

纸上得来终觉浅 绝知此事要躬行

目录视图

摘要视图

RSS 订阅

个人资料



ydogg

关注

发私信



访问：154936次

积分：1848

等级： 4

排名：千里之外

原创：37篇

转载：4篇

译文：0篇

评论：71条

文章搜索

文章分类

ESP8266 (10)

Arduino (5)

OpenWrt (1)

OSS (1)

ACE (3)

ICE (3)

UML (4)

杂项 (3)

示例代码 (3)

转载 (4)

生活 (5)

最新评论

TP-Link 703N刷OpenWrt的实践
ydogg : @qq_15642501:按照如下方式reset 703N后, 重新设置Wlan1)电源重启, 在蓝灯开...

TP-Link 703N刷OpenWrt的实践
qq_15642501 : 通过PC, telnet 192.168.1.1后, 若可连接. 则大功成半, 若不连接怎么办? 路由不分配!...

TP-Link 703N刷OpenWrt的实践
qq_15642501 : 你好, 通过PC, telnet 192.168.1.1后, 若可连接. 则大功成半, 若不连接怎么办? 路由不...

异步赠书：Kotlin领书10本好书 免费直播：AI时代，机器学习如何入门？ 程序员8月书讯 项目管理+代码托管+文档协作，开发更流畅

9. ESP8266使用esp-open-rtos的实践

标签：esp8266 open-rtos 固件

2017-07-15 23:42

510人阅读

评论(0)

收藏

举报

分类：

ESP8266 (9)

版权声明：本文为博主ydogg原创文章，未经博主ydogg允许不得转载。

目录(?)

[+]

一、esp-open-rtos概要

Esp-open-rtos是开源的、基于freertos的、用于ESP8266固件的开发支持环境。最早是从乐鑫官方的ESP8266_RTOS_SDK基础上fork而来，但是目前的代码、框架已经和原SDK相去甚远，有了大幅的变化。从代码结构、编译方法、支撑库等均有不同，而且内置了更多的常见传感器驱动，以及MQTT、TLS、甚至OTA等常用的功能模块。相比官方同样的RTOS SDK、感觉更好用、更便利、开发应用速度更快。

目前，esp-open-rtos在github上获得了550多个STAR，足以证明其被认可的程度。

Github地址：<https://github.com/SuperHouse/esp-open-rtos>

Esp8266的项目较多，容易混淆，总结如下：

- 1) esp-open-sdk esp8266编译工具链
- 2) esp-open-rtos 基于FreeRTOS的esp8266固件
- 3) ESP8266_NONOS_SDK 乐鑫官方非OS版本固件、带AT固件
- 4) ESP8266_RTOS_SDK 乐鑫官方RTOS固件（基于FreeRTOS）
- 5) nodemcu-firmware NodeMCU的lua固件
- 2~5均使用同样的编译工具链1。

二、esp-open-rtos安装

1. 编译工具链的建立 (toolchain)

交叉编译工具链和乐鑫官方SDK相同，均使用esp-open-sdk。

安装方法可参照前面文章：【ESP8266固件的编译1（交叉编译工具链的建立）】。

简便起见，推荐使用【make STANDALONE=y】。

2. 代码下载

TP-Link 703N刷OpenWrt的实践

sinat_21438747 : 楼主,我按照你说的,恢复出厂设置,然后本地设置 "192.168.1.10 255.255.255.255..."

ACE Message Block功能和实现浅析

CalmReason : 经典!!

TP-Link 703N刷OpenWrt的实践

ydogg : 路由上一般都能看到无线设备的连接状况,看到对应的IP,依次SSH一下即可。

TP-Link 703N刷OpenWrt的实践

神经者漫游 : 楼主,我用你的方法设置后,电脑是可以通过703连接到家里的无线路由,可是703的ip是什么?无法te...

编译ICE

热爱生活 : 学习了!我是热爱生活。

Ice笔记 - 利用Ice::Service类简化Ice应用

mo_guang : Service类中的interrupt()的参数列表为什么是空的?在这方法中怎么判断接收到的是哪一种...

Ice笔记 - 利用Ice::Application类简化Ice...

BadGanker : 好东西!谢谢楼主

评论排行

路考步骤、注意点	(49)
TP-Link 703N刷OpenWrt的...	(6)
漫长的学车历程胜利结束,总...	(5)
ACE Message Block功能和...	(3)
Ice笔记 - 利用Ice::Service类...	(2)
ACE Dev Poll Reactor示例	(2)
Ice笔记 - 利用Ice::Applicatio...	(1)
场地九项 - 坡道起步,限宽门...	(1)
编译ICE	(1)
夜考上下车步骤	(1)

推荐文章

- * CSDN日报20170828——《4个方法快速打造你的阅读清单》
- * Android检查更新下载安装
- * 动手打造史上最简单的 Recycleview 侧滑菜单
- * TCP网络通讯如何解决分包粘包问题
- * 程序员的八重境界
- * 四大线程池详解

如下命令从github上下载代码。

```
git clone --recursive https://github.com/Superhouse/esp-open-rtos.git
```

使用recursive,可以将依赖的第三方代码一起获取。

三、代码编译

乐鑫官方的RTOS SDK是先编译底层支撑代码,以库的形式提供。

应用层通过连接库的方式和底层代码链接在一起,build为bin,存在两个阶段。

具体可参照前文:

【ESP8266固件的编译5 (RTOS SDK固件)】

但eps-open-rtos采用了另外一种方式,它没有底层代码编译库的过程。

而是以应用为中心,每个应用都会将所有底层代码复制一份

(从FreeRTOS到lwip,再到各种依赖库和应用层代码)每个应用都会复制并编译属于自己的这份,

并和自身的应用层代码一起编译。(其实最终也会通过库的形式,但是编译过程是一体化的)

另外Esp-open-rtos的编译采用make,摆脱了cmake,感觉清爽、瘦身很多。

编译时可通过make help来看到编译相关的帮助,尤其是make flash。

非常重要,

Esp-open-rtos在examples目录下提供了很多例子(很赞),上手可以很快。

1、编译准备

在编译之前,需要在include目录下,创建private_ssid_config.h文件,

来预定义系统默认wifi参数,内容如下:

```
#define WIFI_SSID "mywifissid"#  
  
define WIFI_PASS "my secret password"
```

2. mqtt_client的编译

上面已经说明过,esp-open-rtos以应用为中心编译,

下面以mqtt_client为例进行说明

编译方法1,在esp-open-rtos根目录下,执行

```
make -C examples/mqtt_client/
```

编译方法2,进入到examples/mqtt_client目录下,直接执行

```
make
```

3. esp8266的相关参数设定

系统默认的esp8266相关的参数,如SPI类型、Flash大小等,定义在根目录下的parameters.mk

中,如下:

```
1 FLASH_SIZE ?= 16  
2 FLASH_MODE ?= qio  
3 FLASH_SPEED ?= 40
```

关闭

?= 是make的语法，意为如果没有定义，就使用=后的值做赋值，相当于默认参数。

如果目标硬件(ESP8266模组)固定，可以直接修改此文件。

若目标硬件经常变化，如经常切换ESP01和ESP12，也可以修改应用目录下的Makefile文件进行配置：

```
1 PROGRAM=mqtt_client
2 FLASH_SIZE ?= 32
3 FLASH_MODE ?= qio
4
5 EXTRA_COMPONENTS = extras/paho_mqtt_c
6 include ../../common.mk
```

因使用安信可的ESP12F模块，所以将FLASH_SIZE修改为32，

如使用nodeMCU，除FLASH_SIZE外，还要将FLASH_MODE改为dio。

注意EXTRA_COMPONENTS宏，是esp-open-rtos的Make体系中指明依赖组件的宏，

所有需要依赖的扩展都要定义在该宏中（空格分隔）

这里可以看出，mqtt的协议是使用的Apache Paho(根据esp8266移植过)

MQTT本身的参数设定，可在在mqtt_client.c中修改。

编译成功后，在examples/mqtt_client/ firmware目录下，

会生成名为mqtt_client.bin的固件文件。

4. mqtt_client代码解析

mqtt_client的代码简单且结构清晰：

主函数启动3个task，分别是：

- wifi管理task
- mqtt管理task
- 以及mqtt连接成功后定时pub的测试task

wifi_alive是一个信号灯(semaphore)，用于协调wifi task和 mqtt task，以保证网络正常后再进行mqtt处理。

publish_queue是一个消息队列，用于beat_task和mqtt task间的数据传递，用于定时向【/beat】主题发布。

另外，mqtt task中在mqtt连接建立后，会订阅【esptopic】主题，并显示收到的消息。

下面是入口函数user_init的实现。

```
1 void user_init(void)
2 {
3     uart_set_baud(0, 115200);
4     printf("SDK version:%s\n", sdk_system_get_version());
5
6     vSemaphoreCreateBinary(wifi_alive);
7     publish_queue = xQueueCreate(3, PUB_MSG_LEN);
8     xTaskCreate(&wifi_task, "wifi_task", 256, NULL, 2, NULL);
9     xTaskCreate(&beat_task, "beat_task", 256, NULL, 3, NULL);
10    xTaskCreate(&mqtt_task, "mqtt_task", 1024, NULL, 4, NULL);
11 }
```

关闭

四、固件下载（刷写）

因为esp-open-rtos内置了esptool.py，可用make flash进行直接做固件刷新。

但在windows上，还是使用官方的下载工具更加方便。

关于固件下载工具的使用，可参照前文【ESP8266固件的编译5（RTOS SDK固件）】。

和其他固件的刷新方法一样，但写入地址方面，和乐鑫官方、nodemcu差别较大。

观察make flash的命令输出：

```
1 esptool.py -p /dev/ttyUSB0 --baud 115200 write_flash -fs 32m -fm qio .
2 0x0 ../../bootloader/firmware_prebuilt/rboot.bin 0x1000 ../../bootloa
```

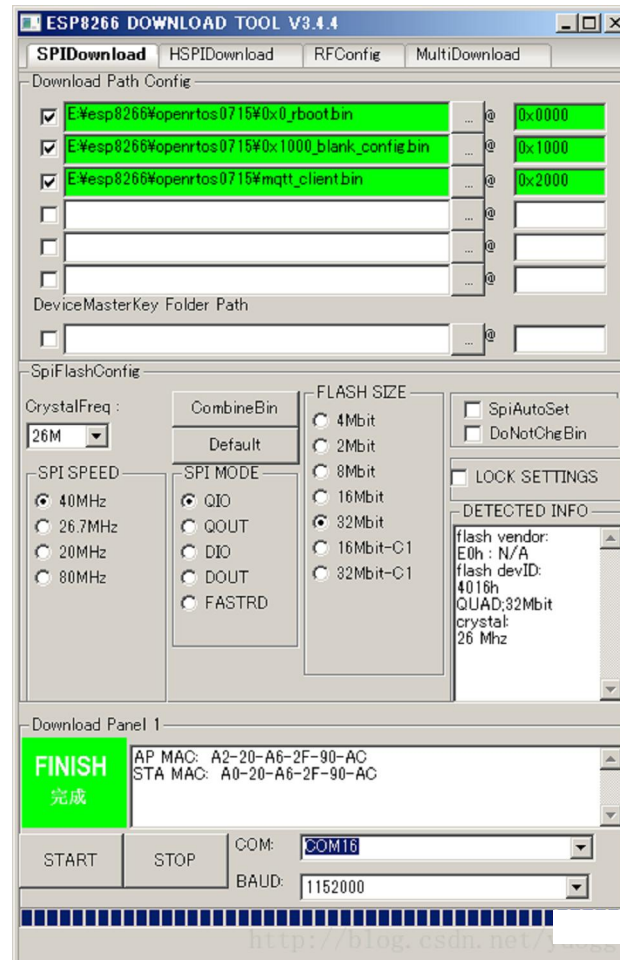
可以看到，出mqtt_client.bin外，还要刷入预置的，

存放于bootloader/firmware_prebuilt目录下rboot.bin和blank_config.bin。

其写入地址为：

```
1 rboot.bin          0x0000
2 blank_config.bin   0x1000
3 mqtt_client.bin    0x2000
```

如下图：



刷入后，如网络征程，即可以看到wifi连接成功后，mqtt的通信状况。

ESP8266侧如图：

关闭

```

ESP-Open-SDK ver: 0.0.1 compiled @ Jul 15 2017 16:05:09
phy ver: 273, pp ver: 8.3

SDK version:0.9.9
mode : sta(a0:20:a6:2f:90:ac)
add if0
mqtt_task: started
mqtt_task: (Re)connecting to MQTT server 192.168.31.1 ... error: -1
WiFi: connecting to WiFi
wifi_task: status = 1
scandone
wifi_task: status = 1
add 0
aid 3
cnt

connected with , channel 2
dhcp client start...
wifi_task: status = 1
wifi_task: status = 1
wifi_task: status = 1
ip:192.168.31.238,mask:255.255.255.0,gw:192.168.31.1
wifi_task: status = 5
WiFi: Connected
mqtt_task: started
mqtt_task: (Re)connecting to MQTT server 192.168.31.1 ... done
Send MQTT connect ... done
beat
got message to publish
beat
got message to publish
received: /esptopic = hello world
beat

```

<http://blog.csdn.net/ydogg>

框内内容分别是，wifi接入，mqtt接入，和收到mqtt消息。

服务端可用如下命令测试（mosquitto）：

发布：

```
mosquitto_pub -u test -P test -t /esptopic -m "hello world"
```

接收：

```
mosquitto_sub -u test -P test -t /beat
```

可以看到收到连续的beat消息。

```

[~]mosquitto_sub -u test -P test -t /beat
Beat 1160

Beat 1161
Beat 1162
Beat 1163
Beat 1164
Beat 1165

```

<http://blog.csdn.net/ydogg>

如上，mqtt在esp8266上非常容易的便可被实现，而且还有多task的支持。

增加传感器，将读取数据通过mqtt送出，即可实现简单的IoT系统。

而且，esp-open-rtos对常见传感器（温度、光线 etc.）的

examples目录下有例子，extras目录下有协议实现。

顶 踩
0 0

- [下一篇](#) [VIM实用技巧](#)

相关文章推荐

- [9. ESP8266使用esp-open-rtos的实践](#)
- [Hadoop生态系统零基础入门](#)
- [ESP8266 RTOS SDK开发 --- \(一\) Ubuntu下编译...](#)
- [系统集成工程师必过冲刺！](#)
- [ESP8266 RTOS SDK使用GPIO中断的方法](#)
- [征服React Native我有妙招](#)
- [ESP8266使用sniffer示例工程](#)
- [FFmpeg音视频高级开发实战](#)
- [使用安信可ESP8266IDE编辑工具使用最新的乐鑫...](#)
- [5天搞定深度学习框架-Caffe](#)
- [ESP8266 wifi模块使用教程，不同模式的使用方法](#)
- [Python数据分析经典案例解析](#)
- [ESP8266_RTOS_SDK编译烧写与运](#)
- [ESP8266使用官方Json接口的例程](#)
- [ESP8266使用小结](#)
- [WIFI 芯片ESP8266 开发视频教程（：](#)



sci论文润色



young 酒店



视频制作学习



北大青鸟学费一览



美容网站

查看评论

暂无评论


您还没有登录,请[\[登录\]](#)或[\[注册\]](#)

* 以上用户言论只代表其个人观点，不代表CSDN网站的观点或立场

[公司简介](#) | [招贤纳士](#) | [广告服务](#) | [联系方式](#) | [版权声明](#) | [法律顾问](#) | [问题报告](#) | [合作伙伴](#) | [论坛反馈](#)

[网站客服](#) [杂志客服](#) [微博客服](#) webmaster@csdn.net 400-660-0108 | 北京创新乐知信息技术有限公司 版权所有 | 江苏知之为计算机有限公司 |

江苏乐知网络技术有限公司

京 ICP 证 09002463 号 | Copyright © 1999-2017, CSDN.NET, All Rights Reserved 

关闭