作业8 Qt MQTT开发库移植与测试

布置时间：5月26日

收集时间：6月1日24：00

提交方式：姓名\_学号\_作业8，如：平毅\_2009060707\_作业8.doc

内容：

1. 在Deepin操作系统（可在Windows版本上先进行开发），利用GitHub Desktop下载Qt MQTT官方库文件包，根据Qt安装版本（课堂中演示的版本位5.12.9）选择对应的MQTT库文件包版本两者需对应。

2. 打开文件包中的qtmqtt.pro工程文件，分别执行build和run qmake（汉化版本名称分别位编译和运行），生成编译输出文件夹，本例中位：build-qtmqtt-Desktop\_Qt\_5\_12\_9\_MinGW\_64\_bit-Debug。

3. 分别将该文件夹中的lib文件夹内6个文件，bin文件夹内2个文件，mkspecs/modules内1个文件复制至C:\Qt\Qt5.12.9\5.12.9\mingw73\_64下对应文件夹（注意实际操作可能存在不同安装路径）。把示例代码（C:\Users\pyhom\Documents\GitHub\qtmqtt\examples）mqtt整个目录copy至QT安装目录（C:\Qt\Qt5.12.9\Examples\Qt-5.12.9）下。

**针对5.12.9的前期设置已完成，紧接着完成代码测试，新建MQTT\_Test工程。**

1.需要在pro文件中添加 QT +=mqtt(注意emqx/qmqtt版本相同，该版本需添加QT +=qtmqtt)

2 进行一次编译无报错后继续添加库文件。

3. 依据5.12.9示例程序，需在mainWindow.h中添加#include <QMqttClient> (课堂上也举例了emqx/qmqtt版本示例，为添加#include “mqtt/qmqtt.h”,#include "mqtt/qmqtt\_client.h"，编译过程中需调整库文件符号)

4. 在private中声明mqttclient对象，5.12.9的构造方法为：QMqttClient \*myMqttclient;（emqx/qmqtt版本，为 QMQTT::Client \*myMqttclient;）

5. 完成编译，如无错则编译成功，可进行MQTT工程的二次开发。

6. 可直接运行GitHub库文件夹中simpleclient示例，完成MQTT功能测试。

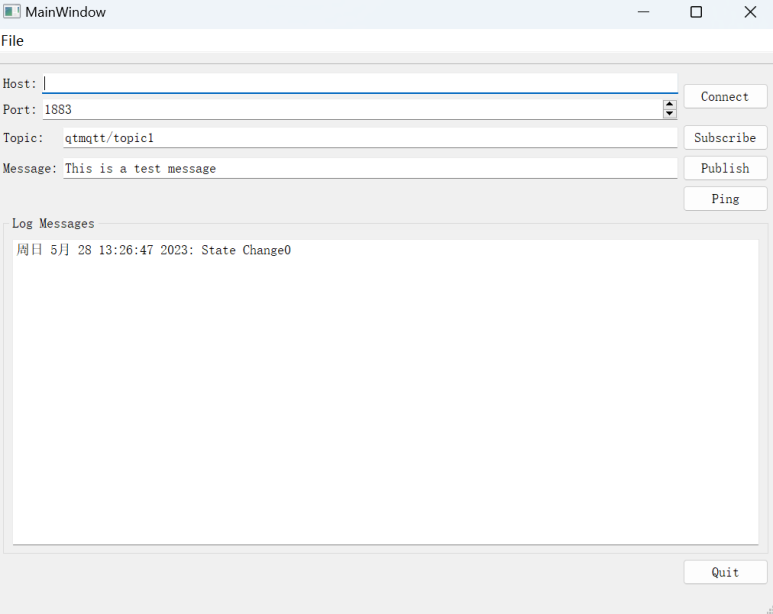


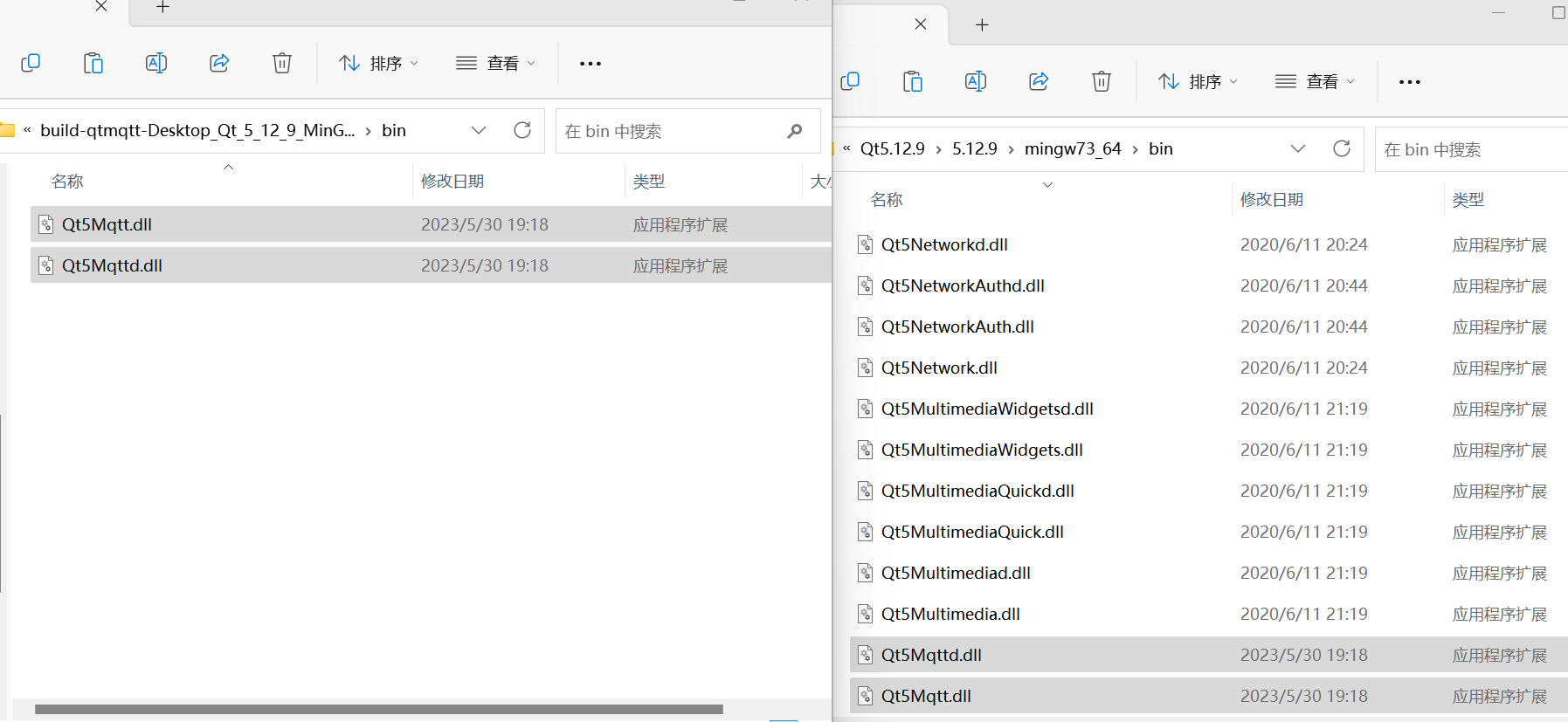
图1 SimpleClient示例运行界面图

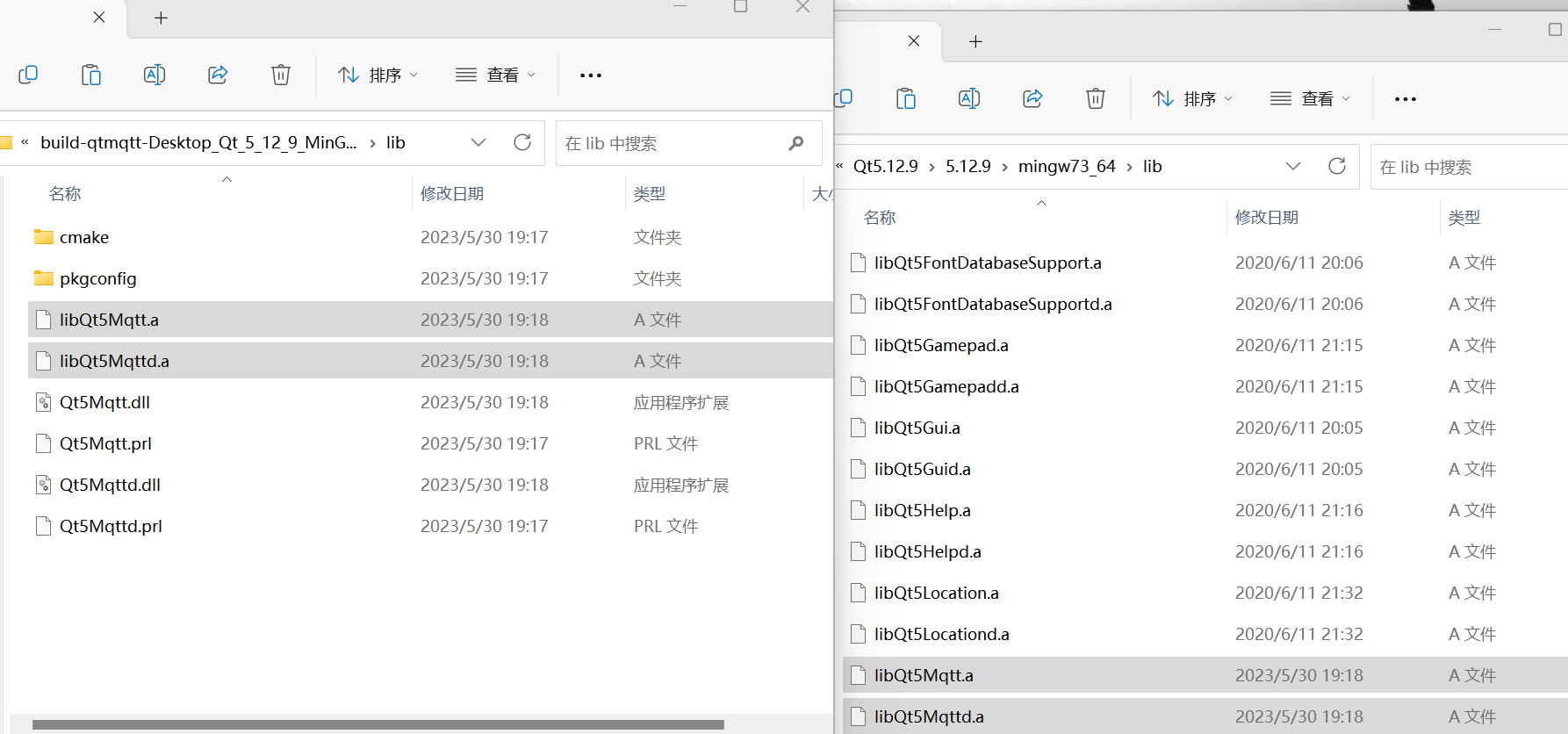
1. 第一步

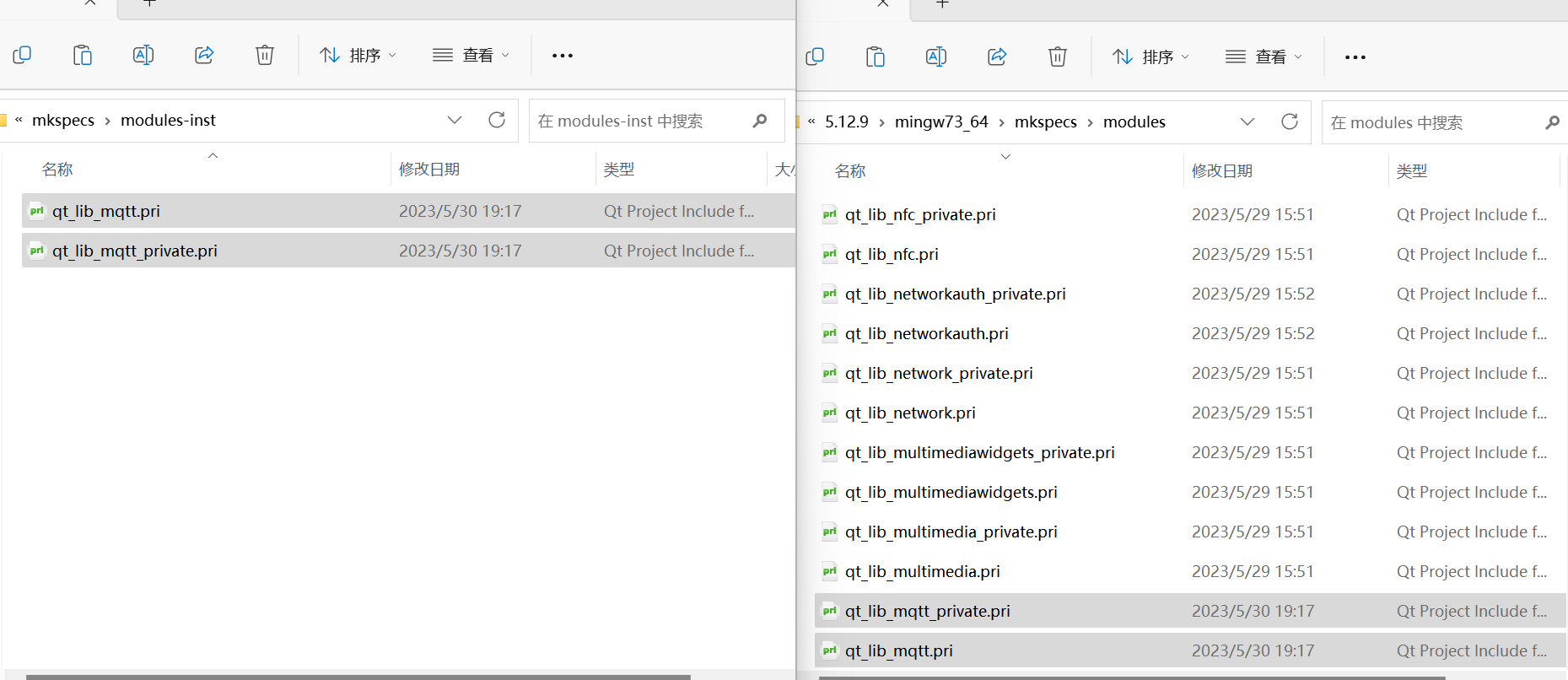
通过GitHub下载并打开文件包中的qtmqtt.pro工程文件，分别执行build和run qmake（汉化版本名称分别位编译和运行），生成编译输出文件夹，本例中位：build-qtmqtt-Desktop\_Qt\_5\_12\_9\_MinGW\_64\_bit-Debug。



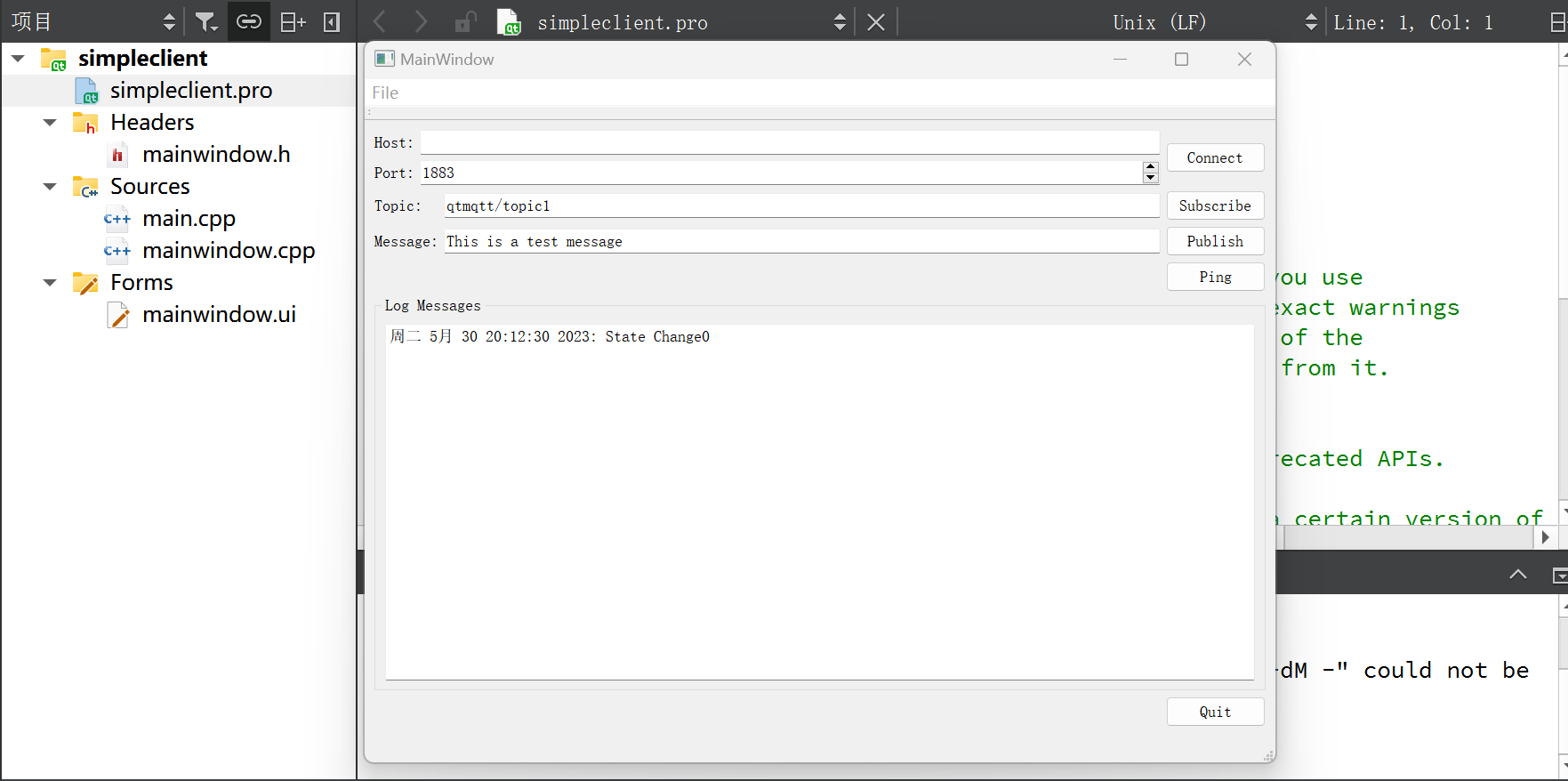
2.分别将该文件夹中的lib文件夹内6个文件，bin文件夹内2个文件，mkspecs/modules内1个文件复制至C:\Qt\Qt5.12.9\5.12.9\mingw73\_64下对应文件夹（注意实际操作可能存在不同安装路径）。







3.打开项目文件，并运行



4.连接以下两个服务器地址：test mosquitto.org 或 broker.hivemq.com

连接服务器，订阅并发布消息。进行测试

