

N1 Smart Module Adding Camera Driver

Version 1.0



Copyright © Neoway Technology Co., Ltd 2016. All rights reserved.

No part of this document may be reproduced or transmitted in any form or by any means without prior written consent of Shenzhen Neoway Technology Co., Ltd.

Neoway 有方 is the trademark of Neoway Technology Co., Ltd.

All other trademarks and trade names mentioned in this document are the property of their respective holders.

Notice

This document provides guide for users to use the N1.

This document is intended for system engineers (SEs), development engineers, and test engineers.

The information in this document is subject to change without notice due to product version update or other reasons.

Every effort has been made in preparation of this document to ensure accuracy of the contents, but all statements, information, and recommendations in this document do not constitute a warranty of any kind, express or implied.

Neoway provides customers complete technical support. If you have any question, please contact your account manager or email to the following email addresses:

Sales@neoway.com

Support@neoway.com

Website: <http://www.neoway.com>

Revision Record

Issue	Changes	Date
V1.0	Initial draft	2016-06

Neoway

Contents

1 Overview.....	1
2 Adding to Kernel.....	1
3 Adding to Outside of Kernel.....	2



1 Overview

Adding camera driver includes two parts: one in the kernel and one outside of the kernel. You can register the device in the way that you add Linux drivers. The rest part of the driver should be added to the **neoway/custdriver** directory. The driver registration is completed after sensor parameters of N1_camera are uploaded to kernel from user's space.

2 Adding to Kernel

Navigate to **kernel/arch/arm/boot/dts/jsr/N1_msm8909-camera-sensor-skue.dtsi** and modify the **.dtsi** file.

Add the camera device node to corresponding I2C bus in the **.dtsi** file.



Index0 indicates rear camera; index1 indicates front camera. For N1, driver code is same.

```
qcom,camera@0 {
    cell-index = <0>;
    compatible = "qcom,camera";
    reg = <0x2>;
    qcom,special-support-sensors="ov5670_wdsen";
    qcom,eeeprom-src = <&eeeprom0>;
    qcom,csiphy-sd-index = <0>;
    qcom,csid-sd-index = <0>;
    qcom,mount-angle = <90>;
    qcom,actuator-src = <&actuator0>;
    qcom,led-flash-src = <&led_flash0>;
    cam_vana-supply = <&pm8909_l17>;
    cam_vio-supply = <&pm8909_l6>;
    cam_vdig-supply = <&pm8909_l2>;
    cam_vaf-supply = <&pm8909_l8>;
    qcom,cam-vreg-name = "cam_vio", "cam_vana", "cam_vdig",
        "cam_vaf";
    qcom,cam-vreg-type = <1 0 0 0>;
    qcom,cam-vreg-min-voltage = <0 2800000 1200000 2850000>;
    qcom,cam-vreg-max-voltage = <0 2850000 1200000 2900000>;
    qcom,cam-vreg-op-mode = <0 80000 80000 100000>;
    pinctrl-names = "cam_default", "cam_suspend";
    pinctrl-0 = <&cam_sensor_mclk0_default
        &cam_sensor_rear_default>;
    pinctrl-1 = <&cam_sensor_mclk0_sleep &cam_sensor_rear_sleep>;
    gpios = <&msm_gpio 26 0>,
        <&msm_gpio 35 0>,
        <&msm_gpio 34 0>;
    qcom,gpio-reset = <1>;
    qcom,gpio-standby = <2>;
    qcom,gpio-req-tbl-num = <0 1 2>;
    qcom,gpio-req-tbl-flags = <1 0 0>;
    qcom,gpio-req-tbl-label = "CAMIF_MCLK",
        "CAM_RESET1",
        "CAM_STANDBY";
    qcom,sensor-position = <0>;
    qcom,sensor-mode = <0>;
    qcom,cci-master = <0>;
    status = "ok";
}
```

3 Adding to Outside of Kernel

1. Add the driver file.

Add a folder that includes the driver file and **Android.mk** to the **neoway/custdriver/camera/sensor/sensor_libs** directory.

 Android.mk	2015/9/7 9:03	MK 文件	2 KB
 ov5670_wdsen_lib.c	2015/9/7 9:03	UltraEdit Docum...	35 KB

In the **ov5670_wdsen_lib.c** file, modify the on/off sequence of the sensor:

```
static struct msm_sensor_power_setting ov5670_wdsen_power_setting[] = {
    {
        .seq_type = SENSOR_VREG,
        .seq_val = CAM_VIO,
        .config_val = 0,
        .delay = 0,
    },
    {
        .seq_type = SENSOR_VREG,
        .seq_val = CAM_VANA,
        .config_val = 0,
        .delay = 0,
    },
    {
        .seq_type = SENSOR_VREG,
        .seq_val = CAM_VDIG,
        .config_val = 0,
        .delay = 0,
    },
    {
        .seq_type = SENSOR_VREG,
        .seq_val = CAM_VAF,
        .config_val = 0,
        .delay = 5,
    },
    {
        .seq_type = SENSOR_GPIO,
        .seq_val = SENSOR_GPIO_STANDBY,
        .config_val = GPIO_OUT_LOW,
    }
};

static struct msm_camera_sensor_slave_info sensor_slave_info = {
    /* Camera slot where this camera is mounted */
    .camera_id = CAMERA_0,
    /* sensor slave address */
    .slave_addr = 0x6D,
    /* sensor i2c frequency*/
    .i2c_freq_mode = I2C_FAST_MODE,
    /* sensor address type */
    .addr_type = MSM_CAMERA_I2C_WORD_ADDR,
    /* sensor id info*/
    .sensor_id_info = {
        /* sensor id register address */
        .sensor_id_reg_addr = 0x300b,
        /* sensor id */
        .sensor_id = 0x5670,
    },
    /* power up / down setting */
    .power_setting_array = {
        .power_setting = ov5670_wdsen_power_setting,
        .size = ARRAY_SIZE(ov5670_wdsen_power_setting),
    },
    .is_flash_supported = SENSOR_FLASH_SUPPORTED,
};
```


The data set `ov5670_wdsen_power_setting` is included in the `sensor_lib_ptr` global variable, which is transited to the following function:

```
int32_t msm_sensor_driver_probe(void *setting,
    struct msm_sensor_info_t *probed_info, char *entity_name)
{
    int32_t rc = 0;
    struct msm_sensor_ctrl_t *s_ctrl = NULL;
    struct msm_camera_cci_client *cci_client = NULL;
    struct msm_camera_sensor_slave_info *slave_info = NULL;
    struct msm_camera_slave_info *camera_info = NULL;

```

Obtain power information from the probe function.

```
    goto ↓free_slave_info;
}

rc = msm_sensor_get_power_settings(setting, slave_info,
    &s_ctrl->sensordata->power_info);
if (rc < 0) {
    pr_err("failed");
    goto ↓free_slave_info;
}

```

Power up in the probe function after obtaining power information.

```
    pr_err("%s failed %d\n", __func__, __LINE__);
    goto ↓free_camera_info;
}

/* Power up and probe sensor */
rc = s_ctrl->func_tbl->sensor_power_up(s_ctrl);
if (rc < 0) {
    pr_err("%s power up failed", slave_info->sensor_name);
    goto ↓free_camera_info;
}

```

Among `ov5670_wdsen_power_setting`:

- `.seq_type`: type of the pin selected
- `SENSOR_GPIO`: Use GPIO to supply power for sensor
- `SENSOR_VREG`: Use the LDO inside 8909 to supply power
- `SENSOR_CLK`: Clock
- `.seq_val`: Function pins of the sensor
- `SENSOR_GPIO_VANA`: Analog voltage pin
- `SENSOR_GPIO_VDIG`: Digital voltage pin
- `SENSOR_GPIO_RESET`: Reset pin

- **SENSOR_CAM_MCLK**: I2C clock pin
- **.delay**: level delay
- **.config_val**: level of pins

2. Add effect files.

Add a folder that contains sub-folders **common**, **preview**, **snapshot**, **video**, **zsl** to the **neoway/cusdriver/camera/chromatix/0301/libchromatix** directory. Each of the sub-folder contains code files and **Android.mk**.

common	2015/9/7 9:03	文件夹	
hfr_60fps	2015/9/7 9:03	文件夹	
hfr_90fps	2015/9/7 9:03	文件夹	
hfr_120fps	2015/9/7 9:03	文件夹	
liveshot	2015/9/7 9:03	文件夹	
preview	2015/9/7 9:03	文件夹	
snapshot	2015/9/7 9:03	文件夹	
video	2015/9/7 9:03	文件夹	
video_hd	2015/9/7 9:03	文件夹	
zsl	2015/9/7 9:03	文件夹	
Android.mk	2015/9/7 9:03	MK 文件	1 KB

3. Modify the **neo-device.mk** file.

neo-device.mk is under the **neoway/cusdriver** directory. Check all **Android.mk** files in previous steps (driver file and effect file) to get the value of **LOCAL_MODULE**.

```
LOCAL_C_INCLUDES += chromatix_ov5670_wdsen_video.h
LOCAL_SRC_FILES := chromatix_ov5670_wdsen_video.c
LOCAL_MODULE := libchromatix_ov5670_wdsen_default_video
```

Add the found values to **MM_CAMERA** in **neo-device.mk** in the **neoway/cusdriver** directory.

```
# camera module
MM_CAMERA := #
MM_CAMERA += cpp_firmware_v1_1_6.fw
MM_CAMERA += libmmcamera_skuf_ov5648_p5v23c
MM_CAMERA += libchromatix_skuf_ov5648_p5v23c_common
MM_CAMERA += libchromatix_skuf_ov5648_p5v23c_preview
MM_CAMERA += libchromatix_skuf_ov5648_p5v23c_snapshot
MM_CAMERA += libchromatix_skuf_ov5648_p5v23c_default_video
MM_CAMERA += libchromatix_skuf_ov5648_p5v23c_zsl
MM_CAMERA += libchromatix_ov5670_wdsen_default_video
```

Note that you need to modify the **Android.mk** file correctly.

 NOTE

For N1, the userspace parameters need to be uploaded to kernelspace via neoway interface. Check if your sensor is supported by our interface before you add the sensor driver. If not, please contact us to update **libmmcamera2_sensor_modules.so** !.

Neoway