

## 7강. DC모터 제어(BA6208)

박 원 업  
010.5451.0113

## 목 차

1. DC모터란?
3. DC모터 구동하기 위한 방법
4. BA6208 DC모터 드라이버 IC
5. 소형 DC모터 구동하기
  - 회로 구성
6. GPIO로 구동시키기
  - 코드 작성
7. PWM으로 구동시키기
  - 코드 작성

### ○ DC모터란?

- DC전원을 입력해주면 회전하는 가장 일반적인 모터.  
(Brushed DC모터 라고도 함)

직류전류로 회전을 일으키는 장치로서 회전자와 고정자로 구성되어 있다. 회전자는 일반적으로 전기자라고도 부르는데 전기자(armature)와 정류자(commutator) 그리고 브러쉬(brush)로 구성되어 있다. 고정자는 계자극(field poles)과 프레임(frame)으로 구성되어 있으며 극당 1개 이상의 권선을 갖고 있다.

[네이버 지식백과] [직류전동기](#) (두산백과)

<http://terms.naver.com/entry.nhn?docId=1188436&contentsParamInfo=isList%3Dfalse%26navCategoryId%3D32374&cid=40942&categoryId=32374>

더 자세한 내용은 위의 링크를 따라가면 된다!

- DC모터를 구동할 때 주의해야 할 점!
  - 모터가 회전하고 있을 때, **모터 샤프트를 손으로 잡는다거나** 회전하지 못하게 방해하는 행위를 하면 안됨. => 소모전류가 증가하여 **주변 회로 및 모터 드라이버나 모터가 손상될 수 있음.**
  - 모터가 정지하였을 때, 모터를 강제로 돌리는 일을 하면 모터의 성능저하가 발생할 수 있음. 또한 순간적으로 강한 전류가 발생하여 회로에 안좋은 영향을 줄 수 있음.
  - 저속에서 큰 전류를 소모함. 따라서 저속으로 오래 구동하는 것은 좋지 않음. => 저속으로 오래 구동시켜야 한다면 기어를 사용하는 것이 바람직 함.
  - 모터 구동시 강한 전류가 필요하므로, **반드시 모터 구동 IC또는 모터 구동 회로를 구성하여** 모터를 구동시킴. 일반적인 I/O 핀에 바로 연결시켜 구동하면 **MCU가 손상될 수 있음!**

- **DC모터를 구동할 때 주의해야 할 점!**

- DC모터는 극성이 바뀌면 회전방향이 바뀜.
- DC모터를 급격하게 회전시키면 순간적으로 많은 전류를 소모하며, 충분한 전류를 공급해주지 못할 경우 전류 부족으로 인한 리셋과 같은 현상이 발생할 수 있음.
- DC모터의 회전 방향을 급격히 바꾸면(또는 갑자기 회전속도를 줄이거나 정지시키면) 역기전력이 발생하여 주변 회로에 안 좋은 영향을 주며, 심한 경우 회로에 손상이 발생.

### ○ DC모터를 구동시키기 위한 방법

- 가장 흔한 방법으로는 시중에 나와있는 모터 드라이버를 이용한다. 즉, MCU와 모터를 바로 연결하지 않고, 중간에 모터 드라이버를 연결하여 **MCU↔모터 드라이버↔모터** 방식으로 모터를 제어한다.

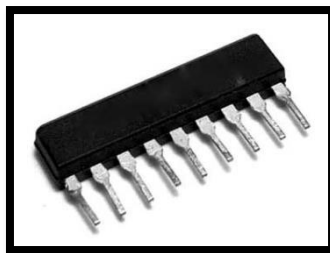
- 모터 드라이버가 하는 역할은 크게

1. 모터에 직접적인 전원 공급
2. MCU의 Logical High, Low를 입력받아 모터에 신호 출력
3. 모터의 회전 방향 제어

라고 볼 수 있다.

### ○ BA6208 Reversible Motor Driver

- 소형 DC 모터를 제어하기 위해 저렴한 가격의 모터 드라이버 IC인 BA6208을 사용



- BA6208의 입력 진리표 (데이터시트 참고)

● Input truth table			
3pin (Ain)	2pin (Bin)	8pin (Aout)	7pin (Bout)
H	L	H	L
L	H	L	H
H	H	L	L
L	L	OPEN	OPEN

Note : HIGH level input is 2.0 V or more.  
LOW level input is 0.8 V or less.

- MABUCHI RF-310T-11400 소형 DC모터

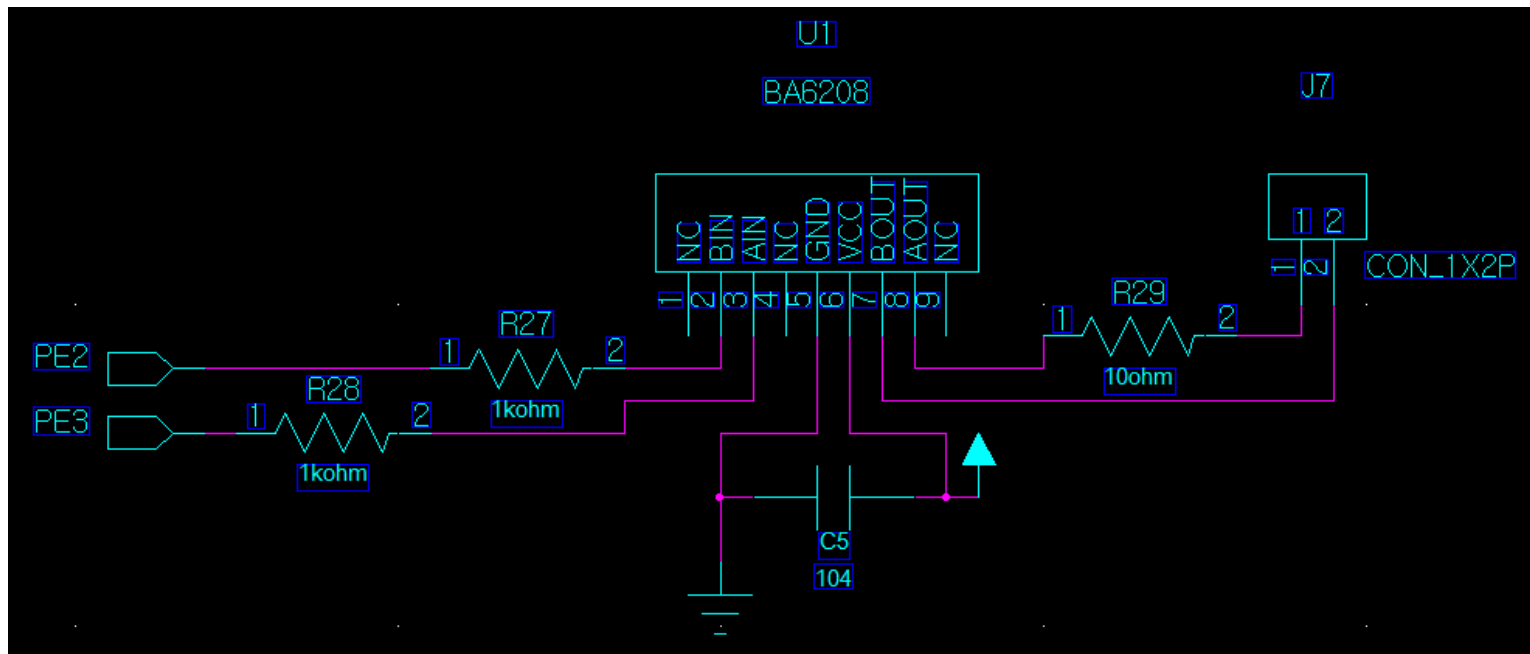
- MABUCHI社 소형 DC 모터. DVD플레이어 등에 사용됨.
- 동작 전압: 3V ~ 6V 정도





## 7강 DC모터 제어(BA6208)

다음과 같이 회로를 구성



위와 같이 BA6208의 2, 3번 핀을 각각 1k옴의 저항을 거쳐 PE.2, PE.3에 연결.  
BA6208의 8, 9번 핀 중 하나를 10옴의 저항을 거쳐 모터에 연결.  
위와 같이 연결하고 PE.2와 PE.3에 신호를 입력하면 모터가 회전한다!

다음과 같은 코드를 작성 (GPIO로 구동시키기)

```
1  #include <mega128.h>
2  #include <delay.h>
3
4  main()
5  {
6      /** GPIO 입출력 설정 **/
7      DDRE=0x0c; //PE.2와 PE.3에 BA6208 연결. 출력설정.
8      PORTE=0x00; //E포트 출력 초기화
9      /***** */
10
11     while(1)
12     {
13         PORTE=0x08; //PE.2에 Low, PE.3에 High => 정방향 회전.
14         delay_ms(1000);
15         PORTE=0x00; //PE.2에 Low, PE.3에 Low => 모터 Open.
16         delay_ms(1000);
17         PORTE=0x04; //PE.2에 High, PE.3에 Low => 역방향 회전.
18         delay_ms(1000);
19     }
20 }
```

BA6208에 신호를  
출력해주기 위해  
다음과 같이 설정.

다음과 같은 코드를 작성 (PWM으로 구동시키기)

모터 구동 함수 정의

```
33 void setDCMotor(unsigned int dir, unsigned int speed)
34 {
35     if(dir == 0) //정방향 회전
36     {
37         PORTE.2 = 0;
38         OCR3AH = (speed>>8)&0xff;
39         OCR3AL = speed&0xff;
40     }
41     else //역방향 회전
42     {
43         PORTE.2 = 1;
44         OCR3AH = ((1023-speed)>>8)&0xff;
45         OCR3AL = (1023-speed)&0xff;
46     }
47
48     //PORTE.2 = dir;
49     //OCR3AH = dir==0?(speed>>8)&0xff:((1023-speed)>>8)&0xff;
50     //OCR3AL = dir==0?speed&0xff:(1023-speed)&0xff;
51 }
```

역방향 회전 시,  
1023-speed를  
해주는 이유에 대해  
생각해보자!

다음과 같은 코드를 작성 (PWM으로 구동시키기)

```
6  main()
7  {
8      /**   GPIO 입출력 설정   **/
9      DDRE=0x0c; //PE.2와 PE.3에 BA6208 연결. 출력설정
10     PORTE=0x00; //E포트 출력 초기화
11     /**   ****   */
12
13     // Timer/Counter 3 initialization
14     // Clock source: System Clock
15     // Clock value: 62.500 kHz
16     // Mode: Ph. correct PWM top=03FFh
17     // OC3A output: Non-Inv.
18     TCCR3A=0x83;
19     TCCR3B=0x04;
20     TCCR3C=0x00;
21
22     while(1)
23     {
24         setDCMotor(0, 500);
25         delay_ms(1000);
26         setDCMotor(0, 0);
27         delay_ms(1000);
28         setDCMotor(1, 500);
29         delay_ms(1000);
30     }
31 }
```

다음과 같은 코드를 작성 (PWM으로 구동시키기)

모터 구동 함수 정의

```
33 void setDCMotor(unsigned int dir, unsigned int speed)
34 {
35     if(dir == 0) //정방향 회전
36     {
37         PORTE.2 = 0;
38         OCR3AH = (speed>>8)&0xff;
39         OCR3AL = speed&0xff;
40     }
41     else //역방향 회전
42     {
43         PORTE.2 = 1;
44         OCR3AH = ((1023-speed)>>8)&0xff;
45         OCR3AL = (1023-speed)&0xff;
46     }
47
48     //PORTE.2 = dir;
49     //OCR3AH = dir==0?(speed>>8)&0xff:((1023-speed)>>8)&0xff;
50     //OCR3AL = dir==0?speed&0xff:(1023-speed)&0xff;
51 }
```

이 두 가지 방식의  
차이점을 생각해보  
자!

문1) 스위치의 입력에 따라 모터의 속도와 방향을 제어하는 프로그램 작성.

스위치 1번 입력 => 모터 방향 설정 (Forward, Backward)

단, 방향을 바꾸면 우선 모터 정지.

스위치 2번 입력 => 모터 속도 증가

문2) 스위치의 입력에 따라 모터의 속도와 방향을 제어하는 프로그램 작성.

스위치 1번 입력 => Forward 방향으로 속도 증가

스위치 2번 입력 => Backward 방향으로 속도 증가

단, 모든 스위치 입력은 외부 인터럽트로 처리!