

Android Device Layout

Broadcom Proprietary and Confidential

Release Note: This document can be shared with customers.

adl - Android Device Layout

This document explains how to setup a device layout for an android device using the offered configuration settings in the Broadcom reference device setup.

By default, the Broadcom device layout is based on a 8GB flash size which is considered to be the smallest suitable flash layout for ATV compliance; if the target device has a larger flash available, it is up to the customer to determine what best layout is suited for the needs, keeping in mind that the layout provided by Broadcom should not be reduced in any parts in order to ensure compliance with Android.

Foreword

This document focuses on Android O onward integration and how it can be adapted for a specific customer device.

This document does not describe how an existing device with an existing layout can be "field upgraded" to a different layout, regardless of the current and target versions running and intended.

- The "field upgrade" aspect of things can be far too complicated to implement and described here (due in particular for the need of redundancy in case of update failure, possibly the need for multi stage update, etc...)
- It often needs to be considered on a case by case basis and so it is not the purpose of this document to be a how to guide on how to achieve that.

Terminology

"Bootloader" and "Android" realms

There are two realms of the device layout that need to always be kept in sync to properly define a device.

- **Bootloader:** the bootloader realm contains the partition table required for the device. The partition table

is built into the bootloader scope (Android bsu) by the build system and is automatically applied when a change has been detected during bsu boot up.

- The exception is if the partition layout changes the position of the bsu, the bsu would refuse to update the layout because it can render the device unusable and unable to boot back into a known state (fastboot state, android boot.img).
- A typical usage example is to flash a new bootloader image with the new partition table, reboot into bootloader mode to force bsu loading which would automatically apply the new partition layout and then subsequently flash all the required android images for the device to boot up properly based on the new layout defined.
- Android: the android realm defines the size of the partitions required for android to work properly, this includes: system image, vendor image, etc...
 - When a device implements a a|b update mode, redundant partitions (eg. boot|system|<etc>) must have the same size.
 - For the partitions on which dm-verity is enabled, the android size of the partition must match exactly the partition table size of the partition as defined in the bootloader realm.

Legacy vs "ab"

Throughout the configuration setup of the device, a couple terms may be found that refer to the device type being considered.

- Legacy: legacy devices are devices which do not implement the a|b (redundant partitions) model. Those devices do not therefore require dual partition layout for what would be the redundant partitions; they do require some different partitions in order to work properly, as example the "recovery" partition is necessary in those cases.
 - Since there is no need for partition redundancy, large partitions (such as "system") can potentially be much larger in size than its equivalent partition in the a|b model realm because the large partition does not need to be duplicated.
 - Elements of the configuration which apply to legacy device model are typically named: XXX_LEGACY in the settings for the device.
- ab: ab devices are implementing the a|b (redundant partitions) model. Those device do require redundant partitions on some area and therefore need to be more conservative on the usage model entertained; they may not require some specific partition that legacy devices do require.
 - Elements of the configuration which apply to legacy device model are typically named: XXX_AB in the settings for the device.

Defining a Device Layout

Now that we have defined the terminology, we can look at the two aspects of the device configuration that need to be adjusted when wanting to do a device layout update.

Those aspects are similar whether you consider a "legacy" or a "ab" device, the difference between those would only be in the actual named settings that need to be changed to make the change applicable to the proper device target.

Android compliance: minimum userdata size

From an android CDD compliance point of view, the only aspect to be considered is the size of the "userdata" partition in a given layout.

For android O onward, the "userdata" must be at least 4GB in size. all Broadcom device integration are based on this requirement.

Step 1: updating the partition table (gpt)

This should always be the first step taken.

Understanding the gpt content

In order to update the partition table of a device, you first need to make sure which gpt layout is being used for a given device. This is typically linked into the device configuration itself:

```
./device/broadcom/elfin.mk:export LOCAL_DEVICE_GPT :=  
device/broadcom/common/gpts/ab-u.o.conf
```

Note: Beware that several device settings may share the same gpt definition in the Broadcom reference integration, so changing one may affect several devices; typically a specific device should associate its own gpt configuration module.

... looking at the linked in gpt, one sees (in this example):

```
$ cat gpts/ab-u.o.conf  
-s 7650410496 -b 0 -v 3 -c  
macadr,17K,512,0  
nvram,-,64K,0  
bsu,-,942K,0  
misc,1M,1M,0  
hwcfg,-,1M,0  
splash,-,12M,0  
eio,-,2M,0  
metadata,-,8M,0  
cache,-,10M,0x0001000000000000  
boot_i,-,64M,0  
boot_e,-,64M,0
```

```
system_i, -, 1418M, 0x0001000000000000  
system_e, -, 1418M, 0x0001000000000000  
vendor_i, -, 100M, 0x0001000000000000  
vendor_e, -, 100M, 0x0001000000000000  
userdata, -, -, 0x0001000000000000
```

... which can be interpreted as follows (only the relevant|important parts are described):

```
-s 7650410496
```

This is the total flash size available for the user data partition of the emmc flash on which this gpt layout will apply. This is generally the total size of the emmc flash minus the size of the reserved partitions (emmc boot partitions and emmc rpmb partition). If not known, this total size can be seen in bootloader at the BOLT command prompt using the "show devices" command.

```
-c
```

This is to ensure the gpt is generated in a format that can be integrated with the bootloader bsu for automatic support of the layout in the bootloader realm.

```
system_i, -, 1418M, 0x0001000000000000  
system_e, -, 1418M, 0x0001000000000000
```

This is a redundant pair of a|b mode partitions for the system image, named with the required partition suffix of "_i" and "_e" in this case. The "-" is simply to tell the gpt generation tool to calculate the start address of the partition based on the built layout thus far, this ensures the total emmc area is packed properly with all the defined partitions, leaving no holes or wasted space. The "1418M" is the total size of the partition in MB; the last argument is just a set of flags to define (in this case) that the partition is an ext4 formatted one, this is used by bootloader (fastboot) operations.

The "userdata" partition is always the last one in the layout; its size is basically calculated last to use all the available remaining space, which is required to be no smaller than the android CDD required size.

Changing the gpt content

For a given gpt setup (as seen in the example), all the partitions defined are needed for android proper integration.

When updating a layout, the simplest change required would be to massage the size of the overall flash area [-s <size>] parameter and|or the size(s) of the desired partitions as applicable.

When additional partitions are included in the layout which do not appear in the bcm reference layout, we recommend to place them before the userdata, therefore keeping the userdata still as the last partition always.

When a layout is being updated, for compliance with android requirements, the minimum size of a given existing partition present in the bcm layout offered cannot be smaller than the size defined in the bcm reference integration. Changing this may expose integration and/or update problems later on because bcm works with google to define this minimum layout required.

Once the gpt layout is updated, rebuilding and re-flashing the bootloader for the device would take care of making the change available.

Step 2: updating the android knowledge

This should always be the second step taken - since it is dependent on the sizes of the gpt layout defined prior.

The android configuration aspects are largely defined in the device/broadcom/common/settings.mk module; however some devices may have a device specific override for some of the settings represented, so it is a good idea to also check there.

The key here is to make sure that the size (expressed in full bytes) matches exactly the size defined in the gpt corresponding for the device. As example:

```
system_i, -, 1418M, 0x0001000000000000
```

... from the gpt tells us that this is a a|b mode (since _i slot is defined) and the size is 1418M; consequently the matching android configuration is:

```
export LOCAL_DEVICE_SYSTEM_AB           ?= 1486880768  # 1418M
```

Both values need to be in sync.

... and so on for all the applicable android partitions.

The only tricky situation is the "userdata" partition, since the userdata is defined in the gpt as "use all remaining space available", it's total size must be calculated accordingly and setup per device; it is also possible to "fix" the userdata size in the partition layout and setup the value accordingly in the android settings if that is preferred.

On dm-verity devices

For devices which enable dm-verity feature by default, it is extremely important to make sure the gpt size and the android size match exactly for those dm-verity aware partitions (which would typically include "system" and "vendor").

If this contract is not respected, the device will fail to boot; the error reported will be that there are dm-verity errors with the partition.