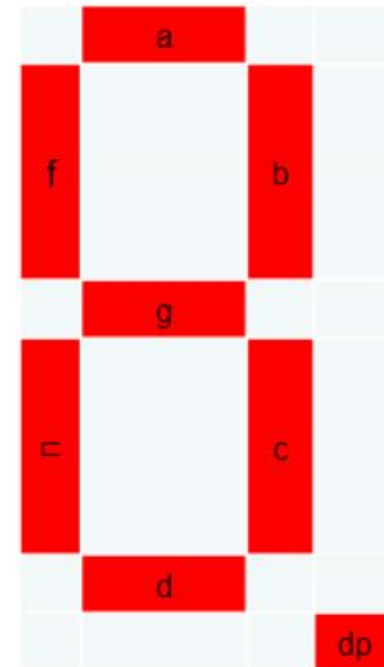
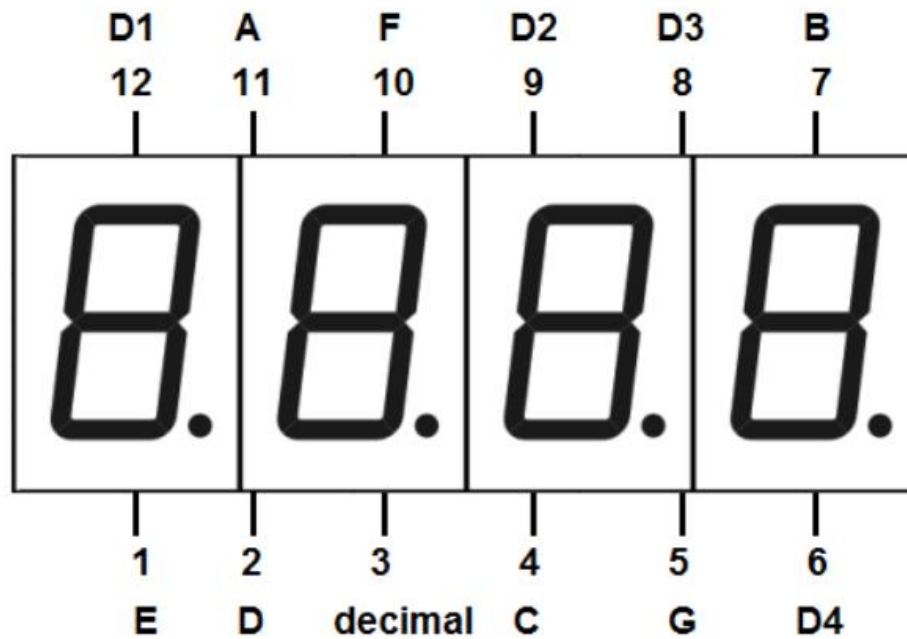


4채널 FND

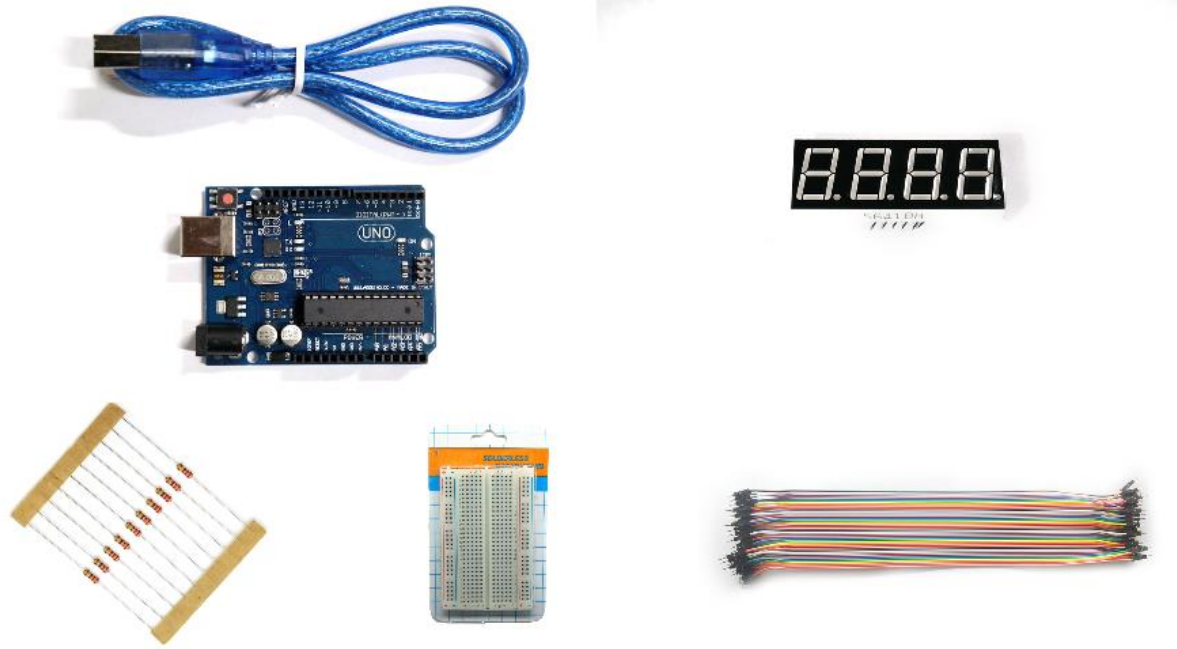
- 기초 학습
- 4채널 FND
 - 개 요
 - 예 제
- 응용 실습

4채널 FND

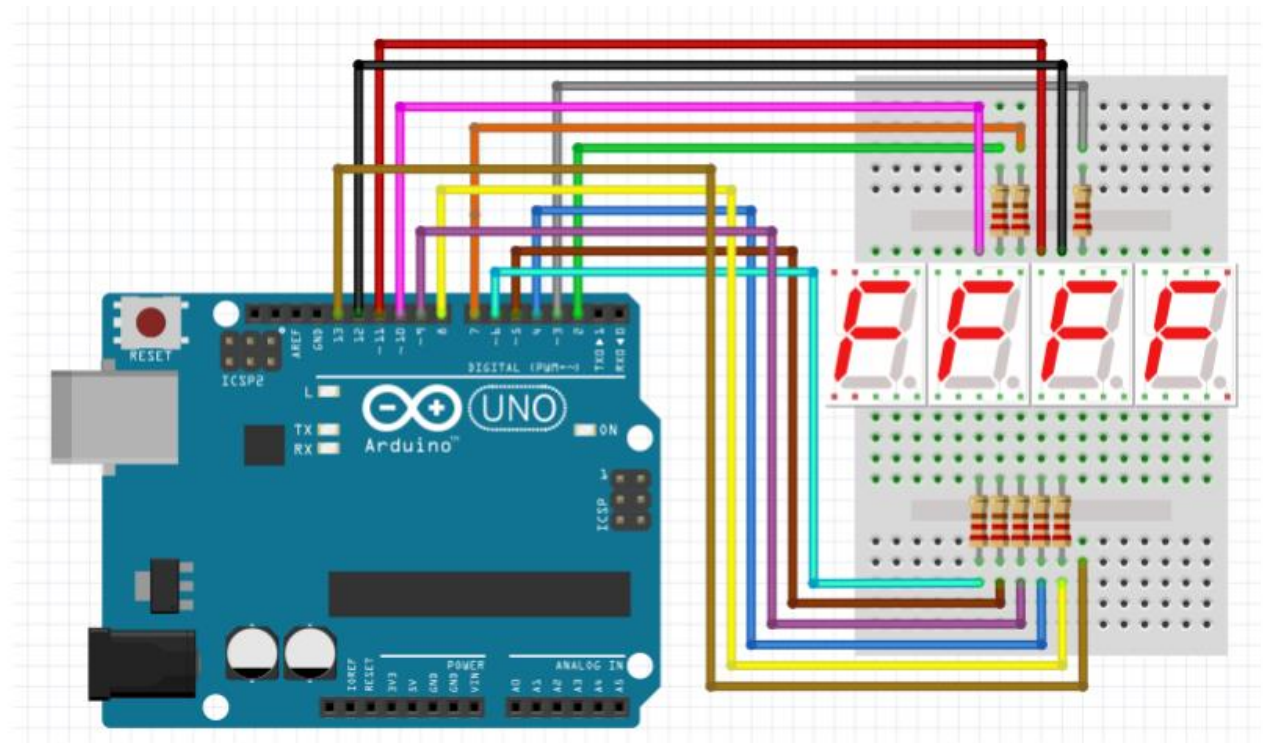
- 실험에 사용된 FND는 캐소드 제품으로 다음과 같은 핀맵을 가짐
- 1채널과 마찬가지로 a, b, c, d, e, f, g, decimal(dp)까지의 제어핀이 있으며
- 추가적으로 4채널로 동작되기 때문에 D1, D2, D3, D4의 채널 핀이 있음
- a, b, c, d, e, f, g, decimal(dp)에 저항 연결
- LOW를 주면 신호가 들어옴



- 기존 1채널 FND에서 4채널로 응용되는 과정으로 스탑워치등의 좀더 많은 표시가 가능
- 준비물
 - 아두이노 & 케이블
 - 4채널 FND(캐소드)
 - 220옴 저항
 - 브레드보드
 - 점퍼케이블



- 4채널 FND 회로 구성
 - 아두이노 - FND
 - D13 – 6
 - D12 – 8
 - D11 – 9
 - D10 – 12
 - D9 – 3
 - D8 – 5
 - D7 – 10
 - D6 – 1
 - D5 – 2
 - D4 – 4
 - D3 – 7
 - D2 – 11



```
#define SEG_A 2 //아두이노 D2 – 세그먼트 11핀(a)
#define SEG_B 3 //아두이노 D3 – 세그먼트 7핀(b)
#define SEG_C 4 //아두이노 D4 – 세그먼트 4핀(c)
#define SEG_D 5 //아두이노 D5 – 세그먼트 2핀(d)
#define SEG_E 6 //아두이노 D6 – 세그먼트 1핀(e)
#define SEG_F 7 //아두이노 D7 – 세그먼트 10핀(f)
#define SEG_G 8 //아두이노 D8 – 세그먼트 5핀(g)
#define SEG_H 9 //아두이노 D9 – 세그먼트 3핀(dp)

#define COM1 10//아두이노 D10 – 세그먼트 12핀(D1, 첫번째 칸)
#define COM1 11//아두이노 D11 – 세그먼트 9핀(D2, 두번째 칸)
#define COM1 12//아두이노 D12 – 세그먼트 8핀(D3, 세번째 칸)
#define COM1 13//아두이노 D13 – 세그먼트 6핀(D4, 네번째 칸)

//정수형 변수 테이블 10가지 8개의 수 지정
//캐소드로 {d p, g, f, e, d, c, b, a}순서
//0은 꺼짐 1일 켜짐
unsigned char table[10][8] =
```

```
{
    {0, 0, 1, 1, 1, 1, 1, 1}, //숫자 표시 0
    {0, 0, 0, 0, 0, 1, 1, 0}, //숫자 표시 1
    {0, 1, 0, 1, 1, 0, 1, 1}, //숫자 표시 2
    {0, 1, 0, 0, 1, 1, 1, 1}, //숫자 표시 3
    {0, 1, 1, 0, 0, 1, 1, 0}, //숫자 표시 4
    {0, 1, 1, 0, 1, 1, 0, 1}, //숫자 표시 5
    {0, 1, 1, 1, 1, 1, 0, 1}, //숫자 표시 6
    {0, 0, 0, 0, 0, 1, 1, 1}, //숫자 표시 7
    {0, 1, 1, 1, 1, 1, 1, 1}, //숫자 표시 8
    {0, 1, 1, 0, 1, 1, 1, 1} //숫자 표시 9
};
void setup(){
    //FND의 8개 표시 출력설정
    pinMode(SEG_A, OUTPUT);    pinMode(SEG_B, OUTPUT);
    pinMode(SEG_C, OUTPUT);    pinMode(SEG_D, OUTPUT);
    pinMode(SEG_E, OUTPUT);    pinMode(SEG_F, OUTPUT);
    pinMode(SEG_G, OUTPUT);    pinMode(SEG_H, OUTPUT);

    //FND의 자릿수 출력설정
    pinMode(COM1, OUTPUT);    pinMode(COM2, OUTPUT);
    pinMode(COM3, OUTPUT);    pinMode(COM4, OUTPUT);
}
```



```
void loop(){
    Display(1, 1); //첫번째 칸 숫자 1출력
    delay(500);
    Display(2, 2); //두번째 칸 숫자 2출력
    delay(500);
    Display(3, 3); //세번째 칸 숫자 3출력
    delay(500);
    Display(4, 4); //네번째 칸 숫자 4출력
    delay(500);
}
//디스플레이설정, com=칸, num=숫자
void Display(unsigned char com, unsigned char num) {
    digitalWrite(SEG_A, LOW);    digitalWrite(SEG_B, LOW);
    digitalWrite(SEG_C, LOW);    digitalWrite(SEG_D, LOW);
    digitalWrite(SEG_E, LOW);    digitalWrite(SEG_F, LOW);
    digitalWrite(SEG_G, LOW);    digitalWrite(SEG_H, LOW);
}
```

```
switch(com){  
  case 1:  
    digitalWrite(COM1, LOW);    digitalWrite(COM2, HIGH);  
    digitalWrite(COM3, HIGH);    digitalWrite(COM4, HIGH);  
    break;  
  case 2:  
    digitalWrite(COM1, HIGH);    digitalWrite(COM2, LOW);  
    digitalWrite(COM3, HIGH);    digitalWrite(COM4, HIGH);  
    break;  
  case 3:  
    digitalWrite(COM1, HIGH);    digitalWrite(COM2, HIGH);  
    digitalWrite(COM3, LOW);    digitalWrite(COM4, HIGH);  
    break;  
  case 4:  
    digitalWrite(COM1, HIGH);    digitalWrite(COM2, HIGH);  
    digitalWrite(COM3, HIGH);    digitalWrite(COM4, LOW);  
    break;  
  default: break;  
}
```

```
digitalWrite(SEG_A, table[num][7]); //캐소드..  
digitalWrite(SEG_B, table[num][6]);  
digitalWrite(SEG_C, table[num][5]);  
digitalWrite(SEG_D, table[num][4]);  
digitalWrite(SEG_E, table[num][3]);  
digitalWrite(SEG_F, table[num][2]);  
digitalWrite(SEG_G, table[num][1]);  
digitalWrite(SEG_H, table[num][0]);  
}
```

응용실습

- 동시에 4자리 숫자가 유지될수있게 코드를 변경해보자