

C h a p t e r

12



DC모터 동작

01

기어드 DC모터 동작

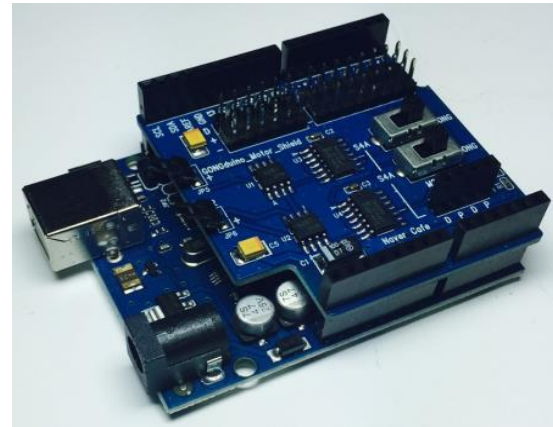
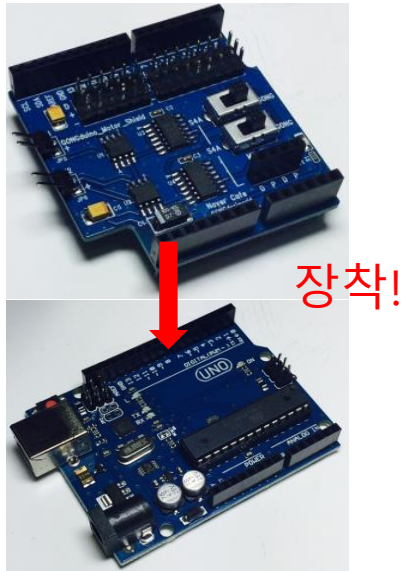
아두이노와 모터셴드를 적층한다.

■ 아두이노와 셴드의 적층

아두이노의 출력 핀에서 공급 가능한 최대 전류(40mA)가 모터 구동 하는 전류에 충분하지 않아 모터 셴드를 사용하여 모터에 충분한 전류를 공급 할 수 있게 한다.

DC모터는 Vcc와 GND 두 개의 연결선을 가지며 연결 방향에 따라 정회전 또는 역회전을 한다.

모터셴드를 사용하여 정, 역회전을 제어할 수 있고, pwm출력으로 모터의 속도 또한 제어 가능하다.



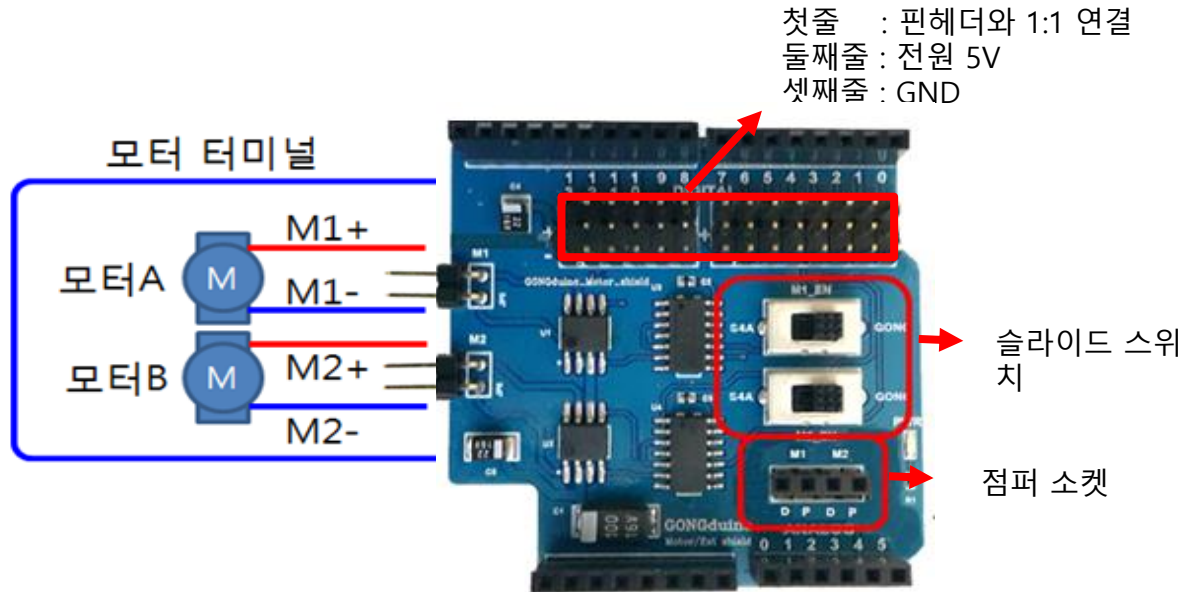
아두이노와 모터셴드 연결사진

02

기어드 DC모터 동작

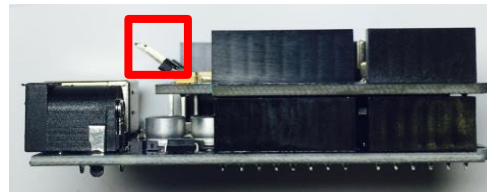
모터실드의 세부 명칭을 익힌다.

■ 모터실드 상세 설명



[모터실드 상세 명칭]

👉 주의
아두이노와 적층시 터미널을 손으로 윗쪽방향
으로 천천히 휘어준다.(핀 부러짐 주의)



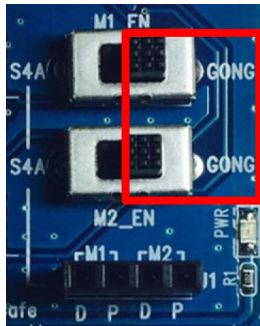
[베이스보드+모터실드 적층 단면]

03

기어드 DC모터 동작

모터실드의 설정방법을 익힌다.

■ 모터 실드의 슬라이드 스위치와 점퍼소켓 사용법



[GONG으로 설정 시]

슬라이드 스위치를 'GONG'으로 설정할 경우

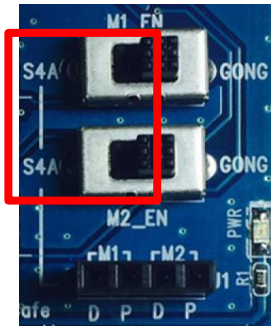
- 점퍼소켓은 사용하지 않고 그대로 둔다.
- 모터실드 내부 회로에서 아두이노 보드의 디지털 입출력핀 중 4,5,6,7번 핀을 사용하도록 연결해 놓았기 때문에 이 넷 핀들은 모터제어에 할당되어진다(적층 시).
- 4,5,6,7핀은 다른 목적으로 사용될 수 없다(아래 표 참조).

아두이노 보드	모터 실드
디지털입출력 핀 4번	모터A의 방향(Direction) 제어
PWM 핀 5번	모터A의 속도(Speed, PWM) 제어
PWM 핀 6번	모터B의 속도(Speed, PWM) 제어
디지털입출력 핀 7번	모터B의 방향(Direction) 제어

"GONG"으로 슬라이딩 한 후, 스케치에서는 4,5,6,7번 핀을 사용하여 모터 제어한다.

■ 모터 실드의 슬라이드 스위치와 점퍼소켓 사용법

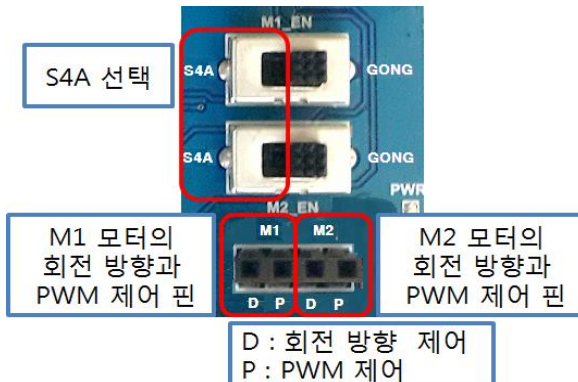
슬라이드 스위치를 'S4A'로 설정할 경우



[S4A로 설정 시]

- 점퍼소켓을 사용하여 원하는 디지털 입출력 핀과 연결시켜준다.
- 모터실드 내부 회로에서 'GONG'으로 연결된 신호 선(4,5,6,7번핀)을 끊음으로써, 점퍼 소켓에 점퍼케이블을 이용하여 어떠한 디지털 입출력 핀과 연결할 수 있다.
- 4,5,6,7번 핀도 사용가능하며, PWM가능한 핀을 자유자재로 정하여 속도제어를 할 수 있다.

점퍼소켓 사용법



- 소켓의 M1과 M2는 실드의 좌측 모터 연결 터미널의 이름과 동일하다.
- 즉, M1의 D와 P는 모터 A의 D(방향:Direction)와 P(속도:PWM)를 위한 연결을 직접 할 수 있다.
- 예를 들면, M1의 D와 아두이노 8번 핀을 점퍼시키고, M1의 P에는 9번 핀을 연결한다.

PWM가능 핀은 "~"표시가 있는 **3,5,6,9,10,11**번 핀이다.

구동 로봇을 전진 또는 후진하기 위해 아두이노 보드에 연결된 모터 쉴드의 제어 신호를 제어하는 방법을 실습한다.

하드웨어 연결

1. 아두이노 보드와 모터 쉴드 보드는 커넥터를 통해 다음 표와 같이 연결되어 있다.

아두이노 보드 제어 핀	모터 쉴드 보드
4번 핀	모터 A의 방향(Direction) 제어
5번 핀	모터 A의 속도(PWM, Speed) 제어
6번 핀	모터 B의 속도(PWM, Speed) 제어
7번 핀	모터 B의 방향(Direction) 제어

2. 모터 제어를 위해 사용되는 아두이노 보드의 4번-7번 핀은 모두 출력(OUTPUT)으로 설정한다.

■ DC 모터의 동작 속도 제어

- 모터의 동작 속도를 제어하려면 `analogWrite()` 함수를 사용한다.
- 아두이노에서 PWM 출력이 가능한 핀 : 디지털 3, 5, 6, 9, 10, 11번 핀.

모터 쉴드 장착 시 5번 6번 핀이 모터 속도에 관련된 핀 이므로
로봇의 속도 제어에서는 5, 6번 핀만 PWM출력을 하면 된다.

■ 사용되는 아두이노 함수 :

```
analogWrite(motorPin, speed);    // speed : 0 ~ 255
speed 변수에 0 ~ 255까지의 값을 넣어 속도를 조절할 수 있다.
```

아래 프로그램을 업로드하여, DC모터의 회전방향이 올바르게 작동되고 있는지 확인한다.

```
void setup() {  
    pinMode(4, OUTPUT);    // M1 Direction  
    pinMode(5, OUTPUT);    // M1 PWM  
    pinMode(6, OUTPUT);    // M2 PWM  
    pinMode(7, OUTPUT);    // M2 Direction  
}  
  
void loop() {  
    digitalWrite(4, LOW);    // M1 모터 방향은 반시계 방향 (LOW)  
    digitalWrite(5, HIGH);    // M1 모터 속도는 최대 속도  
    digitalWrite(6, HIGH);    // M2 모터 속도는 최대 속도  
    digitalWrite(7, HIGH);    // M2 모터 방향은 시계 방향 (HIGH)  
}
```

실습 R02

구동 로봇의 전진/후진과 PWM제어

구동 로봇을 전진 또는 후진하기 위해 공두이노 보드에 연결된 모터 쉴드의 제어 신호를 제어하는 방법을 실습한다.

하드웨어 연결

1. 구동 로봇에 탑재한 아두이노 보드와 모터 쉴드 보드는 커넥터를 통해 다음 표와 같이 연결되어 있다.

아두이노 보드 제어 핀	모터 쉴드 보드
4번 핀	모터 A의 방향(Direction) 제어
5번 핀	모터 A의 속도(PWM, Speed) 제어
6번 핀	모터 B의 속도(PWM, Speed) 제어
7번 핀	모터 B의 방향(Direction) 제어

2. 모터 제어를 위해 사용되는 공두이노 보드의 4번-7번 핀은 모두 출력(OUTPUT)으로 설정한다.

프로그램 작성

DC_PWM

```
void setup() {  
  pinMode(4, OUTPUT); 1)  
  pinMode(5, OUTPUT); 2)  
  pinMode(6, OUTPUT); 3)  
  pinMode(7, OUTPUT); 4)  
}  
void stop() { 5)  
  digitalWrite(5, LOW);  
  digitalWrite(6, LOW);  
}
```

- 1) M1모터 방향 제어 핀
- 2) M1모터 속도 제어 핀
- 3) M2모터 속도 제어 핀
- 4) M2모터 방향 제어 핀
- 5) 바퀴의 회전을 일시 정지

다음 프로그램을 작성하여 아두이노에 업로드 시킨다.

```
void loop() {
  digitalWrite(4, LOW);    6)
  digitalWrite(5, HIGH);
  digitalWrite(6, HIGH);
  digitalWrite(7, HIGH);

  delay(2000);             7)
  stop();
  delay(1000);

  digitalWrite(4, HIGH);   8)
  digitalWrite(5, HIGH);
  digitalWrite(6, HIGH);
  digitalWrite(7, LOW);

  delay(2000);             9)
  stop();
  delay(1000);
}
```

6) 로봇플랫폼을 최대로 전진 시킨다.

7) 2초간 전진시키다가, 1초간 멈춘다.

8) 로봇플랫폼을 최대로 후진 시킨다.

9) 2초간 전진시키다가, 1초간 멈춘다.

프로그램 작성

다음 프로그램을 작성하여 아두이노에 업로드 시킨다.

```
digitalWrite(4, LOW);      10)
digitalWrite(7, HIGH);     11)
analogWrite(5, 150);       12)
analogWrite(6, 150);       13)

delay(2000);               14)
stop();
delay(1000);

digitalWrite(4, HIGH);     15)
digitalWrite(7, LOW);
analogWrite(5, 200);
analogWrite(6, 200);

delay(2000);
stop();
delay(1000);
}
```

- 10) M1모터 정방향 회전
- 11) M2모터 정방향 회전
- 12) M1모터 속도 150으로 회전
- 13) M2모터 속도 150으로 회전
- 14) 2초간 전진시키다가, 1초간 멈춘다.
- 15) M1모터 속도 200으로 회전
M2모터 속도 200으로 회전

-업로드 이후 DC모터가 정상적으로 작동하는 지 확인한다.

- 위 프로그래밍은 정 방향 최대출력→ 역방향 최대출력→ 정 방향 느리게 출력 → 역 방향 느리게 출력 → 정 방향 최대출력 순으로 진행된다.