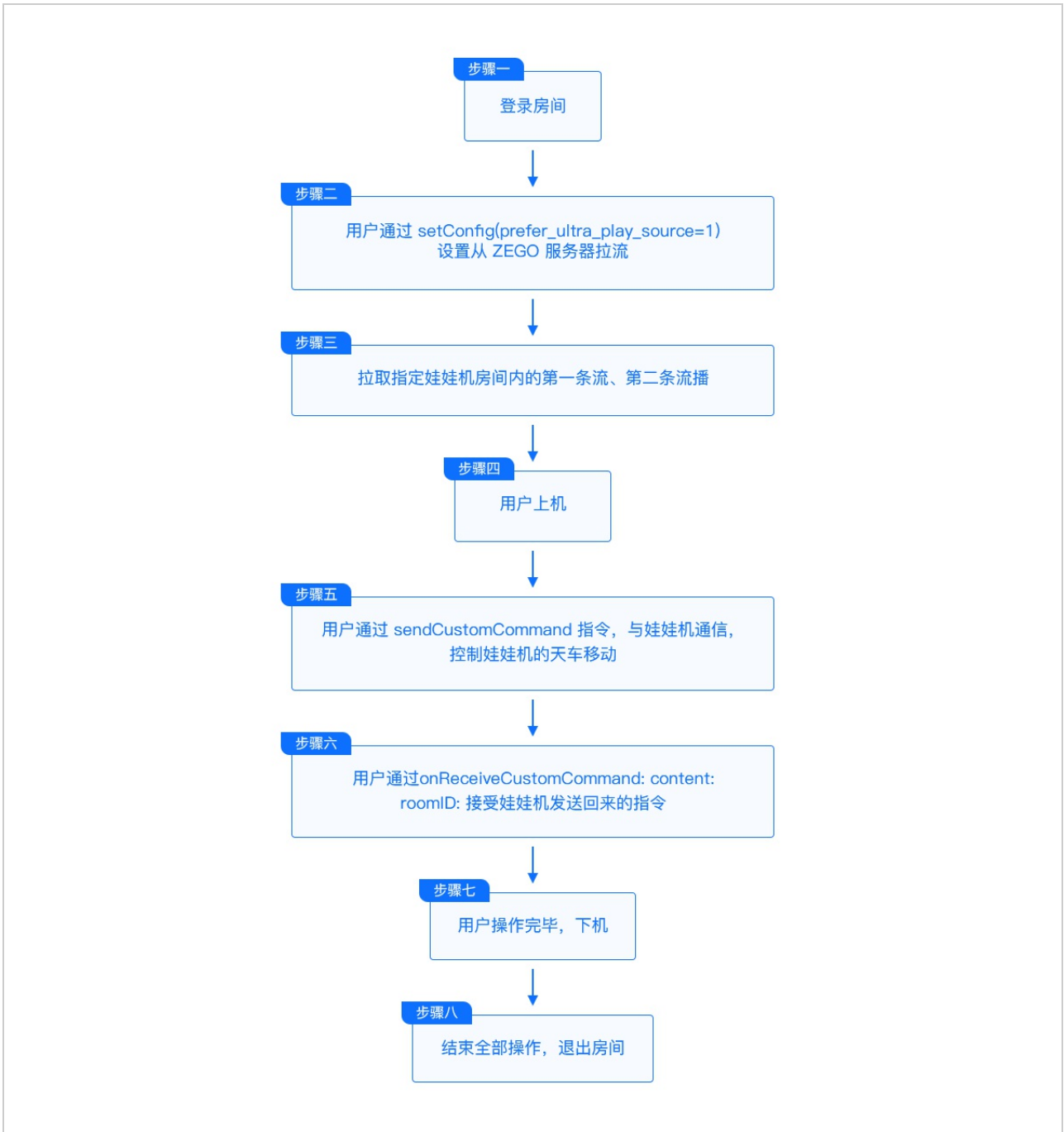


# 功能实现流程

娃娃机客户端 主要结构及流程如下图所示：



请注意：

1. 上图中的 **用户选择上机** 步骤，**娃娃机客户端** 不需要进行流的切换，仅需要处理业务信令、UI 交互等。
2. 开发者要注意区分 **娃娃机客户端** 和 **娃娃机控制端**。**娃娃机客户端** 指提供给玩家抓娃娃的、安装在 iOS 或 Android 平台的 App。**娃娃机控制端** 指安装在 Android 开发板上的 Android 程序(通过串口控制娃娃机)。
3. 为了便于开发者更快理解 **娃娃机客户端** 中的逻辑，后面的小节会通过源码讲解核心功能，开发者亦可直接阅读完整的 **娃娃机客户端** 源码，逻辑是一致的。

## 1 娃娃机系统实现流程

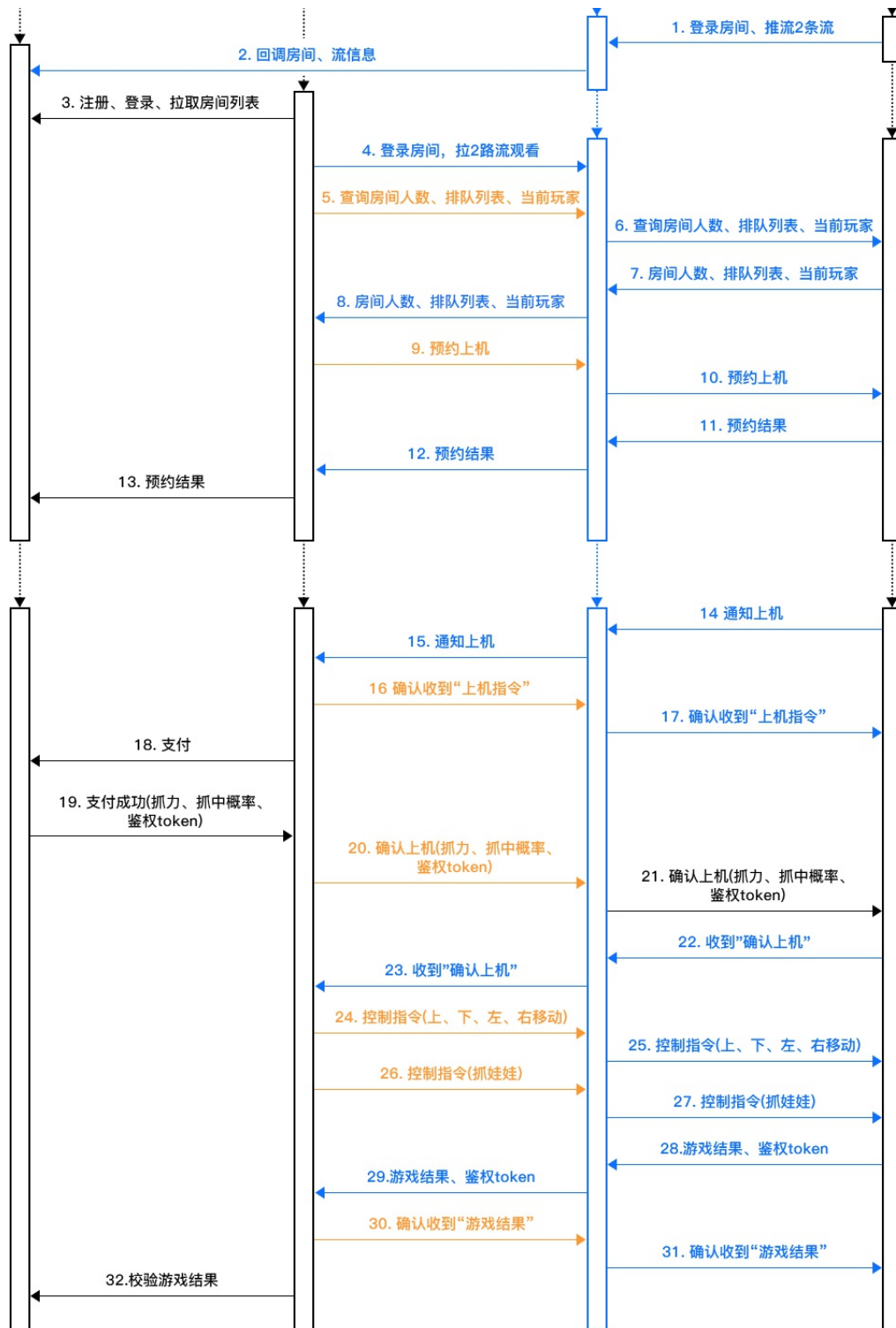
1、安装娃娃机控制端APK到安卓板子上

2、娃娃机启动，推流成功后，Zego后台会给业务后台POST 流创建的相关信息(业务后台提供回调地址)，用于业务侧维护娃娃机列表

3、客户端的开发

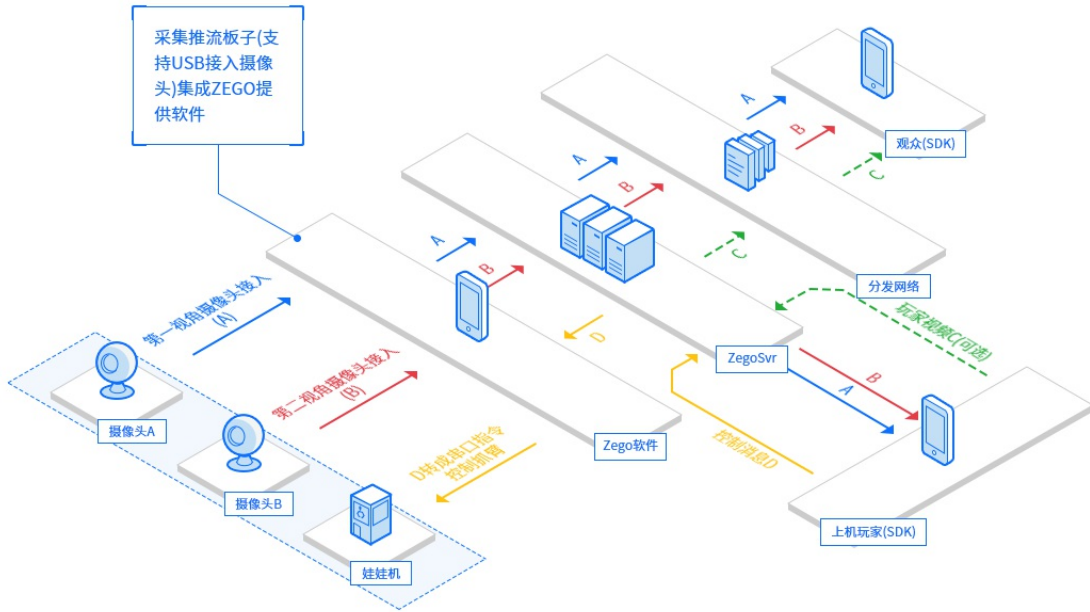
娃娃机系统实现流程如下图所示。该方案中，**娃娃机控制端** 无需与业务后台直接通信。





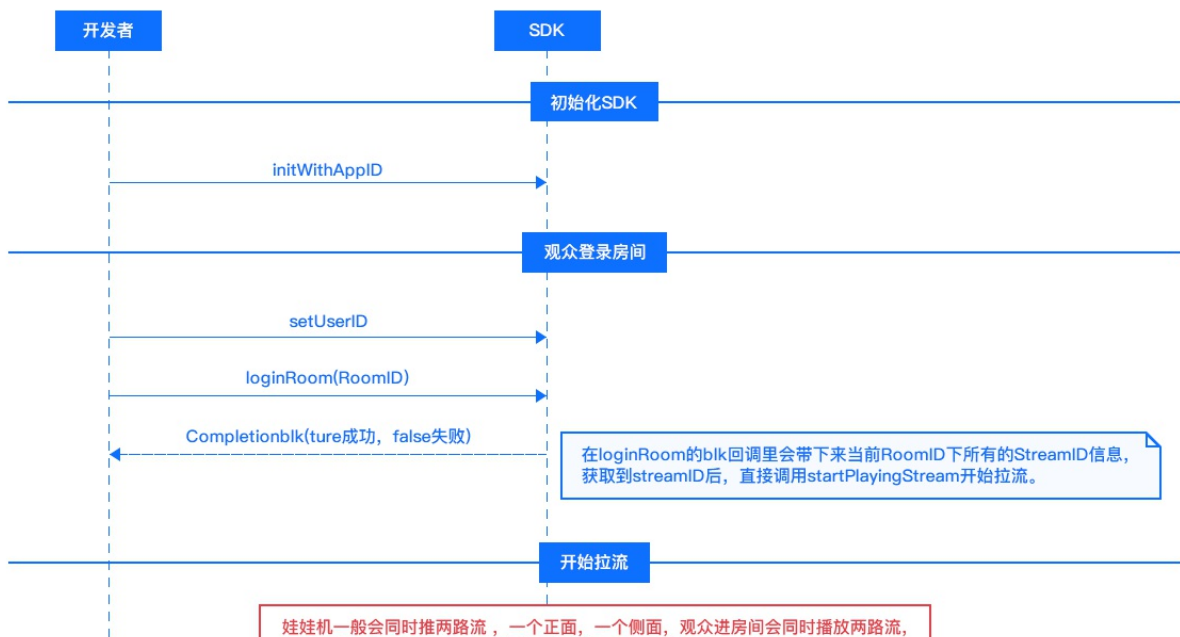
## 2 系统架构图

这里只需要两个普通的摄像头USB接入到安卓板子即可

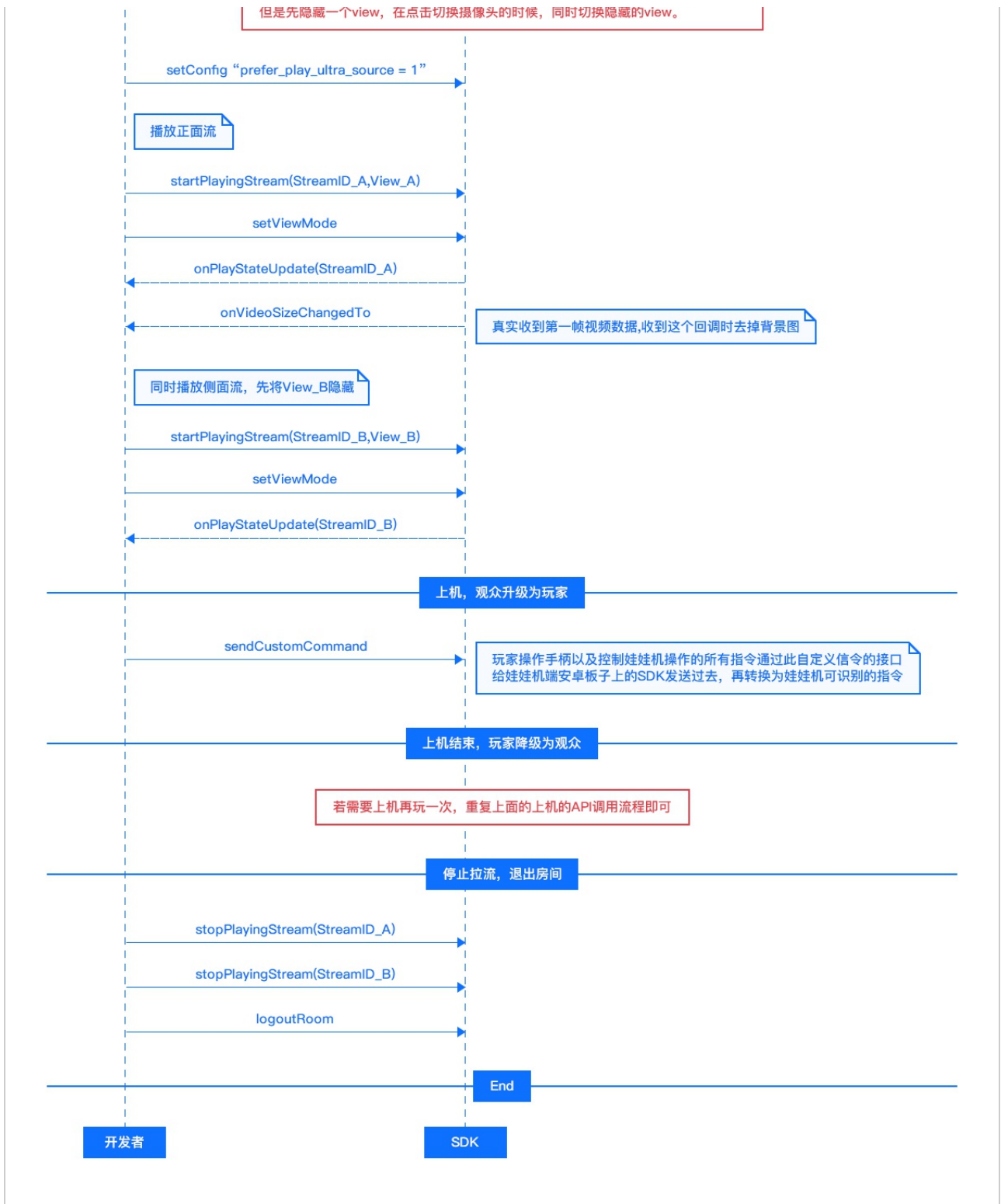


### 3 API 调用时序图

娃娃机用户端API调用流程图



娃娃机一般会同时推两路流，一个正面，一个侧面，观众进房间会同时播放两路流，



## 4 实现流程

### 4.1 登录房间

代码如下，供参考：

```
// 登录房间
mZegoLiveRoom.loginRoom(mRoomID, mRoomName, ZegoConstants.RoomRole.Anchor, new IZegoLoginCompletionCallback() {
    @Override
    public void onLoginCompletion(int errorCode, ZegoStreamInfo[] zegoStreamInfos) {
        if(errorCode == 0) {
            printLog("登录成功");
        } else {
            printLog("登录失败");
        }
    }
});
```

注意：以下所有操作必须在 **登录房间成功** 后才能进行

## 4.2 拉流

登录房间成功后，建议直接从 **ZEGO** 服务器拉流。这种方案 **画面延迟更小，体验效果更佳**。

代码如下，供参考：

```
// 设置开关，直接从 ZEGO 服务器拉流
String config = ZegoConstants.Config.PREFER_PLAY_ULTRA_SOURCE + "=1";
ZegoLiveRoom.setConfig(config);

// 拉第一路流
mZegoLiveRoom.startPlayingStream(mListStream.get(0).streamID, mListStream.get(0).textureView);
mZegoLiveRoom.setViewMode(ZegoVideoViewMode.ScaleAspectFit, mListStream.get(0).streamID);

// 拉第二路流
mZegoLiveRoom.startPlayingStream(mListStream.get(1).streamID, mListStream.get(1).textureView);
mZegoLiveRoom.setViewMode(ZegoVideoViewMode.ScaleAspectFit, mListStream.get(1).streamID);
```

请注意：

1. 目前 **娃娃机客户端** 使用的方案为：进入房间后，创建两个 **view** 分别播放两条视频流，用户可通过手动切换 **view**，继而切换当前可见视频流。

### 4.3 上机

如果 **娃娃机客户端** 设置了从 **ZEGO** 服务器拉流(参考 4.2 节)，当用户选择上机(玩游戏)时，**娃娃机客户端** 不需要进行流的切换，仅需要处理业务信令、UI 交互等。

### 4.4 发送指令

上机成功后，**娃娃机客户端** 需要调用 **ZegoLiveRoom SDK** 提供的 `sendCustomCommand` 函数发送指令给 **娃娃机控制端**，**娃娃机控制端** 收到指令后做相应的处理。

请注意，此处指令是发送给娃娃机控制端，而不是房间里的其他玩家或自己。

代码如下，供参考：

```
// 预约上机
String cmd = "{\"seq\":1,\"cmd\":513,\"data\":{\"config\":\"+W7TSJ/Vm9ohaDL9SmeA2
CJ9RTNdZj7LJk7VV56M1M6u31V0s/S7gCrWe89ZWseTjWYmnD6HouF1CERwkXMq73UCwnpa
LyCnlX+2glnqB4gWdS5Flgtar1IwmCY9t0ZguaRT0kMDHEgtKAw9Lcdj0nZ30n213zNt8mo9
KuStyU\u003d\",\"time_stamp\":1512532685017,\"continue\":0}}";

mZegoLiveRoom.sendCustomCommand(mAnchors, cmd, new IZegoCustomCommandCal
lback() {
    @Override
    public void onSendCustomCommand(int errorCode, String roomId) {
        if (errorCode == 0) {
            printLog("发送成功");
        } else {
            printLog("发送失败");
        }
    }
});
```

娃娃机客户端 与 娃娃机控制端 信令交互流程请参考：[娃娃机-信令交互](#)

## 4.5 接收指令

娃娃机客户端 可通过 `onRecvCustomCommand` 回调接收 娃娃机控制端 返回的指令。

设置回调的代码如下，供参考：



```
mZegoLiveRoom.setZegoRoomCallback(new IZegoRoomCallback() {
    @Override
    public void onRecvCustomCommand(String userID, String userName, String
content, String roomID) {
        // 处理接收到的指令
    }

    // 其他回调
})
```

娃娃机客户端 与 娃娃机控制端 信令交互流程请参考：[娃娃机-信令交互](#)

## 4.6 下机

由于用户下机后，可能继续停留在房间内围观，或者进行下一轮游戏。因此，建议 **娃娃机客户端** 在用户下机后，继续保持从 **ZEGO** 服务器拉流，维持画面和操作的低延迟效果。

## 4.7 退出房间

如果用户不再进行游戏，退出当前的游戏窗口。代码如下，供参考：

```
// 停止拉第一条流
mZegoLiveRoom.stopPlayingStream(mListStream.get(0).streamID);
// 停止拉第二条流
mZegoLiveRoom.stopPlayingStream(mListStream.get(1).streamID);
// 退出房间
mZegoLiveRoom.logoutRoom();
```