

# 시스템 프로그래밍을 위한 C언어

## union을 이용하여 메모리에 할당된 multiple bytes 데이터 패턴을 다양한 타입으로 바라보기

현대자동차 입문교육  
박대진 교수

# Union을 이용하여 특정 메모리에 할당된 데이터 패턴을 다양한 타입으로 표현하기(해석)

```
struct bits_8 {  
    unsigned char b0 : 1;  
    unsigned char b1 : 1;  
    unsigned char b2 : 1;  
    unsigned char b3 : 1;  
    unsigned char b4 : 1;  
    unsigned char b5 : 1;  
    unsigned char b6 : 1;  
    unsigned char b7 : 1;  
};
```

```
union flag_16bits {  
    unsigned char bytes[2];  
    struct bits_8 b[2];  
    unsigned short bits16;  
};
```

```
union flag_16bits flag16;  
printf("flag16 is with %d bytes\n", sizeof(flag16));
```

flag16 is with 2 bytes  
flag16 is 0xF005  
flag16 is 0x7005  
flag16 is 0x700D

```
flag16.bits16 = 0xF005;  
printf("flag16 is 0x%02X%02X\n", flag16.bytes[1], flag16.bytes[0]);
```

```
flag16.b[1].b7 = 0;  
printf("flag16 is 0x%02X%02X\n", flag16.bytes[1], flag16.bytes[0]);  
flag16.b[0].b3 = 1;  
printf("flag16 is 0x%02X%02X\n", flag16.bytes[1], flag16.bytes[0]);
```

# Memory Layout

