

Rockchip OTP 开发指南

文件标识: RK-KF-YF-147

发布版本: V1.0.1

日期: 2021-02-08

文件密级: ☐绝密 ☐秘密 ☐内部资料 ☒公开

免责声明

本文档按“现状”提供, 瑞芯微电子股份有限公司(“本公司”, 下同)不对本文档的任何陈述、信息和内容的准确性、可靠性、完整性、适销性、特定目的性和非侵权性提供任何明示或暗示的声明或保证。本文档仅作为使用指导的参考。

由于产品版本升级或其他原因, 本文档将可能在未经任何通知的情况下, 不定期进行更新或修改。

商标声明

“Rockchip”、“瑞芯微”、“瑞芯”均为本公司的注册商标, 归本公司所有。

本文档可能提及的其他所有注册商标或商标, 由其各自拥有者所有。

版权所有 © 2021 瑞芯微电子股份有限公司

超越合理使用范畴, 非经本公司书面许可, 任何单位和个人不得擅自摘抄、复制本文档内容的部分或全部, 并不得以任何形式传播。

瑞芯微电子股份有限公司

Rockchip Electronics Co., Ltd.

地址: 福建省福州市铜盘路软件园A区18号

网址: www.rock-chips.com

客户服务电话: +86-4007-700-590

客户服务传真: +86-591-83951833

客户服务邮箱: fae@rock-chips.com

前言

概述

本文档主要介绍 Rockchip OTP OEM 区域烧写。

产品版本

芯片名称	内核版本
RK 系列芯片	Linux 4.19

读者对象

本文档（本指南）主要适用于以下工程师：

技术支持工程师

软件开发工程师

修订记录

版本号	作者	修改日期	修改说明
V1.0.0	张学广	2020-10-18	初始版本
V1.0.1	张学广	2021-02-08	格式修订

目录

Rockchip OTP 开发指南

1. 概述
2. OTP Layout
 - 2.1 RV1126/RV1109
3. OEM Zone
 - 3.1 OEM Read
 - 3.2 OEM Write
 - 3.3 Demo

1. 概述

OTP NVM (One Time Programmable Non-Volatile Memory)，即只可编程一次的非易失性存储。作为对比，FLASH 存储可多次擦写。

2. OTP Layout

RK 平台 OTP Layout 结构基本相同，大小和偏移因芯片而异。

2.1 RV1126/RV1109

RV1126/RV1109 OTP 布局如表 1-1 所示：

Type	Range [bytes]	Description
SYSTEM	0x000 ~ 0x0FF	system info, read only
OEM	0x100 ~ 0x1EF	oem zone for customized
RESERVED	0x1F0 ~ 0x1F7	reserved
WP	0x1F8 ~ 0x1FF	write protection for oem zone

表 1-1 RV1126/RV1109 OTP Layout

3. OEM Zone

RK 平台 OTP 预留 OEM 区域，方便客户存储自定义数据，比如：序列号，MAC 地址，产品信息等。通过标准文件读写 API 对 OEM 区域进行读写。参考 [OTP Layout](#) 查询各芯片平台 OEM 支持情况。比如：RV1126的 OTP_OEM_OFFSET 为 0x100，RANGE 为 0x100 ~ 0x1EF，TOTAL SIZE 为 240 bytes。

3.1 OEM Read

```
/*
 * @offset: offset from oem base
 * @buf: buf to store data which read from oem
 * @len: data len in bytes
 */
int rockchip_otp_oem_read(int offset, char *buf, int len)
{
    int fd = 0, ret = 0;
```

```

    fd = open("/sys/bus/nvmem/devices/rockchip-otp0/nvmem", O_RDONLY);
    if (fd < 0)
        return -1;

    ret = lseek(fd, OTP_OEM_OFFSET + offset, SEEK_SET);
    if (ret < 0)
        goto out;

    ret = read(fd, buf, len);
out:
    close(fd);

    return ret;
}

```

3.2 OEM Write

1, 每笔 OEM Write 前都需要使能写开关, 目的是避免误写。

```

int rockchip_otp_enable_write(void)
{
    char magic[] = "1380926283";
    int fd, ret;

    fd = open("/sys/module/nvmem_rockchip_otp/parameters/rockchip_otp_wr_magic",
O_WRONLY);
    if (fd < 0)
        return -1;

    ret = write(fd, magic, 10);
    close(fd);

    return ret;
}

```

2, 写入的数据大小及偏移需要4字节对齐, 数据写入后将被标记写保护, 相应数据写保护将在下次重启后生效。

```

/*
 * @offset: offset from oem base, MUST be 4 bytes aligned
 * @buf: data buf for write
 * @len: data len in bytes, MUST be 4 bytes aligned
 */
int rockchip_otp_oem_write(int offset, char *buf, int len)
{
    int fd = 0, ret = 0;

    /* MUST be 4 bytes aligned */
    if (len % 4)
        return -1;

    fd = open("/sys/bus/nvmem/devices/rockchip-otp0/nvmem", O_WRONLY);
    if (fd < 0)

```

```

        return -1;

    ret = lseek(fd, OTP_OEM_OFFSET + offset, SEEK_SET);
    if (ret < 0)
        goto out;

    ret = write(fd, buf, len);
out:
    close(fd);

    return ret;
}

```

3.3 Demo

1, OEM 区域 偏移0的位置写入 0~15

```

void demo(void)
{
    char buf[16] = { 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15 };
    int ret = 0;

    ret = rockchip_otp_enable_write();
    if (ret < 0)
        return ret;

    rockchip_otp_oem_write(0, buf, 16);
}

```

2, 通过 [OEM Read](#) 或者 hexdump 命令查看结果，如下为通过命令查看 OEM 区域数据

```

# hexdump -C /sys/bus/nvmem/devices/rockchip-otp0/nvmem
00000000  52 56 11 26 91 fe 21 4b  50 41 30 31 37 00 00 00
00000010  00 00 00 00 10 25 16 12  2f 0e 0f 00 08 00 00 00
00000020  00 00 00 e0 0a e0 0a 1e  00 00 00 00 00 00 00 00
00000030  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
*
00000100  00 01 02 03 04 05 06 07  08 09 0a 0b 0c 0d 0e 0f
00000110  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
*
000001e0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
000001f0  00 00 00 00 00 00 00 00  0f 00 00 00 00 00 00 00

```