Blinker APP:

添加设备：进入app主界面，点击右上角加号-->独立设备-->wifi接入-->阿里云（可以尝试其他选项）-->记录下Secret Key-->返回我的设备

(尽量保持 APP主界面中“所有设备“下只有一个设备)

米家 APP:

我的-->其他平台设备-->点击添加-->找到“点灯科技“-->绑定账号-->输入 Blinker 账号密码，点击绑定-->点击同步设备（显示”同步成功“，并且所显示设备的名称和Blinker APP上的设备名称一致，说明设备绑定成功）

小爱同学 APP:

利用米家账号登陆。

苹果手机控制方法：进入主页，点击下方圆形圈圈，显示“倾听中“，即可说出”开灯“或者”关灯“。

安卓手机控制方法：1.和苹果手机采用同样的方法进行控制；2.点击右上角个人中心，在“小爱设置“中点击”语音唤醒“，将语音唤醒打开（根据需要选择APP内唤醒、后台唤醒）；说出”小爱同学“，屏幕上显示”倾听中“时，即可说”开灯“或者“关灯”来控制灯了。

硬件连接：

LED接法与实验2方法一致

PART A:

功能实现：将以下程序编译、下载（将之前记录下的Secret Key、即将要连接的WiFi名称和密码写到对应的位置），复位后打开串口监视器，看到MQTT connect时，说明板子联网成功；根据不同的手机，利用上述方法通过“小爱同学”来控制LED的开和关，当控制LED开灯时，串口监视器上会显示"GEN the 'ON' message of led"，同样的方法也可以看到关灯的信息。PART A部分功能验证完毕。

#define BLINKER\_WIFI

#define BLINKER\_MIOT\_LIGHT

#include <Blinker.h>

const char auth[] = "your own Secret Key";

const char ssid[] = "your wifi name ";

const char pswd[] = "your wifi password";

void miotPowerState(const String & state)

{

if (state == BLINKER\_CMD\_ON) {

Serial.println("GEN the 'ON' message of led");

}

else if (state == BLINKER\_CMD\_OFF) {

Serial.println("GET the 'OFF' message of led");

}

}

void setup() {

Serial.begin(115200);

BLINKER\_DEBUG.stream(Serial);

Blinker.begin(auth, ssid, pswd);

BlinkerMIOT.attachPowerState(miotPowerState);

}

void loop() {

Blinker.run();

}

程序分析：函数BlinkerMIOT.attachPowerState(miotPowerState)的功能为，注册回调函数miotPowerState()，当使用“小爱同学“APP来控制LED时，就会进入到该回调函数中；变量state代表了语音控制的内容，当说出“开灯”时，state的值为BLINKER\_CMD\_ON，于是输出打印信息“GEN the 'ON' message of led”；否则输出打印信息“GEN the 'OFF' message of led”。

PART B:

功能实现：将以下程序编译、下载后，按下复位键，看到板载的LED灯按照程序中的时间间隔交替闪烁，PART A部分功能验证成功

const int myLed = LED\_BUILTIN;

void setup() {

pinMode(myLed, OUTPUT);

}

void loop() {

digitalWrite(myLed, LOW);

delay(1000);

digitalWrite(myLed, HIGH);

delay(2000);

}

程序分析：使用pinMode()函数将myLed引脚设置为输出，使用digitalWrite()将该引脚设置为高电平或者低电平，从而实现控制led亮和灭的效果。

PART C：

功能实现：将以下代码编译、下载（将之前记录下的Secret Key、即将要连接的WiFi名称和密码写到对应的位置）；打开串口监视器，按下复位键，当看到“MQTT Connected!”，说明板子联网成功；按照上述方法，使用”小爱同学“APP，通过语音”开灯“和”关灯“来控制LED的亮和灭。实验三功能验证完毕。

#define BLINKER\_WIFI

#define BLINKER\_MIOT\_LIGHT

#include <Blinker.h>

const char auth[] = "your own Secret Key";

const char ssid[] = "your wifi name ";

const char pswd[] = "your wifi password";

const int myLed = LED\_BUILTIN;

void miotPowerState(const String & state)

{

if (state == BLINKER\_CMD\_ON) {

digitalWrite(myLed, LOW);

BlinkerMIOT.powerState("on");

BlinkerMIOT.print();

}

else if (state == BLINKER\_CMD\_OFF) {

digitalWrite(myLed, HIGH);

BlinkerMIOT.powerState("off");

BlinkerMIOT.print();

}

}

void setup() {

Serial.begin(115200);

BLINKER\_DEBUG.stream(Serial);

pinMode(myLed, OUTPUT);

digitalWrite(myLed, HIGH);

Blinker.begin(auth, ssid, pswd);

BlinkerMIOT.attachPowerState(miotPowerState);

}

void loop() {

Blinker.run();

}

程序分析：该部分为PART A与PART B相结合，当用“小爱同学”APP语音控制LED状态时，就会进入到回调函数miotPowerState()中，根据变量state的值来打开或者关闭LED；同时调用BlinkerMIOT.powerState("on")函数将LED的状态回馈给平台。