

# 分布式控制电动自行车充电桩的网关系统设计

贺松杰

郑州科技学院, 河南 郑州  
Email: 18625515230@163.com

收稿日期: 2021年5月22日; 录用日期: 2021年6月11日; 发布日期: 2021年6月22日

---

## 摘要

本文基于RT-Thread设计了一种分布式控制电动自行车充电桩的网关系统。在这个系统中, 网关负责整个系统与服务器数据的交互, 还要维护与设备之间的通信, 是设备与服务器交互的纽带。本文给出网关系统的硬件电路设计与任务设计, 网关系统主控采用GD32F103VET6, 除了主芯片自带的RTC和看门狗外, 还要有局域网通信模块、4G通信模块、485通信模块、Flash存储模块。任务设计上通过成熟的通信协议来维护内网与外网的通信, 通过可靠的程序设计策略, 保证整个系统有非常高的稳定性。完成充电桩一系列需要采集和上传的指令。

## 关键词

RT-Thread, 分布式控制, 网关, 通信协议, 程序设计策略

---

# Design of Gateway System for Distributed Control of Electric Bicycle Charging Pile

Songjie He

Zhengzhou University of Science and Technology, Zhengzhou Henan  
Email: 18625515230@163.com

Received: May 22<sup>nd</sup>, 2021; accepted: Jun. 11<sup>th</sup>, 2021; published: Jun. 22<sup>nd</sup>, 2021

---

## Abstract

In this paper, a gateway system for distributed control of electric bicycle charging piles is designed based on RT-Thread. In this system, the gateway is responsible for the data interaction between the entire system and the server, and it also maintains the communication with the device, which is the link between the device and the server. This article gives the hardware circuit design and task design of the gateway system. The main control of the gateway system uses GD32F103VET6. In addition to the RTC and watchdog that comes with the main chip, there are also LAN communi-

cation modules, 4G communication modules, 485 communication modules, and Flash storage module. In the task design, the communication between the internal network and the external network is maintained through a mature communication protocol, and a reliable program design strategy is adopted to ensure that the entire system has a very high stability. Complete a series of instructions that need to be collected and uploaded on the charging pile.

## Keywords

RT-Thread, Distributed Control, Gateway, Communication Protocol, Programming Strategy

Copyright © 2021 by author(s) and Hans Publishers Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

## 1. 引言

随着科技的发展和城市化的不断发展,城市交通拥堵问题严重。越来越多的人选择电动自行车出行。据 2020 年统计,我国电动车保有量高达 3.5 亿辆,是真正的国民交通工具。目前电动自行车每年发生火灾的概率为十万分之二。75%的电动自行车火灾,是在充电时发生的。2018 年 5 月 15 日,国务院安委办下发《关于开展电动自行车消防安全综合治理工作的通知》。该通知工作措施第二条,明确鼓励新建住宅小区同步设置集中停放场所和具备定时充电、自动断电、故障报警等功能的智能充电控制设施。电动自行车充电桩往往比较密集,数量比较多,传统的电动自行车充电桩往往设计 1 拖 10 的充电桩,但布线较复杂。分布式控制的充电桩也多采用有线 485 的设计,数据交换速度比较慢。本设计则是基于以太网的局域网而设计的电动自行车充电桩的网关系统,数据交换速度快,布线简单,控制数量多,且可远程升级。实现对充电桩设备一对多的全面管理,维护设备的状态,并与服务器实时交换数据。

## 2. 系统整体设计与功能要求

本系统设计的充电桩系统为分布式控制的充电桩系统,采用网关来对所有的设备进行数据处理和转发,有效降低设备对服务器的依赖性,减小服务器的压力,节省 4G 流量,降低产品的运营成本,具体结构如下图 1 所示。

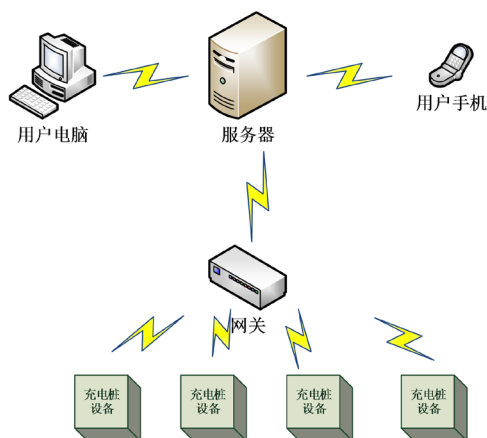


Figure 1. System frame diagram

图 1. 系统框架图

如上图所示，网关直接和服务器通信，服务器不参与充电桩之间的直接数据处理，这样的设计使得设备与网关的通信方式可以有更多的选择，而不局限一种通信方式，而且有利于后期的硬件设备和程序的升级，因为网关的存在，设备不与服务器直接交互，所以升级设备的通信方式，服务器不用做任何修改。设备程序升级时，网关可以缓存设备的升级文件，再进而逐个下发升级，这样大大减少了设备因为升级而消耗更多的流量。

网关要实现与服务器的功能有：远程抄表、远程获取设备状态、远程开启和关闭设备、远程充电信息获取、远程下发和上传充电订单、远程升级网关系统与设备系统。网关与设备之间要实现的功能有：网关发现设备、设备主动获取网关信息、设备状态上传、网关维护设备心跳状态、转发设备与服务器之间开启充电、结束充电、功能和电能信息、主动结束充电等功能。

## 2.1. 硬件框架图

本系统网关与设备之间通过以太网的方式连接，利用成熟的以太网通信方式可靠性高，且数据交换速度快，适合短时延，数据交换量大的场合。网关与设备之间的交互方式如下图 2 所示。

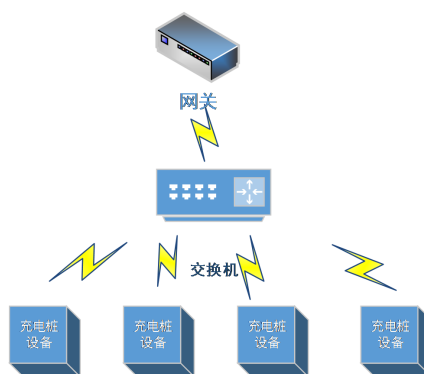


Figure 2. Gateway device connection diagram

图 2. 网关设备连接图

本系统主系统采用基于 CORTEX-M3 的 GD32F103VET6 单片机，对服务器端本系统通过 4G 通信模组连接到服务器进行数据通信，本地设备与网关的连接则采用以太网的局域网通信方式。网关系统通过 W5500 接入交换机，设备端可采用 ESP8266 来接入交换机系统，本文程序是在 RT-Thread Studio 集成开发环境中开发，基于 RT-Thread 操作系统，建立多任务，通过交换机建立局域网，网关系统通过 UDP 通信与设备进行数据交互，通过 4G 完成与服务器的数据交互，最终实现设备到服务器的数据交互。

网关整体系统框架如下图 3。

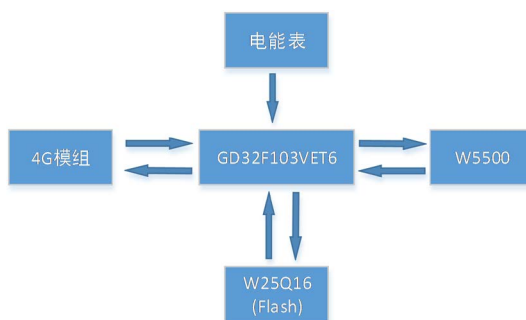


Figure 3. Gateway hardware framework diagram

图 3. 网关硬件框架图

## 2.2. 硬件系统设计

硬件系统的稳定性设计决定了整个系统的稳定性，根据环境的要求，我们设计产品为民用级产品，但是在器件选型方面按工业级的标准来选型。硬件系统决定了整个系统的通信交互方式。

主控芯片与各设备的通信接口如下：通过 485 总线访问电能表来获取当前总的电量以及工作电压和工作电流。通过 SPI 接口与 W5500 进行通信。通过 SPI 接口读写 Flash 芯片。通过异步串口通信访问 4G 模组。主控芯片支持 5 个异步串口、3 个 SPI 接口以及 2 个 I2C 接口，满足系统的要求。

### 2.2.1. 主控系统设计

现今市面上的网关大多采用高端嵌入式处理器加操作系统设计，预置有丰富的应用软件，配置较高，导致成本高，不太适用于本系统场景[1]。本系统采用兆易系列 GD32F103VET6，相较于 ST 系列具有更高的性价比。架构上采用 ARM 公司的 M3 新内核。性能提升 30%，主频可达 108 MHz，可超频到 120 MHz。价格上便宜 20%~30%。由于 GD 系列的 Flash 零等待时间，本工程原来使用 ST 所以在工程上需做稍许更改。因为后期要升级系统版本，本文通过外接 Flash 芯片来存储升级程序。具体电路如下图 4 所示。

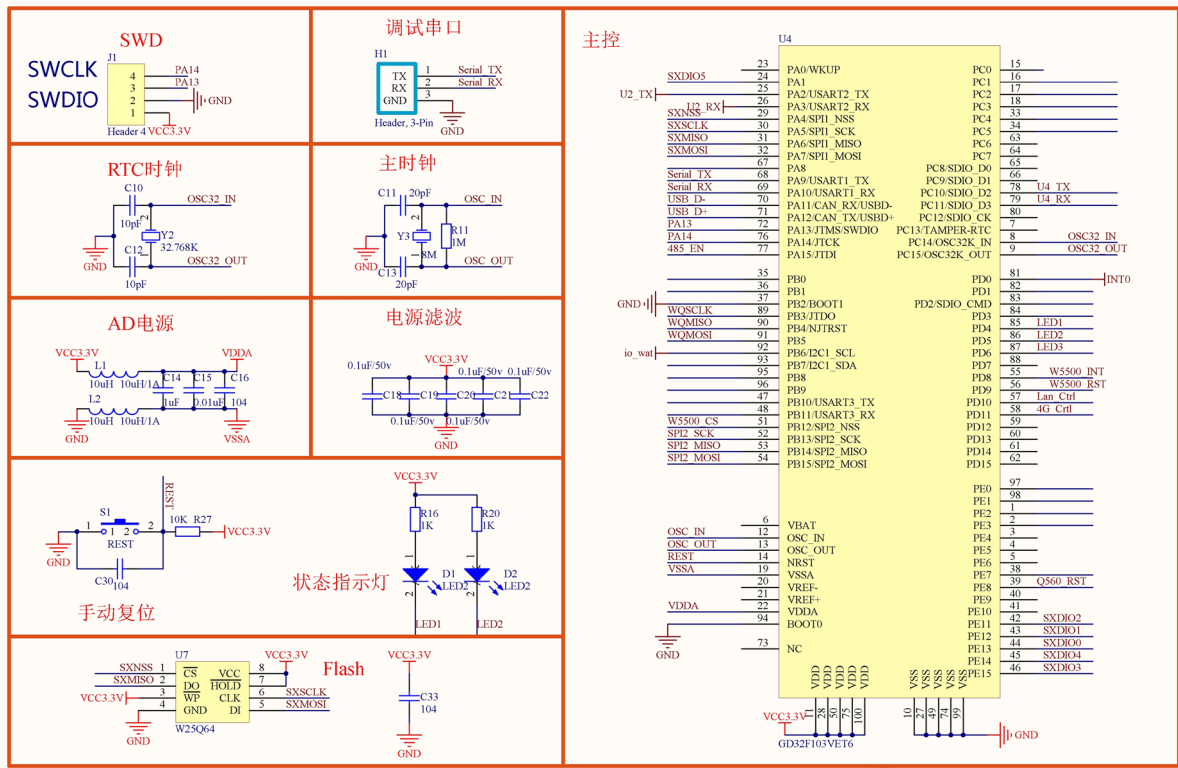


Figure 4. The main control chip circuit  
图 4. 主控芯片电路

### 2.2.2. 电源系统设计

系统的稳定性很大一部分取决于电源的长期稳定性，本系统中网络和主控以及通信部分均采用 3.3 V 供电。本系统选用的 4G 模块工作电流比较大，瞬时电流可达 1.5 A，4G 通信模块的输入电压范围为 5~24 V，模块资料声明电压过低会影响 4G 模块达不到额定-40 度的工作标准，因此本系统采用直流 12V 直接给 4G 供电。通过 AMS1117-3.3 进行前端稳压之后再分别给其他模块供电。本系统设计了断电上传电路，所以通过 DC-DC 设计了 12V 到 5.6 V 的电源电路(图 5)。

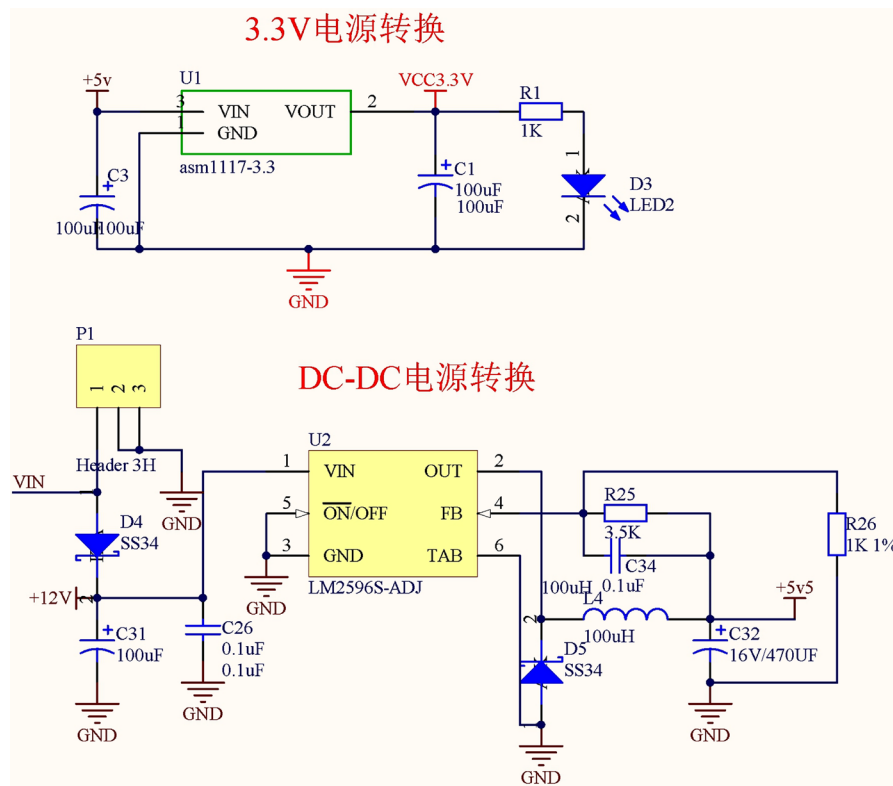


Figure 5. System power circuit  
图 5. 系统电源电路

### 2.2.3. 4G 通信设计

4G 模块选用成都移讯通的 Q560 工业 4G 通信模组，它支持 7 模 15 频，且有多重软件可靠设计与复合式的看门技术。支持 AT 命令进行配置和查询，标准的 TTL 接口通信电平。工业级的通信温度，满足整个系统的要求。与系统的接口比较简单，支持 3.3 V 的串口电平，可直接与模块进行通信连接(图 6)。

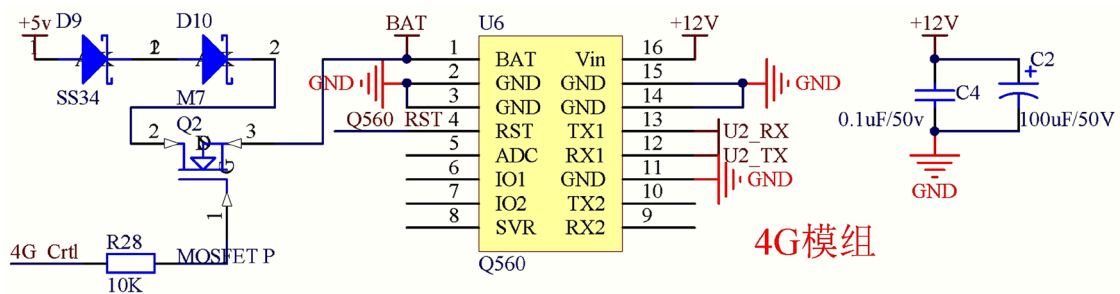


Figure 6. 4G module circuit  
图 6. 4G 模组电路

以太网通信电路选用支持硬件 TCP 协议栈的 W5500，W5500 是 WIZnet 推出的高性能以太网接口芯片系列之一，内部集成全硬件 TCP/IP 协议栈 + MAC + PHY。全硬件协议栈技术采用硬件逻辑门电路实现复杂的 TCP/IP 协议簇，其应用具有简单快速、可靠性高、安全性好等显著优势[2]，支持 TCP，UDP，ICMP，ARP，IPv4，IGMP 及 PPPoE 协议，能提供 8 个独立的 socket，每个 socket 可以含有最大 16 KB 的发送和接收缓存。W5500 的 SPI 接口支持 80 MHz 的通信速率。接口电路如下图 7 所示。

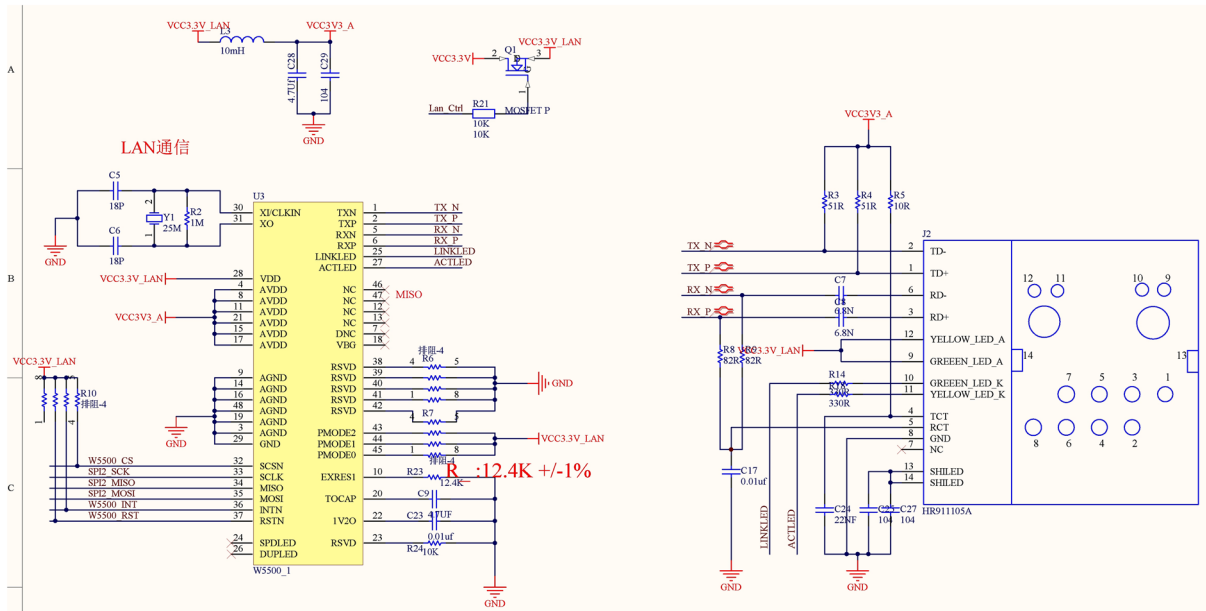


Figure 7. W5500 design circuit  
图 7. W5500 设计电路

### 2.2.4. 485 通信模块设计

485 通信模块主要用于采集外部电能表的信息，来获取当前系统的工作电压、工作电流及电能损耗，实现远程抄表的功能，节省人力资源。系统采用的 485 电路如下图 8 所示。

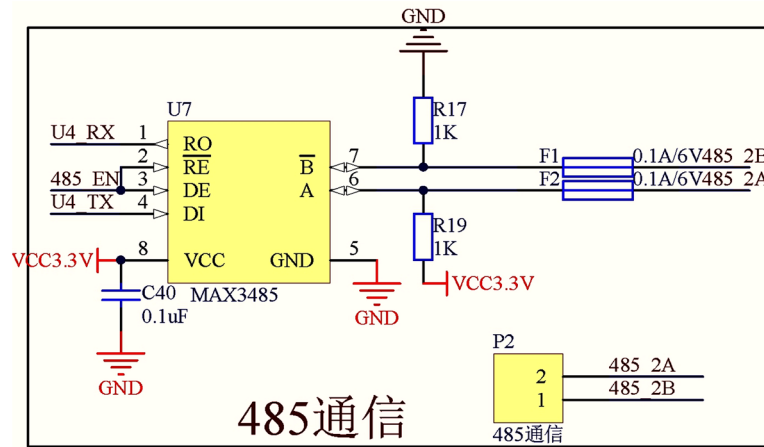


Figure 8. 485 communication circuit  
图 8. 485 通信电路

## 3. 系统软件设计

本系统基于 RT-Thread 系统开发，它是一个嵌入式实时多线程操作系统，基本属性之一是支持多任务线程。根据功能要求我们设定以下 7 个独立线程，线程是 RT-Thread 操作系统中最小的调度单位，线程调度算法是基于优先级的全抢占式多线程调度算法，即在系统中除了中断处理函数、调度器上锁部分的代码和禁止中断的代码是不可抢占的之外，系统的其他部分都是可以抢占的，包括线程调度器自身[3]。



### 3.1. 总体任务设计

优先级的分配比较重要，决定了系统对任务的响应时间，RT-Thread 为抢占式的实时内核，因为要根据实际需要来设计优先级：

任务 1：主函数任务，用于初始化网络设备、4G 通信设备、RTC 时钟、看门狗。对必要的信号量和队列进行初始化。初始化主要的任务函数，之后不断执行看门狗监测任务，主函数的优先级可以在 rtconfig.h 中进行定义 `#define RT_MAIN_THREAD_PRIORITY 10`。

任务 2：UDP 数据处理指令，一旦开始运行后，该任务自动进入挂起状态，等待接收队列的到来，一旦接收到数据则开始对数据进行分析处理。

任务 3：4G 数据处理指令，一旦开始运行后，该任务自动进行挂起状态，等待串口通信分析处理 4G 的数据，若是符合要求的数据则会由串口任务发送数据到队列里，4G 接收任务进入就绪状态，分析并处理数据。

任务 4：UDP 发送任务，把 UDP 发送任务独立出来，减少多任务运行多数据发送的时候引起的冲突问题。

任务 5：设备心跳检测任务，此任务需要维护两个心跳，一个用于和服务器保持长链接的心跳任务，另一个用于定时获取设备状态的任务，网关定时向设备发送心跳，检测设备是否在线。

任务 6：串口数据处理任务，用于处理串口接收到的数据并按协议进行前期的数据处理和分析。

任务 7：串口发送数据任务，用于通过串口向服务器发送数据。此任务在没有数据需要发送时进入到挂起状态，RT-Thread 是抢占式的操作系统，由于串口没有缓冲区，独立成任务可以防止多任务同时使用串口发送造成数据的发送错误。不采用信号量的方式而采用队列，提高数据的传输效率。

### 3.2. 程序设计

#### 3.2.1. 初始化任务设计

本系统在 RT-Thread Studio 中进行编程，在系统配置时已经进行了前期的一些配置，例如串口 1 的相关配置和 W5500 的配置已经自动添加初始化链表中。具体其他的一些初始化配置可参见下图 9。

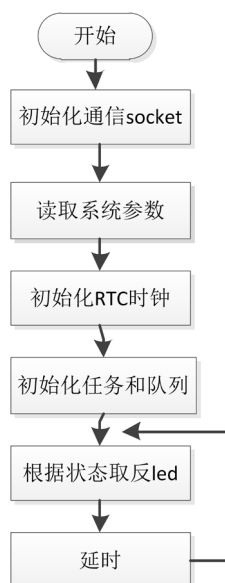


Figure 9. Initialization task

图 9. 初始化任务

系统执行完初始化链表上的任务之后开始执行用户的初始化函数，本系统首先初始了局域网的连接，然后读取 Flash，对用户的一些配置初始化，检测设备是否有升级任务。之后初始化 RTC 时钟，最后初始化系统中需要的队列和信号量以及任务。进行到主函数任务，根据当前状态点亮对应 LED 灯。

### 3.2.2. 4G 通信任务设计

在 main 函数里初始化 4G 相关的任务后，4G 任务的优秀级大于 main 函数，所以会立即运行，开始运行时首先初始化 4G 模块，检测 4G 模块是否存在、读取 4G 信号质量、检测是否连接到网络并读取 IP 地址，初始化服务器地址并开启心跳定时器任务，定时向服务器端发送心跳数据包，来检测设备是否在线。接下来进行挂起状态，等待服务器端数据到来。

与外部通信要完成通过 4G 模块接收信息，然后解析相关数据并进行相关的操作，根据网关要实现的功能我们设计以下流程图 10。

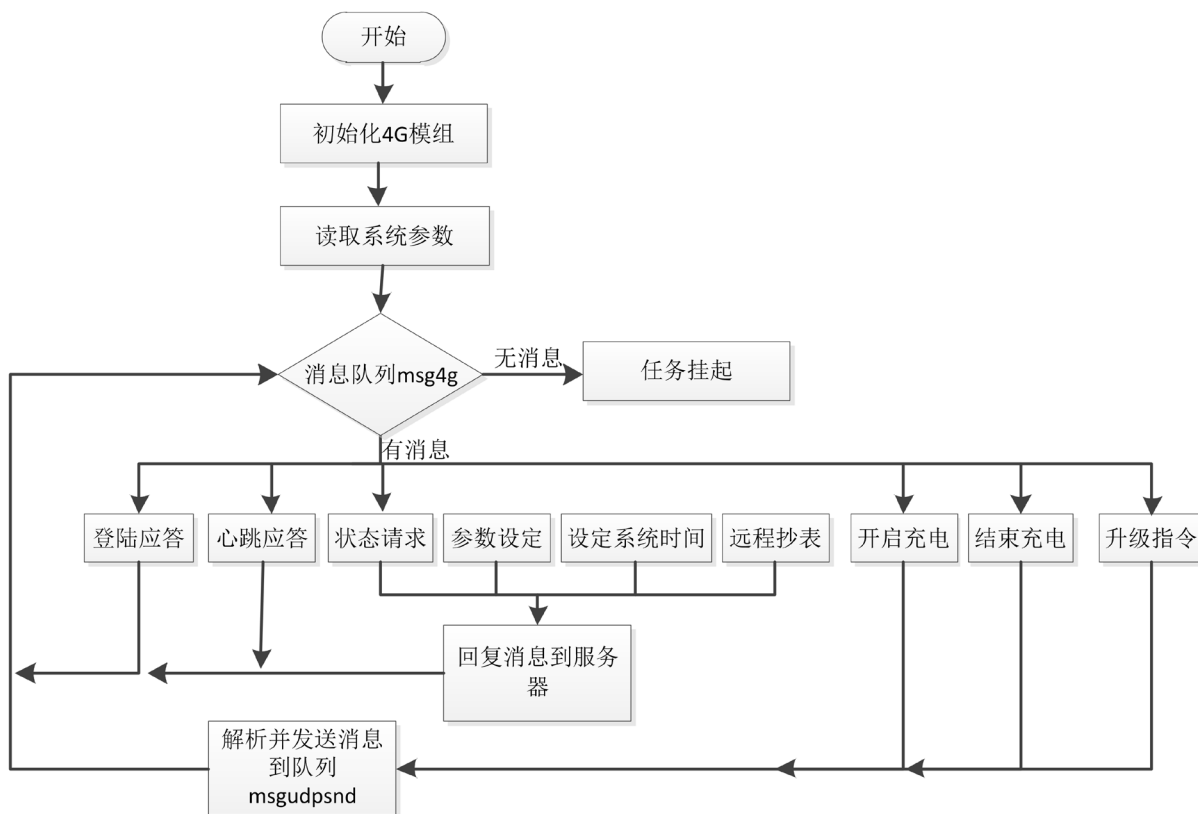


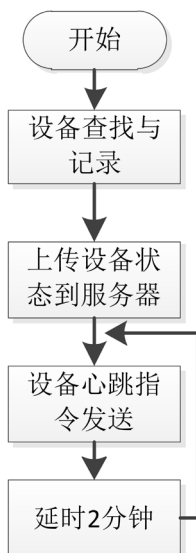
Figure 10. 4G task execution diagram  
图 10. 4G 任务执行图

4G 接收任务并及时处理响应服务器，有些回复信息需要设备的响应，因此数据在处理后应当及时释放 CPU 的控制权。为了保证接收任务能及时的处理，所以设计接收数据队列，根据先进先出的原则依次对信息进行处理。

### 3.2.3. 设备心跳任务设计

在设备出现故障或者网络出现问题时需要及时同步设备状态到服务器端，因为需要定时维护设备心跳，来获取设备的最新状态，心跳任务设计如下图 11 所示。



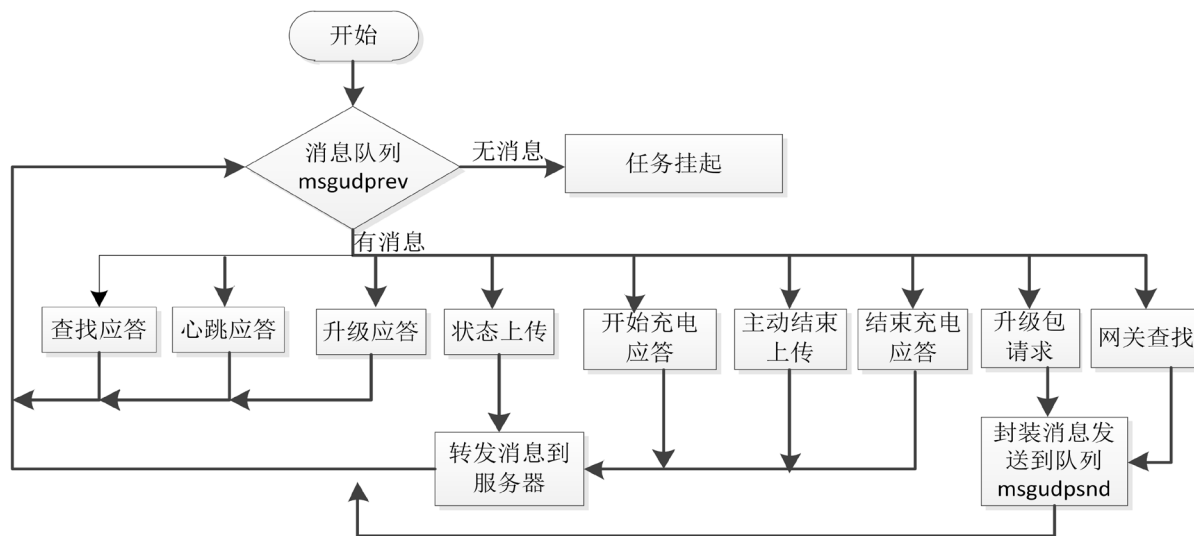


**Figure 11.** Flow chart of device search and heartbeat maintenance  
**图 11.** 设备查找与心跳维护流程图

设备查找是网关通过发送 UDP 广播通信进行设备查找，查找到设备之后会记录设备的 IP 地址以及端口，下次和设备直接就可以直接通过 IP 地址进行通信。设备的心跳维护任务间隔为 2 分钟，以便能及时获得设备状态。

### 3.2.4. 网关与设备通信任务设计

内部通信任务用来维护设备的心跳和状态、设备的升级、转发设备的一些充电相关的指令。通过 UDP 实现点对点的通信，在应用层完成可靠性通信的建立。具体处理流程如下图 12 所示。



**Figure 12.** The gateway processing device message flow chart  
**图 12.** 网关处理设备消息流程图

## 4. 结语

本系统基于 RT-Thread 提供了一种分布式控制充电桩网关的系统设计方法，通过局域网实现一台网

关管理多台充电桩设备，维护设备的状态信息，并实现与服务器间数据的透明转发。网关整体功能考虑周全。本系统在实际长期运行中可靠稳定，具有一定的实用价值。

### 参考文献

- [1] 邹晓康, 刘帅, 张浩然. 基于 STM32 嵌入式多协议网关设计[J]. 微型机与应用, 2015, 35(16): 38-40+43.
- [2] W5500 文档中心[EB/OL]. <http://www.w5500.cn/>
- [3] RT-Thread 文档中心[EB/OL]. <https://www.rt-thread.org/document/site/#/>, 2020.