**终端代码统一管理**

# 背景

项目组从2019年开始陆续开发了能源控制器、台区智能融合终端、集中器、专变（I、III、新型）等终端，这些终端都遵循APP化的软件架构，大部分代码是相同的。

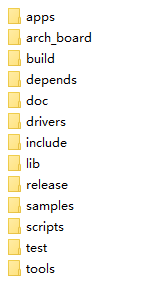
由于历史原因，这些终端的代码没有进行统一管理，对于后续产品维护和开发造成了困扰。主要表现在：

* 一个产品问题的修改需要同步到其他产品；
* 产品新增需求需要同步到；
* 新开发一个产品或者新的核心板需要重新拉一个代码分支开发。

基于当前行情，以上几种终端将长期存在。为了开发维护便利，需要将现有的ECU、SCU、CCU和ZCU（专变）的代码进行统一管理。设计如下的代码架构。

# 新代码架构设计

## 一级目录设计



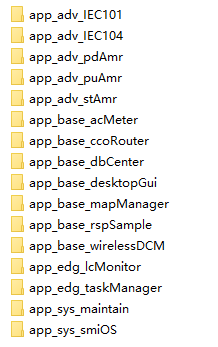
说明：

|  |  |  |
| --- | --- | --- |
| 序号 | 目录名称 | 主要功能 |
| 1 | apps | APP的功能代码 |
| 2 | arch\_board | CPU核心板类型，用于确定gcc工具 |
| 3 | build | 编译脚本，支持设置产品和核心板、地区 |
| 4 | depends | 依赖APP和lib，开源代码或者电科发布的基础APP |
| 5 | doc | 文档 |
| 6 | drivers | 驱动，对应HAL |
| 7 | include | 公共头文件 |
| 8 | lib | 公共库（APP使用的） |
| 9 | release | 编译结果 |
| 10 | samples | 示例 |
| 11 | scripts | 脚本 |
| 12 | test | 测试程序 |
| 13 | tools | 工具 |

## 二级目录设计

* apps

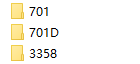
存放APP代码，代码风格继承ECU和SCU。APP通过编译宏区分不通的产品和核心板。



APP代码增加产品宏，通过cmake编译脚本和编译宏适配不通核心板特点。ECU与CCU、专变，最大特点是有模组，三者的模块识别需要做成编译可配置的方式；SCU与ECU相比，APP之间的消息格式变化很大，原有营销的APP还是保持原有的ASN.1格式，有不一致的进行消息格式转换。

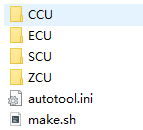
* arch\_board

按照核心板类型创建文件夹，文件夹包含编译需要的基础cmake文件，用于确定gcc版本，当前有3中类型701、701D和3358。



* build

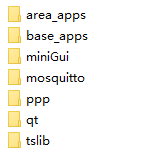
按照一级产品和二级产品分类。



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 序号 | 一级产品 | 二级产品 | 文件 | 说明 |
| 1 | CCU | CCU | make\_apps.sh | 集中器APP编译脚本 |
| 2 | package\_3358.sh | 集中器3358核心板打包脚本 |
| 3 | ECU | FB | make\_apps.sh | 能源控制器电科院APP编译脚本 |
| 4 | package\_701.sh | 701核心板打包脚本 |
| 5 | GB | make\_apps.sh | 能源控制器公变APP编译脚本 |
| 6 | package\_701.sh | 能源控制器701核心板打包脚本 |
| 7 | package\_3358.sh | 能源控制器3358核心板打包脚本 |
| 8 | ZB | make\_apps.sh | 能源控制器专变APP编译脚本 |
| 9 | package\_701.sh | 能源控制器701核心板打包脚本 |
| 10 | package\_3358.sh | 能源控制器3358核心板打包脚本 |
| 11 | SCU | ZX | make\_apps.sh | 提供智芯APP编译脚本 |
| 12 | package\_701D.sh | 701D打包脚本 |
| 13 | XL | make\_apps.sh | 新联自用APP编译脚本 |
| 14 | package\_701D.sh | 701D打包脚本 |
| 15 | ZCU | ZB1 | make\_apps.sh | 专变1型APP编译脚本 |
| 16 | package\_3358.sh | 3358打包脚本 |
| 17 | ZB3 | make\_apps.sh | 专变3型APP编译脚本 |
| 18 | package\_3358.sh | 3358打包脚本 |
| 19 | ZBN | make\_apps.sh | 新型专变APP编译脚本 |
| 20 | package\_3358.sh | 3358打包脚本 |
| 21 | - | - | autotool.ini | 自动编译脚本操作 |
| 22 | - | - | make.sh | HAL、LIB编译，APP编译打包入口 |

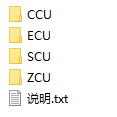
* depends

目前有7个二目录分表表示地区特定APP、基础APP、miniGUI库、mosquitto库（APP）、ppp、QT依赖库、ts触控依赖库。



* drivers

按照产品类型管理。

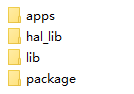


* include、lib

继承ECU、SCU、CCU代码，frame和utility合并到lib中。

* release

编译结果，包括hal、lib、apps和安装包。



* samples

开发样例，目前为空。

* scripts

APP安装、启动相关脚本，按照产品创建目录管理。

* test

测试程序。例如dbCenter的测试程序。

* tools

程序打包工具和补丁制作工具等。