(L)

<b>K</b>	0.3m
<b>L</b>	0.5m
<b>1</b>	1m

ii. Dimensions



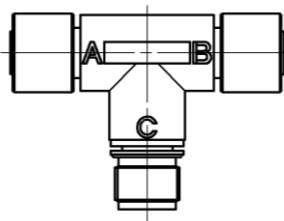
(4) Branch connector

i. How to order

**LEC-CGD**

● Branch connector

ii. Dimensions

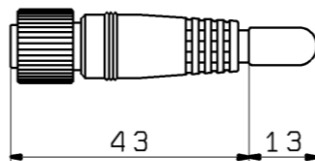


(5) Terminating resistor

i. How to order

**LEC-CGR**

ii. Dimensions





## 8. Reference Information

### 8.1 LRC Checksum Calculation Procedure

#### (1) How to calculate the LRC checksum

The checksum calculation method in Modbus ASCII mode is based on a Longitudinal Redundancy Checking (LRC) method.

- (i) All data in the transmitted message is added together, excluding the Start Code ":" and End Code pair "CR, LF".
- (ii) Subtract the last two bytes of the calculation result in step (1) from FFh.
- (iii) Add 1h to the calculation result in step (2).

#### (2) Calculation example

e.g.) When monitor command "Read alarm history" is sent to the controller with controller ID: 1

Transmission data to be used in the checksum calculation ...

"0", "1", "0", "3", "0", "6", "0", "0", "0", "0", "0", "4"

- (i)  $01h + 03h + 06h + 00h + 00h + 04h = 0Eh$
- (ii)  $FFh - 0Eh = F1h$
- (iii)  $F1h + 01h = F2h$  ... Checksum = "F2" (= 46h, 32h)

### 8.2 CRC Check Sum Calculation Procedure

#### (1) How to calculate the CRC checksum

The check sum calculation method in Modbus RTU mode is based on a Cyclic Redundancy Checking (CRC) method.

All data in the transmitted message is applied in the calculation as follows.

- (i) Preload "FFFFh".
- (ii) Exclusive OR the first byte of the transmitted message with the value in (i).
- (iii) Shift the result of (ii) one bit toward the least significant bit, and fill a zero into the most significant bit position.
- (iv) If the least significant bit was a 1, exclusive OR the result of (iii) with "A001h".  
If the least significant bit was a 0, no exclusive OR takes place.
- (v) Repeat (iii) to (v) until eight shifts have been performed.
- (vi) After the last eight shifts, exclusive OR the result of (vi) with the next 1-byte (2nd byte).
- (vii) The processes (iii) to (vii) are repeated to all the remaining bytes of the message (3rd byte to the last byte).
- (viii) The 2-byte data of the result of (viii) is the CRC value.

#### (2) Calculation example

e.g.) When a communication test of the controller with controller ID 1 using echo back is performed:

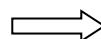
Controller ID: 01

Function code: 08h

Test code: 0000

Test data value: 1234h

Transmitted data (Hexadecimal value): 010800001234



value	Data
1 <sup>st</sup> value	01h
2 <sup>nd</sup> value	08h
3 <sup>rd</sup> value	00h
4 <sup>th</sup> value	00h
5 <sup>th</sup> value	12h
6 <sup>th</sup> value	34h

ExOR flag: The ExOR flag determines whether the result is to be Exclusive ORed with "A001h"

or not after each 1-bit shift.

If the least significant bit was "1", exclusive OR the content with "A001h" after the 1-bit shift.

bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
A001h	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	1

Step	Hex	Arithmetic content	High byte "CRC16(H)"								Low byte "CRC16(L)"								ExOR flag
			15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
1	-	Preload FFFFh	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	-
2	Value of 1st byte: "01h"	Load 1st byte "01h"	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	-
3		ExOR step 1 with step 2	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	-
4		1st shift, Fill in 0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0
5		2nd shift, ExOR A001h	1	0	0	1	1	1	1	1	1	1	1	1	1	1	1	0	1
6		3rd shift	0	1	0	0	1	1	1	1	1	1	1	1	1	1	1	1	0
7		4th shift, ExOR A001h	1	0	0	0	0	1	1	1	1	1	1	1	1	1	1	0	1
8		5th shift	0	1	0	0	0	0	1	1	1	1	1	1	1	1	1	1	0
9		6th shift, ExOR A001h	1	0	0	0	0	0	0	1	1	1	1	1	1	1	1	0	1
10		7th shift	0	1	0	0	0	0	0	0	1	1	1	1	1	1	1	1	0
11		8th shift, ExOR A001h	1	0	0	0	0	0	0	0	0	1	1	1	1	1	1	0	1
12	Value of 2nd byte: "08h"	Load 2nd byte "08h"	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	-
13		ExOR step 11 with step 12	1	0	0	0	0	0	0	0	0	1	1	1	0	1	1	0	-
14		1st shift	0	1	0	0	0	0	0	0	0	0	1	1	1	0	1	1	0
15		2nd shift, ExOR A001h	1	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0	1
16		3rd shift	0	1	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0
17		4th shift	0	0	1	0	0	0	0	0	0	0	0	0	0	1	1	1	0
18		5th shift, ExOR A001h	1	0	1	1	0	0	0	0	0	0	0	0	0	0	1	0	1
19		6th shift	0	1	0	1	1	0	0	0	0	0	0	0	0	0	0	1	0
20		7th shift, ExOR A001h	1	0	0	0	1	1	0	0	0	0	0	0	0	0	0	1	1
21		8th shift, ExOR A001h	1	1	1	0	0	1	1	0	0	0	0	0	0	0	0	1	1
22	Value of 3rd byte: "00h"	Load 3rd byte "00h"	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-
23		ExOR step 21 with step 22	1	1	1	0	0	1	1	0	0	0	0	0	0	0	0	1	-
24		1st shift, ExOR A001h	1	1	0	1	0	0	1	1	0	0	0	0	0	0	0	1	1
25		2nd shift, ExOR A001h	1	1	0	0	1	0	0	1	1	0	0	0	0	0	0	1	1
26		3rd shift, ExOR A001h	1	1	0	0	0	1	0	0	1	1	0	0	0	0	0	1	1
27		4th shift, ExOR A001h	1	1	0	0	0	0	1	0	0	1	1	0	0	0	0	1	1
28		5th shift, ExOR A001h	1	1	0	0	0	0	0	1	0	0	1	1	0	0	0	1	1
29		6th shift, ExOR A001h	1	1	0	0	0	0	0	0	1	0	0	1	1	0	0	1	1
30		7th shift, ExOR A001h	1	1	0	0	0	0	0	0	0	1	0	0	1	1	0	1	1
31		8th shift, ExOR A001h	1	1	0	0	0	0	0	0	0	0	1	0	0	1	1	1	1
32		Load 4th byte "00h"	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-	

Step	Hex	Arithmetic content	High byte "CRC16(H)"								Low byte "CRC16(L)"								ExOR flag
			15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
33	Value of 4th byte: "00h"	ExOR step 31 with step 32	1	1	0	0	0	0	0	0	0	0	1	0	0	1	1	1	-
34		1st shift, ExOR A001h	1	1	0	0	0	0	0	0	0	0	0	1	0	0	1	0	1
35		2nd shift	0	1	1	0	0	0	0	0	0	0	0	0	1	0	0	1	0
36		3rd shift, ExOR A001h	1	0	0	1	0	0	0	0	0	0	0	0	0	1	0	1	1
37		4th shift, ExOR A001h	1	1	1	0	1	0	0	0	0	0	0	0	0	0	1	1	1
38		5th shift, ExOR A001h	1	1	0	1	0	1	0	0	0	0	0	0	0	0	0	0	1
39		6th shift 6	0	1	1	0	1	0	1	0	0	0	0	0	0	0	0	0	0
40		7th shift 7	0	0	1	1	0	1	0	1	0	0	0	0	0	0	0	0	0
41		8th shift 8	0	0	0	1	1	0	1	0	1	0	0	0	0	0	0	0	0
42	Value of 5th byte: "12h"	Load 5th byte "12h"	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0	-
43		ExOR step 41 with step 42	0	0	0	1	1	0	1	0	1	0	0	1	0	0	1	0	-
44		1st shift	0	0	0	0	1	1	0	1	0	1	0	0	1	0	0	1	0
45		2nd shift, ExOR A001h	1	0	1	0	0	1	1	0	1	0	1	0	0	1	0	1	1
46		3rd shift, ExOR A001h	1	1	1	1	0	0	1	1	0	1	0	1	0	0	1	1	1
47		4th shift, ExOR A001h	1	1	0	1	1	0	0	1	1	0	1	0	1	0	0	0	1
48		5th shift	0	1	1	0	1	1	0	0	1	1	0	1	0	1	0	0	0
49		6th shift	0	0	1	1	0	1	1	0	0	1	1	0	1	0	1	0	0
50		7th shift	0	0	0	1	1	0	1	1	0	0	1	1	0	1	0	1	0
51	8th shift, ExOR A001h	1	0	1	0	1	1	0	1	1	0	0	1	1	0	1	1	1	
52	Value of 6th byte: "34h"	Load 6th byte "34h"	0	0	0	0	0	0	0	0	0	0	1	1	0	1	0	0	0
53		ExOR step 51 with step 52	1	0	1	0	1	1	0	1	1	0	1	0	1	1	1	1	0
54		1st shift, ExOR A001h	1	1	1	1	0	1	1	0	1	1	0	1	0	1	1	0	1
55		2nd shift	0	1	1	1	1	0	1	1	0	1	1	0	1	0	1	1	0
56		3rd shift, ExOR A001h	1	0	0	1	1	1	0	1	1	0	1	1	0	1	0	0	1
57		4th shift	0	1	0	0	1	1	1	0	1	1	0	1	1	0	1	0	0
58		5th shift	0	0	1	0	0	1	1	1	0	1	1	0	1	1	0	1	0
59		6th shift, ExOR A001h	1	0	1	1	0	0	1	1	1	0	1	1	0	1	1	1	1
60		7th shift, ExOR A001h	1	1	1	1	1	0	0	1	1	1	0	1	1	0	1	0	1
61	8th shift	0	1	1	1	1	1	0	0	1	1	1	0	1	1	0	1	0	
62	-	Convert result to hexadecimal value	7				C				E				D				-

⇒ CRC data is "ED7Ch". Transmitted data including check sum will be 010800001234ED7C

### Caution

**When appending the CRC field to the transmitted message, note that the low-order byte "CRC16(L)" is appended first, followed by the high-order byte "CRC16(H)".**

## 8.3 Communication Response Time Guides

The time (communication response time) between when an upper level device sends the command and

when the command is received depends on the command.

When requesting each command, the time taken for communication should be calculated referring the table below.

Function	Communication response time [ms] (guideline)					
	MODBUS ASCII			MODBUS RTU		
	2400bps	19200bps	57600bps	2400bps	19200bps	57600bps
Reading of Step Data /Direct Operation Step Data (1 word)	155	25	10	90	15	10
Setting of Step Data /Direct Operation Step Data (1 word)	190	30	10	110	20	10
Reading of input signal status (1 bit)	145	20	10	85	15	10
Reading of output signal status and internal flags (1 bit)	145	20	10	85	15	10
Reading of internal parameters (7 words)	270	40	15	150	25	10
Operation mode selection	165	25	10	95	15	10
Commands for virtual I/O operation by serial communication	195	30	15	110	20	10
Reading of alarm history (4 words)	215	30	15	120	20	10
Clearing of alarm history	245	35	15	135	20	10
Confirmation of communication connection	165	25	10	95	15	10

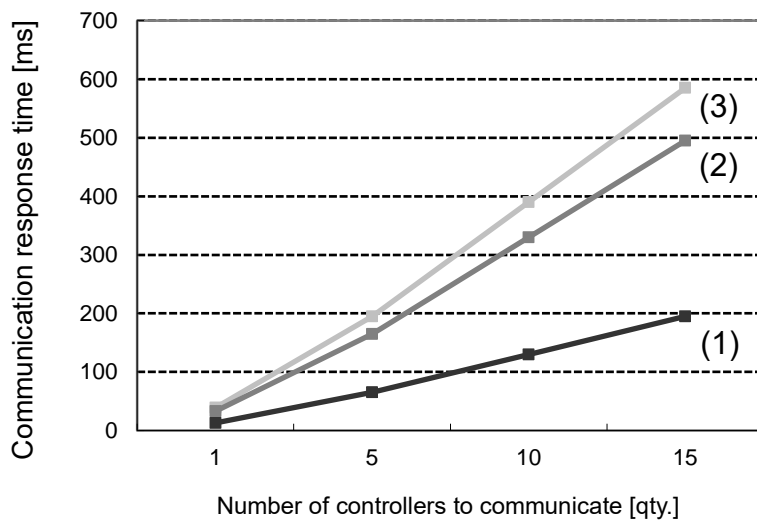
\* The communication response time does not include the processing time of the upper level device. When calculating the time taken for communication, take into account the processing time of the upper level equipment.

Calculation example)

When the MODBUS ASCII with a communication speed of 19200 bps requires five controllers to "read Step Data" and "Command for virtual I/O operation by serial communication"

$$\text{Communication response time } (25 + 30) \times 5 = 275 \text{ ms}$$

Guideline for communication response speed (Modbus ASCII 19200 bps)



- (1) "Confirmation of communication connection"
- (2) "Read internal parameter" + "Command for operation by serial communication"
- (3) "Read alarm history" + "Clear alarm history"

## 8.4 ASCII Code List

Decimal	Hexadecimal	ASCII	Decimal	Hexadecimal	ASCII	Decimal	Hexadecimal	ASCII	Decimal	Hexadecimal	ASCII
0	0	NUL	32	20	SP	64	40	@	96	60	`
1	1	SOH	33	21	!	65	41	A	97	61	a
2	2	STX	34	22	"	66	42	B	98	62	b
3	3	ETX	35	23	#	67	43	C	99	63	c
4	4	EOT	36	24	\$	68	44	D	100	64	d
5	5	ENQ	37	25	%	69	45	E	101	65	e
6	6	ACK	38	26	&	70	46	F	102	66	f
7	7	BEL	39	27	'	71	47	G	103	67	g
8	8	BS	40	28	(	72	48	H	104	68	h
9	9	HT	41	29	)	73	49	I	105	69	i
10	0A	LF	42	2A	*	74	4A	J	106	6A	j
11	0B	VT	43	2B	+	75	4B	K	107	6B	k
12	0C	FF	44	2C	,	76	4C	L	108	6C	l
13	0D	CR	45	2D	-	77	4D	M	109	6D	m
14	0E	SO	46	2E	.	78	4E	N	110	6E	n
15	0F	SI	47	2F	/	79	4F	O	111	6F	o
16	10	DLE	48	30	0	80	50	P	112	70	p
17	11	DC1	49	31	1	81	51	Q	113	71	q
18	12	DC2	50	32	2	82	52	R	114	72	r
19	13	DC3	51	33	3	83	53	S	115	73	s
20	14	DC4	52	34	4	84	54	T	116	74	t
21	15	NAK	53	35	5	85	55	U	117	75	u
22	16	SYN	54	36	6	86	56	V	118	76	v
23	17	ETB	55	37	7	87	57	W	119	77	w
24	18	CAN	56	38	8	88	58	X	120	78	x
25	19	EM	57	39	9	89	59	Y	121	79	y
26	1A	SUB	58	3A	:	90	5A	Z	122	7A	z
27	1B	ESC	59	3B	;	91	5B	[	123	7B	{
28	1C	FS	60	3C	<	92	5C	¥	124	7C	
29	1D	GS	61	3D	=	93	5D	]	125	7D	}
30	1E	RS	62	3E	>	94	5E	^	126	7E	~
31	1F	US	63	3F	?	95	5F	_	127	7F	DEL

#### Revision history

1st edition: September 2016  
A: Revised in June 2020  
B: Revised in January 2023  
C: Revised in February 2024

# SMC Corporation

Tel: + 81 3 5207 8249 Fax: +81 3 5298 5362

URL <https://www.smcworld.com>

---

Note: Specifications are subject to change without prior notice and any obligation on the part of the manufacturer.  
© SMC Corporation All Rights Reserved