



# **XRADIO Flash Layout Developer Guide**

---

**Revision 1.0**

**Nov 22, 2019**

## Declaration

THIS DOCUMENTATION IS THE ORIGINAL WORK AND COPYRIGHTED PROPERTY OF XRADIO TECHNOLOGY ("XRADIO"). REPRODUCTION IN WHOLE OR IN PART MUST OBTAIN THE WRITTEN APPROVAL OF XRADIO AND GIVE CLEAR ACKNOWLEDGEMENT TO THE COPYRIGHT OWNER.

THE PURCHASED PRODUCTS, SERVICES AND FEATURES ARE STIPULATED BY THE CONTRACT MADE BETWEEN XRADIO AND THE CUSTOMER. PLEASE READ THE TERMS AND CONDITIONS OF THE CONTRACT AND RELEVANT INSTRUCTIONS CAREFULLY BEFORE USING, AND FOLLOW THE INSTRUCTIONS IN THIS DOCUMENTATION STRICTLY. XRADIO ASSUMES NO RESPONSIBILITY FOR THE CONSEQUENCES OF IMPROPER USE (INCLUDING BUT NOT LIMITED TO OVERVOLTAGE, OVERCLOCK, OR EXCESSIVE TEMPERATURE).

THE INFORMATION FURNISHED BY XRADIO IS PROVIDED JUST AS A REFERENCE OR TYPICAL APPLICATIONS, ALL STATEMENTS, INFORMATION, AND RECOMMENDATIONS IN THIS DOCUMENT DO NOT CONSTITUTE A WARRANTY OF ANY KIND, EXPRESS OR IMPLIED. XRADIO RESERVES THE RIGHT TO MAKE CHANGES IN CIRCUIT DESIGN AND/OR SPECIFICATIONS AT ANY TIME WITHOUT NOTICE.

NOR FOR ANY INFRINGEMENTS OF PATENTS OR OTHER RIGHTS OF THE THIRD PARTIES WHICH MAY RESULT FROM ITS USE. NO LICENSE IS GRANTED BY IMPLICATION OR OTHERWISE UNDER ANY PATENT OR PATENT RIGHTS OF XRADIO. THIRD PARTY LICENCES MAY BE REQUIRED TO IMPLEMENT THE SOLUTION/PRODUCT. CUSTOMERS SHALL BE SOLELY RESPONSIBLE TO OBTAIN ALL APPROPRIATELY REQUIRED THIRD PARTY LICENCES. XRADIO SHALL NOT BE LIABLE FOR ANY LICENCE FEE OR ROYALTY DUE IN RESPECT OF ANY REQUIRED THIRD PARTY LICENCE. XRADIO SHALL HAVE NO WARRANTY, INDEMNITY OR OTHER OBLIGATIONS WITH RESPECT TO MATTERS COVERED UNDER ANY REQUIRED THIRD PARTY LICENCE.

## Revision History

Version	Date	Summary of Changes
1.0	2019-11-22	Initial Version

**Table 1- 1 Revision History**

## Contents

Declaration.....	2
Revision History.....	3
Contents.....	4
Figures.....	5
1 概述.....	6
1.1 Flash 布局.....	6
1.2 Image 1.....	6
1.3 Padding.....	7
1.4 Image 2/compressed image.....	7
1.4.1 ping pong 模式下.....	7
1.4.2 压缩模式下.....	7
1.5 OTA area.....	7
1.6 Sysinfo 区.....	8
2 Cfg 文件.....	9
2.1 Flash_offs 配置.....	10
2.2 Bin 文件重叠的解决.....	10
2.3 配置自定义 cfg 文件.....	11

## Figures

图 1-1	Flash 布局.....	6
图 2-1	出现重叠情况.....	10
图 2-2	配置自定义 cfg 文件.....	11

# 1 概述

此文档用以从较为宏观的角度来说明 flash 所使用的布局情况，主要介绍整个 image 和 sysinfo 以及 OTA 区域之间的位置关系。可以阅读《XRADIO Image Developer Guide》了解 image 模块代码上的使用，同时也可以阅读《XRADIO\_OTA\_Developer\_Guide-CN》了解 OTA 升级的不同模式。

## 1.1 Flash 布局

Flash 布局和 OTA 升级的模式有关，ping pong 模式下，flash 占用会比在压缩模式下大，两种模式下 flash 的布局如下图所示。

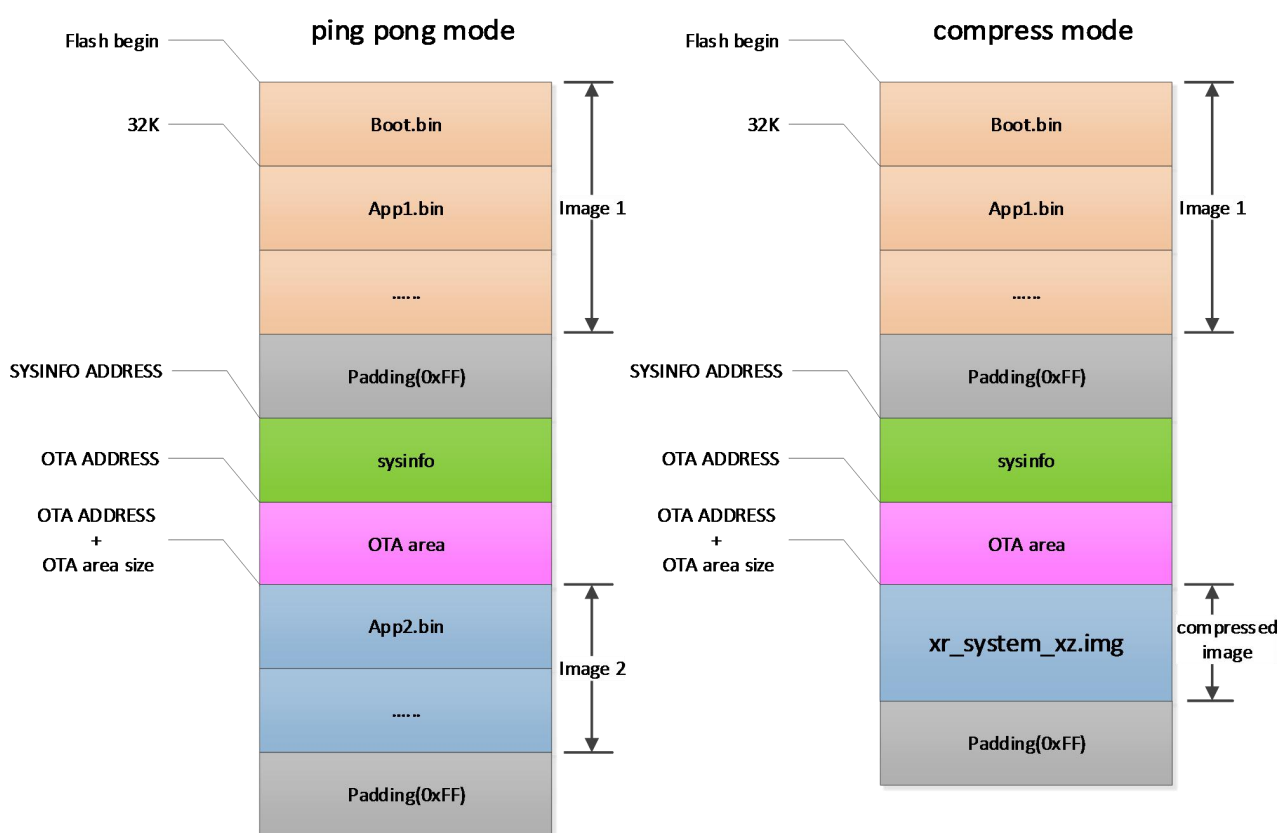


图 1-1 Flash 布局

## 1.2 Image 1

Image 1 是一个完整的镜像，里面包含了 boot.bin、app.bin...xxx.bin，从 flash 的 0 地址开始。

## 1.3 Padding

padding 是后期给 OTA 升级预留的空间。后期代码增加，功能扩展，会导致 image 增加，所以需要给后期升级扩展预留一定的空间。padding 的最大值需要在项目开发前确定好，即需要先确定好 image 的最大值，否则容易出现后期开发扩展时，image 超过最大值而覆盖 sysinfo 数据。假设正常编译的 image 是 900K（包含 bootloader），则可定义 image 的最大值为 1020K，后期开发的 image 就不能超过 1020K。

## 1.4 Image 2/compressed image

### 1.4.1 ping pong 模式下

Image 2 是不包含 boot.bin 的，其起始位置决定于 OTA 的起始位置和 OTA 区域大小，所以 image 2 的 app.bin 是放在 OTA 的起始位置加 OTA 区域大小后。

### 1.4.2 压缩模式下

压缩模式下，flash 的分布和 ping pong 模式下大部分相同，唯一不同的就是 Image 2。ping pong 模式下 Image 2 存放的是一个可运行的 image（去除了 bootloader 的 image）；而压缩模式下，Image 2 存放的是一个压缩的 image。所以在压缩模式下，Image 2 需要的空间会比 ping pong 模式下小很多。

## 1.5 OTA area

OTA area 用于存储 OTA 使用的参数，该区域的地址 OTA ADDRESS 的配置是在 cfg 文件中指定的，默认是指定为 1024K，即 1M 的偏移地址，大小为 4K。如下标红位置即配置 OTA ADDRESS 和大小；

```
{
    "magic"    : "AWIH",
    "version"  : "0.2",
    "OTA"      : {"addr": "1024K", "size": "4K"},
    #if (__CONFIG_OTA_POLICY == 0x01)
    "image"    : {"max_size": "1020K", "xz_max_size": "600K"},
    #else
    "image"    : {"max_size": "1020K"},
    #endif
    "count"    : 7,
    "section"  : [
        {"id": "0xa5ff5a00", "bin": "boot.bin", "cert": "null", "flash_offs":
"0K", "sram_offs": "0x00067000", "ep": "0x00067101", "attr": "0x1"},
        .....
    ]
}
```

需要注意 OTA 地址与 flash 大小之间的关系：

1. 如果没有开启 OTA 功能，则用户只需要保证 flash 大小大于 image 大小即可，因为此时，OTA ADDRESS 之后的区域都不存在，flash 中只有 image1、padding 和 sysinfo。

2. 如果开启 OTA 功能，则用户需要保证在 OTA ADDRESS 之后仍有一个足够的空间进行 OTA 升级用，该空间需要保证能够存下一个 image 或一个压缩的 image。假设一个 image 大小为 900K（包含 bootloader），image 的最大值定义为 1020K，flash 大小为 4M，那么 OTA 地址设置在 1M 到 3M 之间的区域都是可以的，因为 OTA 地址之后还有足够的空间来存放 image。

## 1.6 Sysinfo 区

Sysinfo 是一段用于存储用户自定义数据的区域，其大小默认被设置为 4K，位置被设置为 1020K（1024K - 4K）的地址，默认是放在 OTA 区域之前的位置。Sysinfo 的位置和大小是被定义在工程目录下的 prj\_config.h 文件中，用户可根据需求进行修改，但是需要注意不能与 image 区域和 OTA 区域重叠。

```
/* sysinfo start address */
#define PRJCONF_SYSINFO_ADDR          ((1024 - 4) * 1024)

/* sysinfo size */
#define PRJCONF_SYSINFO_SIZE          (4 * 1024)
```



## 2 Cfg 文件

Cfg 文件包含整个 image 的组织结构，配置文件使用 Json 标记语言对 bin 文件的头部信息进行配置。使用 bin 文件添加头部的功能时，配置文件也是相同的。软件只为第一个 img 的文件增加头部。

如下是一个完整的 cfg 文件，标为红色的部分配置了每一个 bin 文件的起始 flash 偏移地址。以下是一个经过预处理的 cfg 文件，详细的 cfg 文件可参考 image.cfg。

```
{
  "magic" : "AWIH",
  "version" : "0.4",
  "OTA" : {"addr": "1024K", "size": "4K"},
  "image" : {"max_size": "1020K"},
  "count" : 7,
  "section" :
  [
    {"id": "0xa5ff5a00", "bin": "boot_40M.bin", "cert": "null", "flash_offs":
    "0K", "sram_offs": "0x00268000", "ep": "0x00268101", "attr": "0x1"},
    {"id": "0xa5fe5a01", "bin": "app.bin", "cert": "null", "flash_offs":
    "32K", "sram_offs": "0x00201000", "ep": "0x00201101", "attr": "0x1"},
    {"id": "0xa5fd5a02", "bin": "app_xip.bin", "cert": "null", "flash_offs":
    "80K", "sram_offs": "0xffffffff", "ep": "0xffffffff", "attr": "0x2"},
    {"id": "0xa5f65a09", "bin": "app_psram.bin", "cert": "null", "flash_offs":
    "900K", "sram_offs": "0x01400000", "ep": "0x00000000", "attr": "0x1"},
    {"id": "0xa5fa5a05", "bin": "wlan_bl.bin", "cert": "null", "flash_offs":
    "980K", "sram_offs": "0xffffffff", "ep": "0xffffffff", "attr": "0x1"},
    {"id": "0xa5f95a06", "bin": "wlan_fw.bin", "cert": "null", "flash_offs":
    "990K", "sram_offs": "0xffffffff", "ep": "0xffffffff", "attr": "0x1"},
    {"id": "0xa5f85a07", "bin": "wlan_sdd_40M.bin", "cert": "null",
    "flash_offs": "1015K", "sram_offs": "0xffffffff", "ep": "0xffffffff", "attr":
    "0x1"},
    {}
  ]
}
```

各字段含义如下：

magic - 固件魔数，表头标识，固定为 AWIH

version - 版本号

OTA - ota 参数存储地址和大小

image - image 的大小参数

max\_size : image 的最大值（image1 和 image2 区域的最大值）

xz\_max\_size : ota 压缩模式下 compressed image 的最大值，非 ota 压缩模式下，禁止使用该字段。

count - 打包文件列表中文件的数目

section - 文件列表

id - bin 文件的 id

boot\_id : 0xa5ff5a00 app\_id : 0xa5fe5a01 app\_xip\_id : 0xa5fd5a02 wlan\_bl\_id : 0xa5fa5a05

wlan\_fw\_id : 0xa5f95a06 wlan\_sdd\_id : 0xa5f85a07 app\_psrarn\_id : 0xa5f65a09

bin - bin 文件名字

cert - 该段固件对应的证书文件，null 表示没有添加证书

flash\_offs - 该段固件存放在 FLASH 中的位置偏移

sram\_offs - 该段固件加载到 SRAM 中的地址偏移，0xFFFFFFFF 为无效值，使用时忽略

ep - 该段固件的 ENTRY POINT，0xFFFFFFFF 为无效值，使用时忽略

attr - 表示该段固件属性

[1:0]: 文件类型，00-普通文件，01-可执行文件，10-XIP 文件，11-保留

[2:1]: 加密类型，00-不加密，01-带证书，1X-保留

[7:3]: 保留

## 2.1 Flash\_offs 配置

boot.bin 固定需要设置为 0 地址，app.bin 固定需要设置为 32k 的偏移地址，其他的 bin 文件的地址并没有强制性的要求，只要没有互相重叠并且不覆盖到 sysinfo 和 OTA 区域，都是可以正常运行的；

## 2.2 Bin 文件重叠的解决

如果 bin 文件出现了互相重叠的情况，打包工具会进行自动计算，然后生成一个“image\_auto\_cal.cfg”文件，用户可以直接使用该 cfg 文件重新进行打包，也可手动修改默认的 image.cfg 文件。需要注意的是，如果全部的 bin 文件在经过自动计算后，其大小依然超过 sysinfo 的默认位置（OTA ADDRESS - 4K），则不会生成“image\_auto\_cal.cfg”文件。此时，用户需要自己手动进行 cfg 文件的修改。

```
cfg string:
{
  "magic" : "AWIH",
  "version" : "0.4",
  "OTA" : {"addr": "1024K", "size": "4K"},
  "image" : {"max_size": "1020K"},
  "count" : 6,
  "section" :
  [
    {
      "id": "0xa5ff5a00", "bin": "boot_40M.bin", "cert": "null", "flash_offs": "0K", "sram_offs": "0x00268000", "ep": "0x00268101", "attr": "0x1"},
      {"id": "0xa5fe5a01", "bin": "app.bin", "cert": "null", "flash_offs": "32K", "sram_offs": "0x00201000", "ep": "0x00201101", "attr": "0x1"},
      {"id": "0xa5fd5a02", "bin": "app_xip.bin", "cert": "null", "flash_offs": "50K", "sram_offs": "0xffffffff", "ep": "0xffffffff", "attr": "0x2"},
      {"id": "0xa5fa5a05", "bin": "wlan_bl.bin", "cert": "null", "flash_offs": "980K", "sram_offs": "0xffffffff", "ep": "0xffffffff", "attr": "0x1"},
      {"id": "0xa5f95a06", "bin": "wlan_fw.bin", "cert": "null", "flash_offs": "990K", "sram_offs": "0xffffffff", "ep": "0xffffffff", "attr": "0x1"},
      {"id": "0xa5f85a07", "bin": "wlan_sdd_40M.bin", "cert": "null", "flash_offs": "1015K", "sram_offs": "0xffffffff", "ep": "0xffffffff", "attr": "0x1"},
    }
  ]
}

err: bin 1 and bin 2 were overlaped!
Overlapped size: 1608 Byte(2KB)
bin 1 name:app.bin begin: 0x00008000 end: 0x0000CE48
bin 2 name:app_xip.bin begin: 0x0000C800

We've rearranged bin files and generated new cfg file 'image_auto_cal.cfg', the new one is recommended.
Generate image file failed
```

图 2-1 出现重叠情况

上图显示，app.bin 和 app\_xip.bin 重叠了，app.bin 超出了 2K 的数据，覆盖了 app\_xip.bin 前面 2K 数据。说明给 app.bin 的空间太小了，此时可以调整 app\_xip.bin 的 flash\_offs，由原来的 50K 调整为 80K 即可，剩下的 28K 做为预留空间。

## 2.3 配置自定义 cfg 文件

项目工程在默认时使用的 cfg 文件是 project/image\_cfg/image.cfg 文件，如果用户想要使用自己修改的 cfg 文件，可以在对应的项目工程目下的/gcc/Makefile 中添加自定义 cfg 路径。修改方式如下图：

```
72: # image config file
73: #   - relative to "../image/$(__CONFIG_CHIP_TYPE)/", eg. "../image/xr872/"
74: #   - define your own "IMAGE_CFG" to override the default one
75: IMAGE_CFG := ../image/xr872/
76:
77: # image name, default to xr_system
78: IMAGE_NAME := my_image.cfg
```

图 2-2 配置自定义 cfg 文件