

Homework04: GPIO, Serial interface

Tae-Hyun Oh

Associate Professor

Dept. Electrical Engineering

POSTECH, Korea

Slides by
Youngjoo Lee

Intro to GPIO

GPIO(General-Purpose Input/Output)

- ✓ 입력이나 출력을 포함한 동작이 런타임 시에 사용자에게 의해 제어될 수 있는, 집적 회로나 전기 회로 기판의 디지털 신호 핀
- ✓ Raspberry Pi 3에는 아래 그림처럼 GPIO가 구성



Raspberry Pi 3 GPIO Header				
Pin#	NAME		NAME	Pin#
01	3.3v DC Power	Red	DC Power 5v	02
03	GPIO02 (SDA1 , I ² C)	Blue	DC Power 5v	04
05	GPIO03 (SCL1 , I ² C)	Black	Ground	06
07	GPIO04 (GPIO_GCLK)	Green	(TXD0) GPIO14	08
09	Ground	Orange	(RXD0) GPIO15	10
11	GPIO17 (GPIO_GEN0)	Green	(GPIO_GEN1) GPIO18	12
13	GPIO27 (GPIO_GEN2)	Green	Ground	14
15	GPIO22 (GPIO_GEN3)	Green	(GPIO_GEN4) GPIO23	16
17	3.3v DC Power	Red	(GPIO_GEN5) GPIO24	18
19	GPIO10 (SPI_MOSI)	Purple	Ground	20
21	GPIO09 (SPI_MISO)	Purple	(GPIO_GEN6) GPIO25	22
23	GPIO11 (SPI_CLK)	Purple	(SPI_CE0_N) GPIO08	24
25	Ground	Black	(SPI_CE1_N) GPIO07	26
27	ID_SD (I ² C ID EEPROM)	Yellow	(I ² C ID EEPROM) ID_SC	28
29	GPIO05	Green	Ground	30
31	GPIO06	Green	GPIO12	32
33	GPIO13	Green	Ground	34
35	GPIO19	Green	GPIO16	36
37	GPIO26	Green	GPIO20	38
39	Ground	Black	GPIO21	40

Rev. 2
29/02/2016

www.element14.com/RaspberryPi

Intro to GPIO

wiringPi library

- ✓ GPIO의 쉬운 접근을 위한 library
 - Delay, pinMode, digitalWrite/write 등의 함수 제공
 - 각 GPIO에 wiringPi library를 이용한 pin 접근을 위한 number 부여

```
pi@raspberrypi ~/code/wiringPi $ gpio readall
```

	BCM	wPi	Name	Mode	V	Physical	V	Mode	Name	wPi	BCM	
			3.3v			1	2		5v			
	2	8	SDA.1	IN	1	3	4		5V			
	3	9	SCL.1	IN	1	5	6		0v			
	4	7	GPIO. 7	IN	0	7	8	1	ALT0	TxD	15	14
			0v			9	10	1	ALT0	RxD	16	15
	17	0	GPIO. 0	IN	0	11	12	0	IN	GPIO. 1	1	18
	27	2	GPIO. 2	IN	0	13	14		0v			
	22	3	GPIO. 3	IN	0	15	16	0	IN	GPIO. 4	4	23
			3.3v			17	18	0	IN	GPIO. 5	5	24
	10	12	MOSI	IN	0	19	20		0v			
	9	13	MISO	IN	0	21	22	0	IN	GPIO. 6	6	25
	11	14	SCLK	IN	0	23	24	0	IN	CE0	10	8
			0v			25	26	0	IN	CE1	11	7
	0	30	SDA.0	IN	0	27	28	0	IN	SCL.0	31	1
	5	21	GPIO.21	IN	0	29	30		0v			
	6	22	GPIO.22	IN	0	31	32	0	IN	GPIO.26	26	12
	13	23	GPIO.23	IN	0	33	34		0v			
	19	24	GPIO.24	IN	0	35	36	0	IN	GPIO.27	27	16
	26	25	GPIO.25	IN	0	37	38	0	IN	GPIO.28	28	20
			0v			39	40	0	IN	GPIO.29	29	21

wiringPi library

✓ wiringPi library 설치 방법

- 인터넷이 연결된 상태에서

- `$ sudo apt-get update`
- `$ sudo apt-get install fonts-unfonts-core ibus-hangul git-core -y`
- `$ sudo git clone https://github.com/WiringPi/WiringPi`
- `$ cd wiringPi`
- `$ sudo ./build`
- `$ gpio readall`

✓ Vim 설치 방법

- `$ sudo apt-get install vim`

Intro to GPIO

wiringPi library

✓ Code example

- gcc를 이용한 Compile시에 wiringPi library를 링크해줘야 함
 - gcc -o test GPIO.c -lwiringPi

```
#include <wiringPi.h>

#define SW 0    //wiringPi의 0번pin 접근
#define LED 1   //wiringPi의 1번pin 접근

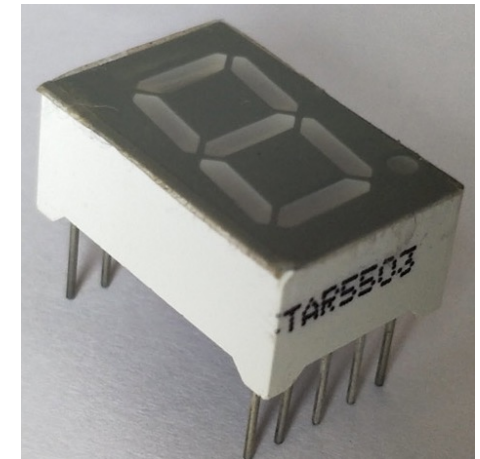
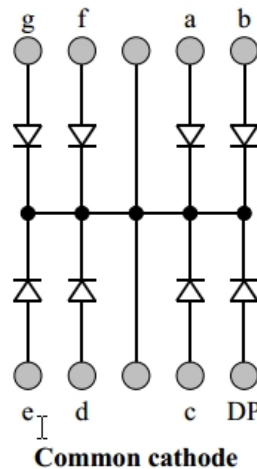
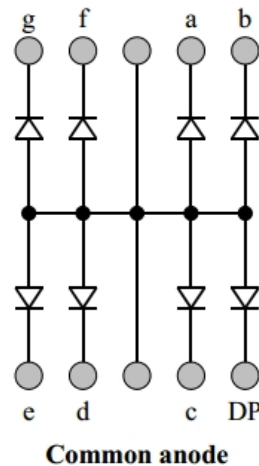
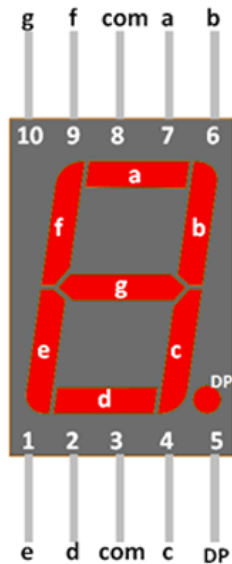
int main(void){
    if (wiringPiSetup()==-1)    //library include 실패시 종료
        return 1;
    pinMode(SW,INPUT);          //switch pin을 input mode로 설정
    pinMode(LED,OUTPUT);        //LED pin을 output mode로 설정

    while(1){                   //프로그램 종료를 방지하기 위한 무한loop
        if(digitalRead(SW)==1){ //switch가 눌러지면 1이 됨
            digitalWrite(LED,1); //LED pin으로 1신호를 내보냄
            delay(1000);          //1초 delay
        }
        digitalWrite(LED,0);     //LED pin으로 0신호를 내보냄
    }
    return 0;
}
```

Intro to GPIO

7-segment 를 이용한 수 표현

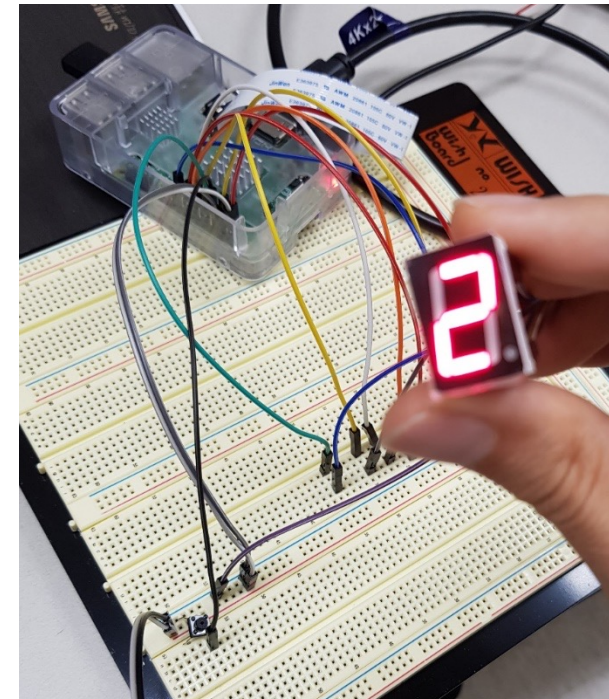
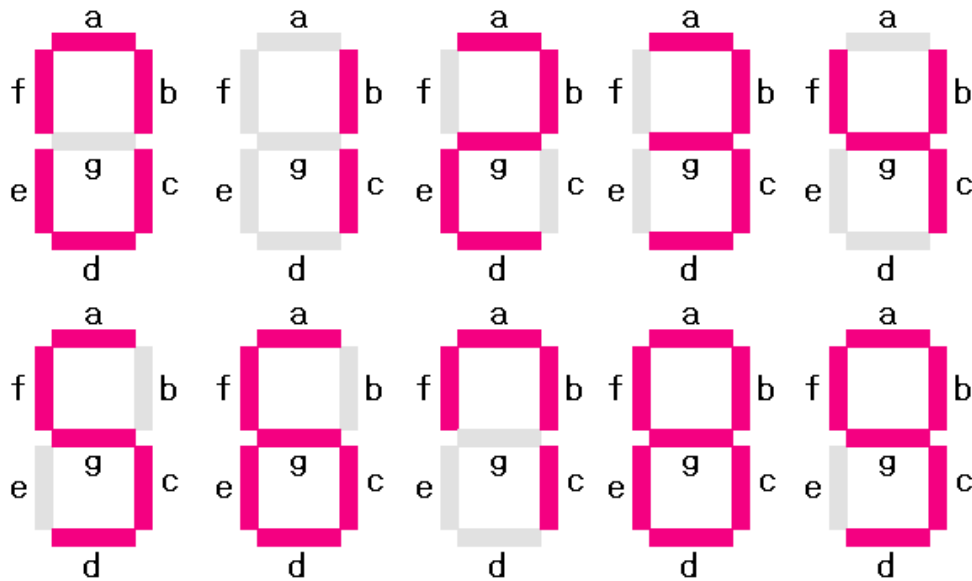
- ✓ 7-segment의 각 LED에 1,0 신호를 보내서 수 표현
- ✓ Common anode, common cathode 두 type 존재
 - 수업에선 common cathode 사용
- ✓ 각 pin과 LED는 아래 그림처럼 연결되어 있음
- ✓ 각 LED pin에 연결 시 220Ω 저항 연결



Intro to GPIO

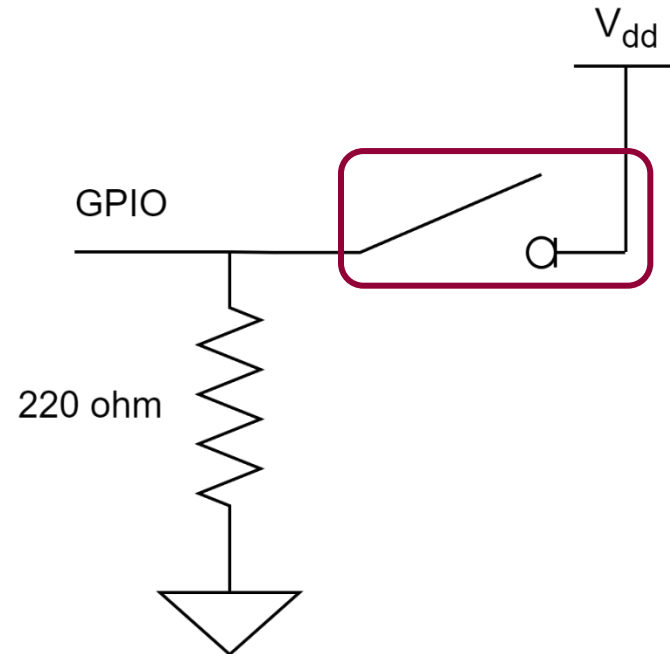
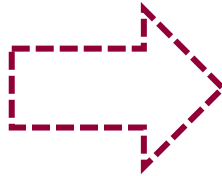
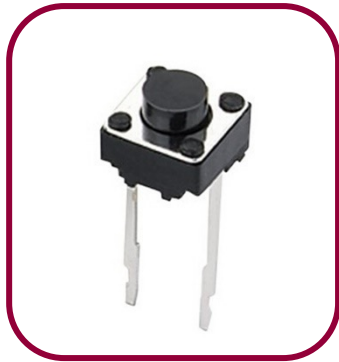
7-segment 를 이용한 수 표현

✓ 7-segment의 각 LED에 1,0 신호를 보내서 수 표현



Intro to GPIO

2 – pin switch 연결 방법



Equipment

Equipment



Rasberry Pi



Bread board



Pack



Mini cable



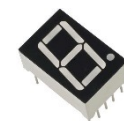
Jumper cable

×18 $\left(\begin{array}{l} \text{FM} \times 13 \\ \text{FF} \times 3 \\ \text{MM} \times 2 \end{array} \right)$



220Ω 저항

×9



7-segment

×1



FT-232

×1



2-pin switch

×1

Basic setting for serial interface

Serial interface를 위한 setting

✓ 키보드 설정

- \$ sudo raspi-config
- 5 Localisation Options
- L3 Change Keyboard Layout
- Generic 101-key PC
- Other
- Korean
- Korean – Korean (101/104 key compatible)
- 처음의 option 창이 뜰 때까지 엔터 연타

Basic setting for serial interface

Serial interface를 위한 setting

✓ 시리얼 통신 enable

- `$ sudo raspi-config`
- 3 Interface Options -> Serial -> Yes
- 다음 키를 차례로 눌러 설정 창에서 나가면 자동으로 재부팅된다.
- → → <Finish> <Yes>
- 재부팅이 안된다면 콘솔 창에 다음을 입력
- `$ sudo reboot`

Basic setting for serial interface

Serial interface를 위한 setting

✓ UART 사용을 위한 라즈베리파이 설정 (Raspberry pi 3 기준)

- `$ sudo nano /boot/cmdline.txt`
- 열린 파일에서 다음 내용을 삭제 후 저장
 - `Console=serial0,115200`
- `Ctrl+O / Enter / Ctrl+X` 를 차례로 입력해서 콘솔창으로 복귀.
- `$ sudo systemctl stop serial-getty@ttyS0.service`
`$ sudo systemctl disable serial-getty@ttyS0.service`
- RPi3 에서는 bluetooth 설정을 해제해야 함. 다음의 명령어로 config.txt 접속
- `$ sudo nano /boot/config.txt`
- 맨 아래에 다음 명령어 삽입
- `#disable bluetooth`
`dtoverlay=pi3-disable-bt`
- `Ctrl+O / Enter / Ctrl+X` 로 콘솔 창으로 복귀. 다음 커맨드 입력.
- `$ sudo systemctl disable hciuart`

Basic setting for serial interface

Serial interface를 위한 setting

✓ UART 사용을 위한 라즈베리 파이 설정

- \$ sudo nano /boot/config.txt
- 맨 아래 다음 명령어가 올바르게 되어 있는 지 확인 or 설정
 - Enable_uart=1
- Ctrl+O / Enter / Ctrl+X 로 콘솔 창으로 복귀.

✓ UART 사용을 위한 C 파일(~ polling.c & interrupt.c) (Raspberry pi 3 기준)

- fd = open(" /dev/serial0 " , O_RDWR|O_NOCTTY); 로 되어있는지 확인하기

❖ 본인의 라즈베리 파이 버전을 꼭 확인할 것!!

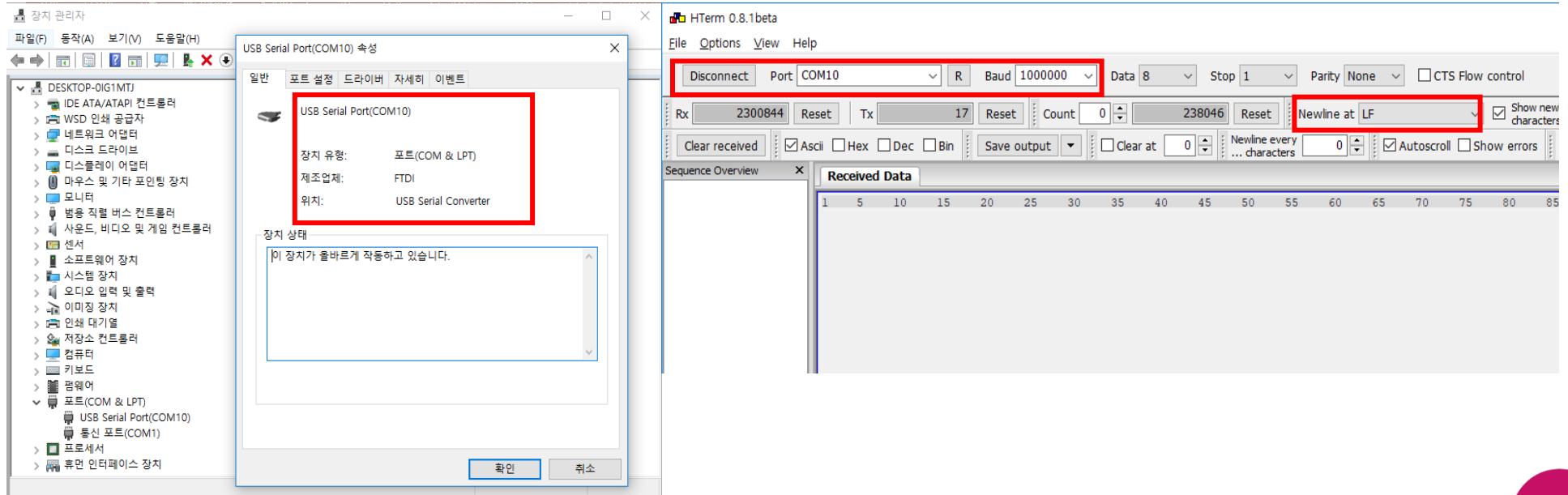
- 위의 설정은 모두 Raspberry pi 3 기준

○

Terminal program

PC terminal program

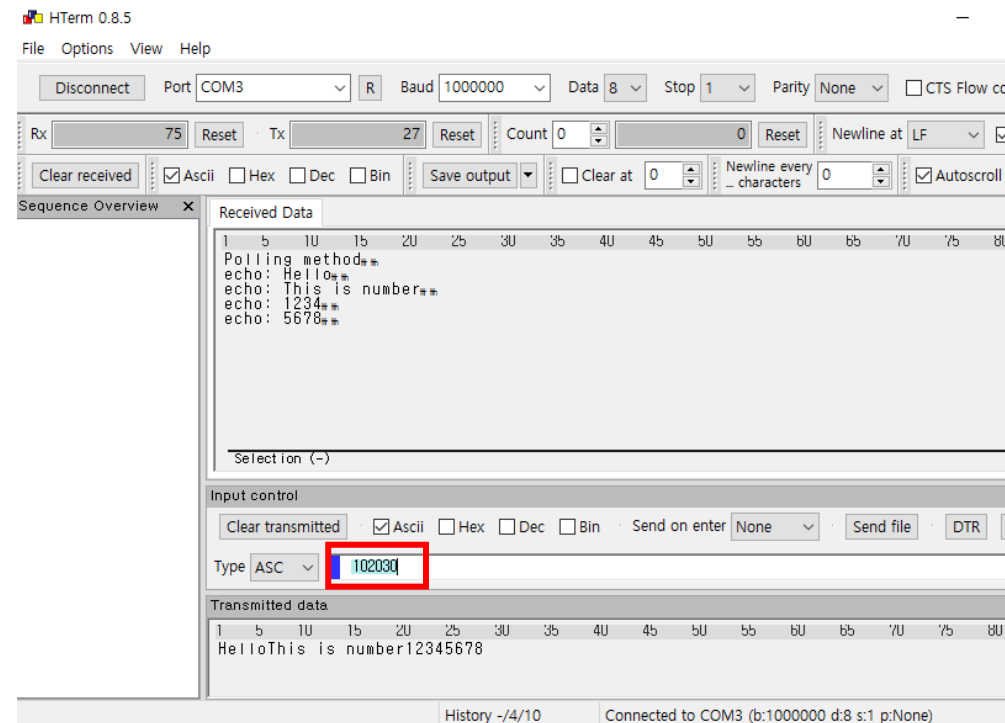
- ✓ Hterm 다운로드 (*Mac 사용 시 유사 프로그램으로 대체)
 - <http://www.der-hammer.info/terminal/>
- ✓ Hterm 실행 후 port를 FT232 모듈의 COM 번호로 설정
- ✓ Connect 누른 후 Baud에 1000000 입력 후 엔터
- ✓ Newline at LF 설정
- ✓ Connect 버튼 클릭하여 연결



Test codes

Polling test

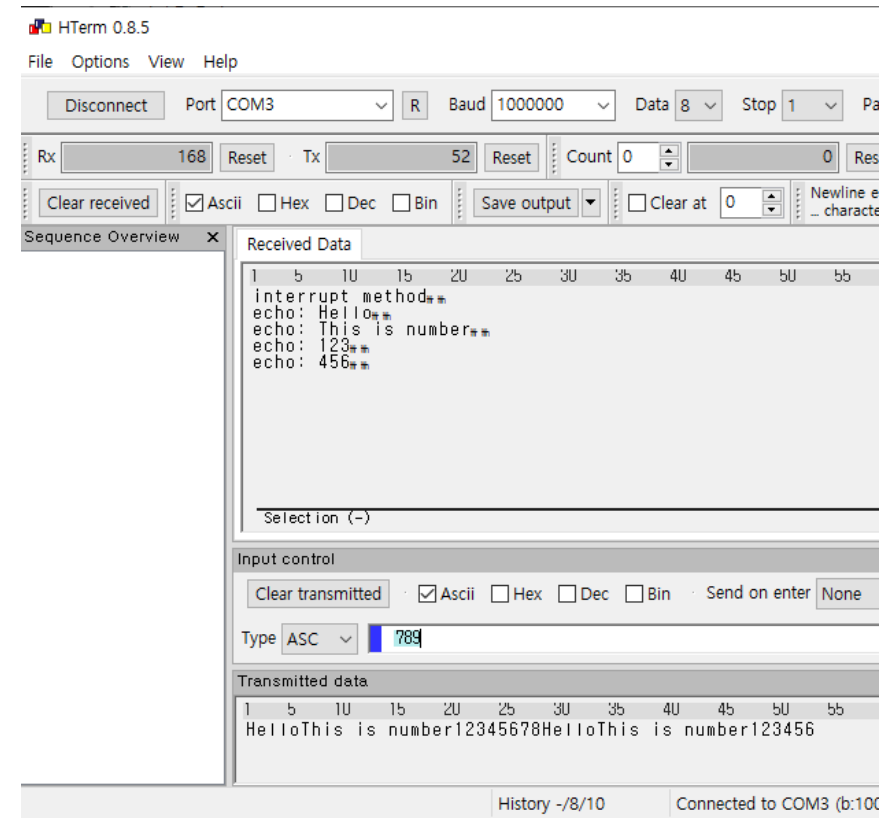
- ✓ 라즈베리파이에서 Echo_polling.c 를 실행
 - 컴파일된 파일은 권한 문제가 있으므로 sudo로 실행
 - `$ gcc echo_polling.c -lwiringPi`
 - `$ sudo ./a.out`
 - Type에 data를 입력해서
PC에서 라즈베리파이로 data 전송
 - 라즈베리파이에서 일정 주기마다
들어온 data를 확인해서 PC로 return



Test codes

Interrupt test

- ✓ 라즈베리파이에서 Echo_interrupt.c 를 실행
 - 컴파일된 파일은 권한 문제가 있으므로 sudo로 실행
 - `$ gcc echo_interrupt.c -lwiringPi`
 - `$ sudo ./a.out`
 - PC에서 라즈베리파이로 data 전송
 - Data가 들어오면 interrupt를 발생시켜
들어온 data를 PC로 바로 return



Problem Definition

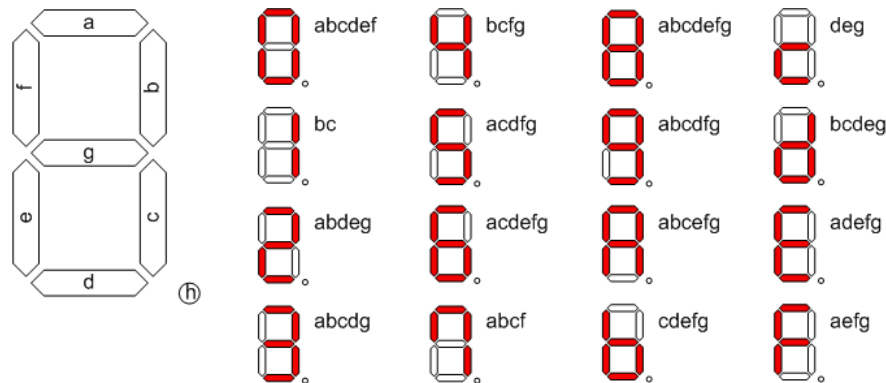
Problem Definition

1. GPIO를 이용한 7-segment 제어

1-1. GPIO pin을 사용하여 1개의 입력 스위치, 1개의 출력 7-segment 제어

1-2. 스위치 누른 횟수를 7-segment에 1자리 16진수 숫자로 출력
(7-segment 상에 0 ~ F 까지 증가할 것, F 이후에는 다시 0으로)

1-3. 파일명: 학번_gpio.c



Problem Definition

2. Serial interface를 이용한 7-segment 제어

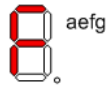
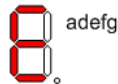
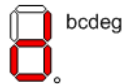
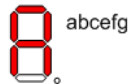
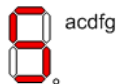
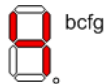
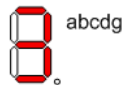
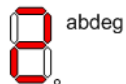
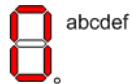
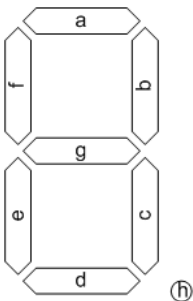
2-1. PC에서 라즈베리파이로 전송한 data를 1개의 7segment에 출력

2-2. 다음 문자가 들어올 때 까지 이전의 문자 계속 출력 하며, 한번에 여러 문자가 들어오면 가장 앞 글자를 출력

2-3. 0~9, A,B,C,D,E,F는 아래 그림처럼 출력하고, 다른 문자가 들어올 경우, 아래 우측 처럼 X를 출력

2-4. 제공받은 echo_polling.c와 echo_interrupt.c 파일을 7-segment에 문자를 출력하도록 수정해서 과제 수행

2-5. C 파일 내의 **task** 함수 수정 금지



Submission & Evaluation

조교가 검사할 수 있는 source code와 결과보고서 PDF를 제출

- ✓ Due date: 5/7(화) 23:59
- ✓ 제출 방식: 학번_이름.zip 파일의 형식으로 plms에 제출
 - 첫번째 과제 source code: 학번_gpio.c
 - 두번째 과제 source code: 학번_echo_polling.c, 학번_echo_interrupt.c
 - 결과 보고서: 학번_이름.pdf
- ✓ 실험에 관한 질문은 Q&A 게시판 활용
- ✓ 담당조교 – 신동연 (shindy@postech.ac.kr)

다음 내용을 포함하여 결과보고서 작성

1. 어떤 pin을 스위치와 7-segment에 할당하였는지 설명
2. UART 통신 방식에 대한 조사 후 Serial interface를 통한 7-segment 제어 방식 설명
3. 어떤 방법으로 serial interface 내부의 함수를 구현하였는지 설명

Demo

- ✓ 5/7(화) 정규 수업시간 (15:30~16:45)
- ✓ 7-segment 제어 구현에 대한 과제 모두 LG동 전산실에서 Demo로 평가

100점 만점으로 채점하며 다음 사안을 고려

1. 스위치 입력 구현 (10)
2. 스위치 입력에 대한 7-segment 출력 구현 (30)
3. Serial interface에서 7-segment에 문자 출력 구현 (40)
4. 보고서 (20)

부정행위 적발 시 -100점 적용