

# Exercise #04 REPORT

Name: 송예린

Student ID: 2171023

### 1. Exercise Objective (10%)

LED 를 작동시킬 때 포인터로 직접 address 에 접근해서 값을 변경할 수도 있지만 미리 특정 값을 define 한 Predefine 된 값을 사용해서 작동시키는 방법도 있다. msp432p4111.h 라는 header file 에 Predefined Macro 와 Data type 이 구조체로 정의되어 있어서 주소,타입이 키워드로 치환되고 키워드를 이용해서 address 에 접근하거나 특정 값을 변경할 수 있다. 포인터가 아닌 Predefined Macro 를 활용해서 LED 의 색상이 바뀌는 실습을 할 예정이다.

### 2. Implementation (60%)

1) LED 의 색상이 연두색->하늘색->파란색->분홍색->빨간색->노란색->흰색으로 바뀌는 실습

RGB 3 가지 색상을 이용하면 여러 색상을 표현할 수 있다. R 은 0 번째 핀, G 는 1 번째 핀 B 는 2 번째 핀을 1 로 바꾸면 표현이 가능하고 2~3 가지 색상을 섞는 것처럼 2~3 개의 핀을 동시에 키면 다른 색상들도 표현할 수 있다. 각 색상을 Hexadecimal 로 바꿔주면 연두색: 0x02 하늘색:0x06 파란색: 0x04 분홍색: 0x05 빨간색: 0x01 노란색: 0x03 흰색: 0x07 이다. P2 는 port 의 Base address 값이 predefine 되어있고 P2 의 구조체로 byte 만큼 떨어진 값들이 predefine 되어있다 P2->DIR 을 하면 Base address 에서 5byte 만큼 떨어진 곳에 값을 변경한다는 의미이다. 해당 핀을 키고 싶다면 DIR 에서 해당 핀의 위치를 1 로 설정해줘야 한다.  $P2->DIR |= 0x02$  ( $P2->DIR = P2->DIR | 00000010$ )를 하면 Base address 에서 5byte 만큼 떨어진 곳에서 OR 연산자로 해당 핀을 1 로 만들겠다는 의미를 가진다. P2->OUT 은 하면 Base address 에서 3byte 만큼 떨어진 주소를 의미한다. P2->out=0x02 로 값을 00000010 로 바꾸면서 불을 켜주고 for 문으로 딜레이를 줬다가 p2->out=0 으로 모두 0 으로 만들어주면서 불을 꺼준다. 다른 색상들도 마찬가지로 P2->DIR 을 OR 연산자를 이용해서 설정해주고 P2->OUT 으로 값을 바꾸고 딜레이를 주고 0 으로 만들고 딜레이를 주는 식으로 작성하였다. 색상변화가 무한대로 반복되므로 모든 코드를 while 문 안에 적었다.

### 3. Discussion & Conclusion (10%)

3 가지 색상 RGB 을 혼합하여 여러가지 색상의 LED 를 만드는 실습을 진행하였는데 주소를 직접 적는 것이 아닌 미리 정의된 값을 키워드만 적어서 코드를 작성하였다.. 코드를 작성할 때마다 주소를 찾는 것이 아닌 미리 정의된 값을 사용했기 때문에 지난번 실습보다 간편하고 알아보기도 수월하다고 생각된다. 이 프로젝트를 통해서 Predefined Macro 을 이용해 주소에 접근하여 값을 변경하는 방법을 배울 수 있었다.

#### 4. Reference(s)

#### 5. Code (20%)

FILE NAME (ex: main.c)

```
#include "msp.h"

/**
 * main.c
 */
void main(void)
{
    int i;
    WDT_A->CTL = WDT_A_CTL_PW | WDT_A_CTL_HOLD; // stop watchdog timer
    while(1){
        P2->DIR|=0x02;
        P2->OUT=0x02;
        for(i=0; i<100000; i++){
            ;
        } //1
        P2->OUT=0;
        for(i=0; i<100000; i++){
            ;
        }
        P2->DIR|=0x06;
        P2->OUT=0x06;
        for(i=0; i<100000; i++){
            ;
        }
        P2->OUT=0;
        for(i=0; i<100000; i++){
            ;
        }
        P2->DIR|=0x05;
        P2->OUT=0x05;
        for(i=0; i<100000; i++){
            ;
        }
        P2->OUT=0;
        for(i=0; i<100000; i++){
            ;
        }
        P2->DIR|=0x01;
        P2->OUT=0x01;
        for(i=0; i<100000; i++){
            ;
        }
        P2->OUT=0;
        for(i=0; i<100000; i++){
            ;
        }
    }
}
```

```
P2->DIR|=0x03;
P2->OUT=0x03;
for(i=0; i<100000; i++){
    ;
}
P2->OUT=0;
for(i=0; i<100000; i++){
    ;
}
P2->DIR|=0x07;
P2->OUT=0x07;
for(i=0; i<100000; i++){
    ;
}
P2->OUT=0;
for(i=0; i<100000; i++){
    ;
}

}
}
```

**FILE NAME (ex: main2.c)**