

Intel® Atom™ Processor E3800 Product Family Image Signal Processor (ISP)

(Fedora* 18 Release)

User Guide

July 2015



By using this document, in addition to any agreements you have with Intel, you accept the terms set forth below.

You may not use or facilitate the use of this document in connection with any infringement or other legal analysis concerning Intel products described herein. You agree to grant Intel a non-exclusive, royalty-free license to any patent claim thereafter drafted which includes subject matter disclosed herein.

INFORMATION IN THIS DOCUMENT IS PROVIDED IN CONNECTION WITH INTEL PRODUCTS. NO LICENSE, EXPRESS OR IMPLIED, BY ESTOPPEL OR OTHERWISE, TO ANY INTELLECTUAL PROPERTY RIGHTS IS GRANTED BY THIS DOCUMENT.

Intel may make changes to specifications and product descriptions at any time, without notice. Designers must not rely on the absence or characteristics of any features or instructions marked "reserved" or "undefined." Do not finalize a design with this information.

The products described in this document may contain design defects or errors known as errata which may cause the product to deviate from published specifications. Current characterized errata are available on request.

Contact your local Intel sales office or your distributor to obtain the latest specifications and before placing your product order.

Copies of documents which have an order number and are referenced in this document, or other Intel literature, may be obtained by calling 1-800-548-4725, or go to: <http://www.intel.com/design/literature.htm>.

Intel, the Intel logo, Atom, Celeron, and Pentium are trademarks of Intel Corporation in the U.S. and/or other countries.

*Other names and brands may be claimed as the property of others.

See Trademarks on intel.com for full list of Intel trademarks.

© 2015, Intel Corporation



Contents

1	Introduction	6
	1.1 Terminology	6
	1.2 Reference Documents	7
2	Driver Information	8
3	System Requirements	9
4	Enabled Features	11
5	Disabled Features	12
6	Fixed Issues and New Features	13
	6.1 Fixed Issues	13
	6.2 New Features	13
7	Known Issues and Limitations	14
	7.1 Known Issues	14
	7.2 Limitations	14
8	Using ISP in Linux*	16
	8.1 Using ISP on the Intel CRB	16
	8.2 Verify the ISP Drivers are Properly Loaded	16
9	Test Application	18
	9.1 Building the Test Application	19
	9.2 Using the Test Application	19
	9.3 Supported Color Formats for -c Flag	22
	9.4 Supported Color Formats for -C Flag	22
	9.5 Main and Viewfinder Views	23
	9.6 Streaming With Viewfinder Turned On	24
	9.7 Limitations	25

Figures

Figure 1.	Annville Add-in Card	9
Figure 2.	File structure of Shader fragment program	23

Tables

Table 1.	Terminology	6
Table 2.	Reference Documents	7
Table 3.	Enabled Features	11
Table 4.	Supported MIPI ports for Bayley Bay or Bakersport CRBs	23

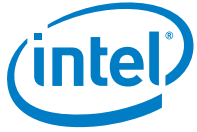


Table 5. Main and Viewfinder Sensor Settings Required for Bayley Bay CRB 24



Revision History

Date	Revision	Description
July 2015	003	Removed mentions of "Timesys*" throughout the document.
July 2015	002	Intel® Atom™ Processor E3800 Product Family Image Signal Processor (ISP) Release. <ul style="list-style-type: none">• Updated the dependency in Chapter 3. System Requirements.• Updated Chapter 8.2 Verify the ISP Drivers are Properly Loaded with new driver messages.• Updated application commands and description in Chapter 9. Test Application.• Updated the test application streaming commands in Chapter 9.5 Main and Viewfinder Views.
October 2014	001	Initial release.

§



1 Introduction

This User Guide describes the typical hardware system requirements, features, testing, and use of the Intel® Atom™ Processor E3800 Product Family (formerly known as Bay Trail-I) Image Signal Processor (ISP).

The ISP driver is available for the Fedora* 18 release on the 32- and 64-bit Intel Atom Processor E3800 Product Family platforms.

1.1 Terminology

Table 1. Terminology

Term	Description
AIC	Add-in Card
BIOS	Basic Input/Output System
CRB	Customer Reference Board
GPU	Graphic Processing Unit
GUI	Graphical User Interface
I/O	Input/Output
ioctl	Input/output control
IOTG	Internet of Things Group
ISP	Intel® Atom™ Image Signal Processor
PCI	Peripheral Connect Interface
SKU	Stock Keeping Unit
SoC	System-on-a-Chip



1.2 Reference Documents

Table 2. Reference Documents

Document	Document No./Location
<i>IOTG – Bay Trail Dashboard</i>	522355
<i>Intel® Atom™ Processor E3800 Product Platform Bayley Bay-I – Customer Reference Board User Guide</i>	523922
<i>Intel® Atom™ Processor E3800 Product Family – Bakersport-I Customer Reference Board (CRB) User Guide</i>	536497
<i>Intel® Atom™ Processor E3800 Product Family – Bayley Bay-I Customer Reference Board (CRB) User Guide</i>	523922
<i>Intel® ATOM™ Processor E3800 Product Family - Bakersport - Customer Reference Board Layout</i>	526062
<i>Intel® Atom™ Processor E3800 Product Family & Intel® Celeron™ Processor N2920 & Intel® Celeron™ Processor J1900 – Bayley Bay- I - Customer Reference Board Layout</i>	538049

S



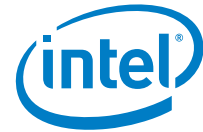
2 *Driver Information*

The Intel® Atom™ Image Signal Processor (ISP) driver is available for both 32- and 64-bit systems and is part of Fedora* 18 Live Image*. Details of the Linux* driver are listed below:

- ISP Linux* Kernel driver
- Two ISP firmware files located in directory **/lib/firmware**.
 - `iaisp_2400_css.bin.big`
 - `iaisp_2400_css.bin.small`

For more information refer to [Table 2](#), *IOTG Bay Trail Dashboard*, Document # 522355 on downloading the Live Image.

§



3 System Requirements

The Intel® Atom™ Image Signal Processor (ISP) driver supports the following platforms:

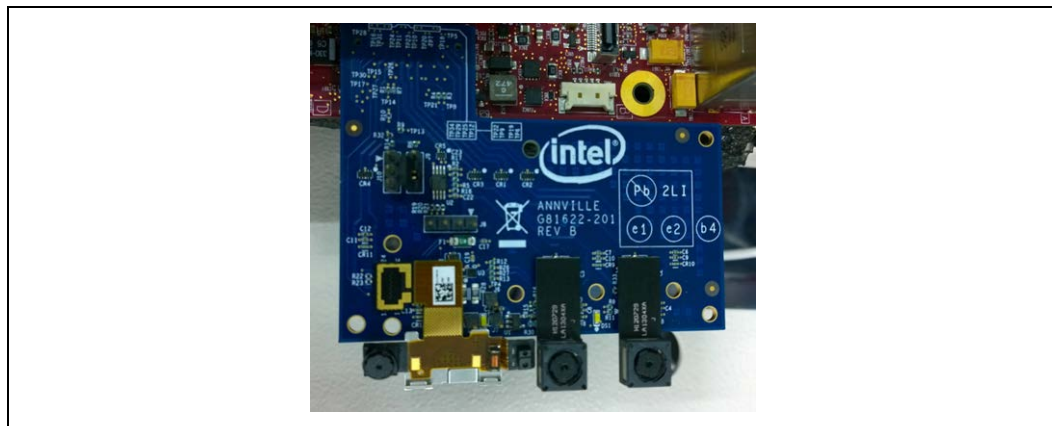
- System-on-a-Chip (SoC)
 - Intel® Atom™ Processor E3800 Product Family (Product Stepping D0)
 - Platform:
 - Bakersport Customer Reference Board (CRB), Fab B

Refer to [Table 2](#), *Intel® ATOM™ Processor E3800 Product Family - Bakersport - Customer Reference Board Layout*, Document # 526062, for detailed information. Rework is required to make the ISP functional.
 - Bayley Bay CRB, Fab 3, Revision 03
- Refer to [Table 2](#) *Intel® Atom™ Processor E3800 Product Family & Intel® Celeron™ Processor N2920 & Intel® Celeron™ Processor J1900 – Bayley Bay- I - Customer Reference Board Layout*, Document # 538049, for detailed information. Rework is required to make the ISP functional.

Additional required hardware for the ISP:

- Annville Add-in Card (AIC) for ISP has the following components
 - Two units of OmniVision* 5640 (OV5640) Image Sensor
 - One unit of Aptina* MT9M114 Image Sensor

Figure 1. Annville Add-in Card



- Dependency
 - Intel® Atom™ Processor E3800 Product Family platform on Fedora* 18 with Linux* kernel 3.10 long term support initiative (LTSI) (32-bit/64-bit).



- Constraint
N/A

§



4 Enabled Features

The Intel® Atom™ Image Signal Processor (ISP) has the following features enabled as outlined in [Table 3](#).

Table 3. Enabled Features

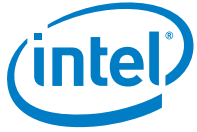
Input ¹ from Sensor	Output from ISP
YUV422	YV16
RGB565	RGB565
RGB888 ²	RGB888
YUV422	RGB888
YUV422	NV12
RGB888	NV12
RGB565	NV12
BA10	NV12

NOTES:

1. Format that the input sensor supports.
2. The hardware listed in [System Requirements](#) does not support the RGB888 format but the drivers and firmware do.

The ISP has a minimum resolution (as tested) of 640x480 and a maximum resolution of 1920x1080 (1080p).

§



5 *Disabled Features*

Three (3) simultaneous streams feature is no longer supported.

§



6 Fixed Issues and New Features

The following are fixed issues and new features for the Intel® Atom™ Image Signal Processor (ISP).

6.1 Fixed Issues

Video pipe and Viewfinder pipe are enabled.

Fastboot and multistream – are module parameters that enable users to start up the driver in several different ways to save time or enable all features at startup depending on requirement.

6.2 New Features

EXPBUFF Input/output control (ioctl) enabled – allows buffers allocated by the ISP to be shared with the Intel® Embedded Media and Graphics Driver.

New VIDIOC_SUBDEV_S_FMT ioctl enabled – allows applications to individually set the I/O formats.

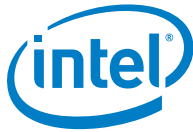
Special ioctl called ATOMISP_FW_LOAD added – switches the firmware from small to large in the case of fastboot.

Optimized ia_css_init and ia_css_unit – reduces the time taken by `atomisp_reset` from ~1.2 to 50 ms.

Released new test application – refer to ISP Test Application

Support for 5MP (2560 x 1920) UYVY format

§



7 Known Issues and Limitations

The following are the known issues and limitations that affect the Intel® Atom™ Image Signal Processor (ISP).

7.1 Known Issues

The following are known issues:

- Aptina* MT9M114 image sensor:

The Aptina MT9M114 image sensor is a one-lane (x1) sensor and can only support up to 1280 x 960 resolution.

Note: The test application limits the resolution to 1280x720, or 720p, only.

- The Preview, or Viewfinder modes. Refer to Section 9.4 for more detail.

Note: These modes are only supported with the Aptina image sensor on **NV12** and **BA10** color format combinations.

- The single-stream feature stops streaming from any sensor when using the VIDEO mode after 40+ hours or 3,883,333 frames, whichever comes first.
- The following issues may appear if the `-DCOLOR_CONVERSION` flag is not used during the `atomisp_testapp` build:
 - When using the RGBP or RGB565 color formats, frames are shown corrupted regardless of sensors and resolutions.
 - The OmniVision 5640 (OV5640_2) image sensor and Aptina MT9M114 image sensor intermittently refuse to start streaming or stream at 0.5 fps.
 - When three simultaneous streams are running after about 20,000 frames, all three streams time out. The issue is under investigation.

7.2 Limitations

The following are ISP limitations:

- Only **YV16**, **NV12**, **RGB565**, and **RGB888** are supported as output streams from the firmware.
- ISP FRAME BUFFER WIDTH 32 MULTIPLE:
 - The ISP requires a number of frame buffers for processing incoming images. The frame buffers can be allocated by an application and passed as buffer pointers to the ISP firmware through the ISP v412 driver. The frame memory size is calculated based on incoming image resolution width and height. For example, when an incoming image width is x and height is y , the application is responsible to allocate a memory size:

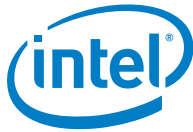
The ISP frame memory width needs to be a multiple of 32.

Known Issues and Limitations



Frame buffer size = ((int) x/32 + 32: 0 if x%32 > 0) * 32 * y * bytes/pixel

§



8 Using ISP in Linux*

8.1 Using ISP on the Intel CRB

When using the Intel® Atom™ Image Signal Processor (ISP) on the Intel CRB, The Internet of Things Group (IOTG) reference BIOS must have the correct ISP settings.

Configure the IOTG reference BIOS “Uncore Configuration” as follows:

1. Set the ISP Enabled/Disabled to **Enabled**.
2. Set the ISP PCI Device to **B0D3F0**.

8.2 Verify the ISP Drivers are Properly Loaded

To verify the ISP Drivers are properly loaded, the following hardware must be present:

- Two (2) of the OmniVision* OV5640 5-megapixel SoC Image Sensors are required (Part Number OV5640-A71A) available at: <http://www.ovt.com/products/sensor.php?id=93&limit=220>
- One (1) Aptina* MT9M114 Digital Image Sensor is required available at: <https://www.aptna.com/products/soc/mt9m114/>

Run the following procedure is to ensure the ISP drivers were properly loaded.

1. Open a terminal in the Fedora* Graphical User Interface (GUI).
2. Enter **dmesg | grep atomisp | grep success** at the command prompt and you should get the following results:

```
[ ....] atomisp: atomisp: ISP firmware iaisp_2400_css.bin.big successfully loaded
[ ....] atomisp 0000:00:03.0: Subdev ov5640-1 successfully register
[ ....] atomisp 0000:00:03.0: Subdev mt9m114 successfully register
[ ....] atomisp 0000:00:03.0: Subdev ov5640-2 successfully register
[ ....] atomisp 0000:00:03.0: atomisp_open: successfully opened video ATOMISP ISP PREVIEW
output
[ ....] atomisp 0000:00:03.0: atomisp_open: successfully opened video ATOMISP ISP
VIEWFINDER output
[ ....] atomisp 0000:00:03.0: atomisp_open: successfully opened video ATOMISP ISP VIDEO
output
[ ....] atomisp 0000:00:03.0: atomisp_open: successfully opened video ATOMISP ISP
VIEWFINDER output
[ ....] atomisp 0000:00:03.0: atomisp_open: successfully opened video ATOMISP ISP CAPTURE
output
[ ....] atomisp 0000:00:03.0: atomisp_open: successfully opened video ATOMISP ISP MEMORY
input
[ ....] atomisp 0000:00:03.0: atomisp_open: successfully opened video ATOMISP ISP PREVIEW
output
[ ....] atomisp 0000:00:03.0: atomisp_open: successfully opened video ATOMISP ISP VIDEO
output
```




```
[ ....] atomisp 0000:00:03.0: atomisp_open: successfully opened video ATOMISP ISP MEMORY
input
[ ....] atomisp 0000:00:03.0: atomisp_open: successfully opened video ATOMISP ISP CAPTURE
output
[ ....] atomisp 0000:00:03.0: atomisp_open: successfully opened video ATOMISP ISP CAPTURE
output
[ ....] atomisp 0000:00:03.0: atomisp_open: successfully opened video ATOMISP ISP
VIEWFINDER output
[ ....] atomisp 0000:00:03.0: atomisp_open: successfully opened video ATOMISP ISP PREVIEW
output
[ ....] atomisp 0000:00:03.0: atomisp_open: successfully opened video ATOMISP ISP VIDEO
output
[ ....] atomisp 0000:00:03.0: atomisp_open: successfully opened video ATOMISP ISP MEMORY
input
```

Note: For enabling video streams with a resolution higher than 1920x1080, (if one can find a camera sensor that can have a higher resolution than that of the 1080p sensor) recompilation is required. Set the kernel configuration as follows:

- **CONFIG_CMA**

The CONFIG_CMA setting is only available in the 32-bit kernel build. For the 64-bit kernel build, follow these steps to update the source code:

1. Go to the [path/to/linux/src]/include/linux directory.
2. Open the mmzone.h file in a text editor.
3. Look for #define MAX_ORDER 11.
4. Change 11 to 12.
5. Save and close the mmzone.h file.

Continue to build the kernel and kernel modules.

- **Set CMA_SIZE_MBYTES to 64 MB**

§



9 Test Application

The Intel® Atom™ ISP test application is delivered in both binary executable and source code under the following license.

Intel® Atom™ ISP Test Application is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

Intel® Atom™ ISP Test Application is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

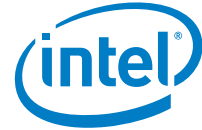
For detailed instructions on how to compile and run the test application, refer to the readme file that is provided with the ISP test application.

To run or compile the source, the following libraries need to be installed:

1. Tools
 - **gcc**
 - **gdb (optional)**
 - **make**
 -
2. Libraries:
 - **libdrm**
 - **libdrm_intel**
 - **wayland-client**
 - **wayland-egl**
 - **EGL**
 - **GLSv2**
 - **X11**
 - **V4L2**
 - **Glibc**

Notes:

- Ensure all libraries installed have 32-bit (i386/i686) and 64-bit (x86_64) variants to allow cross-compilation between them.
- Ensure the “-devel” or “-dev” packages are installed.
- To build the 64-bit application, you will need a 64-bit machine with a 64-bit Linux* OS installed as your development environment.



9.1 Building the Test Application

To build the application for Fedora*/GNOME* or KDE*, run the script within the application directory:

```
./do_make.sh mipi-x -DCOLOR_CONVERSION
```

This builds the application with the name

```
isp-mipi-test
```

The supported CFLAGS by do_make.sh are:

- -DCOLOR_CONVERSION to support color conversion. This enables the -C option.
- -DPIPE_FRAMES_TO_STDOUT to set the application to dump the frames from the GPU to files that can be piped to a capturing application like ffmpeg or gstreamer.

Note: The script defaults to compile as an i686 binary (32-bit).

To enable x86_64 binary (64-bit) generation, the environment variable ARCH needs to be defined first.

```
export ARCH=x86_64
```

Then, run the do_make.sh script as stated above. Notice that the -m64 is supplied to gcc during the build.

Please note that to get the x86_64 binary, all the libraries in x86_64 have to be installed.

9.2 Using the Test Application

For help on the available options in the test application, run the following:

```
/isp-mipi-test --help
```

The application then returns the following:

```
./isp-mipi-test_cc.x.i686 [options] -d, --device <device>
-b, --buffers <number_of_buffers>
-g, --use-dma (use DMA buffer sharing)
-c, --color-format-out <out color_format>
-C, --color-format-in <in color_format> -w, --width <width>
-h, --height <height> -W, --width-in <width>
```



```
-H, --height-in <height>
-p, --mipi-port <mipi-port>
-m, --mipi-main <mipi-main>
-v, --mipi-viewfinder <mipi-viewfinder>
-n, --max-frame-count <max_frame_count>
-i, --interlace (to enable interlace mode)
-q, --quiet (Turn off logging)
-f, --no-frames (Do not render frames)
-F, --inject-file </path/to/raw_file>
-M, --mode <ISP_mode>
--working-directory </path/to/working/directory>
--drm-device <DRM_device_name>
--dump-frame --dump-frame-every
--dump-raw --dump-raw-every
--atomisp-config-3a </path/to/atomisp_3a_config/file>
```

Supported Color Formats From ISP (-c): YV16 [default], NV12, YUYV / YUYV8, RGB565 / RGBP, RGB888 / RGB3

Supported Color Formats To ISP (-C): YV16 [default], RGB565 / RGBP, RGB888 / RGB3, SGRBG10 / BA10 Supported

Supported ISP Running Modes (-M): STILL [default], VIDEO, CONTINUOUS, PREVIEW, NONE

The test application default configurations are also listed.

To stream from a sensor, run the following:

```
./isp-mipi-test -c YV16 -p 0
```

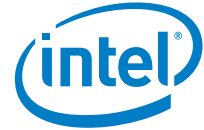
If **-DCOLOR_CONVERSION** is used during build, run the following:

```
./isp-mipi-test -c YV16 -C YV16 -p 0
```

During test execution, the application stores a copy of its activity log in a log file named **isp-mipi-test.[number].log**

The application generates another log file when the frames are streaming, and names it **frames.[number].fps**

Test Application



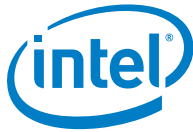
This file logs all the performance numbers of each frame. The following is the frame log format:

```
frame,capture_time (usec),render_time (usec),total_time (usec),fps
```

Each component of the log file represents the following:

- **frame** – Frame number captured during streaming.
- **capture_time** – Time taken from the sensor to the ISP in μ s.
- **render_time** - Time taken to render a frame in μ s.
- **total_time** – Sum of capture time and render time in μ s.
- **fps** – Running total of frames being processed while running the test.

The [number] increments on each run of the application. The log file and fps file share the same increment number.



9.3 Supported Color Formats for `-c` Flag

The Intel Atom ISP test application supports the following color formats for `-c` flag:

- YV16
 - YUV422 planar
 - Full Y, half-width for U and V
 - U and V are not packaged
- RGB565/RGBP
 - 16-bit RGB
 - 6-bit for G, 5-bit for R and B components, respectively
- RGB888/RGB3
 - 24-bit RGB
 - 8-bit for all components
- NV12
 - YUV422 semi-planar
 - Full Y, half-size for U and V
 - U and V are packed

The color formats are important because the test application associates it to the correct GL Shader fragment programs found in the **"shaders/"** directory.

If the listed color formats do not include the one you are looking for, you need to provide your own Shader fragment programs and put them into the **"shaders/fragment/"** directory. Then, proceed to update the following source files:

- **utilities.h**
- **shader.c**
- **isp-mipi-test-c**

9.4 Supported Color Formats for `-C` Flag

The Intel Atom ISP test application support the following color formats for `-C` flag:

- YV16
 - YUV422 planar
 - full Y, half-width for U and V
 - U and V are not packed
- YUYV
 - YUV422 packed.
- RGB565/RGBP
 - o 16-bit RGB
 - o 6-bit for G, 5-bit for R and B components, respectively
- BA10

Note: The RAW color format for Aptina* MT9M114 image sensor is only available in the ISP_VIDEO mode (Preview).

[Figure 2](#) shows the file structure for the Shader fragment programs. Ensure your fragment is put in the right directory.

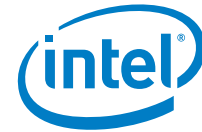


Figure 2. File structure of Shader fragment program

```

shaders
|---fragment
|   |---weston
|   |   |---nv12.c
|   |   |---rgb_passthru.c
|   |   |---uyvy.c
|   |   |---vyuy.c
|   |   |---yuyv.c
|   |   |---yv16.c
|   |   +---yvyu.c
|   +---x
|       |---nv12.c
|       |---rgb_passthru.c
|       |---uyvy.c
|       |---vyuy.c
|       |---yuyv.c
|       |---yv16.c
|       +---yvyu.c
+---vertex
    +---default.c
    
```

Table 4 shows the supported MIPI* ports for Bayley Bay or Bakersport CRBs.

Table 4. Supported MIPI ports for Bayley Bay or Bakersport CRBs

Port	Sensor	Kernel Module
0	OmniVision* Sensor	OV5640_1.ko
1	Aptina Sensor	MT9M114.ko
2	OmniVision Sensor	OV5640_2.ko

9.5 Main and Viewfinder Views

The Main and Viewfinder view options are required when streaming from more than one sensor. The Intel Atom ISP test application needs to know where to temporarily hold the video data without interfering with the data from other sensors.



Note: When the Main and Viewfinder views are provided, the test application defaults them to the /dev/video0 for any sensor and will cause deadlock and timeouts when streaming from multiple sensors.

This test application has been tested on the Bayley Bay CRB with the supported sensors.

To stream from multiple sensors, these commands are used:

```
./isp-mipi-test -c YV16 -C YV16 -p 0 -m /dev/video0 &> /dev/null &  
./isp-mipi-test -c YV16 -C YV16 -p 2 -m /dev/video5 &> /dev/null &
```

or

```
./isp-mipi-test -c YV16 -C YV16 -p 0 -m /dev/video5 &> /dev/null &  
./isp-mipi-test -c YV16 -C YV16 -p 2 -m /dev/video10 &> /dev/null &
```

or

```
./isp-mipi-test -c YV16 -C YV16 -p 0 -m /dev/video5 &> /dev/null &  
./isp-mipi-test -c YV16 -C YV16 -p 1 -m /dev/video0 &> /dev/null &
```

Note: Remove the -C option if the application is tested against ISP prior to 3.0.

[Table 5](#) shows the Main and Viewfinder sensor and port settings required for the Bayley Bay CRB.

Table 5. Main and Viewfinder Sensor Settings Required for Bayley Bay CRB

Sensor	Port	Main	Viewfinder
OmniVision OV5640_1 image sensor	0	/dev/video4	/dev/video6
OmniVision OV5640_2 image sensor	2	/dev/video8	/dev/video10
Aptina MT9M114 image sensor	1	/dev/video0	/dev/video1

9.6 Streaming With Viewfinder Turned On

By default, the test application does not activate the Viewfinder. Use the following command to stream with the Viewfinder turned on:



```
/isp-mipi-test -g -c nv12 -C ba10 -p 1 -w 1280 -h 720 -m /dev/video3 -v  
/dev/video1 -M video
```

9.7 Limitations

The following are the Viewfinder limitations when it is activated:

- BA10 (raw) is the only color format supported for Viewfinder.
- Only the Aptina MT9M114 image sensor can support the BA10 color format.
- Only /dev/video0 for the Main view and /dev/video1 for the Viewfinder view are supported.
- Only a single stream is supported.
- The application does not render the Viewfinder frames.
- The application shows only corrupted frames because the rendering engine is not updated.
- The application automatically reduces the resolution by 12 pixels in width and height.

§