基于Zigbee通信的太阳能终端控制器的程序说明

2014-06-01

[一、程序结构： 3](#_Toc390683288)

[1、KEIL文件结构： 3](#_Toc390683289)

[2、层级结构 3](#_Toc390683290)

[二、主要功能流程图 4](#_Toc390683291)

[1、常规指令执行流程 4](#_Toc390683292)

[2、太阳能数据的查询和定时存储功能 5](#_Toc390683293)

[3、终端自运行功能 6](#_Toc390683294)

[4、加解密功能 6](#_Toc390683295)

[5、升级功能 6](#_Toc390683296)

[三、主要的数据结构 8](#_Toc390683297)

[四、内存规划 8](#_Toc390683298)

# 一、程序结构：

## 1、KEIL文件结构：

应用层：src-ap,prj

中间层：src-mdl，lib-freertos,src-os

驱动层：bsp-broad,bsp-platform，src-drv



## 2、层级结构

应用层

Measure Module

Communication Module

Misc Module

Low priority task

Middle priority task

High priority task

Business Module

HAI(src\_mdl)

RTU Framework

Driver Uart\GPIO\LED\EEPROM\RTC …(bsp-platform)

HAL(bsp-board)

Hardware

Embeded OS(lib-freertos)

Control Module

中间层

驱动层

物理层

A、应用层：包括三个线程，分别是ap\_rtu.c、ap\_comm.c、ap\_messure.c，优先级分别由低到高。其中Ap\_rtu，主要执行指令。Ap\_comm，主要负责数据传输线程。Ap\_messure，主要负责电能计量。

B、中间层：主要包括了OS系统部分、接口部分。其中os部分主要就是freertos系统。接口部分，主要是采用什么通信方式，比如zigbee接口、RS485接口；以及协议打包和解析接口函数

C、驱动层：主要包括MCU外设的驱动，MCU内边特殊功能驱动。

D、物理层：真实的物理电平。

# 二、主要功能流程图

## 1、常规指令执行流程

Connection( uart/message)

Hardware layer

Device(risecomm/renesas)

Protocol ap( iot/huazhi)

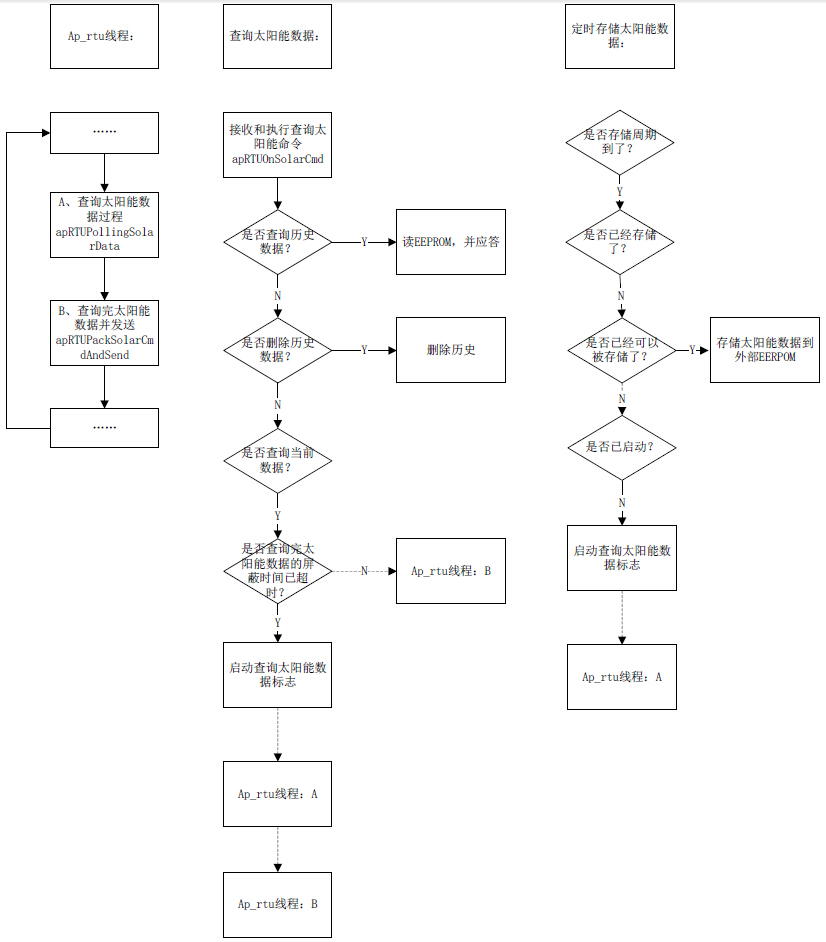
Ap comm/ Ap RTU (message queue)

Connection( uart/message)

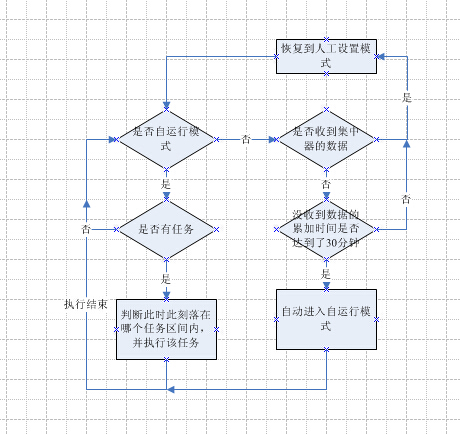
Device(risecomm/renesas)

Protocol ap( iot/huazhi)

## 2、太阳能数据的查询和定时存储功能



## 3、终端自运行功能



## 4、加解密功能

目前支持AES和SM4两种加解密方式。

## 5、升级功能

集中器辅助终端升级流程



应用中升级流程图



BOOT启动流程图



# 三、主要的数据结构

1、关于系统框架

gApList：系统线程框架数据结构

2、关于连接和设备

gConnList：连接的数据结构

gDevList：设备的数据结构

3、关于终端协议和参数

gProApList：解析和打包数据结构

rtu\_para\_t：终端运行参数数据类型

4、关于太阳能数据

gDevList\_level2：接收和发送、太阳能数据解析和打包数据结构

solar\_data\_t：太阳能数据的数据结构

# 四、内存规划

目前选用芯片STM32L15x 有如下存储资源

– Up to 128 KB Flash with ECC

– Up to 16 KB RAM

– Up to 4 KB of true EEPROM with ECC

– 80 Byte Backup Register

* Flash用于存储Boot和App

|  |  |  |
| --- | --- | --- |
| 用途 | 起始地址 | 大小 |
| Boot | 0x8000 0000 | 4K |
| Rsv | 0x8000 1000 | 4K |
| App1 | 0x8000 2000 | 60K |
| App2 | 0x8001 1000 | 60K |
|  |  |  |
|  |  |  |

* EEPROM用于存储参数

终端配置信息

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | 说明 | 例子 | 地址 | 占字节 |
| 开始标志 |  | 固定0x1234 | 0000 | 2 byte |
| 长度 |  |  | 0002 | 1 byte |
| Crc16校验码 |  |  | 0003 | 2 byte |
| 开始地址 |  |  | 0005 |  |
| 地址空间 |  |  | 0000~004f | **80byte** |
| 数据域空间 |  |  | 0005~004f | **75byte** |

灯杆和灯头信息

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | 说明 | 例子 | 地址 | 占字节 |
| 开始标志 |  | 固定0x1234 | 0050 | 2 byte |
| 长度 |  |  | 0052 | 1 byte |
| Crc16校验码 |  |  | 0053 | 2 byte |
| 开始地址 |  |  | 0055 |  |
| 地址空间 |  |  | 0050~009f | **80byte** |
| 数据域空间 |  |  | 0055~009f | **75byte** |

报警信息

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | 说明 | 例子 | 地址 | 占字节 |
| 开始标志 |  | 固定0x1234 | 00a0 | 2 byte |
| 长度 |  |  | 00a2 | 1 byte |
| Crc16校验码 |  |  | 00a3 | 2 byte |
| 开始地址 |  |  | 00a5 |  |
| 地址空间 |  |  | 00a0~014f | **176byte** |
| 数据域空间 |  |  | 00a5~014f | **171byte** |

任务信息

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | 说明 | 例子 | 地址 | 占字节 |
| 开始标志 |  | 固定0x1234 | 0150 | 2 byte |
| 长度 |  |  | 0152 | 1 byte |
| Crc16校验码 |  |  | 0153 | 2 byte |
| 开始地址 |  |  | 0155 |  |
| 地址空间 |  |  | 0150~027c | **300byte** |
| 数据域空间 |  |  | 0155~027c | **295byte** |

* RAM用于运行时参数存储