

Coaxlink

Coaxlink 11.2.1



PCI 
EXPRESSTM

 **PC/104**
Embedded PC Modules

CoaxPress

使用条款

EURESYS s.a. 应保留硬件和软件文档以及 EURESYS s.a. 商标的所有财产权、所有权和利益。

文档中提及的所有公司和产品的名称可能是其各自所有者的商标。

未经事先通知，不得对本书中包含的 EURESYS s.a 的硬件或软件、品牌或文档进行许可、使用、出租、租赁、翻译、复制、复印或修改。

EURESYS s.a. 可能随时自行修改产品规格或更改本文档中给出的信息，恕不另行通知。

EURESYS s.a. 对于使用其硬件或软件而引起的任何类型的收入、利润、商誉、数据、信息系统损失或损害，或与使用其硬件或软件相关的，或因本文档遗漏或错误造成的其他特殊的、偶然的、间接的、后果性的或惩罚性的损害赔偿，概不负责。

本文档随 Coaxlink11.2.1(doc build) 提供。
2088© 2019 EURESYS s.a.

目录

1. 关于本文档	7
1.1. 文件范围	7
1.2. 文档更改	7
2. Coaxlink体系结构	8
2.1. 主要元件	9
2.2. 框图	11
3. CoaXPress主机接口	15
3.1. 主机接口规范	16
3.2. Firmware Variants per Product	20
3.3. 主机连接映射	25
3.4. 链路配置	32
3.5. CoaXPress供电	33
3.6. CoaXPress I/O 通道	37
3.7. CoaXPress主机到设备触发	38
3.8. CoaXPress LED	47
3.9. 连接测试	48
3.10. CoaXPress链接验证工具	49
4. 图像数据路径	55
4.1. FIFO 缓冲器	56
4.2. 采集门	57
4.3. 像素数据处理	58
4.4. 像素数据处理配置	60
4.5. 像素分量解包	63
4.6. 像素分量重新排序	65
4.7. 端序转换	66
4.8. 像素排序	67
4.9. 图像数据传输	68

5. 相机控制原理	75
5.1. 相机循环	76
5.2. 相机循环串联规则	77
5.3. 相机控制方法	79
6. 照明控制原理	81
6.1. 照明装置	82
6.2. 对齐相机和照明循环	83
7. 相机和照明控制器	84
7.1. CIC框图	85
7.2. 循环定时机	86
7.3. 循环管理器	88
7.4. 循环触发管理器	89
7.5. 序列管理器	91
7.6. CIC输出信号路由	93
7.7. CIC时序图	94
循环时序图	95
连续重叠CIC循环	97
循环顺序时序图	102
8. 通用 I/O	104
8.1. I/O线路概述	105
8.2. I/O线路概述	108
8.3. I/O控制块	109
8.4. 线格式和线模式控制	111
8.5. 线路极性控制	115
8.6. 滤波器控制	116
8.7. 线源选择	117
8.8. 逻辑I/O线路状态	118
8.9. 实际I/O线路状态	119
8.10. 线路驱动器实际输出状态	120
8.11. 初始状态	121
9. I/O 工具箱	122
9.1. 介绍I/O工具箱	123
9.2. I/O工具箱组合	126
9.3. 行输入工具	129

9.4. 正交解码器工具	130
9.5. 事件输入工具	133
9.6. 分割器工具	134
9.7. 乘法器/分割器工具	135
9.8. 延迟工具	138
9.9. 用户操作工具和用户输出寄存器	140
10. 事件信令和计数	144
10.1. 简介	145
10.2. 自定义事件源	148
10.3. 特定于事件的上下文数据	151
10.4. 关于GenTL 信令	153
11. 高级功能	154
11.1. 子链路采集	155
11.2. 多流采集	156
11.3. CoaXPress数据转发	157
数据转发原则	158
数据转发连接方案	160
行扫描触发同步	161
配置脚本示例	162
11.4. 平场校正	163
什么是平场校正?	164
Coaxlink FFC	167
FFC向导示例程序	170
11.5. 查找表处理	173
LUT处理简介	174
LUT处理可用性	175
单色查找表格处理	176
LUT内容定义	177
LUT设置程序	184
11.6. Bayer CFA解码	187
Bayer CFA到RGB转换	188
使用Bayer CFA解码器	192
11.7. JPEG 编码	194
11.8. 激光线提取	196
简介	197
激光线提取算法	198
LLE处理核心实现	199
LLE处理核心特性	200
双激光线提取	202
线性滤波器	203
取芯门槛	204

最大检测	205
峰值检测	206
重心	207
用例示例	208
目标	209
定义输入ROI	210
设置过滤和阈值	211
定义LLE算法	212
定义扫描长度和缓冲区大小	214
附录	215
11.9. 行扫描采集	216
线扫描采集原理	217
线扫描采集用例	221
11.10. C2C -链路	224
C2C链接描述	225
C2C链路规范	226
触发传播延迟	228
CIC同步	230
C2C链路设置步骤	233
11.11. OEM 安全钥	235
介绍OEM安全钥	235
使用OEMSafetyKey	236

1. 关于本文档

1.1. 文件范围

7

1.2. 文档更改

7

1.1. 文件范围

本文档描述并解释了当Coaxlink产品与Coaxlink驱动程序版本一起运行时，如何使用Coaxlink产品的功能11.2.1

除非特别说明，否则本文档中描述的功能适用于Coaxlink驱动程序支持的所有Coaxlink产品及其固件变体。

1.2. 文档更改

Coaxlink 11.2

修订了以下主题：

- "C2C -链路" 于页面224
- "CIC同步" 于页面230
- "C2C链路设置步骤" 于页面233

2. Coaxlink体系结构

2.1. 主要元件
2.2. 框图

9
11

2.1. 主要元件

基于Coaxlink图像采集系统的主要功能元件概述。

GenTL层次

Coaxlink的每个功能元件都由属于GenTL模块的GenICam功能配置和控制。

在层次结构的顶部，每台主机PC都有一个**GenTL系统模块**。它绑定了主机PC的所有**GenTL接口模块**。

每个Coaxlink卡都有一个**GenTL接口模块**。它绑定了Coaxlink卡的所有**GenTL设备模块**。

连接到Coaxlink卡的每个摄像机(或成像设备)都有一个**GenTL设备模块**。属于成像设备(摄像机)本身的元件称为**远程设备**。相反，属于帧捕获器的元件也称为**本地设备**。



备注 可连接到Coaxlink卡的最大摄像机数量由安装的固件变体决定。

附在Coaxlink卡上的摄像头所传送的每个数据流都有一个**GenTL数据流模块**。它收集了图像构建中所涉及的元件，并从成像设备传输到GenTL缓冲池。



备注 连接到Coaxlink卡的摄像机的最大数据流数量由安装的固件变体决定。

每个图像缓冲区有一个**GenTL缓冲模块**。

接口模块主要元件(橙色)

I/O线

该 "通用I/O" 于页面104 块收集卡的所有I/O端口。

I/O 工具箱

该 "I/O 工具箱" 于页面122 块收集(从连接到I/O端口输入的触发器和编码器设备)用于构建事件流的工具集合。



备注 这些元件是由Coaxlink卡管理的所有GenTL设备模块共有的(或者可以被共享)

设备模块主元件(绿色)

摄像机和照明控制器

该模块用于控制相机周期和照明频闪。它可以配置为从任何I/O工具箱输出流接收实时(摄像机)周期触发事件。它产生两个实时信号:发送到摄像机触发输入端的“摄像机触发”信号;和发送到与摄像机相关联的照明设备的“频闪”信号。



备注 这个元件是与该成像设备相关的所有GenTL数据流模块共有的(或者可以由该模块共享)。

数据流模块主要元件

图像采集控制器

该模块用于控制采集门。它可以配置为从任何I/O工具箱输出流接收实时“扫描开始”和“扫描结束”的触发事件。

采集门

该“采集门”于[页面57](#)控制数据提取,并过滤掉不需要采集的图像数据。

FIFO 缓冲器

该车载设备“FIFO 缓冲器”于[页面56](#)临时存储原始图像数据以及相关元数据,例如图像大小、像素类型、时间戳...

像素处理

该“像素数据处理”于[页面58](#)执行动态像素处理。

数据格式编排

“像素分量解包”于[页面63](#), “像素分量重新排序”于[页面65](#), “端序转换”于[页面66](#) 并且 “像素排序”于[页面67](#) 根据期望的像素输出格式配置操作。

图像数据传输

该“图像数据传输”于[页面68](#)是将图像数据传输到目标缓冲区。

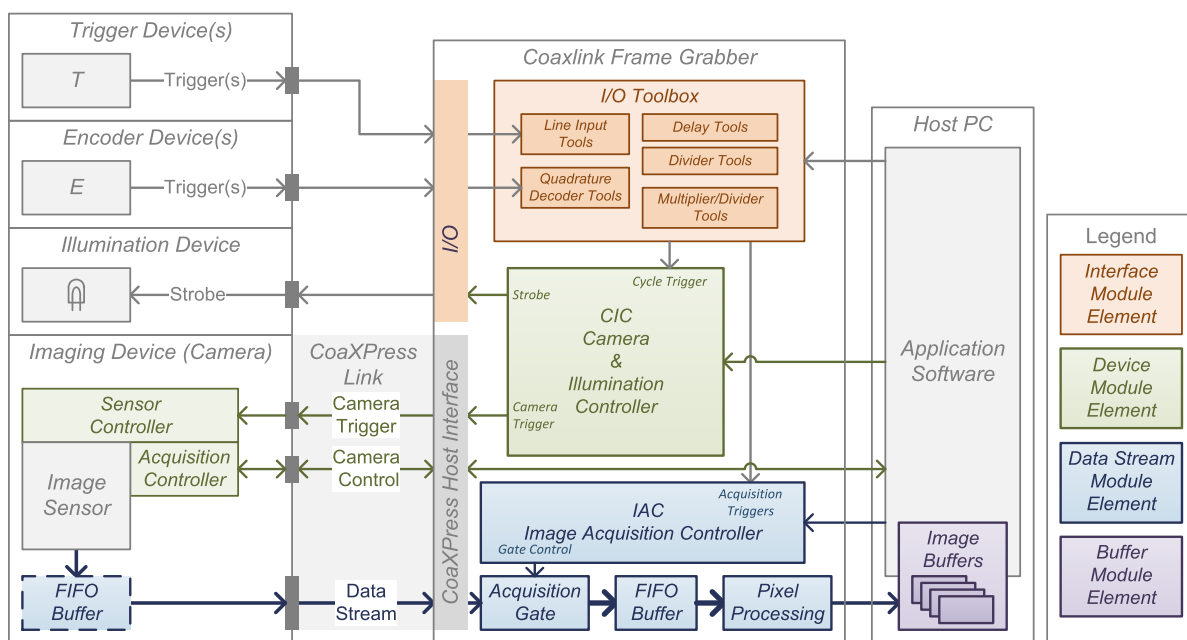
2.2. 框图

本节展示了使用一个Coaxlink卡的三个图像采集系统的框图



备注 在下面的图中，主要元件由矩形表示，它们的关系由线段表示，箭头表示信号或数据流的方向。矩形的填充颜色表示GenTL层次结构中的级别，如图例中所述。

1-camera, 1-data-stream

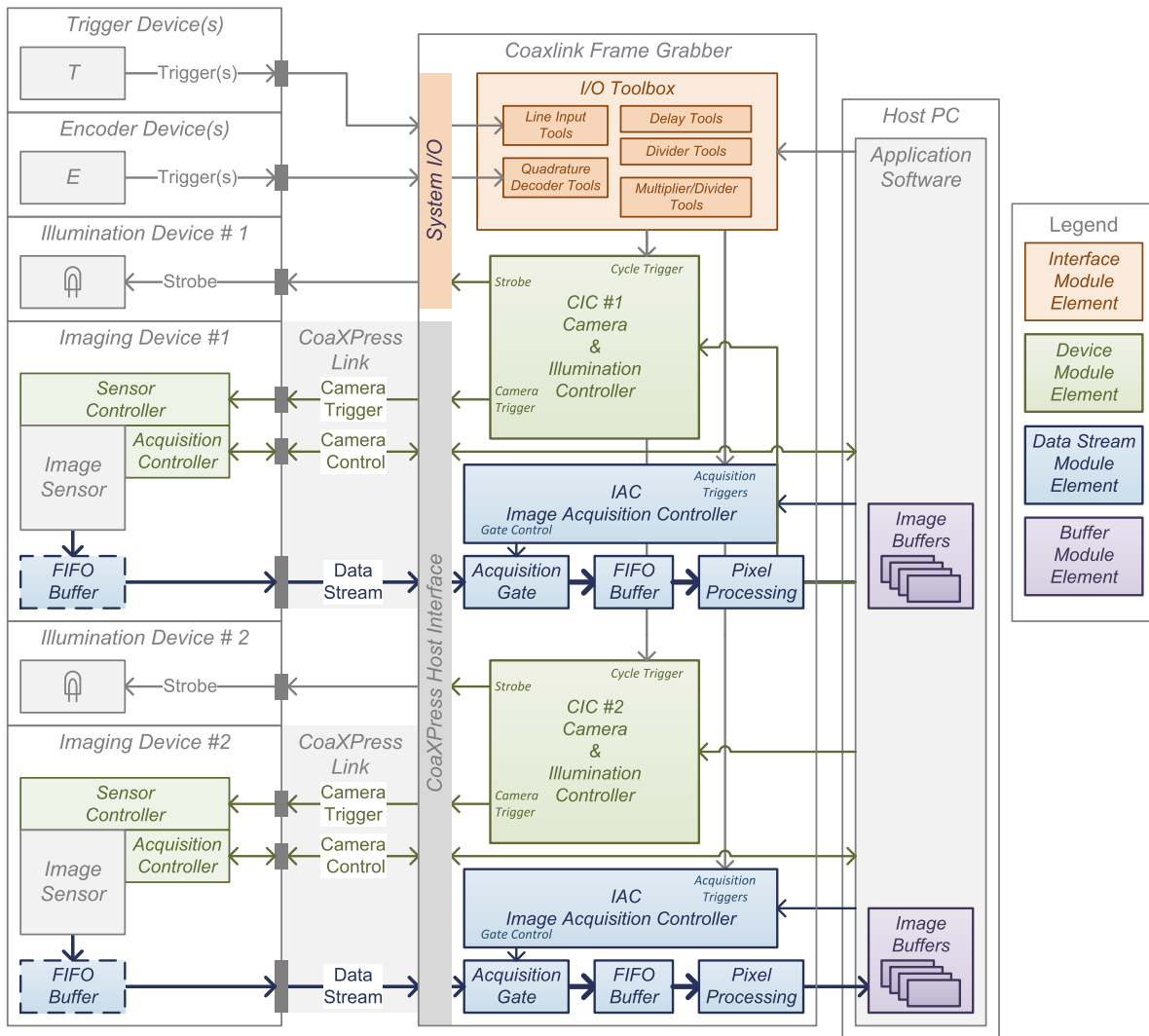


1-camera, 1-data-stream Coaxlink图像采集系统



备注 此配置仅适用于安装了一个 **1-camera** 或一个 **1-camera, line-scan** 固件变体的情况。

2-camera, 1-data-stream

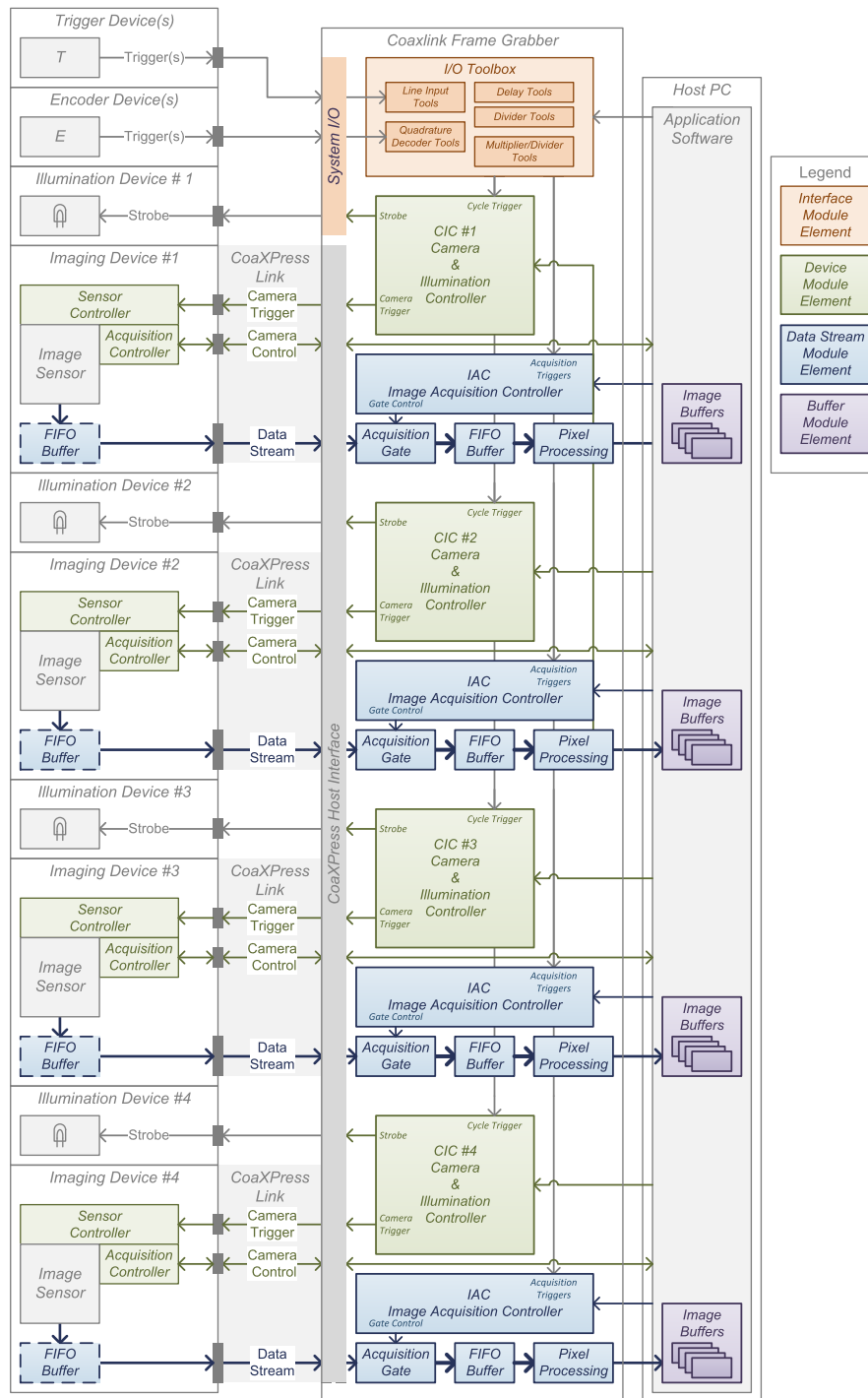


2-camera, 1-data-stream Coaxlink图像采集系统



备注 此配置仅适用于安装了一个 2-camera 或一个 2-camera, line-scan 固件变体的情况。

4-camera, 1-data-stream

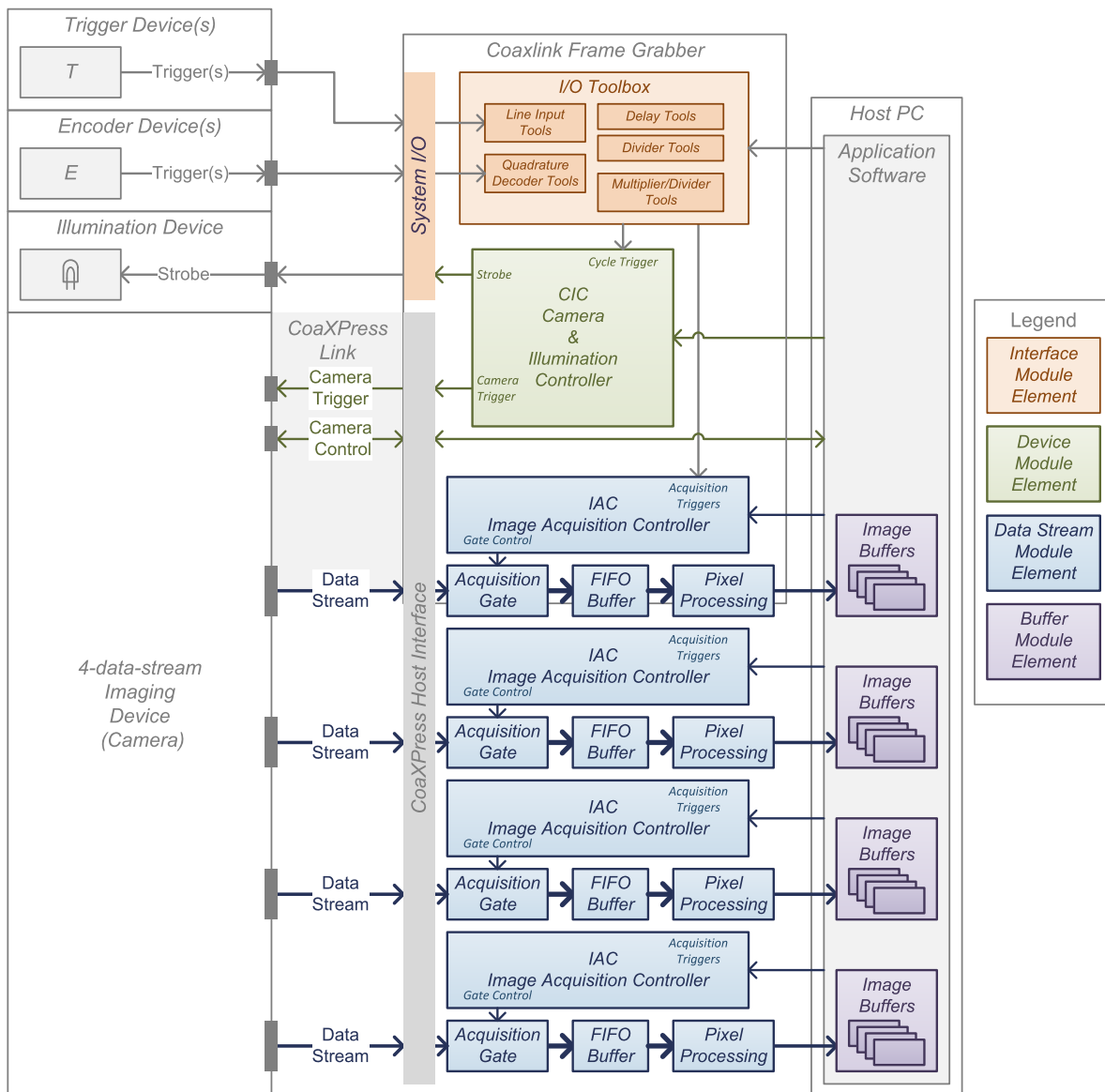


4-camera, 1-data-stream Coaxlink图像采集系统



备注 此配置仅适用于安装了一个 4-camera 或一个 4-camera, line-scan 固件变体的情况。

1-camera, 4-data-stream



1-camera, 4-data-stream Coaxlink图像采集系统



备注 此配置仅适用于安装了一个 **1-camera, 4-data-stream** 固件变体的情况

3. CoaXPress主机接口

3.1. 主机接口规范	16
3.2. Firmware Variants per Product	20
3.3. 主机连接映射	25
3.4. 链路配置	32
3.5. CoaXPress供电	33
3.6. CoaXPress I/O 通道	37
3.7. CoaXPress主机到设备触发	38
3.8. CoaXPress LED	47
3.9. 连接测试	48
3.10. CoaXPress链接验证工具	49

3.1. 主机接口规范

接口(卡)规范

具有一个**GenTL接口模块**范围的规范(即:Coaxlink 卡)

CoaXPress 功能栏	适用产品
CXP-6 DIN 1	Mono
CXP-6 DIN 2	Duo Duo104EMB Duo104MIL
CXP-6 DIN 4	Quad QuadG3 QuadG3LH QuadG3DF
CXP-6 DIN 8	Octo
CXP-12 μ BNC 4	QuadCXP12 QuadCXP12J

最大聚合输入数据速率	适用产品
625,000 Bytes/s	Mono
1,250,000 Bytes/s	Duo Duo104EMB Duo104MIL
2,500,000 Bytes/s	Quad QuadG3 QuadG3LH QuadG3DF
5,000,000 Bytes/s	Octo QuadCXP12 QuadCXP12J

设备计数	适用产品
1 或2 或4 或5 或8	另见: "Firmware Variants per Product" 于页面20

另见: 有关电气规格, 请参阅硬件手册中的 [CoaXPress Host Interface](#) 。

设备(相机)规格

具有一个 **Gentl设备模块** (即连接到Coaxlink卡上的相机) 的规范

设备类型	适用的固件变体
区域扫描相机, 1个数据流或 区域扫描相机, 最多4个数据流 或 行扫描相机, 1个数据流	另见: "Firmware Variants per Product" 于页面20

上表不包括在"CoaXPress数据转发"于页面157中使用的特殊摄像机类型和"子链路采集"于页面155配置。

连接计数(每个相机)	适用产品/固件变体
1 或2 或4 或8	另见: "Firmware Variants per Product" 于页面20
8	另见: "子链路采集" 于页面155

数据流规范

具有一个Gentl数据流模块范围的规范

最大流包大小	适用产品
16,384 Bytes	所有产品
图像流格式	适用的固件变体
矩形图像 (YSize > 0)	Area-scan "Firmware Variants per Product" 于页面20
矩形图像 (YSize = 0)	Line-scan "Firmware Variants per Product" 于页面20
任意图像	不支持
图像扫描法	适用的固件变体
进行	所有固件变体
隔行	不支持
分接头几何	适用的固件变体
1X_1Y	所有固件变体
1X_1Y2, 1X_2YE, 1X_2YM	所有 area-scan 固件变体
像素格式	适用的固件变体
Raw	所有固件变体
Mono8, Mono10, Mono12, Mono14, Mono16	所有固件变体
BayerGR8, BayerRG8, BayerGB8, BayerBG8 BayerGR10, BayerRG10, BayerGB10, BayerBG10 BayerGR12, BayerRG12, BayerGB12, BayerBG12 BayerGR14, BayerRG14, BayerGB14, BayerBG14 BayerGR16, BayerRG16, BayerGB16, BayerBG16	所有固件变体
RGB8, RGB10, RGB12, RGB14, RGB16	所有固件变体
RGBA8, RGBA10, RGBA12, RGBA14, RGBA16	所有固件变体

宽度增量步进



警告 相机的图像宽度设置必须是指定宽度增量步长值的倍数：

适用于: Mono Duo Quad Duo104EMB Duo104MIL Quad3DLLE

位深度	宽度增量步进[像素]
8-位	8
10-, 12-, 14- 和 16-位	4

适用于: QuadG3 QuadG3LH QuadG3DF

位深度	宽度增量步进[像素]	
	Bayer CFA解码器已禁用	Bayer CFA解码器已启用
8-位	4	16
10-, 12-, 14- 和 16-位	2	8

适用于: Octo QuadCXP12

位深度	宽度增量步进[像素]	
	Bayer CFA解码器已禁用	Bayer CFA解码器已启用
8-位	4	32
10-, 12-, 14- 和 16-位	2	16

适用于: QuadCXP12J

位深度	宽度增量步进[像素]
8-位	32

3.2. Firmware Variants per Product

1630 Coaxlink Mono

固件变体	描述	主机连接映射	高级处理
1-camera	一个1-连接区域扫描相机	1D1	LUT

1631 Coaxlink Duo

固件变体	描述	主机连接映射	高级处理
1-camera	一个1-或2-连接区域扫描相机	1D2	LUT
1-camera, line-scan	一个1-或2-连接线扫描相机	1D2	LUT
2-camera	一个或两个1-连接区域扫描相机	2D11	LUT
2-camera, line-scan	一个或两个1-连接线扫描相机	2D11	LUT

1632 Coaxlink Quad

固件变体	描述	主机连接映射	高级处理
1-camera	一个1-或2-或4-连接区域扫描相机	1D4	LUT
1-camera, line-scan	一个1-或2-或4-连接线扫描相机	1D4	LUT
2-camera	一个或两个1-或2-连接区域扫描相机	2D22	LUT

1633 Coaxlink Quad G3

固件变体	描述	主机连接映射	高级处理
1-camera	一个1-或2-或4-连接区域扫描相机	1D4	FFC LUT CFA
1-camera, 4-data-stream	一个1-或2-或4-连接区域扫描相机, 最多4个数据流	1D4S4	-
1-camera, line-scan	一个1-或2-或4-连接线扫描相机	1D4	FFC LUT
1-slm-camera	一个8-8-连接区域扫描相机的主4-连接子链接	1D8SLM4	LUT
1-sls-camera	一个8-8-连接区域扫描相机的从4-连接子链路	1D8SLS4	LUT
2-camera	一个或两个1-或2-连接区域扫描相机	2D22	LUT
2-camera, line-scan	一个或两个1-或2-连接线扫描相机	2D22	LUT
3-camera	一个1-或2-连接和一个或两个1-连接区域扫描相机	3D211	LUT
4-camera	一个或两个或三个或四个1-连接区域扫描相机	4D1111	LUT
4-camera, line-scan	一个或两个或三个或四个1-连接线扫描相机	4D1111	LUT

1633-LH Coaxlink Quad G3 LH

固件变体	描述	主机连接映射	高级处理
1-camera	一个1-或2-或4-连接区域扫描相机	1D4	FFC LUT CFA
1-camera, 4-data-stream	一个1-或2-或4-连接区域扫描相机，最多4个数据流	1D4S4	-
1-camera, line-scan	一个1-或2-或4-连接线扫描相机	1D4	FFC LUT
1-slm-camera	一个8-8-连接区域扫描相机的主4-连接子链接	1D8SLM4	LUT
1-sls-camera	一个8-8-连接区域扫描相机的从4-连接子链路	1D8SLS4	LUT
2-camera	一个或两个1-或2-连接区域扫描相机	2D22	LUT
2-camera, line-scan	一个或两个1-或2-连接线扫描相机	2D22	LUT
3-camera	一个1-或2-连接和一个或两个1-连接区域扫描相机	3D211	LUT
4-camera	一个或两个或三个或四个1-连接区域扫描相机	4D1111	LUT
4-camera, line-scan	一个或两个或三个或四个1-连接线扫描相机	4D1111	LUT

1629 Coaxlink Duo PCIe/104-EMB

固件变体	描述	主机连接映射	高级处理
1-camera	一个1-或2-连接区域扫描相机	1D2	LUT
1-camera, line-scan	一个1-或2-连接线扫描相机	1D2	LUT
2-camera	一个或两个1-连接区域扫描相机	2D11	LUT

1634 Coaxlink Duo PCIe/104-MIL

固件变体	描述	主机连接映射	高级处理
1-camera	一个1-或2-连接区域扫描相机	1D2	LUT
1-camera, line-scan	一个1-或2-连接线扫描相机	1D2	LUT
2-camera	一个或两个1-连接区域扫描相机	2D11	LUT

1635 Coaxlink Quad G3 DF

固件变体	描述	主机连接映射	高级处理
1-camera	一个1-或2-或4-连接区域扫描相机	1D4	LUT CFA
1-df-camera	一个1-或2-或4-连接区域扫描数据转发相机	1DF4	LUT CFA
1-camera, line-scan	一个1-或2-或4-连接线扫描相机	1D4	FFC LUT
1-df-camera, line-scan	一个1-或2-或4-连接线扫描数据转发相机	1DF4	FFC LUT

1637 Coaxlink Quad 3D-LLE

固件变体	描述	主机连接映射	高级处理
1-camera	一个1-或2-或4-连接区域扫描相机	1D4	LLE

3602 Coaxlink Octo

固件变体	描述	主机连接映射	高级处理
1-camera	一个1-或2-或4-或8-连接区域扫描相机	1D8	LUT CFA
1-camera, line-scan	一个1-或2-或4-或8-连接线扫描相机	1D8	LUT
2-camera	一个或两个1-或2-或4-连接区域扫描相机	2D44	FFC LUT CFA
2-camera, line-scan	一个或两个1-或2-或4-连接线扫描相机	2D44	LUT
4-camera	一个或两个或三个或四个1-或2-连接区域扫描相机	4D2222	LUT
4-camera, line-scan	一个或两个或三个或四个1-或2-连接线扫描相机	4D2222	LUT
5-camera	一个1-或2-或4-连接和一个或两个或三个或四个1-连接区域扫描相机	5D41111	LUT
8-camera	最多8个1-连接区域扫描相机	8D11111111	LUT

3603 Coaxlink Quad CXP-12

固件变体	描述	主机连接映射	高级处理
1-camera	一个1-或2-或4-连接区域扫描相机	1D4	LUT CFA
1-camera, line-scan	一个1-或2-或4-连接线扫描相机	1D4	LUT
4-camera	一个或两个或三个或四个1-连接区域扫描相机	4D1111	LUT

3620 Coaxlink Quad CXP-12 JPEG

固件变体	描述	主机连接映射	高级处理
4-camera	一个或两个或三个或四个1-连接区域扫描相机	4D1111	CFA JPEG

3.3. 主机连接映射

CoaXPress标准建议设备(相机或数据转发设备)可以使用自由连接方案连接到主机(帧捕获器)。相反, Coaxlink的主机接口, 需要将设备连接指定给主机连接器。这种分配称为**主机连接映射**。

在产品固件变体中, 主机连接映射是硬编码的。



警告 必须根据所需映射选择Coaxlink产品和固件变体!

主机连接映射命名约定

主机连接映射或**HCMAP**指定如何将Coaxlink卡的主机接口的连接分配给设备(相机)。

主机连接映射-HCMAP-由使用以下Euresys专有命名约定的首字母缩略词指定:

```
<dev#><dev-type> [<str#>S] {<con#>...<con#>} [<SL-con#>]
```

其中:

- <dev#>声明可以添加到主机接口的设备(相机)的最大数量。
 - 1用于单设备主机接口
 - 2用于2-设备主机接口
 - ...
- <dev-type> 声明设备类型。
 - D用于标准CoaXPress设备
 - DF用于数据转发方案中使用的虚拟设备
- <con#> 声明每个设备可用的连接数。此字段对每个设备重复一次。
 - 1个用于单个连接设备
 - 2个用于2-连接装置
 - ...
- <str#>S声明设备允许的最大数据流数量。
 - 当只有1个流时, 省略此字段
 - 每个设备最多4个数据流的4S
- SL<-con#> 声明每个子链路的连接数
 - 如果没有子链路, 则省略此字段。
 - SL4用于4-连接子链路

实例

HCMAP2D22指定一个主机接口，该接口具有2个标准的1-data-stream CoaXPress设备及每个设备有2个连接。

HCMAP1D4S4指定一个主机接口，该主机接口具有1个标准CoaXPress设备、最多4个数据流和每个设备4个连接。

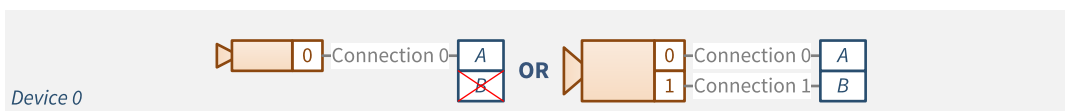
1D1主机连接映射

一个1-连接设备



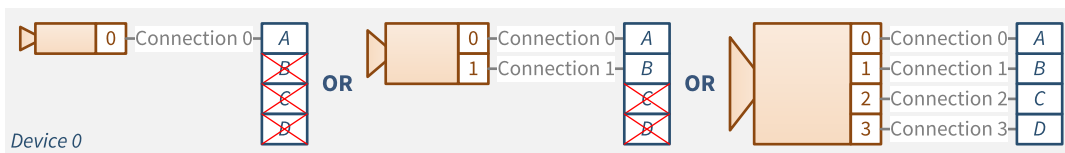
1D2主机连接映射

一个1-或2-连接设备



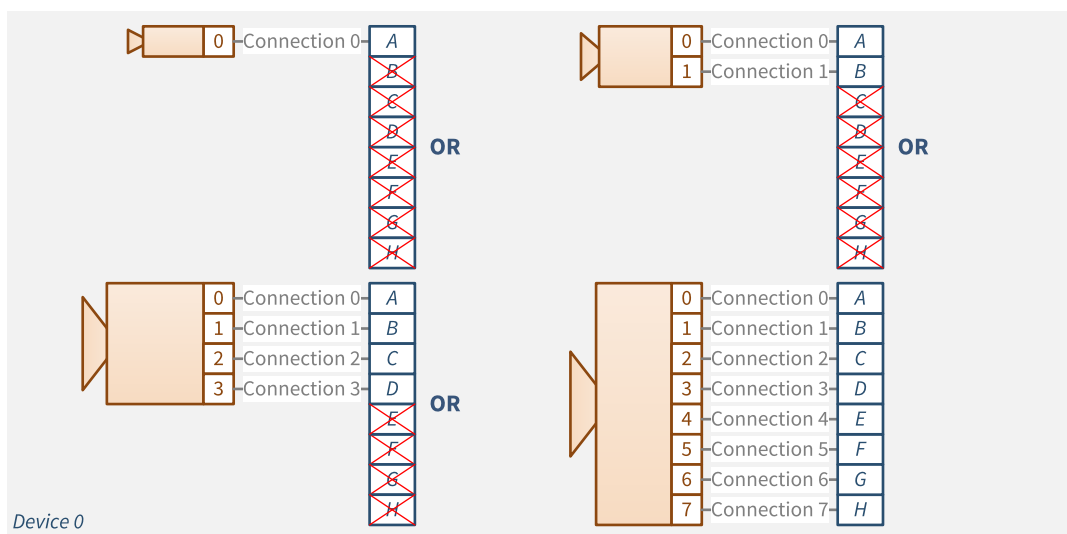
1D4主机连接映射

一个1-或2-或4-连接设备



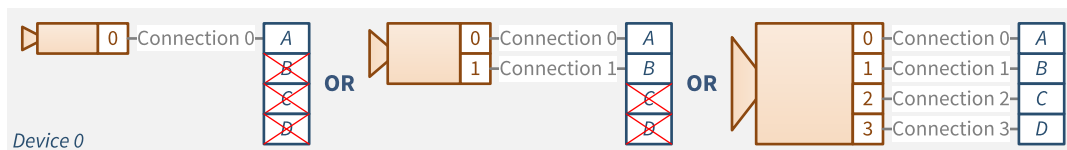
1D8主机连接映射

一个1-或2-或4-或8-连接设备



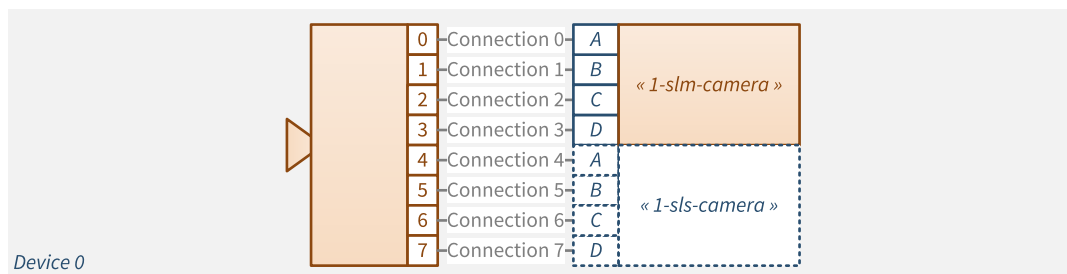
1D4S4主机连接映射

一个1-或2-或4-连接设备，最多4个数据流



1D8SLM4主机连接图

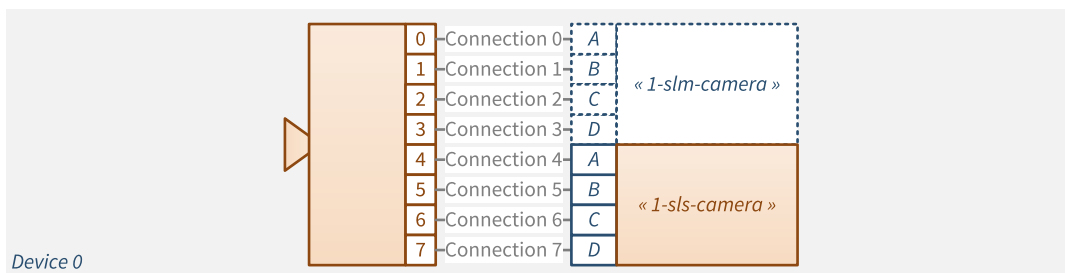
一个8-连接设备的主4-连接子链路



另见：请参阅，8-连接相机与2个Coaxlink卡的连接方案"子链路采集"于页面155。

1D8SLS4主机连接映射

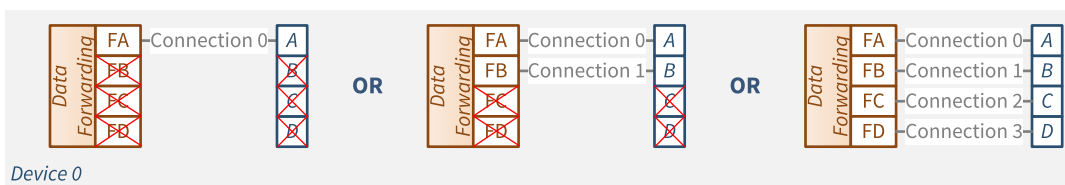
一个8-连接设备的从4-连接子链路



另见：请参阅，8-连接相机与2个Coaxlink卡的连接方案"子链路采集"于页面155。

1F4主机连接映射

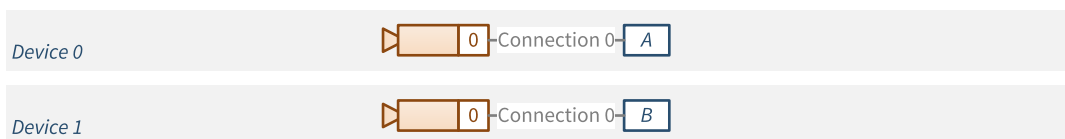
一个1-或2-或4-连接设备



另见：从属数据转发设备的连接方案，请参阅"CoaXPress数据转发"于页面157。

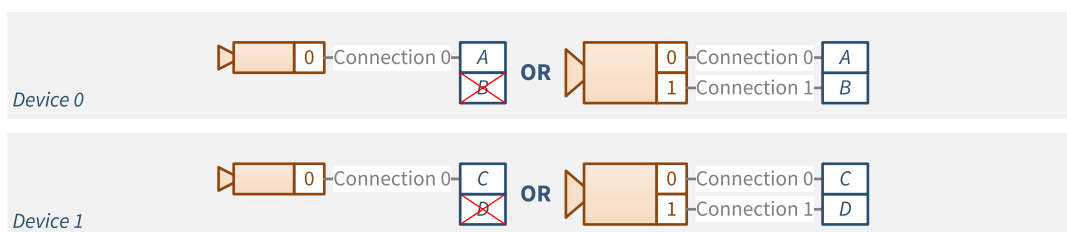
2D11主机连接图

一个或两个1-连接设备



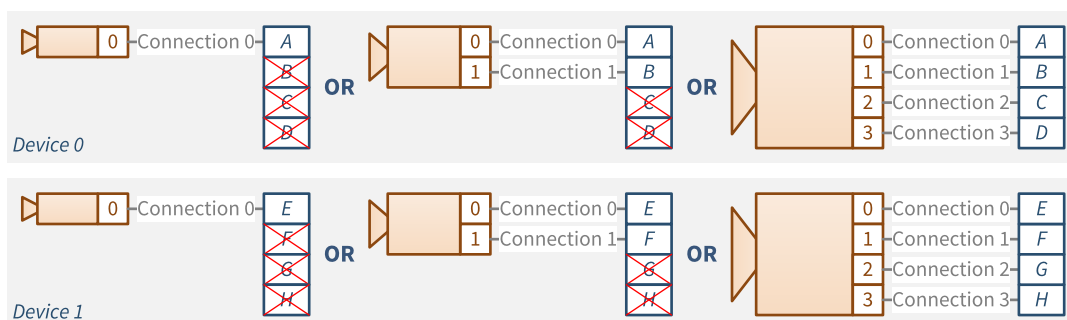
2D22主机连接图

一个或两个1-或2-连接设备



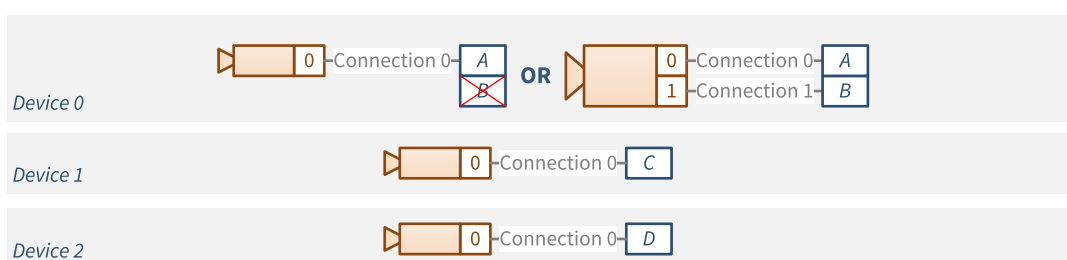
2D44主机连接图

一个或两个1-或2-或4-连接设备



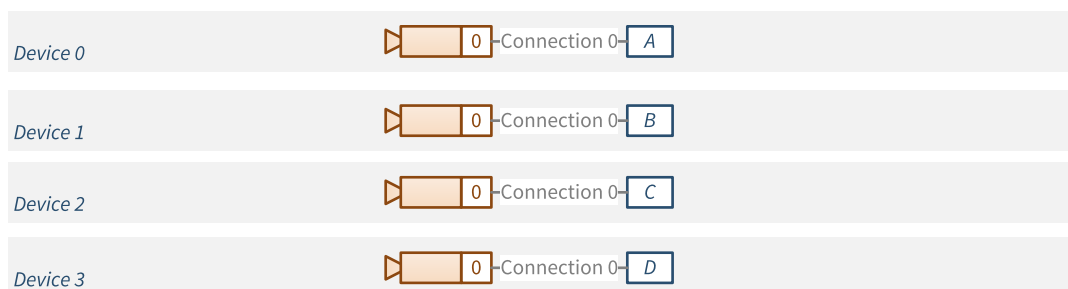
3D211 主机连接图

一个1-或2-连接和一个或两个1-连接设备



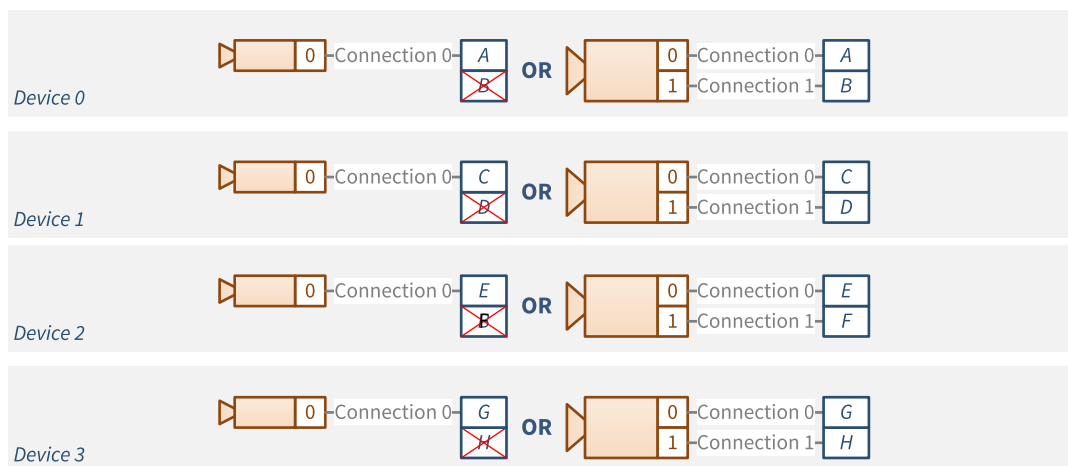
4D1111主机连接映射

一个或两个或三个或四个1-连接设备



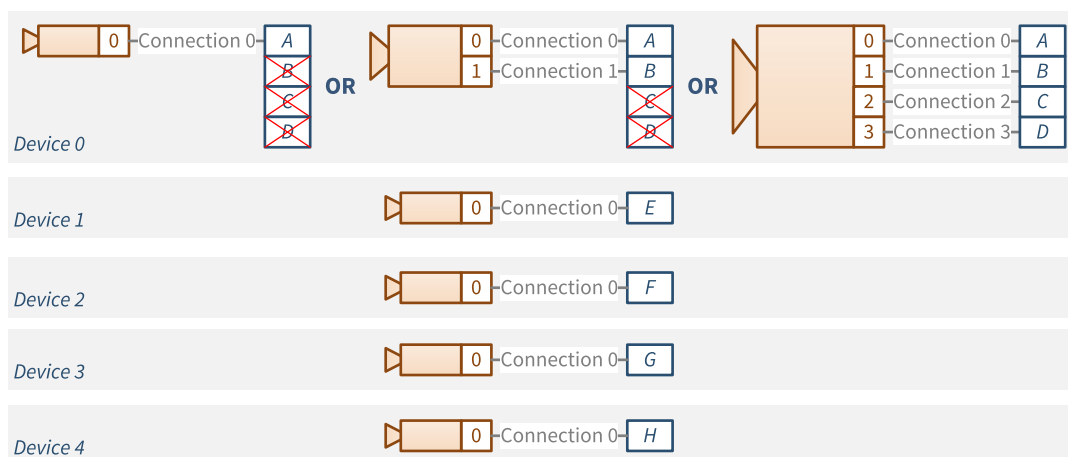
4D2222主机连接映射

一个或两个或三个或四个1-或2-连接设备



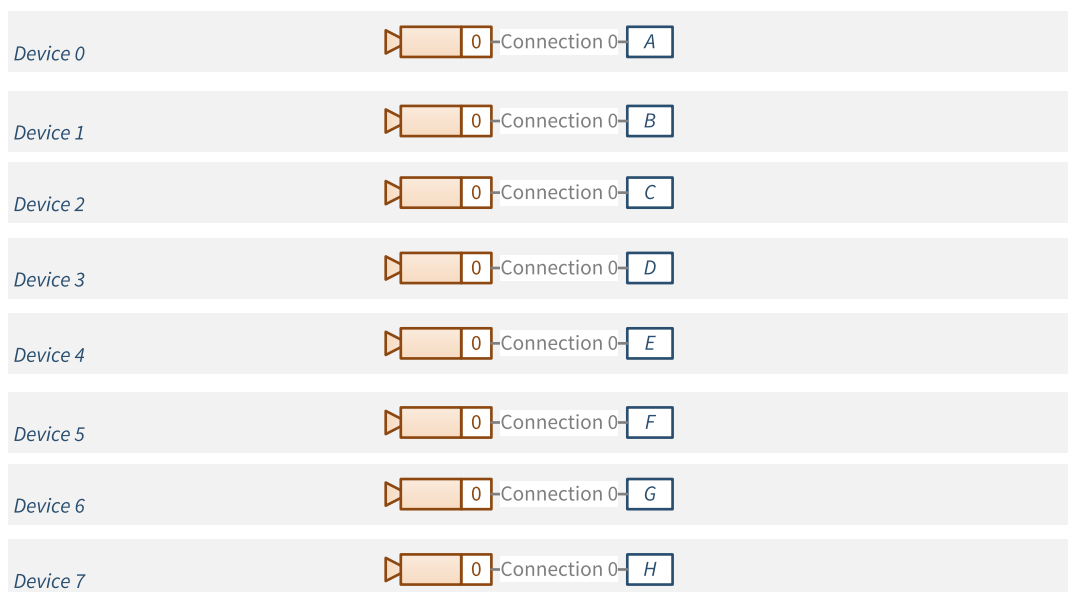
5D411111主机连接映射

一个1-或2-或4-连接和一个或两个或三个或四个1-连接设备



8D11111111主机连接映射

最多八个1-连接设备



3.4. 链路配置

自动链路配置

Coaxlink驱动程序为CoaXPress 1.0和CoaXPress 1.1设备提供了自动链接发现和配置。

对于CoaXPress主机接口的每个连接，发现过程确定：

- CoaXPress 设备的存在
- 断开连接的速度(设备到主机)
- 连接ID

通过接口模块的**CxpConnectionState**，**CxpDownConnectionSpeed**和**CxpDeviceConnectionID**功能，报告发现的结果。

请用户检查生成的链接配置是否适当：

- 对于应用程序需求，在链路带宽方面(链路速度和连接数量)
- 对于CoaxLink，在目标CoaxLink产品/固件组合支持的相机连接方案方面

手动链路配置

如有必要，用户可以手动配置“远程设备”的CoaXPress链接。

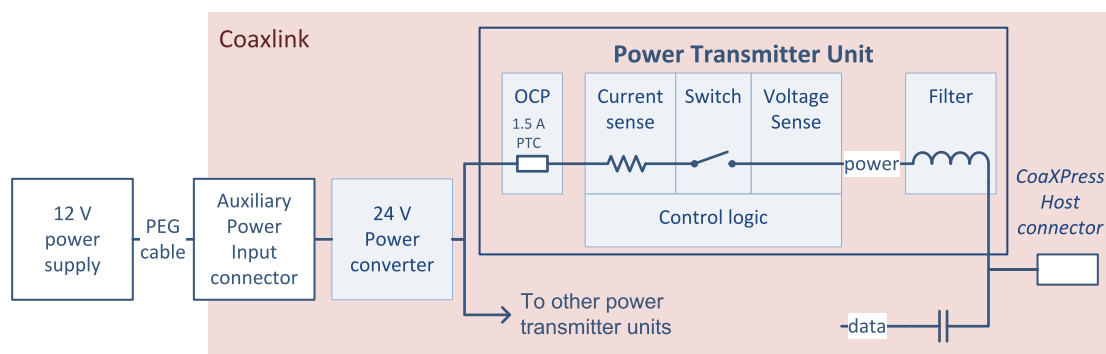
这可以通过将适当的值分配给Coaxlink设备模块的**CxpLinkConfigurationGenApi**功能来实现，而不考虑相机品牌。

指定该值**Preferred**将强制相机的首选链接配置：

- 链接速度设置为指定值
- 链接宽度被设置为指定的值，但可能仅限于Coaxlink侧的可用连接数。

3.5. CoaXPress供电

CoaXPress主机连接器上的每个连接，都能够通过CoaXPress 电缆向相机提供电源。



带12V至24V电源转换器卡的PoCXP功能框图

供电单元

动力传输单元PTU，负责安全供电。

它满足CoaXPress主机对CoaXPress标准的所有要求，即：

- 它能够向连接器提供高达17W的24V直流电源。
- 它执行过电流保护装置—OCP
- 它支持CoaXPress PoCXP检测方法

此外，它还为应用程序提供了以下功能：

- 控制PoCXP自动检测
- 关闭或中断供电
- 跳闸时重置OCP

12V电源输入，内置12V至24V电源转换器

适用于: Mono Duo Quad QuadG3 QuadG3LH QuadG3DF Quad3DLLE
Octo QuadCXP12 QuadCXP12J 3614

PCI Express Coaxlink卡的所有PTU，均由外部12 V电源供电(通过Coaxlink卡上的12V至24V板载电源转换器)。

外部12V电源，通过6针PEG电缆连接到**辅助电源输入接头**。



备注 12V电源通常由主机PC的电源提供。

此**AuxiliaryPowerInput**功能，报告外部电源和Coaxlink辅助电源输入接头之间的PEG电缆连接的状态。

此**CxpPoCxpPowerInputStatus**功能，报告24 V电源转换器的状态。

24伏电源输入

适用于: Duo104EMB Duo104MIL

PCIe/104 Coaxlink卡的所有PTU，均由外部24 V电源供电。

外部24V电源，通过4针电缆连接到**辅助电源输入接头**。

此**AuxiliaryPowerInput**功能，报告外部电源和Coaxlink辅助电源输入连接器之间的电源线连接的状态。

此**CxpPoCxpPowerInputStatus**功能，报告24 V输入的状态。

PTU控制逻辑

执行 **CxpPoCxpAuto** 命令时，PTU控制器启动一个 **PoCXP设备检测步骤**。

如果 **PoCXP设备检测步骤** 成功终止，则PTU通过关闭开关来通电。

如果 **PoCXP设备检测步骤** 失败，控制器则不会通电，并重试新的PoCXP检测步骤。可能的故障原因有：

- 外部电源未连接 (**AuxiliaryPowerInput = AuxiliaryPowerInput**)
- 外部电源关闭 (**CxpPoCxpPowerInputStatus = CxpPoCxpPowerInputStatus**)
- 没有连接相机
- 连接的相机不符合PoCXP

一旦通电，控制器将保持该状态，直到出现以下任一情况：

- 应用程序通过执行 **PoCxpTurnOff** 命令来禁用电源传输。
- 外部电源已断开 (**CxpPoCxpPowerInputStatus = NotOK**)
- 外部电源已关闭 (**CxpPoCxpPowerInputStatus = NotOK**)
- CoaXPress电缆断开 (在0.3秒的时间间隔内测量的平均输出电流小于8 mA)
- OCP跳闸 (输出电压降到，其时间间隔大于20毫秒)

执行 **CxpPoCxpTurnOff** 命令时，PTU关闭开关，并禁用PoCXP电源。在这种状态下，PTU不执行PoCXP检测程序。

该 **CxpPoCxpConfigurationStatus** 功能，报告PTU的配置状态：**Off** 或 **AUTO**。

此 **CxpPoCxpStatus** 功能，报告PTU状态：**Off**，**On** 或 **Tripped**。

重要说明



对于使用Coaxlink 驱动程序早期版本的用户

自驱动程序的版本3.1开始，PoCXP通电在系统通电时启用。

因此，当使用PoCXP供电的相机时，应用程序不再需要通过发出 **CxpPoCxpAuto** 命令来启用PoCXP供电。

过流保护

OCP电路配有一个PTC装置，可提供两种保护：

- 过载保护解决了负载过大的情况。
- 短路保护解决了意外短路的情况。

如果过载，根据电流水平和环境温度，PTC会在几秒或几分钟后跳闸(=逐渐打开电路)。电流越大，跳闸时间越短。这同样也适用于环境温度。

如果发生短路，PTC会立即跳闸。由此，PTU控制器进入跳闸状态，并打开开关。跳闸的PTC装置，在冷却后，恢复导电状态。这可能需要几秒钟。但是，在应用程序发出**CxpPoCxpTripReset**命令之前，PTU控制器将保持在跳闸状态。离开跳闸状态后，PTU可以启动新的PoCXP设备检测，如果成功，则重新建立电源。



备注 PTC能够在整个工作温度范围内保持17 W的功率，而不会跳闸。



备注 禁止超过17 W的功率和/或在工作温度范围以上操作Coaxlink卡，因为它可能导致意外的PTC跳闸。

3.6. CoaXPress I/O 通道

根据 CoaXPress 1.0和1.1标准，CoaXPress I/O通道

- 是 CoaXPress 链接的三个逻辑通道之一 (I/O、流、控制)
- 仅为 CoaXPress 链接的主连接 (连接0) 定义
- 用于在主机和设备之间，传输高优先级“触发”
- 仅在 CoaXPress 1.0上，用于在主机和设备之间，交换 GPIO 寄存器的状态。

Coaxlink只执行 CoaXPress主机到设备的触发！



备注 未实现 CoaXPress 设备到主机触发和 CoaXPress 1.0 gpio。

3.7. CoaXPress主机到设备触发

CoaXPress主机到设备触发，是CoaXPress I/O通道的功能，允许主机(帧捕获器)通过CoaXPress链接触发设备(相机)。

Coaxlink的Coaxpress主机接口为每个连接设备实现一个Coaxpress“主机到设备”触发发送器。

主机到设备触发源

CoaXPress “主机到设备”触发发送器可以来自：

- 相关相机和照明控制器的相机触发输出
- 任何可输入的通用I/O

触发源通过**CameraControlMethod**GenICam功能间接控制

- 当**CameraControlMethod**设置为**RG**或**RC**，触发源是关联相机和照明控制器的相机触发输出。
- 当**CameraControlMethod**设置为**CameraControlMethod**:
 - 触发源是I/O工具箱专用LIN工具的行源：设备0-Lin1、设备1-Lin2、设备2-Lin3和设备3-Lin4。
 - 通过配置专用LIN工具的 **LineInputToolSource**行源，任何支持输入的GPIO测线都可以用作触发源。
 - 外部触发信号的极性，可通过所选I/O控制块的设置**LineInverter**进行控制。
 - glitch-removal滤波器的时间常数，可通过所选I/O控制块的设置**LineFilterStrength**进行调整。
- 当**CameraControlMethod**设置为**NC**时，“主机到设备触发”发送器被禁用。

主机到设备触发发送器-默认设置

CoaXPress “主机到设备触发”发送器的Coaxlink实现，符合CoaXPress 1.0和1.1标准对低速CoaXPress “主机到设备”触发的要求(当其配置为默认设置时)：

- **CxpTriggerMessageFormat = Pulse**
- **CxpTriggerAckTimeout = 20.0**
- **CxpTriggerMaxResendCount = 3**

发送器在触发源信号的两边，启动一个触发事件：

- 它计算了一个延迟值，允许接收设备以固定的延迟重新创建事件。

另见：“相机触发延迟补偿”于页面44

- 它在下一个字符边界处的低速主机到设备连接上，插入一个高优先级的“触发包”。

然后，发射器等待来自设备(相机)确认：

- 如果在超时到期之前收到确认，则事务将正常终止。
- 如果在20微秒的超时时间内没有收到确认，发送器将执行 **重试**：重新发送触发数据包，并为确认启动新的等待期。
- 如果3次后仍未收到确认，则事件将异常终止。

当上一个事务未完成时，发送器不会启动新事件。

Default settings

```
CxpTriggerMessageFormat = Pulse; CxpTriggerAckTimeout = 20.0; CxpTriggerMaxResendCount = 3;
```

Case 1: CameraControlMethod = RG; camera replies immediately with ACK



Case 2: CameraControlMethod = RC; camera replies immediately with ACK



Case 3: no reply from camera: retry after timeout (20 us); transaction aborted after 3 retries



使用默认设置

触发消息事务情形1和情形2: 相机按预期，确认每条消息
情形3: 相机没有确认。重试3次后中止

事件报告

发送器报告以下事件:

- **CxpTriggerAck**:收到CoaXPress主机到设备触发数据包的确认。
- **CxpTriggerAck**: 将重新发送CoaXPress主机到设备触发数据包。

主机到设备触发发送器-备用设置

发射器可定制：

- 使用 **消息格式控件**，仅在源信号的上升沿发送触发消息
- 使用 **消息确认控件**，配置确认超时和重试次数

消息格式控件

Coaxlink的“主机到设备触发”发送器单元，提供了一个具有 **CxpTriggerMessageFormatGenICam**功能的“消息格式”控制。

脉冲消息格式(默认)

在默认情况下，**CxpTriggerMessageFormat**设置为 **Pulse**：发送器在输入脉冲的**两边**生成一个CoaXPress I/O通道主机到设备触发事件。

- 由上升沿触发的事件，将**上升沿触发数据包**从主机传输到设备。
- 由下降沿触发的事件，将**下降沿触发数据包**从主机传输到设备。



备注 每个触发脉冲都需要两个不同的CoaXPress I/O通道事务！

上升沿消息格式

当**CxpTriggerMessageFormat**设置为 **RisingEdge**时，发送器仅在输入脉冲的上升沿上，生成CoaXPress I/O通道主机到设备触发事件。

事务总是将一个**上升沿触发包**从主机传输到设备。



备注 每个触发脉冲都需要一个CoaXPress I/O事件通道。



备注 此格式不允许捕获器控制曝光时间！

切换消息格式

当**CxpTriggerMessageFormat**设置为 **Toggle**时，发送器仅在输入脉冲的上升沿上，生成CoaXPress I/O通道主机到设备触发事件。

该事务交替地发送一个**上升沿触发包**和一个**下降沿触发包**。



备注 每个触发脉冲都需要一个CoaXPress I/O事件通道。



备注 此格式不允许捕获器控制曝光时间！

该**CxpTriggerLevel**功能允许应用程序设置和/或获取CoaXPress主机到设备触发信号的当前级别。

消息确认控制

主机到设备触发发送器单元，提供了用户可配置的触发包确认机制：

- 超时值可以使用 **CxpTriggerAckTimeout** GenICam功能进行配置。
- 可使用 **CxpTriggerMaxResendCount** GenICam 功能配置重试次数。

启用确认检查(默认)

在默认情况下，**CxpTriggerAckTimeout** 设置为 **20.0** (20 微秒) 以及 **CxpTriggerMaxResendCount** 设置为 **3**。

Coaxlink 需要一个I/O通道确认包来响应每个触发包。如果在20微秒超时值内，未收到确认数据包，那么发送器将重新发送触发数据包。它最多执行3次重试。

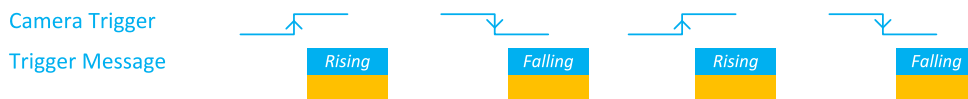
设置较大的**CxpTriggerAckTimeout**值允许设备有更多的时间来确认触发数据包。

禁用确认检查

设置**CxpTriggerAckTimeout**为**CxpTriggerAckTimeout**，禁用确认机制。触发事务在发送触发数据包后立即终止。

Alternate settings

Case 1: `CameraControlMethod = RG;CxpTriggerMessageFormat = Pulse; CxpTriggerAckTimeout = 0;`



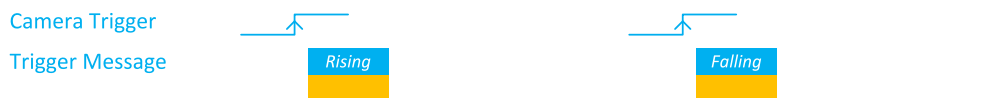
Case 2: `CameraControlMethod = RC;CxpTriggerMessageFormat = Pulse; CxpTriggerAckTimeout = 0;`



Case 3: `CameraControlMethod = RC;CxpTriggerMessageFormat = Rising; CxpTriggerAckTimeout = 0;`



Case 4: `CameraControlMethod = RC;CxpTriggerMessageFormat = Toggle; CxpTriggerAckTimeout = 0;`



使用备用设置来触发消息事务，以允许更高的触发率

最快触发率的备用设置

在以下情况下,可实现最快触发速率: **595.2 kHz @CXP-10和CXP-12链路速度**或 **297.6 kHz @CXP-6和更低链路速度**:

- **CameraControlMethod = RC** =(异步复位相机, 相机控制曝光),
- **CxpTriggerAckTimeout = 0** =(确认CKECHKICK已禁用),
- **CxpTriggerMessageFormat = CxpTriggerMessageFormat** 或 **CxpTriggerMessageFormat = CxpTriggerMessageFormat**.

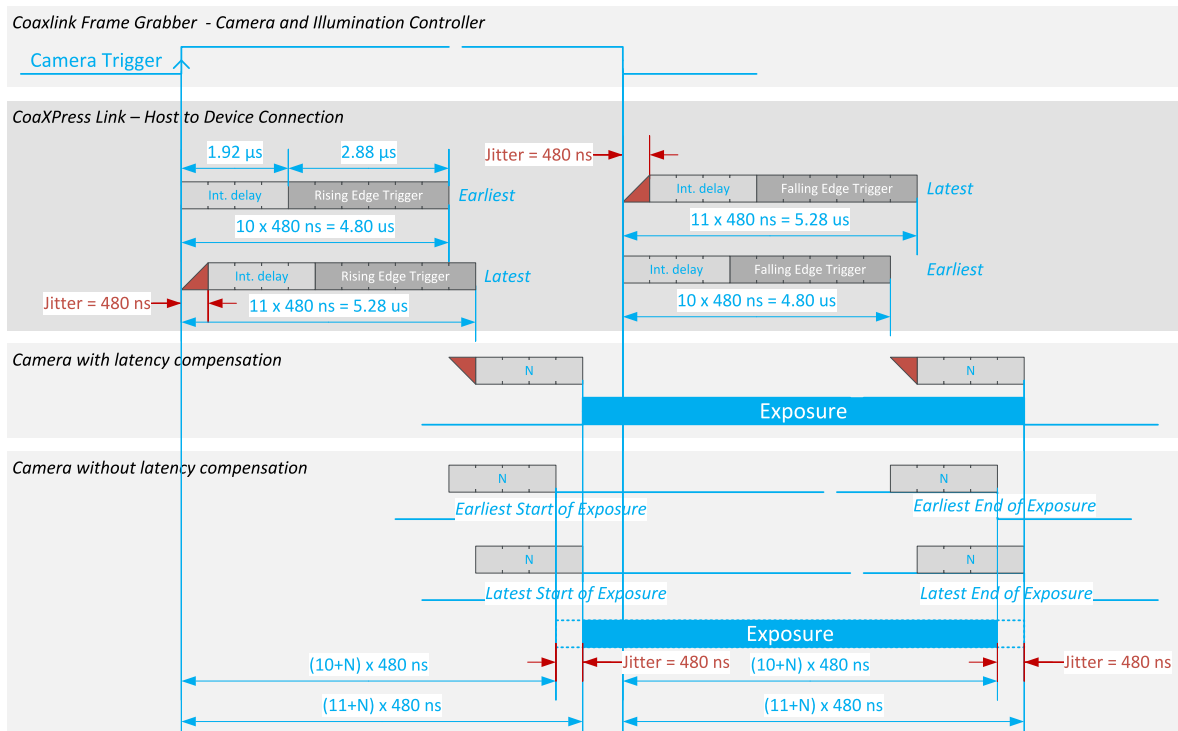
相机触发延迟补偿

触发精度

主机到设备触发数据包，通过CioaXPress链路的低速-加速-连接传输。触发包的传输只能从字符的边界开始。这引入了对应于一个字符传输时间的抖动：**240纳秒@CXP-10和CXP-12链路速度**或**480 纳秒@CXP-6和更低的链路速度**。

为了最小化触发抖动，触发事件和被发送的触发包之间的时间，以延迟值被编码到触发包，该延迟值以比特周期的1/24为单位表示：**1纳秒@CXP-10和CXP-12链路速度**或**2纳秒@CXP-6和更低链路速度**。然后，接收器(相机)可以使用此值，以**固定延迟**重新创建触发事件。它通过(一个字符时间的剩余部分)延迟解码消息来补偿传输抖动。

CoaXPress相机触发传输定时@CXP-6和更低的链路速度



CIC相机触发-传感器曝光定时图@CXP-6和较低链路速度

上图显示了，使用主机到设备CoaXPress触发消息，通过CoaXPress链接，将相机触发事件从帧捕获器传播到相机所需的时间延迟。

上图假设：

- **CameraControlMethod** 设置为 **CameraControlMethod**。
- 相机正确地确认触发信息，并有效地启动新曝光。

从相机触发信号的上(或下)沿(Coaxlink卡内部)到CoaXPress链接的延迟包括：

- 与时间延迟相对应的**0-480 ns**的可变延迟，直到低速CoaXPress连接上的下一个字符边界。
- 与触发发送器实施中的4字符管道延迟相对应的**1.92 μs**的固定延迟。
- 与6个字符的信息传输时间相对应的**2.88 μs**的固定延迟。

从CoaXPress链接到有效曝光开始(或结束)的延迟取决于相机。在上图中，此延迟假定为n个字符的时间($n=4$)。

抖动补偿相机

当相机执行CoaXPress抖动补偿时，发送器引入的单字符抖动(480ns)可以完全补偿。

总的延迟是**固定的**，但它仍然取决于相机：CoaxLink产品可能的最小延迟是**11 x 480 ns**，即。**5.28 μs**。

补偿后的剩余抖动可低至**4 ns**。

抖动-未补偿相机

当相机不执行CoaXPress抖动补偿时，发送器引入的单字符抖动(**480 ns**)将保持不变。

总的延迟是**可变的**，并取决于相机:CoaxLink产品的最小可能延迟时间是**(10~11)x480ns**，即。**(4.80 ~ 5.28) μ s**。

CoaXPress相机触发传输定时@ CXP-10和CXP-12链路速度

从相机触发信号的上(或下)沿(Coaxlink卡内部)到CoaXPress链接的延迟包括:

- 与时间延迟相对应的**0-2400 ns**的可变延迟，直到低速CoaXPress连接上的下一个字符边界。
- 与触发发送器实施中的4字符管道延迟相对应的**0.96 μ s**的固定延迟。
- 与6个字符的信息传输时间相对应的**1.44 μ s**的固定延迟。

从CoaXPress链接到有效曝光开始(或结束)的延迟取决于相机。在上图中，此延迟假定为n个字符的时间(n=4)。

抖动补偿相机

当相机执行CoaXPress抖动补偿时，发送器引入的单字符抖动(480ns)可以完全补偿。

总的延迟是**固定的**，但它仍然取决于相机:CoaxLink产品可能的最小延迟是**11 x 240 ns**，即。**2.64 μ s**。

补偿后的剩余抖动可低至**2 ns**。

抖动-未补偿相机

当相机不执行CoaXPress抖动补偿时，发送器引入的单字符抖动(**240 ns**)将保持不变。

总的延迟是**可变的**，并取决于相机:CoaxLink产品的最小可能延迟时间是**(10~11)x240 ns**，即。**(2.40 ~ 2.64) μ s**。

3.8. CoaXPress LED

CoaXPress 主机接口的每个连接器都与 CoaXPress 主机指示灯 LED 相关联，该指示灯指示 CoaXPress 链路连接的状态。

CoaXPress 主机指示灯 LED 状态描述

符号	LED 状态	意义
	断开	Coaxlink卡未通电
	稳定橙色	系统引导
	快速闪烁交替绿色/橙色	连接检测正在进行中；PoCXP处于活动状态。 即使连接检测更快，这种状态也至少会显示1秒。
	快速闪光橙	正在进行连接检测；PoCXP已关闭。 即使连接检测更快，这种状态也至少会显示1秒。
	稳定红	PoCXP过流保护已经跳闸。
	稳定绿	设备到主机的连接已建立，但未传输任何数据。
	慢脉冲橙色	设备到主机的连接已建立，但主机正在等待触发。
	快速闪烁的绿色。	设备到主机的连接已经建立，图像数据正在传输。

闪烁LED状态定时定义

指示	定时
快速闪光	12.5Hz@25%占空比 : 20 ms开启, 60 ms关闭
快速闪光交替(颜色1/颜色2)	12.5Hz@25%占空比 : 20 ms打开(颜色1), 60 ms关闭, 20 ms打开(颜色2), 60 ms关闭
慢闪	0.5Hz@50%占空比 : 1秒打开, 1秒关闭
慢脉冲(红色/橙色)	1Hz@20%占空比 : 200ms开启, 800ms关闭

3.9. 连接测试

Coaxlink的Coaxpress主机接口，提供了连接测试工具，以测试Coaxpress链接的上下连接质量(根据Coaxpress 1.1标准第8.7节中定义的步骤)

对于每个单独的Coaxpress连接器，它实现

- 测试生成程序
- 测试接收机

测试生成器程序，发送包含由序列生成器生成的已知测试模式的测试数据包。它为传输的每个测试包，增加包计数器。

测试接收器，将接收到的测试数据包内容与其本地序列生成器进行比较。它为数据包中不同的每个字增加错误计数器，并为接收到的每个测试包增加包计数器。



备注 测试包计数器，显示了发送和接收了多少测试包，由此可以判断错误计数器中值的统计含义。



备注 可以同时运行“设备到主机”和“主机到设备”的连接测试。

3.10. CoaXPress链接验证工具

简介

简述

CoaXPress链接验证工具 (CXLVT)可用于验证CoaXPress链接的操作参数。

对于快速测试,运行CXLVT直到达到100%的置信水平,即单个位错误(PER)的概率为 10^{-10} 或更好 10^{-11} 。这只需要花几分钟。

对于广泛的测试,运行CXLVT直到达到100%的置信水平,即PER为 10^{-12} 或更好 10^{-13} 。这需要几个小时。

另见: 有关误码率测试理论的更多信息,请参阅http://en.wikipedia.org/wiki/Bit_error_rate。

主机PC要求

- 主机PC必须配备至少一个Euresys Coaxlink线路板。
- Coaxlink驱动程序必须安装在主机PC上。

相机要求

- 相机必须能够生成静态图像图案。

安装

CXLVT包含在gent1.exe中,这是一个随Coaxlink驱动程序一起提供的命令行工具。无需进一步安装。

gentl ber 命令

通过命令 `ber (gentl.exe)` 调用CXLVT。

```
$ gentl ber --help
GenTL Explorer

gentl ber [OPTIONS]
  Measure bit error rate confidence level (a.k.a. link validation tool)

Flags:
  --if=ID           Interface ID
  --dev=ID          Device ID
  --ds=ID           DataStream ID
  --buffers=INT     Buffer count (default: 4)
  --set=SETTINGS    GenApi settings, such as Module.Feature=INT
  --setup=FILE      Path to script to execute before starting stream
  --run=FILE        Path to script to execute concurrently with stream
  --remotexml=FILE  Use FILE as register description (default:
                    register description is read from remote device)
  -c --create-only  Create a reference pattern and quit (requires
                    --output)
  -i --input=FILE   Input reference pattern file (default:
                    automatically create a reference image before
                    measuring the bit error rate confidence level)
  -o --output=FILE  Output reference pattern file (default: no output
                    file)
  --enable-dump=FILE Enable dump of defective surfaces to files with
                    the given file path prefix

Common flags:
  --cti=LIBPATH     Path to GenTL producer library. Default: use
                    EURESYS_COAXLINK_GENTL64_CTI and
                    GENICAM_GENTL64_PATH environment variables to locate
                    the library.
  -j=N             Limit the number of CPU cores to use to N
                    (default: 2)
  -h --help        Display help message
  -V --version     Print version information
  --numeric-version Print just the version number
  -v --verbose     Loud verbosity
  -q --quiet       Quiet verbosity
```

`gentl ber --帮助`

```
> help
Ber commands:
  levels                show confidence levels
  levels -N             show confidence levels every N seconds
  results              show intermediate bit error rate results
  results -N           show intermediate bit error rate results every N
                      seconds
  report FILE          write current report to a file
  enable-dump FILE     enable dump of defective surfaces to files with the
                      given file path prefix
  disable-dump         disable dump of defective surfaces
General commands:
  quiet                set verbosity level to quiet
  normal               set verbosity level to normal
  loud                 set verbosity level to loud
  help                 display this help message
  exit                 exit the CLI
>
```

gentl ber 命令

测试步骤

要设置CoaxPress 链接验证工具请执行以下操作：

1. 使用genicam浏览器：
 - 将相机配置为正常运行，并选择固定测试模式作为视频源
 - 将帧捕获器配置为正常操作
2. 打开命令shell并执行gentl ber以启动Read-Eval-Print-Loop
3. 使用 results 命令获取中间结果。
 - 检查所采集的图像计数器数量是否定期增加
 - 检查置信水平
4. 运行测试，直到达到所需的置信水平。这可能需要几个小时。

操作

CoaXPress链接验证工具(CXLVT)验证CoaXPress链接安装的操作参数(比特率、电缆类型、电缆长度),从而获得可靠的长期性能。

CXLVT,通过已知的置信水平,估计CoaXPress链接设置中单个位错误的概率来实现这一点。

我们定义:

- **PER:**数字连接(如CoaXPress链接)中单个位错误的概率;这是我们要估计的未知量。
- **BER:**误码率:实际由CXLVT进行测量的误码率。

通常认为,如果 $PER < 10^{-12}$, CoaXPress链接将可靠运行。该准则与其它数字串行图像传输方案中使用的准则相似。当然,更好(更低)的PER将为操作的可靠性提供更大的保证。

CXLVT根据对BER的测量,在足够长的时间内,计算出置信水平(CL),或PER小于一组值(10^{-10} , 10^{-11} , 10^{-12} , 10^{-13} , 10^{-14})的可能性,以积累必要的证据。

当启动时,CXLVT会显示这些置信水平,因为随着时间的推移,证据会积累起来,如下面的屏幕截图所示。

在CXLVT操作期间,输入levels命令,将显示5PER值的置信水平。

```
> levels
-----
Confidence level (rounded) that the probability of error is less than:
      1.0e-10      1.0e-11      1.0e-12      1.0e-13      1.0e-14      Elapsed
      99.99%      64.97%      9.95%      1.04%      0.10%      Time
      0:08:56
-----
>
```

启动后8秒所报告的置信水平

8秒后,我们已经达到99.99%的置信水平,即PER小于 10^{-10} 。也许PER比这要好得多,但在这个阶段,我们没有足够的证据来断定情况就是这样。CXLVT必须继续。

通过输入 `results` 命令(在CXLVT操作期间),可显示额外的信息,之后CXLVT将继续其正常操作,若有必要,即可达到预定的PER所需的置信水平。

```
> results
Intermediate results:
-----
Duration (hours:minutes:seconds):          0:16:37
Duration (seconds):                        997
Acquired images:                           11975
Bad images:                                0
Acquired bits:                             1.986509e11
Bit errors:                                0.000000e0
Average bit errors per bad image:          0
Bit rate (bits per second):                1.992486e8
Bit error rate:                            0.000000e0
Confidence level (rounded) that the probability of error
is less than 1.0e-10:                      99.99%
          1.0e-11:                          86.28%
          1.0e-12:                          18.01%
          1.0e-13:                           1.96%
          1.0e-14:                           0.19%
>
```

16分钟后所报告的中间结果

从这张截图中,我们可以得出结论,16分钟后,PER小于 10^{-11} 的置信水平从64.97%上升到86.28%。

CXLVT 应继续进行,直到PER小于 10^{-12} 的置信水平(最多-更强的测试是PER小于 10^{-13})达到令人满意的水平(至少为95%,更强的置信水平为99%)。这可能需要相当长的时间,因为这些结果需要大量的证据。

```
Intermediate results:
-----
Duration (hours:minutes:seconds):          3:30:12
Duration (seconds):                        12612
Acquired images:                           149753
Bad images:                                0
Acquired bits:                             2.484223e12
Bit errors:                                0.000000e0
Average bit errors per bad image:          0
Bit rate (bits per second):                1.969729e8
Bit error rate:                            0.000000e0
Confidence level (rounded) that the probability of error
is less than 1.0e-10:                      100.00%
          1.0e-11:                          99.99%
          1.0e-12:                          91.66%
          1.0e-13:                          21.99%
          1.0e-14:                           2.45%
```

3.5小时后所报告的中间结果

从这个截图中,我们可以得出结论,在3.5小时后,PER小于 10^{-12} 的置信水平从18.01%上升到91.66%。

要生成报告，请执行report命令。

```
> report ber-report  
Report ber-report-20171107-093451.log successfully created  
> █
```

生成报告命令行

4. 图像数据路径

4.1. FIFO 缓冲器	56
4.2. 采集门	57
4.3. 像素数据处理	58
4.4. 像素数据处理配置	60
4.5. 像素分量解包	63
4.6. 像素分量重新排序	65
4.7. 端序转换	66
4.8. 像素排序	67
4.9. 图像数据传输	68

4.1. FIFO 缓冲器

每个产品的**DRAM**内存大小

产品	DRAM内存大小
1629 Coaxlink Duo PCIe/104-EMB	512 MB
1630 Coaxlink Mono	512 MB
1631 Coaxlink Duo	1 GB
1632 Coaxlink Quad	1 GB
1633 Coaxlink Quad G3	1 GB
1633-LH Coaxlink Quad G3 LH	1 GB
1634 Coaxlink Duo PCIe/104-MIL	512 MB
1635 Coaxlink Quad G3 DF	1 GB
1637 Coaxlink Quad 3D-LLE	1 GB
3602 Coaxlink Octo	2 GB
3603 Coaxlink Quad CXP-12	2 GB
3620 Coaxlink Quad CXP-12 JPEG	2 GB

DRAM内存根据安装的固件变体进行分区。

- 所有固件变体都为每个设备的每个流分配一个名为**FIFO 缓冲器**的分区。
- 支持**FFC**固件变体，为增益和偏移系数的存储分配一个分区。

FIFO缓冲器操作

FIFO缓冲器作为FIFO操作，以将CoaXPress数据流与像素处理和PCI Express数据流分离。

它吸收了PCI Express数据流的临时释放，确保了可靠的CoaXPress数据采集。

无论像素处理器和PCI Express接口的限制如何，它都能以最高的数据速率，实现突发模式的CoaXPress数据采集。

4.2. 采集门

采集门控制从板载FIFO缓冲器中提取图像数据。它丢弃不需要获取的图像数据，并将其输入"[像素数据处理](#)"在下一页。

区域扫描采集

基于 DSStartAcquisition and DSStopAcquisition 功能的应用调用，门在帧边界处打开和关闭。



备注 相机和照明控制器，通过使用各种方案发出相机触发来间接控制采集频闪。

行扫描采集

根据应用程序DSStartAcquisition and DSStopAcquisition功能的调用，以及根据图像采集控制器的设置，到扫描开始和扫描结束触发，门在线边界处打开和关闭。

另见：有关更多信息和配置说明，请参阅"[行扫描采集](#)"于页面216。

4.3. 像素数据处理

图像像素数据处理器，对图像数据流执行以下连续操作：

CoaXPress 位流切片

此操作根据位深-*input-bit-depth*-(由CoaXPress图像头的*PixelF*属性指定)从CoaXPress图像数据提取单个像素分量数据。

所有分量具有相同的像素位深度。可能的值是8-/10-/12-/14-和16位。

切片器为每个图像行提供构建多个像素(由CoaXPress图像头*Xsize*属性指定)所需的所有像素分量。

切片器丢弃CoaXPress行填充数据。

平场校正

此操作对每个像素组件应用线性增益和偏移变换。

另见：有关更多信息和配置说明，请参阅["平场校正"](#)于页面163。

查找表格处理

此操作将查找表格转换应用于每个单独的像素分量。

另见：有关更多信息和配置说明，请参阅["查找表处理"](#)于页面173。

Bayer CFA译码

此操作将相机发出的原始Bayer CFA数据流，转换为RGB颜色数据流。

另见：有关更多信息和配置说明，请参阅["Bayer CFA解码"](#)于页面187。

像素分量解包

此操作将10位、12位和14位像素分量，解包为8位或16位。

它可以禁用单色和Bayer CFA像素格式。

另见：有关更多信息和配置说明，请参阅["像素分量解包"](#)于页面63。

像素分量排序

此操作修改多分量像素数据的分量顺序。

另见：有关更多信息和配置说明，请参阅["像素分量重新排序"](#)于页面65。

端序转换

此操作修改16位像素分量数据的字节顺序。

另见：更多信息，请参阅"端序转换"于页面66。

图像行构建

此操作生成连接图像行所有像素的分量数据：

- 8位像素分量与字节边界对齐
- 16位像素分量(可能通过解包或查找表格处理进行扩展)与字(2字节)边界对齐, 2字节根据小端序约定存储。

线填充

此操作将填充位或字节附加到图像行数据, 以达到硬件实现所需的下一个对齐边界。

对齐边界要求, 是特定于产品的, 例如:

- 64位用于 **1630 Coaxlink Mono**, **1631 Coaxlink Duo**, 和 **1632 Coaxlink Quad**
- 128位用于 **1633 Coaxlink Quad G3**, **1633-LH Coaxlink Quad G3 LH** 和 **1635 Coaxlink Quad G3 DF**

加工性能

像素处理器支持最高的相机像素速率。除非另有规定, 所有上述操作都是在(以可忽略的延迟)将数据传输到GenTL的同时执行的。

PCI Express带宽限制

当获取像素位深度大于8位的像素时, 每个像素将扩展到16位。在这些情况下, 主PC的PCI Express带宽限制, 可能会对可实现的帧或线速率, 产生负面影响。

板载内存带宽限制

对于FFC用例, 板载内存带宽不足以维持完整的CoaXpress数据速率。

4.4. 像素数据处理配置

本主题列出了适用于每类相机像素格式的像素数据处理配置。

单色像素格式-禁用LUT

相机像素格式	FFC	启封模式	输出像素格式
Mono8	关闭/打开	LSB/MSB	Mono8
Mono10pmsb	关闭/打开	LSB	Mono10
	关闭/打开	msb	Mono16
	关闭	关闭	Mono10pmsb
Mono12pmsb	关闭/打开	LSB	Mono12
	关闭/打开	msb	Mono16
	关闭	关闭	Mono12pmsb
Mono14pmsb	关闭/打开	LSB	Mono14
	关闭/打开	msb	Mono16
	关闭	关闭	Mono14pmsb
Mono16	关闭/打开	LSB/MSB	Mono16

单色像素格式-启用LUT

相机像素格式	FFC	查阅表格配置	启封模式	输出像素格式
Mono8	关闭/打开	M_8x8	LSB/MSB	Mono8
Mono10pmsb	关闭/打开	M_10x8	LSB/MSB	Mono8
		M_10x10	LSB	Mono10
			msb	Mono16
		M_10x16	LSB/MSB	Mono16
Mono12pmsb	关闭/打开	M_12x8	LSB/MSB	Mono8
		M_12x12	LSB	Mono12
			msb	Mono16
		M_12x16	LSB/MSB	Mono16

Bayer CFA像素格式-禁用CFA解码器

相机像素格式	FFC	拆包模式	输出像素格式
Bayer**8	关闭/打开	lsb msb	Bayer**8
Bayer**10pmsb	关闭/打开	lsb	Bayer**10
	关闭/打开	msb	Bayer**16
	关闭	关闭	Bayer**10pmsb
Bayer**12pmsb	关闭/打开	lsb	Bayer**12
	关闭/打开	msb	Bayer**16
	关闭	关闭	Bayer**12pmsb
Bayer**14pmsb	关闭/打开	lsb	Bayer**14
	关闭/打开	msb	Bayer**16
	关闭	关闭	Bayer**14pmsb
Bayer**16	关闭/打开	lsb msb	Bayer**16

Bayer CFA像素格式-启用CFA解码器

输入像素格式	FFC	RedBlueSwap	输出像素格式
Bayer**8	关闭/打开	关闭	RGB8
		打开	BGR8
Bayer**10pmsb	关闭/打开	关闭	RGB10
		打开	BGR10
Bayer**12pmsb	关闭/打开	关闭	RGB12
		打开	BGR12
Bayer**14pmsb	关闭/打开	关闭	RGB14
		打开	BGR14
Bayer16	关闭/打开	关闭	RGB16
		打开	BGR16

RGB像素格式

输入像素格式	FFC	启封模式	RedBlueSwap	输出像素格式
RGB8	关闭/打开	LSB/MSB	关闭	RGB8
			打开	BGR8
RGB10pmsb	关闭/打开	LSB	关闭	RGB10
			打开	BGR10
		msb	关闭	RGB16
			打开	BGR16
RGB12pmsb	关闭/打开	LSB	关闭	RGB12
			打开	BGR12
		msb	关闭	RGB16
			打开	BGR16
RGB14pmsb	关闭/打开	LSB	关闭	RGB14
			打开	BGR14
		msb	关闭	RGB16
			打开	BGR16
RGB16	关闭/打开	LSB/MSB	关闭	RGB16
			打开	BGR16

RGBa 像素格式

输入像素格式	FFC	UnpackingMode	RedBlueSwap	输出像素格式
RGBa8	关闭/打开	LSB/MSB	关闭/打开	RGBa8
RGBa10pmsb	关闭/打开	LSB	关闭/打开	RGBa10
		msb	关闭/打开	RGBa16
RGBa12pmsb	关闭/打开	LSB	关闭/打开	RGBa12
		msb	关闭/打开	RGBa16
RGBa14pmsb	关闭/打开	LSB	关闭/打开	RGBa14
		msb	关闭/打开	RGBa16
RGBa16	关闭/打开	LSB/MSB	关闭/打开	RGBa16

4.5. 像素分量解包

简介

像素数据处理器，能够将10位、12位和14位像素分量数据解包为16位像素数据。

解包操作由用户通过GenICam**UnpackingMode**功能进行配置。有三种选择：

- **Lsb**: 解包至LSB(自4.3版以来的默认设置)
- **Msb**: 解包至msb
- **Off**: 不解包

解包至LSB

像素分量数据的有效位与数据容器的**最低有效位**对齐。填充0位(根据需要)放在**最高有效位**中, 以到达下一个8-位边界。

- 10-bit 像素: 0000 00<pp pppp pppp>
- 12-bit 像素: 0000 <pppp pppp pppp>
- 14-bit 像素: 00<pp pppp pppp pppp>



备注 解包到LSB不会修改像素分量值。

解包到msb

像素分量数据的有效位与数据容器**最高有效位**对齐。填充0位(根据需要)放在**最低有效位**中,以到达下一个8-位边界。

- 10-bit 像素:<pppp pppp pp>00 0000
- 12-bit 像素:<pppp pppp pppp> 0000
- 14-bit 像素:<pppp pppp pppp pp>00



提示 将10位、12位和14位像素分量解包到msb,分别是像素分量值64、16和4倍。



备注 解包8位和16位像素分量,是一种单流操作:

- 数据容器的大小不变:一个字节用于8位像素分量;两个字节用于16位像素分量。
- 数据位未被修改。



警告 解包10位、12位和14位像素分量,数据量分别增加160%、133%和114%。

不解包

相机通过CoaXPress链接传输的压缩图像数据,按原样传送到输出缓冲区。



警告 此选项仅适用于提供单色和拜耳CFA像素的相机的区域扫描固件变体!

4.6. 像素分量重新排序

Coaxlink 图像数据流像素处理器可以配置为交换三分量像素的第一和第三分量数据。

交换通过 **RedBlueSwap** 布尔型 GenICam 功能控制：

- 当设置为 **False** (默认设置) 时，将保留原始分量顺序。
- 当设置为 **True** 时，第一个和第三个分量被交换。

此功能可用于从以下位置获取图像：

- 提供三分量像素数据的 RGB 彩色相机，
- BAYER CFA 彩色相机(假如启用 BAYER CFA 解码功能)。

4.7. 端序转换

Coaxlink 图像数据流像素处理器，使用小端序转换提供16位像素分量。

将UnpackingMode is set to Off时，不执行转换。

小端序转换

多字节数据的最低有效字节，存储在最低地址位置。

例如，16位数据存储在两个连续的字节位置中，如下所示：

内存字节位置	内存字节内容
N	数据 [7:0]
N+ 1	数据 [15:8]

4.8. 像素排序

Coaxlink 图像数据流像素处理器，保留了 CoaXPress 数据流的像素顺序。

图像帧的像素数据，存储在连续的地址位置，从位于最低地址的第一行的第一个像素开始。

图像帧的连续行，连接在图像缓冲区中。

4.9. 图像数据传输

缓冲填充

DMA引擎根据行扫描和区域扫描图像采集的不同规则，通过PCI Express总线，将处理后的图像数据传输到分配的GenTL缓冲区。

GenTL缓冲区填充规则-区域扫描相机

在区域扫描成像中，按照以下规则填充GenTL缓冲区：

- 默认情况下，帧的第一个获得的行数据存储在新缓冲区的开头。通过将 **StripeArrangement** 设置为 **StripeArrangement** 来启用垂直图像翻转时，帧第一次所采集的行数据，存储在新缓冲区的最后一整行的位置。
- 当图像传输到主机内存时，缓冲区(可能是部分填充的)可供应用程序处理。
- 当缓冲区太小而不能包含完整的帧时，剩余的数据将被丢弃。

GenTL 缓冲区填充规则-行扫描相机

在行扫描成像中，按照以下规则填充GenTL缓冲区：

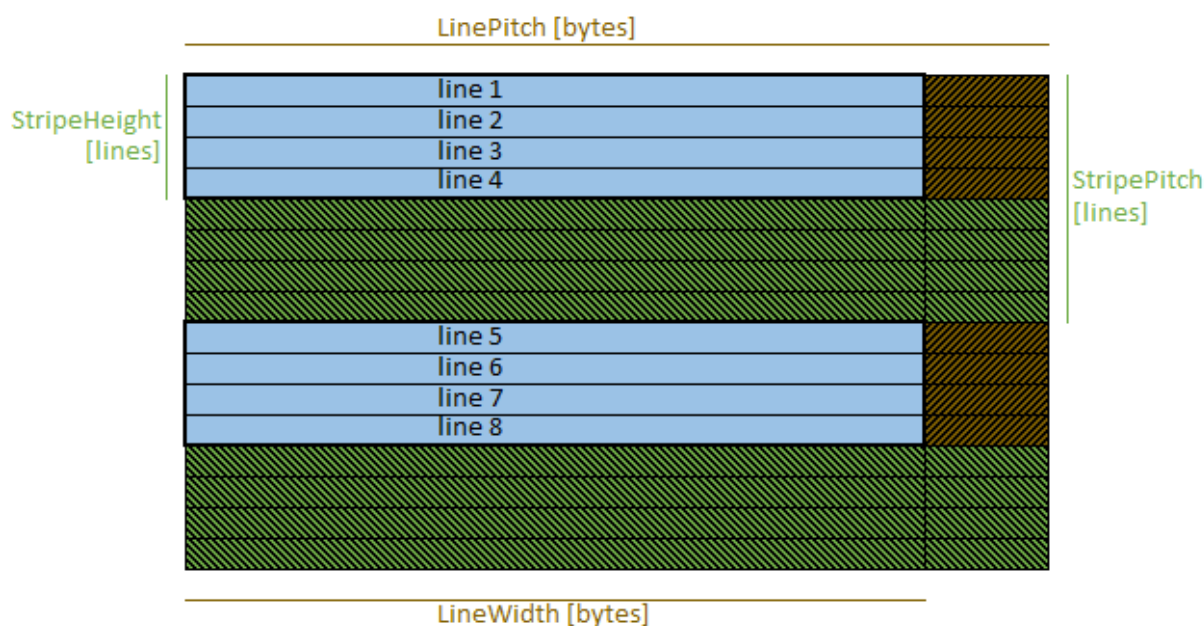
- 在默认情况下，扫描的第一个采集的行数据，存储在新缓冲器的开头。通过将 **StripeArrangement** 设置为 **Geometry_1X_1YE** 来启用垂直图像翻转时，扫描第一次所采集的行数据，存储在新缓冲区的最后一整行的位置。
- 缓冲区包含整数个图像行数据。
- 当缓冲区的剩余空间不足以存储图像行数据时，采集继续进入新的缓冲区，填充的缓冲区可供应用程序处理。
- 当获取扫描的最后一行数据时，可能部分填充的最后一个缓冲区，可供应用程序处理。

图像数据填充

DMA引擎提供了通过添加行填充或条带填充在缓冲区中以不同方式组织数据的能力。



备注 在Coaxlink 驱动程序6.2之前，DMA 引擎将整个图像数据作为单个1D实体传输，而不考虑2D结构：处理后的图像数据行被连接到目标缓冲区。



图像缓冲区填充模型

线填充

LineWidth和**LinePitch**功能控制行填充。

当**LinePitch** > **LineWidth**时，启用行填充：DMA引擎在每个图像行的末尾保留**LinePitch - LineWidth**填充字节。

LinePitch 可设置为 **0** 以禁用行后填充。

条纹填充

条纹是一组相邻的线条。高度为1的条纹是一条线。

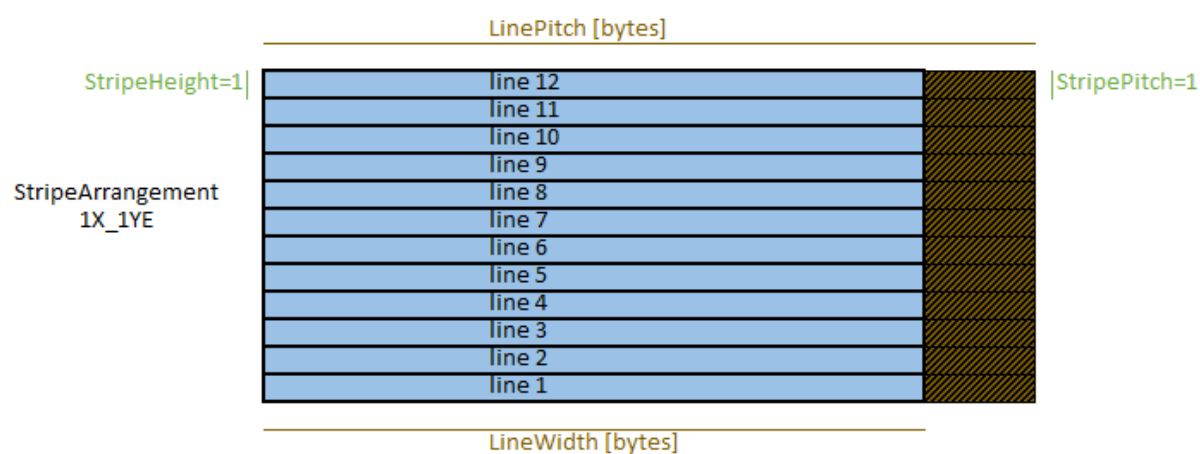
StripeHeight和**StripePitch**功能，控制条纹填充。

当**StripePitch** > **StripeHeight**时，启用条纹填充：DMA引擎会在每个条纹的末尾留下**StripePitch - StripeHeight**填充行。

StripePitch 可设置为 **0** 以禁用行后填充。

垂直图像翻转

DMA引擎提供垂直翻转图像的功能。



翻转的图像数据

垂直图像翻转由数据流模块的 **StripeArrangement** 功能控制。

在默认情况下, **StripeArrangement** 设置为 **1X_1Y**: 垂直图像翻转被禁用。

当把 **StripeArrangement** 设置为 **1X_1YE**时, 驱动程序使用以下公式来确定缓冲区中第一条图像行的位置:

$$\text{BufferBase} + (\text{BufferSize} + \text{LinePitch} - \text{LineWidth}) / \text{LinePitch} * \text{LinePitch} - \text{LinePitch}.$$

因此:

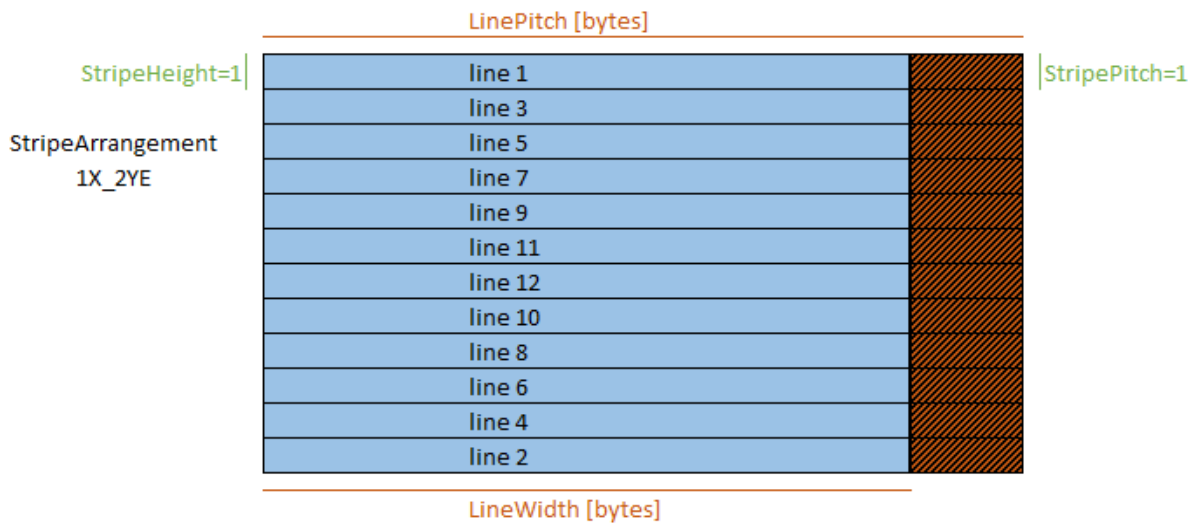
- 如果缓冲区太小, 则会丢失图像的底部部分(由相机给出);
- 行将从 **BufferBase + n * LinePitch**开始;
- 仅传输完整的行;
- 如果缓冲区大小不是 **LinePitch** 字节的倍数, 则缓冲区末尾的一些字节将保持不变。



备注 在计算上述公式时, 如果 **LinePitch** 等于 **0**, 则将使用 **LineWidth**、
同样地, 如果 **StripeHeight** 等于 **0**, 则使用1。

整理 1X_2YE 图像

DMA引擎提供整理 1X_2YE 图像的能力。



将 **StripeArrangement** 设置为 **StripeArrangement** 时，驱动器通过使用以下公式确定相机输出的第二行的目的地(即缓冲区中最后一个图像行的位置):

$$\text{BufferBase} + (\text{BufferSize} + \text{LinePitch} - \text{LineWidth}) / \text{LinePitch} * \text{LinePitch} - \text{LinePitch}$$



备注 这是缓冲区中最后一行的地址，其大小足以接收一整行。

因此：

- 如果缓冲区太小，相机输出的最后一行(即图像的中间部分)将丢失；应用程序应避免出现这种情况，
- 行将从 **BufferBase + n * LinePitch** 开始，
- 仅传输完整的行。(在计算上述公式时，如果 **LinePitch** 等于 **0**，则将使用 **LineWidth**。同样地，如果 **StripeHeight** 等于 **0**，则将使用 **1**)

StripeHeight 和 **StripeHeight** 不能设置为大于 **StripeHeight** 的值。

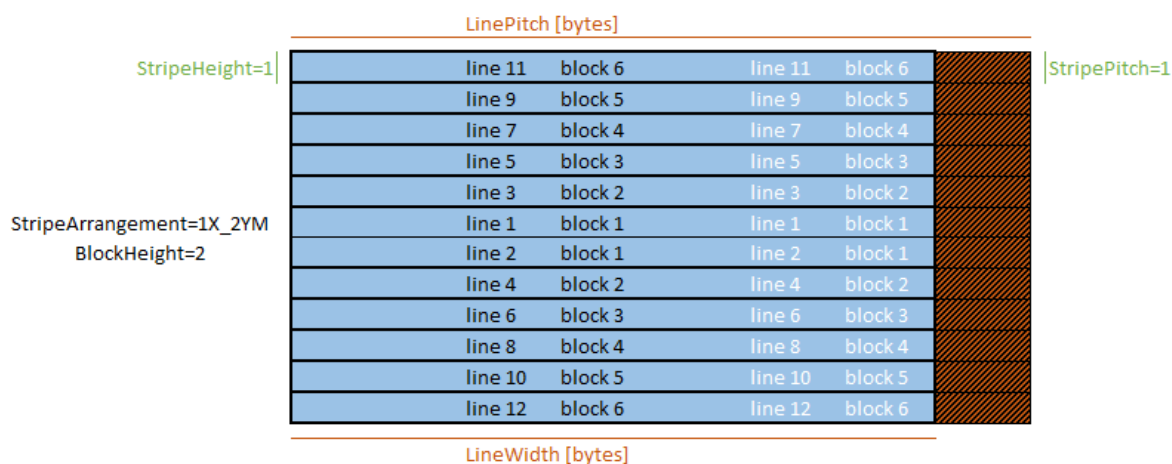
整理 1X_2YM 图像

DMA引擎提供整理 1X_2YM 图像的能力。

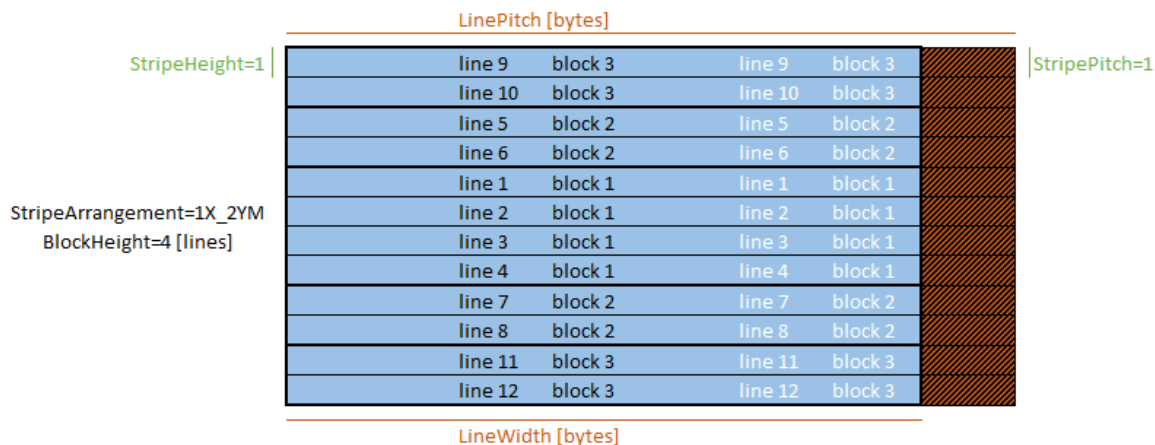


备注 在下图中，由 **设备** 发送的行号和块号是 **白色** 的，由 **主机** 接收的行号和块号是 **黑色** 的。

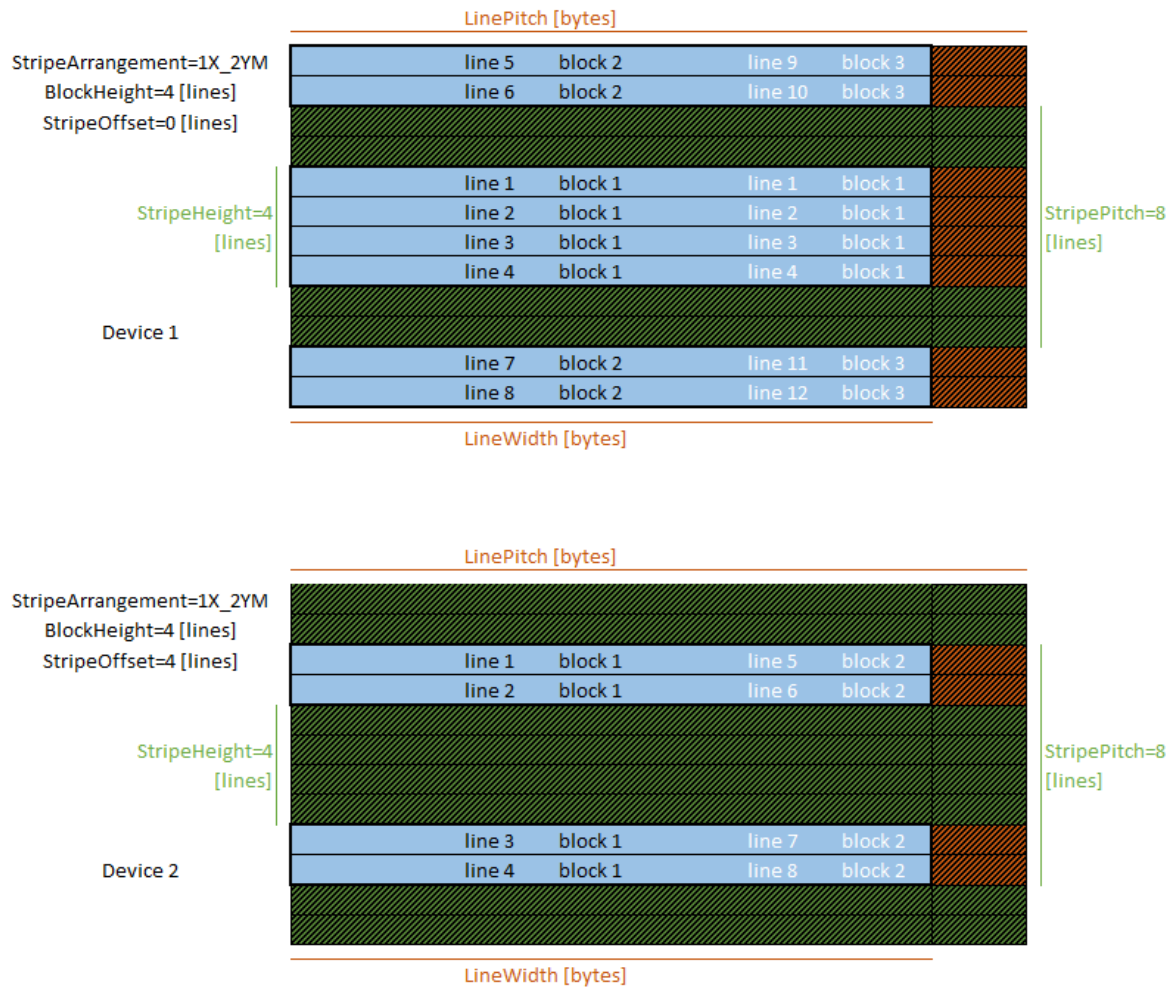
1X_2YM 相机按 2 个块来传送行



1X_2YM 相机按 4 个块来传送行



1X_2YM相机以4个块到2台主机来传送行



统计

流统计工具，在卡输出时监控图像数据流，并为应用程序提供平均帧、行和数据速率。

流统计抽样方法

StatisticsSamplingSelector 确定 **平均间隔**。它可以是以下任何一种：

- 或 **LastTenSeconds**: 最后完成的时隙为1或10秒。
- **Last2Buffers, Last2Buffers, Last2Buffers, Last2Buffers**: 最后的 2、10、100、或1000个 获取的缓冲区
- **LastAcquisition**: 上次采集活动周期。也就是说，从上次DSStartAcquisition()函数调用到现在，不论采集是否处于活动状态，直到最后一次DSStopAcquisition()函数调用。
- **LastAcquisition**: 时间间隔 (**LastAcquisition** 和 **LastAcquisition** 命令之间)

默认采样方法为 **LastSecond**。

统计数据

当获得以下任一特征时，将有效地计算统计数据：

- **StatisticsFrameRate** 报告以帧/秒(区域扫描)表示的平均帧速率。
- **StatisticsLineRate** 报告以行/秒(行扫描)表示的平均行速率。
- **StatisticsDataRate** 报告以兆字节/秒表示的平均数据速率

对于在平均间隔期间填充的每个GenTL缓冲区，工具计数：

- 填充的GenTL 缓冲区的数量和相应的帧数(区域扫描)或行数(行扫描)
- 传输的图像数据字节数。

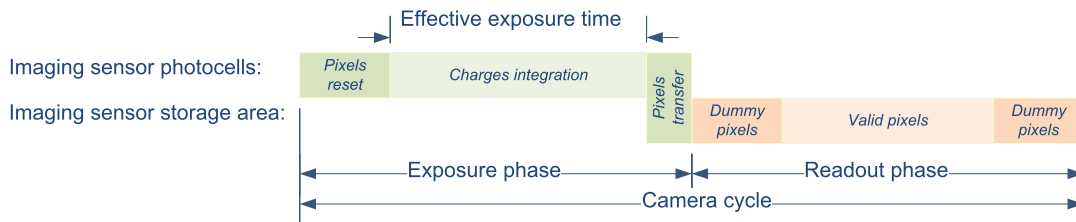
相关的GenICam 特性，很少被收集到GenTL数据流模块的流统计类别中。

5. 相机控制原理

5.1. 相机循环	76
5.2. 相机循环串联规则	77
5.3. 相机控制方法	79

5.1. 相机循环

相机循环由两个连续的阶段组成：曝光阶段和读出阶段。



相机循环

曝光阶段

曝光阶段是成像传感器的光电池对由入射光子感应的电荷进行整合的时间段。

对于具有电子快门的相机，曝光阶段从清除所有传感器光电池的像素重置动作开始。对于永久曝光相机，即没有(或不使用)电子快门的相机，曝光阶段在完成前一曝光阶段之后立即开始。

对于所有类型的相机，曝光阶段以“像素转移”动作结束。光电池中积累的电荷被转移到存储区，以供进一步读出。这个动作清除光电池，新的电荷整合立即开始。

具有电子快门的相机能够异步重置像素，并根据请求启动新曝光。这些相机被称为异步重置相机。

具有控制曝光开始时间(像素复位)和曝光结束时间(像素转移)的能力，其可完全控制：

- 每次图像捕获的定时
- 通过选择曝光时间控制成像传感器的灵敏度

读出阶段

读出阶段是指在一段时间内，每个像素累积的电荷总量被测量，并传送到成像传感器输出端。

读出阶段不受帧捕获器的控制：

- 它在每次像素转移后自动启动。
- 它的持续时间是固定的；它由要传输的像素数据量和传感器的读出结构(一个或多个抽头，抽头输出数据速率)确定。

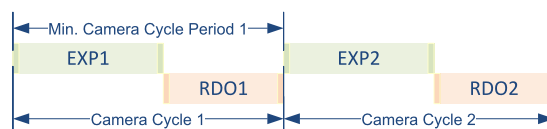
一些传感器提供选择一个或多个感兴趣区域(ROI)的能力，从而加快读出速度，因为需要传输的数据较少。

5.2. 相机循环串联规则

本主题解释了当向异步复位摄像机请求连续相机循环时，帧捕获器必须遵守的规则，以避免相机触发器溢出。

相机不允许重叠的规则

- 在读出阶段完成之前，下一个相机循环可能不会开始。



(不允许循环重叠) 相机可实现的最短循环周期

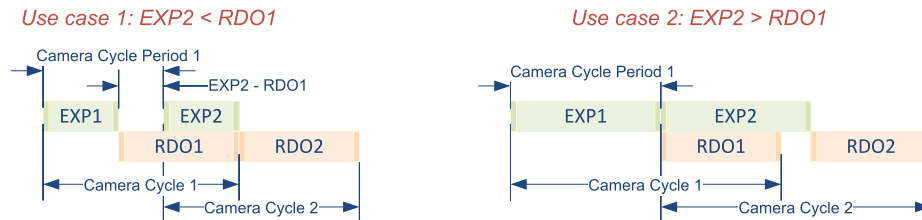
$$\text{最小循环周期}_n = \text{EXP}_n + \text{RDO}_n$$



备注 只有少数工业相机不允许循环重叠！

允许重叠的相机规则

1. 两个连续相机循环的曝光阶段可能永远不会重叠。
2. 两个连续相机循环的读出阶段可能永远不会重叠。



允许循环重叠的相机可实现的最短循环周期

在第一种情况下，第二循环的曝光阶段的持续时间短于第一循环的读出阶段的持续时间。曝光阶段完成后，下一个相机循环可能会开始($EXP_{n+1}-RDO_n$)一段时间。最小循环周期为

$$\text{最小循环周期}_n = EXP_n + RDO_n - EXP_{n+1}$$

在第二种情况下，第二循环的曝光阶段的持续时间长于第一循环的读出阶段的持续时间。下一个相机循环可以在曝光阶段完成后立即开始。最小循环周期为：

$$\text{最小循环周期}_n = EXP_n$$



备注 在机器视觉行业中使用的大多数异步重置相机，都支持相机循环的重叠！

5.3. 相机控制方法

Coaxlink帧捕获器提供了四种相机控制方法，分别是 *NC*, *RC*, *RG* 和 *EXTERNAL*。

NC 相机控制方法

NC相机控制方法以不受帧捕获器控制的相机为目标。这包括

- 不使用任何外部触发信号的自由运行相机
- 使用帧捕获器未传递的外部触发信号异步重置相机。

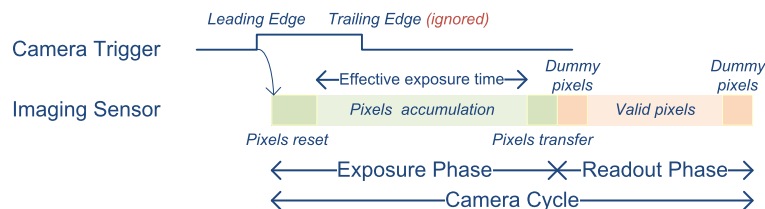


备注 Coaxlink的相机和照明控制器(CIC)未被使用。帧捕获器无法实时控制相机。Coaxlink没有产生相机触发信号。照明没有实时控制。Coaxlink没有产生频闪信号。

RC相机控制方法

RC相机控制方法以异步复位相机为目标，其中只有相机循环速率由帧捕获器控制。曝光持续时间由相机控制。

实时控制通过Coaxlink的相机和照明控制器(CIC)发出的名为“**相机触发器**”的单个上游信号来执行。



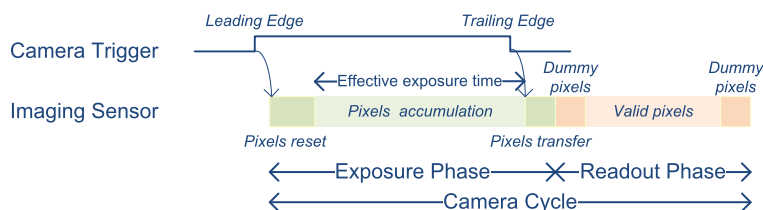
使用RC相机控制方法的捕获器控制的相机循环

CIC在每个相机循环产生一个 **相机触发** 脉冲。该**相机触发** 前沿，触发新的相机循环，并启动新的曝光周期。相机忽略 **相机触发** 后沿。

RG相机控制方法

RG相机控制方法针对异步复位相机，其中相机循环速率和曝光持续时间都由帧捕获器控制。

实时控制是通过Coaxlink的相机和照明控制器(CIC)发出的名为**相机触发器**的单个上游信号来执行的。



使用RG相机控制方法的捕获器控制的相机循环

CIC在每个相机循环产生一个相机触发脉冲。该**相机触发**前沿，触发新的相机循环，并启动新的曝光周期。**相机触发**后沿终止曝光周期，并触发读出。

外部相机控制方法

外部相机控制方法以不受帧捕获器控制的相机为目标。这包括

- 不使用任何外部触发信号的自由运行相机
- 使用帧捕获器未传递的外部触发信号异步重置相机。

外部相机控制方法以异步复位相机为目标，该异步复位相机，由外部控制器施加到捕获器的任何GPIO输入端口的硬件信号控制。



备注 Coaxlink的相机和照明控制器(CIC)未被使用。帧捕获器无法实时控制相机。Coaxlink没有产生相机触发信号。照明没有实时控制。Coaxlink没有产生频闪信号。



警告 外部控制器完全负责相机循环计时！

6. 照明控制原理

6.1. 照明装置	82
6.2. 对齐相机和照明循环	83

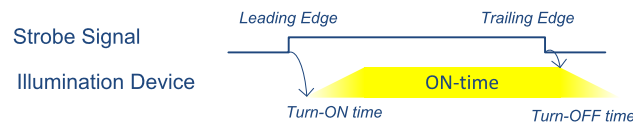
6.1. 照明装置

照明控制器可控制两类照明设备：

- 间歇照明装置
- 频闪式照明装置

间歇照明装置

该照明设备类包括开关光源，其中打开和关闭时间由频闪信号的前沿和下降沿控制。



间歇照明装置的时序图

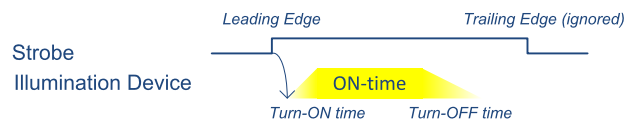
频闪脉冲的宽度决定了光源的开启持续时间。



备注 配置照明控制器时，需要考虑开启时间和关闭时间！

频闪式照明装置

此类照明设备包括开关光源，其中只有接通时间由频闪信号的前沿控制。



频闪式照明装置时序图

开启持续时间由照明设备本身控制或控制。

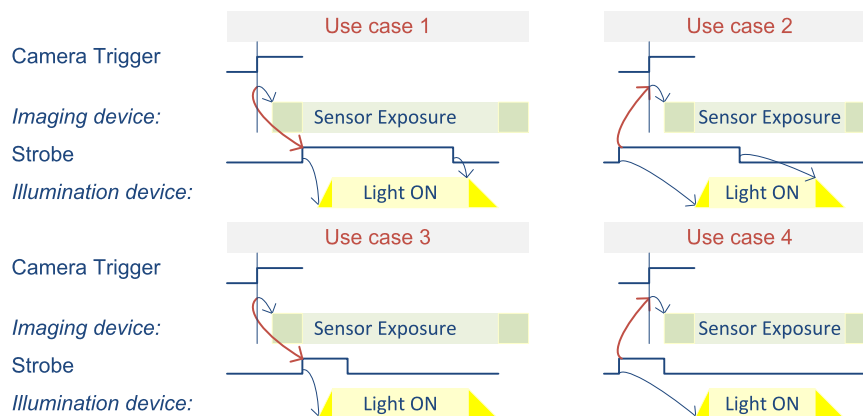


备注 配置照明控制器时，需要考虑开启时间和开启持续时间。

6.2. 对齐相机和照明循环

显然，光源的开启时间必须与成像传感器的曝光相位一致。

因此，必须充分控制频闪信号和**相机触发**信号之间的时间关系。



相机触发器与频闪对准的4个典型使用案例

间歇性光源(用例1和2)

必须充分控制频闪脉冲的持续时间，以提供适当的光照量并获得正确的曝光图像。

应适当地定时传感器曝光，以便在光源关闭后终止传感器曝光。

频闪光源(用例3和4)

应适当地定时传感器曝光，以便在光源关闭后终止传感器曝光。

延迟频闪(用例1和3)

频闪信号的前沿(开始)稍有延迟，以确保在成像设备重置像素时，光源不会过早打开。

早期频闪(用例2和4)

相机触发信号的前沿稍有延迟，以确保传感器的曝光时间尽可能短，并与开启时间紧密匹配。

7. 相机和照明控制器

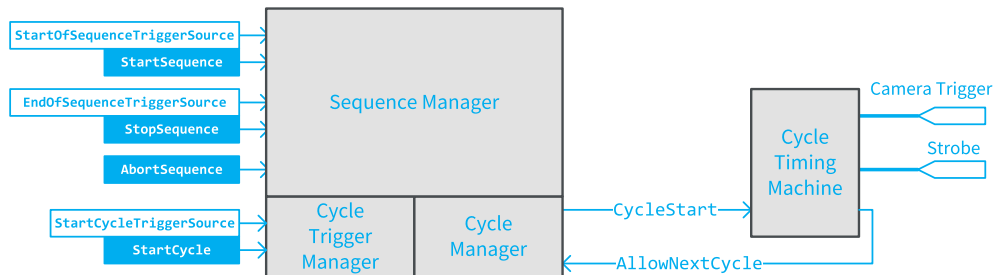
7.1. CIC框图	85
7.2. 循环定时机	86
7.3. 循环管理器	88
7.4. 循环触发管理器	89
7.5. 序列管理器	91
7.6. CIC输出信号路由	93
7.7. CIC时序图	94
循环时序图	95
连续重叠CIC循环	97
循环顺序时序图	102

7.1. CIC框图

相机和照明控制器(缩写为CIC)控制一台相机及其相关照明。

相机由**相机触发**信号控制,照明设备由**频闪**信号控制。

另见: "CIC输出信号路由" 于页面93



相机和照明控制器框图

CIC由4个主要的互联区块组成:

- 循环计时机负责生成精确计时的的事件和信号,构成一个相机和照明控制器循环(CIC循环),即:**相机触发**和**频闪**。

另见: "循环定时机" 在下一页

- 循环管理器负责**CycleStart**事件的生成。当开始循环条件不全部满足并且循环定时机器不允许开始新的循环时,它会防止开始新的循环。

另见: "循环管理器" 于页面88

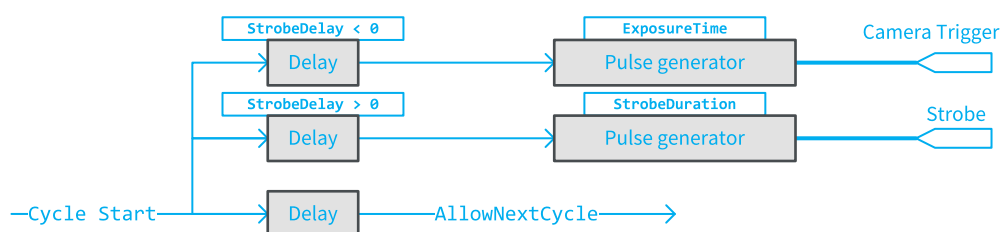
- 循环触发管理器负责(与循环管理器和序列管理器合作)详细说明有效的**CycleStart**事件(该事件启动了循环计时机一个循环)

另见: "循环触发管理器" 于页面89

- 序列管理器根据用户定义的开始序列和停止序列条件,来管理CIC循环序列。

另见: "序列管理器" 于页面91

7.2. 循环定时机



CIC循环定时机框图

CIC定时机，负责生成精确定时事件和信号，其构成一个相机和照明控制器循环(CIC循环)。

在每次发生**Cycle Start**事件时，定时机会生成：

- 在**Camera Trigger**信号上的单脉冲。
- 在**Strobe**信号上的单脉冲。
- 一个**AllowNextCycle**事件

循环内定时

设备模块的三个GenICam功能，用于配置一个循环内输出信号的定时：

- ExposureTime 定义 **Camera Trigger** 脉冲的持续时间。
- StrobeDuration 定义 **Strobe** 脉冲的持续时间。
- StrobeDelay 定义从相机的 **Camera Trigger** 前沿到 **Strobe** 前沿的时间偏移。

另见：有关更多说明和时序图，请参考“循环时序图”于页面95。

循环对循环定时

循环管理器使用 **AllowNextCycle** 事件来确定下一个循环何时开始。

AllowNextCycle 事件的位置不是由用户直接设置的。相反，Coaxlink 驱动程序会根据以下用户设置，对其进行评估：

- 相机模型类别的 `ExposureReadoutOverlap` 功能，定义了相机是否支持曝光/读出重叠。如果允许重叠，`ExposureReadoutOverlap` 事件会更早发布，并获得更快的周期率。
- 该 `ExposeRecovery` feature of the Camera Model category defines the minimum time gap required by the camera between two exposures. This feature is relevant when `ExposureReadoutOverlap = TRUE` 和曝光阶段的持续时间变得大于读数阶段的持续时间。
- “循环控制”类别的 `CycleMinimumPeriod`，定义了最小循环周期。该值不得小于相机执行图像读出而所需的时间！



备注 一些相机在图像数据路径中有数据存储器。这使得能够以(比“相机到帧”捕获器数据链路所能承受的)更高的“循环到循环”速率，来捕捉图像突发。在这种情况下，`CycleMinimumPeriod` 声明了图像传感器可以达到的最小“循环到循环”时间！

另见：有关更多说明和时序图，请参考“连续重叠CIC循环”于页面97。

7.3. 循环管理器

循环管理器负责 **Cycle Start** 事件的生成。

当以下列出的 *循环启动条件* 不满足时，它可防止启动新的循环：

序列激活条件

序列管理器必须处于激活状态：序列已经开始和序列停止条件尚未达到。

此条件始终适用。

下一循环允许条件

“循环计时机”已经发出上一循环的 **Allow Next Cycle** 事件。

此条件始终适用。

空闲内存条件

主板上有足够的空闲内存来获取下一个循环的图像数据。

此条件始终适用。

循环触发事件条件

需要新的循环触发事件来启动新的循环。

仅当 **Cycle TriggerSource** \neq **Immediate** 和 **CycleMaxPendingTriggerCount** = **0** 时，该条件适用。

待定触发条件

至少有一个待定触发(可能是新的触发)等待服务。

仅当 **Cycle TriggerSource** \neq **Immediate** 和 **CycleMaxPendingTriggerCount** > **0** 时，该条件适用。

循环触发事件排序

当 **CycleTriggerSource** 设置为 **StartCycle** 或 **<any I/O toolbox event source>**时，每个循环触发事件都会根据 **触发接受标准** 进行评估，并根据结果进行排序。

被拒绝的循环触发事件，会增加**循环丢失触发计数**。

如果待定触发不能立即服务，被接受的循环触发事件则会增加**循环待定触发计数**。

触发接受标准

当满足以下两个条件时，将**立即接受并执行**循环触发事件。

- 循环序列处于活动状态
- 当前循环管理器正等待立即触发事件，以启动新的循环(接受立即触发)

当满足以下条件时，循环触发事件将**被接受并在以后执行**。

- 循环序列处于活动状态。
- 待定触发数量**CycleMaxPendingTriggerCount**，少于**CycleMaxPendingTriggerCount**。
- 循环管理器尚未准备好启动新的循环。

7.5. 序列管理器

序列管理器是CIC的顶级管理器：它控制循环触发管理器和循环管理器。

它根据用户定义的启动序列和停止序列条件，来定义相同CIC循环的序列。

启动序列

启动序列的条件由 **StartOfSequenceTriggerSource** 定义。

当 **StartOfSequenceTriggerSource** 设置为 **Immediate** (默认设置) 时, 序列管理器不需要任何进一步的操作来允许循环管理器和循环触发管理器继续进行第一个循环。

根据循环管理器的 **CycleTriggerSource** 设置, 将执行第一个循环:

- 立即执行, 当 **CycleTriggerSource** 设置为 **CycleTriggerSource**。
- 执行 **StartCycle** 命令 (当 **CycleTriggerSource** 设置为 **StartCycle** 或)
- 执行 **StartCycle** 命令或在 **CycleTriggerSource** 指定的 I/O 工具箱事件源上发生事件时。

当 **StartOfSequenceTriggerSource** 设置为 **StartSequence**, 序列管理器等待 **StartSequence** 命令的执行, 然后允许循环管理器和循环触发管理器继续进行第一个循环。

当 **StartOfSequenceTriggerSource** 设置为 **<any-event-source>** 时, 序列管理器等待 **StartSequence** 命令的执行或等待指定事件源上 I/O 工具箱事件的发生, 然后允许循环管理器和循环触发器管理器继续进行第一个循环。

停止序列

停止序列的条件由 **EndOfSequenceTriggerSource** 定义。

当 **EndOfSequenceTriggerSource** 设置为 **StopSequence** (默认设置), 序列管理器会在 **StopSequence** 命令执行后的下一个循环边界停止序列。

当 **EndOfSequenceTriggerSource** 设置为 **SequenceLength**, 序列管理器会在执行了由 **SequenceLength** 指定的多个相机循环后自动停止序列。该序列可以在执行 **StopSequence** 命令时, 被预期地停止。默认 **SequenceLength** 值是 1; 允许任何高达 16,777,215 的值。

当 **EndOfSequenceTriggerSource** 设置为 **<any-event-source>**, 序列管理器等待 **StopSequence** 命令的执行或指定事件源上 I/O 工具箱事件的发生, 然后在下一个循环边界停止序列。



备注 **StartOfSequenceTriggerSource** 和 **EndOfSequenceTriggerSource** 设置的任何组合是允许的。

相机抓取时更改序列长度。

从版本 10.5 开始, 如果 **SequenceLength** 在“序列开始”和“序列结束”事件之间改变, 新值将对后续序列有效。



备注 该 **SequenceLength** 的值锁存在“序列开始”事件上。

7.6. CIC输出信号路由

相机触发信号

帧捕获器通过 **Camera Trigger** 信号控制异步复位相机的相机循环。

信号可使用以下媒质从Coaxlink卡传输到相机：

- CoaXPress链接的I/O通道。
- 由TTL I/O驱动的专用线路

CoaXPress I/O 通道

Camera Trigger信号作为高优先级“主机到设备触发”消息在CoaXPress I/O通道上传输到相机

另见：有关“主机到设备触发”发射器的详细信息，请参考“CoaXPress主机到设备触发”于页面38。

TTL I/O线

任何TTL I/O线都可以配置为**Camera Trigger**输出。I/O控制块的极性控制，为每个I/O端口提供单独的极性控制。TTL I/O线路的I/O控制块的模式，控制提供了单独的输出驱动器配置。

频闪信号

I/O线

每条有输出能力的I/O线，都可以配置为“照明频闪”输出。I/O控制块的极性控制，为每个I/O端口提供单独的极性控制。TTL I/O线路的I/O控制块的模式控制，为用作频闪输出的每个I/O端口提供了单独的输出驱动器配置。

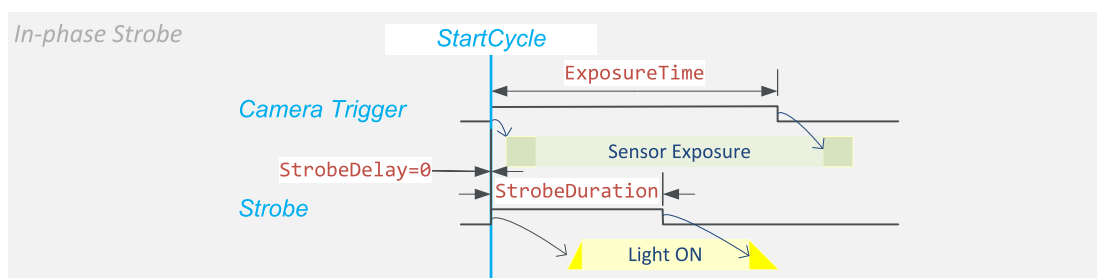
7.7. CIC时序图

循环时序图	95
连续重叠CIC循环	97
循环顺序时序图	102

循环时序图

本主题展示了单个CIC循环的时序图并说明了 ExposureTime, StrobeDuration and StrobeDelay features for 3 use cases corresponding to positive, zero and negative values of StrobeDelay.

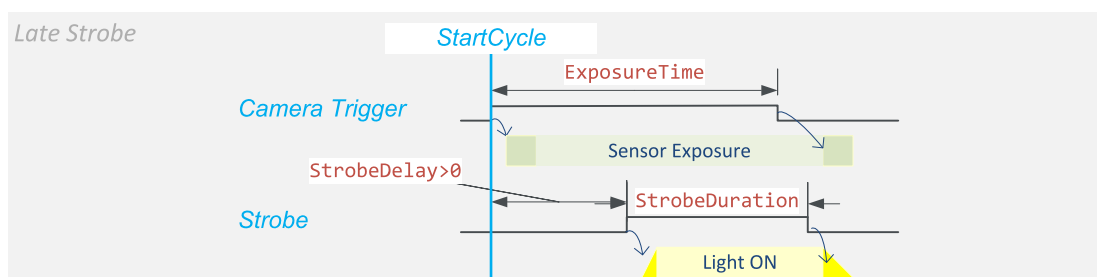
同相频闪情形



StrobeDelay = 0

Camera Trigger和 Camera Trigger信号立即变高(在 Camera Trigger事件之后)。

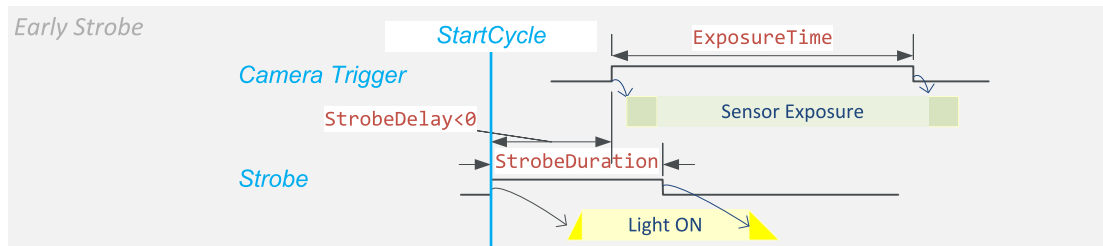
延迟频闪情形



StrobeDelay > 0

该 Camera Trigger 信号立即变高(在 StartCycle 事件之后)以及频闪信号立即变高 (StrobeDelay 几微秒之后)。

早期频闪情形



StrobeDelay < 0

频闪信号立即变高(在 **StartCycle** 事件之后) 以及 **Camera Trigger** 信号变高(在 **StrobeDelay** 微秒相反值的延迟之后)。

连续重叠CIC循环

简介

本主题说明了用户该如何设置 `ExposureRecovery` and `TargetFramePeriod` features when `ExposureReadoutOverlap = True` 以正确配置CIC, 从而避免触发溢出并且最大化循环速率。

Coaxlink驱动程序根据用户定义的设置 `ExposureTime`, `ExposureRecovery` and `TargetFramePeriod`, 通过搜索满足以下条件的最小值来计算相机循环值的持续时间:

- **条件1:** 连续 **Camera Trigger** 脉冲之间的时间间隔 (r - in the drawings) must be greater or equal to the `ExposureRecovery` 设置。这能够确保 **Camera Trigger** 正确地流经触发传输链路。这也确保了新的曝光不会在前一次曝光完成之前开始。
- **条件2:** CIC循环持续时间 (a 在图纸中) 必须足够长, 以确保在上一次读数完成之前, 不会开始新的读数。
- **条件3:** CIC循环持续时间必须足够长, 足以包括 **Camera Trigger** 和频闪信号的转换。

“读出受限”用例, 说明了循环周期等于读出阶段持续时间的情况。

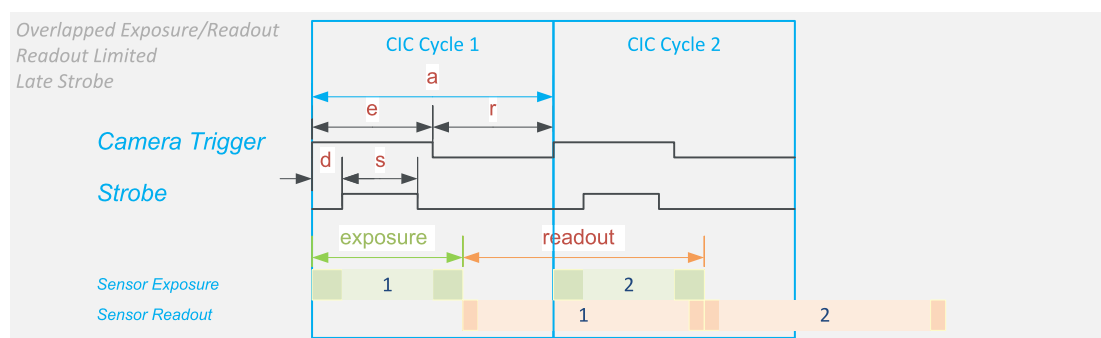
“曝光受限”用例, 说明了循环周期等于曝光阶段持续时间的情况。

在下面的时序图中:

e是 **Camera Trigger** 脉冲宽度。该值由 **Camera Trigger** Camera Trigger定义。

d是 **Strobe** 延迟。该值由 **Strobe** Strobe定义。

情形1: 读出受限(延迟频闪)



相机循环速率仅受相机读出时间的限制

当 e (= Exposure Time) 明显小于读出持续时间时, 就会出现这种情况。

在这种情况下:

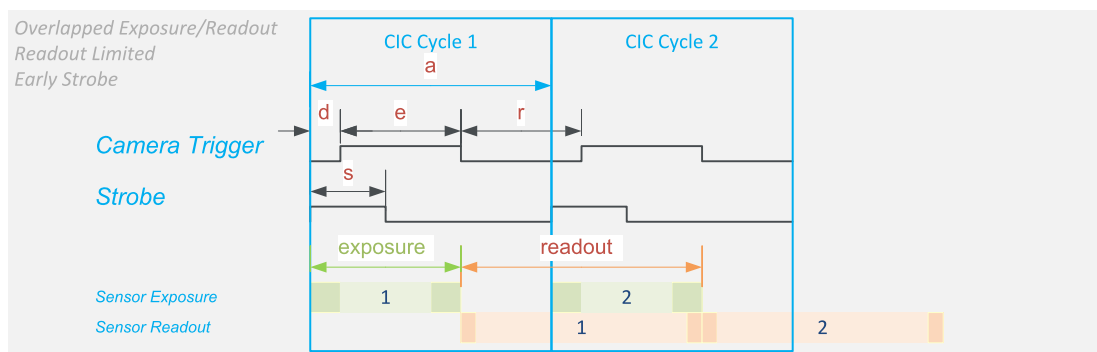
- r 可能会更大ExposureRecovery: **条件1**如果得以满足的话。
- 频闪脉冲位于**Camera Trigger**脉冲的“内部”: **条件3** 变得无关紧要(当**条件1**得以满足时)。
- **条件2**是Coaxlink驱动程序计算循环持续时间的唯一条件。

循环的最佳持续时间, 等于传感器读出阶段的有效持续时间。这是在用户将设置为对应于读出持续时间的值时获得的。`TargetFramePeriod`



备注 读出持续时间, 可以从相机数据表的最大帧速率规格或实验中得出。

情形2: 读出受限(早期频闪)



相机循环率仅受相机读出时间的限制(尽管有早期频闪)

这种情况类似于情形1。它显示，尽管有早期频闪，但仍有可能达到相机的最大循环速率。

当 e (= **ExposureTime**) 明显小于读出持续时间时，就会出现这种情况。

在这种情况下：

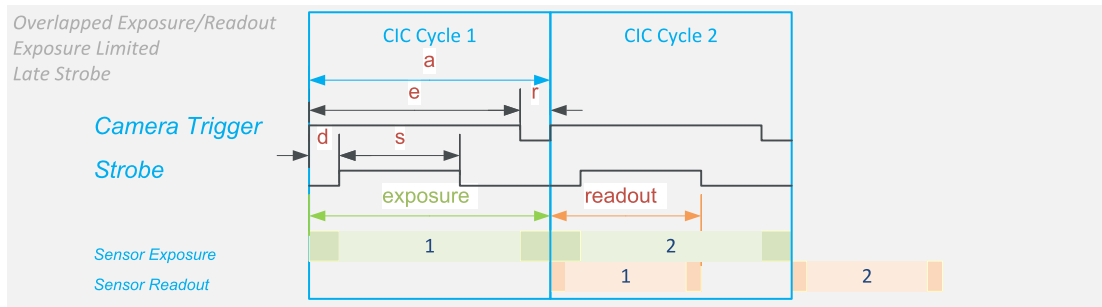
- r 可能会更大 **ExposureRecovery**: **条件1**如果得以满足的话。
- 频闪脉冲在 **Camera Trigger** 脉冲之前终止：**条件3**得以满足的条件是 r is greater than d . This is the case when $(d + e < \text{读出持续时间})$ 。
- **条件2**是Coaxlink驱动程序计算循环持续时间的唯一条件。

循环的最佳持续时间，等于传感器读出阶段的有效持续时间。这是在用户将 **TargetFramePeriod** 设置为对应于读出持续时间的值时获得的。



备注 读出持续时间，可以从相机数据表的最大帧速率规格或实验中得出。

情形3: 曝光受限(延迟频闪)



相机循环速率，受曝光时间设置的限制

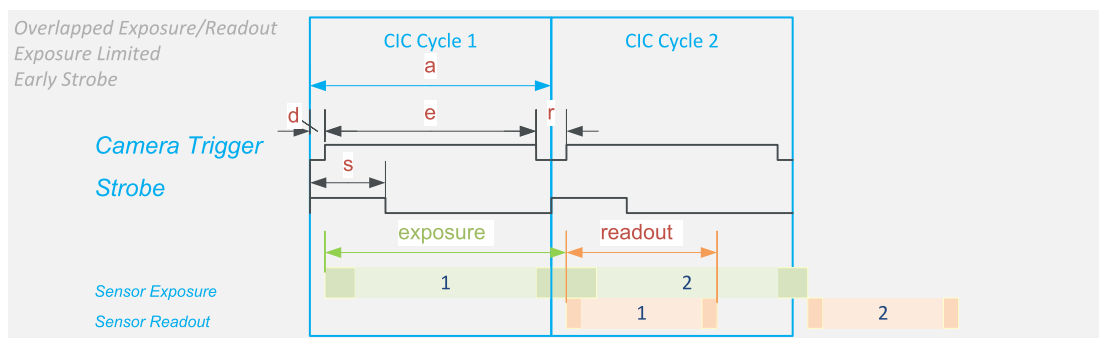
当 e (= Exposure Time) 明显大于读出持续时间时，就会出现这种情况。

在这种情况下：

- 所有循环相同(读出持续时间小于曝光持续时间)意味着**条件2**变得无关紧要。
- 频闪脉冲位于**Camera Trigger**脉冲的“内部”：**条件3**变得无关紧要(当**条件1**得以满足时)。
- **条件1**是Coaxlink驱动程序计算循环持续时间的唯一条件。

循环的最佳持续时间，等于曝光阶段的有效持续时间。当用户将 **ExposeRecovery** 设置为对应于相机允许的连续 **ExposeRecovery** 脉冲之间最小时间间隔的值时，可以获得该值。

情形4 :曝光受限(有早期频闪)



相机循环速率受曝光时间设置的限制(尽管有早期频闪)

这种情况类似于情形3。它表明，尽管有早期频闪脉冲，但在较小负 **StrobeDelay** 值的情况下，仍有可能达到相同的循环速率。

当 e (= **ExposureTime**) 明显大于读出持续时间时，就会出现这种情况。

在这种情况下：

- 所有循环相同,(读出持续时间小于曝光持续时间)意味着 **条件2** 变得无关紧要。
- 频闪脉冲在 **Camera Trigger** 脉冲之前终结: **条件3** 变得无关紧要(当 **条件1** 得以满足且 **Camera Trigger**。
- **条件3** 和 **条件1** 是 Coaxlink 驱动程序计算循环持续时间的唯一条件。

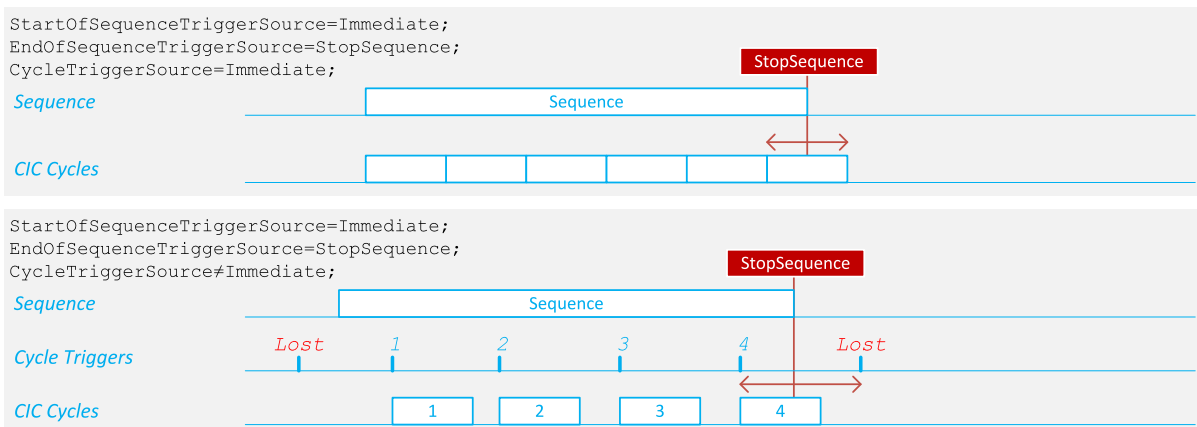
用户必须将 **ExposeRecovery** 设置为对应于以下两个值中最大值的值：

- 相机在连续 **Camera Trigger** 脉冲之间允许的最小时间间隔。
- **StrobeDelay** 的相反值。

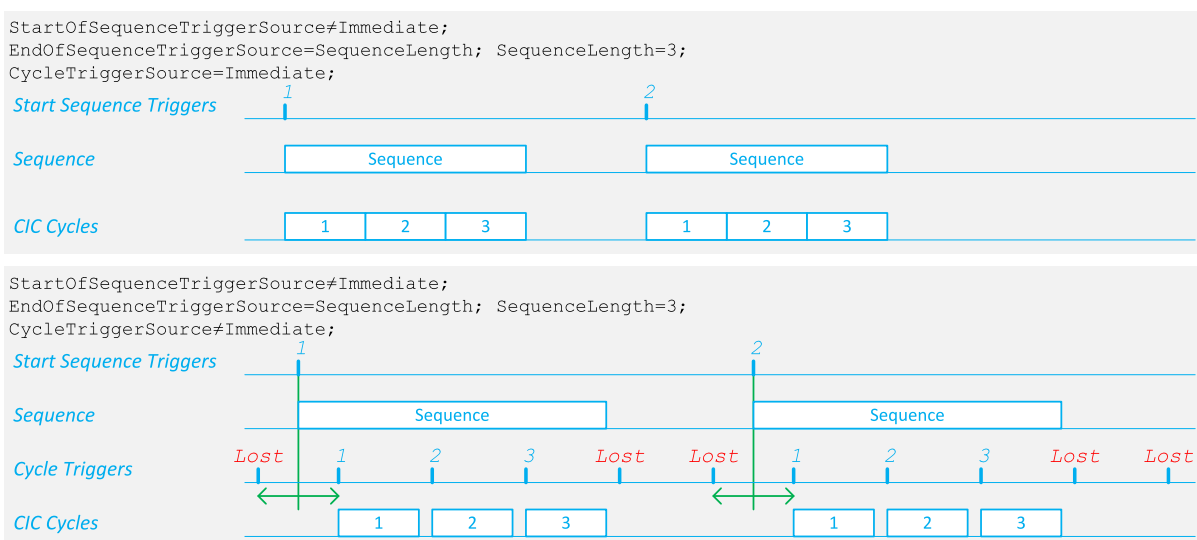


备注 当 **CycleTriggerSource** = **Immediate** 时，可以通过向其分配更大的值到 **TargetFramePeriod**，以此将循环速率降低到期望的速率。

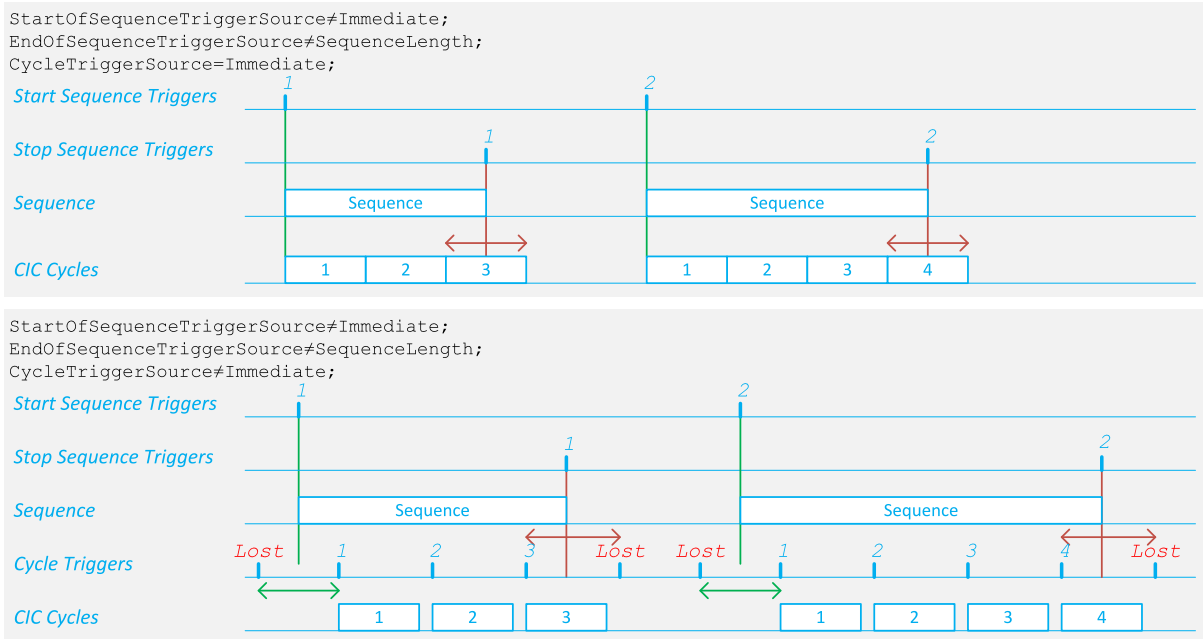
循环顺序时序图



即刻启动的循环序列



具有触发启动和固定长度的循环序列



具有触发启动和触发结束的循环序列

8. 通用I/O

8.1. I/O线路概述	105
8.2. I/O线路概述	108
8.3. I/O控制块	109
8.4. 线格式和线模式控制	111
8.5. 线路极性控制	115
8.6. 滤波器控制	116
8.7. 线源选择	117
8.8. 逻辑I/O线路状态	118
8.9. 实际I/O线路状态	119
8.10. 线路驱动器实际输出状态	120
8.11. 初始状态	121

8.1. I/O线路概述

每个产品的I/O线

产品	标准I/O组		模块I/O组	
	#1	#2		
1629 Coaxlink Duo PCIe/104-EMB	3300	-		
1630 Coaxlink Mono	单板	-	-	
1631 Coaxlink Duo	单板	单板	-	
1632 Coaxlink Quad	单板	单板	-	
1633 Coaxlink Quad G3	单板	单板	-	
1633-LH Coaxlink Quad G3 LH	单板	单板	-	
1634 Coaxlink Duo PCIe/104-MIL	3300	-		
1635 Coaxlink Quad G3 DF	单板	-	-	
1637 Coaxlink Quad 3D-LLE	单板	单板	-	
3602 Coaxlink Octo	单板	3614	3610	3612
3603 Coaxlink Quad CXP-12	单板	单板	3610	3612
3620 Coaxlink Quad CXP-12 JPEG	单板	单板	3610	3612

I/O组

标准I/O组 #1

I/O线路组1由10个I/O组成：

I/O 线类型	I/O线名称	计数
隔离输入	IIN11, IIN12, IIN13, IIN14	4
隔离输出	IOUT11, IOUT12	2
RS-422 输入	DIN11, DIN12	2
TTL I/O	TTLIO11, TTLIO12	2

标准I/O组 #2

I/O线路组2由10个I/O组成：

I/O 线类型	I/O线名称	计数
隔离输入	IIN21, IIN22, IIN23, IIN24	4
隔离输出	IOUT21, IOUT22	2

I/O 线类型	I/O线名称	计数
LVTTL 输入/输出	TTLIO21, TTLIO22	2
RS-422 输入	DIN21, DIN22	2

模块I/O组

模块I/O组，由多达20条单端或10条差分外I/O线组成。

I/O 线类型	I/O线名称	计数
RS-422 输入	MIO1, MIO3...MIO19	0到10
RS-422 输出	MIO1, MIO3...MIO19	0到10
TTL输入	MIO1, MIO2...MIO20	0到20
TTL 输出(3610) CMOS 输出(3612)	MIO1, MIO2...MIO20	0到20

I/O模块

3300 I/O 模块

该 **3300 用于CoaxLink Duo PCIe/104的 HD26F I/O模块** 提供了 "标准I/O组#1" 在上一页。

另见： [3300/3302 Accessories for Coaxlink Duo PCIe/104](#) 有关更多信息，请参考Coaxlink PCIe/104硬件手册。

3610/3612 I/O 模块

该 **3610 HD26F I/O扩展模块TTL-RS422** 和 **3612 HD26F I/O扩展模块TTL-CMOS5V-RS422** 提供了 "模块I/O组" 高于。

另见： 有关更多信息，请参阅硬件手册 [3610/3612 I/O Extension Modules](#)。

3614 I/O 模块

该 **3614 HD26F I/O Extension Module - Standard I/O Set** 提供了 "标准I/O组#2" 在上一页。

另见： 有关更多信息，请参阅硬件手册 [3614 I/O Extension Module](#)。

I/O 线类型

I/O 线类型	电气风格	频率范围	名字前缀
CMOS 输出	5伏CMOS单端输出	高达1兆赫	MIO
隔离输入	符合5 V、12 V和24 V信号电平的隔离电流感应输入	高达50千赫	IIN
隔离输出	隔离的100 mA/30V触点输出	高达50千赫	IOUT
RS-422 输入	RS-422差分输出	高达10兆赫	DIN
		高达10兆赫	MIO
RS-422 输出	RS-422差分输出	高达10兆赫	MIO
TTL I/O	5伏容限LVTTTL单端输入和 3.3V LVTTTL图腾极单端输出或 3.3V开放式发射极单端输出或 3.3V集电极开路单端输出	高达10兆赫	TTLIO
TTL输入	5伏容限LVTTTL单端输入	高达1兆赫	MIO
TTL 输出	3.3 V LVTTTL单端输出	高达1兆赫	MIO

8.2. I/O线路概述

输入线

可输入的I/O线(DIN、TTLIO、IN、MIO) 可用作：

- 通用输入：一种输入信号，其状态可由主机应用程序读取或监控。
- 运动编码器输入：由正交运动编码器传递的一对输入信号，用于从相机触发采集。
- 触发输入：用于触发相机采集的输入信号

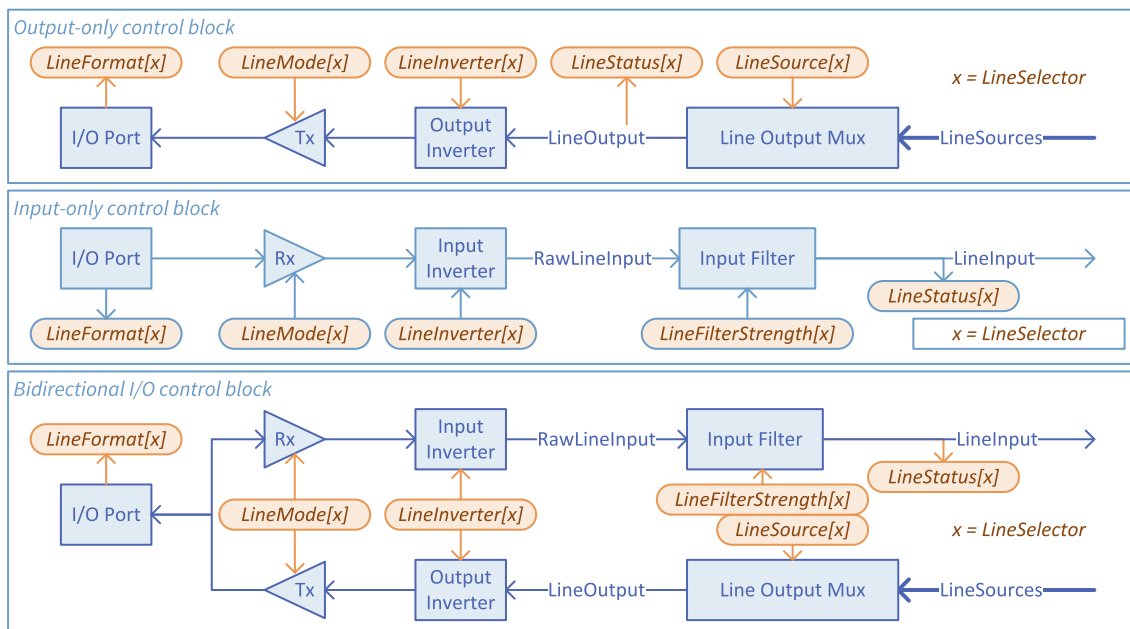
输出线

可输出的I/O线(TTLIO、IOUT、MIO) 可用作：

- 通用输出：一种输出信号，其状态可由主机应用程序设置。
- 频闪输出：通常用来控制闪光灯的输出信号，与相机同步。
- 相机触发输出(仅在TTLIO上可用)：由Coaxlink 卡生成的用于触发相机的输出信号。

8.3. I/O控制块

每条I/O线都通过一个I/O控制块进行控制。根据 I/O线的输入/输出能力，有3种类型的控制块：



在上图中：

- 一条细蓝线代表一条单独的电信号路径。
- 一条粗蓝线表示电信号路径的集合。
- 蓝色箭头表示电信号的传播方向。
- 蓝色形状表示I/O控制块的功能元件。
- 橙色形状(椭圆形)表示GenICam功能；其中的文本是功能名称
- 附加到功能名称的[X]，表示该功能与选择器功能关联(在本例中为：lineselector)
- 橙色箭头表示功能的访问模式：具有传入箭头的只读功能，具有传出箭头的可写功能。

输入路径

仅输入和双向I/O线路共享一个公共输入路径结构，包括：

- 代表I/O连接器上I/O引脚的I/O端口块
- 代表线路接收器电路的RX块
- 代表用户可配置逻辑反相器的输入反相器块
- 代表用户可配置故障排除滤波器的输入滤波器块

输出路径

仅输出和双向I/O线路共享一个公共输出路径结构，包括：

- 代表I/O连接器上I/O引脚的I/O端口块
- 代表线路驱动电路的Tx块
- 代表用户可配置逻辑反相器的输出反相器块
- 代表用户可配置源多路复用器的线路输出MUX块

8.4. 线格式和线模式控制

简介

下表汇总了每条I/O线的I/O控制块的详细信息：

- 第一列标示了 **LineSelector** 值
- 第二列指明了由 **LineStatusAll** 报告的整数中的位位置
- 第三列指明了由 **LineFormat** 报告的值。
- 第四列指明了可以分配给 **LineMode** 的值。

标准I/O组

标准I/O组 #1

LineSelector	Bit#	LineFormat	LineMode
DIN11	0	DIFF	输入
DIN12	1	DIFF	输入
IIN11	4	ISO	输入
IIN12	5	ISO	输入
IIN13	6	ISO	输入
IIN14	7	ISO	输入
IOUT11	12	ISO	输出
IOUT12	13	ISO	输出
TTLIO11	16	TTL	Input, Output, DriveLow 或 DriveHigh
TTLIO12	17	TTL	Input, Output, DriveLow 或 DriveHigh

标准I/O组 #2

LineSelector	Bit#	LineFormat	LineMode
DIN21	1	DIFF	输入
DIN22	3	DIFF	输入
IIN21	8	ISO	输入
IIN22	9	ISO	输入
IIN23	1	ISO	输入
IIN24	11	ISO	输入
IOUT21	14	ISO	输出
IOUT22	15	ISO	输出
TTLIO21	18	TTL	Input, Output, DriveLow 或 DriveHigh
TTLIO22	19	TTL	Input, Output, DriveLow 或 DriveHigh

TTLIO端口模式控制

该**LineMode**功能控制每个单独TTLIO端口的方向和线路驱动模式。可以随时选择四种模式：

- **Input**: 仅输入，图腾柱驱动程序禁用(默认启动设置)，
- **Output**: 能够驱动低和高的图腾柱驱动器，
- **DriveLow**: 仅能驱动低的集电极开路驱动器，
- **Drivehigh**: 仅能驱动高的开放式发射器驱动器，



备注 两种最新配置允许有线-AND配置。可以通过输入端口，回读线路状态。

模块I/O组

3610 HD26F I/O扩展模块TTL-RS422

LineSelector	Bit#	LineFormat	LineMode
MIO1	20	TTL 或 DIFF	Input 或 Output
MIO2	21	TTL	Input 或 Output
:	:	:	:
MIO19	39	TTL 或 DIFF	Input 或 Output
MIO20	40	TTL	Input 或 Output

3612 HD26F I/O扩展模块TTL-CMOS5V-RS422

LineSelector	Bit#	LineFormat	LineMode
MIO1	20	TTL	输入
		CMOS	输出
		DIFF	Input 或 Output
MIO19	21	TTL	输入
		CMOS	输出
MIO1	:	:	:
MIO2	39	TTL	输入
		CMOS	输出
		DIFF	Input 或 Output
MIO20	40	TTL	输入
		CMOS	输出

MIO格式和模式控制

该**LineFormat**功能控制MIO端口的电气风格。可以配置两种格式：

- **TTL**: 单端 (TTL或CMOS)
- **DIFF**: 差分 (RS422)

该**LineMode**功能控制MIO端口的方向。可以配置两种模式：

- **Input**: 仅输入，
- **Output**: 能够驱动低和高的图腾柱驱动器，



备注 只能在模块配置期间更改控件。

另见：有关配置的详细说明，请参阅 [3610/3612 I/O Extension Modules](#)。

8.5. 线路极性控制

所有I/O线都装有极性控制装置。对于双向I/O线路，单个控制对两条路径的影响相等。

线路极性由用户通过 **LineInverter** 控制进行配置。



备注 请用户根据外部信号的极性，设置极性控制，使输入路径的线路输入信号和输出路径的 *LineSources* 信号始终使用正逻辑。

8.6. 滤波器控制

所有I/O输入线路都配有故障排除滤波器。过滤强度由用户通过**LineFilterStrength**控制进行配置。

强度控制提供从“最低”到“最高”的5个位置。默认位置为“低”。

每个位置对应于3个I/O输入线类型中的每个类型的特定滤波器时间常数。

每个I/O输入线类型的线路滤波器时间常数和线路滤波器强度：

LineFilterStrength	差分输入 (DIN)	TTL输入 (TTLIO)	隔离输入 (IIN)
最低的	50 ns	50 ns	500 ns
低	100 ns	100 ns	1 μ s
中	200 ns	200 ns	2 μ s
高	500 ns	500 ns	5 μ s
最高的	1 μ s	1 μ s	10 μ s

请用户根据外部信号的质量，设置滤波器强度。选择滤波器强度，使其时间常数为：

- 大于最长故障持续时间
- 大于信号10%~90%的上升/下降时间
- 至少比最小信号脉冲持续时间小2倍

故障清除滤波器会在输入信号路径中引入延迟。当输入信号有干净的转换时，延迟等于滤波器时间常数。如果信号质量不好，延迟可能会显著增加。

8.7. 线源选择

任何具有输出能力的I/O线路都装有一个源信号多路复用器。

源信号多路复用器实现一个完全填充的信号路由矩阵，允许选择内部信号路由到任何输出线路。

选择信号源

要为具有输出能力的I/O线路的线路驱动器，选择用作信号源的内部信号：

步骤1: 通过分配适当的值给 **LineSelector** 功能，来选择I/O线。

步骤2: 分配适当的值给 **SignalSource** 功能。

输出线路路由矩阵

源信号	可能的目的地
用户输出寄存器的任何位	任意输出线
任何设备模块摄像机触发输出	任何TTLIO输出线
任何设备模块频闪输出	任意输出线
稳定低	任意输出线

8.8. 逻辑I/O线路状态

逻辑I/O线路状态

I/O线路的(逻辑)状态, 是I/O控制块电信号的逻辑状态:

- 对于可输入的I/O线路:*LineInput*信号: I/O控制块输入路径中位于输入反相器后面的一个点。
- 对于仅输出I/O线路:*LineOutput*信号, 位于输出反相器之前的I/O控制块输出路径中的一个点。

获取单个I/O线路的状态

1. **步骤1:** 通过分配适当的值给 **LineSelector** 功能, 来选择I/O线
2. **步骤2:** 通过获取 **LineStatus** 功能的值, 直接获取线路状态

获取单个操作中所有I/O线路的状态

获取**LineStatusAll**功能的值。

整数的每个位对应于一个I/O线路。一位对应一个线路逻辑状态(高)。

8.9. 实际I/O线路状态

I/O线路状态的实际状态不仅取决于读取 LineStatus but also on the following I/O block settings: LineFormat, LineMode, and LineInverter时所报告的值，

LineFormat	LineMode	LineInverter/LineStatus	实际I/O线路状态
DIFF	输入	假/假或真/真	$(VIN+ - VIN-) < VThreshold$
		假/真或真/假	$(VIN+ - VIN-) > VThreshold$ 未连接的I/O线
ISO	输入	假/假或真/真	光耦关闭。线路电流 < 1 mA. 线路可能未连接或连接到错误的极性。
		假/真或真/假	光耦打开。线路电流 > 1 mA.
ISO	输出	假/假或真/真	光耦关闭
		假/真或真/假	光耦开启
TTL	输入	假/假或真/真	线路电压 < 0.8 Volt.
		假/真或真/假	线路电压 > 2.0 Volt.
TTL	输出	假/假或真/真	该线路驱动为低。
		假/真或真/假	该线路驱动为高。
TTL	DriveLow	假/假或真/真	该线路驱动为低。
		假/真或真/假	线路电压 > 2.0 Volt.
TTL	DriveHigh	假/假或真/真	线路电压 < 0.8 Volt.
		假/真或真/假	该线路驱动为高。

8.10. 线路驱动器实际输出状态

线路驱动器输出状态，不仅取决于所选线路输出信号的逻辑电平，还取决于以下I/O块设置：LineFormat, LineMode, and LineInverter:

LineFormat	LineMode	LineInverter / LineOutput 逻辑电平	线路驱动器输出状态
ISO	输出	假/低或真/高	光耦开关关闭。
		假/高或真/低	光耦开关打开。
TTL CMOS	输出	假/低或真/高	I/O线驱动为低。
		假/高或真/低	I/O线驱动为高。
TTL	DriveLow	假/低或真/高	I/O线驱动为低。
		假/高或真/低	I/O线未被驱动。
TTL	DriveHigh	假/低或真/高	I/O线未被驱动。
		假/高或真/低	I/O线驱动为高。

8.11. 初始状态

通电时，I/O线处于以下状态：

差分输入

- **LineInverter = False,**
- **LineFilterStrength = Low.**

TTL输入/输出

- **LineMode = Input** (该行未被驱动) ,
- **LineInverter = False,**
- **LineFilterStrength = Low.**

隔离输入

- **LineInverter = False,**
- **LineFilterStrength = Low.**

隔离输出

- **LineInverter = False,**
- **LineFilterStrength = Low,**
- 光耦是 **OFF**。

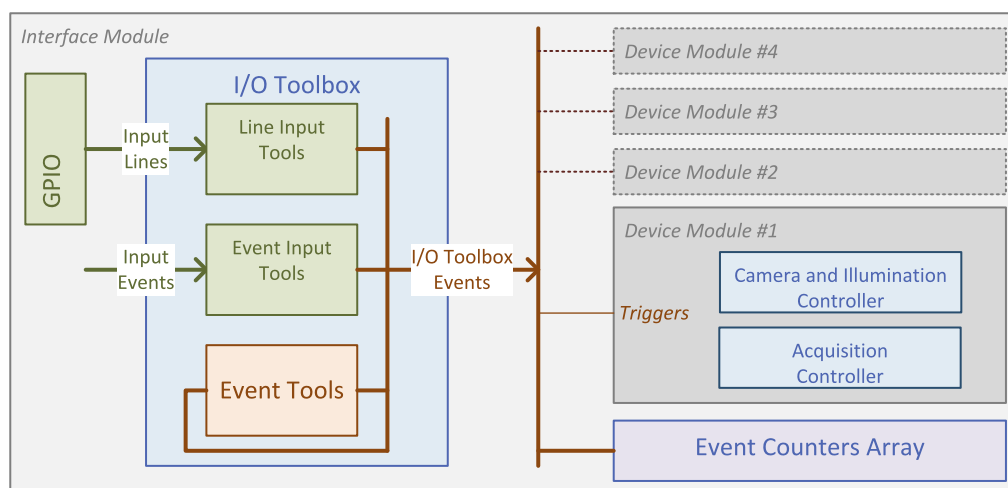
9. I/O 工具箱

9.1. 介绍 I/O 工具箱	123
9.2. I/O 工具箱组合	126
9.3. 行输入工具	129
9.4. 正交解码器工具	130
9.5. 事件输入工具	133
9.6. 分割器工具	134
9.7. 乘法器/分割器工具	135
9.8. 延迟工具	138
9.9. 用户操作工具和用户输出寄存器	140

9.1. 介绍 I/O 工具箱

I/O 工具箱 是一个可配置的工具数组，它从应用于具有输入能力的 I/O 线路的外部信号建立事件脉冲流。

I/O 工具箱上下文



I/O 工具箱上下文框图

每个 CoaxLink 卡只有一个 I/O 工具箱实例。它属于 Gentl 接口模块。

支持输入的 I/O 行 (输入行) 和选定的事件源 (输入事件) 正在输入输入工具 I/O 工具箱。

所有 I/O 工具箱工具都会生成一个 (或多个) 事件流 (I/O 工具箱事件)。

I/O 工具箱事件流，被分发到卡上的各个使用者。

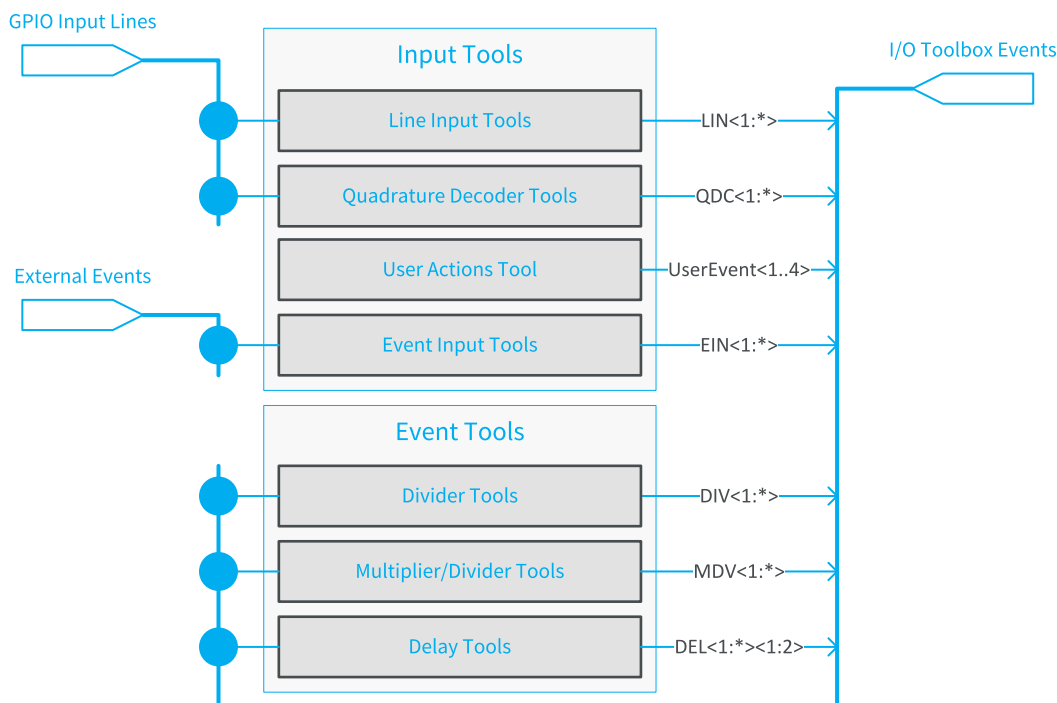
任何 I/O 工具箱事件，都可以在卡的任何 Gentl 设备模块中使用，如下所示：

- 硬件“循环”触发源。
- 硬件“扫描开始”触发源。
- 硬件“扫描结束”触发源。

每个 I/O 工具箱事件流，都与一个 32 位事件计数器相关联。

I/O工具箱结构

I/O工具箱是一个可配置的工具数组，它从应用于具有输入能力的I/O线路的外部信号建立事件脉冲流。



I/O工具箱结构图

I/O工具箱输入工具

I/O工具箱输入工具，由支持输入的GPIO行和从I/O工具箱外部选择的一组硬件事件源提供。

它们提供一个I/O工具箱事件流。

有四种输入工具：

1. **线路输入工具**，用于使用单个GPIO输入线路的传感器和探测器。
2. **正交解码器工具**与使用两个GPIO输入线的正交运动编码器一起使用。
3. 应用程序软件用于生成用户事件的用户操作工具
4. 用于“行扫描数据”转发应用程序的事件输入工具。

GPIO输入线路互连矩阵允许任何GPIO输入线路提供任何线路工具。

外部事件互连矩阵允许外部事件源提供任何事件输入工具。



备注 目前，只有连接器A的CoaXPress主机接口GPIO消息接收器可用。

I/O工具箱事件工具

I/O工具箱事件工具，由一个(或多个) I/O工具箱事件流提供，并传递一个(或多个) I/O工具箱事件流。

事件工具有三种类型：

- 1. 分割器工具**，通过将1个事件从输入流的D个事件中保留出来，来生成一个事件流。
- 2. 乘法器/分割器工具**生成一个事件流，输入流中的每D个事件都有M个事件。
- 3. 延迟工具**延迟一个(或两个)流的事件，可以是一个可配置的时间段，也可以是一个可配置的运动编码器滴答数。

内部事件互连矩阵，允许任何I/O工具箱事件流提供任何事件工具。

可以层叠工具，以形成工具链：

工具链总是以线工具开始。

(行或事件)工具，可以驱动0、1或多个事件工具。

9.2. I/O工具箱组合

I/O工具箱的组合，是特定于产品和固件变体。

1630 Coaxlink Mono

固件变体	#LIN	#QDC	#DIV	#MDV	#DEL	#UAS	#EIN
1-camera	8	1	1	1	2	1	0

1631 Coaxlink Duo

固件变体	#LIN	#QDC	#DIV	#MDV	#DEL	#UAS	#EIN
1-camera	8	1	1	1	2	1	0
1-camera, line-scan	8	1	1	1	2	1	0
2-camera	8	2	2	2	2	1	0
2-camera, line-scan	8	2	2	2	2	1	0

1632 Coaxlink Quad

固件变体	#LIN	#QDC	#DIV	#MDV	#DEL	#UAS	#EIN
1-camera	8	1	1	1	2	1	0
1-camera, line-scan	8	1	1	1	2	1	0
2-camera	8	2	2	2	2	1	0

1633 Coaxlink Quad G3

固件变体	#LIN	#QDC	#DIV	#MDV	#DEL	#UAS	#EIN
1-camera	8	1	1	1	2	1	0
1-camera, 4-data-stream	8	1	1	1	2	1	0
1-camera, line-scan	8	1	1	1	2	1	0
1-slm-camera	8	1	1	1	2	1	0
1-sls-camera	8	1	1	1	2	1	0
2-camera	8	2	2	2	2	1	0
2-camera, line-scan	8	2	2	2	2	1	0
3-camera	8	2	2	2	2	1	0
4-camera	8	4	4	4	4	1	0
4-camera, line-scan	8	4	4	4	4	1	0

1633-LH Coaxlink Quad G3 LH

固件变体	#LIN	#QDC	#DIV	#MDV	#DEL	#UAS	#EIN
1-camera	8	1	1	1	2	1	0
1-camera, 4-data-stream	8	1	1	1	2	1	0
1-camera, line-scan	8	1	1	1	2	1	0
1-slm-camera	8	1	1	1	2	1	0
1-sls-camera	8	1	1	1	2	1	0
2-camera	8	2	2	2	2	1	0
2-camera, line-scan	8	2	2	2	2	1	0
3-camera	8	2	2	2	2	1	0
4-camera	8	4	4	4	4	1	0
4-camera, line-scan	8	4	4	4	4	1	0

1629 Coaxlink Duo PCIe/104-EMB

固件变体	#LIN	#QDC	#DIV	#MDV	#DEL	#UAS	#EIN
1-camera	8	1	1	1	2	1	0
1-camera, line-scan	8	1	1	1	2	1	0
2-camera	8	2	2	2	2	1	0

1634 Coaxlink Duo PCIe/104-MIL

固件变体	#LIN	#QDC	#DIV	#MDV	#DEL	#UAS	#EIN
1-camera	8	1	1	1	2	1	0
1-camera, line-scan	8	1	1	1	2	1	0
2-camera	8	2	2	2	2	1	0

1635 Coaxlink Quad G3 DF

固件变体	#LIN	#QDC	#DIV	#MDV	#DEL	#UAS	#EIN
1-camera	8	1	1	1	2	1	0
1-df-camera	8	1	1	1	2	1	0
1-camera, line-scan	8	1	1	1	2	1	0
1-df-camera, line-scan	8	1	1	1	2	1	2

1637 Coaxlink Quad 3D-LLE

固件变体	#LIN	#QDC	#DIV	#MDV	#DEL	#UAS	#EIN
1-camera	8	1	1	1	2	1	0

3602 Coaxlink Octo

固件变体	#LIN	#QDC	#DIV	#MDV	#DEL	#UAS	#EIN
1-camera	8	1	1	1	2	1	0
1-camera, line-scan	8	1	1	1	2	1	0
2-camera	8	1	1	1	2	1	0
2-camera, line-scan	8	1	1	1	2	1	0
4-camera	8	1	1	1	2	1	0
4-camera, line-scan	8	4	4	4	4	1	0
5-camera	8	1	1	1	2	1	0
8-camera	8	1	1	1	2	1	0

3603 Coaxlink Quad CXP-12

固件变体	#LIN	#QDC	#DIV	#MDV	#DEL	#UAS	#EIN
1-camera	8	1	1	1	2	1	0
1-camera, line-scan	8	1	1	1	2	1	0
4-camera	8	4	4	4	4	1	0

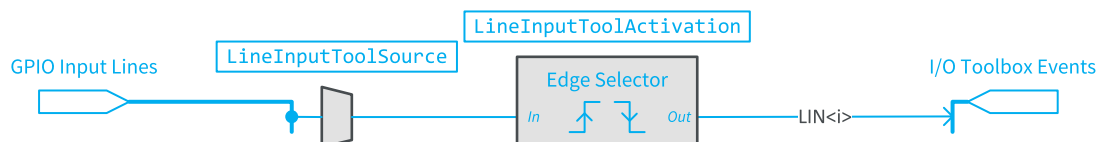
3620 Coaxlink Quad CXP-12 JPEG

固件变体	#LIN	#QDC	#DIV	#MDV	#DEL	#UAS	#EIN
4-camera	8	4	4	4	4	1	0

9.3. 行输入工具

工具名	短名称	输入计数/类型	输出计数/类型/名称
行输入工具	LIN	1 GPIO输入线	1事件流: LIN<i>

图解



LIN工具功能和接线图

可以选择任何支持输入的GPIO线路作为输入源。

该工具提供一个名为 **LIN<i>**的I/O工具箱事件流。

操作

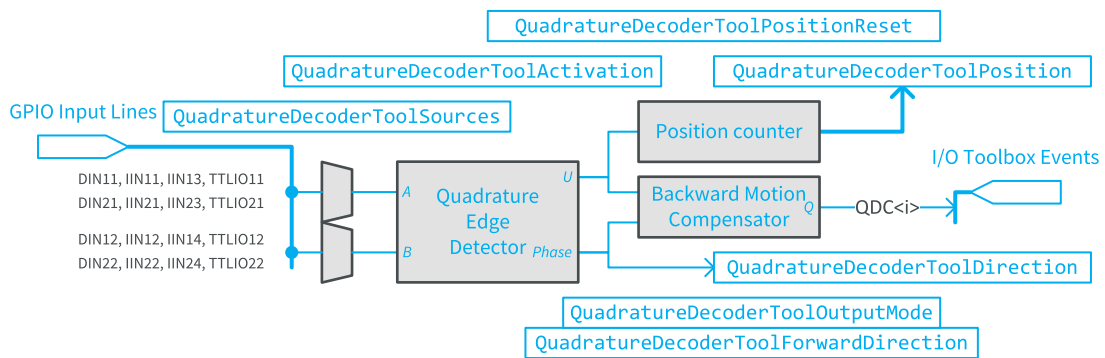
线路输入工具，检测 **LineInput** 信号的上升或下降沿，该信号由 **LineInputToolSource**选择的I/O控制块传送。

线路输入工具，根据设置 **LineInputToolActivation**，在每个上升或下降沿或同时在两个沿，发送一个事件。

9.4. 正交解码器工具

工具名	短名称	输入计数/类型	输出计数/类型/名称
正交解码器工具	QDC	2对I/O输入线	1事件流: QDC<i> 1状态位: QDC<i>Direction 1状态字: QDC<i>Position

图解



QDC工具功能和接线图

正交边缘检测器由**相位正交运动编码器**装置发送的一对名为A和B的信号提供。

源选择器允许选择以下对相邻的支持输入的GPIO输入线作为A/B输入源：DIN11-DIN12、DIN21-DIN22、TTLIO11-TTLIO12、TTLIO21-TTLIO22、IIN11-IIN12、IIN13-IIN14、IIN21-IIN22和IIN23-IIN24。

该工具包括以下功能块：

- 正交边缘检测器
- 后向运动补偿装置
- 32-位位置计数器

该工具提供：

- 一个名为**QDC<i>**的I/O工具箱事件流。
- 指示运动方向的方向状态位。
- 表示位置偏移的位置状态字。

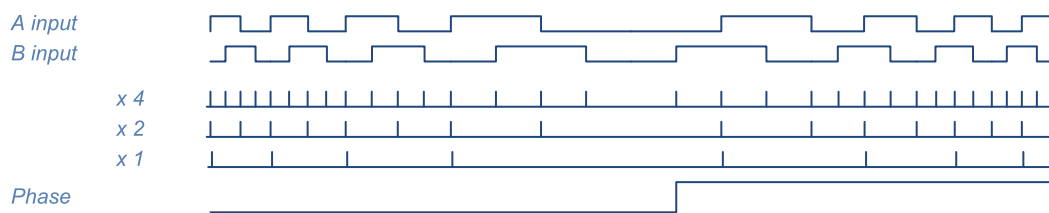
操作

正交解码器工具，对A/B信号进行解码，并在A/B循环中发送1、2或4个事件，可能通过反向运动补偿进行过滤。

正交边缘检测器

正交边缘检测器分析A/B线上的跃迁。它提供：

- 事件流，指定的U，每A/B循环有1、2或4个事件(根据 **QuadratureDecoderToolActivation** 设置)。
- A和B之间的相位标识(A导B或反之)



正交边缘检测器波形

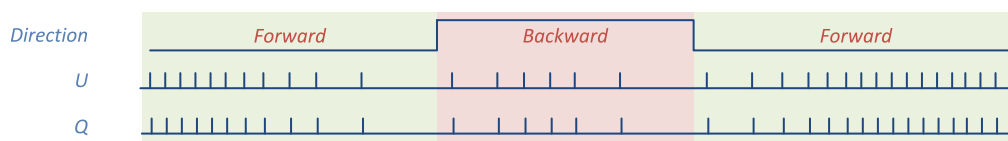
在传送到QDC*i*输出之前，U流可以被反向运动补偿器过滤。

相位指示，可根据设置 **QuadratureDecoderToolForwardDirection** 倒转，然后再传送到方向输出。

后向运动补偿装置

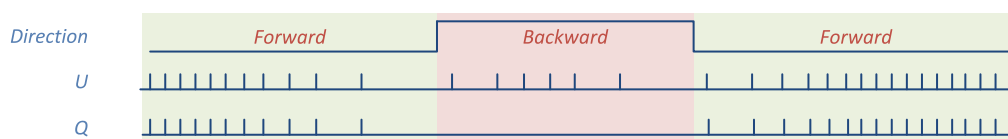
反向运动补偿器(BMC)根据设置 **QuadratureDecoderToolOutputMode** 过滤U流。

当设置为 **Unfiltered** 时，U流的所有事件都将传递到QDC*i*输出：



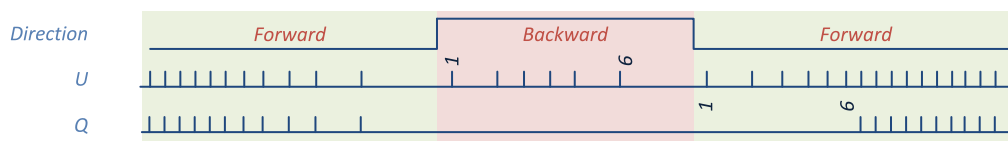
BMC波形-未过滤

设置为 **ForwardOnly** 时，只有与前进方向相对应的事件，才会传递到QDC*i*输出：



BMC波形-仅向前

设置为**FirstPassForwardOnly**时，只有与正向第一次通过相对应的事件，才会传递到QDC输出：



BMC波形-仅第一次向前通过

位置计数器

位置计数器对于与正向相对应的任何U事件递增1，而对于反向递减1。

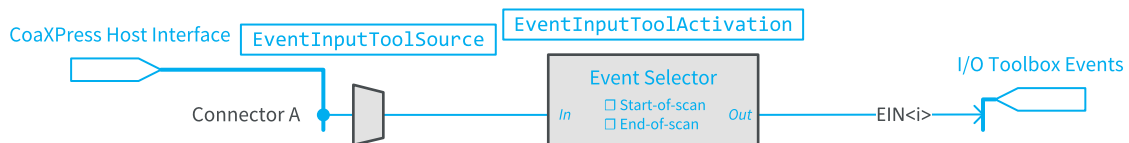
可以使用命令**QuadratureDecoderToolPositionReset**重置计数器。

9.5. 事件输入工具

适用于: **QuadG3DF**

工具名	短名称	输入计数/类型	输出计数/类型/名称
事件输入工具	EIN	1事件	1事件流: EIN<i>

图解



EIN工具功能和接线图

CoaXPress 主机接口的连接器A是唯一可以选择的事件源。

该工具提供一个名为 **EIN<i>**的I/O工具箱事件流。

操作

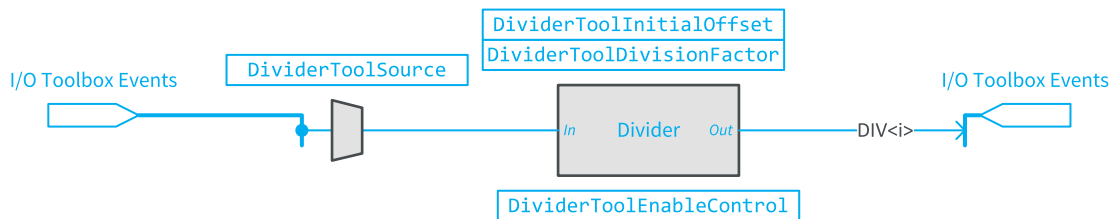
事件输入工具，对从属**1635 Coaxlink Quad G3 DF**的CoaXPress连接器上接收的自定义CoaXPressGPIO信息进行解码。

根据**EventInputToolActivation**设置。事件输入工具在接收“扫描开始信息”或“扫描结束信息”时发送一个事件。

9.6. 分割器工具

工具名	短名称	输入计数/类型	输出计数/类型/名称
分割器工具	DIV	1工具箱事件流	1事件流: DIV<i>

图解

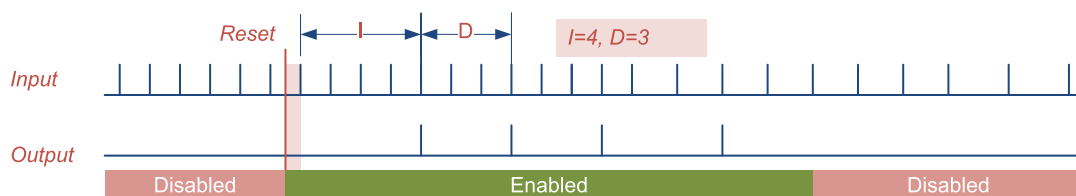


DIV工具功能和接线图

可以选择任何I/O工具箱事件流作为输入源。

该工具提供一个名为 **DIV<i>**的I/O工具箱事件流。

操作



DIV工具波形

一旦启用，分割器工具将跳过第一个-1-输入事件，然后在每个D输入事件传递事件。

分割因子-D-由 **DividerToolDivisionFactor**定义。默认值是 **2** 以及值的范围是 **1 ... 65535**。

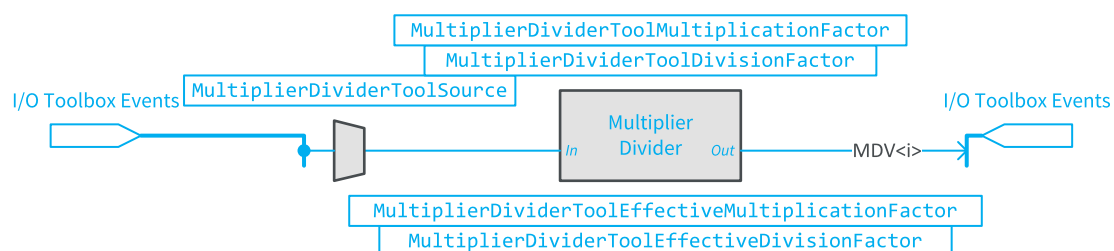
初始偏移量-1-由 **DividerToolInitialOffset**定义。默认值是 **0** 以及值的范围是 **0 ... 65535**。

操作状态由 **DividerToolEnableControl**定义。默认值是 **Disable**。

9.7. 乘法器/分割器工具

工具名	短名称	输入计数/类型	输出计数/类型/名称
乘法器/分割器工具	MDV	1工具箱事件流	1 工具箱事件流: MDV<i>

乘法器/分割器工具接线图



MDV工具功能和接线图

可以选择任何I/O工具箱事件流作为输入源。

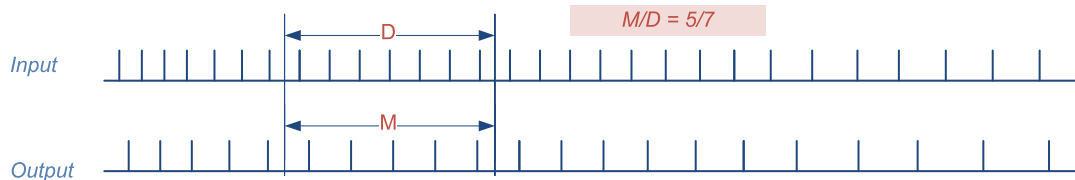
该工具提供一个名为 **MDV<i>** 的I/O工具箱事件流。

乘法器/分割器工具操作

乘法器/分割器工具，将输入速率乘以和/或除以0.001到1000.0范围内的任何速率转换比 (RCR) 值。

乘法器/分割器工具，测量每个连续输入事件之间的时间间隔，并相应地调整输出速率。

乘法器/分割器**频率准确**。如果输入频率稳定(或变化缓慢)，则输出频率与输入频率严格成正比。在这种情况下，乘法器/分割器为每个D输入事件提供M事件。



MDV工具波形

速率转换比率，配置为两个浮点数的比率：

- **M**值由 **MultiplierDividerToolMultiplicationFactor** 定义。默认值是 **MultiplierDividerToolMultiplicationFactor** 以及值的范围是 **MultiplierDividerToolMultiplicationFactor** 到 **MultiplierDividerToolMultiplicationFactor**。
- **M**值由 **MultiplierDividerToolDivisionFactor** 定义。默认值是 **MultiplierDividerToolDivisionFactor** 以及值的范围是 **MultiplierDividerToolDivisionFactor** 到 **MultiplierDividerToolDivisionFactor**。

有效乘除因子分别由 **MultiplierDividerToolMultiplicationFactor** 和 **MultiplierDividerToolDivisionFactor** 报告。



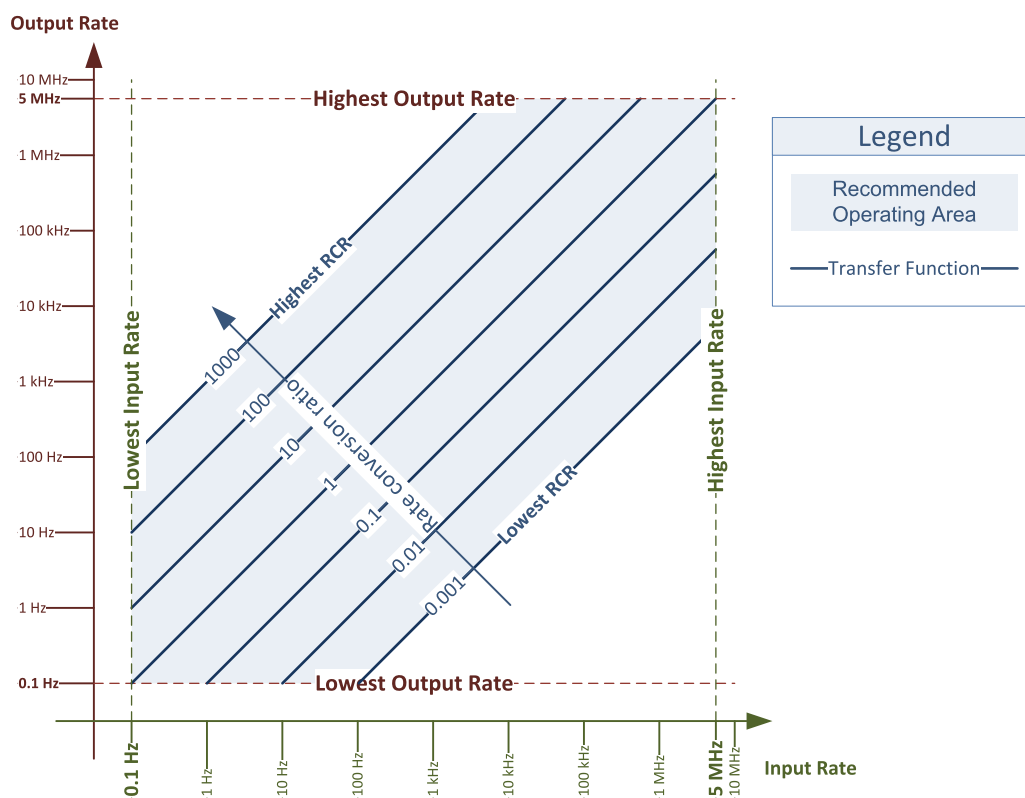
备注 有效值可能与规定值略有不同。但是，RCR相对误差仍然可以忽略不计(小于1/1000)。



备注 将输入事件流的频率变化报告给输出事件流，输入事件流的延迟为1个周期。这种延迟会在输出事件流中引起一些相位错误。当输入频率增加时，输出事件流的相位累积延迟，当输入频率增加时，反之亦然。因此，乘法器/分割器**相位不准确**。

乘法器/分割器工具操作极限

特性	符号	最小	最大
比率换算率	RCR	0.001	1000
输入率	f_{IN}	0.1 Hz	5 MHz
输出率	f_{OUT}	0.1 Hz	5 MHz

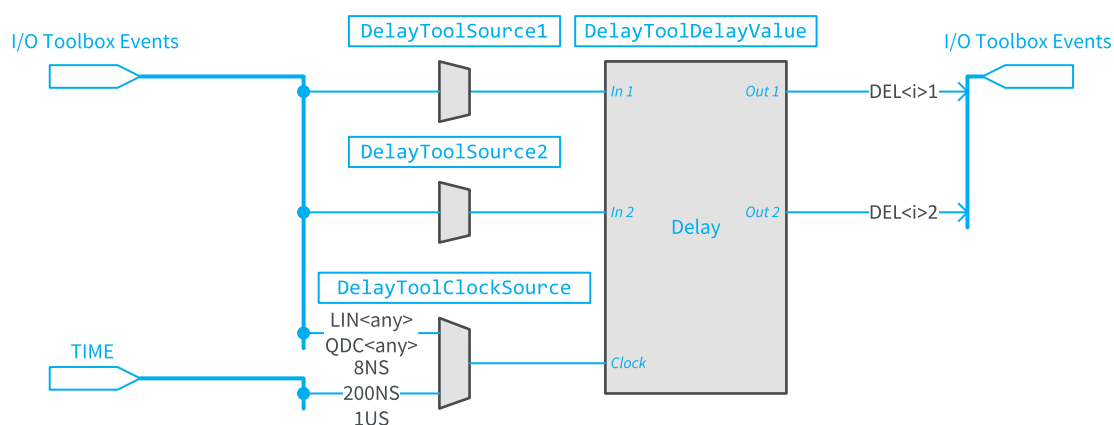


MDV工具操作极限图

9.8. 延迟工具

工具名	短名称	输入计数/类型	输出计数/类型/名称
延迟工具	DEL	2工具箱事件流 1时钟信号	2 工具箱事件流: DEL<i>1 , DEL<i>2

图解



Del工具功能和接线图

可以选择任何I/O工具箱事件流作为输入1源。

可以选择任何I/O工具箱事件流作为输入2源。

该工具提供两个I/O工具箱事件流。工具实例的输出<i>名为 **DEL<i>1** 和 **DEL<i>1**。

操作

应用于任一输入 (**In1** 和 **In2**) 的事件流, 在可配置的时钟滴答事件数之后复制到相应的输出上 (**Out1** 和 **Out2**)。

分别由 **DelayToolSource1** 和 **DelayToolSource2** 选择源。

相同的延迟适用于两个通道。公共延迟由 **DelayToolDelayValue** 定义。

相同的时钟源, 适用于两个信道。时钟源由 **DelayToolClockSource** 定义。它可以是 **时基** 或 **线性工具事件流**。

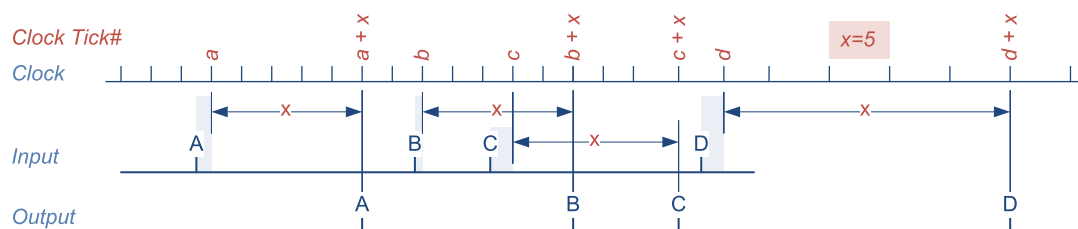
选择一个 **时基** 实现一个延时功能。可用的时间基是:

- **8NS**: 125兆赫高精度常规时基, 通过8纳秒的步进, 允许从 **40纳秒** 到 **134毫秒** 的延迟。
- **200NS**: 5兆赫高精度常规时基, 通过200纳秒的步进, 允许从 **200纳秒** 到 **3.35秒** 的延迟。
- **1US**: 1兆赫高精度常规时基, 通过1微秒的步进, 允许从 **1微秒** 到 **16.7秒** 的延迟。

当测线工具由运动编码器设备供电时, 选择 **线工具事件流** 实现位置偏移功能。任何可用的线路输入工具或正交解码器工具都可以用作延迟时钟源。延迟范围是 **1** 到 **16777215** 个事件。



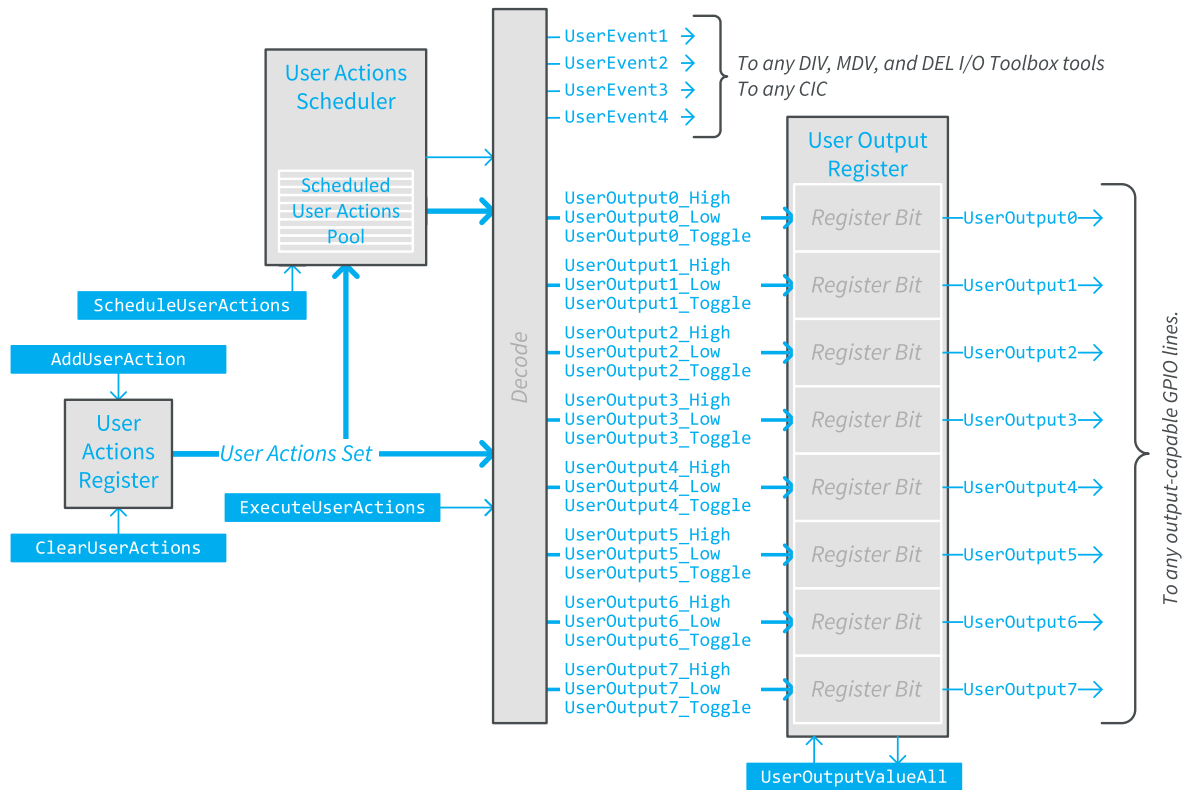
警告 延迟工具作为延迟线工作。当上一个事件尚未交付时, 该工具可以接受新事件! 延迟工具能够全局记录所有通道(多达16个不同的事件)。



Del工具波形

9.9. 用户操作工具和用户输出寄存器

简介



UAS工具功能和接线框图

用户操作工具(UAS)允许应用程序软件执行以下用户操作:

- 在任何用户事件源上生成事件。
- 设置高的用户输出寄存器的任何位。
- 设置低的用户输出寄存器的任何位
- 切换用户输出寄存器的任何位。

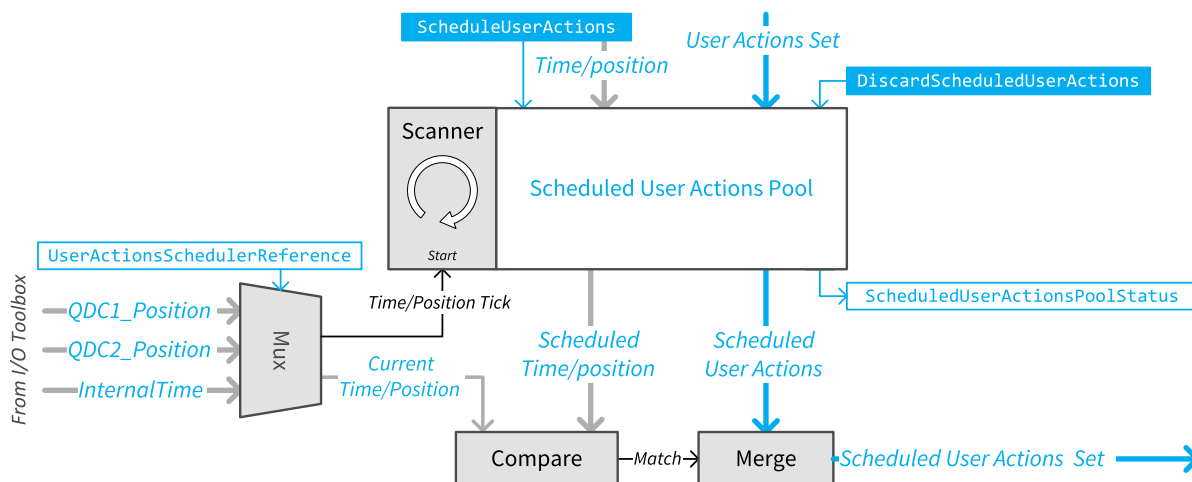
使用 **ClearUserActions**和**AddUserActions**GenICam 功能, 用户应用程序可以定义用户操作集, 该用户动作集包含所有要同时执行的动作。

在定义了用户操作集之后, 应用程序软件有两个执行选项:

- 使用 **ExecuteUserActions**GenICam功能, 立即执行用户操作集中定义的所有操作。

- 使用 **ScheduleUserActions** GenICam 功能，将用户操作集的执行委托给用户操作计划程序。

用户操作计划程序



用户操作计划程序功能图

UAS 功能块，允许应用软件推迟操作的执行。

计划程序引用设置

在使用用户操作计划程序之前，用户应用程序必须选择 32-位引用计数器。这是通过将以下三个值之一指定给 **UserActionsSchedulerReference** GenICam 功能实现的：

- **InternalTime** 选择 **Coaxlink 卡本地时间**：一个单调时基，每 1 微秒递增 1，并在达到最大值 4,294,967,295，大约 71 分钟后环绕。
- **QDC1Position** 和 **QDC2Position** 分别选择正交译码器 QDC1 和 QDC2 的 **位置计数器**。



警告 为了使用位置基准正确操作 UAS，位置计数器必须单调递增，且不超过每微秒一次。



备注 为确保 QDC 位置计数器的单调增量：

- 正确设置 **QuadratureDecoderToolForwardDirection**，使计数器在对象向前移动时递增。
- 如果存在任何向后运动，通过将 **QuadratureDecoderToolOutputMode** 设置为 **ForwardOnly** 或 **FirstPassForwardOnly** 来防止位置计数器递减。

计划的用户操作池操作

计划的用户操作池，是一个内存区域，其中计划的用户操作集由用户应用程序存储。该池有64个位置。

要向池中添加新的计划用户操作，用户应用程序必须：

1. 通过获取 **ScheduledUserActionsPoolStatusGenICam** 功能值，确保至少有一个空闲位置，
2. 定义用户操作集，
3. 确定执行操作时的时间/位置32位值，
4. 将此值设置为 **ScheduledUserActionsGenICam** 功能。

当执行计划的用户操作时，将从池中删除这些操作。

通过执行 **DiscardScheduledUserActionsGenICam**命令，可随时清除池。

扫描器操作和计划的操作执行

在32位(时间或位置)参考计数器的每次增量处，扫描器读取所有位置，并将计划的参考时间/位置与当前时间/位置计数值进行比较。

当这些值相同时，将在扫描结束时选择要执行的计划用户操作集，并将其从池中删除。

当选择多个集合执行时，将会合并它们的操作。



备注 将同一用户输出寄存器位上的设置低和设置高操作合并为切换操作。

扫描结束时：合并的选定操作同时执行。从参考勾选到执行所选操作的时间延迟非常小(亚微秒)且恒定。

用户事件源

有名为 UserEvent1 ... UserEvent4的用户事件源。

可以使用任何用户事件：

- 作为所有相机和照明控制器的循环触发源。
- 作为所有相机和照明控制器的循环序列触发源。
- 作为所有I/O工具箱的分隔器工具的事件源。
- 作为I/O工具箱的所有乘法器/分割器工具的事件源
- 作为I/O工具箱的所有延迟工具的两个通道的事件源。

用户输出寄存器

CoaxLink产品提供一个用户输出寄存器，其中位被命名为UserOutput0 ... UserOutput7。



备注 **1630 Coaxlink Mono** 仅实现4个最低位！

任何用户输出寄存器位，都可以被任何有输出能力的GPIO线路用作信号源。

用户应用程序有两个选项来定义用户输出寄存器位的状态：

- 用户操作选项允许单独更改每个位的状态。
- 将值设置为 **UserOutputValueAll**，以定义所有位的状态。

获取 **UserOutputValueAll** 的值，允许用户应用程序获取所有位的状态。

10. 事件信令和计数

10.1. 简介	145
10.2. 自定义事件源	148
10.3. 特定于事件的上下文数据	151
10.4. 关于GenTL 信令	153

10.1. 简介

简述

CoaxLink产品具有强大的事件管理功能，允许应用程序收到各种事件发生的通知。

除了Gentl标准EVENT_NEW_BUFFER事件之外，CoaxLink Gentl发生器还提供了一系列自定义事件源。

事件源根据其所属的功能块和GenTL模块，按类型分组。

每个自定义事件源都与(一个计数发生次数的)计数器相关联。

对于每个通知的自定义事件，将记录以下事件上下文数据，并使其可供应用程序使用：

- 事件源的标识符
- 时间戳(以微秒表示)
- 3用户定义的上下文数据

每个单独的事件源都是可配置的：

- 事件通知可以启用或禁用。
- 每个用户定义的上下文数据的内容。

事件数据临时存储在“事件队列缓冲区”中。使用中断机制，将通知Coaxlink驱动程序在Coaxlink事件队列缓冲区中一个或多个事件条目的可用性。

Coaxlink驱动程序实现了向应用程序软件报告异步事件发生的GenTL信令机制。

Egrabber API提供了3个回调线程模型：

- **CallbackOnDemand:** 这是最简单的模型，它可以完全控制调用回调的时间和方式。事件按需处理。
- **CallbackSingleThread:** 此模型在专用线程上下文中，按时间顺序，将事件传递给回调。一旦事件可用，就会自动处理。
- **CallbackMultiThread:** 此模型在单独的线程(每个事件DATA类型一个线程)中，向回调传递事件。一旦事件可用，就会自动处理。

事件类型

GenTL 根据事件的类型和相关的对象模块，来识别事件。

标准事件类型

Coaxlink 驱动程序执行以下标准事件类型，以便由 GenTL Consumer 应用程序注册：

GenTL 标准事件类型	GenTL 模块	描述
EVENT_NEW_BUFFER	数据流	有关新填充缓冲区的通知。

自定义事件类型

除了 **标准事件类型**，GenTL 规范还为 **自定义事件类型** 提供了空间。

自定义事件类型特定于 GenTL 发生器实现(在本例中是 CoaxLink 驱动程序)。

Coaxlink 驱动程序实现以下自定义事件类型，以便由 GenTL Consumer 应用程序注册：

事件类型	模块	描述
EVENT_CUSTOM_IO_TOOLBOX	接口(设备)	I/O 工具箱事件通知
EVENT_CUSTOM_CXP_INTERFACE	接口(设备)	CoaXPress 主机接口事件通知
EVENT_CUSTOM_CIC	设备	相机和照明控制事件通知
EVENT_CUSTOM_DATASTREAM	数据流	CoaXPress 数据流事件通知



备注 EVENT_CUSTOM_IO_TOOLBOX 和 EVENT_CUSTOM_CXP_INTERFACE 事件类型也可以在设备模块上注册。



备注 自定义事件类型是通用的；每个类型收集多个事件源。

另见：“自定义事件源”于页面 148 (一份详尽的清单)。

自定义事件计数器

32 位计数器与每个自定义事件源相关联。

无法禁用计数器。当它达到它的最大值， $4\,294\,967\,295 (2^{32} - 1)$ ，它会绕到 0。

在任何时候，用户应用程序都可以：

- 读取所选事件源的计数值。
- 重置所选事件源的计数器。
- 重置模块所有事件源的计数器。

Count 值也可以被任何事件源用作用户定义的上下文数据。

自定义事件配置

事件源是可配置的。

在任何时候，用户应用程序都可以：

- 启用或禁用所选事件源的通知。
- 启用或禁用模块所有事件源的通知。
- 定义所选事件源的每个用户定义上下文数据的内容。

通知

在默认情况下，禁用所有通知。

应用软件必须根据应用程序的需要，配置事件通知过滤器。

通知过滤器配置的配置可以随时修改，而不影响事件计数功能。

上下文数据

事件上下文数据的最后3个32位上下文数据字，可以配置如下：

- 特定于事件的数据。
- 事件发生时，采样的I/O行的状态
- 任何事件计数器的计数值。
- 任何正交解码器(QDC)位置计数器的计数值。

一些事件源提供了其他选项。

另见： "特定于事件的上下文数据" 于页面 151

10.2. 自定义事件源

数据流事件源(数据流模块)

EVENT_CUSTOM_DATASTREAM

事件源	描述
DATASTREAM_START_OF_CAMERA_READOUT	图像帧的第一个像素数据，写入板载FIFO缓冲区。 仅适用于区域扫描固件变体。
DATASTREAM_END_OF_CAMERA_READOUT	图像帧的最后一个像素数据，写入板载FIFO缓冲区。 仅适用于区域扫描固件变体。
DATASTREAM_START_OF_SCAN	图像扫描的第一个像素数据，写入板载FIFO缓冲区。 仅适用于行扫描固件变体。
DATASTREAM_END_OF_SCAN	图像扫描的最后一个像素数据，写入板载FIFO缓冲区。 仅适用于行扫描固件变体。
DATASTREAM_REJECTED_FRAME	图像帧被拒绝(板载FIFO缓冲区已满) 仅适用于区域扫描固件变体。
DATASTREAM_REJECTED_SCAN	图像扫描被拒绝(板载FIFO缓冲区已满) 仅适用于行扫描固件变体。

I/O工具箱自定义事件(接口模块)

EVENT_CUSTOM_IO_TOOLBOX event sources

事件源	描述
IO_TOOLBOX_LIN1	行输入工具1-事件输出
IO_TOOLBOX_LIN2	行输入工具2-事件输出
IO_TOOLBOX_LIN3	行输入工具3-事件输出
IO_TOOLBOX_LIN4	行输入工具4-事件输出
IO_TOOLBOX_LIN5	行输入工具5-事件输出
IO_TOOLBOX_LIN6	行输入工具6-事件输出
IO_TOOLBOX_LIN7	行输入工具7-事件输出
IO_TOOLBOX_LIN8	行输入工具8-事件输出
IO_TOOLBOX_QDC1	正交解码器工具1-事件输出
IO_TOOLBOX_QDC1_DIR	正交解码器工具1-改变方向
IO_TOOLBOX_QDC2	正交解码器工具2-事件输出
IO_TOOLBOX_QDC2_DIR	正交解码器工具2- 改变方向
IO_TOOLBOX_DIV1	除法工具1-事件输出
IO_TOOLBOX_DIV2	除法工具2-事件输出
IO_TOOLBOX_MDV1	乘法/除法工具1-事件输出
IO_TOOLBOX_MDV2	乘法/除法工具2-事件输出
IO_TOOLBOX_DEL11	延迟工具1输出1-事件输出
IO_TOOLBOX_DEL12	延迟工具1输出2-事件输出
IO_TOOLBOX_DEL21	延迟工具2输出1-事件输出
IO_TOOLBOX_DEL22	延迟工具2输出2-事件输出
IO_TOOLBOX_USER_EVENT_1	用户事件1
IO_TOOLBOX_USER_EVENT_2	用户事件2
IO_TOOLBOX_USER_EVENT_3	用户事件3
IO_TOOLBOX_USER_EVENT_4	用户事件4
IO_TOOLBOX_EIN1	事件输入工具1-事件输出
IO_TOOLBOX_EIN2	事件输入工具2-事件输出



备注 检查"I/O工具箱组合"于页面126，是否有适用的值。

CoaXPress主机接口自定义事件(接口模块)

EVENT_CUSTOM_CXP_INTERFACE

事件源	描述
CXP_INTERFACE_CRC_ERROR_CXP_A	在CoaXPress主机接口的连接A上检测到CRC错误。
CXP_INTERFACE_CRC_ERROR_CXP_B	在CoaXPress主机接口的连接B上检测到CRC错误。
CXP_INTERFACE_CRC_ERROR_CXP_C	在CoaXPress主机接口的连接C上检测到CRC错误。
CXP_INTERFACE_CRC_ERROR_CXP_D	在CoaXPress主机接口的连接A上检测到CRC错误。

相机和照明控制器事件源(设备)

EVENT_CUSTOM_CIC 事件源

事件源	描述
CIC_CAMERA_TRIGGER_RISING_EDGE	相机触发输出信号的上升沿 (在RC和RG相机控制方法中的曝光开始)
CIC_CAMERA_TRIGGER_FALLING_EDGE	相机触发输出信号的下降沿 (在RG相机控制方法中的曝光结束)
CIC_STROBE_RISING_EDGE	频闪输出信号的上升沿
CIC_STROBE_FALLING_EDGE	频闪输出信号的上升沿
CIC_ALLOW_NEXT_CYCLE	允许立即开始新的相机循环。
CIC_DISCARDED_CIC_TRIGGER	丢弃CIC循环触发。
CIC_PENDING_CIC_TRIGGER	记录一个CIC循环触发，但它的执行会延迟到CIC就绪。
CIC_CXP_TRIGGER_ACK	接收到响应CoaXPress主机到设备触发消息的肯定确认。
CIC_CXP_TRIGGER_RESEND	执行CoaXPress主机到设备触发消息的重新发送。
CIC_TRIGGER	CIC触发



备注 每个Gentl设备模块，有一个相机和照明控制器实例。每个Coaxlink卡的GenTL设备模块数，由固件变量定义。

10.3. 特定于事件的上下文数据

EVENT_DATA_NUMID_CIC_DISCARDED_CIC_TRIGGER

EVENT_DATA_NUMID_CIC_DISCARDED_CIC_TRIGGER 的 EventSpecific 值，是一个可以根据以下定义解释的位字段：

Bit#	描述
0	原因：图像缓冲区已满。
1	原因：相机循环未完成。
2	原因：已达到特定触发的最大数目。
3	原因：数据流未激活

EVENT_DATA_NUMID_CIC_PENDING_CIC_TRIGGER

EVENT_DATA_NUMID_CIC_PENDING_CIC_TRIGGER 的 EventSpecific 值，是一个可以根据以下定义解释的位字段：

Bit#	描述
0	原因：图像缓冲区已满。
1	原因：相机循环未完成

EVENT_DATA_NUMID_DATASTREAM_START_OF_SCAN

EVENT_DATA_NUMID_DATASTREAM_START_OF_SCAN 的 EventSpecific 值，是可以根据以下定义解释的位字段：

Bit#	描述
1	原因：软件触发
2	原因：硬件触发
3	原因：DSStartAcquisition 或上次扫描结束

EVENT_DATA_NUMID_DATASTREAM_END_OF_SCAN

EVENT_DATA_NUMID_DATASTREAM_END_OF_SCAN的EventSpecific值，是可以根据以下定义解释的位字段：

Bit#	描述
1	原因：软件触发
2	原因：硬件触发
3	原因：原因：达到扫描长度
4	原因：DSStopAcquisition
5	原因：内部异常(图像缓冲区几乎已满)

EVENT_DATA_NUMID_DATASTREAM_REJECTED_FRAME

EVENT_DATA_NUMID_DATASTREAM_REJECTED_FRAME 的EventSpecific值，是一个可以根据以下定义进行解释的位字段：

Bit#	描述
0	原因：图像缓冲区已满。
1	原因：数据流未激活

10.4. 关于 GenTL 信令

Coaxlink 驱动程序实现了 gentl 提供者的信号机制。

下面简要地介绍这种机制。

另见：有关详细说明，请参阅 [GenICam GenTL 标准版本 1.4](#) 第34页开始的第4.2节。

事件注册

来源：*GenTL 规范*

在向 GenTL 用户通知事件之前，必须注册事件对象。在枚举过程中创建模块实例后，可以使用 `GCRegisterEvent()` 函数创建事件对象。此函数返回一个唯一的 `EVENT_HANDLE`，用于标识已注册的事件对象。要获取有关已注册事件的信息，可以使用 `EventGetInfo()` 函数。

每个模块和事件类型只能注册一个事件。

(...)

获取 `EVENT_HANDLE` 后，GenTL 用户可以等待通过调用 `EventGetData()` 函数来通知的事件对象。事件传送后，事件对象将携带数据。当对 `EventGetData()` 的调用成功时，此数据将被复制到 GenTL 用户提供的缓冲区中。

通知和数据检索

来源：*GenTL 规范*

如果向事件对象发出信号，则会在某个时间点，将数据放入事件数据队列。可以调用 `EventGetData()` 函数来检索实际数据。

(...)

使用此函数读取数据时，数据将从队列中删除。然后，GenTL 发生器实现，检查事件数据队列是否为空。如果有更多可用数据，则事件对象将保持发出信号，接下来对 `EventGetData()` 的调用，将传递下一个队列条目。否则，事件对象将重置为未发出信号的状态。

(...)

确切的数据类型，取决于事件类型和 gentl 发生器实现。数据被复制到由 GenTL 用户分配的用户缓冲区中。可以使用 `EventGetDataInfo()` 函数，查询事件数据的内容。要填充的缓冲区的最大大小由事件类型定义，并且可以在缓冲传递后，使用 `EVENT_INFO_DATA_SIZE_MAX` 进行查询。可以使用 `EventGetInfo()` 函数查询此信息。

11. 高级功能

11.1. 子链路采集	155
11.2. 多流采集	156
11.3. CoaXPress数据转发	157
数据转发原则	158
数据转发连接方案	160
行扫描触发同步	161
配置脚本示例	162
11.4. 平场校正	163
什么是平场校正?	164
Coaxlink FFC	167
FFC向导示例程序	170
11.5. 查找表处理	173
LUT处理简介	174
LUT处理可用性	175
单色查找表格处理	176
LUT内容定义	177
LUT设置程序	184
11.6. Bayer CFA解码	187
Bayer CFA到RGB转换	188
使用Bayer CFA解码器	192
11.7. JPEG 编码	194
11.8. 激光线提取	196
简介	197
LLE处理核心实现	199
用例示例	208
附录	215
11.9. 行扫描采集	216
线扫描采集原理	217
线扫描采集用例	221
11.10. C2C -链路	224
C2C链接描述	225
C2C链路规范	226
触发传播延迟	228
CIC同步	230
C2C链路设置步骤	233
11.11. OEM 安全钥	235
介绍OEM安全钥	235
使用OEMSafetyKey	236

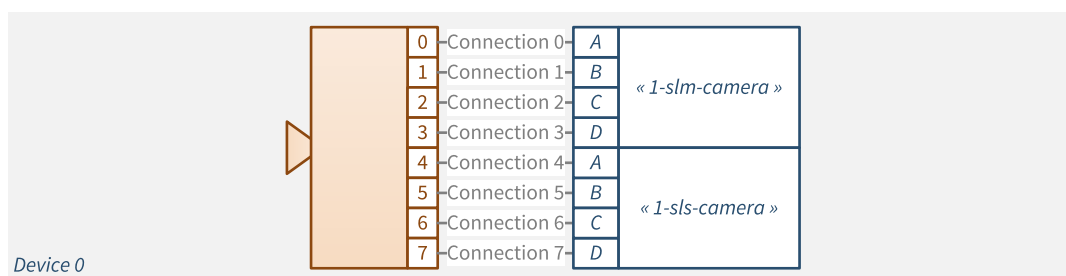
11.1. 子链路采集

适用于: QuadG3 QuadG3LH

原则

该子链路采集功能允许从特定的8-连接 CoaXPress相机采集图像。

8-连接CoaXPress链接分为2个子链接。每个子链接使用1D8SL4连接方案连接到Coaxlink:



使用2个子链接和2个Coaxlink卡的8连接相机

第一个4-连接子链路连接到“子链路(主)捕获器”: 一个配备了 1-slm-camera 固体变件的 Coaxlink卡。

下一个4-连接子链路连接到“子链路(从)捕获器”: 一个配备了 1-sls-camera 固体变件的 Coaxlink卡。

每个捕获器将二分之一的图像帧传送到一个GenTL缓冲区。应用程序必须通过合并两个相应缓冲区的内容来重建整个图像帧。这在320-sublinkEgrabber示例程序中得以显示。

主捕获器控制摄像机,并管理系统触发器。

两个捕获器都被配置为捕获各自子链接所有图像数据。

相机要求

此功能仅适用于具有以下特征的8-连接区域扫描相机:

- 第一行的图像头和图像数据被打包到一个单独的CoaXPress数据包中,并传送到CoaXPress连接0。
- 对于帧(或ROI)的其余行,单个图像行的图像数据打包到单个数据包中,并传递到下一个CoaXPress连接。使用CoaXPress标准包分发顺序规则旋转连接:(0到7,然后返回0)。
- 图像帧(或ROI)高度必须是8行的倍数,以确保最后一个图像行在最后一个连接(连接7)上交付。

11.2. 多流采集

适用于: QuadG3 QuadG3LH

4-数据-流并行采集

1-camera, 4-data-stream 固件变体(**1633 Coaxlink Quad G3** 和 **1633-LH Coaxlink Quad G3 LH**)允许连接一个区域扫描CoaXPress相机, 该相机提供多达4个独立的数据流。

帧捕获器根据CoaXPress `StreamID` 2个最低有效位的值, 对传入的CoaXPress数据块进行排序, 并提供四个独立的数据路径。

每个数据路径, 都能够处理全部CoaXPress链路带宽, 即2.5千兆字节/秒。

它包括:

1. 250 MB FIFO缓冲存储器
2. 一种允许将10/12/14-位数据与16位容器的LSB或MSB对齐的像素处理器。
3. 一种DMA引擎, 可直接将图像数据传送到主机PC的用户存储器空间。

该 `multistream` 示例程序, 演示了如何创建4个EGrabber实例, 并开始对4个并发数据流进行采集。

11.3. CoaXPress数据转发

适用于: **QuadG3DF**

数据转发原则	158
数据转发连接方案	160
行扫描触发同步	161
配置脚本示例	162

数据转发原则

1635 Coaxlink Quad G3 DF 的数据转发功能，允许将图像数据从相机转发到不同主机PC的多个帧捕获器。

数据转发

具有**DF功能**的卡，比如**1635 Coaxlink Quad G3 DF**，将CoaXPress主机连接器上接收到的数据，转发到CoaXPress数据转发连接器。

在Coaxpress主机连接器的连接A、B、C、D上嵌入串行位流中的图像数据包，转发到Coaxpress数据转发连接器的连接FA、FB、FC、FD。

串行位流被重新计时，以始终以CXP-6速度运行，而不管相机的链路速度如何。因此，空闲字符(在必要时)被删除或添加到位流中。添加或删除空闲字符不会影响有效负载。保存图像数据(包括CRC)。为了正确操作数据转发，必须按照Coaxlink 1.1标准§8.2.5.1的要求，相机每100个字至少插入一个空闲字。

图像数据以可忽略的延迟重新传输：通常是32位字符传输时间的几个周期。



备注 数据转发输出端口，不符合Coaxpress设备的规格！它只能供给另一个**1635 Coaxlink Quad G3 DF**。

数据转发链

一个**DF链**是由两个或多个具有数据转发功能的卡组成，其中一个卡的CoaXPress数据转发连接器使用一组1、2或4的同轴电缆(名为**DF-桥**)连接到下一个卡的CoaXPress主机连接器。



备注 DF链中的卡数没有指定的上限。

相机连接到DF链第一张卡的CoaXPress主机连接器上，该卡名为**DF-主卡**。DF链的其他卡被命名为**DF-从卡**。最后一个DF从卡的CoaXPress数据转发连接器未连接。

CoaXPress链接发现和配置

DF主卡负责发现和配置相机的CoaXPress链路。

DF从卡的CoaXPress主机接口，自动配置为与DF主卡发现的相同数量的连接。

例如，如果相机使用两个连接，则每个DF桥只需要两个连接。

固件变体

必须根据相机类型，选择要安装在DF主卡上的固件变体：

- 对于区域扫描相机，安装1-camera 固件变体。
- 对于行扫描摄像头，安装1-camera, line-scan固件变体。

安装在DF从卡上的固件变体，必须与安装在DF主卡上的固件变体相匹配：

- 当1-camera firmware variant is installed on the DF-master, install the 1-df-camera固件变体。
- 当1-camera, line-scan firmware variant is installed on the DF-master, install the 1-df-camera, line-scan 固件变体。

数据采集控制

为了正确操作数据转发，应用程序必须遵守以下规则：

- 数据采集必须在所有的DF从卡上激活，然后才能在DF主卡上激活。
- 数据采集必须先在DF主卡上取消激活，然后再在DF从卡上取消激活。

仅适用于行扫描应用，执行扫描启动和扫描结束同步机制，以确保DF链的所有卡都能捕获相同行的图像数据。

另见：“行扫描触发同步”于页面161

相机循环控制

若应用程序要求，DF主卡负责CoaXPress“主机到设备”触发的确立。这与非数据转发Coaxlink卡的实现方式相同。

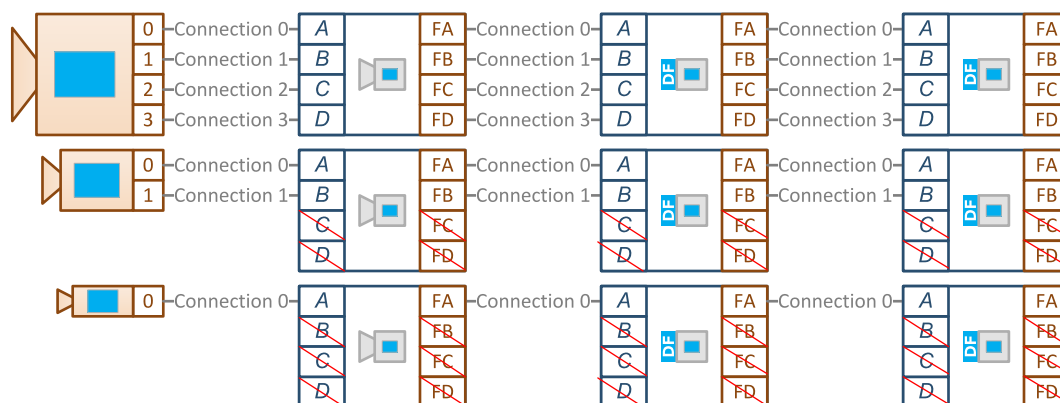
另见：“CoaXPress主机到设备触发”于页面38

数据转发连接方案

区域扫描相机数据转发

下图说明了将区域扫描相机的图像数据转发到3台主机PC的3种连接方案：一种用于4连接相机，一种用于2连接相机，一种用于单连接相机。

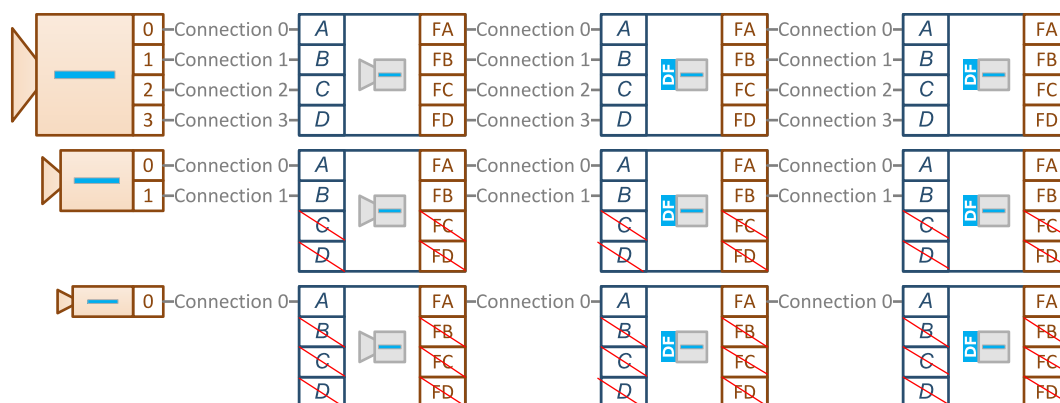
第一个 **1635 Coaxlink Quad G3 DF** 卡必须配置 1-camera firmware variant; the other must be configured with the 1-df-camera 固件变体。



行扫描相机数据转发

下图说明了将行扫描的图像数据转发给3个主机PC的3种连接方案：一种用于4连接相机，一种用于2连接相机，一种用于单连接相机。

第一个 **1635 Coaxlink Quad G3 DF** 卡必须配置 1-camera, line-scan firmware variant; the other must be configured with the 1-df-camera, line-scan 固件变体。



行扫描触发同步

扫描启动和扫描结束触发同步机制，确保DF链的所有卡，捕获相同的图像数据行。

数据转发-主卡

在DF链中，DF-主卡将其扫描启动和扫描结束事件转发给DF从卡。

在DF主卡上，扫描启动和扫描结束事件的生成方式，与非数据转发Coaxlink卡相同。

另见： "行扫描采集" 于页面216

DF-主卡：

- 首先，在下一行首图像数据上，同步异步扫描触发。
- 然后，与所有DF从卡，共享同步扫描触发。

扫描触发的共享，是通过在位流中插入高优先级“自定义GPIO消息”来实现的。这些消息与图像数据一起，由所有的DF从卡转发。

数据转发-从卡

在接收到这样的消息时，DF从卡生成一个硬件事件。可能的两种事件：

- 扫描启动事件。
- 扫描结束事件。

这些事件通过I/O工具箱的"事件输入工具" 于页面133 提供。

对于需要同步行扫描采集的应用程序，必须使用DF从卡的I/O工具箱EIN工具，作为本地扫描启动和扫描结束的触发源。

配置脚本示例

以下脚本，为同步行扫描采集，配置数据转发帧抓取器：

```
for (var grabber of grabbers) {
  if (grabber.InterfacePort.get("InterfaceID").includes('df-camera')) {
    console.log("Configuring slave card");
    // set the Width/Height/PixelFormat of the (virtual) remote device (on
    // the slave card) equal to the Width/Height/PixelFormat of the (real)
    // camera (connected to the master card)
    grabber.RemotePort.set("Width", 8192);
    grabber.RemotePort.set("Height", 1);
    grabber.RemotePort.set("PixelFormat", "Mono8");
    // configure the event input tool EIN1
    grabber.InterfacePort.set("EventInputToolSource[EIN1]", "A");
    grabber.InterfacePort.set("EventInputToolActivation[EIN1]", "StartOfScan");
    // configure the event input tool EIN2
    grabber.InterfacePort.set("EventInputToolSource[EIN2]", "A");
    grabber.InterfacePort.set("EventInputToolActivation[EIN2]", "EndOfScan");
    // configure start/end of scan triggers
    grabber.StreamPort.set("StartOfScanTriggerSource", "EIN1");
    grabber.StreamPort.set("EndOfScanTriggerSource", "ScanLength");
    grabber.StreamPort.set("ScanLength", 1000);
  } else {
    console.log("Configuring master card");
    grabber.RemotePort.set("TestPattern", "GreyDiagonalRampMoving");
    grabber.RemotePort.set("CxpLinkConfiguration", "CXP6_X4");
    grabber.RemotePort.set("CxpLinkConfigurationPreferredSwitch", "CXP6_X4");
    grabber.RemotePort.set("TriggerSource", "CXPin");
    grabber.RemotePort.set("TriggerMode", "On");
    grabber.DevicePort.set("CameraControlMethod", "RG");
    grabber.DevicePort.set("ExposureReadoutOverlap", "True");
    grabber.DevicePort.set("CxpTriggerAckTimeout", "0");
    grabber.DevicePort.set("StrobeDuration", "0");
    grabber.DevicePort.set("ExposureTime", "20");
    grabber.DevicePort.set("ExposureRecoveryTime", "0");
    grabber.DevicePort.set("CycleMinimumPeriod", "50");
    // configure start/end of scan triggers
    grabber.StreamPort.set("StartOfScanTriggerSource", "Immediate");
    grabber.StreamPort.set("EndOfScanTriggerSource", "ScanLength");
    grabber.StreamPort.set("ScanLength", 1000);
  }
}
```



备注 在本例中，扫描启动触发，即从主卡收到扫描启动事件，但扫描结束触发是在本地生成的。另一种选择是将EIN2 as EndOfScanTriggerSource。

11.4. 平场校正

适用于: QuadG3 QuadG3LH Octo

什么是平场校正?	164
Coaxlink FFC	167
FFC向导示例程序	170

什么是平场校正？

平场校正(FFC)是一种用于校正的方法：

- 照机像素传感器之间感光度的差异
- 视野中照明强度的差异
- 光通过透镜传输的差异(例如：虚光)

目标是校正捕获的(Raw)图像的像素，以便当系统(相机和镜头)捕获统一的背景时，生成的输出图像是统一的。

公式

通过对原始图像的每个像素应用以下操作，可以实现此校正：

$$\text{CorrectedPixel} = (\text{RawPixel} - \text{Offset}) * \text{Gain}$$

其中，Offset and Gain 和系数，都是每个像素的特定值。

校准程序

校准程序评估每个像素的Offset and Gain 和FFC系数，是通过计算两个参考图像的像素数据：**暗图像**和**平面图像**。



警告 如果系统的任何部分发生变化(包括相机、照明或光学设备)，则必须重新执行校准程序。

暗图像采集

暗图像(A.K.A. 暗框)是在黑暗中拍摄的图像。这样的图像代表传感器的**暗电流**，被认为是我们在正常情况下获取图像时要消除的固定偏差。要获得暗图像：

1. 用盖子盖住透镜。
2. 获取多个图像，并计算阵列每个像素的平均响应

平面图像采集

平面图像，是覆盖整个视野的一致性目标图像。要获得平面图像：

1. 放置一个宽度足以覆盖整个视野的一致性目标(彩色相机为白色)
2. 调整镜头光圈和照明强度，以获得可能最亮的图像(前提是没有像素饱和)。
3. 获取多个图像，并计算阵列每个像素的平均响应

FFC补偿系数计算

对于每个像素，Offset系数实际上是**暗图像**的像素值。



备注 这种校正也称为**暗帧减法**。

FFC增益系数计算

暗图像和**平面图像**为计算FFC系数Gain提供了足够的**数据**。

为此，我们定义了 CorrectedPixel as the pixel value we consider to be correct for the flat image. Let's set this value as the average pixel value of the flat image (average(Flat)), corrected by the average of the dark image (average(Dark)). 在FFC术语中，这给出了：

$$\text{average(Flat)} - \text{average(Dark)} = (\text{FlatPixel} - \text{DarkPixel}) * \text{Gain}$$

这就产生了Gain值。

$$\text{Gain} = (\text{average(Flat)} - \text{average(Dark)}) / (\text{FlatPixel} - \text{DarkPixel})$$

相同的计算重复(width * height)次，以覆盖平面和暗图像的所有像素。这将导致对图像的每个像素进行特定的校正。

彩色像素格式校准程序

对于彩色像素格式，我们有多种方法计算 `average(Flat)`。In all cases, the Gain computation is repeated (`width * height * componentsPerPixel` 的值，以覆盖所有像素分量，从而对图像的每个像素分量进行特定的校正。

单独处理像素分量

I.e. 在RGB中；

- 使用 `average(Flat[Red])` for computing the Gain values of Red分量)；
- 使用 `average(Flat[Green])` for computing the Gain values of Green分量)；
- 使用 `average(Flat[Blue])` for computing the Gain values of Blue分量)；

一起处理像素分量

I.e. 在RGB, using `average(average(Flat[Red]), average(Flat[Green]), average(Flat[Blue]))` for computing the Gain values of Red, Green and Blue分量)。

这种计算平均值的方法(即在像素分量上)所产生的FFC系数，也可以校正分量之间的平衡。因此，根据用于获取平面图像的一致性背景的质量，FFC可以有效地进行白平衡校正。

Coaxlink FFC

在Coaxlink产品中平面场校正的实现

有些相机有一个内置的FFC模块，而其他相机则不能实现此功能，但是没有此功能的设备，可以通过Coaxlink卡的FFC核心进行校正。

Coaxlink固件的FFC核心，通过应用FFC - 使用与图像中的位置对应的系数(Offset and Gain) - 来修正直接来自相机的像素。由于校正是发生在Coaxlink像素处理链的早期阶段，因此其他像素处理功能，比如RedBlueSwap, LUT, 和Bayer Decoding是在校正后的像素上执行的。

增益和偏移系数格式

校准过程中所计算的系数，可以加载到CoaxLink卡中，前提是它们编码如下：

一个像素分量的Offset and Gain 值打包成一个16位的尾数值：

- Gain 在 UQ2.8 中以9..0位编码。
- Offset 是位15..10上的6位无符号整数。

与像素分量值相关的系数，按照与图像像素分量相同的顺序分别处理。例如，在RGB8 format, one pixel is encoded as 3 successive 8-bit values (Red, Green, Blue), therefore we need 3 successive 16-bit packed coefficients to correct one RGB8在格式中，一个像素被编码为3个连续的8位值(, ,),因此，我们需要3个连续的16位压缩系数来更正一个像素。

如果16位压缩系数(按顺序)存储在二进制文件中(比如'path/to/coefficients.ffc'), 则可以通过调用从Euresys脚本中轻松加载它们。

```
require("coaxlink://ffc/load")(grabber, 'path/to/coefficients.ffc');
```

此处grabber引用了要配置捕获器的脚本变量。



备注 这样的二进制文件可以由Euresys `ffc-wizard`示例应用程序创建。

功能可用性

下表列出了支持FFC的产品/固件变体组合，及其主要特征：

产品/固件变体	固件变体	CoaXPress 接口	DRAM 分区
1633 Coaxlink Quad G3 1633-LH Coaxlink Quad G3 LH	1-camera	CXP-6 DIN4	512 MB用于FFC系数存储 512MB用于FIFO数据缓冲区
3602 Coaxlink Octo	2-camera	CXP-6 DIN4	512 MB用于FFC系数存储 512MB用于FIFO数据缓冲区

另见： "Firmware Variants per Product" 于页面20 并搜索 `FFC`，该搜索在 `高级处理` 列中。

规范

相机类型

FFC功能适用于单色、Bayer CFA和RGB彩色区域扫描相机，每像素组件提供8位、10位、12位、14位或16位数据。

最大图像大小

下表显示了，启用FFC时，所有支持的像素格式的最大图像大小：

位深度	最大图像大小[像素]		
	单色和Bayer CFA格式	RGB 格式	RGBa 格式
8-位	134,217,728 像素	44,739,242 像素	33,554,432 像素
10-位	107,374,182 像素	35,791,394 像素	26,843,545 像素
12-位	89,478,485 像素	29,826,161 像素	22,369,621 像素
14-位	76,695,844 像素	25,565,281 像素	19,173,961 像素
16-位	67,108,864 像素	22,369,621 像素	16,777,216 像素

性能

启用后，FFC功能会向DRAM内存添加大量负载，因为它会为每个处理的像素获取额外的16位系数数据。当启用FFC时，Coaxlink卡只能维持CoaXPress链接可实现的最大数据速率的一小部分。这个无维值被称为“**可持续的相对数据率**”

下表显示了支持FFC的所有位深度和产品/固件变体组合的**可持续相对数据速率**。s

可持续相对数据率

位深度	可持续的相对数据率(a 4-lane CXP-6 最大数据率的百分比)]	
	1633 Coaxlink Quad G3 - 1-camera 1633-LH Coaxlink Quad G3 LH - 1-camera	3602 Coaxlink Octo - 2-camera
8-位	70.0 %	123.2 %
10-位	77.8 %	136.9 %
12-位	84.0 %	147.8 %
14-位	89.1 %	156.8 %
16-位	93.3 %	164.3 %



备注 “可持续的相对数据率”对于所有连接到板上的相机都是全局的。例如，在**3602 Coaxlink Octo -2-camera 8-bit**用例中，123.2%的可持续数据率可以不相等地拆分为，一个相机为100%和其他相机为23.2%的。



备注 CoaxLink卡在消隐间隔期间不获取任何数据。行和帧消隐间隔不消耗内存带宽，因此在计算相机数据速率时，必须将其排除在外。

为了避免延迟、FIFO缓冲区溢出，和帧丢失，Euresys建议相应地限制(全局)相机数据速率。

启用FFC

在数据流模块中，将 **FfcControl** 特征值设置为 **Enable**。

禁用FFC

在数据流模块中，将 **FfcControl** 特征值设置为 **Disable**。

FFC向导示例程序

Euresys提供了一个示例应用程序的源代码，名为`ffc-wizard`，该应用程序计算系数，并将其打包到针对Coaxlink卡的二进制文件中。

此示例代码的目的有三个方面：

1. 指导用户完成校准程序；
2. 提供本文件所述内容的技术和实践翻译；
3. 提供用于开发自定义应用程序的构建基块。

构建

源代码位于单个源文件中：`src/ffc-wizard.cpp`。构建应用程序应该简约；

- 对于Windows，有一个Microsoft Visual Studio项目文件：`ffc-wizard.vcproj`；
- 对于Linux和MacOS，提供了Makefile。

使用

向导是一个控制台应用程序。当给出--help标志时，将显示以下帮助消息。

```
> ffc-wizard.exe --help
Flat Field Correction Wizard
ffc-wizard [OPTIONS]
Options:
  --if=INT           Index of GenTL interface to use
  --dev=INT          Index of GenTL device to use
  --ds=INT           Index of GenTL data stream to use
  --average=INT      Number of images to average (default: 10)
  --roi_x=INT        Horizontal offset in pixels of ROI upper-left corner (default: 0)
  --roi_y=INT        Vertical offset in pixels of ROI upper-left corner (default: 0)
  --roi_width=INT    Width of ROI (default: whole image)
  --roi_height=INT   Height of ROI (default: whole image)
  --balance          Compute flat image average on all components rather than on each
component
  --dark-setup=SCRIPT Path to setup script for dark acquisitions
  --flat-setup=SCRIPT Path to setup script for flat acquisitions
  --timeout=MS       Maximum time in milliseconds to wait for an image (default: 1000)
  --dark-histogram=FILE Path to histogram html page of average dark image to output and open
  --flat-histogram=FILE Path to histogram html page of average flat image to output and open
  --output-ffc=FILE  Path to coefficients output file (Coaxlink ffc binary format)
  --load-ffc=FILE    Load coefficients into Coaxlink coefficients partition (default: computed
coefficients)
  --no-interact      Skip user interaction
  -h --help          Display help message
Note: the ROI options allow defining a rectangular region to consider while computing averages,
      this is useful to eliminate pixels close to borders in images subject to vignetting.
```

选项详情

标志	详情
--if, --def, --ds	标识要使用(和/或配置)的数据流的GenTL模块索引。
--average	为暗采集和平面采集而采集的图像数量;校准过程中仅进一步使用这些采集的平均值。
--roi_x, --roi_y, --roi_width, --roi_height	计算平面图像的平均像素值时要考虑的可选矩形区域(average(Flat));这会影响对每个像素的增益值的评估。
--balance	启用白平衡;即无论是否计算彩色像素分量的系数来平衡分量值;显然,这要求用于获取平面图像的平面背景尽可能接近真实的灰色(所有RGB分量的值都相同)。
--dark-setup, --flat-setup	在黑暗和平坦采集之前,运行的可选配置脚本
--dark-histogram, --flat-histogram	可选;输出文件的路径,显示暗图像和平面图像像素分量的柱状图;这提供了暗电流变化以及平面图像变化的直观概述。
--no-interact	通常,向导会等待用户在开始暗采集和平坦采集之前设置系统;当使用此标志时,向导则不会等待用户(也不会打开创建的直方图HTML页)。

设计

校准程序以及系数的打包由主函数 `ffcWizard` 控制。

ffcWizard 任务

任务	由函数完成
获取一系列暗图像，以生成平均暗图像	<code>acquireImages</code>
获取一系列平面图像，以生成平均平面图像	<code>acquireImages</code>
计算每个像素分量的值 <code>gain</code> 。	<code>computeGain</code>
使用暗像素分量值作为 <code>offset</code> 值	<code>computeOffset</code>
将 <code>offset</code> and <code>gain</code> 值打包为16位的小端值	<code>packCoefficients</code>
将压缩系数写入二进制文件	<code>savePackedCoefficients</code>

定制

示例应用程序已经可以支持一些常见的像素格式：`Mono`, `RGB`, `RGBa` and `Bayer`。

限制: 为了限制示例应用程序的复杂性，我们考虑(对于每个像素具有多个分量的像素格式)所有分量具有相同的大小。通过更新函数，仍然可以支持具有不同分量大小的像素格式。`addImage` and `addComponents`。

为了支持新的像素格式(在先前的限制条件下)，我们需要修改两个功能：

1. `Image::getComponentsPerPixel`, 返回由 **PFNC** 名称识别的新格式的每个像素的分量数。
2. `Image::getComponentFilters`, to return a `std::vector` of `ComponentFilter` 描述新格式(由其PFNC名称识别)像素分量在图像中位置的对象。

该 `ComponentFilter` objects are used to separate the processing of the different pixel components while evaluating the `Gain` and `Offset` values. For example, in `RGB` format, the FFC coefficients related to the `Red` components are computed using the `Red` 组件来自黑色和平面图像中。

有关像素分量布局配置的详细信息，请参见源代码 `Image::getComponentFilters`。

11.5. 查找表处理

LUT处理简介	174
LUT处理可用性	175
单色查找表格处理	176
LUT内容定义	177
LUT设置程序	184

LUT处理简介

Coaxlink 驱动程序在选定的一组产品/固件组合上提供查找表功能。

另见: "LUT处理可用性" 在下一页 (一份详尽的清单)。

这个版本的Coaxlink 驱动程序, 专门为单色像素格式提供查找表处理。

另见: 有关详细说明, 请参阅"单色查找表格处理" 于页面176。

Coaxlink 驱动程序, 提供了四种方法来定义查找表的内容。

另见: "LUT内容定义" 于页面177

另见: "LUT设置程序" 于页面184 设置查阅表格。

LUT处理可用性

具有 *LUT* 选项的所有固件变体都支持以下 *LUT* 配置：

- LUT M_8x8, M_10x8, M_10x10, M_10x16, M_12x8, M_12x12, M_12x16.

另见：“[Firmware Variants per Product](#)”于页面20 并搜索 *LUT*，该搜索在 高级处理 列中。

单色查找表格处理

配置

下表列出了单色像素的所有可用查找表格配置：

配置	输入像素格式[PFNC]	输入位	输出位
M_8x8	Mono8, Raw	8	8
M_10x8	Mono10	10	8
M_10x10			10
M_10x16			16
M_12x8	Mono12	12	8
M_12x12			12
M_12x16			16



备注 单色8-位像素可转换为单色8-位像素，单色10-位像素可转换为单色-8位、10-位或16-位像素，单色12-位像素可转换为单色8-位、12-位或16-位像素。

查找表格数据集

查找表格数据集定义为，为像素的每个分量配置一个查找表格所需的数据集。对于单色像素，查找表格数据集仅包含一个查找表格内容。

可以上载的查找表格数据集的数量，取决于查找表格配置：

配置	数据集
M_8x8	16
M_10x8, M_10x10, M_10x16	4
M_12x8, M_12x12, M_12x16	1

LUT内容定义

Coaxlink 驱动程序，提供了四种方法来定义查阅表格的内容。

响应控制方法

此方法通过四个参数来定义查找表格的传递函数：**Contrast**，**Brightness**，**Visibility**和**Negative**。

Contrast和**Brightness**参数，提供类似于电视监视器的亮度和对比度控制的控制。

Visibility参数，提供控制，以平滑地改变传递函数的形状来覆盖整个输入范围。

Negative参数，允许将图像转换为负图像。

强调法

该方法通过两个参数**Emphasis**和**Negative**来定义查找表的传递函数。

它允许使用幂律表达式(也称为 γ -Gamma-函数)转换图像。

Negative参数，允许将图像转换为负图像。

阈值法

该方法通过五个参数：**SlicingLevel**，**SlicingBand**，**LightResponse**，**BandResponse**，**DarkResponse**，定义了一个双阈值变换律。

表格法

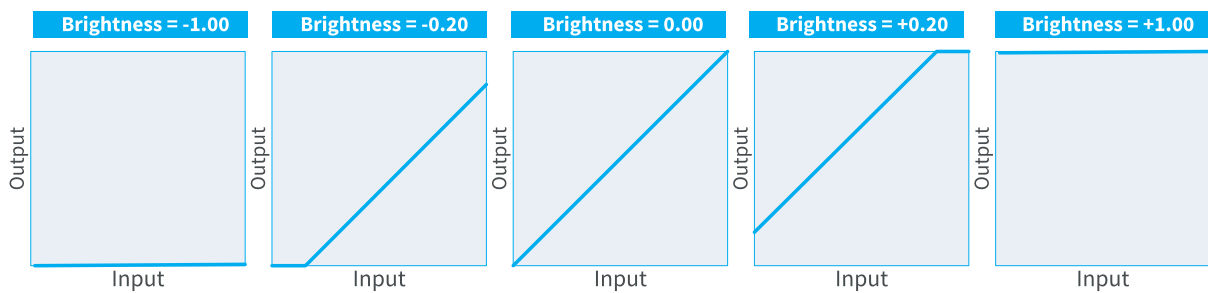
此方法以表格形式，定义查阅表格的传递函数。

亮度参数

Brightness参数，仅适用于“响应控件”查找表定义方法。

它实现了一种类似于电视监视器亮度控制的控制。

亮度	备注
-1.00	最小值： 最暗输出。整个输入范围数据被转换成全黑。此规则适用于任何选定的 对比 值。
0.00	默认值。 将0.5的中间级输入电平转换为0.5的相同输出电平。对于其他参数的任何值都是这样的。 亮度增加到+1.00会导致输出变亮。 亮度降低到-1.00会导致输出变暗。
+1.00	最大值。 最淡输出。整个输入范围的数据转换成全白。此规则适用于任何选定的 Contrast 值。



当所有其他控件设置为其默认值时，**Brightness** 的影响：

Brightness = Brightness; Brightness = Brightness; Brightness = Brightness.

对比度参数

Contrast参数，仅适用于“响应控件”查找表定义方法。

它实现了一种类似于电视监视器对比度控制的控制。

变换律的斜率是增益，它是由参数**Contrast**非线性控制的。

在数学上，这种关系是：

$$\text{增益} = 10^2 \times (\text{对比度} - 1)$$

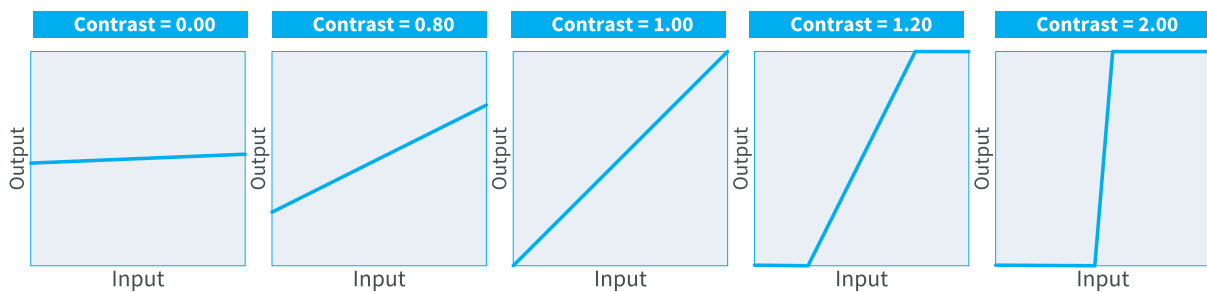
对比度	增益	备注
0.00	0.01	最小Contrast值；最小收益
1.00	1	默认Contrast值；单位增益
2.00	100	最大值。Contrast值；最大收益

为了达到所需的给定增益，对比度控制应设置为：

$$\text{对比度} = 1 + (\log_{10} \text{增益}) / 2$$

如果所需增益以分贝 (dB) 表示：

$$\text{对比度} = 1 + \text{增益 (dB)} / 40$$



当所有其他控件设置为其默认值时，**Contrast** 的影响：

Contrast = Contrast; Contrast = Contrast; Contrast = Contrast

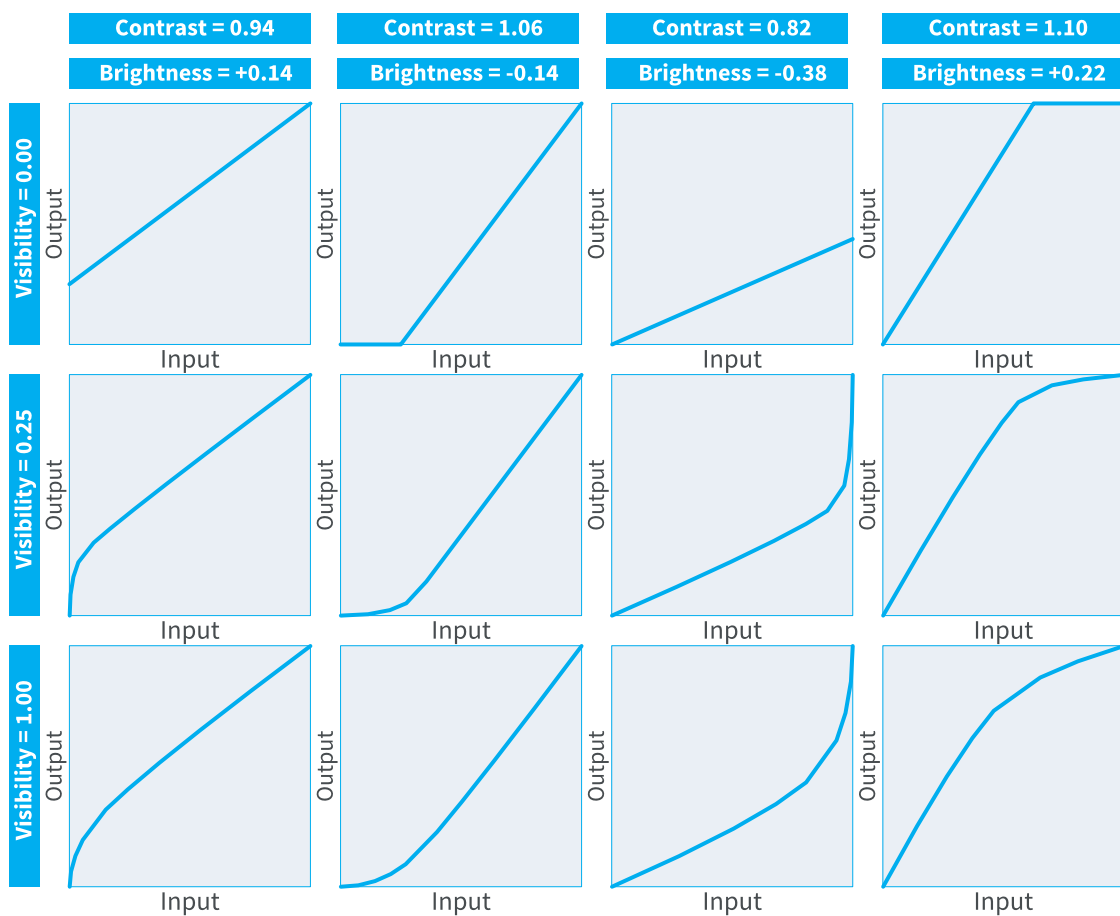
可见性参数

Visibility 参数，仅适用于“响应控件”查找表定义方法。

参数 **Contrast** 和 **Brightness** 的操作偶尔会删除输入动态的某些部分。图像中非常暗的区域，可以转化为全黑，并且变得不可见。这适用于非常明亮的区域，剪裁为全白色。

已创建该 **Visibility** 参数，以平滑地显示图像中的这些隐藏部分。

能见度	增益	备注
0.00	0.01	最小值和默认值。 这将生成分段线性变换曲线。选择接近+1的值会生成更平滑的曲线。
1.00	100	最大值。



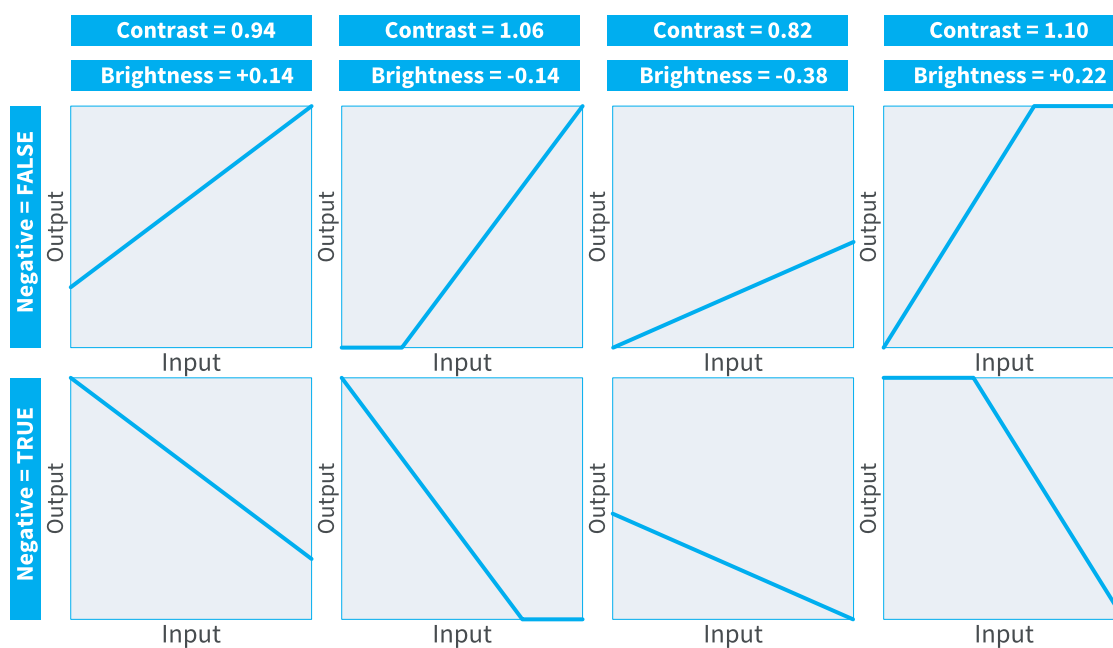
Visibility 对 **Contrast** 和 **Brightness** 参数典型值的影响-假设 **Negative = FALSE**

负参数

该 **Negative** 参数，同时适用于“响应控件”和“强调”查找表定义方法。

此控件允许将图像转换为负图像，其中图像中最亮的区域显示最暗，最暗的区域显示最亮。

负的	备注
FALSE	默认值。
TRUE	转换表在图中围绕垂直轴镜像。这会交换黑白值，并产生照相底片效果。



对其他控制的标准值的 **Negative** 影响

重点参数

该 **Emphasis** 参数，仅适用于“强调”查找表定义方法。

它允许使用幂律表达式来转换图像：

$$\text{输出} = \text{输入}^\gamma$$

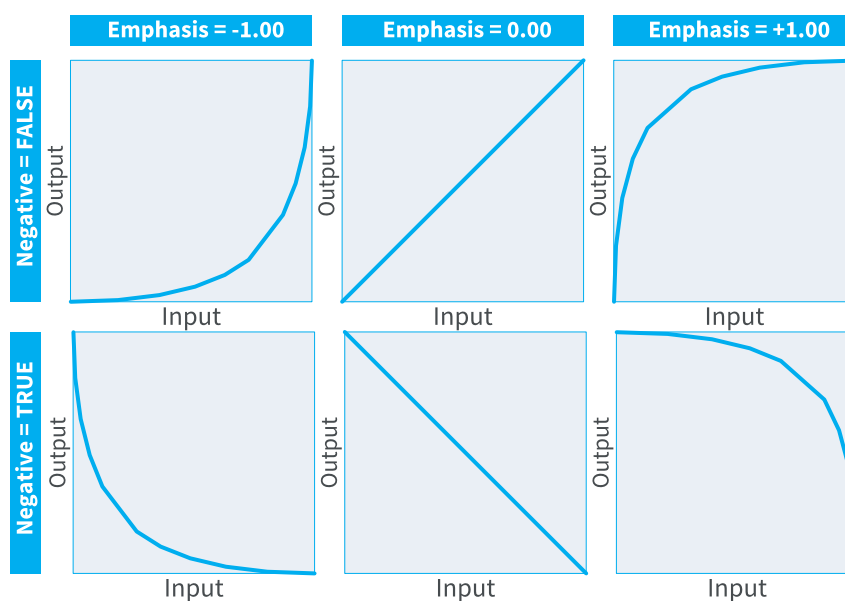
γ -Gamma-指数在数学上与“强调”联系在一起：

$$\gamma = 10^{-\text{强调}}$$

强调	Gamma	备注
1.00	0.1	最大值。强调值；最小 γ 值
0.00	1	默认强调值；线性规律
-1.00	10	最小强调值；最大 γ 值

为达到所需的给定 γ ，Emphasis 应设置为：

$$\text{强调} = -\log_{10}\gamma$$



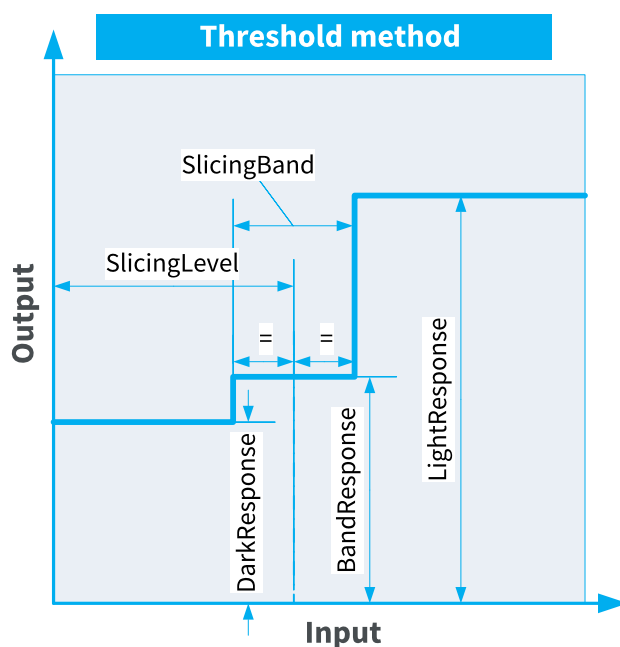
Emphasis 的典型值和 **Negative** 的两个值的强调效果。

阈值方法参数

, **SlicingBand**, **LightResponse**, **BandResponse**和 **DarkResponse**参数, 仅应用于“阈值”查找表定义方法。

如下图所示, 参数集定义了一个双阈值转换定律。

参数	最小值	默认值	最大值
切片电平	0.00	0.50	1.00
切片带	0.00	0.50	1.00
光响应	0.00	0.75	1.00
带响应	0.00	0.50	1.00
暗响应	0.00	0.25	1.00



双阈值传递函数



备注 **SlicingLevel** 指定在输入范围内的两个阈值的平均值。

LUT设置程序

要设置查找表处理，请执行以下操作：

1. 禁用查找表格
2. 定义查找表格配置
3. 定义查找表格的内容
4. 将查找表格内容上载到指定的查找表格数据集中
5. 使用指定的数据集来启用查找表格

禁用查找表格

要禁用查找表格：

- 将功能 **LUTEnable** 设置为 **Off**。

定义查找表配置

要定义查阅表格配置，请根据以下内容设置 **LUTConfiguration** 功能：

- 相机像素类型和位深
- 所需的输出位深。

另见：有关适用于单色像素的配置，请参阅“单色查找表格处理”于页面176。



备注 必须在执行任何其他操作之前，设置查找表格配置。

定义查找表格内容

另见：有关用于定义查阅表格内容的参数化和表格化方法的说明，请参阅“LUT内容定义”于页面177。



备注 必须至少定义一个查找表格集。

上载查找表格内容

要在一个操作中上载查找表格内容，请执行以下操作：

- 通过为 **LUTSet** 功能指定适当的值，选择要访问的查找表格数据集。例如 **Set1**。
- 将 **LUTIndex** 功能设置为 **0**。
- 将一个 **LUTLength** 值字符串，写入 **LUTValue** 功能。



备注 应用程序还可以有选择地上载任何单个查找表格条目，或任何连续查找表格条目块。

读回查找表格数据集

要在一个操作中读回查找表格数据集，请执行以下操作：

- 通过为 **LUTSet feature. For instance Set1**指定适当的值，选择要访问的查找表格数据集。
- 将 **LUTIndex** 功能设置为 **0**。
- 将 **LUTReadBlockLength**功能设置为**LUTLength**返回的值。
- 获取一个**LUTReadBlockLength**值字符串(从功能**LUTValue**中)。



备注 应用程序还可以有选择地单独读取任何查阅表格条目，或连续条目的任何块。

启用查找表格

要启用查找表格：

- 将**LUTEnable**功能设置为指定要使用的查找表格数据集的值。

配置脚本示例

下面的脚本是一个示例，演示如何为单色8-位到8-位的操作配置查找表格，以及如何使用不同的查找表格定义方法，来定义和上载4个查找表格数据集。

```
function configure(g) {
  // Disable the lookup table
  g.StreamPort.set('LUTEnable', 'Off');
  // Configure the lookup table
  g.StreamPort.set('LUTConfiguration', 'M_8x8');

  // Build lookup table data set 1: response control
  g.StreamPort.set('LUTSet', 'Set1');
  require('coaxlink://lut/response-control')(g, { Contrast: 0.94
                                                , Brightness: 0.14
                                                , Visibility: 0.25
                                                , Negative: false });

  // Build lookup table data set 2: emphasis
  g.StreamPort.set('LUTSet', 'Set2');
  require('coaxlink://lut/emphasis')(g, { Emphasis: 0.5
                                          , Negative: true });

  // Build lookup table data set 3: threshold
  g.StreamPort.set('LUTSet', 'Set3');
  require('coaxlink://lut/threshold')(g, { SlicingLevel: 0.5
                                           , SlicingBand: 0.5
                                           , LightResponse: 0.75
                                           , BandResponse: 0.5
                                           , DarkResponse: 0.25 });

  // Build lookup table data set 4: table
  g.StreamPort.set('LUTSet', 'Set4');
  var i;
  for (i = 0; i < 256; ++i) {
    g.StreamPort.set('LUTIndex', i);
    g.StreamPort.set('LUTValue', String(255 - i));
  }
}
configure(grabbers[0]);
```

11.6. Bayer CFA解码

适用于: QuadG3 QuadG3DF Octo QuadCXP12J

Bayer CFA到RGB转换	188
使用Bayer CFA解码器	192

Bayer CFA到RGB转换

Bayer CFA解码器将摄像机发出的原始Bayer CFA数据流转换成RGB颜色数据流。

使用以下插值方法之一，从最近的分量，重建缺失的像素分量：

- 该传统插值方法通过专门应用Mean()函数来计算缺少的颜色分量。
- 该高级插值方法使用Mean()和Median()函数计算缺少的颜色分量。高级插值消除了图像中高度对比急剧过渡的混叠效应。

函数定义

该min()函数从一组2个整数值中返回最低整数值。

该max()函数从一组2个整数值中返回最高整数值。

该mean()函数返回一个整数值，即2个整数的平均值。计算如下：

函数	Mean(a,b)
公式	$(a + b + 1) \gg 1$

该median()函数返回一组两个整数值，即一组四个整数的两个中值。计算如下：

函数	Median(a,b,c,d)
值1公式	Min [Max(a,b); Max(c,d)]
值2公式	Max [Min(a,b); Min(c,d)]

解码器操作

对于源图像的每个像素，CFA解码器从周围像素计算两个缺失的颜色分量。

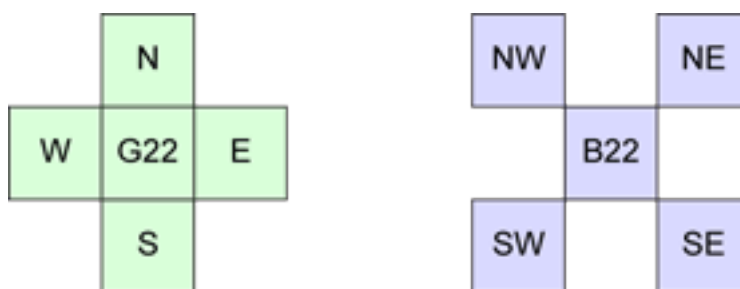
下文描述了如何计算(位于4x4 Bayer CFA阵列的22、32、23和33位置)4个中心像素：

B11	G21	B31	G41
G12	R22	G32	R42
B13	G23	B33	G43
G14	R24	G34	R44

周围像素的相对位置，通过罗盘标记来识别：

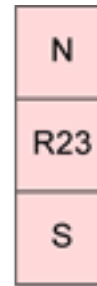
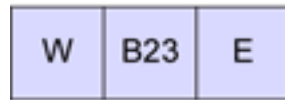
NW	N	NE
W		E
SW	S	SE

位置22的公式



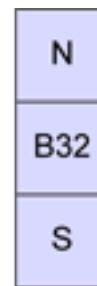
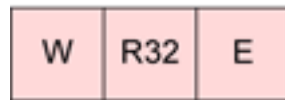
分量	方法	公式
R22	传统, 高级	R22
G22	传统	Mean[Mean(N,S), Mean(W,E)]
	高级	Mean[Median(N, S, E, W)]
B22	传统	Mean[Mean(NW, SW), Mean(NE, SE)]
	高级	Mean[Median(NW, SW, NE, SE)]

位置23的公式



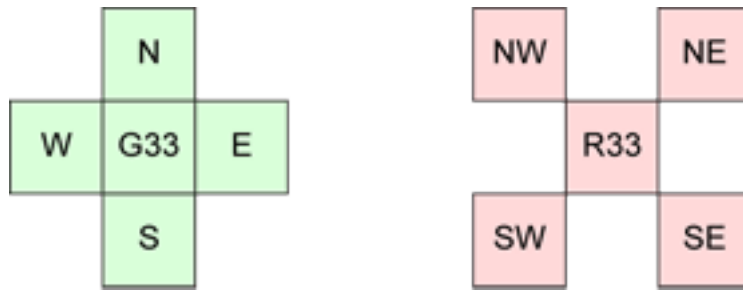
分量	方法	公式
R23	传统, 高级	Mean(N,S)
G23	传统, 高级	G23
B23	传统, 高级	Mean(W,E)

位置32的公式



分量	方法	公式
R32	传统, 高级	Mean(W,E)
G32	传统, 高级	G32
B32	传统, 高级	Mean(N,S)

位置33的公式



分量	方法	公式
R33	传统	$\text{Mean}[\text{Mean}(\text{NW}, \text{SW}), \text{Mean}(\text{NE}, \text{SE})]$
	高级	$\text{Mean}[\text{Median}(\text{NW}, \text{SW}, \text{NE}, \text{SE})]$
G33	传统	$\text{Mean}[\text{Mean}(\text{N}, \text{S}), \text{Mean}(\text{W}, \text{E})]$
	高级	$\text{Mean}[\text{Median}(\text{N}, \text{S}, \text{E}, \text{W})]$
R33	传统, 高级	B33

使用 Bayer CFA 解码器

先决条件

1. 帧捕获器

带有 CFA 解码器的 Coaxlink 卡和固件变体：打开 ["Firmware Variants per Product"](#) 于 [页面 20](#) 并搜索 CFA，该搜索在 [高级处理](#) 列中。

2. 摄像机

a. Bayer CFA area-scan

b. 每行最多 16384 像素

c. 进行第一传输线的前两个第一传输像素的颜色校正 (GR、RG、GB 或 BG)，并在 CoaXPress 图像头的 PixelF 字段中，正确指定像素位深度 (8 位、10 位、12 位、14 位或 16 位)。



警告 当字段 `Xoffs` 和/或 `Yoffs` 大于 0 时，摄像机必须报告与传输数据相对应的调整后的 `PixelF` 值！

Bayer 至 RGB 像素处理配置

当 Bayer CFA 解码器被启用时

- 该 ["像素分量解包"](#) 于 [页面 63](#) 控制器不工作，Coaxlink 卡将 10 位、12 位或 14 位像素解包到 lsb
- 该 ["像素分量重新排序"](#) 于 [页面 65](#) 该功能允许交换红色和蓝色分量，并提供 BGR 或 RGB 像素

输入像素格式	FFC	RedBlueSwap	输出像素格式
Bayer**8	关闭/打开	关闭	RGB8
		打开	BGR8
Bayer**10pmsb	关闭/打开	关闭	RGB10
		打开	BGR10
Bayer**12pmsb	关闭/打开	关闭	RGB12
		打开	BGR12
Bayer**14pmsb	关闭/打开	关闭	RGB14
		打开	BGR14
Bayer16	关闭/打开	关闭	RGB16
		打开	BGR16

启用 Bayer CFA解码器

在数据流模块中，根据所需的插值方法，将 **BayerMethod** 特征值设置为 **Legacy** 或 **Advanced** according to the desired interpolation method.

禁用 Bayer CFA解码器。

在数据流模块中，将 **BayerMethod** 特征值设置为 **Disable**。

Bayer CFA解码器性能

产品	固件变体	峰值像素处理速率
1633 Coaxlink Quad G3 1633-LH Coaxlink Quad G3 LH	1-camera	1,000 Mpixels/s
1635 Coaxlink Quad G3 DF	1-df-camera	1,000 Mpixels/s
3602 Coaxlink Octo	1-camera	2,000 Mpixels/s
	2-camera	1,000 Mpixels/s 每流



备注 Bayer CFA解码器的峰值像素处理速率，与交付RGB8或BGR8像素时DMA传输的性能近似匹配。它明显低于主板可实现的最大聚合输入数据速率！



警告 当PCI Express可用带宽不足或在处理非常短的行时，有效像素处理速率可能与峰值像素处理速率有很大不同。

PCI快速性能

PCI快速接口	可持续PCIe数据速率	RGB8	RGB10, RGB12, RGB14, RGB16
4-通道版本3.0 PCIe 端点	3, 350 Mb/秒(典型的)	~1,117 Mpixels/s	~558 MPixels/s
8-通道版本3.0 PCIe 端点	6,700 MB/s(典型的)	~2,238 Mpixels/s	~1,117 Mpixels/s



备注 PCI接口不是8位位深的瓶颈！



警告 对于10位、12位、14位和16位位深，主机PC上PCI Express Interface的性能进一步限制了可持续的数据输出速率。

等待时间

当数据吞吐量不受可用PCI Express带宽的限制时，硬件CFA解码器以可忽略的延迟，执行动态转换！

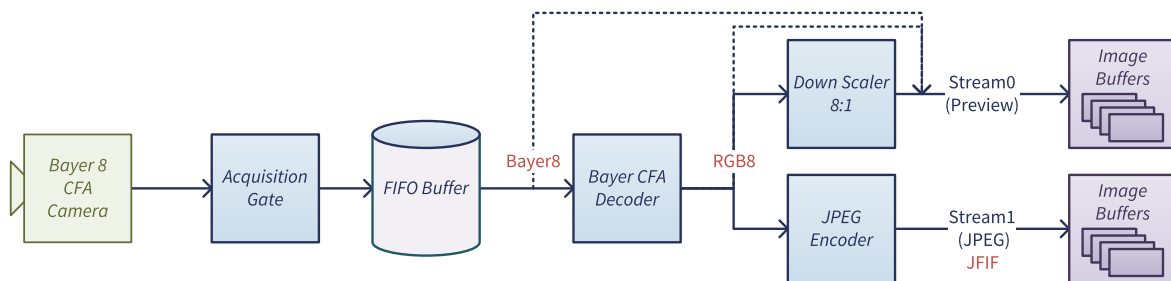
11.7. JPEG 编码

适用于：[QuadCXP12J](#)

该 **3620 Coaxlink Quad CXP-12 JPEG** 是一款四连接的 CoaXPress CXP-12 帧捕获器，配有嵌入式CFA解码器和JPEG编码器。

该 **4-相机** 固件变体实现4个独立通道，并由以下部分组成：

- 一个具有 PoCXP 和相机触发功能的单连接 CoaXPress CXP-12 主机接口，
- 选择要捕获帧的采集门，
- 一个 512MB FIFO 缓冲器，提供大的弹性缓冲器，
- 一个拜耳CFA解码器，其使用高级算法将拜耳8相机数据流转换成 RGB8 流，
- 一个JPEG编码器，其包括一个 RGB8 至 YUV422 转换器前端和一个符合标准 ISO/IEC 10918-1 的JPEG编码器，
- 向下缩放器，从两个方向上将RGB8图像分辨率降低8倍。



相机要求

- 8位拜耳 CFA 区域扫描彩色相机，
- 图像分辨率(HxV)：从128×16到5120×3840，宽度和高度都是8的倍数，
- CXP-1到CXP-12 CoaXPress 接口。

通道规格

- 像素处理速率：高达2.5亿像素/秒
- 输出流：2(“预览”和“JPEG”)
- 编码器控制：JPEG 质量

预览流规格

- 像素格式: RGB8 或 拜尔8
- 分辨率: “全分辨率”或“8:1”(仅适用于RGB8)

JPEG 流规格

- JPEG控制: 质量从1到100(默认: 75)
- 像素格式: CustomJFIF
- 分辨率: “全分辨率”
- 延迟: 一般20行

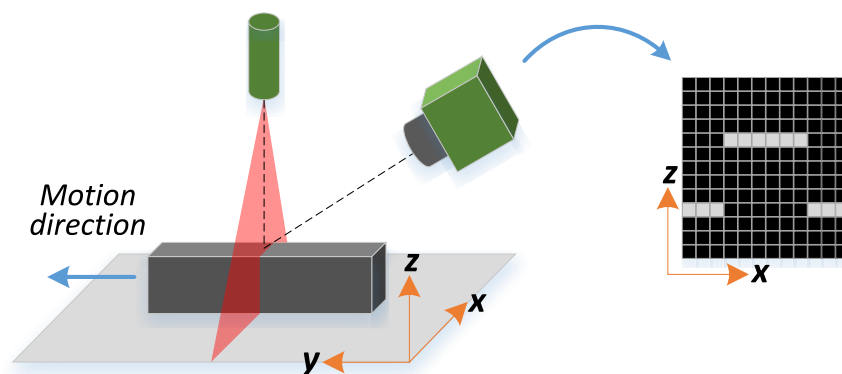
11.8. 激光线提取

适用于: **Quad3DLLE**

简介	197
LLE处理核心实现	199
用例示例	208
附录	215

简介

在激光三角测量系统中，激光线(或任何其他窄光带生成方法)投影在三维物体上。然后使用放置在激光器另外视角的相机来捕获由该对象形状导致变形的线图像。线变形是3D对象在激光线平面中形状的直接表示。通过扫描对象，使其在激光线下移动并拍摄多张图像，您可以重建其3D形状。



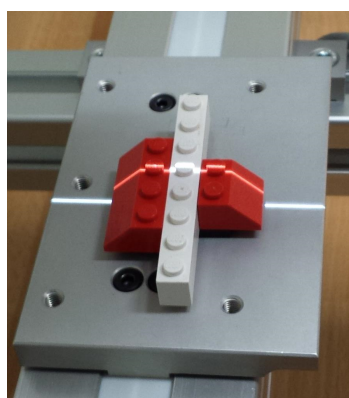
简易激光三角测量系统

激光线提取算法

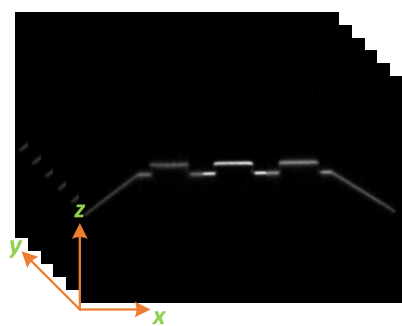
需要一种激光线提取 (LLE) 算法, 根据相机传感器捕获的物体的轮廓序列, 来创建深度图。

LLE算法的目的是估计激光线水平交叉感兴趣区域(ROI)的位置。检测可以通过单独分析帧的每一列来完成。

LLE算法通常输出一个数据阵列, 其中包含沿ROI检测到的激光线的垂直位置, 即每个计算的ROI产生一个单一的数据阵列。

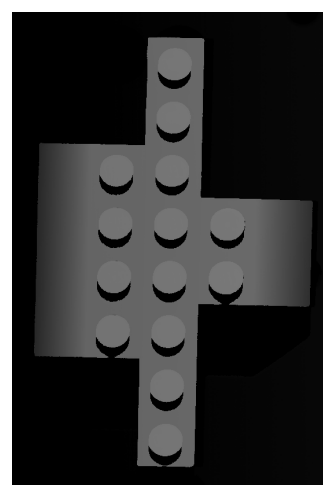


Measured object



Series of profiles

LLE



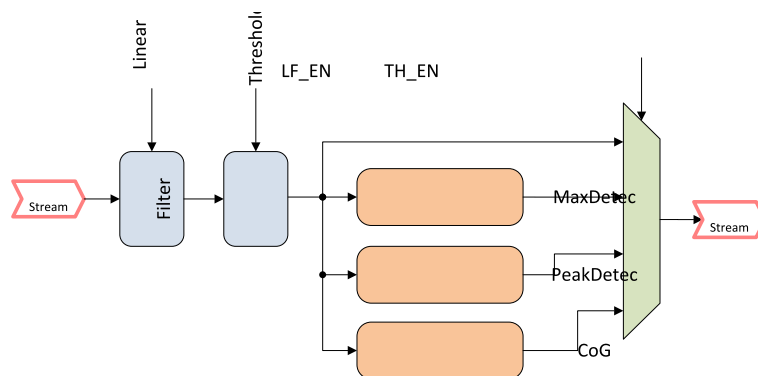
Depth Map

深度图生成

LLE处理核心实现

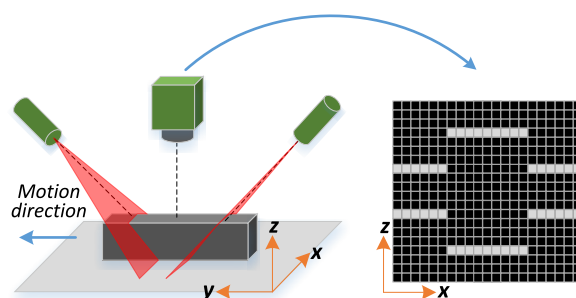
LLE处理核心嵌入**1637 Coaxlink Quad 3D-LLE**到帧捕获器中。它可以计算被测对象的深度图(使用零主机CPU和零延迟)。

它提供了3种激光线提取算法(最大检测, 峰值检测, 和重心), 以及过滤和阈值的预处理阶段。



LLE处理核心简化框图

LLE处理核心还能够从一个单个的输入ROI同时计算多达2个深度图。这一特性称为双激光线提取 (*Dual-LLE*)。此功能对于使用投射在同一物体上的两条激光线的应用非常有用。



简化的双激光三角测量系统

当双LLE模式被激活时，LLE处理核心输入将输入ROI分为两个子ROI，称为LLE-ROI。每个LLE-ROI都由LLE处理核心独立处理，形成相应的深度图。

LLE处理核心特性

绝对最大LLE-ROI XSize

- 所有算法: 8192像素

最大值。有效LLE-ROI YSize[像素]

MaxDetec 8	MaxDetec 16	PeakDetec 11_5	PeakDetec 8_8	CoG 11_5	CoG 8_8
255	65535	2048	256	2048	256

输出格式[GenICam PFNC v2.1]

MaxDetec 8	MaxDetec 16	PeakDetec 11_5	PeakDetec 8_8	CoG 11_5	CoG 8_8
8位(整数)	16位(整数)	UQ11.5	UQ8.8	UQ11.5	UQ8.8

输出像素格式

MaxDetec 8	MaxDetec 16	PeakDetec 11_5	PeakDetec 8_8	CoG 11_5	CoG 8_8
Coord3D_C8	Coord3D_C16	Coord3D_C16	Coord3D_C16	Coord3D_C16	Coord3D_C16

输出亚像素分辨率[像素]

MaxDetec 8	MaxDetec 16	PeakDetec 11_5	PeakDetec 8_8	CoG 11_5	CoG 8_8
1	1	1/32	1/256	1/32	1/256

InvalidDataFlag 值

- MaxDetec 8算法: 0x00
- 其他算法: 0x0000



备注 此值标识一个无效的结果。

有效输出范围

- MaxDetec 8 算法: 0x01 ~ 0xFF
- 其他算法: 0x0001 ~ 0xFFFF

支持的输入像素格式

- 所有算法Mono8

激光线数

- 所有算法: 每个LLE-ROI 1个(每个输入ROI最多2个)

最大性能

- 所有算法: 2500万像素/秒
- 所有算法: 来自2048x256或4096x128图像的9,500个配置文件/秒
- 所有算法: 来自2048x128图像的19,000个配置文件/秒
- 所有算法: 来自1024x128图像的38,000个配置文件/秒
- 所有算法: 来自1024x64图像的76,000个配置文件/秒



备注 上述数字基于4连接CXP-6 Coaxress链路的最大带宽,并考虑到双LLE模式已激活。

可用输出类型

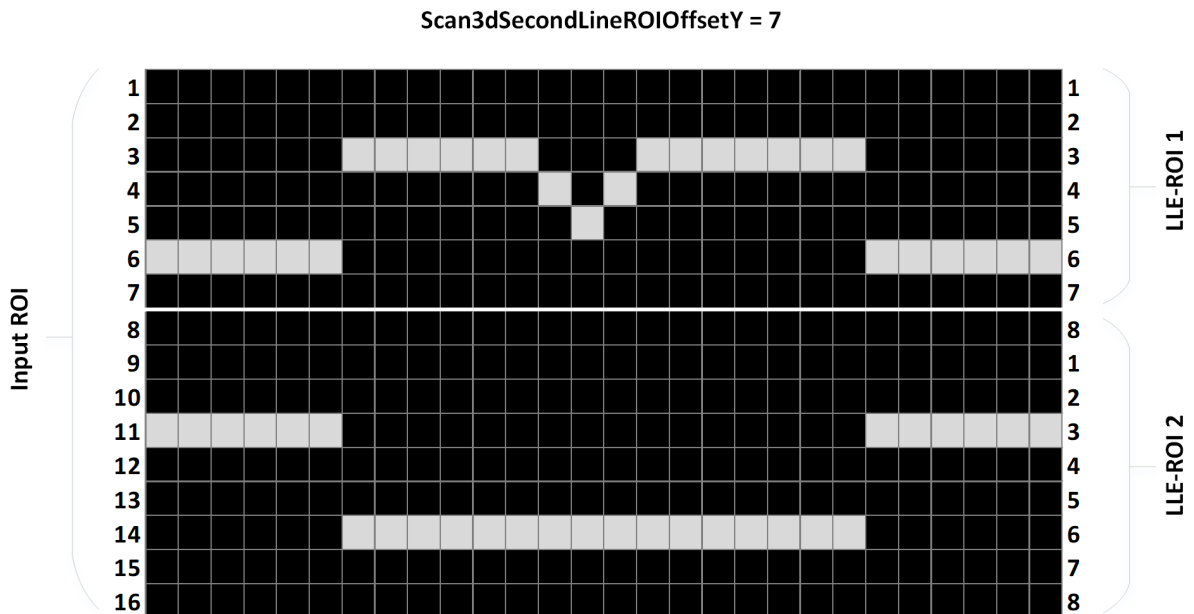
- 所有算法:
 - 深度图
 - 原始相机图像



警告 不是两个输出同时进行!

双激光线提取

双LLE模式可通过将数据流功能 **Scan3dSecondLineROIOffsetY** 设置为不同于 **0** 的值来激活。该特征 **Scan3dSecondLineROIOffsetY** 对应于一个偏移量，该偏移量定义了将输入ROI拆分为两个LLE ROI的位置。

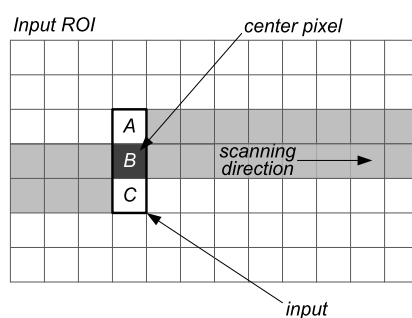


ROI偏移示例

线性滤波器

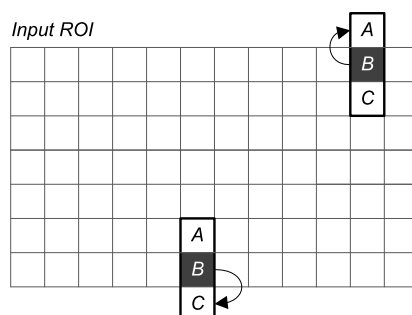
线性滤波器模块在1X3滑动窗口上应用卷积运算符，当数据流来自相机时，直接作用于数据流。卷积内核的3个元件(A、B和C)可以配置为接受任何正整数，其中3个元件的和是1到512之间的值。

此图说明了卷积内核元件在给定LLE-ROI中的位置：



线性滤波器内核配置

当内核的一个元件位于LLE-ROI边界之外时(典型的滑动窗口问题), 其输入像素值将被中心窗口像素替换。



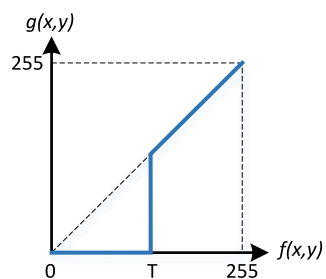
LLE-ROI边界处的线性滤波器行为。

取芯门槛

取芯是一种非线性算子，它执行一种简单的分割技术，根据以下规则将像素分为两类：

$$g(x, y) = \begin{cases} f(x, y), & f(x, y) > T \\ 0, & \text{otherwise} \end{cases}$$

T是取芯门槛。



取芯响应

最大检测

最大检测模块，可以配置为以8位或16位数据宽度输出深度图。这两种模式的区别在于，最大检测模块支持的最大有效LLE-ROI YSize。8位模式可以表示高达255像素的高度，16位模式可以表示高达65535像素的高度。

当在给定的LLE-ROI列的多个像素中检测到最大强度时，最大检测算法将指明位置最高的像素。

InvalidDataFlag

当出现以下情况时，将生成一个*InvalidDataFlag*。

- 检测到的线位置，高于最大LLE-ROI高度。
- 在LLE-ROI列中未检测到最大强度。

峰值检测

峰值检测模块生成由16位定点字表示的深度图。它可配置两种精度：

- *UQ11.5* 其中5位 (LSB) 表示小数部分，11位 (MSB) 表示整数部分。在这种模式下，峰值检测模块支持的最大有效LLE-ROI YSize 几乎为2048像素。
- *UQ8.8*, 其中8位 (LSB) 表示小数部分，8位 (MSB) 表示整数部分。在这种模式下，峰值检测模块支持的最大有效LLE-ROI YSize 几乎为256像素。

当在给定的LLE-ROI列上检测到多个峰值时，峰值检测模块将指明相应的 $f(x)$ 像素具有最高强度的位置。如果多个相应的 $f(x)$ 像素具有相同的条件(即最高强度)，则会指明其中位置最高的像素。

InvalidDataFlag

当出现以下情况时，将生成一个*InvalidDataFlag*：

- 检测到的线位置，高于支持的最大LLE-ROI高度。
- 在LLE-ROI列中未检测到任何行。

重心

重心模块生成由16位定点字表示的深度图。可配置两种精度：

- *UQ11.5*, 其中5位(LSB)表示小数部分, 11位(MSB)表示整数部分。在这种模式下, 由重心模块支持的最大有效LLE-ROI YSize几乎为2048像素。
- *UQ8.8*, 其中8位(LSB)表示小数部分, 8位(MSB)表示整数部分。在这种模式下, 由重心模块支持的最大有效LLE-ROI YSize几乎为256像素。

当在一个给定的LLE-ROI列上检测到多个峰值时, 重心模块将指示相应的 $f(x)$ 像素具有最高强度的位置。如果多个相应的 $f(x)$ 像素具有相同的条件(即最高强度), 则会指明其中位置最高的像素。

InvalidDataFlag

当出现以下情况时, 将生成一个 *InvalidDataFlag*:

- 检测到的线位置, 高于支持的最大LLE-ROI高度。
- 在内部和中, 发生溢出。当在同一个LLE-ROI列中, 大量连续像素的强度高于阈值水平时, 就会出现这种情况。
- 在LLE-ROI列中未检测到任何行。

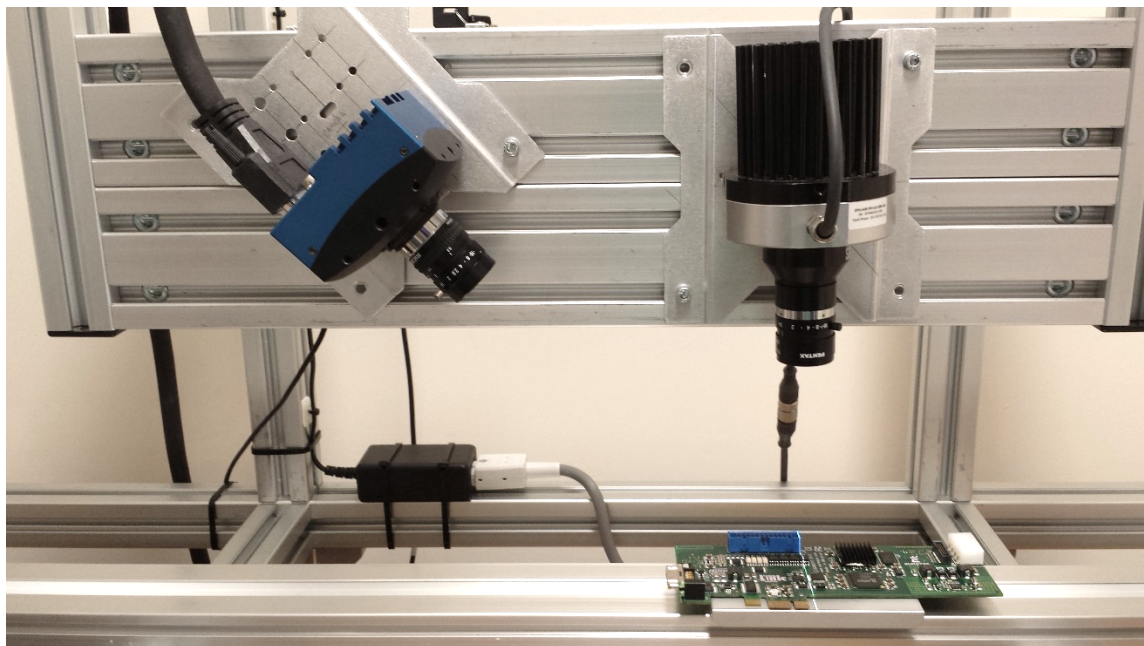
用例示例

目标	209
定义输入ROI	210
设置过滤和阈值	211
定义LLE算法	212
定义扫描长度和缓冲区大小	214

目标

在此示例中，目标是获得PCB表面的可靠测量。为了简化，使用GenICam浏览器设置采集参数，相机不受控制(自由运行)。其他参数，如三角测量几何、透镜、激光颜色和功率，在本例中不涉及。

此图显示了本例中使用的激光线三角测量设置：

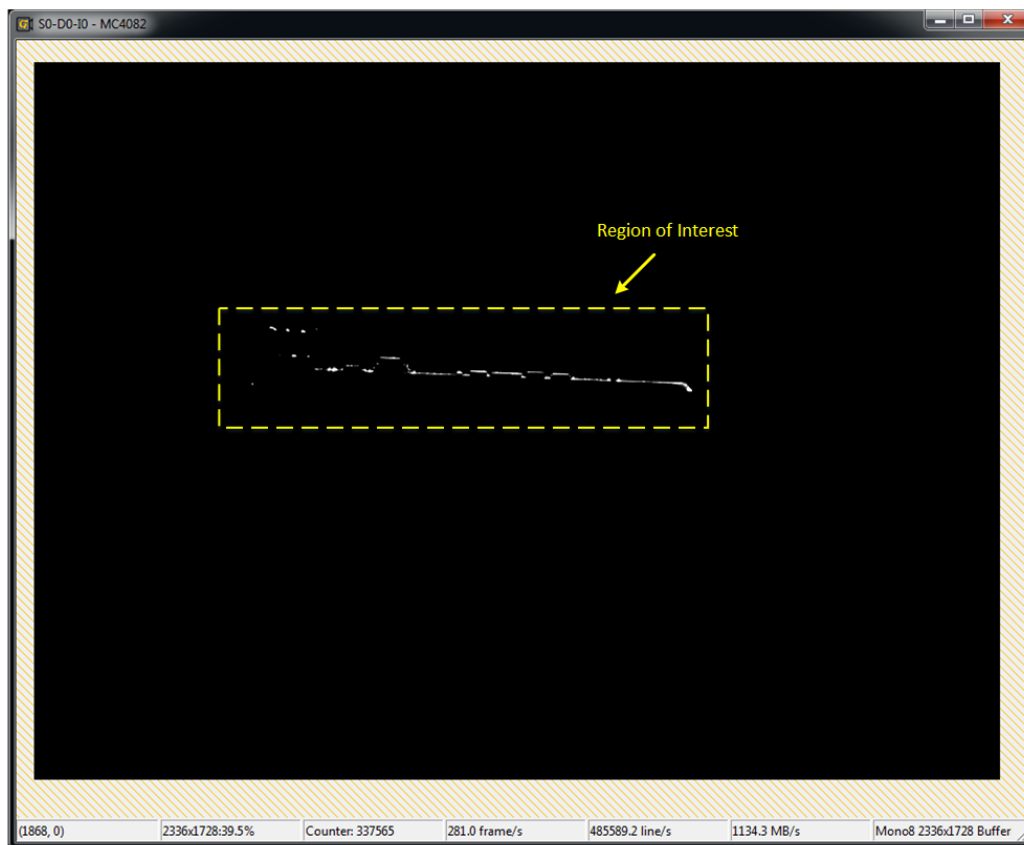


激光线三角测量设置

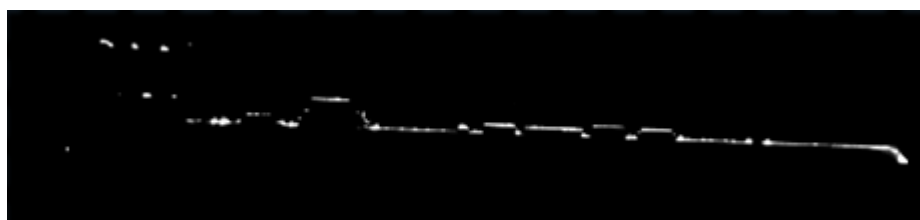
定义输入ROI

定义的输入ROI, 应覆盖发生激光线变化的整个传感器区域。最初的步骤是, 设置一个全分辨率的ROI来验证整个图像。

如下图所示, 可以验证激光线变化发生的有效区域。



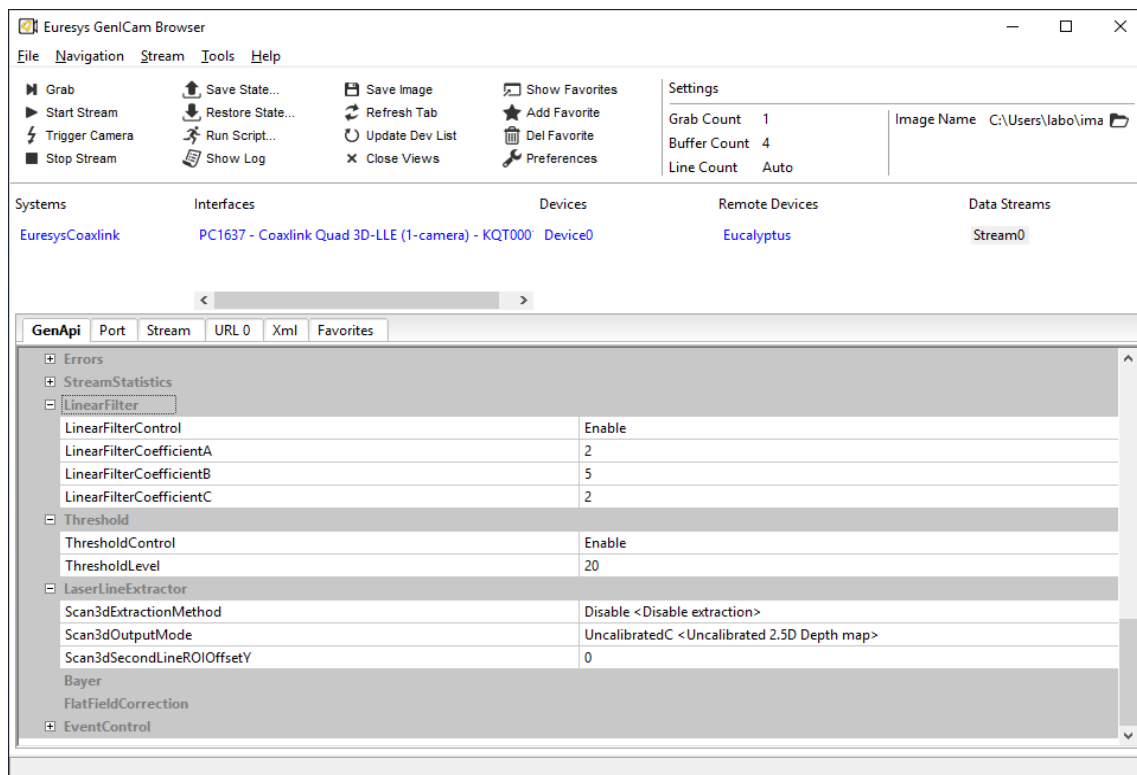
结果图像(全分辨率ROI)



输入ROI扩展视图

设置过滤和阈值

可以在LLE过程中激活预处理阶段。在下面的示例中，阈值级别设置为20，卷积滤波器内核设置为A=2、B=5和C=2。使用这些参数，我们将得到一个更平滑的图像，其中任何低于阈值的背景噪声都将被0替换。

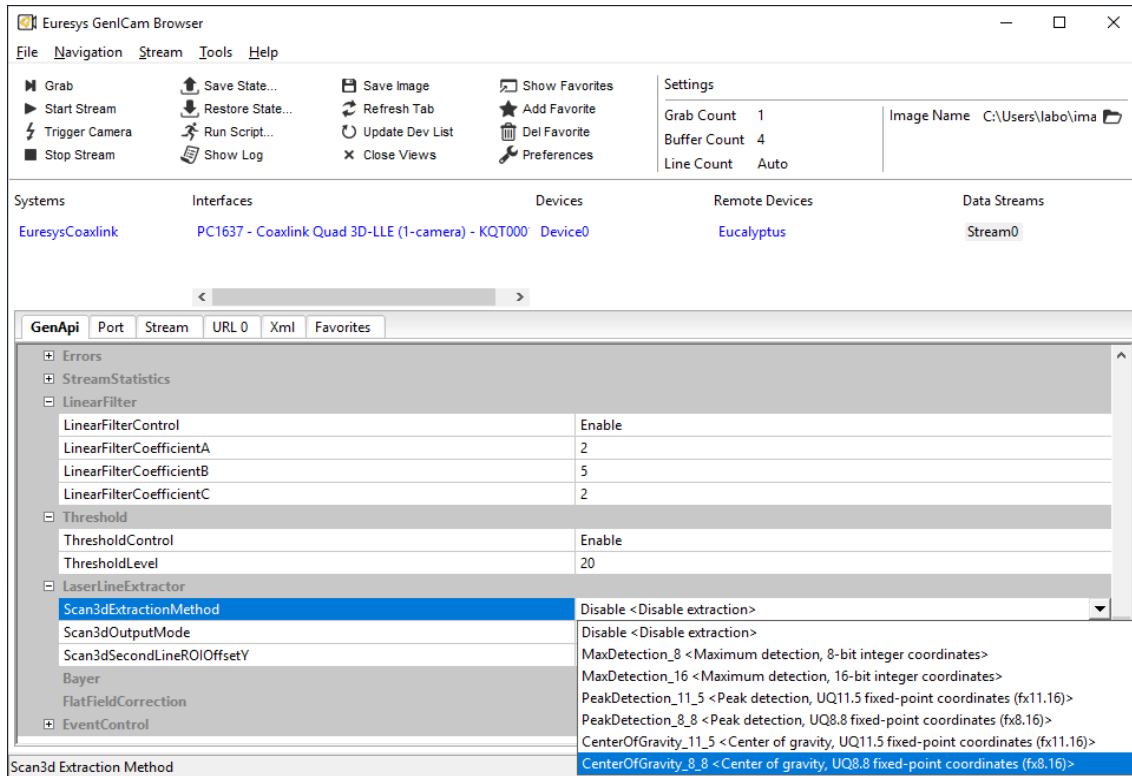


滤波和阈值参数

定义LLE算法

GenICam 功能，**Scan3dExtractionMethod** 提供所有可用的LLE算法。

下图显示了如何在GenICam 浏览器中显示此功能。



激光线提取选项

可用算法之间的选择，取决于与目标应用程序相关的许多因素。我们在这里强调现有算法之间的主要差异。

LLE-ROI YSize

LLE-ROI(ysize) 中有效垂直像素的数量，受可由所选算法的输出格式表示的最大整数值的限制，如下所述：

- **MaxDetection_8** (8位) : 0xFF(255像素) 。
 - 保留值 0x00 以指示 InvalidDataFlag。
- **MaxDetection_16** (16位) : 0xFFFF(65535像素) 。
 - 保留值 0x0000 以指示 InvalidDataFlag。
- **PeakDetection_11_5** (UQ11.5) : 0xffff(几乎 2048像素) 。
 - 保留值 0x0000 以指示 InvalidDataFlag。
- **PeakDetection_8_8** (UQ8.8) : 0xffff(几乎 256像素) 。
 - 保留值 0x0000 以指示 InvalidDataFlag。

- **CenterOfGravity_11_5** (UQ11.5) : 0xffff(几乎2048像素)。
 - 保留值0x0000以指示InvalidDataFlag。
- **CenterOfGravity_8_8** (UQ8.8) : 0xffff(几乎256像素)。
 - 保留值0x0000以指示InvalidDataFlag。

算法分辨率

可用算法的位置分辨率从1像素到1/256像素不等，如下所述：

- **MaxDetection_8** (8位) : 1像素。
- **MaxDetection_16** (16位) : 1像素。
- **PeakDetection_11_5** (UQ11.5) : 1/32像素。
- **PeakDetection_8_8** (UQ8.8) : 1/256像素。
- **CenterOfGravity_11_5** (UQ11.5) : 1/32像素。
- **CenterOfGravity_8_8** (UQ8.8) : 1/256像素。

算法特性

如前所述，LLE算法的选择很大程度上取决于目标应用程序。下面，从应用的角度，对每个LLE算法的主要特征进行描述：

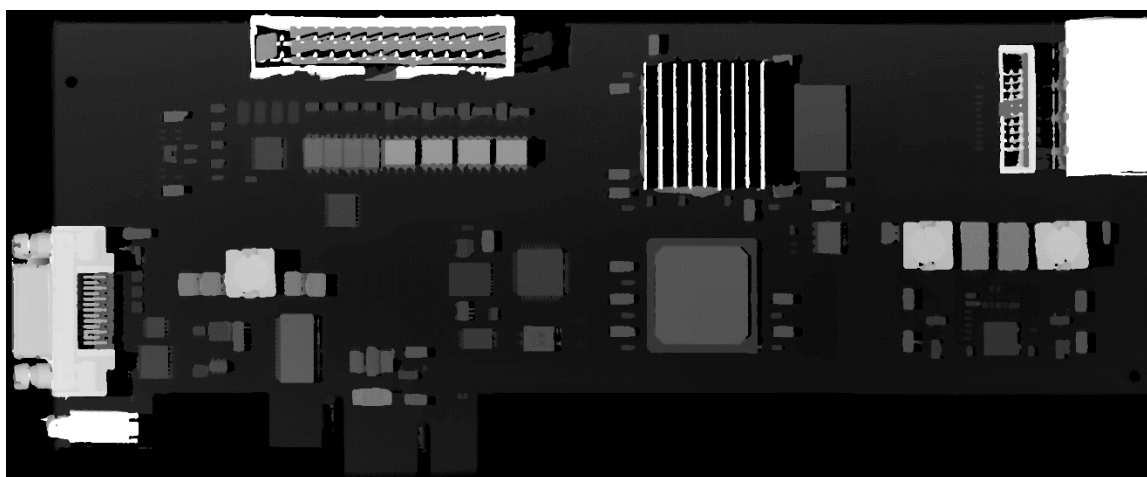
- **MaxDetection_8** (8位) : 此版本的MaxDetection模块具有8位深度图输出，生成的数据量非常小，3D后处理阶段的计算性能要求更低。
- **MaxDetection_16** (16位) : 通过16位整数深度图输出，MaxDetection模块的版本可以支持非常大的输入LLE-ROI。
- **PeakDetection**: 两个版本的PeakDetection 模块在亚像素分辨率和最大有效LLE-ROI大小之间进行权衡。PeakDetection 被设计为对强度依赖的偏差不敏感。
- **CenterOfGravity**: 两个版本的CenterOfGravity 模块，可在亚像素分辨率和最大有效LLE-ROI大小之间的权衡。CenterOfGravity 被设计成对噪声输入和宽线具有鲁棒性。

定义扫描长度和缓冲区大小

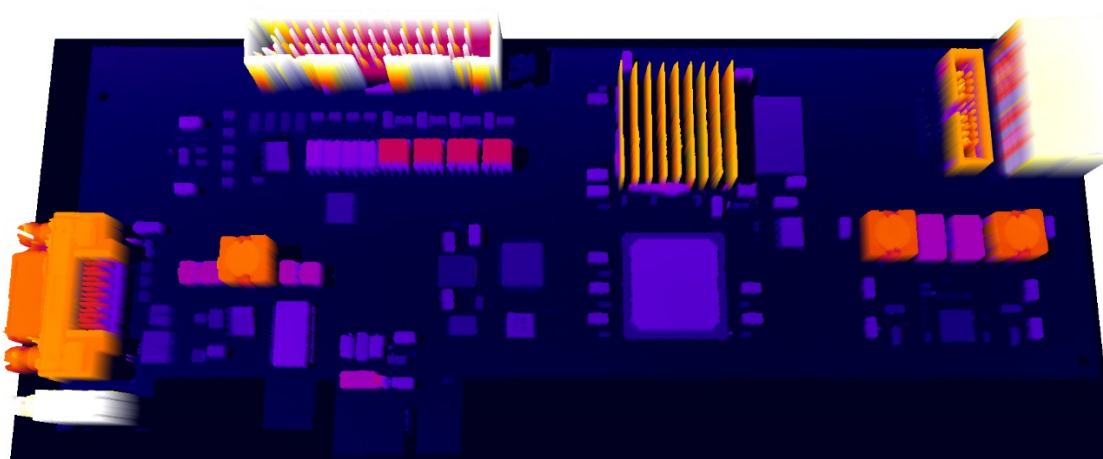
ScanLength 和 BufferHeight 将取决于从给定对象获取的所需配置文件数量。扫描对象的大小、相机的帧速率和对象移动速度是随 ScanLength 定义的基本参数。

扫描结果

测得的 PCB 深度图如下图所示。



深度图示例



深度图示例：三维曲面图

附录

定点“Q格式”符号

UQm.n系统是以Q-格式表示的定点数，其中Q以Q-格式表示数字，Q前面的U表示无符号值，M表示数字整数部分的位数，N表示数字的小数部分。例如，UQ11.5是一个无符号定点值，由16位字表示，其中11个整数位和5个小数位。UQ11.5的其他有效符号是fx11.16和0:11:5。

如何将UQ11.5定点值转换为浮点值

激光线拔取器产生的UQ11.5定点值是16位字，代表11个整数位和5个小数位的值。

要从该格式转换为float，首先将值转换为float，然后除以 $2^5=32$ 。

```
float toFloat(无符号 短 x) {  
    浮点F=x;  
    返回F / 32;  
}
```

如何将“UQ8.8”定点值转换为浮点值

激光线拔取器产生的UQ8.8定点值是16位字，代表8个整数位和8个小数位的值。

要从该格式转换为float，首先将值转换为float，然后除以 $2^8=256$ 。

```
float toFloat(无符号 短 x) {  
    浮点F=x;  
    返回F / 256;  
}
```

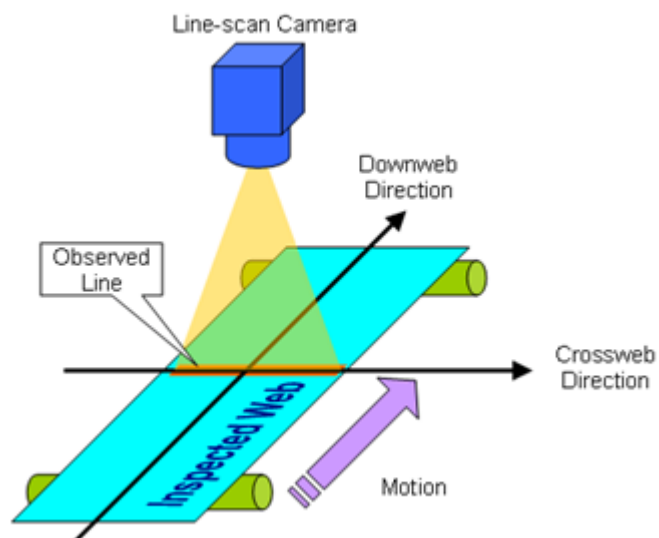
11.9. 行扫描采集

线扫描采集原理
线扫描采集用例

217
221

线扫描采集原理

行扫描成像



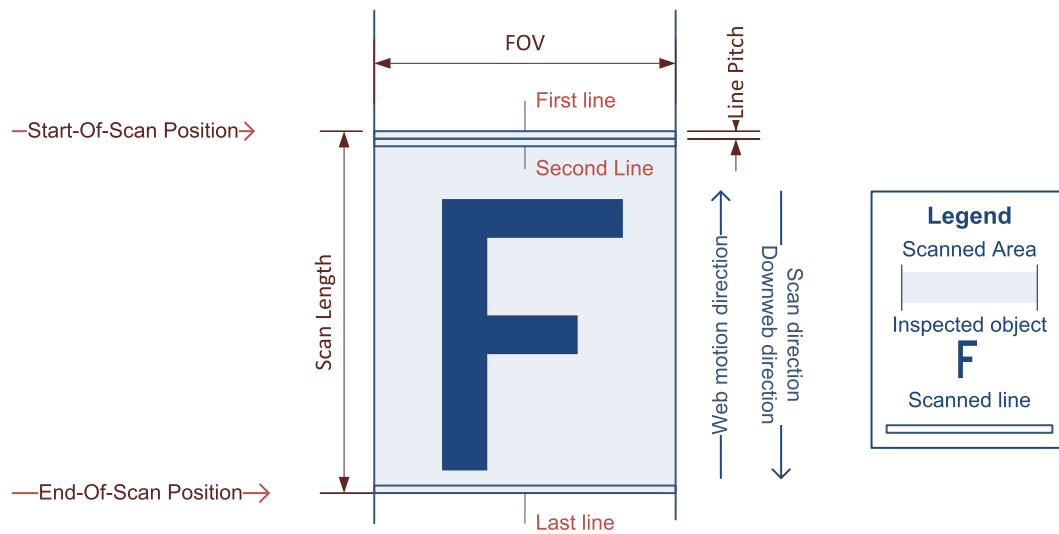
典型线扫描成像系统

表达式“行扫描成像”指定机器视觉应用，其中二维图像是由(从相对移动到对象的一维成像设备捕获的)连续图像行组合获得的。

在行扫描成像中：

- 成像设备通常(但非必须)是行扫描照相机。
- 被检查的对象通常是一个连续的网络，它也可以是具有固定或可变大小的离散对象。
- 被检查的网络相对于相机移动。采集过程中的运动速度可以是固定的或可变的。
 - 跨网方向或横方向是相机观察到的网平面上的轴。
 - 下网方向或轴向是被检网相对于相机的运动方向。

扫描面积



扫描区域定义

扫描区域是Web上的二维区域，其宽度等于FOV，长度等于扫描长度。

在横网方向(上图中的水平方向)，扫描区域由相机的视场(FOV)分隔。

在下网方向(上图中的垂直方向)，扫描区域由“扫描开始”和“扫描结束”位置分隔。线间距由网速与相机线速之比决定。

线扫描相机的**视场(FOV)**仅由光学装置和传感器几何特性决定。

开始扫描位置，是对应于第一采集行之前的扫描行边界的Web上的位置。

扫描结束位置，是对应于最后一条采集线之后的扫描线边界的Web上的位置。

大多数行扫描相机，在每个相机循环提供一行像素。因此，需要**多个相机循环**来构建对象图像。

像素纵横比控制

与区域扫描成像不同，行扫描成像允许应用程序控制图像像素纵横比。

在大多数情况下，成像应用程序需要一个常量，最好是1:1图像像素纵横比。

跨Web节距由传感器节距和光学放大系数锁定，图像像素纵横比仅通过线节距控制来控制。

下表总结了提供适用于Coaxlink的恒定线间距的方法：

方法名称	描述
VCR	<p>可变相机循环。</p> <p>网络速度是可变的以及相机循环速率与网络速度成比例。</p> <p>帧捕获器通过捕捉相机传送的所有连续的线来构建目标图像。</p>
CCC	<p>恒定的相机循环。</p> <p>相机以恒定的循环速率运行，帧捕获器捕获相机传送的所有连续行。</p>

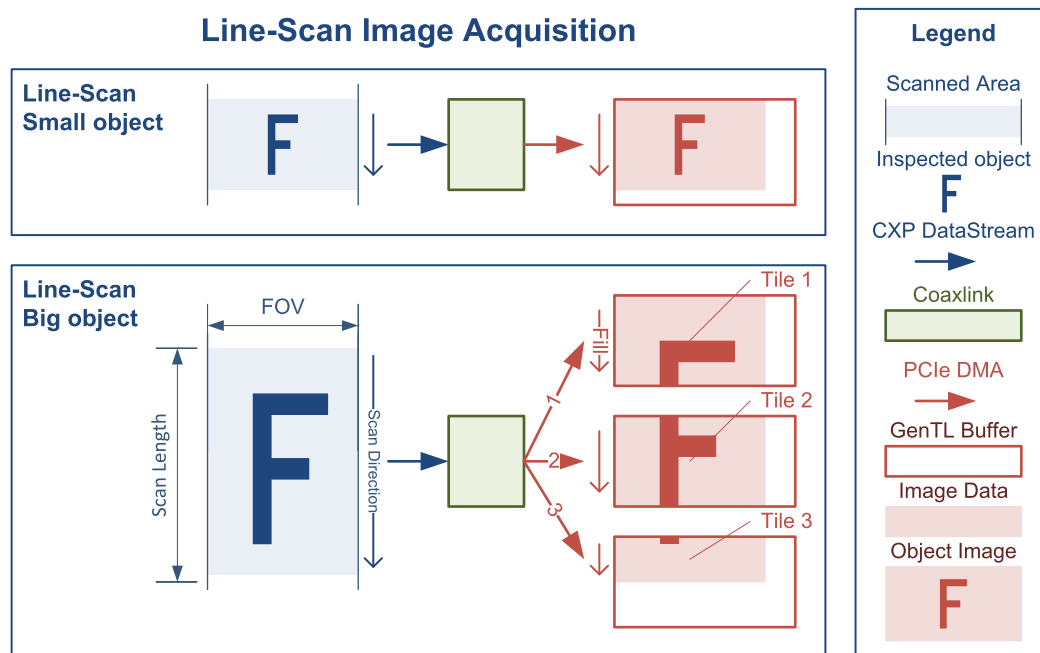
VCR方法要求：

- 用于测量网速的**运动编码器**。
- 运动编码器事件的**实时处理**，以与运动编码器事件速率**成比例**的速率建立相机触发。

成比例的速率，可以通过**除法器工具**或**乘法器/除法器工具**实现：

- 除法器工具以整数值抽取输入速率，它提供N个输入事件中的1个。
- 乘法器/除法器工具允许任何速率转换比值(RCR)在0.001至1000范围内，精度优于RCR值的0.1%，从而实现了对图像像素纵横比的精细控制。

使用线扫描成像设备的图像采集



对于CoaxPress链接上的传输, (大多数) 行扫描相机使用一个**CoaxPress**图像数据流。

关于图像数据的传送方法, 应考虑两种情况:

- 对于小对象, 对象图像数据被传送到**单个GenTL缓冲区**。
- 对于大对象, 对象图像数据被传递到**多个GenTL 缓冲区**中。

在这两种情况下, 图像数据都是通过单个PCIe DMA通道传输的, 并且通过Coaxlink 传输的延迟很低: 一条图像线。

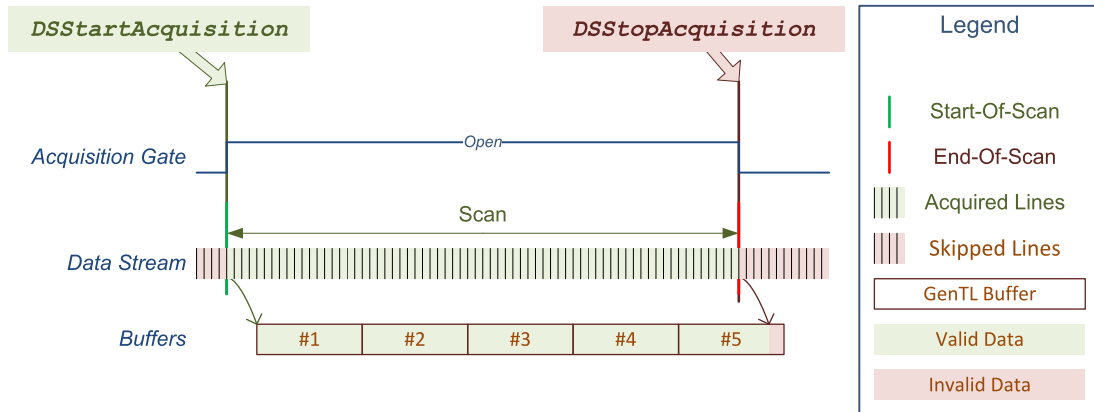
GenTL 缓冲区填充规则-行扫描相机

在行扫描成像中, 按照以下规则填充GenTL缓冲区:

- 在默认情况下, 扫描的第一个采集的行数据, 存储在新缓冲器的开头。通过将 **StripeArrangement** 设置为 **Geometry_1X_1YE**来启用垂直图像翻转时, 扫描第一次所采集的行数据, 存储在新缓冲区的最后一整行的位置。
- 缓冲区包含整数个图像行数据。
- 当缓冲区的剩余空间不足以存储图像行数据时, 采集继续进入新的缓冲区, 填充的缓冲区可供应用程序处理。
- 当获取扫描的最后一行数据时, 可能部分填充的最后一个缓冲区, 可供应用程序处理。

线扫描采集用例

连续物体扫描



连续物体扫描

这种情况适用于**连续物体**的图像扫描。

Coaxlink 采集控制器配置如下：

- **StartOfScanTriggerSource = Immediate.**
- **EndOfScanTriggerSource = StopScan.**

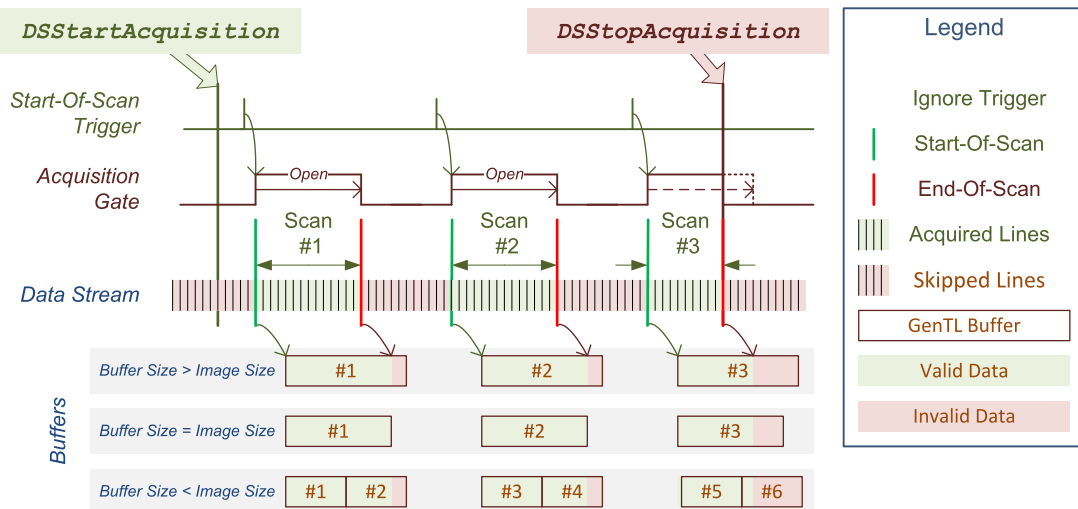
当调用 `DSStartAcquisition` 函数时，扫描从下一条线边界开始。

当应用程序调用 `DSStopAcquisition` 函数时，采集门关闭。

根据分配的缓冲区大小和扫描持续时间，对象图像适合单个缓冲区或需要多个缓冲区。

每个缓冲区一填满就被发送到应用程序。最后一个缓冲区(可能是部分填充的)将在最后一个图像数据写入后立即发送。

离散物体定长扫描



用公共扫描长度扫描离散物体

Coaxlink 采集控制器配置如下：

- **StartOfScanTriggerSource** = **StartScan** 或任何适用的I/O工具箱事件输出，例如：**LIN1**。
- **EndOfScanTriggerSource** = **ScanLength**。
- **ScanLength** 是表示完全捕获对象图像所需的行数的任何正数。

当调用DSStartAcquisition该功能时，采集控制器的“扫描开始”触发已启动。

然后，采集控制器等待有效“扫描开始”触发事件的首次出现。

可以生成有效的“扫描开始”事件：

- 通过使用 **StartScan** 命令的应用程序。
- 通过选定的硬件事件源(如果由 **StartOfScanTriggerSource** 指定)。

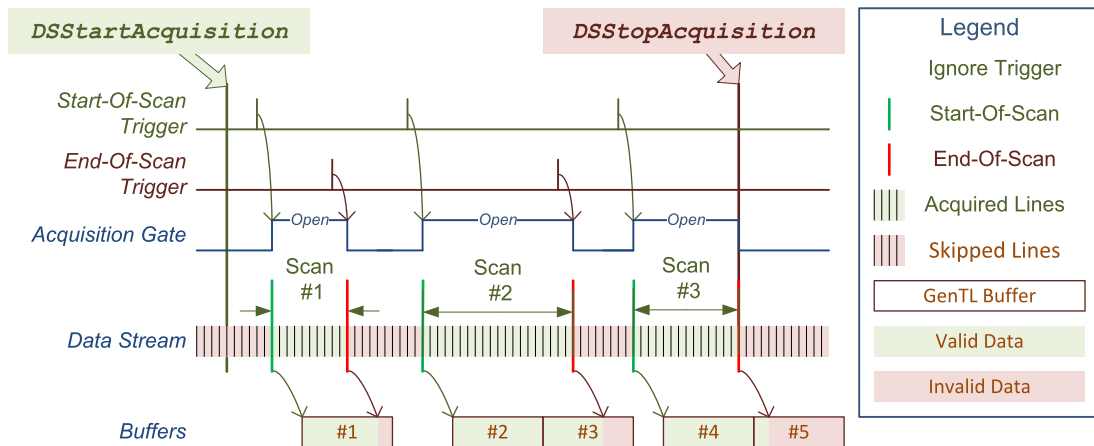
在扫描过程中，采集控制器忽略任何“扫描开始”触发事件。

在“扫描开始”事件后，采集门在第一行边界打开。

采集门会在采集到指定数量的行后自动关闭，或者在应用程序调用DSStopAcquisition函数时自动关闭。

根据分配的缓冲区大小，对象图像适合单个缓冲区或需要多个缓冲区。每个缓冲区一填满就被发送到应用程序。扫描结束时，部分填充的缓冲区立即交付。以下图像采集总是从新的缓冲区开始。

离散物体可变长度扫描



需要可变扫描长度的离散物体扫描

Coaxlink 采集控制器配置如下：

- **StartOfScanTriggerSource** = **StartScan** 或任何适用的I/O工具箱事件输出，例如：**LIN1**。
- **EndOfScanTriggerSource** = **StopScan** 或任何适用的I/O工具箱事件输出，例如：**LIN2**。

当调用DSStartAcquisition该功能时，采集控制器的“扫描开始”触发已启动。

然后，采集控制器等待有效“扫描开始”触发事件的首次出现。

可以生成有效的“扫描开始”事件：

- 通过使用**StartScan**命令的应用程序。
- 通过选定的硬件事件源(如果由**StartOfScanTriggerSource**指定)。

在扫描过程中，采集控制器忽略任何“扫描开始”触发事件。

在“扫描开始”事件后，采集门在第一行边界打开。

在有效的“扫描结束”事件之后，或者当应用程序调用DSStopAcquisition函数时，采集门在第一行边界处关闭。

可以生成有效的“扫描结束”事件：

- 通过使用**StopScan**命令的应用程序。
- 通过选定的硬件事件源(如果由**EndOfScanTriggerSource**指定)。

当没有进行扫描时，采集控制器忽略任何“扫描结束”触发事件。

根据分配的缓冲区大小，对象图像适合单个缓冲区或需要多个缓冲区。每个缓冲区一填满就被发送到应用程序。扫描结束时，部分填充的缓冲区立即交付。下一次图像采集总是从一个新的缓冲区开始。

11.10. C2C-链路

C2C链接描述	225
C2C链路规范	226
触发传播延迟	228
CIC同步	230
C2C链路设置步骤	233

C2C链路描述

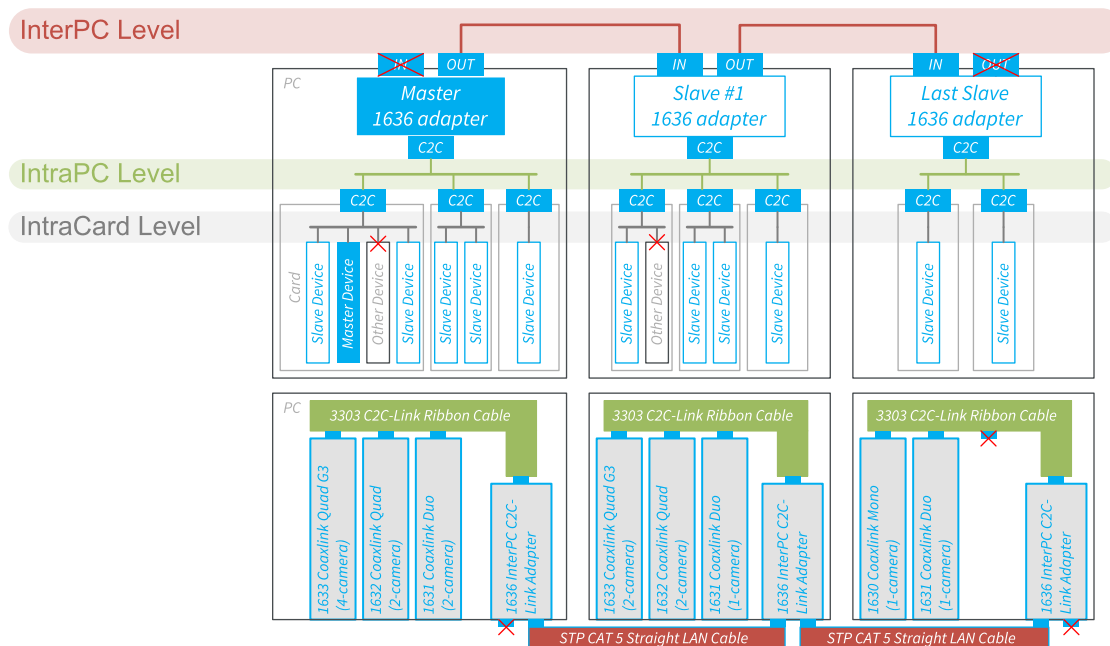
C2C链路是一种硬件通信介质，允许单个“C2C链路主设备”可靠地与多个“C2C链路从设备”共享1或2个触发事件。

C2C链路是可扩展的：它可以互连属于同一Coaxlink卡、或同一台PC中的不同卡，或不同PC中的不同卡的设备。

C2C链路反馈信道，允许任何C2C链路设备向C2C链路主设备指示其不能接受触发。

C2C链路互连最多可以组合三个互连级别：

- **IntraCard**级别使用FPGA内部资源，互连属于同一卡的2个或更多C2C链路设备。
- **IntraPC**级别将C2C链路设备互连在同一台PC的两个或更多个卡上。它需要一根附件电缆，比如**3303 C2C-链路带状电缆**，或为每台PC定制的C2C链路电缆。
- **InterPC**级别互连跨两台或多台PC的C2C链路设备。每台PC需要**1636 InterPC C2C-链路适配器**，每台适配器需要一根RJ45 CAT 5 STP直LAN电缆，但最后一根除外。



使用InterPC，IntraPC和IntraCard级别的C2C链路配置示例

C2C链路配置

C2C链路配置	互连级别
IntraCard	仅限IntraCard
IntraPC	IntraPC+IntraCard(可选)
InterPC	InterPC+IntraPC+IntraCard(可选)

C2C链路规范

定义

触发延迟

触发信号从主设备到从属设备的传播延迟。这种延迟由电子设备(FPGA、适配器...)和互连电缆内的传播延迟组成。

对于电缆，延迟与电缆长度成比例，通常为5 ns/m。

触发延迟偏斜

触发延迟值在属于C2C链路的所有设备上的分散。

触发延迟抖动

触发延迟的变化取决于外部因素，例如温度、信号噪声...

触发率

重复触发事件的发生率。倒数(1/触发率)是连续触发之间所需的最小时间间隔。

IntraCard C2C链路互连级别

参数	最小	类型	最大	单位
触发延迟抖动			5	ns
触发率	0		2.5	MHz

IntraPC C2C链路互连级别

参数	最小	类型	最大	单位
电缆连接器数量(包括1636适配器, 如果有)	2		4	-
电缆长度	-		0.6	m
触发延迟抖动	5		10	ns
触发率	0		2.5	MHz

InterPC C2C链路互连级别

以下规范针对需要**最高触发率**的应用：

参数	最小	类型	最大	单位
1636个适配器计数	2		10	-
适配器到适配器LAN电缆长度	0.3		100	m
累积的适配器到适配器LAN电缆长度	0.3		100	m
触发延迟抖动	10	20	40	ns
触发率	0		2.5	MHz

以下规范针对需要**最长可达距离**的应用：

参数	最小	类型	最大	单位
1636个适配器计数	2		32	-
适配器到适配器电缆长度	0.3		1200	m
累积的适配器到适配器电缆长度	0.3		1200	m
触发延迟抖动			1	μs
触发率	0		200	kHz



备注 假设长度 \times 频率的乘积是恒定的，那么最大触发率规范可以推断为100m和1200m之间的中间距离：在这种情况下为250 [m。MHz]。

触发传播延迟

触发信号从主设备到从属设备的传播延迟，可以通过将信号路径的每个段中遇到的典型延迟相加来粗略估计。

每个C2C链路段的典型延迟值

延迟元件	最小	类型	最大	单位
(1) IntraPC 互连(包括FPGA I/O和IntraPC 电缆在内的整个IntraPC C2C链路互连)	0	5	10	ns
(2) 1636 InterPC C2C-链路适配器 – IntraPC-to-InterPC 延迟		15		ns
(3) 1636 InterPC C2C-链路适配器 – InterPC-to-IntraPC 延迟		20		ns
(4) 1636 InterPC C2C-链路适配器 – InterPC-to-InterPC 延迟		0		ns
(5) InterPC LAN 电缆		5		ns/m

示例1-IntraPC 配置

对于仅IntraPC控制配置，只有一个延迟元件需要考虑：(1)

参数	最小	类型	最大	单位
(1) IntraPC 互连	0	5	10	ns
总触发延迟	0	5	10	ns

示例2-3 -适配器InterPC配置；20m + 20m LAN电缆

此配置由3个Intra-PC段组成。对于与主设备属于同一InPC段的设备，只有一个元件需要考虑。

参数	最小	类型	最大	单位
(1) IntraPC 互连	0	5	10	ns
Master InterPC段设备的总触发延迟	0	5	10	ns

对于与Slave1适配器同属于IntraPC段的设备，需要考虑5个延迟因素：

参数	最小	类型	最大	单位
(1) Master IntraPC 互连	0	5	10	ns
(2) Master 1636 InterPC C2C-链路适配器 – IntraPC-to-InterPC 延迟		15		ns
(5) Master to slave1 InterPC LAN 电缆-20m		100		ns

参数	最小	类型	最大	单位
(3) Slave1 1636 InterPC C2C-链路适配器 - InterPC-to-IntraPC 延迟		20		ns
(1) slave 1 IntraPC 互连	0	5	10	ns
slave1 interPC段设备的总触发延迟		145		ns

对于与Slave2适配器同属于IntraPC段的设备，需要考虑7个延迟因素：

参数	最小	类型	最大	单位
(1) Master IntraPC 互连	0	5	10	ns
(2) Master 1636 InterPC C2C-链路适配器 - IntraPC-to-InterPC 延迟		15		ns
(5) Master to slave1 InterPC LAN 电缆-20m		100		ns
(4) Slave1 1636 InterPC C2C-链路适配器 - InterPC-to-InterPC 延迟		0		ns
(5) Slave1 to slave2 InterPC LAN 电缆- 20 m		100		ns
(3) Slave2 1636 InterPC C2C-链路适配器 - InterPC-to-IntraPC 延迟		20		ns
(1) Slave2 IntraPC 互连	0	5	10	ns
slave 2 interPC段设备的总触发延迟		245		ns

CIC同步



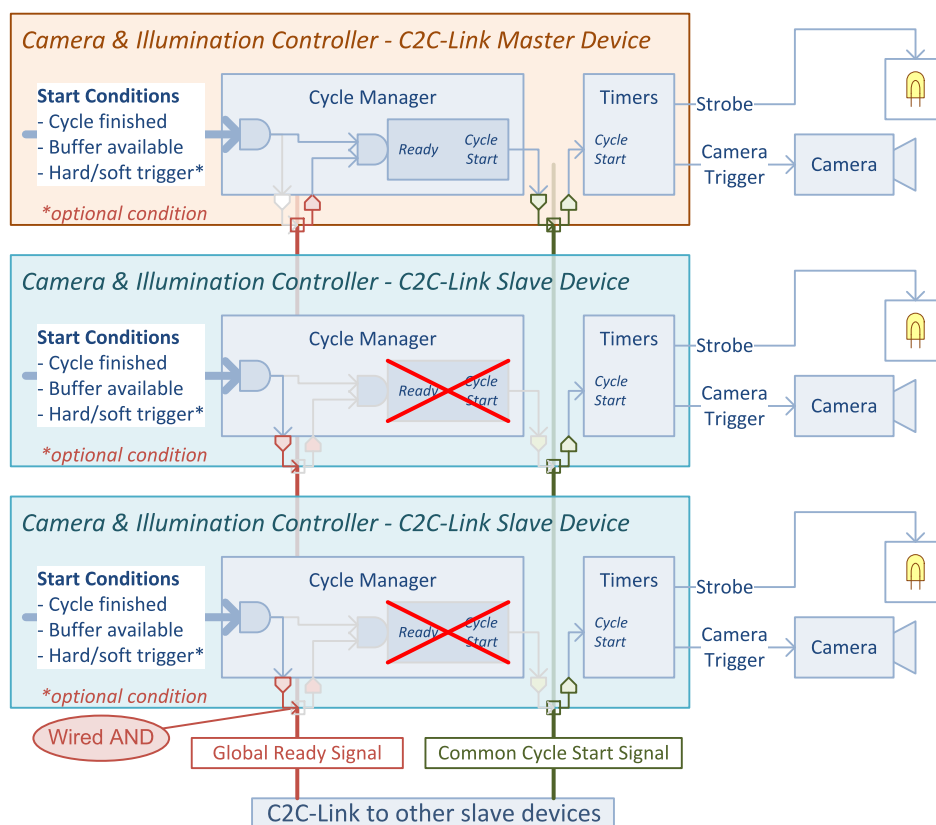
备注 此功能仅适用于区域扫描固件变体。

原则

“C2C链路主设备”的相机和照明控制器的循环触发器事件被分配给一个或多个“C2C链路从属设备”。所有设备的CIC循环同时开始。然而，循环定时参数可以单独配置。

同步是通过同步所有相关CIC的**CIC循环触发**而实现的(使用**C2C-链路互连**)。

C2C链路互连两个或多个相机照明控制器。一个CIC配置为**C2C-Link Master**，其他则配置为**C2C-Link Slave(s)**。



配置一个C2C链路Master(顶部)和两个C2C链路Slave:

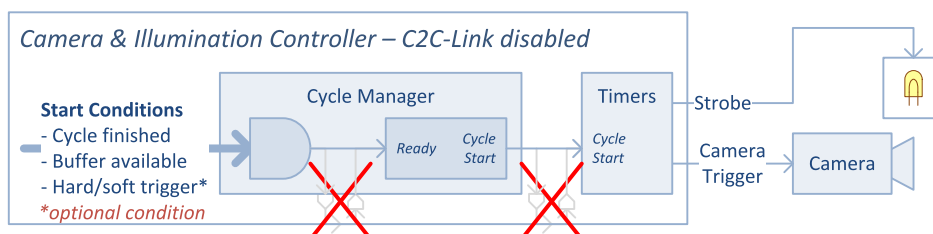
C2C-链路 Master:

- 当所有同步CIC(包括其自身)的所有启动条件都满足时，生成公共**Cycle Start**事件。
- 向所有同步CIC发送一个公共**Cycle Start**信号。
- 当接收到公共**Cycle Start**时，根据定时器设置，生成一个频闪脉冲和一个**Cycle Start**脉冲。

C2C-链路Slave:

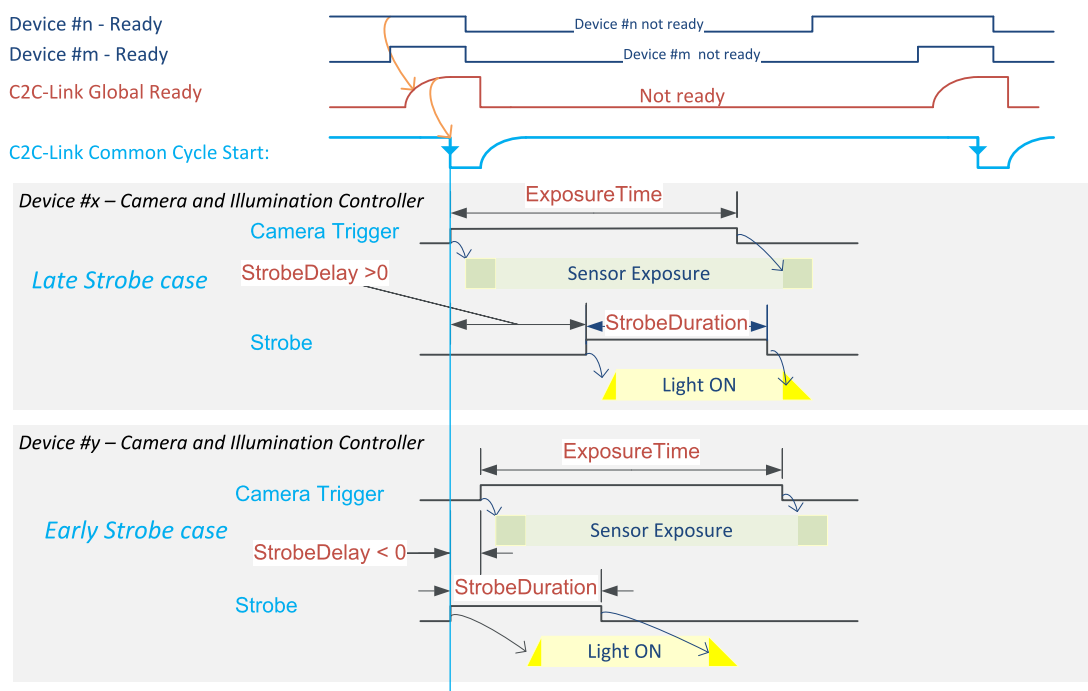
- 强制全局就绪信号为“假”，直到其启动条件全部满足
- 当接收到公共**Cycle Start**时，根据定时器设置，生成一个频闪脉冲和一个**Cycle Start**脉冲。

对于不需要同步的设备，C2C链路保持禁用状态。



禁用C2C链路的CIC设备

时序图



通过C2C链路时序图的CIC同步

上图显示了两个连续公共**Cycle Start**事件的时序图：

C2C链路全局就绪信号保持在低位，直到所有参与设备的就绪为真，防止C2C链路Master发出启动事件。当所有参与设备释放时，它会迅速上升，其上升时间最长为100 ns。

一旦确认C2C链路全局就绪信号为高，主设备就会断言在公共循环启动信号上有一个突然变低的转变；该边缘被传播到所有参与设备计时器的所有启动输入。

一旦循环开始，只要启动下一个循环的所有当地条件都不满足，每个CIC就会强制就绪信号为低。

每个设备的计时器发出一个**Camera Trigger**和一个频闪，以及它们各自的延迟和持续时间设置。通常，所有参与设备的设置都相同；但如果需要，允许应用程序应用不同的设置。

C2C链路允许的最短的**Cycle Start**周期是400 ns；允许理论频率限制为2.5 MHz。

C2C链路设置步骤

硬件设置

该步骤针对每个C2C链路配置：

IntraCard C2C链路配置

此配置专门使用FPGA内部资源来构建C2C链路互连；它不需要任何额外的硬件！

IntraPC C2C链路配置

此配置需要一根附件电缆，比如**3303 C2C-链路带状电缆**，或为每台PC定制的C2C链路电缆。

将C2C链路电缆的C2C链路母接头，插入每个参与Coaxlink卡的C2C链路引脚接头。

InterPC C2C链路配置

每台PC需要**1636 InterPC C2C-链路适配器**，每台适配器需要一根RJ45 CAT 5 STP直LAN电缆，但最后一根除外。

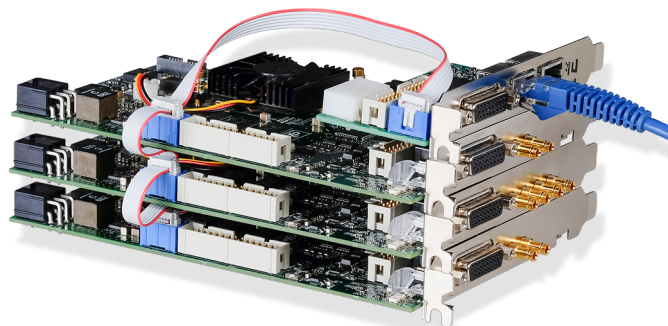
在每台参与电脑中：

1. 将**1636 InterPC C2C-链路适配器** 安装到一个自由槽中并固定支架。
2. 将适配器连接到电源。

另见： "Adapter Powering" topic in the 1636 section of the hardware manual.

3. 使用一个 **3303 C2C-链路带状电缆**，将所有参与卡的C2C链路连接器与适配器卡的C2C链路绑定在一起。
4. 使用LAN电缆，互连适配器。

另见： 请参阅硬件手册1636章节中的“InterPC互连”部分。



InterPC配置的IntraPC段

GenAPI 设置

分配一个值到 **C2CLinkConfiguration** GenTL设备功能：

- 分配值 **Master** 到C2C链路主设备的 **C2CLinkConfiguration** ，
- 分配值 **Slave** 到C2C链路从设备的 **C2CLinkConfiguration** ，

11.11. OEM 安全钥

介绍OEM安全钥

OEM安全钥功能允许应用程序：

- 在Coaxlink 卡的非易失性存储器中，编程一个“OEM安全钥”。
- 检索刚刚编程的OEM安全钥的加密版本。
- 对照编程的OEM安全钥或其加密版本，检查密钥。

OEM安全钥

OEM安全钥是应用程序定义的字符串。允许使用除空字符以外的任何字符。字符串长度不受限制。

密钥编程

当应用程序使用OEM安全钥值设置 **ProgramOemSafetyKey** GenICam功能时，Coaxlink 驱动程序将计算OEM 安全密钥的加密版本，并将其存储在Coaxlink 卡的非易失性内存中。

通过获取 **EncryptedOemSafetyKey** 的值(在设置 **ProgramOemSafetyKey** 之后)，可以检索加密的值。



警告 只允许已设置 **ProgramOemSafetyKey** 的同一应用程序进程检索加密值。只有在设置任何其他GenICam 功能之前，才允许这样做。

密钥检查

应用程序必须选择一个 **OemSafetyKeyVerification** 值

为了验证CoaxLink卡的OEM安全钥，应用程序设置一个“challenge”值到 **CheckOemSafetyKey[selector]** 功能。

当[selector]参数设置为 **EncryptedKey** 时，仅当challenge字符串与加密的OEM安全钥字符串相同时，set操作才会正常终止。

当[selector]参数设置为 **ProgrammingKey** 时，仅当challenge字符串与编程的OEM安全钥字符串相同时，set操作才会正常终止。

当[selector]参数设置为 **ProgrammingKeyOrEncryptedKey** 或省略时，仅当challenge字符串与原始OEM安全钥字符串或加密的OEM安全钥字符串相同时，set操作才会正常终止。

Euresys建议使用 **EncryptedKey** 选择器。这提高了安全级别，因为编程钥不需要出现在最终用户应用程序的任何地方。只有加密密钥，最终用户无法检索原始编程密钥。

使用 OEMSafetyKey

编程步骤-选项 A

使用 GenICam 浏览器：

- 转到接口模块的 GenApi 选项卡。
- 写一个密匙给 ProgramOemSafetyKey
- 复制 EncryptedOemSafetyKey 的值，并将其粘贴到适当的位置。



备注 编程密钥和加密密钥之间有一个直接的关系。一个给定的编程密钥总是会导致相同的加密密钥，即使在不同的计算机上或使用不同的 Coaxlink 卡。这使得可以一次性读取加密密钥，并在必须受到 OEM 安全密钥保护的应用程序中硬编码该值。

编程步骤-选项 B

使用自定义应用程序：

1. 通过将密钥写入 ProgramOemSafetyKey，对 CoaxLink 卡的 OEM 安全密钥进行编程。
2. 通过读取 EncryptedOemSafetyKey，来读回加密密钥。在适当的地方写下这个值。

```
grabber.set<InterfacePort>("ProgramOemSafetyKey", "plain-text key"); // 1
std::string encryptedKey=grabber.get<InterfacePort>("EncryptedOemSafetyKey"); // 2
```

验证步骤

在必须由 OEM 密钥保护的应用程序中：

```
InterfacePort>("CheckOemSafetyKey[EncryptedKey]", "encrypted key retrieved in the programming step");
```



备注 即使发现了加密密钥，并且攻击者使用它对卡进行重新编程，上述验证也将失败。