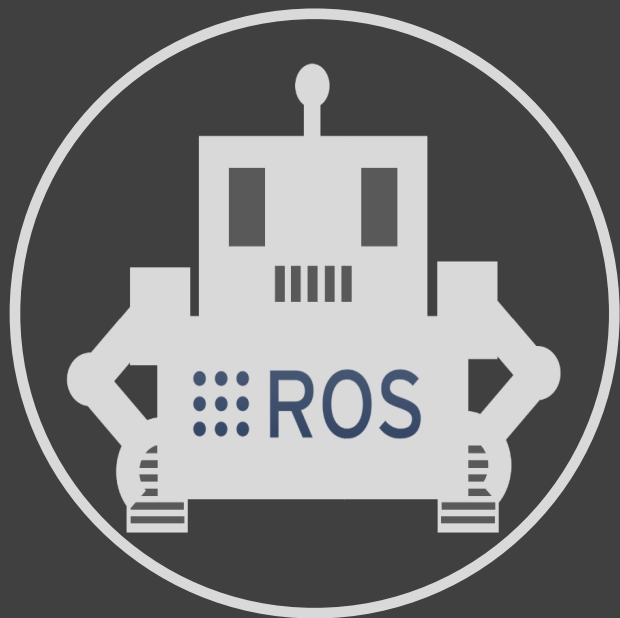



강사 소개



구선생 로보틱스 박형묵



구선생 로보틱스
@PigeonSenseiRobotics 구독자 2.34천명 동영상 103개
로봇공학 채널 구선생 로보틱스입니다. >

채널 맞춤설정 동영상 관리

홈 동영상 재생목록 커뮤니티 채널 정보

동영상 ▶ 모두 재생

ROS 기초 강의
Subscriber Node 생성
6:03
[ROS 기초 강의] Chapter4-2. Subscriber Node 생성
조회수 129회 · 13일 전

ROS 기초 강의
Publisher Node 생성
5:18
[ROS 기초 강의] Chapter4-1. Publisher Node 생성
조회수 237회 · 1개월 전

ROS 기초 강의
ROS Topic 기초
1:49
[ROS 기초 강의] Chapter4. ROS Topic 기초
조회수 228회 · 1개월 전

ROS 기초 강의
ROS 환경 구조
5:00
[ROS 기초 강의] Chapter3. ROS 환경 구조
조회수 324회 · 1개월 전

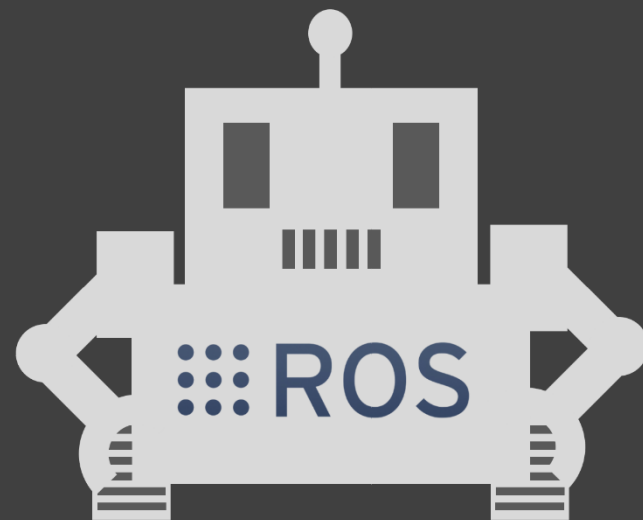
ROS 기초 강의
ROS Node 생성
7:37
[ROS 기초 강의] Chapter2-2. ROS Node 생성
조회수 636회 · 2개월 전

ROS 기초 강의
ROS Node 기초
2:25
[ROS 기초 강의] Chapter2-1. ROS Node 기초
조회수 921회 · 3개월 전

라즈베리 파이 기초

Chapter 1. LED 제어

구선생 로보틱스



라즈베리 파이란 무엇인가?

라즈베리 파이란 무엇인가?



- 저렴한 가격과 작은 크기의 싱글 보드 컴퓨터
- 리눅스 기반 운영체제
- ARM CPU

라즈베리 파이에 원격접속 하기

원격 접속 명령어

```
ssh msrose@192.168.10.10
```

```
Linux raspberrypi 6.1.21-v8+ #1642 SMP PREEMPT Mon Apr  3 17:24:16 BST 2023 aarch64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Fri May 12 04:03:16 2023 from 192.168.170.89
msrose@raspberrypi:~ $
```

원격 접속 후 터미널의 사용자가 msrose로 바뀌었다

Linux 기초 명령어 배우보기

간단한 Python 소스코드 작성

Python 소스코드 생성

```
nano my_first_python.py
```

아래 내용 작성 후 저장

```
1 print("Hello World!!")
2
```

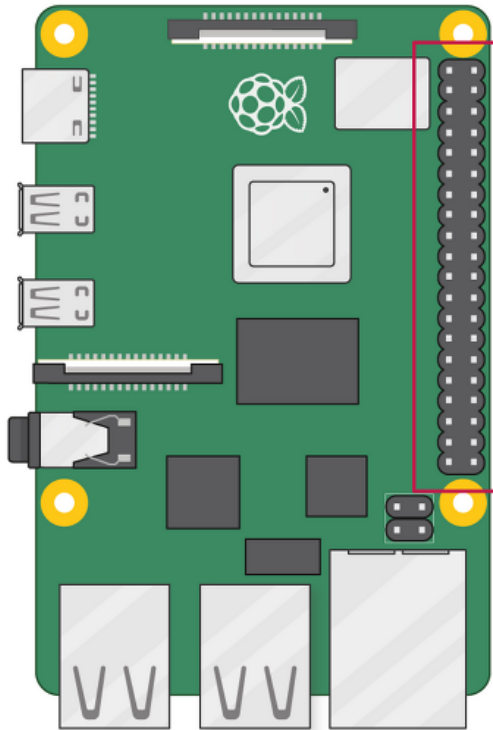
실행

```
python3 my_first_python.py
```

```
msrose@raspberrypi:~ $ python3 my_first_python.py
Hello World!!
```

LED 제어 기초

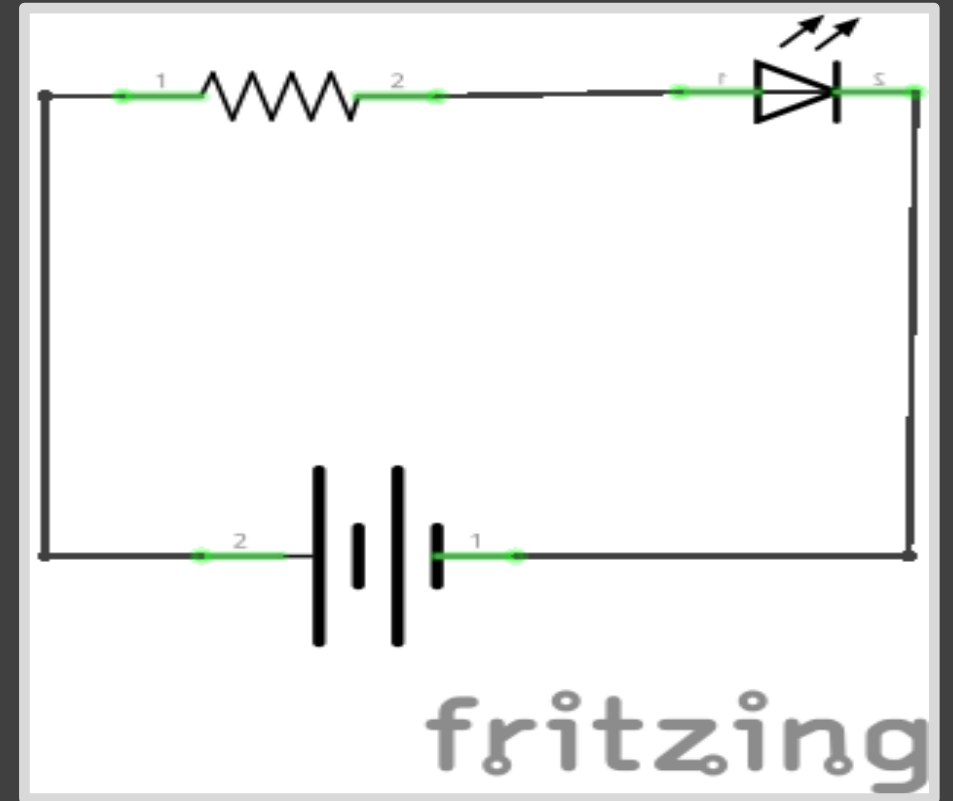
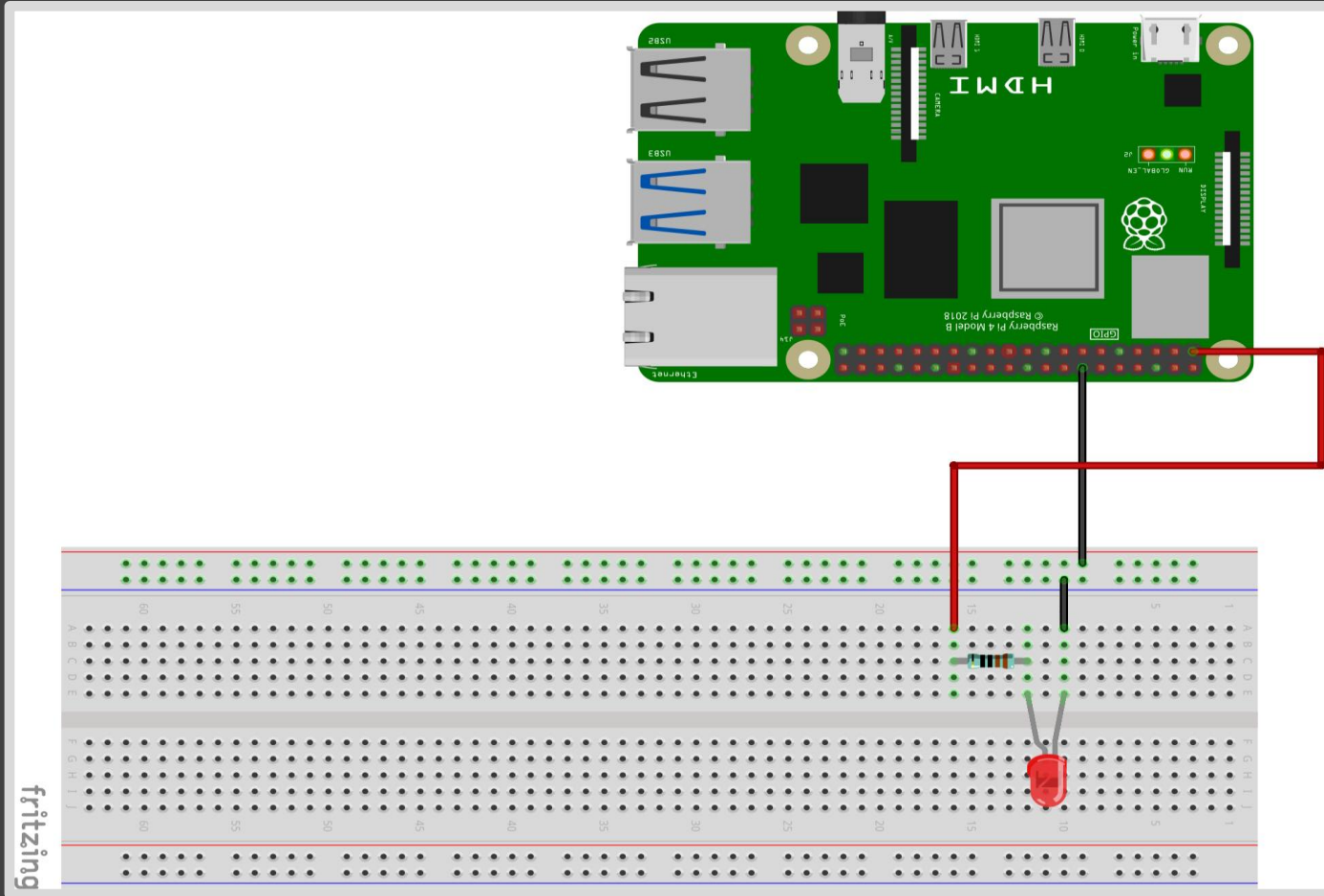
라즈베리 파이 Pin Map



3V3 power	1	2	5V power
GPIO 2 (SDA)	3	4	5V power
GPIO 3 (SCL)	5	6	Ground
GPIO 4 (GPGCLK0)	7	8	GPIO 14 (TXD)
Ground	9	10	GPIO 15 (RXD)
GPIO 17	11	12	GPIO 18 (PCM_CLK)
GPIO 27	13	14	Ground
GPIO 22	15	16	GPIO 23
3V3 power	17	18	GPIO 24
GPIO 10 (MOSI)	19	20	Ground
GPIO 9 (MISO)	21	22	GPIO 25
GPIO 11 (SCLK)	23	24	GPIO 8 (CE0)
Ground	25	26	GPIO 7 (CE1)
GPIO 0 (ID_SD)	27	28	GPIO 1 (ID_SC)
GPIO 5	29	30	Ground
GPIO 6	31	32	GPIO 12 (PWM0)
GPIO 13 (PWM1)	33	34	Ground
GPIO 19 (PCM_FS)	35	36	GPIO 16
GPIO 26	37	38	GPIO 20 (PCM_DIN)
Ground	39	40	GPIO 21 (PCM_DOUT)

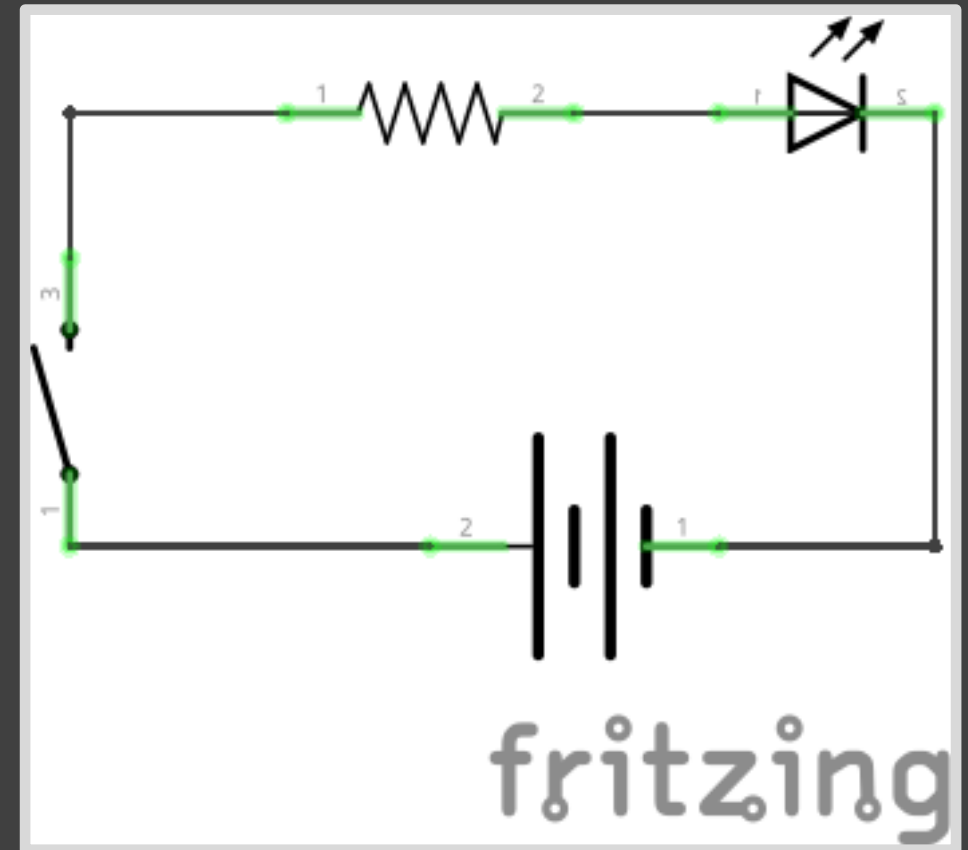
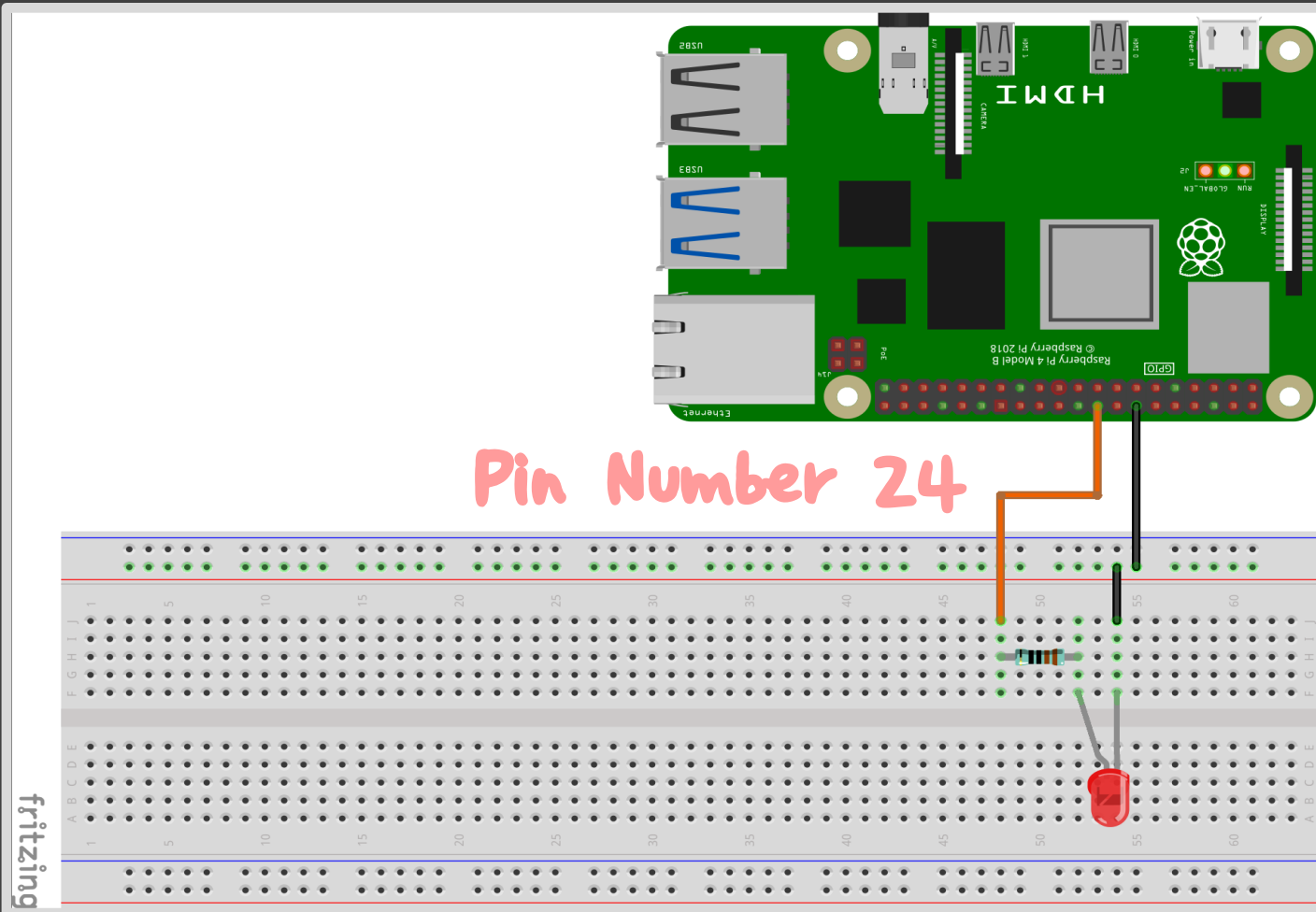
LED 제어 기초

간단한 LED 회로도 구성하기



LED 제어 기초

LED 제어 원리 알기



LED 제어 기초

LED 제어 원리 알기

아래 경로의 파일을 이용하여 Gpio를 제어

```
msrose@raspberrypi: /sys/class/gpio $ ls  
export  gpiochip0  gpiochip504  unexport
```

LED 메모리 할당	echo 24> /sys/class/gpio/export
LED 모드 설정	echo out > /sys/class/gpio/gpio24/direction
LED 켜기	echo 1 > /sys/class/gpio/gpio24/value
LED 메모리 할당 해제	echo 24 > /sys/class/gpio/unexport

LED 제어 기초

Python Code를 이용하여 LED 제어하기

```
1 import RPi.GPIO as GPIO
2 import time
3
4 GPIO.setmode(GPIO.BCM) # GPIO 모드 초기화
5 GPIO.setup(24,GPIO.OUT) # 24번 핀 OUT 모드 설정 / 메모리 할당
6
7 GPIO.output(24,GPIO.HIGH) # 24번 핀 1
8 time.sleep(1.0) # 1초 대기
9 GPIO.output(24,GPIO.LOW) # 24번 핀 0
10
11 GPIO.cleanup() # 모든 GPIO 메모리 할당 해제
```

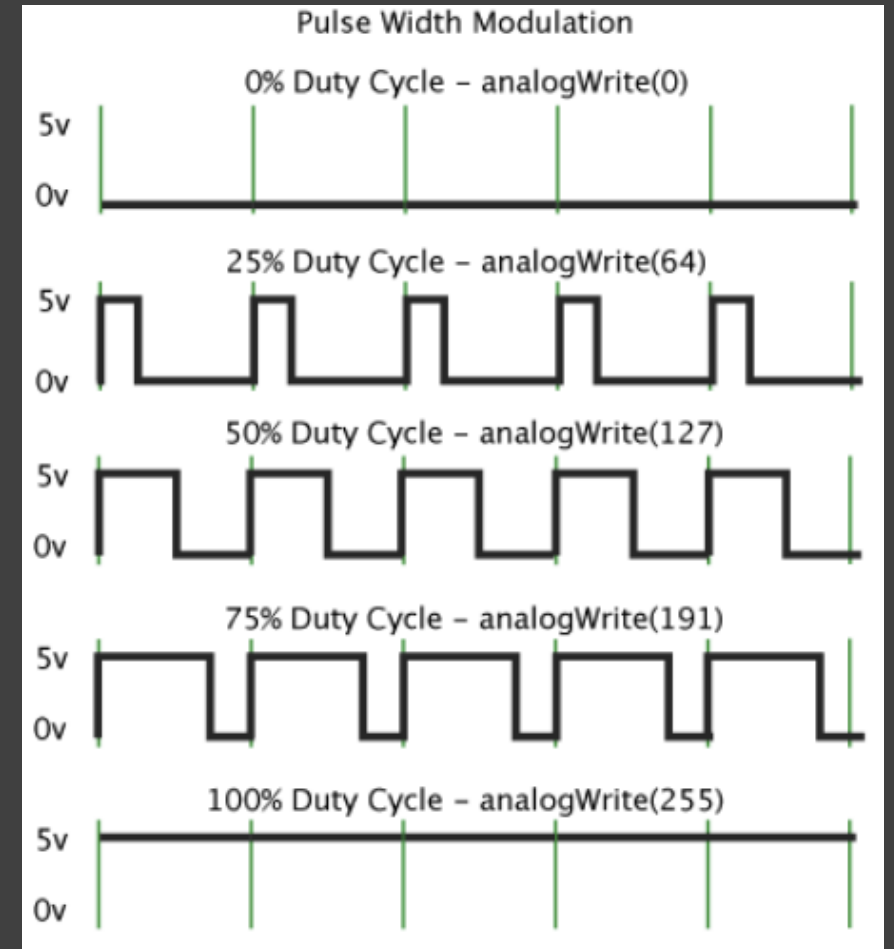
LED 제어 기초

Python Code를 이용하여 LED PWM 제어하기

PWM 제어란?

디지털 신호는 0과 1만이 존재한다. 0~255의 아날로그 신호를 출력하고자 하는 경우 0과 1의 디지털 신호로 제어하기엔 무리가 있다. 이를 위해선 PWM 제어가 필요하다.

HIGH 와 LOW신호를 일정 Hz 만큼 반복하여 아날로그 신호 처럼 제어할 수 있다.



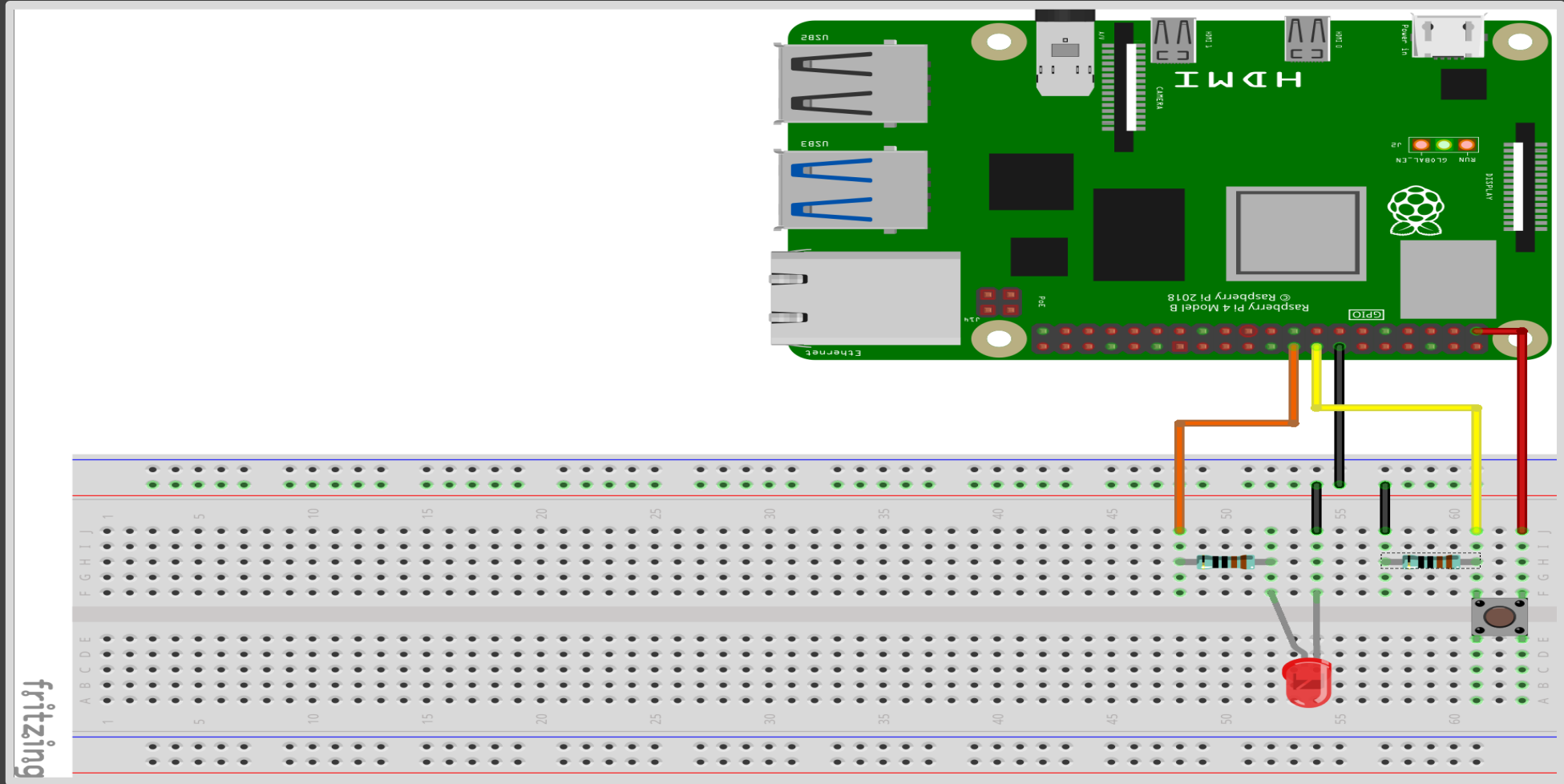
LED 제어 기초

Python Code를 이용하여 LED PWM 제어하기

```
1 import RPi.GPIO as GPIO
2 import time
3
4 GPIO.setmode(GPIO.BCM) # GPIO 모드 초기화
5 GPIO.setup(24, GPIO.OUT) # 24번 핀 OUT 모드 설정 / 메모리 할당
6
7 ledWhitePwm = GPIO.PWM(24, 500) # 24번 핀 PWM 500Hz 설정
8 ledWhitePwm.start(0) # 24번 핀 PWM LOW
9
10 ledWhitePwm.ChangeDutyCycle(0) # 24번 핀 PWM 시작 0% Duty 설정
11 time.sleep(1.0) # 1초 대기
12 ledWhitePwm.ChangeDutyCycle(30) # 24번 핀 PWM 시작 30% Duty 설정
13 time.sleep(1.0) # 1초 대기
14 ledWhitePwm.ChangeDutyCycle(60) # 24번 핀 PWM 시작 60% Duty 설정
15 time.sleep(1.0) # 1초 대기
16 ledWhitePwm.ChangeDutyCycle(100) # 24번 핀 PWM 시작 100% Duty 설정
17
18 GPIO.cleanup() # 모든 GPIO 메모리 할당 해제
```

LED 제어 기초

Python Code를 이용하여 스위치 제어하기



LED 제어 기초

Python Code를 이용하여 스위치 제어하기

```
1 import RPi.GPIO as GPIO
2 import time
3
4 GPIO.setmode(GPIO.BCM) # GPIO 모드 초기화
5 GPIO.setup(24, GPIO.OUT) # 24번 핀 OUT 모드 설정 / 메모리 할당
6 GPIO.setup(23, GPIO.IN, pull_up_down=GPIO.PUD_DOWN) # 23번 핀 IN 모드 설정 / 메모리 할당
7
8 try:
9     while True:
10         switch_state = GPIO.input(23) # 23번 스위치 값 읽기
11         if switch_state:
12             GPIO.output(24, GPIO.HIGH) # 24번 핀 1
13         else:
14             GPIO.output(24, GPIO.LOW) # 24번 핀 0
15
16 # Ctrl+C 입력시 프로그램 종료
17 except KeyboardInterrupt:
18     pass
19
20 GPIO.cleanup()
```

LED 제어 작품 제작하기

작품 01~04 선택하여 제작하기

작품 01) 55P~

작품 02) 63P~

작품 03) 67P~

작품 04) 72P~

감사합니다

구선생 로보틱스

