Mbed Lab 9 Report
Serial RPC

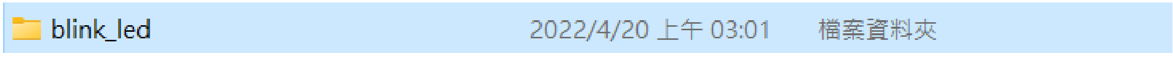
109033130 唐振家

一、Lab Description

1、eRPC Over Serial

說明：

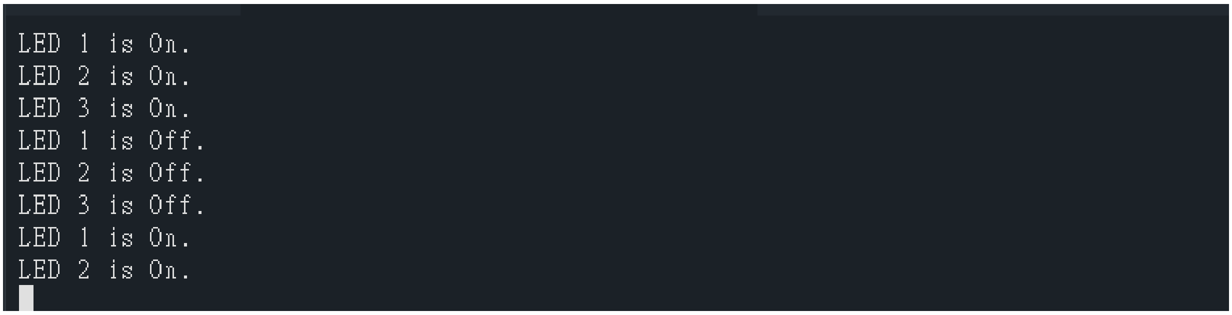
將 erpcgen下載下來，並將 blink_led 檔放入 9_1 資料集夾中。



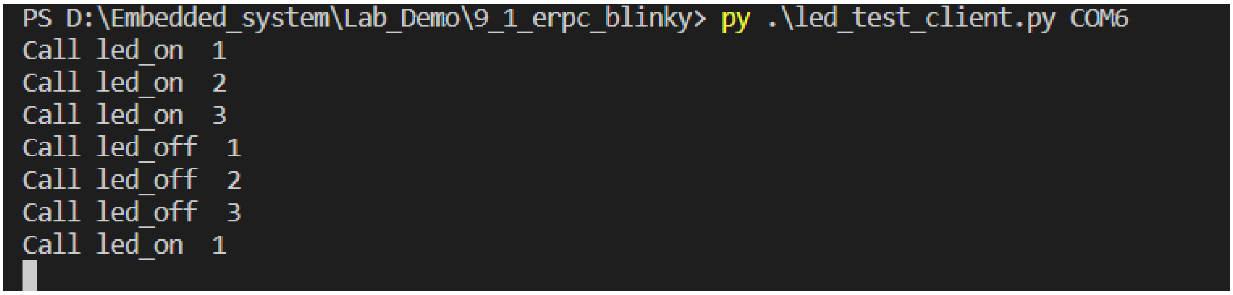
blink_led　2022/4/20 上午 03:01　檔案資料夾

說明：

這次實驗主要是利用 eRPC，達到能夠同時開啟2個terminal，並顯示資訊。



```
LED 1 is On.
LED 2 is On.
LED 3 is On.
LED 1 is Off.
LED 2 is Off.
LED 3 is Off.
LED 1 is On.
LED 2 is On.
```



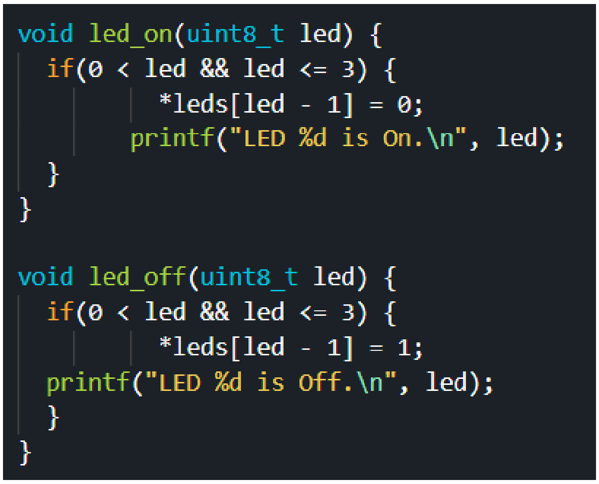
```
PS D:\Embedded_system\Lab_Demo\9_1_erpc_blinky> py .\led_test_client.py COM6
Call led_on  1
Call led_on  2
Call led_on  3
Call led_off  1
Call led_off  2
Call led_off  3
Call led_on  1
```

一、Lab Description

1、eRPC Over Serial

說明：

在mbad 的 server端，先將LED燈的物件進行設置，並進行erpc函式的宣告(void led_on、void led_off)，期將會列印出LED燈的狀態(LED 1/2/3 is Off/On.)，之後以UART作為通訊協定，用ep::UART Transport物件約定鮑率(9600)，在main中設計初始設定，把LEDBlinkService_service物件加入server中，完成以後開始執行，等待接收到client的指令。其中void led_on、void led_off為remote functin。

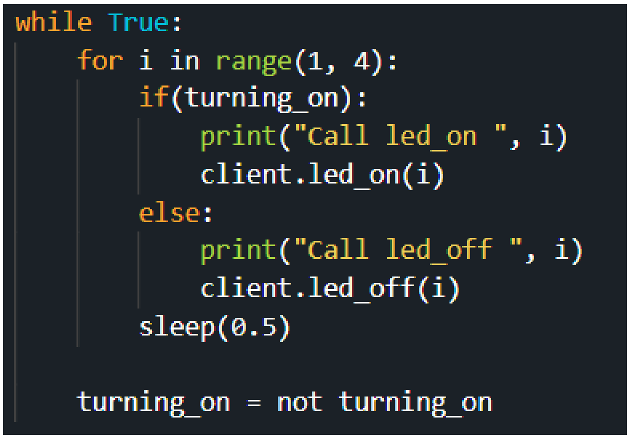


```
void led_on(uint8_t led) {
  if(0 < led && led <= 3) {
          *leds[led - 1] = 0;
        printf("LED %d is On.\n", led);
  }
}

void led_off(uint8_t led) {
  if(0 < led && led <= 3) {
          *leds[led - 1] = 1;
  printf("LED %d is Off.\n", led);
  }
}
```

說明：

在 eRPC 的 client 端，同樣利用UART作為通訊協定，約定鮑率(9600)，寫一個while true迴圈，進行程序並且直到斷定停止，在裡面加上 for 迴圈，進行LED燈的循環亮暗及選擇，同時印出Call led_on/off，顯示現在選擇到哪個LED燈，以及狀態，當client.led_on(i)被執行時，呼叫 server 的 void led_on。



```
while True:
    for i in range(1, 4):
        if(turning_on):
            print("Call led_on ", i)
            client.led_on(i)
        else:
            print("Call led_off ", i)
            client.led_off(i)
        sleep(0.5)

    turning_on = not turning_on
```

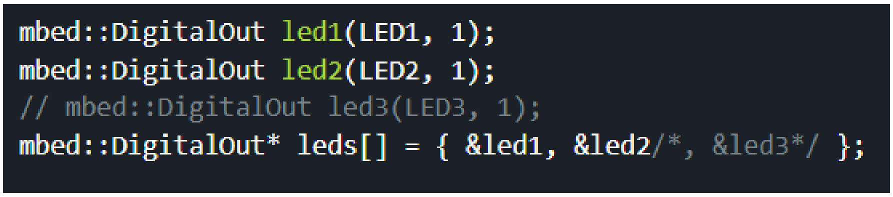
二、Demo and Checkpoints

eRPC function that blink two LEDs in sequence

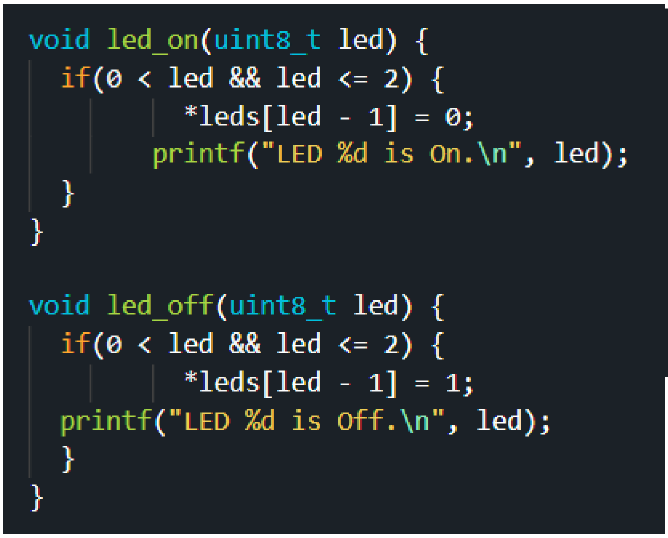
說明：

此次Demo需要修改Code，使得只有2顆LED燈在明暗交替。

因為LED3、LED4是共用迴路的關係，所以將LED3關掉，即可達到只亮暗LED1、LED2的效果，並且修改程式中的LED燈數量，編譯並執行。

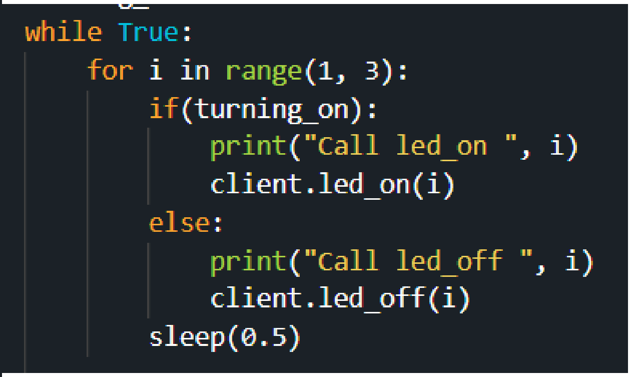


```
mbed::DigitalOut led1(LED1, 1);
mbed::DigitalOut led2(LED2, 1);
// mbed::DigitalOut led3(LED3, 1);
mbed::DigitalOut* leds[] = { &led1, &led2/*, &led3*/ };
```

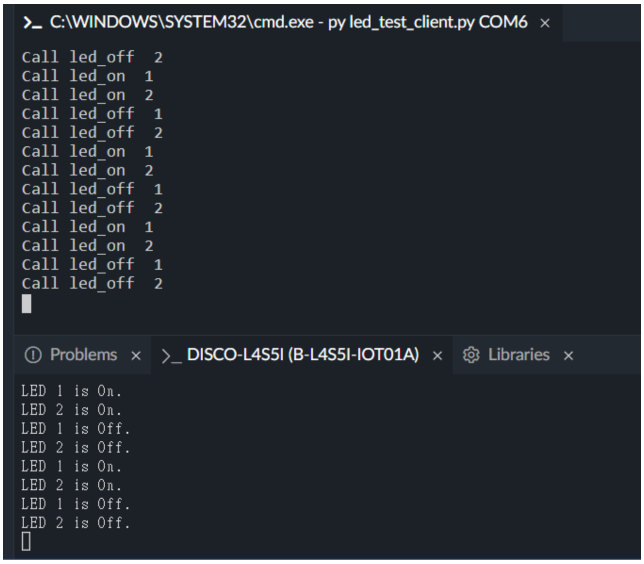


```
void led_on(uint8_t led) {
  if(0 < led && led <= 2) {
        *leds[led - 1] = 0;
       printf("LED %d is On.\n", led);
  }
}

void led_off(uint8_t led) {
  if(0 < led && led <= 2) {
        *leds[led - 1] = 1;
  printf("LED %d is Off.\n", led);
  }
}
```



```
while True:
    for i in range(1, 3):
        if(turning_on):
            print("Call led_on ", i)
            client.led_on(i)
        else:
            print("Call led_off ", i)
            client.led_off(i)
        sleep(0.5)
```



C:\WINDOWS\SYSTEM32\cmd.exe - py led_test_client.py COM6
Problems
DISCO-L4S5I (B-L4S5I-IOT01A)
Libraries

```
Call led_off  2
Call led_on  1
Call led_on  2
Call led_off  1
Call led_off  2
Call led_on  1
Call led_on  2
Call led_off  1
Call led_off  2
Call led_on  1
Call led_on  2
Call led_off  1
Call led_off  2
```

```
LED 1 is On.
LED 2 is On.
LED 1 is Off.
LED 2 is Off.
LED 1 is On.
LED 2 is On.
LED 1 is Off.
LED 2 is Off.
```

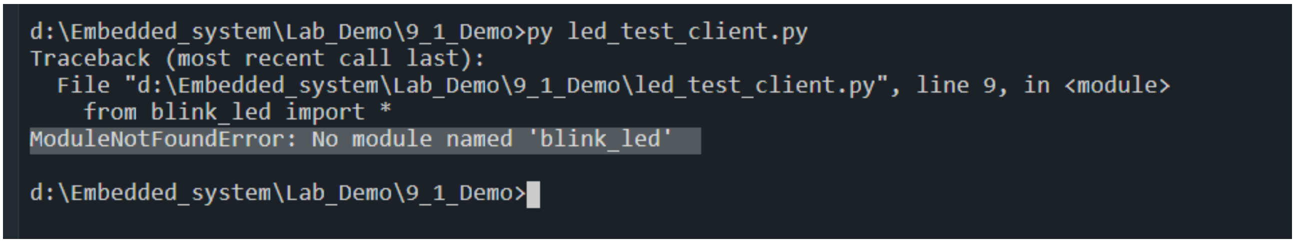
三、遇到的問題

找不到blink_led檔

後來發現是我未將檔案放入資料夾，且指令不能只輸入

py led_test_client.py，而須改成

python3 led_test_client.py



```
d:\Embedded_system\Lab_Demo\9_1_Demo>py led_test_client.py
Traceback (most recent call last):
  File "d:\Embedded_system\Lab_Demo\9_1_Demo\led_test_client.py", line 9, in <module>
    from blink_led import *
ModuleNotFoundError: No module named 'blink_led'

d:\Embedded_system\Lab_Demo\9_1_Demo>
```

四、討論

結論

本次實驗教我如何實做使用eRPC server，將端口屏蔽，並且創建另一個Serial(透過UART通訊協定)，達成開啟兩個以上的Terminal的方法，購過乎將