

## 说明书摘要

---

随着物联网技术的兴起，智能家居的发展越来越受到重视，其利用网络通信、信息家电、设备自动化等技术优势，为人们提供高效、舒适、便捷的居住生活环境。移动互联网的飞速发展，为智能家居控制提供了更优越的平台，使其得以实现多终端远程监控、控制、管理等功能。然而在多移动终端的情况下，智能家居控制需要面对设备状态数据同步的问题。因此，本发明设计一个用于智能家居移动终端数据同步、推送的系统，提高智能家居控制的实时性。

# 摘要附图

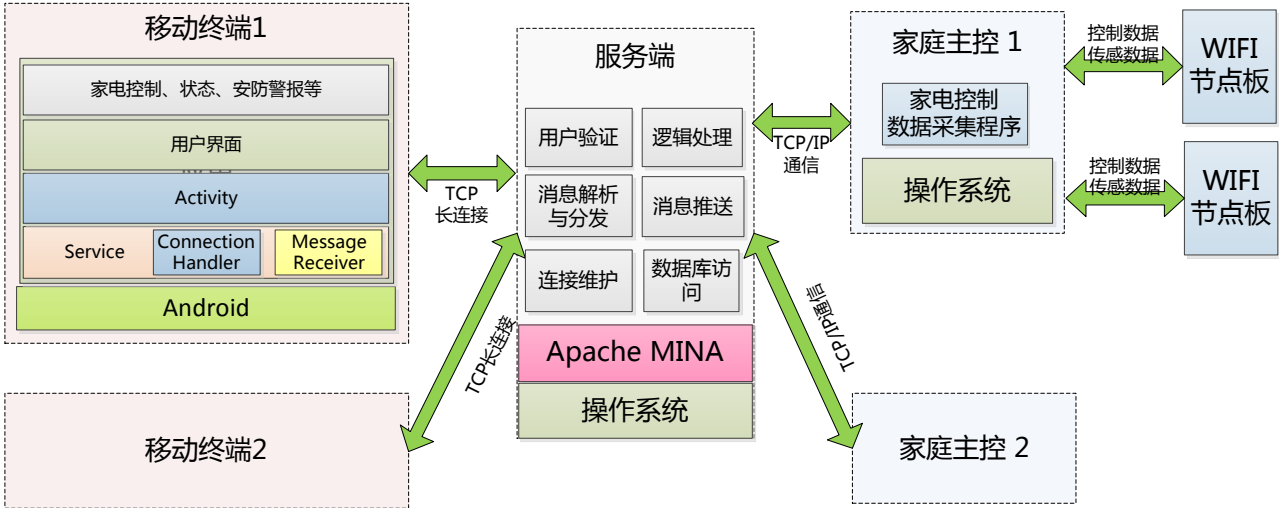


图 1

## 权利要求书

---

# 说明书

---

## 一种基于 Apache MINA 的智能家居多移动终端数据同步与推送系统

### 1 技术领域

本发明属于智能家居移动终端的数据同步设计领域。

### 2 技术背景

近年来随着物联网技术的兴起，以及移动互联网的蓬勃发展，人们也在智能家居控制方面有了新的需求，家居智能化优化人们的生活方式，为人们带来了生活上的便捷体验。许多互联网公司也不断地与家电厂商合作创新，推出各式各样的智能家居控制产品。在这些产品中，大多数都集成了家电控制、监控、管理等功能，在远程移动终端控制方面，尚未做到家居电器的实时信息反馈与多终端同步，实时性欠佳。

目前智能家居控制方面的技术，主流的有基于 WIFI、Zigbee、433Mhz 等通信技术，其家庭主控一般有基于 ARM 架构或 Intel X86 架构的控制终端，移动终端主要以 Android、iOS、Windows Phone 操作系统为主，但是在移动终端上没有提供较好的多终端实时数据同步系统，会导致家电状态无法及时获知，甚至导致多终端控制的状态错乱。

本发明主要介绍了开源框架 Apache MINA 并提出了基于该框架的一种面向

智能家居控制的多移动终端数据同步、数据推送系统的解决方案，主要包括家庭主控数据采集模块、服务端数据处理模块以及 Android 移动终端处理模块。

### 3 发明内容

本发明的目的在于针对智能家居控制系统中的多终端操作情况下的数据实时同步问题，设计并实现了一种面向智能家居的多移动终端数据同步、推送的解决方案。

为了实现本发明，采用的技术方案如下：

一种面向智能家居多移动终端数据同步与推送系统，主要有三大模块，在家庭主控机上部署基于 TCP/IP 传输协议的家电控制与状态数据采集模块，在服务器上部署基于 Apache MINA 的消息同步、推送的核心模块，在 Android 移动终端运行智能家居应用程序。三个模块间均基于 TCP/IP 网络通信协议。在服务端中的核心模块，引入的 Apache MINA 开源框架，是基于 JAVA 异步网络 IO 接口，提供了网络传输上抽象的事件驱动的异步 API，模块基于该框架，维护了多终端与服务器之间的 TCP 长连接，并为家庭主控提供了家电数据反馈的服务接口。

本发明的主要内容包括基于 Apache MINA 框架的一种智能家居控制中数据同步、数据推送的设计方案，如下：

#### 1) 基于 Apache MINA 的核心模块

该模块是基于 Apache MINA 开源网络框架的 Java 应用程序，主要包括的子模块有：Long Connection Handler（TCP 长连接维护）、Logic Handler 以及基于 Hibernate 的数据库访问模块。Long Connection Handler 通过 Apache MINA 的 IoSession 来维护移动终端与服务器之间的 TCP 长连接，该子模块是基于 Apache

MINA 的 `IoHandlerAdapter` 来处理网络数据 IO，并将消息进行解析，区分消息类型，如果是来自客户端应用程序的心跳包，则进行心跳包回复，如果是属于带有数据载荷的消息，则交给 `Logic Handler` 的 `Message Dispatcher` 进行消息分发，随后由不同的逻辑模块去处理消息。各种消息包括了智能家居用户的登录身份验证消息、家庭主控的注册消息、各种类型的推送消息（如状态信息同步的推送消息）等。`Logic Handler` 子模块根据不同类型的消息，进行相应的逻辑处理，在处理消息的过程中，将使用到数据库访问模块。数据库访问子模块基于开源的数据对象关系映射框架 `Hibernate`，提供了简便快捷的数据库访问。

## 2) 在家庭主控上部署的家电数据采集模块

该模块是一个基于 `TCP\IP` 网络通信的应用程序，部署于家庭主控上，负责家电控制状态数据、安防传感设备数据的采集工作。家庭主控是一台 `x86` 架构的微型计算机，在智能家居解决方案中将放置于对应的家庭居住环境里，其与 `WIFI` 节点板块通过 `WIFI` 通信互联，`WIFI` 板块作为家电控制数据的中转站，将家庭主控中的控制程序发来的指令，通过 `433Mhz` 通信来控制家电。与此同时，家电设备以及安防设备可以通过 `433Mhz` 并经由 `WIFI` 节点板块传输给家庭主控中的控制程序。而作为控制程序的一部分，主控上部署的家电数据采集模块，可以从中采集到家电设备反馈的设备状态，以及安防设备反馈的传感器数据值。这些数据值再通过固定格式的消息，发往服务器进行消息处理，并推送到多个移动终端。

## 3) `Android` 移动终端上部署的网络数据处理模块

该模块部署于 `Android` 移动终端上，并将最新的家电设备状态信息、智能家居安防传感器数据信息进行处理后，通过用户界面展现给智能家居移动终端用

户，用户将实时了解家电的最新状态以及安防系统的实时报警状态。在 Android 移动终端上的网络数据处理模块，包括运行于 Android 应用程序的后台 Service，其与消息处理服务端建立 TCP 长连接，同时 Connection Handler 通过心跳包与服务端来维持 TCP 长连接，保障通信链路的畅通，提高了状态数据同步的实时性。每当接收到从服务端推送过来的消息，将进行解析并处理。如果接收到的是安防传感器的警报数据，将通过 Android 的响铃或者震动方式提醒用户，如果接收到的是家电设备状态信息，通过 Android 的 Activity UI 显示给用户。

本发明基于 Apache MINA 开源框架，并通过 TCP 长连接来进行智能家居中多移动终端的数据同步，具有以下几个优点：

- 1) 高并发访问，减少服务端资源开销。Apache MINA 开源网络应用程序框架，是通过 JAVA 的异步网络 IO 提供了 TCP/IP 和 UDP/IP 的支持的，而异步网络 IO 处理，根据网络 socket 的状态轮询结果来进行处理，单进程也可以做到高并发访问，相较于传统的多线程处理网络连接的模型，更节约服务器资源，不用为每个请求连接安排独立的线程去处理。
- 2) 基于维护 TCP 长连接，保障了数据同步的高实时性。移动终端与消息推送服务器之间，以及家庭主控的采集模块与消息推送服务器之间，采用了 TCP 长连接的通信方式，使服务端可以主动推送数据同步的消息到移动终端，而移动终端不需要主动地去轮询服务器。
- 3) 家庭主控的数据采集模块，将状态数据及时推送到移动终端。家电设备的状态数据通过 WIFI 节点板反馈到家庭主控计算机上，数据采集模块将最新获取的需要同步更新的数据，立即发送消息到服务器，服务器对该主控进行数据库查询，并将最新消息推送到已登录的移动终端。用户能

够获取最新的家电设备状态数据。

- 4) 多移动终端的同步支持。通过多用户多移动终端登录，并将用户注册绑定到对应的家庭主控，则家庭主控能够一对多地进行消息推送，能够将家电设备的状态信息同步到多移动终端中。
- 5) 自定义的 TCP 长连接心跳，节省移动互联网数据流量，并能及时检测连接状态。自定义的消息协议，采用了 2 个 byte 的心跳包，仅用于维护 TCP 长连接，不需要轮询请求数据，从而节省了移动互联网数据流量，同时能够及时接收到服务端通过 TCP 长连接传送过来的同步数据。

#### 4 附图说明

图 1 为本发明的智能家居数据同步、推送解决方案总体设计图；

图 2 为本发明的智能家居控制状态数据同步总体架构图；

图 3 为本发明的基于 Apache MINA 的数据同步、推送服务端程序设计图；

图 4 为本发明的智能家居控制状态数据同步协作图；

图 5 为本发明的自定义消息协议说明；

#### 5 具体实施方式

本发明设计了基于 Apache MINA 框架的面向智能家居控制的数据同步、推送系统，它支持多移动终端的家电状态数据、安防设备警报数据同步。下面结合附图，对本发明做进一步的阐述。

##### 5.1. 数据同步、推送的总体架构；

附图 2 所示为本发明设计的硬件环境架构，其中移动终端为 Android 系统的



智能终端，具有移动互联网访问功能，智能家居的远程控制应用程序运行于其上，用户可以通过 Android 的智能家居 App 控制并查看最新的家电状态、安防系统警报信息等。图中服务器接入了互联网，其中部署了基于 Apache MINA 的消息推送、数据同步核心模块程序，其中也部署了需要用到的数据库 MySQL 数据库。在每一个家庭中，会有一个家庭主控，家庭主控为 x86 架构的微型计算机，其中部署了数据控制、采集程序。而 WIFI 节点板块与主控接入以太网中，它通过 433Mhz 与家电设备进行通信，包括家电设备、安防设备的控制信息传输与状态信息反馈等。

家庭中的主控接入互联网，与服务器进行连接，并注册、反馈运行状态；与此同时，移动终端也通过互联网连接至服务端，通过 TCP 长连接进行通信，可以随时接收到服务端的推送消息。而家庭主控采集的家电设备状态信息，就可以通过服务器推送到多个移动终端设备上，通过 UI 显示出来、提醒用户。

## 5.2. 基于 Apache MINA 框架的核心模块；

该模块部署于附图 2 所示的服务器上，附图 3 所示为该模块内部架构。因为 Apache MINA 是基于 Java NIO 的，其需要 Java 虚拟机的支持。在 Apache MINA 之上，主要有 Long Connection Handler（长连接维护子模块）、Logic Handler（逻辑处理子模块）以及另外的基于 Hibernate 的 DBhelper（数据库访问子模块）。长连接维护子模块，利用 Apache MINA 的 IoHandlerAdapter 进行连接处理，其将一个 TCP 连接封装成 IoSession 对象，在服务器处理连接的时候，将触发 sessionOpened()、messageReceived()、sessionIdle()、exceptionCaught()、sessionClosed()等关键回调函数。其中 Protocol Parser 负责消息协议解析，区分消息类型，读取消息载荷，如果是心跳包类型的消息，则交给 Heart Beat Handler

处理心跳回包，经过解析后的消息类型与载荷，由 Message Dispatcher 分发，从而交给逻辑处理子模块，而长连接处理子模块的 Protocol Packer 还负责将逻辑处理的消息载荷封装，通过网络 IO 发送出去。在逻辑处理模块中，根据不同的消息类型，进行不同的逻辑处理，Login Handler 负责登录身份验证，Timer 负责清理 60 秒内连接但未通过登录的连接。而每个连接，都可以在 LongConnHolder 中找得到 Iosession 对象，其 Iosession Map 以 session 的 ID 为键，通过 ID 可以查找到具体的连接会话。在移动终端用户登录或者家庭主控注册状态是，需要把 Iosession 的 ID 保存到数据库中，与设备绑定。而数据库的访问，则交给 DBhelper，它将 DAO 对象通过 Hibernate 框架操作 MySQL 数据库。当消息类型不同时，有不同的 Handler 处理，包括 Alarm Message Handler 处理安防警报信息、Sync Message Handler 处理状态数据同步信息、Update Message Handler 可以处理应用程序更新的消息推送。需要推送的消息，Message Pusher 通过获取对应 ID 的 Iosession 对象，将处理好的回发消息推送出去。

### 5.3. 部署于家庭主控的家电数据采集模块；

在图 1 中可以看到每个家庭有一个家庭主控，而该模块作为一个程序运行于主控中，该模块会实时地从家电控制程序中获取家电设备、安防设备通过 WIFI 节点板反馈的状态数据。该模块与服务端之间是基于 TCP 协议通信的，模块发给服务器的消息，经过服务器上的基于 Apache MINA 的核心模块处理后，推送到对应的已登录的智能移动终端中。

### 5.4. 基于 Android 移动终端的网络数据处理模块；

如图 1 所示，多移动终端通过 TCP 长连接与服务器建立通信信道，而在移动终端中，网络数据处理被放在了 Android 的 Service 组件中，作为 UI 组件的 Activity

则与后台 Service 进行通信。该模块包括了 Connection Handler 来维护 TCP 长连接，它会每隔 25 秒发一个心跳消息至服务器，同时也接受服务器发回的心跳。如果长时间没有收到服务器发回的心跳，则认为此连接已经断开。在 Service 中，还有独立的线程接收网络数据，并把接收到的数据通过 Activity 显示到 UI 上，或者开启震动和声音。

### 5.5. 数据同步、推送过程；

附图 4 所示为家电设备状态数据、安防传感数据的同步、推送流程。对于家庭主控，在建立 TCP 连接后将把家庭主控的唯一识别 ID（UUID）进行在线状态注册。移动终端服务端建立连接后进行登录身份验证，否则连接将自动断开。移动终端与家庭主控，均通过心跳包与服务端保持连接、保持状态，一旦心跳包消失，则表示长连接断开、离线。当家庭主控采集到家电设备发过来的安防警报数据或者是控制状态改变数据时，该模块将数据封装、分类，发送至服务器，服务器接收到主控发来的请求推送的消息后，将根据主控的 ID 查询数据库，把推送消息推送至相应的多个移动终端。移动终端收到推送消息，根据推送消息的种类、进行相应处理，如收到警报消息后通过震动或者响铃方式通知用户，收到家电设备状态同步的推送数据后，进行家电状态的 UI 同步显示。

### 5.6. 消息协议

附图 5 所示为传输的消息自定义协议。心跳包为 2 字节长度，规定第一个字节为 0x01，心跳回包为 0x02，第二个字节为预留。消息头包括 4 个字节，第 1 字节为消息类型（不包含 0x01、0x02），第 2 字节预留，第 3、4 个字节为消息负载的长度。

## 6 说明书附图

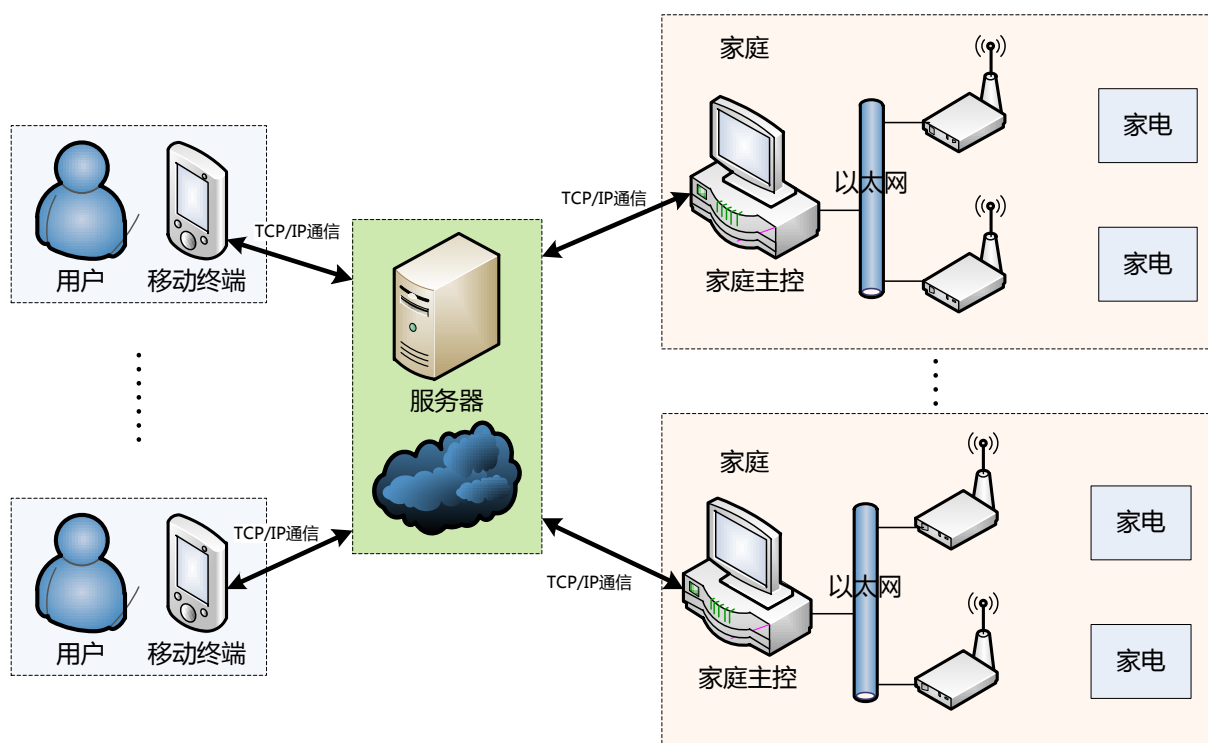


图 2

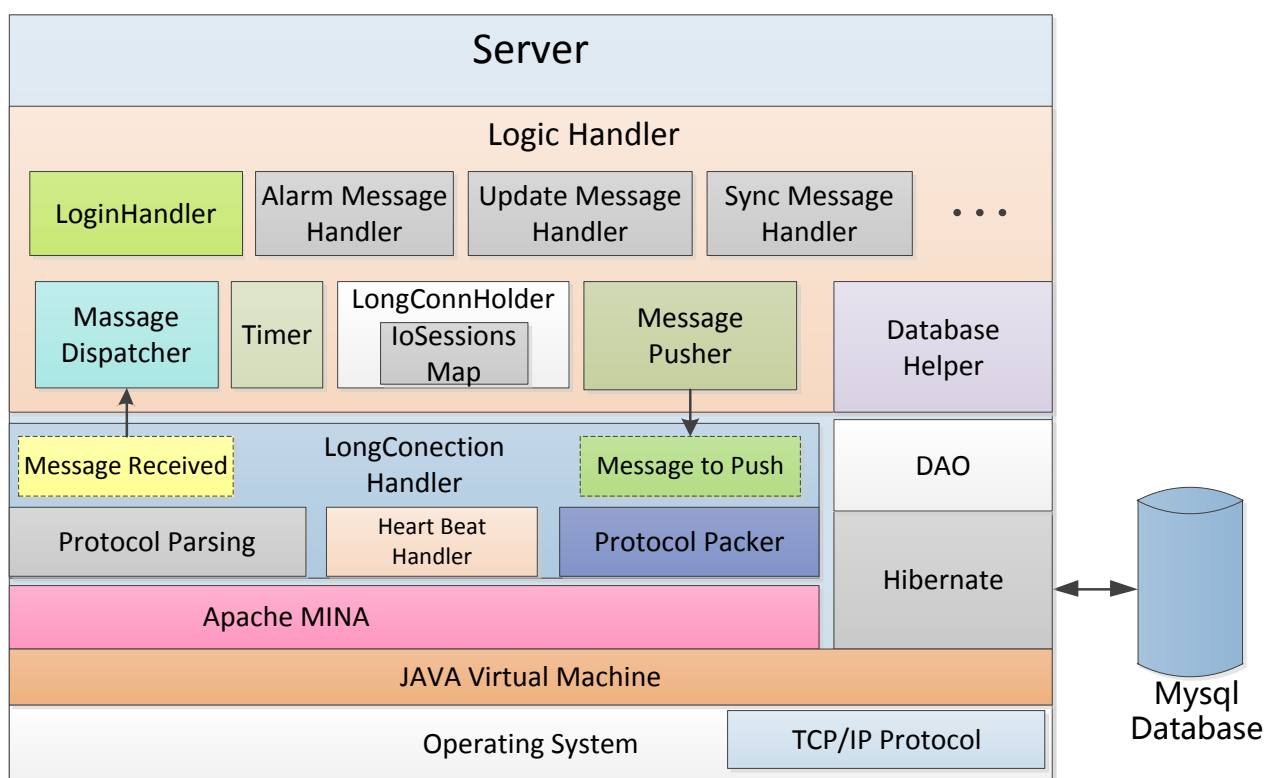


图 3

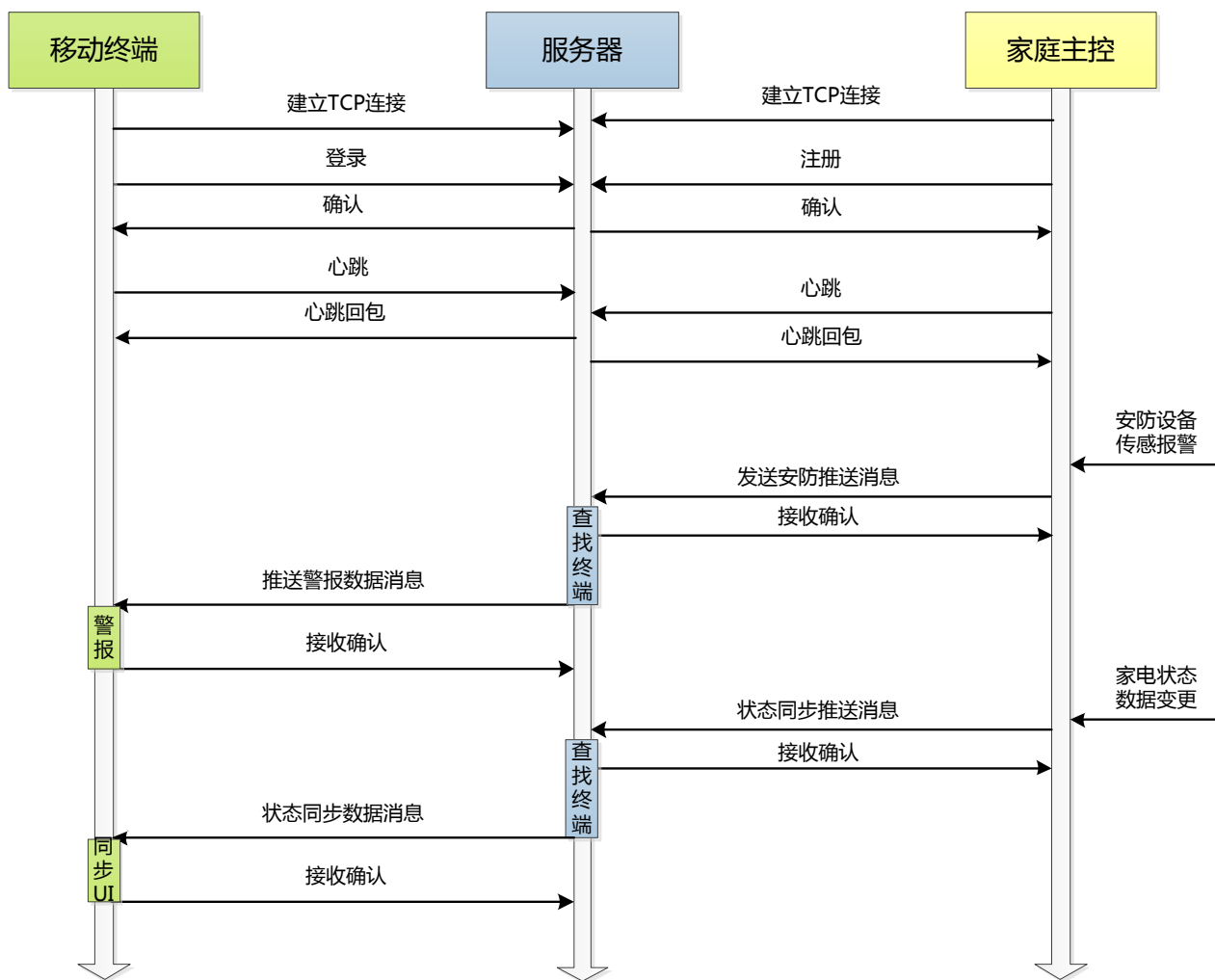


图 4

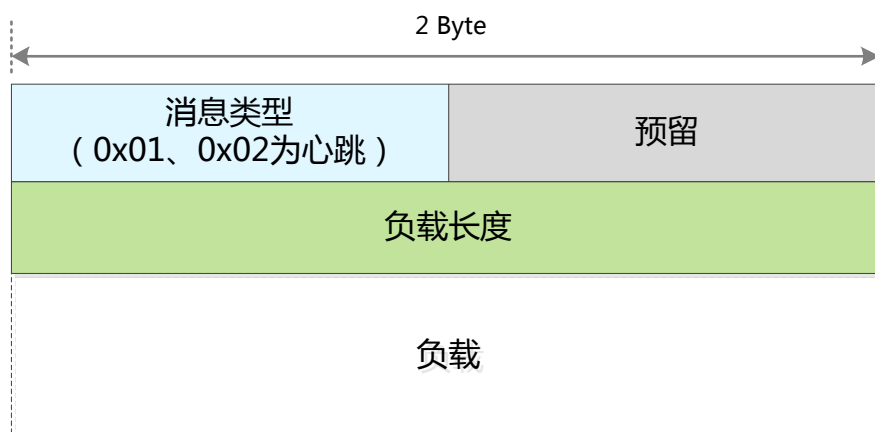


图 5