

라즈베리파이를 이용한 인공지능 스피커 만들기

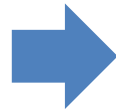
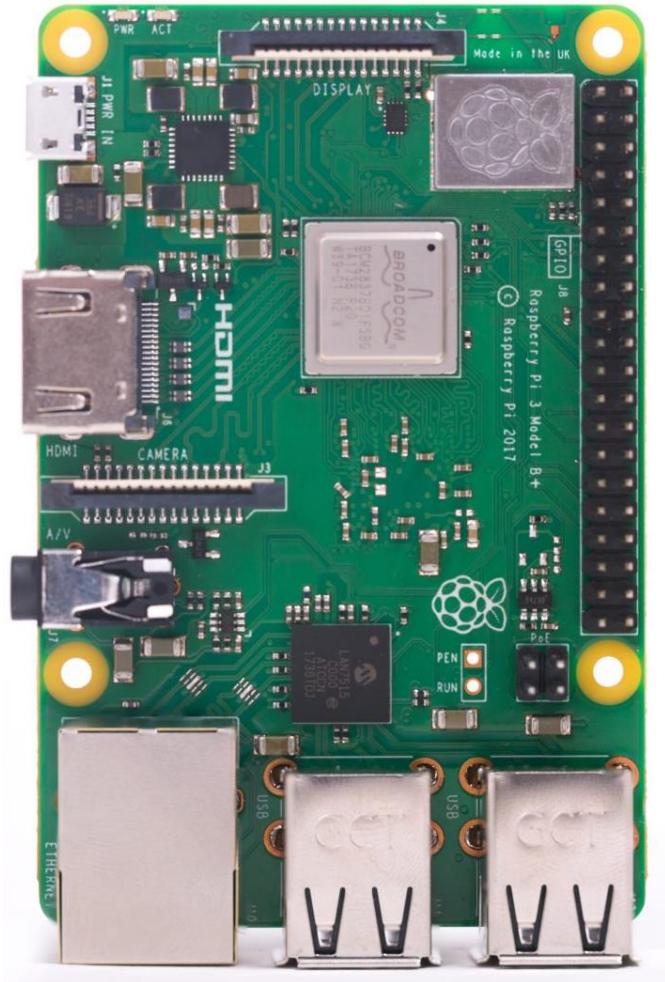
Part 8

김영준

목원대학교 겸임교수
煎 마이크로소프트 수석연구원
헬로앱스 대표이사
[Http://www.helloapps.co.kr](http://www.helloapps.co.kr)
splduino@gmail.com

라즈베리파이와 아두이노 GPIO 연결

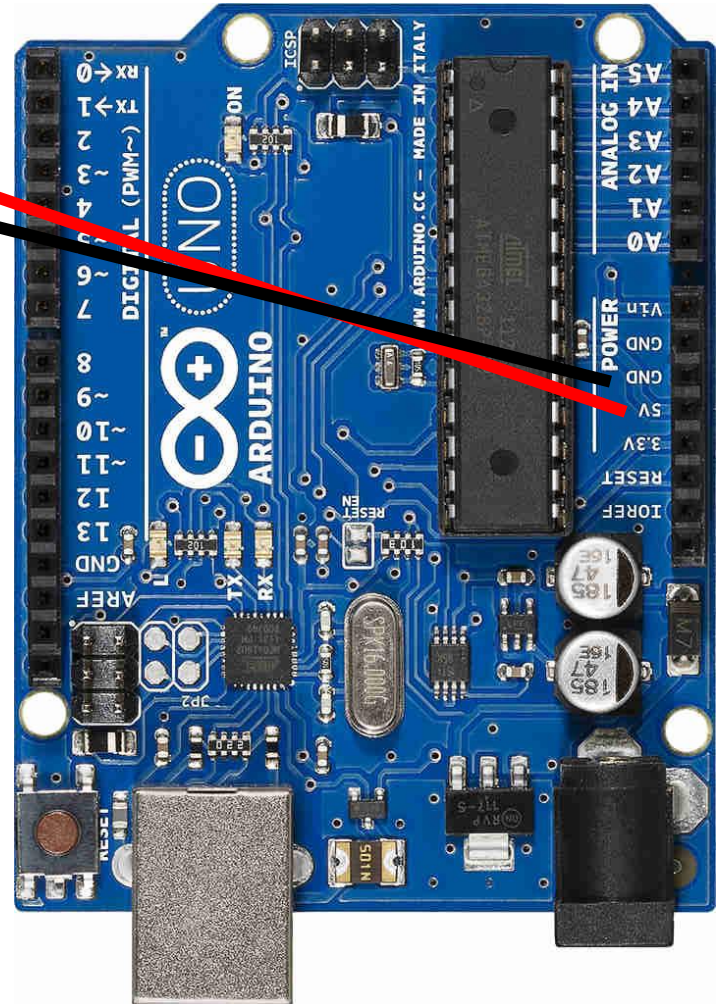
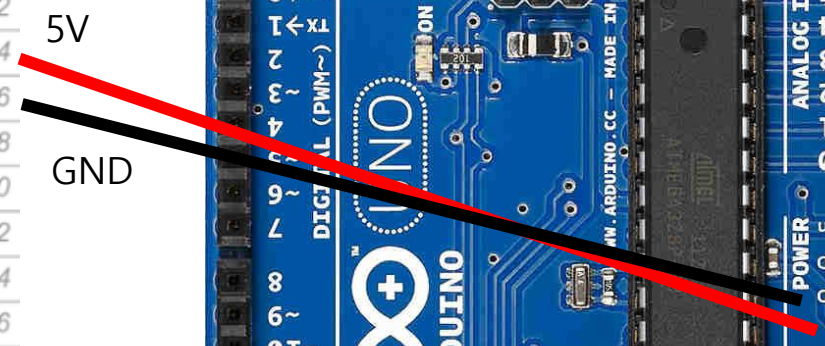
라즈베리파이와 아두이노 연결



Pin#	NAME	NAME	Pin#
01	3.3v DC Power	DC Power 5v	02
03	GPIO02 (SDA1 , I2C)	DC Power 5v	04
05	GPIO03 (SCL1 , I2C)	Ground	06
07	GPIO04 (GPIO_GCLK)	(TXD0) GPIO14	08
09	Ground	(RXD0) GPIO15	10
11	GPIO17 (GPIO_GEN0)	(GPIO_GEN1) GPIO18	12
13	GPIO27 (GPIO_GEN2)	Ground	14
15	GPIO22 (GPIO_GEN3)	(GPIO_GEN4) GPIO23	16
17	3.3v DC Power	(GPIO_GEN5) GPIO24	18
19	GPIO10 (SPI_MOSI)	Ground	20
21	GPIO09 (SPI_MISO)	(GPIO_GEN6) GPIO25	22
23	GPIO11 (SPI_CLK)	(SPI_CE0_N) GPIO08	24
25	Ground	(SPI_CE1_N) GPIO07	26
27	ID_SD (I2C ID EEPROM)	(I2C ID EEPROM) ID_SC	28
29	GPIO05	Ground	30
31	GPIO06	GPIO12	32
33	GPIO13	Ground	34
35	GPIO19	GPIO16	36
37	GPIO26	GPIO20	38
39	Ground	GPIO21	40

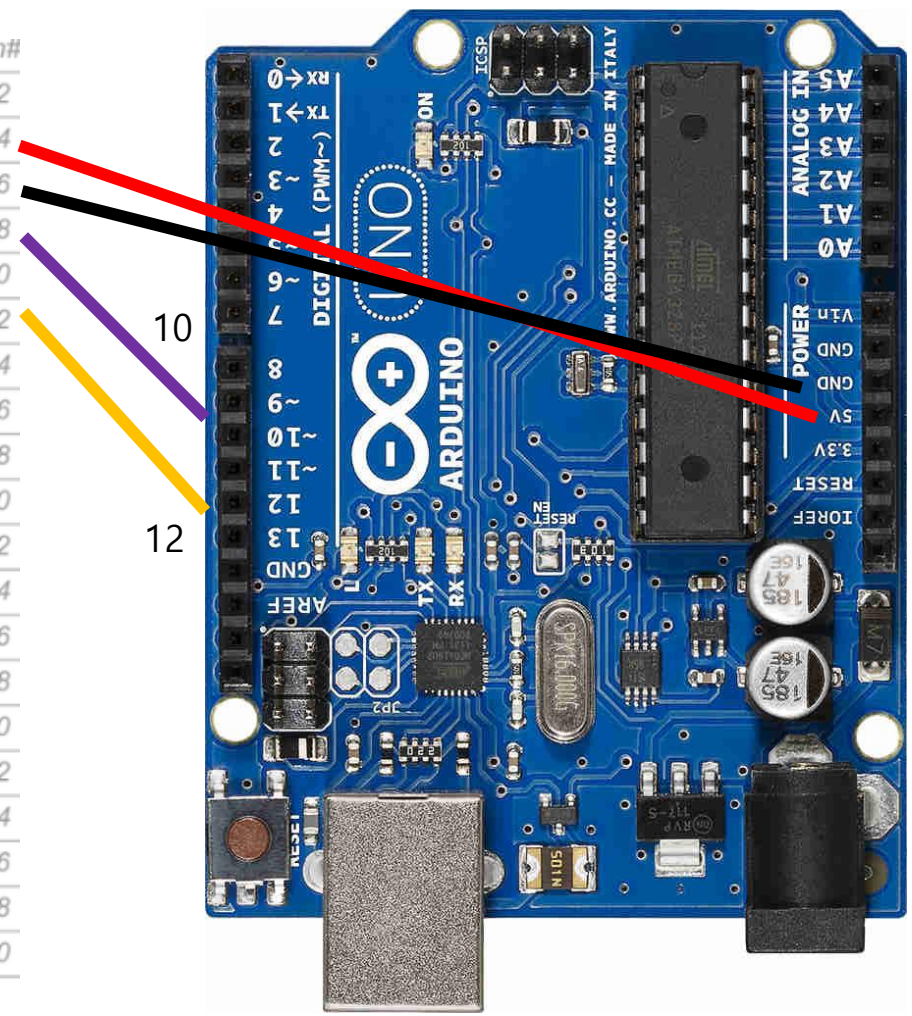
라즈베리파이와 아두이노 연결

Pin#	NAME	NAME	Pin#
01	3.3v DC Power	DC Power 5v	02
03	GPIO02 (SDA1 , I2C)	DC Power 5v	04
05	GPIO03 (SCL1 , I2C)	Ground	06
07	GPIO04 (GPIO_GCLK)	(TXD0) GPIO14	08
09	Ground	(RXD0) GPIO15	10
11	GPIO17 (GPIO_GEN0)	(GPIO_GEN1) GPIO18	12
13	GPIO27 (GPIO_GEN2)	Ground	14
15	GPIO22 (GPIO_GEN3)	(GPIO_GEN4) GPIO23	16
17	3.3v DC Power	(GPIO_GEN5) GPIO24	18
19	GPIO10 (SPI_MOSI)	Ground	20
21	GPIO09 (SPI_MISO)	(GPIO_GEN6) GPIO25	22
23	GPIO11 (SPI_CLK)	(SPI_CE0_N) GPIO08	24
25	Ground	(SPI_CE1_N) GPIO07	26
27	ID_SD (I2C ID EEPROM)	(I2C ID EEPROM) ID_SC	28
29	GPIO05	Ground	30
31	GPIO06	GPIO12	32
33	GPIO13	Ground	34
35	GPIO19	GPIO16	36
37	GPIO26	GPIO20	38
39	Ground	GPIO21	40



라즈베리파이와 아두이노 연결

Pin#	NAME	NAME	Pin#
01	3.3v DC Power	DC Power 5v	02
03	GPIO02 (SDA1 , I2C)	DC Power 5v	04
05	GPIO03 (SCL1 , I2C)	Ground	06
07	GPIO04 (GPIO_GCLK)	(TXD0) GPIO14	08
09	Ground	(RXD0) GPIO15	10
11	GPIO17 (GPIO_GEN0)	(GPIO_GEN1) GPIO18	12
13	GPIO27 (GPIO_GEN2)	Ground	14
15	GPIO22 (GPIO_GEN3)	(GPIO_GEN4) GPIO23	16
17	3.3v DC Power	(GPIO_GEN5) GPIO24	18
19	GPIO10 (SPI_MOSI)	Ground	20
21	GPIO09 (SPI_MISO)	(GPIO_GEN6) GPIO25	22
23	GPIO11 (SPI_CLK)	(SPI_CE0_N) GPIO08	24
25	Ground	(SPI_CE1_N) GPIO07	26
27	ID_SD (I2C ID EEPROM)	(I2C ID EEPROM) ID_SC	28
29	GPIO05	Ground	30
31	GPIO06	GPIO12	32
33	GPIO13	Ground	34
35	GPIO19	GPIO16	36
37	GPIO26	GPIO20	38
39	Ground	GPIO21	40



GPIO14 -> 디지털 10, GPIO18 -> 디지털 12

라즈베리파이 GPIO 코딩

GPIO 라이브러리 설치

Import the GPIO package

To simplify software access to the General Purpose Input/Output (GPIO) pins on the Raspberry Pi, install the [RPi.GPIO](#) package in the virtual environment.

```
(env) $ pip install RPi.GPIO
```



구글 어시스턴트용 (env)용
pip install RPi.GPIO

일반 파이썬 코딩용

```
sudo apt-get install rpi.gpio
```

파이썬 코딩

```
import RPi.GPIO as GPIO
```

```
import time
```

```
GPIO.setmode(GPIO.BCM)
```

```
GPIO.setup(18, GPIO.OUT, initial=GPIO.LOW)
```

```
while True:
```

```
    GPIO.output(18, 1)
```

```
    time.sleep(1)
```

```
    GPIO.output(18, 0)
```

```
    time.sleep(1)
```


구글 어시스턴트에 연결

LED 제어

```
import RPi.GPIO as GPIO
```

Modify the code to set the output pin initially to the low logic state. Do this in the `main()` function, before processing events:

```
with Assistant(credentials, device_model_id) as assistant:  
    events = assistant.start()  
  
    device_id = assistant.device_id  
    print('device_model_id:', device_model_id)  
    print('device_id:', device_id + '\n')  
    GPIO.setmode(GPIO.BCM)  
    GPIO.setup(25, GPIO.OUT, initial=GPIO.LOW)  
    ...
```

18번으로 수정

LED 제어

```
if command == "action.devices.commands.OnOff":
    if params['on']:
        print('Turning the LED on.')
        GPIO.output(25, 1)
    else:
        print('Turning the LED off.')
        GPIO.output(25, 0)
```

18번으로 수정

