



# 프로세서응용종합설계

## 5주차 실습 결과보고서

담당교수 : 황신환 교수  
님

제출일자 : 20.11.11

학 과 : 전자공학과

학 년 : 3학년

이 름 : 배준성

학 번 :



**한국외국어대학교**  
HANKUK UNIVERSITY OF FOREIGN STUDIES

## 1. 코드

```
import smbus
import time

address = 0x48  #ADC, DAC I2C 주소
AOUT1 = 0x40
AIN2 = 0x42
bus = smbus.SMBus(1)  #I2C버스 초기화

while True:
    bus.write_byte(address, AIN2)  #AIN2핀을 이용하여 아날로그 값을 읽어
    오도록 설정, AIN2핀은 가변저항과 연결

    value = bus.read_byte(address)  #아날로그값 읽음
    volt = value*3.3 / 255  #전압으로 환산

    print("ADC - 단계/전압 : {:>5}/{:>5.3f}".format(value, volt))  #웹화면에 값
    표시

    bus.write_byte_data(address, AOUT1, value)  #AOUT1을 이용하여 출력,
    AOUT1핀은 LED에 연결

    time.sleep(0.1)
```

## 2. 결과 사진

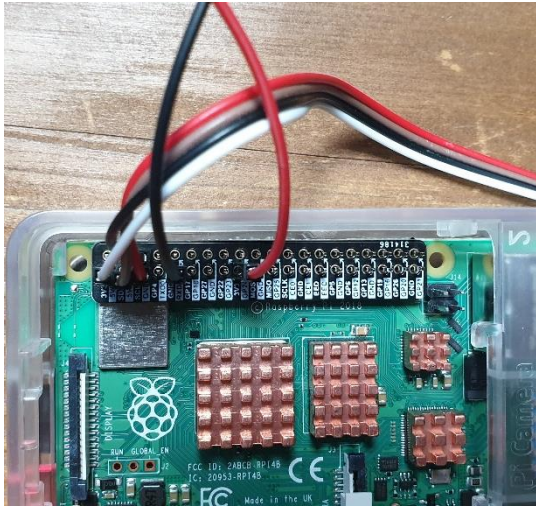


사진 1 - 라즈베리파이 연결 부

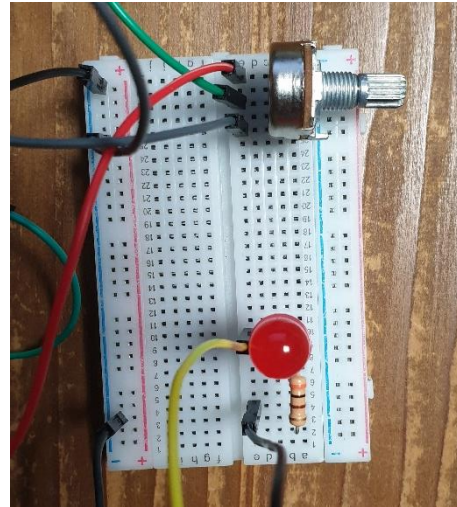


사진 2 - 브레드보드 연결 부

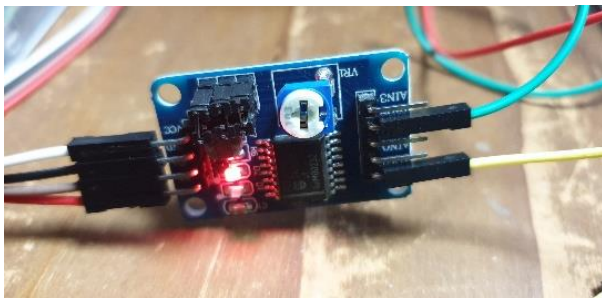


사진 3 - PCF8591 연결 부

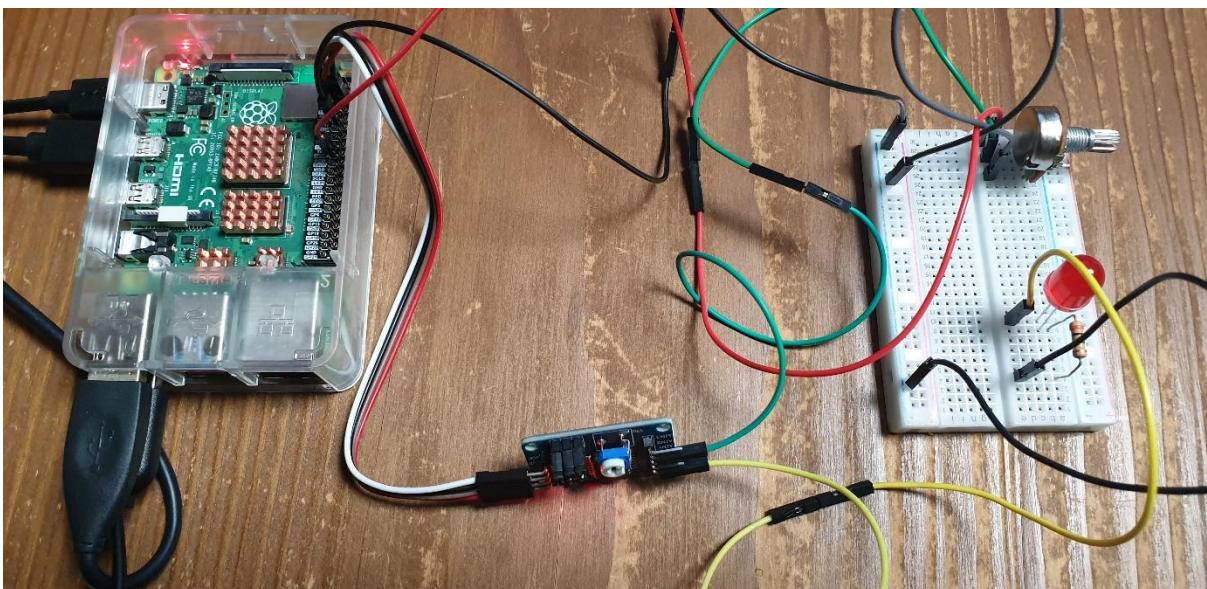


사진 4 - 라즈베리파이를 포함한 전체 회로

### 3. 결론 및 고찰

본 실습을 통하여 ADC/DAC I2C통신을 사용해 보았다. PCF8591 모듈을 이용하여 가변저항의 아날로그 값을 읽어 디지털 신호로 바꾸어 주었다. 다음으로 디지털 신호를 다시 아날로그 신호로 출력해준다. 이러한 방법으로 가변저항을 이용하여 LED의 밝기를 제어할 수 있다.

PCF8591 모듈은 8bit 분해능이므로  $2^8$ 단계로 나누어 신호를 표현한다. 입력 전압이 3.3V이므로 각 단계는  $3.3/256V$  차이가 있다. 따라서 우리는 가변저항에서 읽어온 값을 255로 나누어 준 후 3.3을 곱하여 전압값으로 환산할 수 있다. (256으로 나누지 않고 255로 나누어 주는 이유는 0~255로 256개의 단계를 표현하기 때문이다.)

가변저항으로부터 아날로그 값을 읽어 오기위해서는 AIN2핀을 사용하였으며 LED로 출력하기 위해서 AOUT1핀을 사용하였다. 아날로그 값을 읽어오도록 `bus.write_byte` 함수를 사용하였는데 I2C통신이 가능하도록 마스터에 슬레이브 (PCF8591)의 주소값을 입력해주고, 아날로그 값을 읽어올 수 있도록 핀 주소값 또한 입력해 주었다. 이렇게 읽어온 아날로그 값을 디지털 값으로 전환하여 `value`라는 변수에 저장해 주었다.

다음으로 LED를 제어하기 위해서 읽어온 `value`에 저장된 값을 OUT핀을 통해서 출력해 주었다. 출력을 위해서는 `bus.write_byte_data` 함수를 사용하였으며 I2C통신을 위해서 슬레이브의 주소값, 출력핀 주소와 출력할 값을 지정해 주었다.

이러한 I2C방식은 하나의 마스터와 여러 개의 슬레이브를 이용하여 구성되기 때문에 마스터의 주도하에 데이터 송수신이 이루어진다. 따라서 마스터에 위와 같이 슬레이브의 주소값이 인가된다. 이러한 I2C방식의 장점은 클럭을 통해서 데이터 전송 타이밍을 제어하기 때문에 시리얼 통신과같이 따로 통신속도를 정하지 않아도 된다는 장점이 있다. 반대로 슬레이브 선택을 위해서 항상 주소 데이터가 불기 때문에 긴 데이터에는 부적합하다는 단점이 있다.

이번에 실습해본 실습과제는 I2C 단독이므로 비교해보기는 어려웠다. 하지만 라즈베리파이를 통해서 I2C통신과 ADC/DAC를 하는 방법을 알 수 있었다. 또한 이렇게 ADC할 때 신호의 정확도에 해당하는 분해능에 대해서도 알아볼 수 있는 기회가 되었다.