

CLB 案例开发手册

Revision History

Draft Date	Revision No.	Description
2022/09/10	V1.0	1. 初始版本。

目 录

前 言 ..... 3

1 CLB 模块简介 ..... 4

2 开发环境搭建 ..... 5

    2.1 安装 CCS12.0.0 ..... 6

    2.2 安装 C2000Ware\_4\_01\_00\_00 ..... 11

    2.3 安装 TDM-GCC ..... 14

3 clb\_external\_signal\_AND\_gate2 案例 ..... 17

    3.1 案例说明 ..... 18

    3.2 案例测试 ..... 18

    3.3 案例编译 ..... 20

    3.4 关键代码 ..... 22

更多帮助 ..... 28

前 言

（裸机）CLB 案例基于软件开发包 C2000Ware\_4\_01\_00\_00 和 GNU 编译器 TDM-GCC 进行开发。C2000Ware 是一套全面的软件和文档集，旨在最大限度地缩短开发时间，它包括特定于器件的驱动程序、库文件以及外设示例程序。

（裸机）CLB 案例位于产品资料“4-软件资料\Demo\C2000Ware-demos\”路径下。目录说明如下表，其中 bin 目录存放程序可执行文件，project 目录存放案例工程源文件。

表 1

目录	文件/目录	说明
bin	xxx_ram.out	程序可执行文件，用于加载至 DSP 片内 RAM
	xxx_flash.out	程序可执行文件，用于固化至 DSP 片内 FLASH
project	工程源码	

本文档案例程序仅支持 TMS320F28379D 的核心板，默认通过 TL-XDS200 仿真器加载运行进行操作效果演示。

## 1 CLB 模块简介

CLB(Configurable logic block)模块是集成在 TMS320F28379x 器件中的硬件模块。CLB 包含一组可配置的模块并且能够内部互连,使用户能够按照可使用类似 FPGA 实现的方式创建自定义数字逻辑。例如,可对 CLB 进行配置,以增强现有器件外设的功能或创建新的外设功能。

用户可使用 Code Composer Studio™(CCS)的"SysConfig"图形用户界面(GUI)配置和连接每个 CLB 逻辑块中的子模块。该工具会生成一个 C 头文件,其中包含一组与用户在 GUI 中定义的配置设置相对应的常数。该工具还生成一个 C 源文件,该文件使用 C 头文件中的常数,通过一系列寄存器加载操作将这些常数加载到 CLB 寄存器中,从而初始化 CLB 模块。应用程序仅需在初始化期间调用该 C 源文件中的函数,即可完成 CLB 逻辑块的配置和连接。

CLB 使用详细请参考产品资料“6-开发参考资料\TI 官方参考文档\应用手册\”目录下的《zhcu712》文档。官方案例请参考"C2000Ware\_4\_01\_00\_00\driverlib\f2837xd\examples\cpu1\clb\"。

CLB 工具框架和构建流程,如下图所示。

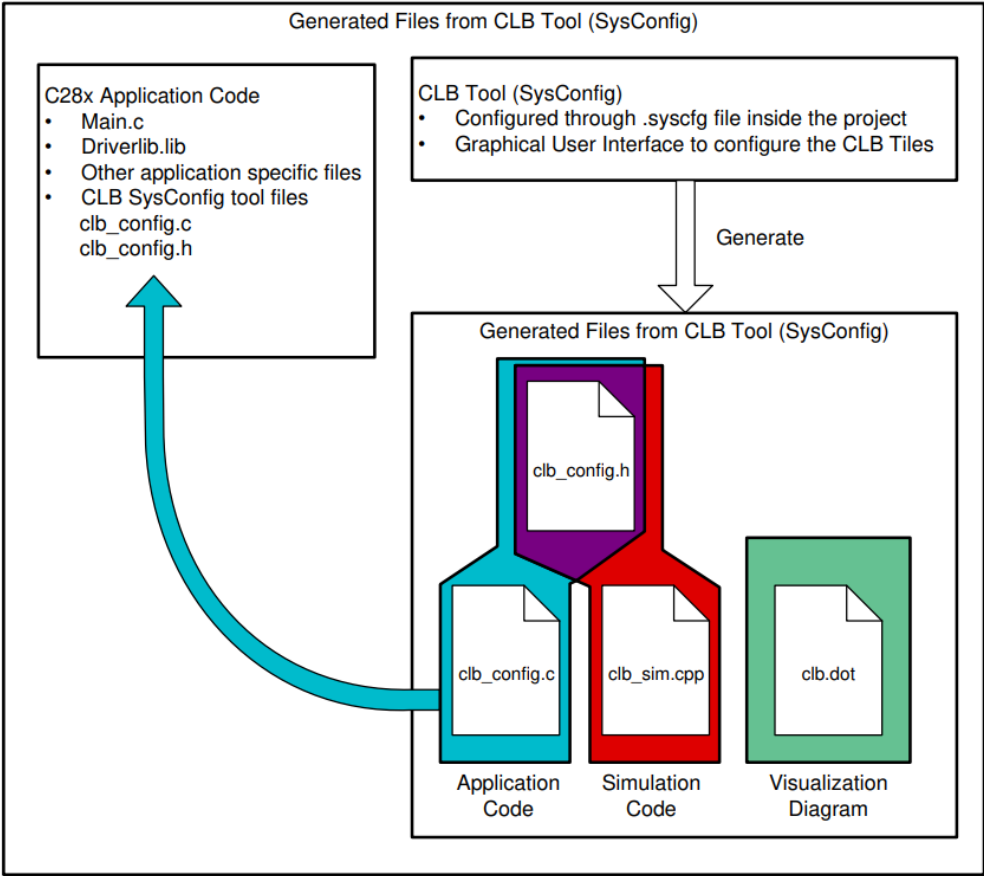


图 1 CLB 工具框架

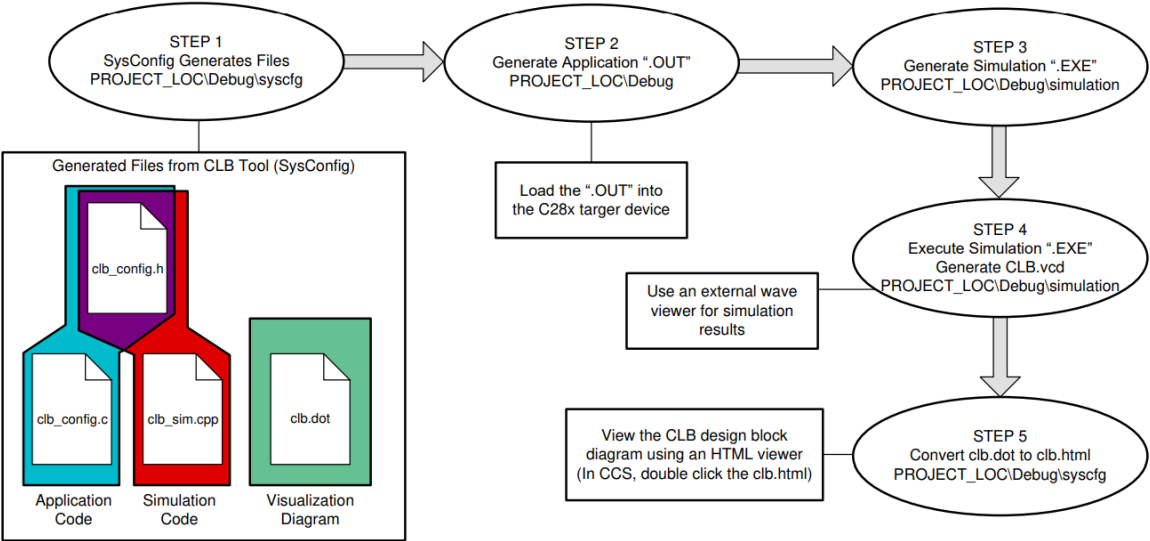


图 2 CLB 工具构建流程

## 2 开发环境搭建

因我们的存在，让嵌入式应用更简单

2.1 安装 CCS12.0.0

CLB 功能需使用 CCS12.0.0 集成开发环境。请将产品资料“4-软件资料\Tools\Windows\”目录下 CCS12.0.0.00009\_win64.zip 压缩包解压至 Windows 非中文目录下。



图 3

双击 ccs\_setup\_12.0.0.00009.exe 程序，弹出如下界面。

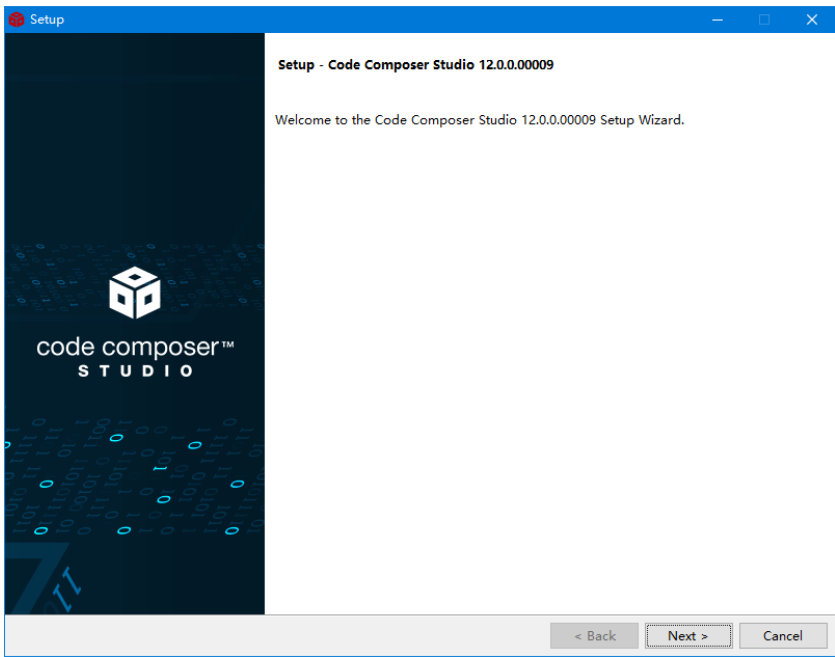


图 4

点击 Next，弹出如下界面。

因我们的存在，让嵌入式应用更简单

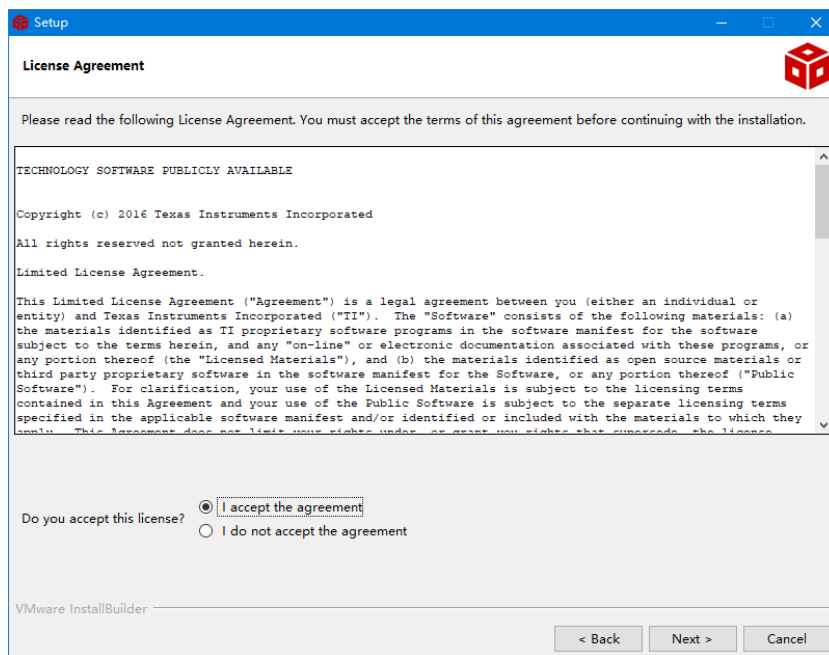


图 5

选择"I accept the agreement", 点击 Next, 弹出如下界面。

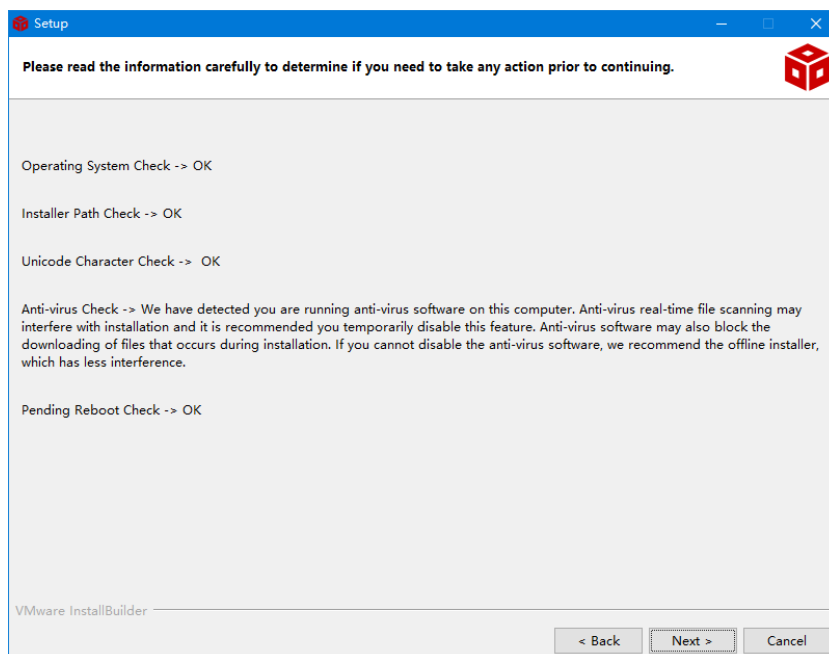


图 6

点击 Next, 弹出如下界面。

因我们的存在，让嵌入式应用更简单

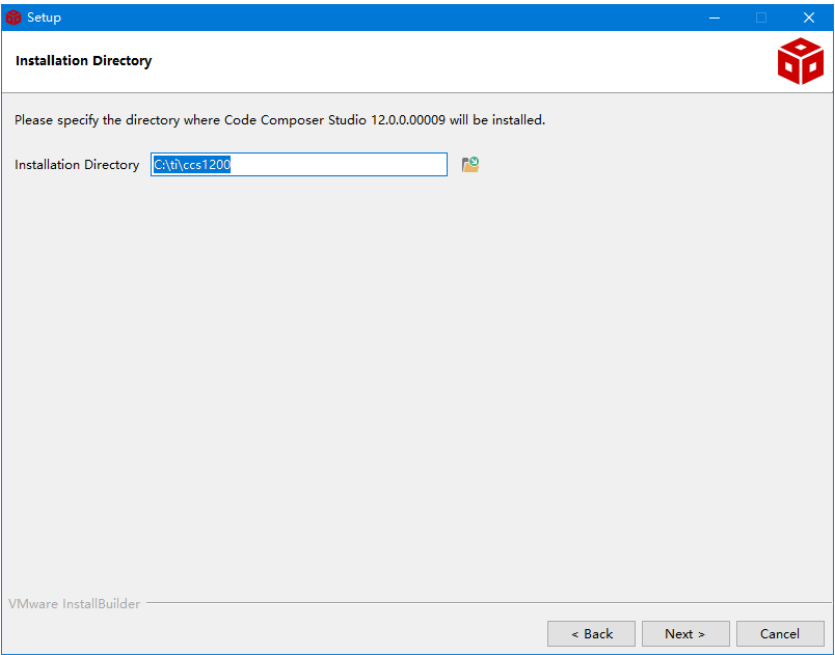


图 7

选择安装路径，点击 **Next**，弹出如下界面。

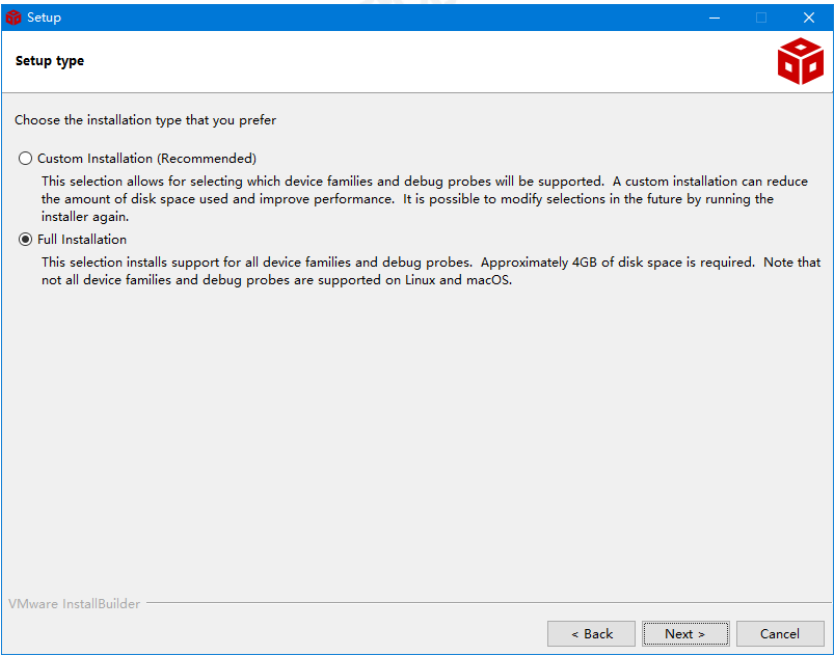


图 8

因我们的存在，让嵌入式应用更简单



选择"Full Installation", 点击 Next, 弹出如下界面。

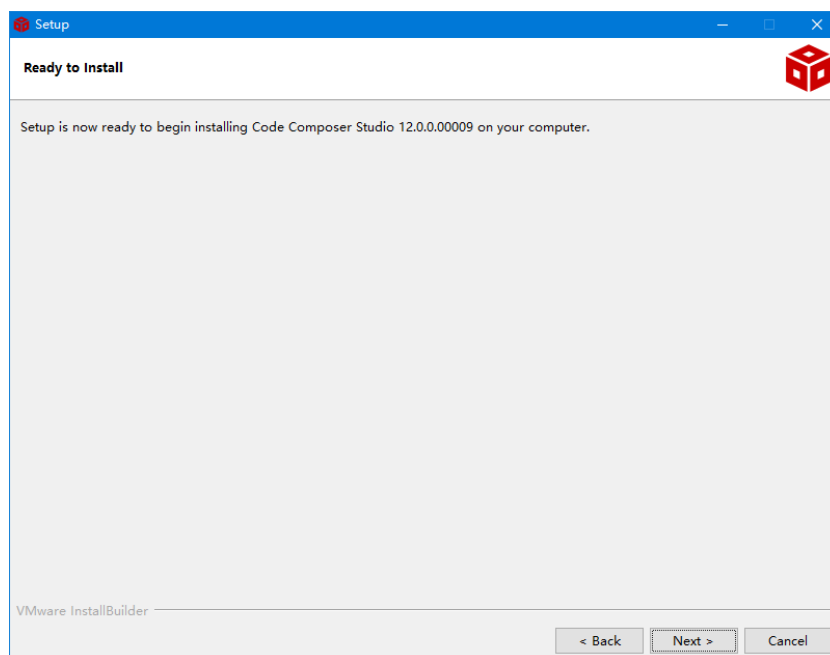


图 9

点击 Next, 弹出如下安装进度条界面。

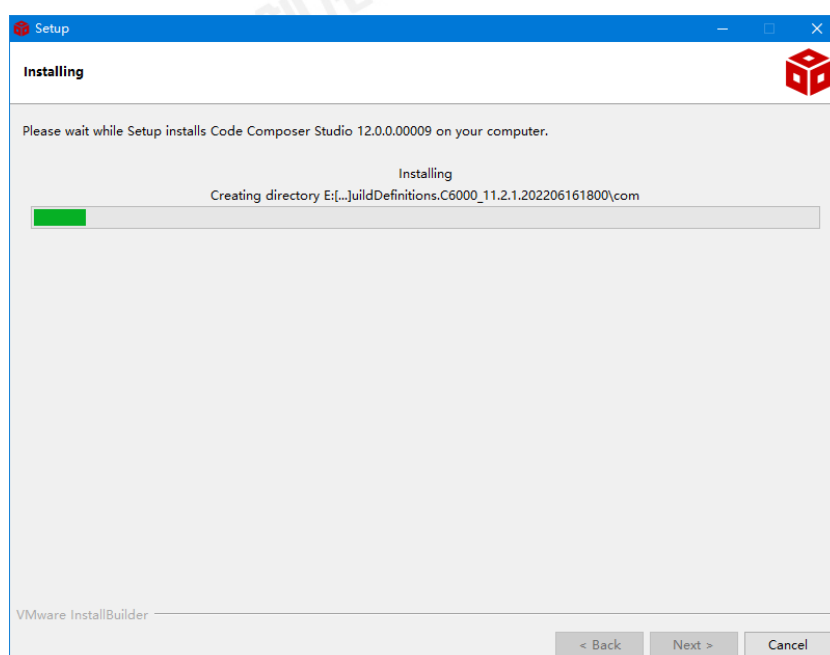


图 10

因我们的存在，让嵌入式应用更简单

安装完成，弹出如下界面，点击确认。

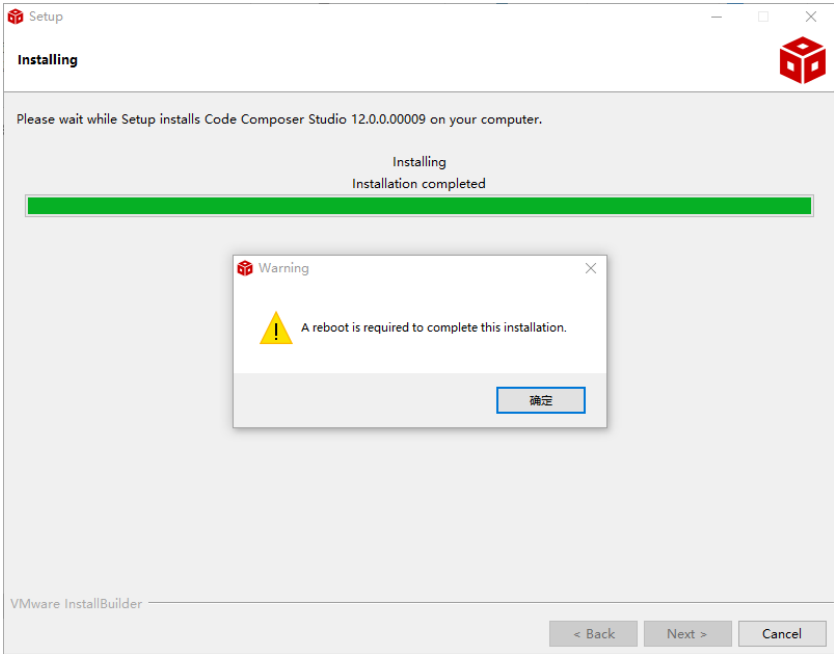


图 11

弹出如下界面，点击 Finish，至此 CCS 安装完成。

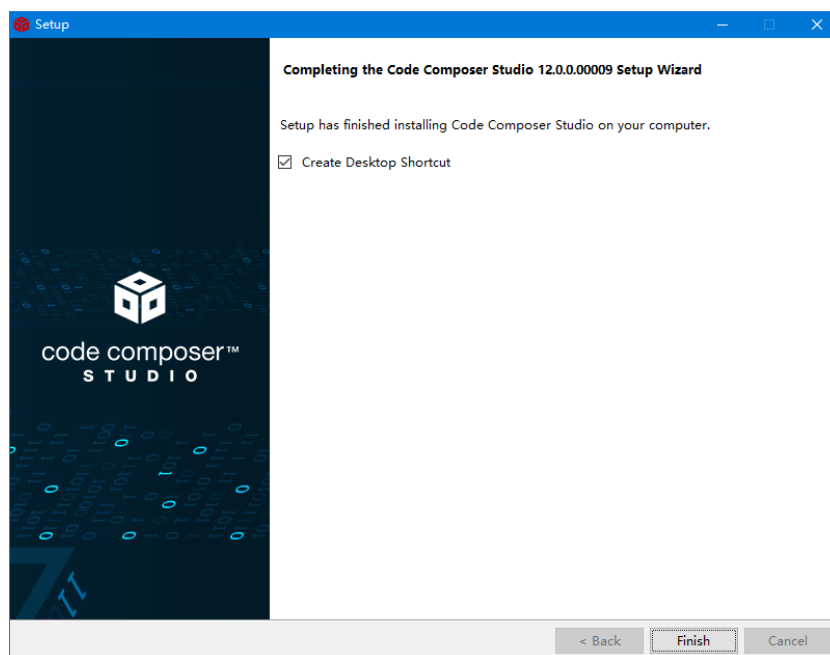


图 12

## 2.2 安装 C2000Ware\_4\_01\_00\_00

双击产品资料“4-软件资料\Tools\Windows\”目录下的 C2000Ware\_4\_01\_00\_00\_setup.exe 程序，弹出如下界面。

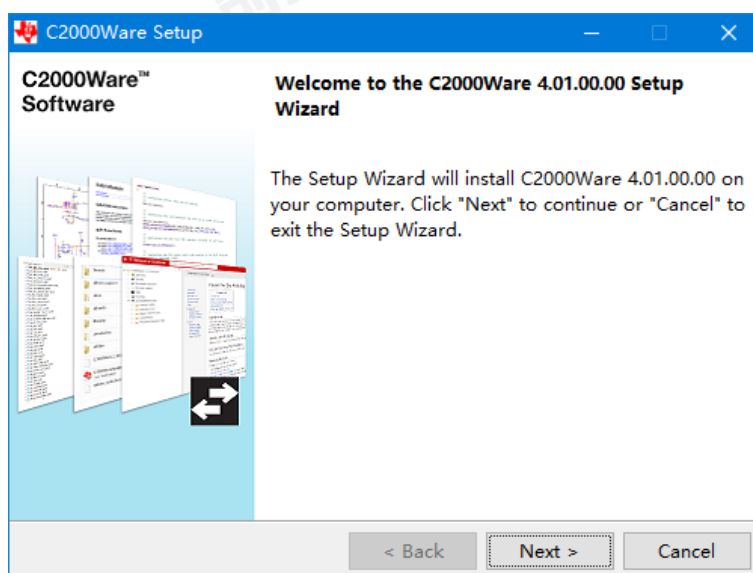


图 13

因我们的存在，让嵌入式应用更简单

点击 Next，弹出如下界面。

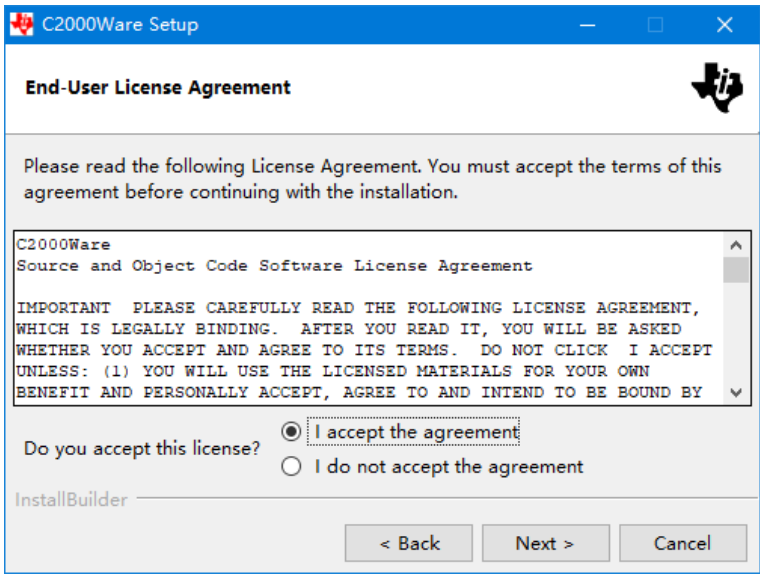


图 14

选择"I accept the agreement"，点击 Next，弹出如下界面。

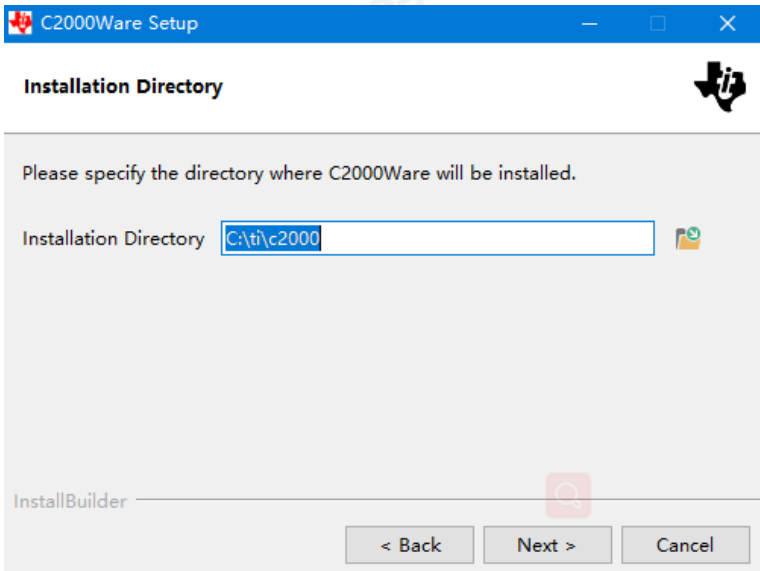


图 15

选择安装路径，点击 Next，弹出如下界面。请将 C2000Ware\_4\_01\_00\_00 安装至 CCS12.0.0 相同安装目录下。

因我们的存在，让嵌入式应用更简单

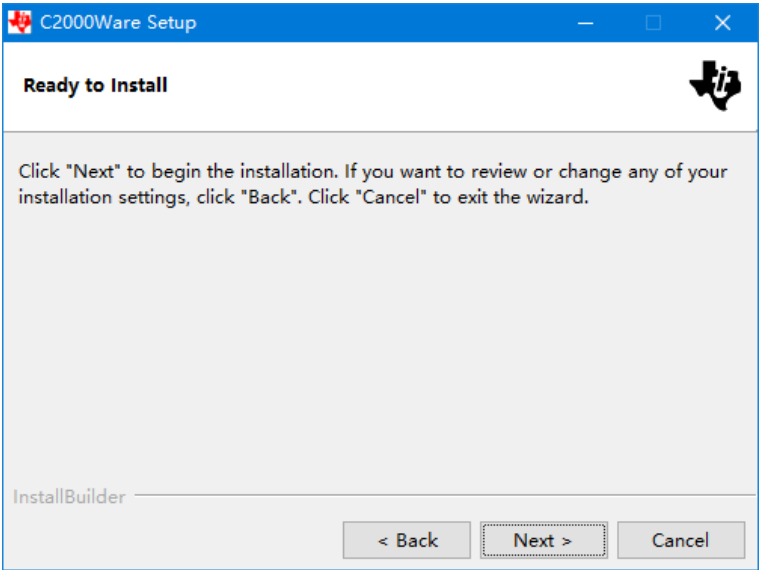


图 16

点击 Next，弹出如下安装进度界面。

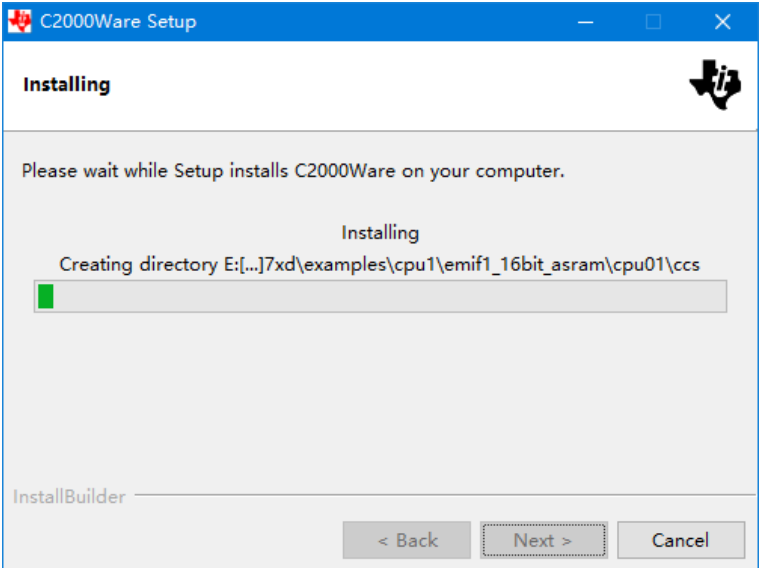


图 17

安装完成，弹出如下界面。

因我们的存在，让嵌入式应用更简单

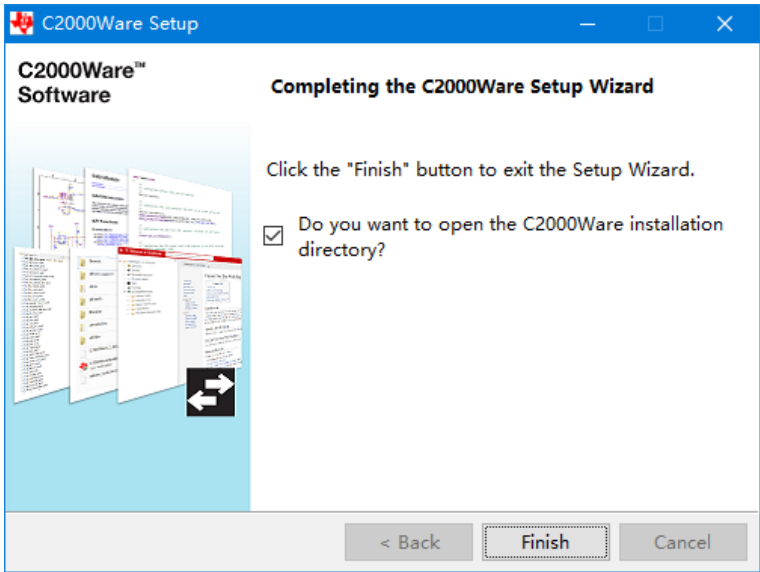


图 18

点击 Finish，至此安装完成。在 C2000Ware 安装路径下可查看 C2000Ware\_4\_01\_00\_00 组件内容。

本地磁盘 (C:) > ti > c2000				搜索 "c2000"	
名称	修改日期	类型	大小		
C2000Ware_1_00_06_00	2022/8/25 16:27	文件夹			
C2000Ware_4_01_00_00	2022/9/15 8:52	文件夹			

图 19

2.3 安装 TDM-GCC

TDM-GCC 为 GNU 编译器。

双击产品资料 “4-软件资料\Tools\Windows\” 目录下的 tdm64-gcc-5.1.0-2.exe 程序，弹出如下界面。

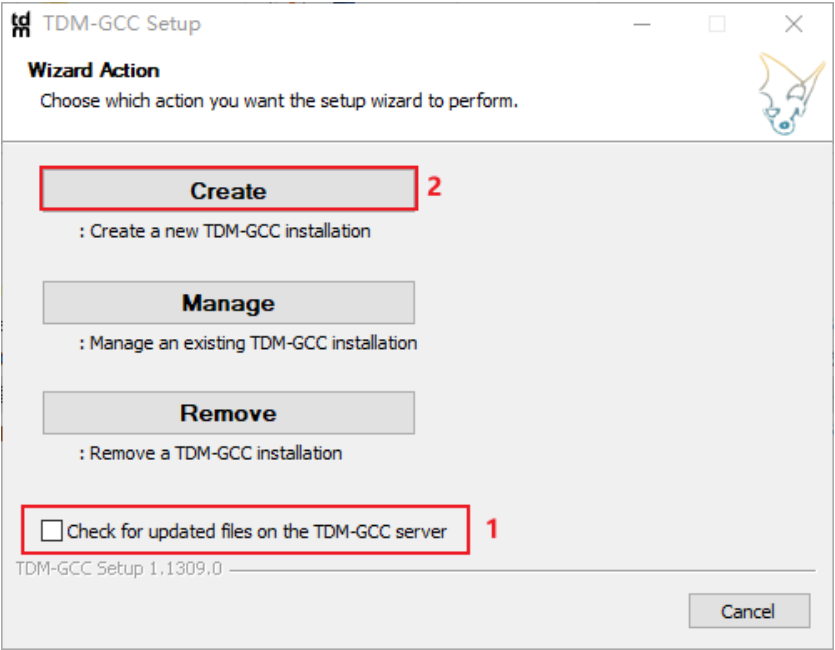


图 20

取消"Check for updated files on the TDM-GCC server"选项，点击 Create，弹出如下界面。

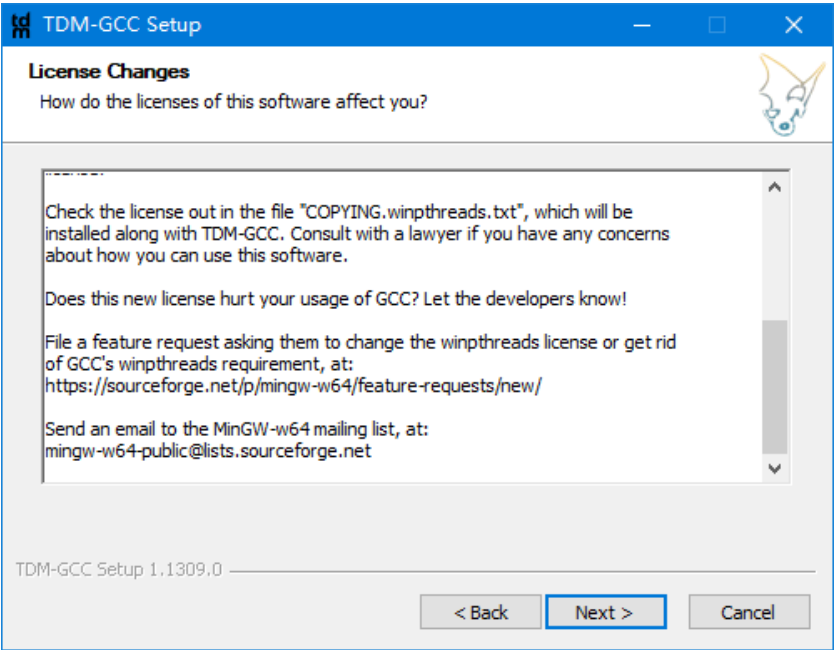


图 21

点击 Next，弹出如下界面。

因我们的存在，让嵌入式应用更简单

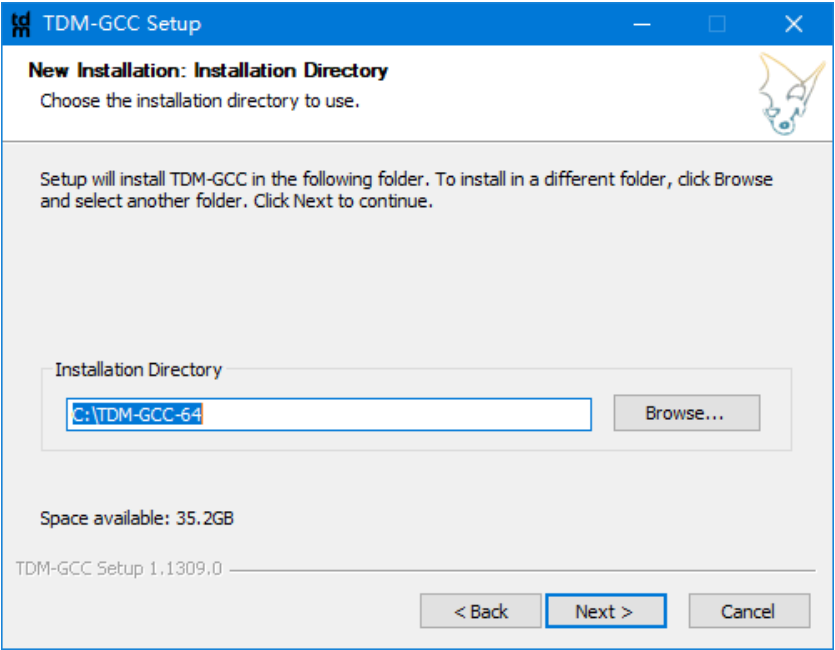


图 22

点击"Browse...", 选择安装路径, 点击 Next, 弹出如下界面。

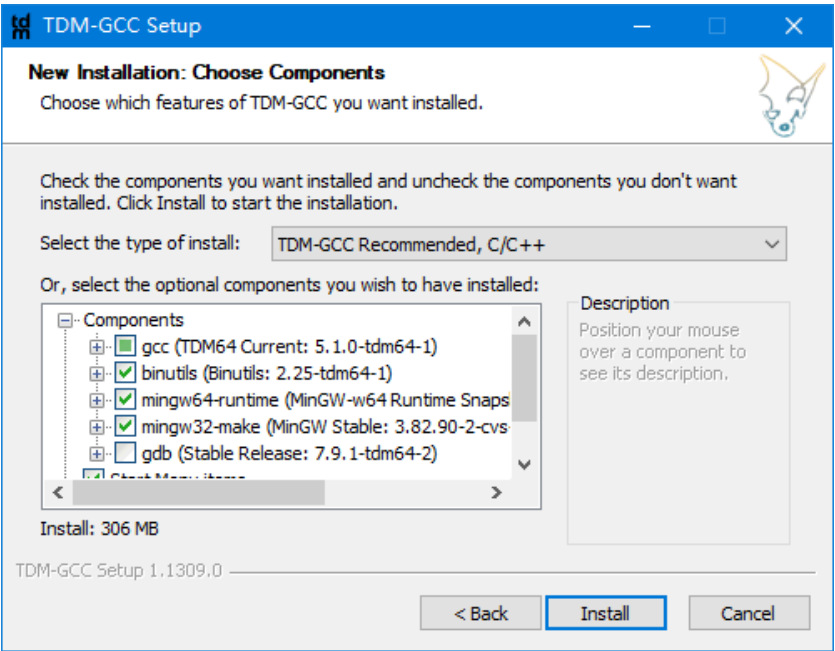


图 23

点击 Install, 弹出如下界面。

因我们的存在，让嵌入式应用更简单



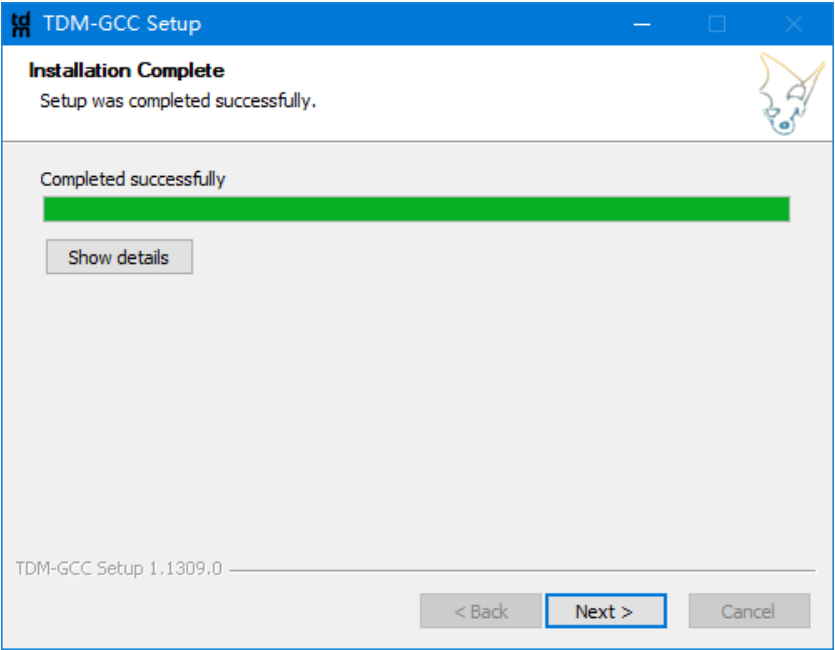


图 24

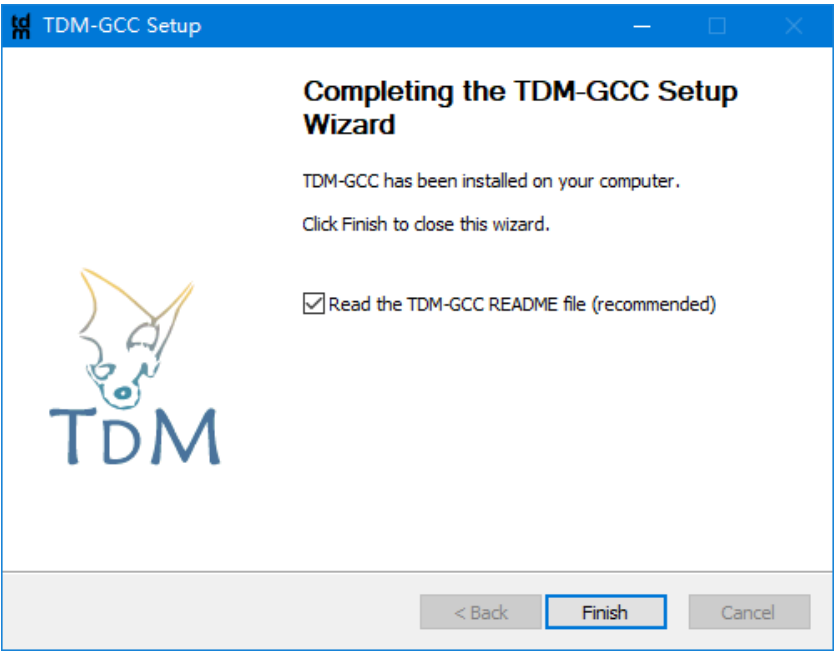


图 25

点击 Finish，安装完成。

### 3 clb\_external\_signal\_AND\_gate2 案例

因我们的存在，让嵌入式应用更简单

3.1 案例说明

案例功能：通过输入交叉开关和 CLB 交叉开关，将 2 个 GPIO(GPIO160、GPIO161)的外部信号传递至 CLB 逻辑块内部进行与操作，并通过输出交叉开关将结果输出至 GPIO24。

3.2 案例测试

将评估板断电，请使用杜邦线将评估底板 J6 接口的 pin 16(GPIO 160)和 pin 17(GPIO 161)分别连接至 3.3V 或 GND，可直接使用评估底板 J6 的 VDD\_3V3\_MAIN 和 GND 引脚。**请务必正确连接，以免损坏核心板。**

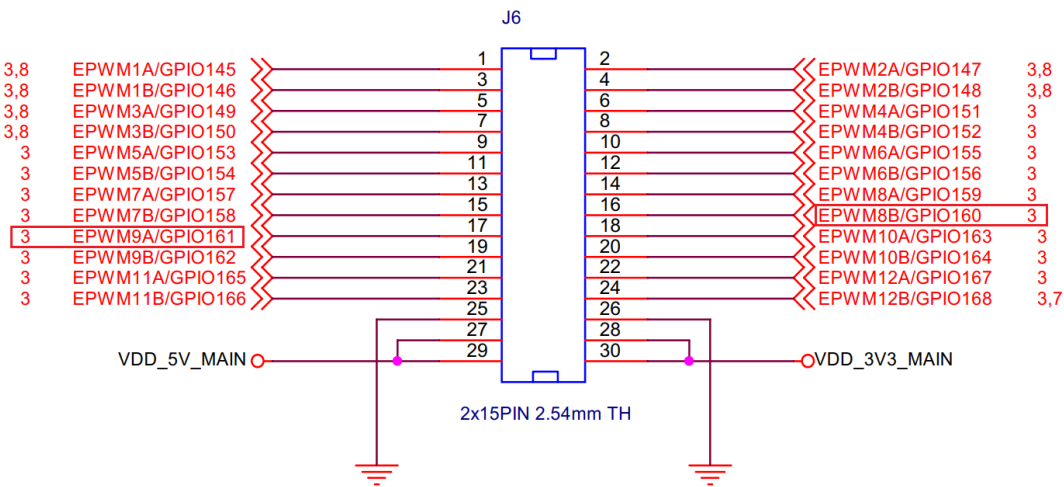


图 26

硬件正常连接后，请将评估板启动并加载程序至 CPU1 核心运行。

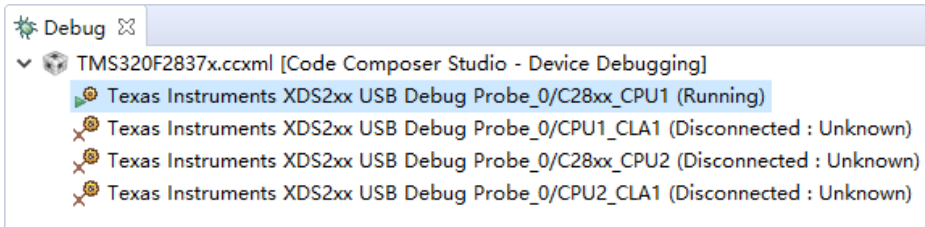


图 27

使用万用表测试评估底板 J7 接口的 pin 16(GPIO24)电平，GPIO24 引脚输出电平为 GPIO160 与 GPIO161 引脚输入电平相与结果。

因我们的存在，让嵌入式应用更简单

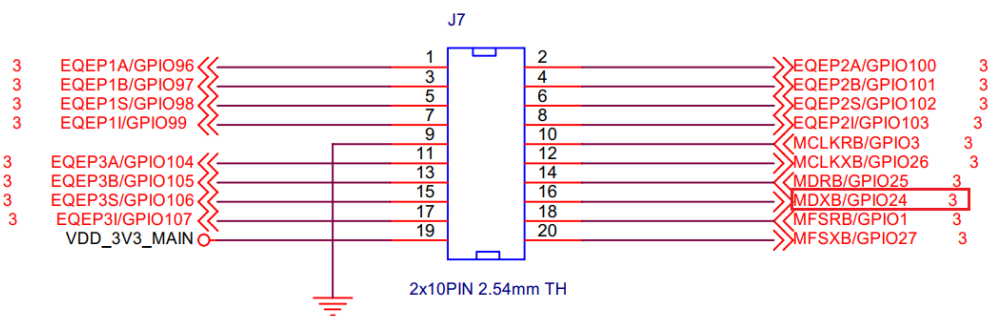


图 28

如 GPIO160 与 GPIO161 均连接至 3.3V, 则 GPIO24 引脚输出电平为 3.344V; 如 GPIO160 和 GPIO161 其中 1 个引脚连接至 GND, 则 GPIO24 引脚输出电平为 0V, 测试结果仅供参考, 如下图所示。



图 29



图 30

因我们的存在，让嵌入式应用更简单

3.3 案例编译

双击桌面快捷方式 Code Composer Studio 12.0.0，打开 CCS 软件并导入工程。  
右键点击工程，选择 Properties。

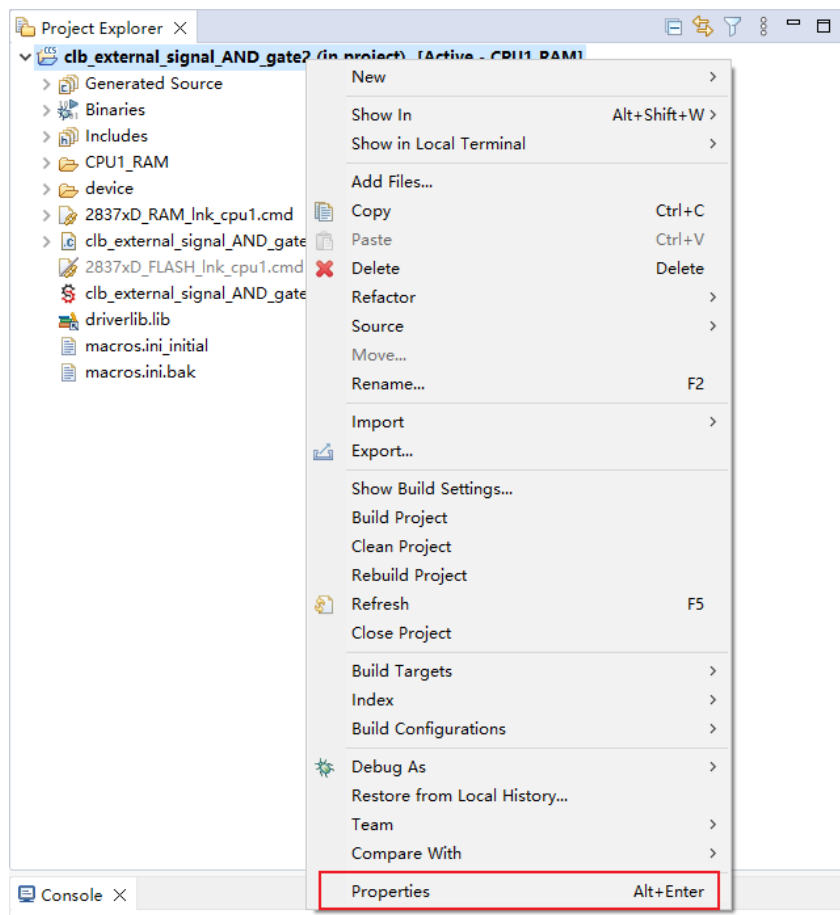


图 31

依次点击"Resource -> Linked Resources"，根据实际情况，修改 CLB\_SIM\_COMPILER、COM\_TI\_C2000WARE\_SOFTWARE\_PACKAGE\_INSTALL\_DIR 路径。  
CLB\_SIM\_COMPILER: TDM-GCC 编译器 bin 目录安装路径。  
COM\_TI\_C2000WARE\_SOFTWARE\_PACKAGE\_INSTALL\_DIR: C2000Ware\_4\_01\_00\_00 安装路径。

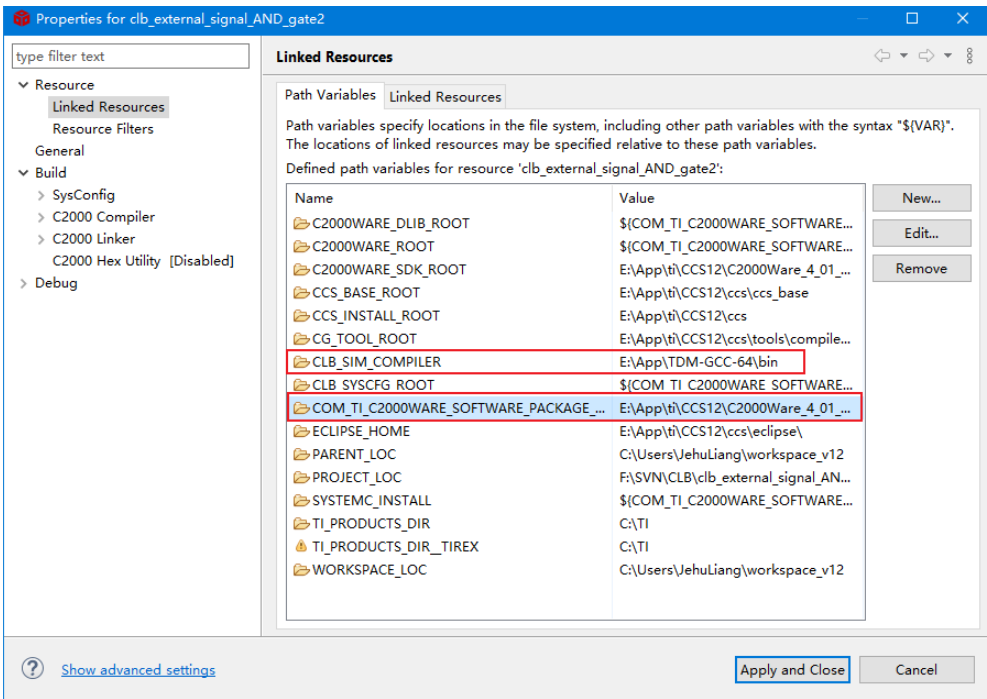


图 32

配置完成，右键点击工程，选择"Build Project"或"Rebuild Project"编译。

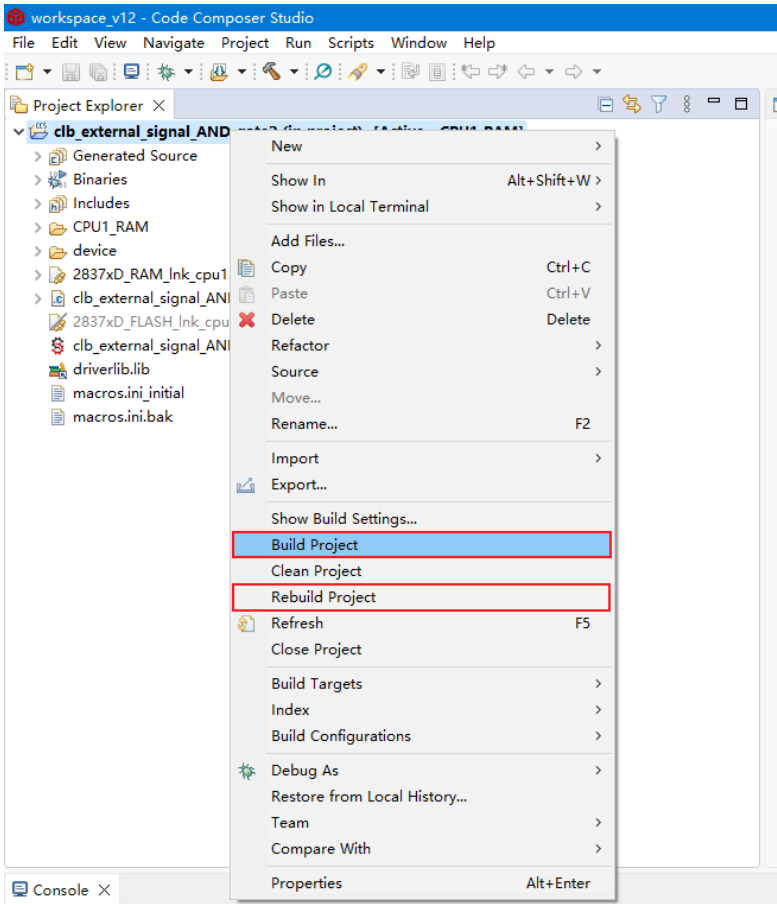


图 33

编译完成，CCS 控制台打印如下信息。



图 34

### 3.4 关键代码

- (1) 主函数：调用 Board\_init 函数，完成 CLB 逻辑块的配置和连接。

因我们的存在，让嵌入式应用更简单

```
26 void main(void)
27 {
28     Device_init();
29     Device_initGPIO();
30
31     Interrupt_initModule();
32     Interrupt_initVectorTable();
33
34     /* Enabling EPWM1 to enable CLB1 */
35     SysCtl_enablePeripheral(SYSCTL_PERIPH_CLK_EPWM1);
36
37     /* device initialization */
38     Board_init();
39
40     /* CLB initialization */
41     initTILE1(myCLBTILE1_BASE);
42     CLB_enableCLB(myCLBTILE1_BASE);
43
44     while(1) {
45         asm(" NOP");
46     }
47 }
```

图 35

- (2) Board\_init 函数在 board.c 文件中定义。该源码文件为 syscfg tools 根据 clb\_external\_signal\_AND\_gate.syscfg 生成的源码文件。如需修改该源码文件，需修改 syscfg 中配置重新生成源码文件。
- (3) clb\_external\_signal\_AND\_gate.syscfg 文件根据如下框图，配置 GPIO 作为 CLB 的输入和输出。

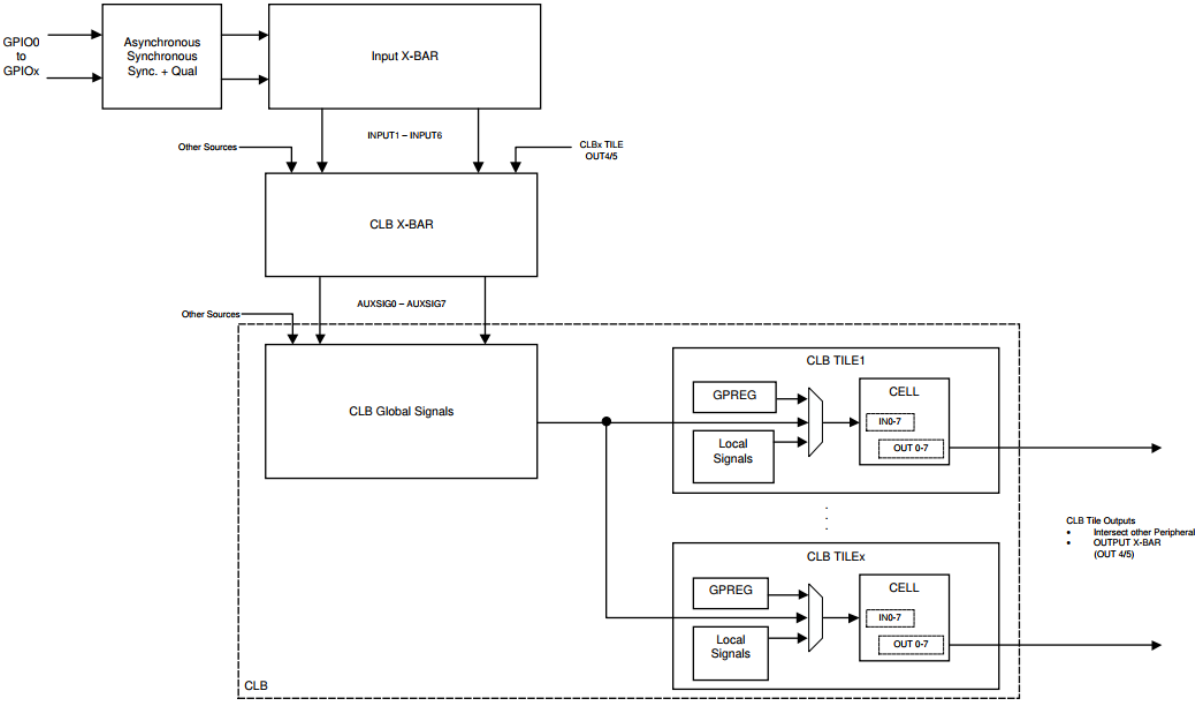


Figure 29-5. GPIO to CLB Tile Connections

<https://blog.csdn.net/ngany>

图 36

a) 配置 GPIO0 和 GPIO1 为输入模式。

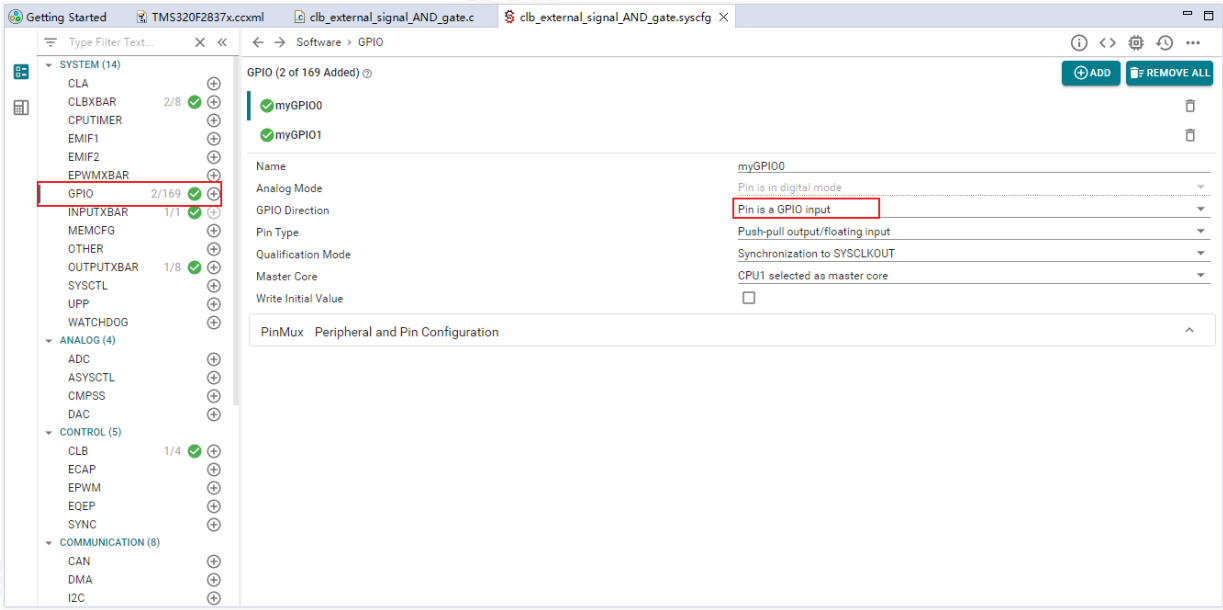


图 37

因我们的存在，让嵌入式应用更简单



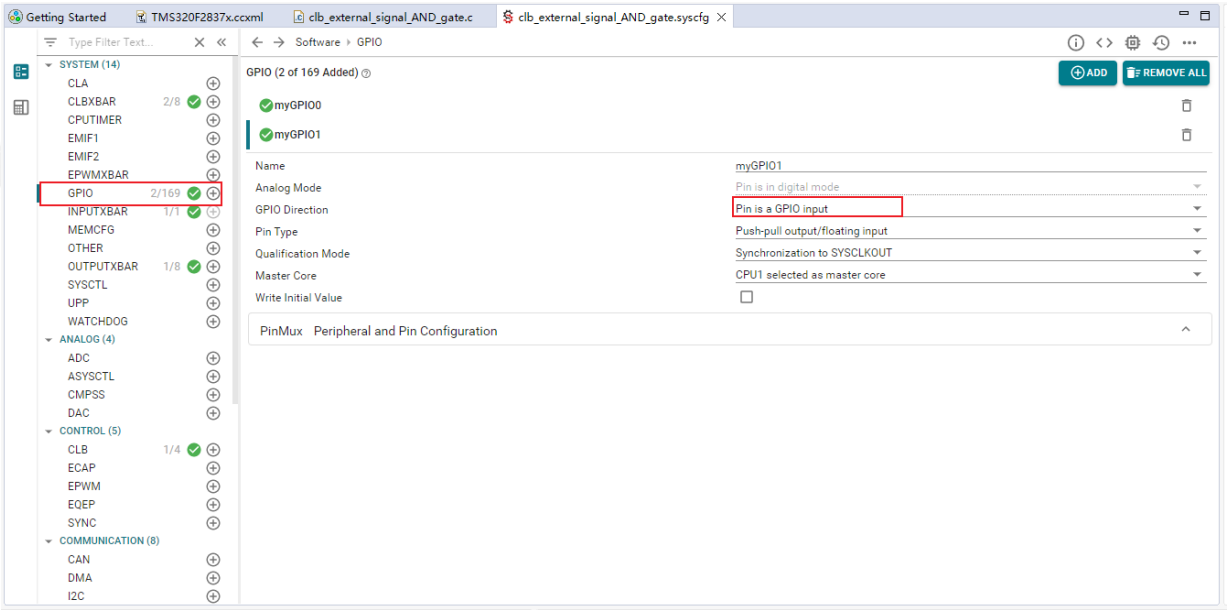


图 38

b) 指定 INPUTXBAR1 为 GPIO160 引脚，INPUTXBAR2 为 GPIO161 引脚。

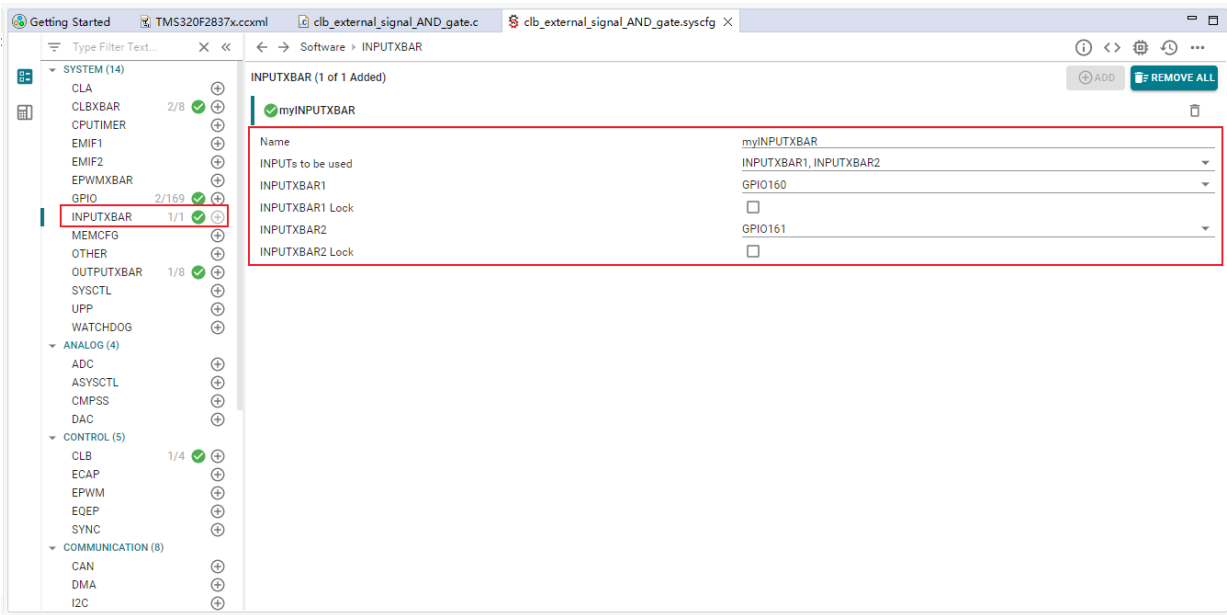


图 39

c) 将 INPUTXBAR1 和 INPUTXBAR2 分别映射到 AUXSIG0 和 AUXSIG1。

因我们的存在，让嵌入式应用更简单

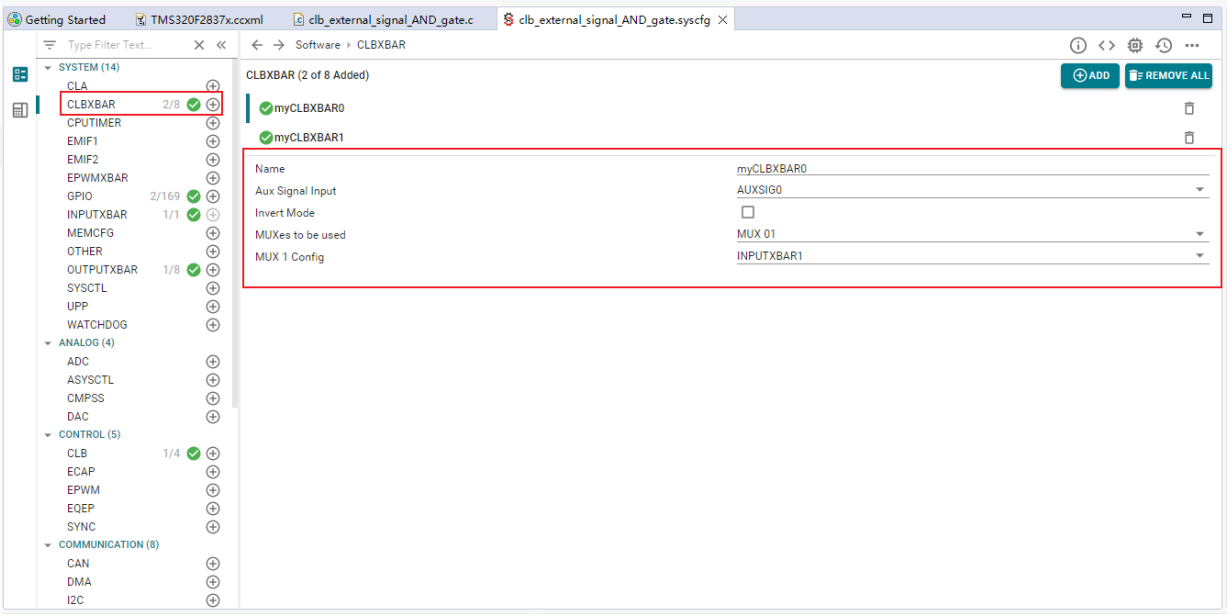


图 40

d) 设置输入引脚为 OUTPUT\_XBAR1，使用 CLB1 OUT4 逻辑结果作为 OUTPUT\_XBAR1 的输出电平。

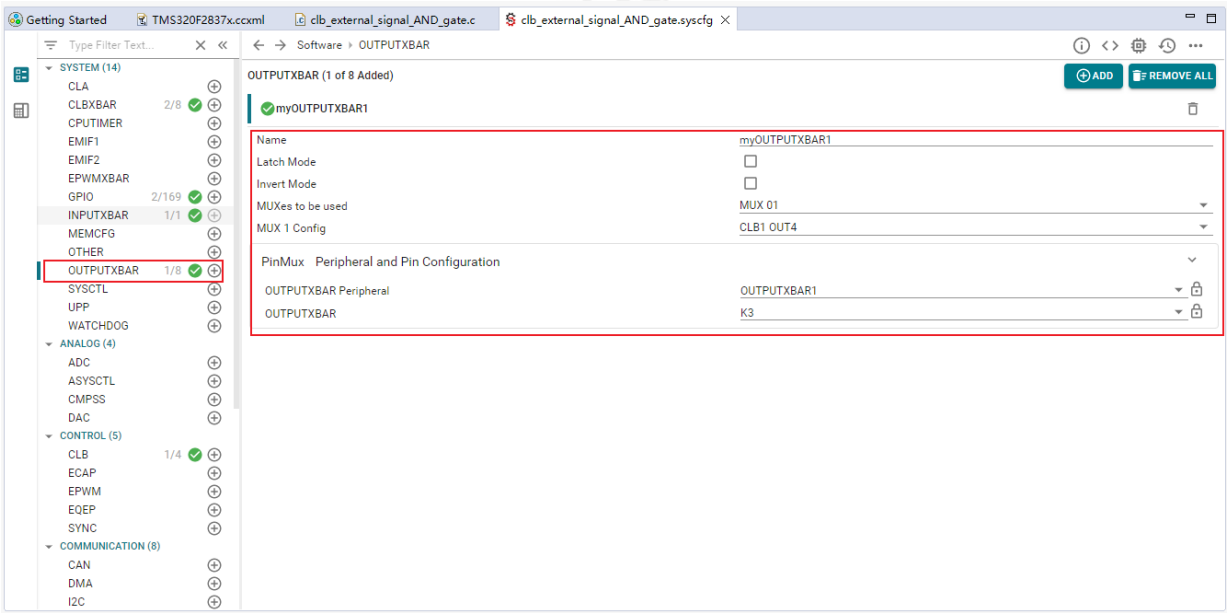


图 41

e) 将 AUXSIG0 和 AUXSIG1 作为 LUT\_0 的输入，LUT\_0 输出为两个输入的相与结果。

因我们的存在，让嵌入式应用更简单

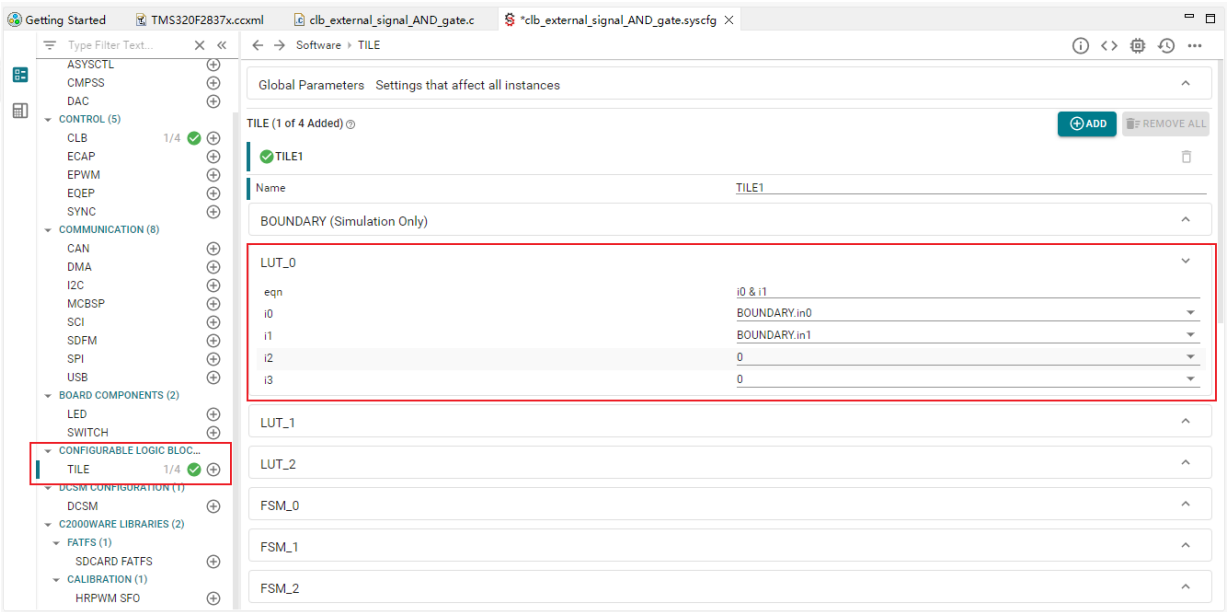


图 42

- f) 将 LUT\_0 输出结果作为 CLB1 OUTLUT\_4 的输入，并将其送至 CLB1 OUTLUT\_4 的输出；  
CLB1 OUTLUT\_4 逻辑结果作为 OUTPUTXBAR1 的输出电平。

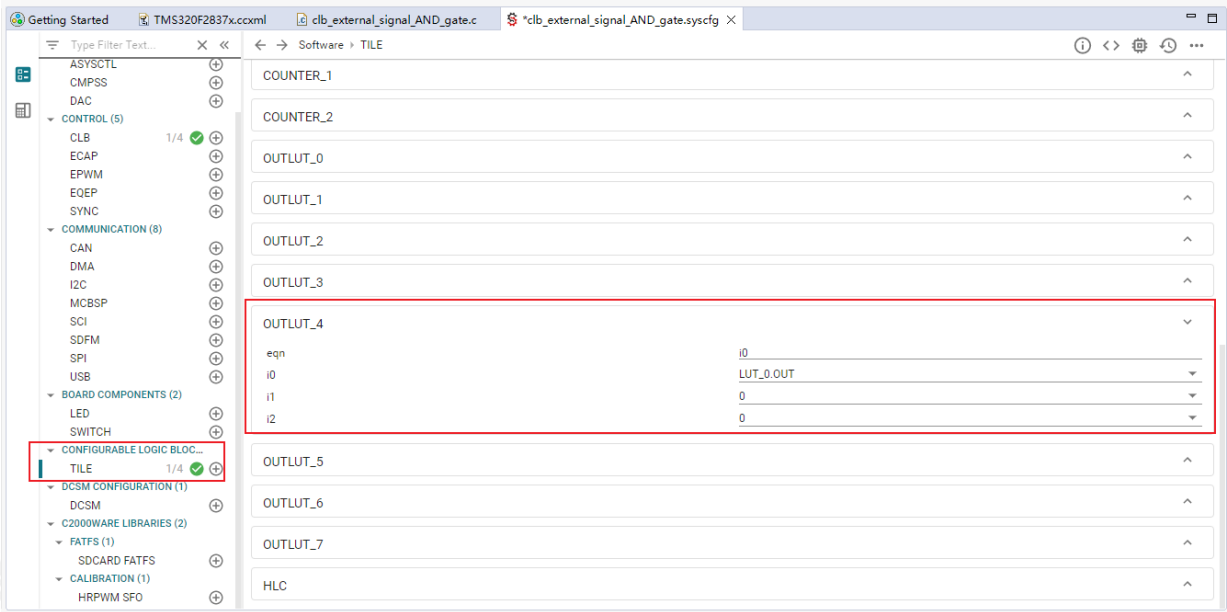


图 43

因我们的存在，让嵌入式应用更简单

## 更多帮助

销售邮箱: [sales@tronlong.com](mailto:sales@tronlong.com)

技术邮箱: [support@tronlong.com](mailto:support@tronlong.com)

创龙总机: 020-8998-6280

技术热线: 020-3893-9734

创龙官网: [www.tronlong.com](http://www.tronlong.com)

技术论坛: [www.51ele.net](http://www.51ele.net)

官方商城: <https://tronlong.tmall.com>