

# TR1D Series

Communication Manual    MCT-TR1DC1-V2.3\_EN

Thank you for purchasing an Autonics product.

This user manual contains information about the product and its proper use,  
and should be kept in a place where it will be easy to access.

**Autonics**



# Contents

|                                                                                              |           |
|----------------------------------------------------------------------------------------------|-----------|
| <b>Preface</b> .....                                                                         | <b>5</b>  |
| <b>Manual Guide</b> .....                                                                    | <b>7</b>  |
| <b>Common Symbols in the Manual</b> .....                                                    | <b>9</b>  |
| <b>1. Modbus RTU Protocol</b> .....                                                          | <b>11</b> |
| 1.1. Function Code Frame Structure .....                                                     | 12        |
| 1.1.1. 01 (0x01) Read Coil Status .....                                                      | 12        |
| 1.1.2. 02 (0x02) Read Input Status .....                                                     | 14        |
| 1.1.3. 03 (0x03) Read Holding Registers .....                                                | 15        |
| 1.1.4. 04 (0x04) Read Input Registers .....                                                  | 16        |
| 1.1.5. 05 (0x05) Force Single Coil .....                                                     | 17        |
| 1.1.6. 06 (0x06) Preset Single Register .....                                                | 18        |
| 1.1.7. 16 (0x10) Preset Multiple Registers .....                                             | 19        |
| 1.2. Exception Handling .....                                                                | 21        |
| <b>2. Modbus ASCII Protocol</b> .....                                                        | <b>23</b> |
| 2.1. Function Code Frame Structure .....                                                     | 24        |
| 2.1.1. 01 (0x01) Read Coil Status .....                                                      | 24        |
| 2.1.2. 02 (0x02) Read Input Status .....                                                     | 26        |
| 2.1.3. 03 (0x03) Read Holding Registers .....                                                | 28        |
| 2.1.4. 04 (0x04) Read Input Registers .....                                                  | 30        |
| 2.1.5. 05 (0x05) Force Single Coil .....                                                     | 32        |
| 2.1.6. 06 (0x06) Preset Single Register .....                                                | 33        |
| 2.1.7. 16 (0x10) Preset Multiple Registers .....                                             | 34        |
| 2.2. Exception Handling .....                                                                | 36        |
| <b>3. Modbus Mapping Table</b> .....                                                         | <b>37</b> |
| 3.1. Read Coil Status.(Func 01) / Force Single Coil(Func 05) [Func:01/05, R/W : R/W] .....   | 37        |
| 3.2. Read Discrete Inputs.(Func 02) [Func:02, R/W : R] .....                                 | 38        |
| 3.3. Read Input Registers (Func 04)[Func:04, R/W : R] .....                                  | 39        |
| 3.4. Read Holding Register(Func 03) / Preset Single Register(Func 06)/.Preset Multiple ..... | 41        |
| Registers(Func 16). [Func:03/06/16, R/W : R/W]                                               |           |
| 3.4.1. SV setting group .....                                                                | 41        |
| 3.4.2. Parameter 1 group .....                                                               | 42        |

3.4.3. Parameter 2 group..... 44

# Preface

Thank you for purchasing Autonics products.

Be sure to read and follow the **Safety Precautions** thoroughly before use.

This manual contains information about the product and how to use it properly, so keep it in a place where users can easily find it.



# Manual Guide

- Use the product after fully reading the contents of the manual.
- The manual explains the product functions in detail and does not guarantee the contents other than the manual.
- Any or all of the manual may not be edited or copied without permission.
- The manual is not provided with the product.  
Download and use from our website ([www.autonics.com](http://www.autonics.com)).
- The contents of the manual are subject to change without prior notice according to the improvement of the product's performance, and upgrade notices are provided through our website.
- We put a lot of effort to make the contents of the manual a little easier and more accurate.  
Nevertheless, if you have any corrections or questions, please feel free to comment through our website.





# Common Symbols in the Manual



Failure to follow instructions may result in serious injury or death.



Failure to follow instructions may result in injury or product damage.



Supplementary explanation of the function



Example of that function



Important information about the feature



# 1. Modbus RTU Protocol

Modbus is an industrial protocol developed in 1979 for serial communication between devices. Its functionality has gradually expanded to allow implementation in TCP/IP and UDP environments, and it is now widely used for simple, stable, and efficient communication in various network environments. This protocol has become the standard communication method in industrial automation systems, SCADA (Supervisory Control and Data Acquisition systems), and other industrial networks.

Modbus RTU transmits data in a continuous binary format, making it more efficient and faster than ASCII transmission. The frame structure does not clearly distinguish the start and end, but defines the start and end by leaving a silent interval (3.5 character times) between frames.



For more detailed information about Modbus protocol, refer to the documentation provided by the developer.

# 1.1. Function Code Frame Structure

## 1.1.1. 01 (0x01) Read Coil Status

Reads the ON/OFF status of output (0X reference, coil) in the slave device.

### Request frame (master)

| Frame element     | Data size | Description                                                              |
|-------------------|-----------|--------------------------------------------------------------------------|
| Slave address     | 1-byte    | Address of the slave device                                              |
| Function code     | 1-byte    | 0x01 (Read Coil Status)                                                  |
| Starting address  | 2-byte    | Starting address of the coil to be read                                  |
| Quantity of coils | 2-byte    | Number of coils                                                          |
| CRC check         | 2-byte    | Adds a CRC-16 checksum at the end of the frame to verify data integrity. |

### Example of a request frame

| Slave address | Function code | Starting address |        | Quantity of coils |        | CRC check |        |
|---------------|---------------|------------------|--------|-------------------|--------|-----------|--------|
| 1-byte        | 1-byte        | High             | Low    | High              | Low    | Low       | High   |
|               |               | 1-byte           | 1-byte | 1-byte            | 1-byte | 1-byte    | 1-byte |
| CRC-16        |               |                  |        |                   |        |           |        |

**Response frame (slave)**

| Frame element | Data size | Description                                                                                                                                                                                                                                                                                 |
|---------------|-----------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Slave address | 1-byte    | Address of the slave device                                                                                                                                                                                                                                                                 |
| Function code | 1-byte    | 0x01 (Read Coil Status)                                                                                                                                                                                                                                                                     |
| Byte count    | 1-byte    | Number of bytes in the returned data                                                                                                                                                                                                                                                        |
| Coil status   | N-byte    | The status of the requested coils is represented.<br>Each coil's status is represented by 1 bit, and 8 coil statuses are grouped into 1 byte. <ul style="list-style-type: none"><li>• N = number of coils / 8</li><li>• If the remainder is not zero when divided by 8: N = N + 1</li></ul> |
| CRC check     | 2-byte    | Adds a CRC-16 checksum at the end of the frame to verify data integrity.                                                                                                                                                                                                                    |

**Example of a response frame**

| Slave address | Function code | Byte count | Coil status     | CRC check |        |
|---------------|---------------|------------|-----------------|-----------|--------|
| 1-byte        | 1-byte        | 1-byte     | <b>N</b> × byte | Low       | High   |
|               |               |            |                 | 1-byte    | 1-byte |
| CRC-16        |               |            |                 |           |        |

### 1.1.2. 02 (0x02) Read Input Status

Reads the ON/OFF status of input (1X reference) in the slave device.

#### Request frame (master)

| Frame element      | Data size | Description                                                              |
|--------------------|-----------|--------------------------------------------------------------------------|
| Slave address      | 1-byte    | Address of the slave device                                              |
| Function code      | 1-byte    | 0x02 (Read Input Status)                                                 |
| Starting address   | 2-byte    | Starting address of the input to be read                                 |
| Quantity of inputs | 2-byte    | Number of inputs                                                         |
| CRC check          | 2-byte    | Adds a CRC-16 checksum at the end of the frame to verify data integrity. |

#### Example of a request frame

| Slave address | Function code | Starting address |        | Quantity of inputs |        | CRC check |        |
|---------------|---------------|------------------|--------|--------------------|--------|-----------|--------|
| 1-byte        | 1-byte        | High             | Low    | High               | Low    | Low       | High   |
|               |               | 1-byte           | 1-byte | 1-byte             | 1-byte | 1-byte    | 1-byte |
| CRC-16        |               |                  |        |                    |        |           |        |

#### Response frame (slave)

| Frame element | Data size | Description                                                                                                                                                                                                                                                                                     |
|---------------|-----------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Slave address | 1-byte    | Address of the slave device                                                                                                                                                                                                                                                                     |
| Function code | 1-byte    | 0x02 (Read Input Status)                                                                                                                                                                                                                                                                        |
| Byte count    | 1-byte    | Number of bytes in the returned data                                                                                                                                                                                                                                                            |
| Input status  | N-byte    | The status of the requested inputs is represented.<br>Each input's status is represented by 1 bit, and 8 input statuses are grouped into 1 byte. <ul style="list-style-type: none"><li>• N = number of inputs / 8</li><li>• If the remainder is not zero when divided by 8: N = N + 1</li></ul> |
| CRC check     | 2-byte    | Adds a CRC-16 checksum at the end of the frame to verify data integrity.                                                                                                                                                                                                                        |

#### Example of a response frame

| Slave address | Function code | Byte count | Input status | CRC check |        |
|---------------|---------------|------------|--------------|-----------|--------|
| 1-byte        | 1-byte        | 1-byte     | N × byte     | Low       | High   |
|               |               |            |              | 1-byte    | 1-byte |
| CRC-16        |               |            |              |           |        |

### 1.1.3. 03 (0x03) Read Holding Registers

Reads the binary data of holding registers (4X reference) in the slave device.

#### Request frame (master)

| Frame element         | Data size | Description                                                              |
|-----------------------|-----------|--------------------------------------------------------------------------|
| Slave address         | 1-byte    | Address of the slave device                                              |
| Function code         | 1-byte    | 0x03 (Read Holding Registers)                                            |
| Starting address      | 2-byte    | Starting address of the first register to be read                        |
| Quantity of registers | 2-byte    | Number of registers                                                      |
| CRC check             | 2-byte    | Adds a CRC-16 checksum at the end of the frame to verify data integrity. |

#### Example of a request frame

| Slave address | Function code | Starting address |        | Quantity of registers |        | CRC check |        |
|---------------|---------------|------------------|--------|-----------------------|--------|-----------|--------|
| 1-byte        | 1-byte        | High             | Low    | High                  | Low    | Low       | High   |
|               |               | 1-byte           | 1-byte | 1-byte                | 1-byte | 1-byte    | 1-byte |
| CRC-16        |               |                  |        |                       |        |           |        |

#### Response frame (slave)

| Frame element  | Data size  | Description                                                                                                                                                   |
|----------------|------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Slave address  | 1-byte     | Address of the slave device                                                                                                                                   |
| Function code  | 1-byte     | 0x03 (Read Holding Registers)                                                                                                                                 |
| Byte count     | 1-byte     | Number of bytes in the returned data<br>(Number of registers read × 2-byte)                                                                                   |
| Register value | N × 2-byte | The values of the requested registers are represented.<br>Each register is represented by 2 bytes.<br>• N is determined by the number of requested registers. |
| CRC check      | 2-byte     | Adds a CRC-16 checksum at the end of the frame to verify data integrity.                                                                                      |

#### Example of a response frame

| Slave address | Function code | Byte count | Register value |               | CRC check |        |
|---------------|---------------|------------|----------------|---------------|-----------|--------|
| 1-byte        | 1-byte        | 1-byte     | High           | Low           | Low       | High   |
|               |               |            | <b>1-byte</b>  | <b>1-byte</b> | 1-byte    | 1-byte |
| CRC-16        |               |            |                |               |           |        |

### 1.1.4. 04 (0x04) Read Input Registers

Reads the binary data of input registers (3X reference) in the slave device.

#### Request frame (master)

| Frame element               | Data size | Description                                                              |
|-----------------------------|-----------|--------------------------------------------------------------------------|
| Slave address               | 1-byte    | Address of the slave device                                              |
| Function code               | 1-byte    | 0x04 (Read Input Registers)                                              |
| Starting address            | 2-byte    | Starting address of the first input register to be read                  |
| Quantity of input registers | 2-byte    | Number of input registers                                                |
| CRC check                   | 2-byte    | Adds a CRC-16 checksum at the end of the frame to verify data integrity. |

#### Example of a request frame

| Slave address | Function code | Starting address |        | Quantity of input registers |        | CRC check |        |
|---------------|---------------|------------------|--------|-----------------------------|--------|-----------|--------|
| 1-byte        | 1-byte        | High             | Low    | High                        | Low    | Low       | High   |
|               |               | 1-byte           | 1-byte | 1-byte                      | 1-byte | 1-byte    | 1-byte |
| CRC-16        |               |                  |        |                             |        |           |        |

#### Response frame (slave)

| Frame element  | Data size  | Description                                                                                                                                                         |
|----------------|------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Slave address  | 1-byte     | Address of the slave device                                                                                                                                         |
| Function code  | 1-byte     | 0x04 (Read Input Registers)                                                                                                                                         |
| Byte count     | 1-byte     | Number of bytes in the returned data<br>(Number of registers read × 2-byte)                                                                                         |
| Register value | N × 2-byte | The values of the requested input registers are represented.<br>Each register is represented by 2 bytes.<br>• N is determined by the number of requested registers. |
| CRC check      | 2-byte     | Adds a CRC-16 checksum at the end of the frame to verify data integrity.                                                                                            |

#### Example of a response frame

| Slave address | Function code | Byte count | Register value |               | CRC check |        |
|---------------|---------------|------------|----------------|---------------|-----------|--------|
| 1-byte        | 1-byte        | 1-byte     | High           | Low           | Low       | High   |
|               |               |            | <b>1-byte</b>  | <b>1-byte</b> | 1-byte    | 1-byte |
| CRC-16        |               |            |                |               |           |        |



### 1.1.5. 05 (0x05) Force Single Coil

Forces a single coil (0X reference) to either ON (0xFF00) or OFF (0x0000) status in the slave device.

#### Request frame (master)

| Frame element | Data size | Description                                                              |
|---------------|-----------|--------------------------------------------------------------------------|
| Slave address | 1-byte    | Address of the slave device                                              |
| Function code | 1-byte    | 0x05 (Force Single Coil)                                                 |
| Coil address  | 2-byte    | Address of the coil to be forced                                         |
| Force data    | 2-byte    | Sets or resets the coil. (0xFF00 = ON, 0x0000 = OFF)                     |
| CRC check     | 2-byte    | Adds a CRC-16 checksum at the end of the frame to verify data integrity. |

#### Example of a request frame

| Slave address | Function code | Coil address |        | Force data |        | CRC check |        |
|---------------|---------------|--------------|--------|------------|--------|-----------|--------|
| 1-byte        | 1-byte        | High         | Low    | High       | Low    | Low       | High   |
|               |               | 1-byte       | 1-byte | 1-byte     | 1-byte | 1-byte    | 1-byte |
| CRC-16        |               |              |        |            |        |           |        |

#### Response frame (slave)

| Frame element | Data size | Description                                                              |
|---------------|-----------|--------------------------------------------------------------------------|
| Slave address | 1-byte    | Address of the slave device                                              |
| Function code | 1-byte    | 0x05 (Force Single Coil)                                                 |
| Coil address  | 2-byte    | Address of the coil that was forced                                      |
| Force data    | 2-byte    | The status of the coil is represented.<br>(0xFF00 = ON, 0x0000 = OFF)    |
| CRC check     | 2-byte    | Adds a CRC-16 checksum at the end of the frame to verify data integrity. |

#### Example of a response frame

| Slave address | Function code | Coil address |        | Force data |        | CRC check |        |
|---------------|---------------|--------------|--------|------------|--------|-----------|--------|
| 1-byte        | 1-byte        | High         | Low    | High       | Low    | Low       | High   |
|               |               | 1-byte       | 1-byte | 1-byte     | 1-byte | 1-byte    | 1-byte |
| CRC-16        |               |              |        |            |        |           |        |

### 1.1.6. 06 (0x06) Preset Single Register

Writes the binary data to a single holding register (4X reference) in the slave device.

#### Request frame (master)

| Frame element    | Data size | Description                                                              |
|------------------|-----------|--------------------------------------------------------------------------|
| Slave address    | 1-byte    | Address of the slave device                                              |
| Function code    | 1-byte    | 0x06 (Preset Single Register)                                            |
| Register address | 2-byte    | Address of the register to be preset                                     |
| Preset data      | 2-byte    | Value to be preset                                                       |
| CRC check        | 2-byte    | Adds a CRC-16 checksum at the end of the frame to verify data integrity. |

#### Example of a request frame

| Slave address | Function code | Register address |        | Preset data |        | CRC check |        |
|---------------|---------------|------------------|--------|-------------|--------|-----------|--------|
| 1-byte        | 1-byte        | High             | Low    | High        | Low    | Low       | High   |
|               |               | 1-byte           | 1-byte | 1-byte      | 1-byte | 1-byte    | 1-byte |
| CRC-16        |               |                  |        |             |        |           |        |

#### Response frame (slave)

| Frame element    | Data size | Description                                                              |
|------------------|-----------|--------------------------------------------------------------------------|
| Slave address    | 1-byte    | Address of the slave device                                              |
| Function code    | 1-byte    | 0x06 (Preset Single Register)                                            |
| Register address | 2-byte    | Address of the register that was preset                                  |
| Preset data      | 2-byte    | Value preset to the register                                             |
| CRC check        | 2-byte    | Adds a CRC-16 checksum at the end of the frame to verify data integrity. |

#### Example of a response frame

| Slave address | Function code | Register address |        | Preset data |        | CRC check |        |
|---------------|---------------|------------------|--------|-------------|--------|-----------|--------|
| 1-byte        | 1-byte        | High             | Low    | High        | Low    | Low       | High   |
|               |               | 1-byte           | 1-byte | 1-byte      | 1-byte | 1-byte    | 1-byte |
| CRC-16        |               |                  |        |             |        |           |        |

### 1.1.7. 16 (0x10) Preset Multiple Registers

Writes the binary data continuously to multiple holding registers (4X reference) in the slave device.

#### Request frame (master)

| Frame element         | Data size          | Description                                                                                                             |
|-----------------------|--------------------|-------------------------------------------------------------------------------------------------------------------------|
| Slave address         | 1-byte             | Address of the slave device                                                                                             |
| Function code         | 1-byte             | 0x10 (Preset Multiple Registers)                                                                                        |
| Starting address      | 2-byte             | Starting address of the first register to be preset                                                                     |
| Quantity of registers | 2-byte             | Number of registers to be preset                                                                                        |
| Byte count            | 1-byte             | Number of bytes for the register values to be preset<br>(Number of registers to be preset $\times$ 2-byte)              |
| Register values       | $N \times 2$ -byte | The values of the registers to be preset are represented.<br>• N is determined by the number of registers to be preset. |
| CRC check             | 2-byte             | Adds a CRC-16 checksum at the end of the frame to verify data integrity.                                                |

#### Example of a request frame

| Slave addr. | Func. code | Starting addr. |        | Quantity of registers |        | Byte count | Register values |               | CRC check |        |
|-------------|------------|----------------|--------|-----------------------|--------|------------|-----------------|---------------|-----------|--------|
| 1-byte      | 1-byte     | High           | Low    | High                  | Low    | 1-byte     | High            | Low           | Low       | High   |
|             |            | 1-byte         | 1-byte | 1-byte                | 1-byte |            | <b>1-byte</b>   | <b>1-byte</b> | 1-byte    | 1-byte |
| CRC-16      |            |                |        |                       |        |            |                 |               |           |        |

Response frame (slave)

| Frame element         | Data size | Description                                                              |
|-----------------------|-----------|--------------------------------------------------------------------------|
| Slave address         | 1-byte    | Address of the slave device                                              |
| Function code         | 1-byte    | 0x10 (Preset Multiple Registers)                                         |
| Starting address      | 2-byte    | Starting address of the first register that was preset                   |
| Quantity of registers | 2-byte    | Number of registers that were preset                                     |
| CRC check             | 2-byte    | Adds a CRC-16 checksum at the end of the frame to verify data integrity. |

Example of a response frame

| Slave address | Function code | Starting address |        | Quantity of registers |        | CRC check |        |
|---------------|---------------|------------------|--------|-----------------------|--------|-----------|--------|
| 1-byte        | 1-byte        | High             | Low    | High                  | Low    | Low       | High   |
|               |               | 1-byte           | 1-byte | 1-byte                | 1-byte | 1-byte    | 1-byte |
| CRC-16        |               |                  |        |                       |        |           |        |



Except when downloading parameters with predefined minimum, maximum, or default values based on the input specifications in the PC loader program, it is recommended to use Preset Single Register rather than Preset Multiple Registers when connecting to external devices such as PLCs or graphic panels.

## 1.2. Exception Handling

In case of a communication error, the slave device sets the highest bit of the received function code to 1 in its response. It then sends only the exception code, without including any data related to the original request.

### Exception response frame

| Slave address | Function code + 0x80 | Exception code | CRC check |
|---------------|----------------------|----------------|-----------|
| 1-byte        | 1-byte               | 1-byte         | 2-byte    |
| CRC-16        |                      |                |           |

### Exception codes

| Exception code | Code name            | Description                                                                                                            |
|----------------|----------------------|------------------------------------------------------------------------------------------------------------------------|
| 01             | ILLEGAL FUNCTION     | If the command is not supported.                                                                                       |
| 02             | ILLEGAL DATA ADDRESS | If the requested data address is invalid.<br>(e.g. The data address does not exist or the address range is incorrect.) |
| 03             | ILLEGAL DATA VALUE   | If the requested data value is invalid.<br>(e.g. The data value is out of the allowable range.)                        |
| 04             | SLAVE DEVICE FAILURE | If the parameter is locked, communication write is prohibited, or the command cannot be processed correctly.           |
| 06             | SLAVE DEVICE BUSY    | If the device is in a state where it cannot execute the requested command.                                             |



## 2. Modbus ASCII Protocol

Modbus is an industrial protocol developed in 1979 for serial communication between devices. Its functionality has gradually expanded to allow implementation in TCP/IP and UDP environments, and it is now widely used for simple, stable, and efficient communication in various network environments. This protocol has become the standard communication method in industrial automation systems, SCADA (Supervisory Control and Data Acquisition systems), and other industrial networks.

Modbus ASCII converts each byte into ASCII code for transmission, which improves data readability but reduces transmission efficiency and speed. The frame structure uses a start character (':') and an end character (CRLF) to clearly distinguish each frame.



For more detailed information about Modbus protocol, refer to the documentation provided by the developer.

# 2.1. Function Code Frame Structure

## 2.1.1. 01 (0x01) Read Coil Status

Reads the ON/OFF status of output (0X reference, coil) in the slave device.

### Request frame (master)

| Frame element     | Data size | Description                                                            |
|-------------------|-----------|------------------------------------------------------------------------|
| Start             | 1-char    | : (Colon, 0x3A)                                                        |
| Slave address     | 2-char    | Address of the slave device                                            |
| Function code     | 2-char    | 0x01 (Read Coil Status)                                                |
| Starting address  | 4-char    | Starting address of the coil to be read                                |
| Quantity of coils | 4-char    | Number of coils                                                        |
| LRC check         | 2-char    | Adds an LRC checksum at the end of the frame to verify data integrity. |
| End               | 2-char    | CR + LF (0x0D + 0x0A)                                                  |

### Example of a request frame

|        |               |               |                  |                   |           |        |
|--------|---------------|---------------|------------------|-------------------|-----------|--------|
| Start  | Slave address | Function code | Starting address | Quantity of coils | LRC check | End    |
| 1-char | 2-char        | 2-char        | 4-char           | 4-char            | 2-char    | 2-char |
| :      | LRC           |               |                  |                   |           | CRLF   |



**Response frame (slave)**

| Frame element | Data size | Description                                                                                                                                                                                                                                                                                                                                                                                 |
|---------------|-----------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Start         | 1-char    | : (Colon, 0x3A)                                                                                                                                                                                                                                                                                                                                                                             |
| Slave address | 2-char    | Address of the slave device                                                                                                                                                                                                                                                                                                                                                                 |
| Function code | 2-char    | 0x01 (Read Coil Status)                                                                                                                                                                                                                                                                                                                                                                     |
| Byte count    | 2-char    | Number of bytes in the returned data                                                                                                                                                                                                                                                                                                                                                        |
| Coil status   | N-char    | The status of the requested coils is represented.<br>Each coil's status is represented by 1 bit, and 8 coil statuses are grouped into 1 byte. <ul style="list-style-type: none"><li>• Final data size (ASCII characters): <math>N \times 2</math></li><li>• <math>N</math> = number of coils / 8</li><li>• If the remainder is not zero when divided by 8: <math>N = N + 1</math></li></ul> |
| LRC check     | 2-char    | Adds an LRC checksum at the end of the frame to verify data integrity.                                                                                                                                                                                                                                                                                                                      |
| End           | 2-char    | CR + LF (0x0D + 0x0A)                                                                                                                                                                                                                                                                                                                                                                       |

**Example of a response frame**

|        |               |               |            |             |           |        |
|--------|---------------|---------------|------------|-------------|-----------|--------|
| Start  | Slave address | Function code | Byte count | Coil status | LRC check | End    |
| 1-char | 2-char        | 2-char        | 2-char     | N-char      | 2-char    | 2-char |
| :      | LRC           |               |            |             |           | CRLF   |

### 2.1.2. 02 (0x02) Read Input Status

Reads the ON/OFF status of input (1X reference) in the slave device.

#### Request frame (master)

| Frame element      | Data size | Description                                                            |
|--------------------|-----------|------------------------------------------------------------------------|
| Start              | 1-char    | : (Colon, 0x3A)                                                        |
| Slave address      | 2-char    | Address of the slave device                                            |
| Function code      | 2-char    | 0x02 (Read Input Status)                                               |
| Starting address   | 4-char    | Starting address of the input to be read                               |
| Quantity of inputs | 4-char    | Number of inputs                                                       |
| LRC check          | 2-char    | Adds an LRC checksum at the end of the frame to verify data integrity. |
| End                | 2-char    | CR + LF (0x0D + 0x0A)                                                  |

#### Example of a request frame

|        |               |               |                  |                    |           |        |
|--------|---------------|---------------|------------------|--------------------|-----------|--------|
| Start  | Slave address | Function code | Starting address | Quantity of inputs | LRC check | End    |
| 1-char | 2-char        | 2-char        | 4-char           | 4-char             | 2-char    | 2-char |
| :      | LRC           |               |                  |                    |           | CRLF   |

### Response frame (slave)

| Frame element | Data size | Description                                                                                                                                                                                                                                                                                                                                                                                     |
|---------------|-----------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Start         | 1-char    | : (Colon, 0x3A)                                                                                                                                                                                                                                                                                                                                                                                 |
| Slave address | 2-char    | Address of the slave device                                                                                                                                                                                                                                                                                                                                                                     |
| Function code | 2-char    | 0x02 (Read Input Status)                                                                                                                                                                                                                                                                                                                                                                        |
| Byte count    | 2-char    | Number of bytes in the returned data                                                                                                                                                                                                                                                                                                                                                            |
| Input status  | N-char    | The status of the requested inputs is represented.<br>Each input's status is represented by 1 bit, and 8 input statuses are grouped into 1 byte. <ul style="list-style-type: none"><li>• Final data size (ASCII characters): <math>N \times 2</math></li><li>• <math>N</math> = number of inputs / 8</li><li>• If the remainder is not zero when divided by 8: <math>N = N + 1</math></li></ul> |
| LRC check     | 2-char    | Adds an LRC checksum at the end of the frame to verify data integrity.                                                                                                                                                                                                                                                                                                                          |
| End           | 2-char    | CR + LF (0x0D + 0x0A)                                                                                                                                                                                                                                                                                                                                                                           |

### Example of a response frame

| Start  | Slave address | Function code | Byte count | Input status | LRC check | End    |
|--------|---------------|---------------|------------|--------------|-----------|--------|
| 1-char | 2-char        | 2-char        | 2-char     | N-char       | 2-char    | 2-char |
| :      | LRC           |               |            |              |           | CRLF   |

### 2.1.3. 03 (0x03) Read Holding Registers

Reads the binary data of holding registers (4X reference) in the slave device.

#### Request frame (master)

| Frame element         | Data size | Description                                                            |
|-----------------------|-----------|------------------------------------------------------------------------|
| Start                 | 1-char    | : (Colon, 0x3A)                                                        |
| Slave address         | 2-char    | Address of the slave device                                            |
| Function code         | 2-char    | 0x03 (Read Holding Registers)                                          |
| Starting address      | 4-char    | Starting address of the first register to be read                      |
| Quantity of registers | 4-char    | Number of registers                                                    |
| LRC check             | 2-char    | Adds an LRC checksum at the end of the frame to verify data integrity. |
| End                   | 2-char    | CR + LF (0x0D + 0x0A)                                                  |

#### Example of a request frame

|        |               |               |                  |                       |           |        |
|--------|---------------|---------------|------------------|-----------------------|-----------|--------|
| Start  | Slave address | Function code | Starting address | Quantity of registers | LRC check | End    |
| 1-char | 2-char        | 2-char        | 4-char           | 4-char                | 2-char    | 2-char |
| :      | LRC           |               |                  |                       |           | CRLF   |

**Response frame (slave)**

| Frame element   | Data size | Description                                                                                                                                                              |
|-----------------|-----------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Start           | 1-char    | : (Colon, 0x3A)                                                                                                                                                          |
| Slave address   | 2-char    | Address of the slave device                                                                                                                                              |
| Function code   | 2-char    | 0x03 (Read Holding Registers)                                                                                                                                            |
| Byte count      | 2-char    | Number of bytes in the returned data<br>(Number of registers read × 2-byte)                                                                                              |
| Register values | N-char    | The values of the requested registers are represented.<br>Each register is converted into 4 ASCII characters before transmission.<br>• N = (The value of byte count) × 2 |
| LRC check       | 2-char    | Adds an LRC checksum at the end of the frame to verify data integrity.                                                                                                   |
| End             | 2-char    | CR + LF (0x0D + 0x0A)                                                                                                                                                    |

**Example of a response frame**

| Start  | Slave address | Function code | Byte count | Register values | LRC check | End    |
|--------|---------------|---------------|------------|-----------------|-----------|--------|
| 1-char | 2-char        | 2-char        | 2-char     | N-char          | 2-char    | 2-char |
| :      | LRC           |               |            |                 |           | CRLF   |

### 2.1.4. 04 (0x04) Read Input Registers

Reads the binary data of input registers (3X reference) in the slave device.

#### Request frame (master)

| Frame element         | Data size | Description                                                            |
|-----------------------|-----------|------------------------------------------------------------------------|
| Start                 | 1-char    | : (Colon, 0x3A)                                                        |
| Slave address         | 2-char    | Address of the slave device                                            |
| Function code         | 2-char    | 0x04 (Read Input Registers)                                            |
| Starting address      | 4-char    | Starting address of the first input register to be read                |
| Quantity of registers | 4-char    | Number of input registers                                              |
| LRC check             | 2-char    | Adds an LRC checksum at the end of the frame to verify data integrity. |
| End                   | 2-char    | CR + LF (0x0D + 0x0A)                                                  |

#### Example of a request frame

|        |               |               |                  |                       |           |        |
|--------|---------------|---------------|------------------|-----------------------|-----------|--------|
| Start  | Slave address | Function code | Starting address | Quantity of registers | LRC check | End    |
| 1-char | 2-char        | 2-char        | 4-char           | 4-char                | 2-char    | 2-char |
| :      | LRC           |               |                  |                       |           | CRLF   |

**Response frame (slave)**

| Frame element   | Data size | Description                                                                                                                                                              |
|-----------------|-----------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Start           | 1-char    | : (Colon, 0x3A)                                                                                                                                                          |
| Slave address   | 2-char    | Address of the slave device                                                                                                                                              |
| Function code   | 2-char    | 0x04 (Read Input Registers)                                                                                                                                              |
| Byte count      | 2-char    | Number of bytes in the returned data<br>(Number of registers read × 2-byte)                                                                                              |
| Register values | N-char    | The values of the requested registers are represented.<br>Each register is converted into 4 ASCII characters before transmission.<br>• N = (The value of byte count) × 2 |
| LRC check       | 2-char    | Adds an LRC checksum at the end of the frame to verify data integrity.                                                                                                   |
| End             | 2-char    | CR + LF (0x0D + 0x0A)                                                                                                                                                    |

**Example of a response frame**

| Start  | Slave address | Function code | Byte count | Register values | LRC check | End    |
|--------|---------------|---------------|------------|-----------------|-----------|--------|
| 1-char | 2-char        | 2-char        | 2-char     | N-char          | 2-char    | 2-char |
| :      | LRC           |               |            |                 |           | CRLF   |

### 2.1.5. 05 (0x05) Force Single Coil

Forces a single coil (0X reference) to either ON (0xFF00) or OFF (0x0000) status in the slave device.

#### Request frame (master)

| Frame element | Data size | Description                                                            |
|---------------|-----------|------------------------------------------------------------------------|
| Start         | 1-char    | : (Colon, 0x3A)                                                        |
| Slave address | 2-char    | Address of the slave device                                            |
| Function code | 2-char    | 0x05 (Force Single Coil)                                               |
| Coil address  | 4-char    | Address of the coil to be forced                                       |
| Force data    | 4-char    | Sets or resets the coil. (0xFF00 = ON, 0x0000 = OFF)                   |
| LRC check     | 2-char    | Adds an LRC checksum at the end of the frame to verify data integrity. |
| End           | 2-char    | CR + LF (0x0D + 0x0A)                                                  |

#### Example of a request frame

|        |               |               |              |            |           |        |
|--------|---------------|---------------|--------------|------------|-----------|--------|
| Start  | Slave address | Function code | Coil address | Force data | LRC check | End    |
| 1-char | 2-char        | 2-char        | 4-char       | 4-char     | 2-char    | 2-char |
| :      | LRC           |               |              |            |           | CRLF   |

#### Response frame (slave)

| Frame element | Data size | Description                                                            |
|---------------|-----------|------------------------------------------------------------------------|
| Start         | 1-char    | : (Colon, 0x3A)                                                        |
| Slave address | 2-char    | Address of the slave device                                            |
| Function code | 2-char    | 0x05 (Force Single Coil)                                               |
| Coil address  | 4-char    | Address of the coil that was forced                                    |
| Force data    | 4-char    | The status of the coil is represented.<br>(0xFF00 = ON, 0x0000 = OFF)  |
| LRC check     | 2-char    | Adds an LRC checksum at the end of the frame to verify data integrity. |
| End           | 2-char    | CR + LF (0x0D + 0x0A)                                                  |

#### Example of a response frame

|        |               |               |              |            |           |        |
|--------|---------------|---------------|--------------|------------|-----------|--------|
| Start  | Slave address | Function code | Coil address | Force data | LRC check | End    |
| 1-char | 2-char        | 2-char        | 4-char       | 4-char     | 2-char    | 2-char |
| :      | LRC           |               |              |            |           | CRLF   |



### 2.1.6. 06 (0x06) Preset Single Register

Writes the binary data to a single holding register (4X reference) in the slave device.

#### Request frame (master)

| Frame element    | Data size | Description                                                            |
|------------------|-----------|------------------------------------------------------------------------|
| Start            | 1-char    | : (Colon, 0x3A)                                                        |
| Slave address    | 2-char    | Address of the slave device                                            |
| Function code    | 2-char    | 0x06 (Preset Single Register)                                          |
| Register address | 4-char    | Address of the register to be preset                                   |
| Preset data      | 4-char    | Value to be preset                                                     |
| LRC check        | 2-char    | Adds an LRC checksum at the end of the frame to verify data integrity. |
| End              | 2-char    | CR + LF (0x0D + 0x0A)                                                  |

#### Example of a request frame

|        |               |               |                  |             |           |        |
|--------|---------------|---------------|------------------|-------------|-----------|--------|
| Start  | Slave address | Function code | Register address | Preset data | LRC check | End    |
| 1-char | 2-char        | 2-char        | 4-char           | 4-char      | 2-char    | 2-char |
| :      | LRC           |               |                  |             |           | CRLF   |

#### Response frame (slave)

| Frame element    | Data size | Description                                                            |
|------------------|-----------|------------------------------------------------------------------------|
| Start            | 1-char    | : (Colon, 0x3A)                                                        |
| Slave address    | 2-char    | Address of the slave device                                            |
| Function code    | 2-char    | 0x06 (Preset Single Register)                                          |
| Register address | 4-char    | Address of the register that was preset                                |
| Preset data      | 4-char    | Value preset to the register                                           |
| LRC check        | 2-char    | Adds an LRC checksum at the end of the frame to verify data integrity. |
| End              | 2-char    | CR + LF (0x0D + 0x0A)                                                  |

#### Example of a response frame

|        |               |               |                  |             |           |        |
|--------|---------------|---------------|------------------|-------------|-----------|--------|
| Start  | Slave address | Function code | Register address | Preset data | LRC check | End    |
| 1-char | 2-char        | 2-char        | 4-char           | 4-char      | 2-char    | 2-char |
| :      | LRC           |               |                  |             |           | CRLF   |

### 2.1.7. 16 (0x10) Preset Multiple Registers

Writes the binary data continuously to multiple holding registers (4X reference) in the slave device.

#### Request frame (master)

| Frame element         | Data size | Description                                                                                                                                                              |
|-----------------------|-----------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Start                 | 1-char    | : (Colon, 0x3A)                                                                                                                                                          |
| Slave address         | 2-char    | Address of the slave device                                                                                                                                              |
| Function code         | 2-char    | 0x10 (Preset Multiple Registers)                                                                                                                                         |
| Starting address      | 4-char    | Starting address of the first register to be preset                                                                                                                      |
| Quantity of registers | 4-char    | Number of registers to be preset                                                                                                                                         |
| Byte count            | 2-char    | Number of bytes for the register values to be preset<br>(Number of registers to be preset × 2-byte)                                                                      |
| Register values       | N-char    | The values of the requested registers are represented.<br>Each register is converted into 4 ASCII characters before transmission.<br>• N = (The value of byte count) × 2 |
| LRC check             | 2-char    | Adds an LRC checksum at the end of the frame to verify data integrity.                                                                                                   |
| End                   | 2-char    | CR + LF (0x0D + 0x0A)                                                                                                                                                    |

#### Example of a request frame

|        |               |               |                  |                       |            |                 |           |        |
|--------|---------------|---------------|------------------|-----------------------|------------|-----------------|-----------|--------|
| Start  | Slave address | Function code | Starting address | Quantity of registers | Byte count | Register values | LRC check | End    |
| 1-char | 2-char        | 2-char        | 4-char           | 4-char                | 2-char     | N-char          | 2-char    | 2-char |
| :      | LRC           |               |                  |                       |            |                 |           | CRLF   |

**Response frame (slave)**

| Frame element         | Data size | Description                                                            |
|-----------------------|-----------|------------------------------------------------------------------------|
| Start                 | 1-char    | : (Colon, 0x3A)                                                        |
| Slave address         | 2-char    | Address of the slave device                                            |
| Function code         | 2-char    | 0x10 (Preset Multiple Registers)                                       |
| Starting address      | 4-char    | Starting address of the first register that was preset                 |
| Quantity of registers | 4-char    | Number of registers that were preset                                   |
| LRC check             | 2-char    | Adds an LRC checksum at the end of the frame to verify data integrity. |
| End                   | 2-char    | CR + LF (0x0D + 0x0A)                                                  |

**Example of a response frame**

|        |               |               |                  |                       |           |        |
|--------|---------------|---------------|------------------|-----------------------|-----------|--------|
| Start  | Slave address | Function code | Starting address | Quantity of registers | LRC check | End    |
| 1-char | 2-char        | 2-char        | 4-char           | 4-char                | 2-char    | 2-char |
| :      | LRC           |               |                  |                       |           | CRLF   |



Except when downloading parameters with predefined minimum, maximum, or default values based on the input specifications in the PC loader program, it is recommended to use Preset Single Register rather than Preset Multiple Registers when connecting to external devices such as PLCs or graphic panels.

## 2.2. Exception Handling

In case of a communication error, the slave device sets the highest bit of the received function code to 1 in its response. It then sends only the exception code, without including any data related to the original request.

### Exception response frame

| Start  | Slave address | Function code + 0x80 | Exception code | LRC check | End    |
|--------|---------------|----------------------|----------------|-----------|--------|
| 1-char | 2-char        | 2-char               | 2-char         | 2-char    | 2-char |
| :      | LRC           |                      |                |           | CRLF   |

### Exception codes

| Exception code | Code name            | Description                                                                                                            |
|----------------|----------------------|------------------------------------------------------------------------------------------------------------------------|
| 01             | ILLEGAL FUNCTION     | If the command is not supported.                                                                                       |
| 02             | ILLEGAL DATA ADDRESS | If the requested data address is invalid.<br>(e.g. The data address does not exist or the address range is incorrect.) |
| 03             | ILLEGAL DATA VALUE   | If the requested data value is invalid.<br>(e.g. The data value is out of the allowable range.)                        |
| 04             | SLAVE DEVICE FAILURE | If the parameter is locked, communication write is prohibited, or the command cannot be processed correctly.           |
| 06             | SLAVE DEVICE BUSY    | If the device is in a state where it cannot execute the requested command.                                             |

# 3. Modbus Mapping Table

## The meaning of HI/LO address

Some parameters divide the value into two addresses (HI/LO) because it exceeds the maximum value ( $2^{16}$ , 16 bit) that can be sent from one address.

When inputting value greater than the maximum ( $2^{16}$ , 16 bit), convert it to hexadecimal (HEX), enter up to 4 digits (low 16bit) into the LO address and over 5 digits (high 16 bit) into the HI address.

For example, when inputting a value '99999' into 'Output hysteresis' parameter, convert '99999' into the hexadecimal value, "**1 869F**". Input '**1**'(0x0001) into HI address (OUT HYS HI [Output hysteresis], 400065(0040)) and '**869F**'(0x869F) into LO address (OUT HYS LO [Output hysteresis], 400066(0041)), and send them.

If under the maximum ( $2^{16}$ , 16 bit) is needed to be set, input '0'(0x0000) into HI address and the hexadecimal value (HEX) into LO address.

## 3.1. Read Coil Status.(Func 01) / Force Single Coil(Func 05) [Func:01/05, R/W : R/W]

| No. (Address)    | Classification | Description                               | Factory default |
|------------------|----------------|-------------------------------------------|-----------------|
| 000001 (0000)    | RUN/STOP       | Control output Run/Stop<br>0: RUN 1: STOP | RUN             |
| 000002 (0001)    | AT             | Auto-tuning Run/Stop<br>0: OFF 1: ON      | OFF             |
| 000003 (0002)    | Alarm Reset    | Alarm output reset<br>0: OFF 1: ON        | OFF             |
| 000004 to 000050 | Reserved       | -                                         | -               |

### 3.2. Read Discrete Inputs.(Func 02) [Func:02, R/W : R]

| No. (Address)    | Classification                    | Description                                                                 | Factory default |
|------------------|-----------------------------------|-----------------------------------------------------------------------------|-----------------|
| 100001 (0000)    | SV                                | SV front indicator<br>0: OFF 1: ON                                          | -               |
| 100002 (0001)    | °F                                | Unit front indicator<br>0: OFF 1: ON                                        | -               |
| 100003 (0002)    | °C                                | Unit front indicator<br>0: OFF 1: ON                                        | -               |
| 100004 (0003)    | OUT1                              | Control output 1 front indicator<br>0: OFF 1: ON                            | -               |
| 100005 (0004)    | OUT2 /<br>Transmission<br>output2 | Control output 2 / Transmission output 2 front<br>indicator<br>0: OFF 1: ON | -               |
| 100006 (0005)    | AL1                               | Alarm output 1 front indicator<br>0: OFF 1: ON                              | -               |
| 100007 (0006)    | AL2                               | Alarm output 2<br>0: OFF 1: ON                                              | -               |
| 100008 (0007)    | UP                                | Deviation high limit front indicator<br>0: OFF 1: ON                        | -               |
| 100009 (0008)    | MID                               | Front indicator<br>0: OFF 1: ON                                             | -               |
| 100010 (0009)    | DOWN                              | Deviation low limit front indicator<br>0: OFF 1: ON                         | -               |
| 100011 to 100050 | Reserved                          | -                                                                           | -               |

### 3.3. Read Input Registers (Func 04)[Func:04, R/W : R]

| No. (Address)    | Classification | Description                    | Factory default          |
|------------------|----------------|--------------------------------|--------------------------|
| 300001 to 300100 | Reserved       | -                              | -                        |
| 300101 (0064)    | -              | Product number H               | 1                        |
| 300102 (0065)    | -              | Product number L               | 5160                     |
| 300103 (0066)    | -              | Hardware version               | <input type="checkbox"/> |
| 300104 (0067)    | -              | Software version               | <input type="checkbox"/> |
| 300105 (0068)    | -              | Model name 1                   | "TR"                     |
| 300106 (0069)    | -              | Model name 2                   | "1D"                     |
| 300107 (006A)    | -              | Model name 3                   | "-1"<br>음션I/O            |
| 300108 (006B)    | -              | Model name 4                   | "4R"<br>전원/OUT<br>1      |
| 300109 (006C)    | -              | Model name 5                   | "N"<br>OUT2              |
| 300110 (006D)    | -              | Model name 6                   | " "                      |
| 300111 (006E)    | -              | Model name 7                   | " "                      |
| 300112 (006F)    | -              | Model name 8                   | " "                      |
| 300113 (0070)    | -              | Model name 9                   | " "                      |
| 300114 (0071)    | -              | Model name 10                  | " "                      |
| 300115 (0072)    | -              | Reserved                       | -                        |
| 300116 (0073)    | -              | Reserved                       | -                        |
| 300117 (0074)    | -              | Reserved                       | -                        |
| 300118 (0075)    | -              | Coil status Start Address      | 0001                     |
| 300119 (0076)    | -              | Coil status Quantity           | 3                        |
| 300120 (0077)    | -              | Input status Start Address     | 0001                     |
| 300121 (0078)    | -              | Input status Quantity          | 10                       |
| 300122 (0079)    | -              | Holding Register Start Address | 0001                     |
| 300123 (007A)    | -              | Holding Register Quantity      | 351                      |
| 300124 (007B)    | -              | Input Register Start Address   | 0001                     |
| 300125 (007C)    | -              | Input Register Quantity        | 1008                     |
| 300126 to 300200 | Reserved       | -                              | -                        |

| No. (Address)    | Classification            | Description                                | Factory default |
|------------------|---------------------------|--------------------------------------------|-----------------|
| 301001 (03E8)    | PV                        | Present value<br>-1999 to 9999 °C/°F       | -               |
| 301002 (03E9)    | DOT                       | Decimal point position<br>0: 0 1: 0.0      | 0               |
| 301003 (03EA)    | UNIT                      | Display unit<br>0: °C 1: °F                | °C              |
| 301004 (03EB)    | SV                        | Setting value<br>L-SV to H-SV °C/°F        | 0               |
| 301005 (03EC)    | Heater Current Monitoring | Heater current monitoring<br>0.0 to 50.0 A | 0.0             |
| 301006 (03ED)    | Front indicator           | Front indicator<br>0: OFF 1: ON            | -               |
| 301007 (03EE)    | HEATING MV                | Heating MV<br>0.0 to 100.0%                | -               |
| 301008 (03EF)    | COOLING MV                | Cooling MV<br>0.0 to 100.0%                | -               |
| 310009 to 310050 | Reserved                  | -                                          | -               |

#### Bit data consistence of 301006 (03ED) front display

| Bit   | Description                          | Set range |
|-------|--------------------------------------|-----------|
| Bit 0 | SV indicator                         | 0 or 1    |
| Bit 1 | °F indicator                         | 0 or 1    |
| Bit 2 | °C indicator                         | 0 or 1    |
| Bit 3 | OUT1 indicator                       | 0 or 1    |
| Bit 4 | OUT2/Transmission output 2 indicator | 0 or 1    |
| Bit 5 | AL1 indicator                        | 0 or 1    |
| Bit 6 | AL2 indicator                        | 0 or 1    |
| Bit 7 | UP indicator                         | 0 or 1    |
| Bit 8 | MID indicator                        | 0 or 1    |
| Bit 9 | DOWN indicator                       | 0 or 1    |



**3.4. Read Holding Register(Func 03) / Preset Single Register(Func 06) / Preset Multiple Registers(Func 16).**  
**[Func:03/06/16, R/W : R/W]**

**3.4.1. SV setting group**

| No. (Address)    | Parameter | Description                            | Factory default |
|------------------|-----------|----------------------------------------|-----------------|
| 400001 (0000)    | Set Value | SV setting value<br>L-SV to H-SV °C/°F | 0               |
| 400002 to 400050 | Reserved  | -                                      | -               |

### 3.4.2. Parameter 1 group

| No.(Address)  | Parameter | Description                                                                                              | Factory default |
|---------------|-----------|----------------------------------------------------------------------------------------------------------|-----------------|
| 400051 (0032) | LOCK      | Lock<br>0: OFF 1: LOC1 2: LOC2 3: LOC3                                                                   | OFF             |
| 400052 (0033) | CT-A      | Heater current monitoring<br>0 to 500: 00.0 to 50.0 A (display range)                                    | -               |
| 400053 (0034) | AT        | Auto-tuning<br>0 : OFF 1 : ON                                                                            | OFF             |
| 400054 (0035) | AL1       | AL1 alarm temperature<br>Deviation alarm: -F.S to F.S °C/°F<br>Absolute value alarm: input specification | 1250            |
| 400055 (0036) | AL2       | AL2 alarm temperature<br>Deviation alarm: -F.S to F.S °C/°F<br>Absolute value alarm: input specification | 1250            |
| 400056 (0037) | H-P       | Heating proportional band<br>0.1 to 999.9 °C/°F                                                          | 10.0            |
| 400057 (0038) | H-I       | Heating integral time<br>0 to 9999 sec.                                                                  | 240             |
| 400058 (0039) | H-D       | Heating derivative time<br>0 to 9999 sec.                                                                | 49              |
| 400059 (003A) | C-P       | Cooling proportional band<br>0.1 to 999.9 °C/°F                                                          | 10.0            |
| 400060 (003B) | C-I       | Cooling integral time<br>0 to 9999 sec.                                                                  | 240             |
| 400061 (003C) | C-D       | Cooling derivative time<br>0 to 9999 sec.                                                                | 49              |
| 400062 (003D) | DB        | Dead band<br>Temp. H<br>-999 to 999<br>Temp. L<br>-199.9 to 999.9 °C/°F                                  | 0               |
| 400063 (003E) | REST      | Manual reset<br>0.0 to 100.0 %                                                                           | 50.0            |
| 400064 (003F) | H.HYS     | Heating hysteresis<br>1 to 100 (0.1 to 100.0)                                                            | 2               |
| 400065 (0040) | H.OST     | Heating OFF<br>offset 1 to 100 (0.1 to 100.0)                                                            | 0               |

| No.(Address)     | Parameter | Description                                   | Factory default |
|------------------|-----------|-----------------------------------------------|-----------------|
| 400066 (0041)    | C.HYS     | Cooling hysteresis<br>1 to 100 (0.1 to 100.0) | 2               |
| 400067 (0042)    | C.OST     | Cooling OFF<br>offset 1 to 100 (0.1 to 100.0) | 0               |
| 400068 to 400100 | Reserved  | -                                             | -               |

### 3.4.3. Parameter 2 group

| No. (Address) | Parameter | Description                                                                                              | Factory default              |
|---------------|-----------|----------------------------------------------------------------------------------------------------------|------------------------------|
| 400101 (0064) | IN-T      | Input type<br>Refer to 'Input type'                                                                      | KCA.H                        |
| 400102 (0065) | UNIT      | Temperature unit<br>0: °C 1: °F                                                                          | °C                           |
| 400103 (0066) | SPL.T     | Sampling period<br>0: 50 ms 1: 100 ms 2: 250 ms                                                          | 50                           |
| 400104 (0067) | IN-B      | Input correction<br>-999 to 999 Digit (-199.9 to 999.9)                                                  | 0                            |
| 400105 (0068) | MAV.F     | Input digital filter<br>0.1 to 120.0 sec.                                                                | 0.1                          |
| 400106 (0069) | L-SV      | SV low limit value<br>Input range of input type °C/°F<br>$L-Sv \leq (H-Sv - 1\text{digit})$              | -50                          |
| 400107 (006A) | H-SV      | SV high limit value<br>Input range of input type °C/°F<br>$H-Sv \geq (L-Sv + 1\text{digit})$             | 1200                         |
| 400108 (006B) | O-FT      | Control output mode<br>0: HEAT 1: COOL 2: H-C                                                            | H-C                          |
| 400109 (006C) | C-MD      | Control type<br>[Standard control]<br>0: PID 1: ONOF<br>[H&C control]<br>0: P.P 1: P.ON 2: ON.P 3: ON.ON | PP                           |
| 400110 (006D) | OUT1      | Control output 1<br>0: CURR 1: SSR                                                                       | CURR                         |
| 400111 (006E) | O1.MA     | Control output 1 range<br>0: 4-20 mA 1: 0-20 mA                                                          | 4-20                         |
| 400112 (006F) | OUT2      | Control output 2<br>0: CURR 1: SSR                                                                       | CURR                         |
| 400113 (0070) | O2.MA     | Control output 2 range<br>0: 4-20 mA 1: 0-20 mA                                                          | 4-20 mA                      |
| 400114 (0071) | H-T       | Heating control cycle<br>0.5 to 120.0 sec.                                                               | [Relay]<br>20.0<br>[SSR] 2.0 |

| No. (Address) | Parameter | Description                                                                                                                                            | Factory default              |
|---------------|-----------|--------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------|
| 400115 (0072) | C-T       | Cooling control cycle<br>0.5 to 120.0 sec.                                                                                                             | [Relay]<br>20.0<br>[SSR] 2.0 |
| 400116 (0073) | AL-1      | AL1 alarm operation<br>Refer to 'Table. AL□ alarm operation'                                                                                           | AM1.A                        |
| 400117 (0074) | A1.HY     | AL1 hysteresis<br>1 to 100 °C/°F                                                                                                                       | 1                            |
| 400118 (0075) | AL-2      | AL2 alarm operation<br>Refer to 'Table. AL□ alarm operation'                                                                                           | AM2.A                        |
| 400119 (0076) | A2.HY     | AL2 hysteresis<br>1 to 100 °C/°F                                                                                                                       | 1                            |
| 400120 (0077) | LBA.T     | LBA time<br>0 to 9999 sec.                                                                                                                             | 0                            |
| 400121 (0078) | LBA.B     | LBA band<br>0 to 999 °C/°F (0 to 999.9)                                                                                                                | 2                            |
| 400122 (0079) | AO.M1     | Transmission output 1 mode<br>0: PV 1: SV 2: H-MV 3: C-MV                                                                                              | PV                           |
| 400123 (007A) | FS1.L     | Transmission output 1 low-limit<br>Input range of input type                                                                                           | -50                          |
| 400124 (007B) | FS1.H     | Transmission output 1 high-limit<br>Input range of input type                                                                                          | 1200                         |
| 400125 (007C) | AO.M2     | Transmission output 2 mode<br>0: PV 1: SV 2: H-MV 3: C-MV                                                                                              | PV                           |
| 400126 (007D) | FS2.L     | Transmission output 2 low-limit<br>Input range of input type                                                                                           | -50                          |
| 400127 (007E) | FS2.H     | Transmission output 2 high-limit<br>Input range of input type                                                                                          | 1200                         |
| 400128 (007F) | DI-K      | Digital input key<br>0: Not used 1: Stop control output 2: Auto-tuning execution 3: Alarm reset,                                                       | Stop control output          |
| 400129 (0080) | ER.MV     | Sensor error, MV<br>[Standard control] 0.0 (OFF) to 100.0 % (ON)<br>[Heating-Cooling control] -100.0 (cooling ON) to 0.0 (OFF) to 100.0 % (heating ON) | 0.0                          |
| 400130 (0081) | DSP       | Screen protection<br>0: OFF 1: 1 min. 2: 30 min. 3: 60 min.                                                                                            | OFF                          |

| No. (Address)    | Parameter | Description                                                                                            | Factory default |
|------------------|-----------|--------------------------------------------------------------------------------------------------------|-----------------|
| 400131 (0082)    | PRCL      | Communication protocol<br>0: RTU, 1: ASCII                                                             | RTU             |
| 400132 (0083)    | ADRS      | Communication address<br>1 to 99                                                                       | 1               |
| 400133 (0084)    | BPS       | Communication speed<br>0: 4800 bps 1: 9600 bps 2: 19200 bps 3: 38400 bps<br>4: 57600 bps 5: 115200 bps | 9600 bps        |
| 400134 (0085)    | PRTY      | Communication parity bit<br>0: NONE 1: EVEN 2: ODD                                                     | NONE            |
| 400135 (0086)    | STP       | Communication stop bit<br>0: 1 bit 1: 2 bit                                                            | 2 bit           |
| 400136 (0087)    | RSW.T     | Communication response time<br>5 to 99 ms                                                              | 20              |
| 400137 (0088)    | COMW      | Communication write<br>0: Enable 1: Disable                                                            | Enable          |
| 400138 (0089)    | INIT      | Parameter reset<br>0: OFF 1: ON                                                                        | OFF             |
| 400139 to 400150 | Reserved  | -                                                                                                      | -               |

**Table 1. AL□ alarm operation**

| Setting | Parameter | Alarm operation            | Alarm option               |
|---------|-----------|----------------------------|----------------------------|
| 0       | AM0       | Not used                   | -                          |
| 10      | AM1.A     | Deviation high limit alarm | Standard alarm             |
| 11      | AM1.B     | Deviation high limit alarm | Alarm latch                |
| 12      | AM1.C     | Deviation high limit alarm | Standby sequence 1         |
| 13      | AM1.D     | Deviation high limit alarm | Alarm latch and sequence 1 |
| 14      | AM1.E     | Deviation high limit alarm | Standby sequence 2         |
| 15      | AM1.F     | Deviation high limit alarm | Alarm latch and sequence 2 |
| 20      | AM2.A     | Deviation low limit alarm  | Standard alarm             |

| Setting | Parameter | Alarm operation                         | Alarm option               |
|---------|-----------|-----------------------------------------|----------------------------|
| 21      | AM2.B     | Deviation low limit alarm               | Alarm latch                |
| 22      | AM2.C     | Deviation low limit alarm               | Standby sequence 1         |
| 23      | AM2.D     | Deviation low limit alarm               | Alarm latch and sequence 1 |
| 24      | AM2.E     | Deviation low limit alarm               | Standby sequence 2         |
| 25      | AM2.F     | Deviation low limit alarm               | Alarm latch and sequence 2 |
| 30      | AM3.A     | Deviation high, low limit alarm         | Standard alarm             |
| 31      | AM3.B     | Deviation high, low limit alarm         | Alarm latch                |
| 32      | AM3.C     | Deviation high, low limit alarm         | Standby sequence 1         |
| 33      | AM3.D     | Deviation high, low limit alarm         | Alarm latch and sequence 1 |
| 34      | AM3.E     | Deviation high, low limit alarm         | Standby sequence 2         |
| 35      | AM3.F     | Deviation high, low limit alarm         | Alarm latch and sequence 2 |
| 40      | AM4.A     | Deviation high, low limit reserve alarm | Standard alarm             |
| 41      | AM4.B     | Deviation high, low limit reserve alarm | Alarm latch                |
| 42      | AM4.C     | Deviation high, low limit reserve alarm | Standby sequence 1         |
| 43      | AM4.D     | Deviation high, low limit reserve alarm | Alarm latch and sequence 1 |
| 44      | AM4.E     | Deviation high, low limit reserve alarm | Standby sequence 2         |
| 45      | AM4.F     | Deviation high, low limit reserve alarm | Alarm latch and sequence 2 |
| 50      | AM5.A     | Absolute value high limit alarm         | Standard alarm             |

| Setting | Parameter | Alarm operation                 | Alarm option               |
|---------|-----------|---------------------------------|----------------------------|
| 51      | AM5.B     | Absolute value high limit alarm | Alarm latch                |
| 52      | AM5.C     | Absolute value high limit alarm | Standby sequence 1         |
| 53      | AM5.D     | Absolute value high limit alarm | Alarm latch and sequence 1 |
| 54      | AM5.E     | Absolute value high limit alarm | Standby sequence 2         |
| 55      | AM5.F     | Absolute value high limit alarm | Alarm latch and sequence 2 |
| 60      | AM6.A     | Absolute value high limit alarm | Standard alarm             |
| 61      | AM6.B     | Absolute value high limit alarm | Alarm latch                |
| 62      | AM6.C     | Absolute value high limit alarm | Standby sequence 1         |
| 63      | AM6.D     | Absolute value high limit alarm | Alarm latch and sequence 1 |
| 64      | AM6.E     | Absolute value high limit alarm | Standby sequence 2         |
| 65      | AM6.F     | Absolute value high limit alarm | Alarm latch and sequence 2 |
| 70      | SBA.A     | Sensor break alarm              | Standard alarm             |
| 71      | SBA.B     | Sensor break alarm              | Alarm latch                |
| 80      | LBA.A     | Loop break alarm                | Standard alarm             |
| 81      | LBA.B     | Loop break alarm                | Alarm latch                |
| 90      | HBA.A     | Heater break alarm              | Standard alarm             |
| 91      | HBA.B     | Heater break alarm              | Alarm latch                |



# Autonics

Dimensions or specifications on this manual are subject to change and some models may be discontinued without notice.

[www.autonics.com](http://www.autonics.com)