

# IoT 환경에서 실시간 $V$ , $I$ , $R$ 측정 AC 220V 회로, Wi-Fi 데이터 전송

—

김주호

# 목표

1. AC220V를 사용하는 환경에서의  $V$ ,  $I$ ,  $R$ 을 측정하여 무선(Wi-Fi)으로 정보를 컴퓨터로 송신.
2. 컴퓨터로 송신되는 데이터를 데이터베이스화

# 부품

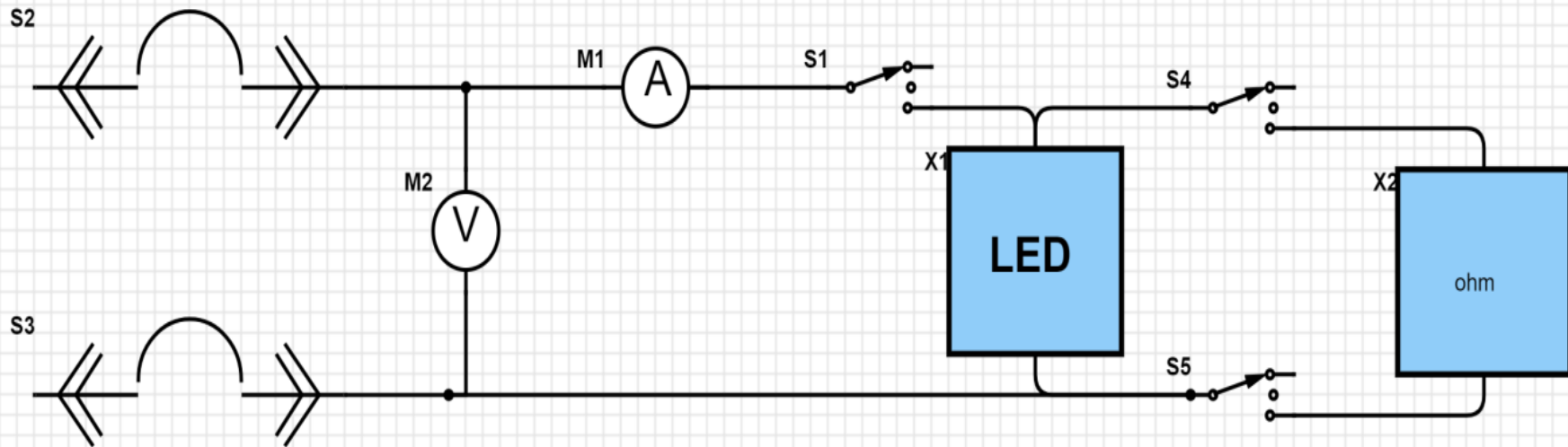
Found around the house!

- 누전차단기
  - AC 전압센서
  - AC 전류센서
  - Relay
  - AC 220V LED전구
  - TI 570 Developer kit
  - ESP8266 Wi-Fi 모듈
-

# BOM

부품명	수량	가격
1 테스트[CH254]소켓 점퍼 케이블 40P (칼라) (F/F)	1	1100
2 테스트[CH254]소켓 점퍼 케이블 40P (칼라) (M/M)	1	1100
3 ACS712 20A 전류 센서 모듈 [SZH-SSBH-094]	1	3520
4 핀헤더소켓 SO127-50핀 (1.27mm)	4	3212
5 핀헤더소켓 SO127-S2-20핀 (1.27mm)	4	1628
6 파워릴레이 JQC-3FF-5VDC-1ZS [SZH-EP058]	6	3300
7 LC 기술 AC 전압 센서 모듈 ZMPT101B 250V 아두이노	1	4220
8 SCT-013 비침습적 AC전류 센서 클램프 계기용 변류기	1	8710
9 와이파이 모듈 ESP8266 아두이노 WIFI ESP-01	1	2360
10 누전차단기 15A	1	8000
11 파워릴레이 JQC-3FF-5VDC-1ZS [SZH-EP058]	6	3300
12 전구소켓	1	1200
13 220V LED 전구	1	
13 택배비	2	2500
14 전원플러그 220V	1	1500
15 TI LaunchPad	1	43000
16 PNP Transistor	1	
현재까지 구매총액		88650

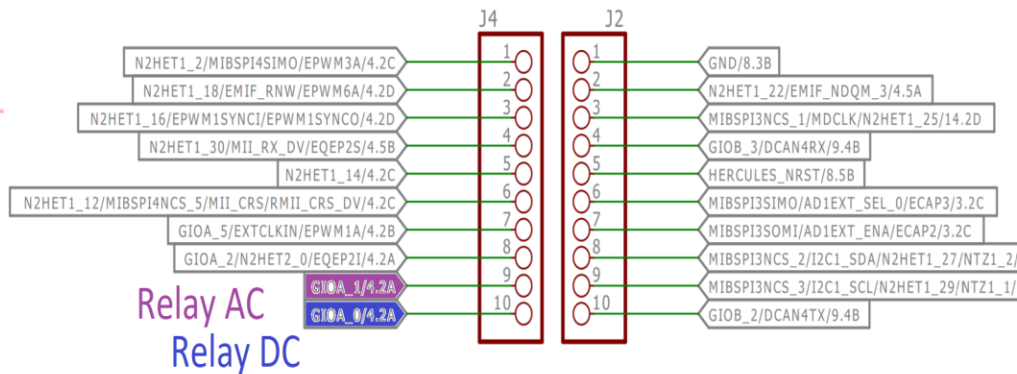
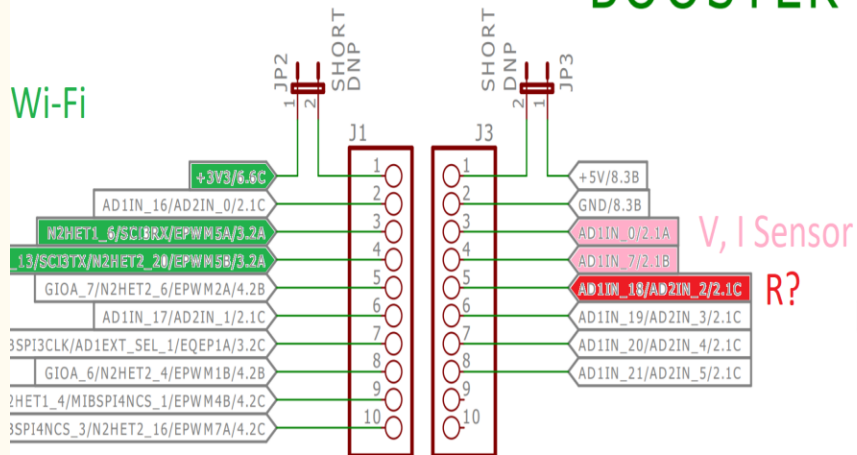
# 회로구성



# 핀 할당

## BOOSTER PACK SITE 1

Wi-Fi



# 소프트웨어 (가)설계

—

1. 센서-ADC
2. 실시간-
3. 통신-SCI
4. 데이터베이스



## 개발할 때 발견된 문제점들 1

# Microcontroller is too fast!

```
Delay(10000); //LED 동작 안 보임
```

```
Delay(33333333); //LED 동작 보임
```

# 센서

## Non invasive(비침습적) AC Current sensor 신뢰성 의심



## 해결책

- 정확한 시간(PWM)으로 릴레이 동작제어를 위해 GPIO에서 HET 포트로 변경(Pin assignment 변경, 20s, Duty 50%)
- Relay 1개당 0.1A 사용: PNP TR설치
- Invasive AC Current Sensor 로 교체



# 개발할 때 발견된 문제점들 2

## 문제점

- MCU에선 C언어와 달리 printf 를 통해 센서에서 들어오는 값을 보는 것이 불가능함.
- SCI(Serial Communication Interface)통신에서 개발에 필요한 부분은 아직 배우지 않았음.

## 해결책

- SCI 통신을 구현하기에 앞서 센서에서 받는 값을 컴퓨터상에서 확인할 다른 방법이 필요함(Local Interconnection Network Interface: LIN?).
- 통신은 앞으로 이루어질 수업에서 가르쳐주실 예정.

### 7.9 Local Interconnect Network Interface (LIN)

The SCI/LIN module can be programmed to work either as an SCI or as a LIN. The core of the module is an SCI. The SCI's hardware features are augmented to achieve LIN compatibility.

The SCI module is a universal asynchronous receiver-transmitter that implements the standard nonreturn to zero format. The SCI can be used to communicate, for example, through an RS-232 port or over a K-line.

The LIN standard is based on the SCI (UART) serial data link format. The communication concept is single-master/multiple-slave with a message identification for multicast transmission between any network nodes.

```

42 void main(void)
43 {
44     /* USER CODE BEGIN (3) */
45
46     /* Enable IRQ */
47     _enable_IRQ();
48
49     /* Initialize SCI module */
50     scInit();
51
52     /* Send user prompt */
53     sciSend(sciInREG, 21);
54
55     /* Await user character */
56     sciReceive(sciInREG);
57
58     /* Infinite loop */
59     while(1);
60
61     /* USER CODE END */
62 }
63
64
65 /* USER CODE BEGIN (4) */
66
67 void sciNotification(sciBASE_t)
68 {
69     /* Echo received character */
70     sciSend(sci, 1, (unsigned char)sciReceive(sci));
71
72     /* Await further character */
73     sciReceive(sci, 1, (unsigned char)sciReceive(sci));
74 }
75
76 /* ESM interrupt notification */
77 void esmGroup1Notification(int)
78 {
79     return;
80 }
81
82 void esmGroup2Notification(int)
83 {
84     return;
85 }

```

Terminal Settings

View Settings

View Title: Terminal

Connection Type: Serial

Settings:

Port: COM8

Baud Rate: 9600

Data Bits: 8

Stop Bits: 2

Parity: None

Flow Control: None

Timeout (sec): 5

OK Cancel

 Console 

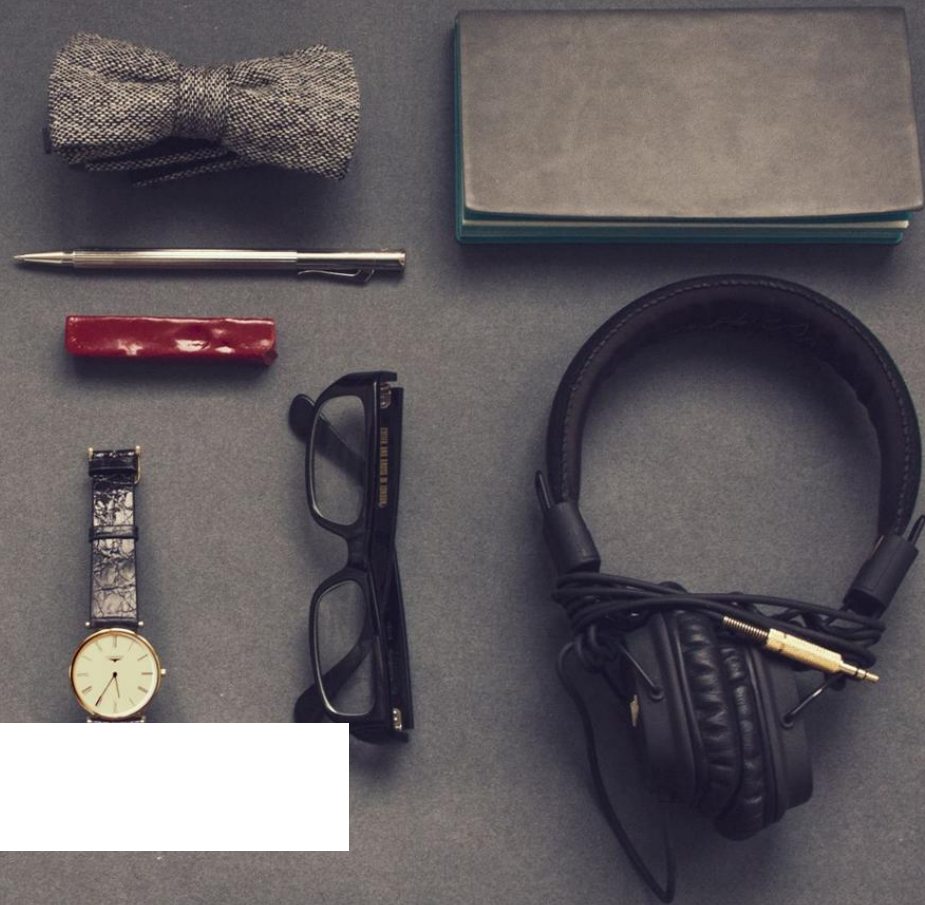
No consoles to display at this time.

## Problems

No Connection Selected



The experiment



# Conclusion

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip.