BASLER A400k

USER'S MANUAL

Document Number: DA00062408

Release Date: 16 November 2005

BASLER VISION TECHNOLOGIES

For customers in the U.S.A.

This equipment has been tested and found to comply with the limits for a 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.

You are cautioned that any changes or modifications not expressly approved in this manual could void your authority to operate this equipment.

The shielded interface cable recommended in this manual must be used with this equipment in order to comply with the limits for a computing device pursuant to Subpart J of Part 15 of FCC Rules.

For customers in Canada

This apparatus complies with the Class A limits for radio noise emissions set out in Radio Interference Regulations.

Pour utilisateurs au Canada

Cet appareil est conforme aux normes Classe A pour bruits radioélectriques, spécifiées dans le Règlement sur le brouillage radioélectrique.

Life Support Applications

These products are not designed for use in life support appliances, devices, or systems where malfunction of these products can reasonably be expected to result in personal injury. Basler customers using or selling these products for use in such applications do so at their own risk and agree to fully indemnify Basler for any damages resulting from such improper use or sale.

Warranty Note

Do not open the housing of the camera. The warranty becomes void if the housing is opened.

Contacting Basler Support Worldwide

Europe:

Basler AG
An der Strusbek 60 - 62
22926 Ahrensburg
Germany
Tel.: +49-4102-463-500
Fax.: +49-4102-463-599
vc.support.europe@baslerweb.com

Americas:

Basler, Inc.
740 Springdale Drive, Suite 100
Exton, PA 19341
U.S.A.
Tel.: +1-877-934-8472
Fax.: +1-877-934-7608
vc.support.usa@baslerweb.com

Asia:

Basler Asia Pte. Ltd
25 Internat. Business Park
#04-15/17 German Centre
Singapore 609916
Tel.: +65-6425-0472
Fax.: +65-6425-0473
vc.support.asia@baslerweb.com

www.basler-vc.com

Table of Contents

3 Basic Operation and Features

4 Configuring the Camera

5 Mechanical Considerations

6 Troubleshooting

1 Introduction

BASLER A400k area scan cameras are high speed CMOS cameras designed for industrial use. Superb CMOS image sensing features are combined with a robust, high precision manufactured housing.

Important features are:

- High speed
- Fast 4-megapixel CMOS digital image sensor
- Fast electronic rolling shutter
- Electronic exposure time control
- Shading correction
- Partial scanning (Area of Interest)
- Programmable area of interest sequencer
- Digital shift (2x, 4x)
- Flash trigger output
- Programmable via an RS-644 serial port
- Complies with the Camera Link standard
- Industrial housing manufactured with high planar, parallel and angular precision

1.1 Document Applicability

This User's Manual applies to A400k cameras with a camera version ID number of 05.

Cameras with a lower or a higher ID number may have fewer features or have more features than described in this manual. Features on cameras with a lower or a higher ID number may not operate exactly as described in this manual.

An easy way to see the camera version ID number for an A400k camera is by using the CCT+. To see the camera version ID number:

1. Double click the CCT+ icon on your desktop or click Start ⇒ All Programs ⇒ Basler Vision Technologies ⇒ CCT+ ⇒ CCT+. The CCT+ window will open and the software will connect to your camera.

2. Scroll down until you find the “Camera Information” group heading. If there is a plus sign beside the Camera Information group heading, click on the plus sign to show the list of parameters in the group.
3. Find the parameter called “Camera Version.” As shown in Figure 1-1, the last two numbers of this parameter are the camera version ID number.

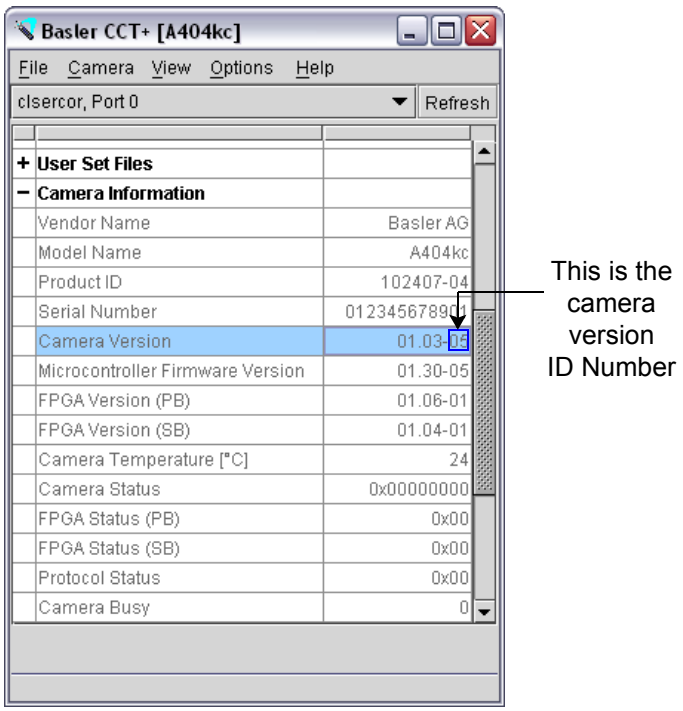


Figure 1-1: CCT+ Window

You can also access the camera version ID number by using binary commands to read the Camera Version Inquiry register. (See Section 4.2.1 for an explanation of inquiry registers and Section 4.3 for information on using binary commands.)

1.2 Camera Versions

A400k series area scan cameras are available in different versions; the version depends on the maximum frame rate and the Camera Link interface. All versions are available in monochrome models (A402k, A403k, A404k) and in color models (A402kc, A403kc, A404kc).

Throughout the manual, the camera will be called the A400k. Passages that are only valid for a specific version will be so indicated.

Throughout the manual, the statements relating to the monochrome versions also apply to the color versions. The color versions will specifically be referred to only when necessary.

Camera Version	Max. Frame Rate	Camera Link Interface
A402k	24 fps	Base configuration
A403k	48 fps	Medium configuration
A404k	96 fps	Full configuration

Table 1-1: Versions of the A400k Series Camera

1.3 Performance Specifications

Specifications	A402k	A402kc	A403k	A403kc	A404k	A404kc
Sensor	Micron MV40 CMOS active-pixel digital image sensor					
Number of Pixels	2,352 (H) x 1,726 (V) (4,059,552 pixels)					
Pixel Size	7.0 µm x 7.0 µm (7.0 µm pixel pitch)					
Pixel Fill Factor	55%					
Sensor Imaging Area	16.46 mm (H) x 12.10 mm (V), 20.43 mm (Diagonal)					
Mono or Color	Mono	Color	Mono	Color	Mono	Color
Digital Responsivity	2500 LSB/lux*s					
Quantum Efficiency	(Figure 1-2)	(Figure 1-3)	(Figure 1-2)	(Figure 1-3)	(Figure 1-2)	(Figure 1-3)
Dynamic Range	54 dB					
Shutter	Fast electronic rolling shutter					
PRNU (Photo Response Non-uniformity)	Typically < 1% rms according to the sensor manufacturer's specification Lower if PRNU shading correction is used.					
DSNU (Dark Signal Non-uniformity)	0.1% rms (if no DSNU shading correction is used) Lower if DSNU shading correction is used.					
Kdrk (Dark Current Temperature Coefficient)	100% / 8° C					
Pixel Clock Speed	50 MHz					
Frame Rate (at full resolution)	24 fps progressive scan		48 fps progressive scan		48 fps (in 4 tap mode) 96 fps (in 8 tap mode) progressive scan	
Video Data Output Type	Camera Link LVDS Base configuration RS-644 LVDS when used with the optional Basler Interface Converter (k-BIC)		Camera Link LVDS Medium configuration		Camera Link LVDS Medium configuration (in 4 tap mode) Camera Link LVDS Full configuration (in 8 tap mode)	

Table 1-2: A400k Performance Specifications

Specifications	A402k	A402kc	A403k	A403kc	A404k	A404kc
Video Data Output Mode(s)	2 taps (2 pixels /clock cycle) Selectable 8 or 10 bit depth		4 taps (4 pixels / clock cycle) Selectable 8 or 10 bit depth		4 taps (4 pixels / clock cycle) Selectable 8 or 10 bit depth 8 taps (8 pixels / clock cycle) 8 bit depth	
Output Data Rate	93 MB/s (2 taps - 8 bit depth) 116 MB/s (2 taps - 10 bit depth)		186 MB/s (4 taps - 8 bit depth) 232 MB/s (4 taps - 10 bit depth)		186 MB/s (4 taps - 8 bit depth) 232 MB/s (4 taps - 10 bit depth) 372 MB/s (8 taps - 8 bit depth)	
Synchronization	Via external ExSync signal or free-run					
Exposure Time Control	Edge-controlled, level-controlled or programmable					
Gain and Offset	Programmable via the frame grabber via a serial link					
Connectors	All versions: one, 26 pin, female MDR connector (data) one, 6 pin, Hirose HR connector (power) one, 4 pin, Hirose HR connector (flash trigger) A403k, A403kc & A404k A404kc:second, 26 pin, female MDR connector (data)					
Power Requirements	12 VDC ± 10% Max 6.5 W @ 12 VDC		12 VDC ± 10% Max 7.0 W @ 12 VDC		12 VDC ± 10% Max 7.5 W @ 12 VDC	
Lens Adapter	F-mount					
Housing Size (L x W x H) Including Connectors	Without lens adapter: 53.8 mm x 90 mm x 90 mm With F-mount adapter: 85.3 mm x 90 mm x 90 mm					
Weight without lens adapter: with F-mount adapter:	~ 500 g ~ 605 g		~ 510 g ~ 615 g		~ 510 g ~ 615 g	
Conformity	CE, FCC					

Table 1-2: A400k Performance Specifications

1.4 Spectral Response

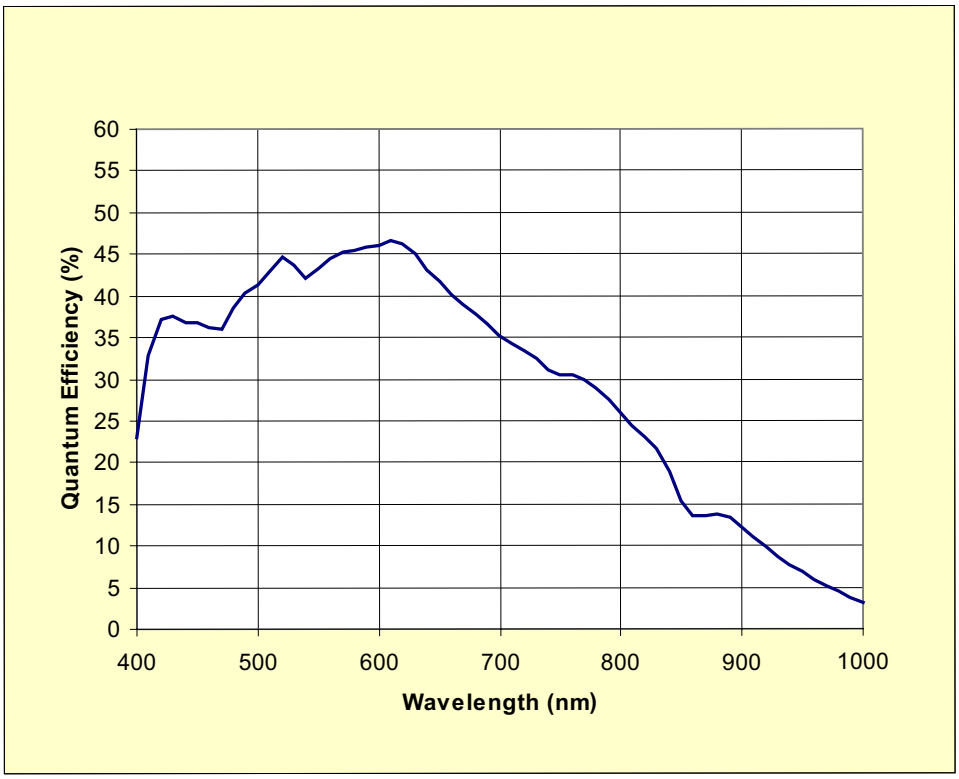


Figure 1-2: Quantum Efficiency for A400k Cameras; Peak at 46% at 620 nm

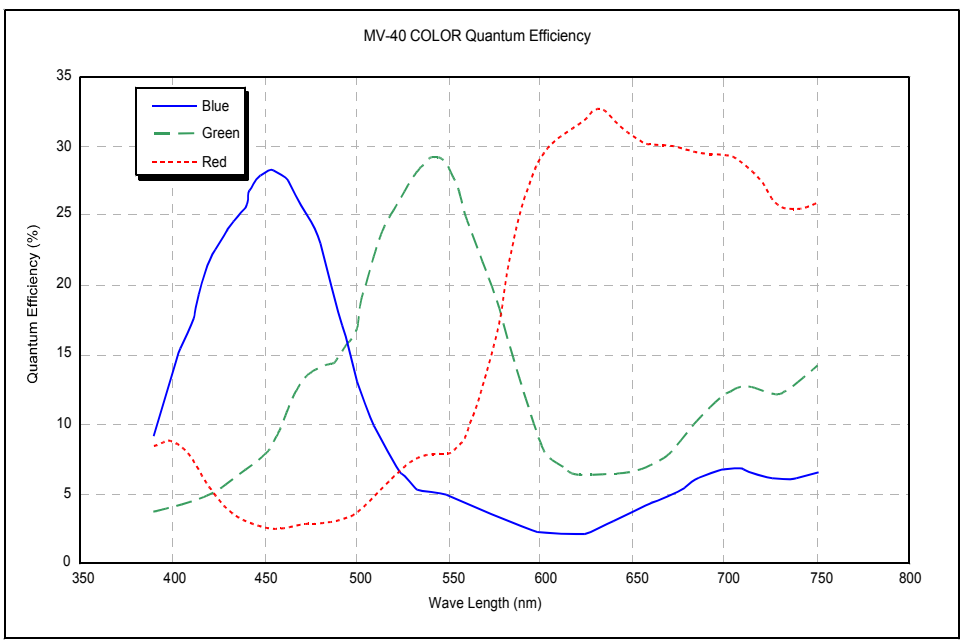


Figure 1-3: Color Quantum Efficiencies for A400kc Cameras

The spectral response curves exclude lens characteristics and light source characteristics.

To obtain best performance regarding the camera's blooming, smearing and dark signal non-uniformity characteristics, use of a dielectric IR cut-off filter is recommended. The filter should transmit in a range of 400 nm to 700...720 nm, and it should cut off from 700...720 nm to 1100 nm.

1.5 Environmental Requirements

1.5.1 Temperature and Humidity

Housing temperature during operation:	0° C ... + 50° C (+ 32° F ... + 122° F)
Humidity during operation:	20% ... 80%, relative, non-condensing

You can read out the camera's *inner* temperature via the temperature register (page 4-8). The maximum recommended inner temperature is 65° C (149° F).

Note that the camera components' life time and the image quality are higher the lower the temperature of the camera.

1.5.2 Ventilation

Allow sufficient air circulation around the camera to prevent internal heat build-up in your system and to keep the camera housing temperature during operation below the maximum shown above. Provide additional cooling such as fans or heat sinks if necessary.

1.6 Precautions

Power

Caution!

Be sure that all power to your system is switched off before you make or break connections to the camera. Making or breaking connections when power is on can result in damage to the camera.

If you can not switch off power, be sure that the power supply connector is the last connector plugged when you make connections to the camera, and the first connector unplugged when you break connections.

The camera is equipped with an undervoltage lockout. An input voltage below 10.8 VDC will cause the camera to automatically switch off.

The camera has no overvoltage protection. An input voltage higher than 13.2 VDC will damage the camera.

The camera is not protected for reverse voltage. If reverse voltage is applied to the camera while it is connected to a frame grabber in a PC, the camera could be seriously damaged.

Read the manual

Read the manual carefully before using the camera.

Keep foreign matter outside of the camera

Do not open the camera housing. Touching internal components may damage them.

Be careful not to allow liquids, dust, sand, flammable, or metallic material inside the camera housing. If operated with any foreign matter inside, the camera may fail or cause a fire.

Electromagnetic Fields

Do not operate the camera in the vicinity of strong electromagnetic fields. Avoid electrostatic charging.

Transporting

Only transport the camera in its original packaging. Do not discard the packaging.

Cleaning

Avoid cleaning the surface of the CMOS sensor if possible. If you must clean it, use a soft, lint free cloth dampened with a small quantity of isopropyl (= pure alcohol). Do not use methylated alcohol. Because electrostatic discharge can damage the CMOS sensor, you must use a cloth that will not generate static during cleaning (cotton is a good choice).

To clean the surface of the camera housing, use a soft, dry cloth. To remove severe stains, use a soft cloth dampened with a small quantity of neutral detergent, then wipe dry.

Do not use volatile solvents such as benzine and thinners; they can damage the surface finish.

2 Camera Interface

2.1 Connections

2.1.1 General Description

All A400k area scan cameras are interfaced to external circuitry via three connectors located on the back of the camera:

- a 26 pin, 0.050 inch Mini D Ribbon (MDR) female connector used to transmit video data, control data, and configuration data,
- a 6 pin, micro-miniature, push-pull receptacle used to provide power to the camera,
- a 4 pin, micro-miniature, push-pull receptacle used to output a TTL flash trigger signal

A403k and A404k area scan cameras have one additional connector, a 26 pin, 0.050 inch Mini D Ribbon (MDR) female connector used to transmit further image data.

A status LED located on the back of the camera is used to indicate power present and signal integrity. See Section 6.1 for details. Figure 2-1 shows the connectors and the LED.

Caution!

Be sure that all power to your system is switched off before you make or break connections to the camera. Making or breaking connections when power is on can result in damage to the camera.

If you can not switch off power, be sure that the power supply connector is the last connector plugged when you make connections to the camera, and the first connector unplugged when you break connections.

The camera is equipped with an undervoltage lockout. An input voltage below 10.8 VDC will cause the camera to automatically switch off.

The camera has no overvoltage protection. An input voltage higher than 13.2 VDC will damage the camera.

The camera is not protected for reverse voltage. If reverse voltage is applied to the camera while it is connected to a frame grabber in a PC, the camera could be seriously damaged.

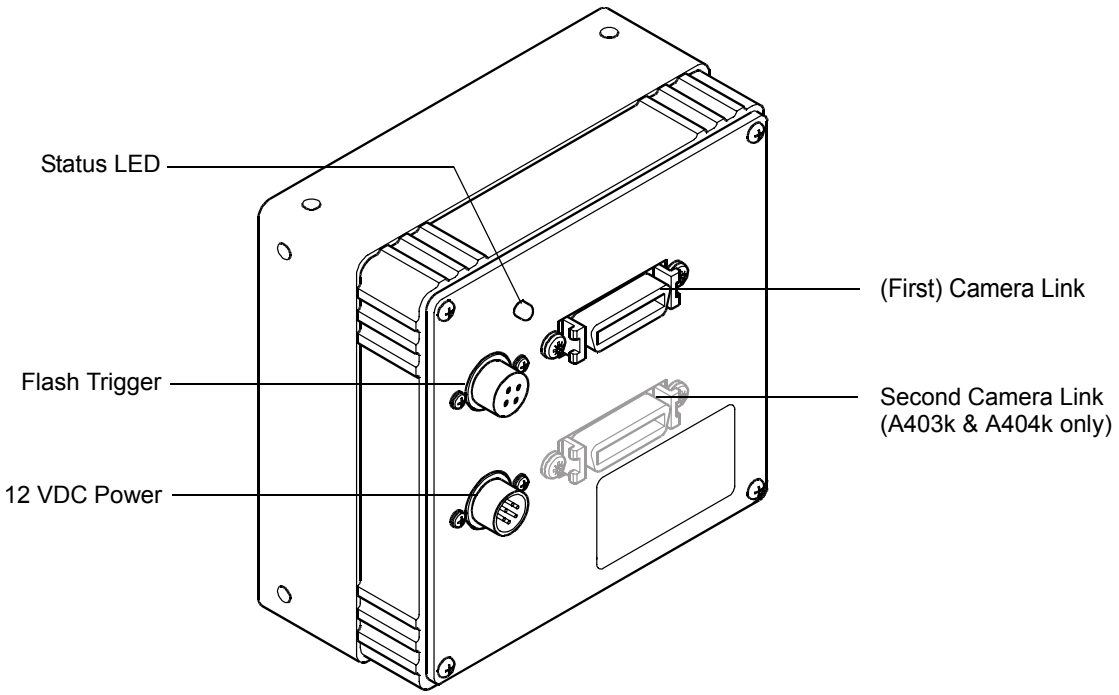


Figure 2-1: A400k Connectors and LED

	The camera housing is not grounded and is electrically isolated from the circuit boards inside of the camera. Note that the connectors at the camera are described, NOT the connectors required at the connecting cables.

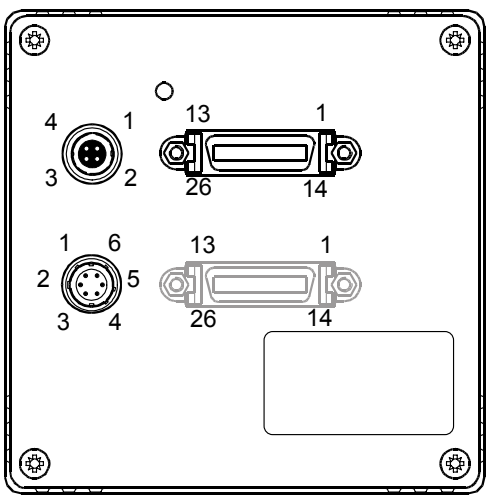


Figure 2-2: A400k Pin Numbering

2.1.2 Pin Assignments for the 26-Pin MDR Connector(s)

The 26-pin connector on the camera is a female 0.050 inch MDR connector as called for in the Camera Link Specification. It is used to interface video data, control signals, and configuration data. The pin assignments for the 26 pin, MDR connector are given in Table 2-1.

Table 2-2 provides the pin assignments for the second 26 pin, MDR connector that is only present on A403k and A404k cameras.

(First) MDR Connector:

Pin Number	Signal Name	Direction	Level	Function
1, 13, 14, 26[1]	Gnd	Input	Ground	Ground for the inner shield of the cable
2	X0-	Output	Camera Link LVDS	Data from Camera Link Transmitter
15	X0+			
3	X1-	Output	Camera Link LVDS	Data from Camera Link Transmitter
16	X1+			
4	X2-	Output	Camera Link LVDS	Data from Camera Link Transmitter
17	X2+			
6	X3-	Output	Camera Link LVDS	Data from Camera Link Transmitter
19	X3+			
5	XClk-	Output	Camera Link LVDS	Transmit Clock from Camera Link Transmitter
18	XClk+			
7	SerTC+	Input	RS-644	Serial Communication Data Receive (SerTC = “Serial to Camera“)
20	SerTC-			
8	SerTFG-	Output	RS-644	Serial Communication Data Transmit (SerTFG = “Serial to Frame Grabber “)
21	SerTFG+			
9	CC1-	Input	RS-644 LVDS	ExSync: External Trigger
22	CC1+			
10	CC2+	Input	RS-644 LVDS	ExClk. The input is not supported.
23	CC2-			
11	CC3-	Input	RS-644 LVDS	ExFlash: External Flash Trigger
24	CC3+			
12	CC4+	Input	RS-644 LVDS	Not used
25	CC4-			

[1] Pins 1, 13, 14, and 26 are all tied together to Gnd inside of the camera.

Table 2-1: A400k Pin Assignments for the (First) 26-pin MDR Connector

Second MDR Connector (A403k and A404k only):

Pin Number	Signal Name	Direction	Level	Function
1, 13, 14, 26[1]	Gnd	Input	Ground	Ground for the inner shield of the cable
2	Y0-	Output	Camera Link LVDS	Data from Camera Link Transmitter
15	Y0+			
3	Y1-	Output	Camera Link LVDS	Data from Camera Link Transmitter
16	Y1+			
4	Y2-	Output	Camera Link LVDS	Data from Camera Link Transmitter
17	Y2+			
6	Y3-	Output	Camera Link LVDS	Data from Camera Link Transmitter
19	Y3+			
5	YClk-	Output	Camera Link LVDS	Transmit Clock from Camera Link Transmitter
18	YClk+			
7	T+			Connected to T- with 100R; not used
20	T-			Connected to T+ with 100R; not used
8	Z0-	Output	Camera Link LVDS	Data from Camera Link Transmitter
21	Z0+			
9	Z1-	Output	Camera Link LVDS	Data from Camera Link Transmitter
22	Z1+			
10	Z2-	Output	Camera Link LVDS	Data from Camera Link Transmitter
23	Z2+			
12	Z3-	Output	Camera Link LVDS	Data from Camera Link Transmitter
25	Z3+			
11	ZClk-	Output	Camera Link LVDS	Transmit Clock from Camera Link Transmitter
24	ZClk+			

[1] Pins 1, 13, 14, and 26 are all tied together to Gnd inside of the camera.

Table 2-2: A403k and A404k Pin Assignments for the Second 26-pin MDR Connector

2.1.3 Pin Assignments for the 6-pin Micro-Miniature Receptacle

The power input connector is a 6 pin, micro-miniature, push-pull locking receptacle, the Hirose HR 10A-7R-6PB. The power supply should deliver 12 V at a minimum of 1.5 A with a voltage accuracy of ±10%. The pin assignment of the plug is given in Table 2-3.

Pin Number	Signal Name	Direction	Level	Function
1, 2[1]	+12 VDC	Input	12 VDC ± 10%	Camera power input
3, 4	-	-	-	Not connected
5, 6[2]	DC Gnd	Input	Ground	DC ground

[1] Pins 1, and 2 are tied together inside of the camera.

[2] Pins 5, and 6 are tied together inside of the camera.

Table 2-3: A400k Pin Assignments for the 6-pin Micro-miniature Receptacle

The recommended mating connector is the Hirose micro-miniature locking plug (part # HR10-7P-6S). A plug of this type will be shipped with each camera. The plug should be used to terminate the cable on the power supply for the camera.

2.1.4 Pin Assignments for the 4-pin Micro-Miniature Receptacle

The flash trigger output connector type is a micro-miniature push-pull locking connector, the Hirose HR 10A-7R-4S. The receptacle provides a TTL signal for an external flash. This signal can be programmed to be deactivated, tied to a flash window signal generated internally, tied to the external ExFlash input, and it can be permanently on (see Section 2.5.8). Figure 2-3 shows the timing diagram.

It can be set to high impedance (default setting) so that the flash trigger is disabled or, it can be selected to be to TTL Active High, Low Side Switch (Open Collector), or High Side Switch. Figure 2-4 shows the three variants of output schematics of the flash trigger connector.

The pin assignment is given in Table 2-4.

Pin Number	Signal Name	Direction	Level	Function
2	Flash Trigger	Output	TTL signal	Flash trigger; the HIGH signal is current limited to 50 mA ±20%.
1, 3	-	-	-	Not connected
4	DC Gnd	Output	Ground	DC ground

Table 2-4: A400k Pin Assignments for the 4-pin Micro-miniature Receptacle

The recommended mating connector is the Hirose HR 10A-7P-4P.

The flash trigger signal is short-circuit proof. Insulation voltage is 100 V.

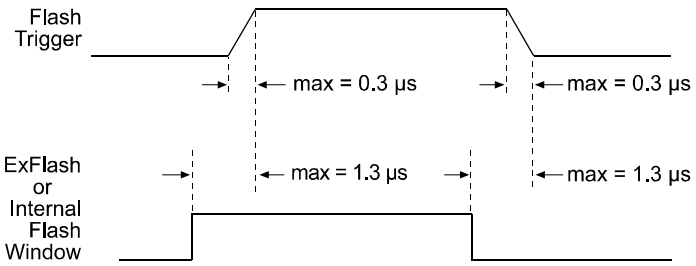


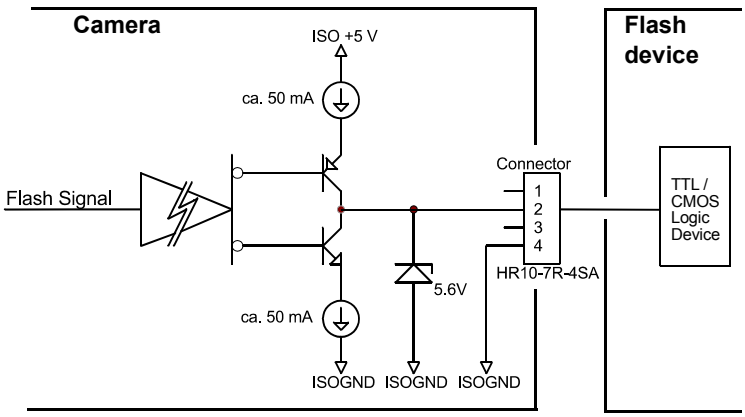
Figure 2-3: Flash Trigger Signal Timing

TTL Active High (Default)

A TTL Active High output signal is typically used together with a TTL / CMOS Logic Device.

The TTL Active High output signal has the following characteristics:

- High output min. 4.5 V at 10 mA output load, shortcut current 50 mA (+40%/-20%)
- Low output max. 0.5 V at -10 mA output load, shortcut current -50 mA (+40%/-20%)

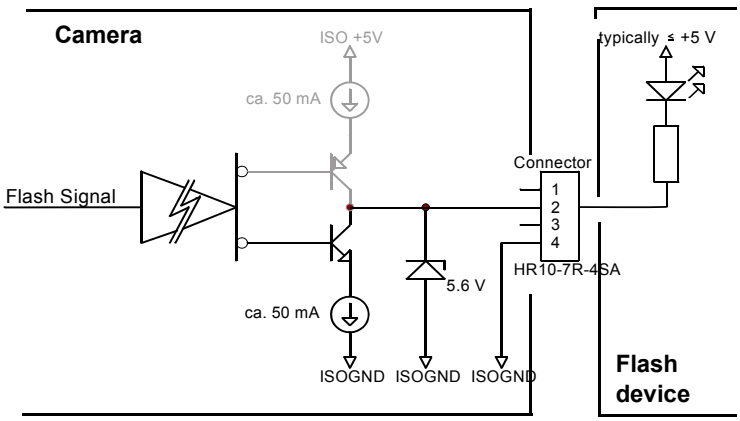


Low Side Switch (Open Collector)

When you select this output signal variant, the upper transistor is deactivated, which is shown by grayed lines in the schematic.

The schematic shows a sample circuit for your flash device.

Calculate your devices so that the maximum output current is 50 mA.



High Side Switch

When you select this output signal variant, the lower transistor is deactivated, which is shown by grayed lines in the schematic.

The schematic shows a sample circuit for your flash device.

Calculate your devices so that the maximum output current is 50 mA.

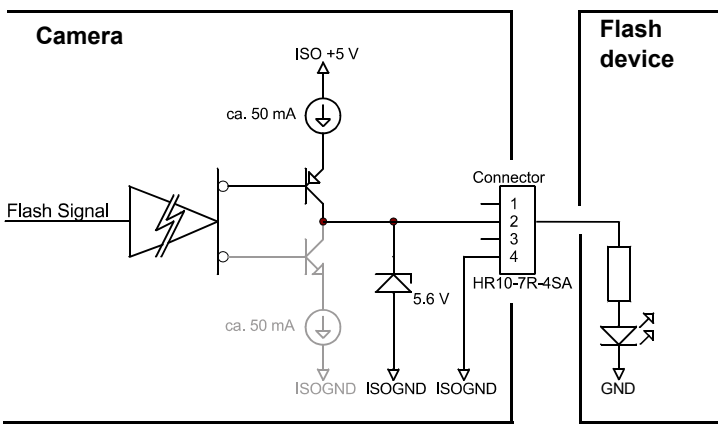


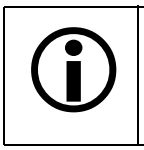
Figure 2-4: Flash Trigger Output Schematics

2.2 Cable Information

2.2.1 Camera Link Cable

A Camera Link compatible MDR cable assembly is available from Basler as a stock item (part # 1000013905 for a 3 meter cable and part # 1000013906 for a 5 meter cable). Alternatively, you can use the cable assembly manufactured by 3M (part # 14X26-SZLB-XXX-0LC).

The maximum allowed length for the MDR cable used with an A400k is 7 meters.



The maximum cable length will decrease when used in an area with severe ambient electromagnetic interference.

2.2.2 Power Cable

A Hirose, 6-pin locking plug will be shipped with each camera. This plug should be used to connect the power supply cable to the camera.

For proper EMI protection, the power supply cable attached to this plug must be a twin-cored, shielded cable. Also, the housing of the Hirose plug must be connected to the cable shield and the cable must be connected to earth ground at the power supply.

Power requirements are given in Section 2.8.

2.3 Camera Link Implementation in the A400k

The A400k uses a National Semiconductor DS90CR287 as a Camera Link transmitter. For a Camera Link receiver, we recommend that you use the National Semiconductor DS90CR288, the National Semiconductor DS90CR288A or an equivalent. Detailed data sheets for these components are available at the National Semiconductor web site (www.national.com). The data sheets contain all of the information that you need to implement Camera Link, including application notes.

Note that the timing used for sampling the data at the Camera Link receiver in the frame grabber varies from device to device. On some receivers, TTL data must be sampled on the rising edge of the receive clock, and on others, it must be sampled on the falling edge. Also, some devices are available which let you select either rising edge or falling edge sampling. Please consult the data sheet for the receiver that you are using for specific timing information.

The A400k uses a National Semiconductor DS90LV048A differential line receiver to receive the RS-644 camera control input signals and the serial communication input signal defined in the Camera Link specification. A DS90LV047A differential line transmitter is used to transmit the serial communication output signal defined in the specification. Detailed spec sheets for these devices are available at the National Semiconductor web site (www.national.com).

The A402k uses the base configuration of Camera Link with one differential line transmitter. The transmitter in the camera is designated as “Transmitter X.” The schematic in Figure 2-5 shows the interface for the A402k and a typical implementation for the frame grabber interface.

The A403k and the A404k (when set for 4 tap output) use the medium configuration of Camera Link with two differential line transmitters. The transmitters in the camera are designated as “Transmitter X” and “Transmitter Y.” The schematic in Figure 2-6 shows the interface for the A403k and a typical implementation for the frame grabber interface.

The A404k (when set for 8 tap output) uses the full configuration of Camera Link with three differential line transmitters. The transmitters in the camera are designated as “Transmitter X”, “Transmitter Y” and “Transmitter Z.” The schematic in Figure 2-7 shows the interface for the A404k and a typical implementation for the frame grabber interface.

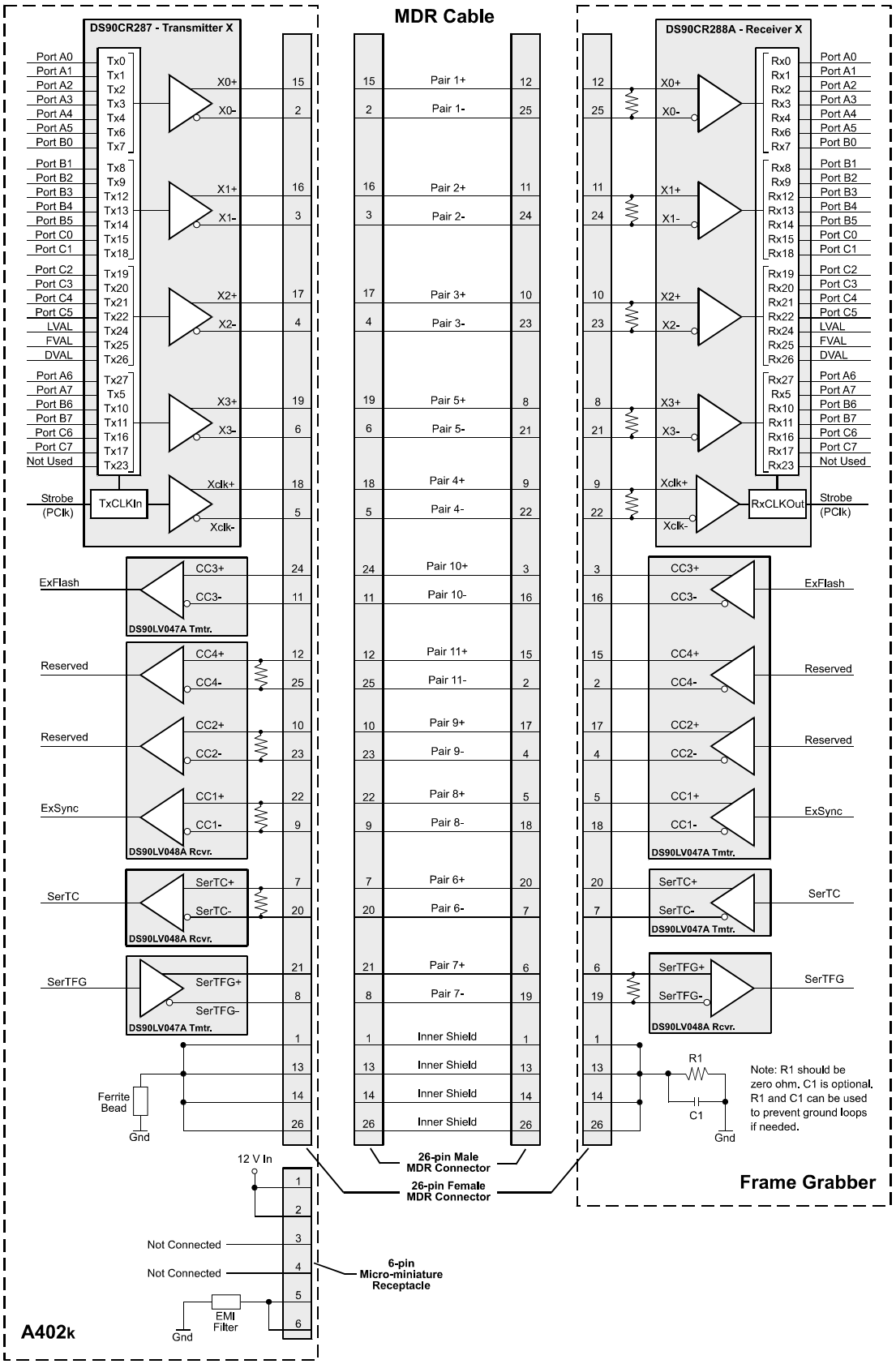


Figure 2-5: A402k Camera / Frame Grabber Interface

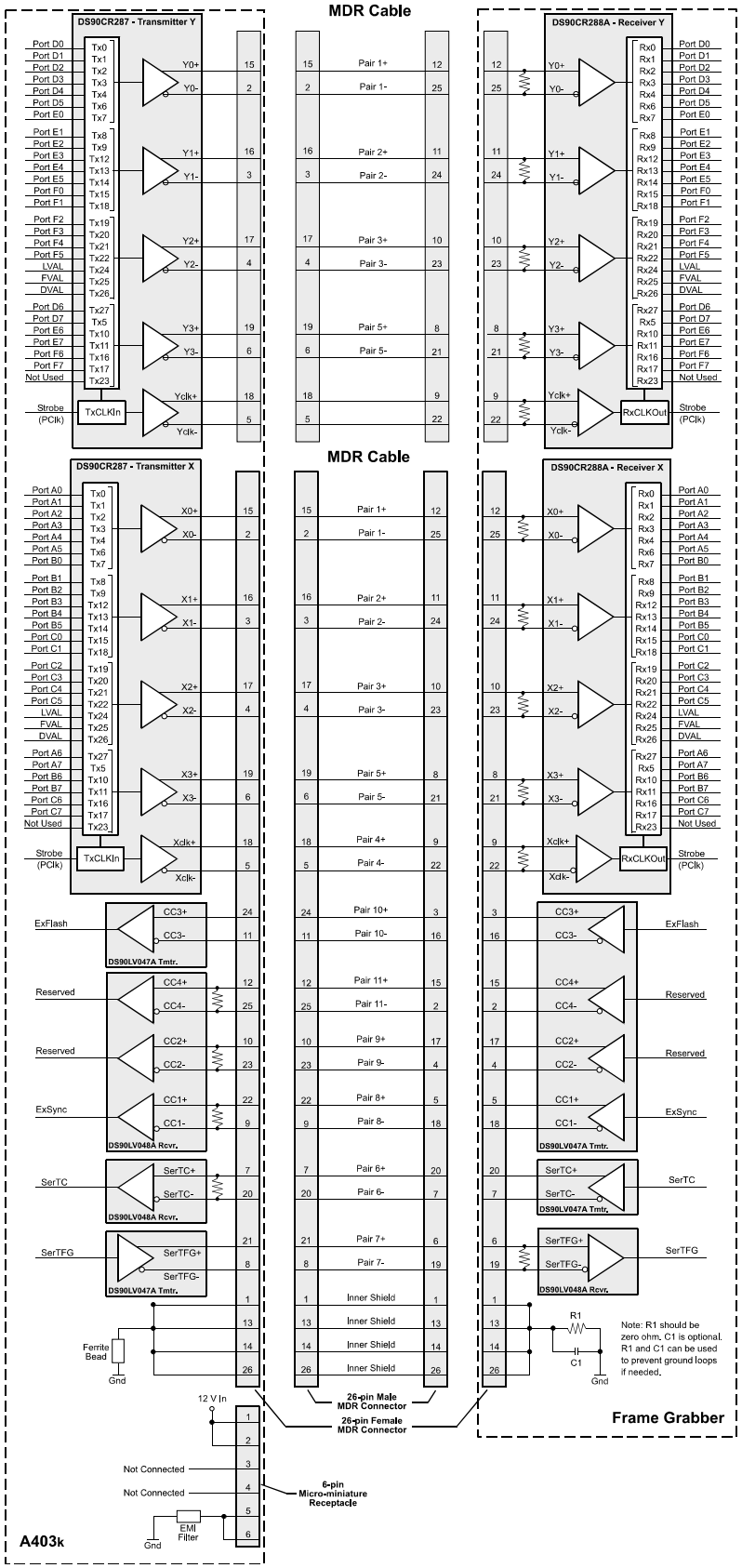


Figure 2-6: A403k Camera / Frame Grabber Interface

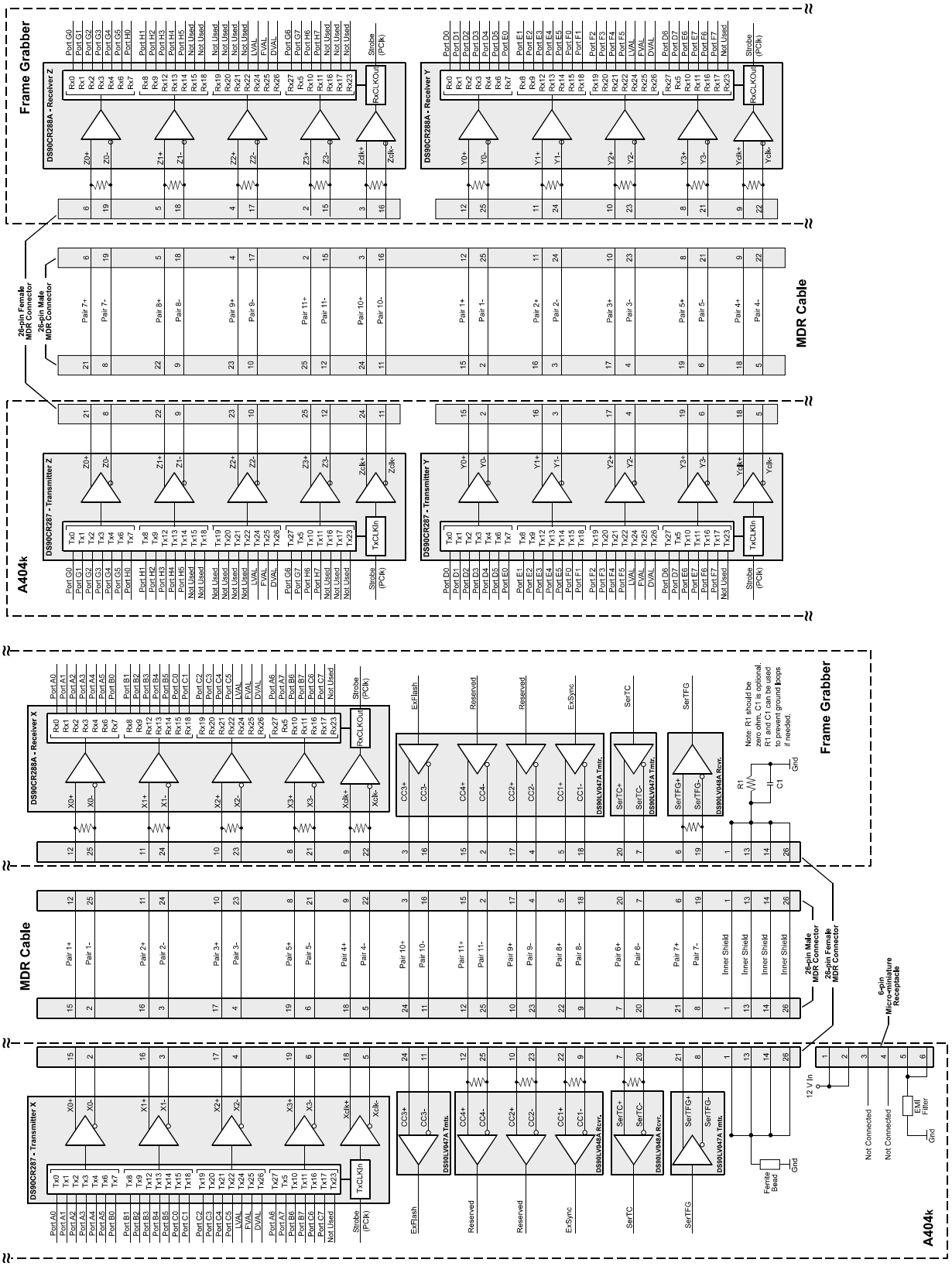


Figure 2-7: A404k Camera / Frame Grabber Interface

2.4 Input Signals

The A400k receives the RS-644 input signals ExSync, ExFlash, and SerTC (“Serial to Camera“) of the serial interface. Section 2.4.1 describes the function of the ExSync signal, Section 2.4.2 describes the function of the ExFlash signal. SerTC of the serial communication is described in Section 2.6.

2.4.1 ExSync: Controls Frame Readout and Exposure Time

The ExSync input signal is used to control exposure and readout of the A400k. ExSync is an LVDS signal as specified for RS-644. The ExSync input corresponds to the camera control signal CC1 as defined in the Camera Link standard. CC2 and CC4 are not used in this camera.

The camera can be programmed to function under the control of an externally generated sync signal (ExSync) in two exposure time control modes. In these modes, level-controlled and programmable, the ExSync signal is used to control exposure time and frame read out. For more detailed information on the two modes, see Section 3.3.

ExSync can be a periodic or non-periodic function. The frequency of the ExSync signal determines the camera’s frame rate in these modes.

$$\text{Maximum frame rate} = \frac{1}{\text{Minimum ExSync signal period}}$$

Note that ExSync is edge sensitive and therefore must toggle.

In ExSync edge-controlled mode and programmable mode, minimum high time for the ExSync signal is 250 ns, minimum low time is also 250 ns. In ExSync level-controlled mode, minimum high time for the ExSync signal is 9.12 µs, minimum low time is 4.56 µs.

The ExSync signal is typically supplied to the camera by a frame grabber board. Refer to the manual supplied with your frame grabber to determine how to set up the ExSync signal.

2.4.2 ExFlash from the Frame Grabber

The first Camera Link contains an LVDS input for the ExFlash signal. With the corresponding register setting, this input can be tied to the output signal of the flash trigger connector. The ExFlash signal is not used by the camera itself. The ExFlash input corresponds to the camera control signal CC3 as defined in the Camera Link standard.

The minimum pulse width of ExFlash is 1 µs, there are no further restrictions.

2.5 Output Signals

Data is output from the A400k using the Camera Link standard. The Pixel Clock signal is described in Section 2.5.1, the Line Valid signal in Section 2.5.2, the Frame Valid signal in Section 2.5.3, and the video data in Section 2.5.4. Video Data output is described in Sections 2.5.5 and 2.5.6. Section 2.5.8 describes the flash trigger signal. SerTFG ("Serial to Frame Grabber") of the serial communication is described in Section 2.6.

2.5.1 Pixel Clock

On the A402k, the pixel clock is assigned to the strobe port (TxClk pin) on Camera Link transmitter X as defined in the Camera Link standard and as shown in Table 2-5. On the A403k , the pixel clock is assigned to the strobe port on Camera Link transmitter X and Camera Link transmitter Y as defined in the standard and as shown in Tables 2-6 and 2-7. On the A404k, the pixel clock is assigned to the strobe port on transmitters X, Y and Z as defined in the standard and as shown in Tables 2-8, 2-9 and 2-10. The pixel clock is used to time the sampling and transmission of pixel data. The Camera Link transmitter(s) used in A400k cameras require pixel data to be sampled and transmitted on the rising edge of the clock.

The frequency of the pixel clock is 50 MHz. For the A402k, on each Pixel Clock signal, two pixels are transmitted at 8 bit or 10 bit depth. For the A403k, on each Pixel Clock signal, four pixels are transmitted at 8 bit or 10 bit depth. For the A404k, on each Pixel Clock signal, four pixels are transmitted at 8 bit or 10 bit depth when the camera is set for 4 tap output. When an A404k is set for 8 tap output, eight pixels at a depth of 8 bits are transmitted on each Pixel Clock signal.

2.5.2 Line Valid Bit

As shown in Figures 2-9 through 2-13, the line valid bit indicates that a valid line is being transmitted. Pixel data is only valid when this bit is high. On the A402k, 1176 pixel clocks are required to transmit one full line. On the A403k, 588 pixel clocks are required to transmit one full line. On the A404k, 588 pixel clocks are required to transmit one full line when the camera is set for 4 tap output and 294 pixel clocks are required when the camera is set for 8 tap output.

On the A402k, line valid is assigned to the line valid port on Camera Link transmitter X as defined in the Camera Link standard. On the A403k, line valid is assigned to the line valid ports on Camera link transmitters X and Y as defined in the standard. On the A404k, line valid is assigned to the line valid ports on transmitters X, Y and Z as defined in the standard (see Tables 2-5 through 2-10).

2.5.3 Frame Valid Bit

As shown in Figures 2-9 through 2-13, the frame valid bit indicates that a valid frame is being transmitted. Pixel data is only valid when the frame valid bit and the line valid bit are both high.

One frame can contain 2 to 1726 Line Valid signals.

On the A402k, frame valid is assigned to the frame valid port on Camera Link transmitter X as defined in the Camera Link standard. On the A403k, frame valid is assigned to the frame valid ports on Camera link transmitters X and Y as defined in the standard. On the A404k, frame valid is assigned to the frame valid ports on transmitters X, Y and Z as defined in the standard. (See Tables 2-5 through 2-10).

2.5.4 Video Data (Bit Assignments)

Table 2-5 and Figure 2-5 show the assignment of pixel data bits to the input pins on the X Camera Link transmitter in an A402k camera. They also show the assignments for the corresponding output pins on the X Camera Link receiver in a frame grabber. The assignments for the frame valid bit and the line valid bit are also listed.

Tables 2-6 and 2-7 and Figure 2-6 show the assignment of pixel data bits to the input pins on the X and Y Camera Link transmitters in an A403k camera. They also show the assignments for the corresponding output pins on the X and Y Camera Link receivers in a frame grabber. The assignments for the frame valid bit and the line valid bit are also listed.

Tables 2-8, 2-9 and 2-10 and Figure 2-7 show the assignment of pixel data bits to the input pins on the X, Y and Z Camera Link transmitters in an A404k camera. They also show the assignments for the corresponding output pins on the X, Y and Z Camera Link receivers in a frame grabber. The assignments for the frame valid bit and the line valid bit are also listed.

A402k, Transmitter X				
Port	Camera	Frame Grabber	Signal	
			2 Tap 8 Bit	2 Tap 10 Bit
Port A0	TxIN0	RxOUT0	D0 Bit 0	D0 Bit 0
Port A1	TxIN1	RxOUT1	D0 Bit 1	D0 Bit 1
Port A2	TxIN2	RxOUT2	D0 Bit 2	D0 Bit 2
Port A3	TxIN3	RxOUT3	D0 Bit 3	D0 Bit 3
Port A4	TxIN4	RxOUT4	D0 Bit 4	D0 Bit 4
Port A5	TxIN6	RxOUT6	D0 Bit 5	D0 Bit 5
Port A6	TxIN27	RxOUT27	D0 Bit 6	D0 Bit 6
Port A7	TxIN5	RxOUT5	D0 Bit 7 (MSB)	D0 Bit 7
Port B0	TxIN7	RxOUT7	D1 Bit 0	D0 Bit 8
Port B1	TxIN8	RxOUT8	D1 Bit 1	D0 Bit 9 (MSB)
Port B2	TxIN9	RxOUT9	D1 Bit 2	Not Used
Port B3	TxIN12	RxOUT12	D1 Bit 3	Not Used
Port B4	TxIN13	RxOUT13	D1 Bit 4	D1 Bit 8
Port B5	TxIN14	RxOUT14	D1 Bit 5	D1 Bit 9 (MSB)
Port B6	TxIN10	RxOUT10	D1 Bit 6	Not Used
Port B7	TxIN11	RxOUT11	D1 Bit 7 (MSB)	Not Used
Port C0	TxIN15	RxOUT15	Not Used	D1 Bit 0
Port C1	TxIN18	RxOUT18	Not Used	D1 Bit 1
Port C2	TxIN19	RxOUT19	Not Used	D1 Bit 2
Port C3	TxIN20	RxOUT20	Not Used	D1 Bit 3
Port C4	TxIN21	RxOUT21	Not Used	D1 Bit 4
Port C5	TxIN22	RxOUT22	Not Used	D1 Bit 5
Port C6	TxIN16	RxOUT16	Not Used	D1 Bit 6
Port C7	TxIN17	RxOUT17	Not Used	D1 Bit 7
LVAL	TxIN24	RxOUT24	Line Valid	Line Valid
FVAL	TxIN25	RxOUT25	Frame Valid	Frame Valid
DVAL	TxIN26	RxOUT26	Not Used	Not Used
Spare	TxIN23	RxOUT23	Not Used	Not Used
Strobe	TxINCLK	RxOUTCLK	Pixel Clock	Pixel Clock

Table 2-5: Bit Assignments for Transmitter X in an A402k (Base Configuration)

A403k, Plug No. 1, Transmitter X				
Port	Camera	Frame Grabber	Signal	
			4 Tap 8 Bit	4 Tap 10 Bit
Port A0	TxIN0	RxOUT0	D0 Bit 0	D0 Bit 0
Port A1	TxIN1	RxOUT1	D0 Bit 1	D0 Bit 1
Port A2	TxIN2	RxOUT2	D0 Bit 2	D0 Bit 2
Port A3	TxIN3	RxOUT3	D0 Bit 3	D0 Bit 3
Port A4	TxIN4	RxOUT4	D0 Bit 4	D0 Bit 4
Port A5	TxIN6	RxOUT6	D0 Bit 5	D0 Bit 5
Port A6	TxIN27	RxOUT27	D0 Bit 6	D0 Bit 6
Port A7	TxIN5	RxOUT5	D0 Bit 7 (MSB)	D0 Bit 7
Port B0	TxIN7	RxOUT7	D1 Bit 0	D0 Bit 8
Port B1	TxIN8	RxOUT8	D1 Bit 1	D0 Bit 9 (MSB)
Port B2	TxIN9	RxOUT9	D1 Bit 2	Not Used
Port B3	TxIN12	RxOUT12	D1 Bit 3	Not Used
Port B4	TxIN13	RxOUT13	D1 Bit 4	D1 Bit 8
Port B5	TxIN14	RxOUT14	D1 Bit 5	D1 Bit 9 (MSB)
Port B6	TxIN10	RxOUT10	D1 Bit 6	Not Used
Port B7	TxIN11	RxOUT11	D1 Bit 7 (MSB)	Not Used
Port C0	TxIN15	RxOUT15	D2 Bit 0	D1 Bit 0
Port C1	TxIN18	RxOUT18	D2 Bit 1	D1 Bit 1
Port C2	TxIN19	RxOUT19	D2 Bit 2	D1 Bit 2
Port C3	TxIN20	RxOUT20	D2 Bit 3	D1 Bit 3
Port C4	TxIN21	RxOUT21	D2 Bit 4	D1 Bit 4
Port C5	TxIN22	RxOUT22	D2 Bit 5	D1 Bit 5
Port C6	TxIN16	RxOUT16	D2 Bit 6	D1 Bit 6
Port C7	TxIN17	RxOUT17	D2 Bit 7 (MSB)	D1 Bit 7
LVAL	TxIN24	RxOUT24	Line Valid	Line Valid
FVAL	TxIN25	RxOUT25	Frame Valid	Frame Valid
DVAL	TxIN26	RxOUT26	Not Used	Not Used
Spare	TxIN23	RxOUT23	Not Used	Not Used
Strobe	TxINCLK	RxOUTCLK	Pixel Clock	Pixel Clock

Table 2-6: Bit Assignments for Plug 1, Transmitter X in an A403k (Medium Configuration)

A403k, Plug No. 2, Transmitter Y				
Port	Camera	Frame Grabber	Signal	
			4 Tap 8 Bit	4 Tap 10 Bit
Port D0	TxIN0	RxOUT0	D3 Bit 0	D3 Bit 0
Port D1	TxIN1	RxOUT1	D3 Bit 1	D3 Bit 1
Port D2	TxIN2	RxOUT2	D3 Bit 2	D3 Bit 2
Port D3	TxIN3	RxOUT3	D3 Bit 3	D3 Bit 3
Port D4	TxIN4	RxOUT4	D3 Bit 4	D3 Bit 4
Port D5	TxIN6	RxOUT6	D3 Bit 5	D3 Bit 5
Port D6	TxIN27	RxOUT27	D3 Bit 6	D3 Bit 6
Port D7	TxIN5	RxOUT5	D3 Bit 7 (MSB)	D3 Bit 7
Port E0	TxIN7	RxOUT7	Not Used	D2 Bit 0
Port E1	TxIN8	RxOUT8	Not Used	D2 Bit 1
Port E2	TxIN9	RxOUT9	Not Used	D2 Bit 2
Port E3	TxIN12	RxOUT12	Not Used	D2 Bit 3
Port E4	TxIN13	RxOUT13	Not Used	D2 Bit 4
Port E5	TxIN14	RxOUT14	Not Used	D2 Bit 5
Port E6	TxIN10	RxOUT10	Not Used	D2 Bit 6
Port E7	TxIN11	RxOUT11	Not Used	D2 Bit 7
Port F0	TxIN15	RxOUT15	Not Used	D2 Bit 8
Port F1	TxIN18	RxOUT18	Not Used	D2 Bit 9 (MSB)
Port F2	TxIN19	RxOUT19	Not Used	Not Used
Port F3	TxIN20	RxOUT20	Not Used	Not Used
Port F4	TxIN21	RxOUT21	Not Used	D3 Bit 8
Port F5	TxIN22	RxOUT22	Not Used	D3 Bit 9 (MSB)
Port F6	TxIN16	RxOUT16	Not Used	Not Used
Port F7	TxIN17	RxOUT17	Not Used	Not Used
LVAL	TxIN24	RxOUT24	Line Valid	Line Valid
FVAL	TxIN25	RxOUT25	Frame Valid	Frame Valid
DVAL	TxIN26	RxOUT26	Not Used	Not Used
Spare	TxIN23	RxOUT23	Not Used	Not Used
Strobe	TxINCLK	RxOUTCLK	Pixel Clock	Pixel Clock

Table 2-7: Bit Assignments for Plug 2, Transmitter Y in an A403k (Medium Configuration)

A404k, Plug No. 1, Transmitter X					
Port	Camera	Frame Grabber	Signal		
			4 Tap 8 Bit	4 Tap 10 Bit	8 Taps 8 Bit
Port A0	TxIN0	RxOUT0	D0 Bit 0	D0 Bit 0	D0 Bit 0
Port A1	TxIN1	RxOUT1	D0 Bit 1	D0 Bit 1	D0 Bit 1
Port A2	TxIN2	RxOUT2	D0 Bit 2	D0 Bit 2	D0 Bit 2
Port A3	TxIN3	RxOUT3	D0 Bit 3	D0 Bit 3	D0 Bit 3
Port A4	TxIN4	RxOUT4	D0 Bit 4	D0 Bit 4	D0 Bit 4
Port A5	TxIN6	RxOUT6	D0 Bit 5	D0 Bit 5	D0 Bit 5
Port A6	TxIN27	RxOUT27	D0 Bit 6	D0 Bit 6	D0 Bit 6
Port A7	TxIN5	RxOUT5	D0 Bit 7 (MSB)	D0 Bit 7	D0 Bit 7 (MSB)
Port B0	TxIN7	RxOUT7	D1 Bit 0	D0 Bit 8	D1 Bit 0
Port B1	TxIN8	RxOUT8	D1 Bit 1	D0 Bit 9 (MSB)	D1 Bit 1
Port B2	TxIN9	RxOUT9	D1 Bit 2	Not Used	D1 Bit 2
Port B3	TxIN12	RxOUT12	D1 Bit 3	Not Used	D1 Bit 3
Port B4	TxIN13	RxOUT13	D1 Bit 4	D1 Bit 8	D1 Bit 4
Port B5	TxIN14	RxOUT14	D1 Bit 5	D1 Bit 9 (MSB)	D1 Bit 5
Port B6	TxIN10	RxOUT10	D1 Bit 6	Not Used	D1 Bit 6
Port B7	TxIN11	RxOUT11	D1 Bit 7 (MSB)	Not Used	D1 Bit 7 (MSB)
Port C0	TxIN15	RxOUT15	D2 Bit 0	D1 Bit 0	D2 Bit 0
Port C1	TxIN18	RxOUT18	D2 Bit 1	D1 Bit 1	D2 Bit 1
Port C2	TxIN19	RxOUT19	D2 Bit 2	D1 Bit 2	D2 Bit 2
Port C3	TxIN20	RxOUT20	D2 Bit 3	D1 Bit 3	D2 Bit 3
Port C4	TxIN21	RxOUT21	D2 Bit 4	D1 Bit 4	D2 Bit 4
Port C5	TxIN22	RxOUT22	D2 Bit 5	D1 Bit 5	D2 Bit 5
Port C6	TxIN16	RxOUT16	D2 Bit 6	D1 Bit 6	D2 Bit 6
Port C7	TxIN17	RxOUT17	D2 Bit 7 (MSB)	D1 Bit 7	D2 Bit 7 (MSB)
LVAL	TxIN24	RxOUT24	Line Valid	Line Valid	Line Valid
FVAL	TxIN25	RxOUT25	Frame Valid	Frame Valid	Frame Valid
DVAL	TxIN26	RxOUT26	Not Used	Not Used	Not Used
Spare	TxIN23	RxOUT23	Not Used	Not Used	Not Used
Strobe	TxINCLK	RxOUTCLK	Pixel Clock	Pixel Clock	Pixel Clock

Table 2-8: Bit Assignments for Plug 1, Transmitter X in an A404k (Full Configuration)

A404k, Plug No. 2, Transmitter Y					
Port	Camera	Frame Grabber	Signal		
			4 Tap 8 Bit	4 Tap 10 Bit	8 Tap 8 Bit
Port D0	TxIN0	RxOUT0	D3 Bit 0	D3 Bit 0	D3 Bit 0
Port D1	TxIN1	RxOUT1	D3 Bit 1	D3 Bit 1	D3 Bit 1
Port D2	TxIN2	RxOUT2	D3 Bit 2	D3 Bit 2	D3 Bit 2
Port D3	TxIN3	RxOUT3	D3 Bit 3	D3 Bit 3	D3 Bit 3
Port D4	TxIN4	RxOUT4	D3 Bit 4	D3 Bit 4	D3 Bit 4
Port D5	TxIN6	RxOUT6	D3 Bit 5	D3 Bit 5	D3 Bit 5
Port D6	TxIN27	RxOUT27	D3 Bit 6	D3 Bit 6	D3 Bit 6
Port D7	TxIN5	RxOUT5	D3 Bit 7 (MSB)	D3 Bit 7	D3 Bit 7 (MSB)
Port E0	TxIN7	RxOUT7	Not Used	D2 Bit 0	D4 Bit 0
Port E1	TxIN8	RxOUT8	Not Used	D2 Bit 1	D4 Bit 1
Port E2	TxIN9	RxOUT9	Not Used	D2 Bit 2	D4 Bit 2
Port E3	TxIN12	RxOUT12	Not Used	D2 Bit 3	D4 Bit 3
Port E4	TxIN13	RxOUT13	Not Used	D2 Bit 4	D4 Bit 4
Port E5	TxIN14	RxOUT14	Not Used	D2 Bit 5	D4 Bit 5
Port E6	TxIN10	RxOUT10	Not Used	D2 Bit 6	D4 Bit 6
Port E7	TxIN11	RxOUT11	Not Used	D2 Bit 7	D4 Bit 7 (MSB)
Port F0	TxIN15	RxOUT15	Not Used	D2 Bit 8	D5 Bit 0
Port F1	TxIN18	RxOUT18	Not Used	D2 Bit 9 (MSB)	D5 Bit 1
Port F2	TxIN19	RxOUT19	Not Used	Not Used	D5 Bit 2
Port F3	TxIN20	RxOUT20	Not Used	Not Used	D5 Bit 3
Port F4	TxIN21	RxOUT21	Not Used	D3 Bit 8	D5 Bit 4
Port F5	TxIN22	RxOUT22	Not Used	D3 Bit 9 (MSB)	D5 Bit 5
Port F6	TxIN16	RxOUT16	Not Used	Not Used	D5 Bit 6
Port F7	TxIN17	RxOUT17	Not Used	Not Used	D5 Bit 7 (MSB)
LVAL	TxIN24	RxOUT24	Line Valid	Line Valid	Line Valid
FVAL	TxIN25	RxOUT25	Frame Valid	Frame Valid	Frame Valid
DVAL	TxIN26	RxOUT26	Not Used	Not Used	Not Used
Spare	TxIN23	RxOUT23	Not Used	Not Used	Not Used
Strobe	TxINCLK	RxOUTCLK	Pixel Clock	Pixel Clock	Pixel Clock

Table 2-9: Bit Assignments for Plug 2, Transmitter Y in an A404k (Full Configuration)

A404k, Plug No. 2, Transmitter Z					
Port	Camera	Frame Grabber	Signal		
			4 Tap 8 Bit	4 Tap 10 Bit	8 Tap 8 Bit
Port G0	TxIN0	RxOUT0	Not Used	Not Used	D6 Bit 0
Port G1	TxIN1	RxOUT1	Not Used	Not Used	D6 Bit 1
Port G2	TxIN2	RxOUT2	Not Used	Not Used	D6 Bit 2
Port G3	TxIN3	RxOUT3	Not Used	Not Used	D6 Bit 3
Port G4	TxIN4	RxOUT4	Not Used	Not Used	D6 Bit 4
Port G5	TxIN6	RxOUT6	Not Used	Not Used	D6 Bit 5
Port G6	TxIN27	RxOUT27	Not Used	Not Used	D6 Bit 6
Port G7	TxIN5	RxOUT5	Not Used	Not Used	D6 Bit 7 (MSB)
Port H0	TxIN7	RxOUT7	Not Used	Not Used	D7 Bit 0
Port H1	TxIN8	RxOUT8	Not Used	Not Used	D7 Bit 1
Port H2	TxIN9	RxOUT9	Not Used	Not Used	D7 Bit 2
Port H3	TxIN12	RxOUT12	Not Used	Not Used	D7 Bit 3
Port H4	TxIN13	RxOUT13	Not Used	Not Used	D7 Bit 4
Port H5	TxIN14	RxOUT14	Not Used	Not Used	D7 Bit 5
Port H6	TxIN10	RxOUT10	Not Used	Not Used	D7 Bit 6
Port H7	TxIN11	RxOUT11	Not Used	Not Used	D7 Bit 7 (MSB)
Spare	TxIN15	RxOUT15	Not Used	Not Used	Not Used
Spare	TxIN18	RxOUT18	Not Used	Not Used	Not Used
Spare	TxIN19	RxOUT19	Not Used	Not Used	Not Used
Spare	TxIN20	RxOUT20	Not Used	Not Used	Not Used
Spare	TxIN21	RxOUT21	Not Used	Not Used	Not Used
Spare	TxIN22	RxOUT22	Not Used	Not Used	Not Used
Spare	TxIN16	RxOUT16	Not Used	Not Used	Not Used
Spare	TxIN17	RxOUT17	Not Used	Not Used	Not Used
LVAL	TxIN24	RxOUT24	Line Valid	Line Valid	Line Valid
FVAL	TxIN25	RxOUT25	Frame Valid	Frame Valid	Frame Valid
DVAL	TxIN26	RxOUT26	Not Used	Not Used	Not Used
Spare	TxIN23	RxOUT23	Not Used	Not Used	Not Used
Strobe	TxINCLK	RxOUTCLK	Pixel Clock	Pixel Clock	Pixel Clock

Table 2-10: Bit Assignments for Plug 2, Transmitter Z in an A404k (Full Configuration)

2.5.5 Video Data Output for the A402k

Depending on the video data output mode selected, A402k cameras output pixel data in either a 2 tap 10 bit, or a 2 tap 8 bit video data stream.

In 2 tap 10 bit mode, on each clock cycle, the camera transmits data for two pixels at 10 bit depth, a frame valid bit and a line valid bit. In 2 tap 8 bit mode, on each clock cycle, the camera transmits data for two pixels at 8 bit depth, a frame valid bit and a line valid bit. The assignment of the bits is shown in Table 2-5.

The pixel clock is used to time data sampling and transmission. As shown in Figures 2-8 and 2-9, the camera samples and transmits data on each rising edge of the pixel clock.

The frame valid bit indicates that a valid frame is being transmitted. The line valid bit indicates that a valid line is being transmitted. Pixel data is only valid when the frame valid bit and the line valid bit are both high.

The image has a maximum size of 2352 x 1726 pixels. Pixels are transmitted at a pixel clock frequency of 50 MHz over the Camera Link X transmitter. With each clock cycle, two pixels are transmitted in parallel at a depth of 10 or 8 bits. Therefore, one line takes a maximum of 1176 clock cycles to be transmitted.

The image is transmitted line by line from top left to bottom right. Frame Valid (FVAL) and Line Valid (LVAL) mark the beginning and duration of frame and line.

In 10 bit mode, all bits of data output from each 10-bit ADC are transmitted. In 8 bit mode, the two least significant bits output from each ADC are dropped and the 8 most significant bits of data per pixel are transmitted.

The data sequence outlined below, along with Figures 2-8 and 2-9, describe what is happening at the inputs to the Camera Link transmitter in the camera.

Note that the timing used for sampling the data at the Camera Link receiver in the frame grabber varies from device to device. On some receivers, data must be sampled on the rising edge of the pixel clock (receive clock), and on others, it must be sampled on the falling edge. Also, some devices are available which let you select either rising edge or falling edge sampling. Please consult the data sheet for the receiver that you are using for specific timing information.

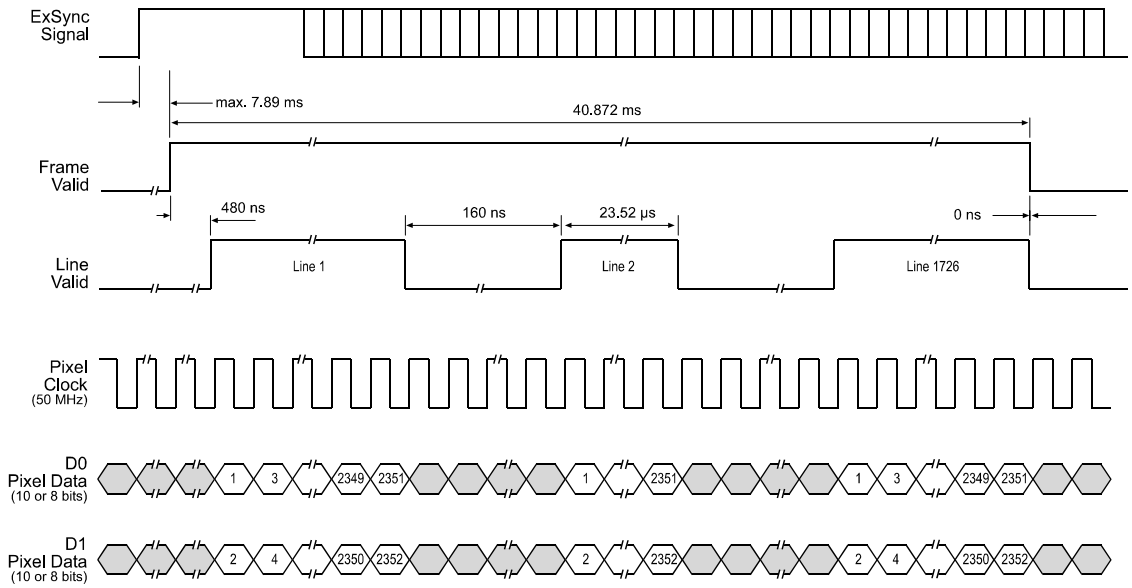
Video Data Sequence for the A402k

When the camera is not transmitting valid data, the frame valid and line valid bits sent on each cycle of the pixel clock will be low. The camera can begin capturing a new frame while it is sending data for a previously captured frame. It can also capture a frame and then send it before beginning capture of a new frame. When frame valid becomes high, the camera starts to send valid data:

- On the pixel clock cycle where frame data transmission begins, the frame valid bit will become high. 24 pixel clocks (480 ns) later, the line valid bit will become high.
- On the pixel clock cycle where data transmission for line one begins, the line valid bit will become high. Two data streams, D0 and D1, are transmitted in parallel during this clock cycle. On this clock cycle, data stream D0 will transmit data for pixel one in line one and data stream D1 will transmit data for pixel two in line one. Depending on the video data output mode selected, the pixel data will be at either 10 bit or 8 bit depth.
- On the next cycle of the pixel clock, the line valid bit will be high. On this clock cycle, data stream D0 will transmit data for pixel three in line one and data stream D1 will transmit data for pixel four in line one.

- On the next cycle of the pixel clock, the line valid bit will be high. On this clock cycle, data stream D0 will transmit data for pixel five in line one and data stream D1 will transmit data for pixel six in line one.
- This pattern will continue until all of the pixel data for line one has been transmitted. (A total of 1176 cycles.)
- Line valid becomes low for eight pixel clocks.
- On the pixel clock cycle where data transmission for line two begins, the line valid bit will become high. On this clock cycle, data stream D0 will transmit data for pixel one in line two and data stream D1 will transmit data for pixel two in line two.
- On the next cycle of the pixel clock, the line valid bit will be high. On this clock cycle, data stream D0 will transmit data for pixel three in line two and data stream D1 will transmit data for pixel four in line two.
- On the next cycle of the pixel clock, the line valid bit will be high. On this clock cycle, data stream D0 will transmit data for pixel five in line two and data stream D1 will transmit data for pixel six in line two.
- This pattern will continue until all of the pixel data for line two has been transmitted. (A total of 1176 cycles.)
- After all of the pixels in line two have been transmitted, the line valid bit will become low for eight cycles indicating that valid data for line two is no longer being transmitted.
- The camera will continue to transmit pixel data for each line as described above until all of the lines in the frame have been transmitted. After all of the lines have been transmitted, the frame valid bit and the line valid will become low indicating that a valid frame is no longer being transmitted.

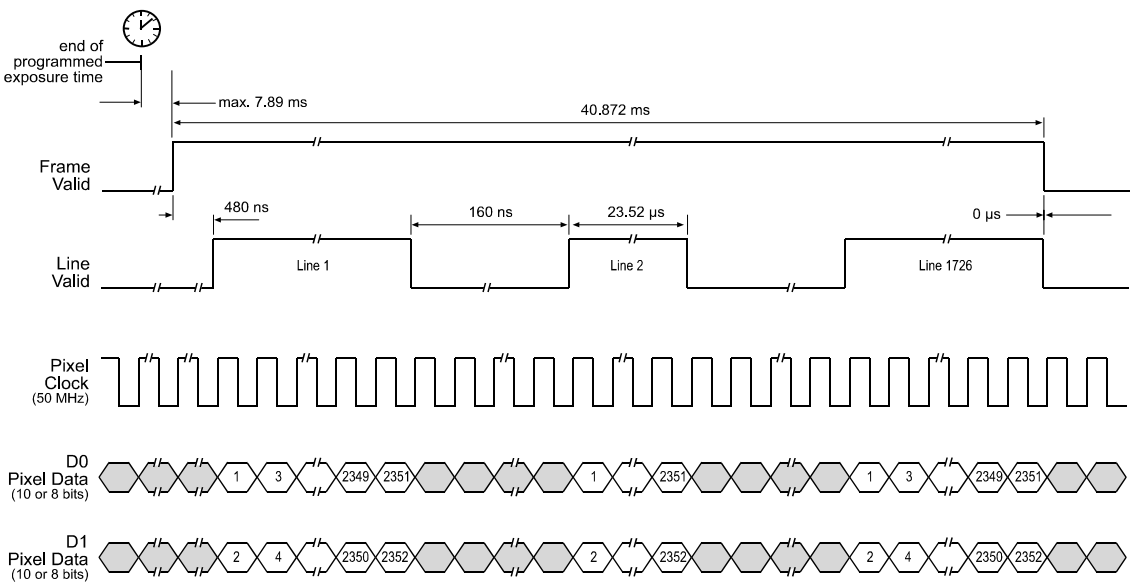
Figure 2-8 shows the data sequence when the camera is operating in edge-controlled or level-controlled exposure mode and Figure 2-9 shows the data sequence when the camera is operating in programmable exposure mode.



TIMING DIAGRAMS ARE NOT DRAWN TO SCALE.

The diagram assumes that the area of interest feature is not being used. With the area of interest feature enabled, the number of pixels transferred could be smaller.

Figure 2-8: A402k 2 Tap Output with Edge or Level Controlled Exposure



TIMING DIAGRAMS ARE NOT DRAWN TO SCALE.

The diagram assumes that the area of interest feature is not being used. With the area of interest feature enabled, the number of pixels transferred could be smaller.

Figure 2-9: A402k 2 Tap Output with Programmable Exposure

2.5.6 Video Data Output for the A403k

Depending on the video data output mode selected, A403k cameras output pixel data in either a 4 tap 10 bit, or a 4 tap 8 bit video data stream.

In 4 tap 10 bit mode, on each clock cycle, the camera transmits data for four pixels at 10 bit depth, a frame valid bit and a line valid bit. In 4 tap 8 bit mode, on each clock cycle, the camera transmits data for four pixels at 8 bit depth, a frame valid bit and a line valid bit. The assignment of the bits is shown in Tables 2-6 and 2-7.

The pixel clock is used to time data sampling and transmission. As shown in Figures 2-10 and 2-11, the camera samples and transmits data on each rising edge of the pixel clock.

The frame valid bit indicates that a valid frame is being transmitted. The line valid bit indicates that a valid line is being transmitted. Pixel data is only valid when the frame valid bit and the line valid bit are both high.

The image has a maximum size of 2352 x 1726 pixels. Pixels are transmitted at a pixel clock frequency of 50 MHz over the Camera Link X and Y transmitters. With each clock cycle, four pixels are transmitted in parallel at a depth of 10 or 8 bits. Therefore, one line takes a maximum of 588 clock cycles to be transmitted.

The image is transmitted line by line from top left to bottom right. Frame Valid (FVAL) and Line Valid (LVAL) mark the beginning and duration of frame and line.

In 10 bit mode, all bits of data output from each 10-bit ADC are transmitted. In 8 bit mode, the two least significant bits output from each ADC are dropped and the 8 most significant bits of data per pixel are transmitted.

The data sequence outlined below, along with Figures 2-10 and 2-11, describe what is happening at the inputs to the Camera Link transmitters in the camera.

Note that the timing used for sampling the data at the Camera Link receivers in the frame grabber varies from device to device. On some receivers, data must be sampled on the rising edge of the pixel clock (receive clock), and on others, it must be sampled on the falling edge. Also, some devices are available which let you select either rising edge or falling edge sampling. Please consult the data sheet for the receiver that you are using for specific timing information.

Video Data Sequence for the A403k

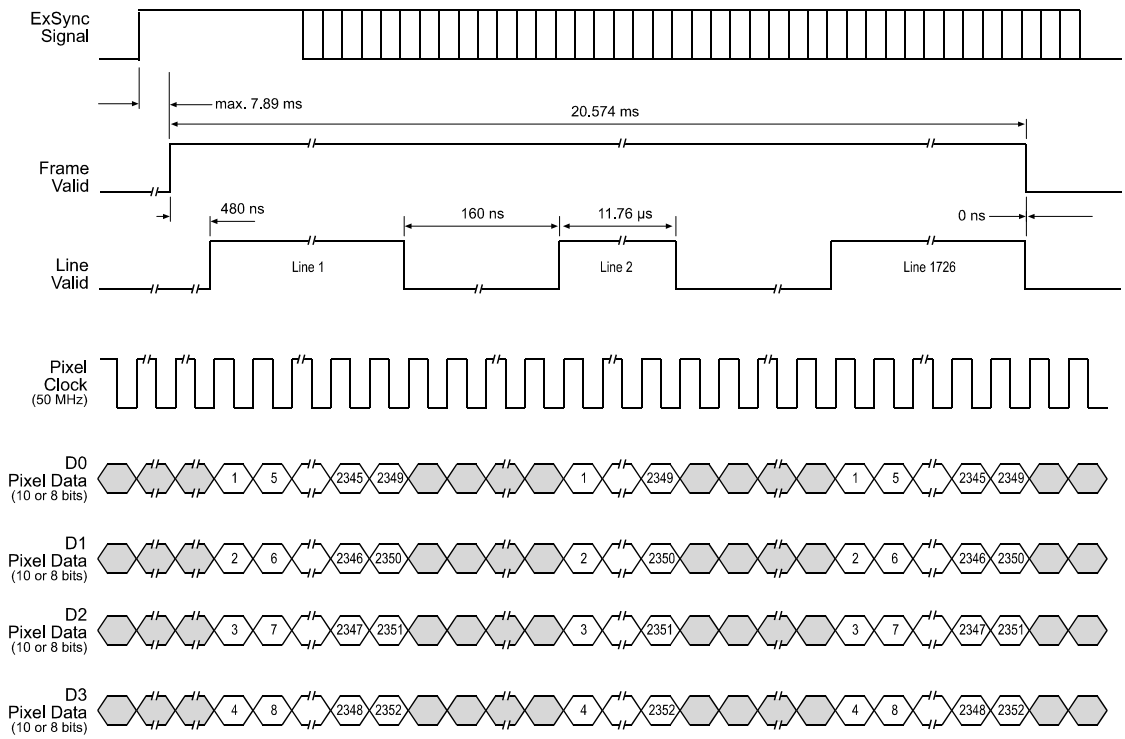
When the camera is not transmitting valid data, the frame valid and line valid bits sent on each cycle of the pixel clock will be low. The camera can begin capturing a new frame while it is sending data for a previously captured frame. It can also capture a frame and then send it before beginning capture of a new frame. When frame valid becomes high, the camera starts to send valid data:

- On the pixel clock cycle where frame data transmission begins, the frame valid bit will become high. 24 pixel clocks (480 ns) later, the line valid bit will become high.
- On the pixel clock cycle where data transmission for line one begins, the line valid bit will become high. Four data streams, D0, D1, D2 and D3 are transmitted in parallel during this clock cycle. On this clock cycle, data stream D0 will transmit data for pixel one in line one. Data stream D1 will transmit data for pixel two in line one. Data stream D2 will transmit data for pixel three in line one. And data stream D3 will transmit data for pixel four in line one. Depending on the video data output mode selected, the pixel data will be at either 10 bit or 8 bit depth.
- On the next cycle of the pixel clock, the line valid bit will be high. On this clock cycle, data stream D0 will transmit data for pixel five in line one. Data stream D1 will transmit data for

pixel six in line one. Data stream D2 will transmit data for pixel seven in line one. And data stream D3 will transmit data for pixel eight in line one.

- On the next cycle of the pixel clock, the line valid bit will be high. On this clock cycle, data stream D0 will transmit data for pixel nine in line one. Data stream D1 will transmit data for pixel ten in line one. Data stream D2 will transmit data for pixel eleven in line one. And data stream D3 will transmit data for pixel twelve in line one.
- This pattern will continue until all of the pixel data for line one has been transmitted. (A total of 588 cycles.)
- Line valid becomes low for eight pixel clocks.
- On the pixel clock cycle where data transmission for line two begins, the line valid bit will become high. On this clock cycle, data stream D0 will transmit data for pixel one in line two. Data stream D1 will transmit data for pixel two in line two. Data stream D2 will transmit data for pixel three in line two. And data stream D3 will transmit data for pixel four in line two.
- On the next cycle of the pixel clock, the line valid bit will be high. On this clock cycle, data stream D0 will transmit data for pixel five in line two. Data stream D1 will transmit data for pixel six in line two. Data stream D2 will transmit data for pixel seven in line two. And data stream D3 will transmit data for pixel eight in line two.
- On the next cycle of the pixel clock, the line valid bit will be high. On this clock cycle, data stream D0 will transmit data for pixel nine in line two. Data stream D1 will transmit data for pixel ten in line two. Data stream D2 will transmit data for pixel eleven in line two. And data stream D3 will transmit data for pixel twelve in line two.
- This pattern will continue until all of the pixel data for line two has been transmitted. (A total of 588 cycles.)
- After all of the pixels in line two have been transmitted, the line valid bit will become low for eight cycles indicating that valid data for line two is no longer being transmitted.
- The camera will continue to transmit pixel data for each line as described above until all of the lines in the frame have been transmitted. After all of the lines have been transmitted, the frame valid bit and the line valid will become low indicating that a valid frame is no longer being transmitted.

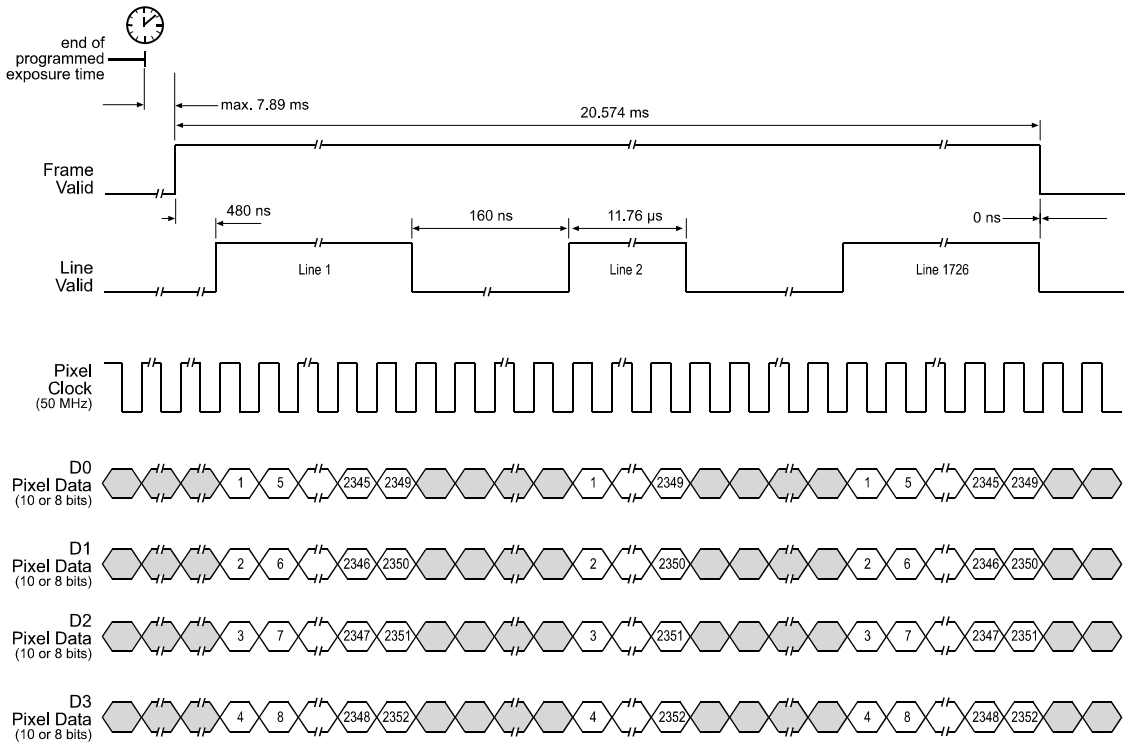
Figure 2-10 shows the data sequence when the camera is operating in edge-controlled or level-controlled exposure mode and Figure 2-11 shows the data sequence when the camera is operating in programmable exposure mode.



TIMING DIAGRAMS ARE NOT DRAWN TO SCALE.

The diagram assumes that the area of interest feature is not being used. With the area of interest feature enabled, the number of pixels transferred could be smaller.

Figure 2-10: A403k or A404k 4 Tap Output with Edge or Level Controlled Exposure



TIMING DIAGRAMS ARE NOT DRAWN TO SCALE.

The diagram assumes that the area of interest feature is not being used. With the area of interest feature enabled, the number of pixels transferred could be smaller.

Figure 2-11: A403k or A404k 4 Tap Output with Programmable Exposure

2.5.7 Video Data Output for the A404k

Depending on the video data output mode selected, A404k cameras output pixel data in either a 4 tap 10 bit, a 4 tap 8 bit or an 8 tap 8 bit video data stream.

2.5.7.1 4 Tap 10 Bit and 4 Tap 8 Bit Output Modes

In 4 tap 10 bit mode, on each clock cycle, the camera transmits data for four pixels at 10 bit depth, a frame valid bit and a line valid bit. In 4 tap 8 bit mode, on each clock cycle, the camera transmits data for four pixels at 8 bit depth, a frame valid bit and a line valid bit. The assignment of the bits is shown in Tables 2-8 and 2-9.

In 10 bit mode, all bits of data output from each 10-bit ADC are transmitted. In 8 bit mode, the two least significant bits output from each ADC are dropped and the 8 most significant bits of data per pixel are transmitted.

The video data output sequence for an A404k camera operating in 4 tap 10 bit or 4 tap 8 bit output mode is similar to the output sequence of an A403k camera operating in 4 tap 10 bit or 4 tap 8 bit output mode. Refer to Section 2.5.6 and Figures 2-10 and 2-11 for a description of the A403k video data output sequence.

2.5.7.2 8 Tap 8 Bit Output Mode

In 8 tap output mode, on each clock cycle, the camera transmits data for eight pixels at 8 bit depth, a frame valid bit and a line valid bit. The assignment of the bits is shown in Tables 2-8, 2-9 and 2-10.

The pixel clock is used to time data sampling and transmission. As shown in Figures 2-12 and 2-13, the camera samples and transmits data on each rising edge of the pixel clock.

The frame valid bit indicates that a valid frame is being transmitted. The line valid bit indicates that a valid line is being transmitted. Pixel data is only valid when the frame valid bit and the line valid bit are both high.

The image has a maximum size of 2352 x 1726 pixels. Pixels are transmitted at a pixel clock frequency of 50 MHz over the Camera Link X, Y, and Z transmitters. With each clock cycle, eight pixels are transmitted in parallel at a depth of 8 bits. Therefore, one line takes a maximum of 294 clock cycles to be transmitted.

The image is transmitted line by line from top left to bottom right. Frame Valid (FVAL) and Line Valid (LVAL) mark the beginning and duration of frame and line.

The data sequence outlined below, along with Figures 2-12 and 2-13, describe what is happening at the inputs to the Camera Link transmitters in the camera.

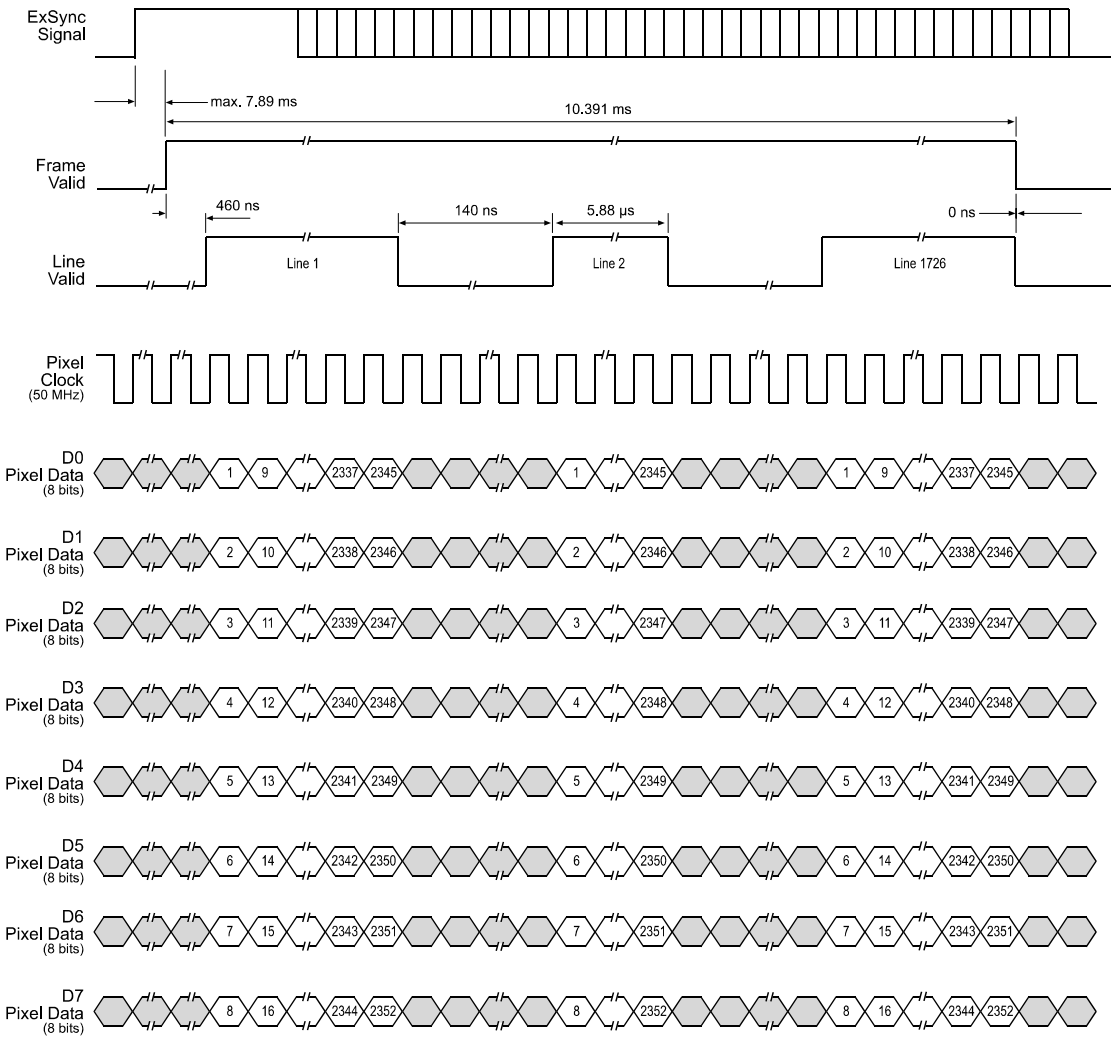
Note that the timing used for sampling the data at the Camera Link receivers in the frame grabber varies from device to device. On some receivers, data must be sampled on the rising edge of the pixel clock (receive clock), and on others, it must be sampled on the falling edge. Also, some devices are available which let you select either rising edge or falling edge sampling. Please consult the data sheet for the receiver that you are using for specific timing information.

Video Data Sequence for the A404k in an 8 Tap Output Mode

When the camera is not transmitting valid data, the frame valid and line valid bits sent on each cycle of the pixel clock will be low. The camera can begin capturing a new frame while it is sending data for a previously captured frame. It can also capture a frame and then send it before beginning capture of a new frame. When frame valid becomes high, the camera starts to send valid data:

- On the pixel clock cycle where frame data transmission begins, the frame valid bit will become high. 23 pixel clocks (460 ns) later, the line valid bit will become high.
- On the pixel clock cycle where data transmission for line one begins, the line valid bit will become high. Eight data streams, D0 through D7, are transmitted in parallel during this clock cycle. On this clock cycle, data stream D0 will transmit data for pixel one in line one. Data stream D1 will transmit data for pixel two in line one. Data stream D2 will transmit data for pixel three in line one. Data stream D3 will transmit data for pixel four in line one. Data stream D4 will transmit data for pixel five in line one. Data stream D5 will transmit data for pixel six in line one. Data stream D6 will transmit data for pixel seven in line one. Data stream D7 will transmit data for pixel eight in line one.
- On the next cycle of the pixel clock, the line valid bit will be high. On this clock cycle, data stream D0 will transmit data for pixel nine in line one. Data stream D1 will transmit data for pixel ten in line one. Data stream D2 will transmit data for pixel eleven in line one. Data stream D3 will transmit data for pixel twelve in line one. Data stream D4 will transmit data for pixel thirteen in line one. Data stream D5 will transmit data for pixel fourteen in line one. Data stream D6 will transmit data for pixel fifteen in line one. Data stream D7 will transmit data for pixel sixteen in line one.
- This pattern will continue until all of the pixel data for line one has been transmitted. (A total of 294 cycle.)
- Line valid becomes low for seven pixel clocks.
- On the pixel clock cycle where data transmission for line two begins, the line valid bit will become high. On this clock cycle, data stream D0 will transmit data for pixel one in line two. Data stream D1 will transmit data for pixel two in line two. Data stream D2 will transmit data for pixel three in line two. Data stream D3 will transmit data for pixel four in line two. Data stream D4 will transmit data for pixel five in line two. Data stream D5 will transmit data for pixel six in line two. Data stream D6 will transmit data for pixel seven in line two. Data stream D7 will transmit data for pixel eight in line two.
- On the next cycle of the pixel clock, the line valid bit will be high. On this clock cycle, data stream D0 will transmit data for pixel nine in line two. Data stream D1 will transmit data for pixel ten in line two. Data stream D2 will transmit data for pixel eleven in line two. Data stream D3 will transmit data for pixel twelve in line two. Data stream D4 will transmit data for pixel thirteen in line two. Data stream D5 will transmit data for pixel fourteen in line two. Data stream D6 will transmit data for pixel fifteen in line two. Data stream D7 will transmit data for pixel sixteen in line two.
- This pattern will continue until all of the pixel data for line two has been transmitted. (A total of 294 cycles.)
- After all of the pixels in line two have been transmitted, the line valid bit will become low for eight cycles indicating that valid data for line two is no longer being transmitted.
- The camera will continue to transmit pixel data for each line as described above until all of the lines in the frame have been transmitted. After all of the lines have been transmitted, the frame valid bit and the line valid will become low indicating that a valid frame is no longer being transmitted.

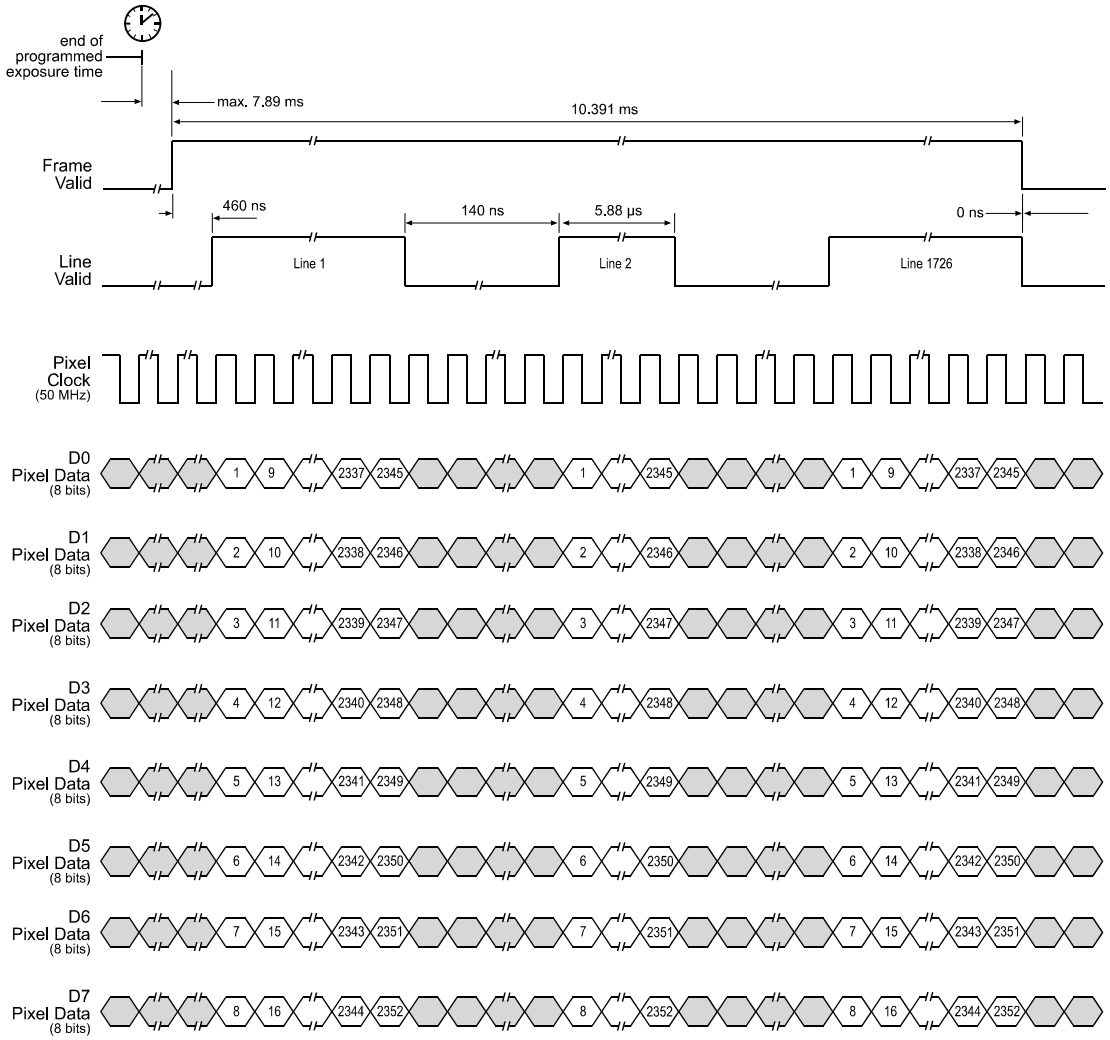
Figure 2-12 shows the data sequence when the camera is operating in edge-controlled or level-controlled exposure mode and figure 2-13 shows the data sequence when the camera is operating in programmable exposure mode.



TIMING DIAGRAMS ARE NOT DRAWN TO SCALE.

The diagram assumes that the area of interest feature is not being used. With the area of interest feature enabled, the number of pixels transferred could be smaller.

Figure 2-12: A404k 8 Tap Output with Edge or Level Controlled Exposure



TIMING DIAGRAMS ARE NOT DRAWN TO SCALE.

The diagram assumes that the area of interest feature is not being used. With the area of interest feature enabled, the number of pixels transferred could be smaller.

Figure 2-13: A404k 8 Tap Output with Programmable Exposure

2.5.8 Flash Trigger Signal

A400k cameras output a flash trigger signal that can be used to trigger a flash exposure. The flash trigger output connector is described in Section 2.1.4.

The flash trigger signal can be programmed to operate in five different modes:

- The signal is always *low*, that is, deactivated.
- The signal is always *high*.
- The signal is *high* as long as the sensor's flash window is open, that is, all pixel lines are exposed to light. The signal goes high when exposure starts in the last pixel line of the area of interest and the signal goes low when exposure ends in the first pixel line.
- The signal is *low* as long as the sensor's flash window is open, that is, all pixel lines are exposed to light. The signal goes low when exposure starts in the last pixel line of the area of interest and the signal goes high when exposure ends in the first pixel line.
- The signal is tied to the ExFlash input signal provided by the frame grabber and the signal is *high* while the ExFlash signal from the frame grabber is high.
- The signal is tied to the ExFlash input signal provided by the frame grabber and the signal is *low* while the ExFlash signal from the frame grabber is high.

In addition, four switching options are programmable:

- TTL
- Open collector or Low Side Switch, 5 V max
- High Side Switch 5 V
- High Impedance

The switching options are explained on page 2-5.

If the exposure time setting on the camera is lower than the minimum flash exposure required (see page 3-9), no flash trigger signal will be output.

2.5.8.1 Setting the Flash Trigger Signal

You can set the flash trigger signal using either the Camera Configuration Tool Plus (CCT+) or binary commands.

With the CCT+

With the CCT+ (see Section 4.1), you use the settings in the Flash Trigger parameter group.

By Setting CSRs

You can program the flash trigger signal by writing a value to the Mode field of the Flash Trigger Output Mode CSR and Flash Trigger Switching Mode CSR (see page 4-31).

See Section 4.2.2 for an explanation of CSRs. See Section 4.3.1 for an explanation of using read/write commands.

2.6 RS-644 Serial Communication

The A400k is equipped for RS-644 serial communication via the frame grabber as specified in the Camera Link standard. The RS-644 serial connection (SerTC/SerTFG) in the Camera Link interface is used to issue commands to the camera for changing modes and parameters. The serial link can also be used to query the camera about its current setup.

The Basler Camera Configuration Tool Plus (CCT+) is a convenient, graphical interface that can be used to change camera modes and parameters via the serial connection. The configuration tool is installed as part of the camera installation procedure shown in the booklet that is shipped with the camera. Section 4.1 provides some basic information about the configuration tool. Detailed instructions for using the tool are included in the on-line help file that is installed with the tool.

Basler has also developed a binary read/write command format that can be used to change camera modes and parameters directly from your own application via the serial connection using the API delivered with the frame grabber. See Section 4.3 for details on the binary read/write command format.

2.6.1 Making the Serial Connection

Frame grabbers compliant with the Camera Link specification are equipped with a serial port integrated into the Camera Link interface that can be used for RS-644 serial communication. The characteristics of the serial port can vary from manufacturer.

If you are using the Basler CCT+ to configure the camera, the tool will detect the characteristics of the serial port on the frame grabber and will determine the appropriate settings so that the tool can open and use the port.

In order for the Camera Configuration Tool Plus to detect and use the port, the characteristics of the port must comply with the Camera Link standard and the DLL called for in the standard must be present.

When the camera is powered on or when a camera reset is performed, your PC may receive some random characters on the serial interface. We recommend clearing the serial input buffers in your PC after a camera power on or reset.

If you are configuring the camera using binary commands from within your application software, your software must be able to access the frame grabber serial port and to determine the appropriate settings so that it can open and use the port. Please consult your frame grabber's documentation to determine the port access method and the port characteristics.

2.7 Converting Camera Link Output to RS-644 with a k-BIC (A402k Only)

On the A400k, video data is output from the camera in Camera Link LVDS format and parameter change commands are issued to the camera using RS-644 serial communication via the frame grabber. On older cameras, video data was output using an RS-644 LVDS format and commands were issued using RS-232 serial communication via the host PC. The output from A402k cameras can be converted to the older style of output by using a Basler Interface Converter for k-series cameras (k-BIC). The k-BIC is a small device which attaches to the A402k with a Camera Link compatible cable. For complete information on the k-BIC, refer to the k-BIC User's Manual and the k-BIC Installation Guide that are available at www.basler-vc.com.

2.8 DC Power

A400k cameras require 12 VDC (±10%) power. The maximum power consumption is 6.5 / 7.0 / 7.5 W for the A402k / A403k / A404k respectively. The maximum current during constant operation is 833 mA. Peak currents may occur. We recommend to use 1.5 A power supplies. Ripple must be less than 1%.

Also, note the information about the 6-pin connector in Section 2.1.3 and on the power cable in Section 2.2.2.

A Hirose plug will be shipped with each camera. This plug should be used to connect the power supply cable to the camera.

For proper EMI protection, the power supply cable attached to this plug must be a twin-core shielded cable. Also, the housing of the Hirose plug must be connected to the cable shield and the cable shield must be connected to earth ground at the power supply.

Make sure that the polarity is correct.

Caution!

Be sure that all power to your system is switched off before you make or break connections to the camera. Making or breaking connections when power is on can result in damage to the camera.

If you can not switch off power, be sure that the power supply connector is the last connector plugged when you make connections to the camera, and the first connector unplugged when you break connections.

The camera is equipped with an undervoltage lockout. An input voltage below 10.8 VDC will cause the camera to automatically switch off.

The camera has no overvoltage protection. An input voltage higher than 13.2 VDC will damage the camera.

The camera is not protected for reverse voltage. The polarity of the input power to the camera must be as shown in Table 2-3. If reverse voltage is applied to the camera while it is connected to a frame grabber in a PC, the camera could be seriously damaged.

3 Basic Operation and Features

3.1 Functional Description

BASLER A400k area scan cameras employ a CMOS-sensor chip which provides features such as an electronic rolling shutter and electronic exposure time control. Exposure time is controlled either internally via an internal sync signal (free-run mode) or externally via an external trigger (ExSync) signal. The ExSync signal facilitates periodic or non-periodic pixel readout.

In any free-run mode, the camera generates its own internal control signal and the internal signal is used to control exposure and charge readout. When operating in free-run, the camera outputs frames continuously.

When exposure is controlled by an ExSync signal, exposure time can be either level-controlled or programmable. In level-controlled mode, charge is accumulated when the ExSync signal is low. The rising edge of ExSync triggers the readout. In programmable mode, exposure time can be programmed to a predetermined time period. In this case, exposure begins on the rising edge of ExSync and accumulated charges are read out when the programmed exposure time ends.

At readout, accumulated charges move out of the light-sensitive sensor elements (pixels). Moveout is clocked according to the camera's 50 MHz internal data rate. As the charges move out of the pixels, they are converted to voltages proportional to the size of each charge.

The sensor has a column-parallel analog-to-digital converter (ADC) architecture that lets the array of 2,352 ADCs on the chip digitize simultaneously the analog data from an entire line of pixels. The analog data is converted into 10-bit digital pixel data by the 10-bit ADCs (shown in Figure 3-1 on page 3-2). The digitized data is then stored in column parallel 10-bit ADC registers.

Now, the digitized pixel data is shifted in portions of 160 bits from the ADC registers to the output registers and output in ascending numerical order from pixel 1 through pixel 2,352 and from the first line through the last line via 16 output ports that each transmit 10-bit pixel data in parallel with each cycle. Finally, the output data is reformatted and transferred out of the camera as shown below:

- In the A402k, the data is reformatted to be output in two data streams in parallel (2 taps).
- In the A403k, the data is reformatted to be output in four data streams in parallel (4 taps).
- In the A404k, the data is reformatted to be output in four data streams in parallel (4 taps) or in eight data streams in parallel (8 taps).

The 8 bit or 10 bit video data is transmitted from the camera to the frame grabber using a Camera Link transmission format (see Section 2.5 for details). The camera can transmit video at an eight bit depth or a ten bit depth.

For optimal digitization, gain and offset are programmable via a serial port.

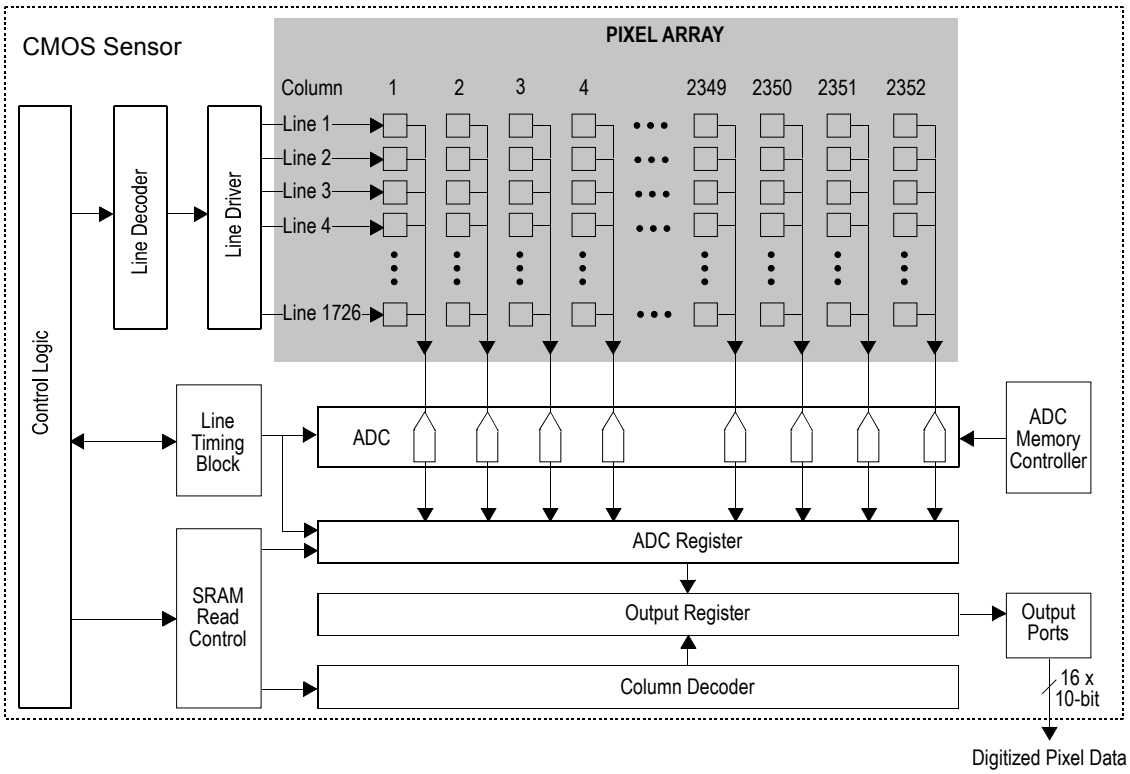


Figure 3-1: A400k Sensor Architecture

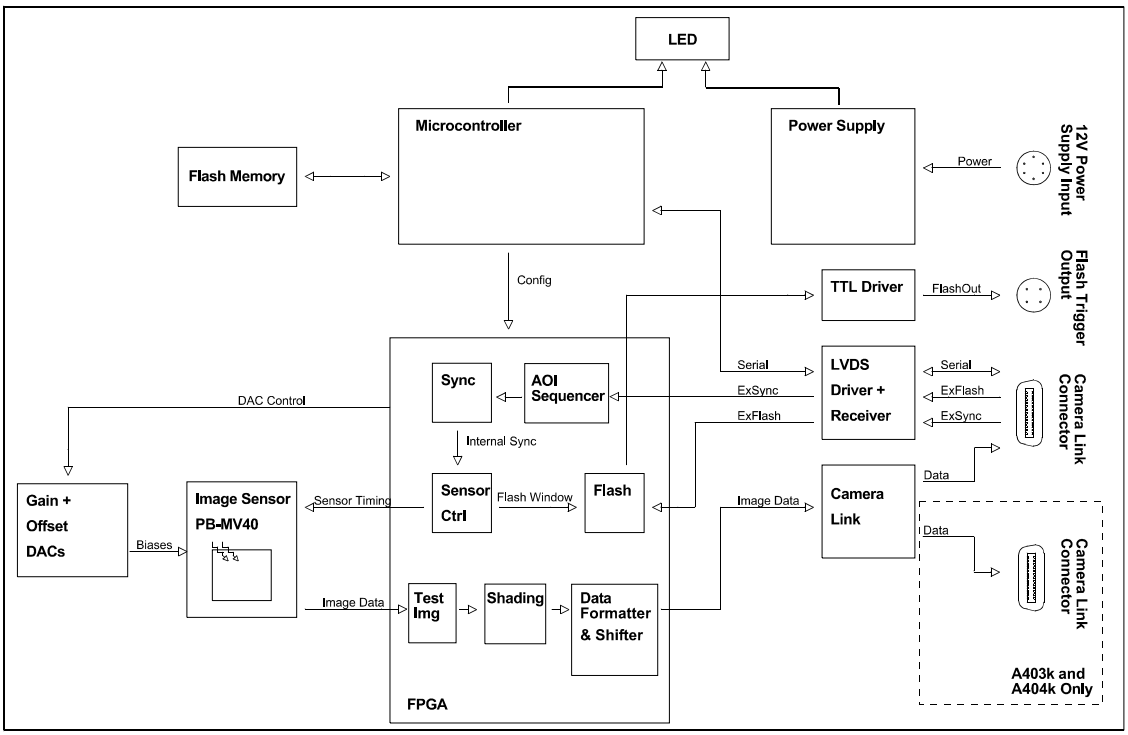


Figure 3-2: A400k Block Diagram

3.2 Video Data Output Modes

The A402k can output video data using two different modes: 2 tap 10 bit mode or 2 tap 8 bit mode. In 2 tap 10 bit mode, the camera outputs data for two pixels on each cycle of the pixel clock and the pixel data is at 10 bit depth. In 2 tap 8 bit mode, the camera outputs data for two pixels on each cycle of the pixel clock and the pixel data is at 8 bit depth. These modes are described in detail in Section 2.5.5.

The A403k can output video data using two different modes: 4 tap 10 bit mode or 4 tap 8 bit mode. In 4 tap 10 bit mode, the camera outputs data for four pixels on each cycle of the pixel clock and the pixel data is at 10 bit depth. In 4 tap 8 bit mode, the camera outputs data for four pixels on each cycle of the pixel clock and the pixel data is at 8 bit depth. These modes are described in detail in Section 2.5.6.

The A404k can output video data using three different modes: 4 tap 10 bit mode, 4 tap 8 bit mode or 8 tap 8 bit mode. In 4 tap 10 bit mode, the camera outputs data for four pixels on each cycle of the pixel clock and the pixel data is at 10 bit depth. In 4 tap 8 bit mode, the camera outputs data for four pixels on each cycle of the pixel clock and the pixel data is at 8 bit depth. In 8 tap 8 bit mode, the camera outputs data for eight pixels on each cycle of the pixel clock and the pixel data is at 8 bit depth. These modes are described in detail in Section 2.5.7.

You can select the video data output mode using either the Camera Configuration Tool Plus (see Section 4.1 and the configuration tool's on-line help) or binary commands (see Section 4.3). With the configuration tool, you use the **Video Data Output Mode** setting in the **Output** group to select the data output mode and with binary commands, you use the Video Data Output Mode binary command.

3.2.1 Setting the Video Data Output Mode

You can set the video data output mode by using the Camera Configuration Tool Plus (CCT+) or by using binary write commands from within your own application to set the camera's control and status registers (CSRs).

With the CCT+

With the CCT+ (see Section 4.1), you use the Video Data Output Mode setting in the Output parameter group to set the output mode.

By Setting CSRs

You can select the video data output mode by writing a value to the Mode field of the Video Data Output Mode CSR (see page 4-15).

See Section 4.2.2 for an explanation of CSRs. See Section 4.3.1 for an explanation of using read/write commands.

3.3 Exposure Time Control Modes

A400k cameras can operate under the control of an external trigger signal (ExSync signal) or can operate in “free-run”. In free-run, the camera generates its own internal control signal and does not require an ExSync signal.

3.3.1 ExSync Controlled Operation

3.3.1.1 Basics of ExSync Controlled Operation

In ExSync operation, the camera’s frame rate and exposure time are controlled by an externally generated (ExSync) signal. The ExSync signal is typically supplied to the camera by a frame grabber board. You should refer to the manual supplied with your frame grabber board to determine how to set up the ExSync signal that is being supplied to the camera.

When the camera is operating under the control of an ExSync signal, the length of the ExSync signal period determines the camera’s frame rate. (Frame rate = 1/Control signal period.)

ExSync can be periodic or non-periodic.

When the camera is operating with an ExSync signal, it has three modes of exposure time control available: edge-controlled mode, level-controlled mode and programmable mode.

- In **ExSync edge-controlled** mode, the pixels are exposed and charge is accumulated over the full period of the ExSync signal (rising edge to rising edge). The falling edge of the ExSync signal is irrelevant. The frame is read out and transferred on the rising edge of ExSync (see Figure 3-3).

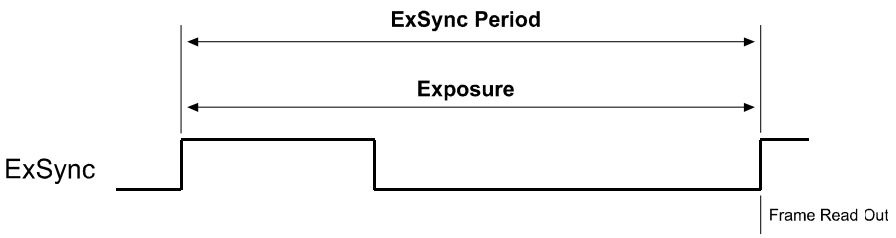


Figure 3-3: ExSync Edge-Controlled Mode

- In **ExSync level-controlled** mode, the exposure time is determined by the time between the falling edge of ExSync and the next rising edge. The pixels are exposed and charge is accumulated only when ExSync is low. The frame is read out and transferred on the rising edge of the ExSync signal (see Figure 3-4).

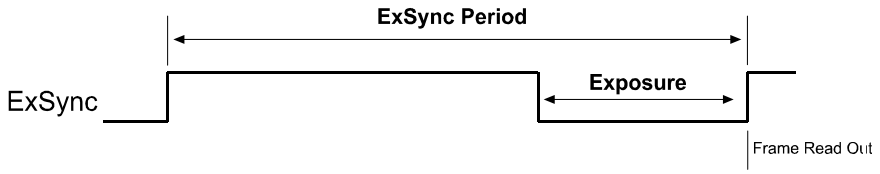


Figure 3-4: ExSync Level-controlled Mode

- In **ExSync programmable** mode, the rising edge of ExSync triggers exposure and charge accumulation for a pre-programmed period of time. The frame is read out and transferred at the end of the pre-programmed period. The length of the pre-programmed exposure period is determined by the exposure time setting.
  The falling edge of ExSync is irrelevant (see Figure 3-5).

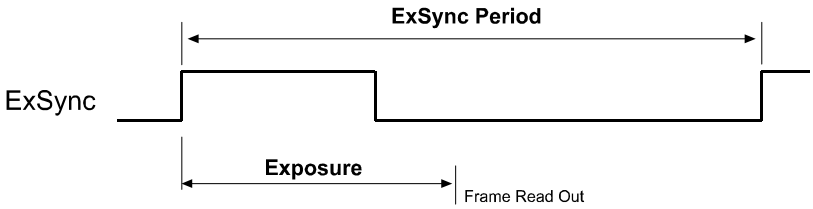


Figure 3-5: ExSync Programmable Mode

3.3.1.2 Guidelines When Using an ExSync Signal

In ExSync edge-controlled mode and programmable mode, minimum high time for the ExSync signal is 2 µs, minimum low time 2 µs. In ExSync level-controlled mode, minimum high time for the ExSync signal is 9.12 µs, minimum low time 4.56 µs.

In ExSync programmable mode, the minimum exposure time setting is 4.56 µs. Due to the sensor design, the exposure time can only be set in integer multiples of 4.56 µs, that is, 4.56 µs, 9.12 µs, 13.68 µs, and so on.

3.3.1.3 Selecting an ExSync Exposure Mode & Setting the Exposure Time

You can select an ExSync exposure time control mode and set the exposure time for the ExSync programmable mode by using the Camera Configuration Tool Plus (CCT+) or by using binary write commands from within your own application to set the camera's control and status registers (CSRs).

With the CCT+

With the CCT+ (see Section 4.1), you use the Exposure Time Control Mode setting in the Exposure parameter group to select the ExSync edge-controlled, ExSync level-controlled or ExSync programmable exposure time control mode. If you select the ExSync programmable mode, the CCT+ will also let you enter an exposure time.

By Setting CSRs

You can select the exposure time control mode by writing a value to the Mode field of the Exposure Time Control Mode CSR (see page 4-15).

If you select the ExSync programmable mode, you will also need to set the exposure time. You can set the exposure time by writing a value to the Raw Exposure Time field or to the Absolute Exposure Time field of the Exposure Time CSR (see page 4-16).

Section 4.2.2 explains CSRs and the difference between using the "raw" field and the "absolute" field in a CSR. Section 4.3.1 explains using read/write commands.

3.3.2 Free Run

3.3.2.1 Basics of Free-run Controlled Operation

In **free-run**, no ExSync signal is required. The camera generates a continuous internal control signal. When the camera is operating in free-run, it exposes and outputs frames continuously.

When the camera is operating in free-run, the length of the control signal period determines the camera's frame rate:

$$\text{Frame rate} = \frac{1}{\text{Control signal period}}$$

The control signal period is equal to the frame period setting.

When the camera is operating in free-run, it has two modes of exposure time control available: edge-controlled mode and programmable mode.

- In **free-run edge-controlled mode**, the camera generates a continuous internal control signal based on the "Frame Period" parameter. The pixels are exposed and charge is accumulated over the full period of the internal control signal (rising edge to rising edge). The falling edge of the control signal is irrelevant. The frame is read out and transferred on the rising edge of the internal control signal (see Figure 3-6).

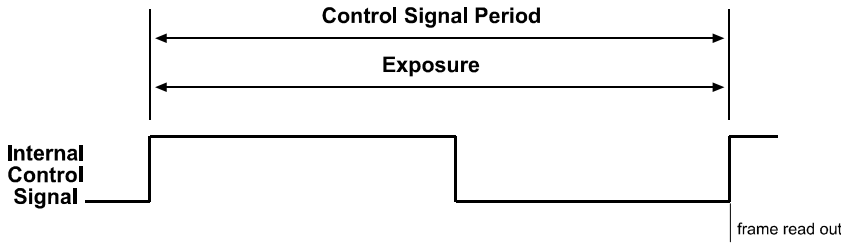


Figure 3-6: Free-run Edge-controlled Mode

- In **free-run programmable mode**, the camera generates a continuous internal control signal based on two programmable parameters: "Exposure Time" and "Frame Period". The Exposure Time setting determines how long the internal control signal will remain low. Pixels are exposed and charge is accumulated when the internal control signal is low. The Frame Period setting determines the control signal period. The frame is read out and transferred on the rising edge of the internal control signal. See Figure 3-7.

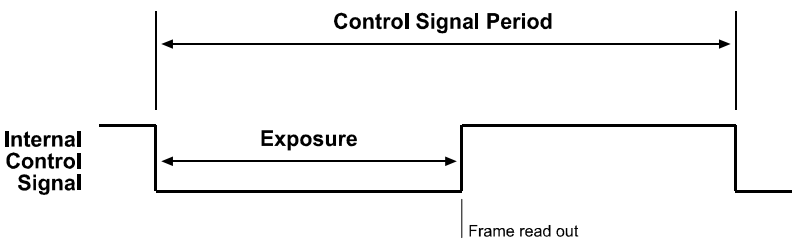


Figure 3-7: Free-run Programmable Mode

3.3.2.2 Guidelines When Using Free-run

In free-run programmable mode, the minimum exposure time setting is 4.56 µs. Due to the sensor design, the exposure time can only be set in integer multiples of 4.56 µs, that is, 4.56 µs, 9.12 µs, 13.68 µs, and so on.

In free-run programmable mode, also the following rule applies:

$$\text{Exposure time setting} \leq \text{Frame period setting}$$

3.3.2.3 Selecting a Free-run Exposure Mode, Setting the Frame Period, and Setting the Exposure Time

You can select a free-run exposure time control mode, set the frame period, and set the exposure time for the free-run programmable mode by using the Camera Configuration Tool Plus (CCT+) or by using binary write commands from within your own application to set the camera's control and status registers (CSRs).

With the CCT+

With the CCT+ (see Section 4.1), you use the Exposure Time Control Mode setting in the Exposure parameter group to select the free-run edge-controlled or free-run programmable exposure time control mode. If you select the free-run programmable mode, the CCT+ will also let you enter an exposure time.

By Setting CSRs

You can select the exposure time control mode by writing a value to the Mode field of the Exposure Time Control Mode CSR (see page 4-15).

You can set the frame period by writing a value to the Raw Frame Period field or to the Absolute Frame Period field of the Frame Period CSR (see page 4-18).

If you select the free-run programmable mode, you will also need to set the exposure time. You can set the exposure time by writing a value to the Raw Exposure Time field or to the Absolute Exposure Time field of the Exposure Time CSR (see page 4-16).

Section 4.2.2 explains CSRs and the difference between using the "raw" field and the "absolute" field in a CSR. Section 4.3.1 explains using read/write commands.

3.4 Rolling Shutter

A rolling shutter is used to control the start and stop of exposure. A rolling shutter requires less in-pixel transistors than a nonrolling shutter. This allows a larger photosensitive area per pixel, that is, a higher fill factor and thus, a higher sensitivity.

The rolling shutter resets, exposes and reads out the pixel lines with a temporal offset of 4.56 µs from one line to the next.

When exposure is triggered, the rolling shutter first resets the top line of pixels, then the second line, then the third line, and so on. The reset progresses down the image from one line to the next until the bottom line of pixels is reached (see Figure 3-8).

The time interval between a pixel line being reset and the pixel line being read out is the exposure time. Exposure time is the same for all lines and determined by the exposure time setting. Due to the pixel lines being reset and read out with an offset of 4.56 µs, the start of exposure has an offset of 4.56 µs from one line to next.

The sequence of pixel readout is timed identically to the reset, starting from the top line and moving down the image until readout of the bottom line is complete.

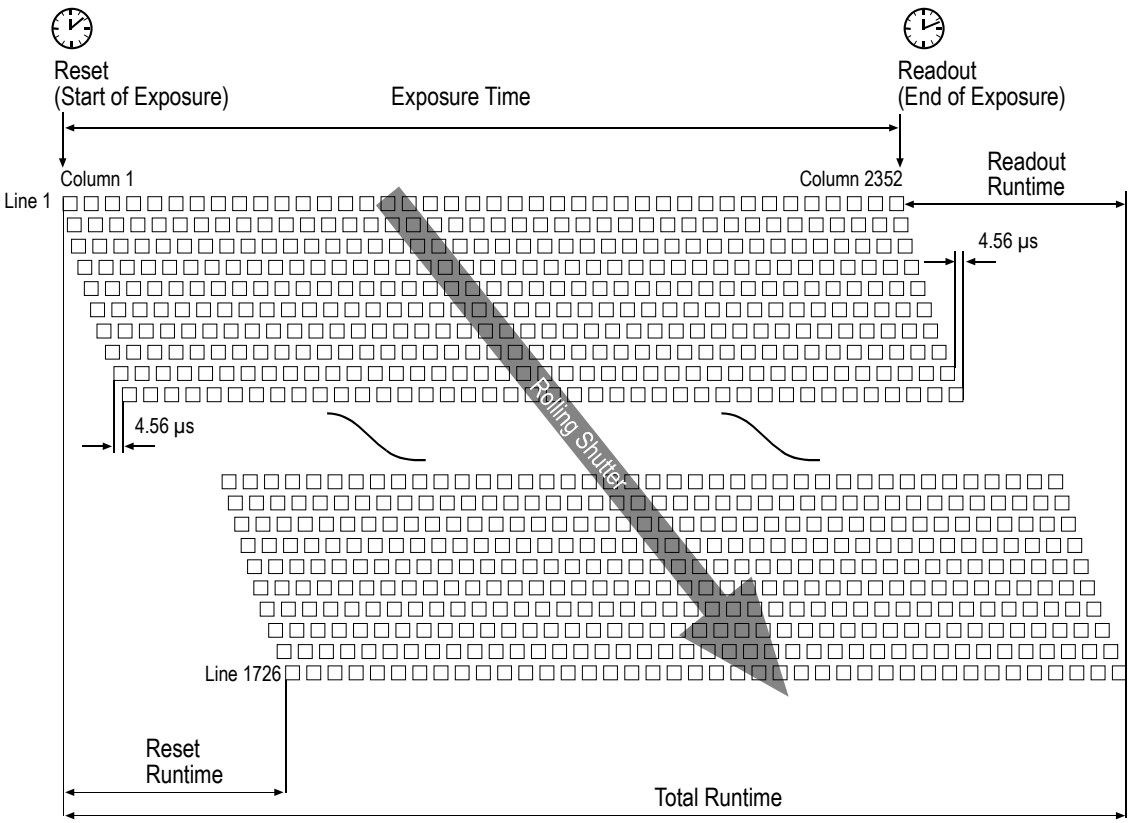


Figure 3-8: Rolling Shutter

3.4.1 Guidelines for Successful Use of the Rolling Shutter

To successfully use the rolling shutter functionality of the camera, make sure that you observe the guidelines listed below.

- Exposure time can only be set incrementally in multiples of 4.56 µs. Formula:

$$\text{Exposure time} = N \times 4.56\ \mu s$$

  where N must be an integer and > 0. Exposure time can range from 4.56 µs to 19.12 s.

- The runtime of the sensor reset and readout depends on the height of the area of interest (AOI, see page 3-24). The formula below can be used to calculate the runtime:

  Reset runtime = Readout runtime = 4.56 µs x (AOI Height - 1)

- The formula below can be used to calculate the total time it takes to reset, expose and read out a single frame:

$$\text{Total frame exposure runtime} = 4.56\ \mu s \times (\text{AOI Height} - 1) + \text{Exposure time}$$

  Examples:

  (A) The height of the area of interest (AOI) is 1726 lines (full resolution):

  Total runtime = 4.56 µs x (1726 - 1) + Exposure time = 7866 µs + Exposure time

  (B) The height of the area of interest (AOI) is only 200 lines:

  Total runtime = 4.56 µs x (200 - 1) + Exposure time = 907.4 µs + Exposure time

- Imaging of fast moving objects requires a flash exposure within the camera's flash window (see Section 3.4.2).

3.4.2 Flash Exposure for Fast Moving Objects

Imaging of fast moving objects requires a flash exposure. If flash exposure is not used, image distortions will occur due to the exposure's 4.56 µs offset from one line to the next.

Due to the exposure's 4.56 µs offset from one line to the next, there is a limited time interval where all pixel lines are open, that is, all pixels are exposed to light simultaneously. This time interval is called the "flash window" of the camera (see Figure 3-9).

The flash window opens when exposure is started in the last pixel line within the area of interest (AOI) and it closes when readout is started in the first pixel line within the AOI. The width of the flash window is calculated using the below formula:

Flash window width [µs] = Exposure time [µs] - (AOI Height x 4.56 µs)

A400k cameras output a flash trigger signal that can be used to trigger flash exposure. The flash trigger signal can be programmed to be high as long as the flash window is open, that is, all pixel lines are exposed to light and the flash should occur (see Section 2.5.8 on page 2-32).

To effectively use the flash exposure, the guidelines below must be observed:

- The flash must occur while the flash window is open, that is, the flash trigger signal is high.
- The exposure time setting on the camera and the duration of the flash must be equal to or higher than the minimum flash exposure time required. The minimum flash exposure time required is calculated using the below formula:

  $$\text{Minimum flash exposure} = 4.56\ \mu s \times (\text{AOI Height} - 1) + T_{Flash}$$

  where T_{Flash} is the time it takes to trigger the flash

  Examples:

  (A) The height of the area of interest (AOI) is 1726 lines (full resolution):

  $$\text{Minimum flash exposure} = 4.56\ \mu s \times (1726 - 1) + \text{Exposure time} = \underline{7866\ \mu s + T_{Flash}}$$

  (B) The height of the area of interest (AOI) is only 200 lines:

  $$\text{Minimum flash exposure} = 4.56\ \mu s \times (200 - 1) + \text{Exposure time} = \underline{907.4\ \mu s + T_{Flash}}$$

	If the exposure time setting on the camera is lower than the minimum flash exposure time required, no flash trigger signal will be output.

- The light intensity of the flash must be considerably higher than the light intensity in the scene when no flash is present.
- Exposure of the next frame can be started while the previous frame is still being read out.

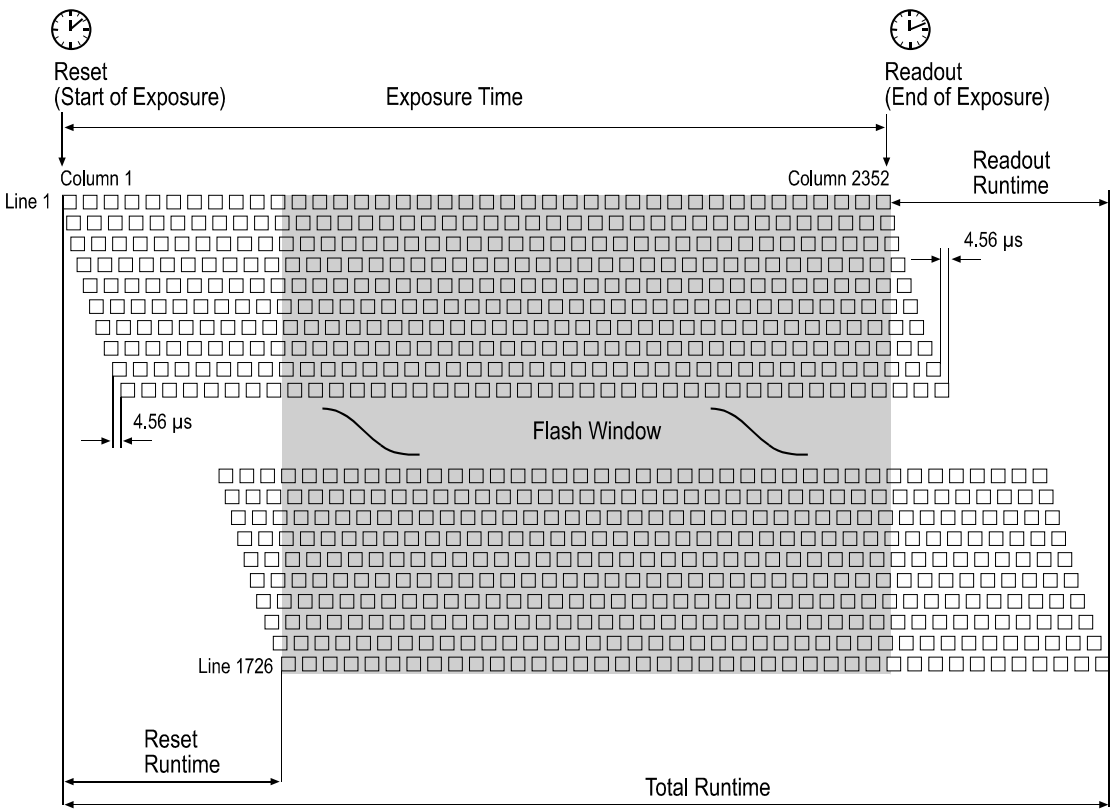


Figure 3-9: Flash Window

3.5 Gain and Offset

3.5.1 Gain

Gain on A400k cameras is adjustable within a range from 0% to 100% where 0 % correspond to the minimum gain and 100 % correspond to the maximum gain.

The minimum gain is always 0 dB. At an offset of 0 %, the maximum gain is approximately 13 dB. The maximum achievable gain decreases as the offset is increased. See Table 3-1.

The default gain is 0%.

Offset	Gain in dB @ 100% Gain
0%	~ 13 dB
100%	~ 8 dB

Table 3-1: Max. Gain

As shown in the graphs in Figure 3-10, increasing the gain setting increases the slope of the camera's response curve and results in a higher camera output for a given amount of light. Decreasing the gain setting decreases the slope of the response curve and results in a lower camera output for a given amount of light.

Increasing gain also increases noise. The signal to noise ratio decreases as gain is increased.

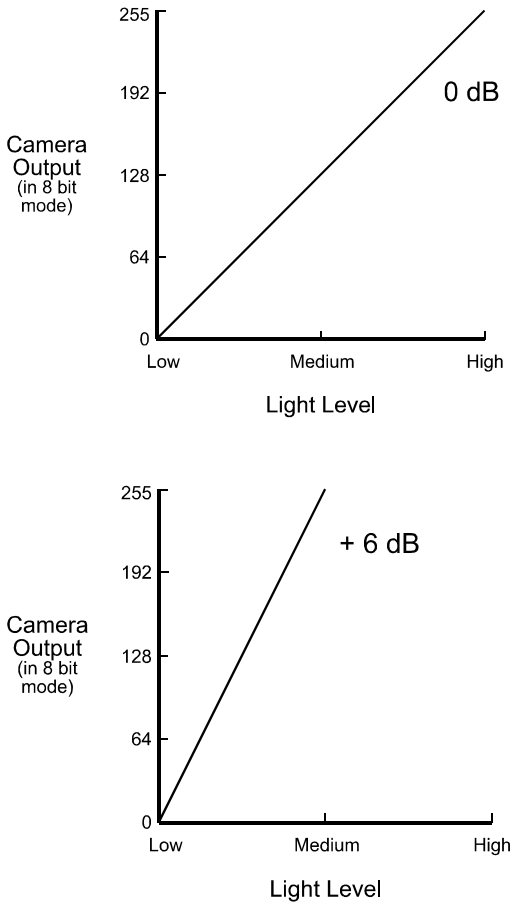


Figure 3-10: Response at Various Gain Settings

3.5.1.1 Setting the Gain

You can set the gain by using the Camera Configuration Tool Plus (CCT+) or by using binary write commands from within your own application to set the camera's control and status registers (CSRs).

With the CCT+

With the CCT+ (see Section 4.1), you use the Gain setting in the Gain & Offset parameter group to set the gain.

By Setting CSRs

You can set the gain by writing a value to the Raw Gain field or to the Absolute Gain field of the Gain CSR (see page 4-20).

Section 4.2.2 explains CSRs and the difference between using the "raw" field and the "absolute" field in a CSR. Section 4.3.1 explains using read/write commands.

3.5.2 Offset

Offset on A400k cameras is adjustable within a range from 0% to 100% where 0% correspond to an offset of 0 gray values and 100% correspond to an offset of approximately 32 gray values (8 bit output mode) or 128 gray values (10 bit output mode).

Increasing the offset by 3% will result in an increase of approximately one gray value (8 bit output mode) or four gray values (10 bit output mode) in the average pixel value for each frame transmitted by the camera. Decreasing the offset by 3% will result in a decrease of approximately one gray value in the average pixel value for each frame (8 bit output mode).

The default offset is 2 gray values (8 bit output mode) or 8 gray values (10 bit output mode).

3.5.2.1 Setting the Offset

You can set the offset by using the Camera Configuration Tool Plus (CCT+) or by using binary write commands from within your own application to set the camera's control and status registers (CSRs).

With the CCT+

With the CCT+ (see Section 4.1), you use the Offset setting in the Gain & Offset parameter group to set the offset.

By Setting CSRs

You can set the offset by writing a value to the Raw Offset field or to the Absolute Offset field of the Offset CSR (see page 4-22).

Section 4.2.2 explains CSRs and the difference between using the "raw" field and the "absolute" field in a CSR. Section 4.3.1 explains using read/write commands.

3.6 Shading Correction

In theory, when a digital camera captures an image of a uniform object, the pixel values output from the camera should be uniform. In practice, however, variations in optics and lighting and small variations in the sensor's performance can cause the camera output to be non-uniform even when the camera is capturing images of a uniform object. A400k cameras are equipped with a shading correction feature that lets the camera correct the captured image for variations caused by optics, lighting, and sensor variations. There are three types of shading correction available on A400k cameras, column FPN shading correction, DSNU shading correction, and PRNU shading correction.

3.6.1 Column FPN Shading Correction

In theory, when an area scan camera with a digital sensor captures an image of a uniform object under homogeneous illumination, the pixels should output the same gray value throughout the entire image. In practice, slight variations in the pixel column amplifiers in the sensor will cause some variation from pixel column to pixel column. This variation is known as column Fixed Pattern Noise (column FPN). Column FPN appears as vertical stripes in the image. The camera's sensor contains special self-calibrating circuitry that enables it to reduce column FPN before the analog pixel data enters the analog-to-digital converters. The column FPN shading correction feature on A400k cameras can further correct for the variations caused by column FPN.

A column FPN shading correction overwrites the column FPN shading correction that is done by the sensor's self-calibrating circuitry.

Generating a Set of Column FPN Shading Correction Values

Before you can use column FPN shading correction, you must generate a column FPN shading correction table. To create the table, perform the following steps:

1. As column FPN varies depending on the temperature, make sure that the camera has reached its operating temperature.
2. Cover the camera lens, close the iris in the camera lens, or darken the room so that the camera will be capturing frames in complete darkness.
3. Signal the camera to generate a set of column FPN shading correction values:
   a) You can start the generation of a set of column FPN shading correction values by using the Camera Configuration Tool Plus (see Section 4.1). With the CCT+, you use the Shading Value Generate parameter in the Column FPN Shading Correction parameters group to start the generation of a set of column FPN shading correction values.
   b) You can also start the generation of the column FPN shading correction table by using a binary write command (see Section 4.3) to write a value to the Generate field of the Column FPN Shading Value Generate CSR (see page 4-23).

   After you signalled the start of column FPN shading correction value generation, generation is a fully-automated process and requires no ExSync signal. When column FPN shading correction value generation is started, the camera stops image capture and data output. During generation (~ 4 seconds), the camera loads a special set of parameters; no image is captured and no data is output from the camera. The camera calculates the column FPN shading correction values and creates the table of correction values.

   When column FPN shading correction value generation is complete, the set of column FPN values will be placed in the camera's volatile memory. This set of values will overwrite any shading values that are already in the memory. After column FPN shading correction value generation is complete, the camera reloads the original set of parameters and continues to capture images and output data.

Enabling Column FPN Shading Correction

A column FPN shading correction overwrites the column FPN shading correction that is done by the sensor's self-calibrating circuitry. Once you have a set of column FPN shading correction values in place, the camera automatically starts to use the generated column FPN shading correction table to apply the appropriate offset to each pixel to correct for column FPN. Use of column FPN shading correction can not be disabled.

Resetting Column FPN Shading Correction

A column FPN shading correction overwrites the column FPN shading correction that is done by the sensor's self-calibrating circuitry. To reproduce the FPN shading correction done by the sensor's self-calibrating circuitry, you can reset the column FPN shading correction to the original values with the Camera Configuration Tool Plus or by using binary read/write commands from within your own application to set the camera's control and status registers (CSRs).

With the CCT+

With the Camera Configuration Tool Plus (see Section 4.1), you use the Shading Value Generate parameter in the Column FPN Shading Correction parameters group to reset column FPN shading correction.

By Setting CSRs

You can start a reset by writing a value to the Generate field of the Column FPN Shading Correction CSR (see page 4-23).

Section 4.2.2 explains CSRs. Section 4.3.1 explains using read/write commands.

Saving a Set of Column FPN Shading Values to a File

When you generate a set of column FPN shading correction values, the values are placed in the camera's volatile memory and they overwrite any shading values that are already in the memory. The current set of values in the volatile memory is used immediately by the camera. Values placed in the camera's volatile memory are lost if the camera is reset or the camera power is switched off.

A400k cameras can save the current column FPN values in the volatile memory to a file in the camera's non-volatile memory. Files saved in the non-volatile memory are not lost at reset or power off. You can save only one set of column FPN values to file in the non-volatile memory.

A save will take four to five minutes.

You can save the current shading values to a file in the non-volatile memory by using the Camera Configuration Tool Plus (CCT+) or by using binary read/write commands from within your own application to set the camera's control and status registers (CSRs).

With the CCT+

With the CCT+ (see Section 4.1), you use the Create Column FPN Shading Values File parameter in the Column FPN Shading Value File parameters group to save the column FPN shading set currently in the volatile memory to a file in the non-volatile memory.

By Setting CSRs

You can save the current shading correction values to a file in the non-volatile memory by writing values to the bulk data CSR for column FPN shading values.

Section 4.2.3 explains the bulk data CSRs and explains how to use the CSRs to save the shading values to a file. Section 4.3.1 explains using read/write commands.

3.6.2 DSNU Shading Correction

In theory, when an area scan camera with a digital sensor captures a frame in complete darkness, all of the pixel values in the frame should be near zero and they should be equal. In practice, slight variations in the performance of the pixels in the sensor will cause some variation the pixel values output from the camera when the camera is capturing frames in darkness. This variation is known as Dark Signal Non-uniformity (DSNU). The DSNU shading correction feature on A400k cameras can correct for the variations caused by DSNU.

Generating a Set of DSNU Shading Correction Values

Before you can use DSNU shading correction, you must generate a set of DSNU shading correction values. To generate a set of values, perform the following steps:

1. For optimum performance, make sure that a set of column FPN shading correction values has been created before. Doing DSNU shading correction before column FPN shading correction can result in image quality degradation.
2. As DSNU varies depending on the temperature, make sure that the camera has reached its operating temperature.
3. Make sure that the area of interest is set to the area where you want to generate values.
4. Cover the camera lens, close the iris in the camera lens, or darken the room so that the camera will be capturing frames in complete darkness.
5. Set the gain as you would for normal system operation.
6. Make sure that the offset is set so all gray values including the noise are around 16 (8-bit mode) or 64 (10-bit mode) or lower.
7. Signal the camera to generate a set of DSNU shading values:
   c) You can start the generation of a set of DSNU shading values by using the Camera Configuration Tool Plus (see Section 4.1). With the CCT+, you set the Shading Value Generate parameter in the DSNU & PRNU Shading Correction parameters group to start the generation of a set of DSNU shading values.
   d) You can also start the generation of the DSNU shading table by using a binary write command (see Section 4.3) to write a value to the Generate field of the DSNU or PRNU Shading Value Generate CSR (see page 4-24).
8. The camera must capture at least eight frames to create a set of DSNU shading correction values. If your camera is set to control exposure with an ExSync signal, you must generate at least eight ExSync signal cycles after you signal the camera to begin generating the values. If you are running the camera in a free-run exposure mode, you must wait long enough for the camera to capture at least eight frames.
9. Once eight frames have been captured, the camera calculates the DSNU shading correction values:
   a) The camera uses the data from the eight captured frames to calculate an average gray value for each pixel in the frame.
   b) The camera finds the pixel with the highest average gray value in the frame.
   c) For each of the other pixels in the frame, the camera determines the offset that would be needed to make the pixel's average value equal to the average value for the highest pixel.
   d) The camera creates a set of DSNU shading values that contains the calculated offsets.

   The set of DSNU values will be placed in the camera's volatile memory. This set of values will overwrite any shading values that are already in the memory. The current set of values in the volatile memory is used whenever DSNU is enabled.

Enabling DSNU Shading Correction

Once you have a DSNU shading table in place you can enable and use DSNU shading correction. With the DSNU correction feature enabled, the camera will use the set of shading values to apply the appropriate offset to each pixel to correct for DSNU.

You can enable DSNU shading correction with the Camera Configuration Tool Plus or by using binary read/write commands from within your own application to set the camera's control and status registers (CSRs).

With the CCT+

With the Camera Configuration Tool Plus (see Section 4.1), you set the Shading Mode parameter in the Shading Correction parameters group to enable DSNU shading correction.

By Setting CSRs

You can enable DSNU shading correction by writing a value to the Mode field of the DNSU and / or PRNU Shading Correction Enable CSR (see page 4-24).

Section 4.2.2 explains CSRs. Section 4.3.1 explains using read/write commands.

Saving a Set of DSNU Shading Values to a File

When you generate a set of DSNU shading correction values, the values are placed in the camera's volatile memory and they overwrite any shading values that are already in the memory. The current set of values in the volatile memory is used immediately by the camera. Values placed in the camera's volatile memory are lost if the camera is reset or the camera power is switched off.

A400k cameras can save the current DSNU values in the volatile memory to a file in the camera's non-volatile memory. Files saved in the non-volatile memory are not lost at reset or power off. You can save only one set of DSNU values to file in the non-volatile memory. A save will take approximately two minutes.

You can save the current shading values to a file in the non-volatile memory by using the Camera Configuration Tool Plus (CCT+) or by using binary read/write commands from within your own application to set the camera's control and status registers (CSRs).

With the CCT+

With the CCT+ (see Section 4.1), you use the Create DSNU Shading Values File parameter in the DSNU Shading Value File parameters group to save the DSNU shading set currently in the volatile memory to a file in the non-volatile memory.

By Setting CSRs

You can save the current shading correction values to a file in the non-volatile memory by writing values to the bulk data CSR for DSNU shading values.

Section 4.2.3 explains the bulk data CSRs and explains how to use the CSRs to save the shading values to a file. Section 4.3.1 explains using read/write commands.

If you save the set of DSNU values to file, enable the use of shading correction and save the current work set to your standard user set, the camera will automatically load and use shading correction at next power on. Loading will take approximately 90 seconds. After 90 seconds, the camera will start image capture and be able to receive commands.

3.6.3 PRNU Shading Correction

In theory, when an area scan camera with a digital sensor captures a frame with the camera viewing a uniform white target in bright light, all of the pixel values in the frame should be near their maximum gray value and they should be equal. In practice, slight variations in the performance of the pixels in the sensor, variations in the optics, and variations in the lighting will cause some variation the pixel values output from the camera. This variation is know as Photo Response Non-uniformity (PRNU). The PRNU shading correction feature on A400k cameras can correct for the variations caused by PRNU. In the color version, PRNU shading correction is executed for each color separately.

Shading correction values will only be generated for the pixels inside of the current area of interest. No changes will be made to the pixels outside of the area of interest.

The camera can have only one set of shading correction values but you can have special shading correction values for each area of interest in the same set if the areas of interest do not overlap. Creating different shading correction values for each area of interest will be necessary if you have two or more areas of interest to be captured after the other under different illumination. For example, in order to create special shading correction values for two areas of interest within the same set, you would set the first area of interest and create correction values under the illumination for the first area of interest so the values go into the set and then, you would set the second area of interest and create correction values under the illumination for the second area of interest so these values also go into the set. The set would then contain shading correction values for the two areas of interest.

Generating a Set of PRNU Shading Values

Before you can use PRNU shading correction, you must generate a set of PRNU shading correction values. If you have two or more areas of interest to be captured under different illumination, repeat the below procedure for each area of interest. Make sure that the areas do not overlap. To generate a set of values, perform the following steps:

1. Make sure that a set of column FPN shading correction values and a set of DSNU shading correction values has been created before. Doing PRNU shading correction before column FPN shading correction or before DSNU shading correction can result in significant image quality degradation.
2. Make sure that the area of interest is set to the area where you want to generate values.
3. Place a uniform white or light colored target in the field of view of the camera. Adjust your lighting and optics as you would for normal system operation.
4. Set the gain on the camera to default.
5. Make sure that no part of the area of interest has reached saturation, that is, all gray values are lower than 255 (8-bit) or 1023 (10-bit).
6. Capture several frames and examine the pixel values returned from the camera. The pixel values should be about 80% of maximum.
   a) If the pixel values are not at 80% of maximum adjust your lighting and/or lens aperture setting to achieve 80%.
   b) If you can not achieve 80% output by adjusting the lighting, then adjust the gain setting to achieve the correct output.
7. Capture several frames and examine the pixel values returned from the camera. In each frame, the values for the darkest pixels must not be less 1/2 of the values for the lightest pixels in the line. (If the values for the darkest pixels are less than 1/2 of the value for the lightest pixels, the camera will not be able to fully correct for shading variations.)
   a) If the values for the darkest pixels are not less than 1/2 of the value for the lightest pixels, go on to step 8.

b) If the values for the darkest pixels are less than 1/2 of the value for the lightest pixels, it usually indicates extreme variations in lighting or poor quality optics. Make corrections as required.

8. Signal the camera to generate a set of PRNU shading values:
   a) You can start the generation of a set of PRNU shading values by using the Camera Configuration Tool Plus (see Section 4.1). With the CCT+, you set the Shading Value Generate parameter in the DSNU & PRNU Shading Correction parameters group to start the generation of a set of PRNU shading values.
   b) You can also start the generation of the PRNU shading table by using a binary write command (see Section 4.3) to write a value to the Generate field of the DSNU or PRNU Shading Value Generate CSR (see page 4-24).
9. The camera must capture at least eight frames to generate a set of PRNU shading correction values. If your camera is set to control exposure with an ExSync signal, you must generate at least eight ExSync signal cycles after you signal the camera to begin generating the values. If you are running the camera in a free-run exposure mode, you must wait long enough for the camera to capture at least eight frames.
10. Once eight frames have been captured, the camera calculates the PRNU shading correction values:
    a) The camera uses the data from the eight captured frames to calculate an average gray value for each pixel in the frame.
    b) The camera finds the pixel with the highest average gray value in the frame.
    c) For each of the other pixels in the frame, the camera determines the additional gain that would be needed to make the pixel's average value equal to the average value for the highest pixel.
    d) The camera creates a set of PRNU shading correction values that contains the calculated gain adjustments.

    The set of PRNU values will be placed in the camera's volatile memory. This set of values will overwrite any shading values that are already in the memory. The current set of values in the volatile memory is used whenever PRNU is enabled.

Enabling PRNU Shading Correction

Once you have a PRNU shading table in place you can enable and use PRNU shading correction. With the PRNU correction feature enabled, the camera will use the set of shading values to apply the appropriate offset to each pixel to correct for PRNU.

You can enable PRNU shading correction with the Camera Configuration Tool Plus or by using binary read/write commands from within your own application to set the camera's control and status registers (CSRs).

With the CCT+

With the Camera Configuration Tool Plus (see Section 4.1), you set the Shading Mode parameter in the Shading Correction parameters group to enable PRNU shading correction.

By Setting CSRs

You can enable PRNU shading correction by writing a value to the Mode field of the DSNU and/or PRNU Shading Correction Enable CSR (see page 4-24).

Section 4.2.2 explains CSRs. Section 4.3.1 explains using read/write commands.

Saving a Set of Shading Values to a File

When you generate a set of PRNU shading correction values, the values are placed in the camera's volatile memory and they overwrite any shading values that are already in the memory. The current set of values in the volatile memory is used immediately by the camera. Values placed in the camera's volatile memory are lost if the camera is reset or the camera power is switched off.

A400k cameras can save the current PRNU values in the volatile memory to a file in the camera's non-volatile memory. Files saved in the non-volatile memory are not lost at reset or power off. You can save only one set of PRNU values to file in the non-volatile memory.

A save will take approximately two minutes.

You can save the current shading values to a file in the non-volatile memory by using the Camera Configuration Tool Plus (CCT+) or by using binary read/write commands from within your own application to set the camera's control and status registers (CSRs).

With the CCT+

With the CCT+ (see Section 4.1), you use the Create PRNU Shading Values File parameter in the PRNU Shading Value File parameters group to save the PRNU shading set currently in the volatile memory to a file in the non-volatile memory.

By Setting CSRs

You can save the current shading correction values to a file in the non-volatile memory by writing values to the bulk data CSR for PRNU shading values.

Section 4.2.3 explains the bulk data CSRs and explains how to use the CSRs to save the shading values to a file. Section 4.3.1 explains using read/write commands.

If you save the set of PRNU values to file, enable the use of shading correction and save the current work set to your standard user set, the camera will automatically load and use shading correction at next power on. Loading will take approximately 90 seconds. After 90 seconds, the camera will start image capture and be able to receive commands.

3.6.4 Guidelines When Using Shading Correction

When using the shading correction feature, make sure to take the following guidelines into account:

- Any time that you make a change to the optics or lighting or if you change the camera's gain setting, you must generate new set of PRNU shading values. Using an out of date PRNU shading set can result in poor image quality.
- When you generate the DSNU and PRNU shading tables, correction values will be calculated for the pixels in the current area of interest only. If you change the AOI settings, you need to generate new shading values.

3.7 Digital Shift

The “digital shift” feature lets you change the group of bits that is output from the ADC. Using the digital shift feature will effectively multiply the output of the CMOS sensor by 2 times or 4 times.

Section 3.7.1 describes how digital shift works when the camera is operating in 10 bit output mode, and Section 3.7.2 describes how digital shift works when the camera is operating in 8 bit output mode.

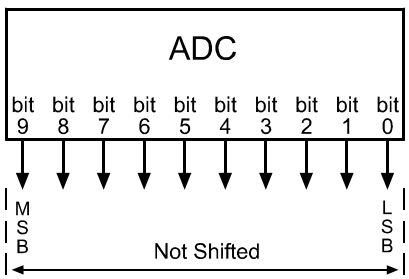
Before you use digital shift, also observe the precautions described in Section 3.7.3.

3.7.1 Digital Shift in 10 Bit Output Mode

No Shift

As mentioned in Section 3.1, the A400k uses 10 bit ADCs to digitize the output from the CMOS sensor.

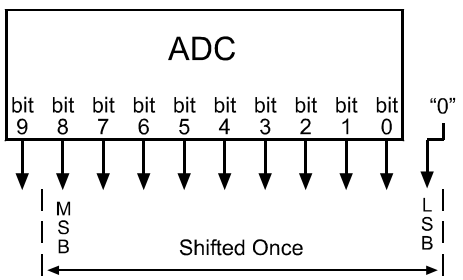
When the camera is operating in 10 bit output mode, by default, the camera transmits the 10 bits that are output from each ADC.



Shift Once

When the camera is set to shift once, the output from the camera will include bit 8 through bit 0 from each ADC along with a zero as an LSB.

The result of shifting once is that the output of the camera is effectively doubled. For example, assume that the camera is set for no shift, that it is viewing a uniform white target, and that under these conditions the reading for the brightest pixel is 100. If you changed the digital shift setting to shift once, the reading would increase to 200.



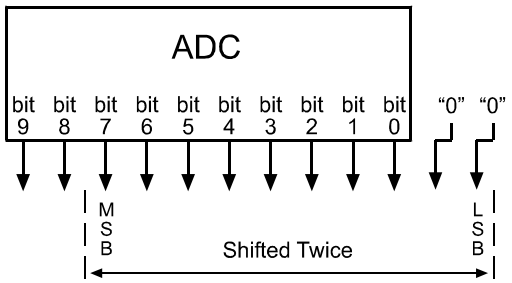
If bit 9 is set to 1, all of the other bits will automatically be set to 1. This means that you should only use the shift once setting when your pixel readings with no digital shift are all below 512.

Since the shift once setting requires that the least significant bit (LSB) always be “0“, no odd gray values can be output. The gray value scale will only include gray values of 2, 4, 6 and so forth. The absence of some gray values is commonly called “Missing Codes“.

Shift Twice

When the camera is set to shift twice, the output from the camera will include bit 7 through bit 0 from each ADC along with two zeros as LSBs.

The result of shifting twice is that the output of the camera is effectively multiplied by four. For example, assume that the camera is set for no shift, that it is viewing a uniform white target, and that under these conditions the reading for the brightest pixel is 100. If you changed the digital shift setting to shift twice, the reading would increase to 400.



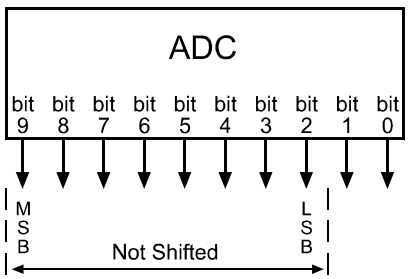
If bit 9 or bit 8 is set to 1, all of the other bits will automatically be set to 1. This means that you should only use the shift twice setting when your pixel readings with no digital shift are all below 256.

Since the shift twice setting requires that the two least significant bits always be "0", the gray value scale will only include every 4th gray value. For example, 4, 8, 16 and so forth.

3.7.2 Digital Shift in 8 Bit Output Mode

No Shift

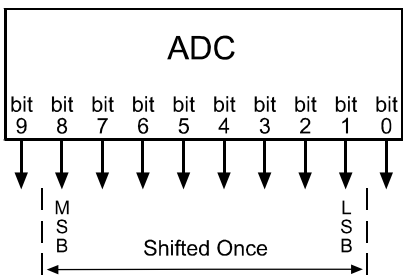
As mentioned in Section 3.1, the A400k uses 10 bit ADCs to digitize the output from the CMOS sensor. When the camera is operating in 8 bit output mode, by default, it drops the least two significant bits from the ADC and transmits the 8 most significant bits (bit 9 through bit 2).



Shift Once

When the camera is set to shift once, the output from the camera will include bit 8 through bit 1 from the ADC.

The result of shifting once is that the output of the camera is effectively doubled. For example, assume that the camera is set for no shift, that it is viewing a uniform white target and that under these conditions the reading for the brightest pixel is 20. If we changed the digital shift setting to shift once, the reading would increase to 40.

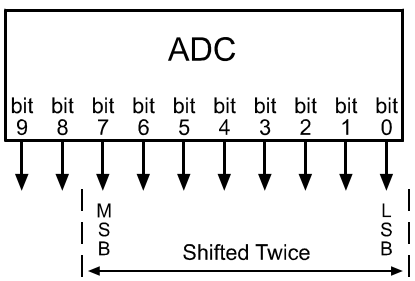


If bit 9 is set to 1, all of the other bits will automatically be set to 1. This means that you should only use the shift once setting when your pixel readings with no digital shift are all below 128.

Shift Twice

When the camera is set to shift twice, the output from the camera will include bit 7 through bit 0 from the ADC.

The result of shifting twice is that the output of the camera is effectively multiplied by four. For example, assume that the camera is set for no shift, that it is viewing a uniform white target, and that under these conditions the reading for the brightest pixel is 20. If we changed the digital shift setting to shift twice, the reading would increase to 80.



If bit 9 or bit 8 is set to 1, all of the other bits will automatically be set to 1. This means that you should only use the shift twice setting when your pixel readings with no digital shift are all below 64.

3.7.3 Precautions When Using Digital Shift

There are several checks and precautions that you must follow before using the digital shift feature. The checks and precautions differ depending on whether you will be using the camera in 10 bit output mode or in 8 bit output mode.

If you will be using the camera in 10 bit output mode, make this check:

1. Use binary commands or the Camera Configuration Tool Plus to put the camera in 10 bit output mode.
2. Use binary commands or the configuration tool to set the camera for no digital shift.
3. Check the output of the camera under your normal lighting conditions with no digital shift and note the readings for the brightest pixels.
   - If any of the readings are above 512, do not use digital shift.
   - If all of the readings are below 512, you can safely use the 2X digital shift setting.
   - If all of the readings are below 256, you can safely use the 2X or 4X digital shift setting.

If you will be using the camera in 8 bit output mode, make this check:

1. Use binary commands or the Camera Configuration Tool Plus to put the camera in 8 bit output mode.
2. Use the binary commands or the configuration tool to set the camera for no digital shift.
3. Check the output of the camera under your normal lighting conditions with no digital shift and note the readings for the brightest pixels.
   - If any of the readings are above 128, do not use digital shift.
   - If all of the readings are below 128, you can safely use the 2X digital shift setting.
   - If all of the readings are below 64, you can safely use the 2X or 4X digital shift setting.

3.7.4 Enabling/Disabling Digital Shift

You can enable or disable the digital shift feature by using the Camera Configuration Tool Plus (CCT+) or by using binary write commands from within your own application to set the camera's control and status registers (CSRs).

With the CCT+

With the CCT+ (see Section 4.1), you use the Digital Shift setting in the Output parameter group to enable/disable digital shift.

By Setting CSRs

You can enable/disable digital shift by writing a value to the Mode field of the Digital Shift CSR (see page 4-25).

See Section 4.2.2 for an explanation of CSRs. See Section 4.3.1 for an explanation of using read/write commands.

3.8 Area of Interest (AOI)

The area of interest feature lets you specify a portion of the CMOS array and during operation, only the pixel information from the specified portion is transferred out of the camera.

The size of the area of interest is defined by declaring a starting column, a width in columns, a starting line and a height in lines. Starting columns can only be selected in multiples of 16 (+1), that is, the starting column can be 1, 17, 33, and so on. The width can only be multiples of 16, that is, 16, 32, 48, and so on.

Reference position is the top left corner of the image. For example, suppose that you specify the starting column as 17, the width in columns as 16, the starting line as 8 and the height in lines as 10. As shown in Figure 3-11, the camera will only transmit pixel data from within the defined area.

Information from the pixels outside of the area of interest is discarded.

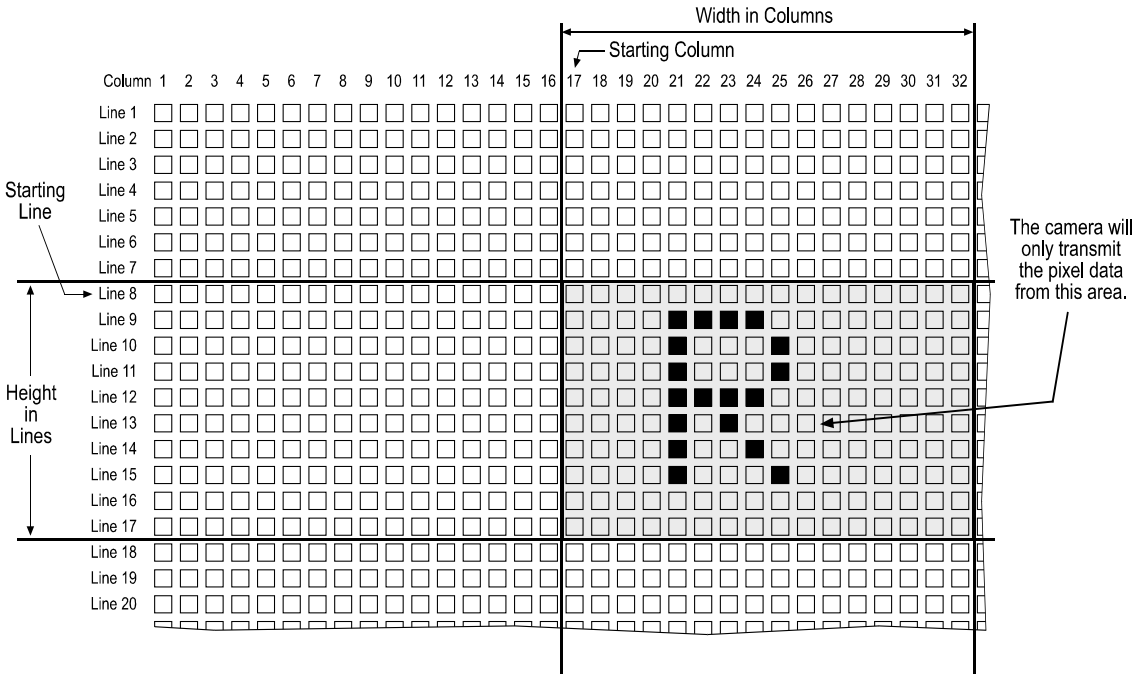


Figure 3-11: Area of Interest

In normal operation, the camera is set to use all of the pixels in the array. To use all of the pixels, the starting column should be set to 1, the width in columns to 2352, the starting line to 1 and the height in lines to 1726.

3.8.1 Area of Interest Setup Rules

When setting up the area of interest, observe the following rules:

- Starting columns can only be selected in multiples of 16 (+1), that is, the starting column can be 1, 17, 33, and so on.
- The width can only be multiples of 16, that is, 16, 32, 48, and so on.
- The sum of the setting for the starting column plus the setting for the width in columns can not exceed 2353.
- The sum of the setting for the starting line plus the setting for the height in lines can not exceed 1727.

3.8.2 Setting the Area of Interest

You can set the area of interest by using the Camera Configuration Tool Plus (CCT+), by using binary write commands from within your own application to set the camera's control and status registers (CSRs) or by using the AOI Editor.

With the CCT+

With the CCT+ (see Section 4.1), you use the AOI Starting Column, AOI Width, AOI Starting Line, and AOI Height settings in the Area of Interest parameters group to set the area of interest.

By Setting CSRs

You can set the AOI starting column by writing a value to the Starting Column field of the AOI Starting Column CSR (see page 4-25).

You can set the AOI width by writing a value to the Width field of the AOI Width CSR (see page 4-27).

You can set the AOI starting line by writing a value to the Line field of the AOI Starting Line CSR (see page 4-28).

You can set the AOI height by writing a value to the Height field of the AOI Height CSR (see page 4-29).

See Section 4.2.2 for an explanation of CSRs. See Section 4.3.1 for an explanation of using read/write commands.

With the AOI Editor

You can set an AOI using the AOI Editor (see Section 3.8.4.2). This involves entering the settings of one AOI in the AOI list of the AOI Editor.

3.8.3 Changes to the Max Frame Rate with Area of Interest

When the area of interest feature is used, the camera's maximum achieveable frame rate increases. The amount that the maximum frame rate increases depends on the number of lines included in the area of interest (AOI Height) and the width of the area of interest (AOI Width). The fewer the number of lines in the area of interest and the smaller the width, the higher the maximum frame rate.

To determine the maximum frame rate for a given AOI, use your AOI settings to calculate a result in each of the two formulas below. These formulas take your AOI size into account. The formula that returns the lowest value will determine the maximum frame rate for the given AOI.

Formula 1:

A402k: Max. frames per second (approximated) $= \frac{50\ \text{MHz}}{\left(\frac{\text{AOI Width}}{2} + 8\right) \times (\text{AOI Height} + 1)}$

A403k: Max. frames per second (approximated) $= \frac{50\ \text{MHz}}{\left(\frac{\text{AOI Width}}{4} + 8\right) \times (\text{AOI Height} + 1)}$

A404k: Max. frames per second (approximated) $= \frac{50\ \text{MHz}}{\left(\frac{\text{AOI Width}}{8} + 7\right) \times (\text{AOI Height} + 1)}$

Formula 2:

$$\text{Maximum frames per second (approximated)} = \frac{1}{(\text{AOI Height} + 2) \times 4.56\ \mu\text{s}}$$

For example, using the full AOI height of 1726 lines, the frame rate cannot be higher than 126.9 fps (frames per second). With an AOI height of 200 lines, the frame rate cannot be higher than 1085.6 fps.

In some exposure modes, you must set the frame period in [µs]. To convert the calculated frame rate (frames per second) into the frame period [µs], use the following formula:

$$\text{Frame period [µs]} = \frac{1}{\text{Frame rate [s]}} \times 1\ 000\ 000$$

3.8.4 Programmable AOI Sequencer

The programmable area of interest sequencer feature lets the camera run a predefined sequence of two or more areas of interest. The sequence can be triggered by the ExSync signal or the camera's internal control signal (free-run).

Up to 32 areas of interest can be included in one sequence. Figure 3-12 illustrates a sequence that includes two areas of interest. The camera repeats the sequence as long as the AOI sequencer feature is enabled.

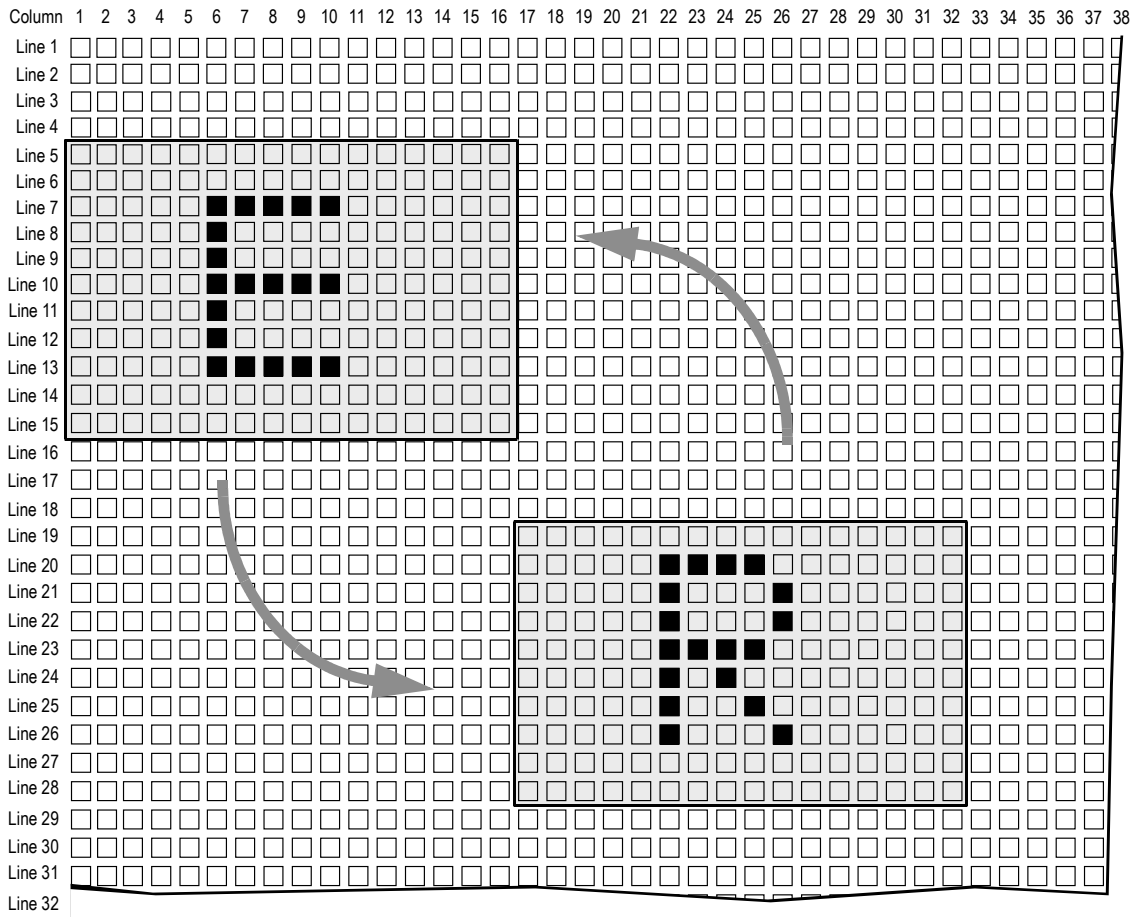


Figure 3-12: Area of Interest Sequence

As explained above, the sequencing can be triggered by the ExSync signal. You can trigger each image capture in the sequence with a separate rising edge of the ExSync signal or you can use a single rising edge of the signal to trigger the complete sequence. The camera can also run the complete sequence non-stop (free-run). In free-run, no ExSync signal is required.

Before you can run a predefined sequence of areas of interest, you must first create an AOI list (Sections 3.8.4.1 and 3.8.4.2). The AOI list defines the areas of interest, the order in which they will run and some other parameters. When the AOI list is complete, you upload the list to the camera (Section 3.8.4.3). To actually run the camera according to the defined sequence, you must finally enable the AOI sequencer feature by enabling one of three trigger modes (Section 3.8.4.4).

3.8.4.1 Setting Up an AOI List

The AOI list defines the areas of interest to be run. For each area of interest, you define an exposure time and delay time. You also define the number of times you want to run the area of interest within the sequence and whether you want the flash trigger signal to be enabled. The order in which the areas of interest are run is determined by their position in the list. The area of interest in the first position is performed first, the area of interest in second position is performed second, and so on. Up to 32 areas of interest can be defined. After the last area of interest in the list has been run, the sequence restarts with the first area of interest, and so on.

Figure 3-13 shows an AOI list that defines five areas of interest. In the AOI list, values of the starting columns must be entered as the value of the actual starting column minus one. The first area of interest's starting column is 1 (the entry is 1-1=0), the width is 1024 pixels, the starting line is 100 and the height in lines is 500. This area of interest will be captured using an exposure time of 600 * 4.56 µs. Exposure of the second area of interest will start 3000 * 4.56 µs after exposure of the first area of interest. The first area of interest will be run once, then the area of interest in second position will follow. For the first area of interest, the flash trigger signal will be enabled.

The second area of interest's starting column is 161 (the entry is 161-1=160), the width is 512 pixels, the starting line is 600 and the height in lines is 300. The second area of interest will be run three times with a delay of 2000 * 4.5 µs between each exposure. The third area of interest will be run 2000 * 4.5 µs after the last exposure for the second area of interest. For the second area of interest, the flash trigger signal will be disabled.

The areas of interest in third, fourth and fifth position are run once each. Then, the sequence is repeated starting with the area of interest in first position, and so on.

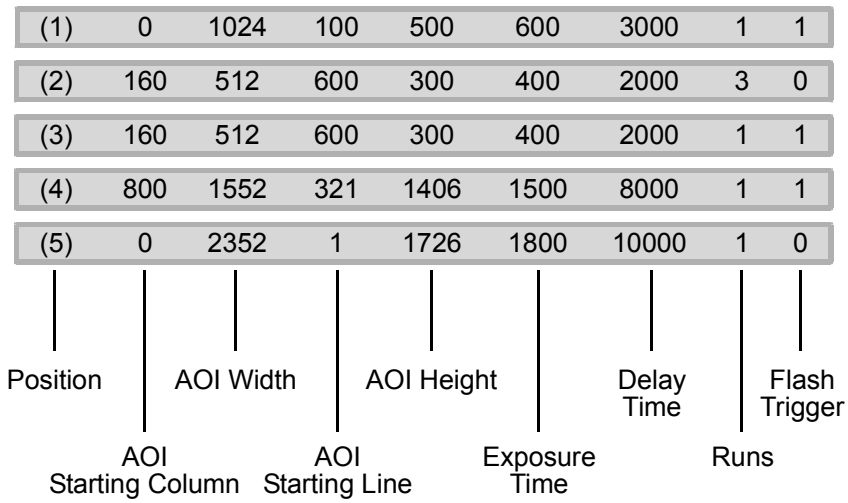


Figure 3-13: AOI List

When setting up the AOI list, a few guidelines must be observed:

- When the AOI sequencer feature is enabled, global area of interest, exposure time, frame period and parameter set cache parameter settings have no effect on the image. If global area of interest, exposure time, frame period and/or parameter set cache parameter settings are modified while the AOI sequencer feature is active, the modifications will be saved but will only become active after the AOI sequencer feature is disabled.
- The area of interest setup guidelines described in Section 3.8.1 must be observed.

- *Exposure time* and *delay time* settings represent multipliers and the actual exposure time is equal to the setting x 4.56 µs. The range of possible settings is 1 to 4,194,303 for the exposure time (4.56 µs to 19.12 s) and 2 to 4,194,303 for the delay time (9.12 µs to 19.12 s).
- 0 to 255 *runs* can be set. If the runs setting is 0, the area of interest will be skipped.
- The *flash trigger* setting can be 1 or 0 where 1 enables the flash trigger signal and 0 disables the flash trigger signal (see also Section 2.5.8).
- If the flash trigger setting is 1 and the flash window signal is output via the flash trigger signal (Sections 2.5.8 and 3.4.2), the *exposure time* setting in the AOI list must be equal to or higher than the sum of the height of the area of interest plus the width of the flash window:

  $$\text{Exposure Time Setting} \geq \text{AOI Height Setting} + \text{Flash Window Width}$$

  where the flash window width is calculated using the formula below:

  $$\text{Flash Window Width} = (\text{Exposure Time Setting} - \text{AOI Height Setting}) * 4.56\ \mu s$$
- If AOI trigger mode 2 or 3 is selected (see Section 3.8.4.4), the *delay time* setting must be equal to or higher than the AOI height setting to avoid overlapping exposures due to subsequent overlapping areas of interest:

  $$\text{Delay Time Setting} \geq \text{AOI Height Setting}$$
- The guidelines described in Section 3.3 must be observed to avoid overtriggering the camera.

3.8.4.2 Creating an AOI List

You can create an AOI list by

- setting up a list in hexadecimal format or by
- using the AOI Editor.

By Setting up a List in Hexadecimal Format

If you create the AOI list in the hexadecimal format, you must create a HEX file. To create a HEX file, you need a hexadecimal editor. If you do not have a hexadecimal editor, you can download a freeware editor from the web. For example, you can download DF Hex Editor from http://www.del-net.com/frmDFHEXEditorE.html.

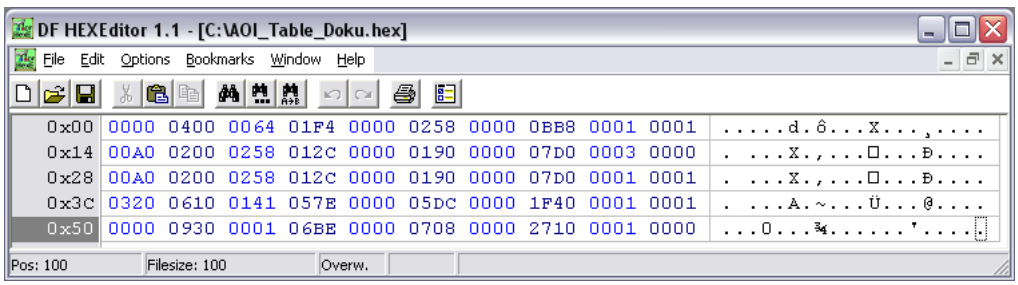


Figure 3-14: AOI List in Hexadecimal Editor

Once you have a hexadecimal editor available, perform the following steps:

1. Open the hexadecimal editor.
2. Enter the settings of the AOI list one after the other in hexadecimal numbers. For example, to enter the list shown in Figure 3-15, you would enter the hexadecimal numbers as shown in Figure 3-16.

(1)	0	1024	100	500	600	3000	1	1
(2)	160	512	600	300	400	2000	3	0
(3)	160	512	600	300	400	2000	1	1
(4)	800	1552	321	1406	1500	8000	1	1
(5)	0	2352	1	1726	1800	10000	1	0

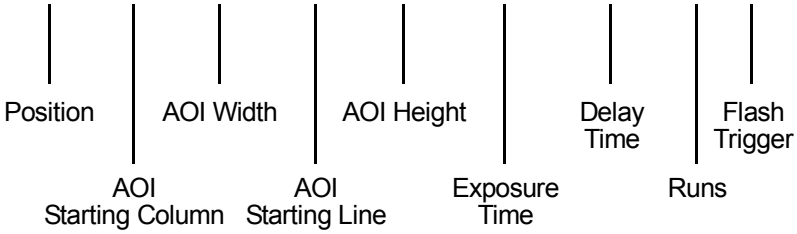


Figure 3-15: AOI List

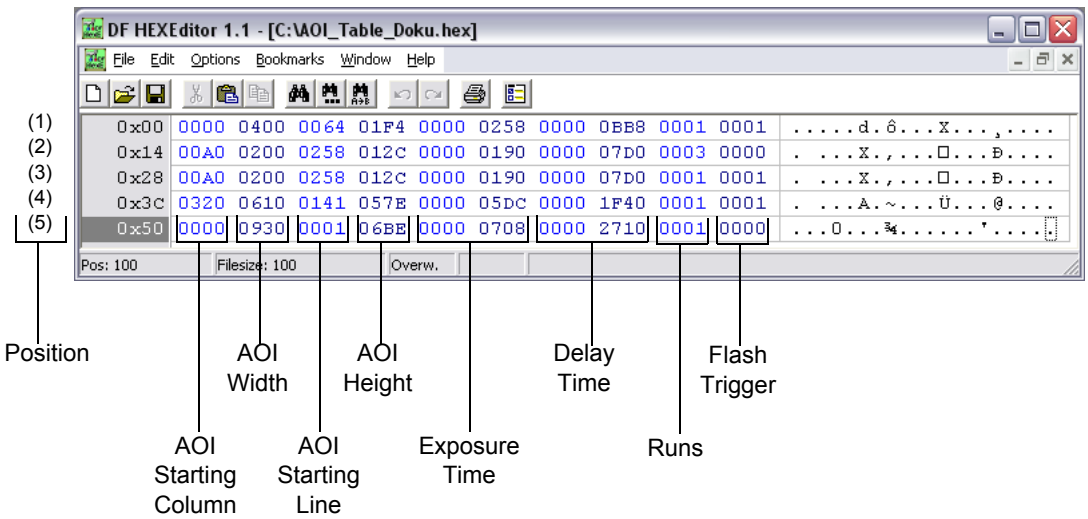


Figure 3-16: AOI List in Hexadecimal Editor

Note that AOI, runs and flash trigger must be 16 bit settings while exposure time and delay time must be 32 bit settings.

3. Save the file.
4. Proceed with Section 3.8.4.3.

With the AOI Editor

The AOI Editor is a convenient graphical interface for defining AOIs with the relevant parameters and for solving conflicting parameter settings. The AOI Editor can also be used for uploading an AOI list to the camera or for downloading an AOI list from the camera for editing. AOIs can be defined by

- decimal entries in an AOI list and
- in a graphical way by drag & drop. This can be done with reference to a full image taken by the camera.

You can downlad the AOI Editor and the pertinent User's Manual free of charge from www.basler-vc.com.

3.8.4.3 Uploading an AOI List to the Camera

Once you have an AOI list hex file in place, you can upload the hex file to the camera. With the hex file uploaded to the camera, the camera will use the settings in the file as soon as the AOI sequencer feature is enabled.

Uploading the hex file will also save the file in the camera's non-volatile memory. If an AOI list file already exists, it will be overwritten.

Uploading a HEX File to the Camera

You can upload the hex file to the camera by using the Camera Configuration Tool Plus CCT+ or by using binary read/write commands from within your own application to set the camera's bulk data control and status registers (CSRs).

With the CCT+

With the CCT+ (see Section 4.1), you use the Upload AOI List File setting in the AOI List File parameters group to upload the hex file to the camera.

By Setting CSRs

You can upload the hex file to the camera by writing values to the bulk data CSR for the Programmable AOI Sequencer feature.

Section 4.2.3 explains bulk data CSRs and using the bulk data activate process. Section 4.3.1 explains using read/write commands.

With the AOI Editor

You can upload an AOI list using the AOI Editor (see Section 3.8.4.2)

3.8.4.4 Enabling/Disabling the AOI List

Once you have uploaded an AOI list hex file to the camera, you can enable the sequencer. To enable the sequencer, that is, run the AOI list, AOI trigger mode 1, 2, or 3 must be set. In modes 1 and 2, the ExSync signal triggers image capture. Mode 3 activates free-run. To disable the feature, mode 0 must be selected (default).

Mode 0 = **Disabled**:
Disables the AOI list. Images are captured using the global area of interest, exposure time, frame period and parameter set cache parameter settings.

Mode 1 = **Image per Trigger**:
Each rising edge of the ExSync signal triggers an image capture. If this mode is applied to the example shown in Figure 3-15, on the first rising edge of the ExSync signal, the image will be captured according to the area of interest settings that are in first position in the AOI list. On the next three rising edges of the ExSync signal, three images will be captured according to the area of interest settings that are in second position in the AOI list since 3 runs have been defined, and so on.
In this mode, the delay time settings have no effect on the image capture, that is, there will be no delay between the rising edge of the ExSync signal and the start of exposure.

Mode 2 = **List per Trigger**:
Each rising edge of the ExSync signal triggers execution of the complete AOI list. If this mode is applied to the example shown in Figure 3-15 on page 3-30, on the first rising edge of the ExSync signal, seven images will be captured according to the area of interest settings in the AOI list, that is, the first image will be captured according to the area of interest settings in first position, the next three images will be captured according to the area of interest settings in second position, and so on. The seventh image will be captured according to the area of interest settings in fifth position and then, image capture will be stopped. On the rising edge of the next ExSync signal, the whole sequence will be done again, and so on.
In this mode, the delay time settings have an effect, that is, there will be the defined delay between the end of exposure of the previous image and the start of exposure of the next image.

Mode 3 = **Free-run**:
The AOI list is started, run and repeated non-stop. After the last position in the AOI list is done, the sequence restarts with the first position, and so on. In this mode, the delay time settings have an effect, that is, there will be the defined delay between the end of exposure of the previous image and the start of exposure of the next image.

You can set mode 1, 2, 3 or 4 by using the Camera Configuration Tool Plus (CCT+) or by using binary write commands from within your own application to set the camera's control and status registers (CSRs).

With the CCT+

With the CCT+ (see Section 4.1), you use the AOI List Trigger Mode setting in the AOI List parameter group to set disable the use of the AOI list or select the trigger mode.

By Setting CSRs

You can set mode 1, 2, 3 or 4 by writing a value to the Mode field of the Programmable AOI Sequencer CSR (see page 4-30).

See Section 4.2.2 for an explanation of CSRs. See Section 4.3.1 for an explanation of using read/write commands.

3.9 Stamp

The stamp feature provides the user with information about the area of interest settings of each captured image. When the stamp feature is enabled, the video data of the last 11 pixels of the last image line, that is, the bottom right of each transmitted image is replaced by 11 stamp pixels. Each stamp pixel carries an 8 bit value that conveys information about the area of interest of the transmitted image.

The table below shows the function of each stamp pixel by position. A more detailed explanation of how to interpret the pixel values follows the table.

Position	Function
S1	AOI Sequence Position Number
S2	AOI Sequence Run Counter
S3	Frame Counter
S4	AOI Starting Column (MSByte)
S5	AOI Starting Column (LSByte)
S6	AOI Width (MSByte)

Position	Function
S7	AOI Width (LSByte)
S8	AOI Starting Line (MSByte)
S9	AOI Starting Line (LSByte)
S10	AOI Height (MSByte)
S11	AOI Height (LSByte)

Table 3-2: Stamp Pixel Functions

Stamp Pixels S1 and S2: Stamp pixels S1 and S2 are only active when the Programmable AOI Sequencer feature is used (see Section 3.8.4). S1 represents the position number of the area of interest within the AOI sequence. You can look up the position number in the AOI list so you know which settings were used to capture the image. S2 represents the run counter. If the Programmable Area of Interest Sequencer feature is disabled, all bits are set to 0.

Stamp Pixel S3: Stamp pixel S3 represents the 8 bit frame counter. The frame counter increments by one for each image captured by the camera. The counter starts at 0 and wraps at 255 (decimal). The frame counter is reset to 0 whenever the camera is switched off or reset. It is also reset to 0 when the stamp feature is disabled.

Stamp Pixels S4 through S11: Stamp pixels S4 and S5, S6 and S7, S8 and S9, and S10 and S11 represent the most significant byte and least significant byte (respectively) of the AOI starting column, AOI width, AOI starting line, and AOI height.

When the camera is operating in an 8 bit output mode, the stamp pixels will be 8 bit values. When the camera is operating in a 10 bit output mode, the stamp pixels will be 10 bit values but only the 8 LSBs will carry information. The two MSBs will be packed with zeros.

Enabling/Disabling the Stamp

You can enable/disable the stamp feature by using the Camera Configuration Tool Plus (CCT+) or by using binary write commands from within your own application to set the camera's control and status registers (CSRs).

With the CCT+

With the CCT+ (see Section 4.1), you use the Stamp parameter in the Output parameters group to enable or disable the stamp feature.

By Setting CSRs

You can enable/disable the stamp feature by writing a value to the Mode field of the Stamp CSR (see page 4-30).

See Section 4.2.2 for an explanation of CSRs. See Section 4.3.1 for an explanation of using read/write commands.

3.10 Mirror Image

When the mirror image feature is enabled, the pixel values for each line will switch end-for-end about the line's center point. If you use full resolution, on each line the value for pixel 1 will be swapped with the value for pixel 2352, the value for pixel 2 will be swapped with the value for pixel 2351, the value for pixel 3 will be swapped with the value for pixel 2350, and so on.

The mirror image feature also works for AOIs. The swapping of pixel values is analogous to the swapping at full resolution. On each line the value of the first pixel is swapped with the value of the last pixel, the value of the second pixel is swapped with the value of the next-to-last pixel, and so on.

Note

If you use the mirror image feature for a color version, remember to also swap the assignments of colors (R, G, B) to the pixels (see Section 3.11) by setting your frame grabber appropriately. If, for example, the original sequence in a line was G, R, G, R, … the new sequence in the same line must be R, G, R, G, …

Note

If you run a sequence of AOIs and if you have enabled the mirror image feature, the mirror image feature will be applied to all AOIs.

Enabling/Disabling the Mirror Image

You can enable/disable the mirror image feature by using the Camera Configuration Tool Plus (CCT+) or by using binary write commands from within your own application to set the camera's control and status registers (CSRs).

With the CCT+

With the CCT+ (see Section 4.1), you use the mirror mode parameter in the Output parameters group to enable or disable the mirror image feature.

By Setting CSRs

You can enable/disable the mirror image feature by writing a value to the Mirror Mode field of the Mirror Image Mode CSR (see page 4-32).

See Section 4.2.2 for an explanation of CSRs. See Section 4.3.1 for an explanation of using read/write commands.

3.11 Color Creation in the A400kc

The CMOS sensor used in the A400kc is equipped with an additive color separation filter known as a Bayer filter. With the Bayer filter, each individual pixel is covered by a micro-lens which lets light of only one color strike the pixel. The pattern of the Bayer filter used in the A400kc is shown in Figure 3-17. As the figure illustrates, within each block of four pixels, one pixel sees only red light, one sees only blue light, and two pixels see only green light. (This combination mimics the human eye's sensitivity to color.)

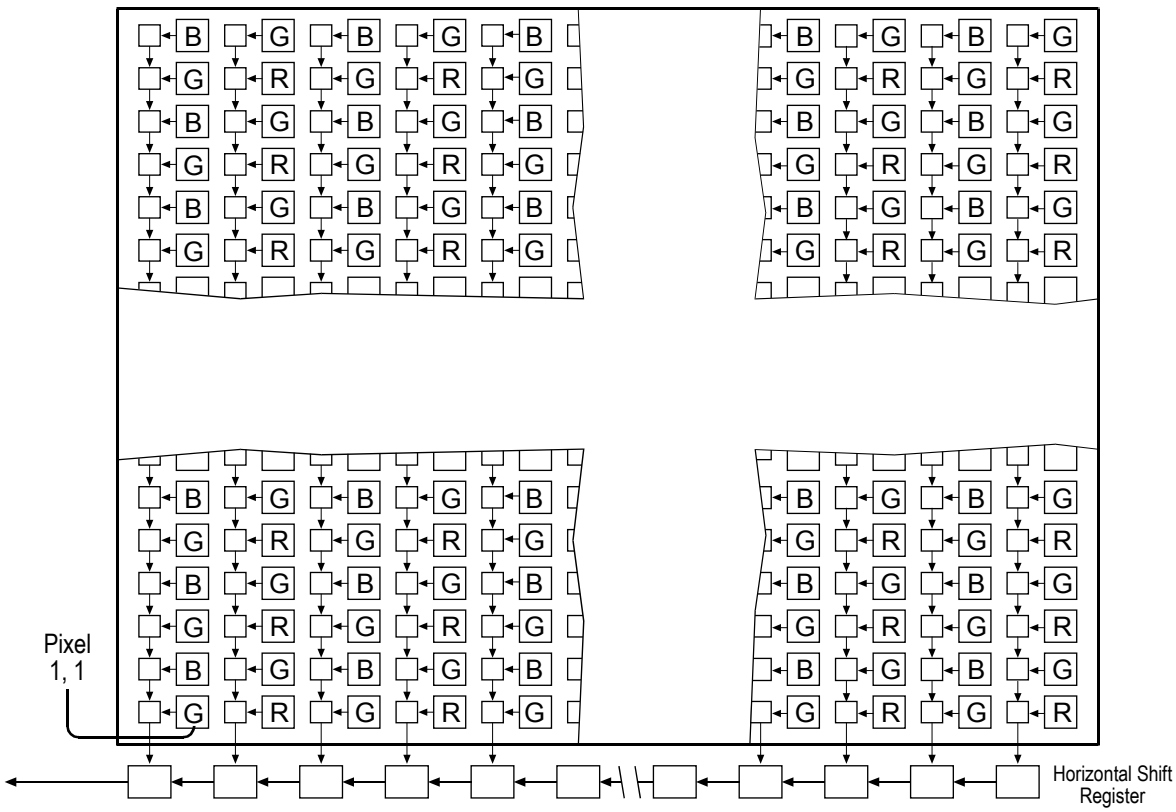


Figure 3-17: Bayer Filter Pattern on the A400kc

A single value is transmitted out of the camera for each pixel in a captured image. If you want to get full RGB color information for a given pixel in the image, you must perform a color interpolation using the information from the surrounding pixels. Some frame grabbers are capable of performing the color interpolation and many algorithms are available for performing the interpolation in your host PC.

3.12 Test Images

The test image mode is used to check the camera's basic functionality and its ability to transmit an image via the video data cable. The test image can be used for service purposes and for failure diagnostics. In test image mode, the image is generated with a software program and the camera's digital devices and does not use the optics, CMOS sensor, or ADCs. Four test images are available.

Note

DSNU and PRNU shading correction produce distortion in the test image. Disable DSNU and PRNU shading correction before you enable a test image.

3.12.1 Test Image One (Vertical Stripe Pattern)

Test image one is useful for determining if your frame grabber has dropped any columns from your image.

The stripes in the vertical stripe test pattern are formed with a gradient that ranges from 0 to 255 (8 bit mode) or 0 to 1023 (10 bit mode). A full stripe is 256 columns (8 bit mode) or 1024 columns (10 bit mode) wide. As an exception, the gray values of the first stripe range from 1 to 255 or from 1 to 1023, respectively.

The pixels in column one of the first stripe all have a value of 1. The pixels in column two of the first stripe all have a value of 2, the pixels in column three of the first stripe all have a value of 3, and so on. This pattern continues until column 255 (8 bit mode), where the pixels have a gray value of 255, or column 1023 (10 bit mode), where the pixels have a value of 1023.

Figure 3-18: Test Image One (8 bit)

In 8 bit mode, a second stripe begins in column 256. The pixels in column 256 have a gray value of 0, the pixels in column 257 have a value of 1, the pixels in column 258 have a value of 2, and so on. This pattern continues until column 511 where the pixels have a gray value of 255.

A third stripe begins in column 512. The pixels in column 512 have a gray value of 0, the pixels in column 513 have a value of 1, the pixels in column 514 have a value of 2, and so on. This pattern continues until column 2352 where the pixels have a value of 48.

In 10 bit mode, a second stripe begins in column 1024. The pixels in column 1024 have a value of 0, the pixels in column 1025 have a value of 1, the pixels in column 1026 have a value of 2, and so on. This pattern continues until column 2047 where the pixels have a value of 1023.

A third stripe begins in column 2048. The pixels in column 2048 have a value of 0, the pixels in column 2049 have a value of 1, the pixels in column 2050 have a value of 2, and so on. This pattern continues until column 2352 where the pixels have a value of 304.

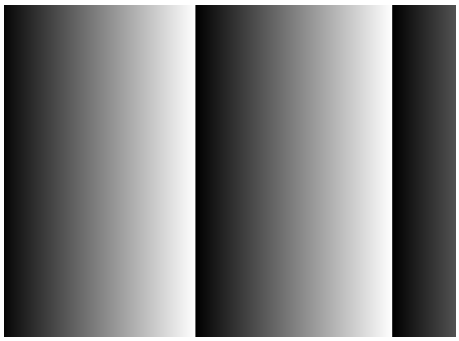


Figure 3-19: Test Image One (10 bit)

3.12.2 Test Image Two (Still Diagonal Stripe Pattern)

Test image two is useful for determining if your frame grabber has dropped any columns or lines from your image.

The stripes in the still diagonal stripe test pattern are formed with repeated gray scale gradients ranging from 0 to 255 (in 8 bit output mode) or 0 to 1023 (in 10 bit output mode). As an exception, the gray values of the first stripe range from 1 to 255 or from 1 to 1023, respectively.

The top line starts with a gray value of 1 on pixel 1. The second line starts with a gray value of 2 on pixel 1. The third line starts with a gray value of 3 on pixel 1, and so on. Line 255 (8 bit mode) or 1023 (10 bit mode) starts with a gray value of 255 or 1023 on pixel 1. Line 256 (8 bit mode) or 1024 (10 bit mode) restarts with a gray value of 0 on pixel 1, and so on.

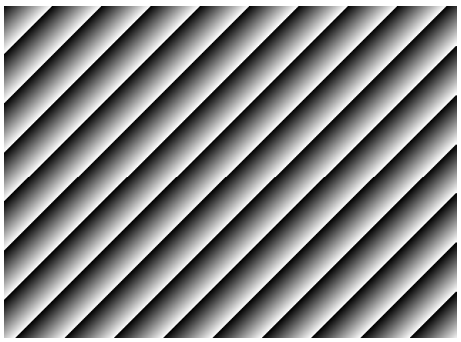


Figure 3-20: Test Image Two (8 Bit)

Depending on the output mode selected on the camera, either the 8 bit test image or the 10 bit test image will be active.

The mathematical expression for the test image is as follows:

8 bit: Gray level = [x + y - 1] MOD 256

10 bit: Gray level = [x + y - 1] MOD 1024

where x and y are natural numbers enumerating lines and columns, respectively. According to the number of pixels present, x ranges in steps of 1 from 1 to 1726 and y ranges in steps of 1 from 1 to 2352.

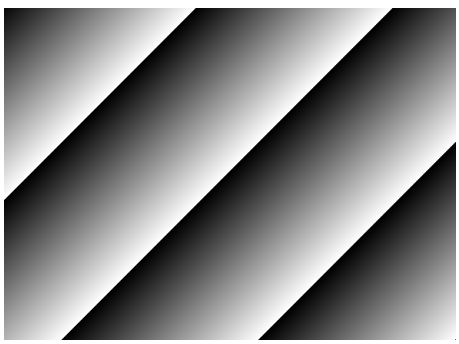


Figure 3-21: Test Image Two (10 Bit)

The expression is shown graphically in Figure 3-22.

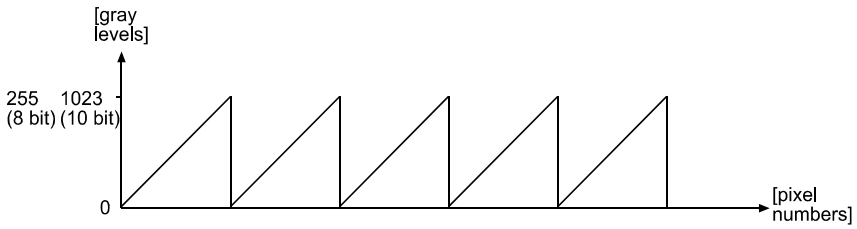


Figure 3-22: Formation of Monochrome Test Image

3.12.3 Test Image Three (Moving Diagonal Stripe Pattern)

Test image three is useful for determining if your camera is reacting to an ExSync signal.

The basic pattern of the test image is a diagonal stripe pattern as explained in Section 3.12.2, but the pattern of the image moves up by one pixel each time the ExSync signal cycles. When you view the output of a camera that is set for test image three, the pattern should appear to be gradually moving up the screen.

If the camera is set for free-run, each cycle of the camera's internal control signal will cause the pattern of the test image to move up by one pixel.

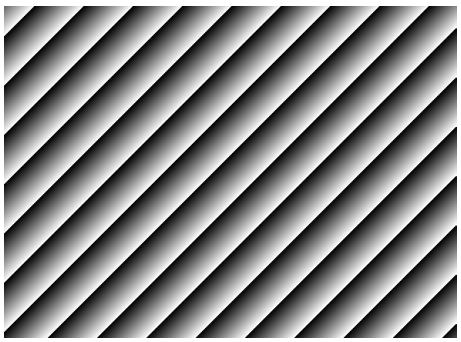


Figure 3-23: Test Image Three (8 Bit)

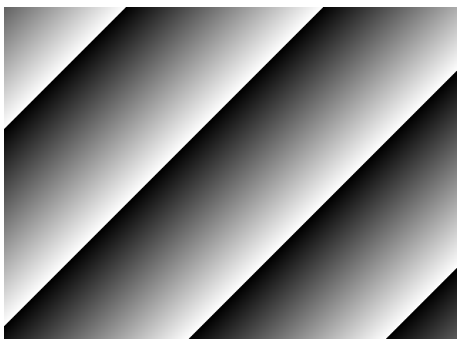


Figure 3-24: Test Image Three (10 Bit)

3.12.4 Test Image Four (Horizontal Stripe Pattern)

Test image four is useful for determining if your frame grabber has dropped the first line from your image.

The stripes in the horizontal stripe test pattern are formed with a gradient that ranges from 0 to 255 (8 bit mode) or 0 to 1023 (10 bit mode). A full stripe is 256 lines (8 bit mode) or 1024 lines (10 bit mode) high. As an exception, the gray values of the first stripe range from 1 to 255 or from 1 to 1023, respectively.

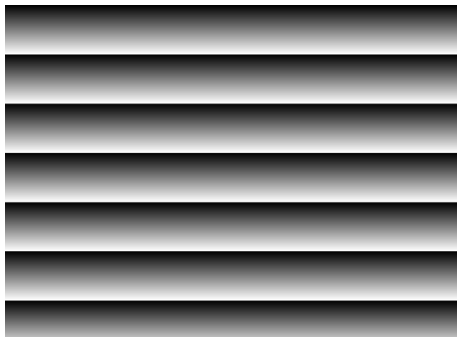


Figure 3-25: Test Image Four (8 bit)

The pixels in line one of the first stripe all have a value of 1. The pixels in line two of the first stripe all have a value of 2, the pixels in line three of the first stripe all have a value of 3, and so on. This pattern continues until line 255 (8 bit mode), where the pixels have a gray value of 255, or line 1023 (10 bit mode), where the pixels have a value of 1023.

In 8 bit mode, a second stripe begins in line 256. The pixels in line 256 have a gray value of 0, the pixels in line 257 have a value of 1, the pixels in line 258 have a value of 2, and so on. This pattern continues until line 511 where the pixels have a gray value of 255.

A third stripe begins in line 512. The pixels in line 512 have a gray value of 0, the pixels in line 513 have a value of 1, the pixels in line 514 have a value of 2, and so on. This pattern continues until line 1726 where the pixels have a value of 190.

In 10 bit mode, a second stripe begins in line 1024. The pixels in line 1024 have a value of 0, the pixels in line 1025 have a value of 1, the pixels in line 1026 have a value of 2, and so on. This pattern continues until line 1726 where the pixels have a value of 702.

Figure 3-26: Test Image Four (10 bit)

3.12.5 Guidelines When Using Test Images

When using a test image, please take the following guidelines into account:

- When a test image is active, the gain, offset, and exposure time have no effect on the image.
- DSNU and PRNU shading correction produce distortion in the test image. Disable DSNU and PRNU shading correction before you enable a test image.
- Digital shift makes test images appear very light. Disable digital shift when a test image is active.
- Use of the area of interest feature will effect the appearance of test images.
- If the camera is set for an exposure mode that uses an ExSync signal, the ExSync signal must be present and must toggle in order for the camera to output test images. If the camera is set for free-run, each cycle of the camera's internal sync signal will trigger the output of a test image.

3.12.6 Enabling/Disabling Test Images

You can enable/disable a test image by using the Camera Configuration Tool Plus (CCT+) or by using binary write commands from within your own application to set the camera's control and status registers (CSRs).

With the CCT+

With the CCT+ (see Section 4.1), you use the Test Image setting in the Output parameter group to enable/disable a test image.

By Setting CSRs

You can enable/disable a test image by writing a value to the Mode field of the Test Image Mode CSR (see page 4-32).

See Section 4.2.2 for an explanation of CSRs. See Section 4.3.1 for an explanation of using read/write commands.

3.13 Camera Temperature

A400k series cameras include a sensor that measures the temperature on one of the electronic boards inside of the camera. The sensor's readings let you monitor whether ventilation is working correctly. The camera's allowed inner temperature is stated in Section 1.5.

3.13.1 Reading the Camera Temperature

You can read out the current temperature by using the Camera Configuration Tool Plus (CCT+) or by using binary read commands from within your own application to read the camera's control and status registers (CSRs).

With the CCT+

With the CCT+ (see Section 4.1), you use the Camera Temperature setting in the Camera Information parameter group to read the camera's inner temperature.

By Setting CSRs

You can read the camera's inner temperature by reading a value from the Camera Temperature field of the Camera Temperature CSR (see page 4-8).

See Section 4.2.1 for an explanation of inquiry CSRs. See Section 4.3.1 for an explanation of using read/write commands.

3.14 Configuration Sets

A configuration set is a set of values that contains all of the parameters needed to control the camera. There are two basic types of configuration sets: the work configuration set and the factory configuration set.

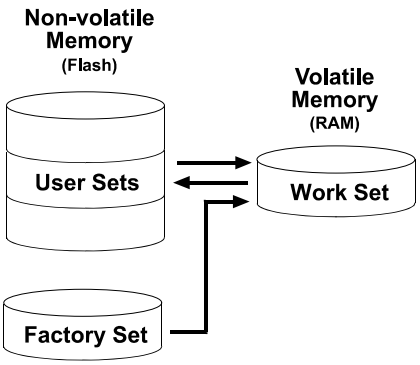


Figure 3-27: Configuration Sets

Work Configuration Set

The work configuration set contains the camera's current settings and thus determines the camera's performance, that is, what your image currently looks like. If you use the CCT+ to change the camera settings or if you change settings by writing to the camera's registers, you are making changes to the work configuration set. The work configuration set is located in the camera's volatile memory and the settings are lost if the camera is reset or if power is switched off. The work configuration set is usually just called the "work set" for short.

Factory Configuration Set

When a camera is manufactured, a test setup is performed on the camera and an optimized configuration is determined. The factory configuration set contains the camera's factory optimized configuration. The factory set is saved in a permanent file in the camera's non-volatile memory. The factory set can not be altered and since it is stored in non-volatile memory, it is not lost when the camera is reset or switched off. The factory configuration set is usually just called the "factory set" for short.

3.14.1 Saving Configuration Sets

As mentioned above, the work configuration set is located in the camera's volatile memory and the settings are lost if the camera is reset or if power is switched off. A400k cameras can save the current work set values in the volatile memory to a file in the camera's non-volatile memory. Files saved in the non-volatile memory are not lost at reset or power off. You can save up to four configuration sets to files in the non-volatile memory. These saved configuration sets are commonly referred to as "user configuration sets" or "user sets".

Saving a Configuration Set

You can save the current work set to a file in the non-volatile memory by using the Camera Configuration Tool Plus (CCT+) or by using binary read/write commands from within your own application to set the camera's control and status registers (CSRs).

With the CCT+

With the CCT+ (see Section 4.1), you use the File Name Select parameter and the Create User Set parameter in the User Set Files parameters group. Make sure that you save the work set to user set 1, 2, 3 or 4 only. Further user sets are offered but must not be used. If you select to save it to user set 5 or higher, the work set will not be saved.

By Setting CSRs

You can save the current work set to a file in the non-volatile memory by writing values to the bulk data CSR for configuration sets. The bulk data "save" process is used to save the work set to a file.

Section 4.2.3 explains the bulk data CSRs and explains how to use the CSRs to save the work set to a file. Section 4.3.1 explains using read/write commands.

3.14.2 “Activating” a Saved User Set File

As explained in Section 3.14.1, you can save configuration sets to files in the camera’s non-volatile memory. These saved configuration sets are commonly referred to as “user configuration sets” or “user sets.”

If you have saved one or more user set files, you can choose to “activate” one of the stored files. When you activate a stored user set file, two things happen:

- The values from the stored user set file are copied into the work set in the camera’s volatile memory. The camera will now use the configuration values that were copied into the work set.
- A link is created between the activated user set file and the camera’s volatile memory. The values in the activated user set file will now be automatically copied into the work set whenever the camera is powered up or reset.

Activating a Stored User Set File

You can activate a stored user set file by using the Camera Configuration Tool Plus (CCT+) or by using binary read/write commands from within your own application to set the camera’s bulk data control and status registers (CSRs).

With the CCT+

With the CCT+ (see Section 4.1), you use the File Name Select parameter and the Activate User Set parameter in the User Set Files parameters group to activate a saved user set file.

By Setting CSRs

You can activate a stored user set file by writing values to the bulk data CSR for configuration sets. The bulk data “activate” process is used to activate a file.

Section 4.2.3 explains bulk data CSRs and using the bulk data activate process. Section 4.3.1 explains using read/write commands.

3.14.3 “Activating” the Factory Set File

As explained on page 3-43, a factory configuration set containing an optimized set of parameters is created when the camera is manufactured. The factory set is saved in a permanent file in the camera’s non-volatile memory. The factory set file can not be altered or deleted and is not lost when the camera is switched off.

You can activate the factory set file in a manner that is similar to activating one of your saved user set files. Activating the factory set file is a good way to return the camera to normal operation if you have severely misadjusted some of the camera’s parameters and you are not sure how to recover.

When you activate the factory set, two things happen:

- The values from the factory set file are copied into the work set in the camera’s volatile memory. The camera will now use the factory set values that were copied into the work set.
- A link is created between the factory set file and the camera’s volatile memory. The values in the factory set will now be automatically copied into the work set whenever the camera is powered up or reset.

Activating the Factory Set File

You can activate the factory set file with the Camera Configuration Tool Plus (CCT+) or by using binary read/write commands from within your own application to set the camera's bulk data control and status registers (CSRs).

With the CCT+

With the CCT+ (see Section 4.1), you use the File Name Select parameter and the Activate User Set parameter in the User Set Files parameters group to activate the factory set file.

By Setting CSRs

You can activate the factory set file by writing values to the bulk data CSR for configuration sets. The bulk data "activate" process is used to activate the factory set file.

Section 4.2.3 explains bulk data CSRs and using the bulk data activate process. Section 4.3.1 explains using read/write commands.

3.14.4 Which Configuration Set File Will Load at Startup or at Reset?

On the initial wake-up after delivery, the camera copies the factory set into the work set.

At each subsequent power on or reset, the configuration set file that was last activated is copied into the work set.

If there is no activated file, the factory set file will be copied into the work set.

3.14.5 Saving a User Set to PC, Loading a User Set from PC

You can save a user set to the hard disk of your computer and load a user set from hard disk into your camera. This is useful if you wish to use this user set on another camera of the same type.

Saving a User Set to PC or Loading a User Set from PC

You can save a user set to PC or load a user set from PC by using the Camera Configuration Tool Plus (CCT+) or by using binary read/write commands from within your own application to set the camera's bulk data control and status registers (CSRs).

With the CCT+

With the CCT+ (see Section 4.1), you use the Save Work Set to File command in the File menu to save the work set to hard disk and you use the Load Work Set from File command in the File menu to load the work set from hard disk.

By Setting CSRs

You can save a user set to PC or load a user set from PC by writing values to the bulk data CSR for configuration sets. The bulk data "download" process is used to save a user set to PC. The bulk data "upload" process is used to load a user set from PC.

Section 4.2.3 explains bulk data CSRs and using the bulk data download and upload processes. Section 4.3.1 explains using read/write commands.

3.15 Parameter Set Cache

When the parameter set cache feature is enabled, you can modify the camera's parameter settings without the modifications becoming effective immediately.

The parameter set cache feature lets you continue valid image capture while you change your parameters. For example, while setting a new area of interest with the parameter set cache feature enabled, you can still capture images using your old area of interest settings.

When the parameter set cache feature is enabled, all modifications are written to the camera but they do not become active. The camera continues to operate under the control of the old settings. The modifications will only become active after the parameter set cache feature is disabled again. When the parameter set cache feature is disabled again, all modifications become active simultaneously after the last valid frame that used the old settings.

Parameter set cache is effective for modifications to the video data format, exposure time control mode, exposure time, frame period, area of interest, and test image only. Modifications to other parameter settings will become active immediately even if parameter set cache is enabled.

To avoid rejections (see Section 3.16), make sure that your order of modifications produces valid combinations after every modification. For example, to change the area of interest from starting column = 861, width = 512 to full resolution, set the starting column to 1 first, and only afterwards the width to 2,352 (see Section 3.8). Setting the width first would cause the modification to be rejected by the camera.

3.15.1 Enabling/Disabling Parameter Set Cache

You can enable/disable the parameter set cache feature by using the Camera Configuration Tool Plus (CCT+) or by using binary write commands from within your own application to set the camera's control and status registers (CSRs).

With the CCT+

With the CCT+ (see Section 4.1), you use the Parameter Set Cache setting in the Parameter Set parameter group to enable/disable parameter set cache.

By Setting CSRs

You can enable/disable parameter set cache by writing a value to the Mode field of the Parameter Set Cache CSR (see page 4-34).

See Section 4.2.2 for an explanation of CSRs and Section 4.3.1 for an explanation of using read/write commands.

3.16 Parameter Validation

Before a modification to a parameter setting becomes active, the microcontroller inside the camera automatically verifies that the setting causes no conflict. If the camera detects a parameter error, it will automatically discard the setting and the old setting remains valid.

A parameter error occurs if the parameter is set out of range, the parameter is set to an invalid value, or the parameters which depend on each other are set in conflict.

Since the CCT+ automatically checks that parameters are set correctly, you will not normally see a parameter error situation when you set parameters with the CCT+. When you set parameters using binary commands, you may see parameter error situations if you inadvertently set parameters to values that are not allowed or are in conflict. If you suspect that the camera is in a parameter error situation, you can read the value in the Camera Status field of the Camera Status inquiry register (see page 4-9). If the parameter error bit is set, then a parameter error situation is present.

A simple way to recover from a parameter error situation is to activate the camera's factory configuration set (see Section 3.14.3). Activating the factory set will load a set of factory determined optimal parameters into the camera.

If you are setting the camera's parameters by using binary commands to write to registers, make sure you check the min, max and increment fields of each register before you set the parameter values. Setting the values within the min and max and using the specified increments will avoid parameter errors.

3.17 Checking the Camera Status

A400k series cameras monitor their status by performing a regular series of self checks. You can view the current camera status in several ways:

- by using the Camera Configuration Tool Plus (see Section 4.1). Check the Camera Status parameter in the Camera Information parameter group to see if any error codes are present.
- by using binary read/write commands from within your own application to read the value in the Camera Status field of the Camera Status inquiry register (see page 4-9).

  See Section 4.2.1 for an explanation of inquiry registers. See Section 4.3.1 for an explanation of using read/write commands.
- by checking the LED on the back of the camera. If certain error conditions are present, the LED will blink (see Section 6.1).

3.18 Status LED

The A400k has a status LED on the back of the camera. The LED is used to indicate that power is present and to indicate an error condition if one is detected. See Section 6.1 for details.

3.19 Resetting the Camera

A400k cameras let the user initiate a camera reset. A reset is the equivalent of switching off power to the camera and switching power back on.

You can initiate a camera reset by using the Camera Configuration Tool Plus (CCT+) or by using binary write commands from within your own application to set the camera's control and status registers (CSRs).

With the CCT+

With the CCT+ (see Section 4.1), click on "Camera" in the menu at the top of the CCT+ window and a drop down list will appear. Click on "Reset Camera" in the drop down list to initiate a reset.

By Setting CSRs

You can initiate a reset by writing a value to the Reset field of the Camera Reset CSR (see page 4-34).

See Section 4.2.2 for an explanation of CSRs. See Section 4.3.1 for an explanation of using read/write commands.

Whenever the camera is powered on or when a camera reset is performed, your PC may receive some random characters on the serial interface. We recommend clearing the serial input buffers in your PC after a camera power on or reset.

4 Configuring the Camera

A400k cameras come with a factory set of configuration parameters and they will work properly for most applications with only minor changes to the configuration. For normal operation, the following parameters are usually configured by the user:

- Video data output mode
- Exposure time control mode
- Exposure time (for ExSync programmable mode or free-run programmable mode)
- Frame period (for ExSync programmable mode or free-run programmable mode)

To customize operation for your particular application, the following parameters can also be configured:

- Gain
- Offset
- Shading Correction
- Digital Shift
- Area of Interest (AOI)
- Stamp
- Programmable AOI Sequencer
- Flash Trigger
- Mirror Image
- Parameter Set Cache

The camera is programmable via the RS-644 serial connection in the Camera Link interface between the frame grabber and the camera. Two methods can be used to change the camera's parameters. The first and easier approach is to change the parameters using the Camera Configuration Tool Plus (CCT+). See Section 4.1 for instructions on using the configuration tool. You can also change the parameters directly from your application by using binary read/write commands to set the camera's registers (see Section 4.2).

4.1 Configuring the Camera with the Camera Configuration Tool Plus (CCT+)

The Camera Configuration Tool Plus (CCT+) is a Windows™ based program used to easily change the camera's parameter settings. The tool communicates via the RS-644 serial connection in the Camera Link interface between the frame grabber and the camera. The tool automatically generates the binary programming commands that are described in Section 4.3. For instructions on installing the tool, see the installation booklet that was shipped with the camera.

This manual assumes that you are familiar with Microsoft Windows and that you have a basic knowledge of how to use programs. If not, please refer to your Microsoft Windows manual.

4.1.1 Opening the Configuration Tool

1. Make sure that the properties for the RS-644 serial port on your frame grabber are properly configured and that the camera has power.
2. To start the CCT+, click **Start**, click **All Programs**, click **Basler Vision Technologies**, and click **CCT**+ (default installation).

   During start-up, a start-up screen can be seen.

   If start-up is successful, the tool will open. To familiarize yourself with using the tool, press the F1 key and look through the online help included with the tool.

   If an error occurs, the tool is automatically closed after start-up. Refer to the CCT+ Installation Guide for possible causes.

4.1.2 Closing the Configuration Tool

Close the CCT+ by clicking on the ☒ button in the upper right corner of the window.

4.1.3 Configuration Tool Basics

The volatile (RAM) memory in the camera contains the set of parameters that controls the current operation of the camera. This set of parameters is known as the work configuration set or "work set" (see Section 3.14). The CCT+ is used to view the present settings for the parameters in the work set or to change the settings.

When the CCT+ is opened and a port is selected, it queries the camera and displays a list of the current settings for the parameters in the work set.

To simplify navigation, parameters are organized in related groups. For example, all parameters related to the camera output can be found in the **Output** group.

When you click on the plus or minus sign beside a group (+ or -), the parameters in this group will be shown or hidden, respectively.

To get an overview of all parameters available on the connected camera, maximize the CCT+ window and click the + sign beside each group.

The camera parameter names always appear in the left column of the list. The current setting for each parameter appears in the right column.

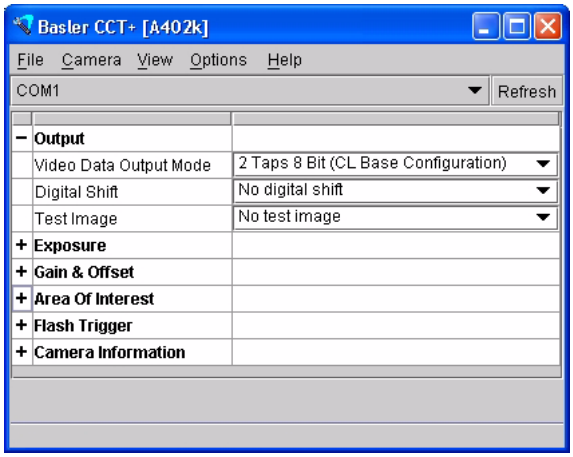


Figure 4-1: Output Group

By default, a **Parameter Description** window is displayed. In this window, you can find basic information on the selected parameter and if present, on the dependencies that may exist between the selected parameter and other parameter(s).

Modifiable parameter settings and available commands appear in black while read-only settings and unavailable commands appear in gray.

If you make a change to one of the parameter settings, that change will immediately be transmitted from the CCT+ to the camera's Work Set. Because the parameters in the Work Set control the current operation of the camera, you will see an immediate change in the camera's operation. If the change limits the range of available settings for other parameters, the available ranges will automatically be refreshed.

By default, the CCT+ also automatically updates the displayed settings every 5 seconds. The feature behind this behavior is called Auto Refresh. If auto refresh is not enabled, the display will not update when a camera setting is changed using another tool, when power to the camera is switched off and on, or when the connected camera is exchanged while the CCT+ is displaying the camera settings. To manually refresh the display, you can use the **Refresh** button in the top right corner of the tool.

Keep in mind that the work set is stored in the camera's volatile memory. Any changes you make to the work set using the configuration tool will be lost when the camera is switched off. To save changes you make to the work set, save the modified work set to one of the camera's four user set files. The user set files are stored in non-volatile memory and will not be lost when the camera is switched off (see Section 3.14).

Alternatively, you can also save the Work Set to the hard disk of your computer and load it from hard disk.

4.1.4 Configuration Tool Help

The CCT+ includes a complete on-line help file which explains how to change parameter settings. It also explains how to copy the work set to a saved user set file and how to copy a saved user set file or the factory set file to the work set. To access on-line help, press the F1 key whenever the configuration tool is active.

4.2 Configuring the Camera By Setting Registers

A400k cameras have blocks of mapped memory space known as registers. By reading values from the registers, you can determine basic information about the camera and information about the camera's current parameter settings. By writing values to the registers, you can set camera parameters and control how the camera's features will operate. There are three types of registers in an A400k:

- Inquiry Registers - these registers provide basic information about the camera. Section 4.2.1 explains inquiry registers in more detail, lists the inquiry registers in the camera, and describes the function of each register.
- Feature Control and Status Registers - these registers let you set the parameters associated with the camera's features. Section 4.2.2 explains feature control and status registers in more detail, lists the feature control and status registers in the camera, and describes the function of each register.
- Bulk Data Control and Status Registers - these registers let you store and recall sets of values that the camera uses as a group. Section 4.2.3 explains bulk data and the bulk data control and status registers in more detail. It also lists the bulk data registers in the camera and describes the function of each register.

A special binary read/write command protocol is used to read from and write to the registers in the camera. Read and write commands are sent to the camera via the standard serial link between the camera and the frame grabber. Section 4.3 describes the binary read/write command protocol in detail and also provides information on using the serial link. Section 4.4 provides code samples which illustrate how to use the read/write commands.

4.2.1 Inquiry Registers

Inquiry registers contain basic information about the camera and information about the camera's current status. Each inquiry register contains one or more fields and each field has an assigned address within the camera's memory space. By using a binary read command, you can read the data in a field and get information about the camera.

The address for any field within a register is equal to the register base address plus the offset for the field. For example, the Vendor Information Inquiry Register (see below) has a Vendor Name field with an address of 0x0101 (the base address of 0x0100 plus an offset of 0x0001). By reading the data at address 0x0101, you can get information about the camera vendor's name.

Section 4.2.1.1 lists the inquiry registers in A400k cameras and shows detailed information about the use of each field within the registers.

4.2.1.1 Inquiry Register Details

Vendor Information Inquiry

Register Base Address: 0x0100			
Field Name: Register Status	**Offset:** 0x0000	**Size:** 1 Byte	**Type:** Read only
Description: The integer value in this field indicates the status of this inquiry register: 0x00 = The register is not available. 0x01 = The register is available.			
Field Name: Vendor Name	**Offset:** 0x0001	**Size:** 20 Bytes	**Type:** Read only
Description: String containing the camera vendor's name. The string is zero terminated if less than 20 bytes are needed and unterminated if all 20 bytes are needed.			

Model Information Inquiry

Register Base Address: 0x0200			
Field Name: Register Status	**Offset:** 0x0000	**Size:** 1 Byte	**Type:** Read only
Description: The integer value in this field indicates the status of this inquiry register: 0x00 = The register is not available. 0x01 = The register is available.			
Field Name: Model Info	**Offset:** 0x0001	**Size:** 20 Bytes	**Type:** Read only
Description: String containing the camera's model number. The string is zero terminated if less than 20 bytes are needed and unterminated if all 20 bytes are needed.			

Product ID Inquiry

Register Base Address: 0x0300			
Field Name: Register Status	**Offset:** 0x0000	**Size:** 1 Byte	**Type:** Read only
Description: The integer value in this field indicates the status of this inquiry register: 0x00 = The register is not available. 0x01 = The register is available.			
Field Name: Product ID	**Offset:** 0x0001	**Size:** 20 Bytes	**Type:** Read only
Description: String containing the camera's product ID number. The string is zero terminated if less than 20 bytes are needed and unterminated if all 20 bytes are needed.			

Serial Number Inquiry

Register Base Address: 0x0400			
Field Name: Register Status	**Offset:** 0x0000	**Size:** 1 Byte	**Type:** Read only
Description: The integer value in this field indicates the status of this inquiry register: 0x00 = The register is not available. 0x01 = The register is available.			
Field Name: Serial Number	**Offset:** 0x0001	**Size:** 20 Bytes	**Type:** Read only
Description: String containing the camera's serial number. The string is zero terminated if less than 20 bytes are needed and unterminated if all 20 bytes are needed.			

Camera Version Inquiry

Register Base Address: 0x0500			
Field Name: Register Status	**Offset:** 0x0000	**Size:** 1 Byte	**Type:** Read only
Description: The integer value in this field indicates the status of this inquiry register: 0x00 = The register is not available. 0x01 = The register is available.			
Field Name: Camera Version	**Offset:** 0x0001	**Size:** 3 Bytes	**Type:** Read only
Description: The value in this field indicates the camera's version information. The bytes in the field are interpreted as follows: Byte 1 = Low byte of the camera version (BCD coded) Byte 2 = High byte of the camera version (BCD coded) Byte 3 = Register layout ID (BCD coded)			

Microcontroller Firmware Version Inquiry

Register Base Address: 0x0700			
Field Name: Register Status	**Offset:** 0x0000	**Size:** 1 Byte	**Type:** Read only
Description: The integer value in this field indicates the status of this inquiry register: 0x00 = The register is not available. 0x01 = The register is available.			
Field Name: Microcontroller Firmware Version	**Offset:** 0x0001	**Size:** 3 Bytes	**Type:** Read only
Description: The value in this field indicates the camera's microcontroller firmware version information. The bytes in the field are interpreted as follows: Byte 1 = Low byte of the firmware version (BCD coded) Byte 2 = High byte of the firmware version (BCD coded) Byte 3 = Register layout ID (BCD coded)			

Processing Board's FPGA Firmware Version Inquiry

Register Base Address: 0x0800			
Field Name: Register Status	**Offset:** 0x0000	**Size:** 1 Byte	**Type:** Read only
Description: The integer value in this field indicates the status of this inquiry register: 0x00 = The register is not available. 0x01 = The register is available.			
Field Name: Main FPGA Firmware Version	**Offset:** 0x0001	**Size:** 3 Bytes	**Type:** Read only
Description: The value in this field indicates the firmware version of the camera's FPGA (field programmable gate array) that is used on the processing board. The bytes in the field are interpreted as follows: Byte 1 = Low byte of the firmware version (BCD coded) Byte 2 = High byte of the firmware version (BCD coded) Byte 3 = Register layout ID (BCD coded)			

Sensor Board's FPGA Firmware Version Inquiry

Register Base Address: 0x0900			
Field Name: Register Status	**Offset:** 0x0000	**Size:** 1 Byte	**Type:** Read only
Description: The integer value in this field indicates the status of this inquiry register: 0x00 = The register is not available. 0x01 = The register is available.			
Field Name: Sensor FPGA Firmware Version	**Offset:** 0x0001	**Size:** 3 Bytes	**Type:** Read only
Description: The value in this field indicates the firmware version of the camera's FPGA (field programmable gate array) that is used on the sensor board. The bytes in the field are interpreted as follows: Byte 1 = Low byte of the firmware version (BCD coded) Byte 2 = High byte of the firmware version (BCD coded) Byte 3 = Register layout ID (BCD coded)			

Camera Temperature Inquiry

Register Base Address: 0x2600			
Field Name: Register Status	**Offset:** 0x0000	**Size:** 1 Byte	**Type:** Read only
Description: The integer value in this field indicates the status of this inquiry register: 0x00 = The register is not available 0x01 = The register is available			
Field Name: Camera Temperature	**Offset:** 0x0001	**Size:** 1 Byte	**Type:** Read only
Description: The value in this field indicates the camera's inner temperature. The result is given in °C as 8 bit signed number.			

Camera Status Inquiry

The camera has been programmed to detect several error conditions. When an error condition is detected, a flag is set. The camera status inquiry register lets you read the error flags.

Register Base Address: 0x0C00

Field Name: Register Status **Offset:** 0x0000 **Size:** 1 Byte **Type:** Read only

Description: The integer value in this field indicates the status of this inquiry register:

0x00 = The register is not available.

0x01 = The register is available.

Field Name: Camera Status **Offset:** 0x0001 **Size:** 4 Bytes **Type:** Read only

Description: Each bit in this field specifies an error condition (see table below). Bit 0 is the least significant bit. If a bit is set to 1, the error condition assigned to that bit is present. If the bit is set to 0, the error is not present.

Bit	Description
0	The camera is unlocked.
1	The camera is booting or is busy performing an internal operation (such as generating shading values).
2	A reset has occurred. This bit is auto-cleared on read.
3	Parameter error, for example, a parameter has been set to a value that is out of range or not allowed or in conflict with other settings.
4	A user set load has failed.
5	A file operation has failed.
6	Reserved
7	A binary read/write command protocol error has been detected. For more information about the error, read the Binary Command Protocol Status Inquiry register (see page 4-12). This bit is auto-cleared on read.
8 ... 15	Reserved
16	An FPGA not ready error has occurred. For more information about the error, read the FPGA Status Inquiry registers (see pages 4-10 and 4-11). This bit clears when you read the FPGA Status Inquiry registers.
17	A trigger error has occurred. Either the frame rate has been exceeded or the ExSync signal is missing. For more information about the error, read the FPGA Status Inquiry registers (see pages 4-10 and 4-11). This bit clears when you read the FPGA Status Inquiry registers.
18	Reserved
19	The last column FPN shading value generation process failed. The column FPN shading value generation process can fail if the pixel values in the frames captured during the generation process are too high. (The process should be performed in darkness or in very low light conditions.) This bit will clear when you perform a successful shading value generation procedure.
20 ... 31	Reserved

Processing Board's FPGA Status Inquiry

The camera has been programmed to detect several error conditions in its field programmable gate array (FPGA) on the processing board. When an error condition is detected, a flag is set. The FPGA status inquiry register lets you read the error flags.

Register Base Address: 0x0C10

Field Name: Register Status **Offset:** 0x0000 **Size:** 1 Byte **Type:** Read only

Description: The integer value in this field indicates the status of this inquiry register:

0x00 = The register is not available.

0x01 = The register is available.

Field Name: Main FPGA Status **Offset:** 0x0001 **Size:** 1 Byte **Type:** Read only

Description: Each bit in this field specifies an error condition (see table below). Bit 0 is the least significant bit. If a bit is set to 1, the error condition assigned to that bit is present. If the bit is set to 0, the error is not present.

Bit	Description
0	No FPGA firmware available.
1	FPGA firmware is available but the firmware has failed to load.
2	The camera's maximum frame rate has been exceeded.
3	There is no ExSync signal.
4	The FPGA is not ready.
5	Parameter error, for example, a parameter has been set to a value that is out of range or not allowed or in conflict with other settings.
6	Reserved
7	Reserved

Sensor Board's FPGA Status Inquiry

The camera has been programmed to detect several error conditions in its field programmable gate array (FPGA) on the sensor board. When an error condition is detected, a flag is set. The FPGA status inquiry register lets you read the error flags.

Register Base Address: 0x0C20

Field Name: Register Status **Offset:** 0x0000 **Size:** 1 Byte **Type:** Read only

Description: The integer value in this field indicates the status of this inquiry register:

0x00 = The register is not available.

0x01 = The register is available.

Field Name: Sensor FPGA Status **Offset:** 0x0001 **Size:** 1 Byte **Type:** Read only

Description: Each bit in this field specifies an error condition (see table below). Bit 0 is the least significant bit. If a bit is set to 1, the error condition assigned to that bit is present. If the bit is set to 0, the error is not present.

Bit	Description
0	No FPGA firmware available.
1	FPGA firmware is available but the firmware has failed to load.
2	The camera's maximum frame rate has been exceeded.
3	There is no ExSync signal.
4	The last shading value generation process failed.
5	The FPGA is not ready.
6	Parameter error, for example, a parameter has been set to a value that is out of range or not allowed or in conflict with other settings.
7	Reserved

Binary Command Protocol Status Inquiry

The camera has been programmed to detect several error conditions. When a protocol error is detected, a flag is set. The protocol status inquiry register lets you read the error flags.

Register Base Address: 0x0C30

Field Name: Register Status **Offset:** 0x0000 **Size:** 1 Byte **Type:** Read only

Description: The integer value in this field indicates the status of this inquiry register:

0x00 = The register is not available.

0x01 = The register is available.

Field Name: Protocol Status **Offset:** 0x0001 **Size:** 1 Byte **Type:** Read only

Description: Each bit in this field specifies an error condition (see table below). Bit 0 is the least significant bit. If a bit is set to 1, the error condition assigned to that bit is present. If the bit is set to 0, the error is not present.

Bit	Description
0	A binary command with no BFS was received (see Section 4.3.1)
1	A byte time-out has occurred (see Section 4.3.1.1).
2	A binary command with an invalid OpCode was received (see Section 4.3.1).
3	A binary command with no BFE was received (see Section 4.3.1).
4	A binary command with an incorrect BCC was received (see Section 4.3.1).
5	A binary command with an address error was received (see Section 4.3.1).
6	Reserved
7	An unknown error has occurred.

4.2.2 Feature Control and Status Registers

The feature control and status registers (CSRs) let you set the parameters for camera features such as exposure mode, gain, offset, and the AOI. These registers also let you check the current parameter settings and the status for each feature.

Each feature has one or more CSRs associated with it. The fields within a feature's CSR(s) are used to control how the feature operates. By using a binary write command to write to fields within a feature's CSR(s) you can change the parameter settings for the feature. By using binary read commands, you can determine the current setting for the parameters and get information about the feature's status.

The address for any field within a register is equal to the register base address plus the offset for the field. Look at the Video Data Output Mode CSR on page 4-15 as an example. The Mode field of this register has an address of 0x1701 (the base address of 0x1700 plus the offset of 0x0001). By writing a value to this address, you can select the video data output mode. By reading the value at this address, you can determine the current output mode setting.

The Video Data Output Mode CSR is a simple CSR with only two fields. Most of the other feature CSRs have several read/write fields that let you set the parameters associated with the feature. They may also have read only fields that contain information about the minimum and maximum allowed setting for each parameter. Section 4.2.2.2 lists the feature CSRs in A400k cameras and shows detailed information about the use of each field within the register.

4.2.2.1 "Raw" Value Fields vs. "Absolute" Value Fields

As you look through the descriptions of the feature CSRs, you will notice that some CSRs have a parameter that can be set by writing a value to a "raw" field or by writing a value to an "absolute" field. You will find this to be true for the Exposure Time, Frame Period, Gain and Offset parameters. The common characteristic among these parameters is that they are expressed as rational numbers. These numbers are on a continuous scale rather than on a scale of discrete integers. Any one of the parameters expressed as rational numbers can be set on a "raw" scale or on an "absolute" scale. A raw scale is simply a range of integer values that has no defined units. An absolute scale is a range of floating point values that has defined units.

Let's look at the Exposure Time parameter as an example:

Setting the Raw Exposure Time

If you adjust the exposure time by writing a value to the Raw Exposure Time field of the Exposure Time CSR, you can write any integer value from 1 to 4 192 982 (decimal). Writing an integer value to the exposure time register sets the exposure time, but it doesn't directly tell you how many microseconds of exposure time you will be getting from the camera at that setting. To determine the microseconds of exposure time you are getting at a particular raw exposure time setting, you must use the formula: microseconds = raw setting x 4.56 µs. For example, if the Raw Exposure Time field is set to 3 (decimal):

microseconds = raw setting x 4.56 µs

microseconds = 3 x 4.56 µs

microseconds = 13.68

So with the Raw Exposure Time field set to 3, the camera would be set for 13.68 µs of exposure.

Setting the Absolute Exposure Time

If you adjust the exposure time by writing a value to the Absolute Exposure Time field of the Exposure Time CSR, you can write a floating point value from 4.56 to 19119997.00 (decimal) in increments of 0.01. Writing a floating point value to the absolute register sets the exposure time directly in microseconds For example, if the absolute exposure time was set to 13.68, then the camera would be set for 13.68 µs of exposure.

Guidelines

When you are working with a parameter that can be entered as either raw or absolute, there are two important things to keep in mind:

- You do not need to enter values in both the raw field and the absolute field. Entering just one value is sufficient and you can choose between fields whichever one suits your needs best.
- However, in the absolute fields, only equivalents of values entered in the raw field are allowed.

  For illustration, let's consider the preceding example using the first formula from the previous page. Calculating the microseconds of exposure that the camera would produce for three different settings in the raw exposure time field of the exposure time CSR, we obtain:

Raw Exposure Time Value Setting	Resulting Exposure Time
1	4.56 µs
2	9.12 µs
3	13.68 µs

  This means that if you entered a value in the absolute field, values of e.g. 4.56, 9.12, or 13.68 would be "valid" because they are the equivalents to 1, 2, and 3, respectively, in the raw field.

  However, for example, a value of 13.69 entered in the absolute field would not be valid because it is not the equivalent of any value you could enter in the raw field.

  What happens if you enter an "invalid" value in the absolute field? This is not really a problem because the camera will automatically change the value you entered to the nearest valid value. For example, if you entered 13.69 in the raw field, the camera would automatically change the value to 13.68.

  Because the camera automatically changes any entry in an absolute field to the nearest "valid" value, you do not need to worry about entering equivalent values. However, you do need to be aware of the camera making small adjustments (unless you entered a „valid" value).

  Accordingly, if you read back a value in an absolute field, you may find that it is slightly different from the value that you entered the field.

4.2.2.2 Feature Control and Status Register Details

Video Data Output Mode CSR

Register Base Address: 0x1700			
Field Name: Register Status	**Offset:** 0x0000	**Size:** 1 Byte	**Type:** Read only
Description: The integer value in this field indicates the status of this control register: 0x00 = The register is not available 0x01 = The register is available. All related settings are OK 0x80 = A value in this register is set out of range			
Field Name: Mode	**Offset:** 0x0001	**Size:** 1 Byte	**Type:** Read / Write
Description: Writing an integer value to this field sets the video data output mode: 0x01 = 2 tap 8 bit output (A402k) 0x03 = 2 tap 10 bit output (A402k) 0x10 = 4 tap 8 bit output (A403k and A404k) 0x12 = 4 tap 10 bit output (A403k and A404k) 0x11 = 8 tap 8 bit output (A404k) See Sections 2.5.5 (A402k), 2.5.6 (A403k) and 2.5.7 (A404k) for descriptions of the video data output modes.			

Exposure Time Control Mode CSR

Register Base Address: 0x1400			
Field Name: Status	**Offset:** 0x0000	**Size:** 1 Byte	**Type:** Read only
Description: The integer value in this field indicates the status of this control register: 0x00 = The register is not available 0x01 = The register is available. All related settings are OK 0x80 = A value in this register is set out of range			
Field Name: Mode	**Offset:** 0x0001	**Size:** 1 Byte	**Type:** Read / Write
Description: Writing an integer value to this field sets the exposure time control mode: 0x00 = Free-run programmable 0x02 = Free-run edge-controlled 0x04 = ExSync level-controlled 0x05 = ExSync programmable 0x06 = ExSync edge-controlled See Section 3.3 for descriptions of the exposure time control modes.			

Exposure Time CSR

Note: The exposure time can be set by writing a floating point value to the Absolute Exposure Time field or by writing an integer value to the Raw Exposure Time field. Refer to Section 4.2.2.1 for an explanation of the difference between these two fields.

Register Base Address: 0x1500			
Field Name: Register Status	**Offset:** 0x0000	**Size:** 1 Byte	**Type:** Read only
Description: The integer value in this field indicates the status of this control register: 0x00 = The register is not available 0x01 = The register is available. All related settings are OK 0x80 = A value in this register is set out of range			
Field Name: Absolute Exposure Time	**Offset:** 0x0001	**Size:** 4 Bytes	**Type:** Read / Write
Description: Writing a floating point value to this field sets the exposure time in µs. For example, if the value in this field is set to 9.12, the exposure time will be 9.12 µs. The exposure time can normally range from 4.56 µs to 19119997.00 µs in increments of 0.01. The actual available range may be limited by the way any related parameters are set. The value in this field is a standard IEEE-754 single precision (32 bits) floating point number. See Section 3.3 for more information about exposure time.			
Field Name: Absolute Min	**Offset:** 0x0005	**Size:** 4 Bytes	**Type:** Read only
Description: Minimum allowed floating point value for the absolute exposure time setting. This field is updated to reflect limitations caused by the way any related features are set. The value in this field is a standard IEEE-754 single precision (32 bits) floating point number.			
Field Name: Absolute Max	**Offset:** 0x0009	**Size:** 4 Bytes	**Type:** Read only
Description: Maximum allowed floating point value for the absolute exposure time setting. This field is updated to reflect any limitations caused by the way any related features are set. The value in this field is a standard IEEE-754 single precision (32 bits) floating point number.			
Field Name: Raw Exposure Time	**Offset:** 0x000D	**Size:** 4 Bytes	**Type:** Read / Write
Description: Writing an integer value to this field sets the exposure time. The value can normally range from 1 (0x00000001) to 4194301 (0x003FFFFD). The actual available range may be limited by the way any related parameters are set. The integer value represents a multiplier and the actual exposure time is equal to the value in this field x 4.56 µs. For example, if the value in this field is set to 3 (0x00000003), then: Exposure Time = 3 x 4.56 µs Exposure Time = 13.68 µs The 4 bytes in this field are interpreted as follows: Byte 1 = Low byte of the raw value Byte 2 = Mid byte of the raw value Byte 3 = High byte of the raw value Byte 4 = Always 0x00 (not used) See Section 3.3 for more information about exposure time.			

Field Name: Raw Min	**Offset:** 0x0011	**Size:** 4 Bytes	**Type:** Read only
Description:	Minimum allowed integer value for the raw exposure time setting. This field is updated to reflect limitations caused by the way any related features are set. The 4 bytes in this field are interpreted as follows: Byte 1 = Low byte of the min value Byte 2 = Mid byte of the min value Byte 3 = High byte of the min value Byte 4 = Always 0x00 (not used)		
Field Name: Raw Max	**Offset:** 0x0015	**Size:** 4 Bytes	**Type:** Read only
Description:	Maximum allowed integer value for the raw exposure time setting. This field is updated to reflect limitations caused by the way any related features are set. The 4 bytes in this field are interpreted as follows: Byte 1 = Low byte of the max value Byte 2 = Mid byte of the max value Byte 3 = High byte of the max value Byte 4 = Always 0x00 (not used)		

Frame Period CSR

Note: The frame period can be set by writing a floating point value to the Absolute Frame Period field or by writing an integer value to the Raw Frame Period field. Refer to Section 4.2.2.1 for an explanation of the difference between these two fields.

Register Base Address: 0x1680			
Field Name: Register Status	**Offset:** 0x0000	**Size:** 1 Byte	**Type:** Read only
Description: The integer value in this field indicates the status of this control register: 0x00 = The register is not available 0x01 = The register is available. All related settings are OK 0x80 = A value in this register is set out of range			
Field Name: Absolute Frame Period	**Offset:** 0x0001	**Size:** 4 Bytes	**Type:** Read / Write
Description: Writing a floating point value to this field sets the frame period in µs. For example, if the value in this field is set to 71.1, the frame period would be 71.1 µs. The range of possible settings depends on the area of interest (AOI). See formulas 1 and 2 in Section 3.8.3. At the full AOI width (2352 columns) and the minimum AOI height (2 lines), the possible range of frame period settings is from 71.04 µs for an A402k, 35.76 µs for an A403k, 35.76 µs for an A404k set for 4 tap output or 18.24 µs for an A404k set for 8 tap output to 19.12 s. At the full AOI width (2352 columns) and the maximum AOI height (1726 lines), the possible range of frame period settings is from 40.90 ms for an A402k, 20.59 ms for an A403k, 20.59 ms for an A404k set for 4 tap output or 10.40 ms for an A404k set for 8 tap output to 19.12 s in increments of 0.01. The value in this field is a standard IEEE-754 single precision (32 bits) floating point number. See Section 3.3 for more information about the frame period.			
Field Name: Absolute Min	**Offset:** 0x0005	**Size:** 4 Bytes	**Type:** Read only
Description: Minimum allowed floating point value for the absolute frame period setting. The value in this field is a standard IEEE-754 single precision (32 bits) floating point number.			
Field Name: Absolute Max	**Offset:** 0x0009	**Size:** 4 Bytes	**Type:** Read only
Description: Maximum allowed floating point value for the absolute frame period setting. The value in this field is a standard IEEE-754 single precision (32 bits) floating point number.			
Field Name: Raw Frame Period	**Offset:** 0x000D	**Size:** 4 Bytes	**Type:** Read / Write
Description: Writing an integer value to this field sets the frame period. The value can range from 1 (0x00000001) to 4194303 (0x003FFFFF). The integer value represents a multiplier and the actual frame period is equal to the value in this field x 4.56 µs. For example, if the value in this field is set to 4 (0x00000004), then: Frame Period = 4 x 4.56 µs Frame Period = 18.24 µs The 4 bytes in this field are interpreted as follows: Byte 1 = Low byte of the raw value Byte 2 = Mid byte of the raw value Byte 3 = High byte of the raw value Byte 4 = Always 0x00 (not used) See Section 3.3 for more information about the frame period.			

Field Name: Raw Min	**Offset:** 0x0011	**Size:** 4 Bytes	**Type:** Read only
Description: Minimum allowed integer value for the raw frame period setting. The 4 bytes in this field are interpreted as follows: Byte 1 = Low byte of the min value Byte 2 = Mid byte of the min value Byte 3 = High byte of the min value Byte 4 = Always 0x00 (not used)			
Field Name: Raw Max	**Offset:** 0x0015	**Size:** 4 Bytes	**Type:** Read only
Description: Maximum allowed integer value for the raw frame period setting. The 4 bytes in this field are interpreted as follows: Byte 1 = Low byte of the max value Byte 2 = Mid byte of the max value Byte 3 = High byte of the max value Byte 4 = Always 0x00 (not used)			

Gain CSR

Note: The gain can be set by writing a floating point value to the Absolute Gain field or by writing an integer value to the Raw Gain field. Refer to Section 4.2.2.1 for an explanation of the difference between these two fields.

Register Base Address: 0x0E00			
Field Name: Status	**Offset:** 0x0000	**Size:** 1 Byte	**Type:** Read only
Description: The integer value in this field indicates the status of this control register: 0x00 = The register is not available 0x01 = The register is available. All related settings are OK 0x80 = A value in this register is set out of range			
Field Name: Absolute Gain	**Offset:** 0x0001	**Size:** 4 Bytes	**Type:** Read / Write
Description: Writing a floating point value to this field sets the gain in %. For example, if the value in this field is set to 10, the gain would be 10%. The gain can normally range from 0% to 100% in increments of 1. The actual available range may be limited by the way any related parameters are set. The value in this field is a standard IEEE-754 single precision (32 bits) floating point number. See Section 3.5 for more information about gain.			
Field Name: Absolute Min	**Offset:** 0x0005	**Size:** 4 Bytes	**Type:** Read only
Description: Minimum allowed floating point value for the absolute gain setting. This field is updated to reflect limitations caused by the way any related features are set. The value in this field is a standard IEEE-754 single precision (32 bits) floating point number.			
Field Name: Absolute Max	**Offset:** 0x0009	**Size:** 4 Bytes	**Type:** Read only
Description: Maximum allowed floating point value for the absolute gain setting. This field is updated to reflect any limitations caused by the way any related features are set. The value in this field is a standard IEEE-754 single precision (32 bits) floating point number.			
Field Name: Raw Gain	**Offset:** 0x0000D	**Size:** 2 Bytes	**Type:** Read / Write
Description: Writing an integer value to this field sets the gain. The value can normally range from 0% (0x00) to 100% (0x64) in increments of 1. The actual available range may be limited by the way any related parameters are set. The 2 bytes in this field are interpreted as 16 bit signed integer: Byte 1 = Low byte of the raw value Byte 2 = High byte of the raw value See Section 3.5 for more information about gain.			
Field Name: Raw Min	**Offset:** 0x000E	**Size:** 2 Bytes	**Type:** Read only
Description: Minimum allowed integer value for the raw gain setting. This field is updated to reflect limitations caused by the way any related features are set. The 2 bytes in this field are interpreted as 16 bit signed integer: Byte 1 = Low byte of the minimum raw value Byte 2 = High byte of the minimum raw value			

Field Name: Raw Max	**Offset:** 0x000F	**Size:** 2 Bytes	**Type:** Read only

Description: Maximum allowed integer value for the raw gain setting. This field is updated to reflect limitations caused by the way any related features are set.

The 2 bytes in this field are interpreted as 16 bit signed integer:

Byte 1 = Low byte of the maximum raw value

Byte 2 = High byte of the maximum raw value

Offset CSR

Note: The offset can be set by writing a floating point value to the Absolute Offset field or by writing an integer value to the Raw Offset field. Refer to Section 4.2.2.1 for an explanation of the difference between these two fields.

Register Base Address: 0x0F00			
Field Name: Register Status	**Offset:** 0x0000	**Size:** 1 Byte	**Type:** Read only
Description: The integer value in this field indicates the status of this control register: 0x00 = The register is not available 0x01 = The register is available. All related settings are OK 0x80 = A value in this register is set out of range			
Field Name: Absolute Offset	**Offset:** 0x0001	**Size:** 4 Bytes	**Type:** Read / Write
Description: Writing a floating point value to this field sets the offset in %. For example, if the value in this field is set to 20, the offset would be 20%. The offset can normally range from 0% to 100% in increments of 1. The actual available range may be limited by the way any related parameters are set. The value in this field is a standard IEEE-754 single precision (32 bits) floating point number. See Section 3.5 for more information about offset.			
Field Name: Absolute Min	**Offset:** 0x0005	**Size:** 4 Bytes	**Type:** Read only
Description: Minimum allowed floating point value for the absolute offset setting. This field is updated to reflect limitations caused by the way any related features are set. The value in this field is a standard IEEE-754 single precision (32 bits) floating point number.			
Field Name: Absolute Max	**Offset:** 0x0009	**Size:** 4 Bytes	**Type:** Read only
Description: Maximum allowed floating point value for the absolute offset setting. This field is updated to reflect any limitations caused by the way any related features are set. The value in this field is a standard IEEE-754 single precision (32 bits) floating point number.			
Field Name: Raw Offset	**Offset:** 0x000D	**Size:** 2 Bytes	**Type:** Read / Write
Description: Writing an integer value to this field sets the offset. The value can normally range from 0% (0x00) to 100% (0x64) in increments of 1. The actual available range may be limited by the way any related parameters are set.. The 2 bytes in this field are interpreted as 16 bit signed integer: Byte 1 = Low byte of the raw value Byte 2 = High byte of the raw value See Section 3.5 for more information about offset.			
Field Name: Raw Min	**Offset:** 0x000F	**Size:** 2 Bytes	**Type:** Read only
Description: Minimum allowed integer value for the raw offset setting. This field is updated to reflect limitations caused by the way any related features are set. The 2 bytes in this field are interpreted as 16 bit signed integer: Byte 1 = Low byte of the minimum raw value Byte 2 = High byte of the minimum raw value			

Field Name: Raw Max	**Offset:** 0x000F	**Size:** 2 Bytes	**Type:** Read only
Description:	Maximum allowed integer value for the raw offset setting. This field is updated to reflect limitations caused by the way any related features are set. The 2 bytes in this field are interpreted as 16 bit signed integer: Byte 1 = Low byte of the maximum raw value Byte 2 = High byte of the maximum raw value		

Column FPN Shading Correction CSR

Register Base Address: 0x2180			
Field Name: Register Status	**Offset:** 0x0000	**Size:** 1 Byte	**Type:** Read only
Description:	The integer value in this field indicates the status of this control register: 0x00 = The register is not available 0x01 = The register is available. All related settings are OK 0x80 = A value in this register is set out of range 0x82 = The generation process failed		
Field Name: Generate	**Offset:** 0x0001	**Size:** 1 Byte	**Type:** Read / Write
Description:	Writing an integer value to this field will either begin the routine that makes the camera generate a set of column FPN shading correction values or make the ADCs in the sensor self-calibrate so the FPN shading correction values are reset to their original values: 0x00 = Do nothing 0x01 = Generate and use column FPN correction values 0x02 = Reset column FPN correction values See Section 3.6 for more information about shading correction.		

DSNU or PRNU Shading Value Generate CSR

Register Base Address: 0x2100			
Field Name: Register Status	**Offset:** 0x0000	**Size:** 1 Byte	**Type:** Read only
Description: The integer value in this field indicates the status of this control register: 0x00 = The register is not available 0x01 = The register is available. All related settings are OK 0x80 = A value in this register is set out of range 0x82 = The generation process failed			
Field Name: Generate	**Offset:** 0x0001	**Size:** 4 Byte	**Type:** Read / Write
Description: Writing an integer value to this field will begin the routine that generates a set of DSNU or PRNU shading correction values: 0x01 = Generate DSNU correction values 0x02 = Generate PRNU correction values The 4 bytes in this field are interpreted as follows: Byte 1 = Low byte Byte 2 = Always 0x00 (not used) Byte 3 = Always 0x00 (not used) Byte 4 = Always 0x00 (not used) See Section 3.6 for more information about shading correction.			

DSNU and/or PRNU Shading Correction Enable CSR

Register Base Address: 0x2000			
Field Name: Register Status	**Offset:** 0x0000	**Size:** 1 Byte	**Type:** Read only
Description: The integer value in this field indicates the status of this control register: 0x00 = The register is not available 0x01 = The register is available. All related settings are OK 0x80 = A value in this register is set out of range			
Field Name: Mode	**Offset:** 0x0001	**Size:** 1 Byte	**Type:** Read / Write
Description: Writing an integer value to this field sets the shading correction mode: 0x00 = Shading correction off 0x01 = Enable DSNU shading correction only 0x02 = Enable PRNU shading correction only 0x03 = Enable DSNU and PRNU shading correction See Section 3.6 for more information about shading correction.			

Digital Shift CSR

Register Base Address: 0x1900			
Field Name: Register Status	**Offset:** 0x0000	**Size:** 1 Byte	**Type:** Read only
Description: The integer value in this field indicates the status of this control register: 0x00 = The register is not available 0x01 = The register is available. All related settings are OK 0x80 = A value in this register is set out of range			
Field Name: Mode	**Offset:** 0x0001	**Size:** 1 Byte	**Type:** Read / Write
Description: Writing an integer value to this field enables/disables digital shift: 0x00 = No digital shift 0x01 = Digital shift once (multiplies output 2X) 0x02 = Digital shift twice (multiplies output 4X) See Section 3.7 for more information about digital shift and precautions you must consider.			

Area of Interest Starting Column CSR

Register Base Address: 0x1040			
Field Name: Status	**Offset:** 0x0000	**Size:** 1 Byte	**Type:** Read only
Description: The integer value in this field indicates the status of this control register: 0x00 = The register is not available 0x01 = The register is available. All related settings are OK 0x80 = A value in this register is set out of range 0X81 = The setting for the AOI starting column conflicts with the setting for the AOI width			
Field Name: Starting Column	**Offset:** 0x0001	**Size:** 2 Bytes	**Type:** Read / Write
Description: Writing an integer value to this field sets the starting column for the area of interest (AOI) feature. The value for the starting column can normally range from column 1 (0x0000) to column 2336 (0x0920). Starting columns can only be selected in increments of 16 (+1), that is, the starting column can be 1, 17, 33, and so on. The actual available range may be limited by the way any related parameters are set. If the value is set to 1, the starting column in the AOI will be column 1. If the value is set to 17 the starting column in the AOI will be column 17. Etc. The 2 bytes in this field are interpreted as follows: Byte 1 = Low byte of the starting column value Byte 2 = High byte of the starting column value See Section 3.8 for more information about the AOI feature.			
Field Name: Min	**Offset:** 0x0003	**Size:** 2 Bytes	**Type:** Read only
Description: Minimum allowed integer value for the starting column setting. This field is updated to reflect limitations caused by the way any related features are set. The 2 bytes in this field are interpreted as follows: Byte 1 = Low byte of the min value Byte 2 = High byte of the min value			

Field Name: Max	**Offset:** 0x0005	**Size:** 2 Bytes	**Type:** Read only
Description:	Maximum allowed integer value for the starting column setting. This field is updated to reflect limitations caused by the way any related features are set. The 2 bytes in this field are interpreted as follows: Byte 1 = Low byte of the max value Byte 2 = Mid byte of the max value		
Field Name: Increment	**Offset:** 0x0007	**Size:** 2 Bytes	**Type:** Read only
Description:	An integer value indicating the increment for the starting column setting. The 2 bytes in this field are interpreted as follows: Byte 1 = Low byte of the max value Byte 2 = Mid byte of the max value		

Area of Interest Width in Columns CSR

Register Base Address: 0x1020			
Field Name: Status	**Offset:** 0x0000	**Size:** 1 Byte	**Type:** Read only
Description: The integer value in this field indicates the status of this control register: 0x00 = The register is not available 0x01 = The register is available. All related settings are OK 0x80 = A value in this register is set out of range 0X81 = The setting for the AOI starting column conflicts with the setting for the AOI width			
Field Name: Width	**Offset:** 0x0001	**Size:** 2 Bytes	**Type:** Read / Write
Description: Writing an integer value to this field sets the width in columns for the area of interest (AOI) feature. The value for the width in columns can normally range from 16 (0x0010) to 2352 (0x0930). The width can only be selected in increments of 16, that is, the width can be 16, 32, 48, and so on. The actual available range may be limited by the way any related parameters are set. If the value is set to 16, the length of the AOI will be 16 columns. If the value is set to 32, the width of the AOI will be 32 columns. Etc. The 2 bytes in this field are interpreted as follows: Byte 1 = Low byte of the length value Byte 2 = High byte of the length value See Section 3.8 for more information about the AOI feature.			
Field Name: Min	**Offset:** 0x0003	**Size:** 2 Bytes	**Type:** Read only
Description: Minimum allowed integer value for the width setting. This field is updated to reflect limitations caused by the way any related features are set. The 2 bytes in this field are interpreted as follows: Byte 1 = Low byte of the min value Byte 2 = High byte of the min value			
Field Name: Max	**Offset:** 0x0005	**Size:** 2 Bytes	**Type:** Read only
Description: Maximum allowed integer value for the width setting. This field is updated to reflect limitations caused by the way any related features are set. The 2 bytes in this field are interpreted as follows: Byte 1 = Low byte of the max value Byte 2 = Mid byte of the max value			
Field Name: Increment	**Offset:** 0x0007	**Size:** 2 Bytes	**Type:** Read only
Description: An integer value indicating the increment for the width setting. The 2 bytes in this field are interpreted as follows: Byte 1 = Low byte of the max value Byte 2 = Mid byte of the max value			

Area of Interest Starting Line CSR

Register Base Address: 0x1050
Field Name: Register Status **Offset:** 0x0000 **Size:** 1 Byte **Type:** Read only **Description:** The integer value in this field indicates the status of this control register: 0x00 = The register is not available 0x01 = The register is available. All related settings are OK 0x80 = A value in this register is set out of range 0X81 = The setting for the AOI starting line conflicts with the setting for the AOI height
Field Name: Starting Line **Offset:** 0x0001 **Size:** 2 Bytes **Type:** Read / Write **Description:** Writing an integer value to this field sets the starting line for the area of interest (AOI) feature. The value for the starting line can normally range from line 1 (0x0001) to line 1725 (0x06BD). The actual available range may be limited by the way any related parameters are set. If the value is set to 1, the starting line of the AOI will be line 1. If the value is set to 2, the starting line of the AOI will be line 2. Etc. The 2 bytes in this field are interpreted as follows: Byte 1 = Low byte of the starting line value Byte 2 = High byte of the starting line value See Section 3.8 for more information about the AOI feature.
Field Name: Min **Offset:** 0x0003 **Size:** 2 Bytes **Type:** Read only **Description:** Minimum allowed integer value for the starting line setting. This field is updated to reflect limitations caused by the way any related features are set. The 2 bytes in this field are interpreted as follows: Byte 1 = Low byte of the min value Byte 2 = High byte of the min value
Field Name: Max **Offset:** 0x0005 **Size:** 2 Bytes **Type:** Read only **Description:** Maximum allowed integer value for the starting line setting. This field is updated to reflect limitations caused by the way any related features are set. The 2 bytes in this field are interpreted as follows: Byte 1 = Low byte of the max value Byte 2 = Mid byte of the max value
Field Name: Increment **Offset:** 0x0007 **Size:** 2 Bytes **Type:** Read only **Description:** An integer value indicating the increment for the starting line setting. The 2 bytes in this field are interpreted as follows: Byte 1 = Low byte of the max value Byte 2 = Mid byte of the max value

Area of Interest Height in Lines CSR

Register Base Address: 0x1030
Field Name: Register Status **Offset:** 0x0000 **Size:** 1 Byte **Type:** Read only **Description:** The integer value in this field indicates the status of this control register: 0x00 = The register is not available 0x01 = The register is available. All related settings are OK 0x80 = A value in this register is set out of range 0X81 = The setting for the AOI starting line conflicts with the setting for the AOI height
Field Name: Height **Offset:** 0x0001 **Size:** 2 Bytes **Type:** Read / Write **Description:** Writing an integer value to this field sets the height in lines for the area of interest (AOI) feature. The value for the height in lines can normally range from 2 lines (0x0002) to 1726 lines (0x06BE) The actual available range may be limited by the way any related parameters are set. If the value is set to 20, the height of the AOI will be 20 lines. If the value is set to 48, the height of the AOI will be 48 lines. Etc. The 2 bytes in this field are interpreted as follows: Byte 1 = Low byte of the length value Byte 2 = High byte of the length value See Section 3.8 for more information about the AOI feature.
Field Name: Min **Offset:** 0x0003 **Size:** 2 Bytes **Type:** Read only **Description:** Minimum allowed integer value for the height setting. This field is updated to reflect limitations caused by the way any related features are set. The 2 bytes in this field are interpreted as follows: Byte 1 = Low byte of the min value Byte 2 = High byte of the min value
Field Name: Max **Offset:** 0x0005 **Size:** 2 Bytes **Type:** Read only **Description:** Maximum allowed integer value for the height setting. This field is updated to reflect limitations caused by the way any related features are set. The 2 bytes in this field are interpreted as follows: Byte 1 = Low byte of the max value Byte 2 = Mid byte of the max value
Field Name: Increment **Offset:** 0x0007 **Size:** 2 Bytes **Type:** Read only **Description:** An integer value indicating the increment for the height setting. The 2 bytes in this field are interpreted as follows: Byte 1 = Low byte of the max value Byte 2 = Mid byte of the max value

AOI List Trigger Mode CSR

Register Base Address: 0x2E00			
Field Name: Register Status	**Offset:** 0x0000	**Size:** 1 Byte	**Type:** Read only
Description: The integer value in this field indicates the status of this control register: 0x00 = The register is not available 0x01 = The register is available. All related settings are OK 0x80 = A value in this register is set out of range			
Field Name: Mode	**Offset:** 0x0001	**Size:** 1 Byte	**Type:** Read / Write
Description: Writing an integer value to this field sets the programmable AOI sequencer mode: 0x00 = Programmable AOI sequencer off 0x01 = Image per trigger 0x02 = List per trigger 0x03 = Free-run See Section 3.8.4 for more information about the programmable AOI sequencer feature.			

Stamp CSR

Register Base Address: 0x2400			
Field Name: Register Status	**Offset:** 0x0000	**Size:** 1 Byte	**Type:** Read only
Description: The integer value in this field indicates the status of this control register: 0x00 = The register is not available 0x01 = The register is available. All related settings are OK 0x80 = A value in this register is set out of range			
Field Name: Mode	**Offset:** 0x0001	**Size:** 1 Byte	**Type:** Read / Write
Description: Writing an integer value to this field sets the stamp mode: 0x00 = Stamp off 0x01 = Stamp enabled See Section 3.9 for more information about the stamp feature.			

Flash Trigger Output Mode CSR

Register Base Address: 0x1D00			
Field Name: Register Status	**Offset:** 0x0000	**Size:** 1 Byte	**Type:** Read only
Description: The integer value in this field indicates the status of this control register: 0x00 = The register is not available 0x01 = The register is available. All related settings are OK 0x80 = A value in this register is set out of range			
Field Name: Operating Mode	**Offset:** 0x0001	**Size:** 1 Byte	**Type:** Read / Write
Description: Writing an integer value to this field sets the mode of the flash trigger output signal: 0x00 = The flash trigger signal is always low. 0x01 = The flash trigger signal is high while the sensor's flash window is open. 0x02 = The flash trigger signal is high while the ExFlash signal from the frame grabber is high. 0x03 = The flash trigger signal is always high. 0x05 = The flash trigger signal is low while sensor's the flash window is open. 0x06 = The flash trigger signal is low while the ExFlash signal from the frame grabber is high. See Section 2.5.8 for more information about the flash trigger signal.			

Flash Trigger Switching Mode CSR

Register Base Address: 0x1E00			
Field Name: Register Status	**Offset:** 0x0000	**Size:** 1 Byte	**Type:** Read only
Description: The integer value in this field indicates the status of this control register: 0x00 = The register is not available 0x01 = The register is available. All related settings are OK 0x80 = A value in this register is set out of range			
Field Name: Switching Mode	**Offset:** 0x0001	**Size:** 1 Byte	**Type:** Read / Write
Description: Writing an integer value to this field sets the switching of the flash trigger output signal: 0x00 = TTL 0x01 = Open collector or Low Side Switch, 5 V max 0x02 = High Side Switch 5 V 0x03 = High impedance (default) See Section 2.5.8 for more information about the flash trigger signal.			

Mirror Image Mode CSR

Register Base Address: 0x3500			
Field Name: Register Status	**Offset:** 0x0000	**Size:** 1 Byte	**Type:** Read only
Description: The integer value in this field indicates the status of this control register: 0x00 = The register is not available 0x01 = The register is available. All related settings are OK 0x80 = A value in this register is set out of range			
Field Name: Mirror Mode	**Offset:** 0x0001	**Size:** 1 Byte	**Type:** Read / Write
Description: Writing an integer value to this field sets the mirror image mode: 0x00 = Mirror image disabled 0x01 = Mirror image enabled See Section 3.10 for more information about the mirror image feature.			

Test Image Mode CSR

Register Base Address: 0x1800			
Field Name: Register Status	**Offset:** 0x0000	**Size:** 1 Byte	**Type:** Read only
Description: The integer value in this field indicates the status of this control register: 0x00 = The register is not available 0x01 = The register is available. All related settings are OK 0x80 = A value in this register is set out of range			
Field Name: Mode	**Offset:** 0x0001	**Size:** 1 Byte	**Type:** Read / Write
Description: Writing an integer value to this field sets the test image mode: 0x00 = No test image 0x01 = Test image one enabled (vertical stripe pattern) 0x02 = Test image two enabled (still diagonal stripe pattern) 0x03 = Test image three enabled (moving diagonal stripe pattern) 0x04 = Test image four enabled (horizontal stripe pattern) See Section 3.12 for more information about test images.			

Serial Communication CSR

An RS-644 serial connection is integrated into the Camera Link interface between the frame grabber installed in your computer and the camera. This serial connection is used to issue commands to the camera for changing modes and parameters. You can use this CSR to set the bitrate for the camera's RS-644 serial port.

The default setting is 9600 bps.

The setting is changed immediately after the successful receipt of this command.

Register Base Address: 0x0D00

Field Name: Register Status **Offset:** 0x0000 **Size:** 1 Byte **Type:** Read only

Description: The integer value in this field indicates the status of this control register:

0x00 = The register is not available

0x01 = The register is available. All related settings are OK

0x80 = A value in this register is set out of range

Field Name: Bitrate **Offset:** 0x0001 **Size:** 1 Byte **Type:** Read / Write

Description: Writing an integer value to this field sets the bit rate:

0x01 = 50 bps	0x0C = 3600 bps
0x02 = 75 bps	0x0D = 4800 bps
0x03 = 110 bps	0x0E = 7200 bps
0x04 = 134.5 bps	0x0F = 9600 bps (default)
0x05 = 150 bps	0x10 = 14400 bps
0x06 = 200 bps	0x11 = 19200 bps
0x07 = 300 bps	0x12 = 38400 bps
0x08 = 600 bps	0x13 = 57600 bps
0x09 = 1200 bps	0x14 = 115200 bps
0x0A = 1800 bps	
0x0B = 2400 bps	

When changing a setting for serial communication, use the following procedure:

1. Issue the write command with the new setting.
2. Wait one second.
3. Change the setting on the serial port that the camera is using:
   a) If you are using a Camera Link frame grabber, change the setting on the frame grabber's RS-644 serial port.
   b) If you are using the camera with a k-BIC, change the bit rate on your PC's RS-232 serial port (A402k only).
4. Resume communication.

The RS-644 serial port on some Camera Link frame grabbers will only support a bitrate of 9600. If you are using a Camera Link frame grabber, check the grabber's documentation before attempting to change the bitrate.

At reset or power off/on, the camera returns to the 9600 bps setting.

Camera Reset CSR

Register Base Address: 0x0B00			
Field Name: Register Status	**Offset:** 0x0000	**Size:** 1 Byte	**Type:** Read only
Description: The integer value in this field indicates the status of this control register: 0x00 = The register is not available 0x01 = The register is available. All related settings are OK 0x80 = A value in this register is set out of range			
Field Name: Reset	**Offset:** 0x0000	**Size:** 1 Byte	**Type:** Write only
Description: Writing an integer value of 1 (0x01) to this field will initiate a camera reset. The reset behavior is similar to a power up reset.			

Parameter Set Cache CSR

Register Base Address: 0x3000			
Field Name: Register Status	**Offset:** 0x0000	**Size:** 1 Byte	**Type:** Read only
Description: The integer value in this field indicates the status of this control register: 0x00 = The register is not available 0x01 = The register is available. All related settings are OK 0x80 = A value in this register is set out of range			
Field Name: Mode	**Offset:** 0x0001	**Size:** 4 Bytes	**Type:** Read / Write
Description: Writing an integer value to this field enables/disables parameter set cache: 0x00 = Parameter set cache disabled 0x01 = Parameter set cache enabled See Section 3.15 for more information about parameter set cache.			

4.2.3 Bulk Data and the Bulk Data Control and Status Registers

The term “bulk data” refers to a collection of values used by the camera as a block. A configuration set (see Section 3.14) is an example of one type of bulk data. A single configuration set contains values for all of the normal parameters needed to configure the camera and the values within a configuration set are used by the camera as a block. On A400k cameras, a set shading values (see Section 3.6) is an example of another type of bulk data. A set of shading values contains all of the values needed to do column FPN, DSNU and PSNU shading correction. Another example of a type of bulk data is the programmable AOI list that is uploaded to the camera to perform a pre-defined sequence of areas of interest.

A400k cameras have a file system similar to the file system used on PCs. A400k cameras can store blocks of bulk data such as configuration sets, a set of shading values or an AOI list in named files within the camera’s non-volatile memory. The camera’s bulk data control and status registers (CSRs) are used to save blocks of bulk data to files in the non-volatile memory. For example, you can use the configuration set bulk data control register to create a named file in the camera and to store the settings from the current work configuration set in that file.

In the case of the A400k, there are three types of bulk data: configuration sets, a set of shading values and an AOI list. There is a separate bulk data control and status register for each type of bulk data. The configuration set bulk data CSR is used to work with configuration sets, the shading value CSR is used to work with a set of shading values and the AOI list CSR is used to upload an AOI list.

By writing to fields within a bulk data CSR you can do things such as saving a block of bulk data to a file in the non-volatile memory, copying a saved bulk data file from the camera to a PC, and creating a list of existing saved bulk data files. Section 4.2.3.2 lists the bulk data CSRs in A400k cameras and provides a general description of the use of each field within the registers.

The best way to understand the use of the bulk data CSRs is to read about the different tasks that you can perform with them. Section 4.2.3.1 describes the tasks that are normally performed by manipulating the bulk data CSRs and provides a procedure for performing each task.

4.2.3.1 Using a Bulk Data CSR to Work with Bulk Data

Saving a Configuration Set

As mentioned in Section 3.14, the work configuration set resides in the camera's volatile memory. Assume that you want to save the values in the current work set to a file named "UserSet01" in the camera's non-volatile memory. To do so, you would follow this procedure:

1. Use a binary write command to write the file name UserSet01 to the Name field of the configuration set bulk data CSR (see page 4-41).
2. Use a binary write command to set the value of the Control field in the configuration set bulk data CSR to 0x06. Setting the value to 0x06 initiates a create function.

This procedure would create a file called UserSet01 in the non-volatile memory and would copy the current work set settings from the camera's volatile memory into the new file.

Sample code that illustrates how to create a bulk data file is available from Basler (see Section 4.4).

You can save up to four configuration set files in the non-volatile memory.

There is a restriction on naming the files that hold saved configuration sets. They must be named UserSet01, UserSet02, UserSet03, or UserSet04.

If you use the name of an existing file, the data in the existing file will be overwritten.

Saving a Set of DSNU Shading Values

As mentioned in Section 3.6, when you generate a set of shading values, those values reside in the camera's volatile memory. To save the DSNU shading values currently in the volatile memory to a file in the camera's non-volatile memory, you would follow this procedure:

1. Use a binary write command to write the file name *offsetshading* to the Name field of the DNSU shading values bulk data CSR (see page 4-42).
2. Use a binary write command to set the value of the Control field in the DSNU shading values bulk data CSR to 0x06. Setting the value to 0x06 initiates a create function.

This procedure would create a file called offsetshading in the non-volatile memory and would copy the current DSNU shading values from the camera's volatile memory into the new file.

Sample code that illustrates how to create a bulk data file is available from Basler (see Section 4.4).

You can save one DSNU shading set file in the non-volatile memory.

There is a restriction on naming the file that holds the saved set of shading tables. The file must be named *offsetshading*.

If the offsetshading file already exists, it will be overwritten.

Saving a Set of PRNU Shading Values

As mentioned in Section 3.6, when you generate a set of shading values, those values reside in the camera's volatile memory. To save the PRNU shading values currently in the volatile memory to a file in the camera's non-volatile memory, you would follow this procedure:

1. Use a binary write command to write the file name *gainshading* to the Name field of the PRNU shading values bulk data CSR (see page 4-43).
2. Use a binary write command to set the value of the Control field in the PRNU shading values bulk data CSR to 0x06. Setting the value to 0x06 initiates a create function.

This procedure would create a file called gainshading in the non-volatile memory and would copy the current PRNU shading values from the camera's volatile memory into the new file.

Sample code that illustrates how to create a bulk data file is available from Basler (see Section 4.4).

You can save one PRNU shading set file in the non-volatile memory.

There is a restriction on naming the file that holds the saved set of shading tables. The file must be named *gainshading*.

If the gainshading file already exists, it will be overwritten.

Activating a Saved Configuration Set File

The process of "activating" an existing configuration set file, accomplishes two things:

- It copies the values from the saved file into the camera's volatile memory. This means that the values will now be actively used by the camera.
- It creates a link to the activated file. If the camera is reset or if it is powered off and then back on, the values from the activated file will be loaded into volatile memory of the camera and actively used by the camera.

As an example, assume that the camera already has a saved configuration set file named "UserSet01" and that you want to activate this file. To do so, you would follow this procedure:

1. Use a binary write command to write the file name UserSet01 to the Name field of the configuration set bulk data CSR (see page 4-41).
2. Use a binary write command to set the value of the Control field in the configuration set bulk data CSR to 0x05. (Setting the value to 0x05 initiates an activate function.)

This procedure would find the UserSet01 file in the non-volatile memory and would copy the values in the file into the camera's volatile memory. It would also create a link to the file so that the values in the file would be loaded into volatile memory after a reset or a power up.

Sample code that illustrates how to activate a bulk data file is available from Basler (see Section 4.4).

If you want to activate the factory configuration set file, use the procedure described above and use "FactorySet" as the file name.

Enumerating Saved Bulk Data Files

Bulk data file enumeration lets you look through a list of existing saved bulk data files.

As an example, assume that you want to see a list of all of the existing saved configuration set files. To do so, you would follow this procedure:

1. Use a binary write command to set the value of the Control field in the configuration set bulk data CSR to 0x00 (see page 4-41). Setting the value to 0x00 initiates an enumerate function.
2. Use a binary read command to read the Name field of the configuration set bulk data CSR.
3. Use a binary read command to read the value in the Info field of the configuration set bulk data CSR.
   a) If the value is 0x00, it means that the file exists and it is not an activated file. Continue to step 4.
   b) If the value is 0x04, it means that the file exists and it is an activated file. Continue to step 4.
   c) If the value is 0x01 no more saved configuration set files exist. Exit the procedure.
4. Use a binary write command to set the value of the Control field in the configuration set bulk data CSR to 0x01. Setting the value to 0x01 initiates an enumerate next function.
5. Return to step 2.

This procedure would list all of the configuration set bulk data files stored in the computer.

If you wanted to enumerate existing DSNU or PRNU shading table bulk data files stored in the camera, you would use a similar procedure but you would write to and read from the DSNU or PRNU shading table bulk data CSR instead.

If you wanted to enumerate the existing AOI list file stored in the camera, you would use a similar procedure but you would write to and read from the AOI list bulk data CSR instead.

Sample code that illustrates how to enumerate bulk data files is available from Basler (see Section 4.4).

Downloading a Saved Bulk Data File from the Camera to a PC

You can download an existing saved bulk data file from the camera's non-volatile memory to your host PC.

As an example, assume that the camera has an existing saved configuration set file named "UserSet02" and that you want to download this file from the camera to your host PC. To do so, you would follow this procedure:

1. Use a binary write command to write the file name UserSet02 to the Name field of the configuration set bulk data CSR.
2. Use a binary read command to read the Size field of the configuration set bulk data CSR. If the file exists, this field will tell you the file size. If the file does not exist, this field will be 0.
3. Use a binary write command to set the value of the Control field in the configuration set bulk data CSR to 0x02. Setting the value to 0x02 places the camera in read mode.
4. Use a binary command to do a bulk data read from the camera. The binary command must have the following characteristics:

   OpCode = 0x05 (This OpCode makes the binary command a bulk data read)

   DataLen = the number of bytes to be read (Max is 255 characters*)

   Address = 0x281B
   (Base address for the configuration set bulk data CSR plus the offset for the Data field)

   Data = none
5. You receive a bulk data response frame from the camera.
   (The amount of data in the response will be as specified by the DataLen in step 4.)
6. Use a binary read command to read the value in the Info field of the configuration set bulk data CSR.
   a) If the value is 0x01 no more data exists in the file. Exit the procedure.
   b) If the value is 0x00, more data exists and this is not an activated file. Return to step 4.
   c) If the value is 0x04, more data exists and this is an activated file. Return to step 4.

* Up to 255 characters can be read with a single binary bulk data read command. If the file is larger than 255 characters, repeated binary bulk data read commands are required. When repeated bulk data read commands are required, the file is read sequentially with each read command starting where the previous read stopped.

This procedure would download the data in the file to the host computer.

If you wanted to download an existing DSNU or PRNU shading value bulk data file to the host PC, you would use a similar procedure but you would write to and read from the DSNU or PRNU shading value bulk data CSR instead.

If you wanted to download an existing AOI list bulk data file to the host PC, you would use a similar procedure but you would write to and read from the AOI list bulk data CSR instead.

Sample code that illustrates how to download bulk data files is available from Basler (see Section 4.4).

Uploading a Bulk Data File from a PC to the Camera

You can upload a bulk data file from your host PC to the camera's non-volatile memory.

When you upload a bulk data file from the PC to the camera, it will overwrite any existing bulk data file in the camera that has the same name.

As an example, assume that you previously downloaded a configuration set bulk data file named "UserSet02" to your PC. Also assume that you now want to upload this file from your host PC to the camera. To do so, you would follow this procedure:

1. Use a binary write command to write the file name UserSet02 to the Name field of the configuration set bulk data CSR.
2. Use a binary write command to set the value of the Control field in the configuration set bulk data CSR to 0x03. Setting the value to 0x03 places the camera in write mode.
3. Use a binary command to do a bulk data write to the camera. The binary write command must have the following characteristics:

   OpCode = 0x04 (This OpCode makes the command a bulk data write)

   DataLen = the number of bytes to be read (Max is 255 characters*)

   Address = 0x281B
   (Base address for the configuration set bulk data CSR plus the offset for the Data field)

   Data = Bytes to be written
4. Repeat step 3 as many times as needed to write all of the data from the PC to the camera.
5. Close the bulk data file in the camera. To close the file, use a binary write command to set the value of the Control field in the configuration set bulk data CSR to 0x02. Setting the value to 0x02 places the camera in read mode and closes the file.

* Up to 255 characters can be read with a single binary bulk data read command. If the file is larger than 255 characters, repeated binary bulk data read commands are required. When repeated bulk data read commands are required, the file is read sequentially with each write command starting where the previous write stopped.

This procedure would upload the data in the file to the camera.

If you wanted to upload existing DSNU or PRNU shading value bulk data files stored in the camera, you would use a similar procedure but you would write to and read from the DSNU or PRNU shading table bulk data CSR instead.

If you wanted to upload existing AOI list bulk data files stored in the camera, you would use a similar procedure but you would write to and read from the AOI list bulk data CSR instead.

Sample code that illustrates how to download bulk data files is available from Basler (see Section 4.4).

4.2.3.2 Bulk Data Control and Status Register Details

Configuration Set CSR

See Section 4.2.3.1 for information about using bulk data control registers.

Register Base Address: 0x2800
Field Name: Register Status **Offset:** 0x0000 **Size:** 1 Byte **Type:** Read only **Description:** The integer value in this field indicates the status of this control register: 0x00 = The register is not available 0x01 = The register is available. All related settings are OK 0x80 = A value in this register is set out of range
Field Name: Control **Offset:** 0x0001 **Size:** 1 Byte **Type:** Read / Write **Description:** Writing an integer value to this field selects a bulk data control function: 0x00 = Enumerate 0x01 = ENext 0x02 = Read 0x03 = Write 0x05 = Activate 0x06 = Create
Field Name: Info **Offset:** 0x0002 **Size:** 1 Byte **Type:** Read only **Description:** If you are performing a bulk data read or write operation, this field will indicate when no more data exits: 0x00 = More data exists 0x01 = No more data exists 0x04 = No more data exists and this is an activated file
Field Name: File Name **Offset:** 0x0003 **Size:** 20 Bytes **Type:** Read / Write **Description:** Character string indicating the name of a bulk data file. The field contains 20 characters (left-aligned, zero-padded).
Field Name: Size **Offset:** 0x0017 **Size:** 4 Bytes **Type:** Read only **Description:** If the file name refers to an already existing bulk data file, this field will contain the file size in bytes. Otherwise, the field contains zeros.
Field Name: Data **Offset:** 0x001B **Size:** Byte **Type:** Read / Write **Description:** Byte-sized register that is used to sequentially write to or read from a bulk data file.

You can save up to four configuration set files in the non-volatile memory.

There is a restriction on naming the files that hold the saved configuration sets. They must be named UserSet01, UserSet02, UserSet03 or UserSet04.

DSNU Shading Value CSR

See Section 4.2.3.1 for information about using bulk data control registers.

Register Base Address: 0x2A80
Field Name: Register Status **Offset:** 0x0000 **Size:** 1 Byte **Type:** Read only **Description:** The integer value in this field indicates the status of this control register: 0x00 = The register is not available 0x01 = The register is available. All related settings are OK 0x80 = A value in this register is set out of range
Field Name: Control **Offset:** 0x0001 **Size:** 1 Byte **Type:** Read / Write **Description:** Writing an integer value to this field selects a bulk data control function: 0x00 = Enumerate 0x01 = ENext 0x02 = Read 0x03 = Write 0x05 = Activate 0x06 = Create
Field Name: Info **Offset:** 0x0002 **Size:** 1 Byte **Type:** Read only **Description:** If you are performing a bulk data read or write operation, this field will indicate when no more data exits: 0x00 = More data exists 0x01 = No more data exists 0x04 = No more data exists and this is an activated file
Field Name: File Name **Offset:** 0x003 **Size:** 20 Bytes **Type:** Read / Write **Description:** Character string indicating the name of a bulk data file. The field contains 20 characters (left-aligned, zero-padded).
Field Name: Size **Offset:** 0x0017 **Size:** 4 Bytes **Type:** Read only **Description:** If the file name refers to an already existing bulk data file, this field will contain the file size in bytes. Otherwise, the field contains zeros.
Field Name: Data **Offset:** 0x001B **Size:** 1 Byte **Type:** Read / Write **Description:** Byte-sized register that is used to sequentially write to or read from a bulk data file.

You can save up to one DSNU shading value file in the non-volatile memory.

There is a restriction on naming the file that holds the saved set of DSNU shading values. The file must be named offsetshading.

PRNU Shading Value CSR

See Section 4.2.3.1 for information about using bulk data control registers.

Register Base Address: 0x2A00
Field Name: Register Status **Offset:** 0x0000 **Size:** 1 Byte **Type:** Read only **Description:** The integer value in this field indicates the status of this control register: 0x00 = The register is not available 0x01 = The register is available. All related settings are OK 0x80 = A value in this register is set out of range
Field Name: Control **Offset:** 0x0001 **Size:** 1 Byte **Type:** Read / Write **Description:** Writing an integer value to this field selects a bulk data control function: 0x00 = Enumerate 0x01 = ENext 0x02 = Read 0x03 = Write 0x05 = Activate 0x06 = Create
Field Name: Info **Offset:** 0x0002 **Size:** 1 Byte **Type:** Read only **Description:** If you are performing a bulk data read or write operation, this field will indicate when no more data exits: 0x00 = More data exists 0x01 = No more data exists 0x04 = No more data exists and this is an activated file
Field Name: File Name **Offset:** 0x003 **Size:** 20 Bytes **Type:** Read / Write **Description:** Character string indicating the name of a bulk data file. The field contains 20 characters (left-aligned, zero-padded).
Field Name: Size **Offset:** 0x0017 **Size:** 4 Bytes **Type:** Read only **Description:** If the file name refers to an already existing bulk data file, this field will contain the file size in bytes. Otherwise, the field contains zeros.
Field Name: Data **Offset:** 0x001B **Size:** 1 Byte **Type:** Read / Write **Description:** Byte-sized register that is used to sequentially write to or read from a bulk data file.

You can save up to one PRNU shading value file in the non-volatile memory.

There is a restriction on naming the file that holds the saved set of PRNU shading values. The file must be named gainshading.

AOI List CSR

See Section 4.2.3.1 for information about using bulk data control registers.

Register Base Address: 0x2D00			
Field Name: Register Status	**Offset:** 0x0000	**Size:** 1 Byte	**Type:** Read only
Description: The integer value in this field indicates the status of this control register: 0x00 = The register is not available 0x01 = The register is available. All related settings are OK 0x80 = A value in this register is set out of range			
Field Name: Control	**Offset:** 0x0001	**Size:** 1 Byte	**Type:** Read / Write
Description: Writing an integer value to this field selects a bulk data control function: 0x00 = Enumerate 0x01 = ENext 0x02 = Read 0x03 = Write 0x05 = Activate 0x06 = Create			
Field Name: Info	**Offset:** 0x0002	**Size:** 1 Byte	**Type:** Read only
Description: If you are performing a bulk data read or write operation, this field will indicate when no more data exits: 0x00 = More data exists 0x01 = No more data exists 0x04 = No more data exists and this is an activated file			
Field Name: File Name	**Offset:** 0x003	**Size:** 20 Bytes	**Type:** Read / Write
Description: Character string indicating the name of a bulk data file. The field contains 20 characters (left-aligned, zero-padded).			
Field Name: Size	**Offset:** 0x0017	**Size:** 4 Bytes	**Type:** Read only
Description: If the file name refers to an already existing bulk data file, this field will contain the file size in bytes. Otherwise, the field contains zeros.			
Field Name: Data	**Offset:** 0x001B	**Size:** 1 Byte	**Type:** Read / Write
Description: Byte-sized register that is used to sequentially write to or read from a bulk data file.			

You can save up to one AOI list file in the non-volatile memory.

There is a restriction on naming the file that holds the saved AOI list. The file must be named aoilist.

4.3 Using Binary Read/Write Commands on the A400k

As explained in Section 4.2, each A400k camera has control and status registers with one or more fields that are used to set the values for parameters associated with a camera feature. For example, the stamp control register has several fields used to set the parameters associated with the stamp feature. By writing values to fields in the control registers, you configure the camera and control how it operates. By reading values from fields in the control registers, you can determine how the camera is currently configured.

Each camera also has inquiry registers with fields that contain basic information such as the camera's serial number and software version numbers. By reading values in the inquiry register fields, you can determine some basic information about the camera.

A "binary read/write command" protocol has been developed for use with A400k cameras.

You can read the data in a register field by sending a binary read command to the camera. For example, you can use a read command to determine the current value of the Enable field of the Test Image control and status register (see page 4-32). When you issue a read command to the camera, the camera responds by sending the requested data back to the host computer.

You can write data to a register field by sending a write command to the camera. For example, you can use a write command to change the value of the Enable field of the Test Image control register. When you issue a write command to the camera, the value in the register field will be replaced and the camera will send a write response back to the host computer.

Each field within a control register or an inquiry register has a specific memory address. When you issue a binary read or a binary write command, the address for field you want to work with is included as part of the command. Section 4.3.1 describes the binary read/write command format in detail. Section 4.4 provides code samples for a binary read and a binary write command.

Binary read/write commands are issued to the A400k via the RS-644 serial connection in the Camera Link interface between the frame grabber and the camera. A standard application programmer's interface (API) for asynchronous serial reading and writing via the RS-644 port on the frame grabber has been defined in the Camera Link standard (Appendix B, API Functions). All Camera Link compatible frame grabbers provide a software library (.dll file) named clser***.dll where *** is specific to the frame grabber vendor. There are four functions exported by that DLL:

- clSerialInit - Initialize the serial communication for a specific board.
- clSerialRead - Read bytes from the camera.
- clSerialWrite - Write bytes to the camera.
- clSerialClose - Close the serial communication.

To execute the binary programming commands, you can call up the functions exported by the DLL.

When the camera is powered on or when a camera reset is performed, your PC may receive some random characters on the serial interface. We recommend clearing the serial input buffers in your PC after a camera power on or reset.

If you are using your camera with an optional Basler Interface Converter (k-BIC), you can use binary commands to configure the camera via the RS-232 serial connection between your PC and the k-BIC.

4.3.1 The Binary Read/Write Command Protocol

With the binary read/write command protocol, data is placed into a “frame” and sent to the camera. When the frame is received, it is checked for validity. If valid, the data is extracted from the frame and the command is executed.

This section describes the basic layout of a binary command frame. Figure 4-2 shows a graphical representation of the fields within a binary command frame. The text below the graphic describes each field of the command frame in detail.

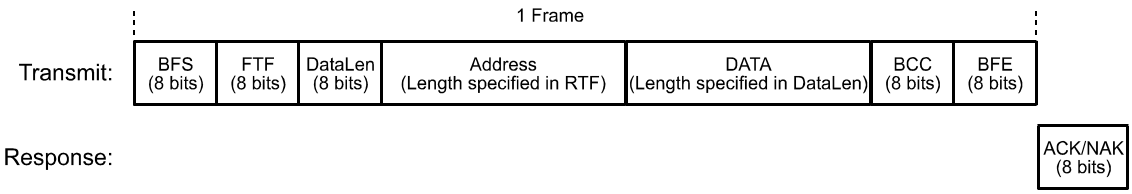


Figure 4-2: Representation of a Command Frame and Response

BFS Binary Frame Start field

Identifies the start of a binary frame.

Size = 1 byte

The value of the BFS byte is always 0x01.

FTF Frame Type and Format field

Identifies the frame type and format.

Size = 1 byte

The bits in the FTF field are assigned as follows:

7	6	5	4	3	2	1	0
OpCode					BCC Code	AddrLen	

The MSB of the FTF field is on the left (highest bit of the opcode) and the LSB of the field is on the right (lowest bit of the address length).

The value in the OpCode portion of the FTF field defines the function of the binary command, that is, whether it is a read command or a write command. The following OpCodes are available:

OpCode	Function
0b00000	This is a write command used to write a single setting to the camera.
0b00001	This is a read command used to read a single setting from the camera.
0b00010	This is a read response frame without an address field. (The AddrLen bits are 'don't care').
0b00100	This is a bulk write command used to upload a file into the camera.
0b00101	This is a bulk read command used to download a file from the camera.
0b00110	This is a bulk read response frame without an address field. (The AddrLen bits are 'don't care')

The BCC-Code portion of the FTF field indicates the presence of a Block Check Character (BCC). (the use of a BCC is optional.)

BCC-Code	Function
0b0	This frame (and also the response frame) contains no BCC field.
0b1	This frame (and also the response frame) contains a BCC field.

The AddrLen portion of the FTF field indicates the size of the command and status register (CSR) address to which the command is being sent.

AddrLen	Length of the Address Field
0b00	16 bits (= 2 bytes)
0b01	32 bits (= 4 bytes)
0b10	48 bits (= 6 bytes)
0b11	64 bits (= 8 bytes)

On A400k cameras, all CSR addresses are 16 bits.

Example of an FTF field:

Assume that you are issuing a write command, that you are using a BCC and that the CSR address you are writing to is a 16 bit address. In this case, the OpCode bits would be 0b00000, the BCC-Code bit would be 0b1 and the AddrLen bits would be 0b00. This would result in a binary value of 0b00000100, which translates to a hex value of 0x04 for the FTF field.

DataLen Data Length field

For read commands, the DataLen field indicates the number of bytes to read from the given CSR address.

For write commands, the DataLen field indicates the number of bytes contained in the Data field.

Size = 1 byte

Range of possible settings: 0 to 255.

DataLen = 0 will result in an ACK, but no further command will be executed.

Address Address field

For read commands, indicates the CSR address for the read.

For write commands, indicates the CSR address for the write.

Size = Number of bytes indicated in the AddrLen portion of the FTF field

Data Data field

For read commands, this field contains no data.

For wire commands, this field contains the data to be written to the CSR.

Size for read commands = 0 bytes.

Size for write commands = the number of bytes indicated in the DataLen field of the frame.

BCC Block Check Character field

The use of a block check character in read/write commands is optional.

If bit 2 of the FTF field is 0, the BCC is not used and the BCC field will contain no data. If bit 2 of the FTF field is 1, the BCC field will contain the block check character.

Size = 0 bytes if bit 2 of the FTF field is 0

1 byte if bit 2 of the FTF field is 1

The block check character is the exclusive-or sum (XOR sum) of the bytes in the FTF, DataLen, Address and Data fields (see section 4.3.2.3).

BFE Binary Frame End field

Identifies the end of a binary frame.

Size = 1 byte

The value of the BFE byte is always 0x03.

ACK/NAK Response Positive frame acknowledge/negative frame acknowledge

Size = 1 byte

The value for a positive frame acknowledgement (ACK) is 0x06 and for a negative frame acknowledgement (NAK) is 0x15.

Note

All values are formatted as little endian (Intel format).

4.3.1.1 Error Checking and Responses

ACK/NAK

When the camera receives a frame, it checks to see if the order of the bytes in the frame is correct. If the FTF field indicates that the frame includes a BCC, the camera checks to see if the XOR sum of the relevant frame fields matches the block check character. The camera also checks to see if the number of bytes in the data field is equal to the number specified in the DataLen field.

If all checks are correct, an ACK is sent to the host. If any check is incorrect, a NAK is sent.

Byte Time-outs

The camera checks the time between the receipt of each byte in the frame. If the time between any two bytes exceeds 0.5 seconds, the camera enters a “garbage state” and discards any more incoming bytes. The camera remains in this state until it sees a new BFS.

Read Commands

In the normal case, when a read command is sent to the camera, the camera responds with an ACK and a returned frame. The returned frame will contain the data requested.

If the camera receives a read command with an unknown or invalid address in the Address field of the frame, it will respond with an ACK but will send no frame.

If the host sends a read command and gets no ACK/NAK, the host can assume that no camera is present or the camera is in a “garbage state” for some reason.

If the host sends a read command and gets an ACK/NAK but does not receive a frame within 500 ms, the host can assume that there was a problem with the read command.

Write Command

In the normal case, when a write command is sent to the camera, the camera responds with an ACK.

If the camera receives a write command with an unknown or invalid address in the Address field of the frame, it will respond with an ACK but will not perform the write.

After a write command has been issued by the host, the host can verify the write by issuing a corresponding read command and checking that the returned data is as expected (unless the address is “write-only”). The host can read the Camera Status fields in the Camera Status inquiry register (see page 4-9) and check the returned data to see if an error condition has been detected.

For many of the write commands listed in the tables on pages 4-34 through 4-44, only data within a specified range or a specified group of values is valid. If the data in a write command is not within the allowed range or specified group of allowed values, the camera **will not** execute the write command.

4.3.2 Basic Read/Write Command Explanations

4.3.2.1 Read Command

This section includes a text description the hex digits included in a command message used to read the Status field of the Test Image Mode CSR (see page 4-32). The intent of this section is to give you a basic understanding of the elements included in a read command. Section 4.4 includes actual samples of the code used to send a read command.

The hex digits included in the read command are:

```
0x01, 0x0C, 0x01, 0x00, 0x18, 0x01, 0x15, 0x03
```

`0x01` is the BFS field.

The value in the BFS field is always 0x01.

`0x0C` is the FTF field.

The hex value of 0x0C in the FTF field converts to a binary value of 0b00001100.

Bits 7 through 3 of the binary value indicate the OpCode. As shown in the table on page 4-47, an OpCode value of 0b00001 indicates that this is a read command frame.

Bit 2 indicates the presence or absence of a BCC in the frame. As shown in the table on page 4-47, when this bit is set to 0b1, it indicates that a BCC is present.

Bits 1 and 0 indicate the AddrLen. As shown in the table on page 4-47, a value of 0b00 for the AddrLen indicates that the address portion of this frame contains a 16-bit address. If you check the table on page 4-32, you will find that the address for the Status field of the Test Image CSR is 0x1800, a 16-bit address. (You are free to use any supported AddrLen as long as the CSR address will fit into it.)

`0x01` is the DataLen field.

This field indicates the data size in bytes that will be transferred by using this read command. As shown in the table on page 4-32, the data size for the Status field of the Test Image CSR is 1 byte.

(Note that for read commands, the data size specified in the DataLen field represents the number of bytes of data that you expect to see in the response frame. No data bytes are actually included in the read command.)

`0x00, 0x18` is the Address field (in little endian).

This field indicates the CSR address from which the data bytes will be read.
The little endian values of 0x00, 0x18 in the address field translate to an address of 0x1800. If you check the table on page 4-32, you will find that 0x1800 is the address for the Status field of the Test Image CSR.

`0x15` is the BCC field.

See page 4-52 for instructions on calculating a BCC.

(Note that the use of a BCC is optional. In this example, we assume that a BCC is used.)

`0x03` is the BFE.

The value in the BFE field is always `0x03`.

4.3.2.2 Write Command

This section includes a text description the hex digits included in a command message used to write a value of 0x01 to the Mode field of the Test Image Mode CSR (see page 4-32). The intent of this section is to give you a basic understanding of the elements included in a write command. Section 4.4 includes actual samples of the code used to send a write command.

The hex digits included in the write command are:

```
0x01, 0x04, 0x04, 0x01, 0x18, 0x01, 0x18, 0x03
```

`0x01` is the BFS field.

The value in the BFS field is always 0x01.

`0x04` is the FTF field.

The hex value of 0x04 in the FTF field converts to a binary value of 0b00000100.

Bits 7 through 3 of the binary value indicate the OpCode. As shown in the table on page 4-47, an OpCode value of 0b00000 indicates that this is a write command frame.

Bit 2 indicates the presence or absence of a BCC in the frame. As shown in the table on page 4-47, when this bit is set to 0b1, it indicates that a BCC is present.

Bits 1 through 0 indicate the AddrLen. As shown in the table on page 4-47, a value of 0b00 for the AddrLen indicates that the Address field in this frame contains a 16-bit address. If you check the table on page 4-32, you will find that the address for the Mode field of the Test Image CSR is 0x1801, a 16-bit address. (You are free to use any supported AddrLen as long as the CSR address will fit into it.)

`0x04` Is the DataLen field.

This field indicates the data size in bytes that will be transferred by using this write command. As shown in the table on page 4-32, the data size for the Mode field of the Test Image Mode CSR is 4 bytes.

`0x01, 0x18` is the Address field in little endian.

This field indicates the CSR address to which the data bytes will be written.
The little endian values of 0x01, 0x18 in the address field translate to an address of 0x1801. If you check the table on page 4-32, you will find that 0x1801 is the address for the Mode field the Test Image Mode CSR. (The address for any field within an A400k CSR is equal to the base address for the CSR plus the offset for the CSR field. In this case, the base address of the Test Image Mode CSR is 1800 and the offset for the Mode field is 0001. This results in an address of 1801 for the Mode field.)

`0x01` is the Data field.

This field contains the data that must be written to the register in order to activate Test Image One (see the table on page 4-32).

`0x18` is the BCC field.

See page 4-52 for instructions on calculating a BCC.
(Note that the use of a BCC is optional. In this example, we assume that a BCC is used.)

`0x03` is the BFE field.

The value in the BFE field is always 0x03.

4.3.2.3 Calculating the Block Check Character (BCC)

The use of a block check character (BCC) in A400k commands is optional (see pages 4-46 and 4-47). If you choose to use a BCC, the BCC will be the exclusive-or sum (XOR sum) of the bytes in the FTF field, the DataLen field, the Address field and the Data field of the command frame. For the write command example shown in Section 4.3.2.2, the block check character is 0x18. Let's consider how this block check character was calculated.

Calculating XOR sums is most easily understood when numbers are shown in their binary form, so in the example calculations shown below, the hexadecimal digits in our command have been converted to binary.

To find the XOR sum of two binary numbers, you add the two digits in each column using the following rules:

If both digits are 0, the result is 0.

If both digits are 1, the result is 0.

If one of the digits is a 1 and the other is a 0, the result is 1.

With all of this in mind, here is how the check digit for the write command shown in Section 4.3.2.2 would be calculated:

```
0 0 0 0  0 1 0 0 = the binary representation of 0x04 (FTF)
0 0 0 0  0 1 0 0 = the binary representation of 0x04 (DataLen)
0 0 0 0  0 0 0 0 = XOR sum

0 0 0 0  0 0 0 0 = Previous XOR Sum
0 0 0 0  0 0 0 1 = the binary representation of 0x00 (Address Byte 1)
0 0 0 0  0 0 0 1 = New XOR sum

0 0 0 0  0 0 0 1 = Previous XOR sum
0 0 0 1  1 0 0 0 = the binary representation of 0x18 (Address Byte 2)
0 0 0 1  1 0 0 1 = New XOR sum

0 0 0 1  1 0 0 1 = Previous XOR Sum
0 0 0 0  0 0 0 1 = the binary representation of 0x01 (Data)
0 0 0 1  1 0 0 0 = Final XOR sum

0 0 0 1  1 0 0 0 = 0x18 = the block check character
```

4.4 Binary Command Sample Code

Sample code that illustrates how to use binary commands with A400k cameras is available at the Basler web site. Please look for the documentation downloads on the A400 page at:

http://www.basler-vc.com

5 Mechanical Considerations

5.1 Camera Dimensions and Mounting Facilities

The A400k camera housing is manufactured with high precision. Planar, parallel, and angular sides guarantee precise mounting with high repeatability.

A400k cameras are equipped with four M4 mounting holes on the front and two M4 mounting holes on each side as indicated in Figure 5-1.

A tripod mount is available as an option. The Basler part number is 1000014110.

Caution!

To avoid collecting dust on the sensor, mount a lens on the camera immediately after unpacking it.

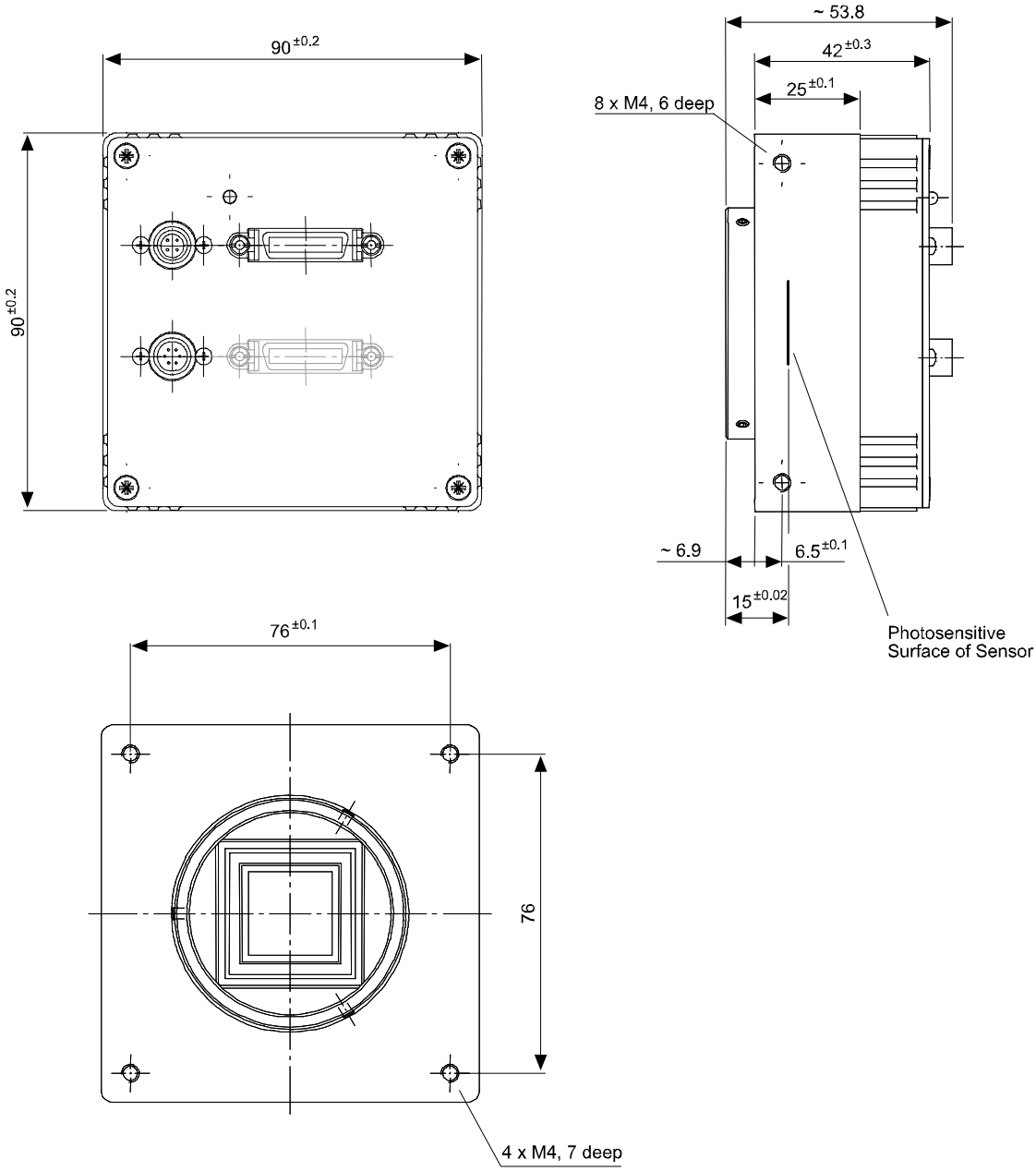
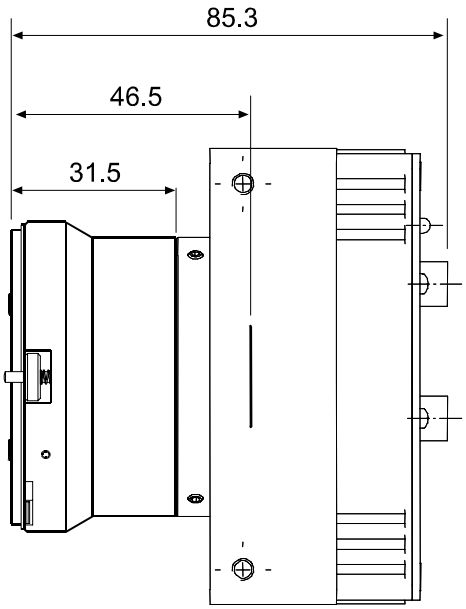


Figure 5-1: A400k Mechanical Dimensions (in mm)

5.2 F-Mount Adapter Dimensions



Drawings are not to scale.

Figure 5-2: F-Mount Adapter Dimensions (in mm)

5.3 Positioning Accuracy of the Sensor Chip

Positioning accuracy of the sensor chip is as shown in Figure 5-3.

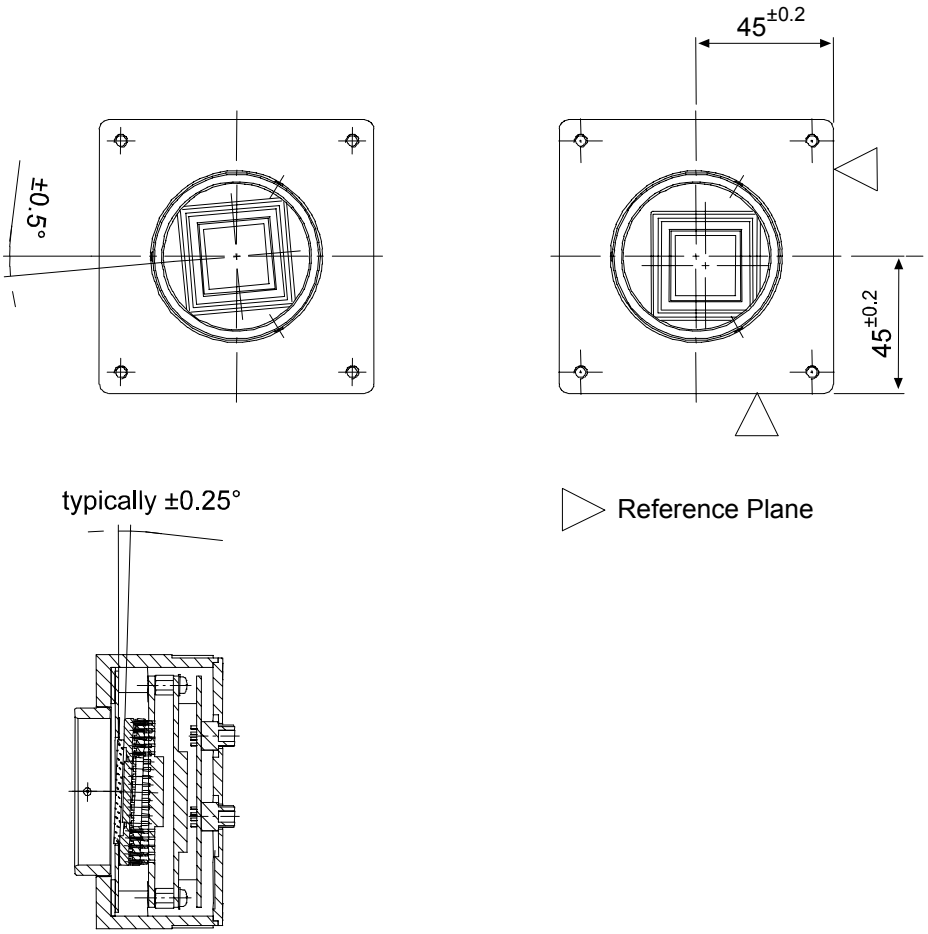


Figure 5-3: Positioning Accuracy

6 Troubleshooting

6.1 Fault Finding Using the Camera LED

If the status LED on the back of the camera is not lit, power to the camera is not present.

When the power supply to the camera is switched on, the LED on the back of the camera will become green colored for several seconds. As the microprocessor in the camera boots up, the LED will blink green and orange. The frequency of blinks will increase until the LED becomes continuously orange colored and then, the blinks will start again. A continuous orange LED indicates that bootup has been completed successfully.

Once bootup is complete, the camera performs a continuous series of self checks. If an error condition is detected, the LED will begin to blink. The number of blinks indicate the detected error as shown in Table 6-1.

If several error states are present, the LED outputs the error code that has the highest priority.

To get more information about the camera's current condition, you can check the camera status as described in Section 3.17.

LED	Description	Priority
Solid orange	The camera has power and is OK.	-
Solid green	The input voltage is less than 10.8 VDC and the camera automatically switched off (undervoltage lockout). To restart the camera, raise the input voltage to the camera as specified in Section 2.8.	1 (highest)
Continuous fast pulses	General error. Please contact Basler support.	2
Repeated pattern of green and orange pulses of increasing frequency	The camera is booting or is busy performing an internal operation (such as generating shading values). When the operation is complete, the blinking stops.	3

Table 6-1: Camera Status Indicators

LED	Description	Priority
Continuous slow pulses	One of the following errors is present: • No sensor board FPGA firmware available. • No processing board FPGA firmware available. • Firmware available but error when booting the sensor FPGA. • Firmware available but error when booting the processing board FPGA. Please contact Basler support.	4
Repeated pattern of 6 slow pulses	An erroneous parameter set has been loaded. Load another parameter set and delete the erroneous set.	5
Repeated pattern of 5 slow pulses	Parameter error. For example, an unavailable parameter setting has been made.	6
Repeated pattern of 4 slow pulses	One of the following errors is present: • A byte time-out has occurred (see Section 4.3.1.1). • Invalid opcode in a read or write command (see Section 4.3.1). • Incoming data has been discarded since no BFS was included (see Section 4.3.1). • Invalid BCC in a read or write command (see Section 4.3.1). • Invalid address in a read or write command (see Section 4.3.1). • Invalid data length in a read or write command (see Section 4.3.1). • An unknown error has occurred. Please contact Basler support.	7
Repeated pattern of 3 slow pulses	The last column FPN shading value generation has failed. The column FPN shading value generation process can fail if the pixel values in the frames captured during the generation process are too high. (The process should be performed in darkness or in very low light conditions.) Check your setup and repeat generation (see Section 3.6.1).	8
Repeated pattern of 2 slow pulses	One of the following errors is present: • The maximum allowed frame rate has been exceeded (see Formula 1 and Formula 2 in Section 3.8.3). • ExSync has not changed state for 5 seconds or longer. If you are not supplying an ExSync signal to the camera, this is a normal condition and should be ignored. Otherwise check the cable and the ExSync generating device.	9 (lowest)

Table 6-1: Camera Status Indicators

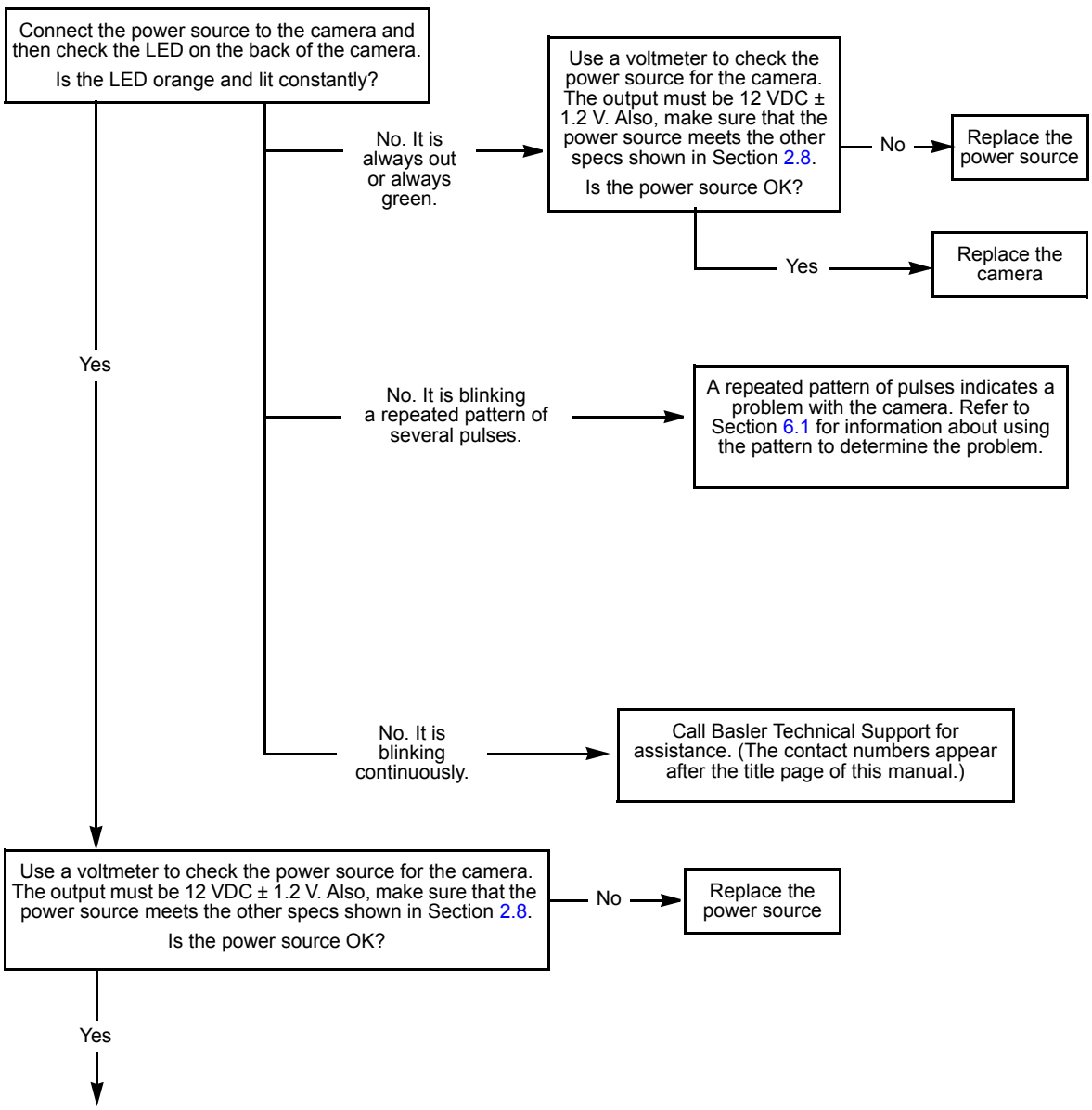
6.2 Troubleshooting Charts

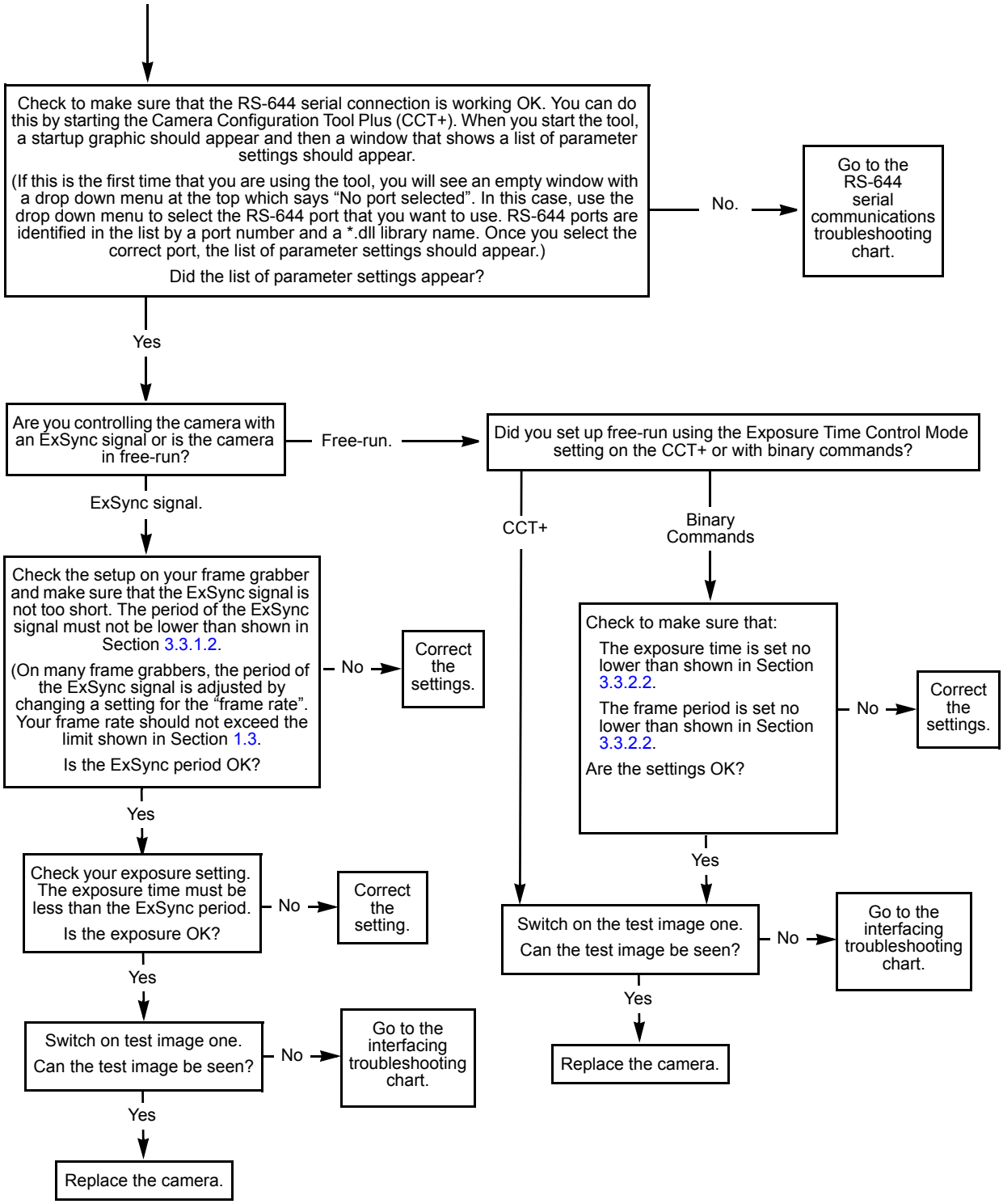
The following pages contain several troubleshooting charts which can help you find the cause of problems that users sometimes encounter. The charts assume that you are familiar with the camera's features and settings and with the settings for your frame grabber. If you are not, we suggest that you review the manuals for your camera and frame grabber before you troubleshoot a problem.

6.2.1 No Image

Use this chart if you see no image at all when you attempt to capture an image with your frame grabber (in this situation, you will usually get a message from the frame grabber such as "time-out"). If you see a poor quality image, a completely black image, or a completely white image, use the chart in Section 6.2.2.

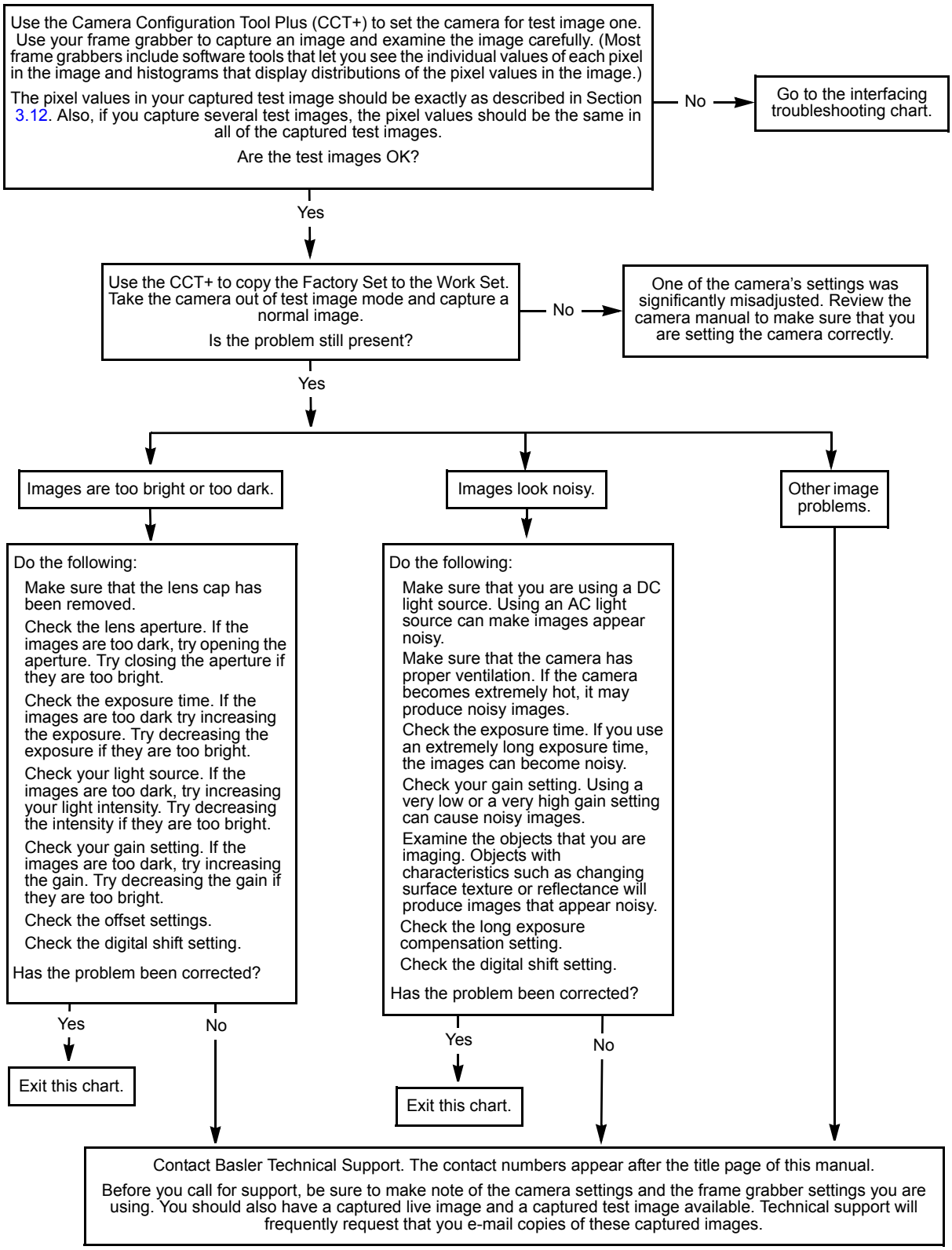
Always switch off power to the system before making or breaking any connection.




Check to make sure that the RS-644 serial connection is working OK. You can do this by starting the Camera Configuration Tool Plus (CCT+). When you start the tool, a startup graphic should appear and then a window that shows a list of parameter settings should appear.
(If this is the first time that you are using the tool, you will see an empty window with a drop down menu at the top which says “No port selected”. In this case, use the drop down menu to select the RS-644 port that you want to use. RS-644 ports are identified in the list by a port number and a *.dll library name. Once you select the correct port, the list of parameter settings should appear.)
Did the list of parameter settings appear?
No.
Go to the RS-644 serial communications troubleshooting chart.
Yes
Are you controlling the camera with an ExSync signal or is the camera in free-run?
Free-run.
Did you set up free-run using the Exposure Time Control Mode setting on the CCT+ or with binary commands?
ExSync signal.
CCT+
Binary Commands
Check the setup on your frame grabber and make sure that the ExSync signal is not too short. The period of the ExSync signal must not be lower than shown in Section 3.3.1.2.
(On many frame grabbers, the period of the ExSync signal is adjusted by changing a setting for the “frame rate”. Your frame rate should not exceed the limit shown in Section 1.3.
Is the ExSync period OK?
No
Correct the settings.
Check to make sure that:
The exposure time is set no lower than shown in Section 3.3.2.2.
The frame period is set no lower than shown in Section 3.3.2.2.
Are the settings OK?
No
Correct the settings.
Yes
Yes
Check your exposure setting. The exposure time must be less than the ExSync period.
Is the exposure OK?
No
Correct the setting.
Switch on the test image one.
Can the test image be seen?
No
Go to the interfacing troubleshooting chart.
Yes
Yes
Switch on test image one.
Can the test image be seen?
No
Go to the interfacing troubleshooting chart.
Yes
Replace the camera.
Yes
Replace the camera.

6.2.2 Poor Quality Image

Use this chart if the image is poor quality, is completely white, or is completely black. If you get no image at all when you attempt to capture an image with the frame grabber, use the chart that appears in Section 6.2.1.

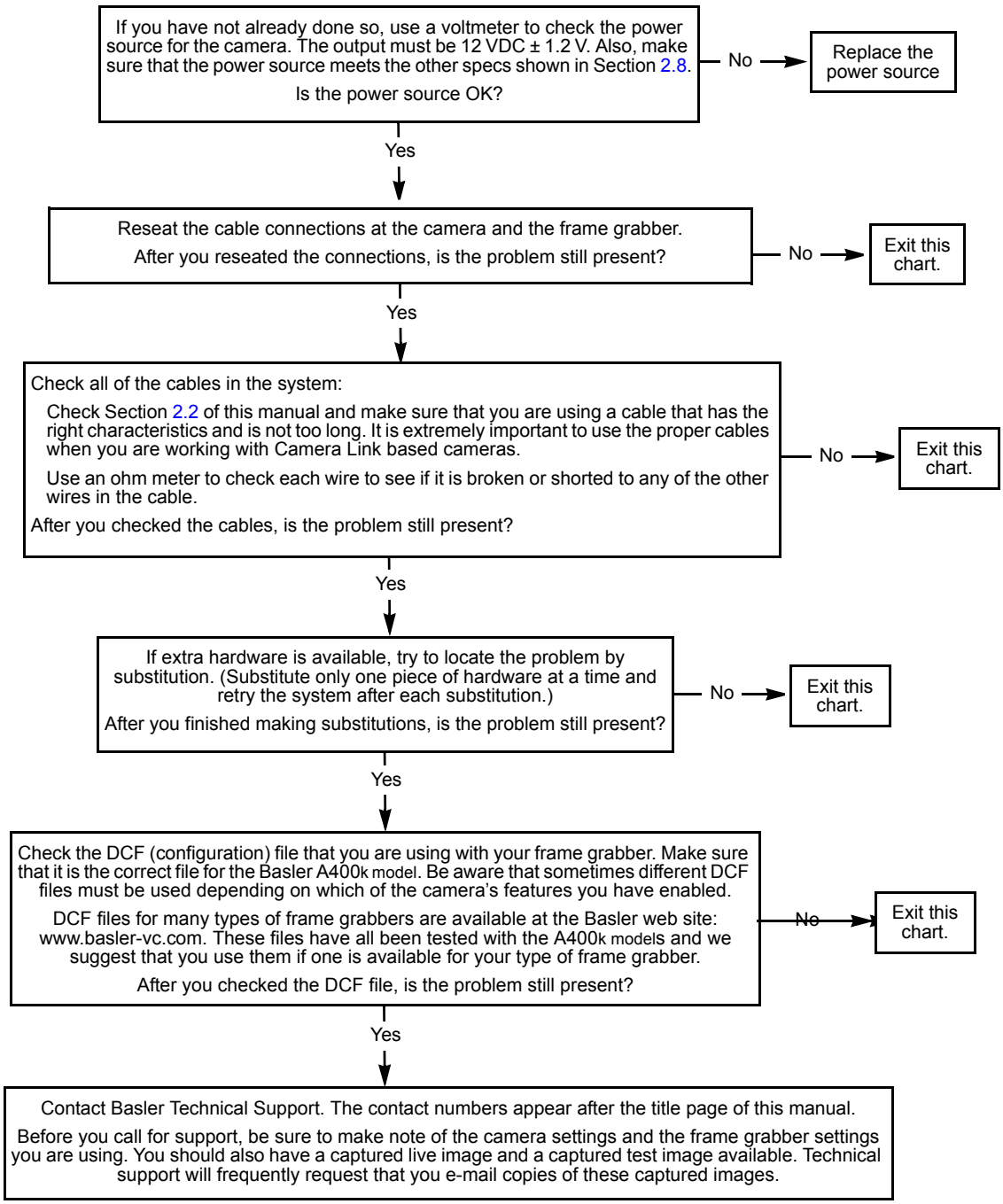


6.2.3 Interfacing

Use the interfacing troubleshooting charts if you think that there is a problem with the cables between your devices or if you have been directed here from another chart.

Interfacing Chart

Always switch off power to the system before making or breaking any connection.

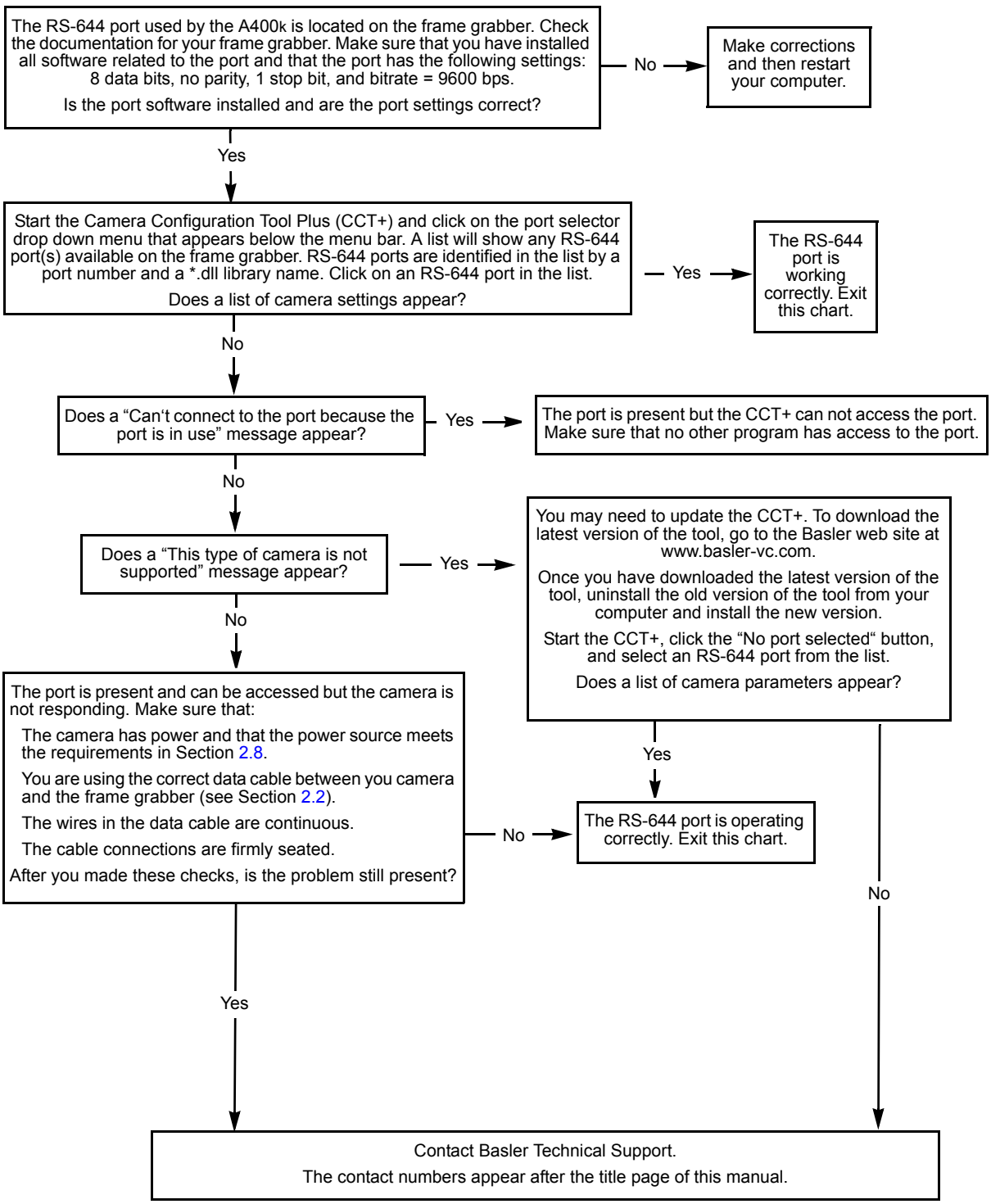


6.2.4 RS-644 Serial Communication

Use the serial communication troubleshooting charts if you think that there is a problem with RS-644 serial communication or if you have been directed here from another chart.

Serial Communication Chart (without a k-BIC)

Always switch off power to the system before making or breaking any connection.



6.3 Before Calling Basler Technical Support

To help you as quickly and efficiently as possible when you have a problem with a Basler camera, it is important that you collect several pieces of information before you contact technical support.

Copy the form that appears on this and the next page (or download it from the support section of www.basler-vc.com), fill it out, and fax the pages to your local dealer or to your nearest Basler support center. Or, you can write an e-mail listing the requested pieces of information and with the requested files attached. Our technical support contact numbers are shown in the front section of this manual.

1 The camera's product ID: _____

2 The camera's serial number: _____

3 The operating system: _____

4 Frame grabber that you use with the camera: _____

_____

5 CCT+ version that you use with the camera: _____

_____

6 Describe the problem in as much detail as possible: _____

(If you need more space, use an extra sheet of paper.) _____

_____

_____

_____

_____

7 If known, what's the cause of the problem? _____

_____

_____

8 When did the problem occur? ☐ After start. ☐ While running.

☐ After a certain action (e.g., a change of parameters):

_____

_____

☐ _____

_____

9 How often did/does the problem occur?

- ☐ Once.
- ☐ Every time.
- ☐ Regularly when: _____
- ☐ Occasionally when: _____

10 How severe is the problem?

- ☐ Camera can still be used.
- ☐ Camera can be used after I take this action: _____
- ☐ Camera can no longer be used.

11 Did your application ever run without problems?

- ☐ Yes
- ☐ No

12 Parameter set

It is very important for Basler Technical Support to get a copy of the exact camera parameters that you were using when the problem occurred.

To make a copy of the parameters, open the parameter set in the CCT+ and save or dump the settings to a file. Send the generated file to Basler Technical Support.

If you cannot access the camera, please try to state the following parameter settings:

- ☐ Video data output mode: _____
- ☐ Exposure time control mode: _____
- ☐ Exposure time: _____
- ☐ Gain: _____
- ☐ Offset: _____

13 Live image/test image

If you are having an image problem, try to generate and save live images that show the problem. Also generate and save test images. Please save the images in BMP format, zip them, and send them to Basler Technical Support.

Revision History

Document Number	Date	Changes
DA00062401	6 February 2004	Initial release version covering prototype cameras only.
DA0006240b	7 June 2004	PRELIMINARY version covering series cameras.
DA00062402	16 July 2004	Initial release version covering series production cameras.
DA00062403	30 August 2004	Corrected the video data output type field in Section 1.3. Corrected the maximum power consumption in Sections 1.3 and 2.8. Corrected the length of the camera housing in Section 1.3. Added a note in Section 3.12. Corrected the drawings in Section 5.
DA00062404	7 October 2004	Added A404k info.
DA00062405	18 March 2005	Added A400kc info Corrected Table 2-1 (functions of pins 7, 20, 8, 21) Corrected Sections 3.12.1, 3.12.2, 3.12.4. Modified in Section 4.2.2.2 descriptions of raw gain and raw offset (settings from 0% to 100%) Removed sample code in Section 4.4. Added a note in Section 4.4.
DA00062406	14 April 2005	In Section 4.2.2.2, the ID for the flash trigger output mode „Always high“ is 0x03 and not 0x04.
DA00062407	21 July 2005	General: Removed „Monochrome Versions Only“ in PRNU Shading Correction description. Replaced Figure 1-3. Added „In the color version, PRNU shading correction is executed for each color separately.“ in Section 3.6.3. Replaced „Effective Exposure Time“ by „Exposure Time“ in Figures 3-8 ”Rolling Shutter” and 3-9 ”Flash Window”. Corrected starting column entries in the AOI list (actual value minus one) in Section 3.8.4 ”Programmable AOI Sequencer”. Introduced the AOI Editor in Section 3.8.4.2 ”Creating an AOI List”. Replaced "The 4 bytes in this field..." by "The 2 bytes in this field..." in tables "Area of Interest Width in Columns CSR" and "Area of Interest Height in Line CSR" in Section 4.2.2.2. Added Section 6.3 ”Before Calling Basler Technical Support”.

Document Number	Date	Changes
DA00062408	16 November 2005	Added Section 1.1 "Document Applicability". Modified description of PRNU in Table 1-2. Modified Figure 2-4: LED of low side switch labelled "typically ≤ +5 V" Added Section 3.10 "Mirror Image" and mirror image mode CSR in Section 4.2.2.2. Corrected Figure 3-17. Modified Figure 5-3: sensor tilt typically ±0.25°. Corrected maximum value in the Raw Exposure Time field of the Exposure Time CSR. Corrected maximum value in the Raw Frame Period field of the Frame Period CSR.

Feedback

Your feedback will help us improve our documentation. Please click the link below to access an online feedback form. Your input is greatly appreciated.

http://www.baslerweb.com/umfrage/survey.html

Index

A

B

C

D

E

F

G

H

I

K

L

M

O

P

Q

R

S

T

V

W