i.MX 8M Plus ISP Using V4L2 Interface User Guide

Contents

Chapter 1
Overview

This document describes the ISP software Application Programming Interface (API) using Video For Linux 2. The ISP software V4L2 API controls the ISP hardware, sensor hardware, and its calibration data from the Linux standard API. The kernel V4L2 driver handles the API commands and requests from the V4L2 user application, communicates to the ISP software stack and delivers image buffers to the V4L2 user application.

Currently, there are no deprecated functions in this API.

1.1 Requirements/dependencies

- Linux environment is compatible with V4L2.

1.2 Supported features

ISP features which are listed in Table 1 are currently supported in the ISP V4L2 API.

Table 1. ISP features

Feature	Abbreviation
Auto Focus	AF
Auto Exposure	AE
Auto White Balance	AWB
Auto Video Stabilization	AVS
Black Level Subtraction	BLS
Chromatic Aberration Correction	CAC
Color Noise Reduction	CNR
Color Processing	CPROC
Demosaic	--
Defect Pixel Cluster Correction	DPCC
De-noising Pre-filter	DPF
High Dynamic Range	HDR
Image Effect	IE
Lens Shade Correction	LSC
Noise Reduce 2D	2DNR
Noise Reduce 3D	3DNR
Wide Dynamic Range	WDR

Sensor features: Additional functionality provided in future releases.

Chapter 2
V4L2 API components

The ISP software V4L2 API is written in ANSI C++ code and is defined in the *v4l2/hal-vivante-camera/ sub folder*. All commands are performed in the user space using an IOCTL interface which calls kernel space actions directly. The IOCTL control words are described in the IOCTL Interface and Commands.

The ISP software V4L2 API components are defined in the following sections:

- Buffer API
- Event API
- Feature control API

2.1 IOCTL interface and commands

V4L2 provides Input and Output Control (IOCTL) interfaces to communicate directly with device drivers. Table 2 lists key IOCTLs relevant to the ISP V4L2 software. Each IOCTL command corresponds to an operation function.

Table 2. Key video IOTCLs

IOCTL	Type	Description
VIDIOC_QUERYCAP	.vidioc_querycap	Query the capabilities of the driver, such as V4L2_CAP_STREAMING
VIDIOC_S_FMT	.vidioc_s_fmt_*	Set format information
VIDIOC_REQBUFS	.vidioc_reqbufs	Request buffers. Buffer types: DMA, MMAP, USER_PTR
VIDIOC_QBUF	.vidioc_qbuf	Enqueue buffer to kernel, then the driver fills this buffer
VIDIOC_QUERYBUF	.vidioc_querybuf	Get buffer information from the kernel and mmap
VIDIOC_DQBUF	.vidioc_dqbuf	De-queue the buffer from the kernel. User gets frame data
VIDIOC_STREAMON	.vidioc_streamon	Start stream
VIDIOC_STREAMOFF	.vidioc_streamoff	Close stream
VIDIOC_G_EXT_CTRLS	.vidioc_g_ext_ctrls	Get feature control commands
VIDIOC_S_EXT_CTRLS	.vidioc_s_ext_ctrls	Set feature control commands

2.2 IOCTL call flow

IOTCL call flow is described in Figure 1 and the ISP reference code is based on this implementation.
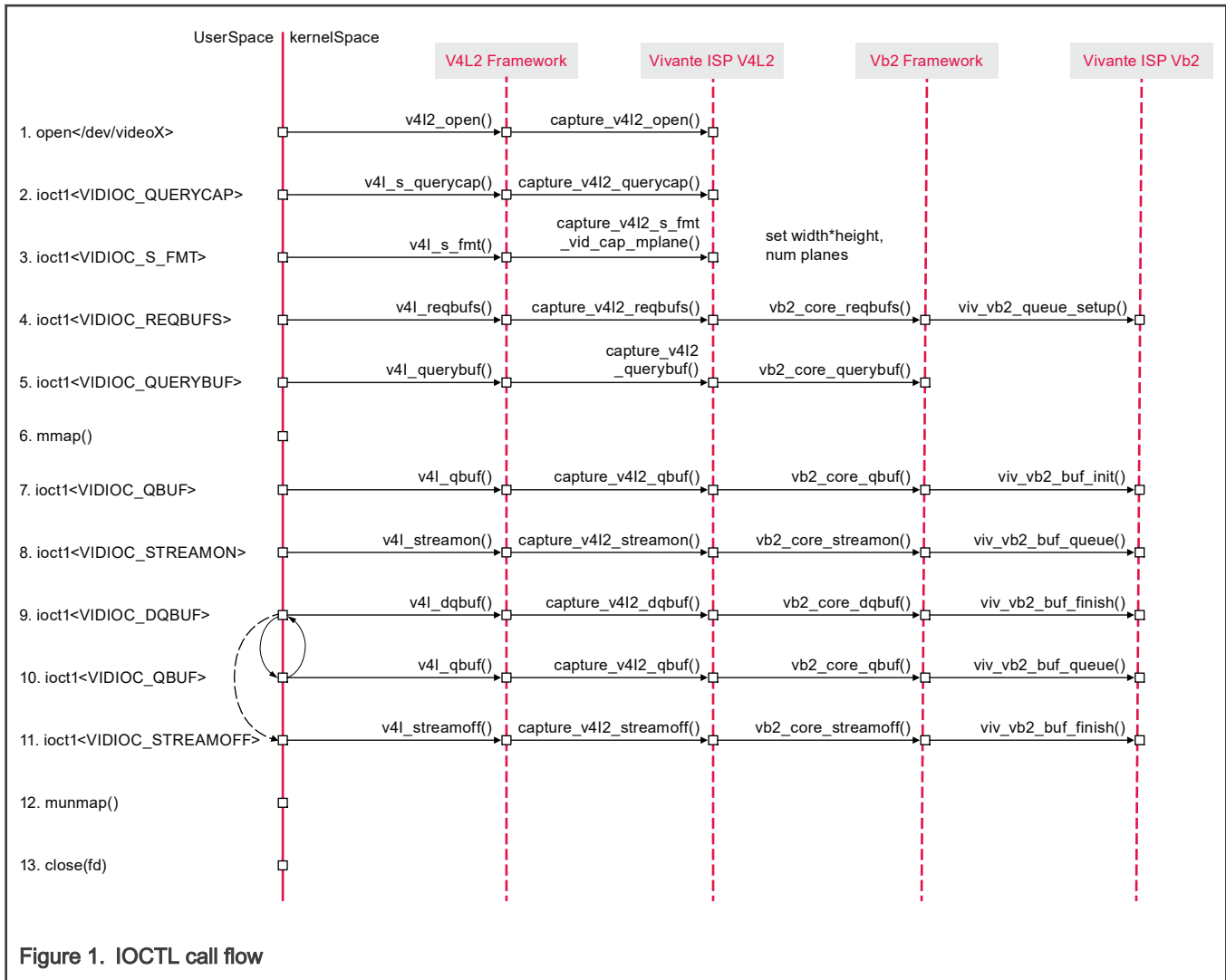
This flow will be expanded in the future.

Figure 1. IOCTL call flow

2.3 Buffer API

A buffer contains data exchanged by the application and driver using memory mapping I/O. Only pointers to buffers are exchanged; the data itself is not copied. Memory mapping is primarily intended to map buffers in device memory into the application's address space.

The V4L2 driver supports the following buffer IOCTLs:

- VIDIOC_REQBUFS
- VIDIOC_QUERYBUF
- VIDIOC_QBUF
- VIDIOC_DQBUF
- VIDIOC_STREAMON
- VIDIOC_STREAMOFF

In addition, the following functions are supported.

- mmap()
- munmap()

- select()

- poll()

2.3.1 Buffer IOCTL control words

- VIDIOC_REQBUFS

    Link: http://www.kernel.org/doc/html/v5.4/media/uapi/v4l/vidioc-reqbufs.html

- VIDIOC_QUERYBUF

    Link: http://www.kernel.org/doc/html/v5.4/media/uapi/v4l/vidioc-querybuf.html

- VIDIOC_QBUF

    Link: http://www.kernel.org/doc/html/v5.4/media/uapi/v4l/vidioc-qbuf.html

- VIDIOC_DQBUF

    Link: http://www.kernel.org/doc/html/v5.4/media/uapi/v4l/vidioc-qbuf.html

- VIDIOC_STREAMON

    Link: http://www.kernel.org/doc/html/v5.4/media/uapi/v4l/vidioc-streamon.html

- VIDIOC_STREAMOFF

    Link: http://www.kernel.org/doc/html/v5.4/media/uapi/v4l/vidioc-streamon.html

2.3.2 Buffer functions

- mmap

    Link: http://www.kernel.org/doc/html/v5.4/media/uapi/v4l/func-mmap.html

- munmap

    Link: http://www.kernel.org/doc/html/v5.4/media/uapi/v4l/func-munmap.html

- poll

    Link: http://www.kernel.org/doc/html/v5.4/media/uapi/v4l/func-poll.html

2.4 Event API

The V4L2 event interface provides a means for a user to get notified immediately on certain conditions taking place on a device.

To receive events, first the user must subscribe to an event using the VIDIOC_SUBSCRIBE_EVENT and the VIDIOC_UNSUBSCRIBE_EVENT IOCTLs. Once an event is subscribed, the events of subscribed types are de-queueable using the VIDIOC_DQEVENT IOCTL. Events may be unsubscribed using the VIDIOC_UNSUBSCRIBE_EVENT IOCTL. The information on de-queueable events is obtained by using poll() system calls on video devices. The V4L2 events use POLLPRI events on poll system calls.

The V4L2 driver supports the following event IOCTLs:

- VIDIOC_SUBSCRIBE_EVENT

- VIDIOC_UNSUBSCRIBE_EVENT

- VIDIOC_DQEVENT

In addition, the following function is supported.

- poll()

2.4.1 Event IOCTL control words

- VIDIOC_SUBSCRIBE_EVENT

Link: http://www.kernel.org/doc/html/v5.4/media/uapi/v4l/vidioc-subscribe-event.html

- VIDIOC_UNSUBSCRIBE_EVENT

  Link: http://www.kernel.org/doc/html/v5.4/media/uapi/v4l/vidioc-subscribe-event.html

- VIDIOC_DQEVENT

  Link: http://www.kernel.org/doc/html/v5.4/media/uapi/v4l/vidioc-dqevent.html

2.4.2 Event functions

- poll

  Link: http://www.kernel.org/doc/html/v5.4/media/uapi/v4l/func-poll.html

2.4.3 Private event

The private event is an extension based on V4L2_EVENT_PRIVATE_START and defines ID of the private event source, defines event data struct knl_v4l2_event_data based on struct v4l2_event.u.data[64].

Private event type:

- KNL_VIVCAM_V4L2_EVENT_TYPE

ID:

- KNL_VIVCAM_NOTIFY

Struct definition:

- Struct knl_v4l2_event_data, 64 bytes.

Table 3. Private event

Structure member	Type	Description
command	unsigned int	Extension based on V4L2_CID_PRIVATE_BASE
status	unsigned int	
session_id	unsigned int	
stream_id	unsigned int	
nop1	unsigned int	Reserved for future extensions
…	…	…
nop12	unsigned int	

2.5 Feature control API

The feature control API, uses JavaScript Object Notation (JSON) objects in user application threads and shares the objects directly with the daemon using share memory methods.

The ISP daemon sets ISPCore feature control words directly with the JSON parameters. In the user space and kernel space transfer, the Json::Value object is translated to a char string and transferred between the user and kernel space as shown in Figure 2.

Figure 2. Feature control block diagram
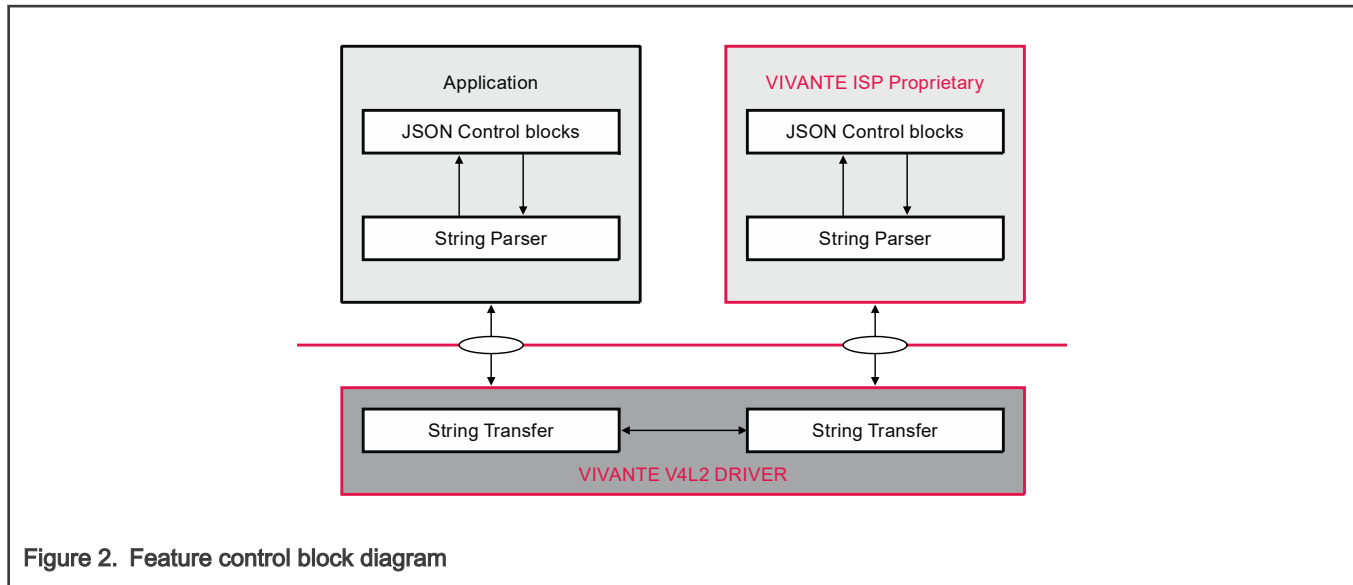
2.5.1 String parser

The JSON format used for the APIs and the string transfer can be handled using open source code.

For example:

1. Json::Value to char string:

```
String Json::Value::toStyledString(Json::Value)
```

2. char string to Json::Value:

```
Json::CharReaderBuilder::parse(const char* beginDoc,
                               const char* endDoc,
                               Value& root, bool collectComments = true);
```

2.5.2 String transfer

All feature-related JSON-String entities are transferred using the following IOCTLs:

- VIDIOC_G_EXT_CTRLS

  Link: http://www.kernel.org/doc/html/v5.4/media/uapi/v4l/vidioc-g-ext-ctrls.html

- VIDIOC_S_EXT_CTRLS

  Link: http://www.kernel.org/doc/html/v5.4/media/uapi/v4l/vidioc-g-ext-ctrls.html

For a detailed example, refer to the code *appshell/vvext/vvext.cpp*.

The char string memory block exchange using the v4l2_ext_control struct, as shown in Table 4.

Table 4. v4l2_ext_control Structure

v4l2_ext_control structure member	Type	Description
id	__u32	V4L2 ISP SW feature control words
size	__u32	String length

Table continues on the next page...

Table 4. v4l2_ext_control Structure (continued)

v4l2_ext_control structure member	Type	Description
reserved2[1]	__u32	
value	union of __s32	
value64	union of __s64	
string	union of char *	String transfer pointer
p_u8	union of __u8 *	
p_u16	union of __u16 *	
p_u32	union of __u32 *	
ptr	union of void*	

2.5.3 Feature control words

Interface header file: *mediacontrol/include_api/ioctl_cmds.h.*

- **IF_AE_G_CFG**

    This macro definition is identical to the string "ae.g.cfg".

    Description: Gets the configuration values for the Auto Exposure control.

    Parameters:

    — Json::Value &jQuery

    — Json::Value &jResponse

Table 5. Control words for IF_AE_G_CFG

Control word	Description
mode	Configuration mode
damp.over	Damping upper limit
damp.under	Damping lower limit
set.point	Set point
clm.tolerance	Calculation accuracy

- **IF_AE_S_CFG**

    This macro definition is identical to the string "ae.s.cfg".

    Description: Sets the configuration values for the Auto Exposure control.

    Parameters:

    — Json::Value &jQuery

    — Json::Value &jResponse

Table 6. Control words for IF_AE_S_CFG

Control word	Description
mode	Configuration mode
damp.over	Damping upper limit
damp.under	Damping lower limit
set.point	Set point
clm.tolerance	Calculation accuracy

- **IF_EC_G_CFG**

    This macro definition is identical to the string "ec.g.cfg".

    Description: Gets the ECM (Exposure Control Module) values for the Auto Exposure control.

    Parameters:

    — Json::Value &jQuery

    — Json::Value &jResponse

Table 7. Control words for IF_EC_G_CFG

Control word	Description
flicker.period	The flag of Auto Exposure flicker period
Afps	Auto FPS control value

- **IF_EC_S_CFG**

    This macro definition is identical to the string "ec.s.cfg".

    Description: Sets the ECM (Exposure Control Module) values for the Auto Exposure control.

    Parameters:

    — Json::Value &jQuery

    — Json::Value &jResponse

Table 8. Control words for IF_EC_S_CFG

Control word	Description
flicker.period	The flag of Auto Exposure flicker period
afps	Auto FPS control value

- **IF_AE_G_EN**

    This macro definition is identical to the string "ae.g.en".

    Description: Gets the enabled/disabled state of the Auto Exposure control.

    Parameters:

    — Json::Value &jQuery

    — Json::Value &jResponse

Table 9. Control words for IF_AE_G_EN

Control word	Description
enable	The state of Auto Exposure

- **IF_AE_S_EN**

    This macro definition is identical to the string "ae.s.en".

    Description: Sets the enabled/disabled state of the Auto Exposure control.

    Parameters:

    — Json::Value &jQuery

    — Json::Value &jResponse

Table 10. Control words for IF_AE_S_EN

Control word	Description
enable	Auto Exposure is enabled

- **IF_AE_RESET**

    This macro definition is identical to the string "ae.reset".

    Description: Reset the Auto Exposure control.

    Parameters:

    — Json::Value &jQuery

    — Json::Value &jResponse

Table 11. Control words for IF_AE_RESET

Control word	Description
N/A	

- **IF_AF_G_CFG**

    This macro definition is identical to the string "af.g.cfg".

    Description: Gets the configuration of the Auto Focus control.

    Parameters:

    — Json::Value &jQuery

    — Json::Value &jResponse

Table 12. Control words for IF_AF_G_CFG

Control word	Description
algorithm	Algorithm type
oneshot	Trigger mode is one shot

- **IF_AF_S_CFG**

    This macro definition is identical to the string "af.s.cfg".

Description: Sets the configuration of the Auto Focus control.

Parameters:

— Json::Value &jQuery

— Json::Value &jResponse

Table 13. Control words for IF_AF_S_CFG

Control word	Description
algorithm	Algorithm type
oneshot	Trigger mode is one shot

- **IF_AF_G_EN**

  This macro definition is identical to the string "af.g.en".

  Description: Gets the enabled/disabled state of the Auto Focus control.

  Parameters:

  — Json::Value &jQuery

  — Json::Value &jResponse

Table 14. Control words for IF_AF_G_EN

Control word	Description
enable	The state of the Auto Focus

- **IF_AF_S_EN**

  This macro definition is identical to the string "af.s.en".

  Description: Sets the enabled/disabled state of the Auto Focus control.

  Parameters:

  — Json::Value &jQuery

  — Json::Value &jResponse

Table 15. Control words for IF_AF_S_EN

Control word	Description
enable	Auto Focus is enabled

- **IF_AWB_G_CFG**

  This macro definition is identical to the string "awb.g.cfg".

  Description: Gets the configuration of the Auto White Balance control.

  Parameters:

  — Json::Value &jQuery

  — Json::Value &jResponse

Table 16. Control Words for IF_AWB_G_CFG

Control word	Description
mode	AWB mode
index	The index of calibration data in the database
damping	Have damped data

- **IF_AWB_S_CFG**

  This macro definition is identical to the string "awb.s.cfg".

  Description: Sets the mode and index of the Auto White Balance control.

  Parameters:

  — Json::Value &jQuery

  — Json::Value &jResponse

Table 17. Control Words for IF_AWB_S_CFG

Control word	Description
mode	AWB mode
index	The index of calibration data in the database
damping	Damping data

- **IF_AWB_G_EN**

  This macro definition is identical to the string "awb.g.en".

  Description: Gets the enabled/disabled state of the Auto White Balance control.

  Parameters:

  — Json::Value &jQuery

  — Json::Value &jResponse

Table 18. Control words for IF_AWB_G_EN

Control word	Description
enable	The state of the AWB control

- **IF_AWB_S_EN**

  This macro definition is identical to the string "awb.s.en".

  Description: Sets the enabled/disabled state of the Auto White Balance control.

  Parameters:

  — Json::Value &jQuery

  — Json::Value &jResponse

Table 19. Control words for IF_AWB_S_EN

Control word	Description
enable	Auto White Balance is enabled

- **IF_AWB_RESET**

    This macro definition is identical to the string "awb.reset".

    Description: Resets the Auto White Balance control.

    Parameters:

    — Json::Value &jQuery

    — Json::Value &jResponse

Table 20. Control words for IF_AWB_RESET

Control word	Description
N/A	-

- **IF_AVS_G_CFG**

    This macro definition is identical to the string "avs.g.cfg".

    Description: Gets the configuration values for the Auto Video Stabilization control.

    Parameters:

    — Json::Value &jQuery

    — Json::Value &jResponse

Table 21. Control words for IF_AVS_G_CFG

Control word	Description
use.params	AVS use params
acceleration	AVS has acceleration
base.gain	AVS's base gain
fall.off	AVS has fall off
num.itp.points	The number of ITP points
theta	Theta
x	The size of width
y	The size of height

- **IF_AVS_S_CFG**

    This macro definition is identical to the string "avs.s.cfg".

    Description: Sets the configuration values for the Auto Video Stabilization control.

    Parameters:

    — Json::Value &jQuery

    — Json::Value &jResponse

Table 22. Control words IF_AVS_S_CFG

Control word	Description
use.params	AVS use params
acceleration	AVS has acceleration
base.gain	AVS's base gain
fall.off	AVS has fall off
num.itp.points	The number of ITP points
theta	Theta

- **IF_AVS_G_EN**

  This macro definition is identical to the string "avs.g.en".

  Description: Gets the enabled/disabled state of the Auto Video Stabilization control.

  Parameters:

  — Json::Value &jQuery

  — Json::Value &jResponse

Table 23. Control words for IF_AVS_G_EN

Control word	Description
enable	The state of the AVS

- **IF_AVS_S_EN**

  This macro definition is identical to the string "avs.s.en".

  Description: Sets the enabled/disabled state of the Auto Video Stabilization control.

  Parameters:

  — Json::Value &jQuery

  — Json::Value &jResponse

Table 24. Control words for IF_AVS_S_EN

Control word	Description
enable	AVS is enabled

- **IF_BLS_G_CFG**

  This macro definition is identical to the string "bls.g.cfg".

  Description: Gets the configuration values for the Black Level Subtraction control.

  Parameters:

  — Json::Value &jQuery

  — Json::Value &jResponse

Table 25. Control words for IF_BLS_G_CFG

Control word	Description
red	The red data information
green.r	The Gr data information
green.b	The Gb data information
blue	The blue data information

- **IF_BLS_S_CFG**

    This macro definition is identical to the string "bls.s.cfg".

    Description: Sets the configuration values for the Black Level Subtraction control.

    Parameters:

    — Json::Value &jQuery

    — Json::Value &jResponse

Table 26. Control words for IF_BLS_S_CFG

Control word	Description
red	The red data information
green.r	The Gr data information
green.b	The Gb data information
blue	The blue data information

- **IF_CAC_G_EN**

    This macro definition is identical to the string "cac.g.en".

    Description: Gets the enabled/disabled state of the Chromatic Aberration Correction control.

    Parameters:

    — Json::Value &jQuery

    — Json::Value &jResponse

Table 27. Control words for IF_CAC_G_EN

Control word	Description
enable	The state of the Chromatic Aberration Correction

- **IF_CAC_S_EN**

    This macro definition is identical to the string "cac.s.en".

    Description: Sets the enabled/disabled state of the Chromatic Aberration Correction control.

    Parameters:

    — Json::Value &jQuery

    — Json::Value &jResponse

Table 28. Control words for IF_CAC_S_EN

Control word	Description
enable	Chromatic Aberration Correction is enabled

- **IF_CNR_G_CFG**

  This macro definition is identical to the string "cnr.g.cfg".

  Description: Gets the configuration values for the Chroma Noise Reduction control.

  Parameters:

  — Json::Value &jQuery

  — Json::Value &jResponse

Table 29. Control words for IF_CNR_G_CFG

Control word	Description
tc1	tc1
tc2	tc2

- **IF_CNR_S_CFG**

  This macro definition is identical to the string "cnr.s.cfg".

  Description: Sets the configuration values for the Chroma Noise Reduction control.

  Parameters:

  — Json::Value &jQuery

  — Json::Value &jResponse

Table 30. Control words for IF_CNR_S_CFG

Control word	Description
tc1	tc1
tc2	tc2

- **IF_CPROC_G_CFG**

  This macro definition is identical to the string "cproc.g.cfg".

  Description: Gets the configuration values for the Color Processing control.

  Parameters:

  — Json::Value &jQuery

  — Json::Value &jResponse

Table 31. Control words for IF_CPROC_G_CFG

Control word	Description
brightness	Brightness value
chroma.out	CPROC chrominance pixel clipping range at output

Table continues on the next page...

Table 31. Control words for IF_CPROC_G_CFG (continued)

Control word	Description
contrast	Contrast value
hue	Hue value
luma.in	CPROC luminance input range (offset processing)
luma.out	CPROC luminance output clipping range
saturation	Saturation value

- **IF_CPROC_S_CFG**

    This macro definition is identical to the string "cproc.s.cfg".

    Description: Sets the configuration values for the Color Processing control.

    Parameters:

    — Json::Value &jQuery

    — Json::Value &jResponse

Table 32. Control words for IF_CPROC_S_CFG

Control word	Description
brightness	Brightness value
chroma.out	CPROC chrominance pixel clipping range at output
contrast	Contrast value
hue	Hue value
luma.in	CPROC luminance input range(offset processing)
luma.out	CPROC luminance output clipping range
saturation	Saturation value

- **IF_CPROC_G_EN**

    This macro definition is identical to the string "cproc.g.en".

    Description: Gets the enabled/disabled state of the Color Processing control.

    Parameters:

    — Json::Value &jQuery

    — Json::Value &jResponse

Table 33. Control words for IF_CPROC_G_EN

Control word	Description
enable	The state of the CPROC

- **IF_CPROC_S_EN**

    This macro definition is identical to the string "cproc.s.en".

    Description: Sets the enabled/disabled state of the Color Processing control.

Parameters:

— Json::Value &jQuery

— Json::Value &jResponse

Table 34. Control words for IF_CPROC_S_EN

Control word	Description
enable	CPROC is enabled

- **IF_DEMOSAIC_G_CFG**

  This macro definition is identical to the string "dmsc.g.cfg".

  Description: Gets the configuration values for the Demosaic control.

  Parameters:

  — Json::Value &jQuery

  — Json::Value &jResponse

Table 35. Control words for IF_DEMOSAIC_G_CFG

Control word	Description
mode	Demosaic mode
threshold	Demosaic threshold

- **IF_DEMOSAIC_S_CFG**

  This macro definition is identical to the string "dmsc.s.cfg".

  Description: Sets the configuration values for the Demosaic control.

  Parameters:

  — Json::Value &jQuery

  — Json::Value &jResponse

Table 36. Control words for IF_DEMOSAIC_S_CFG

Control word	Description
mode	Demosaic mode
threshold	Demosaic threshold

- **IF_DEMOSAIC_G_EN**

  This macro definition is identical to the string "demosaic.g.en".

  Description: Gets the enabled/disabled state of the Demosaic control.

  Parameters:

  — Json::Value &jQuery

  — Json::Value &jResponse

Table 37. Control words for IF_DEMOSAIC_G_EN

Control word	Description
enable	The state of the Demosaic control

- **IF_DEMOSAIC_S_EN**

  This macro definition is identical to the string "demosaic.s.en".

  Description: Sets the enabled/disabled state of the Demosaic control.

  Parameters:

  — Json::Value &jQuery

  — Json::Value &jResponse

Table 38. Control words for IF_DEMOSAIC_S_EN

Control word	Description
enable	Demosaic is enabled

- **IF_DPCC_G_EN**

  This macro definition is identical to the string "dpcc.g.en".

  Description: Gets the enabled/disabled state of the Defect Pixel Cluster Correction control.

  Parameters:

  — Json::Value &jQuery

  — Json::Value &jResponse

Table 39. Control words for IF_DPCC_G_EN

Control word	Description
enable	The state of the Defect Pixel Cluster Correction

- **IF_DPCC_S_EN**

  This macro definition is identical to the string "dpcc.s.en".

  Description: Sets the enabled/disabled state of the Defect Pixel Cluster Correction control.

  Parameters:

  — Json::Value &jQuery

  — Json::Value &jResponse

Table 40. Control words for IF_DPCC_S_EN

Control word	Description
enable	Defect Pixel Cluster Correction is enabled

- **IF_DPF_G_CFG**

  This macro definition is identical to the string "dpf.g.cfg".

  Description: Gets the configuration values for the De-noising Pre-Filter control.

Parameters:

— Json::Value &jQuery

— Json::Value &jResponse

Table 41. Control words for IF_DPF_G_CFG

Control word	Description
gradient	Gradient value for dynamic strength calculation
offset	Offset value for dynamic strength calculation
min	Upper bound for dynamic strength calculation
div	Division factor for dynamic strength calculation
sigma.green	Sigma value for green pixel
sigma.red.blue	Sigma value for red/blue pixel

- **IF_DPF_S_CFG**

    This macro definition is identical to the string "dpf.s.cfg".

    Description: Sets the configuration values for the De-noising Pre-Filter control.

    Parameters:

    — Json::Value &jQuery

    — Json::Value &jResponse

Table 42. Control words for IF_DPF_S_CFG

Control word	Description
gradient	Gradient value for dynamic strength calculation
offset	Offset value for dynamic strength calculation
min	Upper bound for dynamic strength calculation
div	Division factor for dynamic strength calculation
sigma.green	Sigma value for green pixel
sigma.red.blue	Sigma value for red/blue pixel

- **IF_DPF_G_EN**

    This macro definition is identical to the string "dpf.g.en".

    Description: Gets the enabled/disabled state of the De-noising Pre-Filter control.

    Parameters:

    — Json::Value &jQuery

    — Json::Value &jResponse

Table 43. Control words for IF_DPF_G_EN

Control word	Description
enable	The state of the De-noising Pre-Filter

- **IF_DPF_S_EN**

  This macro definition is identical to the string "dpf.s.en".

  Description: Sets the enabled/disabled state of the De-noising Pre-Filter control.

  Parameters:

  — Json::Value &jQuery

  — Json::Value &jResponse

Table 44. Control words for IF_DPF_S_EN

Control word	Description
enable	De-noising Pre-Filter is enabled

- **IF_EC_G_CFG**

  This macro definition is identical to the string "ec.g.cfg".

  Description: Gets the configuration values for the Exposure Control.

  Parameters:

  — Json::Value &jQuery

  — Json::Value &jResponse

Table 45. Control words for IF_EC_G_CFG

Control word	Description
gain	Exposure gain
time	Exposure time

- **IF_EC_S_CFG**

  This macro definition is identical to the string "ec.s.cfg".

  Description: Sets the configuration values for the Exposure Control.

  Parameters:

  — Json::Value &jQuery

  — Json::Value &jResponse

Table 46. Control words for IF_EC_S_CFG

Control word	Description
gain	Exposure gain
time	Exposure time

- **IF_EE_G_CFG**

  This macro definition is identical to the string "ee.g.cfg".

  Description: Gets the configuration values for the Edge Enhancement control.

  Parameters:

  — Json::Value &jQuery

— Json::Value &jResponse

Table 47. Control words for IF_EE_G_CFG

Control word	Description
strength	Strength
sharpen	Sharpen
depurple	Depurple

- **IF_EE_S_CFG**

    This macro definition is identical to the string "ee.s.cfg".

    Description: Sets the configuration values for the Edge Enhancement control.

    Parameters:

    — Json::Value &jQuery

    — Json::Value &jResponse

Table 48. Control words for IF_EE_S_CFG

Control word	Description
strength	Strength
sharpen	Sharpen
depurple	Depurple

- **IF_EE_G_EN**

    This macro definition is identical to the string "ee.g.en".

    Description: Gets the enabled/disabled state of the Edge Enhancement control.

    Parameters:

    — Json::Value &jQuery

    — Json::Value &jResponse

Table 49. Control words for IF_EE_G_EN

Control word	Description
enable	The state of the Edge Enhancement control

- **IF_EE_S_EN**

    This macro definition is identical to the string "ee.s.en".

    Description: Sets the enabled/disabled state of the Edge Enhancement control.

    Parameters:

    — Json::Value &jQuery

    — Json::Value &jResponse

Table 50. Control words for IF_EE_S_EN

Control word	Description
enable	Edge Enhancement is enabled

- **IF_EE_RESET**

  This macro definition is identical to the string "ee.reset".

  Description: Resets the Edge Enhancement control.

  Parameters:

  — Json::Value &jQuery

  — Json::Value &jResponse

Table 51. Control words for IF_EE_RESET

Control word	Description
N/A	-

- **IF_GC_G_CURVE**

  This macro definition is identical to the string "gc.g.curve".

  Description: Gets the configuration values for the Gamma control.

  Parameters:

  — Json::Value &jQuery

  — Json::Value &jResponse

Table 52. Control words for IF_GC_G_CURVE

Control word	Description
curve	Gamma curve

- **IF_GC_S_CURVE**

  This macro definition is identical to the string "gc.s.curve".

  Description: Sets the configuration values for the Gamma Control.

  Parameters:

  — Json::Value &jQuery

  — Json::Value &jResponse

Table 53. Control words for IF_GC_S_CURVE

Control word	Description
curve	Gamma curve

- **IF_GC_G_EN**

  This macro definition is identical to the string "gc.g.en".

  Description: Gets the enabled/disabled state of the Gamma Control.

Parameters:

— Json::Value &jQuery

— Json::Value &jResponse

Table 54. Control words for IF_GC_G_EN

Control word	Description
enable	The state of the Gamma Control

- **IF_GC_S_EN**

  This macro definition is identical to the string "gc.s.en".

  Description: Sets the enabled/disabled state of the Gamma Control.

  Parameters:

  — Json::Value &jQuery

  — Json::Value &jResponse

Table 55. Control words for IF_GC_S_EN

Control word	Description
enable	Gamma Control is enabled

- **IF_HDR_G_CFG**

  This macro definition is identical to the string "hdr.g.cfg".

  Description: Gets the configuration of the High Dynamic Range control.

  Parameters:

  — Json::Value &jQuery

  — Json::Value &jResponse

Table 56. Control words for IF_HDR_G_CFG

Control word	Description
extension.bit	Extension bit
range.start.value	Range start value
very.short.weight	Very short weight

- **IF_HDR_S_CFG**

  This macro definition is identical to the string "hdr.s.cfg".

  Description: Sets the configuration of the High Dynamic Range control.

  Parameters:

  — Json::Value &jQuery

  — Json::Value &jResponse

Table 57. Control words for IF_HDR_S_CFG

Control word	Description
extension.bit	Extension bit
range.start.value	Range start value
very.short.weight	Very short weight

- **IF_HDR_G_EN**

    This macro definition is identical to the string "hdr.g.en".

    Description: Gets the enabled/disabled state of the High Dynamic Range control.

    Parameters:

    — Json::Value &jQuery

    — Json::Value &jResponse

Table 58. Control words for IF_HDR_G_EN

Control word	Description
enable	The state of High Dynamic Range

- **IF_HDR_S_EN**

    This macro definition is identical to the string "hdr.s.en".

    Description: Sets the enabled/disabled state of the High Dynamic Range control.

    Parameters:

    — Json::Value &jQuery

    — Json::Value &jResponse

Table 59. Control words for IF_HDR_S_EN

Control word	Description
enable	High Dynamic Range is enabled

- **IF_HDR_RESET**

    This macro definition is identical to the string "hdr.reset".

    Description: Resets the High Dynamic Range control.

    Parameters:

    — Json::Value &jQuery

    — Json::Value &jResponse

Table 60. Control words for IF_HDR_RESET

Control word	Description
N/A	-

- **IF_IE_G_CFG**

This macro definition is identical to the string "ie.g.cfg".

Description: Gets the configuration values for the Image Effects control.

Parameters:

— Json::Value &jQuery

— Json::Value &jResponse

Table 61. Control words for IF_IE_G_CFG

Control word	Description
Mode	The image can use seven available effect modes
Range	Image Effects configuration range
Config	Image Effects configuration
Tint.cb	Sepia Tint Cb of sepia mode
Tint.cr	Sepia Tint Cr of sepia mode
selection	Color selection of color mode
threshold	Color threshold of color mode
emboss:coeff	Coefficient of emboss mode
sketch:coeff	Coefficient of sketch mode
sharpen:factor	Factor of sharpen mode
sharpen:threshold	Threshold of sharpen mode

- **IF_IE_S_CFG**

  This macro definition is identical to the string "ie.s.cfg".

  Description: Sets the configuration values for the Image Effects control.

  Parameters:

  — Json::Value &jQuery

  — Json::Value &jResponse

Table 62. Control words for IF_IE_S_CFG

Control word	Description
Mode	The image can use seven available effect modes
Range	Image Effects configuration range
Config	Image Effects configuration
Tint.cb	Sepia Tint Cb of sepia mode
Tint.cr	Sepia Tint Cr of sepia mode
selection	Color selection of color mode
threshold	Color threshold of color mode
emboss:coeff	Coefficient of emboss mode

Table continues on the next page...

Table 62. Control words for IF_IE_S_CFG (continued)

Control word	Description
sketch:coeff	Coefficient of sketch mode
sharpen:factor	Factor of sharpen mode
sharpen:threshold	Threshold of sharpen mode
sharpen:coeff	Coefficient of sharpen mode

- **IF_IE_G_EN**

  This macro definition is identical to the string "ie.g.en".

  Description: Gets the enabled/disabled state of the Image Effects control.

  Parameters:

  — Json::Value &jQuery

  — Json::Value &jResponse

Table 63. Control words for IF_IE_G_EN

Control word	Description
Enable	The state of the Image Effects control

- **IF_IE_S_EN**

  This macro definition is identical to the string "ie.s.en".

  Description: Sets the enabled/disabled state of the Image Effects control.

  Parameters:

  — Json::Value &jQuery

  — Json::Value &jResponse

Table 64. Control words for IF_IE_S_EN

Control word	Description
Enable	Image Effects is enabled

- **IF_LSC_G_EN**

  This macro definition is identical to the string "lsc.g.en".

  Description: Gets the enabled/disabled state of the Lens Shade Correction control.

  Parameters:

  — Json::Value &jQuery

  — Json::Value &jResponse

Table 65. Control words for IF_LSC_G_EN

Control word	Description
Enable	The state of Lens Shade Correction

- **IF_LSC_S_EN**

  This macro definition is identical to the string "lsc.s.en".

  Description: Sets the enabled/disabled state of the Lens Shade Correction control.

  Parameters:

  — Json::Value &jQuery

  — Json::Value &jResponse

Table 66. Control words for IF_LSC_S_EN

Control word	Description
Enable	Lens Shade Correction is enabled

- **IF_2DNR_G_CFG**

  This macro definition is identical to the string "2dnr.g.cfg".

  Description: Gets the configuration values for the 2DNR control.

  Parameters:

  — Json::Value &jQuery

  — Json::Value &jResponse

Table 67. Control words for IF_2DNR_G_CFG

Control word	Description
Generation	NR2D generation
Strength	Configuration strength
sigma	Sigma strength

- **IF_2DNR_S_CFG**

  This macro definition is identical to the string "2dnr.s.cfg".

  Description: Sets the configuration values for the 2DNR control.

  Parameters:

  — Json::Value &jQuery

  — Json::Value &jResponse

Table 68. Control words for IF_2DNR_S_CFG

Control word	Description
Generation	NR2D generation
Strength	Configuration strength
sigma	Sigma strength

- **IF_2DNR_G_EN**

  This macro definition is identical to the string "2dnr.g.en".

  Description: Gets the enabled/disabled state of the 2DNR control.

Parameters:

— Json::Value &jQuery

— Json::Value &jResponse

Table 69. Control words for IF_2DNR_G_EN

Control word	Description
Enable	The state of NR2D

- **IF_2DNR_S_EN**

  This macro definition is identical to the string "2dnr.s.en".

  Description: Sets the enabled/disabled state of the 2DNR control.

  Parameters:

  — Json::Value &jQuery

  — Json::Value &jResponse

Table 70. Control words for IF_2DNR_S_EN

Control word	Description
Enable	NR2D is enabled

- **IF_2DNR_RESET**

  This macro definition is identical to the string "2dnr.reset".

  Description: Resets the 2DNR control.

  Parameters:

  — Json::Value &jQuery

  — Json::Value &jResponse

Table 71. Control words for IF_2DNR_RESET

Control word	Description
Generation	NR2D generation

- **IF_3DNR_G_CFG**

  This macro definition is identical to the string "3dnr.g.cfg".

  Description: Gets the configuration values for the 3DNR control.

  Parameters:

  — Json::Value &jQuery

  — Json::Value &jResponse

Table 72. Control words for IF_3DNR_G_CFG

Control word	Description
Generation	NR3D generation

Table continues on the next page...

Table 72. Control words for IF_3DNR_G_CFG (continued)

Control word	Description
Strength	NR3D strength
spatial.denoise	Spatial denoise
temporal.denoise	Temporal denoise

- **IF_3DNR_S_CFG**

  This macro definition is identical to the string "3dnr.s.cfg".

  Description: Sets the configuration values for the 3DNR control.

  Parameters:

  — Json::Value &jQuery

  — Json::Value &jResponse

Table 73. Control words for IF_3DNR_S_CFG

Control word	Description
Generation	NR3D generation
Strength	NR3D strength
spatial.denoise	Spatial denoise
temporal.denoise	Temporal denoise

- **IF_3DNR_G_EN**

  This macro definition is identical to the string "3dnr.g.en".

  Description: Gets the enabled/disabled state of the 3DNR control.

  Parameters:

  — Json::Value &jQuery

  — Json::Value &jResponse

Table 74. Control words for IF_3DNR_G_EN

Control word	Description
enable	The state of NR3D

- **IF_3DNR_S_EN**

  This macro definition is identical to the string "3dnr.s.en".

  Description: Sets the enabled/disabled state of the 3DNR control.

  Parameters:

  — Json::Value &jQuery

  — Json::Value &jResponse

Table 75. Control words for IF_3DNR_S_EN

Control word	Description
Enable	NR3D is enabled

- **IF_3DNR_RESET**

    This macro definition is identical to the string "3dnr.reset".

    Description: Resets the 3DNR control.

    Parameters:

    — Json::Value &jQuery

    — Json::Value &jResponse

Table 76. Control words for IF_3DNR_RESET

Control word	Description
generation	NR3D generation

- **IF_WDR_G_CFG**

    This macro definition is identical to the string "wdr.g.cfg".

    Description: Gets the configuration values for the WDR control.

    Parameters:

    — Json::Value &jQuery

    — Json::Value &jResponse

Table 77. Control words for IF_WDR_G_CFG

Control word	Description
Generation	WDR generation
Curve:d.y	WDR1 curve Dy
Curve:y.m	WDR1 curve Ym
strength	WDR strength
gain.max	WDR3 gain max
strength.global	WDR3 strength global

- **IF_WDR_S_CFG**

    This macro definition is identical to the string "wdr.s.cfg".

    Description: Sets the configuration values for the WDR control.

    Parameters:

    — Json::Value &jQuery

    — Json::Value &jResponse

Table 78. Control words for IF_WDR_S_CFG

Control word	Description
Generation	WDR generation
Curve:d.y	WDR1 curve Dy
Curve:y.m	WDR1 curve Ym
strength	WDR strength
gain.max	WDR3 gain max
strength.global	WDR3 strength global

- **IF_WDR_G_EN**

  This macro definition is identical to the string "wdr.g.en".

  Description: Gets the enabled/disabled state of the WDR control.

  Parameters:

  — Json::Value &jQuery

  — Json::Value &jResponse

Table 79. Control words for IF_WDR_G_EN

Control word	Description
Enable	The state of WDR

- **IF_WDR_S_EN**

  This macro definition is identical to the string "wdr.s.en".

  Description: Sets the enabled/disabled state of the WDR control.

  Parameters:

  — Json::Value &jQuery

  — Json::Value &jResponse

Table 80. Control words for IF_WDR_S_EN

Control word	Description
Enable	WDR is enabled

- **IF_WDR_RESET**

  This macro definition is identical to the string "wdr.reset".

  Description: Resets the WDR control.

  Parameters:

  — Json::Value &jQuery

  — Json::Value &jResponse

Table 81. Control words for IF_WDR_RESET

Control word	Description
generation	WDR generation

- IF_WB_G_CFG

    Description: Gets the configuration values for the WB control.

    This macro definition is identical to the string "wb.g.cfg".

    Parameters:

    — Json::Value &jQuery

    — Json::Value &jResponse

Table 82. Control words for IF_WB_G_CFG

Control word	Description
Matrix	Matrix
offset	Offset
red	Cc offset red
green	Cc offset Green
blue	Cc offset blue
green.r	WB gains green.R
green.b	WB gains green.B
wb.gains	WB gains

- IF_WB_S_CFG

    This macro definition is identical to the string "wb.s.cfg".

    Description: Sets the configuration values for the WB control.

    Parameters:

    — Json::Value &jQuery

    — Json::Value &jResponse

Table 83. Control words for IF_WB_S_CFG

Control word	Description
Matrix	Matrix
offset	Offset
red	Cc offset Red
green	Cc offset Green
blue	Cc offset Blue
green.r	WB gains Green.R

Table continues on the next page...

Table 83. Control words for IF_WB_S_CFG (continued)

Control word	Description
green.b	WB gains Green.B
wb.gains	WB gains

2.5.4 Dewarp control words

- **IF_DWE_S_PARAMS**

  This macro definition is identical to the string "dwe.s.params".

  Description: Sets the dewarp parameters, such as input format, output format, ROI, scale, split, dewarp type, etc.

  Parameters:

  — Json::Value &jQuery

  — Json::Value &jResponse

Table 84. Control words for IF_DWE_S_PARAMS

Control word	Description
node	Dewarp parameters, including input format, output format, ROI, scale, split, dewarp type.

- **IF_DWE_S_HFLIP**

  This macro definition is identical to the string "dwe.s.hflip".

  Description: Sets the image horizontal flip parameters.

  Parameters:

  — Json::Value &jQuery

  — Json::Value &jResponse

Table 85. Control words for IF_DWE_S_HFLIP

Control word	Description
port	Select flip port
hflip	Set vertical flip

- **IF_DWE_S_VFLIP**

  This macro definition is identical to the string "dwe.s.vflip".

  Description: Sets the image vertical flip parameters.

  Parameters:

  — Json::Value &jQuery

  — Json::Value &jResponse

Table 86. Control words for IF_DWE_S_VFLIP

Control word	Description
port	Select flip port
vflip	Set horizontal flip

Chapter 3
ISP software V4L2 programming overview

3.1 General concept

The high-level diagram of the ISP V4L2 software stack is shown in Figure 3.

Figure 3. ISP software V4L2 stack
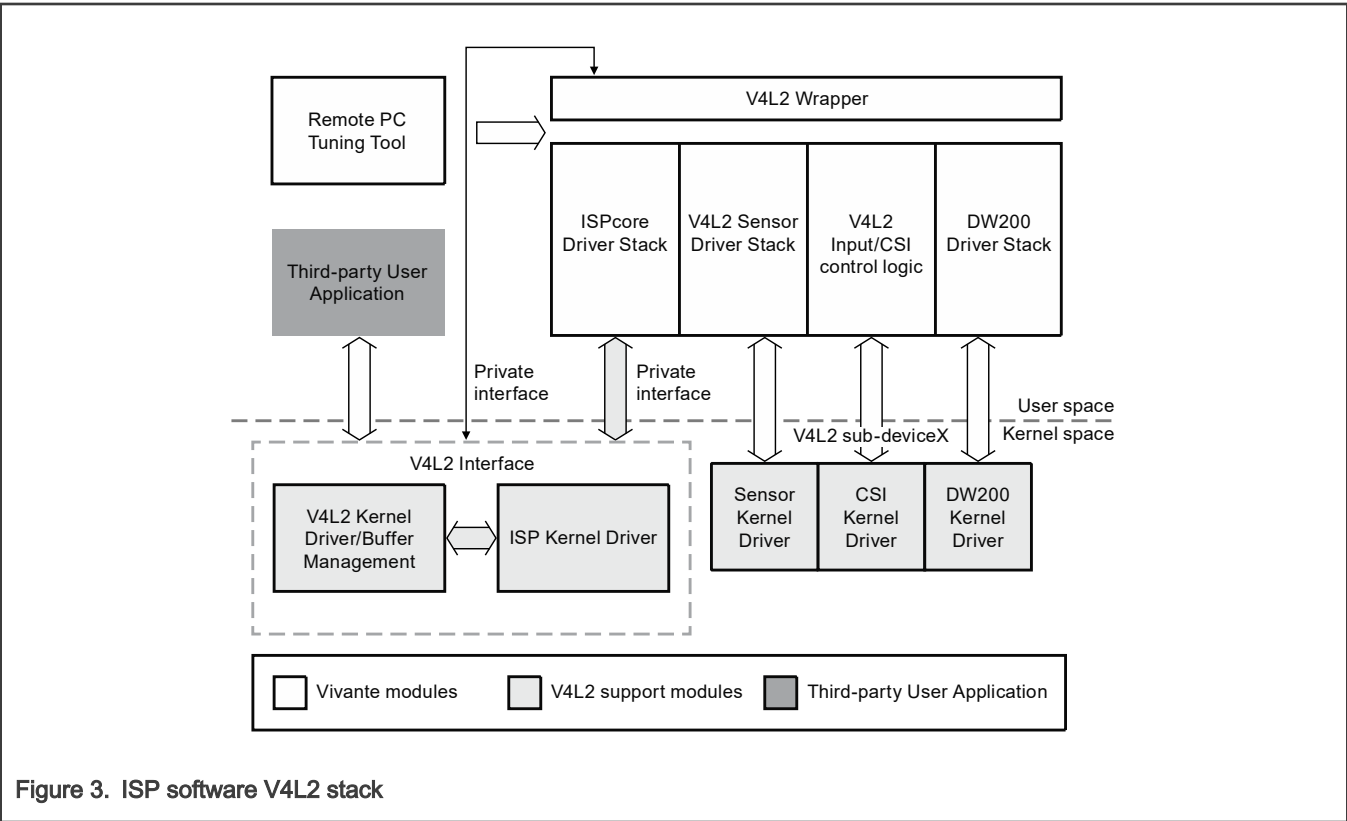
3.2 V4L2 kernel driver block diagram

ISP provides some device nodes in its file structure. Customers can operate the corresponding device through the appropriate device node(s).

Table 87. ISP device nodes

Device node/driver	Description
/dev/mediax	Enumerate video devices and subdevs
/dev/videox	Manage stream related operations and events, such as enqueue/dequeue buffers and enqueue/dequeue events
/dev/v4l2-subdevx	Manage buffers, and Control camera relevant hardware, such as MIPI/Sensor
/dev/xx	Private interface control and dispatch the commands, events, and so on.
V4L2 kernel driver	Register the V4L2_device and video_device and implement the operational functions in the video_device and vb2_queue
ISP kernel driver	ISP kernel driver, implements read/write registers, and so on.

Figure 4. V4L2 kernel driver block diagram
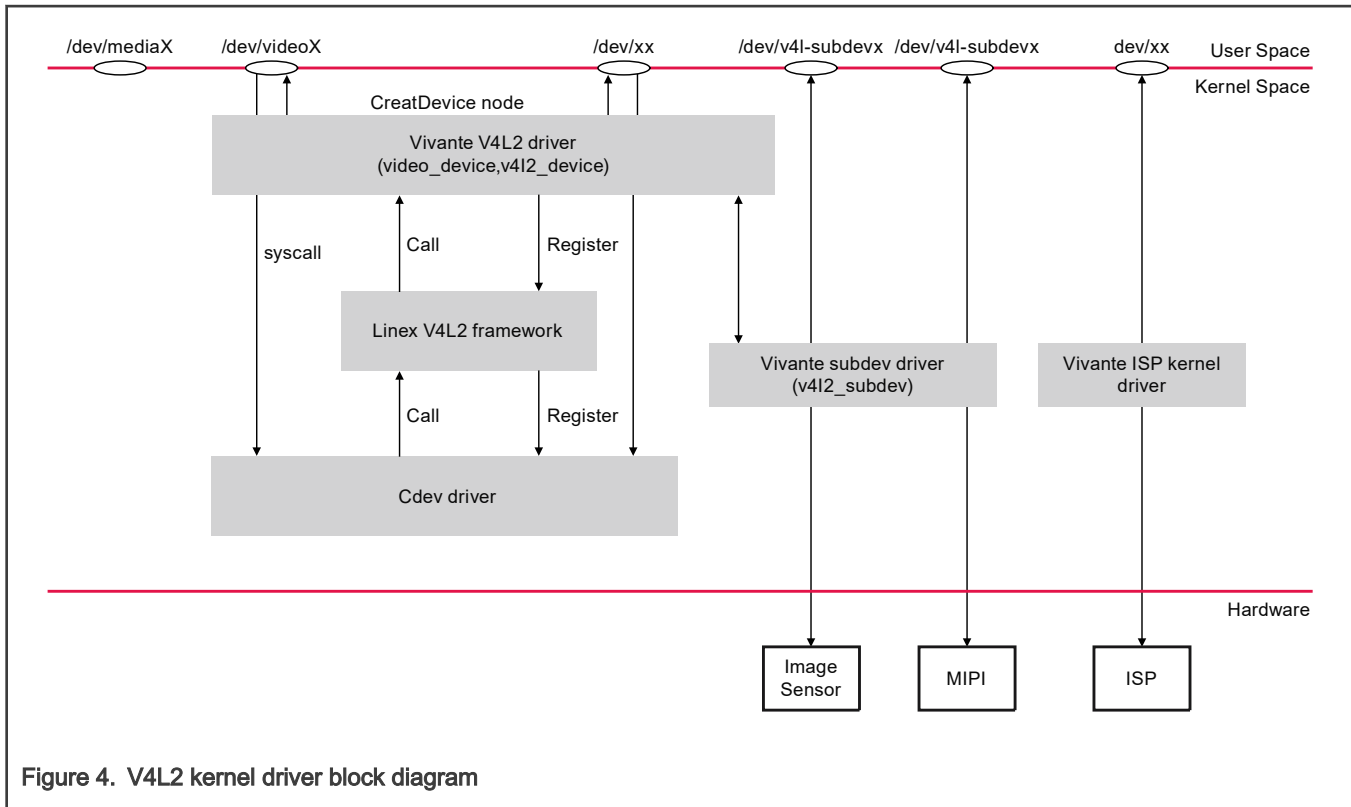
3.3 V4L2 third-party user application and ISP stack communication

The V4L2 third-party user application communicates directly with the kernel with V4L2 standard control words and V4L2 extension control commands. All the user application controls pass to the kernel space to the V4L2 kernel driver.

The V4L2 kernel driver handles the API commands and requests from the V4L2 user application, communicates to the ISP software stack and delivers image buffers to the V4L2 user application.

Sub-modules that handle the event and buffer:

• Event Queue: send/get events to/from ISP proprietary software.

• Buffer Queue: manages the vb2 buffer.
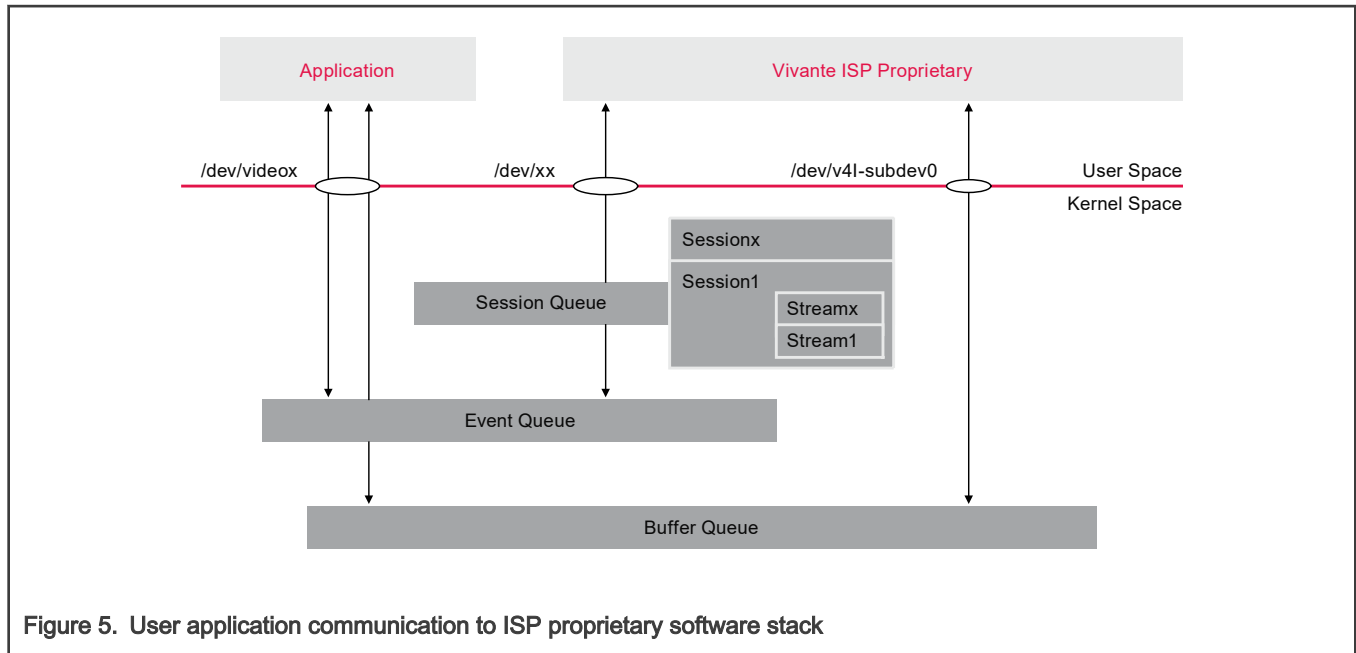
Figure 5. User application communication to ISP proprietary software stack

3.4 ISP V4L2 buffer management

There are three memory types as described in Table 88 and Figure 6.

Table 88. Memory types and buffer allocation

Memory type	Buffer allocation	Behavior
USERPTR	user space	User space and kernel space share the memory by buffer pointer
MMAP	kernel space	User space calls mmap to get pointer from kernel space
DMABUF	kernel space	User space gets the buffers using a file descriptor
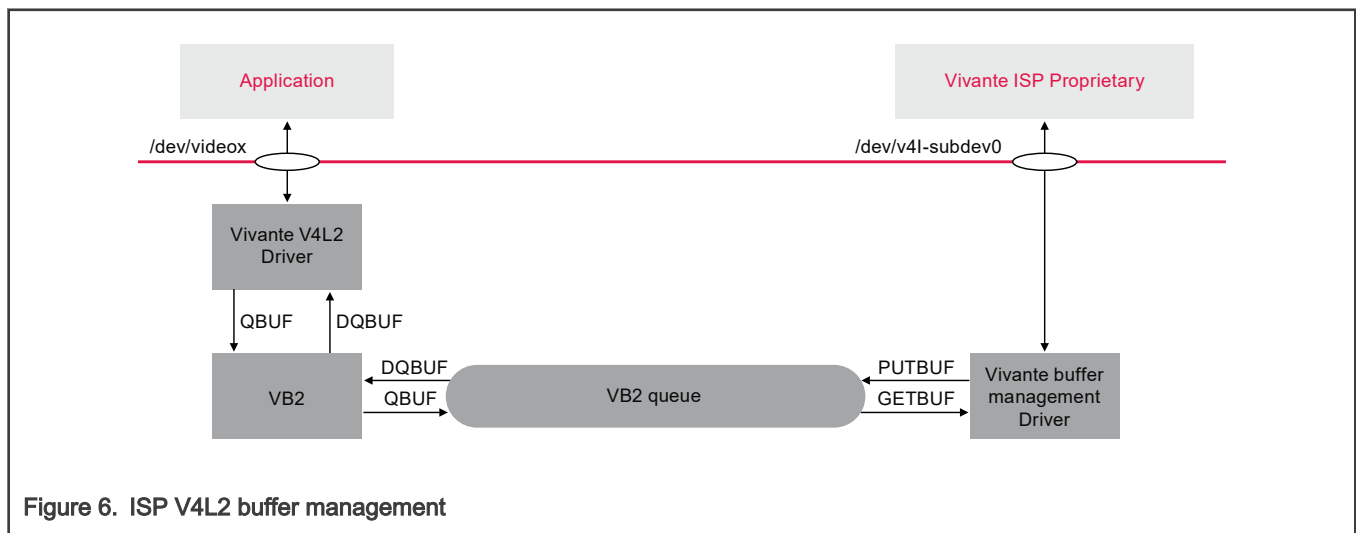
Figure 6. ISP V4L2 buffer management

NOTE
USERPTR mode is not supported.

3.5 ISP proprietary software stack

The camera manager receives messages for the kernel and dispatches these events to the corresponding sub-module for processing.

Figure 7. ISP proprietary software stack
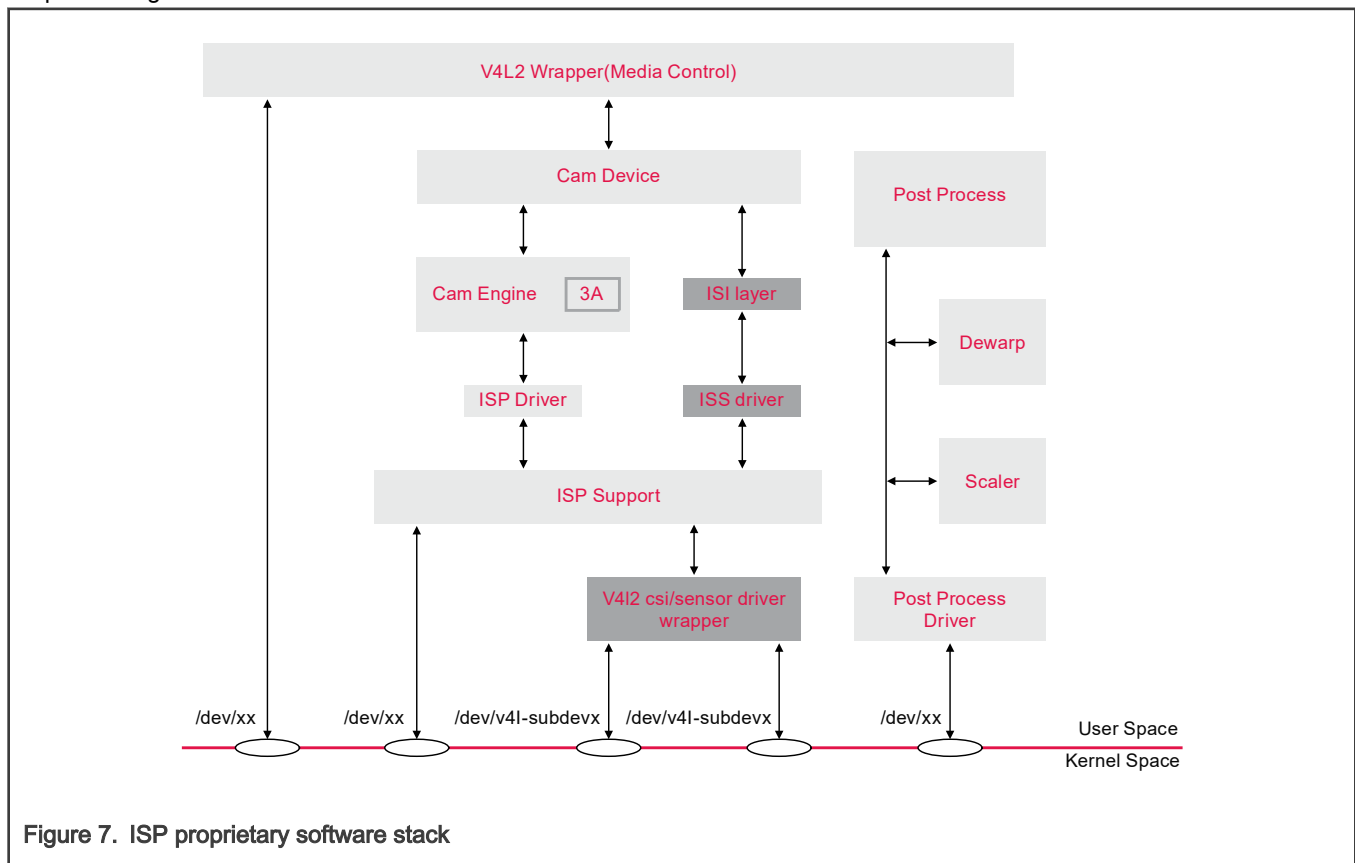
Chapter 4
Revision history

Table 89 summarizes the changes made to this document since the initial release.

Table 89. Revision history

Revision number	Date	Substantive changes
0	03/2021	Initial release

Information in this document is provided solely to enable system and software implementers to use NXP products. There are no express or implied copyright licenses granted hereunder to design or fabricate any integrated circuits based on the information in this document. NXP reserves the right to make changes without further notice to any products herein.

NXP makes no warranty, representation, or guarantee regarding the suitability of its products for any particular purpose, nor does NXP assume any liability arising out of the application or use of any product or circuit, and specifically disclaims any and all liability, including without limitation consequential or incidental damages. "Typical" parameters that may be provided in NXP data sheets and/or specifications can and do vary in different applications, and actual performance may vary over time. All operating parameters, including "typicals," must be validated for each customer application by customer's technical experts. NXP does not convey any license under its patent rights nor the rights of others. NXP sells products pursuant to standard terms and conditions of sale, which can be found at the following address: nxp.com/SalesTermsandConditions.

Security — Customer understands that all NXP products may be subject to unidentified or documented vulnerabilities. Customer is responsible for the design and operation of its applications and products throughout their lifecycles to reduce the effect of these vulnerabilities on customer's applications and products. Customer's responsibility also extends to other open and/or proprietary technologies supported by NXP products for use in customer's applications. NXP accepts no liability for any vulnerability. Customer should regularly check security updates from NXP and follow up appropriately. Customer shall select products with security features that best meet rules, regulations, and standards of the intended application and make the ultimate design decisions regarding its products and is solely responsible for compliance with all legal, regulatory, and security related requirements concerning its products, regardless of any information or support that may be provided by NXP. NXP has a Product Security Incident Response Team (PSIRT) (reachable at PSIRT@nxp.com) that manages the investigation, reporting, and solution release to security vulnerabilities of NXP products.

arm