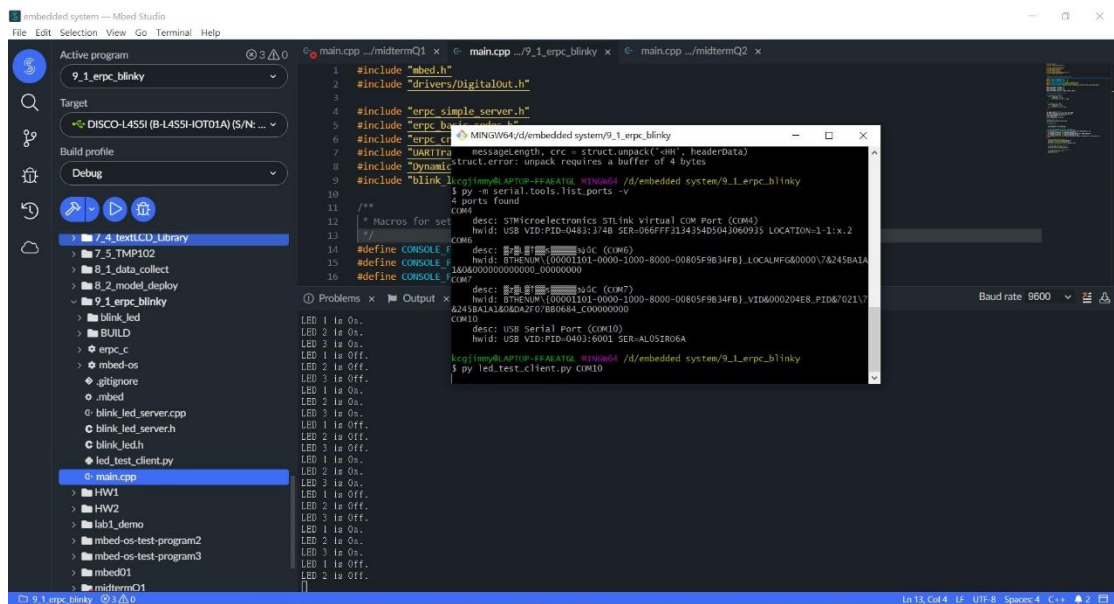


In this lab, we learn how to use serial RPC

## (1) Lab9



```
Problems x Output x >_ DISCO-L455I (B-L455I-IOT01A) x Baud rate 9600 v
Initializing server.
Adding LED server.
Running server.
[]
```



The screenshot shows the Mbed Studio IDE with the following components:

- Left Panel:** Project tree showing the structure of the '9\_1\_erc\_blinky' project, including folders like '7\_4\_textLCD\_Library', '7\_5\_TMP102', '8\_1\_data\_collect', '8\_2\_model\_deploy', '9\_1\_erc\_blinky', and files like 'blink\_led', 'BUILD', 'erc\_c', 'mbed-os', 'gltignore', 'mbed', 'blink\_led\_server.cpp', 'blink\_led\_server.h', 'blink\_led.h', 'led\_test\_client.py', and 'main.cpp'.
- Center Panel:** The 'main.cpp' file is open, showing the following code:

```
1 #include "mbed.h"
2 #include "drivers/DigitalOut.h"
3
4 #include "erc_simple_server.h"
5 #include "erc_blinky_server.h"
6 #include "erc_c"
7 #include "UARTIra"
8 #include "Dynamic"
9 #include "Blink_led_test_client.py"
10
11 /**
12  * Macros for set
13  */
14 #define CONSOLE_F
15 #define CONSOLE_F
16 #define CONSOLE_F
```
- Right Panel:** The 'Output' window shows the serial output of the program, displaying the status of the LED server and the client.

In this lab, we try to use eRPC. Following the steps and we can briefly see how it works.

## (2) Discussion

I think this lab is quite different than I had learned from microprocessor before, in this lab we learn some concepts of network, though we can control mbed through USB, but with eRPC we have the concept of host and clients, by calling function on host we can control the clients. I think in latter labs we will learn more concept with network, and it's getting much harder...