

MS-MSD-224-A1 복합제어세트

모터뱅크

모델명 : MS-MSD-224-A1



제품설명

아두이노우노(R3)와 아두이노 확장실드 그리고 스텝모터 드라이버인 MSD-224을 연결하는 제어세트로 MCU에 다양한 코드를 실습할수 있고 MAS-001 쉴드보드로 쉽게 버튼제어와 속도조절까지 실험해 볼수 있는 세트입니다.

MS-MSD-224-A1 상세구성

- 1) 아두이노 우노보드와 확장실드 및 MSD-224 으로 구성된 세트.
- 2) MSD-224을 사용하여 스텝모터를 쉽게 제어(아두이노 코드).
- 3) 제어는 버튼이나 가변저항뿐만 아니라 코드제어 및 PC제어도 가능.

제품의 전기적 특성 :

아두이노전압 : 별도전압없이 PC_USB(5V)만 구성

아두이노실드전압: 별도전압없이 사용

MS-224 전압 : DC12V ~ DC24V

추가적인 정보는
아래링크에 있습니다.

아두이노와 쉴드보드 사용법
모터뱅크 쇼핑몰 기술자료란

<https://www.motorbank.kr/board/view.php?page=2&bdId=technicaldata&sno=296>

<https://www.motorbank.kr/board/view.php?page=2&bdId=technicaldata&sno=294>

MSD-224 사용자매뉴얼

모터뱅크 쇼핑몰 기술자료란

<https://www.motorbank.kr/board/view.php?page=2&bdId=technicaldata&sno=333>

제품구성 :

본 제품은 다양한 방법으로 활용이 됩니다.

개발환경은 PC에서 아두이노 개발환경을 다운받아서 구축하면 되고

컴파일이 완료된 코드를 USB로 간단하게 아두이노 우노보드로 다운로드 할수 있습니다.

모터뱅크에서

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

모터드라이버에 연결하여 사용하면 다양한 제어를 구성시킬수 있습니다.

가장 일반적이고 쉬운 형태는

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

참고로

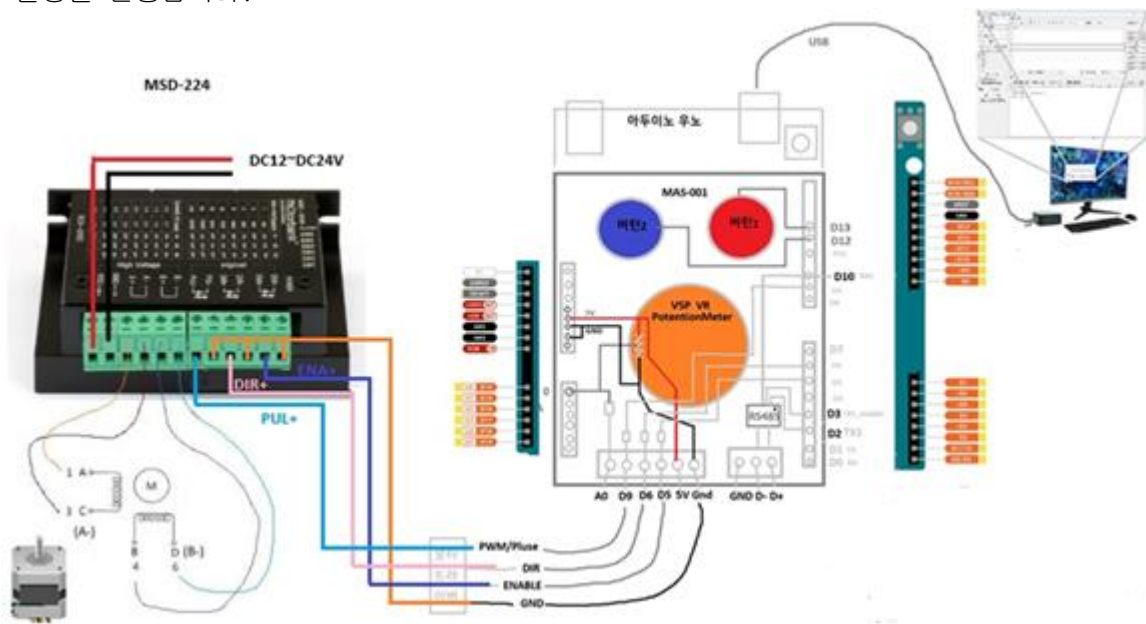
여기서는 아두이노 우노의 개발환경구축은 생략합니다.

다양한 블로그나 유튜브에서 잘 소개하고 자료도 넘치므로

굳이 지면을 할애할 필요가 없다고 생략하니 양해바랍니다.

이후부터는

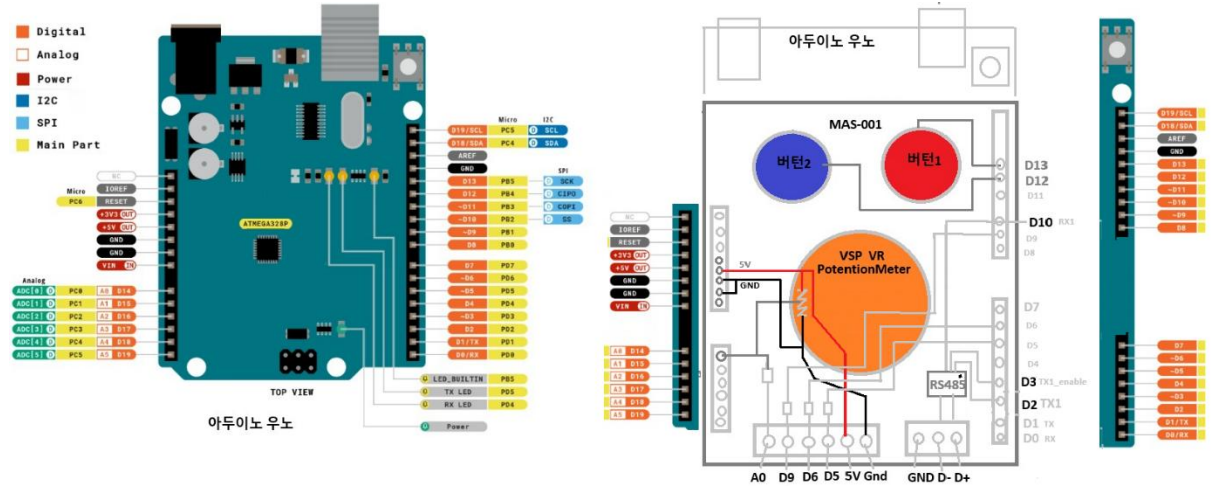
아두이노 우노에 쉴드보드(MAS-100)와 스텝모터드라이브를 연결하는 예로 설명을 진행합니다.



기본적인 모터제어 구성도면

- 1) PC에서 아두이노로 코딩하여 다운로드후 확장실드에서 제어
- 2) PC이 USB를 통하여 PC앱으로 간단제어

간단한 아두이노 우노와 쉴드보드(MAS-001) 구성

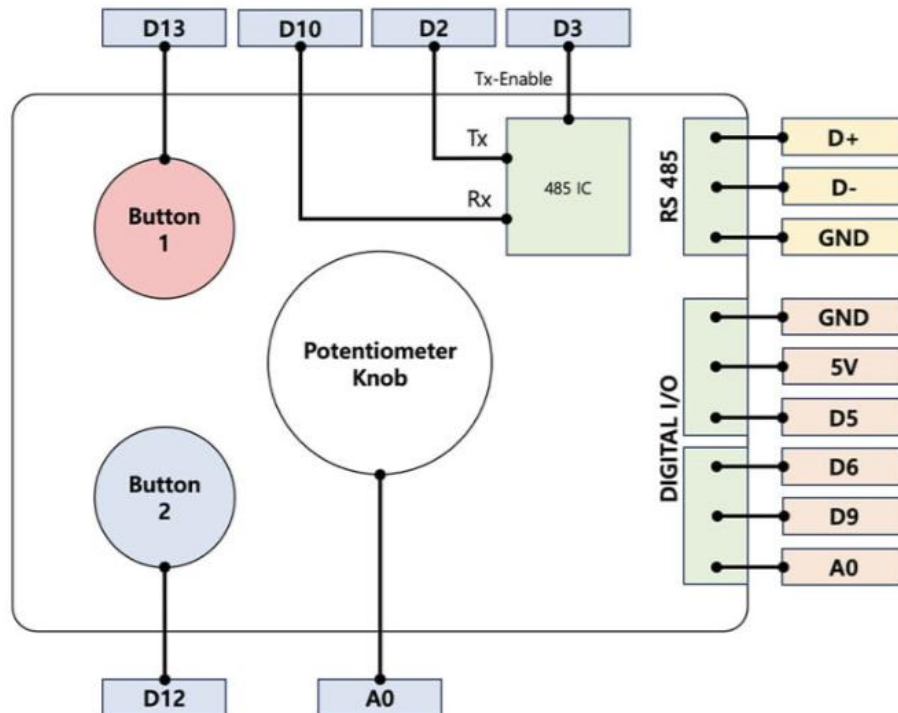


상기와 같이 쉴드보드(MAS-001)을 아두이노 우노보드에 꼽아서 사용하고
아두이노에서 Software 를 코딩하여 버튼과 출력포트(D5,D6,D9)를 매핑하고
가변저항값을 읽고 이것을 PWM 으로 변환하여 D5~D9 로 출력하거나 하면 되는식으로 제어.
특히 외부모터에서 FG 를 D5~D9 로 받아서 PWM 으로 속도제어하면 closeloop 제어도 가능.

결선구성 :

쉴드보드(MAS-001)를 아두이노 우노보드에 꼽으면 됩니다.
그리고 모터 드라이버에 VSP,PWM,DIR,FG,Enable 을 연결합니다.

제품의 연결구성도 :

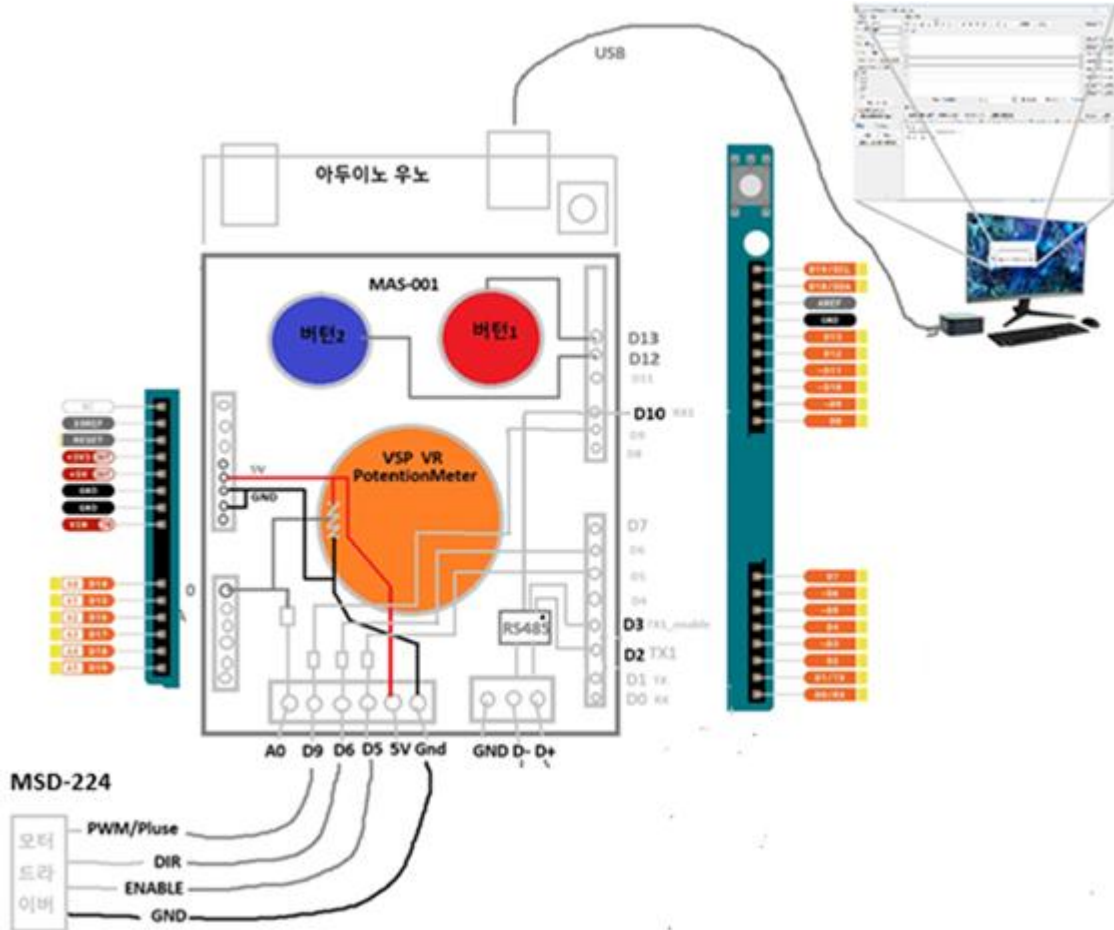


Mas-001 블록다이어그램.

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 + MSD-224 구성예 .



상기같이 구성하면

PC 에서 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 + MSD-224 test project--
//
// Copyright(c)2025 by motorBank. All rights reserved.

#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  5
#define DIR_PIN 6
#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);
}
```

```

class MSD224
{
public:
    MSD224();

    void enable(uint8_t value);
    void direction(uint8_t value);
};

MSD224::MSD224(){
    pinMode(EN_PIN, OUTPUT);
    pinMode(DIR_PIN, OUTPUT);
    //for pluse out at D9
    pinMode(PWM_PIN, OUTPUT);
}

void MSD224::enable(uint8_t value){
    digitalWrite(EN_PIN, value);
}

void MSD224::direction(uint8_t value){
    digitalWrite(DIR_PIN, value);
}

MAS001 myShield;
MSD224 myDevice;

// 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 (하드웨어 시리얼 사용 권장)

    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: 100ms 마다 "Hello RS485" 송신)
            digitalWrite(RS485_DE, HIGH); // 송신 모드 활성화
            delay(10);
        }
    }
}

```

```

    Serial1.println("Hello RS485"); // RS-485 모듈로 데이터 전송
    delay(10);
    digitalWrite(RS485_DE, LOW); // 송신 종료, 수신 모드로 전환
    delay(100);

    }

}

// 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: enable");
    myDevice.enable(LOW);
} else myDevice.enable(HIGH);

if(!myShield.button2Clicked())
{
    //Serial.println("Btn 2 pushed: dir ");
    myDevice.direction(LOW);
} else myDevice.direction(HIGH);

nDelay=myShield.getPort();
//pwm
//analogWrite(9,nDelay/4); //256

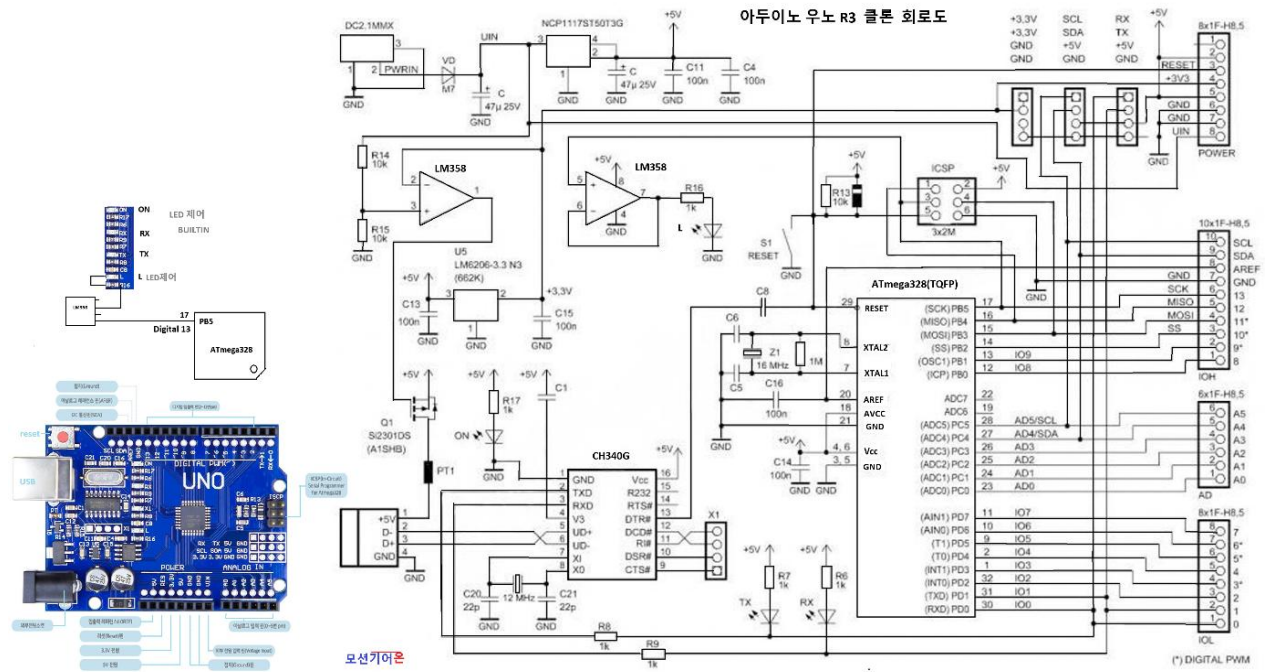
//pluse gen
digitalWrite(PWM_PIN, HIGH);
delayMicroseconds(nDelay);
digitalWrite(PWM_PIN, LOW);
delayMicroseconds(nDelay);
//delay(50);

/*
    bitSet(PORTB,1); //D9
    if( nDelay )delayMicroseconds(nDelay);
    bitClear(PORTB,1);
    if( nDelay )delayMicroseconds(nDelay);
    Serial.print("B pushed: dir ");
    */
//delay(100);
}

//--code_end-----

```

아두이노 우노 R3 클론보드 회로도.



아두이노 우노 확장실드 mas-001 회로도.

