

# BananaPi BPi-R4 eMMC 烧写详细教程

## 概述

BananaPi BPi-R4 是一款强大的开发板，配备四核 ARM A73 处理器、4GB RAM、双 10GbE 光纤端口、四个 1GbE 以太网端口、Wi-Fi 7 连接、8GB 内部存储、NVMe 插槽和 USB 3.0 支持。将 OpenWrt 安装到 eMMC 需要特定的步骤，因为 SD 卡和 eMMC 不能同时访问，所以通常需要先安装到 NAND，然后从 NAND 复制到 eMMC。

## 硬件准备

### 必需设备

- BananaPi BPi-R4 开发板
- USB 转 TTL UART 转换器（用于串口调试）
- SD 卡（至少 8GB）
- 电源适配器
- 网线
- 电脑（用于烧写和调试）

### 可选设备

- Wi-Fi 7 模块（如需使用无线功能）
- NVMe SSD（扩展存储）
- 散热器和风扇

## 开关设置说明

BPi-R4 有两个启动开关，用于选择启动设备：

- 开关 1 和 2 都向下：从 SD 卡启动
- 开关 1 向上，开关 2 向下：从 NAND 启动
- 开关 1 向下，开关 2 向上：从 eMMC 启动

## 安装步骤

方法一：标准安装流程（SD 卡 → NAND → eMMC）

步骤 1：准备 SD 卡镜像

1. 访问 OpenWrt 固件选择器下载最新快照版本（**注意**：如果是准备安装固件到 eMMC，固件大小不要超过 NAND 容量）
2. 使用 balenaEtcher 或 dd 命令将镜像写入 SD 卡
3. 将两个开关都设置为向下位置
4. 插入 SD 卡并启动设备，至此完成 SD 卡安装，如果接受在 SD 卡运行固件，以下步骤可省略

步骤 2：连接串口

1. 连接 USB 转 TTL UART 转换器：
  - RX 连接到 TX
  - TX 连接到 RX
  - GND 连接到 GND
2. 使用串口工具（如 minicom）连接：

```
sudo minicom -s
```

设置波特率为 115200

步骤 3：安装到 NAND

1. 关机上电，启动后在串口界面迅速按键盘按方向键下，以免引导过程很短错过进入 U-Boot 菜单，U-Boot 菜单中选择选项 7
2. 等待安装完成后按 ENTER 并关闭电源并取出 SD 卡
3. 将开关 1 设置为向上位置，开关 2 保持向下

4. 重新启动设备，至此完成安装固件到 NAND ，如果接受在 NAND 运行固件，以下步骤可忽略，但由于 NAND 空间较小，通常是作为安装固件到 eMMC 的过渡或者保存一个纯净系统确保有系统可运行

#### 步骤 4: 从 NAND 安装到 eMMC

1. 关机上电，启动后在串口界面迅速按键盘按方向键下，以免引导过程很短错过进入 U-Boot 菜单，设备从 NAND 启动后，U-Boot 菜单会显示 9 号安装到 eMMC 的选项
2. 选择该选项并等待安装完成
3. 安装完成后按 ENTER 并关闭电源
4. 将开关 1 设置为向下，开关 2 设置为向上
5. 重新启动，系统将从 eMMC 启动 ，至此完成将系统安装到 eMMC

#### 方法二：直接安装到 eMMC（跳过 NAND）

如果您想跳过 NAND 步骤直接安装到 eMMC，可以使用以下方法：

1. 准备一个 USB 存储设备，将 eMMC 镜像文件复制到其中
2. 从预装的 NAND OpenWrt 启动
3. 插入 USB 设备并挂载：

```
mkdir -p /mnt/usb
mount /dev/sda1 /mnt/usb
```
4. 使用 dd 命令将镜像写入 eMMC：

```
dd if=/mnt/usb/openwrt-emmc.img of=/dev/mmcblk0 bs=1M
```
5. 完成后重启并调整开关位置

## 初始配置

### SSH 连接和基础设置

1. 通过 SSH 连接到设备：

```
ssh root@192.168.1.1
```
2. 更新软件包列表：

```
opkg update
```
3. 安装必要的软件包：

```
opkg install block-mount e2fsprogs kmod-usb-storage-uas kmod-usb3 luci kmod-fs-ext4 kmod-nvme
```
4. 重启设备：

```
reboot
```
5. 通过浏览器访问 <http://192.168.1.1> 并设置 root 密码

### 扩展存储配置

由于 eMMC 安装后可用空间可能只有约 400MB，建议使用 NVMe SSD 扩展存储：

1. 安装 NVMe SSD 到 M.2 插槽
2. 格式化为 ext4 文件系统
3. 挂载并配置为额外存储空间

## 常见问题解决

1. 无法从 eMMC 启动
  - 检查开关位置是否正确（开关 1 向下，开关 2 向上）
  - 确认 eMMC 镜像写入成功
  - 尝试重新执行安装流程
2. 串口无输出
  - 检查 TX/RX 连接是否正确（注意交叉连接）
  - 确认波特率设置为 115200
  - 检查 USB 转 TTL 转换器驱动是否正确安装
3. 存储空间不足
  - 安装 NVMe SSD 扩展存储
  - 清理不必要的软件包

- 使用外部 USB 存储设备

## 注意事项

1. 重要：SD 卡和 eMMC 不能同时访问，这是硬件限制
2. 安装过程中请勿断电，可能导致设备变砖
3. 建议在操作前备份重要配置
4. Wi-Fi 7 模块需要额外安装并开启相应开关
5. 设备空闲功耗约 9W（包含 Wi-Fi 7 模块、NVMe 和主动散热）

## 进阶配置

### Wi-Fi 7 模块安装

1. 像安装双槽 NVMe 驱动器一样插入 Wi-Fi 7 模块
2. 拧紧螺丝并连接天线
3. 贴上散热片
4. 记得打开 Wi-Fi 开关

### 性能优化

- 根据需求调整 CPU 频率策略
- 配置适当的内存管理参数
- 优化网络接口设置以充分利用 10GbE 端口

## 分区布局和存储管理

### eMMC 分区结构

BPi-R4 的 eMMC 默认分区布局如下：

Device	Start	End	Sectors	Size	Type
/dev/mmcbk0p1	34	8192	8158	4M	unknown
/dev/mmcbk0p2	8192	16383	8192	4M	unknown
/dev/mmcbk0p3	16384	24575	8192	4M	unknown
/dev/mmcbk0p4	24576	90111	65536	32M	unknown
/dev/mmcbk0p5	90112	1138687	1048576	512M	Linux filesystem

### 解决 overlay 空间问题

许多用户发现即使使用 8GB eMMC，/overlay 分区只有约 431MB 可用空间。这是因为默认镜像的分区大小是固定的。解决方法：

1. 扩展现有分区：

```
# 查看当前分区
```

```
fdisk -l /dev/mmcbk0
```

```
# 使用 parted 扩展分区
```

```
parted /dev/mmcbk0
```

```
(parted) resizepart 5 100%
```

```
(parted) quit
```

```
# 扩展文件系统
```

```
resize2fs /dev/mmcbk0p5
```

## 2. 重新分区（高级用户）:

```
# 备份重要数据后
fdisk /dev/mmcblk0

# 删除并重新创建分区 5, 使用全部剩余空间
```

## 从串口控制台安装

### 使用 USB 存储设备安装

如果需要从串口控制台直接安装到 NAND 或 eMMC, 可以使用以下方法:

#### 1. 准备 USB 设备:

```
# 在另一台电脑上准备 USB 设备

# 格式化为 ext4 或 FAT32
mkfs.ext4 /dev/sdX1
```

```
# 复制镜像文件
mount /dev/sdX1 /mnt
cp openwrt-*-emmc.img /mnt/
umount /mnt
```

#### 2. 从串口控制台操作:

```
# 插入 USB 到 BPi-R4

# 查找 USB 设备
dmesg | grep sd

# 挂载 USB
mkdir -p /mnt/usb
mount /dev/sda1 /mnt/usb

# 写入镜像
dd if=/mnt/usb/openwrt-*-emmc.img of=/dev/mmcblk0 bs=4M status=progress
sync
```

#### 3. 验证安装:

```
# 检查分区表
fdisk -l /dev/mmcblk0

# 重启前调整启动开关
reboot
```

## 高级启动选项

### U-Boot 环境变量配置

BPi-R4 使用 U-Boot 作为引导程序，可以通过串口访问 U-Boot 菜单进行高级配置：

1. 进入 U-Boot 控制台：
  - 开机时快速按任意键中断启动
  - 输入 `help` 查看可用命令
2. 查看当前环境变量：  
`printenv`
3. 修改启动参数：  
`setenv bootargs 'console=ttyS0,115200 root=/dev/mmcblk0p5 rootwait'`  
`saveenv`

### 多系统启动配置

BPi-R4 支持在不同存储介质上安装多个系统：

1. SD 卡：用于测试和恢复
2. NAND：用于备用系统或最小化系统
3. eMMC：主要运行系统
4. NVMe：可以安装其他 Linux 发行版

## 网络配置优化

### 10GbE 光纤端口配置

# 编辑网络配置

`vi /etc/config/network`

# 添加 10G 接口配置

```
config interface 'sfp1'
    option device 'eth1'
    option proto 'static'
    option ipaddr '10.0.0.1'
    option netmask '255.255.255.0'
```

```
config interface 'sfp2'
    option device 'eth2'
    option proto 'dhcp'
```

### 性能调优

1. 启用硬件加速：
  - # 安装硬件加速包  
`opkg install kmod-crypto-hw-safexcel`
  - # 启用流量加速  
`echo 1 > /sys/kernel/debug/mtk_ppe/bind_rate`

2. CPU 频率管理：

# 查看当前 CPU 频率

```
cat /sys/devices/system/cpu/cpu*/cpufreq/scaling_cur_freq
```

# 设置性能模式

```
echo performance > /sys/devices/system/cpu/cpu0/cpufreq/scaling_governor
```

# 故障排除进阶

## 串口调试技巧

### 1. 启用详细日志:

```
# 修改内核启动参数

# 在 U-Boot 中添加 loglevel=7
setenv bootargs '${bootargs} loglevel=7'
```

### 2. 捕获启动日志:

```
# 使用 minicom 记录日志

minicom -C boot.log
```

## 恢复模式

如果系统无法启动，可以使用以下恢复方法:

### 1. 使用 SD 卡恢复:

- 准备一个可启动的 SD 卡
- 调整开关到 SD 卡启动位置
- 启动后重新安装系统

### 2. TFTP 恢复 (需要网络):

```
# 在 U-Boot 中
setenv ipaddr 192.168.1.10
setenv serverip 192.168.1.100
tftpboot 0x46000000 openwrt-recovery.img
mmc write 0x46000000 0 0x20000
```

# 安全加固

## 基础安全设置

### 1. 更改默认密码:

```
passwd root
```

### 2. 配置防火墙规则:

```
# 编辑防火墙配置
```

```
vi /etc/config/firewall
```

```
# 添加自定义规则
```

```
config rule
    option name 'Block-SSH-WAN'
    option src 'wan'
    option dest_port '22'
    option target 'DROP'
```

### 3. 启用 SSH 密钥认证:

```
# 生成密钥对 (在客户端)
```

```
ssh-keygen -t ed25519
```

```
# 复制公钥到路由器
```

```
ssh-copy-id root@192.168.1.1
```

```
# 禁用密码登录

uci set dropbear.@dropbear[0].PasswordAuth='0'
uci commit dropbear
/etc/init.d/dropbear restart
```

## 监控和维护

### 系统监控

1. 安装监控工具：  
opkg install htop iftop vnstat
2. 温度监控：

```
# 查看 CPU 温度

cat /sys/class/thermal/thermal_zone*/temp
```

```
# 创建温度监控脚本

cat > /root/temp_monitor.sh << 'EOF'
#!/bin/sh
while true; do
    temp=$(cat /sys/class/thermal/thermal_zone0/temp)
    echo "CPU Temperature: $((temp/1000))° C"
    sleep 5
done
EOF
chmod +x /root/temp_monitor.sh
```

### 定期维护

1. 自动备份配置：

```
# 创建备份脚本

cat > /etc/backup.sh << 'EOF'
#!/bin/sh
DATE=$(date +%Y%m%d)
sysupgrade -b /tmp/backup-{DATE}.tar.gz
scp /tmp/backup-{DATE}.tar.gz user@backup-server:/path/to/backups/
EOF
chmod +x /etc/backup.sh
```

```
# 添加到 cron
echo "0 3 * * 0 /etc/backup.sh" >> /etc/crontabs/root
```

2. 日志轮转：

```
# 配置 logrotate
opkg install logrotate
```

这些高级配置和优化技巧可以帮助您充分发挥 BPi-R4 的性能潜力，并确保系统稳定运行。根据实际需求选择合适的配置选项。