

강의명: 임베디드 시스템

숙제 번호: 2

숙제 제목: Analog input and output(아날로그 입출력)

학생 이름: 정우성

학번: 201810890

1. 프로그램 constant-voltage

1.1 프로그램 코드 쓰기

```
#include "mbed.h"

AnalogOut vout(DAC0_OUT); // vout = DAC0_OUT

int main() {
    unsigned short v;
    while(true) {
        v = 0xFFFF * (1.5 / 3.3); // 1.5 V when VDD = 3.3V
        vout.write_u16(v);
        thread_sleep_for(1000);
    }
}
```

1.2 프로그램 작성 아이디어 혹은 이유 설명 쓰기

AnalogOut라는 class에 객체를 DAC0_OUT핀에 연결하여 아날로그 핀을 활용하기 위한 설정을 한다. write_u16를 활용해 아날로그 핀에 전압을 조절한다. 이때 이 함수는 unsigned short 형에 인자를 전달받아 0xFFFF면 3.3V부터 0x0000면 0V로 이 사이 값이면 그 값에 맞는 비율에 전압을 생성한다. 따라서 $0xFFFF * (1.5 / 3.3)$ 를 인자로 전달하면 DAC0_OUT라는 핀에 1.5V를 생성한다. 그리고 thread sleep for함수를 이용해 1초 대기시간을 가진다.

1.3 하드웨어 구성 사진 첨부하기

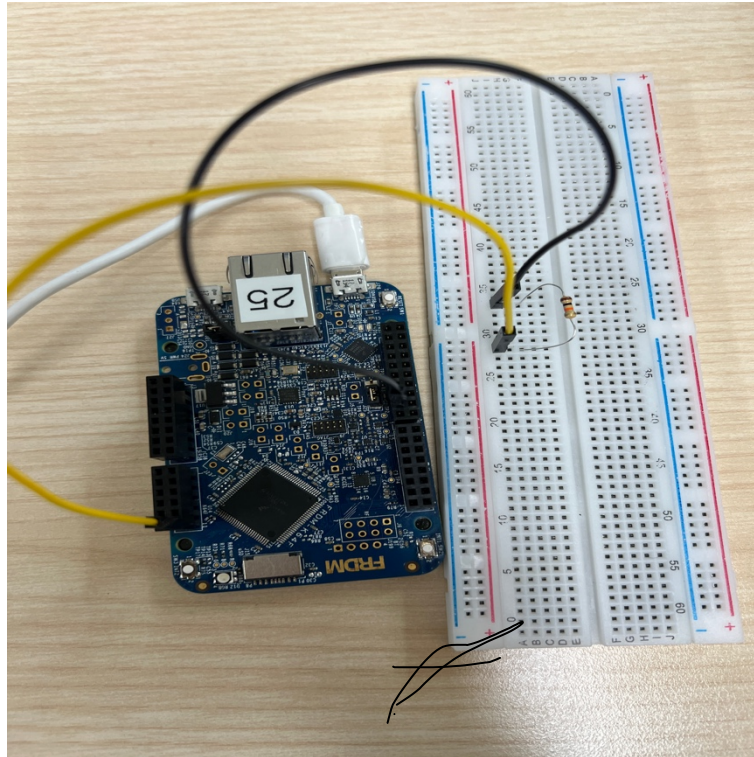


Figure 1 constant-voltage 하드웨어 구성사진

1.4 프로그램 수행 사진/동영상(Youtube 링크) 첨부하기

<https://youtube.com/shorts/WdcjBgvvCIY>

2 프로그램 led-dimmer

2.1 프로그램 코드 쓰기

```
#include "mbed.h"
AnalogOut led(DAC0_OUT);
int main() {
    float f;
    // led = DAC0_OUT
    while(true) {
        for(f = 0.5; f <= 1.0; f = f + 0.01) { // on LED from 50% - 100%
            led.write(f);
            thread_sleep_for(3000/50);
        }
        for(f = 1.0; f >= 0.5; f = f - 0.01) { // on LED from 50% - 100%
            led.write(f);
            thread_sleep_for(3000/50);
        }
    }
}
```

2.2 프로그램 작성 아이디어 혹은 이유 설명 쓰기

DAC0_OUT라는 핀에 아날로그 출력을 사용하기위해 AnalogOut라는 class에 객체를

생성한다. 그 후 write라는 멤버함수는 0에서 1사이의 값을 인자로 받아 0이면 0V를 1이면 3.3V에 값을 이 인자에 비율에 맞게 출력한다. 0부터 0.5사이의 값은 차이가 많이 나지 않으므로 그 위의 값들만 인자로 사용한다.

따라서 이 함수를 for문 안에서 f가 0.5부터 0.01씩 50번 더하면서 1.0까지 이 값을 write에 인자로 전달하면 점점 전압이 커지는 구조를 만들 수 있다. 그리고 이 for문 속 thread_sleep_for 함수에 인자를 (지속시간/반복횟수)에 값을 전달해 전압이 변하는 속도를 조절할 수 있다. 그러므로 이 f를 더하는 반복문, 빼는 반복문 두 파트로 나누어 코드를 구현하면 커지는 전압과 작아지는 전압 신호 2가지 경우를 DAC0_OUT에 생성하는 프로그램이 된다.

2.3 하드웨어 구성 사진 첨부하기

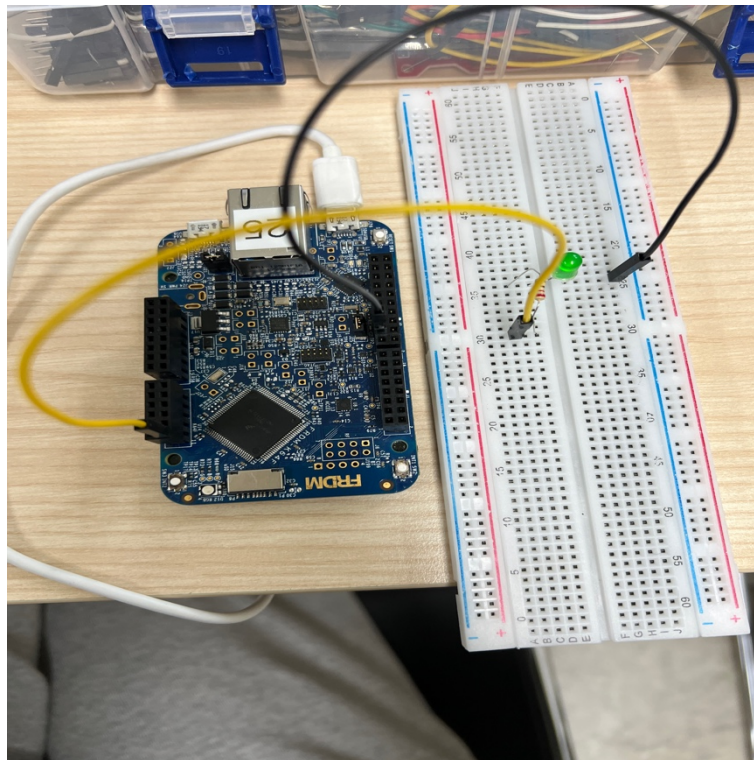


Figure 2 led-dimmer 하드웨어 사진

2.4 프로그램 수행 사진/동영상(Youtube 링크) 첨부하기

<https://youtube.com/shorts/vFzr4CamAuQ>

3. 프로그램 sawtooth-wave

3.1 프로그램 코드 쓰기

```
#include "mbed.h"
AnalogOut vout(DAC0_OUT);
Serial pc(USBTX, USBRX, 115200); // baud rate 115200
int main() {
```

```

float f;
int t = 0, i;

printf("Time, Vout\r\n");
for(f = 0.0; f < 1.1; f = f + 0.1) {
    vout = f;
    for(i = 0; i < 10; i = i + 1)
        printf("%d, %f\r\n", t++, f*3.3);
}
for(f = 0.0; f < 1.1; f = f + 0.1) {
    vout = f;
    for(i = 0; i < 10; i = i + 1)
        printf("%d, %f\r\n", t++, f*3.3);
}
while(true) ;
}

```

3.2 프로그램 작성 아이디어 혹은 이유 설명 쓰기

테라팀에 Vout을 print를 하기위해 SerialPc를 115200으로 선언한다. DAC0_OUT를 아날로그 출력으로 사용하기위해 AnalogOUT이라는 class에 객체를 생성한다.

for문에서 f로 $3.3 \times V$ 에 해당하는 전압을 write함수로 출력하고 그 값과 시간 측에 값 t를 테라팀에 출력한다. 그리고 이 첫번째 for문 반복을 연속적으로 진행하면 연속적으로 sawtooth-wave를 출력하는 프로그램이 된다. 본 코드에서는 2번 반복하고 이것을 테라팀에 출력해 확인하였다.

3.3 하드웨어 구성 사진 첨부하기

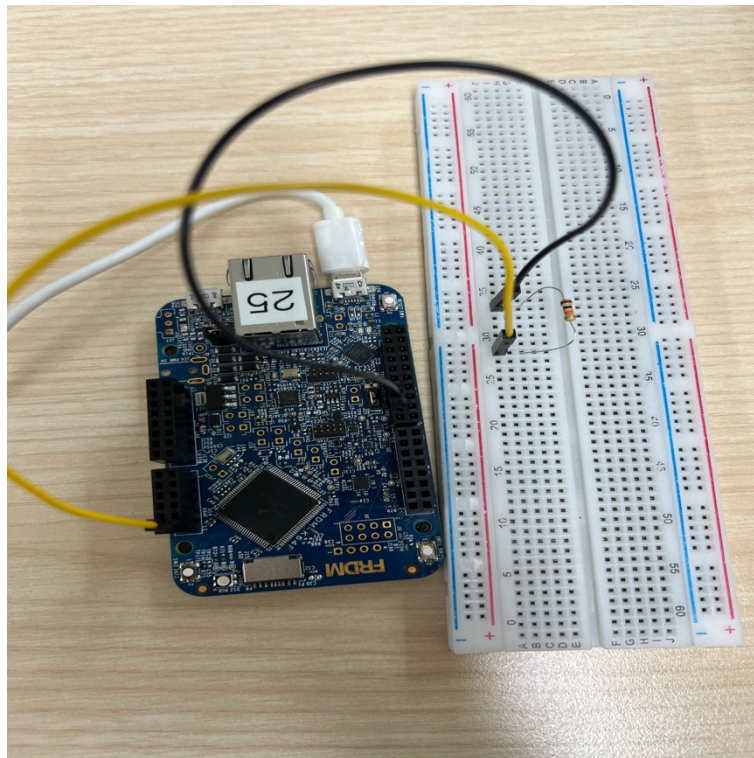


Figure 3 sawtooth-wave 하드웨어 사진

3.4 프로그램 수행 사진/동영상(Youtube 링크) 첨부하기

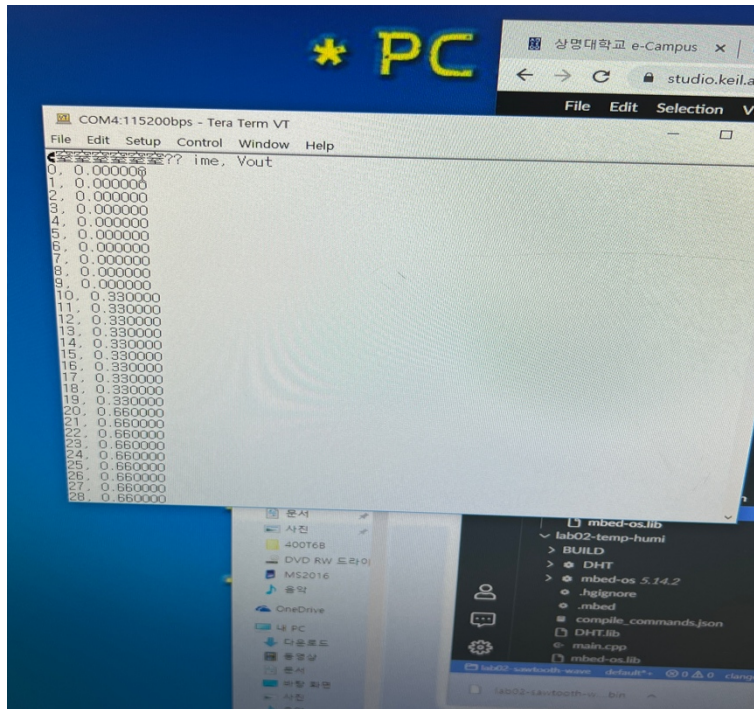


Figure 4 Vout 테라텀 print 사진

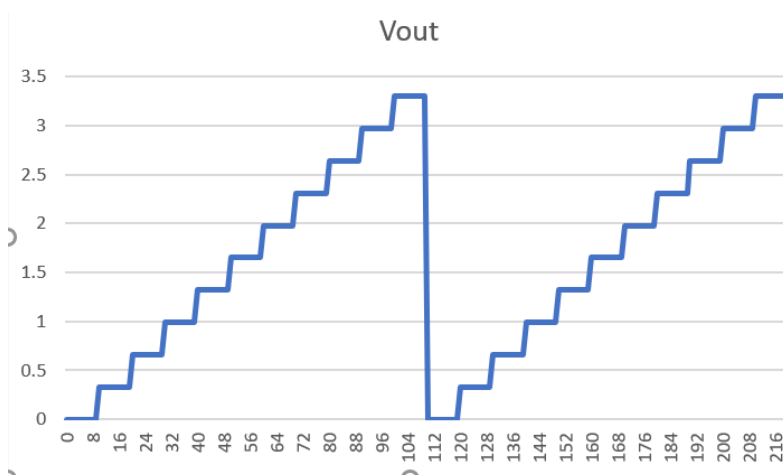


Figure 5 Vout 출력결과 그래프 변환 사진

4 프로그램 temp-humi

4.1 프로그램 코드 쓰기

```
#include "mbed.h"
#include "DHT.h"

Serial pc(USBTX, USBRX, 115200); // baud rate 115200

DHT sensor(A0, SEN11301P); // sensor = SEN11301P
int main() {
```

```

int error;
printf("\r\n===== \r\n");
thread_sleep_for(1000);
while(true) {
error = sensor.readData();
if(error == 0) {
printf("Temperature:\t%4.2f C, %4.2f F, %4.2f K\r\n",
sensor.ReadTemperature(CELCIUS), sensor.ReadTemperature(FARENHEIT),
sensor.ReadTemperature(KELVIN));
printf("Humidity:\t%4.2f%\r\n", sensor.ReadHumidity());
printf("===== \r\n");
thread_sleep_for(5000);
} else { thread_sleep_for(1000);
} }
}

```

4.2 프로그램 작성 아이디어 혹은 이유 설명 쓰기

DHT.h라는 라이브러리에서 제공되는 DHT라는 class를 사용했으며 이 객체는 DHT11이라는 센서에서 온 데이터를 온도, 습도에 데이터를 값으로 반환해주는 멤버함수를 제공한다.

DHT class를 사용하기위해 DHT.h라는 헤더파일을 include하고 DHT라는 객체를 생성한다. 그 후 while문으로 들어가 readData라는 멤버함수를 통해 error상황인지를 판별하고 error상황이 아니면 if문으로 맞으면 else로 진행해 1초 대기한다. 이때 if문으로 잘 넘어오면 ReadTemperature함수 통해 온도, ReadHumidity함수를 통해 습도를 얻어 printf로 출력한다. 그리고 ReadTemperature라는 함수에 전달하는 인자에 따라 섭씨, 화씨, 절대온도를 얻는다. 그리고 5초동안 대기하며 온도와 습도를 반복해서 구하는 프로그램이 완성된다.

4.3 하드웨어 구성 사진 첨부하기

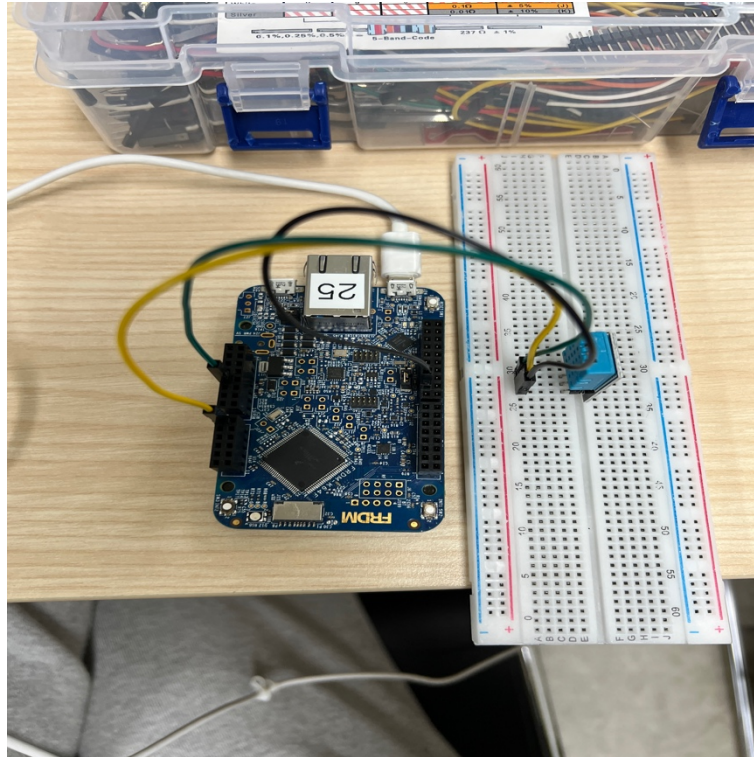


Figure 6 temp-humi 하드웨어 사진

4.4 프로그램 수행 사진/동영상(Youtube 링크) 첨부하기

<https://youtube.com/shorts/G3yhUIde764>

끝.