

# 基于 TencentOS tiny 的 LoRaWAN 开发入门指南



作者	supowang	时间	2019-12-18
审核	sheldondai	时间	2019-12-19
批准	alexguo	时间	2019-12-20



深圳市腾讯计算机系统有限公司

# 目录

1.	ST P-NUCLEO-LRWAN3 套件简介 .....	3
1.1	开发套件简介 .....	3
1.2	LoRaWAN 节点硬件详解 .....	4
1.3.1	NUCLEO-L073RZ 基于 CORTEX-M0+ 的评估板 .....	4
1.3.2	LRWAN_NS1 扩展板硬件 .....	5
2.	LoRaWAN 节点软件开发环境准备 .....	7
2.1	MDK 软件介绍 .....	7
2.2	MDK 安装 .....	8
2.3	Pack 安装 .....	12
2.4	ST-Link 驱动安装 .....	13
2.5	编程软件(MDK)配置 .....	15
2.6	串口调试助手的安装与使用 .....	17
3.	TencentOS tiny 基础内核实验 .....	18
3.1	打开 TencentOS tiny 提供的 Hello_world 工程 .....	18
3.2	编译 HelloWorld 工程 .....	18
3.3	下载运行 .....	19
3.4	查看运行结果 .....	21
4.	使用 TencentOS tiny AT 框架驱动 LoRaWAN 节点对接腾讯云 IoT Explorer .....	22
4.1	打开 TencentOS tiny 提供的 LoRaWAN 例程 .....	22
4.2	示例工程代码简介 .....	23

# 1. ST P-NUCLEO-LRWAN3 套件简介

## 1.1 开发套件简介

P-NUCLEO-LRWAN3 是完整的套件，包括网关和节点，可用于评估 LoRaWAN 网络。使用该套件，用户可以轻松设置 LPWAN 网络，帮助用户学习 LoRaWAN 技术，了解如何在自己的应用程序中使用 LoRaWAN 技术。

LoRa 网关套件由 ST Nucleo-F746ZG 底板和瑞兴恒方 (www.risinghf.com) 基于 SX1301 的 LRWAN\_GS 模块组成。

ST Nucleo LoRa 节点套件由 LRWAN\_NS1 扩展板和 ST Nucleo-L073 底板组成。其中 LRWAN\_NS1 扩展板集成瑞兴恒方的 RHF0M003 LoRaWAN 模组，并集成了温湿度传感器 HTS221、气压传感器 LPS22HB、3 轴磁力传感器 LIS3MDL、6 轴姿态传感器 LSM6DS3 共 4 个 I2C 传感器器件。

开发套件实物图如下：

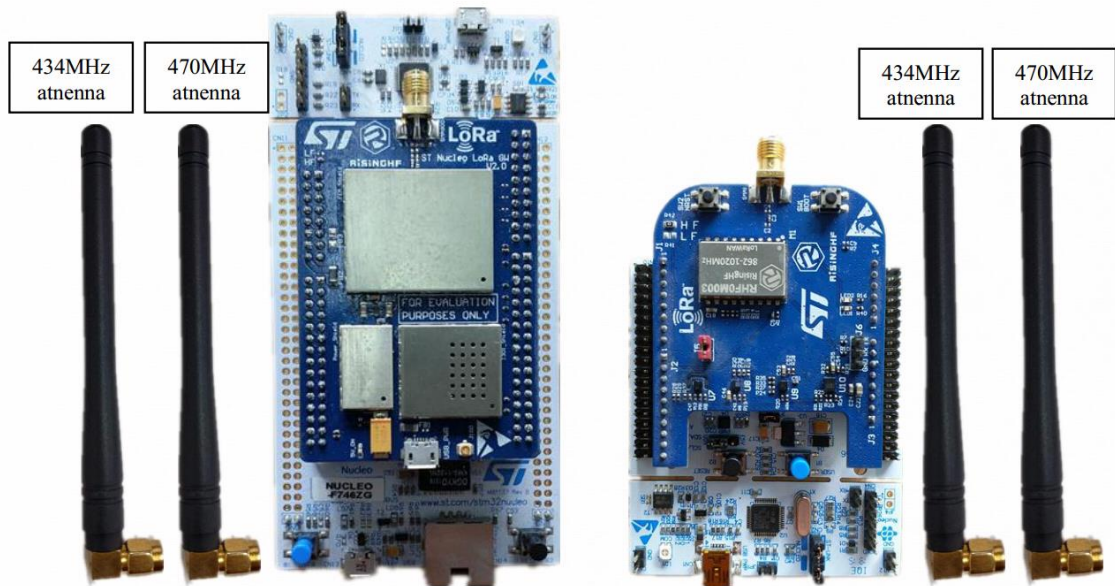
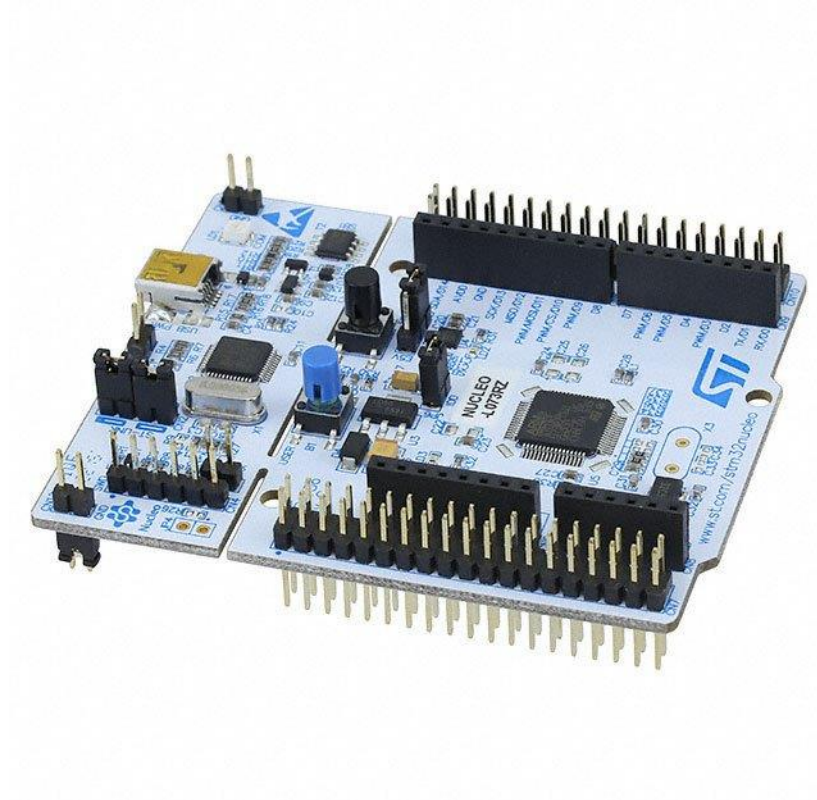


Figure 1 P-NUCLEO-LRWAN3 kit

## 1.2 LoRaWAN 节点硬件详解

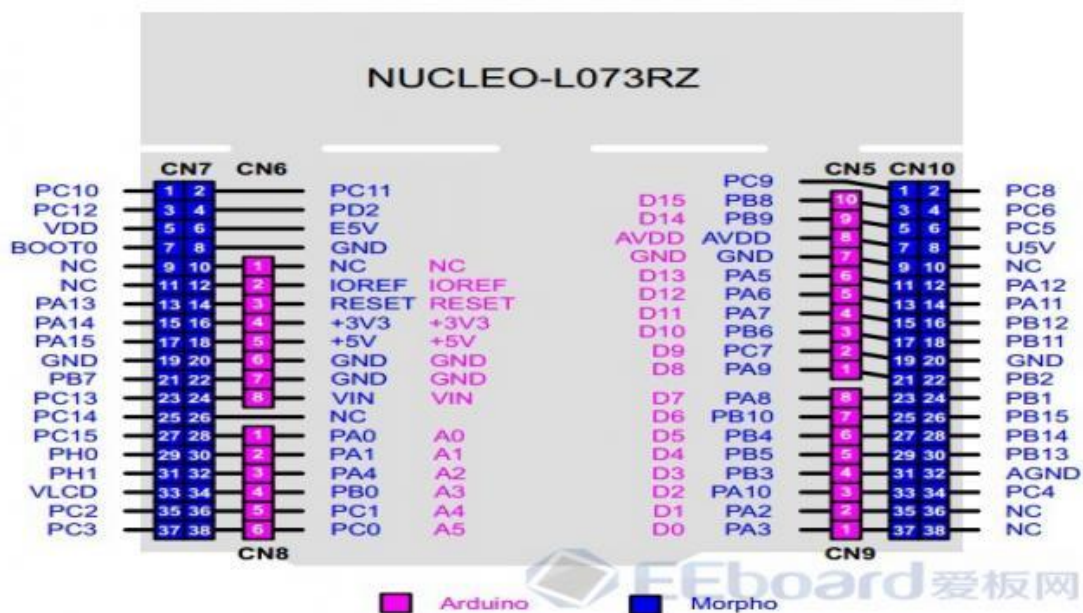
### 1.3.1 NUCLEO-L073RZ 基于 CORTEX-M0+的评估板



NUCLEO-L073RZ 基于 NUCLEO 64 公版 PCB 设计, 可以方便的更换 ST 的其它 64PIN MCU。  
主要特性包括：

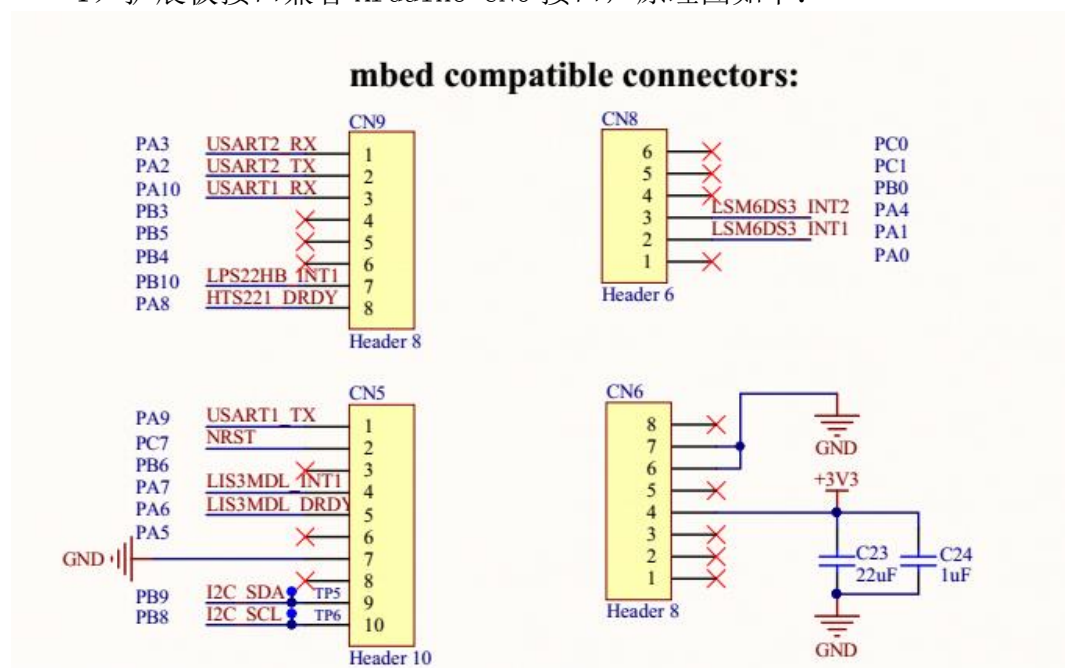
- LQFP64 STM32-L073RZT6 MCU
- 20KB SRAM 192KB Flash 6KB data EEPROM
- 两种类型扩展接口：Arduino UNO 扩展接口；STM32 MORPHO 扩展接口
- 板载 ST-LINK/V2-1 调试/编程器
- 灵活的供电方式选择
- 板载 3 个 LED 灯，LD1、LD3 指示工作状态，LD2 为用户可编程 LED 灯
- 两个按钮，复位及用户可编程按钮
- USB 重枚举能力，支持 VCP，MSD，DEBUG 功能
- 大量 IDE 开发环境支持

扩展接口如下图所示：

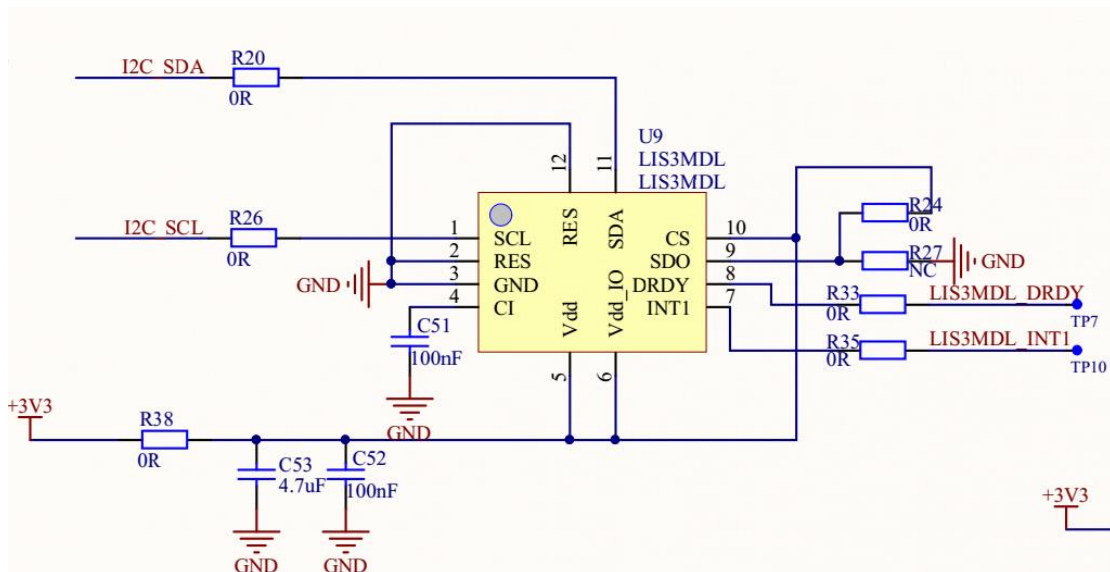
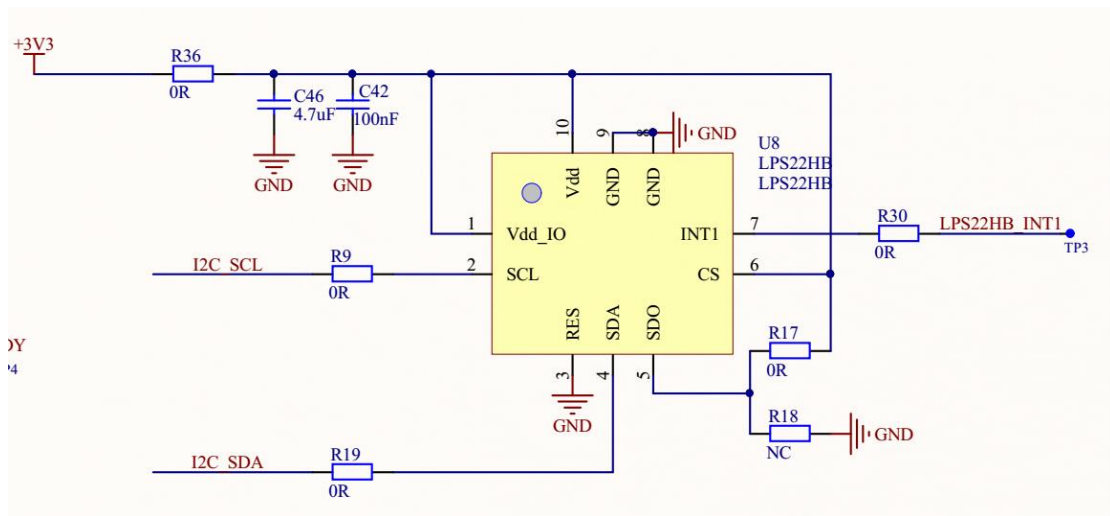
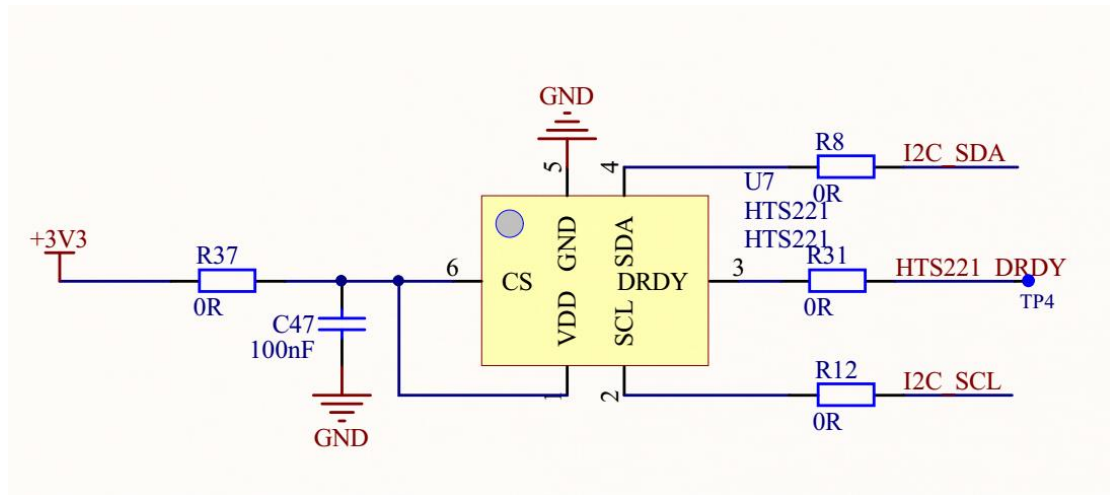


### 1.3.2 LRWAN\_NS1 扩展板硬件

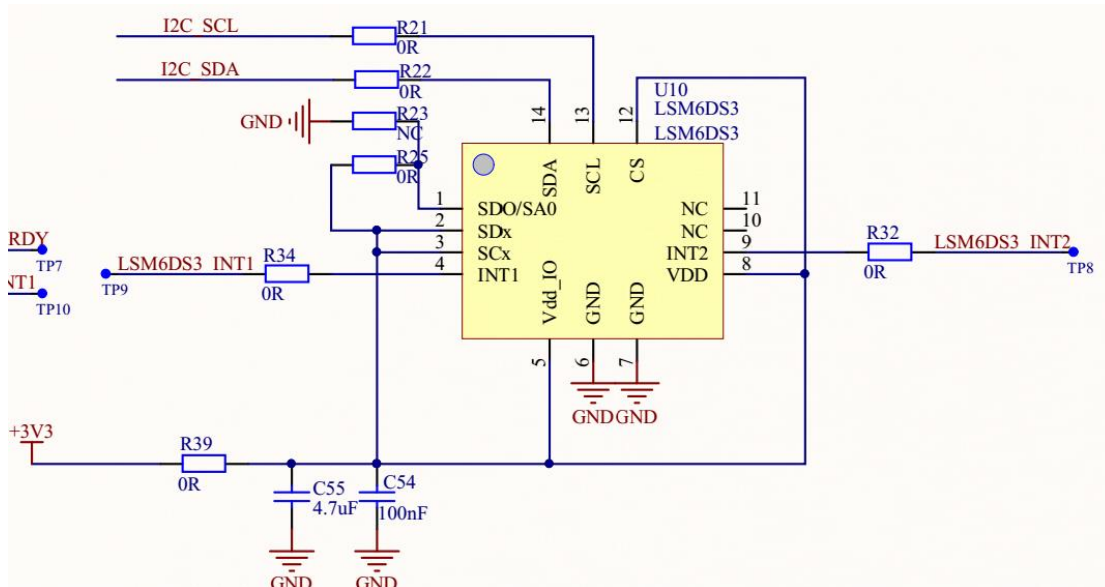
1) 扩展板接口兼容 Arduino UNO 接口，原理图如下：



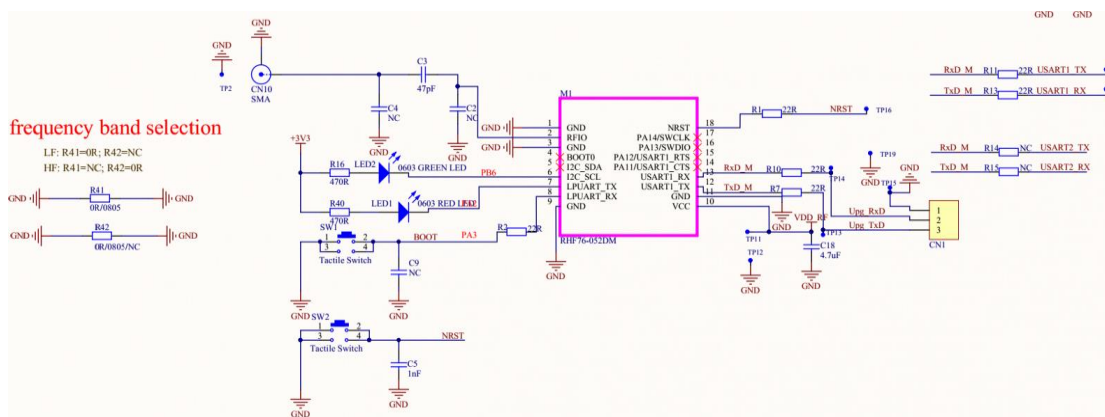
2) 扩展板上同一个 I2C 端口，挂载了 4 个 I2C 接口的传感器，分别是温湿度传感器 HTS221、气压传感器 LPS22HB、3 轴磁力传感器 LIS3MDL、6 轴姿态传感器 LSM6DS3，原理图如下：







3) LoRaWAN 通信模组电路原理图



## 2. LoRaWAN 节点软件开发环境准备

### 2.1 MDK 软件介绍

MDK 即 RealView MDK 或 MDK-ARM (Microcontroller Development kit), 是 ARM 公司收购 Keil 公司以后, 基于 uVision 界面推出的针对 ARM7、ARM9、Cortex-M0、Cortex-M1、Cortex-M2、Cortex-M3、Cortex-M4 等 ARM 处理器的嵌入式软件开发工具。MDK-ARM 集成了业内最领先的技术, 包括 uVision4 集成开发环境与 RealView 编译器 RVCT。支持 ARM7、ARM9 和最新的 Cortex-M3/M1/M0 核处理器, 自动配置启动代码, 集成 Flash 烧写模块, 强大的 Simulation 设备模拟, 性能分析等功能, 与 ARM 之前的工具包 ADS 等相比, RealView 编译器的最新版本可将性能改善超过 20%。

Keil 公司开发的 ARM 开发工具 MDK, 是用来开发基于 ARM 核的系列微控制器的嵌入式应用程序。它适合不同层次的开发者使用, 包括专业的应用程序开发工

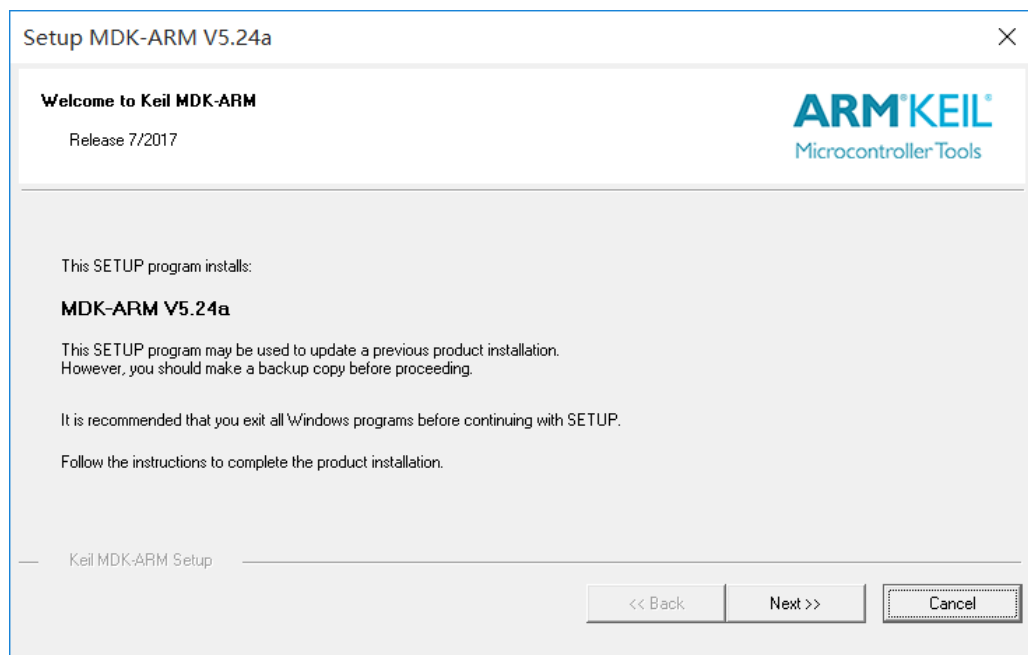
程师和嵌入式软件开发的入门者。MDK 包含了工业标准的 Keil C 编译器、宏汇编器、调试器、实时内核等组件，支持所有基于 ARM 的设备，能帮助工程师按照计划完成项目。

## 2.2 MDK 安装

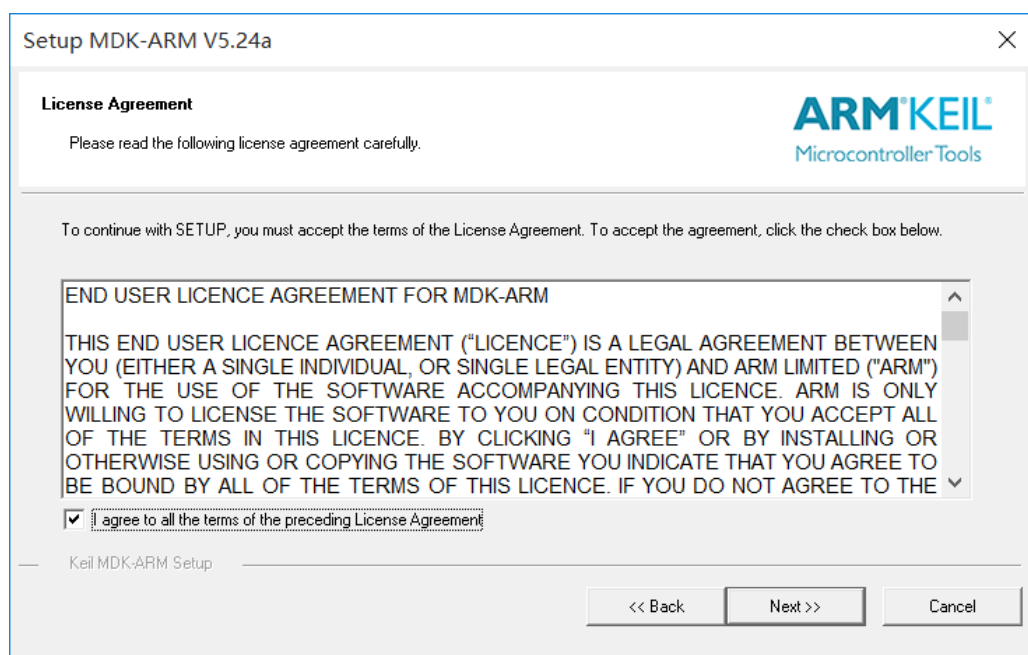
点击下面的地址下载 MDK 软件，软件版本最好是 5.24 及以上：

<https://www.keil.com/download/product/>

教程以 5.24 版本为例，双击 MDK524 应用程序文件，点击 next>>。

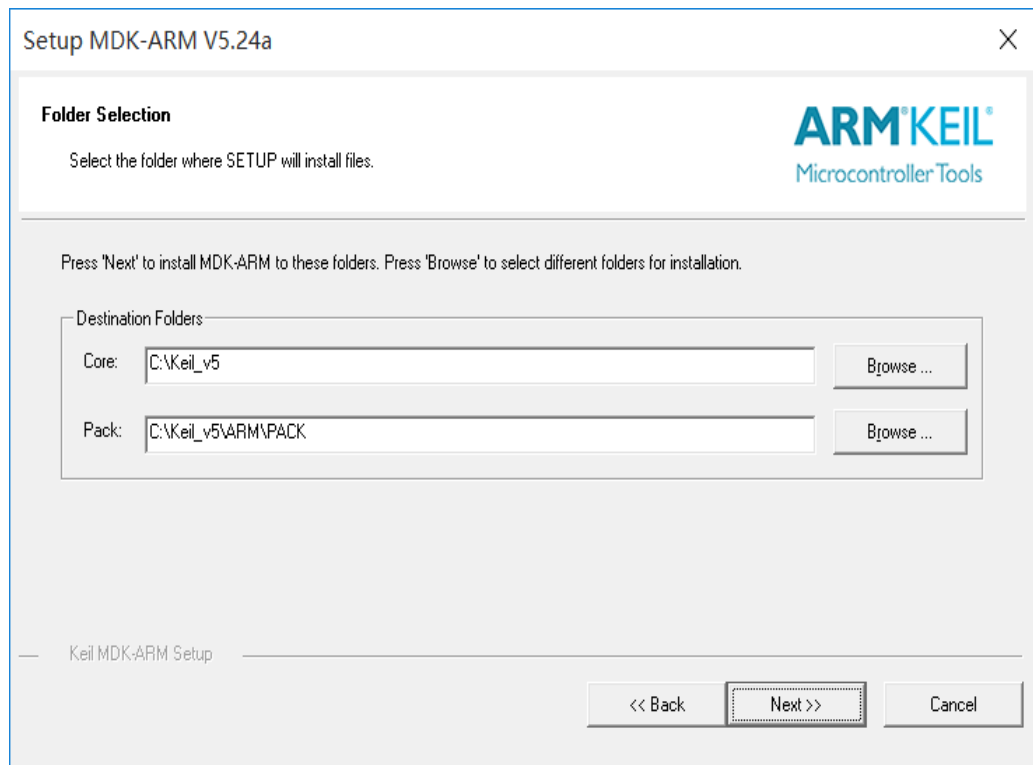


打上 I agree 前面的勾勾，即是同意一些安装协议。点击 next>>。



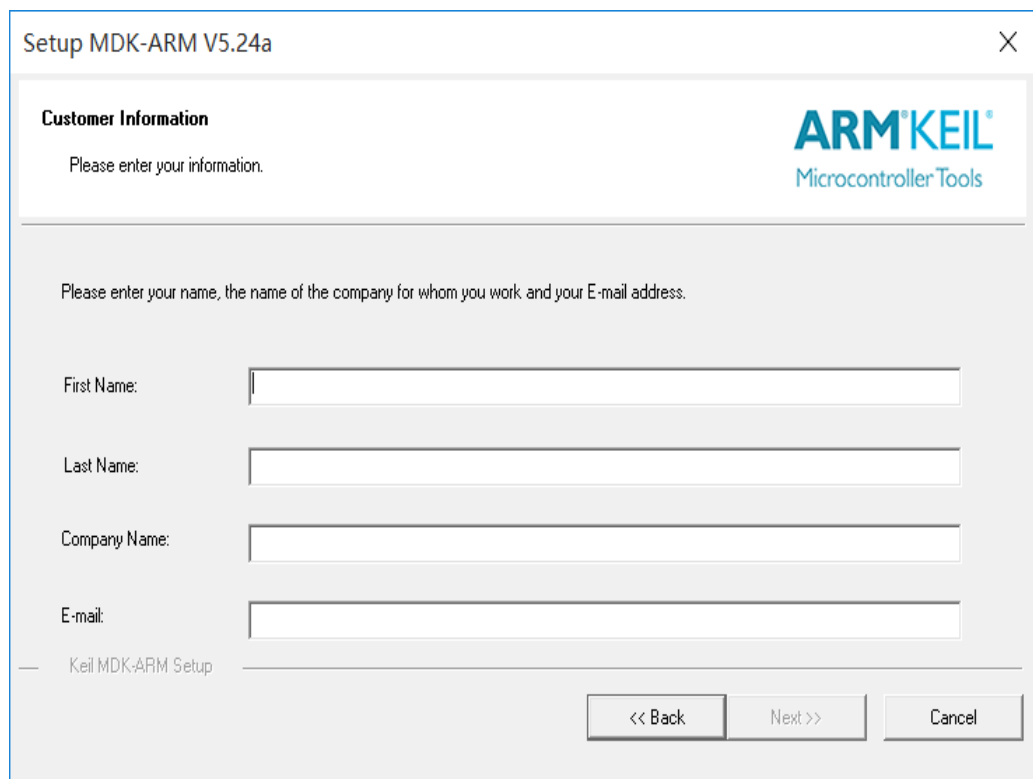


选择安装路径，可以默认也可以安装在我们自己建立的文件夹下。点击 next>>。



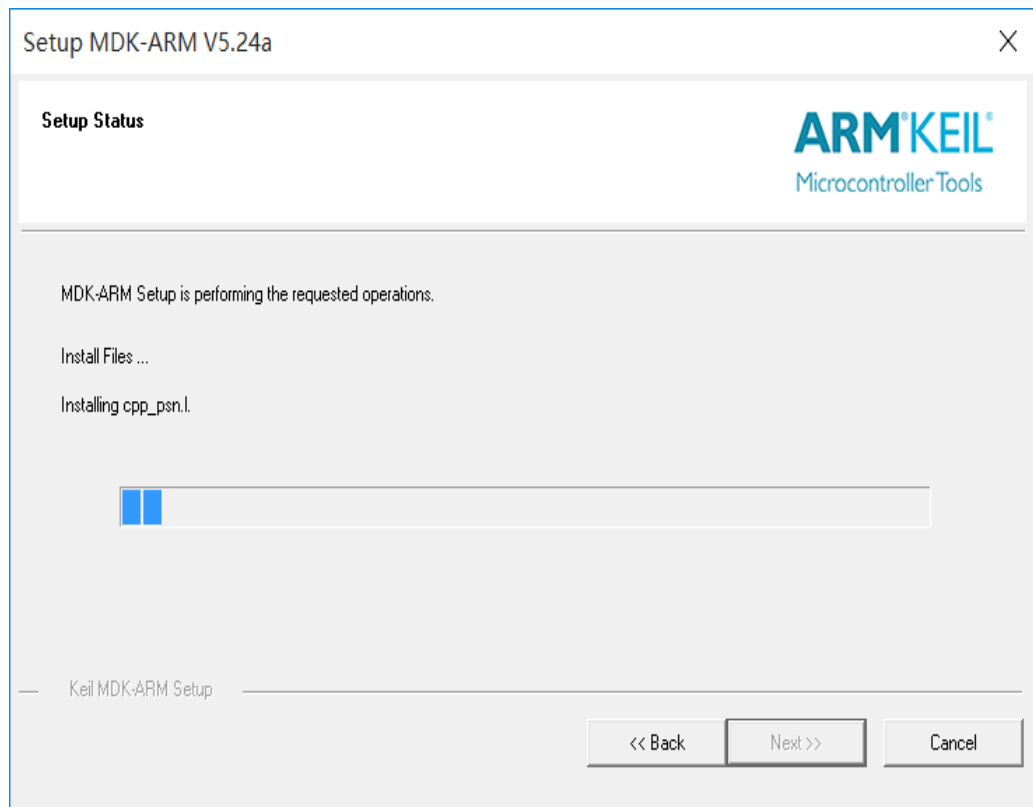
The screenshot shows the 'Folder Selection' window of the 'Setup MDK-ARM V5.24a' installer. The window has a title bar with the text 'Setup MDK-ARM V5.24a' and a close button. Below the title bar is a header section with the 'ARM KEIL Microcontroller Tools' logo. The main content area is titled 'Folder Selection' and contains the instruction: 'Select the folder where SETUP will install files.' Below this, a sub-instruction reads: 'Press 'Next' to install MDK-ARM to these folders. Press 'Browse' to select different folders for installation.' There are two input fields: 'Core:' with the value 'C:\Keil\_v5' and 'Pack:' with the value 'C:\Keil\_v5\ARM\PACK'. Each field has a 'Browse ...' button to its right. At the bottom of the window, there is a progress bar labeled 'Keil MDK-ARM Setup' and three buttons: '<< Back', 'Next >>' (which is highlighted with a dashed border), and 'Cancel'.

这里填写的是我们的一些信息，填写完整后，继续 next>>。

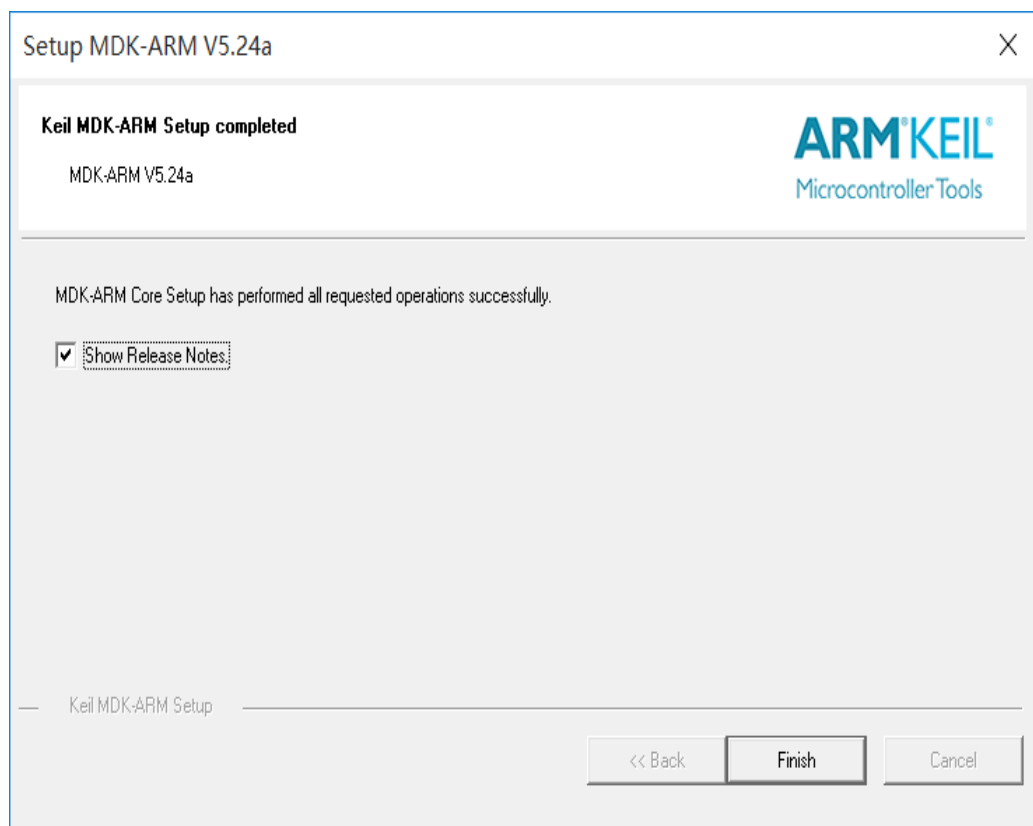


The screenshot shows the 'Customer Information' window of the 'Setup MDK-ARM V5.24a' installer. The window has a title bar with the text 'Setup MDK-ARM V5.24a' and a close button. Below the title bar is a header section with the 'ARM KEIL Microcontroller Tools' logo. The main content area is titled 'Customer Information' and contains the instruction: 'Please enter your information.' Below this, a sub-instruction reads: 'Please enter your name, the name of the company for whom you work and your E-mail address.' There are four input fields: 'First Name:', 'Last Name:', 'Company Name:', and 'E-mail:'. At the bottom of the window, there is a progress bar labeled 'Keil MDK-ARM Setup' and three buttons: '<< Back', 'Next >>', and 'Cancel'.

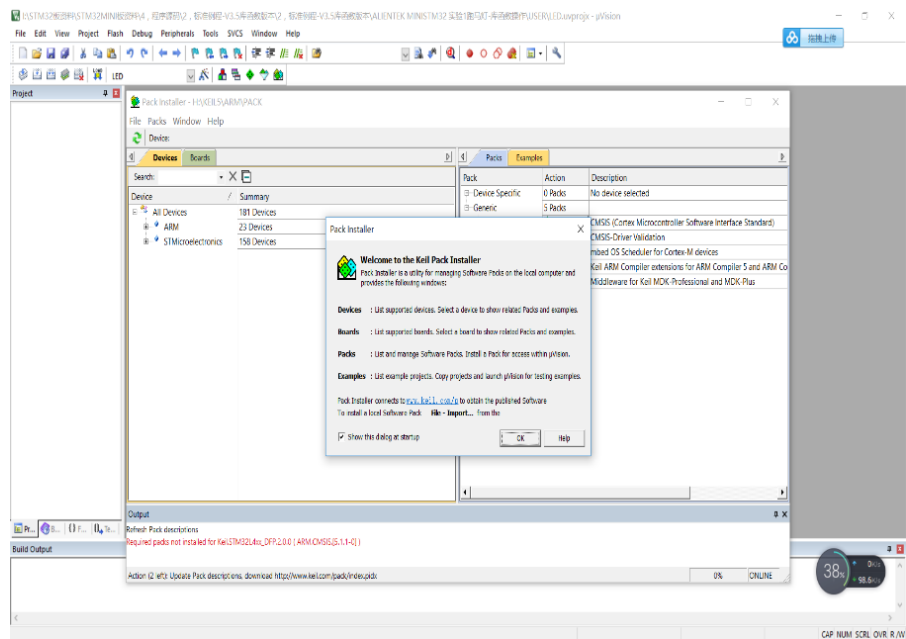
然后等待安装完成即可。



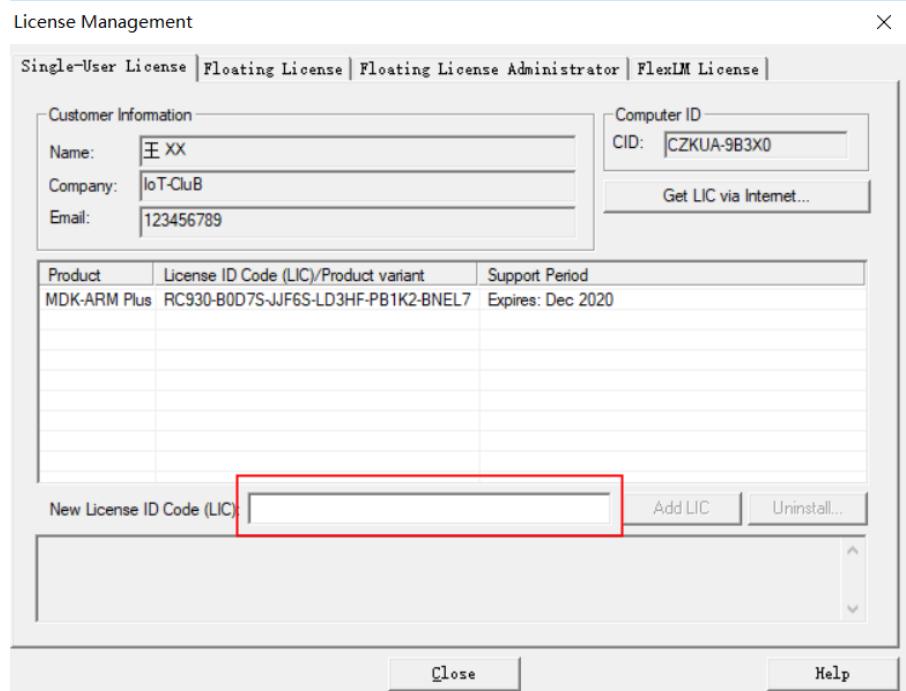
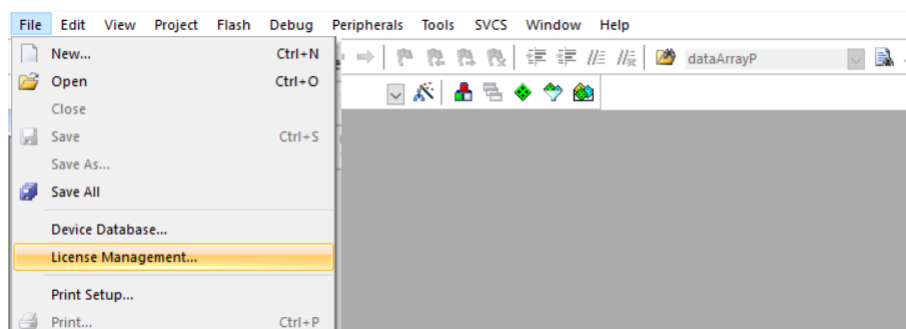
安装完成，点击 Finish。



然后会跳出来这个界面，这个我们后面再讲，先点 OK，把弹框都叉掉。



- 激活 MDK，导入 License, 激活 MDK 后便可使用了。



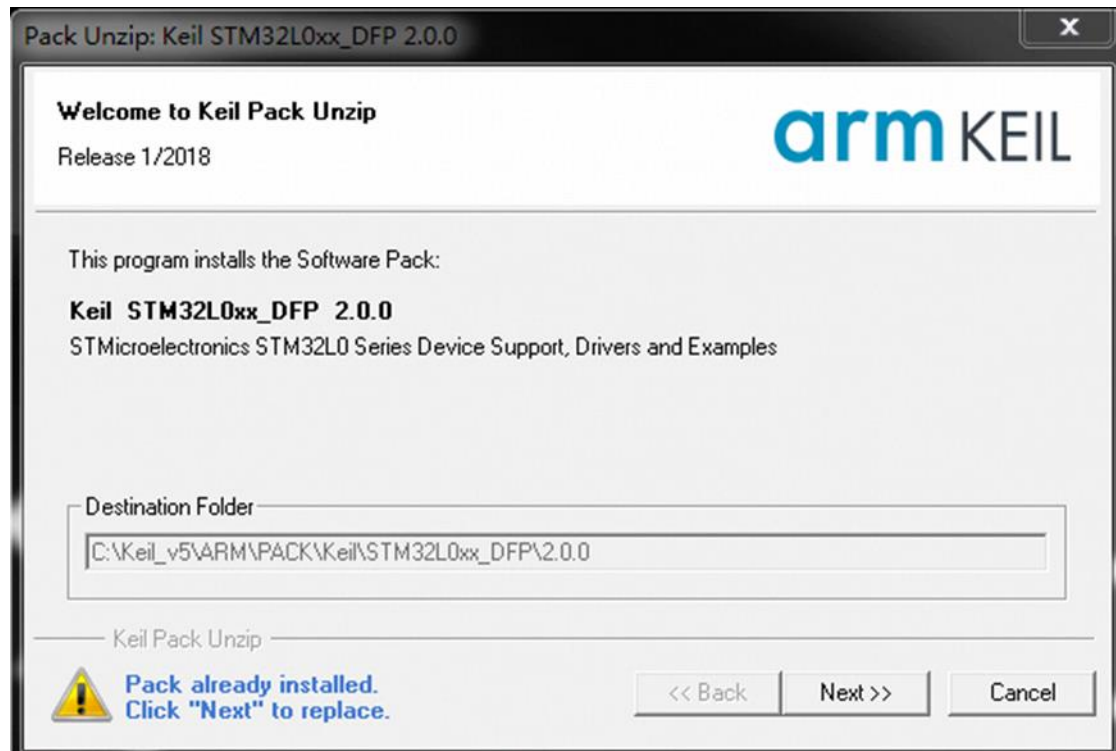
**特别提示：**一定要输入 License 激活 MDK 软件，建议购买正版 License。

## 2.3 Pack 安装

安装完 MDK 后，我们需要安装开发套件中单片机型号对应的 Pack。

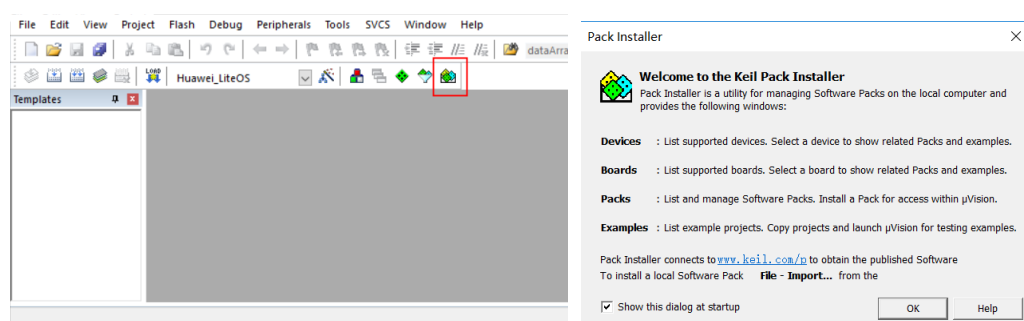
安装方式一：登录官网：<http://www.keil.com/dd2/pack/>

下载 Keil.STM32L0xx\_DFP.2.0.0.pack 后安装，如下图

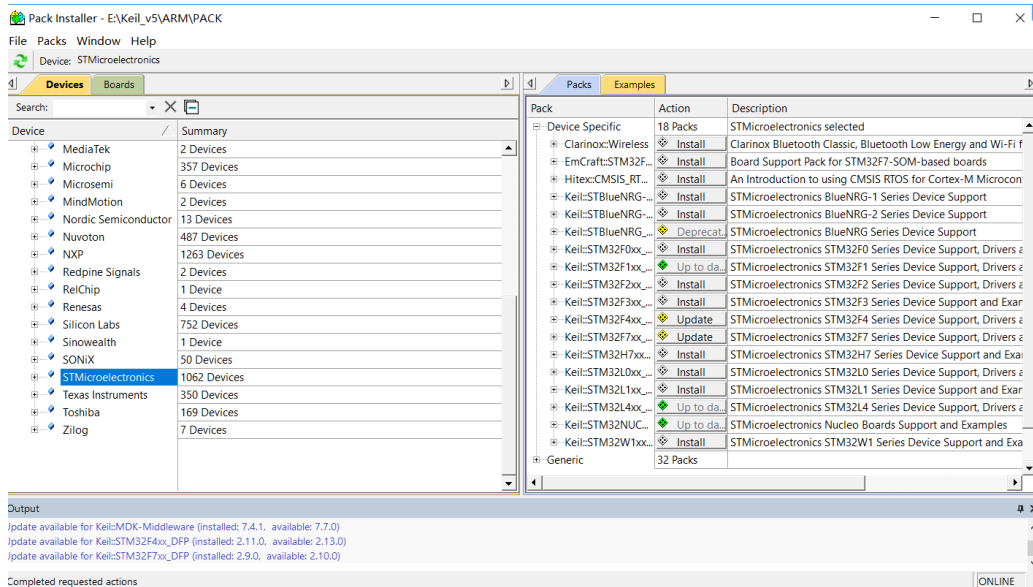


安装方式二：MDK 软件上在线安装

打开软件，在导航栏打开 Pack 安装界面，然后选择 ok 选项。



进入在线安装界面，选着 STM32L0XX Pack,点击 Install 进行安装。



至此，我们开发板的单片机程序开发环境已经搭建完毕，重启 MDK 软件就可以使用了。

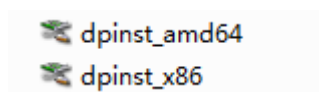
## 2.4 ST-Link 驱动安装

前面讲了开发板单片机程序的开发环境的搭建，但是为了将程序烧录到开发板中我们还需要使用仿真器。我们这套开发板选用 ST 公司的 ST-Link V2 仿真器进行开发板程序的烧写和仿真，下面介绍 ST-Link 驱动的安装及环境搭建。

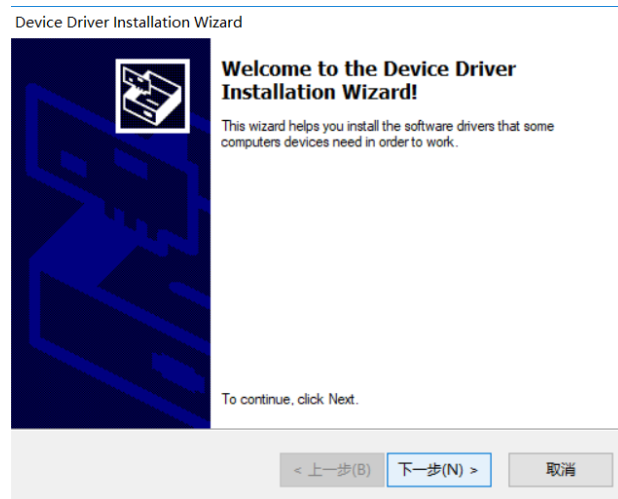
在 ST 官网下载 ST-Link 驱动，

[https://www.st.com/content/st\\_com/zh/products/development-tools/software-development-tools/stm32-software-development-tools/stm32-utilities/stsw-link009.html](https://www.st.com/content/st_com/zh/products/development-tools/software-development-tools/stm32-software-development-tools/stm32-utilities/stsw-link009.html)

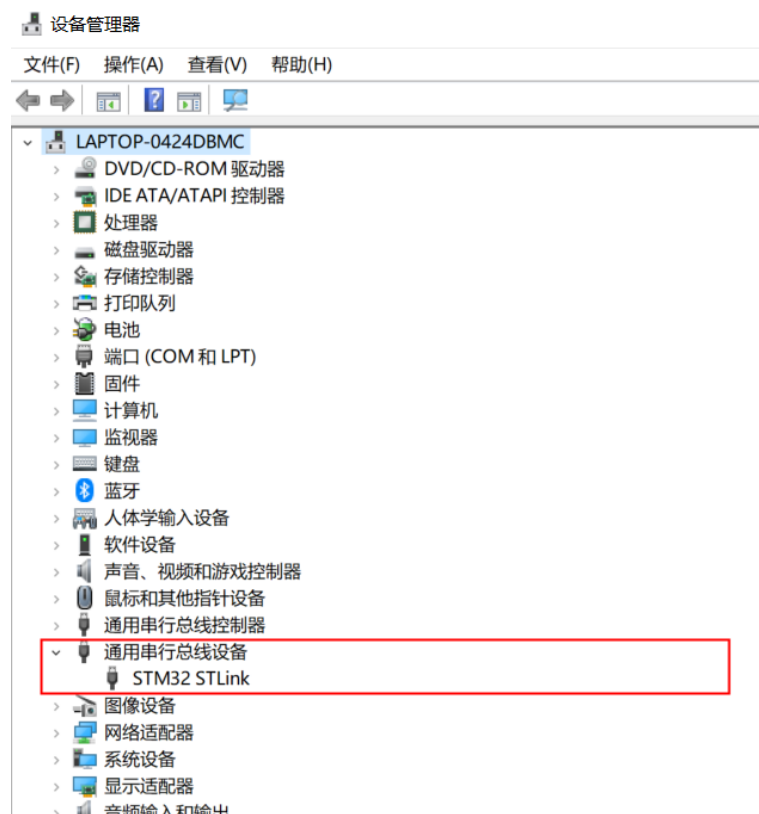
(驱动有 2 种：32 位电脑系统安装“dpinst\_x86”、64 位电脑系统安装“dpinst\_amd64”)。



运行对应的驱动，安装 ST-Link V2 驱动程序。安装路径尽量保持默认路径。



安装完成后，将 ST-Link 通过 USB 接口连入电脑。 打开“设备管理器”。若看到如下图所示，表示驱动安装成功。



这里提醒 2 点：

1，各种 windows 版本设备名称和所在设备管理器栏目可能不一样，例如 WIN10 插上 STLINK 后显示的是 STM32 STLINK。

2，如果设备名称旁边显示的是黄色的叹号，请直接点击设备名称，然后在弹出的界面点击更新设备驱动

至此， ST-Link 驱动已经安装完成。接下来大家只需要在 MDK 工程里面配置一

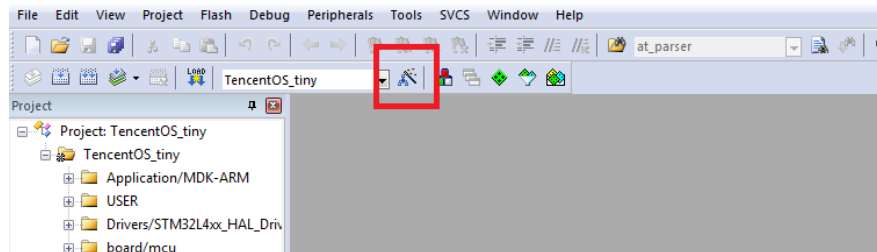


下 ST-Link 即可。

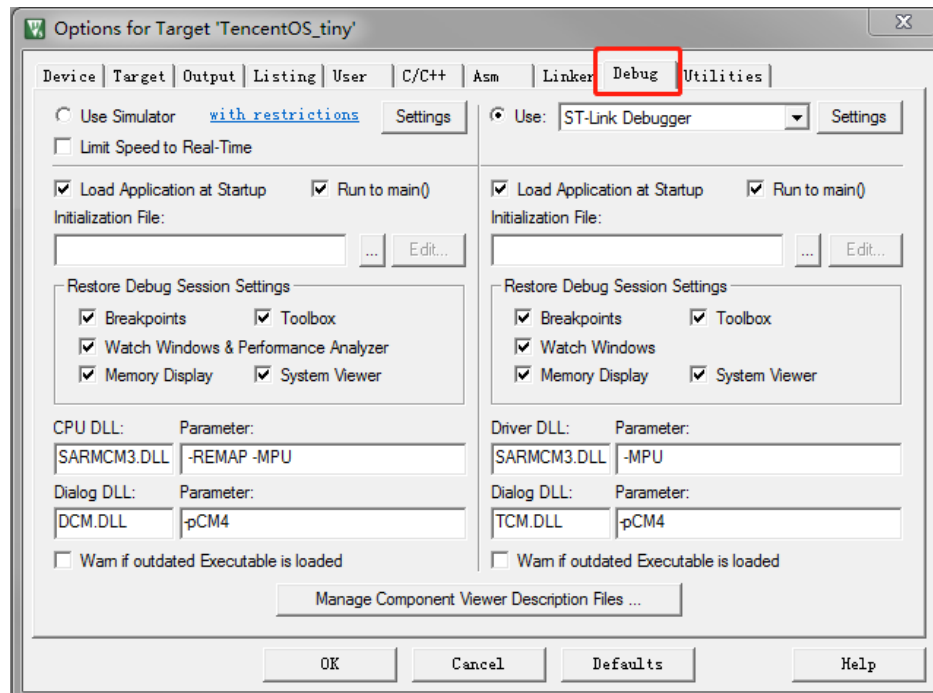
## 2.5 编程软件(MDK)配置

安装驱动成功后，打开 MDK 软件，配置程序烧写和仿真的环境。

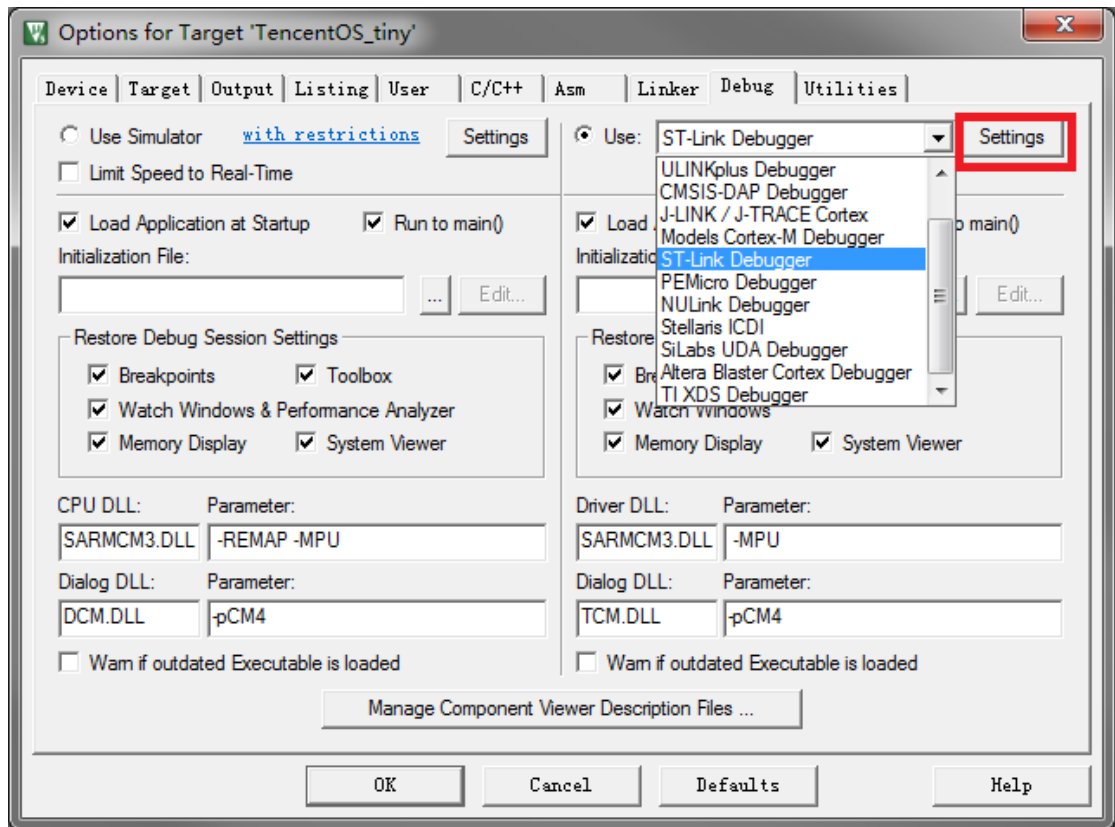
点击进入工程配置界面按钮，进入工程配置界面。



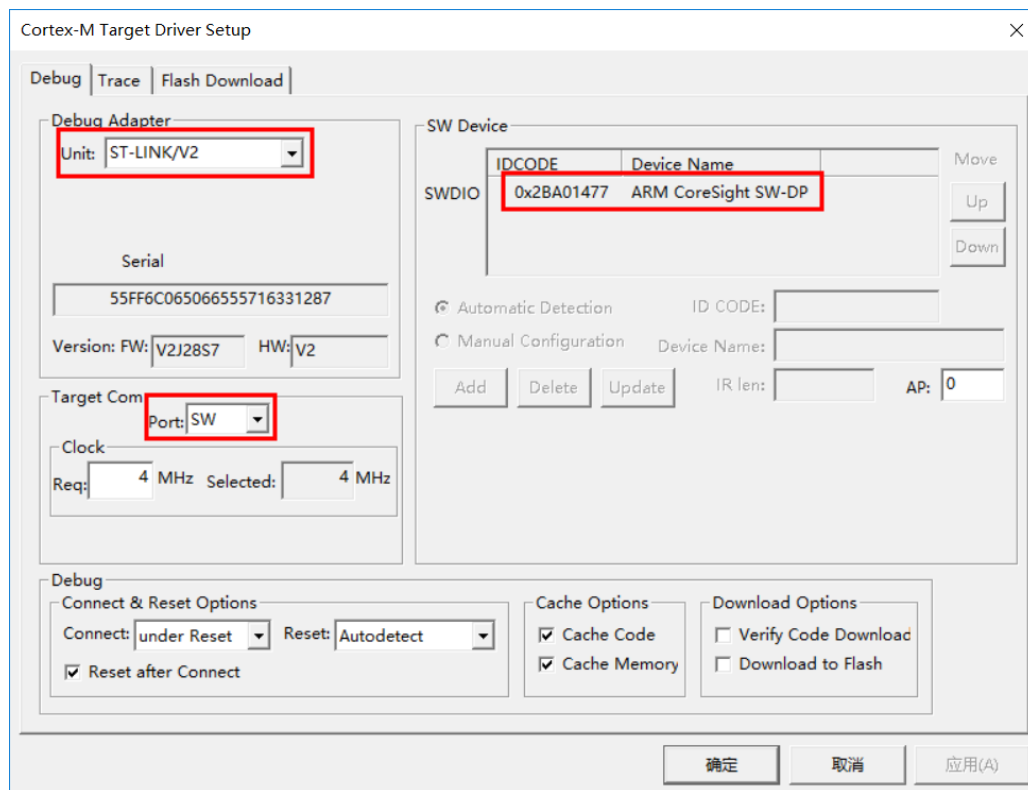
选择 Debug 选项，进入仿真器设置界面。



下拉仿真器选择列表，选着 ST-Link Debugger 并勾选右侧 Use,点击 Settings 进入 ST-Link V2 仿真器配置界面。



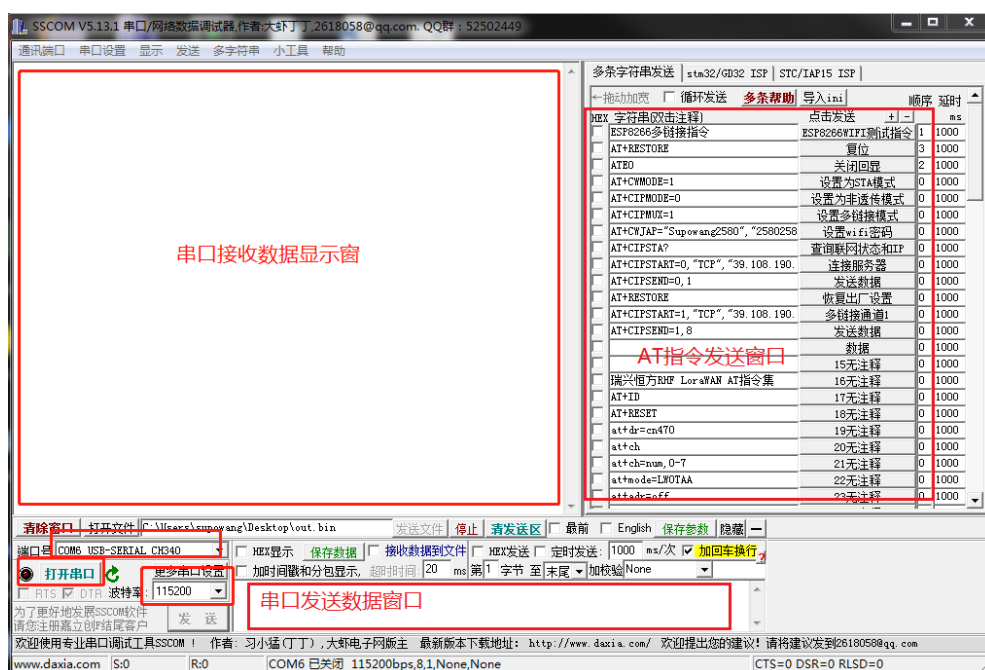
开发板设计的程序烧录方式为 SW，此处 Unit 选择 ST-Link/V2,且 Port 选择 SW,并确认右侧框内是否检测出 SW 设备，点击<确认>保存配置。



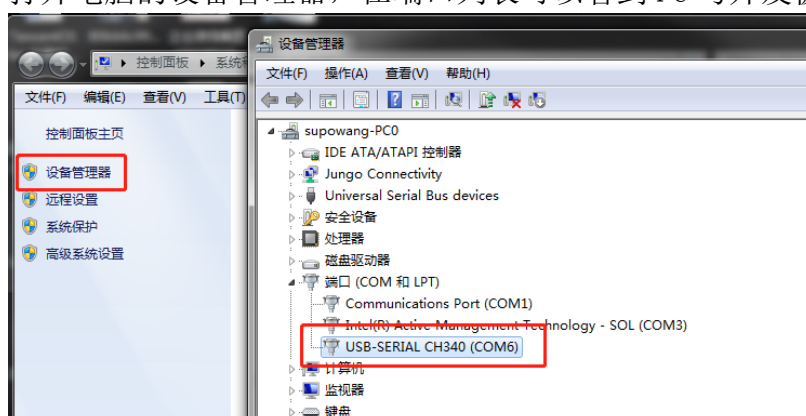
## 2.6 串口调试助手的安装与使用

工具下载: <http://www.daxia.com/download/sscom.rar>

安装方法: 串口调试助手 sscom5.13.1 是免安装的, 解压出压缩包即可直接使用。



根据 PC 和终端之间的连接，选择正确的串行端口。  
打开电脑的设备管理器，在端口列表可以看到 PC 与开发板连接的端口号。



我这里显示的是 COM6，所以要在 sscom 工具中选择 COM6，开发板程序波特率设置为 115200，所以我在 sscom 串口工具中选择 115200 波特率。

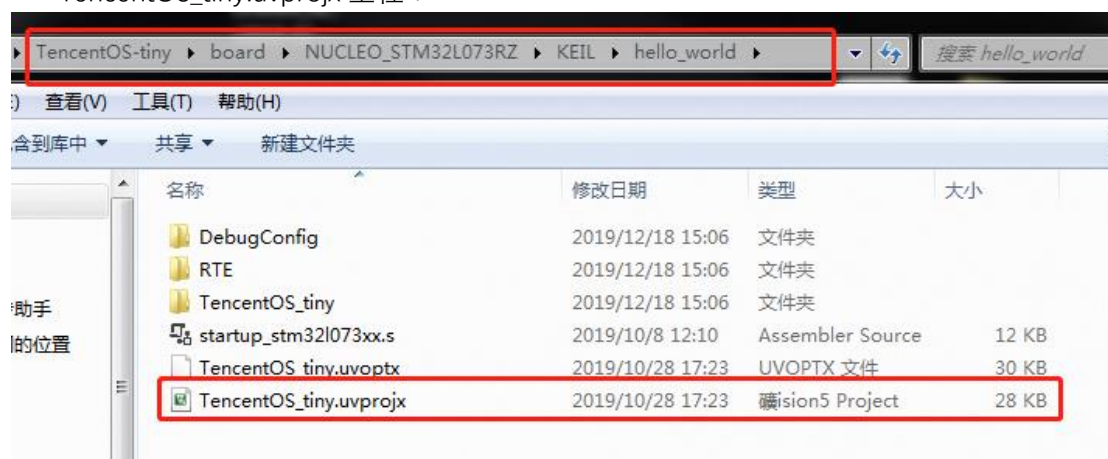
### 3. TencentOS tiny 基础内核实验

#### 3.1 打开 TencentOS tiny 提供的 Hello\_world 工程

TencentOS tiny 官方开源仓下载源码，地址为：

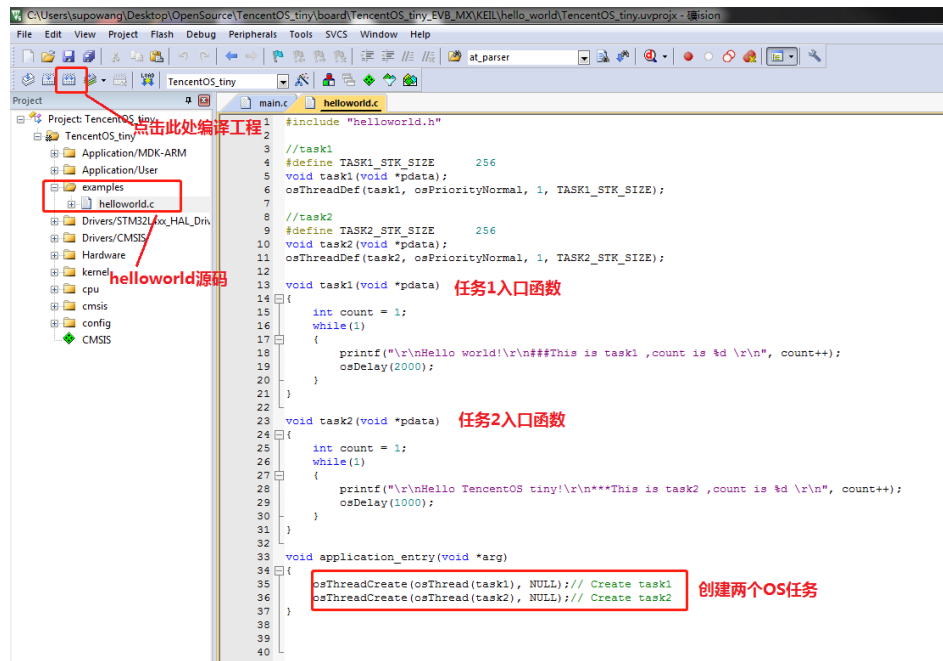
<https://github.com/Tencent/TencentOS-tiny>

进入 < TencentOS-tiny/board/NUCLEO\_STM32L073RZ/KEIL/hello\_world > 目录，打开 TencentOS\_tiny.uvprojx 工程：



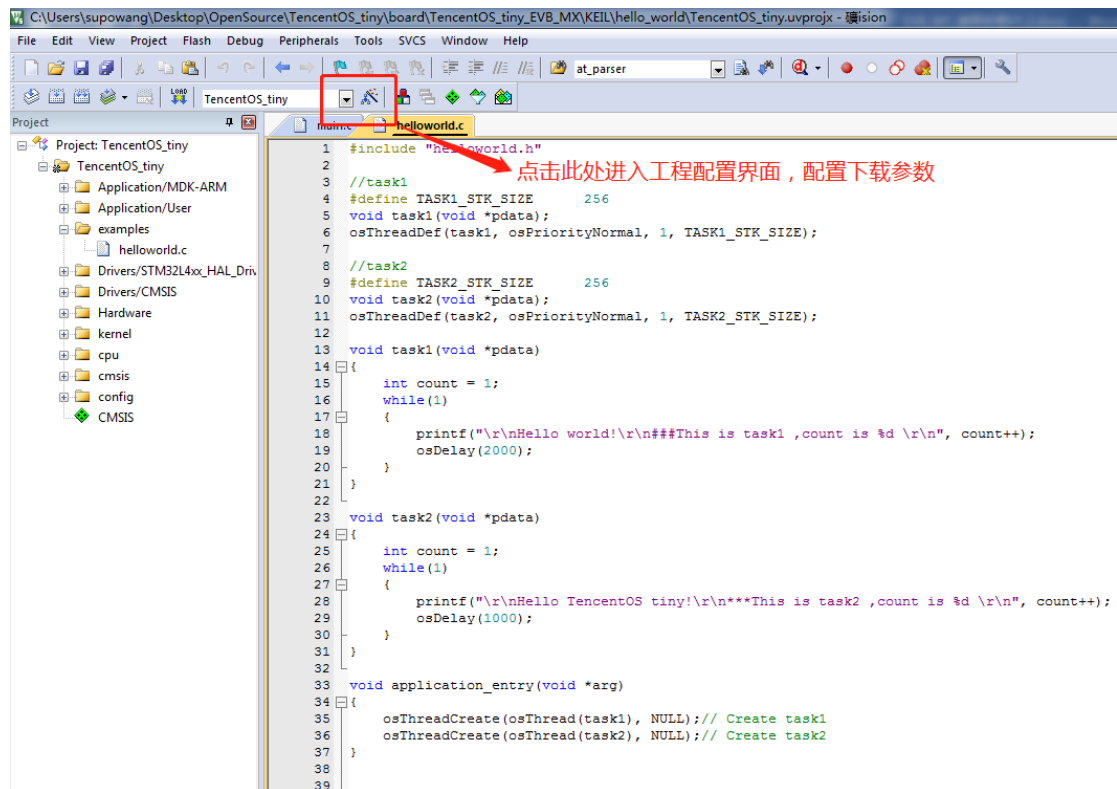
#### 3.2 编译 HelloWorld 工程

打开工程后，我们在左侧的工程文件导航页面展开 examples 目录，可以看到 helloworld.c 源码，这里创建了两个 TencentOS tiny 的任务，交替运行打印任务。开发者安装下图指示，点击编译按钮即可编译工程，如图：

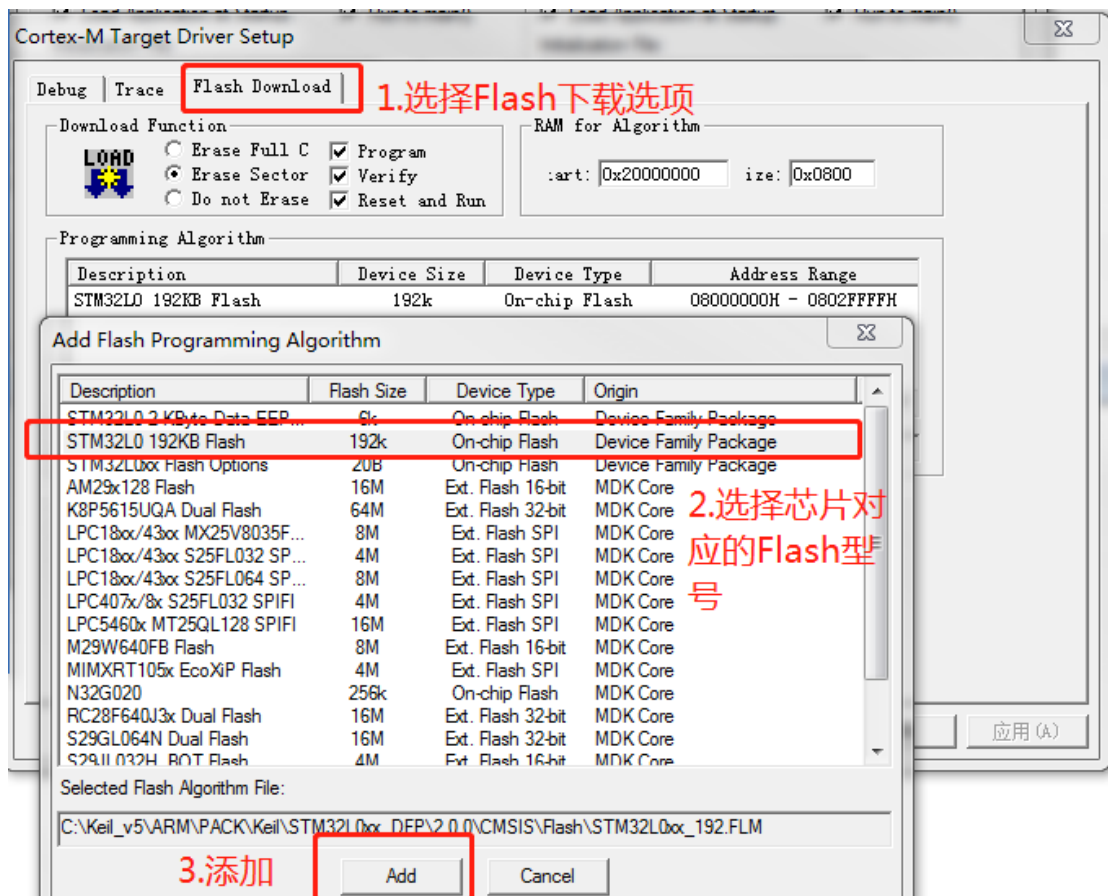
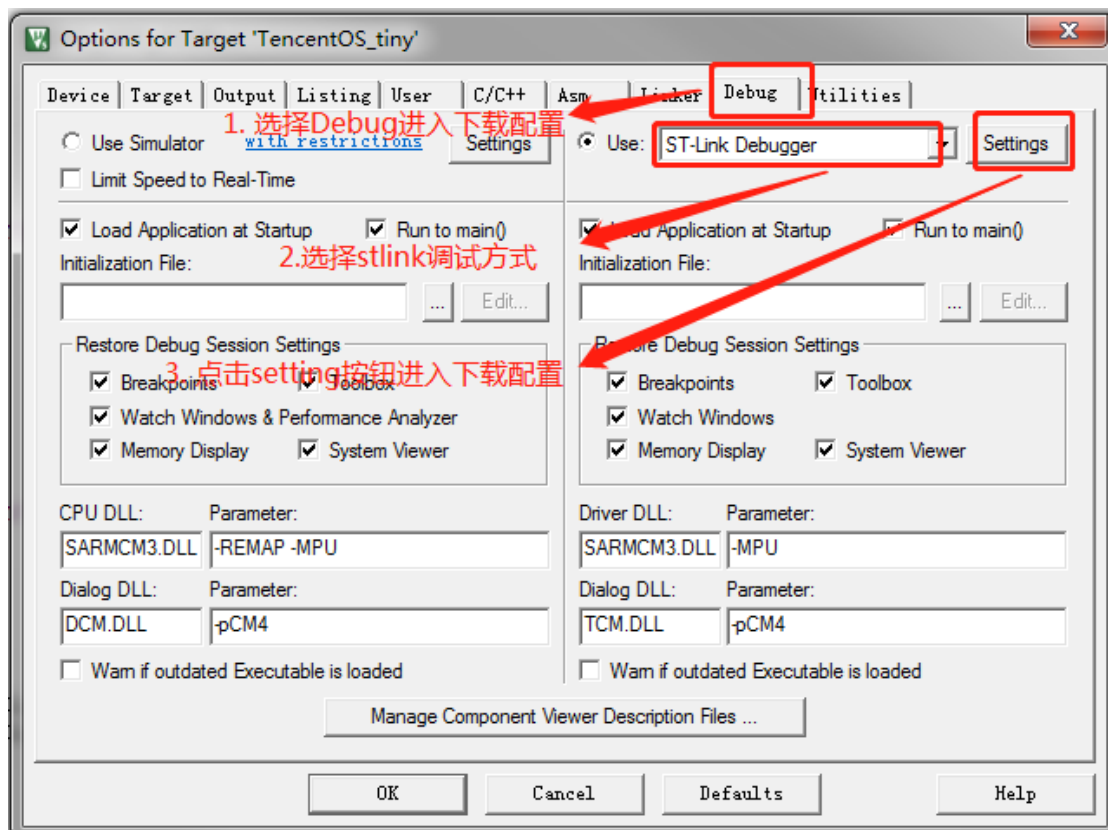


### 3.3 下载运行

首先需要配置下载环境

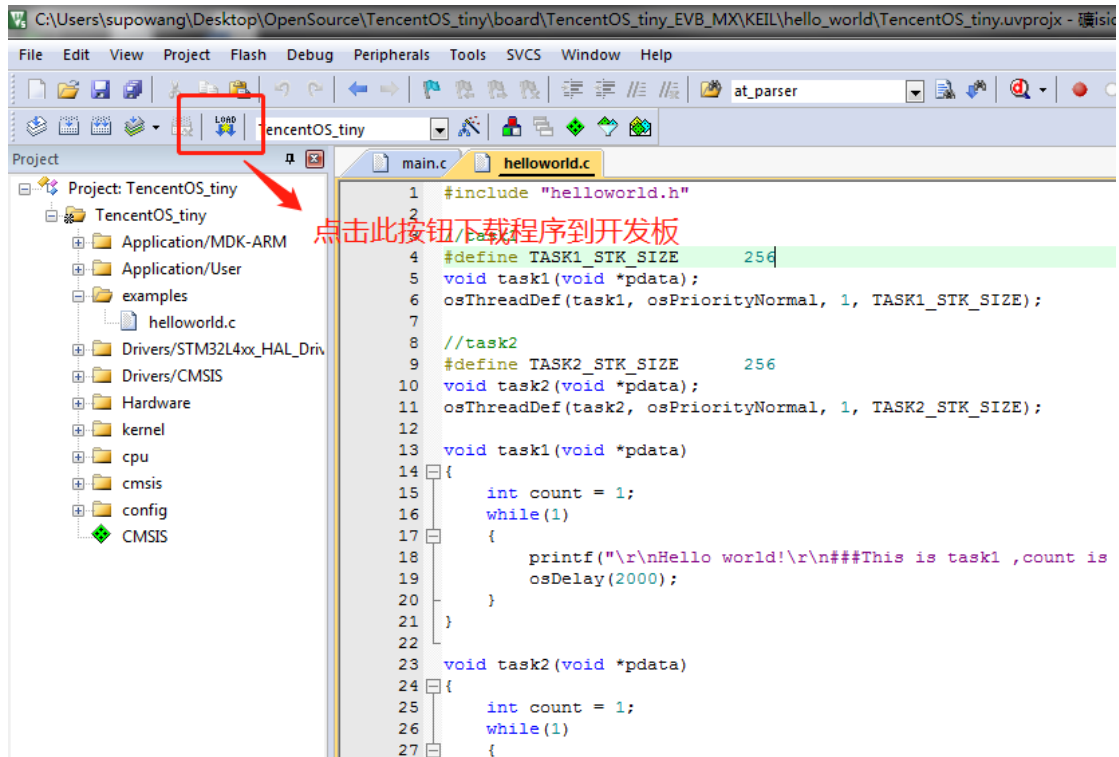


按下图所示配置下载参数



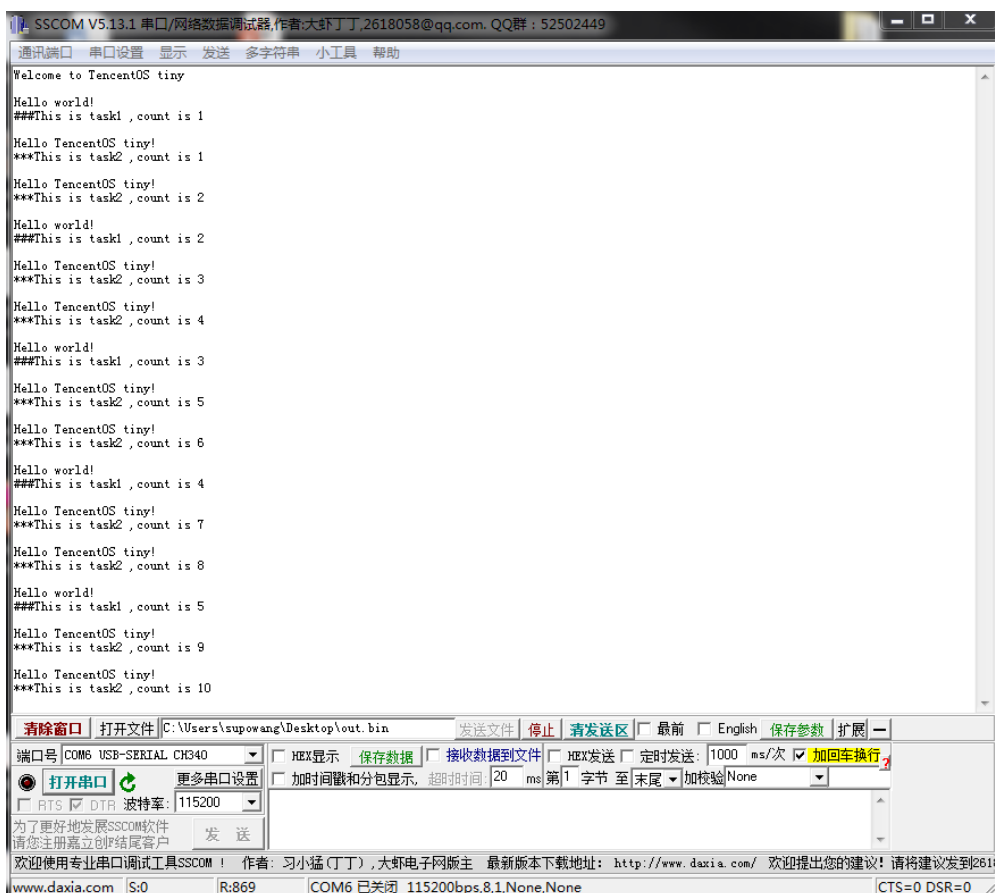
编译完成后点击如图所示“LOAD”按钮下载程序即可。





### 3.4 查看运行结果

连接好串口，在 PC 的串口助手可以看到 TencentOS tiny 的两个任务交替运行，打印消息并完成任务计数，如下图所示：



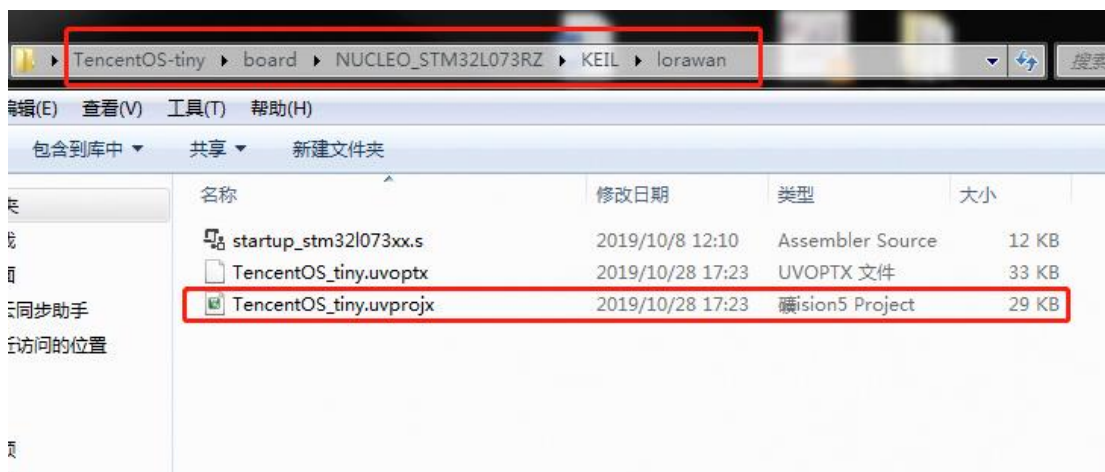
更多 TencentOS tiny 基础内核的使用，请参考内核开发指南文档：  
[TencentOS\\_tiny\doc\04.TencentOS\\_tiny\\_Development\\_Guide.md](#)  
[TencentOS\\_tiny\doc\05.TencentOS\\_tiny\\_SDK\\_documentation.md](#)

## 4. 使用 TencentOS tiny AT 框架驱动 LoRaWAN 节点 对接腾讯云 IoT Explorer

### 4.1 打开 TencentOS tiny 提供的 LoRaWAN 例程

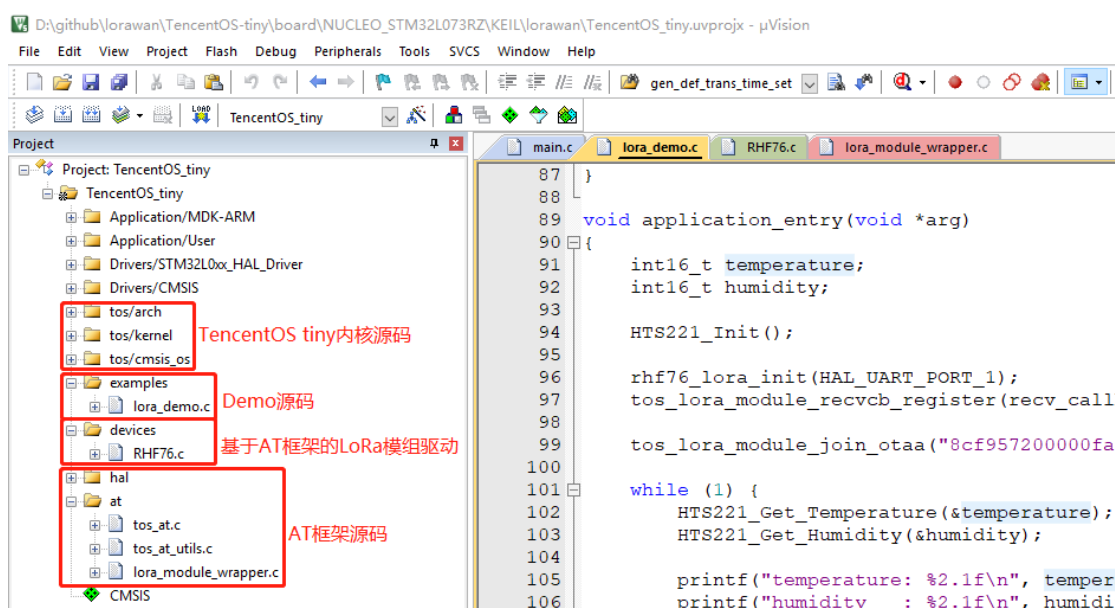
TencentOS tiny 官方开源仓下载源码，地址为：  
<https://github.com/Tencent/TencentOS-tiny>

进入 <[TencentOS-tiny\board\NUCLEO\\_STM32L073RZ\KEIL\lorawan](#)> 目录，打开  
TencentOS\_tiny.uvprojx 工程：



## 4.2 示例工程代码简介

示例工程包含 STM32L073 外设驱动、TencentOS tiny 内核、AT 框架、RHF76 LoRaWAN 模组驱动、LoRaWAN 示例案例，如下图所示：



```

lora_demo.c  RHf76.c  lora_module_wrapper.c
}

void application_entry(void *arg)
{
    int16_t temperature;
    int16_t humidity;

    HTS221_Init(); 初始化温湿度传感器

    rhf76_lora_init(HAL_UART_PORT_1); 初始化LoRa模组

    tos_lora_module_rcvcb_register(rcv_callback); 注册接收数据处理回调

    tos_lora_module_join_otaa("8cf957200000fa57", "8cf957200000fa572059aaaaad204a72");
                                                                    otaa模式接入网关

    while (1) {
        HTS221_Get_Temperature(&temperature);
        HTS221_Get_Humidity(&humidity);

        printf("temperature: %2.1f\n", temperature / 10.0);
        printf("humidity : %2.1f\n", humidity / 10.0);
                                                                    获取并组装上行数据

        dev_data_wrapper.u.dev_data.temperature = temperature / 10;
        dev_data_wrapper.u.dev_data.humidity    = humidity / 10;
        dev_data_wrapper.u.dev_data.period      = report_period;

        tos_lora_module_send(dev_data_wrapper.u.serialize, sizeof(dev_data_t));
        tos_task_delay(report_period);
                                                                    发送数据
    }
}

```

TencentOS tiny 提供的 LoRaWAN API 函数在 lora\_module\_wrapper.h 头文件中声明，用户调用 rhf76\_lora\_init(HAL\_UART\_PORT\_1);函数进行模组初始化，然后调用 tos\_lora\_module\_join\_otaa（如果是 ABP 模式，调用 tos\_lora\_module\_join\_abp 接口）入网，入网成功后即可使用 tos\_lora\_module\_send 函数向 LoRaWAN 网关和服务端发送数据了。

如果用户需要处理服务端下发的数据，调用 tos\_lora\_module\_rcvcb\_register(lora\_rcv\_callback\_t rcv\_callback)函数注册一个回调，用于处理下行数据。