ESWIN

ISP Developer Manual

EIC7x Series AI Digital SoC

Engineering Draft / Rev 0.5

2025-7-7

Statement

Revision History

Version	Date	Modifier	Instructions
V0.1	2025/6/13	Zhenya Shen	Initial draft
V0.2	2025/6/27	Shangjuan Wei	Added TPG and crop modules
V0.3	2025/7/4	Fangxian Yu	Added VI overall structure and dvp2axi explanation
V0.4	2025/7/11	Zhenya Shen	Updated isp operation process
V0.5	2025/7/7	Fangxian Yu	Updated device tree configuration

Contents

List of Figures

List of Tables

1. Overview

1.1 Basic Introduction

ISP (Image Signal Processor) is a dedicated processor or module for real - time processing and optimization of image/video signals.

1.2 VI Structure

1.2.1 Hardware Path

The EIC7700 chip contains 6 combo cdphy, and the current version has implemented the configuration of dphy.

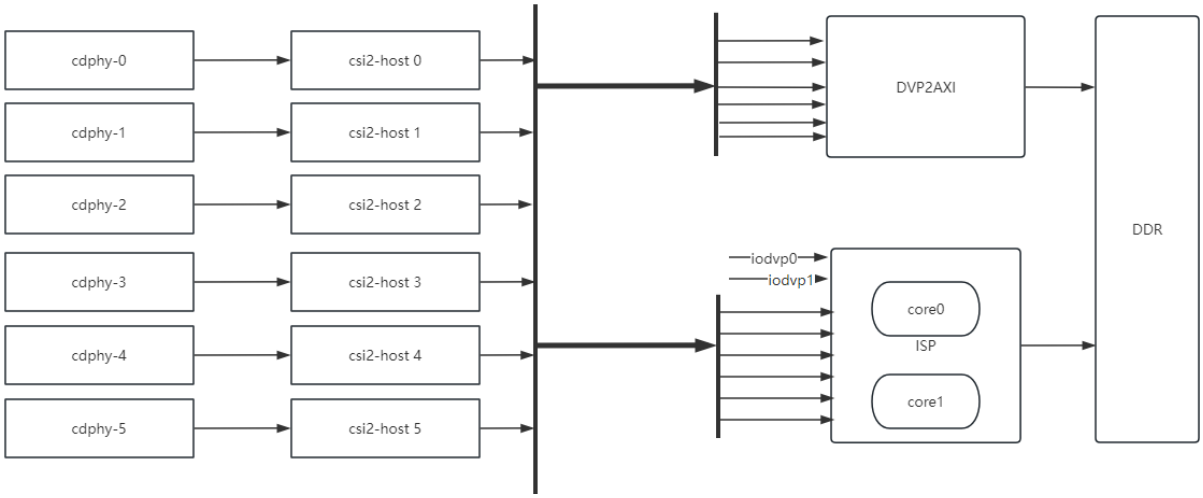

Figure 1 - 1 VI Structure

VI includes two different pipelines:

sensor->phy->csi host ->dvp2axi, Currently, dvp2axi is mainly used as a debugging tool to dump raw data.

sensor->phy-csi host->isp

1.2.2 Multi-sensor Support

EIC7700 currently supports two modes: 6x2lanes and 3x4lanes.

In these two modes, dvp2axi supports a single - path maximum of 4k@60 raw10, isp single - core supports a maximum of 4k@60 raw10, and when a single isp is connected to multiple paths, the performance will be prorated.

The 6x2lanes mode is shown in Figure 1 above, where each phy has a corresponding csi, and each csi is connected to a path of dvp2axi or isp, supporting up to 6 2lanes sensors. The 3x4lanes mode combines two phy into one phy to form a 4lane, as shown in Figure 2 below. This mode supports up to 3 4lane sensors.

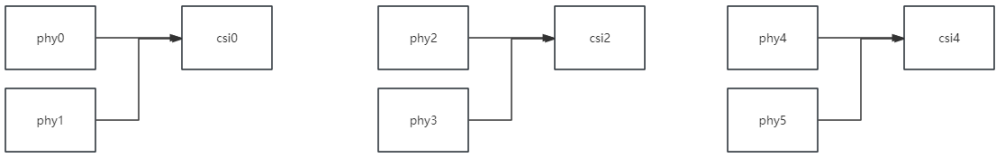

Figure 1 - 2 3x4lanes Mode

2. CIS Driver

2.1 Device Registration

2.1.1 MIPI Interface

For the eic7700 platform, there are 6 independent mipi cdphy, and dphy is currently adapted. Dphy supports up to 4lanes, combined by two phy. The maximum rate is 3.5Gbps/lane.

The specific dts configuration of VI can refer to the xxx - imx219.dtsi in the kernel. VI can include two chains, the first is sensor->dphy->csi->isp, and the second is sensor->dphy->csi->dvp2axi_mipi_lvds (only raw images can be captured).

For the new 4lane sensor device tree, refer to the existing device tree for the port connection relationship. Since 4lane combines two phy into one connected to a csi2 host, the single - number csi2_dphy_hw_x and csi2_dphy_x need to be disabled, and the single - number mipi_csi2 also needs to be disabled. Dvp2axi_mipi_lvds here corresponds to csi, but in the registration stage, all need to be turned on, only the video devices created by the odd - number dvp2axi will not be connected to the csi2 module. ISP has no such restrictions. Other points to note are as follows.

```
&vi_subsys{
    phy_mode = <5>; //select mipi interface as 4 lane mode
}
&csi2_dphy_hw {
    lanes = <1 2 3 4>; //4lanes
    lanes-dp-dn = <0 0 0 0>;// 4lanes: positive on dn/ negative on dp to hs and lp mode
    status = "okay";
}
&csi2_dphy {
    status=okay;
}
&mipi_csi2_0{
    num_lanes = <4>; //4lanes
    snps,en-ppi-width = <1>; // 1: 16-bit, 0: 8-bit
    ports {
        #address-cells = <1>;
        #size-cells = <0>;
        port@0 {
```

```
            reg = <0>;
            mipi_csi2_input: endpoint {
                remote-endpoint = <&csidphy0_out>;
                data-lanes = <1 2>;
            };
        };
        port@1 {
            reg = <1>;
            mipi_csi2_output: endpoint {
                remote-endpoint = <&axi2dvp_mipi2_in0>; //port 1 is connected to dvp2axi
            };
        };
        port@2 {
            reg = <2>;
            mipi_csi2_output1: endpoint {
                remote-endpoint = <&vvcam_isp_input_port>; //port2 is connected to isp
            };
        };
    };
}
```

For a 4-lane MIPI interface using a 2-lane sensor, use dphy0/2/4, csi2-0/2/4, and dvp2axi_mipi_lvds-0/2/4. The other odd-numbered modules of dphy and csi2 can be disabled. However, all dvp2axi_mipi_lvds modules need to be enabled, although the created video devices will have no connections. Pay attention to the following points when modifying the device tree.

```
&vi_subsys{
    phy_mode = <2>; //select mipi interface as 2lane mode
}

&csi2_dphy_hw {
    lanes = <1 2>; // phy change to 2lanes
    lanes-dp-dn = <0 0 >;  //2lanes: positive on dn/ negative on dp to hs and lp mode
    status = "okay";
}

&mipi_csi2_0{
    num_lanes = <2>; //csi2 host change to 2lanes
}
```

If the board has a 2-lane MIPI interface, the dphy, csi2, and dvp2axi_mipi_lvds correspond one-to-one, creating the link sequentially.

```
Example
First path：
Sensor0->dphy0->csi2-0->dvp2axi_mipi_lvds0
```

```
Sensor0->dphy0->csi2-0->isp0
Sencond path：
Sensor1->dphy1->csi2-1->dvp2axi_mipi_lvds1
Sensor1->dphy1->csi2-1->isp1（The multipath feature of the ISP is still being improved.）
```

The key considerations for the sensor lie in the clock frequency and configuring the port with the corresponding number of lanes and link-frequency. The sensor node reference for a 4-lane IMX219 is provided below. Note that assigned-clock-rates = <24000000>; Currently, the EIC7700 clock divider supports 6~64MHz and three fixed frequency points: 37.125/74.25/54MHz. When using clock frequencies other than these three fixed points, the divided clock may not fully match. For example, the IMX219 requires 24000000 here, but the actual divided output may be 23750000. Therefore, modifications to the clk_rate judgment in the driver are necessary to avoid mismatches with the driver's clock, which could cause sensor registration to fail.

```
imx219_0: imx219@10 {
    compatible = "sony,imx219";
    status = "okay";
    address-cells = <1>;
    size-cells = <0>;
    reg = <0x10>;
    clocks = <&d0_clock EIC7700_CLK_VI_SHUTTER_0>;
    assigned-clocks = <&d0_clock EIC7700_CLK_VI_SHUTTER_0>;
    assigned-clock-rates = <24000000>;
    clock-names = "xvclk";
    pinctrl-names = "default";
    pinctrl-0 = <&pinctrl_gpio78_default>;
    reset-gpio = <&portc 14 GPIO_ACTIVE_HIGH>;

    port {
        imx219_out2: endpoint {
            remote-endpoint = <&mipidphy2_in_ucam0>;
            data-lanes = <1 2 3 4>;
            link-frequencies = /bits/ 64 <364000000>;
        };
    };
};
```

Currently, the DT overlay feature has not been merged into Eswin Linux. Replacing different sensors requires recompiling the Linux kernel.

3. Instructions for Use

3.1 ISP

3.1.1 Start isp media server

```
isp_media_server 0 &
```

Configure ISP algorithm (using imx219 as an example)

```
echo 0 input_type=sensor mode=0 xml=/usr/lib/firmware/eic7x/isp/IMX219_3280x2464.xml
manu_json=/usr/lib/firmware/eic7x/isp//vvbcfg/project_json_file/manual_ext.json
auto_json=/usr/lib/firmware/eic7x/isp/vvbcfg/project_json_file/auto.json > /proc/vsi/isp_subdev0
```

3.1.2 Start ISP

```
v4l2-ctl -d /dev/video6 --set-fmt-video=width=3280,height=2464,pixelformat=NV12 --stream-
mmap=10 --stream-skip=2 --stream-to=dump.yuv --stream-count=10 --stream-poll
```

3.2 DVP2AXI

The Dvp2axi hardware aligns memory to 256 bytes, so the width of the dumped data will be greater than the image width. For example, the command reference for IMX219 is as follows. After multi-path support is added, use `media-ctl -p` to check the video device number used by the dvp2axi device.

```
v4l2-ctl -d /dev/video0 --set-fmt-video=width=3280,height=2464,pixelformat=RG10 --stream-
mmap=8    --stream-skip=2 --stream-to=data.raw --stream-count=20 –stream-poll
```

3.3 SENSOR

The sensor provides the functionality to manually set some parameters, but for different sensors, some parameter settings may be overwritten when the stream starts. Refer to the following for specific configuration methods.

```
echo 1 > /sys/modules/<sensor_name>/parameters/debug    // Enable manual settings function
v4l2-ctl -d /dev/v4l-subdev2 --list-ctrls (Assuming v4l-subdev2 is a sensor) //View the ctrl ops
supported by the sensor
v4l2-ctl -d /dev/v4l-subdev2 --set-ctrl “horizontal_blanking=10000" //set horizontal_blanking
```

4. API reference

The ISP supports V4L2 (Video for Linux 2). Below are some reference commands for v4l2-ctl. After multi-path support is added, the video device number of the ISP may change. Use `media-ctl -p` to check the specific video device number corresponding to `vvcam-video.0.0`.

Start the data flow API:

```
v4l2-ctl -d /dev/video4 --set-fmt-video=width=3280,height=2464,pixelformat=NV12 --stream-
mmap=4
```

Capture API (Capture one frame NV12):

```
v4l2-ctl -d /dev/video4 --set-fmt-video=width=3280,height=2464,pixelformat=NV12 --stream-
mmap=4 --stream-to=r0.bin --stream-count=1 --stream-poll
```

Algorithm Control API (Query and Control Algorithm):

```
v4l2-ctl -d /dev/video4 --list-ctrls
v4l2-ctl -d /dev/video4 --set-ctrl=isp_awb_enable=0
```

5. TPG And Crop

5.1 TPG Mode

5.1.1 Overview

TPG (Test Pattern Generator) mode is a test pattern primarily used for development, debugging, and system testing. If using TPG mode, assign the value 2 to the input_type parameter when configuring the ISP algorithm.

5.1.2 TPG Mode Figure

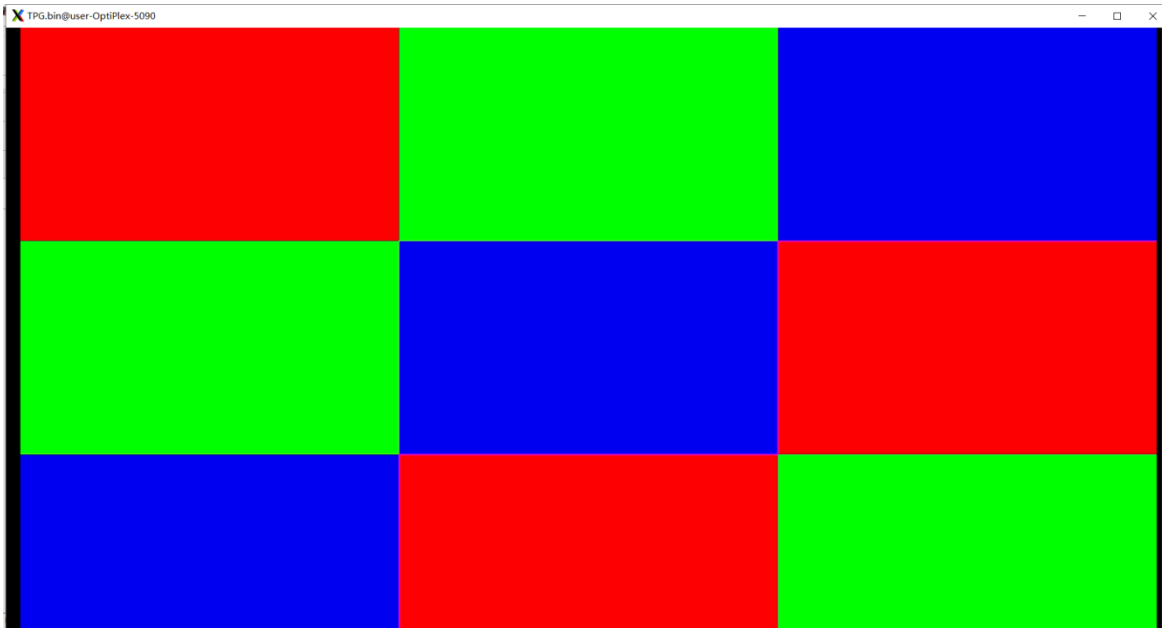

Figure 5-1 checkerboard

5.2 crop

```
v4l2-ctl -d /dev/video4 \
--set-selection=target=crop,width=1280,height=720,top=180,left=320 \      # input crop
--set-fmt-video=width=720,height=576,pixelformat=NV12 \                   # output Scale
--stream-mmap=10 --stream-to=cropped.bin --stream-count=1 --stream-poll
```

Principle: Original Image -> Crop (1280x720) -> Scale (720x576) -> Output (NV12 Format) -> Save to File cropped.bin.

The cropping area must not exceed the original resolution range of the sensor. For example, if the original resolution is 1920x1080, then: top(180) + height(576) <= 1080. left(320) + width(720) <= 1920.

6. ISP proc Debug Information Explanation

The debugging information utilizes the proc filesystem in Linux, which can reflect the real-time operating status of the current system. The recorded information can be used for problem localization and analysis.

6.1 View Method

When ISP related programs are running, use the'cat'command on the console to view information, such as cat/doc/vsi/isp0.

6.2 Debug information and parameter description

Debug infomation as follow:

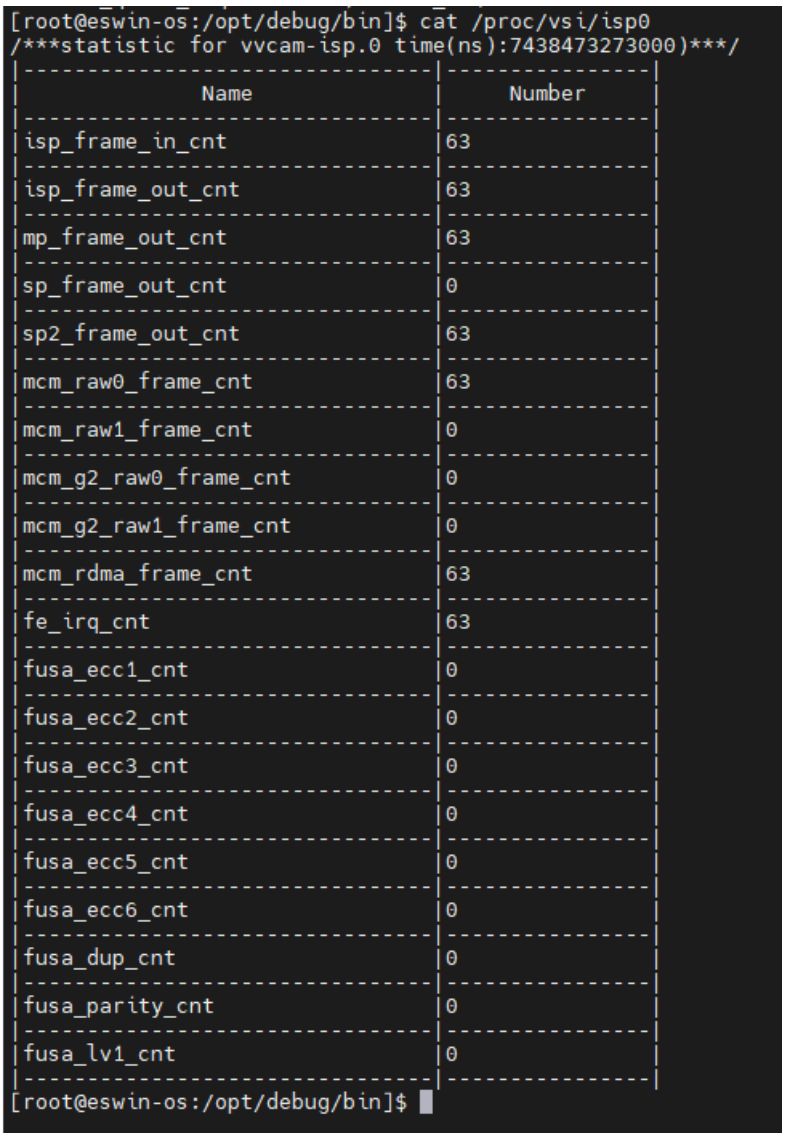

Figure 6-1 ISP Debug Info

Table 6-1Parameter Meaning

Parameter	Meaning
Name	Interrupt Name
Number	Interrupt Number

7. Appendix

Table 7-1 Sensor Support List

Sensor	Lanes	Phy
IMX219	2lanes/4lanes	MIPI DPHY
IMX327	4lanes	MIPI DPHY
IMX415	4lanes	MIPI DPHY
IMX586	4lanes	MIPI DPHY