

**마이크로프로세서응용설계, 2019년 1학기**

**실습 4**  
**Dot Matrix 디바이스 드라이버 및**  
**응용 프로그램 작성**

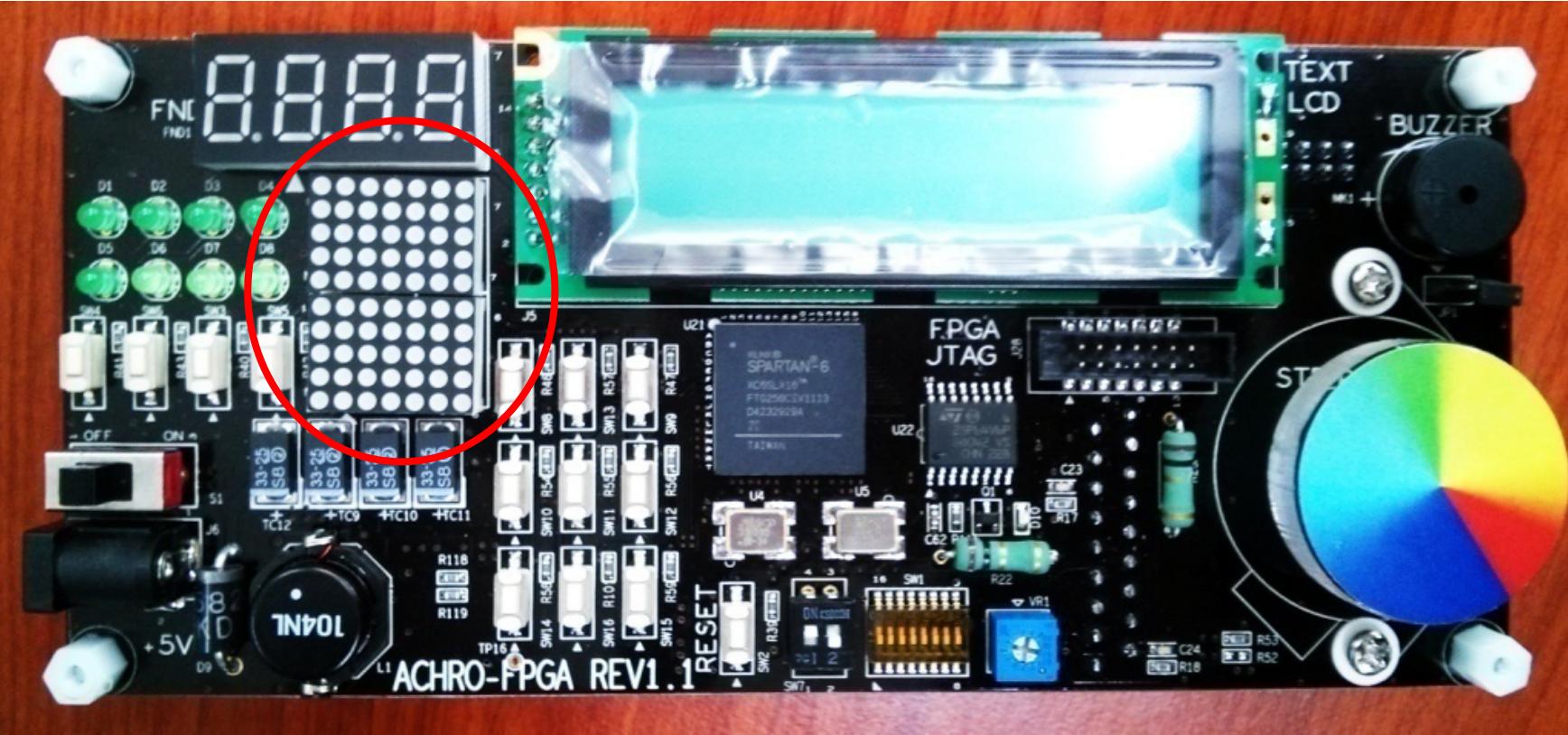
**컴퓨터공학부**

# I/O 장치들

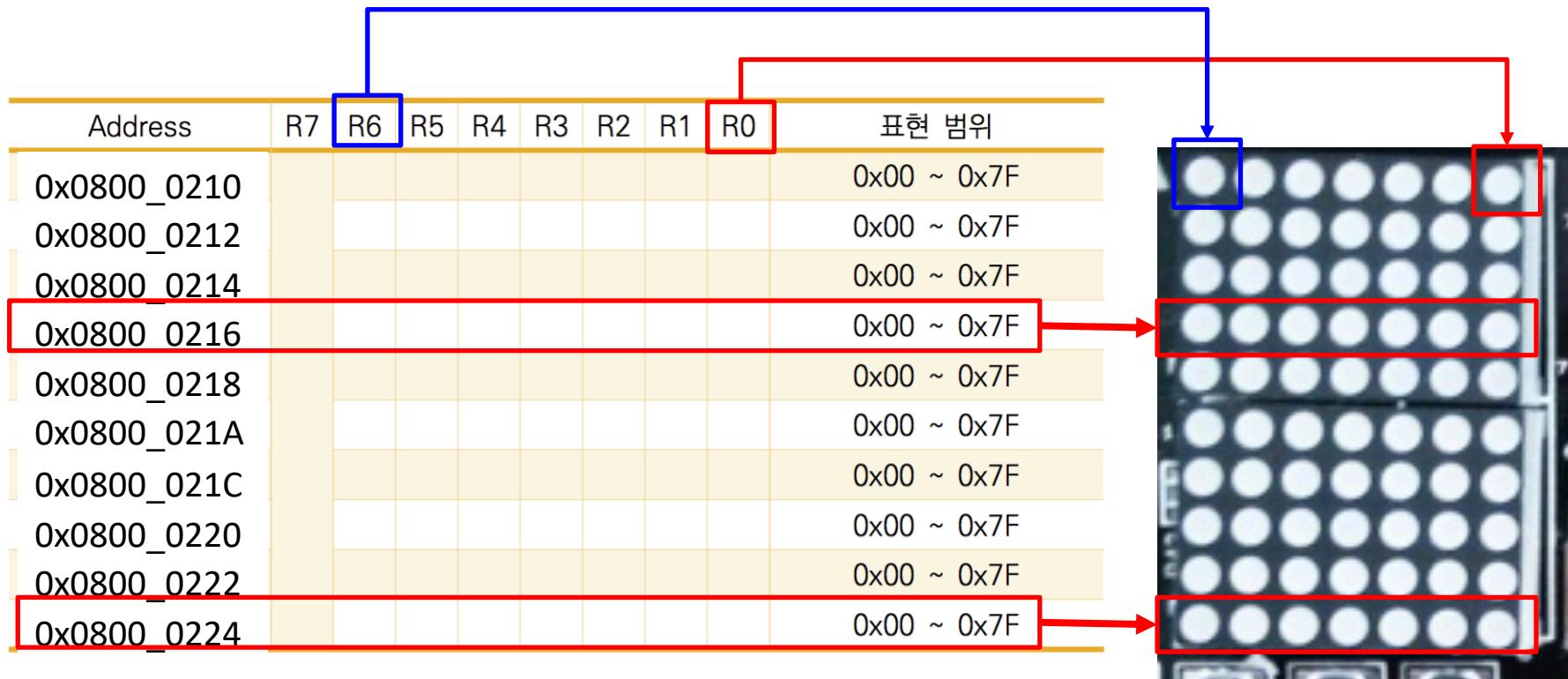
## ■ I/O 장치들의 물리 주소, 디바이스 노드, 주번호

Num	Device	Address	Major
1	LED	0x0800_0016	260
2	Seven Segment (FND)	0x0800_0004	261
<b>3</b>	<b>Dot Matrix</b>	<b>0x0800_0210</b>	<b>262</b>
4	Text LCD	0x0800_0100	263
5	Buzzer	0x0800_0020	264
6	Push Switch	0x0800_0050	265
7	3Dip Switch	0x0800_0000	266
8	Step Motor	0x0800_000C	267
EN	Demo Register	0x0800_0300	N/A

# Dot Matrix



# Address Map



# Dot Matrix에 출력하는 방법

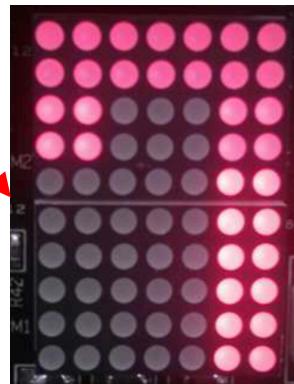
# 출력의 예

x	0	0	0	1	1	0	0	0c
x	0	0	1	1	1	0	0	1c
x	0	0	1	1	1	0	0	1c
x	0	0	0	1	1	0	0	0c
x	0	0	0	1	1	0	0	0c
x	0	0	0	1	1	0	0	0c
x	0	0	0	1	1	0	0	0c
x	0	0	0	1	1	0	0	0c
x	0	0	0	1	1	0	0	0c
x	0	0	1	1	1	1	0	1e

x	1	1	0	0	0	0	0	66
x	1	1	0	0	1	1	0	66
x	1	1	0	0	1	1	0	66
x	1	1	0	0	1	1	0	66
x	1	1	0	0	1	1	0	66
x	1	1	0	0	1	1	0	66
x	1	1	0	0	1	1	0	66
x	1	1	1	1	1	1	1	71
x	1	1	1	1	1	1	1	71
x	0	0	0	0	1	1	0	06
x	0	0	0	0	1	1	0	06

x	1	1	0	0	0	0	0	60
x	1	1	0	0	0	0	0	60
x	1	1	0	0	0	0	0	60
x	1	1	0	0	0	0	0	60
x	1	1	1	1	1	1	0	7e
x	1	1	1	1	1	1	1	7f
x	1	1	0	0	0	1	1	63
x	1	1	0	0	0	1	1	63
x	1	1	1	1	1	1	1	7f
x	0	1	1	1	1	1	0	3e

x	0	1	1	1	1	1	0	3
x	1	1	1	1	1	1	1	7
x	1	1	0	0	0	1	1	6
x	1	1	0	0	0	1	1	6
x	1	1	1	1	1	1	1	7
x	1	1	1	1	1	1	1	7
x	1	1	0	0	0	1	1	6
x	1	1	0	0	0	1	1	6
x	1	1	1	1	1	1	1	7
x	0	1	1	1	1	1	0	3



# Dot Matrix 디바이스 드라이버 구조

## dot\_matrix\_driver.c

```
// header files and device definitions
#include ...
#define ...

// global variable definitions
...

// functions declarations
...

// device file operation
struct file_operations csemad_dot_matrix_fops = { ... };

// initialize push dot matrix
int __init csemad_dot_matrix_init(void) { ... }

// release and exit dot matrix port
void __exit csemad_dot_matrix_exit(void) { ... }

// write to dot matrix port
ssize_t csemad_dot_matrix_write(struct file *inode, char *gdata, size_t length, loff_t *off_what) { ... }

module_init(csemad_dot_matrix_init);
module_exit(csemad_dot_matrix_exit);

MODULE_LICENSE("GPL");
MODULE_AUTHOR("CSEMAD");
```

- 각 함수들의 구현은 Lab2의 LED driver와 동일함

# 실습 3

## ■ 응용 프로그램

- 파일명: test\_dot\_matrix.c
- 커맨드라인에서 첫번째 인자로 0-9 사이의 정수를 인자로 받아 dot matrix에 출력함
- 0-9 사이의 정수가 아닌 값을 인자로 받았을때는 에러 메시지를 출력하고 프로그램을 종료함
- 다음 페이지의 dot\_matrix\_font.h 를 헤더파일로 포함하여 폰트로 사용할 것

```
write(dev, dot_matrix_font[number], sizeof(dot_matrix_font[number]));
```

## ■ 디바이스 드라이버 구조 및 Makefile은 LED 실습과 매우 유사하므로 이전 실습의 내용을 참고하여 만들것

# dot\_matrix\_font.h

```
#ifndef __CSEMAP__
#define __CSEMAP__


unsigned char dot_matrix_font[10][10] = {
    {0x3e,0x7f,0x63,0x73,0x73,0x6f,0x67,0x63,0x7f,0x3e}, // 0
    {0x0c,0x1c,0x1c,0x0c,0x0c,0x0c,0x0c,0x0c,0x0c,0x1e}, // 1
    {0x7e,0x7f,0x03,0x03,0x3f,0x7e,0x60,0x60,0x7f,0x7f}, // 2
    {0xfe,0x7f,0x03,0x03,0x7f,0x7f,0x03,0x03,0x7f,0x7e}, // 3
    {0x66,0x66,0x66,0x66,0x66,0x66,0x7f,0x7f,0x06,0x06}, // 4
    {0x7f,0x7f,0x60,0x60,0x7e,0x7f,0x03,0x03,0x7f,0x7e}, // 5
    {0x60,0x60,0x60,0x60,0x7e,0x7f,0x63,0x63,0x7f,0x3e}, // 6
    {0x7f,0x7f,0x63,0x63,0x03,0x03,0x03,0x03,0x03,0x03}, // 7
    {0x3e,0x7f,0x63,0x63,0x7f,0x7f,0x63,0x63,0x7f,0x3e}, // 8
    {0x3e,0x7f,0x63,0x63,0x7f,0x3f,0x03,0x03,0x03,0x03} // 9
};

unsigned char dot_matrix_full[10] = {
    0x7f,0x7f,0x7f,0x7f,0x7f,0x7f,0x7f,0x7f,0x7f,0x7f
};

unsigned char dot_matrix_blank[10] = {
    0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00
};

#endif
```