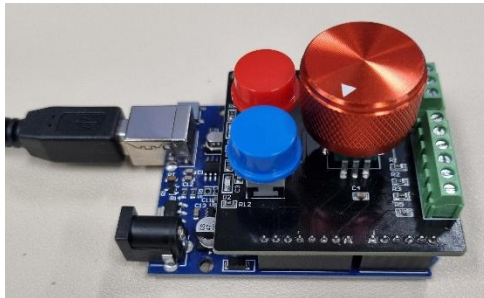


MAS-001 + 아두이노 우노 R3 호환보드

모터뱅크

모델명 : MAS-001 과
아두이노 우노 R3 호환보드



제품설명

아두이노 우노 R3 는
아두이노 확장섀드와
각종 모터드라이브를 연결해서
제어용으로 사용할수 있는 보드로
MCU 에 다양한 코드를 실습할수 있습
니다. 그리고 MAS-001 섀드보드를 추
가해서 쉽게 버튼제어와 속도조절까
지 실험해 볼수 있는 세트입니다.

아두이노 우노 R3 상세구성

- 1) 아두이노 우노보드로 정품보드와 동일형식 구성보드.
- 2) 아두이노중 가장 많은 사용층이 있는 우노보드이며 다양한 자료들이 많은게 장점.
- 3) MCU : Atmega328P (SMD or DIP).

제품의 전기적 특성 :

로직전압 : DC 5V
Vin 전압(Jack) : 7 ~ 12V
I/O 핀당 작동 전류 : 20mA
디지털 I/O 핀 : 14 개
PWM 핀 : 6 개(디지털 I/O 핀과 중복사용)
아날로그 입력핀 : 6 개

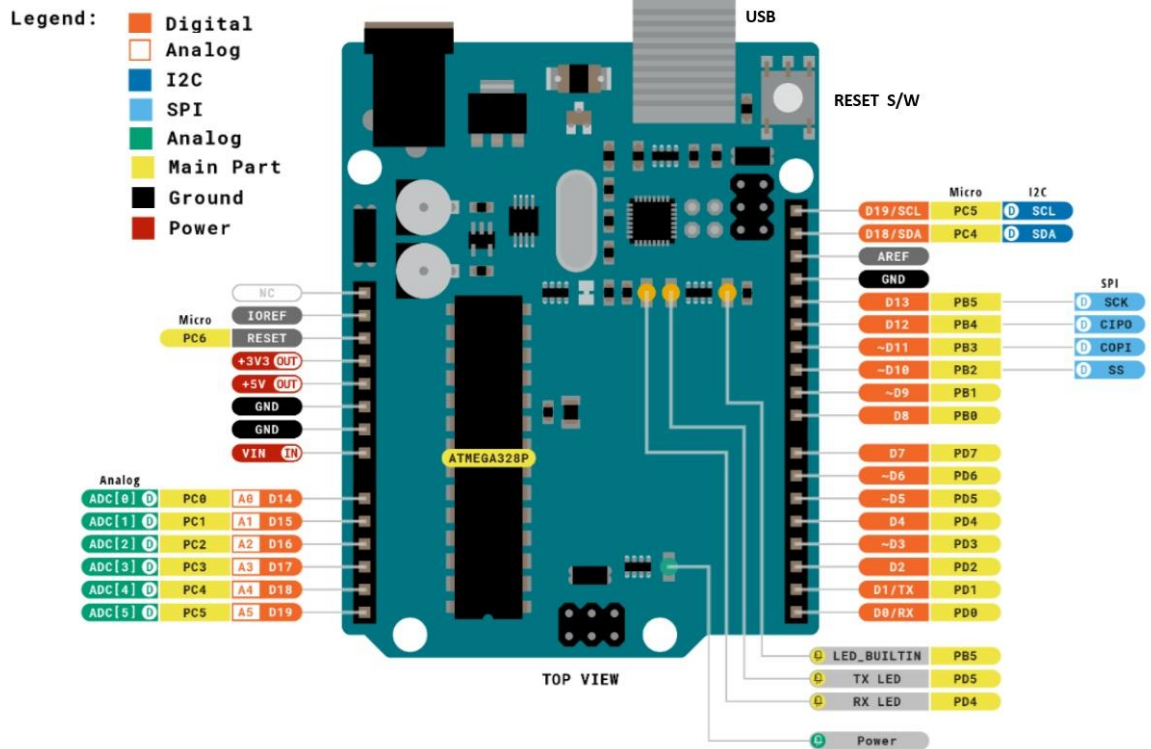
사양 :

MCU : Atmega328P (DIP 혹은 SMD)
16 MHZ, 2kB SRAM, 1KB EEPROM
32KB Flash(0.5KB bootrom)

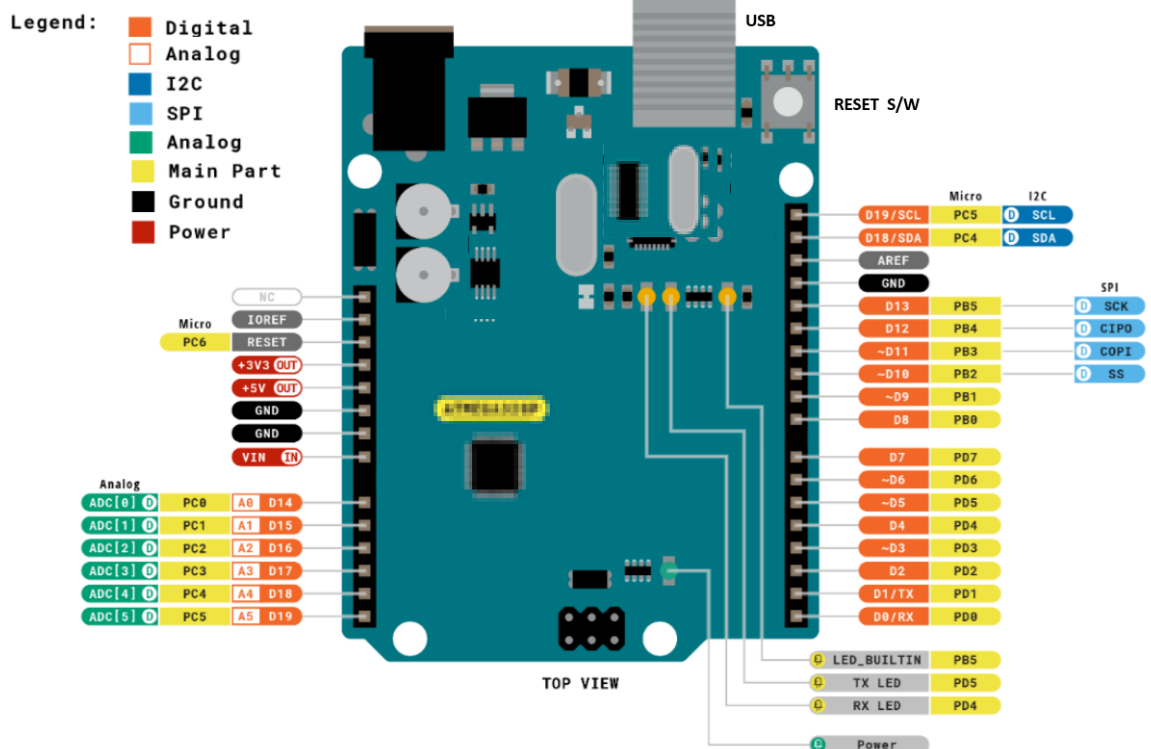
▶ [Arduino Uno9\(R3\)](#)

아두이노 Arduino Uno(R3)	
Microcontroller	ATmega328
Operating Voltage	5V
Input Voltage (recommended)	7-12V
Input Voltage (limits)	6-20V
Digital I/O Pins	14 (of which 6 provide PWM output)
Analog Input Pins	6
DC Current per I/O Pin	40 mA
DC Current for 3.3V Pin	50 mA
Flash Memory	32 KB (ATmega328) of which 0.5 KB used by bootloader
SRAM	2 KB (ATmega328)
EEPROM	1 KB (ATmega328)
Clock Speed	16 MHz

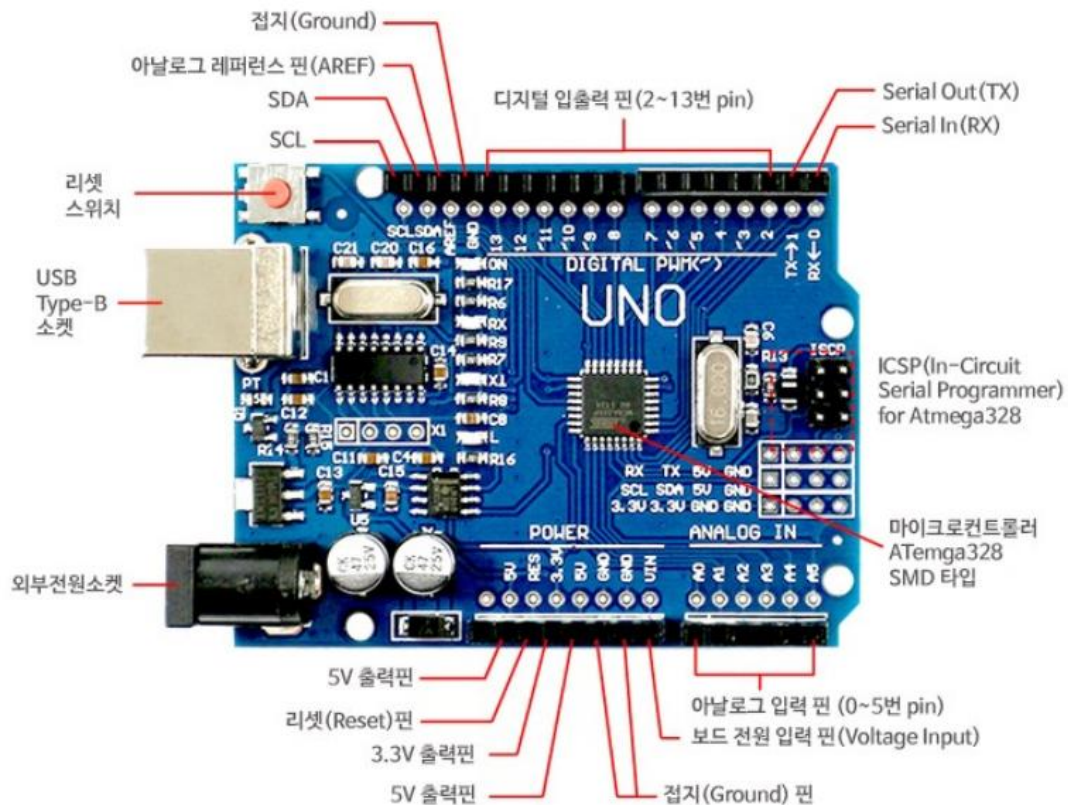
DIP type Board



SMD Type Board



두가지 클론보드가 경우에 따라 판매되는데
SMD 가 DIP 보다 대량생산에 유리하여 점차 SMD 로 바뀌고 있습니다.
DIP 은 H/W 디버깅과 고장시 수리가 용이한 점이 있지만 요즘은 장점이 크지가 않습니다.



제품치수(Dimension Drawing) :

68mm(L) x 52mm(W) Unit : mm

본 제품은 다양한 방법으로 활용이 됩니다.

개발환경은 PC 에서 아두이노 개발환경을 다운받아서 구축하면 되고

컴파일이 완료된 코드를 USB 로 간단하게 아두이노 우노보드로 다운로드 할수 있습니다.

모터뱅크에서

주로 사용하는 것은 모터(DC,BLDC,STEP)들을 돌리기 위한 방법으로 사용되고

모터드라이버에 연결하여 사용하면 다양한 제어기를 구성시킬수 있습니다.

가장 일반적이고 쉬운 형태는

아두이노 우노용 쉴드보드(MAS-100)를 추가하여 간단한 코딩으로 수동제어와 자동제어를 연습해보고 확장하는 방법입니다

참고로

여기서는 아두이노 우노의 개발환경구축은 생략합니다.

다양한 블로그나 유튜브에서 잘 소개하고 자료도 넘치므로

굳이 지면을 할애할 필요가 없다고 생략하니 양해바랍니다.

이후부터는

아두이노 우노에 쉴드보드(MAS-100)와 모터드라이브를 연결하는 예로

설명을 진행합니다.

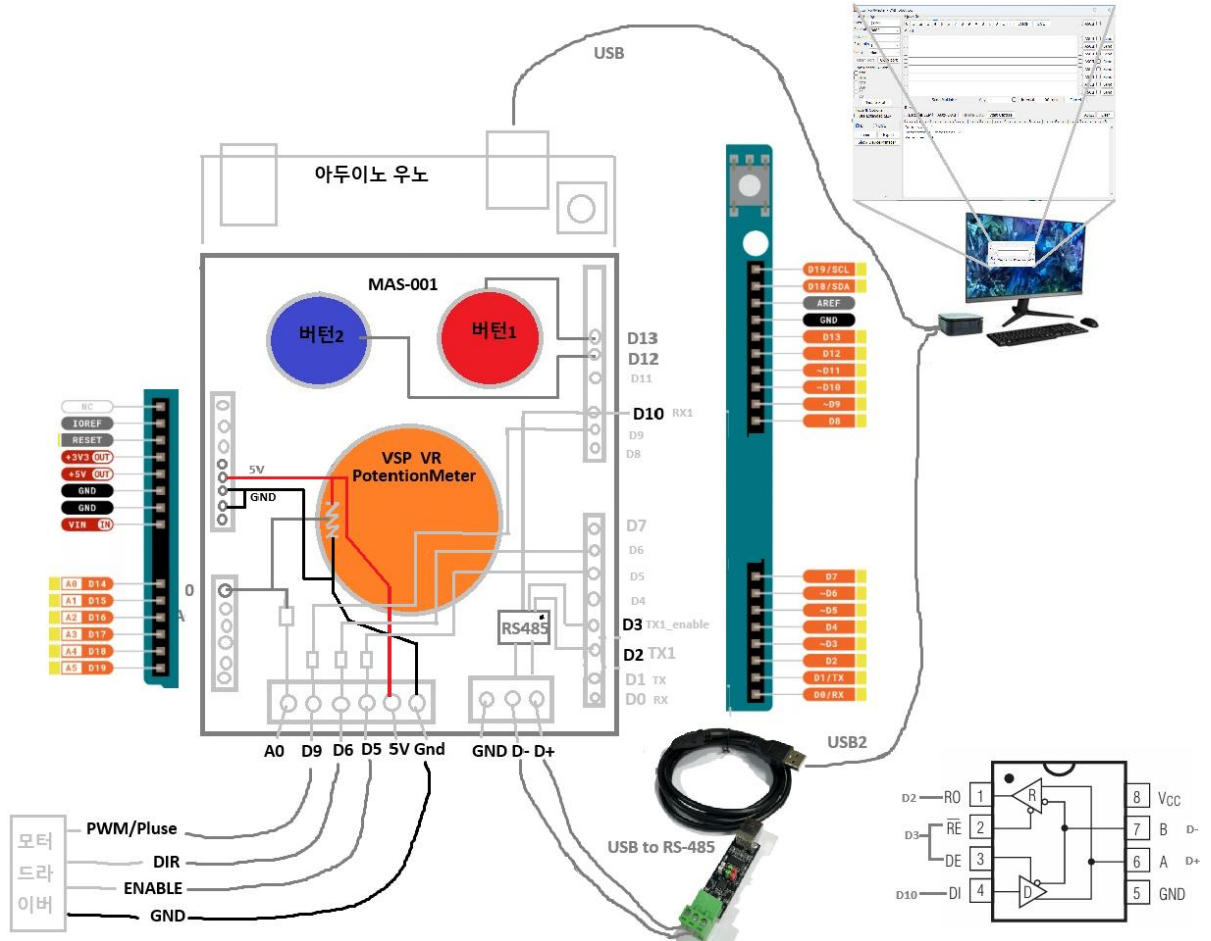
The diagram illustrates the internal components and pin connections of an Arduino Uno board. The main components shown include the ATmega328P microcontroller, the FT232RL USB-to-UART bridge, and the MAX3232 DC-DC converter. The board is populated with various passive components like resistors and capacitors. The pin headers are labeled with their functions: Digital pins (0-13), Analog pins (A0-A5), Power pins (GND, 5V, VIN), and I2C/SPI pins. The bottom right shows a detailed view of the pin headers with their corresponding pin numbers and functions.

4

아두이노 코딩방법 :

- 1) 버튼 1 이 디지털 I/O 의 D13 에 연결되어있어 이 포트를 읽고 D5 로 리매핑 .
- 2) 버튼 2 가 디지털 I/O 의 D12 에 연결되어있어 이 포트를 읽고 D6 로 리매핑 .
- 3) MAS-001 의 포트 D9 에 아두이노 D9 가 연결되어있어 이 포트를 PWM/pluse 출력으로 이용하면 가변저항의 변화를 읽어 속도제어로 연결가능합니다..
- 4) 비슷하게 RS485 나 FG 도 처리하면 됩니다

Mas-001 + 아두이노 우노 + RS-485 구성예 .



상기같이 구성하면

PC 에서 앱 시리얼툴을 켜고 모든제어환경을 사용하고 모니터링도 가능합니다 .
아두이노에 시리얼포트를 2 개열고(1 개는 H/W,또하나는 S/W) 그중 하나로는
PC 에서 모니터링하고 다른 하나로는 타장비를 제어하기 위한 RS-485 용으로
사용하게 구성할수 있습니다 .

그리고 가변저항볼륨조절과 스위치 두개를 PWM 혹은 펄스출력으로 하며 방향과
Stop/Start 식으로 연결할수 있기에 효과적인 실습이 가능합니다 .

아두이노 우노의 코딩소스 예;

참조: 같은 소스가 현디렉토리에 upload 되어 있습니다.

```
//arduino uno + mas-001 test project--

#include <SoftwareSerial.h>

//portD 0~7: arduino 0~7
//portB 0~7: arduino 8~13
//portA 0~7: arduino 14~19

#define RS485_RX 10
#define RS485_TX 2
#define RS485_DE 3
#define EN_PIN 6
#define DIR_PIN 5
#define PWM_PIN 9

#define MAS_BTN1 13
#define MAS_BTN2 12
#define MAS_PORT A0

class MAS001
{
public:
    MAS001();
    bool button1Clicked();
    bool button2Clicked();
    uint16_t getPort();
};

MAS001::MAS001(){
    pinMode(MAS_BTN1, INPUT);
    pinMode(MAS_BTN2, INPUT);
}

bool MAS001::button1Clicked(){
    if(digitalRead(MAS_BTN1) == LOW) return true;
    return false;
}

bool MAS001::button2Clicked(){
    if(digitalRead(MAS_BTN2) == LOW) return true;
    return false;
}

uint16_t MAS001::getPort(){
    return analogRead(MAS_PORT);
}

MAS001 myShield;

// RX=10 번 핀, TX=2 번 핀으로 설정
SoftwareSerial Serial1(10, 2);
```



```

int nDelay;

void setup() {
  Serial.begin(9600); //아두이노의 기본 시리얼 속도 (PC 모니터용)
  Serial1.begin(9600); // RS-485 모듈과 통신할 시리얼 (SW 모드)
  pinMode(RS485_DE, OUTPUT);

  pinMode(EN_PIN, OUTPUT);
  pinMode(DIR_PIN, OUTPUT);
  pinMode(PWM_PIN, OUTPUT);

  nDelay=1;
}

void loop()
{

  if (Serial.available() > 0) { // PC 에서 데이터가 수신되면
    char data = Serial.read(); // 데이터 읽기
    Serial.print("Received: ");
    Serial.println(data);      // 받은 데이터 출력
    if( data=='4' )
    {
      Serial.print("Received:4 ");

      // 데이터 송신 (예: 4 수신시 마다 "Hello RS485" 송신)
      digitalWrite(RS485_DE, HIGH); // 송신 모드 활성화
      delay(2);
      Serial1.println("Hello RS485"); // RS-485 모듈로 데이터 전송
      delay(2);
      digitalWrite(RS485_DE, LOW); // 송신 종료, 수신 모드로 전환
      delay(10);

    }

  }

  // 485 데이터 수신 (데이터가 오면 PC 시리얼로 출력)
  if (Serial1.available()) {
    char data = Serial1.read();
    Serial.print(data);
  }

  if(!myShield.button1Clicked()) Serial.println("Btn 1 push");
  if(!myShield.button2Clicked())
  {
    //Serial.print("Btn 2 pushed: ");
    nDelay=myShield.getPort();
    //Serial.println(nDelay);

    //pwm
  }
}

```

```

//analogWrite(9,nDelay/4); //256

//pulse gen
digitalWrite(PWM_PIN, HIGH);
delayMicroseconds(nDelay);
digitalWrite(PWM_PIN, LOW);
delayMicroseconds(nDelay);
}else{ nDelay=0;}

/*
bitSet(PORTB,1); //D9
if( nDelay )delayMicroseconds(nDelay);
bitClear(PORTB,1);
if( nDelay )delayMicroseconds(nDelay);
*/
//delay(100);
}

```


아두이노 무노 R3 클론 회로도

모션가이든