

MPINO SERIES MPINO-16A8R

사용 설명서

저희 (주)아이로직스 제품을 구입해 주셔서 감사합니다.



사용 전에 안전을 위한 주의사항을 반드시 읽고 사용하십시오.

□ 안전을 위한 주의사항

- ※ '안전을 위한 주의사항'은 제품을 안전하고 올바르게 사용하여 사고나 위험을 미리 막기 위한 것이므로 반드시 지켜야 합니다.
 - ※ 주의사항은 '경고'와 '주의' 두 가지로 구분되어 있으며, '경고'와 '주의'의 의미는 다음과 같습니다.
- 지시사항을 위반하였을 때.
- ⚠ **경고** 심각한 상해나 사망이 발생할 가능성이 있는 경우
 - ⚠ **주의** 경미한 상해나 제품 손상이 발생할 가능성이 있는 경우
- ※ 제품과 취급설명서에 표시된 그림기호의 의미는 다음과 같습니다.
- ⚠는 특정조건 하에서 위험이 발생할 우려가 있으므로 주의하라는 기호입니다.

⚠ 경고

- 인명이나 재산상에 영향이 큰 기기(예: 원자력 제어장치, 의료기기, 선박, 차량, 철도, 항공기, 연소장치, 안전장치, 방범/방재장치 등)에 사용할 경우에는 반드시 2중으로 안전장치를 부착한 후 사용해야 합니다. 화재, 인사사고, 재산상의 막대한 손실이 발생할 수 있습니다.
- 자사 수리 기술자 이외에는 제품을 개조하지 마십시오. 감전이나 화재의 우려가 있습니다.

⚠ 주의

- 실외에서 사용하지 마십시오. 제품의 수명이 짧아지는 원인이 되며 감전의 우려가 있습니다. 본 제품은 실내 환경에 적합하도록 제작되었습니다. 실내가 아닌 외부환경으로부터 영향을 받을 수 있는 장소에서 사용할 수 없습니다. (예 : 비, 황사, 먼지, 서리, 햇빛, 결로 등)
- 인화성, 폭발성 가스 환경에서 사용하지 마십시오. 화재 및 폭발의 우려가 있습니다.
- 사용 전압 범위를 초과하여 사용하지 마십시오. 제품이 파손될 수 있습니다.
- 전원의 극성 등 오배선을 하지 마십시오. 제품이 파손될 수 있습니다.
- 진동이나 충격이 많은 곳에서 사용하지 마십시오. 제품이 파손될 수 있습니다.
- 청소 시 물, 유기 용제를 사용하지 마십시오. 감전 및 화재의 우려가 있습니다.

□ 손해배상책임

(주)아이로직스는 제품을 사용하다 발생하는 인적, 물적 자원에 대해 책임을 지지 않습니다. 충분한 테스트와 안전장치를 사용하여 주시기 바랍니다.

□ 사양서

구분	개수	접점명	특징
전원	-	전원전압	<ul style="list-style-type: none"> DC 9V ~ 12V (12V 1A 이상)
디지털 입력	16 포인트 (절연)	I(22)~ I(37) (8P/1COM)	<ul style="list-style-type: none"> 오퍼레이팅 입력 전압 : DC 0 ~ 40V HIGH 인식 전압 : DC 5V 이상 입력저항 : 2.2kΩ 8POINT / 1COM
릴레이 출력	8 포인트 (절연)	R(39)~ R(46) (4P/1COM)	<ul style="list-style-type: none"> 오퍼레이팅 출력 전압 - 0~30V D.C, 0~250V A.C 최대 출력 허용전류 : 5A / 포인트 15A / COM R(39)~R(42) ↔ COM2 R(43)~R(46) ↔ COM3
아날로그 입력	4 포인트 (비절연)	A(0) ~ A(3)	<ul style="list-style-type: none"> 오퍼레이팅 입력 전압 : 0 ~ 5V D.C Optional : 0 ~ 10V D.C Optional : 0(4) ~ 20mA Optional : NTC.10kΩ(25℃) 온도센서 분해능 : 10Bit (0~1023) 입력저항 : 100MΩ (0 ~ 5V) 입력저항 : 200MΩ (0 ~ 10V) 입력저항 : 250Ω(1%) (0(4) ~ 20mA) 풀업저항 : 10kΩ(1%) NTC 온도센서
인터럽트 (펄스입력) (엔코더)	1 채널 <비절연>	SDA (20), SCL (21)	<ul style="list-style-type: none"> 오퍼레이팅 전압 : DC 0 ~ 5V HIGH 인식 전압 : GND (LOW) 풀업 저항 : 4.7kΩ 엔코더 사용시 I2C통신은 사용불가
통신 채널	1 채널 (비절연)	I2C	I2C 1채널 지원
	1 채널 (비절연)	RS232	<ul style="list-style-type: none"> Serial1 Modbus RTU Master / Slave 지원
	1 채널 (비절연)	RS485	<ul style="list-style-type: none"> Serial2 Modbus RTU Master / Slave 지원
	1 채널 (비절연)	UART	<ul style="list-style-type: none"> Serial3 Modbus RTU Master / Slave 지원

□ 메모리 사양서

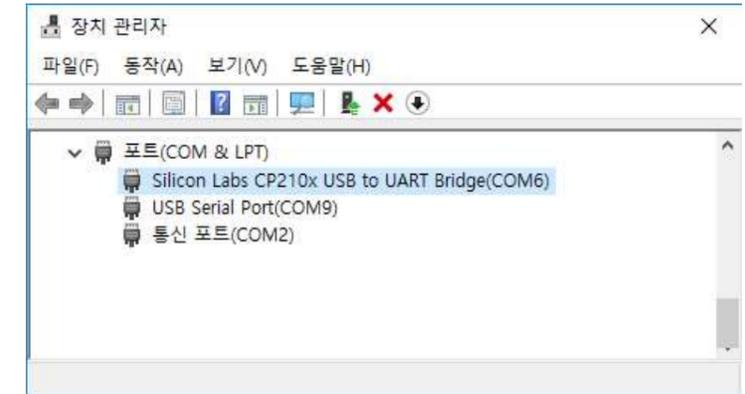
- 256Kbyte Flash Memory (8K byte BootLoader Memory)
- 4Kbyte SRAM Data Memory
- 4Kbyte EEPROM Memory

□ 프로그램 코딩 및 다운로드 소프트웨어

- 아이로직스 블로그에서 아두이노 IDE 소프트웨어를 다운로드 받고 설치합니다. (<https://www.arduino.cc/en/software/>)
- 컴퓨터의 USB포트와 제품(MPINO-16A8R)에 다운로드 포트를 연결합니다.



- 아이로직스 자료실에서 Silicon Labs CP210x Usb Driver를 다운로드를 받아 설치합니다. (<https://www.ilogics.co.kr/article/자료실/7/18/>)
- 윈도우의 장치관리자에서 COM포트를 확인합니다. (PC와 MPS-16A8R 제품이 USB로 연결되어 있어야 합니다)

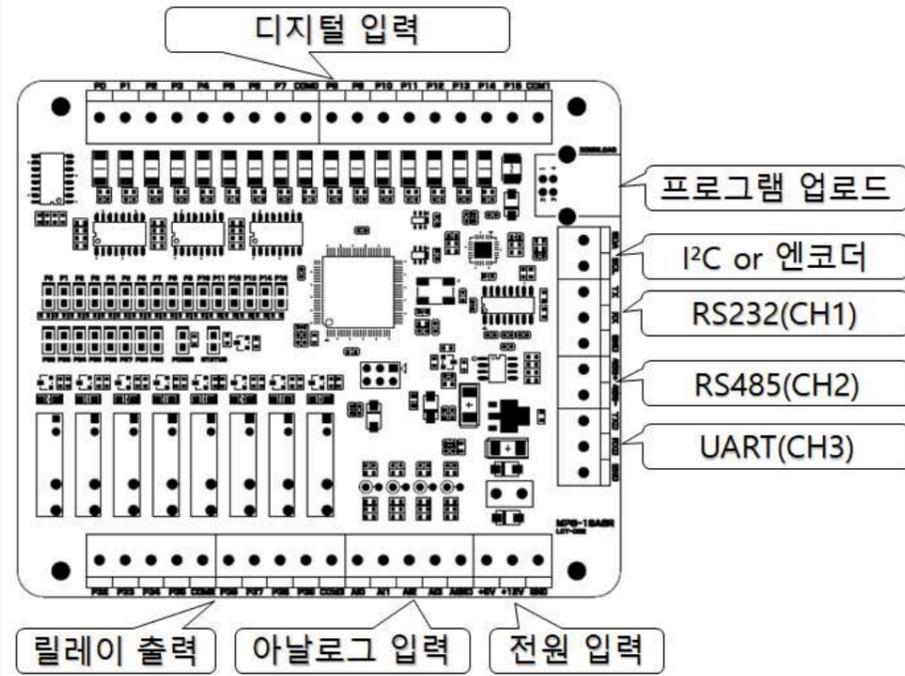


- Arduino IDE에서 "Arduino Mega or Mega 2560"을 선택합니다. (Tools -> Board -> Arduino AVR Boards)
- Arduino IDE에서 장치관리자에서 확인한 COM포트를 선택합니다. (Tools -> Port)
- Arduino IDE에서 프로그래머를 AVR ISP로 선택합니다. (Tools -> Programmer)
- 프로그래밍을 하고, 업로드를 합니다.
- ※ 아래 링크에서 "필수설치파일"을 다운로드 받아 설치하시면 보다 편리하게 사용이 가능합니다. (<https://www.ilogics.co.kr/article/자료실/7/243/>)
- ※ 통합라이브러리를 설치하시면 Arduino IDE에서 MPINO-16A8R을 선택하여 사용이 가능합니다. (Tools -> Board -> ILOGICS)

□ 명령어 설명서

- Arduino IDE에서 Help -> Reference 를 실행하거나 다음 링크에서 확인할 수 있습니다. (<https://www.arduino.cc/reference/ko/>)
- EEPROM과 I2C(Wire) 등의 라이브러리는 아래 링크에서 확인할 수 있습니다. (<https://blog.naver.com/ilogics/223802536404>)

□ 기능별 위치



□ 전원

- 전원입력은 9V ~ 12V D.C를 사용할 수 있습니다. 9V ~ 12V D.C는 LM1117 DC-DC Regulator를 통하여 5V D.C로 전환되어 내부회로에 전원을 공급합니다. 전원의 효율은 12V일 때보다 9V일 때가 좋습니다.
- 다운로드 포트에 USB 연결선으로 컴퓨터와 연결하면, 컴퓨터의 5V 전원을 사용하여 제품이 동작됩니다. 이때, 컴퓨터의 5V 전원을 아날로그 입력의 기준전압으로 사용하게 됩니다. 때문에, 아날로그의 값이 많이 흔들릴 수 있습니다. 이러한 현상은 9V ~ 12V D.C를 사용하여 정상적으로 전원을 주게 되면 많이 개선되어 집니다. 전원을 동시에 투입해도 괜찮습니다.
- 전원입력포트의 +5V 단자는 전원입력 또는 전원출력 공통으로 사용할 수 있습니다. 단, 전원출력 사용할 경우, 최대 100mA 이하로 사용해 주시기 바랍니다.

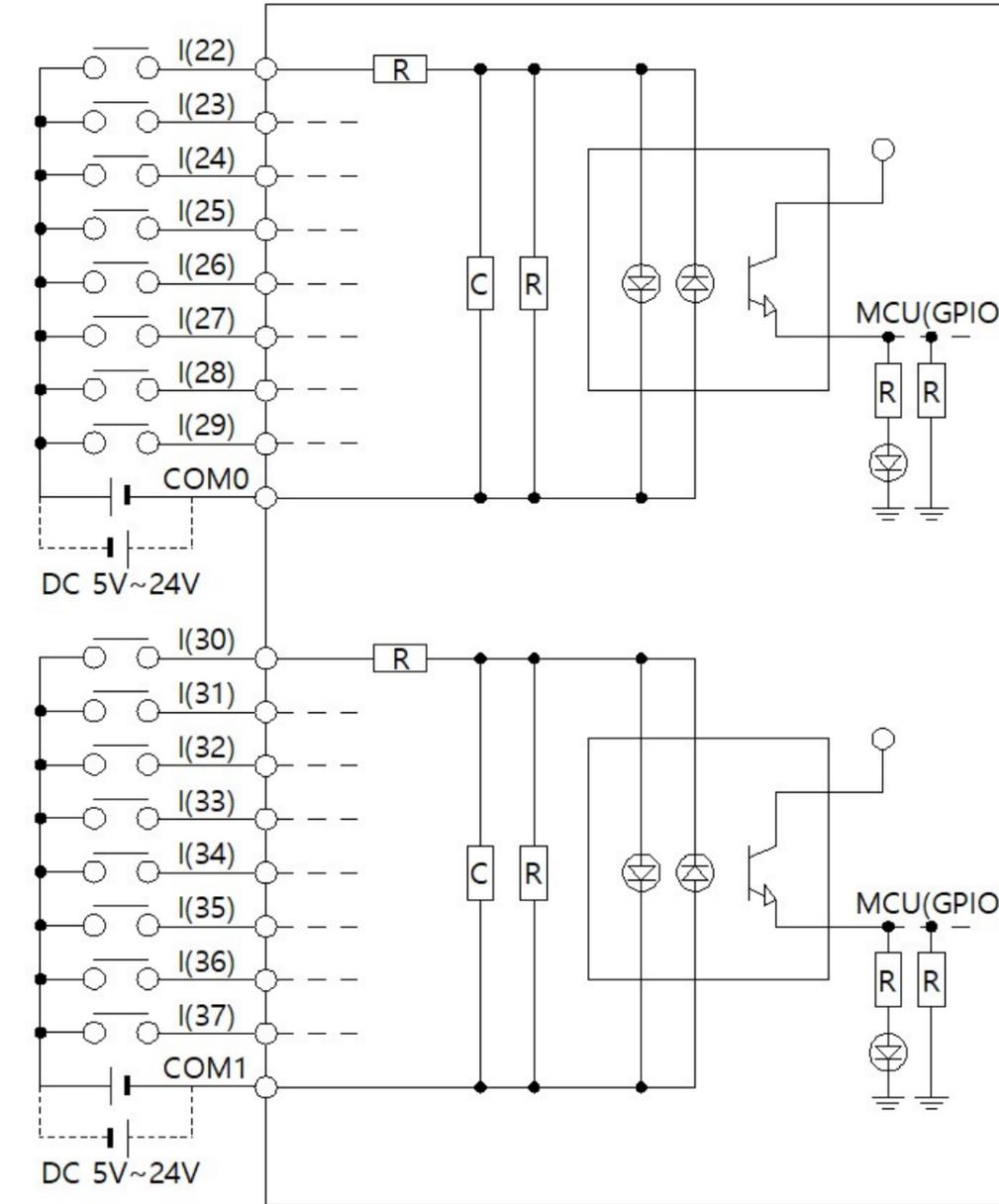
□ 정전유지

- MPINO-8A4R-S 제품은 DC 5V 전원으로 모든 동작이 가능하도록 설계되어 있습니다. DC +5V 단자 대에 배터리를 연결하여, 정전 시에는 배터리의 전원으로 절체 되도록 하여 정전유지가 가능합니다.
- 비휘발성 메모리인 EEPROM을 이용하여 메모리를 보존할 수 있습니다. 단, EEPROM은 100,000번 이상 기록(Write)을 할 경우, 해당 색션의 불량 발생 할 수 있으므로 빠른 속도로 변경되는 데이터를 기록하는 것은 올바르지 않습니다. 상시로 변하는 값을 EEPROM을 통하여 정전유지를 하고자 한다면 아래 블로그에서 방법을 설명하고 있으니 참고해주시기 바랍니다.

(<https://blog.naver.com/ilogics/223805063172>)

□ 디지털 입력

I(22) ~ I(29)에 DC 5V ~ 24V D.C가 스위치, 센서 등에 의해 입력된다면, COM0은 GND를 연결해야 합니다. 반대로, I(22) ~ I(29)에 GND가 스위치, 센서 등에 의해 입력된다면, COM0은 DC 5V ~ 24V D.C를 연결해야 합니다.



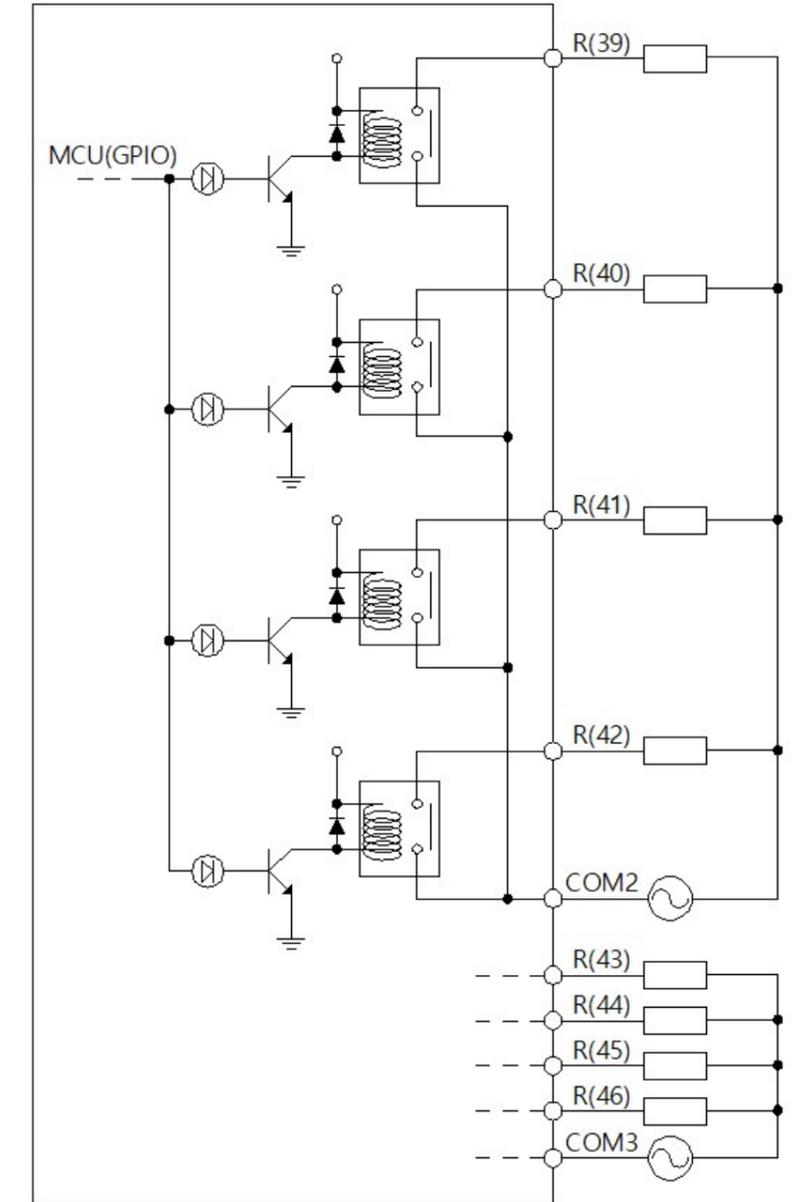
□ 디지털 입력 및 릴레이 출력 프로그램

디지털입력 I(22)가 HIGH 이면, 릴레이출력 R(39)을 ON 시키고, 디지털입력 I(22)가 LOW 이면, 릴레이출력 R(39)을 OFF 시키는 프로그램입니다.

```
void setup() {
    pinMode(39, OUTPUT); // R(39)을 출력모드로 설정합니다.
}
void loop() {
    // I(22)가 HIGH 이면, R(39)을 ON 시킵니다.
    if (digitalRead(22) == 1) { digitalWrite(39, HIGH); }
    // I(22)가 HIGH가 아니면, 즉 LOW 이면, R(39)을 OFF 시킵니다.
    else { digitalWrite(39, LOW); }
}
```

□ 릴레이 출력

COM2에 연결한 전원이 R(39) ~ R(42)로 출력됩니다. 릴레이는 스위치를 누르는 것과 같이 COM2와 R(39) ~ R(42)을 물리적으로 연결하기 때문에 D.C 와 A.C를 모두 ON/OFF 시킬 수 있습니다.



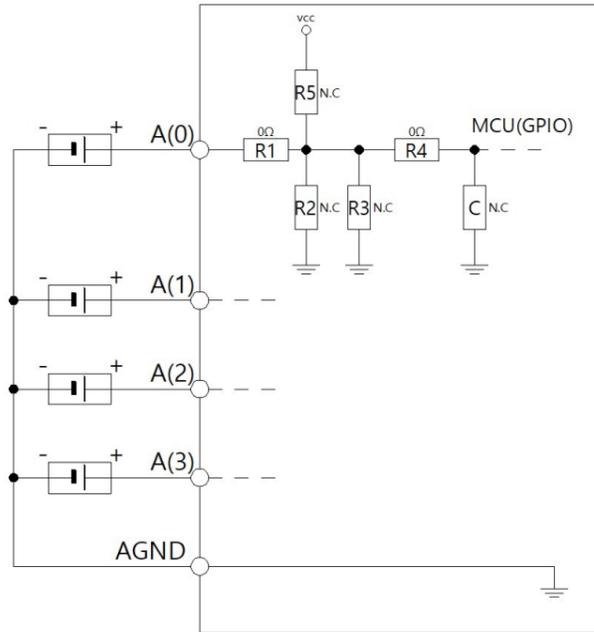
□ 1초마다 출력을 ON/OFF 시키는 예제

delay(ms) 명령어를 사용하여 시간지연을 사용할 수 있습니다.

```
void setup() {
    pinMode(39, OUTPUT); // R(39)을 출력모드로 설정합니다.
}
void loop() {
    digitalWrite(39, HIGH); // R(39)을 ON 시킵니다.
    delay(1000);           // 1000ms 동안 기다립니다.
    digitalWrite(39, LOW); // R(39)을 OFF 시킵니다.
    delay(1000);           // 1000ms 동안 기다립니다.
}
```

□ 아날로그 입력

☞ 아날로그 입력포트 A(0) ~ A(3)에 입력된 아날로그 전기신호(DC 0V~5V)를 analogRead(pin) 명령어를 사용하여 디지털 값으로 변환하여 사용합니다.

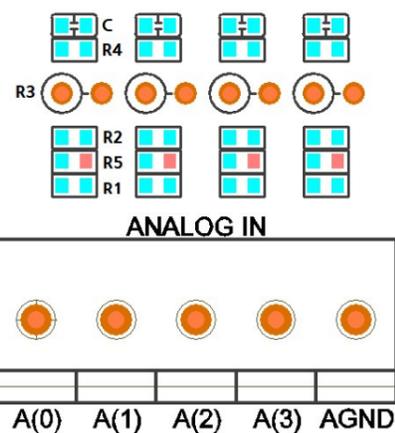


☞ 제품 출하는 R1과 R4가 0Ω으로 삽입되어 있으며 R2,R3,R5,C는 삽입이 되어 있지 않습니다.

□ 아날로그 입력범위 변경

☞ 저항 값을 변경하여, 아날로그 입력의 전기적인 입력신호의 범위를 변경할 수 있습니다. 0~5V가 기본설정으로 저항이 삽입되어 있습니다.

	R1	R2	R3	R5
0~5V	0	N.C	N.C	N.C
0~10V	100M	100M	N.C	N.C
0~20mA	0	N.C	250	N.C
NTC	0	N.C	N.C	10k



R4와 C는 LPF(Low Pass Filter)입니다. 노이즈를 제거하기 위한 주파수가 있을 경우, R4와 C를 적절히 사용합니다

□ 아날로그 입력 프로그램 예

☞ A(0) 포트에 입력된 0V~5V의 전기신호를 0~1023의 디지털 값으로 변환하여 ADC0 변수에 저장하는 프로그램입니다. analogReference(EXTERNAL)로 기준전압 값을 설정해야 하며, analogRead(pin)으로 아날로그 값을 읽어올 수 있습니다.

```

unsigned int ADC0; // ADC0 변수 생성

void setup() {
  // 아날로그 최댓값 기준을 VREF핀에 입력된 전압으로 설정
  // MPINO-16A8R은 5V가 VREF에 연결되어 있습니다.
  analogReference(EXTERNAL);
}

void loop() {
  // A(0)에 입력된 0~5V를 0~1023의 디지털 값으로 변환하여 ADC0에 저장
  ADC0 = analogRead(A0);
}

```

□ 아날로그 입력값 스케일 계산

☞ analogRead(pin) 명령어를 이용하여 읽어들인 0 ~ 1023 디지털 값을 프로그램에서 로직으로서 사용하기 위해 실제 센서의 Range로 디지털 값을 변환하기 위해 스케일 계산법으로 디지털 범위를 재설정하는 방법입니다.

$$\text{Scale} = (\text{In} / 1023) * (\text{Scale_Max} - \text{Scale_Min}) + \text{Scale_Min}$$

```

unsigned int ADC0; // ADC0 변수 생성

void setup() {
  // 아날로그 최댓값 기준을 VREF핀에 입력된 전압으로 설정
  // MPINO-16A8R은 5V가 VREF에 연결되어 있습니다.
  analogReference(EXTERNAL);
}

void loop() {
  //A(0)에 입력된 0~5V를 0~1023의 디지털 값으로 변환하여 ADC0에 저장
  ADC0 = analogRead(A0);
  // 스케일공식 = (in/in_Max) * (Scale_Max - Scale_Min) + Scale_Min
  // 0V일 때 0, 5V일 때 3000으로 디지털 값 범위를 재설정.
  ADC0 = ((unsigned long)ADC0 * (3000-0)) / 1023 + 0;
}

```

□ 인터럽트 (attachInterrupt)

☞ SDA(20), SCL(21) 포트로 총 2개의 인터럽트를 사용할 수 있습니다.
☞ 4.7kΩ 풀업 저항이 내장되어 있습니다.

☞ 인터럽트 포트는 MCU의 GPIO와 Direct로 연결되어 있습니다. 때문에 DC 5V 이상의 전압이 인가되면 MCU가 손상될 수 있습니다.

☞ 관련 명령어

attachInterrupt(digitalPinToInterrupt(pin)), ISR, mode);

- pin : SDA(D20), SCL(D21)
- ISR : 호출되는 함수명
- mode : LOW, CHANGE, RISING, FALLING
 - FALLING : 하강검출 (입력상태가 ON에서 OFF로 될 때)
 - CHANGE : 변경검출 (입력상태가 변경될 때)
 - RISING : 상승검출 (입력상태가 OFF에서 ON으로 될 때)

☞ SCL 입력이 OFF였다가 ON될 때, _INT0() 함수를 호출하는 예제입니다.

```

void setup() {
  Serial.begin(9600); // 다운로드포트를 9600보레이트로 오픈
  // PWM2포트에 상승엣지 입력이 검출되면, _INT0 함수 호출실행
  attachInterrupt(digitalPinToInterrupt(SCL),_INT0, RISING);
}

void loop() {

  // SCL포트에 상승엣지 입력이 검출되면, 실행되는 함수.
  void _INT0(){
    // 다운로드포트로 "ok"를 송신
    Serial.println("ok");
  }
}

```

□ 엔코더 입력

☞ A상과 B상이 출력되는 Open-Collector 방식의 엔코더를 1개 연결할 수 있습니다.

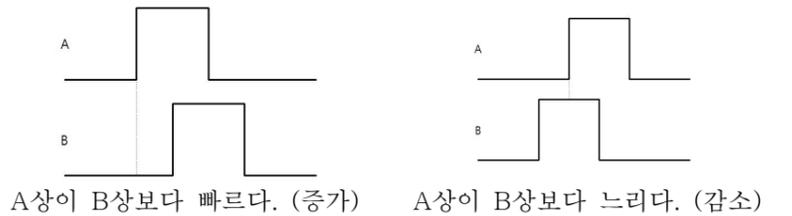
☞ I2C포트의 SDA(20)에 A상을 SCL(21)에 B상을 연결합니다.

- 4.7kΩ Pull-Up 저항이 내장되어 있습니다.

- 엔코더 연결시, I2C 통신은 사용할 수 없습니다.

☞ 입력가능 주파수는 50kHz 이상 가능합니다.

☞ 일반적으로 A상의 펄스신호의 상승엣지 기준으로 B상이 늦으면 엔코더 값은 증가하고 B상이 빠르면 감소합니다.



□ 엔코더 사용방법

☞ 아래 코드는 SDA(20)<->A상, SCL(21)<->B상 포트를 사용하여 측정하는 예제입니다.

```

volatile long encoderValue = 0; // 엔코더 값 저장 변수
void setup() {
  Serial.begin(115200);
  attachInterrupt(digitalPinToInterrupt(SDA), updateEncoder, FALLING);
}

void loop() {
  Serial.print("Encoder Value: "); Serial.println(encoderValue);
  delay(500);
}

void updateEncoder() {
  if (digitalRead(SCL) == false) { encoderValue++; }
  else { encoderValue--; }
}

```

□ 상태 LED

☞ LED_BUILTIN 변수명 또는 D13 핀으로 제품에 삽입되어 있는 STATUS LED를 ON/OFF 시킬 수 있습니다.

```
void setup() {
  pinMode(LED_BUILTIN, OUTPUT); //LED_BUILTIN을 출력모드로 설정
}
void loop() {
  // I(22)가 HIGH 이면, LED_BUILTIN를 ON 시킵니다.
  if (digitalRead(22) == 1) { digitalWrite(LED_BUILTIN, HIGH); }
  // I(22)가 HIGH가 아니면, 즉 LOW 이면, LED_BUILTIN를 OFF 시킵니다.
  else { digitalWrite(LED_BUILTIN, LOW); }
}
```

□ I2C 통신포트

- ☞ 1개의 I2C 통신포트를 지원합니다.
- ☞ 명령어는 아두이노 공식 홈페이지의 Wire 함수 설명서를 참조해 주시기 바랍니다. (<https://www.arduino.cc/en/Reference/Wire>)

□ 디버깅

☞ Debug는 Serial 함수를 이용해 주세요.

```
unsigned int ADC0; // 아날로그입력값 변수를 선언
void setup() {
  // 아날로그입력 기준전압을 VREF핀에 연결되어 있는 5V로 설정
  analogReference(EXTERNAL);
  Serial.begin(9600); // 업로드 포트를 보레이트가 9600인 시리얼포트로 정의
  for(int i = 39; i<=46; i++){pinMode(i, OUTPUT);}
}
void loop() {
  // 디지털입력이 ON일 때 릴레이출력을 ON, OFF일 때 릴레이출력을 OFF
  for (int k = 0; k < 8; k++) {
    if (digitalRead(22+k)==1 || digitalRead(30+k)==1) digitalWrite(39+k, 1);
    else digitalWrite(39 + k, 0);
  }
  // A(0)에 입력된 아날로그 신호를 0~1023으로 변환하여 ADC0 변수에 저장
  ADC0 = analogRead(A0);
  // 아날로그입력값 ADC0을 스케일 연산하여 0 ~ 3000으로 범위 변경
  // 스케일공식 = (in/in_Max) * (Scale_Max - Scale_Min) + Scale_Min
  ADC0 = ((unsigned long)ADC0 * (3000-0)) / 1023 + 0;
  Serial.print("Analog Input Value : ");
  Serial.println(ADC0);
  delay(500);
}
```

□ 시리얼 통신포트

통신 키워드	RS-232	RS-485	UART
	Serial1	Serial2	Serial3

- ☞ 시리얼 통신방법은 아래의 아두이노에서 제공하는 설명을 참고해 주시기 바랍니다. ([아두이노 IDE의 Serial 함수 사용설명서](#))
- ☞ 산업현장에서는 산업 범용 프로토콜인 모드버스 통신 프로토콜을 주로 사용합니다.
- ☞ ILIB로 Modbus RTU Master를 사용하는 예제를 참고해 주시기 바랍니다. (<https://blog.naver.com/ilogics/222453991523>)
- ☞ ILIB로 Modbus RTU Slave를 사용하는 예제를 참고해 주시기 바랍니다. (<https://blog.naver.com/ilogics/222453993604>)

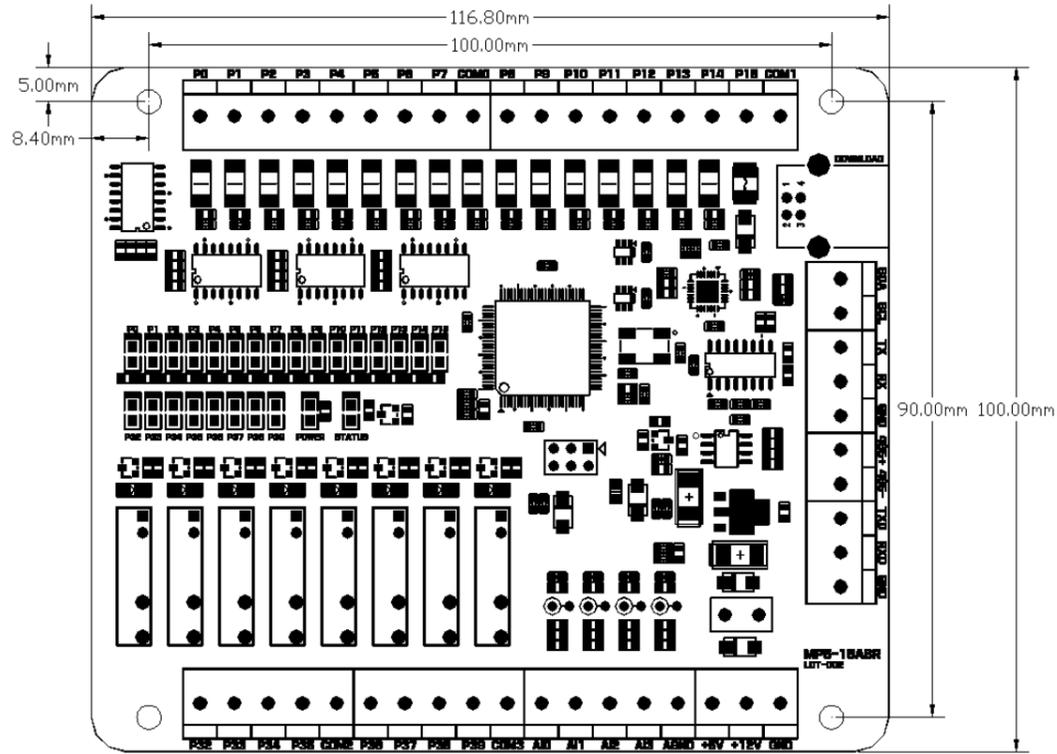
□ MPINO STUDIO

☞ 저희 (주)아이로직스에서는 산업에서 사용하기 쉽도록 Arduino 와 Ladder Logic을 모두 사용하여 프로그램 할 수 있는 MPINO STUDIO를 무료로 제공하고 있습니다.

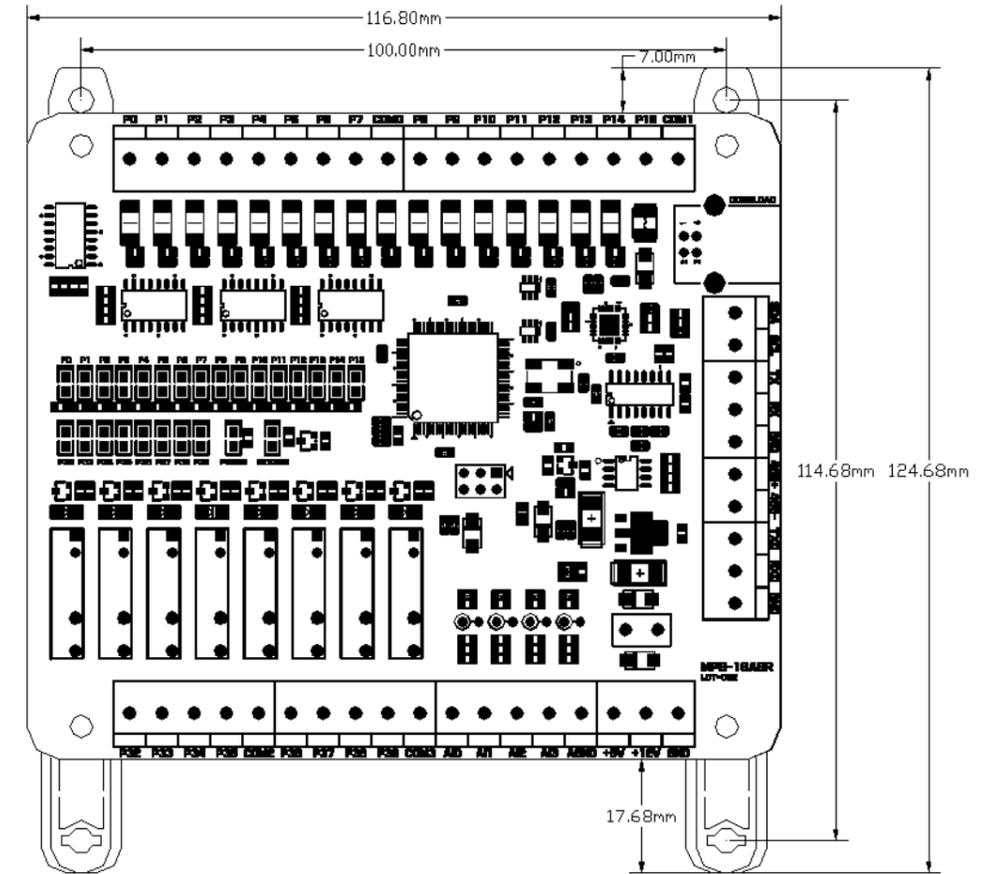
□ 감사드립니다.

- ☞ 저희 (주)아이로직스의 제품을 구매해주셔서 감사드립니다.
- ☞ 구매는 <https://www.ilogics.co.kr> 쇼핑몰에서 하실 수 있습니다.
- ☞ 구매/기술 상담은 0507-1362-5020으로 전화 주시기 바랍니다. (상담시간은 오전10시~오후5시 입니다)

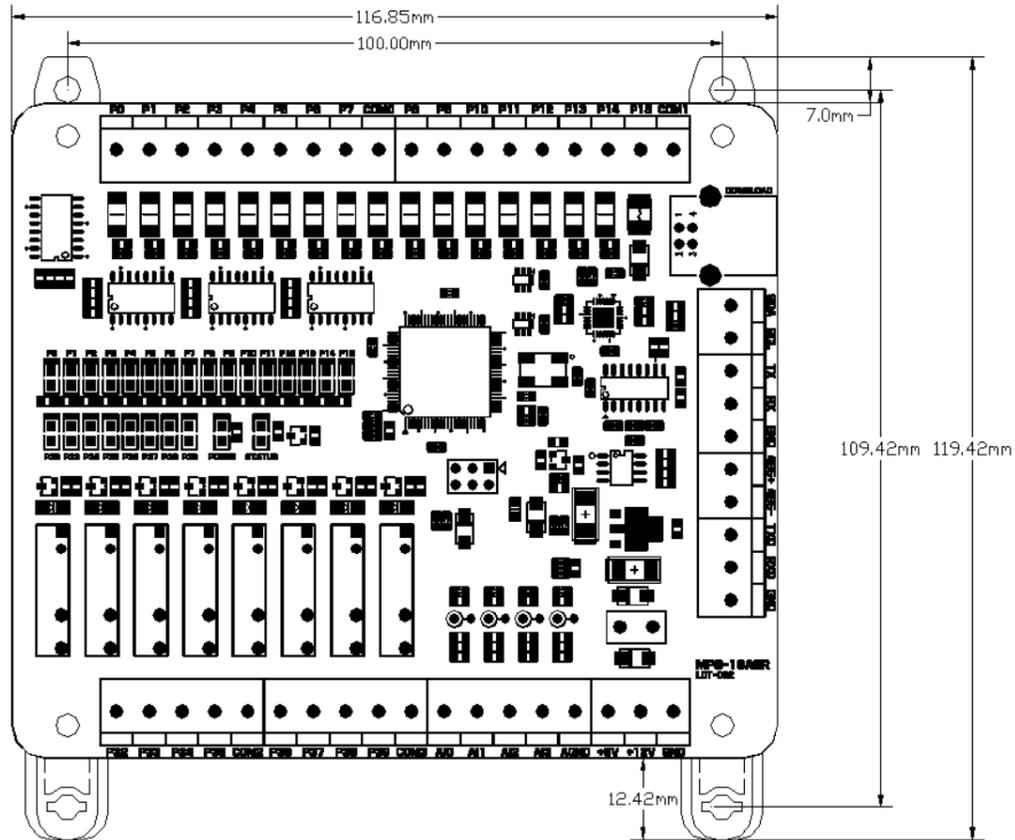
□ DIMENSION (PCB / 단레일 사용하지 않을 경우)



□ DIMENSION (클립 열었을 때 / 단레일 체결 전)



□ DIMENSION (클립 닫았을 때 / 단 레일 체결 후)



□ DIMENSION (단 레일 : 35mm)

