	Title: S70A_S907x_SDK_入门指南_FreeRTOS	
	Document No.:	SC2G017
	Version No.:	V1.3
	Effective Date:	2019-09-19
	Page:	1

S70A_S907x_SDK_入门指南_FreeRTOS

Document Revision: V1.3

Document Release: 2019/09/19

SmartChip Integration Inc.

9B, Science Plaza, International
Science Park, 1355 Jinjihu
Avenue, Suzhou Industrial Park,
Suzhou, Jiangsu, China.

ZIP: 215021

Telephone: +86-512-62620006

Fax: +86-512-62620002

E-mail: sales@sci-inc.com.cn

Website: <http://www.sci-inc.com.cn>

目 录

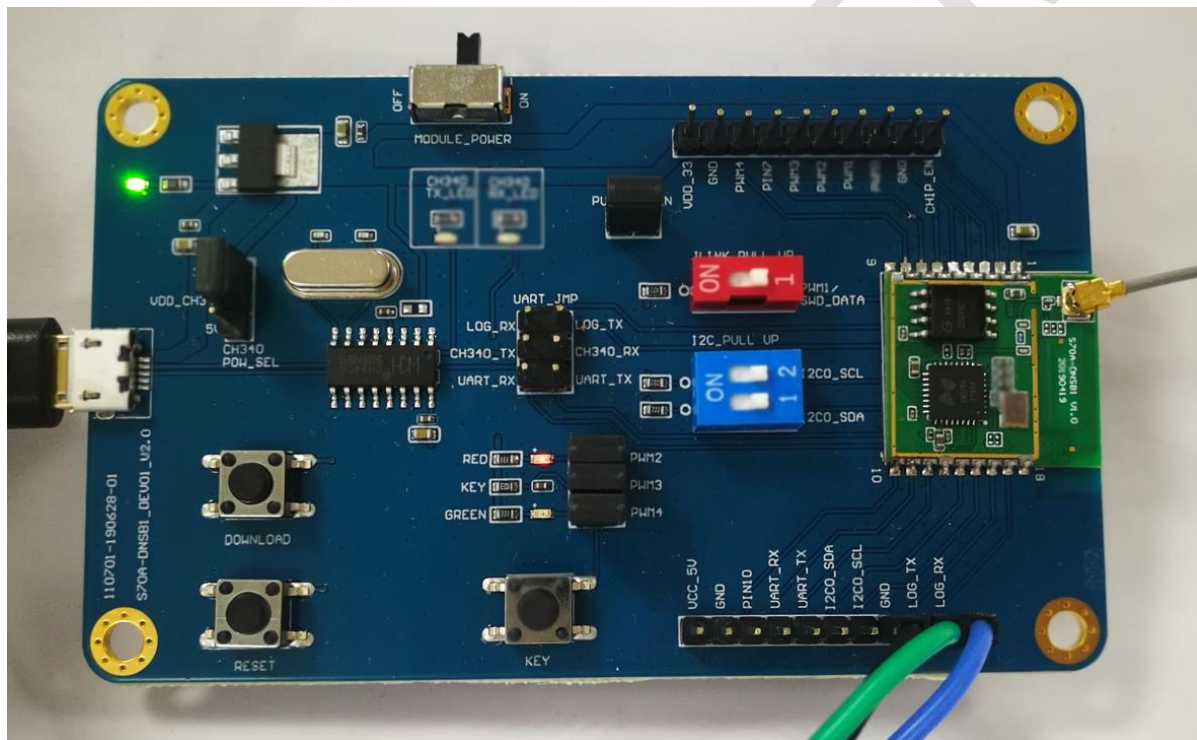
1 概述.....	3
2 硬件准备.....	3
3 软件准备.....	3
3.1 FreeRTOS SDK.....	4
3.2 S907x 工具集.....	5
3.2.1 SCI_Embedded_DebugTest	5
4 编译 SDK	6
4.1 编译准备	6
4.1.1 LWIP 协议栈配置	6
4.1.2 系统模块功能配置	6
4.1.3 系统 Debug 配置.....	7
4.2 开始编译	8
4.2.1 IAR 编译	8
4.2.2 KEIL 编译.....	8
4.2.3 命令行编译	8
4.2.4 VSCODE 编译	9
5 版本信息.....	12

1 概述

S907x SDK, 基于 FreeRTOS 系统开发, 使用 IAR/KEIL IDE, 可以实现代码的编译和仿真调试。SDK 代码中提供有 IAR/KEIL 的工程目录。具体工程目录是 s907x_sdk\example\main\project\

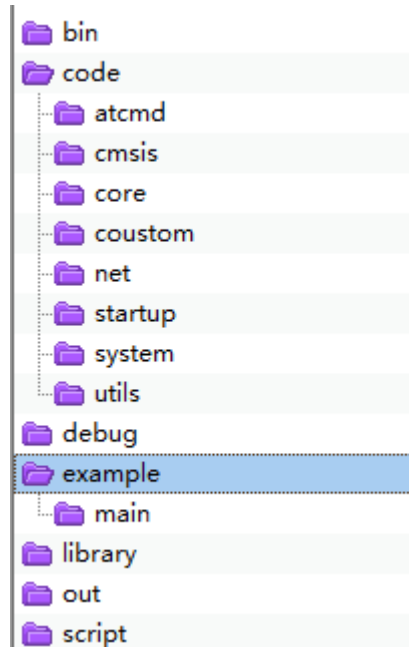
2 硬件准备

现阶段 S907x 提供开发板方案,



3 软件准备

3.1 FreeRTOS SDK



bin: SCI 发布的固件，包括 boot 和 wlan 驱动的 bin

code: SDK 的主目录，可供客户二次开发的代码

atcmd: AT 指令的代码实现。目前对接了 ESP8266 的 AT 指令，同时包含 SCI 原生的 AT 指令

cmsis: ARM 适配相关的实现

core: 包含 hal 的接口、FreeRTOS 系统源码、wlan 驱动的接口

coustom: 客户自己的应用代码包含 lib 库应该放在此目录下面。

net: 开源的网络功能代码，包括 DHCP, DHCP, LWIP 协议栈，TLS 功能代码

startup: 系统启动代码。

system: 系统配置的头文件。

utils: 包含一些 ota_hash 等相关文件。

debug: 此目录目前只对于 IAR 有关,IAR IDE 可以直接用 jlink 烧录仿真,别的暂时不支持这个功能

example: 工程目录。

library: SCI 发布的库文件，包括 s907x_net.a (sci 自行实现的网络相关功能)，s907x_platform.a。

out: IAR/KEIL 编译后的文件目录, **s907x_image.bin** 是合成 bin(包含 boot,ota1,wlan driver),可以通过 SCI_Embedded_DebugTest 进行烧录,**s907x_ota1.bin** 是提供给客户进行 OTA 的 bin 文件, **s907x_ota1_wlan.bin** 也是 OTA 的 bin 文件但是与 s907x_ota1.bin 的区别是包含了 wlan_driver.bin, OTA 的时候会同时更新 ota1 和 wlan_driver。

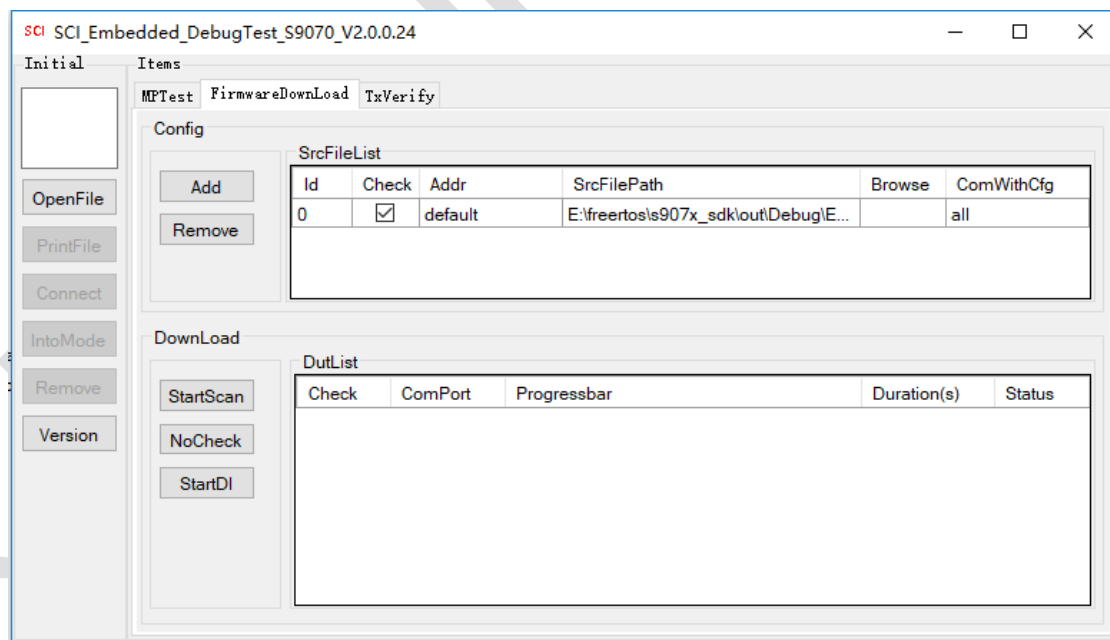
script: IAR 编译相关的脚本文件

3.2 S907x 工具集

本地安装 IAR 的 IDE, 用于 SDK 的编译, 以及仿真调试。KEIL IDEL 可以编译并生成相应的 bin 文件 (目前不支持仿真调试)。s907x SDK 后续会更新基于 GCC 的编译环境。

3.2.1 SCI_Embedded_DebugTest

版本烧录工具, 可以实现完整的单固件, 或者 Boot、Wlan、Ota 固件的烧录。



详细烧录说明请参考文档 **SC2D016-S70A-DNSB1_DEV01+开发板使用指南+V2.0**

4 编译 SDK

4.1 编译准备

4.1.1 LWIP 协议栈配置

LWIP 协议栈的配置文件，参考 `code/net/lwipopts.h`。其中包含 LWIP 具体功能的开关，如果没有特殊配置需求，保持默认即可。

备注： `code/net/lwip` 中的 `opt.h` 的参数配置会被该 `lwipopts.h` 中参数覆盖。

4.1.2 系统模块功能配置

系统级的配置参数，以及相关的模块开关，参考 `code/net/system/system_config.h` 文件。

主要配置代码如下：

```
//AT commad config

#define M_AT_ENABLE            1

#if M_AT_ENABLE == 1

#define M_AT_ESP              0

#define M_AT_TEST              1

//AT ESP config

#if M_AT_ESP && M_AT_TEST

#error "close M_AT_TEST if you want to open M_AT_ESP"

#endif

#endif

//low power config

#define S907X_LOW_POWER        1
```



```
//heap ram end config

#define HEAP_DATA_END      0x10046FFF

//LWIP config

#define CONFIG_HIGH_TP      0

//LWIP select

#define LWIP_TUYA           0 //1.4.1

#define LWIP_SCI            1 //2.0.3

#define LWIP_VERSION        LWIP_TUYA
```

宏配置说明:

M_AT_ENABLE 如果配置为 0,这种情况 s907x SDK 不会支持任何 AT 指令解析, 适用于客户开发自己的方案。

M_AT_ESP 打开这个宏的前提是 M_AT_ENABLE 开启,这个宏将 SDK 配置为一套标准的 AT 指令(效仿 esp8266),具体差异可以参考差异文档说明。

M_AT_TEST 打开这个宏的前提是 M_AT_ENABLE 开启,这个宏将 SDK 配置成测试模式,用户可以通过测试命令测试 S9070 的各个功能包含 wlan,外设,ping,iperf....等。详细指令说明可以参考文档。

S907X_LOW_POWER 低功耗配置宏开关, 1 使能系统低功耗模式,S9070A 低功耗是由操作性自动管理低功耗可以在尽可能低的情况下保持与路由器连接。0 是关闭低功耗模式

CONFIG_HIGH_TP 主要是配置 lwip 的性能,1 使能 lwip 高吞吐量但是会有较大的内存消耗,默认 0 关闭。

LWIP_VERSION lwip 版本选择, 默认使用 1.4.1 客户也可以使用 2.0.3 需要重新编译相关文件,详细联系 sci 技术人员。

4.1.3 系统 Debug 配置

如果有 SDK 调试需求,可以开启 Debug 功能,输出更多的 Log 信息来进行调试。可以参考 code/net/system/system_debug.h 文件。



```
//config debug enable
#define HALTEST_DEBUG_EN    1
#define HTTPC_DEBUG_EN     1
#define PING_DEBUG_EN      1
#define AT_CMD_DEBUG       0
```

4.2 开始编译

4.2.1 IAR 编译

在 sdk 目录下的 `example/main/project/IAR` 目录下，使用 IAR IDE 加载打开 `S907x.eww` 后，直接进行编译即可。默认的 IAR 配置，已经配置完成，并包含在 `example/main` 的 IAR 工程目录下。点击 **Project->Rebuild all** 完成编译,输出目录在 `out/IAR/Debug/Exe/s907x_image.bin`

4.2.2 KEIL 编译

在 sdk 目录下的 `example/main/project/KEIL` 目录下，使用 KEIL IDE 加载打开 `s907x_freertos_keil.uvprojx` 后，直接进行编译即可。默认的 KEIL 配置，已经配置完成，并包含在 `example/main` 的 KEIL 工程目录下。点击 **Project->Rebuild all target files** 完成编译,输出目录在 `out/KEIL/Debug/Exe/s907x_image.bin`

4.2.3 命令行编译

在 sdk 根目录下面，直接调用脚本即可完成编译过程。GCC 编译支持 window 和 Linux 两种不同的宿主环境。

Windows 环境下，调用根目录下脚本 `gcc_build.bat` 进行编译；Linux 环境下，调用根目录下的 `gcc_linux.sh` 进行编译。

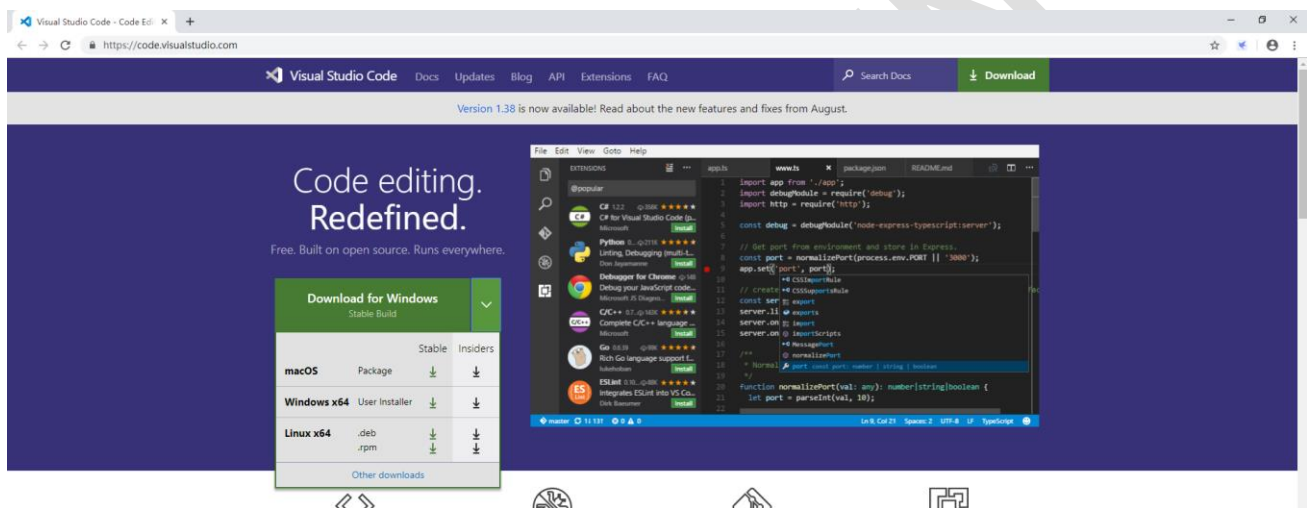
编译完成后输出目录在 `out/GCC/Debug/Exe/s907x_image.bin`。

备注: 在第一次编译之前, 建议先 clean。比如在 windows 环境下, 先调用“`gcc_build.bat clean`”来删除之前编译生成的中间文件, 再调用“`gcc_build.bat`”重新进行编译并生成固件。

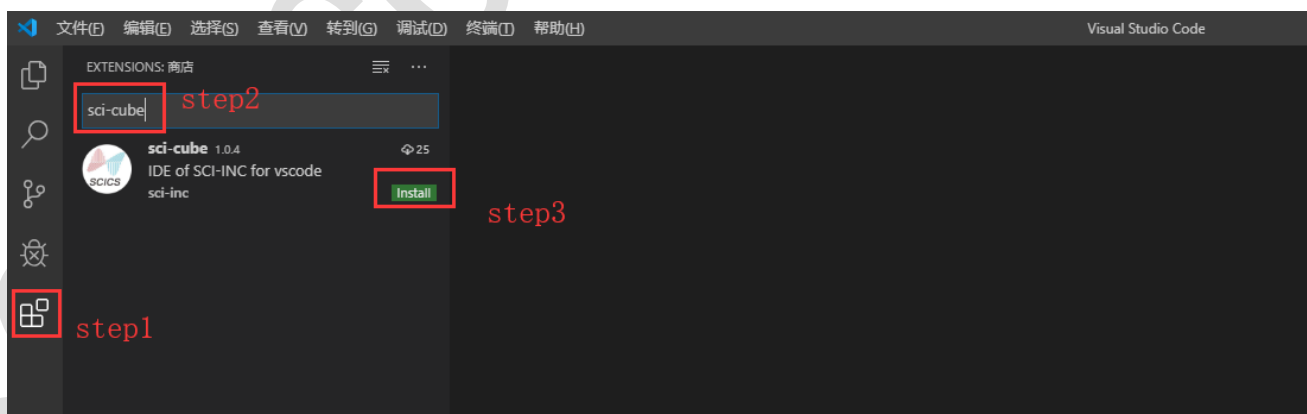
4.2.4 VSCODE 编译

1、安装 VSCODE

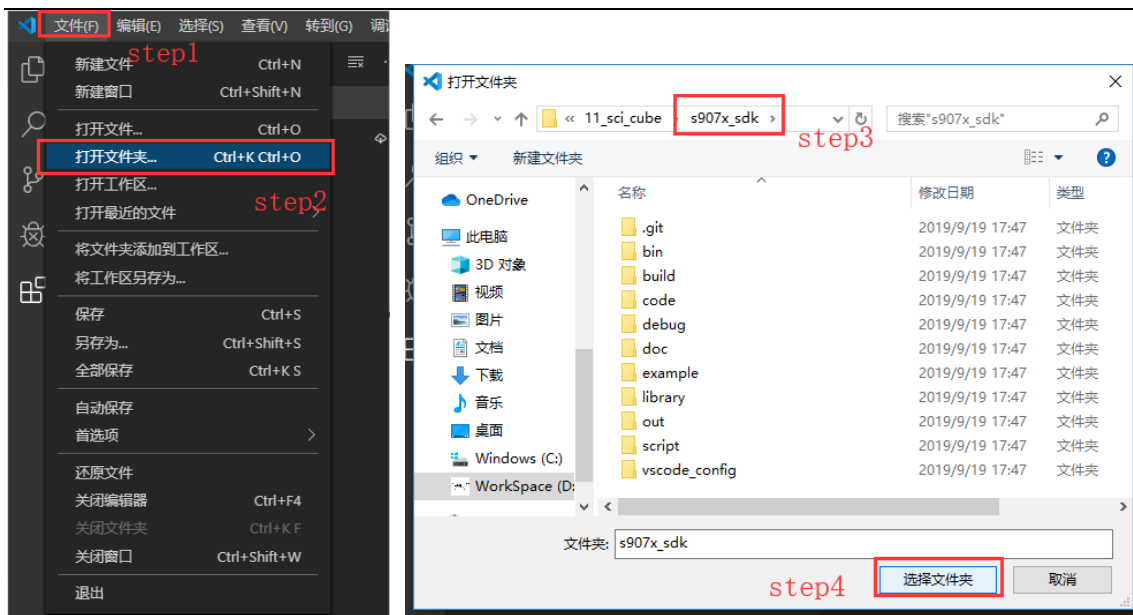
访问 VSCODE 官网, 进入相关页面进行下载合适的安装包 <https://code.visualstudio.com/>



2、在 VSCODE 中安装 sci-cube 插件



3、利用 VSCODE 打开源码



4、build、clean

方式 1：快捷键

如下图：build 快捷键为 ctrl+F5；clean 的快捷键为 ctrl+F4

命令(2)		
名称	说明	键盘快捷方式
sci_inc.onBuild	build	Ctrl + F5
sci_inc.onClean	clean	Ctrl + F4

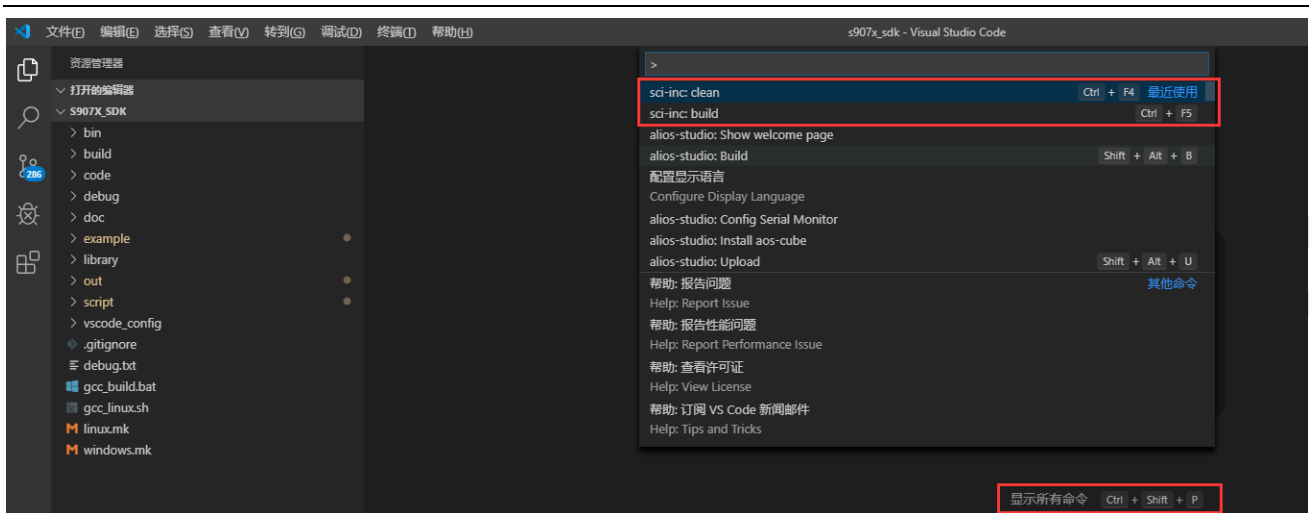
方式 2：状态栏按钮

点击如下图的两个按钮即可



方式 3：运行任务

先按快捷键 ctrl+shift+P 显示所有命令，在弹窗运行 sci-inc:build 以及 sci-inc :clean:



在终端中显示如下窗口成功 clean:

```
PS D:\01_codes\11_sci_cube\s907x_sdk> ./gcc_build.bat clean
"Clean in windows enviroment."
D:/01_codes/11_sci_cube/s907x_sdk/build/cmd/win32/rm: cannot remove `D:/01_codes/11_sci_cube/s907x_sdk/out/GCC/Debug/Obj/*.o': Invalid argument
make: *** [windows.mk:130: clean] Error 1
PS D:\01_codes\11_sci_cube\s907x_sdk>
PS D:\01_codes\11_sci_cube\s907x_sdk> []
```

在终端中显示如下信息即成功编译:

```
PS D:\01_codes\11_sci_cube\s907x_sdk> ./gcc_build.bat
"Compile in windows enviroment."
"D:/01_codes/11_sci_cube/s907x_sdk/code"
D:/01_codes/11_sci_cube/s907x_sdk/build/cmd/win32/make -C D:/01_codes/11_sci_cube/s907x_sdk/code
make[1]: Entering directory 'D:/01_codes/11_sci_cube/s907x_sdk/code'
"./atcmd ./cmsis ./core ./net ./startup ./utils"
D:/01_codes/11_sci_cube/s907x_sdk/build/cmd/win32/make -C atcmd
make[2]: Entering directory 'D:/01_codes/11_sci_cube/s907x_sdk/code/atcmd'
"Compiling ./at_cmd.c..."
"Compiling ./at_esp/at_esp_init.c..."
"Compiling ./at_esp/at_network.c..."
"Compiling ./at_esp/at_wlan.c..."
"Compiling ./at_test/adc_test.c..."
"Compiling ./at_test/at_test_6.c..."
```

5 版本信息

日期	版本	更新内容	作者
	1.0	初版发布	钱涛
	1.1		
	1.2		
2019-9-19	1.3	文档格式标准化 增加 VSCODE 编译章节	Steven