2023년 IoT기반 스마트 솔루션 개발자 양성과정

Embedded Application

7-Switch Input

담당 교수 : 윤 종 이
010-9577-1696
ojo1696@naver.com
https://cafe.naver.com/yoons2023

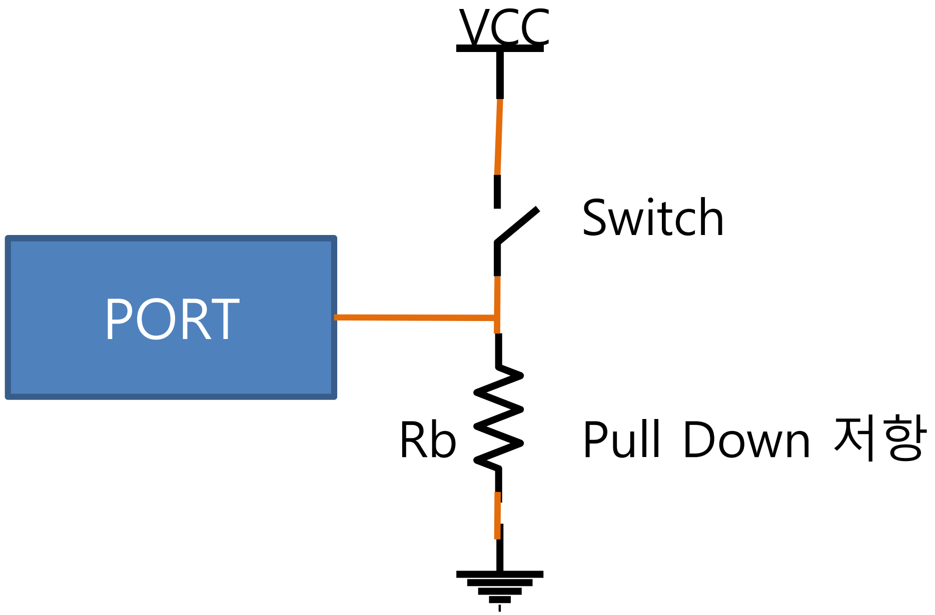
충북대학교 공동훈련센터

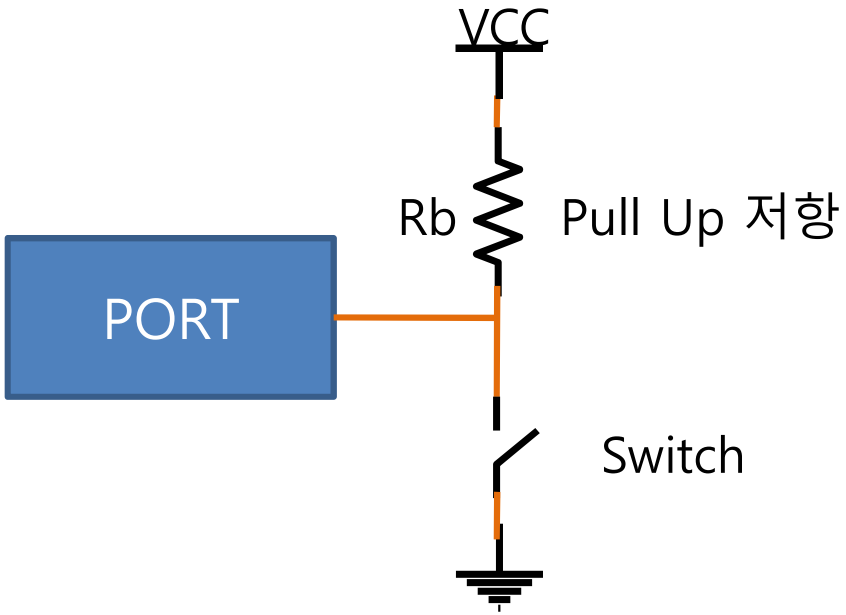
스위치

- 전류의 흐름을 개폐함(On/OFF)
- 신호의 입력용으로 사용(Tact Switch)
- Off시 입력 논리값을 명확히 함
  - Pull Up / Pull Down 저항을 함께 연결

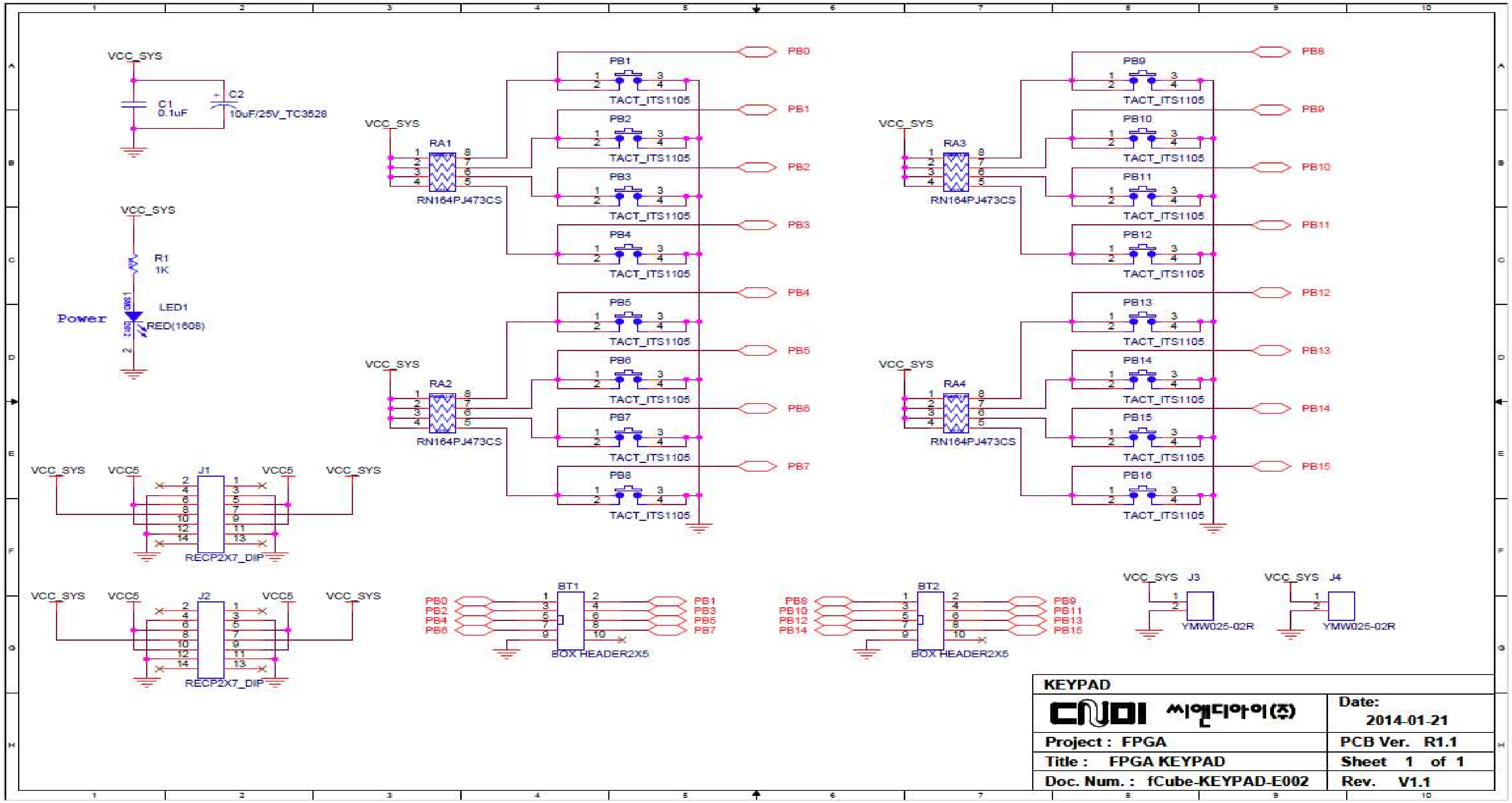
스위치의 사용

- 정논리 입력 : 스위치를 누르면 1(High Voltage)
- 부논리 입력 : 스위치를 누르면 0(Low Voltage)
- Pull Up/ Pull Down 저항 : 4.7K~10KΩ





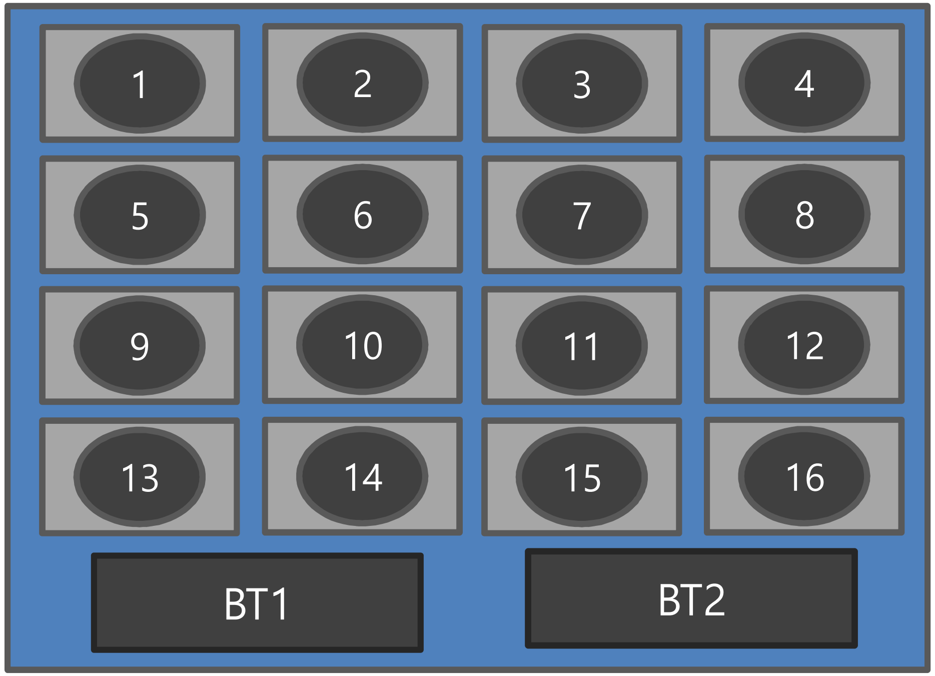
Keypad Schematic



충북대학교 공동훈련센터

Keypad Layout

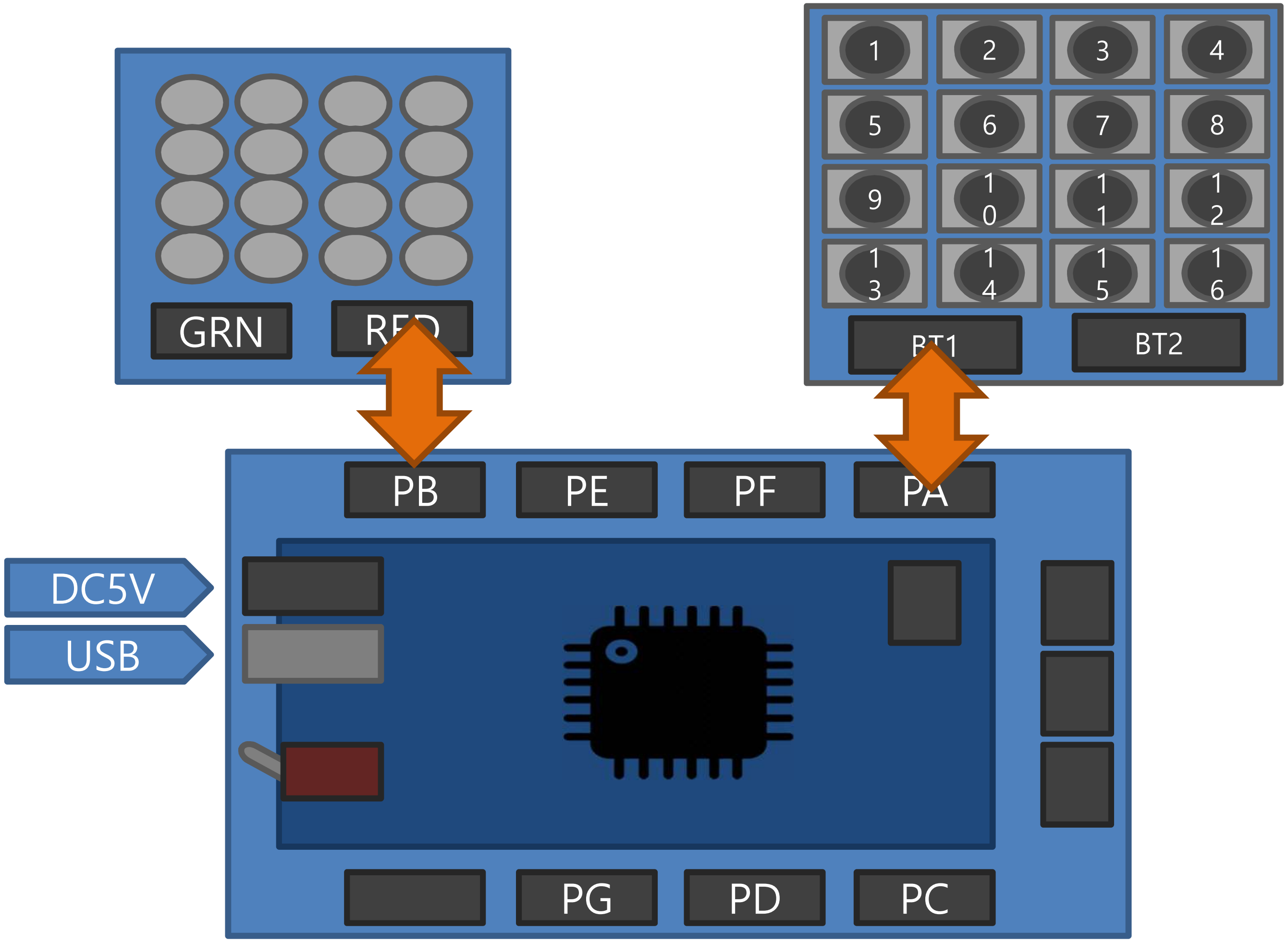
BT1 : SW1~SW8
BT2 : SW9~SW16



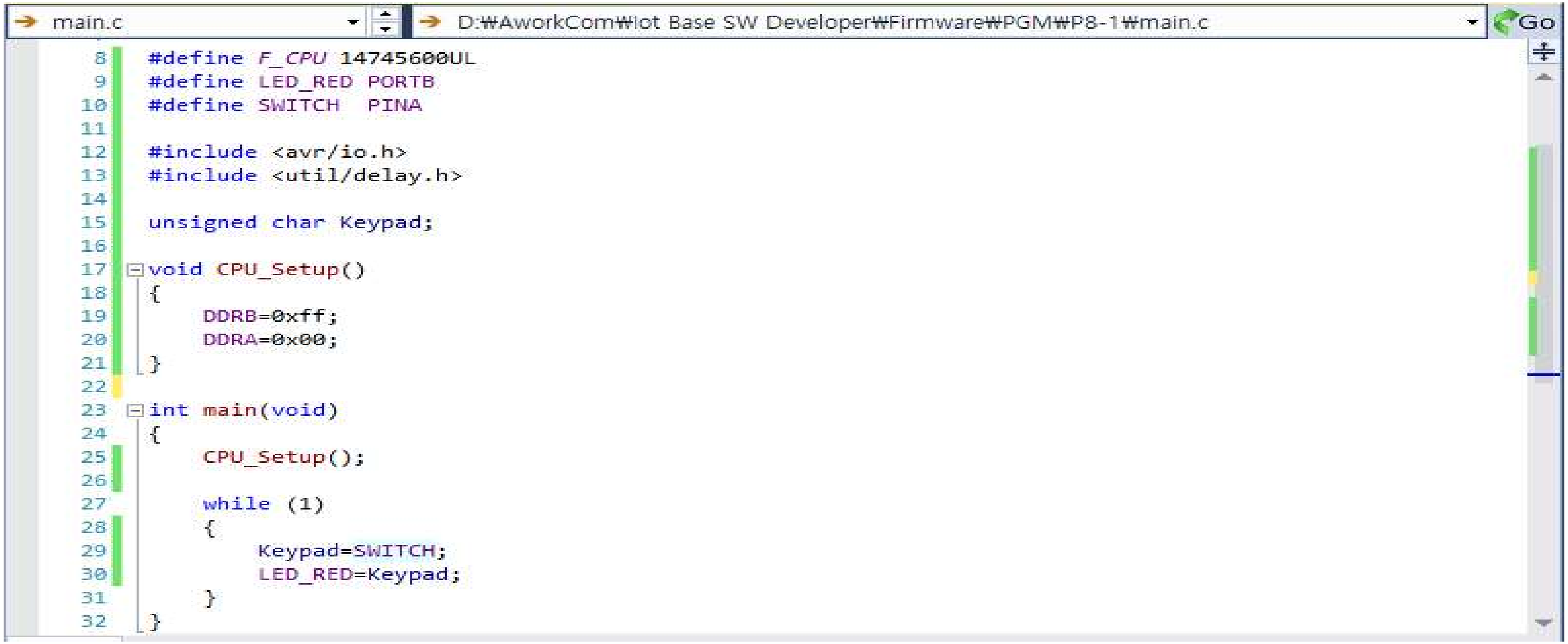
1	2	3	4
5	6	7	8
9	10	11	12
13	14	15	16

BT1 BT2

Ex-1 : Wiring

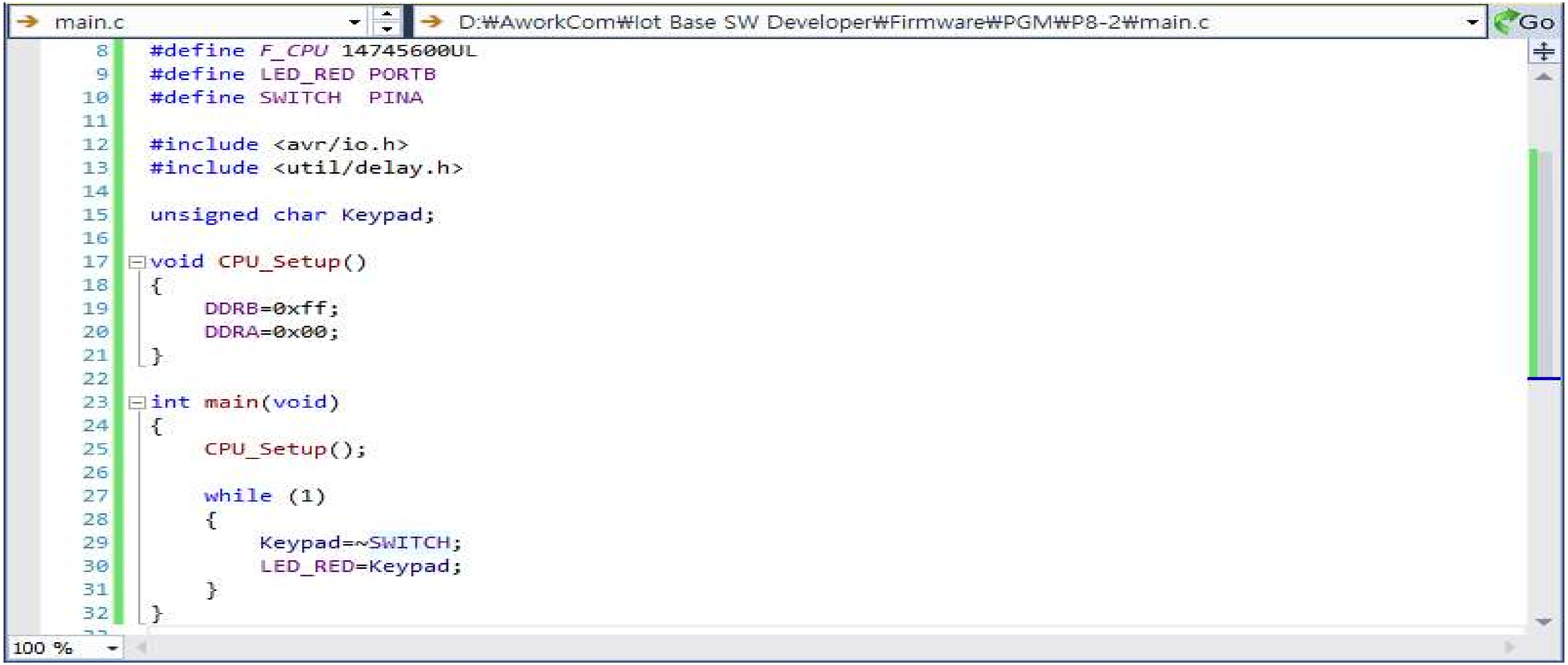

GRN
RED
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
BT1
BT2
PB
PE
PF
PA
DC5V
USB
PG
PD
PC

Ex-1 : Switch Input(부논리)



```c
#define F_CPU 14745600UL
#define LED_RED PORTB
#define SWITCH  PINA

#include <avr/io.h>
#include <util/delay.h>

unsigned char Keypad;

void CPU_Setup()
{
    DDRB=0xff;
    DDRA=0x00;
}

int main(void)
{
    CPU_Setup();

    while (1)
    {
        Keypad=SWITCH;
        LED_RED=Keypad;
    }
}
```

Ex-2 : Switch Input(정논리)



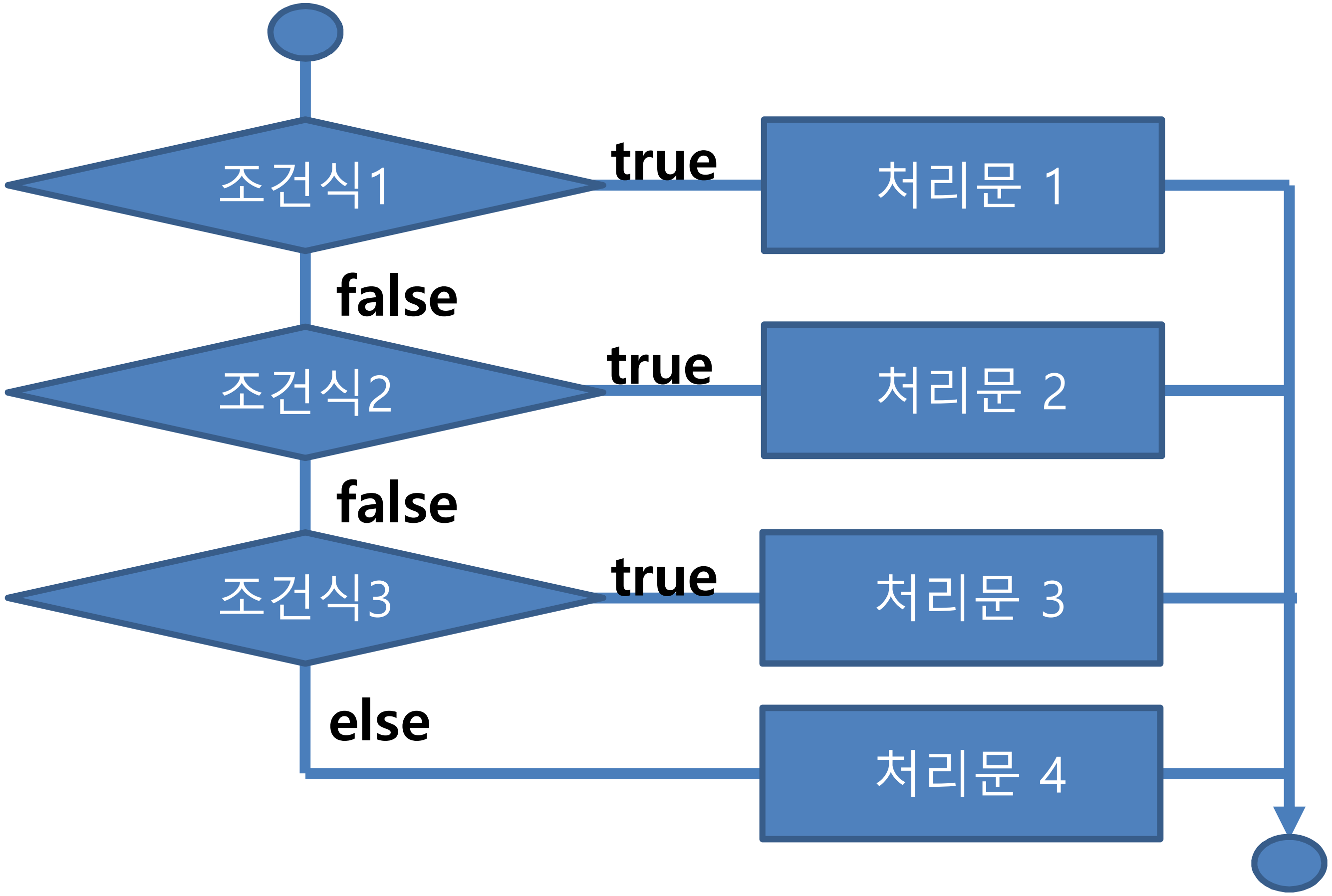
```c
#define F_CPU 14745600UL
#define LED_RED PORTB
#define SWITCH  PINA

#include <avr/io.h>
#include <util/delay.h>

unsigned char Keypad;

void CPU_Setup()
{
    DDRB=0xff;
    DDRA=0x00;
}

int main(void)
{
    CPU_Setup();

    while (1)
    {
        Keypad=~SWITCH;
        LED_RED=Keypad;
    }
}
```

if 문(비교문)

- 조건에 따라 다른 문장을 실행

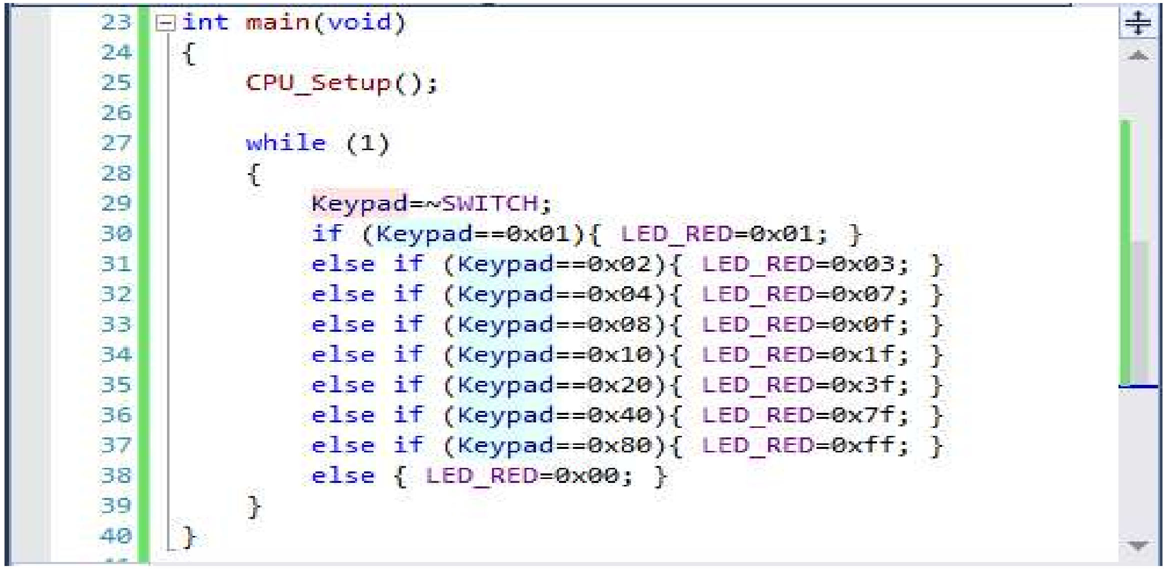
```
if (조건식1) {
        처리문 1
}
else if (조건식2) {
         처리문 2
}
else if (조건식3) {
         처리문 3
}
else {
         처리문 4
}
```

if 문 플로우차트


조건식1
true
처리문 1
false
조건식2
true
처리문 2
false
조건식3
true
처리문 3
else
처리문 4

Ex-3 : Switch Input(if~)

- 스위치 1번-> 1번 LED
- 스위치 2번-> 1,2번 LED
- 스위치 3번-> 1,2,3번 LED
- 스위치 4번-> 1,2,3,4번 LED
- 스위치 5번-> 1,2,3,4,5번 LED
- 스위치 6번-> 1,2,3,4,5,6번 LED
- 스위치 7번-> 1,2,3,4,5,6,7번 LED
- 스위치 8번-> 1,2,3,4,5,6,7,8번 LED
- 기타 -> LED OFF



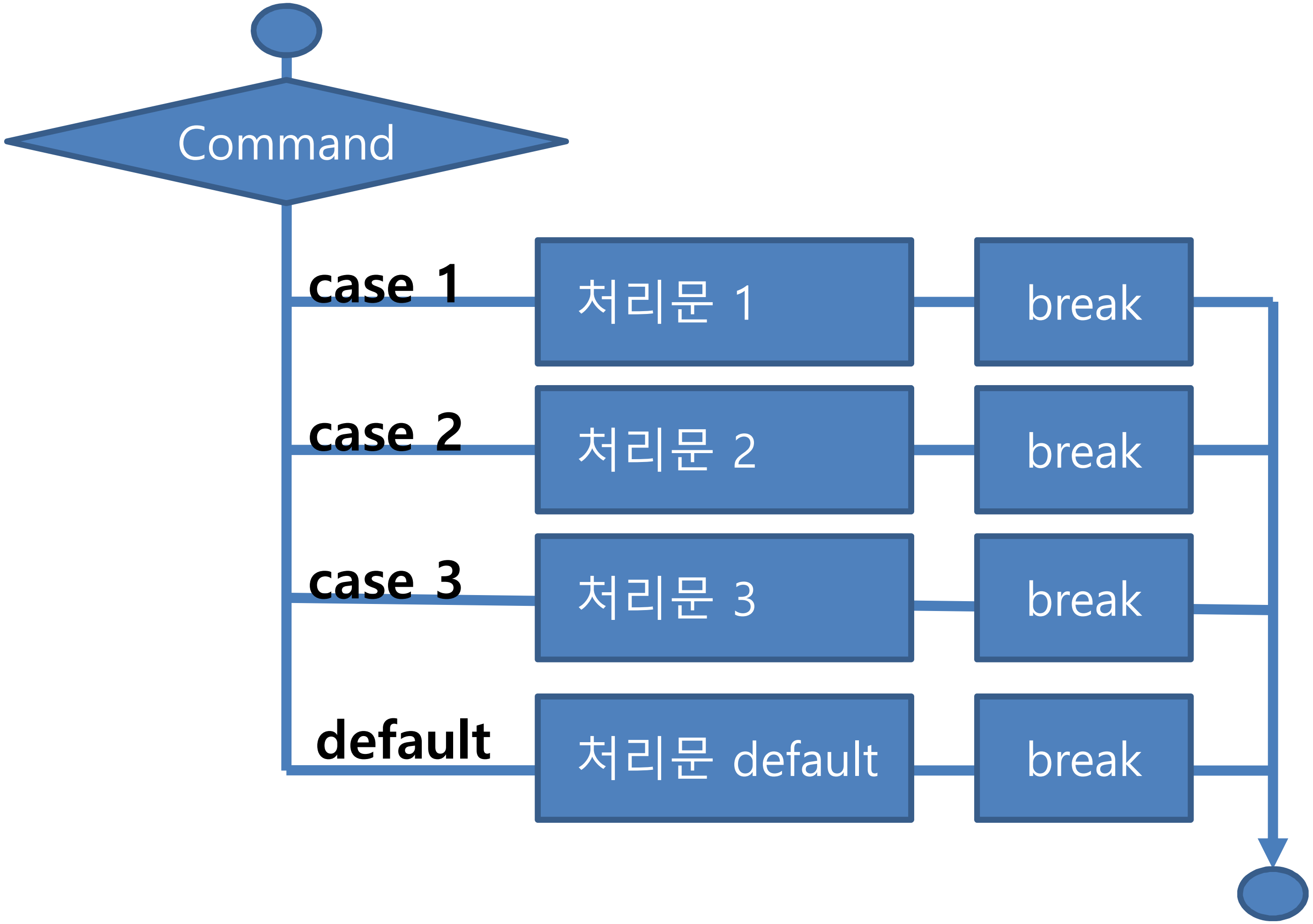
```
int main(void)
{
    CPU_Setup();

    while (1)
    {
        Keypad=~SWITCH;
        if (Keypad==0x01){ LED_RED=0x01; }
        else if (Keypad==0x02){ LED_RED=0x03; }
        else if (Keypad==0x04){ LED_RED=0x07; }
        else if (Keypad==0x08){ LED_RED=0x0f; }
        else if (Keypad==0x10){ LED_RED=0x1f; }
        else if (Keypad==0x20){ LED_RED=0x3f; }
        else if (Keypad==0x40){ LED_RED=0x7f; }
        else if (Keypad==0x80){ LED_RED=0xff; }
        else { LED_RED=0x00; }
    }
}
```

switch 문(분기문)

- 조건에 따라 다른 문장을 실행

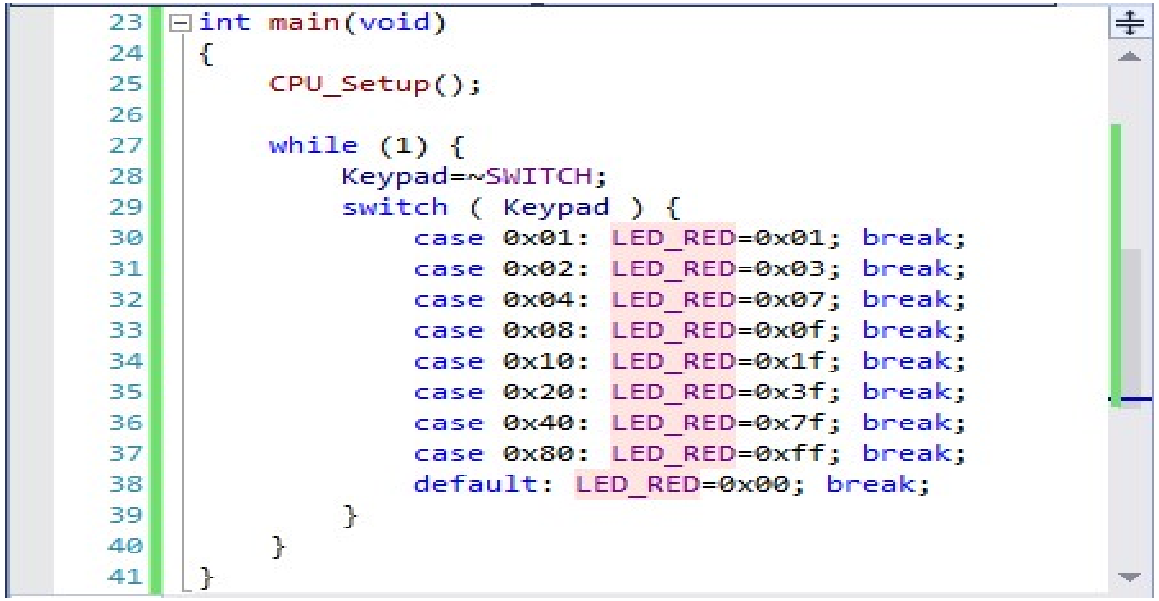
```
switch ( 조건 ) {
case 1:
      //Exp가 1 이면
      break;
case 2:
      //Exp가 2 이면
      break;
case 3:
      //Exp가 3 이면
      break;
default:
      //그 외에
      break;
}
```

switch 문 플로우차트



Ex-4 : Switch Input(switch~)

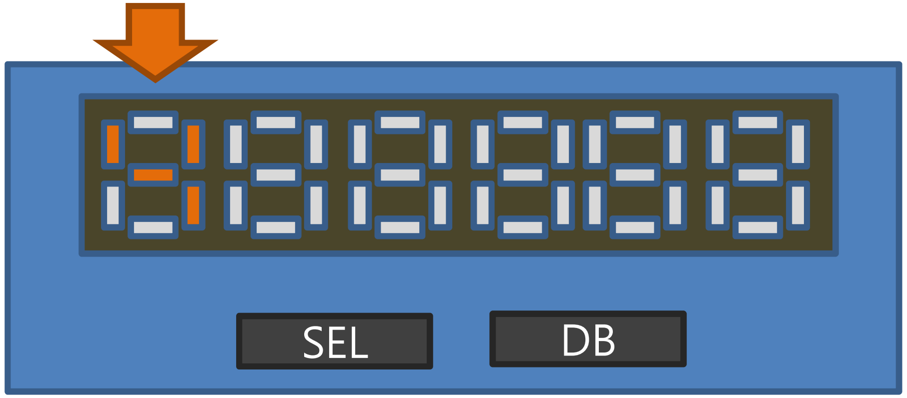
- 스위치 1번-> 1번 LED
- 스위치 2번-> 1,2번 LED
- 스위치 3번-> 1,2,3번 LED
- 스위치 4번-> 1,2,3,4번 LED
- 스위치 5번-> 1,2,3,4,5번 LED
- 스위치 6번-> 1,2,3,4,5,6번 LED
- 스위치 7번-> 1,2,3,4,5,6,7번 LED
- 스위치 8번-> 1,2,3,4,5,6,7,8번 LED
- 기타 -> LED OFF

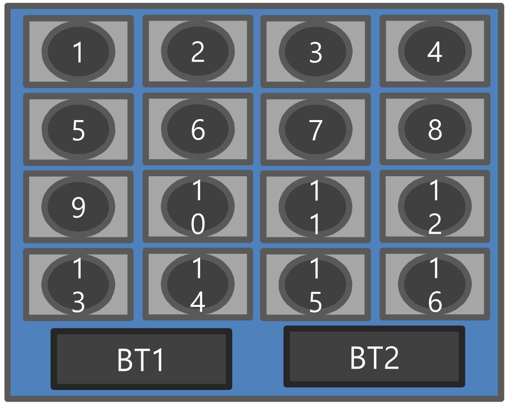


```
int main(void)
{
        CPU_Setup();

        while (1) {
                Keypad=~SWITCH;
                switch ( Keypad ) {
                        case 0x01: LED_RED=0x01; break;
                        case 0x02: LED_RED=0x03; break;
                        case 0x04: LED_RED=0x07; break;
                        case 0x08: LED_RED=0x0f; break;
                        case 0x10: LED_RED=0x1f; break;
                        case 0x20: LED_RED=0x3f; break;
                        case 0x40: LED_RED=0x7f; break;
                        case 0x80: LED_RED=0xff; break;
                        default: LED_RED=0x00; break;
                }
        }
}
```

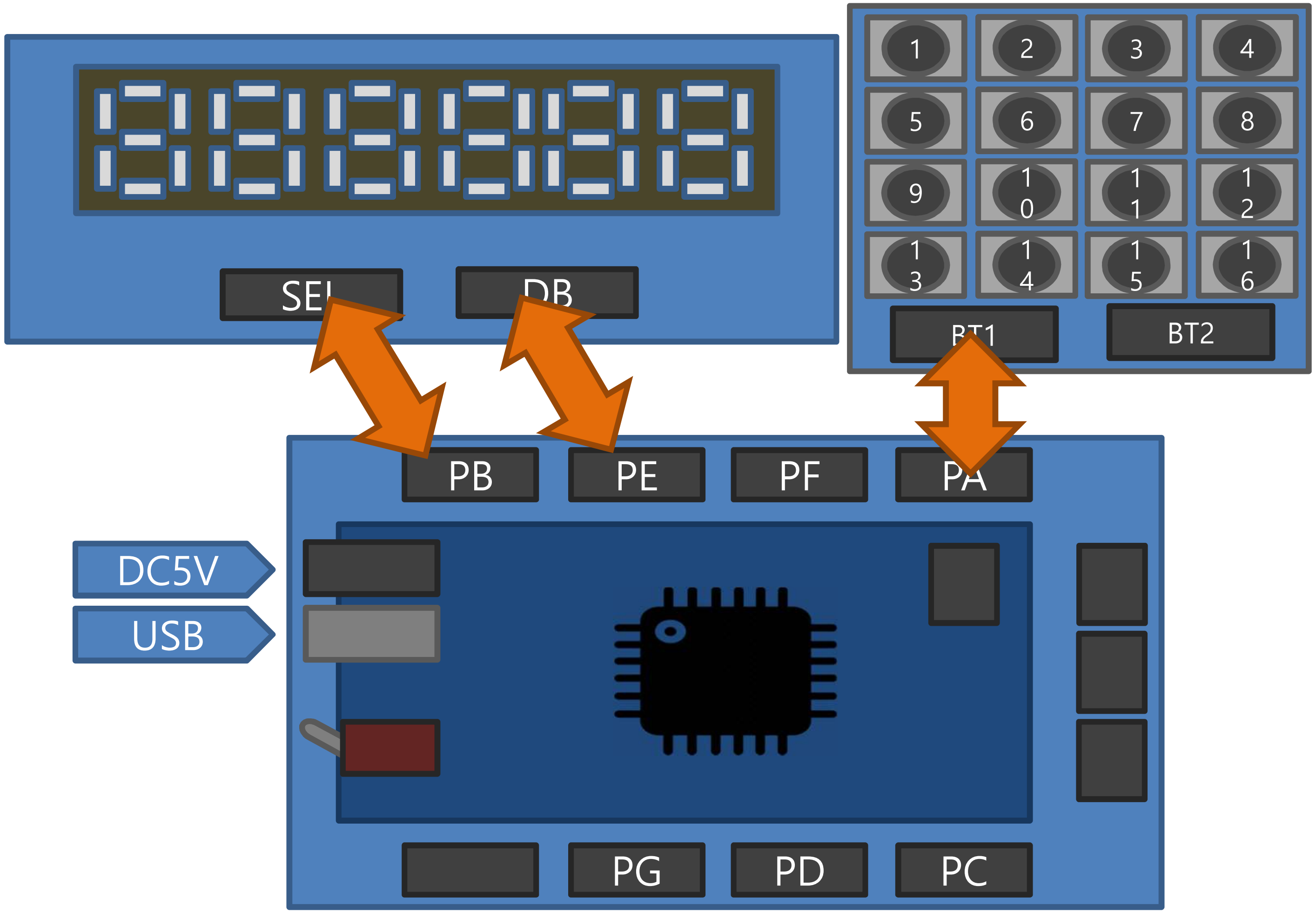
Ex-5 : 버튼 숫자 Display

- 스위치 1번-> 0
- 스위치 2번-> 1
- 스위치 3번-> 2
- 스위치 4번-> 3
- 스위치 5번-> 4
- 스위치 6번-> 5
- 스위치 7번-> 6
- 스위치 8번-> 7
- 기타 -> -





Ex-5 : Wiring


SEL
DB
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
BT1
BT2
PB
PE
PF
PA
DC5V
USB
PG
PD
PC

Ex-5 : Program-define

```
#define F_CPU 14745600UL
#define FND_SEL PORTB
#define FND_DB PORTE
#define SWITCH PINA

#include <avr/io.h>

unsigned char FND[11]={0x3f, 0x06, 0x5b, 0x4f, 0x66, 0x6d, 0x7d, 0x27, 0x7f, 0x6f, 0x40};
unsigned char DGT[6]={0xfe, 0xfd, 0xfb, 0xf7, 0xef, 0xdf};
unsigned char Keypad;

unsigned long Count=0;

void CPU_Setup( ) {
    DDRB=0xff;
    DDRE=0xff;
    DDRA=0x00;
}
```

Ex-5 : Program-main

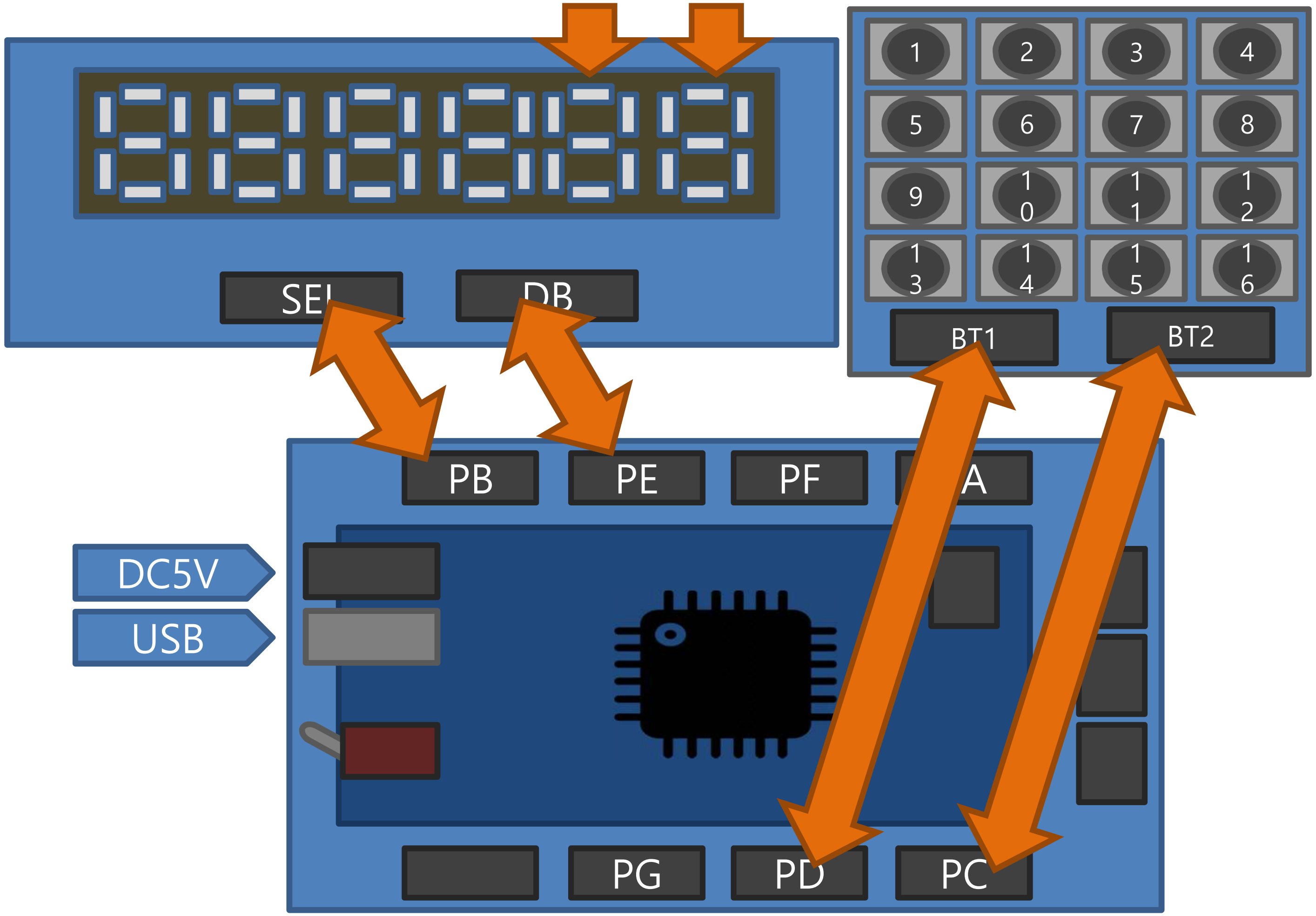
```c
int main(void) {
    CPU_Setup( );

    while (1) {
        Keypad=~SWITCH;
        FND_SEL=DGT[5];
        switch ( Keypad ) {
            case 0x01: FND_DB=FND[0]; break;
            case 0x02: FND_DB=FND[1]; break;
            case 0x04: FND_DB=FND[2]; break;
            case 0x08: FND_DB=FND[3]; break;
            case 0x10: FND_DB=FND[4]; break;
            case 0x20: FND_DB=FND[5]; break;
            case 0x40: FND_DB=FND[6]; break;
            case 0x80: FND_DB=FND[7]; break;
            default: FND_DB=FND[10]; break;
        }
    }
}
```

Ex-6 : sub function

```
char SwitchIn(void) {
       char KeyNo;
       unsigned char KeyIn =~SWITCH

       switch ( KeyIn ) {
              case 0x01: KeyNo =0;   break;
              case 0x02: KeyNo =1;   break;
              case 0x04: KeyNo =2;   break;
              case 0x08: KeyNo =3;   break;
              case 0x10: KeyNo =4;   break;
              case 0x20: KeyNo =5;   break;
              case 0x40: KeyNo =6;   break;
              case 0x80: KeyNo =7;   break;
              default: KeyNo =10;   break;
       }
      return KeyNo;
}
```

```
int main(void) {
    CPU_Setup( );

    while (1) {
        Keypad=SwitchIn( );
        FND_SEL=DGT[5];
        FND_DB=FND[Keypad];
    }
}
```

Ex-7 : 16key Input



Ex-7 : Program-define

```
#define F_CPU 14745600UL
#define FND_SEL PORTB
#define FND_DB PORTE
#define SWITCH1 PIND
#define SWITCH2 PINC
#define dTime 3

#include <avr/io.h>
#include <util/delay.h>

unsigned char FND[11]={0x3f, 0x06, 0x5b, 0x4f, 0x66, 0x6d, 0x7d, 0x27, 0x7f, 0x6f, 0x40};
unsigned char DGT[6]={0xfe, 0xfd, 0xfb, 0xf7, 0xef, 0xdf};
unsigned char NUM[6]={0x00, 0x00, 0x00, 0x00, 0x00, 0x00};
unsigned char Keypad;

void CPU_Setup( ) {
    DDRB=0xff;
    DDRE=0xff;
    DDRC=0x00;
    DDRD=0x00;
}
```

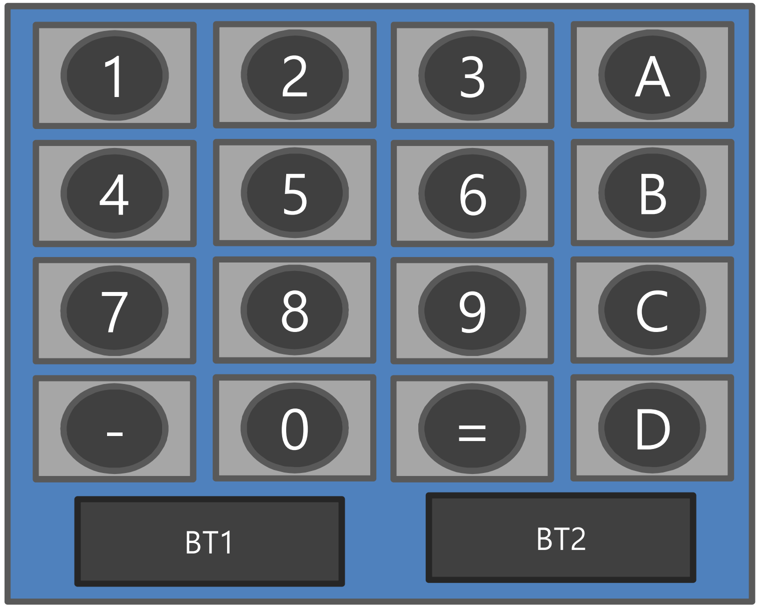
Ex-7 : sub function

```
char SwitchIn(void) {
        char KeyNo;
        unsigned int KeyIn =(~SWITCH2<<8) | (~SWITCH1 & 0x00ff);

        switch ( KeyIn ) {
               case 0x0001: KeyNo =0;  break;
               case 0x0002: KeyNo =1;  break;
               case 0x0004: KeyNo =2;  break;
               case 0x0008: KeyNo =3;  break;
               case 0x0010: KeyNo =4;  break;
               case 0x0020: KeyNo =5;  break;
               case 0x0040: KeyNo =6;  break;
               case 0x0080: KeyNo =7;  break;
               case 0x0100: KeyNo =8;  break;
               case 0x0200: KeyNo =9;  break;
               case 0x0400: KeyNo =10;  break;
               case 0x0800: KeyNo =11;  break;
               case 0x1000: KeyNo =12;  break;
               case 0x2000: KeyNo =13;  break;
               case 0x4000: KeyNo =14;  break;
               case 0x8000: KeyNo =15;  break;
               default: KeyNo =16;  break;
        }
       return KeyNo;
}
```

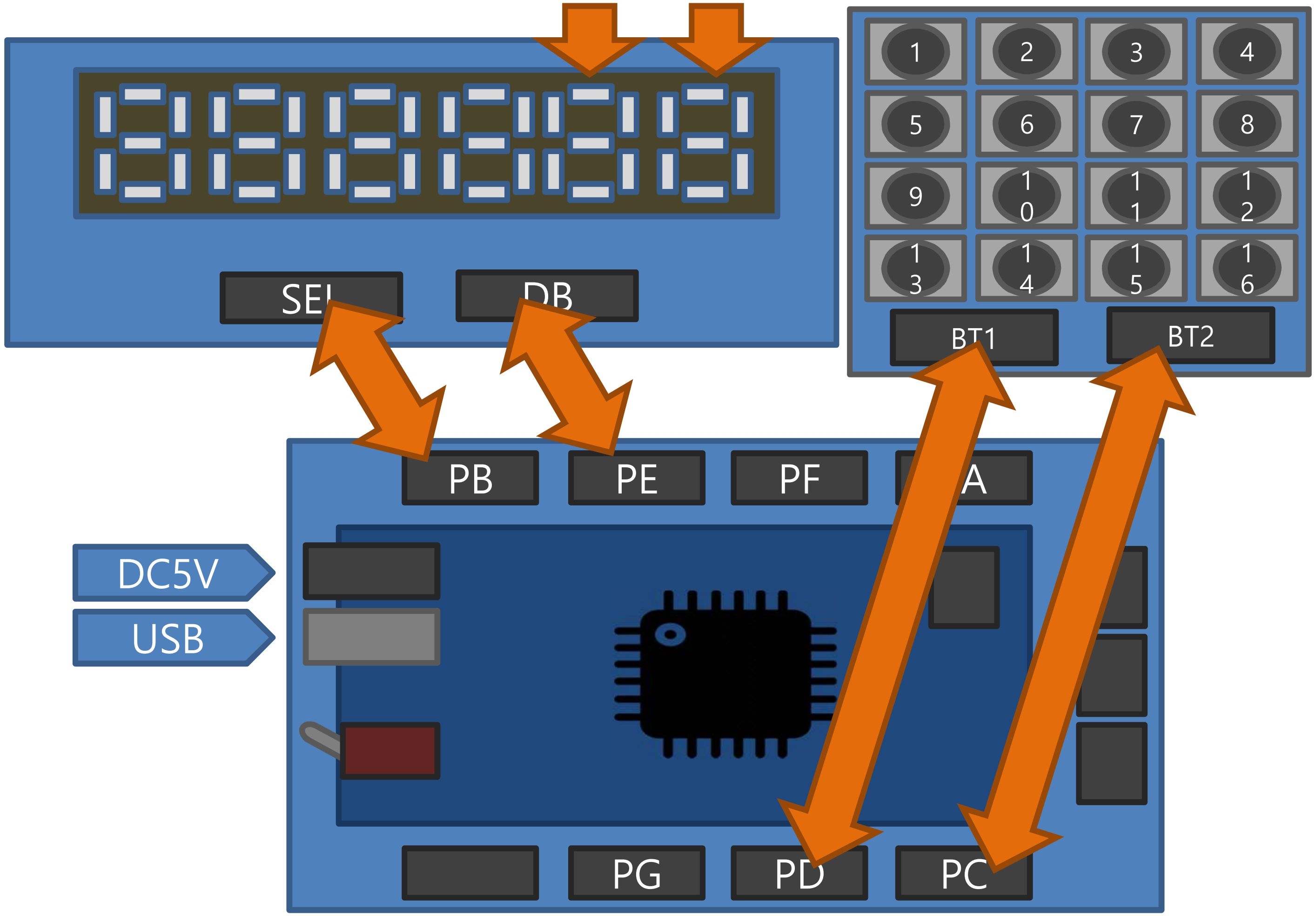
```
void Hex2Dec(unsigned char No) {
     NUM[1]=No / 10;
     NUM[0]=No % 10;
}

int main(void) {
    CPU_Setup( );

    while (1) {
        Keypad=SwitchIn( );
        if (Keypad>15){
           NUM[1]=10;     NUM[0]=10;
        }else{
           Hex2Dec(Keypad);
        }

        for (unsigned char k=0;k<2;k++){
           FND_SEL=DGT[k];
           FND_DB=FND[ NUM[k] ];
           _delay_ms(dTime);
        }
    }
}
```

Ex-8 : Hex Key Pad

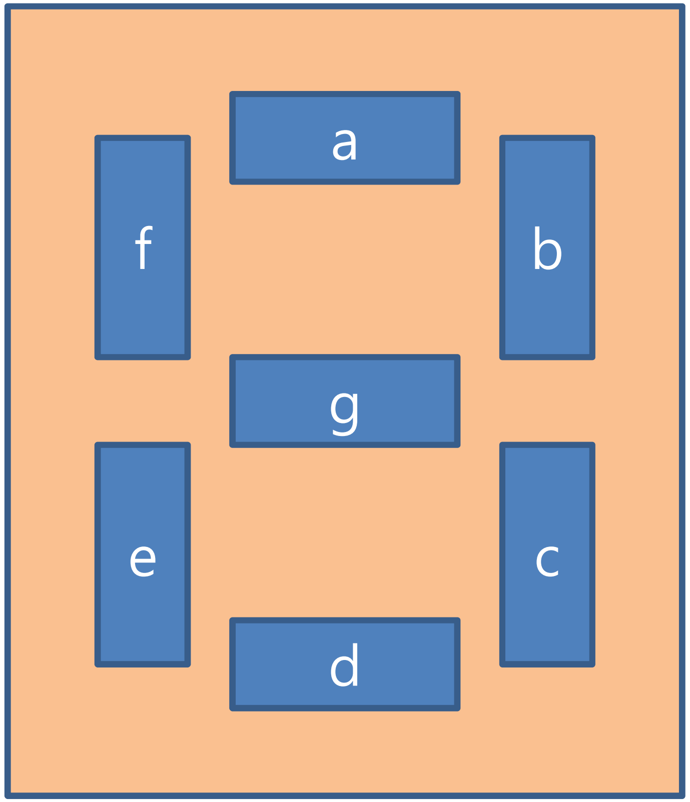
- Keypad의 배열을 다음과 같이 바꾸어 보자



Ex-8 : wiring


SEL
DB
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
BT1
BT2
PB
PE
PF
A
DC5V
USB
PG
PD
PC

FND Lookup Table



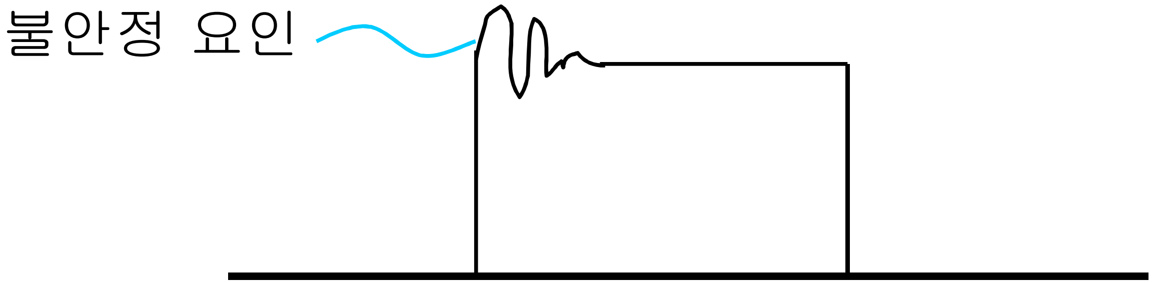
No	x	g	f	e	d	c	b	a	Hex
0	0	0	1	1	1	1	1	1	3f
1	0	0	0	0	0	1	1	0	06
2	0	1	0	1	1	0	1	1	5b
3	0	1	0	0	1	1	1	1	4f
4	0	1	1	0	0	1	1	0	66
5	0	1	1	0	1	1	0	1	6d
6	0	1	1	1	1	1	0	1	7d
7	0	0	1	0	0	1	1	1	27
8	0	1	1	1	1	1	1	1	7f
9	0	1	1	0	1	1	1	1	6f
A	0	1	1	1	0	1	1	1	77
B	0	1	1	1	1	1	0	0	7c
C	0	1	0	1	1	0	0	0	58
D	0	1	0	1	1	1	1	0	5e
-	0	1	0	0	0	0	0	0	40
=	0	1	0	0	1	0	0	1	49

FND Lookup Table

```
unsigned char FND[17]={0x3f, 0x06, 0x5b, 0x4f, 0x66, 0x6d, 0x7d, 0x27, 0x7f, 0x6f,
                        0x77, 0x7c, 0x58, 0x5e, 0x40, 0x49, 0x08};

unsigned char DGT[6]={0xfe, 0xfd, 0xfb, 0xf7, 0xef, 0xdf};
unsigned char NUM[6]={0x00, 0x00, 0x00, 0x00, 0x00, 0x00};
unsigned char Keypad;
```

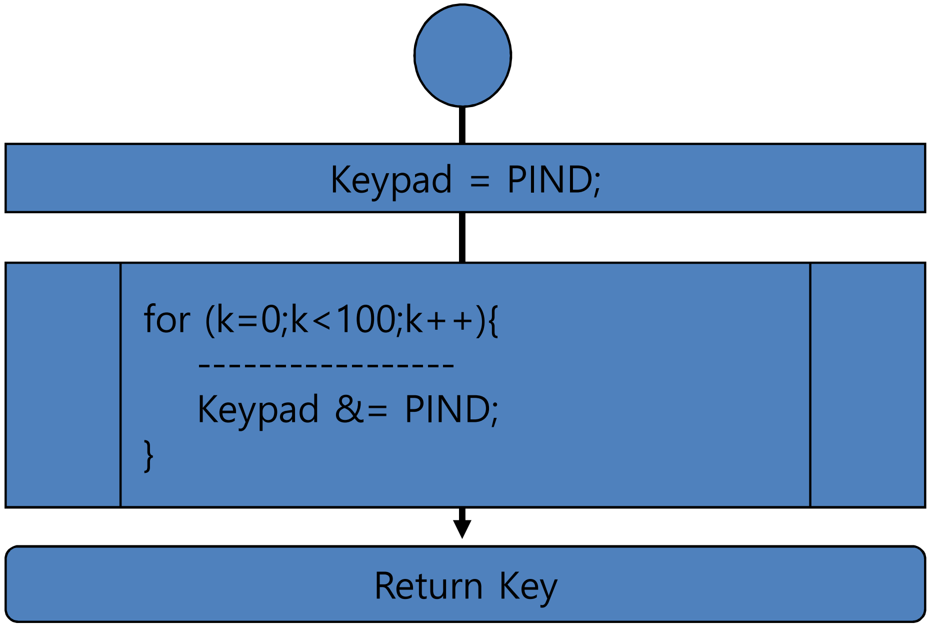
채터링[Chattering]

- 전자 회로내에 접점이 ON/OFF되는 순간 접점이 붙었다가 떨어지는 반복적인 현상
- 하드웨어적인 방지
  - 지연 회로 또는 적분 회로를 사용함
- 소프트웨어적인 방지
  - 접점의 상태를 논리적으로 반복하여 적분함



KeyIn Flow

High Active

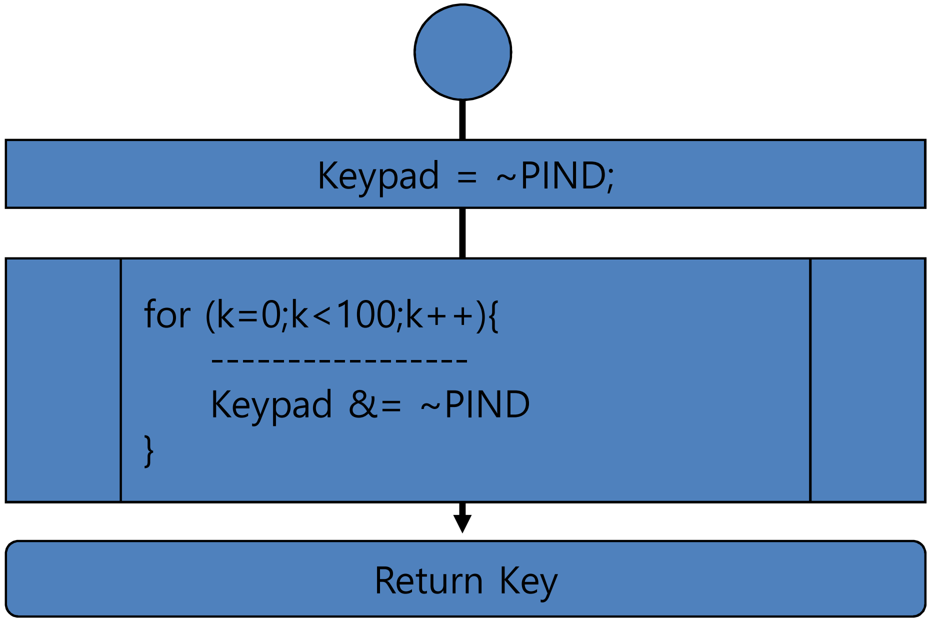


```
Keypad = PIND;
```

```
for (k=0;k<100;k++){
    ------------------
    Keypad &= PIND;
}
```

Return Key

Low Active

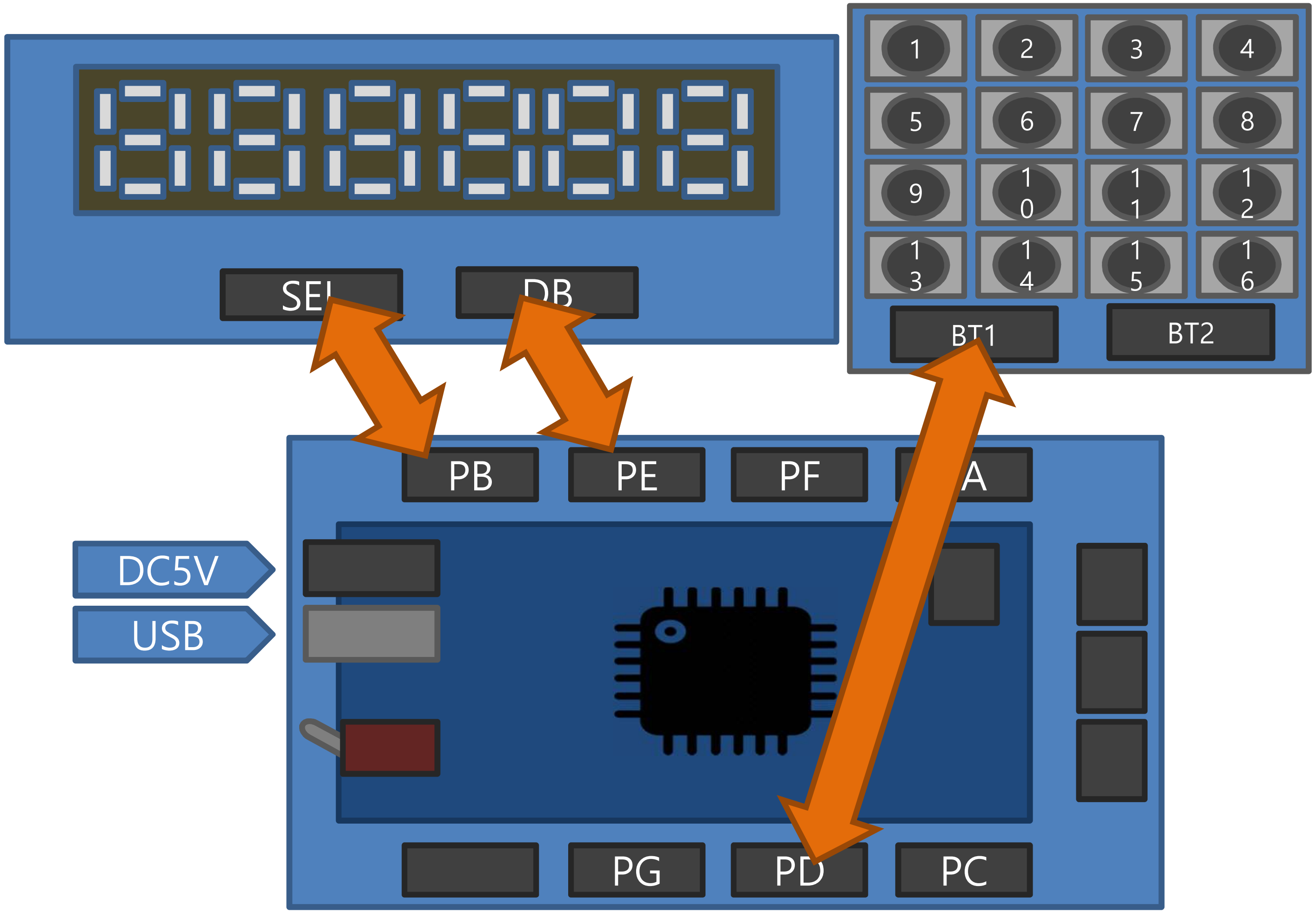


```
Keypad = ~PIND;
```

```
for (k=0;k<100;k++){
    ------------------
    Keypad &= ~PIND
}
```

Return Key

Ex-9 : 채터링 제거


SEL
DB
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
BT1
BT2
PB
PE
PF
A
DC5V
USB
PG
PD
PC

Ex-9 : Program-define

```c
#define F_CPU 14745600UL
#define FND_SEL PORTB
#define FND_DB PORTE
#define SWITCH1 PIND
#define dTime 3

#include <avr/io.h>
#include <util/delay.h>

unsigned char FND[10]={0x3f, 0x06, 0x5b, 0x4f, 0x66, 0x6d, 0x7d, 0x27, 0x7f, 0x6f};
unsigned char DGT[6]={0xfe, 0xfd, 0xfb, 0xf7, 0xef, 0xdf};
unsigned char NUM[6]={0x00, 0x00, 0x00, 0x00, 0x00, 0x00};
signed int A;
signed int B;

void CPU_Setup( ) {
    DDRB=0xff;
    DDRE=0xff;
    DDRD=0x00;
}
```

Ex-9 : KeyProc

```
unsigned char KeyIn( ) {
   unsigned char Key =~SWITCH1;
   for ( int k=0;k<100;k++){
       Key &=~SWITCH1;
   }
   return Key;
}

void KeyProc( ) {
   unsigned char tKey= KeyIn( );
   switch ( tKey ){
         case 0x01:   if (++A>999) A=999;    break;
         case 0x08:   if (++B>999) B=999;    break;
         case 0x10:   if (--A<0)   A=0;    break;
         case 0x80:   if (--B<0)   B=0;    break;
   }
}
```

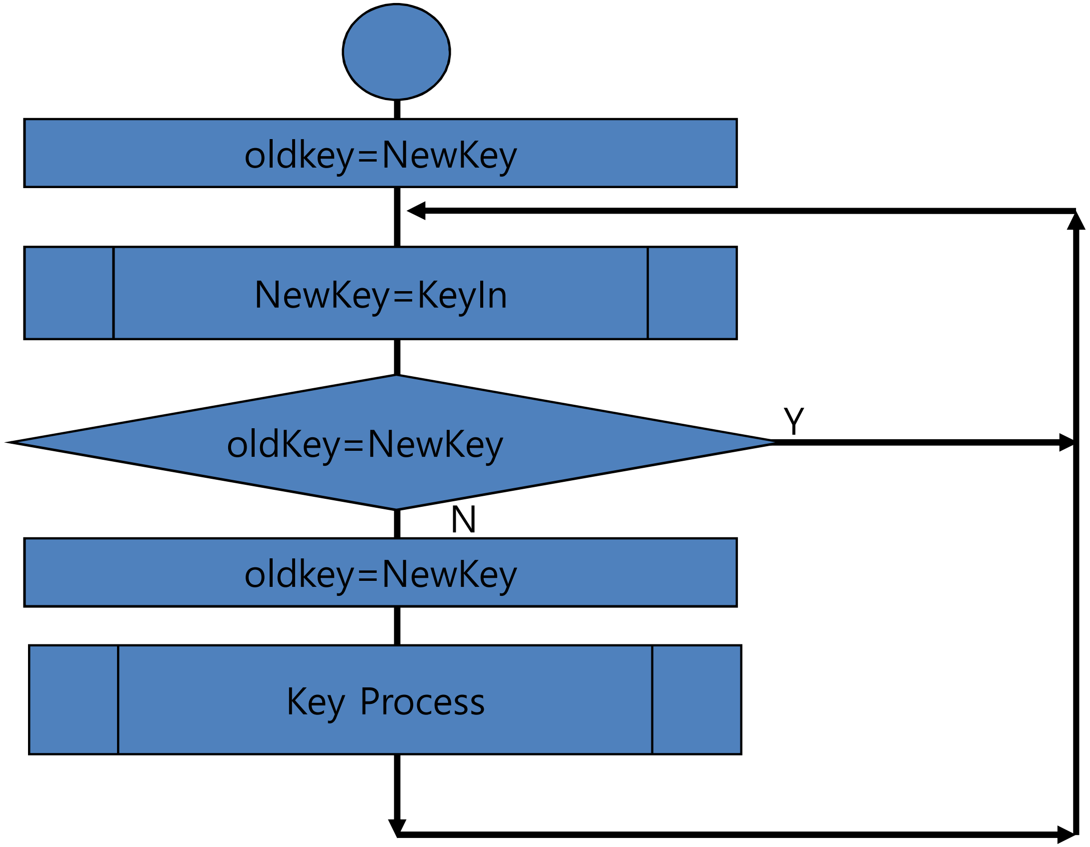
Ex-9 : sub function

```
void Hex2Dec( ){
    unsigned int tmp=A;
    NUM[5]=tmp/100;
    tmp=tmp%100;
    NUM[4]=tmp/10;
    NUM[3]=tmp%10;

    tmp=B;
    NUM[2]=tmp/100;
    tmp=tmp%100;
    NUM[1]=tmp/10;
    NUM[0]=tmp%10;
}
```

```
int main(void) {
    CPU_Setup( );

    while (1) {
       for (unsigned char k=0; k<6; k++) {
            FND_SEL=DGT[k];
            FND_DB=FND[ NUM[k] ];
            _delay_ms(dTime );
       }
       KeyProc( );
       Hex2Dec( );
    }
}
```

중복처리 방지

- Key가 계속 눌려있을때 처리하지 않음

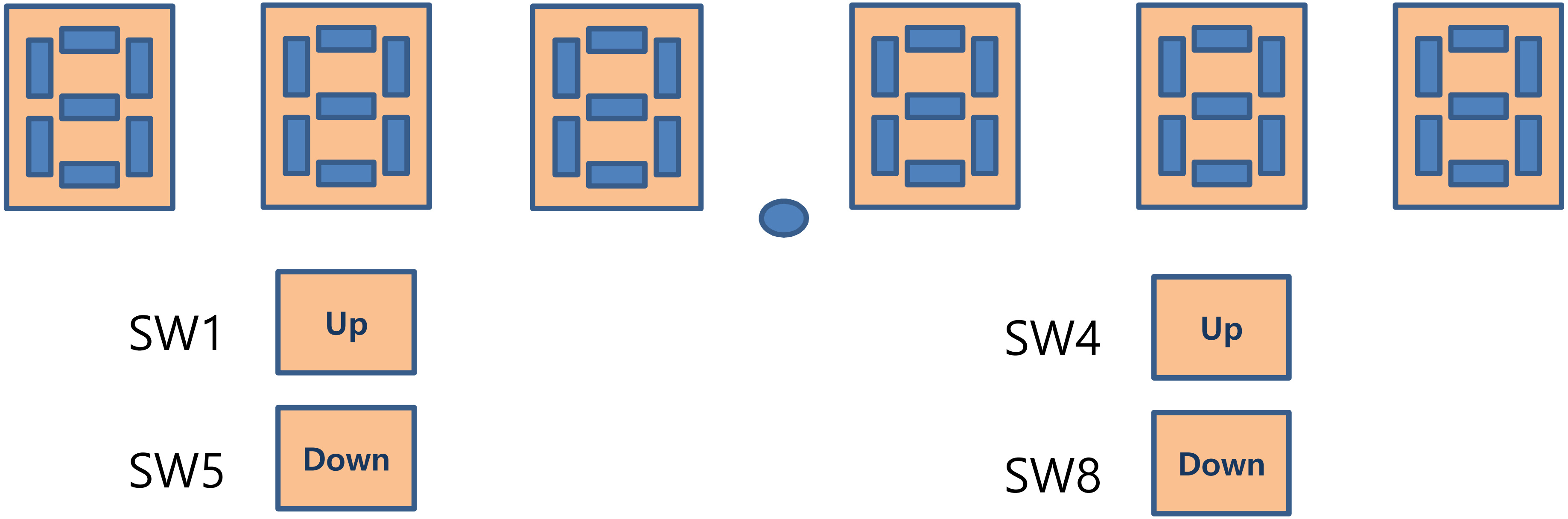


Ex-10

- SW를 누를때마다 1증가 또는 1 감소하게 하자

A: 000~999 (초기값 : 123)

B: 000~999분 (초기값 : 456)



SW1 Up

SW5 Down

SW4 Up

SW8 Down

Ex-10 : Program-define

```
#define F_CPU 14745600UL
#define FND_SEL PORTB
#define FND_DB PORTE
#define SWITCH1 PIND
#define SWITCH2 PINC
#define dTime 3

#include <avr/io.h>
#include <util/delay.h>

unsigned char FND[10]={0x3f, 0x06, 0x5b, 0x4f, 0x66, 0x6d, 0x7d, 0x27, 0x7f, 0x6f};
unsigned char DGT[6]={0xfe, 0xfd, 0xfb, 0xf7, 0xef, 0xdf};
unsigned char NUM[6]={0x00, 0x00, 0x00, 0x00, 0x00, 0x00};
unsigned char NewKey;
unsigned char oldKey;
int A;
int B;

void CPU_Setup( ) {
    DDRB=0xff;
    DDRE=0xff;
    DDRD=0x00;
}
```

Ex-10 : KeyProc

```
unsigned char KeyIn( ) {
   unsigned char Key =~SWITCH1;
   for ( int k=0;k<100;k++){
       Key &=~SWITCH1;
   }
   return Key;
}

void KeyProc( ) {
   unsigned char tKey= KeyIn( );
   NewKey=KeyIn( );
   if (NewKey != oldKey){
      switch ( tKey ){
         case 0x01:   if (++A>999) A=999;    break;
         case 0x08:   if (++B>999) B=999;    break;
         case 0x10:   if (--A<0)  A=0;    break;
         case 0x80:   if (--B<0)  B=0;    break;
      }
     oldKey=NewKey;
   }
}
```

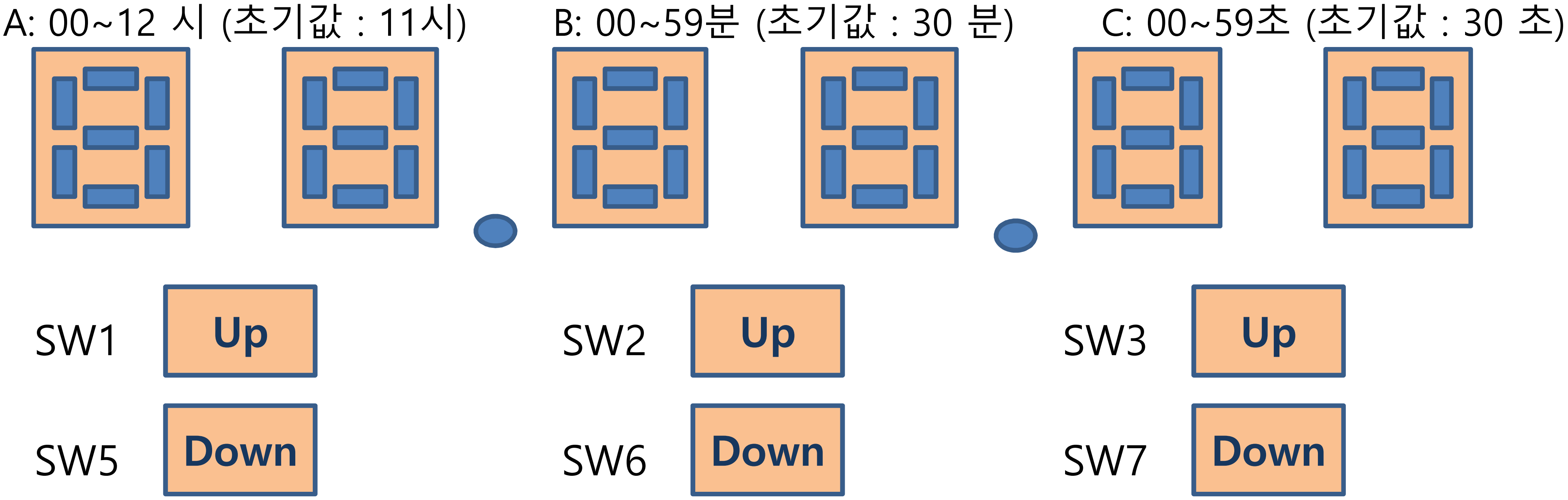
Ex-10 : sub function

```
void Hex2Dec( ){
    unsigned int tmp=A;
    NUM[5]=tmp/100;
    tmp=tmp%100;
    NUM[4]=tmp/10;
    NUM[3]=tmp%10;

    tmp=B;
    NUM[2]=tmp/100;
    tmp=tmp%100;
    NUM[1]=tmp/10;
    NUM[0]=tmp%10;
}
```

```
int main(void) {
    CPU_Setup( );

    while (1) {
       for (unsigned char k=0; k<6; k++) {
            FND_SEL=DGT[k];
            FND_DB=FND[ NUM[k] ];
            _delay_ms(dTime );
       }
       KeyProc( );
       Hex2Dec( );
    }
}
```

Ex-11 : 시간 설정

- SW를 누를때마다 1증가 또는 1 감소하게 하자
- 단, 분값이 59보다 크면 시 증가하게 하여야 함



A: 00~12 시 (초기값 : 11시)

B: 00~59분 (초기값 : 30 분)

C: 00~59초 (초기값 : 30 초)

SW1 Up

SW5 Down

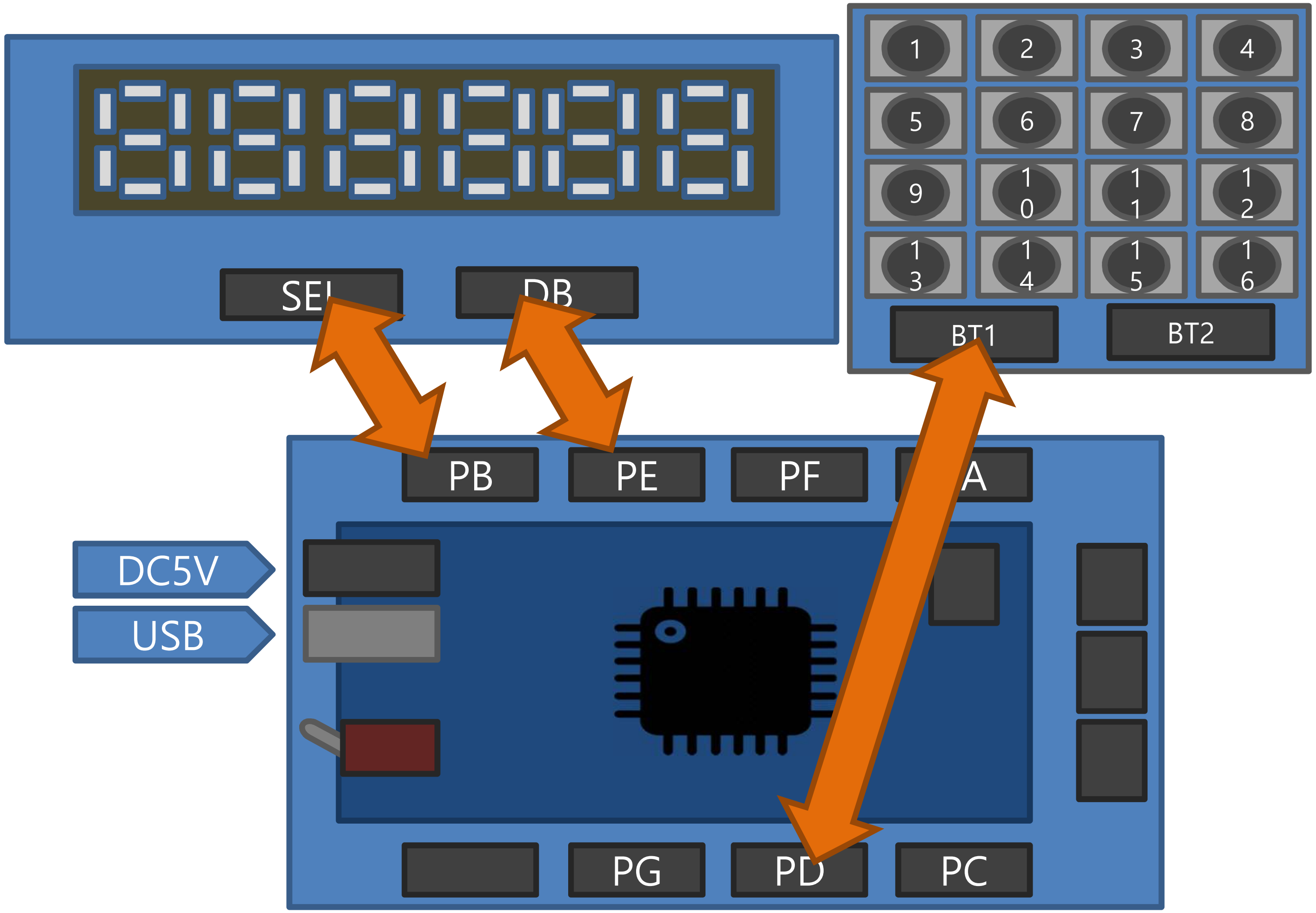
SW2 Up

SW6 Down

SW3 Up

SW7 Down

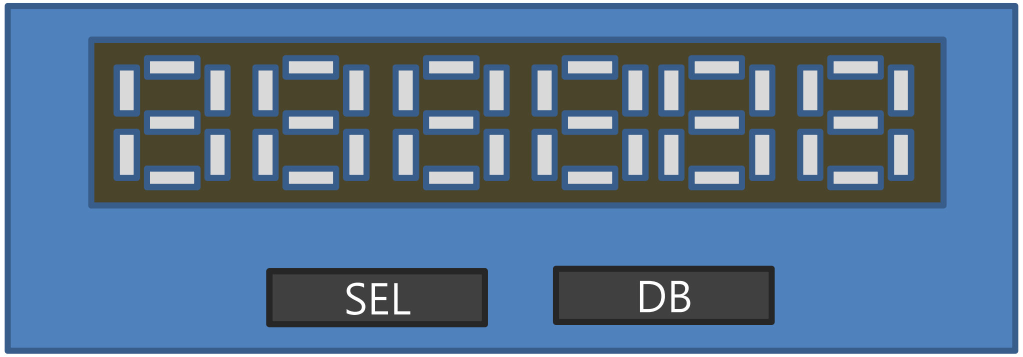
Ex-11 : wiring

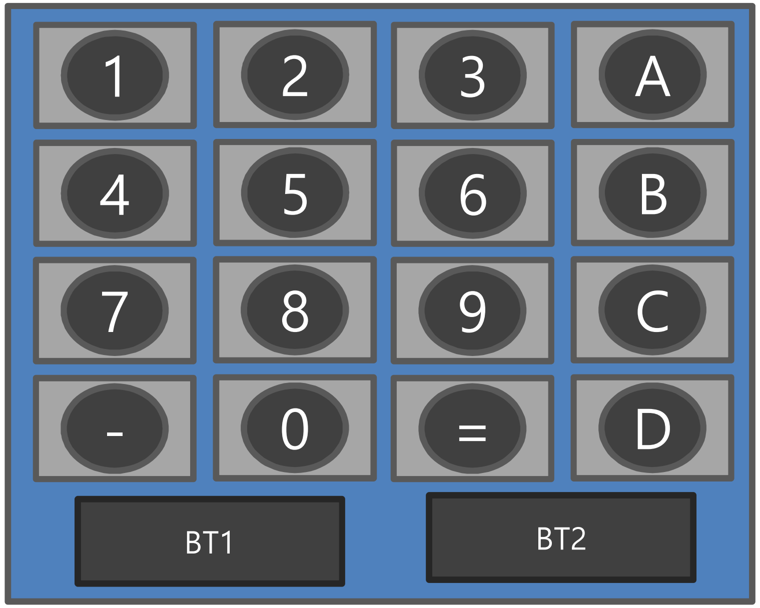

SEL
DB
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
BT1
BT2
PB
PE
PF
A
DC5V
USB
PG
PD
PC

Ex-11 : Program

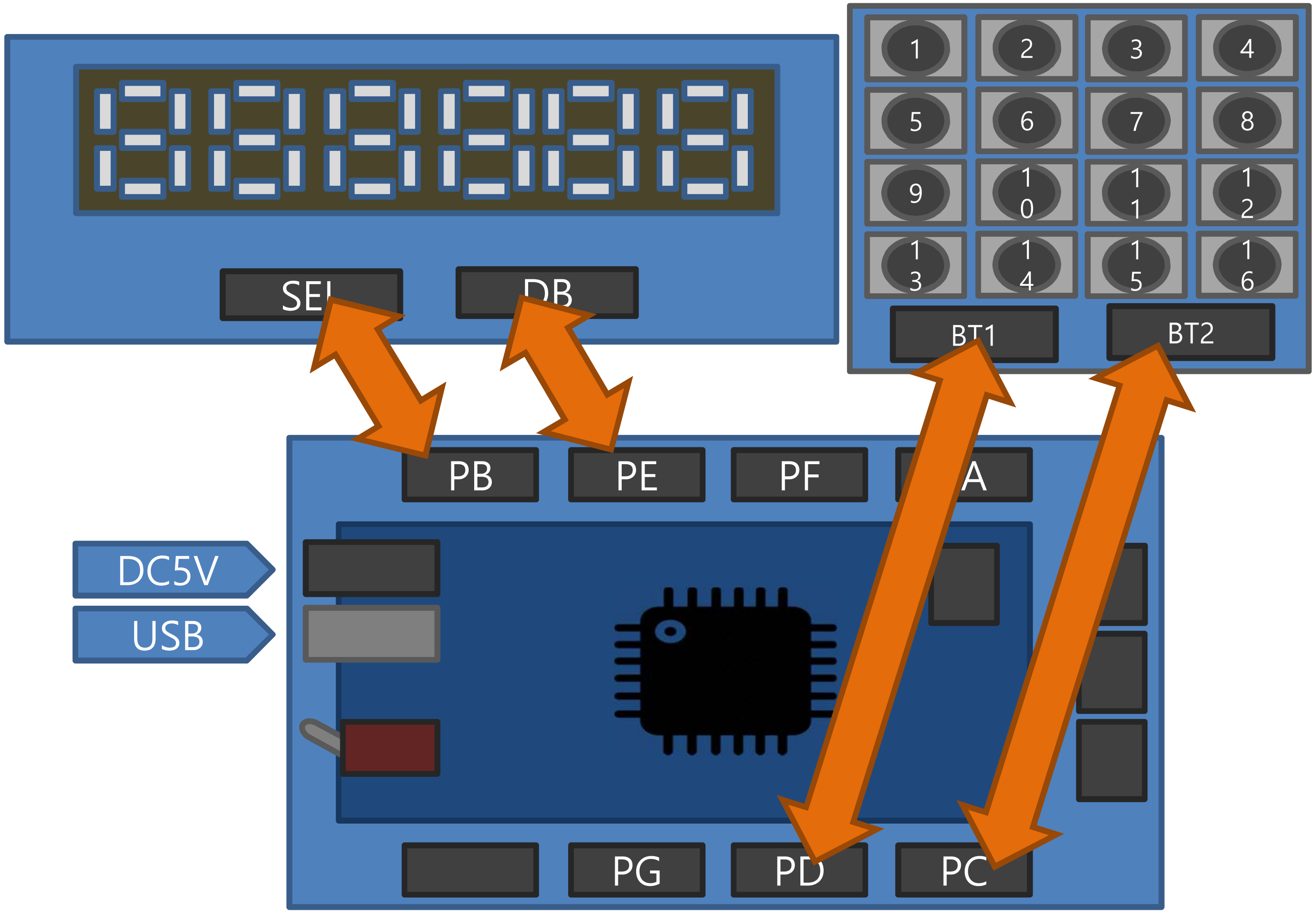
Ex-12 : 10Key Input

- Key button을 눌러 10진수 숫자를 입력하자
- 숫자는 우측에서 좌측으로 쉬프트





Ex-12 : wiring


1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
SEL
DB
BT1
BT2
PB
PE
PF
A
DC5V
USB
PG
PD
PC

Ex-12 : Program-define

```c
#define F_CPU 14745600UL
#define FND_SEL PORTB
#define FND_DB PORTE
#define SWITCH1 PIND
#define SWITCH2 PINC
#define dTime 3

#include <avr/io.h>
#include <util/delay.h>

unsigned char FND[17]={0x3f, 0x06, 0x5b, 0x4f, 0x66, 0x6d, 0x7d, 0x27, 0x7f, 0x6f, 0x77,
                        0x7c, 0x58, 0x5e, 0x40, 0x49, 0x08};
unsigned char DGT[6]={0xfe, 0xfd, 0xfb, 0xf7, 0xef, 0xdf};
unsigned char NUM[6]={0x00, 0x00, 0x00, 0x00, 0x00, 0x00};
unsigned char NewKey=16;
unsigned char oldKey=16;

void CPU_Setup( ) {
    DDRB=0xff;
    DDRE=0xff;
    DDRC=0x00;
    DDRD=0x00;
}
```

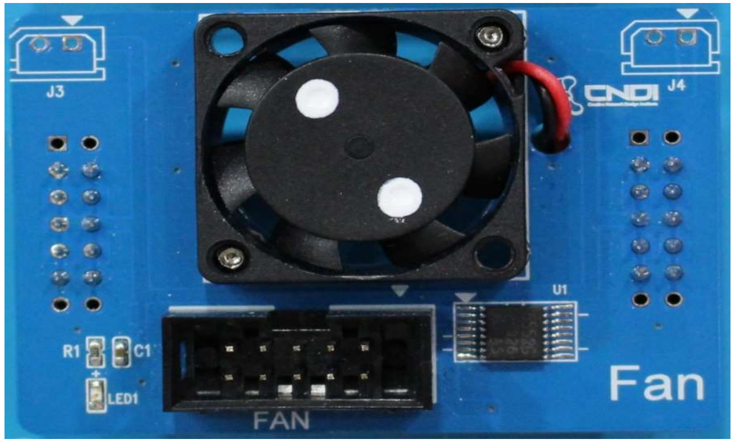
Ex-12 : sub function

```
char SwitchIn(void) {
        char KeyNo;
        unsigned int KeyIn =(~SWITCH2<<8) | (~SWITCH1 & 0x00ff);

        switch ( KeyIn ) {
               case 0x0001: KeyNo =1;   break;
               case 0x0002: KeyNo =2;   break;
               case 0x0004: KeyNo =3;   break;
               case 0x0008: KeyNo =10;  break;
               case 0x0010: KeyNo =4;   break;
               case 0x0020: KeyNo =5;   break;
               case 0x0040: KeyNo =6;   break;
               case 0x0080: KeyNo =11;  break;
               case 0x0100: KeyNo =7;   break;
               case 0x0200: KeyNo =8;   break;
               case 0x0400: KeyNo =9;   break;
               case 0x0800: KeyNo =12;  break;
               case 0x1000: KeyNo =14;  break;
               case 0x2000: KeyNo =0;   break;
               case 0x4000: KeyNo =15;  break;
               case 0x8000: KeyNo =13;  break;
               default: KeyNo =16;  break;
        }
       return KeyNo;
}
```

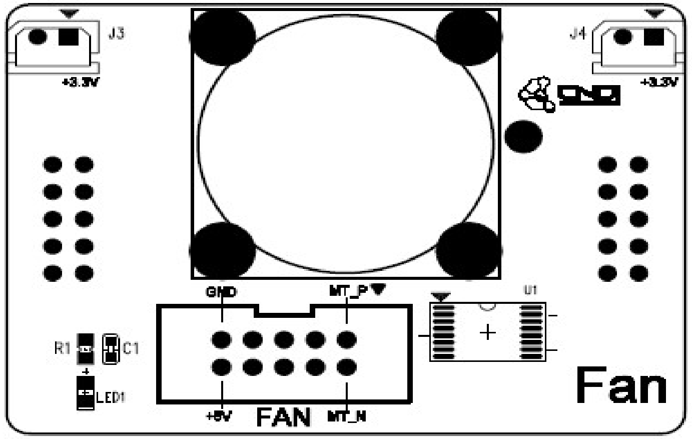
Ex-12 : main function

```
int main(void) {
    CPU_Setup( );

    while (1) {
        NewKey=SwitchIn( );
        if (NewKey != oldKey){
            oldKey=NewKey;
            if (NewKey != 16){
                for (unsigned char k=5;k>0;k--){
                     NUM[k]=NUM[k-1];
                }
                NUM[0]=NewKey;
            }
        }

       for (unsigned char k=0; k<6; k++) {
            FND_SEL=DGT[k];
            FND_DB=FND[ NUM[k] ];
            _delay_ms(dTime );
       }
    }
}
```

FAN

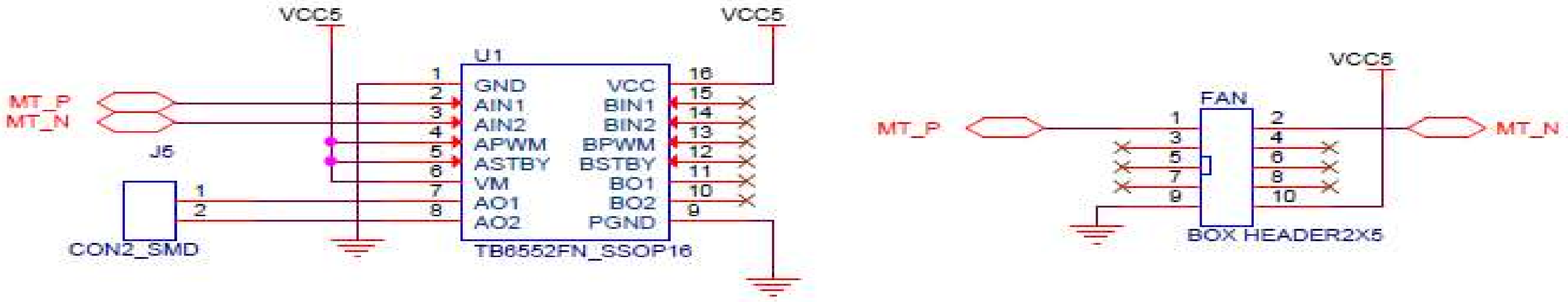


J3
J4
CNDI
U1
R1
C1
LED1
FAN
Fan



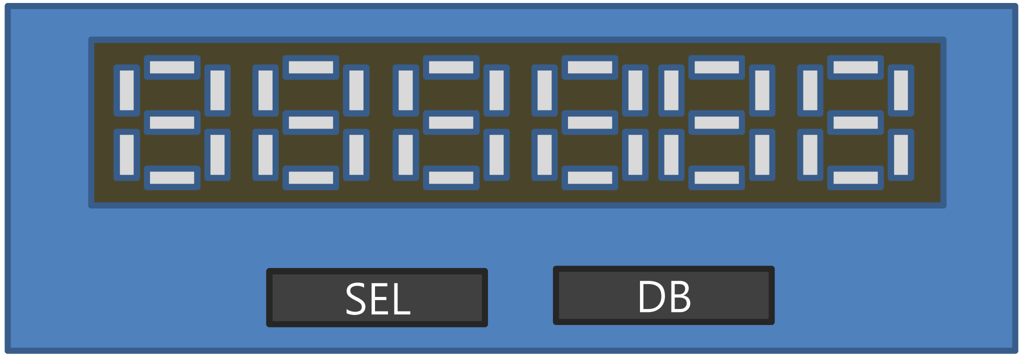
J3
+3.3V
J4
+3.3V
CNDI
GND
MT_P
U1
R1
C1
LED1
+5V
FAN
MT_N
Fan

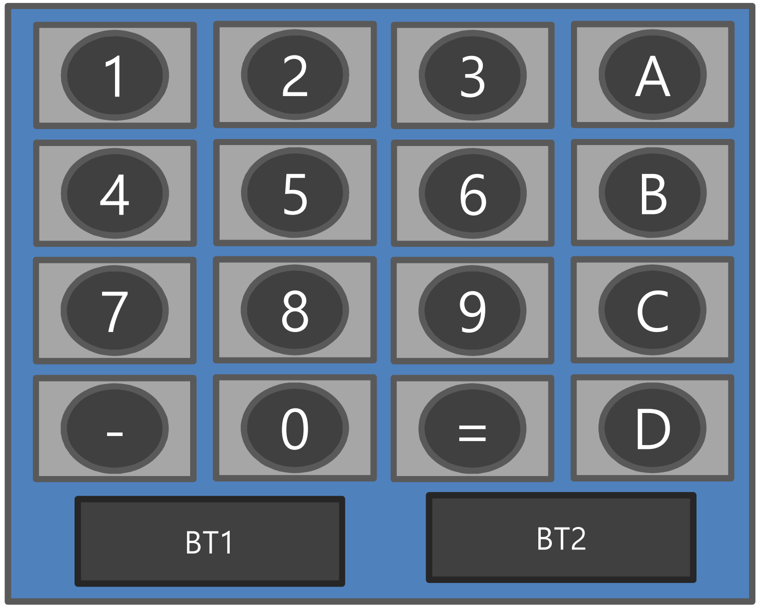
Fan Schematic



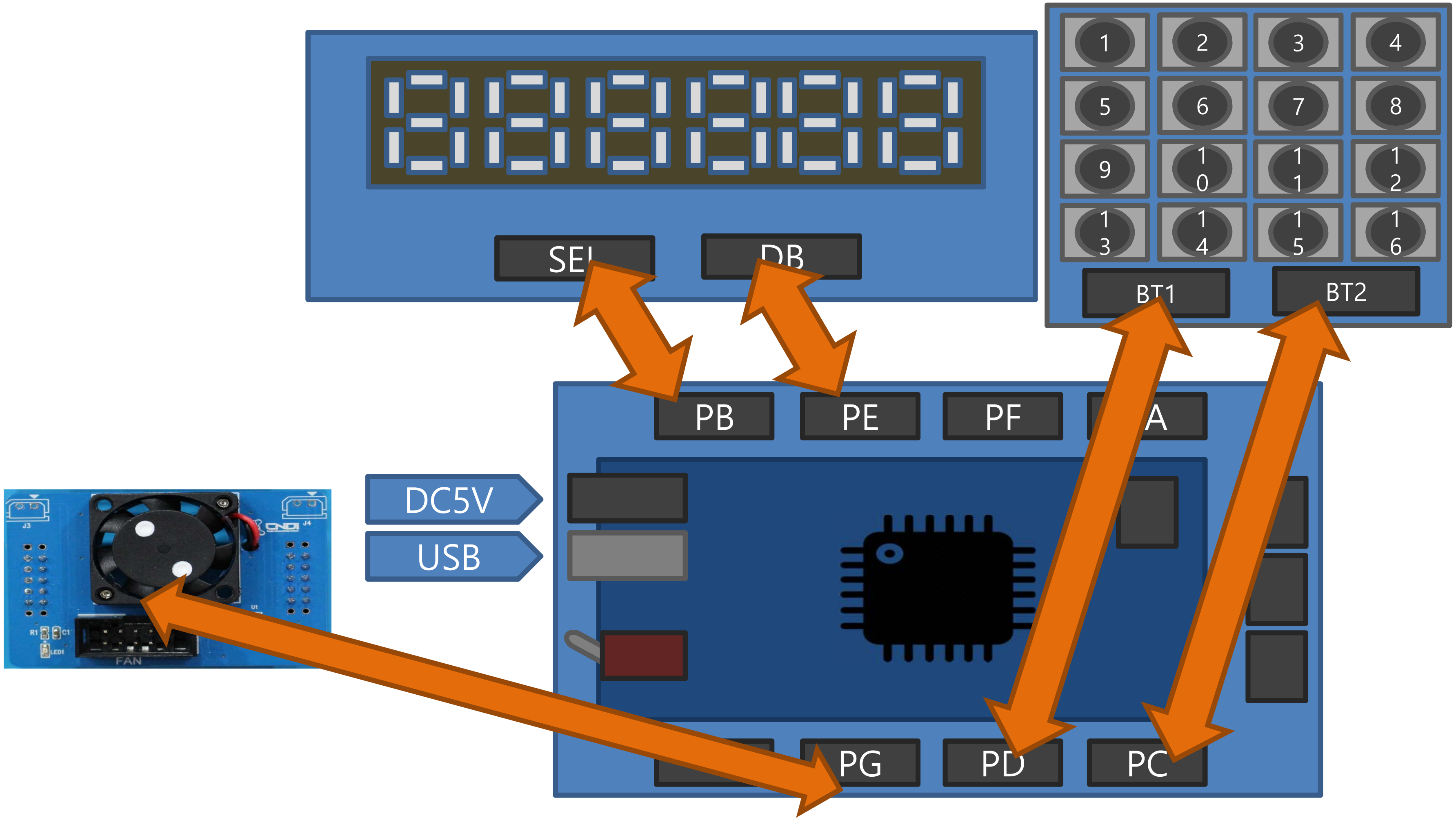
Ex-13 : Pass Word

- Key button을 눌러 10진수 숫자를 입력하자
- 단, 0~9이외의 키는 입력 받지 않는다
- 숫자 6개를 입력하고 A키를 눌러 PW가 맞으면
- FAN을 구동, FND는 '------'표시
- D키가 입력되면 FAN을 정지하자





Ex-13 : wiring


SEL
DB
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
BT1
BT2
PB
PE
PF
A
DC5V
USB
FAN
PG
PD
PC

Ex-13 : Program-define

```c
#define F_CPU 14745600UL
#define FND_SEL PORTB
#define FND_DB PORTE
#define SWITCH1 PIND
#define SWITCH2 PINC
#define FAN PORTG
#define dTime 3

#include <avr/io.h>
#include <util/delay.h>

unsigned char FND[17]={0x3f, 0x06, 0x5b, 0x4f, 0x66, 0x6d, 0x7d, 0x27, 0x7f, 0x6f, 0x77,
                        0x7c, 0x58, 0x5e, 0x40, 0x49, 0x08};
unsigned char DGT[6]={0xfe, 0xfd, 0xfb, 0xf7, 0xef, 0xdf};
unsigned char NUM[6]={16, 16, 16, 16, 16, 16};
unsigned char PW[6]={0x01, 0x02, 0x03, 0x04, 0x05, 0x06};
unsigned char NewKey=16;
unsigned char oldKey=16;

void CPU_Setup( ) {
    DDRB=0xff;
    DDRE=0xff;
    DDRC=0x00;
    DDRD=0x00;
    DDRG=0x03;
}
```

Ex-13 : Program

```
FAN=0x01;  //Fan On
FAN=0x00;  //Fan Off
```