

MicoKit快速入门

wanges@mxchip.com

2015.5.19

目录

0、快速入门

1、准备工作

- 1.1 获取MicoKit开发板
- 1.2 下载手机APP

2、设置开发板

- 2.1 连接开发板
- 2.2 设置Wi-Fi连接
- 2.3 激活设备

3、控制开发板

4、定制开发板

- 4.1 注册开发者账号
- 4.2 获取SDK开发包
- 4.2 安装开发环境
- 4.2 获取demo源代码
- 4.4 云端创建新产品
- 4.5 激活设备到新产品
- 4.6 设备添加新模块
- 4.7 手机APP访问新模块

5、FAQ

0、快速入门

0、快速入门

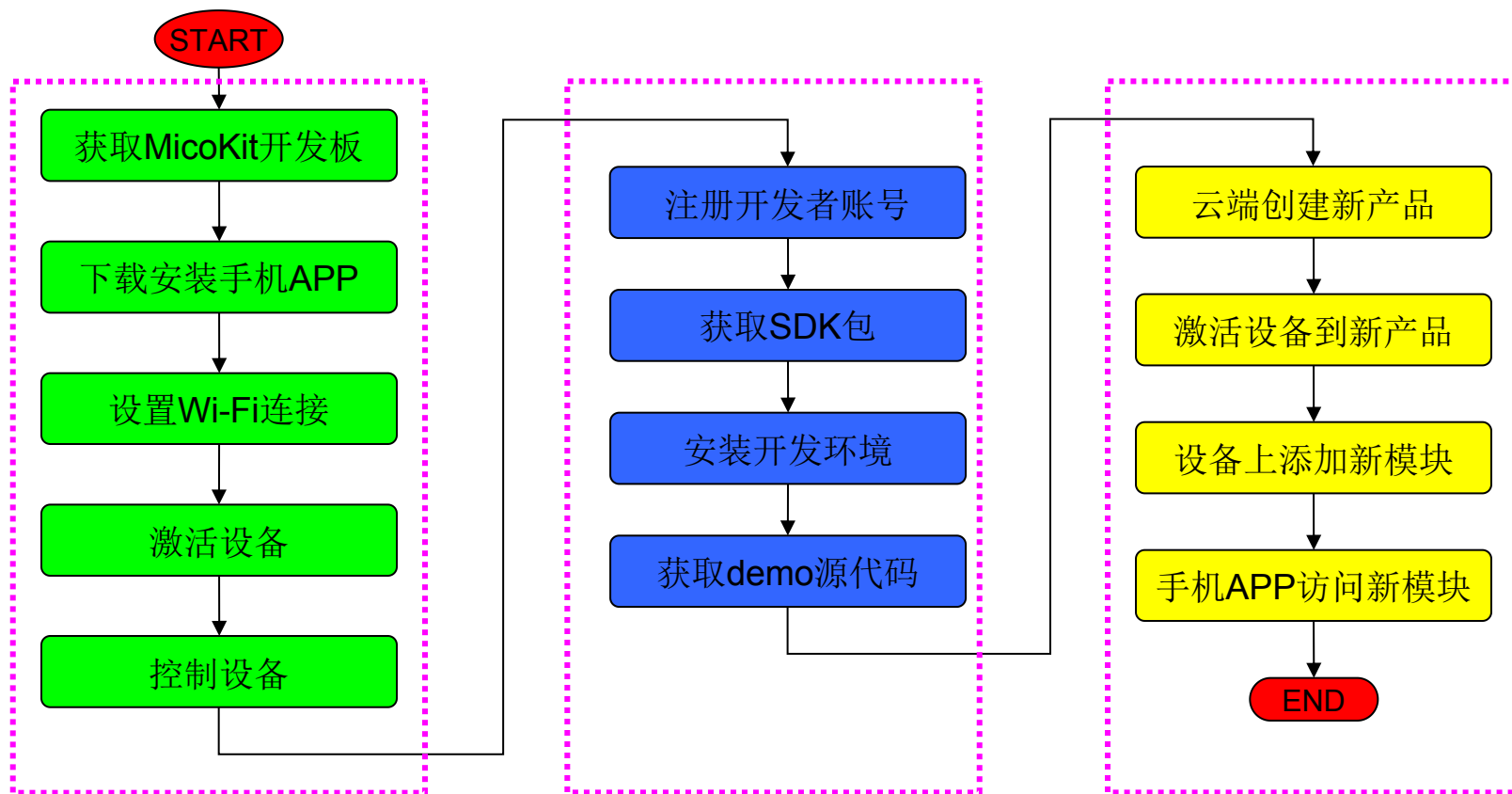
庆科物联网系统由三大部分组成：MiCO设备、FogCloud云端服务、控制端APP。

本文主要介绍如何使用MicoKit作为设备端，并使用手机APP(Android/iOS)对MicoKit进行控制；以及设备固件开发方法。



0、快速入门

- 使用、开发流程



1、上手体验

2、开发准备

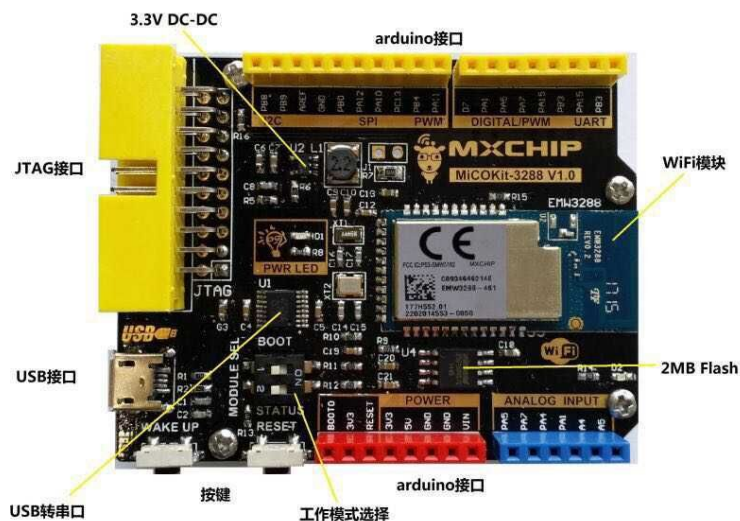
3、定制开发板

1、准备工作

1、准备工作

1.1、获取MicoKit开发板

请联系我司销售申请MicoKit开发板。



底板



扩展板

MicoKit开发板由底板和扩展板上下两层组成；上层是外设扩展板，连接各种外设传感器，用于模拟实际应用功能；下层是底板，主要包括庆科Wi-Fi模块和供电、调试、以及扩展接口。

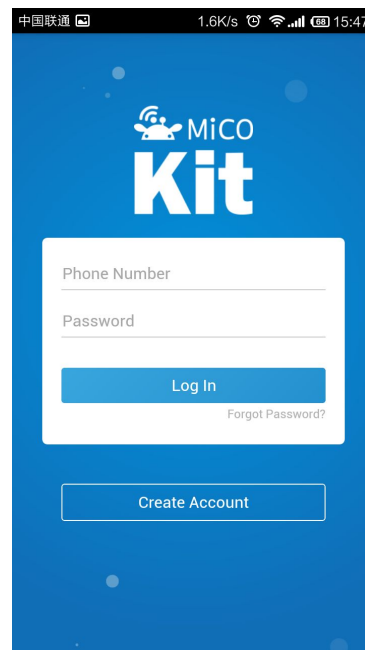
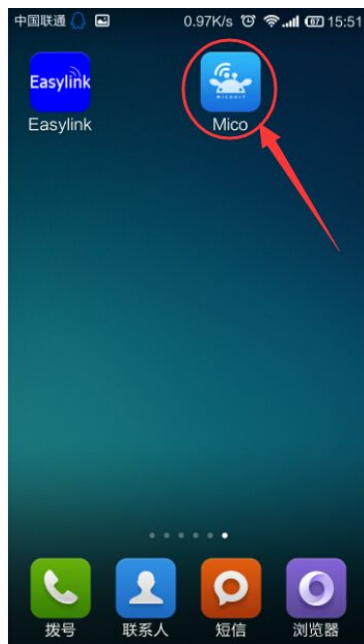
1、准备工作

1.2、下载手机APP

开发套件提供配套的手机APP(Android/iOS)，供演示及固件开发使用。

下载、安装方法：

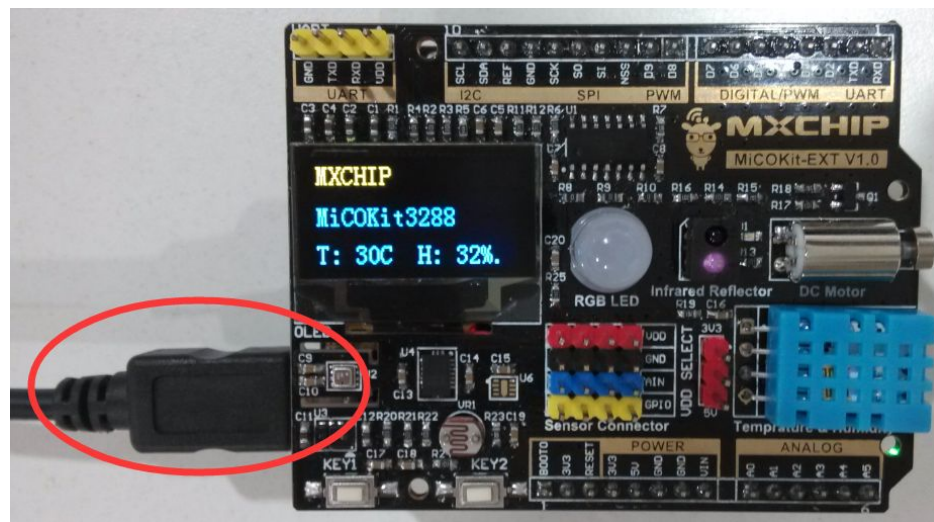
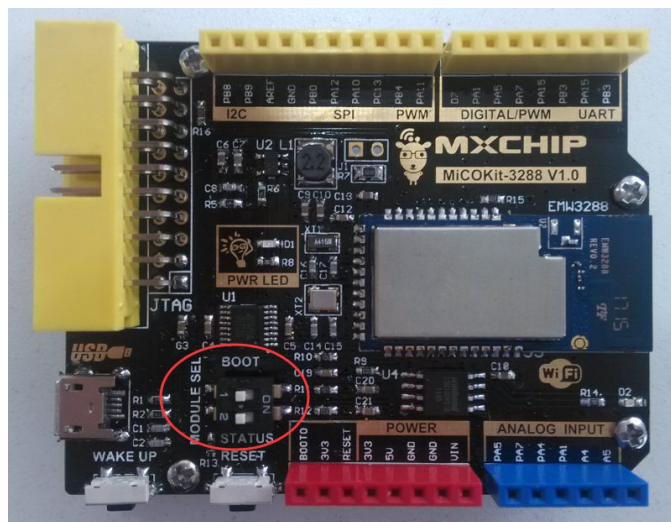
- 1、扫描开发板背面的二维码，关注微信公众号；
- 2、进入公众号首页，根据提示下载相应版本的APP(Android/iOS)，并安装；
- 3、注册APP用户账号并登陆。



2、设置开发板

2.1、连接开发板

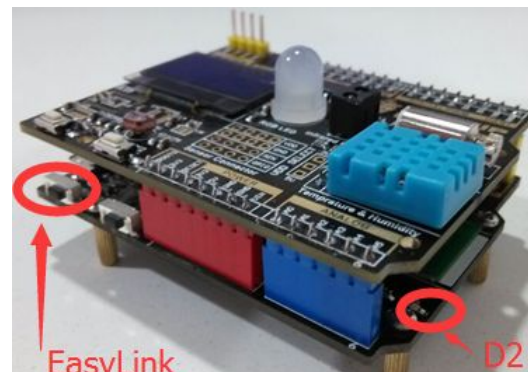
- 1、底板上的SW开关均置于OFF，系统默认执行Application模式，并插好扩展板；
- 2、使用mini USB电缆将底板上的USB口连接到PC（这样可以通过PC的串口工具查看设备运行LOG），也可以使用5V的电源供电。



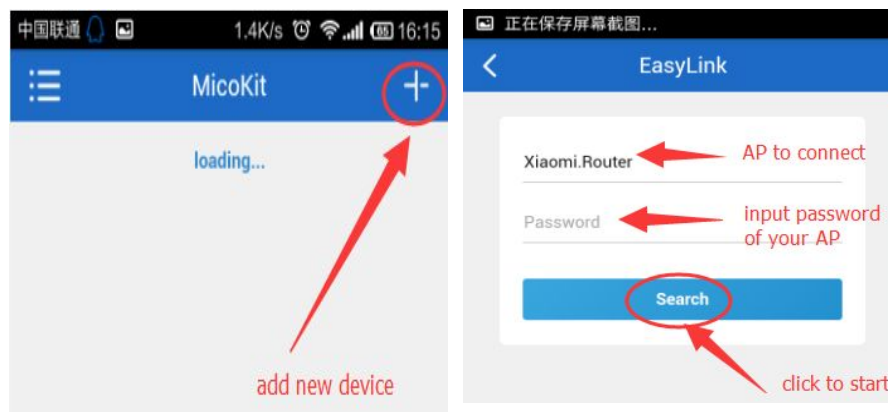
2、设置开发板

2.2、设置Wi-Fi连接

使用MicoKit需要先连入可访问互联网的Wi-Fi网络，当连接成功时，D2（底板右下角）灯会常亮，当网络环境变化时，需要按下述方法重新配置Wi-Fi网络。



- 1、**[APP]**点击APP右上角的加号，进入EasyLink页面；
- 2、**[KIT]**短按设备上的EasyLink按键，进入EasyLink模式（D2快闪）；
- 3、**[APP]**APP上输入要连接的Wi-Fi的密码，点击"search"开始配置；
- 4、设备收到APP发送的SSID和密码后，尝试连接AP，D2慢闪；连接成功后，D2常亮，此时Wi-Fi配置成功，设备自动重启，重新连接刚设置的Wi-Fi，成功后D2常亮。



注意：

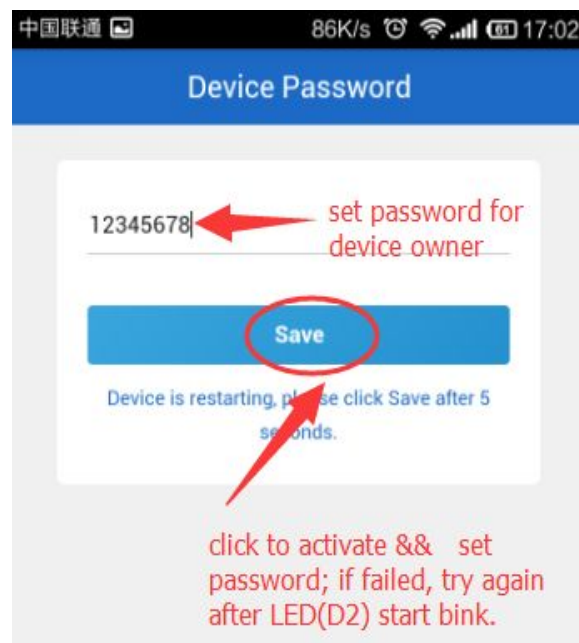
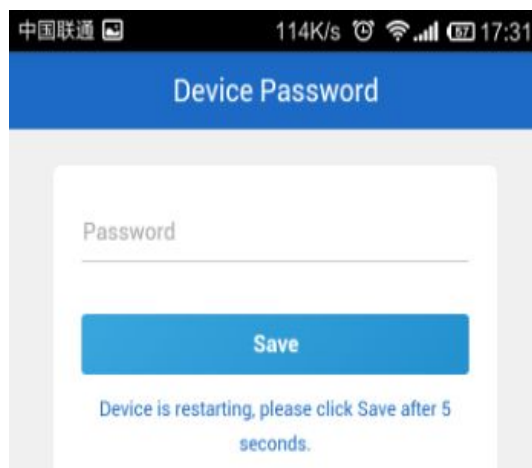
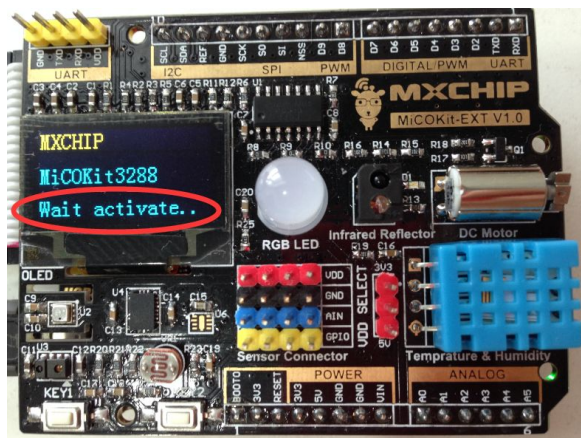
- (1) Wi-Fi配置成功后APP上会跳转到设置密码页面，提示用户设置密码以激活（多用户绑定）设备，即将进入下一个步骤：激活（绑定）设备，具体方法请看下一步骤2.3。
- (2) 如果设备已经激活（绑定）过，则可不进行后续操作，直接退出。

2、设置开发板

2.3、激活设备

设备Wi-Fi配置完成后，APP自动进入设备激活（多用户绑定）步骤。APP上提示用户输入设备密码（多用户授权时验证密码）。

- 1、[KIT] 设备OLED上提示“Wait activate...”，等待APP激活设备；
- 2、[APP]APP上设置密码，点击"save",开始激活设备；
- 3、[KIT] 设备激活成功后，开始连接云端；
- 4、[APP] APP会返回到设备列表，刷新即可看到新设备。



3、控制开发板

3、控制开发板

经过以上配置步骤，MicoKit开发板已经激活到了我们事先创建的测试产品类中，并且已经连接到了FogCloud云端，我们可以使用Demo App进行控制演示了。

- 1、打开手机APP，登录后进入首页设备列表；
- 2、下拉刷新设备列表，即可看到已绑定的设备；
- 3、点解相应的设备，进入控制界面；

- 4、进入RGB LED灯界面，可以调节开发板上的LED灯的颜色、亮度等；
- 5、向开发板吹一口气，改变环境温湿度，APP上温湿度的值会相应更新；
- 6、用手挡住光照传感器、红外反射器，APP上相应的数据会发生变化。

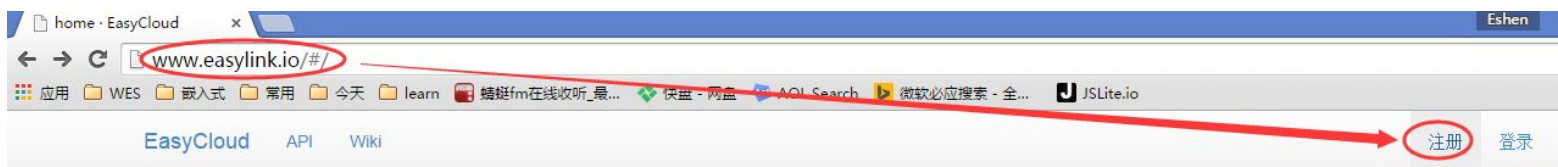


4、定制开发板

4、定制开发板

4.1、注册开发者账号

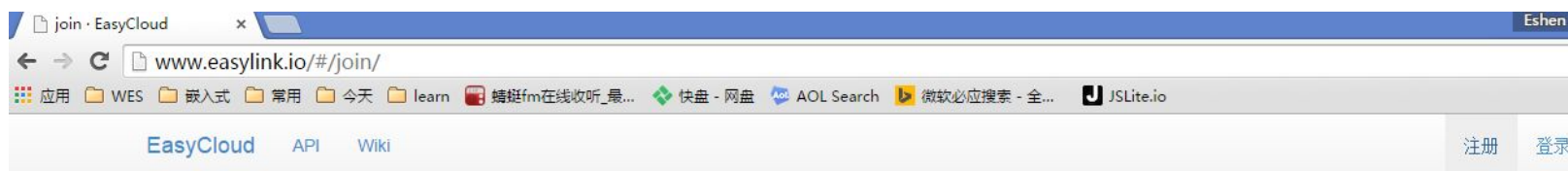
1、登录FogCloud开发者网站(<http://www.easylink.io>), 注册FogCloud开发者账号。



4、定制开发板

4.1、注册开发者账号

2、填写资料，完成注册，并登录。



注册

用户名	<input type="text" value="Email/Phone"/>	← 手机号
密码	<input type="password" value="Password"/>	← 账号密码
	<input type="button" value="获取验证码"/>	← 获取手机验证码
验证码	<input type="text" value="验证码"/>	← 填写获取的验证码
	<input type="button" value="注册"/>	← 填写完成，开始注册

4、定制开发板

4.2、下载SDK开发包

登录MICO开发者网站(mico.io), 可下载MicoKit相关文档和固件SDK开发包(MicoKit Firmware SDK)。



The screenshot shows a web browser window with the URL mico.io in the address bar. The page features a navigation menu with the following items: 首页 (Home), 技术 (Technology), 解决方案 (Solutions), 生态 (Ecosystem), and 开发者中心 (Developer Center). The '开发者中心' link is circled in red, and a red arrow points from the address bar to it. Below the navigation menu, there is a blue banner with the text '把未联网的都连接起来!' (Connect everything that is not connected!). To the right of the banner is a diagram illustrating the IoT ecosystem, showing a mobile app connected to a cloud, which is connected to a central MICO/MXCHIP chip. This chip is connected to various IoT devices: a smart home (家电), lighting (照明), health (健康), security (安防), and entertainment (娱乐).

把未联网的都连接起来!

庆科 (MXCHIP) 与合作伙伴们共同的愿景: 基于标准的快速物联网设备开发和规模化商业应用。

大家一起努力, 我们就能创造全新的IoT操作系统、软件、工具、平台乃至整个生态, 使这个愿景成为现实。

我们的理念

4、定制开发板

4.2、下载SDK开发包

MicoKit Firmware SDK目录结构如下：

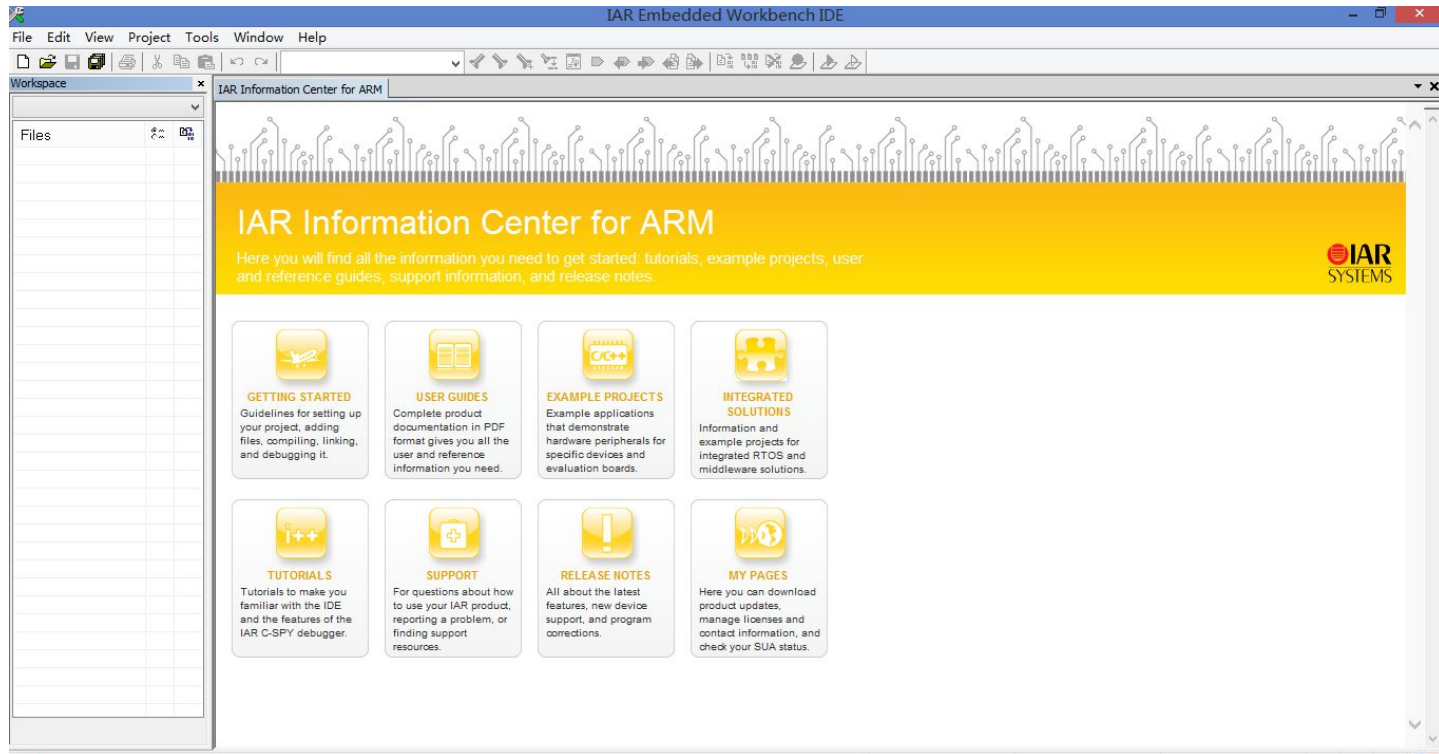
电脑 ▶ 新加卷 (D:) ▶ MXCHIP ▶ MICO ▶ MicoKit ▶ MicoKit3288 ▶ SDK ▶ MicoKit_Firmware_SDK_v1.0.0 ▶

名称	修改日期	类型	大小
 bin	2015/5/18 9:55	文件夹	
 Docs	2015/5/18 16:13	文件夹	
 Project	2015/5/15 14:27	文件夹	
 Tools	2015/5/15 16:23	文件夹	
 Readme.txt	2015/5/15 16:37	文本文档	2 KB

4、定制开发板

4.3、安装开发环境

1、安装IAR workbench for ARM on Windows(7.30.1及以上)(安装文件请自行到IAR官网下载<https://www.iar.com/>);

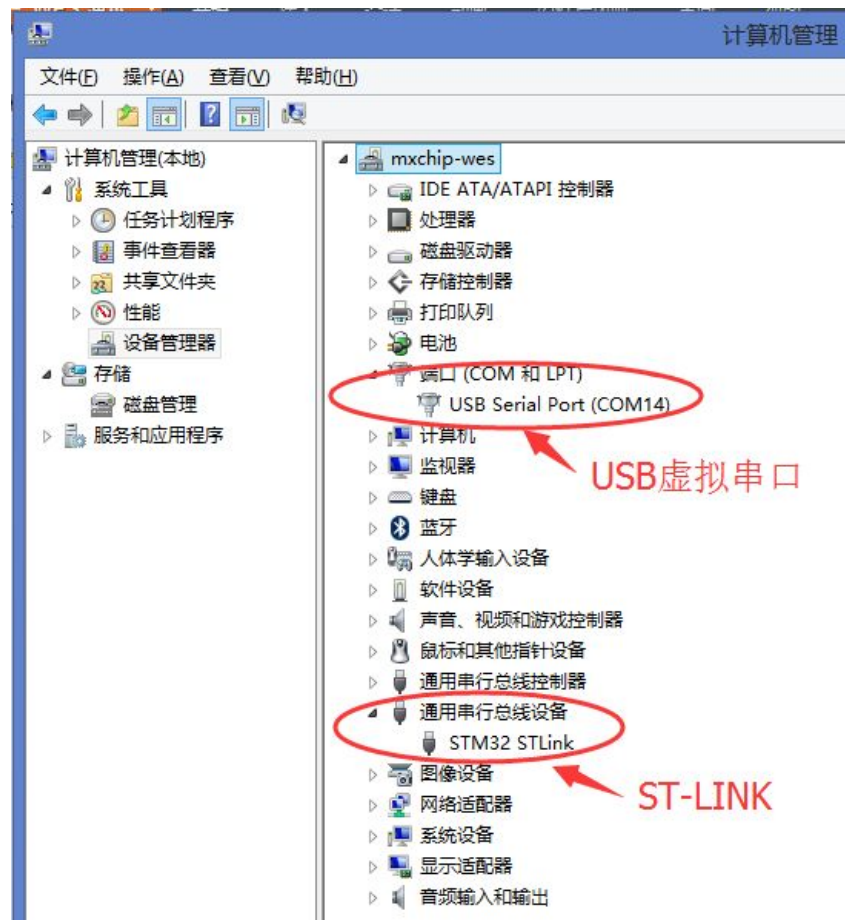


4、定制开发板

4.3、安装开发环境

- 2、解压MicoKit_Firmware_SDK包；
- 3、安装ST-LINK驱动；
- 4、安装USB 虚拟串口驱动(FTDI)。

驱动安装完成后，将ST-LINK连接开发板和PC，并使用USB线连接底板USB口和PC，在设备管理器会看到相应的设备。



4、定制开发板

4.4、获取demo源代码

在MicoKit Firmware SDK包中提供了demo固件的IAR版工程源代码，供开发者修改使用。

1、工程路径：MicoKit_Firmware_SDK_v1.0.0\Project\;

电脑 > 新加卷 (D:) > MXCHIP > MICO > MicoKit > MicoKit3288 > MicoKit_Firmware_SDK_v1.0.0 > Project

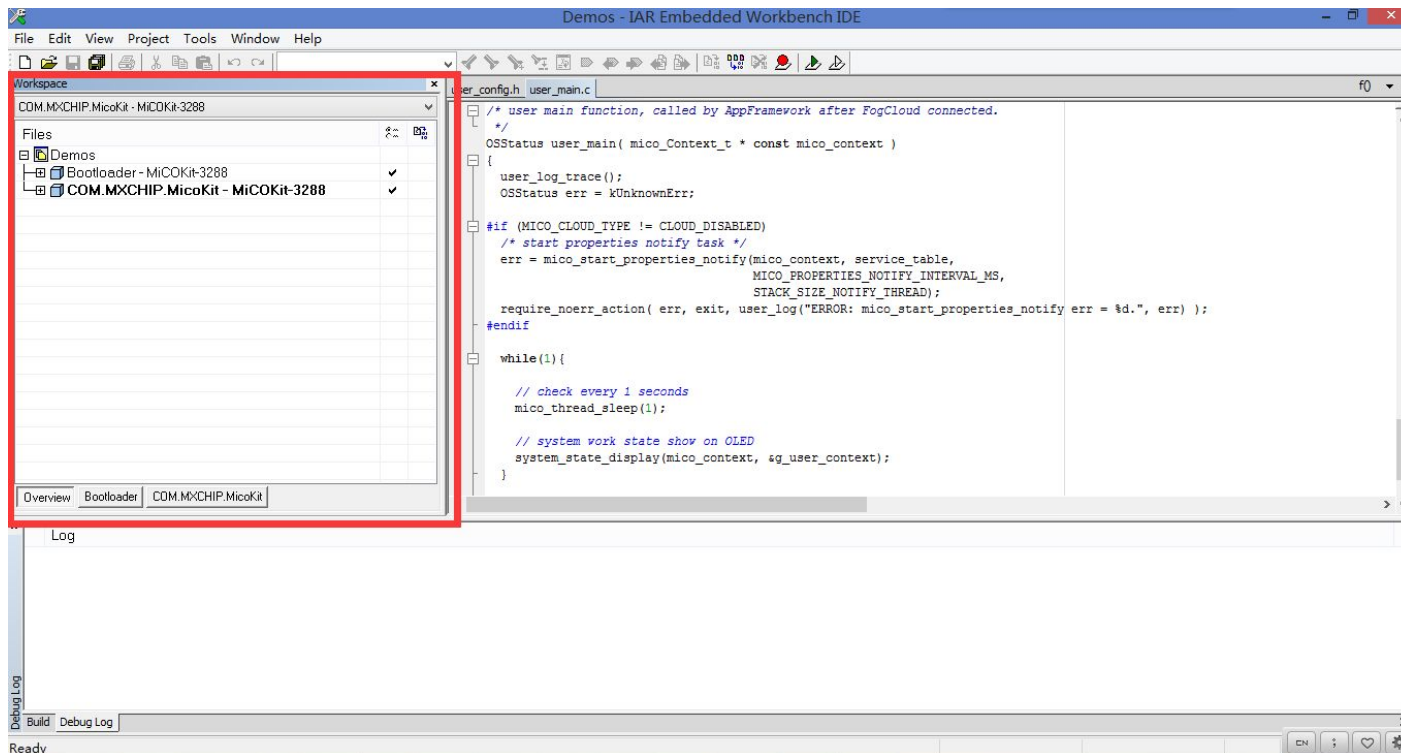
名称	修改日期	类型	大小
Board	2015/5/15 14:27	文件夹	
Bootloader	2015/5/15 14:27	文件夹	
Build	2015/5/15 14:43	文件夹	
Demos	2015/5/15 14:27	文件夹	
Document	2015/5/15 14:27	文件夹	
External	2015/5/15 14:27	文件夹	
include	2015/5/15 14:27	文件夹	
MICO	2015/5/15 14:27	文件夹	
Platform	2015/5/15 14:27	文件夹	
Projects	2015/5/15 14:27	文件夹	
Support	2015/5/15 14:27	文件夹	

4、定制开发板

4.4、获取demo源代码

2、使用IAR打开\Projects\STM32F4xx\Demo.eww工程环境，有两个子工程：Bootloader和COM.MXCHIP.MicoKit

工程配置选择COM.MXCHIP.MicoKit-MiCOKit-3288



4、定制开发板

4.5、云端创建新产品

1、登录FogCloud，进入产品管理，创建一个新产品；



EasyCloud API Wiki

创建新产品

产品管理 APP管理

产品管理

搜索 product name, serial, desc 产品 - 选择产品 - 状态 - 选择状态 - + 创建

 编辑	Id	EasyCloud_demo
	Name	EasyCloud_demo
	Serial	standard (3个月前)
	Status	deployed
	Description	EasyCloud standard demo
	Activated / Total	1 / 2
		50%
 编辑	Id	Wechat spp demo
	Name	Wechat spp demo
	Serial	EMW3162 (1个月前)
	Status	deployed
	Description	wechat spp demo
	Activated / Total	0 / 2
		0%

4、定制开发板

4.5、云端创建新产品

2、填写信息，创建新的产品；

The screenshot shows a web browser window with the URL `www.easylink.io/#/products/create`. The page title is "Product" and the form is titled "create product". The form contains the following fields and controls:

- 名称** (Name): Input field containing "test_product". A red arrow points to this field with the label "产品名称" (Product Name).
- 型号** (Model): Input field containing "EMW3288". A red arrow points to this field with the label "产品型号" (Product Model).
- 描述** (Description): Input field containing "EMW3288 test". A red arrow points to this field with the label "产品描述" (Product Description).
- 云端转码?** (Cloud transcoding?): A checkbox that is currently unchecked. A red arrow points to it with the label "是否需要云端转码功能" (Do you need cloud transcoding function?).
- 创建** (Create): A blue button at the bottom left, circled in red. A red arrow points to it with the label "开始创建" (Start creating).

4、定制开发板

4.5、云端创建新产品

3、获取产品ID/KEY;

The screenshot shows the product detail page for a product with ID 9715ee31 and serial EMW3288. The product name is 'test_product' and its status is 'deployed'. The 'secret' value is '89bcc84c-1791-4fe4-bfba-5c8252c74849'. The page includes a table of attributes and a '+ 创建' button.

名字	类型	读写属性	属性名	描述	时间序列	操作
+ 创建						

设备列表

搜索 导出 同步到微信 + 创建

4、定制开发板

4.6、激活设备到新产品

1、将新创建的产品ID和KEY写入固件代码，修改配置版本号以清除FLASH参数，

The screenshot displays the IAR Embedded Workbench IDE interface. On the left, the 'Files' pane shows the project structure for 'MiCOKit-3288', with 'user_config.h' highlighted under the 'User' directory. A red arrow points to this file with the label '用户配置文件'. The main editor window shows the content of 'user_config.h' with several lines of code. Red circles and arrows highlight specific values: '9715ee31' is circled with an arrow pointing to the text '替换你的产品ID'; 'e9bcc84c-1791-4fe4-bfba-5c8252c74049' is circled with an arrow pointing to '替换你的产品KEY'; and '0x00000002' is circled with an arrow pointing to '修改配置号（与上次不同即可），以清除FLASH中的配置'. The code includes definitions for product type, ID, key, ROM version, device name, manufacturer, configuration mode, cloud type, stack sizes, and notification intervals. The configuration version is defined as 0x00000002, with a comment indicating it should be updated if the default configuration is changed.

```
/* product type */
#define PRODUCT_ID
#define PRODUCT_KEY

#define DEFAULT_ROM_VERSION "v1.0.0"
#define DEFAULT_DEVICE_NAME "MicoKit3288" // device name upload to cloud
#define DEFAULT_MANUFACTURER "MXCHIP" // device manufacturer

/* =====
 * CONNECTING
 * =====
*/
/* Wi-Fi configuration mode */
#define MICO_CONFIG_MODE CONFIG_MODE_EASYLINK_WIIFI_SOFTAP

/* MICO cloud service type */
#define MICO_CLOUD_TYPE CLOUD_FOGCLOUD

/* =====
 * RESOURCES
 * =====
*/
#define STACK_SIZE_USER_MAIN_THREAD 0x800
#define STACK_SIZE_NOTIFY_THREAD 0x800
#define MICO_PROPERTIES_NOTIFY_INTERVAL_MS 1000

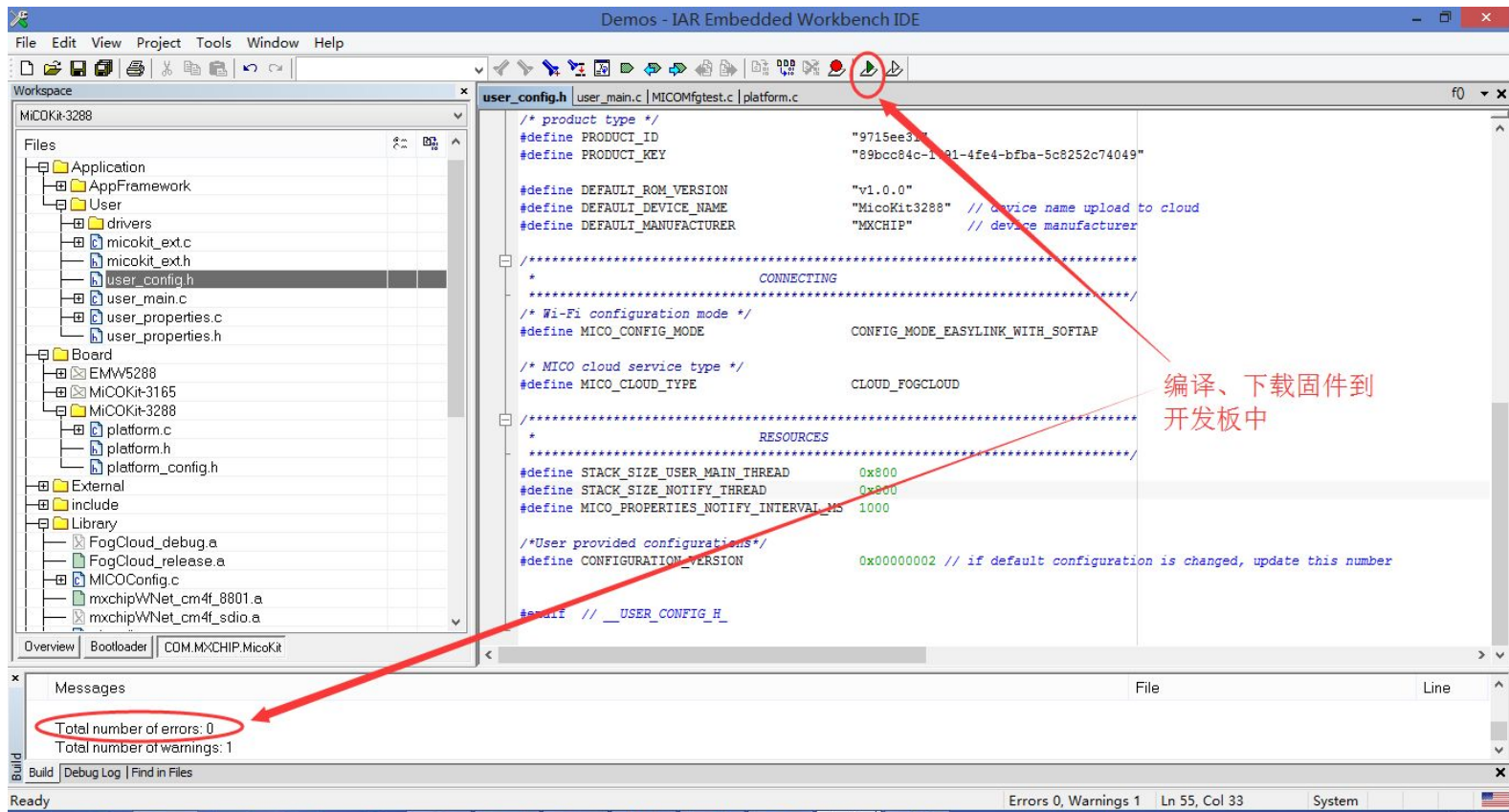
/*User provided configurations*/
#define CONFIGURATION_VERSION 0x00000002 // if default configuration is changed, update this number

#endif // __USER_CONFIG_H
```

4、定制开发板

4.6、激活设备到新产品

2、编译下载固件到开发板中。

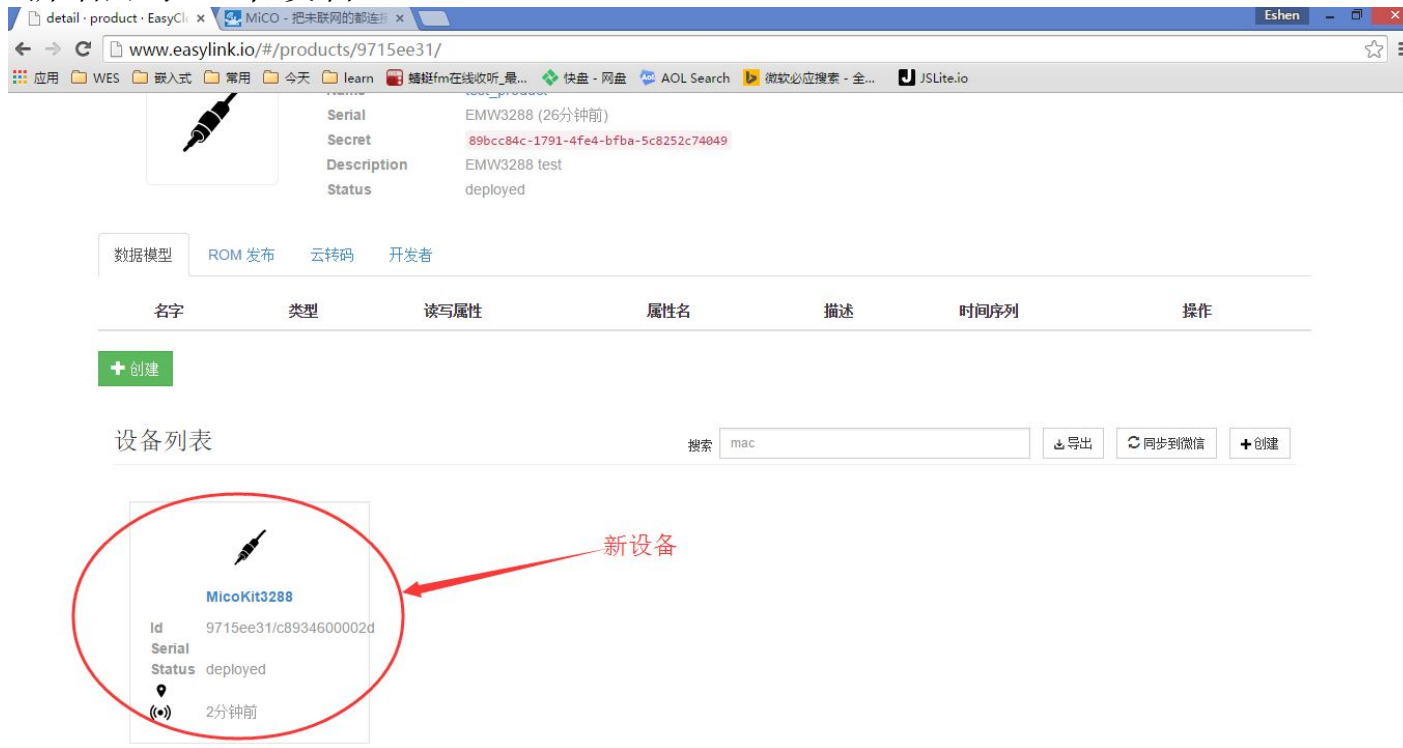


4、定制开发板

4.6、激活设备到新产品

3、重启开发板，重新配置Wi-Fi网络、激活设备，即可使用手机APP控制开发板；

4、进入FogCloud开发者网站(www.easylink.io)，进入刚创建的产品管理，可以看到该产品中新增加了一个设备。



The screenshot shows the FogCloud developer website interface. At the top, there is a browser window with the URL www.easylink.io/#/products/9715ee31/. Below the browser window, there is a product detail view for 'MicoKit3288'. The product details include:

- Serial: EMW3288 (26分钟前)
- Secret: 89bcc84c-1791-4fe4-bfba-5c8252c74049
- Description: EMW3288 test
- Status: deployed

Below the product details, there are tabs for '数据模型', 'ROM 发布', '云转码', and '开发者'. A table with columns '名字', '类型', '读写属性', '属性名', '描述', '时间序列', and '操作' is visible. A green '+ 创建' button is present. Below the table, there is a '设备列表' section with a search bar containing 'mac', and buttons for '导出', '同步到微信', and '+ 创建'. A red circle highlights a device card in the '设备列表' section, with a red arrow pointing to it from the text '新设备'. The device card shows:

- Icon: MicoKit3288
- Id: 9715ee31/c8934600002d
- Serial
- Status: deployed
- Location: 2分钟前

4、定制开发板

4.7、设备添加新模块

在user_properties.h/user_properties.c中添加新模块。

1、添加属性值变量到user_context中；

The screenshot shows the IAR Embedded Workbench IDE interface. The workspace tree on the left displays the project structure for 'COM.MXCHIP.MiCoKit - MiCoKit-3288'. The 'User' folder contains several files, with 'user_properties.h' highlighted by a red circle and an arrow pointing to it with the text 'user property 配置文件:'. The main editor window shows the contents of 'user_properties.h', which includes the following code:

```
int rgb_led_hues;
int rgb_led_saturation;
int rgb_led_brightness;

// DC Motor
int dc_motor_switch; // 0: off; others: on, for later use

}user_config_t;

// user module status
typedef struct _user_status_t {
    bool user_config_need_update; // if set, user context config need to write back to flash.

// light sensor (ADC1_4)
int light_sensor_data;

// infrared reflective sensor
int infrared_reflective_data;

// temperature && humidity
int temperature;
int humidity_saved; // save humidity value when get temperature, because we can not read DHT11 within 1 second.
int humidity; // real humidity prop value.

// uart
char uart_rx_buf[MAX_UART_BUFFER_SIZE]; // use a buffer to store data received
```

Red annotations include a circle around the line `// DC Motor` and an arrow pointing to the line `int dc_motor_switch; // 0: off; others: on, for later use` with the text '添加dc motor开关量'. The workspace tree also has a red circle around 'user_properties.h' with an arrow pointing to it and the text 'user property 配置文件:'.

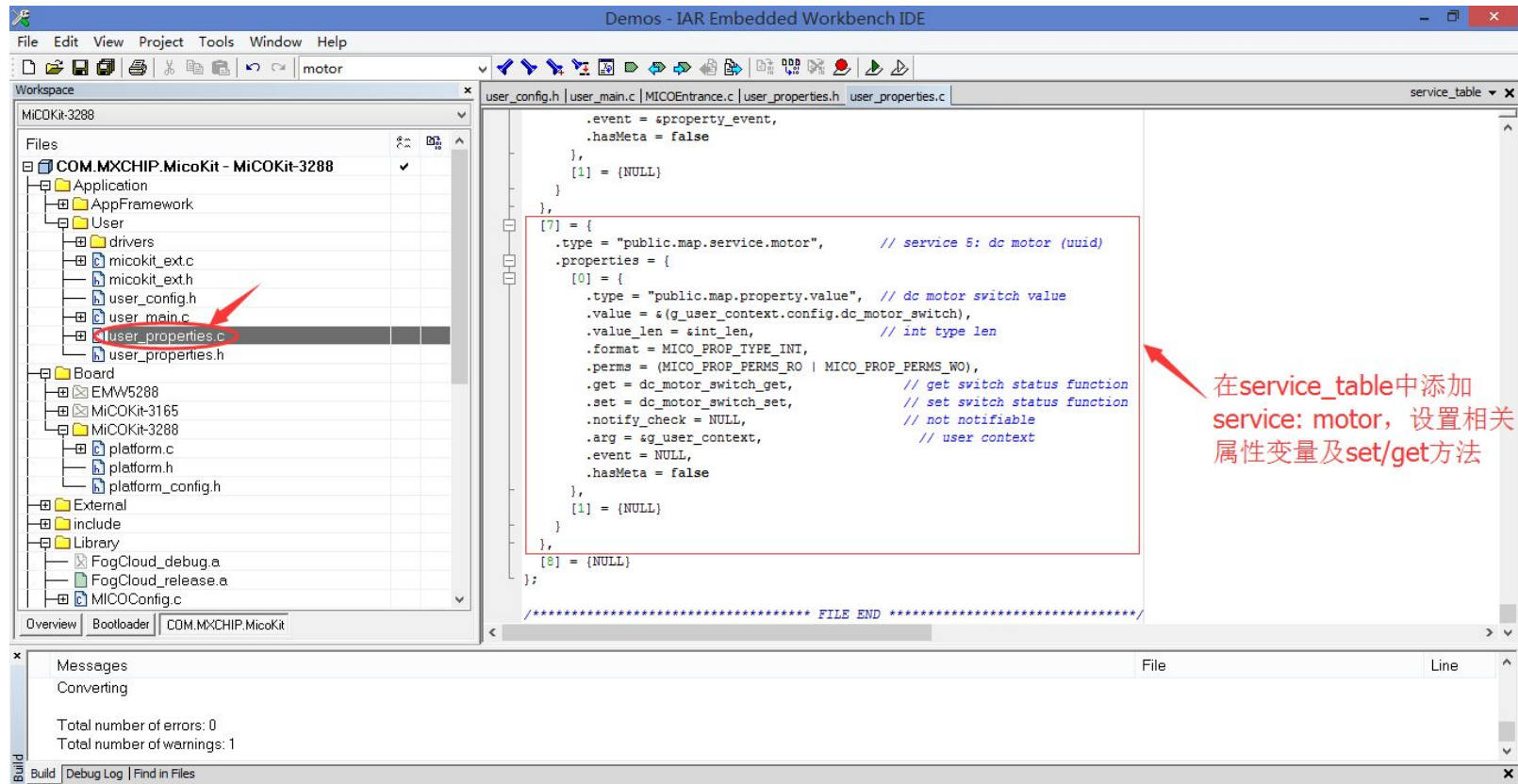
The Log window at the bottom shows the following output:

```
Log
Tue May 19, 2015 16:39:14: Hardware reset with strategy 0 was performed
Tue May 19, 2015 16:39:14: Initial reset was performed
Tue May 19, 2015 16:39:14: 1248 bytes downloaded (15.63 Kbytes/sec)
Tue May 19, 2015 16:39:14: Loaded debugee: C:\Program Files (x86)\IAR Systems\Embedded Workbench 7.0\arm\config\flashloader\ST\FIashSTM32F4xxxRAM128K.out
Tue May 19, 2015 16:39:14: Target reset
Tue May 19, 2015 16:39:29: Unloaded macro file: C:\Program Files (x86)\IAR Systems\Embedded Workbench 7.0\arm\config\flashloader\ST\FIashSTM32F4xxx.mac
Tue May 19, 2015 16:39:29: Downloaded D:\MXCHIP\MICO\MiCoKit\MiCoKit3288\SDK\MiCoKit_Firmware_SDK_v1.0.0\Project\Build\COM.MXCHIP.MiCoKit\Output\MiCoKit-3288\COM.MXCHIP.MiCoKit.out to flash memory.
```

4、定制开发板

4.7、设备添加新模块

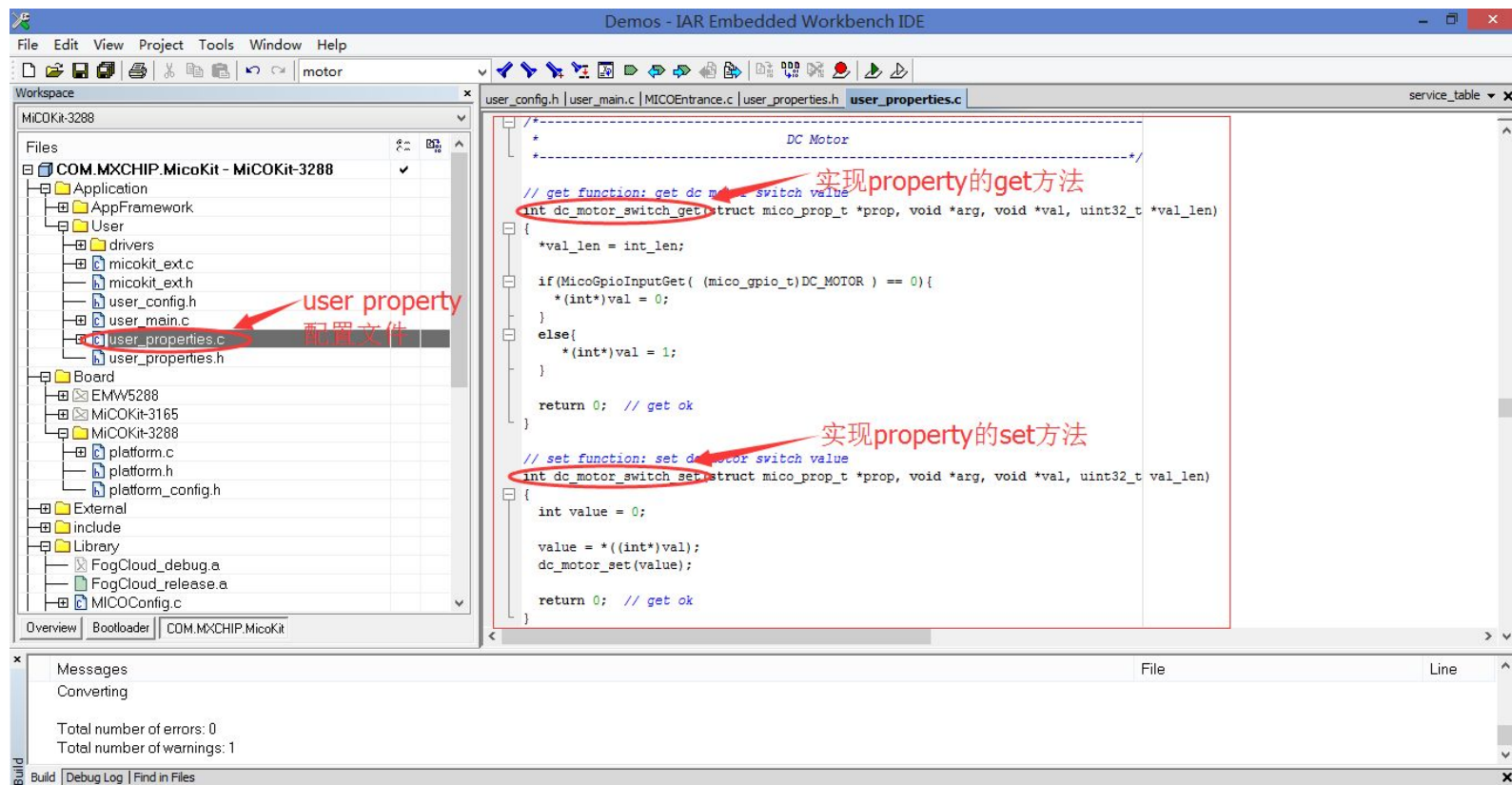
2、添加模块service及属性操作方法到service_table中。



4、定制开发板

4.7、设备添加新模块

3、实现属性的set/get方法



重新编译、下载固件，并重启设备即可。

4、定制开发板

4.8、手机APP访问新模块

固件添加完新模块后，APP无需修改，可直接访问新模块。

- 1、重启设备，连接上FogCloud后，APP进入控制页面，看到最低端出现了Motor的控制项；
- 2、点击开关，可以开关电机。



5、FAQ

5、FAQ

Q: 如何将设备从某一产品中删除（注销设备）？

A: 长按底板上的**EasyLink**按键，**5s**后设备会清除所有参数设置，然后重启进入**EasyLink**模式；重新配置**Wi-Fi**网络后，设备会先向云端请求**reset**设备(从云端注销该设备)，成功后再次重启，然后连接**Wi-Fi**后等待重新激活。此时该设备即为一个全新的设备，不属于任何产品。

Q: 如何多人控制一个设备？

A: **MicoKit APP**上点击左上角的菜单按钮，进入用户界面，点击“**Authorize**”，根据提示将设备给其他**APP**用户。