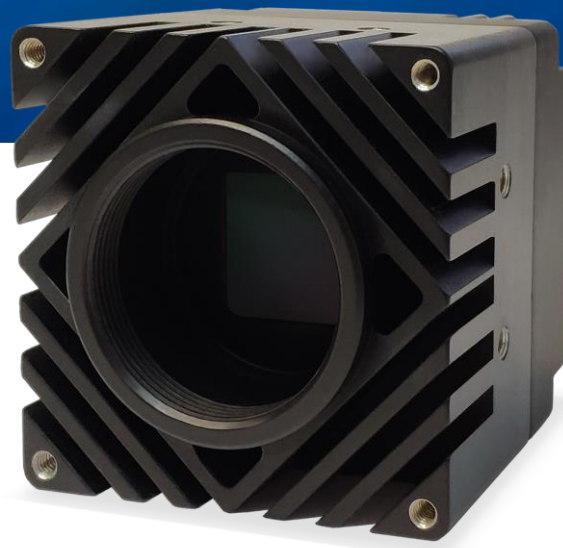


IRON

SDI

User Manual



November 2021 - Rev 2.1



International Distributors

sky blue
microsystems

Sky Blue Microsystems GmbH
Geisenhausenerstr. 18
81379 Munich, Germany
+49 89 780 2970, info@skyblue.de
www.skyblue.de

ZERIF
TECHNOLOGIES LTD.
A SKY BLUE COMPANY, FOUNDED 1999

In Great Britain:
Zerif Technologies Ltd.
Winnington House, 2 Woodberry Grove
Finchley, London N12 0DR
+44 115 855 7883, info@zerif.co.uk
www.zerif.co.uk

Table of Contents

| | | |
|-------|---------------------------------|----|
| 1 | Figures & Tables | 5 |
| 1.1 | List of Figures | 5 |
| 1.2 | List of Tables | 5 |
| 2 | Revision History | 6 |
| 3 | Introduction | 7 |
| 3.1 | Safety Precautions | 7 |
| 3.2 | Disclaimer | 8 |
| 4 | Overview | 9 |
| 5 | Supported Formats | 10 |
| 6 | Quick Start Guide | 11 |
| 6.1 | Camera Connection | 11 |
| 6.1.1 | Power and Image Streaming | 11 |
| 6.1.2 | Advanced Control | 11 |
| 6.2 | Troubleshooting | 12 |
| 7 | Image Processing Flow | 13 |
| 8 | Camera Operation | 14 |
| 8.1 | SDIControlPoint | 14 |
| 8.2 | Terminal Usage | 14 |
| 8.2.1 | Terminal | 14 |
| 8.2.2 | Terminal Settings | 14 |
| 8.2.3 | General Command Format | 14 |
| 8.3 | Commands | 15 |
| 8.3.1 | Help Dialog | 15 |
| 8.3.2 | Set Functions | 15 |
| 8.3.3 | Get Functions | 15 |
| 8.3.4 | List of Commands | 15 |
| 8.4 | RS485 Interface | 19 |
| 8.4.1 | rs485_baud <baud_rate> | 19 |
| 8.4.2 | rs485_addr <address> | 19 |
| 8.4.3 | rs485_bc_addr <address> | 20 |
| 8.4.4 | rs485_bc_master <address> | 20 |
| 8.5 | Settings Handling | 21 |
| 8.5.1 | save_settings <id> | 21 |
| 8.5.2 | load_settings <id> | 21 |
| 8.5.3 | default_settings <id> | 21 |
| 8.5.4 | reset_settings | 21 |
| 8.5.5 | dump_settings | 21 |
| 8.6 | System Commands | 22 |
| 8.6.1 | prompt <flag> | 22 |
| 8.6.2 | reboot | 22 |
| 8.6.3 | fw_update | 22 |
| 8.6.4 | version | 22 |
| 8.6.5 | name <name string> | 23 |
| 8.6.6 | flip <mode> | 23 |
| 8.6.7 | temp <sensor id> | 23 |
| 8.7 | Camera Commands | 24 |
| 8.7.1 | cam_gain <gain> | 24 |
| 8.7.2 | cam_exposure <time> | 24 |
| 8.7.3 | cam_info | 24 |
| 8.7.4 | identify | 25 |
| 8.8 | Video Commands | 26 |

| | | |
|---------|--|----|
| 8.8.1 | video_mode <mode>..... | 26 |
| 8.8.2 | downscale <channel=1> <downscale> <reserved=0> | 27 |
| 8.8.3 | sdi_black <offset>..... | 28 |
| 8.8.4 | sdi_white <offset> | 28 |
| 8.8.5 | sdi_range <flag>..... | 28 |
| 8.8.6 | post_bright <offset> | 28 |
| 8.8.7 | post_cont <factor> | 29 |
| 8.8.8 | post_sat <factor>..... | 29 |
| 8.8.9 | post_hue <offset>..... | 29 |
| 8.8.10 | wb | 30 |
| 8.8.11 | wb_threshold <threshold> | 30 |
| 8.8.12 | awb <flag> | 30 |
| 8.8.13 | wb_preset <id>..... | 30 |
| 8.8.14 | gain_red <gain> | 30 |
| 8.8.15 | gain_blue <gain>..... | 31 |
| 8.8.16 | gain_green <gain> | 31 |
| 8.8.17 | black_master <offset-red> <offset-green> <offset-blue> | 31 |
| 8.8.18 | black_red <offset> | 31 |
| 8.8.19 | black_blue <offset> | 32 |
| 8.8.20 | black_green <offset> | 32 |
| 8.8.21 | flare <red level> <green level> <blue level>..... | 32 |
| 8.8.22 | color_cross <c0> .. <c8>..... | 32 |
| 8.8.23 | color_cross_offset <red_offset> <green_offset> <blue_offset>..... | 33 |
| 8.8.24 | color_conv <c0> .. <c8> | 33 |
| 8.8.25 | color_space <color space>..... | 34 |
| 8.8.26 | stat_roi <width><height><offsetX><offsetY> | 34 |
| 8.8.27 | stat_roi_info <maxWidth><maxHeight><widthStep><heightStep> | 34 |
| 8.8.28 | cam_roi_offset <offsetX><offsetY>..... | 34 |
| 8.8.29 | cam_roi_offset_info <offsetXMax><offsetYMax><offsetXStep><offsetYStep> | 34 |
| 8.9 | Defect Pixel Correction..... | 35 |
| 8.9.1 | dpc <flag> | 35 |
| 8.9.2 | dpc_add_pixel <x> <y> | 35 |
| 8.9.3 | dpc_del_pixel..... | 36 |
| 8.9.4 | dpc_save | 36 |
| 8.9.5 | dpc_load | 36 |
| 8.10 | Knee Function..... | 36 |
| 8.10.1 | knee <flag> <knee_point> <knee_slope> <white_clip> | 36 |
| 8.11 | Loop-up Table Management | 37 |
| 8.11.1 | lut_enable <reserved=0> <flag>..... | 37 |
| 8.11.2 | lut_mode <mode> | 38 |
| 8.11.3 | lut_preset <index>..... | 38 |
| 8.11.4 | lut_sample_master <xi_0> <yi_0> ... <xi_7> <yi_7>..... | 38 |
| 8.11.5 | lut_sample_red <xi_0> <yi_0> ... <xi_7> <yi_7>..... | 39 |
| 8.11.6 | lut_sample_green <xi_0> <yi_0> ... <xi_7> <yi_7>..... | 39 |
| 8.11.7 | lut_sample_blue <xi_0> <yi_0> ... <xi_7> <yi_7>..... | 39 |
| 8.11.8 | lut_interpolate..... | 39 |
| 8.11.9 | lut_interpolate_red | 39 |
| 8.11.10 | lut_interpolate_green | 39 |
| 8.11.11 | lut_interpolate_blue..... | 39 |
| 8.11.12 | lut_reset_master | 39 |
| 8.11.13 | lut_reset_red | 40 |
| 8.11.14 | lut_reset_green | 40 |
| 8.11.15 | lut_reset_blue | 40 |

| | | |
|---------|---|----|
| 8.11.16 | lut_fun_rec709 <threshold> <linear-contrast> <linear-brightness> <contrast> <gamma> <brightness> | 40 |
| 8.11.17 | lut_fast_gamma <gamma> | 42 |
| 8.11.18 | lut_fixed_mode <mode> | 42 |
| 8.12 | Image statistic commands | 42 |
| 8.12.1 | stat_rgb | 42 |
| 8.13 | Auto Exposure | 43 |
| 8.13.1 | aec <enable> <setPoint> <speed> <clmTolerance> <activeGain> <activeExposure> <activeApt> <maxExposure> <maxGain> <reserved> | 43 |
| 8.14 | Genlock | 45 |
| 8.14.1 | genlock <mode> | 45 |
| 8.14.2 | genlock_status | 45 |
| 8.14.3 | genlock_lol_filter <time_ms> | 45 |
| 8.14.4 | genlock_offset <v_offset> <h_offset> | 46 |
| 8.14.5 | genlock_offset_info <v_offset_phase><h_offset_phase> | 46 |
| 8.14.6 | genlock_term <enable> | 46 |
| 8.14.7 | genlock_crosslock <vmode> | 46 |
| 8.15 | SDI Time Code | 47 |
| 8.15.1 | timecode <hour> <minute> <second> | 47 |
| 8.15.2 | timecode_hold <flag> | 47 |
| 9 | Hardware Reference | 48 |
| 9.1 | Status LED | 48 |
| 9.2 | Micro BNC Connector | 48 |
| 10 | Appendix 1: Firmware Update | 49 |

Preliminary

1 Figures & Tables

1.1 List of Figures

| | |
|--|----|
| Figure 1 – Camera connections diagram..... | 11 |
| Figure 2 – Image Processing Flow | 13 |
| Figure 3 – Downscale example..... | 27 |
| Figure 4 – ROI position in relation to the origin | 34 |
| Figure 5 – Defect pixel correction position | 35 |
| Figure 6 – Pushing the Micro-BNC connector into place | 48 |
| Figure 7 – Twisting the connector and securing it in position..... | 48 |
| Figure 8 – Serial communication example | 49 |
| Figure 9 – Firmware terminal initiation..... | 49 |
| Figure 10 – Firmware update fail | 50 |
| Figure 11 – Firmware update process..... | 50 |
| Figure 12 – Firmware update succession..... | 50 |

1.2 List of Tables

| | |
|---|----|
| Table 1 – Revision History | 6 |
| Table 2 – Supported SDI parameters | 10 |
| Table 3 – List of commands..... | 18 |
| Table 4 – Connector indicator lamp states | 48 |

2 Revision History

| Ver | Date | Notes |
|-----|---------|---|
| 1.0 | 01.2020 | Initial release |
| 1.1 | 05.2020 | Added the Subsampling mode |
| 1.2 | 12.2020 | Added section no. 6: Quick Startup Guide Added section no. 7: Image Processing Flow. Added appendix no. 1: firmware update. Section no. 8: updated new commands and re-ordered them. |
| 1.3 | 02.2021 | Section no. 8: added subsection 8.13 – “Genlock” |
| 2.0 | 05.2021 | Added Region of Interest support Added RS485 interface support and Multi-Camera Mode Added video output time-code control support Added extended auto-exposure features configurations |
| 2.1 | 11.2021 | Added Genlock crosslock and termination Updated genlock_offset command |

Table 1 – Revision History

Preliminary

3 Introduction

3.1 Safety Precautions

With your *Iron* camera in hand, please take the time to read through the precautions listed below in order to prevent preventable and unnecessary injuries and damage to you, other personnel or property. Read these safety instructions carefully prior to your first use of the product, as these precautions contain safety instructions that must be observed. After reading through this manual, be sure to follow it to prevent misuse of product.



Caution! Read Carefully and do not disregard these instructions.

In the event of a failure, disconnect the power supply

Disconnect the power supply immediately and contact our sales personnel for repair. Continuing to use the product in this state may result in a fire or electric shock.

If an unpleasant smell or smoking occurs, disconnect the power supply.

Disconnect the power supply immediately! Continuing to use the product in this state may result in a fire or electric shock. After verifying that no smoking is observed, contact our sales personnel for repair.

Do not disassemble, repair or modify the product.

This may result in a fire or electric shock due to a circuit shortage or heat generation. Contact our sales personnel prior to inspection, modification or repair.

Do not place the product on unstable surfaces.

Otherwise, it may drop or fall, resulting in injury to persons or the camera.

Do not use the product if dropped or damaged.

Otherwise, a fire or electric shock may occur.

Do not touch the product with metallic objects.

Otherwise, a fire or electric shock may occur.

Do not place the product in dusty or humid environments, nor where water may splash.

Otherwise, a fire or electric shock may occur.

Do not wet the product or touch it with wet hands.

Otherwise, the product may fail or it may cause a fire, smoking or electric shock.

Do not touch the gold-plated sections of the connectors on the product.

Otherwise, the surface of the connector may be contaminated by sweat or skin-oil, resulting in contact failure of a connector, malfunction, fire or electric shock due to static electricity discharge.

Do not use or place the product in the following locations.

- Unventilated areas such as closets or bookshelves.
- Near oils, smoke or steam.
- Next to heat sources.
- A closed (and not running) car where the temperature becomes high.
- Static electricity replete locations
- Near water or chemicals.

Otherwise, a fire, electric shock, accident or deformation may occur due to a short circuit or heat generation.

Do not place heavy objects on the product.

Otherwise, the product may be damaged.

Be sure to discharge static electricity from body before touching any sensitive electronic components.

The electronic circuits in your computer and the circuits on the *Iron* camera and the *Predator II* board are sensitive to static electricity and surges. Improper handling may seriously damage the circuits. In addition, do not let your clothing come in contact with the circuit boards or components. Otherwise, the product may be damaged.

3.2 Disclaimer

This product should only be used for image capturing and processing. **KAYA Instruments** will assume no responsibility for any damage that may ensue by the use of the camera for any purpose other than intended, as previously stated. Without detracting from what was previously written, please be advised that the company will take no responsibility for any damages caused by:

- Earthquake, thunder strike, natural disasters, fire caused by use beyond our control, wilful and/or accidental misuse and/or use under other abnormal and/or unreasonable conditions.
- Secondary damages caused by the use of this product or its unusable state (business interruption or others).
- Use of this product in any manner that contradicts this manual or malfunctions that may occur due to connection to other devices. Damage to this product that is out of our control or failure due to modification
- Accidents and/or third parties that may be involved.

Additionally, **KAYA Instruments** assumes no responsibility or liability for:

- Erasure or corruption of data caused by the use of this product.
- Any consequences or other abnormalities following the use of this product

Repairs to this product are carried out by replacing it on a chargeable basis and not by repairing the faulty device. Non-chargeable replacement is offered for initial failure, as long as it is reported no later than two weeks post-delivery of the product.

4 Overview



This user manual provides a detailed overview of KAYAS's IRON SDI cameras operation. All cameras are suited for a wide variety of applications such as low light surveillance, special effects, sports broadcasting, etc.

With our customers' convenience in mind we had made sure that connecting and streaming can be easily achieved in few easy steps and require little configurations. Control of the camera as well as advanced configurations are made easy using the [SDIControlPoint](#) software and terminal, enabling enhanced streaming and image processing of the cameras' outputs. Hardware reference is covered in chapter 9, [Hardware Reference](#).

It is important to note that some parameters might vary slightly compared to this document or may be absent entirely, subject to the active firmware capabilities: a firmware upgrade might be needed to support complete functionality set. Please feel free to contact our support team at info@skyblue.de with any questions that may arise.

5 Supported Formats

Iron SDI camera support multiple standard video formats including HD-SDI, 3G-SDI, 6G-SDI and 12G-SDI, depending on the camera's sensor resolution. Changing format can be done dynamically, using serial interface, or format configuration can be saved and loaded on startup.

Table no. 2 describes the supported formats:

| Mode | IRON SDI 265 | IRON SDI 305 | Video Standard | Resolutions supported | Frame rates supported (fps) |
|----------------------------|--------------|--------------|-------------------------------------|----------------------------|-----------------------------|
| HD-SDI | V | V | ST 292 (ST 274) | 1080i 10-bit 4:2:2 | 50, 59.94, 60 |
| | | | | 1080p 10-bit 4:2:2/RAW | 23.98, 24, 25, 29.97, 30 |
| ST 292 (ST 2048-2) | | | 2K 10-bit 4:2:2 | 23.98, 24, 25, 29.97, 30 | |
| 3G-SDI | | | ST 425-1 (ST 274) | 1080p 10-bit 4:2:2/RAW | 50, 59.94, 60 |
| 6G-SDI | X | V | ST 425-1 (ST 2048-2) | 2K 10-bit 4:2:2 | 47.95, 48, 50, 59.94, 60 |
| | | | | ST 2081-10 M1, (ST 2036-1) | UHD 10-bit 4:2:2 |
| ST 2081-10 M1, (ST 2048-1) | | | 4K 10-bit 4:2:2 | 23.98, 24, 25, 29.97, 30 | |
| 12G-SDI | | | ST 2082-10 M1, ST 425-5 (ST 2036-1) | UHD 10-bit 4:2:2 | 50, 59.94, 60 |
| 12G-SDI | | | ST 2082-10 M1, ST 425-5 (ST 2048-1) | 4K 10-bit 4:2:2 | 47.95, 48, 50, 59.94, 60 |

Table 2 – Supported SDI parameters

6 Quick Start Guide

6.1 Camera Connection

KAYA's Iron SDI cameras are easy to set up and operate. The camera needs to be connected to a power source, using the supplied cable, and to an SDI capture card or monitor. Under normal circumstances the camera will start streaming to an SDI input as soon as it is powered.

Advanced camera settings can be configured using the provided serial command line interface. The camera can also be controlled using [KAYA's SDIControlPoint](#) application. For more details, see section 8, [Camera Operation](#).

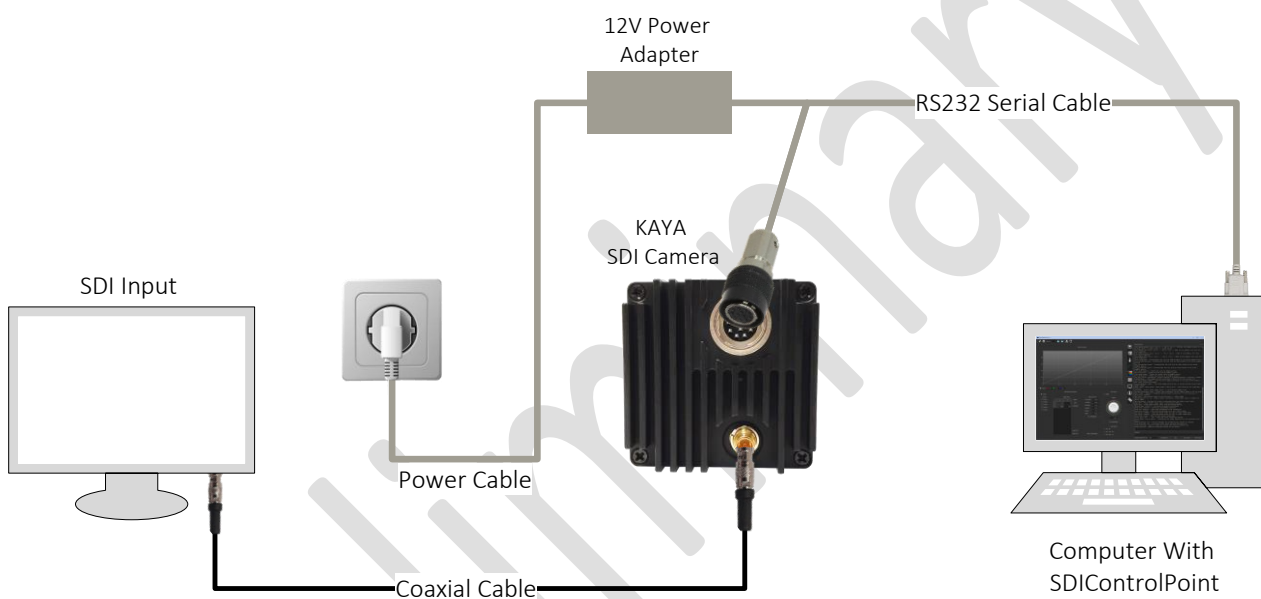


Figure 1 – Camera connections diagram

6.1.1 Power and Image Streaming

1. Connect the camera to an SDI Input using a coaxial cable
2. Connect the camera to a 7-18V power source.
3. Video should start streaming to the SDI capture card or monitor.

6.1.2 Advanced Control

1. Connect the camera's GPIO to a serial RS232 port (cable provided).
2. Connect to the camera's command user interface via a standard serial terminal (see section 8.2, [Terminal Usage](#)) or [KAYA's SDIControlPoint](#) application.
3. To change the video mode use the video_mode command (see section 8.8.1, [video_mode](#)).
4. To save the current settings use the save_settings command (see section 8.5.1, [save_settings](#)).

6.2 Troubleshooting

1. Make sure that the camera is connected to a 7-18V power source.
2. Make sure that all components are properly connected (camera, cables and capture card). See Figure 1 and section 9.2 [Micro BNC Connector](#).
3. Make sure that the coaxial cable supports the proper output stream bandwidth:
 - 3G SDI for 2k video streaming.
 - 12G SDI for 4k video streaming.
4. Check to see whether the LED on the back of the camera is blinking green. The issue may be resolved by power cycling the camera (turning power off and back on). More LED indicators are described in section no. 9.1 ([Status LED](#)).
5. Make sure the SDI input (SDI capture card or monitor) supports the output video format of the camera – contact the SDI input’s manufacturer for support and additional information. To change the camera’s output video mode, see section 8.8.1, [video_mode <mode>](#).

7 Image Processing Flow

Figure 2 describes the image processing flow in KAYA’s SDI cameras, from a raw image acquired by the sensor and to an SDI-standard compatible output. All commands implemented in the processing pipe are described in chapter no. 8, “Camera Operation”.

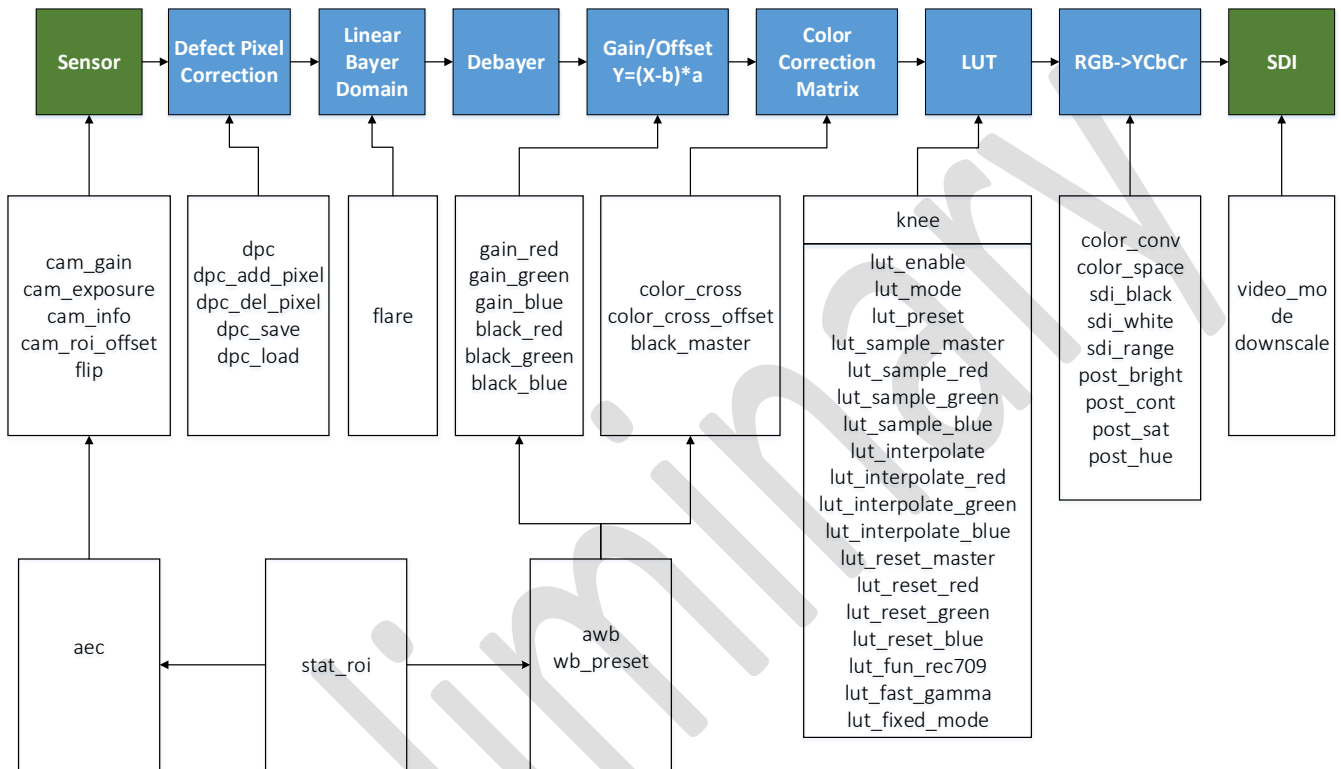


Figure 2 – Image Processing Flow

8 Camera Operation

8.1 SDIControlPoint

KAYA Instruments provides a free, open-source camera control software (named: SDIControlPoint). A downloadable file as well as the software's user manual (KAYA SDIControlPoint Manual.pdf) are available at the following address for your convenience: cloud.kayainstruments.com/s/SDI

8.2 Terminal Usage

8.2.1 Terminal

The Iron SDI can be controlled via a simple terminal connection. All commands consist of ASCII characters.

8.2.2 Terminal Settings

The terminal interfaces are either RS232 or RS485 and are connected via the GPIO connector. The default interface settings are:

- 115200 baud
- 8 bit data
- no parity
- 1 stop bit
- no flow control

The camera accepts commands in text form and responds in text messages. Every command is confirmed by either an "OK" or a "FAIL" to indicate whether it had been successfully implemented or not, respectively; errors have to be handled by the user or a host software which is used to control the device. Depending on the **prompt** setting one may receive a "=>" prompt after start up. Commands can then be sent to control the device or receive information.

8.2.3 General Command Format

If the **prompt** command is enabled the firmware will send its prompt ("=> ") when awaiting commands. Every command line is accepted as a single text line, terminated by either CR or LF. The command consists of a command string, possibly followed by one or more parameters separated by a single space. Depending of the command the parameters can come in one of these formats:

- Signed decimal : -323, 422
- Hexadecimal: 0x35ff34aa
- String : any_string

The command will be executed and may produce some text output in a single line or more, followed by a single status line consisting of either "OK" or "FAIL". In case of failure an error code may be added in the same line, e.g. "FAIL 3". The lines sent by the firmware are terminated with both CR and LF. Most commands do have parameters. The parameters depend on the given command.

8.3 Commands

8.3.1 Help Dialog

Typing **help** will give a full list of supported commands. It is possible to get a detailed help for each command by typing "help <cmd>".

Example:

```
help video_mode<Enter>
video_mode <video mode> - set video mode
```

8.3.2 Set Functions

Running a command with its required matching parameters will change the settings.

Example:

```
Command: video_mode 5
Response: OK (returned)
```

8.3.3 Get Functions

Any setting can be checked by running a function **without parameters**. The output of the **get** function represents a valid command followed by parameters. This string can be parsed by external applications to extract the current settings out of the system.

Example:

```
Command: video_mode
Response: video_mode 5 (returned)
        OK (returned)
```

8.3.4 List of Commands

| # | Command | Parameters | Description |
|----|------------------|-------------|--|
| 1 | help | | Gives a full list of supported commands |
| 2 | rs485_baud | <baud_rate> | Sets the baud rate of the serial communication |
| 3 | rs485_addr | <address> | Sets the unique device serial address |
| 4 | rs485_bc_addr | <address> | Sets the serial broadcast address |
| 5 | rs485_bc_master | <address> | Sets master device for command acknowledge of broadcast commands |
| 6 | save_settings | <id> | Saves the current settings into camera non-volatile memory |
| 7 | load_settings | <id> | load saved settings on demand |
| 8 | default_settings | <id> | Select the default settings set applied on startup |
| 9 | reset_settings | | Resets the system into factory default run |
| 10 | dump_settings | | dumps current settings configured on the device |
| 11 | prompt | <flag> | Set prompt mode of the command output |

| | | | |
|----|--------------------|--|---|
| 12 | reboot | | Performs a system reboot |
| 13 | fw_update | | Set the system into firmware update mode |
| 14 | version | | Dumps a detailed version information about the system |
| 15 | name | <name string> | Sets the device name |
| 16 | flip | <mode> | Sets the image flip or rotation mode |
| 17 | temp | <sensor id> | Sets the image flip or rotation mode |
| 18 | cam_gain | <gain> | Set the analog gain for the sensor |
| 19 | cam_exposure | <time> | Sets the exposure time/shutter width |
| 20 | cam_info | | Reports the min / max gain and exposure |
| 21 | identify | | display device connection details |
| 22 | video_mode | <mode> | Sets the output video mode |
| 23 | downscale | <channel=1> <downscale> <reserved=0> | The downscale command is used to output Full HD or 2K resolution from UHD or 4K downsampled source image |
| 24 | sdi_black | <offset> | Sets the black level for SDI in legal range mode |
| 25 | sdi_white | <offset> | Sets the white level for SDI in legal range mode |
| 26 | sdi_range | <flag> | Sets the SDI output range type |
| 27 | post_bright | <offset> | Sets post processing brightness |
| 28 | post_cont | <factor> | Sets post processing contrast |
| 29 | post_sat | <factor> | Sets post processing color saturation |
| 30 | post_hue | <offset> | Sets post processing color hue offset angle |
| 31 | wb | | Triggers single shot white-balance |
| 32 | wb_threshold | <threshold> | Sets wb maximum calculation threshold |
| 33 | awb | <flag> | Enable continuous white balance adjustment. |
| 34 | wb_preset | <id> | Sets calibrated white balance presets |
| 35 | gain_red | <gain> | Sets gain factor for red component for selected output channel |
| 36 | gain_blue | <gain> | Sets gain factor for blue component for selected output channel |
| 37 | gain_green | <gain> | Sets gain factor for green component for selected output channel |
| 38 | black_master | <offset-red> <offset-green> <offset-blue> | Sets the black-level offset for red, green and blue-components for selected output channel. |
| 39 | black_red | <offset> | Sets offset for red component for selected output channel as black level setting |
| 40 | black_blue | <offset> | Sets offset for blue component for selected output channel as black level setting |
| 41 | black_green | <offset> | Sets offset for green component for selected output channel as black level setting |
| 42 | flare | <red level> <green level> <blue level> | Sets flare compensation level (= Defogging) |
| 43 | color_cross | <c0> .. <c8> | Sets the color cross talk matrix |
| 44 | color_cross_offset | <red_offset> <green_offset> <blue_offset> | Sets the color cross talk offset |
| 45 | color_conv | <c0> .. <c8> | Sets the color conversion matrix |
| 46 | color_space | <color space> | Change the color space matrix |
| 47 | stat_roi | <width> <height> <offsetX> <offsetY> | Set the ROI where auto compensation calculations are made |
| 48 | stat_roi_info | <maxWidth> <maxHeight> <widthStep> <heightStep> | Gets the auto compensation max ROI coordinates |
| 49 | cam_roi_offset | <offsetX> <offsetY> | Moves the absolute image ROI position across the sensor. Default position in center |

| | | | |
|----|---------------------------------------|--|---|
| 50 | cam_roi_offset_info | <offsetXMax> <offsetYMax> <offsetXStep> <offsetYStep> | Gets the maximum camera roi offset |
| 51 | dpc | <flag> | Enable deflect pixel correction |
| 52 | dpc_add_pixel | <x> <y> | Adds a pixel coordinate to defect pixel table or dumps the whole table |
| 53 | dpc_del_pixel | | Deletes all pixel coordinates from the defect pixel table |
| 54 | dpc_save | | Saves defect pixel table to persistent memory |
| 55 | dpc_load | | Loads defect pixel table from persistent memory |
| 56 | knee | <flag> <knee_point> <knee_slope> <white_clip> | Knee function for highlight control |
| 57 | lut_enable | <reserved=0> <flag> | Enables the function for look up table |
| 58 | lut_mode | <mode> | Selects the LUT operational mode |
| 59 | lut_preset | <index> | Selects the preset storage for the current LUT interpolator |
| 60 | lut_sample_master | <xi_0><yi_0> ... <xi_7> <yi_7> | Defines the sample points in a lookup table for all colors |
| 61 | lut_sample_red | <xi_0><yi_0> ... <xi_7> <yi_7> | Same as lut_sample, but only for the red component |
| 62 | lut_sample_green | <xi_0><yi_0> ... <xi_7> <yi_7> | Same as lut_sample, but only for the green component |
| 63 | lut_sample_blue | <xi_0><yi_0> ... <xi_7> <yi_7> | Same as lut_sample, but only for the blue component |
| 64 | lut_interpolate | | Interpolates all look up tables based on the given sample point |
| 65 | lut_interpolate_red | | Interpolates the red look up table based on the given sample points |
| 66 | lut_interpolate_green | | Interpolates the green look up table based on the given sample points |
| 67 | lut_interpolate_blue | | Interpolates the blue look up table based on the given sample points |
| 68 | lut_reset_master | | Clears all color channels look up sample points |
| 69 | lut_reset_red | | Clears all red look up sample points |
| 70 | lut_reset_green | | Clears all green look up sample points |
| 71 | lut_reset_blue | | Clears all blue look up sample points |
| 72 | lut_fun_rec709 | <threshold> <linear-contrast> <linear-brightness> <contrast> <gamma> <brightness> | Sets new LUT sample points according to REC.709 for selected LUT preset |
| 73 | lut_fast_gamma | <gamma> | The fast gamma function uses the same formula as the lut_fun_rec709 command, but the user only has to specify the desired gamma value. |
| 74 | lut_fixed_mode | <mode> | The fixed gamma mode contains three presets which are shown in the table below |
| 75 | stat_rgb | | Dumps the average image value for each color channel (RGB) |
| 76 | aec | <enable> <setPoint> <speed> <clmTolerance> <activeGain> <activeExposure> <activeApt> <maxExposure> <maxGain> | Auto Exposure and Gain are used to control the picture brightness by adjusting Exposure and Gain values automatically in order to reach desired luminance level |
| 77 | genlock | <mode> | The genlock mechanism is used to synchronize multiple cameras video signals |
| 78 | genlock_status | | Used to check the current genlock status |

| | | | |
|----|-------------------------------------|--------------------------------------|---|
| 79 | genlock_loi_filter | <time_ms> | Can be configured to prevent glitches in the unstable genlock signals |
| 80 | genlock_offset | <v_offset> <h_offset> | The vertical position offset that is added to the reference sync signal |
| 81 | genlock_offset_info | <v_offset_phase> <h_offset_pahse> | The vertical position offset that is added to the reference sync signal |
| 82 | genlock_term | <enable> | Modifies the genlock input/output 75R termination |
| 83 | genlock_crosslock | <vmode> | Set the genlock input Tri-level signal other than current video mode |
| 84 | timecode | <hour> <minute> <second> | Sets SDI time code which is embedded into the SDI signal |
| 85 | timecode_hold | <flag> | This command can be used to hold the timecode on the SDI output |

Table 3 – List of commands

Preliminary

8.4 RS485 Interface

An RS485 Bus allows for multiple Iron SDI systems to be used simultaneously with a single RS485 Master controller; please note that an RS485 to RS232 adapter must be used.

Commands can also be sent to a group of cameras at once as long as they are connected to the same RS485 bus. To accommodate for this, each camera has a personal address and a broadcast address which can be set via the [rs485_addr](#) and [rs485_bc_addr](#) commands, respectively. The former is used for individual control while the latter is used to communicate commands to a group.

Master and slave communication is commenced in the following command format for both individual and group communication: <address> <command> <parameter>.

8.4.1 rs485_baud <baud_rate>

Sets the baud rate for the RS485 terminal interface. Possible baud rates are:

| # | Baud Rate |
|---|------------------|
| 1 | 9600 |
| 2 | 14400 |
| 3 | 19200 |
| 4 | 57600 |
| 5 | 115200 (default) |

Remarks:

1. It is advised to leave the baud rate at 115200 (default) for faster device response.
2. Setting RS485 baud rate will also change the RS232 interface baud rate.

8.4.2 rs485_addr <address>

Sets the address for the RS485 interface. It is impossible to set the device's address to be the same as the broadcast address, an attempt to do so will return "FAIL".

| Value | Default | Minimal | Maximal |
|----------------|---------|---------|---------|
| Device Address | 1 | 0 | 99 |

Remarks:

1. Make sure, that each address is unique on your RS485 bus system. Address conflicts can be checked and resolved at any time via RS232 console, or by removing devices from the bus until the contention is solved.
2. Address 100 is reserved as a fail-safe address: all devices will always replay to commands sent over this address, as if it was their current device address. This can be used to identify or change the device address if it has been lost.

8.4.3 rs485_bc_addr <address>

Sets the broadcast address for the RS485 interface. It is impossible to set the broadcasting address to be the same as the device's address, an attempt to do so will return "FAIL".

| Value | Default | Minimal | Maximal |
|-------------------|---------|---------|---------|
| Broadcast Address | 0 | 0 | 99 |

To change the broadcast address of an already existing broadcast group, simply send an **rs_485_bc_addr** command over the broadcast channel

Example 1:

```
// change the broadcast address of all connected cameras to 8
rs485_bc_addr 8
```

Example 2:

```
// change the broadcast address of all connected cameras associated
// with existing broadcast group 8 to new broadcast address 9
8 rs485_bc_addr 9
```

8.4.4 rs485_bc_master <address>

This command enables the broadcast master mode on a camera with a given device address. If the command is transmitted as a get command (without an argument) it will reply with a flag (0 or 1) rather than the broadcast master address. The flag indicates whether this camera currently is the broadcast master or not (0 for no and 1 for yes).

| Value | Default | Minimal | Maximal |
|--|---------------------------|---------|---------|
| Device Address of the Broadcast Master | Broadcast Master Disabled | 0 | 99 |

Remarks:

1. Sending an address value of "-1" will disable the broadcast master mode.

Example 1:

```
// Disable broadcast master mode on all cameras in the group with the "0" broadcast address.
0 rs485_bc_master -1
```

Example 2:

```
// Sends a command to all cameras in the group (with the "0" broadcast address) and sets the camera
// with a "1"
// device address as the broadcast master, whilst disabling "broadcast master" on all other cameras.
0 rs485_bc_master 1
```

8.5 Settings Handling

8.5.1 save_settings <id>

Saves the current settings into the selected configuration in the camera non-volatile memory. There are 8 optional sets of settings which can be saved, with an id range of 0-7.

Remarks:

1. A **save_setteings** command with no parameters will save the first configuration and set it as the default.

Example:

```
save_settings 2
```

8.5.2 load_settings <id>

The **load_settings** command can be used to load saved settings on demand, which will overwrite the current camera settings. There are 8 optional sets of settings available, with an id range of 0-7. Only previously stored settings configuration will be applied and return an OK, while other commands will return a FAIL.

Remarks:

1. **load_settings** command with no parameters will load the first configuration.

Example:

```
load_settings 2
```

8.5.3 default_settings <id>

The **load_settings** command selects the default settings configuration which will be automatically loaded on camera startup. There are 8 optional sets of settings available, with an id range of 0-7. Reading value of “-1” will indicate that the factory settings are currently applied. **reset_settings** should be used to set factory as default settings.

Example:

```
default_settings 2
```

8.5.4 reset_settings

Resets the system into factory default run.

8.5.5 dump_settings

The **dump_settings** command dumps current settings configured on the device.

Remarks:

1. Some commands have no settings that can be dumped and thus do not show up when **dump_settings** is used.

8.6 System Commands

8.6.1 prompt <flag>

Set prompt mode of the command output.

| Flag | Function |
|-----------|-------------|
| 0 | No prompt |
| 1 (reset) | '=>' prompt |

8.6.2 reboot

Performs a system reboot (warm start). A full reboot may take several seconds.

8.6.3 fw_update

Set the system into firmware update mode. The firmware update file should be uploaded using XMODEM protocol via standard serial port interface. Please see [Appendix 1: Firmware Update](#) for a detailed explanation.

Remarks:

1. Only official firmware update file, provided by KAYA Instruments should be used to update camera. Other binary files may, or may not harm the camera and make it un-usable.

8.6.4 version

Dumps a detailed version information about the system with system ID and firmware version.

Examples:

Command: **version**

Response: **platform: IronSDI**
device name: Empty
system-id: 003F001E-30324703-37313437-FFFFFFF
hw revision: 00000101
system validity: LICENSED
feature mask HW: 0000003F
feature mask SW: 00000000
resolution mask: 00000000-001FFFFF-000FFCFF
loader version: 1 (0)
sw-release-id: V3.0
sw-release-date: 2021-01-01
sw-build-date: 2021-01-01
OK

8.6.5 name <name string>

The device name, which is shown in the output of the **version** command can be changed using **name** command. The chosen name must consist of maximum 32 characters with no spaces. The **save_settings** command must be used after a name change for it to take permanent effect.

Example:

```
name New_device_name_1
```

8.6.6 flip <mode>

Sets the image flip or rotation mode.

| Mode | Function |
|-----------|------------------|
| 0 (reset) | Normal (no flip) |
| 1 | Vertical flip |
| 2 | Horizontal flip |
| 3 | Rotated by 180° |

8.6.7 temp <sensor id>

Dumps camera temperature values in degree centigrade. The output has the format **temp <id> <value> <name>**

| Temperature Sensor ID | Function |
|-----------------------|---|
| 0 | Processor temperature (with 0.1°C accuracy) |
| 1 | Sensor temperature (with 0.1°C accuracy) |

Example 1:

```
Command: temp 0
Response: temp 0 43.3 Processor
OK
```

Example 2:

```
Command: temp 1
Response: temp 1 45.2 Sensor
OK
```

8.7 Camera Commands

8.7.1 cam_gain <gain>

Set the analog gain for the sensor. Gain can be read at any time, including when the auto exposure control is enabled.

| Value | Reset | Minimal | Maximal |
|-------|-------|---------------|--------------------|
| gain | 1000 | 1000 (1x 0dB) | 252000 (252x 48dB) |

Example:

```
// Example to set gain of 3x => input value should be 3000 = round (3 * 1000)
cam_gain 3000
```

8.7.2 cam_exposure <time>

Sets the exposure time/shutter width, in microseconds [μ sec], in which sensor is exposed to light.

Remarks:

1. The exposure time might be adjusted, if video mode is changed using **video_mode** command and the value exceeds the maximum allowed value for the new video mode.
2. Exposure can be read at any time, also when the auto exposure control is enabled.

Example:

```
//Example to set exposure of 10000usec
cam_exposure 10000
```

8.7.3 cam_info

Reports the min / max gain and exposure in the following order: minimum gain, maximum gain, minimum exposure, maximum exposure, gain normalized multiplication factor (1000 = 1x)

Example:

```
Command: cam_info
Response: cam_info 1000 252000 75 33333 1000
OK
```


8.7.4 identify

Provides essential system information including the platform, RS485 configuration and the device name. This command can be used to identify all devices which are connected to one RS485 bus by sending it to device address 100 (the fail-safe device address).

Each device will wait in turn before sending its ID string. The higher value the RS485 address of the device, the longer the device will wait before sending its stats: this ensures that the bus does not get corrupted.

Each device will report the following parameters:

- [RS485 ID](#), [RS485 Broadcast Address](#) and [RS485 Broadcast Master](#).
- [Device Name](#): name which can be set by the user.

Example:

```
Command: name New_device_name_1 // Set camera name (optional)
         100 identify // Identify
Response: id: IronSDI 1 0 0 New_device_name_1
         OK
```

8.8 Video Commands

8.8.1 video_mode <mode>

Sets the output video mode.

| Mode ID | Resolution | Frame Rate |
|---------|------------------|------------|
| 4 | 1920x1080p (FHD) | 30 |
| 5 | 1920x1080p (FHD) | 25 |
| 6 | 1920x1080p (FHD) | 24 |
| 7 | 1920x1080p (FHD) | 23.98 |
| 8 | 1920x1080p (FHD) | 29.97 |
| 9 | 1920x1080p (FHD) | 50 |
| 10 | 1920x1080p (FHD) | 60 |
| 11 | 1920x1080i (FHD) | 60 |
| 12 | 1920x1080i (FHD) | 50 |
| 13 | 1920x1080i (FHD) | 59.94 |
| 14 | 1920x1080p (FHD) | 59.94 |
| 15 | 2048x1080p (2K) | 30 |
| 16 | 2048x1080p (2K) | 25 |
| 17 | 2048x1080p (2K) | 24 |
| 18 | 2048x1080p (2K) | 23.98 |
| 19 | 2048x1080p (2K) | 29.97 |
| 20 | 2048x1080p (2K) | 50 |
| 21 | 2048x1080p (2K) | 60 |
| 22 | 2048x1080p (2K) | 59.94 |
| 23 | 2048x1080p (2K) | 48 |
| 24 | 2048x1080p (2K) | 47.96 |
| 25 | 3840x2160p (UHD) | 30 |
| 26 | 3840x2160p (UHD) | 25 |
| 27 | 3840x2160p (UHD) | 24 |
| 28 | 3840x2160p (UHD) | 23.98 |
| 29 | 3840x2160p (UHD) | 29.97 |
| 30 | 3840x2160p (UHD) | 50 |
| 31 | 3840x2160p (UHD) | 60 |
| 32 | 3840x2160p (UHD) | 59.94 |
| 33 | 3840x2160p (UHD) | 48 |
| 34 | 3840x2160p (UHD) | 47.96 |
| 35 | 4096x2160p (4K) | 30 |
| 36 | 4096x2160p (4K) | 25 |
| 37 | 4096x2160p (4K) | 24 |
| 38 | 4096x2160p (4K) | 23.98 |
| 39 | 4096x2160p (4K) | 29.97 |
| 40 | 4096x2160p (4K) | 50 |
| 41 | 4096x2160p (4K) | 60 |
| 42 | 4096x2160p (4K) | 59.94 |
| 43 | 4096x2160p (4K) | 48 |
| 44 | 4096x2160p (4K) | 47.96 |

8.8.2 downscale <channel=1> <downscale> <reserved=0>

The **downscale** command is used to output Full HD or 2K resolution from UHD or 4K downsampled source image. The downscale is only applied if the current video mode allows it and the camera support 4K resolutions.

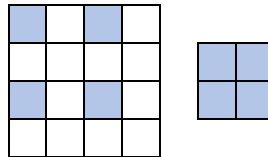


Figure 3 – Downscale example

The following table lists video modes which support the downscale.

| Mode ID | Mode |
|---------|------------------|
| 4 | 1920x1080p 30 |
| 5 | 1920x1080p 25 |
| 6 | 1920x1080p 24 |
| 7 | 1920x1080p 23.98 |
| 8 | 1920x1080p 29.97 |
| 9 | 1920x1080p 50 |
| 10 | 1920x1080p 60 |
| 11 | 1920x1080i 60 |
| 12 | 1920x1080i 50 |
| 13 | 1920x1080i 59.94 |
| 14 | 1920x1080p 59.94 |
| 15 | 2048x1080p 30 |
| 16 | 2048x1080p 25 |
| 17 | 2048x1080p 24 |
| 18 | 2048x1080p 23.98 |
| 19 | 2048x1080p 29.97 |
| 20 | 2048x1080p 50 |
| 21 | 2048x1080p 60 |
| 22 | 2048x1080p 59.94 |
| 23 | 2048x1080p 48 |
| 24 | 2048x1080p 47.96 |

Remarks:

1. <channel> input parameter should always be 1
2. <reserved> input parameter should always be 0
3. The downscale is only applied if the current video mode allows it and the camera support 4K resolutions.

Example:

downscale 1 1 0

8.8.3 sdi_black <offset>

Sets the black level for SDI in legal range mode (see [sdi_range](#)).

An offset value of 0 will result in SDI black value of 64 (SMTP conform). When changing this value, the black level on the SDI interface can be set different than 64 (not SMTP conform). This value changes the SDI range limiter and will stretch output values to adapt to the new range.

| Value | Reset | Minimal | Maximal |
|--------|-------|---------|---------|
| Offset | 0 | -60 | +60 |

8.8.4 sdi_white <offset>

Sets the white level for SDI in legal range mode (see [sdi_range](#)). An offset value of 0 will result in an SDI white value of 940 (SMTP conform). When changing this value, the white level on the SDI interface can be set different than 40 (not SMTP conform). This value changes the SDI range limiter and will stretch output values to adapt to the new range.

| Value | Reset | Minimal | Maximal |
|--------|-------|---------|---------|
| offset | 0 | -80 | +79 |

8.8.5 sdi_range <flag>

Sets the SDI output range type.

| Flag | Mode | Digital Code Range | Note |
|------|----------|---|--|
| 0 | Legal | Y ranges from 64 + sdi_black offset Y ranges from 940 + sdi_white offset U/V range from 64 to 960 | Used for broadcast and monitors with defined black and white levels. |
| 1 | Extended | Y/U/V range from 4 to 1019 | Used for recoding with maximum dynamic. |

Example:

```
// Enable the extended pixel range which depends on sdi_black and sdi_white values.
sdi_range 0
```

8.8.6 post_bright <offset>

Sets post processing brightness.

$Y_{out} = Y + offset$

Neutral value is 0.

| Value | Reset | Minimal | Maximal |
|--------|-------|---------|---------|
| offset | 0 | -128 | +127 |

8.8.7 post_cont <factor>

Sets post processing contrast.

$$Y_{out} = Y * factor / 128$$

Neutral value is 128.

| Value | Reset | Minimal | Maximal |
|--------|-------|---------|---------|
| factor | 128 | 0 | 255 |

Example:

```
// Set a contrast value of 1.23 => input value should be 157 = round (1.23 * 128.0)
post_cont 157
```

8.8.8 post_sat <factor>

Sets post processing color saturation.

$$Cb,r_{out} = Cb,r * factor / 128$$

Neutral value is 128.

| Value | Reset | Minimal | Maximal |
|--------|-------|---------|---------|
| factor | 128 | 0 | 255 |

Example:

```
// Set a saturation value of 1.23 => input value should be 157 = round (1.23 * 128.0)
post_sat 157
```

8.8.9 post_hue <offset>

Sets post processing color hue offset angle.

$$Cb' = Cb * \cos(\text{offset} * 90 / 128) + Cr * \sin(\text{offset} * 90 / 128)$$

$$Cr' = -Cb * \sin(\text{offset} * 90 / 128) + Cr * \cos(\text{offset} * 90 / 128)$$

Neutral value is 0.

| Value | Reset | Minimal | Maximal |
|-------|-------|-------------------|------------------|
| value | 0 | -128 (-90 degree) | 127 (+89 degree) |

Example:

```
// Set a hue value of 12.3 => input value should be 17 = round (12.3 * 128.0 / 90.0)
post_hue 17
```

8.8.10 wb

Triggers single shot white-balance.

Example:

wb

8.8.11 wb_threshold <threshold>

Limits the maximum threshold value when the white balance compensation algorithm counts. Assists with ignoring over saturated pixels in calculations.

| Value | Reset | Minimal | Maximal |
|-------|-------|---------|---------|
| value | 4000 | 0 | 4095 |

8.8.12 awb <flag>

Enable continuous white balance adjustment. Compensate sensor output colors to true colors. The algorithm works on the assumption that average color of image in selected ROI is gray.

| Enable | Function |
|--------|----------|
| 0 | disable |
| 1 | enable |

8.8.13 wb_preset <id>

Sets calibrated white balance presets (gains and color-cross matrices). List of calibrated presets are described in the following table:

| ID | Illumination | Color temperature |
|----|-----------------------|-------------------|
| 0 | disable preset | |
| 1 | horizon | 2200K |
| 2 | candle light (A) | 2700K |
| 3 | fluorescent (CWF) | 4000K |
| 4 | daylight sunny (D50) | 5000K |
| 5 | daylight (D65) | 6500K |
| 6 | daylight cloudy (D75) | 7500K |

8.8.14 gain_red <gain>

Sets gain factor for red component for selected output channel. This function is for basic color correction or white balance.

$$\text{red_out} = \text{red} * \text{gain} / 256$$

| Value | Reset | Minimal | Maximal |
|-------|-------|---------|---------|
| gain | 256 | 0 | 2047 |

Example:

```
// Set a red gain value of 1.23 => input value should be 315 = round (1.23 * 256.0)
gain_red 315
```

8.8.15 gain_blue <gain>

Same as gain_red but for blue color channel.

8.8.16 gain_green <gain>

Same as gain_red but for green color channel.

8.8.17 black_master <offset-red> <offset-green> <offset-blue>

Sets the black-level offset for red, green and blue-components for selected output channel. The processing is done in linear RGB domain (pre gamma).

$$\text{red_out} = (\text{red_in} - \text{offset-red}) * 4095 / (4095 - \text{offset-red})$$

| Value | Reset | Minimal | Maximal |
|--------|-------|---------|---------|
| offset | 0 | -2047 | 2048 |

Example 1:

```
// Set black-level for all components to 100
black_master 100
```

Example 2:

```
// Set black-level for red to 10, for green to 20 and blue to 30
black_master 10 20 30
```

8.8.18 black_red <offset>

Sets offset for red component for selected output channel as black level setting. The processing is done after de-bayering. The offset is defined as signed value. A value of zero is treated as neutral.

$$\text{red_out} = \text{red} + \text{offset}$$

| Value | Reset | Minimal | Maximal |
|--------|-------|---------|---------|
| offset | 0 | -4096 | 4095 |

8.8.19 black_blue <offset>

Same as black_red but for blue color channel.

8.8.20 black_green <offset>

Same as black_red but for green color channel.

8.8.21 flare <red level> <green level> <blue level>

Sets flare compensation level (= Defogging). The processing is done in the linear BAYER domain (pre de-bayering). The level is defined as unsigned value. A value of zero is treated as neutral (disable flare).

$$\text{red}_{\text{out}} = \frac{4095 * (\text{red}_{\text{in}} - \text{level} * -\text{red})}{4095 - \text{level} * -\text{red}}$$

| Value | Reset | Minimal | Maximal |
|--------|-------|---------|---------|
| offset | 0 | 0 | 65535 |

Example 1:

```
// Set flare value of 0.1 (10%) => input value should be 6554 = round (0.1 * 65536.0)
flare 6554 6554 6554
```

Example 2:

```
Set same flare level for all components:
flare 6554
```

8.8.22 color_cross <c0> .. <c8>

Sets the color cross talk matrix, which can be used for correction of cross talk effects and color space shifts. The cross talk compensation unit performs a regular RGB to R'G'B' color space conversion, to compensate the cross talk between color components of the image. The matrix coefficients provide the ability to correct each pixel value with the following matrix operation:

$$R' = (R * c_0 + G * c_1 + B * c_2) / 4096$$

$$G' = (R * c_3 + G * c_4 + B * c_5) / 4096$$

$$B' = (R * c_6 + G * c_7 + B * c_8) / 4096$$

| Value | Reset | Minimal | Maximal |
|----------------|-------|---------|---------|
| color_cross_c0 | 4096 | -32768 | 32767 |
| color_cross_c1 | 0 | -32768 | 32767 |
| color_cross_c2 | 0 | -32768 | 32767 |
| color_cross_c3 | 0 | -32768 | 32767 |
| color_cross_c4 | 4096 | -32768 | 32767 |
| color_cross_c5 | 0 | -32768 | 32767 |
| color_cross_c6 | 0 | -32768 | 32767 |

| | | | |
|----------------|------|--------|-------|
| color_cross_c7 | 0 | -32768 | 32767 |
| color_cross_c8 | 4096 | -32768 | 32767 |

Example:

```
// Set a color_cross_c2 value of 1.23 => input value should be 5038 = round (1.23 * 4096)
color_cross 4096 0 5038 0 4096 0 0 0 4096
```

8.8.23 color_cross_offset <red_offset> <green_offset> <blue_offset>

Sets the color cross talk offset. In addition to the matrix multiplication (see **color_cross**) an offset can be added to the pixel values for R, G and B separately. This offset is applied after the matrix multiplication. A value of zero is treated as no offset.

| Value | Reset | Minimal | Maximal |
|--------------|-------|---------|---------|
| red_offset | 0 | -2048 | 2047 |
| green_offset | 0 | -2048 | 2047 |
| blue_offset | 0 | -2048 | 2047 |

8.8.24 color_conv <c0> .. <c8>

Sets the color conversion matrix. The following formula is used for the conversion:

$$Y = (c0 * R + c1 * G + c2 * B) / 4096 + 64$$

$$Cb = (c3 * R + c4 * G + c5 * B) / 4096 + 512$$

$$Cr = (c6 * R + c7 * G + c8 * B) / 4096 + 512$$

| Value | Reset | Minimal | Maximal |
|---------------|-------|---------|---------|
| color_conv_c0 | 871 | -8192 | 8191 |
| color_conv_c1 | 2929 | -8192 | 8191 |
| color_conv_c2 | 296 | -8192 | 8191 |
| color_conv_c3 | -469 | -8192 | 8191 |
| color_conv_c4 | -1579 | -8192 | 8191 |
| color_conv_c5 | 2048 | -8192 | 8191 |
| color_conv_c6 | 2048 | -8192 | 8191 |
| color_conv_c7 | -1860 | -8192 | 8191 |
| color_conv_c8 | -188 | -8192 | 8191 |

Example:

```
// Set a color_cross_c2 value of 1.23 => input value should be 5038 = round (1.23 * 4096)
color_conv 871 2929 5038 -469 -1579 2048 2048 -1860 -188
```

8.8.25 color_space <color space>

Change the color space matrix by updating the **color_conv** matrix coefficients according to standard RGB to YCbCr conversion method.

| Value | Color Space | Usually used for |
|-----------|-------------|------------------|
| 0 (reset) | Rec.709 | HD / SDR |
| 1 | Rec.2020 | UHD / HDR |

8.8.26 stat_roi <width><height><offsetX><offsetY>

Defines the ROI for auto compensation algorithms such as auto exposure, auto gain and white balance. “ROI definition” refers to Region of Interest which will be used for brightness calculations. The ROI OffsetX and OffsetY refer to the distance of the ROI from top left corner of the output image area:

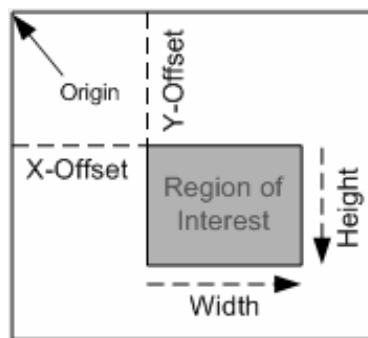


Figure 4 – ROI position in relation to the origin

By default, ROI defined to be the maximum possible area: i.e. the horizontal and vertical offsets are 0 and the ROI width and height as the output image size.

8.8.27 stat_roi_info <maxWidth><maxHeight><widthStep><heightStep>

Reports the maximum allowed values for **stat_roi** command parameters. The values depend on the selected video mode

8.8.28 cam_roi_offset <offsetX><offsetY>

While the image resolution is defined by the selected video mode (see the **video_mode** command), the **cam_roi_offset** provides an option to move absolute image ROI position across the sensor. The ROI offset is in the center of the sensor by default, but can be modified in case a digital ROI compensation is required as an alternative to physical camera shift.

The maximal allowed compensation varies according to the maximal sensor size and selected video mode: these values can be acquired using **cam_roi_offset_info** command.

8.8.29 cam_roi_offset_info <offsetXMax><offsetYMax><offsetXStep><offsetYStep>

Reports the maximum allowed values for **cam_roi_offset** parameters

8.9 Defect Pixel Correction

The defected pixel correction will correct up to 32 pixels in the sensor and up to 2 adjacent pixels in a row. The pixel correction coordinates represent pixels of sensor's visible ROI, therefore identifying the correct X and Y coordinate should be done using default, full resolution image. The algorithm will correct the defect pixel based on the value of existing adjacent pixels. The correction for Mono and Color sensor is slightly different and described as follows:

The defect pixel $P(x, y)$ value will be the average value of two pixels from both sides of pixel $P(x, y)$ in the same row, corresponding to the same Bayer color element.

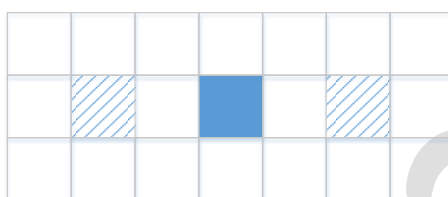


Figure 5 – Defect pixel correction position

8.9.1 dpc <flag>

Enable defect pixel correction.

| Flag | Function |
|-----------|--------------------|
| 0 | No correction |
| 1 (reset) | Correction enabled |

8.9.2 dpc_add_pixel <x> <y>

Adds a pixel coordinate to defect pixel table or dumps the whole table. Valid values for the x and y coordinates.

| Value | Minimal | Maximal |
|--------------|---------|------------------|
| X coordinate | 0 | <max_width> - 1 |
| Y coordinate | 0 | <max_height> - 1 |

Example 1:

```
// Add pixel coordinate (123, 456) into defect pixel table
dpc_add_pixel 123 456
```

Example 2:

```
// Dump dead pixel table
Command: dpc_add_pixel
Response: dpc_add_pixel 100 100
          dpc_add_pixel 200 200
          OK
```

8.9.3 dpc_del_pixel

Deletes all pixel coordinates from the defect pixel table.

8.9.4 dpc_save

Saves defect pixel table to persistent memory.

8.9.5 dpc_load

Loads defect pixel table from persistent memory.

8.10 Knee Function

8.10.1 knee <flag> <knee_point> <knee_slope> <white_clip>

Knee function for highlight control. 3 parameters can be set for the knee function:

| Value | Reset | Minimal | Maximal |
|----------------|---------|---------|---------|
| flag (enable) | 0 (off) | 0 | 1 |
| knee_point [%] | 85 | 1 | 100 |
| knee_slope | 140 | 100 | 1600 |
| white_clip [%] | 109 | 100 | 109 |

Example:

```
// Enable knee at 85% with a slope of 1.4 and white clip at 109%
knee 1 85 140 109
```

8.11 Loop-up Table Management

The LUT Control can be used to re-map the image linear output in different manner. Mostly to compensate for the non-linear scene emission. Each index at the LUT corresponds to the original pixel value, and the LUT value at this index corresponds to the new value that the pixels should be replaced with.

Pixel value is replaced according to the following equation:

$$\begin{aligned} \overline{P_{\text{red}}(x,y)} &= \text{LUT}_{\text{red}}[P_{\text{red}}(x,y)] \\ \overline{P_{\text{green}}(x,y)} &= \text{LUT}_{\text{green}}[P_{\text{green}}(x,y)] \\ \overline{P_{\text{blue}}(x,y)} &= \text{LUT}_{\text{blue}}[P_{\text{blue}}(x,y)] \end{aligned}$$

Where $P(x,y)$ is the pixel at offset X in horizontal and Y in vertical, of specific color.

Lookup table can be set in one of the following methods:

1. LUTs can be programmed using up to 48 sample points. All intermediate values are calculated with a spline interpolator (see **lut_sample_<channel>** commands).
2. LUTs can be programmed by specifying a set of parameters which will be used to calculate and apply a gamma curve according to REC.709 standard (see **lut_fun_rec709** command).
3. LUTs can be fast configured by specifying only the desired gamma value of the REC.709 gamma function (see **lut_fast_gamma** command).
4. LUTs can be configured with a fixed configuration of three presets which include the default REC.709 gamma curve and two HDR gamma curves: PQ and HLG (see **lut_fixed_mode** command).

8.11.1 lut_enable <reserved=0> <flag>

Enables the function for look up table.

| Flag | Function |
|-----------|------------------|
| 0 | Disable (linear) |
| 1 (reset) | Enable |

Remarks:

1. <reserved> input parameter should always be 0

Example:

```
// Apply LUT table configuration.
lut_enable 0 1
```

8.11.2 lut_mode <mode>

Selects the LUT operational mode which are described in the following table:

| Mode | Description |
|------|---|
| 0 | Table based using interpolation. The user has to specify a table with a maximum of 48 values. Intended for offline calibration. |
| 1 | Fast gamma mode where the user only specifies the desired gamma value. Can be used for gamma changes during runtime. |
| 2 | Fixed gamma mode where the user selects one of three fixed gamma tables. |

8.11.3 lut_preset <index>

Selects the preset storage for the current LUT interpolator. There are up to 5 presets which can be modified.

| Index | Factory preset |
|-------|----------------|
| 0 | Linear |
| 1 | Linear |
| 2 | Linear |
| 3 | Linear |
| 4 | Linear |

Example:

```
// Select lut preset 1
lut_preset 1
```

8.11.4 lut_sample_master <xi_0> <yi_0> ... <xi_7> <yi_7>

Defines the sample points in a lookup table for all colors. The x-value is the input value, the y-value is output.

| Value | Minimal | Maximal |
|-------|---------|---------|
| xi_n | 0 | 4095 |
| Yi_n | 0 | 4095 |

Sets up to 8 sample points at a time, up to a total of 48 for the look up definition for all colors. For setting more than 8 points, the function can be called multiple times. In case a x-position is set twice, the previous value is overwritten.

Remarks:

1. When more than 48 points are defined, an error message is dumped.
2. lut_sample_<color> command should be used to read back the current sample point. lut_sample_master will not return sample data.

Example:

```
// Add LUT sample points to preset 1 and start curve interpolation.
// Set following points: (0, 200), (4095, 800), (300, 500), (700, 600). This will result in new image range
// from 200 (index 0 and up) to 800 (index 4095 and down)

lut_enable 0 0 // disable lut
lut_mode 0 // select lut mode 0 (Table based using interpolation)
lut_preset 1 // select preset 1 to modify
lut_reset // reset preset 1
lut_sample_master 0 200 4095 800 // up to 8 pairs of point
lut_sample_master 300 500 700 600 // up to 8 pairs of point
lut_interpolate // interpolate lut table, fill in values between the specified samples
lut_enable 0 1 // enable lut
```

8.11.5 lut_sample_red <xi_0> <yi_0> ... <xi_7> <yi_7>

Same as [lut_sample_master](#), but only for red component.

8.11.6 lut_sample_green <xi_0> <yi_0> ... <xi_7> <yi_7>

Same as [lut_sample_master](#), but only for green component.

8.11.7 lut_sample_blue <xi_0> <yi_0> ... <xi_7> <yi_7>

Same as [lut_sample_master](#), but only for red component.

8.11.8 lut_interpolate

Interpolates all look up tables based on the given sample points - fill in lut values in between the specified samples according to a spline interpolator.

8.11.9 lut_interpolate_red

Interpolates the red look up table based on the given sample points.

8.11.10 lut_interpolate_green

Interpolates the red look up table based on the given sample points.

8.11.11 lut_interpolate_blue

Interpolates the red look up table based on the given sample points.

8.11.12 lut_reset_master

Clears all color channels look up sample points.

8.11.13 lut_reset_red

Clears all red look up sample points.

8.11.14 lut_reset_green

Clears all green look up sample points.

8.11.15 lut_reset_blue

Clears all blue look up sample points.

8.11.16 lut_fun_rec709 <threshold> <linear-contrast> <linear-brightness> <contrast> <gamma> <brightness>

Sets new LUT sample points according to REC.709 for selected LUT preset. Sets the gamma curve for all color channels.

| Value | Minimal | Maximal | REC.709 |
|------------------------|---------|---------|---------|
| Threshold | 0 | 1000 | 18 |
| Contrast (linear) | 0 | 20000 | 4500 |
| Brightness (linear) | -1000 | 1000 | 0 |
| Contrast (non-linear) | 0 | 10000 | 1099 |
| Gamma (non-linear) | 1100 | 3000 | 2222 |
| Brightness (nonlinear) | -1000 | 1000 | -99 |

The following image shows the normalized REC.709 gamma curve and it's transition from linear to non-linear (power function) part. The linear part ranges from 0 to <0.018 and is computed by the following formula:

$$V_{out} = 4.5 * V_{in}$$

The maximum value in this range is:

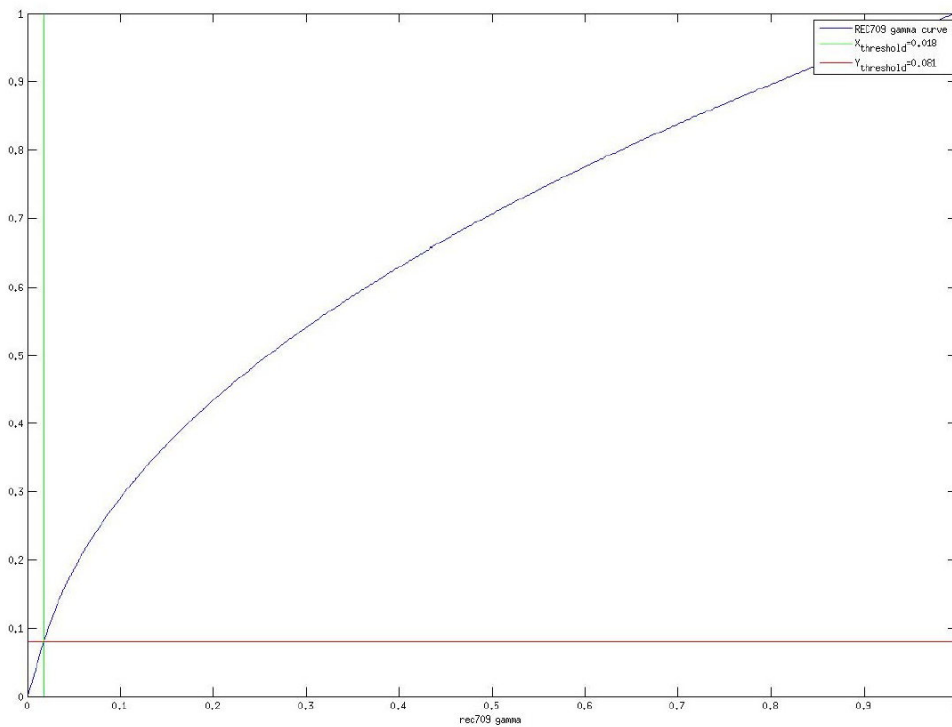
$$V_{out,min} = \lim_{V_{in} \rightarrow 0.018} (4.5 * V_{in}) = 4.5 * 0.018 = 0.081$$

The non-linear part ranges from 0.018 to 1 and is computed by the following formula:

$$V_{out} = 1.099 * V_{in}^{(1/2.222)} - 0.099$$

The minimum value in this range is:

$$V_{out,min} = 1.099 * 0.018^{(1/2.222)} - 0.099 = 0.081$$



To avoid a discontinuity in the gamma-function select the brightness (offset) for the non-linear part by the following formula:

$$V_{\text{out,linear max}} \approx V_{\text{out,linear min}} \text{brightness}_{\text{non-linear}} \approx (\text{Contrast}_{\text{linear}} * \text{threshold} + \text{Brightness}_{\text{linear}}) - \text{Contrast}_{\text{non-linear}} * \text{threshold}^{\text{Gamma}}$$

Remarks:

1. The values are normalized to a range from 0.0 to 1.0 and multiplied by a scaling coefficient of 1000.

Example:

```
// Compute LUT sample points according to REC.709 and store in preset 1, then start curve interpolation
```

```
// <threshold>      18 => 0.018 * 1000
// <linear-contrast> 4500 => 4.5 * 1000
// <linear-brightness> 0 => 0 * 1000
// <contrast>       1099 => 1.099 * 1000
// <gamma>          2222 => 2.222 * 1000
// <brightness>     -99 => -0.099 * 1000
```

```
lut_enable 0 0 // disable lut
lut_mode 0 // select lut mode 0 (Table based using interpolation)
lut_preset 1 // select preset 1 to modify
lut_fun_rec709 18 4500 0 1099 2222 -99
lut_interpolate // interpolate lut table – fill in values between specified samples
lut_enable 0 1 // enable lut
```

8.11.17 lut_fast_gamma <gamma>

The fast gamma function uses the same formula as the lut_fun_rec709 command, but the user only has to specify the desired gamma value. All other values are fixed or computed as needed.

| Value | Minimal | Maximal | REC.709 |
|-------|---------|---------|---------|
| Gamma | 1100 | 3000 | 2222 |

Remarks:

1. The value is normalized to a range from 0.0 to 1.0 and multiplied by a scaling coefficient of 1000.
2. Setting a value of 2222 will result in the default REC.709 gamma curve.

Example:

```
lut_mode 1           // select lut mode 1 (Fast gamma mode)
lut_enable 0 1      // make sure lut is enabled
lut_fast_gamma 2222
```

8.11.18 lut_fixed_mode <mode>

The fixed gamma mode contains three presets which are shown in the table below:

| Mode | Description |
|-------------|--|
| 0 (default) | REC.709 gamma curve |
| 1 | PQ gamma curve specified in ITU-R BT.2100 which can be used for HDR content |
| 2 | HLG gamma curve specified in ITU-R BT.2100 which can be used for HDR content |

Examples:

```
lut_mode 2           // select lut mode 2 (Fixed gamma mode)
lut_enable 0 1      // make sure lut is enabled
lut_fixed_mode 0    // select fixed lut mode 0 (REC.709 gamma curve)
```

8.12 Image statistic commands

8.12.1 stat_rgb

Dumps the average image value for each color channel (RGB). Ranges from 0 to 4095.

Examples:

```
Command: stat_rgb
Response: stat_rgb 1040 896 1200
```

8.13 Auto Exposure

8.13.1 aec <enable> <setPoint> <speed> <clmTolerance> <activeGain> <activeExposure> <activeApt> <maxExposure> <maxGain> <reserved>

Auto Exposure and Gain are used to control the picture brightness by adjusting Exposure and Gain values automatically in order to reach desired luminance level. The algorithm calculates the average picture intensiveness inside the defined ROI and tries to adjust it to desired luminance level, by changing the exposure time and/or analog gain level.

The target luminance is set by the setPoint, this will determine how bright the image looks. A higher setPoint results in a brighter exposed image.

The speed determines how fast the exposure control reacts to luminescence changes. A higher value means a fast reaction to changes.

The clmTolerance sets the effective ratio of average luminescence level as complement to peak luminescence level, whilst the activeGain enables the auto gain algorithm and the activeExposure enables the auto exposure algorithm.

The maxGain parameter can be used to change the maximum sensor gain used by the algorithm. Notice the high maxGain may result in grainy saturate image.

| Parameter | Description | Minimal | Maximal | Default | Remarks |
|----------------|---|---------|---------|---------|---|
| enable | Auto brightness compensation | 0 | 1 | 0 | 0: Disable 1: Enable |
| setPoint | Target luminance | 0 | 4095 | 2048 | |
| speed | Control speed | 0 | 1000 | 250 | 0: No filter is applied on luminance update 1000: Maximum low pass filter is applied on luminance update |
| clmTolerance | Impact of average and peak value on setPoint calculation | 0 | 1000 | 600 | 0: Compare SetPoint only with peak value 1000: Compare SetPoint only with average value Other: Compare SetPoint with combination of peak and average values |
| activeGain | Active auto analog gain | 0 | 1 | 1 | |
| activeExposure | Active auto exposure | 0 | 1 | 1 | |
| activeApt | Active auto aperture | | | | For future use |
| maxExposure | Maximum sensor exposure which is used by the auto exposure control [μ s] | | | | *See can_info command for valid value range |
| maxGain | Maximum sensor gain which is used by the auto exposure control | 1000 | 252000 | 252000 | Default: x100 *See can_info command for valid value range |
| reserved | | | | | For future use |

Example 1:

```
// Disable aec completely  
aec 0
```

Example 2:

```
// Enable aec with luminance 1234. Remain other parameters with same value  
aec 1 1234
```

Example 3:

```
// Enable aec with luminance 1234. Set speed at 25% => input value be 250 = round (0.25 * 1000). Set  
calculation tolerance at 90% => input value should be 900 = round (0.9 * 1000). All other parameters  
remain with the same value  
aec 1 1234 8
```

Example 4:

```
// Activate auto exposure AND auto gain. Limit gain to 12.3x => input value should be 12300 = round  
(12.3 * 1000). Limit exposure to 20000  $\mu$ s (microseconst)  
aec 1 1234 250 600 1 1 0 20000 12300 0
```

8.14 Genlock

8.14.1 genlock <mode>

The genlock mechanism is used to synchronize multiple cameras video signals.

| Mode | Description |
|-------------|---|
| 0 (default) | Disabled – the genlock is off and the camera will work in free-run mode |
| 1 | <p>Input Slave Auto – automatic genlock signal synchronization. When the genlock input cable is connected the camera, it will try to lock on the external tri-level sync signal, otherwise the camera will run in free-run mode. On lol (loss-of-lock) event the camera will return to free-run mode after specified timeout (see genlock_lol_filter command). Image may flicker or even disappear while camera tries to lock on external tri-level sync signal.</p> <p>The configured video mode (see video_mode command) must match the input sync signal for successful genlock operation.</p> |
| 2 | <p>Input Slave Force – genlock is always active and try to sync on external tri-level sync signal. Camera will not output any image unless genlock is synchronized.</p> <p>The configured video mode (see video_mode command) must match the input sync signal for successful genlock operation.</p> |

8.14.2 genlock_status

This command is used to check the current genlock status.

| Status | Description |
|--------|---|
| 0 | Genlock is disabled or genlock is enabled and successfully synchronized on external tri-level sync signal. |
| 1 | Genlock is enabled and failed to synchronize on external tri-level sync signal. This status indicates that either there is no input signal, the reference input signal doesn't match the configured video mode or the signal is unstable. |

Example:

Command: **genlock_status**

Response: **genlock_status 0**

8.14.3 genlock_lol_filter <time_ms>

The loss-of-link (lol) timeout can be configured to prevent glitches in the unstable genlock signal. A loss-of-link is only registered when an unstable link is detected for more than a specified lol timeout, in which case the connection will be reset and an attempt will be made to synchronize the signal again. If a camera re-synchronizes often, it is advised to increase the lol timeout.

| Value | Reset | Minimal | Maximal |
|---------|-------|---------|---------|
| time_ms | 1000 | 0 | 10000 |

8.14.4 genlock_offset <v_offset> <h_offset>

The vertical and horizontal position offset that is added to the reference sync signal and can be adjusted.

| Value | Minimal | Maximal |
|----------|---------------------------------|-------------------------------|
| v_offset | - v_offset_phase ^[1] | v_offset_phase ^[1] |
| h_offset | - h_offset_phase ^[1] | h_offset_phase ^[1] |

Remarks:

1. Minimal and maximal values are symmetric and can be extracted using the [genlock_offset_info](#) command.

8.14.5 genlock_offset_info <v_offset_phase><h_offset_phase>

The phase of vertical and horizontal position offset that can be added to the reference sync signal. Maximum value is +offset_phase and minimum value is - offset_phase for each of the settings.

8.14.6 genlock_term <enable>

Modifies the genlock input/output 75R termination.

| Mode | Description |
|------|-----------------|
| 0 | Termination off |
| 1 | Termination on |

8.14.7 genlock_crosslock <vmode>

When camera genlock crosslock is disabled, the tri-level sync signal should match the current video mode. The genlock could lock on other input signal which can be set using the crosslock configuration. In general genlock can work with an input signal that is half, same or double of the video format rate, and also interlaced and progressive signals are interchangeable.

| VMode | Genlock Tri-level sync input signal |
|-------|-------------------------------------|
| 0 | Disabled (same as video mode) |
| 4 | 1080p30 |
| 5 | 1080p25 |
| 6 | 1080p24 |
| 7 | 1080p23.98 |
| 8 | 1080p29.97 |
| 9 | 1080p50 |
| 10 | 1080p60 |
| 11 | 1080i60 |
| 12 | 1080i50 |
| 13 | 1080i59.94 |
| 14 | 1080p59.94 |

8.15 SDI Time Code

8.15.1 timecode <hour> <minute> <second>

Sets SDI time code which is embedded into the SDI output. The time code can be set manually.

Example 1:

Command: **timecode** // get current system time

Response: **timecode 0 57 24**

OK

Example 2:

Command: **timecode 1 1 1** // set sdi time code to 1h 1m 1s

Response: **OK** // system acknowledgement

8.15.2 timecode_hold <flag>

This command can be used to hold the timecode on the SDI output, and can be used to trigger recording in an external recorder.

| Flag | Timecode Hold |
|------|---|
| 0 | Normal operation, SDI timecode is running |
| 1 | On hold, SDI timecode is unchanged until hold is released |

Remarks:

1. The the timecode will continue to run in the background: as soon as hold is released the SDI timecode will jump to the current time value.

9 Hardware Reference

9.1 Status LED

All SDI camera are equipped with a bi-color LED. Color coded indications are shown in the following table:




| LED state | Indication |
|--|--|
|  Fast flash green | Camera is connected, data is being transferred |
|  Solid green | Camera is connected, no data being transferred or settings are being updated |
|  Solid red | Internal error occurred / BIT failed |

Table 4 – Connector indicator lamp states

9.2 Micro BNC Connector

To connect the Micro-BNC cable, first need to align the pin on the male end with the “L” shaped track on the female-connector of the Coaxial cable. Once aligned, the connector should be pushed in place (see figure no. 1). Only mild pressure should be applied to achieve this operation, otherwise it may cause unnecessary damage to the cable or the card.

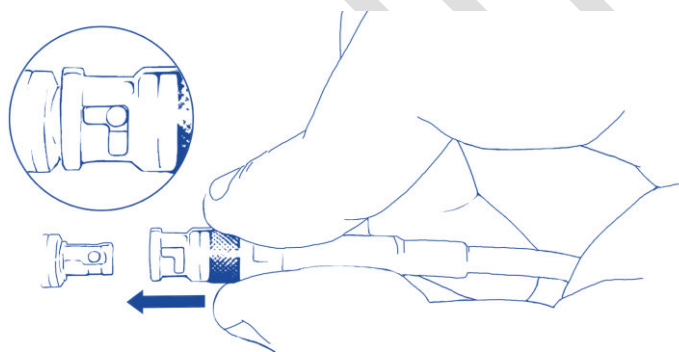


Figure 6 – Pushing the Micro-BNC connector into place

Once pushed all the way through, twist the connector clock-wise. The pin will move in the track locking the connector in position:

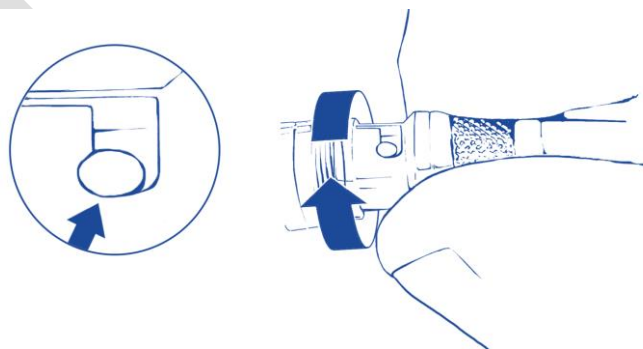


Figure 7 – Twisting the connector and securing it in position

The cable can be removed by reversing the steps: twisting the connector counter-clock-wise and pulling it out. Do not force the cable out! In case of resistance check for the pin location in relation to the track. Adjust as needed and only then pull the cable out.

10 Appendix 1: Firmware Update

Please note that the firmware can also be updated via the SDIControlPoint software that offers a user friendly GUI. For more information please refer to section [7.1 SDIControlPoint](#).

The Iron SDI Camera supports firmware update via serial terminal over RS232 protocol. To initiate a firmware update, follow the next steps:

5. Download the latest firmware from KAYA's website.
6. Please make sure to download the correct FW for the camera that you have, as installing an incorrect FW may cause the camera to malfunction.
7. Open serial emulated terminal (usage of Tera Term terminal is recommended) and set serial communication protocol to baud rate 115200, 8bit data, 1bit start, 1bit stop and no flow control. For example, in the Tera Term terminal, this should look as following (the port number might be different):

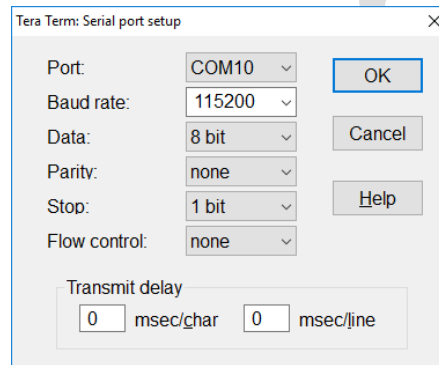


Figure 8 – Serial communication example

8. Choose the firmware update option by entering “fw_update”.
9. Under the “File” tab use the terminal “transfer” capability using the XMODEM protocol to initiate the firmware update. Choose “Send” and the firmware update file.

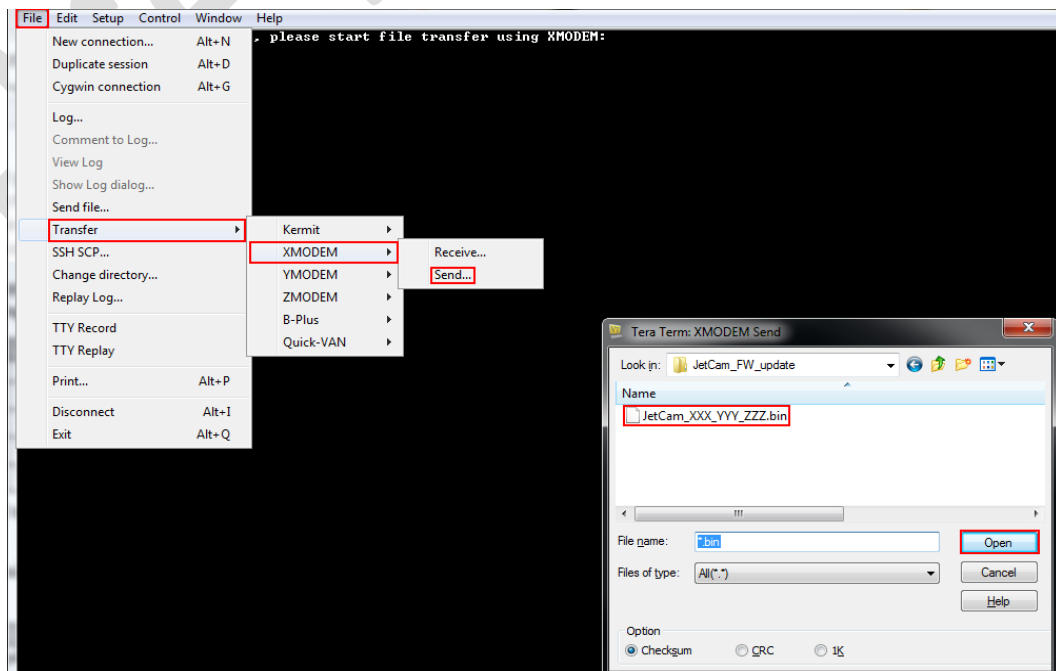


Figure 9 – Firmware terminal initiation

10. If no firmware will be sent during 1 minute, or in case of an error, the firmware update will fail and return to the previous operation mode.

```
IRON:> firmware
Now starting firmware update, please start file transfer using XMODEM:
The Firmware Update was UNSUCCESSFUL. Error: -1
IRON:>
```

Figure 10 – Firmware update fail

11. The firmware update process will take about 10 minutes.

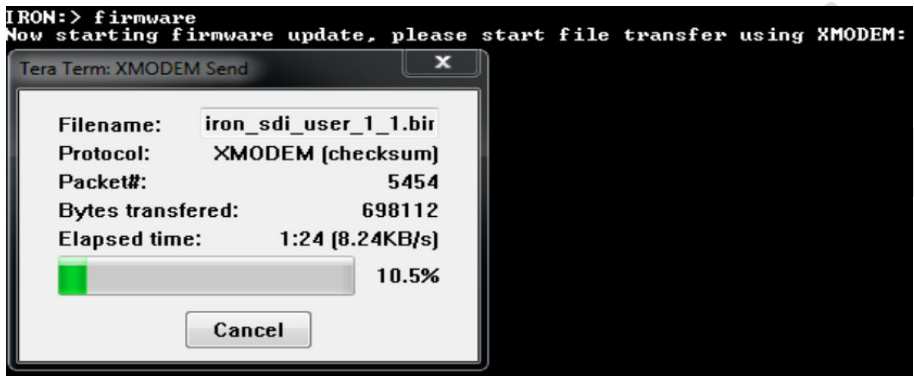


Figure 11 – Firmware update process

12. A successful update will result in an appropriate message

```
IRON:> firmware
Now starting firmware update, please start file transfer using XMODEM:
The Firmware Update was SUCCESSFUL file size: 6653440
IRON:>
```

Figure 12 – Firmware update succession

To apply the new firmware, the camera should be power cycled by disconnecting the power supply from the camera and connecting it back after a few seconds. After camera has been powered, it should be kept at least ~30 seconds with power before new firmware will be applied.

International Distributors



Sky Blue Microsystems GmbH
Geisenhausenerstr. 18
81379 Munich, Germany
+49 89 780 2970, info@skyblue.de
www.skyblue.de



In Great Britain:
Zerif Technologies Ltd.
Winnington House, 2 Woodberry Grove
Finchley, London N12 0DR
+44 115 855 7883, info@zerif.co.uk
www.zerif.co.uk