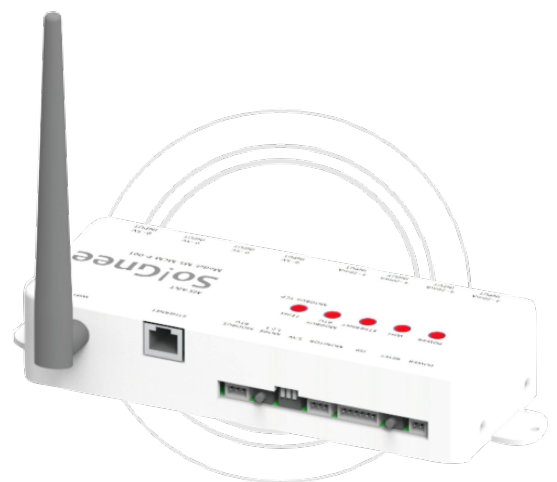
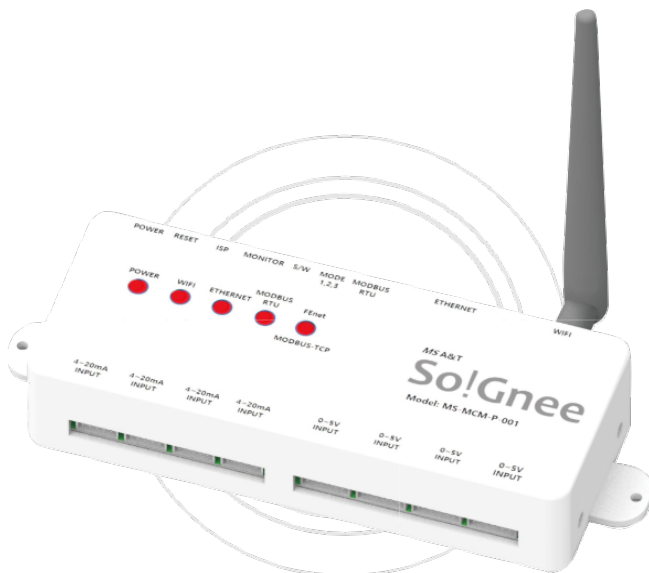


다채널 통신 모듈

- . 진동/온도 모듈 8개 연결
- . 아날로그 16개 입력 채널
- . 다양한 통신 방식 지원
- . 사용자가 통신 방식을 선택할 수 있음
- . 전압 및 전류 타입 센서 사용 가능
- . PLC 및 HMI(SCADA) 통신
- . SSID, PASSWORD, IP , MAC 주소 Monintor를 통해설정

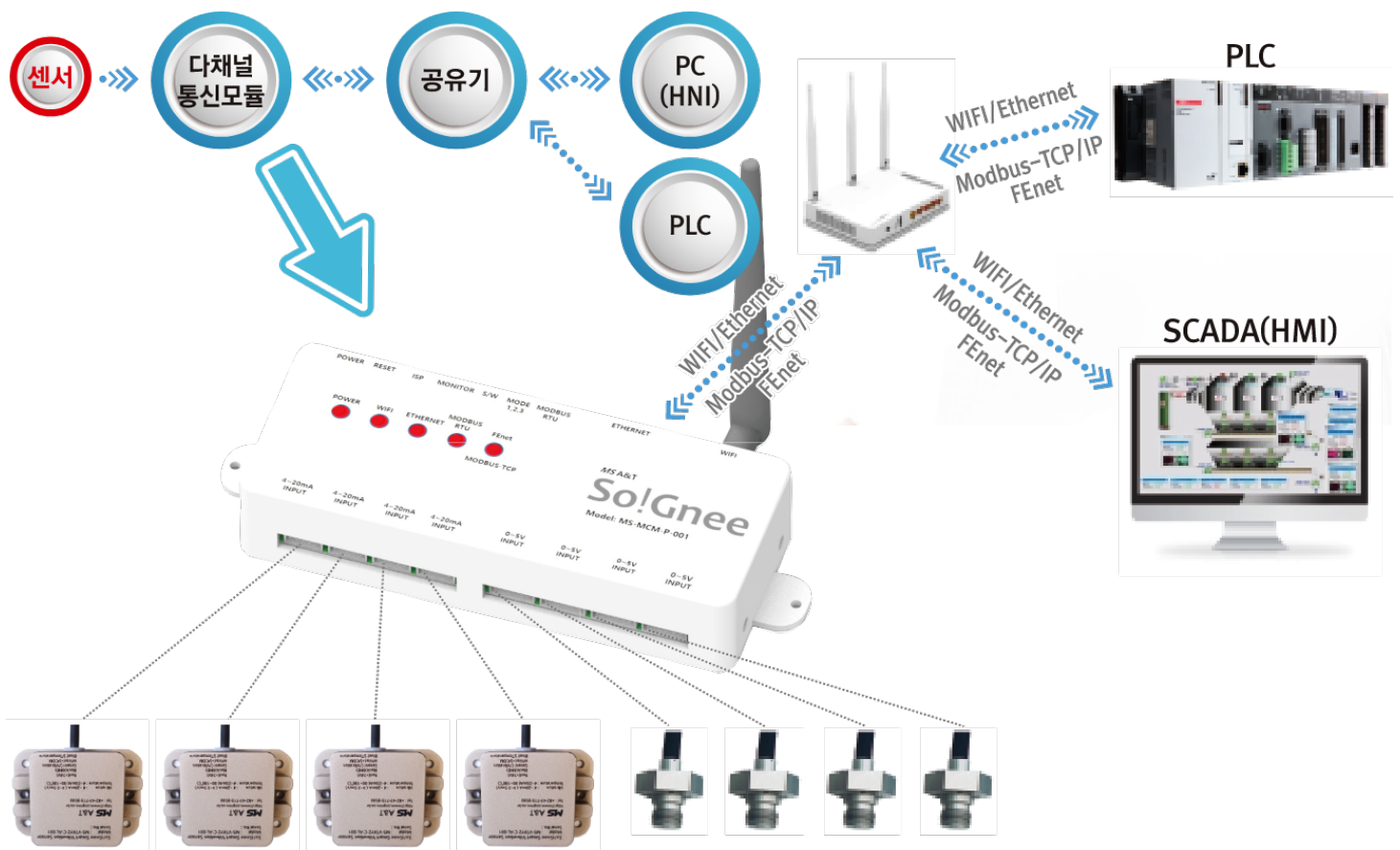
No.	Mode	
	FUNCTION	
	H	L
1	WIFI	Ethernet
2	FEnet	Modbus-TCP/IP
3	WIFI/Ethernet	Modbus-RTU



사양

Item	MS-MCM-P-001	단위
INPUT POWER	DC 20 ~ 26	V
WIFI	802.11b/g/n, 2.4GHz	
Ethernet	10/100 base-T	
Modbus-RTU	19600(RS485통신기반)	BPS
Modbus-TCP	Industrial Network	
FEnet	LS 전용통신(XGT,XGK)	
Analog Voltage Input	0 ~ 5 (8-Channel)	V
Analog Current Input	4 ~ 20 (8-Channel)	mA
Monitor	통신 상태 모니터링 / Setup	V
S/W (Switch)	Setup Mode	
Reset	재부팅/초기화	

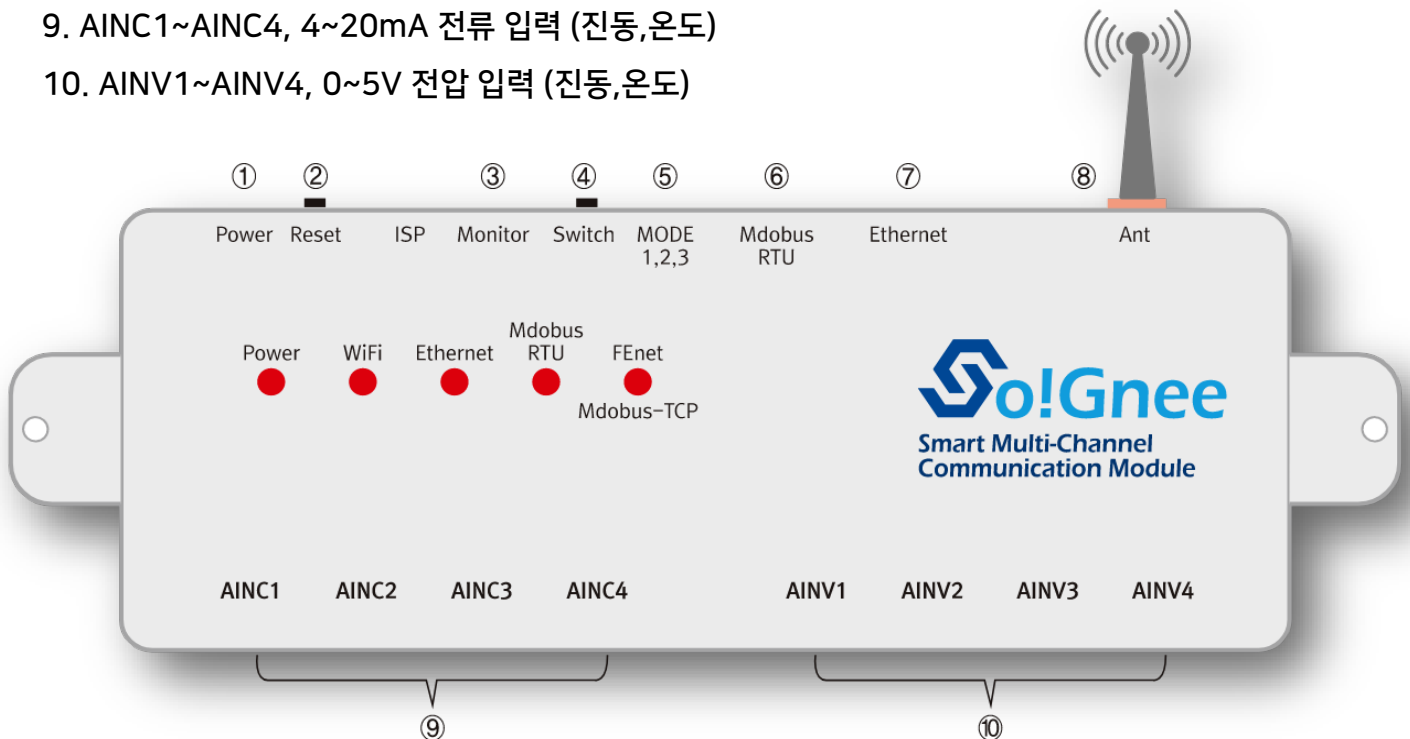
다채널 통신 모듈을 활용한 시스템 구성도



각 포트 별 설명

1. Power : 24V 입력
2. Reset : MODE와 Monitor 셋업 후 리셋 처리
3. Monitor : 셋업 설정, IP와 Mac 주소 등 설정, 통신 출력 상태 모니터링
4. Switch : Monitor 상태에서 설정 모드로의 진입은 'q' 또는 "Switch" 버튼 통해 이루어짐
5. MODE 1,2,3 : MODE 참조
6. Modbus RTU : Modbus RTU 통신 포트 19600 (RS485통신 기반)
7. Ethernet : 이더넷 통신 포트
8. ANT : WiFi용 안테나
9. AINC1~AINC4, 4~20mA 전류 입력 (진동,온도)
10. AINV1~AINV4, 0~5V 전압 입력 (진동,온도)

No.	Mode	
	FUNCTION	
	H	L
1	WIFI	Ethernet
2	FEnet	Modbus-TCP/IP
3	WIFI/Ethernet	Modbus-RTU



시리얼 모니터 하네스 사양 (USB to SERIAL CONVERTOR)

SMH250-03 (연호 커넥터)



③ ② ①

커넥터	하네스 케이블					
	1번	2번	3번			
SMH250-03	GND	RX	TX			

Modbus-RTU 하네스 사양

SMH250-03 (연호 커넥터)



③ ② ①

커넥터	하네스 케이블					
	1번	2번	3번			
SMH250-03	B(-)	A(+)	GND			

기본 명령

명령 형식	인자 형식	설명	명령 예시
all		현재 설정된 모든 설정 값 조회	all
apply		변경된 설정 값을 적용하여 통신 모듈 등을 초기화	apply
q		장치 동작 일시 정지	q
resume		장치 동작 재시작	resume

WiFi 관련 설정 명령

명령 형식	인자 형식	설명	명령 예시
ssid		현재 설정된 WiFi SSID 값 조회	ssid
set ssid <인자>	공백 없는 문자열	WiFi 통신 시 해당 SSID를 사용하도록 설정	set ssid <i>iptime</i>
Pass		현재 설정된 WiFi 암호 값 조회	pass
set pass <인자>	공백 없는 문자열	WiFi 통신 시 해당 암호를 사용하도록 설정	set pass <i>12345678</i>

※ WiFi SSID는 IEEE 802.11 표준에 의거하여, 최대 32 바이트까지 설정을 지원함.

※ WiFi 암호는 IEEE 802.11i 표준의 WPA2-PSK 규격에 의거하여, 최소 8 바이트에서 최대 63 바이트까지 설정을 지원함.

WiFi 및 Ethernet 관련 설정 명령

명령 형식	인자 형식	설명	명령 예시
ip		현재 설정된 Static IP 주소 조회	ip
set ip <인자>	10진 IPv4 주소 (구분자: .)	통신 시 해당 Static IP 주소를 사용하도록 설정	set ip 192.168.1.1
Mac		현재 설정된 MAC 주소 조회	mac
set mac <인자>	16진 MAC 주소 (구분자: -)	통신 시 해당 MAC 주소를 사용하도록 설정	set mac 02-CA-FE-BA-BE-00

Modbus TCP 및 RTU 관련 설정 명령

명령 형식	인자 형식	설명	명령 예시
slave		현재 설정된 Slave ID 조회	slave
set slave <인자>	1바이트 비부호 정수	통신 시 해당 Slave ID를 사용하도록 설정	set slave 123

기능 관련 설정 명령

명령 형식	인자 형식	설명	명령 예시
pmode		현재 설정된 포트 모드 조회	pmode
set pmode <인자>	포트 모드 번호	해당 포트 모드로 동작하도록 설정	set pmode 1
scale		모든 포트에 대하여 현재 설정된 scale up 값 조회	scale
scale <인자>	포트 번호	특정 포트에 대하여 현재 설정된 scale up 값 조회	scale 1

set scale all <인자>	정수 또는 실수	모든 포트가 해당 scale up 값을 사용하도록 설정	set scale all 1 set scale all 1.23
set scale <인자1> <인자2>	<인자1>: 포트 번호 <인자2>: 정수 또는 실수	특정 포트가 해당 scale up 값을 사용하도록 설정	set scale 10 1 set scale 10 1.23

※ ‘scale <인자>’ 및 ‘set scale <인자1> <인자2>’ 명령에서, 포트 번호는 현재 포트 모드에서 활성 상태의 포트 번호이어야 함.

※ 각 포트에 설정된 scale up 값은 해당 포트가 비활성화되더라도 설정 값에 남아 있음. 즉, 해당 포트를 다시 활성화 시 해당 값이 적용됨.

※ ‘set scale all <인자>’ 명령은 오류 방지를 위해, 현재 포트 모드에서 비활성화된 포트를 포함하여 모든 포트의 scale up 값을 해당 값으로 지정함.

포트 모드 종류

포트 모드	활성 포트	활성 포트 수	데이터	샘플링 횟수
0	0~1번	2개	전압 (짝수 번 포트) 온도 알고리즘 1 (홀수 번 포트)	500회
1	0~7번	8개		
2	0~1번	2개	전류 알고리즘 1 (짝수 번 포트) 온도 알고리즘 1 (홀수 번 포트)	500회
3	0~7번	8개		
4	8~15번	8개	진동 (8, 10, 12, 14번 포트) 온도 알고리즘 2 (9, 11, 13, 15번 포트)	10회
5	0~15번	16개	전압 (0, 2, 4, 6번 포트) 온도 알고리즘 1 (1, 3, 5, 7번 포트) 진동 (8, 10, 12, 14번 포트)	500회 (전압, 온도) 10회 (진동, 온도)

			온도 알고리즘 2 (9, 11, 13, 15번 포트)	
6	0~15번	16개	전류 알고리즘 2	500회

오류 메시지 유형

- Too few or many arguments: 인자의 개수가 일치하지 않는 경우
- Invalid argument value: 인자 값의 형식 또는 범위가 잘못된 경우
- Unknown command: 알 수 없는 명령어인 경우
- Port #*n* is not available in this port mode: 현재 포트 모드에서 사용 불가능한 포트 번호를 입력한 경우

XGT FEnet TCP 및 Modbus TCP/RTU 통신 매뉴얼

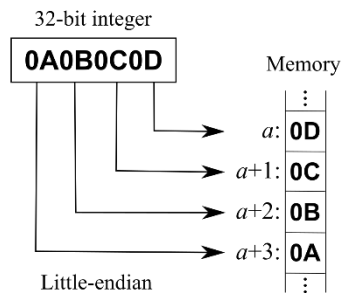
본 문서는 본사에서 개발한 센서 모듈의 XGT FEnet TCP 및 Modbus TCP, Modbus RTU 프로토콜 기반의 네트워크 통신에 대한 것으로서, 각 프로토콜의 데이터 프레임 구조와 해당 통신 과정에 있어서의 센서 모듈의 동작, 그리고 이러한 센서 모듈의 동작을 가상으로 시뮬레이션하기 위한 테스트 도구 소프트웨어의 사용 방법에 대하여 기술한다.

1. 프로토콜 명세

센서 모듈에서는 XGT FEnet TCP 및 Modbus TCP, Modbus RTU 프로토콜의 일부 읽기 요청 명령을 지원하며, 각 프로토콜의 세부 사항은 다음과 같다. 또한, 모든 프로토콜에 대하여 센서에서는 읽기 요청 패킷을 수신한 경우, 해당 요청이 유효한지 검증한 후 측정 데이터를 포함하는 읽기 응답 패킷을 송신한다.

1.1. XGT FEnet TCP

XGT FEnet TCP는 LS ELECTRIC(구 LS산전)에서 FEnet I/F 모듈의 네트워크 통신을 위해 개발한 프로토콜로서, 아래에서는 해당 프로토콜의 데이터 프레임 구조와 각 영역의 데이터에 대한 센서 모듈의 동작에 대하여 기술한다. 본 내용은 당사의 ‘XGT FEnet I/F 모듈 프로토콜 규격 (2005. 3. 30.)’ 등의 문서에 기반한 것으로, 해당 문서는 LS ELECTRIC 홈페이지의 다운로드 자료실에서 확인할 수 있다.



본 프로토콜의 데이터 프레임은 Little-Endian 형태의 byte order를 가지며(이를테면, 2 바이트 데이터 0x1234는 데이터 프레임 상에서 0x3412와 같이 바이트 순서가 역순으로 배열된다), 프레임은 Application Header와 Application Instruction으로 구성된다.

또한, 본 프로토콜은 기본적으로 TCP 2004 포트를 사용하므로, 본 센서와 통신 시 해당 포트를 사용하여 통신하여야 한다.

Application Header

본 영역은 XGT FEnet TCP 데이터 프레임의 앞 부분에 공통적으로 포함되는 것으로서, 프레임 방향, 프레임 순서, 바이트 길이 등 데이터 프레임 자체의 기본적인 정보를 포함한다.

Company ID	Reserved	PLC Info	CPU Info	Source of Frame	Invoke ID	Length	FEnet Position	Reserved (BCC)
8 bytes	2 bytes	2 bytes	1 byte	1 byte	2 bytes	2 bytes	1 byte	1 byte

- **Company ID** (8 bytes): 문자열 “LSIS-XGT”의 ASCII 코드 값을 갖는다. 센서 모듈에서는 읽기 요청 프레임의 본 영역의 값을 통해 유효한 XGT FEnet 프로토콜의 패킷인지의 여부를 확인한다.
- **Reserved** (2 bytes): 예약 영역으로서 0x0000의 값을 갖는다. 센서 모듈에서는 읽기 요청 프레임의 본 영역의 값을 확인하지 않으므로 임의의 값을 포함하여도 된다.
- **PLC Info** (2 bytes): 클라이언트 → 서버의 경우 0x0000의 값을, 서버 → 클라이언트 경우에는 0x0000이 아닌 값을 가지며 CPU Type, Master/Slave, CPU 동작 상태, 시스템 상태 등의 정보를 포함할 수 있다. 센서 모듈에서는 요청 프레임의 본 영역의 값을 확인하지 않으므로 임의의 값을 포함하여도 되며, 센서 모듈 또한 응답 프레임에서 본 영역을 통해 어떠한 정보도 제공하지 않는다.
- **CPU Info** (1 byte): CPU 정보(XGK, XGI, XGR 시리즈)를 나타내기 위한 영역이다. 센서 모듈에서는 요청 프레임의 본 영역의 값을 확인하지 않으므로 임의의 값을 포함하여도 되며, 센서 모듈 또한 응답 프레임에서 본 영역을 통해 어떠한 정보도 제공하지 않는다.
- **Source of Frame** (1 byte): 프레임 방향 값으로서, 클라이언트 → 서버(센서)의 경우 0x33의 값을, 서버(센서) → 클라이언트의 경우에는 0x11의 값을 갖는다. 센서 모듈에서는 요청 프레임에서 본 영역의 값을 확인하여 0x33이 아닌 경우, 잘못된 요청으로 간주하여 응답하지 않을 수 있다.
- **Invoke ID** (2 bytes): 프레임 간의 순서를 구별하기 위해 임의로 지정할 수 있는 ID이다. 센서 모듈에서는 읽기 요청 프레임을 수신하면, 본 영역의 값을 읽기 응답 프레임에도 동일한 값으로 포함하여 송신한다.
- **Length** (2 bytes): Application Instruction의 바이트 크기를 나타낸다. 센서 모듈에서는 요청 프레임의 본 영역의 값이 실제 Application Instruction의 바이트 크기와 일치하지 않을 경우, 유효하지 않은 요청으로 간주하여 응답하지 않을 수 있다.
- **FEnet Position** (1 byte): 상위 4비트와 하위 4비트는 각각 FEnet I/F 모듈의 슬롯 번호와 베이스 번호를 나타낸다. 센서 모듈에서는 요청 프레임의 본 영역의 값을 확인하지 않으므로 임의의 값을 포함하여도 되며, 센서 모듈 또한 응답 프레임에서의 본 영역을 통해 어떠한 정보도 제공하지 않는다.
- **Reserved (BCC)** (1 byte): 예약 영역이며, 이 영역을 제외한 Application Header의 모든 바이트를 합한 값의 Least Significant Byte를 나타내기도 한다. (이를테면, 본 영역을 제외한 Application Header의 바이트 합이 0x1234인 경우, 0x34를 취한다는 것이다.) TCP 자체가 데이터의 무결성을 보장하기에 센서 모듈에서는 요청 프레임의 본 영역의 값을 확인하지 않으므로 임의의 값을 포함하여도 되며, 센서 모듈에서는 응답 프레임의 본 영역에 상기 바이트 합의 LSB의 값을 포함하여 송신한다.

읽기 요구 명령의 Application Instruction 구조

본 영역은 읽기 요청을 위한 XGT FEnet TCP 데이터 프레임에 포함되는 것으로서 Application Header 뒤에 위치하며, 읽기 요청의 대상이 되는 데이터에 대한 정보를 포함한다. 센서의 데이터를 읽기 위해서는, 본 요청을 송신하여야 한다.

명령어	데이터 타입	예약 영역	변수 개수	변수명 길이	변수명	데이터 개수
2 bytes	2 bytes	2 bytes	2 bytes	2 bytes	N bytes	2 bytes

- **명령어** (2 bytes): 읽기 요구(read request) 명령은 0x0054의 값을 갖는다. 센서 모듈에서는 요청 프레임의 본 영역의 값을 확인하며, 읽기 요구 명령만을 지원한다.
- **데이터 타입** (2 bytes): 연속 데이터 타입은 0x0014의 값을 갖는다. 센서 모듈에서는 요청 프레임의 본 영역의 값을 확인하며, 연속 데이터 타입만을 지원한다.
- **예약 영역** (2 bytes): 0x0000의 값을 갖는다. 센서 모듈에서는 요청 프레임의 본 영역의 값을 확인하지 않으므로, 임의의 값을 포함하여도 된다.
- **변수 개수** (2 bytes): 읽기 요구의 대상이 되는 변수의 개수를 나타낸다. 센서 모듈에서는 요청 프레임의 본 영역의 값을 확인하며, 1개의 변수 개수만을 지원한다.
- **변수명 길이** (2 bytes): 읽기 요구의 대상이 되는 변수 이름의 바이트 길이를 나타낸다. 센서 모듈은 이후 영역의 값을 읽기 위해 요청 프레임의 본 영역의 값을 확인하며, 이 값이 변수명 영역의 실제 바이트 길이와 일치하지 않을 경우, 데이터를 잘못 인식하여 응답하지 않을 수도 있다.
- **변수명** (N bytes): 읽기 요구의 대상이 되는 변수 이름의 ASCII 코드 값을 나타내며, 이 영역의 바이트 길이는 '변수명 길이' 영역의 값과 같다. 센서 모듈은 요청 프레임에서 이 영역의 값을 확인하지 않으므로, 임의의 값을 포함하여도 된다.
- **데이터 개수** (2 bytes): 읽기 요구 대상이 되는 데이터의 바이트 길이를 나타낸다. 센서 모듈은 요청 프레임에서 이 영역의 값을 확인하여, 해당 모듈에서 제공하는 데이터의 바이트 길이(센서 개수×각 센서 데이터의 바이트 길이)와 일치하지 않을 경우, 잘못된 요청으로 간주하여 응답하지 않을 수도 있다. 이를테면, 센서에서 8개의 데이터를 제공하는 경우, 각 데이터는 부호 있는(signed) 16비트(2 바이트) 정수이므로 16 바이트를 요청하여야 한다.

읽기 요구 명령의 샘플 데이터 프레임

원문 (16진수)	※ 회색 부분이 Application Header
4C 53 49 53 2D 58 47 54 00 00 00 00 A0 33 00 00 12 00 00 40 54 00 14 00 00 00 01 00 06 00 25 44 57 35 30 30 10 00	

원문 (16진수)	영역	의미
Application Header		
4C 53 49 53 2D 58 47 54	Company ID	“LSIS-XGT”에 대한 ASCII 코드
00 00	Reserved	
00 00	PLC Info	
A0	CPU Info	
33	Source of Frame	클라이언트 → 서버(센서) 방향
00 00	Invoke ID	0번째 프레임
12 00	Length	Application Instruction의 길이가 18(=0x12) 바이트
00	FEnet Position	
40	Reserved (BCC)	바이트 합의 Least Significant Byte
Application Instruction		
54 00	명령어	읽기 요구 (read request)
14 00	데이터 타입	연속 데이터
00 00	예약 영역	
01 00	변수 개수	요청하는 변수가 1개
06 00	변수명 길이	요청하는 변수 이름이 6 바이트
25 44 57 35 30 30	변수명	“%DW500”에 대한 ASCII 코드
10 00	데이터 개수	16 (=0x10) 바이트의 데이터 요청

읽기 응답의 Application Instruction 구조

본 영역은 읽기 응답에 해당하는 XGT FEnet TCP 데이터 프레임에 포함되는 것으로서 Application Header 뒤에 위치하며, 읽기 결과에 대한 정보를 포함한다. 센서에서는 읽기 요청에 대한 응답으로서 본 영역을 포함하는 데이터 프레임을 송신한다.

명령어	데이터 타입	예약 영역	에러 상태	변수 개수	데이터 크기	데이터
2 bytes	2 bytes	2 bytes	2 bytes	2 bytes	2 bytes	<i>N</i> bytes

- **명령어** (2 bytes): 읽기 응답(read response)은 0x0055의 값을 갖는다.
- **데이터 타입** (2 bytes): 연속 데이터 타입이므로 0x0014의 값을 갖는다.
- **예약 영역** (2 bytes): 0x0000의 값을 갖는다.
- **에러 상태** (2 bytes): 정상 응답의 경우 0x0000의 값을, 오류의 경우 0x0000이 아닌 값을 갖는다.
- **변수 개수** (2 bytes): 1개의 변수이므로 0x0001의 값을 갖는다.
- **데이터 크기** (2 bytes): 읽기 결과 데이터의 바이트 길이를 나타낸다.
- **데이터** (*N* bytes): 읽기 결과 데이터이다. 센서에서는 각 데이터를 포트 번호가 작은 순서대로 배열하며, 각 데이터는 부호 있는 16비트(2 바이트) 정수이다.

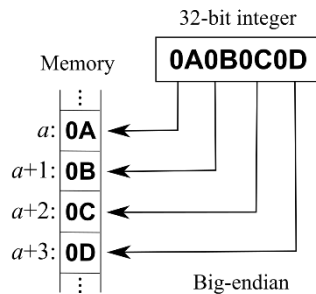
읽기 응답의 샘플 데이터 프레임

원문 (16진수)	※ 회색 부분이 Application Header
4C 53 49 53 2D 58 47 54 00 00 00 00 A0 11 00 00 1C 00 00 28 55 00 14 00 00 00 00 01 00 10 00 57 04 AE 08 05 0D 5C 11 B3 15 0A 1A 61 1E B8 22	

원문 (16진수)	영역	의미
Application Header		
4C 53 49 53 2D 58 47 54	Company ID	“LSIS-XGT”에 대한 ASCII 코드
00 00	Reserved	
00 00	PLC Info	
A0	CPU Info	
11	Source of Frame	서버(센서) → 클라이언트 방향
00 00	Invoke ID	0번째 프레임
1C 00	Length	Application Instruction의 길이가 28(=0x1C) 바이트
00	FENet Position	
28	Reserved (BCC)	바이트 합의 Least Significant Byte
Application Instruction		
55 00	명령어	읽기 응답 (read response)
14 00	데이터 타입	연속 데이터
00 00	예약 영역	
00 00	에러 상태	정상
01 00	변수 개수	읽은 변수가 1개
10 00	데이터 개수	16 (=0x10) 바이트 데이터 읽음
57 04	0번 포트의 센서 데이터	측정 값: 1,111 (=0x0457)
AE 08	1번 포트의 센서 데이터	측정 값: 2,222 (=0x08AE)
05 0D	2번 포트의 센서 데이터	측정 값: 3,333 (=0x0D05)
5C 11	3번 포트의 센서 데이터	측정 값: 4,444 (=0x115C)
B3 15	4번 포트의 센서 데이터	측정 값: 5,555 (=0x15B3)
0A 1A	5번 포트의 센서 데이터	측정 값: 6,666 (=0x1A0A)
61 1E	6번 포트의 센서 데이터	측정 값: 7,777 (=0x1E61)
B8 22	7번 포트의 센서 데이터	측정 값: 8,888 (=0x22B8)

1.2. Modbus TCP

Modbus 프로토콜은 1979년 Modicon (현 Schneider Electric)사에서 PLC 통신을 위해 개발된 것으로서, 산업용 전자 장비 간의 통신을 위해 사실상 표준(de facto standard)으로 널리 사용된다. Modbus TCP는 TCP/IP 환경에서 활용하기 위한 것으로 Modbus 프로토콜의 일종이다.



본 프로토콜의 데이터 프레임은 Big-Endian 형태의 byte order를 가지며(데이터 프레임 상에서 바이트 순서가 그대로 배열된다), 프레임은 헤더와 데이터 영역으로 구성된다.

또한, 본 프로토콜은 기본적으로 TCP 502 포트를 사용하므로, 본 센서와 통신 시 해당 포트를 사용하여 통신하여야 한다.

Modbus TCP 헤더

본 영역은 Modbus TCP 데이터 프레임의 앞 부분에 공통적으로 포함되는 것으로서, 프레임 순서, 바이트 길이, Slave 번호 등의 기본적인 정보를 포함한다.

Transaction identifier	Protocol identifier	Length field	Unit identifier
2 bytes	2 bytes	2 bytes	1 byte

- **Transaction identifier** (2 bytes): XGT FEnet TCP의 Invoke ID와 동일한 것으로서, 프레임 간의 순서를 구별하기 위해 임의로 지정할 수 있는 ID이다. 센서 모듈에서는 읽기 요청 프레임을 수신하면, 본 영역의 값을 읽기 응답 프레임에도 동일한 값으로 포함하여 송신한다.
- **Protocol identifier** (2 bytes): 프로토콜 식별자로서 Modbus TCP는 0x0000의 값을 갖는다. 센서 모듈에서는 읽기 요청 프레임의 본 영역의 값을 통해 유효한 Modbus TCP 프로토콜의 패킷인지의 여부를 확인한다.
- **Length field** (2 bytes): 이어지는 데이터 프레임(헤더 및 데이터 영역 포함)의 바이트 크기를 나타낸다. 센서 모듈에서는 요청 프레임의 본 영역의 값이 실제 이어지는 데이터 프레임의 바이트 크기와 일치하지 않을 경우, 유효하지 않은 요청으로 간주하여 응답하지 않을 수 있다.
- **Unit identifier** (1 byte): Slave (장치) 번호를 나타낸다. 센서 모듈은 요청 프레임에서 이 영역의 값을 확인하지 않으므로, 임의의 값을 포함하여도 된다.

읽기(Read Registers) 요청의 데이터 영역 구조

Function code	Starting address	Quantity of registers
1 byte	2 bytes	2 bytes

- **Function code** (1 byte): Read Holding Registers의 경우 0x03, Read Input Registers의 경우 0x04의 값을 갖는다. 센서 모듈에서는 요청 프레임의 본 영역의 값을 확인하며, Read Holding Register 및 Read Input Register 두 경우 모두 동일하게 동작한다.
- **Starting address** (2 bytes): 읽기 요청의 대상이 되는 주소이다. 유효한 주소 범위는 0x0000 ~ 0xFFFF이나, 센서 모듈은 요청 프레임에서 이 영역의 값을 확인하지 않으므로, 임의의 값을 포함하여도 된다.
- **Quantity of registers** (2 bytes): 읽기 요청의 대상이 되는 Register (2 바이트)의 수이다.

읽기 요청의 샘플 데이터 프레임

원문 (16진수)	※ 회색 부분이 헤더 영역
00 00 00 00 00 06 12 04 34 56 00 08	

원문 (16진수)	영역	의미
헤더 영역		
00 00	Transaction identifier	0번째 프레임
00 00	Protocol identifier	Modbus TCP
00 06	Length field	이어지는 프레임이 6 바이트
12	Unit identifier	18(=0x12)번 Slave 장치
데이터 영역		
04	Function code	Read Input Register
34 56	Starting address	0x3456 주소의 데이터 요청
00 08	Quantity of registers	8개의 Register 데이터 요청

읽기(Read Registers) 응답의 데이터 영역 구조

Function code	Byte count	Register values
1 byte	1 byte	2 bytes

- **Function code** (1 byte): Read Holding Registers의 경우 0x03, Read Input Registers의 경우 0x04의 값을 갖는다.
- **Byte count** (1 byte): 읽기 결과 데이터의 바이트 길이를 나타낸다.
- **Register values** (2 bytes): 읽기 결과 데이터이다. 센서에서는 각 데이터를 포트 번호가 작은 순서대로 배열하며, 각 데이터는 부호 있는 16비트(2 바이트) 정수이다.

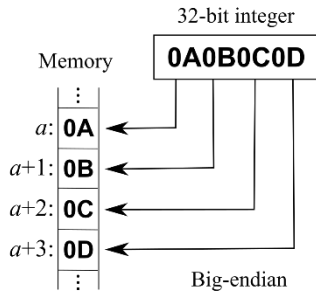
읽기 응답의 샘플 데이터 프레임

원문 (16진수)	※ 회색 부분이 헤더 영역
00 00 00 00 00 13 12 04 10 57 04 AE 08 05 0D 5C 11 B3 15 0A 1A 61 1E B8 22	

원문 (16진수)	영역	의미
헤더 영역		
00 00	Transaction identifier	0번째 프레임
00 00	Protocol identifier	Modbus TCP
00 13	Length field	이어지는 프레임이 19 바이트
12	Unit identifier	18(=0x12)번 Slave 장치
데이터 영역		
04	Function code	Read Input Register
10	Byte count	16 (=0x10) 바이트 데이터 읽음
57 04	0번 포트의 센서 데이터	측정 값: 1,111 (=0x0457)
AE 08	1번 포트의 센서 데이터	측정 값: 2,222 (=0x08AE)
05 0D	2번 포트의 센서 데이터	측정 값: 3,333 (=0x0D05)
5C 11	3번 포트의 센서 데이터	측정 값: 4,444 (=0x115C)
B3 15	4번 포트의 센서 데이터	측정 값: 5,555 (=0x15B3)
0A 1A	5번 포트의 센서 데이터	측정 값: 6,666 (=0x1A0A)
61 1E	6번 포트의 센서 데이터	측정 값: 7,777 (=0x1E61)
B8 22	7번 포트의 센서 데이터	측정 값: 8,888 (=0x22B8)

1.2. Modbus RTU

Modbus RTU는 시리얼 환경에서 활용하기 위한 Modbus 프로토콜의 일종이다.



본 프로토콜의 데이터 프레임은 Big-Endian 형태의 byte order를 가지며(데이터 프레임 상에서 바이트 순서가 그대로 배열된다), 프레임은 헤더와 데이터 영역으로 구성된다.

Modbus RTU 헤더

본 영역은 Modbus RTU 데이터 프레임의 앞 부분에 공통적으로 포함되는 것으로서, 타 프로토콜과는 달리 단순히 Address 정보만을 포함한다.

Address
1 byte

- **Address** (1 byte): Modbus TCP의 Unit address와 유사한 것으로, Station (장치) 번호를 나타낸다. 센서 모듈은 요청 프레임에서 이 영역의 값을 확인하지 않으므로, 임의의 값을 포함하여도 된다.

읽기(Read Registers) 요청의 데이터 구조

Function code	Starting address	Quantity of registers	CRC
1 byte	2 bytes	2 bytes	2 bytes

- **Function code** (1 byte): Read Holding Registers의 경우 0x03, Read Input Registers의 경우 0x04의 값을 갖는다. 센서 모듈에서는 요청 프레임의 본 영역의 값을 확인하며, Read Holding Register 및 Read Input Register 두 경우 모두 동일하게 동작한다.
- **Starting address** (2 bytes): 읽기 요청의 대상이 되는 주소이다. 유효한 주소 범위는 0x0000~0xFFFF이나, 센서 모듈은 요청 프레임에서 이 영역의 값을 확인하지 않으므로, 임의의 값을 포함하여도 된다.
- **Quantity of registers** (2 bytes): 읽기 요청의 대상이 되는 Register (2 바이트)의 수이다.
- **CRC** (2 bytes): CRC (cyclic redundancy check) 체크 값이다.

읽기 요청의 샘플 데이터 프레임

원문 (16진수)	※ 회색 부분이 헤더 영역
11 03 00 6B 00 03 76 87	

원문 (16진수)	영역	의미
헤더 영역		
11	Address	11(=0x17)번 Station 장치
데이터 영역		
03	Function code	Read Holding Register
00 6B	Starting address	0x006B 주소의 데이터 요청
00 03	Quantity of registers	3개의 Register 데이터 요청
76 87	CRC	CRC 체크 값

읽기(Read Registers) 응답의 데이터 구조

Function code	Byte count	Register values	CRC
1 byte	1 byte	2 bytes	2 bytes

- **Function code** (1 byte): Read Holding Registers의 경우 0x03, Read Input Registers의 경우 0x04의 값을 갖는다.
- **Byte count** (1 byte): 읽기 결과 데이터의 바이트 길이를 나타낸다.
- **Register values** (2 bytes): 읽기 결과 데이터이다. 센서에서는 각 데이터를 포트 번호가 작은 순서대로 배열하며, 각 데이터는 부호 있는 16비트(2 바이트) 정수이다.
- **CRC** (2 bytes): CRC (cyclic redundancy check) 체크 값이다.

읽기 응답의 샘플 데이터 프레임

원문 (16진수)	※ 회색 부분이 헤더 영역
11 03 06 AE 41 56 52 43 40 49 AD	

원문 (16진수)	영역	의미
헤더 영역		
11	Address	11(=0x17)번 Station 장치
데이터 영역		
03	Function code	Read Holding Register
06	Byte count	6 (=0x6) 바이트 데이터 읽음
AE 41	0번 포트의 센서 데이터	측정 값: 44,609 (=0xAE41)
56 52	1번 포트의 센서 데이터	측정 값: 22,098 (=0x5652)
43 40	2번 포트의 센서 데이터	측정 값: 17,216 (=0x4340)
49 AD	CRC	CRC 체크 값