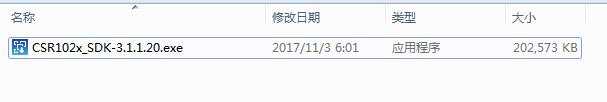
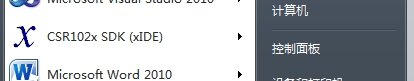
CSR102X Bluetooth Relay Control

1. xIDE development environment  
 1. xIDE environment installation

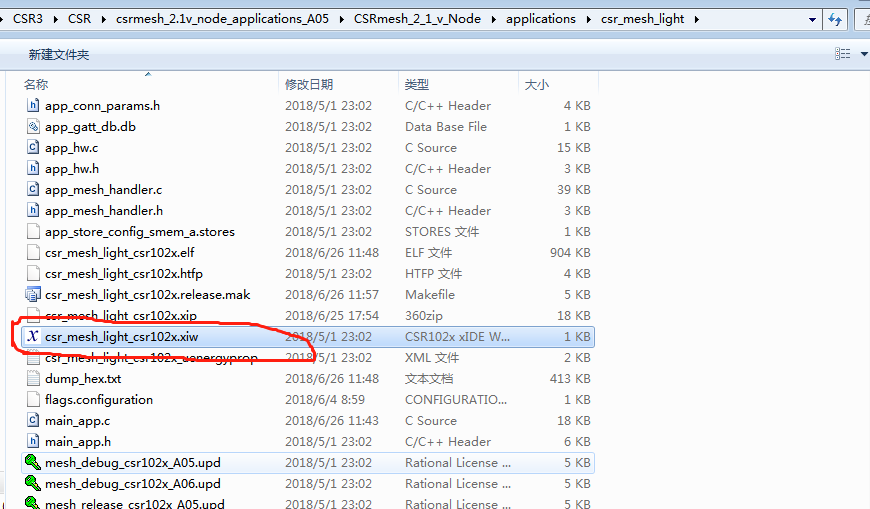
xIDE is an IDE integrated development environment developed by Qualcomm for the CSR platform. The xIDE SDK can quickly complete the development and design of the Bluetooth control project based on the CSR102x platform.  
Download the CSR102x development SDK and demo code, find CSR102x\_SDK-3.1.1.20.exe, double-click to install the SDK



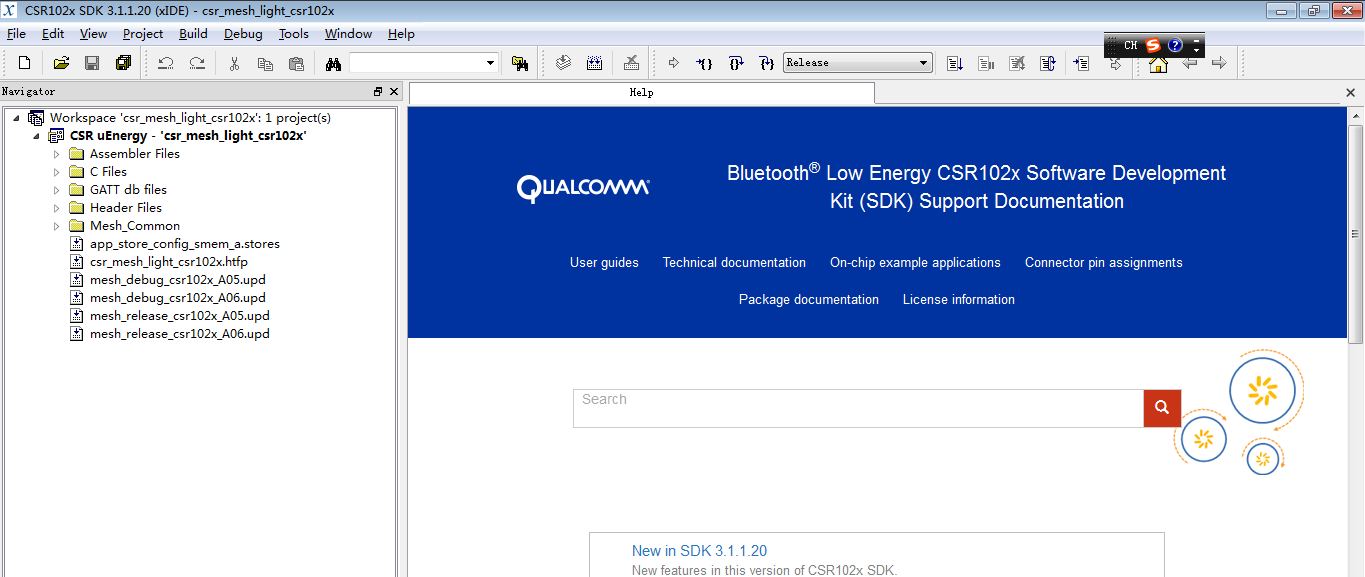
After the installation is completed, the xIDE application will be generated under Windows, and the application will be the integrated development platform of CSR102X, as shown in the following figure.



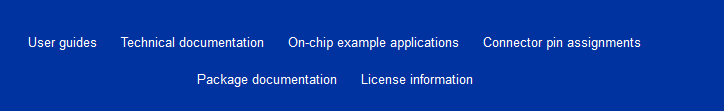
2. Use xIDE to preview the CSR102X project code and open the demo code as shown below.



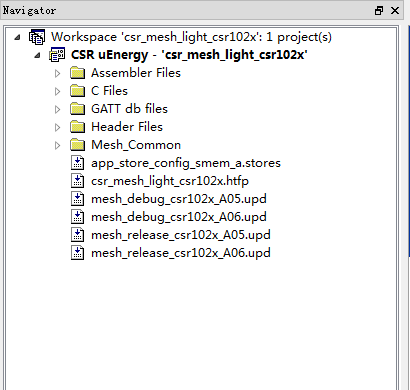
Click on the file in the red area to launch xIDE to open the project, as shown below:



In the following figure area, you can get the relevant API interface description and help documentation about the use of the SDK.



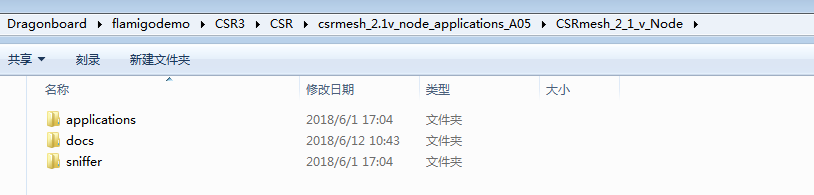
In the image below, you can preview the relevant files and double-click to edit the relevant code files:



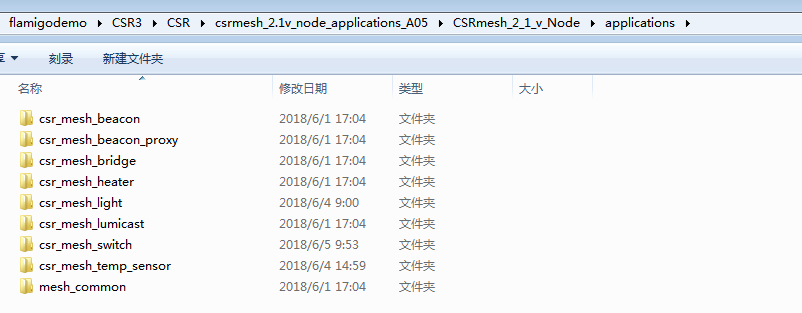
3. Code compilation and debugging  
 The following figure shows the code compiling and online debugging. Before debugging, you need to connect the emulator.



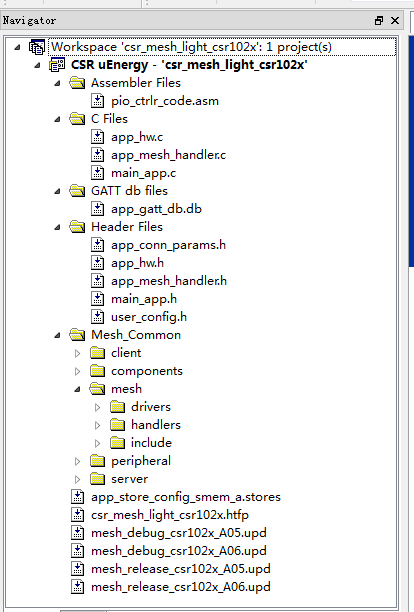
2. Development implementation steps  
 1. Reference and read existing demos  
 Find the demo by downloading the SDK of CSR102X, as shown below:



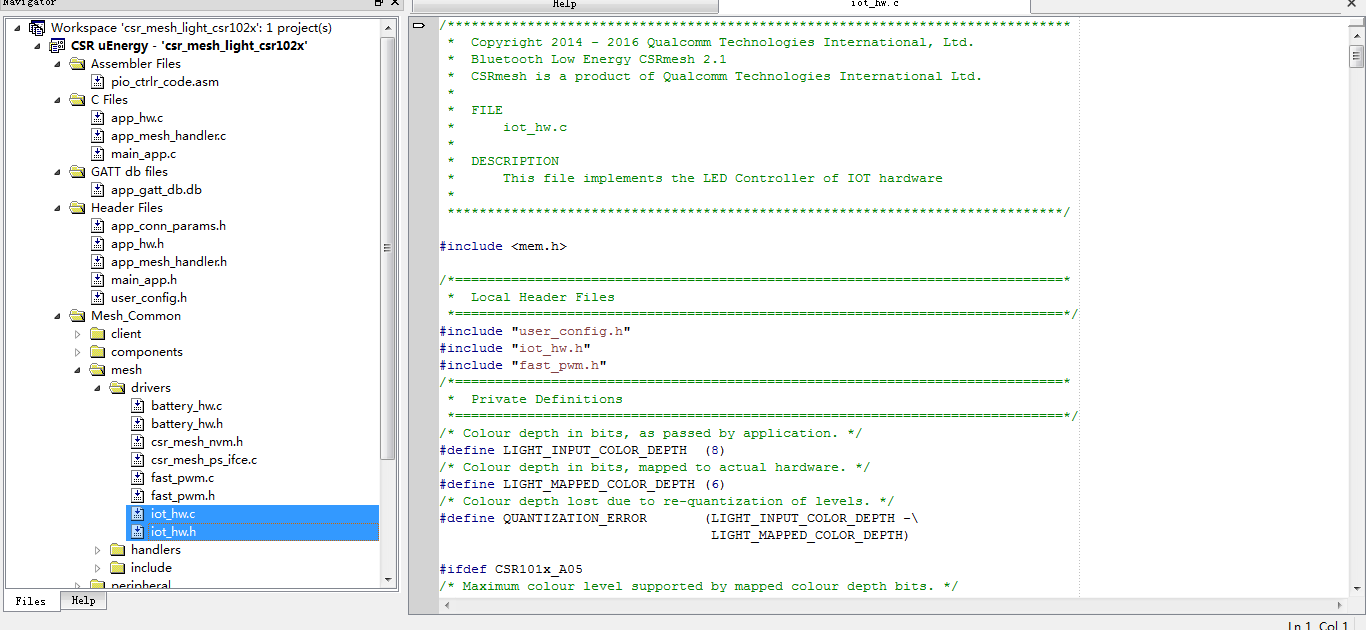
This includes the source code and documentation for the demo. The applications below include a simple demo, as shown below:



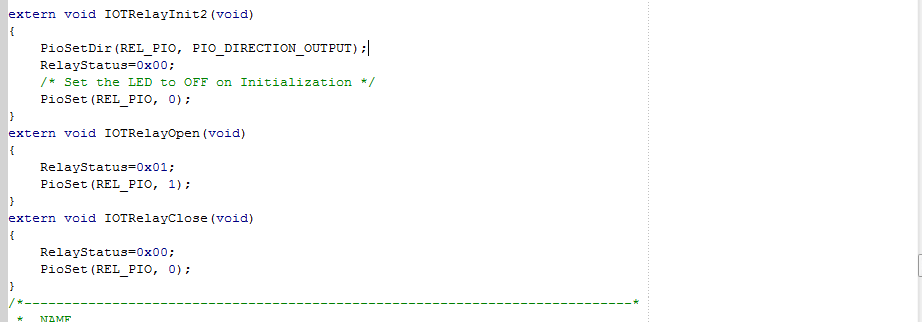
By reading the relevant demo instance, you can select a demo based on the development process, add and modify related modules on the existing basis, and quickly complete the development. Here, the code of csr\_mesh\_lightdemo is selected as the basis, in the code. Based on the development, the project code structure is as follows:



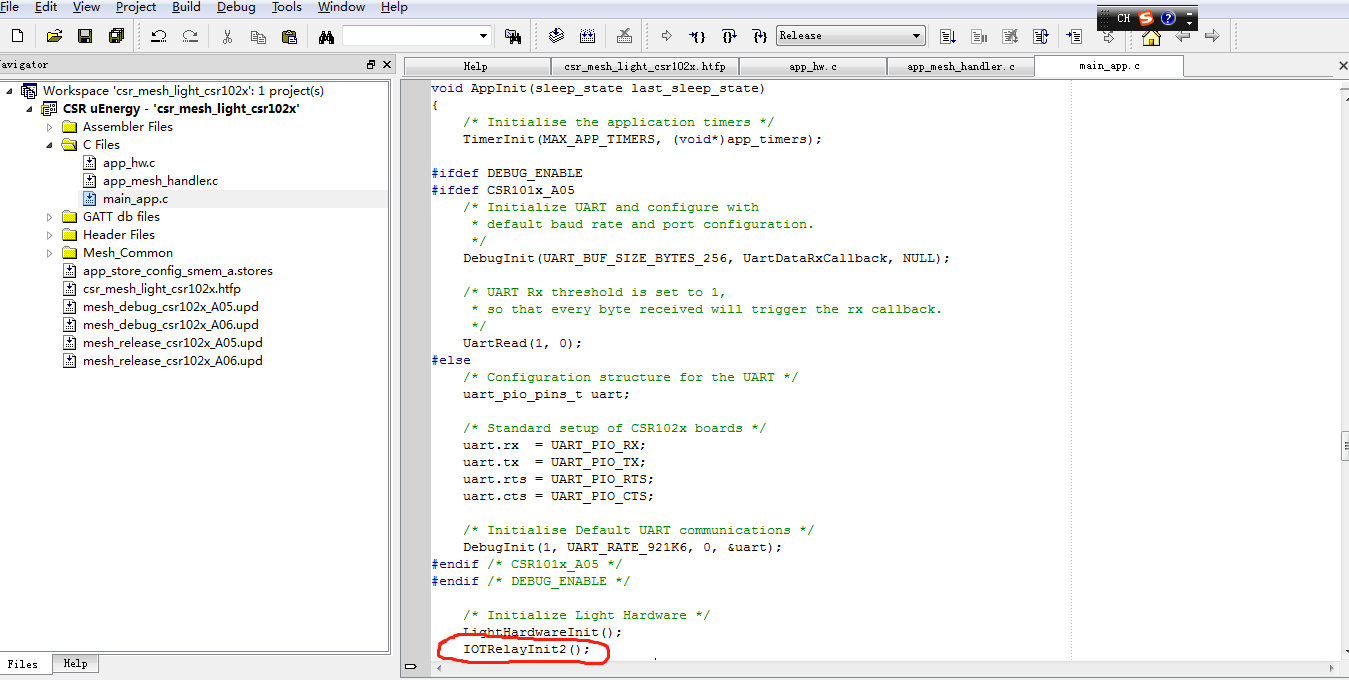
2 modify the code to achieve control  
 Considering the need to implement the control of the relay, it is necessary to modify the code of the relevant driver layer. Firstly, the control operation of the IO is realized to realize the control of the relay. The code is implemented in the lowest IO file, as shown in the following figure. :



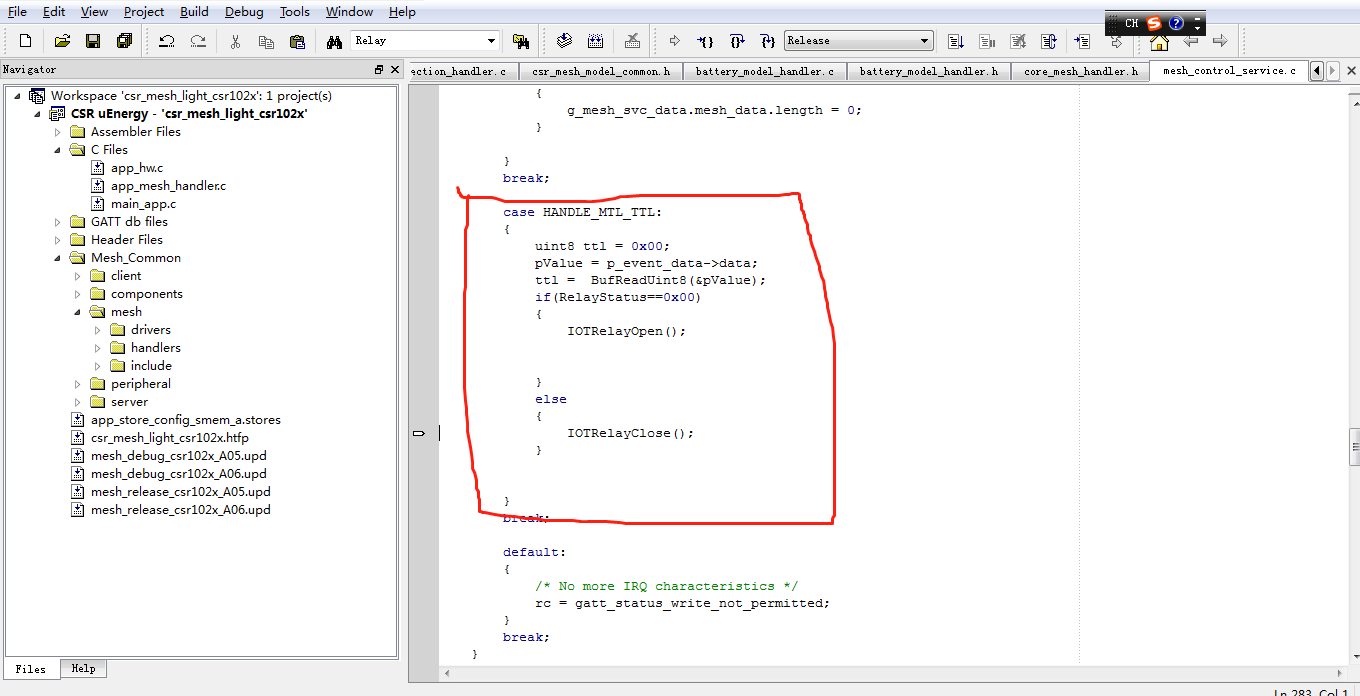
According to the above ideas, to realize the control of the relay, it is only necessary to implement the interface function in these two files. The specific implementation is as follows:



The three functions of init, open and close are implemented. The specific implementation code can refer to the code .c and .h files.  
 3 initialization



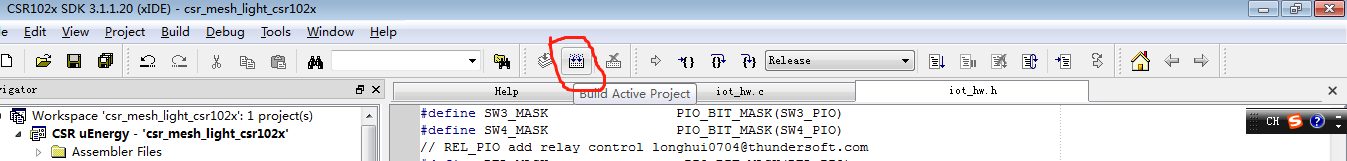
4 added to the Bluetooth command response  
 Here we use the HANDLE\_MTL\_TTL command to respond to the status of the Relay and change the state of the relay, as shown in the following figure:



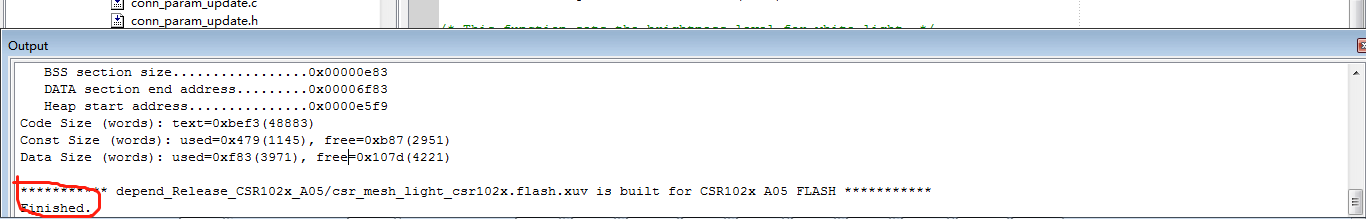
3 Compile and run

After the demo code is modified, you can compile and download the code to the device to run the observation. Specific steps are as follows:

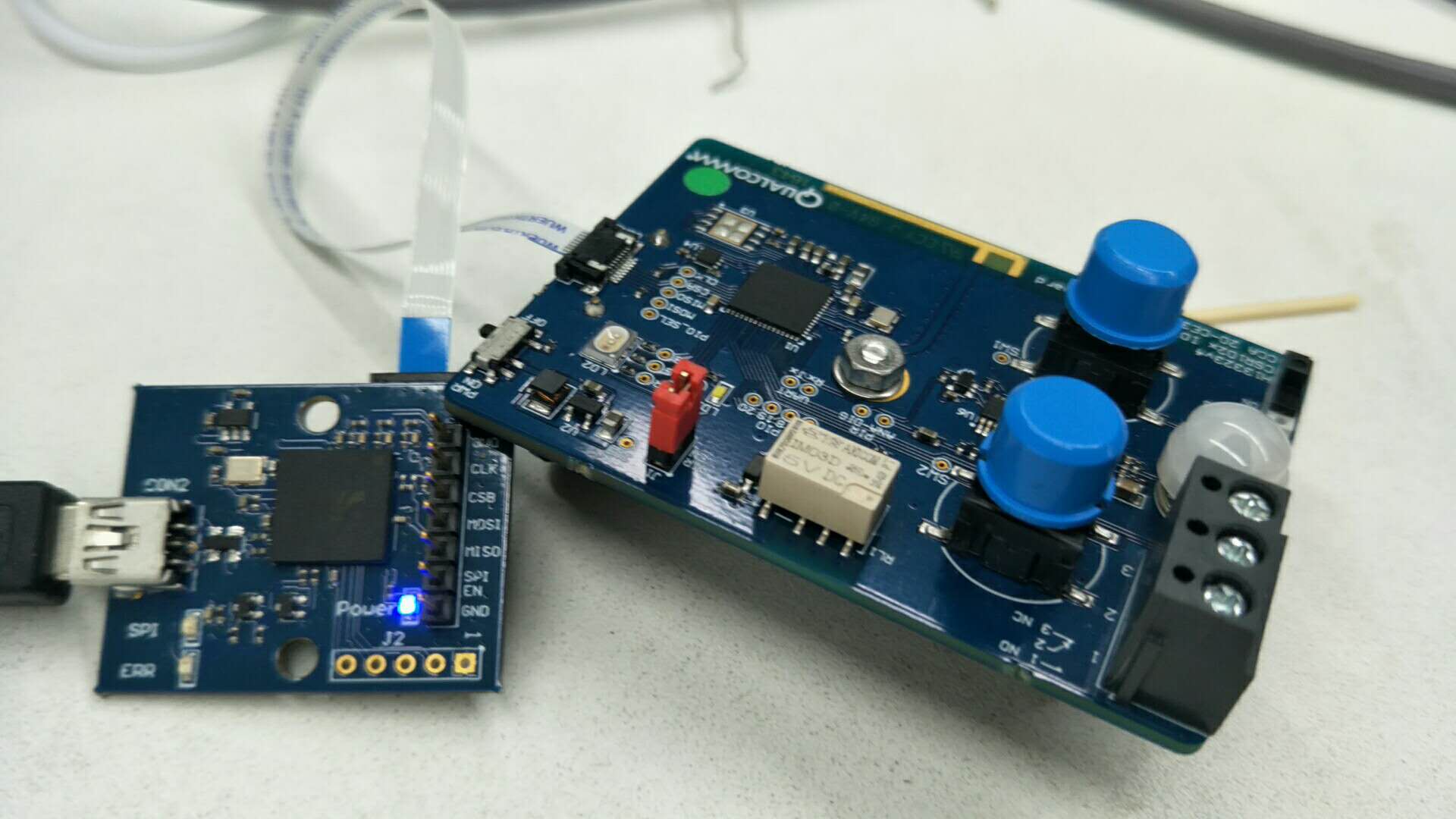
1.Compile the code  
 As shown in the figure below, open the project document and click the red area button to compile the code.



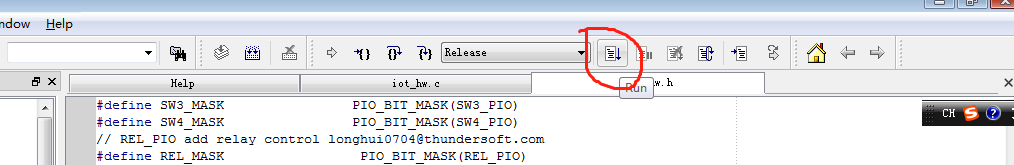
After the compilation is completed, the output column is as follows:



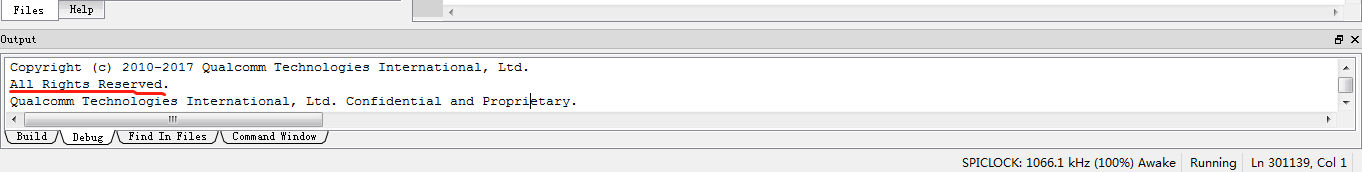
2. Download to the development board  
 If you need to download the compiled program to the development board or online debugging, you need an emulator to connect to the development board, as shown below:



The emulator is connected to the computer via USB, connected to the development board hrough the cable, and displays the blue light in the idle state.  
 At the same time, after starting the development board switch, you will see that the three-color light on the development board starts to light blue and flashes.  
 Then you can click the run button to download to the development board, as shown below:



At the same time, the output information in the output window is as follows:



A red prompt indicates that all files have been downloaded successfully and are running.  
 At this point, you can see that the emulator yellow light flashes, and the development board blue light flashes, and the development board runs normally.

3.Test verification  
At this point you can connect the relay on the development board to the multimeter or the lamp, and then control the specific control method via Bluetooth as follows:

1) Query Bluetooth devices  
 On the dragonboard410c with the Linux system installed, run sudo hcitool lescan on the command terminal to query the Bluetooth device. The device name is light2 and you can see the device address.  
 2) Connect the device by device address  
 Sudo gatttool -b EE:52:5B:04:00:02 -t public –I

3) Enter the connection mode, then execute the connection command to connect the device  
 [EE:52:5B:04:00:02][LE]> connect  
 Prompt Connection successful indicates that the connection successfully entered [ EE:52:5B:04:00:02][LE]> Control mode  
 4) Execute switch relay command  
 Char-write-req 0x0022 0x01 turn on the relay  
 Char-write-req 0x0022 0x00 turn off the relay  
 5) Can use the multimeter or LED light to detect success during the execution of the command