



Modbus TCP 통신 구성하기

We reserve the right to change the content of this manual without prior notice. The information contained herein is believed to be accurate as of the date of publication, however, B&R makes no warranty, expressed or implied, with regards to the products or the documentation contained within this document. B&R shall not be liable in the event if incidental or consequential damages in connection with or arising from the furnishing, performance or use of these products. The software names, hardware names and trademarks used in this document are registered by the respective companies.

I Versions

Version	Date	Comment	Edited by
1.0	2017.11.20	First Edition	Hyeonjae Yun
2.0	2018.03.28	Modbus TCP – Master / Slave 구성 및 테스트 방법 추가	Eun Im, Sunggu Kang

Table 1: Versions

II Distribution

Name	Company, Department	Amount	Remarks

Table 2: Distribution

III Safety Notices

Safety notices in this document are organized as follows:

Safety notice	Description
Danger!	Disregarding the safety regulations and guidelines can be life-threatening.
Warning!	Disregarding the safety regulations and guidelines can result in severe injury or heavy damage to material.
Caution!	Disregarding the safety regulations and guidelines can result in injury or damage to material.
Information:	Important information used to prevent errors.

Table 3: Safety notices

IV Table of Contents

1 Modbus	4
1.1 Modbus 란?.....	4
1.2 Modbus TCP Frame format	4
1.3 Function Code	5
2 Automation Studio 에서 Modbus TCP 설정하기	6
2.1 Modbus TCP Slave 설정하기	6
2.1.1 Automation Studio 에서 기본 환경 구성	7
2.1.2 Automation Studio 에서 Modbus TCP Slave 설정	8
2.1.3 시뮬레이션 프로그램에서 테스트하기	11
2.2 Modbus TCP Master 설정하기	15
2.2.1 Automation Studio 에서 기본 환경 구성	16
2.2.2 Automation Studio 에서 Modbus TCP Master 설정	17
2.2.3 시뮬레이션 프로그램에서 테스트하기	21
3 예시	24
3.1 Festo 사 장비와 연결하기	24
4 자료출처	31

1 Modbus

1.1 Modbus 란?



Modbus¹는 Modicon(현재 Schneider Electric)에서 1979 년도에 PLCs(Programmable logic controllers)에서 사용하고자 만든 시리얼 통신 프로토콜이다. 현재는 산업분야에 다양한 전자 디바이스와 연결하는데 사용되는 표준 통신 프로토콜 중 하나이다.

Modbus 는 Slave/Master 기반의 아키텍처이며, 다양한 종류의 프로토콜이 존재한다. 그 중 Modbus RTU, Modbus ASCII, Modbus TCP/IP 등이 있다.

1.2 Modbus TCP Frame format

Name	Length (Bytes)	Function
Transaction identifier	2	서버와 클라이언트의 메시지 동기화
Protocol identifier	2	프로토콜 ID 를 나타내며, Modbus TCP 일 경우 0 를 사용
Length field	2	Length 필드 이후 해당 프레임의 마지막까지의 길이를 나타냄
Unit identifier	1	슬레이브 주소 (사용하지않을 경우 255)
Function code	1	평션 코드
Data bytes	n	데이터

- Unit identifier 는 독립 된 여러 개의 Modbus unit 을 지원하기 위해 하나의 IP 어드레스를 사용하는 Bridges, Routers, Gateways 와 같은 디바이스로 통신하는데 사용함.
- Modbus TCP 는 지정된 전용 포트번호 502 를 사용함.
- 프레임 포맷의 Function code 와 Data 부분을 Protocol Data Unit(PDU)라고 부르며, Request PDU, Response PDU, Exception Response PDU 로 정의함.
- 데이터 타입은 비트단위 또는 워드(16Bits)로 정보 조작이 용이함.

¹ <https://en.wikipedia.org/wiki/Modbus>

1.3 Function Code

Function Type		Function name	Function Code
Bit access	Physical Discrete Inputs	Read Discrete Inputs	2
	Internal Bits or Physical Coils	Read Coils	1
		Write Single Coil	5
		Write Multiple Coils	15
2 Bytes access	Physical Input Registers	Read Input Registers	4
	Internal Registers or Physical Output Registers	Read Multiple Holding Registers	3
		Write Single Holding Register	6
		Write Multiple Holding Registers	16

- FC1 은 Output 을 위한 Read, FC2 는 Input 을 위한 Read

2 Automation Studio 에서 Modbus TCP 설정하기

B&R PLC 는 Modbus TCP Master 와 Slave 를 지원한다.

Modbus TCP 에 대한 자세한 정보는 Automation Studio help 에서 확인할 수 있다.



Communication W Fieldbus systems W Modbus TCP

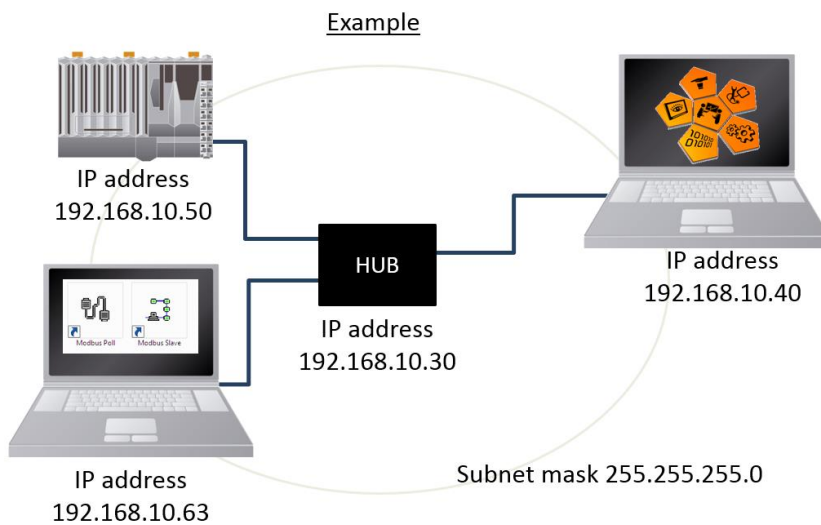
2.1 Modbus TCP Slave 설정하기

Automation Studio 에서 Modbus TCP Slave 설정 방법을 설명한다.

테스트 하드웨어 목록

- Automation Studio 4.3.4 가 설치된 PC 1 대
- Modbus Poll² 이 설치된 PC 1 대
- X20CP1586 1 대
- 허브 1 대

네트워크 구성

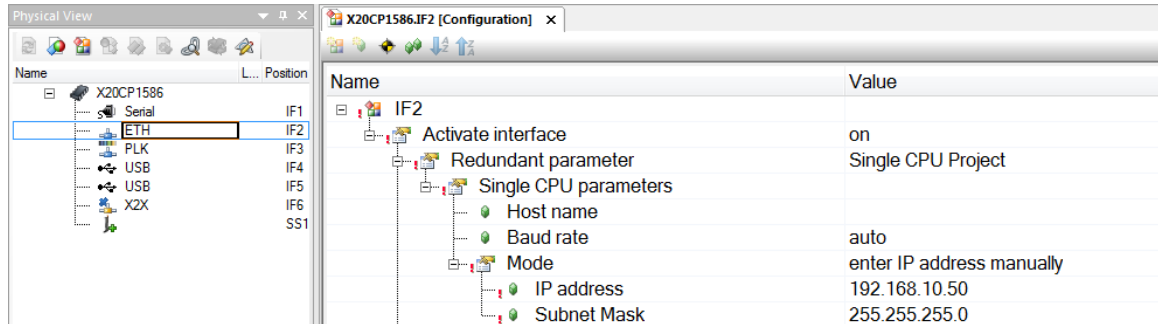


	B&R PLC (X20CP1586)	Automation Studio 가 설치된 PC	Modbus TCP 시뮬레이션이 설치된 PC	허브(HUB)
IP Address	192.168.10.50	192.168.10.40	192.168.10.63	192.168.10.30
Subnet Mask	255.255.255.0	255.255.255.0	255.255.255.0	255.255.255.0
기타	-	-		자동 할당 IP 대역 1192.168.10.40~49

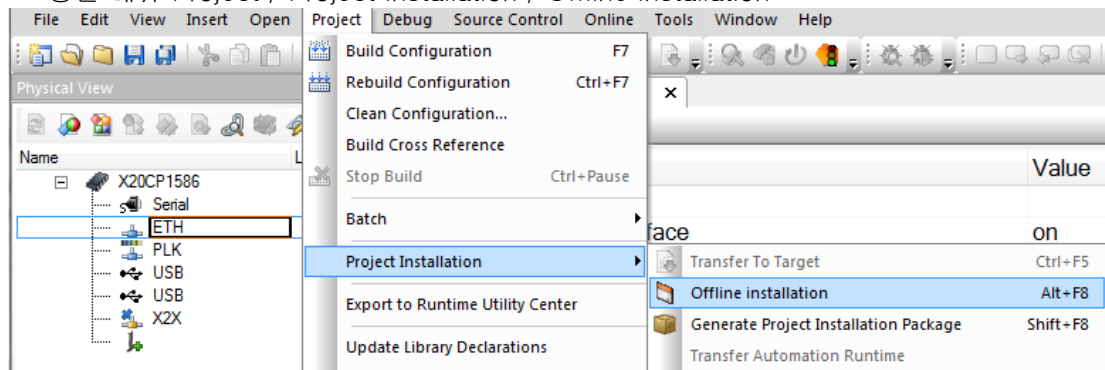
² <http://www.modbustools.com/>: 모드버스 통신 시뮬레이션 프로그램, Master simulation

2.1.1 Automation Studio 에서 기본 환경 구성

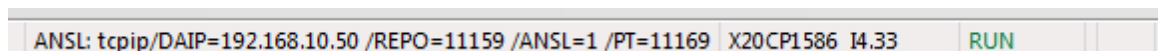
- 1) Automation Studio 에서 새 프로젝트 생성
- 2) 하드웨어 구성하기
예시에서는 X20CP1586 을 기준으로 하드웨어를 구성한다.
- 3) Physical View 에서 ETH 선택 후 마우스 우 클릭 / Configuration 누르기
- 4) PLC 의 IP 설정



- 5) CF 카드를 카드리더기에 연결하고 CF 카드를 굽기
상단 메뉴 Project / Project Installation / Offline installation

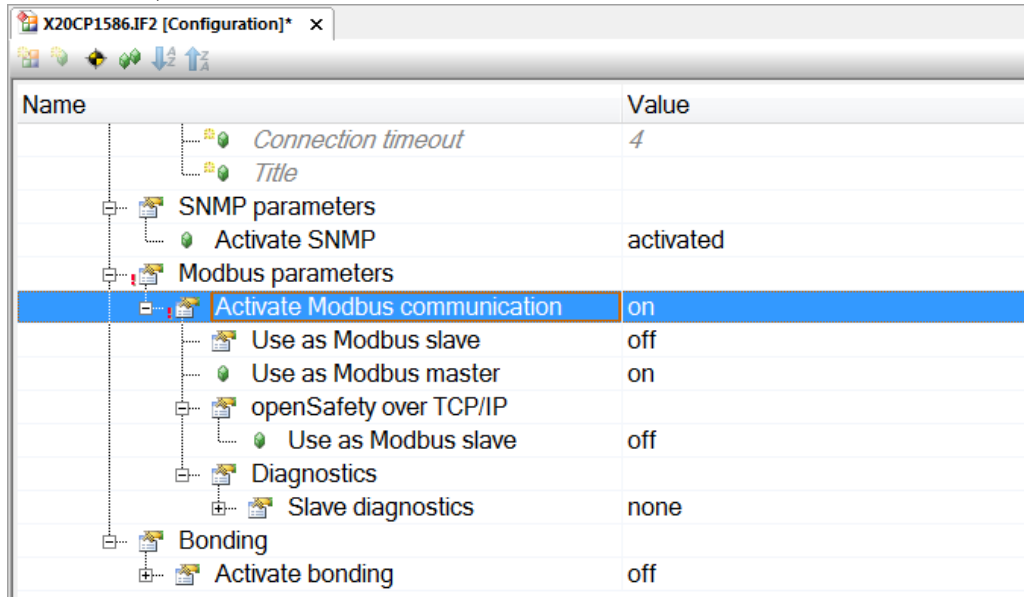


- 6) PLC 에 CF 카드를 끼우고 전원 공급 후, RUN 확인하기
- 7) PLC 와 Automation Studio 연결하기
상단 메뉴 Online / Settings 이용하여 Status Bar 에 RUN 확인



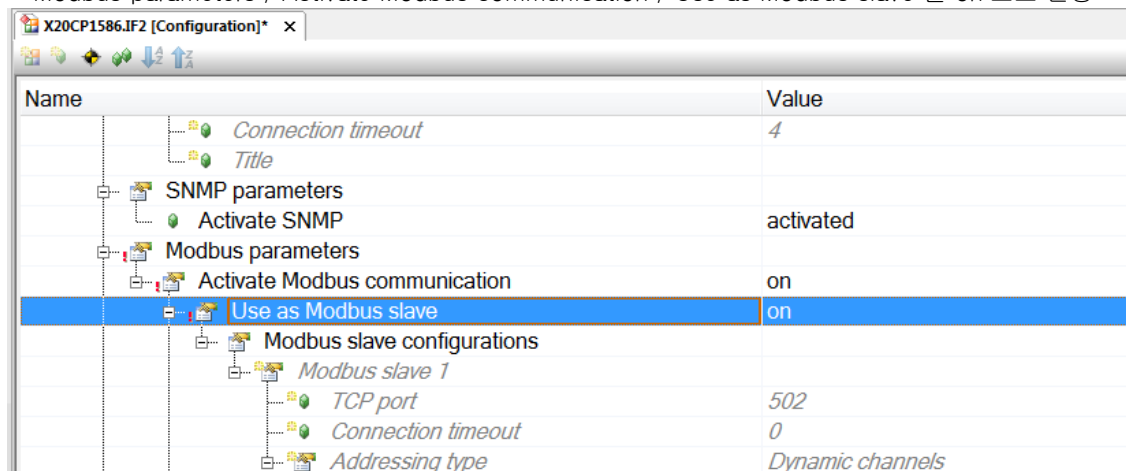
2.1.2 Automation Studio 에서 Modbus TCP Slave 설정

- 1) Physical View 에서 ETH 선택 후 마우스 우 클릭 / Configuration 누르기
Modbus parameters / Activate Modbus communication 을 On 으로 변경



Name	Value
Connection timeout	4
Title	
SNMP parameters	
Activate SNMP	activated
Modbus parameters	
Activate Modbus communication	on
Use as Modbus slave	off
Use as Modbus master	on
openSafety over TCP/IP	
Use as Modbus slave	off
Diagnostics	
Slave diagnostics	none
Bonding	
Activate bonding	off

- 2) Modbus slave 를 사용하기 위해서,
Modbus parameters / Activate Modbus communication / Use as Modbus slave 를 on 으로 변경



Name	Value
Connection timeout	4
Title	
SNMP parameters	
Activate SNMP	activated
Modbus parameters	
Activate Modbus communication	on
Use as Modbus slave	on
Modbus slave configurations	
Modbus slave 1	
TCP port	502
Connection timeout	0
Addressing type	Dynamic channels

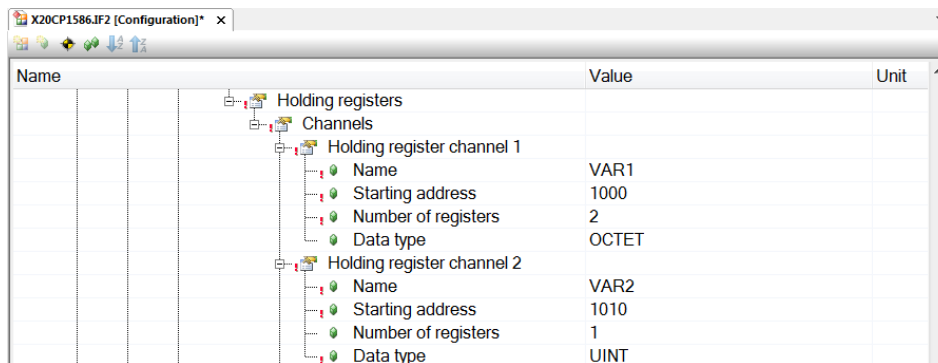
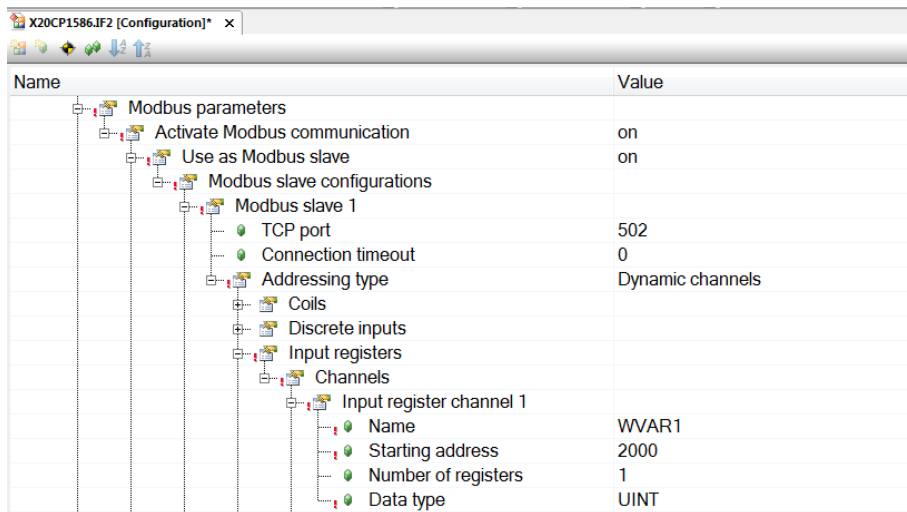
- 3) 통신에 필요한 파라미터 설정
레지스터 이름 Value 에 고유 이름을 기입하고 엔터를 누르면 짙은 검정 글자로 활성화된다.
Modbus slave 의 TCP port 는 기본 설정 값, 502 를 사용한다.

<예시>

Modbus slave 1/ TCP port	502
--------------------------	-----

Addressing type/	Input registers
Addressing type/ Input registers/ Channels/ Input register channel 1 /Name	WVAR1
Addressing type/ Input registers/ Channels/ Input register channel 1 /Starting address	2000
Addressing type/ Input registers/ Channels/ Input register channel 1 /Number of registers	1
Addressing type/ Input registers/ Channels/ Input register channel 1 /Data type	UINT

Addressing type/	Holding registers	Holding registers
Addressing type/ Holding registers / Channels/ Input register channel 1 /Name	VAR1	VAR2
Addressing type/ Holding registers / Channels/ Input register channel 1 /Starting address	1000	1010
Addressing type/ Holding registers / Channels/ Input register channel 1 /Number of registers	2	1
Addressing type/ Holding registers / Channels/ Input register channel 1 /Data type	OCTECT	UINT



- 4) 프로그램에서 통신 데이터를 활용하려면 아래 과정을 진행한다.
ToolBox 에서 ST program 을 추가하고 프로그램 이름은 'TEST'로 변경한다

- 5)아래 표와 같이 변수를 생성한다.

Name	Type
VAR_A	UDINT
VAR_B	UINT
VAR_RA	UINT
VAR_RB	UDINT

- 6)ST 프로그램에 아래와 같이 코딩한다.

```
PROGRAM _CYCLIC

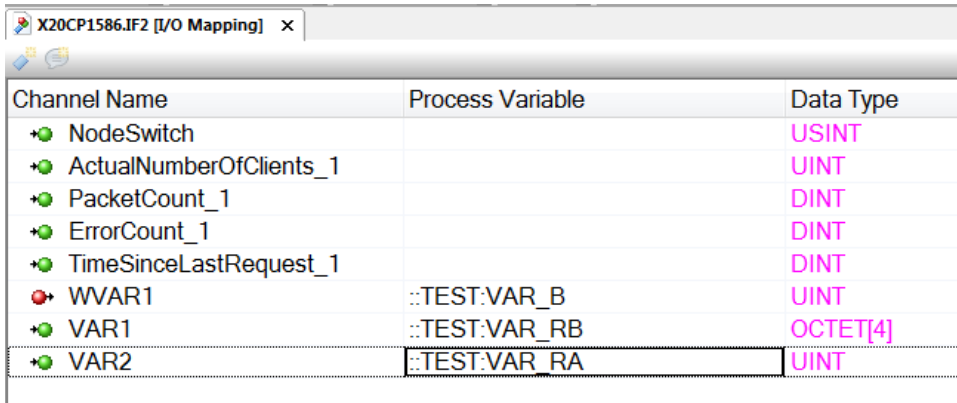
    VAR_A;
    VAR_B;

    VAR_RA;
    VAR_RB;

END_PROGRAM
```

- 7)Physical View 에서 ETH 선택 후 마우스 우 클릭 / I/O Mapping 누르기

- 8)Process Variables 에 필요한 변수 맵핑하기



Channel Name	Process Variable	Data Type
NodeSwitch		USINT
ActualNumberOfClients_1		UINT
PacketCount_1		DINT
ErrorCount_1		DINT
TimeSinceLastRequest_1		DINT
WVAR1	::TEST:VAR_B	UINT
VAR1	::TEST:VAR_RB	OCTET[4]
VAR2	::TEST:VAR_RA	UINT

- 9)프로젝트 빌드(Build) / 트랜스퍼(Transfer)

- 10) Warm restart 수행

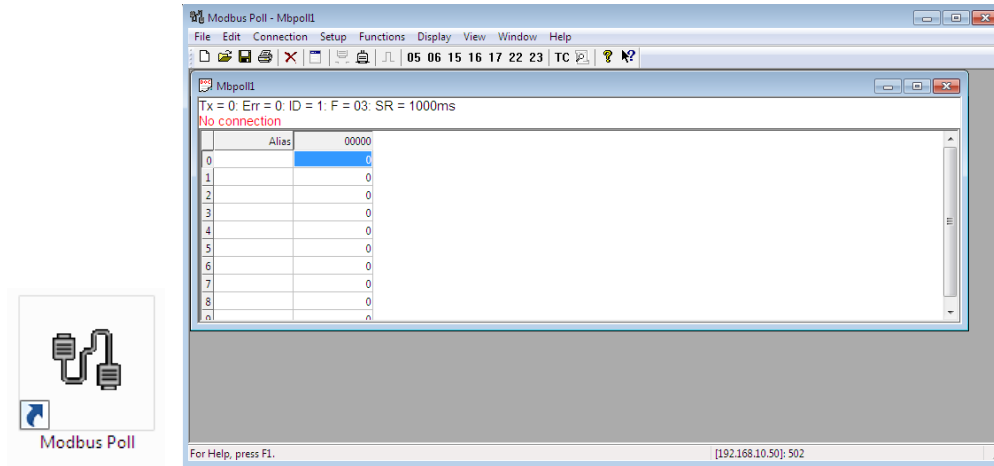
- 11) Automation Studio 에서 Modbus TCP Slave 설정 완료

2.1.3 시뮬레이션 프로그램에서 테스트하기

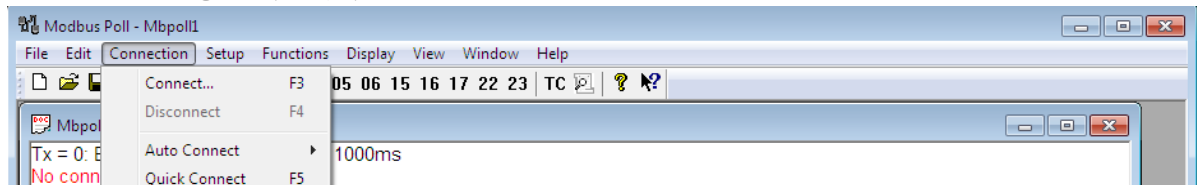
Modbus TCP Slave 를 테스트하기 위해서는 Modbus Poll 프로그램을 이용하여 테스트 할 수 있다.
 Modbus Poll 만 실행할 경우 PLC 와 같은 네트워크 상에 존재하면 된다.
 (PLC 의 앞 세자리 IP 는 동일하고 마지막 네번째만 다른 숫자, 서브넷 마스크는 동일)

Modbus Poll 사용법에 대한 자세한 내용은 Modbus Poll 도움말을 참조하세요.
 이 매뉴얼에서는 B&R CPU 와의 간단한 연동 테스트 방법만 설명합니다.

1) Modbus Poll 프로그램 실행하기

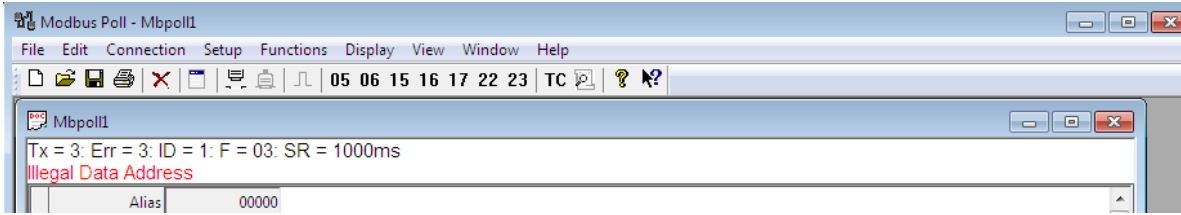


- 2) Modbus Slave 연결하기 Connection / Connect...
- 3) Modbus Slave 정보 기입하기

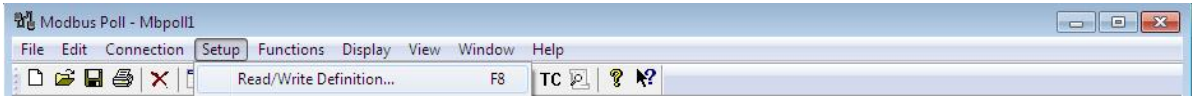


Modbus Poll 연결 설정 화면	Automation Studio 의 설정 화면 Physical View 에서 ETH 선택 후 마우스 우 클릭 / Configuration

4) 연결은 되었으나 레지스터 정보가 올바르지 않으면 'Illegal Data Address' 가 나타난다.



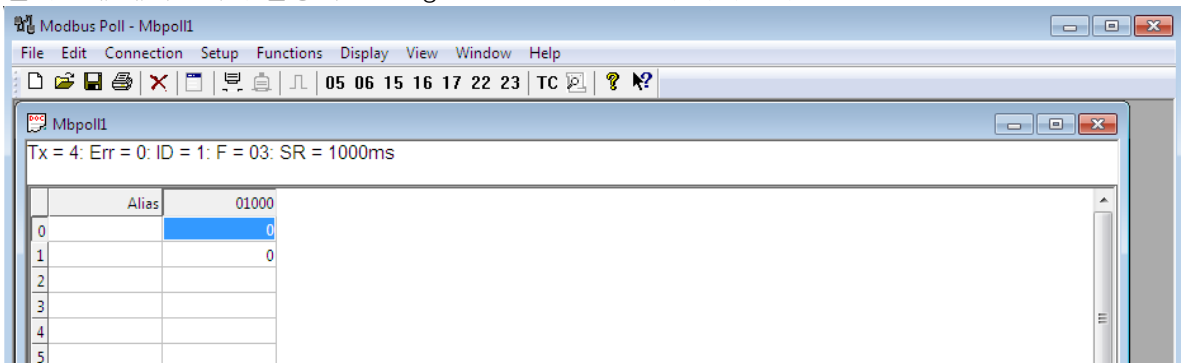
5) 해당 레지스터 정의하기 Setup / Read/Write Definition..



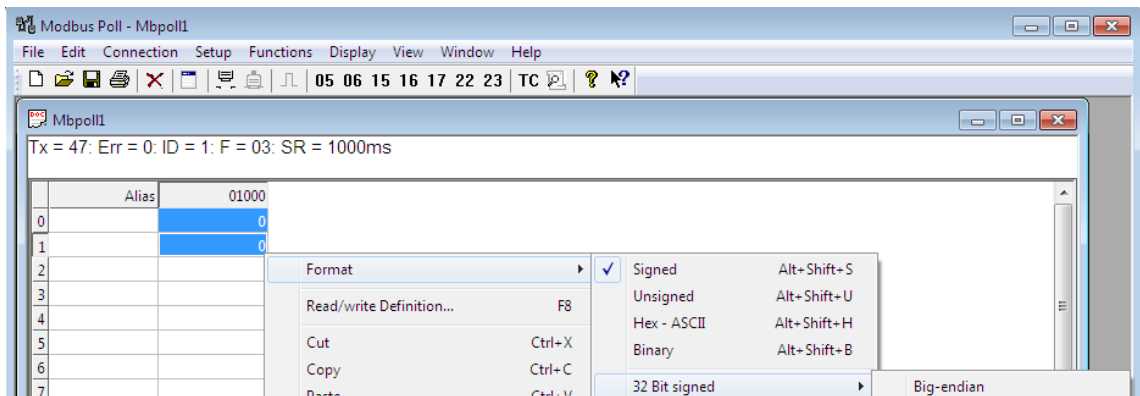
6) 첫 번째 예시 - Holding registers / VAR1

<p>Modbus Poll 설정 화면</p>	<p>Automation Studio 의 설정 화면 Physical View 에서 ETH 선택 후 마우스 우 클릭 / Configuration</p> <table border="1"> <thead> <tr> <th>Name</th> <th>Value</th> </tr> </thead> <tbody> <tr> <td>Holding registers</td> <td></td> </tr> <tr> <td>Channels</td> <td></td> </tr> <tr> <td> Holding register channel 1</td> <td></td> </tr> <tr> <td> Name</td> <td>VAR1</td> </tr> <tr> <td> Starting address</td> <td>1000</td> </tr> <tr> <td> Number of registers</td> <td>2</td> </tr> <tr> <td> Data type</td> <td>OCTET</td> </tr> <tr> <td> Holding register channel 2</td> <td></td> </tr> <tr> <td> Name</td> <td>VAR2</td> </tr> <tr> <td> Starting address</td> <td>1010</td> </tr> <tr> <td> Number of registers</td> <td>1</td> </tr> <tr> <td> Data type</td> <td>UINT</td> </tr> </tbody> </table>	Name	Value	Holding registers		Channels		Holding register channel 1		Name	VAR1	Starting address	1000	Number of registers	2	Data type	OCTET	Holding register channel 2		Name	VAR2	Starting address	1010	Number of registers	1	Data type	UINT
Name	Value																										
Holding registers																											
Channels																											
Holding register channel 1																											
Name	VAR1																										
Starting address	1000																										
Number of registers	2																										
Data type	OCTET																										
Holding register channel 2																											
Name	VAR2																										
Starting address	1010																										
Number of registers	1																										
Data type	UINT																										

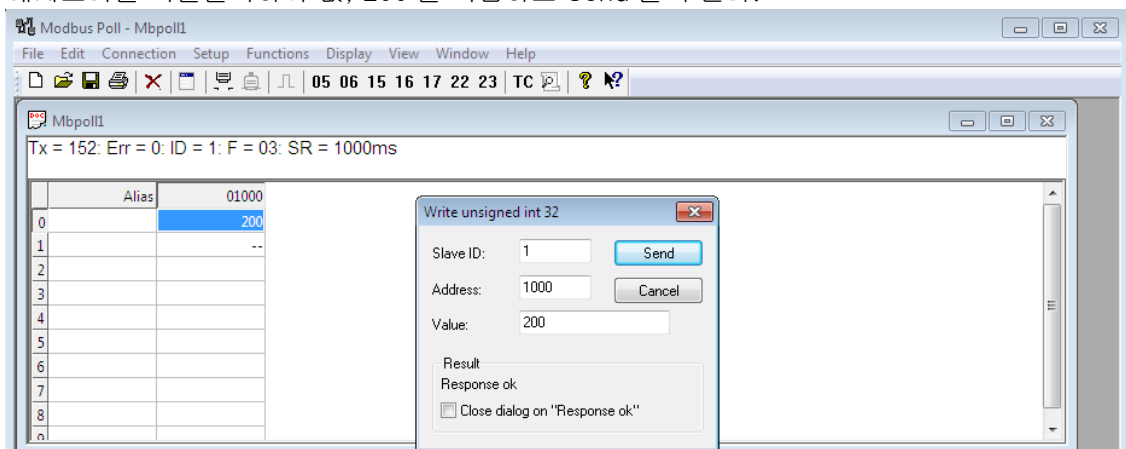
7) 올바르게 레지스터가 설정되면 'Illegal Data Address' 가 사라진다.



8) 정확한 값을 주고 받기 위하여 데이터 형식을 변경한다.
두 칸을 동시에 선택하고(Number of registers 2) 마우스 우 클릭.
Format / 32Bit signed / Big-endian 선택



9) 레지스터를 더블클릭하여 값, 200 을 기입하고 Send 를 누른다.

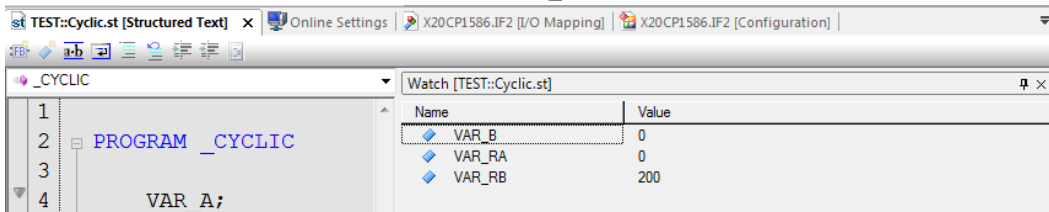


10) Automation Studio / Monitor 모드(단축키 Ctrl + M 또는 상단메뉴 View/ Monitor)에서 해당 레지스터 값을 확인한다.
 위치: Physical view/ ETH 선택 후 마우스 우 클릭 / I/O mapping

I/O mapping 에서는 확인 할 수 없다.

Channel Name	Physical Value	ForceActivated	ForceActivated Value	Process Variable
NodeSwitch	20	<input type="checkbox"/>	0	
ActualNumberOfClients_1	1	<input type="checkbox"/>	0	
PacketCount_1	882	<input type="checkbox"/>	0	
ErrorCount_1	159	<input type="checkbox"/>	0	
TimeSinceLastRequest_1	810	<input type="checkbox"/>	0	
WVAR1	0	<input type="checkbox"/>	0	::TEST:VAR_B
VAR1	Not available	<input type="checkbox"/>		::TEST:VAR_RB
VAR2	0	<input type="checkbox"/>	0	::TEST:VAR_RA
MODBUS.ModbusMasterDiag	FALSE	<input type="checkbox"/>	FALSE	

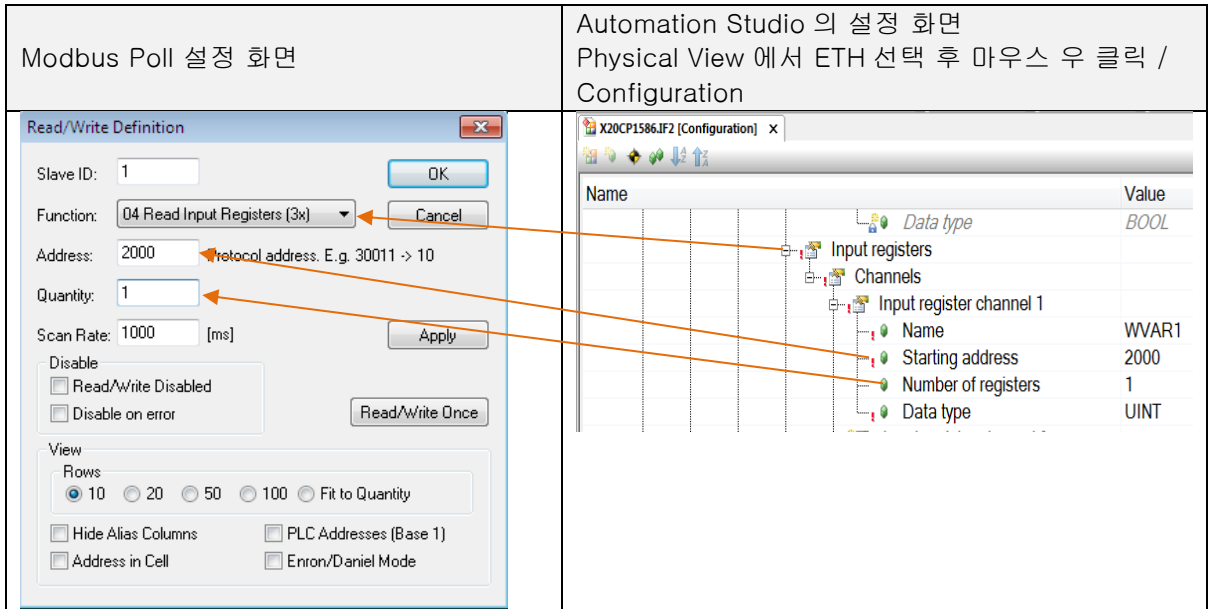
11) 프로그램에서 확인한다. VAR1 에 연결된 VAR_B 값이 200 으로 들어온 것을 확인할 수 있다.



12) 완료

1~5 는 위와 동일

6) 두 번째 예시 – Input registers / WVAR1

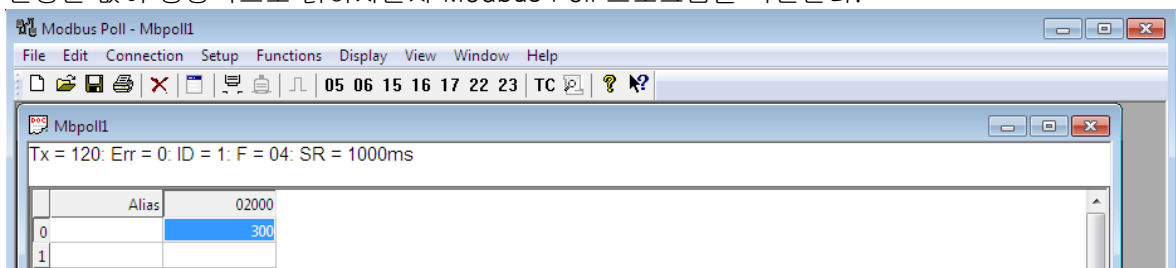


7) Automation Studio / Monitor 모드에서 값을 기입한다. 프로그램에서 기입하거나, Physical view/ ETH 선택 후 마우스 우 클릭 / I/O mapping 에서 WVAR1 에 연결된 변수 VAR_B 를 이용하여 값을 300 으로 변경한다.

Name	Value
VAR_B	300
VAR_RA	0
VAR_RB	200

Channel Name	Physical Value	ForceActivated	ForceActivated Value	Process Variable
NodeSwitch	20	<input type="checkbox"/>	0	
ActualNumberOfClients_1	1	<input type="checkbox"/>	0	
PacketCount_1	1069	<input type="checkbox"/>	0	
ErrorCount_1	161	<input type="checkbox"/>	0	
TimeSinceLastRequest_1	200	<input type="checkbox"/>	0	
WVAR1	300	<input checked="" type="checkbox"/>	300	::TEST:VAR_B
VAR1	Not available	<input type="checkbox"/>		::TEST:VAR_RB
VAR2	0	<input type="checkbox"/>	0	::TEST:VAR_RA
MODBUS.ModbusMasterDiag	FALSE	<input type="checkbox"/>	FALSE	

8) 변경된 값이 정상적으로 읽혀지는지 Modbus Poll 프로그램을 확인한다.



9) 완료

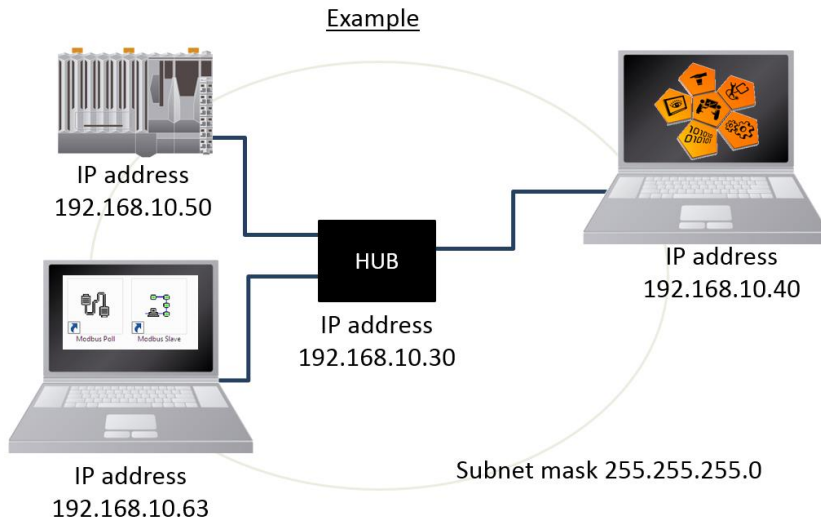
2.2 Modbus TCP Master 설정하기

Automation Studio 에서 Modbus TCP Master 설정 방법을 설명한다.

테스트 하드웨어 목록

- Automation Studio 4.3.4 가 설치된 PC 1 대
- Modbus Slave³ 가 설치된 PC 1 대
- X20CP1586 1 대
- 허브 1 대

네트워크 구성

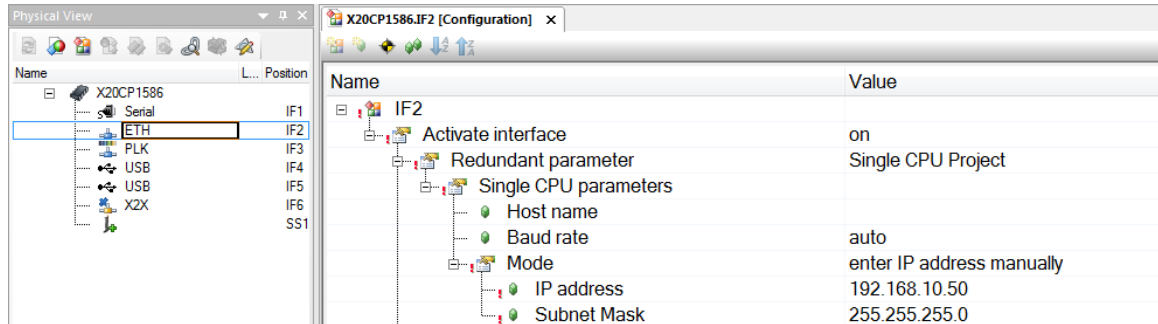


	B&R PLC (X20CP1586)	Automation Studio 가 설치된 PC	Modbus TCP 시뮬레이션이 설치된 PC	허브(HUB)
IP Address	192.168.10.50	192.168.10.40	192.168.10.63	192.168.10.30
Subnet Mask	255.255.255.0	255.255.255.0	255.255.255.0	255.255.255.0
기타	-	-		자동 할당 IP 대역 1192.168.10.40~49

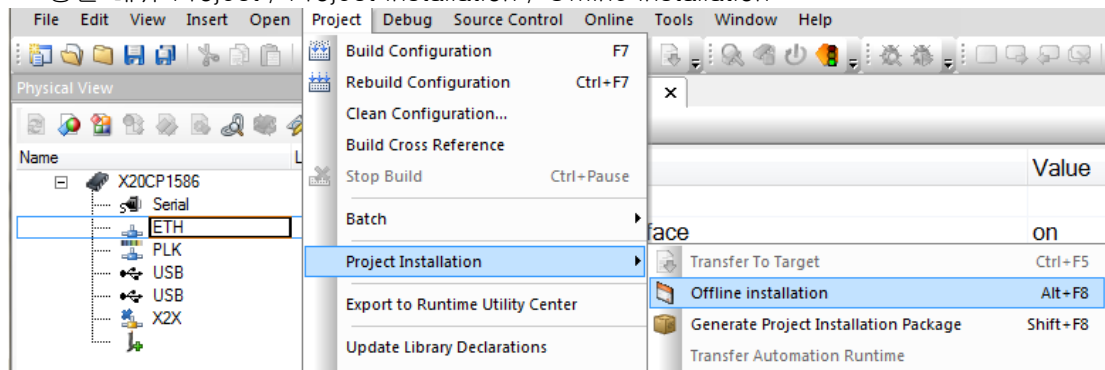
³ <http://www.modbustools.com/>: 모드버스 통신 시뮬레이션 프로그램, Slave simulation

2.2.1 Automation Studio 에서 기본 환경 구성

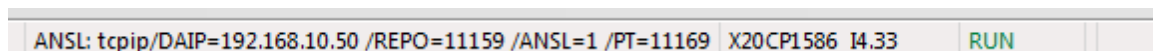
- 1) Automation Studio 에서 새 프로젝트 생성
- 2) 하드웨어 구성하기
예시에서는 X20CP1586 을 기준으로 하드웨어를 구성한다.
- 3) Physical View 에서 ETH 선택 후 마우스 우 클릭 / Configuration 누르기
- 4) PLC 의 IP 설정



- 5) CF 카드를 카드리더기에 연결하고 CF 카드를 굽기
상단 메뉴 Project / Project Installation / Offline installation



- 6) PLC 에 CF 카드를 끼우고 전원 공급 후, RUN 확인하기
- 7) PLC 와 Automation Studio 연결하기
상단 메뉴 Online / Settings 이용하여 Status Bar 에 RUN 확인

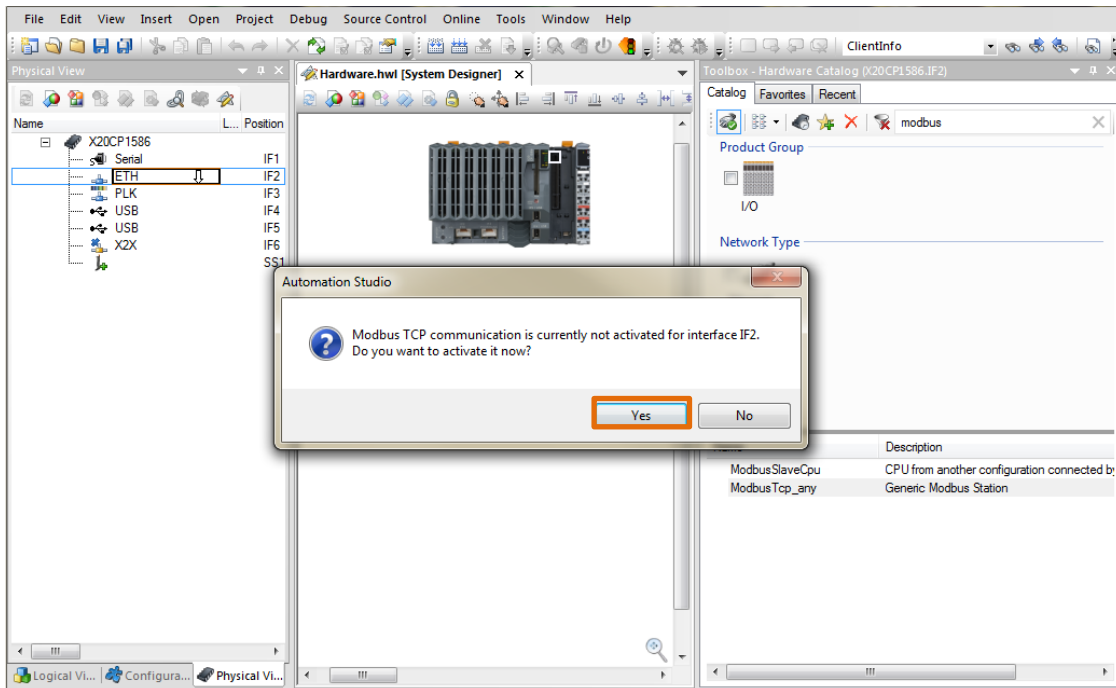
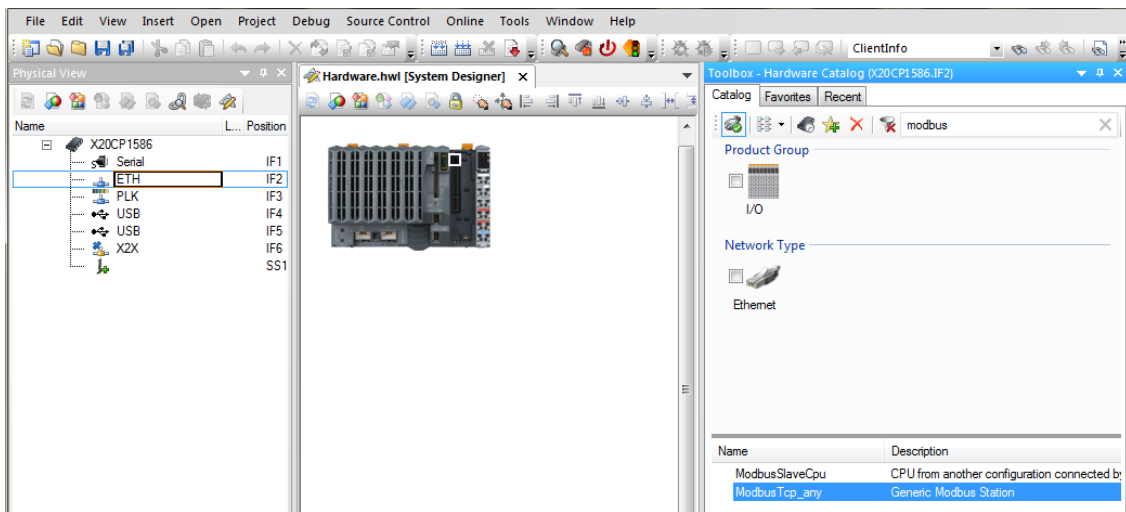


2.2.2 Automation Studio 에서 Modbus TCP Master 설정

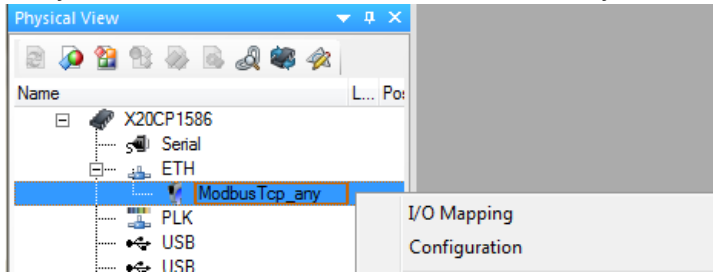
- 1) ToolBox 에서 Physical View / (CPU) / ETH 에 “Modbus Tcp_any”를 추가하기
 하드웨어 추가 후, 인터페이스 IF3 에 ModbusTCP 통신 활성화를 위한 팝업 창에서 ‘Yes’ 를
 선택함. (Modbus TCP 통신 기본 설정: 비활성화)

※(참조) ModbusTCP 통신을 직접 활성화 설정할 경우

- Physical View 에서 ETH 선택 후 마우스 우 클릭 / Configuration 누르기
- Modbus parameters / Activate Modbus communication 을 On 으로 변경

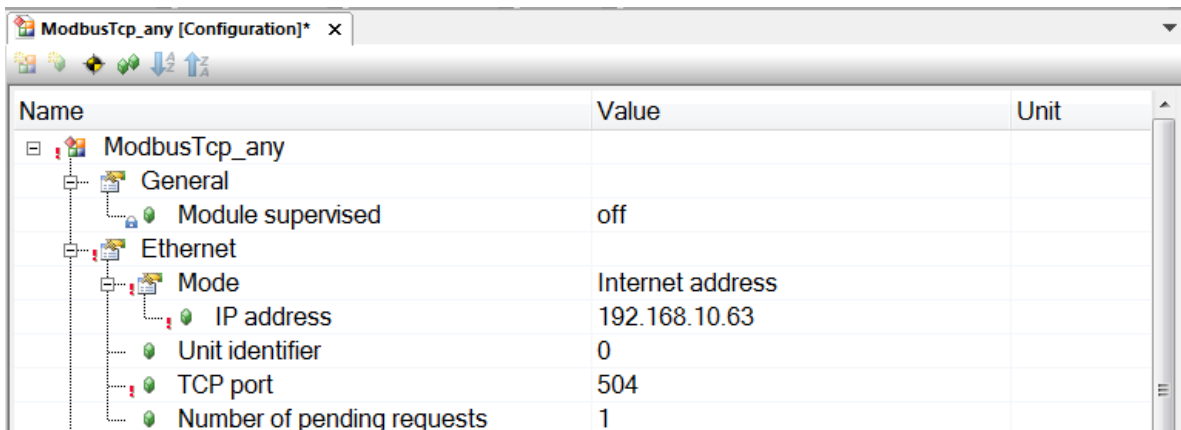


2) Physical View / (CPU) / ETH / ModbusTCP_any 선택 후 마우스 우 클릭 / Configuration



3) 해당 Master 의 IP 와 TCP port 를 기입하기

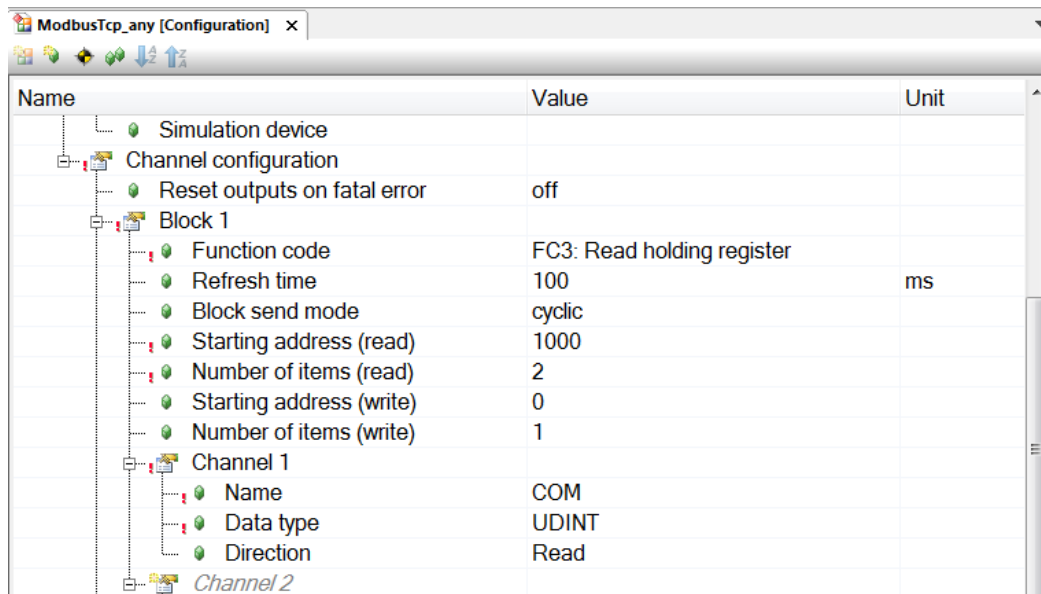
Ethernet /Mode /IP address	192.168.10.63
Ethernet /TCP port	504



4) 평선 코드를 선택하면 활성화되면서 검은 글자로 나타난다.

<예시>

Channel configuration / Block1/ Function code	FC3: Read holding register
Channel configuration / Block1/ Refresh time	100
Channel configuration / Block1/ Block send mode	Cyclic
Channel configuration / Block1/ Starting address (read)	1000
Channel configuration / Block1/ Number of items (read)	2
Channel configuration / Block1/ Starting address (write)	0
Channel configuration / Block1/ Number of items (write)	1
Channel configuration / Block1/ Channel 1/ Name	COM
Channel configuration / Block1/ Channel 1/ Data type	UDINT
Channel configuration / Block1/ Channel 1/ Direction	Read



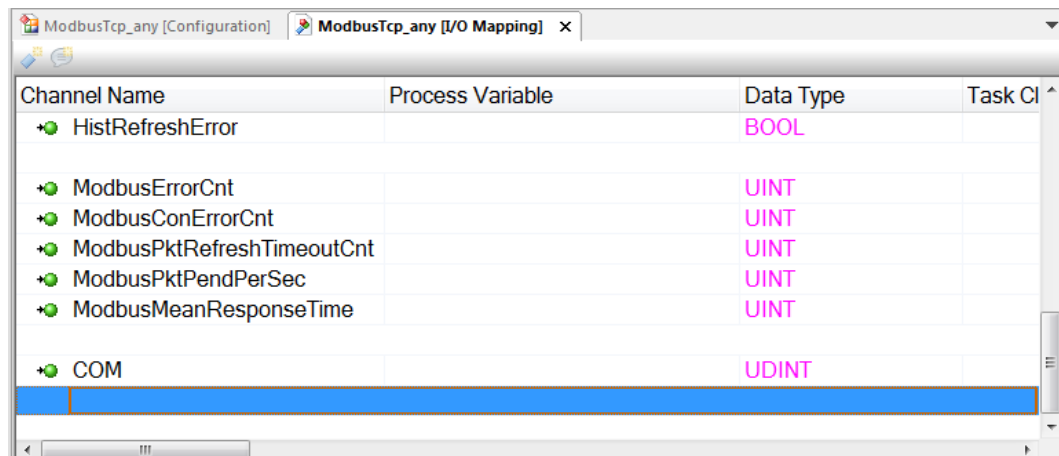
5) 필요시 변수를 만들고 해당 변수를 I/O mapping 을 수행한다.

위치: Physical View / (CPU) / ETH / ModbusTCP_any 선택 후 마우스 우 클릭 / I/O mapping

6) 현 예제에서는 관련 설정을 생략한다.

필요시 '[Automation Studio 에서 Modbus TCP Slave 설정](#)'을 참조한다.

7) Physical View / (CPU) / ETH / ModbusTCP_any 선택 후 마우스 우 클릭 / I/O mapping 을 열면 "COM" 이 생성된 것을 확인할 수 있다.



참조		설명
General	Module monitoring	모듈 모니터링은 Modbus TCP Slaves 에서 지원하지 않으므로 항상 Disabled 되어 있으며, 변경이 불가함.
Ethernet	Mode	Modbus TCP Slave 의 IP Address 또는 호스트 네임
	IP Address	Modbus TCP Slaves 의 IP Address
	Unit identifier	Slave 가 Modbus RTU Slave 로 사용하지 않는다면 값은 0,1 또는 255 로 설정 가능.
	TCP port	Modbus 를 위한 TCP port 는 항상 502 로 설정해야한다.
	Number of simultaneous package requests	동시에 보내는 Modbus packets 수 설정. 조금 더 복잡한 Configuration 을 위한 네트워크 Load 를 분배에 도움을 준다.
Timeouts occur	Cycle time violation [ms]	Modbus 확인 요청이 설정 시간 동안 응답이 없을 경우 Cycle time violation 발생.
	Plug and play timer [s]	네트워크에 새로운 Slave module 이 존재하는지를 결정하기 위해 Modbus Master 가 사용하는 간격.

8) 빌드(Build) 후 트랜스퍼(Transfer)

9) Warm restart 수행

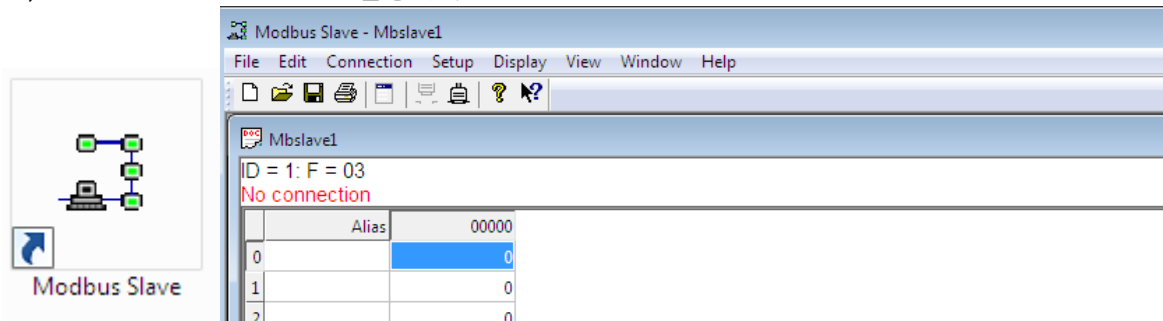
10) Automation Studio 에서 Modbus TCP Master 설정 완료

2.2.3 시뮬레이션 프로그램에서 테스트하기

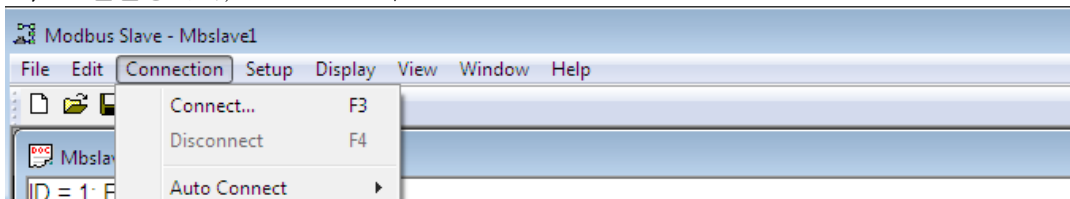
Modbus TCP Master 를 테스트하기 위해서는 Modbus Slave 프로그램을 이용하여 테스트 할 수 있다. Modbus Slave 만 실행할 경우 PLC 에서 설정한 Master 와 동일한 IP 와 서브넷 마스크로 설정해야 한다. (현재 예제에서는 192.168.10.63 으로 설정하였습니다.)

Modbus Slave 사용법에 대한 자세한 내용은 Modbus Slave 도움말을 참조하세요. 이 매뉴얼에서는 B&R CPU 와의 간단한 연동 테스트 방법만 설명합니다.

1) Modbus Slave 프로그램 실행하기



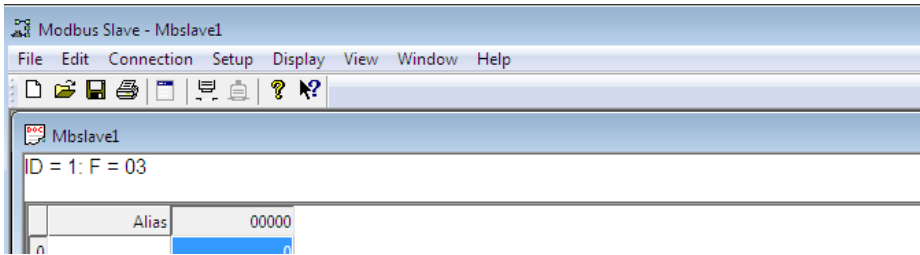
2) 연결설정하기, Connection / Connect...



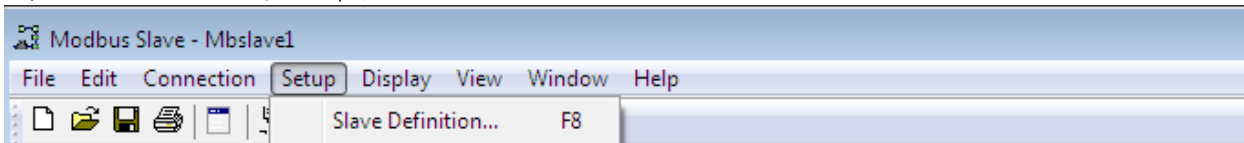
Modbus Slave 연결 설정 화면	Automation Studio 의 설정 화면 Physical View / (CPU) / ETH / ModbusTCP_any 선택 후 마우스 우 클릭 / Configuration

※주의: Modbus Slave 프로그램이 설치된 PC 도 동일한 IP(192.168.10.63)로 설정되어야만 정상 작동함.

3) 정상 연결 후, No Connection 이 사라짐

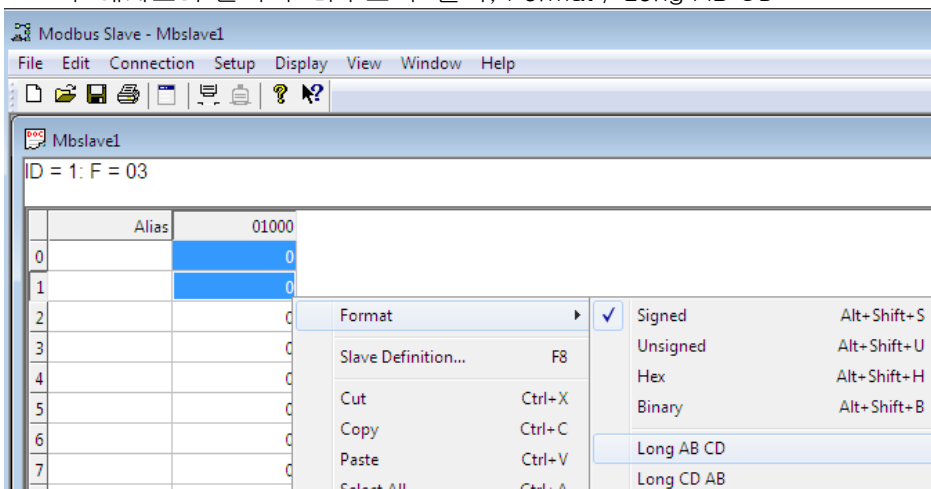


4) 레지스터 정의하기, Setup / Slave Definition 클릭

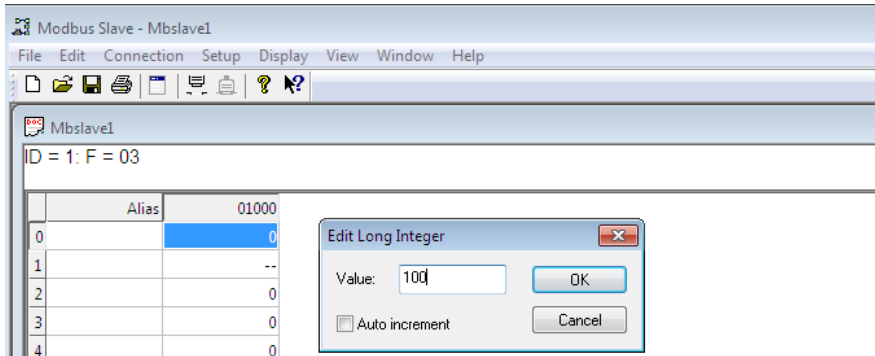


Modbus Slave 설정 화면	Automation Studio 의 설정 화면 Physical View / (CPU) / ETH / ModbusTCP_any 선택 후 마우스 우 클릭 / Configuration																																				
<p>The 'Slave Definition' dialog box is shown. It contains the following fields and options:</p> <ul style="list-style-type: none"> Slave ID: 1 Function: 03 Holding Register (4x) Address: 1000 Quantity: 10 View: Rows (10, 20, 50, 100, Fit to Quantity) Hide Alias Columns: <input type="checkbox"/> PLC Addresses (Base 1): <input type="checkbox"/> Error Simulation: Skip response <input type="checkbox"/> Response Delay: 0 [ms] Insert CRC/LRC error (Not when using TCP/IP): <input type="checkbox"/> Return exception 06, Busy: <input type="checkbox"/> 	<p>The 'Configuration' table for 'ModbusTCP_any' is shown. It lists various parameters and their values:</p> <table border="1"> <thead> <tr> <th>Name</th> <th>Value</th> </tr> </thead> <tbody> <tr> <td>Plug and play timer</td> <td>10</td> </tr> <tr> <td>Simulation</td> <td></td> </tr> <tr> <td>Simulation device</td> <td></td> </tr> <tr> <td>Channel configuration</td> <td></td> </tr> <tr> <td>Reset outputs on fatal error</td> <td>off</td> </tr> <tr> <td>Block 1</td> <td></td> </tr> <tr> <td>Function code</td> <td>FC3: Read holding register</td> </tr> <tr> <td>Refresh time</td> <td>100</td> </tr> <tr> <td>Block send mode</td> <td>cyclic</td> </tr> <tr> <td>Starting address (read)</td> <td>1000</td> </tr> <tr> <td>Number of items (read)</td> <td>2</td> </tr> <tr> <td>Starting address (write)</td> <td>0</td> </tr> <tr> <td>Number of items (write)</td> <td>1</td> </tr> <tr> <td>Channel 1</td> <td></td> </tr> <tr> <td>Name</td> <td>COM</td> </tr> <tr> <td>Data type</td> <td>UDINT</td> </tr> <tr> <td>Direction</td> <td>Read</td> </tr> </tbody> </table>	Name	Value	Plug and play timer	10	Simulation		Simulation device		Channel configuration		Reset outputs on fatal error	off	Block 1		Function code	FC3: Read holding register	Refresh time	100	Block send mode	cyclic	Starting address (read)	1000	Number of items (read)	2	Starting address (write)	0	Number of items (write)	1	Channel 1		Name	COM	Data type	UDINT	Direction	Read
Name	Value																																				
Plug and play timer	10																																				
Simulation																																					
Simulation device																																					
Channel configuration																																					
Reset outputs on fatal error	off																																				
Block 1																																					
Function code	FC3: Read holding register																																				
Refresh time	100																																				
Block send mode	cyclic																																				
Starting address (read)	1000																																				
Number of items (read)	2																																				
Starting address (write)	0																																				
Number of items (write)	1																																				
Channel 1																																					
Name	COM																																				
Data type	UDINT																																				
Direction	Read																																				

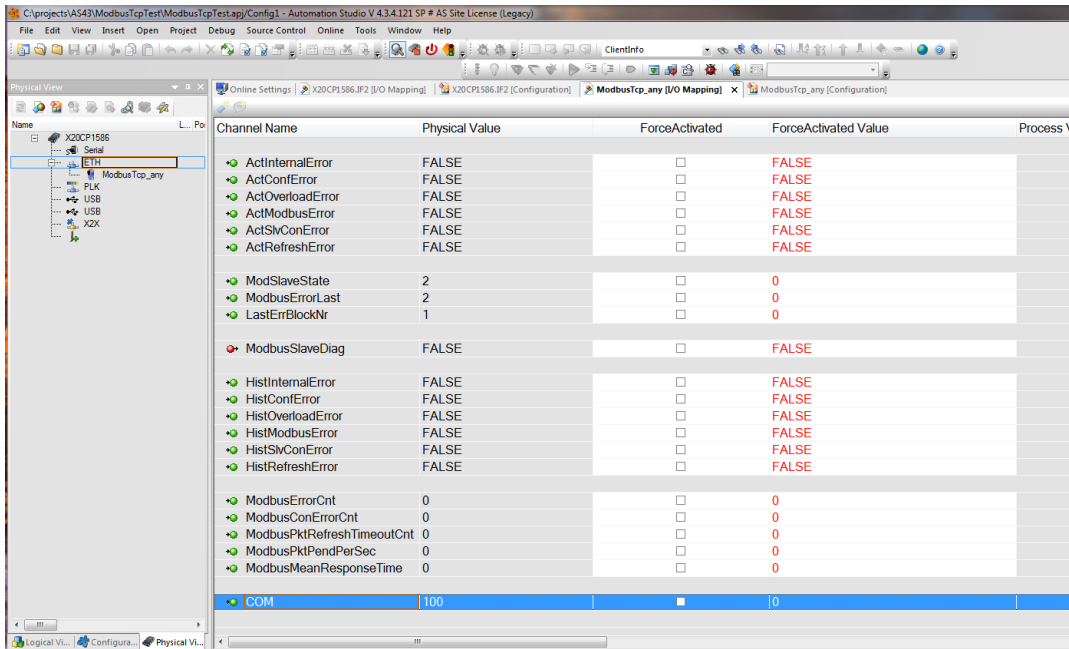
5) 해당 값을 보기 편하기 위하여 포맷 변경하기
두 레지스터 선택 후 마우스 우 클릭, Format / Long AB CD



6) 레지스터를 더블 클릭하여 원하는 값, 100 을 기입하고 OK 누르기



7) Automation Studio / Monitor 모드에서 COM 에 '100'이 들어오는지 확인하기



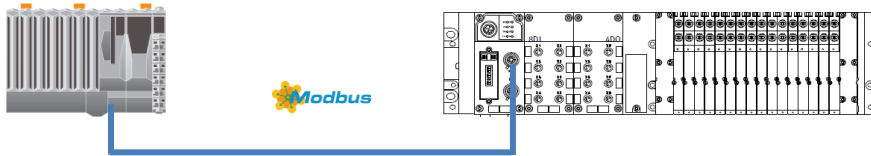
8) 완료

3 예시

3.1 Festo 사 장비와 연결하기

Festo 장비와 B&R PLC 사이의 Modbus TCP 통신 설정을 할 것이다.
이를 통해 B&R PLC 에서 Festo 의 Digital IO 와 MPA 공압 모듈에 연결된 공압을 제어할 수 있다.

하드웨어 토폴로지

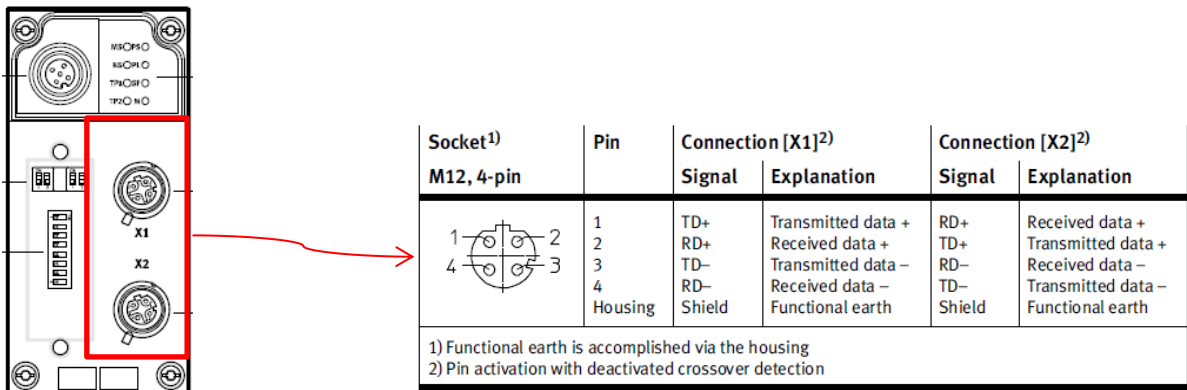


	B&R PLC	Festo PLC
IP Address	192.168.1.150	192.168.1.038
Subnet Mask	255.255.255.0	필요 없음

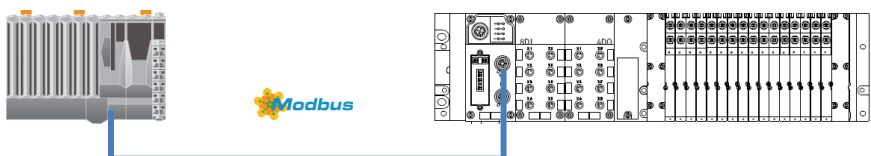
하드웨어 연결방법

하드웨어 연결은 M12 – RJ45 Ethernet Cable 을 통해 B&R PLC 이더넷 포트에 연결됨

- 참조 1

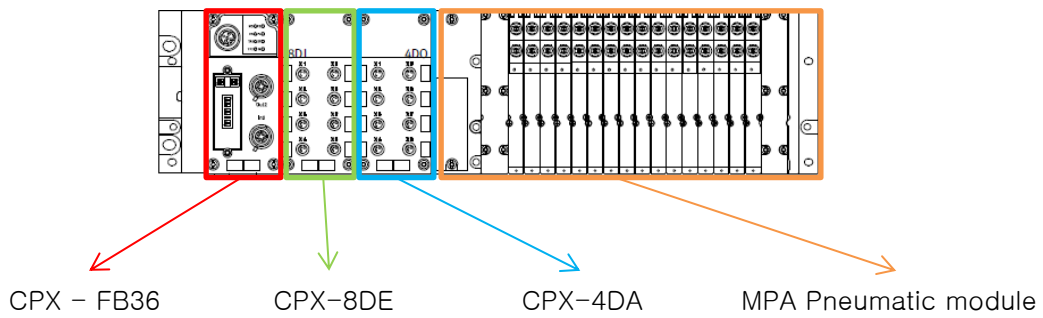


- 참조 2

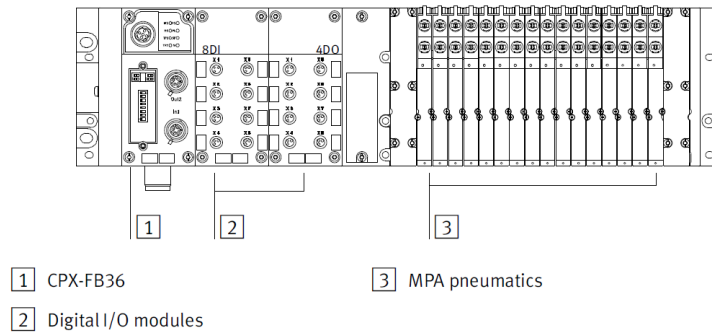


테스트 하드웨어

순번	제조회사	모델명	설명
1	B&R	PPC2100	PLC
2	“	5AP1120.1505-000	판넬(터치 스크린)
3	Festo	CPX-FB36	Bus Node
4	“	CPX-8DE	Digital Input
5	“	CPX-4DA	Digital Output
6	“	MPA Pneumatic module	공압 모듈



CPX terminal with digital I/O modules and MPA pneumatics

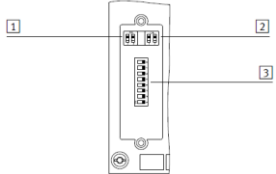











1) Automation Studio 에서 Modbus TCP 로 설정

?
Communication W Fieldbus systems W Modbus TCP

2) Modbus TCP 하드웨어 설정

- 장비에 따라 다른 프로토콜을 제공하므로 해당 장비 매뉴얼을 확인 필수. 본 테스트에서 사용한 CPX-FB36 은 Ethernet/IP, Modbus TCP 를 지원. 따라서 Mode 변경 작업이 필요함.

DIL switches					
	1	Operating Mode and network protocol		Remote I/O	PLC 를 사용할 경우
				Remote Controller	PLC 없이 통합된 장비를 사용할 경우
				Ethernet/IP	-
				Modbus TCP	Modbus TCP 로 설정할 경우
	2	diagnostic mode		I/O Diagnostics interface and status bits switched off	-
				I/O Diagnostics interface is switched on	-
				Status bits switched on	-
				Reserved	-
	3	IP Addressing		$2^7 = 128$ $2^6 = 64$ $2^5 = 32$ $2^4 = 16$ $2^3 = 8$ $2^2 = 4$ $2^1 = 2$ $2^0 = 1$	-

- PLC 를 사용하기 때문에 DIL switches 1 에서 Remote IO 와 Modbus TCP 를 설정
- 어플리케이션에 맞게 Diagnostic 을 사용하고자 하면 위 표를 보고 참조
- IP 어드레스도 물리적으로 설정을 해주어야 함. 여기서 중요한 점은 앞에 세 자리는 192.168.1.X 로 고정되어 있고 마지막 자리의 수만 물리적으로 변경해 주어야함.

예제		
	DIL 3.8: Off	$2^1 + 2^2 + 2^5 = 38$ Set IP Address : 192.168.1.038
	DIL 3.7: Off	
	DIL 3.6: ON $2^5 = 32$	
	DIL 3.5: Off	
	DIL 3.4: Off	
	DIL 3.3: ON $2^2 = 4$	
	DIL 3.2: ON $2^1 = 2$	
	DIL 3.1: Off	

- 여기서 설정한 IP 는 Festo 장비의 IP Address 이며, Automation Studio 에서 연결할 ETH 의 IP Address 도 앞 세 자리는 같고 마지막 자리의 수만 변경해주면 됨.

3)Automation Studio 에서 IP Setting

- 192.168.1.X 는 고정이며, 나머지 자리 수는 개발자가 맞게 변경함.
예시에서 192.168.1.38 을 Modbus TCP 의 Address 로 사용함.

Name	Value
Module1	
General	
Module supervised	off
Ethernet	
Mode	Internet address
IP address	192.168.1.38
Unit identifier	1
TCP port	502
Number of pending requests	1

- PLC / ETH 의 Address 또한 세자리 수는 고정이며, 나머지 자리 수만 변경함.
예시에서 192.168.1.150 을 사용함.

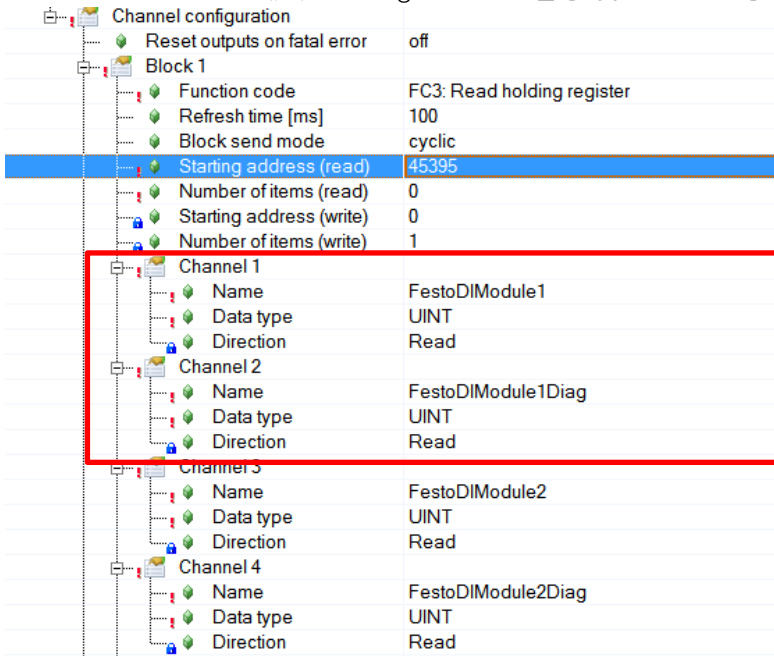
Name	Value
5PPC2100 BY34 000.IF4	
Activate interface	on
Device parameters	
Host name	
Baud rate	auto
Mode	enter IP address manually
IP address	192.168.1.150
Subnet Mask	255.255.255.0

4)Starting Address 찾기

- 사용하려는 Function code 에 따라서 시작하는 Starting Address 가 다르다. 해당 장비의 매뉴얼을 확인해야 한다.
- 아래 표를 기준으로 Digital Input module 의 시작 Address 는 **45395** 이고 다음 어드레스는 Diagnostic 이다. 진단용 어드레스에 다음 Digital Input 을 맵핑하지 않도록 주의한다. Digital Input 의 Input data 는 16 비트중 아래의 8 자리를 사용.
- 아래 Hardware 구성은 예시이다. Hardware 구성은 다양하게 설정할 수 있다. 따라서 Digital Input 을 여러 개 사용할 수 있으며 다음 Digital input 의 Starting address 는 **45397** 이다.

Module	Location	Modbus® address	Input data				
			Bit 15	8	7	4	3
CPX-FB36 Remote I/O	0	45392	Result of access to the I/O diagnostic interface				
		45393	Data from the system table (read access)				
		45394	Diagnostic data				
Digital input module (CPX-8DE)	1	45395	0	Input data			
		45396	Diagnostic data				
Digital output module (CPX-4DA)	2	45397	x				Echo output data
		45398	Diagnostic data				
MPA pneumatic module with diagnostic function	3	45399	x				Echo output data
		45400	Diagnostic data				
MPA pneumatic module with diagnostic function	4	45401	x				Echo output data
		45402	Diagnostic data				
MPA pneumatic module with diagnostic function	5	45403	x				Echo output data
		45404	Diagnostic data				
MPA pneumatic module with diagnostic function	6	45405	x				Echo output data
		45406	Diagnostic data				

5) Automation Studio 에서 Starting Address 설정 및 Channel 형성하기.



- Starting Address 는 앞 단계에서 결정한 **45395** 로 설정
- Channel 에는 데이터 타입과 IO 의 변수 이름을 설정
- 다음 블록을 활성화 시켜 다른 변수를 할당 할 수 있다. (예시: Channel2)

- 예제에서 사용하는 Digital Input module 어드레스의 다음 어드레스는 진단용(Diagnostic) 이므로 사용하지 않더라도 Channel 을 생성하여 Address 를 할당한다.

6) 변수 맵핑

- 변수를 해당 Channel 이름에 연결하면 통신 준비가 완료되었다.
- | | | |
|---|--------------------|---------------------------------------|
| + | FestoDIModule1 | ::gFestoIOMapping.arrDIModuleValue[0] |
| + | FestoDIModule1Diag | |
| + | FestoDIModule2 | ::gFestoIOMapping.arrDIModuleValue[1] |
| + | FestoDIModule2Diag | |

7) 보충

- Digital Output 의 Starting Address 는 40003 이며, 해당 모듈에는 Diagnostic 기능이 없다.

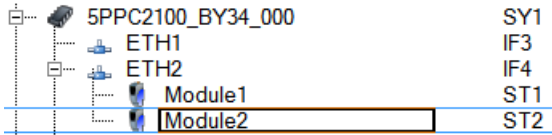
Module	Location	Modbus® address	Output data					
			Bit 15	8	7	4	3	0
CPX-FB36 Remote I/O	0	40001	Result of access to the I/O diagnostic interface					
		40002	Data for the system table (write access)					
Digital output module (CPX-4DA)	2	40003	x					Output data
MPA pneumatic module with diagnostic function	3	40004	x					Output data
MPA pneumatic module with diagnostic function	4	40005	x					Output data
MPA pneumatic module with diagnostic function	5	40006	x					Output data
MPA pneumatic module with diagnostic function	6	40007	x					Output data

- Channel 과 데이터 타입을 설정하고, 변수를 맵핑한다.

Block 2	
Function code	FC16: Write multiple registers
Refresh time [ms]	100
Block send mode	cyclic
Starting address (read)	0
Number of items (read)	1
Starting address (write)	40003
Number of items (write)	0
Channel 1	
Name	FestoDOModule1
Data type	UINT
Direction	Write
Channel 2	
Name	FestoDOModule2
Data type	UINT
Direction	Write
Channel 3	
Name	FestoDOModule3
Data type	UINT
Direction	Write

FestoDOModule1	::gFestoIOMapping.arrDOModuleValue[0]
FestoDOModule2	::gFestoIOMapping.arrDOModuleValue[1]
FestoDOModule3	::gFestoIOMapping.arrDOModuleValue[2]

- Festo Module 이 2 개면, IP Address 는 다르게 설정하지만 Starting Address 는 동일하게 설정 할 수 있다.



- IP 설정시 앞 세 자리는 고정이며, 마지막 자리 수만 변경하면 2 개 모듈을 연결 할 수 있다.

Module1		Module2	
General		General	
Module supervised	off	Module supervised	off
Ethernet		Ethernet	
Mode	Internet address	Mode	Internet address
IP address	192.168.1.38	IP address	192.168.1.101
Unit identifier	1	Unit identifier	0
TCP port	502	TCP port	502
Number of pending requests	1	Number of pending requests	1

- 한 개 Digital Input 모듈 당 8 개 인풋이 있으며, 2 진수로 받을 수 있다.
 예를 들어 0000 0011 이라는 숫자를 Channel 에 맵핑 된 변수를 통해 받으면 변수는 10 진수 3 이 들어온다. 개발자에 따라서 10 진수로 사용할지 2 진수로 사용할지에 추가적인 소프트웨어 핸들링이 필요하다.
- MPA 공압 모듈은 4 개 모듈 카드가 변수 한 개를 공유하여 제어된다. MPA 공압 모듈 1 개는 변수 한 개, Channel 한 개를 이용하여 공압 Output 8 개를 제어 한다.
 예를 들어 MPA 공압 모듈은 모듈 3 개로 구성되어있고, Channel 은 3 개가 필요하며 24 개 변수 제어가 가능하다. 변수를 통해 10 진수 3 을 보내면, 2 진수로 0000 0011 이므로 2 개 아웃풋이 출력된다.

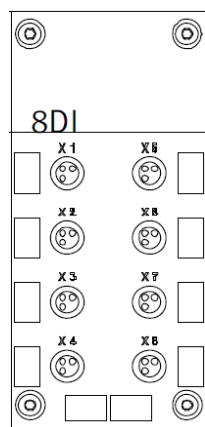


그림 1 Digital Module

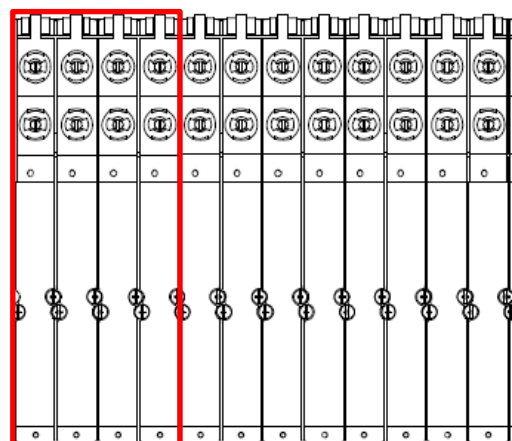


그림 2 MPA 공압 Output Module

4 자료출처

- Festo 사 CPX-FB36 매뉴얼
- Automation Studio help



Communication W Fieldbus systems W Modbus TCP