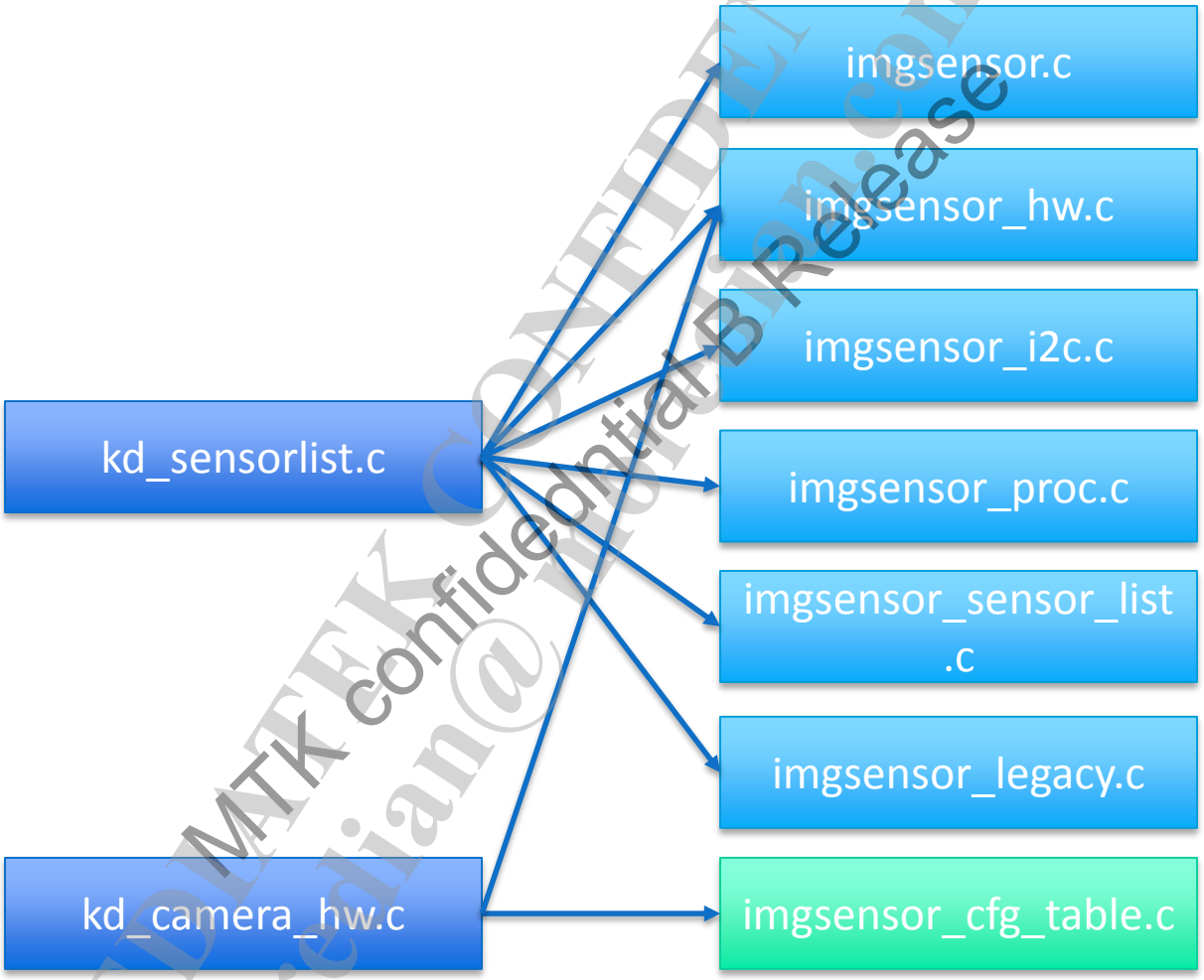
Sensor Porting Guide --- MT6763

Outline

- What's Changed
- Sensor Driver Architecture
- Sensor Driver Porting
  - How To Add A New Sensor

WHAT'S CHANGED

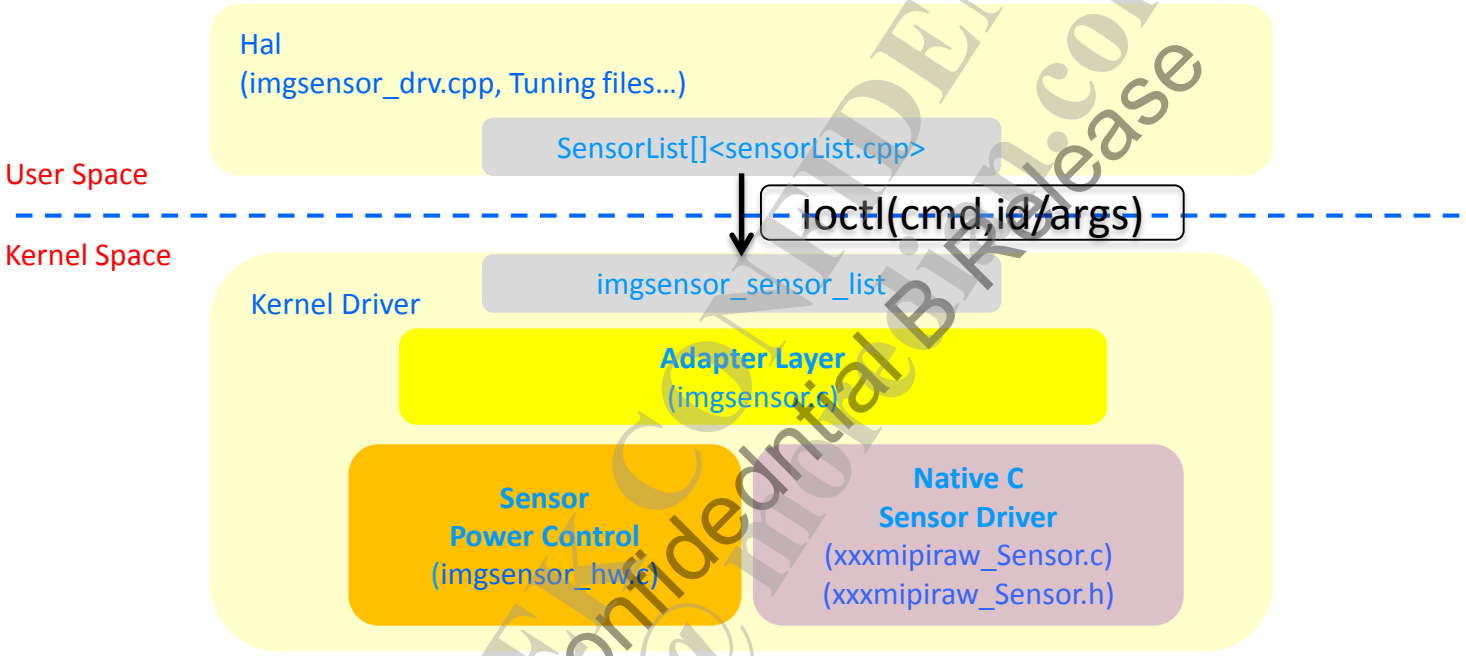
Split large file into multiple files.



File Description

File	Decription
imgsensor.c	Sensor driver adapter and driver entry point
imgsensor_hw.c	Sensor power control.
imgsensor_i2c.c	I2C read/write.
imgsensor_proc.c	PROC related part.
imgsensor_sensor_list.c	List of all sensors init function
imgsensor_legacy.c	Legacy part of sensor. Mainly I2C related API.
imgsensor_cfg_table.c	Sensor power and I2C configuration table

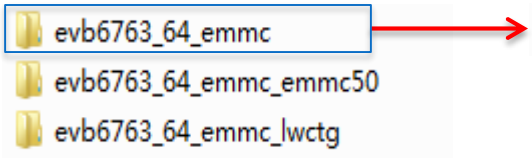
Driver Architecture



- Imgsensor Drv
  - User space driver
- Adapter layer
  - A adapter layer for Linux character device driver and native sensor driver
- Sensor power control
  - Control the sensor power on/off

Sensor Driver Porting (N0)

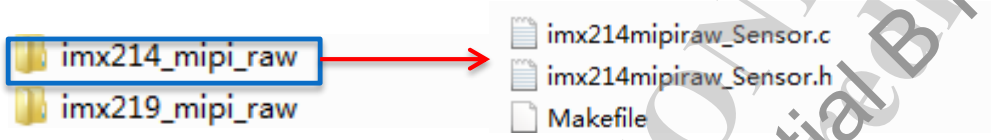
- Config Files
  - /device/mediatek /${project}/



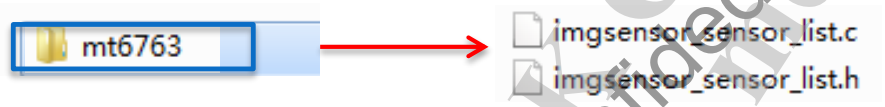
  - /kernel-4.4/arch/arm64/configs/

Sensor Driver Porting (N0)

- Kernel Driver
  - /kernel-4.4/drivers/misc/mediatek/imgsensor/src/ /${platform}/



  - /kernel-4.4/drivers/misc/mediatek/imgsensor/src/

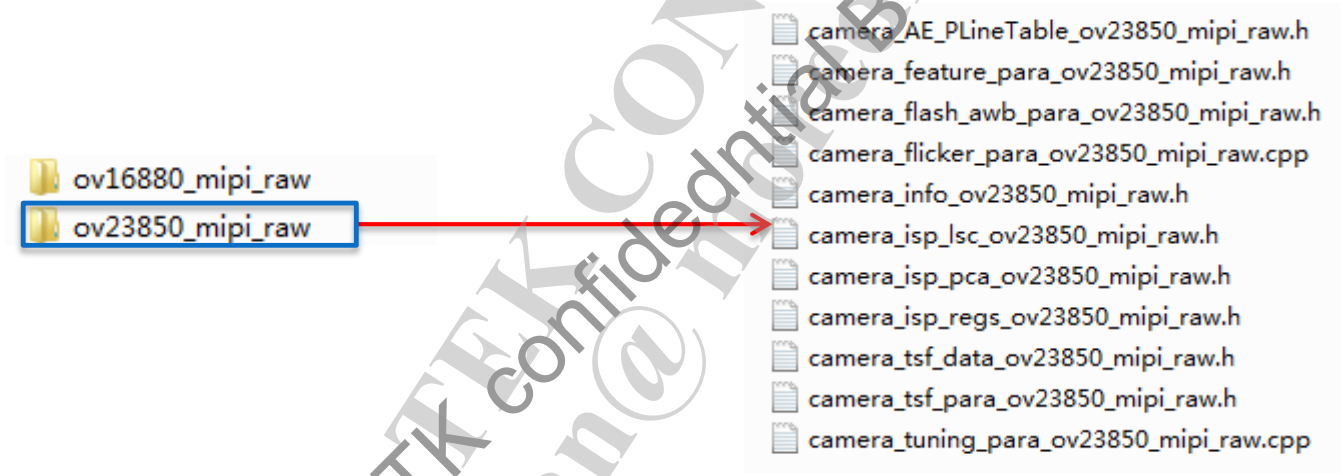


  - /kernel-4.4/drivers/misc/mediatek/imgsensor/

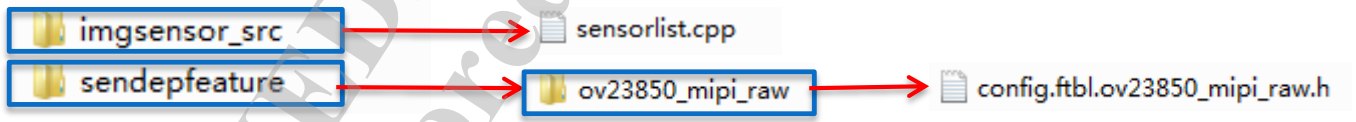
  - kernel-4.4/drivers/misc/mediatek/imgsensor/src/mt6763/

Sensor Driver Porting (N0)

- Hal Driver
  - /vendor/mediatek/proprietary/custom/ ${platform}/hal/imgsensor/



  - /vendor/mediatek/proprietary/custom/${platform}/hal/



How To Add A New Sensor

- Step1

/device/mediatek/ ${project} /**ProjectConfig.mk**

/kernel-4.4/arch/arm64/configs/ **${project}_debug_defconfig**

/kernel-4.4/arch/arm64/configs**/${project}_defconfig**

1.1ProjectConfig.mk 如下修改

(a) 修改imgsensor相关

eg:main(后摄) xxx_mipi_raw, sub (前摄)xxxx_mipi_raw, main2(stereo)xxxx_mipi_raw)

- CUSTOM_HAL_IMGSENSOR = **xxxx_mipi_raw**
- CUSTOM_KERNEL_IMGSENSOR = **xxxx_mipi_raw**
- CUSTOM_HAL_MAIN_IMGSENSOR = **xxxx_mipi_raw**
- CUSTOM_HAL_SUB_IMGSENSOR = **xxxx_mipi_raw**
- CUSTOM_KERNEL_MAIN_IMGSENSOR = **xxxx_mipi_raw**
- CUSTOM_KERNEL_SUB_IMGSENSOR = **xxxx_mipi_raw**
- CUSTOM_HAL_MAIN2_IMGSENSOR = **xxxx_mipi_raw**
- CUSTOM_KERNEL_MAIN2_IMGSENSOR = **xxxx_mipi_raw**

How To Add A New Sensor

(b) 修改lens相关

sensor porting时，先将lens配置为dummy。

- CUSTOM_HAL_LENS = **dummy_lens**
- CUSTOM_KERNEL_LENS = **dummy_lens**
- CUSTOM_HAL_MAIN_LENS = **dummy_lens**
- CUSTOM_HAL_SUB_LENS = **dummy_lens**
- CUSTOM_KERNEL_MAIN_LENS = **dummy_lens**
- CUSTOM_KERNEL_SUB_LENS = **dummy_lens**

(c) 修改flashlight相关

支持Flashlight设置为constant_flashlight, 不支持设置为dummy_flashlight

- CUSTOM_HAL_FLASHLIGHT = dummy_flashlight
- CUSTOM_KERNEL_FLASHLIGHT = dummy_flashlight

1.2 ${project}_debug_defconfig 或者/${project}_defconfig

CONFIG_CUSTOM_KERNEL_IMGSENSOR="**xxxx**_mipi_raw **xxxx**_mipi_raw "

How To Add A New Sensor

Step2

1)/kernel-4.4/drivers/misc/mediatek/imgsensor/inc/kd_imgsensor.h，

2)device/mediatek/common/kernel-headers/kd_imgsensor.h，

- Add new sensor ID define ,eg:

```
/* Add new sensor ID define */
#define OV5648MIPI_SENSOR_ID
```

0x5648

The value of Sensor ID learn from specifc sensor datasheet

- Add a new sensor name define

```
/* Add a new sensor name define */
#define SENSOR_DRVNAME_OV5648_MIPI_RAW          "ov5648mipiraw"
```

Step3

How To Add A New Sensor

/kernel-4.4/drivers/misc/mediatek/imgsensor/src/mt6763/imgsensor_sensor_list.h

- Add new sensor init function declaration, eg:

```
/* Add new sensor init function declaration */
UINT32 OV5648MIPISensorInit(PSENSOR_FUNCTION_STRUCT *pfFunc);
```

/kernel-4.4/drivers/misc/mediatek/imgsensor/src/mt6763/imgsensor_sensor_list.c

- Add new sensor in kernel kdSensorList[],eg:

```
#if defined(OV5648_MIPI_RAW)
    {OV5648MIPI_SENSOR_ID, SENSOR_DRVNAME_OV5648_MIPI_RAW, OV5648MIPISensorInit},
#endif
```

Sensor ID

Sensor name

Sensor init Funciton

How To Add A New Sensor

Step4

/vendor/mediatek/proprietary/custom/mt6763/hal/imgsensor_src/**sensorlist.cpp**

- Add new sensor in hal SensorList[]

```
#if defined(OV5648_MIPI_RAW)
    RAW_INFO(OV5648MIPI_SENSOR_ID, SENSOR_DRVNAME_OV5648_MIPI_RAW, NULL),
#endif
```

Note:sensorlist.cpp中的SensorList[]与imgsensor_sensor_list.c中的kdSensorList[]sensor的顺序**必须一致**，否则user space 和kernel space在通过ioctl传递命令id时会对应错误。

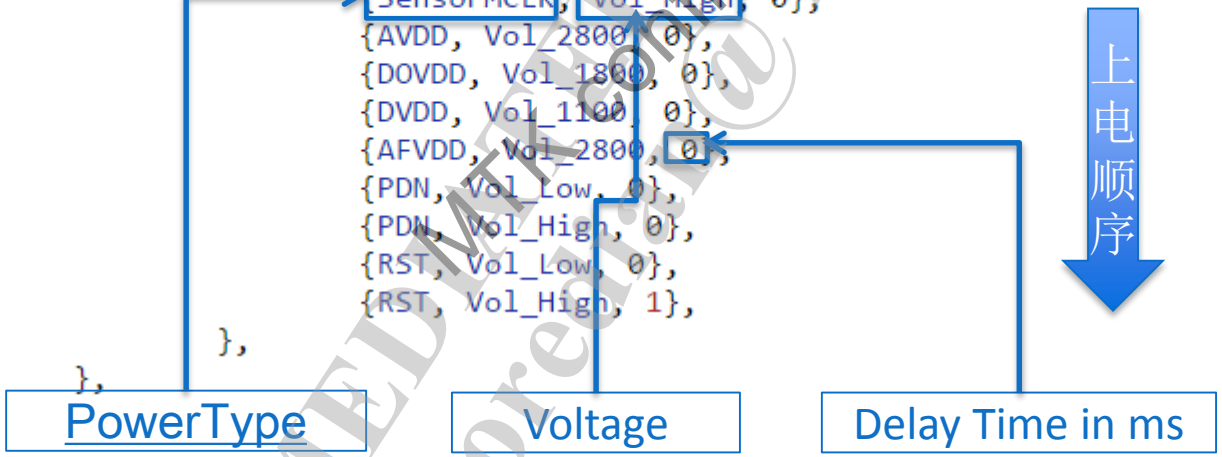
How To Add A New Sensor
Kernel 4.4

Step5 Power On/Off

/drivers/misc/mediatek/imgsensor/src/${platform}/camera_hw/imgsensor_cfg_table.c,

as for Power On/Off Sequence ,please refer to specific sensor datasheet;

```
struct IMGSENSOR_HW_POWER_SEQ sensor_power_sequence[] = {
#if defined(IMX398_MIPI_RAW)
  {
          SENSOR_DRVNAME_IMX398_MIPI_RAW,
          {
                  {SensorMCLK, Vol_High, 0},
                  {AVDD, Vol_2800, 0},
                  {DOVDD, Vol_1800, 0},
                  {DVDD, Vol_1100, 0},
                  {AFVDD, Vol_2800, 0},
                  {PDN, Vol_Low, 0},
                  {PDN, Vol_High, 0},
                  {RST, Vol_Low, 0},
                  {RST, Vol_High, 1},
          },
  },
```



How To Add A New Sensor

Step5 Power On/Off<Using PMIC>

- DVDD/DOVDD/AVDD/AFVDD的电压支持情况可以参考specific Pmic datasheet,eg mt6356:

Power Type	voltage (v)
DVDD	1.0/1.1/1.2/1.3/1.5/1.8
DOVDD	0.9/0.95/1.0/1.05/1.2/1.5/1.8
AVDD	1.8/2.2/2.375/2.8
AFVDD	1.2/1.3/1.5/1.8/2.0/2.8/3.0/3.3

- 或者查询/kernel-4.4/drivers/misc/mediatek/pmic/mt6356/v1/regulator_codegen.c这支文件中的定义,eg:

mt6356.dtsi

```
static const int vcamd_voltages[] = {
    1000000,
    1100000,
    1200000,
    1300000,
    1500000,
    1800000,
};
```

How To Add A New Sensor

Step5 Power On/Off<Using PMIC>

<c>在regulator.c文件中添加如下定义：

imgsensor/src/mt6763/camera_hw/regulator/regulator.c

```
struct REGULATOR_CTRL regulator_control[REGULATOR_TYPE_MAX_NUM] = {
    {"vcama"},
    {"vcamd"},
    {"vcamio"},
    {"vcamaf"},
    {"vcama_sub"},
    {"vcamd_sub"},
    {"vcamio_sub"},
    {"vcama_main2"},
    {"vcamd_main2"},
    {"vcamio_main2"},
    {"vcama_sub2"},
    {"vcamd_sub2"},
    {"vcamio_sub2"}
};

    ......
```

How To Add A New Sensor

Step5 Power On/Off<Using PMIC>

获取特定的PMIC接口：

1. 获取regulator

```
static enum IMGSENSOR_RETURN regulator_init(void *pinstance)
{
        struct REGULATOR *preg = (struct REGULATOR *)pinstance;
        struct REGULATOR_CTRL    *pregulator_ctrl = regulator_control;
        struct device            *pdevice;
        struct device_node       *pof_node;
        struct regulator *IOVDD;
        ......
        IOVDD = regulator_get(pdevice, "vcamio");
        ......
}
```

2. regulator_set_voltage设置电压

How To Add A New Sensor

Step5 Power On/Off<LDO using GPIO Enable>

<a>在$(project).dts文件中增加如下子节点，如下图：

```
&kd_camera_hw1 {
    pinctrl-names = "default",
            "cam0_rst0", "cam0_rst1", "cam0_pnd0", "cam0_pnd1",
            "cam1_rst0", "cam1_rst1", "cam1_pnd0", "cam1_pnd1",
            "cam2_rst0", "cam2_rst1", "cam2_pnd0", "cam2_pnd1",
            "cam_ldo_vcama_0", "cam_ldo_vcama_1",
            "cam_ldo_vcamd_0", "cam_ldo_vcamd_1",
    /* ..... */
    /* for ldo control by gpio */
    pinctrl-13 = <&camera_pins_cam_ldo_vcama_0>;
    pinctrl-14 = <&camera_pins_cam_ldo_vcama_1>;
    pinctrl-15 = <&camera_pins_cam_ldo_vcamd_0>;
    pinctrl-16 = <&camera_pins_cam_ldo_vcamd_1>;
    /*......*/
    status = "okay";
};
```

Name for pinctrl_lookup_state()

How To Add A New Sensor

Step5 Power On/Off<LDO Using GPIO Enable>

<b>在dts文件中pio节点中增加如下子节点的定义，如下图

```
camera_pins_cam_ldo_vcama_0: cam0@vcama0 {
    pins_cmd_dat {
        pins = <PINMUX_GPIO253__FUNC_GPIO253>;
        slew-rate = <1>;
        output-low;
    };
};
camera_pins_cam_ldo_vcama_1: cam1@vcama1 {
    pins_cmd_dat {
        pins = <PINMUX_GPIO253__FUNC_GPIO253>;
        slew-rate = <1>;
        output-high;
    };
};
```

```
camera_pins_cam_ldo_vcamd_0: cam0@vcamd0 {
    pins_cmd_dat {
        pins = <PINMUX_GPIO110__FUNC_GPIO110>;
        slew-rate = <1>;
        output-low;
    };
};
camera_pins_cam_ldo_vcamd_1: cam1@vcamd1 {
    pins_cmd_dat {
        pins = <PINMUX_GPIO110__FUNC_GPIO110>;
        slew-rate = <1>;
        output-high;
    };
};
```

0: input; 1: output

GPIO output low

GPIO output high

Set GPIO110to GPIO mode defined at /kernel-4.4/arch/arm64/boot/dts/include/dt-bindings/pinctrl/mt6763-pinfunc.h

How To Add A New Sensor

Step5 Power On/Off<LDO Using GPIO Enable>

<c>在gpio.c文件中添加如下定义：

/drivers/misc/mediatek/imgsensor/src/mt6763/camera_hw/gpio/gpio.c

```
struct GPIO_PINCTRL gpio_pinctrl_list[GPIO_CTRL_STATE_MAX_NUM] = {
  /* Main */
  {"cam0_pnd1"},
  {"cam0_pnd0"},
  {"cam0_rst1"},
  {"cam0_rst0"},
  {"cam_ldo_vcama_1"},
  {"cam_ldo_vcama_0"},
  {"cam_ldo_vcamd_1"},
  {"cam_ldo_vcamd_0"},
  {"cam_ldo_vcamio_1"},
  {"cam_ldo_vcamio_0"},
  {"cam_ldo_vcamaf_1"},
  {"cam_ldo_vcamaf_0"},
  /* Sub */
  {"cam1_pnd1"},
  {"cam1_pnd0"},
    ......
```

若使用mipi switch，把MIPI_SWITCH宏打开。

需要添加自定义GPIO时，可以仿照mipi switch gpio的写法。

```
#ifdef MIPI_SWITCH
  {"cam_mipi_switch_en_1"},
  {"cam_mipi_switch_en_0"},
  {"cam_mipi_switch_sel_1"},
  {"cam_mipi_switch_sel_0"}
#endif
```

How To Add A New Sensor

Step5 Power On/Off<LDO Using GPIO Enable>

<d>在gpio.c文件中gpio_init函数中会获取dts定义的GPIO信息：

```
for(i=0; i<GPIO_CTRL_STATE_MAX_NUM; i++, pgpio_pinctrl++) {
        if (pgpio_pinctrl->ppinctrl_lookup_state &&
                IS_ERR(pgpio->ppinctrl_state[i] =
                pinctrl_lookup_state(pgpio->ppinctrl, pgpio_pinctrl->ppinctrl_lookup_state))) {
                PK_ERR("%s : pinctrl err, %s\n", __func__, pgpio_pinctrl->ppinctrl_lookup_state);
                ret = IMGSENSOR_RETURN_ERROR;
        }
}
```

gpio_set() 内实现GPIO电平控制

```
static enum IMGSENSOR_RETURN gpio_set

  {       pinctrl_select_state(pgpio->ppinctrl, ppinctrl_state);
```

Init操作完成以后，就可以通过如下方法设置GPIO 的电平了

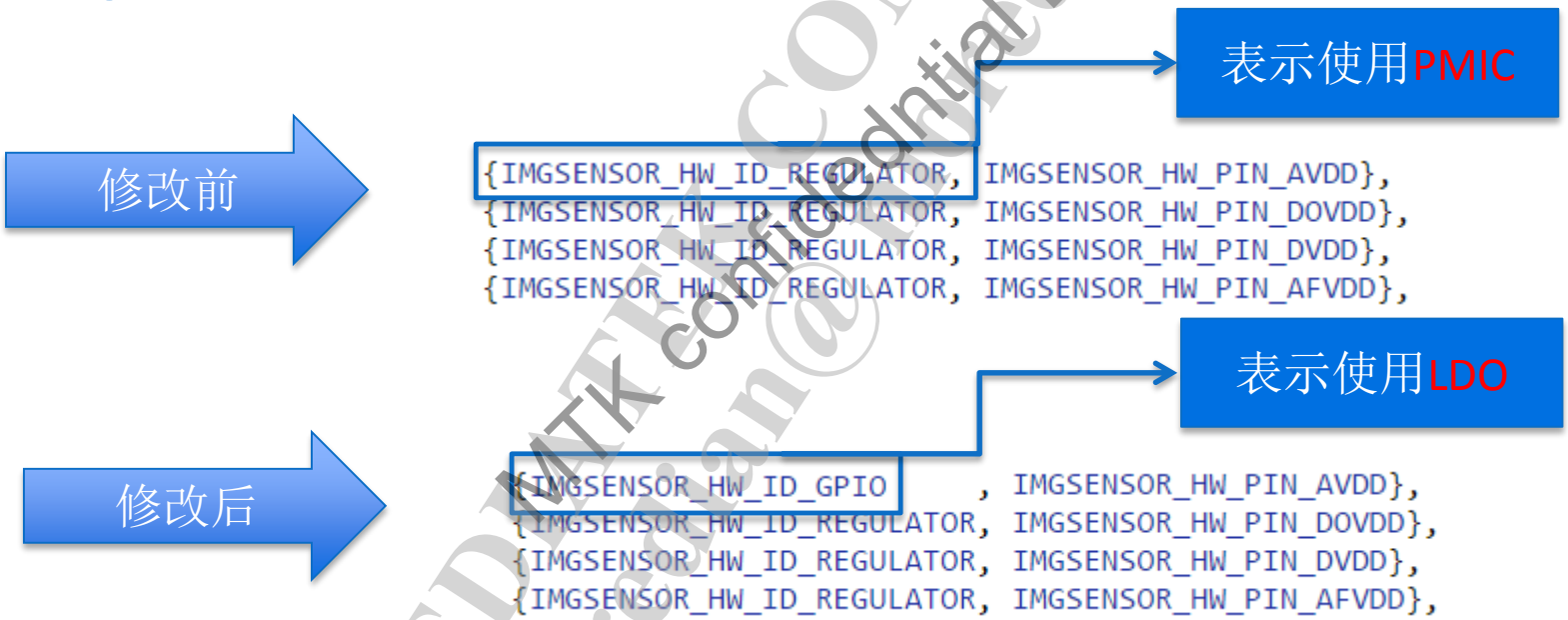
Eg:pinctrl_select_state(camctrl, cam_ldo_vcamd_l);

How To Add A New Sensor

Step5 Power On/Off<LDO Using GPIO Enable>

<e>修改在imgsensor_cfg_table.c文件中imgsensor_custom_config中成员的值，

eg:如果main sensor 使用的是外部LDO，请做如下修改：



MIPI SWITCH

/kernel-4.4/drivers/misc/mediatek/imgsensor/src/mt6763/imgsensor_cfg_table.c

```
struct IMGSENSOR_HW_POWER_SEQ platform_power_sequence[] = {
#ifdef MIPI_SWITCH
  {
        IMGSENSOR_SENSOR_IDX_NAME_SUB,
        {
              {
                    IMGSENSOR_HW_PIN_MIPI_SWITCH_EN,
                    IMGSENSOR_HW_PIN_STATE_LEVEL_0,
                    0,
                    IMGSENSOR_HW_PIN_STATE_LEVEL_HIGH,
                    0
              },
              {
                    IMGSENSOR_HW_PIN_MIPI_SWITCH_SEL,
                    IMGSENSOR_HW_PIN_STATE_LEVEL_HIGH,
                    0,
                    IMGSENSOR_HW_PIN_STATE_LEVEL_0,
                    0
              },
        }
  },
```

操作sub sensor

EN管脚

上电拉低

上电延时

下电拉高

下电延时

SEL管脚

上电拉高

上电延时

下电　拉低

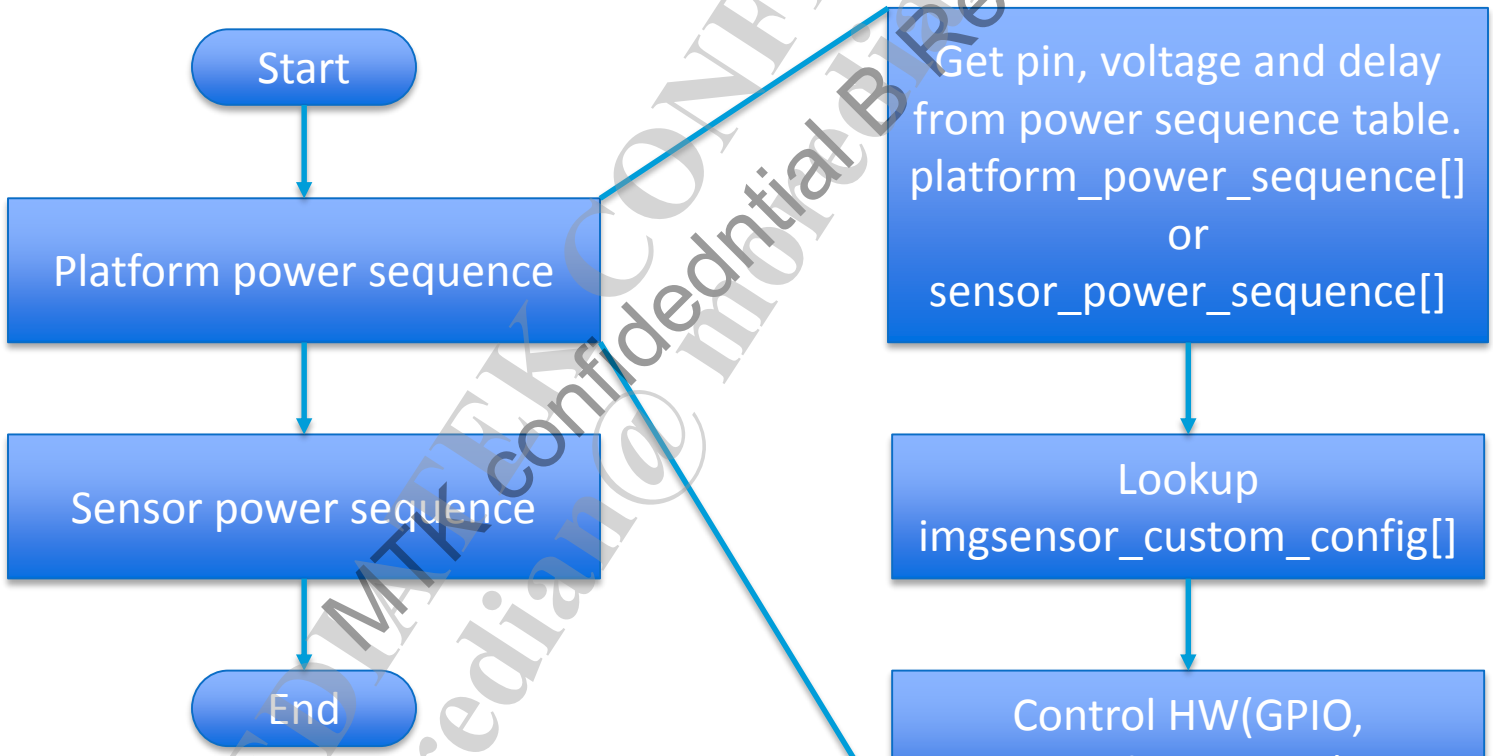
下电延时

```
struct IMGSENSOR_HW_POWER_INFO {
    enum IMGSENSOR_HW_PIN       pin;
    enum IMGSENSOR_HW_PIN_STATE pin_state_on;
    u32  pin_on_delay;
    enum IMGSENSOR_HW_PIN_STATE pin_state_off;
    u32  pin_off_delay;
};
```

Power On Sequence

上电顺序：

1. 平台上电，初始化mipi switch
2. Sensor上电



1. 平台上电，初始化mipi switch

```
IMGSENSOR_HW_CFG imgsensor_custom_config[] = {
……
    {
        IMGSENSOR_SENSOR_IDX_SUB,
        IMGSENSOR_I2C_DEV_1,
        {
            {IMGSENSOR_HW_ID_REGULATOR, IMGSENSOR_HW_PIN_AVDD},
            {IMGSENSOR_HW_ID_REGULATOR, IMGSENSOR_HW_PIN_DOVDD},
            {IMGSENSOR_HW_ID_GPIO, IMGSENSOR_HW_PIN_DVDD},
            {IMGSENSOR_HW_ID_GPIO, IMGSENSOR_HW_PIN_PDN},
            {IMGSENSOR_HW_ID_GPIO, IMGSENSOR_HW_PIN_RST},
#ifdef MIPI_SWITCH
            {IMGSENSOR_HW_ID_GPIO, IMGSENSOR_HW_PIN_MIPI_SWITCH_EN},
            {IMGSENSOR_HW_ID_GPIO, IMGSENSOR_HW_PIN_MIPI_SWITCH_SEL},
#endif
            {IMGSENSOR_HW_ID_NONE, IMGSENSOR_HW_PIN_NONE},
        },
    },
    ...
}
```

1

```
struct IMGSENSOR_HW_POWER_SEQ
platform_power_sequence[] = {
#ifdef MIPI_SWITCH
{
    IMGSENSOR_SENSOR_IDX_NAME_SUB,
    {
        {
            IMGSENSOR_HW_PIN_MIPI_SWITCH_EN,
            IMGSENSOR_HW_PIN_STATE_LEVEL_HIGH,
            0,
            IMGSENSOR_HW_PIN_STATE_LEVEL_0,
            0
        },
        {
            IMGSENSOR_HW_PIN_MIPI_SWITCH_SEL,
            IMGSENSOR_HW_PIN_STATE_LEVEL_HIGH,
            0,
            IMGSENSOR_HW_PIN_STATE_LEVEL_0,
            0
        },
    }
},
```

2

3

```
static IMGSENSOR_RETURN gpio_set(
    void *pinstance,
    IMGSENSOR_SENSOR_IDX sensor_idx,
    IMGSENSOR_HW_PIN hw_pin,
    IMGSENSOR_HW_PIN_STATE pin_state)
{
    ...
}
```

2. Sensor上电

```
IMGSENSOR_HW_CFG imgsensor_custom_config[] = {
    {
        IMGSENSOR_SENSOR_IDX_MAIN,
        IMGSENSOR_I2C_DEV_0,
        {
            {IMGSENSOR_HW_ID_REGULATOR, IMGSENSOR_HW_PIN_AVDD},
            {IMGSENSOR_HW_ID_REGULATOR, IMGSENSOR_HW_PIN_DOVDD},
            {IMGSENSOR_HW_ID_REGULATOR, IMGSENSOR_HW_PIN_DVDD},
            {IMGSENSOR_HW_ID_REGULATOR, IMGSENSOR_HW_PIN_AFVDD},
            {IMGSENSOR_HW_ID_GPIO, IMGSENSOR_HW_PIN_PDN},
            {IMGSENSOR_HW_ID_GPIO, IMGSENSOR_HW_PIN_RST},
            {IMGSENSOR_HW_ID_NONE, IMGSENSOR_HW_PIN_NONE},
        },
    },
    {
        IMGSENSOR_SENSOR_IDX_SUB,
        IMGSENSOR_I2C_DEV_1,
        {
            {IMGSENSOR_HW_ID_REGULATOR, IMGSENSOR_HW_PIN_AVDD},
            {IMGSENSOR_HW_ID_REGULATOR, IMGSENSOR_HW_PIN_DOVDD},
            {IMGSENSOR_HW_ID_GPIO, IMGSENSOR_HW_PIN_DVDD},
            {IMGSENSOR_HW_ID_GPIO, IMGSENSOR_HW_PIN_PDN},
            {IMGSENSOR_HW_ID_GPIO, IMGSENSOR_HW_PIN_RST},
#ifdef MIPI_SWITCH
            {IMGSENSOR_HW_ID_GPIO, IMGSENSOR_HW_PIN_MIPI_SWITCH_EN},
            {IMGSENSOR_HW_ID_GPIO, IMGSENSOR_HW_PIN_MIPI_SWITCH_SEL},
#endif
            {IMGSENSOR_HW_ID_NONE, IMGSENSOR_HW_PIN_NONE},
        },
    },
    ...
}
```

```
static IMGSENSOR_RETURN
regulator_set(
    void *pinstance,
    IMGSENSOR_SENSOR_IDX sensor_idx,
    IMGSENSOR_HW_PIN hw_pin,
    IMGSENSOR_HW_PIN_STATE pin_state)
{
    ...
}
```

6

5

```
struct IMGSENSOR_HW_POWER_SEQ
sensor_power_sequence[] = {
#if defined(IMX398_MIPI_RAW)
{
SENSOR_DRVNAME_IMX398_MIPI_RAW,
{
{SensorMCLK, Vol_High, 0},
{AVDD, Vol_2800, 0},
{DOVDD, Vol_1800, 0},
{DVDD, Vol_1100, 0},
{AFVDD, Vol_2800, 0},
{PDN, Vol_Low, 0},
{PDN, Vol_High, 0},
{RST, Vol_Low, 0},
{RST, Vol_High, 1},
{VDD_None, Vol_Low, 0}
},
},
#endif
}
```

4

No power OFF state => Automatically set level_0 when power OFF.

How To Add A New Sensor

Step6 I2C Bus Config

<a>According to HW layout,Using dct tools to Config I2C device

dct path:\vendor\mediatek\proprietary\scripts\dct\ DrvGen.exe

Slave Device	Channel	Device Address
EXT_DISP	I2C_CHANNEL_1	0x39
MSENSOR	I2C_CHANNEL_1	0x0c
GYRO	I2C_CHANNEL_1	0x69
GSENSOR	I2C_CHANNEL_1	0x68
BAROMETER	I2C_CHANNEL_1	0x77
ALSPS	I2C_CHANNEL_1	0x51
HUMIDITY	I2C_CHANNEL_1	[illegible]
CAMERA_MAIN	I2C_CHANNEL_2	0x36
CAMERA_MAIN_AF	I2C_CHANNEL_2	0x72

表示main camera device 挂载到 I2C Bus2上面

同一条I2C bus的device Address不能冲突

dws path: kernel-4.4\drivers\misc\mediatek\dws\mt6763\

How To Add A New Sensor

Step6 I2C Bus Config

<b> Using DCT Tool to generate cust.dtsi

Check: \out\target\product\$project\obj\KERNEL_OBJ\arch\arm64\boot\dts\cust.dtsi

:

```
&i2c2 {
        #address-cells = <1>;
        #size-cells = <0>;
        clock-frequency = <400000>;
        mediatek,use-open-drain;
        camera_main@36 {
            compatible = "mediatek,camera_main";
            reg = <0x36>;
            status = "okay";
        };
        camera_main_af@72 {
            compatible = "mediatek,camera_main_af";
            reg = <0x72>;
            status = "okay";
        };
};
```

Name of match I2C driver

Device Address

Check Mipi Port & Mclk

/vendor/mediatek/proprietary/custom/mt6763/hal/imgsensor_src/cfg_setting_imgsensor.cpp

```
MINT32  getMipiSensorPort(EDevId const eDevId)
{
    switch  (eDevId)
    {
        case eDevId_ImgSensor0://main
            return  EMipiPort_CSI0;
        case eDevId_ImgSensor1://sub
            return  EMipiPort_CSI1;
        case eDevId_ImgSensor2://3d
            return  EMipiPort_CSI1;
        default:
            break;
    }
    return  -1;

}
```

配置mipi port

配置Sensor使用的MCLK

```
MINT32  getSensorMclkConnection(EDevId const eDevId)
{
    switch  (eDevId)
    {
        case eDevId_ImgSensor0://main
            return  eMclk_1;
        case eDevId_ImgSensor1://sub
            return  eMclk_2;
        case eDevId_ImgSensor2://3d
            return  eMclk_3;
        default:
            break;
    }
    return  -1;

}
```

Check Power On Status

```
546   546 D [  119.596461].(2)[546:cameraserver]: OV8856[feature_control] feature_id = 3107
546   546 E [  119.596631].(2)[546:cameraserver]: [imgsensor_hw_power]  curr_sensor_name=ov8856_mipi_raw sensor_idx=1
// sensor_idx=1, search sub sensor

546   546 E [  119.596635].(2)[546:cameraserver]: [imgsensor_hw_power_sequence] pwr_status=1
546   546 E [  119.596641].(2)[546:cameraserver]: [imgsensor_hw_power_sequence]  ppwr_info->pin=7 ppwr_info->pin_state_on=10 id=0
546   546 E [  119.596638].(2)[546:cameraserver]gpio_set: pinctrl err, PinIdx 7, Val 10
//platform powerOn [initialize mipi switch]

546   546 E [  119.601695].(2)[546:cameraserver]: [imgsensor_hw_power_sequence] pwr_status=1
546   546 E [  119.602179].(0)[546:cameraserver]: [imgsensor_hw_power_sequence] ppwr_info->pin=3 ppwr_info->pin_state_on=9 id=0
546   546 E [  119.602591].(0)[546:cameraserver]: [imgsensor_hw_power_sequence] ppwr_info->pin=5 ppwr_info->pin_state_on=7 id=0
546   546 E [  119.603026].(0)[546:cameraserver]: [imgsensor_hw_power_sequence] ppwr_info->pin=4 ppwr_info->pin_state_on=3 id=0
546   546 E [  119.603064].(0)[546:cameraserver]: [imgsensor_hw_power_sequence] ppwr_info->pin=2 ppwr_info->pin_state_on=0 id=1
546   546 E [  119.613125].(0)[546:cameraserver]: [imgsensor_hw_power_sequence] ppwr_info->pin=2 ppwr_info->pin_state_on=10 id=1
//sensor powerOn [AVDD/IOVDD/DVDD/PWD/RST]
```

PowerOn

PMIC

LDO

```
enum IMGSENSOR_HW_PIN {
  IMGSENSOR_HW_PIN_NONE = 0,
  IMGSENSOR_HW_PIN_PDN,
  IMGSENSOR_HW_PIN_RST,
  IMGSENSOR_HW_PIN_AVDD,
  IMGSENSOR_HW_PIN_DVDD,
  IMGSENSOR_HW_PIN_DOVDD,
  IMGSENSOR_HW_PIN_AFVDD,
#ifdef MIPI_SWITCH
  IMGSENSOR_HW_PIN_MIPI_SWITCH_EN,
  IMGSENSOR_HW_PIN_MIPI_SWITCH_SEL,
#endif
  IMGSENSOR_HW_PIN_MCLK1,
  IMGSENSOR_HW_PIN_MCLK2,
  IMGSENSOR_HW_PIN_MCLK3,
  IMGSENSOR_HW_PIN_MCLK4,

  IMGSENSOR_HW_PIN_MAX_NUM,
  IMGSENSOR_HW_PIN_UNDEF = -1
};
```

```
enum IMGSENSOR_HW_PIN_STATE {
  IMGSENSOR_HW_PIN_STATE_LEVEL_0,
  IMGSENSOR_HW_PIN_STATE_LEVEL_1000,
  IMGSENSOR_HW_PIN_STATE_LEVEL_1100,
  IMGSENSOR_HW_PIN_STATE_LEVEL_1200,
  IMGSENSOR_HW_PIN_STATE_LEVEL_1210,
  IMGSENSOR_HW_PIN_STATE_LEVEL_1220,
  IMGSENSOR_HW_PIN_STATE_LEVEL_1500,
  IMGSENSOR_HW_PIN_STATE_LEVEL_1800,
  IMGSENSOR_HW_PIN_STATE_LEVEL_2500,
  IMGSENSOR_HW_PIN_STATE_LEVEL_2800,
  IMGSENSOR_HW_PIN_STATE_LEVEL_HIGH,

  IMGSENSOR_HW_PIN_STATE_NONE = -1
};
```

How To Add A New Sensor

Step7 Add Sensor driver

根据各个接口函数，在sample code基础上进行修改。

- Kernel:

/kernel-4.4/drivers/misc/mediatek/imgsensor/src/${platform}/xxx_mipi_raw

- Hal:

/vendor/mediatek/proprietary/custom/ /${platform}//hal/imgsensor/xxx_mipi_raw

How To Add A New Sensor

Step8 Compile

- Modify projectConfig.mk，重新full build

  Eg: make clean && androidq make –j24 2>&1 tee | build.log
- Modify kernel part, remake kernel bootimage, 只下载boot.img即可

  Eg: mmm kernel-4.4:kernel && make bootimage-nodeps
- Modify hal part, remake android,只下载system.img或build libcameracustom.so push该so即可

  Eg: mmm vendor\mediatek\proprietary\custom\mt6797\hal

Note:编译的时候注意文件的优先级 project>platform