라즈베리파이 실습
- 사운드 센서

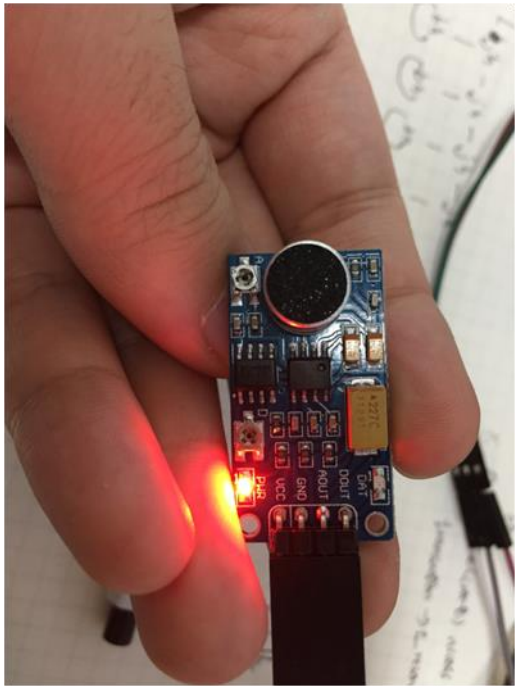
GPIO pins

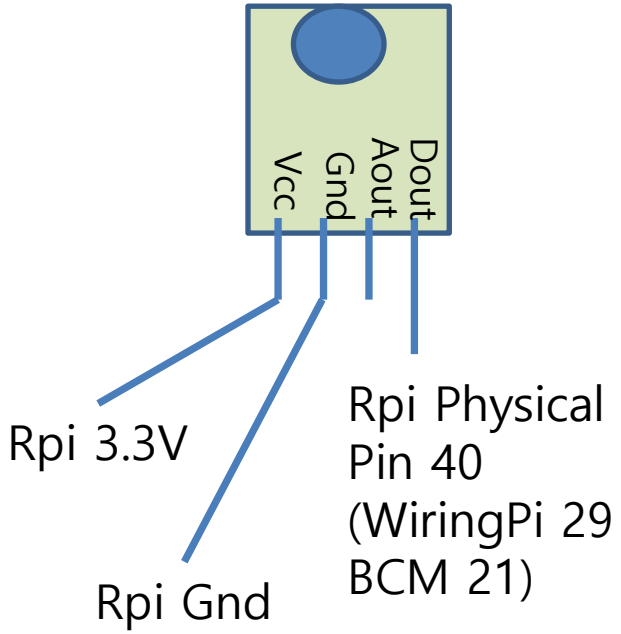
Raspberry Pi GPIO Header

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

Sound Sensor

- **소리가 감지될 경우 소리의 세기만큼 LED가 밝게 켜짐**
  - 사용 핀: VCC | GND | Analog output | Digital output





SoundSensor in python

```
import RPi.GPIO as gpio #Use the GPIO module of RPi
import time #Use time module

gpio.setmode(gpio.BCM) #Works with gpio in BCM(Broadcom chip-specific pin numbers mode)

sound = 21                  # Physical 40

print("start sound sensor")

gpio.setup(sound, gpio.IN) #Set sound digital output pin to IN
try: #for Exception
        while True:
                if gpio.input(sound) == 1:
                        print("Sound detected!")
                else:
                        print("No sound detected!")
                time.sleep(0.5)
except:
        gpio.cleanup()
```

SoundSensor in C

```
/* sound.c */

#include <stdio.h>
#include <wiringPi.h>
#define SOUND 29      // BCM 21  // physical pin 40

int main(void) {
        if (wiringPiSetup() == -1) {
                return 1;
        }
        pinMode(SOUND, INPUT);
        while (1) {
                if (digitalRead(SOUND) == 1)
                        printf("Sound detected₩n");
                else
                        printf("No sound detected₩n");
                delay(500);
        }
        return 0;
}
```

라즈베리파이 실습
- infra red

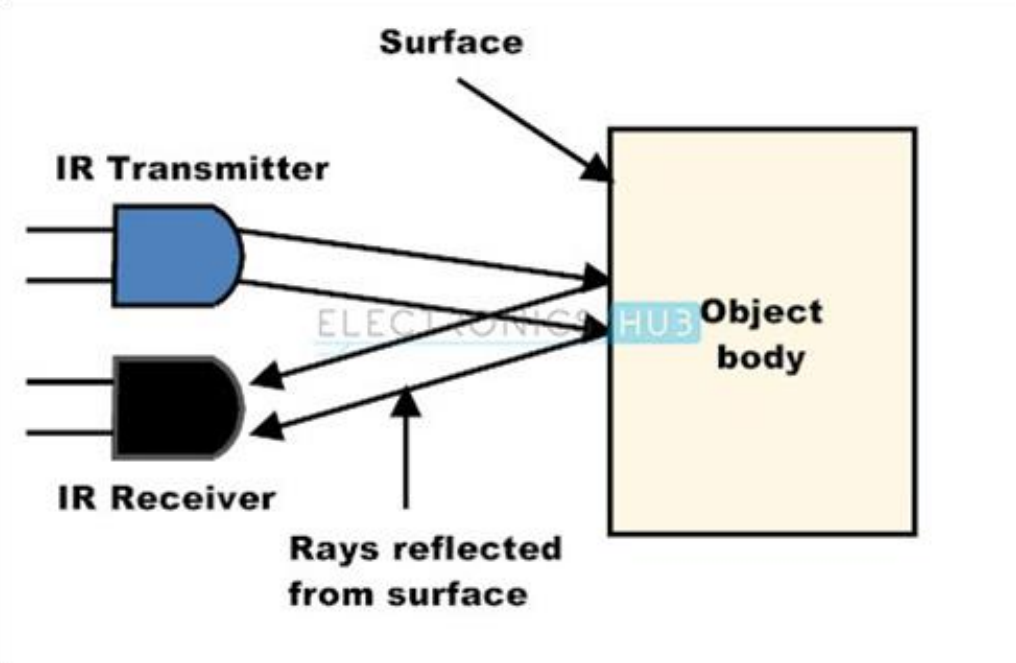
GPIO pins

Raspberry Pi GPIO Header

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

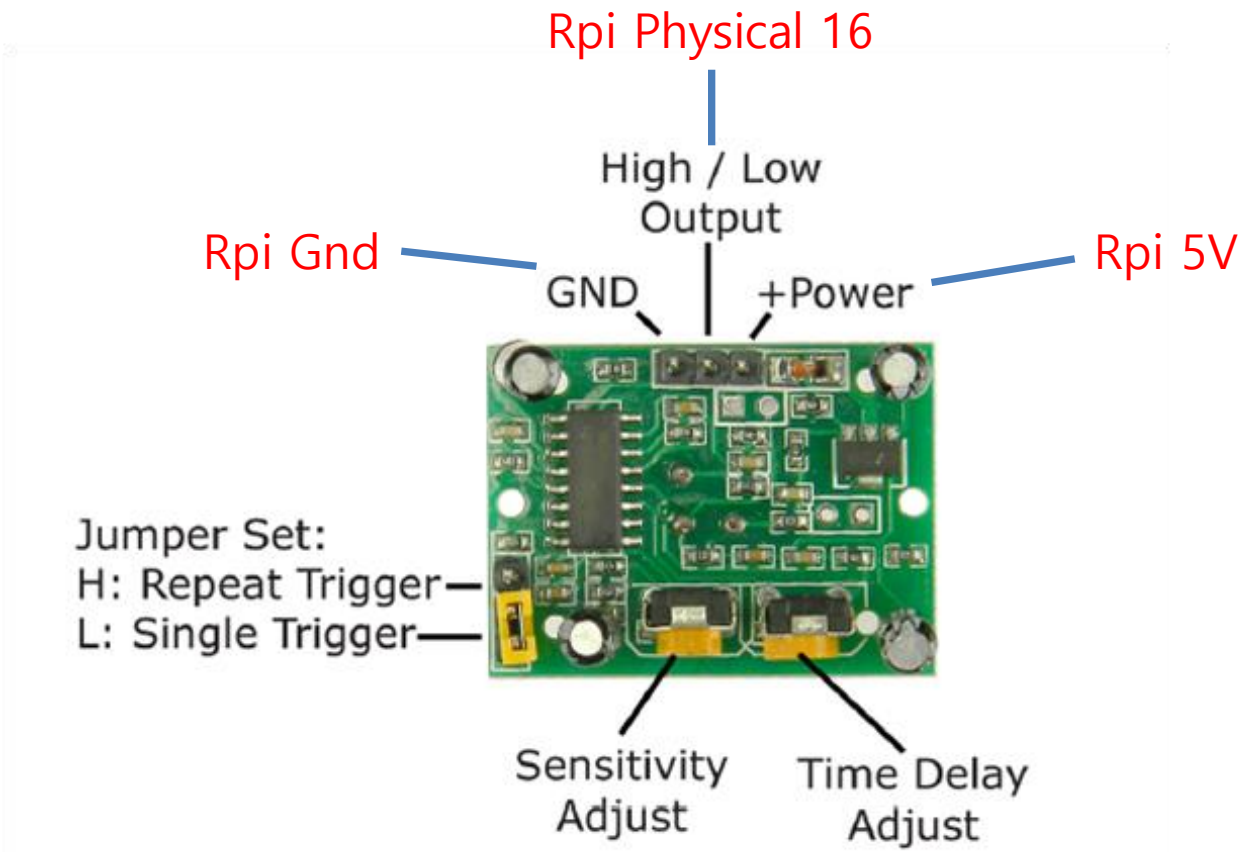
IR Sensor

- 적외선 송신기에서 적외선 신호를 전송
- 물체에 반사된 적외선 신호를 감지하여 가까이 있는 물체를 감지



IR sensor

- 체온(36.5도)와 움직임이 감지되면 신호를 보냄

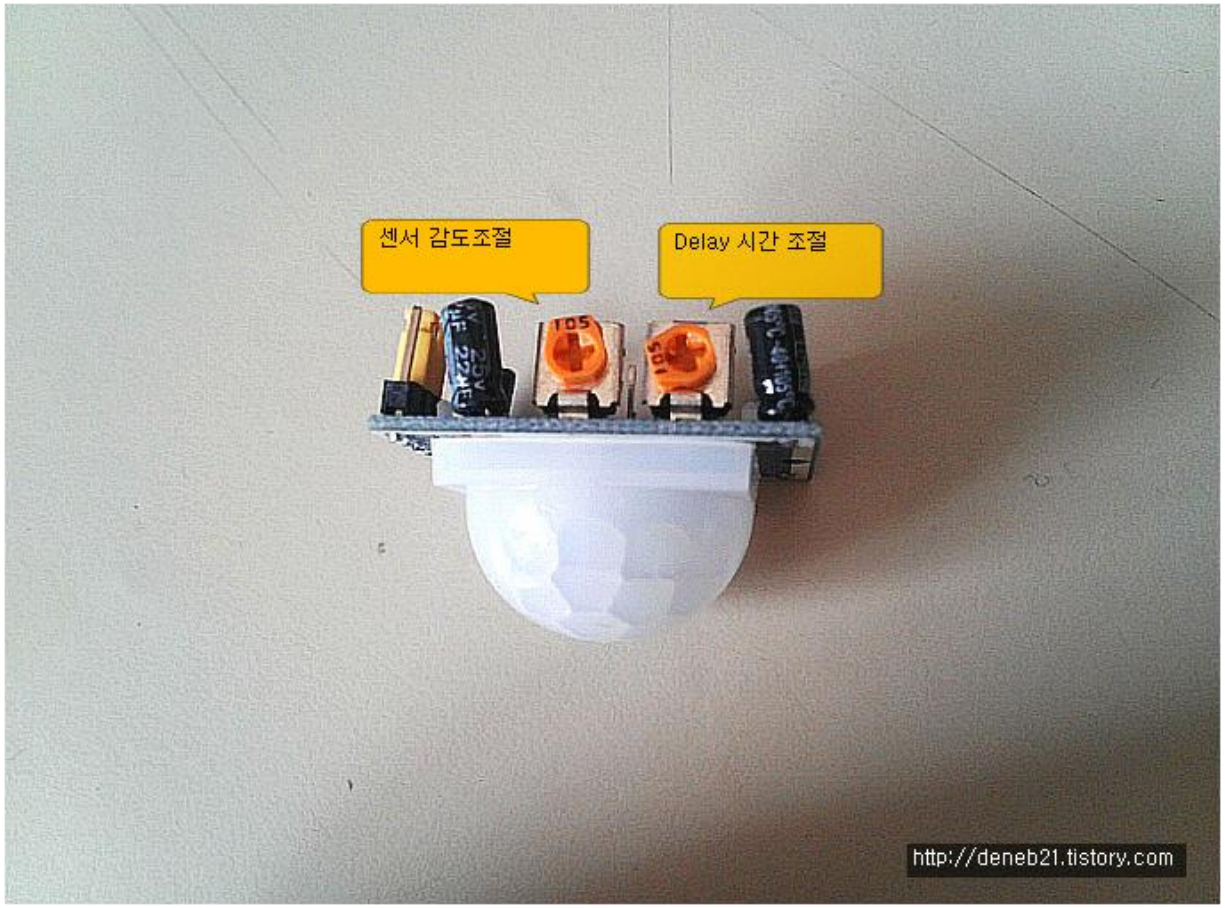


IR sensor

- 체온(36.5도)와 움직임이 감지되면 신호를 보냄

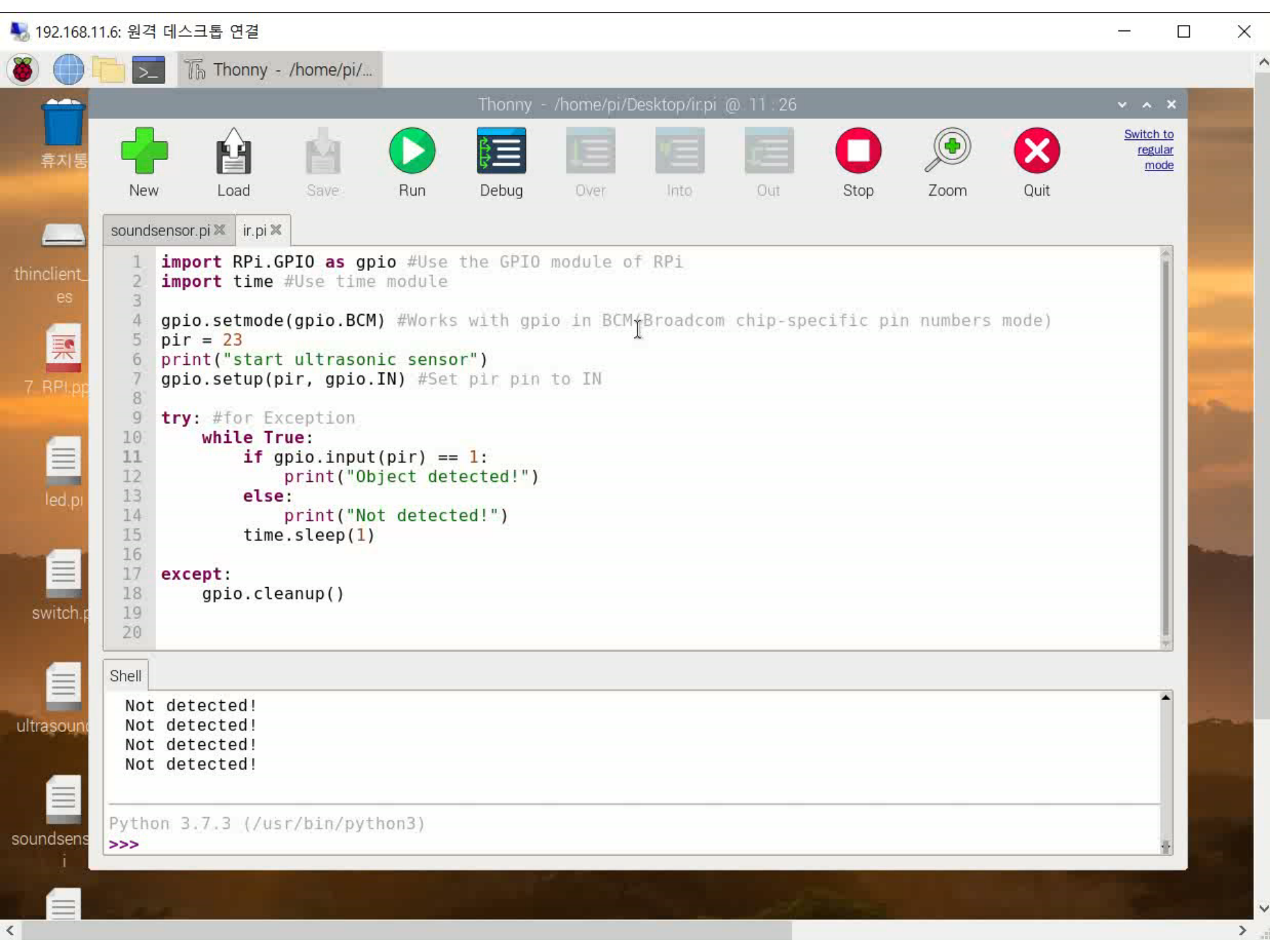
IR sensor

- 적외선 센서의 감도, delay 시간 조절 가능



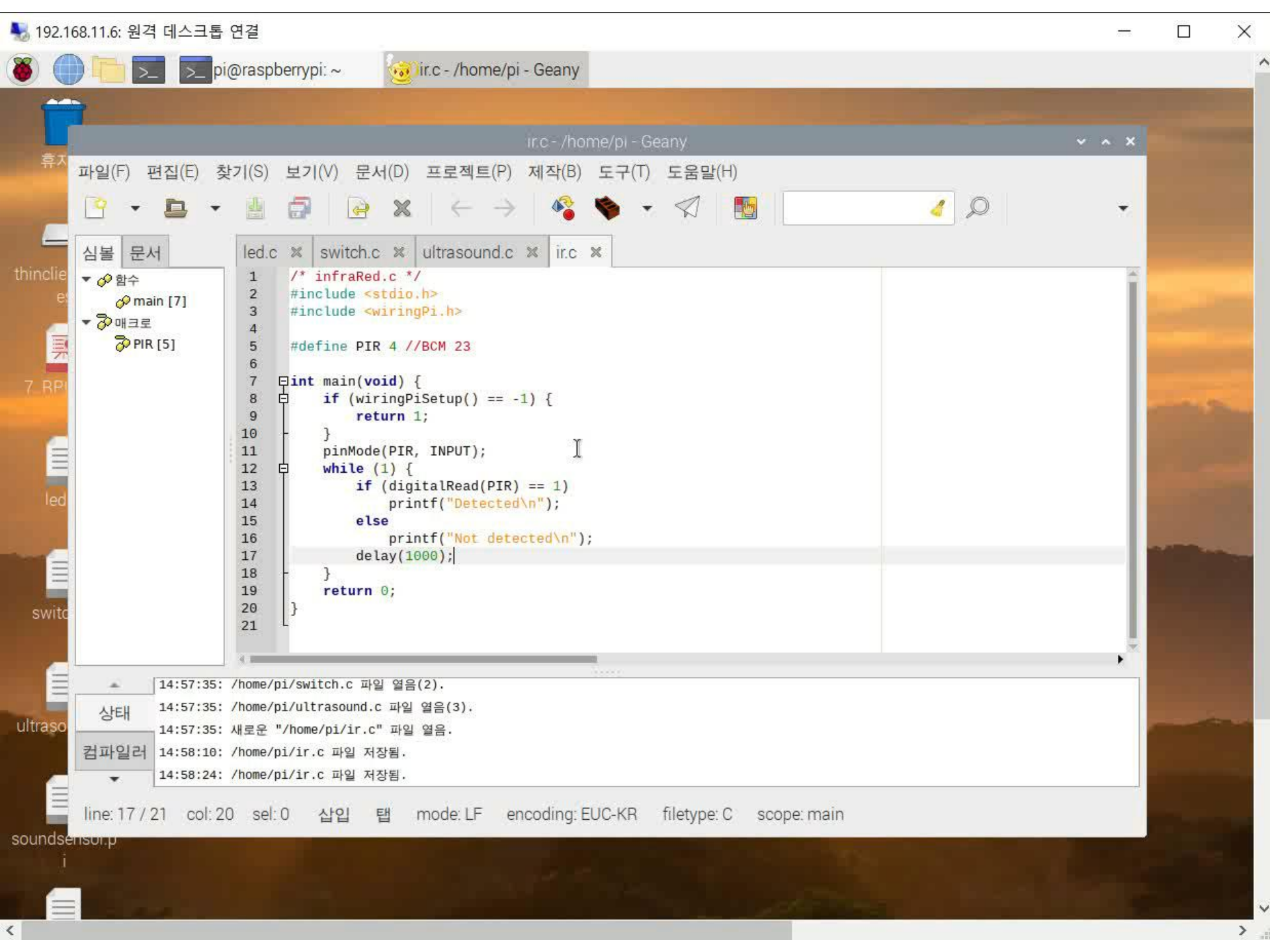
IR in python

```
import RPi.GPIO as gpio #Use the GPIO module of RPi
import time #Use time module

gpio.setmode(gpio.BCM) #Works with gpio in BCM(Broadcom chip-specific pin numbers mode)
pir = 23
print("start ultrasonic sensor")
gpio.setup(pir, gpio.IN) #Set pir pin to IN

try: #for Exception
        while True:
                if gpio.input(pir) == 1:
                        print("Object detected!")
                else:
                        print("Not detected!")

                time.sleep(1)

except:
        gpio.cleanup()
```


192.168.11.6: 원격 데스크톱 연결
Thonny - /home/pi/...
Thonny - /home/pi/Desktop/ir.pi @ 11 : 26
New
Load
Save
Run
Debug
Over
Into
Out
Stop
Zoom
Quit
Switch to regular mode
soundsensor.pi
ir.pi
1 import RPi.GPIO as gpio #Use the GPIO module of RPi
2 import time #Use time module
3
4 gpio.setmode(gpio.BCM) #Works with gpio in BCM(Broadcom chip-specific pin numbers mode)
5 pir = 23
6 print("start ultrasonic sensor")
7 gpio.setup(pir, gpio.IN) #Set pir pin to IN
8
9 try: #for Exception
10 while True:
11 if gpio.input(pir) == 1:
12 print("Object detected!")
13 else:
14 print("Not detected!")
15 time.sleep(1)
16
17 except:
18 gpio.cleanup()
19
20
Shell
Not detected!
Not detected!
Not detected!
Not detected!
Python 3.7.3 (/usr/bin/python3)
>>>

IR in C

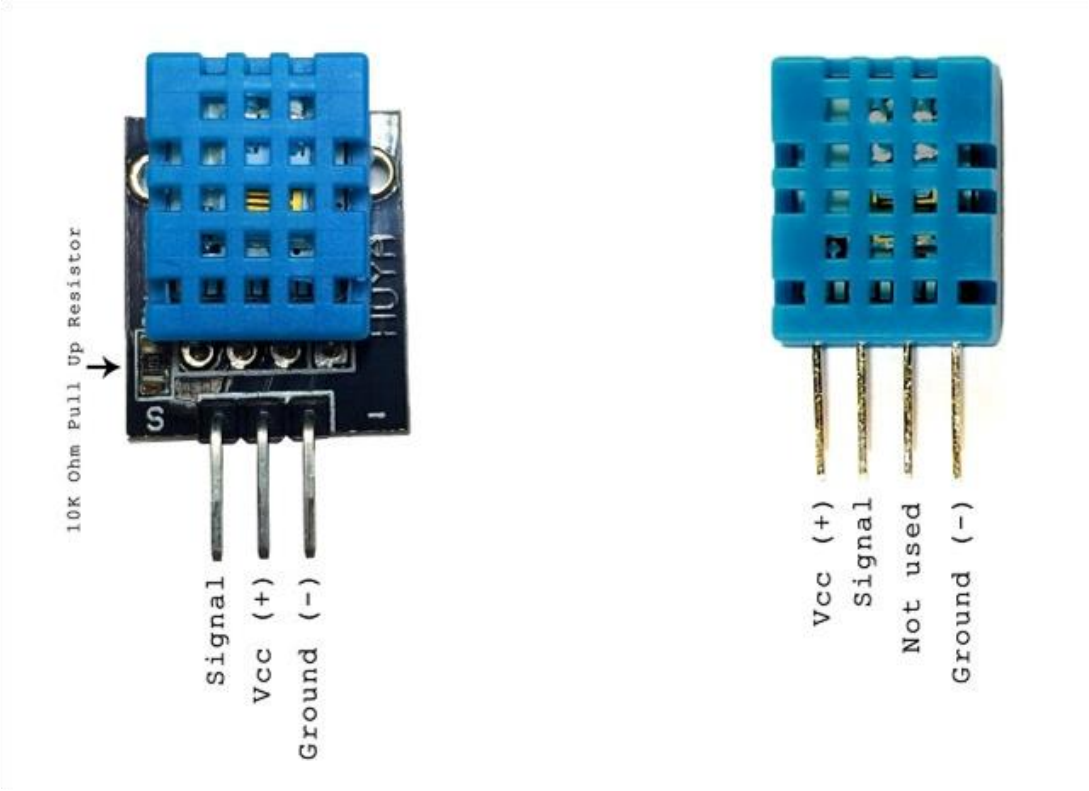
```
/* infraRed.c */
#include <stdio.h>
#include <wiringPi.h>

#define PIR 4 //BCM 23

int main(void) {
	if (wiringPiSetup() == -1) {
		return 1;
	}
	pinMode(PIR, INPUT);
	while (1) {
		if (digitalRead(PIR) == 1)
			printf("Detected₩n");
		else
			printf("Not detected₩n");
		delay(1000);
	}
	return 0;
}
```


192.168.11.6: 원격 데스크톱 연결
pi@raspberrypi: ~
ir.c - /home/pi - Geany
ir.c - /home/pi - Geany
파일(F) 편집(E) 찾기(S) 보기(V) 문서(D) 프로젝트(P) 제작(B) 도구(T) 도움말(H)
심볼 문서
led.c switch.c ultrasound.c ir.c
/* infraRed.c */
#include <stdio.h>
#include <wiringPi.h>
#define PIR 4 //BCM 23
int main(void) {
if (wiringPiSetup() == -1) {
return 1;
}
pinMode(PIR, INPUT);
while (1) {
if (digitalRead(PIR) == 1)
printf("Detected\n");
else
printf("Not detected\n");
delay(1000);
}
return 0;
}
함수
main [7]
매크로
PIR [5]
14:57:35: /home/pi/switch.c 파일 열음(2).
14:57:35: /home/pi/ultrasound.c 파일 열음(3).
14:57:35: 새로운 "/home/pi/ir.c" 파일 열음.
14:58:10: /home/pi/ir.c 파일 저장됨.
14:58:24: /home/pi/ir.c 파일 저장됨.
상태
컴파일러
line: 17 / 21 col: 20 sel: 0 삽입 탭 mode: LF encoding: EUC-KR filetype: C scope: main

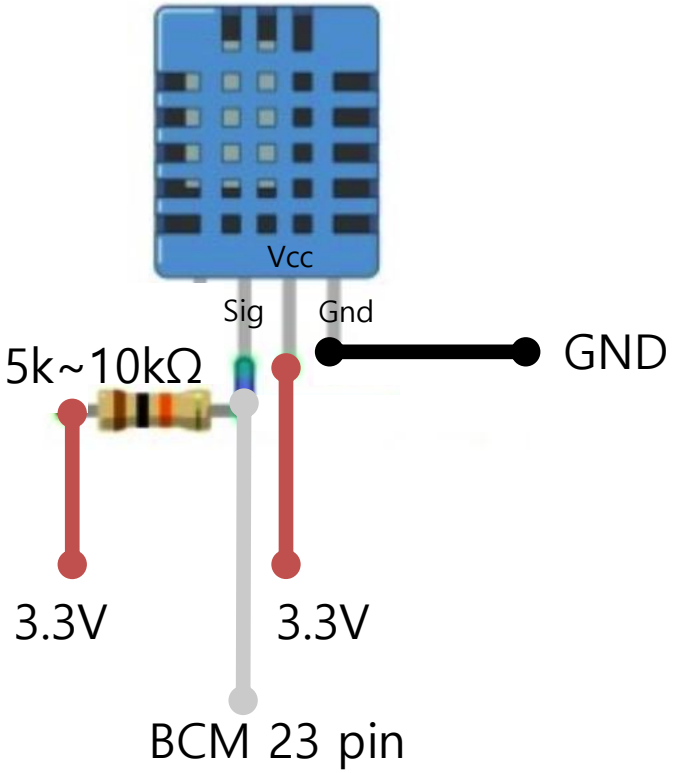
라즈베리파이 실습
- temperature & humidity sensor

dht11

- **Digital output Temperature & Humidity sensor**



dht11 pull-up 회로



dht11

Overview
-Onboard sensor DHT11, detects both temperature and humidity

Specifications
Temperature measuring range: 0℃ ~ 50℃
Temperature tolerance: ±2℃
Humidity measuring range: 20% ~ 95% (0℃ ~ 50℃)
Humidity tolerance: ±5%
Dimension: 29.0mm * 18.0mm
Mounting holes size: 2.0mm

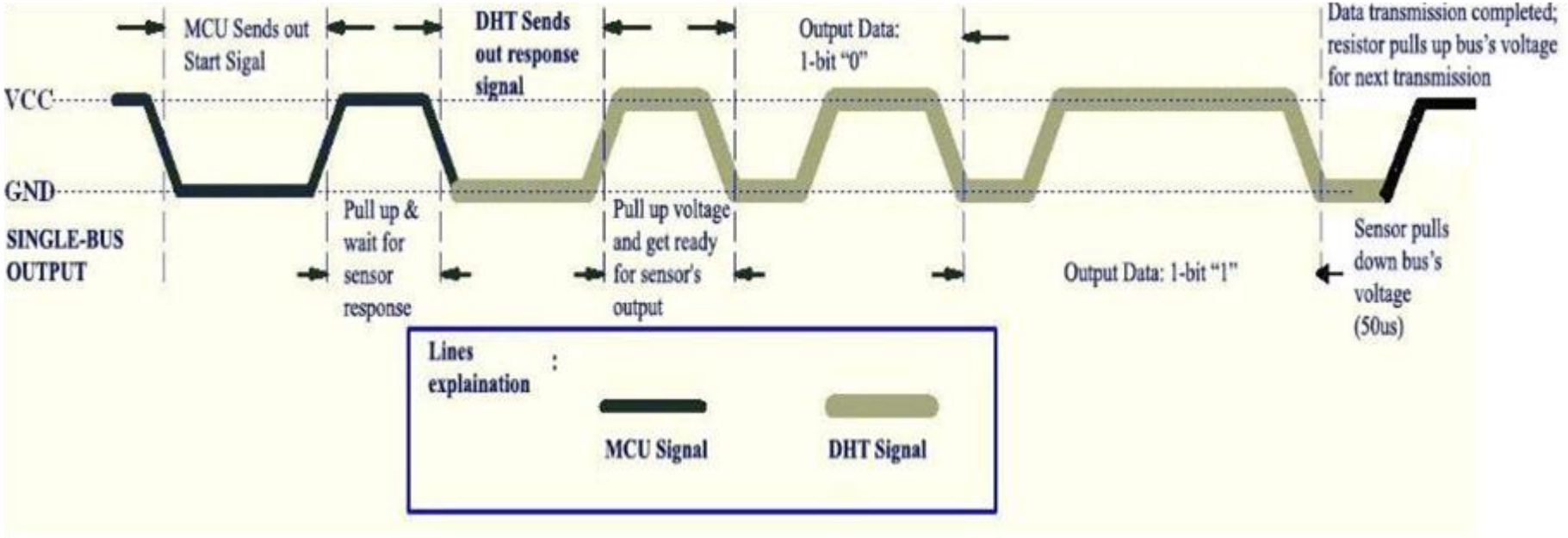
Applications
Ambient temperature and humidity detection

How to Use
In the case of working with a MCU:

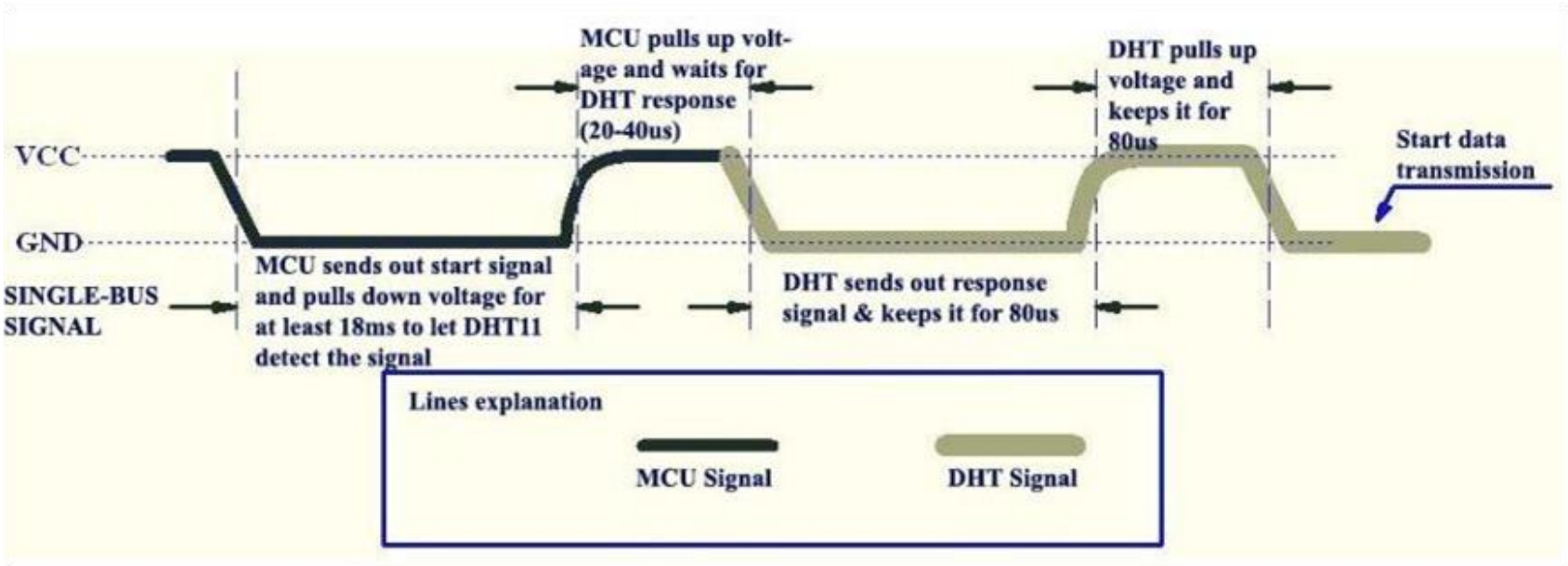
-VCC ‹ › 3.3V ~ 5.5V
-GND ‹ › power supply ground
-DOUT ‹ › MCU.IO

dht11

- Digital output(single bus)가 MCU와 DHT11 센서의 데이터 커뮤니케이션, 동기화에 사용됨
- 커뮤니케이션 프로세스는 약 4ms 동안 지속됨
- 데이터 전송의 정확도를 위해 8비트 체크섬 연산

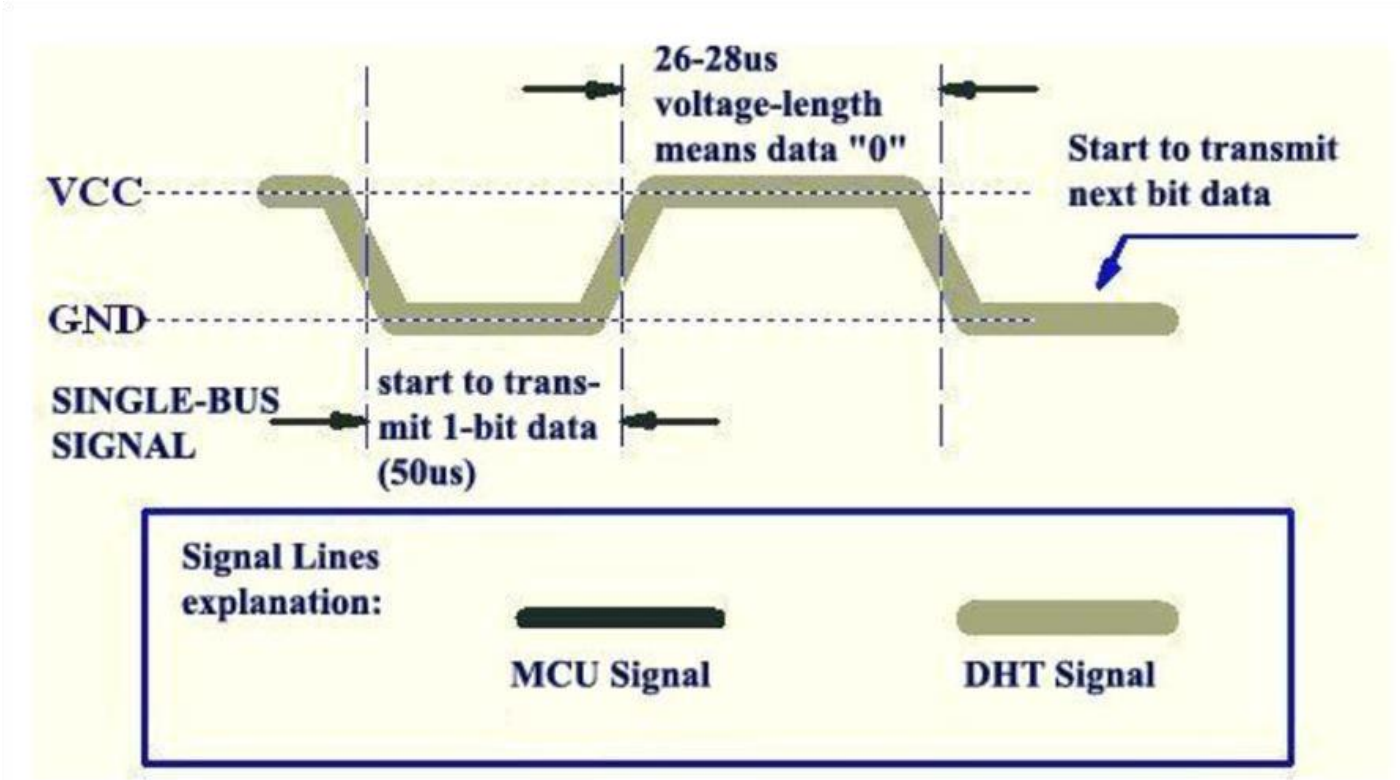


- DHT11이 MCU 신호를 감지할 경우 DHT 신호를 전송



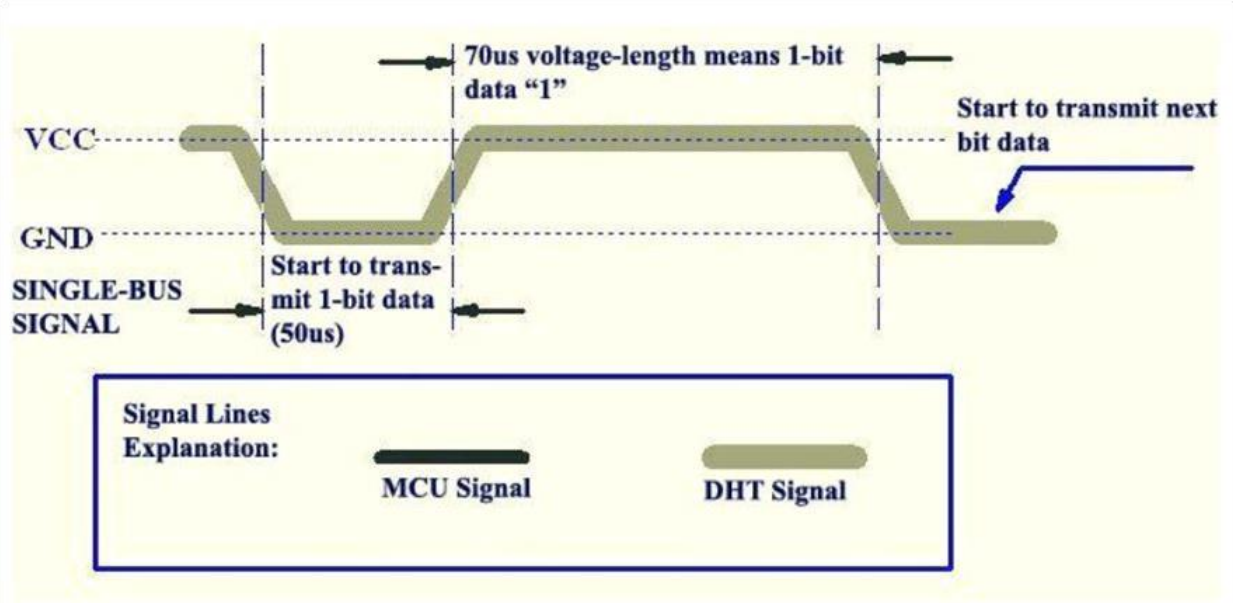
dht11

- 데이터 비트 0의 경우 약 20-28마이크로초 동안 HIGH



dht11

- 데이터 비트 1의 경우 약 70마이크로초 동안 HIGH



dht11

- **sudo apt-get update**
- **sudo apt-get install build-essential python-dev**
- **git clone https://github.com/adafruit/Adafruit_Python_DHT.git**
  - username: embeddedUOS, passwd: asdf1234
- **cd Adafruit_Python_DHT**
- **sudo python setup.py install**

dht11

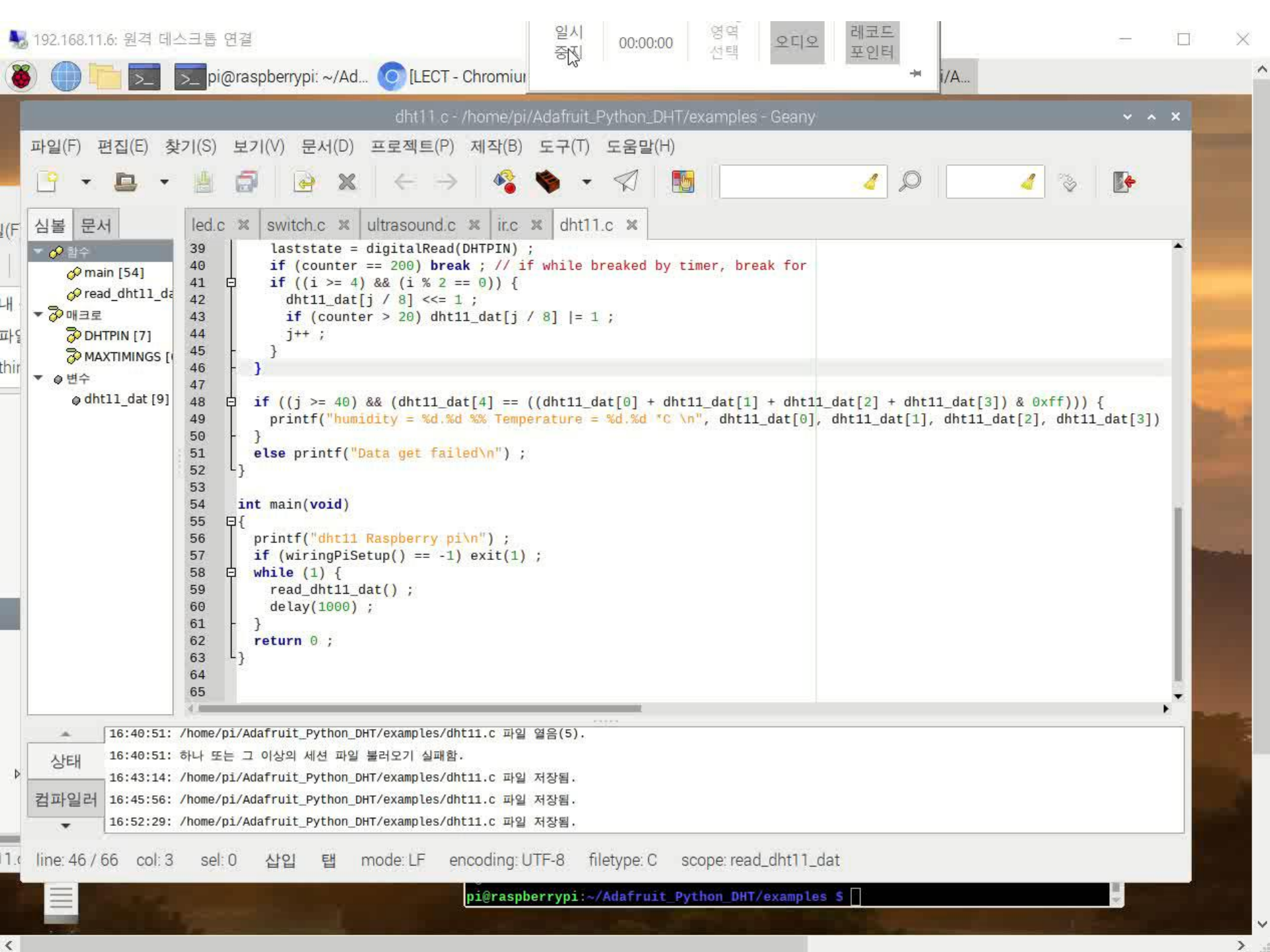
- 현재 디렉토리(Adafruit_Python_DHT) 확인
- cd examples
- sudo ./AdafruitDHT.py 11 23
  - 센서는 dht11이고 gpio 포트는 BCM

```
pi@raspberrypi00:~/Adafruit_Python_DHT $ cd examples/
pi@raspberrypi00:~/Adafruit_Python_DHT/examples $ ./AdafruitDHT.py 11 23
Temp=28.0*  Humidity=17.0%
pi@raspberrypi00:~/Adafruit_Python_DHT/examples $
```

dht11.c

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

#define MAXTIMINGS 83
#define DHTPIN 4

int dht11_dat[5] = {0, } ;

void read_dht11_dat()
{
  uint8_t laststate = HIGH ;
  uint8_t counter = 0 ;
  uint8_t j = 0, i ;
  uint8_t flag = HIGH ;
  uint8_t state = 0 ;
  float f ;


  dht11_dat[0] = dht11_dat[1] = dht11_dat[2] = dht11_dat[3] =
dht11_dat[4] = 0 ;
  pinMode(DHTPIN, OUTPUT) ;
  digitalWrite(DHTPIN, LOW) ;
  delay(18) ;

  digitalWrite(DHTPIN, HIGH) ;
  delayMicroseconds(30) ;
  pinMode(DHTPIN, INPUT) ;
```

```
  for (i = 0; i < MAXTIMINGS; i++) {
    counter = 0 ;
    while ( digitalRead(DHTPIN) == laststate) {
      counter++ ;
      delayMicroseconds(1) ;
      if (counter == 200) break ;
    }
    laststate = digitalRead(DHTPIN) ;
    if (counter == 200) break ; // if while breaked by timer, break
for
    if ((i >= 4) && (i % 2 == 0)) {
      dht11_dat[j / 8] <<= 1 ;
      if (counter > 20) dht11_dat[j / 8] |= 1 ;
      j++ ;
    }
  }

  if ((j >= 40) && (dht11_dat[4] == ((dht11_dat[0] + dht11_dat[1]
+ dht11_dat[2] + dht11_dat[3]) & 0xff))) {
    printf("humidity = %d.%d %% Temperature = %d.%d *C ₩n",
dht11_dat[0], dht11_dat[1], dht11_dat[2], dht11_dat[3]) ;
  }
  else printf("Data get failed₩n") ;
}

int main(void)
{
  printf("dht11 Raspberry pi₩n") ;
  if (wiringPiSetup() == -1) exit(1) ;
  while (1) {
    read_dht11_dat() ;
    delay(1000) ;
  }
  return 0 ;
}
```


192.168.11.6: 원격 데스크톱 연결
일시 중지
00:00:00
영역 선택
오디오
레코드 포인터
pi@raspberrypi: ~/Ad...
[LECT - Chromiu
dht11.c - /home/pi/Adafruit_Python_DHT/examples - Geany
파일(F) 편집(E) 찾기(S) 보기(V) 문서(D) 프로젝트(P) 제작(B) 도구(T) 도움말(H)
심볼 문서
led.c switch.c ultrasound.c ir.c dht11.c
함수
main [54]
read_dht11_da
매크로
DHTPIN [7]
MAXTIMINGS [
변수
dht11_dat [9]
39 laststate = digitalRead(DHTPIN) ;
40 if (counter == 200) break ; // if while breaked by timer, break for
41 if ((i >= 4) && (i % 2 == 0)) {
42 dht11_dat[j / 8] <<= 1 ;
43 if (counter > 20) dht11_dat[j / 8] |= 1 ;
44 j++ ;
45 }
46 }
47
48 if ((j >= 40) && (dht11_dat[4] == ((dht11_dat[0] + dht11_dat[1] + dht11_dat[2] + dht11_dat[3]) & 0xff))) {
49 printf("humidity = %d.%d %% Temperature = %d.%d *C \n", dht11_dat[0], dht11_dat[1], dht11_dat[2], dht11_dat[3])
50 }
51 else printf("Data get failed\n") ;
52 }
53
54 int main(void)
55 {
56 printf("dht11 Raspberry pi\n") ;
57 if (wiringPiSetup() == -1) exit(1) ;
58 while (1) {
59 read_dht11_dat() ;
60 delay(1000) ;
61 }
62 return 0 ;
63 }
64
65
16:40:51: /home/pi/Adafruit_Python_DHT/examples/dht11.c 파일 열음(5).
16:40:51: 하나 또는 그 이상의 세션 파일 불러오기 실패함.
16:43:14: /home/pi/Adafruit_Python_DHT/examples/dht11.c 파일 저장됨.
16:45:56: /home/pi/Adafruit_Python_DHT/examples/dht11.c 파일 저장됨.
16:52:29: /home/pi/Adafruit_Python_DHT/examples/dht11.c 파일 저장됨.
상태
컴파일러
line: 46 / 66 col: 3 sel: 0 삽입 탭 mode: LF encoding: UTF-8 filetype: C scope: read_dht11_dat
pi@raspberrypi:~/Adafruit_Python_DHT/examples $

dht11 in Python

- dht11.py와 dht11_example.py를 강의록 게시판에서 다운받을 것
- python 인터프리터에서 dht11_example.py를 실행

dht11_example.py

```
#dht11_example.py

import RPi.GPIO as GPIO
import dht11 #use this class
import time
import datetime

# initialize GPIO
GPIO.setwarnings(False)
GPIO.setmode(GPIO.BCM)
GPIO.cleanup()

# read data using pin 23
instance = dht11.DHT11(pin=23)

while True:
    result = instance.read()
    if result.is_valid():
        print("Last valid input: " + str(datetime.datetime.now()))
        print("Temperature: %d C" % result.temperature)
        print("Humidity: %d %%" % result.humidity)
    time.sleep(1)
```

192.168.11.6: 원격 데스크톱 연결

pi@raspberrypi: ~/Ad... [LECT - Chromium] examples Thonny - /home/pi/...

Thonny - /home/pi/Adafruit_Python_DHT/examples/dht11_example.py @ 14 : 31

New Load Save Run Debug Over Into Out Stop Zoom Quit Switch to regular mode

dht11_example.py

```
1  #dht11_example.py
2
3  import RPi.GPIO as GPIO
4  import dht11 #use this class
5  import time
6  import datetime
7
8  # initialize GPIO
9  GPIO.setwarnings(False)
10 GPIO.setmode(GPIO.BCM)
11 GPIO.cleanup()
12
13 # read data using pin 23
14 instance = dht11.DHT11(pin=23)
15
16 while True:
17     result = instance.read()
18     if result.is_valid():
19         print("Last valid input: " + str(datetime.datetime.now()))
20         print("Temperature: %d C" % result.temperature)
```

Shell

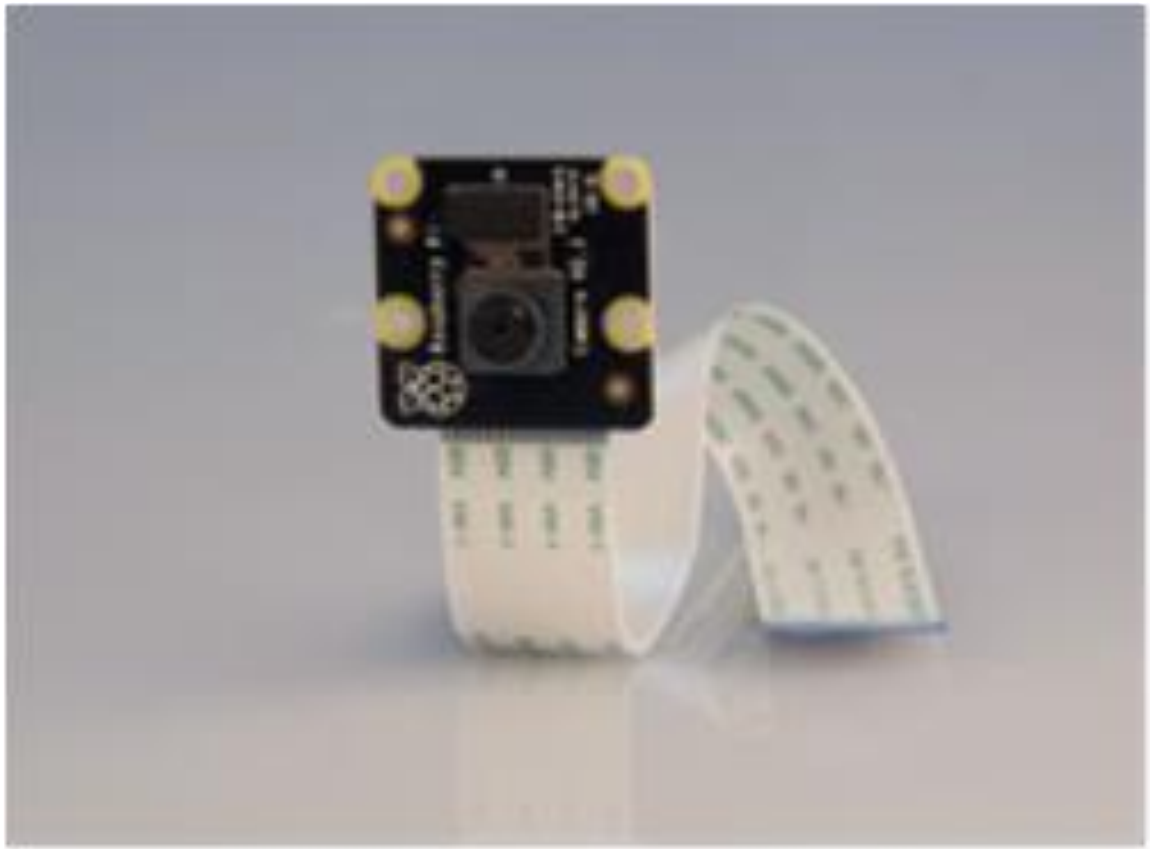
```
Humidity: 42 %
Last valid input: 2020-09-18 17:12:22.225444
Temperature: 26 C
Humidity: 42 %

Python 3.7.3 (/usr/bin/python3)
>>>
```

^C
pi@raspberrypi:~/Adafruit_Python_DHT/examples $

라즈베리파이 실습
- Camera

Camera

- **Raspberry Pi Camera Module**



Camera

Specifications

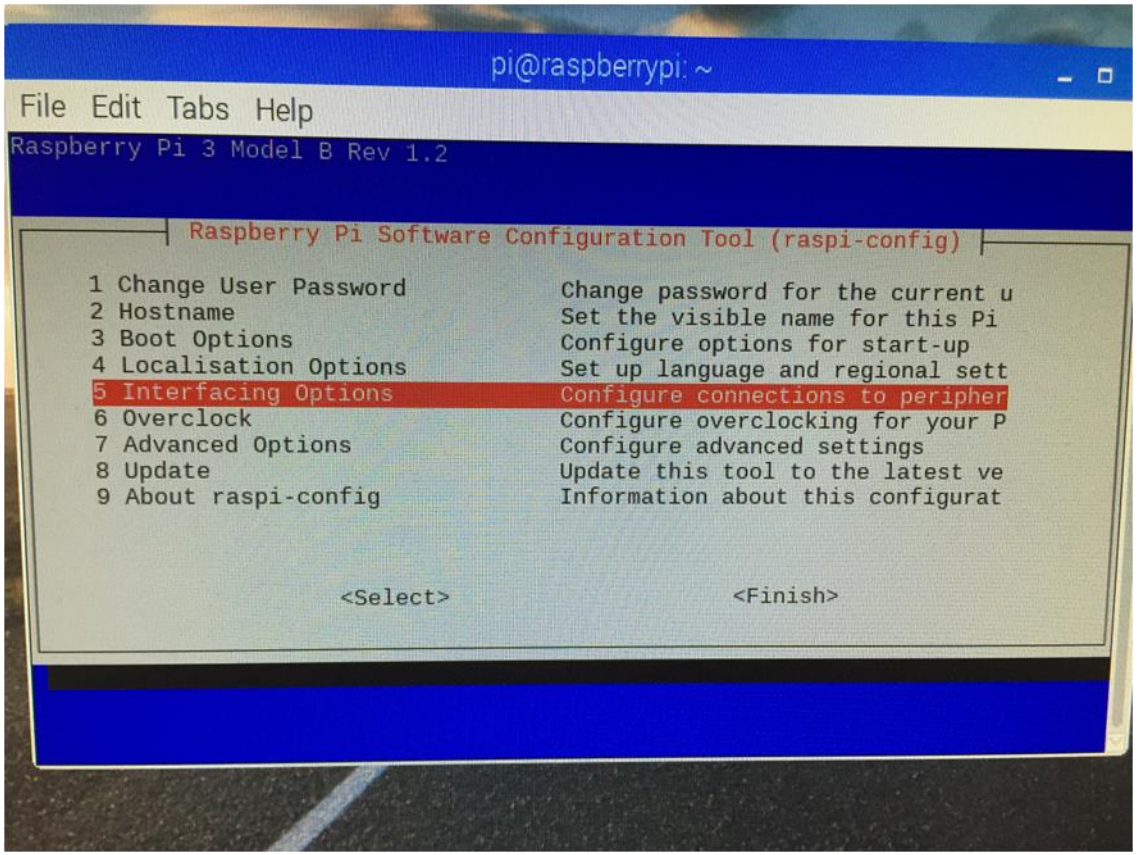
Image Sensor	Sony IMX 219 PQ CMOS image sensor in a fixed-focus module.
Resolution	8-megapixel
Still picture resolution	3280 x 2464
Max image transfer rate	1080p: 30fps (encode and decode) 720p: 60fps
Connection to Raspberry Pi	15-pin ribbon cable, to the dedicated 15-pin MIPI Camera Serial Interface (CSI-2).
Image control functions	Automatic exposure control Automatic white balance Automatic band filter Automatic 50/60 Hz luminance detection Automatic black level calibration
Temp range	Operating: -20° to 60° Stable image: -20° to 60°
Lens size	1/4"
Dimensions	23.86 x 25 x 9mm
Weight	3g

Camera

- 카메라 모듈 인터페이스의 파란 테이프 부분이 이더넷 방향으로

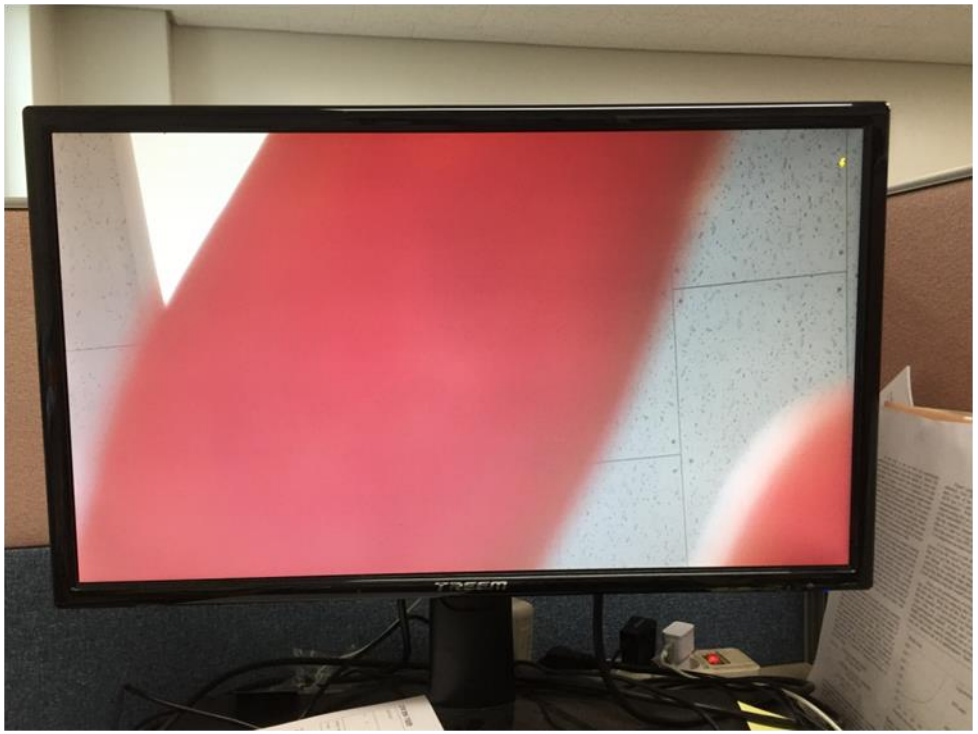
Camera

- **sudo raspi-config > Interfacing options > Camera enable**



Camera - Image

- **5초 뒤 카메라 사진을 찍어 jpg 포맷으로 저장**
  - $ raspistill -o image.jpg



Camera - Image

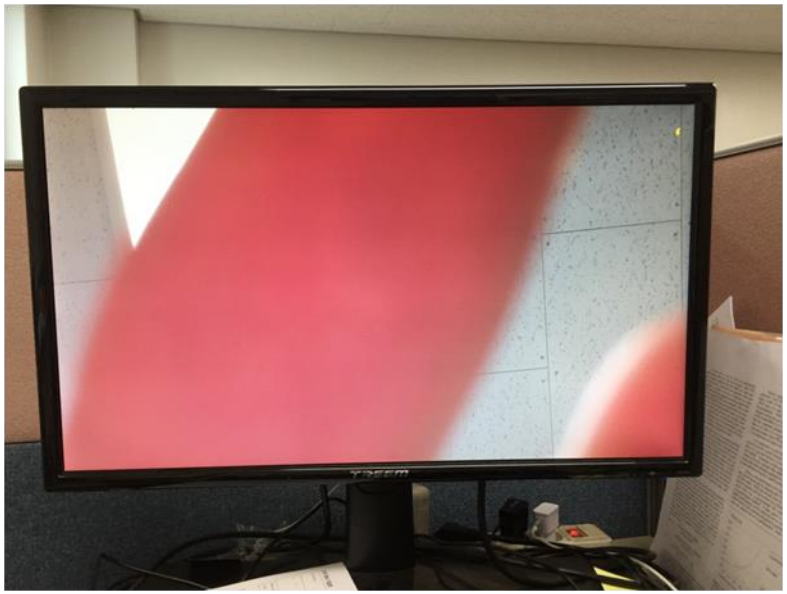
- **raspistill 명령어 옵션**
  - -o 출력 파일명
  - -t 카메라 캡처까지의 시간(default = 5000ms)
  - -vf, -hf 수직, 수평 뒤집기
  - -tl 타임랩스 모드
  - -w, -h 이미지 사이즈
  - -br 이미지 밝기

Camera - image

```
from picamera import PiCamera
from time import sleep

camera = PiCamera()

camera.start_preview() #카메라 미리보기 시작
sleep(5) #5초 뒤 촬영
#python library의 capture 메소드에서 use_video_port가 false인 경우
#이미지 인코딩 과정에서 에러가 발생할 수 있음
camera.capture('/home/pi/image.jpeg', use_video_port = True)
camera.stop_preview()
camera.close()
```

Camera - Video

- **5초 간 동영상을 찍어 h264 format으로 저장**
  - $ raspivid -o video.h264
  - $ omxplayer video.h264



Camera - Video

```
from picamera import PiCamera
from time import sleep

camera = PiCamera()

camera.start_preview() #카메라 미리보기 시작
camera.start_recording('./video.h264')
sleep(5) #5초 동안 촬영
camera.stop_recording()
camera.stop_preview()
camera.close()
```

Camera - rotation

- 180도 회전된 이미지 캡쳐

```
from picamera import PiCamera
from time import sleep

camera = PiCamera()

camera.start_preview()
sleep(5)
camera.rotation = 180
camera.capture('/home/pi/rotation.jpeg', use_video_port = True)
camera.stop_preview()
camera.close()
```

Camera - resolution

- 카메라 해상도 조절

```
from picamera import PiCamera
from time import sleep

camera = PiCamera()

camera.resolution = (1800, 1600)
camera.framerate = 15
camera.start_preview()
sleep(5)
camera.capture('/home/pi/resolution.jpeg', use_video_port = True)
camera.stop_preview()
camera.close()
```

Camera - annotate

- **5초 동안 화면 위에 카운트를 표시한 후 사진이 찍힘**

```
from picamera import PiCamera Color
from time import sleep


camera = PiCamera()
camera.annotate_text_size = 100
camera.annotate_background = Color('black')
camera.annotate_foreground = Color('white')


camera.start_preview()
for i in range(5):
        j = 5 - i
        camera.annotate_text = "%s" % i
        sleep(1)
camera.annotate_text = ""
camera.capture('/home/pi/count.jpg', use_video_port = True)
camera.stop_preview()
camera.close()
```

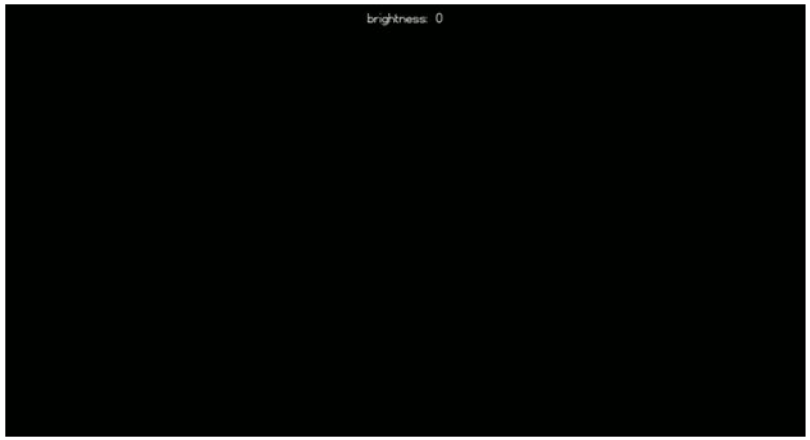
Camera - brightness

- 카메라의 밝기를 점차적으로 증가

```
from picamera import PiCamera
from time import sleep

camera = PiCamera()

camera.start_preview()

for i in range (100):
        camera.annotate_text = "Brightness: %s" % i
        camera.brightness = i
        sleep(0.1)
camera.stop_preview()
camera.close
```

Camera - brightness

- 5초 동안 화면 위에 카운트를 표시한 후 사진이 찍힘



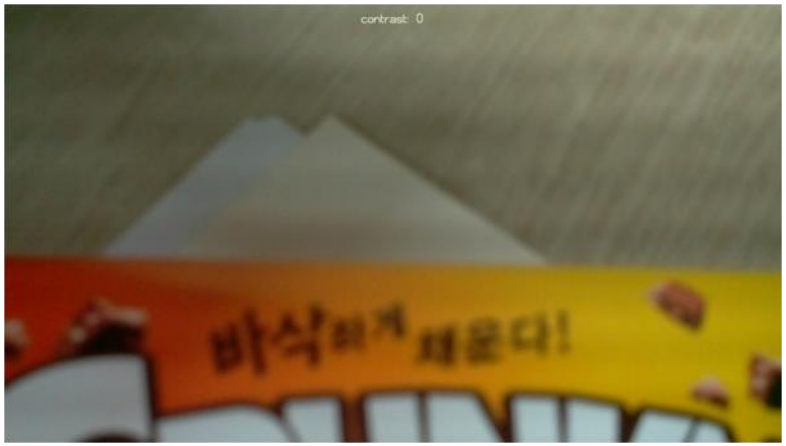
Camera - contrast

- 카메라의 대비를 점차적으로 증가

```
from picamera import PiCamera
from time import sleep

camera = PiCamera()

camera.start_preview()

for i in range (100):
        camera.annotate_text = "Contrast: %s" % i
        camera.contrast = i
        sleep(0.1)
camera.stop_preview()
camera.close
```

Camera - contrast

- 카메라의 대비를 점차적으로 증가



Camera - effect

- 5초마다 이펙트가 변화

```
from picamera import PiCamera
from time import sleep

camera = PiCamera()

camera.start_preview()
for effect in camera.IMAGE_EFFECTS:
        camera.image_effect = effect
        camera.annotate_text = "Effect: %s" % effect
        sleep(5)
        camera.capture("./Eff_%s.jpg" % effect, use_video_port = True)
camera.stop_preview()
camera.close()
```

Camera - effect

라즈베리파이 실습
- 동영상 스트리밍

flask

- 파이썬으로 웹 서비스를 개발하기 위한 웹 프레임워크
- WSGI (Web Server Gateway Interface) 마이크로프레임워크
- 누구나 쉽고 간편하게 웹서버를 만들 수 있음
- 웹프로그램에 파이썬 라이브러리를 사용 가능

flask

- **Flask 파일들**
  - 파이썬 실행 파일
  - html 파일
  - 파이썬 라이브러리
- **Flask 설치**
  - $ sudo apt-get install python-pip
  - $ sudo pip install flask

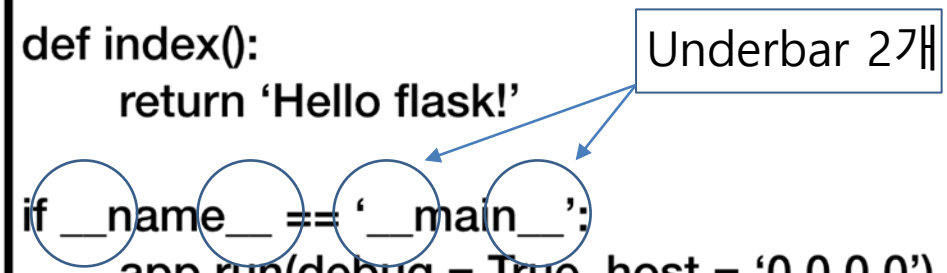
flask 프로젝트

- $ mkdir helloflask
- $ cd ./helloflask
- $ mkdir templates static
- $ mkdir static/css static/js static/img
- $ > routes.py (빈 파이썬 파일 생성)

hello flask

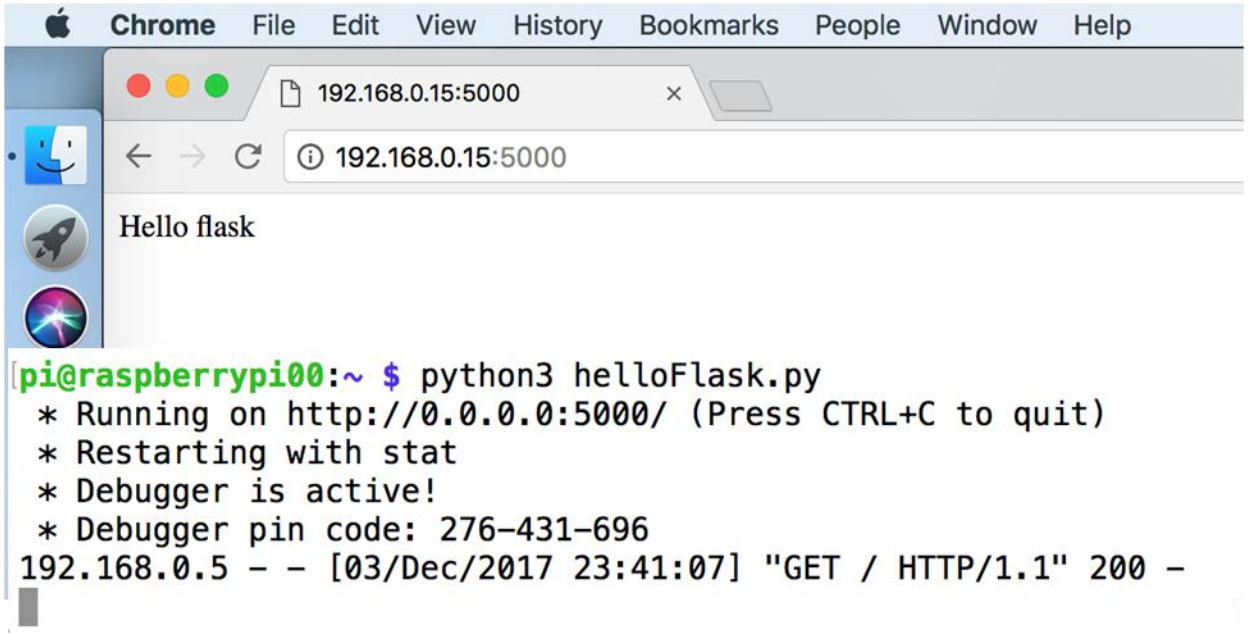
- 코드 작성 후 해당 파일 실행

```
#helloFlask/routes.py

from flask import Flask

app = Flask(__name__)

@app.route('/')

def index():
    return 'Hello flask!'

if __name__ == '__main__':
    app.run(debug = True, host = '0.0.0.0')
```



hello flask

- 다른 PC의 웹브라우저에서 http://[라즈베리파이의 ip 주소]:5000 로 접속
- 라즈베리파이 ip 주소로 접속 또는 localhost로 접속
  - 라즈베리파이 ip 주소확인: # ifconfig 명령



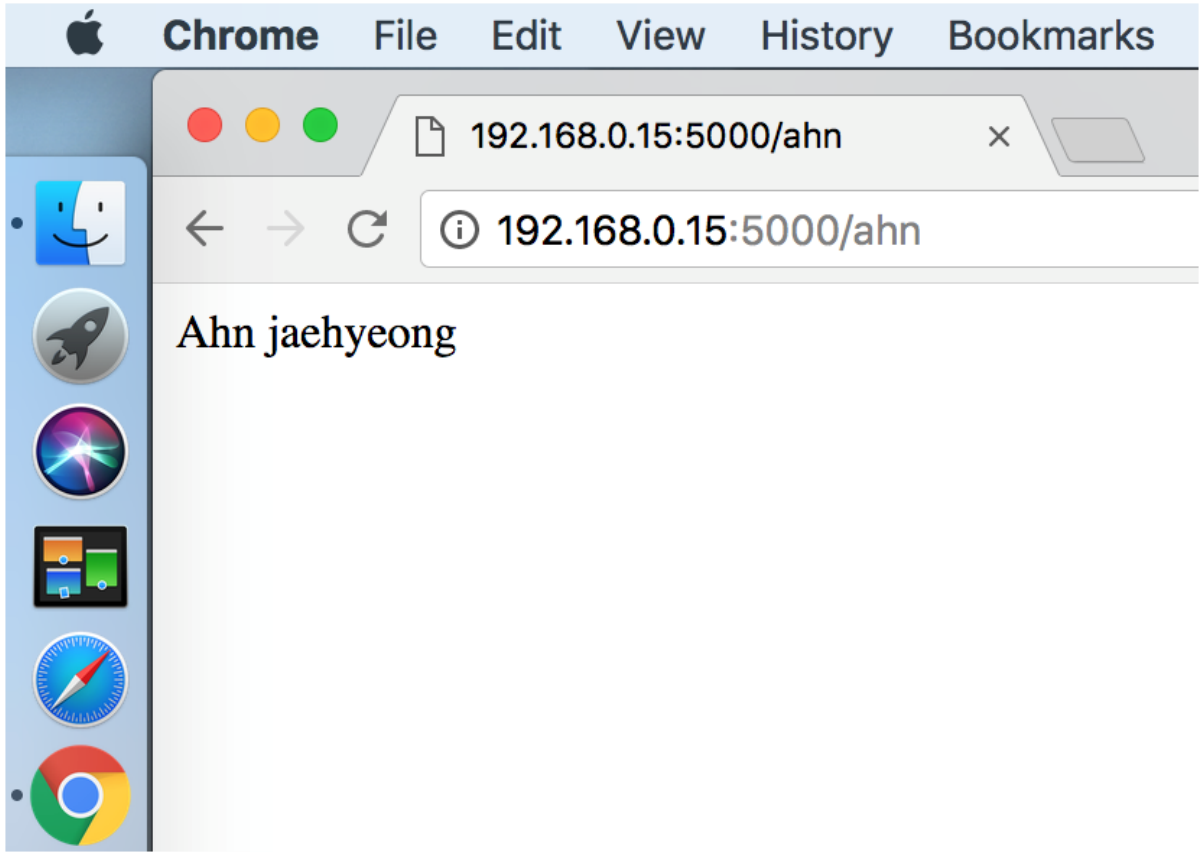
flask - route

```
#helloFlask.py

from flask import Flask

app = Flask(__name__)

@app.route('/')
def index():
    return 'Hello flask!'

@app.route('/ahn')
def ahn():
    return 'Ahn jaehyeong'

if __name__ == '__main__':
    app.run(debug = True, host = '0.0.0.0')
```

Underbar 2개

flask - route

- http://[라즈베리파이 ip 주소]:5000/ahn



flask - templates

- $ cd templates
- $ > index.html
- $ > layout.html (빈 html 파일 생성)
- $ cd ..

templates

- **layout.html**

```
<!DOCTYPE html>
<html>
    <head>
        <title>Flask App</title>
    </head>
    <body>
        <h1>
            Flask App
        </h1>
        <div class="container">
            {% block content %}
            {% endblock %}
        </div>
    </body>
</html>
```

templates

- **index.html**

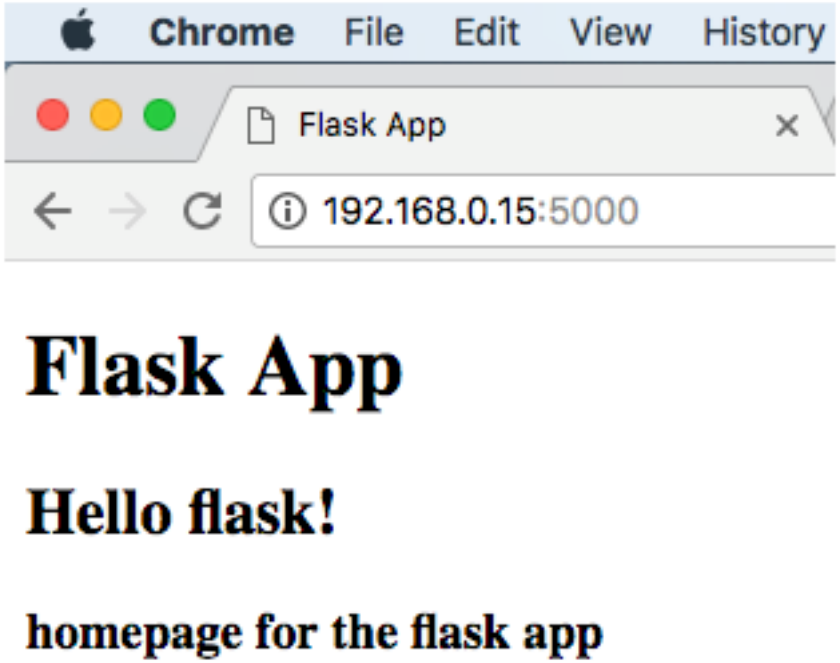
```
{% extends “layout.html” %}
{% block content %}
        <div class=“jumbo”>
            <h2>Hello Flask!</h2>
            <h3>home page for the flask app</h3>
        </div>
{% endblock %}
```

flask - templates

```
#app.py

from flask import Flask, render_template

app = Flask(__name__)

@app.route('/')
def index():
        return render_template('index.html')

if __name__ == '__main__':
        app.run(debug = True, host = '0.0.0.0')
```

flask - templates

- http://[라즈베리파이 아이피 주소]:5000



flask - templates

```
#app.py

from flask import Flask, render_template

app = Flask(__name__)

@app.route('/')
def index():
    return render_template('index.html')

@app.route('/hello/<_name>')
def hello(_name):
    return render_template('page.html', name=_name)

if __name__ == '__main__':
    app.run(debug = True, host = '0.0.0.0')
```

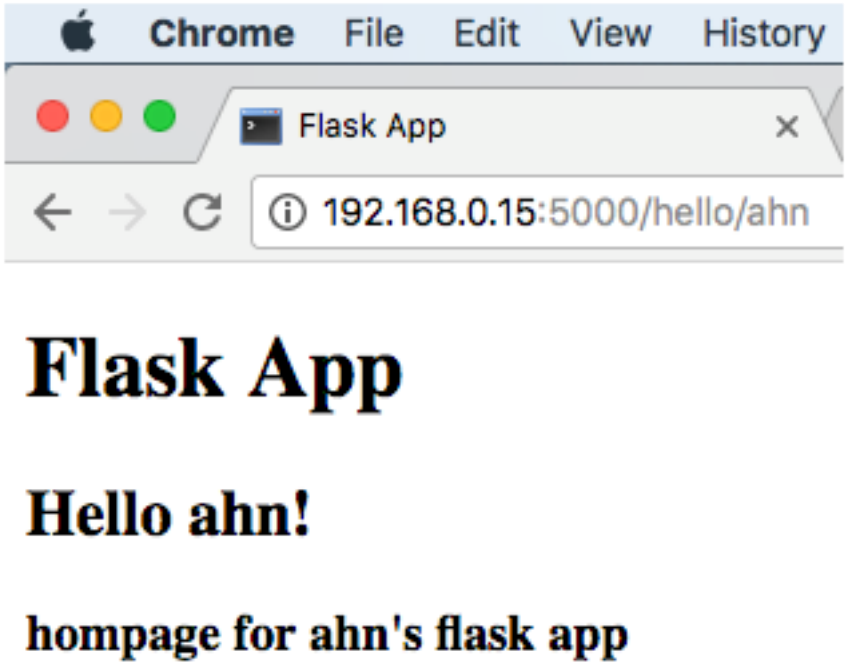
flask - templates

- $ cd templates/
- $ > page.html
- $ cd ..

```
{% extends “layout.html” %}
{% block content %}
        <div class=“jumbo”>
            <h2>Hello {{ name }}!</h2>
            <h3>home page for {{ name }}’s flask app</h3>
        </div>
{% endblock %}
```

flask - templates

- http:///[ip address]/hello/ahn



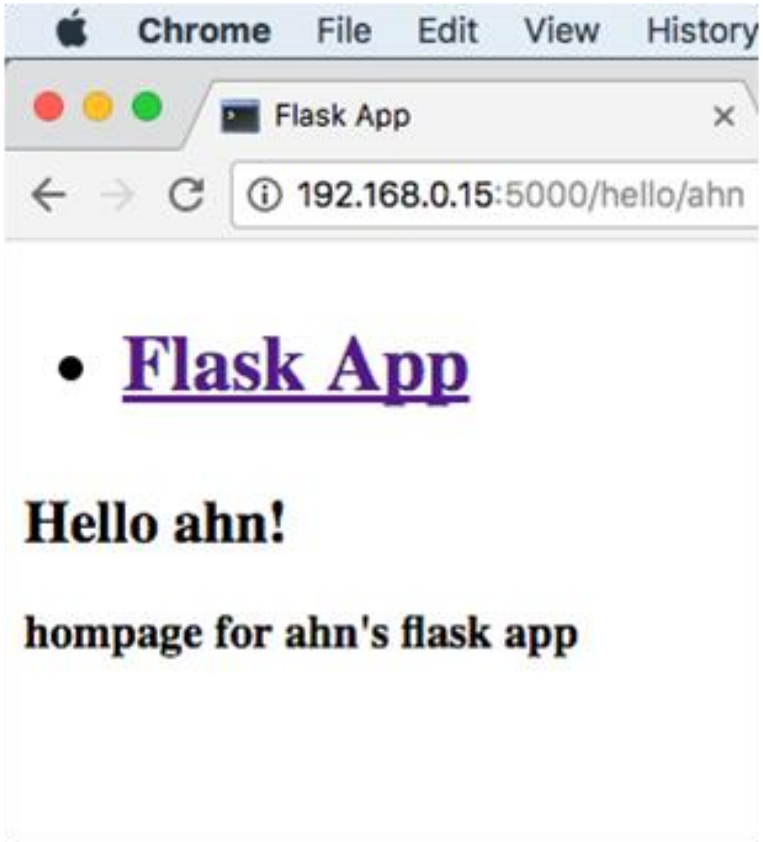
flask - templates

- layout.html 을 다음과 같이 수정

```
...
        <h1><strong><nav>
            <ul class="menu">
                <a href="{{ url_for('hello', _name = 'kim') }}">
                    Flask App
                </a>
            </ul>
        </nav></strong></h1>
        <div class="container">
            {% block content %}
            {% endblock %}
        </div>
...
```

flask - templates

- Flask App을 클릭시 index() 파이썬 함수가 실행됨



flask - camera

```
#camera.py

import time
import io
import threading
import picamera

class Camera(object):
    thread = None
    frame = None
    last_access = 0

    def initialize(self):
        if Camera.thread is None:
            Camera.thread = threading.Tread(target=self._thread)
            Camera.thread.start()
```

Thread

flask - camera

```
            while self.frame is None:
                time.sleep(0)

    def get_frame(self):
        Camera.last_access = time.time()
        self.initialize()
        return self.frame

    @classmethod
    def _thread(cls):
        with picamera.PiCamera() as camera:
            camera.resolution(320, 240)

            camera.start_preview()
            time.sleep(2)
```

flask - camera

- **camera.py**

```
                stream = io.BytesIO()
                for foo in camera.capture_continuous(stream, 'jpeg',
use_video_port=True):
                        stream.seek(0)
                        cls.frame = stream.read()

                        stream.seek(0)
                        stream.truncate()

                        if time.time() - cls.last_access > 10:
                                break
        cls.thread = None
```

flask - camera

```
<!— templates/index.html —>
<html>
        <head>
                <title>Pet Feeder</title>
                <script src="//code.jquery.com/jquery-1.11.3.min.js"></script>
        </head>
        <body>
                <h1>Camera from RPi</h1>
                <img src = "{{ url_for('video_feed') }}">
                <br><br>
                <button>Button</button>
```

flask - camera

- **templates/index.html**

```
        <script>
            $( 'button' ).click(function() {
                $.get( 'button' ).done(function( data ) {
                    if(data == '1'){
                        alert('Success!');
                    } else {
                        alert('Fail!');
                    }
                });
            });
        </script>
    </body>
</html>
```

flask - camera

```
# app.py

from flask import Flask, render_template, Response
from camera import Camera

app = Flask(__name__)

@app.route('/')
def index():
    return render_template('index.html')

def gen(camera):
    while True:
        frame = camera.get_frame()
        yield (b'—frame\r\n' b'Content-Type: image\r\n\r\n' + frame
+ b'\r\n')
cam = Camera()
```

-- 27개

flask - camera

- app.py

```
@app.route('/video_feed')
def video_feed():
	return Response(gen(cam), mimetype='multipart/x-mixed-
replace; boundary=frame')

@app.route("/button")
def button():
	try:
		#make your function
		return "1"
	except:
		pass
	return "0"

if __name__ == '__main__':
	app.run(host='0.0.0.0', debug=True, threaded=True)
```

flask - camera