

手把手教你使用 Grace 开发 LaunchPad

开发环境：Code Composer Studio v4.2

操作系统：Windows XP SP3 32bit

开发板：TI LaunchPad

Grace 是 TI 推出的图形外设配置工具，用于加速用户对 MSP430 的使用开发，下面简单介绍下如何使用 Grace 对 LaunchPad 的开发。

1、安装 CCS 和 Grace，这两个软件可以到 TI 的官网里下载到，<http://www.ti.com.cn/>。建议使用页面右上角的检索工具，找资料那些挺方便的：



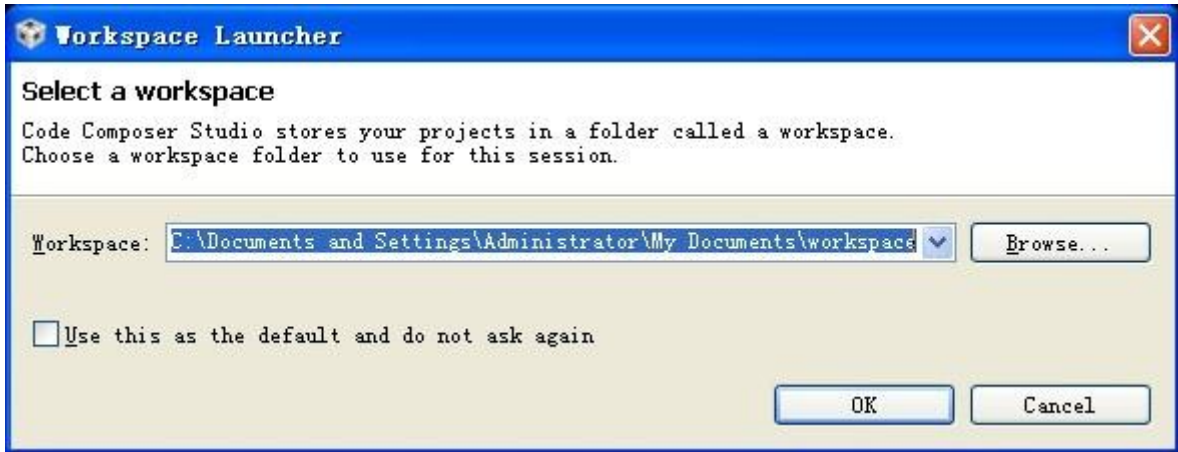
附：CCS 下载地址：

<http://www.ti.com.cn/tool/cn/ccstudio>

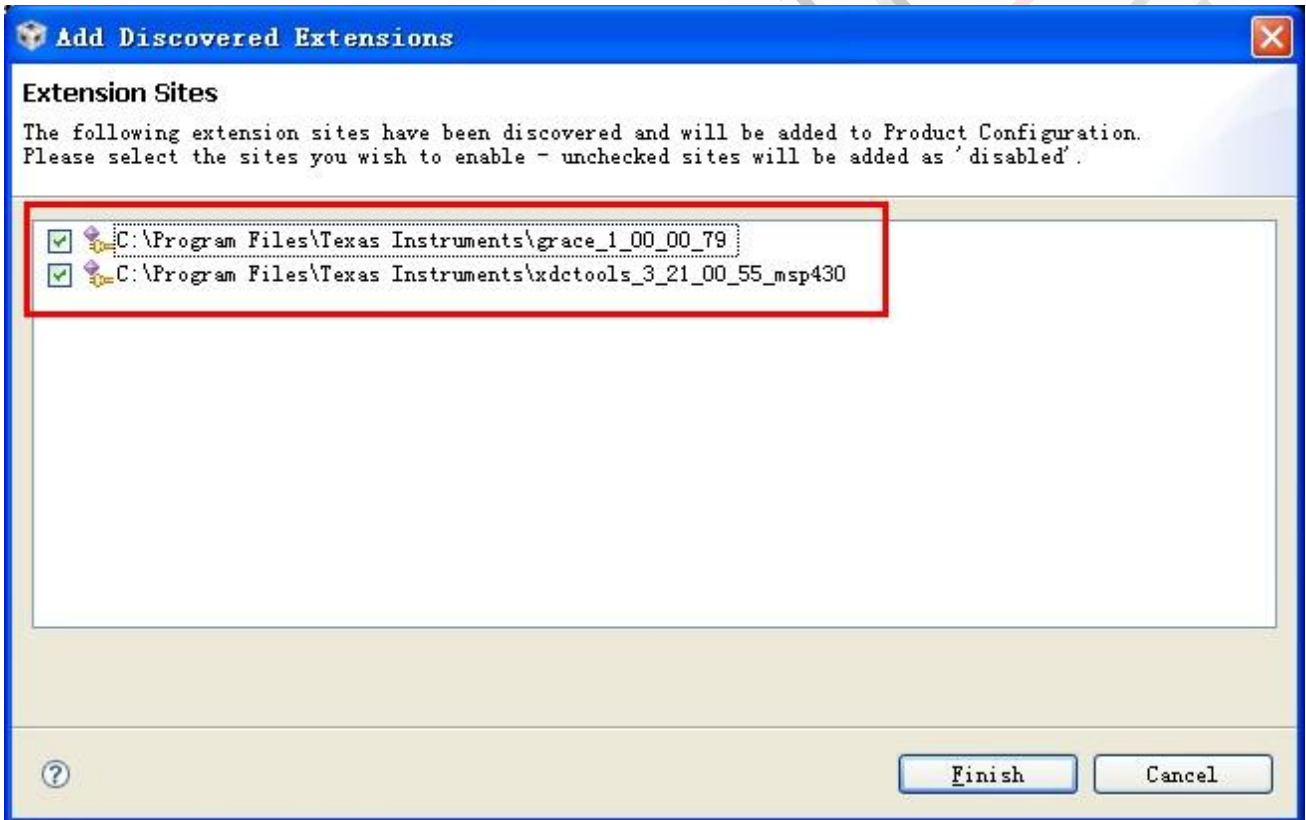
Grace 下载地址：

<http://www.ti.com.cn/tool/cn/grace>

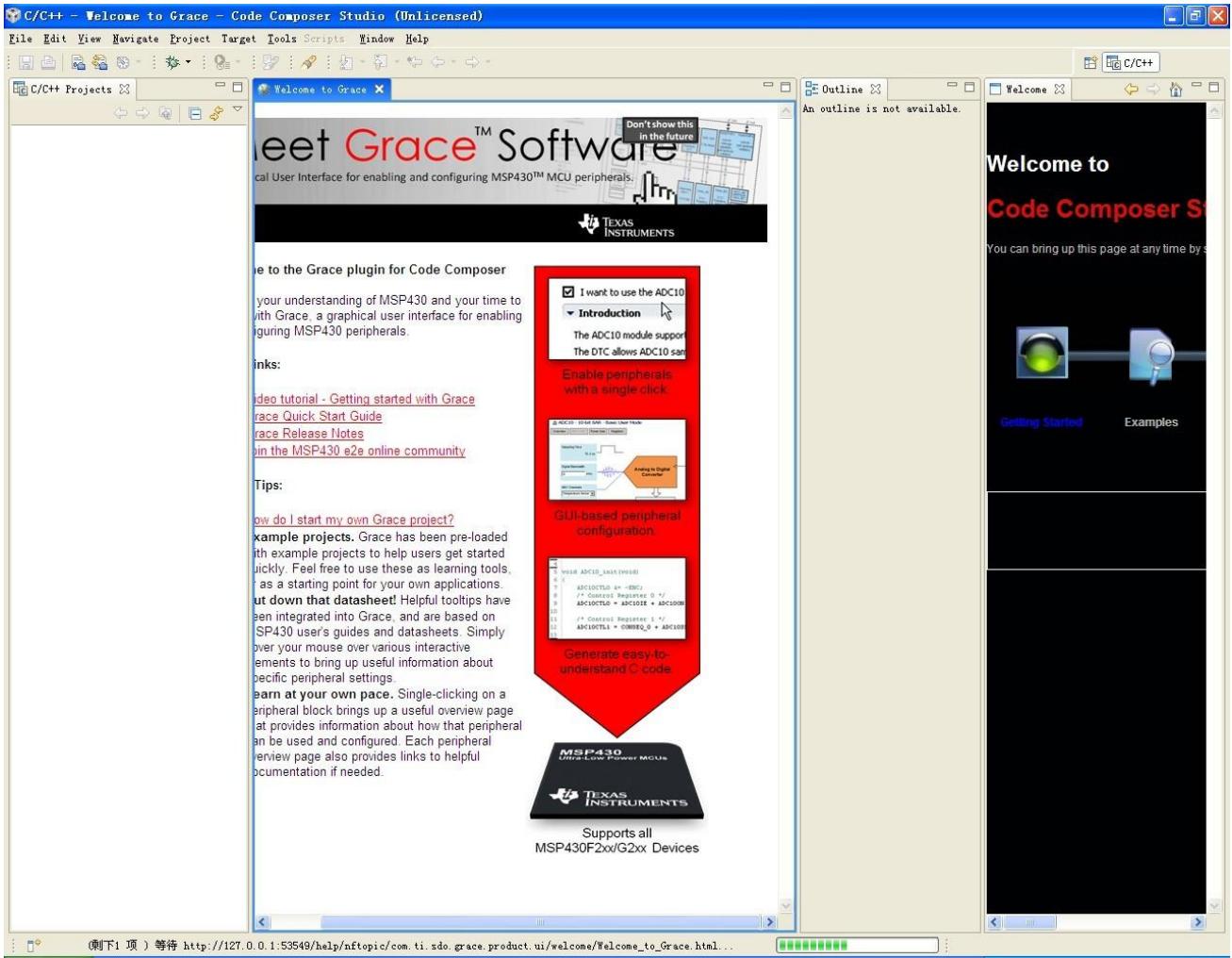
2、安装完成之后，启动 Code Composer Studio v4，第一次使用，会提示 Select a workspace，这个根据自己需要选择合适目录作为工作路径。



3、由于前面安装了 Grace，CCS 会提示安装插件 Add Discovered Extensions，选中后点击 Finish，CCS 会提示需要重新启动。

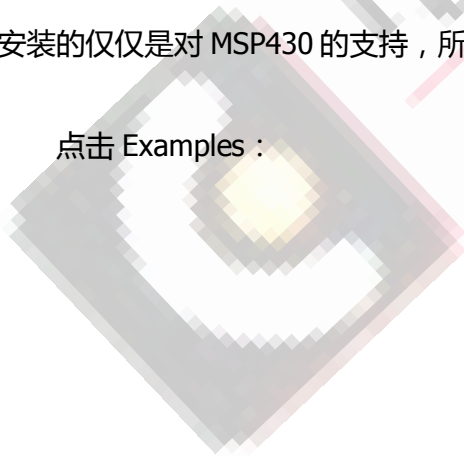


4、CCS 重新启动之后，就看见看到 CCS v4 的华丽界面了。至于弹出来的注册界面，注册方法已是公开的秘密，据说那个 Crack 是对 CCS v4.1 版及后续版本都支持的。



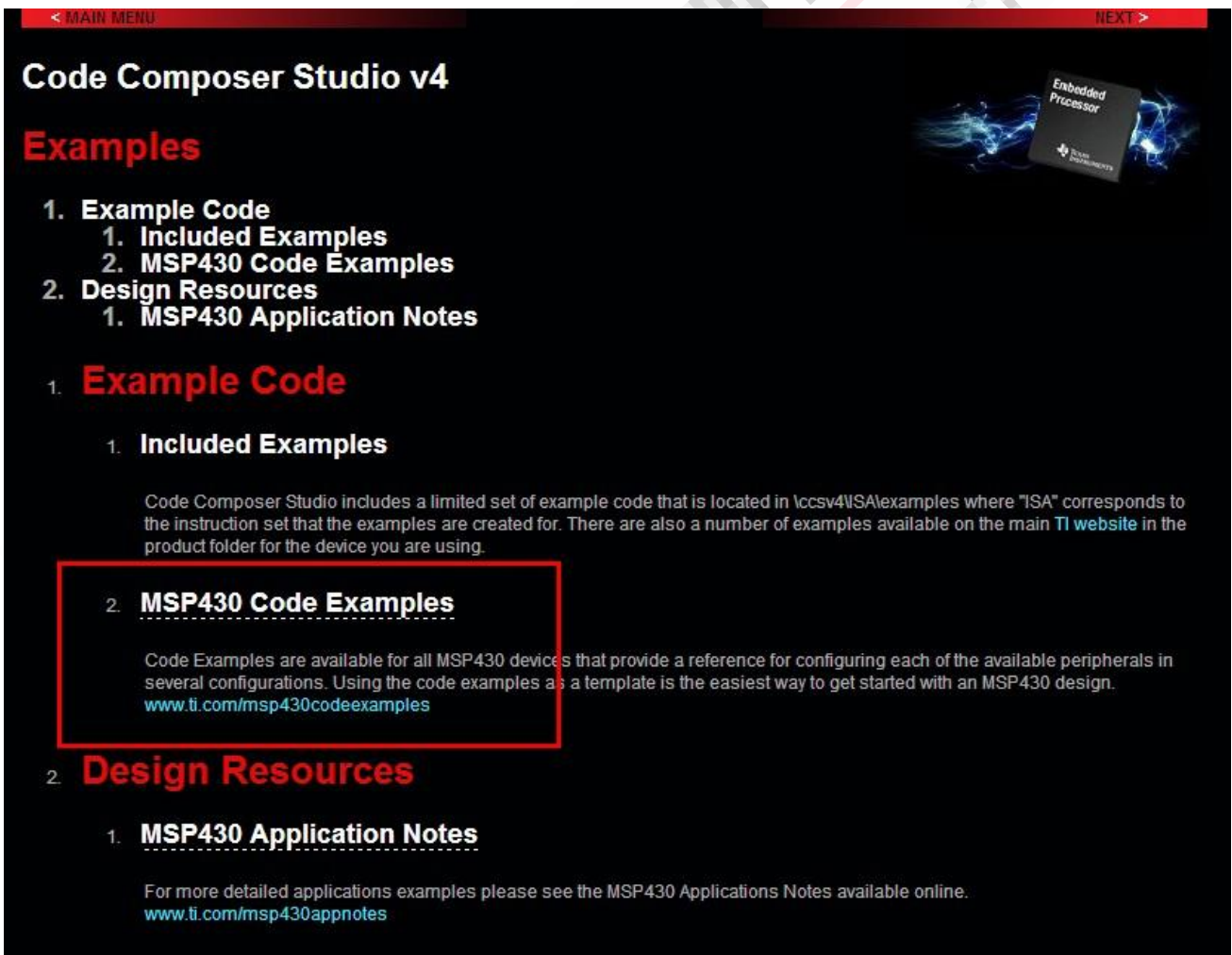
把 CCS 的欢迎界面 Welcome 最大化，我们可以通过这个欢迎界面，来找到需要的资料。我这里安装的仅仅是对 MSP430 的支持，所以页面内容都是 MSP430 相关的。

点击 Examples :





在弹出来的页面里点击 MSP430 Code Examples :



点击之后，我们可以看到 TI 官网上的示例代码页面：

Microcontrollers (MCU)

Design Support Getting Started Selection Tool Training & Events Developer Network University

MSP430™ 16-bit Ultra-Low Power MCUs (345)

16-Bit Ultra-Low Power MSP430™ Microcontrollers

MSP430 16-bit Microcontroller Code Examples and Function Library

Code examples are available for every MSP430 device that configures each of the integrated peripherals for various application needs. Each zip file below includes both C and assembly code examples that can be directly imported into IAR or CCS. Developers may use the code examples "as-is" or modify the various code modules to jumpstart custom applications. Check the readme to see the complete list of available code snippets along with brief descriptions. Please read applicable section under terms of use prior to utilizing any software. Happy coding!

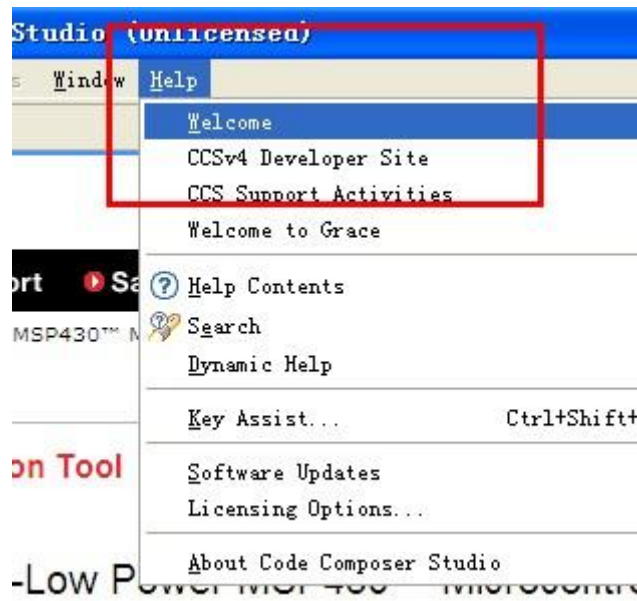
MSP430 Code Examples

Devices Supported - Includes Assembly (.asm, .s43) and C (.c) Code Examples	.zip contents
MSP430x11x1, MSP430F21x1	Read me
MSP430F11x2, MSP430F12x, MSP430F12x2	Read me
MSP430x13x, MSP430F14x, MSP430F15x, MSP430F16x	Read me
MSP430AFE25x	
MSP430F20xx	Read me
MSP430G2xx1	
MSP430G2xx2	
MSP430G2xx3	

再点击我们需要的芯片例程，就可以直接下载到了：



是不是方便了很多咧，这个页面也就相当于一个浏览器，在里面我们可以直接链接到 TI 的官网里，教程、源码等随时可得。TI 给我们提供了极大的便利，用不用，怎么用就因人而异了。闲着没事的时候，这里点点，那里按按，说不定就有意外的收获。如果不小心把这个 Welcome 给关了，我们可以通过以下方式重新打开：



5、回到主题，关于 Grace 的使用。其实在刚刚 CCS 启动之后的页面里，Meet Grace Software 的标签里已经有相关的介绍了，这个教程的内容，也只不过是把它说的东西实践一遍而已。下面开始介绍如何建立 Grace 工程，以及如何使用 Grace。



Don't show this in the future

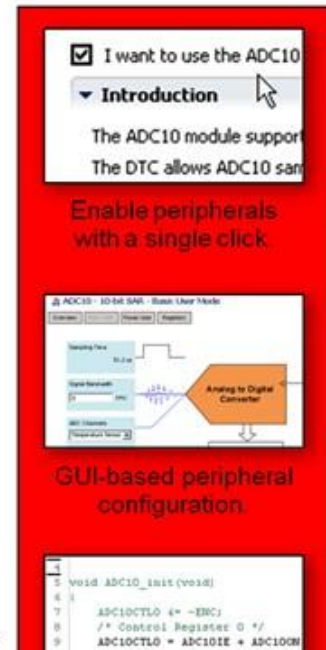
Welcome to the Grace plugin for Code Composer Studio
Quicken your understanding of MSP430 and your time to market with Grace, a graphical user interface for enabling and configuring MSP430 peripherals.

Quick Links:

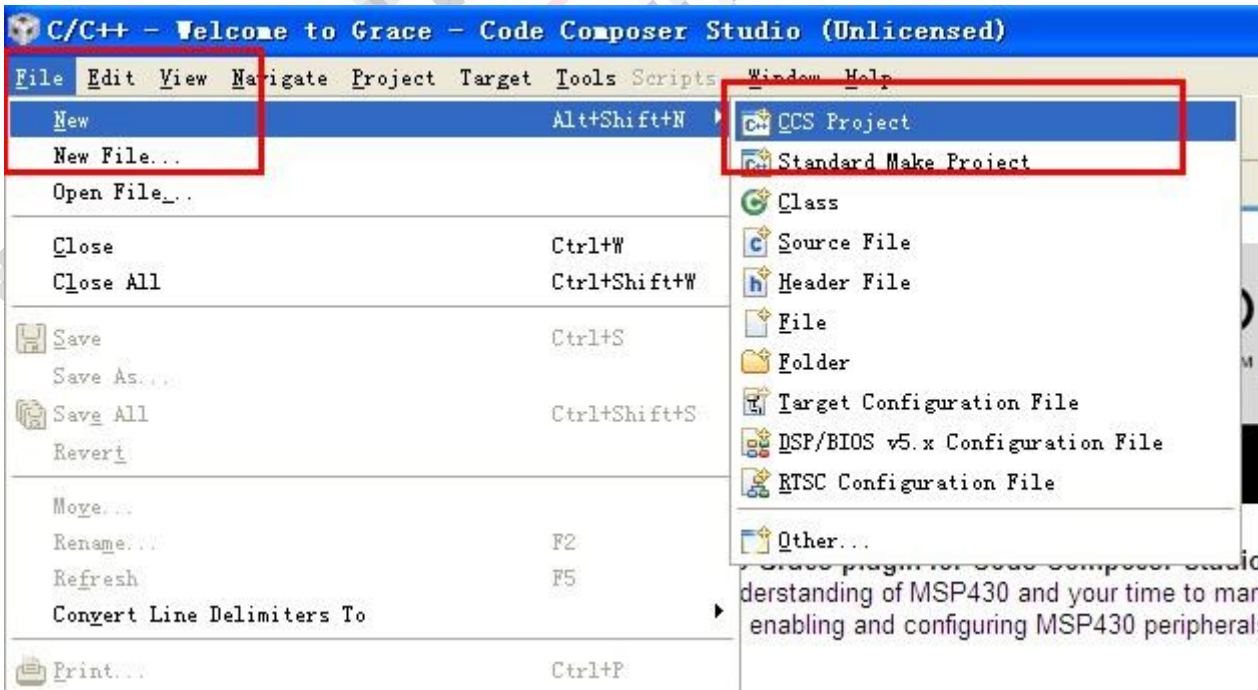
- [Video tutorial - Getting started with Grace](#)
- [Grace Quick Start Guide](#)
- [Grace Release Notes](#)
- [Join the MSP430 e2e online community](#)

Helpful Tips:

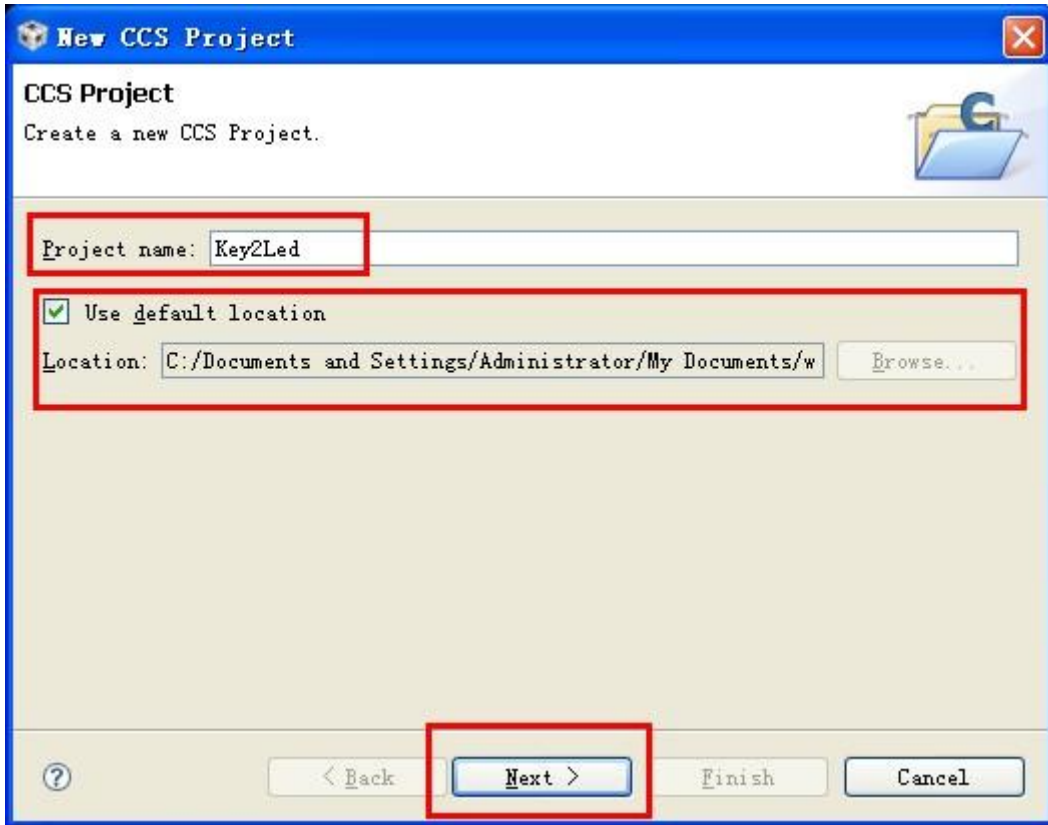
- [How do I start my own Grace project?](#)
 1. In Code Composer Studio, select **File->New->CCS Project**.
 2. Choose a name for your new project. Click on the **Next** button.
 3. Select **MSP430** for the Project Type. Click on the **Next** button.
 4. If this new project is dependent on another pre-existing project, you can select that now. Otherwise, click on the **Next** button.
 5. Choose **Executable** for the Output Type. Select the device being used under



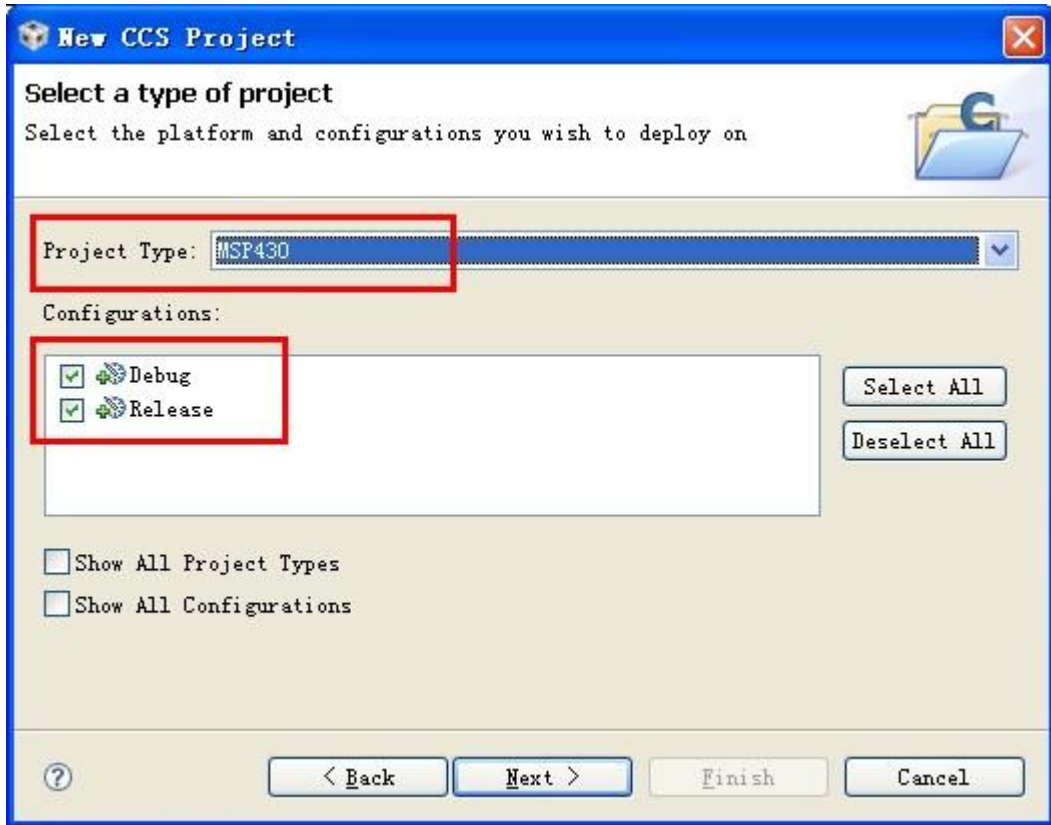
6、开始新建工程，File -> New -> CCS Project：



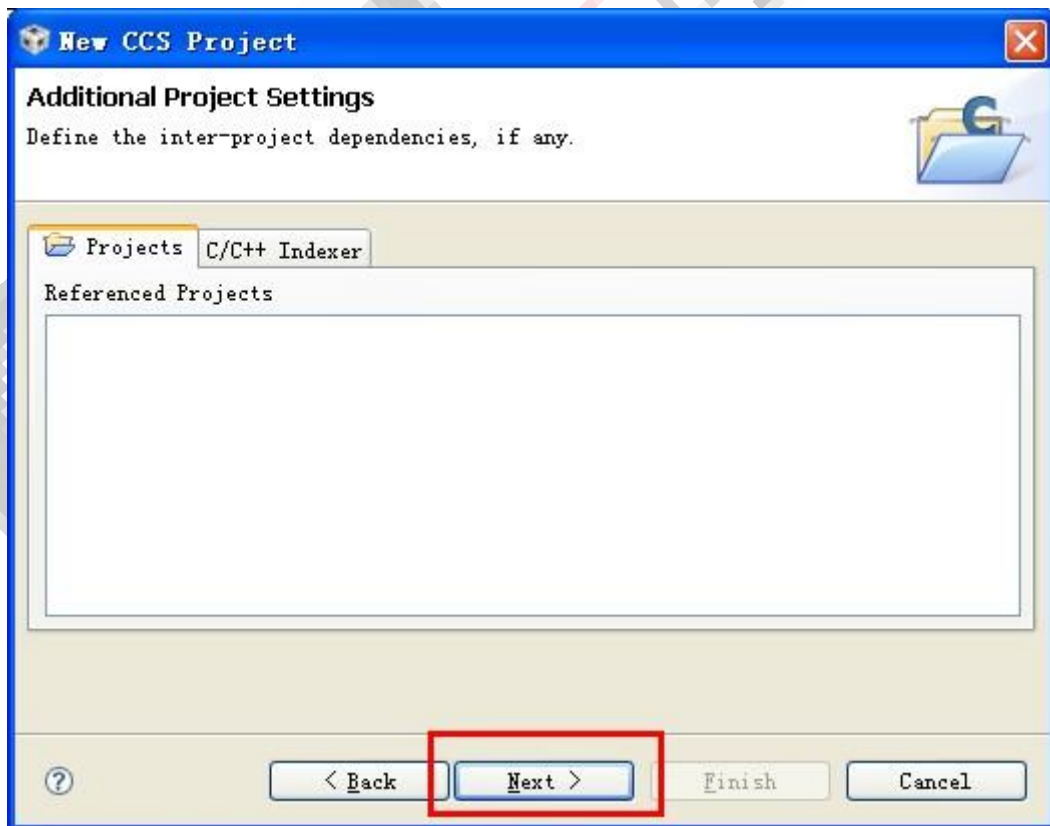
7、选择工程路径，输入工程名称，这里我的工程放在默认目录里，点击 Next：



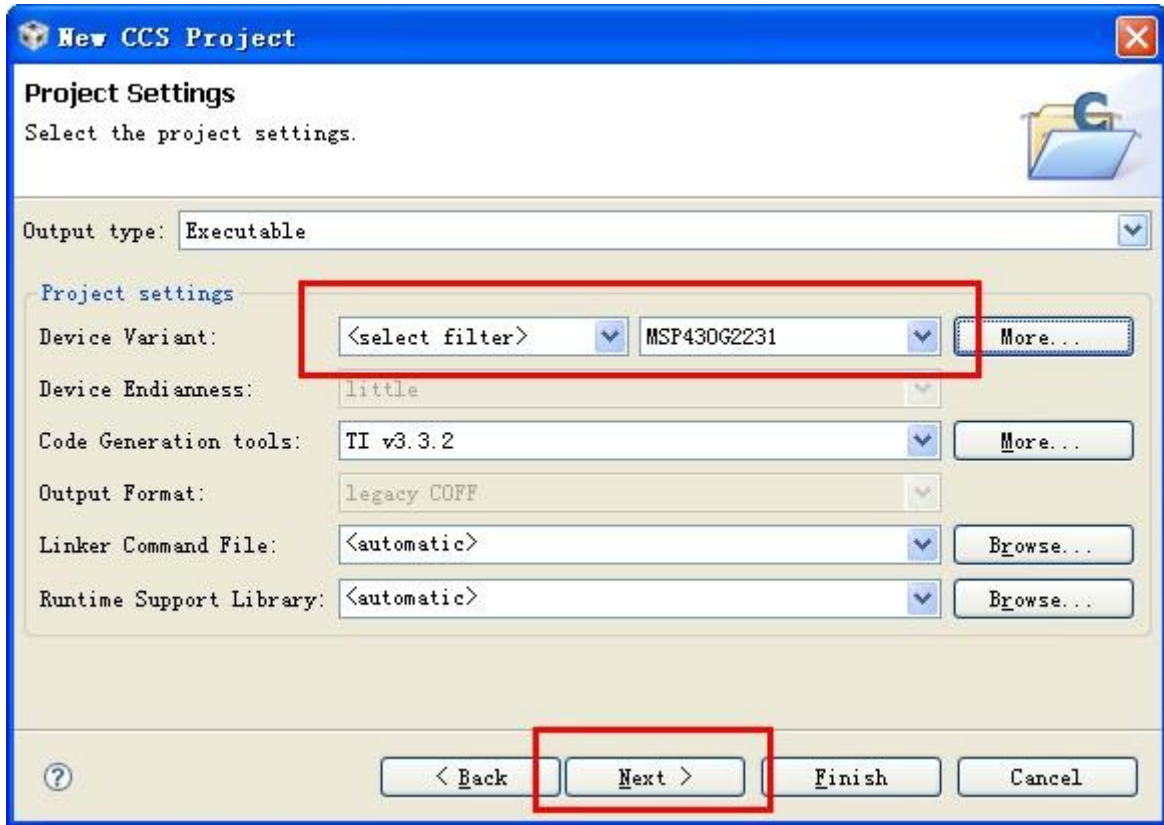
8、选择工程类型，我这里只安装 MSP430 的，没有安装 Stellaris、DSP 那些，默认的是 MSP430，就不用选了。Debug 是我们开发过程中调试的，Release 是调试完成后用于发布的，细致的差别，我没有深究过，按照它默认来就行了。如果你勾选了下面的 show all...，你会发现 Project Type 选项卡里多了不少项，这里我们按照它默认的 MSP430 来就行了，其它的需要的时候再去了解。点击 Next：



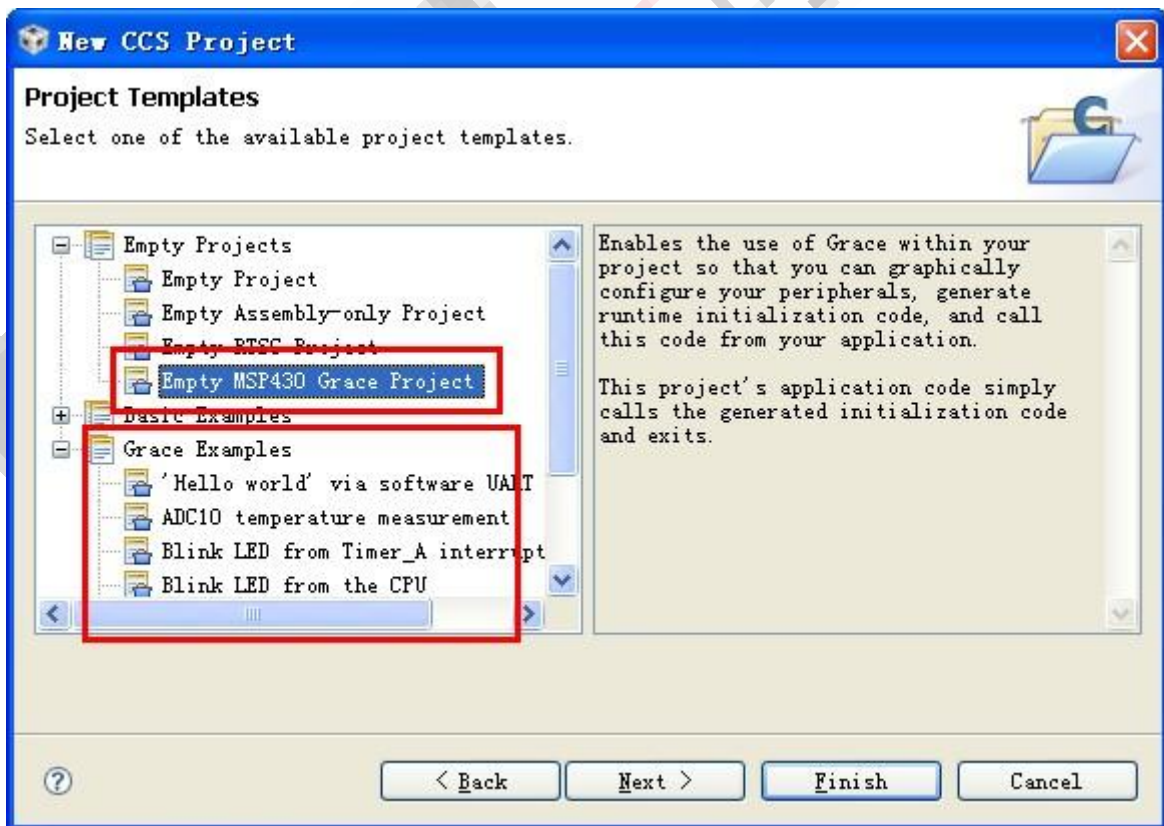
9、下面是 Additional Project Settings ，直接点 Next ，



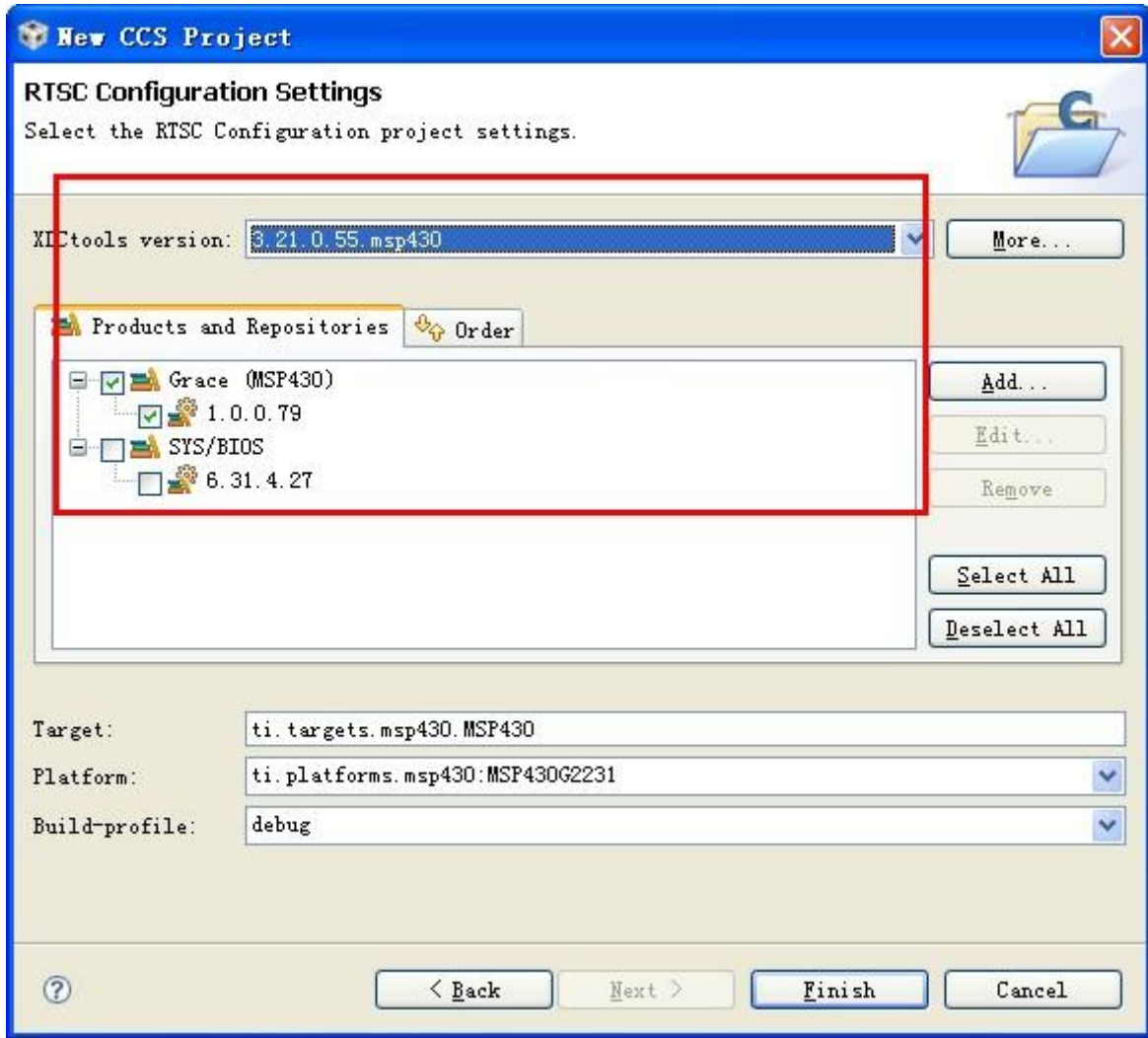
10、下面选择我们目标板的芯片，这里是 MSP430G2231 ，点击 Next :



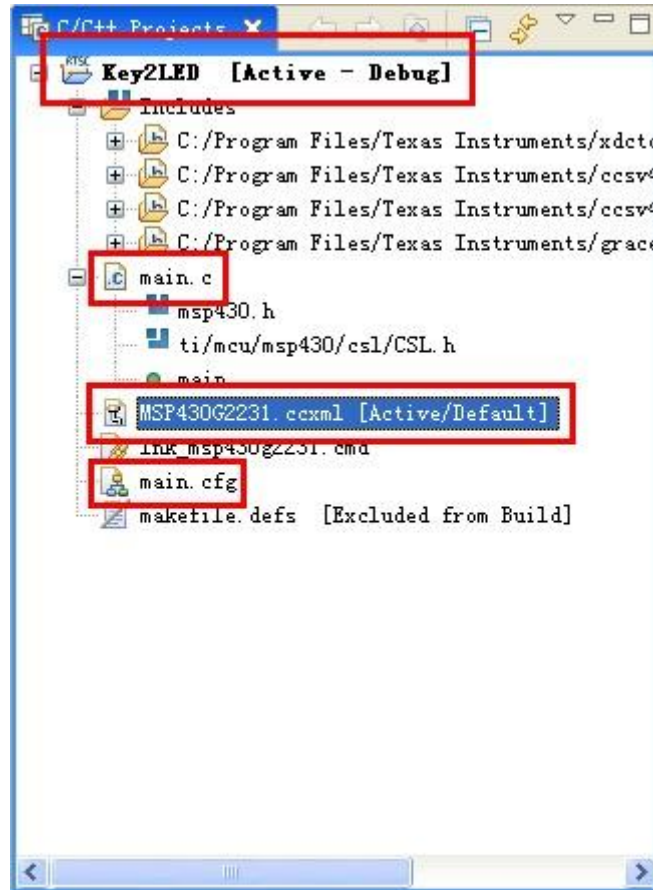
11、选择 Empty MSP430 Grace Project , 下面还有 Grace 的一些例程。点击 Next :



12、在这个 RTSC Configuration Settings 里按照它们默认设置 , 点击 Finish.

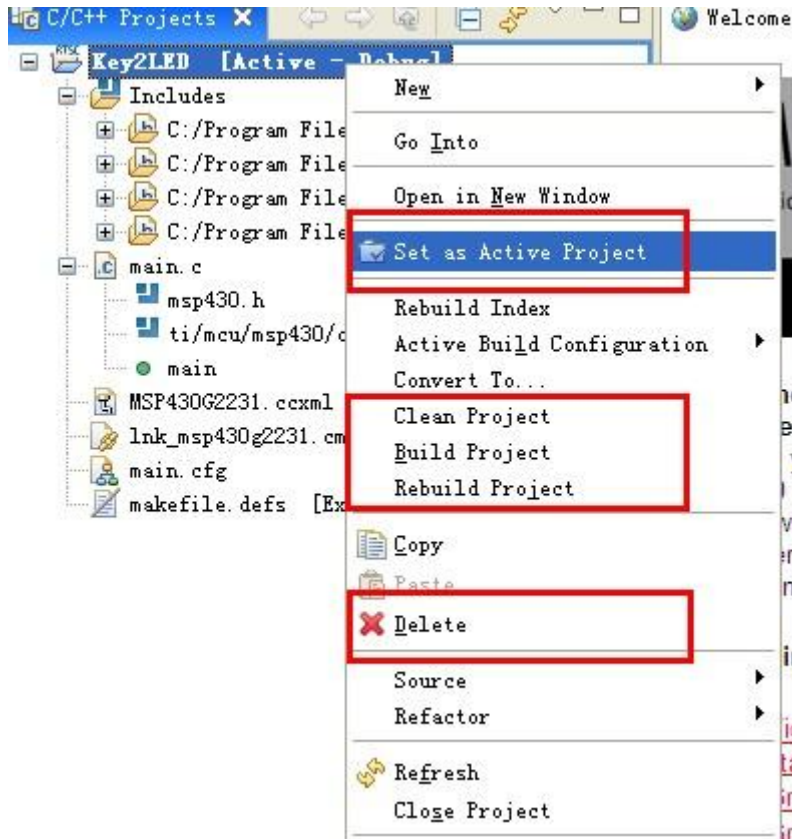


13、提醒一下，我安装的 CCS 只勾选了对 MSP430 的支持，如果安装了 Stellaris、DSP 等其它的话，选项可能会有出入，按照以上图中的来选就行。点击工程前的+号，展开这个工程：

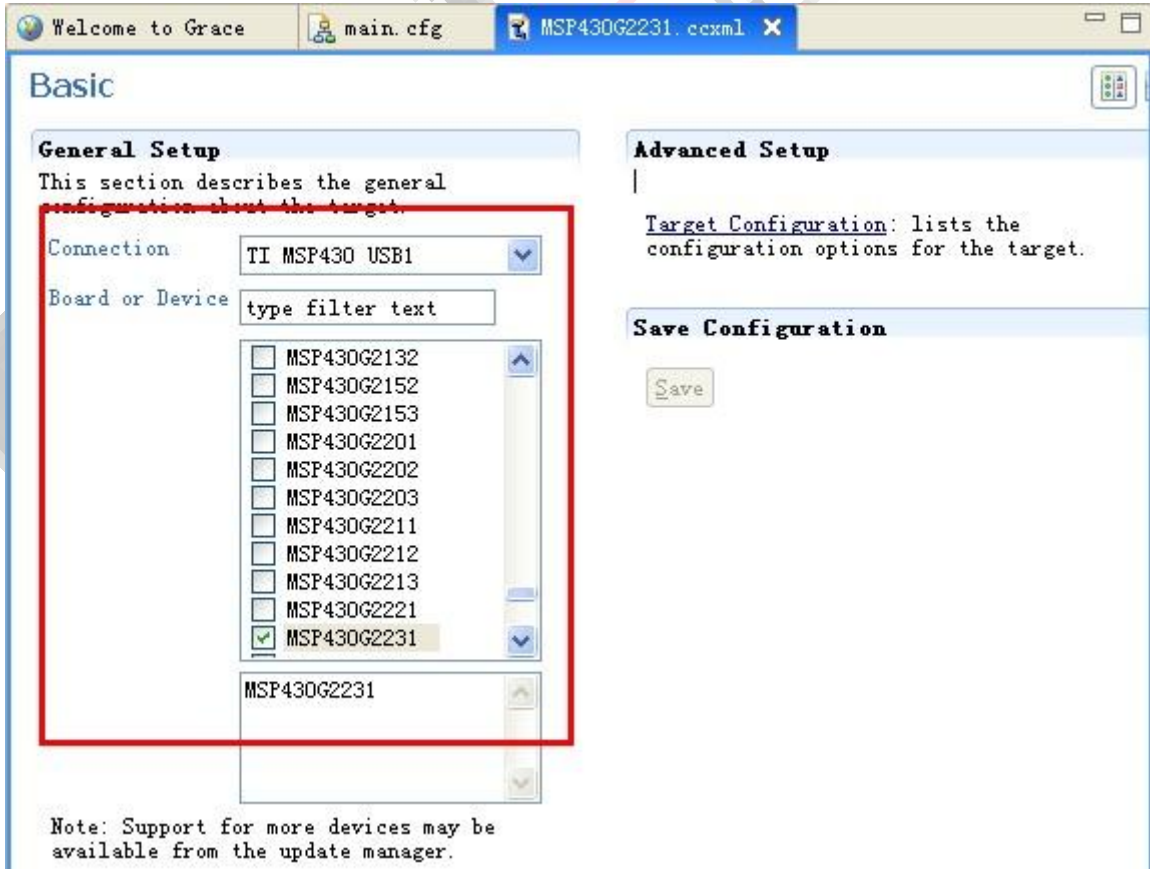


这里对工程解释一下，工程名称字体加粗，表明这个工程当前使用的 Active，如果你有多个工程的话，可以看出差别。编译、仿真操作都是针对当前工程 Active 而言的。右击工程可以选择要激活使用的工程，还有一些编译、删除等操作。

有必要提醒一下，如果你右击对工程进行删除的时候，如果选择对工程文件也删除的话，请确保该工程路径下的文件都可以删除的，因为这个删除操作会把整个文件夹删掉，而且是永久删除，没有放回收站的。我曾经试过随便建工程，然后删除的时候，把其它.dwg、.doc 啥的全删了，悲剧。TI 应当考虑修改下，起码删除要经回收站保留一下。不然就只能通过其它方式恢复数据了。

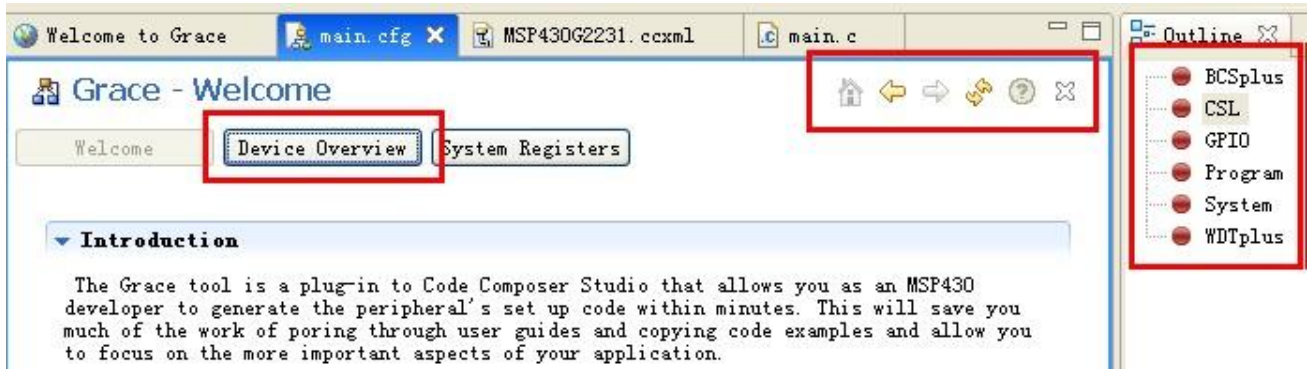


MSP430G2231.ccxml，这个文件是关于仿真设置的。选择使用的仿真器和要仿真的芯片型号。



main.c 就是我们的主函数所在的.c 文件了。

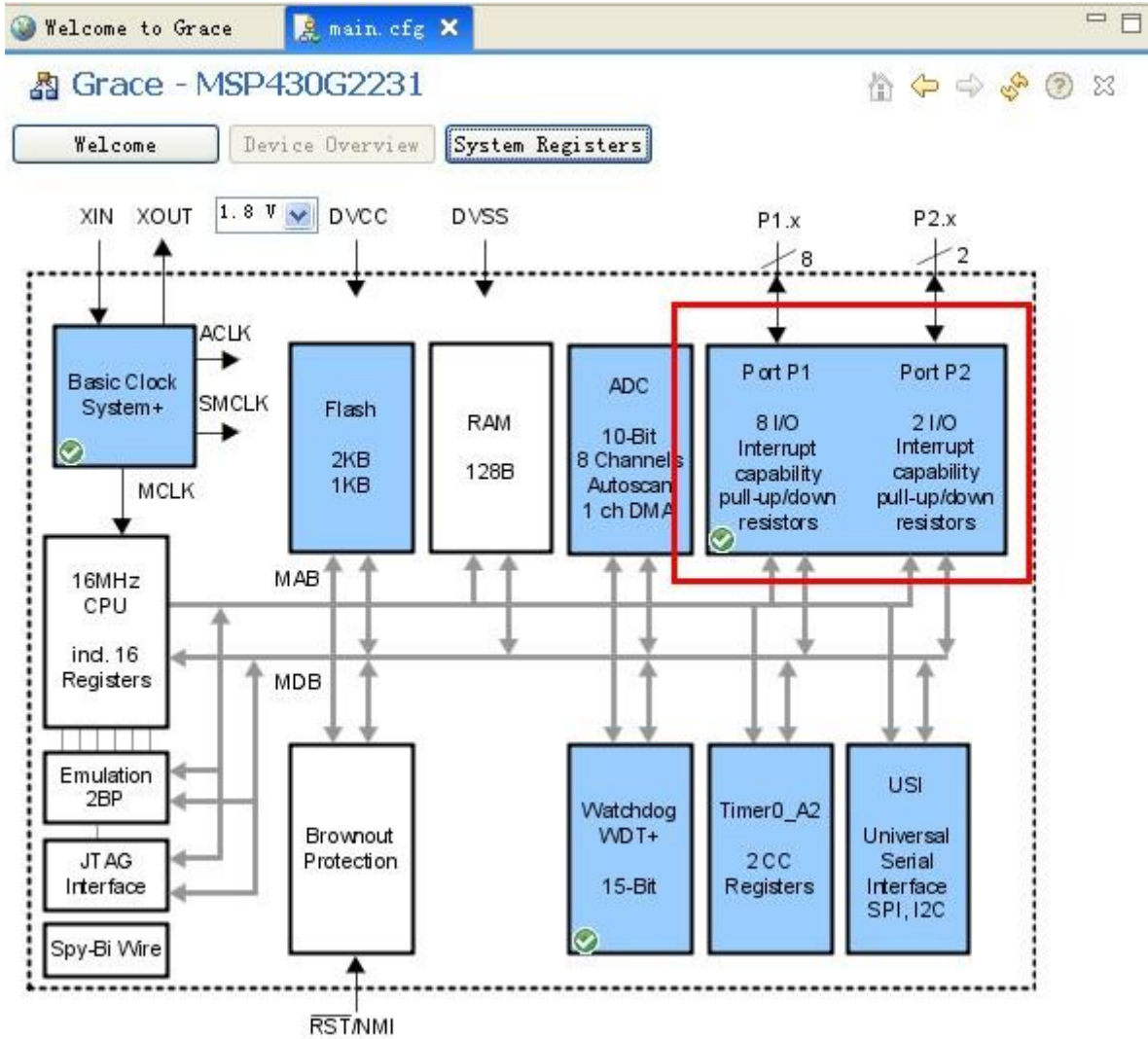
main.cfg 这个是 Grace 文件，双击它，我们就可以通过它来初始化 MSP430 了，当然初始化过程都是图形化的，勾勾选选就可以了。



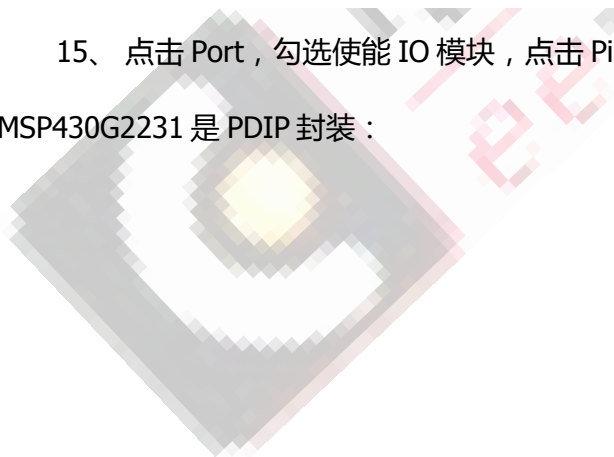
点 Device Overview，就可以看到整个 MSP430G2231 的外设架构了，双击右侧 Outline 里的项可以直接对该项进行设置，比如 GPIO。这时，我们如果要回到 Grace 初始界面，我们可以通过页面上角的主页、前进、后退那些来回到一开始的界面。

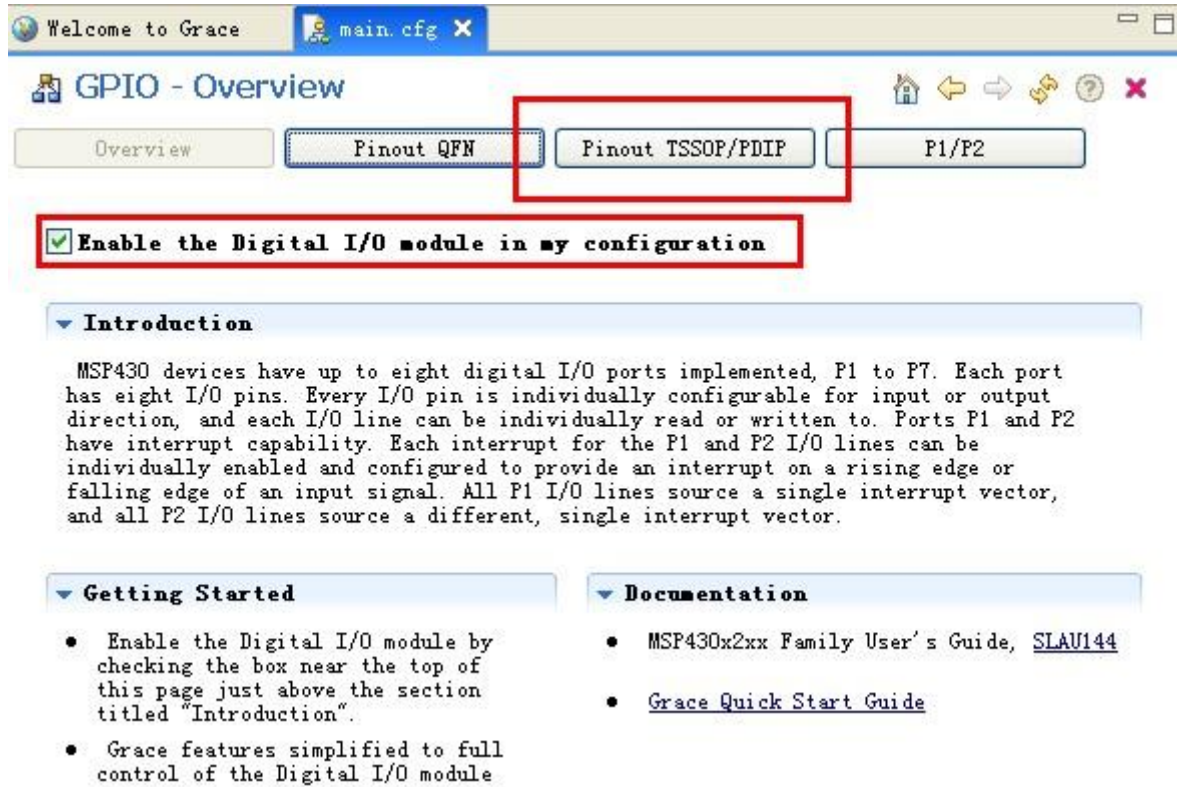
14、点击点 Device Overview，这里我们要弄一个简单的程序，实现按下 S2 按键/P1.3，LED1/P1.0 点亮，LED2/P1.6 也点亮，未按则熄灭：



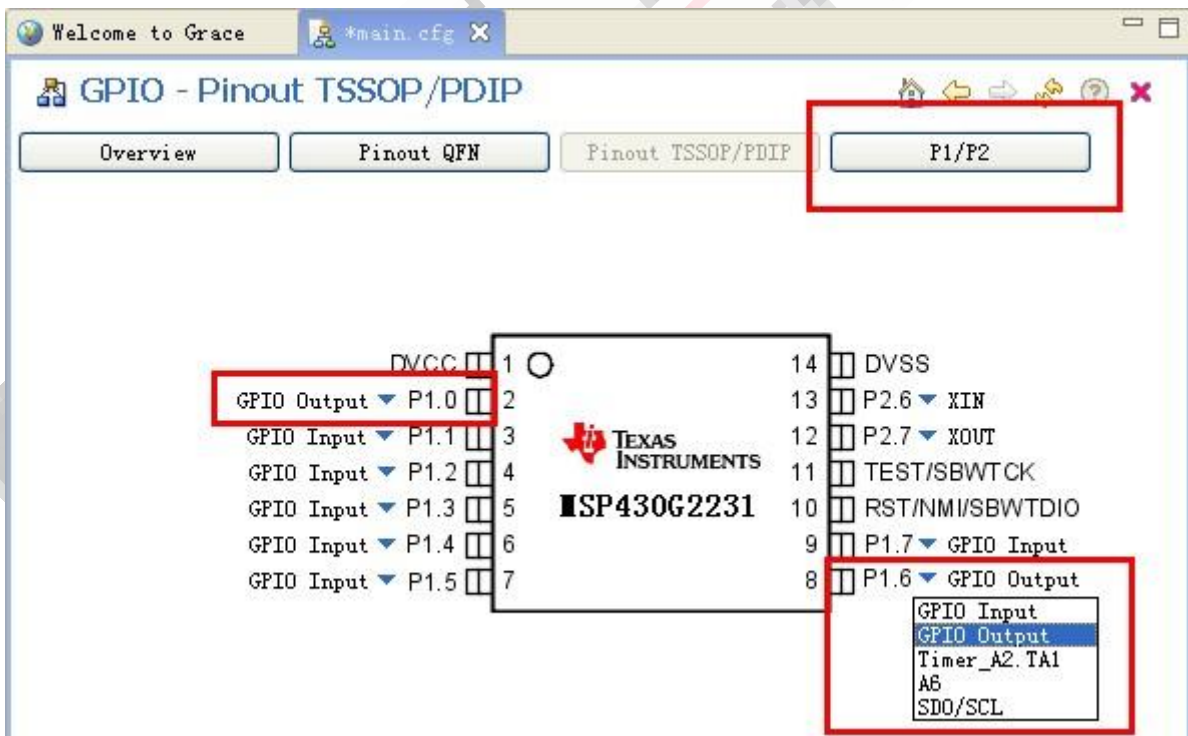


15、点击 Port，勾选使能 IO 模块，点击 Pinout TSSOP/PDIP，因为 LaunchPad 板载的 MSP430G2231 是 PDIP 封装：





16、通过下拉选项，我们点选 P1.0、P1.6 为 GPIO Output 引脚：



17、点击右上角的 P1/P2 按钮，进入引脚的详细设置，这里配置 P1.3 为带上拉输入模式，P1.0、P1.6 为输出模式，当你用鼠标指向选框的时候 Grace 还会给出相应的提示：

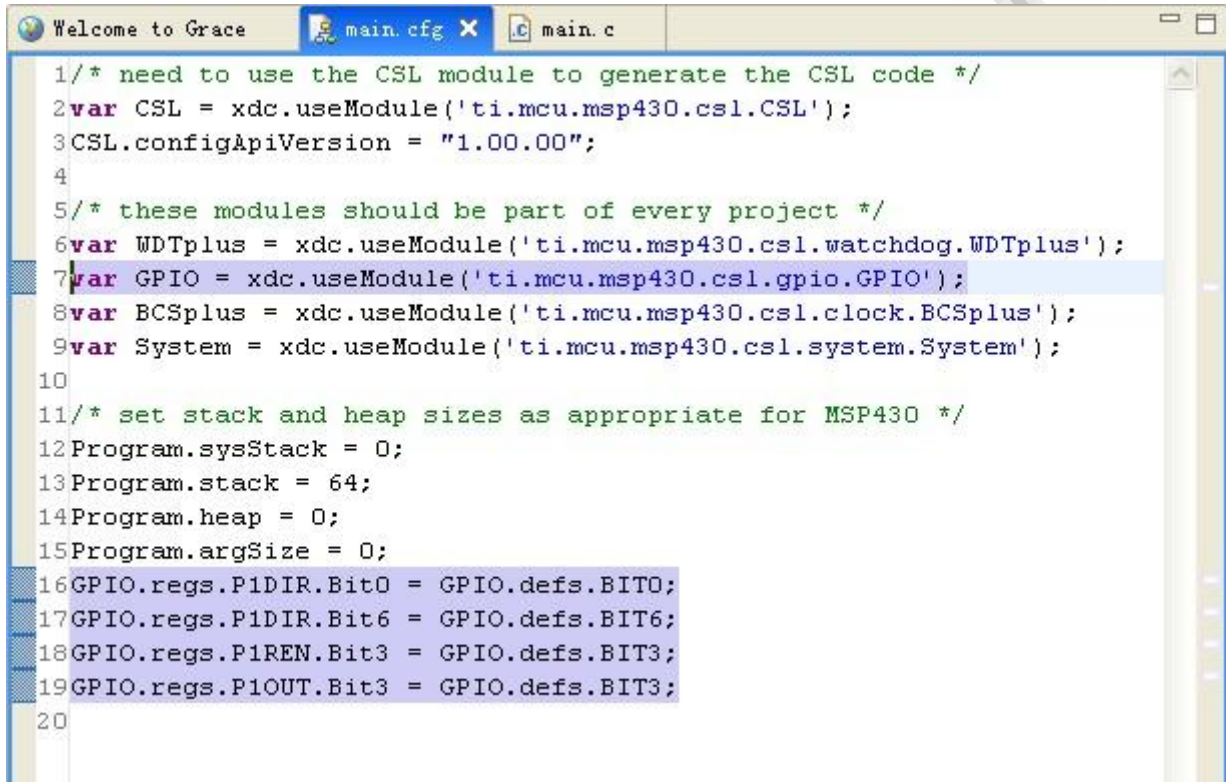
The screenshot shows the TI Configurator interface for GPIO Port 1 and Port 2. The 'P1/P2' tab is selected. The interface is divided into two columns for PORT 1 and PORT 2. Each column contains several registers with bit fields (0-7) and checkboxes. Red boxes highlight the following settings:

- PORT 1 Output Register:** Bit 3 (OUTx) is checked.
- PORT 1 Direction Register:** Bit 6 (DIRx) and Bit 0 (DIRx) are checked.
- PORT 1 Resistor Enable Register:** Bit 3 (RENx) is checked.
- PORT 2 Port Select Register:** Bit 7 and Bit 6 (SELx) are checked.

At the bottom, the 'Interrupt Handler' and 'After Interrupt' fields are visible. A tooltip for the Resistor Enable Register bit 3 provides the following information:

GPIO Enables or Disables Pullup/Pulldown Bit 3
 Bit = 0: Pullup/pulldown resistor disabled
 Bit = 1: Pullup/pulldown resistor enabled
 Set Pullup/down via PxOUT 0 = Down; 1 = Up

18、用过 ICCAVR 向导或者飞思卡尔 PE 工具的可能会有疑问，ICCAVR 那些配置完成后，会有源码给我们校验，有很多外设的初始化函数，而这个 Grace 配置完成后，在 main.c 里并没有初始化代码，而仅仅是调用了一个初始化函数 CSL_init();而已。其实也是有的，点击上图中左下角的 Source 就可以看到了，不过，确实跟其它厂商的不一样，当效用是类似的。如果 TI 的 Grace 初始化后能提供通用源码，那就可以直接移植到 IAR，这样倒真的方便不少。



```

Welcome to Grace  main.cfg X  main.c
1 /* need to use the CSL module to generate the CSL code */
2 var CSL = xdc.useModule('ti.mcu.msp430.csl.CSL');
3 CSL.configApiVersion = "1.00.00";
4
5 /* these modules should be part of every project */
6 var WDTplus = xdc.useModule('ti.mcu.msp430.csl.watchdog.WDTplus');
7 var GPIO = xdc.useModule('ti.mcu.msp430.csl.gpio.GPIO');
8 var BCSplus = xdc.useModule('ti.mcu.msp430.csl.clock.BCSplus');
9 var System = xdc.useModule('ti.mcu.msp430.csl.system.System');
10
11 /* set stack and heap sizes as appropriate for MSP430 */
12 Program.sysStack = 0;
13 Program.stack = 64;
14 Program.heap = 0;
15 Program.argSize = 0;
16 GPIO.regs.P1DIR.Bit0 = GPIO.defs.BIT0;
17 GPIO.regs.P1DIR.Bit6 = GPIO.defs.BIT6;
18 GPIO.regs.P1REN.Bit3 = GPIO.defs.BIT3;
19 GPIO.regs.P1OUT.Bit3 = GPIO.defs.BIT3;
20

```

19、至此，我们对 MSP430G2231 的初始化配置就已经完成了，至于时钟那些，我们按照它的默认配置来就行，使用内部的时钟。

20、双击左侧工程栏的 main.c，添加如下代码：

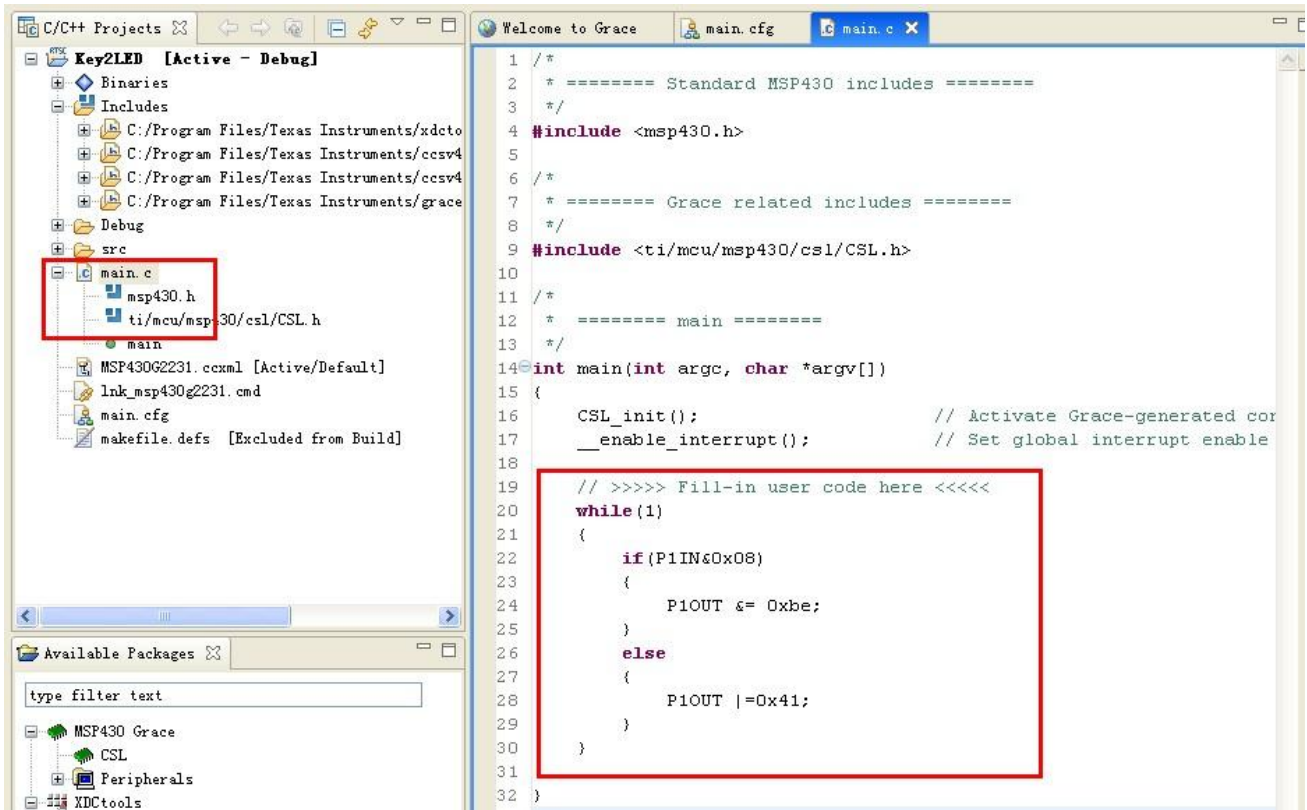
```

// >>>> Fill-in user code here <<<<<
while(1)
{
    if(P1IN&0x08)
    {
        P1OUT &= 0xbe;
    }
    else
    {

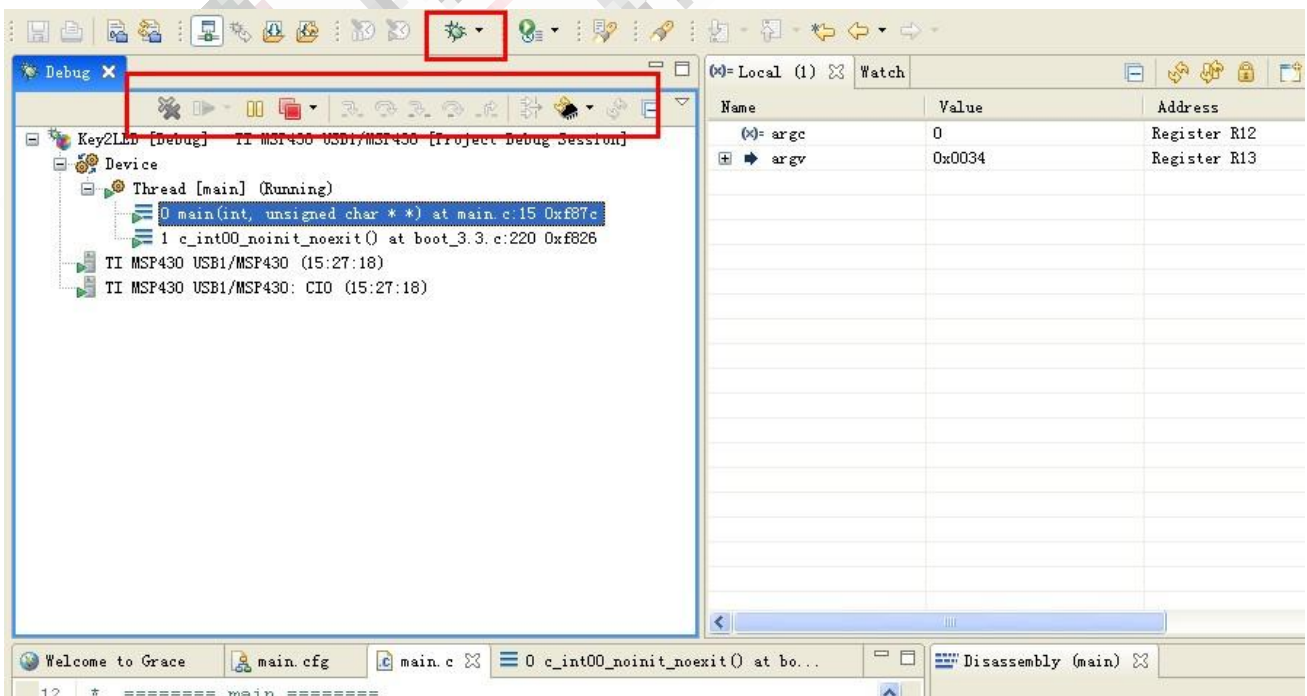
```

```
P1OUT |=0x41;
```

```
}
}
```



21、 点击工具栏的绿甲壳虫 Debug Launch，提示保存更改，然后就会自动编译下载，进入仿真了：



22、程序运行结果是按下 S2 按键，LED1、LED2 都点亮，没有按键，两者都熄灭。

如果进入仿真时提示 error：



MSP430: Error initializing emulator: Could not initialize device interface

MSP430: Warning: Could not communicate with FET

MSP430: Program loaded. Code Size - Text: 216 bytes Data: 2 bytes

解决方法一般是重新拔插一下 USB 线即可。

至此，Grace 简单入门完成。如果要用好 Grace，对芯片的还是要有一定的了解，比如寄存那些，因为 Grace 有不少都是通过勾选的方式来配置的，知其然还要知其所以然，用起来才会得心应手。

David Lee

2011-09-21 15:31:38