

기공식

1. 아두이노 기초 소개

2. 아두이노 소개

아두이노 UNO / DUE / TRE : 간단한 형태의 하드웨어
 1번 2번 3번 (납땀 없이 프로그램 작성가능)
 (286 CPU, 8bit) ↑ ↑
 32bit 아두이노 + 비록 볼록한 것임

스크래치 : 블록을 이용해 쉽게 프로그래밍 가능

프로그래밍 : 예제 하나만 봐도 이 미디어 아트를 쉽게 프로그래밍 할 수 있다.

아두이노 : 일반인과 예술가들이 쉽게 만질 수 있는 하드웨어 플랫폼
 (오픈소스 H/W) → 인터페이스 공개된 소스를 내려 받아 쉽게 S/W 구성 가능

오픈소스 H/W
 블록 { 아두이노 (8bit CPU)
 라즈베리 파이 (리눅스)
 비록 볼록 (리눅스 CPU)

가상화 (웨어블 플랫폼, 아두이노 호환 가능)

아두이노 하드 : 다양한 하드웨어 플랫폼으로 기능 확장 가능 (적용성)

센서 및 액추에이터 : 외부의 어떤 아날로그 값의 변화로 감지하여 (센서)
 액추에이터로 해당하는 반응을 내보내기 가능

브레드 보드 : 납땀 없이 선을 연결해 회로 구성 가능 (하드웨어)
 쉽게 구성 가능

No.

Date. / /

2. 아두이노

기초 설치

<https://www.arduino.cc> 접속

software → downloads → Arduino IDE 설치

설치 파일 실행 후 그대로 설치 (Arduino Setup)

arduino : C++ 문법 사용, 코딩 가능

void setup() { ^{실행하는 코드} 초기화 / 변수 선언 /

: 프로그램 최초 1회만 실행함

반드시
있어야함

void loop() { ^{실행하는 코드} 실행할 코드

: 아두이노 실행시 항상 반복해서 실행

// 주석 : 실행 X, 코드에 대한 설명을 작성

3. 아두이노

기초 전자

아두이노 : 입출력 기능을 갖춘 소형 컴퓨터

- 아두이노 보드에 다양한 센서/모듈을 설치하여 기능을 추가할 수 있음

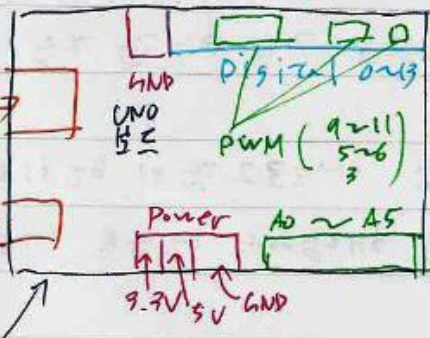
- 저렴한 가격 / 쉬운 사용법

아두이노 부품 : LED / 초음파 센서 / 서보 모터 / 압력 센서 (버튼) / 스피커 등
저항 / 브레드 보드 / 케이블 등

아두이노 핀

USB
연결

전원
어댑터
연결



5~12V 어댑터 사용 가능

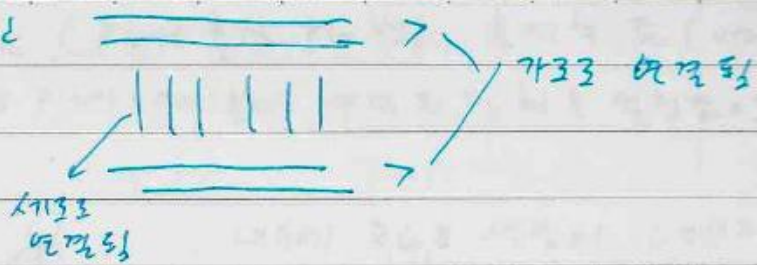
1) power: +와 -

5V, 3.3V, GND (3개)
+극 -극

2) 디지털 핀 : 0~31번 (입력, 출력 공통)

3) 아날로그 핀 : A0~A5 (입력)
PWM ~ (출력)

bread board



LED



: 2V의 전압이 공급되어야함 (다른 연결이 저항 필요함)

↳ 전압이 생기는 라인을 공급시 열/연기 발생

아두이노 보드에 순서 ↑

저항

: 220Ω (빨간색 - 빨간색 - 갈색) / 대부분 사용

power의 +극 (3.3V 또는 5V) 을 브레드 보드의 가로 줄에 연결,
 power의 -극 (GND) 을 브레드 보드의 나머지 가로 줄에 연결

↳ 각각의 연결하는 부품들을 선과 저항을 사용하여 서로 줄에
 놓고 가로줄과 연결한다.



4. 아두이노

PWM 채널

출력 : 아두이노 → 작동상태

전기 0 : 작동 0

전기 X : 작동 X

입력 : 외부 환경 → 센서 → 아두이노

1. 센서가 외부 환경의 값을 측정함

2. 측정된 값을 아두이노가 읽어들이고

디지털 : 0 또는 1로 이루어짐 (LED / 스위치 등)


1 : high, on (켜짐)

0 : low, off (꺼짐)

아날로그 : 중간값들이 존재함 (LED / 모터 등)

최대 ~ 최소값 사이에 연속적인 값 존재

analogWrite (PWM 출력 pin 번호, 출력할 값 (0~255));
 ↳ 해당 pin이 (PWM 출력) 연결된 광다이오드 센싱값으로 출력함

3색 LED :  : 내부에 R, G, B 색상의 LED가 모두 존재함
 (각각 PWM 핀에 연결해야 함)
 R G B
 ↓ (각각 색상의 +극) → 자체 회로에 저항 네개씩 ^{추가적} 저항 연결 필요함
 GND 연결 0~255 출력 가능

(analogWrite (red Led pwm pin 번호, 빨간색 밝기 값);
 analogWrite (green Led pwm pin 번호, 초록색 밝기 값);
 analogWrite (blue Led pwm pin 번호, 파란색 밝기 값);)
 ↳ R, G, B 색의 조합으로 여러색을 나타낼 수 있음

delay (대기시간 (ms)); : 입력한 시간만큼 대기함
 random (n); : 0 ~ n-1 까지의 범위를 무작위값으로 구함

randomSeed (analogRead (아날로그 입력 pin)) : 랜덤값의 시드값 ^{seed} 생성함
 ↳ random () 함수를 항상 초기 무작위로 내보내함 (setup에 사용)

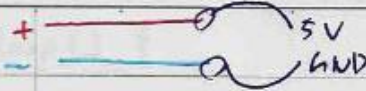
5. 아두이노

초음파센서

초음파 센서 : 초음파를 이용해서 사물이나 벽까지의 거리를 측정하는 센서

- 초음파를 발사하는 부분 (Trig), 반는 부분 (echo) 으로 구성됨
- 일정한 속도로 돌아와 echo 핀의 전압이 high 되었을 때 거리의 시간을 거리로 환산하여 구할 수 있음
- 경우에 따라 큰 값 범위가 달라질 수 있음

센서원리 : Trig 핀이 high 되면 초음파가 발사되고, 사물이나 벽을 만나고 돌아와 echo 핀의 전압이 high 되었을 때 거리의 시간을 거리로 환산하여 구할 수 있음

1.  : 전원과 브레드 보드를 연결한다.

2. LED와 저항을 구성한다. (빨간색 : 디지털 13, 노랑 : 디지털 10, 파랑 : 디지털 7)

3. 초음파 센서를 가르고 잘른다. (각각 V_{CC} , Trig, echo, GND)

4. 초음파 센서의 $V_{CC}(+)$ 를 5V와 연결된 브레드보드 라인에

5. 초음파 센서의 GND(-)를 GND와 연결된 브레드보드

6. 초음파 센서의 Trig 핀을 디지털 라인에

7. 초음파 센서의 Echo 핀을 디지털

라인에 연결한다.

↑ H/W
↓ S/W

핀 번호

4번부터
21까지

`int Trig = 2, echo = 3;`

`int rLED = 13, yLED = 10, bLED = 7;`

`void setup() { serial.begin(9600);` ← 시리얼 통신 설정

`pinMode(Trig, OUTPUT)`

`pinMode(echo, INPUT)`

`pinMode(rLED, OUTPUT)`

`pinMode(yLED, OUTPUT)`

`pinMode(bLED, OUTPUT)`

`}`


```
void loop() {
    float duration, distance;
```

초음파 센서

```
    { digitalWrite(trig, HIGH);
      delay(1);           ← 1초 동안 HIGH 상태 유지
      digitalWrite(trig, LOW);
```

시각용 거리
cm 단위로 표시

```
    { duration = ((float)(duration * 340) / 10000) / 2;
```

시리얼 모니터에 거리 출력

```
    { Serial.print("거리 : ");
      Serial.print(distance);
      Serial.print("cm");
```

→ 거리 : (초음파 센서 거리) cm
단위로 출력

```
    if (distance <= 10) {
```

```
    }
```

```
    else if (distance <= 20) {
```

```
    }
```

```
    else {
```

```
    }
```

→ 거리에 따라 LED 거리
digitalWrite() 사용

코드 작성 후 컴파일
아두이노로 전송된 코드를 보드에 업로드하여 시리얼 모니터와 아두이노(기기)를
동시에 테스트한다.

6. 아두이노

모터 방향을 전환

할 속도제어

L293D



output 1, 2 → 모터에 연결

Vcc → 모터에 공급하는 전원 연결

GND → -극 연결 (GND)

enable → +5V에 연결 (IC 칩 작동)

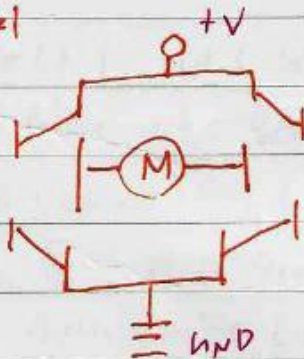
Enable 1	1	16	Vcc
Input 1	2	15	Input 4
output 1	3	14	output 4
GND	4	17	GND
GND	5	12	GND
Output 2	6	11	output 3
Input 2	7	10	Input 3
Vcc	8	9	enable 2

모터 1개만

회전 방향

제어

H 브리지

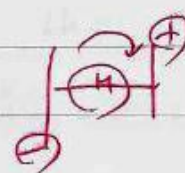


회전 방향: 정전



정전

(시계 방향으로 회전)



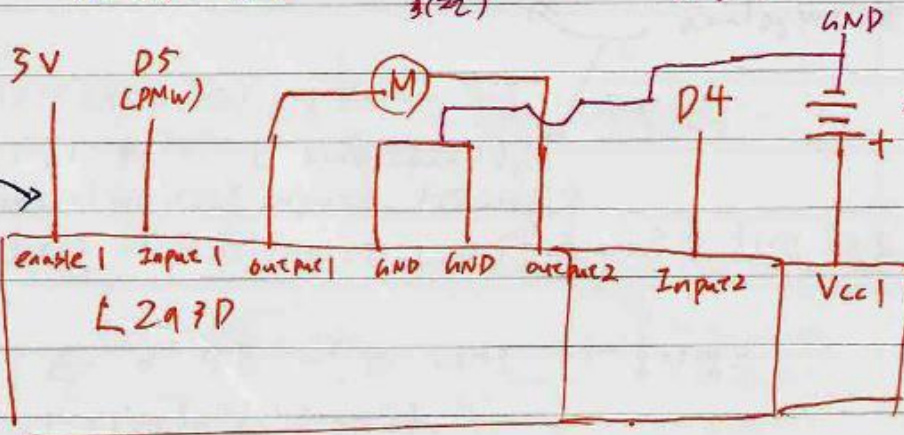
정전

(시계 방향으로 회전)



정전

(시계 방향으로 회전)



D9

PWM으로
제어 연결시

모터 속도 조절 가능

← 모터를 구동시키기
위해 추가 전원
필요

Input 1 High Low

Input 2 Low High

시계 방향으로
회전시계 방향으로
회전

핀 번호
변수에
지정함

```
int input1 = 4;
int input2 = 5;
```

void setup() {

Serial.begin(9600); ← 시리얼 통신 설정

핀 모드
설정

```
pinMode(input1, OUTPUT);
pinMode(input2, OUTPUT);
Serial.println(" ");
```

pinMode(9, OUTPUT); ← 4533228
← 조작법 출력

void loop() { if (Serial.available()) {

char ch = Serial.read(); ← ch에 입력된 문자 저장

if (ch == '+') { ← '+'가 들어옴

Serial.print("forward");

DigitalWrite(input1, LOW);

DigitalWrite(input2, HIGH);

else if (ch == '-') { ← '-'가 들어옴

Serial.println("backward");

DigitalWrite(input1, HIGH);

DigitalWrite(input2, LOW); ← 후진 시(2번 작동)

else { ← '+' 또는 '-'가 아닌 키가 들어옴

Serial.println("stop");

DigitalWrite(input1, LOW);

DigitalWrite(input2, LOW);

7. TINKERCAD 설치, : 아두이노 없이 회로 구성 및 시뮬레이션 가능 (인터넷 연결 필요)

사용, 응용 등

<https://www.tinkercad.com> 에 접속

↳ 스크롤 내리시 프로그래밍, 시뮬레이션 밑 3집 (전과장님!)

아두이노
코딩 블록 : 블록 코딩 제공 → 아두이노 코드로 전환 가능

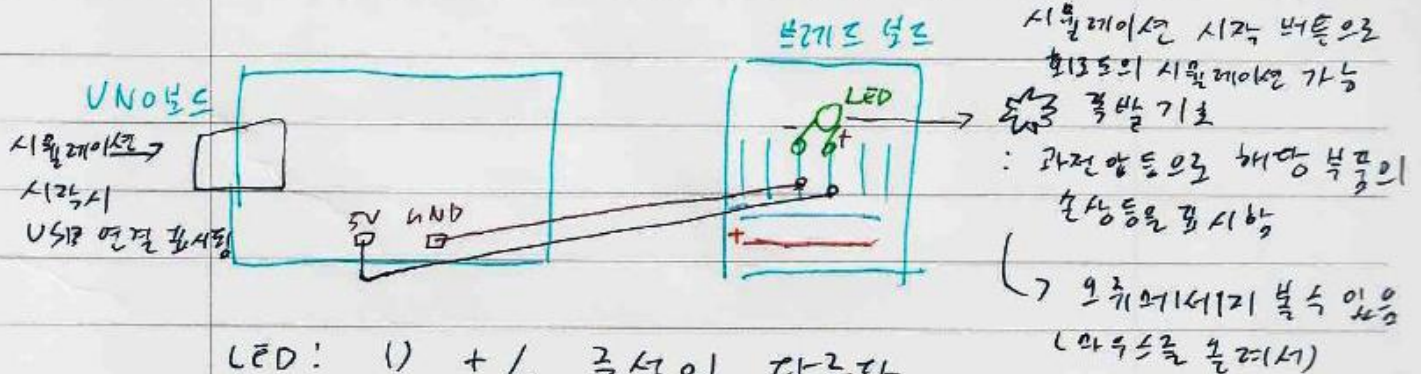
회원가입 → 개인 계정 생성 → 로그인

Circuits → 새 회로 작성 → 회로 구성 가능

회로 삭제 이전 다음
(도움말)

회로 구성 요소 목록에서 드래그로 가져올 수 있음

LED 연결시 이름과 색상 지정 가능



LED: 1) + / - 극성이 다르다
극의 차이
2) 저항과 함께 사용해야 함

