

LoRa Module

Modbus Protocol & Modbus 프로그램 사용법

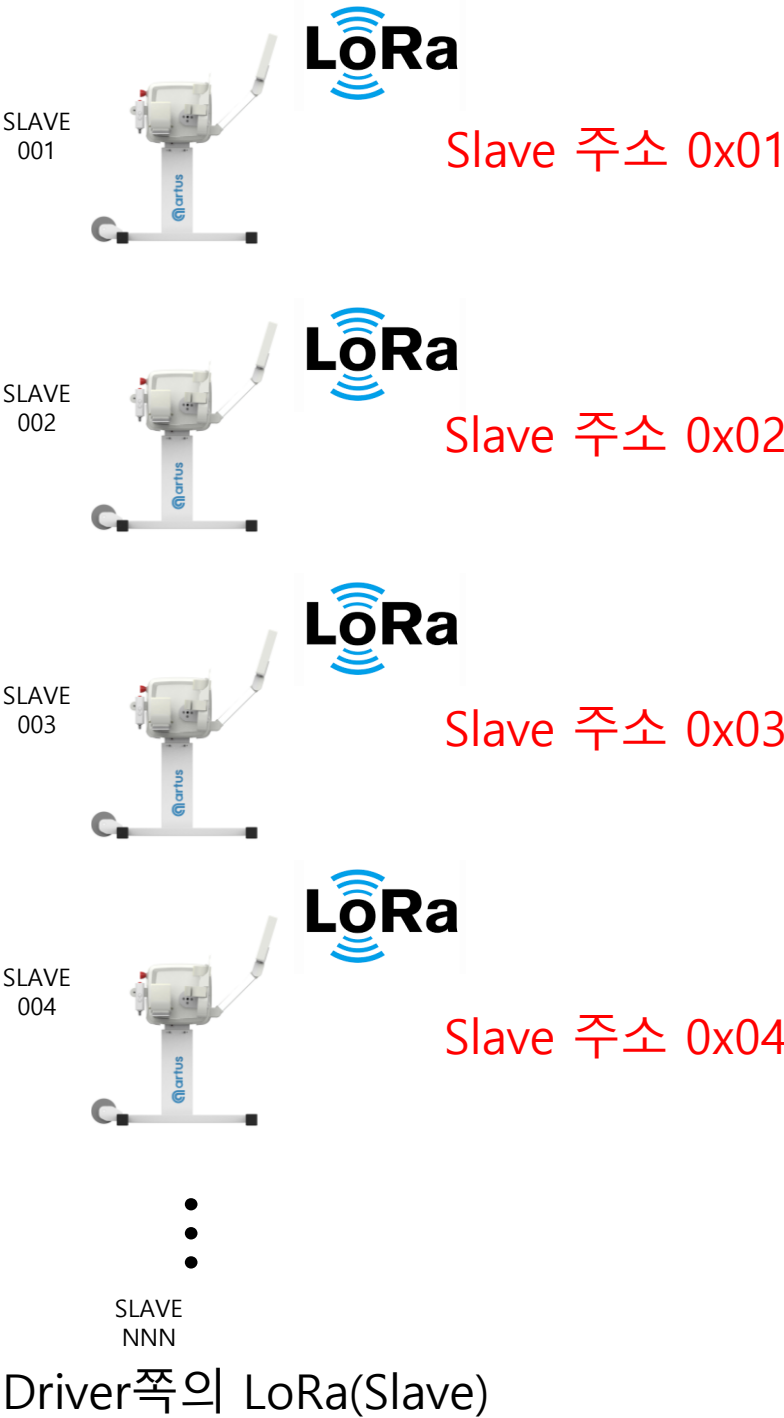
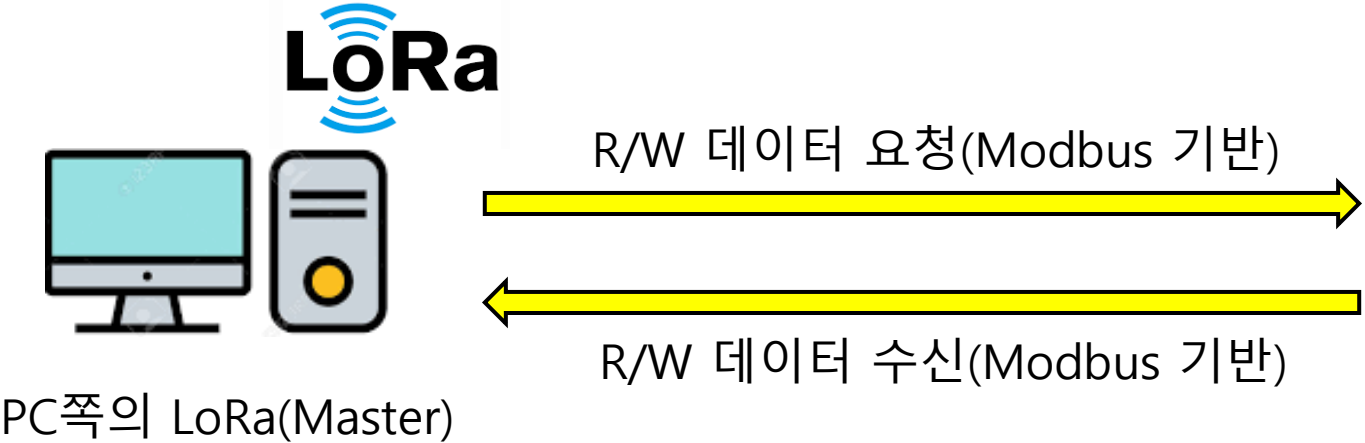


Modbus Protocol



Modbus_Application_Protocol_V1_1b3.pdf

PC 프로그램 - 로라(마스터) - 로라(슬레이브)



※ 주의사항

모터 드라이버 1대와 로라 모듈4대(마스터 1대, 슬레이브 3대)중

제가 전달 드릴 모터 드라이버는 Motor-Driver 1, Motor-Driver 2, Motor-Driver 3 이라고 적혀있습니다

이는 Slave 주소가 각각 0x01(Motor-Driver 1), 0x02(Motor-Driver 2), 0x03(Motor-Driver 3)이라는 뜻이며 이 각각의 모터 드라이버에는 [**ex**)Motor-Driver 1 - Slave 1]이라고 적혀있는 로라를 장착해야만 통신이 이루어지게 됩니다

나머지 제공드리는 슬레이브 3대는 Slave 1, Slave 2, Slave 3이라고 적혀있는데 이는 Slave 주소가 각각 Slave 1은 0x01, Slave 2은 0x02 , Slave 3은 0x03이라는 뜻입니다.

모드 버스 프로토콜을 이용해서 데이터를 요청하실때 슬레이브 주소가 모터 드라이버의 주소와 같은지 확인해 주길 바랍니다!

Ex) 모터 드라이버를 Motor-Driver1을 사용하고 로라 모듈을 Slave 1을 사용하실 경우 모드 버스 프로토콜로 데이터를 요청할때 슬레이브 주소를 0x01로 하셔야 합니다. 로라 모듈 Slave 2일 경우에는 Motor-Driver2에 부착 후, 슬레이브 주소를 0x02

Slave Address

01~FF

1~255

[0]

Slave의 주소



Modbus 프로그램 사용법

Modbus 프로그램 설치방법

1

ModbusPollSetup64Bit

2

Modbus Poll - 64 Bit

License Agreement

Please review the license terms before installing Modbus Poll - 64 Bit.

Press Page Down to see the rest of the agreement.

End User License Agreement.
You should carefully read the following terms and conditions before using Modbus Poll. Unless you have a different license agreement signed by Witte Software, your use of this software indicates your acceptance of this license agreement and warranty. If you do not accept these terms you must cease using this software immediately.

Copyright.
Modbus Poll ("The Software") is copyright 2002-2023 by Witte Software.

If you accept the terms of the agreement, select the first option below. You must accept the agreement to install Modbus Poll - 64 Bit. Click Next to continue.

- ☒ I accept the terms of the License Agreement
☐ I do not accept the terms of the License Agreement

Modbus Poll-10.5.2.2003.

Next >

Cancel

3

Modbus Poll - 64 Bit

Choose Install Location

Choose the folder in which to install Modbus Poll - 64 Bit.

Setup will install Modbus Poll - 64 Bit in the following folder. To install in a different folder, click Browse and select another folder. Click Next to continue.

Destination Folder

C:\Program Files\Modbus Tools\Modbus Poll

Browse...

Space required: 3.6 MB

Space available: 94.1 GB

Modbus Poll-10.5.2.2003.

< Back

Next >

Cancel

4

Modbus Poll - 64 Bit

Choose Components

Choose which features of Modbus Poll - 64 Bit you want to install.

Check the components you want to install and uncheck the components you don't want to install. Click Install to start the installation.

Select components to install:

- ☒ Program files (required)
☒ Desktop Shortcut
☒ Quick launch Shortcut

Space required: 3.6 MB

Modbus Poll-10.5.2.2003.

< Back

Install

Cancel

5

Modbus Poll - 64 Bit

Installation Complete

Setup was completed successfully.



Completed

[Show details](#)

Modbus Poll-10.5.2.2003,

< Back

Next >

Cancel

6

Modbus Poll - 64 Bit

**Completing Modbus Poll - 64 Bit Setup**

Modbus Poll - 64 Bit has been installed on your computer.

Click Finish to close Setup.

☒ Show Readme

< Back

Finish

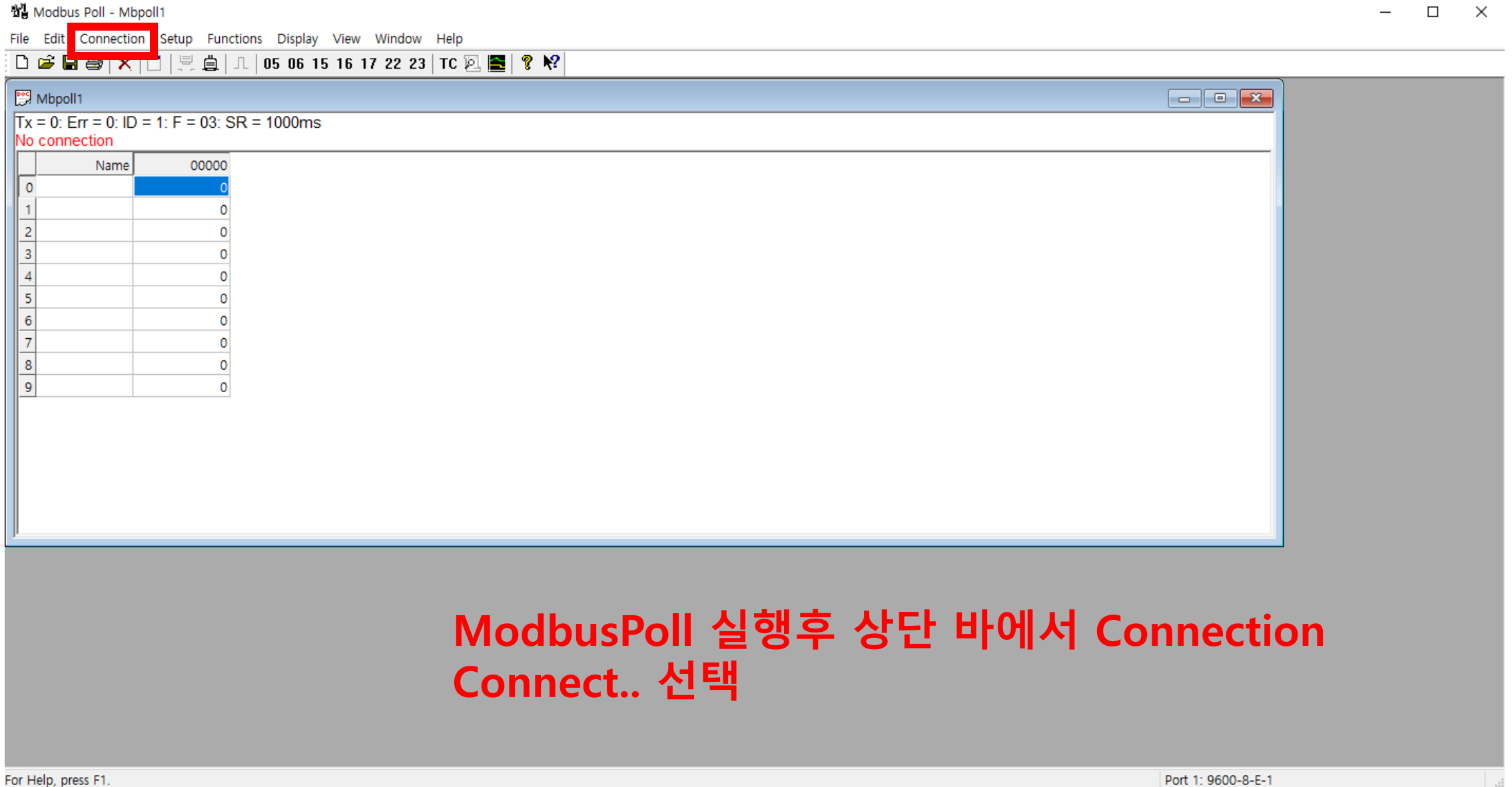
Cancel

7

실행

Modbus Poll

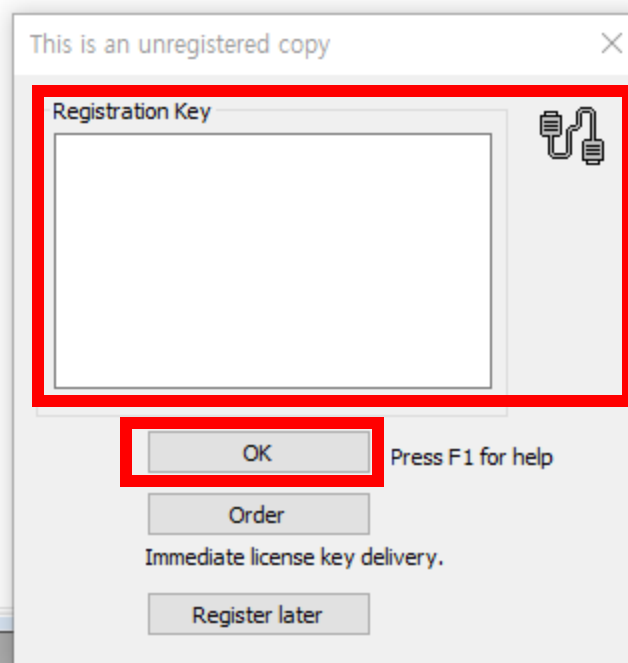
Modbus 프로그램 Registration Key 등록 방법



Tx = 0: Err = 0: ID = 1: F = 03: SR = 1000ms

No connection

| | Name | 00000 |
|---|------|-------|
| 0 | | 0 |
| 1 | | 0 |
| 2 | | 0 |
| 3 | | 0 |
| 4 | | 0 |
| 5 | | 0 |
| 6 | | 0 |
| 7 | | 0 |
| 8 | | 0 |
| 9 | | 0 |



Registration Key에

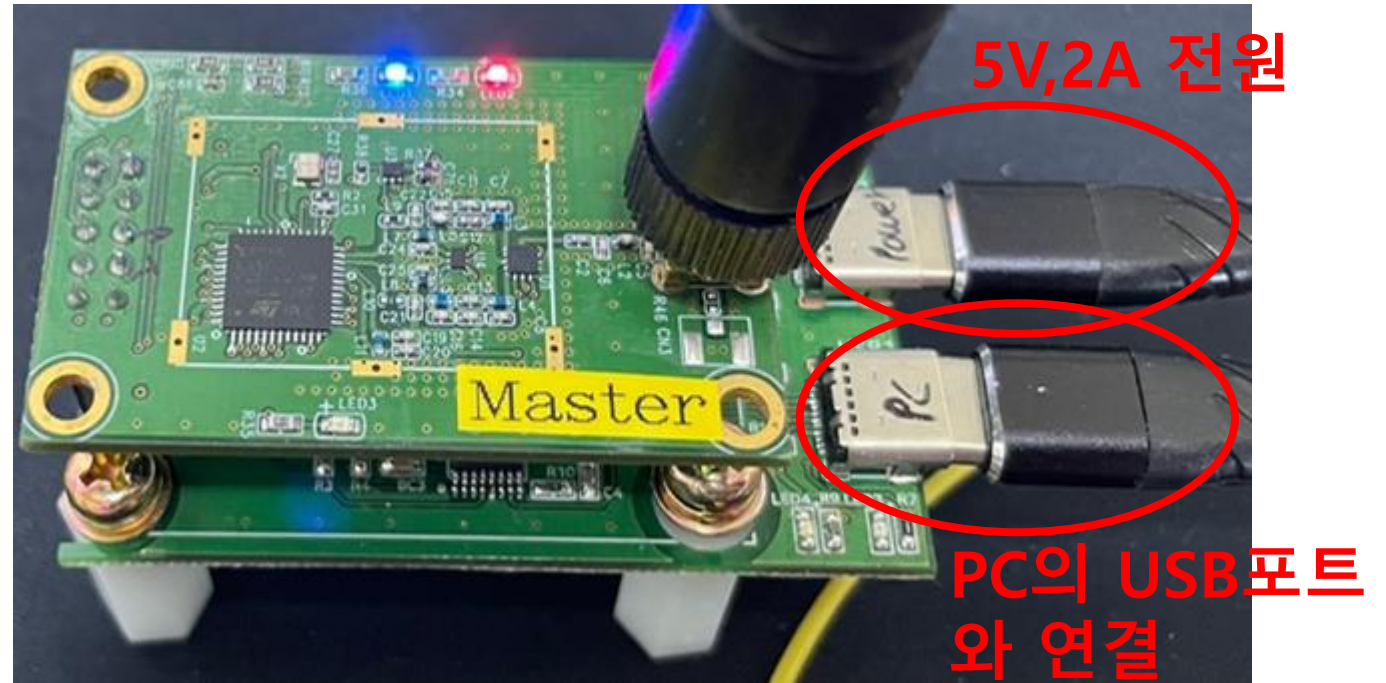
“5A5742575C5D7D5C415C57564311FD9AF3B4D3BB9C11391A17627B6C0202F8”을 입력후 OK 버튼을 누르면 등록이 완료된다.(ModBusPoll_key 이름이라는 메모장에 있다.)



Modbus 프로그램과 Master - LoRa 연결

Modbus 프로그램으로 통신하기전 작업

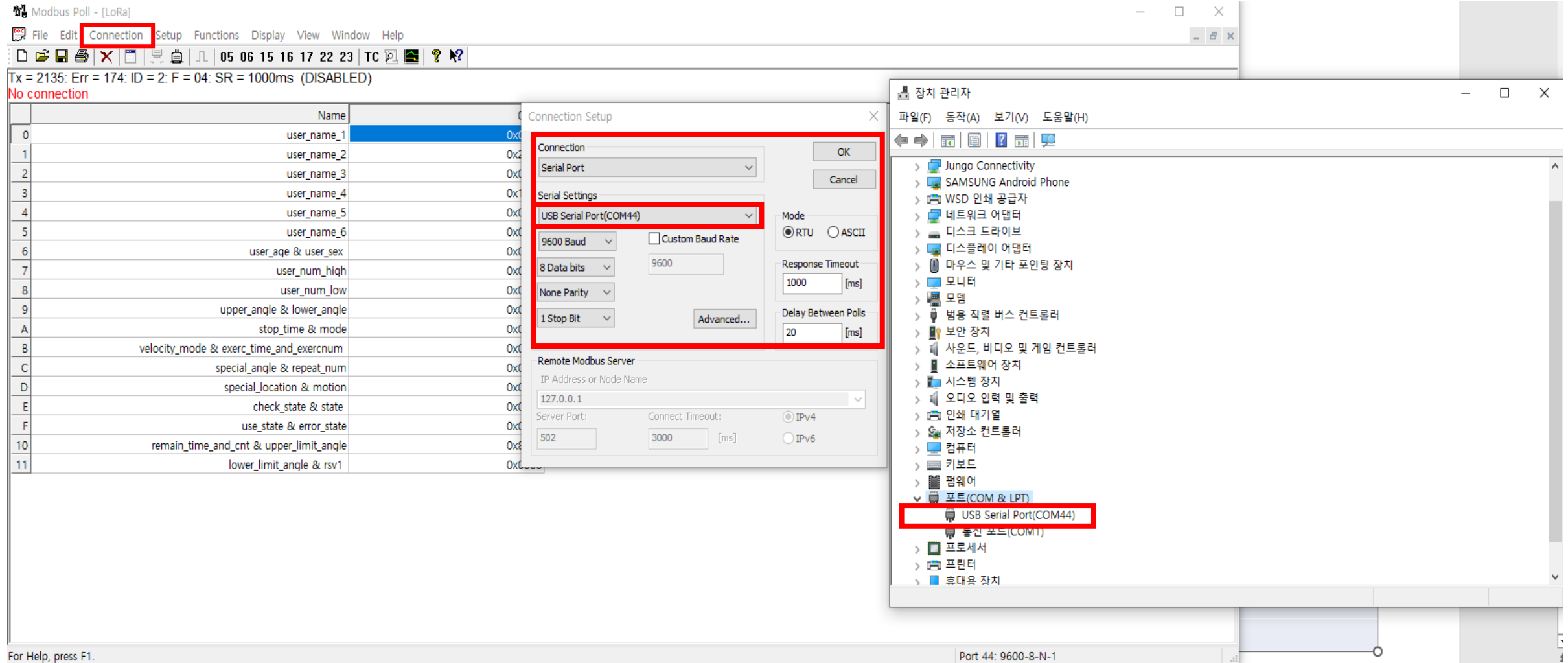
부분만 전원 연결해주면 됨



위와 같이 Mater LoRa의 USB2 포트에는 5V 2A 전원을 따로 공급해주고, USB1 포트는 PC의 USB부분과 연결해준다.

그후 모터드라이버에 Slave LoRa를 장착한후에 24V를 인가한다.

Modbus 프로그램에서 Master - LoRa Connection 하는 법



모터 드라이버와 Slave 로라 모듈을 연결하고 PC와 Master 로라 모듈을 연결한 상태에서 장치 관리자에서 포트(COM & LPT)를 열어 현재 USB 포트를 확인한후, Modbus 프로그램에서 상단바에 있는 Connection을 누르고 Connection Setup을 위와 같이 설정한후 OK 버튼을 누르면 연결이 완료된다.

※ 꼭 위와 같이 맞출것!!(9600 Baud, 8 Data bits, None Parity, 1 Stop Bit)

Modbus 프로그램을 이용하여 데이터 Read
요청하는 방법

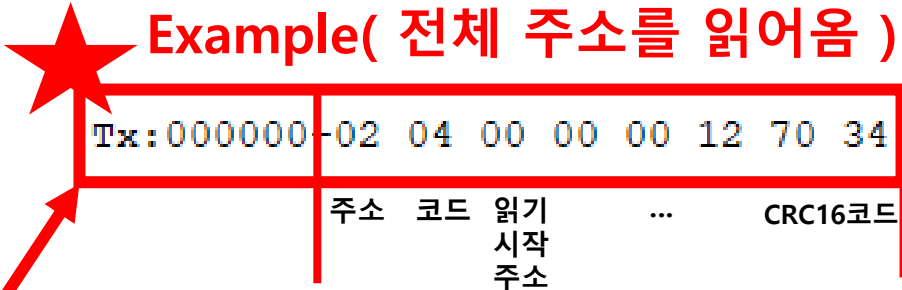
Modbus Protocol (Read)

| R/W | FC (Dec) | Function Name | Description |
|------|----------|----------------------|----------------------|
| Read | 04 | Read Input Registers | (16bit) Analog input |

❖ Read Input Registers (FC = 04)

▶ 이 명령은 아날로그 입력 레지스터를 읽어오기 위한 요청임.

[PC -> LoRa 모듈로 읽기 요청] : (Master(PC)) ---> (Slave(LoRa))

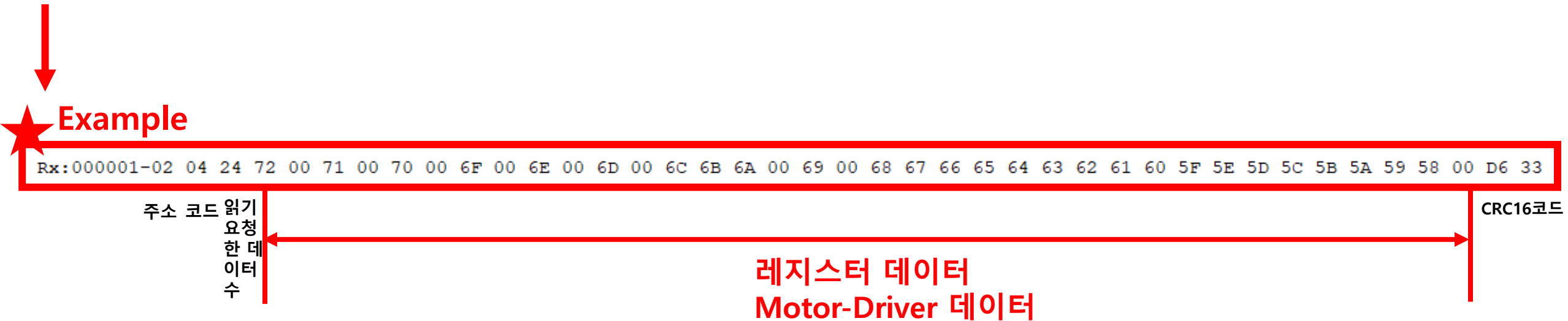


| Field Name | | Hex | Dec | Array Index | 설명 |
|--------------------------------------------------|---------------------------------------------|-------|---------|-------------|--------------------------------------------------------------------------------------|
| Slave Address | | 01~FF | 1~255 | [0] | Slave의 주소 |
| Function Code | | 04 | 4 | [1] | 함수 코드 |
| Data Address [High] (8 bits) | Data Address (16 bits) | 0000 | 0~65535 | [2] | 읽기 시작 주소 |
| Data Address [Low] (8 bits) | | FFFF | | [3] | |
| Quantity of input registers[High] (8 bits) | Quantity of input registers (16 bits) | 0001 | 1~125 | [4] | 읽어올 레지스터의 수(레지스터 하나당 16bits) (단위 : word = 2bytes) 한 번에 최대 125word 까지만 읽어올 수 있음 |
| Quantity of input registers[Low] (8 bits) | | 007D | | [5] | |
| CRC16 Code [Low] | | xx | - | [6] | CRC16 코드 |
| CRC16 Code [High] | | xx | - | [7] | |
| Total : 8 bytes | | | | | 총 전송 데이터(단위 : byte) |

Modbus Protocol (Read)

[LoRa -> PC로 읽기 응답] : (Slave(LoRa)) ---> (Master(PC))

| Field Name | Hex | Dec | Array Index | 설명 |
|--------------------------------------------|-------|-------|-------------|--------------------------------------------------------------------------|
| Slave Address | 01~FF | 1~255 | [0] | Slave의 주소 |
| Function Code | 04 | 4 | [1] | 함수 코드 |
| The number of data bytes | 02~FA | 2~250 | [2] | Master에서 읽기 요청한 데이터 수(단위:byte) (최대 125 register × 2 bytes = 250bytes) |
| Input Register Data (N byte) [2 ≤ N ≤ 250] | - | - | [3 ~ 3+N-1] | 레지스터 데이터 |
| CRC16 Code [Low] | xx | - | [4+N-1] | CRC16 코드 |
| CRC16 Code [High] | xx | - | [5+N-1] | |
| Total : 5+N bytes | | | | 총 전송 데이터(단위 : byte) |



Modbus 프로그램을 이용 (LoRa모듈로 Read 요청)

실행

슬레이브 주소 2 = 0x02로, LoRa Slave 2 + Motor-Driver 2 연결

Setup -> Read/Write Definition 선택

Modbus Poll - [LoRa]

File Edit Connection **Setup** Functions Display View Window Help

Tx = 0: Err = 0: ID = 2: F = 04: SR = 1000ms (DISABLED)
No connection

| | Name | Address |
|----|-----------------------------------------|---------|
| | | 0000 |
| 0 | user_name_1 | 0x7200 |
| 1 | user_name_2 | 0x7100 |
| 2 | user_name_3 | 0x7000 |
| 3 | user_name_4 | 0x6F00 |
| 4 | user_name_5 | 0x6E00 |
| 5 | user_name_6 | 0x6D00 |
| 6 | user_age & user_sex | 0x6C68 |
| 7 | user_num_high | 0x6A00 |
| 8 | user_num_low | 0x6900 |
| 9 | upper_angle & lower_angle | 0x6867 |
| A | stop_time & mode | 0x6665 |
| B | velocity_mode & exerc_time_and_exercnum | 0x6463 |
| C | special_angle & repeat_num | 0x6261 |
| D | special_location & motion | 0x605F |
| E | check_state & state | 0x5E5D |
| F | use_state & error_state | 0x5C5B |
| 10 | remain_time_and_cnt & upper_limit_angle | 0x5A59 |
| 11 | lower_limit_angle & rsv1 | 0x5800 |

Read/Write Definition

Slave ID: 2

Function: 04 Read Input Registers (3x)

Address mode: ☐ Dec ☒ Hex

Address: 0 PLC address = 30001

Quantity: 18

Scan Rate: 1000 [ms]

Disable: ☒ Read/Write Disabled ☐ Disable on error

View: Rows ☐ 16 ☐ 32 ☐ 64 ☐ 128 ☒ Fit to Quantity

☐ Hide Name Columns ☐ PLC Addresses (Base 1)

☐ Address in Cell ☐ Enron/Daniel Mode

Request: RTU 02 04 00 00 00 12 70 34

ASCII 3A 30 32 30 34 30 30 30 30 31 32 45 38 0D 0A

Communication Traffic

Exit Stop Clear Save Copy Log ☐ Stop on Error ☐ Time stamp

For Help, press F1. Port 39: 9600-8-E-1

3

※ LoRa 모듈과 Motor-Driver의 국번이 Slave ID와 일치 해야 통신 가능!!!

Modbus Poll - [LoRa]

File Edit Connection Setup Functions Display View Window Help

Tx = 0: Err = 0: ID = 2: F = 04: SR = 1000ms (DISABLED)

| | Name | Address |
|----|-----------------------------------------|---------|
| 0 | user_name_1 | 0x7200 |
| 1 | user_name_2 | 0x7100 |
| 2 | user_name_3 | 0x7000 |
| 3 | user_name_4 | 0x6F00 |
| 4 | user_name_5 | 0x6E00 |
| 5 | user_name_6 | 0x6D00 |
| 6 | user_age & user_sex | 0x6C68 |
| 7 | user_num_high | 0x6A00 |
| 8 | user_num_low | 0x6900 |
| 9 | upper_angle & lower_angle | 0x6867 |
| A | stop_time & mode | 0x6665 |
| B | velocity_mode & exerc_time_and_exercnum | 0x6463 |
| C | special_angle & repeat_num | 0x6261 |
| D | special_location & motion | 0x6061 |
| E | check_state & state | 0x5E5D |
| F | use_state & error_state | 0x5C5B |
| 10 | remain_time_and_cnt & upper_limit_angle | 0x5A59 |
| 11 | lower_limit_angle & rsv1 | 0x5800 |

Read/Write Definition

Slave ID: 2

Function: 04 Read Input Registers (3x)

Address mode: ☐ Dec ☒ Hex

Address: 0 PLC address = 30001

Quantity: 18

Scan Rate: 1000 [ms]

Disable: ☒ Read/Write Disabled ☐ Disable on error

View: Rows ☐ 16 ☐ 32 ☐ 64 ☐ 128 ☒ Fit to Quantity

☐ Hide Name Columns ☐ PLC Addresses (Base 1)

☐ Address in Cell ☐ Enron/Daniel Mode

Request: RTU ASCII

Communication Traffic

Exit Stop Clear Save Copy Log ☐ Stop on Error ☐ Time stamp

Port 39: 9600-8-E-1

Lora 모듈로 Scan Rate ms마다 Read 요청 (Scan Rate가 1000이면 1초마다 주기적으로 요청)

Lora 모듈로 Read 요청을 한번 수행

Communication Traffic

✕

Exit

Stop

Clear

Save

Copy

Log

☐ Stop on Error ☐ Time stamp

```

Tx:000363-02 04 00 00 00 12 70 34
Rx:000364-02 04 24 AF BE 45 76 70 89 AB CD EF DA 03 25 78 65 47 56 23 45 74 56 23 54 79 86 23 45 24 53 00 87 56 78 40 56 69 78 08 16
Tx:000365-02 04 00 00 00 12 70 34
Rx:000366-02 04 24 AF BE 45 76 70 89 AB CD EF DA 03 25 78 65 47 56 23 45 74 56 23 54 79 86 23 45 24 53 00 87 56 78 40 56 69 78 08 16
Tx:000367-02 04 00 00 00 12 70 34
Rx:000368-02 04 24 AF BE 45 76 70 89 AB CD EF DA 03 25 78 65 47 56 23 45 74 56 23 54 79 86 23 45 24 53 00 87 56 78 40 56 69 78 08 16
Tx:000369-02 04 00 00 00 12 70 34
Rx:000370-02 04 24 AF BE 45 76 70 89 AB CD EF DA 03 25 78 65 47 56 23 45 74 56 23 54 79 86 23 45 24 53 00 87 56 78 40 56 69 78 08 16
Tx:000371-02 04 00 00 00 12 70 34
Rx:000372-02 04 24 AF BE 45 76 70 89 AB CD EF DA 03 25 78 65 47 56 23 45 74 56 23 54 79 86 23 45 24 53 00 87 56 78 40 56 69 78 08 16
Tx:000373-02 04 00 00 00 12 70 34
Rx:000374-02 04 24 AF BE 45 76 70 89 AB CD EF DA 03 25 78 65 47 56 23 45 74 56 23 54 79 86 23 45 24 53 00 87 56 78 40 56 69 78 08 16
Tx:000375-02 04 00 00 00 12 70 34
Rx:000376-02 04 24 AF BE 45 76 70 89 AB CD EF DA 03 25 78 65 47 56 23 45 74 56 23 54 79 86 23 45 24 53 00 87 56 78 40 56 69 78 08 16

```

Modbus Poll - [LoRa]

File Edit Connection Setup Functions Display View Window Help

05 06 15 16 17 22 23 TC

Tx = 12: Err = 0: ID = 2: F = 04: SR = 1000ms

| | Name | |
|----|-----------------------------------------|--------|
| | | 0000 |
| 0 | user_name_1 | 0xAFBE |
| 1 | user_name_2 | 0x4576 |
| 2 | user_name_3 | 0x7089 |
| 3 | user_name_4 | 0xABCD |
| 4 | user_name_5 | 0xEFDA |
| 5 | user_name_6 | 0x0325 |
| 6 | user_age & user_sex | 0x7865 |
| 7 | user_num_high | 0x4756 |
| 8 | user_num_low | 0x2345 |
| 9 | upper_angle & lower_angle | 0x7456 |
| A | stop_time & mode | 0x2354 |
| B | velocity_mode & exerc_time_and_exercnum | 0x7986 |
| C | special_angle & repeat_num | 0x2345 |
| D | special_location & motion | 0x2453 |
| E | check_state & state | 0x0087 |
| F | use_state & error_state | 0x5678 |
| 10 | remain_time_and_cnt & upper_limit_angle | 0x4056 |
| 11 | lower_limit_angle & rsv1 | 0x6978 |

For Help, press F1.

Read/Write Definition

✕

Slave ID: 2

OK

Function: 04 Read Input Registers (3x)

Cancel

Address mode

☐ Dec☒ Hex

Address: 0 PLC address = 30001

Quantity: 18

Scan Rate: 1000 [ms]

Apply

Disable

☐ Read/Write Disabled☐ Disable on error

Read/Write Once

View

Rows

☐ 16☐ 32☐ 64☐ 128☒ Fit to Quantity☐ Hide Name Columns☐ PLC Addresses (Base 1)☐ Address in Cell☐ Enron/Daniel Mode

Request

RTU

02 04 00 00 00 12 70 34

ASCII

3A 30 32 30 34 30 30 30 30 31 32 45 38 0D 0A

39: 9600-8-N-1

Modbus 프로그램을 이용하여 데이터 Write
요청하는 방법

Modbus Protocol (Write)

| R/W | FC (Dec) | Function Name | Description |
|-------|----------|---------------------------|--------------------------------|
| Write | 16 | Preset Multiple Registers | (16bit) Multiple Analog output |

❖ Preset Multiple Registers (FC = 16)

▶ 이 명령은 다중 레지스터에 데이터를 쓰기 위한 요청임.

[PC -> LoRa 모듈로 쓰기 요청] : (Master(PC)) ---> (Slave(LoRa))

| Field Name | | Hex | Dec | Array Index | 설명 |
|-------------------------------------------------------|-----------------------------------|-------|---------|-------------|-----------------------------------------------------------------------------------------------|
| Slave Address | | 01~FF | 1~255 | [0] | Slave의 주소 |
| Function Code | | 10 | 16 | [1] | 함수 코드 |
| Data Address [High] (8 bits) | Data Address (16 bits) | 0000 | 0~65535 | [2] | 쓰기 시작 주소 |
| Data Address [Low] (8 bits) | | FFFF | | [3] | |
| Quantity of registers [High] (8 bits) | Quantity of registers (16bits) | 0001 | 1~123 | [4] | 쓰기 요청할 레지스터의 수(레지스터 하나당 16bits) (단위 : word = 2bytes) 한 번에 최대 123word 까지만 쓸 수 있음 |
| Quantity of registers [Low] (8 bits) | | 007B | | [5] | |
| The number of data bytes | | 02~F6 | 2~246 | [6] | 쓰기 요청할 데이터 수(단위:byte) (최대 123 register × 2 bytes = 246bytes) |
| The registers data to write (N byte) [2 ≤ N ≤ 246] | | - | - | [7 ~ 7+N-1] | 쓰기 요청할 레지스터 데이터 |
| CRC16 Code [Low] | | xx | - | [8+N-1] | CRC16 코드 |
| CRC16 Code [High] | | xx | - | [9+N-1] | |
| Total : 9+N bytes | | | | | 총 전송 데이터(단위 : byte) |

Example

Tx:003601-02 10 00 00 00 12 24 12 43 45 76 70 89 AB CD EF DA 03 25 78 65 47 56 23 45 74 56 23 54 79 86 23 45 24 53 56 87 56 78 47 56 69 78 75 37

Modbus Protocol (Write)

[LoRa -> PC로 쓰기 응답] : (Slave(LoRa)) ---> (Master(PC))

| Field Name | | Hex | Dec | Array Index | 설명 |
|------------------------------------------|------------------------------------|-------------------|---------|-------------|-------------------------------------------------|
| Slave Address | | 01~FF | 1~255 | [0] | Slave의 주소 |
| Function Code | | 10 | 16 | [1] | 함수 코드 |
| Data Address [High] (8 bits) | Data Address (16 bits) | 0000 ~ FFFF | 0~65535 | [2] | 쓰기 시작 주소 |
| Data Address [Low] (8 bits) | | [3] | | | |
| Quantity of registers [High] (8 bits) | Quantity of registers (16 bits) | 0001 ~ 007B | 1~123 | [4] | Master에서 요청한 쓰기 레지스터의 수 (단위 : word = 2bytes) |
| Quantity of registers [Low] (8 bits) | | [5] | | | |
| CRC16 Code [Low] | | xx | - | [6] | CRC16 코드 |
| CRC16 Code [High] | | xx | - | [7] | |
| Total : 8 bytes | | | | | 총 전송 데이터(단위 : byte) |



★ Example

Rx:003602-02 10 00 00 00 12 40 37

Modbus 프로그램을 이용 (LoRa모듈로 Write 요청)

1

Communication Traffic

Exit Stop Clear Save Copy Log ☐ Stop on Error ☐ Time stamp

Rx:000507-02 04 24 AF BE 45 76 70 89 AB CD EF DA 03 25 78 65 47 56 23 45 74 56 23 54 79 86 23 45 24 53 00 87 56 78 40 56 69 78 08 16
Tx:000508-02 04 00 00 00 12 70 34
Rx:000509-02 04 24 AF BE 45 76 70 89 AB CD EF DA 03 25 78 65 47 56 23 45 74 56 23 54 79 86 23 45 24 53 00 87 56 78 40 56 69 78 08 16
Tx:000510-02 04 00 00 00 12 70 34
Rx:000511-02 04 24 AF BE 45 76 70 89 AB CD EF DA 03 25 78 65 47 56 23 45 74 56 23 54 79 86 23 45 24 53 00 87 56 78 40 56 69 78 08 16
Tx:000512-02 04 00 00 00 12 70 34
Rx:000513-02 04 24 AF BE 45 76 70 89 AB CD EF DA 03 25 78 65 47 56 23 45 74 56 23 54 79 86 23 45 24 53 00 87 56 78 40 56 69 78 08 16
Tx:000514-02 04 00 00 00 12 70 34
Rx:000515-02 04 24 AF BE 45 76 70 89 AB CD EF DA 03 25 78 65 47 56 23 45 74 56 23 54 79 86 23 45 24 53 00 87 56 78 40 56 69 78 08 16
Tx:000516-02 04 00 00 00 12 70 34
Rx:000517-02 04 24 AF BE 45 76 70 89 AB CD EF DA 03 25 78 65 47 56 23 45 74 56 23 54 79 86 23 45 24 53 00 87 56 78 40 56 69 78 08 16
Tx:000518-02 04 00 00 00 12 70 34
Rx:000519-02 04 24 AF BE 45 76 70 89 AB CD EF DA 03 25 78 65 47 56 23 45 74 56 23 54 79 86 23 45 24 53 00 87 56 78 40 56 69 78 08 16
Tx:000520-02 04 00 00 00 12 70 34

Modbus Poll - [LoRa]

File Edit Connection Setup Functions Display View Window Help

05 06 15 16 17 22 23 TC ? ?

Tx = 84: Err = 1: ID = 2: F = 04: SR = 1000ms

| | Name | 0000 |
|----|-----------------------------------------|--------|
| 0 | user_name_1 | 0xAFBE |
| 1 | user_name_2 | 0x4576 |
| 2 | user_name_3 | 0x7089 |
| 3 | user_name_4 | 0xABCD |
| 4 | user_name_5 | 0xEFDA |
| 5 | user_name_6 | 0x0325 |
| 6 | user_age & user_sex | 0x7865 |
| 7 | user_num_high | 0x4756 |
| 8 | user_num_low | 0x2345 |
| 9 | upper_angle & lower_angle | 0x7456 |
| A | stop_time & mode | 0x2354 |
| B | velocity_mode & exerc_time_and_exercnum | 0x7986 |
| C | special_angle & repeat_num | 0x2345 |
| D | special_location & motion | 0x2453 |
| E | check_state & state | 0x0087 |
| F | use_state & error_state | 0x5678 |
| 10 | remain_time_and_cnt & upper_limit_angle | 0x4056 |
| 11 | lower_limit_angle & rsv1 | 0x6978 |

16 (0x10) Write Multiple Registers

Slave ID: 2 Address: 0 Hex Quantity: 18 Type: Signed

Update Value List

0 = 0
1 = 0
2 = 0
3 = 0
4 = 0
5 = 0
6 = 0
7 = 0
8 = 0
9 = 0
A = 0
B = 0
C = 0
D = 0
E = 0
F = 0
10 = 0
11 = 0

Send Cancel Edit Open Save

Write multiple registers

2

아래와 같은 창에서 주소(0 ~ D)를 더블클릭하여 값을 입력한다.

Communication Trainer

Exit Stop Clear Save Copy Log ☐ Stop on Error ☐ Time stamp

```
Rx:001203-02 04 24 AF BE 45 76 70 89 AB CD EF DA 03 25 78 65 47 56 23 45 74 56 23 54 79 86 23 45 24 53 00 87 56 78 40 56 69 78 08 16
Tx:001204-02 04 00 00 00 12 70 34
Rx:001205-02 04 24 AF BE 45 76 70 89 AB CD EF DA 03 25 78 65 47 56 23 45 74 56 23 54 79 86 23 45 24 53 00 87 56 78 40 56 69 78 08 16
Tx:001206-02 04 00 00 00 12 70 34
Rx:001207-02 04 24 AF BE 45 76 70 89 AB CD EF DA 03 25 78 65 47 56 23 45 74 56 23 54 79 86 23 45 24 53 00 87 56 78 40 56 69 78 08 16
Tx:001208-02 04 00 00 00 12 70 34
Rx:001209-02 04 24 AF BE 45 76 70 89 AB CD EF DA 03 25 78 65 47 56 23 45 74 56 23 54 79 86 23 45 24 53 00 87 56 78 40 56 69 78 08 16
Tx:001210-02 04 00 00 00 12 70 34
Rx:001211-02 04 24 AF BE 45 76 70 89 AB CD EF DA 03 25 78 65 47 56 23 45 74 56 23 54 79 86 23 45 24 53 00 87 56 78 40 56 69 78 08 16
Tx:001212-02 04 00 00 00 12 70 34
Rx:001213-02 04 24 AF BE 45 76 70 89 AB CD EF DA 03 25 78 65 47 56 23 45 74 56 23 54 79 86 23 45 24 53 00 87 56 78 40 56 69 78 08 16
Tx:001214-02 04 00 00 00 12 70 34
Rx:001215-02 04 24 AF BE 45 76 70 89 AB CD EF DA 03 25 78 65 47 56 23 45 74 56 23 54 79 86 23 45 24 53 00 87 56 78 40 56 69 78 08 16
Tx:001216-02 04 00 00 00 12 70 34
```

Modbus Poll - [LoRa]

File Edit Connection Setup Functions Display View Window Help

05 06 15 16 17 22 23 TC

Tx = 442: Err = 21: ID = 2: F = 04: SR = 1000ms

| | Name | Value |
|----|-----------------------------------------|--------|
| 0 | user_name_1 | 0xAFBE |
| 1 | user_name_2 | 0x4576 |
| 2 | user_name_3 | 0x7089 |
| 3 | user_name_4 | 0xABCD |
| 4 | user_name_5 | 0xEFDA |
| 5 | user_name_6 | 0x0325 |
| 6 | user_age & user_sex | 0x7865 |
| 7 | user_num_high | 0x4756 |
| 8 | user_num_low | 0x2345 |
| 9 | upper_angle & lower_angle | 0x7456 |
| A | stop_time & mode | 0x2354 |
| B | velocity_mode & exerc_time_and_exercnum | 0x7986 |
| C | special_angle & repeat_num | 0x2345 |
| D | special_location & motion | 0x2453 |
| E | check_state & state | 0x0087 |
| F | use_state & error_state | 0x5678 |
| 10 | remain_time_and_cnt & upper_limit_angle | 0x4056 |
| 11 | lower_limit_angle & rsv1 | 0x6978 |

16 (0x10) Write Multiple Registers

Slave ID: 2 Address: 0 Hex Quantity: 14 Type: Hex

0 = 0x0000
1 = 0x0000
2 = 0x0000
3 = 0x0000
4 = 0x0000

Enter hex value

Value (HEX): ABCD

OK Cancel

D = 0x0000

Send Cancel Edit Open Save

For Help, press F1. Port 39: 9600-8-N-1

3

Send 버튼을 눌러 LoRa 모듈로 Write 요청을 보낸다.

Communication Traffic

Exit Stop Clear Save Copy Log ☐ Stop on Error ☐ Time stamp

Tx:001310-02 04 00 00 00 12 70 34
Rx:001311-02 04 24 AF BE 45 76 70 89 AB CD EF DA 03 25 78 65 47 56 23 45 74 56 23 54 79 86 23 45 24 53 00 87 56 78 40 56 69 78 08 16
Tx:001312-02 04 00 00 00 12 70 34
Rx:001313-02 04 24 AF BE 45 76 70 89 AB CD EF DA 03 25 78 65 47 56 23 45 74 56 23 54 79 86 23 45 24 53 00 87 56 78 40 56 69 78 08 16
Tx:001314-02 04 00 00 00 12 70 34
Rx:001315-02 04 24 AF BE 45 76 70 89 AB CD EF DA 03 25 78 65 47 56 23 45 74 56 23 54 79 86 23 45 24 53 00 87 56 78 40 56 69 78 08 16
Tx:001316-02 04 00 00 00 12 70 34
Rx:001317-02 04 24 AF BE 45 76 70 89 AB CD EF DA 03 25 78 65 47 56 23 45 74 56 23 54 79 86 23 45 24 53 00 87 56 78 40 56 69 78 08 16
Tx:001318-02 04 00 00 00 12 70 34
Rx:001319-02 04 24 AF BE 45 76 70 89 AB CD EF DA 03 25 78 65 47 56 23 45 74 56 23 54 79 86 23 45 24 53 00 87 56 78 40 56 69 78 08 16
Tx:001320-02 04 00 00 00 12 70 34
Rx:001321-02 04 24 AF BE 45 76 70 89 AB CD EF DA 03 25 78 65 47 56 23 45 74 56 23 54 79 86 23 45 24 53 00 87 56 78 40 56 69 78 08 16
Tx:001322-02 04 00 00 00 12 70 34
Tx:001323-02 04 00 00 00 12 70 34

Modbus Poll - [LoRa]

File Edit Connection Setup Functions Display View Window Help

05 06 15 16 17 22 23 TC ? ?

Tx = 497: Err = 24: ID = 2: F = 04: SR = 1000ms
Timeout error

| | Name | 0000 |
|----|-----------------------------------------|--------|
| 0 | user_name_1 | 0xAFBE |
| 1 | user_name_2 | 0x4576 |
| 2 | user_name_3 | 0x7089 |
| 3 | user_name_4 | 0xABCD |
| 4 | user_name_5 | 0xEFDA |
| 5 | user_name_6 | 0x0325 |
| 6 | user_age & user_sex | 0x7865 |
| 7 | user_num_high | 0x4756 |
| 8 | user_num_low | 0x2345 |
| 9 | upper_angle & lower_angle | 0x7456 |
| A | stop_time & mode | 0x2354 |
| B | velocity_mode & exerc_time_and_exercnum | 0x7986 |
| C | special_angle & repeat_num | 0x2345 |
| D | special_location & motion | 0x2453 |
| E | check_state & state | 0x0087 |
| F | use_state & error_state | 0x5678 |
| 10 | remain_time_and_cnt & upper_limit_angle | 0x4056 |
| 11 | lower_limit_angle & rsv1 | 0x6978 |

16 (0x10) Write Multiple Registers

Slave ID: 2
Address: 0 Hex
Quantity: 14
Type: Hex
Update Value List

0 = 0xABCD
1 = 0x0000
2 = 0x0000
3 = 0x0000
4 = 0x0000
5 = 0x0000
6 = 0x0000
7 = 0x0000
8 = 0x0000
9 = 0x0000
A = 0x0000
B = 0x0000
C = 0x0000
D = 0x0000

Send
Cancel
Edit
Open
Save

For Help, press F1.

Port 39: 9600-8-N-1

Send버튼을 눌러 LoRa 모듈로 Write 요청을 보내고 난후 다시 Read 요청을 보내면 값이 Write 요청한 값으로 바뀐것을 확인할 수 있다.

The screenshot displays the Modbus Poll application window. At the top, there's a menu bar (File, Edit, Connection, Setup, Functions, Display, View, Window, Help) and a toolbar. Below the toolbar, a status bar shows 'Tx = 606: Err = 53: ID = 2: F = 04: SR = 1000ms'. The main area contains a table with 12 rows and 3 columns: 'Name', 'Value', and 'Hex'. The 'Name' column lists various variables like 'user_name', 'user_age & user_sex', 'user_num_high', etc. The 'Value' column shows hexadecimal values, and the 'Hex' column shows their decimal equivalents. A red rectangle highlights the 'Value' and 'Hex' columns. Overlaid on the table is a '16 (0x10) Write Multiple Registers' dialog box. This dialog has fields for 'Slave ID' (2), 'Address' (0), 'Quantity' (14), and 'Type' (Hex). It also has a 'Send' button and a 'Response ok' confirmation dialog box. The 'Response ok' dialog has a yellow warning icon and a '확인' (OK) button. The background dialog also has a 'Value List' button and a list of values from 0 to D.

| | Name | Value | Hex |
|----|-----------------------------------------|--------|--------|
| 0 | user_name | 0xABCD | 0x0000 |
| 1 | user_name | 0x0000 | 0x0000 |
| 2 | user_name | 0x0000 | 0x0000 |
| 3 | user_name | 0x0000 | 0x0000 |
| 4 | user_name | 0x0000 | 0x0000 |
| 5 | user_name | 0x0000 | 0x0000 |
| 6 | user_age & user_sex | 0x0000 | 0x0000 |
| 7 | user_num_high | 0x0000 | 0x0000 |
| 8 | user_num_low | 0x0000 | 0x0000 |
| 9 | upper_angle & lower_angle | 0x0000 | 0x0000 |
| A | stop_time & mode | 0x0000 | 0x0000 |
| B | velocity_mode & exerc_time_and_exercnum | 0x0000 | 0x0000 |
| C | special_angle & repeat_num | 0x0000 | 0x0000 |
| D | special_location & motion | 0x0000 | 0x0000 |
| E | check_state & state | 0x0087 | 0x0000 |
| F | use_state & error_state | 0x5678 | 0x0000 |
| 10 | remain_time_and_cnt & upper_limit_angle | 0x4056 | 0x0000 |
| 11 | lower_limit_angle & rsv1 | 0x6978 | 0x0000 |

16 (0x10) Write Multiple Registers

Slave ID: 2
Address: 0 Hex
Quantity: 14
Type: Hex
Update Value List

0 = 0xABCD
1 = 0x0000
2 = 0x0000
3 = 0x0000
4 = 0x0000
5 = 0x0000
6 = 0x0000
7 = 0x0000
8 = 0x0000
9 = 0x0000
A = 0x0000
B = 0x0000
C = 0x0000
D = 0x0000

Send
Cancel
Edit
Open
Save

Response ok
확인

For Help, press F1.

Port 39: 9600-8-N-1

Port 39: 9600-8-N-1

Modbus 프로그램을 이용하여 데이터 Read
결과(예시)

Modbus Protocol (Read 결과)

LoRa Module로 부터 받아온 데이터를 PC 화면에 출력
아래와 같은 데이터를 통해 환자 운동 정보를 얻을 수 있다.

Tx:000000

CRC16코드

02 04 00 00 00 12 70 34

Rx:000001

주소 코드 ...

02 04 24 72 00 71 00 70 00 6F 00 6E 00 6D 00 6C 6B 6A 00 69 00 68 67 66 65 64 63 62 61 60 5F 5E 5D 5C 5B 5A 59 58 00 D6 33

CRC16코드

| | Name | 0000 |
|----|-----------------------------------------|--------|
| 0 | user_name_1 | 0x7200 |
| 1 | user_name_2 | 0x7100 |
| 2 | user_name_3 | 0x7000 |
| 3 | user_name_4 | 0x6F00 |
| 4 | user_name_5 | 0x6E00 |
| 5 | user_name_6 | 0x6D00 |
| 6 | user_age & user_sex | 0x6C6B |
| 7 | user_num_high | 0x6A00 |
| 8 | user_num_low | 0x6900 |
| 9 | upper_angle & lower_angle | 0x6867 |
| A | stop_time & mode | 0x6665 |
| B | velocity_mode & exerc_time_and_exercnum | 0x6463 |
| C | special_angle & repeat_num | 0x6261 |
| D | special_location & motion | 0x605F |
| E | check_state & state | 0x5E5D |
| F | use_state & error_state | 0x5C5B |
| 10 | remain_time_and_cnt & upper_limit_angle | 0x5A59 |
| 11 | lower_limit_angle & rsv1 | 0x5800 |

| Address | Read/Write | Type | Name | Description | Unit | Default |
|---------|------------|-------|-----------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------|---------|
| 0 | R/W | 16bit | user_name_1 | 0xD64D (홍) | None | 0xD64D |
| 1 | R/W | 16bit | user_name_2 | 0xAE38 (길) | None | 0xAE38 |
| 2 | R/W | 16bit | user_name_3 | 0xB3D9 (동) | None | 0xB3D9 |
| 3 | R/W | 16bit | user_name_4 | 0x0000 | None | 0 |
| 4 | R/W | 16bit | user_name_5 | 0x0000 | None | 0 |
| 5 | R/W | 16bit | user_name_6 | 0x0000 | None | 0 |
| 6 | R/W | 16bit | user_age & user_sex | user_age(상위 바이트) : 환자 나이 1 ~ 120세까지 가능 , user_sex(하위 바이트) : 환자 성별 (남 : 0, 여 : 1) | None | 0 |
| 7 | R/W | 16bit | user_num_high | user_num_high(상위 바이트) : 환자 등록번호 상위 2바이트 | None | 0 |
| 8 | R/W | 16bit | user_num_low | user_num_low(하위 바이트) : 환자 등록번호 하위 2바이트 | None | 0 |
| 9 | R/W | 16bit | upper_angle & lower_angle | upper_angle(상위 바이트) : 상한각 (-50 ~ 180) (0 ~ 230 통신후 실제 변수에 담기는 값은 -50 처리후 넣을것) lower_angle(하위 바이트) : 하한각 (-70 ~ 160) (0 ~ 230 통신후 실제 변수에 담기는 값은 -70 처리후 넣을것) | None | 0 |
| 10 | R/W | 16bit | stop_time & mode | stop_time(상위 바이트) : 상한 정지 시간(상위 4bit), 하한 정지 시간(하위 4bit) mode(하위 바이트) : 0x00(일반 등속), 0x01(적응 운동), 0x02 (집중 운동), 0x03(일반 가속) | None | 0 |
| 11 | R/W | 16bit | velocity_mode & exerc_time_and_exercnum | velocity_mode(상위 바이트) : 0x01 : 1단계 ~ 0x09 : 9단계 exerc_time_and_exercnum(하위 바이트) : 운동시간 또는 운동횟수 선택이므로 MSB가 0이면 운동시간, 1이면 운동횟수 | None | 0 |
| 12 | R/W | 16bit | special_angle & repeat_num | special_angle(상위 바이트) : 집중/적응 운동 각도 최소 5°, 최대 15° repeat_num(하위 바이트) : 집중/적응 운동 반복 횟수 최소 3회, 최대 10회 | None | 0 |
| 13 | R/W | 16bit | special_location & rsv1 | special_location(상위 바이트) : 상한각 : 0x00, 하한각 : 0x01, 상하한각 : 0x02 rsv1(하위 바이트) : Dummy Data(실제로는 필요없는 값이므로 항상 0이다. 무시할것) | None | 0 |
| 14 | R | 16bit | check_state & state | check_state(상위 바이트) : 에러체크 완료 : 0x00, 전류 에러체크 진행중 : 0x02, 엔코더 에러체크 진행중 : 0x04, 각도 에러체크 진행중 : 0x08 state(하위 바이트) : 운동중 or 사용중 : 0x00, 대기중 : 0x01, 고장 : 0x02 | None | 0 |
| 15 | R | 16bit | use_state & error_state | use_state(상위 바이트) : 운동중 : 0x00, 정지 : 0x01, 측정중 : 0x02, 수동 : 0x03 / 측정완료 데이터 있을시 MSB가 1로 (예: 0x81, 측정데이터 전송이후에 다시 0으로) error_state(하위 바이트) : 비상정지 : 0x01, 과전류 : 0x02, 엔코더 : 0x03, 각도이탈 : 0x04, 통신에러 : 0x05 | None | 0 |
| 16 | R | 16bit | remain_time_and_cnt & upper_limit_angle | remain_time_and_cnt(상위 바이트) : 남은 운동시간 또는 남은 운동횟수 선택이므로 MSB가 0이면 남은 운동시간, 1이면 남은 운동횟수 upper_limit_angle(하위 바이트) : 측정 상한각 (-50 ~ 180) (0 ~ 230 통신후 실제 변수에 담기는 값은 -50 처리후 넣을것) | None | 0 |
| 17 | R | 16bit | lower_limit_angle & motion | lower_limit_angle(상위 바이트) : 측정 하한각 (-70 ~ 160) (0 ~ 230 통신후 실제 변수에 담기는 값은 -70 처리후 넣을것) motion : 0x00 : 팔꿈치, 0x01 : 어깨, 0x02 : 무릎, 0x03 : 손목, 0x04 : 발목 | None | 0 |
| | | | | | 총 18Word = 36Byte | |

Modbus 프로그램을 이용하여 데이터 Write
결과(예시)

Modbus Protocol (Write 후 Read 결과)

Write

Read

Tx:000000-02 10 00 00 00 00 0E 1C B3 00 D5 5C BB FC AD 6D C0 AC B7 8C 5B 00 00 00 00 00 50 1E 00 00 08 CB 00 00 00 01 89 24
Rx:000001-02 10 00 00 00 00 0E 41 FE

Tx:000002-02 04 00 00 00 00 12 70 34
Rx:000003-02 04 24 B3 00 D5 5C BB FC AD 6D C0 AC B7 8C 5B 00 00 00 00 00 50 1E 00 00 08 CB 00 00 00 0E 01 00 00 01 00 00 00 13 81

주소 코드

...

CRC16코드

| | Name | 0000 |
|----|-----------------------------------------|--------|
| 0 | user_name_1 | 0xB300 |
| 1 | user_name_2 | 0xD55C |
| 2 | user_name_3 | 0xBBFC |
| 3 | user_name_4 | 0xAD6D |
| 4 | user_name_5 | 0xC0AC |
| 5 | user_name_6 | 0xB78C |
| 6 | user_age & user_sex | 0x5B00 |
| 7 | user_num_high | 0x0000 |
| 8 | user_num_low | 0x0000 |
| 9 | upper_angle & lower_angle | 0x501E |
| A | stop_time & mode | 0x0000 |
| B | velocity_mode & exerc_time_and_exercnum | 0x08CB |
| C | special_angle & repeat_num | 0x0000 |
| D | special_location & motion | 0x0000 |
| E | check_state & state | 0x0E01 |
| F | use_state & error_state | 0x0000 |
| 10 | remain_time_and_cnt & upper_limit_angle | 0x0100 |
| 11 | lower_limit_angle & rsv1 | 0x0000 |

| Address | Read/Write | Type | Name | Description | Unit | Default |
|---------|------------|-------|-----------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------|---------|
| 0 | R/W | 16bit | user_name_1 | 0xD64D (홍) | None | 0xD64D |
| 1 | R/W | 16bit | user_name_2 | 0xAE38 (길) | None | 0xAE38 |
| 2 | R/W | 16bit | user_name_3 | 0xB3D9 (동) | None | 0xB3D9 |
| 3 | R/W | 16bit | user_name_4 | 0x0000 | None | 0 |
| 4 | R/W | 16bit | user_name_5 | 0x0000 | None | 0 |
| 5 | R/W | 16bit | user_name_6 | 0x0000 | None | 0 |
| 6 | R/W | 16bit | user_age & user_sex | user_age(상위 바이트) : 환자 나이 1 ~ 120세까지 가능 , user_sex(하위 바이트) : 환자 성별 (남 : 0, 여 : 1) | None | 0 |
| 7 | R/W | 16bit | user_num_high | user_num_high(상위 바이트) : 환자 등록번호 상위 2바이트 | None | 0 |
| 8 | R/W | 16bit | user_num_low | user_num_low(하위 바이트) : 환자 등록번호 하위 2바이트 | None | 0 |
| 9 | R/W | 16bit | upper_angle & lower_angle | upper_angle(상위 바이트) : 상한각 (-50 ~ 180) (0 ~ 230 통신후 실제 변수에 담기는 값은 -50 처리후 넣을것) lower_angle(하위 바이트) : 하한각 (-70 ~ 160) (0 ~ 230 통신후 실제 변수에 담기는 값은 -70 처리후 넣을것) | None | 0 |
| 10 | R/W | 16bit | stop_time & mode | stop_time(상위 바이트) : 상한 정지 시간(상위 4bit), 하한 정지 시간(하위 4bit) mode(하위 바이트) : 0x00(일반 등속), 0x01(적응 운동), 0x02(집중 운동), 0x03(일반 가속) | None | 0 |
| 11 | R/W | 16bit | velocity_mode & exerc_time_and_exercnum | velocity_mode(상위 바이트) : 0x01 : 1단계 ~ 0x09 : 9단계 exerc_time_and_exercnum(하위 바이트) : 운동시간 또는 운동횟수 선택이므로 MSB가 0이면 운동시간, 1이면 운동횟수 | None | 0 |
| 12 | R/W | 16bit | special_angle & repeat_num | special_angle(상위 바이트) : 집중/적응 운동 각도 최소 5°, 최대 15° repeat_num(하위 바이트) : 집중/적응 운동 반복 횟수 최소 3회, 최대 10회 | None | 0 |
| 13 | R/W | 16bit | special_location & rsv1 | special_location(상위 바이트) : 상한각 : 0x00, 하한각 : 0x01, 상하한각 : 0x02 rsv1(하위 바이트) : Dummy Data(실제로는 필요없는 값이므로 항상 0이다. 무시할것) | None | 0 |
| 14 | R | 16bit | check_state & state | check_state(상위 바이트) : 에러체크 완료 : 0x00, 전류 에러체크 진행중 : 0x02, 엔코더 에러체크 진행중 : 0x04, 각도 에러체크 진행중 : 0x08 state(하위 바이트) : 운동중 or 사용중 : 0x00, 대기중 : 0x01, 고장 : 0x02 | None | 0 |
| 15 | R | 16bit | use_state & error_state | use_state(상위 바이트) : 운동중 : 0x00, 정지 : 0x01, 측정중 : 0x02, 수동 : 0x03 / 측정완료 데이터 있을시 MSB가 1로 (예: 0x81, 측정데이터 전송이후에 다시 0으로) error_state(하위 바이트) : 비상정지 : 0x01, 과전류 : 0x02, 엔코더 : 0x03, 각도이탈 : 0x04, 통신에러 : 0x05 | None | 0 |
| 16 | R | 16bit | remain_time_and_cnt & upper_limit_angle | remain_time_and_cnt(상위 바이트) : 남은 운동시간 또는 남은 운동횟수 선택이므로 MSB가 0이면 남은 운동시간, 1이면 남은 운동횟수 upper_limit_angle(하위 바이트) : 측정 상한각 (-50 ~ 180) (0 ~ 230 통신후 실제 변수에 담기는 값은 -50 처리후 넣을것) | None | 0 |
| 17 | R | 16bit | lower_limit_angle & motion | lower_limit_angle(상위 바이트) : 측정 하한각 (-70 ~ 160) (0 ~ 230 통신후 실제 변수에 담기는 값은 -70 처리후 넣을것) motion : 0x00 : 팔꿈치, 0x01 : 어깨, 0x02 : 무릎, 0x03 : 손목, 0x04 : 발목 | None | 0 |
| | | | | | 총 18Word = 36Byte | |

사용자 이름
user_name_1 = 0xB300(대), user_name_2 = 0xD55C(한),
user_name_3 = 0xBBFC(민), user_name_4 = 0xAD6D(국),
user_name_5 = 0xC0AC(사), user_name_6 = 0xB78C(람)

참고 자료 [유니코드(UTF-8) 한글 코드 표]
<https://pooboo.tistory.com/131>

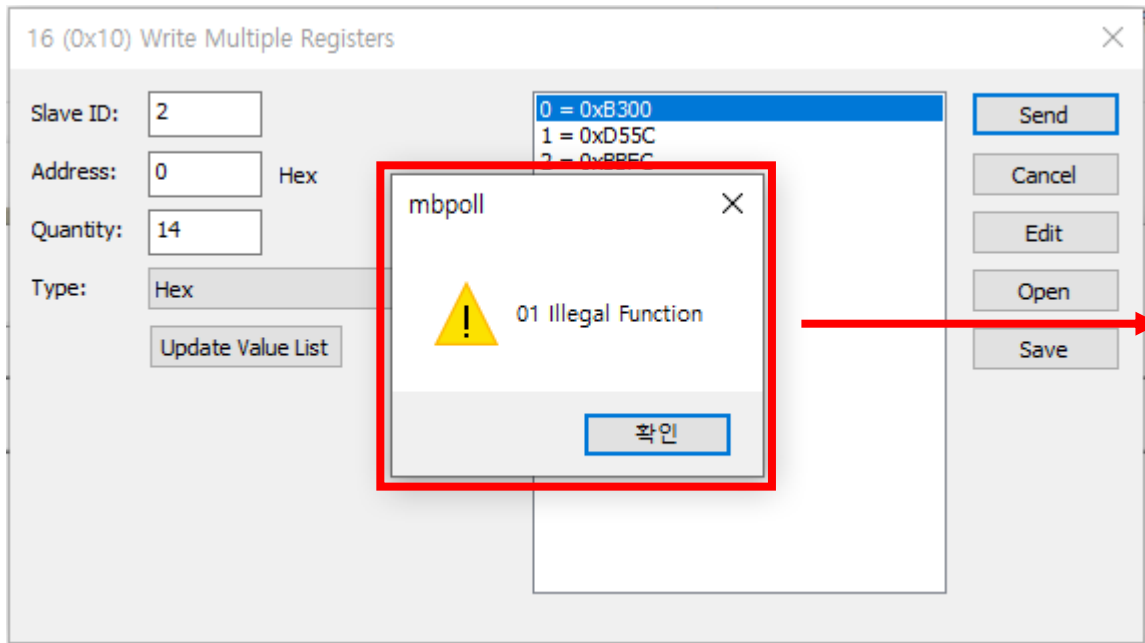


LCD 디스플레이 보드를 통해 LoRa가 모터드라이버로 데이터를 제대로 Write한 것을 확인할 수 있다.

Modbus 프로그램을 이용하여 운동 중 일 때 데이터
Write 시 예외 응답
결과(예시)

Modbus Protocol (모터 드라이버가 운동 중일 때 Write 시 데이터가 전송되지 않고 다시 예외처리로 반환되는 값이 온다.)

Tx:000004-02 10 00 00 00 0E 1C B3 00 D5 5C BB FC AD 6D C0 AC B7 8C 5B 00 00 00 00 00 50 1E 00 00 08 CB 00 00 00 01 89 24
Rx:000005-02 90 01 7D C0 → 모터드라이버에서 데이터를 이렇게 반환하면 데이터를 현재 쓸 수 없는 상황이다.



| ※ Exception Codes | | |
|-------------------|------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Exception Code | Name | Meaning |
| 01(01 hex) | Illegal Function | 쿼리에서 수신된 기능 코드는 슬레이브에 허용되는 작업이 아닙니다. 이는 기능 코드가 최신 장치에만 적용되고 선택한 장치에 구현되지 않았기 때문일 수 있습니다. 또한 슬레이브가 이 유형의 요청을 처리할 수 없는 잘못된 상태에 있음을 나타낼 수도 있습니다(예: 구성되지 않았고 레지스터 값을 반환하라는 요청을 받고 있기 때문). Poll Program Complete 명령이 실행된 경우 이 코드는 프로그램 기능이 선행되지 않았음을 나타냅니다. |

※ Exception Responses(예외 응답)

| Function Code in Request | Function Code in Exception Response |
|--------------------------|-------------------------------------|
| 01 (01 hex) 0000 0001 | 129 (81 hex) 1000 0001 |
| 02 (02 hex) 0000 0010 | 130 (82 hex) 1000 0010 |
| 03 (03 hex) 0000 0011 | 131 (83 hex) 1000 0011 |
| 04 (04 hex) 0000 0100 | 132 (84 hex) 1000 0100 |
| 05 (05 hex) 0000 0101 | 133 (85 hex) 1000 0101 |
| 06 (06 hex) 0000 0110 | 134 (86 hex) 1000 0110 |
| 15 (0F hex) 0000 1111 | 143 (8F hex) 1000 1111 |
| 16 (10 hex) 0001 0000 | 144 (90 hex) 1001 0000 |

▶ 예외 요청 예시

[Request] : (Master) ---> (Slave)

0A 01 04 A1 00 01 AC 63

0A : The Slave Address

01 : The Function Code 1(read Coil Status)

04 : The Data Address(High)

A1 : The Data Address(Low)

00 : The total number of coils requested(High)

01 : The total number of coils requested(Low)

AC : The CRC(cyclic redundancy check) for error checking(Low)

63 : The CRC(cyclic redundancy check) for error checking(High)

[Response] : (Slave) ---> (Master)

0A 81 02 B0 53

0A : The Slave Address

81 : The Function Code 1(read Coil Status - with the highest bit set)

02 : The Exception Code

B0 : The CRC(cyclic redundancy check) for error checking(Low)

53 : The CRC(cyclic redundancy check) for error checking(High)

Modbus CRC16

© Modbus CRC16

(Cyclic Redundancy Check 16비트)

이더넷, 모드버스, 블루투스 등 다양한 통신 프로토콜에서 널리 사용되는 오류 감지 방법

CRC16은 전송되는 데이터를 기반으로 체크섬 값을 생성하는 수학적 알고리즘이며 이러한 체크섬 값은 데이터에 추가되어 함께 전송된다.

데이터 수신자는 데이터에 대해 동일한 수학적 연산을 수행하고 결과 체크섬 값을 데이터와 함께 전송된 값과 비교합니다. 체크섬 값이 일치하면 데이터가 정확하게 전송된 것으로 간주한다.

단, 체크섬 값이 일치하지 않는 경우에는 전송 중 오류가 발생한 것으로 간주한다.

※현재 Modbus 프로그램에서는 자동으로 CRC16값을 계산후, CRC16데이터를 추가하여 전송하지만 프로그램이 없을때는 아래와 같이 함수를 만들어서 계산후에 보내줘야 한다.

```
word CRC16(byte *buf, word size)
{
    word crc=0xFFFF;
    byte i;
    unsigned char *p = buf;

    while (size--)
    {
        i=(crc ^ (*p++));
        crc = CRC16_Table[i & 0xFF] ^ (crc >> 8);
    }

    return crc;
}
```

```
// CRC16 is based on the polynomial x^16+x^15+x^2+1
word CRC16_Table[256] =
{
    0x0000, 0xC0C1, 0xC181, 0x0140, 0xC301, 0x03C0, 0x0280, 0xC241, 0xC601, 0x06C0, 0x0780, 0xC741, 0x0500, 0xC5C1, 0xC481, 0x0440,
    0xCC01, 0x0CC0, 0x0D80, 0xCD41, 0x0F00, 0xCFC1, 0xCE81, 0x0E40, 0x0A00, 0xCAC1, 0xCB81, 0x0B40, 0xC901, 0x09C0, 0x0880, 0xC841,
    0xD801, 0x18C0, 0x1980, 0xD941, 0x1B00, 0xDBC1, 0xDA81, 0x1A40, 0x1E00, 0xDEC1, 0xDF81, 0x1F40, 0xDD01, 0x1DC0, 0x1C80, 0xDC41,
    0x1400, 0xD4C1, 0xD581, 0x1540, 0xD701, 0x17C0, 0x1680, 0xD641, 0xD201, 0x12C0, 0x1380, 0xD341, 0x1100, 0xD1C1, 0xD081, 0x1040,
    0xF001, 0x30C0, 0x3180, 0xF141, 0x3301, 0xF3C1, 0xF281, 0x3240, 0x3600, 0xF6C1, 0xF781, 0x3740, 0xF501, 0x35C0, 0x3480, 0xF441,
    0x3C00, 0xFCC1, 0xFD81, 0x3D40, 0xFF01, 0x3FC0, 0x3E80, 0xFE41, 0xFA01, 0x3AC0, 0x3B80, 0xFB41, 0x3900, 0xF9C1, 0xF881, 0x3840,
    0x2800, 0xE8C1, 0xE981, 0x2940, 0xEB01, 0x2BC0, 0x2A80, 0xEA41, 0xEE01, 0x2EC0, 0x2F80, 0xEF41, 0x2D00, 0xEDC1, 0xEC81, 0x2C40,
    0xE401, 0x24C0, 0x2580, 0xE541, 0x2700, 0xE7C1, 0xE681, 0x2640, 0x2200, 0xE2C1, 0xE381, 0x2340, 0xE101, 0x21C0, 0x2080, 0xE041,
    0xA001, 0x60C0, 0x6180, 0xA141, 0x6301, 0xA3C1, 0xA281, 0x6240, 0x6600, 0xA6C1, 0xA781, 0x6740, 0xA501, 0x65C0, 0x6480, 0xA441,
    0x6C00, 0xACC1, 0xAD81, 0x6D40, 0xAF01, 0x6FC0, 0x6E80, 0xAE41, 0xAA01, 0x6AC0, 0x6B80, 0xAB41, 0x6900, 0xA9C1, 0xA881, 0x6840,
    0x7800, 0xB8C1, 0xB981, 0x7940, 0xBB01, 0x7BC0, 0x7A80, 0xBA41, 0xBE01, 0x7EC0, 0x7F80, 0xBF41, 0x7D00, 0xBDC1, 0xBC81, 0x7C40,
    0xB401, 0x74C0, 0x7580, 0xB541, 0x7700, 0xB7C1, 0xB681, 0x7640, 0x7200, 0xB2C1, 0xB381, 0x7340, 0xB101, 0x71C0, 0x7080, 0xB041,
    0x5000, 0x90C1, 0x9181, 0x5140, 0x9301, 0x53C0, 0x5280, 0x9241, 0x9600, 0x96C1, 0x9780, 0x9741, 0x9501, 0x95C0, 0x9480, 0x9441,
    0x9C01, 0x5CC0, 0x5D80, 0x9D41, 0x5F00, 0x9FC1, 0x9E81, 0x5E40, 0x5A00, 0x9AC1, 0x9B81, 0x5B40, 0x9901, 0x59C0, 0x5880, 0x9841,
    0x8801, 0x48C0, 0x4980, 0x8941, 0x4B00, 0x8BC1, 0x8A81, 0x4A40, 0x4E00, 0x8EC1, 0x8F81, 0x4F40, 0x8D01, 0x4DC0, 0x4C80, 0x8C41,
    0x4400, 0x84C1, 0x8581, 0x4540, 0x8701, 0x47C0, 0x4680, 0x8641, 0x8201, 0x42C0, 0x4380, 0x8341, 0x4100, 0x81C1, 0x8081, 0x4040
};
```

CRC16 함수(밑에는 예시)

MT_Driver_crc = CRC16(Rx4_buf, cal_number);

CRC16 테이블