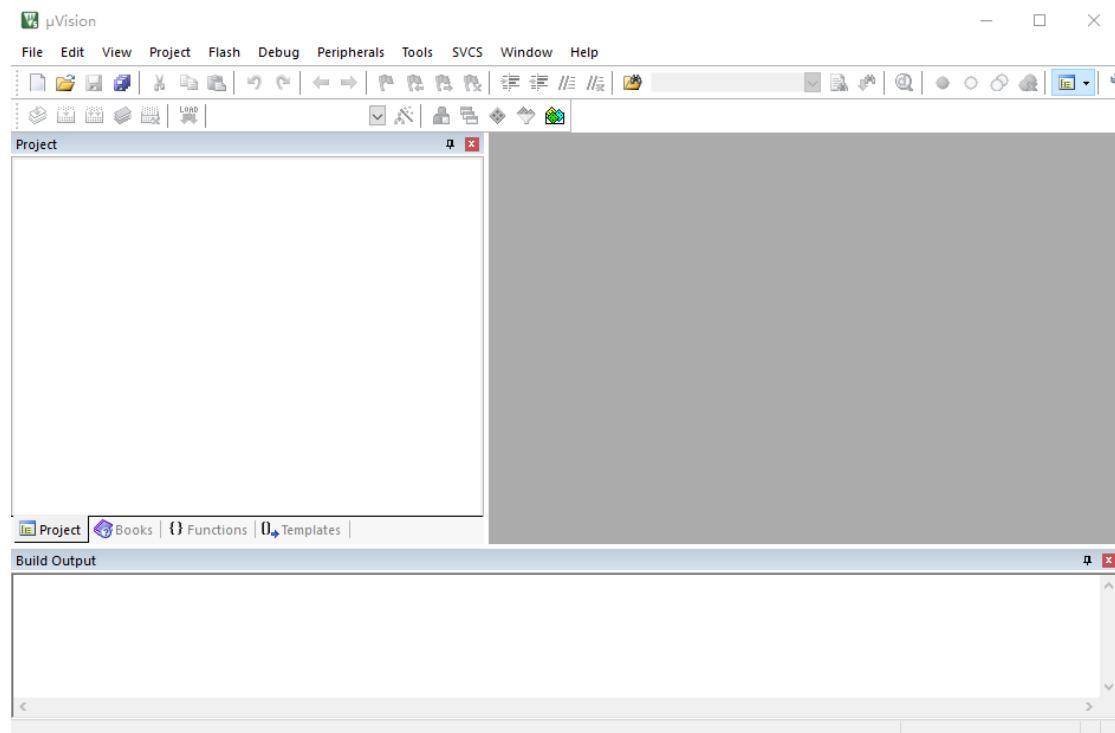


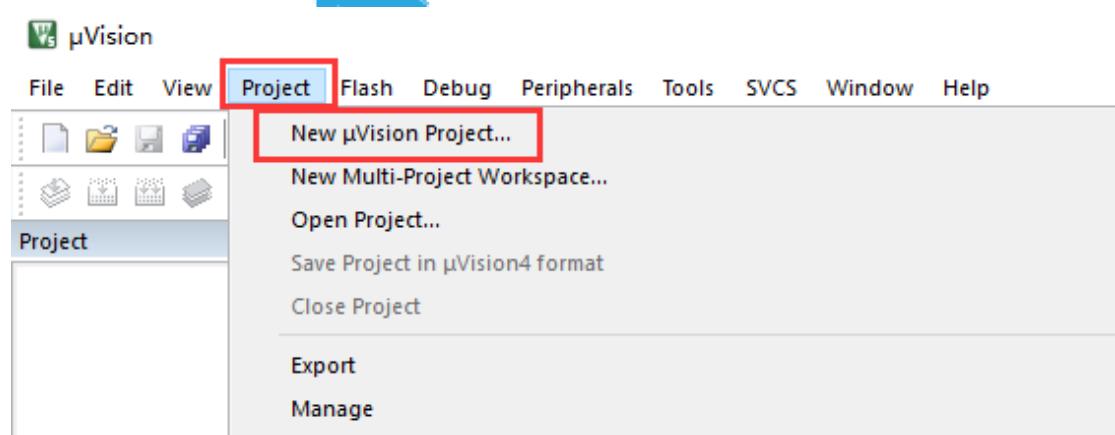
## Steps to create a new project with Keil uVision5



1. You need to double-click  to enter KeiluVision5 programming interface.



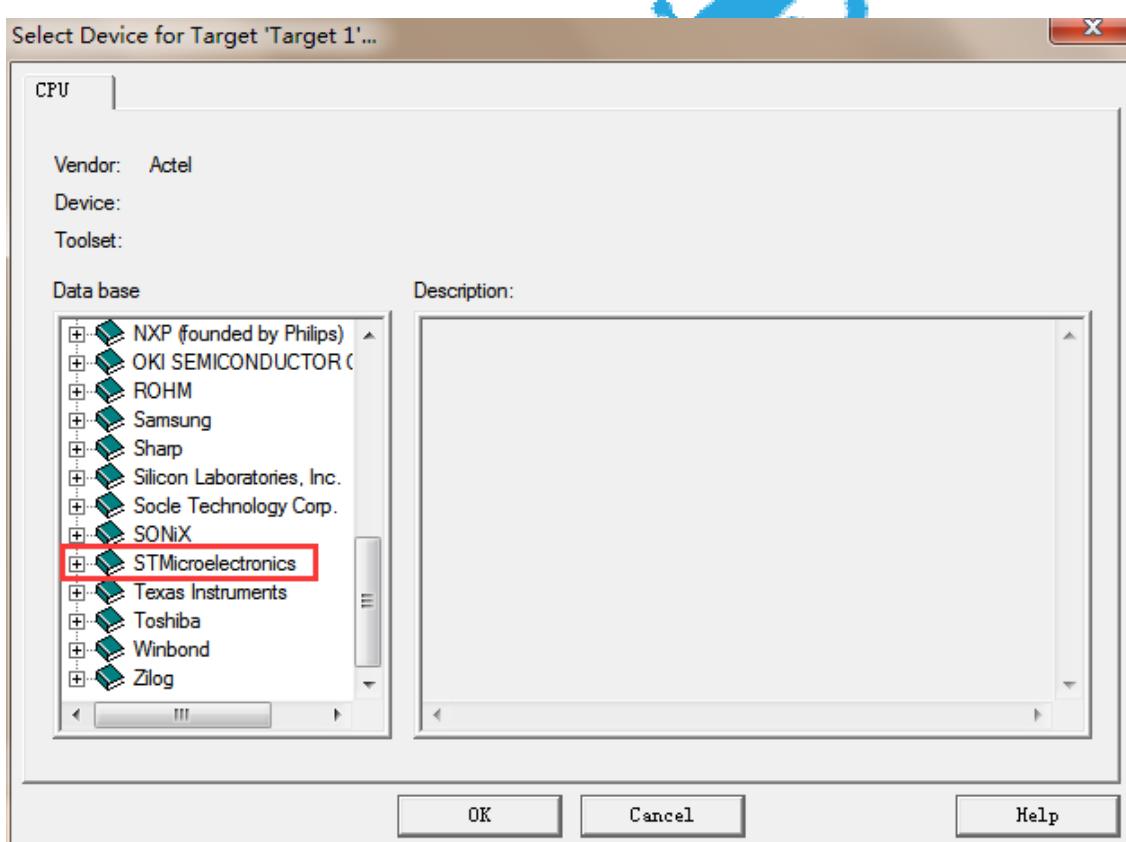
2. You need to click  【Project】 ---- 【new uvision project】 to create project.



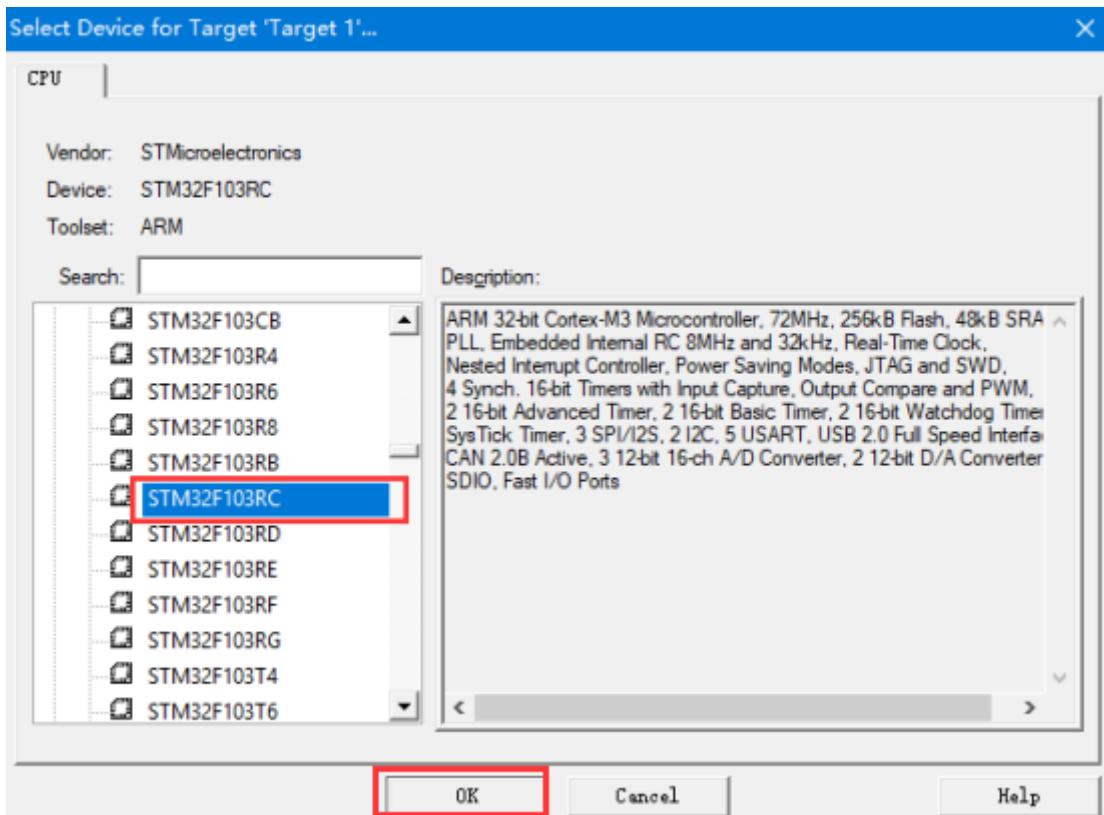
3. You need to select the location where the project is stored and name the project.



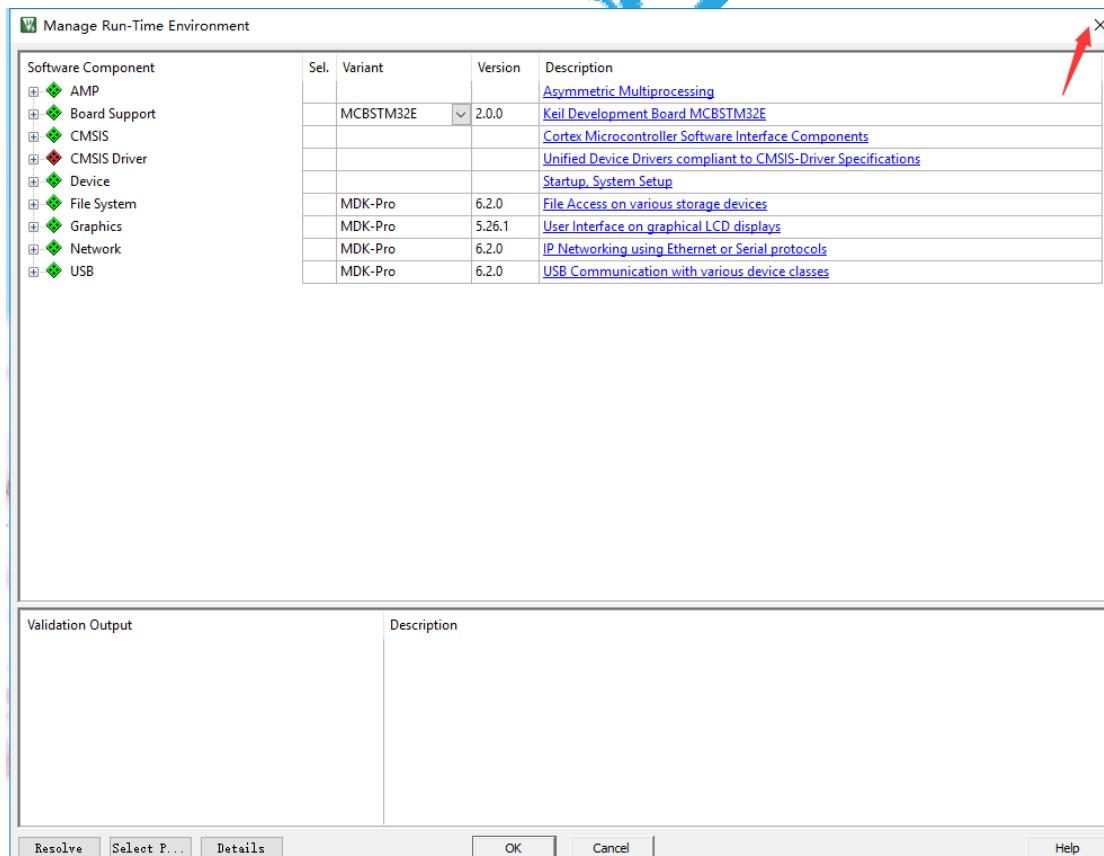
#### 4. You need to select STMicroelectronics.



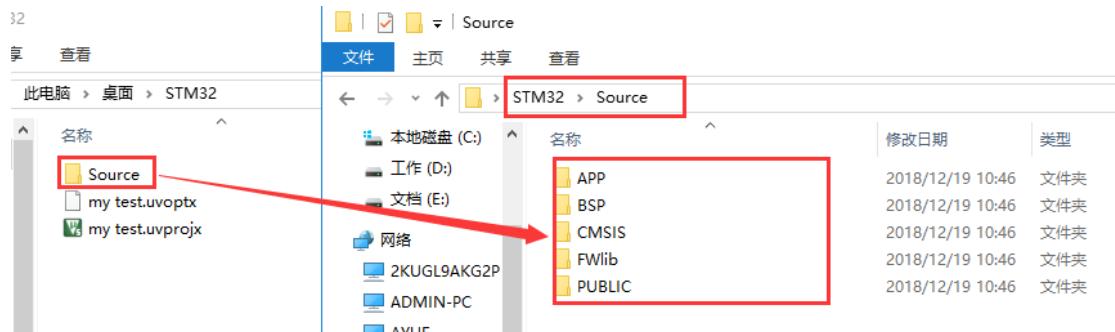
#### 5. The STM32 Visual Tracking Car, we use is the STM32F103RCT6 chip, so we need to select STM32F103RC, then we need to click OK.



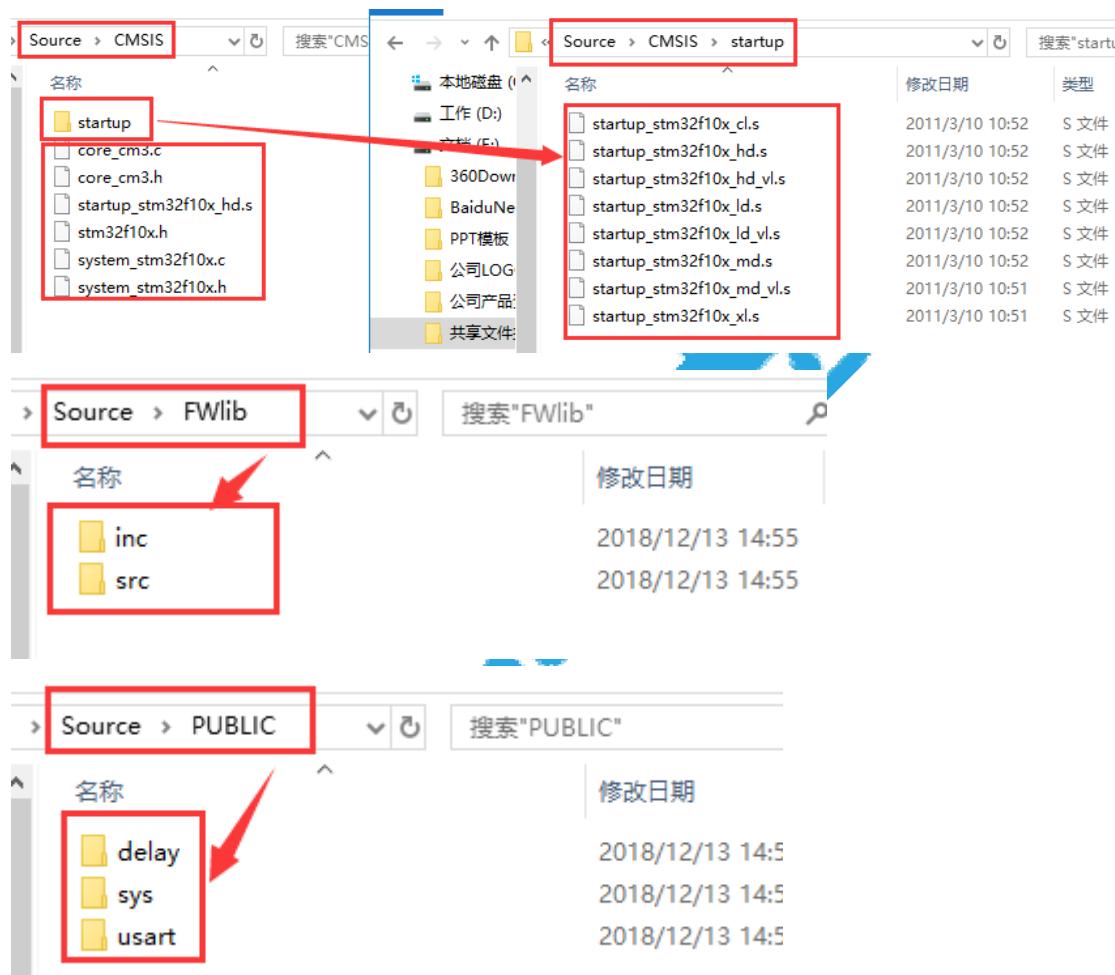
6. Close the window as shown below



7. You need to add some folders in our newly created project folder to store some files. As shown in the figure below, only examples.



8. Put some files into these folders separately, as shown in the figure below.

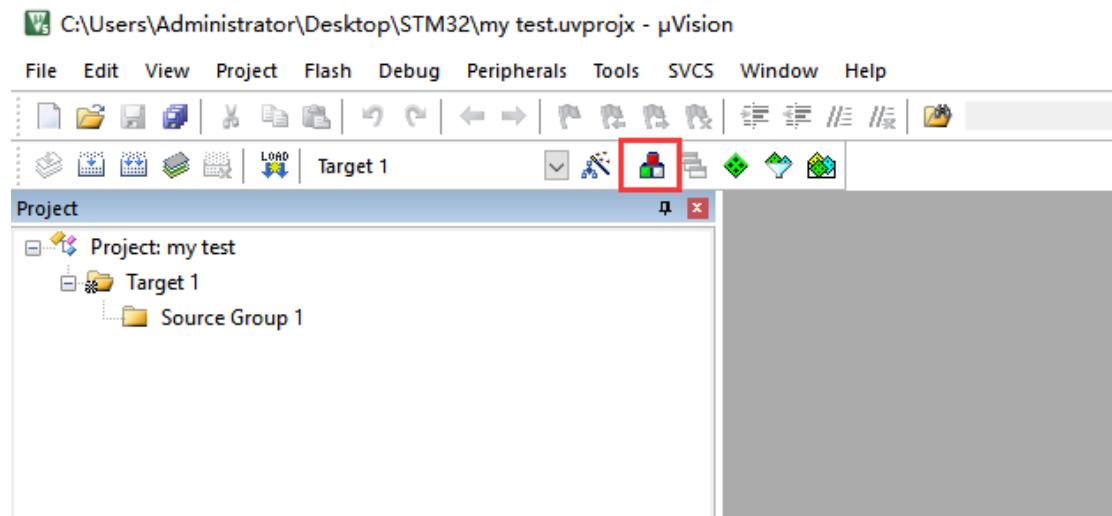


**The above three folders store the firmware library that comes with STM32.  
We have provided these files, so you only need to copy directly.**

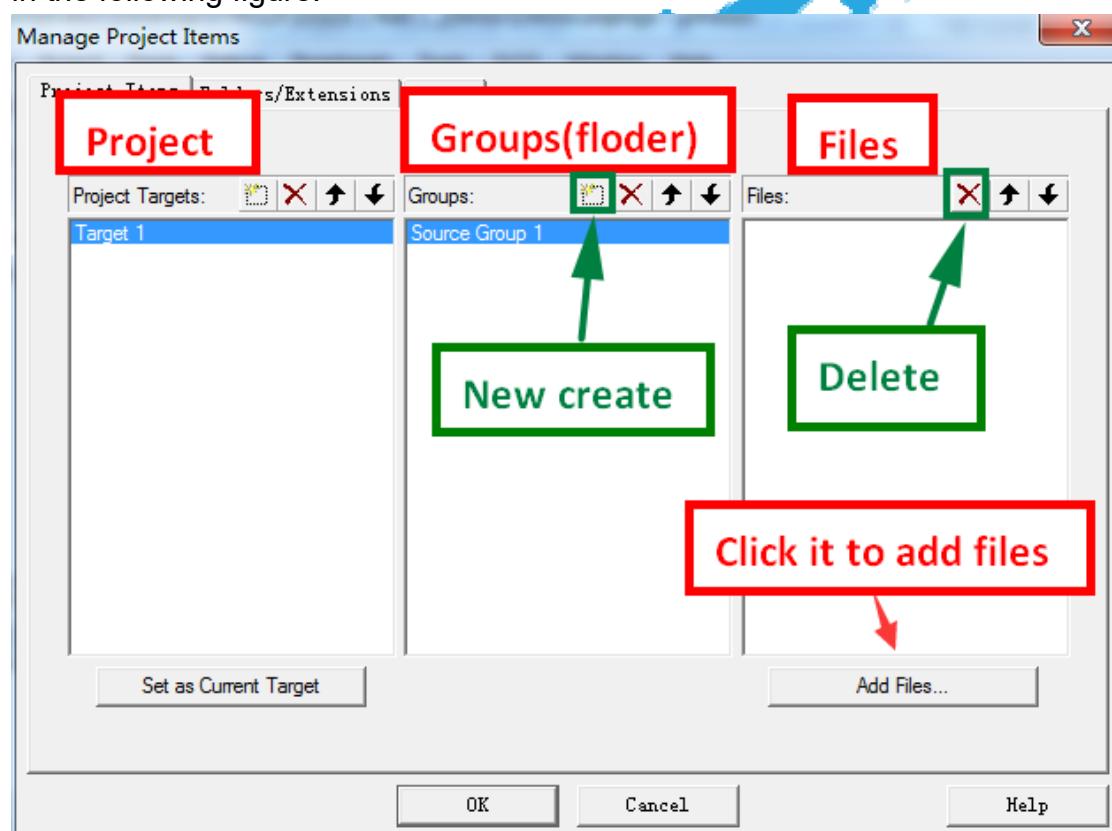
BSP is used to store files related to hardware configuration

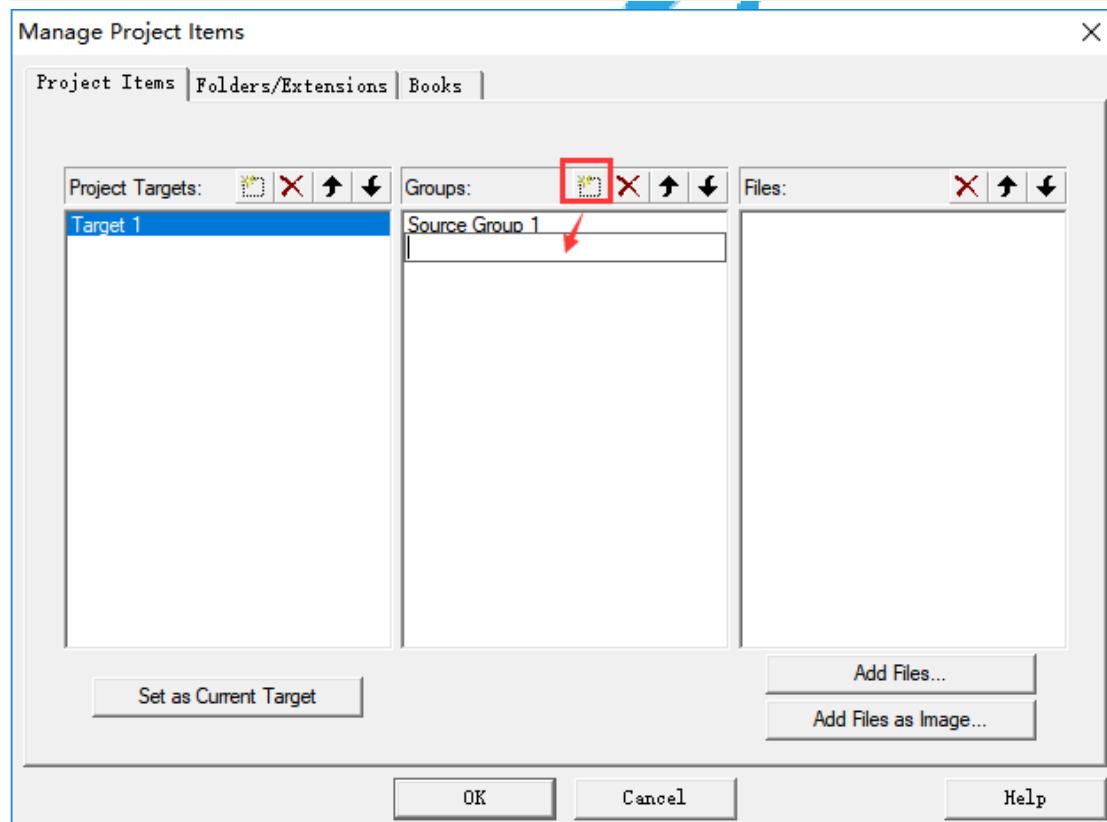
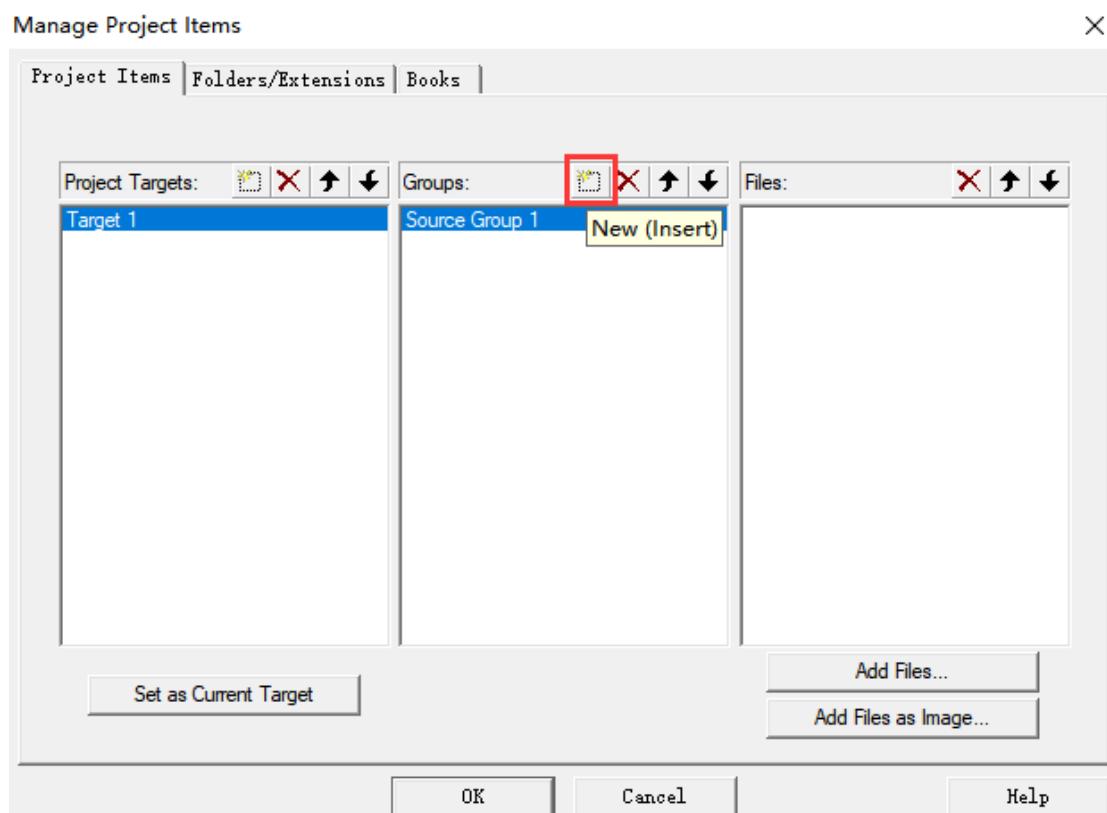
APP is used to store application files

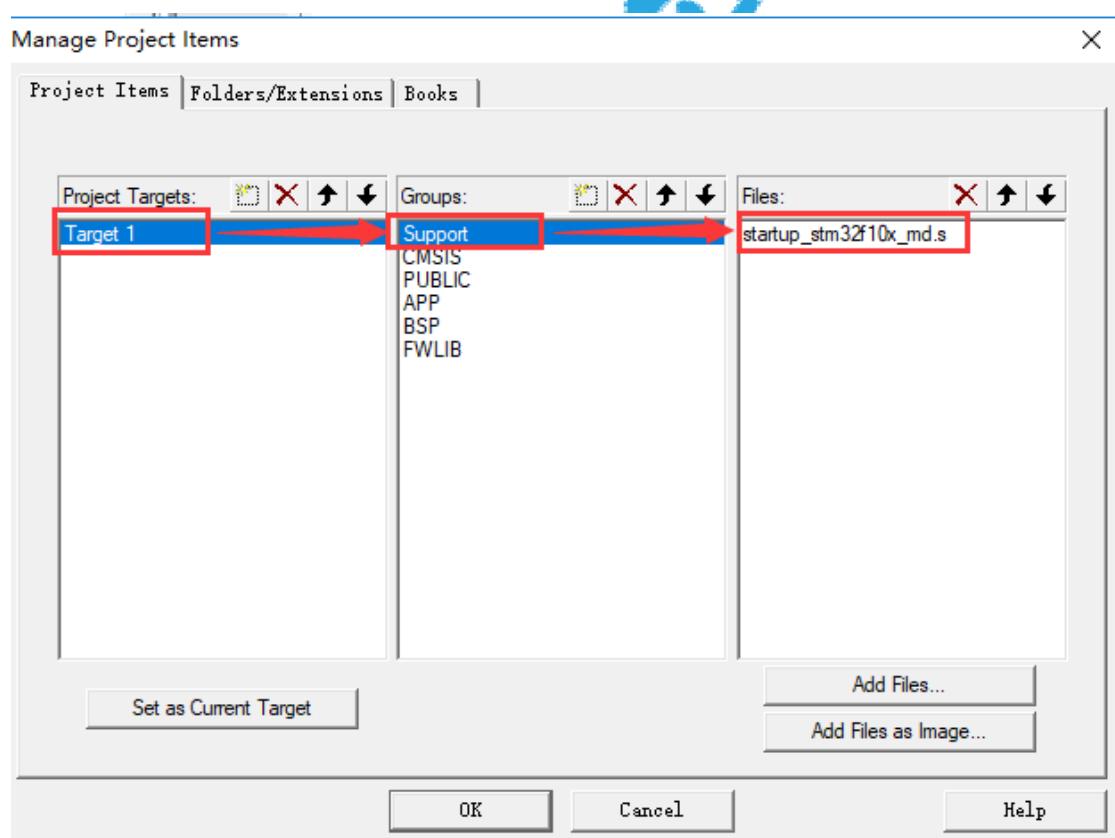
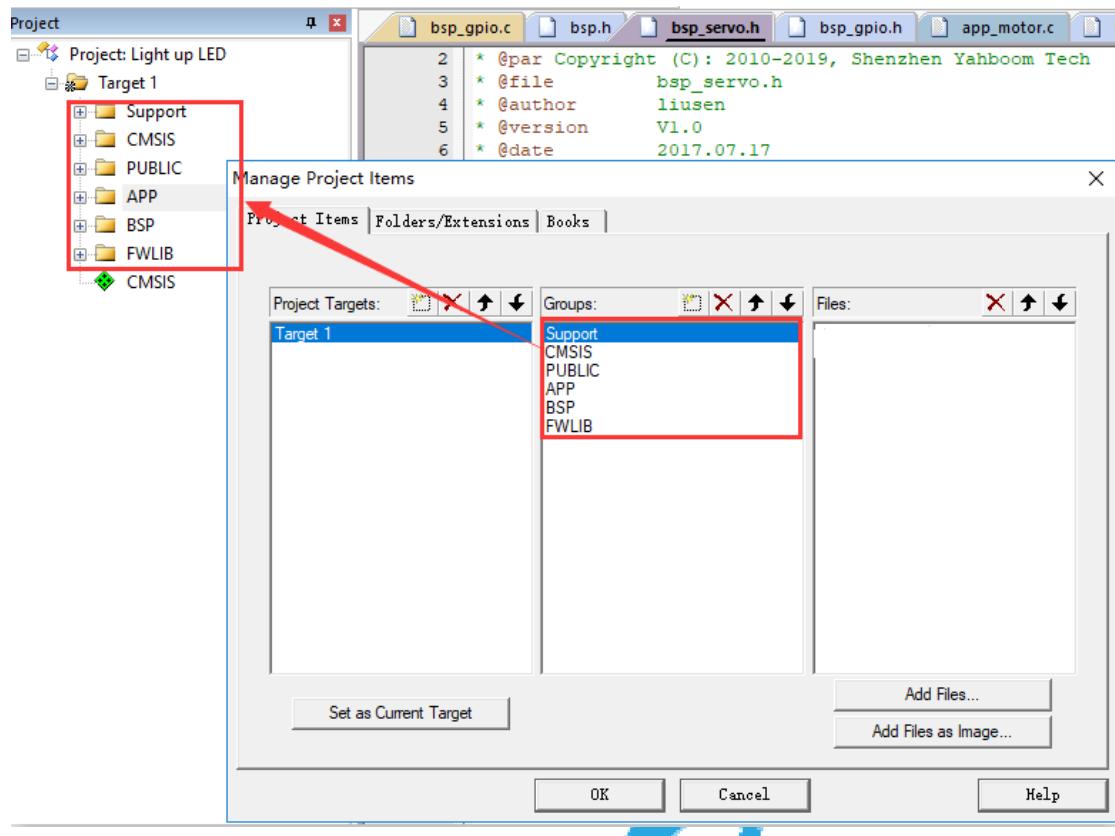
9. You need to Click on the location shown below to set up the compiled project environment.

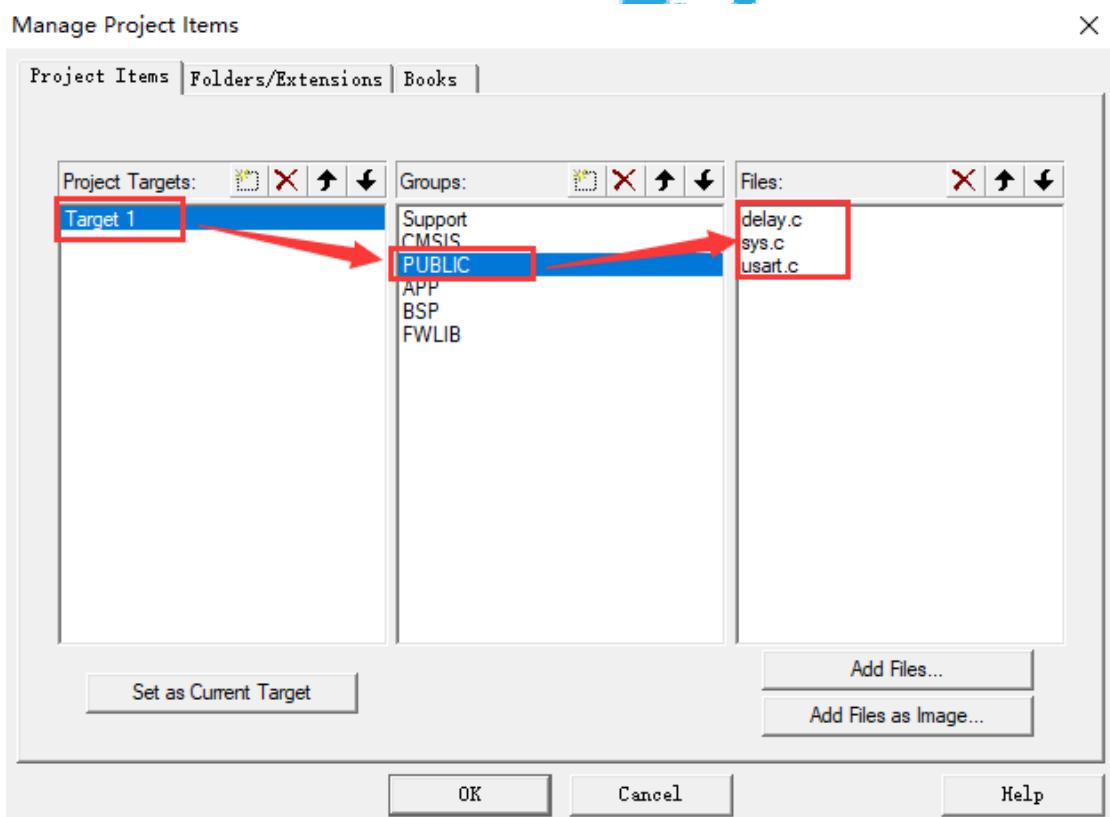
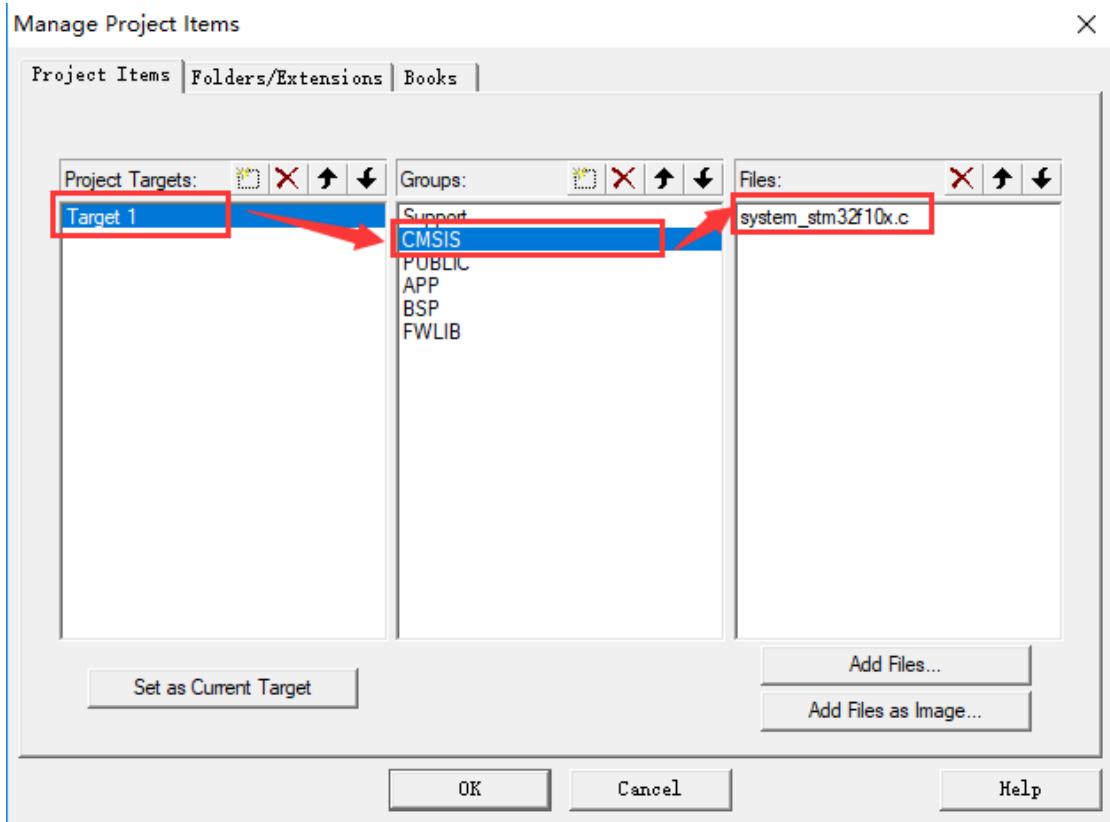


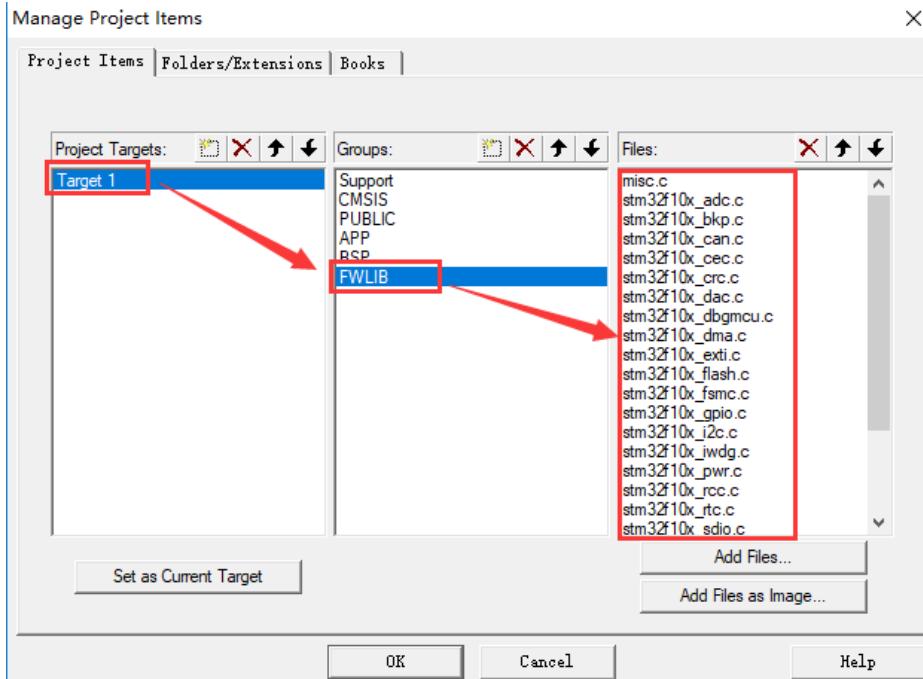
10. You will see the interface shown below, and create a new folder as shown in the following figure.







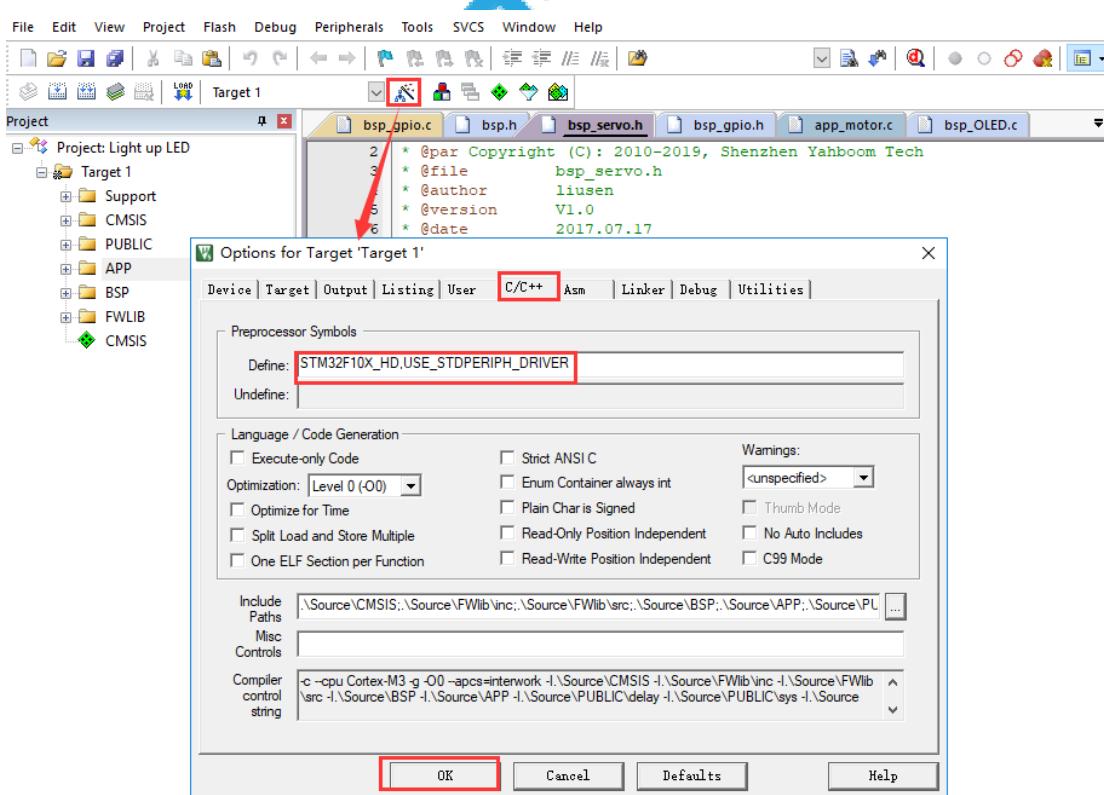




11. Keil *μ*Vision 5 allows the user to configure the project based on the actual conditions of the target hardware. By clicking the target toolbar icon (red box) or clicking the menu item **Project->Options for Target**, you can specify the target hardware and related parameters of the selected device on-chip components in the pop-up Target page. The following figure is related to the settings of this example.

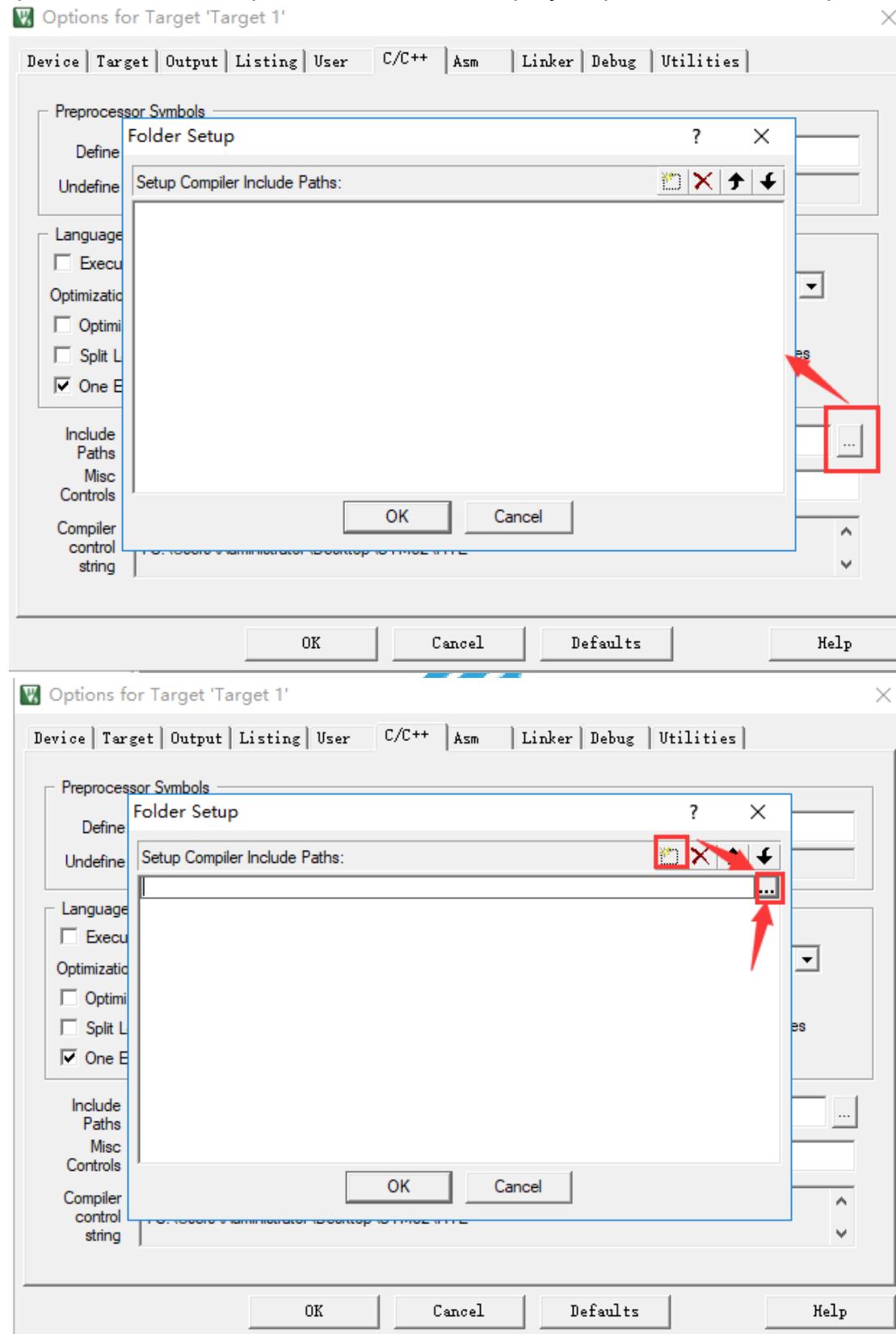
12. Click on the C/C++ option and enter:

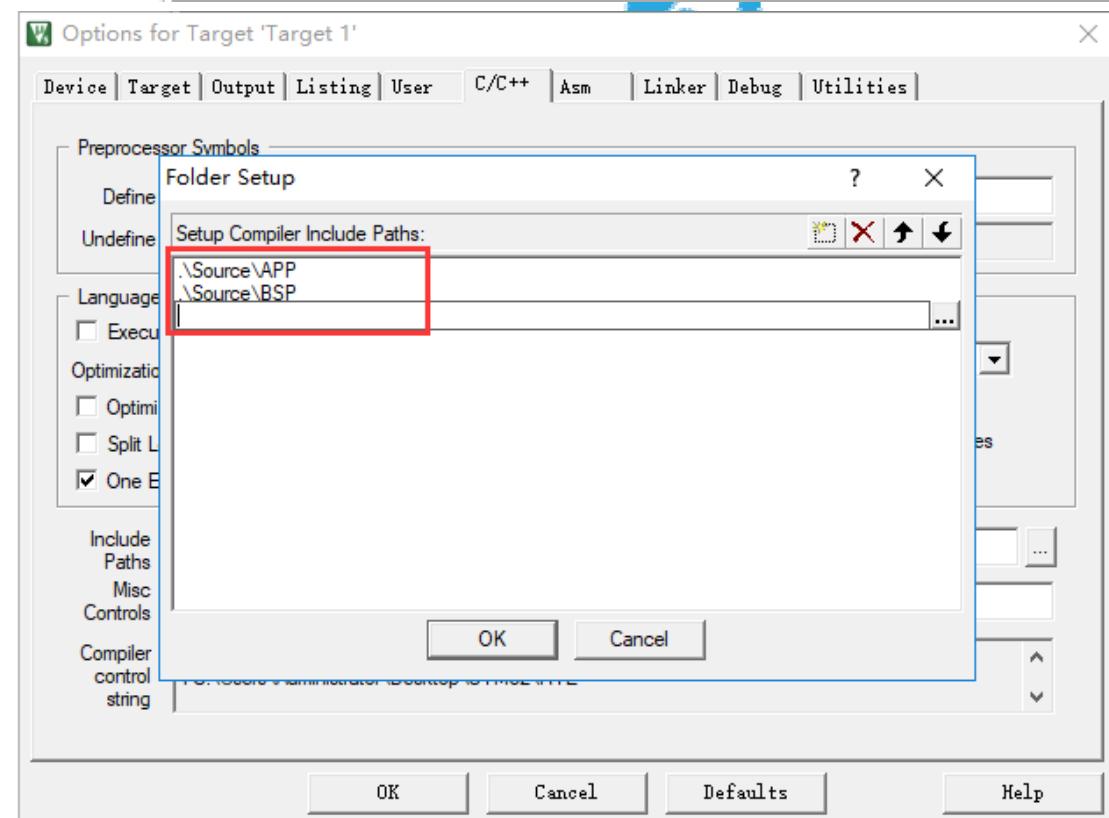
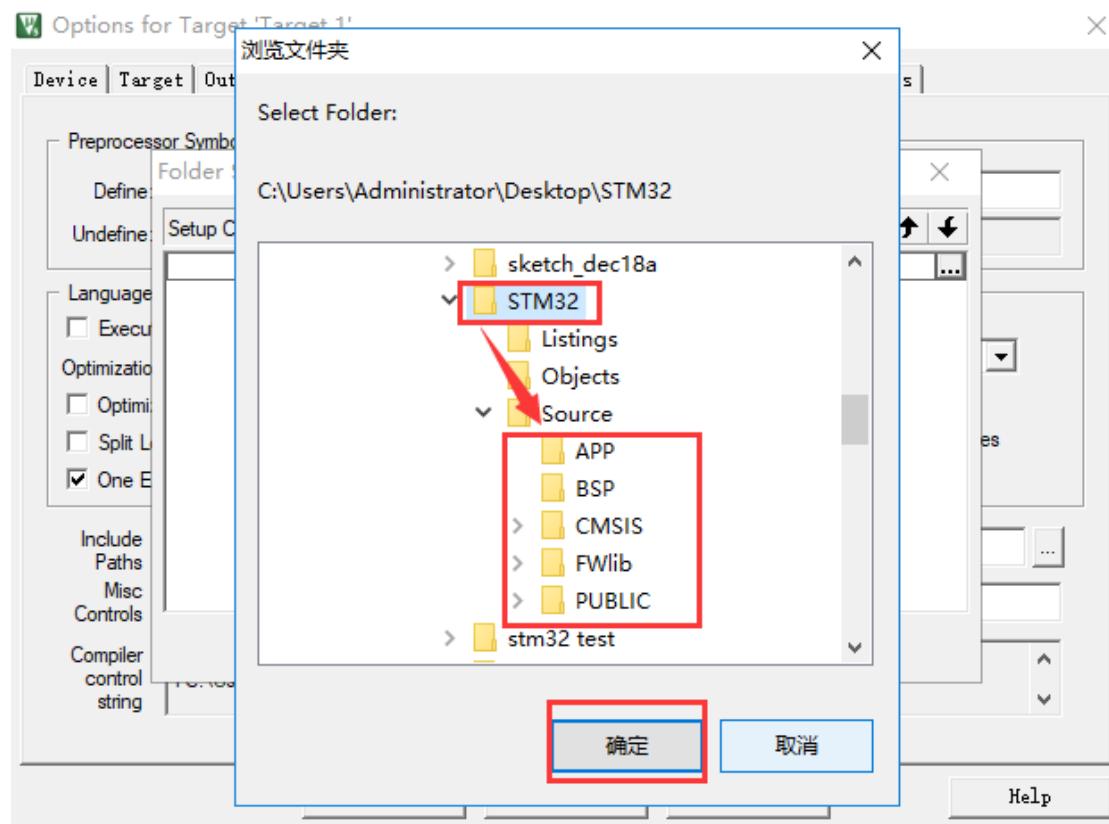
**STM32F10X\_HD,USE\_STDPERIPH\_DRIVER**



### 13. Include paths:

Since there are many folders in our project, we need to add the corresponding path so that the compiler can find it. Add step by step until the final completion





**Options for Target 'Target 1'**

Device | Target | Output | Listing | User | C/C++ | Asm | Linker | Debug | Utilities |

Preprocessor Symbols

Define: STM32F10X\_HD,USE\_STDPERIPH\_DRIVER

Undefine:

Language / Code Generation

Execute-only Code

Optimization: Level 0 (-O0)

Optimize for Time

Split Load and Store Multiple

One ELF Section per Function

Strict ANSI C

Enum Container always int

Plain Char is Signed

Read-Only Position Independent

Read-Write Position Independent

Warnings: All Warnings

Thumb Mode

No Auto Includes

C99 Mode

Include Paths: \Source\APP; \Source\BSP; \Source\CMSIS; \Source\FWlib\inc; \Source\FWlib\src; \Source\PU...

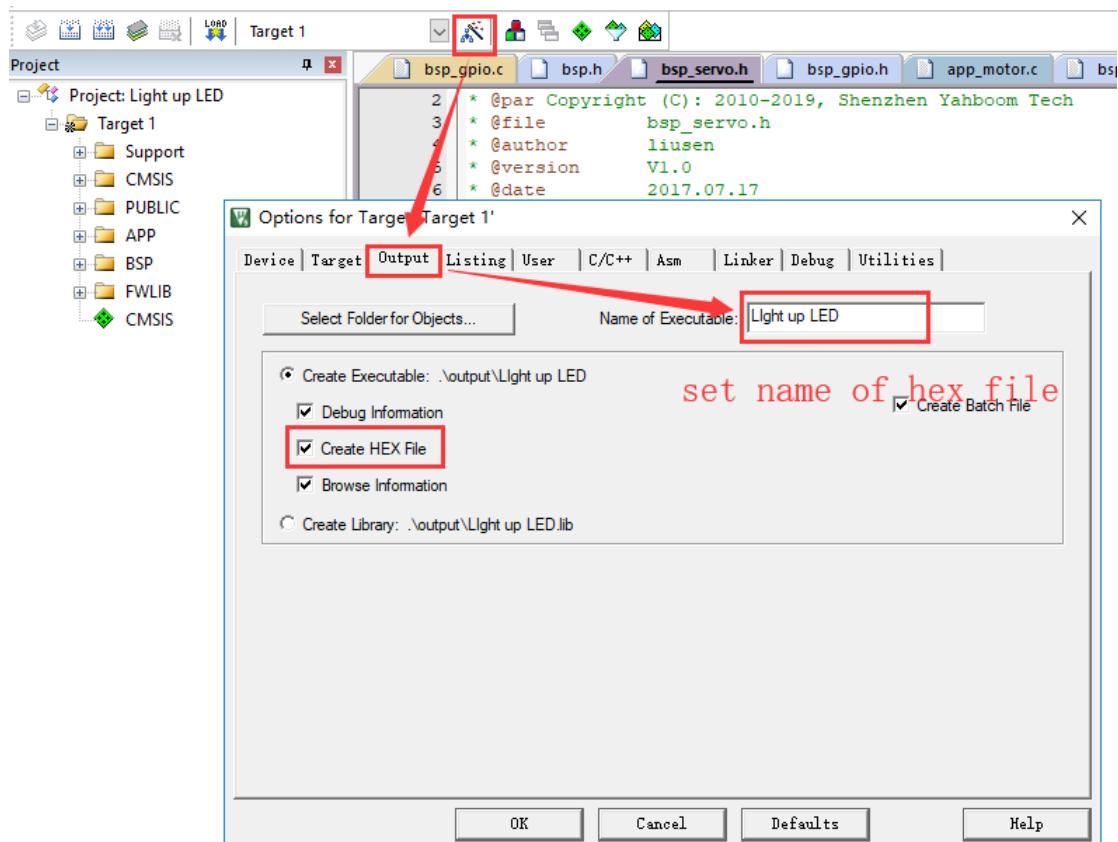
Misc Controls:

Compiler control string: -c -cpu Cortex-M3 -g -O0 -apcs=interwork -split\_sections -I.\Source\APP -I.\Source\BSP -I.\Source\CMSIS -I.\Source\FWlib\inc -I.\Source\FWlib\src -I.\Source\PUBLIC\delay -I.\Source...

OK Cancel Defaults Help

**Note:** Here is just an example. If you have other files in your project, you need to add the path.

#### 14. Setting the HEX file



15. We need to create a new Output folder Under the project folder to store the HEX file.

