

**Ingenic SDK编译体系说明**

文档历史：

|  |  |  |
| --- | --- | --- |
| 版本 | 作者 | 注释 |
| 1.0 | 王秋伟 |  |

**目录**

[1 简介 3](#_Toc517733938)

[**1.1 目录结构** 3](#_Toc517733939)

[1.2 编译过程 3](#_Toc517733940)

[2 编译系统说明 4](#_Toc517733941)

[**2.1 sdk/根目录的编译系统** 4](#_Toc517733942)

[**2.2 source目录的编译系统** 5](#_Toc517733943)

[**2.3 examples目录的编译系统** 6](#_Toc517733944)

1 简介

本文档将介绍如何编译sdk 以及讲解sdk的编译系统设计思路。

**1.1 目录结构**

进入sdk的根目录，执行ls命令查看目录结构，如下图:



各个目录说明:

configs: 存放 sdk的配置文件，例如ms800平台的配置文件是ms800\_sdk\_defconfig

documents: 存放sdk的API说明文档和第三方库的说明文档

examples: 存放sdk各个模块API的使用demo程序

firmware: 存放sdk某些模块API依赖的固件，根据配置文件决定是否安装到文件系统

include: 存放sdk各个模块API的头文件，供应用程序开发使用

lib: 存放各个sdk配置文件编译出来的动态库，全局编译时会安装到文件系统

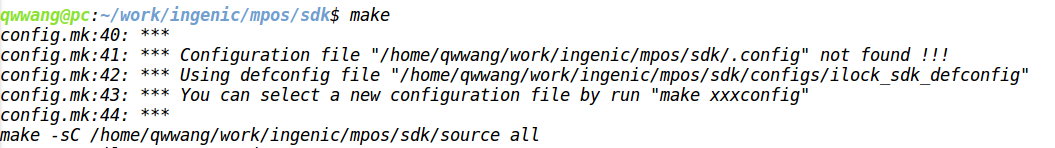
source: 存放sdk各个模块API的源码等，目前对客户不开放

config.mk和Makefile: 用于编译sdk

1.2 编译过程

1) 配置，以ms800平台为例: make ms800\_sdk\_defconfig, 相当于把configs/ms800\_sdk\_defconfig

复制到sdk的根目录 .config。如果sdk的根目录 .config不存在直接编译，如下图:



当sdk/.config不存在时，打印出警告信息并且默认使用ilock\_sdk\_defconfig这个配置，所以在sdk/.config不存在或修改了configs/下的配置文件时，都需要执行 make xxxconfig。

2) 编译: make -j4, 顺利完成将把examples/测试demo、lib/$(TARGET\_DEVICE)/的动态库和firmware/下需要的固件，输出到sdk/out/下的相应目录。

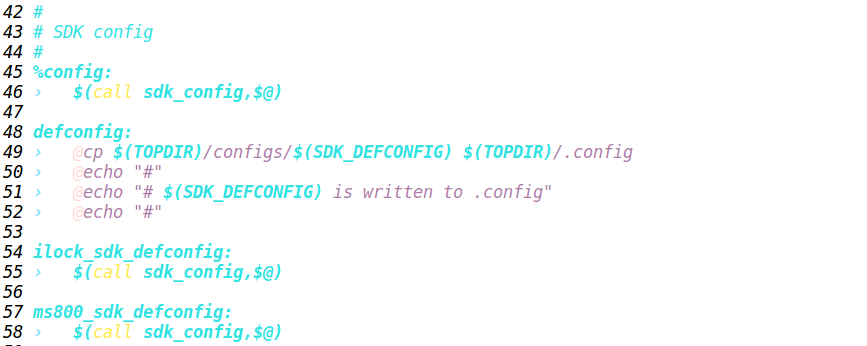
2 编译系统说明

sdk的编译系统包含三部分，分别是sdk/根目录、examples/目录以及source/目录，这三个地方都包含各自的config.mk和Makefile文件，config.mk都是在Makefile的开头包含进来。

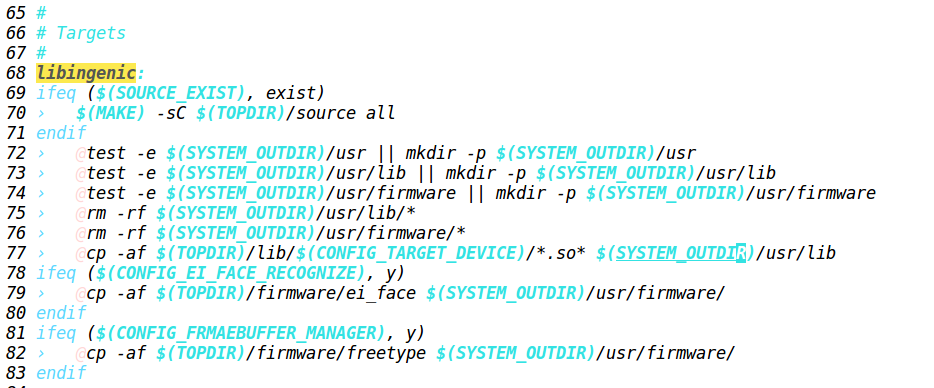
**2.1 sdk/根目录的编译系统**

控制sdk编译的整体流程，包括:

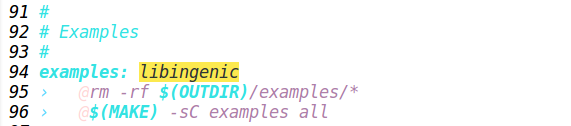
1. 配置sdk, 如下图:



1. 判断source是否要进入source/目录进行编译，创建sdk的输出目录，输出firmware和动态库存到输出目录在，如下图:



1. 进入examples/目录编译测试demo，如下图:



1. 编译清理工作，注意make distclean 才会删除 sdk/.config

**2.2 source目录的编译系统**

这个目录下的config.mk和Makefile主要的工作是编译生成libingenic.so和输出libingenic.so及其依赖的动态库到sdk/lib/$(TARGET\_DEVICE)/，流程如下:

1. 根据sdk/.config决定将哪些模块编译进libingenic.so，目标的生成办法:



1. 调用scripts/libporter.sh将libingenic.so依赖的其他动态库拷贝到sdk/lib/$(TARGET\_DEVICE)/，如下图:



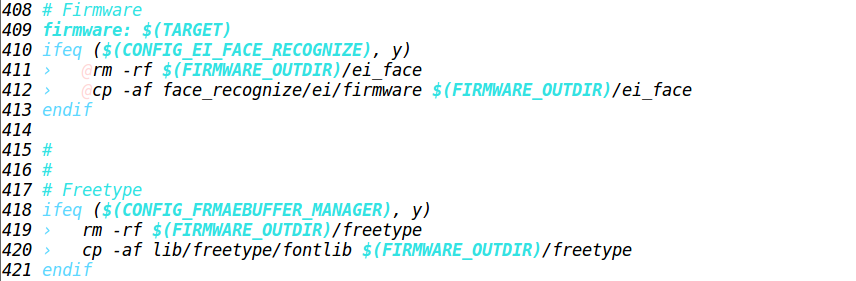
说明: 传给scripts/libporter.sh的参数有三个:

CONFIG\_TARGET\_DEVICE: 在sdk的配置文件中定义，如ms800\_sdk\_defconfig，这参数的值就是“ms800”，跟在编译环境时执行source build/envsetup.sh; lunch得到的TARGET\_DEVICE环境变量是一样的值。

TOPDIR: sdk的根目录

LDFLAGS: 在source/config.mk定义

1. 更新固件到sdk/firmware/，如下图:



**2.3 examples目录的编译系统**

这个目录的config.mk和Makefile目的就是根据sdk/.config编译某些模块API的示例demo，并输出到sdk/out/examples/目录下，编译过程连接的动态库目录是sdk/lib/$(TARGET\_DEVICE)/，如下图:

config.mk

Makefile



编译某个模块API的示例demo的方法:

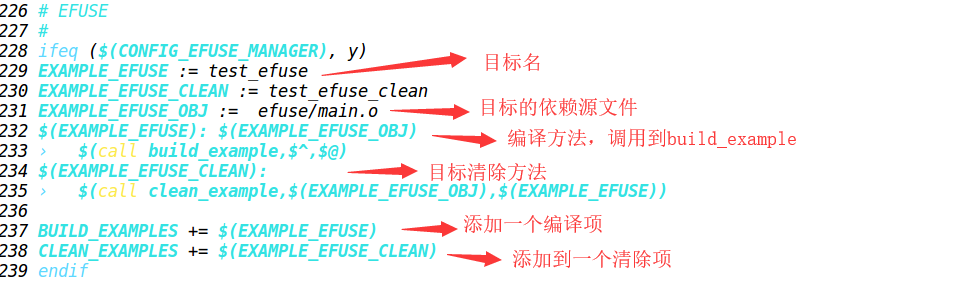
Makefile



config.mk



以编译efuse的示例demo讲解编译过程:





当在sdk/examples/ 执行make all时，各个编译项展开，并调用build\_example方法生成示例demo。