



**CONFIDENTIAL B**

# Basic Tuning Flow – PDC Check

# Outline

- PDC Enable
- PDC simulation

# PDC Enable

1、PDC\_EN = 1，Enable PDC

File name :sensorname\_scenario\_ISP\_RAW.cpp

Eg:ov13855mipiraw\_Scene\_Capture\_ISP\_RAW.cpp

```
|const ISP_NVRAM_BNR_PDC_T s5k3p9sxmipiraw_BNR_PDC_0000 = {  
|.con      = {.bits={.PDC_EN=0, .rsv_1=0, .PDC_CT=1, .rsv_5=0,
```

If you want to enable PDC , change PDC\_EN

From

.PDC\_EN=0

To

.PDC\_EN=1

# PDC Enable

2、 which scenario need enable PDC

2.1、 which scenario have PD pixels confirm from sensor driver :if `*(MUINT32 *) (uintptr_t) (*(feature_data+1)) = 1;` then Corresponding scenario have PD pixels

Eg: case SENSOR\_FEATURE\_GET\_SENSOR\_PDAF\_CAPACITY:

```
case SENSOR_FEATURE_GET_SENSOR_PDAF_CAPACITY:
```

```
LOG_INF("SENSOR_FEATURE_GET_SENSOR_PDAF_CAPACITY scenarioId:%d\n", (UINT16) *feature_data);
```

```
//PDAF capacity enable or not, 2p8 only full size support PDAF
```

```
switch (*feature_data) {
```

```
case MSDK_SCENARIO_ID_CAMERA_CAPTURE_JPEG: //enum value is 1
```

```
*(MUINT32 *) (uintptr_t) (*(feature_data+1)) = 1;
```

```
break;
```

```
case MSDK_SCENARIO_ID_VIDEO_PREVIEW: //enum value is 2
```

```
*(MUINT32 *) (uintptr_t) (*(feature_data+1)) = 1; // video & capture use same settin
```

```
break;
```

```
case MSDK_SCENARIO_ID_HIGH_SPEED_VIDEO:
```

```
*(MUINT32 *) (uintptr_t) (*(feature_data+1)) = 0;
```

```
break;
```

```
case MSDK_SCENARIO_ID_SLIM_VIDEO:
```

```
*(MUINT32 *) (uintptr_t) (*(feature_data+1)) = 0;
```

```
break;
```

```
case MSDK_SCENARIO_ID_CAMERA_PREVIEW: // binning size preview, enum value is 0
```

```
*(MUINT32 *) (uintptr_t) (*(feature_data+1)) = 1;
```

```
break;
```

```
case MSDK_SCENARIO_ID_CUSTOM1: //enum value is 5
```

```
*(MUINT32 *) (uintptr_t) (*(feature_data+1)) = 1;
```

```
break;
```

```
default:
```

```
*(MUINT32 *) (uintptr_t) (*(feature_data+1)) = 0;
```

```
break;
```

# PDC Enable

2、 which scenario need enable PDC

2.2、 Confirm enum value of the scenario from kd\_imgsensor\_define.h

Note: Theoretically, scenario should same with SensorMode in MT6771\_NVRAM\_IF\_Common.xlsx

SensorMode in MT6771\_NVRAM\_IF\_Common.xlsx

SensorMode
Preview
Capture
Video
SlimVideo1
SlimVideo2
Custom1
Custom2
Custom3
Custom4
Custom5

scenario

```
typedef enum {  
    MSDK_SCENARIO_ID_CAMERA_PREVIEW = 0,  
    MSDK_SCENARIO_ID_CAMERA_CAPTURE_JPEG,  
    MSDK_SCENARIO_ID_VIDEO_PREVIEW,  
    MSDK_SCENARIO_ID_HIGH_SPEED_VIDEO,  
    MSDK_SCENARIO_ID_SLIM_VIDEO,  
    MSDK_SCENARIO_ID_CUSTOM1,  
    MSDK_SCENARIO_ID_CUSTOM2,  
    MSDK_SCENARIO_ID_CUSTOM3,  
    MSDK_SCENARIO_ID_CUSTOM4,  
    MSDK_SCENARIO_ID_CUSTOM5,  
    /* Legacy scenario */  
    MSDK_SCENARIO_ID_CAMERA_ZSD,  
    MSDK_SCENARIO_ID_CAMERA_3D_PREVIEW,  
    MSDK_SCENARIO_ID_CAMERA_3D_CAPTURE,  
    MSDK_SCENARIO_ID_CAMERA_3D_VIDEO,  
    MSDK_SCENARIO_ID_TV_OUT,  
    MSDK_SCENARIO_ID_MAX,  
} MSDK_SCENARIO_ID_ENUM;
```

# PDC Enable

2、which scenario need enable PDC

2.3、confirm each mode's scenario (eg. normal preview, normal video, normal capture, wechat preview, wechat capture etc. mode)

Confirmation method :

method 1:

Search "setScenario" from mail log, Scenario id is the enum value of the scenario

```
ImgSensorDrv: [setScenario][setScenario]DevID = 0, m_LineTimeInus = 9086  
Scenario id = 2, PixelClk = 560000000, PixelInLine = 5088, Framelength =  
3668
```

method 2:

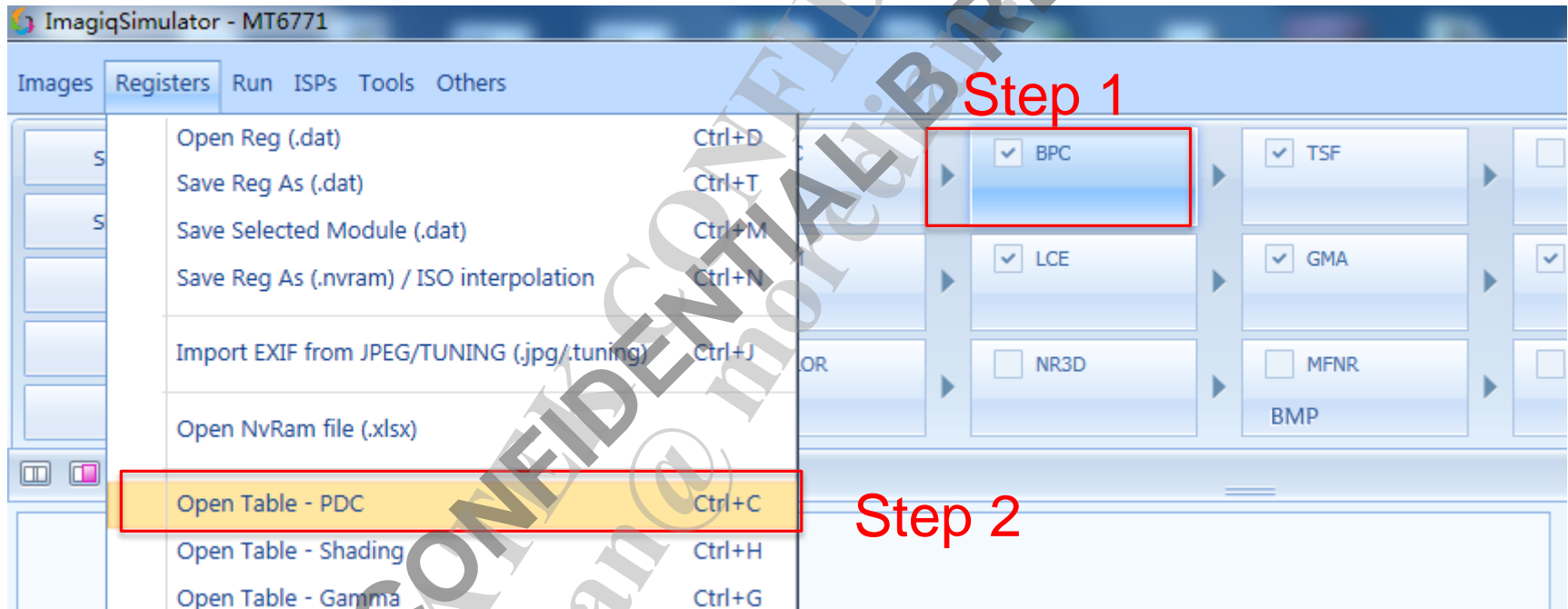
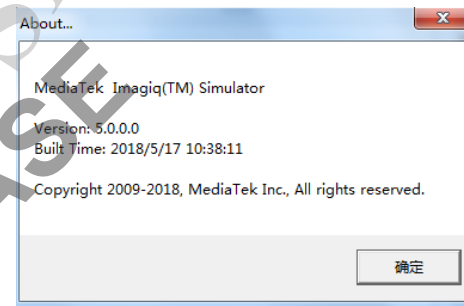
Search "SM" from mail log, (need cmd:adb shell setprop debug.mapping\_mgr.enable 1 before recording log)

```
MtkCam/MappingMgr: [query] [Dev:1-Mod:BNR_PDC(5)] (Idx 44) (PF Preview, SM Video, Bin 0, P2 1, FLASH  
0, APP Video, FD 0, ZOOM 0, LV 4, CT 8, ISO 4, CUSTOM 0)
```

Note: according to 2.2 ,we know Scenario id = 2 means SM Video

# PDC simulation

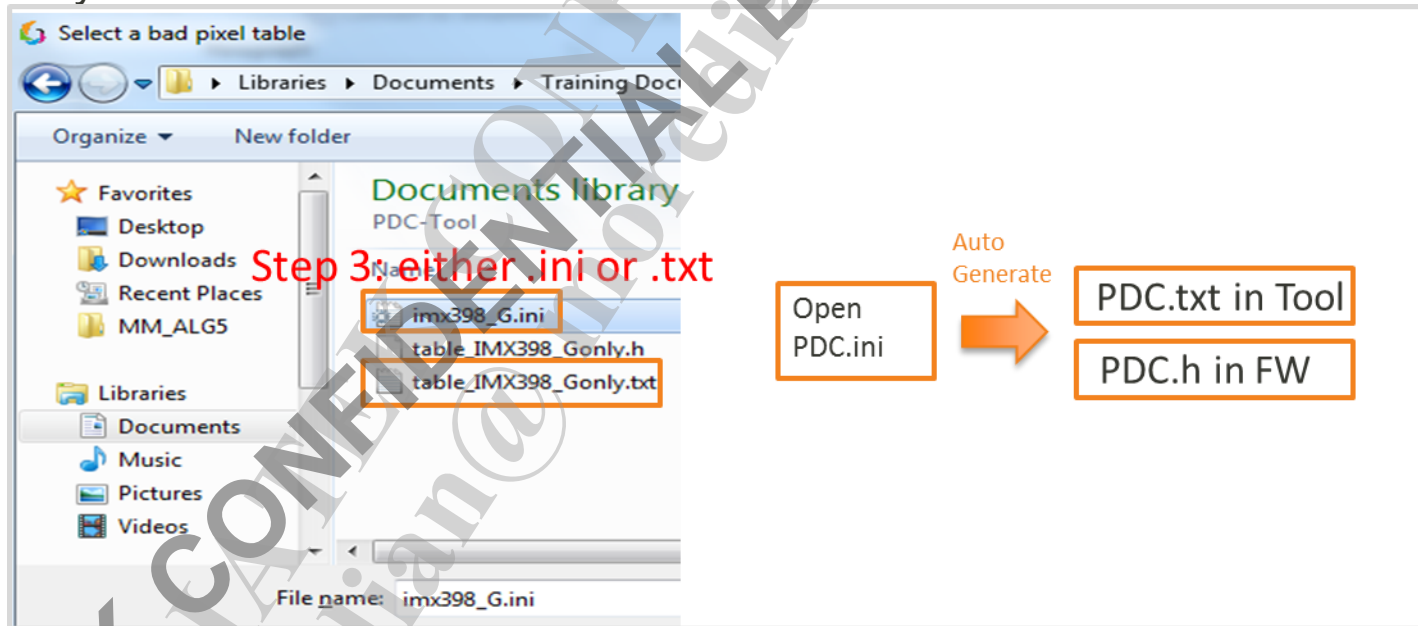
- Step 1: Active BPC module
- Step 2: Open PDC description file in PDC.ini or PDC.txt file type.



Note: A PD Pixel Description File in **.ini file** type should request to sensor vendor

# PDC simulation

- Step 3:
  - Opening PDC.ini file is required at the first time, tool would generate a PDC.txt file which could comply with Imadiq Simulator.
  - Next you could open .txt been stored into the same folder as PDC.ini once PDC Table in .txt file type is generated automatically.



Note:

- 1、PDC.h content no need to copy to FW file anymore
- 2、when simulate, please load .ini, or load .txt automatic generated by tool
- 3、when simulate, PDC En is read from dump exif instead of tool UI. if PDC is enable in exif, then a PDC table is required when simulate



# PDC simulation

- Step 4: Activate PDC Enable and select Mode as requirement by following below info.

PDC CON	
PDC OUT	<input type="checkbox"/>
PDC MODE	1
PDC CT	<input checked="" type="checkbox"/>
PDC EN	<input checked="" type="checkbox"/>

Step 4:

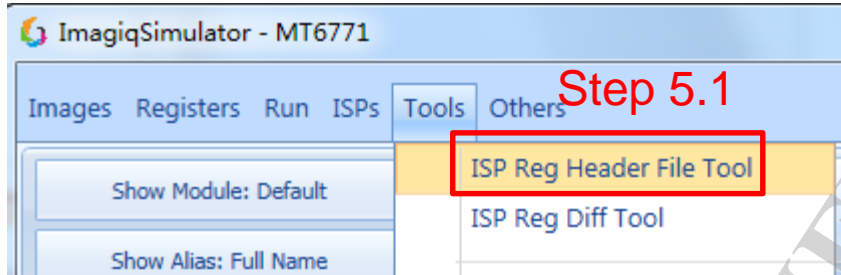
- PDC Enable
- Mode 1

PDC GAIN R4		PDC SL	
PDC GCF R03	0	PDC GCF NORM	13
PDC GCF R12	0	PDC ASL	7
PDC TH GB		PDC ISL	7
PDC BTH	40	PDC BSL	6
PDC GTH	4095	PDC GSL	8
PDC TH IA		PDC POS	
PDC ATH	4095	PDC YCENTER	0
PDC ITH	0	PDC XCENTER	0
PDC TH HD			
PDC DTH	512		
PDC NTH	80		

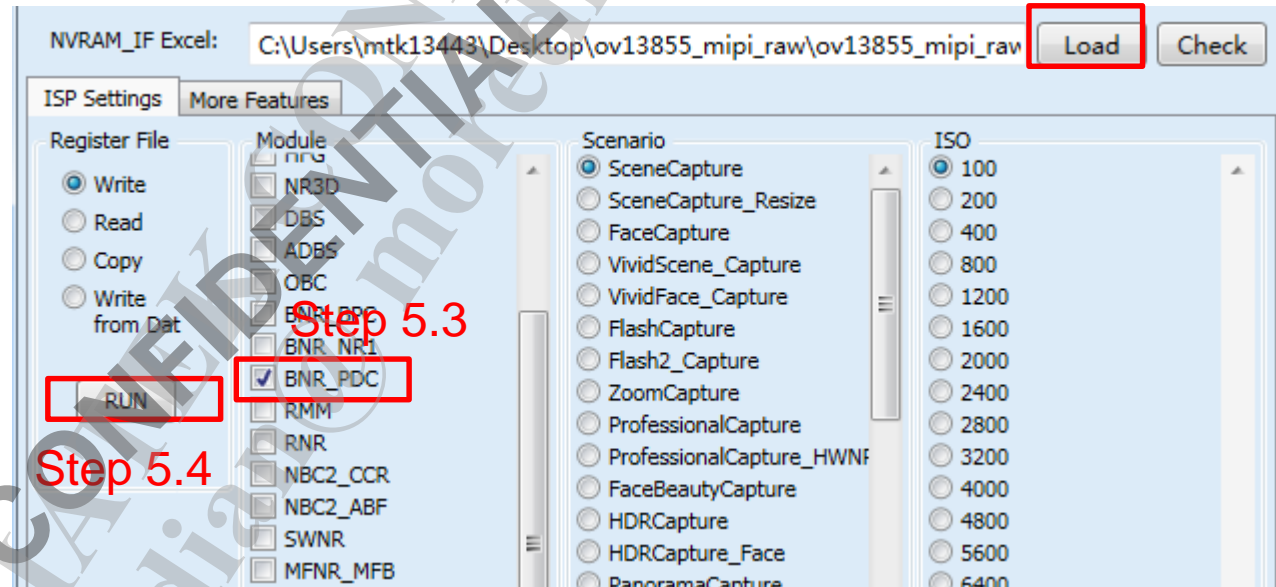
- Ensure PDC param it's correct, if not sure, use ImadiqSimulator default PDC param

# PDC simulation

- Step 5: Update ISP PDC Params.



Step 5.2



# PDC simulation

## Step 6: Ensure it's correct in Reg File

Eg: ov13855mipiraw\_Scene\_Capture\_ISP\_RAW.cpp

```
const ISP_NVRAM_BNR_PDC_T ov13855mipiraw_BNR_PDC_0000 = {  
    .con      = {.bits = {.PDC_EN=1, .rsv_1=0, .PDC_CT=0, .rsv_5=0, .PDC_MODE=1, .rsv_10=0, .PDC_OUT=0, .rsv_17=0}},  
    .gain_l0  = {.bits = {.PDC_GCF_L00=241, .rsv_12=0, .PDC_GCF_L10=0, .rsv_28=0}},  
    .gain_l1  = {.bits = {.PDC_GCF_L01=0, .rsv_12=0, .PDC_GCF_L20=0, .rsv_28=0}},  
    .gain_l2  = {.bits = {.PDC_GCF_L11=0, .rsv_12=0, .PDC_GCF_L02=0, .rsv_28=0}},  
    .gain_l3  = {.bits = {.PDC_GCF_L30=0, .rsv_12=0, .PDC_GCF_L21=0, .rsv_28=0}},  
    .gain_l4  = {.bits = {.PDC_GCF_L12=0, .rsv_12=0, .PDC_GCF_L03=0, .rsv_28=0}},  
    .gain_r0  = {.bits = {.PDC_GCF_R00=256, .rsv_12=0, .PDC_GCF_R10=0, .rsv_28=0}},  
    .gain_r1  = {.bits = {.PDC_GCF_R01=0, .rsv_12=0, .PDC_GCF_R20=0, .rsv_28=0}},  
    .gain_r2  = {.bits = {.PDC_GCF_R11=0, .rsv_12=0, .PDC_GCF_R02=0, .rsv_28=0}},  
    .gain_r3  = {.bits = {.PDC_GCF_R30=0, .rsv_12=0, .PDC_GCF_R21=0, .rsv_28=0}},  
    .gain_r4  = {.bits = {.PDC_GCF_R12=0, .rsv_12=0, .PDC_GCF_R03=0, .rsv_28=0}},  
    .th_gb    = {.bits = {.PDC_GTH=4095, .rsv_12=0, .PDC_BTH=40, .rsv_28=0}},  
    .th_ia    = {.bits = {.PDC_ITH=0, .rsv_12=0, .PDC_ATH=4095, .rsv_28=0}},  
    .th_hd    = {.bits = {.PDC_NTH=80, .rsv_12=0, .PDC_DTH=512, .rsv_28=0}},  
    .sl       = {.bits = {.PDC_GSL=8, .PDC_BSL=6, .PDC_ISL=7, .PDC_ASL=7, .PDC_GCF_NORM=13, .rsv_20=0}},  
}
```

(Param content Example)