
Spark karaoke SDK

Quick Start Guide



文档履历

版本号	日期	制/修订人	制/修订记录
V1.0	2022-05-16		初始版本



目录

Spark karaoke SDK	1
1. SDK 下载指南	4
2. SDK 目录结构	5
3. 编译环境搭建指南	6
1. 安装 Python	6
2. 安装 scons	7
4. 编译指南	7
5. 烧写指南	8
1. 开发板烧录	8
6. Declaration	9



1. SDK 下载指南

1. 安装 git bash. 路径为: SDK:spark_sdk\configs\develop_install\Git-2.18.0-64-bit.exe
2. 在 git bash 上生成 ssh key. 命令为: ssh-keygen -t rsa -b 2048 -C "email@example.com" (你的邮箱地址)

```
lichaojie@LAPTOP-VSJ5TKD8 MINGW64 /d/workspace/tem
$ ssh-keygen -t rsa -b 2048 -C "smart@example.com"
Generating public/private rsa key pair.
Enter file in which to save the key (/c/Users/lichaojie/.ssh/id_rsa):
```

提示输入密码, 全部为空, 按 enter 键即可。

```
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /c/Users/lichaojie/.ssh/id_rsa.
Your public key has been saved in /c/Users/lichaojie/.ssh/id_rsa.pub.
The key fingerprint is:
SHA256:J4lgK6F3S3MZ9ehuH4OeS5PtyFPSiE7EICH34FBpLMI smart@example.com
The key's randomart image is:
+---[RSA 2048]---+
|+O=O          |
|oE=o. . . o   |
|.OO.+O. . .   |
|. o oo= .     |
|. o =.+So.    |
|. + +O.O*O    |
|.O Bo+       |
|. =.* O       |
|*OO          |
+-----[SHA256]-----+
```

最后, 根据系统提示 key 生成的路径: c/Users/xx/.ssh 找到公钥 id_rsa.pub 文件。

3. 把公钥文件发给 SDK 管理员,待开通权限后,管理员会提供 sdk 下载的路径。
4. 根据管理员提供的路径用 git clone 指令下载代码:

```
$ git clone ssh://git@git.smartlinktech.com.cn:29418/Smartlink/Karaoke_sdk.git
Cloning into 'Karaoke_sdk'...
remote: Counting objects: 5395, done
remote: Finding sources: 100% (5395/5395)
remote: Getting sizes: 100% (2489/2489)
remote: Total 5395 (delta 2902), reused 5395 (delta 2902)
Receiving objects: 100% (5395/5395), 162.79 MiB | 1.36 MiB/s, done.
Resolving deltas: 100% (2902/2902), done.
Updating files: 100% (5827/5827), done.
```

如图, 显示下载完成, 如果 key 生成不对, 会提示鉴权失败, 无法下载。

5. 每次需要同步代码的时候, 使用命令: git pull
 6. 查看当前工作区修改情况, 使用命令: git status .
 7. 查看日志记录, 使用命令: git log .
 8. 切换新分支, 使用命令: git checkout -b xx
- 其他 git 操作, 请客户自行学习。



2. SDK 目录结构

apps	//应用层目录
karaoke	//产品类型
app_config	//应用相关配置文件
xx	//xx 方案型号
bin	//已编译二进制文件
eq_profile_customer.eq2	//eq 文件
tone.bin	//提示音文件
ota.bin	//ble ota 升级引导文件
tone	//提示音原文件
config	//用户配置及音频参数配置
dongle_board_audio_params_config.h	//音频参数
dongle_board_user_config.h	//用户配置
dongle_board_ui.c	//ui 控制实现代码
app_audio	//音频应用代码
app_ble	//ble 应用代码
app_device	//sd linein usb 等模块检测代码
app_disp	//灯光应用代码
app_exception	//运行异常应用代码
app_key	//按键应用代码
app_lcd	//lcd 应用代码
app_led	//led 应用代码
app_pmu	//pmu 应用代码
app_psm	//psm 应用代码
app_pwm	//pwm 应用代码
app_rtc	//rtc 应用代码
app_sensor	//外设驱动实现代码
app_spi	//spi 总线应用控制代码
app_timer	//硬件定时器应用代码
app_twi	//twi 总线应用控制代码
app_usb_device	//usb device 应用代码
console	//uart 应用控制代码
doc	//文档
gcc	//编译实现代码
tw.s.ld	//ld 文件
SConstruct.py	//app 层编译脚本
SConstruct	//sdk 编译控制脚本
CreateProject.py	//创建新方案实现脚本
makeup.json	//分区文件
dst	//编译生成产物文件
mode	//模式应用代码



		soundcard	//音频通路相关代码
		temp	//编译中间文件（不用修改）
		ui	//ui 控制代码实现
		main.c	//main 文件
		bootloader	//启动镜像目录
		bin	//bin 文件
		build	//编译规则，链接规则的脚本以及编译器
		toolchain	//编译工具链
		gcc.py	//gcc 编译控制脚本
		makecode.py	//up 生成脚本
		path_rule.py	//编译路径控制脚本
		config	//安装包，向导程序
		doc	//开发文档
		inc	//库实现头文件
		lib	//库文件
		tools	//固件烧录和调参工具，调试工具
		SetupFlashloaderSL4.13.exe	//固件下载及调参工具包
		otabox	//空中升级盒子固件包
		testbox	//测试盒固件包

3. 编译环境搭建指南

Spark SDK 使用的是 python 及 scons 配合，调用 gcc 编译工具链。需要安装 python 及 scons 这两个工具。默认的操作系统是 windows7 或以上，工具在 SDK:spark_sdk\configs\develop_install 路径下。

1. 安装 Python

安装 python-3.6.8-amd64.exe ,确保系统变量中有:

C:\Users\xx\AppData\Local\Programs\Python\Python36

C:\Users\xx\AppData\Local\Programs\Python\Python36\Scripts

C:\Windows\system32
C:\Windows
C:\Windows\System32\Wbem
C:\Windows\System32\WindowsPowerShell\v1.0\
C:\Windows\System32\OpenSSH\
C:\Program Files (x86)\NVIDIA Corporation\PhysX\Common
C:\Program Files (x86)\Git\cmd
C:\Program Files\TortoiseGit\bin
C:\work\cygwin64\bin\
C:\Program Files\dotnet\
C:\Users\asus\AppData\Local\Programs\Python\Python36
C:\Users\asus\AppData\Local\Programs\Python\Python36\Scripts



安装完后，在 window cmd 窗口输入 `python -v` 如果有显示 `python` 版本就证明 `python` 安装成功。

2. 安装 scons

把 `spark_sdk\configs\develop_install\Scons` 复制到本地，双击 `install.bat` 自动安装 `scons`。

4. 编译指南

进入 `\spark_sdk\apps\karaoke\gcc` 路径：

```
$ scons -c target=dongle_board
```

或

```
scons -c target=handset_board
```

可清除编译时产生的临时文件。

```
$ scons -s cmd=lunch
```

可获取目前 `sdk` 之前的方案目录。这个步骤只是列出方案列表，不影响编译过程，平时编译可直接跳过到下一步。

```
$ scons -s cmd=lunch
COMMAND_LINE_TARGETS is []
=====
project list:
dongle_board
handset_board
=====
```

```
$ scons -s target=xx(设备程序)
```

编译选择的设备程序，完成以后，会在 `\spark_sdk\apps\karaoke\gcc\dst\xx\` 下，生成镜像文件 `xx.up`，可用于固件升级。

```
$ scons -s target=handset_board
COMMAND_LINE_TARGETS is []
=====
APP_GCC_PATH: E:\test_code\spark_sdk\spark_sdk\apps\karaoke\gcc
ROOT_PATH: E:\test_code\spark_sdk\spark_sdk
APP_PATH: E:\test_code\spark_sdk\spark_sdk\apps\karaoke
BUILD_PATH: E:\test_code\spark_sdk\spark_sdk\build
PROJECT: karaoke
OUT_PATH: .out
=====
board_name: handset_board
```

备注：



如果使用 Smartlink 的开发板开发、调试，需要把首次接入电池关机功能关掉，具体如下：

spark_sdk\apps\karaoke\app_config\dongle_board\config\dongle_board_user_config.h 文件：

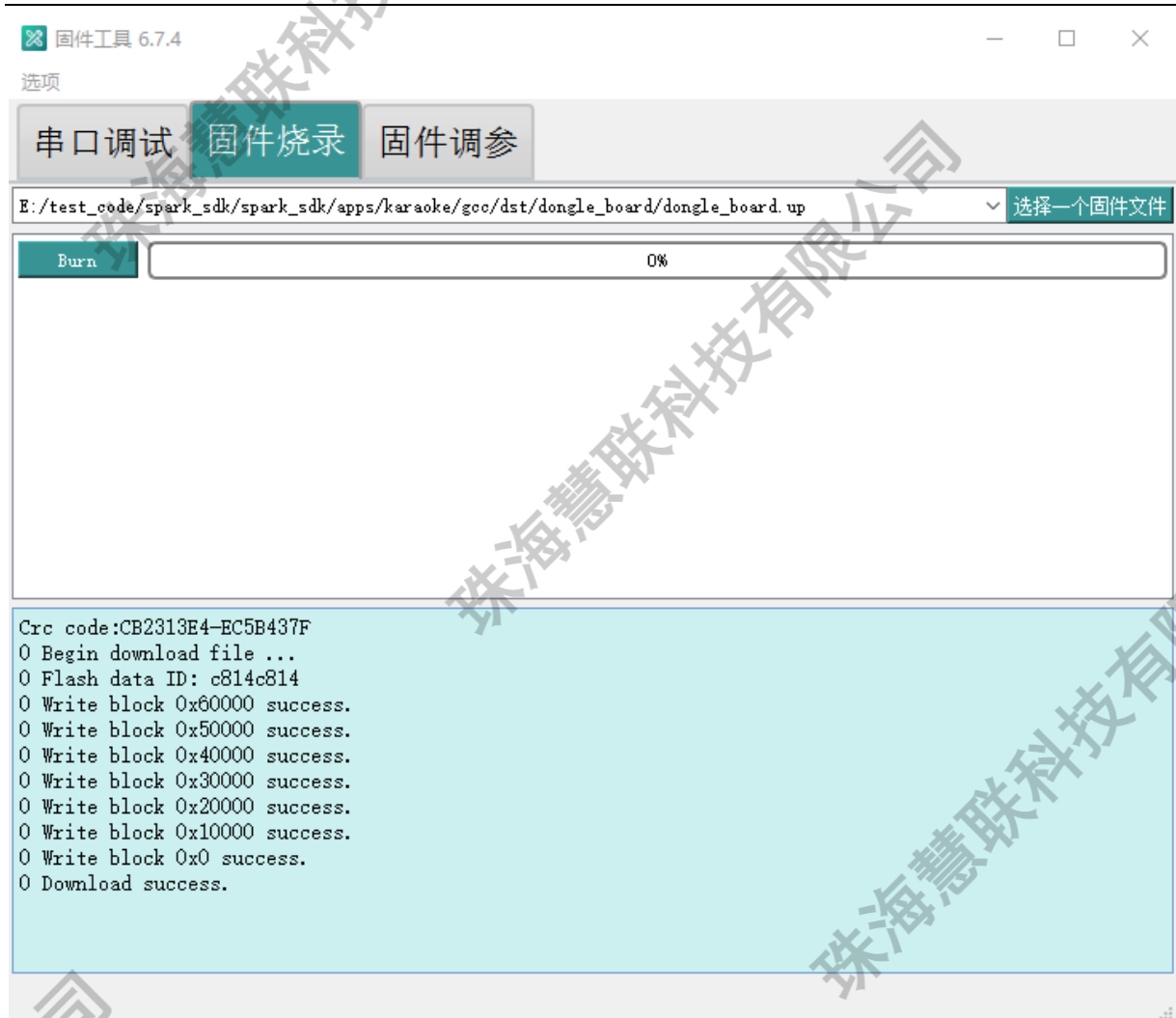
```
#define PMU_POWEROFF_WHEN_BAT_FIRST_IN    0    //使能第一次插入电池时关机
```

5. 烧写指南

1. 开发板烧录

1. 安装 spark_sdk\tools\SetupFlashloaderSL6.7.4.exe，开发工具包
2. 打开工具：固件工具 SL 研发模式，选择固件包。
3. 按住按键拉低 PD0，进入烧录模式。
4. 配对地址等参数会存在参数区，如需要擦除，需要先 选项--开发配置，勾选“仅擦除”烧录一遍，然后再取消勾选下载.up 固件





6. Declaration

This document is the original work and copyrighted property of SmartLink Technology ("SmartLink"). Reproduction in whole or in part must obtain the written approval of SmartLink

珠海慧联科技有限公司

Copyright © 2018 by Smartlink. All rights reserved

Page 9 of 10



and give clear acknowledgement to the copyright owner.

The information furnished by SmartLink is believed to be accurate and reliable. SmartLink reserves the right to make changes in circuit design and/or specifications at any time without notice. SmartLink does not assume any responsibility and liability for its use. Nor for any infringements of patents or other rights of the third parties which may result from its use. No license is granted by implication or otherwise under any patent or patent rights of SmartLink. This datasheet neither states nor implies warranty of any kind, including fitness for any particular application.