

임베디드응용 및 실습

(8주차 과제)

2020161047 박종혁

다음은 소스코드입니다.

```
import serial
import RPi.GPIO as GPIO
import time
import threading

PWMA = 18
PWMB = 23
AIN1 = 22
AIN2 = 27
BIN1 = 25
BIN2 = 24
GPIO.setwarnings(False)
GPIO.setmode(GPIO.BCM)
GPIO.setup(PWMA,GPIO.OUT)
GPIO.setup(PWMB,GPIO.OUT)
GPIO.setup(AIN1,GPIO.OUT)
GPIO.setup(AIN2,GPIO.OUT)
GPIO.setup(BIN1,GPIO.OUT)
GPIO.setup(BIN2,GPIO.OUT)
L_Motor = GPIO.PWM(PWMA,500)
R_Motor = GPIO.PWM(PWMB,500)
L_Motor.start(0)
R_Motor.start(0)
bleSerial = serial.Serial("/dev/ttyS0", baudrate = 9600, timeout=1)
gData = ""
# 스레드를 수행할 함수를 정의
def serial_thread():
    global gData
    while True:
        data = bleSerial.readline()
        data = data.decode()
        gData = data
```

자동차의 동작을 수행할 코드를 main함수 안에 작성

```
def main():
    global gData
    try:
        while True:
            if gData.find("go") >=0:
                GPIO.output(AIN1,0)
                GPIO.output(AIN2,1)
                L_Motor.ChangeDutyCycle(50)
                GPIO.output(BIN1,0)
                GPIO.output(BIN2,1)
                R_Motor.ChangeDutyCycle(50)
                time.sleep(1)
                gData = ""
                print("ok go")
            elif gData.find("back")>=0:
                GPIO.output(AIN1,1)
                GPIO.output(AIN2,0)
                L_Motor.ChangeDutyCycle(50)
                GPIO.output(BIN1,1)
                GPIO.output(BIN2,0)
                R_Motor.ChangeDutyCycle(50)
                time.sleep(1)
                gData = ""
                print("ok back")
            elif gData.find("left")>=0:
                GPIO.output(AIN1,0)
                GPIO.output(AIN2,0)
                GPIO.output(BIN1,0)
                GPIO.output(BIN2,1)
                R_Motor.ChangeDutyCycle(50)
                time.sleep(1)
                gData = ""
                print("ok left")
            elif gData.find("right")>=0:
                GPIO.output(AIN1,0)
                GPIO.output(AIN2,1)
```

```

        GPIO.output(BIN1,0)
        GPIO.output(BIN2,0)
        L_Motor.ChangeDutyCycle(50)
        time.sleep(1)
        gData = ""
        print("ok right")
    elif gData.find("stop")>=0:
        GPIO.output(AIN1,0)
        GPIO.output(AIN2,0)
        L_Motor.ChangeDutyCycle(50)
        GPIO.output(BIN1,0)
        GPIO.output(BIN2,0)
        R_Motor.ChangeDutyCycle(50)
        time.sleep(1)
        gData = ""
        print("ok stop")

except KeyboardInterrupt:
    pass
# main함수가 실행될 때 thread가 수행하도록 if문으로 작성
if __name__ == '__main__':
    task1 = threading.Thread(target = serial_thread)
    task1.start()
    main()
    bleSerial.close()

```

코드해석:

핀 번호를 각각 설정해줍니다.

그리고 GPIO.setup()으로 활성화 시켜줍니다.

왼쪽과 오른쪽의 모터를 구동가능상태로 다음과 같이 만들어주고 블루투스 시리얼 연결을 합니다.

```
L_Motor = GPIO.PWM(PWMA,500)
```

```
R_Motor = GPIO.PWM(PWMB,500)
```

```
L_Motor.start(0)
```

```
R_Motor.start(0)
```

```
bleSerial = serial.Serial("/dev/ttyS0", baudrate = 9600, timeout=1)
```

gData에는 입력받은 문자를 저장하는 변수로 사용합니다.

serial_thread()로 블루투스로 연결된 기기의 문자열 값을 가져와서 decode합니다.

decode한 값을 gData에 저장해줍니다.

main함수를 생성하고 구현시킬 동작을 코딩합니다.

gData.find()로 블루투스로 연결된 기기로부터 온 문자값을 인식하고 일치하는 문자가 들어오면 직진,우회전,좌회전,후진,정지를 수행합니다.

조건문으로 들어오는 문자에 맞게 동작하도록 코딩합니다.

GPIO.output()의 파라미터로 모터의 동작을 제어하여 방향을 제어합니다.

마지막으로 if문을 사용하여 main함수가 동작할 때 스레드가 동작하도록 하였습니다.

serial_thread()를 target으로 하여 스레드를 수행하고 main함수도 수행하도록 하였습니다. 따라서 입력받는 작업과 동작을 수행하는 작업을 거의 동시에 수행할 수 있습니다.

실행결과를 영상으로 올립니다.