



Camera 模组自适应说明

V1.0

2013-05-29

Confidential

Revision History

Version	Date	Changes compared to previous issue
V1.0	2013-05-29	初始版本

Confidential

目录

1. Camera 模组自适应功能介绍.....	4
2. Camera 模组自适应功能使用说明.....	5
2.1. 功能开启.....	5
2.2. 功能关闭.....	6
3. Camera 模组自适应功能使用限制.....	7
3.1. 双模组支持.....	7
3.2. 模组替换.....	8
4. Declaration.....	9

1. Camera 模组自适应功能介绍

为了满足客户提出的相同固件能够适应 Camera 模组不同的多款机器的需求，A20 添加了 Camera 模组自适应功能。开启该功能即可实现 Camera 模组的自动检测，确认该款机器使用的模组型号并保存相关参数供后面使用。关闭该功能即可恢复 Camera 模组固定配置的模式。

在 `sys_config.fex` 配置中，可配 Camera 模组自适应功能的相关参数，包括开启标志位，以及需要检测的 Camera 模组列表。将这些设置项配置好，在 android 层的 `init.rc` 中配置加载 Camera 模组自适应驱动以及需要检测的各种 Camera 模组驱动。这样，在系统启动时，Camera 模组自适应驱动会自动检测当前机器的 Camera 模组是否在模组支持列表中，如确定了模组型号参数，会将该模组的相关参数保存用于开启 Camera 时调用。

下面对 Camera 模组自适应功能的配置说明等进行阐述。

2. Camera 模组自适应功能使用说明

2.1. 功能开启

首先, 对 Android 层进行配置, 在 `android4.2.1\device\softwinner\wing-xxx\init.sun7i.rc` 中配置启动自动加载自动检测模块和一系列需要自适应的 Camera 模组。如下所示:

```
#csi module
insmod /system/vendor/modules/camera.ko // 加载自动检测模块
.....
insmod /system/vendor/modules/gc0308.ko
insmod /system/vendor/modules/gc2035.ko
insmod ..... //加载一系列需要支持的 sensor 列表
```

然后, 对驱动层进行配置, 在 `lichee\tools\pack\chips\sun7i\configs\android\wing-xxx` 的 `sys_config.fex` 中添加 camera 模组的自动检测参数列表。如下所示:

```
[camera_list_para]
camera_list_para_used = 1
ov7670 = 0
gc0308 = 1
gc2035 = 1
gt2005 = 0
hi704 = 0
sp0838 = 0
mt9m112 = 0
mt9m113 = 0
ov2655 = 0
hi253 = 0
gc0307 = 0
mt9d112 = 0
ov5640 = 0
gc2015 = 0
ov2643 = 0
gc0329 = 0
gc0309 = 0
tvp5150 = 0
s5k4ec = 0
```

ov5650_mv9335 = 0
siv121d = 0

其中，`camera_list_para_used` 负责 Camera 模组自适应功能的开启和关闭，1 为开启，0 为关闭，其余参数负责各模组自动检测的开启和关闭，1 表示对该模组型号进行检测，0 表示不检测。

2.2. 功能关闭

关闭该功能只需将 `lichee/tools/pack/chips/sun7i/configs/android/wing-xxx` 的 `sys_config.fex` 中的 `camera_list_para_used` 置为 0 即可。

需要注意的是：即使关闭了 Camera 模组自适应功能，也要加载 `camera.ko`，否则 `camera` 模块将无法正常工作。

3. Camera 模组自适应功能使用限制

3.1. 双模组支持

目前自 Camera 模组自适应功能只支持两个模组挂在同一个 CSI 上的情况，对于两个 sensor 分别挂在 CSI0, CSI1 上的情况，请关闭自动检测功能。

对应的 csi0_para 配置如下：

```
[csi0_para]
csi_used                = 1

csi_dev_qty             = 2
csi_stby_mode           = 0
csi_mname               = "gc0308"
csi_twi_id              = 1
csi_twi_addr            = 0x42
csi_if                  = 0
csi_vflip               = 0
csi_hflip               = 0
csi_iovdd               = "axp20_p11"
csi_avdd                = ""
csi_dvdd                = ""
csi_vol_iovdd           = 2800
csi_vol_dvdd            =
csi_vol_avdd            =
csi_flash_pol           = 0

csi_mname_b             = "gc2035"
csi_twi_id_b            = 1
csi_twi_addr_b          = 0x78
csi_if_b                 = 0
csi_vflip_b             = 0
csi_hflip_b             = 0
csi_iovdd_b             = "axp20_p11"
csi_avdd_b              = ""
csi_dvdd_b              = ""
csi_vol_iovdd_b         = 2800
csi_vol_avdd_b          =
csi_vol_dvdd_b          =
```

```
csi_flash_pol_b      = 0

csi_pck              = port:PE00<3><default><default><default>
csi_ck               = port:PE01<3><default><default><default>
csi_hsync            = port:PE02<3><default><default><default>
csi_vsync            = port:PE03<3><default><default><default>
csi_d0               = port:PE04<3><default><default><default>
csi_d1               = port:PE05<3><default><default><default>
csi_d2               = port:PE06<3><default><default><default>
csi_d3               = port:PE07<3><default><default><default>
csi_d4               = port:PE08<3><default><default><default>
csi_d5               = port:PE09<3><default><default><default>
csi_d6               = port:PE10<3><default><default><default>
csi_d7               = port:PE11<3><default><default><default>
csi_reset            = port:PH13<1><default><default><0>
csi_power_en         = port:PH16<1><default><default><0>
csi_stby             = port:PH18<1><default><default><0>
csi_flash            =
csi_af_en            =
csi_reset_b          = port:PH13<1><default><default><0>
csi_power_en_b       =
csi_reset_b          = port:PH19<1><default><default><0>
csi_flash_b          =
csi_af_en_b          =
```

这里需要注意的是，因为是自适应，对应 csi0_para 中的“csi_mname”以及“csi_twi_addr”可以不填，我们可以自动检测出来，但是前面填写的那一组 gpio 上对应的摄像头将会先检测出来，我们置为 camera 0；而后面那一组 gpio（即：csi_reset_b, csi_reset_b）上对应的摄像头后面才会被检测出来，我们置为 camera 1。注意 camera 0 要对应 camera.cfg 中的参数 device_id =0 的摄像头。而 camera 1 要对应 camera.cfg 中的参数 device_id =1 的摄像头。s

3.2. 模组替换

模组替换时要保证替换模组的最大分辨率与原模组相同，即 30W 的模组只能用 30W 的模组替换，而不能用 200W 的模组替换。同时，要尽量保证模组的相似性，即白平衡，曝光等一些其他参数的一致性。

4. Declaration

This document is the original work and copyrighted property of Allwinner Technology (“Allwinner”). Reproduction in whole or in part must obtain the written approval of Allwinner and give clear acknowledgement to the copyright owner.

The information furnished by Allwinner is believed to be accurate and reliable. Allwinner reserves the right to make changes in circuit design and/or specifications at any time without notice. Allwinner does not assume any responsibility and liability for its use. 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 Allwinner. This datasheet neither states nor implies warranty of any kind, including fitness for any particular application.