

MIXCHIP-KIT 开发板用户手册

Version 1.0



版权声明

版权所有：深圳市银尔达电子有限公司。深圳市银尔达电子有限公司保留所有权利。

说明

本应用指南对应产品为 MIXCHIP-KIT 评估板。

本应用指南的使用对象是嵌入式工程师，开发工程师及测试工程师。

深圳市银尔达电子有限公司专注于物联网解决方案，并且为客户提供全方位的技术支持，任何垂询请直接联系您的客户经理或发送邮件至一下邮箱：

Eric@yinerda.com

jason@yinerda.com

vito@yinerda.com

公司网站：<http://www.yinerda.com>

联系电话：0755-23732189

联系地址：深圳市龙华区大浪街道工业园路浦华科技园 5 栋



修改记录

版本号	修改记录	发布时间
V1.0		2018-11-9



目录

一、获取资料方法.....	7
二、底板版本区分.....	8
三、 底板硬件介绍.....	9
3.1、硬件功能介绍.....	9
3.2、硬件注意事项.....	11
四、 AT 固件测试.....	12
4.1、确定固件版本.....	12
4.2、安装 CP2102 驱动.....	12
4.3、安装串口调试助手.....	12
4.4、把 UART JMP 跳线帽如下.....	12
4.5、AT1.0 命令测试.....	13
4.6、AT2.0 测试.....	15
五、固件升级.....	16
5.1、升级文档教程.....	16
5.2、升级注意事项.....	16
六、二次开发环境搭建.....	17
6.1、安装 python.....	18
6.2、安装 Git.....	19
6.3、安装 MiCoder Tools.....	19
6.4、安装 MiCO-Cube.....	19
6.5、设置 MiCoder Tools 路径.....	19
6.6、安装 MiCoder IDE.....	20
6.7、安装 CP2102 驱动.....	20
6.8、安装 jlink 驱动.....	20
6.9、jlink 驱动转换.....	20
七、二次开发.....	22
7.1、下载 mico-demo 工程.....	22
7.2、添加银尔达开发板测试工程.....	23
7.3、MiCoder IDE 创建工程.....	23
7.4、修改 yinerda_mixchip_kit 代码.....	26
7.5、添加编译 target.....	26
7.6、编译下载固件.....	28
7.7、固件测试.....	30
八、二次开发注意事项.....	31
8.1、EMW3060 开发注意事项.....	31
8.2、EMW3165 开发注意事项.....	31
九、原理图分析.....	32



银尔达电子 专注于物联网解决方案

公司介绍

深圳市银尔达电子有限公司是全球领先的WIFI/蓝牙/ZIGBEE等物联网模组供应商，并且提供整套智能家居智能语音、智慧照明、智慧农业、智慧工业、智能锁解决方案。

核心价值观

公司以“专注物联，用心服务”为核心价值，希望通过我们的专业水平和不屑努力，让智能硬件开发变得更简单，更易用，更智能。

联系我们

公司名字：深圳市银尔达电子有限公司

公司电话：0755-23732189

淘宝购买地址：<https://yinerda.taobao.com>

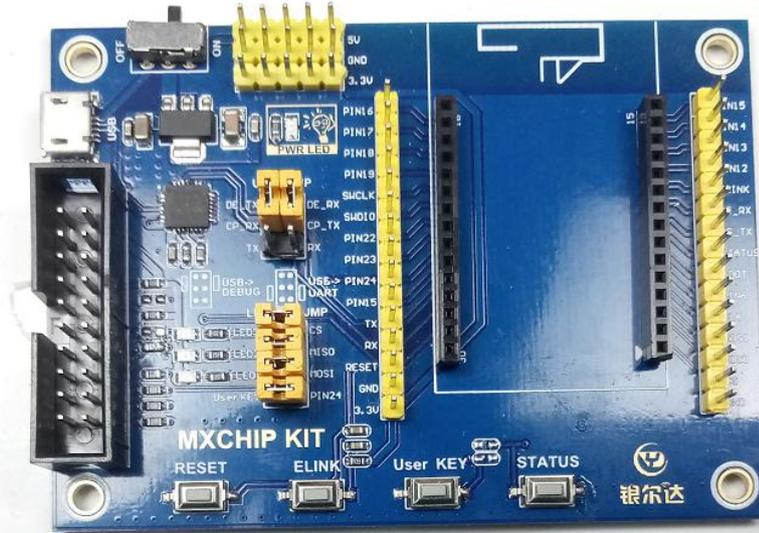
公司官网：<http://www.yinerda.com>

联系电话：18923435189

微信：18923435189



MIXCHIP 评估板是由银尔达 (yinerda) 推出针对庆科模组的评估开发板，板载 USB 转 TTL 串口、LED、按键、JLINK 下载口，可以方便的与我们推出的 WIFI 转接板配使用，引出了 WIFI 模组的全部 IO 口，方便客户对模组 AT 命令的测试和二次开发的下载调试。支持的模组有 EMW3080、EMW3031、EMW3060、EMW3239、EMW3166、EMW3165 等模组。



一、获取资料方法

- 1、每个产品在对应淘宝详情页->资料下载 下载对应产品的资料
- 2、AT 命令手册和二次开发资料链接:<http://developer.mxchip.com>

[首页](#) [视频中心](#) [资料中心](#)

物联网工程师开发服务平台

[文档中心](#) [下载中心](#) [AT指令v1.0](#) [AT指令v2.0](#) [LoraWAN AT指令](#)

WiFi 或 WiFi/BT 模组

1. 固件概述
2. 工作模式
3. 出厂设置
4. 管脚分配

AT v1.0 透传固件 – 固件概述

AT 透传固件 是由 MXCHIP 开发的，运行于 EMW 系列Wi-Fi 模块或 Wi-Fi/BT 组合快速地为嵌入式设备增加Wi-Fi/BT通信功能。大大缩短开发周期，实现快速上市。

目录

- 3、AT 固件下载链:<http://developer.mxchip.com>

[首页](#) [视频中心](#) [资料中心](#) 注册

物联网工程师开发服务平台

硬件设备

1. 模块产品
2. 开发板
 - 2.1 EXT-AT3080 开发板
 - 2.2 MXKit 开发套件
 - 2.3 Microchip 合作开发板
3. AT固件
 - 3.0 固件下载
 - 3.1 固件概述
 - 3.2 工作模式
 - 3.3 出厂设置
 - 3.4 管脚分配
 - 3.5 指令说明
 - 3.6 使用示例
 - 3.7 直连ilop云平台指

AT v2.x 固件

AT 固件是由 MXCHIP 开发的，运行于 Wi-Fi 或 Wi-Fi/BT 无线模块上的软件指令系统。通过该指令，用户可以快速地为嵌入式设备增加无线通信功能，大大缩短开发周期，实现快速上市。

点击下载: [AT 固件使用手册.pdf](#)。

型号	版本	支持直连的云	固件下载	烧录ota.bin	烧录 all.bin
MOC108 / EMW3060	v2.0.9	通用版AT	MOC108_0000.0000.A209.zip	参见压缩包中“烧录方法”	请联系代理
MOC108A / EMW110A	v2.0.7	通用版AT	MOC108A_0000.0000.A207.zip	参见压缩包中“烧录方法”	请联系代理
MX1290 (适用于 EMW3080B和5080B)	v2.1.1	通用版AT	MX1290_0000.0000.A211.zip	更新ota.bin	更新 all.bin
	v2.1.4	阿里飞燕 ilop平台	MX1290_0000.ILOP.A214.zip	同上	同上



二、底板版本区分

底板分为两个版本，底板支持的模组有 EMW3080、EMW3031、EMW3060、EMW3239、EMW3166、EMW3165 等模组，但是 EMW3060 是 JLINK 下载接口，其他的模组是 SWD 下载接口。不同的版本支持的模组不一样，需要注意。

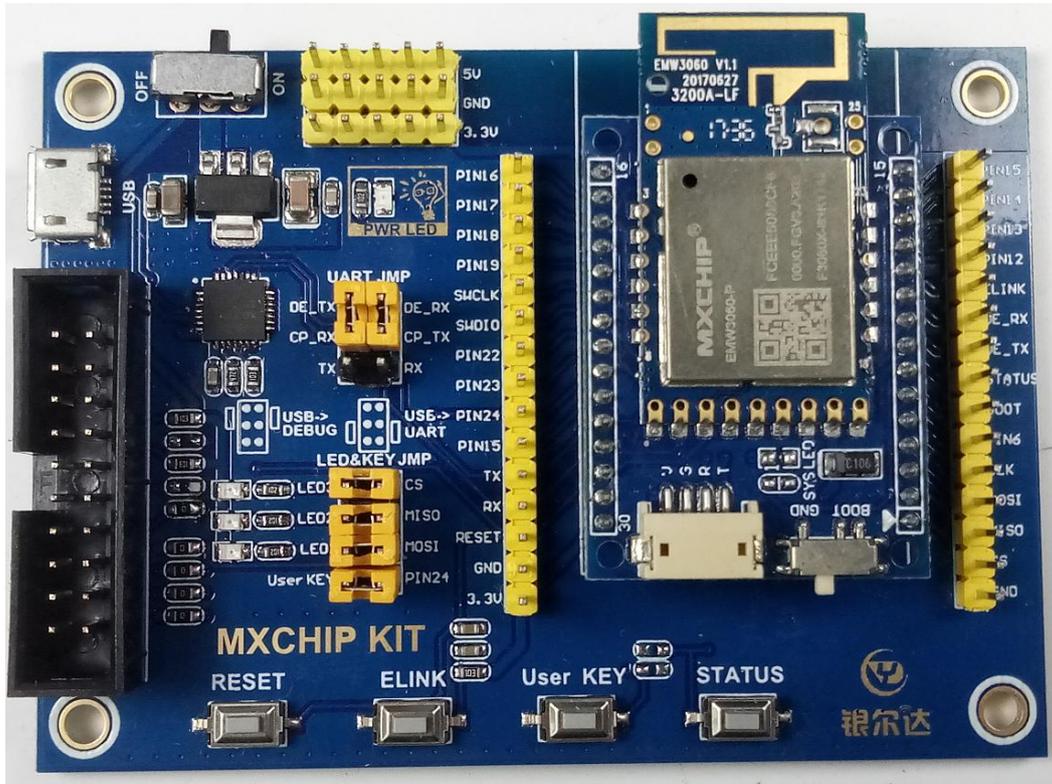


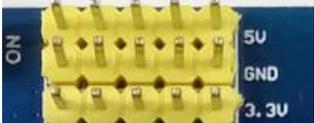
左边的图电阻焊接方式表示支持 EMW3080、EMW3031、EMW3239、EMW3166、EMW3165；

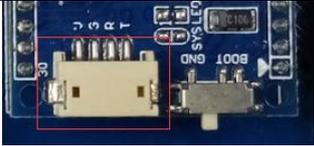
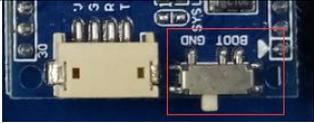
右边的图电阻焊接方式表示支持 EMW3060；

三、底板硬件介绍

3.1、硬件功能介绍



编号	位置	作用	备注
1		电源开关	丝印写反了，真实的 OFF 才是开
2		扩展电源	
3		USB 接口	用于供电和 USB 转 TTL 串口，可以用于调试和发送 AT 命令； 可以用户串口方式的固件下载；
4		USB 接口功能选择	当跳线帽跳到上面，表示 USB 与 WIFI 模组的调试串口相连；跳到下面表示 USB 与 WIFI 模组的通讯串口相连

5		LED 和按键功能选择	前面上面 3 个跳线帽用于选择 3 个 LED 是否与 WIFI 模组管脚相连； 注意 EMW3165 由于管脚已经被 Flash 使用，所有不能接 LED 管脚的跳线帽； 第 4 个跳线帽用于选择 User Key 是否与 WIFI 模组管脚相连；
6		按键	Reset 用户复位 WIFI 模组 ELINK 用于默认配网管脚 User key 是用户管脚，是否有效由跳线帽决定 Status 用于控制进入下载模式管脚
7		WIFI 串口	WIFI 模组的通讯串口，3.3V 供电
8		WIFI BOOT 管脚	与 Status 管脚配合使用，进入下载模式和产测模式
9		扩展接口	引出了 WIFI 模组的所有 GPIO
10		Jlink 接口	使用 jlink 或者 swd 接口给 WIFI 模组下载程序

3.2、硬件注意事项

3.2.1、电源开关

丝印印反了，OFF 是开，ON 是关。

3.2.2、EMW3156 的 LED 灯

在底板配套 EMW3165 模组的时候，板载的 LED 管脚是 EMW3165 的 flash 管脚，用户不可用，所以板载的跳线帽需要去掉，使用杜邦线把 LED 和其他可以使用的 GPIO 相连。

3.2.3、EMW3060 JLINK 下载程序

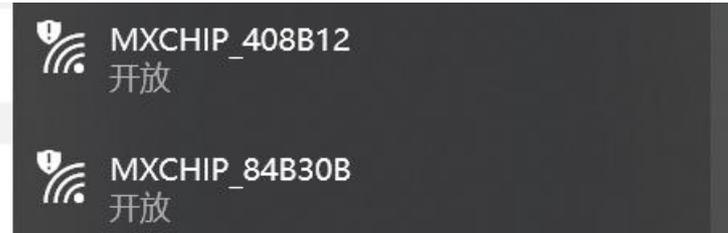
在底板配套 EMW3060 的时候，使用 JLINK 下载程序的时候，需要把模组的 BOOT 开关拨到 GND，然后才能下载，下载完成后，把开关拨到 BOOT，按下复位，模组就运行程序。



四、AT 固件测试

4.1、确定固件版本

默认情况下，出厂的 EMW3080、EMW3031、EMW3239、EMW3166、EMW3165 模组都是 AT1.X 固件，EMW3060 模组默认是 AT2.0 固件。AT1.x 固件和 AT2.x 固件不兼容，AT1.x 可以通过升级成为 AT2.x 版本。AT1.x 的固件默认上电会生成热点，名称：MXCHIP_XXXXXX（XXXXXX=MAC 地址最后六位），没有密码；而 AT2.0 固件不会生成。所以当不知道自己的模组是什么固件的时候，可以用这种方法确定。

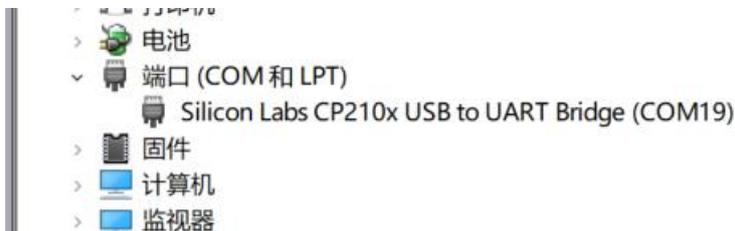


不同版本的命令手册需要参考对应的资料：<http://developer.mxchip.com/>



4.2、安装 CP2102 驱动

安装串口驱动目录下的 CP210x_VCP_Windows，插上 USB 后会在电脑设备管理器出现 COM 口表示成功



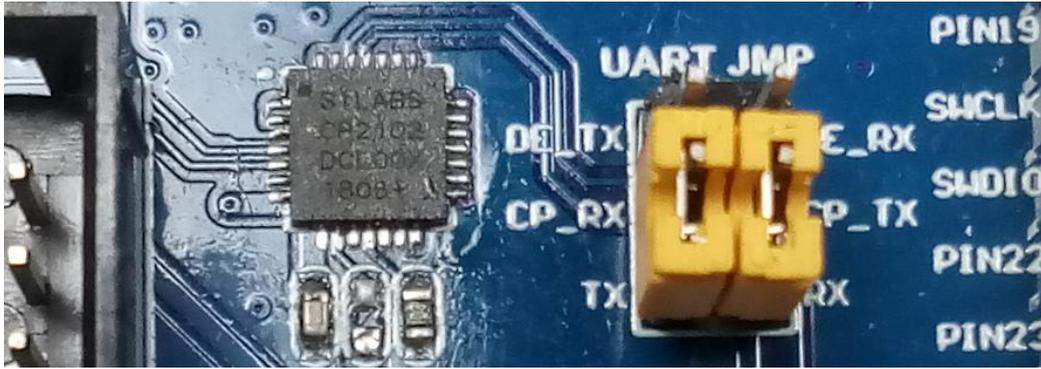
4.3、安装串口调试助手

安装串口调试助手 格西烽火 GBeacon-2.3-Fx.exe，安装完成后，才能打开.bsp 工程

4.4、把 UART JMP 跳线帽如下

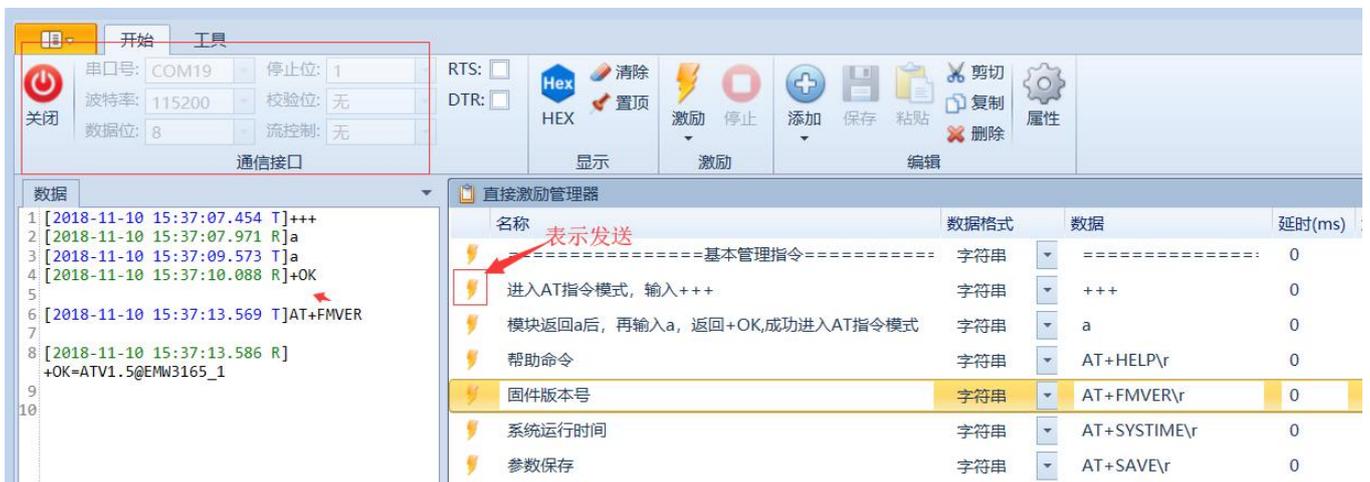
表示 USB 串口与 WIFI 模组的通讯串口(发送 AT 命令)相连





4.5、AT1.0 命令测试

打开工程 MiCO_AT_v1.0_CMD.bsp;



选择 COM 口, 串口波特率设置 115200, 其他默认;先输入+++ , 然后模组回复 a, 然后发送 a, 模组回复+OK 表示模组进入了 AT 模式, 然后就可以发送其他命令, 比如配置路由器, 服务器等。

模式切换时序如下:

WiFi 或 WiFi/BT 模组

1. 固件概述
2. 工作模式
3. 出厂设置
4. 管脚分配
5. 网页配置
6. BT 快速开始
7. AT 指令概述
8. AT指令详述
9. AT固件及指令使用

BLE 模组

1. AT 透传固件
2. AT 指令集

透明传输模式通过输入 “+++” 和 “a” 切换到命令模式, 规则:

- 1) 在串口输入 “+++”, 模块收到 “+++” 后, 返回确认码 “a” ;
- 2) 在串口输入确认码 “a”, 模块收到确认码后, 返回 “+OK”, 进入命令控制模式

注意: 输入 “+++” 和 “a” 需要在一定时间内完成, 以减少误入命令的概率, 规则:

输入“+”	输入“+”	输入“+”	输入“a”
<100ms	<100ms	<3s	

具体进入命令模式的时间消耗:

前一帧	>100ms延时 UARTFT时间间隔	“+++”发送时间	>100ms延时 UARTFT时间间隔	“a”返回时间	“a”发送时间 <3s	>100ms延时 UARTFT时间间隔	“+OK” 返回时间	AT命令
-----	------------------------	-----------	------------------------	---------	----------------	------------------------	---------------	------

AT1. x 其他测试例程:



物联网工程师开发服务平台

[文档中心](#)[下载中心](#)[AT指令v1.0](#)[AT指令v2.0](#)[LoraWAN AT指令](#)

WiFi 或 WiFi/BT 模组

1. 固件概述
2. 工作模式
3. 出厂设置
4. 管脚分配
5. 网页配置
6. BT 快速开始
7. AT 指令概述
8. AT指令详述
9. AT固件及指令使用

AT v1.x 透传固件及指令使用示例

本文将详细介绍：如何更新 AT 应用程序固件，如何实现工作模式切换，及如何开启模块 WiFi 通信等功能。

这里提供一个格西烽火串口指令工程下载链接，[AT_v1.0_CMD.bsp](#)，用户可快速进行 AT 指令的

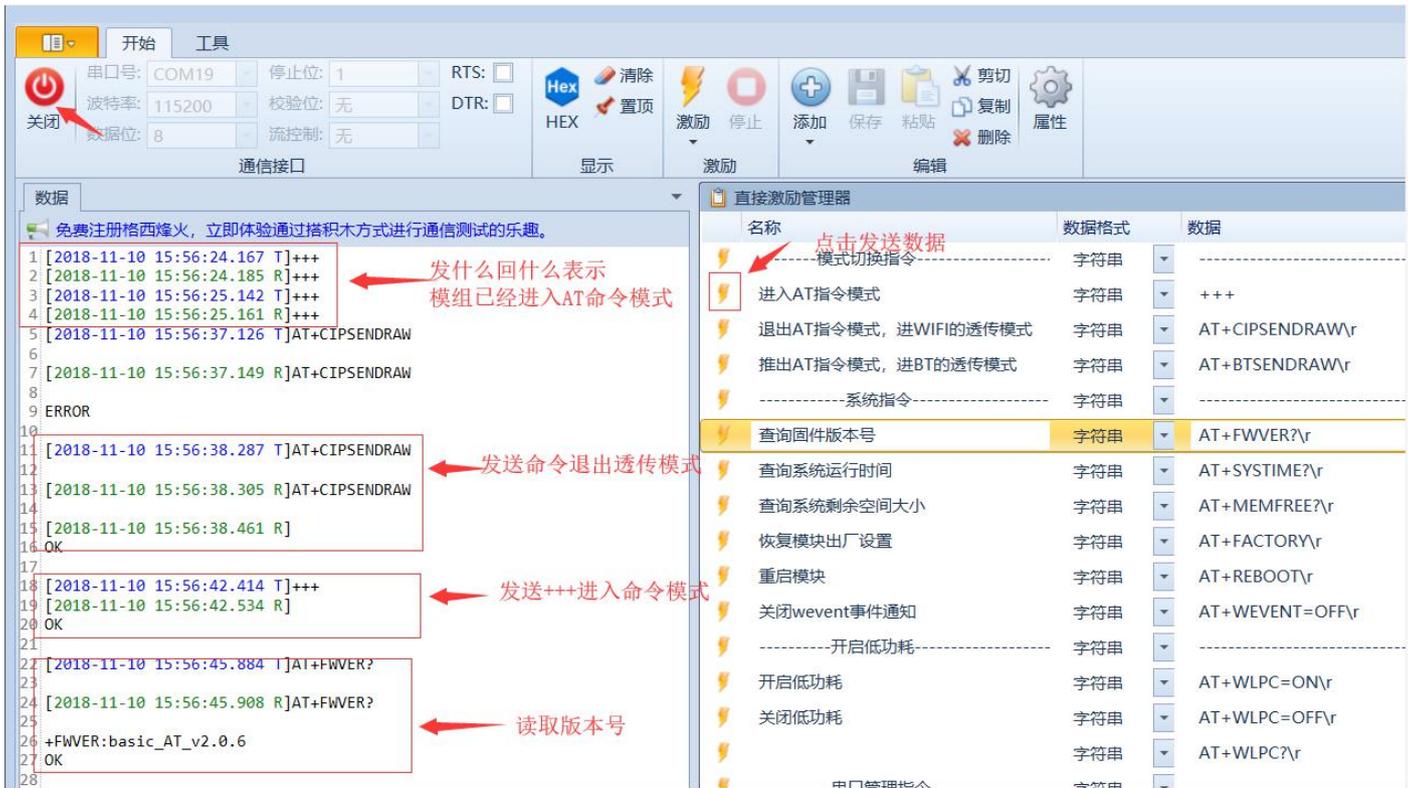
目录

- AT透传固件更新
 - 通过Bootloader模式更新固件
 - 通过网页功能更新固件



4.6、AT2.0 测试

打开工程 MiCO_AT_v2.0_CMD.bsp;



The screenshot shows a serial terminal window with the following content:

```

1 [2018-11-10 15:56:24.167 T]+++
2 [2018-11-10 15:56:24.185 R]+++
3 [2018-11-10 15:56:25.142 T]+++
4 [2018-11-10 15:56:25.161 R]+++
5 [2018-11-10 15:56:37.126 T]AT+CIPSENDRAW
6
7 [2018-11-10 15:56:37.149 R]AT+CIPSENDRAW
8
9 ERROR
10
11 [2018-11-10 15:56:38.287 T]AT+CIPSENDRAW
12
13 [2018-11-10 15:56:38.305 R]AT+CIPSENDRAW
14
15 [2018-11-10 15:56:38.461 R]
16 OK
17
18 [2018-11-10 15:56:42.414 T]+++
19 [2018-11-10 15:56:42.534 R]
20 OK
21
22 [2018-11-10 15:56:45.884 T]AT+FWVER?
23
24 [2018-11-10 15:56:45.908 R]AT+FWVER?
25
26 +FWVER:basic_AT_v2.0.6
27 OK
28
    
```

Annotations in red text point to specific lines:

- Line 1-4: 发什么回什么表示模组已经进入AT命令模式
- Line 11-13: 发送命令退出透传模式
- Line 18-19: 发送+++进入命令模式
- Line 22-24: 读取版本号

The right side of the image shows the 'Direct Command Manager' (直接激励管理器) with a table of commands:

名称	数据格式	数据
模式切换指令	字符串	-----
进入AT指令模式	字符串	+++
退出AT指令模式, 进WiFi的透传模式	字符串	AT+CIPSENDRAW\r
推出AT指令模式, 进BT的透传模式	字符串	AT+BTSSENDRAW\r
-----系统指令	字符串	-----
查询固件版本号	字符串	AT+FWVER?\r
查询系统运行时间	字符串	AT+SYSTIME?\r
查询系统剩余空间大小	字符串	AT+MEMFREE?\r
恢复模块出厂设置	字符串	AT+FACTORY\r
重启模块	字符串	AT+REBOOT\r
关闭wevent事件通知	字符串	AT+WEVENT=OFF\r
-----开启低功耗	字符串	-----
开启低功耗	字符串	AT+WLPC=ON\r
关闭低功耗	字符串	AT+WLPC=OFF\r
-----	字符串	AT+WLPC?\r

AT2. x 测试历程:



The screenshot shows the 'IoT Engineer Development Service Platform' (物联网工程师开发服务平台) website. The navigation bar includes '文档中心', '下载中心', 'AT指令v1.0', 'AT指令v2.0', and 'LoraWAN AT指令'. The 'AT指令v2.0' link is highlighted with a red arrow.

The main content area is titled 'AT v2.0 指令 – 使用用例' (AT v2.0 Commands – Usage Examples). It contains the following text:

本文将详细介绍：如何更新 AT 应用程序固件；如何实现工作模式切换；如何开启模块 WiFi 功能的不同工作模式；以及如何建立 Socket 连接并通信等功能。

- 这里提供一个 榕西烽火 串口调试工程：AT v2.0_CMD.bsp, 用户可快速进行 AT 指令的开发与调试。
- 同时提供通过 MQTT 协议连接 FOG 云 和 AWS 亚马逊云 相关 AT 指令：at cmd fog_mqtt he at cmd aws_mqtt.
- AT 指令 v2.0 使用用例，视频教程观看地址：<http://t.elecfans.com/3770.html>

A sidebar on the left lists various topics, with '6. 通用指令使用示例' (General Command Usage Examples) highlighted by a red arrow.

At the bottom of the page, there is a '目录' (Table of Contents) section.

五、固件升级

AT1.x 模组可以升级到 AT2.x，AT2.x 的模组也可以降级到 AT1.x。升级固件只有在自己评估不同的固件才会使用到，如果是批量的时候，只需要确定固件后，我们会发对应固件。

每个种模组的升级方式和方法不一样，详情可以看官网文档。当使用串口升级出错(烧到最后报错，重启不生效)的时候，必须使用 jlink 烧写才能恢复。

5.1、升级文档教程



物联网工程师开发服务平台

AT v2.x 固件

AT 固件是由 MXCHIP 开发的，运行于 Wi-Fi 或 Wi-Fi/BT 无线模块上的软件指令系统。通过该指令，用户可以快速地为嵌入式设备增加无线通信功能，大大缩短开发周期，实现快速上市。

点击下载：[AT 固件使用手册.pdf](#)。

型号	版本	支持直连的云	固件下载	烧录ota.bin	烧录all.bin
MOC108 / EMW3060	v2.0.9	通用版AT	MOC108_0000.0000.A209.zip	参见压缩包中“烧录方法”	请联系代理
MOC108A / EMW110A	v2.0.7	通用版AT	MOC108A_0000.0000.A207.zip	参见压缩包中“烧录方法”	请联系代理
MX1290 (适用于 EMW3080B和5080B)	v2.1.1	通用版AT	MX1290_0000.0000.A211.zip	更新ota.bin	更新all.bin
	v2.1.4	阿里飞燕 ilop平台	MX1290_0000.ILOP.A214.zip	同上	同上

5.2、升级注意事项

- 1、从 AT1.0 固件升级到 AT2.0 必须烧写 all.bin 文件。
- 2、EMW3060 升级串口是调试串口，波特率 921600；其他模组都是通讯串口升级



六、二次开发环境搭建

模组都可以使用 MICO 做二次开发，以实现 SOC 方案开发。

推荐的 JLINK 使用 V9 版本。

参考资料

首页 视频中心 资料中心

物联网工程师开发服务平台

文档中心 下载中心 AT指令v1.0 AT指令v2.0 LoraWAN AT指令

关于MICO

- 1.手册导读
- 2.MiCO简介
- 3.MiCO支持的开发板
- 4.项目和组件清单

开发工具

- 1.简介
- 2.MiCO Cube编译工具
- 3.MiCoder IDE环境
- 4.JLink下载工具
- 5.串口调试终端
- 6.第三方驱动
- 7.软件小工具
- 8.第一个MiCO应用程序

第一个MiCO应用程序

本文介绍如何将第一个 MiCO 应用程序跑起来。即：如何借助 mico-cube 工具 和 MiCoder IDE 序：Helloworld 工程获取，编译，下载，运行与调试。

- 硬件准备
 - 硬件清单
 - 硬件连接
- 软件准备
 - 安装MiCO Cube
 - 安装MiCoder IDE
 - 安装串口驱动及串口调试软件
 - 更新jlink驱动
- 开始使用
 - 导入helloworld项目
 - 更新mico os至最新版本
 - MiCO Cube命令行编译与下载

在资料包里面有下载好的工具，可以自己安装

板模组资料 > MIXCHIP-KIT开发板资料 > Software > 庆科二次开发环境 >

名称	修改日期	类型	大小
JLink Driver Update.zip	2018/5/16 18:43	360压缩 ZIP 文件	10,082 KB
jlink_driver_for_mico.zip	2018/5/17 18:46	360压缩 ZIP 文件	10,094 KB
MiCoder_IDE_1_2_1_Win32_x64.exe	2018/5/16 18:48	应用程序	405,156 KB
MiCoder_v1.1.Win32.zip	2018/5/16 18:43	360压缩 ZIP 文件	149,662 KB
python-2.7.13.amd64.msi	2018/5/17 10:36	Windows Install...	19,612 KB
python-2.7.13.msi	2018/5/16 20:47	Windows Install...	18,712 KB
Setup_JLink_V600i.zip	2018/5/17 18:46	360压缩 ZIP 文件	21,816 KB



6.1、安装 python

注意：一定要安装 2.7.13 版本

Python 的可执行文件目录 (Python) 需要添加到系统的 PATH 环境变量中，在 Python 安装时指定，Windows 系统中如下图。



确认安装好 Python 后，找到 Python 的安装目录，接着找到 pip.exe，一般而言它会在 Scripts 文件夹下，这里选择的是 pip2.7.exe，如图。

台电脑 ▶ 本地磁盘 (C:) ▶ Python27 ▶ Scripts

名称	修改日期	类型	大小
img2png	2013/2/27 12:14	.XIT	1 KB
img2png.bat	2017/2/10 21:11	Windows 批处理...	1 KB
img2py	2013/2/27 12:14	文件	1 KB
img2py.bat	2017/2/10 21:11	Windows 批处理...	1 KB
img2xpm	2013/2/27 12:14	文件	1 KB
img2xpm.bat	2017/2/10 21:11	Windows 批处理...	1 KB
pip.exe	2016/8/23 20:47	应用程序	88 KB
pip2.7.exe	2016/8/23 20:47	应用程序	88 KB
pip2.exe	2016/8/23 20:47	应用程序	88 KB
pyalacarte	2013/2/27 12:14	文件	1 KB



接下来，win+r，输入cmd，回车打开系统命令行工具，将 pip2.7.exe 拖动至命令提示符窗口，出现如下代码，然后回车。



```
命令提示符
Microsoft Windows [版本 6.3.9600]
(c) 2013 Microsoft Corporation。保留所有权利。
C:\Users\Admin>C:\Python27\Scripts\pip2.7.exe
```

至此，Python 2.7.13 安装完毕。可在终端上输入命令：`python --version` 来验证 Python 是否正确安装，如下：

```
python --version
```

```
Python 2.7.13
```

6.2、安装 Git

要求版本：1.9.5 及以上；可在终端上输入命令：`git --version`，来验证 Git 是否正确安装，如下：

```
git --version
```

```
git version 2.9.2.windows.1
```

在 win10 安装 git 的时候，可能提示 git 没有安装，把 null.sys 文件复制到 C:\Windows\System32\drivers>null.sys 里面，重启电脑。

6.3、安装 MiCoder Tools

把下载的 MiCoder_v1.1.Win32.zip 工具解压到一个不带中文和空格的目录，比如 C:\MiCoder_v1.1.Win32\MiCoder。

6.4、安装 MiCO-Cube

在 cmd 命令行中输入 `pip install mico-cube`，然后等待安装完成

6.5、设置 MiCoder Tools 路径

执行命令：`mico config --global MICODER C:\MiCoder_v1.1.Win32\MiCoder`



执行命令:mico config --list 检测 MICODER 变量如下

```
C:\Users\91902>mico config --global MICODER C:\MiCoder_v1.1.Win32\MiCoder
[mico] C:\MiCoder_v1.1.Win32\MiCoder now set as global MICODER

C:\Users\91902>mico config --list
[mico] Global config:
MICODER=C:\MiCoder_v1.1.Win32\MiCoder

[mico] Local config (C:\Users\91902):
Couldn't find valid mico program in C:\Users\91902

C:\Users\91902>_
```

6.6、安装 MiCoder IDE

安装 MiCoder_IDE_1_2_1_Win32_x64.exe, 用于编辑代码

6.7、安装 CP2102 驱动

安装 CP210x_VCP_Windows.zip, 用于串口调试

6.8、安装 jlink 驱动

安装 Setup_JLink_V600i.zip 驱动, jlink 推荐使用 V9



6.9、jlink 驱动转换

安装了 jlink 驱动后, 一定要转换驱动后, 才能使用 Mico Ide 和 mico cube 下载。

注意:在使用过程中, 如果突然发现 jlink 下载不了了, 需要检查一下驱动是否转换了。

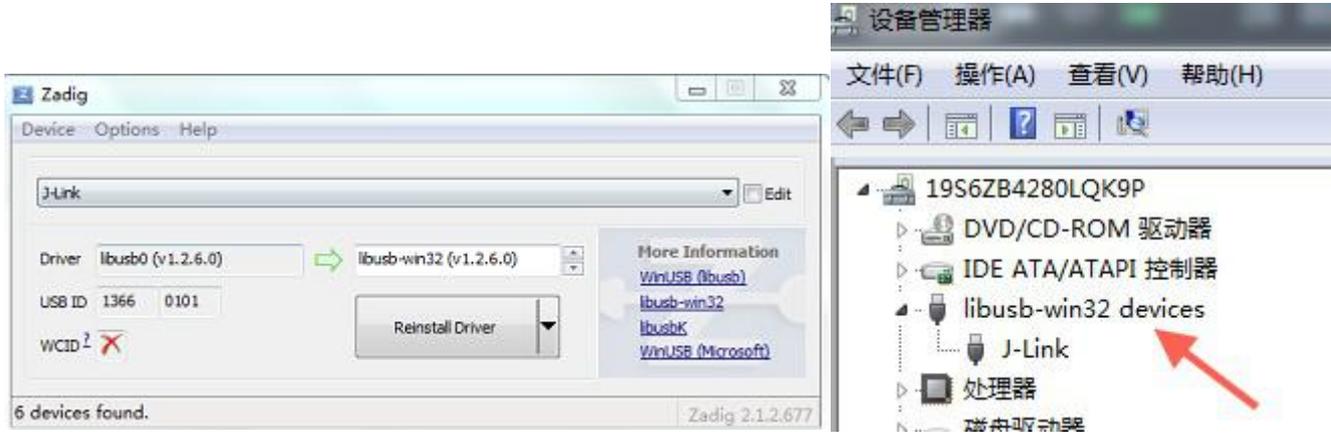
使用 jlink_driver_for_mico.zip 工具进行转换。解压缩后运行: zadig.exe 或 zadig_xp.exe (用于 Windows XP)

名称	修改日期	类型	大小
.DS_Store	2017/1/10 14:57	DS_STORE 文件	7 KB
how_to_use.txt	2017/1/10 15:02	文本文档	1 KB
zadig_2.2.exe	2017/1/10 14:44	应用程序	5,077 KB
zadig_xp_2.2.exe	2017/1/10 14:45	应用程序	5,191 KB

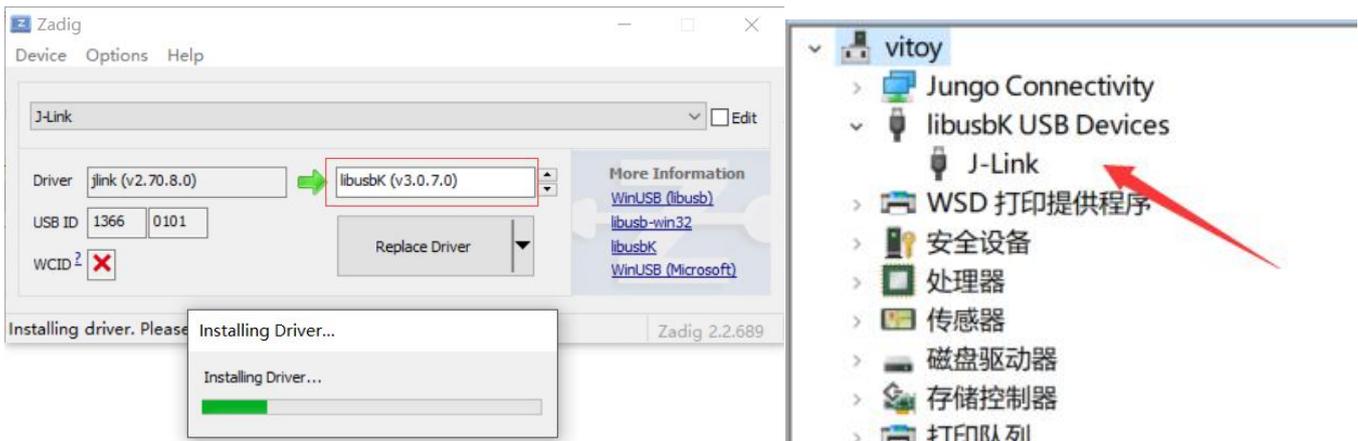


选择菜单：Options->List All Devices，下拉列表选择 J-Link，Driver 选择 “libusb-win32 (v1.2.6.0)”，点击 “Replace Driver”，返回 “successful” 后即可关闭。在设备管理器中会出现 “libusb-win32 devices”。

如果您用的是 JLinkV8，更新过程如下：



如果您用的是 JLinkV9，更新过程如下：



七、二次开发

7.1、下载 mico-demo 工程

查看 Mico 提供的代码方法



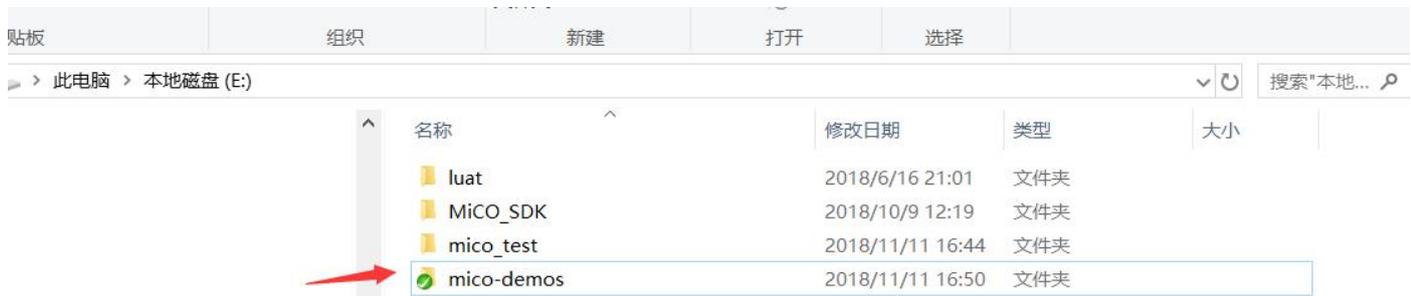
Mico-demo 里面包含了官方提供的全部 demo

下载链接: <https://code.aliyun.com/mico/mico-demos.git>

- 1、使用 cmd 命令行进入自己的工程目录
- 2、执行 `mico import https://code.aliyun.com/mico/mico-demos.git` 命令开始下载最新工程



等待下载结束为止，这个过程可能比较久

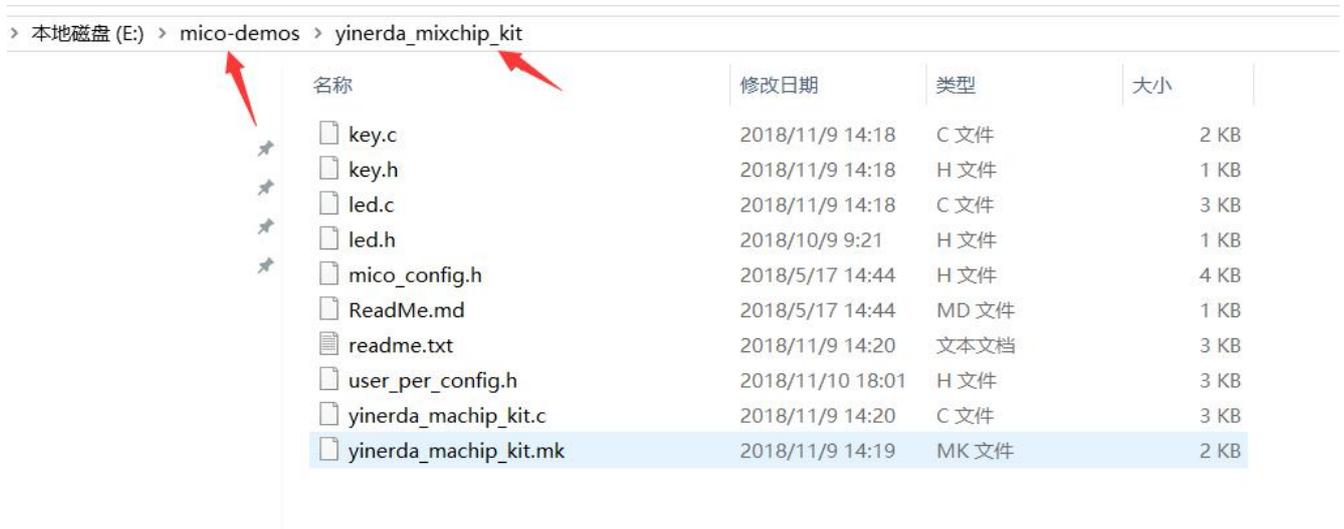


下载好的工程就是 Mico-demos



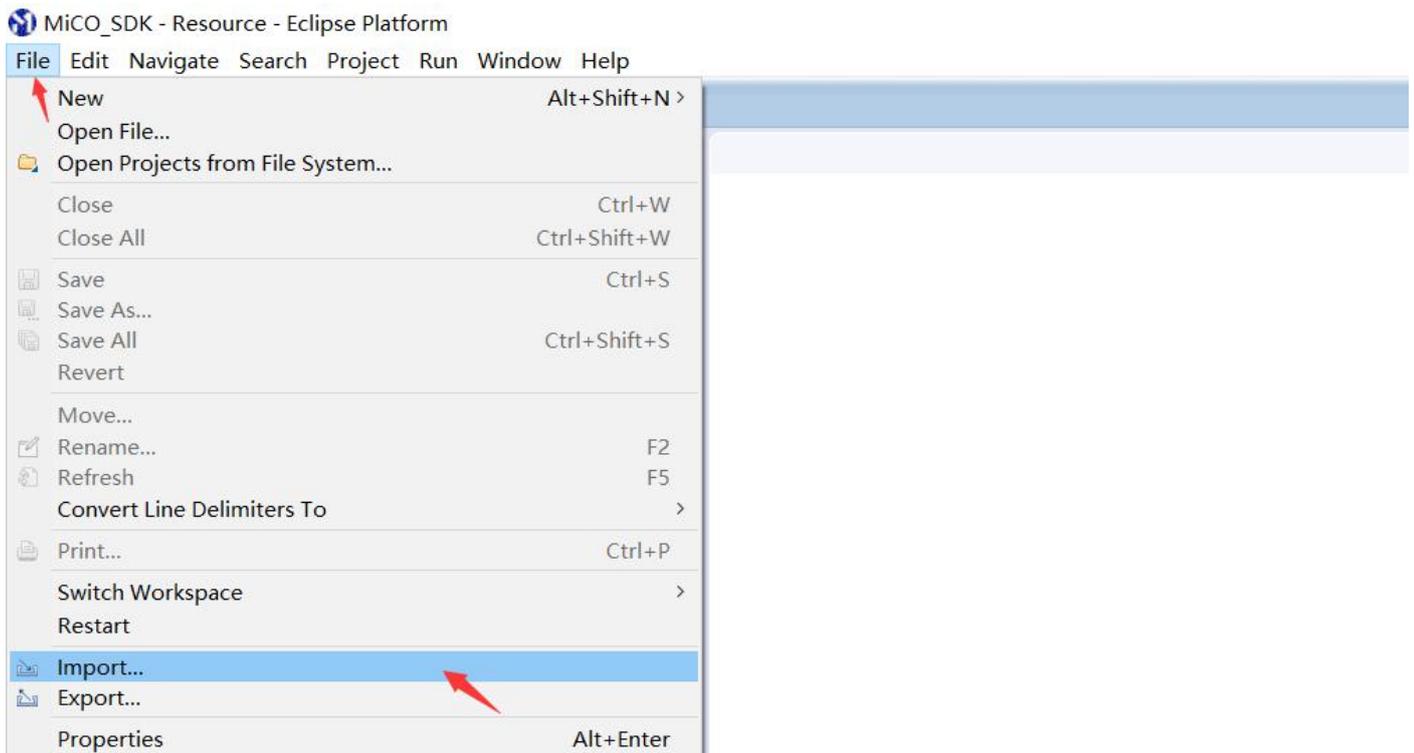
7.2、添加银尔达开发板测试工程

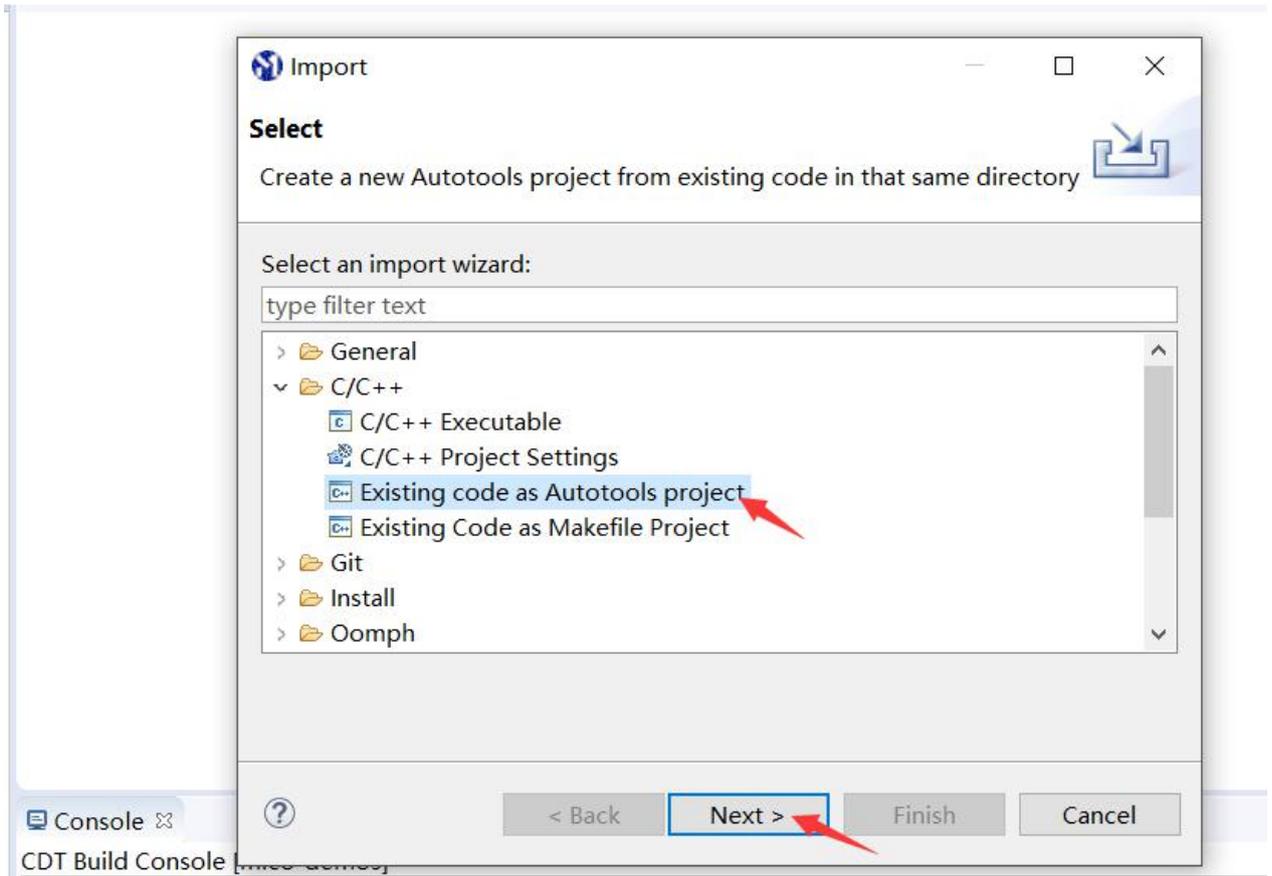
把提供的 Program\庆科开发板测试例程 yinerda_mixchip_kit.zip 解压后拷贝到下载的 mico-demo 目录下



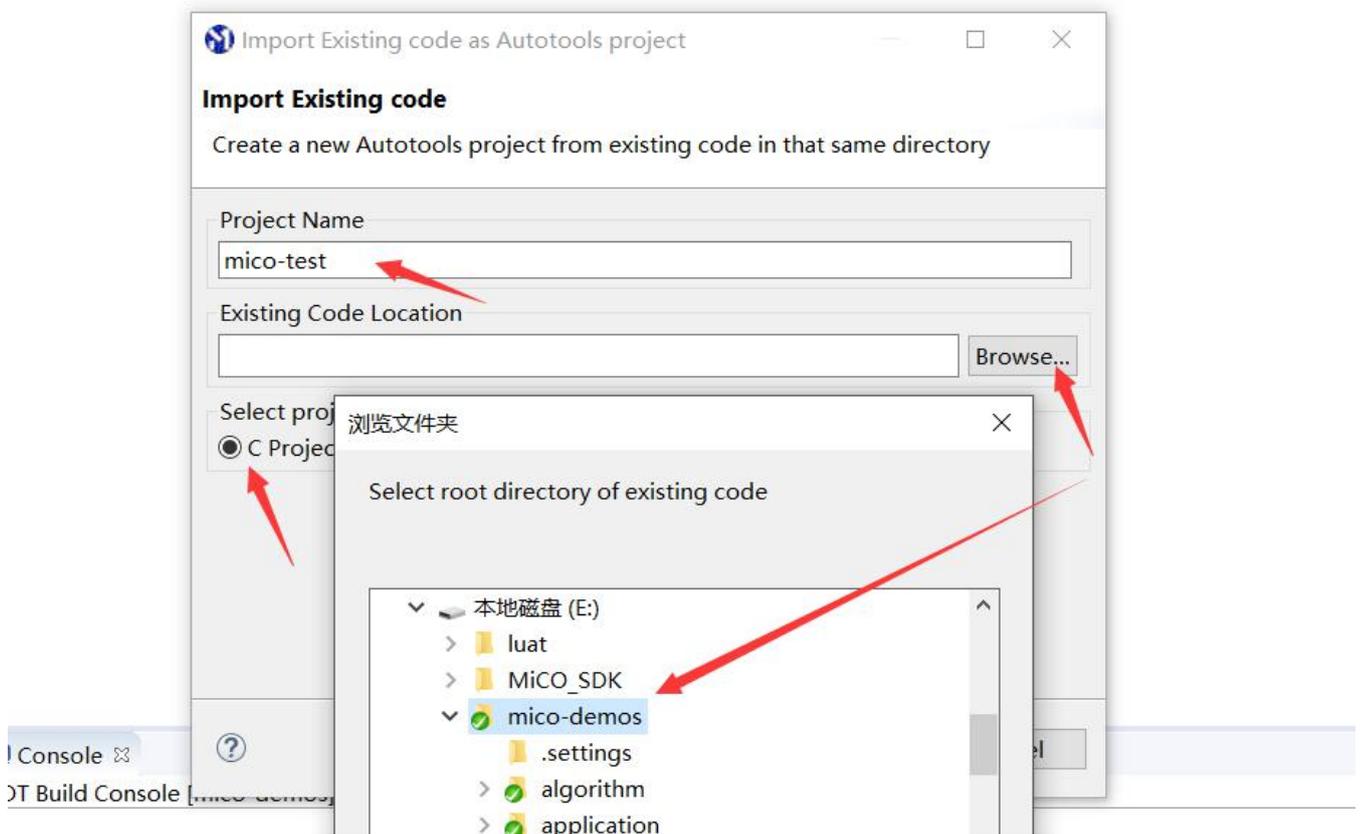
7.3、MiCoder IDE 创建工程

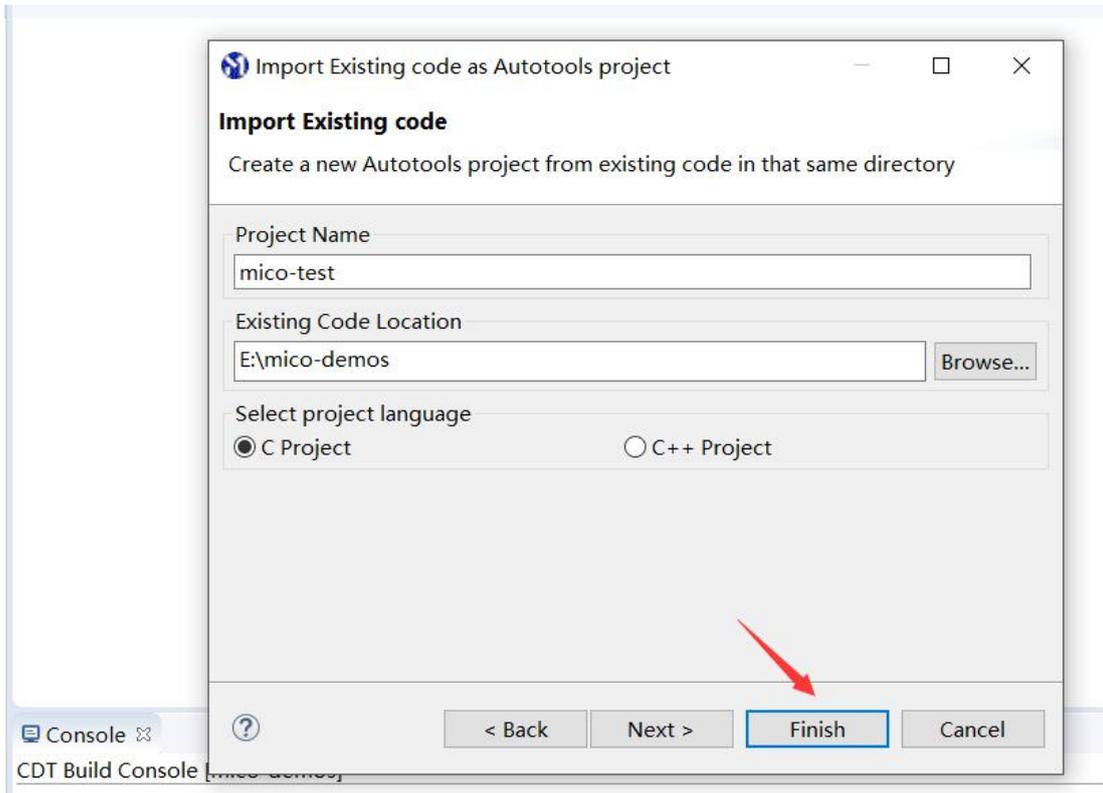
Micodre ide 和 eclipse 的使用方式是一样的。



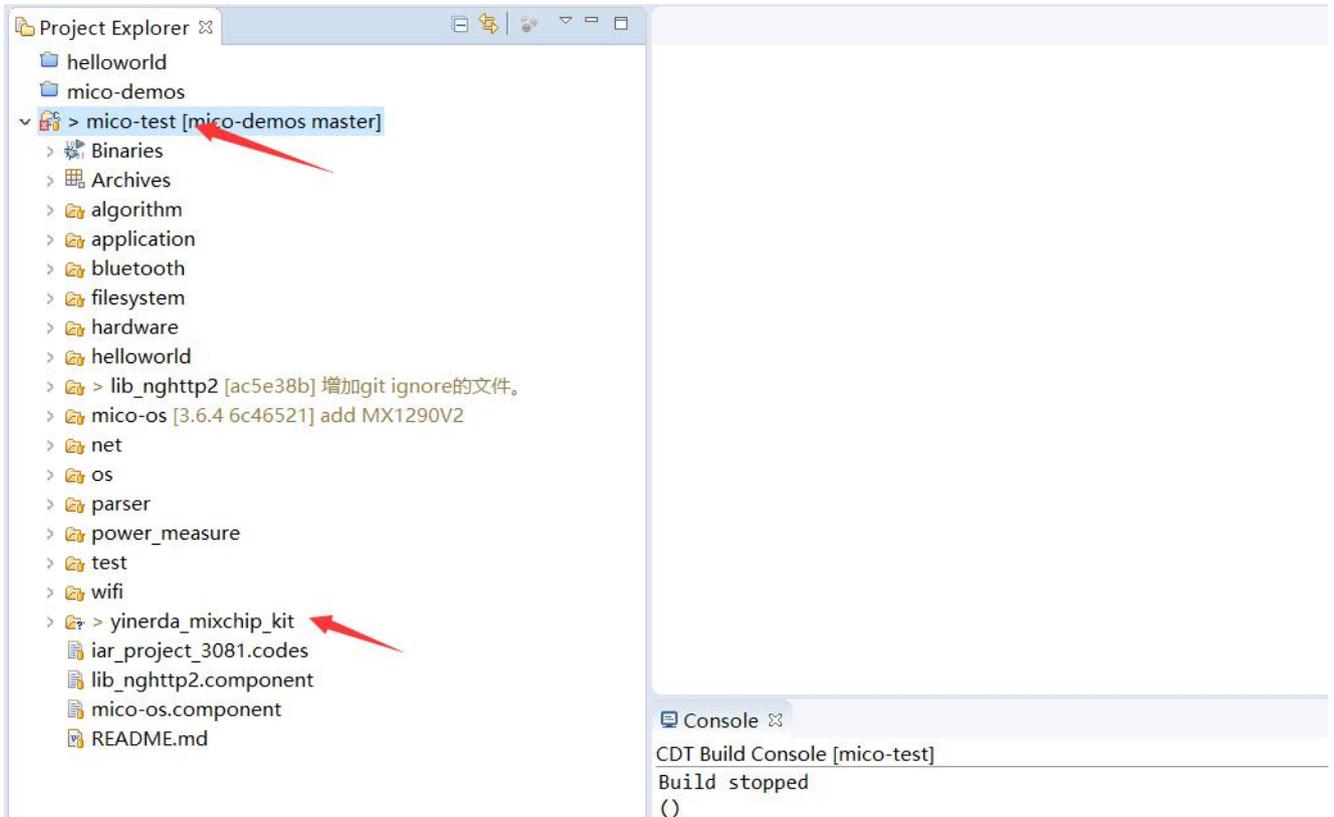


选择下载的 mico-demos 目录





Mico-demo 就是创建的代码工程，yinerda_mixchip_kit 就是我们提供的 demo 工程。



7.4、修改 yinerda_mixchip_kit 代码

yinerda_mixchip_kit 是测试工程，根据开发板配套的模组，修改 user_per_config.h 的宏定义，如果是 EMW3080 就打卡 MY_EMW3080 的宏，屏蔽其他宏。程序里面会根据定义的宏，来确定 LED 和按键的对应管脚。下图以 3080 为例

```
user_per_config.h
2+ * user_per_config.h
7
8 #ifndef YINERDA_MIXCHIP_KIT_USER_PER_CONFIG_H_
9 #define YINERDA_MIXCHIP_KIT_USER_PER_CONFIG_H_
10
11
12 /**需要测试那个模块就打开对应的宏定义,只能打开一个*/
13 #define MY_EMW3080
14- //#define MY_EMW3031
15- //#define MY_EMW3239
16- //#define MY_EMW3165
17- //#define MY_EMW3166
18- //#define MY_EMW3060
19
```

7.5、添加编译 target

不同的模组有不同的编译选项，具体含义参考官方 MiCO Cube 编译命令文档。

```
yinerda_mixchip_kit@MK3080B@moc total download run
```

```
yinerda_mixchip_kit@MK3031@moc total download
```

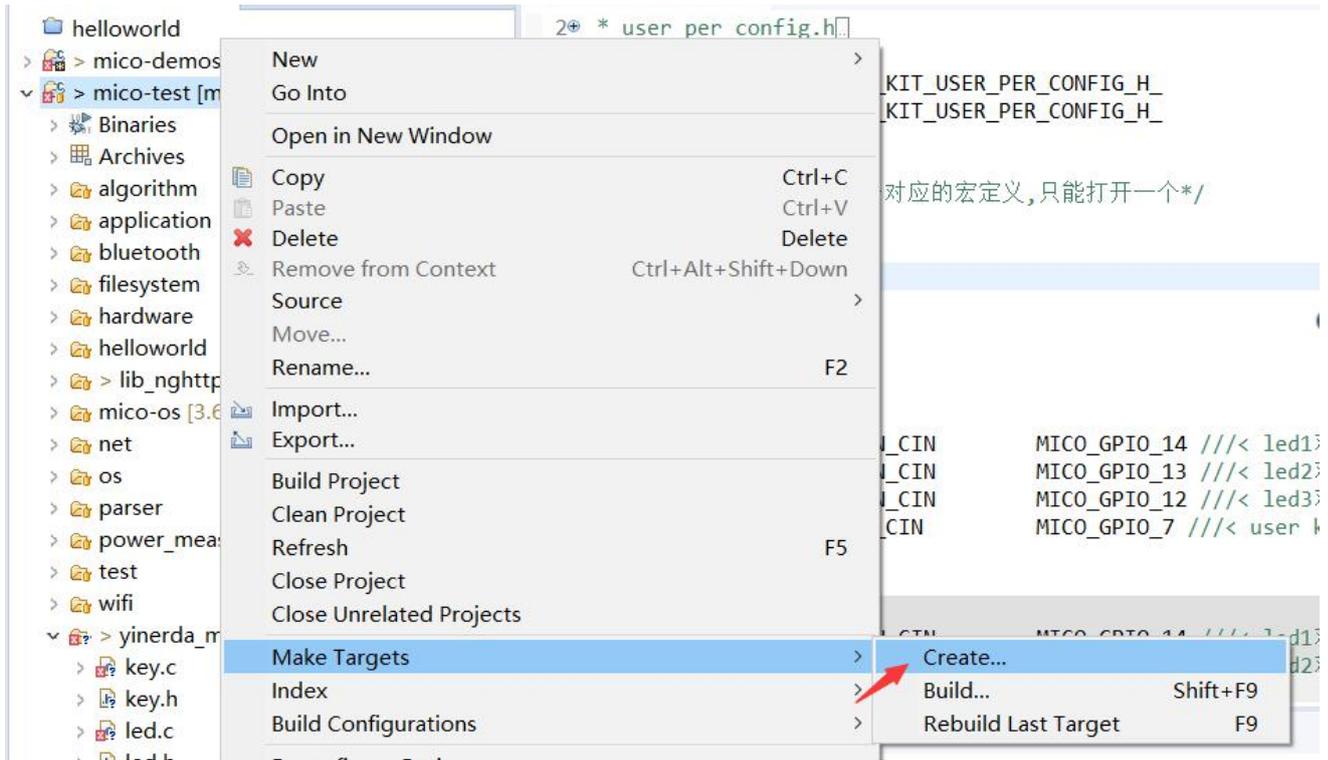
```
yinerda_mixchip_kit@MK3239 download run
```

```
yinerda_mixchip_kit@MK3165 download run
```

```
yinerda_mixchip_kit@MK3166 download run
```

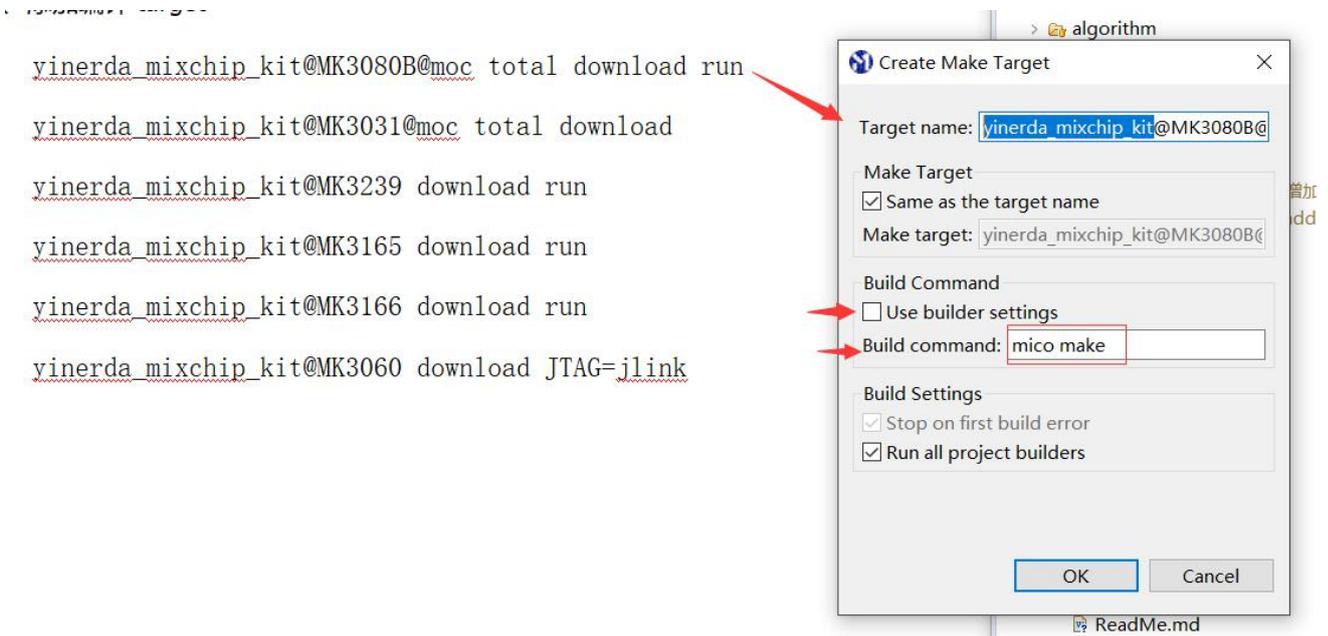
```
yinerda_mixchip_kit@MK3060 download JTAG=jlink
```





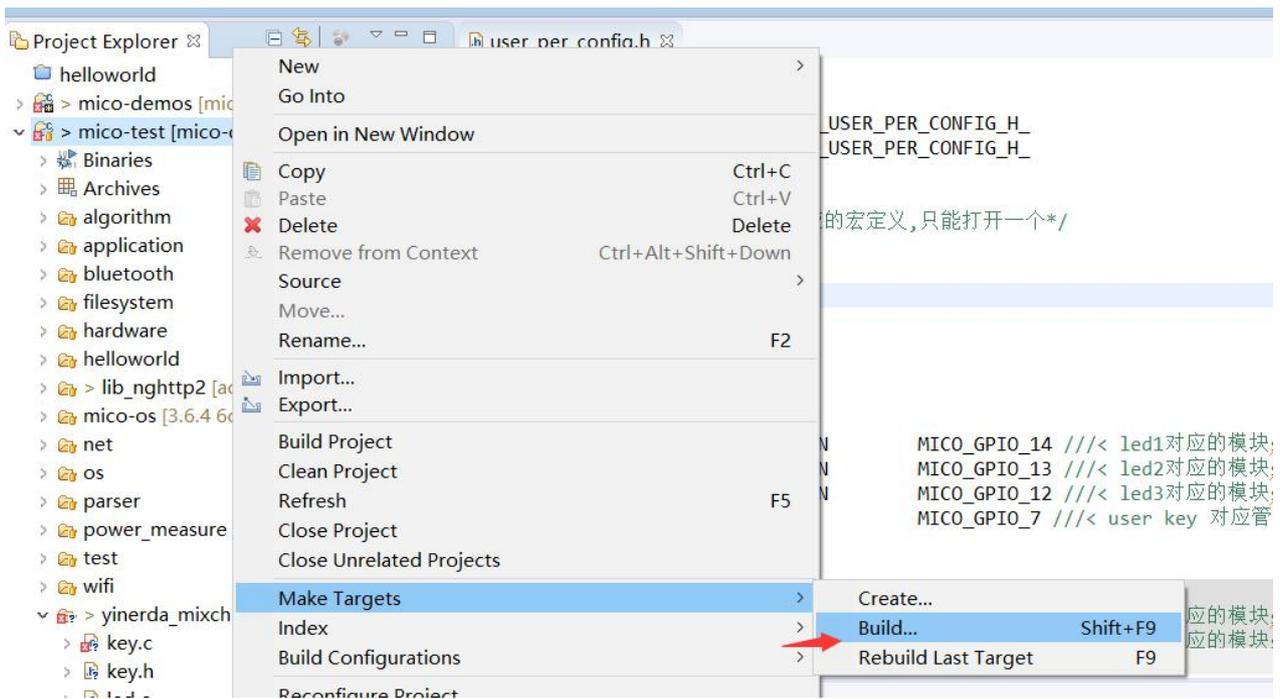
在 target name 中输入需要编译的命令

在 build command 中修改为 mico make



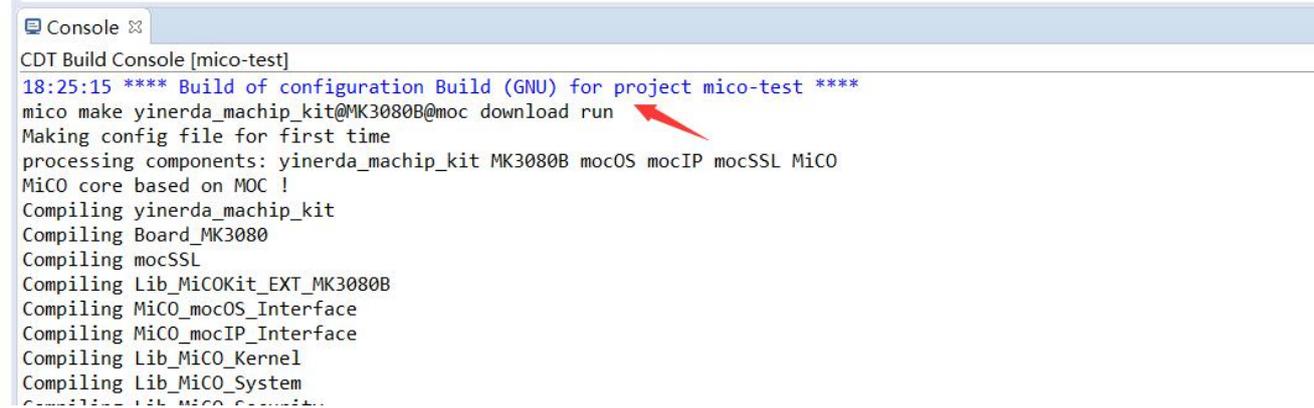
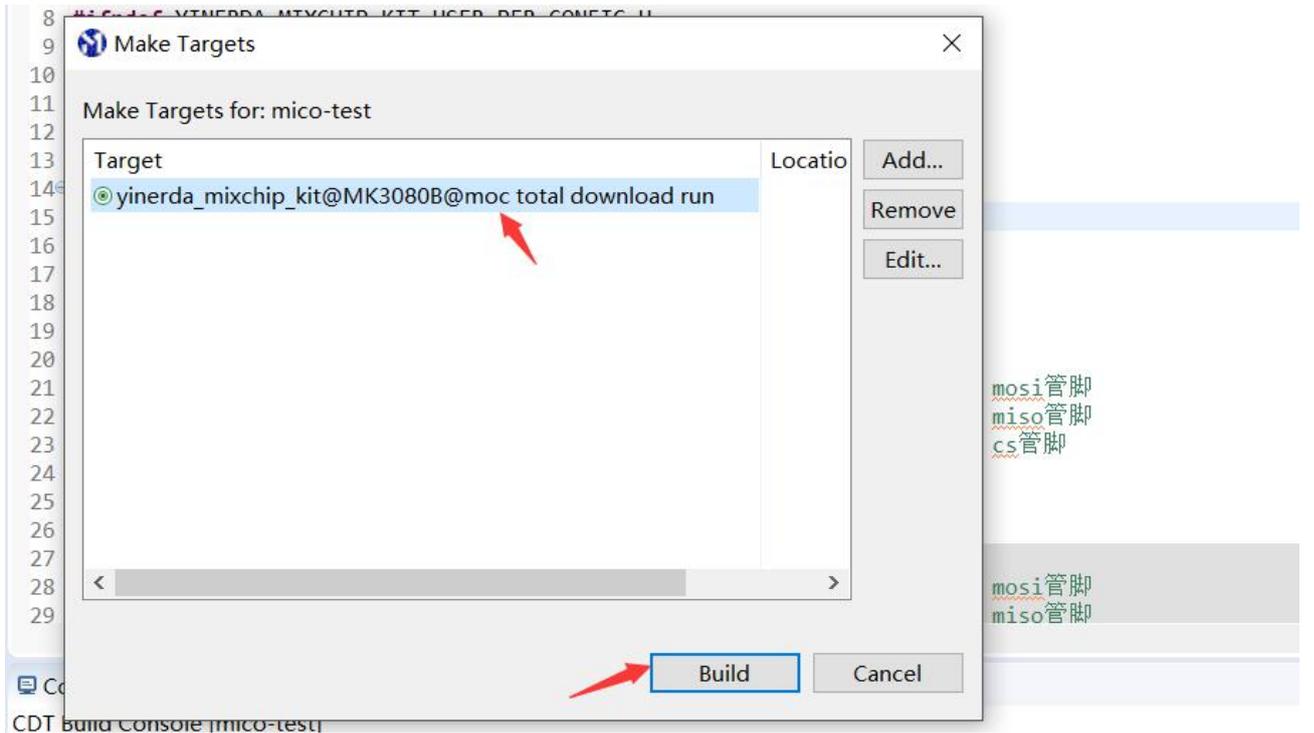
7.6、编译下载固件

首先检查 jlink 是否正常转换驱动，如果没有转换，需要提前转换驱动，转换参考《6.9、jlink 驱动转换》。



选择自己需要编译的命令，点击 build 就可以编译和下载了





编译成功、下载成功

```

Generate MOC OTA Image: ./build/yinerda_machip_kit@MK3080B@moc/binary/yinerda_machip_kit@MK3080B@moc.ota.bin
Generate MOC Flash Images: ./build/yinerda_machip_kit@MK3080B@moc/binary/yinerda_machip_kit@MK3080B@moc.all.bin

Downloading MOC APP to partition: 2 offset: 0x75000 size: 42876 bytes...
Download complete

Resetting target
Target running
Build complete
Making .gdbinit
Making .openocd_cfg

18:46:51 Build Finished (took 13s.786ms)
    
```

编译成功、下载失败，当提示错误的时候，先检查 JLINK 版本，是 V9 还是 V8；然后检查 JLINK 驱动是否转换。

```

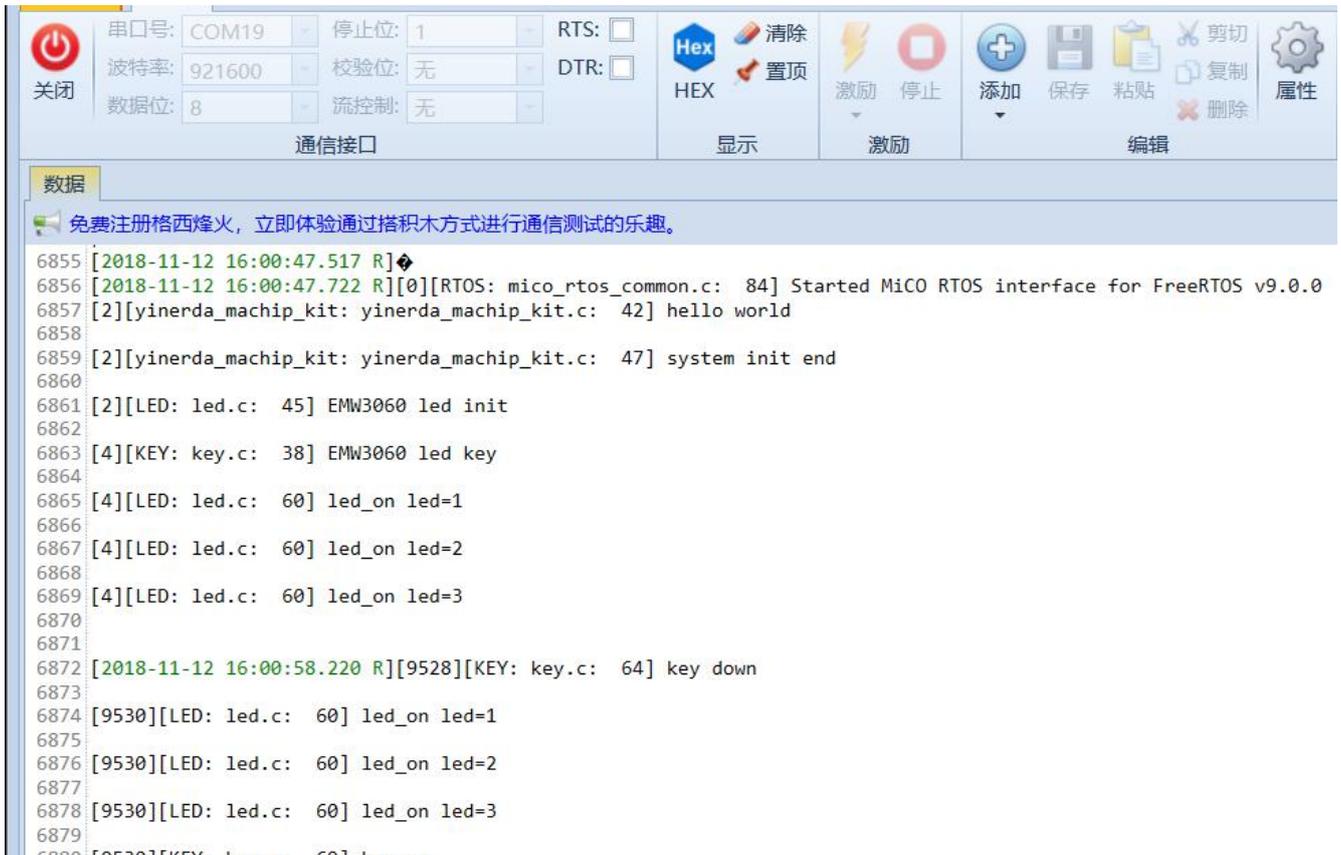
Generate MOC OTA Image: ./build/yinerda_machip_kit@MK3080B@moc/binary/yinerda_machip_kit@MK3080B@moc.ota.bin
Generate MOC Flash Images: ./build/yinerda_machip_kit@MK3080B@moc/binary/yinerda_machip_kit@MK3080B@moc.all.bin

Downloading MOC APP to partition: 2 offset: 0x75000 size: 42876 bytes...
Download failed
Resetting target
mico-os/makefiles/mico_moc_targets.mk:65: recipe for target 'run' failed
make.exe[1]: *** [run] Error 1
mico-os/makefiles/Makefile:144: recipe for target 'main_app' failed
make.exe: *** [main_app] Error 2

18:26:27 Build Finished (took 1m:11s.909ms)
    
```

7.7、固件测试

下载固件后，按下复位按键后，按 USER KEY LED 会亮，松开会熄灭。



The screenshot shows a serial terminal window with the following settings and data:

- 串口号:** COM19
- 波特率:** 921600
- 数据位:** 8
- 停止位:** 1
- 校验位:** 无
- 流控制:** 无
- RTS:**
- DTR:**

数据

```

[2018-11-12 16:00:47.517 R]
[2018-11-12 16:00:47.722 R][0][RTOS: mico_rtos_common.c: 84] Started MiCO RTOS interface for FreeRTOS v9.0.0
[2][yinerda_machip_kit: yinerda_machip_kit.c: 42] hello world
[2][yinerda_machip_kit: yinerda_machip_kit.c: 47] system init end
[2][LED: led.c: 45] EMW3060 led init
[4][KEY: key.c: 38] EMW3060 led key
[4][LED: led.c: 60] led_on led=1
[4][LED: led.c: 60] led_on led=2
[4][LED: led.c: 60] led_on led=3
[2018-11-12 16:00:58.220 R][9528][KEY: key.c: 64] key down
[9530][LED: led.c: 60] led_on led=1
[9530][LED: led.c: 60] led_on led=2
[9530][LED: led.c: 60] led_on led=3
    
```

八、二次开发注意事项

8.1、EMW3060 开发注意事项

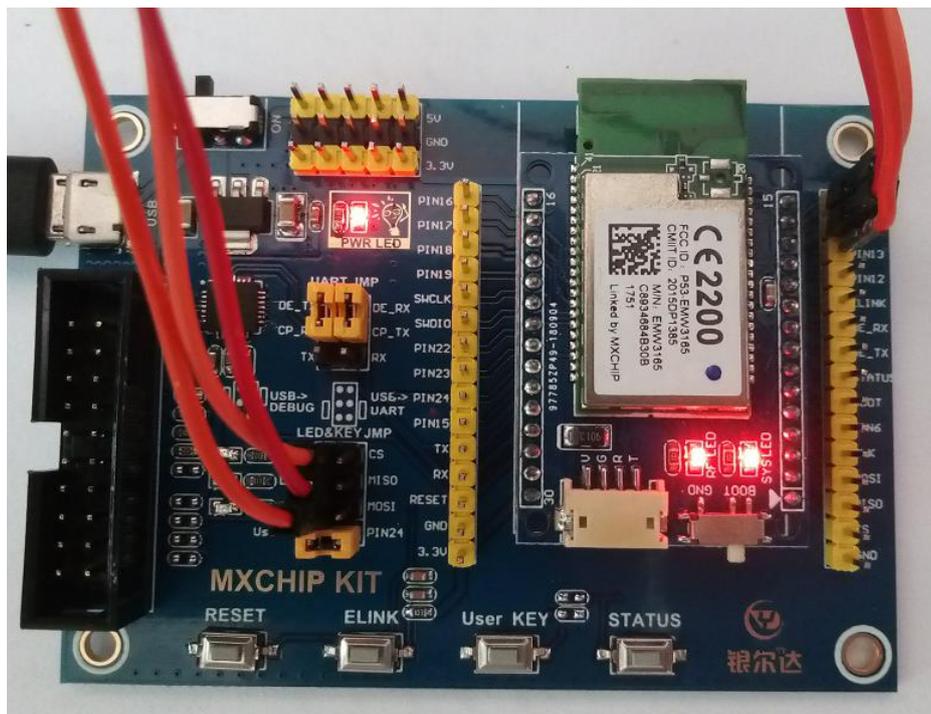
EMW3060 模组在使用 JLINK 下载的时候，WIFI 转接板上面的 BOOT 开关拨到 GND，按下 Reset 按键后才能下载。当下载完成后，把 BOOT 拨到 BOOT，按下复位按键就能运行新下载的程序。



8.2、EMW3165 开发注意事项

EMW3165 底板的 LED1、LED2、LED3 的管脚已经被 EMW3165 的 flash 管脚占用了，用户是不能使用的，所以需要把 LED 与 WIFI 管脚连接的跳线帽取掉。然后使用其他的可还是用管脚来控制 LED。

在 yinerda_mixchip_kit demo 中，EMW3165 是使用底板的 PIN5, PIN4, PIN3 来控制 LED 的

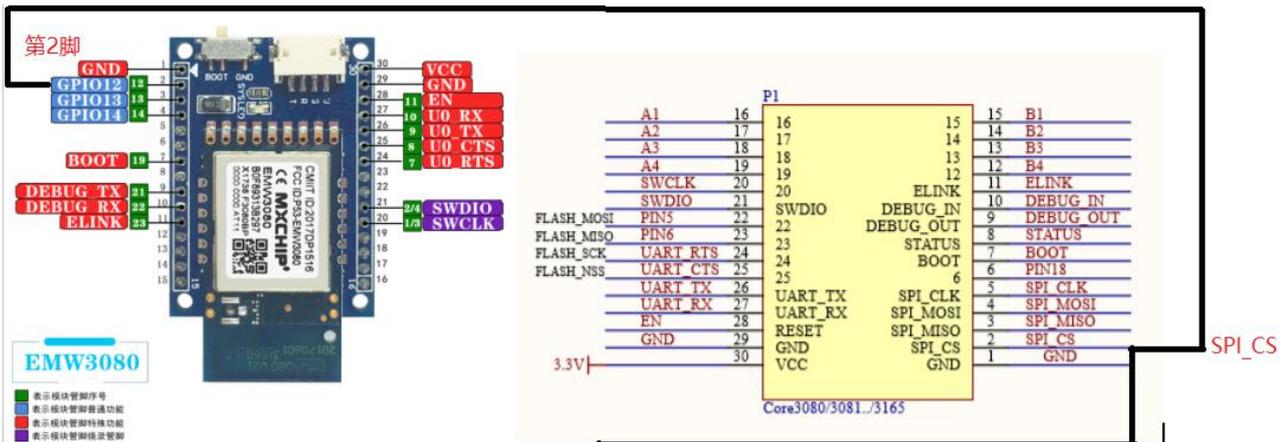
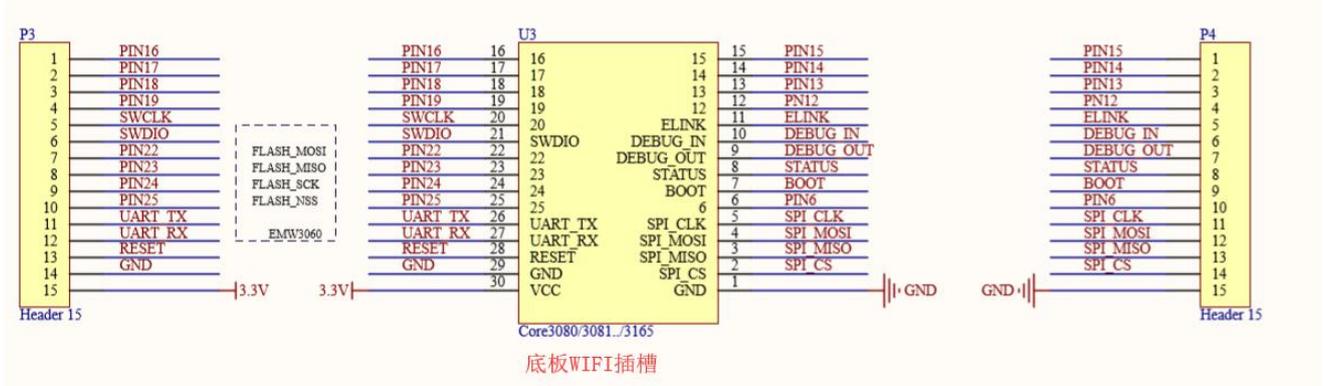


九、原理图分析

EMW3080、EMW3060、EMW3081、EMW3031 转接板原理图为《Core3080_3060_3081_3031.pdf》

EMW3165, EMW3166、EMW3239 转接板原理图为《Core3165_3166_3239.pdf》

原理图分析原理是相同的，EMW3080、EMW3060、EMW3081、EMW3031 的封装一样，下面以 EMW3080 为例。



模组的实物照片三角形为第 1 脚 (GND)。

底板的管脚和模组转接板管脚一一对应。

P1 是转接板的管脚图； U3 (EMW3031) 是模组的管脚图，上面编号与模组的编号一一对应。

分析 EMW3080:

分析底板的 LED1, LED1 对应底板的第 2 脚；底板的第 2 脚对应转接板的第 2 脚；P1 的第 2 脚是 SPI_CS, U3 的 SPI_CS 对比管脚编号是 12, 所以就对应模组的 12 管脚，及 MICO_PWM3, 根据《DS0069CN_EMW3080(AB)_V1.3.pdf》可以知道 MICO_PWM3 是 MICO_GPIO_12。

分析 EMW3165:

分析底板的 LED1, LED1 对应底板的第 2 脚；底板的第 2 脚对应转接板的第 2 脚；P1 的第 2 脚是 SPI_CS；《Core3165_3166_3239.pdf》中 U1 的 WIFI 模组 SPI_CS 对比管脚编号是 5, 所以就对应模组的 5 管脚，根据《DS0007C_EMW3165_V2.1.pdf》可以知道 EMW3165 的第 5 脚是 PA15(Flash_SPI1_NSS), 并且这个管脚用户不能使用。所以 EMW3165 在使用底板的时候，LED 需要使用跳线帽连接到其他用户可以使用的管脚，在 yinerda_mixchip_kit_demo 中使用了 PB10。

