

Chapter 08

따라 하면서 배우는

아두이노

아두이노 메가2560으로 마이크로컨트롤러 시작하기

ESP32 아두이노

Wemos D1 R32 소개

■ 아두이노 Wemos D1 R32

- 아두이노와 형태는 같지만, ESP32칩을 사용

Features

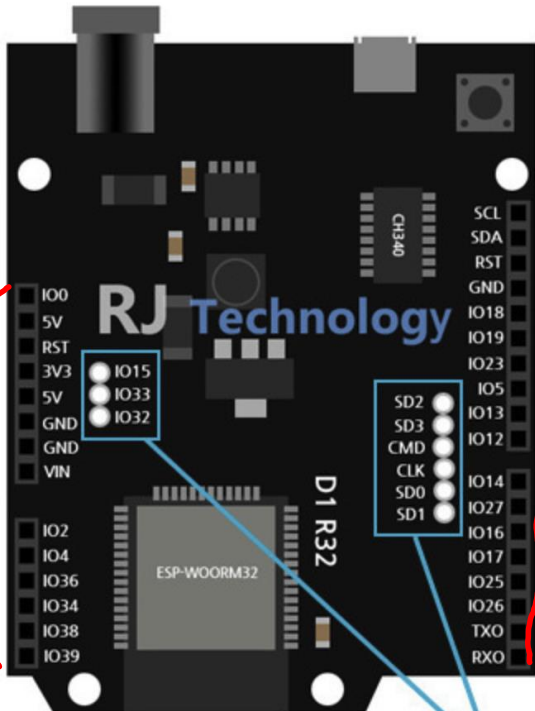
DC 5V-12V

1 analog input(3.2V max input)

Micro USB connection

Compatible with Arduino

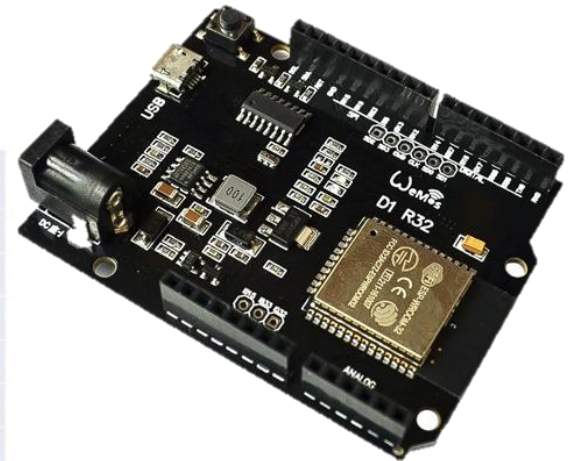
WiFi+Bluetooth+UNO



헤더핀 납땜후 확장가능

Pinagem Wemos D1 R32

IO0	SCL
5V	SDA
RST	RST
3V3	GND
5V	IO18
GND	IO19
GND	IO23
VIN	IO5
	IO13
	IO12
IO2	
IO4	
IO36	IO14
IO34	IO27
IO38	IO16
IO39	IO17
	IO25
	IO26
	TX0
	RX0

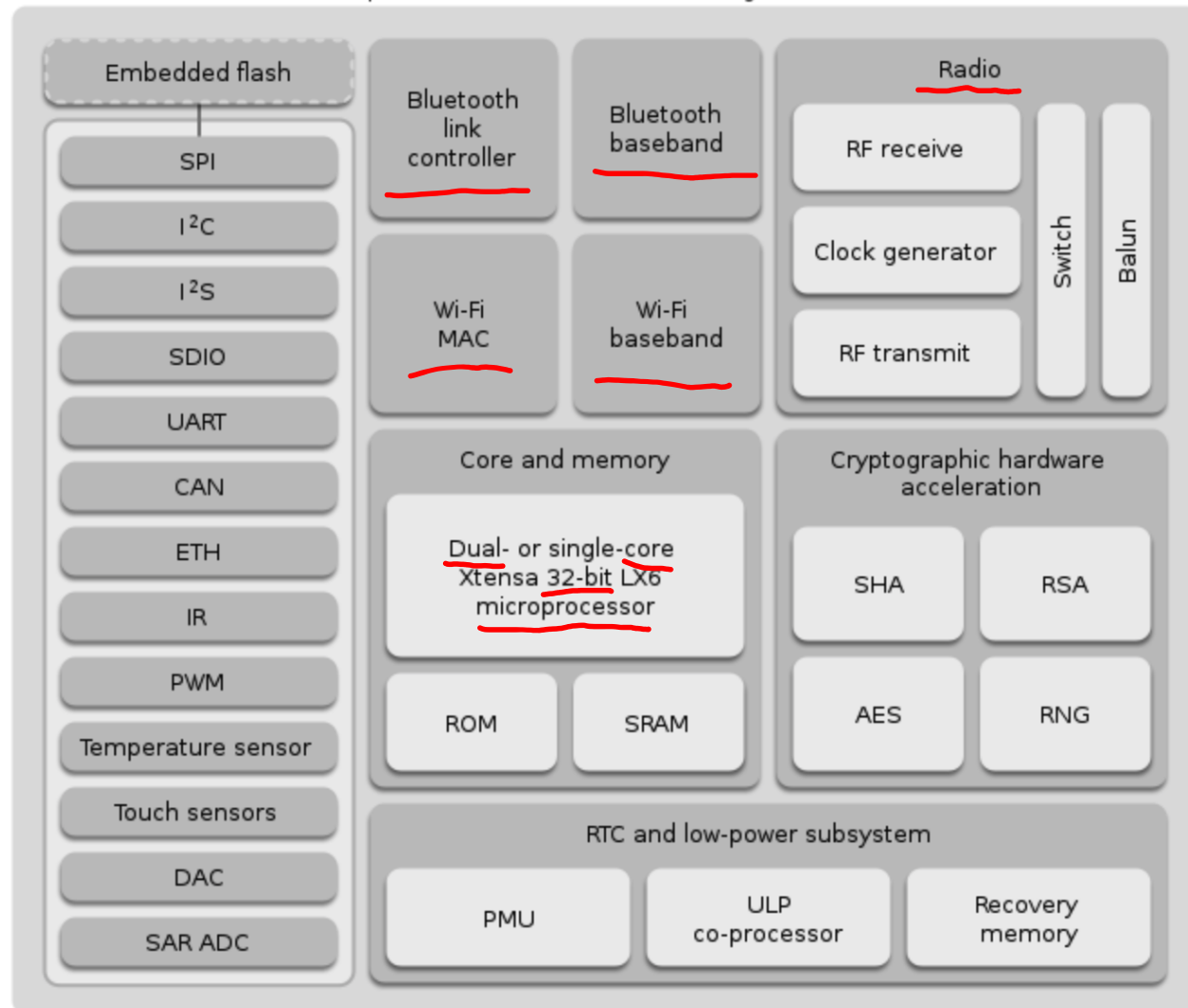


Wemos D1 R32 소개

■ 아두이노 Wemos D1 R32

- ESP32 Function Block

Espressif ESP32 Function Block Diagram



■ 아두이노 Wemos D1 R32

- https://docs.wemos.cc/en/latest/ch340_driver.html

Docs » CH340 Driver

 Edit on GitHub

CH340 Driver

- Windows V3.5
- Mac OSX V1.5

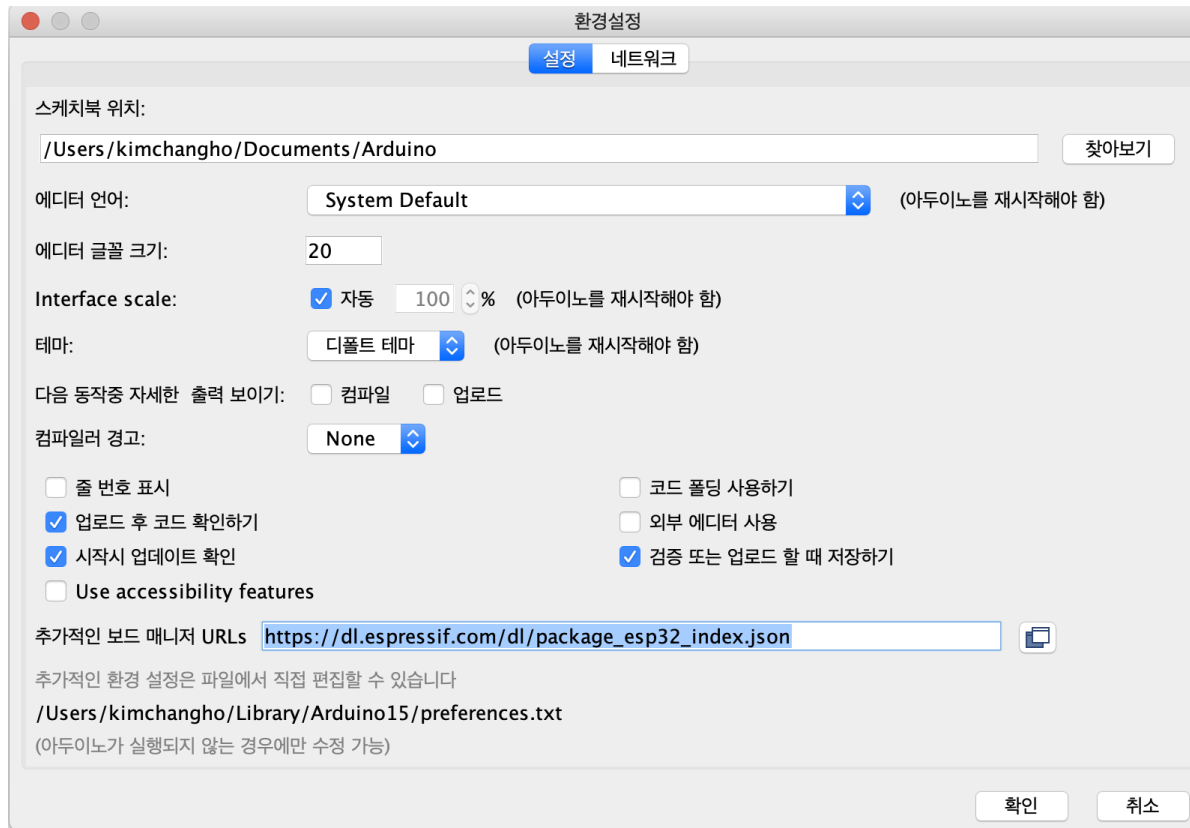
© Copyright 2019, wemos.cc Revision d994a5da.

Built with [Sphinx](#) using a [theme](#) provided by [Read the Docs](#).

Wemos D1 R32 설치

■ 아두이노 Wemos D1 R32

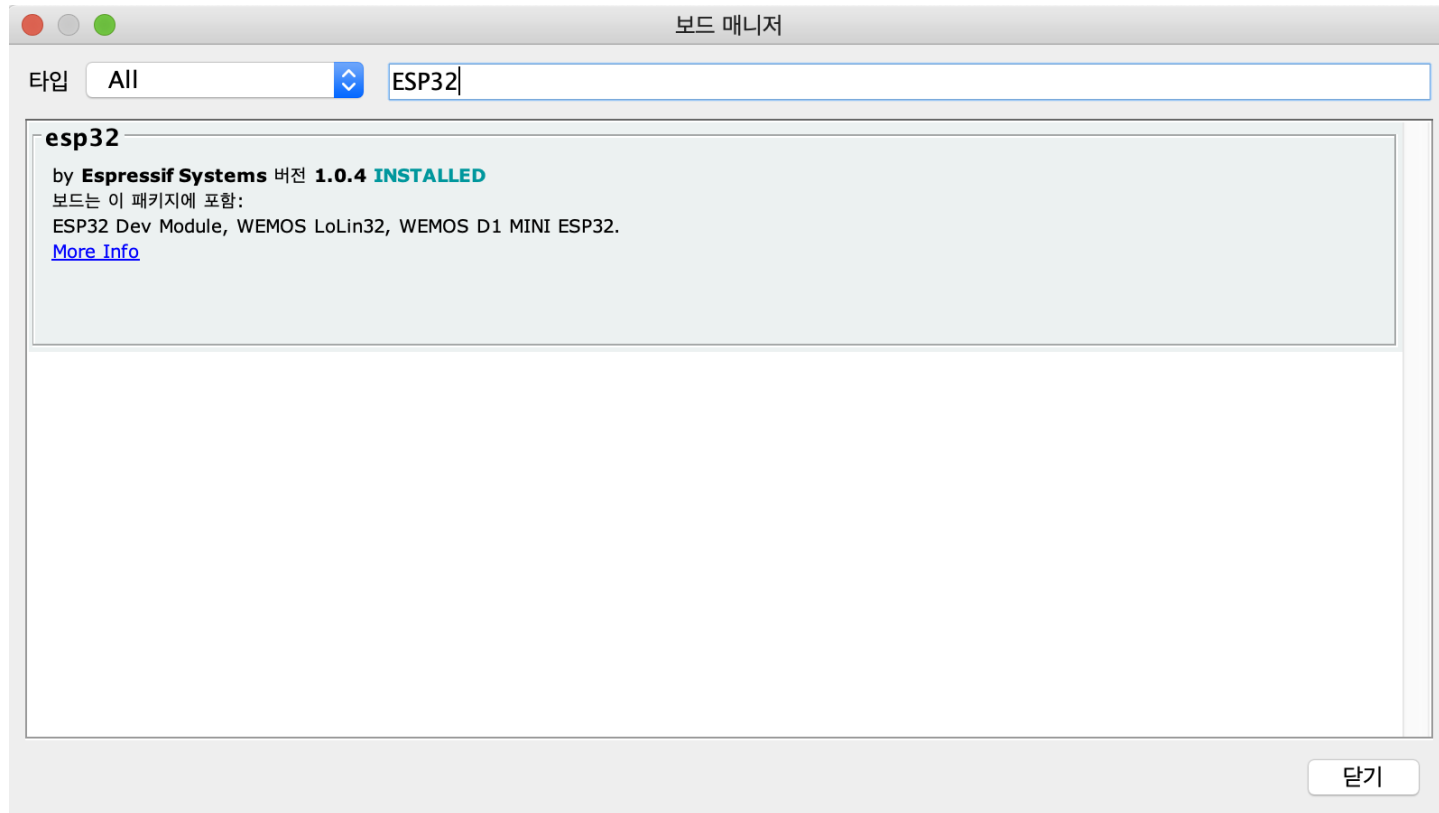
- https://dl.espressif.com/dl/package_esp32_index.json



Wemos D1 R32 설치

■ 아두이노 Wemos D1 R32

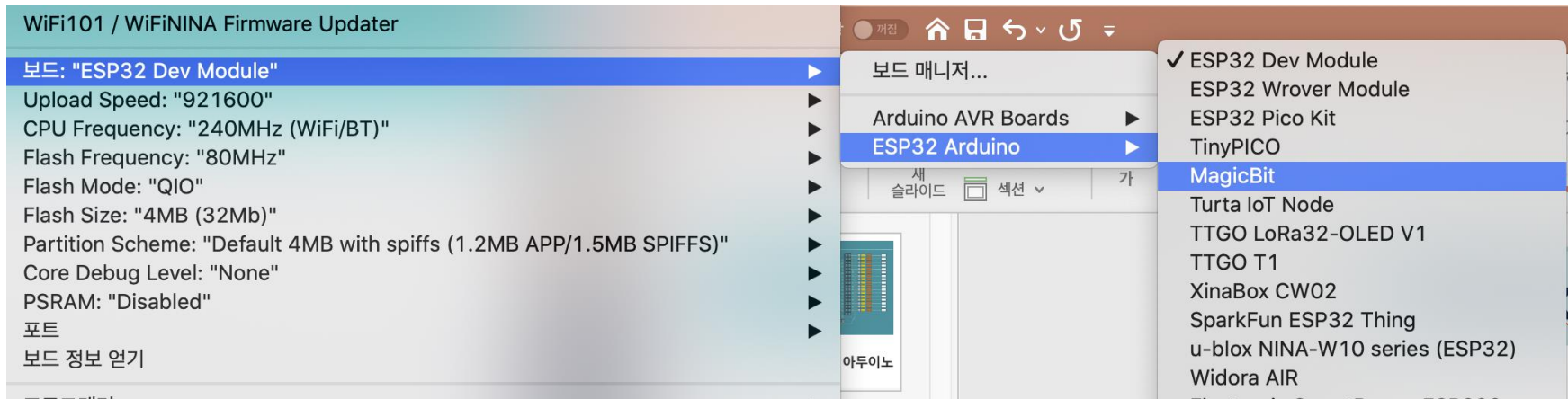
- 툴 -> 보드 -> 보드매니저 -> "ESP32" 검색 -> 설치



Wemos D1 R32 설치

■ 아두이노 Wemos D1 R32

- 툴 -> 보드 -> "ESP32 Dev Module"



Wemos D1 R32 예제 1

■ 아두이노 Wemos D1 R32

- 포트설정은 두개 중 뒤에 것으로!!
- RTOS



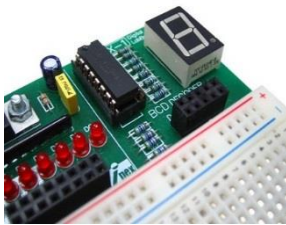
```
esp32_ex1 | 아두이노 1.8.13  
esp32_ex1  
void setup() {  
    // put your setup code here, to run once:  
    Serial.begin(9600);  
    pinMode(2, OUTPUT);  
    vTaskDelay(1000/portTICK_PERIOD_MS);  
    xTaskCreate(task1, "task1", 2048, NULL, 1, NULL);  
    xTaskCreate(task2, "task2", 2048, NULL, 1, NULL);  
}  
void loop() {  
    // put your main code here, to run repeatedly:  
    vTaskDelay(1000/portTICK_PERIOD_MS);  
}  
void task1(void *parameter)  
{  
    while(1){  
        Serial.println("Hello World");  
        vTaskDelay(1000/portTICK_PERIOD_MS);  
    }  
}  
void task2(void * parameter)  
{  
    while(1){  
        digitalWrite(2, HIGH);  
        vTaskDelay(500/portTICK_PERIOD_MS);  
        digitalWrite(2, LOW);  
        vTaskDelay(500/portTICK_PERIOD_MS);  
    }  
}  
}
```

Leaving...
Hard resetting via RTS pin...

1 ESP32 Dev Module on /dev/cu.wchusbserial14430

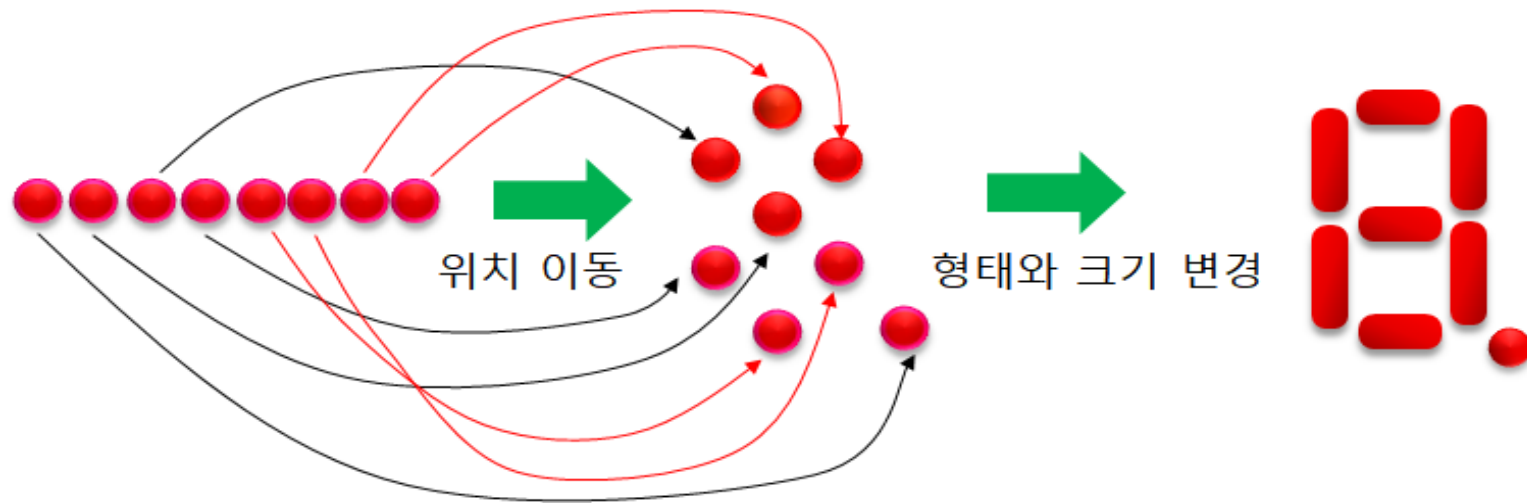
Wemos D1 R32 예제 1-digit Segment

- FND : Flexible Numeric Display(가변 숫자 표시기)
 - 보통 7-segment LED라고도 함



Wemos D1 R32 예제 1-digit Segment

■ FND와 LED는 형제

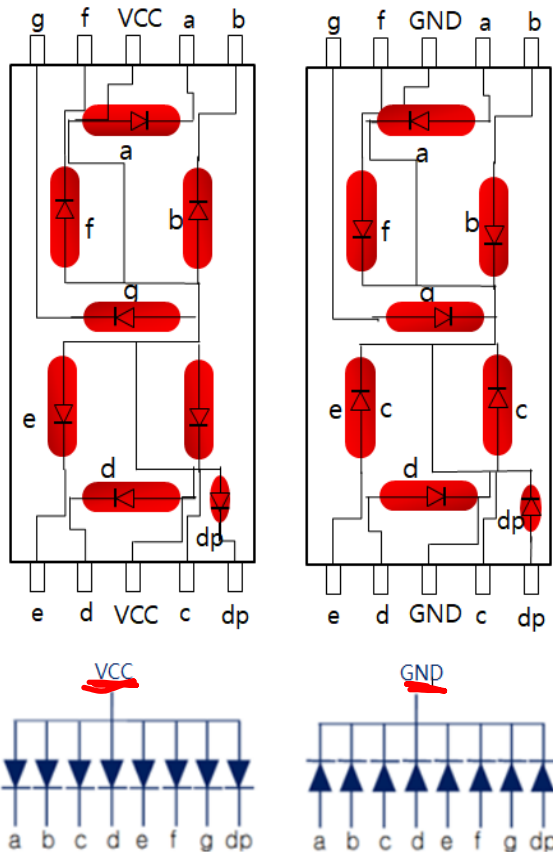


- 1FND = 8 LEDs

Wemos D1 R32 예제 1-digit Segment

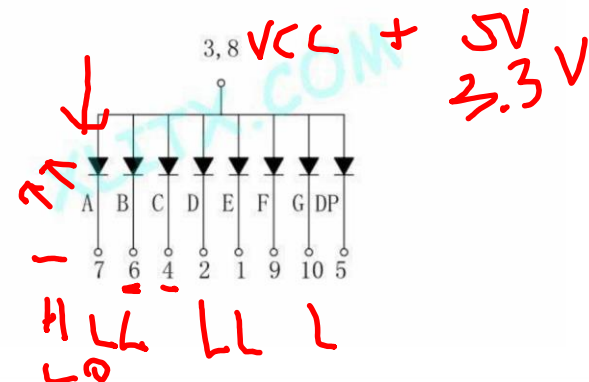
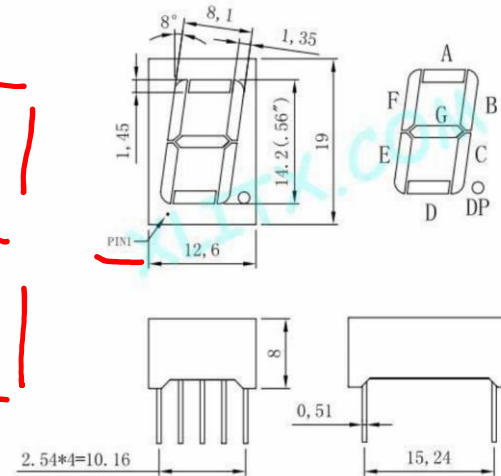
XLITX® - 5611BH

■ FND 내부



(a) 공통 애노드 타입 FND

(b) 공통 캐소드 타입 FND



Wemos D1 R32 예제 1-digit Segment

■ FND와 LED는 형제

```
ledcAttachPin(pwm_ledPin,pwm_channel);
```

```
ledcSetup(pwm_channel, 100, 8);
```

```
ledcWrite(pwm_channel, pwm_value);
```

```
1digitSegment | 아두이노 1.8.13
1digitSegment
int segmentLEDs[] = {26,25,17,16,27,14,12,13};
int digitForNum[10][8] = {
  {0, 0, 0, 0, 0, 0, 1, 1}, //0
  {1, 0, 0, 1, 1, 1, 1, 1}, //1
  {0, 0, 1, 0, 0, 1, 0, 1}, //2
  {0, 0, 0, 0, 1, 1, 0, 1}, //3
  {1, 0, 0, 1, 1, 0, 0, 1}, //4
  {0, 1, 0, 0, 1, 0, 0, 1}, //5
  {0, 1, 0, 0, 0, 0, 0, 1}, //6
  {0, 0, 0, 1, 1, 1, 1, 1}, //7
  {0, 0, 0, 0, 0, 0, 0, 1}, //8
  {0, 0, 0, 0, 1, 0, 0, 1} //9
};
void setup() {
  // put your setup code here, to run once:
  for(int i = 0 ; i < 8 ; i++) {
    pinMode(segmentLEDs[i], OUTPUT);
  }
  ledcAttachPin(5,0);
  ledcSetup(0,100,8);
}

void loop() {
  ledcWrite(0, 50);
  // put your main code here, to run repeatedly:
  for (int i = 0 ; i < 10 ; i++) {
    for (int j = 0 ; j < 8 ; j++) {
      digitalWrite(segmentLEDs[j], digitForNum[i][j]);
    }
    delay(1000);
  }
}
```

0 → 1

업로드 완료.

Leaving...

Hard resetting via RTS pin...

Wemos D1 R32 예제 4-digit Segment

4digit Segment



7H 소스

33m

10>4 Pin

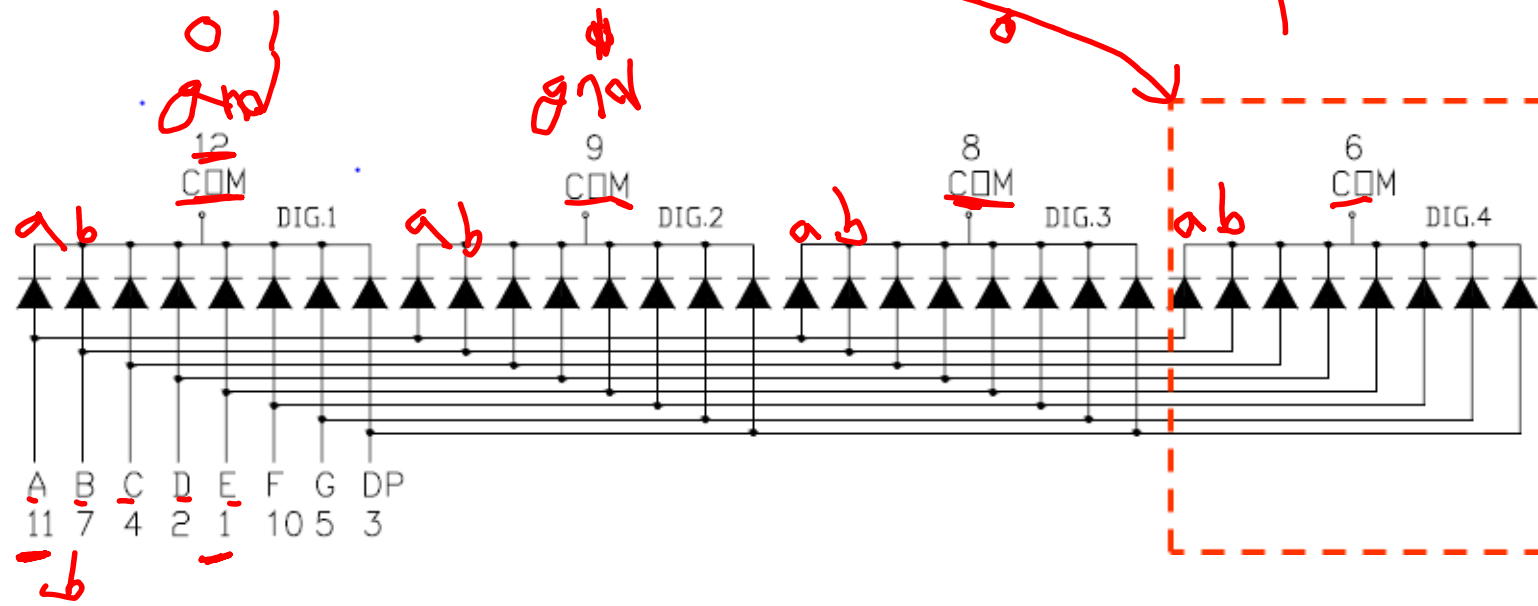
x 4

40 3.3

- 24
8

+ 4

12



Wemos D1 R32 예제 4-digit Segment

■ 4digit Segment

- 4-digit FND의 특성상 특정 순간에 4개의 다른 숫자를 동시에 디스플레이하는 것은 불가능(8개 데이터라인을 공유하기 때문)
- 이를 해결하기 위하여 눈의 '잔상효과'를 이용하는 방법을 사용하는데, '잔상효과'는 특정한 정지영상 여러 개를 순차적으로 빠르게 움직였을 때, 하나의 움직이는 영상으로 보이는 효과로, 최대 1/30초(약 33ms) 이내로 실행하여야 효과가 있음



(a) 4개의 FND에 1개씩 돌아가면서
빠르게 디스플레이하기

(b) (a)의 결과가 실제로 눈에 보이는 형태

Thank you!!

