

Chap07. 프로젝트

Contents

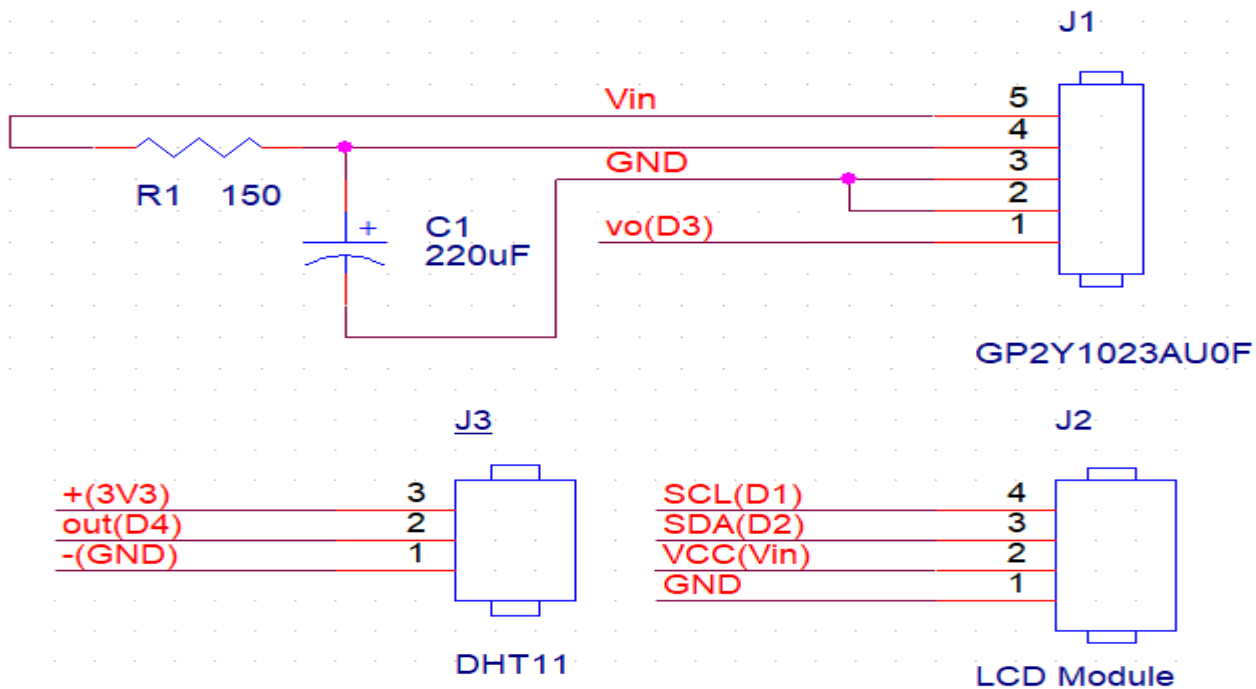
7.1 실내 환경 모니터링시스템

7.2 침입감지시스템

7.3 조명감지시스템

7.1 실내환경 모니터링 시스템

◆ 회로도



7.1 실내환경 모니터링 시스템

◆ Blynk widget 설정

- Value Display(2ea)
 - V0(state), V1(ugm3)
- Level V(3ea)
 - V1(ugm3:300), V2(Humi:100), V3(Temp:40)

7.1 실내환경 모니터링 시스템

◆ Blynk widget 설정 및 결과 확인



7.1 실내환경 모니터링 시스템

◆ 소스 코드

```
2 // 실내환경 모니터링시스템 -> monitoring_vpin(NodeMCU -> 스마트폰)
3
4 #include <ESP8266WiFi.h>
5 #include <BlynkSimpleEsp8266.h>
6 #include <DHT.h>
7 #include <LiquidCrystal_I2C.h>
8
9 #define BLYNK_PRINT Serial
10 #define DHTTYPE DHT11 // DHT 11
11
12 char auth[] = "KWiEYYjACwL_4t3z-5wNjHSxw3Uftqy4";
13 char ssid[] = "sjpark";
14 char pass[] = "12345678";
15
16 int dust_pin=D3;
17 int dht_pin=D4;
18
19 float h = 0.0;
20 float t = 0.0;
21 float FineDust=0.0;
22
23 DHT dht(dht_pin, DHTTYPE);
24 LiquidCrystal_I2C lcd(0x27, 16, 2);
```

7.1 실내환경 모니터링 시스템

◆ 소스 코드

```
26 void setup()
27 {
28     lcd.begin(16,2);
29     lcd.init();
30     lcd.backlight();
31     Serial.begin(9600);
32     Blynk.begin(auth, ssid, pass);
33     dht.begin();
34     pinMode(dust_pin, INPUT);
35 }
36
37 void refresh_DHT()
38 {
39     h = dht.readHumidity();
40     t = dht.readTemperature();
41 }
42
43 void refresh_dust()
44 {
45     unsigned long pulse = pulseIn(dust_pin, LOW, 20000);
46     FineDust = (pulse-1400)/14.0;
47     if(FineDust > 300) FineDust = 0;
48 }
```

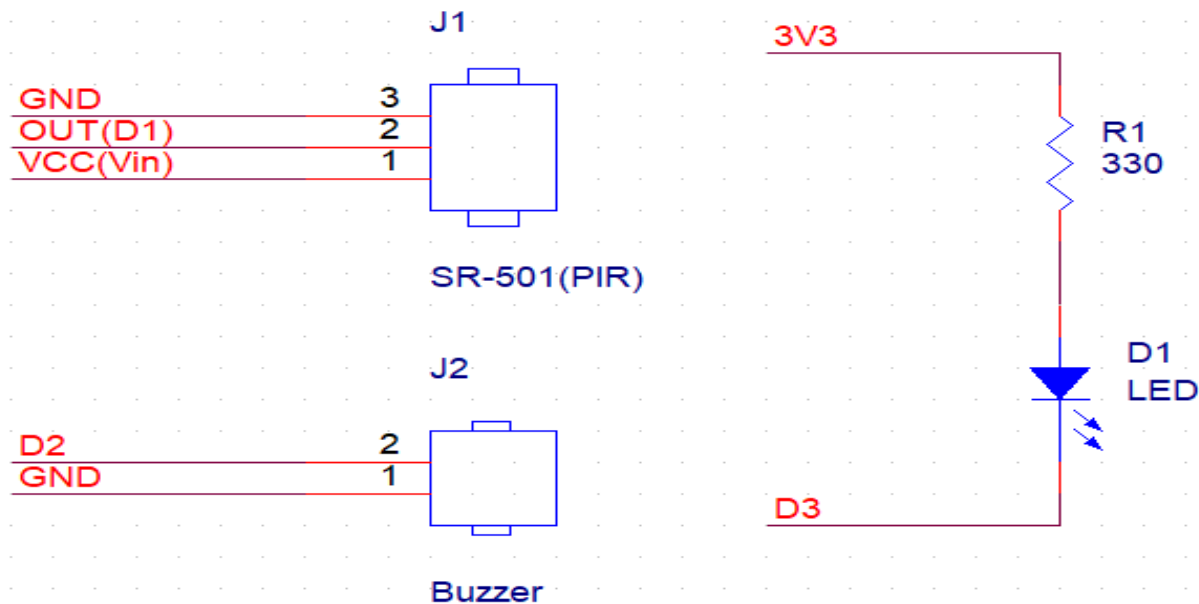
7.1 실내환경 모니터링 시스템

◆ 소스 코드

```
50 void loop()
51 {
52     refresh_dust();
53     refresh_DHT();
54     Blynk.run();
55     lcd.clear();
56
57     lcd.setCursor(0,0);
58     if(FineDust<=1)
59     {
60         Blynk.virtualWrite(V0,"좋음");
61         lcd.print("good");
62     }
63     else if(FineDust<=100)
64     {
65         Blynk.virtualWrite(V0,"보통");
66         lcd.print("soso");
67     }
68     else
69     {
70         Blynk.virtualWrite(V0,"나쁨");
71         lcd.print("bad");
72
73
74         lcd.setCursor(7,0);
75         lcd.print((int)FineDust);
76         lcd.print("ugm3");
77         Blynk.virtualWrite(V1,FineDust);
78
79         lcd.setCursor(0,1);
80         lcd.print((int)h);
81         lcd.print("%");
82         Blynk.virtualWrite(V2,h);
83
84         lcd.setCursor(8,1);
85         lcd.print((int)t);
86
87         lcd.print((char)223);
88         lcd.print("C");
89         Blynk.virtualWrite(V3,t);
90
91         delay(3000);
92 }
```


7.2 침입감지시스템

◆ 회로도



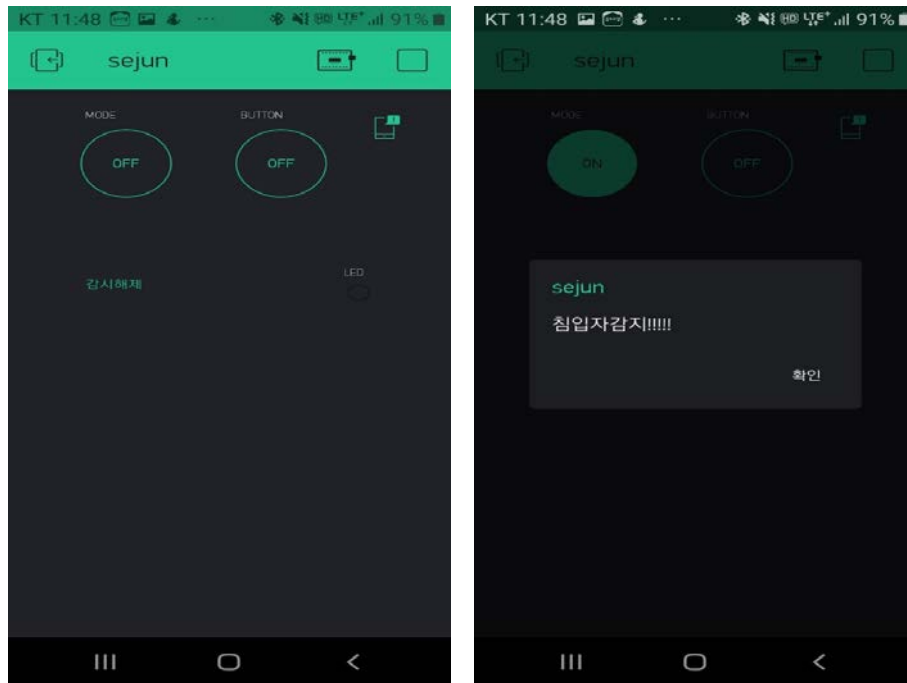
7.2 침입감지시스템

◆ Blynk widget 설정

- Button(2ea)
 - V5(Switch)-mode, V6(Push)-all off
- LED(1ea) – V1
- Notification(1ea) – Not setting
- Value Display(1ea) – V0

7.2 침입감지시스템

◆ Blynk widget 설정 및 결과 확인



7.2 침입감지시스템

◆ 소스 코드

```
2 // 침입감지시스템 -> detection_vpin(NodeMCU <-> 스마트폰)
3
4 #define BLYNK_PRINT Serial
5 #include <ESP8266WiFi.h>
6 #include <BlynkSimpleEsp8266.h>
7
8 char auth[] = "KWiEYYjACwL_4t3z-5wNjHSxw3Uftqy4";
9 char ssid[] = "sjpark";
10 char pass[] = "12345678";
11
12 int LED_PIN = D3;
13 int BUZZER_PIN = D2;
14 int PIR_PIN = D1;
15
16 bool mode = false;
17 bool alert = false;
18
19 WidgetLED v_led(V1);
20
21 // 상태label=V0, led=V1, 모드=V5, 경보해제=V6
```

7.2 침입감지시스템

◆ 소스 코드

```
23 void setup() {
24   Serial.begin(9600);
25   Blynk.begin(auth, ssid, pass);
26   pinMode(PIR_PIN, INPUT);
27   pinMode(BUZZER_PIN, OUTPUT);
28   pinMode(LED_PIN, OUTPUT);
29 }
30
31 BLYNK_WRITE(V5)
32 {
33   mode=param.asInt();
34   if(mode==true)Blynk.virtualWrite(V0,"감시중");
35   else
36   {
37     Blynk.virtualWrite(V0,"감시해제");
38     alert=false;
39     digitalWrite(LED_PIN, LOW);
40     v_led.off();
41   }
42 }
```

```
44 BLYNK_WRITE(V6)
45 {
46   if(param.asInt()==1 && alert==true)
47   {
48     Blynk.virtualWrite(V0,"경보해제");
49     alert=false;
50     v_led.off();
51     digitalWrite(LED_PIN, LOW);
52     noTone(BUZZER_PIN);
53   }
54 }
```

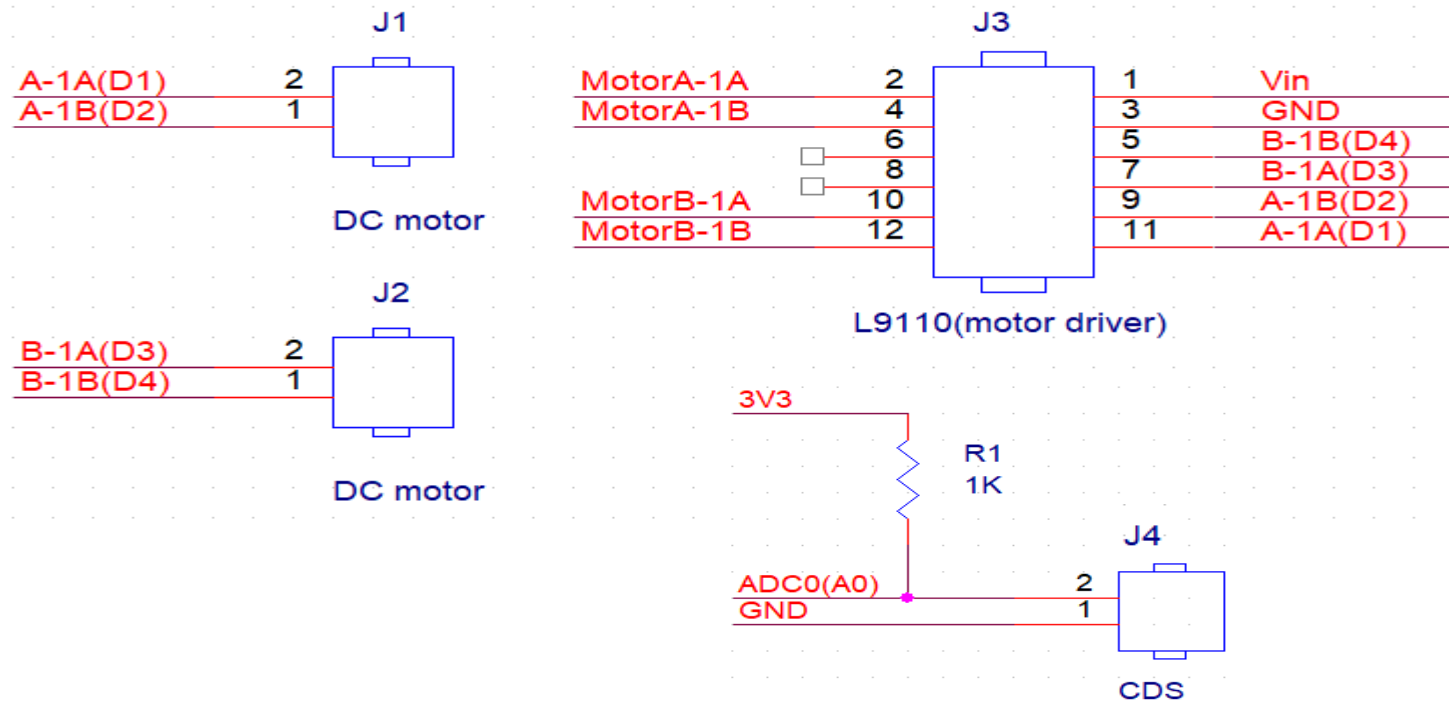
7.2 침입감지시스템

◆ 소스 코드

```
56 void loop()
57 {
58   Blynk.run();
59   if(mode==true&&alert==false)
60   {
61     if(digitalRead(PIR_PIN)==true)
62     {
63       Blynk.virtualWrite(V0, "침입자 감지됨");
64       Blynk.notify("침입자감지!!!!!!");
65       alert=true;
66       v_led.on();
67     }
68   }
70   if(alert==true)
71   {
72     for(int freq = 150; freq <=1800; freq = freq + 2)
73     {
74       tone(BUZZER_PIN, freq, 10);
75       delay(1);
76     }
77     digitalWrite(LED_PIN, !digitalRead(LED_PIN));
78     for(int freq = 1800; freq >=150; freq = freq - 2)
79     {
80       tone(BUZZER_PIN, freq, 10);
81       delay(1);
82     }
83     digitalWrite(LED_PIN, !digitalRead(LED_PIN));
84   }
85 }
```

7.3 조명감지시스템

◆ 회로도



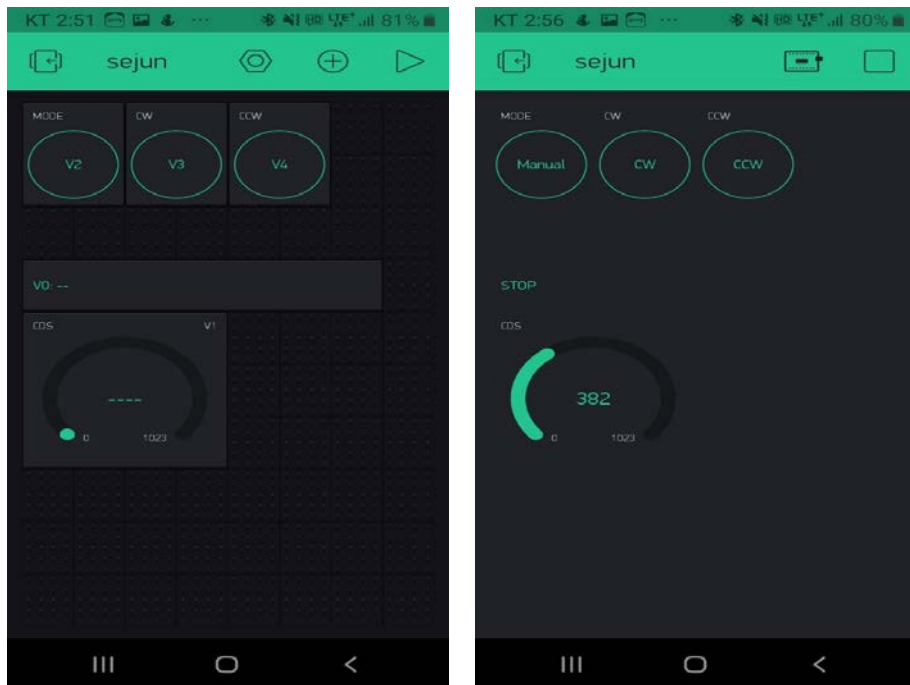
7.3 조명감지시스템

◆ Blynk widget 설정

- Button(3ea)
 - V2(Switch)-mode, off(Manual), on(Auto)
 - V3(Push)-CW, off(CW), on(CW)
 - V4(Push)-CCW, off(CCW), on(CCW)
- Value Display(1ea) – V0
- Gauge(1ea) – V1(Cds)

7.3 조명감지시스템

◆ Blynk widget 설정 및 결과 확인



7.3 조명감지시스템

◆ 소스 코드

```
2 // 조명감지시스템 -> light_detection_vpin(NodeMCU <-> 스마트폰)
3
4 #include <ESP8266WiFi.h>
5 #include <BlynkSimpleEsp8266.h>
6 #define BLYNK_PRINT Serial
7
8 char auth[] = "KWiEYYjACwL_4t3z-5wNjHSxw3Uftqy4";
9 char ssid[] = "sjpark";
10 char pass[] = "12345678";
11
12 int motor_a=D1;
13 int motor_b=D2;
14 int cds=A0;
15
16 bool mode=false;
17 int motor_status=0;
18
19 void setup()
20 {
21     Serial.begin(9600);
22     Blynk.begin(auth, ssid, pass);
23     pinMode(motor_a,OUTPUT);
24     pinMode(motor_b,OUTPUT);
25     pinMode(cds,INPUT);
26 }
```

7.3 조명감지시스템

◆ 소스 코드

```
28 BLYNK_WRITE (V2)
29 {
30     mode=param.asInt();
31 }
32 BLYNK_WRITE (V3)
33 {
34     if(mode==false && param.asInt()==1) cw();
35 }
36 BLYNK_WRITE (V4)
37 {
38     if(mode==false && param.asInt()==1) ccw();
39 }
40 }
```

```
41 void cw()
42 {
43     Blynk.virtualWrite (V0, "CW");
44     digitalWrite (motor_a, HIGH);
45     digitalWrite (motor_b, LOW);
46     delay (1000);
47     digitalWrite (motor_a, LOW);
48     digitalWrite (motor_b, LOW);
49     delay (1);
50 }
51 void ccw()
52 {
53     Blynk.virtualWrite (V0, "CCW");
54     digitalWrite (motor_a, LOW);
55     digitalWrite (motor_b, HIGH);
56     delay (1000);
57     digitalWrite (motor_a, LOW);
58     digitalWrite (motor_b, LOW);
59     delay (1);
60 }
61 }
```

7.3 조명감지시스템

◆ 소스 코드

```
63 void loop()
64 {
65     Blynk.run();
66     int cds_val = analogRead(cds);
67     Blynk.virtualWrite(V1, cds_val);
68     if(mode)
69     {
70         if(cds_val>300 && motor_status!=1)
71         {
72             cw();
73             motor_status=1;
74         }
75         else if(cds_val<300 && motor_status!=-1)
76         {
77             ccw();
78             motor_status=-1;
79             Blynk.virtualWrite(V0, "CCW");
80         }
81         else
82         {
83             Blynk.virtualWrite(V0, "STOP");
84         }
85     }
86     else
87     {
88         motor_status=0;
89         Blynk.virtualWrite(V0, "STOP");
90     }
91     delay(100);
92 }
```