ELIIXA+ Family

CMOS Multi-Line Camera

User Manual

ELIIXA+ 8K/4K CL COLOR

Table of Contents

1 CAMERA OVERVIEW

1.1 Features

- Dual Line Cmos Colour Sensor :
  - 8192 RGB Pixels, 5 x 5µm (Full Definition)
  - 4096 RGB Pixels 10x10µm (True Colour)
- Interface : CameraLink® (up to 10 Taps at 85MHz)
- Line Rate :
  - Up to 50000 l/s In 8k Full Definition Mode
  - Up to 66000 l/s in 4k True Colour Mode
- Bit Depth : 24bits (RGB 8bits)
- Scan Direction
- Flat Field Correction
- Automatic White Balance
- Color Matrix Correction on "BHx" Models
- Interpolation algorithms embedded to remove colour artefacts
- Low Power Consumption : <9W
- F-Mount compliance

1.2 Key Specifications

Characteristics	Typical Value		Unit
Sensor Characteristics at Maximum Pixel Rate			
Resolution	8192	4096	RGB Pixels
pixel size (square)	5	10	µm
Max Line Rate	50	66	kHz
Radiometric Performance at Maximum Pixel Rate and minimum camera gain			
Bit depth	3 x 8		Bits
Response non linearity	< 1		%
PRNU HF Max	3		%
Dynamic range	65		dB
Peak Response (All Modes)			
Red	11.8		LSB 8bits/(nJ/cm²)
Green	11.2		LSB 8bits/(nJ/cm²)
Blue	7.8		LSB 8bits/(nJ/cm²)

Functionality (Programmable via GenICam Control Interface)		
Analog Gain	Up to 12 (x4)	dB
Offset	-4096 to +4096	LSB
Trigger Mode	Timed (Free run) and triggered (Ext Trig, Ext ITC) modes	
Sensor Modes	• True Color Enhanced : 4096 RGB Pixels of 10x10µm • True Color Single : 4096 RGB Pixels of 10x10µm • Full Definition Enhanced : 8192 RGB Pixels 5x5µm • Full Definition Single : 8192 RGB Pixels 5x5µm	
Mechanical and Electrical Interface		
Size (w x h x l)	126 x 60 x 35	mm
Weight	360	g
Lens Mounts	F, T2, M42	-
Sensor alignment	±100	µm
Sensor flatness	±35	µm
Power supply	12 - 24	V
Power dissipation	< 9	W
General Features		
Operating temperature	0 to 55 (front face) or 70 (Internal)	°C
Storage temperature	-40 to 70	°C
Regulatory	CE, FCC and RoHS compliant	

1.3 Description

e2v's next generation of line scan cameras are setting new, high standards for line rate and image quality. Thanks to e2v's recently developed multi-line CMOS technology, the camera provides an unmatched 100,000 lines/s and combines high response with an extremely low noise level; this delivers high signal to noise ratio even when short integration times are required or when illumination is limited. The 5µm pixel size is arranged in four active lines and dual line filter configuration allowing the camera to be operated in several modes: True colour mode with 10µm RGB pixels to provide equivalent colour fidelity to 10µm pixel tri-linear solutions with advanced immunity to web variation or Full definition mode with a 8192 RGB pixel resolution.

1.4 Typical Applications

- Raw material surface inspection
- Flat panel display inspection
- PCB inspection
- Solar cell inspection
- Parcel and postal sorting
- High resolution document scanning
- Print and paper inspection

1.5 Models

Part Number	Definition / Max Speed	True Color / TC Enhanced	Full Definition / FD Enhanced	New Sensor Generation	RGB Matrix
EV71YC4CCL8005-BA0	8k/50kHz – 4k/66kHz	Yes	Yes	With new Release	No
EV71YC4CCL8005-BH0	8k/50kHz	No	Yes	Yes	Yes
EV71YC4CCL4010-BH0	4k/66kHz	Yes	No	Yes	Yes

- The Sensor Modes "True Color", "True Color Enhanced", Full Definition", "Full Definition Enhanced" are detailed hereafter
- New Sensor version is embedded in EV71YC4CCL8005-BA0 with release version 2.0.0

2 CAMERA PERFORMANCES

2.1 Camera Characterization

	Unit	True Color (8k)		Full Definition Single		Full Definition Enhanced	
		Typ.	Max	Typ.	Max	Typ.	Max
Dark Noise RMS	LSB	0.12	1.2	0.11	1.2	0.12	1.2
Dynamic Range	-	2125:1	-	2125:1	-	2125:1	-
RMS Noise (3/4 Sat)	LSB	2.2	-	2.15	4	2.2	4
Full Well Capacity	e- (per color)	13650	-	13650	-	13650	-
SNR (3/4 Sat)	dB	40	-	40	-	40	-
Peak Response (460/530/660nm)	LSB 8bits/ (nJ/cm2)	8/10/12	-	4/5/6	-	8/10/12	-
Non Linearity	%	0,3	-	0,3	-	0,3	-
Without Flat Field Correction :							
FPN rms	LSB	0.21	1	0.23	1	0.22	1
FPN pk-pk	LSB	1	2	1	2	1	2
PRNU hf (3/4 Sat)	%	0.13	0,35	0.123	0,35	0.14	0,35
PRNU pk-pk (3/4 Sat)	%	1.1	3	1	3	1.25	3

Test conditions :

- All values are given at Nominal Gain (0dB) : Preamp Gain x1, Amp Gain 0dB
- Figures in LSB are for a 8bits format
- Measured at exposure time = 400µs and line period = 400µs in Ext Trig Mode (Max Exposure Time)
- Maximum data rate

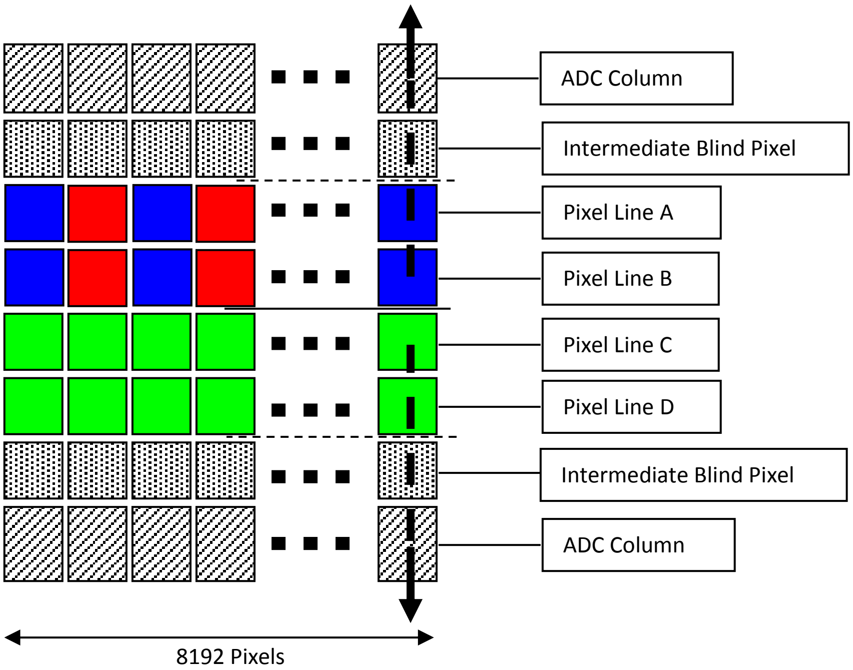
2.2 Image Sensor and color modes

The Eliixa+ Colour 8k sensor is composed of two pairs of sensitive lines.

The Colour version has been completed with RGB colour Filter and disposed as detailed beside.

Each pair of lines uses the same Analog to Digital Column converter (ADC Column). An appropriate (embedded) Time delay in the exposure between each line this allows to combine two successive exposures in order to double the sensitivity of a single line.

This Time Delay Exposure is used only in the Full Definition Enhanced mode (See Below).



2.2.1 Full Definition Single Mode (FDS)

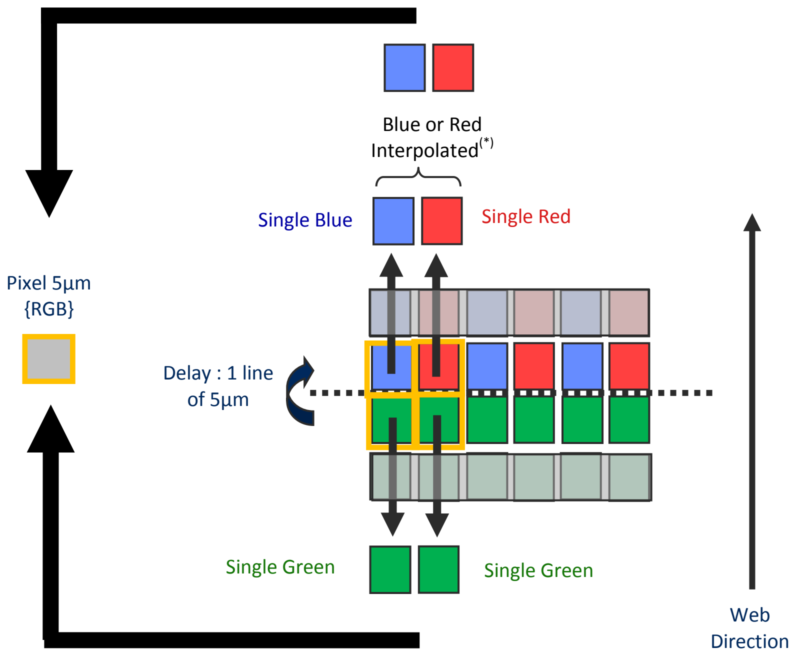
5µm Pixels (R,G,B)

Same definition than B&W

Requires x3 the data flow of the B&W

- Sensitivity is half of the TC mode available : Equivalent to 3 x Pixels of 5µm (with their respective colour filters).

- "Full Exposure control" not needed in this mode as the Time Delay Exposure is not active. The Exposure time can be control as for a single line mode.



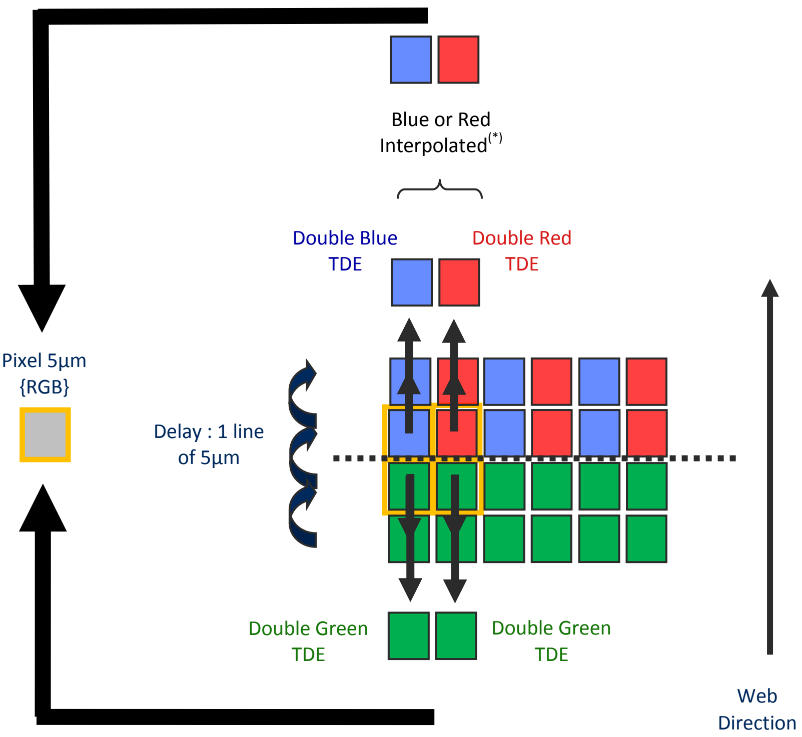
2.2.2 Full Definition Enhanced Mode (FDE)

5µm Pixels (R,G,B)

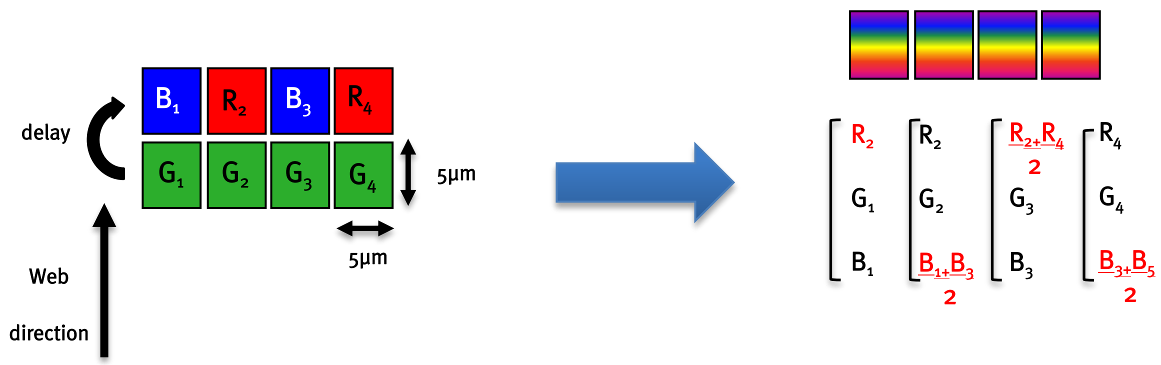
Same definition than B&W

Requires x3 the data flow of the B&W

- Sensitivity is the same as the TC mode available : Equivalent to 6 x Pixels of 5µm (with their respective colour filters).
- "Full Exposure control" is activated in this mode as the Time Delay Exposure is active.



2.2.3 Color Interpolation in Full Definition modes.

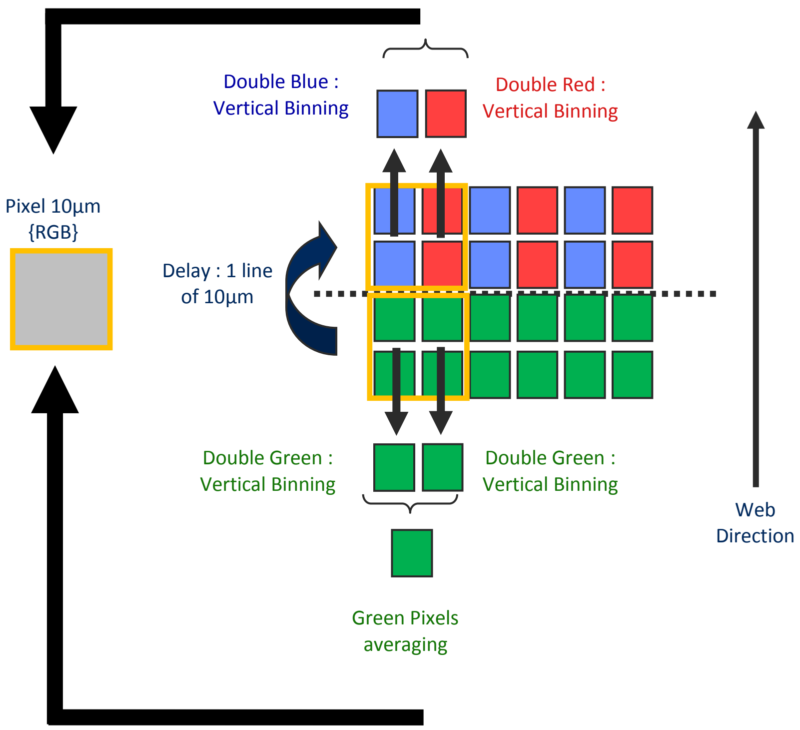


This color mode (5µm) requires the indication of "Forward/Reverse" to the camera in order to manage the delay between the two coloured lines.

2.2.4 True Colour Enhanced Mode (TCE)

10µm Pixels (R,G,B)
Twice less pixels than B/W
Requires x3/2 the data flow of B&W

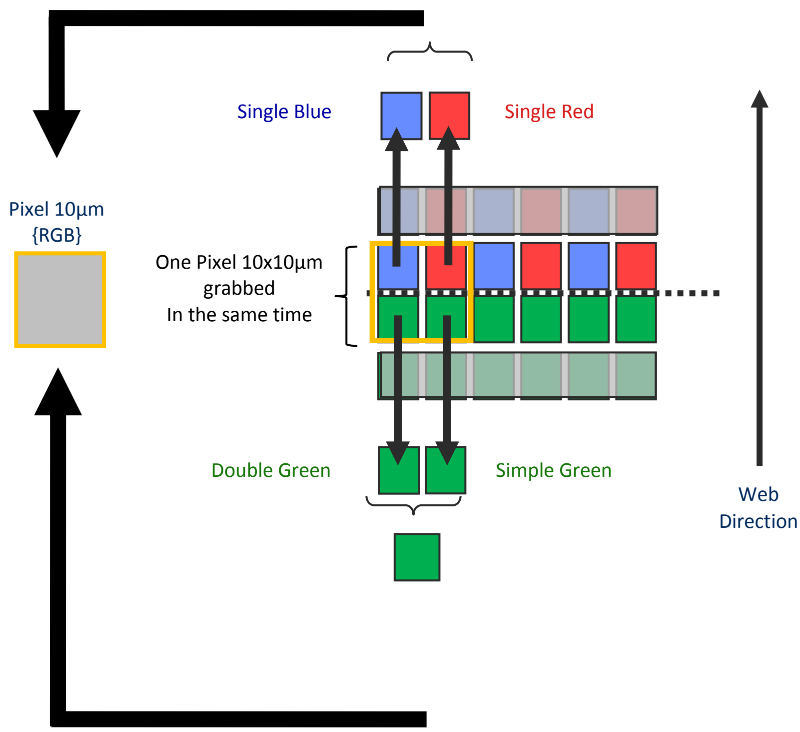
- High Sensitivity True Color mode: Equivalent to 6 x Pixels of 5µm (with their respective colour filters).
- "Full Exposure control" not needed in TC as the TDI is not active (only binning). The Exposure time can be control as for a single line mode.



2.2.5 True Colour Single Mode (TCS)

10µm Pixels (R,G,B)
Twice less pixels than B/W
Requires x3/2 the data flow of B&W

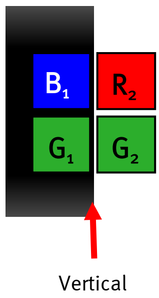
- Sensitivity Half of the TCE mode: Equivalent to 6 x Pixels of 5µm (with their respective colour filters).
- "Full Exposure control" not needed in TC as the TDI is not active (only binning). The Exposure time can be control as for a single line mode.
- Not sensitive to the Scanning direction and the variation of the aspect ratio of the image.



2.2.6 Interpolation Corrections for True Color Modes

Column Interpolation Correction

This interpolation is used to compensate the color error in the Red or the Blue in case of a vertical transition on the web : The Red of the blue value of each colored pixel is corrected if the variation between two neighbour green pixels is significant.



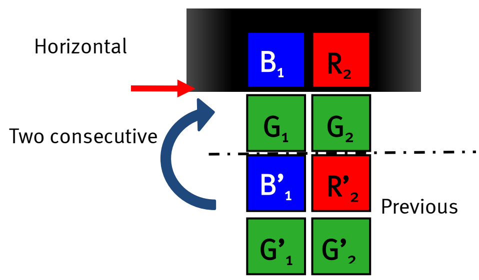
$B_1' = \alpha_B \times B_1$ and α_B is the blue correction, calculated with the variation (G_1–G_2)

$R_2' = \alpha_R \times R_2$ and α_R is the red correction, calculated with the variation (G_1–G_2)

- This interpolation is available for all pixel sizes : 5x5µm but also 10x10µm
- It can be disabled by the customer. By default, it is enabled.

Line Interpolation Correction

This interpolation is used to compensate the color error in the Red or the Blue in case of a horizontal transition on the web in the same "True Color" pixel : A line is memorized and the Red of the blue value of each colored pixel is corrected if the variation between two consecutive green values (previous to next line) is significant :

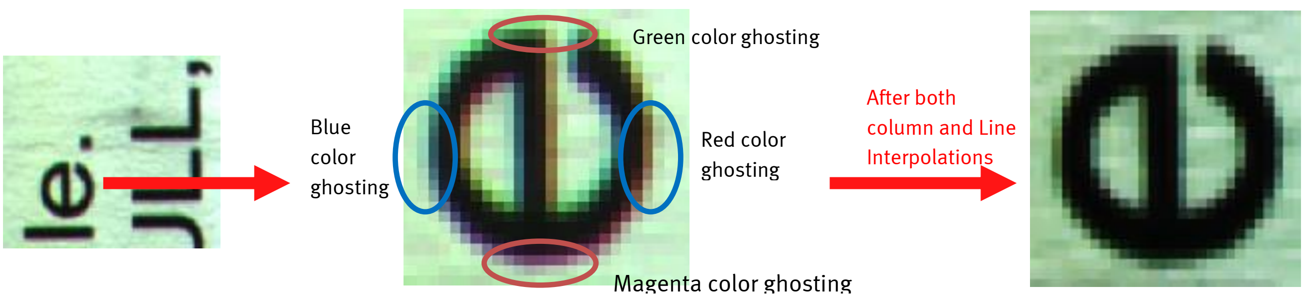


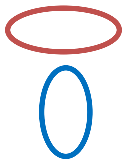
$B_1' = \alpha_B \times B_1$ and α_B is the blue correction, calculated with the variation (G_1–G'_1)

$R_2' = \alpha_R \times R_2$ and α_R is the red correction, calculated with the variation (G_2–G'_2)

- This interpolation is available <u>only for pixel size 10x10µm</u> (True Color Single only)
- It can be enabled by the customer. By default, it is disabled
- This interpolation requires the Forward/Reverse indication sent to the camera for the memorized line.

Effects of the interpolation Corrections for True Color



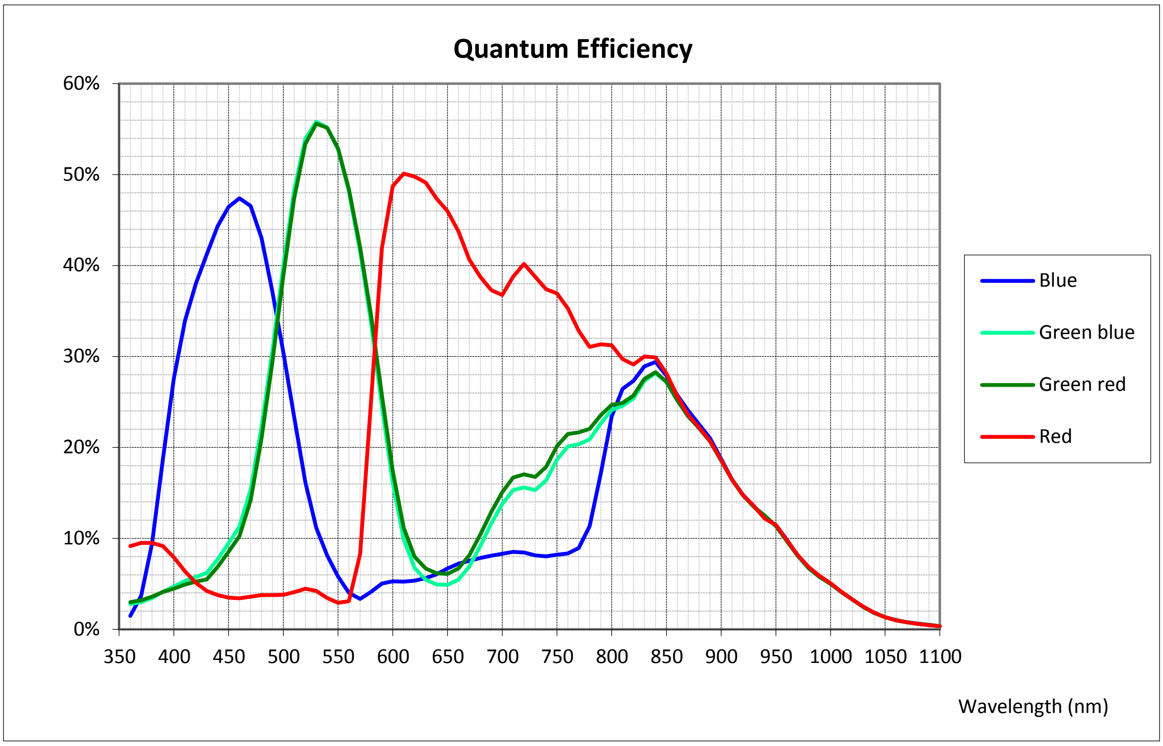


Horizontal transition effect reduced by the "Line Interpolation"

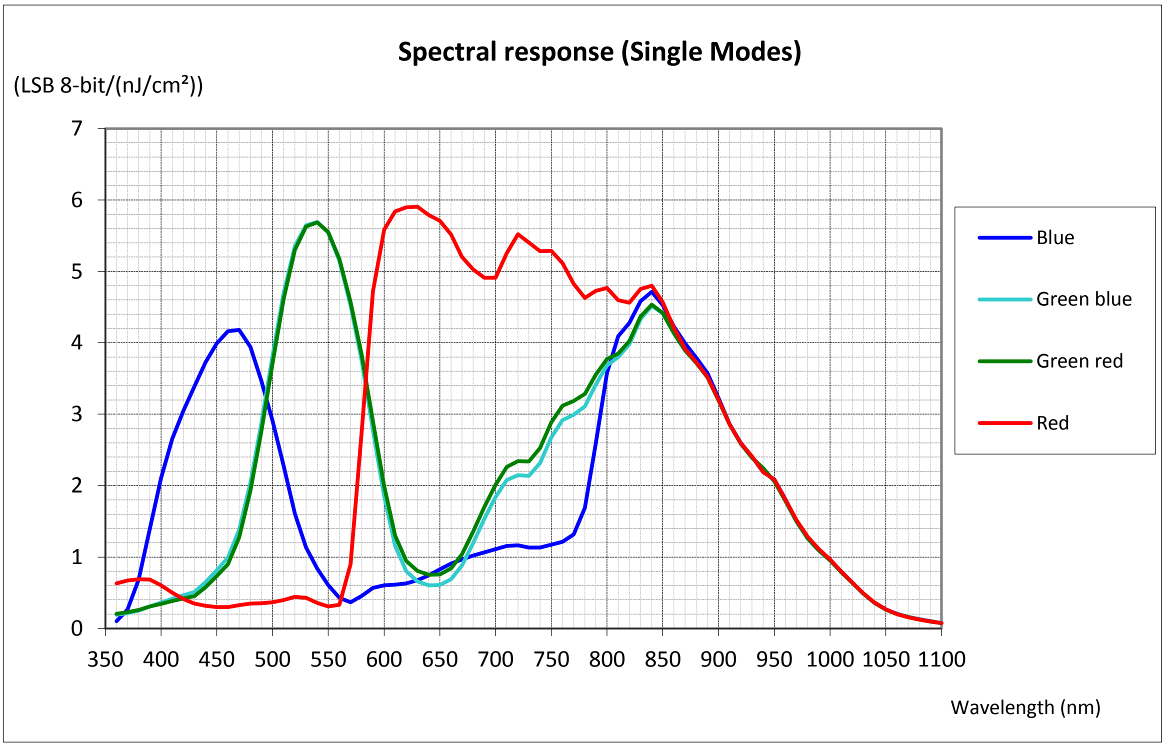
Vertical transition effect reduced by the "Column Interpolation"

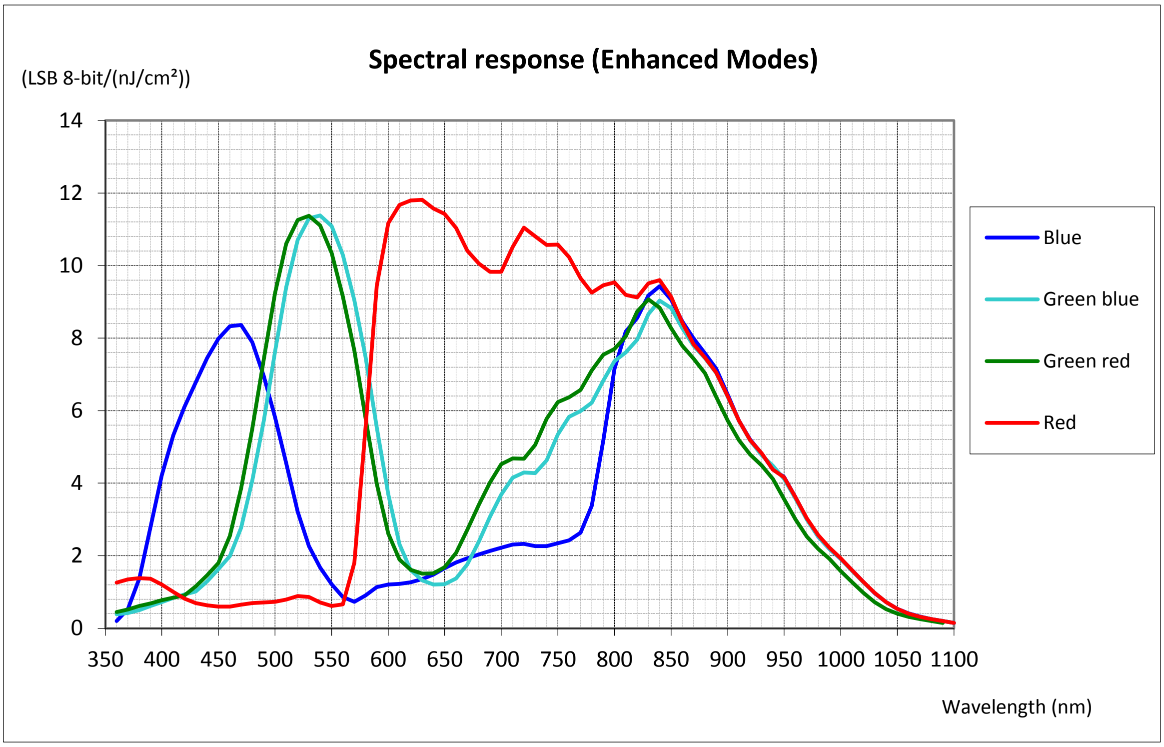
2.3 Response & QE curves

2.3.1 Quantum Efficiency



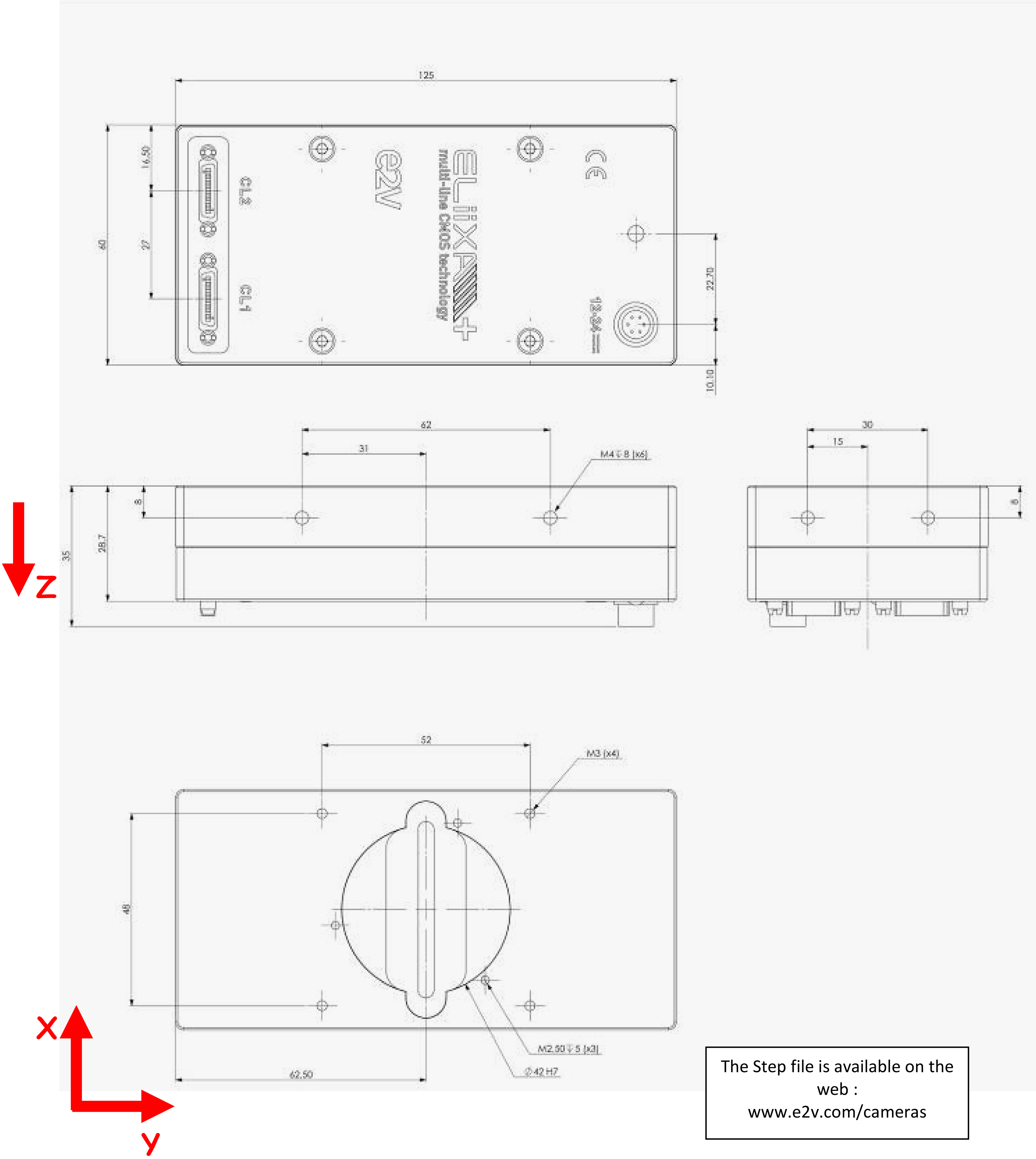
2.3.2 Spectral Response




Spectral response (Enhanced Modes)
(LSB 8-bit/(nJ/cm²))
14
12
10
8
6
4
2
0
350 400 450 500 550 600 650 700 750 800 850 900 950 1000 1050 1100
Wavelength (nm)
Blue
Green blue
Green red
Red

3 CAMERA HARDWARE INTERFACE

3.1 Mechanical Drawings



The Step file is available on the web : www.e2v.com/cameras

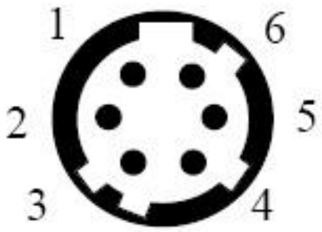
Sensor alignment	
Z = -10.3 mm	±100µm
X = 9.5 mm	±100 µm
Y = 62.5mm	±100 µm
Flatness	50 µm
Rotation (X,Y plan)	±0,15°
Tilt (versus lens mounting plane)	50µm

3.2 Input/output Connectors and LED

3.2.1 Power Connector

Camera connector type: Hirose HR10A-7R-6PB (male)

Cable connector type: Hirose HR10A-7P-6S (female)



Camera side description

Signal	Pin	Signal	Pin
PWR	1	GND	4
PWR	2	GND	5
PWR	3	GND	6

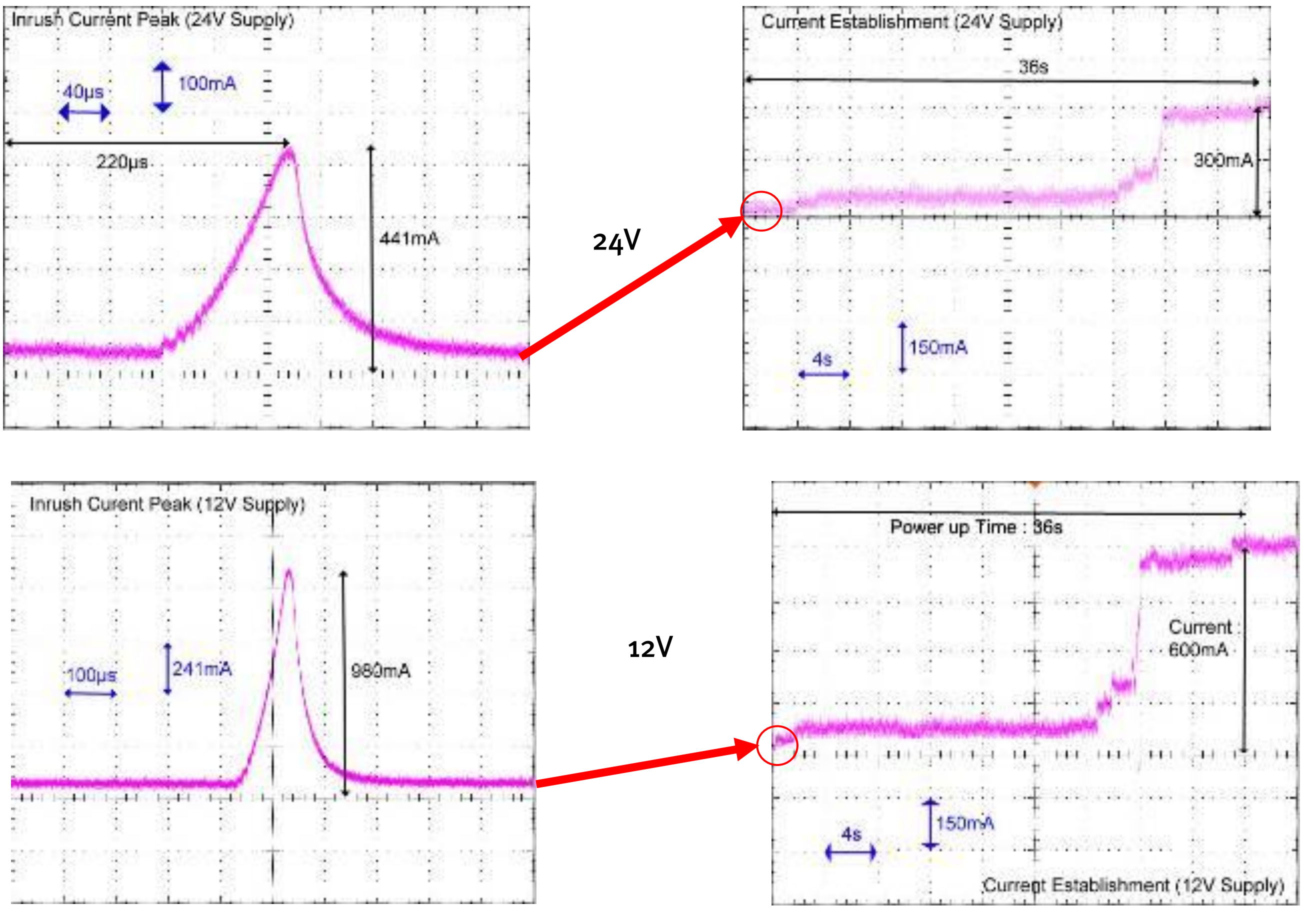
Power supply from 12 to 24v
Power 7,5W max with an typical inrush current peak of **1A** during power up

Typical current/Power during the grab (possible variation : +/- 5%)

Camera supply (Line Period Minimum)	Supply 12V		Supply 24V	
	I(mA)	P(W)	I(mA)	P(W)
Full 8Taps	605	7.26	303	7.272
Deca 10Taps	613	7.356	308	7.392
Base 3Taps RGB	589	7.068	298	7.152
Medium 2x 3Taps RGB	598	7.176	302	7.248

Power Time : Max 40s (Green Light)

<u>**Inrush Current Peak**</u> **Current Establishment time and level**



3.2.2 Status LED Behaviour

After less than 2 seconds of power establishment, the LED first lights up in ORANGE. Then after a Maximum of 40 seconds, the LED must turn in a following colour :

Colour and state	Meaning
Green and continuous	OK
Green and blinking slowly	Waiting for Ext Trig (Trig1 and/or Trig2)
Red and continuous	Camera out of order : Internal firmware error

3.2.3 CameraLink Output Configuration

	Adjacent Channels
Base : 3 Channels RGB 8bits	3 x 85MHz
Medium : 2 x 3 Channels RGB 8bits	2x 3 x 85MHz
Full : 8 Channels 8bits	8 x 85MHz
Deca : 10 Channels 8bits	10 x 85MHz

True Color (Single or Enhanced)

<u>In Base Output Mode (Interpolated mode)</u>

Tap1	RED 1	RED 3	RED 5	RED 7	RED 9	RED 11	RED 13	RED 15	RED 8181	RED 8183	RED 8185	RED 8187	RED 8189	RED 8191
Tap2	GREEN 1	GREEN 2	GREEN 3	GREEN 4	GREEN 5	GREEN 6	GREEN 7	GREEN 8	GREEN 4091	GREEN 4092	GREEN 4093	GREEN 4094	GREEN 4095	GREEN 4096
Tap3	BLUE 2	BLUE 4	BLUE 6	BLUE 8	BLUE 10	BLUE 12	BLUE 14	BLUE 16	BLUE 8182	BLUE 8184	BLUE 8186	BLUE 8188	BLUE 8190	BLUE 8192

<u>In Medium or Dual Base Output Mode (Interpolated mode)</u>

Connector 1

Tap1	RED 1	RED 2	RED 3	RED 4	RED 5	RED 6	RED 7	RED 8	RED 2043	RED 2044	RED 2045	RED 2046	RED 2047	RED 2048
Tap2	GREEN 1	GREEN 2	GREEN 3	GREEN 4	GREEN 5	GREEN 6	GREEN 7	GREEN 8	GREEN 2043	GREEN 2044	GREEN 2045	GREEN 2046	GREEN 2047	GREEN 2048
Tap3	BLUE 1	BLUE 2	BLUE 3	BLUE 4	BLUE 5	BLUE 6	BLUE 7	BLUE 8	BLUE 2043	BLUE 2044	BLUE 2045	BLUE 2046	BLUE 2047	BLUE 2048

Connector 2

Tap4	RED 2049	RED 2050	RED 2051	RED 2052	RED 2053	RED 2054	RED 2055	RED 2056	RED 4091	RED 4092	RED 4093	RED 4094	RED 4095	RED 4096
Tap5	GREEN 2049	GREEN 2050	GREEN 2051	GREEN 2052	GREEN 2053	GREEN 2054	GREEN 2055	GREEN 2056	GREEN 4091	GREEN 4092	GREEN 4093	GREEN 4094	GREEN 4095	GREEN 4096
Tap6	BLUE 2049	BLUE 2050	BLUE 2051	BLUE 2052	BLUE 2053	BLUE 2054	BLUE 2055	BLUE 2056	BLUE 4091	BLUE 4092	BLUE 4093	BLUE 4094	BLUE 4095	BLUE 4096

In Full 8 Taps Output Mode (Interpolated mode)

Connector 1

Tap1	RED 1	BLUE 3	GREEN 6	RED 9	BLUE 11	GREEN 14	RED 17	BLUE 19	RED 4081	BLUE 4083	GREEN 4086	RED 4089	BLUE 4091	GREEN 4094
Tap2	GREEN 1	RED 4	BLUE 6	GREEN 9	RED 12	BLUE 14	GREEN 17	RED 20	GREEN 4081	RED 4084	BLUE 4086	GREEN 4089	RED 4092	BLUE 4094
Tap3	BLUE 1	GREEN 4	RED 7	BLUE 9	GREEN 12	RED 15	BLUE 17	GREEN 20	BLUE 4081	GREEN 4084	RED 4087	BLUE 4089	GREEN 4092	RED 4095

Connector 2

Tap4	RED 2	BLUE 4	GREEN 7	RED 10	BLUE 12	GREEN 15	RED 18	BLUE 20	RED 4082	BLUE 4084	GREEN 4087	RED 4090	BLUE 4092	GREEN 4095
Tap5	GREEN 2	RED 5	BLUE 7	GREEN 10	RED 13	BLUE 15	GREEN 18	RED 21	GREEN 4082	RED 4085	BLUE 4087	GREEN 4090	RED 4093	BLUE 4095
Tap6	BLUE 2	GREEN 5	RED 8	BLUE 10	GREEN 13	RED 16	BLUE 18	GREEN 21	BLUE 4082	GREEN 4085	RED 4088	BLUE 4090	GREEN 4093	RED 4096
Tap7	RED 3	BLUE 5	GREEN 8	RED 11	BLUE 13	GREEN 16	RED 19	BLUE 21	RED 4083	BLUE 4085	GREEN 4088	RED 4091	BLUE 4093	GREEN 4096
Tap8	GREEN 3	RED 6	BLUE 8	GREEN 11	RED 14	BLUE 16	GREEN 19	RED 22	GREEN 4083	RED 4086	BLUE 4088	GREEN 4091	RED 4094	BLUE 4096

In Deca 10 Taps Output Mode (Interpolated mode)

Connector 1

Tap1	RED 1	GREEN 4	BLUE 7	RED 11	GREEN 14	BLUE 17	RED 21	GREEN 24	BLUE 4077	RED 4081	GREEN 4084	BLUE 4087	RED 4091	GREEN 4094
Tap2	GREEN 1	BLUE 4	RED 8	GREEN 11	BLUE 14	RED 18	GREEN 21	BLUE 24	RED 4078	GREEN 4081	BLUE 4084	RED 4088	GREEN 4091	BLUE 4094
Tap3	BLUE 1	RED 5	GREEN 8	BLUE 11	RED 15	GREEN 18	BLUE 21	RED 25	GREEN 4078	BLUE 4081	RED 4085	BLUE 4088	BLUE 4091	RED 4095

Connector 2

Tap4	RED 2	GREEN 5	BLUE 8	RED 12	GREEN 15	BLUE 18	RED 22	GREEN 25	BLUE 4078	RED 4082	GREEN 4085	BLUE 4088	RED 4092	GREEN 4095
Tap5	GREEN 2	BLUE 5	RED 9	GREEN 12	BLUE 15	RED 19	GREEN 22	BLUE 25	RED 4079	GREEN 4082	BLUE 4085	RED 4089	GREEN 4092	BLUE 4095
Tap6	BLUE 2	RED 6	GREEN 9	BLUE 12	RED 16	GREEN 19	BLUE 22	RED 26	GREEN 4079	BLUE 4082	RED 4086	GREEN 4089	BLUE 4092	RED 4096
Tap7	RED 3	GREEN 6	BLUE 9	RED 13	GREEN 16	BLUE 19	RED 23	GREEN 26	BLUE 4079	RED 4083	GREEN 4086	BLUE 4089	RED 4093	GREEN 4096
Tap8	GREEN 3	BLUE 6	RED 10	GREEN 13	BLUE 16	RED 20	GREEN 23	BLUE 26	RED 4080	GREEN 4083	BLUE 4086	RED 4090	GREEN 4093	BLUE 4096
Tap9	BLUE 3	RED 7	GREEN 10	BLUE 13	RED 17	GREEN 20	BLUE 23	RED 27	GREEN 4080	BLUE 4083	RED 4087	GREEN 4090	BLUE 4093	
Tap10	RED 4	GREEN 7	BLUE 10	RED 14	GREEN 17	BLUE 20	RED 24	GREEN 27	BLUE 4080	RED 4084	GREEN 4087	BLUE 4090	RED 4094	

Full Definition (Single or Enhanced)

In Base Output Mode (interpolated mode)

Tap1	RED 1	RED 2	RED 3	RED 4	RED 5	RED 6	RED 7	RED 8		RED 8187	RED 8188	RED 8189	RED 8190	RED 8191	RED 8192
Tap2	GREEN 1	GREEN 2	GREEN 3	GREEN 4	GREEN 5	GREEN 6	GREEN 7	GREEN 8		GREEN 8187	GREEN 8188	GREEN 8189	GREEN 8190	GREEN 8191	GREEN 8192
Tap3	BLUE 1	BLUE 2	BLUE 3	BLUE 4	BLUE 5	BLUE 6	BLUE 7	BLUE 8		BLUE 8187	BLUE 8188	BLUE 8189	BLUE 8190	BLUE 8191	BLUE 8192

In Medium or Dual Base Output Mode (Interpolated mode)

Connector 1

Tap1	RED 1	RED 2	RED 3	RED 4	RED 5	RED 6	RED 7	RED 8		RED 4091	RED 4092	RED 4093	RED 4094	RED 4095	RED 4096
Tap2	GREEN 1	GREEN 2	GREEN 3	GREEN 4	GREEN 5	GREEN 6	GREEN 7	GREEN 8		GREEN 4091	GREEN 4092	GREEN 4093	GREEN 4094	GREEN 4095	GREEN 4096
Tap3	BLUE 1	BLUE 2	BLUE 3	BLUE 4	BLUE 5	BLUE 6	BLUE 7	BLUE 8		BLUE 4091	BLUE 4092	BLUE 4093	BLUE 4094	BLUE 4095	BLUE 4096

Connector 2

Tap4	RED 4097	RED 4098	RED 4099	RED 4100	RED 4101	RED 4102	RED 4103	RED 4104		RED 8187	RED 8188	RED 8189	RED 8190	RED 8191	RED 8192
Tap5	GREEN 4097	GREEN 4098	GREEN 4099	GREEN 4100	GREEN 4101	GREEN 4102	GREEN 4103	GREEN 4104		GREEN 8187	GREEN 8188	GREEN 8189	GREEN 8190	GREEN 8191	GREEN 8192
Tap6	BLUE 4097	BLUE 4098	BLUE 4099	BLUE 4100	BLUE 4101	BLUE 4102	BLUE 4103	BLUE 4104		BLUE 8187	BLUE 8188	BLUE 8189	BLUE 8190	BLUE 8191	BLUE 8192

In Full 8 Taps Output Mode (Raw mode)

Connector 1

Tap1	BLUE 1	RED 2	BLUE 3	RED 4	BLUE 5	RED 6	BLUE 7	RED 8		BLUE 2043	RED 2044	BLUE 2045	RED 2046	BLUE 2047	RED 2048
Tap2	GREEN 1	GREEN 2	GREEN 3	GREEN 4	GREEN 5	GREEN 6	GREEN 7	GREEN 8		GREEN 2043	GREEN 2044	GREEN 2045	GREEN 2046	GREEN 2047	GREEN 2048
Tap3	BLUE 2049	RED 2050	BLUE 2051	RED 2052	BLUE 2053	RED 2054	BLUE 2055	RED 2056		BLUE 4091	RED 4092	BLUE 4093	RED 4094	BLUE 4095	RED 4096

Connector 2

Tap4	GREEN 2049	GREEN 2050	GREEN 2051	GREEN 2052	GREEN 2053	GREEN 2054	GREEN 2055	GREEN 2056		GREEN 4091	GREEN 4092	GREEN 4093	GREEN 4094	GREEN 4095	GREEN 4096
Tap5	BLUE 4097	RED 4098	BLUE 4099	RED 4100	BLUE 4101	RED 4102	BLUE 4103	RED 4104		BLUE 6139	RED 6140	BLUE 6141	RED 6142	BLUE 6143	RED 6144
Tap6	GREEN 4097	GREEN 4098	GREEN 4099	GREEN 4100	GREEN 4101	GREEN 4102	GREEN 4103	GREEN 4104		GREEN 6139	GREEN 6140	GREEN 6141	GREEN 6142	GREEN 6143	GREEN 6144
Tap7	BLUE 6145	RED 6146	BLUE 6147	RED 6148	BLUE 6149	RED 6150	BLUE 6151	RED 6152		BLUE 8187	RED 8188	BLUE 8189	RED 8190	BLUE 8191	RED 8192
Tap8	GREEN 6145	GREEN 6146	GREEN 6147	GREEN 6148	GREEN 6149	GREEN 6150	GREEN 6151	GREEN 6152		GREEN 8187	GREEN 8188	GREEN 8189	GREEN 8190	GREEN 8191	GREEN 8192

In Deca 10 Taps Output Mode (Raw mode)

Connector 1

Tap1	BLUE 1	RED 2	BLUE 3	RED 4	BLUE 5	RED 6	BLUE 7	RED 8	BLUE 1633	RED 1634	BLUE 1635	RED 1636	BLUE 1637	RED 1638
Tap2	GREEN 1	GREEN 2	GREEN 3	GREEN 4	GREEN 5	GREEN 6	GREEN 7	GREEN 8	GREEN 1633	GREEN 1634	GREEN 1635	GREEN 1636	GREEN 1637	GREEN 1638
Tap3	BLUE 1639	RED 1640	BLUE 1641	RED 1642	BLUE 1643	RED 1644	BLUE 1645	RED 1646	BLUE 3271	RED 3272	BLUE 3273	RED 3274	BLUE 3275	RED 3276

Connector 2

Tap4	GREEN 1639	GREEN 1640	GREEN 1641	GREEN 1642	GREEN 1643	GREEN 1644	GREEN 1645	GREEN 1646	GREEN 3271	GREEN 3272	GREEN 3273	GREEN 3274	GREEN 3275	GREEN 3276
Tap5	BLUE 3277	RED 3278	BLUE 3279	RED 3280	BLUE 3281	RED 3282	BLUE 3283	RED 3284	BLUE 4909	RED 4910	BLUE 4911	RED 4912	BLUE 4913	RED 4914
Tap6	GREEN 3277	GREEN 3278	GREEN 3279	GREEN 3280	GREEN 3281	GREEN 3282	GREEN 3283	GREEN 3284	GREEN 4909	GREEN 4910	GREEN 4911	GREEN 4912	GREEN 4913	GREEN 4914
Tap7	BLUE 4915	RED 4916	BLUE 4917	RED 4918	BLUE 4919	RED 4920	BLUE 4921	RED 4922	BLUE 6547	RED 6548	BLUE 6549	RED 6550	BLUE 6551	RED 6552
Tap8	GREEN 4915	GREEN 4916	GREEN 4917	GREEN 4918	GREEN 4919	GREEN 4920	GREEN 4921	GREEN 4922	GREEN 6547	GREEN 6548	GREEN 6549	GREEN 6550	GREEN 6551	GREEN 6552
Tap9	BLUE 6553	RED 6554	BLUE 6555	RED 6556	BLUE 6557	RED 6558	BLUE 6559	RED 6560	BLUE 8185	RED 8186	BLUE 8187	RED 8188	BLUE 8189	RED 8190
Tap10	GREEN 6553	GREEN 6554	GREEN 6555	GREEN 6556	GREEN 6557	GREEN 6558	GREEN 6559	GREEN 6560	GREEN 8185	GREEN 8186	GREEN 8187	GREEN 8188	GREEN 8189	GREEN 8190

Notes :

“Interpolated mode” means that the Triplet (RGB) of each pixel has already been calculated or “interpolated”.

“Raw mode” means that all the available colored information are outputted without Associating one RGB Triplet for each pixel : This requires to make this interpolation in the frame grabber or in the application.

4 STANDARD CONFORMITY

The ELIIXA+ cameras have been tested using the following equipment:

- A shielded power supply cable
- A Camera Link data transfer cable ref. MVC-1-1-5-2M from CEI (Component Express, Inc.)

e2v recommends using the same configuration to ensure the compliance with the following standards.

4.1 CE Conformity

The ELIIXA+ cameras comply with the requirements of the EMC (European) directive 2004/108/EC (EN50081-2, EN 61000-6-2).

4.2 FCC Conformity

The ELIIXA+ cameras further comply with Part 15 of the FCC rules, which states that: Operation is subject to the following two conditions:

- This device may not cause harmful interference, and
- This device must accept any interference received, including interference that may cause undesired operation

This equipment has been tested and found to comply with the limits for Class A digital device, pursuant to part 15 of the FCC rules. These limits are designed to provide reasonable protection against harmful interference when the equipment is operated in a commercial environment. This equipment generates, uses and can radiate radio frequency energy and, if not installed and used in accordance with the instruction manual, may cause harmful interference to radio communications. Operation of this equipment in a residential area is likely to cause harmful interference in which case the user will be required to correct the interference at his own expense.

Warning: Changes or modifications to this unit not expressly approved by the party responsible for compliance could void the user's authority to operate this equipment.

4.3 RoHs Conformity

ELIIXA+ cameras comply with the requirements of the RoHS directive 2011/65/EU.

5 GETTING STARTED

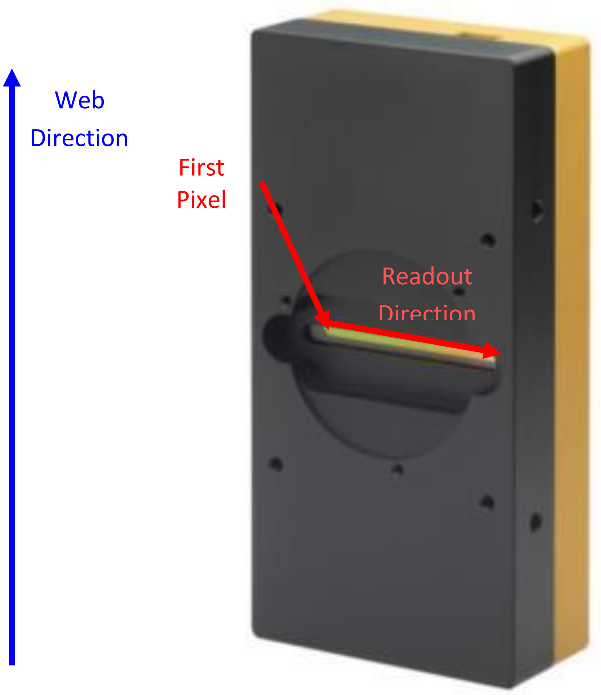
5.1 Out of the box

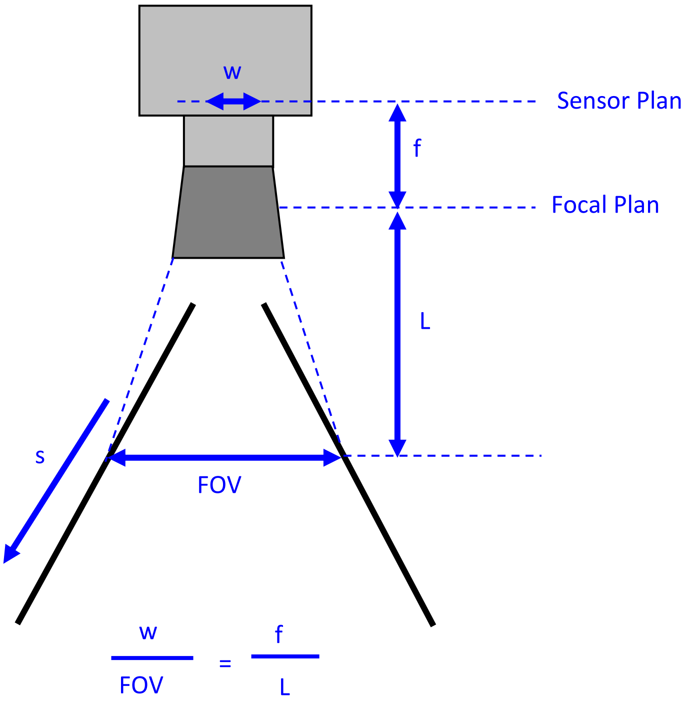
The contains of the Camera box is the following :

- One Camera ELIIXA+
- Power connector (Hirose HR10A-7P-6S -female)

There is no CDROM delivered with the Camera : Both User Manual (this document) and CommCam control software have to be downloaded from the web site : This ensure you to have an up-to-date version.
Main Camera page : **www.e2v.com/cameras**
On the appropriate Camera Page (ELIIXA+ 8k/4k color) you'll find a download link
first version of CommCam compliant is indicated in the last Chapter
CommCam download requires a login/password :
Login : **commcam**
Password : **chartreuse**

5.2 Setting up in the system





The Compliant Lenses Mounts are detailed in Appendix D

6 CAMERA SOFTWARE INTERFACE

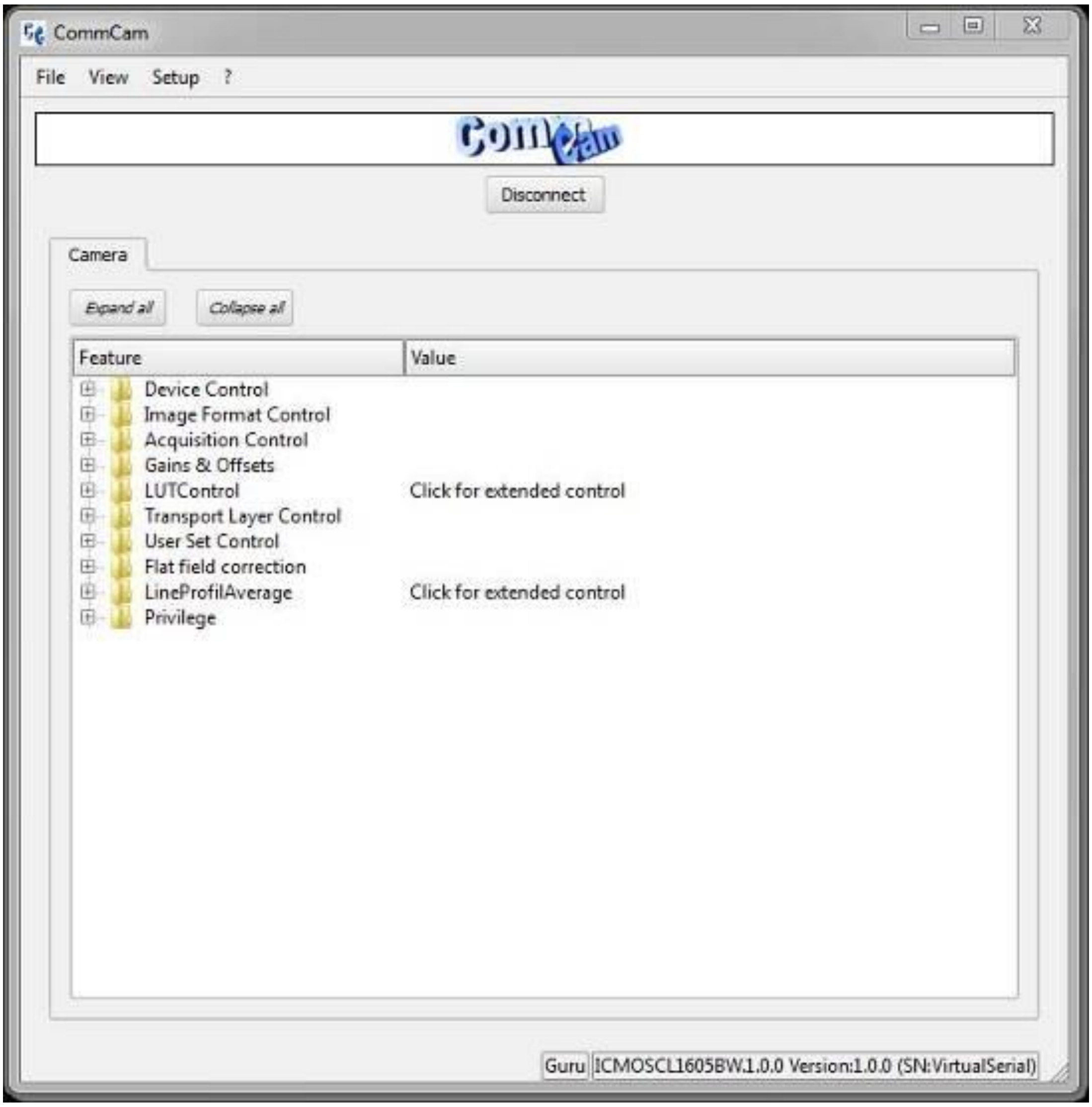
6.1 Control and Interface

As all the e2v Cameras, the ELIIXA+ CL is delivered with the friendly interface control software COMMCAM.UCL (as "Ultimate Camera Link") which is based on the GenICam standard

COMMCAM recognizes and detects automatically all the UCL Cameras connected on any transport layers (Camera Link or COM ports) of your system.

Once connected to the Camera you have an easy access to all its features. The visibility of these features can be associated to three types of users: Beginner, Expert or Guru. Then you can make life easy for simple users.

Minimum version of CommCam is 2.2.0 in order to recognize the ELIIXA+ 8k/4k color Camera.



6.2 Serial Protocol and Command Format

The Camera Link interface provides two LVDS signal pairs for communication between the camera and the frame grabber. This is an asynchronous serial communication based on RS-232 protocol.

The serial line configuration is:

- Full duplex/without handshaking
- 9600 bauds (default), 8-bit data, no parity bit, 1 stop bit. The baud rate can be set up to 115200

6.2.1 Syntax

Internal camera configurations are activated by write or readout commands.
The command syntax for write operation is:

w <command_name> <command_parameters><CR>

The command syntax for readout operation is:

r <command_name><CR>

6.2.2 Command Processing

Each command received by the camera is processed:

- The setting is implemented (if valid)
- The camera returns “>”<return code><CR>
- The camera return code has to be received before sending a new command.

The camera return code has to be received before sending a new command. Some commands are longer than the others : Waiting for the return code ensure a good treatment of all the commands
Without saturating the buffer of the camera

6.2.3 Camera Returned Code

Returned code	meaning
>0	(or “>OK”) : All right, the command will be implemented
>3	Error Bad CRC (for write command only)
>16	Invalid Command ID (Command not recognized or doesn't exist)
>33	Invalid Access (the receipt of the last command has failed).
>34	Parameter out of range (the parameter of the last command sent is out of range).
>35	Access Failure (bad communication between two internal devices).

6.2.4 GenICam ready

The CameraLink Standard is not yet compliant with GenICam Standard, but as much as possible, each command of the ELIIXA+ will have its correspondence with the Standard Feature Naming Convention of the Genlcam Standard.

This correspondence is given in parenthesis for each feature/command as the following example :

- **Vendor name** (*DeviceVendorName*) **: “e2v”**

7 Camera Commands

7.1 Information

These values allow to identify the Camera. They can be accessed in CommCam software in the "Info" section
All these values are fixed in factory and can't be changed (shaded) except the Camera User ID which can be fixed by the Customer :

- **Vendor name** (*DeviceVendorName*) : "e2v"
  - ⇨ Read function : "**r vdnm**";
    Returned by the camera : "e2v", string of 32 bytes (including "/0")
  - ⇨ Can not be written

- **Model Name** (*DeviceModelName*) : Internal name for GenICam :
  - ⇨ Read function : "**r mdnm**";
    Returned by the camera : String of 32 bytes (including "/0") :
  - ⇨ Can not be written

- **Device Manufacturer Info** (*DeviceManufacturerInfo*) : Get Camera ID
  - ⇨ Read function : "**r idnb**";
    Returned by the camera : String of 128 bytes (including "/0")
  - ⇨ Can not be written

- **Device Version** (*DeviceVersion*) : Get Camera Hardware version
  - ⇨ Read function : "**r dhwv**";
    Returned by the camera : String of 32 bytes (including "/0")
  - ⇨ Can not be written

- **Device Firmware Version** (*DeviceFirmwareVersion*): Get camera synthetic firmware
  - ⇨ Read function : "**r dfwv**";
    Returned by the camera : String of 16 bytes (including "/0")
  - ⇨ Can not be written

- **Device SFNC Version : 1.5.0**
  These Parameters (Major, Minor, Sub Minor) are only virtual ones in order to give the SFNC compliance of the Camera.

- **Device ID** (*DeviceID*) : Camera Factory identifier ID
  - ⇨ Read function : "**r deid**";
    Returned by the camera : String of 128 bytes (including "/0")
  - ⇨ Write function : "**w deid <idstr>**"

- **Device User ID** (*DeviceUserID*) : Camera user identifier ID
  - ⇨ Read function : "**r cust**";
    Returned by the camera : String of 128 bytes (including "/0")
  - ⇨ Write function : "**w cust <idstr>**"

- **Electronic board ID** (*ElectronicBoardID*) : Get PcB Board ID
  - ⇨ Read function : "**r boid**";
    Returned by the camera : String of 32 bytes (including "/0")
  - ⇨ Can not be written
- **Device Temperature Selector** (*DeviceTemperatureSelector*) : **MainBoard**
  - ⇨ Can not be written

- **Device Temperature** (*DeviceTemperature*) : Get Main Board Temperature
  ⇨ Read function : "**r temp**";
  Return by the camera : Temperature in Q10.2 format (8 bits signed + 2 bits below comma). Value is between -512 to 511 in °C.

- **Device Serial Port Selection** : Indicates the Serial Port on which the Camera is connected.

- **Device Serial Port Baud Rate** (*ComBaudRate*): Set the Camera BaudRate
  ⇨ Read function : "**r baud**";
  Returned by the camera : Value of the Baud Rate
  ⇨ Write function : "**w baud**" <index> with the index as follows :
    - 1 : 9600 Bauds (default value at power up)
    - 2 : 19200Bauds
    - 6 : 57600Bauds
    - 12 : 115200Bauds

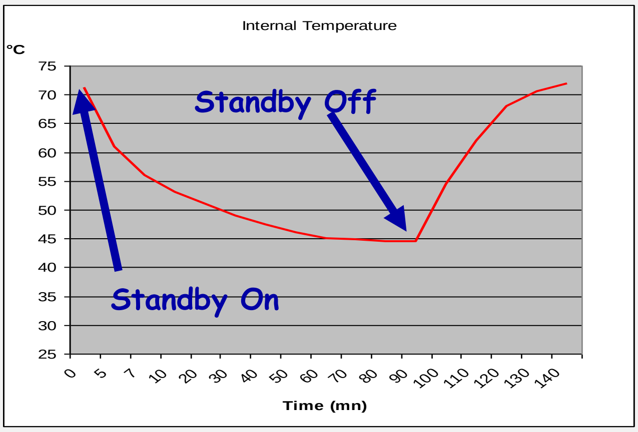
- **Standby Mode** (*Standby*) : Activation of the Standby mode of the Camera
  ⇨ Read function : "**r stby**";
  Returned by the camera : Boolean.
    - 0 : Disable Standby mode (False)
    - 1 : Enable stanby mode (True)

  ⇨ Write function : "**w stby <val>**"; <val> is 0 or 1.

A standby mode, what for ?

The Standby mode stops all activity on the sensor level. The power dissipation drops down to about 6W. During the standby mode, the grab is stopped

Once the Standby mode turned off, the Camera recovers in less than 1ms to send images again from the sensor.



- **Camera status** : Get the Camera status register (32bits Integer)
  ⇨ Read function : "**r stat**";
  Returned by the camera : 32bits integer :

    - **Bit 0** : (*StatusWaitForTrigger*) : True if no trig received from more than 1sec
    - **Bit 1** : (*StatusTriggerTooFast*) : Missing triggers. Trig signal too fast
    - **Bit 2** : (*StatusSensorConnection*) : True is the Sensor pattern is checked as failed.
    - Bit 3, 4, 5, 6, 7 : Reserved
    - **Bit 8** : (*StatusWarningOverflow*) : True is an overflow occurs during FFC or Tap balance processing.
    - **Bit 9** : (*StatusWarningUnderflow*) : True is an underflow occurs during FFC or Tap balance processing
    - Bits 10 : Reserved
    - **Bits 11** : Scrolling Direction : 0 = Forward, 1 = Reverse. Updated only by external CC3 (CameraLink)
    - Bits, 12, 13, 14, 15 : Reserved
    - **Bit 16** : (*StatusErrorHardware*) : True if hardware error detected
    - Bits 17 to 31 : Reserved

7.1.1 Command Table

Feature	CL Command	Description
DeviceVendorName	r vdnm	Get camera vendor name as a string (32 bytes long including '\0')
DeviceModelName	r mdnm	Get camera model name as a string (32 bytes long including '\0')
DeviceManufacturerInfo	r idnb	Get camera ID as a string (**48** bytes long including '\0')
DeviceVersion	r dhwv	Get camera version as a string (hardware version) (32 bytes long including '\0')
DeviceFirmwareVersion	r dfwv	Get camera synthetic firmware version (PKG version) as a string (**32** bytes long including '\0')
DeviceSFNCVersionMajor	Xml Virtual	1
DeviceSFNCVersionMinor	Xml Virtual	5
DeviceSFNCVersionSubMinor	Xml Virtual	0
DeviceID	r deid	Read Serial Nb
DeviceUserID	r cust	Get device user identifier as a string (**16** bytes long including '\0')
	w cust <idstr>	Set camera identifier to <idstr>
ElectronicBoardID	r boid	Read Electronic Board ID
DeviceTemperature	r temp	Read Mainboard internal temperature (format signed Q10.2 = signed 8 bits, plus 2 bits below comma. Value from -512 to +511) in °C
DeviceTemperatureSelector	Xml Virtual	
Standby	r stby	Read Standby state (CMOS sensor)
	w stby 0	Disable standby mode ("False")
	w stby 1	Enable standby mode ("True"), no more video available but save power and temperature
Device Serial Port Baud Rate	r baud	Get current baud rate (This feature is not saved in camera)
	w baud 1	Set baud rate to "9600Bds"
	w baud 2	Set baud rate to "19200Bds"
	w baud 6	Set baud rate to "57600Bds"
	w baud 12	Set baud rate to "115200Bds"

Feature	CL Command	Description
STATUS	r stat	Get camera status (see below for details)
StatusWaitForTrigger		Bit 0: true if camera waits for a trigger during more than 1s
Satus trigger too fast		Bit 1: true if camera trigger is too fast
StatusWarningOverflow		Bit 8: true if a an overflow occurs during FFC calibration or Tap balance (available only for integrator/user mode)
StatusWarningUnderflow		Bit 9: true if a an underflow occurs during FFC calibration or Tap balance (available only for integrator/user mode)
Cc3 Scrolling direction		Bit 11: 0 : forward, 1: reverse
StatusErrorHardware		Bit 16 : true if hardware error detected

7.2 Image Format

- **Sensor Width** (*SensorWidth*) : Get the physical width of the Sensor. This value is available in the CommCam "Image Format Control" section :
  - ⇨ Read function : "**r snsw**";
    Return by the sensor : Integer 8192.
  - ⇨ Can not be written;

- **Sensor Height** (*SensorHeight*) : Get the physical height of the Sensor. This value is available in the CommCam "Image Format Control" section :
  - ⇨ No Access. Virtual command in xml"; Value always = 1

- **Width Max** (*WidthMax*) : Get the Maximum Width of the Sensor. This value is available in the CommCam "Image Format Control" section :
  - ⇨ No Access. The value is mapped on "SensorWidth"

- **Height Max** (*HeigthMax*) : Get the Maximum height of the Sensor. This value is available in the CommCam "Image Format Control" section :
  - ⇨ No Access. Virtual command in xml"; Value always = 1

- **Output mode** (*OutputMode*) : Set the CameraLink Output mode (refer also to Chapter : CameraLink Output Configuration). This command is available in the CommCam "Image Format Control" section :
  - ⇨ Read function : "**r mode**";
    Returned by the camera : Output mode from 0 to 3 (see table below).
  - ⇨ Write function : **"w mode"** <value> :
    detailed in the table below :

Modes	Connector CL1	Connector CL2	Mode value
Base 3 Channels RGB 8 bits	3 x 8 bits	-	0
Dual Base 3 Channels RGB 8 bits	3 x 8 bits	3 x 8 bits	1
Full 8 Channels 8bits	8 x 8 bits		2
Full+ 10 Channels 8bits	10 x 8 bits		3

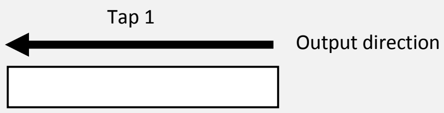
  - "0" : BaseRGB8bits
  - "1" : DualBaseRGB8bits
  - "2" : RawFull8Outputs8bits
  - "3" : RawFullPlus10Outputs8bits

Switching between Sensor modes

The "Raw" output modes (8 or 10Taps) are achieved by loading another FPGA firmware. Then the switch time between Base or Dual Base modes and Full 8taps or Full+ 10Taps mode is about several seconds (**maximum 9s**). **When these output modes are activated, the Color selection (see below p29) is no more possible.**

Structure of the Camera Link Channels for interfacing

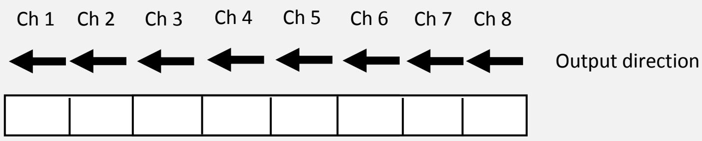
- **Base Mode :** 1 Tap RGB 24 bits (3 channels), outputted from Left to Right.
  - 3x4096 pixels (1 RGB Tap) in True Color Mode
  - 3x8192 pixels (1 RGB Tap) in Full Definition Modes



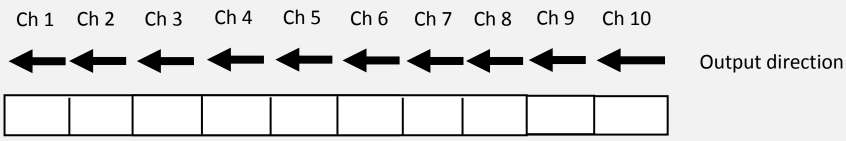
- **Dual Base Mode :** 2 Taps RGB 24 bits (2 x 3 channels), outputted from Left to Right
  - 2 x (3x2048) pixels (2 RGB Taps) in True Color Mode
  - 2 x (3x4096) pixels (2 RGB Taps) in Full Definition Modes

The two following output modes are considered as "Monochrome" on the Frame Grabber side. A specific interpolation on the application level is required to get back the color buffer.

- **FULL Mode :** 8 Taps Separate, outputted from Left to Right.
  - 8x1536 pixels each Channel in True Color Mode : 12288 pixels total : 4096 Green, 4096 Red and 4096 Blue pixels
  - 8x2048 pixels each Channel in Full Definition Modes : 16384 pixels total : 8192 Green, 4096 Red and 4096 Blue pixels



- **FULL+ (Deca) Mode :** 10 Taps Separate, outputted from Left to Right.
  - 10x1229 pixels each Channel in True Color Mode : 12290 pixels total : 4096 Green, 4096 Red and 4096 Blue pixels. The last pixel of Tap9 and Tap10 are valid but black (2 pixels).
  - 10x1638 pixels each Channel in Full Definition Modes : 16380 pixels total : 8190 Green, 4095 Red and 4095 Blue pixels



- **Output Frequency** (*OutputFrequency*) : Set the CameraLink Data Output Frequency. This value is available in the CommCam "Image Format Control" section :
  - ⇨ Read function : "**r clfq**";
    Return by the Camera : Frequency from 0 to 5
  - ⇨ Write Function : "**w clfq** <value>"
    - "0" : 85MHz (default).
    - "1" : 60MHz.
    - "2" : 65MHz.
    - "3" : 70MHz.
    - "4" : 75MHz.
    - "5" : 80MHz.

- **Sensor Mode** (*SensorMode*) : Defines the number of Line used on the Sensor. This command is available in the CommCam "Image Format Control" section :
  - ⇨ Read function : "**r smod**";
    Returned by the camera : Integer from 0 to 2
  - ⇨ Write function : "**w smod**" <value> :
    - "0" : True Color Enhanced
    - "1" : Full definition Single
    - "2" : Full Definition Enhanced
    - "3" : True Color Single

7.2.1 Structure of the Sensor

Pixels in « True Color » modes :

$P_n = R_{(n)}, G_{B(n)}, B_{(n)}$

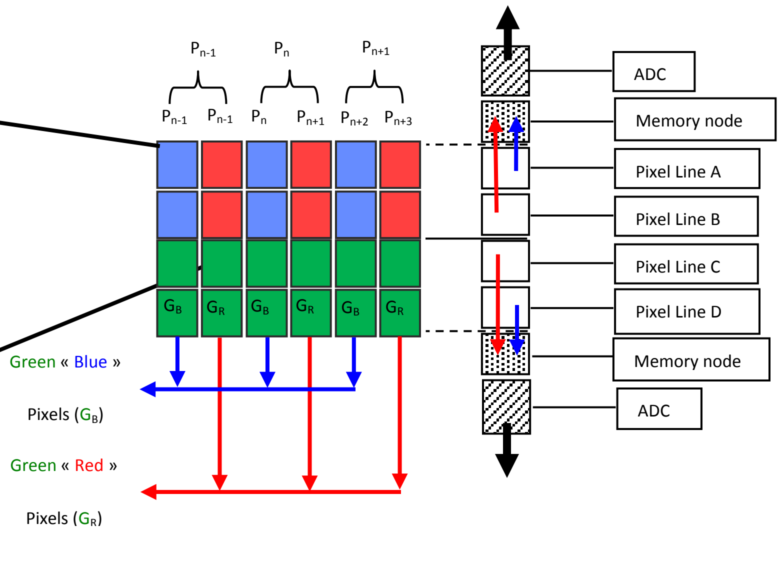
Each Pixel has to be considered as a 10x10µm pixel including 2xGreen + 1xRed and 1x Blue, all Pixels of 5x5µm. The Enhanced mode has the double of this information

Pixels in « Full Definition » modes :

$P_n = R_{(n-1)}, G_{B(n)}, B_{(n)}$

$P_{n+1} = R_{(n+1)}, G_{R(n+1)}, B_{(n)}$

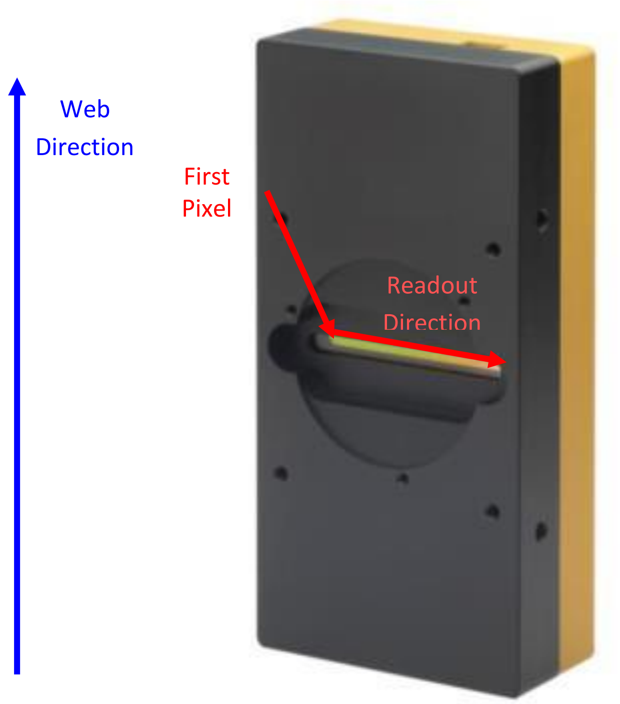
Each pixel has to be considered as a 5x5µm pixels. The Red or Blue information is alternatively interpolated from the neighbour pixel. The Enhanced mode has the double of this information : In this mode, the sensor works in TDI Mode and requires a specific mode ("Full Exposure Control") when the User wants to control the exposure



7.2.2 Forward/Reverse

Forward/reverse information has to be set correctly For the re-ordering of the colors.

The Forward direction is defined as detailed beside :



<u>**Note**</u> : The delay for the Camera to take in account a change in the ScanDirection value is minimum **100ms**.

This information can be set dynamically by using the **CC3 Trig signal** of the CameraLink connector (change the direction "on the fly").

In these case, the Trigger level signification is :

- "0" : Forward.
- "1" : Reverse

- **Scan Direction** (*ScanDirection*) : Set the scan direction for the sensor. This value is available in the CommCam "Image Format Control" section :
  - ⇨ Read function : "**r scdi**";
    Return by the Camera : 0, 1 or 2 (Forward/reverse/external)
  - ⇨ Write function : "**w scdi** <value>";
    - "0" : Forward.
    - "1" : Reverse
    - "2" : Externally controlled (by CC3 of the CameraLink Sync signals)

> The Scan Direction is useful only in <u>Enhanced Modes</u> (True Colour or Full Definition) as a delay is used for the acquisition between lines.
> It is also necessary when the Horizontal Interpolation is activated (True Colour Single only)

7.2.3 Test Image Pattern Selector

- **Test Image Selector** (*TestImageSelector*) : Defines if the data comes from the Sensor or the FPGA (test Pattern). This command is available in the CommCam "Image Format" section :
  ⇨ Read function : "**r srce**";
  Returned by the camera : "0" if Source from the Sensor and "1 to 5" if test pattern active
  ⇨ Write function : "**w srce**" <value> :
    - "0" : To switch to CCD sensor image
    - "1" : Grey Horizontal Ramp (Fixed) : **See AppendixA**
    - "2" : White Pattern (Uniform white image : 255)
    - "3" : Grey Pattern (Uniform middle Grey : 128 on each color))
    - "4" : Black Pattern (Uniform white image : 0)
    - "5" : Grey vertical Ramp (moving)

The test pattern is generated in the FPGA : It's used to point out any interface problem with the Frame Grabber.
When any of the Test pattern is enabled, the whole processing chain of the FPGA is disabled.

7.2.4 Color Selection

- **Color Selection** :
  Disables each of the 3 colors.. This command is available in the CommCam "Image Format" section.
  ⇨ Read function : "**r cold**";
  Returned by the camera : Integer corresponding to one of the 3 different step values :
    - Bit 0 : Red color disabled if set to 1
    - Bit 1 : Blue color disabled if set to 1
    - Bit 2 : Green (both GreenRed and GreenBlue) color disabled if set to 1

  ⇨ Write function : "**w cold**" <val> ;

> The Color Selection is not possible when the Camera output mode are full or Deca (8/10 Taps) Raw modes.

7.2.5 Command Table

Feature	Command	Description
SensorWidth	r snsw	Get sensor physical width.
SensorHeight	Xml virtual	
WidthMax	Map on SensorWidth	
HeightMax	Xml virtual	
Height	Xml virtual	
Width	Xml virtual	Depends on (OuputRegion, OuputRegionWidth) and SensorWidth
SensorMode	r smod	Get sensor mode
	w smod 0	Set sensor mode to "True Color Enhanced"(*)
	w smod 1	Set sensor mode to "Full Definition single" (*)
	w smod 2	Set sensor mode to "Full Definition Enhanced" (*)
	w smod 3	Set sensor mode to "True Color Single" (*)

Feature	Command	Description
ScanDirection	r scdi	Get scan direction
	w scdi 0	Set scan direction to “forward”
	w scdi 1	Set scan direction to “reverse”
	w scdi 2	Set scan direction to “Externally controlled direction via CC3 Camera Link (CC3=0 forward, CC3=1 reverse)”
OutputMode	r mode	Get output mode (CameraLink configuration and CMOS sensor resolution)
	w mode 0	Set output mode to “BaseRGB8bits”
	w mode 1	Set output mode to “DualBaseRGB8bits”
	w mode 2	Set output mode to “RawFull8Outputs8bits”
	w mode 3	Set output mode to “RawFullPlus10Outputs8bits”
OutputFrequency	r clfq	Get Camera Link frequency
	w clfq 0	Set Camera Link frequency to 85MHz
	w clfq 1	Set Camera Link frequency to **60MHz**
	w clfq 2	Set Camera Link frequency to **65MHz**
	w clfq 3	Set Camera Link frequency to **70MHz**
	w clfq 4	Set Camera Link frequency to **75MHz**
	w clfq 5	Set Camera Link frequency to **80MHz**
TestImageSelector	r srce	Get test (output FPGA) image pattern
	w srce 0	Set test (output FPGA) image pattern to “Off”, processing chaine activated
	w srce 1	Set test (output FPGA) image pattern to “GreyHorizontalRamp”, processing chaine desactivated
	w srce 2	Set test (output FPGA) image pattern to “White pattern”, processing chaine desactivated
	w srce 3	Set test (output FPGA) image pattern to “gray pattern”, processing chaine desactivated
	w srce 4	Set test (output FPGA) image pattern to “Black pattern”, processing chaine desactivated
	w srce 5	Set test (output FPGA) image pattern to “GreyVerticalRampMoving”, processing chaine desactivated
Color Selection	r cold	Read the color selection
	w cold <val>	Set the color selection. Val : - Bit 0 : Disables the Red color - Bit 1 : Disables the Blue color - Bit 2 : Disables both $\text{Green}_{\text{Red}}$ and $\text{Green}_{\text{Blue}}$ colors

7.3 Acquisition Control

This section deals with all the Exposure, Line period and synchronisation modes

- **Synchronisation Mode** (TriggerPreset) : Timed or Triggered, it defines how the grabbing is synchronized**.** This command is available in the CommCam "Acquisition Control" section :

  ⇨ Read function : "**r sync**";
  Returned by the camera :
  - "0" : Internal Line Trigger with Exposure time Internally Controlled (Free Run). Not available when Sensor mode is set in "Full Definition Enhanced"
  - "1" : External Trigger with Exposure Time Internally Controlled. Available also when Sensor mode is set in "Full Definition Enhanced" (See : "Full Exposure Control" below)
  - "2" : External Trigger with maximum Exposure time
  - "3" : One External with Exposure Time Externally Controlled. The same Trigger signal defines the line period and its low level defines the exposure time. Available also when Sensor mode is set in "Full Definition Enhanced" (See : "Full Exposure Control" below).
  - "4" : Two External Triggers with Exposure Time Externally Controlled : CC2 defines the start of the exposure (and also the start Line) and CC1 defines the Stop of the exposure. Not available when Sensor mode is set in "Full Definition Enhanced".
  - "5" : Internal Line Trigger with maximum Exposure Time

  ⇨ Write function : "**w sync**" <value>

> The Timing diagrams associated to each Synchronization mode and the Timing values associated are detailed in the APPENDIX B of this document.

- **Exposure time** (*ExposureTime*): Defines the exposure time when set in the Camera. This command is available in the CommCam "Acquisition Control" section :

  ⇨ Read function : "**r tint**";
  Returned by the camera : Integer from 15 to 65535 (=1,5µs to 6553,5µs by step o 0,1µs)

  ⇨ Write function : "**w tint**" <value> ;
  This value of exposure time is taken in account only when the synchronisation mode is "free run" (0) or "Ext Trig with Exposure time set" (1). Otherwise it's ignored.

> Due to the limitation of the timing pixel inside the sensor, the Exposure time has to be set by taking in account the limitation detailed in the APPENDIX B of this document.
> The **Minimum exposure time** which can be set is **1,5µs**

- **Line Period** (*TriggerTooSlow*) : Defines the Line Period of the Camera in Timed mode**.** This command is available in the CommCam "Acquisition Control" section :
  - ⇨ Read function : "**r tper**";
    Returned by the camera : Integer from 151 to 65536 (=15,1µs to 6553,6µs by step o 100ns)
  - ⇨ Write function : "**w tper**" <value> ;
    The line period is active only in Free Run modes. It's also disabled if in this mode, the Integration time is set higher than the Line Period.

The Tables of the minimum Line Period (Max Line Rate) versus the Data rate and the output mode chosen are given in Appendix C (Chap. 9.2) of this document.

- **Trigger Too Slow :** Defines the Time limit (in ms) for the Camera to consider that the Incoming trigger is too slow**.** This command is available in the CommCam "Acquisition Control" section :
  - ⇨ Read function : "**r tgts**";
    Returned by the camera : Integer from 1 to 5368 (1 to 5368 milliseconds by step of 1ms)
  - ⇨ Write function : "**w tgts**" ‹value› ;

Trigger too Slow

By default, the trigger is considered too slow after 1000ms of missing Incoming Trigger.

This limit can be tuned now by the User. This tuning is particularly important when the camera is in **Full Definition Enhanced** with the **Exposure control active** : In this mode the incoming Line Period is delayed from one line to be reproduced in the camera after an exact measurement of the Line Period. If the trigger stops for a period of time below the limit, this will considered as a "long time Line" and not a stop : Then the next line will be delayed from the same value with the risk to loose new incoming triggers.

The Trigger too Slow limit has to be set at a value which is considered in the Application as the minimum value for a real stop in the incoming trigger.

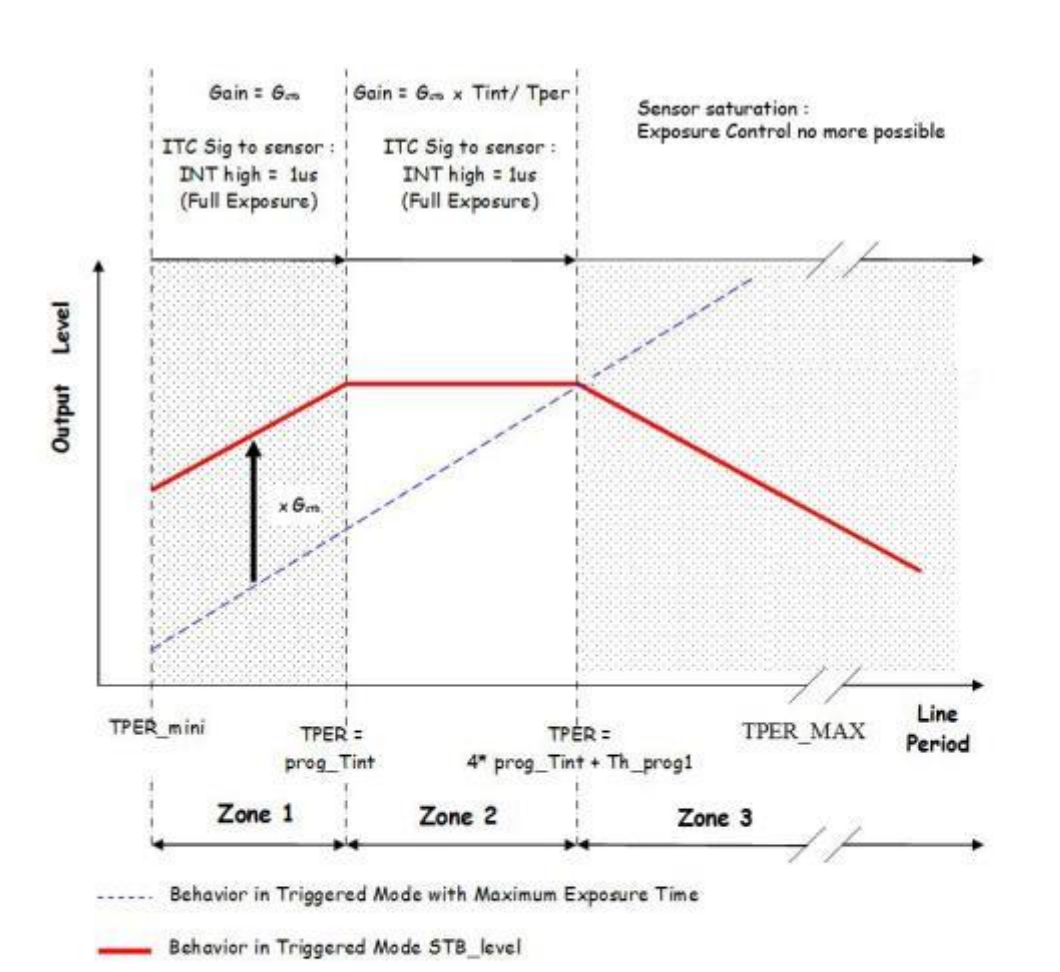
7.3.1 Full Exposure Control Mode

The Full Exposure Control

In Full Definition Enhanced Sensor Mode, the Sensor is working as a double TDI (Time Integration Delay) : The two Top Pixels and the two bottom Pixels are working together in TDI with a delay between their exposure and outputting by the same Memory node and ADC. The summation of the pixels is done in the "charge domain" before the Digital Conversion.

In TDI, control of the exposure is not possible: Only the full Exposure during the Line Period is possible. In order to allow the User to control the exposure in this Sensor mode (Synchronization Modes 1 and 3, described in the Acquisition control chapter), The ELIIXA+ Camera implement a "Full

Exposure Control Mode" :



When the User selects a synchronization mode which requires the control of the exposure, the camera enters a specific mode:
The Line Period (measured) is **Tper**, its minimum value is **TPer$_{mini}$** (10µs on this camera) and the exposure time set by the User is **Prog_Tint.**

<u>Zone 1</u> **: If Tper < Prog_Tint**
Not relevant. **Prog_Tint** has to be smaller than **Tper.** The Camera can't be use in that area

<u>Zone 2</u> **: If Tper < 4 x Max (TPer$_{mini}$, Prog_Tint) + 10µs**
The Sensor works in Full Exposure during the whole Line Period (LP) and the gain applied on the output is variable (max x 4), set by User = **G$_{stb}$**
The Output is multiplied by the following Gain = **G$_{stb}$ x Prog_Tint / Tper**

<u>Zone 3</u> **: If Tper >= 4 x Max (TPer$_{mini}$, Prog_Tint) + 10µs**
The risk is the sensor saturates then the calculation above is no more valid : after showing an incorrect white balance, the image level will decrease down to 0 as the Line Period is increasing.
=> **The Camera can't be use in that area**

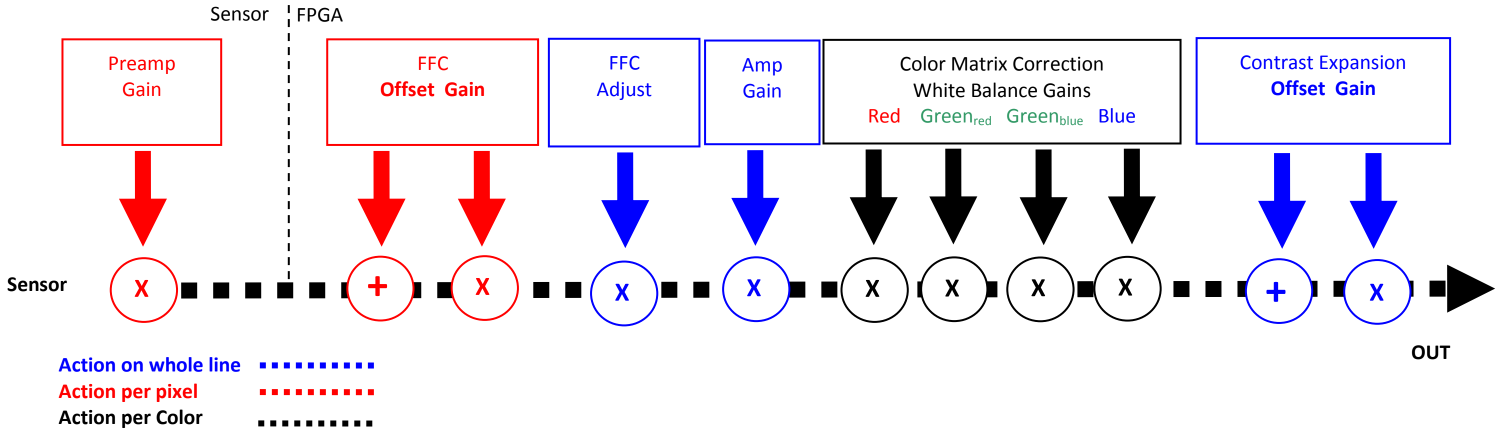
<u>Gain for the "Full Exposure Control Mode"</u> (See "Gain & Offset" Section below)

G$_{stb}$: The User Can set this Gain with a value up to x4 (Gain Section). The value recommended is the one which allows to cover the variation of the line period : 10% of variation requires a Gain at least of x1.2 (+/- 10%).
By default this value is set at x4.

7.3.2 Command Table

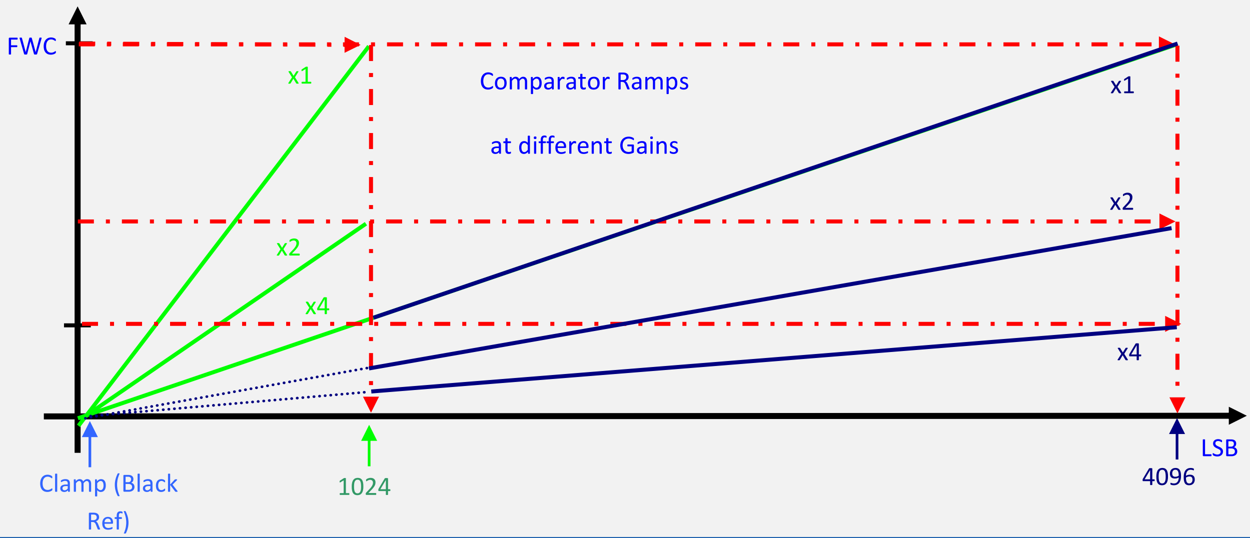
Feature	Commands	Description
LinePeriod	r tper	Get current line period
	w tper <val>	Set line period, from from 150 (15µs) to 65535 (6553,5µs), step 1 (0,1µs)
LinePeriodMin	r tpmi	Get current line period min (15…65535 step 0,1µs)
AcquisitionLineRate	Xml Virtual	= 1 / LinePeriod en Hertz
ExposureTime	r tint	Get exposure time
	w tint <val>	Set exposure time, from 1 (0,1µs) to 65535 (6553,5µs), step 1 (0,1µs)
TriggerPreset	r sync	Get trigger preset mode
	w sync 0	Set trigger preset mode to Free run timed mode, with exposure time and line period programmable. Not available in FDE sensor mode.
	w sync 1	Set trigger preset mode to Triggered mode with exposure time settings
	w sync 2	Set trigger preset mode to Triggered mode with maximum exposure time
	w sync 3	Set trigger preset mode to Triggered mode with exposure time controlled by one signal
	w sync 4	Set trigger preset mode to Triggered mode with exposure time controlled by two signals. Not available in FDE sensor mode.
	w sync 5	Set trigger preset mode to Free run mode, with max exposure time and programmable line period
TriggerTooSlow	r tgts	Read Trigger too slow value (in ms)
	w tgts <Val>	Write Trigger too slow value from 1 to 5368 ms (step 1ms)

7.4 Gain and Offset



Analog Gain in the ADC

The only analog Gain available in the ELIIXA+ is located at the sensor level, in the ADC converter.
This “Preamp Gain” is in fact a variation of the ramp of the comparator of the ADC.
Then 3 Values are available : x1, x2 and x4. A gain x1 in a 12 bits conversion is equivalent to x4 in 10 bits.



- **Preamp Gain** : (*Gain* with *GainSelector= AnalogAll*)
  Set the Pre-amplification Gain**.** This command is available in the CommCam “Gain & Offset” section.
  ⇨ Read function : “**r pamp**”;
  Returned by the camera : Integer corresponding to one of the 3 different step values :
    - 0 : x1 (0dB)
    - 1 : x2 (6dB)
    - 2 : x4 (12dB)

  ⇨ Write function : “**w pamp**” <int> ;

- **Gain:** (*Gain* with *GainSelector= GainAll*)
  Set the Amplification Gain**.** This command is available in the CommCam “Gain & Offset” section :

⇨ Read function : "**r gain**";
Returned by the camera : Value from 0 to 6193 corresponding to a Gain range of 0dB to +8dB calculated as following : Gain(dB) = 20.log(1+ Gain/4096).
⇨ Write function : "**w gain**" <int> ;

- **Tap Gain (*Gain* with *GainSelector*=*Tap*X) :**
  - ⇨ Read function : "r **fga**<tap>"; <tap> is 1 or 2
    Returns the Gain value for the tap. Ex : "*r fga1*" returns Gain value Tap1.
  - ⇨ Write function : "w **fga**<tap> <value>"
    - **<tap>** : 1 or 2
    - **<value> :** from -128 to +127 by step of 1 (0,0021dB each step)

- **Digital Gain** (*Gain* with *GainSelector=DigitalAll*) : Set the global Digital Gain**.** This command is available in the CommCam "Gain & Offset" section :
  - ⇨ Read function : "**r gdig**";
    Returned by the camera : Integer value from 0 to 255. The corresponding Gain is calculated as 20log(1+val/64) in dB
  - ⇨ Write function : "**w gdig**" <int> ;

- **Digital Offset** (*BlackLevelRaw* with *BlackLevelSelector=All*) : Set the global Digital Offset**.** This command is available in the CommCam "Gain & Offset" section :
  - ⇨ Read function : "**r offs**";
    Returned by the camera : Value from –4096 to +4095 in LSB
  - ⇨ Write function : "**w offs**" <int> ;

- **Tap Balance Gains Enable Switch (*TapBalanceGainEnable*) :**
  - ⇨ Read function : "r **fgae**";
    Returns the Tap Balance Status.
  - ⇨ Write function : "w **fgae** <val>" with <val> : 0 or 1
    - 0 : Disables the Tap Balance Gains
    - 1 : Enables the Tap Balance Gains

- **Full Exposure Control Gain (*StbGain*) :**
  - ⇨ Read function : "r **stbg**";
  - ⇨ Write function : "w **stbg** <val>" with <val> = gain from 16384 (x1) to 65535 (x4) by step of 1/16384

7.4.1 Command Table

Feature	Commands	Description
Preamp Gain (GainSelector= AnalogAll)	r pamp	Get the current Pre-Amp gain
	w pamp <val>	Set pre amplifier gain to: 0 (-12dB), 1 (-6dB), 2 (0dB) (analog gain)
Gain (GainSelector= gainAll)	r gain	Get current Amp gain (Digital)
	w gain <val>	Set gain from 0dB(0) to +8 dB (6193)
Digital Gain (GainSelector=DigitalAll)	r gdig	Get contrast expansion digital gain
	w gdig <val>	Set contrast expansion digital gain from 0 (0 dB) to 255 (+14 dB), step 1 (TBD dB)
Offset (BlackLevelRaw) BlackLevelSelector=All	r offs	Get common black level.
	w offs <val>	Set common black from -4096 to 4095, step 1
Digital Tap Gain (GainAbs) GainSelector=DigitalTap<j>	r fga<j> <val>	Get tap<j> digital gain. Dynamically updated on AnalogAll gain changes

Feature	Commands	Description
	w fga<j> <val>	Set tap<j> digital gain from -128 to 127 by step 1 (0.0021dB). Dynamically updated on AnalogAll gain changes
TapBalanceGainEnable	r fgae	Get the status of the Tap balance
	w fgae <val>	Enables the Tap Balance : 0 : Disables the Tap Balance Gains 1 : Enables the Tap Balance Gains
Full Exposure Control gain	r stbg	Read stabilized gain
	w stbg <val>	Write Stabilized gain from 16384 (x1) to 65535 (x4) step 1/16384

7.5 Color Management

As described in chapter 6.3.2, the structure of the sensor differentiates Green pixels facing Blue or Red pixels. Then the white balance is associated with 4 color Gains :

- Red Gain
- $\text{Green}_{\text{Red}}$ Gain
- $\text{Green}_{\text{Blue}}$ Gain
- Blue Gain

The Color Selection or enabling (Image Format Chapter) can affect the way you're performing the white balance : For example, if you disable the Blue and the Red color, the "White Balance" will be performed only between the two Green Gains.

The dissociation of Green (blue) and Green (Red) is justified by the possible difference of response of the two types of Green because of their respective neighbor color influence and then the necessity to tune them separately.

As usual, for a perfect White balance, provide to the Camera a non-saturating white (gray) target in the center of the sensor.

The White balance has to be performed after the Flat Field Correction as each color is performing its own FFC with its own reference.

In any case, the best tuning of the Camera Gains is performed from the left to the right of the Gain Chain described above : Preamp Gain first and quarter Gains last (if required).

7.5.1 Interpolation Corrections for True Color Modes

- **Column Interpolation Enable Switch (ColumnInterpolation)** : Enables the Column Interpolation. This command is available in the CommCam "Color Management" section :
  - ⇨ Read function : "r **ccit**";
    Returns the Colum Interpolation Enable Status.
  - ⇨ Write function : "w **ccit** <val>" with <val> : 0 or 1
    - 0 : Disables the Colum Interpolation
    - 1 : Enables the Colum Interpolation

- **Line Interpolation Enable Switch (ColumnInterpolation)** : Enables the Line Interpolation. This command is available in the CommCam "Color Management" section :
  - ⇨ Read function : "r **clit**";
    Returns the Line Interpolation Enable Status.
  - ⇨ Write function : "w **clit** <val>" with <val> : 0 or 1
    - 0 : Disables the Line Interpolation
    - 1 : Enables the Line Interpolation

Line interpolation is only available in True Color Single Mode
No Interpolation available when the output mode of the Camera is "Raw" (8 or 10 Taps)

Colum and Line Interpolation.

Please, refer to chapter §2.2 for a detailed explanation of these two interpolations available for the User.

7.5.2 White Balance & ROI

- **Digital Red Gain** (*Gain* with *GainSelector=DigitalRed*) : Set the Red Gain for the white balance**.** This command is available in the CommCam "Gain & Offset" section :
  - ⇨ Read function : "**r gwbr**";
    Returned by the camera : Integer value from 0 to 1548. The corresponding Gain is calculated as 20.log(1 + <val>/1024) in dB
  - ⇨ Write function : "**w gwbr**" <val> ;

- **Digital Blue Gain** (*Gain* with *GainSelector=DigitalBlue*) : Set the Blue Gain for the white balance**.** This command is available in the CommCam "Gain & Offset" section :
  - ⇨ Read function : "**r gwbb**";
    Returned by the camera : Integer value from 0 to 1548. The corresponding Gain is calculated as 20.log(1 + <val>/1024) in dB
  - ⇨ Write function : "**w gwbb**" <val> ;

- **Digital Green$_{Red}$ Gain** (*Gain* with *GainSelector=DigitalGreenR*) : Set the Green$_{Red}$ Gain for the white balance**.** This command is available in the CommCam "Gain & Offset" section :
  - ⇨ Read function : "**r gwbg**";
    Returned by the camera : Integer value from 0 to 1548. The corresponding Gain is calculated as 20.log(1 + <val>/1024) in dB
  - ⇨ Write function : "**w gwbg**" <val> ;

- **Digital Green$_{Blue}$ Gain** (*Gain* with *GainSelector=DigitalGreenB*) : Set the Green$_{Blue}$ Gain for the white balance**.** This command is available in the CommCam "Gain & Offset" section :
  - ⇨ Read function : "**r gwbj**";
    Returned by the camera : Integer value from 0 to 1548. The corresponding Gain is calculated as 20.log(1 + <val>/1024) in dB
  - ⇨ Write function : "**w gwbj**" <val> ;

- **White Balance Enable Switch (***WhiteBalanceEnable***)** : Enables the White Balance Gains. This command is available in the CommCam "Gain & Offset" section :
  - ⇨ Read function : "r **gwbe**";
    Returns the White Balance Gain Enable Status.
  - ⇨ Write function : "w **gwbe** <val>" with <val> : 0 or 1
    - 0 : Disables the White Balance Gains
    - 1 : Enables the White Balance Gains

- **Auto White Balance Control** (*AutoWhiteBalanceStart*) : Launch or abort of the White Balance process for the RGB Gains calculation**.** This command is available in the CommCam "Gain & Offset" section :
  - ⇨ No Read Function

- ⇨ Write function :
  - "**w awbc 1**" : Launch the White Balance Calibration Process.
  - "**w awbc 0**" : Abort the White Balance Calibration Process.

The White Balance can be computed on a User-defined ROI which is set with the 2 following parameters :

- **White Balance ROI Start (**AutoWhiteBalanceRoiStart**):** set the Start point for the WB ROI.
  - ⇨ Read function : "**r gwbs**".
    Returns the Target value (from 1 to 8191)
  - ⇨ Write function : "**w gwbs <value>**" :
    Value Range [0: Width – AutoWhiteBalanceRoiWidth - 1] by Step of 1

- **White Balance ROI Width (**AutoWhiteBalanceRoiWidth**):** set the Start point for the WB Width.
  - ⇨ Read function : "**r gwbw**".
    Returns the Target value (from 1 to 8191)
  - ⇨ Write function : "**w gwbw <value>**" :
    Range [2: Width – AutoWhiteBalanceRoiStart] by Step of 2

7.5.3 White Balance Adjust

- **White Balance Adjust Enable Switch (**WBAdjust**)** : Enables the White Balance Adjust feature. This command is available in the CommCam "Color Management" section :
  - ⇨ Read function : "r **adje**";
    Returns the White Balance Adjust Enable Status.
  - ⇨ Write function : "w **adje** <val>" with <val> : 0 or 1
    - 0 : Disables White Balance Adjust
    - 1 : Enables White Balance Adjust

- **White Balance Auto Adjust Target Level (**WBAdjustAutoTarget**):** set the value for the User Target.
  - ⇨ Read function : "**r adjt**". Returns the Target value (from 1 to 255)
  - ⇨ Write function : "**w adjt <value>**" : Set the Target Value (in 8bits)

- **White Balance Adjust Gain (**WBAdjustGain**):** set the value for the Adjust Gain
  - ⇨ Read function : "**r adjg**".
    Returns the Target value (from 1 to 8191)
  - ⇨ Write function : "**w adjg <value>**" :
    Set the Target Value (x0.00024 to x1.99976) : 1 + Gain / 8192

White Balance Adjust : A good usage.

When there are several Cameras to set up in a system on a single line, the most difficult is to have a uniform lightning whole along the line.
If each Camera performs its own White Balance then its own Flat field correction, relative to the max of each color line, the result will be a succession of Camera lines at different levels.
=> The White Balance Adjust function allows to set the same target value for all the Cameras in the system and then to get a perfect uniform line whole along the system with a precision of 1 LSB to the Target.
The Maximum correction is x2 the highest value of the line.

7.5.4 RGB Color Matrix

In the "BHx" Models (8k and 4k) the Color RGB Matrix has been included.
This RGB Color Matrix allows to fine adjust the colors to the Light source used.

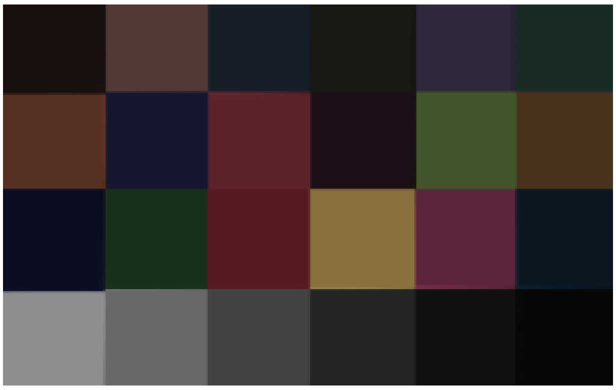
$$(R' \quad G' \quad B') = \begin{pmatrix} C_{RR} & C_{RG} & C_{RB} \\ C_{GR} & C_{GG} & C_{GB} \\ C_{BR} & C_{BG} & C_{BB} \end{pmatrix} \times \begin{pmatrix} R \\ G \\ B \end{pmatrix}$$

Each (RGB) Color Triplet becomes a Linear combination of each color :

$$R' = C_{RR}.R + C_{RG}.G + C_{RB}.B$$
$$G' = C_{GR}.R + C_{GG}.G + C_{GB}.B$$
$$B' = C_{BR}.R + C_{BG}.G + C_{BB}.B$$

An Example of fine adjustment with white LEDs :

Before



After

- **Enable Color Matrix (ColorMatrixEnable)** : Enables the RGB Color Matrix. This command is available in the CommCam "Color Management" section :
  - ⇨ Read function : "r **cmxe**";
    Returns the Color Matrix Enable status.
  - ⇨ Write function : "w **cmxe** <val>" with <val> : 0 or 1
    - 0 : Disables the Color Matrix
    - 1 : Enables the Color Matrix

- **Color Matrix Red$_{Red}$** : Set RGB Matrix Coefficient Red$_{Red}$
  - ⇨ Read function : "**r cmrr**".
    Returns the Coefficient value from -4 (-256) to 4 (255)
  - ⇨ Write function : "**w cmrr <value>**" :
    Set the Coefficient value

- **Color Matrix Red$_{Green}$** : Set RGB Matrix Coefficient Red$_{Green}$
  - ⇨ Read function : "**r cmrg**".
    Returns the Coefficient value from -4 (-256) to 4 (255)
  - ⇨ Write function : "**w cmrg <value>**" :
    Set the Coefficient value

- **Color Matrix Red$_{Blue}$** : Set RGB Matrix Coefficient Red$_{Blue}$
  - ⇨ Read function : "**r cmrb**".
    Returns the Coefficient value from -4 (-256) to 4 (255)
  - ⇨ Write function : "**w cmrb <value>**" :
    Set the Coefficient value

- **Color Matrix Green$_{Red}$** : Set RGB Matrix Coefficient Green$_{Red}$
  - ⇨ Read function : "**r cmgr**".
    Returns the Coefficient value from -4 (-256) to 4 (255)
  - ⇨ Write function : "**w cmgr <value>**" :
    Set the Coefficient value

- **Color Matrix Green$_{Green}$** : Set RGB Matrix Coefficient Green$_{Green}$
  - ⇨ Read function : "**r cmgg**".
    Returns the Coefficient value from -4 (-256) to 4 (255)
  - ⇨ Write function : "**w cmgg <value>**" :
    Set the Coefficient value

- **Color Matrix Green$_{Blue}$** : Set RGB Matrix Coefficient Green$_{Blue}$
  - ⇨ Read function : "**r cmgb**".
    Returns the Coefficient value from -4 (-256) to 4 (255)
  - ⇨ Write function : "**w cmgb <value>**" :
    Set the Coefficient value

- **Color Matrix Blue$_{Red}$** : Set RGB Matrix Coefficient Blue$_{Red}$
  - ⇨ Read function : "**r cmbr**".
    Returns the Coefficient value from -4 (-256) to 4 (255)
  - ⇨ Write function : "**w cmbr <value>**" :
    Set the Coefficient value

- **Color Matrix Blue$_{Green}$** : Set RGB Matrix Coefficient Blue$_{Green}$
  - ⇨ Read function : "**r cmbg**".
    Returns the Coefficient value from -4 (-256) to 4 (255)
  - ⇨ Write function : "**w cmbg <value>**" :
    Set the Coefficient value

- **Color Matrix Blue$_{Blue}$** : Set RGB Matrix Coefficient Blue$_{Blue}$
  - ⇨ Read function : "**r cmbb**".
    Returns the Coefficient value from -4 (-256) to 4 (255)
  - ⇨ Write function : "**w cmbb <value>**" :
    Set the Coefficient value

Color Matrix calculation

Contact the e2v Hotline to help you about the Color Matrix calculation :
hotline-cam@e2v.com

7.5.5 Save & Restore Colour Features

The Colour features of the Camera (white balance and Color Matrix in the same time) can be saved in x4 different User banks.

This function is available in the Colour Management section :

- **Load settings from Bank :** Allows to restore the Camera settings.
  - ⇨ Read function : "**r rcmx**" : Get the current Tap Bank in use
  - ⇨ Write function : "**w rcmx <val>**" : Load settings from bank <val> (1 to 4)

- **Save settings to Bank :** Allows to save the Camera settings in User or Integrator Bank
  - ⇨ Write function : "**w scmx <val>**" : Save the current settings in the User bank <val> (1 to 4)

This Color Banks are available only for the "BHx" Models.
For the "BAx" Models, these features are saved in the Configuration setting banks

7.5.6 Command Table

Feature	Commands	Description
Red Gain GainSelector= DigitalRed	r gwbr	Get the Red Gain for the white balance
	w gwbr <val>	Set the Red Gain from 0 (0dB) to 1548 (8dB) : (1 + <val>/1024)
Blue Gain GainSelector= DigitalBlue	r gwbb	Get the Blue Gain for the white balance
	w gwbb <val>	Set the Blue Gain from 0 (0dB) to 1548 (8dB) : (1 + <val>/1024)
Green(Red) Gain GainSelector= DigitalGreenR	r gwbg	Get the Green$_{Red}$ Gain for the white balance
	w gwbg <val>	Set the Green$_{Red}$ Gain from 0 (0dB) to 1548 (8dB) : (1 + <val>/1024)
Green(Blue) Gain GainSelector= DigitalGreenB	r gwbj	Get the Green$_{Blue}$ Gain for the white balance
	w gwbj <val>	Set the Green$_{Blue}$ Gain from 0 (0dB) to 1548 (8dB) : (1 + <val>/1024)
AutoWhiteBalanceStart	w awbc 0	Stops the Auto white Balance calibration process
	w awbc 1	Starts the Auto white Balance calibration process
AutoWhiteBalanceEnable	w gwbe 0	Disables the White Balance
	w gwbe 1	Enables the White Balance
ColumnInterpolation	r ccit	Read Column Interpolation
	w ccit	Write Column Interpolation 0 : disable 1: enable
LineInterpolation	r clit	Read Line Interpolation
	w clit	Write Line Interpolation 0 : disable 1: enable
WBAdjust	r adje	Get WB Adjust enable
	w adje <val>	Write White Balance Adjust Enable 0: Disable 1: Enable
WBAdjustAutoTargetLevel	r adjt	Get Target adjust

Feature	Commands	Description
	w adjt <val>	Set Adjust Target from 1 to 255 step 1
WBAdjustGain	r adjg	Get Adust Gain
	w adjg <val>	Set Adjust Gain from White Balance Ajust Gain Value from 1 to 8191 (x0.00024 to x1.99976)
Color Bank Selector(*)	No Command	Select the Color bank to use
RestoreColorBank(*)	r rcmx	Get the current color Bank
	w rcmx <val>	Restore color Bank <val> bank (from 1 to 4)
SaveColorBank(*)	w scmx <val>	save color Bank into <val> bank (from 1 to 4)
ColorMatrixEnable(*)	r cmxe	Read Color Matrix enable
	w cmxe 0	Enable color matrix
	w cmxe 1	Disable color matrix
ColorMatrix Red(red) (*)	r cmrr	Read Color Matrix coefficient (Red(red))
	w cmrr <val>	Write Color Matrix Coefficient (Red(red) <val> from x(-4)(-256) to x4(255)
ColorMatrix Red(green) (*)	r cmrg	Read Color Matrix coefficient (Red(green))
	w cmrg <val>	Write Color Matrix Coefficient (Red(green) <val> from x(-4)(-256) to x4(255)
ColorMatrix Red(Blue) (*)	r cmrb	Read Color Matrix coefficient (Red(Blue))
	w cmrb <val>	Write Color Matrix Coefficient (Red(Blue) <val> from x(-4)(-256) to x4(255)
ColorMatrix Green(red) (*)	r cmgr	Read Color Matrix coefficient (Green(red))
	w cmgr <val>	Write Color Matrix Coefficient (Green(red) <val> from x(-4)(-256) to x4(255)
ColorMatrix Green(green) (*)	r cmgg	Read Color Matrix coefficient (Green(green))
	w cmgg <val>	Write Color Matrix Coefficient (Green(green) <val> from x(-4)(-256) to x4(255)
ColorMatrix Green(Blue) (*)	r cmgb	Read Color Matrix coefficient (Green(Blue))
	w cmgb <val>	Write Color Matrix Coefficient (Green(Blue) <val> from x(-4)(-256) to x4(255)
ColorMatrix Blue(red) (*)	r cmbr	Read Color Matrix coefficient (Blue(red))
	w cmbr <val>	Write Color Matrix Coefficient (Blue(red) <val> from x(-4)(-256) to x4(255)
ColorMatrix Blue(green) (*)	r cmbg	Read Color Matrix coefficient (Blue(green))
	w cmbg <val>	Write Color Matrix Coefficient (Blue(green) <val> from x(-4)(-256) to x4(255)
ColorMatrix Blue(Blue) (*)	r cmbb	Read Color Matrix coefficient (Blue(Blue))
	w cmbb <val>	Write Color Matrix Coefficient (Blue(Blue) <val> from x(-4)(-256) to x4(255)
AutoWhiteBalanceRoiStart	w gwbs val	Set auto white Balance ROI Start value Range [0: Width – AutoWhiteBalanceRoiWidth - 1] By Step of 1
	r gwbs	Get the auto white Balance ROI Start value
AutoWhiteBalanceRoiWidth	w gwbw val	Set auto white Balance ROI Width value Range [2: Width – AutoWhiteBalanceRoiStart] By Step of 2
	r gwbw	Get the auto white Balance ROI Width value

7.6 Save & Restore Settings

The settings (or Main configuration) of the Camera can be saved in 4 different User banks and one Integrator bank. This setting includes also the FFC enable

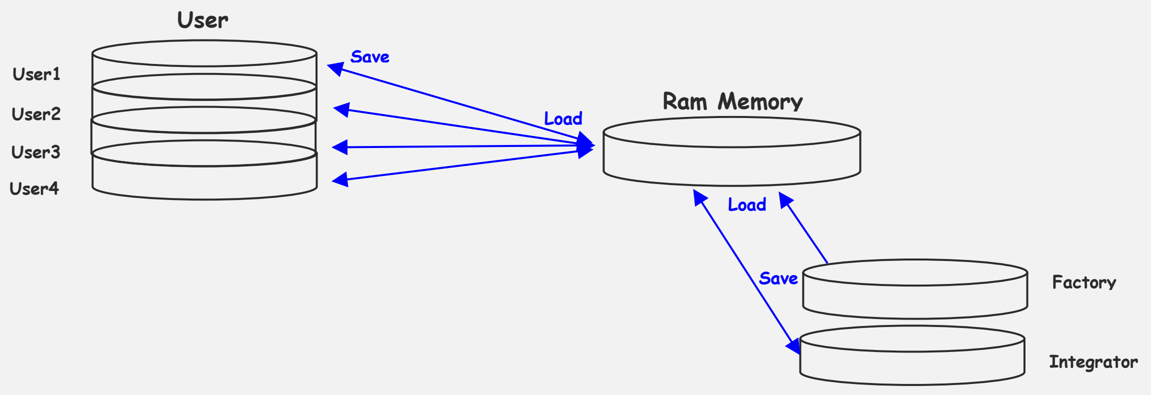
This function is available in the Save & Restore Settings section :

- **Load settings from Bank :** Allows to restore the Camera settings.
  - ⇨ Read function : "**r rcfg**" : Get the current Tap Bank in use
  - ⇨ Write function : "**w rcfg <val>**" : Load settings from bank <val> (0: Factory , 1 to 4 for Users, 5 for Integrator)

- **Save settings to Bank :** Allows to save the Camera settings in User or Integrator Bank
  - ⇨ Write function : "**w scfg <val>**" : Save the current settings in the User bank <val> (1 to 4 for User, 5 for Integrator)

The integrator bank (User Set5) can be written only if the Camera is set in integrator mode (Privilege level = 1). This integrator bank can be used as a « Factory default » by a system integrator.

Configuration Bank Usage



At the power up : Last User Bank used is loaded in RAM
"Integrator" Bank (5) can be locked by switching the Camera in "User" mode (cf : Privilege feature). Then it can't be saved any more without switching back the Camera in "Integrator" Mode.

7.6.1 Command Table

Feature	Commands	Description
UserSetLoad	r rcfg	Get the current user configuration bank (saved or restored)
	w rcfg <val>	Restore current UserSet from UserSet bank number <val>, from 0 to 5; <val> comes from UserSetSelector.
UserSetSave	w scfg <val>	Save current UserSet to UserSet bank number <val>, from 1 to 5; <val> comes from UserSetSelector. 0 cannot be saved. 5 (Integrator) can't be saved in User mode

Feature	Commands	Description
UserSetControl	Xml virtual	
RestoreFFCFromBank	r rffc	Get the current FFC bank (save or restore)
	w rffc <val>	Restore current FFC (including FPN and FFCGain) from FFC bank number <val>, from 1 to 4; <val> comes from UserFFCSelector (XML feature).
SaveFFCToBank	w sffc <val>	Save current FFC (including FPN and FFCGain) to FFC bank number <val>, from 1 to 4; <val> comes from FFCSelector (XML feature).

7.7 Flat Field Correction

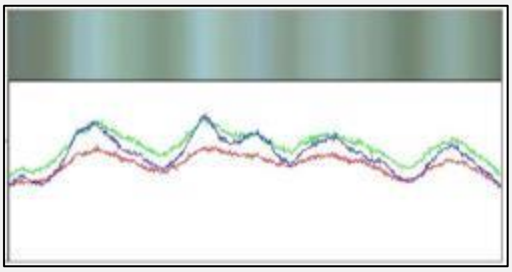
How is performed the Flat Field Correction ?

What is the Flat Field correction (FFC) ?

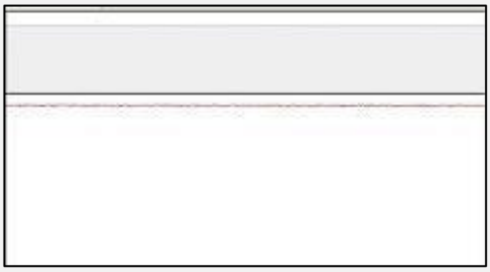
The Flat Field Correction is a digital correction on each pixel which allows :

- To correct the Pixel PRNU (Pixel Response Non Uniformity) and DSNU (Dark Signal Non Uniformity)
- To Correct the shading due to the lens
- To correct the Light source non uniformity

Before



After



How is calculated / Applied the FFC ?

The FFC is a digital correction on the pixel level for both Gain and Offset.

Each Pixel is corrected with :

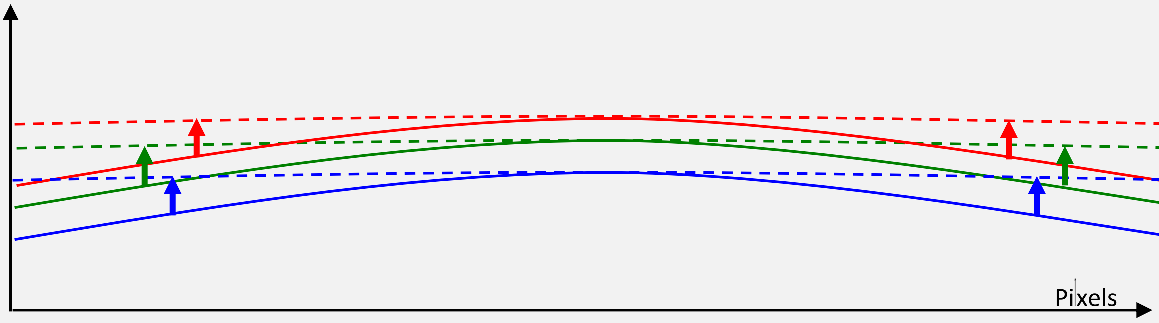
- An Offset on 10 bits (Signed Int S9.1). They cover a dynamic of ±256LSB in 12bits with a resolution of 1/2 LSB 12bits. Offet : the MSB is the sign, the rest of 9bits is from 0 .. 256 with precision of 1/2
- A Gain on 12 bits (Unsigned Int U2.12) with a max gain value of **x4.999**

  The calculation of the new pixel value is : **P' = (P + Off).(1 + Gain/1024).** Gain : 0 to 4095

The FFC processing can be completed with an automatic adjustment to a global target. This function is designed as "**FFC Adjust**". This adjustment to a User target is done by an internal hidden gain which is re-calculated each time the FFC is processed while the FFC adjust function is enabled.

The FFC is always processed with the max pixel value of the line as reference. If enabled, the FFC adjust module (located at the output of the FFC module) calculates the adjustment gain to reach the target defined by the User.

When the FFC result is saved in memory, the adjust gain and target are saved in the same time in order to associate this gain value with the FFC result.



Standard FFC computed on the max of the line for each color ($Green_{Blue}$ and $Green_{Red}$ are treated separately). Then the White Balance will overlay the colors

How to perform the Flat Field Correction ?

FPN/DSNU Calibration

- ⇨ Cover the lens
- ⇨ Launch the FPN Calibration : Grab and calculation is performed in few seconds

PRNU Calibration

The User must propose a white/grey uniform target to the Camera (not a fixed paper).
The Gain/Light conditions must give a non saturated image in any Line.
The Camera must be set in the final conditions of Light/ Gain and in the final position in the System.
I f required, set a user target for the FFC adjust and enable it.

- ⇨ White uniform (moving) target.
- ⇨ Use The FFC Low Band Filter if the Target can't move. This will remove the defects of the target itself
- ⇨ Enable and Set your White Balance Target is necessary
- ⇨ Launch the FFC
- ⇨ Enable the FFC
- ⇨ You can save the FFC result (both FPN+PRNU in the same time) in one of the 4 x FFC User Banks.
- ⇨ The user target and Gain are saved with the associated FFC in the same memory.
- ⇨ Remove the FFC Low Band filter (set to 0) if used during the Process.

Advices

The UNIIQA+ Cameras have 4 x FFC Banks to save 4 x different FFC calibrations. You can use this feature if your system needs some different conditions of lightning and/or Gain because of the inspection of different objects : You can perform one FFC to be associated with one condition of Gain/setting of the Camera (4 Max) and recall one of the four global settings (Camera Configuration + FFC + Line Quarters Balance) when required.

7.7.1 FFC Activation and Low Band Filter

- **FFC Activation** (*FFCEnable*) : Enable/disable the Flat Field Correction**.** This command is available in the CommCam "Flat Field Correction" section :
  - ➪ Read function : "**r ffcp**" : Returns the FFC Status (0 if disabled, 1 if enabled)
  - ➪ Write function :
    - "**w ffcp 1**" : Enable the FFC.
    - "**w ffcp 0**" : Disabled the FFC

- **FFC Low Band Filter** (*FFCAutoTargetLevel*)**:** set the value for the User Target.
  - ➪ Read function : "**r lffw**". Returns the Filter Interval size (from 0 to 255)
  - ➪ Write function : "**w lffw <value>**" : Set the Interval size for the filter (0 / 1 … 255)
    - 0 : Disables the FFC Low Band Filter
    - 1 to 255 : Set the interval size (+/- the value around the pixel) for the Low Band filter

When you can't provide a moving Target to the Camera during the PRNU Calibration you can setup the FFC Low Band Filter in order to remove the defect from the Target before calculating the FFC parameters. The Value set in the FFC filter defined the size of the interval around each pixel : The Filter will replace each pixel value by the average on the interval.

7.7.2 Automatic Calibration

FPN/DSNU Calibration :

- **FPN Calibration Control** (*FPNCalibrationCtrl*) : Launch or abort of the FPN process for the Offsets calculation. These commands are available in the CommCam "Flat Field Correction / Automatic Calibration " section :
  - ➪ Read function : "**r calo**" : Returns the FPN Calculation Process Status (0 if finished, 1 if processing)
  - ➪ Write function :
    - "**w calo 1**" : Launch the FPN Calibration Process.
    - "**w calo 0**" : Abort the FPN Calibration Process.

- **FPN Coefficient Reset** (*FPNReset*) : Reset the FPN (Offsets) coefficient in Memory. This command is available in the CommCam "Flat Field Correction / Manual Calibration " section :
  - ➪ Write function : "**w rsto 0**" : Reset (set to 0) the FPN coefficients in memory. This doesn't affect the FFC User Memory Bank but only the active coefficients in Memory.

PRNU Calibration :

- **PRNU Calibration Control** (*FFCCalibrationCtrl*) : Launch or abort of the PRNU process for the Gains calculation. This command is available in the CommCam "Flat Field Correction / Automatic Calibration " section :
  - ➪ Read function : "**r calg**" : Returns the PRNU Calculation Process Status (0 if finished, 1 if processing)
  - ➪ Write function :
    - "**w calg 1**" : Launch the PRNU Calibration Process.
    - "**w calg 0**" : Abort the PRNU Calibration Process.

- **PRNU coefficient Reset** (*PRNUReset*) : Reset the PRNU (Gains) coefficient in Memory. This command is available in the CommCam "Flat Field Correction / Manual Calibration " section :
  - ➪ Write function **: "w rstg 0" :** Reset (set to "x1") the PRNU coefficients in memory. This doesn't affect the FFC User Memory Bank but only the active coefficients in Memory.

Some Warnings can be issued from the PRNU/FPN Calibration Process as “pixel Overflow” of “Pixel Underflow” because some pixels have been detected as too high or too low in the source image to be corrected efficiently.
The Calculation result will be proposed anyway as it’s just a warning message.
The Status Register is the changed and displayed in CommCam “Status” section :
Register status is detailed chap §6.3.3.

7.7.3 Manual Flat Field Correction

The FFC Coefficients can also be processed outside of the Camera or changed manually by accessing directly their values in the Camera : This is the “Manual” FFC.

In CommCam, the User can access to a specific interface by clicking on “click for extended control” in both “Manual FFC calibration” and “Manual FPN calibration sections” :

This will allow the user to upload/download out/in the Camera the FFC coefficients in/from a binary or text file that can be processed externally.

It is recommended to setup the baud rate at the maximum value possible (115000 for example) otherwise the transfer can take a long time.

- **FPN coefficients modification :** Direct access to the FPN coefficients for reading or writing**.**
  The FPN coefficients are read packets of x128 coefficients : Format: S9.1 => -256..+255.5 step ½
  - ⇨ Read function : “**r ffco <addr>**” : Read 128 consecutive FPN user coefficients starting from **<addr>** address. Returned value is in hexadecimal, without space between values (one unsigned short per coefficient).
    - Coefficient from address 0 to 4095 are for red pixels
    - Coefficient from address 4096 to 8191 are for blue pixels
    - Coefficient from address 8192 to 12287 are for greenRed pixels
    - Coefficient from address 12288 to 163837 are for greenBlue pixels
  - ⇨ Write function :” **w ffco <addr><val>** : Write 128 consecutive FPN user coefficients starting from the **<addr>** address. **<val>** is the concatenation of individual FPN values, without space between the values (one unsigned short per coefficient).

- **PRNU coefficients modification :** Direct access to the PRNU coefficients for reading or writing**.** The PRNU coefficients are read packets of x128 coefficients. Format : U2.12 (1+coeff/1024) => x1 to x3.999 by step of 1/1024
  - ⇨ Read function : "**r ffcg <addr>**" : Read 128 consecutive PRNU user coefficients starting from **<addr>** address. Returned value is in hexadecimal, without space between values (one unsigned short per coefficient).
    - - Coefficient from address 0 to 4095 are for red pixels
    - - Coefficient from address 4096 to 8191 are for blue pixels
    - - Coefficient from address 8192 to 12287 are for greenRed pixels
    - - Coefficient from address 12288 to 163837 are for greenBlue pixels
  - ⇨ Write function :" **w ffcg <addr><val>** : Write 128 consecutive PRNU user coefficients starting from the **<addr>** address. **<val>** is the concatenation of individual PRNU values, without space between the values (one unsigned short per coefficient).

7.7.4 FFC User Bank Management

The new-processed FFC values can be saved or restored in/from 4 x User banks.

Both Gains and Offsets in the same time but also the FFC Adjust User target and associated gain.

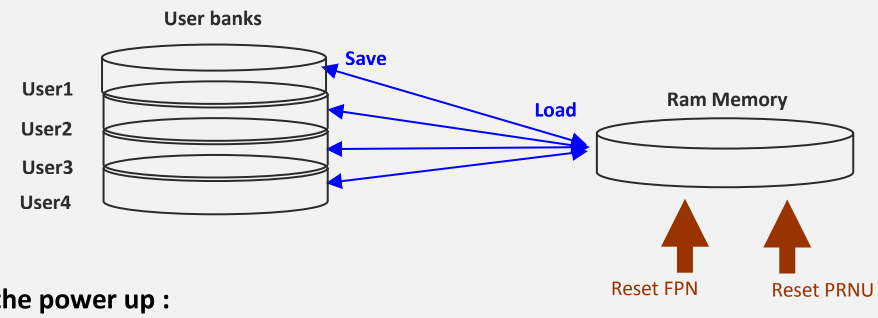
These functions are available in the Flat Field correction/Save & Restore FFC section :

- **Restore FFC from Bank** (*RestoreFFCFromBank*) : Restore the FFC from a Bank in the current FFC.
  ⇨ Read function : "**r rffc**" : Get the current FFC Bank used
  Returned by the camera : 0 for Factory bank or 1 to 4 for User banks
  ⇨ Write function : "**w rffc <val>**" : Bank <val> 1 to 4 for User banks

  Note : Factory means neutral FFC (no correction).

- **Save FFC in User Bank** (*SaveFFCToBank*) : Save current FFC in User Bank
  ⇨ Can not de read
  ⇨ Write function : "**w sffc <val>**" : User bank <val> if from 1 to 4.

FFC User Bank Usage



At the power up :
- Last User Bank used is loaded in RAM
Reset a User bank :
- Reset the RAM (FPN/PRNU individually)
- Save in the bank to reset

7.7.5 Command Table

Feature	Commands	Description
FFCEnable	r ffcp	Get Flat Field Correction processing status
	w ffcp 0	Disable Flat Field Correction (“False”)
	w ffcp 1	Enable Flat Field Correction (“True”)
FPNReset	w rsto 0	Reset FPN coefficients
PRNUReset	w rstg 0	Reset PRNU coefficients
No direct feature	r ffco <addr>	Read 128 Fpn coefficients starting from address <addr>. Return value is in hexadecimal, without space between values (one unsigned short per coef). Format: S9.1 => -256 to +255.5 by step of 1/2
	w ffco <addr> <val>	Write 128 Fpn coefficients (straight to FPGA) starting from address <addr>. <val> is the concatenation of individual Fpnvalue, without space between values. - Coefficient from address 0 to 4095 are for red pixels - Coefficient from address 4096 to 8191 are for blue pixels - Coefficient from address 8192 to 12287 are for green$_{Red}$ pixels - Coefficient from address 12288 to 163383 are for green$_{Blue}$ pixels
No direct feature	r ffcg <addr>	Read 128 Prnu coefficients (straight from FPGA) starting from address <addr>. Return value is in hexadecimal, without space between values. (one unsigned short per coef) U2.12 (1+coeff/1024) => x1 to x4.999 by step of 1/1024
	w ffcg <addr> <val>	Write 128 Prnu coefficients (straight to FPGA) starting from address <addr>. <val> is the concatenation of individual PRNUvalue, without space between values. - Coefficient from address 0 to 4095 are for red pixels - Coefficient from address 4096 to 8191 are for blue pixels - Coefficient from address 8192 to 12287 are for green$_{Red}$ pixels - Coefficient from address 12288 to 163837 are for green$_{Blue}$ pixels
FFCCalibrationCtrl	r calg	Get the PRNU calibration status
	w calg 0	Abort PRNU calibration by setting it to “Off” (no effect if already stopped)
	w calg 1	Launch PRNU calibration by setting it to “Once” (no effect if already launched)
PrnuCalibrationCtrl	r calo	Get the fpn calibration status
	w calo 0	Abort fpn calibration by setting it to “Off” (no effect if already stopped)
	w calo 1	Launch fpn calibration by setting it to “Once” (no effect if already launched)
Low Frequency Filter Width	r lffw	
	w lffw	Configure windows (width) around the pixel (+/- val) 0 : filter is disable 1-255 : nb pixels around the pixel to filter
RestoreFFCFromBank	r rffc	Get the current FFC bank (save or restore)
	w rffc <val>	Restore current FFC (including FPN and FFCGain) from FFC bank number <val>, from 1 to 4; <val> comes from UserFFCSelector (XML feature).
SaveFFCToBank	w sffc <val>	Save current FFC (including FPN and FFCGain) to FFC bank number <val>, from 1 to 4; <val> comes from FFCSelector (XML feature).

7.8 Privilege Level

There are 3 privilege levels for the camera :

- Factory (0) : Reserved for the Factory
- Integrator (1) : Reserved for system integrators
- User (2) : For all Users.

The Cameras are delivered in Integrator mode. They can be locked in User mode and a specific password is required to switch back the Camera in Integrator mode. This password can be generated with a specific tool available from the hotline (hotline-cam@e2v.com)

This function is available in the Privilege section :

- **Privilege level Management** (*PrivilegeLevel*) : Get the current Camera privilege level..
  - ⇨ Read function : "**r lock**" : Get the current privilege
    Returned by the camera : 0 to 2
  - ⇨ Write function : "**w lock <val>**" : <val> is as follow
    - **2** : Lock the Camera in Integrator or "privilege User"
    - **<computed value**> : Unlock the Camera back in Integrator mode

7.8.1 Command Table

Feature	Commands	Description
PrivilegeLevel	r lock	Get camera running privilege level 0 = Privilege Factory 1 = Privilege Advanced User 2 = Privilege User
ChangePrivilegeLevel	w lock 1	Lock camera privilege to "Advanced User"
	w lock 2	Lock camera privilege to "User"
	w lock <val>	Unlock camera privilege depending on <val> (min=256; max=2^{32}-1)

7.9 Image Control

- **Save Image** (*SaveImageControl*) : control the recording of the image in the camera
  This command is available in the CommCam "Image Control" section :

⇨ Read function (ASCII): "**r recl**";
Returned by the camera :
  - "0" : No Record in Progress
  - "1" : Record in Progress

⇨ Write function (ASCII): "**w recl**" <value> :
  - "0" : Stop Recording
  - "1" : Start Recording

- **Play Image** (*PlayImageControl*) : control the Replay of the image in the camera
  This command is available in the CommCam "Image Control" section :

⇨ Read function (ASCII): "**r play**";
Returned by the camera :
  - "0" : Camera display the "Live Image"
  - "1" : Camera display recorded Image

⇨ Write function (ASCII): "**w play**" <value> :
  - "0" : Start Playing Live Image
  - "1" : Start Playing Recorded Image

This Function is available starting with the version 1.5.0 (BA0) or 1.1.0 (BH0) of the Firmware

7.9.1 Command Table

Feature	Commands	Description
Save Image	w recl <val>	Record current Image : 0 : Abort recording 1 : record active image
Play Image	r play <val>	Read output status : 0 : Camera output live image 1 : Camera output recorded image
	w play <val>	Camera Output selection : 0 : Output live image 1 : Output recorded image

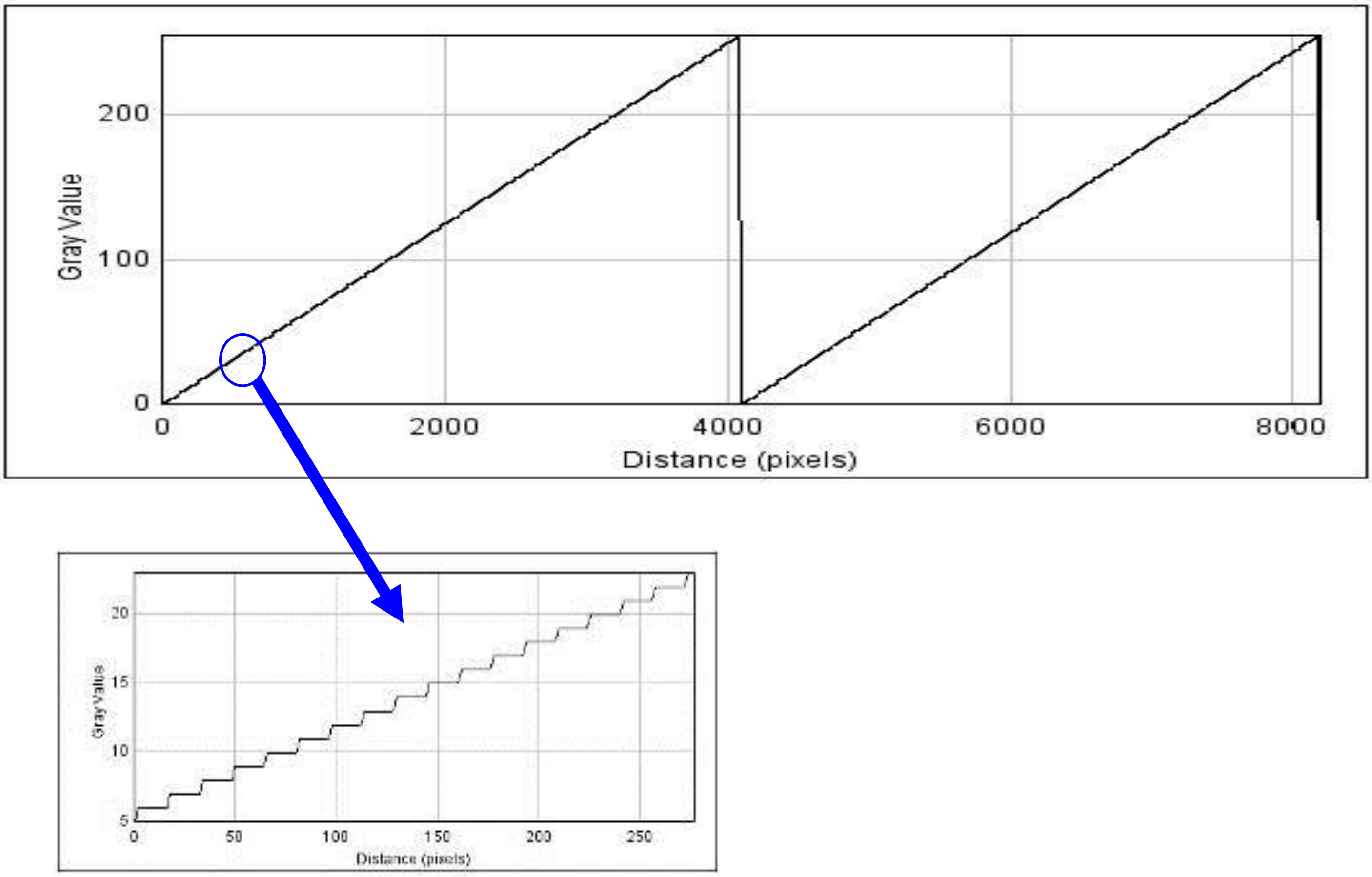
APPENDIX

Appendix A. Test Patterns

A.1 Test Pattern 1: Vertical wave

The Test pattern 1 is a vertical moving wave : each new line will increment of 1 gray level in regards with the previous one : level reaches 255 before switching down to 0

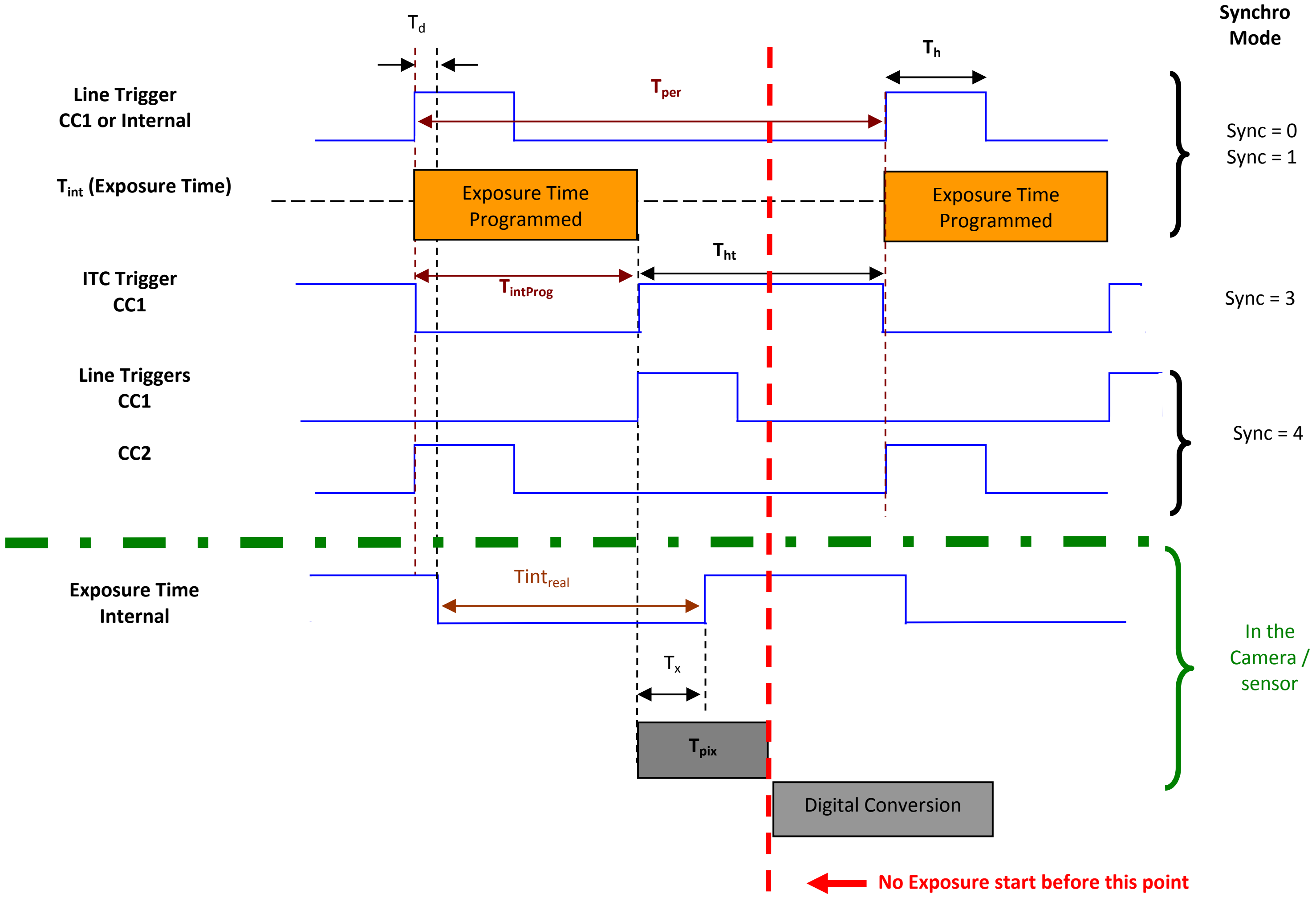
A.2 Test Pattern 2: Fixed Horizontal Ramps



Starting at 0, an increment of 1 LSB is made every 16 pixels. When it reaches 255, turns back to 0 and starts again.

Appendix B. Timing Diagrams

B.1 Synchronization Modes with Variable Exposure Time



T_{pix} : Timing Pixel. During this uncompressible period, the pixel and its black reference are read out to the Digital converter. During the first half of this timing pixel (read out of the black reference), we can consider that the exposure is still active.

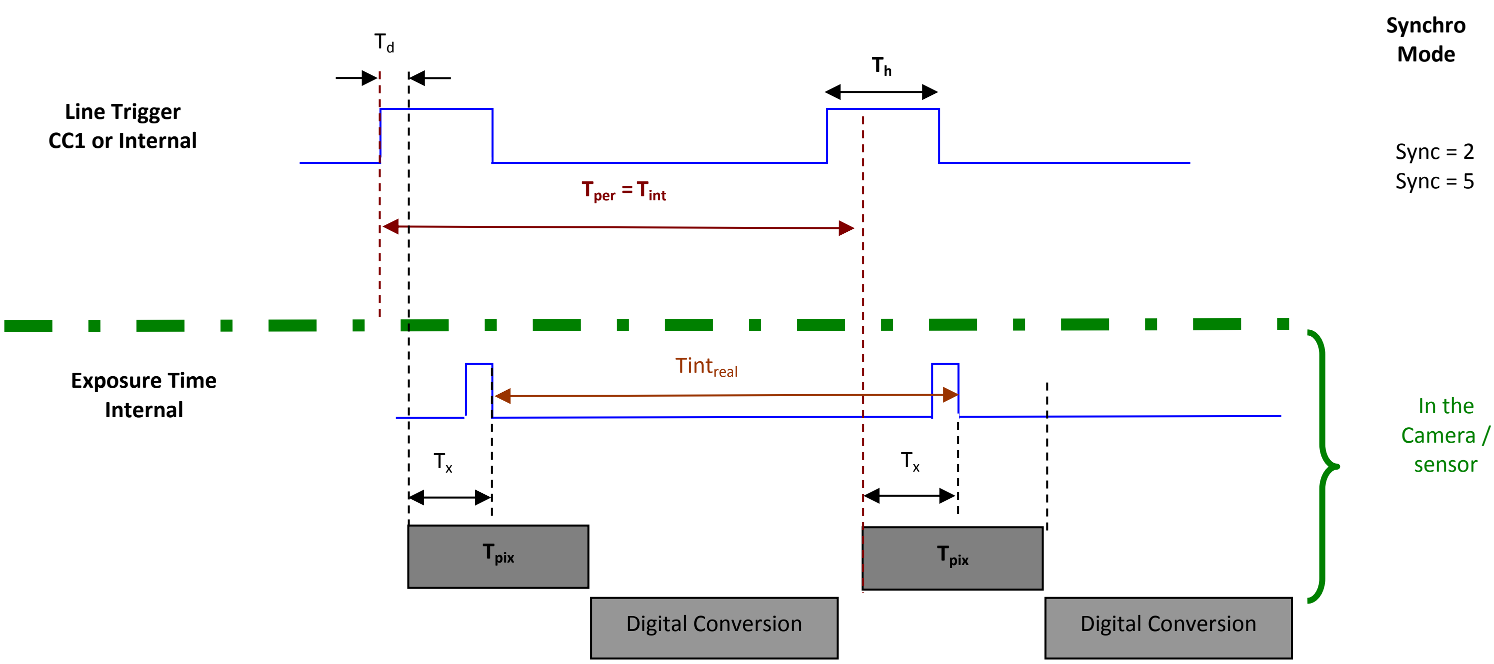
Digital Conversion : During the conversion, the analog Gain is applied by the gradient of the counting ramp (see next chapter : Gain & Offset). The conversion time depends is **6µs** (in 8bits per color/Pixel)

This conversion is done in masked time, eventually during the next exposure period.

T_d : Delay between the Start exposure required and the real start of the exposure.

If T_{per} is the Line Period (internal or external coming from the Trigger line), in order to respect this line Period, the Exposure Time as to be set by respecting : $T_{int} + T_{pix} <= T_{per}$
*Then, the real exposure time is : **$Tint_{real}$** $= T_{int} + T_x - T_d$.*
In the same way, The high level period of the Trig signal in sync=3 mode, $T_{ht} >= T_{pix}$
For a Line Period of *LinePer*, the maximum exposure time possible without reduction of line rate is : **$Tint_{max}$** $= T_{per} - T_{pix}$ (T_{pix} is defined above) but the effective Exposure Time will be about **$Tint_{real}$** $= T_{int} + T_x - T_d$.

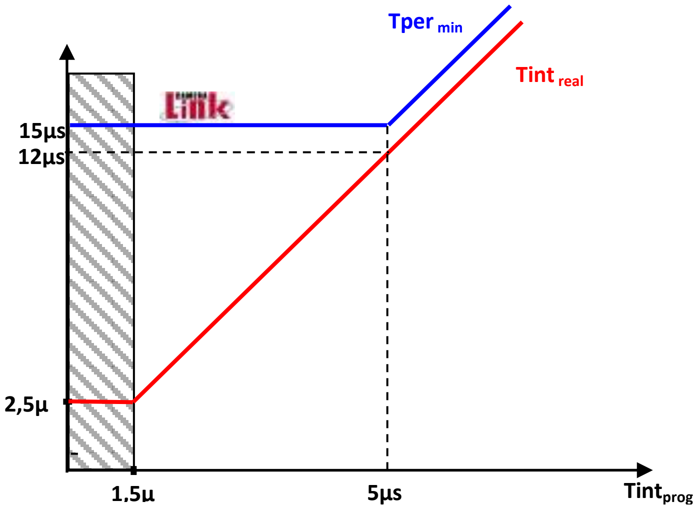
B.2 Synchronization Modes with Maximum Exposure Time



In these modes, the rising edge of the Trigger (internal or External) starts the readout process (T_{pix}) of the previous integration. The Real exposure time ($Tint_{real}$) is finally equal to the Line Period (***T_{per}***) even if it's delayed from (***$T_x + T_d$***) from the rising edge of the incoming Line Trigger.

B.3 Timing Values

Label	Min	Unit
T_{pix}	5	µs
T_x	3,1	µs
T_h	0,120	µs
T_{ht}	T_{pix}	µsec
T_d	1.1	µs

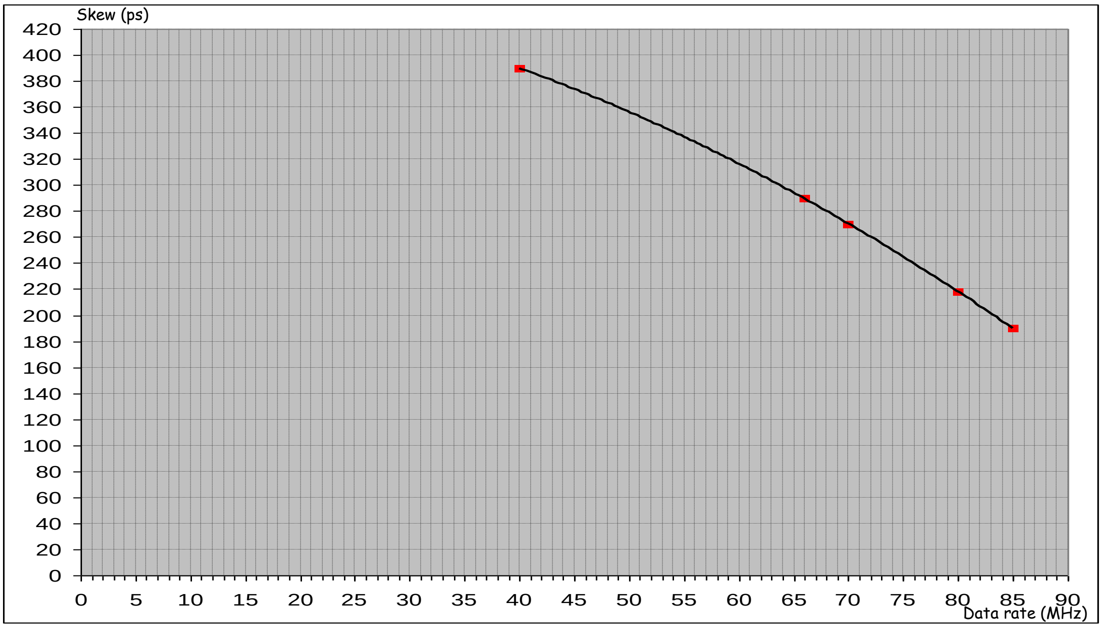


Appendix C. CameraLink Data Cables

C.1 Choosing the Cable

You may check the compliance of your CameraLink cables with the transportation of the 85MHz data rate.
The main parameter to be checked in the cable specification is the skew (in picoseconds)
This parameter is given for a dedicated maximum value per meter of cable (as max : 50ps/m)

The CameraLink Standards defines the maximum total skew possible for each data rate :



Here is a following example of cable and the cable length limitation in accordance with the standard :

Conductor Size:	28 AWG Stranded
Propogation Velocity:	1.25 ns/ft [4.1 ns/m]
Skew (within pair):	50 ps/meter maximum
Skew (channel skew per chipset):	50 ps/meter maximum

DataRate	Skew	Cable Length
40Mhz	390ps	7,8m
66MHz	290ps	5,8m
70MHz	270ps	5,4m
80MHz	218ps	4,36m
85MHz	190ps	3,8m

C.2 Choosing the Data Rate

Maximum Line Rates tables versus Data rate and Pixel Format

Data Frequency : 85MHz								
Sensor Mode	**Base : 3x8bits**		**Dual Base : 2x 3x8bits**		**Full 8x8bits**		**Full+ : 10x8bits**	
	Line Rate Max (kHz)	Tper Min (µs)	Line Rate Max (kHz)	Tper Min (µs)	Line Rate Max (kHz)	Tper Min (µs)	Line Rate Max (kHz)	Tper Min (µs)
True Color (4k 10µm)	20	50	40	25	52.9	18.9	66.2	15.1
Full Def. (8K 5µm)	10	100	20	50	40	25	50	20

Data Frequency : 80MHz								
Sensor Mode	**Base : 3x8bits**		**Dual Base : 2x 3x8bits**		**Full 8x8bits**		**Full+ : 10x8bits**	
	Line Rate Max (kHz)	Tper Min (µs)	Line Rate Max (kHz)	Tper Min (µs)	Line Rate Max (kHz)	Tper Min (µs)	Line Rate Max (kHz)	Tper Min (µs)
True Color (4k 10µm)	18.8	53.2	37.6	26.6	49.8	20.1	62.1	16.1
Full Def. (8K 5µm)	9.4	106.3	18.8	53.2	37.6	26.6	46.9	21.3

Data Frequency : 75MHz								
Sensor Mode	**Base : 3x8bits**		**Dual Base : 2x 3x8bits**		**Full 8x8bits**		**Full+ : 10x8bits**	
	Line Rate Max (kHz)	Tper Min (µs)	Line Rate Max (kHz)	Tper Min (µs)	Line Rate Max (kHz)	Tper Min (µs)	Line Rate Max (kHz)	Tper Min (µs)
True Color (4k 10µm)	17.6	56.7	35.2	28.4	46.5	21.5	58.1	17.2
Full Def. (8K 5µm)	8.8	113.4	17.6	56.7	26	38.4	44	22.7

Data Frequency : 70MHz								
Sensor Mode	**Base : 3x8bits**		**Dual Base : 2x 3x8bits**		**Full 8x8bits**		**Full+ : 10x8bits**	
	Line Rate Max (kHz)	Tper Min (µs)	Line Rate Max (kHz)	Tper Min (µs)	Line Rate Max (kHz)	Tper Min (µs)	Line Rate Max (kHz)	Tper Min (µs)
True Color (4k 10µm)	16.4	60.8	32.9	30.4	43.5	23	54.3	18.4
Full Def. (8K 5µm)	8.2	121.5	16.4	60.8	32.9	30.4	41.1	24.3

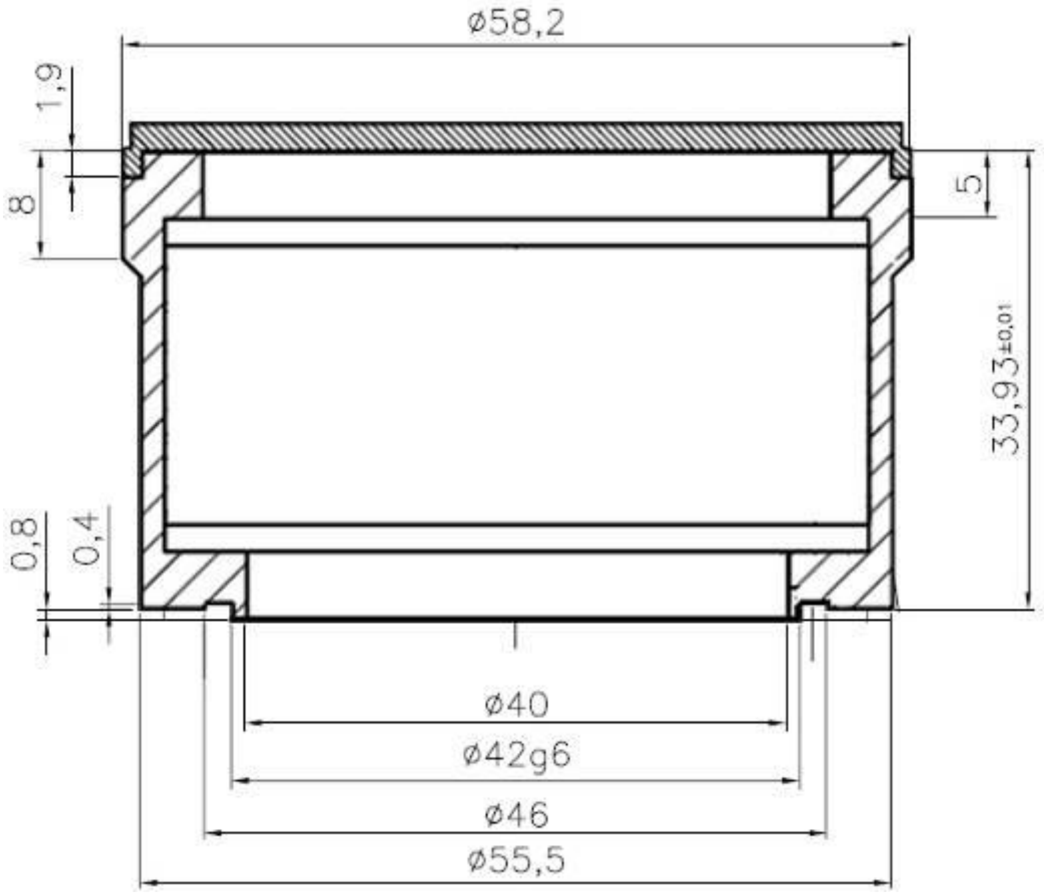
Data Frequency : 65MHz								
Sensor Mode	**Base : 3x8bits**		**Dual Base : 2x 3x8bits**		**Full 8x8bits**		**Full+ : 10x8bits**	
	Line Rate Max (kHz)	Tper Min (µs)	Line Rate Max (kHz)	Tper Min (µs)	Line Rate Max (kHz)	Tper Min (µs)	Line Rate Max (kHz)	Tper Min (µs)
True Color (4k 10µm)	15.3	65.4	30.5	32.7	40.3	24.8	50.5	19.8
Full Def. (8K 5µm)	7.6	130.8	15.3	65.4	30.5	32.7	38.1	26.2

Data Frequency : 60MHz								
Sensor Mode	**Base : 3x8bits**		**Dual Base : 2x 3x8bits**		**Full 8x8bits**		**Full+ : 10x8bits**	
	Line Rate Max (kHz)	Tper Min (µs)	Line Rate Max (kHz)	Tper Min (µs)	Line Rate Max (kHz)	Tper Min (µs)	Line Rate Max (kHz)	Tper Min (µs)
True Color (4k 10µm)	14.1	70.9	28.1	35.5	37.3	26.8	46.7	21.4
Full Def. (8K 5µm)	7	141.7	14.1	70.9	28.1	35.5	35.2	28.4

Appendix D. Lens Mounts

D.1 F-Mount

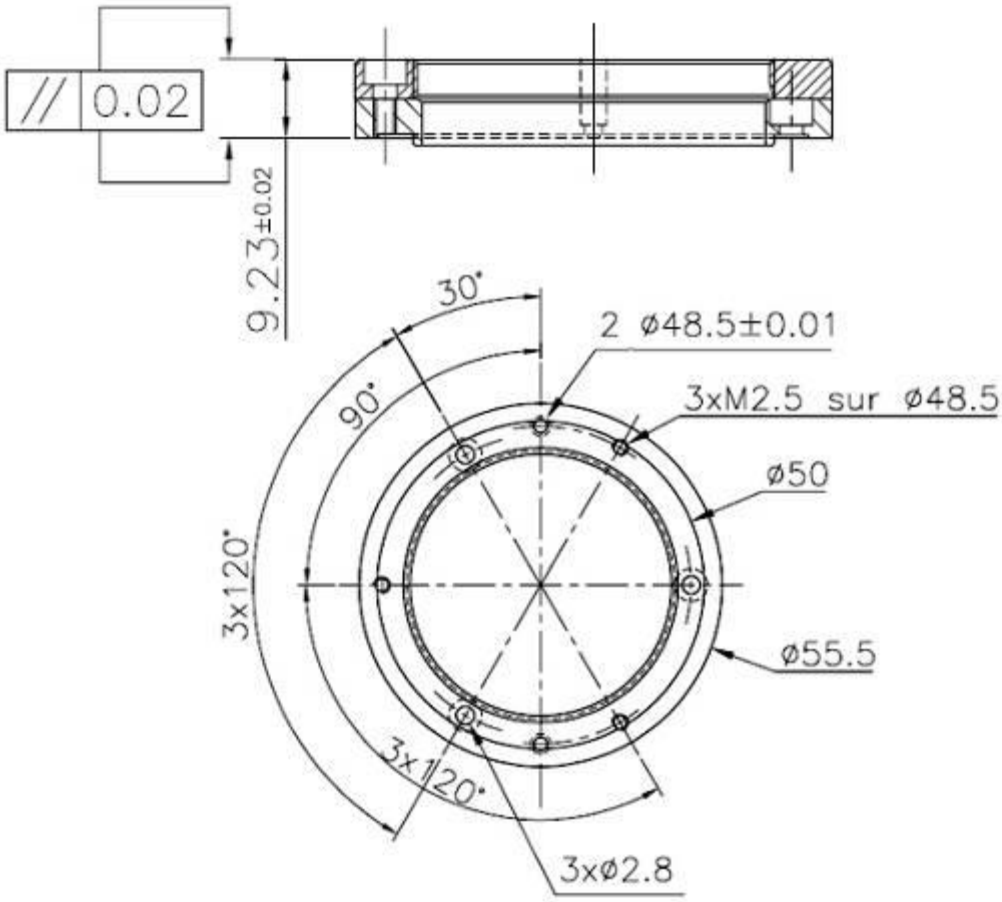
F Mount : Kit10 (Part number EV71KFPAVIVA-ABA)



D.2 T2 & M42x1 Mounts

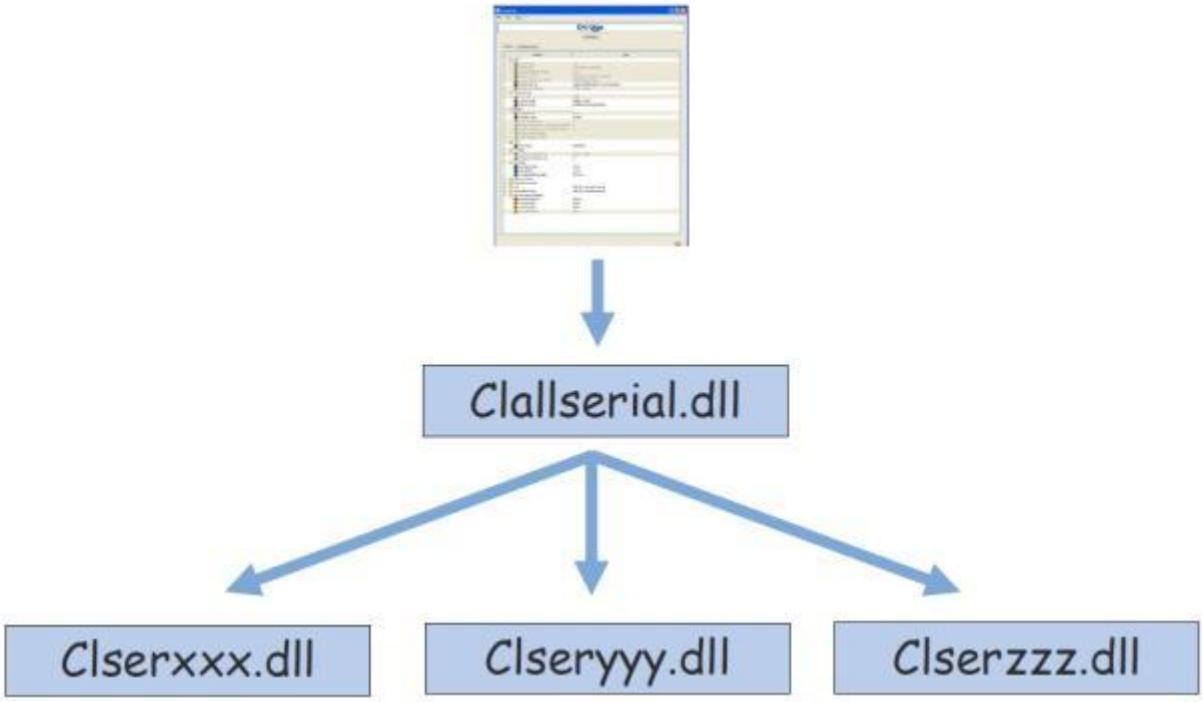
M42x0,75 (T2 Mount) : Kit30 (Part number AT71KFPAVIVA-AKA)

M42x1 Mount : Kit40 (Part number AT71KFPAVIVA-ADA)



Appendix E. Connection

The Frame Grabber has to be compliant with Camera Link 1.1

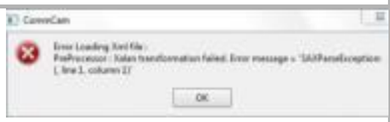


Clallserial.dll (Standard CameraLink Services Library)

- In 32bits : Must be located in : program files\CamerLink\serial and location added to PATH variable
- In 64bits : Must be located in : program files\CamerLink\serial or
  - For 32bits version : Must be located in : program files(x86)\CamerLink\serial
    and both locations added to PATH variable

Clserxxx.dll (FG Manufacturer dedicated CameraLink Services Library)

- **In 32bits** : in the directory defined by the Register Key :
  CLSERIALPATH (REG_SZ) in HKEY_LOCAL_MACHINE\software\cameralink
  The directory should be program files\CamerLink\serial or any other specified
- **In 64bits, for a 64bits version** : in the directory defined by the Register Key : CLSERIALPATH (REG_SZ) in HKEY_LOCAL_MACHINE\software\cameralink
  The directory should be program files\CamerLink\serial or any other specified
- **In Windows 64bits, for a 32bits version** : in the directory defined by the Register Key : CLSERIALPATH (REG_SZ) in HKEY_LOCAL_MACHINE\Wow6432Node\software\cameralink
  The directory should be program files(x86)\CamerLink\serial or any other specified

Defect	Detail	Solutions
<u>CommCam Can't find the Camera</u> : After launching CommCam, the Icon of the Camera is not visible.		■ The Camera is not powered up or the boot sequence is not finished. ■ The CameraLink cable is not connected or connected on the bad connector. ■ Check if the CameraLink libraries (clallserial.dll and clserXXX.dll) are in the same directory (either *system32* or *program files/cameralink/serial*) ■ The Frame Grabber is compliant with CameraLink standard 1.1 > Contact the hotline : hotline-cam@e2v.com
<u>An e2v Camera is detected but not identified</u> : A "question Mark" icon appears in place of the one of the AVIIVA2		■ The version of CommCam used is too old : You have to use the version **1.2.x** and after.
<u>Impossible to connect to the identified Camera</u> : The message "Impossible to open device" is displayed		■ There is a possible mismatch between the major version of xml file used by CommCam and the firmware version of the Camera ■ Possible Hardware error or Camera disconnected after being listed. > Contact the hotline : hotline-cam@e2v.com
<u>Error messages is displayed just after/before the connection</u> :	 	■ There is a possible mismatch between the minor version of xml file used by CommCam and the firmware version of the Camera ■ Default values of the Camera out of range > Contact the hotline : hotline-cam@e2v.com

Appendix F. Revision History

Manual Revision	Comments / Details	Firmware version	1st CommCam compliant Version
Rev A	First release	1.0.3	2.2.1
Rev B	Update Firmware	1.1.0	2.3.1
Rev C	True Color Single Change Documentation Template	1.2.0	2.3.3
Rev D	Documentation Corrections in the Color mode description	1.2.0	2.3.3
Rev E	New Documentation Template Vertical and Horizontal Interpolation Low Pass Filter for FFC Trigger Too Slow White Balance Adjust EV71YC4CCL8005-BA0 : EV71YC4CCL8005-BH0 :	1.3.0 1.4.0 1.1.0	2.5.1 2.6.0 2.6.0
Rev F	Line Interpolation not available in True Color Enhanced User Gain for Full Exposure Control Mode Color Management Section Record / Replay Image EV71YC4CCL8005-BA0 : EV71YC4CCL8005-BH0 :	1.5.0 1.2.0	3.0.0
Rev G	New Part Number definition EV71YC4CCL8005-BA0 : New Sensor version EV71YC4CCL8005-BH0 : 8k Only. New Sensor Version EV71YC4CCL4010-BH0 : 4k Only. New Sensor Version	2.0.1 2.0.0 2.0.0	3.0.1
Rev H	RGB Matrix for the “BH0” part numbers EV71YC4CCL8005-BA0 : EV71YC4CCL8005-BH0 : EV71YC4CCL4010-BH0 :	2.0.3 2.0.1 2.0.1	3.0.2
Rev I	New Teledyne-e2v Chart ROI For White Balance on 8k Versions : EV71YC4CCL8005-BA0 : EV71YC4CCL8005-BH0 : No Change for 4k : EV71YC4CCL4010-BH0 :	2.1.0 2.1.0 2.0.2	3.1.7