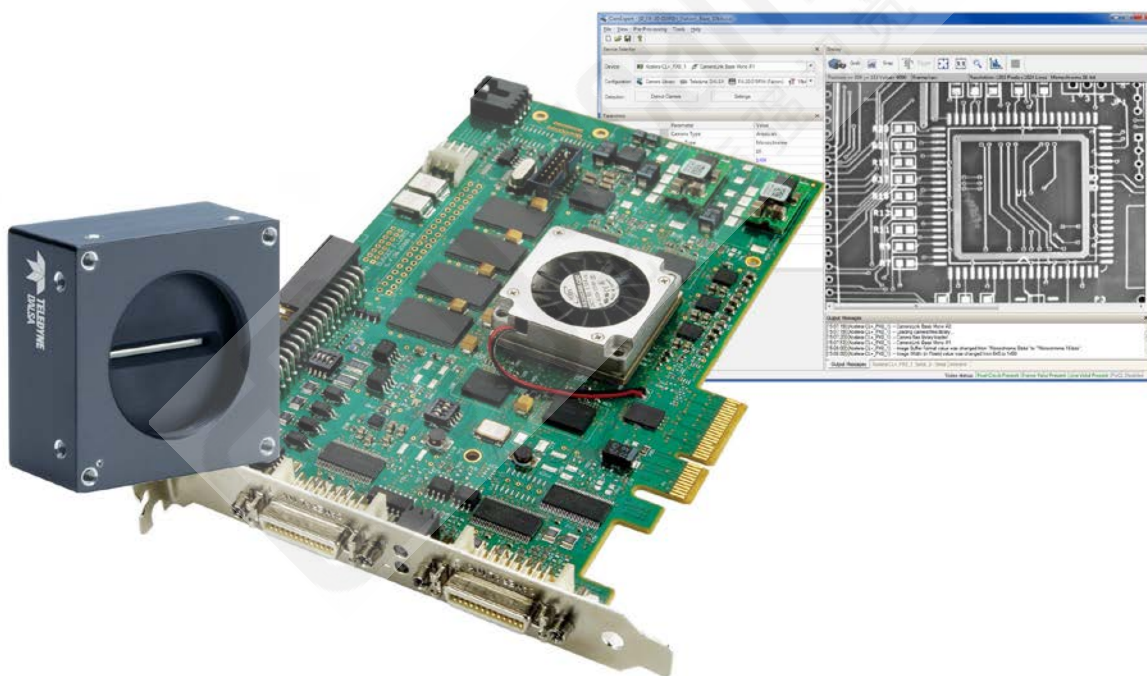


# Sapera LT™ 8.0

## Acquisition Parameters Reference Manual

sensors | cameras | frame grabbers | processors | **software** | vision solutions



P/N: OC-SAPM-APR00  
[www.teledynedalsa.com](http://www.teledynedalsa.com)

 **TELEDYNE DALSA**  
Everywhere you look™

## NOTICE

© 2015 Teledyne DALSA, Inc. All rights reserved.

This document may not be reproduced nor transmitted in any form or by any means, either electronic or mechanical, without the express written permission of TELEDYNE DALSA. Every effort is made to ensure the information in this manual is accurate and reliable. Use of the products described herein is understood to be at the user's risk. TELEDYNE DALSA assumes no liability whatsoever for the use of the products detailed in this document and reserves the right to make changes in specifications at any time and without notice.

Microsoft® is a registered trademark; Windows®, Windows® XP, Windows® Vista, Windows® 7, Windows® 8 are trademarks of Microsoft Corporation.

All other trademarks or intellectual property mentioned herein belongs to their respective owners.

Printed on March 10, 2015

Document Number: OC-SAPM-APR00  
Printed in Canada

### About Teledyne DALSA

Teledyne DALSA is an international high performance semiconductor and electronics company that designs, develops, manufactures, and markets digital imaging products and solutions, in addition to providing wafer foundry services.

Teledyne DALSA Digital Imaging offers the widest range of machine vision components in the world. From industry-leading image sensors through powerful and sophisticated cameras, frame grabbers, vision processors and software to easy-to-use vision appliances and custom vision modules.

# Contents

<b>INTRODUCTION</b>	<b>4</b>
OVERVIEW OF THE MANUAL .....	4
<b>ACQUISITION PARAMETER DEFINITIONS</b>	<b>5</b>
INTRODUCTION .....	5
USING THE ACQUISITION PARAMETERS.....	5
ACQUISITION PARAMETERS .....	6
<i>Acquisition Management Related Parameters</i> .....	6
<i>VIC Related Parameters</i> .....	13
DATA STRUCTURES.....	63
<i>Pin Connector Description</i> .....	63
SIGNAL NAME DEFINITIONS.....	65
STRUCTURE DEFINITIONS .....	66
CAMERA CONTROL METHOD DEFINITIONS .....	67
<i>Camera Reset Method</i> .....	67
<i>Camera Trigger Methods</i> .....	68
<i>Frame Integrate Methods</i> .....	70
<i>Line Integrate Methods</i> .....	72
<i>Line Trigger Methods</i> .....	79
<i>Time Integrate Methods</i> .....	80
<i>Strobe Methods</i> .....	89
CAMERA VIDEO TIMING DEFINITIONS .....	92
<i>Analog Area Scan Video Timings</i> .....	92
<i>Digital Area Scan Video Timings</i> .....	92
<i>Linescan Video Timings</i> .....	92
CUSTOM CAMERA CONTROL I/O DESCRIPTION .....	93
SHAFT ENCODER DESCRIPTION.....	94
PLANAR INPUT SOURCES DESCRIPTION .....	95
<b>ADVANCED ACQUISITION CONTROL</b>	<b>96</b>
INTRODUCTION .....	96
CAMERA RELATED PARAMETERS .....	96
<i>Camera Related Parameters By Groups</i> .....	96
<i>Camera Related Parameters By ID</i> .....	98
<b>CONFIGURATION FILE FORMATS</b>	<b>131</b>
OVERVIEW .....	131
CAMERA DEFINITION FILE DESCRIPTION (CCA) .....	132
VIC PARAMETER FILE DESCRIPTION (CVI) .....	136
<b>APPENDIX: TAP GEOMETRY SETTINGS</b>	<b>140</b>
1 SINGLE TAP GEOMETRIES.....	140
<i>One Tap with Two Channels</i> .....	141
DUAL TAP GEOMETRIES .....	143
3 TAP GEOMETRIES .....	147
4 TAP GEOMETRIES .....	148
8 TAP GEOMETRIES .....	153
<b>CONTACT INFORMATION</b>	<b>154</b>
SALES INFORMATION.....	154
TECHNICAL SUPPORT.....	154

# Introduction

---

## Overview of the Manual

The *Sapera++ LT Programmer's* manual, *Sapera LT .NET Programmer's* and the *Sapera LT Basic Modules Reference* manual are the reference documents for the C++, .NET and C APIs, respectively. The *Sapera LT Acquisition Parameters Reference* manual complements these manuals by describing the parameters, capabilities, and concepts related to the acquisition process. The functions using acquisition parameters are described in the appropriate API reference manual mentioned above.

The *Sapera LT Acquisition Parameters Reference* online manual contains additional references to acquisition parameters and capabilities that typically do not need to be used by the user application.

This manual covers the following topics:

**Sapera LT Acquisition Parameters Definitions**

Description of the Sapera Acquisition parameters plus the related data structures and definitions.

**Teledyne DALSA Contact Information**

Phone numbers, web site, and important email addresses.

**Advanced Acquisition Controls**

Description of acquisition controls including camera parameters and capabilities.

**Appendix A: Acquisition Configuration File Formats**

Description of the Sapera camera configuration files (.CCA, .CVI, .CCF) fields.

# Acquisition Parameter Definitions

---

## Introduction

This section describes the parameters and definitions required for control of the acquisition process.

Refer to the Advanced Acquisition Control section to add advanced controls (such as detection of frame grabber capabilities) to the imaging application.

---

## Using the Acquisition Parameters

A Sopera acquisition configuration is defined through the Acquisition Parameters. These parameters are divided in two categories:

Camera parameters

Video Input Conditioning (VIC) parameters

The Camera parameters describe the signal specifications of the video source (digital or analog). These parameters define the video source capabilities and modes of operation. Consult the section Advanced Acquisition Control for a description of the Camera related parameters.

The VIC related parameters define how the acquisition front end is configured in regards to the video source.

The Camera and VIC parameters typically are stored in CAM & VIC files (files with the .CCA and .CVI extension, respectively) or combined in a unique camera configuration file (file with the .CCF extension) and reloaded at will. This provides a convenient and portable method to initialize the frame grabber with predefined configurations. Sopera LT ships with an extensive list of camera configuration files for supported cameras. In addition, .CVI/CCF files are provided with Teledyne DALSA application notes to support the described camera modes or are generated by the Sopera CamExpert program as required by the imaging application.

CamExpert, the Sopera camera configuration utility, allows configuring the frame grabber (camera configuration file) using existing camera definition files included with the Sopera LT package. The user can also create or modify camera configuration files for new or custom cameras. Multiple .CCF files with different VIC parameters can be created from an existing .CCA file to support various camera operating modes.



It is strongly recommended to start interfacing a camera with your frame grabber using CamExpert instead of experimenting directly with one of the supplied demo programs. CamExpert is designed to guide you through the camera interfacing process with minimum effort.

When dynamic control is required, such as brightness and contrast, the Sopera API provides functions for direct access to any Camera or VIC parameter.

The possible values of an acquisition parameter and its availability are generally indicated by Sopera Acquisition capabilities (CORACO\_CAP\_\*).



Note: Sopera Acquisition capabilities are INT32 values, unless specified otherwise.

---

# Acquisition Parameters

This section describes the Acquisition Management and VIC related parameters. Unlike the VIC parameters, the Acquisition Management parameters are not stored in any acquisition configuration files.

## Acquisition Management Related Parameters

ID	Parameter
0x700	CORACQ_PRM_LABEL
0x701	CORACQ_PRM_EVENT_TYPE
0x702	CORACQ_PRM_EVENT_COUNT
0x703	CORACQ_PRM_EVENT_SERVER
0x704	CORACQ_PRM_EVENT_CALLBACK
0x705	CORACQ_PRM_EVENT_CONTEXT
0x706	CORACQ_PRM_HSYNC_TIMEOUT
0x707	CORACQ_PRM_VSYNC_TIMEOUT
0x708	CORACQ_PRM_SIGNAL_STATUS
0x709	Reserved
0x70a	Reserved
0x70b	Reserved
0x70c	CORACQ_PRM_FLAT_FIELD_SELECT
0x70d	CORACQ_PRM_FLAT_FIELD_ENABLE
0x70e	CORACQ_PRM_EVENT_TYPE_EX
0x70f	CORACQ_PRM_TIME_STAMP

### CORACQ\_PRM\_EVENT\_CALLBACK

**Description** Pointer to the Callback function registered using the function CorAcqRegisterCallback and CorAcqRegisterCallbackEx.

**Type** PCORCALLBACK

**Note** This parameter is read-only.

### CORACQ\_PRM\_EVENT\_CONTEXT

**Description** Context pointer registered using the function CorAcqRegisterCallback.

**Type** void \*

**Note** This parameter is read-only.

### CORACQ\_PRM\_EVENT\_COUNT

**Description** Number of events that have occurred since a callback function was registered using the CorAcqRegisterCallback function.

**Type** UINT32

**Note** This parameter is read-only.

### CORACQ\_PRM\_EVENT\_SERVER

**Description** Handle to a server to which an event notification is made via a callback function.

**Type** CORSERVER

**Note** This parameter is read-only.



## CORACQ\_PRM\_EVENT\_TYPE

<b>Description</b>	Event to be signaled while a transfer is in progress, unless otherwise specified.
<b>Type</b>	UINT32
<b>Limits</b>	The CORACQ_CAP_EVENT_TYPE capability specifies the event type(s) supported by the acquisition module. The capability returns the supported values ORed together.
<b>Values</b>	<p>The values may be ORed if more than one event is desired.</p> <p>CORACQ_VAL_EVENT_TYPE_DATA_OVERFLOW (0x00004000) Call the callback function when a data overflow occurs during live acquisition. This error can usually occur if the acquisition device cannot sustain the data rate of the incoming images.</p> <p>CORACQ_VAL_EVENT_TYPE_END_OF_EVEN (0x00400000) Call the callback function at end of even field.</p> <p>CORACQ_VAL_EVENT_TYPE_END_OF_FIELD (0x00100000) Call the callback function at end of odd or even field.</p> <p>CORACQ_VAL_EVENT_TYPE_END_OF_FRAME (0x00800000) Call the callback function at end of frame.</p> <p>CORACQ_VAL_EVENT_TYPE_END_OF_LINE (0x04000000) Call the callback function at end of line <i>n</i>.</p> <p>CORACQ_VAL_EVENT_TYPE_END_OF_NLINES (0x08000000) Call the callback function at end of <i>n</i> lines.</p> <p>CORACQ_VAL_EVENT_TYPE_END_OF_ODD (0x00200000) Call the callback function at end of odd field.</p> <p>CORACQ_VAL_EVENT_TYPE_END_OF_TRANSFER (0x00000002) Call the callback function at end of transfer.</p> <p>CORACQ_VAL_EVENT_TYPE_EXTERNAL_TRIGGER (0x01000000) Call the callback function upon receiving an external trigger which will then acquire at least one image. Therefore, the maximum callback rate cannot be greater than the acquisition video frame rate. See also CORACQ_PRM_EXT_TRIGGER_ENABLE</p> <p>CORACQ_VAL_EVENT_TYPE_EXTERNAL_TRIGGER_IGNORED (0x00002000) Call the callback function when an external trigger event is dropped. This occurs when the external trigger rate is faster than the acquisition frame rate. See also CORACQ_PRM_EXT_TRIGGER_ENABLE.</p> <p>CORACQ_VAL_EVENT_TYPE_EXT_LINE_TRIGGER_TOO_SLOW (0x00000400) Call the callback function if the detected line trigger rate is too slow for the hardware to process. This event can occur when using the shaft encoder multiplier.</p> <p>CORACQ_VAL_EVENT_TYPE_FRAME_LOST (0x00008000) Call the callback function for each frames lost during live acquisition. This error can usually occur if there is not enough bandwidth to transfer images to host memory.</p> <p>CORACQ_VAL_EVENT_TYPE_HSYNC_LOCK (0x00000800) Call the callback function if a horizontal sync unlock to lock condition is detected.</p> <p>CORACQ_VAL_EVENT_TYPE_HSYNC_UNLOCK (0x00001000) Call the callback function if an horizontal sync lock to unlock condition is detected.</p> <p>CORACQ_VAL_EVENT_TYPE_LINE_TRIGGER_TOO_FAST (0x00000008) Call the callback function if no line is received on the frame grabber following a line trigger to a camera. Usually this occurs when the line trigger rate is too fast for the camera.</p> <p>CORACQ_VAL_EVENT_TYPE_LINK_ERROR (0x00000010) Call the callback function when an error occurs on the link between the camera and the frame grabber (for HSLink cameras only). The exact error condition may be one of the following: 8-bit/10-bit encoding, packet header error, CRC error, bad revision, or lost idle lock.</p> <p>CORACQ_VAL_EVENT_TYPE_LINK_LOCK (0x10000000LL) Call the callback function when all required lanes are locked (for HSLink and CLHS cameras only).</p> <p>CORACQ_VAL_EVENT_TYPE_LINK_UNLOCK (0x20000000LL) Call the callback function if at least one of the required lanes loses the lock (for HSLink and CLHS cameras only)</p>

CORACQ\_VAL\_EVENT\_TYPE\_NO\_HSYNC (0x10000000)

Call the callback function if a timeout occurs due to a missing horizontal sync during live acquisition. The timeout value is specified by CORACQ\_PRM\_HSYNC\_TIMEOUT. The event is only generated once, unless a new CorXferStart command is issued or a new horizontal sync is detected.

CORACQ\_VAL\_EVENT\_TYPE\_NO\_PIXEL\_CLK (0x40000000)

Call the callback function if no pixel clock is detected. The event is only generated once, unless a new CorXferStart command is issued or the pixel clock is detected again and then lost.

CORACQ\_VAL\_EVENT\_TYPE\_NO\_VSYNC (0x20000000)

Call the callback function if a timeout occurs due to a missing vertical sync during live acquisition. The timeout value is specified by CORACQ\_PRM\_VSYNC\_TIMEOUT. The event is only generated once, unless a new CorXferStart command is issued or a new vertical sync is detected.

CORACQ\_VAL\_EVENT\_TYPE\_PIXEL\_CLK (0x80000000)

Call the callback function if a pixel clock is detected. The event is only generated once, unless a new CorXferStart command is issued or the pixel clock is lost again and then detected.

CORACQ\_VAL\_EVENT\_TYPE\_SHAFT\_ENCODER\_REVERSE\_COUNT\_OVERFLOW  
(0x00000004)

Call the callback function when an overflow of the shaft encoder reverse counter occurs.

CORACQ\_VAL\_EVENT\_TYPE\_START\_OF\_EVEN (0x00040000)

Call the callback function at start of even field.

CORACQ\_VAL\_EVENT\_TYPE\_START\_OF\_FIELD (0x00020000)

Call the callback function at start of odd or even field.

CORACQ\_VAL\_EVENT\_TYPE\_START\_OF\_FRAME (0x00080000)

Call the callback function at start of frame.

CORACQ\_VAL\_EVENT\_TYPE\_START\_OF\_ODD (0x00020000)

Call the callback function at start of odd field.

CORACQ\_VAL\_EVENT\_TYPE\_USER\_DEFINE (0x00000200)

Call the callback function when a "user defined" event occurs. Applicable when custom firmware which supports the user defined event, is loaded on to the acquisition board. This event does not have any other identification thus only the application can know the meaning of the user defined event.

CORACQ\_VAL\_EVENT\_TYPE\_VERTICAL\_SYNC (0x02000000)

Call the callback function on every vertical sync, even if not acquiring.

CORACQ\_VAL\_EVENT\_TYPE\_VERTICAL\_TIMEOUT (0x00000040)

Call the callback function if the end of the vertical sync (analog cameras) or beginning of frame valid (digital cameras) is not received within the specified delay. The timeout value is specified by CORACQ\_PRM\_VERTICAL\_TIMEOUT\_DELAY.

## CORACQ\_PRM\_EVENT\_TYPE\_EX

<b>Description</b>	Event to be signaled while a transfer is in progress, unless otherwise specified.
<b>Type</b>	UINT64
<b>Limits</b>	The CORACQ_CAP_EVENT_TYPE_EX capability specifies the event type(s) supported by the acquisition module. The capability returns the supported values ORed together.
<b>Values</b>	The list of values are the same as CORACQ_PRM_EVENT_TYPE.
<b>Notes</b>	This parameter allows for the future expansion beyond the current 32-bit limitation of CORACQ_PRM_EVENT_TYPE.



### **CORACQ\_PRM\_FLAT\_FIELD\_ENABLE**

<b>Description</b>	Enable or disable the flat field resource.
<b>Type</b>	UINT32
<b>Availability</b>	Available only if CORACQ_CAP_FLAT_FIELD is TRUE
<b>Values</b>	TRUE (0x00000001), Enable the flat field FALSE (0x00000000), Disable the flat field
<b>CVI entry</b>	None
<b>Related Capabilities</b>	<p>The flat field correction algorithm can be further characterized by the following capabilities.</p> <p>Below are the relative minimum and maximum pixel gains: CORACQ_CAP_FLAT_FIELD_GAIN_MIN and CORACQ_CAP_FLAT_FIELD_GAIN_MAX</p> <p>Divide the relative pixel gain by CORACQ_CAP_FLAT_FIELD_GAIN_DIVISOR to get the actual gain value.</p> <p>Example: for: CORACQ_CAP_FLAT_FIELD_GAIN_MIN = 0x01 CORACQ_CAP_FLAT_FIELD_GAIN_MAX = 0xFF CORACQ_CAP_FLAT_FIELD_GAIN_DIVISOR = 0x80 then: Minimum gain is <math>1 / 0x80 = 0.0078125</math> Maximum gain is <math>0xFF / 0x80 = 1.9921875</math></p> <p>Below are the minimum and maximum gray level pixel offsets: CORACQ_CAP_FLAT_FIELD_OFFSET_MIN and CORACQ_CAP_FLAT_FIELD_OFFSET_MAX</p> <p>CORACQ_CAP_FLAT_FIELD_PIXEL_REPLACEMENT returns TRUE if pixel replacement is supported. A gain of zero indicates a pixel replacement.</p>

### **CORACQ\_PRM\_FLAT\_FIELD\_SELECT**

<b>Description</b>	Selects the active flat field resource created using the function CorAcqNewFlatfield.
<b>Type</b>	UINT32
<b>Availability</b>	Available only if CORACQ_CAP_FLAT_FIELD is TRUE
<b>Values</b>	0 ... (n-1), where 'n' is the number of flat field resources created. The maximum number that can be created is limited by the amount of memory available on the PC and/or on the device.
<b>CVI entry</b>	None

### **CORACQ\_PRM\_HSYNC\_TIMEOUT**

<b>Description</b>	Timeout value (in $\mu$ sec) used to generate the event "horizontal loss of sync" (CORACQ_VAL_EVENT_TYPE_NO_HSYNC).
<b>Type</b>	UINT32

### **CORACQ\_PRM\_LABEL**

<b>Description</b>	Acquisition device ID: Zero-terminated array of characters with a fixed size of 128 bytes.
<b>Type</b>	CHAR[128]
<b>Note</b>	This parameter is read-only.

## CORACQ\_PRM\_SIGNAL\_STATUS

<b>Description</b>	Status of input signals connected to the acquisition device. The returned value is the ORed combination of all valid values.
<b>Type</b>	UINT32
<b>Limits</b>	The CORACQ_CAP_SIGNAL_STATUS capability returns the supported values ORed together.
<b>Values</b>	<p>CORACQ_VAL_SIGNAL_HSYNC_PRESENT (0x00000001) True if an horizontal sync signal (analog video source) or a line valid (digital video source) has been detected by the acquisition device.</p> <p>CORACQ_VAL_SIGNAL_VSYNC_PRESENT (0x00000002) True if a vertical sync signal (analog video source) or a frame valid (digital video source) has been detected by the acquisition device.</p> <p>CORACQ_VAL_SIGNAL_PIXEL_CLK_PRESENT (0x00000004) True if a pixel clock signal has been detected by the acquisition device.</p> <p>CORACQ_VAL_SIGNAL_PIXEL_CLK_1_PRESENT (0x00000004) For CameraLink devices, this status returns true if a clock signal is detected on the base cable.</p> <p>CORACQ_VAL_SIGNAL_PIXEL_CLK_2_PRESENT (0x00000200) For CameraLink devices, this status returns true if a clock signal is detected on the medium cable.</p> <p>CORACQ_VAL_SIGNAL_PIXEL_CLK_3_PRESENT (0x00000400) For CameraLink devices, this status returns true if a clock signal is detected on the full cable.</p> <p>CORACQ_VAL_SIGNAL_PIXEL_CLK_ALL_PRESENT (0x00000800) For Camera Link devices, true if all required pixel clock signals have been detected by the acquisition device based on the CameraLink configuration selected.</p> <p>CORACQ_VAL_SIGNAL_CHROMA_PRESENT (0x00000008) True if a color burst signal has been detected by the acquisition device. This is valid for NTSC and PAL video signals.</p> <p>CORACQ_VAL_SIGNAL_HSYNC_LOCK (0x00000010) True if the acquisition device has been able to lock to an horizontal sync signal (analog video source).</p> <p>CORACQ_VAL_SIGNAL_VSYNC_LOCK (0x00000020) True if the acquisition device has been able to lock to a vertical sync signal (analog video source).</p> <p>CORACQ_VAL_SIGNAL_POWER_PRESENT (0x00000040) True if power is available for a camera. When true, this indicates only that power is available at the camera connector, where it might be supplied from the board PCI bus or from the board PC power connector (whether this power is used by the camera is unknown). When false, the circuit fuse is blown and power cannot be supplied to any connected camera. (See board manual for information on any fused power supply for cameras).</p> <p>CORACQ_VAL_SIGNAL_POCL_ACTIVE (0x00000080) True if power is applied to the camera through the 1<sup>st</sup> CameraLink cable to the camera connector and the camera is PoCL compliant. When false and the parameter CORACQ_PRM_POCL_ENABLE is TRUE, means the Camera is not PoCL compliant, the wrong cable is used, or the camera is not connected.</p> <p>CORACQ_VAL_SIGNAL_POCL_ACTIVE_2 (0x00000100) True if power is applied to the camera through the 2<sup>nd</sup> CameraLink cable to the camera connector and the camera is PoCL compliant. When false and the parameter CORACQ_PRM_POCL_ENABLE is TRUE, means the Camera is not PoCL compliant, the wrong cable is used, or the camera is not connected.</p> <p>CORACQ_VAL_SIGNAL_LINK_LOCK (0x00001000) For HSLink and CLHS devices, true if all lane lock signals necessary have been detected by the acquisition device based on the configuration selected.</p>

CORACQ\_VAL\_SIGNAL\_LANE1\_LOCK (0x00002000)

For HSLink and CLHS devices, true rue if the lane 1 lock signal has been detected by the acquisition device.

CORACQ\_VAL\_SIGNAL\_LANE2\_LOCK (0x00004000)

For HSLink and CLHS devices, true rue if the lane 2 lock signal has been detected by the acquisition device.

CORACQ\_VAL\_SIGNAL\_LANE3\_LOCK (0x00008000)

For HSLink and CLHS devices, true rue if the lane 3 lock signal has been detected by the acquisition device.

CORACQ\_VAL\_SIGNAL\_LANE4\_LOCK (0x00010000)

For HSLink and CLHS devices, true rue if the lane 4 lock signal has been detected by the acquisition device.

CORACQ\_VAL\_SIGNAL\_LANE5\_LOCK (0x00020000)

For HSLink and CLHS devices, true rue if the lane 5 lock signal has been detected by the acquisition device.

CORACQ\_VAL\_SIGNAL\_LANE6\_LOCK (0x00040000)

For HSLink and CLHS devices, true rue if the lane 6 lock signal has been detected by the acquisition device.

CORACQ\_VAL\_SIGNAL\_LANE7\_LOCK (0x00080000)

For HSLink and CLHS devices, true rue if the lane 7 lock signal has been detected by the acquisition device.

**Note**

This parameter is read-only.

## **CORACQ\_PRM\_TIME\_STAMP**

<b>Description</b>	Returns the current value of the acquisition device time stamp. This value is normally expressed in microseconds. This time stamp is passed to events and can also be used to time stamp host buffers. Note that the time stamp base units can be selected using the <a href="#">CORACQ_PRM_TIME_STAMP_BASE</a> parameter.
<b>Type</b>	UINT64
<b>Values</b>	Time stamp
<b>Note</b>	Writing to the parameter will reset the time stamp counter to 0.

## **CORACQ\_PRM\_VSYNC\_TIMEOUT**

<b>Description</b>	Timeout value (in $\mu\text{sec}$ ) used to generate the event "vertical loss of sync" (CORACQ_VAL_EVENT_TYPE_NO_VSYNC).
<b>Type</b>	UINT32

## VIC Related Parameters

The following table lists VIC parameters by functional groups. A table listing VIC parameters sorted by their ID is available in the section VIC Parameters by ID.

Typically the acquisition hardware is initialized with Camera and VIC parameters by loading a camera configuration file. These parameters (such as the ones controlling brightness and contrast) can then be modified individually at runtime by the user application.

### VIC Parameters by Groups

General	
CORACQ_PRM_VIC_NAME	

Input	
CORACQ_PRM_BIT_ORDERING	CORACQ_PRM_CAMSEL
CORACQ_PRM_PLANAR_INPUT_SOURCES	

Signal Conditioning	
CORACQ_PRM_BRIGHTNESS	CORACQ_PRM_FIX_FILTER_ENABLE
CORACQ_PRM_BRIGHTNESS_RED	CORACQ_PRM_FIX_FILTER_SELECTOR
CORACQ_PRM_BRIGHTNESS_GREEN	CORACQ_PRM_FIX_FILTER_SELECTOR_STR
CORACQ_PRM_BRIGHTNESS_BLUE	CORACQ_PRM_HUE
CORACQ_PRM_CONTRAST	CORACQ_PRM_SCALE_VERT
CORACQ_PRM_CONTRAST_RED	CORACQ_PRM_PROG_FILTER_ENABLE
CORACQ_PRM_CONTRAST_GREEN	CORACQ_PRM_PROG_FILTER_FREQ
CORACQ_PRM_CONTRAST_BLUE	CORACQ_PRM_SATURATION
CORACQ_PRM_DC_REST_MODE	CORACQ_PRM_SHARPNESS
CORACQ_PRM_DC_REST_START	

Stream Conditioning	
CORACQ_PRM_CROP_ACTIVATION	CORACQ_PRM_LUT_FORMAT
CORACQ_PRM_CROP_LEFT	CORACQ_PRM_LUT_MAX
CORACQ_PRM_CROP_TOP	CORACQ_PRM_LUT_NENTRIES
CORACQ_PRM_CROP_HEIGHT	CORACQ_PRM_LUT_NUMBER
CORACQ_PRM_CROP_WIDTH	CORACQ_PRM_PIXEL_MASK
CORACQ_PRM_DECIMATE_COUNT	CORACQ_PRM_SCALE_HORZ
CORACQ_PRM_DECIMATE_METHOD	CORACQ_PRM_SCALE_HORZ_METHOD
CORACQ_PRM_EXT_TRIGGER_FRAME_COUNT	CORACQ_PRM_SCALE_VERT
CORACQ_PRM_FRAME_LENGTH	CORACQ_PRM_SCALE_VERT_METHOD
CORACQ_PRM_FLIP	CORACQ_PRM_SNAP_COUNT
CORACQ_PRM_HSYNC_REF	CORACQ_PRM_VSYNC_REF
CORACQ_PRM_LUT_ENABLE	

Control Signals	
CORACQ_PRM_CAM_CONTROL_PULSE0_HD_ALIGN	CORACQ_PRM_INT_FRAME_TRIGGER_FREQ
CORACQ_PRM_CAM_CONTROL_PULSE1_HD_ALIGN	CORACQ_PRM_INT_LINE_TRIGGER_ENABLE
CORACQ_PRM_CAM_RESET_DELAY	CORACQ_PRM_INT_LINE_TRIGGER_FREQ
CORACQ_PRM_CAM_RESET_ENABLE	CORACQ_PRM_INT_LINE_TRIGGER_FREQ_MIN
CORACQ_PRM_CAM_TRIGGER_DELAY	CORACQ_PRM_INT_LINE_TRIGGER_FREQ_MAX

Control Signals	
CORACQ_PRM_CAM_TRIGGER_ENABLE	CORACQ_PRM_LINE_INTEGRATE_DURATION
CORACQ_PRM_BOARD_SYNC_OUTPUT1	CORACQ_PRM_LINE_INTEGRATE_ENABLE
CORACQ_PRM_BOARD_SYNC_OUTPUT2	CORACQ_PRM_LINE_TRIGGER_ENABLE
CORACQ_PRM_EXT_FRAME_TRIGGER_DETECTION	CORACQ_PRM_LINE_TRIGGER_AUTO_DELAY
CORACQ_PRM_EXT_FRAME_TRIGGER_ENABLE	CORACQ_PRM_LINESCAN_DIRECTION_OUTPUT
CORACQ_PRM_EXT_FRAME_TRIGGER_LEVEL	CORACQ_PRM_MASTER_MODE
CORACQ_PRM_EXT_FRAME_TRIGGER_SOURCE	CORACQ_PRM_MASTER_MODE_HSYNC_POLARITY
CORACQ_PRM_EXT_LINE_TRIGGER_DETECTION	CORACQ_PRM_MASTER_MODE_VSYNC_POLARITY
CORACQ_PRM_EXT_LINE_TRIGGER_ENABLE	CORACQ_PRM_SHAFT_ENCODER_ENABLE
CORACQ_PRM_EXT_LINE_TRIGGER_LEVEL	CORACQ_PRM_SHAFT_ENCODER_LEVEL
CORACQ_PRM_EXT_LINE_TRIGGER_SOURCE	CORACQ_PRM_SHAFT_ENCODER_DROP
CORACQ_PRM_EXT_LINE_TRIGGER_SOURCE_STR	CORACQ_PRM_SHAFT_ENCODER_MULTIPLY
CORACQ_PRM_EXT_TRIGGER_DELAY	CORACQ_PRM_SHAFT_ENCODER_SOURCE
CORACQ_PRM_EXT_TRIGGER_DELAY_TIME_BASE	CORACQ_PRM_SHAFT_ENCODER_SOURCE_STR
CORACQ_PRM_EXT_TRIGGER_DETECTION	CORACQ_PRM_STROBE_DELAY
CORACQ_PRM_EXT_TRIGGER_DURATION	CORACQ_PRM_STROBE_DELAY_2
CORACQ_PRM_EXT_TRIGGER_ENABLE	CORACQ_PRM_STROBE_DURATION
CORACQ_PRM_EXT_TRIGGER_IGNORE_DELAY	CORACQ_PRM_STROBE_ENABLE
CORACQ_PRM_EXT_TRIGGER_LEVEL	CORACQ_PRM_STROBE_LEVE
CORACQ_PRM_EXT_TRIGGER_SOURCE	CORACQ_PRM_STROBE_METHOD
CORACQ_PRM_EXT_TRIGGER_SOURCE_STR	CORACQ_PRM_STROBE_POLARITY
CORACQ_PRM_FIX_FILTER_SELECTOR_STR	CORACQ_PRM_TIME_INTEGRATE_DELAY
CORACQ_PRM_FRAME_INTEGRATE_COUNT	CORACQ_PRM_TIME_INTEGRATE_DURATION
CORACQ_PRM_SHAFT_ENCODER_DIRECTION	CORACQ_PRM_TIME_INTEGRATE_ENABLE
CORACQ_PRM_FRAME_INTEGRATE_ENABLE	CORACQ_PRM_VERTICAL_TIMEOUT_DELAY
CORACQ_PRM_INT_FRAME_TRIGGER_ENABLE	

Output	
CORACQ_PRM_OUTPUT_ENABLE (obsolete) use CORACQ_PRM_EXT_TRIGGER_ENABLE	CORACQ_PRM_OUTPUT_FORMAT

Shared Control Signals	
CORACQ_PRM_SHARED_CAM_RESET	CORACQ_PRM_SHARED_STROBE
CORACQ_PRM_SHARED_CAM_TRIGGER	CORACQ_PRM_SHARED_TIME_INTEGRATE
CORACQ_PRM_SHARED_EXT_TRIGGER	CORACQ_PRM_WEN_ENABLE
CORACQ_PRM_SHARED_FRAME_INTEGRATE	

Color Signals
CORACQ_PRM_COLOR_DECODER_ENABLE
CORACQ_PRM_COLOR_DECODER_METHOD
CORACQ_PRM_BAYER_DECODER_SATURATION_FACTOR
CORACQ_PRM_BAYER_DECODER_SATURATION_WEIGHT_BLUE
CORACQ_PRM_BAYER_DECODER_SATURATION_WEIGHT_GREEN
CORACQ_PRM_BAYER_DECODER_SATURATION_WEIGHT_RED
CORACQ_PRM_WB_GAIN_RED



CORACQ_PRM_WB_GAIN_GREEN
CORACQ_PRM_WB_GAIN_BLUE
CORACQ_PRM_WB_OFFSET_RED
CORACQ_PRM_WB_OFFSET_GREEN
CORACQ_PRM_WB_OFFSET_BLUE

### VIC Parameters by ID

0x800	CORACQ_PRM_CAMSEL
0x801	CORACQ_PRM_PIXEL_MASK
0x802	CORACQ_PRM_DC_REST_MODE
0x803	CORACQ_PRM_BRIGHTNESS
0x804	CORACQ_PRM_BRIGHTNESS_RED
0x805	CORACQ_PRM_BRIGHTNESS_GREEN
0x806	CORACQ_PRM_BRIGHTNESS_BLUE
0x807	CORACQ_PRM_CONTRAST
0x808	CORACQ_PRM_CONTRAST_RED
0x809	CORACQ_PRM_CONTRAST_GREEN
0x80a	CORACQ_PRM_CONTRAST_BLUE
0x80b	CORACQ_PRM_HUE
0x80c	CORACQ_PRM_SATURATION
0x80d	CORACQ_PRM_FIX_FILTER_ENABLE
0x80e	CORACQ_PRM_FIX_FILTER_SELECTOR
0x80f	CORACQ_PRM_PROG_FILTER_ENABLE
0x810	CORACQ_PRM_PROG_FILTER_FREQ
0x811	CORACQ_PRM_CROP_LEFT
0x812	CORACQ_PRM_CROP_TOP
0x813	CORACQ_PRM_CROP_WIDTH
0x814	CORACQ_PRM_CROP_HEIGHT
0x815	CORACQ_PRM_SCALE_HORZ
0x816	CORACQ_PRM_SCALE_VERT
0x817	CORACQ_PRM_SCALE_HORZ_METHOD
0x818	CORACQ_PRM_SCALE_VERT_METHOD
0x819	CORACQ_PRM_DECIMATE_METHOD
0x81a	CORACQ_PRM_DECIMATE_COUNT
0x81b	CORACQ_PRM_LUT_ENABLE
0x81c	CORACQ_PRM_LUT_NUMBER
0x81d	CORACQ_PRM_STROBE_ENABLE
0x81e	CORACQ_PRM_STROBE_METHOD
0x81f	CORACQ_PRM_STROBE_POLARITY
0x820	CORACQ_PRM_STROBE_DURATION
0x821	CORACQ_PRM_STROBE_DELAY
0x822	CORACQ_PRM_FRAME_INTEGRATE_ENABLE
0x823	CORACQ_PRM_FRAME_INTEGRATE_COUNT
0x824	CORACQ_PRM_TIME_INTEGRATE_ENABLE
0x825	CORACQ_PRM_TIME_INTEGRATE_DURATION
0x826	CORACQ_PRM_CAM_TRIGGER_ENABLE
0x827	CORACQ_PRM_CAM_RESET_ENABLE
0x828	CORACQ_PRM_OUTPUT_FORMAT

0x829-0x82b	Reserved
0x82c	CORACQ_PRM_OUTPUT_ENABLE (obsolete) use CORACQ_PRM_EXT_TRIGGER_ENABLE
0x82d	CORACQ_PRM_VIC_NAME
0x82e	CORACQ_PRM_LUT_MAX
0x82f	CORACQ_PRM_EXT_TRIGGER_DETECTION
0x830	CORACQ_PRM_DC_REST_START
0x831	CORACQ_PRM_DC_REST_WIDTH
0x832	CORACQ_PRM_LUT_FORMAT
0x833	CORACQ_PRM_VSYNC_REF
0x834	CORACQ_PRM_HSYNC_REF
0x835	CORACQ_PRM_LINE_INTEGRATE_ENABLE
0x836	CORACQ_PRM_LINE_INTEGRATE_DURATION
0x837	CORACQ_PRM_LINE_TRIGGER_ENABLE
0x838	CORACQ_PRM_EXT_FRAME_TRIGGER_ENABLE
0x839	CORACQ_PRM_EXT_FRAME_TRIGGER_DETECTION
0x83a	CORACQ_PRM_EXT_LINE_TRIGGER_ENABLE
0x83b	CORACQ_PRM_EXT_LINE_TRIGGER_DETECTION
0x83c	CORACQ_PRM_SNAP_COUNT
0x83d	CORACQ_PRM_INT_LINE_TRIGGER_ENABLE
0x83e	CORACQ_PRM_INT_LINE_TRIGGER_FREQ
0x83f	CORACQ_PRM_LINESCAN_DIRECTION_OUTPUT
0x840	CORACQ_PRM_BIT_ORDERING
0x841	CORACQ_PRM_EXT_TRIGGER_LEVELCORACQ_PRM_EXT_TRIGGER_LEVEL
0x842	CORACQ_PRM_STROBE_LEVE
0x843	CORACQ_PRM_EXT_FRAME_TRIGGER_LEVEL
0x844	CORACQ_PRM_EXT_LINE_TRIGGER_LEVEL
0x845	CORACQ_PRM_INT_LINE_TRIGGER_FREQ_MIN
0x846	CORACQ_PRM_INT_LINE_TRIGGER_FREQ_MAX
0x847	CORACQ_PRM_MASTER_MODE
0x848	CORACQ_PRM_MASTER_MODE_HSYNC_POLARITY
0x849	CORACQ_PRM_MASTER_MODE_VSYNC_POLARITY
0x84a	CORACQ_PRM_SHAFT_ENCODER_DROP
0x84b	CORACQ_PRM_SHAFT_ENCODER_ENABLE
0x84c	CORACQ_PRM_EXT_TRIGGER_FRAME_COUNT
0x84d	CORACQ_PRM_INT_FRAME_TRIGGER_ENABLE
0x84e	CORACQ_PRM_INT_FRAME_TRIGGER_FREQ
0x84f	CORACQ_PRM_SHARED_EXT_TRIGGER
0x850	CORACQ_PRM_SHARED_CAM_RESET
0x851	CORACQ_PRM_SHARED_CAM_TRIGGER
0x852	CORACQ_PRM_SHARED_TIME_INTEGRATE
0x853	CORACQ_PRM_SHARED_FRAME_INTEGRATE
0x854	CORACQ_PRM_SHARED_STROBE
0x855	CORACQ_PRM_STROBE_DELAY_2
0x856	CORACQ_PRM_FRAME_LENGTH
0x857	CORACQ_PRM_FLIP
0x858	CORACQ_PRM_SHARPNESS
0x859	CORACQ_PRM_EXT_TRIGGER_DURATION

0x85a	CORACQ_PRM_TIME_INTEGRATE_DELAY
0x85b	CORACQ_PRM_CAM_RESET_DELAY
0x85c	CORACQ_PRM_CAM_TRIGGER_DELAY
0x85d	CORACQ_PRM_SHAFT_ENCODER_LEVEL
0x85e	CORACQ_PRM_WEN_ENABLE
0x85f	CORACQ_PRM_LUT_NENTRIES
0x860	CORACQ_PRM_EXT_FRAME_TRIGGER_SOURCE
0x861	CORACQ_PRM_EXT_LINE_TRIGGER_SOURCE
0x862	CORACQ_PRM_EXT_TRIGGER_SOURCE
0x863	CORACQ_PRM_SHAFT_ENCODER_MULTIPLY
0x864	CORACQ_PRM_PLANAR_INPUT_SOURCES
0x865	CORACQ_PRM_EXT_TRIGGER_DELAY
0x866	CORACQ_PRM_EXT_TRIGGER_DELAY_TIME_BASE
0x867	CORACQ_PRM_COLOR_DECODER_ENABLE
0x868	CORACQ_PRM_COLOR_DECODER_METHOD
0x869	CORACQ_PRM_WB_GAIN_RED
0x86a	CORACQ_PRM_WB_GAIN_GREEN
0x86b	CORACQ_PRM_WB_GAIN_BLUE
0x86c	CORACQ_PRM_WB_OFFSET_RED
0x86d	CORACQ_PRM_WB_OFFSET_GREEN
0x86e	CORACQ_PRM_WB_OFFSET_BLUE
0x86f	CORACQ_PRM_CAM_CONTROL_PULSE0_HD_ALIGN
0x870	CORACQ_PRM_CAM_CONTROL_PULSE1_HD_ALIGN
0x871	CORACQ_PRM_EXT_TRIGGER_IGNORE_DELAY
0x872	CORACQ_PRM_BOARD_SYNC_OUTPUT1_SOURCE
0x873	CORACQ_PRM_BOARD_SYNC_OUTPUT2_SOURCE
0x874	CORACQ_PRM_FIX_FILTER_SELECTOR_STR
0x875	CORACQ_PRM_EXT_LINE_TRIGGER_SOURCE_STR
0x876	CORACQ_PRM_EXT_TRIGGER_SOURCE_STR
0x877	CORACQ_PRM_VERTICAL_TIMEOUT_DELAY
0x878	CORACQ_PRM_BAYER_DECODER_SATURATION_FACTOR
0x879	CORACQ_PRM_BAYER_DECODER_SATURATION_WEIGHT_RED
0x87a	CORACQ_PRM_BAYER_DECODER_SATURATION_WEIGHT_GREEN
0x87b	CORACQ_PRM_BAYER_DECODER_SATURATION_WEIGHT_BLUE
0x87c	CORACQ_PRM_POCL_ENABLE
0x87d	CORACQ_PRM_CROP_ACTIVATION
0x87e	CORACQ_PRM_SHAFT_ENCODER_SOURCE
0x87f	CORACQ_PRM_SHAFT_ENCODER_SOURCE_STR
0x880	CORACQ_PRM_SHAFT_ENCODER_DIRECTION
0x881	CORACQ_PRM_LINE_TRIGGER_AUTO_DELAY
0x882	CORACQ_PRM_TIME_STAMP_BASE
0x883	CORACQ_PRM_BOARD_SYNC_OUTPUT1_SOURCE_STR
0x884	CORACQ_PRM_BOARD_SYNC_OUTPUT2_SOURCE_STR
0x885	CORACQ_PRM_SHAFT_ENCODER_ORDER
0x886	CORACQ_PRM_CAM_FRAMES_PER_TRIGGER

## **CORACQ\_PRM\_COLOR\_DECODER\_ENABLE**

<b>Description</b>	Enables or disables the hardware Bayer Decoder of the acquisition device. When enabled, it instructs the acquisition device to use the Bayer Decoder to convert the incoming Bayer video data into the specified output format specified by CORACQ_PRM_OUTPUT_FORMAT.
<b>Type</b>	UINT32
<b>Availability</b>	Onboard hardware Bayer Decoder is supported if the CORACQ_CAP_COLOR_DECODER capability returns TRUE.
<b>Values</b>	TRUE (0x00000001), Enable the Bayer Decoder FALSE (0x00000000), Disable the Bayer Decoder
<b>CVI Entry</b>	[Stream Conditioning] Bayer Decoder Enable

## **CORACQ\_PRM\_COLOR\_DECODER\_METHOD**

<b>Description</b>	Selects the Color Decoder method to apply to convert incoming color (for example Bayer) images into the specified output format.
<b>Type</b>	UINT32
<b>Limits</b>	The parameter value must match one of the supported methods of the acquisition device given by CORACQ_CAP_COLOR_DECODER_METHOD. The capability returns the ORed combination of all supported values.
<b>Values</b>	<p>CORACQ_VAL_COLOR_DECODER_METHOD_1 Technique based on bilinear interpolation. Fast, but tends to smooth the edges of the image.</p> <p>CORACQ_VAL_COLOR_DECODER_METHOD_2 Advanced technique, better for preserving the edges of the image. However, it works well only when the image has a strong content in green. Otherwise, small amounts of noise may be visible within objects.</p> <p>CORACQ_VAL_COLOR_DECODER_METHOD_3 Advanced technique, almost as good as Method 2 for preserving the edges, but independent of the image content in green. Small color artifacts of 1 pixel may be visible at the edges.</p> <p>CORACQ_VAL_COLOR_DECODER_METHOD_4 Technique based on 2x2 interpolation. This is the simplest and fastest algorithm. Compared to a 3x3 kernel, it is better at preserving edge sharpness but introduces a slight jitter in pixel position. In practice it is a good choice for image display but less recommended than 3x3 for accurate image processing.</p> <p>CORACQ_VAL_COLOR_DECODER_METHOD_5 Technique based on a set of linear filters. This method assumes that edges have a much stronger luminance than chrominance component.</p> <p>CORACQ_VAL_COLOR_DECODER_METHOD_7 Support for the Teledyne DALSA Piranha 4 line scan camera color output. If the appropriate camera firmware is loaded, the driver will return this value in the capability CORACQ_CAP_COLOR_DECODER_METHOD.</p>
<b>CVI Entry</b>	[Stream Conditioning] Bayer Decoder Method
<b>Note</b>	Validated only if CORACQ_PRM_COLOR_DECODER_ENABLE is TRUE.

## **CORACQ\_PRM\_BAYER\_DECODER\_SATURATION\_FACTOR**

<b>Description</b>	Adjusts the image saturation after Bayer decoding.
<b>Type</b>	UINT32
<b>Limits</b>	Range limits: CORACQ_CAP_BAYER_DECODER_SATURATION_FACTOR_MIN to CORACQ_CAP_BAYER_DECODER_SATURATION_FACTOR_MAX
<b>Values</b>	$\text{saturationFactor} = \text{CORACQ\_PRM\_BAYER\_DECODER\_SATURATION\_FACTOR} / \text{CORACQ\_CAP\_BAYER\_DECODER\_SATURATION\_FACTOR\_DIVISOR}$ $\text{WeightRed} = \text{CORACQ\_PRM\_BAYER\_DECODER\_SATURATION\_WEIGHT\_RED} / \text{CORACQ\_CAP\_BAYER\_DECODER\_SATURATION\_WEIGHT\_RED\_DIVISOR}$ $\text{WeightGreen} = \text{CORACQ\_PRM\_BAYER\_DECODER\_SATURATION\_WEIGHT\_GREEN} / \text{CORACQ\_CAP\_BAYER\_DECODER\_SATURATION\_WEIGHT\_GREEN\_DIVISOR}$ $\text{WeightBlue} = \text{CORACQ\_PRM\_BAYER\_DECODER\_SATURATION\_WEIGHT\_BLUE} / \text{CORACQ\_CAP\_BAYER\_DECODER\_SATURATION\_WEIGHT\_BLUE\_DIVISOR}$ $\text{mono} = \text{red} * \text{WeightRed} + \text{blue} * \text{WeightBlue} + \text{green} * \text{WeightGreen}$ $\text{red} = \text{red} + (\text{red} - \text{mono}) * \text{saturationFactor} ;$ $\text{green} = \text{green} + (\text{green} - \text{mono}) * \text{saturationFactor} ;$ $\text{blue} = \text{blue} + (\text{blue} - \text{mono}) * \text{saturationFactor} ;$
<b>Availability</b>	Onboard hardware Bayer Decoder is supported if the CORACQ_CAP_COLOR_DECODER capability returns TRUE.  Onboard hardware Bayer Decoder Saturation is supported if the CORACQ_CAP_BAYER_DECODER_SATURATION_FACTOR_MIN is not equal to CORACQ_CAP_BAYER_DECODER_SATURATION_FACTOR_MAX
<b>CVI Entry</b>	[Stream Conditioning] Bayer Decoder Saturation Factor
<b>Note</b>	Validated only if CORACQ_PRM_COLOR_DECODER_ENABLE is TRUE.

## CORACQ\_PRM\_BAYER\_DECODER\_SATURATION\_WEIGHT\_BLUE

**Description** Change the image saturation of the pixel blue component value after Bayer decoding.

**Type** UINT32

**Limits** Range limits: CORACQ\_CAP\_BAYER\_DECODER\_SATURATION\_WEIGHT\_BLUE\_MIN to CORACQ\_CAP\_BAYER\_DECODER\_SATURATION\_WEIGHT\_BLUE\_MAX

**Values**  $\text{saturationFactor} = \text{CORACQ\_PRM\_BAYER\_DECODER\_SATURATION\_FACTOR} / \text{CORACQ\_CAP\_BAYER\_DECODER\_SATURATION\_FACTOR\_DIVISOR}$   
 $\text{WeightRed} = \text{CORACQ\_PRM\_BAYER\_DECODER\_SATURATION\_WEIGHT\_RED} / \text{CORACQ\_CAP\_BAYER\_DECODER\_SATURATION\_WEIGHT\_RED\_DIVISOR}$   
 $\text{WeightGreen} = \text{CORACQ\_PRM\_BAYER\_DECODER\_SATURATION\_WEIGHT\_GREEN} / \text{CORACQ\_CAP\_BAYER\_DECODER\_SATURATION\_WEIGHT\_GREEN\_DIVISOR}$   
 $\text{WeightBlue} = \text{CORACQ\_PRM\_BAYER\_DECODER\_SATURATION\_WEIGHT\_BLUE} / \text{CORACQ\_CAP\_BAYER\_DECODER\_SATURATION\_WEIGHT\_BLUE\_DIVISOR}$

$\text{mono} = \text{red} * \text{WeightRed} + \text{blue} * \text{WeightBlue} + \text{green} * \text{WeightGreen}$

$\text{blue} = \text{blue} + (\text{blue} - \text{mono}) * \text{saturationFactor} ;$

**Availability** Onboard hardware Bayer Decoder is supported if the CORACQ\_CAP\_BAYER\_DECODER capability returns TRUE.

Onboard hardware Bayer Decoder Saturation Weight Blue is supported if the CORACQ\_CAP\_BAYER\_DECODER\_SATURATION\_WEIGHT\_BLUE\_MIN is not equal to CORACQ\_CAP\_BAYER\_DECODER\_SATURATION\_WEIGHT\_BLUE\_MAX

**CVI Entry** [Stream Conditioning]  
Bayer Decoder Weight Blue

**Note** Validated only if CORACQ\_PRM\_COLOR\_DECODER\_ENABLE is TRUE.



## CORACQ\_PRM\_BAYER\_DECODER\_SATURATION\_WEIGHT\_GREEN

**Description** Change the image saturation of the pixel green component value after Bayer decoding.

**Type** UINT32

**Limits** Range limits: CORACQ\_CAP\_BAYER\_DECODER\_SATURATION\_WEIGHT\_GREEN\_MIN to CORACQ\_CAP\_BAYER\_DECODER\_SATURATION\_WEIGHT\_GREEN\_MAX

**Availability** Onboard hardware Bayer Decoder is supported if the CORACQ\_CAP\_BAYER\_DECODER capability returns TRUE.

Onboard hardware Bayer Decoder Saturation Weight Green is supported if the CORACQ\_CAP\_BAYER\_DECODER\_SATURATION\_WEIGHT\_GREEN\_MIN is not equal to CORACQ\_CAP\_BAYER\_DECODER\_SATURATION\_WEIGHT\_GREEN\_MAX

**Values** saturationFactor = CORACQ\_PRM\_BAYER\_DECODER\_SATURATION\_FACTOR / CORACQ\_CAP\_BAYER\_DECODER\_SATURATION\_FACTOR\_DIVISOR

WeightRed = CORACQ\_PRM\_BAYER\_DECODER\_SATURATION\_WEIGHT\_RED / CORACQ\_CAP\_BAYER\_DECODER\_SATURATION\_WEIGHT\_RED\_DIVISOR

WeightGreen = CORACQ\_PRM\_BAYER\_DECODER\_SATURATION\_WEIGHT\_GREEN / CORACQ\_CAP\_BAYER\_DECODER\_SATURATION\_WEIGHT\_GREEN\_DIVISOR

WeightBlue = CORACQ\_PRM\_BAYER\_DECODER\_SATURATION\_WEIGHT\_BLUE / CORACQ\_CAP\_BAYER\_DECODER\_SATURATION\_WEIGHT\_BLUE\_DIVISOR

mono = red \* WeightRed + blue \* WeightBlue + green \* WeightGreen

green = green + (green-mono)\*saturationFactor ;

**CVI Entry** [Stream Conditioning]  
Bayer Decoder Weight Green

**Note** Validated only if CORACQ\_PRM\_COLOR\_DECODER\_ENABLE is TRUE.

### **CORACQ\_PRM\_BAYER\_DECODER\_SATURATION\_WEIGHT\_RED**

**Description** Change the image saturation of the pixel red component value after Bayer decoding.

**Type** UINT32

**Limits** Range limits: CORACQ\_CAP\_BAYER\_DECODER\_SATURATION\_WEIGHT\_RED\_MIN to CORACQ\_CAP\_BAYER\_DECODER\_SATURATION\_WEIGHT\_RED\_MAX

**Availability** Onboard hardware Bayer Decoder is supported if the CORACQ\_CAP\_BAYER\_DECODER capability returns TRUE.

Onboard hardware Bayer Decoder Saturation Weight Red is supported if the CORACQ\_CAP\_BAYER\_DECODER\_SATURATION\_WEIGHT\_RED\_MIN is not equal to CORACQ\_CAP\_BAYER\_DECODER\_SATURATION\_WEIGHT\_RED\_MAX

**Values**  $\text{saturationFactor} = \text{CORACQ\_PRM\_BAYER\_DECODER\_SATURATION\_FACTOR} / \text{CORACQ\_CAP\_BAYER\_DECODER\_SATURATION\_FACTOR\_DIVISOR}$

$\text{WeightRed} = \text{CORACQ\_PRM\_BAYER\_DECODER\_SATURATION\_WEIGHT\_RED} / \text{CORACQ\_CAP\_BAYER\_DECODER\_SATURATION\_WEIGHT\_RED\_DIVISOR}$

$\text{WeightGreen} = \text{CORACQ\_PRM\_BAYER\_DECODER\_SATURATION\_WEIGHT\_GREEN} / \text{CORACQ\_CAP\_BAYER\_DECODER\_SATURATION\_WEIGHT\_GREEN\_DIVISOR}$

$\text{WeightBlue} = \text{CORACQ\_PRM\_BAYER\_DECODER\_SATURATION\_WEIGHT\_BLUE} / \text{CORACQ\_CAP\_BAYER\_DECODER\_SATURATION\_WEIGHT\_BLUE\_DIVISOR}$

$\text{mono} = \text{red} * \text{WeightRed} + \text{blue} * \text{WeightBlue} + \text{green} * \text{WeightGreen}$

$\text{red} = \text{red} + (\text{red} - \text{mono}) * \text{saturationFactor} ;$

**CVI Entry** [Stream Conditioning]  
Bayer Decoder Weight Red

**Note** Validated only if CORACQ\_PRM\_COLOR\_DECODER\_ENABLE is TRUE.

### **CORACQ\_PRM\_WB\_GAIN\_RED**

**Description** Bayer Decoder White Balance Gain for the red channel.

**Type** UINT32

**Limits** Range limits: CORACQ\_CAP\_WB\_GAIN\_MIN .. CORACQ\_CAP\_WB\_GAIN\_MAX.  
A gain of 1 = 100000

**CVI Entry** [Stream Conditioning]  
Bayer Decoder White Balance Gain Red

**Note** Validated only if CORACQ\_PRM\_COLOR\_DECODER\_ENABLE is TRUE.

### **CORACQ\_PRM\_WB\_GAIN\_GREEN**

**Description** Bayer Decoder White Balance Gain for the green channel.

**Type** UINT32

**Limits** Range limits: CORACQ\_CAP\_WB\_GAIN\_MIN .. CORACQ\_CAP\_WB\_GAIN\_MAX.  
A gain of 1 = 100000

**CVI Entry** [Stream Conditioning]  
Bayer Decoder White Balance Gain Green

**Note** Validated only if CORACQ\_PRM\_COLOR\_DECODER\_ENABLE is TRUE.

### **CORACQ\_PRM\_WB\_GAIN\_BLUE**

<b>Description</b>	Bayer Decoder White Balance Gain for the blue channel.
<b>Type</b>	UINT32
<b>Limits</b>	Range limits: CORACQ_CAP_WB_GAIN_MIN .. CORACQ_CAP_WB_GAIN_MAX A gain of 1 = 100000
<b>CVI Entry</b>	[Stream Conditioning] Bayer Decoder White Balance Gain Blue
<b>Note</b>	Validated only if CORACQ_PRM_COLOR_DECODER_ENABLE is TRUE.

### **CORACQ\_PRM\_WB\_OFFSET\_RED**

<b>Description</b>	Bayer Decoder White Balance Offset for the red channel.
<b>Type</b>	INT32
<b>Limits</b>	Range limits: CORACQ_CAP_WB_OFFSET_MIN .. CORACQ_CAP_WB_OFFSET_MAX Offset in gray level units.
<b>CVI Entry</b>	[Stream Conditioning] Bayer Decoder White Balance Offset Red
<b>Note</b>	Validated only if CORACQ_PRM_COLOR_DECODER_ENABLE is TRUE.

### **CORACQ\_PRM\_WB\_OFFSET\_GREEN**

<b>Description</b>	Bayer Decoder White Balance Offset for the green channel.
<b>Type</b>	INT32
<b>Limits</b>	Range limits: CORACQ_CAP_WB_OFFSET_MIN .. CORACQ_CAP_WB_OFFSET_MAX Offset in gray level units.
<b>CVI Entry</b>	[Stream Conditioning] Bayer Decoder White Balance Offset Green
<b>Note</b>	Validated only if CORACQ_PRM_COLOR_DECODER_ENABLE is TRUE.

### **CORACQ\_PRM\_WB\_OFFSET\_BLUE**

<b>Description</b>	Bayer Decoder White Balance Offset for the blue channel.
<b>Type</b>	INT32
<b>Limits</b>	Range limits: CORACQ_CAP_WB_OFFSET_MIN .. CORACQ_CAP_WB_OFFSET_MAX Offset in gray level units.
<b>CVI Entry</b>	[Stream Conditioning] Bayer Decoder White Balance Offset Blue
<b>Note</b>	Validated only if CORACQ_PRM_COLOR_DECODER_ENABLE is TRUE.

## **CORACQ\_PRM\_BIT\_ORDERING**

<b>Description</b>	The camera digital bit ordering.
<b>Type</b>	UINT32
<b>Limits</b>	Applies to digital video acquisition only. This value must match one of the supported capabilities of the acquisition device given by CORACQ_CAP_BIT_ORDERING. The capability returns the ORed combination of all supported values.
<b>Values</b>	CORACQ_VAL_BIT_ORDERING_STD (0x00000001) Standard digital bit ordering.  CORACQ_VAL_BIT_ORDERING_9_10 (0x00000002) For some 10-bit digital cameras, video data bits 9 and 10 are swapped with bits 0 and 1, as required by some 10-bit Kodak camera models.  CORACQ_VAL_BIT_ORDERING_MSB_10 (0x00000004) For some 8-bit digital cameras, video data bits 0-7 connect to the acquisition device input bits 2-9, as required by some Kodak camera models.  CORACQ_VAL_BIT_ORDERING_MSB_12 (0x00000008) For use with 12-bit digital cameras, video data bits 4-11 are directed to the input bits 0-7 of the acquisition device.  CORACQ_VAL_BIT_ORDERING_INVERT (0x00000010) For use with digital cameras, the video data bits are inverted (logical NOT) before going to the acquisition device.
<b>CVI Entry</b>	[Input] Bit Ordering

## **CORACQ\_PRM\_BOARD\_SYNC\_OUTPUT1\_SOURCE\_STR**

<b>Description</b>	Returns a string representation of the currently selected CORACQ_PRM_BOARD_SYNC_OUTPUT1_SOURCE.
<b>Type</b>	CHAR[32]
<b>Values</b>	Null terminated string (up to 32 characters including the Null character)
<b>Limits</b>	None
<b>CVI Entry</b>	[Input] Bit Ordering
<b>Note</b>	Read-only parameter. This parameter is device dependent.

## **CORACQ\_PRM\_BOARD\_SYNC\_OUTPUT2\_SOURCE\_STR**

<b>Description</b>	Returns a string representation of the currently selected CORACQ_PRM_BOARD_SYNC_OUTPUT2_SOURCE.
<b>Type</b>	CHAR[32]
<b>Values</b>	Null terminated string (up to 32 characters including the Null character)
<b>Limits</b>	None
<b>CVI Entry</b>	[Input] Bit Ordering
<b>Note</b>	Read-only parameter. This parameter is device dependent.

### **CORACQ\_PRM\_BRIGHTNESS**

<b>Description</b>	Percentage of brightness to be applied to the composite video signal. Applies to analog video signals only.
<b>Type</b>	INT32
<b>Availability</b>	Available only if CORACQ_CAP_BRIGHTNESS is set to TRUE.
<b>Limits</b>	Range limits: CORACQ_CAP_BRIGHTNESS_MIN to CORACQ_CAP_BRIGHTNESS_MAX. Adjust the parameter by increments of at least CORACQ_CAP_BRIGHTNESS_STEP percent (%) in order for a change to occur in the video signal (10000 = 100%).
<b>CVI Entry</b>	[Signal Conditioning] Brightness

### **CORACQ\_PRM\_BRIGHTNESS\_BLUE**

<b>Description</b>	Percentage of brightness to be applied to the blue video signal. Applies to analog video signals only.
<b>Type</b>	INT32
<b>Availability</b>	Available only if CORACQ_CAP_BRIGHTNESS_BLUE is set to TRUE.
<b>Limits</b>	Range limits: CORACQ_CAP_BRIGHTNESS_BLUE_MIN to CORACQ_CAP_BRIGHTNESS_BLUE_MAX. Adjust the parameter by increments of at least CORACQ_CAP_BRIGHTNESS_BLUE_STEP percent (%) in order for a change to occur in the video signal (10000 = 100%).
<b>CVI Entry</b>	[Signal Conditioning] Brightness Blue

### **CORACQ\_PRM\_BRIGHTNESS\_GREEN**

<b>Description</b>	Percentage of brightness to be applied to the green video signal. Applies to analog video signals only.
<b>Type</b>	INT32
<b>Availability</b>	Available only if CORACQ_CAP_BRIGHTNESS_GREEN is set to TRUE.
<b>Limits</b>	Range limits: CORACQ_CAP_BRIGHTNESS_GREEN_MIN to CORACQ_CAP_BRIGHTNESS_GREEN_MAX. Adjust the parameter by increments of at least CORACQ_CAP_BRIGHTNESS_GREEN_STEP percent (%) in order for a change to occur in the video signal (10000 = 100%).
<b>CVI Entry</b>	[Signal Conditioning] Brightness Green

### **CORACQ\_PRM\_BRIGHTNESS\_RED**

<b>Description</b>	Percentage of brightness to be applied to the red video signal. Applies to analog video signals only.
<b>Type</b>	INT32
<b>Availability</b>	Available only if CORACQ_CAP_BRIGHTNESS_RED is set to TRUE.
<b>Limits</b>	Range limits: CORACQ_CAP_BRIGHTNESS_RED_MIN to CORACQ_CAP_BRIGHTNESS_RED_MAX. Adjust the parameter by increments of at least CORACQ_CAP_BRIGHTNESS_CONTRAST_RED_STEP percent (%) in order for a change to occur in the video signal (10000 = 100%).
<b>CVI Entry</b>	[Signal Conditioning] Brightness Red

### **CORACQ\_PRM\_CAM\_CONTROL\_PULSE0\_HD\_ALIGN**

<b>Description</b>	Specifies if the camera control pulse'0' will be aligned with the master HD.	
<b>Type</b>	UINT32	
<b>Values</b>	CORACQ_VAL_CAM_CONTROL_HD_ALIGN_AUTO (0x00000000)	Device Dependent.
	CORACQ_VAL_CAM_CONTROL_HD_ALIGN_ON (0x00000001)	Pulse 0 aligned with HD
	CORACQ_VAL_CAM_CONTROL_HD_ALIGN_OFF (0x00000002)	Pulse 0 not aligned with HD
<b>Limits</b>	Supported only if CORACQ_CAP_CAM_CONTROL_PULSE0_HD_ALIGN is TRUE.	
<b>CVI Entry</b>	[Control Signals] Camera Control Pulse 0 HD Align	

### **CORACQ\_PRM\_CAM\_CONTROL\_PULSE1\_HD\_ALIGN**

<b>Description</b>	Specifies if the camera control pulse'1' will be aligned with the master HD.	
<b>Type</b>	UINT32	
<b>Values</b>	CORACQ_VAL_CAM_CONTROL_HD_ALIGN_AUTO (0x00000000)	Device Dependent.
	CORACQ_VAL_CAM_CONTROL_HD_ALIGN_ON (0x00000001)	Pulse 1 aligned with HD
	CORACQ_VAL_CAM_CONTROL_HD_ALIGN_OFF (0x00000002)	Pulse 1 not aligned with HD
<b>Limits</b>	Supported only if CORACQ_CAP_CAM_CONTROL_PULSE1_HD_ALIGN is TRUE.	
<b>CVI Entry</b>	[Control Signals] Camera Control Pulse 1 HD Align	

### **CORACQ\_PRM\_CAM\_RESET\_DELAY**

<b>Description</b>	Reset pulse delay (in $\mu$ s).After receiving a trigger pulse (external, internal, or software), the acquisition device will wait for this delay before generating the reset pulse.	
<b>Type</b>	UINT32	
<b>Limits</b>	Range limits: CORACQ_CAP_CAM_RESET_DELAY_MIN to CORACQ_CAP_CAM_RESET_DELAY_MAX.	
<b>CVI Entry</b>	[Control Signals] Camera Reset Delay	
<b>Note</b>	This value is only validated if CORACQ_PRM_CAM_RESET_ENABLE is TRUE.	

### **CORACQ\_PRM\_CAM\_RESET\_ENABLE**

<b>Description</b>	Enables or disables the rest pulse to the camera. Applies to area scan cameras only.	
<b>Type</b>	UINT32	
<b>Availability</b>	Available only if CORACQ_CAP_CAM_RESET is TRUE.	
<b>Values</b>	TRUE (0x00000001)	Enable
	FALSE (0x00000000)	Disable
<b>CVI Entry</b>	[Control Signals] Camera Reset Enable	
<b>Note</b>	This parameter is mutually exclusive with CORACQ_PRM_FRAME_INTEGRATE_ENABLE, CORACQ_PRM_CAM_TRIGGER_ENABLE and CORACQ_PRM_TIME_INTEGRATE_ENABLE.	



### **CORACQ\_PRM\_CAM\_TRIGGER\_DELAY**

<b>Description</b>	Trigger pulse delay (in $\mu$ s). After receiving a trigger pulse (external, internal or software), the acquisition device will wait this delay before generating the trigger pulse.
<b>Type</b>	UINT32
<b>Limits</b>	The value must be in the range CORACQ_CAP_CAM_TRIGGER_DELAY_MIN ... CORACQ_CAP_CAM_TRIGGER_DELAY_MAX.
<b>CVI Entry</b>	[Control Signals] Camera Trigger Delay
<b>Note</b>	This value is only validated if CORACQ_PRM_CAM_TRIGGER_ENABLE is TRUE.

### **CORACQ\_PRM\_CAM\_TRIGGER\_ENABLE**

<b>Description</b>	Enables or disables the frame trigger pulse to the camera. Applies to area scan cameras only.
<b>Type</b>	UINT32
<b>Availability</b>	Available only if CORACQ_CAP_CAM_TRIGGER is TRUE..
<b>Values</b>	TRUE (0x00000001)      Enable FALSE (0x00000000)      Disable
<b>CVI Entry</b>	[Control Signals] Camera Trigger Enable
<b>Note</b>	This parameter is mutually exclusive with CORACQ_PRM_FRAME_INTEGRATE_ENABLE, CORACQ_PRM_CAM_RESET_ENABLE and CORACQ_PRM_TIME_INTEGRATE_ENABLE.

### **CORACQ\_PRM\_CAMSEL**

<b>Description</b>	Numerical value representing the camera selector to acquire from.
<b>Type</b>	UINT32
<b>Limits</b>	If CORACQ_PRM_VIDEO is equal to CORACQ_VAL_VIDEO_MONO: 0 ... CORACQ_CAP_CAMSEL_MONO – 1. Applies to composite cameras.  If CORACQ_PRM_VIDEO is equal to CORACQ_VAL_VIDEO_COLOR : 0 ...CORACQ_CAP_CAMSEL_COLOR - 1. Applies to composite cameras.  If CORACQ_PRM_VIDEO is equal to CORACQ_VAL_VIDEO_YC: 0 ... CORACQ_CAP_CAMSEL_YC – 1. Applies to Y/C cameras.  If CORACQ_PRM_VIDEO is equal to CORACQ_VAL_VIDEO_RGB: 0 ... CORACQ_CAP_CAMSEL_RGB – 1. Applies to RGB cameras.
<b>CVI Entry</b>	[Input] Camera Selector

### **CORACQ\_PRM\_CONTRAST**

<b>Description</b>	Percentage of contrast to be applied to the composite video signal. Applies to analog video signals only.
<b>Type</b>	UINT32
<b>Availability</b>	Available only if CORACQ_CAP_CONTRAST is set to TRUE.
<b>Limits</b>	Range limits: CORACQ_CAP_CONTRAST_MIN to CORACQ_CAP_CONTRAST_MAX.  Adjust the parameter by increments of at least CORACQ_CAP_CONTRAST_STEP percent (%) in order for a change to occur in the video signal (100000 = 100%).
<b>CVI Entry</b>	[Signal Conditioning] Contrast

### **CORACQ\_PRM\_CONTRAST\_BLUE**

<b>Description</b>	Percentage of contrast to be applied to the blue video signal. Applies to analog video signals only.
<b>Type</b>	UINT32
<b>Availability</b>	Available only if CORACQ_CAP_CONTRAST_BLUE is set to TRUE.
<b>Limits</b>	Range Limits: CORACQ_CAP_CONTRAST_BLUE_MIN to CORACQ_CAP_CONTRAST_BLUE_MAX. Adjust the parameter by increments of at least CORACQ_CAP_CONTRAST_BLUE_STEP percent (%) in order for a change to occur in the video signal (100000 = 100%).
<b>CVI Entry</b>	[Signal Conditioning] Contrast Blue

### **CORACQ\_PRM\_CONTRAST\_GREEN**

<b>Description</b>	Percentage of contrast to be applied to the green video signal. Applies to analog video signals only.
<b>Type</b>	UINT32
<b>Availability</b>	Available only if CORACQ_CAP_CONTRAST_GREEN is set to TRUE.
<b>Limits</b>	Range Limits: CORACQ_CAP_CONTRAST_GREEN_MIN to CORACQ_CAP_CONTRAST_GREEN_MAX. Adjust the parameter by increments of at least CORACQ_CAP_CONTRAST_GREEN_STEP percent (%) in order for a change to occur in the video signal (100000 = 100%).
<b>CVI Entry</b>	[Signal Conditioning] Contrast Green

### **CORACQ\_PRM\_CONTRAST\_RED**

<b>Description</b>	Percentage of contrast to be applied to the red video signal. Applies to analog video signals only.
<b>Type</b>	UINT32
<b>Availability</b>	Available only if CORACQ_CAP_CONTRAST_RED is set to TRUE.
<b>Limits</b>	Range limits: CORACQ_CAP_CONTRAST_RED_MIN to CORACQ_CAP_CONTRAST_RED_MAX. Adjust the parameter by increments of at least CORACQ_CAP_CONTRAST_RED_STEP percent (%) in order for a change to occur in the video signal.
<b>CVI Entry</b>	[Signal Conditioning] Contrast Red

### **CORACQ\_PRM\_BOARD\_SYNC\_OUTPUT1\_SOURCE**

<b>Description</b>	Specifies the signal that will be output on board sync output 1. This parameter permits the synchronization of two acquisition devices using a signal from one acquisition device, and synching the second acquisition device with it.
<b>Type</b>	UINT32
<b>Limits</b>	Range Limits: 0 .. CORACQ_CAP_BOARD_SYNC_OUTPUT1_SOURCE – 1. The capability returns the ORed combination of all supported values.
<b>Values</b>	Validated only if CORACQ_PRM_BOARD_SYNC_OUTPUT1_ENABLE is TRUE. A value of 0 disables the output to board sync 1.
<b>CVI Entry</b>	[Control Signals] Board Sync Output 1 Source

## **CORACQ\_PRM\_BOARD\_SYNC\_OUTPUT2\_SOURCE**

<b>Description</b>	Specifies the signal that will be output on board sync output 2. This parameter permits the synchronization of two acquisition devices using a signal from one acquisition device, and syncing the second acquisition device with it.
<b>Type</b>	UINT32
<b>Limits</b>	Range Limits: 0 .. CORACQ_CAP_BOARD_SYNC_OUTPUT2_SOURCE – 1.
<b>Values</b>	Validated only if CORACQ_PRM_BOARD_SYNC_OUTPUT2_ENABLE is TRUE. A value of 0 disables the output to board sync 1.
<b>CVI Entry</b>	[Control Signals] Board Sync Output 2 Source

## **CORACQ\_PRM\_CAM\_FRAMES\_PER\_TRIGGER**

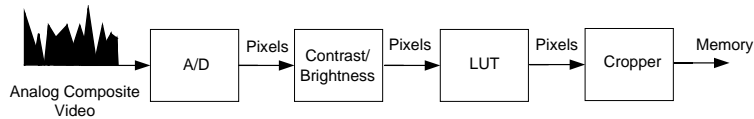
<b>Description</b>	Specifies the number of frames output by the camera per camera trigger. Valid only for area scan cameras.
<b>Type</b>	UINT32
<b>Limits</b>	The value must be in the range 1 .. CORACQ_CAP_CAM_FRAMES_PER_TRIGGER_MAX.
<b>CVI Entry</b>	[Control Signals] Camera Frames Per Trigger
<b>Note</b>	Parameter is only available if CORACQ_CAP_CAM_FRAMES_PER_TRIGGER_MAX is supported. Parameter is only validated if a camera trigger/integrate method is enabled.

## **CORACQ\_PRM\_CROP\_ACTIVATION**

<b>Description</b>	Selects the activation method for the cropper
<b>Type</b>	UINT32
<b>Values</b>	0x00000000 CORACQ_VAL_CROP_ACTIVATION_AUTO Board specific behavior, either of LEVEL or EDGE as described below.  0x00000001 CORACQ_VAL_CROP_ACTIVATION_LEVEL The cropper will only be active during the appropriate signal (HSync/LVAL polarity and VSync/FVAL polarity). In this mode, it is not possible to acquire pixels outside the active region of the video.  0x00000002 CORACQ_VAL_CROP_ACTIVATION_EDGE The cropper will be activated when the appropriate edge is detected. In this mode, it is possible to acquire pixels outside the active region of the video.
<b>Limits</b>	This value must match one of the supported capabilities of the acquisition device given by CORACQ_CAP_ACTIVATION.
<b>CVI Entry</b>	[Stream Conditioning] Crop Activation

## CORACQ\_PRM\_CROP\_HEIGHT

**Description** Cropped height of the acquisition camera image (in lines per frame).



The acquisition device supports vertical cropping if the CORACQ\_CAP\_CROP\_VERT capability returns TRUE.

**Type** UINT32

**Limits** The value must be in the range CORACQ\_CAP\_CROP\_HEIGHT\_MIN to CORACQ\_CAP\_CROP\_HEIGHT\_MAX, and must be a multiple of CORACQ\_CAP\_CROP\_HEIGHT\_MULT.

The value must also be in the range CORACQ\_CAP\_SYNC\_CROP\_HEIGHT\_MIN to CORACQ\_CAP\_SYNC\_CROP\_HEIGHT\_MAX and must be a multiple of CORACQ\_CAP\_SYNC\_CROP\_HEIGHT\_MULT

The value (CORACQ\_PRM\_CROP\_TOP + CORACQ\_PRM\_CROP\_HEIGHT) must be smaller or equal to CORACQ\_PRM\_VACTIVE.

Scale Down limit:

The value  $\text{CORACQ\_PRM\_CROP\_HEIGHT} / (\text{CORACQ\_CAP\_SCALE\_VERT\_MIN\_FACTOR} / \text{CORACQ\_VAL\_SCALE\_FACTOR})$  must be smaller or equal to CORACQ\_PRM\_SCALE\_VERT.

CORACQ\_CAP\_SCALE\_VERT\_MIN\_FACTOR specifies the factor used in calculating the minimum vertical downscaling ratio supported by the acquisition device. The minimum vertical downscaling ratio is equal to  $1/(\text{CORACQ\_CAP\_SCALE\_VERT\_MIN\_FACTOR} / \text{CORACQ\_VAL\_SCALE\_FACTOR})$ .

Scale Up limit:

The value  $\text{CORACQ\_PRM\_CROP\_HEIGHT} * (\text{CORACQ\_CAP\_SCALE\_VERT\_MAX\_FACTOR} / \text{CORACQ\_VAL\_SCALE\_FACTOR})$  must be greater or equal to CORACQ\_PRM\_SCALE\_VERT.

CORACQ\_CAP\_SCALE\_VERT\_MAX\_FACTOR specifies the factor used in calculating the maximum vertical upscaling ratio supported by the acquisition device. The maximum vertical upscaling ratio is equal to  $\text{CORACQ\_CAP\_SCALE\_VERT\_MAX\_FACTOR} / \text{CORACQ\_VAL\_SCALE\_FACTOR}$ .

**CVI Entry** [Stream Conditioning]  
Crop Height

**Note** You should not directly use the function CorAcqSetPrm to set the value of this parameter. This may yield an error condition when it is validated together with the other cropping parameters (CORACQ\_PRM\_CROP\_LEFT, CORACQ\_PRM\_CROP\_TOP, and CORACQ\_PRM\_CROP\_WIDTH).

Instead, you should first retrieve the current VIC parameters using CorAcqGetPrms, then modify the cropping parameters using CorVicSetPrm, and finally apply the new values as a block using CorAcqSetPrms.

## CORACQ\_PRM\_CROP\_LEFT

<b>Description</b>	<p>Number of pixels to crop from the left side of the acquisition camera image.</p> <p>Includes the number of pixels in the horizontal blanking. The horizontal blanking includes the horizontal back porch and the horizontal back invalid parameters. If the horizontal sync reference is set to CORACQ_VAL_SYNC_REF_BEGIN, then the horizontal sync is also included.</p> <p>The acquisition device supports horizontal cropping if the CORACQ_CAP_CROP_HORZ capability returns TRUE.</p>
<b>Type</b>	UINT32
<b>Limits</b>	<p>The value must be in the range CORACQ_CAP_CROP_LEFT_MIN to CORACQ_CAP_CROP_LEFT_MAX, and must be a multiple of CORACQ_CAP_CROP_LEFT_MULT.</p> <p>The value (CORACQ_PRM_CROP_LEFT + CORACQ_PRM_CROP_WIDTH) must be smaller or equal to CORACQ_PRM_HACTIVE.</p> <p>The value (CORACQ_PRM_HBACK_PORCH + CORACQ_PRM_HBACK_INVALID + CORACQ_PRM_CROP_LEFT) must be in the range CORACQ_CAP_SYNC_CROP_LEFT_MIN...CORACQ_CAP_SYNC_CROP_LEFT_MAX, and must be a multiple of CORACQ_CAP_SYNC_CROP_LEFT_MULT.</p> <p>The value (CORACQ_PRM_HBACK_PORCH + CORACQ_PRM_HBACK_INVALID + CORACQ_PRM_CROP_LEFT + CORACQ_PRM_CROP_WIDTH) must be in the range CORACQ_CAP_SYNC_CROP_WIDTH_MIN...CORACQ_CAP_SYNC_CROP_WIDTH_MAX, and must be a multiple of CORACQ_CAP_SYNC_CROP_WIDTH_MULT.</p>
<b>CVI Entry</b>	[Stream Conditioning] Crop Left
<b>Note</b>	<p>You should not directly use the function CorAcqSetPrm to set the value of this parameter. This may yield an error condition when it is validated together with the other cropping parameters (CORACQ_PRM_CROP_HEIGHT, CORACQ_PRM_CROP_TOP, and CORACQ_PRM_CROP_WIDTH).</p> <p>Instead, you should first retrieve the current VIC parameters using CorAcqGetPrms, then modify the cropping parameters using CorVicSetPrm, and finally apply the new values as a block using CorAcqSetPrms.</p>

## CORACQ\_PRM\_CROP\_TOP

<b>Description</b>	<p>Number of lines per acquisition frame to crop from the top of the camera image.</p> <p>It includes the number of lines in the vertical blanking. The vertical blank includes the vertical back porch and the vertical back invalid parameters. If the vertical sync reference is set to CORACQ_VAL_SYNC_REF_BEGIN, then the vertical sync is also included.</p> <p>The acquisition device supports vertical cropping if the CORACQ_CAP_CROP_VERT capability returns TRUE.</p>
<b>Type</b>	UINT32
<b>Limits</b>	<p>The value must be in the range CORACQ_CAP_CROP_TOP_MIN to CORACQ_CAP_CROP_TOP_MAX, and must be a multiple of CORACQ_CAP_CROP_TOP_MULT.</p> <p>The value (CORACQ_PRM_CROP_TOP + CORACQ_PRM_CROP_HEIGHT) must be smaller or equal to CORACQ_PRM_VACTIVE.</p> <p>The value (CORACQ_PRM_VBACK_PORCH + CORACQ_PRM_VBACK_INVALID + CORACQ_PRM_CROP_TOP) must be in the range CORACQ_CAP_SYNC_CROP_TOP_MIN...CORACQ_CAP_SYNC_CROP_TOP_MAX, and must be a multiple of CORACQ_CAP_SYNC_CROP_TOP_MULT.</p> <p>The value (CORACQ_PRM_VBACK_PORCH + CORACQ_PRM_VBACK_INVALID + CORACQ_PRM_CROP_TOP + CORACQ_PRM_CROP_HEIGHT) must be in the range CORACQ_CAP_SYNC_CROP_HEIGHT_MIN...CORACQ_CAP_SYNC_CROP_HEIGHT_MAX, and must be a multiple of CORACQ_CAP_SYNC_CROP_HEIGHT_MULT. See CORACQ_PRM_CROP_HEIGHT for capability information.</p>
<b>CVI Entry</b>	[Stream Conditioning] Crop Top
<b>Note</b>	<p>You should not directly use the CorAcqSetPrm function to set the value of this parameter. This may yield an error condition when it is validated together with the other cropping parameters (CORACQ_PRM_CROP_HEIGHT, CORACQ_PRM_CROP_LEFT, and CORACQ_PRM_CROP_WIDTH).</p> <p>Instead, you should first retrieve the current VIC parameters using CorAcqGetPrms, then modify the cropping parameters using CorVicSetPrm, and finally apply the new values as a block using CorAcqSetPrms. See the <i>Sapera LT Basic Modules Reference Manual</i> for function descriptions referred to in this table.</p>



## CORACQ\_PRM\_CROP\_WIDTH

<b>Description</b>	<p>Cropped width of the acquisition camera image (in pixels).</p> <p>The acquisition device supports horizontal cropping if the CORACQ_CAP_CROP_HORZ capability returns TRUE.</p>
<b>Type</b>	UINT32
<b>Limits</b>	<p>The value must be in the range CORACQ_CAP_CROP_WIDTH_MIN to CORACQ_CAP_CROP_WIDTH_MAX, and must be a multiple of CORACQ_CAP_CROP_WIDTH_MULT.</p> <p>The value must also be in the range CORACQ_CAP_SYNC_CROP_WIDTH_MIN to CORACQ_CAP_SYNC_CROP_WIDTH_MAX and must be a multiple of CORACQ_CAP_SYNC_CROP_WIDTH_MULT. See CORACQ_PRM_CROP_LEFT for capability information.</p> <p>The value (CORACQ_PRM_CROP_LEFT + CORACQ_PRM_CROP_WIDTH) must be smaller or equal to CORACQ_PRM_HACTIVE.</p> <p><u>Scale Down limit:</u> The value <math>\text{CORACQ\_PRM\_CROP\_WIDTH} / (\text{CORACQ\_CAP\_SCALE\_HORZ\_MIN\_FACTOR} / \text{CORACQ\_VAL\_SCALE\_FACTOR})</math> must be smaller or equal to CORACQ_PRM_SCALE_HORZ. CORACQ_CAP_SCALE_HORZ_MIN_FACTOR specifies the factor used in calculating the minimum horizontal downscaling ratio supported by the acquisition device. The minimum horizontal downscaling ratio is equal to <math>1/(\text{CORACQ\_CAP\_SCALE\_HORZ\_MIN\_FACTOR} / \text{CORACQ\_VAL\_SCALE\_FACTOR})</math>.</p> <p><u>Scale Up limit:</u> The value <math>\text{CORACQ\_PRM\_CROP\_WIDTH} * (\text{CORACQ\_CAP\_SCALE\_HORZ\_MAX\_FACTOR} / \text{CORACQ\_VAL\_SCALE\_FACTOR})</math> must be greater or equal to CORACQ_PRM_SCALE_HORZ. CORACQ_CAP_SCALE_HORZ_MAX_FACTOR specifies the factor used in calculating the maximum horizontal upscaling ratio supported by the acquisition device. The maximum horizontal upscaling ratio is equal to <math>\text{CORACQ\_CAP\_SCALE\_HORZ\_MAX\_FACTOR} / \text{CORACQ\_VAL\_SCALE\_FACTOR}</math>.</p>
<b>CVI Entry</b>	[Stream Conditioning] Crop Width
<b>Note</b>	<p>You should not directly use the function CorAcqSetPrm to set the value of this parameter. This may yield an error condition when it is validated together with the other cropping parameters (CORACQ_PRM_CROP_HEIGHT, CORACQ_PRM_CROP_LEFT, and CORACQ_PRM_CROP_TOP).</p> <p>Instead, you should first retrieve the current VIC parameters using CorAcqGetPrms, then modify the cropping parameters using CorVicSetPrm, and finally apply the new values as a block using CorAcqSetPrms.</p>

## CORACQ\_PRM\_DC\_REST\_MODE

<b>Description</b>	DC restoration mode control. Applies to analog video signals only. The acquisition device supports DC restoration if the CORACQ_CAP_DC_REST capability returns TRUE.
<b>Type</b>	UINT32
<b>Limits</b>	This value must match one of the supported capabilities of the acquisition device given by CORACQ_CAP_DC_REST_MODE. The capability returns the ORed combination of all supported values.
<b>Values</b>	<p>CORACQ_VAL_DC_REST_MODE_AUTO (0x00000001) The acquisition device automatically activates or deactivates DC restoration and selects the proper values for the start and width of the sampling pulse. The pulse starting location is set to CORACQ_PRM_HSYNC pixels and the pulse width is set to 0.8 <math>\mu</math>s (expressed in pixels).</p> <p>CORACQ_VAL_DC_REST_MODE_ON (0x00000002) The acquisition device activates DC restoration using user-defined values.</p> <p>CORACQ_VAL_DC_REST_MODE_OFF (0x00000004) The acquisition device deactivates DC restoration.</p>
<b>CVI Entry</b>	[Signal Conditioning] DC Restoration Mode

## CORACQ\_PRM\_DC\_REST\_START

<b>Description</b>	DC restoration sampling pulse start location relative to the horizontal sync, in pixels. Applies to analog video signals only.  The acquisition device supports DC restoration if the CORACQ_CAP_DC_REST capability returns TRUE.
<b>Type</b>	UINT32
<b>Limits</b>	Range limits: CORACQ_CAP_DC_REST_START_MIN to CORACQ_CAP_DC_REST_START_MAX.
<b>CVI Entry</b>	[Signal Conditioning] DC Restoration Start
<b>Note</b>	Validated when CORACQ_PRM_DC_REST_MODE is equal to CORACQ_VAL_DC_REST_MODE_ON.

## CORACQ\_PRM\_DC\_REST\_WIDTH

<b>Description</b>	DC restoration sampling pulse width, in pixels. Applies to analog video signals only.  The acquisition device supports DC restoration if the CORACQ_CAP_DC_REST capability returns TRUE.
<b>Type</b>	UINT32
<b>Limits</b>	Range limits: CORACQ_CAP_DC_REST_WIDTH_MIN to CORACQ_CAP_DC_REST_WIDTH_MAX.
<b>CVI Entry</b>	[Signal Conditioning] DC Restoration Width
<b>Note</b>	Validated only if CORACQ_PRM_DC_REST_MODE is equal to CORACQ_VAL_DC_REST_MODE_ON.

## CORACQ\_PRM\_DECIMATE\_COUNT

<b>Description</b>	The number of fields or frames to decimate per second.
<b>Type</b>	UINT32
<b>Limits</b>	The value must be smaller than the number of acquisition fields or frames per second, depending on the decimation method requested.
<b>CVI Entry</b>	[Stream Conditioning] Decimate Count

## CORACQ\_PRM\_DECIMATE\_METHOD

<b>Description</b>	Field and frame decimation method.  The acquisition device supports field/frame decimation if the CORACQ_CAP_DECIMATE capability returns TRUE.	
<b>Type</b>	UINT32	
<b>Limits</b>	This value must match one of the supported capabilities of the acquisition device given by CORACQ_CAP_DECIMATE_METHOD. The capability returns the ORed combination of all supported values.	
<b>Values</b>	CORACQ_VAL_DECIMATE_DISABLE (0x00000001) CORACQ_VAL_DECIMATE_FIELD (0x00000002) CORACQ_VAL_DECIMATE_FRAME (0x00000004) CORACQ_VAL_DECIMATE_ODD (0x00000008) CORACQ_VAL_DECIMATE_EVEN (0x00000010)	No decimation Decimate fields Decimate frames Decimate odd fields only Decimate even fields only
<b>CVI Entry</b>	[Stream Conditioning] Decimate Method	

## CORACQ\_PRM\_EXT\_FRAME\_TRIGGER\_DETECTION

<b>Description</b>	Defines the signal detected that generates an external frame trigger event to the acquisition device. Applies to linescan cameras only.	
<b>Type</b>	UINT32	
<b>Limits</b>	This value must match one of the supported capabilities of the acquisition device given by CORACQ_CAP_EXT_FRAME_TRIGGER_DETECTION. The capability returns the ORed combination of all supported values.	
<b>Values</b>	CORACQ_VAL_ACTIVE_LOW (0x00000001), Active low signal. Acquisition starts on falling edge of trigger 1 - ends on rising edge of trigger 1 or CORACQ_PRM_CROP_HEIGHT numbers of lines acquired. CORACQ_VAL_ACTIVE_HIGH (0x00000002), Active high signal. Acquisition starts on rising edge of trigger 1 - ends on falling edge of trigger 1 or CORACQ_PRM_CROP_HEIGHT numbers of lines acquired. CORACQ_VAL_RISING_EDGE (0x00000004), Rising signal edge. Acquisition starts on rising signal edge and ends on when CORACQ_PRM_CROP_HEIGHT numbers of lines acquired. CORACQ_VAL_FALLING_EDGE (0x00000008), Falling signal edge. Acquisition starts on falling signal edge and ends on when CORACQ_PRM_CROP_HEIGHT numbers of lines acquired. CORACQ_VAL_BOTH_EDGE (0x00000010), Both signal edges. CORACQ_VAL_DOUBLE_PULSE_RISING_EDGE (0x00000020), Acquisition starts on rising edge of trigger 1 – ends on rising edge of trigger 2. CORACQ_VAL_DOUBLE_PULSE_FALLING_EDGE (0x00000040), Acquisition starts on falling edge of trigger 1 – ends on falling edge of trigger 2.	
<b>CVI Entry</b>	[Control Signals] External Frame Trigger Detection	
<b>Note</b>	Validated only if CORACQ_PRM_EXT_FRAME_TRIGGER_ENABLE is TRUE.	

### **CORACQ\_PRM\_EXT\_FRAME\_TRIGGER\_ENABLE**

<b>Description</b>	Enable or disable external frame trigger on the acquisition device. Applies to linescan cameras only. This feature is used for trigger acquisitions of virtual frames from a linescan camera. For area scan cameras. See CORACQ_PRM_EXT_TRIGGER_ENABLE. The acquisition device may be able to simulate an external trigger. See CORACQ_PRM_EXT_TRIGGER_ENABLE for information concerning the CORACQ_CAP_SOFTWARE_TRIGGER capability.	
<b>Type</b>	UINT32	
<b>Availability</b>	Available only if CORACQ_CAP_EXT_FRAME_TRIGGER is TRUE. This feature is used to trigger the acquisition of a virtual frame from a linescan camera. For area scan cameras, see  CORACQ_PRM_EXT_TRIGGER_ENABLE for information concerning the CORACQ_CAP_EXT_TRIGGER capability.	
<b>Values</b>	TRUE (0x00000001) FALSE (0x00000000)	Enable Disable
<b>CVI Entry</b>	[Control Signals] External Frame Trigger Enable	

### **CORACQ\_PRM\_EXT\_FRAME\_TRIGGER\_LEVEL**

<b>Description</b>	Defines the external frame trigger level connected to the acquisition device. Applies to linescan cameras only.	
<b>Type</b>	UINT32	
<b>Limits</b>	This value must match one of the supported capabilities of the acquisition device given by CORACQ_CAP_EXT_FRAME_TRIGGER_LEVEL. The capability returns the ORed combination of all supported values.	
<b>Values</b>	CORACQ_VAL_LEVEL_TTL (0x00000001) CORACQ_VAL_LEVEL_422 (0x00000002) CORACQ_VAL_LEVEL_LVDS (0x00000004) CORACQ_VAL_LEVEL_24VOLTS (0x00000008) CORACQ_VAL_LEVEL_OPTO (0x00000010) CORACQ_VAL_LEVEL_LVTTL (0x00000020) CORACQ_VAL_LEVEL_12VOLTS (0x00000040)	TTL signal level. RS-422 signal level. LVDS signal level. 24V signal level. Opto-coupled signal level. Low voltage TTL signal level. 12V signal level.
<b>CVI Entry</b>	[Control Signals] External Frame Trigger Level	
<b>Note</b>	Validated only if CORACQ_PRM_EXT_FRAME_TRIGGER_ENABLE is TRUE.	

### **CORACQ\_PRM\_EXT\_FRAME\_TRIGGER\_SOURCE**

<b>Description</b>	Specifies the physical input source the external frame trigger is connected to on the acquisition device, in the case where the acquisition device has more than one input.	
<b>Type</b>	UINT32	
<b>Limits</b>	Range Limits: 0... CORACQ_CAP_EXT_FRAME_TRIGGER_SOURCE – 1 in the case where CORACQ_CAP_EXT_FRAME_TRIGGER_SOURCE is not 0. This capability will have a non-zero value if there is more than one physical input in which to connect an external frame trigger.	
<b>CVI Entry</b>	[Control Signals] External Frame Trigger Source	
<b>Note</b>	Validated only if CORACQ_PRM_EXT_FRAME_TRIGGER_ENABLE is TRUE.	

## CORACQ\_PRM\_EXT\_LINE\_TRIGGER\_DETECTION

<b>Description</b>	Defines the signal detected that generates an external line trigger event to the acquisition device. Applies to linescan cameras only.	
<b>Type</b>	UINT32	
<b>Limits</b>	This value must match one of the supported capabilities of the acquisition device given by CORACQ_CAP_EXT_LINE_TRIGGER_DETECTION. The capability returns the ORed combination of all supported values.	
<b>Values</b>	CORACQ_VAL_RISING_EDGE (0x00000004) CORACQ_VAL_FALLING_EDGE (0x00000008)	Rising signal edge. Falling signal edge.
<b>CVI Entry</b>	[Control Signals] External Line Trigger Detection	
<b>Note</b>	Validated only if CORACQ_PRM_EXT_LINE_TRIGGER_ENABLE is TRUE.	

## CORACQ\_PRM\_EXT\_LINE\_TRIGGER\_ENABLE

<b>Description</b>	Enable or disable external line trigger on the acquisition device. Applies to linescan cameras only. This controls the acquisition line rate of linescan cameras. The acquisition device may be able to simulate an external trigger. See CORACQ_PRM_EXT_TRIGGER_ENABLE for information concerning the CORACQ_CAP_SOFTWARE_TRIGGER capability.	
<b>Type</b>	UINT32	
<b>Availability</b>	Available only if CORACQ_CAP_EXT_LINE_TRIGGER is TRUE.	
<b>Values</b>	TRUE (0x00000001), Enable FALSE (0x00000000), Disable	
<b>CVI Entry</b>	[Control Signals] External Line Trigger Enable	
<b>Note</b>	This parameter is mutually exclusive with CORACQ_PRM_INT_LINE_TRIGGER_ENABLE and CORACQ_PRM_SHAFT_ENCODER_ENABLE.	

## CORACQ\_PRM\_EXT\_LINE\_TRIGGER\_LEVEL

<b>Description</b>	Defines the external line trigger signal level connected to the acquisition device. Applies to linescan cameras only.	
<b>Type</b>	UINT32	
<b>Limits</b>	This value must match one of the supported capabilities of the acquisition device given by CORACQ_CAP_EXT_LINE_TRIGGER_LEVEL. The capability returns the ORed combination of all supported values.	
<b>Values</b>	CORACQ_VAL_LEVEL_TTL (0x00000001) CORACQ_VAL_LEVEL_422 (0x00000002) CORACQ_VAL_LEVEL_LVDS (0x00000004) CORACQ_VAL_LEVEL_24VOLTS (0x00000008) CORACQ_VAL_LEVEL_OPTO (0x00000010) CORACQ_VAL_LEVEL_LVTTL (0x00000020) CORACQ_VAL_LEVEL_12VOLTS (0x00000040)	TTL signal level RS-422 signal level LVDS signal level 24V signal level. Opto-coupled signal level. Low voltage TTL signal level. 12V signal level.
<b>CVI Entry</b>	[Control Signals] External Line Trigger Level	
<b>Note</b>	Validated only if CORACQ_PRM_EXT_LINE_TRIGGER_ENABLE is TRUE.	

### **CORACQ\_PRM\_EXT\_LINE\_TRIGGER\_SOURCE**

<b>Description</b>	Specifies the physical input source the external line trigger is connected to on the acquisition device, in the case where the acquisition device has more than one input.
<b>Type</b>	UINT32
<b>Limits</b>	Range Limits: 0... CORACQ_CAP_EXT_LINE_TRIGGER_SOURCE – 1 in the case where CORACQ_CAP_EXT_LINE_TRIGGER_SOURCE is not 0. This capability will have a non-zero value if there is more than one physical input in which to connect an external line trigger.
<b>CVI Entry</b>	[Control Signals] External Line Trigger Source
<b>Note</b>	Validated only if CORACQ_PRM_EXT_LINE_TRIGGER_ENABLE is TRUE.

### **CORACQ\_PRM\_EXT\_LINE\_TRIGGER\_SOURCE\_STR**

<b>Description</b>	Returns a string representation of the currently selected CORACQ_PRM_EXT_LINE_TRIGGER_SOURCE.
<b>Type</b>	CHAR[32]
<b>Values</b>	Null terminated string (up to 32 characters including the Null character).
<b>Note</b>	Read-only parameter. This parameter is device dependent.

### **CORACQ\_PRM\_EXT\_TRIGGER\_DELAY**

<b>Description</b>	External trigger delay in units specified by CORACQ_PRM_EXT_TRIGGER_DELAY_TIME_BASE. This is the delay between the reception of the trigger signal and the start of image acquisition.
<b>Type</b>	UINT32
<b>Limits</b>	Range limits: CORACQ_CAP_EXT_TRIGGER_DELAY_MIN to CORACQ_CAP_EXT_TRIGGER_DELAY_MAX.
<b>CVI Entry</b>	[Control Signals] External Trigger Delay



## **CORACQ\_PRM\_EXT\_TRIGGER\_DELAY\_TIME\_BASE**

<b>Description</b>	External trigger delay time base
<b>Type</b>	UINT32
<b>Limits</b>	This value must match one of the supported capabilities of the acquisition device given by CORACQ_CAP_EXT_TRIGGER_DELAY_TIME_BASE. The capability returns the ORed combination of all supported values.

### **Values**

CORACQ\_VAL\_TIME\_BASE\_US (0x00000001): Time base is in microseconds.  
CORACQ\_VAL\_TIME\_BASE\_MS (0x00000002): Time base is in milliseconds.  
CORACQ\_VAL\_TIME\_BASE\_LINE\_VALID (0x00000004): Time base is in line counts.  
CORACQ\_VAL\_TIME\_BASE\_LINE\_TRIGGER (0x00000008): Time base is in external line trigger or shaft encoder pulse counts (after drop or/and multiply factors).  
CORACQ\_VAL\_TIME\_BASE\_FRAME\_VALID (0x00000010): Time base is in video frame counts.  
CORACQ\_VAL\_TIME\_BASE\_\_FRAME\_TRIGGER (0x00000020): the time base is the external frame trigger.  
CORACQ\_VAL\_TIME\_BASE\_SHAFT\_ENCODER (0x00000040): the time base is the shaft encoder input (before drop or/and multiply factors).  
CORACQ\_VAL\_TIME\_BASE\_NS (0x00000080), the time base is in nanoseconds  
CORACQ\_VAL\_TIME\_BASE\_PIXEL\_CLK (0x00000100), the time base is in camera pixel clock  
CORACQ\_VAL\_TIME\_BASE\_100NS (0x00000200), the time base is in 100 nanosecond

<b>CVI Entry</b>	[Control Signals] External Trigger Delay Time Base
------------------	-------------------------------------------------------

## **CORACQ\_PRM\_EXT\_TRIGGER\_DETECTION**

<b>Description</b>	Defines the signal detected that generates an external trigger event to the acquisition device.
<b>Type</b>	UINT32
<b>Limits</b>	This value must match one of the supported capabilities of the acquisition device given by CORACQ_CAP_EXT_TRIGGER_DETECTION. The capability returns the ORed combination of all supported values.

<b>Values</b>	CORACQ_VAL_ACTIVE_LOW (0x00000001)	Active low signal
	CORACQ_VAL_ACTIVE_HIGH (0x00000002)	Active high signal
	CORACQ_VAL_RISING_EDGE (0x00000004)	Rising edge of signal
	CORACQ_VAL_FALLING_EDGE (0x00000008)	Falling edge of signal

<b>CVI Entry</b>	[Control Signals] External Trigger Detection
------------------	-------------------------------------------------

<b>Note</b>	Validated only if external trigger is enabled. See CORACQ_PRM_EXT_TRIGGER_ENABLE.
-------------	-----------------------------------------------------------------------------------

## **CORACQ\_PRM\_EXT\_TRIGGER\_DURATION**

<b>Description</b>	Minimum external trigger pulse duration (in $\mu$ s), needed for the pulse to be acknowledged by the acquisition device. If the duration of the pulse is shorter, the pulse will be discarded. This feature is useful for trigger pulse debouncing. If the value is '0', no validation will be done.
--------------------	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

<b>Type</b>	UINT32
-------------	--------

<b>Limits</b>	This value must be in the range CORACQ_CAP_EXT_TRIGGER_DURATION_MIN ... CORACQ_CAP_EXT_TRIGGER_DURATION_MAX. A value of 0 means that the device cannot validate the pulse duration.
---------------	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

<b>CVI Entry</b>	[Control Signals] External Trigger Duration
------------------	------------------------------------------------



## CORACQ\_PRM\_EXT\_TRIGGER\_ENABLE

<b>Description</b>	<p>Replaces CORACQ_PRM_OUTPUT_ENABLE (obsolete). Enables or disables the external trigger feature of the acquisition device. When enabled, the acquisition device acquires frames upon receiving an external trigger.</p> <p>The CorAcqSoftwareTrigger function can be used to simulate a hardware trigger. The CORACQ_CAP_SOFTWARE_TRIGGER capability specifies the software trigger type(s) that can be simulated by the acquisition device. See the CorAcqSoftwareTrigger function in the <i>Sapera LT Basic Modules Reference Manual</i> for further information.</p> <p>The capability returns the ORed combination of all valid values as defined below:</p> <p>CORACQ_VAL_SOFTWARE_TRIGGER_EXT (0x00000001) Simulate an external trigger</p> <p>CORACQ_VAL_SOFTWARE_TRIGGER_EXT_FRAME (0x00000002) Simulate an external frame trigger</p> <p>CORACQ_VAL_SOFTWARE_TRIGGER_EXT_LINE (0x00000004) Simulate an external line trigger</p>	
<b>Type</b>	UINT32	
<b>Availability</b>	Available only if CORACQ_CAP_EXT_TRIGGER is TRUE. Note that CORACQ_CAP_OUTPUT_ENABLE is obsolete.	
<b>Values</b>	CORACQ_VAL_EXT_TRIGGER_OFF (0x00000001) CORACQ_VAL_EXT_TRIGGER_ON (0x00000008)	External Trigger is turned off The acquisition device will acquire images whenever an external trigger signal is detected.
<b>CVI Entry</b>	[Control Signals] External Trigger Enable	
<b>Note</b>	If the CVI entry does not exist or the value is 0, then Output Enable will be used as the default for backward compatibility. See also other parameters in the CORACQ_PRM_EXT_TRIGGER_xxx series.	

## CORACQ\_PRM\_EXT\_TRIGGER\_FRAME\_COUNT

<b>Description</b>	<p>Number of images to acquire upon receiving an external trigger.</p> <p>The acquisition device can acquire more than one frame per trigger if the CORACQ_CAP_EXT_TRIGGER_FRAME_COUNT capability returns TRUE.</p>	
<b>Type</b>	UINT32	
<b>Limits</b>	The value must be in the range: 1... CORACQ_CAP_EXT_TRIGGER_FRAME_COUNT_MAX To grab an infinite number of frames set to CORACQ_PRM_EXT_TRIGGER_FRAME_COUNT_INFINITE	
<b>CVI Entry</b>	[Stream Conditioning] External Trigger Frame Count	
<b>Note</b>	<p>Validated only if external trigger is enabled. See CORACQ_PRM_EXT_TRIGGER_ENABLE.</p> <p>CORACQ_CAP_EXT_TRIGGER_FRAME_COUNT_MAX returns the maximum number of images that can be acquired per trigger that is supported by the device when setting a specific value (CORACQ_PRM_EXT_TRIGGER_FRAME_COUNT_INFINITE is also supported even though it is greater than CORACQ_CAP_EXT_TRIGGER_FRAME_COUNT_MAX).</p> <p>When set to CORACQ_PRM_EXT_TRIGGER_FRAME_COUNT_INFINITE, you must explicitly stop or abort the acquisition.</p>	

## CORACQ\_PRM\_EXT\_TRIGGER\_IGNORE\_DELAY

<b>Description</b>	Following a valid external trigger, this parameter specifies the time delay, in $\mu\text{sec}$ , where if another external trigger occurs, it will be ignored. The start of the delay (time '0') is the end of the next vertical sync for analog cameras, or the beginning of the next frame valid for digital cameras, following the valid external trigger. If the parameter CORACQ_PRM_CAM_CONTROL_DURING_READOUT is FALSE, time '0' will be the end of the last line acquired from a frame. All external triggers received between the valid external trigger and the Time '0' will also be ignored. Applies to area scan cameras only. For linescan cameras, the external trigger invalid region always extends to the end of the next virtual frame valid following a valid external trigger.
<b>Type</b>	UINT32
<b>Values</b>	Numerical value representing the delay in $\mu\text{sec}$ .
<b>Limits</b>	Range Limits: CORACQ_CAP_EXT_TRIGGER_IGNORE_DELAY_MIN ... CORACQ_CAP_EXT_TRIGGER_IGNORE_DELAY_MAX.
<b>CVI Entry</b>	[Control Signals] External Trigger Ignore Delay
<b>Note</b>	Validated only if external trigger is enabled. See CORACQ_PRM_EXT_TRIGGER_ENABLE  See also the related event CORACQ_PRM_EVENT_TYPE: CORACQ_VAL_EVENT_TYPE_EXTERNAL_TRIGGER_IGNORED  For analog cameras, if the WEN signal is used, time '0' will be the start of this WEN signal. For analog cameras, if synching to blanking signals, time '0' will be the end of the blanking signal.

## CORACQ\_PRM\_EXT\_TRIGGER\_LEVEL

<b>Description</b>	Defines the external trigger level connected to the acquisition device.														
<b>Type</b>	UINT32														
<b>Limits</b>	This value must match one of the supported capabilities of the acquisition device given by CORACQ_CAP_EXT_TRIGGER_LEVEL. The capability returns the ORed combination of all supported values.														
<b>Values</b>	<table><tr><td>CORACQ_VAL_LEVEL_TTL (0x00000001)</td><td>TTL signal level</td></tr><tr><td>CORACQ_VAL_LEVEL_422 (0x00000002)</td><td>RS-422 signal level</td></tr><tr><td>CORACQ_VAL_LEVEL_LVDS (0x00000004)</td><td>LVDS signal level</td></tr><tr><td>CORACQ_VAL_LEVEL_24VOLTS (0x00000008)</td><td>24V signal level.</td></tr><tr><td>CORACQ_VAL_LEVEL_OPTO (0x00000010)</td><td>Opto-coupled signal level.</td></tr><tr><td>CORACQ_VAL_LEVEL_LVTTL (0x00000020)</td><td>Low voltage TTL signal level.</td></tr><tr><td>CORACQ_VAL_LEVEL_12VOLTS (0x00000040)</td><td>12V signal level.</td></tr></table>	CORACQ_VAL_LEVEL_TTL (0x00000001)	TTL signal level	CORACQ_VAL_LEVEL_422 (0x00000002)	RS-422 signal level	CORACQ_VAL_LEVEL_LVDS (0x00000004)	LVDS signal level	CORACQ_VAL_LEVEL_24VOLTS (0x00000008)	24V signal level.	CORACQ_VAL_LEVEL_OPTO (0x00000010)	Opto-coupled signal level.	CORACQ_VAL_LEVEL_LVTTL (0x00000020)	Low voltage TTL signal level.	CORACQ_VAL_LEVEL_12VOLTS (0x00000040)	12V signal level.
CORACQ_VAL_LEVEL_TTL (0x00000001)	TTL signal level														
CORACQ_VAL_LEVEL_422 (0x00000002)	RS-422 signal level														
CORACQ_VAL_LEVEL_LVDS (0x00000004)	LVDS signal level														
CORACQ_VAL_LEVEL_24VOLTS (0x00000008)	24V signal level.														
CORACQ_VAL_LEVEL_OPTO (0x00000010)	Opto-coupled signal level.														
CORACQ_VAL_LEVEL_LVTTL (0x00000020)	Low voltage TTL signal level.														
CORACQ_VAL_LEVEL_12VOLTS (0x00000040)	12V signal level.														
<b>CVI Entry</b>	[Control Signals] External Trigger Level														
<b>Note</b>	Validated only if external trigger is enabled. See CORACQ_PRM_EXT_TRIGGER_ENABLE.														

## CORACQ\_PRM\_EXT\_TRIGGER\_SOURCE

<b>Description</b>	Specifies the physical input source the external trigger is connected to on the acquisition device, in the case where the acquisition device has more than one input.
<b>Type</b>	UINT32
<b>Limits</b>	Range Limits: 0... CORACQ_CAP_EXT_TRIGGER_SOURCE – 1 in the case where CORACQ_CAP_EXT_TRIGGER_SOURCE is not 0. This capability will have a non-zero value if more than one physical input to connect an external trigger is present. Use CORACQ_PRM_EXT_TRIGGER_SOURCE_STR to get string descriptions for each possible setting.
<b>CVI Entry</b>	[Control Signals] External Trigger Source
<b>Note</b>	Validated only if CORACQ_PRM_EXT_TRIGGER_ENABLE is TRUE.

### **CORACQ\_PRM\_EXT\_TRIGGER\_SOURCE\_STR**

<b>Description</b>	Returns a string representation of the currently selected CORAQ_PRM_EXT_TRIGGER_SOURCE for area scan cameras and CORACQ_PRM_EXT_FRAME_TRIGGER_SOURCE for linescan cameras.
<b>Type</b>	CHAR[32]
<b>Values</b>	Null terminated string (up to 32 characters including the Null character).
<b>Note</b>	Read-only parameter. This parameter is device dependent.

### **CORACQ\_PRM\_FIX\_FILTER\_ENABLE**

<b>Description</b>	Enable or disable the fixe-frequency filter if available on the acquisition device. Applies to analog video signals only.
<b>Type</b>	UINT32
<b>Availability</b>	Available only if CORACQ_CAP_FIX_FILTER is TRUE.
<b>Values</b>	TRUE (0x00000001), Enable the filter. FALSE 0x00000000), Disable the filter
<b>CVI Entry</b>	[Signal Conditioning] Fix Filter Enable

### **CORACQ\_PRM\_FIX\_FILTER\_SELECTOR**

<b>Description</b>	Selects one of the available fixed-frequency filters. Applies to analog video signals only.
<b>Type</b>	UINT32
<b>Limits</b>	Range Limits: 0... CORACQ_CAP_FIX_FILTER_MAX – 1.
<b>CVI Entry</b>	[Signal Conditioning] Fix Filter Selector
<b>Note</b>	Validated only if CORACQ_PRM_FIX_FILTER_ENABLE is TRUE.

### **CORACQ\_PRM\_FIX\_FILTER\_SELECTOR\_STR**

<b>Description</b>	Returns a string representation of the currently selected CORAQ_PRM_FIX_FILTER_SELECTOR.
<b>Type</b>	CHAR[32]
<b>Values</b>	Null terminated string (up to 32 characters including the Null character).
<b>Note</b>	Read-only parameter. This parameter is device dependent.

### **CORACQ\_PRM\_FLIP**

<b>Description</b>	Flipping mode control.						
<b>Type</b>	UINT32						
<b>Limits</b>	This value must match one of the supported capabilities of the acquisition device given by CORACQ_CAP_FLIP. The capability returns the ORed combination of all supported values.						
<b>Values</b>	<table><tr><td>CORACQ_VAL_FLIP_OFF (0x00000000)</td><td>Incoming lines and frames are not flipped.</td></tr><tr><td>CORACQ_VAL_FLIP_HORZ (0x00000001)</td><td>The acquisition device will flip incoming lines. The right most pixels become the left most pixels</td></tr><tr><td>CORACQ_VAL_FLIP_VERT (0x00000002)</td><td>The acquisition device will flip incoming frames. The bottom lines become the top lines.</td></tr></table>	CORACQ_VAL_FLIP_OFF (0x00000000)	Incoming lines and frames are not flipped.	CORACQ_VAL_FLIP_HORZ (0x00000001)	The acquisition device will flip incoming lines. The right most pixels become the left most pixels	CORACQ_VAL_FLIP_VERT (0x00000002)	The acquisition device will flip incoming frames. The bottom lines become the top lines.
CORACQ_VAL_FLIP_OFF (0x00000000)	Incoming lines and frames are not flipped.						
CORACQ_VAL_FLIP_HORZ (0x00000001)	The acquisition device will flip incoming lines. The right most pixels become the left most pixels						
CORACQ_VAL_FLIP_VERT (0x00000002)	The acquisition device will flip incoming frames. The bottom lines become the top lines.						
<b>CVI Entry</b>	[Stream Conditioning] Flip						

### **CORACQ\_PRM\_FRAME\_INTEGRATE\_COUNT**

<b>Description</b>	Number of frames to integrate. Applies to area scan cameras only.
<b>Type</b>	UINT32
<b>Limits</b>	The value is limited to 1... CORACQ_CAP_FRAME_INTEGRATE_COUNT_MAX.
<b>CVI Entry</b>	[Control Signals] Frame Integrate Count
<b>Note</b>	Validated only if CORACQ_PRM_FRAME_INTEGRATE_ENABLE is TRUE.

### **CORACQ\_PRM\_FRAME\_INTEGRATE\_ENABLE**

<b>Description</b>	Enables or disables frame integration control. Applies to area scan cameras only.
<b>Type</b>	UINT32
<b>Availability</b>	Available only if CORACQ_CAP_FRAME_INTEGRATE is TRUE.
<b>Values</b>	TRUE (0x00000001)                      Enable frame integration control. FALSE (0x00000000)                    Disable frame integration control.
<b>CVI Entry</b>	[Control Signals] Frame Integrate Enable
<b>Note</b>	This parameter is mutually exclusive with CORACQ_PRM_CAM_RESET_ENABLE, CORACQ_PRM_CAM_TRIGGER_ENABLE and CORACQ_PRM_TIME_INTEGRATE_ENABLE.

### **CORACQ\_PRM\_FRAME\_LENGTH**

<b>Description</b>	Specifies if the image output by the acquisition device have a fixed or variable frame length.
<b>Type</b>	UINT32
<b>Limits</b>	This value must match one of the supported capabilities of the acquisition device given by CORACQ_CAP_FRAME_LENGTH. The capability returns the ORed combination of all supported values.
<b>Values</b>	CORACQ_VAL_FRAME_LENGTH_FIX (0x00000001)                      Fixed length images CORACQ_VAL_FRAME_LENGTH_VARIABLE (0x00000002)                      Variable length images
<b>CVI Entry</b>	[Stream Conditioning] Frame Length

### **CORACQ\_PRM\_HSYNC\_REF**

<b>Description</b>	Defines the horizontal sync reference edge used for horizontal timing.
<b>Type</b>	UINT32
<b>Limits</b>	This value must match one of the supported capabilities of the acquisition device given by CORACQ_CAP_HSYNC_REF. The horizontal sync reference is used as the starting point when counting the pixels in a line. Selecting the reference as the end of the sync is useful when dealing with a sync that might be variable. This is usually the case when time-integrating a video signal. The capability returns the ORed combination of all supported values.
<b>Values</b>	CORACQ_VAL_SYNC_REF_BEGIN (0x00000001)    Beginning of horizontal sync. CORACQ_VAL_SYNC_REF_END (0x00000002)      End of horizontal sync. CORACQ_VAL_SYNC_REF_HV_DEPENDENT (0x00000004)      Horizontal and Vertical sync reference are dependent on if the acquisition device grabs analog or digital video.
<b>CVI Entry</b>	[Stream Conditioning] Horizontal Sync Reference

## CORACQ\_PRM\_HUE

<b>Description</b>	Hue control: Phase change in degrees applied to the hue control. Applies only to NTSC analog color video signals (composite or Y/C).
<b>Type</b>	INT32
<b>Limits</b>	Range: CORACQ_CAP_HUE_MIN to CORACQ_CAP_HUE_MAX. Adjust the parameter by increments of at least CORACQ_CAP_HUE_STEP percent (%) in order for a change to occur in the video signal.
<b>Availability</b>	Available only if CORACQ_CAP_HUE is set to TRUE.
<b>CVI Entry</b>	[Signal Conditioning] Hue

## CORACQ\_PRM\_INT\_FRAME\_TRIGGER\_ENABLE

<b>Description</b>	Enable/disable the acquisition device's internal frame trigger feature. Applies to area scan cameras only.
<b>Type</b>	UINT32
<b>Availability</b>	Available only if CORACQ_CAP_INT_FRAME_TRIGGER is TRUE.
<b>Values</b>	TRUE (0x00000001)      Enable FALSE (0x00000000)      Disable
<b>CVI Entry</b>	[Control Signals] Internal Frame Trigger Enable
<b>Note</b>	Controls the rate that video frames are triggered and acquired.

## CORACQ\_PRM\_INT\_FRAME\_TRIGGER\_FREQ

<b>Description</b>	Internal frame trigger frequency in milli-Hz, output by the acquisition device. Applies to area scan cameras only.
<b>Type</b>	UINT32
<b>Limits</b>	Range limits: CORACQ_CAP_INT_FRAME_TRIGGER_FREQ_MIN ... CORACQ_CAP_INT_FRAME_TRIGGER_FREQ_MAX.
<b>CVI Entry</b>	[Control Signals] Internal Frame Trigger Freq
<b>Note</b>	Validated only if CORACQ_PRM_INT_FRAME_TRIGGER_ENABLE is TRUE.

## CORACQ\_PRM\_INT\_LINE\_TRIGGER\_ENABLE

<b>Description</b>	Enable/disable the acquisition device's internal line trigger feature. Applies to linescan cameras only.
<b>Type</b>	UINT32
<b>Availability</b>	Available only if CORACQ_CAP_INT_LINE_TRIGGER is TRUE. This feature is used when the acquisition device itself triggers lines out of a camera.
<b>Values</b>	TRUE (0x00000001)      Enable FALSE (0x00000000)      Disable
<b>CVI Entry</b>	[Control Signals] Internal Line Trigger Enable
<b>Note</b>	Controls the rate video lines are triggered and acquired. This parameter is mutually exclusive with CORACQ_PRM_EXT_LINE_TRIGGER_ENABLE and CORACQ_PRM_SHAFT_ENCODER_ENABLE.

### **CORACQ\_PRM\_INT\_LINE\_TRIGGER\_FREQ**

<b>Description</b>	Frequency (in Hz) of the internal line trigger signal output by the acquisition device. Applies to linescan cameras only.
<b>Type</b>	UINT32
<b>Limits</b>	Acquisition device range limits: CORACQ_PRM_INT_LINE_TRIGGER_FREQ_MIN to CORACQ_PRM_INT_LINE_TRIGGER_FREQ_MAX.  Camera range limits: CORACQ_PRM_CAM_LINE_TRIGGER_FREQ_MIN to CORACQ_PRM_CAM_LINE_TRIGGER_FREQ_MAX.
<b>CVI Entry</b>	[Control Signals] Internal Line Trigger Freq
<b>Note</b>	Validated only if CORACQ_PRM_INT_LINE_TRIGGER_ENABLE is TRUE.

### **CORACQ\_PRM\_INT\_LINE\_TRIGGER\_FREQ\_MAX**

<b>Description</b>	Maximum frequency (in Hz) of the internal line trigger signal output by the acquisition device. Applies to linescan cameras only.
<b>Type</b>	UINT32
<b>CVI Entry</b>	None
<b>Note</b>	Read-only parameter. This parameter may be dependent on the pixel clock setting. Always read the parameter after setting the required pixel clock.

### **CORACQ\_PRM\_INT\_LINE\_TRIGGER\_FREQ\_MIN**

<b>Description</b>	Minimum frequency (in Hz) of the internal line trigger signal output by the acquisition device. Applies to linescan cameras only.
<b>Type</b>	UINT32
<b>CVI Entry</b>	None
<b>Note</b>	Read-only parameter. This parameter may be dependent on the pixel clock setting. Always read the parameter after setting the required pixel clock.

### **CORACQ\_PRM\_LINE\_INTEGRATE\_DURATION**

<b>Description</b>	Line integrate pulse width in units specified by CORACQ_PRM_LINE_INTEGRATE_TIME_BASE. In the case where the units are in CORACQ_VAL_TIME_BASE_NS, the CORACQ_CAP_LINE_INTEGRATE_TIME_BASE_MULT returns the resolution of the time base, in nsec. Applies to linescan cameras only.
<b>Type</b>	UINT32
<b>Limits</b>	Range limits: CORACQ_CAP_LINE_INTEGRATE_DURATION_MIN... CORACQ_CAP_LINE_INTEGRATE_DURATION_MAX.
<b>CVI Entry</b>	[Control Signals] Line Integrate Duration
<b>Note</b>	Validated only if CORACQ_PRM_LINE_INTEGRATE_ENABLE is TRUE.

### **CORACQ\_PRM\_LINE\_INTEGRATE\_ENABLE**

<b>Description</b>	Enable or disable the line integration control signal to the camera. Applies to linescan cameras only.
<b>Type</b>	UINT32
<b>Availability</b>	Available only if CORACQ_CAP_LINE_INTEGRATE is TRUE.
<b>Values</b>	TRUE (0x00000001)      Enable line integration pulse. FALSE (0x00000000)      Disable line integration pulse.
<b>CVI Entry</b>	[Control Signals] Line Integrate Enable
<b>Note</b>	This parameter is mutually exclusive with CORACQ_PRM_LINE_TRIGGER_ENABLE.



### **CORACQ\_PRM\_LINE\_INTEGRATE\_TIME\_BASE**

<b>Description</b>	Time based used by all Line Integrate and Line Trigger delay/duration parameters.
<b>Type</b>	UINT32
<b>Limits</b>	This value must match one of the supported capabilities of the acquisition device given by CORACQ_CAP_LINE_INTEGRATE_TIME_BASE. The capability returns the ORed combination of all supported values.
<b>Values</b>	CORACQ_VAL_TIME_BASE_NS (0x00000080), the time base is in nano-seconds CORACQ_VAL_TIME_BASE_PIXEL_CLK (0x00000100), the time base is in pixel clock ticks
<b>CVI Entry</b>	[Control Signals] Line Integrate Time Base

### **CORACQ\_PRM\_LINE\_TRIGGER\_ENABLE**

<b>Description</b>	Enable or disable the line trigger signal pulse to the camera. Applies to linescan cameras only.
<b>Type</b>	UINT32
<b>Availability</b>	Available only if CORACQ_CAP_LINE_TRIGGER is TRUE.
<b>Values</b>	TRUE (0x00000001)                      Enable FALSE (0x00000000)                    Disable
<b>CVI Entry</b>	[Control Signals] Line Trigger Enable
<b>Note</b>	This parameter is mutually exclusive with CORACQ_PRM_LINE_INTEGRATE_ENABLE.

### **CORACQ\_PRM\_LINE\_TRIGGER\_AUTO\_DELAY**

<b>Description</b>	Enables delaying line triggers to a camera based on the selected method. Delaying a line trigger is used to avoid over-triggering a camera.
<b>Type</b>	UINT32
<b>Limits</b>	This value must match one of the supported capabilities of the acquisition device given by CORACQ_CAP_LINE_TRIGGER_AUTO_DELAY. The capability returns the ORed combination of all supported values.
<b>Values</b>	CORACQ_VAL_LINE_TRIGGER_AUTO_DELAY_DISABLE                      (0x00000000) No delays will be added before triggering a line from a camera.  CORACQ_VAL_LINE_TRIGGER_AUTO_DELAY_END_OF_LVAL (0x00000001) If the end of the LVAL from a previous line trigger has not yet been received, the acquisition device will delay the line trigger to the camera until it receives this LVAL.  CORACQ_VAL_LINE_TRIGGER_AUTO_DELAY_FREQ_MAX                      (0x00000002). If the time between 2 consecutive line triggers is shorter than the maximum frequency specified by CORACQ_PRM_CAM_LINE_TRIGGER_FREQ_MAX, the acquisition device will delay the line trigger to match the maximum frequency line trigger specified.
<b>CVI Entry</b>	[Control Signals] Line Trigger Auto Delay
<b>Note</b>	While waiting on one of the conditions, if another line trigger is requested, the CORACQ_VAL_EVENT_TYPE_LINE_TRIGGER_TOO_FAST event will be received if enabled. Validated only if CORACQ_PRM_LINE_TRIGGER_ENABLE or CORACQ_PRM_LINE_INTEGRATE_ENABLE is TRUE.



### CORACQ\_PRM\_LINESCAN\_DIRECTION\_OUTPUT

<b>Description</b>	Linescan direction control. Applies to linescan cameras only.	
<b>Type</b>	UINT32	
<b>Limits</b>	Value can only be set to CORACQ_VAL_LINESCAN_DIRECTION_REVERSE if CORACQ_CAP_LINESCAN_DIRECTION is TRUE and CORACQ_PRM_LINESCAN_DIRECTION is TRUE. For Teledyne DALSA cameras, this control is called the TDI scan direction.	
<b>Values</b>	CORACQ_VAL_LINESCAN_DIRECTION_FORWARD (0x00000001)	Forward direction.
	CORACQ_VAL_LINESCAN_DIRECTION_REVERSE (0x00000002)	Reverse direction.
<b>CVI Entry</b>	[Control Signals] LineScan Direction Output	

### CORACQ\_PRM\_LUT\_ENABLE

<b>Description</b>	Enable or disable the input LUT.	
<b>Type</b>	UINT32	
<b>Availability</b>	At least one LUT is available if CORACQ_CAP_LUT is TRUE. CORACQ_CAP_LUT_ENABLE will then return TRUE if it can be enabled/disabled.	
<b>Values</b>	TRUE (0x00000001)	Enable the input LUT.
	FALSE (0x00000000)	Disable the input LUT.
<b>CVI Entry</b>	[Stream Conditioning] Lut Enable	
<b>Note</b>	The LUT cannot be disabled on some acquisition devices.	
<b>Description</b>	Input LUT format based on the current pixel depth and output format.	
<b>Type</b>	UINT32	
<b>CVI Entry</b>	None	
<b>Values</b>	Possible values are of the type CORLUT_VAL_FORMAT_ and must match the possible values as defined by the CORACQ_CAP_PIXEL_DEPTH capability that specifies the number of bits per pixel per tap supported by the acquisition device.	
<b>Note</b>	Read-only parameter. This parameter may depend on CORACQ_PRM_PIXEL_DEPTH and CORACQ_PRM_OUTPUT_FORMAT.	

### CORACQ\_PRM\_LUT\_FORMAT

<b>Description</b>	Input LUT format based on the current pixel depth and output format.	
<b>Type</b>	UINT32	
<b>CVI Entry</b>	None	
<b>Values</b>	Possible values are of the type CORLUT_VAL_FORMAT_ and must match the possible values as defined by the CORACQ_CAP_PIXEL_DEPTH capability that specifies the number of bits per pixel per tap supported by the acquisition device.	
<b>Note</b>	Read-only parameter. This parameter may depend on CORACQ_PRM_PIXEL_DEPTH and CORACQ_PRM_OUTPUT_FORMAT.	

### CORACQ\_PRM\_LUT\_MAX

<b>Description</b>	Maximum number of LUTs available based on the current pixel depth and output format.	
<b>Type</b>	UINT32	
<b>CVI Entry</b>	None	
<b>Note</b>	Read-only parameter. This parameter may depend on CORACQ_PRM_PIXEL_DEPTH and CORACQ_PRM_OUTPUT_FORMAT.	

### CORACQ\_PRM\_LUT\_NENTRIES

<b>Description</b>	The number of elements in the input lookup table.
<b>Type</b>	UINT32
<b>Values</b>	Usually ranges from 256 to 65536.
<b>CVI Entry</b>	None
<b>Note</b>	Read only parameter. This parameter may depend on CORACQ_PRM_PIXEL_DEPTH and CORACQ_PRM_OUTPUT_FORMAT.

### CORACQ\_PRM\_LUT\_NUMBER

<b>Description</b>	Selects which Lute use.
<b>Type</b>	UINT32
<b>Limits</b>	The value must be in the range 0...CORACQ_PRM_LUT_MAX – 1.
<b>CVI Entry</b>	[Stream Conditioning] Lut Number
<b>Note</b>	Validated only if CORACQ_PRM_LUT_ENABLE is TRUE.

### CORACQ\_PRM\_MASTER\_MODE

<b>Description</b>	Specifies if the acquisition device drives the horizontal and/or the vertical sync of the camera.
<b>Type</b>	UINT32
<b>Limits</b>	This value must match one of the supported capabilities of the acquisition device given by CORACQ_CAP_MASTER_MODE.
<b>Values</b>	CORACQ_VAL_MASTER_MODE_DISABLE (0x00000000), Master mode disabled. CORACQ_VAL_MASTER_MODE_HSYNC_VSYNC (0x00000001), The acquisition device drives the horizontal and vertical sync of the camera. CORACQ_VAL_MASTER_MODE_HSYNC (0x00000002), The acquisition device drives the horizontal sync of the camera. CORACQ_VAL_MASTER_MODE_VSYNC (0x00000004), The acquisition device drives the vertical sync of the camera.
<b>CVI Entry</b>	[Control Signals] Master Mode

### CORACQ\_PRM\_MASTER\_MODE\_HSYNC\_POLARITY

<b>Description</b>	Specifies the horizontal sync polarity that the acquisition device outputs in master mode.
<b>Type</b>	UINT32
<b>Limits</b>	This value must match one of the supported capabilities of the acquisition device given by CORACQ_CAP_MASTER_MODE_HSYNC_POLARITY. The capability returns the ORed combination of all supported values.
<b>Values</b>	CORACQ_VAL_ACTIVE_LOW (0x00000001)      Horizontal sync is active low. CORACQ_VAL_ACTIVE_HIGH (0x00000002)      Horizontal sync is active high.
<b>CVI Entry</b>	[Control Signals] Master Mode Horizontal Sync Polarity
<b>Note</b>	Validated only if CORACQ_PRM_MASTER_MODE is not equal to CORACQ_VAL_MASTER_MODE_DISABLE.

## **CORACQ\_PRM\_MASTER\_MODE\_VSYNC\_POLARITY**

<b>Description</b>	Specifies the vertical sync polarity that the acquisition device outputs in master mode.	
<b>Type</b>	UINT32	
<b>Limits</b>	This value must match one of the supported capabilities of the acquisition device given by CORACQ_CAP_MASTER_MODE_VSYNC_POLARITY. The capability returns the ORed combination of all supported values.	
<b>Values</b>	CORACQ_VAL_ACTIVE_LOW (0x00000001) CORACQ_VAL_ACTIVE_HIGH (0x00000002)	Vertical sync is active low. Vertical sync is active high.
<b>CVI Entry</b>	[Control Signals] Master Mode Vertical Sync Polarity	
<b>Note</b>	Validated only if CORACQ_PRM_MASTER_MODE is not equal to CORACQ_VAL_MASTER_MODE_DISABLE.	

## **CORACQ\_PRM\_OUTPUT\_ENABLE (obsolete)**

<b>Description</b>	Video data output mode. Obsolete, use CORACQ_PRM_EXT_TRIGGER_ENABLE.	
<b>Type</b>	UINT32	
<b>Limits</b>	This value must match one of the supported capabilities of the acquisition device given by CORACQ_CAP_OUTPUT_ENABLE. The capability returns the ORed combination of all supported values.	
<b>Values</b>	CORACQ_VAL_OUTPUT_ENABLE_AUTO (0x00000001), The video data is output whenever a frame has been requested and there is a valid frame.  CORACQ_VAL_OUTPUT_ENABLE_ON (0x00000002), Video data output enabled always. CORACQ_VAL_OUTPUT_ENABLE_OFF (0x00000004), Video data is not output. CORACQ_VAL_OUTPUT_ENABLE_ON_EXTRIG (0x00000008), The video data is output on the next valid frame when a frame has been requested and upon receiving an external trigger signal.	
<b>CVI Entry</b>	[Output] Output Enable	
<b>Note</b>	When using CORACQ_VAL_OUTPUT_ENABLE_ON_EXTRIG, see also CORACQ_PRM_EXT_TRIGGER_DETECTION, CORACQ_PRM_EXT_TRIGGER_FRAME_COUNT and CORACQ_PRM_EXT_TRIGGER_LEVEL.	

## CORACQ\_PRM\_OUTPUT\_FORMAT

<b>Description</b>	Data format output by the acquisition device.
<b>Type</b>	UINT32 [64]
<b>Limits</b>	<p>This value must match one of the supported output formats given by CORACQ_CAP_OUTPUT_FORMAT.</p> <p>This capability returns the different output formats supported by the acquisition device as a UINT32 list. The list terminates upon reaching an output format with a value of 0. An array of at least 64 elements must be allocated to obtain the full list of supported formats.</p>
<b>Values</b>	<p>CORACQ_VAL_OUTPUT_FORMAT_MONO1 CORACQ_VAL_OUTPUT_FORMAT_MONO8 CORACQ_VAL_OUTPUT_FORMAT_MONO16 CORACQ_VAL_OUTPUT_FORMAT_MONO32 CORACQ_VAL_OUTPUT_FORMAT_RGB5551 CORACQ_VAL_OUTPUT_FORMAT_RGB565 CORACQ_VAL_OUTPUT_FORMAT_RGB888 CORACQ_VAL_OUTPUT_FORMAT_RGB8888 CORACQ_VAL_OUTPUT_FORMAT_RGB101010 CORACQ_VAL_OUTPUT_FORMAT_RGB161616 CORACQ_VAL_OUTPUT_FORMAT_RGB16161616 CORACQ_VAL_OUTPUT_FORMAT_RGBP8 CORACQ_VAL_OUTPUT_FORMAT_RGBP16CORACQ_VAL_OUTPUT_FORMAT_RGBR888 CORACQ_VAL_OUTPUT_FORMAT_UYVY CORACQ_VAL_OUTPUT_FORMAT_YUY2 CORACQ_VAL_OUTPUT_FORMAT_YVYU CORACQ_VAL_OUTPUT_FORMAT_YUYV CORACQ_VAL_OUTPUT_FORMAT_Y411 CORACQ_VAL_OUTPUT_FORMAT_Y211 CORACQ_VAL_OUTPUT_FORMAT_HSV CORACQ_VAL_OUTPUT_FORMAT_HSI CORACQ_VAL_OUTPUT_FORMAT_HSIP8 CORACQ_VAL_OUTPUT_FORMAT_BICOLOR88 CORACQ_VAL_OUTPUT_FORMAT_BICOLOR1616</p>
<b>CVI Entry</b>	<p>[Output] Output Format</p>

## CORACQ\_PRM\_PIXEL\_MASK

<b>Description</b>	Defines the pixel mask values. If any mask bits are set to 0, then the corresponding pixel bits are also set to 0.
<b>Type</b>	UINT32
<b>Availability</b>	Available only if CORACQ_CAP_PIXEL_MASK is TRUE.
<b>CVI Entry</b>	<p>[Stream Conditioning] Pixel Mask</p>

## CORACQ\_PRM\_PLANAR\_INPUT\_SOURCES

<b>Description</b>	Specifies which video input sources will be acquired synchronously and transferred to a vertical planar buffer.
<b>Type</b>	UINT32
<b>Availability</b>	Available only if CORACQ_CAP_PLANAR_INPUT_SOURCES is TRUE.
<b>Values</b>	Bit field representing the video input sources that are to be enabled for synchronized acquisition into a vertical planar buffer. The board video input is enabled if the corresponding bit is 1. For example, a value of 0x00000005 indicates that bit 0 and 2 are active, and camera #1 and #3 will be acquired from.
<b>CVI Entry</b>	[Input] Planar Input Sources
<b>Note</b>	The acquisition module might have limitations on which inputs can be acquired synchronously. See the board's User's Manual for more details.

## CORACQ\_PRM\_POCL\_ENABLE

<b>Description</b>	Enable or disable ending power through the camera link cable..
<b>Type</b>	BOOL32
<b>Availability</b>	Available only if CORACQ_CAP_POCL is TRUE.
<b>Values</b>	TRUE (0x00000001)                      Enable camera link power. FALSE (0x00000000)                      Disable camera link power.
<b>CVI Entry</b>	[Control Signal] PoCL Enable
<b>Note</b>	The camera must be PoCL compliant and use a PoCL cable. To validate if PoCL is active when PoCL is enabled, check the CORACQ_PRM_SIGNAL_STATUS for the CORACQ_VAL_SIGNAL_POCL_ACTIVE flag.

## CORACQ\_PRM\_PROG\_FILTER\_ENABLE

<b>Description</b>	Enable or disable the programmable frequency filter. Applies to analog video signals only.
<b>Type</b>	UINT32
<b>Availability</b>	Available only if CORACQ_CAP_PROG_FILTER is TRUE.
<b>Values</b>	TRUE (0x00000001)                      Enable the programmable filter. FALSE (0x00000000)                      Disable the programmable filter.
<b>CVI Entry</b>	[Signal Conditioning] Programmable Filter Enable

## CORACQ\_PRM\_PROG\_FILTER\_FREQ

<b>Description</b>	Programmable filter frequency in Hz. Applies to analog video signals only.
<b>Type</b>	UINT32
<b>Limits</b>	The value must be in the range CORACQ_CAP_PROG_FILTER_FREQ_MIN ... CORACQ_CAP_PROG_FILTER_FREQ_MAX.
<b>CVI Entry</b>	[Signal Conditioning] Programmable Filter Frequency
<b>Note</b>	Validated only if CORACQ_PRM_PROG_FILTER_ENABLE is TRUE.

## CORACQ\_PRM\_SATURATION

<b>Description</b>	Color saturation percentage control applied to analog composite color video signals.
<b>Type</b>	UINT32
<b>Availability</b>	Available only if CORACQ_CAP_SATURATION is set to TRUE
<b>Limits</b>	Range limits: CORACQ_CAP_SATURATION_MIN to CORACQ_CAP_SATURATION_MAX. Adjust the parameter by increments of at least CORACQ_CAP_SATURATION_STEP percent (%) in order for a change to occur in the video signal.
<b>CVI Entry</b>	[Signal Conditioning] Saturation

## CORACQ\_PRM\_SCALE\_HORZ

<b>Description</b>	Number of pixels per line output by the scalar.
<b>Type</b>	UINT32
<b>Limits</b>	The value must be in the range CORACQ_CAP_SCALE_HORZ_MIN to CORACQ_CAP_SCALE_HORZ_MAX, and must be a multiple of CORACQ_CAP_SCALE_HORZ_MULT. <u>Scale Down limit:</u> The value $\text{CORACQ\_PRM\_CROP\_WIDTH} / (\text{CORACQ\_CAP\_SCALE\_HORZ\_MIN\_FACTOR} / \text{CORACQ\_VAL\_SCALE\_FACTOR})$ must be smaller or equal to CORACQ_PRM_SCALE_HORZ. <u>Scale Up limit:</u> The value $\text{CORACQ\_PRM\_CROP\_WIDTH} * (\text{CORACQ\_CAP\_SCALE\_HORZ\_MAX\_FACTOR} / \text{CORACQ\_VAL\_SCALE\_FACTOR})$ must be greater or equal to CORACQ_PRM_SCALE_HORZ. See CORACQ_PRM_CROP_WIDTH for information on both CORACQ_CAP_SCALE_HORZ_MIN_FACTOR and CORACQ_CAP_SCALE_HORZ_MAX_FACTOR.
<b>CVI Entry</b>	[Stream Conditioning] Scale Horizontal
<b>Note</b>	Available only if CORACQ_PRM_SCALE_HORZ_METHOD is not equal to CORACQ_VAL_SCALE_METHOD_DISABLE.

## CORACQ\_PRM\_SCALE\_HORZ\_METHOD

<b>Description</b>	Horizontal scaling method.
<b>Type</b>	UINT32
<b>Limits</b>	This value must match one of the supported capabilities of the acquisition device given by CORACQ_CAP_SCALE_HORZ_METHOD. The capability returns the ORed combination of all supported values.
<b>Values</b>	CORACQ_VAL_SCALE_METHOD_DISABLE (0x00000001), Disable horizontal scaling. CORACQ_VAL_SCALE_METHOD_SIMPLE (0x00000002), Horizontal scaling drops pixels. CORACQ_VAL_SCALE_METHOD_INTERPOLATION (0x00000004), Horizontal scaling interpolates pixels. CORACQ_VAL_SCALE_METHOD_POW2 (0x00000008), Horizontal scaling must be a power of 2.
<b>CVI Entry</b>	[Stream Conditioning] Scale Horizontal Method

## CORACQ\_PRM\_SCALE\_VERT

<b>Description</b>	Number of lines per frame output by the scalar.
<b>Type</b>	UINT32
<b>Limits</b>	<p>The value must be in the range CORACQ_CAP_SCALE_VERT_MIN ... CORACQ_CAP_SCALE_VERT_MAX, and must be a multiple of CORACQ_CAP_SCALE_VERT_MULT.</p> <p><u>Scale Down limit:</u> The value CORACQ_PRM_CROP_HEIGHT / (CORACQ_CAP_SCALE_VERT_MIN_FACTOR / CORACQ_VAL_SCALE_FACTOR) must be smaller or equal to CORACQ_PRM_SCALE_VERT.</p> <p><u>Scale Up limit:</u> The value CORACQ_PRM_CROP_HEIGHT * (CORACQ_CAP_SCALE_VERT_MAX_FACTOR / CORACQ_VAL_SCALE_FACTOR) must be greater or equal to CORACQ_PRM_SCALE_VERT.</p> <p>See CORACQ_PRM_CROP_HEIGHT for information on both CORACQ_CAP_SCALE_VERT_MIN_FACTOR and CORACQ_CAP_SCALE_VERT_MAX_FACTOR.</p>
<b>CVI Entry</b>	[Stream Conditioning] Scale Vertical
<b>Note</b>	Available only if CORACQ_PRM_SCALE_VERT_METHOD is not equal to CORACQ_VAL_SCALE_METHOD_DISABLE.

## CORACQ\_PRM\_SCALE\_VERT\_METHOD

<b>Description</b>	Vertical scaling method.
<b>Type</b>	UINT32
<b>Limits</b>	This value must match one of the supported capabilities of the acquisition device given by CORACQ_CAP_SCALE_VERT_METHOD. The capability returns the ORed combination of all supported values.
<b>Values</b>	<p>CORACQ_VAL_SCALE_METHOD_DISABLE (0x00000001), Disable vertical scaling.</p> <p>CORACQ_VAL_SCALE_METHOD_SIMPLE (0x00000002), Vertical scaling drops lines.</p> <p>CORACQ_VAL_SCALE_METHOD_INTERPOLATION (0x00000004), Vertical scaling interpolates lines.</p> <p>CORACQ_VAL_SCALE_METHOD_POW2 (0x00000008), Vertical scaling must be a power of 2.</p>
<b>CVI Entry</b>	[Stream Conditioning] Scale Vertical Method



## **CORACQ\_PRM\_SHAFT\_ENCODER\_DIRECTION**

<b>Description</b>	Selects the direction of the shaft encoder that increments/decrements the acquisition device encoder counter. Support of dual phase encoders might require that the direction of motion be considered. This is the case where system vibrations and/or conveyor backlash can cause the encoder to momentarily travel backwards. The acquisition device must in those cases count the reverse steps and subtract the forward steps such that only pulses after the reverse count reaches zero are considered valid.
<b>Type</b>	UINT32
<b>Values</b>	<p>CORACQ_VAL_SHAFT_ENCODER_DIRECTION_IGNORED (0x00000000) Do not take into account the shaft encoder direction. All shaft encoder pulses are considered valid.</p> <p>CORACQ_VAL_SHAFT_ENCODER_DIRECTION_FORWARD (0x00000001) Increment the shaft encoder counter when a forward motion is detected. A forward motion is detected when the order of the pulses are A/B.</p> <p>CORACQ_VAL_SHAFT_ENCODER_DIRECTION_REVERSE (0x00000002) Increment the shaft encoder counter when a reverse motion is detected. A reverse motion is detected when the order of the pulses are B/A.</p>
<b>Limits</b>	This value must match one of the supported capabilities of the acquisition device given by CORACQ_CAP_SHAFT_ENCODER_DIRECTION
<b>CVI Entry</b>	[Control Signals] Shaft Encoder Direction
<b>Note</b>	Validated only if CORACQ_PRM_SHAFT_ENCODER_ENABLE is TRUE .

## **CORACQ\_PRM\_SHAFT\_ENCODER\_DROP**

<b>Description</b>	Number of signal edges dropped when video acquisitions are controlled by a shaft encoder. Applies to linescan cameras only.
<b>Type</b>	UINT32
<b>Limits</b>	Range limits CORACQ_CAP_SHAFT_ENCODER_DROP_MIN to CORACQ_CAP_SHAFT_ENCODER_DROP_MAX.
<b>CVI Entry</b>	[Control Signals] Shaft Encoder Pulse Drop
<b>Note</b>	Validated only if CORACQ_PRM_SHAFT_ENCODER_ENABLE is TRUE. For more details about the shaft encoder, see "Shaft Encoder Description".

## **CORACQ\_PRM\_SHAFT\_ENCODER\_ENABLE**

<b>Description</b>	Enable or disable the shaft encoder support of the acquisition device.				
<b>Type</b>	UINT32				
<b>Availability</b>	Available only if CORACQ_CAP_SHAFT_ENCODER is TRUE.				
<b>Values</b>	<table><tr><td>TRUE (0x00000001)</td><td>Enable</td></tr><tr><td>FALSE (0x00000000)</td><td>Disable</td></tr></table>	TRUE (0x00000001)	Enable	FALSE (0x00000000)	Disable
TRUE (0x00000001)	Enable				
FALSE (0x00000000)	Disable				
<b>CVI Entry</b>	[Control Signals] Shaft Encoder Enable				
<b>Note</b>	This parameter is mutually exclusive with CORACQ_PRM_INT_LINE_TRIGGER_ENABLE and CORACQ_PRM_EXT_LINE_TRIGGER_ENABLE. For more details about the shaft encoder, see "Shaft Encoder Description".				

## CORACQ\_PRM\_SHAFT\_ENCODER\_LEVEL

<b>Description</b>	Shaft encoder level fed to the acquisition device. Applies to linescan cameras only.	
<b>Type</b>	UINT32	
<b>Limits</b>	This value must match one of the supported capabilities of the acquisition device given by CORACQ_CAP_SHAFT_ENCODER_LEVEL. The capability returns the ORed combination of all supported values.	
<b>Values</b>	CORACQ_VAL_LEVEL_TTL (0x00000001)	TTL signal.
	CORACQ_VAL_LEVEL_422 (0x00000002)	RS-422 signal.
	CORACQ_VAL_LEVEL_LVDS (0x00000004)	LVDS signal.
	CORACQ_VAL_LEVEL_24VOLTS (0x00000008)	24V signal level.
	CORACQ_VAL_LEVEL_OPTO (0x00000010)	Opto-coupled signal level.
	CORACQ_VAL_LEVEL_LVTTL (0x00000020)	Low voltage TTL signal level.
	CORACQ_VAL_LEVEL_12VOLTS (0x00000040)	12V signal level.
<b>CVI Entry</b>	[Control Signals] Shaft Encoder Level	
<b>Note</b>	Validated only if CORACQ_PRM_SHAFT_ENCODER_ENABLE is TRUE.	

## CORACQ\_PRM\_SHAFT\_ENCODER\_MULTIPLY

<b>Description</b>	Number of signal edges generated for each shaft encoder signal edge, when video acquisitions are controlled by an external shaft encoder trigger. Applies to linescan cameras only.	
<b>Type</b>	UINT32	
<b>Limits</b>	Range limits CORACQ_CAP_SHAFT_ENCODER_MULTIPLY_MIN to CORACQ_CAP_SHAFT_ENCODER_MULTIPLY_MAX by increments specified by CORACQ_CAP_SHAFT_ENCODER_MULTIPLY_STEP.  Adjust the parameter by minimum increments as specified by CORACQ_CAP_SHAFT_ENCODER_MULTIPLY_STEP. This capability is a 32-bit bitfield containing the minimum step (bit0 to bit15) and the step type (linear or exponential, bit16 to bit31).	

Bits 31 - 16	Bits 15 - 0
Step Type	Step Value

The parameter varies as described below:

Step Type	CORACQ_PRM_SHAFT_ENCODER_MULTIPLY
CORSTEP_INCREMENT_LINEAR (0x10000000)	SHAFT_ENCODER_MULTIPLY_MIN + N * <i>step</i>
CORSTEP_INCREMENT_EXPONENTIAL (0x20000000)	SHAFT_ENCODER_MULTIPLY_MIN * <i>step</i> <sup>N</sup> Where N >= 0.

For example, if the CORACQ\_CAP\_SHAFT\_ENCODER\_STEP value is 0x20000002, the step type is CORSTEP\_INCREMENT\_EXPONENTIAL, with a *step* of 2 .If CORACQ\_CAP\_SHAFT\_ENCODER\_MULTIPLY\_MIN = 1, CORACQ\_PRM\_SHAFT\_ENCODER\_MULTIPLY would be 1, 2, 4, 8...

<b>CVI Entry</b>	[Control Signals] Shaft Encoder Pulse Multiply
<b>Note</b>	Validated only if CORACQ_PRM_SHAFT_ENCODER_ENABLE is TRUE. For more details about the shaft encoder, see "Shaft Encoder Description". See your board User's manual for any hardware limitations of this feature.

### **CORACQ\_PRM\_SHAFT\_ENCODER\_ORDER**

<b>Description</b>	Selects the order of the drop/multiply operation of the shaft encoder.
<b>Type</b>	UINT32
<b>Limits</b>	This value must match one of the supported capabilities of the acquisition device given by CORACQ_CAP_SHAFT_ENCODER_ORDER. The capability returns the ORed combination of all supported values.
<b>Values</b>	CORACQ_VAL_SHAFT_ENCODER_ORDER_AUTO: Device Specific CORACQ_VAL_SHAFT_ENCODER_ORDER_DROP_MULTIPLY: Drop-Multiply CORACQ_VAL_SHAFT_ENCODER_ORDER_MULTIPLY_DROP: Multiply-Drop
<b>CVI Entry</b>	[Control Signals] Shaft Encoder Order

### **CORACQ\_PRM\_SHAFT\_ENCODER\_SOURCE**

<b>Description</b>	Specifies the physical input source the shaft encoder is connected to on the acquisition device, in the case where the acquisition device has more than one input.
<b>Type</b>	UINT32
<b>Limits</b>	Range Limits: 0 ... CORACQ_CAP_SHAFT_ENCODER_SOURCE – 1 in the case where CORACQ_CAP_SHAFT_ENCODER_SOURCE is not 0. This capability will have a non-zero value if more than one physical input to connect a shaft encoder is present.
<b>CVI Entry</b>	[Control Signals] Shaft Encoder Source
<b>Note</b>	Validated only if CORACQ_PRM_SHAFT_ENCODER_ENABLE is TRUE.

### **CORACQ\_PRM\_SHAFT\_ENCODER\_SOURCE\_STR**

<b>Description</b>	Returns a string representation of the currently selected CORACQ_PRM_SHAFT_ENCODER_SOURCE
<b>Type</b>	CHAR[32]
<b>Values</b>	Null terminated string (up to 32 characters including the Null character).
<b>Note</b>	Read-only parameter. This parameter is device dependent.

### **CORACQ\_PRM\_SHARED\_CAM\_RESET**

<b>Description</b>	Synchronize the reset output signal of the current acquisition module with another acquisition module of the board.
<b>Type</b>	UINT32
<b>Limits</b>	This value can only be set to a value different than CORACQ_VAL_SHARED_CONTROL_DISABLE if CORACQ_CAP_SHARED_CAM_RESET is TRUE.  CORACQ_CAP_SHARED_CAM_RESET is required to synchronize resetting more than 1 camera simultaneously. The master acquisition device must be acquiring in order for the slaved acquisition device to acquire.
<b>Values</b>	The acquisition module's index (master device) that the reset output signal will synchronize with, or CORACQ_VAL_SHARED_CONTROL_DISABLE ( = -1) if not used.
<b>CVI Entry</b>	[Shared Control Signals] Camera Reset

### **CORACQ\_PRM\_SHARED\_CAM\_TRIGGER**

<b>Description</b>	Synchronize the trigger output signal of the current acquisition module with another acquisition module of the board.
<b>Type</b>	UINT32
<b>Limits</b>	This value can only be set to a value different than CORACQ_VAL_SHARED_CONTROL_DISABLE if CORACQ_CAP_SHARED_CAM_TRIGGER is TRUE.  CORACQ_CAP_SHARED_CAM_TRIGGER is required to synchronize triggering more than 1 camera simultaneously. The master acquisition device must be acquiring in order for the slaved acquisition device to acquire.
<b>Values</b>	The acquisition module's index (master device) that the trigger output signal will synchronize with, or CORACQ_VAL_SHARED_CONTROL_DISABLE (= -1) if not used.
<b>CVI Entry</b>	[Shared Control Signals] Camera Trigger

### **CORACQ\_PRM\_SHARED\_EXT\_TRIGGER**

<b>Description</b>	Share the external trigger signal from another acquisition module.
<b>Type</b>	UINT32
<b>Limits</b>	This value can only be set to a value different than CORACQ_VAL_SHARED_CONTROL_DISABLE if CORACQ_CAP_SHARED_EXT_TRIGGER is TRUE.  CORACQ_CAP_SHARED_EXT_TRIGGER is required to trigger more than 1 acquisition module simultaneously using a single external trigger input signal. The master acquisition device must be acquiring in order for the slaved acquisition device to acquire properly.
<b>Values</b>	The acquisition module's index (master device) from which the external trigger signal will originate, or CORACQ_VAL_SHARED_CONTROL_DISABLE (= -1) if not used.
<b>CVI Entry</b>	[Shared Control Signals] External Trigger

### **CORACQ\_PRM\_SHARED\_FRAME\_INTEGRATE**

<b>Description</b>	Synchronize the frame integration output signal of the current acquisition module with another acquisition module of the board.
<b>Type</b>	UINT32
<b>Limits</b>	This value can only be set to a value different than CORACQ_VAL_SHARED_CONTROL_DISABLE if CORACQ_CAP_SHARED_FRAME_INTEGRATE is TRUE.  CORACQ_CAP_SHARED_FRAME_INTEGRATE is required to synchronize frame integration using multiple cameras simultaneously. The master acquisition device must be acquiring in order for the slaved device to acquire.
<b>Values</b>	The acquisition module's index (master device) that the frame integration output signal will synchronize with, or CORACQ_VAL_SHARED_CONTROL_DISABLE (= -1) if not used.
<b>CVI Entry</b>	[Shared Control Signals] Frame Integrate

### CORACQ\_PRM\_SHARED\_STROBE

<b>Description</b>	Share the strobe output signal from another acquisition module.
<b>Type</b>	UINT32
<b>Limits</b>	This value can only be set to a value different than CORACQ_VAL_SHARED_CONTROL_DISABLE if CORACQ_CAP_SHARED_STROBE is TRUE.  CORACQ_CAP_SHARED_STROBE is required when using a single strobe while acquiring with more than one camera simultaneously. The master acquisition device must be acquiring in order for the slaved acquisition device to acquire.
<b>Values</b>	The acquisition module's index (master device) from which the strobe output signal will originate, or CORACQ_VAL_SHARED_CONTROL_DISABLE (= -1) if not used.
<b>CVI Entry</b>	[Shared Control Signals] Strobe

### CORACQ\_PRM\_SHARED\_TIME\_INTEGRATE

<b>Description</b>	Synchronize the time integration output signal of the current acquisition module with another acquisition module of the board.
<b>Type</b>	UINT32
<b>Limits</b>	This value can only be set to a value different than CORACQ_VAL_SHARED_CONTROL_DISABLE if CORACQ_CAP_SHARED_TIME_INTEGRATE is TRUE.  CORACQ_CAP_SHARED_TIME_INTEGRATE is required when synchronizing time integration with multiple cameras simultaneously. The master acquisition device must be acquiring in order for the slaved device to acquire.
<b>Values</b>	The acquisition module's index (master device time integration output signal) which will be synchronized with, or CORACQ_VAL_SHARED_CONTROL_DISABLE (= -1) if not used.
<b>CVI Entry</b>	[Shared Control Signals] Time Integrate

### CORACQ\_PRM\_SHARPNESS

<b>Description</b>	Analog composite video sharpness control applied to the video signal. Applies to analog composite video signals only.
<b>Type</b>	UINT32
<b>Limits</b>	Range limits: CORACQ_CAP_SHARPNESS_MIN ... CORACQ_CAP_SHARPNESS_MAX.
<b>CVI Entry</b>	[Signal Conditioning] Sharpness
<b>Note</b>	This parameter has no units. Sharpness values are dependent on the board hardware used.

### CORACQ\_PRM\_SNAP\_COUNT

<b>Description</b>	Number of images to acquire per transfer count.
<b>Type</b>	UINT32
<b>Limits</b>	The value must be in the range: $1 \dots (2^{32}) - 1$ .
<b>Availability</b>	Available only if CORACQ_CAP_SNAP_COUNT is TRUE.
<b>CVI Entry</b>	[Stream Conditioning] Snap Count
<b>Notes</b>	CORACQ_CAP_SNAP_COUNT_MAX returns the maximum number of images per transfer count that is supported by the device.

### **CORACQ\_PRM\_STROBE\_DELAY**

<b>Description</b>	Strobe pulse delay #1 (in $\mu$ s).
<b>Type</b>	UINT32
<b>Limits</b>	Range limits: CORACQ_CAP_STROBE_DELAY_MIN to CORACQ_CAP_STROBE_DELAY_MAX.
<b>CVI Entry</b>	[Control Signals] Strobe Delay
<b>Note</b>	Validated only if CORACQ_PRM_STROBE_ENABLE is TRUE. See "Strobe Methods" for details on using the pulse delay #1 parameter.

### **CORACQ\_PRM\_STROBE\_DELAY\_2**

<b>Description</b>	Strobe pulse delay #2 (in $\mu$ s).
<b>Type</b>	UINT32
<b>Limits</b>	Range limits: CORACQ_CAP_STROBE_DELAY_2_MIN to CORACQ_CAP_STROBE_DELAY_2_MAX.
<b>Note</b>	Validated only if CORACQ_PRM_STROBE_ENABLE is TRUE. See "Strobe Methods" for details on using the pulse delay #2 parameter.

### **CORACQ\_PRM\_STROBE\_DURATION**

<b>Description</b>	Strobe pulse width (in $\mu$ s).
<b>Type</b>	UINT32
<b>Limits</b>	Range limits: CORACQ_CAP_STROBE_DURATION_MIN to CORACQ_CAP_STROBE_DURATION_MAX.
<b>CVI Entry</b>	[Control Signals] Strobe Duration
<b>Note</b>	Validated only if CORACQ_PRM_STROBE_ENABLE is TRUE.

### **CORACQ\_PRM\_STROBE\_ENABLE**

<b>Description</b>	Enable or disable the strobe pulse.
<b>Type</b>	UINT32
<b>Availability</b>	Available only if CORACQ_CAP_STROBE is TRUE.
<b>Values</b>	TRUE (0x00000001) Enable the strobe pulse. FALSE (0x00000000) Disable the strobe pulse.
<b>CVI Entry</b>	[Control Signals] Strobe Enable



### CORACQ\_PRM\_STROBE\_LEVEL

<b>Description</b>	Strobe signal level output by the acquisition device.	
<b>Type</b>	UINT32	
<b>Limits</b>	This value must match one of the supported capabilities of the acquisition device given by CORACQ_CAP_STROBE_LEVEL. The capability returns the ORed combination of all supported values.	
<b>Values</b>	CORACQ_VAL_LEVEL_TTL (0x00000001) CORACQ_VAL_LEVEL_422 (0x00000002) CORACQ_VAL_LEVEL_LVDS (0x00000004) CORACQ_VAL_LEVEL_24VOLTS (0x00000008) CORACQ_VAL_LEVEL_OPTO (0x00000010) CORACQ_VAL_LEVEL_LVTTL (0x00000020) CORACQ_VAL_LEVEL_12VOLTS (0x00000040)	TTL signal level. RS-422 signal level. LVDS signal level. 24V signal level. Opto-coupled signal level. Low voltage TTL signal level. 12V signal level.
<b>CVI Entry</b>	[Control Signals] Strobe Level	
<b>Note</b>	Validated only if CORACQ_PRM_STROBE_ENABLE is TRUE.	

### CORACQ\_PRM\_STROBE\_METHOD

<b>Description</b>	Select the strobe pulse output method.	
<b>Type</b>	UINT32	
<b>Limits</b>	This value must match one of the supported capabilities of the acquisition device given by CORACQ_CAP_STROBE_METHOD. The capability returns the ORed combination of all supported values.	
<b>Values</b>	See "Strobe Methods".	
<b>CVI Entry</b>	[Control Signals] Strobe Method	
<b>Note</b>	Validated only if CORACQ_PRM_STROBE_ENABLE is TRUE.	

### CORACQ\_PRM\_STROBE\_POLARITY

<b>Description</b>	Strobe pulse polarity.	
<b>Type</b>	UINT32	
<b>Limits</b>	This value must match one of the supported capabilities of the acquisition device given by CORACQ_CAP_STROBE_POLARITY. The capability returns the ORed combination of all supported values.	
<b>Values</b>	CORACQ_VAL_ACTIVE_LOW (0x00000001) CORACQ_VAL_ACTIVE_HIGH (0x00000002)	Strobe pulse will be active low. Strobe pulse will be active high.
<b>CVI Entry</b>	[Control Signals] Strobe Polarity	
<b>Note</b>	Validated only if CORACQ_PRM_STROBE_ENABLE is TRUE.	

### CORACQ\_PRM\_TIME\_INTEGRATE\_DELAY

<b>Description</b>	Time integration delay (in $\mu$ s). After receiving a trigger pulse (external, internal or software), the acquisition device will wait this delay before generating the time integration pulse(s).	
<b>Type</b>	UINT32	
<b>Limits</b>	Range limits: CORACQ_CAP_TIME_INTEGRATE_DELAY_MIN ... CORACQ_CAP_TIME_INTEGRATE_DELAY_MAX.	
<b>CVI Entry</b>	[Control Signals] Time Integrate Delay	
<b>Note</b>	Validated only if CORACQ_PRM_TIME_INTEGRATE_ENABLE is TRUE.	



## CORACQ\_PRM\_TIME\_INTEGRATE\_DURATION

<b>Description</b>	Time integration pulse width (in $\mu$ s). Applies to area scan cameras only.
<b>Type</b>	UINT32
<b>Limits</b>	Acquisition device range limits: CORACQ_CAP_TIME_INTEGRATE_DURATION_MIN to CORACQ_CAP_TIME_INTEGRATE_DURATION_MAX.  Camera range limits: CORACQ_PRM_CAM_TIME_INTEGRATE_DURATION_MIN to CORACQ_PRM_CAM_TIME_INTEGRATE_DURATION_MAX.
<b>CVI Entry</b>	[Control Signals] Time Integrate Duration
<b>Note</b>	Validated only if CORACQ_PRM_TIME_INTEGRATE_ENABLE is TRUE.

## CORACQ\_PRM\_TIME\_INTEGRATE\_ENABLE

<b>Description</b>	Enable or disable the time integration signal pulse to the camera. Applies to area scan cameras only.
<b>Type</b>	UINT32
<b>Availability</b>	Available only if CORACQ_CAP_TIME_INTEGRATE is TRUE.
<b>Values</b>	TRUE (0x00000001) Enable time integration pulse. FALSE (0x00000000) Disable time integration pulse.
<b>CVI Entry</b>	[Control Signals] Time Integrate Enable
<b>Note</b>	This parameter is mutually exclusive with CORACQ_PRM_CAM_TRIGGER_ENABLE and CORACQ_PRM_FRAME_INTEGRATE_ENABLE.

## CORACQ\_PRM\_TIME\_STAMP\_BASE

<b>Description</b>	Sets the acquisition device time stamp basic units.
<b>Type</b>	UINT32
<b>Limits</b>	This value must match one of the supported capabilities of the acquisition device given by CORACQ_CAP_TIME_STAMP_BASE. The capability returns the ORed combination of all supported values.
<b>Values</b>	CORACQ_VAL_TIME_BASE_US (0x00000001), the time base is in microseconds CORACQ_VAL_TIME_BASE_MS (0x00000002), the time base is in milliseconds CORACQ_VAL_TIME_BASE_LINE_VALID (0x00000004), the time base is in line valid received NOTE: This macro replaces obsolete one: CORACQ_VAL_TIME_BASE_LINE CORACQ_VAL_TIME_BASE_LINE_TRIGGER (0x00000008), the time base is in external line trigger or shaft encoder pulse (after drop/multiply operation) CORACQ_VAL_TIME_BASE_FRAME_VALID (0x00000010), the time base is in frame valid received. NOTE: This macro replaces obsolete one: CORACQ_VAL_TIME_BASE_FRAME CORACQ_VAL_TIME_BASE_FRAME_TRIGGER (0x00000020), the time base is in valid external frame trigger received (does not count the ones that are ignored). CORACQ_VAL_TIME_BASE_SHAFT_ENCODER (0x00000040), the time base is in external line trigger or shaft encoder pulse (before drop/multiply operation) CORACQ_VAL_TIME_BASE_NS (0x00000080), the time base is in nanoseconds CORACQ_VAL_TIME_BASE_PIXEL_CLK (0x00000100), the time base is in camera pixel clock CORACQ_VAL_TIME_BASE_100NS (0x00000200), the time base is in 100 nanoseconds
<b>CVI Entry</b>	[General] Time Stamp Base
<b>Note</b>	If the acquisition device does not support this feature, the CORACQ_CAP_TIME_STAMP_BASE capability returns 0 or CORSTATUS_CAP_INVALID.

## CORACQ\_PRM\_VERTICAL\_TIMEOUT\_DELAY

<b>Description</b>	Following a valid external/internal/software trigger, this parameter specifies the time delay before which the end of a vertical sync (analog cameras) or beginning of a frame valid (digital cameras) must be detected. If none are detected after this delay, a vertical timeout delay event will be generated if the event is activated. Once a vertical timeout is detected, the acquisition device resets itself and waits for the next valid external/internal/software trigger. Applies to area scan cameras only.
<b>Type</b>	UINT32
<b>Values</b>	Numerical value representing the delay in $\mu$ sec.
<b>Limits</b>	Range Limits: CORACQ_CAP_VERTICAL_TIMEOUT_DELAY_MIN ... CORACQ_CAP_VERTICAL_TIMEOUT_DELAY_MAX.
<b>CVI Entry</b>	[Control Signals] Vertical Timeout Delay
<b>Note</b>	See also the related event CORACQ_PRM_EVENT_TYPE:CORACQ_VAL_EVENT_TYPE_VERTICAL_TIMEOUT  For analog cameras, if the WEN signal is used, the beginning of the WEN must be detected before the programmed delay expires. For analog cameras, if synching to blanking signals, the end of the blanking signal must be detected before the programming delay expires.

## CORACQ\_PRM\_VIC\_NAME

<b>Description</b>	VIC parameter file description field (up to 63 characters long).
<b>Type</b>	BYTE [64]
<b>CVI Entry</b>	[General] Vic Name

## CORACQ\_PRM\_VSYNC\_REF

<b>Description</b>	Vertical sync reference.
<b>Type</b>	UINT32
<b>Limits</b>	This value must match one of the supported capabilities of the acquisition device given by CORACQ_PRM_VSYNC_REF. The vertical sync reference is used as the starting point for counting video frame lines. Selecting the end of sync as the reference is useful when dealing with a variable width sync. This is often the case when time-integrating a video signal. The capability returns the ORed combination of all supported values.
<b>Values</b>	CORACQ_VAL_SYNC_REF_BEGIN (0x00000001), Beginning of vertical sync. CORACQ_VAL_SYNC_REF_END (0x00000002), End of vertical sync.  CORACQ_VAL_SYNC_REF_HV_DEPENDENT (0x00000004), Horizontal and Vertical sync reference are locked together.
<b>CVI Entry</b>	[Stream Conditioning] Vertical Sync Reference

## CORACQ\_PRM\_WEN\_ENABLE

<b>Description</b>	Enable or disable use of the WEN (Write ENable) signal from the camera.
<b>Type</b>	UINT32
<b>Availability</b>	Available only if CORACQ_CAP_WEN is TRUE.
<b>Values</b>	TRUE (0x00000001)                      Enable the use of the WEN signal. FALSE (0x00000000)                    Disable the use of the WEN signal
<b>CVI Entry</b>	[Control Signals] WEN Enable

---

# Data Structures

Defines Data Structures

## Pin Connector Description

Certain frame grabbers provide connectors that are configurable; that is, it is possible to assign a control signal—such as pixel clock, HSync, or VSync—to specific pins on a given connector. Sapera LT provides a list of camera parameters to describe the pin assignment for a given camera (see the "Connector Description" parameters list within the Camera Related Parameters section in Advanced Acquisition Control). This allows the frame grabber to automatically configure its pins to meet the camera specifications. Refer to your frame grabber user's manual for a description of the board's capabilities.

Teledyne DALSA's CamExpert allows for the creation of a camera file (CCA file) with the desired connector descriptions. The bit field description below is provided for users who want to interpret or edit the camera files manually. It represents the value assigned to each of the connector description parameters.

Bits	31-24	23-16	15-0
Description	Connector #	Connector Type	Pin #

Bit Field	Description
Pin #	<i>Pin number on connector (1.. n).</i> <b>Note:</b> The macro CORACQ_VAL_CONNECTOR_PIN( <i>value</i> ) is provided to extract the pin #, where the <i>value</i> is a valid pin connector description.
Connector Type	Type of connector: <i>CORACQ_VAL_CONNECTOR_TYPE_HIROSE12</i> <i>12-pin Hirose connector</i> <i>CORACQ_VAL_CONNECTOR_TYPE_CAMLINK</i> <i>Camera Link connector.</i> <i>The pin number represents the camera control line #: CC1, CC2, CC3 &amp; CC4.</i> <i>CORACQ_VAL_CONNECTOR_TYPE_CAM_CONTROL</i> <i>Generic camera control connector. The pin numbers (up to 8) are device dependent.</i> <i>CORACQ_VAL_CONNECTOR_TYPE_CX4</i> <i>CX4 camera connector.</i> <i>CORACQ_VAL_CONNECTOR_TYPE_CLHS</i> <i>CLHS camera connector.</i> <b>Note:</b> The macro CORACQ_VAL_CONNECTOR_TYPE( <i>value</i> ) is provided to extract the connector type, where the <i>value</i> is a valid connector type.
Connector #	<i>Number of the connector (in the event the camera has more than 1 connector, 1 .. n).</i> <b>Note:</b> The macro CORACQ_VAL_CONNECTOR_NUMBER( <i>value</i> ) is provided to extract the connector number, where the <i>value</i> is a valid connector number.

The following are the related capabilities that give the valid values that can be applied to the connector number, connector type, and pin number.

---

## CORACQ\_CAP\_CONNECTOR\_TYPE

<b>Description</b>	Specifies the different connector types available on the device.	
<b>Type</b>	UINT32	
<b>Values</b>	CORACQ_VAL_CONNECTOR_TYPE_HIROSE12 (0x00000001)	12-pin Hirose connector
	CORACQ_VAL_CONNECTOR_TYPE_CAMLINK (0x00000002)	Camera Link connector
	CORACQ_VAL_CONNECTOR_TYPE_CAM_CONTROL (0x00000004)	Generic camera control connector
	CORACQ_VAL_CONNECTOR_TYPE_CX4 (0x00000008)	CX4 camera connector
	CORACQ_VAL_CONNECTOR_TYPE_CLHS (0x00000010)	CLHS camera connector

## CORACQ\_CAP\_CONNECTOR\_CAMLINK

<b>Description</b>	Specifies the different signals that the acquisition device can route to the Cam Link CC1, CC2, CC3, and CC4 connector pins.
<b>Type</b>	UINT32[4]
<b>Values</b>	Each entry in the table represents a bit field representing the valid signals that can be routed to the respective CameraLink pins. See "Signal Name Definitions" for CORACQ_VAL_SIGNAL_NAME_xxx definitions.

## CORACQ\_CAP\_CONNECTOR\_HIROSE12

<b>Description</b>	Specifies the different signals that the acquisition device can route to the Hirose-12 connector pins.
<b>Type</b>	UINT32[12]
<b>Values</b>	Each entry in the table represents a bit field representing the valid signals that can be routed to the respective Hirose-12 pins. See "Signal Name Definitions" for CORACQ_VAL_SIGNAL_NAME_xxx definitions.

## CORACQ\_CAP\_CONNECTOR\_CAM\_CONTROL

<b>Description</b>	Specifies the different signals that the acquisition device can route to the generic camera control connector pins.
<b>Type</b>	UINT32[8]
<b>Values</b>	Each entry in the table represents a bit field representing the valid signals that can be routed to the respective generic camera control pins. See "Signal Name Definitions" or CORACQ_VAL_SIGNAL_NAME_xxx definitions.

---

## Signal Name Definitions

Define	Value	Definition
CORACQ_VAL_SIGNAL_NAME_NO_CONNECT	0x00000001	No Connection
CORACQ_VAL_SIGNAL_NAME_HD	0x00000002	Horizontal Drive
CORACQ_VAL_SIGNAL_NAME_VD	0x00000004	Vertical Drive
CORACQ_VAL_SIGNAL_NAME_PULSE0	0x00000008	Camera Control Pulse 0
CORACQ_VAL_SIGNAL_NAME_PULSE1	0x00000010	Camera Control Pulse 1
CORACQ_VAL_SIGNAL_NAME_PIXEL_CLOCK_IN	0x00000020	Pixel Clock In
CORACQ_VAL_SIGNAL_NAME_PIXEL_CLOCK_OUT	0x00000040	Pixel Clock Out
CORACQ_VAL_SIGNAL_NAME_LINESCAN_DIRECTION	0x00000080	Linescan Direction
CORACQ_VAL_SIGNAL_NAME_WEN	0x00000100	WEN (Write ENable)
CORACQ_VAL_SIGNAL_NAME_EXT_TRIGGER	0x00000200	External Trigger
CORACQ_VAL_SIGNAL_NAME_EXT_LINE_TRIGGER	0x00000400	External Line Trigger
CORACQ_VAL_SIGNAL_NAME_INT_FRAME_TRIGGER	0x00000800	Internal Frame Trigger
CORACQ_VAL_SIGNAL_NAME_INT_LINE_TRIGGER	0x00001000	Internal Line Trigger
CORACQ_VAL_SIGNAL_NAME_SOFTWARE_TRIGGER	0x00002000	Software Trigger
CORACQ_VAL_SIGNAL_NAME_GND	0x00004000	Ground
CORACQ_VAL_SIGNAL_NAME_POWER_12V	0x00008000	Power 12V
CORACQ_VAL_SIGNAL_NAME_VIDEO	0x00010000	Video
CORACQ_VAL_SIGNAL_NAME_VIDEO_GND	0x00020000	Video Ground

---

## Structure Definitions

Defines CORACQ\_CAM\_IO\_CONTROL

CORACQ\_CAM\_IO\_CONTROL

```
typedef struct
{
    char    label[12];        //User defined descriptive label of the camera control
                                //(for example, BIN, GAIN...)
    UINT32  connectorInput;   // Pin Connector Description
    UINT32  nbBits;           //Number of bits needed for this control

    UINT32  level;            //CORACQ_VAL_LEVEL_TTL (0x00000001)
                                //CORACQ_VAL_LEVEL_422 (0x00000002)
                                //CORACQ_VAL_LEVEL_LVDS (0x00000004)
                                //CORACQ_VAL_LEVEL_24VOLTS (0x00000008)
                                //CORACQ_VAL_LEVEL_OPTO (0x00000010)
                                //CORACQ_VAL_LEVEL_LVTTL (0x00000020)
                                //CORACQ_VAL_LEVEL_12VOLTS (0x00000040)

    UINT32  direction;        //CORACQ_VAL_DIR_INPUT (0x00000001)
                                //CORACQ_VAL_DIR_OUTPUT (0x00000002)

    UINT32  polarity;         //Used only for information purposes by an application.
                                //The driver does not make any use of this member.
                                //CORACQ_VAL_ACTIVE_LOW (0x00000001)
                                //CORACQ_VAL_ACTIVE_HIGH (0x00000002)

    UINT32  value;            //The control's default value when used as an output.
                                //If a bit is set to '1', the corresponding output
                                //will be set to on or high;
                                //otherwise, the output will be set to off or low.

} CORACQ_CAM_IO_CONTROL, *PCORACQ_CAM_IO_CONTROL;
```

# Camera Control Method Definitions

This section provides definitions and timing diagrams for the camera control methods supported by Sapera LT. Topics covered are:

- Camera Reset Method
- Camera Trigger Methods
- Frame Integrate Methods
- Line Integrate Methods
- Line Trigger Methods
- Time Integrate Methods
- Strobe Methods

## Camera Reset Method

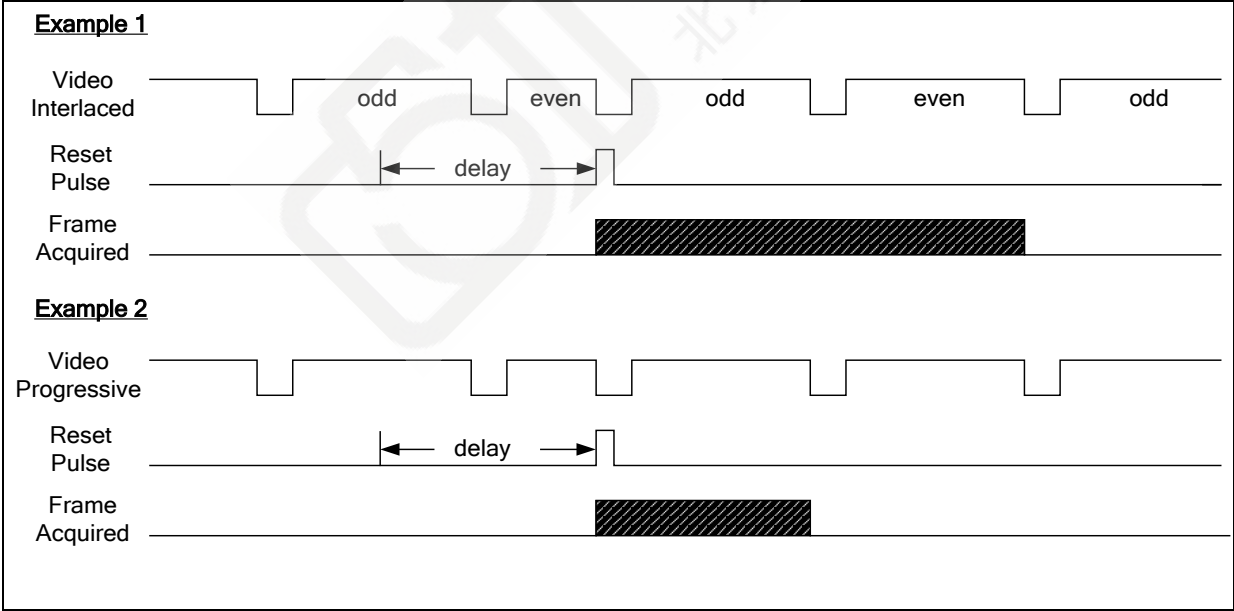
The following camera reset method is available:

CORACQ\_VAL\_CAM\_RESET\_METHOD\_1

CORACQ\_VAL\_CAM\_RESET\_METHOD\_1

<b>Value</b>	0x00000001 (Camera Reset Method #1)
<b>Description</b>	Method selection is via the parameter CORACQ_PRM_CAM_RESET_METHOD. This method generates an asynchronous reset pulse to a camera. The next generated frame then acquired. The reset pulse is defined by the following parameters: Delay CORACQ_PRM_CAM_RESET_DELAY Duration CORACQ_PRM_CAM_RESET_DURATION Polarity CORACQ_PRM_CAM_RESET_POLARITY

### Example:





# Camera Trigger Methods

The following camera trigger methods are available (area scan only):

- CORACQ\_VAL\_CAM\_TRIGGER\_METHOD\_1
- CORACQ\_VAL\_CAM\_TRIGGER\_METHOD\_2
- CORACQ\_VAL\_CAM\_TRIGGER\_METHOD\_3

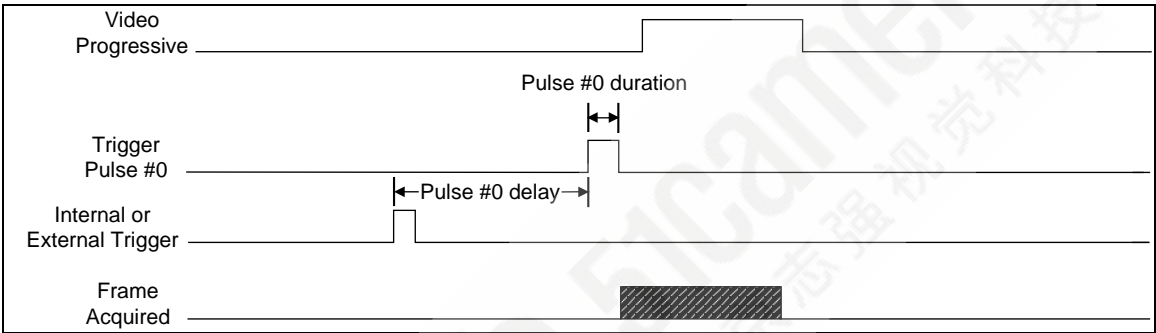
## CORACQ\_VAL\_CAM\_TRIGGER\_METHOD\_1

**Numerical Value** 0x00000001 (Camera Trigger Method 1)

**Description** Method selection is via the parameter CORACQ\_PRM\_CAM\_TRIGGER\_METHOD. This method generates an asynchronous trigger pulse to a camera (area scan only). The next generated frame is then acquired. The trigger pulse is defined by the following parameters: .

Delay CORACQ\_PRM\_CAM\_TRIGGER\_DELAY  
Duration CORACQ\_PRM\_CAM\_TRIGGER\_DURATION  
Polarity CORACQ\_PRM\_CAM\_TRIGGER\_POLARITY

### Example:



### CORACQ\_VAL\_CAM\_TRIGGER\_METHOD\_2

**Numerical Value** 0x00000002 (Camera Trigger Method #2)

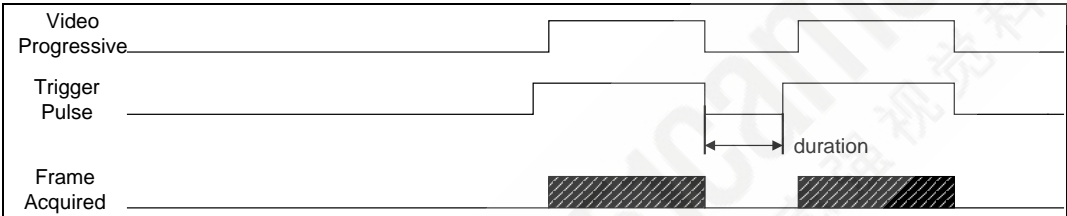
**Description** Method selection is via the parameter CORACQ\_PRM\_CAM\_TRIGGER\_METHOD. This method generates an asynchronous trigger pulse to a camera (area scan only). The next generated frame is then acquired. This method's trigger pulse controls the number of lines output by the camera and is usually used to control the length of the frame output by the camera (partial scanning). The trigger pulse is defined by the parameter CORACQ\_PRM\_CAM\_TRIGGER\_POLARITY. Its length is dependent on the number of lines to acquire.

The parameters CORACQ\_PRM\_VSYNC + CORACQ\_PRM\_VBACK\_PORCH + CORACQ\_PRM\_CROP\_TOP + CORACQ\_PRM\_CROP\_HEIGHT represent (in this case) the minimum time between triggers to the camera. Required for cameras where the CCD has a minimum reset time before it can be triggered again.

The trigger pulse is defined by the following parameters:

Delay	CORACQ_PRM_CAM_TRIGGER_DELAY
Duration	CORACQ_PRM_VSYNC + CORACQ_PRM_VBACK_PORCH + CORACQ_PRM_CROP_TOP + CORACQ_PRM_CROP_HEIGHT
Polarity	CORACQ_PRM_CAM_TRIGGER_POLARITY

#### Example:



### CORACQ\_VAL\_CAM\_TRIGGER\_METHOD\_3

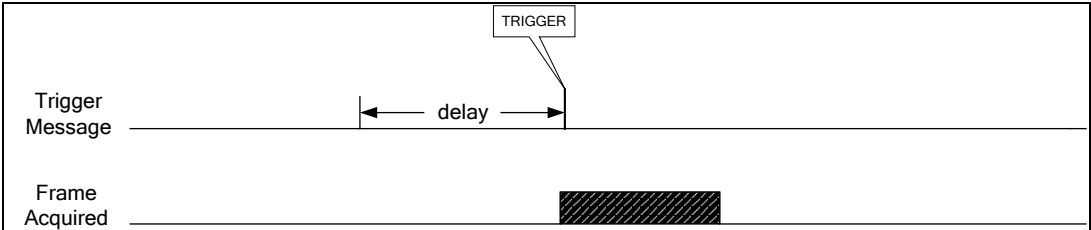
**Numerical Value** 0x00000004 (Camera Trigger Method #3)

**Description** Method selection is via the parameter CORACQ\_PRM\_CAM\_TRIGGER\_METHOD. This method generates a camera trigger message to a camera (area scan only). The next generated frame is then acquired.

The trigger pulse is defined by the following parameters:

Delay	CORACQ_PRM_CAM_TRIGGER_DELAY
Duration	N/A
Polarity	N/A

#### Example:



# Frame Integrate Methods

The following frame integrate methods are available:

CORACQ\_VAL\_FRAME\_INTEGRATE\_METHOD\_1

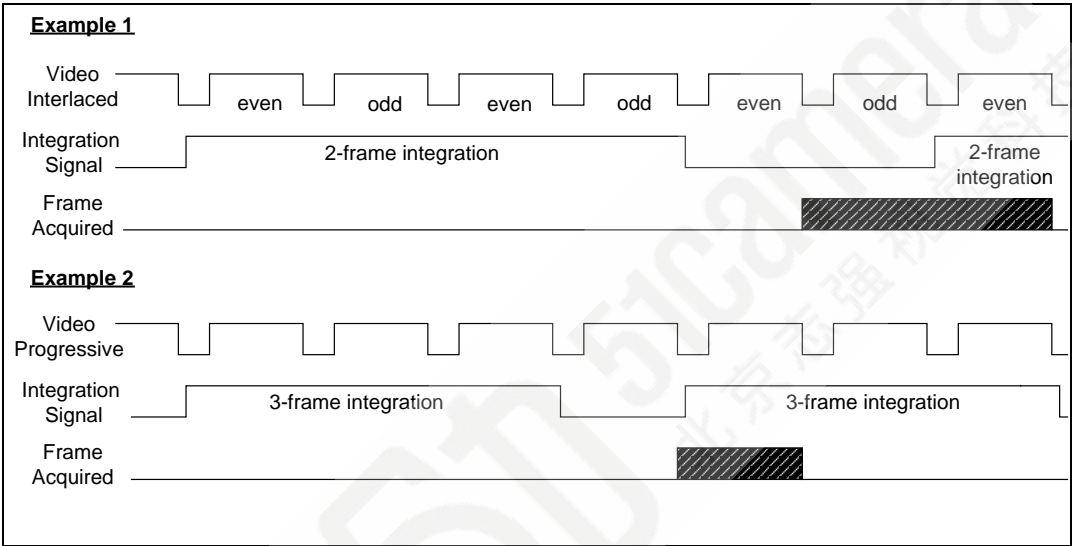
CORACQ\_VAL\_FRAME\_INTEGRATE\_METHOD\_2

## CORACQ\_VAL\_FRAME\_INTEGRATE\_METHOD\_1

**Numerical Value** 0x00000001 (Frame Integration Method #1)

**Description** The frame integration signal is sent prior to the first field to be integrated. The signal is then held until the last field to be integrated is reached. The next frame is then acquired. Method selection is via the parameter CORACQ\_PRM\_FRAME\_INTEGRATE\_METHOD. The polarity of this signal is specified by the parameter CORACQ\_PRM\_FRAME\_INTEGRATE\_POLARITY. The number of frames to integrate is specified with the parameter CORACQ\_PRM\_FRAME\_INTEGRATE\_COUNT.

### Example:

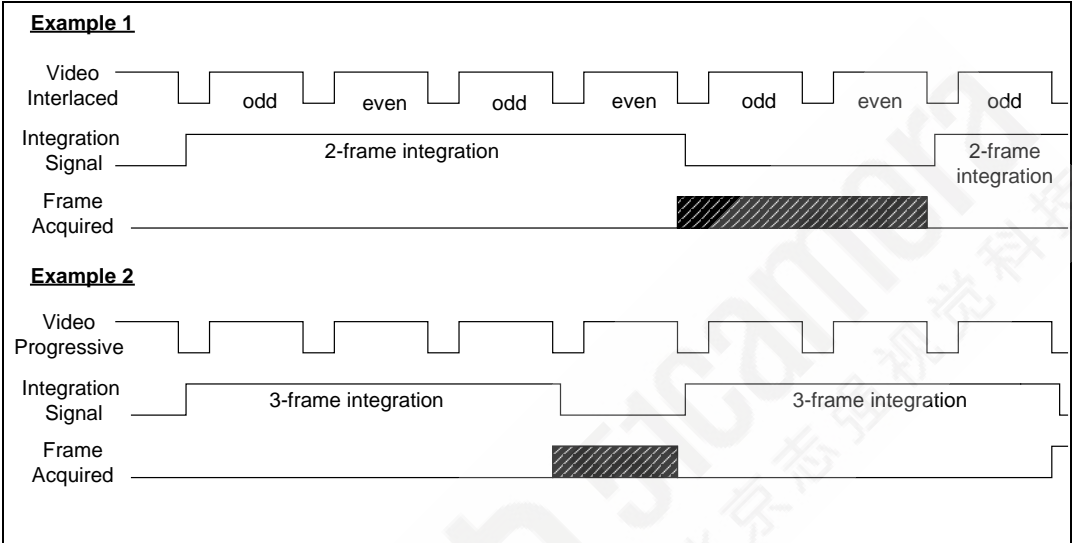


CORACQ\_VAL\_FRAME\_INTEGRATE\_METHOD\_2

**Numerical Value** 0x00000002 (Frame Integration Method #2)

**Description** The frame integration signal is sent during the vertical sync of the first field to be integrated. The signal is then held until the first field to be acquired is reached. The current frame is then acquired. Method selection is via the parameter CORACQ\_PRM\_FRAME\_INTEGRATE\_METHOD.  
The polarity of this signal is specified by the parameter CORACQ\_PRM\_FRAME\_INTEGRATE\_POLARITY.  
The number of frames to integrate is specified with the parameter CORACQ\_PRM\_FRAME\_INTEGRATE\_COUNT.

Example:



# Line Integrate Methods

The following line integrate methods are available for line scan cameras:

- CORACQ\_VAL\_LINE\_INTEGRATE\_METHOD\_1
- CORACQ\_VAL\_LINE\_INTEGRATE\_METHOD\_2
- CORACQ\_VAL\_LINE\_INTEGRATE\_METHOD\_3
- CORACQ\_VAL\_LINE\_INTEGRATE\_METHOD\_4
- CORACQ\_VAL\_LINE\_INTEGRATE\_METHOD\_7
- CORACQ\_VAL\_LINE\_INTEGRATE\_METHOD\_8
- CORACQ\_VAL\_LINE\_INTEGRATE\_METHOD\_9
- CORACQ\_VAL\_LINE\_INTEGRATE\_METHOD\_10

## CORACQ\_VAL\_LINE\_INTEGRATE\_METHOD\_1

**Numerical Value** 0x00000001 (Line Integration Method #1)

**Description** Method selection is via the parameter CORACQ\_PRM\_LINE\_INTEGRATE\_METHOD. This method generates two pulses on two different outputs. The distance between the end of the first pulse and the start of the second pulse is the integration time (as specified by the parameter CORACQ\_PRM\_LINE\_INTEGRATE\_DURATION). The second pulse is also the Line Trigger input to the camera. For example, on a Teledyne DALSA camera, the first pulse is the 'PRIN' signal while the second pulse is the 'EXSYNC' signal.

The pulses are defined by the following parameters:

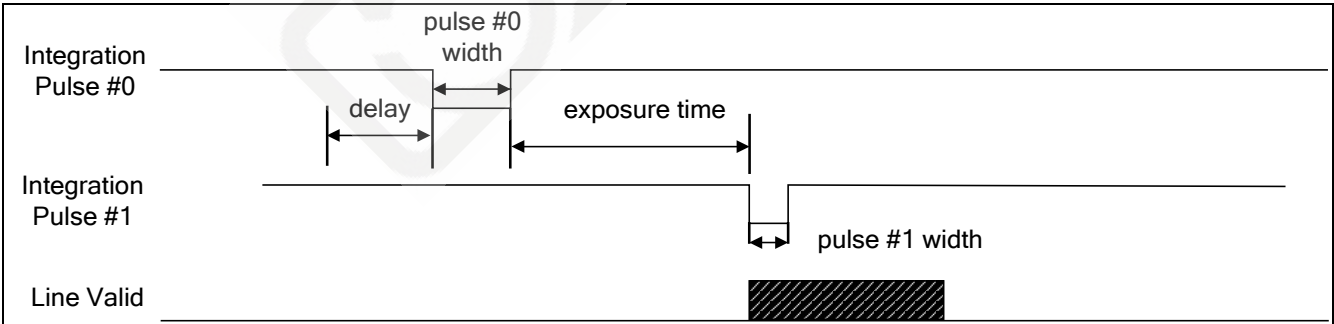
1<sup>st</sup> Pulse

Delay CORACQ\_PRM\_LINE\_INTEGRATE\_PULSE0\_DELAY  
Duration CORACQ\_PRM\_LINE\_INTEGRATE\_PULSE0\_DURATION  
Polarity CORACQ\_PRM\_LINE\_INTEGRATE\_PULSE0\_POLARITY

2<sup>nd</sup> Pulse

Delay CORACQ\_PRM\_LINE\_INTEGRATE\_PULSE0\_DELAY +  
CORACQ\_PRM\_LINE\_INTEGRATE\_PULSE0\_DURATION +  
CORACQ\_PRM\_LINE\_INTEGRATE\_DURATION  
Duration CORACQ\_PRM\_LINE\_INTEGRATE\_PULSE1\_DURATION  
Polarity CORACQ\_PRM\_LINE\_INTEGRATE\_PULSE1\_POLARITY

### Example:



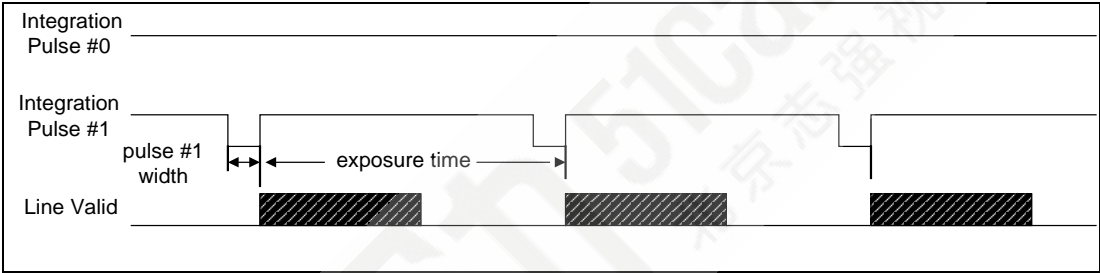
CORACQ\_VAL\_LINE\_INTEGRATE\_METHOD\_2

**Numerical Value** 0x00000002 (Line Integration Method #2)

**Description** Method selection is via the parameter CORACQ\_PRM\_LINE\_INTEGRATE\_METHOD. This method generates two consecutive trigger pulses on the camera's Line Trigger input. The time interval between the end of the two trigger pulses represents the integration time (as specified by the parameter CORACQ\_PRM\_LINE\_INTEGRATE\_DURATION). An optional signal with a fixed level might be present. For example, on a Teledyne DALSA camera, the Line Trigger input would be the 'EXSYNC' signal and the optional signal would be the 'PRIN' signal. Both pulses are described by the parameters CORACQ\_PRM\_LINE\_INTEGRATE\_PULSE1\_DURATION and CORACQ\_PRM\_LINE\_INTEGRATE\_PULSE1\_POLARITY. The optional signal with a fixed level is described by the parameter CORACQ\_PRM\_LINE\_INTEGRATE\_PULSE0\_POLARITY.

- 1<sup>st</sup> Pulse
- Delay N/A
  - Duration N/A
  - Polarity CORACQ\_PRM\_LINE\_INTEGRATE\_PULSE0\_POLARITY
- 2<sup>nd</sup> Pulse
- Delay N/A
  - Duration CORACQ\_PRM\_LINE\_INTEGRATE\_PULSE1\_DURATION
  - Polarity CORACQ\_PRM\_LINE\_INTEGRATE\_PULSE1\_POLARITY

Example:



CORACQ\_VAL\_LINE\_INTEGRATE\_METHOD\_3

**Numerical Value** 0x00000004 (Line Integration Method #3)

**Description** Method selection via the parameter CORACQ\_PRM\_LINE\_INTEGRATE\_METHOD. This method generates an asynchronous line integration pulse to a camera. The width of this pulse represents the integration time (as specified by the parameter CORACQ\_PRM\_LINE\_INTEGRATE\_DURATION). An optional signal with a fixed level might be present. For example, on a Teledyne DALSA camera, the integration pulse would be the 'EXSYNC' signal and the optional signal would be the 'PRIN' signal. The integration pulse is described by the parameter CORACQ\_PRM\_LINE\_INTEGRATE\_PULSE1\_POLARITY and CORACQ\_PRM\_LINE\_INTEGRATE\_PULSE1\_DELAY. The optional signal with a fixed level is described by the parameter CORACQ\_PRM\_LINE\_INTEGRATE\_PULSE0\_POLARITY.

- 1<sup>st</sup> Pulse
- Delay

N/A
- Duration

N/A
- Polarity

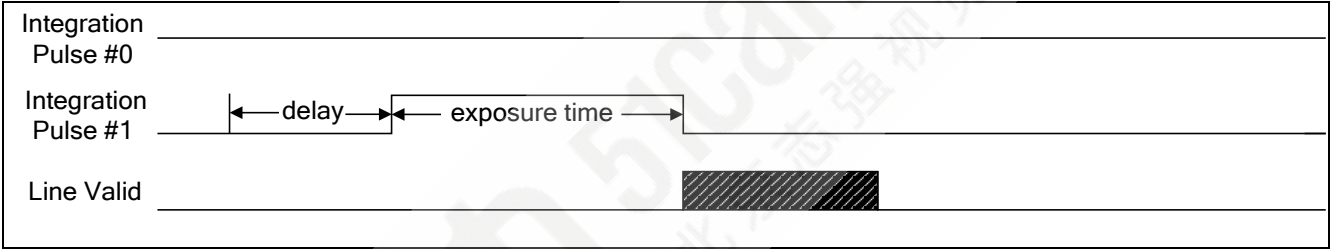
CORACQ\_PRM\_LINE\_INTEGRATE\_PULSE0\_POLARITY
- 2<sup>nd</sup> Pulse
- Delay

CORACQ\_PRM\_LINE\_INTEGRATE\_PULSE1\_DELAY
- Duration

CORACQ\_PRM\_LINE\_INTEGRATE\_PULSE1\_DURATION
- Polarity

CORACQ\_PRM\_LINE\_INTEGRATE\_PULSE1\_POLARITY

Example:





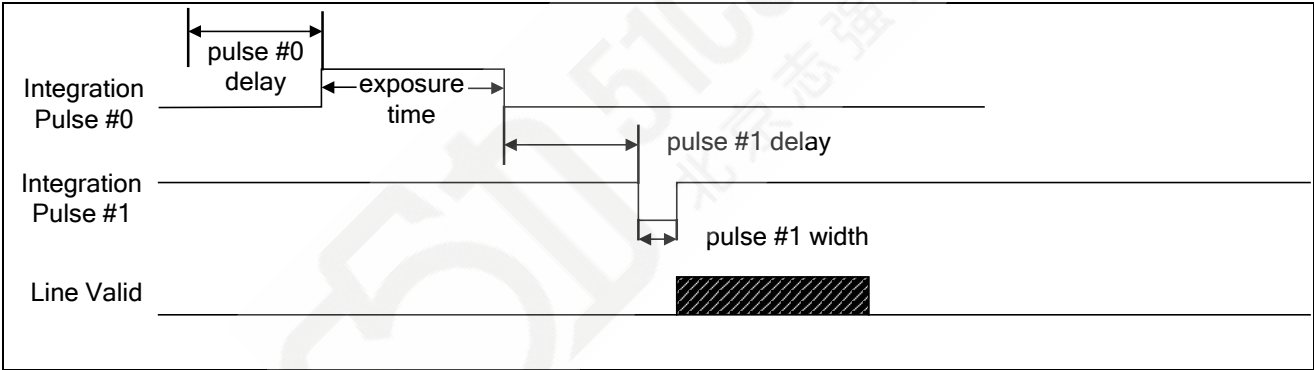
CORACQ\_VAL\_LINE\_INTEGRATE\_METHOD\_4

**Numerical Value** 0x00000008 (Line Integration Method #4)

**Description** Method selection is via the parameter CORACQ\_PRM\_LINE\_INTEGRATE\_METHOD. This method generates an integration pulse followed by a trigger pulse on the camera's line trigger. The width of the integration pulse represents the integration time (as specified by the parameter CORACQ\_PRM\_LINE\_INTEGRATE\_DURATION). The first pulse is described by the parameter CORACQ\_PRM\_LINE\_INTEGRATE\_PULSE0\_POLARITY. The second pulse is described by the parameters CORACQ\_PRM\_LINE\_INTEGRATE\_PULSE1\_DELAY, CORACQ\_PRM\_LINE\_INTEGRATE\_PULSE1\_DURATION and CORACQ\_PRM\_LINE\_INTEGRATE\_PULSE1\_POLARITY.

1 <sup>st</sup> Pulse	
Delay	CORACQ_PRM_LINE_INTEGRATE_PULSE0_DELAY
Duration	CORACQ_PRM_LINE_INTEGRATE_PULSE0_DURATION
Polarity	CORACQ_PRM_LINE_INTEGRATE_PULSE0_POLARITY
2 <sup>nd</sup> Pulse	
Delay	CORACQ_PRM_LINE_INTEGRATE_PULSE0_DELAY + CORACQ_PRM_LINE_INTEGRATE_PULSE1_DELAY + CORACQ_PRM_LINE_INTEGRATE_DURATION
Duration	CORACQ_PRM_LINE_INTEGRATE_PULSE1_DURATION
Polarity	CORACQ_PRM_LINE_INTEGRATE_PULSE1_POLARITY

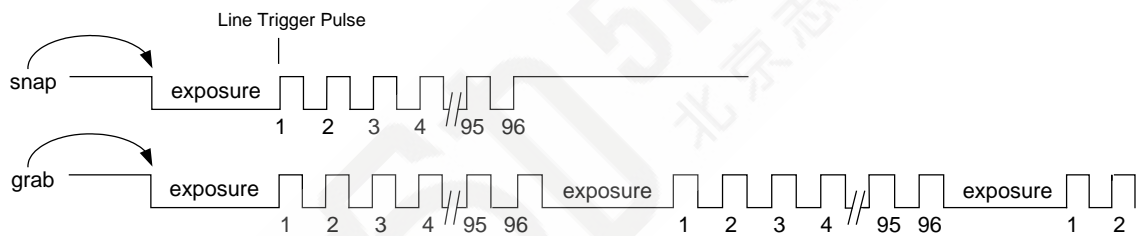
Example:



**CORACQ\_VAL\_LINE\_INTEGRATE\_METHOD\_7**

Numerical Value	0x00000040 (Line Integration Method #7)
Description	<p>Method selection is via the parameter CORACQ_PRM_LINE_INTEGRATE_METHOD. This method generates two type of pulses on the same output. The distance between the start of the first pulse and the start of the second pulse is the exposure time (as specified by the parameter CORCAM_PRM_LINE_INTEGRATE_PULSE0_DURATION). The second pulse is also the Line Trigger input to the camera. The first pulse is defined by the parameters CORACQ_PRM_LINE_INTEGRATE_PULSE0_DURATION and CORACQ_PRM_LINE_INTEGRATE_PULSE0_POLARITY. The second pulse is defined by the parameters CORACQ_PRM_LINE_INTEGRATE_PULSE1_DURATION and CORACQ_PRM_LINE_INTEGRATE_PULSE1_POLARITY.</p> <p>1<sup>st</sup> Pulse</p> <p>Delay                   CORACQ_PRM_LINE_INTEGRATE_PULSE0_DELAY</p> <p>Duration               CORACQ_PRM_LINE_INTEGRATE_PULSE0_DURATION</p> <p>Polarity               CORACQ_PRM_LINE_INTEGRATE_PULSE0_POLARITY</p> <p>2<sup>nd</sup> Pulse</p> <p>Delay                   CORACQ_PRM_LINE_INTEGRATE_PULSE1_DURATION</p> <p>Duration               CORACQ_PRM_LINE_INTEGRATE_PULSE1_DURATION</p> <p>Polarity               CORACQ_PRM_LINE_INTEGRATE_PULSE1_POLARITY</p>
Note	This camera is always integrating lines so the first few frames will have a saturated image after a grab.

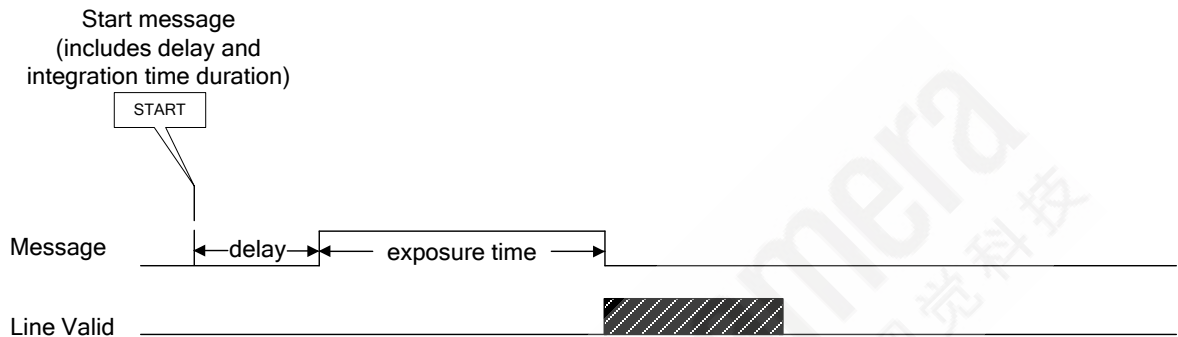
**Example:**



CORACQ\_VAL\_LINE\_INTEGRATE\_METHOD\_8

Numerical Value	0x00000080 (Line Integration Method #8)
Description	This method generates a line integration message to a camera. The next generated line will be acquired. The integration message is described by the parameter CORACQ_PRM_LINE_INTEGRATE_DURATION and CORACQ_PRM_LINE_INTEGRATE_DELAY.  Delay CORACQ_PRM_LINE_INTEGRATE_DELAY Duration CORACQ_PRM_LINE_INTEGRATE_DURATION Polarity N/A
Note	Method 8 is similar to Method 3 except the physical trigger signal pulse is a message.

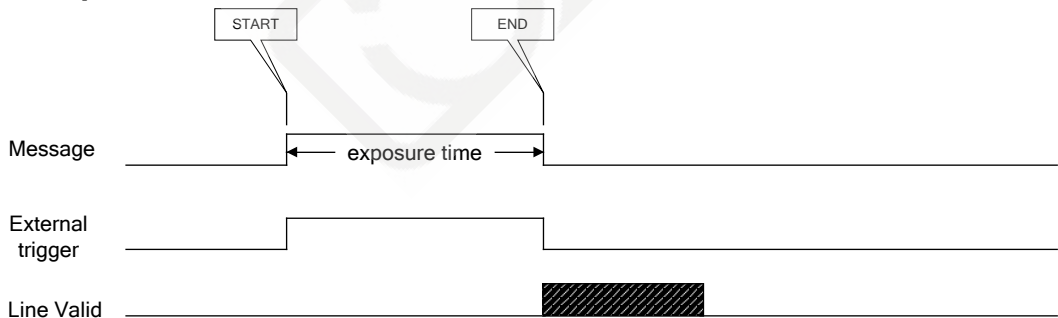
Example:



CORACQ\_VAL\_LINE\_INTEGRATE\_METHOD\_9

Numerical Value	0x00000100 (Line Integration Method #9)
Description	This method generates start/stop line integration messages to a camera. The next generated line will be acquired. The time difference between the start/stop messages represent the integration time and are controlled by a physical external line trigger signal.

Example:

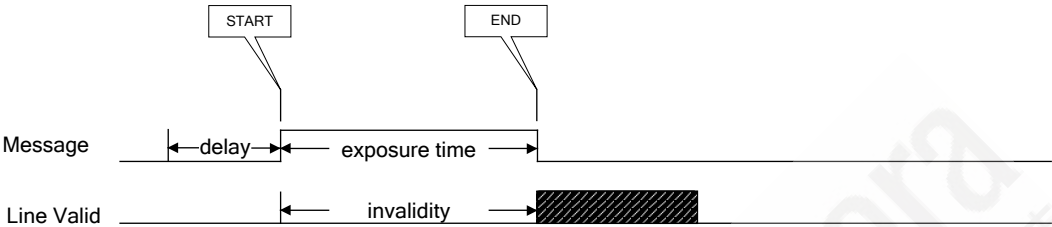


**CORACQ\_VAL\_LINE\_INTEGRATE\_METHOD\_10**

Numerical Value      0x00000200 (Line Integration Method #10)

Description      This method generates start/stop line integration messages to a camera. The next generated line will be acquired. The time difference between the start/stop messages represent the integration time and is controlled by the parameters CORACQ\_PRM\_LINE\_INTEGRATE\_DURATION and CORACQ\_PRM\_LINE\_INTEGRATE\_DELAY.

Delay              CORACQ\_PRM\_LINE\_INTEGRATE\_DELAY  
Duration          CORACQ\_PRM\_LINE\_INTEGRATE\_DURATION  
Polarity          N/A



# Line Trigger Methods

The following line trigger methods are available for line scan cameras:

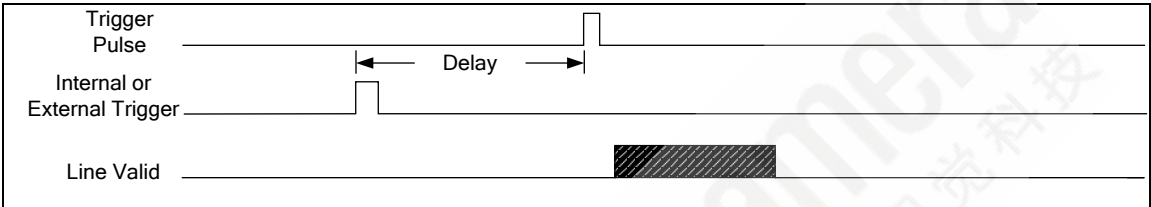
- CORACQ\_VAL\_LINE\_TRIGGER\_METHOD\_1
- CORACQ\_VAL\_LINE\_TRIGGER\_METHOD\_2

## CORACQ\_VAL\_LINE\_TRIGGER\_METHOD\_1

**Numerical Value** 0x00000001 (Line Trigger Method #1)

**Description** Method selection is via the parameter CORACQ\_PRM\_LINE\_TRIGGER\_METHOD. This method generates an asynchronous line trigger pulse to a camera. The next generated frame will be acquired. The trigger pulse is described by the parameters CORACQ\_PRM\_LINE\_TRIGGER\_DURATION and CORACQ\_PRM\_LINE\_TRIGGER\_POLARITY. The delay is set using CORACQ\_PRM\_LINE\_TRIGGER\_DELAY.

### Example:



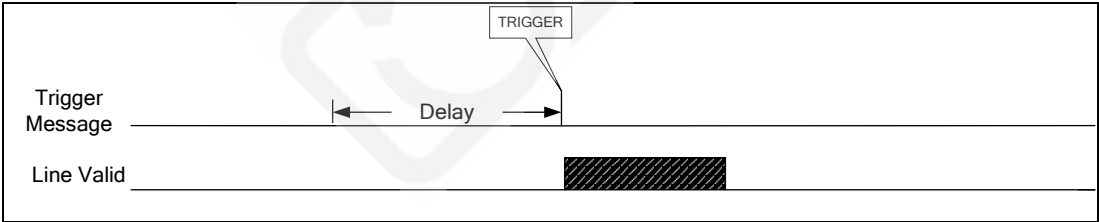
## CORACQ\_VAL\_LINE\_TRIGGER\_METHOD\_2

**Numerical Value** 0x00000002 (Line Trigger Method #2)

**Description** Method selection is via the parameter CORACQ\_PRM\_LINE\_TRIGGER\_METHOD. This method generates a line trigger message to a camera. The next generated frame will be acquired. The delay is set using CORACQ\_PRM\_LINE\_TRIGGER\_DELAY.

**Note** This method is similar to Method #1 except the physical trigger signal is a message.

### Example:



# Time Integrate Methods

The following time integrate methods are available for area scan cameras:

- CORACQ\_VAL\_TIME\_INTEGRATE\_METHOD\_1
- CORACQ\_VAL\_TIME\_INTEGRATE\_METHOD\_2
- CORACQ\_VAL\_TIME\_INTEGRATE\_METHOD\_3
- CORACQ\_VAL\_TIME\_INTEGRATE\_METHOD\_4
- CORACQ\_VAL\_TIME\_INTEGRATE\_METHOD\_5
- CORACQ\_VAL\_TIME\_INTEGRATE\_METHOD\_6
- CORACQ\_VAL\_TIME\_INTEGRATE\_METHOD\_7
- CORACQ\_VAL\_TIME\_INTEGRATE\_METHOD\_8
- CORACQ\_VAL\_TIME\_INTEGRATE\_METHOD\_10
- CORACQ\_VAL\_TIME\_INTEGRATE\_METHOD\_11
- CORACQ\_VAL\_TIME\_INTEGRATE\_METHOD\_12

## CORACQ\_VAL\_TIME\_INTEGRATE\_METHOD\_1

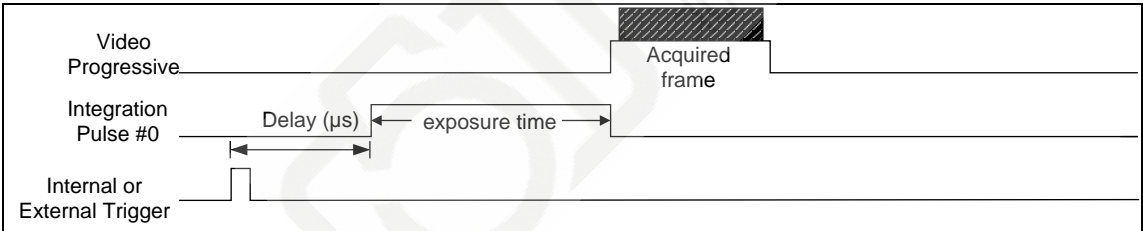
**Numerical Value** 0x00000001 (Time Integration Method #1)

**Description** Method selection is via the parameter CORACQ\_PRM\_TIME\_INTEGRATE\_METHOD. This method generates an asynchronous time integration pulse to a camera (area scan only). The width of the pulse (as specified by the parameter CORACQ\_PRM\_TIME\_INTEGRATE\_DURATION), represents the integration time. The delay between the trigger and start of exposure is specified using the parameter CORACQ\_PRM\_TIME\_INTEGRATE\_DELAY.

The integration pulse is defined by the following parameters:

Delay	CORACQ_PRM_TIME_INTEGRATE_DELAY
Duration	CORACQ_PRM_TIME_INTEGRATE_DURATION
Polarity	CORACQ_PRM_TIME_INTEGRATE_PULSE0_POLARITY

### Example:



**CORACQ\_VAL\_TIME\_INTEGRATE\_METHOD\_2**

**Numerical Value** 0x00000002 (Time Integration Method #2)

**Description** Method selection is via the parameter CORACQ\_PRM\_TIME\_INTEGRATE\_METHOD. This method generates two consecutive trigger pulses on the VD (Vertical Drive) input of the camera (area scan only). The time interval between the end of the two trigger pulses (as specified by the parameter CORACQ\_PRM\_TIME\_INTEGRATE\_DURATION) represents the integration time. The VD trigger pulses are described by the parameters CORACQ\_PRM\_TIME\_INTEGRATE\_PULSE1\_DURATION and CORACQ\_PRM\_TIME\_INTEGRATE\_PULSE1\_POLARITY.

The VD triggers are defined by the following parameters:

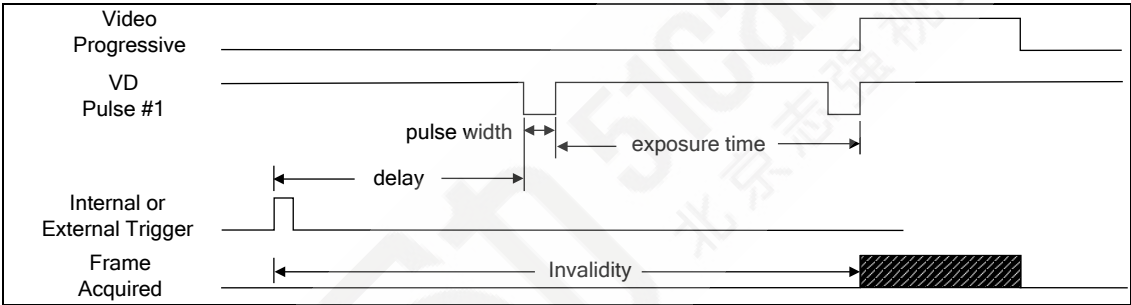
1<sup>st</sup> VD Pulse

- Delay CORACQ\_PRM\_TIME\_INTEGRATE\_DELAY
- Duration CORACQ\_PRM\_TIME\_INTEGRATE\_PULSE1\_DURATION
- Polarity CORACQ\_PRM\_TIME\_INTEGRATE\_PULSE1\_POLARITY

2<sup>nd</sup> VD Pulse

- Delay CORACQ\_PRM\_TIME\_INTEGRATE\_DELAY + CORACQ\_PRM\_TIME\_INTEGRATE\_DURATION
- Duration CORACQ\_PRM\_TIME\_INTEGRATE\_PULSE1\_DURATION
- Polarity CORACQ\_PRM\_TIME\_INTEGRATE\_PULSE1\_POLARITY

**Example:**





CORACQ\_VAL\_TIME\_INTEGRATE\_METHOD\_3

**Numerical Value** 0x00000004 (Time Integration Method #3)

**Description** Also known as the E-Donpisha mode (area scan only). Method selection is via the parameter CORACQ\_PRM\_TIME\_INTEGRATE\_METHOD. This method generates an integration pulse on the camera trigger input, followed by a trigger pulse on the camera VD input. The width of the integration pulse (as specified by the parameter CORACQ\_PRM\_TIME\_INTEGRATE\_DURATION) represents the integration time. The polarity of the integration pulse is specified with the CORACQ\_PRM\_TIME\_INTEGRATE\_PULSE0\_POLARITY parameter. The delay before the integration pulse is set using CORACQ\_PRM\_TIME\_INTEGRATE\_DELAY. The VD trigger pulse is described by the parameters CORACQ\_PRM\_TIME\_INTEGRATE\_PULSE1\_DELAY, CORACQ\_PRM\_TIME\_INTEGRATE\_PULSE1\_DURATION and CORACQ\_PRM\_TIME\_INTEGRATE\_PULSE1\_POLARITY, where the delay is the interval between the end of the integration pulse and the start of the VD trigger pulse.

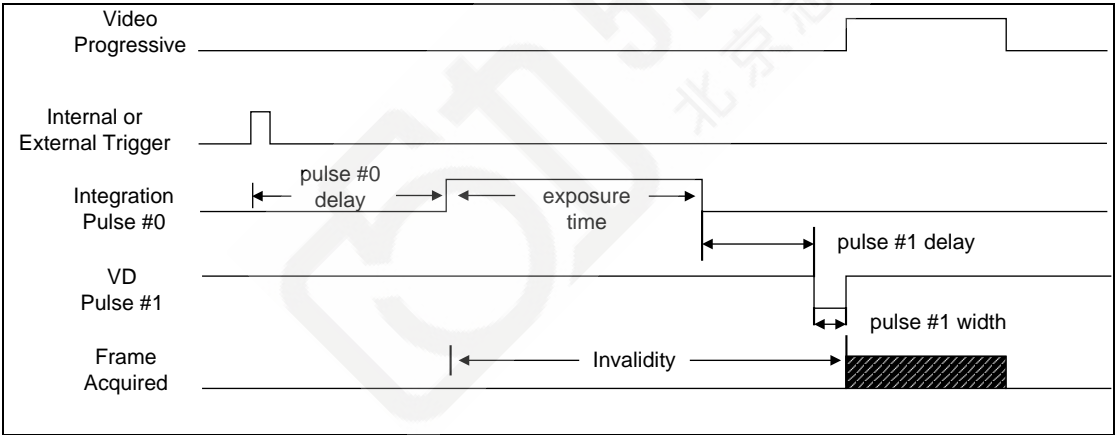
1<sup>st</sup> Integration Pulse

Delay CORACQ\_PRM\_TIME\_INTEGRATE\_DELAY  
Duration CORACQ\_PRM\_TIME\_INTEGRATE\_DURATION  
Polarity CORACQ\_PRM\_TIME\_INTEGRATE\_PULSE0\_POLARITY

2<sup>nd</sup> VD Pulse

Delay CORACQ\_PRM\_TIME\_INTEGRATE\_DELAY +  
CORACQ\_PRM\_TIME\_INTEGRATE\_DURATION +  
CORACQ\_PRM\_TIME\_INTEGRATE\_PULSE1\_DELAY  
Duration CORACQ\_PRM\_TIME\_INTEGRATE\_PULSE1\_DURATION  
Polarity CORACQ\_PRM\_TIME\_INTEGRATE\_PULSE1\_POLARITY

Example:



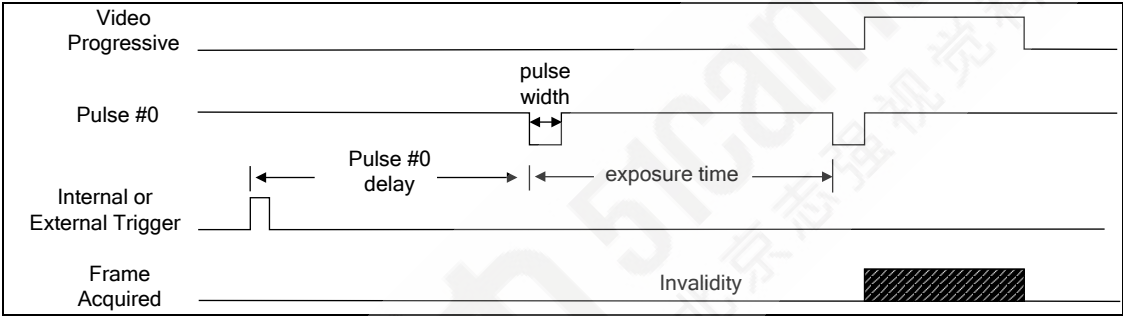
**CORACQ\_VAL\_TIME\_INTEGRATE\_METHOD\_4**

**Numerical Value** 0x00000008 (Time Integration Method #4)

**Description** Method selection is via the parameter CORACQ\_PRM\_TIME\_INTEGRATE\_METHOD. This method generates two consecutive trigger pulses on the camera trigger input. The time interval between the start of the two trigger pulses (as specified by the parameter CORACQ\_PRM\_TIME\_INTEGRATE\_DURATION) represents the integration time. The trigger pulses are described by the parameters CORACQ\_PRM\_TIME\_INTEGRATE\_PULSE0\_DURATION and CORACQ\_PRM\_TIME\_INTEGRATE\_PULSE0\_POLARITY.

1 <sup>st</sup> Pulse	
Delay	CORACQ_PRM_TIME_INTEGRATE_DELAY
Duration	CORACQ_PRM_TIME_INTEGRATE_PULSE0_DURATION
Polarity	CORACQ_PRM_TIME_INTEGRATE_PULSE0_POLARITY
2 <sup>nd</sup> Pulse	
Delay	CORACQ_PRM_TIME_INTEGRATE_DELAY + CORACQ_PRM_TIME_INTEGRATE_DURATION
Duration	CORACQ_PRM_TIME_INTEGRATE_PULSE1_DURATION
Polarity	CORACQ_PRM_TIME_INTEGRATE_PULSE1_POLARITY

**Example:**

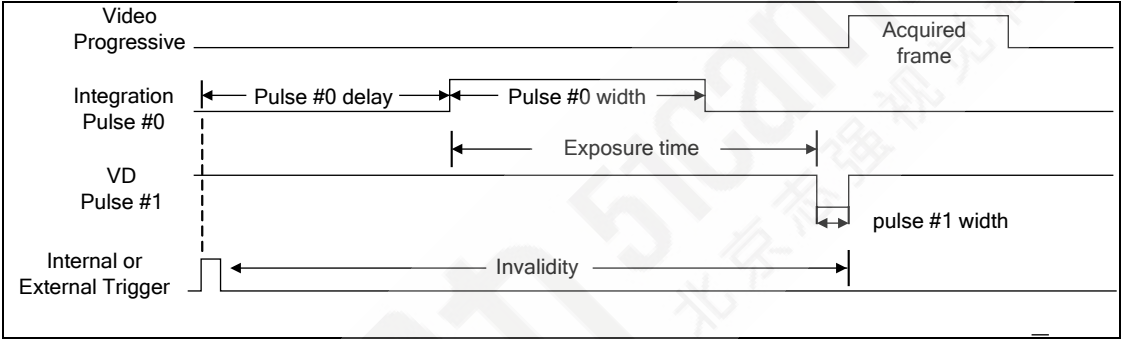


**CORACQ\_VAL\_TIME\_INTEGRATE\_METHOD\_5**

**Numerical Value** 0x00000010 (Time Integration Method #5)

**Description** Method selection is via the parameter CORACQ\_PRM\_TIME\_INTEGRATE\_METHOD. This method generates a trigger pulse (#0) on the camera trigger input, followed by a trigger pulse (#1) on the camera VD input. The interval between the start of the two pulses (as specified by the parameter CORACQ\_PRM\_TIME\_INTEGRATE\_DURATION) represents the integration time. The trigger pulse (#0) on the camera trigger input is defined by the parameters CORACQ\_PRM\_TIME\_INTEGRATE\_PULSE0\_DURATION and CORACQ\_PRM\_TIME\_INTEGRATE\_PULSE0\_POLARITY. The VD trigger pulse is defined by the parameters CORACQ\_PRM\_TIME\_INTEGRATE\_PULSE1\_DURATION and CORACQ\_PRM\_TIME\_INTEGRATE\_PULSE1\_POLARITY.

1 <sup>st</sup> Pulse	
Delay	CORACQ_PRM_TIME_INTEGRATE_DELAY
Duration	CORACQ_PRM_TIME_INTEGRATE_PULSE0_DURATION
Polarity	CORACQ_PRM_TIME_INTEGRATE_PULSE0_POLARITY
2 <sup>nd</sup> Pulse	
Delay	CORACQ_PRM_TIME_INTEGRATE_DELAY + CORACQ_PRM_TIME_INTEGRATE_DURATION
Duration	CORACQ_PRM_TIME_INTEGRATE_PULSE1_DURATION
Polarity	CORACQ_PRM_TIME_INTEGRATE_PULSE1_POLARITY



**CORACQ\_VAL\_TIME\_INTEGRATE\_METHOD\_6**

**Numerical Value** 0x00000020 (Time Integration Method #6)

**Description** Method selection is via the parameter CORACQ\_PRM\_TIME\_INTEGRATE\_METHOD. This method generates a trigger pulse (#0) on the camera trigger input, followed by a trigger pulse (#1) on the camera VD input. The interval between the start of pulse #0 and end of pulse #1 (as specified by the parameter CORACQ\_PRM\_TIME\_INTEGRATE\_DURATION) is the integration time. The trigger pulse (#0) on the camera trigger input is defined by the parameters CORACQ\_PRM\_TIME\_INTEGRATE\_PULSE0\_DURATION and CORACQ\_PRM\_TIME\_INTEGRATE\_PULSE0\_POLARITY. The VD trigger pulse is defined by the parameters CORACQ\_PRM\_TIME\_INTEGRATE\_PULSE1\_DURATION and CORACQ\_PRM\_TIME\_INTEGRATE\_PULSE1\_POLARITY.

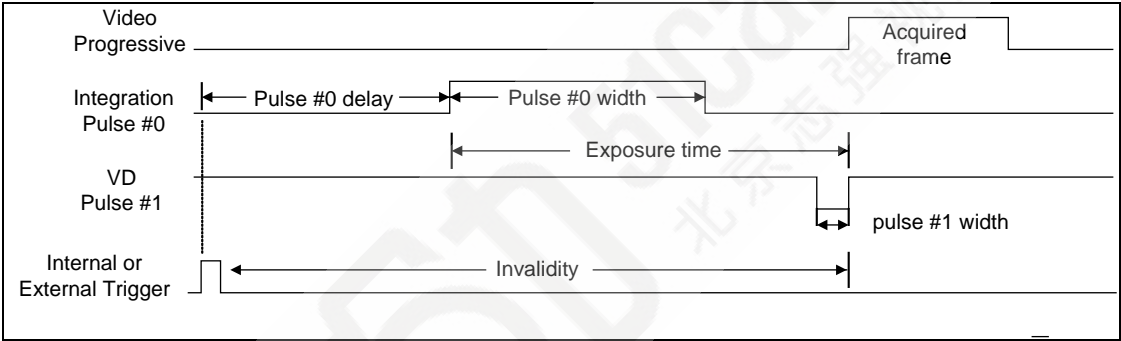
1<sup>st</sup> Integration Pulse

Delay CORACQ\_PRM\_TIME\_INTEGRATE\_DELAY  
Duration CORACQ\_PRM\_TIME\_INTEGRATE\_PULSE0\_DURATION  
Polarity CORACQ\_PRM\_TIME\_INTEGRATE\_PULSE0\_POLARITY

2<sup>nd</sup> VD Pulse

Delay CORACQ\_PRM\_TIME\_INTEGRATE\_DELAY +  
CORACQ\_PRM\_TIME\_INTEGRATE\_DURATION +  
CORACQ\_PRM\_TIME\_INTEGRATE\_PULSE1\_DURATION  
Duration CORACQ\_PRM\_TIME\_INTEGRATE\_PULSE1\_DURATION  
Polarity CORACQ\_PRM\_TIME\_INTEGRATE\_PULSE1\_POLARITY

**Example:**



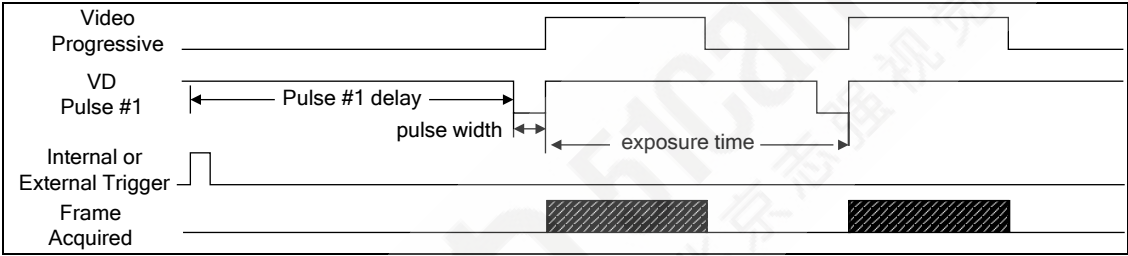
**CORACQ\_VAL\_TIME\_INTEGRATE\_METHOD\_7**

**Numerical Value** 0x00000040 (Time Integration Method #7)

**Description** Method selection is via the parameter CORACQ\_PRM\_TIME\_INTEGRATE\_METHOD. This method generates two consecutive trigger pulses (#1) on the camera VD (Vertical Drive) input. The time interval between the end of the two trigger pulses (as specified by the parameter CORACQ\_PRM\_TIME\_INTEGRATE\_DURATION) is the integration time. This method differs from method #2, since a valid frame is available during the integration time. The VD trigger pulses are described by the parameters CORACQ\_PRM\_TIME\_INTEGRATE\_PULSE1\_DURATION and CORACQ\_PRM\_TIME\_INTEGRATE\_PULSE1\_POLARITY.

1 <sup>st</sup> Pulse	
Delay	CORACQ_PRM_TIME_INTEGRATE_DELAY
Duration	CORACQ_PRM_TIME_INTEGRATE_PULSE1_DURATION
Polarity	CORACQ_PRM_TIME_INTEGRATE_PULSE1_POLARITY
2 <sup>nd</sup> Pulse	
Delay	CORACQ_PRM_TIME_INTEGRATE_DELAY + CORACQ_PRM_TIME_INTEGRATE_DURATION
Duration	CORACQ_PRM_TIME_INTEGRATE_PULSE1_DURATION
Polarity	CORACQ_PRM_TIME_INTEGRATE_PULSE1_POLARITY

**Example:**

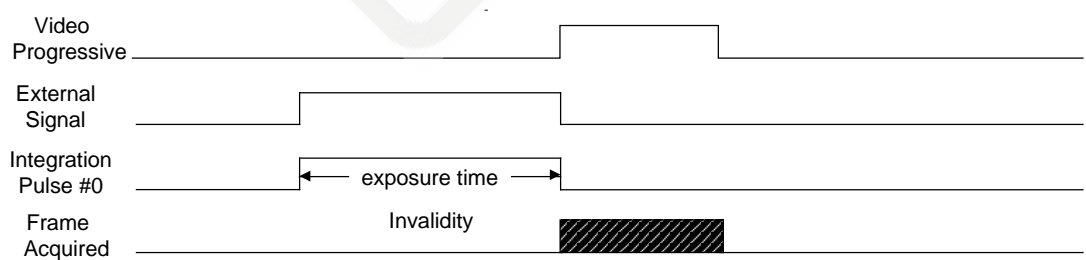


**CORACQ\_VAL\_TIME\_INTEGRATE\_METHOD\_8**

**Numerical Value** 0x00000080 (Time Integration Method #8)

**Description** Method selection is via the parameter CORACQ\_PRM\_TIME\_INTEGRATE\_METHOD. This method generates an asynchronous time integration pulse (#0) to a camera. The width of the pulse represents the integration time and is controlled by an external signal.

**Example:**

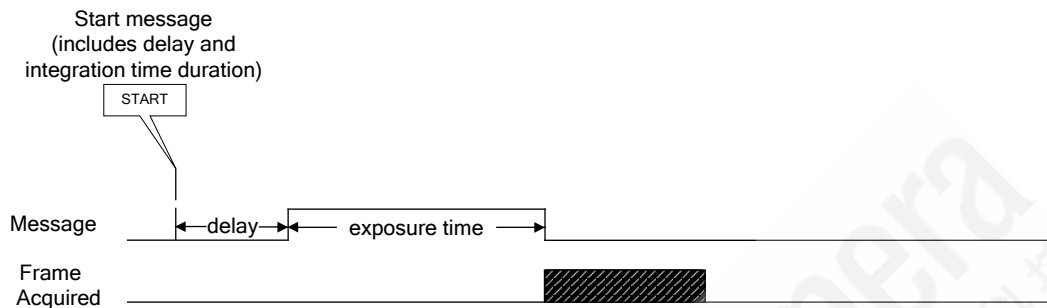


**CORACQ\_VAL\_TIME\_INTEGRATE\_METHOD\_10**

**Numerical Value** 0x00000200 (Time Integration Method #10)

**Description** This method generates a time integration message to a camera. The next generated frame will be acquired. The integration message is described by the parameters CORACQ\_PRM\_TIME\_INTEGRATE\_DURATION and CORACQ\_PRM\_TIME\_INTEGRATE\_DELAY.

Delay	CORACQ_PRM_TIME_INTEGRATE_DELAY
Duration	CORACQ_PRM_TIME_INTEGRATE_DURATION
Polarity	N/A

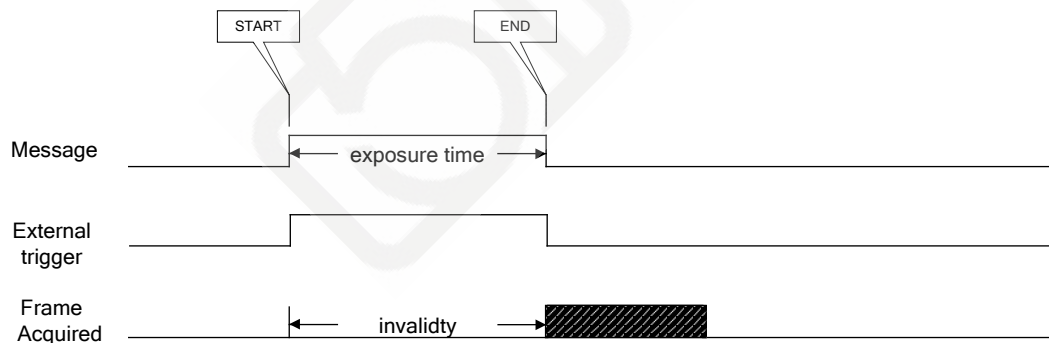


**CORACQ\_VAL\_TIME\_INTEGRATE\_METHOD\_11**

**Numerical Value** 0x00000400 (Time Integration Method #11)

**Description** This method generates start/stop frame integration messages to a camera. The next generated frame will be acquired. The time difference between the start/stop messages represent the integration time and are controlled by a physical external frame trigger signal.

**Example:**



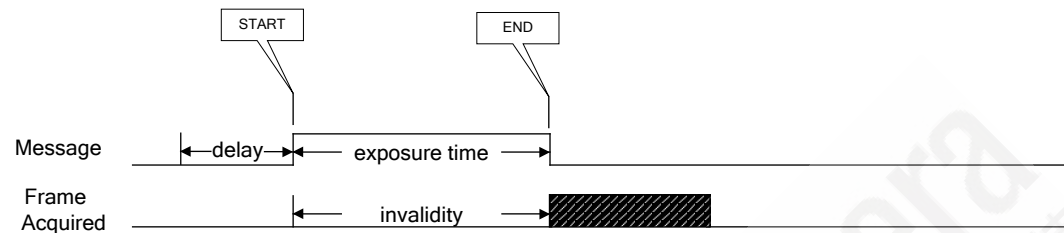
**CORACQ\_VAL\_TIME\_INTEGRATE\_METHOD\_12**

**Numerical Value** 0x00000400 (Time Integration Method #12)

**Description** This method generates start/stop frame integration messages to a camera. The next generated frame will be acquired. The time difference between the start/stop messages represent the integration time and are controlled by the parameters CORACQ\_PRM\_TIME\_INTEGRATE\_DURATION and CORACQ\_PRM\_TIME\_INTEGRATE\_DELAY.

Delay CORACQ\_PRM\_TIME\_INTEGRATE\_DELAY  
Duration CORACQ\_PRM\_TIME\_INTEGRATE\_DURATION  
Polarity N/A

**Example:**





# Strobe Methods

The following strobe methods are available:

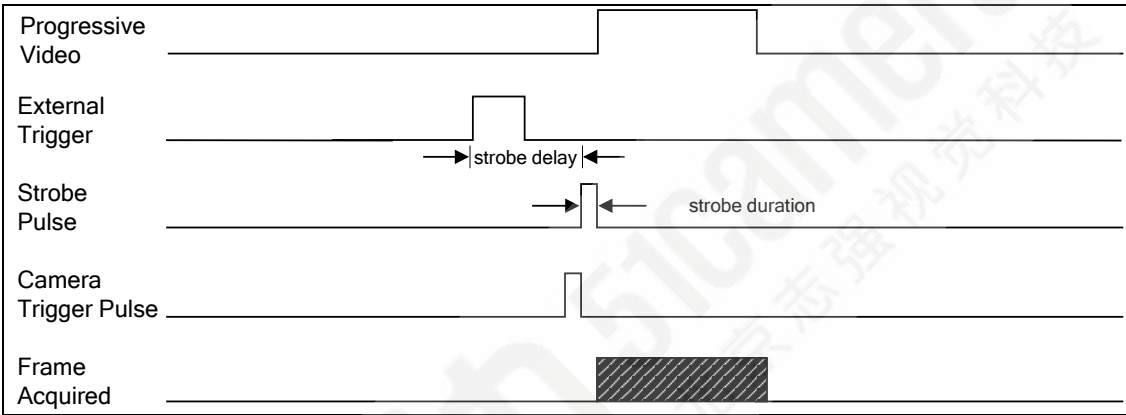
- CORACQ\_VAL\_STROBE\_METHOD\_1
- CORACQ\_VAL\_STROBE\_METHOD\_2
- CORACQ\_VAL\_STROBE\_METHOD\_3
- CORACQ\_VAL\_STROBE\_METHOD\_4
- CORACQ\_VAL\_STROBE\_METHOD\_5

## CORACQ\_VAL\_STROBE\_METHOD\_1

**Numerical Value** 0x00000001 (Strobe Method #1)

**Description** Method selection is via the parameter CORACQ\_PRM\_STROBE\_METHOD. This method generates a synchronous strobe pulse relative to a trigger signal (external, internal, software) depending on the mode of operation. The strobe pulse is described by the parameters CORACQ\_PRM\_STROBE\_DELAY, CORACQ\_PRM\_STROBE\_DURATION, and CORACQ\_PRM\_STROBE\_POLARITY.

### Example:

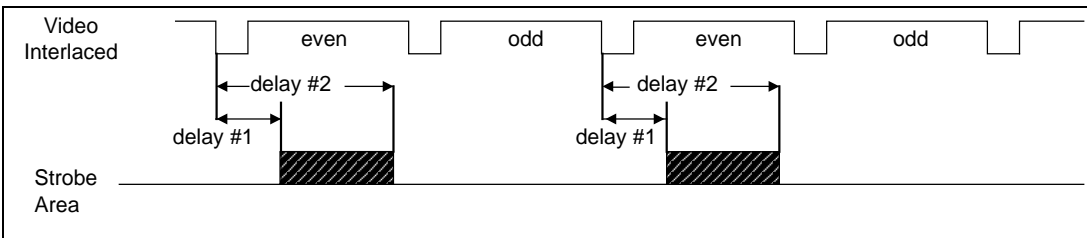


## CORACQ\_VAL\_STROBE\_METHOD\_2

**Numerical Value** 0x00000002 (Strobe Method #2)

**Description** Method selection is via the parameter CORACQ\_PRM\_STROBE\_METHOD. This method generates an asynchronous strobe pulse. The pulse is generated outside the region comprising the start of a vertical sync up to the specified strobe delay, but not later than the second strobe delay. If interlaced video is acquired, then the strobe will be generated on the field previous to the acquired frame: even if the field ordering is odd-even (typical), odd if the field ordering even-odd, or any if the field ordering is next two fields. The strobe pulse is described by the parameters CORACQ\_PRM\_STROBE\_DELAY, CORACQ\_PRM\_STROBE\_DELAY\_2, CORACQ\_PRM\_STROBE\_DURATION, and CORACQ\_PRM\_STROBE\_POLARITY.

### Example: Interlaced, Odd-Even acquisition

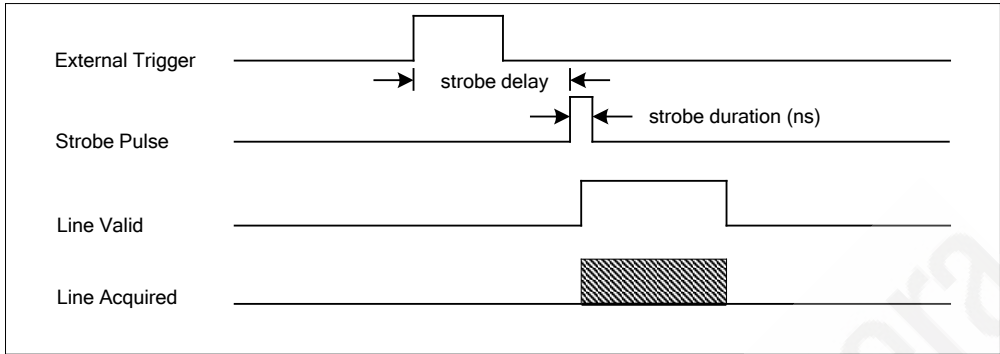


### CORACQ\_VAL\_STROBE\_METHOD\_3

**Numerical Value** 0x00000004 (Strobe Method #3)

**Description** Method selection is performed via the parameter CORACQ\_PRM\_STROBE\_METHOD. This method generates a synchronous strobe pulse relative to a line trigger signal (external, internal, software) depending on the mode of operation. The strobe pulse is described by the parameters CORACQ\_PRM\_STROBE\_DELAY, CORACQ\_PRM\_STROBE\_DURATION and CORACQ\_PRM\_STROBE\_POLARITY.

#### External: External Line Trigger



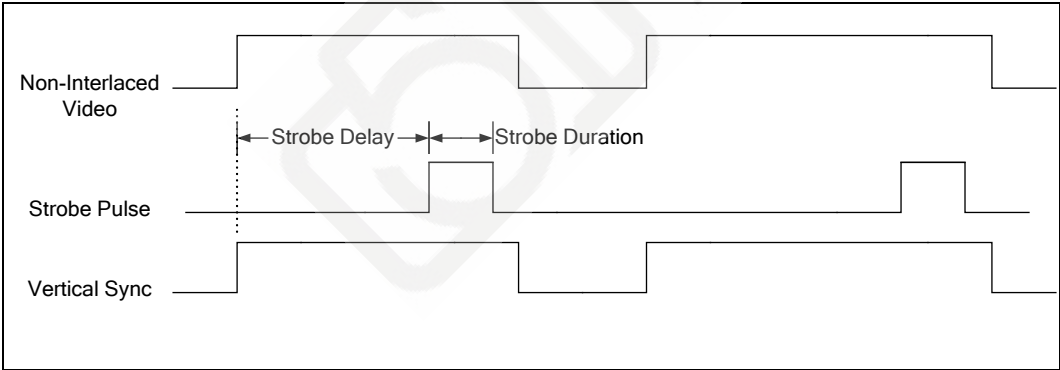
### CORACQ\_VAL\_STROBE\_METHOD\_4

**Numerical Value** 0x00000008 (Strobe Method #4)

**Description** Method selection is via the parameter CORACQ\_PRM\_STROBE\_METHOD. This method generates a synchronous strobe pulse relative to a vertical sync signal.. The strobe pulse is described by the parameters CORACQ\_PRM\_STROBE\_DELAY, CORACQ\_PRM\_STROBE\_DURATION, and CORACQ\_PRM\_STROBE\_POLARITY.

Both area scan and line scan cameras support this method. Note that in linescan, there will be one strobe pulse output per virtual frame.

#### Example:

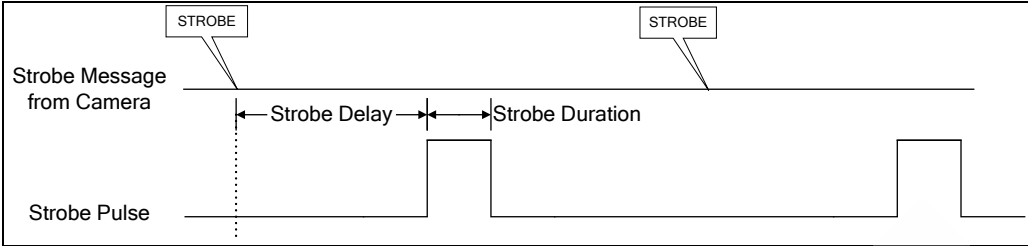


**CORACQ\_VAL\_STROBE\_METHOD\_5**

**Numerical Value** 0x00000010 (Strobe Method #5)

**Description** This method generates a synchronous strobe pulse relative to a trigger message received from a camera. The strobe pulse is described by the parameters CORACQ\_PRM\_STROBE\_DELAY, CORACQ\_PRM\_STROBE\_DURATION and CORACQ\_PRM\_STROBE\_POLARITY.

**Example:**

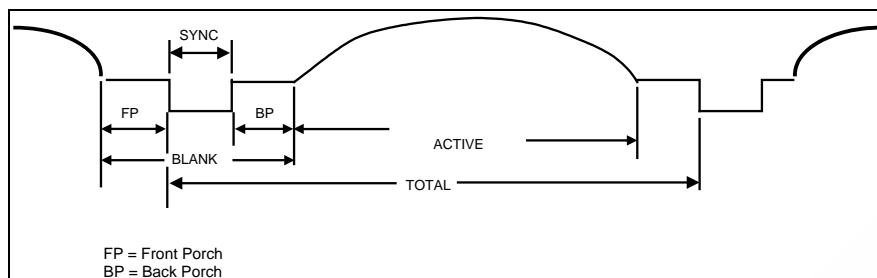


# Camera Video Timing Definitions

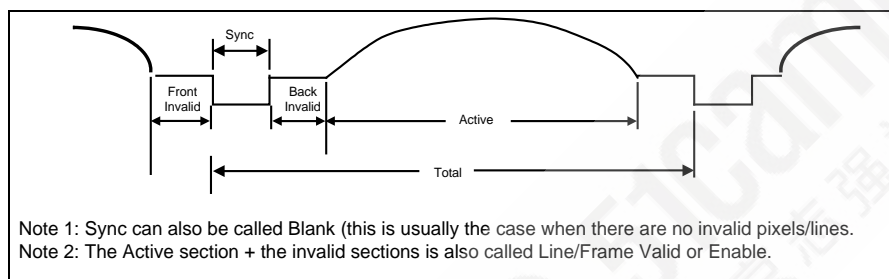
Generic camera timing diagrams describe the terminology and relationships used in Sapera LT applications. Topics covered are:

- Area Scan Analog Video Timings
- Area Scan Digital Video Timings
- Linescan Video Timings

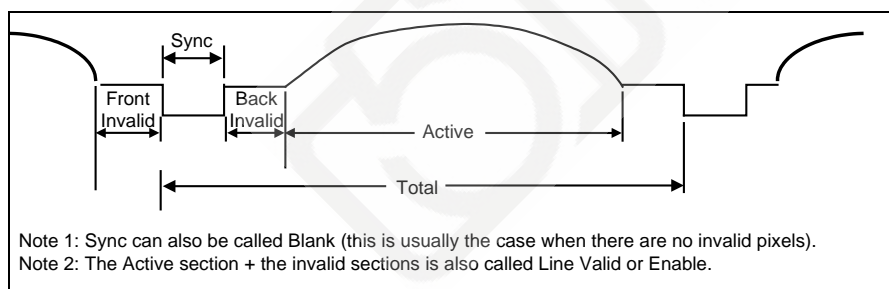
## Analog Area Scan Video Timings



## Digital Area Scan Video Timings



## Linescan Video Timings



---

## Custom Camera Control I/O Description

The acquisition module currently has specific parameters to control the following standard inputs/outputs: integration, camera trigger, camera reset, and strobe.

Custom camera I/Os are useful to control non-standard inputs/outputs from a camera, such as Gain and Binning. These custom controls are defined in the CCA file. The description of a custom I/O includes a label, the number of I/O bits used, the signal level of the I/Os (TTL/RS-422/LVDS), the direction of the I/Os (Inputs or Outputs), the polarity of the control for an active signal, and a default value in the case of an Output. The custom camera I/O information in the CCA file is passed to the acquisition module through the parameter `CORACQ_PRM_CAM_IO_CONTROL`. This is a complex parameter that can accommodate up to 32 different controls. The size of the parameter is therefore  $32 * \text{size of}(\text{CORACQ\_CAM\_IO\_CONTROL})$ .

When applied, the driver scans the entries until a control specifies that 0 I/O bits is needed. It is therefore recommended to first initialize the `CORACQ_CAM_IO_CONTROL` to 0 before filling in control definitions. The driver assigns the necessary I/Os in an orderly fashion, following the order in which they are defined in the CCA/CCF file. At the function level, the I/O assignment can be setup by using the standard method of loading a CCA/CCF file (`CorCamLoad + CorAcqSetPrms`), or the `CorAcqSetPrmEx` function can be simply called with an `CORACQ_PRM_CAM_IO_CONTROL` parameter. To get/set the value of an I/O, use the Sapera functions (`CorAcqDetectSync` and `CorAcqSetCamIOControl`) where the label argument is the string representation of the I/O control as specified in the CCA/CCF file.



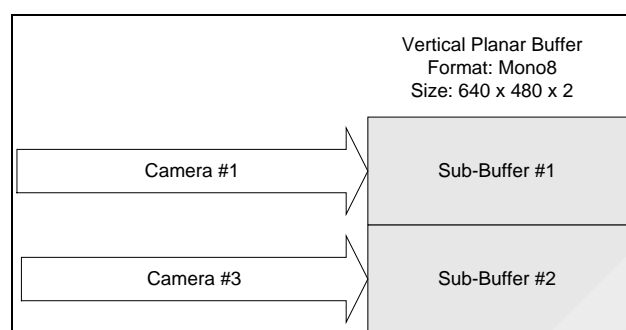
---

## Planar Input Sources Description

The planar input sources parameter, **CORACQ\_PRM\_PLANAR\_INPUT\_SOURCES**, is used to acquire from multiple synchronized video sources. It enables selecting which input sources will be acquired into a vertical planar buffer. The parameter value is a bit field. Each bit represents an acquisition input. If the bit is 1, then the source connected to that input are acquired into a vertical planar buffer. All video sources must be synchronized together. The vertical planar buffer format is simply a buffer which has been created with a height that is 'n' times longer than the size of one video source vertical resolution, 'n' being the number of inputs that are to be acquired synchronously. The acquisition function will automatically divide the buffer into sub-buffers which are assigned to each input. Important: the parameter **CORACQ\_PRM\_CAMSEL** is used to select the sync signal source.

### Example:

If **CORACQ\_PRM\_PLANAR\_INPUT\_SOURCES** = 0x00000005, then bit 0 and 2 are active, and camera #1 and #3 will be acquired from.





# Advanced Acquisition Control

## Introduction

The Acquisition Module controls the acquisition device and its functions. It is used in conjunction with the VIC and Camera modules.

## Camera Related Parameters

The camera related parameters, as their name implies, modelize the video source irrelevant of the actual source itself (camera, etc.). These parameters define the video capabilities and modes of operation.

### Camera Related Parameters By Groups

General	
CORACQ_PRM_CAM_NAME	CORACQ_PRM_CAM_COMPANY_NAME
CORACQ_PRM_CAM_MODEL_NAME	
Signal Description	
CORACQ_PRM_CHANNEL	CORACQ_PRM_CHANNELS_ORDER
CORACQ_PRM_COUPLING	CORACQ_PRM_FIELD_ORDER
CORACQ_PRM_FRAME	CORACQ_PRM_INTERFACE
CORACQ_PRM_PIXEL_DEPTH	CORACQ_PRM_SCAN
CORACQ_PRM_SIGNAL	CORACQ_PRM_TAP_OUTPUT
CORACQ_PRM_TAP_1_DIRECTION	CORACQ_PRM_TAP_2_DIRECTION
CORACQ_PRM_TAP_3_DIRECTION	CORACQ_PRM_TAP_4_DIRECTION
CORACQ_PRM_TAP_5_DIRECTION	CORACQ_PRM_TAP_6_DIRECTION
CORACQ_PRM_TAP_7_DIRECTION	CORACQ_PRM_TAP_8_DIRECTION
CORACQ_PRM_TAPS	CORACQ_PRM_VIDEO
CORACQ_PRM_VIDEO_LEVEL_MAX	CORACQ_PRM_VIDEO_LEVEL_MIN
CORACQ_PRM_VIDEO_STD	
Signal Timings	
CORACQ_PRM_HACTIVE	CORACQ_PRM_HBACK_INVALID
CORACQ_PRM_HBACK_PORCH	CORACQ_PRM_HFRONT_INVALID
CORACQ_PRM_HFRONT_PORCH	CORACQ_PRM_HSYNC
CORACQ_PRM_VACTIVE	CORACQ_PRM_VBACK_INVALID
CORACQ_PRM_VBACK_PORCH	CORACQ_PRM_VFRONT_INVALID
CORACQ_PRM_VFRONT_PORCH	CORACQ_PRM_VSYNC
CORACQ_PRM_TIMESLOT	
Pixel Clock	
CORACQ_PRM_PIXEL_CLK_DETECTION	CORACQ_PRM_PIXEL_CLK_EXT
CORACQ_PRM_PIXEL_CLK_INT	CORACQ_PRM_PIXEL_CLK_11
CORACQ_PRM_PIXEL_CLK_SRC	

## Synchronization Signals

CORACQ_PRM_HSYNC_POLARITY	CORACQ_PRM_SYNC
CORACQ_PRM_VSYNC_POLARITY	

## Control Signals

CORACQ_PRM_CAM_CONTROL_DURING_READOUT	
CORACQ_PRM_CAM_LINE_TRIGGER_FREQ_MAX	
CORACQ_PRM_CAM_LINE_TRIGGER_FREQ_MIN	
CORACQ_PRM_CAM_RESET_DURATION	
CORACQ_PRM_CAM_RESET_POLARITY	
CORACQ_PRM_CAM_RESET_METHOD	
CORACQ_PRM_CAM_TIME_INTEGRATE_DURATION_MAX	
CORACQ_PRM_CAM_TIME_INTEGRATE_DURATION_MIN	
CORACQ_PRM_CAM_TRIGGER_DURATION	
CORACQ_PRM_CAM_TRIGGER_METHOD	CORACQ_PRM_CAM_TRIGGER_POLARITY
CORACQ_PRM_DATA_VALID_ENABLE	CORACQ_PRM_DATA_VALID_POLARITY
CORACQ_PRM_FRAME_INTEGRATE_METHOD	
CORACQ_PRM_FRAME_INTEGRATE_POLARITY	
CORACQ_PRM_LINE_INTEGRATE_METHOD	
CORACQ_PRM_LINE_INTEGRATE_PULSE0_DELAY	
CORACQ_PRM_LINE_INTEGRATE_PULSE0_DURATION	
CORACQ_PRM_LINE_INTEGRATE_PULSE0_POLARITY	
CORACQ_PRM_LINE_INTEGRATE_PULSE1_DELAY	
CORACQ_PRM_LINE_INTEGRATE_PULSE1_DURATION	
CORACQ_PRM_LINE_INTEGRATE_PULSE1_POLARITY	
CORACQ_PRM_LINE_TRIGGER_DELAY	
CORACQ_PRM_LINE_TRIGGER_DURATION	CORACQ_PRM_LINE_TRIGGER_METHOD
CORACQ_PRM_LINE_TRIGGER_POLARITY	CORACQ_PRM_LINESCAN_DIRECTION
CORACQ_PRM_LINESCAN_DIRECTION_POLARITY	
CORACQ_PRM_TIME_INTEGRATE_METHOD	
CORACQ_PRM_TIME_INTEGRATE_PULSE0_DELAY	
CORACQ_PRM_TIME_INTEGRATE_PULSE0_DURATION	
CORACQ_PRM_TIME_INTEGRATE_PULSE0_POLARITY	
CORACQ_PRM_TIME_INTEGRATE_PULSE1_DELAY	
CORACQ_PRM_TIME_INTEGRATE_PULSE1_DURATION	
CORACQ_PRM_TIME_INTEGRATE_PULSE1_POLARITY	
CORACQ_PRM_WEN_POLARITY	

## Connector Description

CORACQ_PRM_CAMLINK_CONFIGURATION
CORACQ_PRM_CONNECTOR_EXPOSURE_INPUT
CORACQ_PRM_CONNECTOR_HD_INPUT
CORACQ_PRM_CONNECTOR_LINE_INTEGRATE_INPUT
CORACQ_PRM_CONNECTOR_LINE_TRIGGER_INPUT
CORACQ_PRM_CONNECTOR_LINESCAN_DIRECTION_INPUT
CORACQ_PRM_CONNECTOR_PIXEL_CLK_OUTPUT
CORACQ_PRM_CONNECTOR_RESET_TRIGGER_INPUT

**Custom Camera I/O Control Signals**

CORACQ\_PRM\_CAM\_IO\_CONTROL

**Camera Related Parameters By ID**

ID	Parameter
0x00	CORACQ_PRM_CHANNEL
0x01	CORACQ_PRM_FRAME
0x02	CORACQ_PRM_INTERFACE
0x03	CORACQ_PRM_SCAN
0x04	CORACQ_PRM_SIGNAL
0x05	CORACQ_PRM_VIDEO
0x06	CORACQ_PRM_PIXEL_DEPTH
0x07	CORACQ_PRM_VIDEO_STD
0x08	<i>Reserved</i>
0x09	CORACQ_PRM_FIELD_ORDER
0x0a	CORACQ_PRM_HACTIVE
0x0b	CORACQ_PRM_HSYNC
0x0c	CORACQ_PRM_VACTIVE
0x0d	CORACQ_PRM_VSYNC
0x0e	CORACQ_PRM_HFRONT_PORCH
0x0f	CORACQ_PRM_HBACK_PORCH
0x10	CORACQ_PRM_COUPLING
0x11	<i>Reserved</i>
0x12	CORACQ_PRM_VFRONT_PORCH
0x13	CORACQ_PRM_VBACK_PORCH
0x14	CORACQ_PRM_HFRONT_INVALID
0x15	CORACQ_PRM_HBACK_INVALID
0x16	CORACQ_PRM_VFRONT_INVALID
0x17	CORACQ_PRM_VBACK_INVALID
0x18	CORACQ_PRM_PIXEL_CLK_SRC
0x19	CORACQ_PRM_PIXEL_CLK_INT
0x1a	CORACQ_PRM_PIXEL_CLK_11
0x1b	CORACQ_PRM_PIXEL_CLK_EXT
0x1c	CORACQ_PRM_SYNC
0x1d	CORACQ_PRM_HSYNC_POLARITY
0x1e	CORACQ_PRM_VSYNC_POLARITY
0x1f	CORACQ_PRM_FRAME_INTEGRATE_METHOD
0x20	CORACQ_PRM_FRAME_INTEGRATE_POLARITY
0x21	CORACQ_PRM_TIME_INTEGRATE_METHOD
0x22	CORACQ_PRM_TIME_INTEGRATE_PULSE0_POLARITY
0x23	CORACQ_PRM_CAM_TRIGGER_METHOD
0x24	CORACQ_PRM_CAM_TRIGGER_POLARITY
0x25	CORACQ_PRM_CAM_TRIGGER_DURATION
0x26	CORACQ_PRM_CAM_RESET_METHOD

0x27	CORACQ_PRM_CAM_RESET_POLARITY
0x28	CORACQ_PRM_CAM_RESET_DURATION
0x29	CORACQ_PRM_CAM_NAME
0x2a	CORACQ_PRM_LINE_INTEGRATE_METHOD
0x2b	CORACQ_PRM_LINE_INTEGRATE_PULSE0_POLARITY
0x2c	CORACQ_PRM_LINE_INTEGRATE_PULSE0_DELAY
0x2d	CORACQ_PRM_LINE_TRIGGER_METHOD
0x2e	CORACQ_PRM_LINE_TRIGGER_POLARITY
0x2f	CORACQ_PRM_LINE_TRIGGER_DELAY
0x30	CORACQ_PRM_LINE_TRIGGER_DURATION
0x31	CORACQ_PRM_TAPS
0x32	CORACQ_PRM_TAP_OUTPUT
0x33	CORACQ_PRM_TAP_1_DIRECTION
0x34	CORACQ_PRM_TAP_2_DIRECTION
0x35	CORACQ_PRM_TAP_3_DIRECTION
0x36	CORACQ_PRM_TAP_4_DIRECTION
0x37	CORACQ_PRM_TAP_5_DIRECTION
0x38	CORACQ_PRM_TAP_6_DIRECTION
0x39	CORACQ_PRM_TAP_7_DIRECTION
0x3a	CORACQ_PRM_TAP_8_DIRECTION
0x3b	CORACQ_PRM_PIXEL_CLK_DETECTION
0x3c	CORACQ_PRM_CHANNELS_ORDER
0x3d	CORACQ_PRM_LINESCAN_DIRECTION
0x3e	CORACQ_PRM_LINESCAN_DIRECTION_POLARITY
0x3f	CORACQ_PRM_CAM_LINE_TRIGGER_FREQ_MIN
0x40	CORACQ_PRM_CAM_LINE_TRIGGER_FREQ_MAX
0x41	CORACQ_PRM_CAM_TIME_INTEGRATE_DURATION_MIN
0x42	CORACQ_PRM_CAM_TIME_INTEGRATE_DURATION_MAX
0x43	CORACQ_PRM_CONNECTOR_HD_INPUT
0x44	CORACQ_PRM_CONNECTOR_VD_INPUT
0x45	CORACQ_PRM_CONNECTOR_RESET_TRIGGER_INPUT
0x46	CORACQ_PRM_TIME_INTEGRATE_PULSE1_POLARITY
0x47	CORACQ_PRM_TIME_INTEGRATE_PULSE1_DELAY
0x48	CORACQ_PRM_TIME_INTEGRATE_PULSE1_DURATION
0x49	CORACQ_PRM_CAM_IO_CONTROL
0x4a	CORACQ_PRM_CONNECTOR_EXPOSURE_INPUT
0x4b	CORACQ_PRM_TIME_INTEGRATE_PULSE0_DELAY
0x4c	CORACQ_PRM_TIME_INTEGRATE_PULSE0_DURATION
0x4d	CORACQ_PRM_LINE_INTEGRATE_PULSE1_POLARITY
0x4e	CORACQ_PRM_LINE_INTEGRATE_PULSE1_DELAY
0x4f	CORACQ_PRM_LINE_INTEGRATE_PULSE1_DURATION
0x50	CORACQ_PRM_LINE_INTEGRATE_PULSE0_DURATION
0x51	CORACQ_PRM_CAM_COMPANY_NAME
0x52	CORACQ_PRM_CAM_MODEL_NAME
0x53	CORACQ_PRM_VIDEO_LEVEL_MIN
0x54	CORACQ_PRM_VIDEO_LEVEL_MAX
0x55	CORACQ_PRM_CONNECTOR_LINE_TRIGGER_INPUT
0x56	CORACQ_PRM_CONNECTOR_LINE_INTEGRATE_INPUT

0x57	CORACQ_PRM_CONNECTOR_LINESCAN_DIRECTION_INPUT
0x58	CORACQ_PRM_CAMLINK_CONFIGURATION
0x59-0x5e	Reserved
0x5f	CORACQ_PRM_DATA_VALID_ENABLE
0x60	CORACQ_PRM_DATA_VALID_POLARITY
0x61	CORACQ_PRM_CONNECTOR_PIXEL_CLK_OUTPUT
0x62	CORACQ_PRM_CONNECTOR_WEN_OUTPUT
0x63	CORACQ_PRM_WEN_POLARITY
0x64-0x6b	Reserved
0x6c	CORACQ_PRM_TIMESLOT
0x6d	
0x6e	CORACQ_PRM_CAM_CONTROL_DURING_READOUT
0x6f-0x70	Reserved
0x71	CORACQ_PRM_CX4_CONFIGURATION
0x72-0x73	Reserved
0x74	CORACQ_PRM_DATA_LANES
0x75	Reserved
0x76	CORACQ_PRM_CLHS_BIT_TRANSFER_RATE

#### **CORACQ\_PRM\_CLHS\_BIT\_TRANSFER\_RATE**

<b>Description</b>	Bit transfer rate between camera and acquisition device.
<b>Type</b>	UINT32
<b>Limits</b>	<p>This value must match one of the supported capabilities of the acquisition device given by CORACQ_CAP_BIT_TRANSFER_RATE_MULT and CORACQ_CAP_BIT_TRANSFER_RATE. The capability returns the ORed combination of all supported values.</p> <p>CORACQ_CAP_BIT_TRANSFER_RATE_MULT returns the basic bit rate in Mbps. The CORACQ_CAP_BIT_TRANSFER_RATE is a bitfield where if bit 'x' is 1, then the <math>(x + 1) * \text{Mbps}</math> transfer rate is supported.</p> <p>Example: CORACQ_CAP_BIT_TRANSFER_RATE_MULT = 1250, CORACQ_CAP_BIT_TRANSFER_RATE = 0x00000010 Then acquisition device supports only <math>(\text{Bit } 4 + 1) * 125 = 6250 \text{ Mbps}</math> or 6.25 Gbps.</p>
<b>CVI Entry</b>	[Signal Description] Bit Transfer Rate
<b>Note</b>	Validated only for CLHS connector type.

#### **CORACQ\_PRM\_CAM\_COMPANY\_NAME**

<b>Description</b>	The camera company name for which the camera file is intended for.
<b>Type</b>	BYTE [32]
<b>Values</b>	String up to 31 characters long.
<b>CCA Entry</b>	[General] Camera Name

### **CORACQ\_PRM\_CAM\_CONTROL\_DURING\_READOUT**

<b>Description</b>	Specifies if the camera control signals can be sent during the readout of a frame.
<b>Type</b>	UINT32
<b>Values</b>	<p>CORACQ_VAL_CAM_CONTROL_DURING_READOUT_INVALID (0x00000000)</p> <p>Camera controls will not be sent during the readout of a frame. Once a camera is triggered, the next trigger will not occur until the end of FVAL is reached.</p> <p>CORACQ_VAL_CAM_CONTROL_DURING_READOUT_VALID (0x00000001)</p> <p>Camera controls can be sent during the readout of a frame. Once a camera is triggered, the next trigger can be sent when the frame grabber receives the corresponding FVAL.</p> <p>CORACQ_VAL_CAM_CONTROL_DURING_READOUT_IGNORE (0x00000002)</p> <p>Camera controls are sent whenever an external/internal frame trigger is received. It is the responsibility of the user to not over trigger the camera.</p>
<b>Limits</b>	This value must match one of the supported capabilities of the acquisition device given by CORACQ_CAP_CAM_CONTROL_DURING_READOUT. The capability returns the ORed combination of all supported values. Note that CORACQ_VAL_CAM_CONTROL_DURING_READOUT_INVALID is always supported.
<b>CCA Entry</b>	[Control Signals] Camera Control During Readout
<b>Note</b>	Valid only for Area Scan cameras.

### **CORACQ\_PRM\_CAM\_IO\_CONTROL**

<b>Description</b>	Description of the non-standard camera I/O controls.
<b>Type</b>	CORACQ_CAM_IO_CONTROL[32]
<b>Values</b>	List of the non-standard camera I/O controls.
<b>CCA Entry</b>	[Custom Camera IO Control Signals] Max Control Control_x (x takes a value from 0 to 31)
<b>Note</b>	See "Custom Camera I/O Control Description" for more information.

### **CORACQ\_PRM\_CAM\_LINE\_TRIGGER\_FREQ\_MAX**

<b>Description</b>	Maximum line trigger frequency supported by the camera (in Hz).
<b>Type</b>	UINT32
<b>Limits</b>	This value must be greater or equal to CORACQ_PRM_CAM_LINE_TRIGGER_FREQ_MIN
<b>CCA Entry</b>	[Control Signals] Camera Line Trigger Frequency Maximum
<b>Note</b>	Applies to linescan cameras only.

### **CORACQ\_PRM\_CAM\_LINE\_TRIGGER\_FREQ\_MIN**

<b>Description</b>	Minimum line trigger frequency supported by the camera (in Hz).
<b>Type</b>	UINT32
<b>Limits</b>	This value must be smaller or equal to CORACQ_PRM_CAM_LINE_TRIGGER_FREQ_MAX
<b>CCA Entry</b>	[Control Signals] Camera Line Trigger Frequency Minimum
<b>Note</b>	Applies to linescan cameras only.

### **CORACQ\_PRM\_CAM\_MODEL\_NAME**

<b>Description</b>	The camera model name or which the camera file is intended for.
<b>Type</b>	BYTE [32]
<b>Values</b>	String up to 31 characters long.
<b>CCA Entry</b>	[General] Model Name

### **CORACQ\_PRM\_CAM\_NAME**

<b>Description</b>	The name or description of the camera related parameters.
<b>Type</b>	BYTE [64]
<b>Values</b>	String, up to 63 characters long.
<b>CCA Entry</b>	[General] Camera Name

### **CORACQ\_PRM\_CAM\_RESET\_DURATION**

<b>Description</b>	Reset pulse width (in $\mu$ s). Applies to area scan cameras only.
<b>Type</b>	UINT32
<b>Limits</b>	The value must be in the range CORACQ_CAP_CAM_RESET_DURATION_MIN ... CORACQ_CAP_CAM_RESET_DURATION_MAX.
<b>CCA Entry</b>	[Control Signals] Camera Reset Duration
<b>Note</b>	Validated only when CORACQ_PRM_CAM_RESET_ENABLE is TRUE.

### **CORACQ\_PRM\_CAM\_RESET\_METHOD**

<b>Description</b>	Method used to generate the reset pulse. Applies to area scan cameras only.
<b>Type</b>	UINT32
<b>Limits</b>	This value must match one of the supported capabilities of the acquisition device given by CORACQ_CAP_CAM_RESET_METHOD. The capability returns the ORed combination of all supported values.
<b>Values</b>	See Camera Reset Method
<b>CCA Entry</b>	[Control Signals] Camera Reset Method
<b>Note</b>	Available only if CORACQ_CAP_CAM_RESET is TRUE. Validated only when CORACQ_PRM_CAM_RESET_ENABLE is TRUE.

### **CORACQ\_PRM\_CAM\_RESET\_POLARITY**

<b>Description</b>	Reset pulse polarity. Applies to area scan cameras only.
<b>Type</b>	UINT32
<b>Limits</b>	This value must match one of the supported capabilities of the acquisition device given by CORACQ_CAP_CAM_RESET_POLARITY. The capability returns the ORed combination of all supported values.
<b>Values</b>	CORACQ_VAL_ACTIVE_LOW (0x00000001)      Reset pulse will be active low. CORACQ_VAL_ACTIVE_HIGH (0x00000002)      Reset pulse will be active high.
<b>CCA Entry</b>	[Control Signals] Camera Reset Polarity
<b>Note</b>	Available only if CORACQ_CAP_CAM_RESET is TRUE. Validated only when CORACQ_PRM_CAM_RESET_ENABLE is TRUE.



### **CORACQ\_PRM\_CAM\_TIME\_INTEGRATE\_DURATION\_MAX**

<b>Description</b>	Maximum time integration supported by the camera (in $\mu$ s). Applies to area scan cameras only.
<b>Type</b>	UINT32
<b>Limits</b>	This value must be greater or equal to CORACQ_PRM_CAM_TIME_INTEGRATE_DURATION_MIN.
<b>CCA Entry</b>	[Control Signals] Camera Time Integrate Duration Maximum

### **CORACQ\_PRM\_CAM\_TIME\_INTEGRATE\_DURATION\_MIN**

<b>Description</b>	Minimum time integration supported by the camera (in $\mu$ s). Applies to area scan cameras only.
<b>Type</b>	UINT32
<b>Limits</b>	This value must be smaller or equal to CORACQ_PRM_CAM_TIME_INTEGRATE_DURATION_MAX.
<b>CCA Entry</b>	[Control Signals] Camera Time Integrate Duration Minimum

### **CORACQ\_PRM\_CAM\_TRIGGER\_DURATION**

<b>Description</b>	Frame trigger pulse width (in $\mu$ s). Applies to area scan cameras only.
<b>Type</b>	UINT32
<b>Limits</b>	The value must be in the range CORACQ_CAP_CAM_TRIGGER_DURATION_MIN ... CORACQ_CAP_CAM_TRIGGER_DURATION_MAX.
<b>CCA Entry</b>	[Control Signals] Camera Trigger Duration
<b>Note</b>	Available only if CORACQ_CAP_CAM_TRIGGER is TRUE. Validated only when CORACQ_PRM_CAM_TRIGGER_ENABLE is TRUE.

### **CORACQ\_PRM\_CAM\_TRIGGER\_METHOD**

<b>Description</b>	Frame trigger pulse output method. Applies to area scan cameras only.
<b>Type</b>	UINT32
<b>Limits</b>	This value must match one of the supported capabilities of the acquisition device given by CORACQ_CAP_CAM_TRIGGER_METHOD. The capability returns the ORed combination of all supported values.
<b>Values</b>	See Camera Trigger Methods.
<b>CCA Entry</b>	[Control Signals] Camera Trigger Method
<b>Note</b>	Available only if CORACQ_CAP_CAM_TRIGGER is TRUE. Validated only when CORACQ_PRM_CAM_TRIGGER_ENABLE is TRUE.

## **CORACQ\_PRM\_CAM\_TRIGGER\_POLARITY**

<b>Description</b>	Frame trigger pulse polarity. Applies to area scan cameras only.	
<b>Type</b>	UINT32	
<b>Limits</b>	This value must match one of the supported capabilities of the acquisition device given by CORACQ_CAP_CAM_TRIGGER_POLARITY. The capability returns the ORed combination of all supported values.	
<b>Values</b>	CORACQ_VAL_ACTIVE_LOW (0x00000001)	Frame trigger pulse will be active low.
	CORACQ_VAL_ACTIVE_HIGH (0x00000002)	Frame trigger pulse will be active high.
<b>CCA Entry</b>	[Control Signals] Camera Trigger Polarity	
<b>Note</b>	Available only if CORACQ_CAP_CAM_TRIGGER is TRUE. Validated only when CORACQ_PRM_CAM_TRIGGER_ENABLE is TRUE.	

## **CORACQ\_PRM\_CAMLINK\_CONFIGURATION**

<b>Description</b>	Defines the CameraLink connector configuration	
<b>Type</b>	UINT32	
<b>Limits</b>	This value must match one of the supported capabilities of the acquisition device given by CORACQ_CAP_CAMLINK_CONFIGURATION. The capability returns the ORed combination of all supported values.	
<b>Values</b>	CORACQ_VAL_CAMLINK_CONFIGURATION_BASE (0x00000001)	Base configuration (1 connector)
	CORACQ_VAL_CAMLINK_CONFIGURATION_MEDIUM (0x00000002)	Medium configuration (2 connectors)
	CORACQ_VAL_CAMLINK_CONFIGURATION_FULL (0x00000004)	Full configuration (2 connectors)
	CORACQ_VAL_CAMLINK_CONFIGURATION_2BASE (0x00000008)	Dual base configuration (2 connectors)
	CORACQ_VAL_CAMLINK_CONFIGURATION_10TAPS_FORMAT1 (0x00000010)	10 Taps (2 connectors) for example, CMC-1000
	CORACQ_VAL_CAMLINK_CONFIGURATION_16TAPS (0x00000020)	16 Taps (4 connectors)
	CORACQ_VAL_CAMLINK_CONFIGURATION_10TAPS_FORMAT2 (0x00000040)	10 Taps (2 connectors) for example, Basler A504
	CORACQ_VAL_CAMLINK_CONFIGURATION_8TAPS_10BITS (0x00000080)	8 taps @ 10 bits (2 connectors) for example, Basler A403
	CORACQ_VAL_CAMLINK_CONFIGURATION_FULL_PACKED (0x00000100)	The video data is packed on the 8 ports of the Camera Link cable
	CORACQ_VAL_CAMLINK_CONFIGURATION_80BITS_PACKED (0x00000200)	The video data is packed on the 10 ports of the Camera Link cable.
	CORACQ_VAL_CAMLINK_CONFIGURATION_FLAG_BGR (0x80000000)	By default, RGB formats are received in the RGB order on the CameraLink ports. By using this value, the order will be considered as BGR
<b>CCA Entry</b>	[Connector Description] Camlink Configuration	

## **CORACQ\_PRM\_CHANNEL**

<b>Description</b>	Number of channels output by the video source. Applies to area scan cameras only.	
<b>Type</b>	UINT32	
<b>Limits</b>	This value must match one of the supported capabilities of the acquisition device given by CORACQ_CAP_CHANNEL.	
<b>Values</b>	CORACQ_VAL_CHANNEL_SINGLE (0x00000001)	One video channel is fed to the acquisition device.
	CORACQ_VAL_CHANNEL_DUAL (0x00000002)	Two synchronous video channels are fed to the acquisition device.
<b>CCA Entry</b>	[Signal Description] Channel	
<b>Note</b>	For a description of tap geometries and corresponding parameter settings see Appendix: Tap Geometry Settings.	

## **CORACQ\_PRM\_CHANNELS\_ORDER**

<b>Description</b>	Order of the channels. Applies to area scan cameras only.	
<b>Type</b>	UINT32	
<b>Limits</b>	This value must match one of the supported capabilities of the acquisition device given by CORACQ_CAP_CHANNELS_ORDER. The capability returns the ORed combination of all supported values.	
<b>Values</b>	CORACQ_VAL_CHANNELS_ORDER_NORMAL (0x00000001) The camera outputs the first line of the video on channel 1 (or A), the second line on channel 2 (or B), ...	
	CORACQ_VAL_CHANNELS_ORDER_REVERSE (0x00000002) The camera outputs the first line of the video on channel 2 (or B), the second line on channel 1 (or A), ...	
	CORACQ_VAL_CHANNELS_ORDER_DETECT (0x00000004) Auto detects the channel order by means of an external signal usually called FI (field index). If the signal is high, then the channel order is considered normal; otherwise it is reversed.	
	CORACQ_VAL_CHANNELS_ORDER_SEGMENTED (0x00000008) Use when the number of channels is greater than 2. The camera outputs: the video lines 0 to n-1 on the first channel the video lines n to 2n-1 on the second channel the video lines 2n to 3n-1 on the third channel ... the video lines 7n to 7n-1 on the 8th channel	
<b>CCA Entry</b>	[Signal Description] Channels Order	
<b>Note</b>	For a description of tap geometries and corresponding parameter settings see Appendix: Tap Geometry Settings.	

## **CORACQ\_PRM\_COLOR\_ALIGNMENT**

<b>Description</b>	Specifies the Bayer or Bicolor alignment of the image output by the video source.
<b>Type</b>	UINT32
<b>Limits</b>	The parameter value must match one of the supported alignments of the acquisition device given by CORACQ_CAP_COLOR_ALIGNMENT. The capability returns the ORed combination of all supported values as defined below.
<b>Values</b>	CORACQ_VAL_COLOR_ALIGNMENT_GB_RG (0x00000001) CORACQ_VAL_COLOR_ALIGNMENT_BG_GR (0x00000002) CORACQ_VAL_COLOR_ALIGNMENT_RG_GB (0x00000004) CORACQ_VAL_COLOR_ALIGNMENT_GR_BG (0x00000008) CORACQ_VAL_COLOR_ALIGNMENT_RG_BG (0x00000010) CORACQ_VAL_COLOR_ALIGNMENT_BG_RG (0x00000020)
<b>CCA Entry</b>	[Signal Description] Bayer Alignment
<b>Note</b>	Validated only if CORACQ_PRM_COLOR_DECODER_ENABLE is TRUE.

## **CORACQ\_PRM\_CONNECTOR\_EXPOSURE\_INPUT**

<b>Description</b>	Camera exposure input pin description.
<b>Type</b>	UINT32
<b>Values</b>	See Pin Connector Description
<b>CCA Entry</b>	[Connector Description] Exposure Input

## **CORACQ\_PRM\_CONNECTOR\_HD\_INPUT**

<b>Description</b>	Camera horizontal drive input/output pin description.
<b>Type</b>	UINT32
<b>Values</b>	See Pin Connector Description
<b>CCA Entry</b>	[Connector Description] HD Input

## **CORACQ\_PRM\_CONNECTOR\_LINE\_INTEGRATE\_INPUT**

<b>Description</b>	Camera line integrate pin description. Applies to linescan cameras only.
<b>Type</b>	UINT32
<b>Values</b>	See Pin Connector Description
<b>CCA Entry</b>	[Connector Description] Line Integrate Input
<b>Note</b>	Some cameras define this input as PRIN.

## **CORACQ\_PRM\_CONNECTOR\_LINE\_TRIGGER\_INPUT**

<b>Description</b>	Camera line trigger/exposure pin description. Applies to linescan cameras only.
<b>Type</b>	UINT32
<b>Values</b>	See Pin Connector Description
<b>CCA Entry</b>	[Connector Description] Line Trigger Input
<b>Note</b>	Some cameras define this input as EXSYNC.

#### **CORACQ\_PRM\_CONNECTOR\_LINESCAN\_DIRECTION\_INPUT**

<b>Description</b>	Camera linescan direction pin description. Applies to linescan cameras only.
<b>Type</b>	UINT32
<b>Values</b>	See Pin Connector Description
<b>CCA Entry</b>	[Connector Description] Linescan Direction Input

#### **CORACQ\_PRM\_CONNECTOR\_PIXEL\_CLK\_OUTPUT**

<b>Description</b>	Camera pixel clock output pin description.
<b>Type</b>	UINT32
<b>Values</b>	See Pin Connector Description
<b>CCA Entry</b>	[Connector Description] Pixel Clock Output

#### **CORACQ\_PRM\_CONNECTOR\_WEN\_OUTPUT**

<b>Description</b>	Camera WEN (Write ENable) output pin description.
<b>Type</b>	UINT32
<b>Values</b>	See Pin Connector Description
<b>CCA Entry</b>	[Connector Description] WEN Output

#### **CORACQ\_PRM\_CONNECTOR\_RESET\_TRIGGER\_INPUT**

<b>Description</b>	Camera Reset/Trigger input pin description.
<b>Type</b>	UINT32
<b>Values</b>	See Pin Connector Description
<b>CCA Entry</b>	[Connector Description] Reset/Trigger Input

#### **CORACQ\_PRM\_CONNECTOR\_VD\_INPUT**

<b>Description</b>	Camera vertical drive input/output pin description.
<b>Type</b>	UINT32
<b>Values</b>	See Pin Connector Description
<b>CCA Entry</b>	[Connector Description] VD Input

## CORACQ\_PRM\_CX4\_CONFIGURATION

<b>Description</b>	Defines the number of lanes the video will be transferred on and if the communication lane is dedicated or overlaps a data lane. Applies to HS-Link devices only.
<b>Type</b>	UINT32
<b>Limits</b>	Bits 0..7: Number of lanes used by the video data. Must be in the range 1..(CORACQ_CAP_CX4_CONFIGURATION & 0xff). Macro CORACQ_VAL_CX4_CONFIGURATION_LANES_MASK can be used to get the number of lanes out of this parameter.  Bit 31: 1 indicates that the communication lane overlaps the 1 <sup>st</sup> video lane. Can only be set to 1 if CORACQ_CAP_CX4_CONFIGURATION & 0x80000000 is true. Macro CORACQ_VAL_CX4_CONFIGURATION_COMM_OVERLAP can be used to get/set the communication lane overlap state of this parameter.
<b>CCA Entry</b>	[Signal Description] CX4 Configuration

## CORACQ\_PRM\_COUPLING

<b>Description</b>	Video source coupling type. Applies to analog video signals only.
<b>Type</b>	UINT32
<b>Limits</b>	This value must match one of the supported capabilities of the acquisition device given by CORACQ_CAP_COUPLING. The capability returns the ORed combination of all supported values.
<b>Values</b>	CORACQ_VAL_COUPLING_AC (0x00000001)      AC coupled. CORACQ_VAL_COUPLING_DC (0x00000002)      DC coupled.
<b>CCA Entry</b>	[Signal Description] Coupling

## CORACQ\_PRM\_DATA\_LANES

<b>Description</b>	Number of Data Lanes output by the Camera.
<b>Limits</b>	Range Limits: 1..CORACQ_CAP_CLHS_LANES_MAX.
<b>CCA Entry</b>	[Signal Description] Data Lanes
<b>Note</b>	Valid only for CLHS connector type.

## CORACQ\_PRM\_DATA\_VALID\_ENABLE

<b>Description</b>	Specifies if the acquisition device uses the camera data valid signal.
<b>Type</b>	UINT32
<b>Limits</b>	This value must match the capability of the acquisition device given by CORACQ_CAP_DATA_VALID_ENABLE = TRUE.
<b>Values</b>	FALSE (0x00000000)      Data valid signal is ignored. TRUE (0x00000001)      Data valid signal is used.
<b>CCA Entry</b>	[Control Signals] Data Valid Enable

## CORACQ\_PRM\_DATA\_VALID\_POLARITY

<b>Description</b>	Specifies the camera data valid polarity received from the acquisition device.	
<b>Type</b>	UINT32	
<b>Limits</b>	This value must match one of the supported capabilities of the acquisition device given by CORACQ_CAP_DATA_VALID_POLARITY. The capability returns the ORed combination of all supported values.	
<b>Values</b>	CORACQ_VAL_ACTIVE_LOW (0x00000001)	Data valid signal active low.
	CORACQ_VAL_ACTIVE_HIGH (0x00000002)	Data valid signal active high.
<b>CCA Entry</b>	[Control Signals] Data Valid Polarity	
<b>Note</b>	Validated only if CORACQ_DATA_VALID_ENABLE is TRUE	

## CORACQ\_PRM\_FIELD\_ORDER

<b>Description</b>	Field order output by the video source. Applies to area scan cameras only.	
<b>Type</b>	UINT32	
<b>Limits</b>	This value must match one of the supported capabilities of the acquisition device given by CORACQ_CAP_FIELD_ORDER. The capability returns the ORed combination of all supported values.	
<b>Values</b>	CORACQ_VAL_FIELD_ORDER_ODD_EVEN (0x00000001) For an interlaced signal, the odd field is acquired first, followed by the even field. For a non-interlaced signal, this value is invalid.	
	CORACQ_VAL_FIELD_ORDER_EVEN_ODD (0x00000002) For an interlaced signal, the even field is acquired first, followed by the odd field. For a non-interlaced signal, this value is invalid.	
	CORACQ_VAL_FIELD_ORDER_NEXT_FIELD (0x00000004) For an interlaced signal, the next field is acquired whether it is odd or even. This is the standard value for a non-interlaced signal.	
	CORACQ_VAL_FIELD_ORDER_FVAL_LINE1 (0x00000008) For a linescan camera, the FVAL is active to indicate a grouping of lines. In the case of a Bayer video source, the FVAL will group 2 lines together. When converting to RGB data, the 1 <sup>st</sup> line will be considered as the 1 <sup>st</sup> one, and the 2 <sup>nd</sup> one as the 2 <sup>nd</sup> .	
	CORACQ_VAL_FIELD_ORDER_FVAL_LINE2 (0x00000010) For a linescan camera, the FVAL is active to indicate a grouping of lines. In the case of a Bayer video source, the FVAL will group 2 lines together. When converting to RGB data, the 2 <sup>nd</sup> line will be considered as the 1 <sup>st</sup> one, and the 2 <sup>nd</sup> one as the 1 <sup>st</sup> .	
<b>CCA Entry</b>	[Signal Description] Field Order	



## CORACQ\_PRM\_FRAME

<b>Description</b>	Video source frame type. Applies to area scan cameras only.	
<b>Type</b>	UINT32	
<b>Limits</b>	This value must match one of the supported capabilities of the acquisition device given by CORACQ_CAP_FRAME. The capability returns the ORed combination of all supported values.	
<b>Values</b>	CORACQ_VAL_FRAME_INTERLACED (0x00000001)	Interlaced video.
	CORACQ_VAL_FRAME_PROGRESSIVE (0x00000002)	Progressive/non-interlaced video.
<b>CCA Entry</b>	[Signal Description] Frame	

## CORACQ\_PRM\_FRAME\_INTEGRATE\_METHOD

<b>Description</b>	Method to be used to control the camera's frame integration. Applies to area scan cameras only.	
<b>Type</b>	UINT32	
<b>Limits</b>	This value must match one of the supported capabilities of the acquisition device given by CORACQ_CAP_FRAME_INTEGRATE_METHOD. The capability returns the ORed combination of all supported values.	
<b>Values</b>	See Frame Integrate Methods	
<b>CCA Entry</b>	[Control Signals] Frame Integrate Method	
<b>Note</b>	Available only if CORACQ_CAP_FRAME_INTEGRATE is TRUE. Validated only when CORACQ_PRM_FRAME_INTEGRATE_ENABLE is TRUE.	

## CORACQ\_PRM\_FRAME\_INTEGRATE\_POLARITY

<b>Description</b>	Frame integration pulse polarity. Applies to area scan cameras only.	
<b>Type</b>	UINT32	
<b>Limits</b>	This value must match one of the supported capabilities of the acquisition device given by CORACQ_CAP_FRAME_INTEGRATE_POLARITY. The capability returns the ORed combination of all supported values.	
<b>Values</b>	CORACQ_VAL_ACTIVE_LOW (0x00000001)	Frame integration pulse will be active low.
	CORACQ_VAL_ACTIVE_HIGH (0x00000002)	Frame integration pulse will be active high.
<b>CCA Entry</b>	[Control Signals] Frame Integrate Polarity	
<b>Note</b>	Available only if CORACQ_CAP_FRAME_INTEGRATE is TRUE. Validated only when CORACQ_PRM_FRAME_INTEGRATE_ENABLE is TRUE.	

## CORACQ\_PRM\_HACTIVE

<b>Description</b>	Horizontal active portion of the video (in pixels/tap).	
<b>Type</b>	UINT32	
<b>Limits</b>	Range limits: CORACQ_CAP_HACTIVE_MIN...CORACQ_CAP_HACTIVE_MAX, and also must be a multiple of CORACQ_CAP_HACTIVE_MULT.	
<b>CCA Entry</b>	[Signal Timings] Horizontal Active	

## CORACQ\_PRM\_HBACK\_INVALID

<b>Description</b>	Invalid horizontal portion of the video following the horizontal blanking (in pixels/tap).
<b>Type</b>	UINT32
<b>Limits</b>	Range limits: CORACQ_CAP_HBACK_INVALID_MIN ... CORACQ_CAP_HBACK_INVALID_MAX, and also must be a multiple of CORACQ_CAP_HBACK_INVALID_MULT.
<b>CCA Entry</b>	[Signal Timings] Horizontal Back Invalid

## CORACQ\_PRM\_HBACK\_PORCH

<b>Description</b>	The video's horizontal back porch (in pixels/tap). Applies to analog video signals only.
<b>Type</b>	UINT32
<b>Limits</b>	Range limits: CORACQ_CAP_HBACK_PORCH_MIN ... CORACQ_CAP_HBACK_PORCH_MAX, and must be a multiple of CORACQ_CAP_HBACK_PORCH_MULT.
<b>CCA Entry</b>	[Signal Timings] Horizontal Back Porch

## CORACQ\_PRM\_HFRONT\_INVALID

<b>Description</b>	Invalid horizontal portion of the video preceding the horizontal blanking (in pixels/tap).
<b>Type</b>	UINT32
<b>Limits</b>	This value must be in the range CORACQ_CAP_HFRONT_INVALID_MIN...CORACQ_CAP_HFRONT_INVALID_MAX, and must be a multiple of CORACQ_CAP_HFRONT_INVALID_MULT.
<b>CCA Entry</b>	[Signal Timings] Horizontal Front Invalid

## CORACQ\_PRM\_HFRONT\_PORCH

<b>Description</b>	The video's horizontal front porch (in pixels/tap). Applies to analog video signals only.
<b>Type</b>	UINT32
<b>Limits</b>	This value must be in the range CORACQ_CAP_HFRONT_PORCH_MIN...CORACQ_CAP_HFRONT_PORCH_MAX, and must be a multiple of CORACQ_CAP_HFRONT_PORCH_MULT.
<b>CCA Entry</b>	[Signal Timings] Horizontal Front Porch

## CORACQ\_PRM\_HSYNC

<b>Description</b>	The videos horizontal sync (in pixels/tap).
<b>Type</b>	UINT32
<b>Limits</b>	Range limits: CORACQ_CAP_HSYNC_MIN...CORACQ_CAP_HSYNC_MAX, and also must be a multiple of CORACQ_CAP_HSYNC_MULT.
<b>CCA Entry</b>	[Signal Timings] Horizontal Sync

## CORACQ\_PRM\_HSYNC\_POLARITY

<b>Description</b>	Horizontal sync polarity of the video source.	
<b>Type</b>	UINT32	
<b>Limits</b>	This value must match one of the supported capabilities of the acquisition device given by CORACQ_CAP_HSYNC_POLARITY. The capability returns the ORed combination of all supported values.	
<b>Values</b>	CORACQ_VAL_ACTIVE_LOW (0x00000001)	Horizontal sync pulse is active low.
	CORACQ_VAL_ACTIVE_HIGH (0x00000002)	Horizontal sync pulse is active high.
<b>CCA Entry</b>	[Synchronization Signals] Horizontal Sync Polarity	

## CORACQ\_PRM\_INTERFACE

<b>Description</b>	Video source interface type.	
<b>Type</b>	UINT32	
<b>Limits</b>	This value must match one of the supported capabilities of the acquisition device given by CORACQ_CAP_INTERFACE since only one interface type is supported per acquisition device.	
<b>Values</b>	CORACQ_VAL_INTERFACE_ANALOG (0x00000001)	Analog video source.
	CORACQ_VAL_INTERFACE_DIGITAL (0x00000002)	Digital video source.
<b>CCA Entry</b>	[Signal Description] Interface	

## CORACQ\_PRM\_LINE\_INTEGRATE\_DELAY

<b>Description</b>	Obsolete. Use instead the equivalent parameter CORACQ_PRM_LINE_INTEGRATE_PULSE0_DELAY
--------------------	------------------------------------------------------------------------------------------

## CORACQ\_PRM\_LINE\_INTEGRATE\_METHOD

<b>Description</b>	Method to use for controlling the camera's line integration. Applies to linescan cameras only.
<b>Type</b>	UINT32
<b>Limits</b>	This value must match one of the supported capabilities of the acquisition device given by CORACQ_CAP_LINE_INTEGRATE_METHOD. The capability returns the ORed combination of all supported values.
<b>Values</b>	See Line Integrate Methods
<b>CCA Entry</b>	[Control Signals] Line Integrate Method
<b>Note</b>	Available only if CORACQ_CAP_LINE_INTEGRATE is TRUE. Validated only when CORACQ_PRM_LINE_INTEGRATE_ENABLE is TRUE.

## CORACQ\_PRM\_LINE\_INTEGRATE\_POLARITY

<b>Description</b>	Obsolete. Use instead the equivalent parameter CORACQ_PRM_LINE_INTEGRATE_PULSE0_POLARITY
--------------------	---------------------------------------------------------------------------------------------

### CORACQ\_PRM\_LINE\_INTEGRATE\_PULSE0\_DELAY

<b>Description</b>	Line integration pulse #0 delay in units specified by CORACQ_PRM_LINE_INTEGRATE_TIME_BASE. In the case where the units are in CORACQ_VAL_TIME_BASE_NS, the CORACQ_CAP_LINE_INTEGRATE_TIME_BASE_MULT returns the resolution of the time base in nsec. Applies to linescan cameras only.
<b>Type</b>	UINT32
<b>Limits</b>	Range limits: CORACQ_CAP_LINE_INTEGRATE_PULSE0_DELAY_MIN ... CORACQ_CAP_LINE_INTEGRATE_PULSE0_DELAY_MAX.
<b>CCA Entry</b>	[Control Signals] Line Integrate Pulse 0 Delay
<b>Note</b>	Available only if CORACQ_CAP_LINE_INTEGRATE is TRUE. Validated only when CORACQ_PRM_LINE_INTEGRATE_ENABLE is TRUE. See Line Integrate Methods for the different usages of the pulse #0 delay parameter.

### CORACQ\_PRM\_LINE\_INTEGRATE\_PULSE0\_DURATION

<b>Description</b>	Line integration pulse #0 width in units specified by CORACQ_PRM_LINE_INTEGRATE_TIME_BASE. In the case where the units are in CORACQ_VAL_TIME_BASE_NS, the CORACQ_CAP_LINE_INTEGRATE_TIME_BASE_MULT returns the resolution of the time base in nsec. Applies to linescan cameras only.
<b>Type</b>	UINT32
<b>Limits</b>	Range limits: CORACQ_CAP_LINE_INTEGRATE_PULSE0_DURATION_MIN ... CORACQ_CAP_LINE_INTEGRATE_PULSE0_DURATION_MAX.
<b>CCA Entry</b>	[Control Signals] Line Integrate Pulse 0 Duration
<b>Note</b>	Available only if CORACQ_CAP_LINE_INTEGRATE is TRUE. Validated only when CORACQ_PRM_LINE_INTEGRATE_ENABLE is TRUE. See Line Integrate Methods for the different usages of the pulse #0 duration parameter.

### CORACQ\_PRM\_LINE\_INTEGRATE\_PULSE0\_POLARITY

<b>Description</b>	Line integration pulse #0 polarity. Applies to linescan cameras only.
<b>Type</b>	UINT32
<b>Limits</b>	This value must match one of the supported capabilities of the acquisition device given by CORACQ_CAP_LINE_INTEGRATE_PULSE0_POLARITY. The capability returns the ORed combination of all supported values.
<b>Values</b>	CORACQ_VAL_ACTIVE_LOW (0x00000001) Time integration trigger pulse is active low. CORACQ_VAL_ACTIVE_HIGH (0x00000002) Time integration trigger pulse is active high.
<b>CCA Entry</b>	[Control Signals] Line Integrate Pulse 0 Polarity
<b>Note</b>	Available only if CORACQ_CAP_LINE_INTEGRATE is TRUE. Validated only when CORACQ_PRM_LINE_INTEGRATE_ENABLE is TRUE. See Line Integrate Methods for the different usages of the pulse #0. Note, if a constant signal is required, set this parameter to have an active signal polarity opposite to that of the constant signal. For example, to have a constant high signal the polarity would be set to CORACQ_VAL_ACTIVE_LOW.

### CORACQ\_PRM\_LINE\_INTEGRATE\_PULSE1\_DELAY

<b>Description</b>	Line integration pulse #1 delay in units specified by CORACQ_PRM_LINE_INTEGRATE_TIME_BASE. In the case where the units are in CORACQ_VAL_TIME_BASE_NS, the CORACQ_CAP_LINE_INTEGRATE_TIME_BASE_MULT returns the resolution of the time base in nsec. Applies to linescan cameras only.
<b>Type</b>	UINT32
<b>Limits</b>	Range limits: CORACQ_CAP_LINE_INTEGRATE_PULSE1_DELAY_MIN ... CORACQ_CAP_LINE_INTEGRATE_PULSE1_DELAY_MAX.
<b>CCA Entry</b>	[Control Signals] Line Integrate Pulse 1 Delay
<b>Note</b>	Available only if CORACQ_CAP_LINE_INTEGRATE is TRUE. Validated only when CORACQ_PRM_LINE_INTEGRATE_ENABLE is TRUE. See Line Integrate Methods for the different usages of the pulse #1 delay parameter.

### CORACQ\_PRM\_LINE\_INTEGRATE\_PULSE1\_DURATION

<b>Description</b>	Line integration pulse #1 width in units specified by CORACQ_PRM_LINE_INTEGRATE_TIME_BASE. In the case where the units are in CORACQ_VAL_TIME_BASE_NS, the CORACQ_CAP_LINE_INTEGRATE_TIME_BASE_MULT returns the resolution of the time base in nsec. Applies to linescan cameras only.
<b>Type</b>	UINT32
<b>Limits</b>	Range limits: CORACQ_CAP_LINE_INTEGRATE_PULSE1_DURATION_MIN ... CORACQ_CAP_LINE_INTEGRATE_PULSE1_DURATION_MAX.
<b>CCA Entry</b>	[Control Signals] Line Integrate Pulse 1 Duration
<b>Note</b>	Available only if CORACQ_CAP_LINE_INTEGRATE is TRUE. Validated only when CORACQ_PRM_LINE_INTEGRATE_ENABLE is TRUE. See Line Integrate Methods for the different usages of the pulse #1 duration parameter.

### CORACQ\_PRM\_LINE\_INTEGRATE\_PULSE1\_POLARITY

<b>Description</b>	Line integration pulse #1 polarity. Applies to linescan cameras only.	
<b>Type</b>	UINT32	
<b>Limits</b>	This value must match one of the supported capabilities of the acquisition device given by CORACQ_CAP_LINE_INTEGRATE_PULSE1_POLARITY. The capability returns the ORed combination of all supported values.	
<b>Values</b>	CORACQ_VAL_ACTIVE_LOW (0x00000001)	Line integration trigger pulse is active low.
	CORACQ_VAL_ACTIVE_HIGH (0x00000002)	Line integration trigger pulse is active high.
<b>CCA Entry</b>	[Control Signals] Line Integrate Pulse 1 Polarity	
<b>Note</b>	Available only if CORACQ_CAP_LINE_INTEGRATE is TRUE. Validated only when CORACQ_PRM_LINE_INTEGRATE_ENABLE is TRUE. See Line Integrate Methods for the different usages of the pulse #1. Note, if a constant signal is required, set this parameter to have an active signal polarity opposite to that of the constant signal. For example, to have a constant high signal the polarity would be set to CORACQ_VAL_ACTIVE_LOW.	

### **CORACQ\_PRM\_LINE\_TRIGGER\_DELAY**

<b>Description</b>	Line trigger pulse delay in units specified by CORACQ_PRM_LINE_INTEGRATE_TIME_BASE. In the case where the units are in CORACQ_VAL_TIME_BASE_NS, the CORACQ_CAP_LINE_INTEGRATE_TIME_BASE_MULT returns the resolution of the time base in nsec. Applies to linescan cameras only.
<b>Type</b>	UINT32
<b>Limits</b>	Range limits: CORACQ_CAP_LINE_TRIGGER_DELAY_MIN ... CORACQ_CAP_LINE_TRIGGER_DELAY_MAX.
<b>CCA Entry</b>	[Control Signals] Line Trigger Delay
<b>Note</b>	Available only if CORACQ_CAP_LINE_TRIGGER is TRUE. Validated only when CORACQ_PRM_LINE_TRIGGER_ENABLE is TRUE. See Line Trigger Methods for the different usages of the trigger delay parameter.

### **CORACQ\_PRM\_LINE\_TRIGGER\_DURATION**

<b>Description</b>	Line Trigger pulse width in units specified by CORACQ_PRM_LINE_INTEGRATE_TIME_BASE. In the case where the units are in CORACQ_VAL_TIME_BASE_NS, the CORACQ_CAP_LINE_INTEGRATE_TIME_BASE_MULT returns the resolution of the time base in nsec. Applies to linescan cameras only.
<b>Type</b>	UINT32
<b>Limits</b>	Range limits: CORACQ_CAP_LINE_TRIGGER_DURATION_MIN ... CORACQ_CAP_LINE_TRIGGER_DURATION_MAX.
<b>CCA Entry</b>	[Control Signals] Line Trigger Duration
<b>Note</b>	Available only if CORACQ_CAP_LINE_TRIGGER is TRUE. Validated only when CORACQ_PRM_LINE_TRIGGER_ENABLE is TRUE. See Line Trigger Methods for the different usages of the trigger duration parameter.

### **CORACQ\_PRM\_LINE\_TRIGGER\_METHOD**

<b>Description</b>	Line trigger pulse output method. Applies to linescan cameras only.
<b>Type</b>	UINT32
<b>Limits</b>	This value must match one of the supported capabilities of the acquisition device given by CORACQ_CAP_LINE_TRIGGER_METHOD. The capability returns the ORed combination of all supported values.
<b>Values</b>	See Line Trigger Methods
<b>CCA Entry</b>	[Control Signals] Line Trigger Method
<b>Note</b>	Available only if CORACQ_CAP_LINE_TRIGGER is TRUE. Validated only when CORACQ_PRM_LINE_TRIGGER_ENABLE is TRUE.

### CORACQ\_PRM\_LINE\_TRIGGER\_POLARITY

<b>Description</b>	Line trigger pulse polarity. Applies to linescan cameras only.	
<b>Type</b>	UINT32	
<b>Limits</b>	This value must match one of the supported capabilities of the acquisition device given by CORACQ_CAP_LINE_TRIGGER_POLARITY. The capability returns the ORed combination of all supported values.	
<b>Values</b>	CORACQ_VAL_ACTIVE_LOW (0x00000001)	Line trigger pulse is active low.
	CORACQ_VAL_ACTIVE_HIGH (0x00000002)	Line trigger pulse is active high.
<b>CCA Entry</b>	[Control Signals] Line Trigger Polarity	
<b>Note</b>	Available only if CORACQ_CAP_LINE_TRIGGER is TRUE. Validated only when CORACQ_PRM_LINE_TRIGGER_ENABLE is TRUE.	

### CORACQ\_PRM\_LINESCAN\_DIRECTION

<b>Description</b>	Specifies if the camera has a direction scan input control.	
<b>Type</b>	UINT32	
<b>Values</b>	TRUE (0x00000001),	Camera has a direction scan input control.
	FALSE (0x00000000),	Camera does not have a direction scan input control.
<b>CCA Entry</b>	[Control Signals] LineScan Direction	
<b>Note</b>	Applies to linescan cameras only. On Teledyne DALSA cameras, this control is called the TDI scan direction.	

### CORACQ\_PRM\_LINESCAN\_DIRECTION\_POLARITY

<b>Description</b>	Camera direction scan signal polarity. Applies to linescan cameras only.	
<b>Type</b>	UINT32	
<b>Limits</b>	This value must match one of the supported capabilities of the acquisition device given by CORACQ_CAP_LINESCAN_DIRECTION_POLARITY. The capability returns the ORed combination of all supported values.	
<b>Values</b>	CORACQ_VAL_ACTIVE_LOW (0x00000001)	Forward direction scan signal is active low.
	CORACQ_VAL_ACTIVE_HIGH (0x00000002)	Forward direction scan signal is active high.
<b>CCA Entry</b>	[Control Signals] LineScan Direction Polarity	
<b>Note</b>	This value is only available if CORACQ_CAP_LINESCAN_DIRECTION is TRUE.	

### CORACQ\_PRM\_PIXEL\_CLK\_11

<b>Description</b>	Pixel clock frequency (in Hz) so that the camera image has a 1:1 aspect ratio.	
<b>Type</b>	UINT32	
<b>Limits</b>	1.. (2**32) – 1	
<b>CCA Entry</b>	[Pixel Clock] Pixel Clock Frequency 1:1	
<b>Note</b>	This value is only given as information. Useful to accurately calculate distances between objects from an acquired image.	



## CORACQ\_PRM\_PIXEL\_CLK\_DETECTION

<b>Description</b>	Specifies the type of pixel clock detection of the video source.	
<b>Type</b>	UINT32	
<b>Limits</b>	This value must match one of the supported capabilities of the acquisition device given by CORACQ_CAP_PIXEL_CLK_DETECTION. The capability returns the ORed combination of all supported values.	
<b>Values</b>	CORACQ_VAL_RISING_EDGE (0x00000004)	Sampling of a pixel is done on the rising edge of the pixel clock.
	CORACQ_VAL_FALLING_EDGE (0x00000008)	Sampling of a pixel is done on the falling edge of the pixel clock.
<b>CCA Entry</b>	[Pixel Clock] Pixel Clock Detection	

## CORACQ\_PRM\_PIXEL\_CLK\_EXT

<b>Description</b>	External pixel clock frequency (in Hz).	
<b>Type</b>	UINT32	
<b>Limits</b>	The value must be in the range CORACQ_CAP_PIXEL_CLK_EXT_MIN...CORACQ_CAP_PIXEL_CLK_EXT_MAX.	
<b>CCA Entry</b>	[Pixel Clock] Pixel Clock Frequency External	
<b>Note</b>	Validated only if CORACQ_PRM_PIXEL_CLK_SRC specifies that an external pixel clock is needed.	

## CORACQ\_PRM\_PIXEL\_CLK\_INT

<b>Description</b>	Internal pixel clock frequency (in Hz).	
<b>Type</b>	UINT32	
<b>Limits</b>	The value must be in the range CORACQ_CAP_PIXEL_CLK_INT_MIN...CORACQ_CAP_PIXEL_CLK_INT_MAX.	
<b>CCA Entry</b>	[Pixel Clock] Pixel Clock Frequency Internal	
<b>Note</b>	This value is validated only if CORACQ_PRM_PIXEL_CLK_SRC specifies that an internal pixel clock is needed.	

## CORACQ\_PRM\_PIXEL\_CLK\_SRC

<b>Description</b>	Specifies the source of the acquisition device pixel clock.	
<b>Type</b>	UINT32	
<b>Limits</b>	This value must match one of the supported capabilities of the acquisition device given by CORACQ_CAP_PIXEL_CLK_SRC. The capability returns the ORed combination of all supported values.	
<b>Values</b>	CORACQ_VAL_PIXEL_CLK_SRC_INT (0x00000001)	Internal pixel clock.
	CORACQ_VAL_PIXEL_CLK_SRC_EXT (0x00000002)	External pixel clock.
	CORACQ_VAL_PIXEL_CLK_SRC_EXT_INT (0x00000004)	The external pixel clock is used while the acquisition device simultaneously outputs its own internal pixel clock for other use.
<b>CCA Entry</b>	[Pixel Clock] Pixel Clock Source	

## CORACQ\_PRM\_PIXEL\_DEPTH

**Description** Pixel depth of the digitized video.

**Type** UINT32

**Limits** This value must match one of the supported capabilities of the acquisition device given by CORACQ\_CAP\_PIXEL\_DEPTH.

This capability returns a structure of the following type:

typedef struct

```
{  
    UINT32 pixelDepth;  
    UINT32 numberOfLuts;  
    UINT32 lutFormat;  
} CAP_PIXEL_DEPTH;  
CAP_PIXEL_DEPTH capPixelFormat[43];
```

The amount of memory required for the capability is 512 bytes. Since there are 12 bytes per structure element, this means that you must allocate at least  $512/12=43$  (after rounding) such elements. The end of the list is reached when the pixelDepth value is 0.

pixelDepth: pixel depth in bits.

numberOfLuts: number of LUTs available

lutFormat: LUT format.

**CCA Entry** [Signal Description]  
Pixel Depth

**Note** For analog cameras, this parameter is read-only and represents the number of bits digitized by the acquisition device's A/D.

## CORACQ\_PRM\_SCAN

**Description** Video source scan type.

**Type** UINT32

**Limits** This value must match one of the supported capabilities of the acquisition device given by CORACQ\_CAP\_SCAN. The capability returns the ORed combination of all supported values.

<b>Values</b>	CORACQ_VAL_SCAN_AREA (0x00000001)	Area scan video source.
	CORACQ_VAL_SCAN_LINE (0x00000002)	Linescan video source.

**CCA Entry** [Signal Description]  
Scan

## CORACQ\_PRM\_SIGNAL

**Description** Video source signal type.

**Type** UINT32

**Limits** This value must match one of the supported capabilities of the acquisition device given by CORACQ\_CAP\_SIGNAL. The capability returns the ORed combination of all supported values.

<b>Values</b>	CORACQ_VAL_SIGNAL_SINGLE_ENDED (0x00000001)	Single ended signal.
	CORACQ_VAL_SIGNAL_DIFFERENTIAL (0x00000002)	Differential signal.

**CCA Entry** [Signal Description]  
Signal

## CORACQ\_PRM\_SYNC

<b>Description</b>	Synchronization source.
<b>Type</b>	UINT32
<b>Limits</b>	This value must match one of the supported capabilities of the acquisition device given by CORACQ_CAP_SYNC. The capability returns the ORed combination of all supported values.
<b>Values</b>	<p>CORACQ_VAL_SYNC_COMP_VIDEO (0x00000001), Composite video source.</p> <p>CORACQ_VAL_SYNC_COMP_SYNC (0x00000002), Composite sync source.</p> <p>CORACQ_VAL_SYNC_SEP_SYNC (0x00000004), Separate horizontal and vertical sync source.</p> <p>CORACQ_VAL_SYNC_INT_SYNC (0x00000008) Internal horizontal and vertical syncs generated by the acquisition device. See also CORACQ_PRM_MASTER_MODE.</p> <p>CORACQ_VAL_SYNC_RED (0x00000010), Composite video source from the red channel.</p> <p>CORACQ_VAL_SYNC_GREEN (0x00000020), Composite video source from the green channel.</p> <p>CORACQ_VAL_SYNC_BLUE (0x00000040) Composite video source from the blue channel.</p>
<b>CCA Entry</b>	[Synchronization Signals] Synchronization Source

## CORACQ\_PRM\_TAP\_1\_DIRECTION

<b>Description</b>	Specifies the direction of tap #1 of the video source.
<b>Type</b>	UINT32
<b>Limits</b>	This value must match one of the supported capabilities of the acquisition device given by CORACQ_CAP_TAP_DIRECTION. The capability returns the ORed combination of all supported values.
<b>Values</b>	<p>CORACQ_VAL_TAP_DIRECTION_LR (0x00000001) Pixels from the tap have a left to right order.</p> <p>CORACQ_VAL_TAP_DIRECTION_RL (0x00000002) Pixels from the tap have a right to left order.</p> <p>CORACQ_VAL_TAP_DIRECTION_UD (0x00000004) Lines from the tap have a top-bottom direction (up-down).</p> <p>CORACQ_VAL_TAP_DIRECTION_DU (0x00000008) Lines from the tap have a bottom-up direction (down-up).</p> <p>CORACQ_VAL_TAP_DIRECTION_FROM_TOP (0x00000010) Lines from the tap start at the top of the camera image.</p> <p>CORACQ_VAL_TAP_DIRECTION_FROM_MID (0x00000020) Lines from the tap start in the middle of the camera image.</p> <p>CORACQ_VAL_TAP_DIRECTION_FROM_BOT (0x00000040) Lines from the tap start at the bottom of the camera image.</p>
<b>CCA Entry</b>	[Signal Description] Tap 1 Direction
<b>Note</b>	For a description of tap geometries and corresponding parameter settings see Appendix: Tap Geometry Settings.

### **CORACQ\_PRM\_TAP\_2\_DIRECTION**

<b>Description</b>	Specifies the direction of tap #2 of the video source.
<b>Type</b>	UINT32
<b>Limits</b>	This value must match one of the supported capabilities of the acquisition device given by CORACQ_CAP_TAP_DIRECTION. The capability returns the ORed combination of all supported values.
<b>Values</b>	See CORACQ_PRM_TAP_1_DIRECTION.
<b>CCA Entry</b>	[Signal Description] Tap 2 Direction
<b>Note</b>	For a description of tap geometries and corresponding parameter settings see Appendix: Tap Geometry Settings.

### **CORACQ\_PRM\_TAP\_3\_DIRECTION**

<b>Description</b>	Specifies the direction of tap #3 of the video source.
<b>Type</b>	UINT32
<b>Limits</b>	This value must match one of the supported capabilities of the acquisition device given by CORACQ_CAP_TAP_DIRECTION. The capability returns the ORed combination of all supported values.
<b>Values</b>	See CORACQ_PRM_TAP_1_DIRECTION
<b>CCA Entry</b>	[Signal Description] Tap 3 Direction
<b>Note</b>	For a description of tap geometries and corresponding parameter settings see Appendix: Tap Geometry Settings.

### **CORACQ\_PRM\_TAP\_4\_DIRECTION**

<b>Description</b>	Specifies the direction of tap #4 of the video source.
<b>Type</b>	UINT32
<b>Limits</b>	This value must match one of the supported capabilities of the acquisition device given by CORACQ_CAP_TAP_DIRECTION. The capability returns the ORed combination of all supported values.
<b>Values</b>	See CORACQ_PRM_TAP_1_DIRECTION.
<b>CCA Entry</b>	[Signal Description] Tap 4 Direction
<b>Note</b>	For a description of tap geometries and corresponding parameter settings see Appendix: Tap Geometry Settings.

### **CORACQ\_PRM\_TAP\_5\_DIRECTION**

<b>Description</b>	Specifies the direction of tap #5 of the video source.
<b>Type</b>	UINT32
<b>Limits</b>	This value must match one of the supported capabilities of the acquisition device given by CORACQ_CAP_TAP_DIRECTION. The capability returns the ORed combination of all supported values.
<b>Values</b>	See CORACQ_PRM_TAP_1_DIRECTION.
<b>CCA Entry</b>	[Signal Description] Tap 5 Direction
<b>Note</b>	For a description of tap geometries and corresponding parameter settings see Appendix: Tap Geometry Settings.

### **CORACQ\_PRM\_TAP\_6\_DIRECTION**

<b>Description</b>	Specifies the direction of tap #6 of the video source.
<b>Type</b>	UINT32
<b>Limits</b>	This value must match one of the supported capabilities of the acquisition device given by CORACQ_CAP_TAP_DIRECTION. The capability returns the ORed combination of all supported values.
<b>Values</b>	See CORACQ_PRM_TAP_1_DIRECTION
<b>CCA Entry</b>	[Signal Description] Tap 6 Direction
<b>Note</b>	For a description of tap geometries and corresponding parameter settings see Appendix: Tap Geometry Settings.

### **CORACQ\_PRM\_TAP\_7\_DIRECTION**

<b>Description</b>	Specifies the direction of tap #7 of the video source.
<b>Type</b>	UINT32
<b>Limits</b>	This value must match one of the supported capabilities of the acquisition device given by CORACQ_CAP_TAP_DIRECTION. The capability returns the ORed combination of all supported values.
<b>Values</b>	See CORACQ_PRM_TAP_1_DIRECTION
<b>CCA Entry</b>	[Signal Description] Tap 7 Direction
<b>Note</b>	For a description of tap geometries and corresponding parameter settings see Appendix: Tap Geometry Settings.

### **CORACQ\_PRM\_TAP\_8\_DIRECTION**

<b>Description</b>	Specifies the direction of tap #8 of the video source.
<b>Type</b>	UINT32
<b>Limits</b>	This value must match one of the supported capabilities of the acquisition device given by CORACQ_CAP_TAP_DIRECTION. The capability returns the ORed combination of all supported values.
<b>Values</b>	See CORACQ_PRM_TAP_1_DIRECTION.
<b>CCA Entry</b>	[Signal Description] Tap 8 Direction
<b>Note</b>	For a description of tap geometries and corresponding parameter settings see Appendix: Tap Geometry Settings.

### **CORACQ\_PRM\_TAP\_9\_DIRECTION**

<b>Description</b>	Specifies the direction of tap #9 of the video source.
<b>Type</b>	UINT32
<b>Limits</b>	This value must match one of the supported capabilities of the acquisition device given by CORACQ_CAP_TAP_DIRECTION. The capability returns the ORed combination of all supported values.
<b>Values</b>	See CORACQ_PRM_TAP_1_DIRECTION
<b>CCA Entry</b>	[Signal Description] Tap 9 Direction
<b>Note</b>	For a description of tap geometries and corresponding parameter settings see Appendix: Tap Geometry Settings.

### **CORACQ\_PRM\_TAP\_10\_DIRECTION**

<b>Description</b>	Specifies the direction of tap #10 of the video source.
<b>Type</b>	UINT32
<b>Limits</b>	This value must match one of the supported capabilities of the acquisition device given by CORACQ_CAP_TAP_DIRECTION. The capability returns the ORed combination of all supported values.
<b>Values</b>	See CORACQ_PRM_TAP_1_DIRECTION
<b>CCA Entry</b>	[Signal Description] Tap 10 Direction
<b>Note</b>	For a description of tap geometries and corresponding parameter settings see Appendix: Tap Geometry Settings.

### **CORACQ\_PRM\_TAP\_11\_DIRECTION**

<b>Description</b>	Specifies the direction of tap #11 of the video source.
<b>Type</b>	UINT32
<b>Limits</b>	This value must match one of the supported capabilities of the acquisition device given by CORACQ_CAP_TAP_DIRECTION. The capability returns the ORed combination of all supported values.
<b>Values</b>	See CORACQ_PRM_TAP_1_DIRECTION
<b>CCA Entry</b>	[Signal Description] Tap 11 Direction
<b>Note</b>	For a description of tap geometries and corresponding parameter settings see Appendix: Tap Geometry Settings.

### **CORACQ\_PRM\_TAP\_12\_DIRECTION**

<b>Description</b>	Specifies the direction of tap #12 of the video source.
<b>Type</b>	UINT32
<b>Limits</b>	This value must match one of the supported capabilities of the acquisition device given by CORACQ_CAP_TAP_DIRECTION. The capability returns the ORed combination of all supported values.
<b>Values</b>	See CORACQ_PRM_TAP_1_DIRECTION
<b>CCA Entry</b>	[Signal Description] Tap 12 Direction
<b>Note</b>	For a description of tap geometries and corresponding parameter settings see Appendix: Tap Geometry Settings.

### **CORACQ\_PRM\_TAP\_13\_DIRECTION**

<b>Description</b>	Specifies the direction of tap #13 of the video source.
<b>Type</b>	UINT32
<b>Limits</b>	This value must match one of the supported capabilities of the acquisition device given by CORACQ_CAP_TAP_DIRECTION. The capability returns the ORed combination of all supported values.
<b>Values</b>	See CORACQ_PRM_TAP_1_DIRECTION
<b>CCA Entry</b>	[Signal Description] Tap 13 Direction
<b>Note</b>	For a description of tap geometries and corresponding parameter settings see Appendix: Tap Geometry Settings.

### **CORACQ\_PRM\_TAP\_14\_DIRECTION**

<b>Description</b>	Specifies the direction of tap #14 of the video source.
<b>Type</b>	UINT32
<b>Limits</b>	This value must match one of the supported capabilities of the acquisition device given by CORACQ_CAP_TAP_DIRECTION. The capability returns the ORed combination of all supported values.
<b>Values</b>	See CORACQ_PRM_TAP_1_DIRECTION
<b>CCA Entry</b>	[Signal Description] Tap 14 Direction
<b>Note</b>	For a description of tap geometries and corresponding parameter settings see Appendix: Tap Geometry Settings.

### **CORACQ\_PRM\_TAP\_15\_DIRECTION**

<b>Description</b>	Specifies the direction of tap #15 of the video source.
<b>Type</b>	UINT32
<b>Limits</b>	This value must match one of the supported capabilities of the acquisition device given by CORACQ_CAP_TAP_DIRECTION. The capability returns the ORed combination of all supported values.
<b>Values</b>	See CORACQ_PRM_TAP_1_DIRECTION
<b>CCA Entry</b>	[Signal Description] Tap 15 Direction
<b>Note</b>	For a description of tap geometries and corresponding parameter settings see Appendix: Tap Geometry Settings.

### **CORACQ\_PRM\_TAP\_16\_DIRECTION**

<b>Description</b>	Specifies the direction of tap #16 of the video source.
<b>Type</b>	UINT32
<b>Limits</b>	This value must match one of the supported capabilities of the acquisition device given by CORACQ_CAP_TAP_DIRECTION. The capability returns the ORed combination of all supported values.
<b>Values</b>	See CORACQ_PRM_TAP_1_DIRECTION
<b>CCA Entry</b>	[Signal Description] Tap 16 Direction
<b>Note</b>	For a description of tap geometries and corresponding parameter settings see Appendix: Tap Geometry Settings.



## CORACQ\_PRM\_TAP\_OUTPUT

<b>Description</b>	Specifies the tap output type of the video source.
<b>Type</b>	UINT32
<b>Limits</b>	This value must match one of the supported capabilities of the acquisition device given by CORACQ_CAP_TAP_OUTPUT. The capability returns the ORed combination of all supported values.
<b>Values</b>	<p>CORACQ_VAL_TAP_OUTPUT_ALTERNATE (0x00000001) Construction of a line is done by concatenating the taps 2 by 2, with a pixel in turn from each tap. So the first two taps makes up the first segment of the line, the next two taps make up the second segment... Must be an even number of tabs.</p> <p>CORACQ_VAL_TAP_OUTPUT_SEGMENTED (0x00000002) Construction of a line is done by concatenating the output of each tap.</p> <p>CORACQ_VAL_TAP_OUTPUT_PARALLEL (0x00000004) Construction of a line is done by concatenating a pixel in turn from each tap.</p>
<b>CCA Entry</b>	[Signal Description] Tap Output
<b>Note</b>	For a description of tap geometries and corresponding parameter settings see Appendix: Tap Geometry Settings.

## CORACQ\_PRM\_TAPS

<b>Description</b>	Number of taps output by the video source.
<b>Type</b>	UINT32
<b>Limits</b>	This value must be in the range 1..CORACQ_CAP_TAPS.
<b>CCA Entry</b>	[Signal Description] Taps
<b>Note</b>	For a description of tap geometries and corresponding parameter settings see Appendix: Tap Geometry Settings.

## CORACQ\_PRM\_TIMESLOT

<b>Description</b>	Number of pixel clocks needed to output 1 pixel on every tap
<b>Type</b>	UINT32
<b>Limits</b>	This value must match one of the supported capabilities of the acquisition device given by CORACQ_CAP_TIMESLOT
<b>Values</b>	<p>CORACQ_VAL_TIMESLOT_1 (0x01): for each pixel clock, a pixel from each tap is output (default)</p> <p>CORACQ_VAL_TIMESLOT_2 (0x02): 2 pixel clock cycles are needed to output 1 pixel from each tap</p> <p>CORACQ_VAL_TIMESLOT_3 (0x04): 3 pixel clock cycles are needed to output 1 pixel from each tap</p> <p>CORACQ_VAL_TIMESLOT_4 (0x08): 4 pixel clock cycles are needed to output 1 pixel from each tap</p>
<b>CCA Entry</b>	[Signal Description] Timeslot

## **CORACQ\_PRM\_TIME\_INTEGRATE\_METHOD**

<b>Description</b>	Method to use to control a camera's time integration. Applies to area scan cameras only.
<b>Type</b>	UINT32
<b>Limits</b>	This value must match one of the supported capabilities of the acquisition device given by CORACQ_CAP_TIME_INTEGRATE_METHOD. The capability returns the ORed combination of all supported values.
<b>Values</b>	See Time Integrate Methods
<b>CCA Entry</b>	[Control Signals] Time Integrate Method
<b>Note</b>	Available only if CORACQ_CAP_TIME_INTEGRATE is TRUE. CORACQ_CAP_TIME_INTEGRATE is obsolete. Use the equivalent parameter CORACQ_CAP_TIME_INTEGRATE_PULSE0_POLARITY. CORACQ_CAP_TIME_INTEGRATE_PULSE0_POLARITY Values: CORACQ_VAL_ACTIVE_LOW Time integration trigger pulse can be active low. CORACQ_VAL_ACTIVE_HIGH Time integration trigger pulse can be active high. Validated only when CORACQ_PRM_TIME_INTEGRATE_ENABLE is TRUE.

## **CORACQ\_PRM\_TIME\_INTEGRATE\_POLARITY**

<b>Description</b>	Obsolete. Use instead the equivalent parameter CORACQ_PRM_TIME_INTEGRATE_PULSE0_POLARITY
--------------------	------------------------------------------------------------------------------------------

## **CORACQ\_PRM\_TIME\_INTEGRATE\_PULSE\_DELAY**

<b>Description</b>	Obsolete. Use instead the equivalent parameter CORACQ_PRM_TIME_INTEGRATE_PULSE1_DELAY
--------------------	---------------------------------------------------------------------------------------

## **CORACQ\_PRM\_TIME\_INTEGRATE\_PULSE\_DURATION**

<b>Description</b>	Obsolete. Use instead the equivalent parameter CORACQ_PRM_TIME_INTEGRATE_PULSE1_DURATION
--------------------	------------------------------------------------------------------------------------------

## **CORACQ\_PRM\_TIME\_INTEGRATE\_PULSE\_POLARITY**

<b>Description</b>	Obsolete. Use instead the equivalent parameter CORACQ_PRM_TIME_INTEGRATE_PULSE1_POLARITY
--------------------	------------------------------------------------------------------------------------------

## **CORACQ\_PRM\_TIME\_INTEGRATE\_PULSE0\_DELAY**

<b>Description</b>	Time integration pulse #0 delay (in $\mu$ s). Applies to area scan cameras only.
<b>Type</b>	UINT32
<b>Limits</b>	Range limits: CORACQ_CAP_TIME_INTEGRATE_PULSE0_DELAY_MIN ... CORACQ_CAP_TIME_INTEGRATE_PULSE0_DELAY_MAX.
<b>CCA Entry</b>	[Control Signals] Time Integrate Pulse 0 Delay
<b>Note</b>	Available only if CORACQ_CAP_TIME_INTEGRATE is TRUE. Validated only when CORACQ_PRM_TIME_INTEGRATE_ENABLE is TRUE. See Time Integrate Methods for the different usages of the pulse #0 delay parameter.

### **CORACQ\_PRM\_TIME\_INTEGRATE\_PULSE0\_DURATION**

<b>Description</b>	Time integration pulse #0 width (in $\mu$ s). Applies to area scan cameras only.
<b>Type</b>	UINT32
<b>Limits</b>	Range limits: CORACQ_CAP_TIME_INTEGRATE_PULSE0_DURATION_MIN ... CORACQ_CAP_TIME_INTEGRATE_PULSE0_DURATION_MAX.
<b>CCA Entry</b>	[Control Signals] Time Integrate Pulse 0 Duration
<b>Note</b>	Available only if CORACQ_CAP_TIME_INTEGRATE is TRUE. Validated only when CORACQ_PRM_TIME_INTEGRATE_ENABLE is TRUE. See Time Integrate Methods for the different usages of the pulse #0 duration parameter.

### **CORACQ\_PRM\_TIME\_INTEGRATE\_PULSE0\_POLARITY**

<b>Description</b>	Time integration pulse #0 polarity. Applies to area scan cameras only.
<b>Type</b>	UINT32
<b>Limits</b>	This value must match one of the supported capabilities of the acquisition device given by CORACQ_CAP_TIME_INTEGRATE_PULSE0_POLARITY. The capability returns the ORed combination of all supported values. See CORACQ_PRM_TIME_INTEGRATE_METHOD for further information on CORACQ_CAP_TIME_INTEGRATE_PULSE0_POLARITY.
<b>Values</b>	CORACQ_VAL_ACTIVE_LOW (0x00000001), Time integration pulse is active low. CORACQ_VAL_ACTIVE_HIGH (0x00000002), Time integration pulse is active high.
<b>CCA Entry</b>	[Control Signals] Time Integrate Pulse 0 Polarity
<b>Note</b>	Available only if CORACQ_CAP_TIME_INTEGRATE is TRUE. Validated only when CORACQ_PRM_TIME_INTEGRATE_ENABLE is TRUE. See Time Integrate Methods for the different usages of the pulse #0.

### **CORACQ\_PRM\_TIME\_INTEGRATE\_PULSE1\_DELAY**

<b>Description</b>	Time integration pulse #1 delay (in $\mu$ s). Applies to area scan cameras only.
<b>Type</b>	UINT32
<b>Limits</b>	Range limits CORACQ_CAP_TIME_INTEGRATE_PULSE1_DELAY_MIN ... CORACQ_CAP_TIME_INTEGRATE_PULSE1_DELAY_MAX.
<b>CCA Entry</b>	[Control Signals] Time Integrate Pulse 1 Delay
<b>Note</b>	Available only if CORACQ_CAP_TIME_INTEGRATE is TRUE. Validated only when CORACQ_PRM_TIME_INTEGRATE_ENABLE is TRUE. See Time Integrate Methods for the different usages of the pulse #1 delay parameter.

### **CORACQ\_PRM\_TIME\_INTEGRATE\_PULSE1\_DURATION**

<b>Description</b>	Time integration pulse #1 width (in $\mu$ s). Applies to area scan cameras only.
<b>Type</b>	UINT32
<b>Limits</b>	Range limits: CORACQ_CAP_TIME_INTEGRATE_PULSE1_DURATION_MIN ... CORACQ_CAP_TIME_INTEGRATE_PULSE1_DURATION_MAX.
<b>CCA Entry</b>	[Control Signals] Time Integrate Pulse 1 Duration
<b>Note</b>	Available only if CORACQ_CAP_TIME_INTEGRATE is TRUE. Validated only when CORACQ_PRM_TIME_INTEGRATE_ENABLE is TRUE. See Time Integrate Methods for the different usages of the pulse #1 duration parameter.

### CORACQ\_PRM\_TIME\_INTEGRATE\_PULSE1\_POLARITY

<b>Description</b>	Time integration pulse #1 polarity. Applies to area scan cameras only.	
<b>Type</b>	UINT32	
<b>Limits</b>	This value must match one of the supported capabilities of the acquisition device given by CORACQ_CAP_TIME_INTEGRATE_PULSE1_POLARITY. The capability returns the ORed combination of all supported values.	
<b>Values</b>	CORACQ_VAL_ACTIVE_LOW (0x00000001)	Time integration trigger pulse is active low.
	CORACQ_VAL_ACTIVE_HIGH (0x00000002)	Time integration trigger pulse is active high.
<b>CCA Entry</b>	[Control Signals] Time Integrate Pulse 1 Polarity	
<b>Note</b>	Available only if CORACQ_CAP_TIME_INTEGRATE is TRUE. Validated only when CORACQ_PRM_TIME_INTEGRATE_ENABLE is TRUE. See Time Integrate Methods for the different usages of the pulse #1.	

### CORACQ\_PRM\_TRIGGER\_EXP\_SIGNAL

<b>Description</b>	Obsolete. Use CORACQ_PRM_CONNECTOR_xxx parameters to describe the pinout of the camera.
--------------------	-----------------------------------------------------------------------------------------

### CORACQ\_PRM\_VACTIVE

<b>Description</b>	Vertical active portion of the video (in lines per field). Applies to area scan cameras only.
<b>Type</b>	UINT32
<b>Limits</b>	Range limits: CORACQ_CAP_VACTIVE_MIN ... CORACQ_CAP_VACTIVE_MAX, and also must be a multiple of CORACQ_CAP_VACTIVE_MULT.
<b>CCA Entry</b>	[Signal Timings] Vertical Active

### CORACQ\_PRM\_VBACK\_INVALID

<b>Description</b>	Invalid vertical portion of the video following the vertical blanking (in lines per field). Applies to area scan cameras only.
<b>Type</b>	UINT32
<b>Limits</b>	Range limits: CORACQ_CAP_VBACK_INVALID_MIN ... CORACQ_CAP_VBACK_INVALID_MAX, and must be a multiple of CORACQ_CAP_VBACK_INVALID_MULT.
<b>CCA Entry</b>	[Signal Timings] Vertical Back Invalid

### CORACQ\_PRM\_VBACK\_PORCH

<b>Description</b>	Vertical back porch portion of the video (in lines per field). Applies to analog video signals only.
<b>Type</b>	UINT32
<b>Limits</b>	Range limits: CORACQ_CAP_VBACK_PORCH_MIN ... CORACQ_CAP_VBACK_PORCH_MAX, and must be a multiple of CORACQ_CAP_VBACK_PORCH_MULT.
<b>CCA Entry</b>	[Signal Timings] Vertical Back Porch

### CORACQ\_PRM\_VFRONT\_INVALID

<b>Description</b>	Invalid vertical portion of the video preceding the vertical blanking (in lines per field). Applies to area scan cameras only.
<b>Type</b>	UINT32
<b>Limits</b>	Range limits: CORACQ_CAP_VFRONT_INVALID_MIN ... CORACQ_CAP_VFRONT_INVALID_MAX, and must be a multiple of CORACQ_CAP_VFRONT_INVALID_MULT.
<b>CCA Entry</b>	[Signal Timings] Vertical Front Invalid

### CORACQ\_PRM\_VFRONT\_PORCH

<b>Description</b>	The video's vertical front porch (in lines per field). Applies to analog video signals only.
<b>Type</b>	UINT32
<b>Limits</b>	Range limits: CORACQ_CAP_VFRONT_PORCH_MIN ... CORACQ_CAP_VFRONT_PORCH_MAX, and must be a multiple of CORACQ_CAP_VFRONT_PORCH_MULT.
<b>CCA Entry</b>	[Signal Timings] Vertical Front Porch

### CORACQ\_PRM\_VIDEO

<b>Description</b>	Video type source.												
<b>Type</b>	UINT32												
<b>Limits</b>	This value must match one of the supported capabilities of the acquisition device given by CORACQ_CAP_VIDEO.												
<b>Values</b>	<table><tr><td>CORACQ_VAL_VIDEO_MONO (0x00000001)</td><td>Monochrome composite video source.</td></tr><tr><td>CORACQ_VAL_VIDEO_COLOR (0x00000002)</td><td>Color composite video source.</td></tr><tr><td>CORACQ_VAL_VIDEO_YC (0x00000004)</td><td>Y/C video source.</td></tr><tr><td>CORACQ_VAL_VIDEO_RGB (0x00000008)</td><td>RGB video source.</td></tr><tr><td>CORACQ_VAL_VIDEO_BAYER (0x00000010)</td><td>Bayer video source.</td></tr><tr><td>CORACQ_VAL_VIDEO_BICOLOR (0x00000020)</td><td>Bi-Color video source.</td></tr></table>	CORACQ_VAL_VIDEO_MONO (0x00000001)	Monochrome composite video source.	CORACQ_VAL_VIDEO_COLOR (0x00000002)	Color composite video source.	CORACQ_VAL_VIDEO_YC (0x00000004)	Y/C video source.	CORACQ_VAL_VIDEO_RGB (0x00000008)	RGB video source.	CORACQ_VAL_VIDEO_BAYER (0x00000010)	Bayer video source.	CORACQ_VAL_VIDEO_BICOLOR (0x00000020)	Bi-Color video source.
CORACQ_VAL_VIDEO_MONO (0x00000001)	Monochrome composite video source.												
CORACQ_VAL_VIDEO_COLOR (0x00000002)	Color composite video source.												
CORACQ_VAL_VIDEO_YC (0x00000004)	Y/C video source.												
CORACQ_VAL_VIDEO_RGB (0x00000008)	RGB video source.												
CORACQ_VAL_VIDEO_BAYER (0x00000010)	Bayer video source.												
CORACQ_VAL_VIDEO_BICOLOR (0x00000020)	Bi-Color video source.												
<b>CCA Entry</b>	[Signal Description] Video												

### CORACQ\_PRM\_VIDEO\_LEVEL\_MAX

<b>Description</b>	Maximum value (in $\mu\text{V}$ ) of the video signal. Applies to analog video signal only.
<b>Type</b>	UINT32
<b>Limits</b>	This value must be greater or equal to CORACQ_PRM_VIDEO_LEVEL_MIN and must be in the range: $[-(2^{31}) \dots (2^{31}) - 1]$ .
<b>CCA Entry</b>	[Signal Description] Video Level Maximum
<b>Note</b>	For NTSC/RS-170 video standard signal, this value is usually equal to 714000 $\mu\text{V}$ . For PAL/CCIR video standard signal, this value is usually equal to 700000 $\mu\text{V}$ . If CORACQ_PRM_VIDEO_LEVEL_MIN and CORACQ_PRM_VIDEO_LEVEL_MAX are both set to 0, then the following default values will be used: if PAL/CCIR video standard is selected: min = 0, max = 700000 else min = 53550, max = 714000.

### CORACQ\_PRM\_VIDEO\_LEVEL\_MIN

<b>Description</b>	Minimum value (in $\mu\text{V}$ ) of the video signal. Applies to analog video signals only.
<b>Type</b>	UINT32
<b>Limits</b>	This value must be smaller or equal to CORACQ_PRM_VIDEO_LEVEL_MAX and must be in the range: [ $-(2^{**31}) \dots (2^{**31})-1$ ].
<b>CCA Entry</b>	[Signal Description] Video Level Minimum
<b>Note</b>	For NTSC/RS-170 video standard signal, this value is usually equal to 53550 $\mu\text{V}$ . For PAL/CCIR video standard signal, this value is usually equal to 0 $\mu\text{V}$ . If CORACQ_PRM_VIDEO_LEVEL_MIN and CORACQ_PRM_VIDEO_LEVEL_MAX are both set to 0, then the following default values will be used: if PAL/CCIR video standard is selected: min = 0, max = 700000 else min = 53550, max = 714000.

### CORACQ\_PRM\_VIDEO\_STD

<b>Description</b>	Video source video standard.								
<b>Type</b>	UINT32								
<b>Limits</b>	This value must match one of the supported capabilities of the acquisition device given by CORACQ_CAP_VIDEO_STD. The capability returns the ORed combination of all supported values.								
<b>Values</b>	<table><tr><td>CORACQ_VAL_VIDEO_STD_NON_STD (0x00000001)</td><td>Non-standard video source.</td></tr><tr><td>CORACQ_VAL_VIDEO_STD_RS170_NTSC (0x00000002)</td><td>RS-170 and/or NTSC video source.</td></tr><tr><td>CORACQ_VAL_VIDEO_STD_CCIR_PAL (0x00000004)</td><td>CCIR and/or PAL video source.</td></tr><tr><td>CORACQ_VAL_VIDEO_STD_SECAM (0x00000008)</td><td>SECAM video source.</td></tr></table>	CORACQ_VAL_VIDEO_STD_NON_STD (0x00000001)	Non-standard video source.	CORACQ_VAL_VIDEO_STD_RS170_NTSC (0x00000002)	RS-170 and/or NTSC video source.	CORACQ_VAL_VIDEO_STD_CCIR_PAL (0x00000004)	CCIR and/or PAL video source.	CORACQ_VAL_VIDEO_STD_SECAM (0x00000008)	SECAM video source.
CORACQ_VAL_VIDEO_STD_NON_STD (0x00000001)	Non-standard video source.								
CORACQ_VAL_VIDEO_STD_RS170_NTSC (0x00000002)	RS-170 and/or NTSC video source.								
CORACQ_VAL_VIDEO_STD_CCIR_PAL (0x00000004)	CCIR and/or PAL video source.								
CORACQ_VAL_VIDEO_STD_SECAM (0x00000008)	SECAM video source.								
<b>CCA Entry</b>	[Signal Description] Video Standard								

### CORACQ\_PRM\_VSYNC

<b>Description</b>	The video's vertical sync (in lines per field). Applies to area scan cameras only.
<b>Type</b>	UINT32
<b>Limits</b>	This value must be in the range CORACQ_CAP_VSYNC_MIN...CORACQ_CAP_VSYNC_MAX, and must be a multiple of CORACQ_CAP_VSYNC_MULT.
<b>CCA Entry</b>	[Signal Timings] Vertical Sync

### CORACQ\_PRM\_VSYNC\_POLARITY

<b>Description</b>	Vertical sync polarity. Applies to area scan cameras only.				
<b>Type</b>	UINT32				
<b>Limits</b>	This value must match one of the supported capabilities of the acquisition device given by CORACQ_CAP_VSYNC_POLARITY. The capability returns the ORed combination of all supported values.				
<b>Values</b>	<table><tr><td>CORACQ_VAL_ACTIVE_LOW (0x00000001)</td><td>Vertical sync pulse is active low.</td></tr><tr><td>CORACQ_VAL_ACTIVE_HIGH (0x00000002)</td><td>Vertical sync pulse is active high.</td></tr></table>	CORACQ_VAL_ACTIVE_LOW (0x00000001)	Vertical sync pulse is active low.	CORACQ_VAL_ACTIVE_HIGH (0x00000002)	Vertical sync pulse is active high.
CORACQ_VAL_ACTIVE_LOW (0x00000001)	Vertical sync pulse is active low.				
CORACQ_VAL_ACTIVE_HIGH (0x00000002)	Vertical sync pulse is active high.				
<b>CCA Entry</b>	[Synchronization Signals] Vertical Sync Polarity				

**CORACQ\_PRM\_WEN\_POLARITY**

<b>Description</b>	Specifies the WEN (rite ENable) signal polarity that the acquisition device will consider as valid.	
<b>Type</b>	UINT32	
<b>Limits</b>	This value must match one of the supported capabilities of the acquisition device given by CORACQ_CAP_WEN_POLARITY. The capability returns the ORed combination of all supported values.	
<b>Values</b>	CORACQ_VAL_ACTIVE_LOW (0x00000001)	WEN is active low.
	CORACQ_VAL_ACTIVE_HIGH (0x00000002)	WEN is active high.
<b>CCA Entry</b>	[Control Signals] WEN Polarity	
<b>Note</b>	Validated only if CORACQ_PRM_WEN_ENABLE is TRUE.	



# Configuration File Formats

---

## Overview

This section covers the format descriptions for the information files describing camera definition parameters (.CCA) and acquisition parameters (.CVI). The camera configuration file (.CCF) is the combination of the .CCA and .CVI files into one file.

These parameters are stored in Sapera camera configuration files which an application loads to initialize the acquisition hardware. Note that all camera related parameters can be individually loaded by the application if a single acquisition source (hard-coded) program is desired.

Sapera LT supplies a number of camera definition files for popular cameras available on the market. The Sapera CamExpert tool simplifies making or modifying Sapera camera files and is described in the *Sapera LT User's manual*. Refer also to the CamExpert online help file and descriptive popup help for the various parameter fields.

Note that, in addition to the key names defined in the .CCA and .CVI files, the .CCF file may contain additional keys. This happens, for example, when saving a .CCF file for Teledyne DALSA cameras explicitly supported by CamExpert through the Camera Link serial port, for example, the Piranha Color. In some cases, the .CCA and .CVI keys are completely absent, and only the additional keys are present. This is the case for Genie cameras, which do not make use of the acquisition module with its capabilities and parameters.

---

## Camera Definition File Description (CCA)

Sapera camera files (\*.cca) contain the parameters of specific cameras. Most of the information found in these files is the default settings that should never change for a given camera. Values can be written in decimal (for example, 16) or in hexadecimal (for example, 0x10).

The following tables contain each key name used by camera files. Under normal circumstances each \*.cca file only the information required for a given camera. Note that the \*.cca file contains all Sapera camera related parameters whether they are used or needed by the camera.

Key Name [General]	Related Parameter
Camera Name	CORACQ_PRM_CAM_NAME
Company Name	CORACQ_PRM_CAM_COMPANY_NAME
Model Name	CORACQ_PRM_CAM_MODEL_NAME
Version	Version of this file. This entry does not correspond to any parameter. 100: Initial Version 200: Formats are now indexes into a fix table independent of the Sapera values 300: Parameter CORACQ_PRM_TIME_INTEGRATE_POLARITY is now called CORACQ_PRM_TIME_INTEGRATE_PULSE0_POLARITY 301: New parameters CORACQ_PRM_LINE_INTEGRATE_PULSE_xxx, CORACQ_PRM_VIDEO_LEVEL_MIN/MAX

Key Name [Signal Description]	Related Parameter
Bayer Alignment	CORACQ_PRM_COLOR_ALIGNMENT
Channel	CORACQ_PRM_CHANNEL
Channels Order	CORACQ_PRM_CHANNELS_ORDER
Coupling	CORACQ_PRM_COUPLING
Field Order	CORACQ_PRM_FIELD_ORDER
Frame	CORACQ_PRM_FRAME
Interface	CORACQ_PRM_INTERFACE
Pixel Depth	CORACQ_PRM_PIXEL_DEPTH
Scan	CORACQ_PRM_SCAN
Signal	CORACQ_PRM_SIGNAL
Tap Output	CORACQ_PRM_TAP_OUTPUT
Tap 1 Direction	CORACQ_PRM_TAP_1_DIRECTION
Tap 2 Direction	CORACQ_PRM_TAP_2_DIRECTION
Tap 3 Direction	CORACQ_PRM_TAP_3_DIRECTION
Tap 4 Direction	CORACQ_PRM_TAP_4_DIRECTION
Tap 5 Direction	CORACQ_PRM_TAP_5_DIRECTION
Tap 6 Direction	CORACQ_PRM_TAP_6_DIRECTION
Tap 7 Direction	CORACQ_PRM_TAP_7_DIRECTION
Tap 8 Direction	CORACQ_PRM_TAP_8_DIRECTION
Taps	CORACQ_PRM_TAPS
Video	CORACQ_PRM_VIDEO
Video Level Maximum	CORACQ_PRM_VIDEO_LEVEL_MAX
Video Level Minimum	CORACQ_PRM_VIDEO_LEVEL_MIN
Video Standard	CORACQ_PRM_VIDEO_STD

**Key Name**  
**[Signal Timings]**

Horizontal Active	CORACQ_PRM_HACTIVE
Horizontal Back Invalid	CORACQ_PRM_HBACK_INVALID
Horizontal Back Porch	CORACQ_PRM_HBACK_PORCH
Horizontal Front Invalid	CORACQ_PRM_HFRONT_INVALID
Horizontal Front Porch	CORACQ_PRM_HFRONT_PORCH
Horizontal Sync	CORACQ_PRM_HSYNC
Vertical Active	CORACQ_PRM_VACTIVE
Vertical Back Invalid	CORACQ_PRM_VBACK_INVALID
Vertical Back Porch	CORACQ_PRM_VBACK_PORCH
Vertical Front Invalid	CORACQ_PRM_VFRONT_INVALID
Vertical Front Porch	CORACQ_PRM_VFRONT_PORCH
Vertical Sync	CORACQ_PRM_VSYNC

**Key Name**  
**[Pixel Clock]**

Pixel Clock Detection	CORACQ_PRM_PIXEL_CLK_DETECTION
Pixel Clock Frequency External	CORACQ_PRM_PIXEL_CLK_EXT
Pixel Clock Frequency Internal	CORACQ_PRM_PIXEL_CLK_INT
Pixel Clock Frequency 1:1	CORACQ_PRM_PIXEL_CLK_11
Pixel Clock Source	CORACQ_PRM_PIXEL_CLK_SRC

**Key Name**  
**[Synchronization Signals]**

Horizontal Sync Polarity	CORACQ_PRM_HSYNC_POLARITY
Synchronization Source	CORACQ_PRM_SYNC
Vertical Sync Polarity	CORACQ_PRM_VSYNC_POLARITY

**Key Name**  
**[Control Signals]**

Camera Line Trigger Frequency Maximum	CORACQ_PRM_CAM_LINE_TRIGGER_FREQ_MAX
Camera Line Trigger Frequency Minimum	CORACQ_PRM_CAM_LINE_TRIGGER_FREQ_MIN
Camera Reset Duration	CORACQ_PRM_CAM_RESET_DURATION
Camera Reset Method	CORACQ_PRM_CAM_RESET_METHOD
Camera Reset Polarity	CORACQ_PRM_CAM_RESET_POLARITY
Camera Time Integrate Duration Maximum	CORACQ_PRM_CAM_TIME_INTEGRATE_DURATION_MAX
Camera Time Integrate Duration Minimum	CORACQ_PRM_CAM_TIME_INTEGRATE_DURATION_MIN
Camera Trigger Duration	CORACQ_PRM_CAM_TRIGGER_DURATION
Camera Trigger Method	CORACQ_PRM_CAM_TRIGGER_METHOD
Camera Trigger Polarity	CORACQ_PRM_CAM_TRIGGER_POLARITY
Data Valid Enable	CORACQ_PRM_DATA_VALID_ENABLE
Data Valid Polarity	CORACQ_PRM_DATA_VALID_POLARITY

**Key Name**  
**[Control Signals]****Related Parameter**

Frame Integrate Method	CORACQ_PRM_FRAME_INTEGRATE_METHOD
Frame Integrate Polarity	CORACQ_PRM_FRAME_INTEGRATE_POLARITY
Line Integrate Method	CORACQ_PRM_LINE_INTEGRATE_METHOD
Line Integrate Pulse 0 Delay	CORACQ_PRM_LINE_INTEGRATE_PULSE0_DELAY
Line Integrate Pulse 0 Duration	CORACQ_PRM_LINE_INTEGRATE_PULSE0_DURATION
Line Integrate Pulse 0 Polarity	CORACQ_PRM_LINE_INTEGRATE_PULSE0_POLARITY
Line Integrate Pulse 1 Delay	CORACQ_PRM_LINE_INTEGRATE_PULSE1_DELAY
Line Integrate Pulse 1 Duration	CORACQ_PRM_LINE_INTEGRATE_PULSE1_DURATION
Line Integrate Pulse 1 Polarity	CORACQ_PRM_LINE_INTEGRATE_PULSE1_POLARITY
Line Trigger Delay	CORACQ_PRM_LINE_TRIGGER_DELAY
Line Trigger Duration	CORACQ_PRM_LINE_TRIGGER_DURATION
Line Trigger Method	CORACQ_PRM_LINE_TRIGGER_METHOD
Line Trigger Polarity	CORACQ_PRM_LINE_TRIGGER_POLARITY
LineScan Direction	CORACQ_PRM_LINESCAN_DIRECTION
LineScan Direction Polarity	CORACQ_PRM_LINESCAN_DIRECTION_POLARITY
Time Integrate Method	CORACQ_PRM_TIME_INTEGRATE_METHOD
Time Integrate Pulse 0 Delay	CORACQ_PRM_TIME_INTEGRATE_PULSE0_DELAY
Time Integrate Pulse 0 Duration	CORACQ_PRM_TIME_INTEGRATE_PULSE0_DURATION
Time Integrate Pulse 0 Polarity	CORACQ_PRM_TIME_INTEGRATE_PULSE0_POLARITY
Time Integrate Pulse 1 Delay	CORACQ_PRM_TIME_INTEGRATE_PULSE1_DELAY
Time Integrate Pulse 1 Duration	CORACQ_PRM_TIME_INTEGRATE_PULSE1_DURATION
Time Integrate Pulse 1 Polarity	CORACQ_PRM_TIME_INTEGRATE_PULSE1_POLARITY
WEN Polarity	CORACQ_PRM_WEN_POLARITY

**Key Name**  
**[Connector Description]****Related Parameter**

Camera Link Configuration	CORACQ_PRM_CAMLINK_CONFIGURATION
Exposure Input	CORACQ_PRM_CONNECTOR_EXPOSURE_INPUT
HD Input	CORACQ_PRM_CONNECTOR_HD_INPUT
Line Integrate Input	CORACQ_PRM_CONNECTOR_LINE_INTEGRATE_INPUT
Line Trigger Input	CORACQ_PRM_CONNECTOR_LINE_TRIGGER_INPUT
Linescan Direction Input	CORACQ_PRM_CONNECTOR_LINESCAN_DIRECTION_INPUT
Pixel Clock Output	CORACQ_PRM_CONNECTOR_PIXEL_CLK_OUTPUT
Reset/Trigger Input	CORACQ_PRM_CONNECTOR_RESET_TRIGGER_INPUT
VD Input	CORACQ_PRM_CONNECTOR_VD_INPUT
WEN Output	CORACQ_PRM_CONNECTOR_WEN_OUTPUT

**Key Name**  
**[Custom Camera IO Control Signals]****Related Parameter**

Max Control	This entry does not correspond to any parameter. The entry represents the number of custom I/O control defined in this section of the CCA file. ex. Max Control = 4
Control_0	CORACQ_PRM_CAM_IO_CONTROL

**Key Name**  
**[Custom Camera IO Control**  
**Signals]**

**Related Parameter**

This entry has the following format:  
label, bits, level, input/output, polarity, default

label: user defined descriptive label of the camera control (for example, BIN)

bits: number of bits used by this control

level: TTL/RS-422/24VOLTS/OPTO/LVTTL/12VOLTS

input/output: direction of the control

polarity: active high/low

default: default value

pin(optional): pin connector description

ex.     Control\_1=CC1, 1, 2, 2, 2, 1  
          Control\_0=CC1, 1, 2, 2, 2, 1,0x01020001  
          Control\_1=CC2, 1, 2, 2, 2, 1,0x01020002  
          Control\_2=CC3, 1, 2, 2, 2, 1,0x01020003  
          Control\_3=CC4, 1, 2, 2, 2, 1,0x01020004

see also CORACQ\_CAM\_IO\_CONTROL

see also Pin Connector Description

Control\_31

Control\_31, 1, 2, 2, 2, 0 or Control\_31,1,2,2,2,0,0x01020001

---

## VIC Parameter File Description (CVI)

VIC parameter files (\*.cvi) contain the VIC settings for a specific acquisition module. Values can be written in decimal (for example, 16) or in hexadecimal (for example, 0x10). The following tables contain the key names used by the VIC parameter files.

Key Name [General]	Related Parameter
Vic Name	CORACQ_PRM_VIC_NAME
Version	Version of this file. This entry does not correspond to any parameter. 100: Initial Version 200: Formats are now indexes into a fix table independent of the Sapera values 300: New Parameters CORACQ_PRM_SHARED_xxx, CORACQ_PRM_FRAME_LENGTH, CORACQ_PRM_INT_FRAME_TRIGGER_xxx, CORACQ_PRM_EXT_TRIGGER_FRAME_COUNT

Key Name [Input]	Related Parameter
Bit Ordering	CORACQ_PRM_BIT_ORDERING
Camera selector	CORACQ_PRM_CAMSEL
Planar Input Sources	CORACQ_PRM_PLANAR_INPUT_SOURCES

Key Name [Signal Conditioning]	Related Parameter
Brightness	CORACQ_PRM_BRIGHTNESS
Brightness Red	CORACQ_PRM_BRIGHTNESS_RED
Brightness Green	CORACQ_PRM_BRIGHTNESS_GREEN
Brightness Blue	CORACQ_PRM_BRIGHTNESS_BLUE
Contrast	CORACQ_PRM_CONTRAST
Contrast Red	CORACQ_PRM_CONTRAST_RED
Contrast Green	CORACQ_PRM_CONTRAST_GREEN
Contrast Blue	CORACQ_PRM_CONTRAST_BLUE
DC Restoration Mode	CORACQ_PRM_DC_REST_MODE
DC Restoration Start	CORACQ_PRM_DC_REST_START
DC Restoration Width	
Fix Filter Enable	CORACQ_PRM_FIX_FILTER_ENABLE
Fix Filter Selector	CORACQ_PRM_FIX_FILTER_SELECTOR
Hue	CORACQ_PRM_HUE
Programmable Filter Enable	CORACQ_PRM_PROG_FILTER_ENABLE
Programmable Filter Frequency	CORACQ_PRM_PROG_FILTER_FREQ
Saturation	CORACQ_PRM_SATURATION
Sharpness	CORACQ_PRM_SHARPNESS

Key Name [Stream Conditioning]	Related Parameter
Bayer Decoder Enable	CORACQ_PRM_COLOR_DECODER_ENABLE

<b>Key Name</b> <b>[Stream Conditioning]</b>	<b>Related Parameter</b>
Bayer Decoder Method	CORACQ_PRM_COLOR_DECODER_METHOD
Bayer Decoder White Balance Gain Red	CORACQ_PRM_WB_GAIN_RED
Bayer Decoder White Balance Gain Green	CORACQ_PRM_WB_GAIN_GREEN
Bayer Decoder White Balance Gain Blue	CORACQ_PRM_WB_GAIN_BLUE
Bayer Decoder White Balance Offset Red	CORACQ_PRM_WB_OFFSET_RED
Bayer Decoder White Balance Offset Green	CORACQ_PRM_WB_OFFSET_GREEN
Bayer Decoder White Balance Offset Blue	CORACQ_PRM_WB_OFFSET_BLUE
Crop Left	CORACQ_PRM_CROP_LEFT
Crop Top	CORACQ_PRM_CROP_TOP
Crop Height	CORACQ_PRM_CROP_HEIGHT
Crop Width	CORACQ_PRM_CROP_WIDTH
Decimate Count	CORACQ_PRM_DECIMATE_COUNT
Decimate Method	CORACQ_PRM_DECIMATE_METHOD
External Trigger Frame Count	CORACQ_PRM_EXT_TRIGGER_FRAME_COUNT
Frame Length	CORACQ_PRM_FRAME_LENGTH
Horizontal Sync Reference	CORACQ_PRM_HSYNC_REF
Lut Enable	CORACQ_PRM_LUT_ENABLE
Lut Number	CORACQ_PRM_LUT_NUMBER
Pixel Mask	CORACQ_PRM_PIXEL_MASK
Scale Horizontal	CORACQ_PRM_SCALE_HORZ
Scale Horizontal Method	CORACQ_PRM_SCALE_HORZ_METHOD
Scale Vertical	CORACQ_PRM_SCALE_VERT
Scale Vertical Method	CORACQ_PRM_SCALE_VERT_METHOD
Snap Count	CORACQ_PRM_SNAP_COUNT
Vertical Sync Reference	CORACQ_PRM_VSYNC_REF

<b>Key Name</b> <b>[Control Signals]</b>	<b>Related Parameter</b>
Camera Control Pulse 0 HD Align	CORACQ_PRM_CAM_CONTROL_PULSE0_HD_ALIGN
Camera Control Pulse 1 HD Align	CORACQ_PRM_CAM_CONTROL_PULSE1_HD_ALIGN
Camera Reset Delay	CORACQ_PRM_CAM_RESET_DELAY
Camera Reset Enable	CORACQ_PRM_CAM_RESET_ENABLE
Camera Trigger Delay	CORACQ_PRM_CAM_TRIGGER_DELAY
Camera Trigger Enable	CORACQ_PRM_CAM_TRIGGER_ENABLE
Control Signal Output 1	CORACQ_PRM_BOARD_SYNC_OUTPUT1
Control Signal Output 2	CORACQ_PRM_BOARD_SYNC_OUTPUT2
External Frame Trigger Detection	CORACQ_PRM_EXT_FRAME_TRIGGER_DETECTION
External Frame Trigger Enable	CORACQ_PRM_EXT_FRAME_TRIGGER_ENABLE



<b>Key Name [Control Signals]</b>	<b>Related Parameter</b>
External Frame Trigger Level	CORACQ_PRM_EXT_FRAME_TRIGGER_LEVEL
External Frame Trigger Source	CORACQ_PRM_EXT_FRAME_TRIGGER_SOURCE
External Line Trigger Detection	CORACQ_PRM_EXT_LINE_TRIGGER_DETECTION
External Line Trigger Enable	CORACQ_PRM_EXT_LINE_TRIGGER_ENABLE
External Line Trigger Level	CORACQ_PRM_EXT_LINE_TRIGGER_LEVEL
External Line Trigger Source	CORACQ_PRM_EXT_LINE_TRIGGER_SOURCE
External Trigger Detection	CORACQ_PRM_EXT_TRIGGER_DETECTION
External Trigger Duration	CORACQ_PRM_EXT_TRIGGER_DURATION
External Trigger Enable	CORACQ_PRM_EXT_TRIGGER_ENABLE
External Trigger Ignore Delay	CORACQ_PRM_EXT_TRIGGER_IGNORE_DELAY
External Trigger Level	CORACQ_PRM_EXT_TRIGGER_LEVEL
External Trigger Source	CORACQ_PRM_EXT_TRIGGER_SOURCE
Frame Integrate Count	CORACQ_PRM_FRAME_INTEGRATE_COUNT
Frame Integrate Enable	CORACQ_PRM_FRAME_INTEGRATE_ENABLE
Internal Frame Trigger Enable	CORACQ_PRM_INT_FRAME_TRIGGER_ENABLE
Internal Frame Trigger Freq	CORACQ_PRM_INT_FRAME_TRIGGER_FREQ
Internal Line Trigger Enable	CORACQ_PRM_INT_LINE_TRIGGER_ENABLE
Internal Line Trigger Freq	CORACQ_PRM_INT_LINE_TRIGGER_FREQ
Line Integrate Duration	CORACQ_PRM_LINE_INTEGRATE_DURATION
Line Integrate Enable	CORACQ_PRM_LINE_INTEGRATE_ENABLE
Line Trigger Enable	CORACQ_PRM_LINE_TRIGGER_ENABLE
LineScan Direction Output	CORACQ_PRM_LINESCAN_DIRECTION_OUTPUT
Master Mode	CORACQ_PRM_MASTER_MODE
Master Mode Horizontal Sync Polarity	CORACQ_PRM_MASTER_MODE_HSYNC_POLARITY
Master Mode Vertical Sync Polarity	CORACQ_PRM_MASTER_MODE_VSYNC_POLARITY
Shaft Encoder Enable	CORACQ_PRM_SHAFT_ENCODER_ENABLE
Shaft Encoder Level	CORACQ_PRM_SHAFT_ENCODER_LEVEL
Shaft Encoder Pulse Drop	CORACQ_PRM_SHAFT_ENCODER_DROP
Strobe Delay	CORACQ_PRM_STROBE_DELAY
Strobe Delay 2	CORACQ_PRM_STROBE_DELAY_2
Strobe Duration	CORACQ_PRM_STROBE_DURATION
Strobe Enable	CORACQ_PRM_STROBE_ENABLE
Strobe Level	CORACQ_PRM_STROBE_LEVE
Strobe Method	CORACQ_PRM_STROBE_METHOD
Strobe Polarity	CORACQ_PRM_STROBE_POLARITY
Time Integrate Delay	CORACQ_PRM_TIME_INTEGRATE_DELAY
Time Integrate Duration	CORACQ_PRM_TIME_INTEGRATE_DURATION
Time Integrate Enable	CORACQ_PRM_TIME_INTEGRATE_ENABLE
Vertical Timeout Delay	CORACQ_PRM_VERTICAL_TIMEOUT_DELAY
WEN Enable	CORACQ_PRM_WEN_ENABLE

<b>Key Name [Output]</b>	<b>Related Parameter</b>
------------------------------	--------------------------

Output Enable	CORACQ_PRM_OUTPUT_ENABLE	
Output Format	1: Mono 8	15: RGB161616
	3: Mono 16	19: UYVY
	7: Mono 32	20: YUY2
	10: RGB5551	21: YVYU
	11: RGB565	22: YUYV
	12: RGB888	23: Y411
	13: RGB8888	24: Y211
	14: RGB101010	38: RGB16161616
See also CORACQ_PRM_OUTPUT_FORMAT		

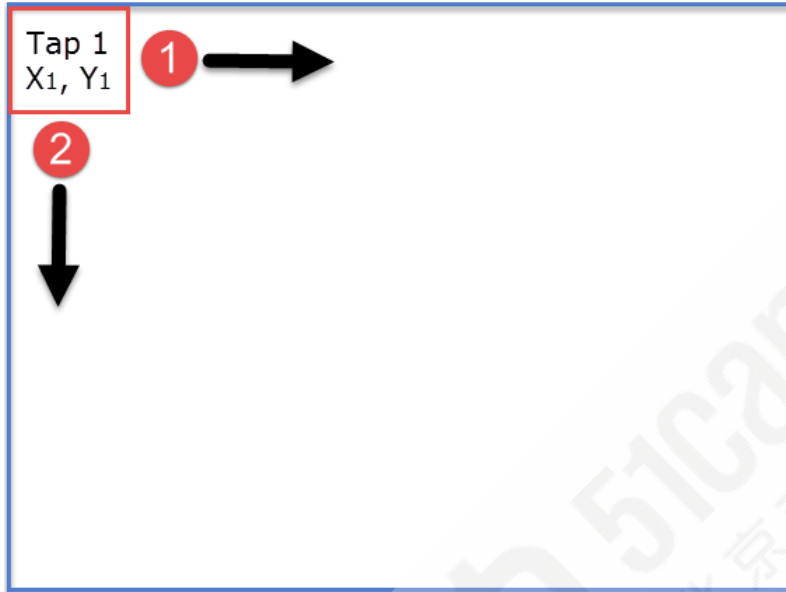
Key Name [Shared Control Signals]	Related Parameter
Camera Reset	CORACQ_PRM_SHARED_CAM_RESET
Camera Trigger	CORACQ_PRM_SHARED_CAM_TRIGGER
External Trigger	CORACQ_PRM_SHARED_EXT_TRIGGER
Frame Integrate	CORACQ_PRM_SHARED_FRAME_INTEGRATE
Strobe	CORACQ_PRM_SHARED_STROBE
Time Integrate	CORACQ_PRM_SHARED_TIME_INTEGRATE

# Appendix: Tap Geometry Settings

The following sections describe widely used tap geometries and the required parameter settings. Currently, only are scan geometries are described, but line scan geometries can be inferred from these settings. The GeniCam standard naming convention is used, including corresponding descriptions.

## 1 Single Tap Geometries

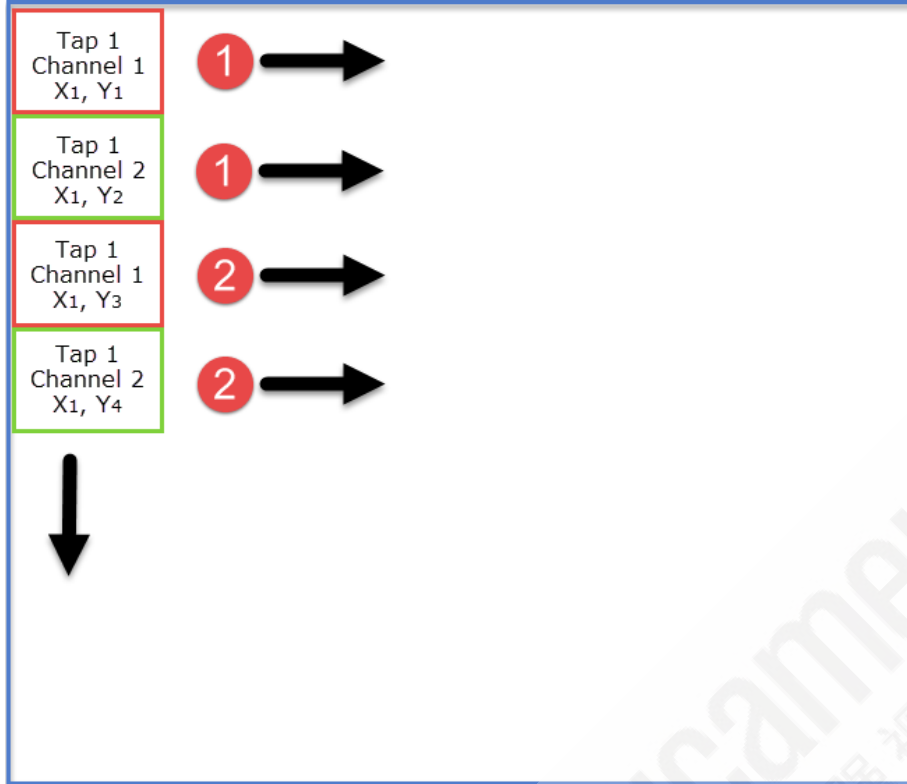
**1X-1Y (area-scan): 1 zone in X, 1 Zone in Y = One tap left to right.**



Parameter	Value
CORACQ_PRM_TAPS	1
CORACQ_PRM_TAP_OUTPUT	CORACQ_VAL_TAP_OUTPUT_SEGMENTED (0x00000002)
CORACQ_PRM_TAP_1_DIRECTION	The following values ORed (decimal = 21, hex = 0x15) CORACQ_VAL_TAP_DIRECTION_LR (0x00000001) CORACQ_VAL_TAP_DIRECTION_UD (0x00000004) CORACQ_VAL_TAP_DIRECTION_FROM_TOP (0x00000010)
CORACQ_PRM_CHANNEL	1
CORACQ_PRM_CHANNELS_ORDER	CORACQ_VAL_CHANNELS_ORDER_NORMAL

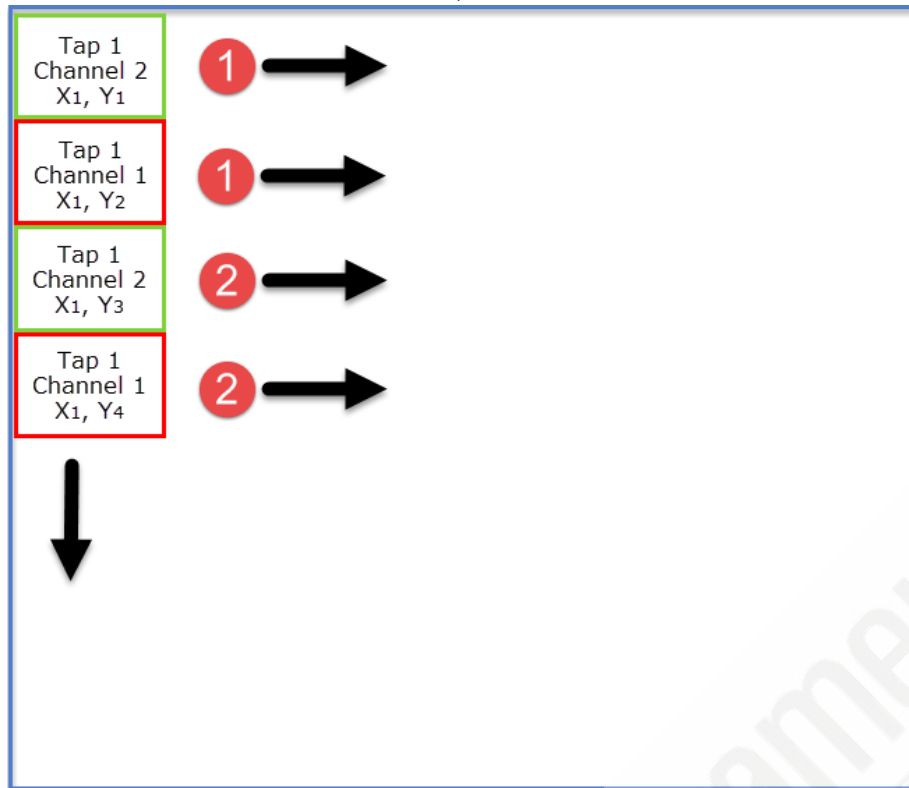
## One Tap with Two Channels

**1X-1Y2 (area-scan): 1 zones in X, 2 zones in Y: 2 interline channel, even A**



Parameter	Value
CORACQ_PRM_TAPS	1
CORACQ_PRM_TAP_OUTPUT	CORACQ_VAL_TAP_OUTPUT_SEGMENTED (0x00000002)
CORACQ_PRM_TAP_1_DIRECTION	The following values ORed (decimal = 21, hex = 0x15) CORACQ_VAL_TAP_DIRECTION_LR (0x00000001) CORACQ_VAL_TAP_DIRECTION_UD (0x00000004) CORACQ_VAL_TAP_DIRECTION_FROM_TOP (0x00000010)
CORACQ_PRM_CHANNEL	2
CORACQ_PRM_CHANNELS_ORDER	CORACQ_VAL_CHANNELS_ORDER_NORMAL (0x00000001)

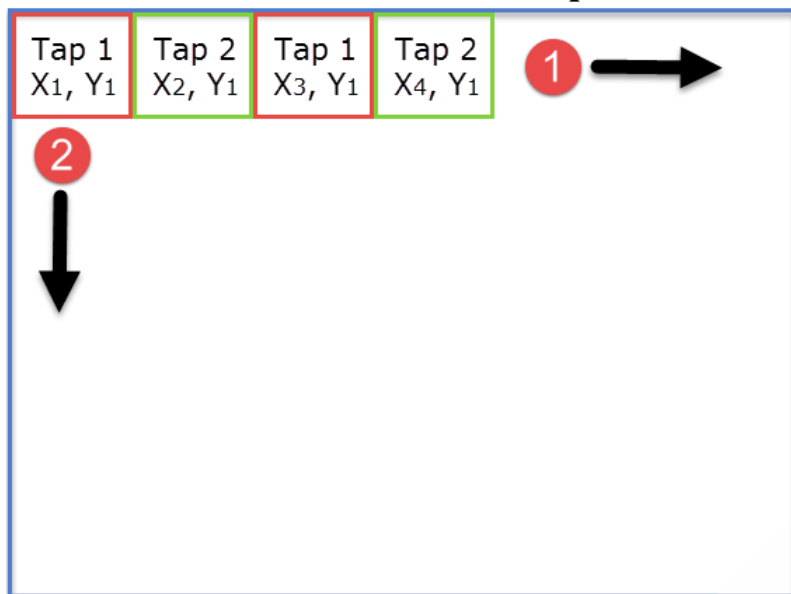
**1X-1Y2 (area-scan): 1 zones in X, 1 zone in Y: 2 interline channel, even B**



Parameter	Value
CORACQ_PRM_TAPS	1
CORACQ_PRM_TAP_OUTPUT	CORACQ_VAL_TAP_OUTPUT_SEGMENTED (0x00000002)
CORACQ_PRM_TAP_1_DIRECTION	The following values ORed (decimal = 21, hex = 0x15) CORACQ_VAL_TAP_DIRECTION_LR (0x00000001) CORACQ_VAL_TAP_DIRECTION_UD (0x00000004) CORACQ_VAL_TAP_DIRECTION_FROM_TOP (0x00000010)
CORACQ_PRM_CHANNEL	2
CORACQ_PRM_CHANNELS_ORDER	CORACQ_VAL_CHANNELS_ORDER_REVERSE (0x00000002)

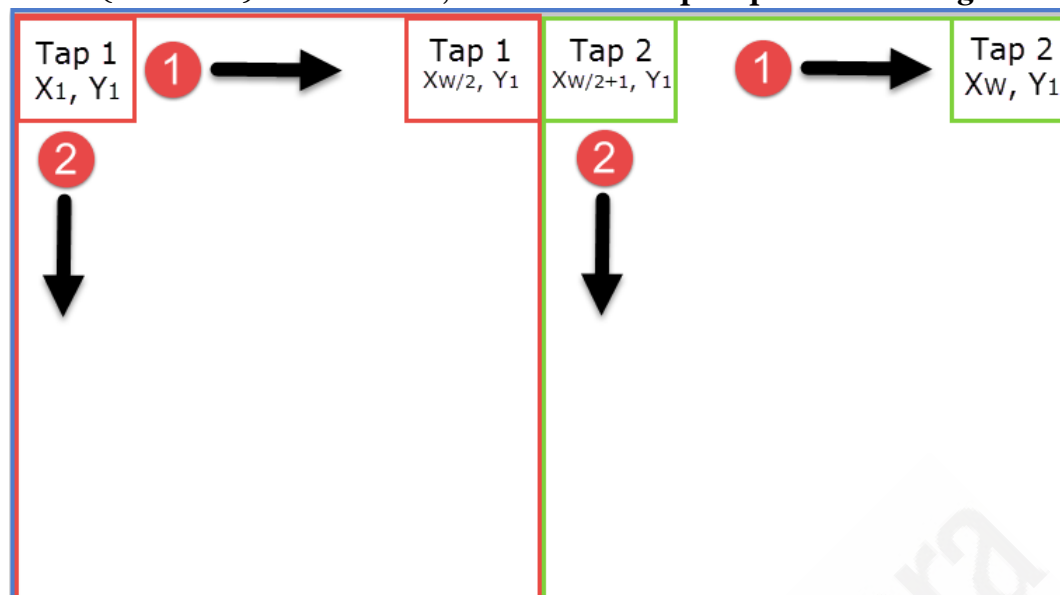
## Dual Tap Geometries

**1X2-1Y (area-scan): 1 zone in X with 2 taps, 1 Zone in Y = 2 Taps Interleaved**



Parameter	Value
CORACQ_PRM_TAPS	2
CORACQ_PRM_TAP_OUTPUT	CORACQ_VAL_TAP_OUTPUT_PARALLEL (0x00000004)
CORACQ_PRM_TAP_1_DIRECTION	The following values ORed (decimal = 21, hex = 0x15) CORACQ_VAL_TAP_DIRECTION_LR (0x00000001) CORACQ_VAL_TAP_DIRECTION_UD (0x00000004) CORACQ_VAL_TAP_DIRECTION_FROM_TOP (0x00000010)
CORACQ_PRM_TAP_2_DIRECTION	Same as CORACQ_PRM_TAP_1_DIRECTION
CORACQ_PRM_CHANNEL	1
CORACQ_PRM_CHANNELS_ORDER	CORACQ_VAL_CHANNELS_ORDER_NORMAL

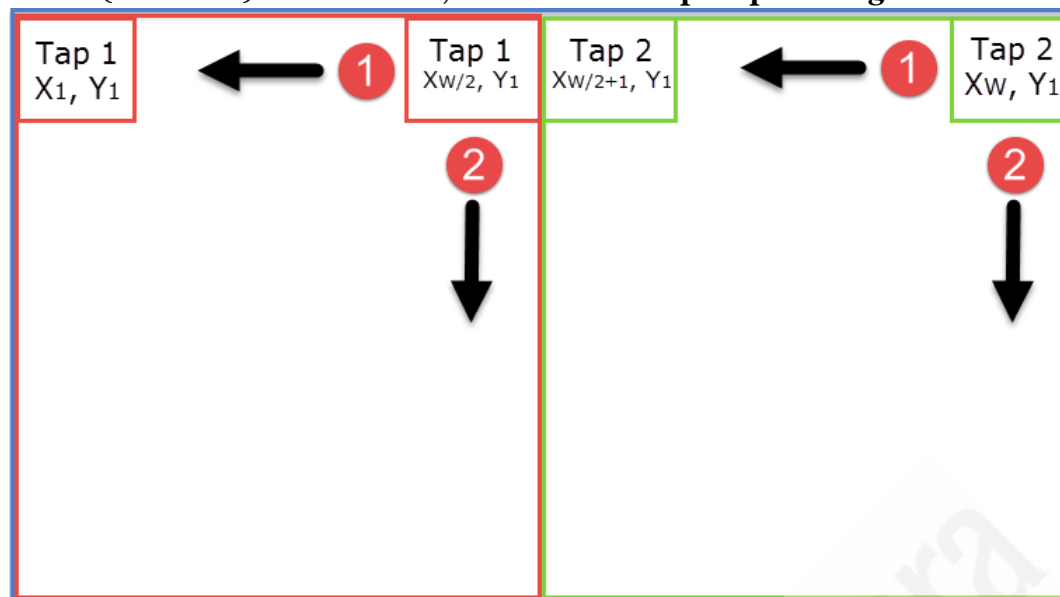
**2X-1Y (area-scan): 2 zones in X, 1 zone in Y: 2 taps separate left to right**



Parameter	Value
CORACQ_PRM_TAPS	2
CORACQ_PRM_TAP_OUTPUT	CORACQ_VAL_TAP_OUTPUT_SEGMENTED (0x00000002)
CORACQ_PRM_TAP_1_DIRECTION	The following values ORed (decimal = 21, hex = 0x15) CORACQ_VAL_TAP_DIRECTION_LR (0x00000001) CORACQ_VAL_TAP_DIRECTION_UD (0x00000004) CORACQ_VAL_TAP_DIRECTION_FROM_TOP (0x00000010)
CORACQ_PRM_TAP_2_DIRECTION	Same as CORACQ_PRM_TAP_1_DIRECTION
CORACQ_PRM_CHANNEL	1
CORACQ_PRM_CHANNELS_ORDER	CORACQ_VAL_CHANNELS_ORDER_NORMAL

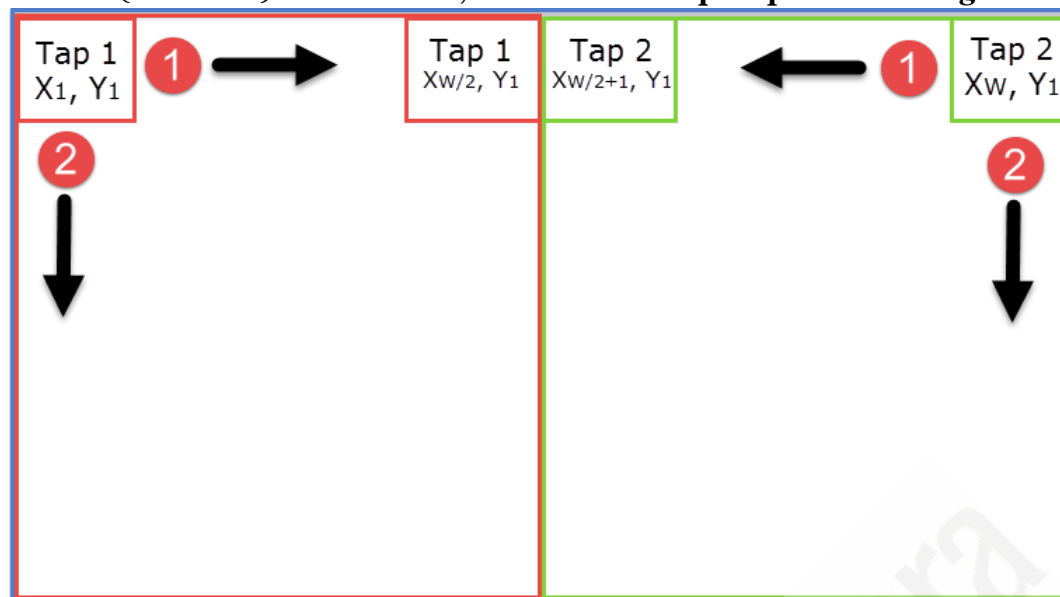


**2X-1Y (area-scan): 2 zones in X, 1 zone in Y: 2 taps separate right to left**



Parameter	Value
CORACQ_PRM_TAPS	2
CORACQ_PRM_TAP_OUTPUT	CORACQ_VAL_TAP_OUTPUT_SEGMENTED (0x00000002)
CORACQ_PRM_TAP_1_DIRECTION	The following values ORed (decimal = 22, hex = 0x16) CORACQ_VAL_TAP_DIRECTION_RL (0x00000002) CORACQ_VAL_TAP_DIRECTION_UD (0x00000004) CORACQ_VAL_TAP_DIRECTION_FROM_TOP (0x00000010)
CORACQ_PRM_TAP_2_DIRECTION	Same as CORACQ_PRM_TAP_1_DIRECTION
CORACQ_PRM_CHANNEL	1
CORACQ_PRM_CHANNELS_ORDER	CORACQ_VAL_CHANNELS_ORDER_NORMAL

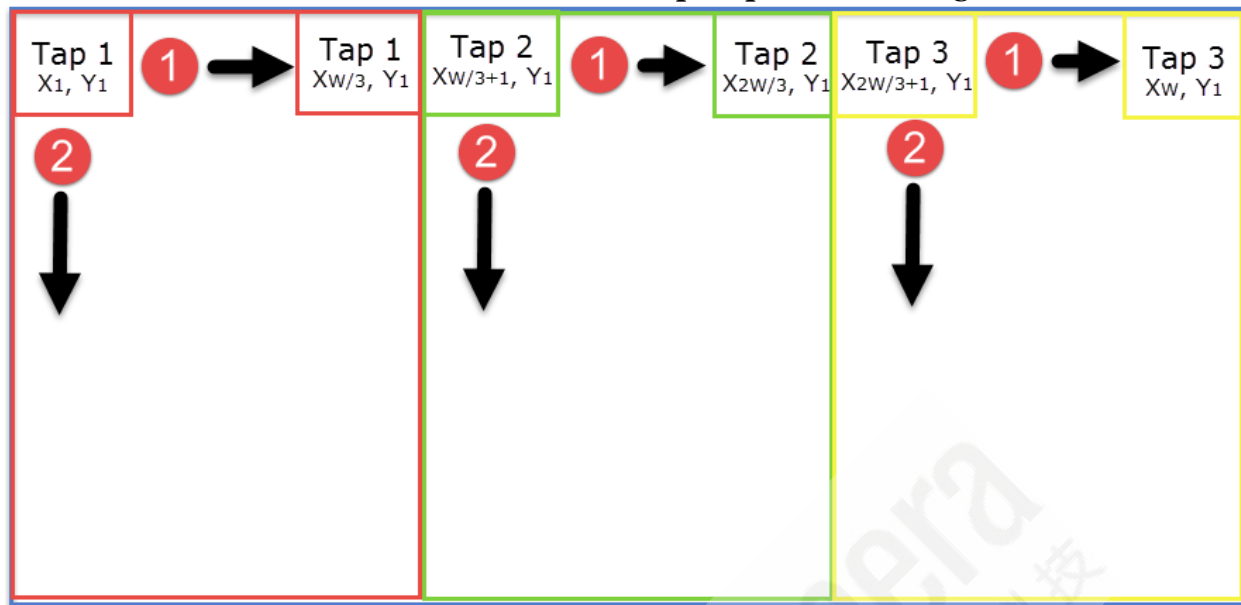
**2XE-1Y (area-scan): 2 zones in X, 1 zone in Y: 2 taps separate converge**



Parameter	Value
CORACQ_PRM_TAPS	2
CORACQ_PRM_TAP_OUTPUT	CORACQ_VAL_TAP_OUTPUT_SEGMENTED (0x00000002)
CORACQ_PRM_TAP_1_DIRECTION	The following values ORed (decimal = 21, hex = 0x15) CORACQ_VAL_TAP_DIRECTION_LR (0x00000001) CORACQ_VAL_TAP_DIRECTION_UD (0x00000004) CORACQ_VAL_TAP_DIRECTION_FROM_TOP (0x00000010)
CORACQ_PRM_TAP_2_DIRECTION	The following values ORed (decimal = 22, hex = 0x16) CORACQ_VAL_TAP_DIRECTION_RL (0x00000002) CORACQ_VAL_TAP_DIRECTION_UD (0x00000004) CORACQ_VAL_TAP_DIRECTION_FROM_TOP (0x00000010)
CORACQ_PRM_CHANNEL	1
CORACQ_PRM_CHANNELS_ORDER	CORACQ_VAL_CHANNELS_ORDER_NORMAL

## 3 Tap Geometries

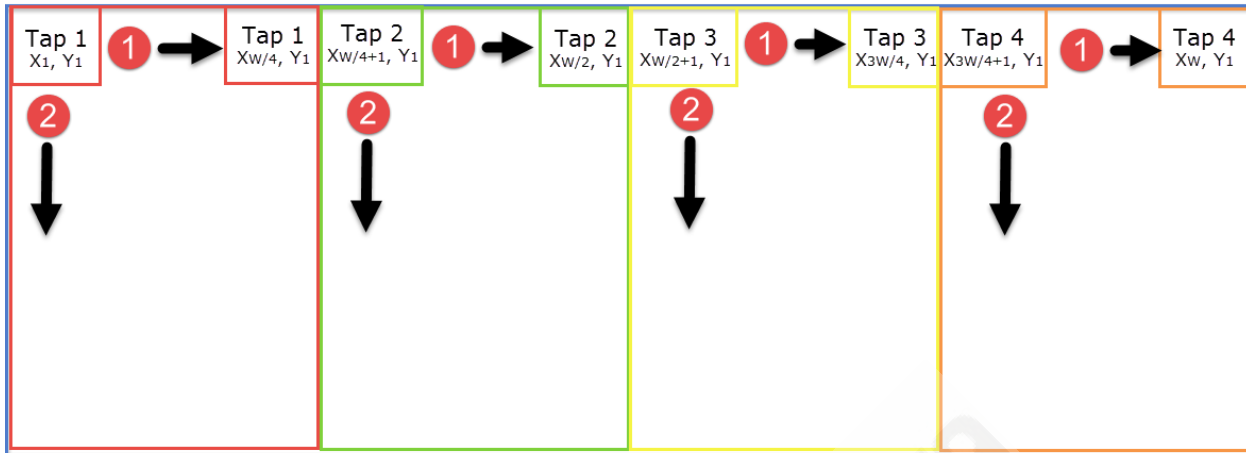
3X-1Y (area-scan): 2 zones in X, 1 zone in Y: 3 taps separate left to right



Parameter	Value
CORACQ_PRM_TAPS	3
CORACQ_PRM_TAP_OUTPUT	CORACQ_VAL_TAP_OUTPUT_SEGMENTED (0x00000002)
CORACQ_PRM_TAP_1_DIRECTION	The following values ORed (decimal = 21, hex = 0x15) CORACQ_VAL_TAP_DIRECTION_LR (0x00000001) CORACQ_VAL_TAP_DIRECTION_UD (0x00000004) CORACQ_VAL_TAP_DIRECTION_FROM_TOP (0x00000010)
CORACQ_PRM_TAP_2_DIRECTION	Same as CORACQ_PRM_TAP_1_DIRECTION
CORACQ_PRM_TAP_3_DIRECTION	Same as CORACQ_PRM_TAP_1_DIRECTION
CORACQ_PRM_CHANNEL	1
CORACQ_PRM_CHANNELS_ORDER	CORACQ_VAL_CHANNELS_ORDER_NORMAL (0x00000001)

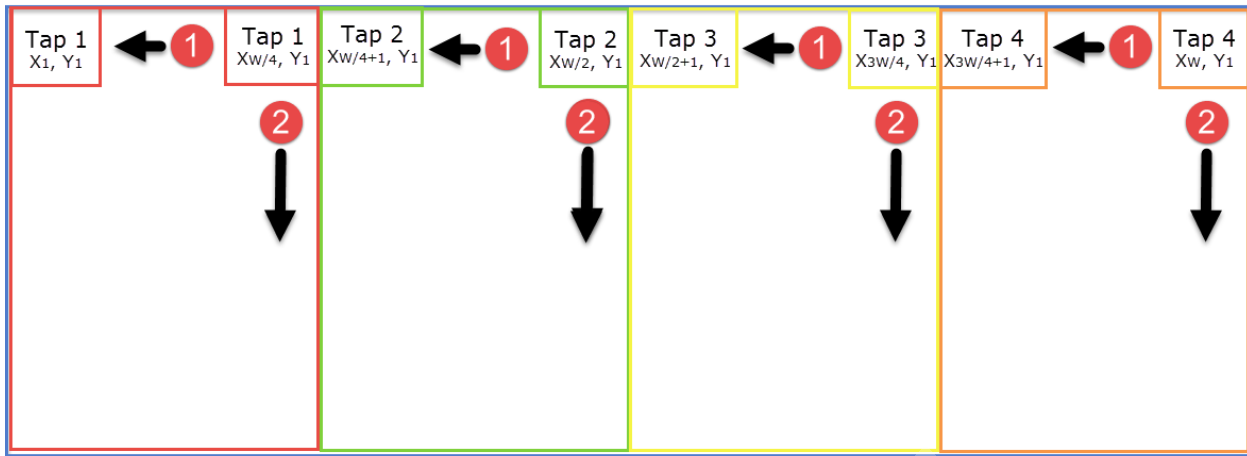
## 4 Tap Geometries

4X-1Y (area-scan): 4 zones in X, 1 zone in Y: 4 taps separate left to right



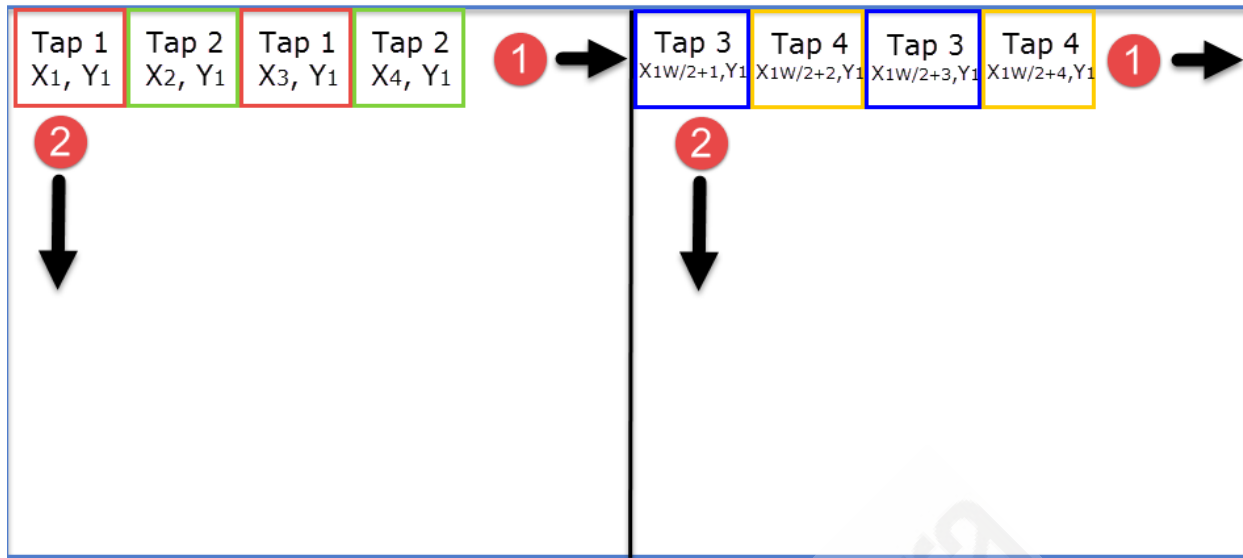
Parameter	Value
CORACQ_PRM_TAPS	4
CORACQ_PRM_TAP_OUTPUT	CORACQ_VAL_TAP_OUTPUT_SEGMENTED (0x00000002)
CORACQ_PRM_TAP_1_DIRECTION	The following values ORed (decimal = 21, hex = 0x15) CORACQ_VAL_TAP_DIRECTION_LR (0x00000001) CORACQ_VAL_TAP_DIRECTION_UD (0x00000004) CORACQ_VAL_TAP_DIRECTION_FROM_TOP (0x00000010)
CORACQ_PRM_TAP_2_DIRECTION	Same as CORACQ_PRM_TAP_1_DIRECTION
CORACQ_PRM_TAP_3_DIRECTION	Same as CORACQ_PRM_TAP_1_DIRECTION
CORACQ_PRM_TAP_4_DIRECTION	Same as CORACQ_PRM_TAP_1_DIRECTION
CORACQ_PRM_CHANNEL	1
CORACQ_PRM_CHANNELS_ORDER	CORACQ_VAL_CHANNELS_ORDER_NORMAL (0x00000001)

**4X-1Y (area-scan): 4 zones in X, 1 zone in Y: 4 taps separate right to left**



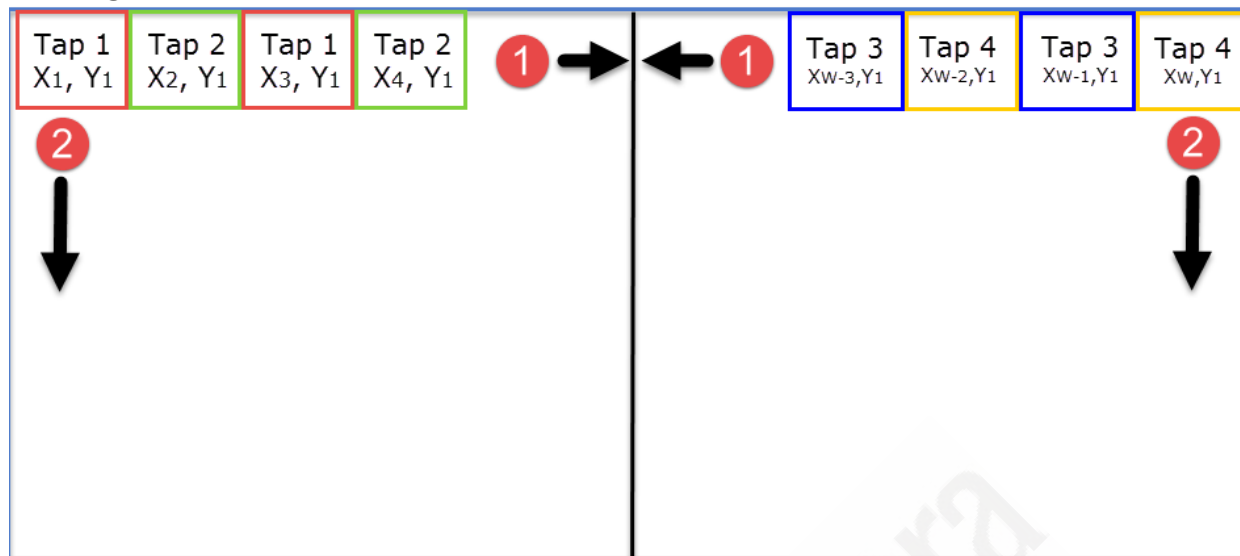
Parameter	Value
CORACQ_PRM_TAPS	4
CORACQ_PRM_TAP_OUTPUT	CORACQ_VAL_TAP_OUTPUT_SEGMENTED (0x00000002)
CORACQ_PRM_TAP_1_DIRECTION	The following values ORed (decimal = 22, hex = 0x16) CORACQ_VAL_TAP_DIRECTION_RL (0x00000002) CORACQ_VAL_TAP_DIRECTION_UD (0x00000004) CORACQ_VAL_TAP_DIRECTION_FROM_TOP (0x00000010)
CORACQ_PRM_TAP_2_DIRECTION	Same as CORACQ_PRM_TAP_1_DIRECTION
CORACQ_PRM_TAP_3_DIRECTION	Same as CORACQ_PRM_TAP_1_DIRECTION
CORACQ_PRM_TAP_4_DIRECTION	Same as CORACQ_PRM_TAP_1_DIRECTION
CORACQ_PRM_CHANNEL	1
CORACQ_PRM_CHANNELS_ORDER	CORACQ_VAL_CHANNELS_ORDER_NORMAL (0x00000001)

**2X2-1Y (area-scan): 2 zones in X with 2 taps, 1 zone in Y: 4 taps 2 segments interleaved**



Parameter	Value
CORACQ_PRM_TAPS	4
CORACQ_PRM_TAP_OUTPUT	CORACQ_VAL_TAP_OUTPUT_ALTERNATE (0x00000001)
CORACQ_PRM_TAP_1_DIRECTION	The following values ORed (decimal = 21, hex = 0x15) CORACQ_VAL_TAP_DIRECTION_LR (0x00000001) CORACQ_VAL_TAP_DIRECTION_UD (0x00000004) CORACQ_VAL_TAP_DIRECTION_FROM_TOP (0x00000010)
CORACQ_PRM_TAP_2_DIRECTION	Same as CORACQ_PRM_TAP_1_DIRECTION
CORACQ_PRM_TAP_3_DIRECTION	Same as CORACQ_PRM_TAP_1_DIRECTION
CORACQ_PRM_TAP_4_DIRECTION	Same as CORACQ_PRM_TAP_1_DIRECTION
CORACQ_PRM_CHANNEL	1
CORACQ_PRM_CHANNELS_ORDER	CORACQ_VAL_CHANNELS_ORDER_NORMAL (0x00000001)

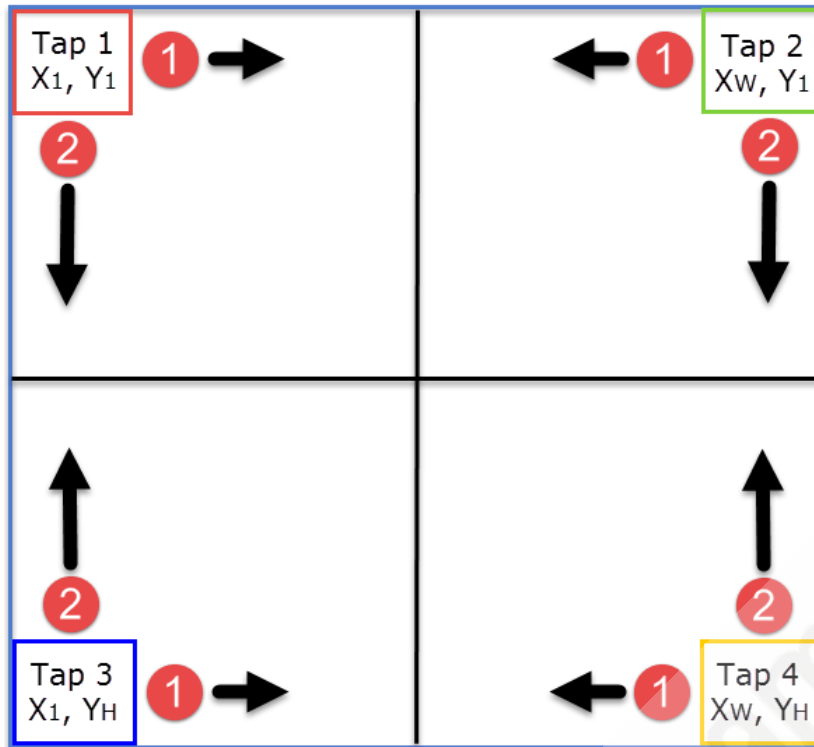
**2X2E-1Y (area-scan): 2 zones in X with 2 taps and end extraction, 1 zone in Y: 4 taps interleaved converge**



Parameter	Value
CORACQ_PRM_TAPS	4
CORACQ_PRM_TAP_OUTPUT	CORACQ_VAL_TAP_OUTPUT_ALTERNATE (0x00000001)
CORACQ_PRM_TAP_1_DIRECTION	The following values ORed (decimal = 21, hex = 0x15) CORACQ_VAL_TAP_DIRECTION_LR (0x00000001) CORACQ_VAL_TAP_DIRECTION_UD (0x00000004) CORACQ_VAL_TAP_DIRECTION_FROM_TOP (0x00000010)
CORACQ_PRM_TAP_2_DIRECTION	Same as CORACQ_PRM_TAP_1_DIRECTION
CORACQ_PRM_TAP_3_DIRECTION	The following values ORed (decimal = 22, hex = 0x16) CORACQ_VAL_TAP_DIRECTION_RL (0x00000002) CORACQ_VAL_TAP_DIRECTION_UD (0x00000004) CORACQ_VAL_TAP_DIRECTION_FROM_TOP (0x00000010)
CORACQ_PRM_TAP_4_DIRECTION	Same as CORACQ_PRM_TAP_3_DIRECTION
CORACQ_PRM_CHANNEL	1
CORACQ_PRM_CHANNELS_ORDER	CORACQ_VAL_CHANNELS_ORDER_NORMAL (0x00000001)



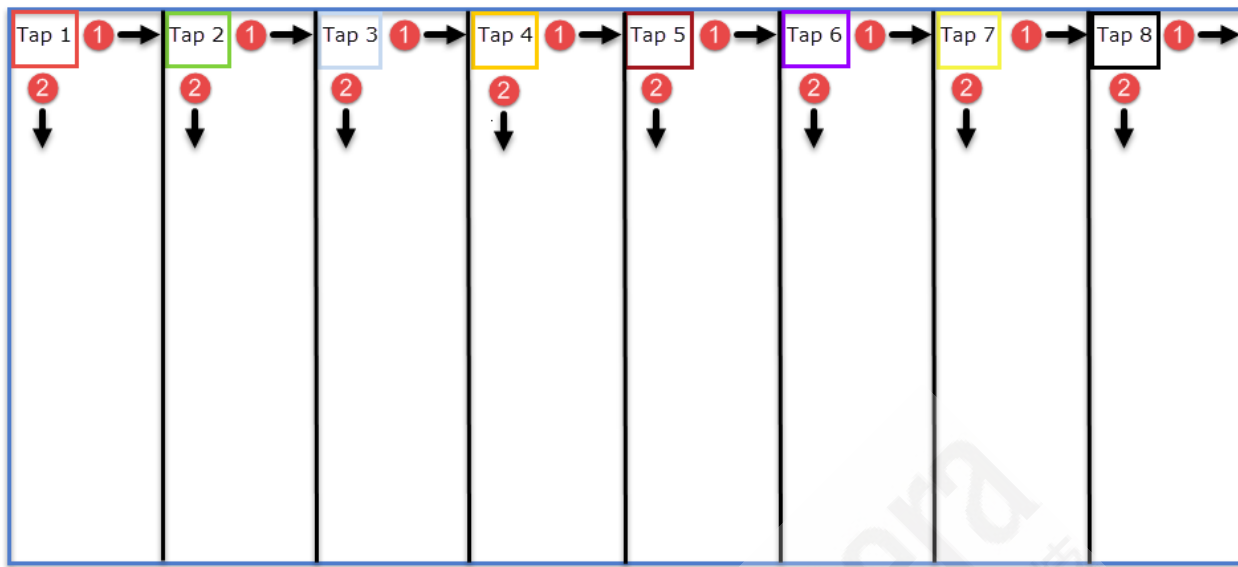
**2X2E-2YE (area-scan): 2 zones in X with 2 taps, 2 zones in Y with 2 taps and end extraction, 1 zone in Y: 4 quadrant converge**



Parameter	Value
CORACQ_PRM_TAPS	4
CORACQ_PRM_TAP_OUTPUT	CORACQ_VAL_TAP_OUTPUT_SEGMENTED (0x00000002)
CORACQ_PRM_TAP_1_DIRECTION	The following values ORed (decimal = 21, hex = 0x15) CORACQ_VAL_TAP_DIRECTION_LR (0x00000001) CORACQ_VAL_TAP_DIRECTION_UD (0x00000004) CORACQ_VAL_TAP_DIRECTION_FROM_TOP (0x00000010)
CORACQ_PRM_TAP_2_DIRECTION	The following values ORed (decimal = 22, hex = 0x16) CORACQ_VAL_TAP_DIRECTION_RL (0x00000002) CORACQ_VAL_TAP_DIRECTION_UD (0x00000004) CORACQ_VAL_TAP_DIRECTION_FROM_TOP (0x00000010)
CORACQ_PRM_TAP_3_DIRECTION	The following values ORed (decimal = 73, hex = 0x49) CORACQ_VAL_TAP_DIRECTION_LR (0x00000001) CORACQ_VAL_TAP_DIRECTION_UD (0x00000004) CORACQ_VAL_TAP_DIRECTION_FROM_BOT (0x00000040)
CORACQ_PRM_TAP_4_DIRECTION	The following values ORed (decimal = 74, hex = 0x4A) CORACQ_VAL_TAP_DIRECTION_RL (0x00000002) CORACQ_VAL_TAP_DIRECTION_UD (0x00000004) CORACQ_VAL_TAP_DIRECTION_FROM_BOT (0x00000040)
CORACQ_PRM_CHANNEL	2
CORACQ_PRM_CHANNELS_ORDER	CORACQ_VAL_CHANNELS_ORDER_NORMAL (0x00000001)

## 8 Tap Geometries

1X8-1Y (area-scan): 1 zones in X with 8 taps, 1 zone in Y: 8 taps interleaved



Parameter	Value
CORACQ_PRM_TAPS	4
CORACQ_PRM_TAP_OUTPUT	CORACQ_VAL_TAP_OUTPUT_ALTERNATE (0x00000001)
CORACQ_PRM_TAP_1_DIRECTION	The following values ORed (decimal = 21, hex = 0x15) CORACQ_VAL_TAP_DIRECTION_LR (0x00000001) CORACQ_VAL_TAP_DIRECTION_UD (0x00000004) CORACQ_VAL_TAP_DIRECTION_FROM_TOP (0x00000010)
CORACQ_PRM_TAP_2_DIRECTION	Same as CORACQ_PRM_TAP_1_DIRECTION
CORACQ_PRM_TAP_3_DIRECTION	Same as CORACQ_PRM_TAP_1_DIRECTION
CORACQ_PRM_TAP_4_DIRECTION	Same as CORACQ_PRM_TAP_1_DIRECTION
CORACQ_PRM_TAP_5_DIRECTION	Same as CORACQ_PRM_TAP_1_DIRECTION
CORACQ_PRM_TAP_6_DIRECTION	Same as CORACQ_PRM_TAP_1_DIRECTION
CORACQ_PRM_TAP_7_DIRECTION	Same as CORACQ_PRM_TAP_1_DIRECTION
CORACQ_PRM_TAP_8_DIRECTION	Same as CORACQ_PRM_TAP_1_DIRECTION
CORACQ_PRM_CHANNEL	1
CORACQ_PRM_CHANNELS_ORDER	CORACQ_VAL_CHANNELS_ORDER_NORMAL (0x00000001)

# Contact Information



---

## Sales Information

Visit our web site:

[www.teledynedalsa.com/imaging](http://www.teledynedalsa.com/imaging)

Email:

<mailto:info@teledynedalsa.com>

### Canadian Sales

Teledyne DALSA — Head office  
605 McMurray Road  
Waterloo, Ontario, Canada, N2V 2E9  
Tel: 519 886 6000  
Fax: 519 886 8023

Teledyne DALSA — Montreal office  
880 Rue McCaffrey  
St. Laurent, Quebec, Canada, H4T 2C7  
Tel: (514) 333-1301  
Fax: (514) 333-1388

### USA Sales

Teledyne DALSA — Billerica office  
700 Technology Park Drive  
Billerica, Ma. 01821  
Tel: (978) 670-2000  
Fax: (978) 670-2010

### European Sales

Teledyne DALSA GMBH  
Felix-Wankel-Str. 1  
82152 Krailling, Germany  
Tel: +49 – 89 89 – 54 57 3-80  
Fax: +49 – 89 89 – 54 57 3-46

### Asian Sales

Teledyne DALSA Asia Pacific  
Ikebukuro East 13F  
3-4-3 Higashi Ikebukuro,  
Toshima-ku, Tokyo, Japan  
Tel: +81 3 5960 6353  
Fax: +81 3 5960 6354

Shanghai Industrial Investment Building  
Room G, 20F, 18 North Cao Xi Road,  
Shanghai, China 200030  
Tel: +86-21-64279081  
Fax: +86-21-64699430

---

## Technical Support

Submit any support question or request via our web site:

Technical support form via our web page:  
Support requests for imaging product installations,  
Support requests for imaging applications

<http://www.teledynedalsa.com/imaging/support>

Camera support information

Product literature and driver updates