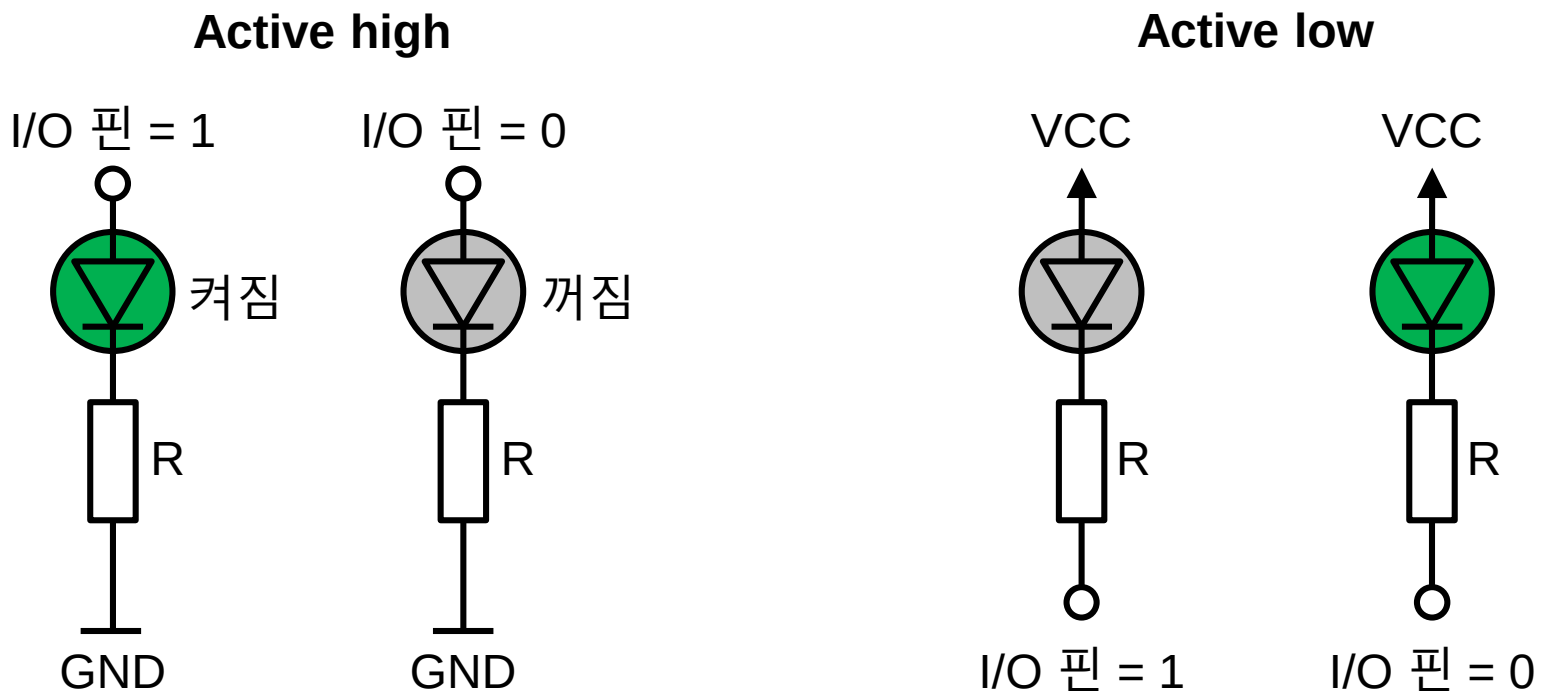
Lecture 04

FND 제어

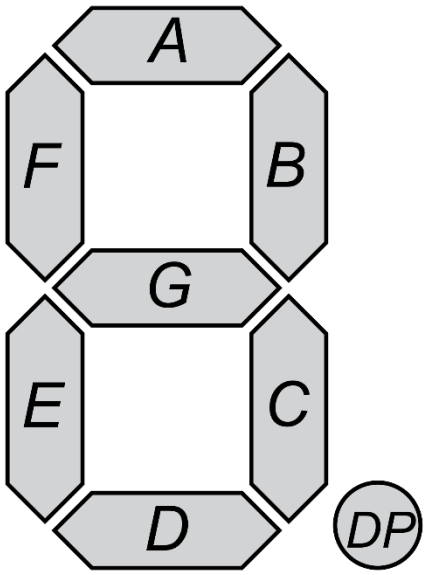
LED 제어

- LED 제어 방법

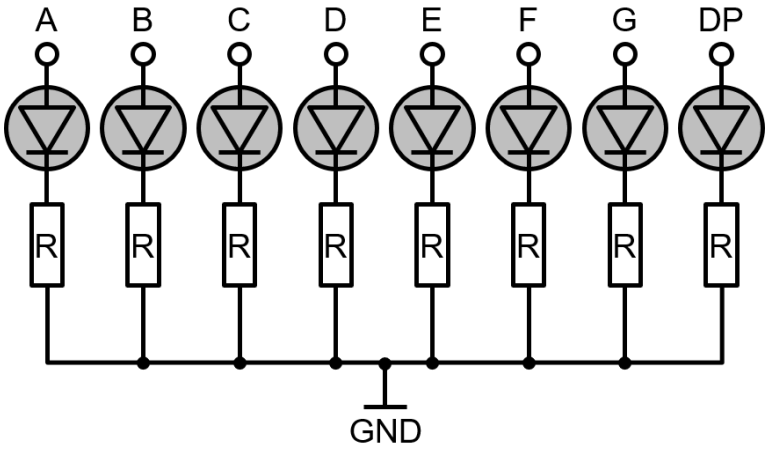


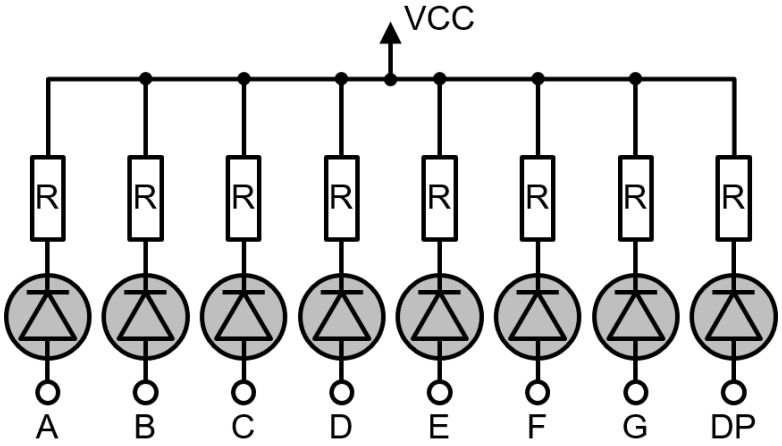
FND (Flexible Numeric Display)

- FND (또는 7-segment LED)



Active high 또는 common cathode (CC)





Active low 또는 common anode (CA)

FND (Flexible Numeric Display)

- FND 제어 방법

숫자	Active high (또는 CC)								16진수
	DP	G	F	E	D	C	B	A	
0	0	0	1	1	1	1	1	1	0x3f
1	0	0	0	0	0	1	1	0	0x06
2	0	1	0	1	1	0	1	1	0x5b
3	0	1	0	0	1	1	1	1	0x4f
4	0	1	1	0	0	1	1	0	0x66
5	0	1	1	0	1	1	0	1	0x6d
6	0	1	1	1	1	1	0	1	0x7d
7	0	0	1	0	0	1	1	1	0x27
8	0	1	1	1	1	1	1	1	0x7f
9	0	1	1	0	0	1	1	1	0x67

숫자	Active high (또는 CC)								16진수
	DP	G	F	E	D	C	B	A	
a	0	1	1	1	0	1	1	1	0x77
b	0	1	1	1	1	1	0	0	0x7c
c	0	1	0	1	1	0	0	0	0x58
d	0	1	0	1	1	1	1	0	0x5e
e	0	1	1	1	1	0	0	1	0x79
f	0	1	1	1	0	0	0	1	0x71

FND (Flexible Numeric Display)

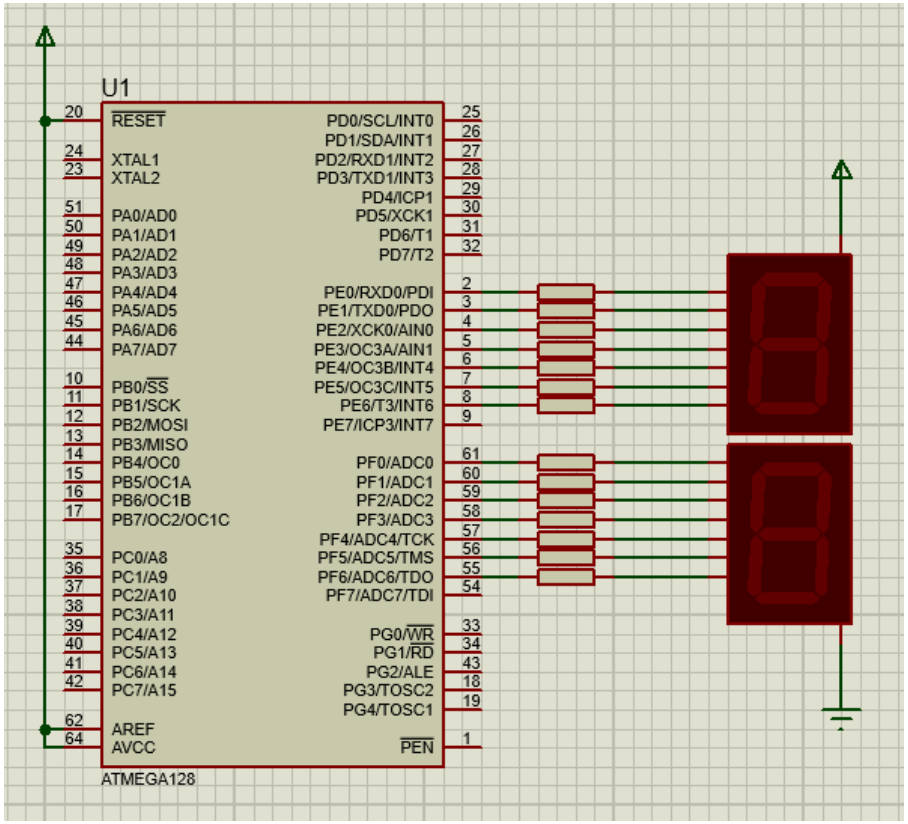
- FND 제어 방법

숫자	Active low (또는 CA)								16진수
	DP	G	F	E	D	C	B	A	
0	1	1	0	0	0	0	0	0	0xc0
1	1	1	1	1	1	0	0	1	0xf9
2	1	0	1	0	0	1	0	0	0xa4
3	1	0	1	1	0	0	0	0	0xb0
4	1	0	0	1	1	0	0	1	0x99
5	1	0	0	1	0	0	1	0	0x92
6	1	0	0	0	0	0	1	0	0x82
7	1	1	0	1	1	0	0	0	0xd8
8	1	0	0	0	0	0	0	0	0x80
9	1	0	0	1	1	0	0	0	0x98

숫자	Active low (또는 CA)								16진수
	DP	G	F	E	D	C	B	A	
a	1	0	0	0	1	0	0	0	0x88
b	1	0	0	0	0	0	1	1	0x83
c	1	0	1	0	0	1	1	1	0xa7
d	1	0	1	0	0	0	0	1	0xa1
e	1	0	0	0	0	1	1	0	0x86
f	1	0	0	0	1	1	1	0	0x8e

FND (Flexible Numeric Display)

- CC타입 및 CA타입 FND 실습



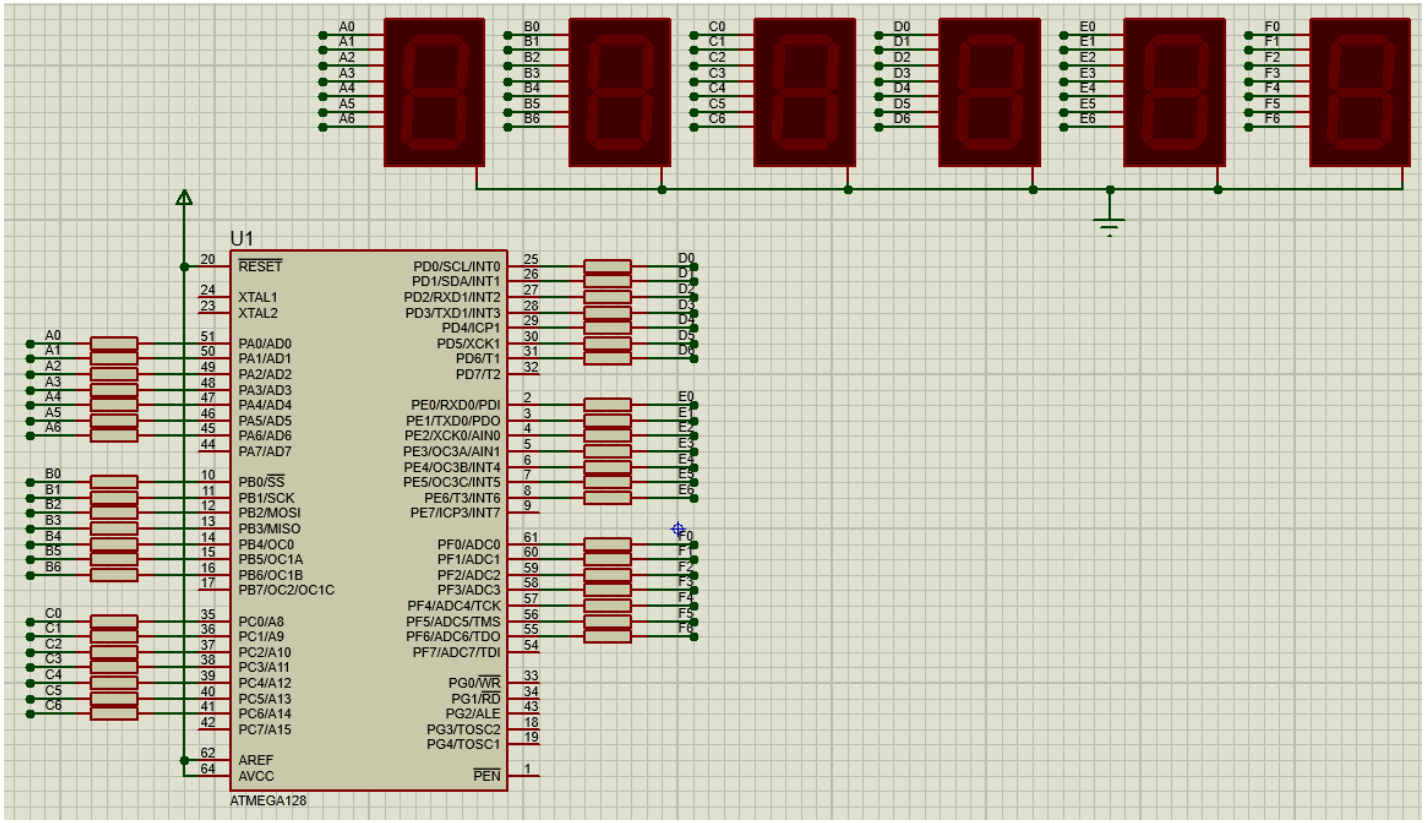
FND (Flexible Numeric Display)

- CC타입 및 CA타입 FND 실습

```
unsigned char CC[] = {0x3f, 0x06, 0x5b, 0x4f, 0x66, 0x6d, 0x7d,
0x27, 0x7f, 0x67, 0x77, 0x7c, 0x58, 0x5e, 0x79, 0x71, 0x80};
unsigned char CA[] = {0xc0, 0xf9, 0xa4, 0xb0, 0x99, 0x92, 0x82,
0xd8, 0x80, 0x98, 0x88, 0x83, 0xa7, 0xa1, 0x86, 0x8e, 0x7f};

int main(void) {
   DDRE = DDRF = 0xff;
   int i;
   while(1) {
      for (i=0; i<16; i++) {
         PORTE = CA[i];
         PORTF = CC[i];
         delay(DELAY_TIME);
      }
   }
}
```

FND (Flexible Numeric Display)

- 디지털 시계 실습



FND (Flexible Numeric Display)

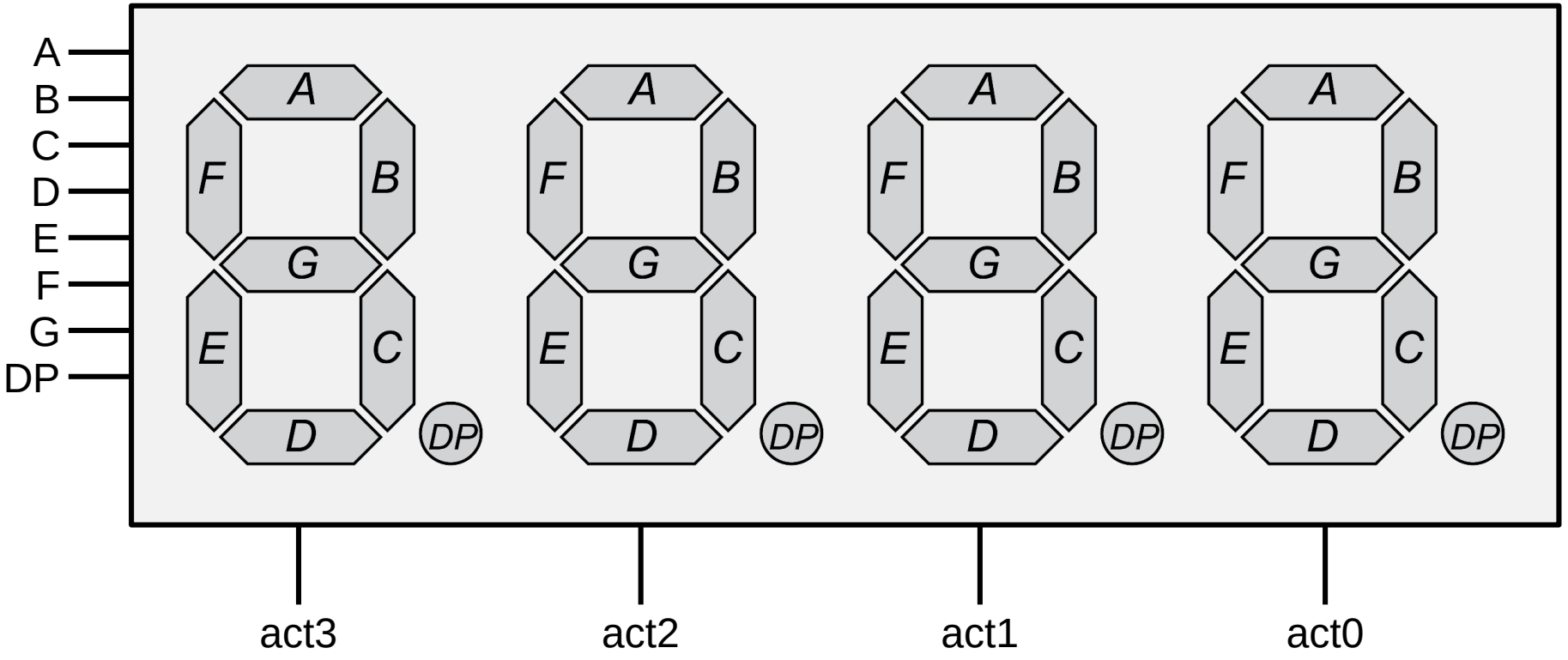
- 디지털 시계 실습

```
void delay(int);
void display(void);
void time_process(void);

unsigned char CC[] = {0x3f, 0x06, 0x5b, 0x4f, 0x66, 0x6d, 0x7d,
0x27, 0x7f, 0x67, 0x77, 0x7c, 0x58, 0x5e, 0x79, 0x71, 0x80};
unsigned char CA[] = {0xc0, 0xf9, 0xa4, 0xb0, 0x99, 0x92, 0x82,
0xd8, 0x80, 0x98, 0x88, 0x83, 0xa7, 0xa1, 0x86, 0x8e, 0x7f};
unsigned char h, m, s;

int main(void) {
   DDRA = DDRB = DDRC = DDRD = DDRE = DDRF = 0xff;
   h = m = s = 0;
   while(1) {
      display();
      delay(DELAY_TIME);
      time_process();
   }
}
```

```
void display(void) {
   PORTA = CC[h/10];
   PORTB = CC[h%10];
   PORTC = CC[m/10];
   PORTD = CC[m%10];
   PORTE = CC[s/10];
   PORTF = CC[s%10];
}

void time_process(void) {
   s++;
   if (s>59) {
      s = 0;
      m++;
      if (m>59) {
         m = 0;
         h++;
         if (h>12) h = 1;
      }
   }
}
```

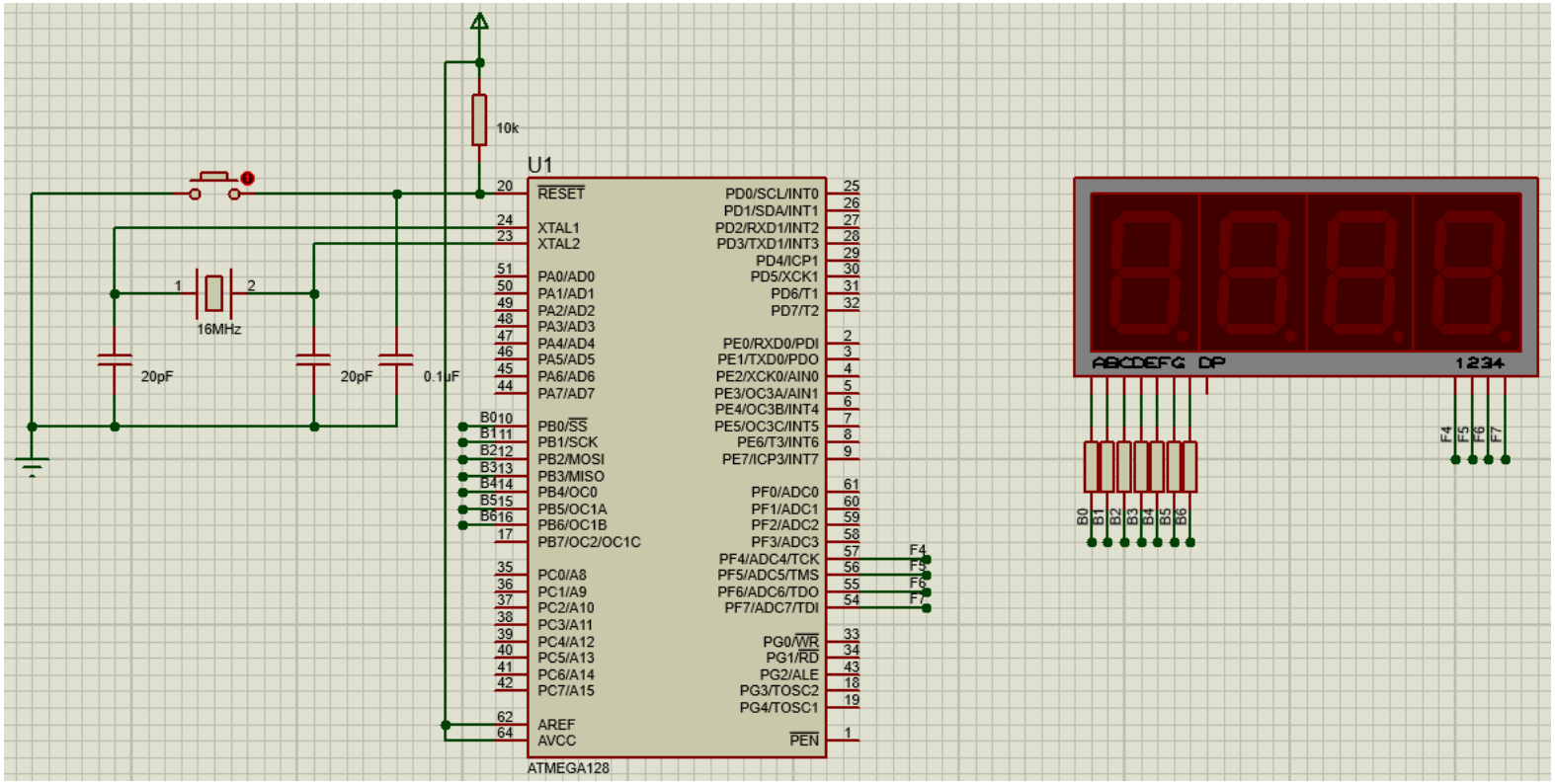
FND (Flexible Numeric Display)

- FND 4개



FND (Flexible Numeric Display)

- FND 4개 실습



FND (Flexible Numeric Display)

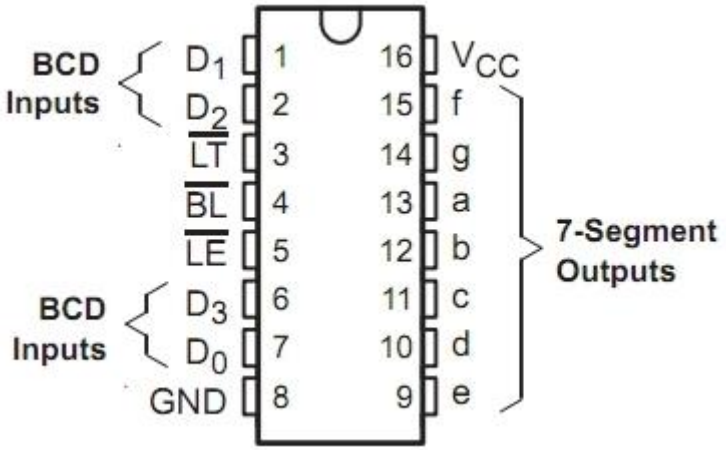
- FND 4개 실습

```
unsigned char CC[] = {0x3f, 0x06, 0x5b, 0x4f, 0x66, 0x6d, 0x7d, 0x27, 0x7f, 0x67, 0x77,
0x7c, 0x58, 0x5e, 0x79, 0x71, 0x80};
unsigned char CA[] = {0xc0, 0xf9, 0xa4, 0xb0, 0x99, 0x92, 0x82, 0xd8, 0x80, 0x98, 0x88,
0x83, 0xa7, 0xa1, 0x86, 0x8e, 0x7f};
unsigned char act[4] = {0xef, 0xdf, 0xbf, 0x7f};

int main(void) {
   DDRB = DDRF = 0xff;
   int i, j;
   while(1) {
      for (i=0; i<4; i++) {
         for (j=0; j<16; j++) {
            PORTF = act[i];
            PORTB = CC[j]|0x80;
            delay(DELAY_TIME);
         }
      }
   }
}
```

FND (Flexible Numeric Display)

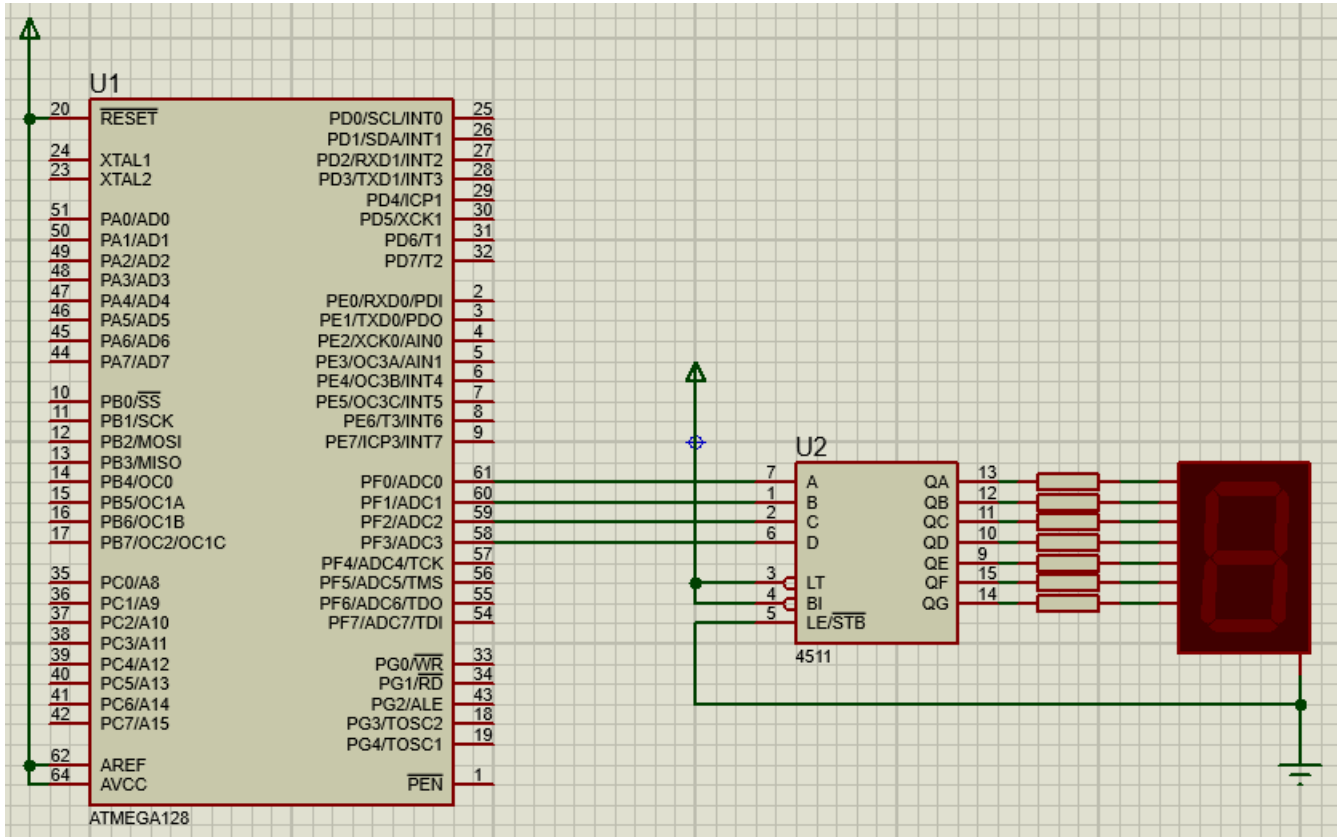
- CD4511: BCD-to-7-Segment Decoder

숫자	Active high (또는 CC)								16진수
	DP	G	F	E	D	C	B	A	
0	0	0	1	1	1	1	1	1	0x3f
1	0	0	0	0	0	1	1	0	0x06
2	0	1	0	1	1	0	1	1	0x5b
3	0	1	0	0	1	1	1	1	0x4f
4	0	1	1	0	0	1	1	0	0x66
5	0	1	1	0	1	1	0	1	0x6d
6	0	1	1	1	1	1	0	1	0x7d
7	0	0	1	0	0	1	1	1	0x27
8	0	1	1	1	1	1	1	1	0x7f
9	0	1	1	0	0	1	1	1	0x67



FND (Flexible Numeric Display)

- CD4511: BCD-to-7-Segment Decoder



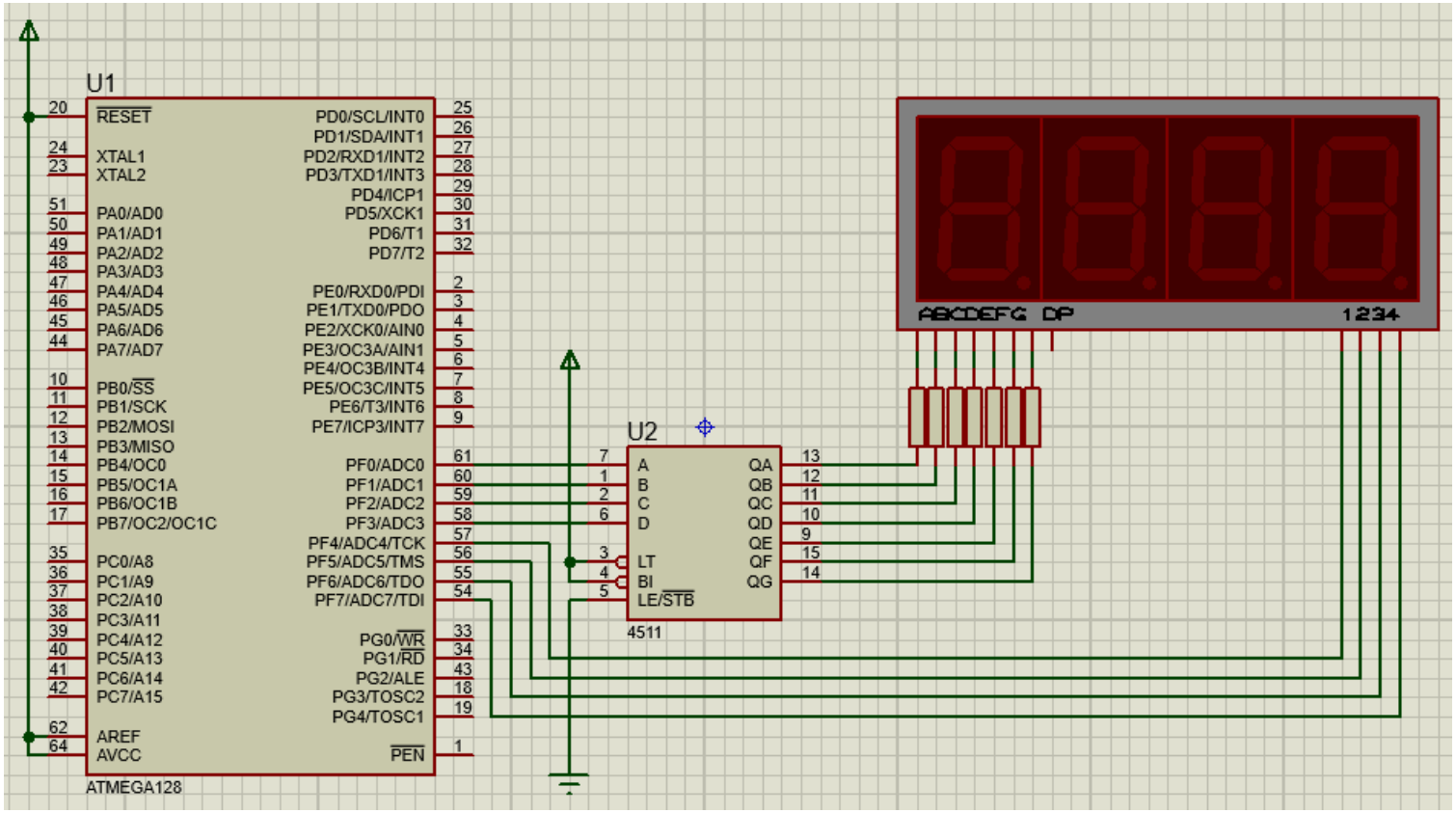
FND (Flexible Numeric Display)

- CD4511: BCD-to-7-Segment Decoder

```
int main(void) {
    int i = 0;
    DDRF = 0xff;
    while(1) {
        PORTF = i;
        delay(DELAY_TIME);
        i++;
        if (i>9) i = 0;
    }
}
```

FND (Flexible Numeric Display)

- CD4511: BCD-to-7-Segment Decoder



FND (Flexible Numeric Display)

- CD4511: BCD-to-7-Segment Decoder

```
unsigned char act[4] = {0xef, 0xdf, 0xbf, 0x7f};

int main(void) {
   int i, j;
   DDRF = 0xff;
   while(1) {
      for (i=0; i<4; i++) {
         for (j=0; j<10; j++) {
            PORTF = (j&0x0f) | (act[i]&0xf0);
            delay(DELAY_TIME);
         }
      }
   }
}
```