

라즈베리파이를 이용한 인공지능 스피커 만들기

Part 9

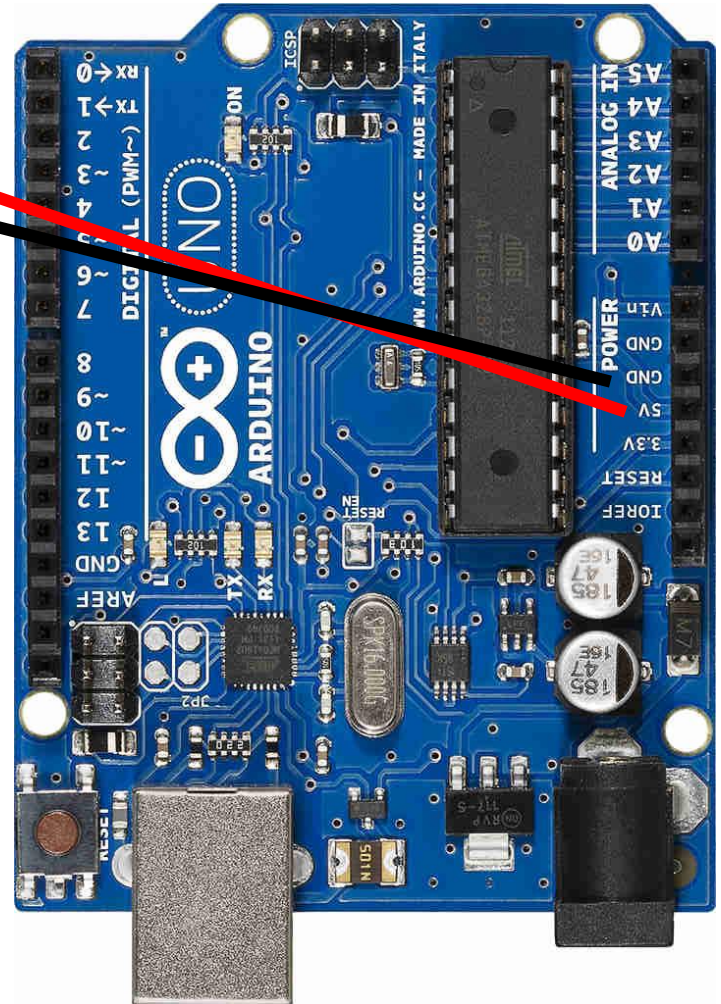
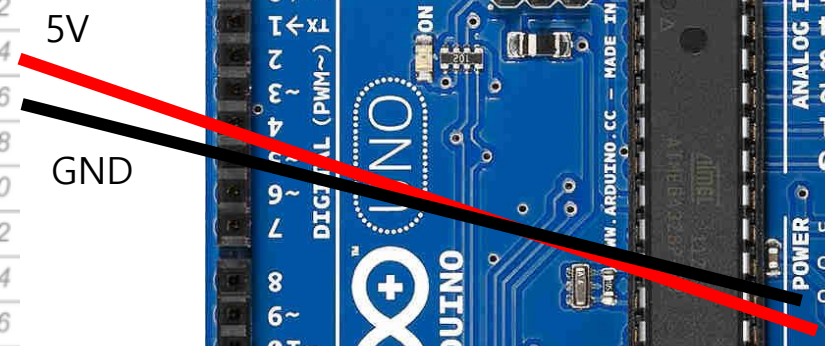
김영준

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

라즈베리파이와 아두이노 시리얼 연결

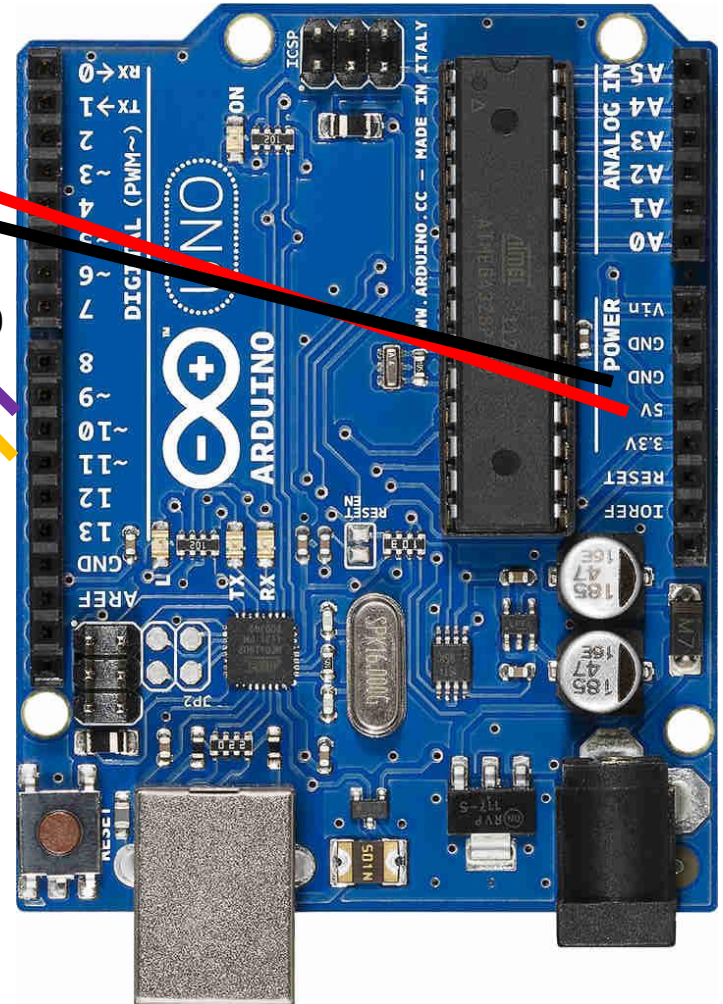
라즈베리파이와 아두이노 연결

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



GPIO14 -> 디지털 10, GPIO15 -> 디지털 11

라즈베리파이 시리얼 통신

시리얼 통신 라이브러리 설치

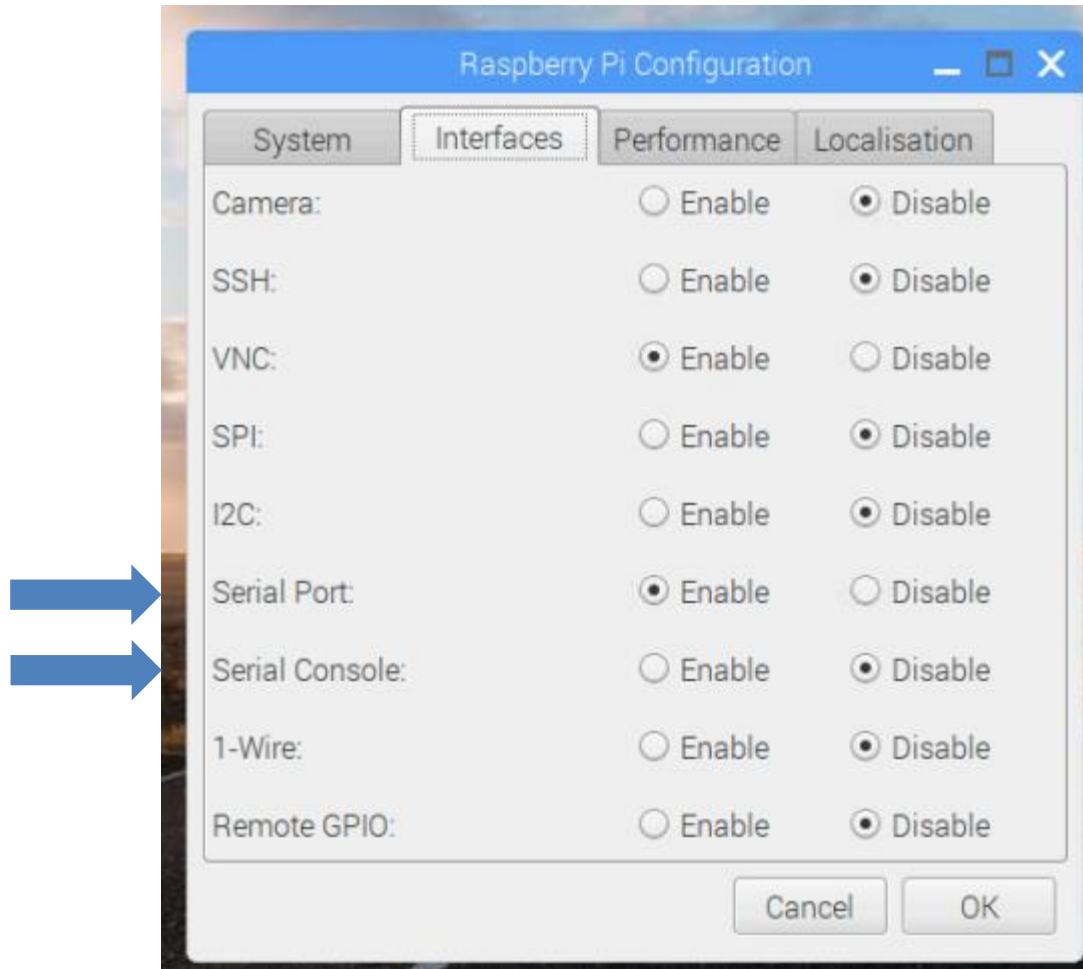
구글 어시스턴트용 (env)용

```
(env)$ pip install pyserial
```

일반 파이썬 코딩용

```
sudo apt-get install python-serial
```

라즈베리파이 환경 설정



라즈베리파이 환경 설정

블루투스 Turn Off



라즈베리파이 환경 설정

터미널에서

```
sudo nano /boot/config.txt
```

명령어 실행 후, 맨 아래에 아래 라인 추가

```
dtoverlay=pi3-disable-bt
```

라즈베리파이 환경 설정

터미널에서 아래 명령어 실행

```
sudo systemctl disable hciuart
```

재시작

시리얼 포트 찾기

터미널에서 아래 명령어 실행

```
dmesg | grep tty
```

시리얼 통신 예제

시리얼로 LED 제어

```
import serial  
import time
```

```
ser = serial.Serial('/dev/ttyAMA0', baudrate=9600, timeout=1.0)
```

```
while True:
```

```
    ser.write("LED:on;".encode())  
    time.sleep(1)
```

```
    ser.write("LED:off;".encode())  
    time.sleep(1)
```

구글 어시스턴트에 연결

Import 추가

```
#####  
import subprocess as sp  
import serial  
import time  
#####
```

시리얼 통신 초기화 명령어 추가

...

```
ser = serial.Serial('/dev/ttyAMA0', baudrate=9600, timeout=1.0)
```

```
if ser.isOpen():  
    print("Serial Opened!")
```

```
def process_event(event):  
    """Pretty prints events.  
    ...
```


전역변수 선언 추가

```
.....
```

```
global ser
```

```
if event.type == EventType.ON_CONVERSATION_TURN_STARTED:  
    print()
```

```
print(event)
```

```
...
```

시리얼 통신 LED 제어 기능 추가

```
for command, params in event.actions:
```

```
    print('Do command', command, 'with params', str(params))
```

```
    if command == "action.devices.commands.OnOff":
```

```
        if params['on']:
```

```
            print('Turning the LED on.')
```

```
            s = "LED:on;"
```

```
            print(s)
```

```
            ser.write(s.encode())
```

```
        else:
```

```
            print('Turning the LED off.')
```

```
            s = "LED:off;"
```

```
            print(s)
```

```
            ser.write(s.encode())
```

밝기 제어하기

밝기 제어하기

```
if command == "action.devices.commands.BrightnessAbsolute":  
    if params['brightness']:  
        if params['brightness'] > 50:  
            print('brightness > 50')  
        else:  
            print('brightness <= 50')
```

```
s = 'brightness:' + str(params['brightness']) + ";"  
print(s)  
ser.write(s.encode())
```

색상 제어하기

색상 제어하기

```
if command == "action.devices.commands.ColorAbsolute":
    if params['color']:
        if params['color'].get('name') == "blue":
            print('The color is blue.')
        else:
            print('The color is not blue.')

    s = 'color:' + params['color'].get('name') + ";"
    print(s)
    ser.write(s.encode())
```