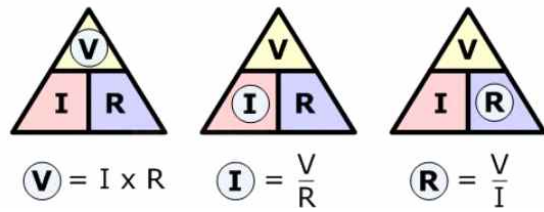


<하드웨어>

정공 : 전자를 잃은 빈자리

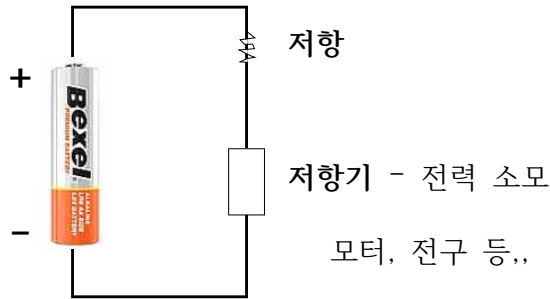
전류 : (-)에서 (+)로 흐름,,(전자의 이동) (이론상 +에서 -로 흐른다고 배움,,)

옴의 법칙 :  $V=IR$  (V:전압, I:전류, R:저항)



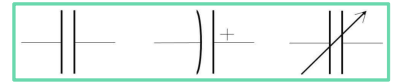
정격전력 : 최대치 전력

저항 : 전류의 흐름을 막음 -> 전압차가 생김



콘덴서 : 전류가 흐르는 공간을 띄워놓고, 흐르지 못하는 것으로 채움 → 전류가 포집이 됨 → ‘댐’처럼 됨 (캐패시터) 고주파는 거른다..(노이즈 제거) 필터역할 → 깨끗한 DC전류를 만들.

ex. 모터에서 갑자기 큰 전류를 필요로 할 때 전류를 충분히 보내주기 위함.



정전용량 : 콘덴서가 모을 수 있는 전류량

커플링 : 노이즈. 증폭기역할(ex.스피커) // 디커플링 : 노이즈제거

인덕터 : ac → dc. 유도전류 이용하여 낮은전류 ↔ 높은전류로 전환. 자기 ↔ 전기 ft.플레밍

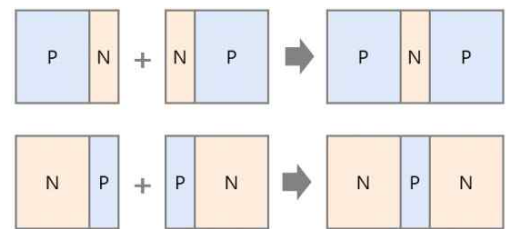
P ; positive(Anode, S극, +), N ; Negative(Cathode, N극, -)

다이오드 : 전류를 한방향으로 흐르게 함

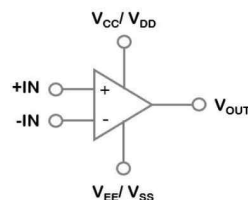
트랜지스터 : 증폭기. BJT타입, FET타입

다이오드+다이오드 = 트랜지스터 (PNP, NPN)

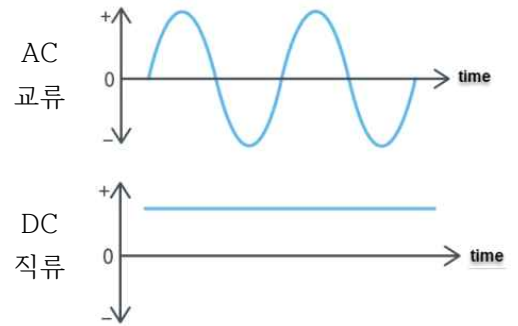
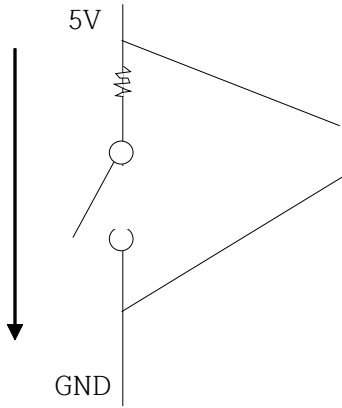
작은 전류로 큰 전류를 제어



연산증폭기(OP Amp) : 전류 증폭..안정적으로 제공



하이 임피던스 High impedance 상태,, 0인지 1인지 알 수 없음 → 초기값 필요



DC에선 (-)는 없음

+ : VCC, VDD // 0 : GND (ground)

필요한 전력이 다 다름 → 칩으로 만듦 (PMIC) - 전력을 분배

**PWM** ; Pulse Width Modulation - 디지털로 아날로그를 흉내냄

ex. 0.5초는 1, 0.5초는 0으로 하면, 평균적으로 0.5가 됨... (LED 밝기 중간정도로 하기)

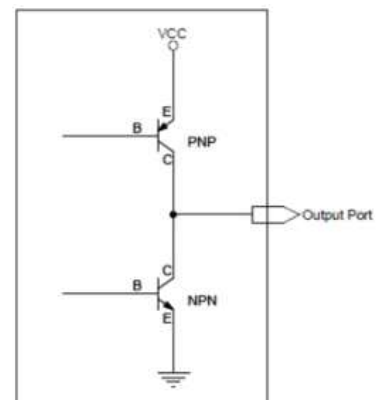
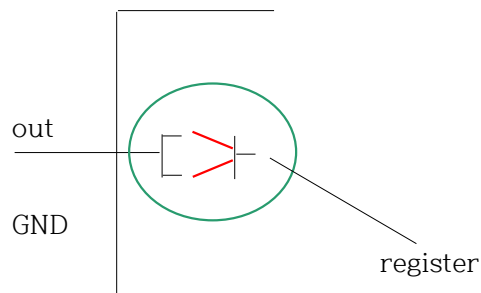
- 소프트웨어적으로도 가능하지만, 지연시간이 있어서 정교하지 못함 → PWM 컨트롤러가 있음..
- 칩끼리 통신 spi, i2c // uart - 컴퓨터끼리 통신 연결

gpio - 제어/cu,

general purpose io(input/output)

gpio가 기본이고, 부가기능을 더 붙일 수 있음 alternate

push-pull 방식

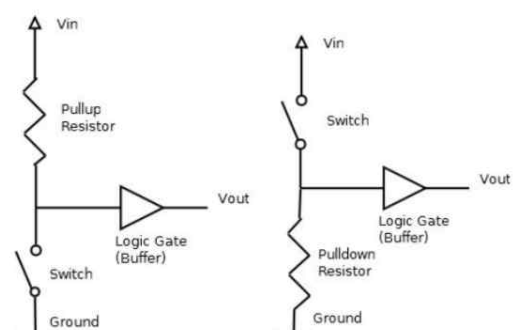


BJT 회로

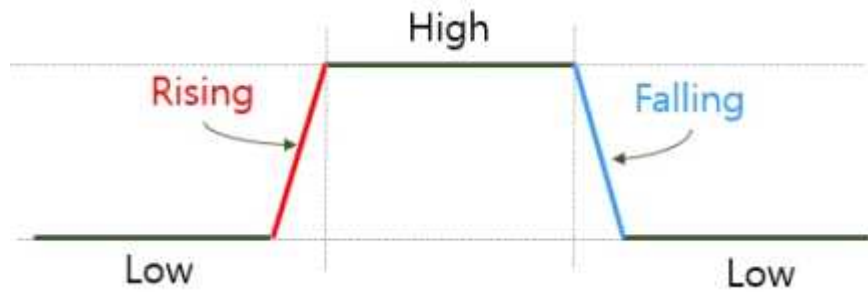
7번 핀 읽기

$r = S \gg 7$  쉬프트 연산후,  $\&0x01$  출력? (마스크연산)

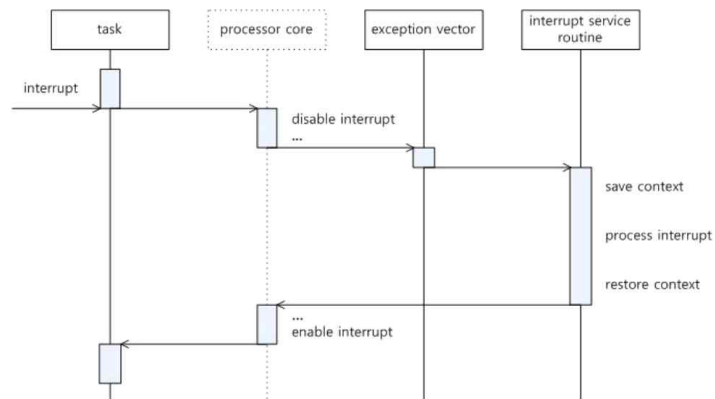
풀업/풀다운 - 저항이 위/아래 위치,, active Low/ High 스위치 => 입력신호



신호 low ↔ high  
 rising, falling ⇒ edge  
 (active High에서)  
 // ↔ active Low



## 인터럽트



## ADC 아날로그→디지털

샘플링 - 신호를 잘게 쪼갬 (양자화)

동기방식 - 데이터라인과 클럭라인이 연결. 저속,노멀,고속(클럭 조절)

비동기방식 - 데이터라인만 있음. 각자의 클럭으로 맞춰 사용. ex. UART

i2c ⇒ low, high, (low→high), (high→low) 4가지 신호

동작안할땐 high상태

CPU - Master / 주변장치 - Slave

주소	0x04	<table border="1"><tr><td>1</td><td>0</td><td>0</td></tr></table>	1	0	0	0x06	<table border="1"><tr><td>1</td><td>1</td><td>0</td></tr></table>	1	1	0
	1	0	0							
1	1	0								
0x05		0x07								

마지막은 읽기/쓰기 비트

<센서제어>

입력, 출력

wiringPiSetupGpio()

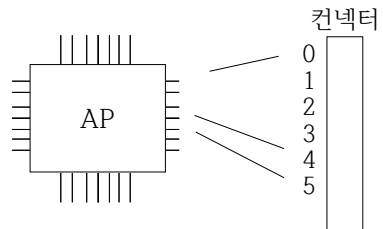
GPIO 번호

- 핀 맵

// 회로도, 칩 정보(종류)

AP : Application Processor

GPIO PIN



PyC Basic 핀 정보

const uint8

{15, 16, 17, 6, 19, 20, 21, 22}

{7, 27}

{13}

{18}

{3}

{1}

{2}

{0}

{0x40}

{0x39}

{0x3C}

LED

Push Switch

Piezo

Led Strip

PSD

CDS

Sound

POTENTIOMETER

Temp / Humi

Gesture

OLED

GPIO

GPIO

HwPwm

HwPwm

SpiAdc

SpiAdc

SpiAdc

SpiAdc

I2C

I2C

I2C

main.cpp

>> gcc -o main main.cpp -lwiringPi

>> ./main

```
#include <stdio.h>
#include <wiringPi.h>

#define LED 20

int main(void)
{
    if(wiringPiSetup()==-1)
        return 1;

    pinMode(LED, OUTPUT);

    while(TRUE)
    {
        printf("LED HIGH\n");
        digitalWrite(LED, HIGH);
        delay(1000);

        printf("LED LOW\n");
        digitalWrite(LED, LOW);
        delay(1000);
    }

    return 0;
}
```