FY-XDS100V3 and CCS6.2 Tutorial

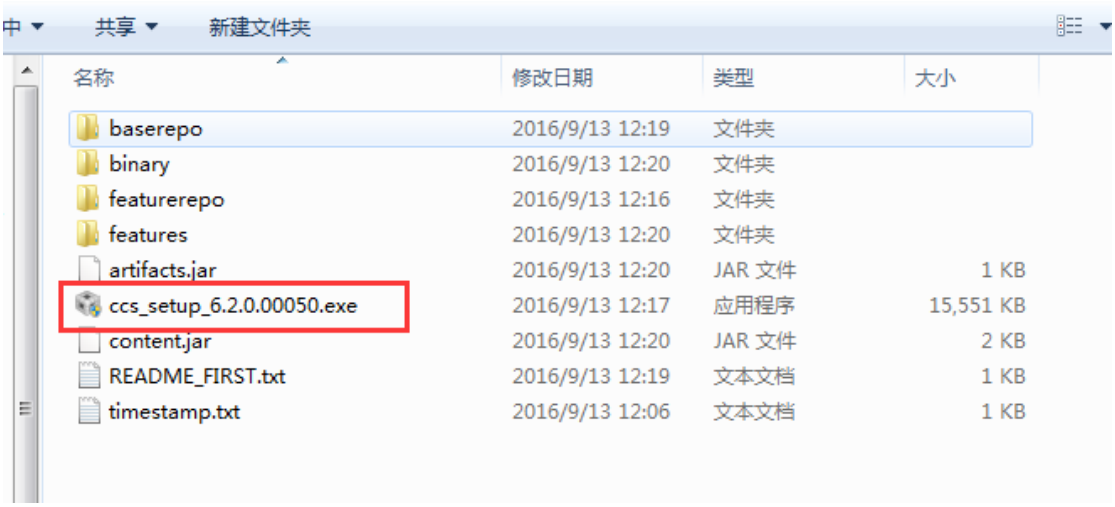
Note: The downloader does not provide external power:5This pin does not output power. This pin needs to be connected to the power supply of the target board, otherwise it will not work!4FeetGNDIt is also required to accept.

Note: Do not plug or unplug while power is on20PPlease plug in and out after power off.20POtherwise, the downloader may be damaged.

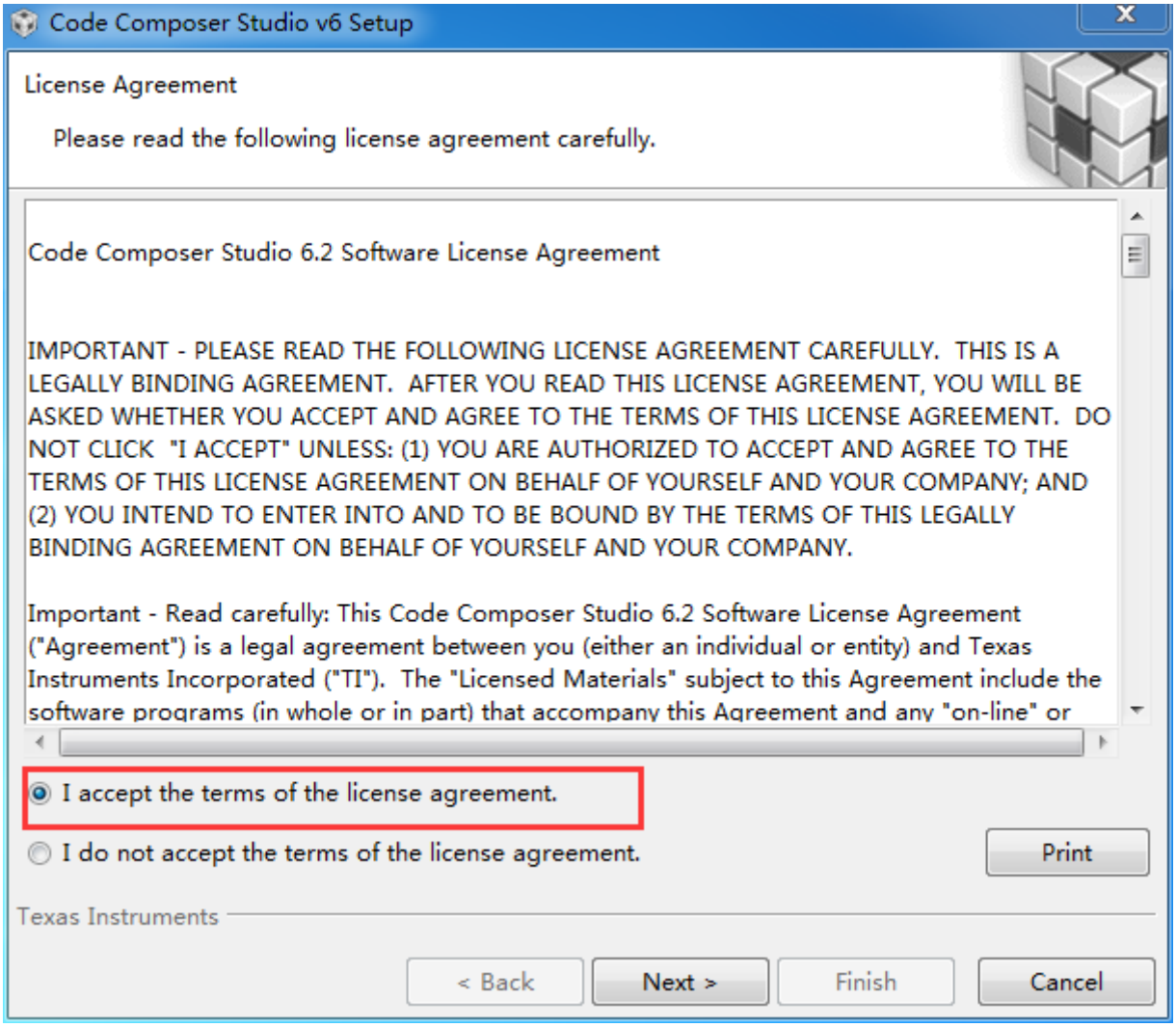
Chapter 1CCS6.2Software Installation

First we need to installTI DSPSoftware development environmentCCS（Code Composer Studio). If you have used emulator products from other companies. We recommend usingCCS6.2version, because temporarily Hezhongda'sXDS510PLUSandXDS560PLUSThe emulator only supportsCCS6.2, the compatibility of higher versions is not very good. In addition, we chooseXDS100 V2The emulator is only applicable toCCS4.0And above versions of the development environment, XDS100V3The emulator is only suitable forCCS5.2And above versions of the development environment.

Note: Before installation, turn off anti-virus software and360, PC Manager and other security protection software, otherwise a warning will appear when clicking the installation program, and forced installation will result in file loss. Double-clickccs_setup_6.2.0.00050.exeThe file appears as shown below1-2The interface shown:

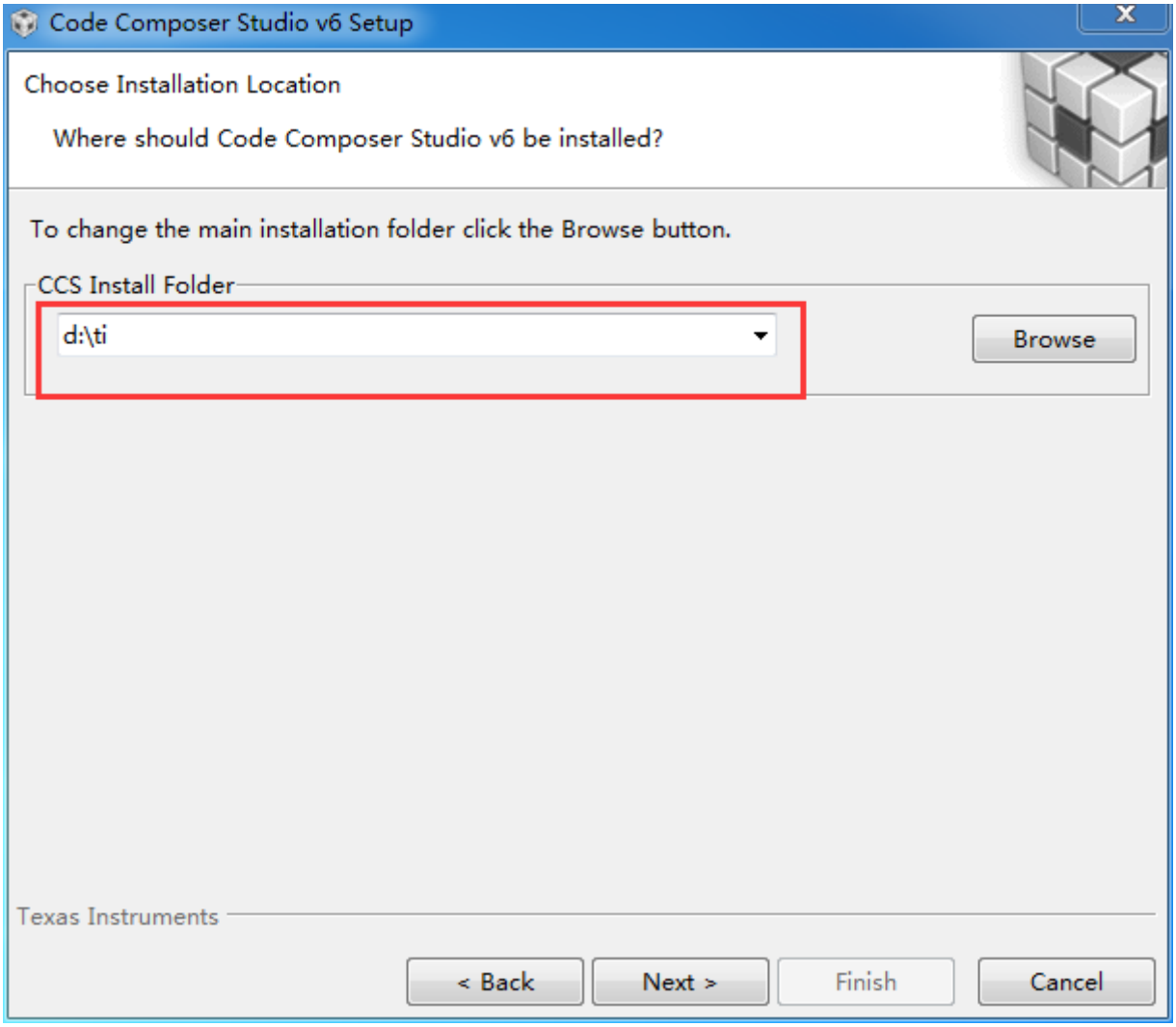


picture1-1



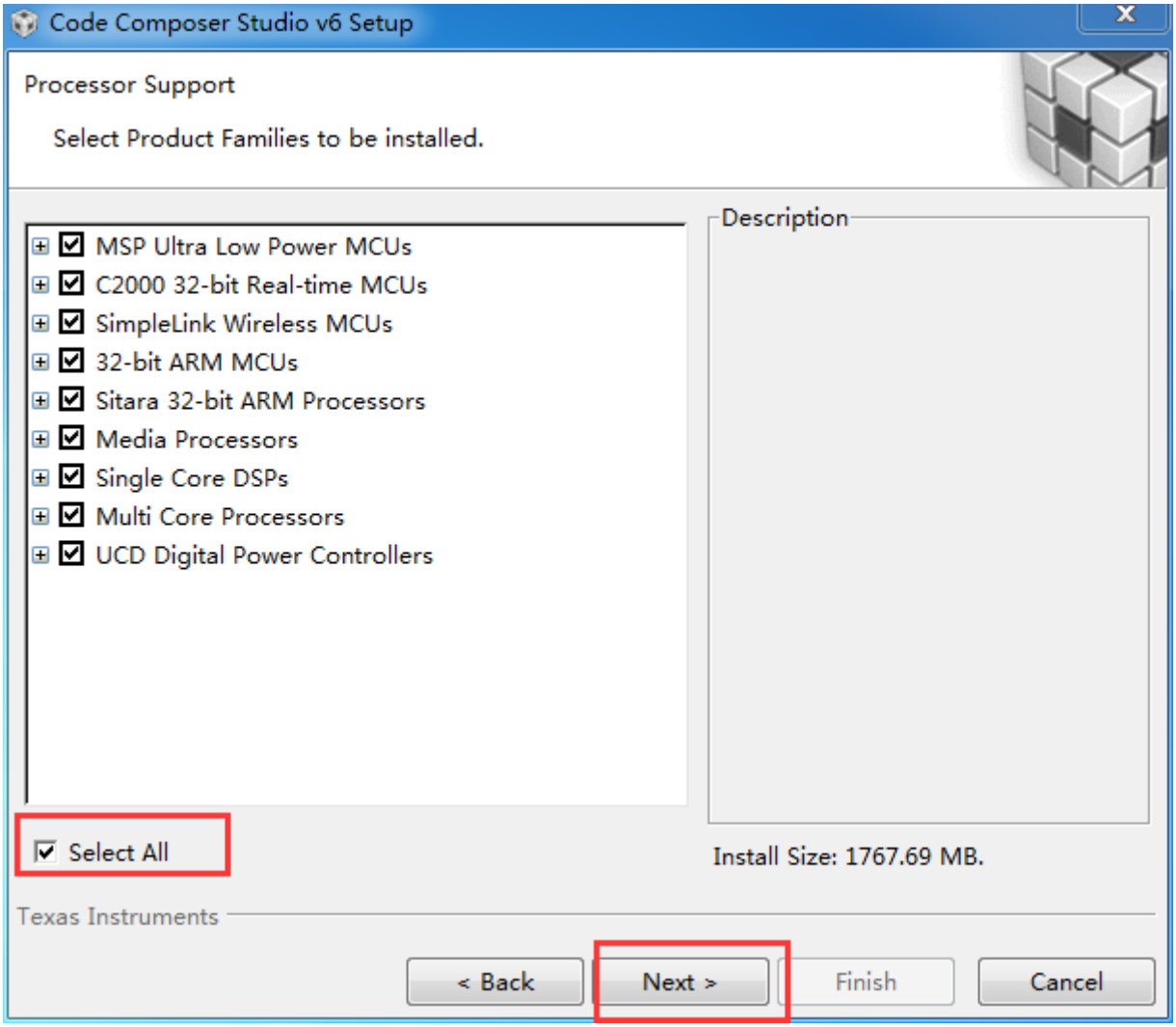
picture1-2

choose"I accept the terms of the license agreement”, clickNext” As shown below1-3



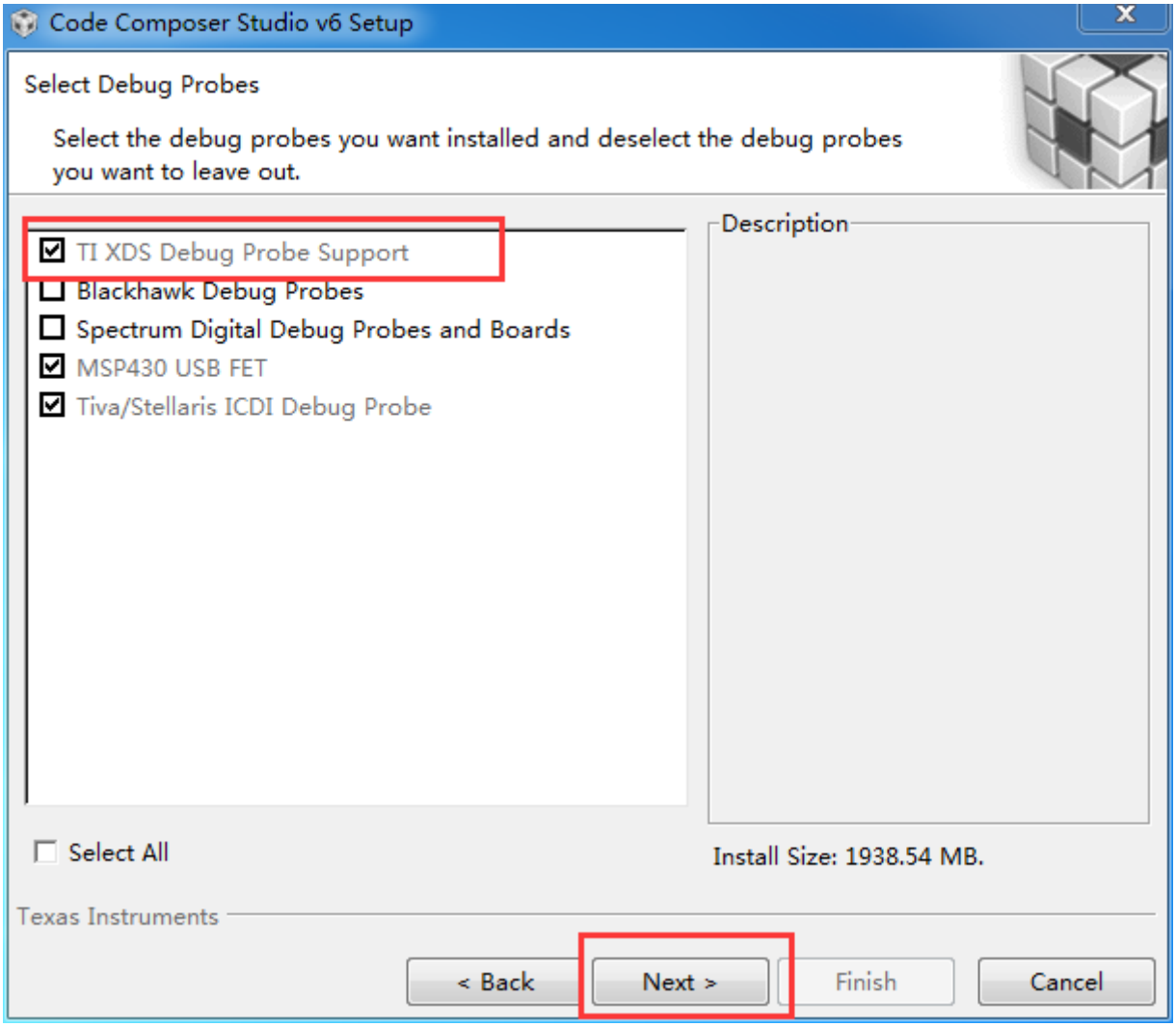
picture1-3

Click "Browse”Select the installation path (note: the path cannot contain Chinese characters), but the default path is recommended.D:\tiClick "Next”See the figure below1-4As shown:



picture1-4

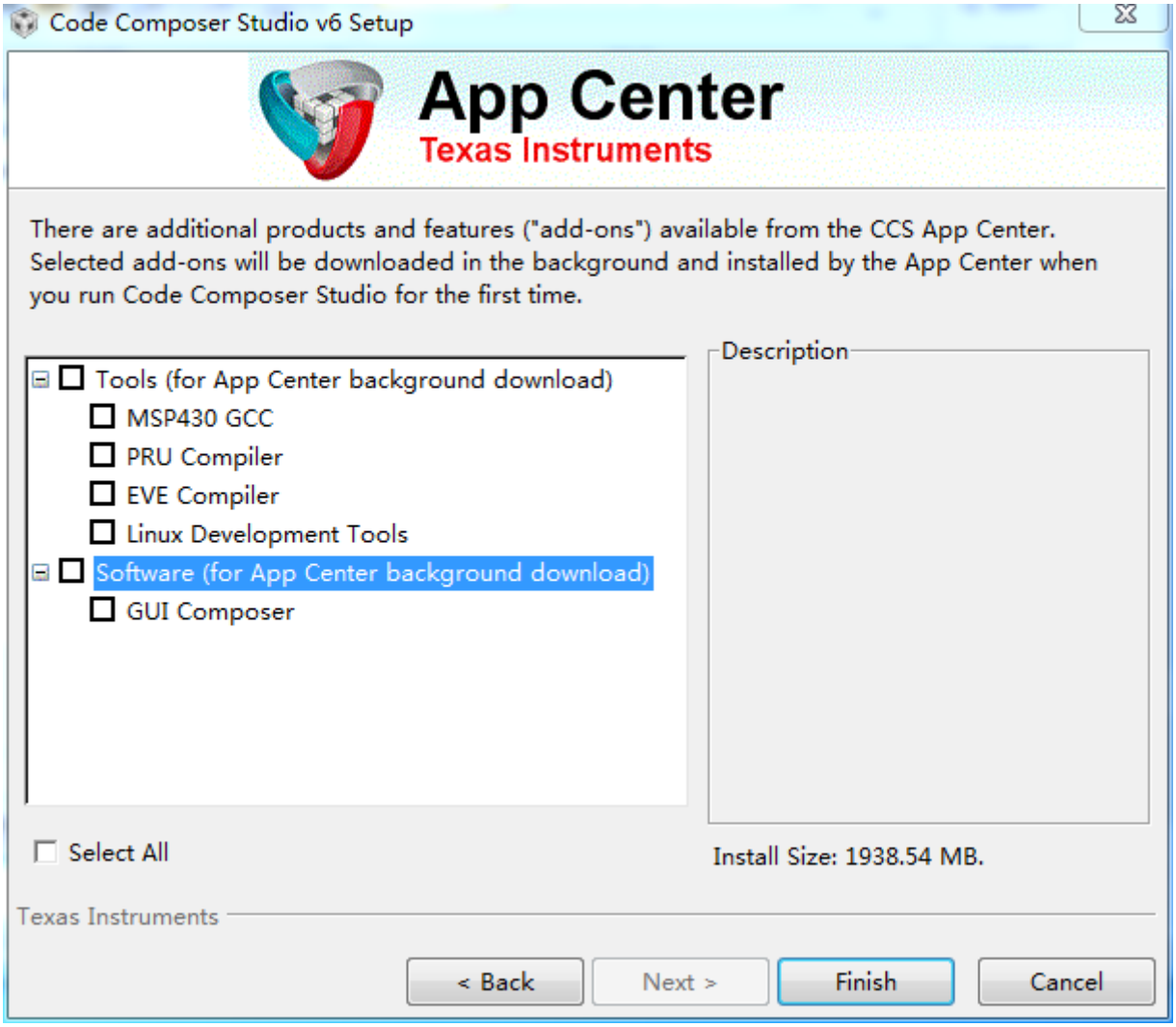
Select the content to be installed according to your needs. Select "Select All”, then clickNext” See the figure below1-5Shown



picture1-5

Still choose the simulation device driver type according to your needs, here selectTI XDS Debug Probe Support, then click “Next”

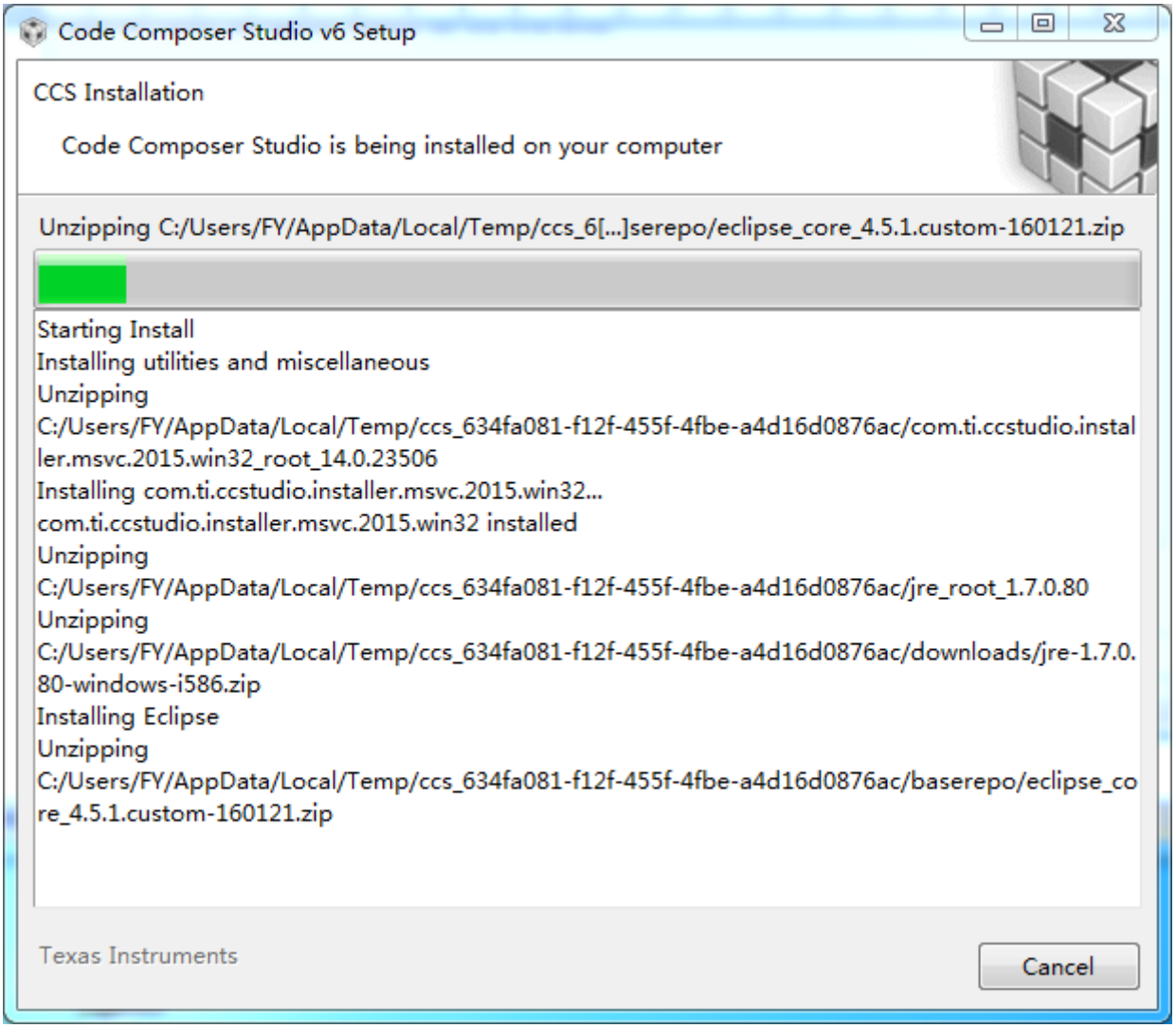
See the figure below1-6As shown:



picture1-6

Select according to your needs, unselect all here, and then click "Finish" During the installation process, the following figure will pop up1-7Do not click

the dialog box that displays some installation functions.Cancelbutton, otherwise this feature will not be installed during the installation process.



picture1-7

After the installation is complete, click "Finish" as shown in Figure 1-8 below:

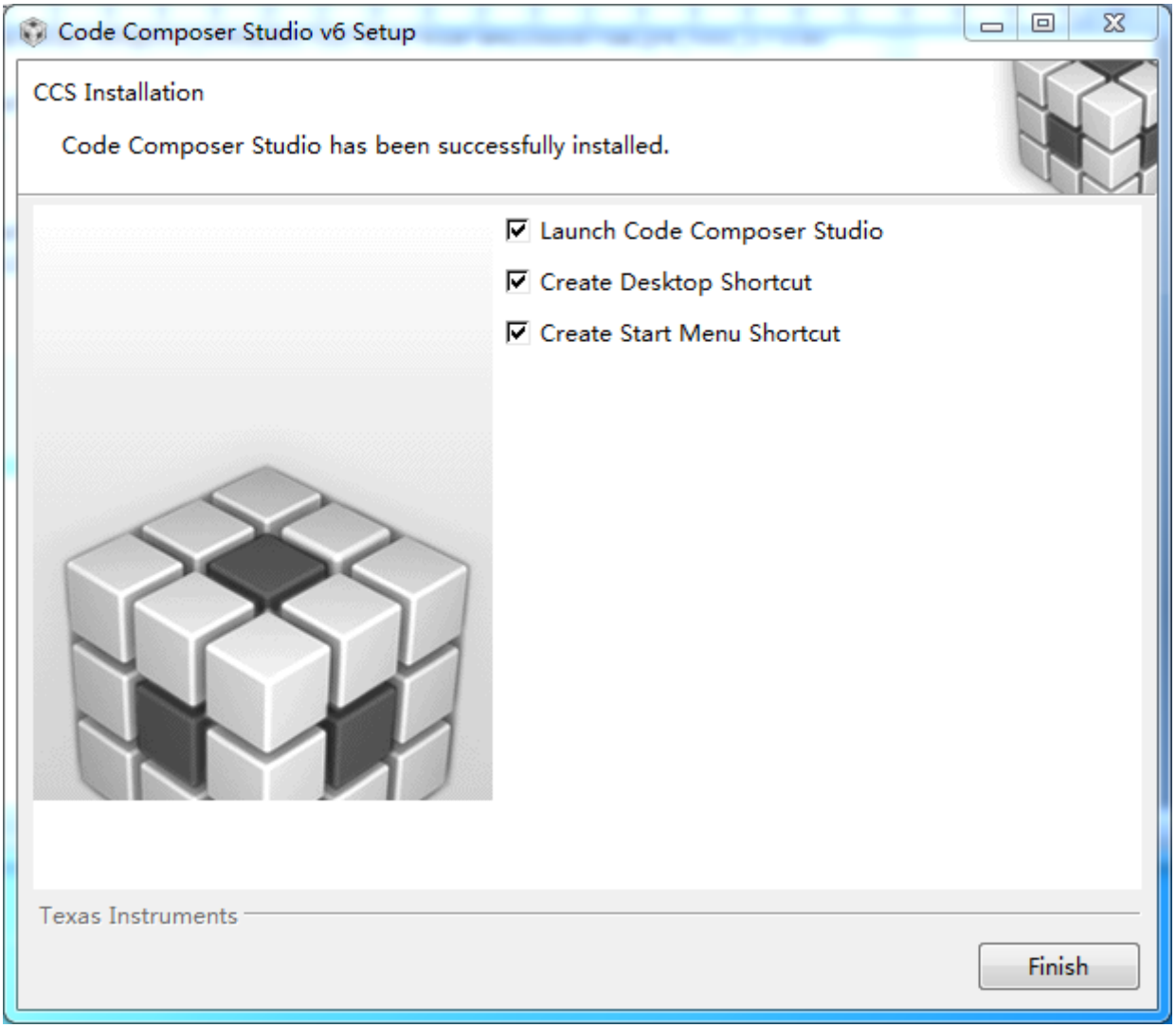


Figure 1-8

Chapter 2CCSConnection to the emulator

becauseXDS100 V3The emulator driver has already done thisCCSV6So when the user installsCCSV6After the version of the software is installed, the emulator driver has been installed. Next, the user only needs toUSBand PCMachineUSBThe driver is also automatically recognized and installed. When it prompts that the driver is installed and can be used, the user can use the emulator to simulate the target board.

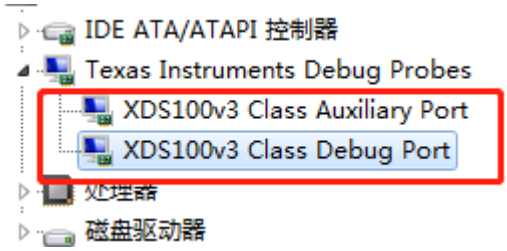
Next we willTMS320F28035Take the development board as an example to explain how to use itXDS100 V3 Simulate the target board.DSP, the operation process is the same.

In the device manager, you can see the following:

The computer will automatically install the driver.

The following two drivers are displayed as normal. (You can see it in the device manager) **The old driver shows:**

New driver display:



2.1 Define the workspace directory

The first thing CCSv6 requires is the definition of a workspace, a directory that holds all the elements used in the development process (projects and links to projects, and possibly source code).

By default, the workspace will be created in C:\Users\<user>\Documents or C:\Documents and Settings\<user>\My Documents, but you can choose any location. Each execution of CCSv6 will ask for the workspace directory. If you plan to use one directory for all projects, just check the "Use this as the default and do not ask again" option, as shown in Figure 2-1-1

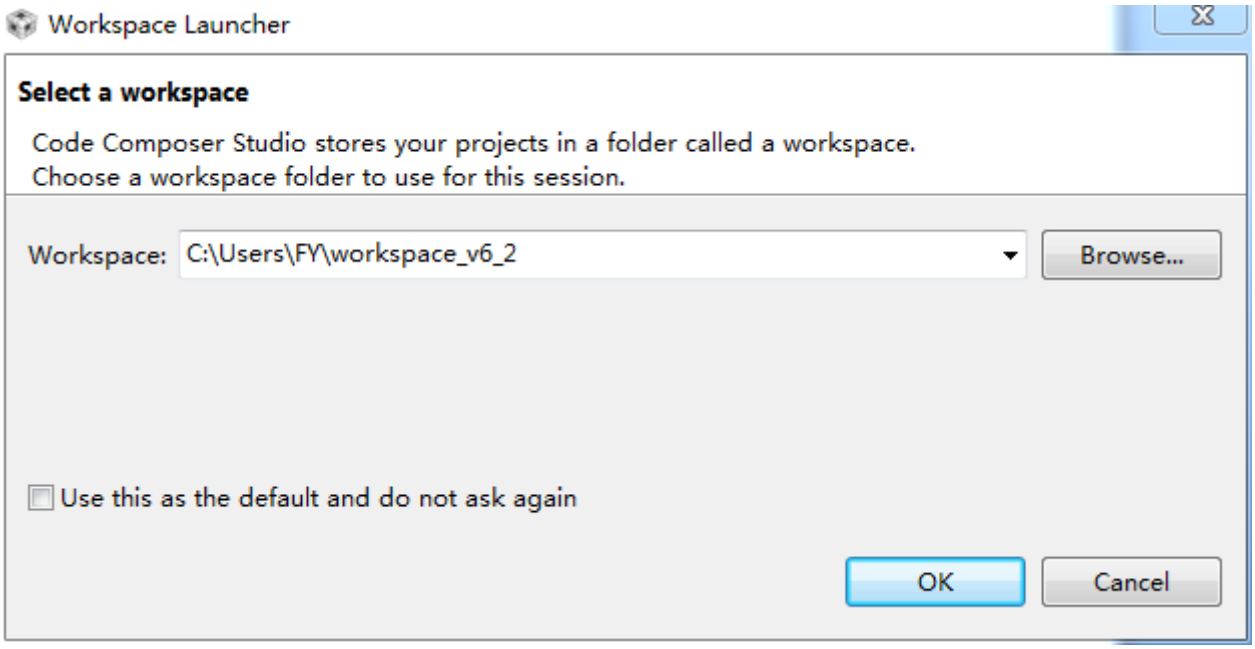
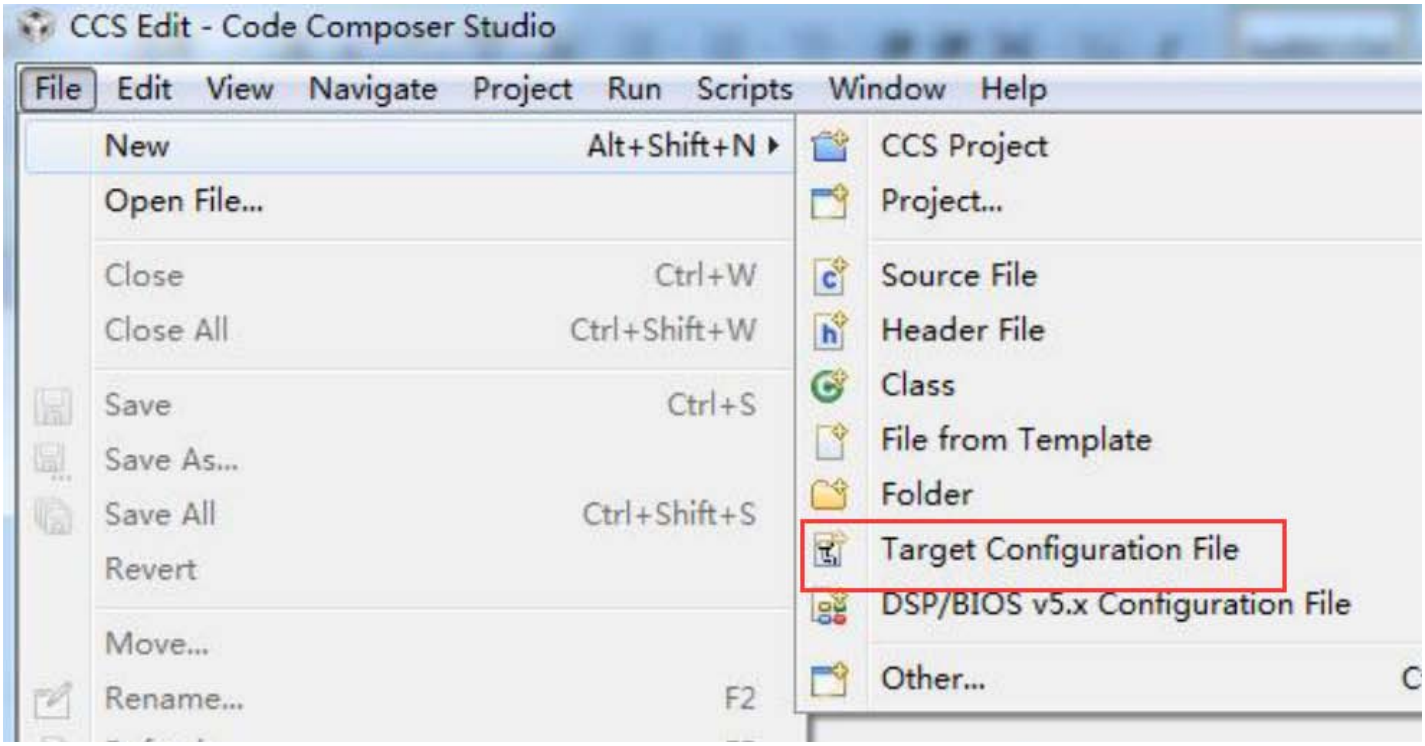


Figure 2-1-1

2.2Establish target board configuration environment

If usedCCSUsers of lower versions are not unfamiliar with this.CCSv6In the version environment, it is also necessary to establish a simulation configuration environment, but the human-machine interface is different.CCSIn lower versions, useCCS SETUPTo establish, I will not introduce it here. The following mainly introduces how toCCSv6Established in:

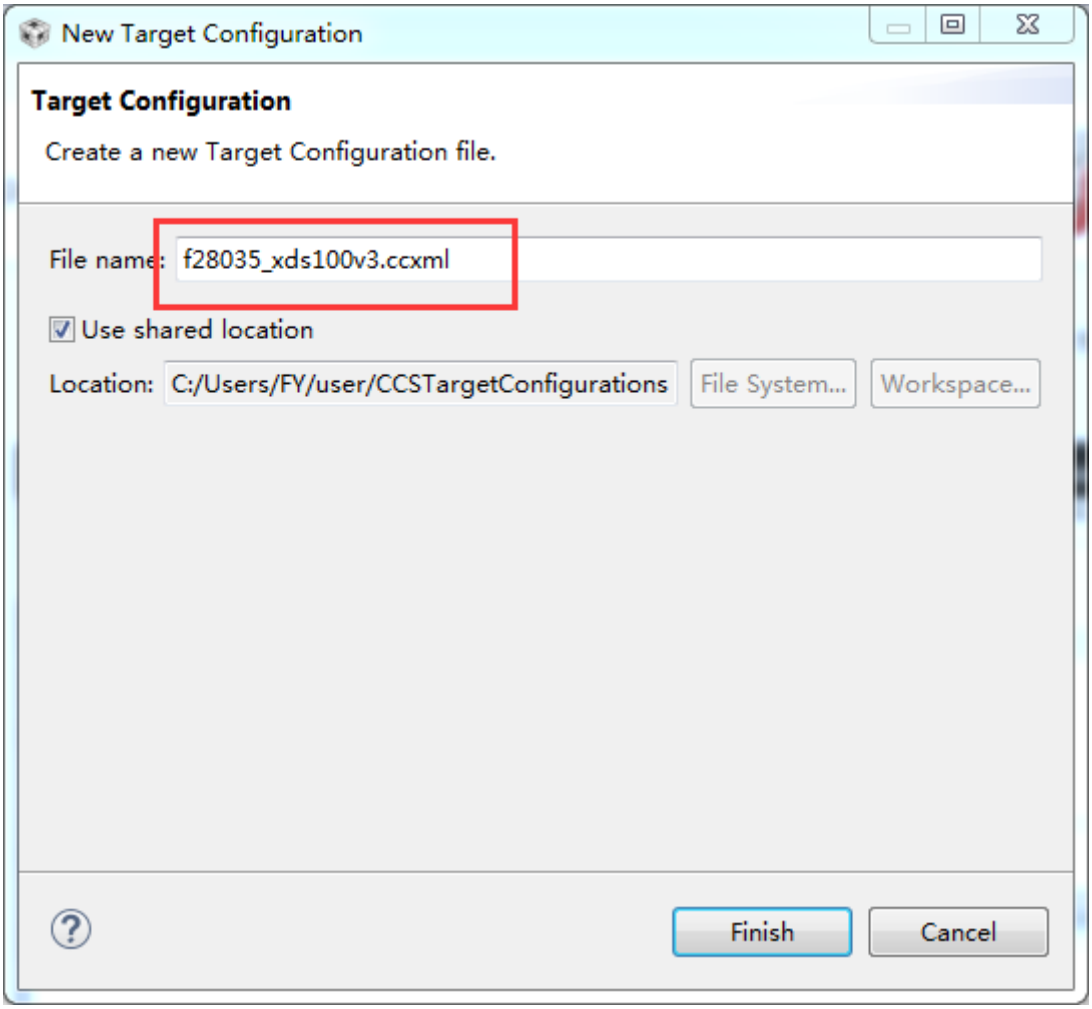
In the toolbar, clickFile ->New->Target Configuration File”, as shown below2-2-1As shown:



picture2-2-1

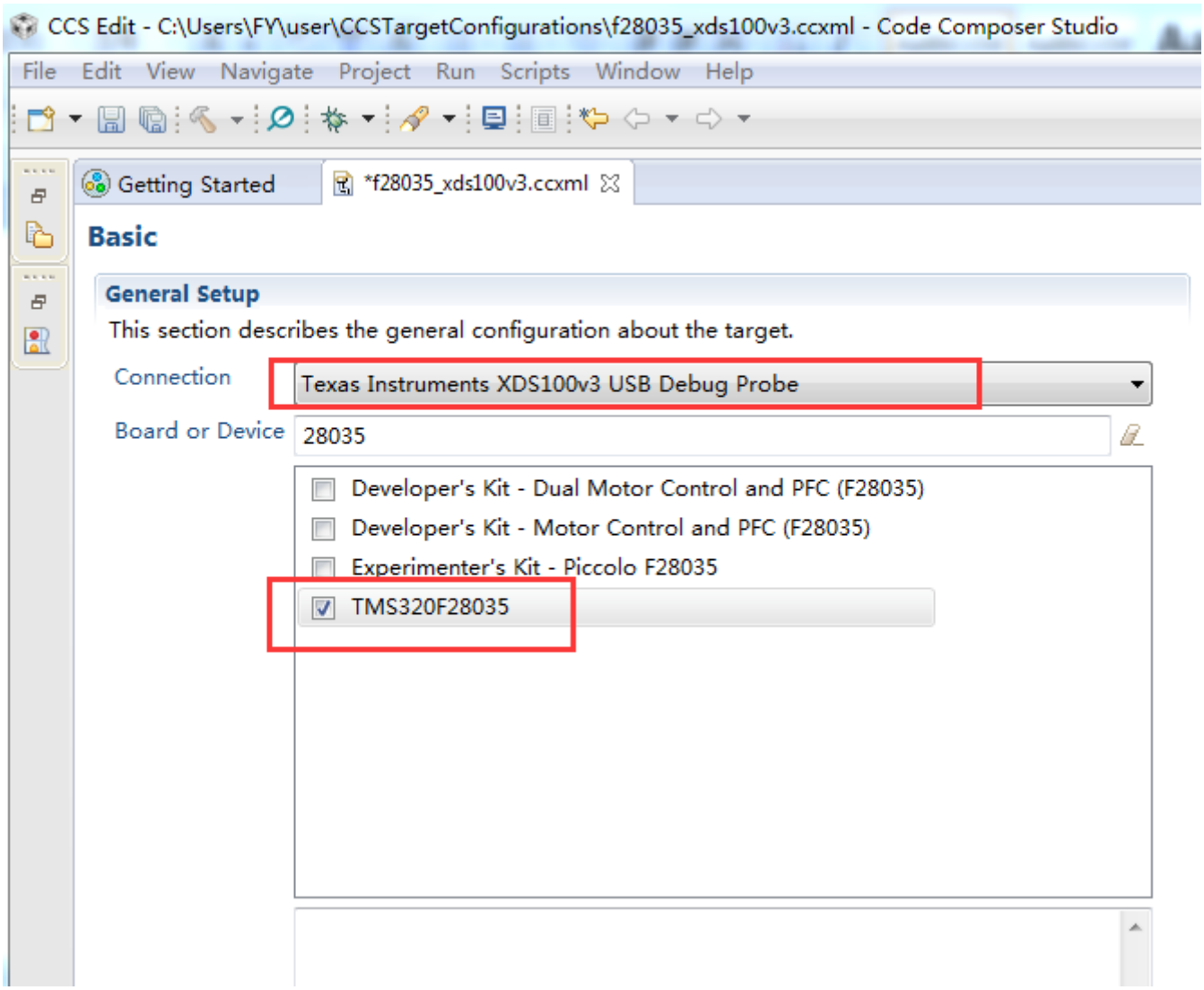
Name this configuration "f28035_xds100v3.ccxml"(You can name it yourself), click "Finsh”,

As shown below2-2-2As shown:



picture2-2-2

exist"connectionIn the "Debugger Type" column, select "Texas Instruments XDS100v3 USB Emulator","DeviceIn the "(Chip Type)" column, select "TMS320F28035", as shown below2-2-3Shown:



picture2-2-3

If you use the defaultGELfile, then clickSave” As shown in Figure 2-2-4

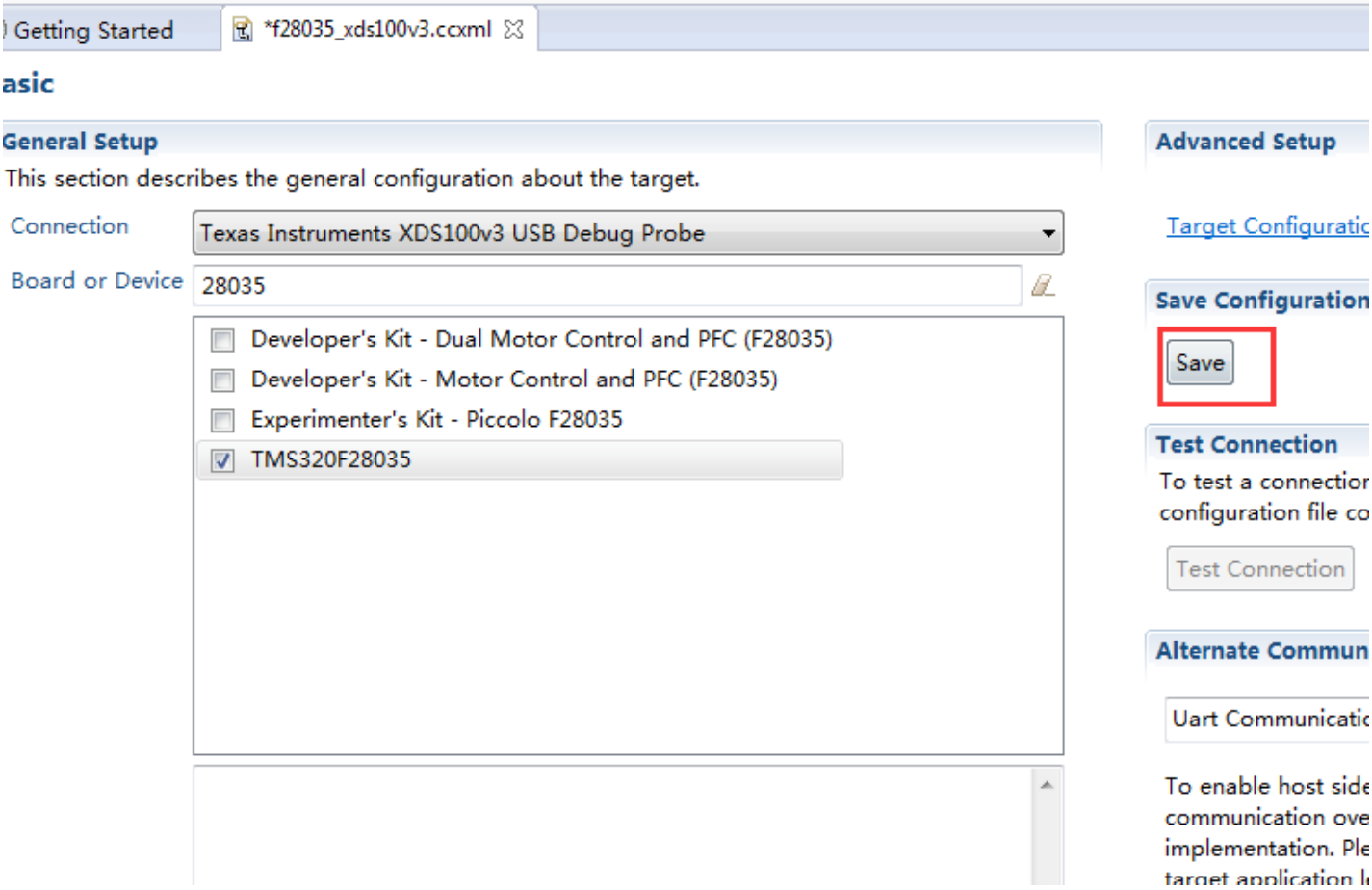


Figure 2-2-4

So far, this configuration environment has been established.

2.3 Connecting to the target board

Find the "View->Target configurations" button on the toolbar and click it to switch to the configuration interface. Then right-click the configuration file in the configuration interface and select "Set as Default" to set the newly created configuration file as the default state. Start debugging, right-click and select "Launch Selected Configuration" of the configured project. Detailed operations are shown in Figure 2-3-1 below:

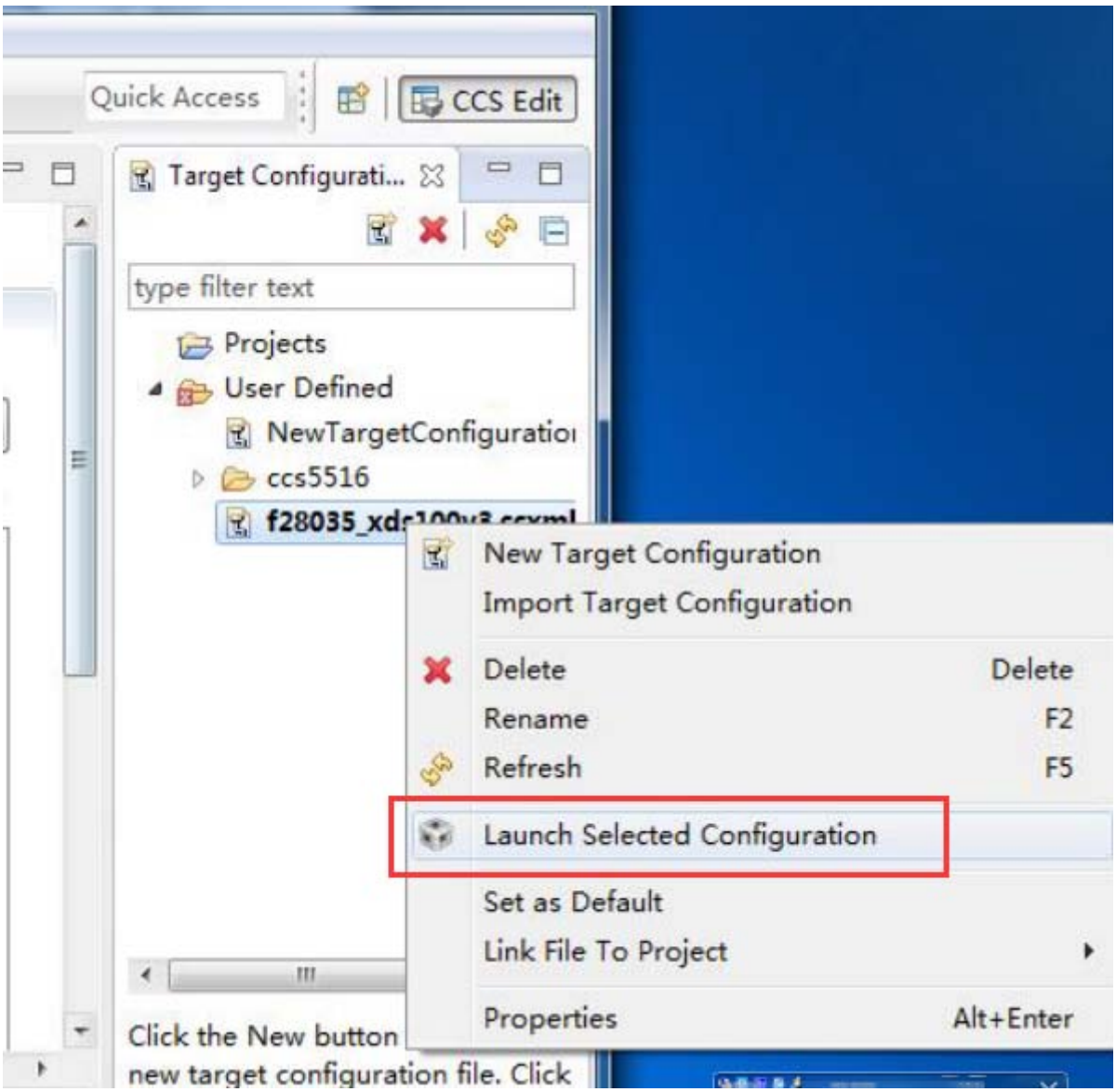


Figure 2-3-1

After successful startup, click "Run->Connect Target" in the toolbar to connect to the target board, as shown in Figure 2-3-2 below:

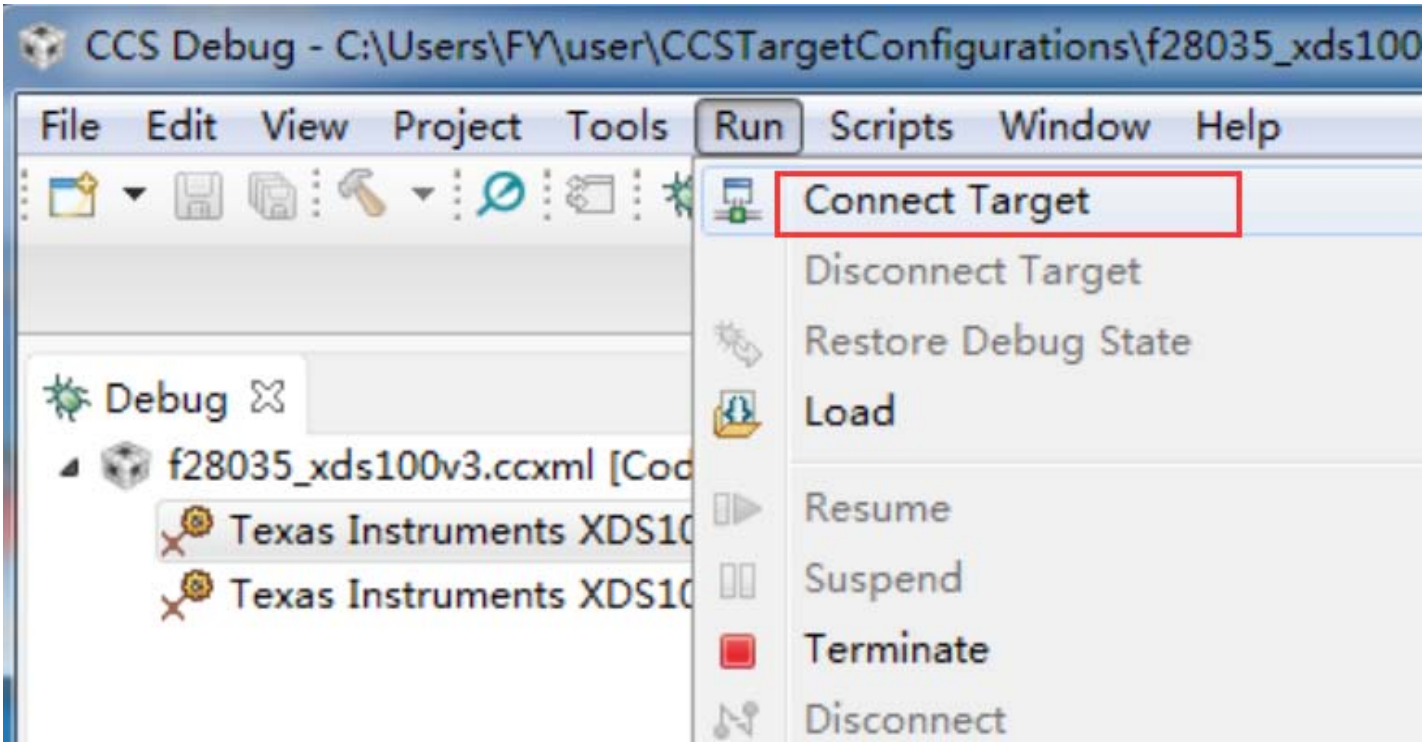


Figure 2-3-2

The interface shown in Figure 2-3-3 below appears, indicating that the connection is successful. Next, the user can conduct simulation experiments by himself.

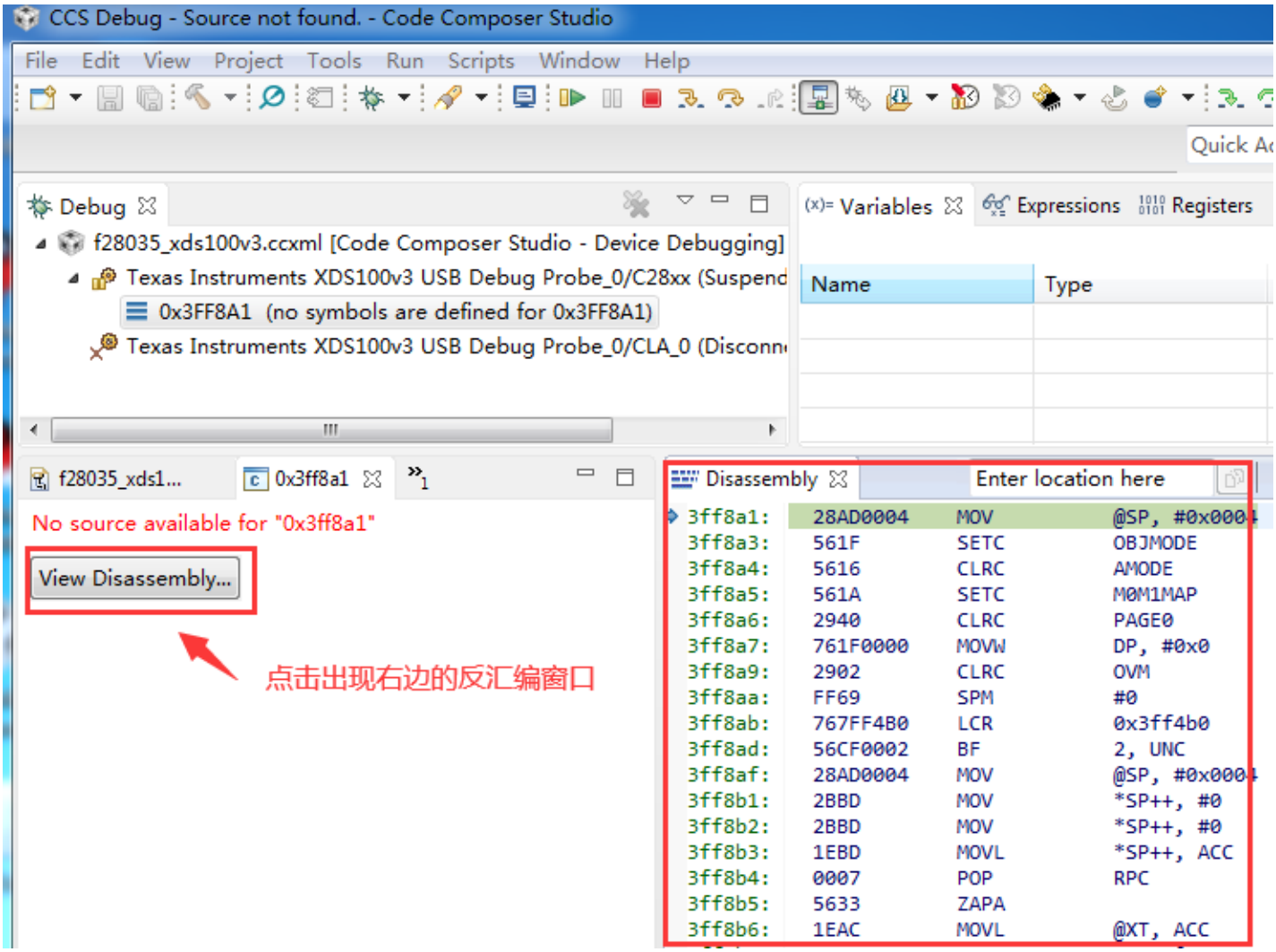
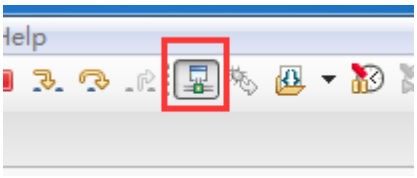


Figure 2-3-3



Connect or disconnect shortcut keys

Chapter 3 CreationCCS6.0project

3.1 Create a project

After the Welcome screen closes, the following workspace will be displayed, and you can now create a new project. Go to the menu "File -> New -> CCS Project" as shown in Figure 3-1 below:

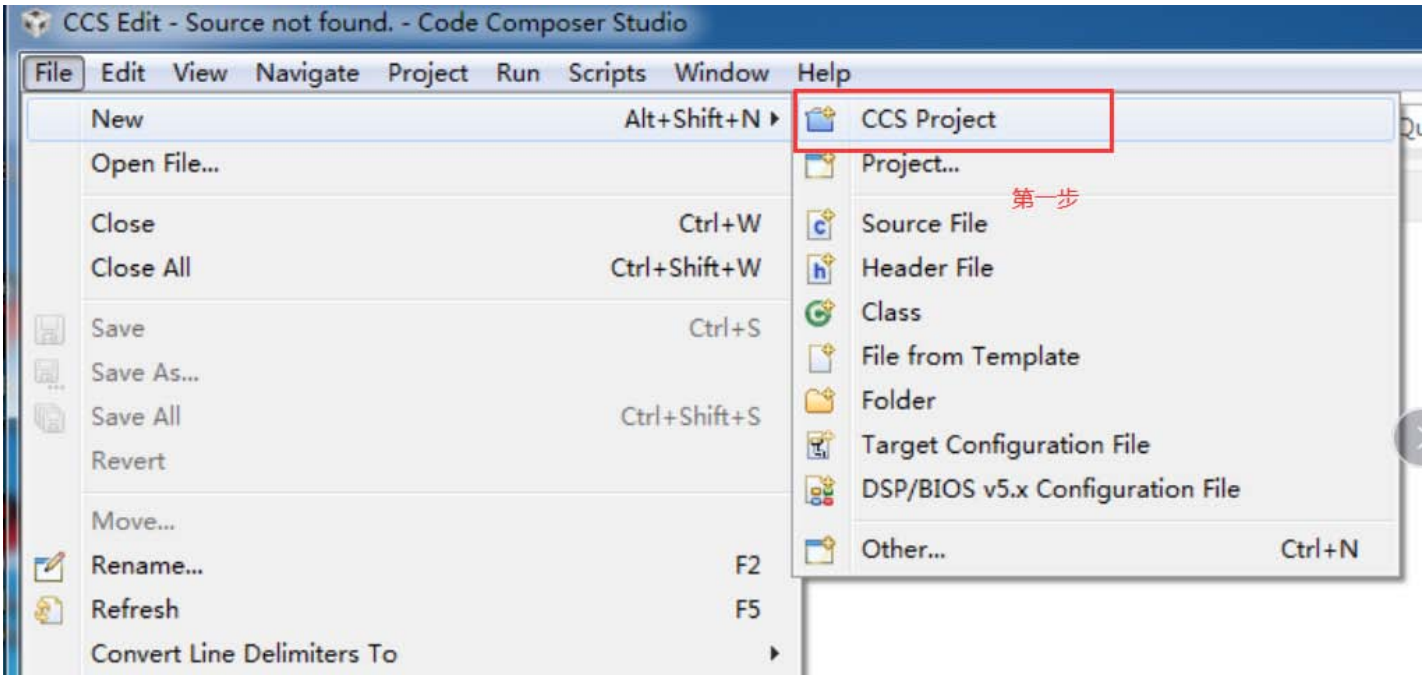


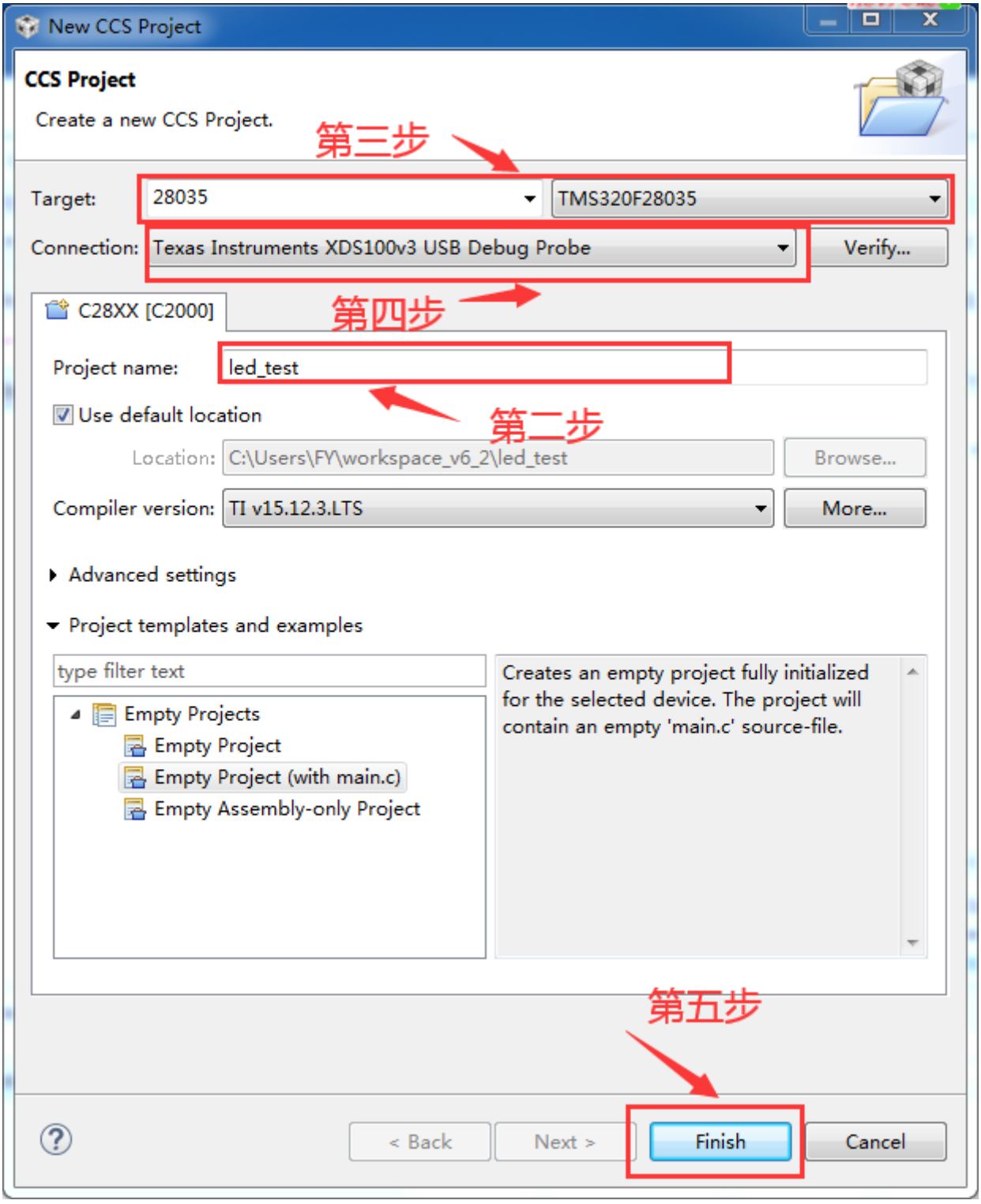
Figure 3-1

② In "Project Name(Project Name) field, type a name for the new project. If you selectUse default locationUse default location" option (enabled by default) will create the project in the workspace folder. Uncheck this option to choose a new location (usingBrowse...(Browse...)" button). Name the project " led_test" .

③ In "Target" Select the chip type to be used from the menu.

④ In " "Connection" Select Debugger.

⑤ Click "Finish(Complete)" to create the project., as shown below3-2As shown:



picture3-2

⑥ To create files for the project,C/C++ Projects（C/C++Project)" (Path: Toolbar Window->Show View->OtherNextC/C++->C/C++ Projects) view and selectNew -> Source File(New->Source File)". In the text box that opens, type a file with a valid extension (.c、.C、.cpp、.C++、.asm、.s64、.s55ClickFinish(Finish)".

⑦ To add existing source files to the project,C/C++ Projects（C/C++Right-click the project name in the Projects tab and selectAdd Files to Project(Add files to project)", add the source files

You can also select "Link Files to Project(Link File to Project)" to create a file reference, which keeps the file in its original directory. This is necessary if your source code contains files in a very specific directory structure.

3.2 Generate Project

Once the project has been created and all files have been added or created, you need to build the project. Simply go to the menu "Project -> Build Active Project". The "Rebuild Active Project" option rebuilds all source files and referenced projects. However, if the project is large, this can be a lengthy process.

Note: If build errors are encountered and no executable is created, the console window at the bottom of the screen will display an error or warning message and the debugging session will not start.

Chapter 4 Project Introduction

4.1 Importing CCS high version projects

In the "C/C++ Projects" (path: Window->Show View->C/C++->C/C++ Projects under Other in the toolbar), that is, the "CCS Edit" view, click "Project->Import CCS Projects..." on the toolbar as shown in Figure 4-1-1 below:

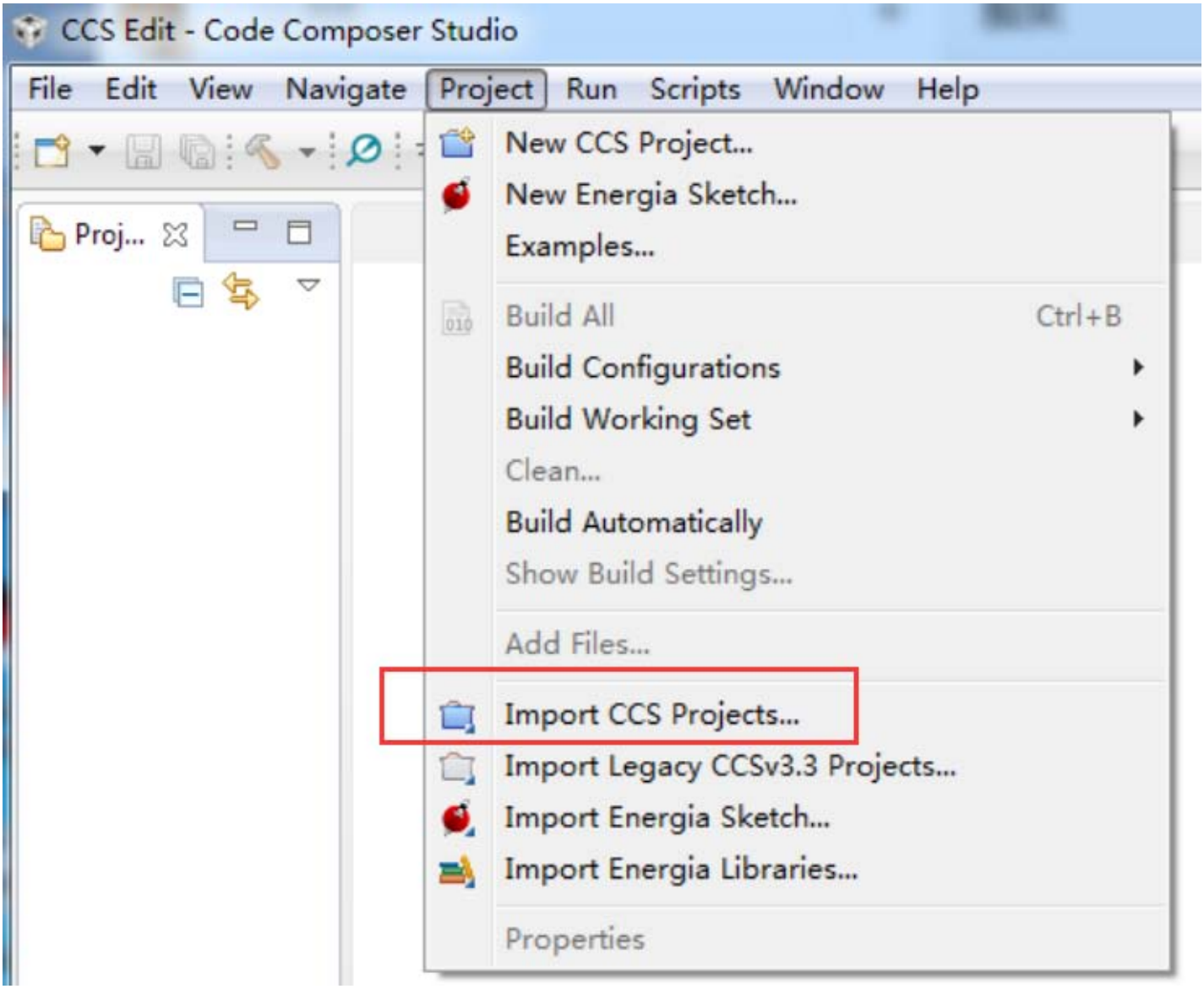


Figure 4-1-1

In the pop-up window, perform the following operations as shown in Figure 4-1-2:

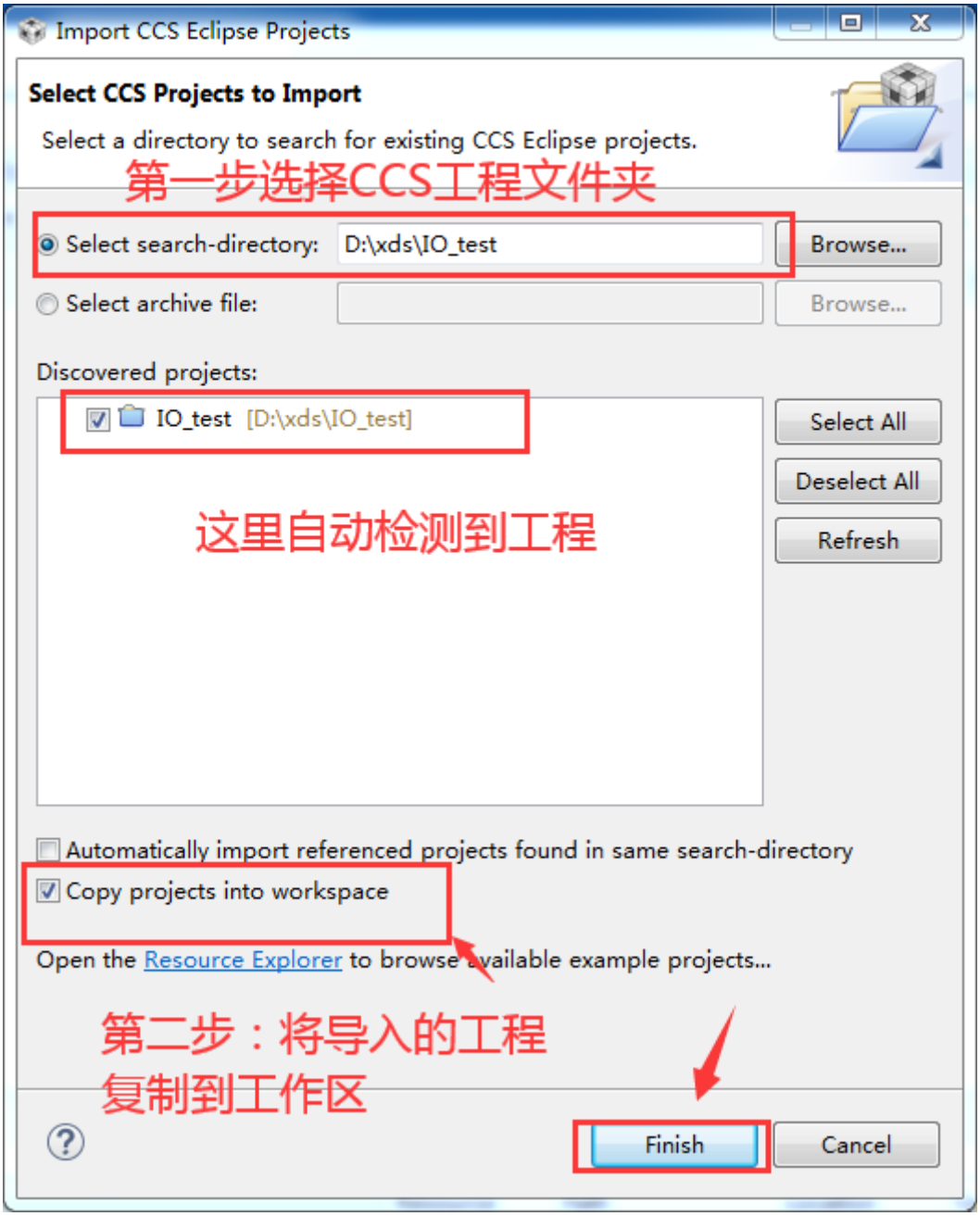


Figure 4-1-2

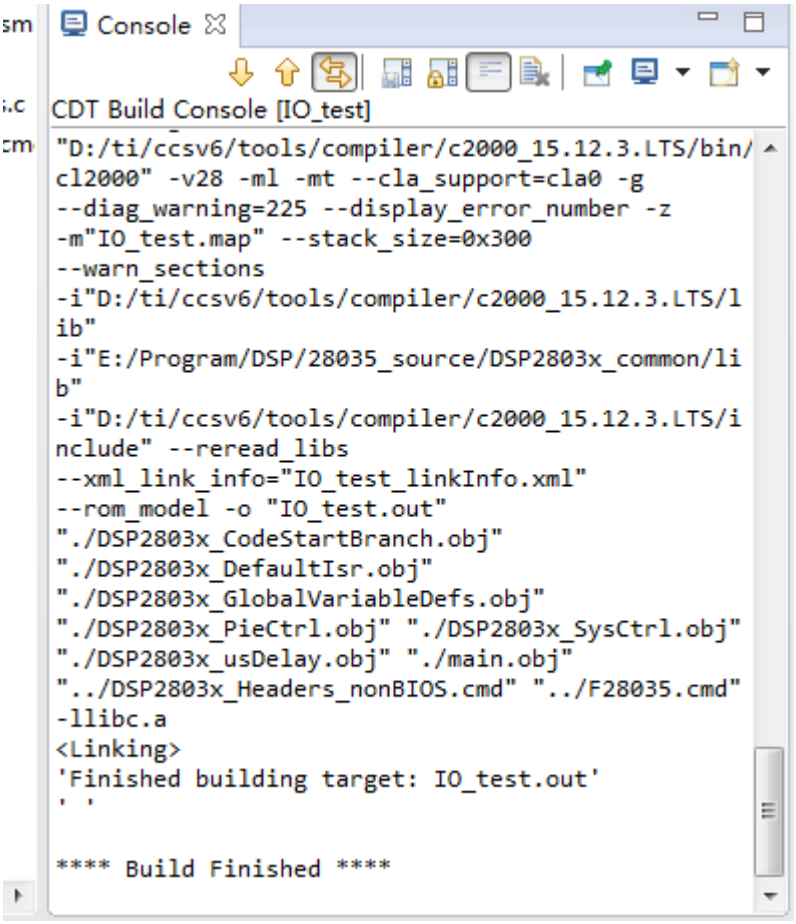
After completion, right-click the project and select "Build Project" to compile. After the compilation is completed, the entire import process is completed.

Chapter 5 Simulation Debugging of CCS6.0

5.1 Simulation Operation of CCS6.2

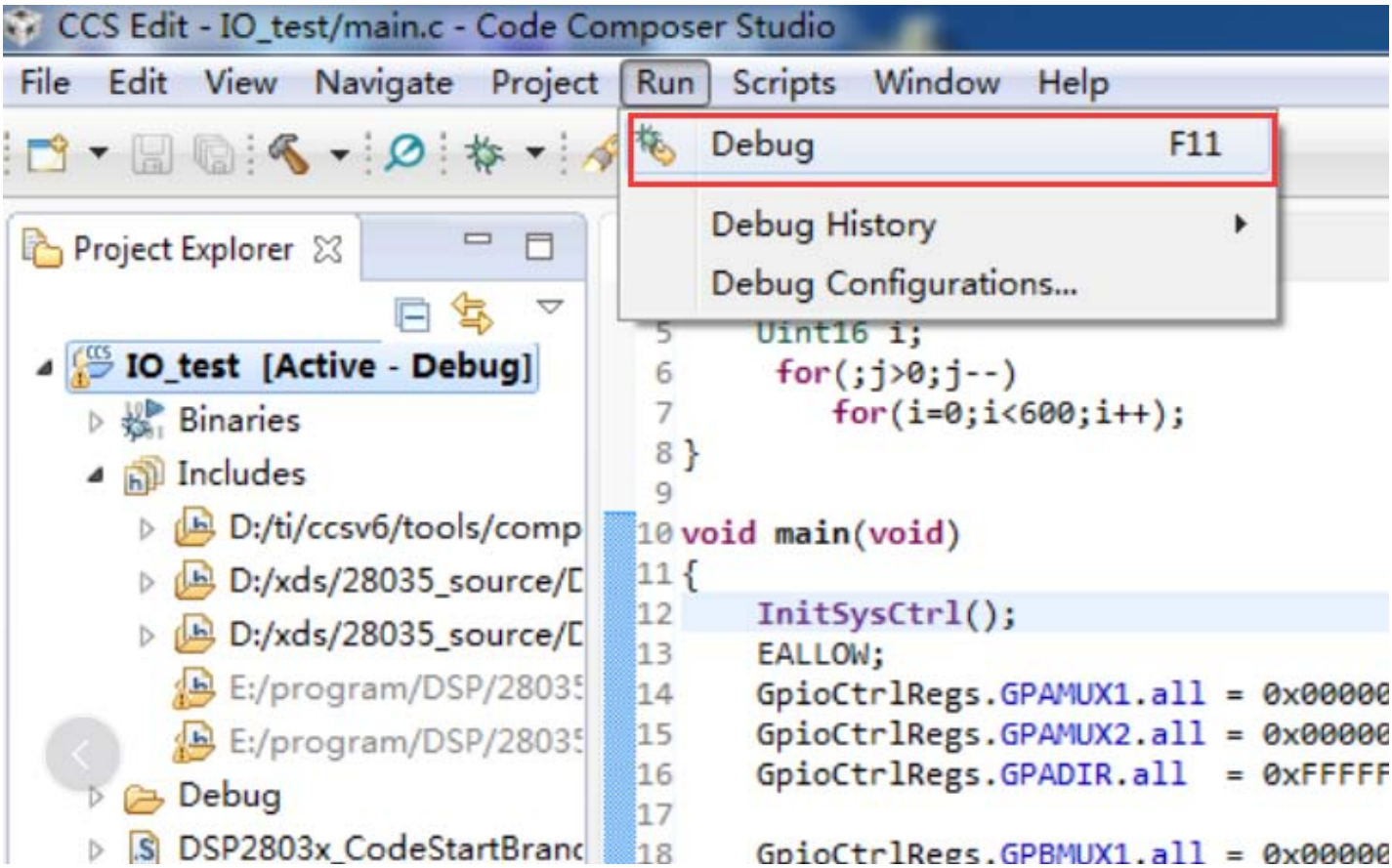
After the routine is imported, it is time to debug CCS and the development board. Here, the TMS320F28035 chip is taken as an example for simulation and burning test.

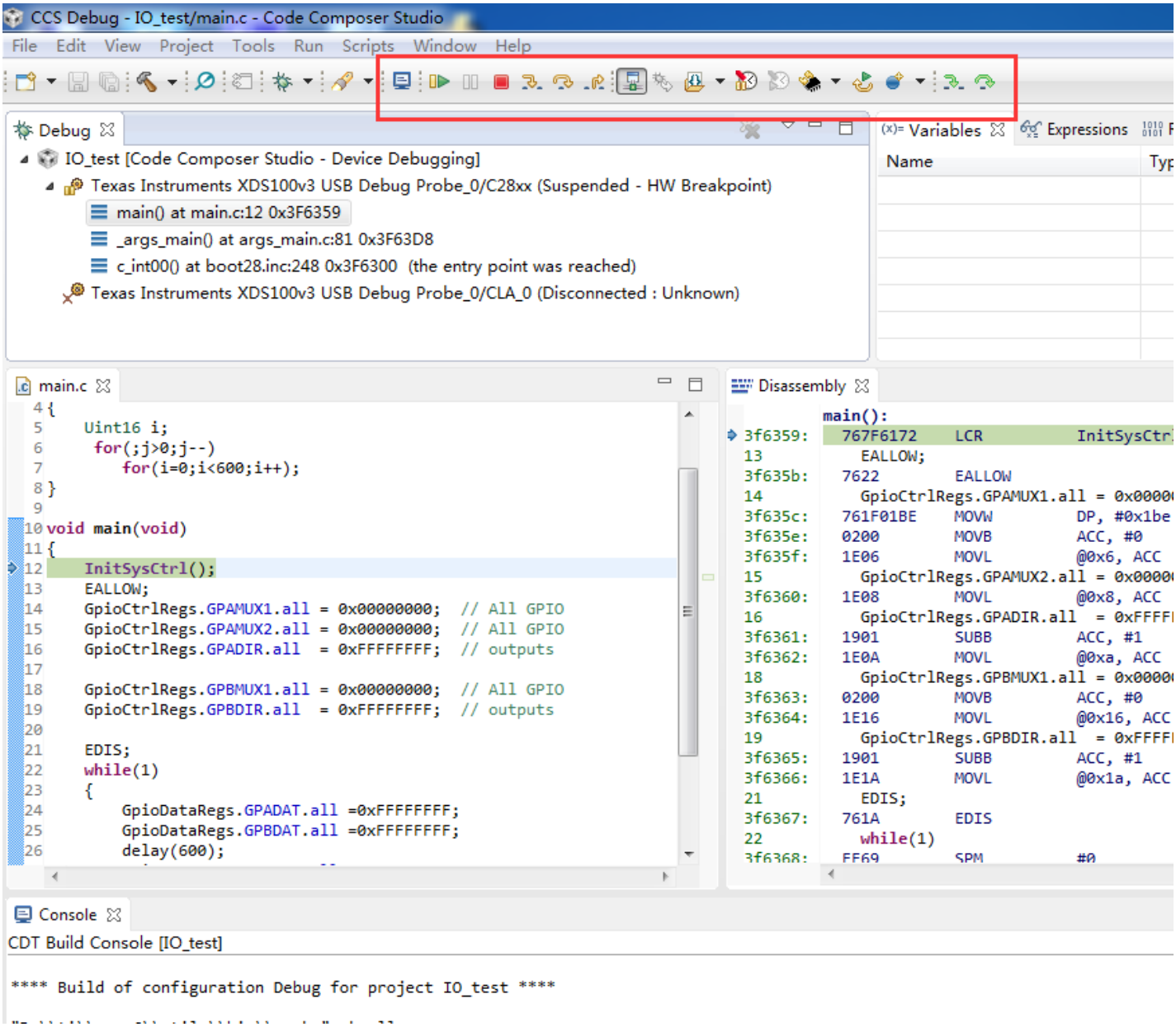
After importing the project above, compile directly and you can see that the io_test.out file has been generated.



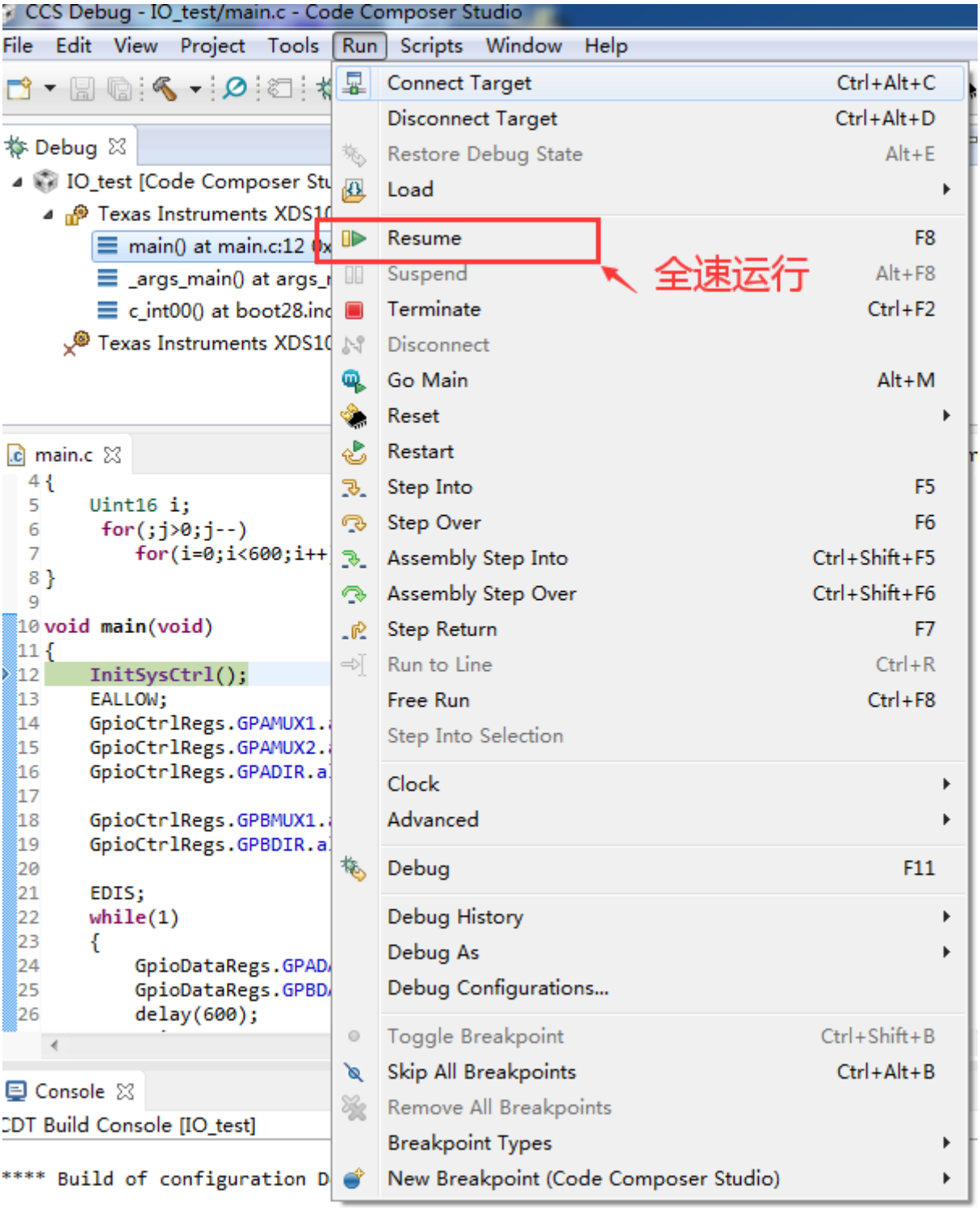
You can see that it compiled successfully.

pointdebugOr pressF11You can enter the simulation debugging state


CCS Edit - IO_test/main.c - Code Composer Studio
File Edit View Navigate Project Run Scripts Window Help
Debug F11
Debug History
Debug Configurations...
Project Explorer
IO_test [Active - Debug]
Binaries
Includes
D:/ti/ccsv6/tools/comp
D:/xds/28035_source/D
D:/xds/28035_source/D
E:/program/DSP/2803
E:/program/DSP/2803
Debug
DSP2803x_CodeStartBranc
Uint16 i;
for(;j>0;j--)
for(i=0;i<600;i++);
}
void main(void)
{
InitSysCtrl();
EALLOW;
GpioCtrlRegs.GPAMUX1.all = 0x00000
GpioCtrlRegs.GPAMUX2.all = 0x00000
GpioCtrlRegs.GPADIR.all = 0xFFFFF
GpioCtrlRegs.GPBMUX1.all = 0x00000



From the above you can see a series of debugging operation buttons


CCS Debug - IO_test/main.c - Code Composer Studio
File Edit View Project Tools Run Scripts Window Help
Debug
IO_test [Code Composer Stu
Texas Instruments XDS10
main() at main.c:12
_args_main() at args_
c_int00() at boot28.inc
Texas Instruments XDS10
main.c
4 {
5 Uint16 i;
6 for(;j>0;j--)
7 for(i=0;i<600;i++
8 }
9
10 void main(void)
11 {
12 InitSysCtrl();
13 EALLOW;
14 GpioCtrlRegs.GPAMUX1.
15 GpioCtrlRegs.GPAMUX2.
16 GpioCtrlRegs.GPADIR.a
17
18 GpioCtrlRegs.GPBMUX1.
19 GpioCtrlRegs.GPBDIR.a
20
21 EDIS;
22 while(1)
23 {
24 GpioDataRegs.GPAD
25 GpioDataRegs.GPBD
26 delay(600);
Console
CDT Build Console [IO_test]
**** Build of configuration D
Connect Target Ctrl+Alt+C
Disconnect Target Ctrl+Alt+D
Restore Debug State Alt+E
Load
Resume F8
Suspend Alt+F8
Terminate Ctrl+F2
Disconnect
Go Main Alt+M
Reset
Restart
Step Into F5
Step Over F6
Assembly Step Into Ctrl+Shift+F5
Assembly Step Over Ctrl+Shift+F6
Step Return F7
Run to Line Ctrl+R
Free Run Ctrl+F8
Step Into Selection
Clock
Advanced
Debug F11
Debug History
Debug As
Debug Configurations...
Toggle Breakpoint Ctrl+Shift+B
Skip All Breakpoints Ctrl+Alt+B
Remove All Breakpoints
Breakpoint Types
New Breakpoint (Code Composer Studio)
全速运行