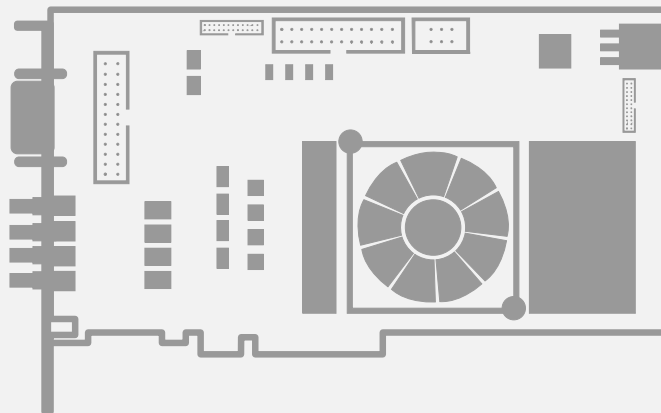


Coaxlink

Coaxlink 12.4 Functional Guide

1629 Coaxlink Duo PCIe/104-EMB
1630 Coaxlink Mono
1631 Coaxlink Duo
1632 Coaxlink Quad
1633 Coaxlink Quad G3
1633-LH Coaxlink Quad G3 LH
1634 Coaxlink Duo PCIe/104-MIL

1635 Coaxlink Quad G3 DF
1637 Coaxlink Quad 3D-LLE
3602 Coaxlink Octo
3603 Coaxlink Quad CXP-12
3620 Coaxlink Quad CXP-12 JPEG
3621-LH Coaxlink Mono CXP-12 LH
3622 Coaxlink Duo CXP-12



사용 약관

EURESYS s.a. 는 EURESYS s.a.의 하드웨어 및 소프트웨어의 부속 문서, 상표의 모든 재산권, 소유권, 이권을 보유합니다.

이 설명서에 언급된 회사 및 제품의 모든 이름은 해당 소유자의 상표일 수 있습니다.

이 문서에 포함된 EURESYS s.a.의 자료, 하드웨어 또는 소프트웨어, 브랜드를 사전 통지 없이 라이선싱, 사용, 임대, 임차, 번역, 재현, 복사 또는 수정하는 행위는 허용되지 않습니다.

EURESYS s.a. 는 언제든지 자사 재량에 따라 사전 통지 없이 제품 사양을 수정하거나 이 문서에서 제공하는 정보를 변경할 수 있습니다.

EURESYS s.a. 는 EURESYS s.a.의 하드웨어 또는 소프트웨어 사용과 관련하여 발생하는 일체의 매출, 수익, 영업권, 데이터, 정보 시스템의 손실 또는 피해 또는 기타 특별하거나, 우발적이거나, 간접적이거나, 필연적인 또는 징벌적인 손해에 대해 책임을 지지 않으며, 이는 본 문서의 누락 또는 오류로 인한 결과일 경우에도 마찬가지입니다.

이 문서는 Coaxlink 12.4.0 의 부속 자료입니다(문서 빌드 2106).

www.euresys.com

목차

1. 이 문서 정보	7
1.1. 문서 범위	7
1.2. 문서 변경	7
2. Coaxlink 아키텍처	8
2.1. 주요 요소	9
2.2. 블록 다이어그램	11
3. CoaXPress 호스트 인터페이스	15
3.1. 호스트 인터페이스 사양	16
3.2. Firmware Variants per Product	20
3.3. 호스트 연결 맵	26
3.4. 링크 설정	33
3.5. CoaXPress를 증가하는 힘	34
3.6. CoaXPress I/O 채널	38
3.7. CoaXPress 호스트-장치 트리거	39
3.8. CoaXPress LED 램프	49
3.9. 연결 테스트	51
3.10. CoaXPress 2.0 오류 카운터	52
3.11. CoaXPress 링크 유효성 검사 도구	54
4. 이미지 데이터 경로	60
4.1. FIFO 버퍼	61
4.2. 수집 게이트	62
4.3. 픽셀 데이터 처리	63
4.4. 픽셀 데이터 처리 구성	65
4.5. 픽셀 구성 요소 언패킹	68
4.6. 픽셀 구성 요소 순서 재정렬	70
4.7. 엔디안 변환	71
4.8. 픽셀 순서	72
4.9. 이미지 데이터 전송	73
4.10. 데이터 전송률 테스트 프로그램	80

5. 카메라 조절 원리	83
5.1. 카메라 주기	84
5.2. 카메라 주기 연결 규칙	85
5.3. 카메라 조절 방법	87
6. 조명 제어 원리	89
6.1. 조명 장치	90
6.2. 카메라와 조명주기 맞추기	91
7. 카메라 및 조명 컨트롤러	92
7.1. CIC 블록 다이어그램	93
7.2. 주기 타이밍 머신	94
7.3. 주기 관리자	96
7.4. 주기 트리거 관리자	97
7.5. 시퀀스 관리자	99
7.6. CIC 출력 신호 라우팅	101
7.7. CIC 타이밍 다이어그램	102
주기 타이밍 다이어그램	103
CIC주기 연속 중첩	105
주기 타이밍 다이어그램	110
8. 범용 I/O	112
8.1. I/O 라인 개요	113
8.2. I/O 라인 사용법	116
8.3. I/O 제어 블록	117
8.4. 라인 포맷 및 라인 모드 컨트롤	119
8.5. 라인 극성 제어	122
8.6. 필터 조절	123
8.7. 라인 소스 선택	124
8.8. 논리적 I/O 라인 상태	125
8.9. 물리적 I/O 라인 상태	126
8.10. 라인 드라이버 물리적 출력 상태	127
8.11. 초기 상태	128
9. I/O 도구 상자	129
9.1. I/O 도구 상자 소개	130
9.2. I/O Toolbox 구성	133
9.3. 라인 입력 도구	136

9.4. 쿼드러처 디코더 도구	137
9.5. Divider 도구	140
9.6. Multiplier/Divider 도구	141
9.7. 지연 도구	144
9.8. 사용자 작업 도구 및 사용자 출력 레지스터	146
9.9. 이벤트 입력 도구	150
9.10. C2C 링크 동기화 도구	151
10. 이벤트 신호 및 카운팅	153
10.1. 소개	154
10.2. 맞춤 이벤트 소스	158
10.3. 이벤트 특정 컨텍스트 데이터	162
10.4. GenTL 신호 정보	164
11. 고급 기능	165
11.1. 하위 링크 수집	166
11.2. 멀티 스트림 수집	168
11.3. CoaXPress 데이터 전달	169
데이터 전달 원칙	170
데이터 전달 연결 체계	172
라인 스캔 트리거 동기화	173
설정 스크립트 예제	174
11.4. 플랫폼 필드 보정	175
플랫폼 필드 보정란 무엇입니까?	176
Coaxlink FFC	179
FFC 마법사 샘플 프로그램	182
11.5. 조회 테이블 처리	189
LUT 처리 소개	190
단색 조회 테이블 처리	191
LUT 콘텐츠 정의	192
LUT 설정 절차	199
11.6. 바이엘 CFA 디코딩	202
바이엘 CFA에서 RGB 변환	203
Bayer CFA 디코더 사용	207
11.7. JPEG 인코딩	210
11.8. 레이저 선 추출	212
소개	213
레이저 라인 추출 알고리즘	214
코어 구현을 처리하는 LLE	215
코어 특징을 처리하는 LLE	216
이중 레이저 라인 추출	218
선형 필터	219
코어링 임계 값	220

최대 감지	221
피크 감지	222
중력의 중심	223
사용 사례의 예	224
오브젝티브	225
입력 ROI 정의	226
필터링 및 임계값 설정	227
LLE 알고리즘 정의	228
스캔 길이 및 버퍼 크기 정의	230
부록	231
11.9. 라인 스캔 수집	232
라인 스캔 수집 원칙	233
라인 스캔 수집 사례	237
11.10. C2C- 링크	240
C2C 링크 상호 연결	241
C2C-링크 전기 사양	243
트리거 전파 지연	245
주기 트리거 동기화	247
C2C-링크 설정 순서	250
11.11. OEM 안전 키	252
OEM 안전 키 소개	252
OEMSafetyKey 사용	253

1. 이 문서 정보

1.1. 문서 범위	7
1.2. 문서 변경	7

1.1. 문서 범위

이 문서는 Coaxlink 제품이 Coaxlink 드라이버 버전 12.4.0로 작동할 때 Coaxlink 제품의 기능을 사용하는 방법을 설명합니다.

명시되지 않은 한 이 문서에서 설명하는 기능은 Coaxlink 드라이버에서 지원하는 모든 Coaxlink 제품 및 해당 펌웨어 변형에 적용됩니다.

1.2. 문서 변경

Coaxlink 12.4

다음 주제가 추가되었습니다:

- "CoaXPress 2.0 오류 카운터" 페이지52
- "데이터 전송률 테스트 프로그램" 페이지80

다음 주제가 개정되었습니다:

- "CoaXPress를 능가하는 힘" 페이지34
- "I/O 라인 개요" 페이지113
- "I/O Toolbox 구성" 페이지133

2. Coaxlink 아키텍처

2.1. 주요 요소	9
2.2. 블록 다이어그램	11

2.1. 주요 요소

Coaxlink 기반 이미지 수집 시스템의 주요 기능 요소에 대한 간략한 개요.

GenTL 계층 구조

Coaxlink의 각 기능 요소는 GenTL 모듈에 속한 GenICam 기능에 의해 구성되고 제어됩니다.

계층 구조의 상단에는 호스트 PC 당 하나의 *GenTL 시스템 모듈*이 있습니다. 호스트 PC의 모든 *GenTL 인터페이스 모듈*을 바인딩합니다.

각 Coaxlink 카드에 대해 하나의 *GenTL 인터페이스 모듈*이 있습니다. Coaxlink 카드의 모든 *GenTL 장치 모듈*을 바인딩합니다.

Coaxlink 카드에 부착 된 각 카메라 (또는 이미징 장치)에 대해 하나의 *GenTL 장치 모듈*이 있습니다. 이미징 장치 (카메라) 자체에 속한 요소는 *원격 장치*라고 합니다. 반대로, 프레임 그래버에 속하는 요소는 *로컬 장치*라고도 합니다.



참고 Coaxlink 카드에 연결할 수 있는 최대 카메라 수는 설치된 펌웨어 변형에 따라 결정됩니다.

Coaxlink 카드에 연결된 카메라로 전달되는 각 데이터 스트림에 대해 하나의 *GenTL 데이터 스트림 모듈*이 있습니다. 이미지 빌드 업에 관련된 요소를 수집하고 이미징 장치에서 GenTL 버퍼 풀로 전송합니다.



참고 Coaxlink 카드에 연결된 카메라의 최대 데이터 스트림 수는 설치된 펌웨어 변형에 따라 결정됩니다.

각 이미지 버퍼마다 하나의 *GenTL 버퍼 모듈*이 있습니다.

인터페이스 모듈 주요 구성 요소 (주황색)

I/O 라인

블록 "[범용 I/O](#)" [페이지112](#)는 카드의 모든 I/O 포트를 수집합니다.

I/O 도구 상자

"[I/O 도구 상자](#)" [페이지129](#) 블록은 I/O 포트 입력에 연결된 트리거 및 인코더 장치에서 이벤트 스트림을 작성하는 데 사용되는 도구 모음을 수집합니다.



참고 이러한 요소는 Coaxlink 카드가 관리하는 모든 GenTL 장치 모듈에 공통적으로 사용되거나 (또는 공유될 수 있습니다)

장치 모듈 기본 요소 (녹색)

카메라 및 조명 컨트롤러

이 블록은 카메라주기 및 조명 스트로브를 제어하는 데 사용됩니다. I/O 도구 상자 출력 스트림에서 실시간 (카메라) 사이클 트리거 이벤트를 수신하도록 구성할 수 있습니다. 2개의 실시간 신호를 생성합니다: 카메라 트리거 신호, 카메라 트리거 입력으로 전송된 스트로브 신호 및 카메라와 관련된 조명 장치로 전송된 스트로브 신호.



참고이 요소는 해당 이미징 장치와 관련된 모든 GenTL 데이터 스트림 모듈과 공통입니다(또는 공유할 수 있습니다).

데이터 스트림 모듈 주요 요소

이미지 캡처 컨트롤러

이 블록은 수집 게이트를 제어하는 데 사용됩니다. I/O 도구 상자 출력 스트림에서 실시간 시작 및 트리거 종료 이벤트를 수신하도록 구성할 수 있습니다.

캡처 게이트

"수집 게이트" 페이지62는 데이터 추출을 제어하고 수집할 필요가 없는 이미지 데이터를 필터링합니다.

FIFO 버퍼

온보드 "FIFO 버퍼" 페이지61는 이미지 크기, 픽셀 유형, 시간 스탬프 등과 같은 관련 메타 데이터와 함께 원시 이미지 데이터를 임시로 저장합니다.

픽셀 처리

"픽셀 데이터 처리" 페이지63는 직접 픽셀 처리를 수행합니다.

데이터 포매팅

"픽셀 구성 요소 언패킹" 페이지68, "픽셀 구성 요소 순서 재정렬" 페이지70, "엔디안 변환" 페이지71 및 "픽셀 순서" 페이지72 작업은 원하는 픽셀 출력 형식에 따라 구성됩니다.

이미지 데이터 전송

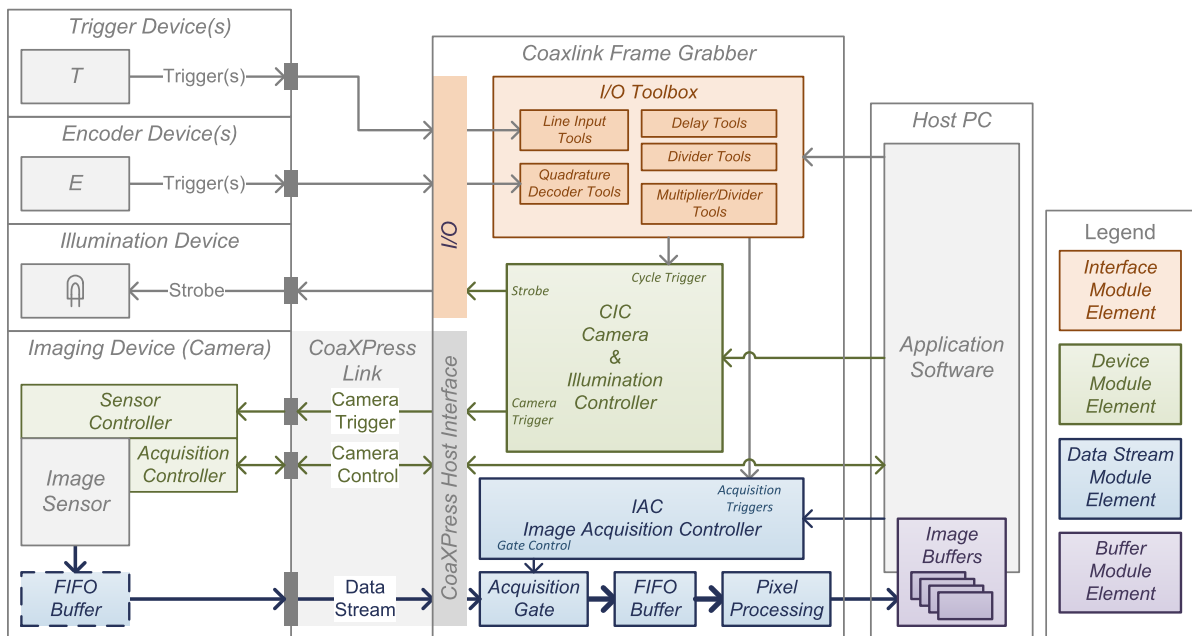
"이미지 데이터 전송" 페이지73는 이미지 데이터를 대상 버퍼로 전송합니다.

2.2. 블록 다이어그램

이 섹션에서는 하나의 Coaxlink 카드를 사용하는 세 개의 이미지 수집 시스템의 블록 다이어그램을 보여줍니다

참고 이후 다이어그램에서 주요 요소는 직사각형으로 표시되며 그 관계는 신호 방향 또는 데이터 흐름을 나타내는 화살표가 있는 선분으로 표시됩니다. 직사각형 채우기 색은 범례에서 설명한 대로 GenTL 계층 구조의 수준을 나타냅니다.

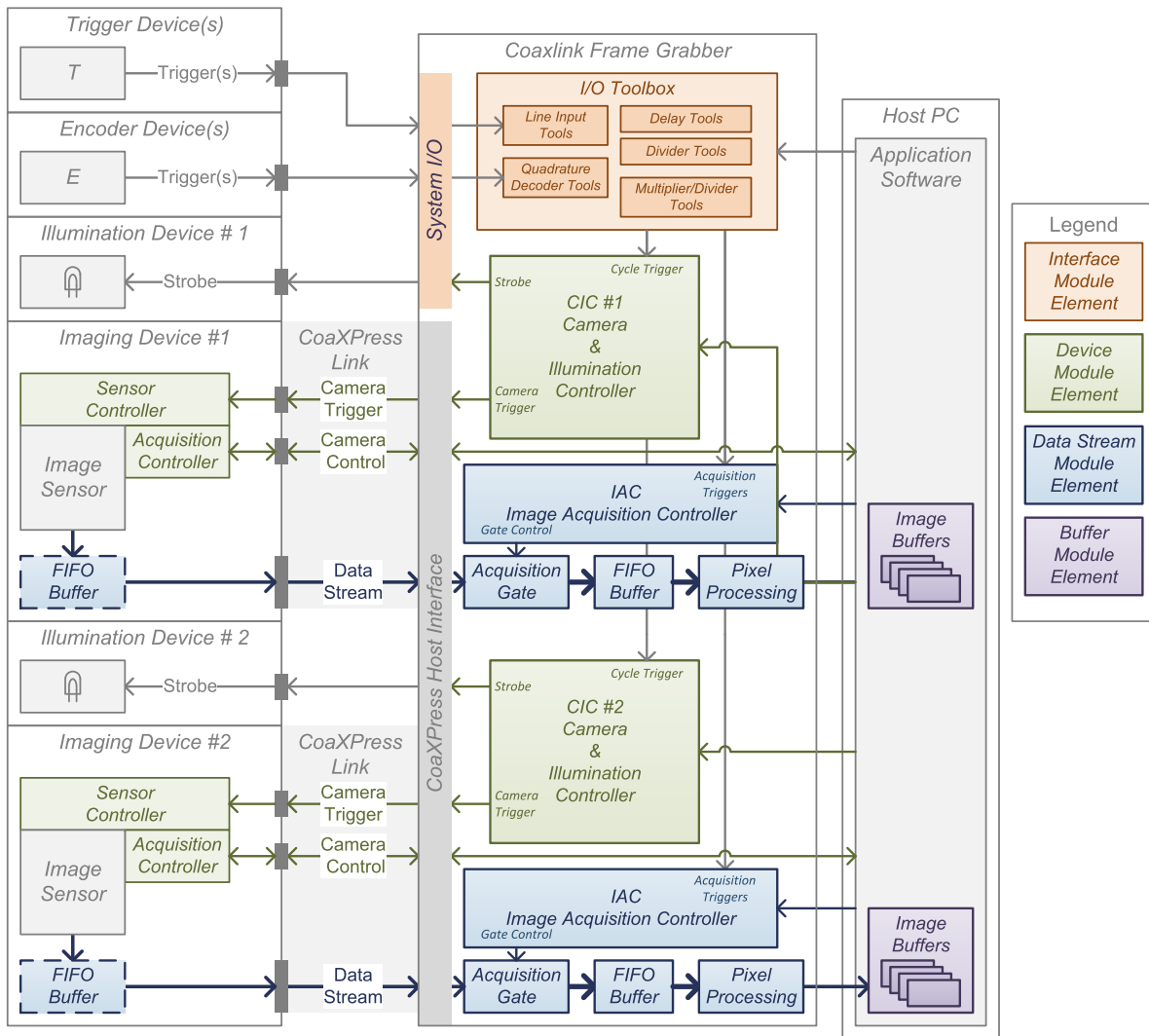
1-카메라, 1-데이터 스트림



1-카메라, 1-데이터 스트림 Coaxlink 이미지 캡처 시스템

참고 이 설정은 1-카메라 또는 1-카메라, 라인 스캔펌웨어 변형이 설치된 경우에만 적용됩니다.

2-카메라, 1-데이터 스트림

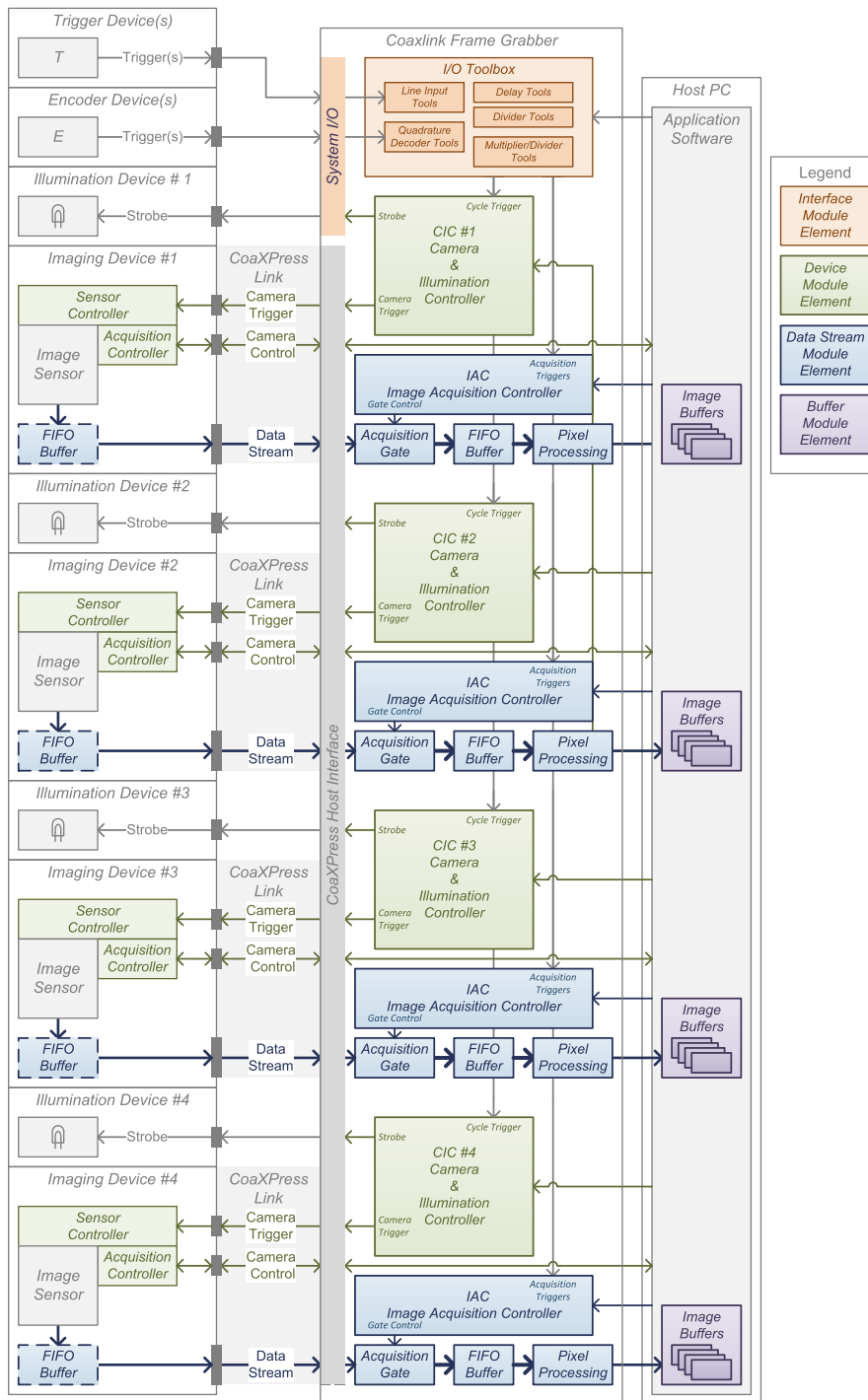


2-카메라, 1-데이터 스트림 Coaxlink 이미지 캡처 시스템



참고이 설정은 2-카메라 또는 2-카메라, 라인 스캔펌웨어 변형이 설치된 경우에만 적용됩니다.

4-카메라, 1-데이터 스트림

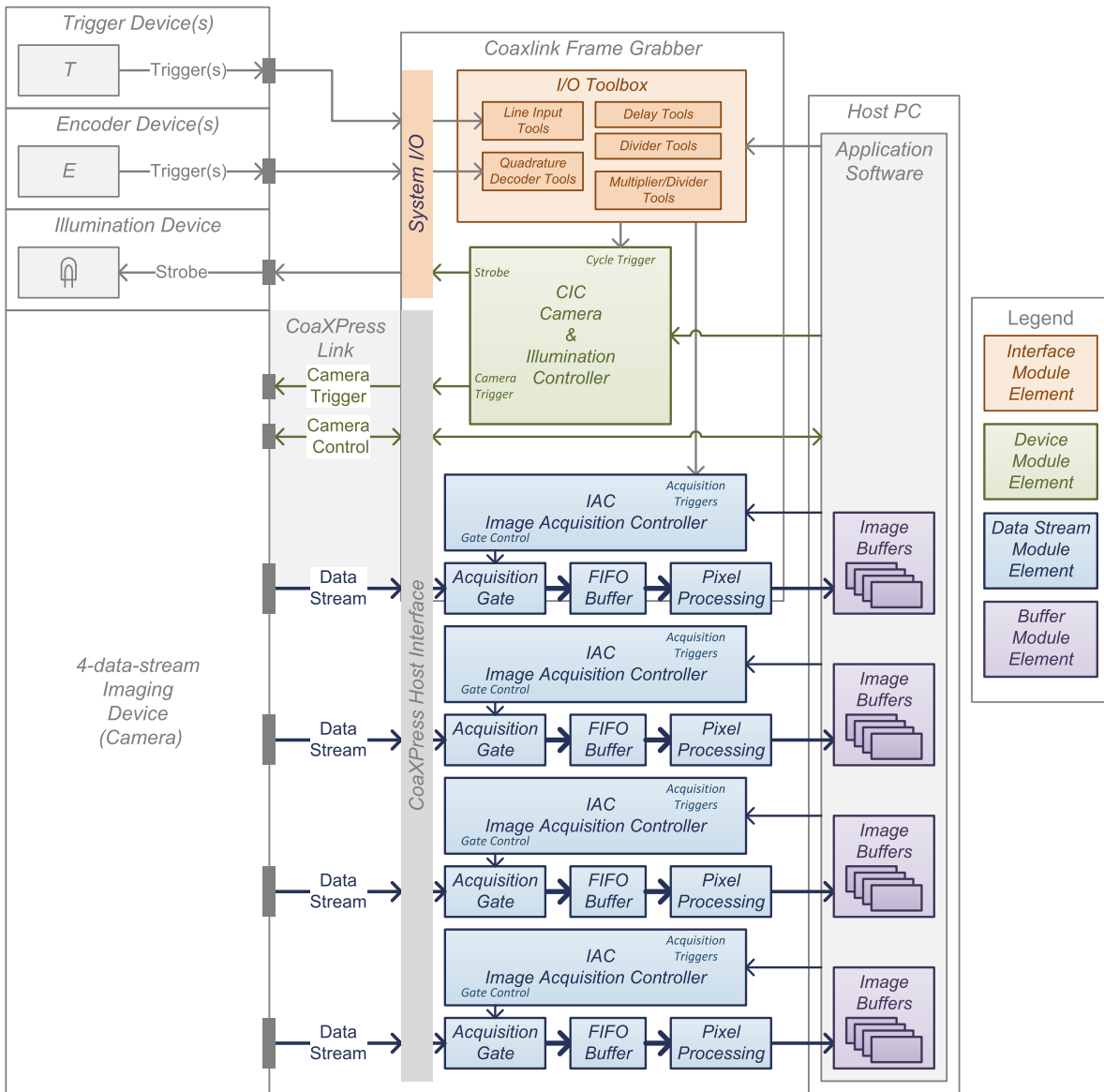


4-카메라, 1-데이터 스트림 Coaxlink 이미지 캡처 시스템



참고이 설정은 4-카메라 또는 4-카메라, 라인 스캔펌웨어 변형이 설치된 경우에만 적용됩니다.

1-카메라, 4-데이터 스트림



1-카메라, 4-데이터 스트림 Coaxlink 이미지 캡처 시스템

참고이 설정은 1-카메라, 4-데이터 스트림 펌웨어 변형이 설치된 경우에만 적용됩니다.

3. CoaXPress 호스트 인터페이스

3.1. 호스트 인터페이스 사양	16
3.2. Firmware Variants per Product	20
3.3. 호스트 연결 맵	26
3.4. 링크 설정	33
3.5. CoaXPress를 능가하는 힘	34
3.6. CoAXPress I/O 채널	38
3.7. CoaXPress 호스트-장치 트리거	39
3.8. CoaXPress LED 램프	49
3.9. 연결 테스트	51
3.10. CoaXPress 2.0 오류 카운터	52
3.11. CoaXPress 링크 유효성 검사 도구	54

3.1. 호스트 인터페이스 사양

인터페이스 (카드) 사양

하나의 GenTL 인터페이스 모듈의 범위를 갖는 사양 (즉, Coaxlink 카드)

CoaXPress 기능 모음	해당 제품
CXP-6 DIN 1	Mono
CXP-6 DIN 2	Duo Duo104EMB Duo104MIL
CXP-6 DIN 4	Quad QuadG3 QuadG3LH QuadG3DF Quad3DLLE
CXP-6 DIN 8	Octo
CXP-12 µBNC 1	MonoCXP12LH
CXP-12 µBNC 2	DuoCXP12
CXP-12 µBNC 4	QuadCXP12 QuadCXP12J

최대 통합 입력 데이터 속도	해당 제품
625,000 바이트/초	Mono
1,250,000 바이트/초	Duo Duo104EMB Duo104MIL MonoCXP12LH
2,500,000 바이트/초	Quad QuadG3 QuadG3LH QuadG3DF Quad3DLLE DuoCXP12
5,000,000 바이트/초	Octo QuadCXP12 QuadCXP12J

기기 수	해당 제품
1 또는 2 또는 4 또는 5 또는 8	참조: "Firmware Variants per Product" 페이지20

참조: 전기 사양에 대한 하드웨어 설명서의 CoaXPress Host Interface.

장치 (카메라) 사양

하나의 GenTL 장치 모듈 (즉, Coaxlink 카드에 부착 된 카메라)

기기 종류	적용 가능한 펌웨어 변형
영역 스캔 카메라, 1 데이터 스트림 또는 영역 스캔 카메라, 최대 4 데이터 스트림 또는 라인 스캔 카메라, 1 데이터 스트림	참조: "Firmware Variants per Product" 페이지20

위의 표에는 "CoaXPress 데이터 전달" 페이지169 및 "하위 링크 수집" 페이지166설정에서 사용되는 특수 카메라 유형이 제외되어 있습니다.

연결 수 (카메라 당)	적용 가능한 제품/ 펌웨어 변형
1 또는 2 또는 4 또는 8	참조: "Firmware Variants per Product" 페이지20
8	참조: "하위 링크 수집" 페이지166

데이터 스트림 사양

하나의 GenTL 데이터 스트림 모듈의 범위를 갖는 사양

최대 스트림 패킷 크기	해당 제품
16,384 바이트	모든 제품
이미지 스트림 형식	적용 가능한 펌웨어 변형
직사각형 이미지 (YSize > 0)	영역 스캔 "Firmware Variants per Product" 페이지 20
직사각형 이미지 (YSize = 0)	라인 스캔 "Firmware Variants per Product" 페이지 20
임의의 이미지	지원되지 않음
이미지 스캐닝 방법	적용 가능한 펌웨어 변형
프로그레시브	모든 펌웨어 변형
인터레이스	지원되지 않음
탭 형상	적용 가능한 펌웨어 변형
1X_1Y	모든 펌웨어 변형
1X_1Y2, 1X_2YE, 1X_2YM	모든 영역 스캔 펌웨어 변형
픽셀 형식	적용 가능한 펌웨어 변형
Raw	모든 펌웨어 변형
Mono8, Mono10, Mono12, Mono14, Mono16	모든 펌웨어 변형
BayerGR8, BayerRG8, BayerGB8, BayerBG8 BayerGR10, BayerRG10, BayerGB10, BayerBG10 BayerGR12, BayerRG12, BayerGB12, BayerBG12 BayerGR14, BayerRG14, BayerGB14, BayerBG14 BayerGR16, BayerRG16, BayerGB16, BayerBG16	모든 펌웨어 변형
RGB8, RGB10, RGB12, RGB14, RGB16	모든 펌웨어 변형
RGBA8, RGBA10, RGBA12, RGBA14, RGBA16	모든 펌웨어 변형

너비 증가 단계



경고카메라의 이미지 너비 설정은 너비 증가 단계 값의 배수여야 합니다:

Applies to:

Mono	Duo	Quad	Duo104EMB	Duo104MIL	Quad3DLE
------	-----	------	-----------	-----------	----------

비트 너비	너비 증가 단계 [픽셀]
8-비트	8
10-, 12-, 14- 및 16-비트	4

Applies to:

QuadG3	QuadG3LH	QuadG3DF
--------	----------	----------

비트 너비	너비 증가 단계 [픽셀]	
	바이엘 CFA 디코더 비활성화	바이엘 CFA 디코더 활성화
8-비트	4	16
10-, 12-, 14- 및 16-비트	2	8

Applies to:

Octo	QuadCXP12	MonoCXP12LH	DuoCXP12
------	-----------	-------------	----------

비트 너비	너비 증가 단계 [픽셀]	
	바이엘 CFA 디코더 비활성화	바이엘 CFA 디코더 활성화
8-비트	4	32
10-, 12-, 14- 및 16-비트	2	16

Applies to:

QuadCXP12J

비트 너비	너비 증가 단계 [픽셀]
8-비트	32

3.2. Firmware Variants per Product

1630 Coaxlink Mono

펌웨어 변형	설명	호스트 연결 맵	고급 프로세싱
1-카메라	1-연결 영역 스캔 카메라 1개	1D1	LUT

1631 Coaxlink Duo

펌웨어 변형	설명	호스트 연결 맵	고급 프로세싱
1-카메라	1- 또는 2-연결 영역 스캔 카메라 1개	1D2	LUT
1-카메라, 라인 스캔	1 또는 2-연결 영역 스캔 카메라 1개	1D2	LUT
2-카메라	1-연결 영역 스캔 카메라 1개 또는 2개	2D11	LUT
2-카메라, 라인 스캔	1-연결 영역 스캔 카메라 1개 또는 2개	2D11	LUT

1632 Coaxlink Quad

펌웨어 변형	설명	호스트 연결 맵	고급 프로세싱
1-카메라	1- 또는 2- 또는 4-연결 영역 스캔 카메라 1개	1D4	LUT
1-카메라, 라인 스캔	1 또는 2- 또는 4-연결 영역 스캔 카메라 1개	1D4	LUT
2-카메라	1- 또는 2-연결 영역 스캔 카메라 1개 또는 2개	2D22	LUT

1633 Coaxlink Quad G3

펌웨어 변형	설명	호스트 연결 맵	고급 프로세싱
1-카메라	1- 또는 2- 또는 4-연결 영역 스캔 카메라 1개	1D4	FFC LUT CFA
1-카메라, 라인 스캔	1 또는 2- 또는 4-연결 영역 스캔 카메라 1개	1D4	FFC LUT
1-slm- 카메라	8-연결 8-연결 영역 스캔 카메라의 마스터 4-연결 서브 링크	1D8SLM4	LUT
1-sls-카메라	8-연결 8-연결 영역 스캔 카메라의 슬레이브 4-연결 서브 링크	1D8SLS4	LUT
2-카메라	1-또는 2-연결 영역 스캔 카메라 1개 또는 2개	2D22	LUT
2-카메라, 라인 스캔	1-또는 2-연결 라인 스캔 카메라 1개 또는 2개	2D22	LUT
3-카메라	1-또는 2-또는 1개 및 1-연결 영역 스캔 카메라 1 또는 2개	3D211	LUT
4-카메라	1-연결 영역 스캔 카메라 1 또는 2 또는 3 또는 4개	4D1111	LUT
4-카메라, 라인 스캔	1-연결 라인 스캔 카메라 1 또는 2 또는 3 또는 4개의	4D1111	LUT
1-카메라, 4-데이터 스트림	1 또는 2 또는 4 영역 스캔 카메라 1개, 최대 4 데이터 스트림	1D4S4	-

1633-LH Coaxlink Quad G3 LH

펌웨어 변형	설명	호스트 연결 맵	고급 프로세싱
1-카메라	1- 또는 2-또는 4-연결 영역 스캔 카메라 1개	1D4	FFC LUT CFA
1-카메라, 라인 스캔	1 또는 2- 또는 4-연결 영역 스캔 카메라 1개	1D4	FFC LUT
1-slm- 카메라	8-연결 8-연결 영역 스캔 카메라의 마스터 4-연결 서브 링크	1D8SLM4	LUT
1-sls-카메라	8-연결 8-연결 영역 스캔 카메라의 슬레이브 4-연결 서브 링크	1D8SLS4	LUT
2-카메라	1-또는 2-연결 영역 스캔 카메라 1개 또는 2개	2D22	LUT
2-카메라, 라인 스캔	1-또는 2-연결 라인 스캔 카메라 1개 또는 2개	2D22	LUT
3-카메라	1-또는 2-또는 1개 및 1-연결 영역 스캔 카메라 1 또는 2개	3D211	LUT
4-카메라	1-연결 영역 스캔 카메라 1 또는 2 또는 3 또는 4개	4D1111	LUT
4-카메라, 라인 스캔	1-연결 라인 스캔 카메라 1 또는 2 또는 3 또는 4개의	4D1111	LUT
1-카메라, 4-데이터 스트림	1 또는 2 또는 4 영역 스캔 카메라 1개, 최대 4 데이터 스트림	1D4S4	-

1629 Coaxlink Duo PCIe/104-EMB

펌웨어 변형	설명	호스트 연결 맵	고급 프로세싱
1-카메라	1- 또는 2-연결 영역 스캔 카메라 1개	1D2	LUT
1-카메라, 라인 스캔	1 또는 2-연결 영역 스캔 카메라 1개	1D2	LUT
2-카메라	1-연결 영역 스캔 카메라 1개 또는 2개	2D11	LUT

1634 Coaxlink Duo PCIe/104-MIL

펌웨어 변형	설명	호스트 연결 맵	고급 프로세싱
1-카메라	1- 또는 2-연결 영역 스캔 카메라 1개	1D2	LUT
1-카메라, 라인 스캔	1 또는 2-연결 영역 스캔 카메라 1개	1D2	LUT
2-카메라	1-연결 영역 스캔 카메라 1개 또는 2개	2D11	LUT

1635 Coaxlink Quad G3 DF

펌웨어 변형	설명	호스트 연결 맵	고급 프로세싱
1-카메라	1- 또는 2-또는 4-연결 영역 스캔 카메라 1개	1D4	LUT CFA
1-df-카메라	1-또는 2-또는 4-연결 영역 스캔 데이터 전달 카메라 1개	1DF4	LUT CFA
1-카메라, 라인 스캔	1 또는 2- 또는 4-연결 영역 스캔 카메라 1개	1D4	FFC LUT
1-df-카메라, 라인 스캔	1-또는 2-또는 4-연결 라인 스캔 데이터 전달 카메라 1개	1DF4	FFC LUT

1637 Coaxlink Quad 3D-LLE

펌웨어 변형	설명	호스트 연결 맵	고급 프로세싱
1-카메라	1- 또는 2-또는 4-연결 영역 스캔 카메라 1개	1D4	LLE

3602 Coaxlink Octo

펌웨어 변형	설명	호스트 연결 맵	고급 프로세싱
1-카메라	1-또는 2-또는 4- 또는 8-연결 영역 스캔 카메라 1개	1D8	LUT CFA
1-카메라, 라인 스캔	1-또는 2-또는 4-또는 8-연결 라인 스캔 카메라 1개	1D8	LUT
2-카메라	1-또는 2-또는 4-연결 영역 스캔 카메라 1개 또는 2개	2D44	FFC LUT CFA
2-카메라, 라인 스캔	1-또는 2-또는 4-연결 라인 스캔 카메라 1개 또는 2개	2D44	LUT
4-카메라	1-또는 2-연결 영역 스캔 카메라 1 또는 2 또는 3 또는 4개	4D2222	LUT
4-카메라, 라인 스캔	1-또는 2-연결 라인 스캔 카메라 1 또는 2 또는 3 또는 4개	4D2222	LUT
5-카메라	1-또는 2-또는 4-연결 1개 및 1-연결 영역 스캔 카메라 1 또는 2개 또는 3개 또는 4개	5D41111	LUT
8-카메라	최대 8개의 1-연결 영역 스캔 카메라	8D11111111	LUT

3603 Coaxlink Quad CXP-12

펌웨어 변형	설명	호스트 연결 맵	고급 프로세싱
1-카메라	1- 또는 2-또는 4-연결 영역 스캔 카메라 1개	1D4	LUT CFA
1-카메라, 라인 스캔	1 또는 2- 또는 4-연결 영역 스캔 카메라 1개	1D4	LUT
4-카메라	1-연결 영역 스캔 카메라 1 또는 2 또는 3 또는 4개	4D1111	LUT

3620 Coaxlink Quad CXP-12 JPEG

펌웨어 변형	설명	호스트 연결 맵	고급 프로세싱
4-카메라	1-연결 영역 스캔 카메라 1 또는 2 또는 3 또는 4개	4D1111	CFA JPEG

3621-LH Coaxlink Mono CXP-12 LH

펌웨어 변형	설명	호스트 연결 맵	고급 프로세싱
1-카메라	1-연결 영역 스캔 카메라 1개	1D1	LUT

3622 Coaxlink Duo CXP-12

펌웨어 변형	설명	호스트 연결 맵	고급 프로세싱
1-카메라	1- 또는 2-연결 영역 스캔 카메라 1개	1D2	LUT
2-카메라	1-연결 영역 스캔 카메라 1개 또는 2개	2D11	LUT

3.3. 호스트 연결 맵

CoaXPress 표준은 장치(카메라 또는 데이터 전달 장치)를 무료 연결 구성표를 사용하여 호스트(프레임 그래버)에 연결할 수 있음을 나타냅니다. 대신, Coaxlink의 호스트 인터페이스에는 호스트 커넥터에 대한 장치 연결의 특정 할당이 필요합니다. 이러한 지정의 이름은 **호스트 연결 맵**입니다.

호스트 연결 맵은 제품펌웨어 변형에서 하드 코딩됩니다.



경고 Coaxlink 제품 및 펌웨어 변형은 필수 매핑에 따라 선택해야 합니다!

호스트 연결 맵 명명 규칙

호스트 연결 맵 또는 *HCMAP*는 Coaxlink 카드의 호스트 인터페이스 연결이 장치 (카메라)에 할당되는 방법을 지정합니다.

호스트 연결 맵 - HCMAP -는 Euresys 독점 명명 규칙을 사용하여 머리 글자로 지정됩니다.

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

위치:

- <dev#>는 호스트 인터페이스에 연결할 수 있는 장치 (카메라)의 최대 수를 선언합니다.
 - 단일 장치 호스트 인터페이스의 경우 1
 - 2-장치 호스트 인터페이스의 경우 2
 - ...
- <dev-type>는 장치 유형을 선언합니다.
 - 표준 CoAXPress 장치용 D
 - 데이터 전달 체계에 사용되는 가상 장치의 DF
- <con#>는 각 장치에 사용할 수 있는 연결 수를 선언합니다.
 - 단일 연결 장치의 경우 1
 - 2-연결 장치용 2
 - ...
 - 이 필드는 각 장치에 대해 한 번 반복됩니다.
- <str #> S는 장치가 허용하는 최대 데이터 스트림 수를 선언합니다.
 - 스트림이 하나 뿐인 경우 이 필드는 생략됩니다
 - 장치 당 최대 4개의 데이터 스트림을 위한 4S
- SL<-con#>는 서브 링크 당 연결 수를 선언합니다.
 - 4-연결 서브 링크를 위한 SL4

이 필드는 하위 링크가 없을 때 생략됩니다.

예제

HCMAP 2D22는 2개의 표준 1-데이터-스트림 CoaXPress 장치 및 각 장치에 2개의 연결을 갖춘 호스트 인터페이스를 지정합니다.

HCMAP 1D4S4는 1개의 표준 CoaXPress 장치, 최대 4개의 데이터 스트림 및 장치 당 4개의 연결로 호스트 인터페이스를 지정합니다.

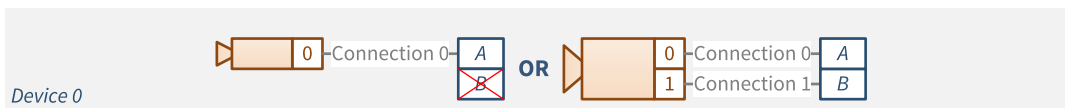
1D1 호스트 연결 맵

1 연결 장치 하나



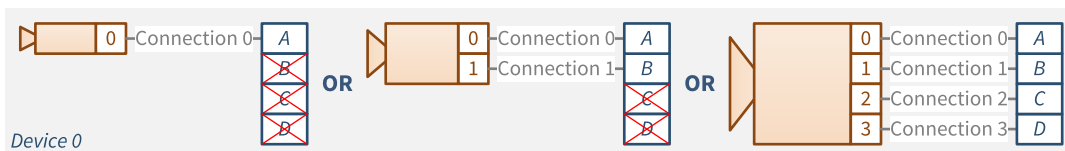
1D2 호스트 연결 맵

1 또는 2 연결 장치용 하나



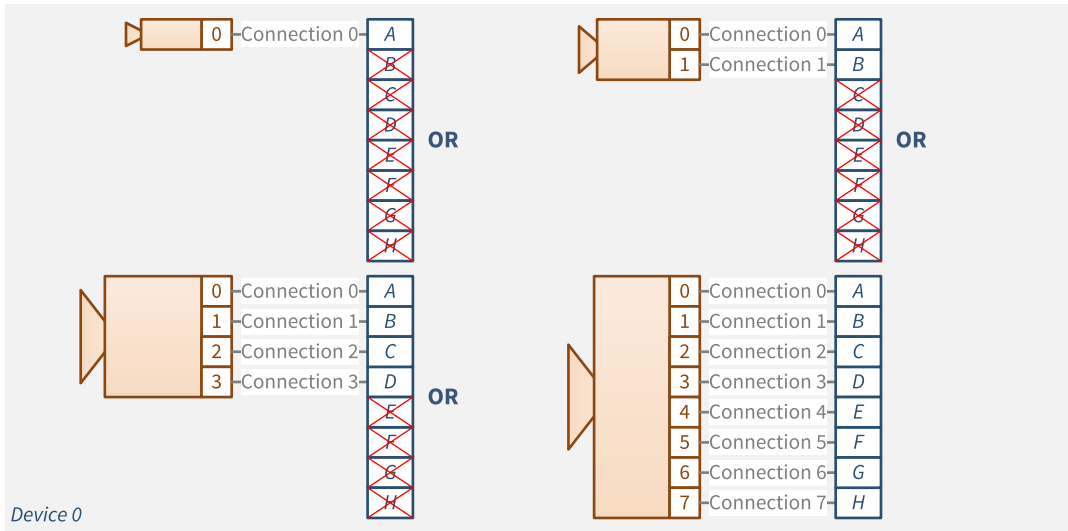
1D4 호스트 연결 맵

1 또는 2 또는 4 연결 장치 하나



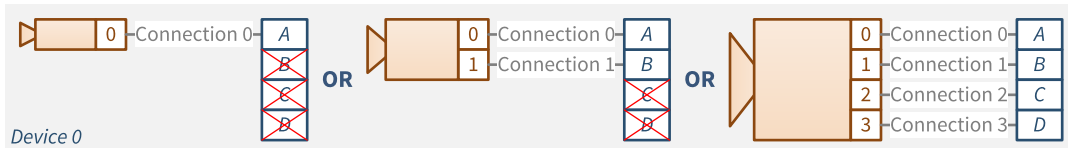
1D8 호스트 연결 맵

1 또는 2 연결 장치용 하나



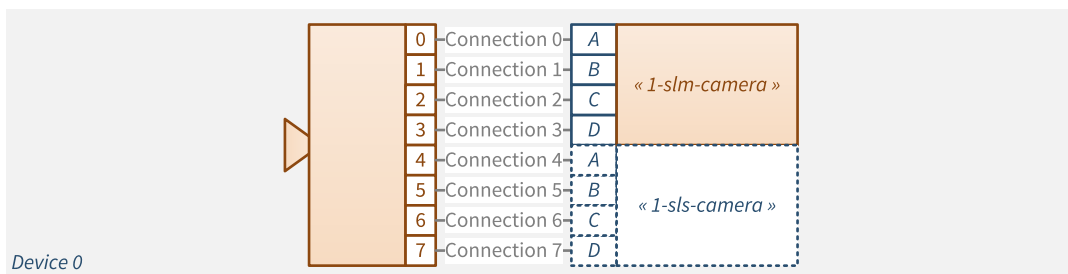
1D4S4 호스트 연결 맵

1 또는 2 또는 4 연결 장치 하나, 최대 4 데이터 스트림



1D8SLM4 호스트 연결 맵

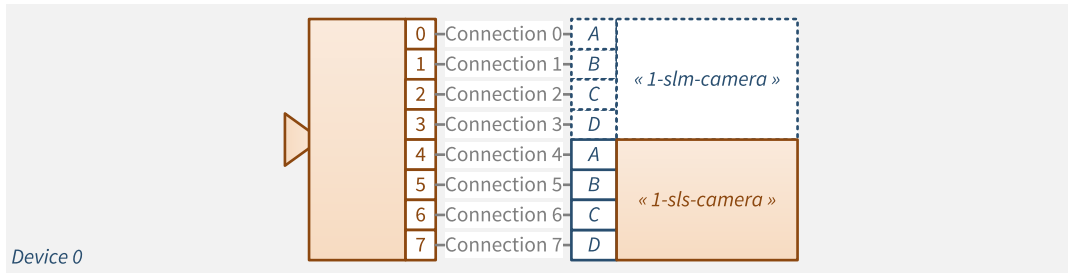
마스터 8 연결 장치의 4 연결 서브 링크



참조: "하위 링크 수집" 페이지166는 8개의 연결 카메라와 2개의 Coaxlink 카드 연결 방식을 나타냅니다.

1D8SLS4 호스트 연결 맵

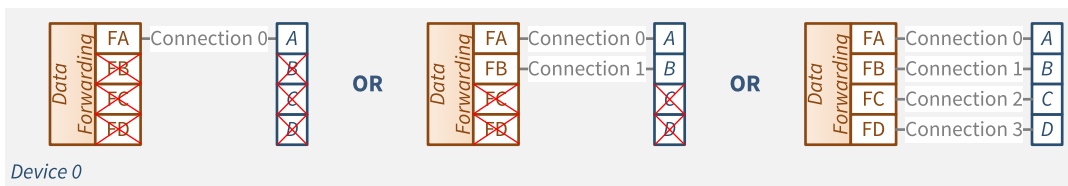
8 연결 장치의 슬레이브 4 연결 서브 링크



참조: "하위 링크 수집" 페이지166는 8개의 연결 카메라와 2개의 Coaxlink 카드 연결 방식을 나타냅니다.

1DF4 호스트 연결 맵

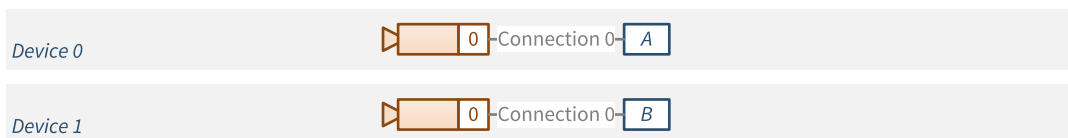
1 또는 2 또는 4 연결 장치 하나



참조: "CoaXPress 데이터 전달" 페이지169는 슬레이브 데이터 포워딩 장치의 연결 방식을 나타냅니다.

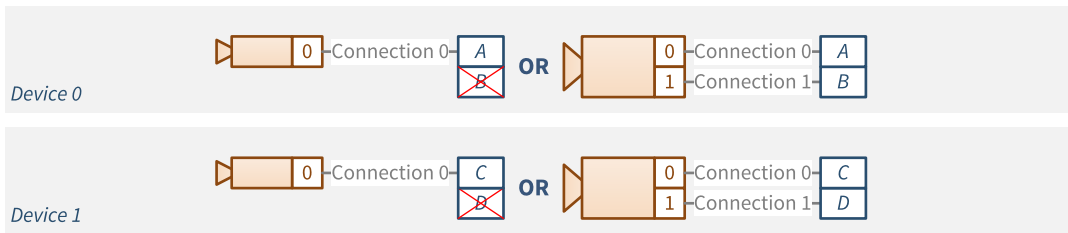
2D11 호스트 연결 맵

1 연결 장치 하나 또는 둘



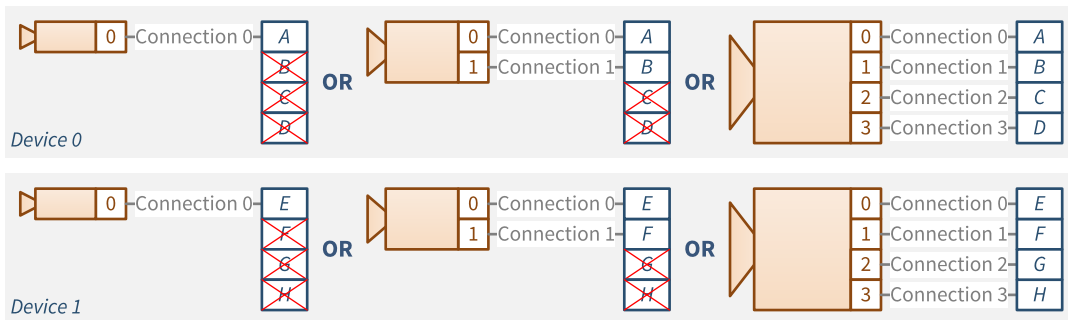
2D22 호스트 연결 맵

1 또는 2 연결 장치용 하나 또는 둘



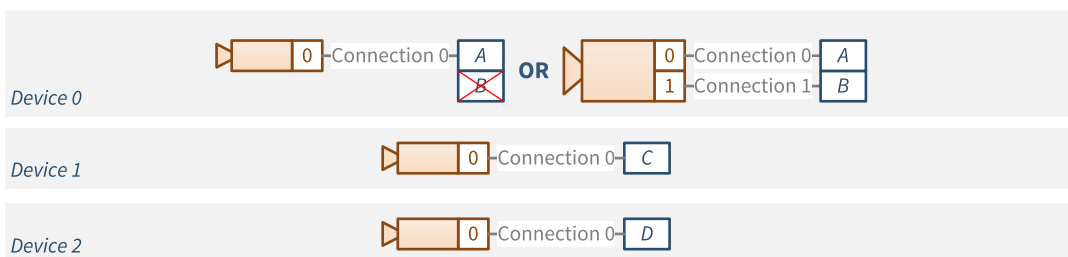
2D44 호스트 연결 맵

1 또는 2 또는 4 연결 장치 하나 또는 둘



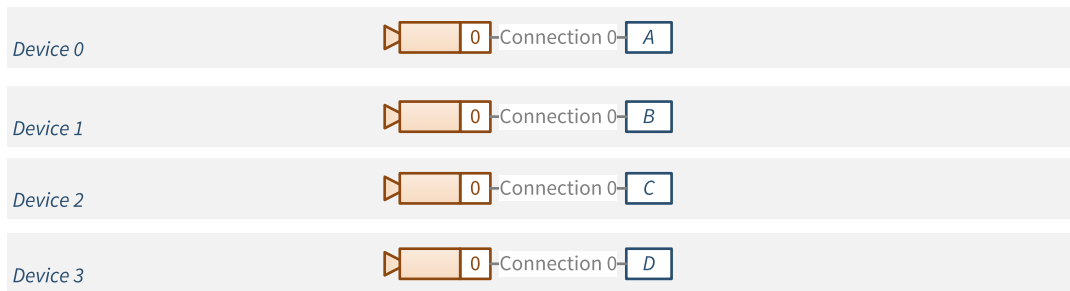
3D211 호스트 연결 맵

1 또는 2 연결 하나 및 1 연결 장치 하나 또는 둘



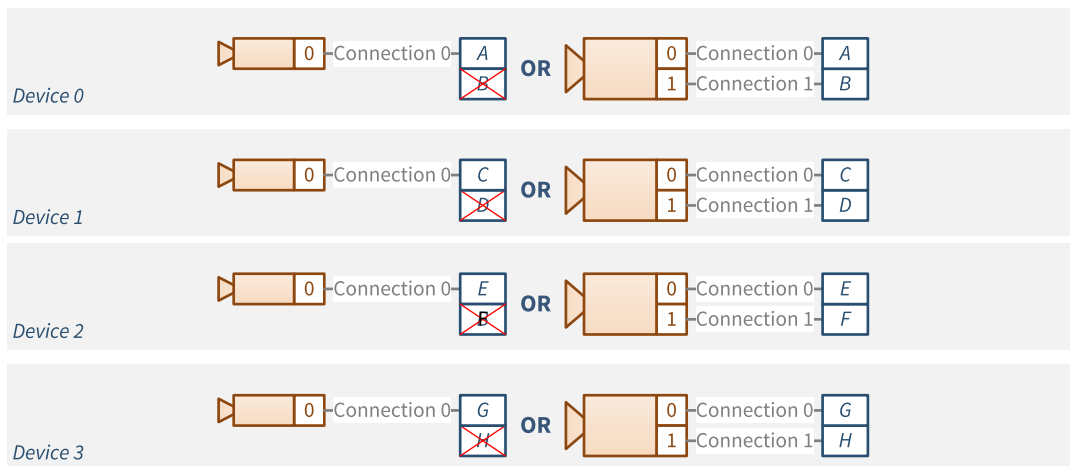
4D1111 호스트 연결 맵

1 또는 1 연결 장치용 하나 또는 둘 또는 셋 또는 넷



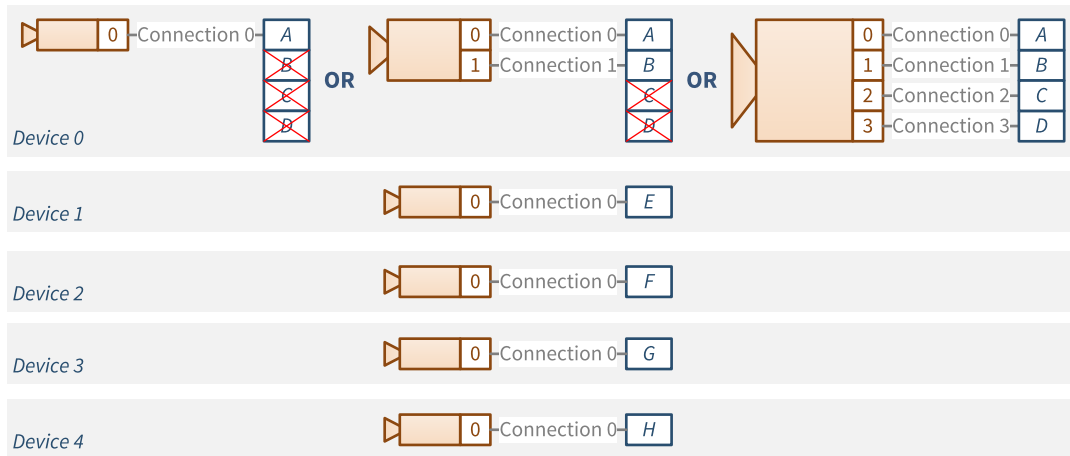
4D2222 호스트 연결 맵

1 또는 2 연결 장치 하나 또는 둘 또는 셋



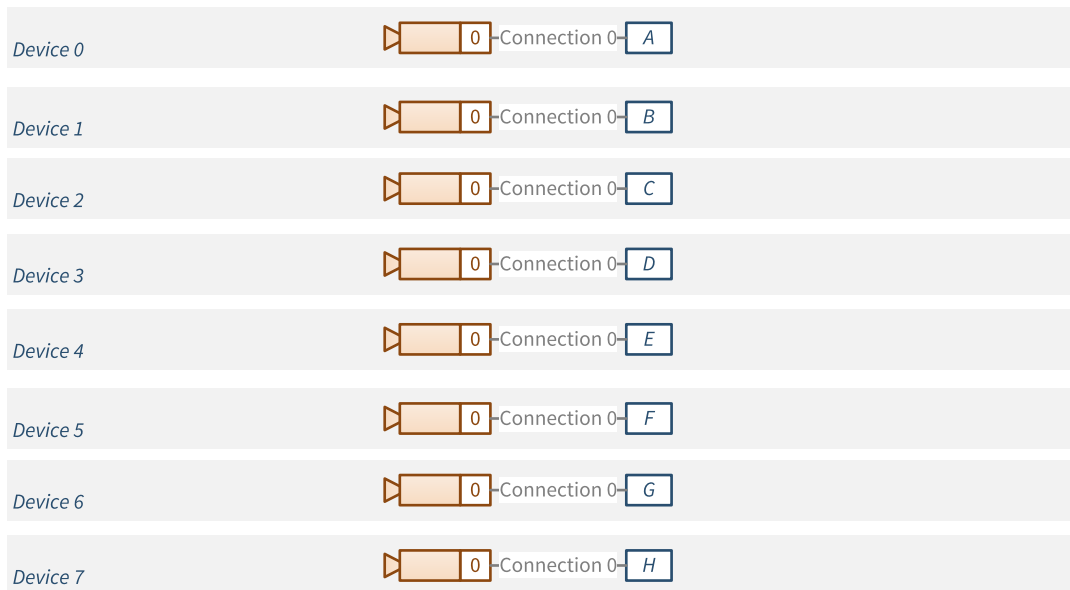
5D41111 호스트 연결 맵

1 또는 2 또는 4 연결 장치용 하나 또는 1 연결 장치 하나 또는 둘 또는 셋 또는 넷



8D1111111 호스트 연결 맵

1 연결 장치 최대 여덟



3.4. 링크 설정

자동 링크 설정

Coaxlink 드라이버는 CoaXPress 1.0 및 CoaXPress 1.1 장치에 대한 자동 링크 검색 및 설정을 제공합니다.

CoaXPress 호스트 인터페이스의 각 연결에 대해 검색 절차는 다음을 결정합니다:

- CoaXPress 장치의 존재
- 다운 연결 속도 (장치-호스트)
- 연결 ID

검색 결과는 인터페이스 모듈의 `CxpConnectionState`, `CxpDownConnectionSpeed` 및 `CxpDeviceConnectionID` 기능을 통해 보고됩니다.

결과 링크 설정이 적절한지 확인하기 위해 사용자가 초대됩니다:

- 링크 대역폭 (링크 속도 및 연결 수) 측면에서 애플리케이션 요구 사항에 대해,
- 대상 Coaxlink 제품 / 펌웨어 조합이 지원하는 카메라 연결 방식면에서 Coaxlink에 대해

수동 링크 설정

필요한 경우 사용자는 원격 장치의 CoaXPress 링크를 수동으로 설정할 수 있습니다.

이것은 Coaxlink 장치 모듈의 `CxpLinkConfiguration` GenApi 기능에 적절한 값을 할당하여 카메라 브랜드에 관계없이 달성할 수 있습니다.

Preferred 값을 지정하면 카메라의 기본 연결 구성이 적용됩니다:

- 링크 속도가 지정된 값으로 설정됩니다
- 링크 폭은 지정된 값으로 설정되지만 Coaxlink 측에서 사용 가능한 연결 수로 제한될 수 있습니다

3.5. CoaXPress를 능가하는 힘

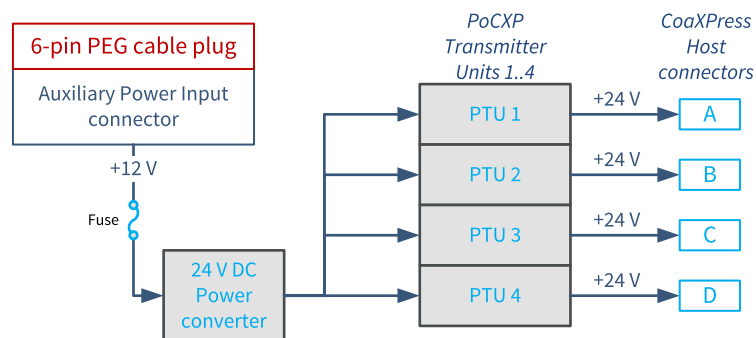
CoaXPress 호스트 커넥터의 각 연결은 CoaXPress 케이블을 통해 카메라에 전원을 공급할 수 있습니다.

전원

12V DC 전원

Applies to:

Mono	Duo	Quad	QuadG3	QuadG3LH	QuadG3DF
Quad3DLLE	Octo	QuadCXP12	QuadCXP12J	MonoCXP12LH	DuoCXP12



12V DC 전원 공급 장치가 있는 카드의 PoCXP 전원 분배 체계

외부 12V DC 공급 장치는 6핀 PEG 케이블을 통해 보조 전원 입력 커넥터에 연결됩니다. 12V 전력은 12V~24V 조정 컨버터로 변환되어 각 전력 전송 장치로 전달됩니다.



참고 12V DC 전원은 일반적으로 호스트 PC의 전원 공급 장치를 통해 공급됩니다.

이 기능 `AuxiliaryPowerInput` 장치와 Coaxlink 보조 전원 입력 커넥터 사이의 연결 상태를 보고합니다.

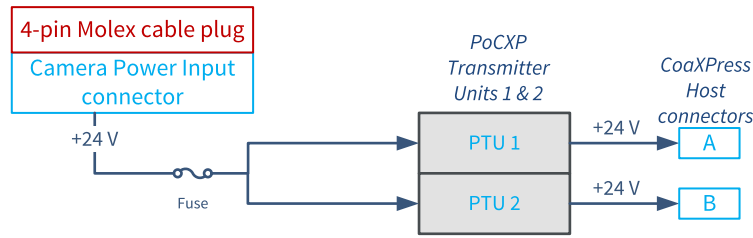
이 기능 `CxpPoCxpPowerInputStatus`

24 V DC 전원

Applies to:

Duo104EMB

Duo104MIL



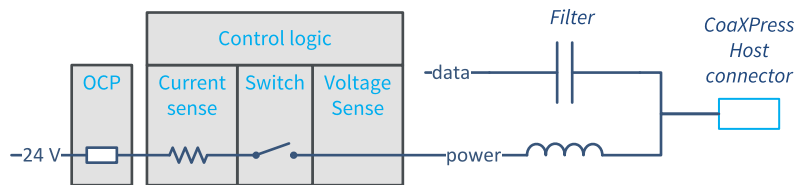
24 V DC 전원 공급 장치가 있는 카드의 PoCXP 전원 분배 체계

외부 24V 전원 공급 장치는 4핀 케이블을 통해 보조 전원 입력 커넥터에 연결되어 각 전력 전송 유닛으로 전달됩니다.

이 기 `AuxiliaryPowerInputxlink` 보조 전원 입력 커넥터 사이의 전원 케이블 연결 상태를 보고합니다.

이 기 `CxpPoCxpPowerInputStatus`

전력 전달 장치 유닛



PoCXP 전력 전송 유닛

PTU(전력 전달 장치 유닛)는 장치에 안전한 전원 공급을 담당합니다. CoaXPress 호스트를 위한 CoaXPress 표준의 요구 사항을 충족합니다.

- 커넥터 당 17W*의 24V DC 전원을 장치에 제공 할 수 있습니다.
(*) 3621-LH Coaxlink Mono CXP-12 LH 의 경우 25W
- 과전류 보호 장치 - OCP
- 자동 CoaXPress PoCXP 탐지 방법을 지원합니다.

또한 다음과 같은 기능을 어플리케이션에 제공합니다:

- 자동 전원 공급 중단 또는 중단
- 트립시 OCP 재설정
- PoCXP 감지 저항 의 범위 제어

자동 PoCXP 제어

`CxpPoCxpAuto` 명령의 실행시, PTU 제어기는 *PoCXP 장치 검출 절차*를 시작합니다.



참고 Coaxlink 드라이버 버전 3.1부터 시스템 전원을 켤 때 자동 PoCXP 전원 공급이 활성화됩니다. 애플리케이션은 더 이상 `CxpPoCxpAuto` 명령을 발행하여 PoCXP 전원 공급을 활성화할 필요가 없습니다.

PoCXP 장치 검출 절차가 성공적으로 종료되면, PTU는 스위치를 폐쇄함으로써 전력을 인가합니다.

*PoCXP 장치 검출 절차가 실패*하면, 컨트롤러는 전력을 공급하지 않고 새로운 PoCXP 검출 절차를 재시도합니다. 가능한 실패 원인은 다음과 같습니다:

- 외부 전원이 연결되지 않음 (=) `AuxiliaryPowerInputAuxiliaryPowerInput`
- 외부 전원이 꺼져 있음 (=) `CxpPoCxpPowerInputStatusCxpPoCxpPowerInputStatus`
- 카메라가 연결되어 있지 않습니다
- 연결된 카메라가 PoCXP와 호환되지 않습니다

전원이 공급되면 컨트롤러는 다음 상황 중 하나가 발생할 때까지 그 상태를 유지합니다:

- 애플리케이션은 명령을 실행하여 전원 공급을 비활성화합니다. `PoCxpTurnOff`
- 외부 전원이 연결되어 있지 않음 (=) `CxpPoCxpPowerInputStatusCxpPoCxpPowerInputStatus`
- 외부 전원이 꺼져 있음 (=) `CxpPoCxpPowerInputStatusCxpPoCxpPowerInputStatus`
- CoaXPress 케이블 연결이 끊어졌습니다 (0.3초의 시간 간격 동안 측정된 평균 출력 전류가 8mA 미만임)
- OCP 트립 (20 밀리 초 이상의 시간 동안 출력 전압이 이하로 떨어짐)

수동 PoCXP 제어

명령 실행시 PTU는 스위치를 끄고 `CxpPoCxpTurnOff` 활성화합니다. 그 상태에서, PTU는 PoCXP 검출 절차를 수행하지 않습니다.

기능

은 `CxpPoCxpConfigurationStatusCxpPoCxpConfigurationStatusCxpPoCxpConfigurationStatus`

이 기 `CxpPoCxpStatus`합니다: , 또는 `.CxpPoCxpStatusCxpPoCxpStatusCxpPoCxpStatus`

PoCXP 감지 모드 제어

인터페이스 모듈의 **CxpPoCxpDetectionMode** 기능은 CoaXPress 감지 모드에서 표준 또는 확장(기본값) 전원을 선택합니다.

Extended로 설정하면 Coaxlink 카드의 PoCXP 장치 감지가 확장된 저항값 범위로 구성됩니다. 이를 통해 PoCXP 감지 저항의 범위 사양을 완전히 준수하지 않는 카메라를 유효한 PoCXP 카메라로 감지하고 전원을 공급할 수 있습니다. 초기화 후의 기본값입니다.

Standard로 설정하면 Coaxlink 카드의 PoCXP 장치 감지가 확장된 저항값, 즉 $4.7k\Omega \pm 10\%$ 범위로 구성됩니다.



경고 이 설정은 영구적이지 않습니다.

과전류 보호

OCP 회로는 두 종류의 보호 기능을 제공하는 PTC 장치로 제작됩니다:

- 과부하 보호는 부하가 과도한 경우를 처리합니다.
- 단락 보호 기능은 우발적인 단락의 경우를 다룹니다.

과부하의 경우 PTC는 현재 수준 및 주변 온도에 따라 몇 초 또는 몇 분 후 트립됩니다(= 순차적으로 회로가 열림). 전류가 높을수록 트립 시간이 짧아집니다. 주변 온도에도 동일하게 적용됩니다.

단락의 경우 PTC가 즉시 이동합니다. 결과적으로, PTU 컨트롤러는 트립된 상태가 되어 스위치를 개방합니다. 트립된 PTC 장치는 냉각된 후에 전도 상태로 되돌아갑니다. 이 작업은 몇 초가 걸릴 수 있습니다. 그러나 PTU 제어기는 응용 프로그램이 명령을 내릴 때까지 트립된 상태를 유지합니다. **CxpPoCxpTripReset** 트립 상태를 벗어나지 않은 상태에서 PTU는 새로운 PoCXP 장치 감지를 시작할 수 있으며, 성공하면 전원을 다시 설정합니다.



참고 PTC는 트리핑없이 전체 작동 온도 범위에서 17W의 전력을 유지할 수 있는 크기입니다.

(*) **3621-LH Coaxlink Mono CXP-12 LH** 의 경우 25W



참고 17 W 이상의 전력을 추출하거나 작동 온도 범위를 초과하여 Coaxlink 카드를 작동시키는 것은 예기치 않은 PTC 트립을 유발할 수 있으므로 금지됩니다.

(*) **3621-LH Coaxlink Mono CXP-12 LH** 의 경우 25W

3.6. CoAXPress I/O 채널

CoaXPress 1.0 및 1.1 표준에 따르면, CoaXPress I/O 채널:

- CoaXPress 링크 (I/O, 스트림, 컨트롤)의 세 가지 논리 채널 중 하나입니다
- CoaXPress 링크의 마스터 연결(연결 0)에 대해서만 정의됩니다
- 호스트와 장치 사이의 우선 순위가 높은 "트리거" 전송에 사용됩니다
- CoaXPress 1.0에서만 호스트와 장치간에 GPIO 레지스터 상태를 교환하는 데 사용됩니다

Coaxlink는 CoaXPress 호스트-장치 트리거만 구현합니다!



참고 CoaXPress 장치-호스트 트리거 및 CoaXPress 1.0 GPIO는 구현되지 않습니다.

3.7. CoaXPress 호스트-장치 트리거

CoaXPress 호스트-장치 트리거는 호스트(프레임 그래버)가 CoaXPress 링크를 통해 장치(카메라)를 트리거할 수 있게 해주는 CoaXPress I/O 채널의 기능입니다.

Coaxlink의 CoaXPress 호스트 인터페이스는 연결된 각 장치에 대해 하나의 CoaXPress 호스트 대 장치 트리거 송신기를 구현합니다.

호스트-장치 트리거 소스

CoaXPress 호스트-장치 트리거 송신기는 다음 출처에서 제공할 수 있습니다:

- 관련 카메라 및 조명 컨트롤러의 Camera Trigger 출력
- 모든 입력 가능 범용 I/O

트리거 소스는 **CameraControlMethodGenICam** 기능을 통해 간접적으로 조절됩니다.

- **CameraControlMethod**가 **RG** 또는 **RC**로 설정된 경우 트리거 소스는 연관된 카메라 및 조명 컨트롤러의 카메라 트리거 출력입니다.
- **CameraControlMethod**가 **EXTERNAL**로 설정된 경우:
 - 트리거 소스는 I/O 도구 상자의 전용 LIN 도구(Device0의 경우 LIN1, Device1의 경우 LIN2, Device2의 경우 LIN3, Device3의 경우 LIN4)의 라인 소스입니다.
 - 모든 입력 가능 GPIO 라인은 전용 LIN 툴의 라인 소스-**LineInputToolSource**-를 구성하여 트리거 소스로 사용할 수 있습니다.
 - 외부 트리거 신호의 극성은 선택된 I/O 조절 블록의 **LineInverter** 설정으로 설정할 수 있습니다.
 - 글리치 제거 필터의 시간 상수는 선택된 I/O 조절 블록의 **LineFilterStrength** 설정을 통해 조정할 수 있습니다.
- **CameraControlMethod**를 **NC**로 설정하면 호스트-장치 트리거 송신기가 비활성화됩니다.

호스트-장치 트리거 송신기 - 기본 설정

CoaxPress 호스트-장치 트리거 송신기의 Coaxlink 구현은 CoaXPress 1.0 및 1.1 표준의 저속 CoaXPress 호스트-장치 트리거 요구 사항을 기본 설정으로 구성할 때 준수합니다:

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

송신기는 트리거 소스 신호의 양쪽 에지에서 트리거 트랜잭션을 시작합니다:

- 수신 장치가 일정한 대기 시간으로 이벤트를 재생성할 수 있도록 지연값을 계산합니다.

참조: "카메라 트리거 대기 시간 보정" 페이지46

- 다음 문자 경계에서 저속 호스트 - 장치 연결에 우선 순위가 높은 "트리거 패킷"을 삽입합니다.

그런 다음 송신기는 장치(카메라)로부터 확인을 기다립니다:

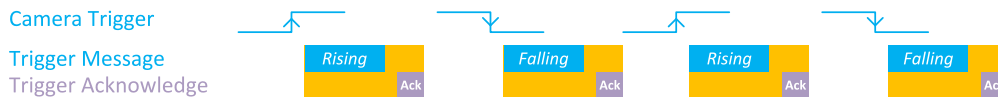
- 타임 아웃이 만료되기 전에 확인 응답이 수신되면 트랜잭션은 정상적으로 종료됩니다.
- 20 μs 시간 초과 내에 수신 확인이 수신되지 않으면 송신기는 재시도를 수행합니다: 즉, 트리거 패킷을 재전송하고 수신 확인을 위한 새로운 대기 기간을 시작합니다.
- 3회 후에 승인이 수신되지 않으면 트랜잭션이 비정상적으로 종료됩니다.

트랜스미터는 이전 트랜잭션이 완료되지 않은 상태에서 새 트랜잭션을 시작하지 않습니다.

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



기본 설정을 사용하여 메시지 트랜잭션 트리거
 사례 1과 사례 2: 카메라가 예상대로 각 메시지를 확인합니다
 사례3: 카메라의 응답이 없습니다. 3회 재 시도 후 중단

이벤트 보고

송신기는 다음 이벤트를 보고합니다:

- **CxpTriggerAck**: CoaXPress 호스트-장치 트리거 패킷에 대한 확인 응답을 받았습니다.
- **CxpTriggerResend**: CoaXPress 호스트-장치 트리거 패킷 재전송.

호스트-장치 트리거 송신기 - 대체 설정

송신기는 사용자 정의할 수 있습니다:

- 메시지 형식 조절을 사용하여 소스 신호의 상승 에지에서만 트리거 메시지를 보내기
- 메시지 확인 조절을 사용하여 응답 시간 제한 및 재시도 횟수를 구성하기

메시지 형식 조절

Coaxlink의 호스트 대 장치 트리거 송신기 장치는 `CxpTriggerMessageFormat` GenICam 기능과 함께 "메시지 형식" 조절 기능을 제공합니다.

펄스 메시지 형식 (기본값)

기본적으로 `CxpTriggerMessageFormat`는 `Pulse`로 설정됩니다: 송신기는 입력 펄스의 양쪽 가장자리에서 CoaXPress I/O 채널 호스트-장치 트리거 트랜잭션을 생성합니다:

- 상승 에지에 의해 시작된 트랜잭션은 *상승 에지 트리거 패킷*을 호스트에서 장치로 전송합니다.
- 하강 에지에 의해 시작된 트랜잭션은 *하강 에지 트리거 패킷*을 호스트에서 디바이스로 전송합니다.



참고모든 트리거 펄스에는 두 개의 서로 다른 CoAXPress I/O 채널 트랜잭션이 필요합니다!

상승 에지 메시지 형식

`CxpTriggerMessageFormat`가 `RisingEdge`로 설정되면, 송신기는 입력 펄스의 *상승 에지*에서만 CoaXPress I/O 채널 호스트-장치 트리거 트랜잭션을 생성합니다.

트랜잭션은 항상 호스트에서 장치로 *상승 에지 트리거 패킷*을 전송합니다.



참고모든 트리거 펄스에는 단일 CoAXPress I/O 채널 트랜잭션이 필요합니다.



참고이 형식은 그래버가 노출 시간을 제어하는 것을 허용하지 않습니다!

토글 메시지 형식

`CxpTriggerMessageFormat`가 `Toggle`로 설정되면, 송신기는 입력 펄스의 *상승 에지*에서만 CoaXPress I/O 채널 호스트-장치 트리거 트랜잭션을 생성합니다.

트랜잭션은 *상승 에지 트리거 패킷*과 *하강 에지 트리거 패킷*을 번갈아 전송합니다.



참고모든 트리거 펄스에는 단일 CoAXPress I/O 채널 트랜잭션이 필요합니다.



참고이 형식은 그래버가 노출 시간을 제어하는 것을 허용하지 않습니다!

이 `CxpTriggerLevel` 기능을 사용하면 애플리케이션에서 CoaXPress 호스트의 현재 레벨을 장치 트리거 신호로 설정 및/또는 가져올 수 있습니다.

메시지 확인 조절

호스트-장치 트리거 송신기 장치는 사용자가 구성 할 수 있는 트리거 패킷 확인 메커니즘을 제공합니다:

- 타임 아웃 값은 `CxpTriggerAckTimeout` GenICam 기능을 사용하여 설정할 수 있습니다.
- 재시도 횟수는 `CxpTriggerMaxResendCount` GenICam 기능을 사용하여 설정할 수 있습니다.

확인 검사 사용 (기본값)

기본적으로 `CxpTriggerAckTimeout`는 20.0(20 마이크로초)로 설정되고 `CxpTriggerMaxResendCount`는 3로 설정됩니다.

Coaxlink는 모든 트리거 패킷에 대한 응답으로 I/O 채널 확인 패킷을 기대합니다. 응답 패킷이 20 μ s 시간 초과값 내에 수신되지 않으면 송신기는 트리거 패킷을 다시 보냅니다. 최대 3회까지 재시도합니다.

큰 `CxpTriggerAckTimeout` 값을 설정하면 장치가 트리거 패킷을 확인하는 데 더 많은 시간이 소요됩니다.

확인 검사 비활성화

`CxpTriggerAckTimeout`를 0로 설정하면 승인 0 메커니즘이 비활성화됩니다. 트리거 트랜잭션은 트리거 패킷을 보낸 직후 종료됩니다.

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;



대체 설정을 사용하여 더 높은 트리거 속도를 허용하는 메시지 트랜잭션 트리거

가장 빠른 트리거 속도를 위한 대체 설정

590.2 kHz @ CXP-10 및 CXP-12 링크 속도 또는 297.6 kHz @ CXP-6 및 더 낮은 링크 속도는 다음의 경우에 가장 빠른 트리거 속도를 얻을 수 있습니다.

- CameraControlMethod = RC (비동기 리셋 카메라, 카메라 제어 노출),
- CxpTriggerAckTimeout = 0 (확인 체크 사용 안 함),
- CxpTriggerMessageFormat = Rising 또는 CxpTriggerMessageFormat = Toggle.

카메라 트리거 대기 시간 보정

트리거 정확도

호스트-장치 트리거 패킷은 CoaXPress Link의 저속 연결을 통해 전송됩니다. 트리거 패킷의 전송은 문자의 경계에서만 시작할 수 있습니다. .이로 인해 한 문자 전송 시간에 해당하는 지터가 발생합니다.

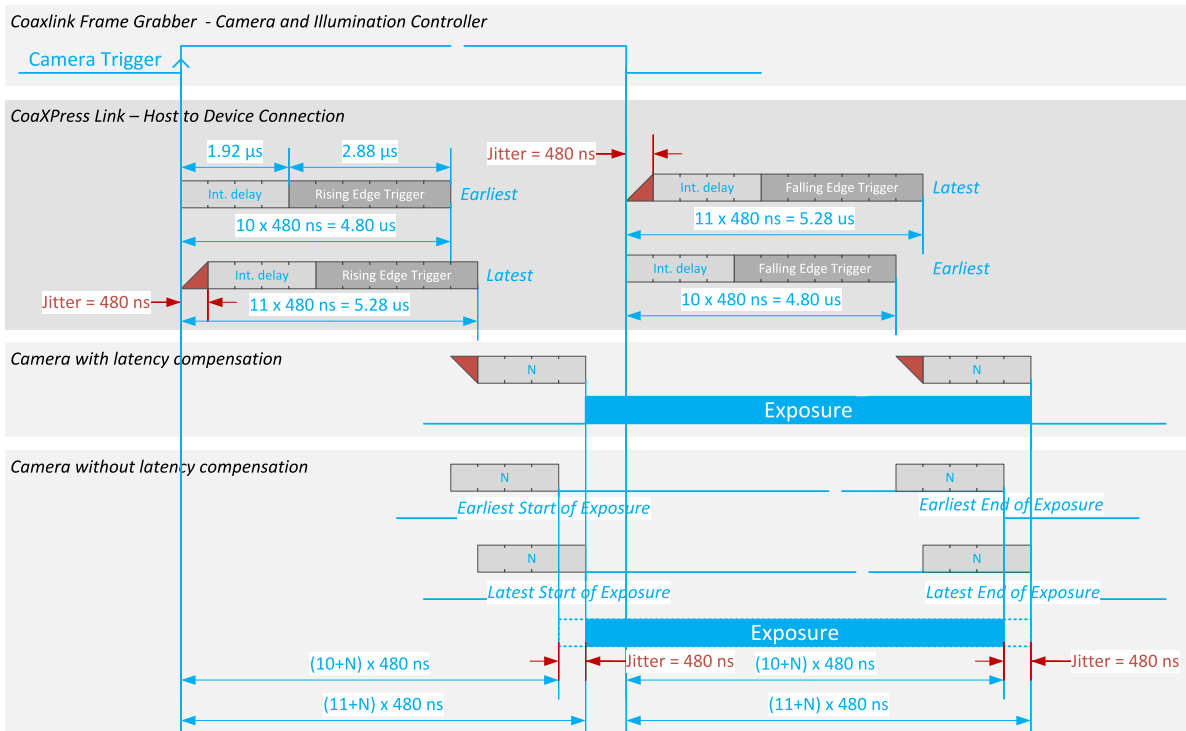
- CXP-10 및 CXP-12 링크 속도에서 240 나노초 또는
- CXP-6 및 낮은 링크 속도에서 480 나노초 또는.

트리거 지터를 최소화하기 위해 트리거 이벤트와 전송되는 트리거 패킷 사이의 시간은 트리거 패킷에 비트주기의 1/24 단위로 표현된 지연 값으로 인코딩됩니다.

- CXP-10 및 CXP-12 링크 속도에서 1 나노초 또는
- CXP-6 및 낮은 링크 속도에서 2 나노초 또는.

수신기 (카메라)는 이 값을 사용하여 *일정한 지연 시간*으로 트리거 이벤트를 재생성할 수 있습니다. 디코딩된 메시지를 한 문자 시간의 나머지 부분만큼 지연시켜 전송 지터를 보상합니다.

CoaXPress 카메라 트리거 트랜스미션 타이밍 @ CXP-6 및 더 낮은 링크 속도



CIC 카메라 트리거-센서 노출 타이밍 다이어그램 @ CXP-6 및 더 낮은 링크 속도

위 다이어그램은 호스트에서 장치 CoaXPress 트리거 메시지를 사용하여 프레임 그래버에서 카메라까지 CoaXPress 링크를 통해 카메라 트리거 이벤트를 전파하는 데 필요한 시간 지연을 보여줍니다.

위의 다이어그램은 다음을 가정합니다:

- CameraControlMethod는 로 설정됩니다.RG
- 카메라가 트리거 메시지를 올바르게 인식하고 새 노출을 효과적으로 시작합니다.

카메라 트리거 신호(Coaxlink 카드 내부)의 상승 에지(또는 하강 에지)에서부터 CoaXPress 링크까지의 지연은 다음으로 설정됩니다.

- 저속 CoaXPress 연결의 다음 문자 경계까지의 시간 지연에 해당하는 0-480 ns의 가변 지연.
- 트리거 전송기 구현에서 4- 문자 파이프 라인 지연에 해당하는 1.92 μs의 고정 지연
- 2.88 μs의 고정 지연은 6 문자 메시지 전송 시간에 해당합니다.

CoaXPress Link에서 노출의 시작(또는 끝)까지의 지연은 카메라에 따라 다릅니다. 위의 그림에서,이 지연은 N 문자 시간 (N=4)으로 가정합니다.

지터-보상 카메라

카메라가 CoaXPress 지터 보상을 구현하면 송신기에서 도입된 1-문자 지터 (480 ns)를 완전히 보상할 수 있습니다.

전반적인 대기 시간은 고정되어 있지만 카메라에 따라 달라집니다. Coaxlink 제품에서 가능한 최저 대기 시간은 $11 \times 480 \text{ ns}$ 입니다. $5.28 \mu\text{s}$.

보정 후 잔류 지터는 4 ns 정도로 낮을 수 있습니다.

지터-보상되지 않은 카메라

카메라가 CoaXPress 지터 보상을 구현하지 않으면 송신기에서 도입된 1-문자 지터(480 ns)가 유지됩니다.

전반적인 대기 시간은 가변적이며 카메라에 따라 다릅니다. Coaxlink 제품에서 가능한 최저 대기 시간은 $(10 \sim 11) \times 480 \text{ ns}$ 입니다. $(4.80 \sim 5.28) \mu\text{s}$.

CoaXPress 카메라 트리거 트랜스미션 타이밍 @ CXP-10 및 CXP-12 링크 속도

카메라 트리거 신호(Coaxlink 카드 내부)의 상승 에지(또는 하강 에지)에서부터 CoaXPress 링크까지의 지연은 다음으로 설정됩니다.

- 저속 CoaXPress 연결의 다음 문자 경계까지의 시간 지연에 해당하는 $0\text{-}2400 \text{ ns}$ 의 가변 지연.
- 트리거 전송기 구현에서 4-문자 파이프 라인 지연에 해당하는 $0.96 \mu\text{s}$ 의 고정 지연
- 1.44 Ms 의 고정 지연은 6 문자 메시지 전송 시간에 해당합니다.

CoaXPress Link에서 노출의 시작(또는 끝)까지의 지연은 카메라에 따라 다릅니다. 위의 그림에서, 이 지연은 N 문자 시간 ($N=4$)으로 가정합니다.

지터-보상 카메라

카메라가 CoaXPress 지터 보상을 구현하면 송신기에서 도입된 1-문자 지터 (480 ns)를 완전히 보상할 수 있습니다.

전반적인 대기 시간은 고정되어 있지만 카메라에 따라 달라집니다. Coaxlink 제품에서 가능한 최저 대기 시간은 $11 \times 240 \text{ ns}$ 입니다. $2.64 \mu\text{s}$.

보정 후 잔류 지터는 2 ns 정도로 낮을 수 있습니다.

지터-보상되지 않은 카메라

카메라가 CoaXPress 지터 보상을 구현하지 않으면 송신기에서 도입된 1-문자 지터(240 ns)가 유지됩니다.

전반적인 대기 시간은 가변적이며 카메라에 따라 다릅니다. Coaxlink 제품에서 가능한 최저 대기 시간은 $(10 \sim 11) \times 240 \text{ ns}$ 입니다. $(2.40 \sim 2.64) \mu\text{s}$.

3.8. CoaXPress LED 램프

CoaXPress 호스트 인터페이스의 각 커넥터는 브래킷에 장착된 LED 램프와 관련이 있습니다 (PCIe 카드만 해당).

LED 램프 모드 제어

인터페이스 모듈의 **LampMode** 기능은 램프 작동 모드를 정의합니다:

- **Standard** (기본값)로 설정하면 램프가 CoaXPress Link 연결 상태를 나타냅니다.
- **Dark**로 설정하면 모든 램프가 꺼집니다.
- **Error**로 설정하면 오류 조건이 감지되지 않으면 모든 램프가 꺼집니다.
- **Custom**로 설정하면 모든 램프는 **LampCustomLedA ... LampCustomLedH** 부울 기능에 의해 각 비트가 주황색이면 1, 0이면 꺼짐으로 매핑되는 비트 필드인 **LampCustomValue**에 의해 제어됩니다.

CoaXPress 호스트 표시기 LED 램프 상태

상태 설명

심벌	표시	상태
	끄기	전원 없음
	주황색	시스템 부팅
	AlternateFlash_12_5 녹색/주황색 ¹	연결 감지가 진행 중입니다. PoCXP가 활성 상태입니다
	Flash_12_5 주황색 ²	연결 감지가 진행 중입니다. PoCXP 미사용
	AlternateFlash_0_5 적색/녹색	호환되지 않는 장치/호스트. PoCXP 활성
	AlternateFlash_0_5 적색/주황색	호환되지 않는 장치/호스트. PoCXP 미사용

¹연결 감지가 더 빠르더라도 최소 1초 동안 표시됩니다.

²연결 감지가 더 빠르더라도 최소 1초 동안 표시됩니다.

심벌	표시	상태
	빨간색	PoCXP 과전류
	녹색	장치/호스트가 연결되었지만 전송중인 데이터가 없습니다
	Flash_1 주황색	장치/호스트 연결, 이벤트 대기(예: 트리거, 노출 펄스)
	Flash_12_5 녹색	장치/호스트가 연결되었지만 데이터가 전송중입니다
	500 밀리초 적색 펄스 ¹	데이터 전송 중 오류(예: CRC 오류, 단일 비트 오류 감지)
	AlternateFlash_0_5 녹색/주황색	전송되는 연결 테스트 패킷
	Flash_12_5 적색	시스템 오류 (예: 내부 오류)

깜박이는 상태 타이밍 정의

표시	빈도	듀티 사이클
Flash_12_5	12.5 Hz	25%(20밀리초 켜기, 60밀리초 끄기)
Flash_1	1 Hz	20%(200밀리초, 800밀리초)
Flash_0_5	0.5 Hz	50%(1초 켜기, 1초 끄기)
AlternateFlash_12_5	12.5 Hz	25%(컬러 1 20밀리초 켜짐, 60밀리초 꺼짐, 컬러 2 20밀리초 켜짐, 60밀리초 꺼짐)
AlternateFlash_0_5	0.5 Hz	50%(컬러 1 1초 켜짐, 1초 꺼짐, 컬러 2 1초 켜짐, 1초 꺼짐)

¹다중 오류의 경우, 다음 오류가 표시되기 전에 적어도 두 개의 녹색 Flash_12_5 펄스가 있어야 합니다.

3.9. 연결 테스트

Coaxlink 호스트 인터페이스는 CoaXPress 1.1 표준의 섹션 8.7에 정의된 절차에 따라 CoaXPress 링크의 품질 업/다운 연결을 테스트하는 연결 테스트 기능을 제공합니다.

각 개별 CoaXPress 커넥터의 경우,

- 테스트 발생기
- 테스트 수신기

테스트 발생기는 시퀀스 발생기에 의해 생성된 알려진 테스트 패턴을 포함하는 테스트 데이터 패킷을 송신합니다. 전송된 각 테스트 패킷에 대한 패킷 카운터를 증가시킵니다.

테스트 수신기는 수신된 테스트 데이터 패킷 콘텐츠를 자신의 로컬 시퀀스 생성기와 비교합니다. 데이터 패킷에서 다른 각 단어에 대한 오류 카운터를 증가시키고 수신된 각 테스트 패킷에 대한 패킷 카운터를 증가시킵니다.



참고 테스트 패킷 카운터는 전송 및 수신된 테스트 패킷 수를 표시하므로 오류 카운터의 값의 통계적 의미에 대한 판단을 허용합니다.



참고 장치-호스트 및 호스트-장치 연결 테스트는 동시에 실행할 수 있습니다.

3.10. CoaXPress 2.0 오류 카운터

‘CoaXPress 2.0 오류 카운터’는 CoaXPress 프로토콜이 각 개별 CoaXPress 연결에서 감지할 수 있는 오류를 추적합니다.

오류 카운터

연결 잠금 손실 카운터

CoaXPress 호스트 커넥터 인스턴스 당 하나의 카운터가 있어 CoaXPress 수신기에서 발생한 잠금 손실 수를 계산합니다.

8b/10b 인코딩 오류 카운터

CoaXPress 호스트 커넥터 인스턴스 당 하나의 카운터가 있어 CoaXPress 수신기에서 발생한 8b/10b 인코딩 오류 수를 계산합니다.

중복 문자 불일치 카운터

CoaXPress 호스트 커넥터 인스턴스 당 두 개의 카운터가 있어 CoaXPress 수신기에서 발생한 중복 문자 불일치 수를 계산합니다.

- 첫 번째 카운터는 수정할 수 있는 발생 횟수를 계산합니다.
- 두 번째 카운터는 수정할 수 없는 발생 횟수를 계산합니다.

CRC 오류 카운터

CoaXPress 호스트 커넥터 인스턴스 당 CoaXPress 수신기에서 발생한 CRC 오류 수를 세는 카운터가 세 개 있습니다.

- 첫 번째 카운터는 데이터 패킷에서 발생 횟수를 계산합니다.
- 두 번째 카운터는 제어 패킷에서 발생 횟수를 계산합니다.
- 세 번째 카운터는 이벤트 패킷에서 발생 횟수를 계산합니다.

오류 카운터 관리

이 응용 프로그램은 [CoaXPressErrorCounters Category](#)의 인터페이스 모듈 기능을 통해 카운터를 사용합니다

현재 카운트 값 얻기

1. `CxpHostConnectionSelector`를 설정하여 커넥터 인스턴스를 선택합니다.
2. 해당 인터페이스 모듈 기능을 읽기 (예: `CxpLinkLockLossCount`)

카운터 재설정

1. `CxpHostConnectionSelector`를 설정하여 커넥터 인스턴스를 선택합니다.
2. 해당 인터페이스 모듈 기능에 쓰기 (예: `CxpLinkLockLossCountReset`)

관련 인터페이스 모듈 GenICam 기능

- `CoaXPressErrorCounters` 매개 변수 카테고리
- `CxpLinkLockLossCount` 및 `CxpLinkLockLossCountReset`
- `Cxp8b10bErrorCount` 및 `Cxp8b10bErrorCountReset`
- `CxpDuplicatedCharactersUncorrectedErrorCount` 및 `CxpDuplicatedCharactersUncorrectedErrorCountReset`
- `CxpDuplicatedCharactersCorrectedErrorCount` 및 `CxpDuplicatedCharactersCorrectedErrorCountReset`
- `CxpStreamDataPacketCrcErrorCount` 및 `CxpStreamDataPacketCrcErrorCountReset`
- `CxpControlPacketCrcErrorCount` 및 `CxpControlPacketCrcErrorCountReset`
- `CxpEventPacketCrcErrorCount` 및 `CxpEventPacketCrcErrorCountReset`

3.11. CoaXPress 링크 유효성 검사 도구

소개

간단한 설명

CoAXPress 링크 유효성 검사 도구 (CXLVT)를 사용하여 CoaXPress 링크의 작동 매개 변수를 검증할 수 있습니다.

빠른 테스트를 위해 CXLVT를 실행하여 단일 비트 오류 (PER)의 확률이 10^{-10} 또는 이상인 10^{-11} 인 100% 신뢰 수준에 도달할 때까지 실행합니다. 이 작업은 단지 몇 분 정도만 소요됩니다.

광범위한 테스트를 위해 PER은 10^{-12} 또는 그 이상 10^{-13} 인 100%의 신뢰 수준에 도달할 때까지 CXLVT를 실행합니다. 이 작업은 몇 시간이 걸릴 것입니다.

참조: 비트 오류율 테스트 이론에 대한 자세한 내용은 http://en.wikipedia.org/wiki/Bit_error_rate를 참조하십시오.

호스트 PC 요구 사항

- 호스트 PC에는 적어도 하나의 EURESYS Coaxlink 보드가 장착되어 있어야 합니다.
- Coaxlink 드라이버 가 호스트 PC에 설치되어 있어야 합니다.

카메라 요구 사항

- 카메라는 정적 이미지 패턴을 생성할 수 있어야 합니다.

설치

CXLVT는 Coaxlink 드라이버와 함께 `gent1.exe`라인 도구인 에 포함되어 있습니다. 더 이상 설치할 필요가 없습니다.

gentl ber 커맨드

CXLVT는 의 명령으로 호출됩니다.berber

```

$ 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 --도움말

```

> 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 커맨드

테스트 절차

CoaxPress 링크 유효성 검사 도구를 설정하려면 다음과 같이 하십시오.

1. GenICam 브라우저과 더불어GenICam 브라우저
 - 카메라를 정상 작동 상태로 구성하고 고정된 테스트 패턴을 비디오 소스로 선택하십시오
 - 정상적인 작동을 위해 프레임 그레버를 구성하십시오
2. 명령 셸을 열고 를 실행하여 Read-Eval-Print-Lgentl ber시오
3. 명령을 사용하여 중간 결과 얻기results
 - 수집한 이미지 카운터 수가 정기적으로 증가하는지 확인하십시오
 - 신뢰 수준을 확인하십시오
4. 필요한 신뢰 수준에 도달할 때까지 테스트를 실행하십시오. 이 작업에는 몇 시간이 걸릴 수 있습니다.

작동

CoAXPress 링크 유효성 검사 도구(CXLVT)는 CoaXPress Link 설치 (비트 속도, 케이블 유형, 케이블 길이)의 작동 매개 변수를 검증하여 안정적이고 장기적인 성능을 제공합니다.

CXLVT는 알려진 신뢰성 수준으로 CoaXPress 링크 설정에서 단일 비트 오류 확률을 추정하여 이를 수행합니다.

다음을 정의합니다:

- *PER*: CoaXPress 링크와 같은 디지털 연결에서 단일 비트 오류 확률이 추정하고자 하는 미지의 양입니다.
- *BER*: CXLVT에 의해 실제로 측정된 비트 오류율

일반적으로 CoAXPress Link는 $PER < 10^{-12}$ 인 경우 안정적으로 작동합니다. 이 기준은 다른 디지털 직렬 이미지 전송 방식에서 사용되는 것과 유사합니다. 물론, 더 좋은 (더 낮은) PER은 동작이 신뢰성있다는 것을 훨씬 더 확실하게 보장합니다.

CXLVT는 필요한 증거를 축적하기에 충분한 시간 동안 PER이 BER의 측정값 기반의 값 집합 (10^{-10} , 10^{-11} , 10^{-12} , 10^{-13} , 10^{-14}),보다 작음을 나타내는 신뢰도 (CL) 또는 가능성을 계산합니다.

CXLVT가 시작되면 CXLVT는 아래의 스크린 샷에서 볼 수 있듯이 시간이 지나면서 증거가 축적됨에 따라 이러한 신뢰 수준을 표시합니다.

CXLVT 작동 중 몇 `levels`면 5 PER 값에 대한 신뢰도가 표시됩니다.

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

신뢰 수준은 시작 8 초 후에 보고됩니다.

8초 후, 우리는 PER이 10^{-10} 미만이라는 99.99% 신뢰 수준에 도달했습니다. PER은 그보다 훨씬 좋을 수도 있지만, 이 단계에서 이것이 사실이라고 결론 내릴만한 증거가 충분하지 않습니다. CXLVT는 계속되어야 합니다.

results 명령을 입력하면 CXLVT가 작동하는 동안 추가 정보가 표시될 수 있습니다. 그런 다음 CXLVT는 필요에 따라 정상 작동을 계속 수행하여 미리 결정된 PER에 대해 필요한 신뢰 수준을 얻습니다.

```
> 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%
> █
```

중간 결과는 16분 후에 보고됩니다.

이 스크린 샷에서 우리는 16분 후에 PER이 10^{-11} 미만인 신뢰도가 64.97%에서 86.28%로 상승했다고 이미 결론 지을 수 있습니다.

CXLVT는 PER이 10^{-12} 미만 (최대 10^{-13} PER보다 강한 테스트)의 신뢰도가 만족스러운 수준 (더 강한 결과를 얻으려면 적어도 95% 및 99%)에 도달할 때까지 계속되어야 합니다. 이러한 결과에는 상당한 양의 증거가 필요하기 때문에 상당한 시간이 걸릴 수 있습니다.

```
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 시간 후 중간 결과보고

이 스크린 샷에서 우리는 PER이 10^{-12} 미만인 신뢰도가 3.5 시간 후에 18.01%에서 91.66%로 상승했다고 결론 지을 수 있습니다.

보고서를 생성하려면 명령을 실행하십시오.`report`

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

보고서 커맨드 라인 생성

4. 이미지 데이터 경로

4.1. FIFO 버퍼	61
4.2. 수집 게이트	62
4.3. 픽셀 데이터 처리	63
4.4. 픽셀 데이터 처리 구성	65
4.5. 픽셀 구성 요소 언패킹	68
4.6. 픽셀 구성 요소 순서 재정렬	70
4.7. 엔디안 변환	71
4.8. 픽셀 순서	72
4.9. 이미지 데이터 전송	73
4.10. 데이터 전송률 테스트 프로그램	80

4.1. FIFO 버퍼

제품 당 DRAM 메모리 크기

제품	DRAM 메모리 크기
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 메모리는 설치된 펌웨어 변형에 따라 분할됩니다.

- 모든 펌웨어 변형은 각 장치의 각 스트림에 대해 *FIFO 버퍼*라는 하나의 파티션을 할당합니다.
- FFC를 지원하는 펌웨어 변형은 이득 및 오프셋 계수의 저장을 위해 하나의 파티션을 할당합니다

Fifo 버퍼 작동

*Fifo 버퍼*는 FIFO 처리로 CoaXPress 데이터 흐름을 픽셀 처리 및 PCI Express 데이터 흐름에서 분리합니다.

PCI Express 데이터 흐름의 일시적인 탈락을 흡수하여 신뢰할 수 있는 CoaXPress 데이터 수집을 보장합니다.

Pixel Processor 및 PCI Express 인터페이스의 한계에 관계없이 가장 높은 데이터 속도로 버스트 모드 CoaXPress 데이터 수집이 가능합니다.

4.2. 수집 게이트

수집 게이트는 온보드 FIFO 버퍼에서 이미지 데이터 추출을 제어합니다. 획득하여 "픽셀 데이터 처리" 다음 페이지에 공급할 필요가 없는 이미지 데이터를 버립니다.

영역 스캔 수집

게이트는 응용 프로그램의 `DSStartAcquisition` and `DSStopAcquisition` 함수 호출을 기반으로 프레임 경계에서 열고 닫습니다.



참고 카메라 및 조명 컨트롤러는 다양한 구성표를 사용하여 카메라 트리거를 발행하여 획득 게이팅을 간접적으로 제어합니다.

라인 스캔 수집

게이트는 애플리케이션 `DSStartAcquisition` and `DSStopAcquisition` 함수 호출에 따라 라인 경계에서 열고 닫히고 Image Acquisition Controller의 설정에 따라 스캔 시작 및 스캔 종료 트리거에 연결됩니다.

참조: 자세한 정보 및 구성 지시 사항은 "라인 스캔 수집" 페이지 232를 참조하십시오.

4.3. 픽셀 데이터 처리

이미지 픽셀 데이터 프로세서는 이미지 데이터 스트림에 대해 다음과 같은 연속 동작을 수행합니다:

CoaXPress 비트 스트림 슬라이싱

이 작업은 CoaXPress 이미지 헤더의 'PixelF' 속성에 지정된 비트 심도 - 입력 비트 심도에 따라 CoaXPress 이미지 데이터 비트 스트림에서 개별 픽셀 구성 요소 데이터를 추출합니다.

모든 구성 요소는 동일한 픽셀 비트 심도를 가집니다. 가능한 값은 8-/10-/12-/14- 및 16비트입니다.

슬라이서는 각 이미지 라인에 대해 CoaXPress 이미지 헤더의 'Xsize' 속성으로 지정된 여러 픽셀을 만드는 데 필요한 모든 픽셀 구성 요소를 제공합니다.

슬라이서는 CoaXPress 라인 패딩 데이터를 버립니다.

플랫 필드 보정

이 작업은 각 픽셀 구성 요소에 선형 이득 및 오프셋 변환을 적용합니다.

참조: 자세한 정보 및 구성 지시 사항은 "플랫 필드 보정" 페이지 175를 참조하십시오.

조회 테이블 처리

이 작업은 각 개별 픽셀 구성 요소에 찾아보기 테이블 변환을 적용합니다.

참조: 자세한 정보 및 구성 지시 사항은 "조회 테이블 처리" 페이지 189를 참조하십시오.

Bayer CFA 디코딩

이 작업은 카메라에서 발행한 원시 Bayer CFA 데이터 스트림을 RGB 색상 데이터 스트림으로 변환합니다.

참조: 자세한 정보 및 구성 지시 사항은 "바이엘 CFA 디코딩" 페이지 202를 참조하십시오.

픽셀 구성 요소 언패킹

이 작업은 10비트, 12비트 및 14비트 픽셀 구성 요소의 압축을 8비트 또는 16비트로 압축합니다.

단색 및 Bayer CFA 픽셀 형식에서는 비활성화할 수 있습니다.

참조: 자세한 정보 및 구성 지시 사항은 "픽셀 구성 요소 언패킹" 페이지 68를 참조하십시오.

픽셀 구성 요소 순서

이 조작은, 멀티 컴퍼넌트 픽셀 데이터의 컴퍼넌트 순서를 변경합니다.

참조: 자세한 정보 및 구성 지시 사항은 "픽셀 구성 요소 순서 재정렬" 페이지70를 참조하십시오.

엔디안 변환

이 연산은 16비트 픽셀 구성 요소 데이터의 바이트 순서를 수정합니다.

참조: 자세한 내용은 "엔디안 변환" 페이지71를 참조하십시오.

이미지 라인 구축

이 작업은 이미지 행의 모든 픽셀에 대한 구성 요소 데이터를 연결합니다:

- 8비트 픽셀 구성 요소는 바이트 경계에 정렬됩니다.
- 16비트 픽셀 구성 요소(언팩 또는 조희 테이블 처리로 확장 가능)가 워드(2바이트) 경계에 정렬되면 2바이트가 리틀 엔디안 규칙에 따라 저장됩니다.

라인 패딩

이 조작은 패딩 비트 또는 바이트를 이미지 라인 데이터에 추가하여 하드웨어 구현에 필요한 다음 정렬 경계에 도달합니다.

정렬 경계 요구 사항은 제품별입니다. 예를 들면 다음과 같습니다:

- **1630 Coaxlink Mono** , **1631 Coaxlink Duo** , 및 **1632 Coaxlink Quad** 용 64비트
- **1633 Coaxlink Quad G3** , **1633-LH Coaxlink Quad G3 LH** 및 **1635 Coaxlink Quad G3 DF** 용 128비트

처리 성능

픽셀 프로세서는 가장 높은 카메라 픽셀 속도를 유지합니다. 달리 명시되어 있지 않는 한, 지연 시간을 무시하고 GenTL로 데이터를 전송하는 동안 위의 모든 작업이 실행됩니다.

PCI Express 대역폭 제한

픽셀 비트 심도가 8비트보다 큰 픽셀을 획득할 때, 각 픽셀은 16비트로 확장됩니다. 이러한 경우 호스트 PC의 PCI Express 대역폭 제한이 달성 가능한 프레임 또는 회선 속도에 부정적인 영향을 미칠 수 있습니다.

온보드 메모리 대역폭 제한

FFC 사용 사례의 경우, 온보드 메모리 대역폭만으로는 전체 CoAXpress 데이터 속도를 유지할 수 없습니다.

4.4. 픽셀 데이터 처리 구성

이 항목에서는 모든 클래스의 카메라 픽셀 형식에 적용 가능한 픽셀 데이터 처리 구성을 나열합니다

단색 픽셀 형식 - LUT 비활성화 됨

카메라 픽셀 형식	FFC	Unpacking Mode	출력 픽셀 형식
Mono8	off on	lsb msb	Mono8
Mono10pmsb	off on	lsb	Mono10
	off on	msb	Mono16
	off	off	Mono10pmsb
Mono12pmsb	off on	lsb	Mono12
	off on	msb	Mono16
	off	off	Mono12pmsb
Mono14pmsb	off on	lsb	Mono14
	off on	msb	Mono16
	off	off	Mono14pmsb
Mono16	off on	lsb msb	Mono16

단색 픽셀 형식 - LUT 활성화됨

카메라 픽셀 형식	FFC	조희 테이블 구성	언패킹 모드	출력 픽셀 형식
Mono8	off on	M_8x8	lsb msb	Mono8
Mono10pmsb	off on	M_10x8	lsb msb	Mono8
		M_10x10	lsb	Mono10
			msb	Mono16
		M_10x16	lsb msb	Mono16
Mono12pmsb	off on	M_12x8	lsb msb	Mono8
		M_12x12	lsb	Mono12
			msb	Mono16
		M_12x16	lsb msb	Mono16

Bayer CFA 픽셀 형식 - CFA 디코더 사용 불가

Camera Pixel Format	FFC	Unpacking Mode	Output Pixel Format
Bayer**8	off on	lsb msb	Bayer**8
Bayer**10pmsb	off on	lsb	Bayer**10
	off on	msb	Bayer**16
	off	off	Bayer**10pmsb
Bayer**12pmsb	off on	lsb	Bayer**12
	off on	msb	Bayer**16
	off	off	Bayer**12pmsb
Bayer**14pmsb	off on	lsb	Bayer**14
	off on	msb	Bayer**16
	off	off	Bayer**14pmsb
Bayer**16	off on	lsb msb	Bayer**16

Bayer CFA 픽셀 형식 - CFA 디코더 활성화됨

Input Pixel Format	FFC	RedBlueSwap	Output Pixel Format
Bayer**8	off on	off	RGB8
		on	BGR8
Bayer**10pmsb	off on	off	RGB10
		on	BGR10
Bayer**12pmsb	off on	off	RGB12
		on	BGR12
Bayer**14pmsb	off on	off	RGB14
		on	BGR14
Bayer16	off on	off	RGB16
		on	BGR16

RGB 픽셀 형식

Input Pixel Format	FFC	Unpacking Mode	RedBlueSwap	Output 픽셀 형식
RGB8	off on	lsb msb	off	RGB8
			on	BGR8
RGB10pmsb	off on	lsb	off	RGB10
			on	BGR10
		msb	off	RGB16
			on	BGR16
RGB12pmsb	off on	lsb	off	RGB12
			on	BGR12
		msb	off	RGB16
			on	BGR16
RGB14pmsb	off on	lsb	off	RGB14
			on	BGR14
		msb	off	RGB16
			on	BGR16
RGB16	off on	lsb msb	off	RGB16
			on	BGR16

RGBa 픽셀 형식

입력 픽셀 형식	FFC	UnpackingMode	RedBlueSwap	출력 픽셀 형식
RGBa8	off on	lsb msb	off on	RGBa8
RGBa10pmsb	off on	lsb	off on	RGBa10
		msb	off on	RGBa16
RGBa12pmsb	off on	lsb	off on	RGBa12
		msb	off on	RGBa16
RGBa14pmsb	off on	lsb	off on	RGBa14
		msb	off on	RGBa16
RGBa16	off on	lsb msb	off on	RGBa16

4.5. 픽셀 구성 요소 언패킹

소개

픽셀 데이터 프로세서는 10비트, 12비트 및 14비트 픽셀 구성 요소 데이터를 16비트 픽셀 데이터로 언패킹할 수 있습니다.

언패킹 작업은 **UnpackingMode** GenICam 기능을 통해 사용자가 구성할 수 있습니다. 세 가지 옵션을 사용할 수 있습니다.

- **Lsb**: lsb로 언패킹(4.3 이후의 기본 설정)
- **Msb**: msb로 언패킹
- **Off**: 언패킹 불가

lsb로 언패킹

픽셀 구성 요소 데이터의 유효 비트는 데이터 컨테이너의 **최하위 비트**에 정렬됩니다. 패딩 '0'비트는 필요에 따라 **최상위 비트**에 배치되어 다음 8 비트 경계에 도달합니다.

- 10비트 픽셀: 0000 00<pp pppp pppp>
- 12비트 픽셀: 0000 <pppp pppp pppp>
- 14비트 픽셀: 00<pp pppp pppp pppp>



참고 lsb에 압축을 풀면 픽셀 구성 요소 값이 수정되지 않습니다.

msb에 언패킹

픽셀 구성 요소 데이터의 유효 비트는 데이터 컨테이너의 **최상위 비트**에 정렬됩니다. 패딩 '0'비트는 필요에 따라 **최하위 비트**에 배치되어 다음 8 비트 경계에 도달합니다.

- 10비트 픽셀: <pppp pppp pp>00 0000
- 12비트 픽셀: <pppp pppp pppp> 0000
- 14비트 픽셀: <pppp pppp pppp pp>00



팁msb에 대한 10비트, 12비트 및 14비트 픽셀 구성 요소의 언패킹은 픽셀 구성 요소 값에 각각 64, 16 및 4를 곱합니다.



노트 8비트 및 16비트 픽셀 구성 요소의 언팩은 독립적인 작업입니다:

- 데이터 컨테이너의 크기는 변경되지 않습니다. 8비트 픽셀 구성 요소의 경우 1바이트. 16비트 픽셀 구성 요소의 경우 2바이트
- 데이터 비트는 수정되지 않습니다.



경고10비트, 12비트 및 14비트 픽셀 구성 요소의 압축을 풀면 데이터 양이 각각 160%, 133% 및 114% 증가합니다!

언패킹 불가

CoaXPress Link를 통해 카메라로 전송된 압축된 이미지 데이터는 그대로 출력 버퍼로 전달됩니다.



경고이 옵션은 단색 및 Bayer CFA 픽셀을 제공하는 카메라의 영역 스캔 펌웨어 변형에서만 사용할 수 있습니다!

4.6. 픽셀 구성 요소 순서 재정렬

동축 링크 이미지 데이터 스트림 픽셀 프로세서는 3-구성 요소 픽셀의 제1 및 제3 구성 요소 데이터를 교환하도록 구성될 수 있습니다.

스와핑은 **RedBlueSwap** 불린 GenICam 기능을 통해 제어됩니다:

- **False**(기본 설정)로 설정하면 원래 구성 요소 순서가 유지됩니다.
- **True**로 설정하면 첫 번째와 세 번째 구성 요소가 서로 바뀝니다.

이 기능은 다음 위치에서 이미지를 수집 할 수 있습니다:

- 3-구성 요소 픽셀 데이터를 전달하는 RGB 컬러 카메라,
- BAYER CFA 디코딩이 가능하도록 제공하는 BAYER CFA 컬러 카메라.

4.7. 엔디안 변환

Coaxlink 이미지 데이터 스트림 픽셀 프로세서는 리틀 엔디안 방식을 사용하여 16비트 픽셀 구성 요소를 제공합니다.

UnpackingMode가 **Off**로 설정된 경우 변환이 수행되지 않습니다.

리틀 엔디안 변환

다중 바이트 데이터의 최하위 바이트는 최하위 주소 위치에 저장됩니다.

예를 들어, 16비트 데이터는 다음과 같이 두 개의 연속적인 바이트 위치에 저장됩니다:

메모리 바이트 위치	메모리 바이트 내용
N	Data[7:0]
N+1	Data[15:8]

4.8. 픽셀 순서

Coaxlink 이미지 데이터 스트림 픽셀 프로세서는 CoaXPress 데이터 스트림의 픽셀 순서를 보존합니다.

이미지 프레임의 픽셀 데이터는 최저 어드레스에서 제 1라인의 제 1픽셀로 시작하는 연속적인 어드레스 위치에 저장됩니다.

이미지 프레임의 연속 라인은 이미지 버퍼에서 연결됩니다.

4.9. 이미지 데이터 전송

버퍼 채우기

DMA 엔진은 라인 스캔 및 영역 스캔 이미지 수집과 다른 규칙에 따라 PCI Express 버스를 통해 처리된 이미지 데이터를 할당된 GenTL 버퍼로 전송합니다.

GenTL 버퍼 채우기 규칙 - 영역 스캔 카메라

영역 스캔 이미징에서 GenTL 버퍼는 다음 규칙에 따라 채워집니다.

- 스캔의 첫 번째 수집된 라인 데이터는 기본적으로 새 버퍼의 시작 부분에 저장됩니다. **StripeArrangement**를 로 설정하여 세로 이미지 반전을 사용하면 프레임의 첫 번째 수집된 라인 데이터가 새 버퍼의 마지막 **Geometry_1X_1YE**막 전체 라인 위치에 저장됩니다.
- 호스트 메모리로의 이미지 전송이 완료되면 부분적으로 채워질 수 있는 버퍼가 처리를 위해 응용 프로그램에서 사용 가능하게 됩니다.
- 버퍼가 너무 작아서 완전한 프레임을 포함 할 수 없는 경우, 나머지 데이터는 버려집니다.

GenTL 버퍼 채우기 규칙 - 라인 스캔 카메라

라인 스캔 이미징에서 GenTL 버퍼는 다음 규칙에 따라 채워집니다.

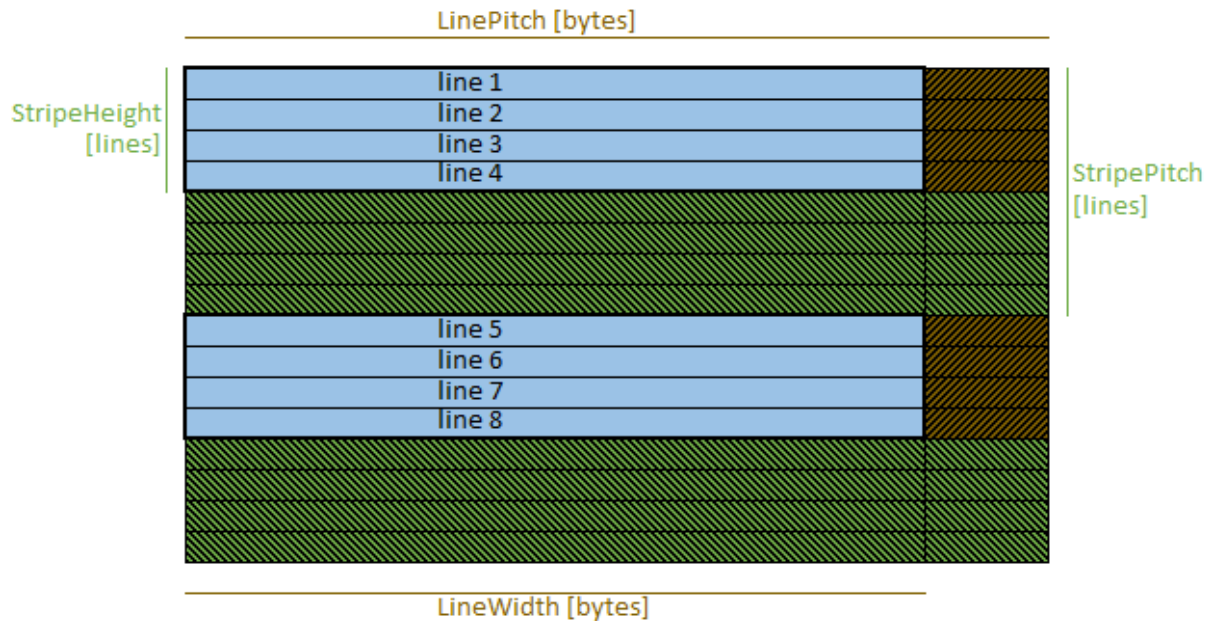
- 스캔의 첫 번째 수집된 라인 데이터는 기본적으로 새 버퍼의 시작 부분에 저장됩니다. **StripeArrangement**를 **Geometry_1X_1YE**로 설정하여 세로 이미지 반전을 사용하면 스캔의 첫 번째 수집된 라인 데이터가 새 버퍼의 마지막 전체 라인 위치에 저장됩니다.
- 버퍼는 정수 개의 이미지 라인 데이터를 포함합니다.
- 버퍼의 나머지 공간이 이미지 라인 데이터를 저장하기에 충분하지 않을 때, 획득은 새로운 버퍼로 계속되고, 채워진 버퍼는 처리를 위해 애플리케이션에 이용 가능해집니다.
- 스캔의 마지막 라인 데이터가 수집되면, 부분적으로 채워질 수 있는 마지막 버퍼가 처리를 위해 애플리케이션에서 사용 가능하게 됩니다.

이미지 데이터 패딩

DMA 엔진은 라인 패딩 또는 스트라이프 패딩을 추가하여 버퍼에서 데이터를 다르게 구성할 수 있는 기능을 제공합니다.



참고 Coaxlink 드라이버 6.2 이전에 DMA 엔진은 2D 구조와 관계없이 전체 이미지 데이터를 단일 1D 엔티티로 전송했습니다. 처리된 이미지 데이터 라인이 대상 버퍼로 연결됩니다.



이미지 버퍼 패딩 모델

라인 패딩

LineWidth 및 **LinePitch** 기능은 라인 패딩을 제어합니다.

LinePitch > LineWidth를 사용하면 라인 패딩을 사용할 수 있습니다: DMA 엔진은 각 이미지 라인의 끝에 패딩의 **LinePitch - LineWidth** 바이트를 남겨 둡니다.

LinePitch는 0로 설정하여 라인 다음에 패딩을 사용 불가능하게 할 수 있습니다.

스트라이프 패딩

스트라이프는 인접한 라인의 그룹입니다. 높이 1의 스트라이프가 라인입니다.

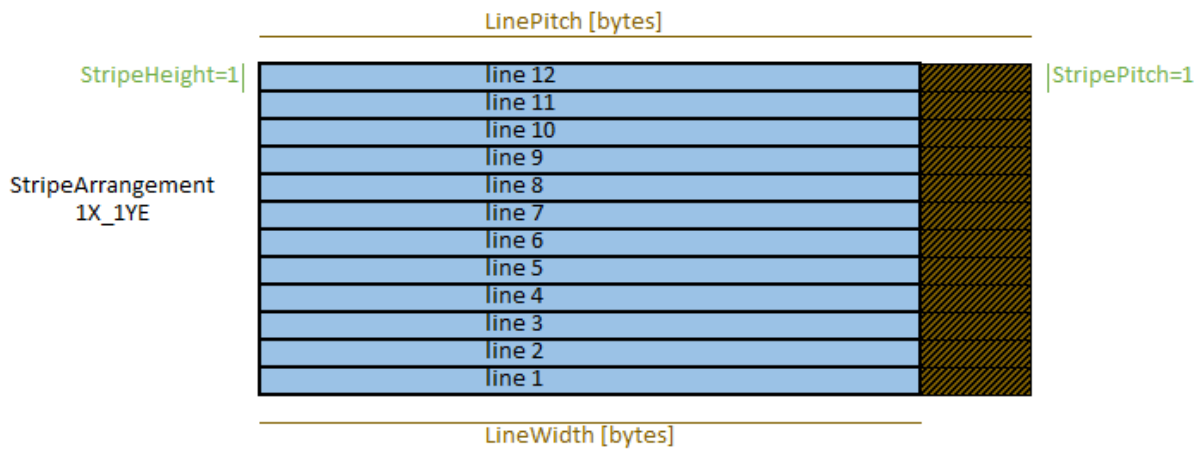
StripeHeight 및 **StripePitch** 기능은 스트라이프 패딩을 제어합니다.

StripePitch > StripeHeight를 사용하면 스트라이프 패딩을 사용할 수 있습니다: DMA 엔진은 각 스트라이프 라인의 끝에 패딩의 **StripePitch - StripeHeight** 라인을 남겨 둡니다.

StripePitch는 0로 설정하여 라인 다음에 패딩 사용을 불가능하게 할 수 있습니다.

세로 이미지 뒤집기

DMA 엔진은 이미지를 수직으로 뒤집을 수 있는 기능을 제공합니다.



뒤집힌 이미지 데이터

수직 이미지 플립은 데이터 스트림 모듈의 **StripeArrangement** 기능에 의해 제어됩니다.

기본적으로 **StripeArrangement**는 **1X_1Y**로 설정됩니다: 수직 이미지 뒤집기가 비활성화됩니다.

StripeArrangement가 **1X_1YE**로 설정되면 드라이버는 다음 수식을 사용하여 버퍼에서 첫 번째 이미지 라인의 위치를 결정합니다.

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

그 결과:

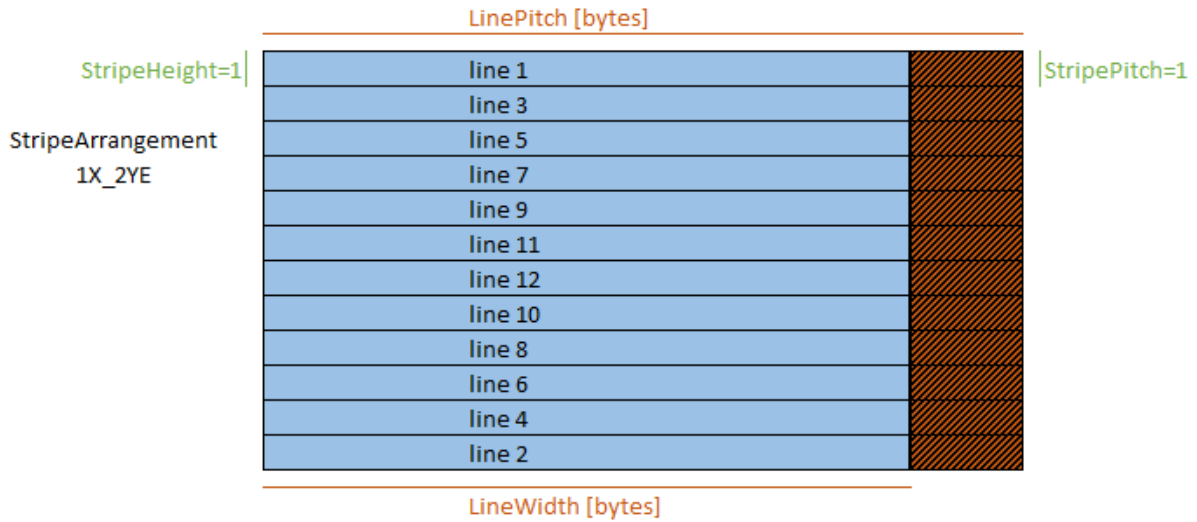
- 버퍼가 너무 작으면, 그것은 (카메라에 의해 주어진) 잃어버린 이미지의 바닥 부분입니다;
- 라인은 $\text{BufferBase} + n * \text{LinePitch}$ 에서 시작할 것입니다.
- 완전한 라인만 전송됩니다;
- 버퍼 크기가 **LinePitch** 바이트의 배수가 아닌 경우 버퍼 끝에 있는 일부 바이트는 변경되지 않습니다.



참고로 공식을 계산할 때 **LinePitch**가 0과 같으면 **LineWidth**가 대신 사용됩니다. 마찬가지로 **StripeHeight**가 0이면 1이 대신 사용됩니다.

1X_2YE 이미지의 스크램블 해제

DMA 엔진은 1X_2YE 이미지를 언 스크램블하는 기능을 제공합니다.



StripeArrangement가 로 설정되면, 드라이버는 **Geometry_1X_2YE** 다음 수식을 사용하여 카메라가 출력하는 두 번째 라인의 목적지(즉, 버퍼의 마지막 이미지 라인의 위치)를 결정합니다.

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



참고 이것은 하나의 완전한 라인을 수신할 만큼 충분히 큰 버퍼의 마지막 라인의 주소입니다.

그 결과:

- 버퍼가 너무 작으면, 카메라에 의해 출력된 최종 라인(즉, 이미지의 중간 부분)은 손실될 것입니다; 응용 프로그램이 이를 방지하도록 합니다.
- 라인은

$$\text{BufferBase} + n * \text{LinePitch},$$

- 전체 라인만 전송됩니다. (위 공식을 계산할 때 **LinePitch**가 0와 같으면 **LineWidth**가 대신 사용됩니다. 마찬가지로 **StripeHeight**가 0인 경우 1가 대신 사용됩니다.)
- **StripeHeight** 및 **StripePitch**보다 큰 값으로 설정할 수 없습니다.¹

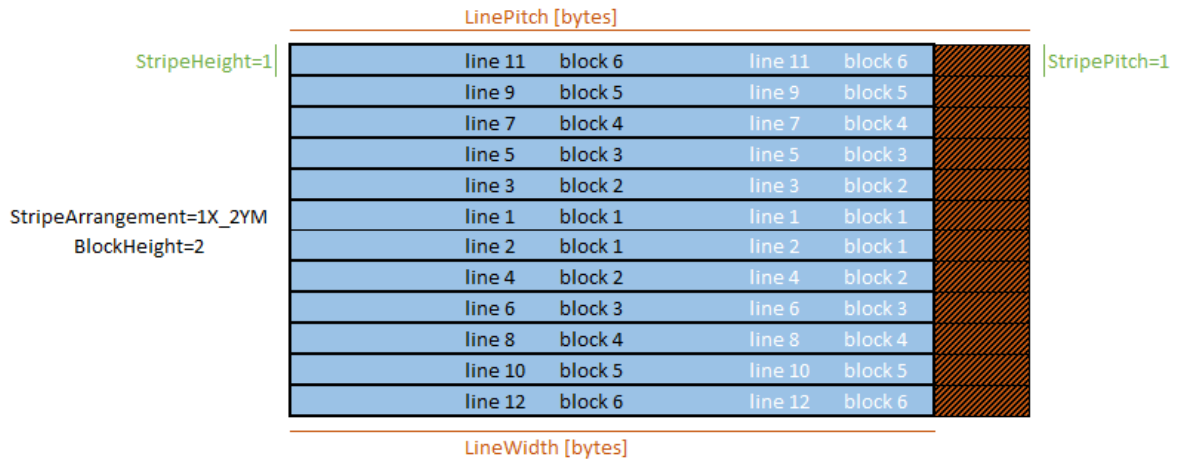
1X_2YM 이미지의 스크램블 해제

DMA 엔진은 1X_2YM 이미지를 언 스크램블하는 기능을 제공합니다.

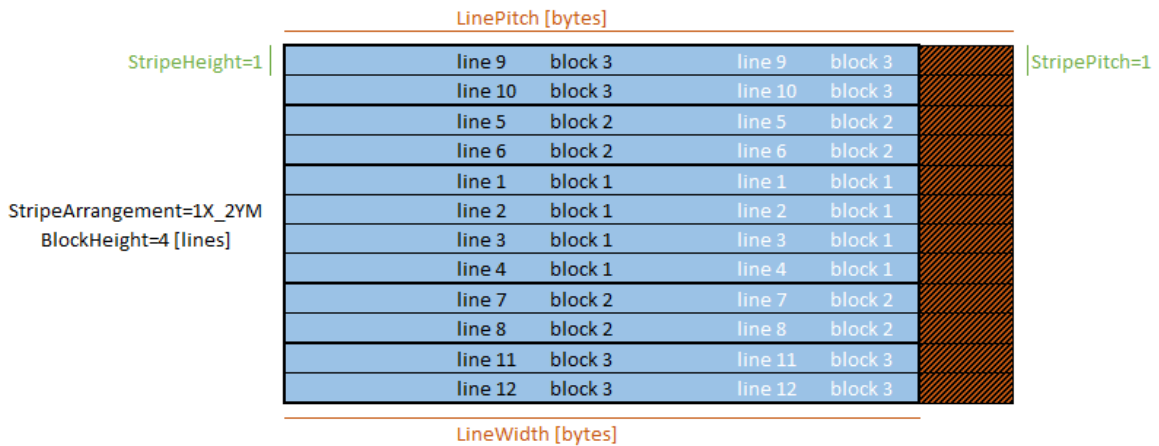


참고다음 그림에서 장치가 보낸 라인 번호와 블록 번호는 **흰색**이고 호스트가 수신한 라인 번호와 블록 번호는 **검정색**입니다.

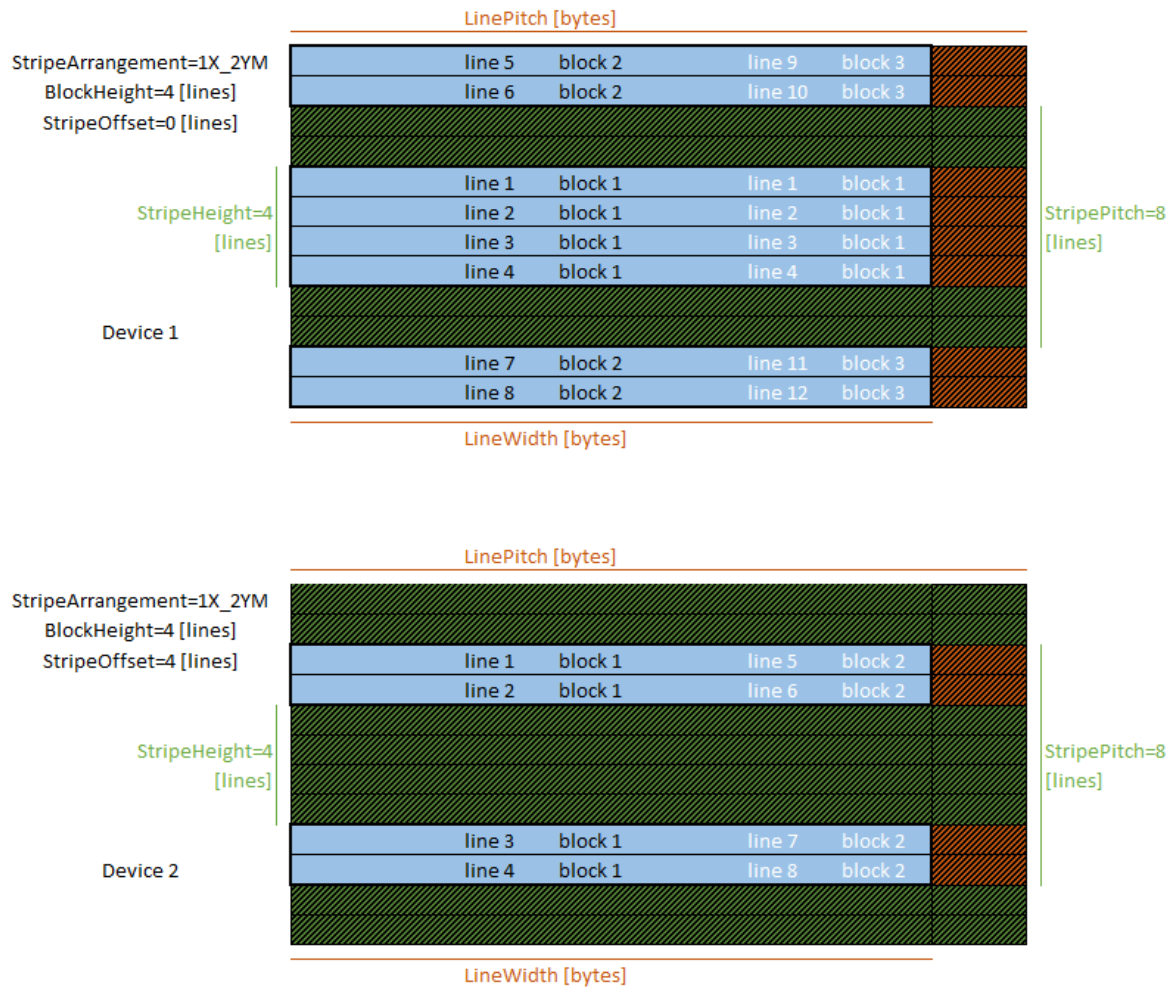
1X_2YM 카메라는 2 블록 단위로 라인을 전송합니다.



1X_2YM 카메라는 4 블록 단위로 라인을 전송합니다.



1X_2YM 카메라는 4 블록 단위로 선 두 호스트에 라인을 전송합니다.



통계

스트림 통계 도구는 카드 출력에서 이미지 데이터 스트림을 모니터링하고 평균 프레임, 라인 및 데이터 속도를 응용 프로그램에 제공합니다.

스트림 통계 샘플링 방법

`StatisticsSamplingSelector`는 평균 간격을 결정합니다. 다음 중 하나일 수 있습니다:

- `LastSecond` 또는 `LastTenSeconds`: 마지막으로 완료된 시간 슬롯은 1초 또는 10초입니다.
- `Last2Buffers`, `Last10Buffers`, `Last100Buffers`, `Last1000Buffers`: 마지막으로 2, 10, 100 또는 1000 개의 획득한 버퍼
- `LastAcquisition`: 마지막 획득 활동 기간. 즉 마지막 `DSStartAcquisition()` 함수가 지금까지 호출되었기 때문에 획득이 여전히 활성화되어 있지 않으면 마지막 `DStopAcquisition()` 함수 호출까지 호출됩니다.
- `LastAcquisition`:와 명령 사이의 시간 간격. `StatisticsStartSamplingStatisticsStopSampling`

기본 샘플링 방법은 `LastSecond`입니다.

통계 데이터

통계 데이터는 다음 기능 중 하나를 얻을 때 효과적으로 계산됩니다.

- `StatisticsFrameRate`는 프레임/초 단위로 표시된 평균 프레임 속도를 보고합니다 (영역 스캔).
- `StatisticsLineRate`는 프레임/초 단위로 표시된 평균 프레임 속도를 보고합니다 (라인 스캔).
- `StatisticsDataRate`는 메가 바이트/초로 표시되는 평균 데이터 속도를 보고합니다.

평균화 간격 동안 채워진 모든 GenTL 버퍼에 대해 도구는 다음을 계산합니다:

- 채워진 GenTL 버퍼의 수와 해당 프레임 수(영역 스캔) 또는 라인(라인 스캔)
- 이미지 데이터의 전송 바이트 수입니다.

관련 GenICam 기능은 GenTL 데이터 스트림 모듈의 스트림 통계 범주에 모여서 희소합니다.

4.10. 데이터 전송률 테스트 프로그램

소개

DTR(데이터 전송률 테스트 프로그램)을 사용하여 실제 조건에서 효과적인 PCI Express 데이터 전송 속도를 측정할 수 있습니다.

호스트 PC 요구 사항

- 호스트 PC에는 적어도 하나의 EURESYS Coaxlink 보드가 장착되어 있어야 합니다.
- Coaxlink 드라이버 버전 12.4 이상이 호스트 PC에 설치되어 있어야 합니다.

카메라 요구 사항

- 이미지 데이터를 지속적으로 전달하도록 카메라를 구성해야 합니다.

설치

DTR은 Coaxlink 드라이버와 함께 제공되는 커맨드 라인 도구인 `gent1.exe`에 포함되어 있습니다. 더 이상 설치할 필요가 없습니다.

측정 원리

DTR은 내부 프레임 저장소를 완전히 채우고 이미지를 호스트 컴퓨터로 전송하여 데이터 전송 속도를 측정합니다.

1. 모든 버퍼가 큐에서 제외됩니다(데이터 스트림에서 사용할 수 없음)
2. 데이터 스트림과 원격 장치가 시작되었습니다
3. 프레임 저장소가 가득 차면 원격 장치가 중지됩니다
4. 현재 타임 스탬프가 검색됩니다(t_0)
5. 모든 버퍼는 데이터 스트림에 대기하고 전송 시작합니다
6. 데이터 스트림에서 버퍼가 나옵니다
7. 프레임 저장소가 비어 있고 모든 버퍼가 검색되면 데이터 스트림이 중지됩니다.

DTR 프로그램은 다음과 같이 `data transfer rate`를 계산합니다.

```
-바이트 수 = 각 버퍼 BUFFER_INFO_SIZE_FILLED의 합
-t1 = 마지막 버퍼의 BUFFER_INFO_CUSTOM_EVENT_TIMESTAMP
-지속 시간 = t1-t0
-데이터 전송률 = 바이트 수/지속 시간
```


gentl --도움말

```

GenTL Explorer

gentl [COMMAND] ... [OPTIONS]

Commands:
  info      Show detailed information about the transport layer system
  report    Generate a GenTL report archive (for Euresys tech support)
  xml       Download GenApi files (XML register descriptions)
  play      Open a data stream and acquire images (no display)
  view      Open a data stream and display images
  grab      Grab N images
  genapi    Enter the GenApi command-line interface or perform a GenApi operation
  read      Read data from a GenTL port
  write     Write data to a GenTL port
  event     Wait for events and display information about them
  script    Execute script
  run       Run an action
  dtr      Measure PCIe data transfer rate
  ber       Measure bit error rate confidence level (a.k.a. link validation tool)

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 --도움말

```

GenTL Explorer

gentl dtr [OPTIONS]

Flags:
  --if=ID           Interface ID
  --dev=ID          Device ID
  --ds=ID           Data stream ID
  --device-access=ACCESS Access flags used to open the device (GenTL standard access
flags:
  DEVICE_ACCESS_READONLY, DEVICE_ACCESS_CONTROL, DEVICE_
ACCESS_EXCLUSIVE;
  Coaxlink custom access flags:
  DEVICE_ACCESS_CUSTOM_READONLY_DEVICE_READONLY_STREAM)
(default:
  DEVICE_ACCESS_CONTROL)
  --ro              Open the device as read-only (shorthand for
  --device-access=DEVICE_ACCESS_READONLY)
  --buffers=INT    Buffer count (default: 4)
  --buffer-size=INT Buffer size
  --width=WIDTH    Buffer width
  --height=HEIGHT  Buffer height
  --pixel-format=ITEM PFNC Pixel format
  --bayer=BAYERDECODINGMETHOD Bayer method (Legacy, Advanced) (default: Advanced)
  --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
  --timeout=INT    Acquisition timeout, in milliseconds (default: infinite)
  --zero          Zero memory when queuing buffers (default: memory is only
zeroed when buffers
  are allocated)
  --remotexml=FILE Use FILE as register description (default: register
description is read from
  remote device)
  -n --repeat[=N] Measure data transfer rate N times (default: 1)
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

```



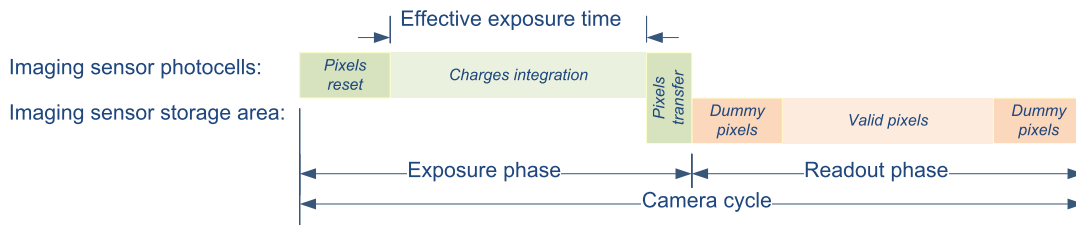
팁 측정 정확도를 높이려면 `gentl ber -n` 옵션을 사용하여 여러 측정을 반복적으로 실행합니다. DTR 프로그램은 결과를 평균화합니다.

5. 카메라 조절 원리

5.1. 카메라 주기	84
5.2. 카메라 주기 연결 규칙	85
5.3. 카메라 조절 방법	87

5.1. 카메라 주기

카메라 주기는 노출 단계와 판독 두 단계로 구성됩니다.



카메라 주기

노출 단계

노출 단계는 이미지 센서의 광전지가 들어오는 광자에 의해 유도된 전하를 통합하는 기간입니다.

전자 셔터가 있는 카메라의 경우, 모든 센서 광전지를 지우는 픽셀 리셋 동작으로 노출 단계가 시작됩니다. 영구 노광 카메라의 경우, 즉 전자 셔터가 없는 (또는 사용하지 않는) 카메라의 경우, 노출 단계는 이전 노출 단계의 완료 직후에 시작됩니다.

모든 유형의 카메라에서 노출 단계는 "픽셀 전송" 동작으로 종료됩니다. 포토 셀의 축적된 전하는 추가 판독을 위해 저장 영역으로 전송됩니다. 이 작업으로 포토 셀이 지워지고 새로운 전하 통합이 즉시 시작됩니다.

전자 셔터가 있는 카메라는 비동기적으로 픽셀을 리셋하고 요청에 따라 새로운 노출을 개시하는 능력을 갖습니다. 이 카메라는 비동기 리셋 카메라로 명명됩니다.

노광 개시 시간(픽셀 리셋) 및 노광 종료 시간(픽셀 전송)을 제어하는 능력을 갖는 것은:

- 각 이미지 캡처의 타이밍
- 노출 시간을 선택하여 이미지 센서의 민감도

판독 단계

판독 단계는 각 픽셀에 의해 축적된 총 전하량이 측정되고 이미징 센서 출력으로 전달되는 시간에 대한 기간입니다.

판독 단계는 프레임 그래버에 의해 조절되지 않습니다:

- 지속 시간은 고정되어 있습니다.
- 이는 전송될 픽셀 데이터의 양 및 센서의 판독 구조 (하나 이상의 탭, 탭 출력 데이터 속도)에 의해 결정됩니다.

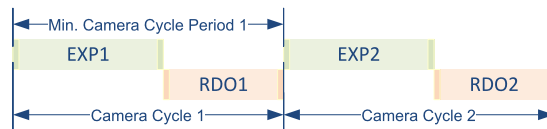
일부 센서는 전송해야 할 데이터가 적기 때문에 판독 속도를 높이는 하나 이상의 관심 영역 (ROI)을 선택할 수 있는 기능을 제공합니다.

5.2. 카메라 주기 연결 규칙

이 항목에서는 비동기 재설정 카메라에 연속 카메라 사이클을 요청할 때 *카메라 트리거* 오버런을 피하기 위해 프레임 그래버에서 반드시 준수해야하는 규칙에 대해 설명합니다.

중복을 허용하지 않는 카메라에 대한 규칙

- 다음 카메라 사이클은 판독 단계가 완료되기 전에 시작할 수 없습니다.



사이클 오버랩을 허용하지 않는 카메라에 의해 가능한 최단 사이클주기

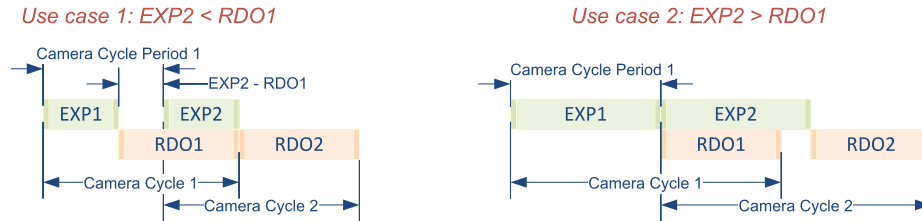
$$\text{Min Cycle Period}_n = \text{EXP}_n + \text{RDO}_n$$



참고산업 카메라 중 소수만 이 주기 중복을 허용하지 않습니다!

중복을 허용하는 카메라에 대한 규칙

1. 두 개의 연속적인 카메라 주기의 노출 단계가 겹치지 않을 수도 있습니다.
2. 2개의 연속적인 카메라 주기의 판독 위상은 결코 중첩되지 않을 수 있습니다



주기 오버랩을 허용하는 카메라에 의해 달성 가능한 최단 주기

첫 번째 경우에는 두 번째 사이클의 노출 단계의 지속 시간이 첫 번째 사이클의 판독 단계의 지속 시간보다 짧습니다. 다음 카메라 사이클은 노출 단계가 완료된 후 ($EXP_{n+1} - RDO_n$) 동안 시작될 수 있습니다. 최소 주기는

$$Min\ Cycle\ Period_n = EXP_n + RDO_n - EXP_{n+1}$$

두 번째 경우에서, 제 2 주기의 노출 단계의 지속 기간은 제 1 주기의 판독 단계의 지속 기간보다 길어집니다. 다음 카메라 주기는 노출 단계 완료 직후에 시작될 수 있습니다. 최소 주기는:

$$Min\ Cycle\ Period_n = EXP_n$$



참고머신 비전 업계에서 사용되는 대부분의 비동기식 리셋 카메라는 카메라 사이클의 중첩을 지원합니다!

5.3. 카메라 조절 방법

Coaxlink 프레임 그래버는 **NC**, **RC**, **RG** 및 **EXTERNAL**이라는 네 가지 카메라 조절 방법을 제공합니다.

NC 카메라 조절 방법

NC 카메라 조절 방법은 프레임 그래버에 의해 조절되지 않는 카메라를 대상으로 합니다. 여기에서는 다음을 포함합니다

- 외부 트리거 신호를 사용하지 않는 프리 런 카메라
- 외부 트리거 신호를 사용하는 비동기식 재설정 카메라는 프레임 그래버가 제공하지 않습니다.

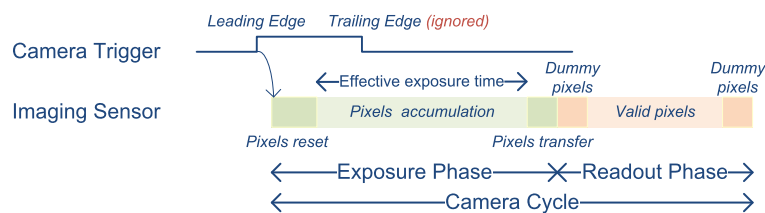


참고 Coaxlink의 카메라 및 조명 컨트롤러 (CIC)는 사용되지 않습니다. 프레임 그래버에 의한 카메라의 실시간 조절은 없습니다. Coaxlink에서 생성된 카메라 트리거 신호는 없습니다. 조명의 실시간 조절은 없습니다. Coaxlink에서 생성된 스트로브 신호는 없습니다.

RC 카메라 조절 방법

RC 카메라 조절 방법은 카메라주기 속도만 프레임 그래버에 의해 제어되는 비동기식 재설정 카메라를 대상으로 합니다. 노출 지속 시간은 카메라에 의해 제어됩니다.

실시간 조절은 Coaxlink의 카메라 및 illumination 컨트롤러 (CIC)에서 발행한 “카메라 트리거”라는 단일 업스트림 신호를 통해 수행됩니다.



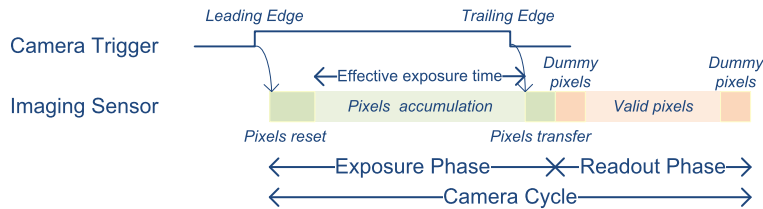
RC 카메라 조절 방법을 사용하는 그래버 조절 카메라 사이클

CIC는 매 카메라 사이클마다 하나의 카메라 트리거 펄스를 생성합니다. 카메라 트리거 리딩 에지는 새로운 카메라 사이클을 트리거하고 새로운 노출 시간을 시작합니다. 카메라 트리거 트레일링 에지는 카메라에서 무시됩니다.

RG 카메라 조절 방법

RG 카메라 조절 방법은 비동기식 재설정 카메라를 대상으로 하며 카메라주기 속도와 노출 지속 시간은 프레임 그래버에 의해 조절됩니다.

실시간 조절은 Coaxlink의 카메라 및 illumination 컨트롤러 (CIC)에서 발행한 *카메라 트리거* 라는 단일 업스트림 신호를 통해 수행됩니다.



RG 카메라 조절 방법을 사용하는 그래버 조절 카메라 사이클

CIC는 매 카메라 사이클마다 하나의 카메라 트리거 펄스를 생성합니다. *카메라 트리거* 리딩 에지는 새로운 카메라 사이클을 트리거하고 새로운 노출 시간을 시작합니다. *카메라 트리거* 트레일링 에지는 노출 시간을 종료하고 판독을 트리거합니다.

외부 카메라 조절 방법

외부 카메라 조절 방법은 프레임 그래버에 의해 조절되지 않는 카메라를 대상으로 합니다. 여기에서는 다음을 포함합니다

- 외부 트리거 신호를 사용하지 않는 프리 런 카메라
- 외부 트리거 신호를 사용하는 비동기식 재설정 카메라는 프레임 그래버가 제공하지 않습니다.

외부 카메라 조절 방법은 외부 컨트롤러에 의해 그래버의 모든 GPIO 입력 포트에 적용되는 하드웨어 신호에 의해 조절되는 비동기 리셋 카메라를 목표로 합니다.



참고 Coaxlink의 카메라 및 조명 컨트롤러 (CIC)는 사용되지 않습니다. 프레임 그래버에 의한 카메라의 실시간 조절은 없습니다. Coaxlink에서 생성된 카메라 트리거 신호는 없습니다. 조명의 실시간 조절은 없습니다. Coaxlink에서 생성된 스트로브 신호는 없습니다.



경고 외부 컨트롤러는 카메라주기 타이밍 전체를 담당합니다!

6. 조명 제어 원리

6.1. 조명 장치	90
6.2. 카메라와 조명주기 맞추기	91

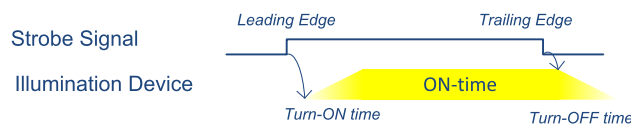
6.1. 조명 장치

2개의 부류의 조명 장치는 조명 제어기에 의해 제어될 수 있습니다:

- 간헐 조명 장치
- 스트로보 조명 장치

간헐 조명 장치

이 조명 장치 부류는 턴온 및 턴 오프 시간이 스트로브 신호의 상승 및 하강 에지에 의해 제어되는 스위치 형 광원을 포함합니다.



간헐 조명 장치의 타이밍 다이어그램

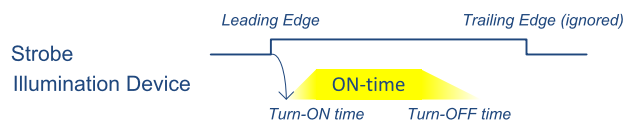
스트로브 펄스의 폭은 광원의 ON 지속 시간을 결정합니다



참고조명 컨트롤러를 구성할 때 켜기 시간과 꺼짐 시간을 고려해야 합니다!

스트로보 조명 장치

이 조명 장치 부류는 턴온 시간만이 스트로브 신호의 리딩 에지에 의해 제어되는 스위치 형 광원을 포함합니다.



스트로보 조명 장치의 타이밍 다이어그램

정시 지속 시간은 제어되지 않거나 또는 조명 장치 자체에 의해 제어됩니다.

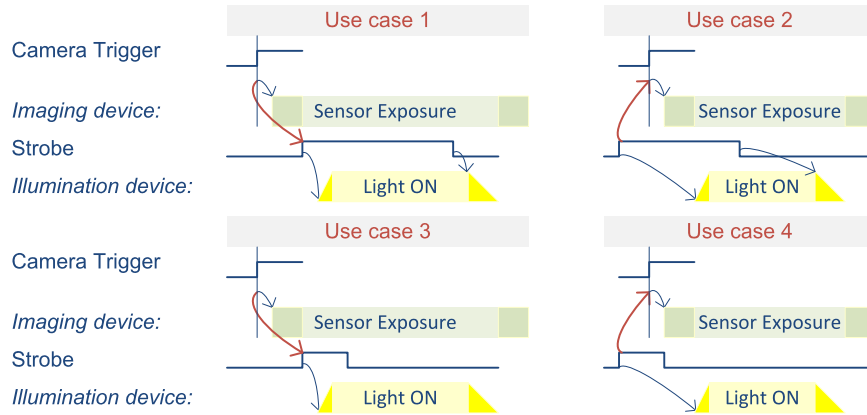


참고조명 제어를 구성할 때는 켜기 시간과 ON 지속 시간을 고려해야 합니다.

6.2. 카메라와 조명주기 맞추기

물론, 광원의 ON 시간은 이미징 센서의 노출 단계와 일치해야 합니다.

따라서 스트로브 신호와 Camera Trigger 신호 간의 시간 관계를 적절히 제어해야 합니다.



카메라 트리거 대 스트로브 정렬의 일반적인 사용 사례

간헐 광원 (사용 사례 1 및 2)

적절한 양의 빛을 제공하고 올바르게 노출된 이미지를 얻으려면 스트로브 펄스의 지속 시간을 적절하게 제어해야 합니다.

조명이 꺼진 후 센서 노출을 종료하려면 센서 노출 시간을 적절히 조정해야 합니다.

스트로브 라이트 소스 (사용 사례 3 및 4)

조명이 꺼진 후 센서 노출을 종료하려면 센서 노출 시간을 적절히 조정해야 합니다.

늦은 스트로브 (사용 사례 1 및 3)

스트로브 신호의 리딩 에지(시작)는 이미징 장치가 픽셀을 리셋하는 동안 빛이 너무 빨리 켜지지 않도록 약간 지연됩니다.

빠른 스트로브 (사용 사례 2 및 4)

Camera Trigger 신호의 리딩 에지는 센서 노출 시간이 가능한 한 짧게 유지되고 켜짐 시간과 거의 일치하도록 하기 위해 약간 지연됩니다.

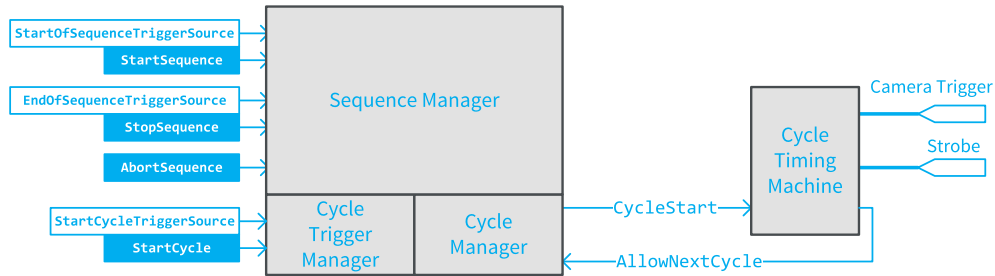
7. 카메라 및 조명 컨트롤러

7.1. CIC 블록 다이어그램	93
7.2. 주기 타이밍 머신	94
7.3. 주기 관리자	96
7.4. 주기 트리거 관리자	97
7.5. 시퀀스 관리자	99
7.6. CIC 출력 신호 라우팅	101
7.7. CIC 타이밍 다이어그램	102
주기 타이밍 다이어그램	103
CIC주기 연속 중첩	105
주기 타이밍 다이어그램	110

7.1. CIC 블록 다이어그램

카메라 및 illumination 컨트롤러 (CIC로 약칭)는 하나의 카메라 및 관련 조명을 조절합니다. 카메라는 **Camera Trigger** 신호로 조절되며 조명 장치는 **Strobe** 신호로 제어됩니다.

참조: "CIC 출력 신호 라우팅" 페이지101



카메라 및 조명 컨트롤러 블록 다이어그램

CIC는 4 개의 주요 상호 연결된 블록으로 구성됩니다.

- *Cycle Timing Machine*은 하나의 카메라 및 조명 컨트롤러 사이클 (CIC 사이클)을 구성하는 정확한 타이밍 이벤트 및 신호의 생성을 담당합니다. 즉 **Camera Trigger** 및 **Strobe**입니다.

참조: "주기 타이밍 머신" 다음 페이지

- *Cycle Manager*는 **CycleStart** 이벤트를 생성합니다. 시작 주기 조건이 모두 충족되지 않는 동안 및 주기 타이밍 머신이 새 주기를 시작할 수 없는 동안 새 주기를 시작하는 것을 방지합니다.

참조: "주기 관리자" 페이지96

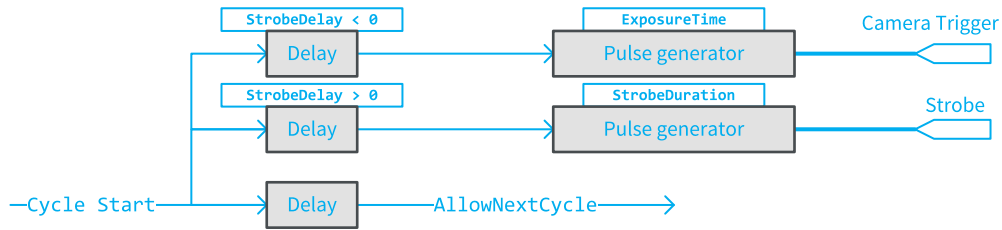
- *Cycle Trigger Manager*는 *Cycle Manager* 및 *Sequence Manager*와 *Cycle Timing Machine*의 한 주기를 시작하는 효과적인 **CycleStart** 이벤트를 작성합니다.

참조: "주기 트리거 관리자" 페이지97

- *Sequence Manager*는 사용자 정의 시작 순서 및 정지 순서 조건에 따라 CIC주기의 순서를 관리합니다.

참조: "시퀀스 관리자" 페이지99

7.2. 주기 타이밍 머신



CIC 주기 타이밍 머신 블록 다이어그램

CIC 타이밍 머신은 하나의 카메라 및 조명 컨트롤러 사이클 (CIC 사이클)을 구성하는 정확한 타이밍 이벤트 및 신호의 생성을 담당합니다.

Cycle Start 이벤트가 발생할 때마다 타이밍 머신이 다음을 생성합니다.

- Camera Trigger 신호에 하나의 단일 펄스.
- Strobe 신호에 하나의 단일 펄스.
- 하나의 AllowNextCycle 이벤트입니다.

주기적 타이밍

장치 모듈의 세 가지 GenICam 기능은 주기 내에서 출력 신호의 타이밍을 구성하는 데 사용됩니다:

- ExposureTime는 Camera Trigger 펄스의 지속 기간을 정의합니다.
- StrobeDuration는 Strobe 펄스의 지속 기간을 정의합니다.
- StrobeDelay는 Camera Trigger의 리딩 엣지에서 Strobe의 리딩 엣지까지의 시간 오프셋을 정의합니다.

참조: 자세한 설명과 타이밍 다이어그램을 위한 "주기 타이밍 다이어그램" 페이지 103

주기-주기 타이밍

`AllowNextCycle` 이벤트는 사이클 관리자가 다음 사이클을 시작할 시기를 결정하는 데 사용됩니다.

`AllowNextCycle` 이벤트의 위치는 사용자가 직접 설정하지 않습니다. 대신 다음 사용자 설정에 따라 Coaxlink 드라이버에서 평가됩니다:

- 카메라 모델 범주의 `ExposureReadoutOverlap` 기능은 카메라가 노출/판독 중첩을 지원하는지 여부를 정의합니다. 겹치기가 허용되면 `AllowNextCycle` 이벤트가 더 일찍 발생하고 더 빠른 주기 속도가 얻어집니다.
- 카메라 모델 범주의 `ExposeRecovery` 기능은 두 번의 노출 사이에 카메라가 요구하는 최소 시간 간격을 정의합니다. 이 기능은 `ExposureReadoutOverlap = TRUE`이고 노출 단계의 지속 시간이 판독 단계의 지속 시간보다 클 때 적합합니다.
- 주기 조절 범주의 `CycleMinimumPeriod`는 최소 주기를 정의합니다. 이 값은 카메라가 이미지 판독을 수행하는 데 필요한 시간보다 짧을 수 없습니다!



참고 일부 카메라에는 이미지 데이터 경로에 데이터 저장소가 있습니다. 이를 통해 카메라-프레임 그래버 데이터 링크가 유지할 수 있는 것보다 더 높은 주기-주기 비율로 이미지 버스트를 캡처할 수 있습니다. 이 경우, `CycleMinimumPeriod`는 이미지 센서가 얻을 수 있는 가장 작은 주기-주기 시간을 선언합니다!

참조: 자세한 설명 및 타이밍 다이어그램은 "[CIC주기 연속 중첩](#)" 페이지 105를 참조하십시오.

7.3. 주기 관리자

주기 관리자는 `Cycle Start` 이벤트를 생성합니다.

이후 열거된 주기 시작 조건이 충족되지 않으면 새 주기가 시작되지 않습니다:

시퀀스 활성화 조건

시퀀스 관리자는 활성화 상태여야 합니다: 즉, 시퀀스가 시작되었고 시퀀스 중지 조건에 아직 도달하지 않았습니다.

이 조건은 항상 적용됩니다.

다음 주기 허용 조건

주기 타이밍 머신은 이미 이전 사이클의 `Allow Next Cycle` 이벤트를 발행했습니다.

이 조건은 항상 적용됩니다.

사용 가능한 메모리 조건

다음 주기의 이미지 데이터를 수집하기에 충분한 여유 메모리가 보드에 있습니다.

이 조건은 항상 적용됩니다.

주기 트리거 이벤트 조건

새 주기를 시작하려면 새 주기 트리거 이벤트가 필요합니다.

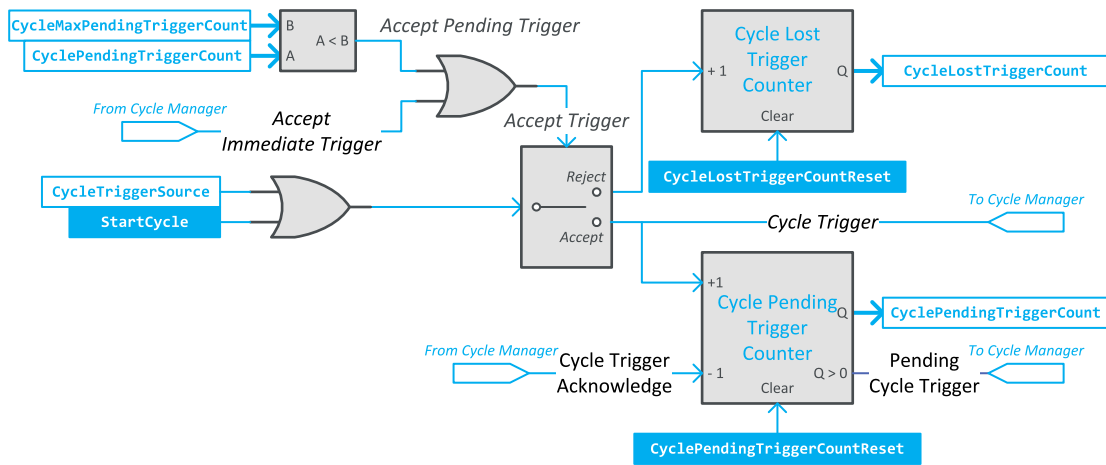
이 조건은 `Cycle TriggerSource ≠ Immediate`이고 `CycleMaxPendingTriggerCount = 0`인 경우에만 적용됩니다.

보류 중 트리거 조건

서비스를 기다리는 최소 하나 이상의 보류중인 트리거 (가능하면 새 트리거)가 있습니다.

이 조건은 `Cycle TriggerSource ≠ Immediate`이고 `CycleMaxPendingTriggerCount > 0`인 경우에만 적용됩니다.

7.4. 주기 트리거 관리자



CIC 주기 트리거 관리자 블록 다이어그램

주기 트리거 관리자는 주기 관리자와 협력하여 CIC 타이밍 머신의 한 주기를 시작하는 효과적인 `Cycle Trigger` 이벤트를 작성합니다.

주기 트리거 소스

주기 트리거의 소스는 `CycleTriggerSource`로 정의됩니다

로 설정하면 `Immediate`, 주기 트리거 관리자가 자동으로 트리거됩니다. 시퀀스 시작 직후 주기 트리거를 생성한 다음 `CycleMinimumPeriod` 주기마다 반복적으로 주기 트리거를 생성합니다.

`StartCycle` 또는 `<any I/O toolbox event source>`로 설정하면 사이클 트리거 관리자가 시퀀스 시작 직후 시작되지 않고 대신 `StartCycle` 명령의 실행 또는 선택된 I/O 도구 상자 이벤트 소스에서 이벤트 발생을 기다립니다.

다양한 주기 트리거 이벤트 소스 세트를 사용할 수 있습니다. 여기에는 LIN, QDC, MDV, DIV, DEL, EIN 및 사용자 이벤트와 같은 모든 I/O 도구 상자 이벤트가 포함됩니다.

주기 트리거 래치 메커니즘

주기 관리자에는 즉시 제공할 수 없는 주기 트리거를 래칭할 수 있는 트리거 래치 메커니즘이 장착되어 있습니다. 이러한 트리거는 해당 CIC주기가 시작될 때까지 실행이 단순히 연기되므로 "대기 중 트리거"로 명명됩니다.

기록 할 수 있는 보류중인 트리거의 최대 수는 `CycleMaxPendingTriggerCount`에 의해 정의됩니다. `CycleMaxPendingTriggerCount` = 일 때 트리거 래치 메커니즘이 비활성화됩니다. 이것은 기본값입니다. 트리거 래치 메커니즘을 사용하려면 `CycleMaxPendingTriggerCount` 을 1-7 범위의 정수값으로 설정하십시오.

보류중인 트리거의 수는 `CyclePendingTriggerCount`에 의해 보고됩니다.

주기 트리거 이벤트 정렬

`CycleTriggerSource`가 `StartCycle` 또는 `<any I/O toolbox event source>`로 설정된 경우 모든주기 트리거 이벤트는 *트리거 허용 기준*과 비교하여 평가되고 결과에 따라 정렬됩니다.

거부된 주기 트리거이벤트는 *Cycle Lost Trigger Counter*를 증가시킵니다.

대기 중인 트리거를 즉시 처리할 수 없는 경우 허용된 주기 트리거 이벤트가 *Cycle Pending Trigger Counter*를 증가시킵니다.

트리거 허용 기준

주기 트리거 이벤트는 두 조건이 모두 충족 될 때 즉시 수락되어 실행됩니다.

- 주기 시퀀스 활성화
- 주기 관리자는 현재 즉시 트리거 이벤트가 새 주기를 시작하기를 기다리고 있습니다 (즉시 트리거 허용)

주기 트리거 이벤트는 다음 조건이 충족될 때 나중에 받아들여지고 실행됩니다.

- 주기 시퀀스 활성화
- 보류중인 트리거 `CycleMaxPendingTriggerCount`의 수는 보다 적습니다. `CycleMaxPendingTriggerCount`
- 주기 관리자는 (아직) 새로운 주기를 시작할 준비가 되어 있지 않습니다.

7.5. 시퀀스 관리자

시퀀스 관리자는 CIC의 최상위 관리자입니다. 주기 트리거 관리자와 주기 관리자를 조절합니다.

If는 사용자 정의 시작 순서 및 정지 순서 조건에 따라 동일한 CIC 사이클의 순서를 정의합니다.

시퀀스 시작하기

시퀀스 시작 조건은 `StartOfSequenceTriggerSource`에 의해 정의됩니다.

`StartOfSequenceTriggerSource`가 `Immediate`(기본 설정)로 설정된 경우, 시퀀스 관리자는 주기 관리자 및 주기 트리거 관리자가 첫 번째 주기를 진행할 수 있도록 더 이상의 조치가 필요하지 않습니다.

주기 관리자의 `CycleTriggerSource` 설정에 따라 첫 번째 주기가 실행됩니다:

- `CycleTriggerSource`가 로 설정된 즉시 `Immediate`
- `CycleTriggerSource`가 `StartCycle`로 설정된 경우 `StartCycle` 명령을 실행할 때 또는
- `StartCycle` 명령을 실행하거나 `CycleTriggerSource`에 의해 지정된 I/O 도구 상자 이벤트 소스에서 이벤트가 발생하는 경우.

`StartOfSequenceTriggerSource`가 `StartSequence`로 설정되면, 시퀀스 관리자는 주기 관리자와 주기 트리거 관리자가 첫 번째 주기를 진행하도록 허용하기 전에 `StartSequence` 명령의 실행을 기다립니다.

When `StartOfSequenceTriggerSource`가 `<any-event-source>`로 설정되면, 시퀀스 관리자는 주기 관리자와 주기 트리거 관리자를 허용하기 전에 `StartSequence` 명령의 실행 또는 I/O 도구 상자 이벤트의 발생을 기다립니다.

시퀀스 중지

시퀀스 정지 조건은 `EndOfSequenceTriggerSource`에 의해 정의됩니다.

`EndOfSequenceTriggerSource`가 `StopSequence`(기본 설정)로 설정되면 시퀀스 관리자는 `StopSequence` 명령을 실행한 후 다음 주기 경계에서 시퀀스를 중지합니다.

`EndOfSequenceTriggerSource`가 `SequenceLength`로 설정된 경우, 로 지정된 수의 카메라 주기를 실행한 후 시퀀스 관리자가 시퀀스를 자동으로 중지합니다. `SequenceLength StopSequence` 명령을 실행하면 예상대로 시퀀스를 중지할 수 있습니다. 기본 `SequenceLength` 값은 1입니다. 16,777,215까지의 모든 값이 허용됩니다.

`EndOfSequenceTriggerSource`가 `<any-event-source>`로 설정되면 시퀀스 관리자는 다음 주기 경계에서 시퀀스를 중지하기 전에 `StopSequence` 명령 실행 또는 지정된 이벤트 소스에서 I/O 도구 상자 이벤트 발생을 기다립니다.



참고 `StartOfSequenceTriggerSource` 및 설정의 조합은 허용됩니다. `EndOfSequenceTriggerSource`

카메라가 잡는 동안 시퀀스 길이 변경하기

릴리스 10.5부터 `SequenceLength`가 시퀀스 시작 및 종료 이벤트 사이에서 변경되면 새 값이 후속 시퀀스에 유효합니다.



참고 `SequenceLength`의 값은 시퀀스 시작 이벤트에서 래치됩니다.

7.6. CIC 출력 신호 라우팅

카메라 트리거 신호

프레임 그래버는 Camera Trigger 신호를 통해 비동기 리셋 카메라의 카메라 주기를 조절합니다.

신호는 다음 매체 중 하나를 사용하여 Coaxlink 카드에서 카메라로 전송될 수 있습니다:

- CoaXPress 링크의 I/O 채널
- TTL I/O로 구동되는 전용 배선

CoAXPress I/O 채널

Camera Trigger 신호는 CoaXPress I/O 채널의 우선 순위가 높은 호스트-장치 트리거 메시지로 카메라로 전송됩니다

참조: 호스트 - 디바이스 트리거 송신기에 대한 자세한 정보는 "[CoaXPress 호스트-장치 트리거](#)" 페이지39를 참조하십시오.

TTL I/O 라인

모든 TTL I/O 라인은 Camera Trigger 출력으로 설정할 수 있습니다. I/O 제어 블록의 극성 조절은 각 I/O 포트에 대한 개별 극성 조절을 제공합니다. TTLIO 라인의 I/O 조절 블록의 모드 조절은 개별 출력 드라이버 설정을 제공합니다.

스트로브 신호

I/O 라인

모든 출력 가능 I/O 라인은 일루미네이션 스트로브 출력으로 설정할 수 있습니다. I/O 제어 블록의 극성 조절은 각 I/O 포트에 대한 개별 극성 조절을 제공합니다. TTLIO 라인의 I/O 제어 블록의 모드 제어는 스트로브 출력으로 사용되는 각 I/O 포트에 대한 개별 출력 드라이버 설정을 제공합니다.

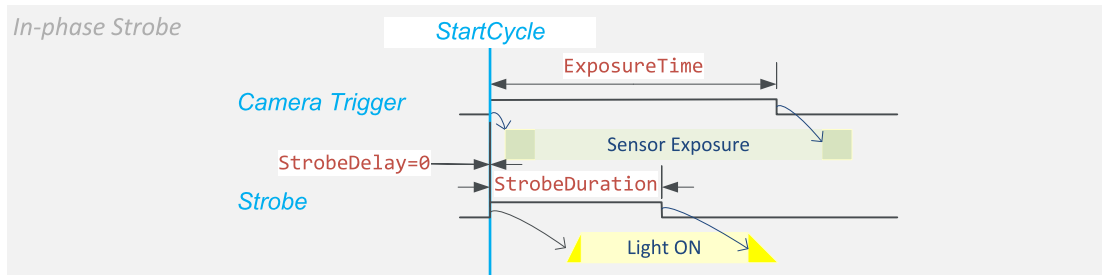
7.7. CIC 타이밍 다이어그램

주기 타이밍 다이어그램	103
CIC주기 연속 중첩	105
주기 타이밍 다이어그램	110

주기 타이밍 다이어그램

이 항목에서는 개별 cic주기의 타이밍 다이어그램을 보여주고 **StrobeDelay**의 양수, 0 및 음수 값에 해당하는 3 가지 사용 사례에 대한 **ExposureTime**, **StrobeDuration** 및 기능을 보여줍니다. **StrobeDelay**

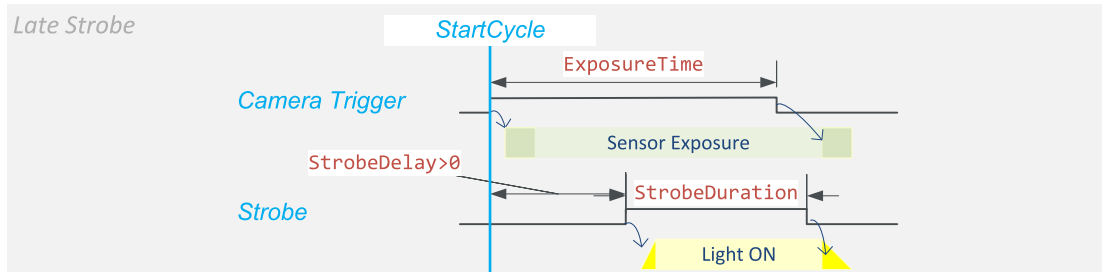
정시 스트로브 케이스



StrobeDelay = 0

Camera Trigger와 Strobe 신호는 StartCycle 이벤트 직후에 높아집니다.

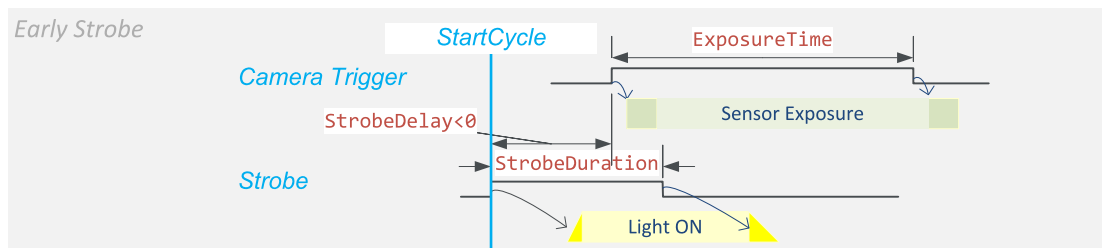
늦은 스트로브 케이스



StrobeDelay > 0

Camera Trigger 신호는 StartCycle 이벤트 직후 하이가되고 StrobeDelay 마이크로초 후에 스트로브 신호가 하이가 됩니다.

빠른 스트로브 케이스



StrobeDelay < 0

스트로브 신호는 StartCycle 이벤트 직후 하이가 되고 Camera Trigger 신호는 StrobeDelay 마이크로초의 반대값과 동일한 시간 지연 후에 하이가 됩니다.

CIC주기 연속 중첩

소개

이 항목에서는 `ExposureReadoutOverlap = True`일 때 트리거 오버런을 피하고 주기 속도를 최대한화하도록 CIC를 올바르게 설정할 때 사용자가 `ExposureRecovery` 및 `TargetFramePeriod` 기능을 설정해야 하는 방법을 설명합니다.

Coaxlink 드라이버는 다음 조건을 만족하는 가장 작은 값을 검색하여 사용자 정의 설정 `ExposureTime`, `ExposureRecovery` 및 `TargetFramePeriod` 에서 카메라 주기값의 지속 시간을 계산합니다:

- 조건 1: 연속적인 `Camera Trigger` 펄스 사이의 시간 간격 (그림에서 r)은 `ExposureRecovery` 설정보다 크거나 같아야 합니다. 이렇게 하면 `Camera Trigger`가 트리거 전송 링크를 통해 올바르게 흐르게 됩니다. 또한 이전 노출이 완료되기 전에 새로운 노출이 시작되지 않도록 합니다.
- 조건 2: CIC 주기 지속 시간(그림의 a)은 새로운 판독값이 이전 판독값의 완료 전에 시작되지 않도록 충분히 커야 합니다.
- 조건 3: CIC 주기 지속 시간은 `Camera Trigger`와 스트로브 신호의 전환을 모두 포함할 만큼 커야 합니다.

"판독 제한적" 사용 사례는 주기 기간이 판독 단계의 지속 기간과 동일한 상황을 나타냅니다.

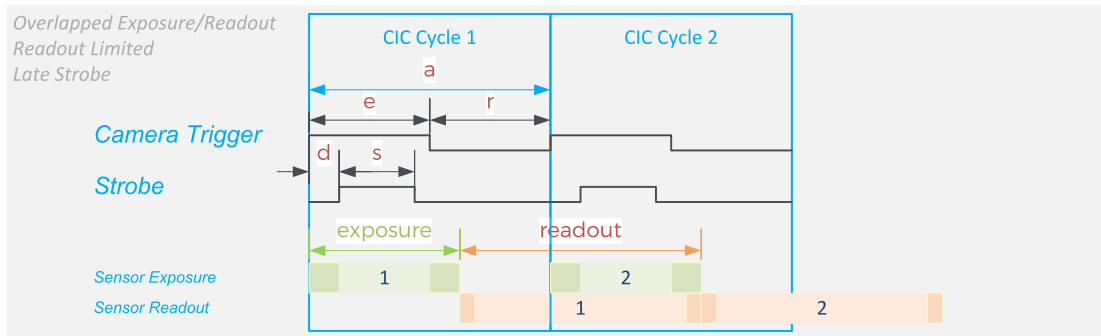
"노출 제한" 사용 사례는 주기 기간이 노출 단계의 지속 기간과 동일한 상황을 나타냅니다.

다음 타이밍 다이어그램에서:

e 는 `Camera Trigger` 펄스 폭입니다. 이 값은 `ExposureTime`에 의해 정의됩니다.

d 는 `Strobe` 지연입니다. 이 값은 `StrobeDelay`에 의해 정의됩니다.

사례 1: 제한적 판독값 (늦은 스트로브)



카메라 순환 속도는 카메라 판독 시간에 의해서만 제한됩니다

이 상황은 e (= Exposure Time)가 판독 기간보다 현저히 짧을 때 발생합니다.

그 상황에서:

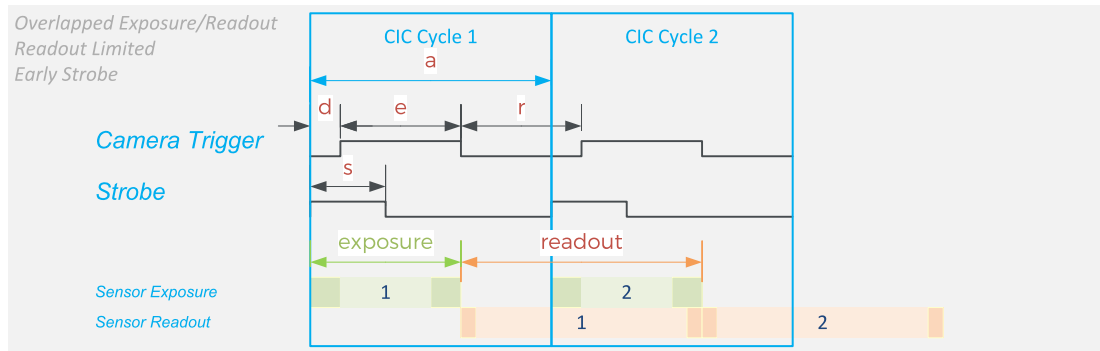
- r 은 **ExposureRecovery**: 조건 1이 충족될 가능성이 큼니다.
- Camera Trigger 펄스의 "내부"에 있는 스트로브 펄스: 조건 1이 충족되면 조건 3은 무의미합니다.
- 조건 2는 주기 지속 기간을 계산하기 위해 Coaxlink 드라이버가 사용하는 유일한 조건입니다

주기의 최적 지속 시간은 센서 판독 단계의 유효 기간과 같습니다. 이는 사용자가 **TargetFramePeriod**를 판독 기간에 해당하는 값으로 설정한 경우에 얻어집니다.



참고 판독 기간은 카메라 데이터 시트의 최대 프레임 속도 사양 또는 실험적으로 구할 수 있습니다.

사례 2: 판독 제한 (빠른 스트로브)



카메라 주기 속도는 (빠른 스트로브에도 불구하고) 카메라 판독 시간에 의해서만 제한됩니다

이 상황은 사례 1과 유사합니다. 빠른 스트로브에도 불구하고 카메라의 최대 주기 속도에 도달할 수 있음을 보여줍니다.

이 상황은 e (= ExposureTime)가 판독 기간보다 현저히 짧을 때 발생합니다.

그 상황에서:

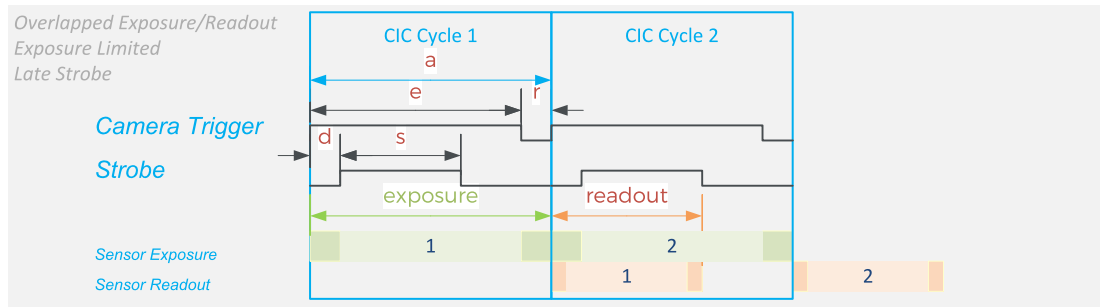
- r 은 ExposureRecovery: 조건 1이 충족될 가능성이 큽니다.
- 스트로브 펄스가 Camera Trigger 펄스 이전에 중단됨: 조건 3은 r 이 d 보다 크면 충족됩니다. 이것은 $(d + e < \text{판독 기간})$ 인 경우입니다.
- 조건 2는 주기 지속 기간을 계산하기 위해 Coaxlink 드라이버가 사용하는 유일한 조건입니다.

주기의 최적 지속 시간은 센서 판독 단계의 유효 기간과 같습니다. 이는 사용자가 TargetFramePeriod를 판독 기간에 해당하는 값으로 설정한 경우에 얻어집니다.



참고판독 기간은 카메라 데이터 시트의 최대 프레임 속도 사양 또는 실험적으로 구할 수 있습니다.

사례 3: 제한적 판독값 (스트로브 늦음)



카메라 사이클 속도는 노출 시간 설정에 따라 제한됩니다

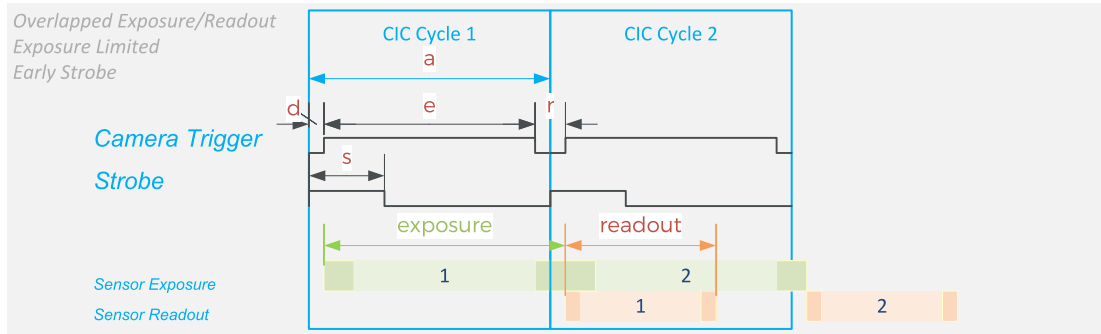
이 상황은 e (= Exposure Time)가 판독 기간보다 현저히 클 때 발생합니다.

그 상황에서:

- 모든 주기가 동일하고, 판독 지속 시간이 노출 지속 시간보다 작으면, 조건 2가 부적합함을 의미합니다.
- Camera Trigger 펄스의 "내부"에 있는 스트로브 펄스: 조건 1이 충족되면 조건 3은 무의미합니다.
- 조건 1은 주기 지속 기간을 계산하기 위해 Coaxlink 드라이버가 사용하는 유일한 조건입니다.

주기의 최적 기간은 노출 단계의 유효 기간과 동일합니다. 이것은 사용자가 **ExposeRecovery**를 연속 펄스 사이에서 카메라가 허용한 최소 시간 간격에 해당하는 값으로 설정한 경우에 얻어집니다. Camera Trigger

사례 4: 빠른 스트로브로 노출이 제한됨



카메라 주기 속도는 (빠른 스트로브에도 불구하고) 노출 시간 설정에 의해 제한됩니다

이 상황은 사례 3과 유사합니다. 빠른 스트로브에도 불구하고 작은 음수 **StrobeDelay** 값의 동일한 사이클 속도로 이끌 수 있음을 보여줍니다.

이 상황은 $e (= \text{ExposureTime})$ 가 판독 기간보다 현저히 클 때 발생합니다.

그 상황에서:

- 모든 주기가 동일하고, 판독 지속 시간이 노출 지속 시간보다 작 으면 조건2가 부적합함을 의미합니다.
- Camera Trigger 펄스 이전에 종료되는 스트로브 펄스: 조건1이 충족되고 $d < r$ 일 때 조건3은 무의미합니다.
- 조건3 및 조건1은 주기 지속 기간을 계산하기 위해 Coaxlink 드라이버가 사용하는 유일한 조건입니다.

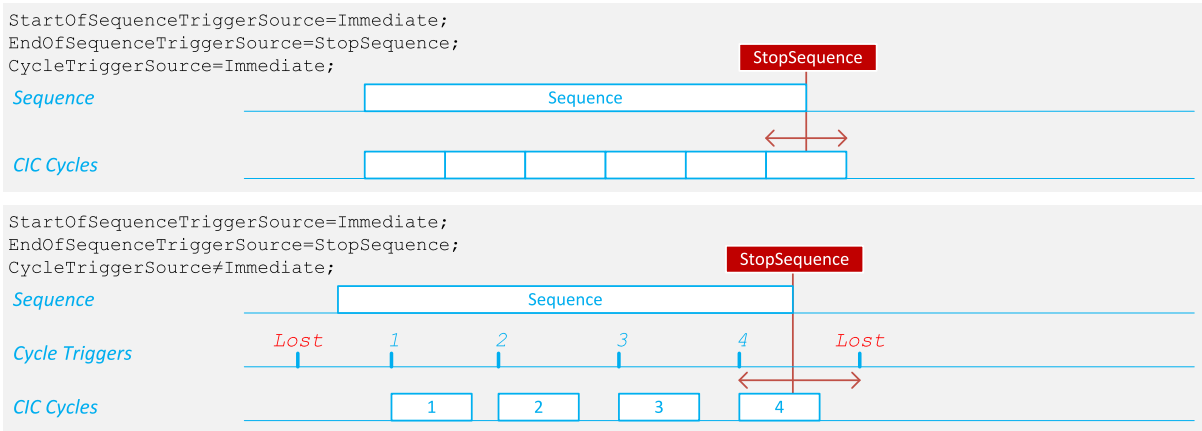
사용자는 **ExposeRecovery**를 다음 두 값 중 가장 큰 값에 해당하는 값으로 설정해야 합니다.

- 연속적인 Camera Trigger 펄스 사이에서 카메라가 허용하는 최소 시간 간격.
- **StrobeDelay**의 반대값

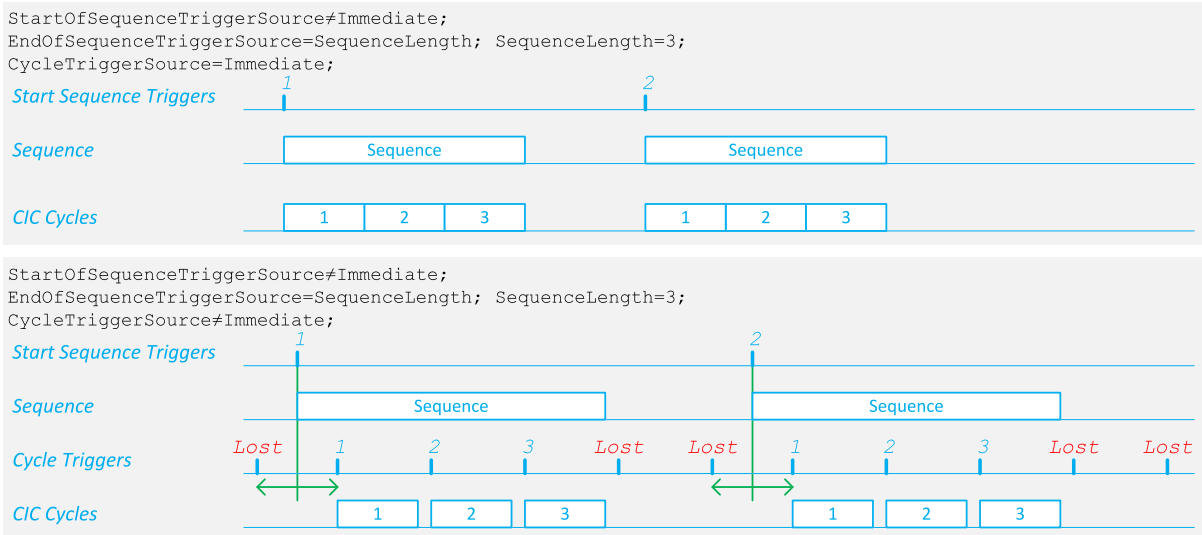


참고 **CycleTriggerSource = Immediate**일 때, **TargetFramePeriod**에 더 큰 값을 할당하여 주기율을 원하는 비율로 낮출 수 있습니다.

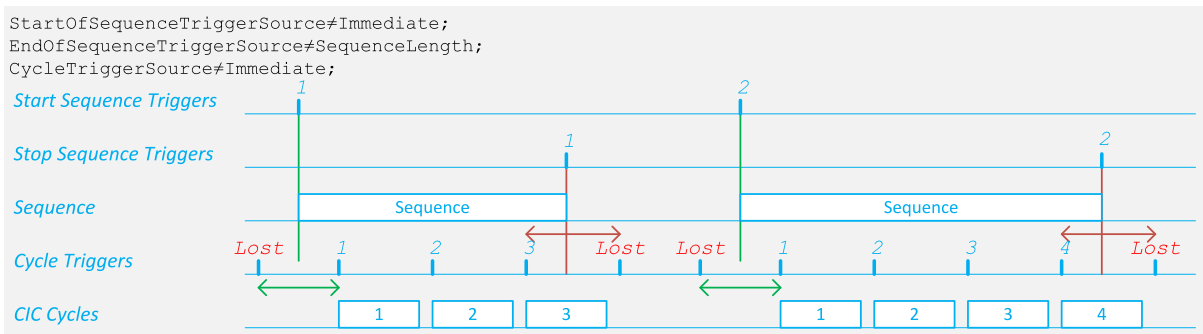
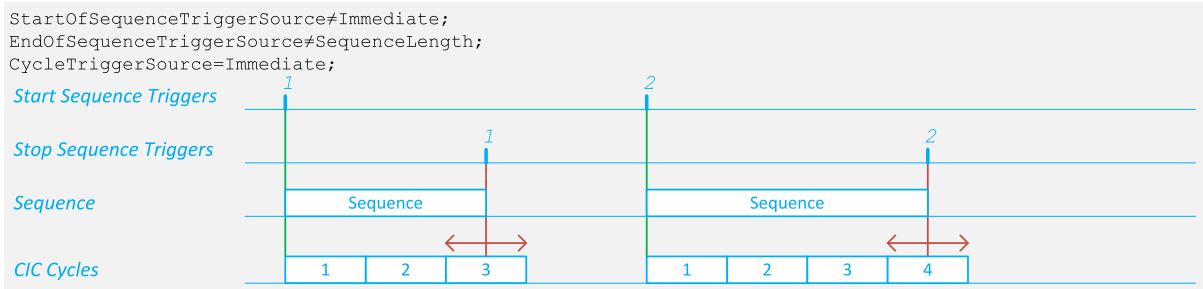
주기 타이밍 다이어그램



즉각적인 시작의 주기 시퀀스



트리거된 시작 및 고정 길이의 주기 시퀀스



트리거된 시작 및 트리거된 끝이 있는 주기 시퀀스

8. 범용 I/O

8.1. I/O 라인 개요	113
8.2. I/O 라인 사용법	116
8.3. I/O 제어 블록	117
8.4. 라인 포맷 및 라인 모드 컨트롤	119
8.5. 라인 극성 제어	122
8.6. 필터 조절	123
8.7. 라인 소스 선택	124
8.8. 논리적 I/O 라인 상태	125
8.9. 물리적 I/O 라인 상태	126
8.10. 라인 드라이버 물리적 출력 상태	127
8.11. 초기 상태	128

8.1. I/O 라인 개요

제품 당 I/O 라인

제품	표준 I/O 세트		I/O 확장
	#1	#2	
1629 Coaxlink Duo PCIe/104-EMB	3300	-	
1630 Coaxlink Mono	온보드	-	-
1631 Coaxlink Duo	온보드	온보드	-
1632 Coaxlink Quad	온보드	온보드	-
1633 Coaxlink Quad G3	온보드	온보드	-
1633-LH Coaxlink Quad G3 LH	온보드	온보드	-
1634 Coaxlink Duo PCIe/104-MIL	3300	-	
1635 Coaxlink Quad G3 DF	온보드	-	-
1637 Coaxlink Quad 3D-LLE	온보드	온보드	-
3602 Coaxlink Octo	온보드	3614	✓
3603 Coaxlink Quad CXP-12	온보드	온보드	✓
3620 Coaxlink Quad CXP-12 JPEG	온보드	온보드	✓
3621-LH Coaxlink Mono CXP-12 LH	온보드	3614	✓
3622 Coaxlink Duo CXP-12	온보드	3614	✓

I/O 확장 모듈: 3610 3612

표준 I/O 세트

표준 I/O 세트 #1

I/O 라인 유형	I/O 라인 이름	수
절연 입력	IIN11, IIN12, IIN13, IIN14	4
절연 출력	IOU11, IOU12	2
RS-422 입력	DIN11, DIN12	2
TTL I/O	TTLIO11, TTLIO12	2
총		10

표준 I/O 세트 #2

I/O 라인 유형	I/O 라인 이름	수
절연 입력	IIN21, IIN22, IIN23, IIN24	4
절연 출력	IOOUT21, IOOUT22	2
LVTTL 입/출력	TTLIO21, TTLIO22	2
RS-422 입력	DIN21, DIN22	2
총		10

I/O 및 I/O 확장 모듈

3300 I/O 모듈

3300 HD26F I/O module for Coaxlink Duo PCIe/104 는 "표준 I/O 세트 #1" 이전 페이지를 제공합니다.

참조: 자세한 내용은 Coaxlink PCIe/104 하드웨어 설명서의 [3300/3302 Accessories for Coaxlink Duo PCIe/104](#)를 참조하십시오.

3610/3612 I/O 확장 모듈

3610 HD26F I/O Extension Module TTL-RS422 및 **3612 HD26F I/O Extension Module TTL-CMOS5V-RS422** 는 "3610/3612 모듈" I/O 세트를 제공합니다:

I/O 라인 유형	I/O 라인 이름	수
RS-422 입력	MIO1, MIO3...MIO19	0-10
RS-422 출력	MIO1, MIO3...MIO19	0-10
TTL 입력	MIO1, MIO2...MIO20	0-20
TTL 출력 (3610) CMOS 출력 (3612)	MIO1, MIO2...MIO20	0-20
총		10 RS-422 0 TTL/CMOS ... 0 RS-422 20 TTL/CMOS

참조: 자세한 내용은 하드웨어 설명서의 [3610/3612 I/O Extension Modules](#)를 참조하십시오.

3614 I/O 확장 모듈

3614 HD26F I/O Extension Module - Standard I/O Set 는 "표준 I/O 세트 #2" 위를 제공합니다.

참조: 자세한 내용은 하드웨어 설명서의 [3614 I/O Extension Module](#)를 참조하십시오.

I/O 라인 유형

I/O 라인 유형	전기 스타일	주파수 범위	이름 접두사
CMOS 출력	5V CMOS 단일 종단 출력	최대 1 MHz	MIO
절연 입력	5V, 12V 및 24V 신호 레벨을 준수하는 절연 전류 감지 입력	최대 50 kHz	IIN
절연 출력	절연된 100mA/30 V 접점 출력	최대 50 kHz	IOUT
RS-422 입력	RS-422 차동 입력	최대 5 MHz	DIN
		최대 10 MHz	MIO
RS-422 출력	RS-422 차동 출력	최대 10 MHz	MIO
TTL I/O	5V 내성 LVTTTL 단일 종단형 입력 및 3.3 V LVTTTL 토템폴 단일 종단 출력 또는 3.3 V 오픈 이미터 단일 종단 출력 또는 3.3 V 오픈 컬렉터 단일 종단 출력	최대 5 MHz	TTLIO
TTL 입력	5V 내성 LVTTTL 단일 종단형 입력	최대 1 MHz	MIO
TTL 출력	3.3 V LVTTTL 단일 종단 출력	최대 1 MHz	MIO

8.2. I/O 라인 사용법

입력 라인

입력 가능 I/O 라인 (DIN, TTLIO, IN, MIO)은 다음과 같이 사용할 수 있습니다:

- 범용 입력 : 호스트 응용 프로그램에서 상태를 읽거나 모니터할 수 있는 입력 신호.
- 모션 인코더 입력 : 쿼드 러처 모션 인코더에 의해 전달되고 카메라에서 획득을 트리거하는 데 사용되는 한 쌍의 입력 신호.
- 트리거 입력 : 카메라에서 획득을 트리거하는 데 사용되는 입력 신호.

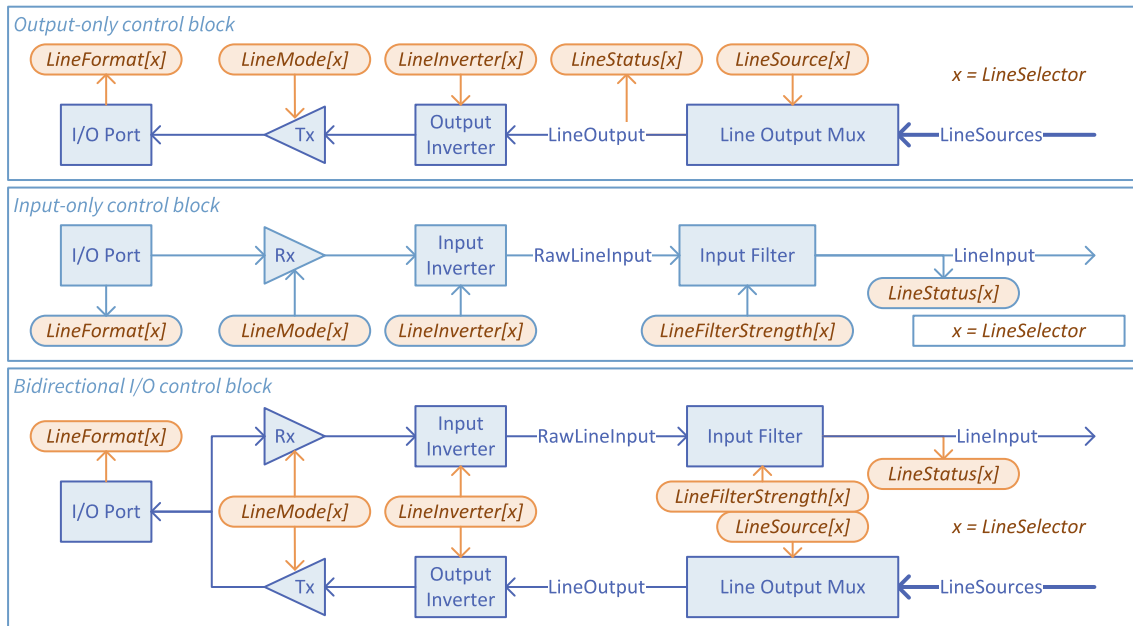
출력 라인

출력 가능 I/O 라인 (TTLIO, IN, MIO)은 다음과 같이 사용할 수 있습니다:

- 범용 출력: 호스트 응용 프로그램에서 상태를 설정할 수 있는 출력 신호.
- 스트로브 출력: 일반적으로 카메라와 동기화되어 스트로브 라이트를 제어하는 데 사용되는 출력 신호입니다.
- 카메라 트리거 출력 (TTLIO에서만 사용 가능): Coaxlink 카드에서 생성되고 카메라 트리거에 사용되는 출력 신호.

8.3. I/O 제어 블록

모든 I/O 라인은 하나의 I/O 제어 블록을 통해 제어됩니다. I/O 라인의 입력/출력 기능에 따라 3가지 유형의 제어 블록이 있습니다.



위 그림에서:

- 가는 파란색 선은 하나의 개별 전기 신호 경로를 나타냅니다.
- 두꺼운 파란색 선은 전기 신호 경로 모음을 나타냅니다.
- 파란색 화살표는 전기 신호의 전파 방향을 보여줍니다.
- 파란색 모양은 I/O 제어 블록의 기능 요소를 나타냅니다.
- 주황색(타원형) 모양은 GenICam 기능을 나타냅니다. 내부의 텍스트는 기능 이름입니다.
- 기능 이름에 [x]가 붙으면 기능이 선택터 기능과 연결되어 있음을 나타냅니다 (이 경우 LineSelector).
- 주황색 화살표 끝은 기능의 액세스 모드를 나타냅니다. 들어오는 화살표가 있는 읽기 전용 기능, 나가는 화살표가 있는 쓰기 가능한 기능.

입력 경로

입력 전용 및 양방향 I/O 라인은 다음을 포함하는 공통 입력 경로 구조를 공유합니다.

- I/O 커넥터의 I/O 핀을 나타내는 I/O 포트 블록
- 라인 수신기 회로를 나타내는 Rx 블록
- 사용자 구성 가능한 로직 인버터를 나타내는 입력 인버터 블록
- 사용자가 구성할 수 있는 글리치 제거 필터를 나타내는 입력 필터 블록

출력 경로

출력 전용 및 양방향 I/O 라인은 다음을 포함하는 공통 출력 경로 구조를 공유합니다.

- I/O 커넥터의 I/O 핀을 나타내는 I/O 포트 블록
- 라인 드라이버 회로를 나타내는 Tx 블록
- 사용자 구성 가능한 논리 인버터를 나타내는 출력 인버터 블록
- 사용자 구성 가능한 소스 멀티플렉서를 나타내는 라인 출력 멀티플렉서 블록

8.4. 라인 포맷 및 라인 모드 컨트롤

소개

다음 표는 각 I/O 회선의 I/O 제어 블록에 대한 세부 사항을 요약한 것입니다:

- 첫 번째 열은 값을 나타냅니다. **LineSelector**
- 두 번째 열은 에 의해 보고된 정수 값의 비트 위치를 나타냅니다. **LineStatusAll**
- 세 번째 열은 가 보고한 값을 나타냅니다. **LineFormat**
- 네 번째 열은 에 할당할 수 있는 값을 나타냅니다. **LineMode**

표준 I/O 세트

표준 I/O 세트 #1

LineSelector	비트#	LineFormat	LineMode
DIN11	0	DIFF	입력
DIN12	1	DIFF	입력
IIN11	4	ISO	입력
IIN12	5	ISO	입력
IIN13	6	ISO	입력
IIN14	7	ISO	입력
IOUT11	12	ISO	출력
IOUT12	13	ISO	출력
TTLIO11	16	TTL	Input또는OutputDriveLowDriveHigh
TTLIO12	17	TTL	Input또는OutputDriveLowDriveHigh

표준 I/O 세트 #2

LineSelector	비트#	LineFormat	LineMode
DIN21	1	DIFF	입력
DIN22	3	DIFF	입력
IIN21	8	ISO	입력
IIN22	9	ISO	입력
IIN23	1	ISO	입력
IIN24	11	ISO	입력
IOUT21	14	ISO	출력
IOUT22	15	ISO	출력
TTLIO21	18	TTL	Input또는OutputDriveLowDriveHigh
TTLIO22	19	TTL	Input또는OutputDriveLowDriveHigh

TTLIO 포트 모드 제어

기능은 LineModeIO 포트의 방향과 라인 드라이버 모드를 제어합니다. 언제든지 네 가지 모드를 선택할 수 있습니다:

- Input전용, 토템폴 드라이버 사용 불가 (디 파트 파워 업 설정),
- Output이를 구동할 수 있는 토템폴 드라이버,
- DriveLow동 할 수있는 오픈 콜렉터 드라이버,
- Drivehigh할 수 있는 오픈 에미 터 드라이버.



참고 두 가지 최신 구성은 와이어드 AND 구성을 허용합니다. 라인 상태는 입력 포트를 통해 다시 읽을 수 있습니다.

I/O 모듈

3610/3612 I/O 확장 모듈

LineSelector	비트#	LineFormat	LineMode
MIO1	20	TTLDIFF	InputOutput
MIO2	21	TTL	InputOutput
:	:	:	:
MIO19	38	TTLDIFF	InputOutput
MIO20	39	TTL	InputOutput

MIO 형식 및 모드 컨트롤

LineFormat 기능은 MIO 포트의 전기적 스타일을 제어합니다. 가능한 값:

- TTL일 종단형 (TTL 또는 CMOS)
- DIFF (RS422)

LineMode 기능은 MIO 포트의 방향을 제어합니다. 가능한 값:

- Input: 입력만
- Output: 이를 구동할 수 있는 토렘폴 드라이버.



참고 컨트롤은 모듈 구성 중에만 변경할 수 있습니다.

참조: 구성에 대한 자세한 설명은 [3610/3612 I/O Extension Modules](#)를 참조하십시오.

8.5. 라인 극성 제어

모든 I/O 라인에는 극성 제어 장치가 장착되어 있습니다. 양방향 I/O 회선의 경우 단일 제어가 두 경로에 똑같이 영향을 줍니다.

라인 극성은 **LineInverter** 조절을 통해 사용자가 설정할 수 있습니다.



참고참고 : 입력 경로의 *Line Input* 신호와 출력 경로의 *LineSources* 신호가 항상 양의 논리를 사용하도록 외부 신호의 극성에 따라 극성 제어를 설정하도록 사용자를 안내합니다.

8.6. 필터 조절

모든 I/O 입력 라인에는 글리치 제거 필터가 장착되어 있습니다. 필터 조절은 **LineFilterStrength** 컨트롤을 통해 사용자가 구성할 수 있습니다.

강도 조절은 '최저'에서 '최고'까지 5가지를 제공합니다. 기본 위치는 '낮음'입니다.

각 위치는 3개의 I/O 입력 라인 유형 각각에 대한 특정 필터 시간 상수에 해당합니다.

I/O 입력 라인 유형 및 LineFilterStrength 당 라인 필터 시간 상수:

LineFilterStrength	차동 입력 (DIN)	TTL 입력 (TTLIO)	절연 입력 (IIN)
최저	50 ns	50 ns	500 ns
저	100 ns	100 ns	1 μ s.
중	200 ns	200 ns	2 μ s.
고	500 ns	500 ns	5 μ s.
최고	1 μ s.	1 μ s.	10 μ s.

사용자는 외부 신호의 품질에 따라 필터 강도를 설정하도록 요청받습니다. 시간 상수가 다음과 같이 되도록 필터 강도를 선택합니다:

- 가장 긴 글리치 지속 시간보다 큼
- 신호의 10%~90 % 상승/하강 시간보다 큼
- 최소 신호 펄스 지속 시간보다 최소 2배 이상 작음

글리치 제거 필터는 입력 신호 경로에 대기 시간을 도입합니다. 대기 시간은 들어오는 신호의 깨끗한 전환이 있을 때의 필터 시정 수와 같습니다. 품질이 좋지 않은 신호의 경우 대기 시간이 크게 늘어날 수 있습니다.

8.7. 라인 소스 선택

모든 출력 가능 I/O 라인에는 소스 신호 멀티플렉서가 장착되어 있습니다.

소스 신호 멀티플렉서는 완전히 채워진 신호 라우팅 매트릭스를 구현하여 내부 신호 선택이 모든 출력 라인으로 라우팅되도록 합니다.

신호 소스 선택하기

출력 가능 I/O 라인의 라인 드라이버에 대한 신호 소스로 사용할 내부 신호를 선택하려면:

1 단계: **LineSelector** 기능에 적절한 값을 할당하여 I/O 라인을 선택하십시오

2 단계: **SignalSource** 기능에 적절한 값을 지정

출력-라인 라우팅 매트릭스

소스 신호	가능한 목적지
사용자 출력 레지스터의 임의의 비트	모든 출력 라인
모든 장치 모듈 카메라 트리거 출력	모든 TTLIO 출력 라인
모든 장치 모듈 스트로브 출력	모든 출력 라인
계속 낮음	모든 출력 라인

8.8. 논리적 I/O 라인 상태

논리적 I/O 라인 상태

I/O 라인의 (논리적) 상태는 I/O 제어 블록의 전기 신호의 논리적 상태입니다:

- 입력 가능 I/O 라인의 경우: *LineInput* 신호: 입력 인버터 다음에 있는 I/O 제어 블록의 입력 경로에 있는 지점입니다.
- 출력 전용 I/O 라인의 경우: *LineOutput* 신호는 출력 인버터 앞에 있는 I/O 제어 블록의 출력 경로에 있는 점입니다.

단일 I/O 라인의 상태 가져 오기

1. 1 단계: **LineSelector** 기능에 적절한 값을 할당하여 I/O 라인을 선택하십시오
2. 2 단계: **LineStatus** 기능의 값을 가져와서 라인 상태를 직접 얻습니다.

단일 작업으로 모든 I/O 라인의 상태 가져오기

LineStatusAll 기능의 값을 가져옵니다.

정수의 각 비트는 I/O 라인에 해당합니다. 하나의 비트는 하이인 라인 논리 상태에 대응합니다.

8.9. 물리적 I/O 라인 상태

I/O 라인 상태의 물리적 상태는 `LineStyle` but also on the following I/O block settings: `LineFormat`, `LineMode`, and `LineInverter`를 읽을 때 보고된 값에 의존하기만 하지 않음

LineFormat	LineMode	LineInverter/LineStyle	물리적 I/O 라인 상태
DIFF	입력	False/False 또는 True/True	$(VIN+ - VIN-) < VThreshold$
		False/True 또는 True/False	$(VIN+ - VIN-) > Vthreshold$ 또는 연결되지 않은 I/O 회선
ISO	입력	False/False 또는 True/True	광 커플러는 OFF. 라인 전류는 $< 1 \text{ mA}$. 라인이 연결되지 않거나 잘못된 극성으로 연결될 수 있습니다.
		False/True 또는 True/False	광 커플러는 ON. 라인 전류는 $> 1 \text{ mA}$.
ISO	출력	False/False 또는 True/True	광 커플러는 OFF
		False/True 또는 True/False	광 커플러는 ON
TTL	입력	False/False 또는 True/True	라인 전압은 $< 0.8 \text{ V}$.
		False/True 또는 True/False	라인 전압은 $> 2.0 \text{ V}$.
TTL	출력	False/False 또는 True/True	라인이 LOW로 구동됩니다.
		False/True 또는 True/False	라인이 HIGH로 구동됩니다.
TTL	DriveLow	False/False 또는 True/True	라인이 LOW로 구동됩니다.
		False/True 또는 True/False	라인 전압은 $> 2.0 \text{ V}$.
TTL	DriveHigh	False/False 또는 True/True	라인 전압은 $< 0.8 \text{ V}$.
		False/True 또는 True/False	라인이 HIGH로 구동됩니다.

8.10. 라인 드라이버 물리적 출력 상태

라인 드라이버 출력 상태는 선택된 라인 출력 신호의 논리 레벨뿐만 아니라 다음과 같은 I/O 블록 설정에 따라 달라집니다: `LineFormat`, `LineMode`, and `LineInverter`.

LineFormat	LineMode	LineInverter / LineOutput 논리 레벨	라인 드라이버 출력
ISO	출력	False / low 또는 True / high	광 커플러 스위치가 꺼집니다.
		False / high 또는 True / low	광 커플러 스위치가 켜집니다.
TTL CMOS	출력	False / low 또는 True / high	I/O 라인이 LOW로 구동됩니다.
		False / high 또는 True / low	I/O 라인이 HIGH로 구동됩니다.
TTL	DriveLow	False / low 또는 True / high	I/O 라인이 LOW로 구동됩니다.
		False / high 또는 True / low	I/O 라인은 구동되지 않습니다.
TTL	DriveHigh	False / low 또는 True / high	I/O 라인은 구동되지 않습니다.
		False / high 또는 True / low	I/O 라인이 HIGH로 구동됩니다.

8.11. 초기 상태

전원이 켜지면 I/O 라인은 다음과 같은 상태가 됩니다:

차동 입력

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

TTL 입/출력

- LineMode = **Input** (라인이 구동되지 않습니다),
- LineInverter = **False**,
- LineFilterStrength = **Low**.

절연 입력

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

절연 출력

- LineInverter = **False**,
- LineFilterStrength = **Low**,
- 광 커플러는 **OFF**입니다.

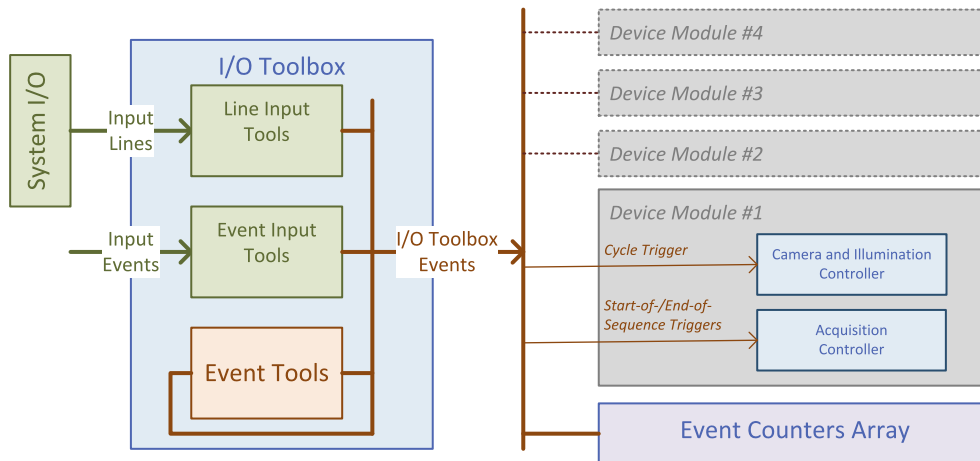
9. I/O 도구 상자

9.1. I/O 도구 상자 소개	130
9.2. I/O Toolbox 구성	133
9.3. 라인 입력 도구	136
9.4. 쿼드러처 디코더 도구	137
9.5. Divider 도구	140
9.6. Multiplier/Divider 도구	141
9.7. 지연 도구	144
9.8. 사용자 작업 도구 및 사용자 출력 레지스터	146
9.9. 이벤트 입력 도구	150
9.10. C2C 링크 동기화 도구	151

9.1. I/O 도구 상자 소개

I/O 도구 상자는 입력 가능한 I/O 라인에 적용된 외부 신호로부터 이벤트 펄스 스트림을 빌드하는 구성 가능한 배열의 도구입니다.

I/O 도구 상자 컨텍스트



I/O 도구 상자 컨텍스트 블록 다이어그램

Coaxlink 카드 당 하나의 I/O 도구 상자 인스턴스만 있습니다. 그것은 GenTL 인터페이스 모듈에 속합니다.

입력 가능한 I/O 라인 (*Input Lines*)과 선택된 이벤트 소스 (*Input Events*)는 *Input Events* I/O 도구 박스를 공급합니다.

모든 I/O 도구 상자 도구는 하나 이상의 (또는 그 이상의) 이벤트 스트림(I/O 도구 상자 이벤트)을 생성합니다.

I/O 도구 상자 이벤트 스트림은 카드의 다양한 소비자에게 배포됩니다.

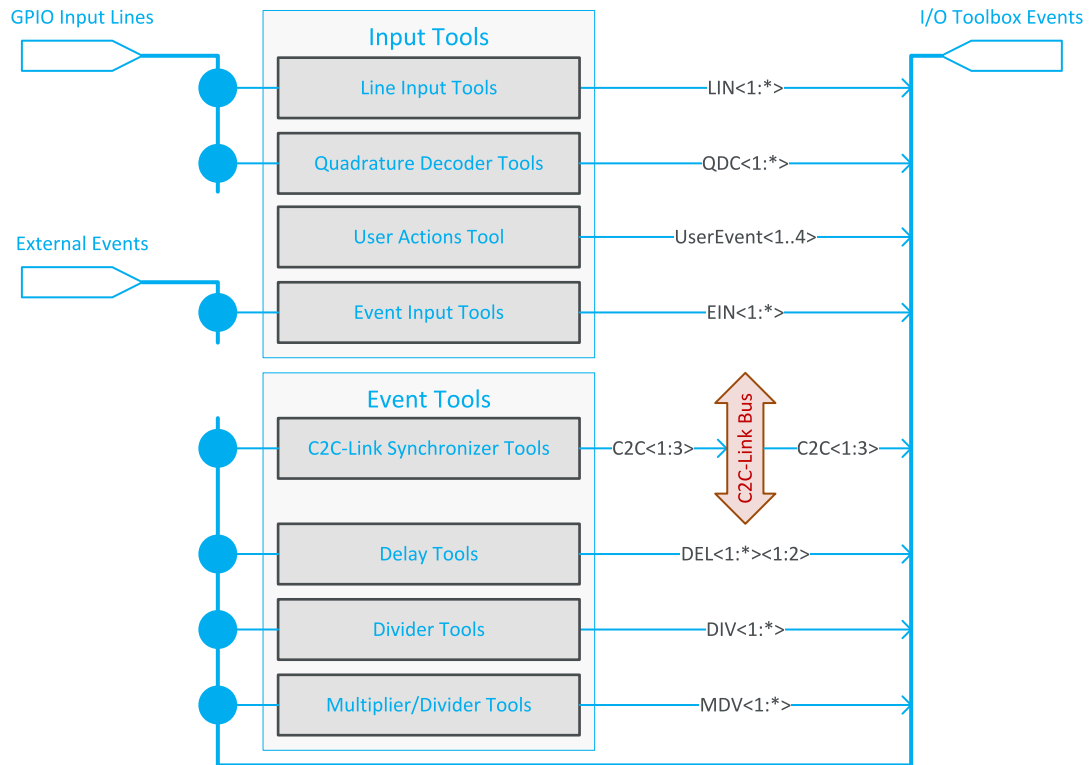
모든 I/O 도구 상자 이벤트는 다음과 같이 카드의 모든 GenTL 장치 모듈에서 사용할 수 있습니다.

- 하드웨어 사이클 트리거 소스.
- 하드웨어 스타트-오브-스캔 트리거 소스.
- 하드웨어 엔드-오브-스캔 트리거 소스.

모든 I/O 도구 상자 이벤트 스트림은 32비트 이벤트 카운터와 연관됩니다.

I/O 도구 상자 구조

I/O 도구 상자는 입력 가능한 I/O 라인에 적용된 외부 신호로부터 이벤트 펄스 스트림을 빌드하는 구성 가능한 배열의 도구입니다.



I/O 도구 상자 구조 다이어그램

I/O 도구 상자 입력 도구

I/O 도구 상자 입력 도구는 입력 가능 GPIO 라인과 I/O 도구 상자 외부에서 선택한 하드웨어 이벤트 소스 세트로 공급됩니다.

하나의 I/O 도구 상자 이벤트 스트림을 전달합니다.

입력 도구에는 다음과 같은 네 가지 유형이 있습니다.

1. 단일 GPIO 입력 라인을 사용하는 센서 및 감지기와 함께 사용하는 *라인 입력 도구*.
2. 2개의 GPIO 입력 라인을 사용하는 직교 모션 엔코더와 함께 사용하는 *쿼드러처 디코더 도구*.
3. 응용 프로그램 소프트웨어가 사용자 이벤트를 생성하는 데 사용하는 *사용자 작업 도구*.
4. 라인 스캔 데이터 전달 응용 프로그램에서 사용하기 위한 *이벤트 입력 도구*.

GPIO 입력 라인 상호 연결 매트릭스는 모든 GPIO 입력 라인에 의해 모든 라인 톨을 공급할 수 있게 합니다.

외부 이벤트 상호 연결 매트릭스는 외부 이벤트 소스가 이벤트 입력 도구를 제공 할 수 있게 합니다.



참고 현재 커넥터 A의 CoaXPress 호스트 인터페이스 GPIO 메시지 수신기만 사용할 수 있습니다.

I/O 도구 상자 이벤트 도구

I/O 도구 상자 이벤트 도구는 하나 이상의 (또는 그 이상의) I/O 도구 상자 이벤트 스트림으로 제공되고 하나 이상의 (또는 그 이상의) I/O Toolbox 이벤트 스트림을 전달할 수 있습니다.

이벤트 도구에는 다음과 같은 세 가지 유형이 있습니다:

1. *디바이더 도구*는 입력 스트림의 D 이벤트에서 1개의 이벤트를 유지하여 이벤트 스트림을 생성합니다.
2. *Multiplier/Divider 도구*는 입력 스트림의 D 이벤트마다 M개의 이벤트를 갖는 이벤트 스트림을 생성합니다.
3. *지연 도구*는 구성 가능한 기간 또는 구성 가능한 동작 엔코더 틱 수에 따라 하나 또는 두 개의 스트림 이벤트를 지연시킵니다.

내부 상호 연결 매트릭스를 사용하면 모든 이벤트 도구가 모든 I/O 도구 상자 이벤트 스트림에 의해 공급될 수 있습니다.

도구를 연쇄적으로 연결하여 도구 체인을 만들 수 있습니다:

도구 체인은 항상 라인 도구로 시작합니다.

(라인 또는 이벤트) 도구는 0개, 1개 또는 여러 개의 이벤트 도구를 구동할 수 있습니다.

9.2. I/O Toolbox 구성

I/O Toolbox의 구성은 제품 및 펌웨어 유형에 따라 다릅니다:

1630 Coaxlink Mono

펌웨어 변형	#LIN	#QDC	#DIV	#MDV	#DEL	#UAS	#EIN
1-카메라	8	1	1	1	2	1	0

1631 Coaxlink Duo

펌웨어 변형	#LIN	#QDC	#DIV	#MDV	#DEL	#UAS	#EIN
1-카메라	8	1	1	1	2	1	0
1-카메라, 라인 스캔	8	1	1	1	2	1	0
2-카메라	8	2	2	2	2	1	0
2-카메라, 라인 스캔	8	2	2	2	2	1	0

1632 Coaxlink Quad

펌웨어 변형	#LIN	#QDC	#DIV	#MDV	#DEL	#UAS	#EIN
1-카메라	8	1	1	1	2	1	0
1-카메라, 라인 스캔	8	1	1	1	2	1	0
2-카메라	8	2	2	2	2	1	0

1633 Coaxlink Quad G3

펌웨어 변형	#LIN	#QDC	#DIV	#MDV	#DEL	#UAS	#EIN
1-카메라	8	1	1	1	2	1	0
1-카메라, 라인 스캔	8	1	1	1	2	1	0
1-slm- 카메라	8	1	1	1	2	1	0
1-sls-카메라	8	1	1	1	2	1	0
2-카메라	8	2	2	2	2	1	0
2-카메라, 라인 스캔	8	2	2	2	2	1	0
3-카메라	8	2	2	2	2	1	0
4-카메라	8	4	4	4	4	1	0
4-카메라, 라인 스캔	8	4	4	4	4	1	0
1-카메라, 4-데이터 스트림	8	1	1	1	2	1	0

1633-LH Coaxlink Quad G3 LH

펌웨어 변형	#LIN	#QDC	#DIV	#MDV	#DEL	#UAS	#EIN
1-카메라	8	1	1	1	2	1	0
1-카메라, 라인 스캔	8	1	1	1	2	1	0
1-slm- 카메라	8	1	1	1	2	1	0
1-sls-카메라	8	1	1	1	2	1	0
2-카메라	8	2	2	2	2	1	0
2-카메라, 라인 스캔	8	2	2	2	2	1	0
3-카메라	8	2	2	2	2	1	0
4-카메라	8	4	4	4	4	1	0
4-카메라, 라인 스캔	8	4	4	4	4	1	0
1-카메라, 4-데이터 스트림	8	1	1	1	2	1	0

1629 Coaxlink Duo PCIe/104-EMB

펌웨어 변형	#LIN	#QDC	#DIV	#MDV	#DEL	#UAS	#EIN
1-카메라	8	1	1	1	2	1	0
1-카메라, 라인 스캔	8	1	1	1	2	1	0
2-카메라	8	2	2	2	2	1	0

1634 Coaxlink Duo PCIe/104-MIL

펌웨어 변형	#LIN	#QDC	#DIV	#MDV	#DEL	#UAS	#EIN
1-카메라	8	1	1	1	2	1	0
1-카메라, 라인 스캔	8	1	1	1	2	1	0
2-카메라	8	2	2	2	2	1	0

1635 Coaxlink Quad G3 DF

펌웨어 변형	#LIN	#QDC	#DIV	#MDV	#DEL	#UAS	#EIN
1-카메라	8	1	1	1	2	1	0
1-df-카메라	8	1	1	1	2	1	0
1-카메라, 라인 스캔	8	1	1	1	2	1	0
1-df-카메라, 라인 스캔	8	1	1	1	2	1	2

1637 Coaxlink Quad 3D-LLE

펌웨어 변형	#LIN	#QDC	#DIV	#MDV	#DEL	#UAS	#EIN
1-카메라	8	1	1	1	2	1	0

3602 Coaxlink Octo

펌웨어 변형	#LIN	#QDC	#DIV	#MDV	#DEL	#UAS	#EIN
1-카메라	8	1	1	1	2	1	0
1-카메라, 라인 스캔	8	1	1	1	2	1	0
2-카메라	8	1	1	1	2	1	0
2-카메라, 라인 스캔	8	1	1	1	2	1	0
4-카메라	8	1	1	1	2	1	0
4-카메라, 라인 스캔	8	4	4	4	4	1	0
5-카메라	8	1	1	1	2	1	0
8-카메라	8	1	1	1	2	1	0

3603 Coaxlink Quad CXP-12

펌웨어 변형	#LIN	#QDC	#DIV	#MDV	#DEL	#UAS	#EIN
1-카메라	8	1	1	1	2	1	0
1-카메라, 라인 스캔	8	1	1	1	2	1	0
4-카메라	8	4	4	4	4	1	0

3620 Coaxlink Quad CXP-12 JPEG

펌웨어 변형	#LIN	#QDC	#DIV	#MDV	#DEL	#UAS	#EIN
4-카메라	8	4	4	4	4	1	0

3621-LH Coaxlink Mono CXP-12 LH

펌웨어 변형	#LIN	#QDC	#DIV	#MDV	#DEL	#UAS	#EIN
1-카메라	8	1	1	1	2	1	0

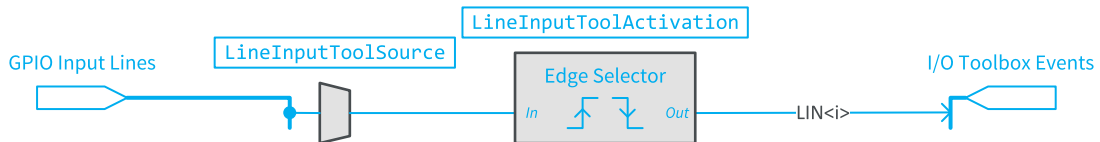
3622 Coaxlink Duo CXP-12

펌웨어 변형	#LIN	#QDC	#DIV	#MDV	#DEL	#UAS	#EIN
1-카메라	8	1	1	1	2	1	0
2-카메라	8	2	2	2	2	1	0

9.3. 라인 입력 도구

도구 이름	짧은 이름	입력 수/유형	출력 개수/유형/이름
라인 입력 도구	LIN	GPIO 입력 라인 1개	1 이벤트 스트림: LIN<i>

도표



LIN 도구 기능 및 배선도

모든 입력 가능 GPIO 라인을 입력 소스로 선택할 수 있습니다.

도구는 LIN<i>라는 하나의 I/O 도구 상자 이벤트 스트림을 제공합니다.

작동

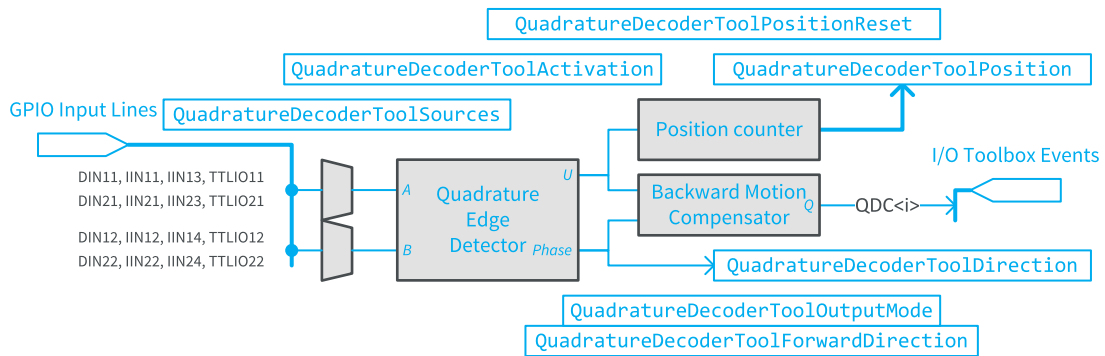
라인 입력 도구는 **LineInputToolSource**에 의해 선택된 I/O 컨트롤 블록에 의해 전달된 **LineInput** 신호의 상승 또는 하강 에지를 감지합니다.

라인 입력 도구는 **LineInputToolActivation** 설정에 따라 모든 상승 또는 하강 에지 또는 둘 모두에서 하나의 이벤트를 전달합니다.

9.4. 쿼드러처 디코더 도구

도구 이름	짧은 이름	입력 수/유형	출력 개수/유형/이름
쿼드러처 디코더 도구	QDC	2쌍의 I/O 입력 라인	1 이벤트 스트림: QDC<i> 1 상태 비트: QDC<i>Direction 1 상태 단어: QDC<i>Position

도표



QDC 도구 기능 및 배선도

쿼드러처 에지 검출기는 위상-직교 모션 엔코더디바이스에 의해 전달된 A 및 B로 명명된 한 쌍의 신호에 의해 공급됩니다.

소스 선택기는 인접한 입력 가능한 GPIO 입력 라인의 다음 쌍을 A / B 입력 소스로 선택할 수 있도록 합니다: DIN11-DIN12, DIN21-DIN22, TTLIO11-TTLIO12, TTLIO21-TTLIO22, IIN11-IIN12, IIN13-IIN14, IIN21-IIN22, 및 IIN23-IIN24.

이 도구는 다음 기능 블록을 포함합니다:

- 쿼드 러처 에지 검출기
- 역방향 모션 보상기
- 32비트 위치 카운터

이 도구는 다음을 제공합니다:

- QDC <i>라는 하나의 I/O Toolbox 이벤트 스트림.
- 모션의 방향을 나타내는 **방향** 상태 비트.
- 위치 오프셋을 나타내는 **위치** 상태 단어.

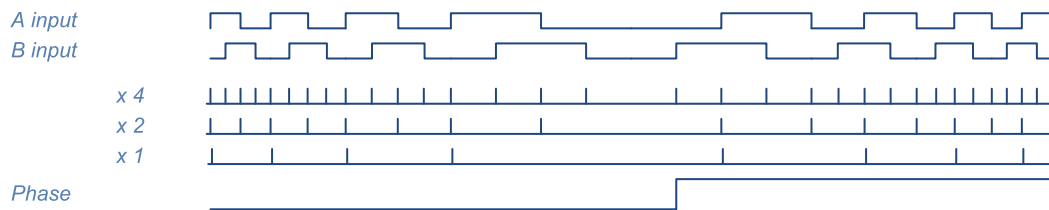
작동

쿼드러처 디코더 도구는 A/B 신호를 디코딩하고 A/B주기마다 1, 2 또는 4개의 이벤트를 전달하며 가능하면 역방향 모션 보정기로 필터링됩니다.

쿼드러처 에지 검출기

쿼드러처 에지 검출기는 A/B 라인의 전이를 분석합니다. 다음을 제공합니다:

- **QuadratureDecoderToolActivation** 설정에 따라 A/B주기마다 1, 2 또는 4개의 이벤트가 있는 U라는 이벤트 스트림
- A와 B 사이의 위상 식별 (A 리드 B 또는 그 반대)



쿼드러처 에지 검출기 파형

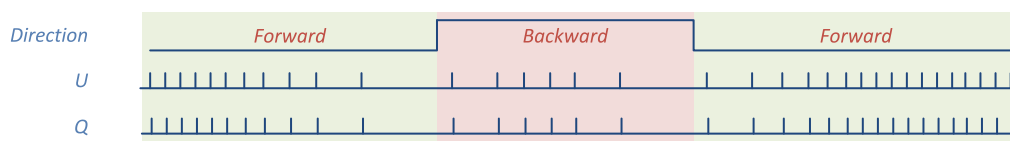
U 스트림은 QDC<i> 출력으로 전달되기 전에 역방향 모션 보상기에 의해 필터링 될 수 있습니다.

위상 지시는 방향 출력으로 전달되기 전에 **QuadratureDecoderToolForwardDirection** 설정에 따라 반전될 수 있습니다.

역방향 모션 보상기

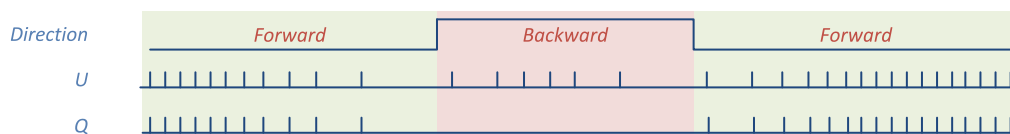
역방향 모션 보상기 (BMC)는 **QuadratureDecoderToolOutputMode** 설정에 따라 U 스트림을 필터링합니다.

Unfiltered로 설정하면 U 스트림의 모든 이벤트가 QDC<i> 출력으로 전달됩니다.



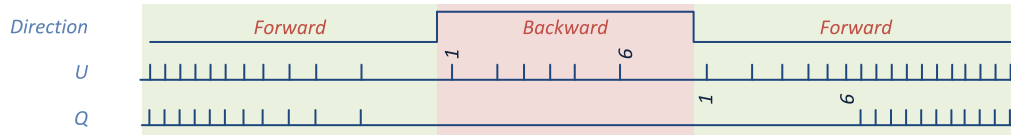
BMC 파형 - 필터링되지 않음

ForwardOnly로 설정하면 순방향에 해당하는 이벤트만 QDC<i> 출력으로 전달됩니다:



BMC 파형 - 순방향 전용

FirstPassForwardOnly로 설정하면 순방향의 첫 번째 전달에 해당하는 이벤트만 QDC<i> 출력으로 전달됩니다.



BMC 파형 - 첫 번째 순방향 전용

위치 카운터

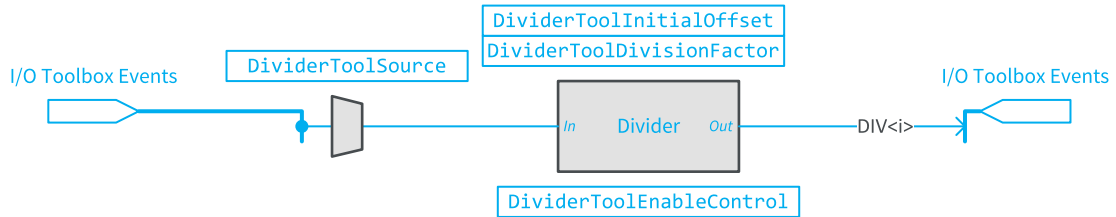
위치 카운터는 순방향에 해당하는 모든 U 이벤트에 대해 1씩 증가하고 역방향에 대해 1씩 감소합니다.

카운터는 **QuadratureDecoderToolPositionReset** 명령을 사용하여 재설정할 수 있습니다.

9.5. Divider 도구

도구 이름	짧은 이름	입력 수/유형	출력 개수/유형/이름
Divider 도구	DIV	1 도구 상자 이벤트 스트림	1 이벤트 스트림: DIV<i></i>

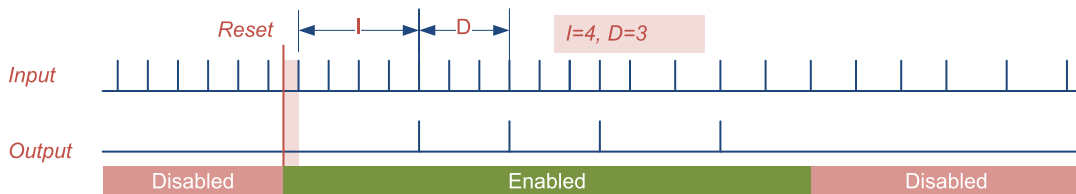
도표



DIV 도구 기능 및 배선도

모든 I/O 도구 상자 이벤트 스트림을 입력 소스로 선택할 수 있습니다. 도구는 DIV<i></i>라는 하나의 I/O 도구 상자 이벤트 스트림을 제공합니다.

작동



DIV 도구 파형

활성화되면 Divider 도구는 모든 D 입력 이벤트를 전달하기 전에 첫 번째 - I - 입력 이벤트를 건너 뛩니다.

나누기 계수 - D는 DividerToolDivisionFactor로 정의됩니다. 기본값은 2이고 1값 범위는 ... 65535입니다.

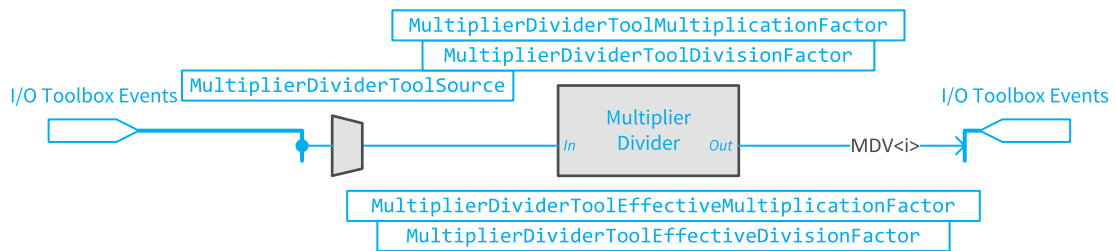
초기 오프셋 - I는 -DividerToolInitialOffset로 정의됩니다. 기본값은 0이고 값 범위는 ... 65535입니다.0

작동 상태는 DividerToolEnableControl에 의해 정의됩니다. 기본값은 Disable입니다.

9.6. Multiplier/Divider 도구

도구 이름	짧은 이름	입력 수/유형	출력 개수/유형/이름
Multiplier/Divider 도구	MDV	1 도구 상자 이벤트 스트림	1 도구 상자 이벤트 스트림: MDV<i>

Multiplier/Divider 도구 배선도



MDV 도구 기능 및 배선도

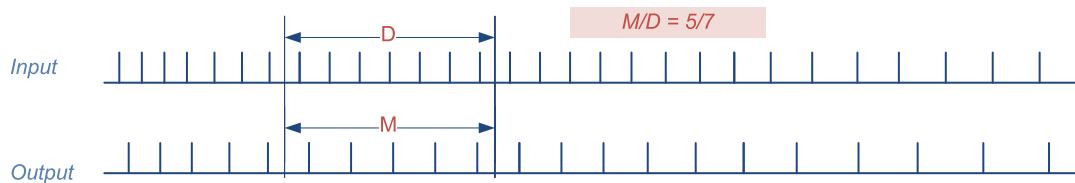
모든 I/O 도구 상자 이벤트 스트림을 입력 소스로 선택할 수 있습니다.
 도구는 MDV<i>라는 하나의 I/O 도구 상자 이벤트 스트림을 제공합니다.

Multiplier/Divider 도구 작동

Multiplier/Divider 도구는 입력 비율을 0.001에서 1000.0 범위의 모든 비율 환산 비율 - RCR - 값으로 곱하거나 나눕니다.

Multiplier/Divider 도구는 모든 연속 입력 이벤트 간의 시간 간격을 측정하고 이에 따라 출력 속도를 조정합니다.

Multiplier/Divider는 주파수 정확성이 있습니다. 출력 주파수는 입력 주파수가 안정적이거나 (또는 천천히 변하는 경우) 입력 주파수에 엄격하게 비례합니다. 이러한 조건에서 Multiplier/Divider는 모든 D 입력 이벤트에 대해 M 이벤트를 전달합니다.



MDV 도구 파형

비율 변환 비율은 두 개의 부동 소수점 숫자의 비율로 구성됩니다.

- M 값은 `MultiplierDividerToolMultiplicationFactor`에 의해 정의됩니다. 기본값은 0.001이고 값 범위는 1.0.1000.0에서입니다.
- D 값은 `MultiplierDividerToolDivisionFactor`에 의해 정의됩니다. 기본값은 1.0이고 값 범위는 0.001에서 1000.0입니다.

유효 곱셈 및 나눗셈 인자는 각각 `MultiplierDividerToolMultiplicationFactor`과 `MultiplierDividerToolDivisionFactor`에 의해 보고됩니다.



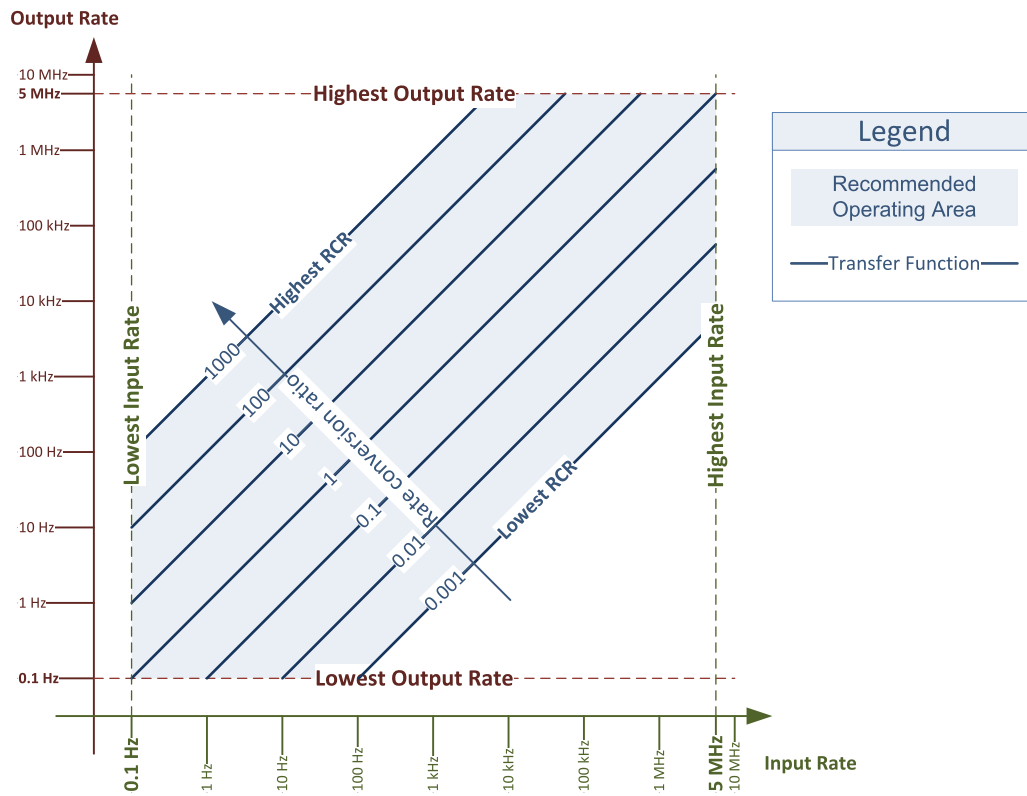
참고 유효 값은 지정된 값과 약간 다를 수 있습니다. 그러나 RCR 상대 오차는 무시할 만합니다(1/1000 미만).



참고 입력 이벤트 스트림의 주파수 변동은 입력 이벤트 스트림의 1주기의 대기 시간으로 출력 이벤트 스트림에 보고됩니다. 이러한 대기 시간은 출력 이벤트 스트림에서 일부 위상 오류를 유발합니다. 입력 이벤트 스트림의 위상은 입력 주파수가 증가하면 지연되고 입력 주파수가 증가하면 그 반대로 마찬가지로입니다. 따라서 Multiplier/Divider는 위상이 정확하지 않습니다.

Multiplier/Divider 도구 작동 제한

특성	심벌	최소	최대
비율 환산율	RCR	0.001	1000
입력 비율	f_{IN}	0.1 Hz	5 MHz
출력 비율	f_{OUT}	0.1 Hz	5 MHz

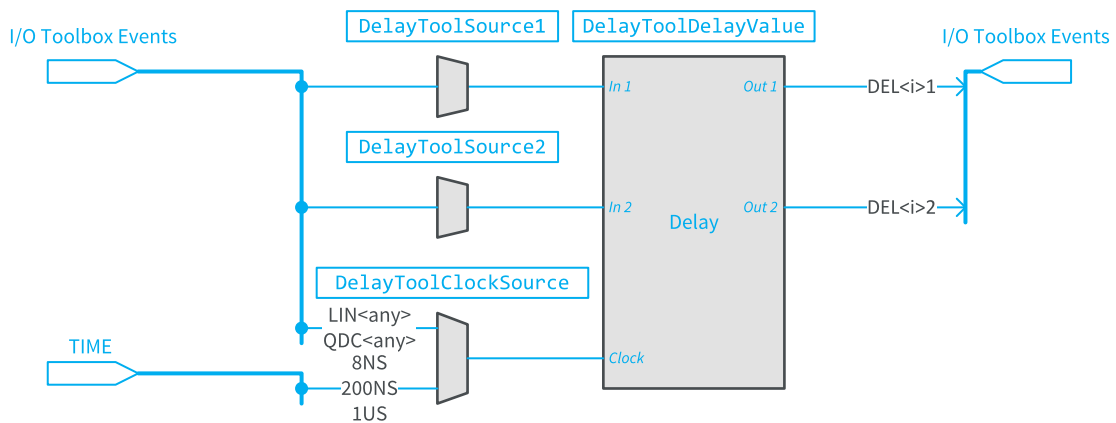


MDV 도구 작동 한계 도표

9.7. 지연 도구

도구 이름	짧은 이름	입력 수/유형	출력 개수/유형/이름
지연 도구	DEL	2 도구 상자 이벤트 스트림 1 클럭 신호	2 도구 상자 이벤트 스트림: DEL<i>1, DEL<i>2

도표



DEL 도구 기능 및 배선도

모든 I/O 도구 상자 이벤트 스트림을 입력 1 소스로 선택할 수 있습니다.

모든 I/O 도구 상자 이벤트 스트림을 입력 2 소스로 선택할 수 있습니다.

이 도구는 두 개의 I/O 도구 상자 이벤트 스트림을 제공합니다. 도구 인스턴스 <i>의 출력은 DEL<i>1 및 로 지정됩니다. DEL<i>2

작동

입력 (In1 및 In2)에 적용된 이벤트 스트림은 구성 가능한 클럭 틱 이벤트 수 이후에 해당 출력 (Out1 및 Out2)에 복제됩니다.

소스는 각각 DelayToolSource1 및 DelayToolSource2로 선택됩니다.

두 채널 모두 동일한 지연이 적용됩니다. 공통 지연은 DelayToolDelayValue로 정의됩니다.

두 채널 모두 동일한 지연이 적용됩니다. 클럭 소스는 DelayToolClockSource로 정의됩니다. 타임 베이스 또는 라인 도구 이벤트 스트림일 수 있습니다.

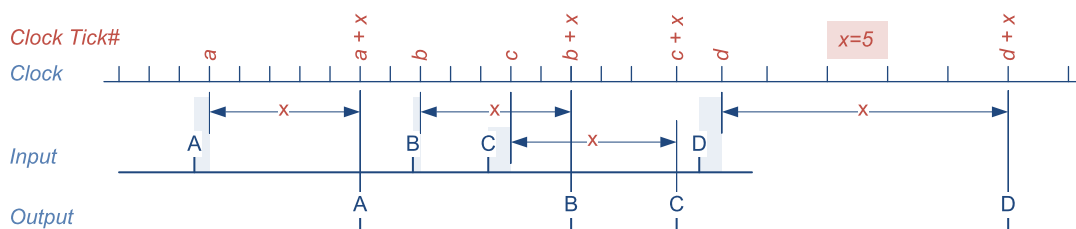
타임 베이스를 선택하면 시간 지연 기능이 구현됩니다. 사용 가능한 시간 기준은 다음과 같습니다:

- **8NS:** 40 나노초에서 최대 134 밀리초, 8 나노초 단위의 지연을 허용하는 125 MHz 고정밀 표준 시간축
- **200NS:** 200 나노초부터 최대 3.35 초, 200 나노초 단위로 지연시킬 수 있는 5MHz 고정밀 표준 시간축
- **1US:** 1 마이크로 초의 단계별로 1 마이크로 초에서 최대 16.7 초까지의 지연을 허용하는 1 MHz 고정밀 표준 시간축

라인 도구 이벤트 스트림을 선택하면 라인 도구가 모션 인코더 장치에 의해 공급될 때 위치 오프셋 기능이 구현됩니다. 사용 가능한 라인 입력 도구 또는 구적 디코더 도구를 지연 클럭 소스로 사용할 수 있습니다. 지연 범위는 1부터 최대 16,777,215 개의 이벤트입니다.



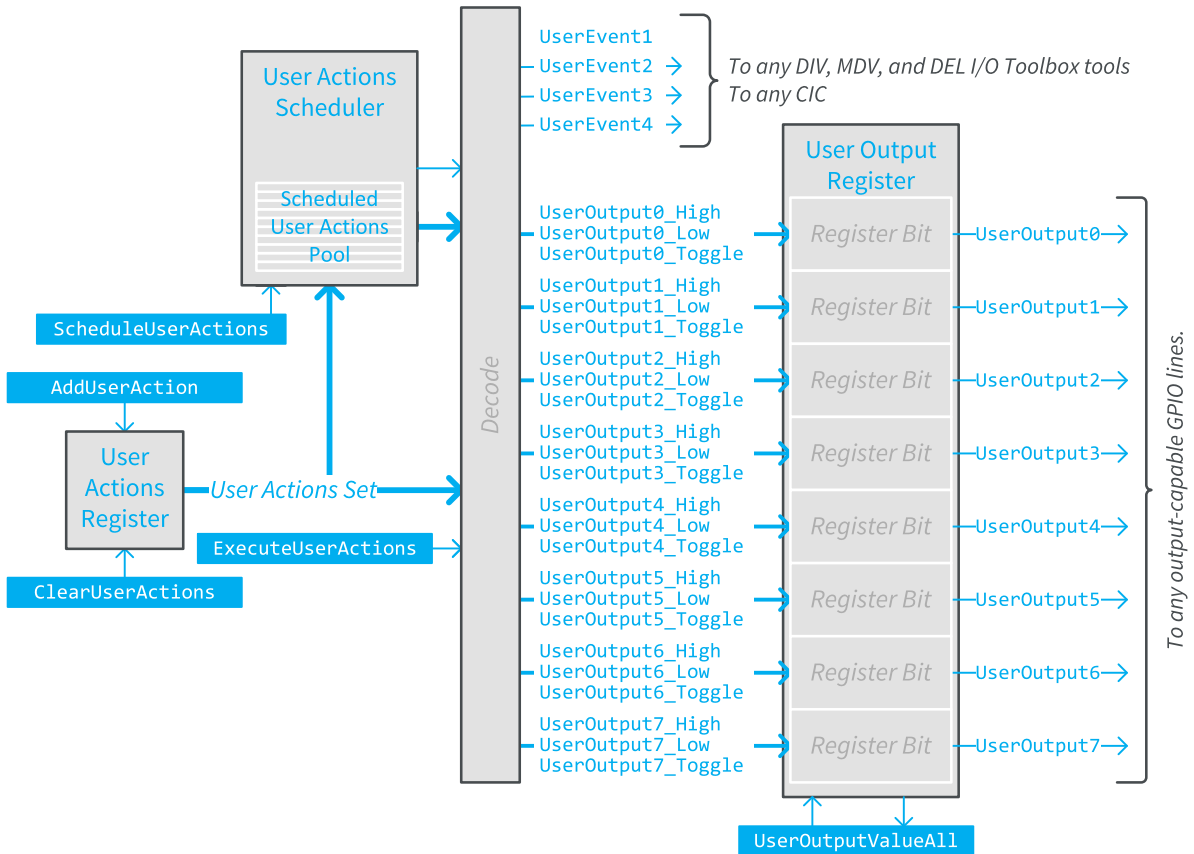
경고 지연 도구는 지연 라인으로 작동합니다. 이전 이벤트가 아직 전달되지 않은 상태에서 새 이벤트를 수락할 수 있습니다. 지연 도구는 모든 채널에 대해 글로벌로 최대 16개의 별개 이벤트를 녹화할 수 있습니다.



DEL 도구 파형

9.8. 사용자 작업 도구 및 사용자 출력 레지스터

소개



UAS 도구 기능 및 배선 블록 다이어그램

사용자 작업 도구 (UAS)를 사용하면 응용 프로그램 소프트웨어에서 다음과 같은 사용자 작업을 수행할 수 있습니다:

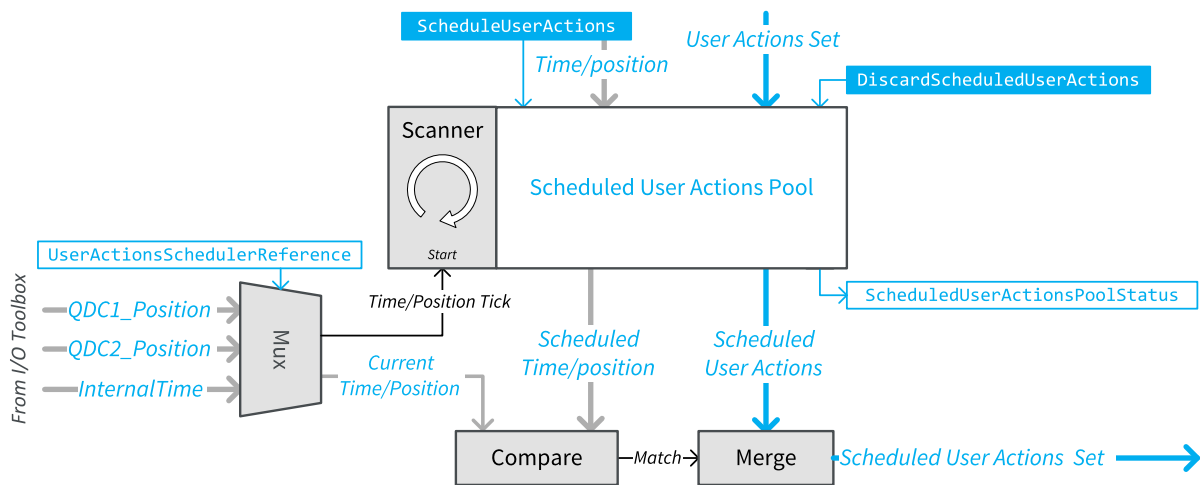
- 모든 사용자 이벤트 소스에서 이벤트를 생성.
- 용자 출력 레지스터의 모든 비트를 높게 설정.
- 사용자 출력 레지스터의 임의의 비트를 낮게 설정.
- 사용자 출력 레지스터의 비트를 토글링.

`ClearUserActions` 및 `AddUserActions` GenICam 기능을 통해 사용자 응용 프로그램은 동시에 실행될 모든 동작으로 구성된 사용자 작업 집합을 정의할 수 있습니다.

사용자 작업 세트를 정의한 후 응용 프로그램 소프트웨어에는 실행을 위한 두 가지 옵션이 있습니다:

- **ExecuteUserActions** GenICam 기능을 사용하여 사용자 작업 집합에 정의된 모든 동작을 즉시 실행합니다.
- **ScheduleUserActions** GenICam 기능을 사용하여 사용자 작업 집합의 실행을 사용자 작업 스케줄러에 위임합니다.

사용자 작업 스케줄러



사용자 작업 스케줄러 기능 다이어그램

UAS 기능 블록을 사용하면 응용 프로그램 소프트웨어가 작업 실행을 연기할 수 있습니다.

스케줄러 참조 설정

사용자 작업 스케줄러를 사용하기 전에 사용자 응용 프로그램은 32 비트 참조 카운터를 선택해야 합니다. 이는 다음 세 가지 값 중 하나를 `UserActionsSchedulerReference` GenICam 기능에 할당하여 수행됩니다:

- `InternalTime`는 *Coaxlink* 카드 *현재 시간*을 선택합니다: 1 마이크로초마다 1씩 증가하고 약 4,294,967,295의 최대값에 도달하면 약 71분 후에 랩어라운드하는 단조로운 타임베이스입니다.
- `QDC1Position` 및 `QDC2Position`는 쿼드러처 디코더 도구 QDC1 및 QDC2의 *위치 카운터*를 각각 선택합니다.



경고 위치 참조로 UAS를 올바르게 작동 시키려면 위치 카운터가 매 순간마다 증가하지 않고 단조롭게 증가해야 합니다.



참고 QDC 위치 카운터의 단조 증가를 보장하려면:

- 물체가 정방향으로 움직일 때 카운터가 증가하도록 `QuadratureDecoderToolForwardDirection`를 올바르게 설정하십시오.
- 뒤로 이동이 있으면 `QuadratureDecoderToolOutputMode`를 `ForwardOnly` 또는 `FirstPassForwardOnly`로 설정하여 위치 카운터가 감소하지 않도록 합니다.

예약된 사용자 작업 풀 작업

예약된 사용자 작업 풀은 예약된 사용자 작업 집합이 사용자 응용 프로그램에 저장되는 메모리 영역입니다. 풀에는 64개의 위치가 있습니다.

새 예약된 사용자 작업을 풀에 추가하려면 사용자 응용 프로그램에서 다음을 수행해야 합니다.

1. `ScheduledUserActionsPoolStatus` GenICam 기능의 값을 가져와 최소한 하나의 사용 가능한 위치가 있는지 확인.
2. 사용자 작업 집합 정의,
3. 동작이 실행될 때 시간/위치 32비트 값을 결정,
4. 이 값을 `ScheduledUserActions` GenICam 기능으로 설정.

예약된 사용자 작업은 실행될 때 풀에서 제거됩니다.

풀은 `DiscardScheduledUserActions` GenICam 명령을 실행하여 언제든지 지울 수 있습니다.

스캐너 작업 및 예약된 작업 실행

32비트 (시간 또는 위치) 참조 카운터가 증가할 때마다 스캐너는 모든 위치를 읽고 예정된 기준 시간/위치를 현재 시간/위치 카운트 값과 비교합니다.

값이 동일하면 예약된 사용자 작업 세트가 검사가 끝날 때 실행되도록 선택되고 풀에서 제거됩니다.

실행을 위해 여러 세트가 선택되면 해당 세트가 병합됩니다.



참고 동일한 사용자 출력 레지스터 비트에 설정된 하위 및 상위 상위 동작을 병합하면 토글 동작이 발생합니다.

검사가 끝나면 병합된 선출 작업이 동시에 실행됩니다. 기준 틱에서 선출된 행동의 실행까지의 시간 지연은 매우 작고(서브 마이크로 초) 일정합니다.

사용자 이벤트 소스

UserEvent1 ... UserEvent4라는 사용자 이벤트 소스가 있습니다.

모든 사용자 이벤트를 사용할 수 있습니다:

- Cycle Trigger는 모든 카메라 및 조명 컨트롤러에 의해 소스로 사용됩니다.
- Cycle Sequence Trigger는 모든 카메라 및 조명 컨트롤러에 의해 소스로 사용됩니다.
- I/O Toolbox의 모든 Divider 도구로 이벤트 소스로 사용됩니다.
- I/O Toolbox의 모든 Multiplier/Divider 도구를 통한 이벤트 소스로 사용
- I/O Toolbox의 모든 Delay 툴의 두 채널에 대한 이벤트 소스).

사용자 출력 레지스터

Coaxlink 제품은 비트가 UserOutput0 ... UserOutput7인 사용자 출력 레지스터를 제공합니다.



참고 1630 Coaxlink Mono 는 4개의 최하위 비트만 구현합니다!

모든 사용자 출력 레지스터 비트는 출력 가능한 GPIO 라인에 의해 신호 소스로 사용될 수 있습니다.

사용자 응용 프로그램에는 사용자 출력 레지스터 비트의 상태를 정의하는 두 가지 옵션이 있습니다:

- 사용자 작업 옵션을 사용하면 각 비트의 상태를 개별적으로 변경할 수 있습니다.
- UserOutputValueAll에 값을 설정하여 모든 비트의 상태를 정의합니다.

UserOutputValueAll의 값을 얻으면 사용자 응용 프로그램이 모든 비트의 상태를 가져올 수 있습니다.

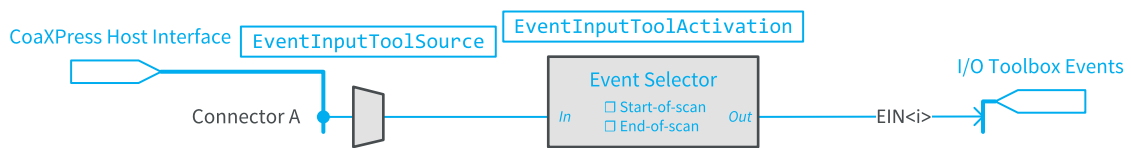
9.9. 이벤트 입력 도구

Applies to:

QuadG3DF

도구 이름	짧은 이름	입력 수/유형	출력 개수/유형/이름
이벤트 입력 도구	EIN	1 이벤트	1 이벤트 스트림: EIN<i>

도표



EIN 도구 기능 및 배선도

CoaXPress 호스트 인터페이스의 커넥터 A는 선택할 수 있는 유일한 이벤트 소스입니다. 도구는 EIN<i>라는 하나의 I/O 도구 상자 이벤트 스트림을 제공합니다.

작동

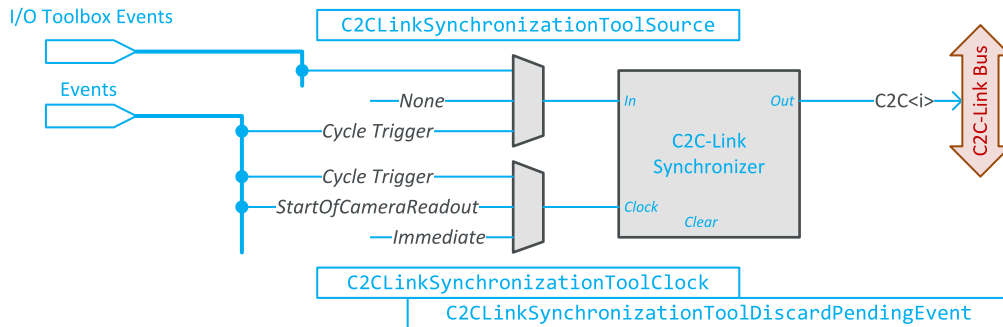
이벤트 입력 도구는 슬레이브 **1635 Coaxlink Quad G3 DF**의 CoaXPress 커넥터 A에서 수신한 사용자 정의 CoaXPress GPIO 메시지를 디코딩합니다.

이벤트 입력 도구는 **EventInputToolActivation** 설정에 따라 "스캔 시작 메시지" 또는 "스캔 종료 메시지" 수신시 하나의 이벤트를 전달합니다.

9.10. C2C 링크 동기화 도구

도구 이름	짧은 이름	입력 수/유형	출력 개수/유형/이름
C2C 링크 SynchronizerTool	C2C	1 또는 2 이벤트 스트림	1 이벤트 스트림: C2C<i>

도표



C2C 도구 기능 및 배선도

C2C 링크 동기화 도구 (C2C) 도구는 하나의 이벤트 스트림을 C2C 링크 버스 드라이버로 전달합니다. 그것은 다음과 같은 블록을 포함합니다:

- 소스 선택기
- 클릭 소스 셀렉터
- 명확한 제어 기능이 있는 이벤트 동기화

작동

소스 선택기

소스 선택기는 도구 입력(In)에 적용된 이벤트 스트림을 선택합니다. 다음과 같은 옵션을 제공합니다:

- C2C1 인스턴스에만 해당: 카메라 및 조명 컨트롤러에 의해 구동되는 **Cycle Trigger** 이벤트 스트림.
- C2C2 및 C2C3 인스턴스에만 해당: 모든 I/O 도구 상자 이벤트.

동기화 제어

클릭 소스 선택기는 이벤트 스트림 동기화를 제어합니다:

- **C2CLinkSynchronizationToolClock**가 **Immediate**로 설정되면 입력(In)에 적용된 이벤트 스트림이 즉시 출력으로 전송됩니다.

- C2C2 및 C2C3 인스턴스에서만: `C2CLinkSynchronizationToolClock`가 `CycleTrigger`로 설정되면 이벤트가 래치되고 다음 `Cycle Trigger` 이벤트까지 지연됩니다.
- C2C2 및 C2C3 인스턴스에서만: `C2CLinkSynchronizationToolClock`가 `StartOfCameraReadout`로 설정되면 이벤트는 다음 `Start of Camera Readout` 이벤트까지 래치되고 지연됩니다.

`C2CLinkSynchronizationToolDiscardPendingEvent` 명령은 수신되었지만 전달되지 않은 이벤트를 버립니다.



참고영역 스캔 펌웨어 변형은 2개의 C2C 도구 인스턴스를 제공합니다. 라인 스캔 펌웨어 변형은 3가지 인스턴스를 제공합니다!

10. 이벤트 신호 및 카운팅

10.1. 소개	154
10.2. 맞춤 이벤트 소스	158
10.3. 이벤트 특정 컨텍스트 데이터	162
10.4. GenTL 신호 정보	164

10.1. 소개

간단한 설명

Coaxlink 제품은 강력한 이벤트 관리 기능을 갖추고 있어 다양한 이벤트 발생을 애플리케이션에 알릴 수 있습니다.

GenTL 표준 `EVENT_NEW_BUFFER` 이벤트 외에도 Coaxlink GenTL 프로듀서는 다양한 사용자 정의 이벤트 소스를 제공합니다.

이벤트 소스는 기능 블록과 해당 모듈이 속한 GenTL 모듈에 따라 유형별로 그룹화됩니다.

각 사용자 지정 이벤트 원본은 발생 횟수를 계산하는 카운터와 연결됩니다.

통지된 각 사용자 정의 이벤트에 대해 다음 이벤트 컨텍스트 데이터가 기록되어 애플리케이션에서 사용 가능하게 됩니다.

- 이벤트 소스 식별자
- 타임 스탬프 (마이크로 초로 표시)
- 3개의 사용자 정의 컨텍스트 데이터

각 개별 이벤트 소스를 구성할 수 있습니다:

- 이벤트 알림은 사용 또는 사용 안 함으로 설정할 수 있습니다.
- 각 사용자 정의 컨텍스트 데이터의 내용.

이벤트 데이터는 이벤트 대기열 버퍼에 임시 저장됩니다. Coaxlink 이벤트 큐 버퍼에서 하나 이상의 이벤트 항목이 사용 가능하다는 것을 인터럽트 메커니즘을 사용하여 Coaxlink 드라이버에 알립니다.

Coaxlink 드라이버는 비동기 이벤트 발생을 애플리케이션 소프트웨어에 보고하기 위해 GenTL 신호 메커니즘을 구현합니다.

EGrabber API는 3 가지 콜백 스레딩 모델을 제공합니다:

- *CallbackOnDemand*: 콜백이 호출되는 시기와 방법을 완벽하게 조절할 수 있는 가장 간단한 모델입니다. 이벤트는 필요에 따라 처리됩니다.
- *CallbackSingleThread*: 이 모델은 콜백에 대한 이벤트를 전용 스레드 컨텍스트에서 연대순으로 순차적으로 전달합니다. 이벤트는 사용 가능한 즉시 자동으로 처리됩니다.
- *CallbackMultiThread*: 이 모델은 별도의 스레드 (이벤트 DATA 유형 당 하나의 스레드)에서 콜백에 이벤트를 전달합니다. 이벤트는 사용 가능한 즉시 자동으로 처리됩니다.

이벤트 유형

GenTL은 유형 및 해당 객체 모듈에 따라 이벤트를 식별합니다.

표준 이벤트 유형

Coaxlink 드라이버는 GenTL Consumer 애플리케이션에 등록하기 위해 다음 표준 이벤트 유형을 구현합니다:

GenTL 표준 이벤트 유형	GenTL 모듈	설명
EVENT_NEW_BUFFER	데이터 스트림	새로 채워진 버퍼에 대한 알림.

맞춤 이벤트 유형

표준 이벤트 유형 외에도 GenTL 사양은 맞춤 이벤트 유형을 위한 공간을 제공합니다.

사용자 지정 이벤트 유형은 GenTL Producer 구현 (이 경우 Coaxlink 드라이버)에만 해당됩니다.

Coaxlink 드라이버는 GenTL Consumer 애플리케이션에 등록하기 위해 다음과 같은 사용자 지정 이벤트 유형을 구현합니다.

이벤트 유형	모듈	설명
EVENT_CUSTOM_IO_TOOLBOX	인터페이스 (장치)	I/O 도구 상자 이벤트 알림
EVENT_CUSTOM_CXP_INTERFACE	인터페이스 (장치)	CoaXPress 호스트 인터페이스 이벤트 알림
EVENT_CUSTOM_CIC	장치	카메라 및 조명 제어 이벤트 알림
EVENT_CUSTOM_DATASTREAM	데이터 스트림	CoaXPress 데이터 스트림 이벤트 알림



참고 이벤트 모듈 EVENT_CUSTOM_IO_TOOLBOX 및 EVENT_CUSTOM_CXP_INTERFACE 이벤트 유형도 장치 모듈에 등록할 수 있습니다.



참고 맞춤 이벤트 유형은 일반적인 항목입니다. 각각은 여러 이벤트 소스를 수집합니다.

참조: "맞춤 이벤트 소스" 페이지 158는 철저한 목록입니다.

맞춤 이벤트 카운터

32 비트 카운터는 모든 사용자 정의 이벤트 소스와 연관됩니다.

카운터를 비활성화할 수 없습니다. 최대값인 4 294 967 295 ($2^{32} - 1$)에 도달하면 0으로 바뀝니다.

언제든지 사용자 애플리케이션은 다음을 수행할 수 있습니다:

- 선택한 이벤트 소스의 카운트 값을 읽습니다.
- 선택한 이벤트 소스의 카운터를 재설정합니다.
- 모듈의 모든 이벤트 소스 카운터를 재설정합니다.

카운트 값은 모든 이벤트 소스에 의해 사용자 정의 컨텍스트 데이터로 사용될 수도 있습니다.

맞춤 이벤트 설정

각 개별 이벤트 소스를 구성할 수 있습니다.

언제든지 사용자 애플리케이션은 다음을 수행할 수 있습니다:

- 선택한 이벤트 소스의 알림을 사용 또는 사용하지 않도록 설정합니다.
- 모듈의 모든 이벤트 소스에 대한 알림을 사용 또는 사용하지 않도록 설정합니다.
- 선택한 이벤트 소스의 각 사용자 정의 컨텍스트 데이터의 내용을 정의합니다.

알림

기본적으로 모든 알림은 비활성화됩니다.

애플리케이션 소프트웨어는 애플리케이션 요구 사항에 따라 이벤트 알림 필터를 설정해야 합니다.

알림 필터 구성의 설정은 이벤트 카운팅 기능을 방해하지 않고 언제든지 수정할 수 있습니다.

컨텍스트 데이터

이벤트 컨텍스트 데이터의 마지막 32 비트 컨텍스트 데이터 워드는 다음과 같이 설정할 수 있습니다:

- 이벤트 특정 데이터.
- 이벤트 발생 시간에 샘플링된 I/O 라인의 상태
- 이벤트 카운터의 카운트 값.
- 임의의 직교 디코더 (QDC) 위치 카운터의 카운트 값.

일부 이벤트 소스는 추가 옵션을 제공합니다.

참조: "이벤트 특정 컨텍스트 데이터" 페이지 162

10.2. 맞춤 이벤트 소스

데이터 스트림 이벤트 소스 (데이터 스트림 모듈)

EVENT_CUSTOM_DATASTREAM

이벤트 소스	설명
DATASTREAM_START_OF_CAMERA_READOUT	이미지 프레임의 첫 번째 픽셀 데이터는 온보드 FIFO 버퍼에 기록됩니다. 영역 스캔 펌웨어 변형에만 적용됩니다.
DATASTREAM_END_OF_CAMERA_READOUT	이미지 프레임의 마지막 픽셀 데이터는 온보드 FIFO 버퍼에 기록됩니다. 영역 스캔 펌웨어 변형에만 적용됩니다.
DATASTREAM_START_OF_SCAN	이미지 스캔 첫 번째 픽셀 데이터는 온보드 FIFO 버퍼에 기록됩니다. 라인 스캔 펌웨어 변형에만 적용됩니다.
DATASTREAM_END_OF_SCAN	이미지 스캔 마지막 픽셀 데이터는 온보드 FIFO 버퍼에 기록됩니다. 라인 스캔 펌웨어 변형에만 적용됩니다.
DATASTREAM_REJECTED_FRAME	이미지 프레임이 거부됩니다 (온보드 FIFO 버퍼가 꽉 찼습니다) 영역 스캔 펌웨어 변형에만 적용됩니다.
DATASTREAM_REJECTED_SCAN	이미지 스캔이 거부됩니다 (온보드 FIFO 버퍼가 꽉 찼습니다) 라인 스캔 펌웨어 변형에만 적용됩니다.

I/O 도구 상자 사용자 지정 이벤트 (인터페이스 모듈)

EVENT_CUSTOM_IO_TOOLBOX 이벤트 소스

이벤트 소스	설명
IO_TOOLBOX_LIN1	라인 입력 도구 1 - 이벤트 출력
IO_TOOLBOX_LIN2	라인 입력 도구 2 - 이벤트 출력
IO_TOOLBOX_LIN3	라인 입력 도구 3 - 이벤트 출력
IO_TOOLBOX_LIN4	라인 입력 도구 4 - 이벤트 출력
IO_TOOLBOX_LIN5	라인 입력 도구 5 - 이벤트 출력
IO_TOOLBOX_LIN6	라인 입력 도구 6 - 이벤트 출력
IO_TOOLBOX_LIN7	라인 입력 도구 7 - 이벤트 출력
IO_TOOLBOX_LIN8	라인 입력 도구 8 - 이벤트 출력
IO_TOOLBOX_QDC1	복소 디코더 도구 1 - 이벤트 출력
IO_TOOLBOX_QDC1_DIR	복소 디코더 도구 1 - 방향 변경
IO_TOOLBOX_QDC2	복소 디코더 도구 2 - 이벤트 출력
IO_TOOLBOX_QDC2_DIR	복소 디코더 도구 2 - 변경된 방향
IO_TOOLBOX_DIV1	디바이더 도구 1 - 이벤트 출력
IO_TOOLBOX_DIV2	디바이더 도구 2 - 이벤트 출력
IO_TOOLBOX_MDV1	멀티플라이어/디바이더 1 - 이벤트 출력
IO_TOOLBOX_MDV2	멀티플라이어/디바이더 2 - 이벤트 출력
IO_TOOLBOX_DEL11	지연 도구 1 출력 1 - 이벤트 출력
IO_TOOLBOX_DEL12	지연 도구 1 출력 2 - 이벤트 출력
IO_TOOLBOX_DEL21	지연 도구 2 출력 1 - 이벤트 출력
IO_TOOLBOX_DEL22	지연 도구 2 출력 2 - 이벤트 출력
IO_TOOLBOX_USER_EVENT_1	사용자 이벤트 1
IO_TOOLBOX_USER_EVENT_2	사용자 이벤트 2
IO_TOOLBOX_USER_EVENT_3	사용자 이벤트 3
IO_TOOLBOX_USER_EVENT_4	사용자 이벤트 4
IO_TOOLBOX_EIN1	이벤트 입력 도구 1 - 이벤트 출력
IO_TOOLBOX_EIN2	이벤트 입력 도구 2 - 이벤트 출력



참고적용 가능한 값은 "I/O Toolbox 구성" 페이지 133를 확인하십시오

CoaXPress 호스트 인터페이스 맞춤 이벤트 (인터페이스 모듈)***EVENT_CUSTOM_CXP_INTERFACE***

이벤트 소스	설명
CXP_INTERFACE_CRC_ERROR_CXP_A	CoAXPress 호스트 인터페이스의 연결 A에서 CRC 오류가 감지되었습니다.
CXP_INTERFACE_CRC_ERROR_CXP_B	CoAXPress 호스트 인터페이스의 연결 B에서 CRC 오류가 감지되었습니다.
CXP_INTERFACE_CRC_ERROR_CXP_C	CoAXPress 호스트 인터페이스의 연결 C에서 CRC 오류가 감지되었습니다.
CXP_INTERFACE_CRC_ERROR_CXP_D	CoAXPress 호스트 인터페이스의 연결 A에서 CRC 오류가 감지되었습니다.

카메라 및 조명 컨트롤러 이벤트 소스 (장치)

EVENT_CUSTOM_CIC event sources

이벤트 소스	설명
CIC_CAMERA_TRIGGER_RISING_EDGE	카메라 트리거 출력 신호의 상승 에지 (RC 및 RG 카메라 조절 방법에서 노출 시작)
CIC_CAMERA_TRIGGER_FALLING_EDGE	카메라 트리거 출력 신호의 하강 에지 (RG 카메라 조절 방법의 노출 종료)
CIC_STROBE_RISING_EDGE	스트로브 출력 신호의 상승 에지
CIC_STROBE_FALLING_EDGE	스트로브 출력 신호의 상승 에지
CIC_ALLOW_NEXT_CYCLE	새로운 카메라 사이클을 즉시 시작할 수 있습니다.
CIC_DISCARDED_CIC_TRIGGER	CIC 주기 트리거가 삭제됩니다.
CIC_PENDING_CIC_TRIGGER	CIC 주기 트리거가 기록되지만 실행은 CIC가 준비될 때까지 지연됩니다.
CIC_CXP_TRIGGER_ACK	CoaXPress 호스트-장치 트리거 메시지에 대한 응답으로 긍정적인 확인이 수신됩니다.
CIC_CXP_TRIGGER_RESEND	CoaXPress 호스트-장치 트리거 메시지의 재전송이 실행됩니다.
CIC_TRIGGER	CIC 트리거



참고 GenTL 장치 모듈 당 하나의 카메라 및 조명 컨트롤러 인스턴스가 있습니다. Coaxlink 카드 당 GenTL 장치 모듈의 수는 펌웨어 변형에 의해 정의됩니다.

10.3. 이벤트 특정 컨텍스트 데이터

EVENT_DATA_NUMID_CIC_DISCARDED_CIC_TRIGGER

EVENT_DATA_NUMID_CIC_DISCARDED_CIC_TRIGGER의 EventSpecific 값은 다음 정의에 따라 해석할 수 있는 비트 필드입니다:

비트#	설명
0	원인: 이미지 버퍼가 꽉 찼습니다.
1	원인: 카메라 기가 완료되지 않았습니다.
2	원인: 이미 대기중인 최대 트리거 수입입니다.
3	원인: 데이터 스트림이 활성화되어 있지 않습니다

EVENT_DATA_NUMID_CIC_PENDING_CIC_TRIGGER

EVENT_DATA_NUMID_CIC_PENDING_CIC_TRIGGER의 EventSpecific 값은 다음 정의에 따라 해석될 수 있는 비트 필드입니다:

비트#	설명
0	원인: 이미지 버퍼가 꽉 찼습니다.
1	원인: 카메라 주기가 완료되지 않았습니다

EVENT_DATA_NUMID_DATASTREAM_START_OF_SCAN

EVENT_DATA_NUMID_DATASTREAM_START_OF_SCAN의 EventSpecific 값은 다음 정의에 따라 해석할 수 있는 비트 필드입니다:

비트#	설명
1	원인: 소프트웨어 트리거.
2	원인: 하드웨어 트리거
3	원인: DSStartAcquisition 또는 이전 검사 종료

EVENT_DATA_NUMID_DATASTREAM_END_OF_SCAN

EVENT_DATA_NUMID_DATASTREAM_END_OF_SCAN의 EventSpecific 값은 다음 정의에 따라 해석할 수 있는 비트 필드입니다:

비트#	설명
1	원인: 소프트웨어 트리거.
2	원인: 하드웨어 트리거
3	원인: 원인: 스캔 길이에 도달했습니다
4	원인: DSStopAcquisition
5	원인: 내부 예외 (이미지 버퍼가 거의 가득 찼습니다)

EVENT_DATA_NUMID_DATASTREAM_REJECTED_FRAME

EVENT_DATA_NUMID_DATASTREAM_REJECTED_FRAME의 EventSpecific 값은 다음 정의에 따라 해석될 수 있는 비트 필드입니다:

비트#	설명
0	원인: 이미지 버퍼가 꽉 찼습니다.
1	원인: 데이터 스트림이 활성화되어 있지 않습니다

10.4. GenTL 신호 정보

Coaxlink 드라이버는 GenTL 공급자의 신호 메커니즘을 구현합니다.

이 메커니즘은 이후 간단히 설명됩니다.

참조: 광범위한 설명을 위해 [GenCam GenTL Standard Version 1.4의 34 페이지에서 시작하는 4.2 절](#).

이벤트 등록

출처: *GenTL 사양*

GenTL Consumer가 이벤트에 대해 알기 전에 이벤트 객체를 등록해야 합니다. 열거 프로세스에서 모듈 인스턴스를 만든 후 `GCRegisterEvent()` 함수를 사용하여 이벤트 객체를 만들 수 있습니다. 이 함수는 등록된 이벤트 객체를 식별하는 고유한 `EVENT_HANDLE`을 반환합니다. 등록된 이벤트에 대한 정보를 얻으려면 `EventGetInfo()` 함수를 사용할 수 있습니다.



경고 모듈 및 이벤트 유형별로 등록된 이벤트는 하나만 있어야 합니다!

(...)

`EVENT_HANDLE`이 수집된 후 GenTL 소비자는 함수를 호출하여 이벤트 개체가 신호를 받을 때까지 기다릴 수 있습니다. `EventGetData()` 이벤트가 전달되면 이벤트 객체는 데이터를 전달합니다. 이 데이터는 `EventGetData()`에 대한 호출이 성공했을 때 GenTL Consumer가 제공한 버퍼에 복사됩니다.

알림 및 데이터 검색

출처: *GenTL 사양*

이벤트 오브젝트에 신호가 보내지면 데이터는 특정 시점에서 이벤트 데이터 대기열에 놓입니다. `EventGetData()` 함수를 호출하여 실제 데이터를 검색할 수 있습니다.

(...)

이 기능으로 데이터를 읽을 때 데이터는 대기열에서 제거됩니다. 이후 GenTL Producer 구현은 이벤트 데이터 대기열이 비어 있는지 여부를 확인합니다. 사용 가능한 데이터가 더 많은 경우 이벤트 객체는 신호를 보낸 상태로 유지되고 `EventGetData()`를 호출하면 다음 대기열 항목이 전달됩니다. 그렇지 않으면 이벤트 객체가 신호되지 않은 상태로 재설정됩니다.

(...)

정확한 데이터 유형은 이벤트 유형 및 GenTL Producer 구현에 따라 다릅니다. 데이터는 GenTL 소비자가 할당 한 사용자 버퍼로 복사됩니다. 이벤트 데이터의 내용은 `EventGetDataInfo()` 함수를 사용하여 쿼리할 수 있습니다. 채울 버퍼의 최대 크기는 이벤트 유형에 의해 정의되며 버퍼가 전달된 후 `EVENT_INFO_DATA_SIZE_MAX`를 사용하여 쿼리할 수 있습니다. 이 정보는 `EventGetInfo()` 함수를 사용하여 쿼리할 수 있습니다.

11. 고급 기능

11.1. 하위 링크 수집	166
11.2. 멀티 스트림 수집	168
11.3. CoaXPress 데이터 전달	169
데이터 전달 원칙	170
데이터 전달 연결 체계	172
라인 스캔 트리거 동기화	173
설정 스크립트 예제	174
11.4. 플랫폼 보정	175
플랫폼 보정란 무엇입니까?	176
Coaxlink FFC	179
FFC 마법사 샘플 프로그램	182
11.5. 조회 테이블 처리	189
LUT 처리 소개	190
단색 조회 테이블 처리	191
LUT 콘텐츠 정의	192
LUT 설정 절차	199
11.6. 바이엘 CFA 디코딩	202
바이엘 CFA에서 RGB 전환	203
Bayer CFA 디코더 사용	207
11.7. JPEG 인코딩	210
11.8. 레이저 선 추출	212
소개	213
코어 구현을 처리하는 LLE	215
사용 사례의 예	224
부록	231
11.9. 라인 스캔 수집	232
라인 스캔 수집 원칙	233
라인 스캔 수집 사례	237
11.10. C2C- 링크	240
C2C 링크 상호 연결	241
C2C-링크 전기 사양	243
트리거 전파 지연	245
주기 트리거 동기화	247
C2C-링크 설정 순서	250
11.11. OEM 안전 키	252
OEM 안전 키 소개	252
OEMSafetyKey 사용	253

11.1. 하위 링크 수집

Applies to:

QuadG3

QuadG3LH



경고

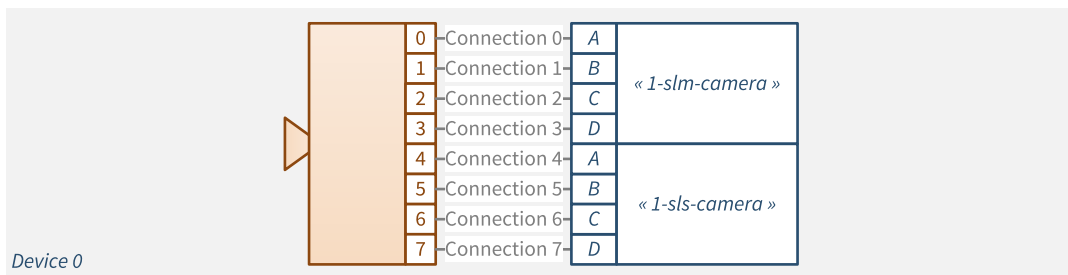
이 기능은 제한된 제품 및 펌웨어 변형 조합에만 사용할 수 있습니다!

참조: "Firmware Variants per Product" 페이지 20를 클릭하고 펌웨어 변형 열에서 "*sls-camera*" 및 "*slm-camera*"를 검색하십시오.

원칙

하위 링크 수집 기능은 특정 8 연결 CoaXPress 카메라에서 이미지를 수집할 수 있게 합니다.

8-연결 CoaXPress 링크는 2개의 하위 링크로 구분됩니다. 각 하위 링크는 1D8SL4 연결 구성표를 사용하여 Coaxlink 카드에 연결합니다.



2개의 서브 링크와 2개의 Coaxlink 카드를 사용하는 8 연결 카메라

첫 번째 4연결 서브 링크는 "하위 링크 마스터 그래버"에 연결됩니다: *1-slm-camera* 펌웨어 변형이 장착된 Coaxlink 카드.

다음 4 연결 서브 링크는 "서브 링크 슬레이브 그래버"에 연결됩니다: *1-sls-camera* 펌웨어 변형이 장착된 Coaxlink 카드.

각 그래버는 이미지 프레임의 절반을 GenTL 버퍼에 전달합니다. 애플리케이션은 두 개의 해당 버퍼의 내용을 병합하여 전체 이미지 프레임을 재구성해야 합니다. 이것은 320-sublinkEGrabber 샘플 프로그램에 나와 있습니다.

마스터 그래버는 카메라를 제어하고 시스템 트리거를 관리합니다.

두 그래버는 각각의 하위 링크의 모든 이미지 데이터를 캡처하도록 구성됩니다.

카메라 요구 사항

이 기능은 다음과 같은 특성을 갖는 8-연결 영역 스캔 카메라에만 적용됩니다.

- 첫 번째 라인의 이미지 헤더와 이미지 데이터는 단일 CoaXPress 데이터 패킷으로 함께 포장되어 CoaXPress Connection 0으로 전달됩니다.
- 프레임 (또는 ROI)의 나머지 라인의 경우 단일 이미지 라인의 이미지 데이터가 단일 패킷으로 압축되어 다음 CoaXPress Connection으로 전달됩니다. 연결은 CoaXPress 표준 패킷 배포 순서 규칙 (0에서 7까지, 그 다음에 0)을 사용하여 회전됩니다.
- 이미지 프레임 (또는 ROI) 높이는 마지막 이미지 라인이 마지막 연결(연결 7)에서 전달 되도록 8 라인의 배수여야 합니다.

11.2. 멀티 스트림 수집

Applies to:

QuadG3

QuadG3LH



경고

이 기능은 제한된 제품 및 펌웨어 변형 조합에만 사용할 수 있습니다!

참조: "Firmware Variants per Product" 페이지20 및 고급 처리 열에서 "-데이터-스트림" 접미사를 검색하십시오.

4-데이터 스트림 동시 수집

1633 Coaxlink Quad G3 및 **1633-LH Coaxlink Quad G3 LH** 의 1-카메라, 4-데이터-스트림 펌웨어 변형은 최대 4 개의 독립적인 데이터 스트림을 제공하는 영역 스캔 CoaXPress 카메라를 연결할 수 있습니다.

프레임 그래버는 들어오는 CoaXPress 데이터 블록을 CoaXPress StreamID의 2 개의 최하위 비트 값에 따라 정렬하고 네 개의 독립적인 데이터 경로를 제공합니다.

각 데이터 경로는 전체 CoaXPress 링크 대역폭, 즉 2.5 기가 바이트/초를 처리할 수 있습니다.

다음은 포함합니다:

1. 250MB FIFO 버퍼 메모리
2. 10/12/14 비트 데이터를 LSB 또는 16비트 컨테이너의 MSB에 맞출 수 있는 픽셀 프로세서.
3. 이미지 데이터를 호스트 PC의 사용자 메모리 공간으로 직접 전송하는 DMA 엔진.

`multistream` 샘플 프로그램은 4개의 EGrabber 인스턴스를 생성하고 4개의 동시 데이터 스트림에서 수집을 시작하는 방법을 보여줍니다.

11.3. CoaXPress 데이터 전달

Applies to:

QuadG3DF



경고

이 기능은 제한된 제품 및 펌웨어 변형 조합에만 사용할 수 있습니다!

참조: "Firmware Variants per Product" 페이지20 및 고급 처리 열에서 "-df."을 검색하십시오.

데이터 전달 원칙	170
데이터 전달 연결 체계	172
라인 스캔 트리거 동기화	173
설정 스크립트 예제	174

데이터 전달 원칙

1635 Coaxlink Quad G3 DF의 데이터 전달 기능을 사용하여 카메라의 이미지 데이터를 다른 호스트 PC의 여러 프레임 그래버로 전달할 수 있습니다.

데이터 전달

1635 Coaxlink Quad G3 DF와 같은 *DF-가능*는 CoAXPress 호스트 커넥터에서 수신된 데이터를 CoAXPress 데이터 전달 커넥터로 전달합니다.

CoAXPress 호스트 커넥터의 연결 A, B, C, D의 직렬 비트 스트림에 포함된 이미지 데이터 패킷은 CoAXPress 데이터 전달 커넥터의 FA, FB, FC, FD 연결로 전달됩니다.

직렬 비트 스트림은 카메라의 링크 속도에 관계없이 항상 CXP-6 속도로 작동하도록 다시 타이밍됩니다. 따라서 유틸 문자는 필요한 경우 제거되거나 비트 스트림에 추가됩니다. 유틸 문자의 추가 또는 제거는 페이로드에 영향을 주지 않습니다. 이미지 데이터는 CRC를 포함하여 보존됩니다. 데이터 포워딩을 올바르게 수행하려면 Coaxlink 1.1 표준 §8.2.5.1에서 요구하는대로 카메라가 100단어 당 적어도 한 번 IDLE 단어 하나를 삽입해야 합니다.

이미지 데이터는 무시할 수 있는 대기 시간 (일반적으로 32비트 문자 전송 시간의 몇 주기)으로 재전송됩니다.



참고 데이터 전달 출력 포트가 CoAXPress 장치 사양을 준수하지 않습니다! 다른 **1635 Coaxlink Quad G3 DF**만 피드할 수 있습니다.

데이터 전달 체인

*DF 체인*은 두 개 이상의 데이터 전달 가능 카드로 구성됩니다. 한 카드의 CoAXPress 데이터 전달 커넥터는 *DF 브리지*라고 불리는 1, 2 또는 4개의 동축 케이블 세트를 사용하여 다음 카드의 CoAXPress 호스트 커넥터에 연결됩니다.



참고 DF 체인의 카드 수에 대한 특정 상한은 없습니다.

카메라는 DF 체인의 첫 번째 카드의 CoAXPress 호스트 커넥터에 연결되며 이 카드의 이름은 **DF-마스터**입니다. DF 체인의 다른 카드는 **DF 슬레이브**라고 합니다. 마지막 DF 슬레이브 카드의 CoAXPress 데이터 전달 커넥터는 연결되지 않은 채로 있습니다.

CoAXPress 링크 검색 및 구성

DF-마스터 카드는 카메라의 CoAXPress 링크 검색 및 구성을 담당합니다.

DF-슬레이브의 CoAXPress 호스트 인터페이스는 DF-마스터가 검색한 것과 같은 수의 연결로 자동 구성됩니다.

예를 들어 카메라가 두 개의 연결을 사용하는 경우 모든 DF-브리지에 두 개의 연결만 필요합니다.

펌웨어 변형

DF-마스터에 설치할 펌웨어 변형은 카메라 유형에 따라 선택해야 합니다.

- 영역 스캔 카메라의 경우 1-camera 펌웨어 변형을 설치하십시오.
- 라인 스캔 카메라의 경우 1-camera, line-scan 펌웨어 변형을 설치하십시오.

DF-슬레이브에 설치할 펌웨어 변형은 DF-마스터에 설치된 펌웨어 변형과 일치해야 합니다:

- 1-camera firmware variant is installed on the DF-master, install the 1-df-camera 펌웨어 변형을 설치하십시오.
- 1-camera, line-scan firmware variant is installed on the DF-master, install the 1-df-camera, line-scan 펌웨어 변형을 설치하십시오.

데이터 수집 조절

데이터 전달의 올바른 작동을 위해 애플리케이션은 다음 규칙을 준수해야 합니다.

- 데이터 수집은 DF-마스터에서 활성화되기 전에 모든 DF-슬레이브에서 활성화되어야 합니다.
- 데이터 수집은 DF-슬레이브에서 비활성화되기 전에 DF-마스터에서 비활성화되어야 합니다.

라인 스캔 애플리케이션의 경우, 스캔 시작 및 스캔 종료 동기화 메커니즘이 구현되어 DF 체인의 모든 카드가 동일한 라인의 이미지 데이터를 캡처할 수 있습니다.

참조: "라인 스캔 트리거 동기화" 페이지173

카메라 주기 조절

애플리케이션에서 필요한 경우 DF-마스터 카드는 CoaXPress 호스트-장치 트리거를 구성합니다. 이는 데이터 전송이 아닌 Coaxlink 카드와 동일한 방식으로 수행됩니다.

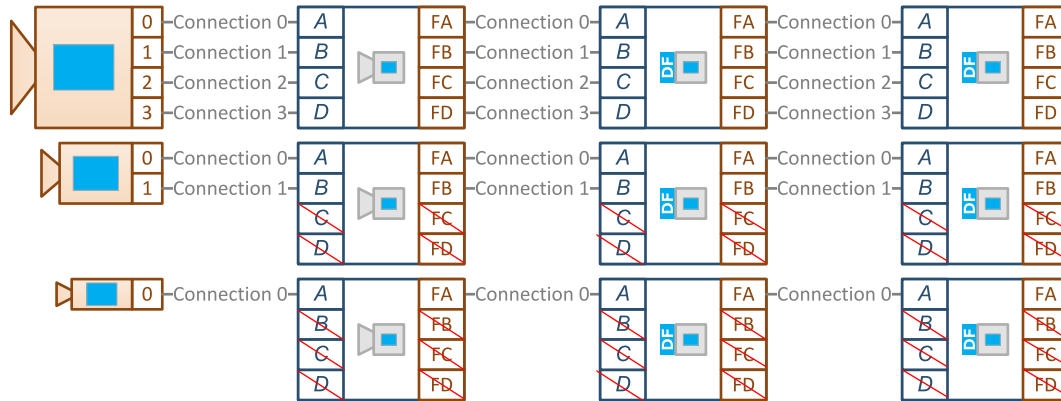
참조: "CoaXPress 호스트-장치 트리거" 페이지39

데이터 전달 연결 체계

영역 스캔 카메라 데이터 포워딩

다음 그림은 영역 스캔 카메라의 이미지 데이터가 3개의 호스트 PC (4-연결 카메라용, 2-연결 카메라용, 단일 연결 카메라용)에 전달되는 3가지 연결 체계를 보여줍니다.

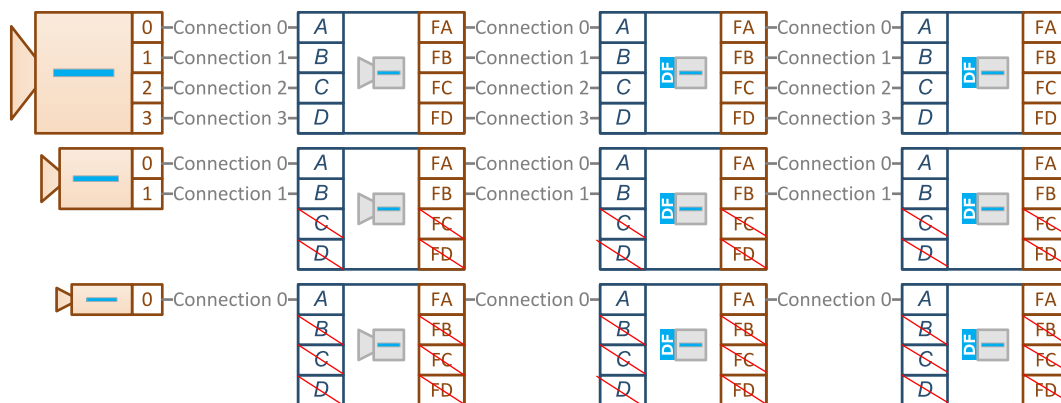
첫 번째 **1635 Coaxlink Quad G3 DF** 카드는 1-camera firmware variant; the other must be configured with the 1-df-camera 펌웨어 변형으로 구성되어야 합니다.



라인 스캔 카메라 데이터 포워딩

다음 그림은 라인 스캔의 이미지 데이터가 3개의 호스트 PC (4-연결 카메라용, 2-연결 카메라용, 단일 연결 카메라용)에 전달되는 3가지 연결 구성을 보여줍니다.

첫 번째 **1635 Coaxlink Quad G3 DF** 카드는 1-camera, line-scan firmware variant; the other must be configured with the 1-df-camera, line-scan 펌웨어 변형으로 구성되어야 합니다.



라인 스캔 트리거 동기화

스캔 시작 및 스캔 종료 트리거 동기화 메커니즘을 사용하면 DF 체인의 모든 카드가 동일한 이미지 데이터 라인을 캡처할 수 있습니다.

데이터 전달 - 마스터 카드

DF-체인에서 DF-마스터 카드는 스캔 시작 및 스캔 종료 이벤트를 DF-슬레이브 카드로 전달합니다.

DF-마스터에서 스캔 시작 및 스캔 종료 이벤트 생성은 비데이터 전달 Coaxlink 카드와 동일한 방식으로 수행됩니다.

참조: "라인 스캔 수집" 페이지232

DF-마스터 카드:

- 먼저 비동기식 스캔 트리거를 다음 라인 시작 이미지 데이터에 동기화합니다.
- 그런 다음 동기화된 스캔 트리거를 모든 DF 슬레이브와 공유합니다.

스캔 트리거의 공유는 우선 순위가 높은 "사용자 정의 GPIO 메시지"를 비트 스트림에 삽입하여 이루어집니다. 이 메시지는 모든 DF-슬레이브가 이미지 데이터와 함께 전달합니다.

데이터 전달 - 슬레이브 카드

이러한 메시지를 수신하면 DF-슬레이브는 하드웨어 이벤트를 생성합니다. 두 종류의 이벤트가 가능합니다:

- 스캔 시작 이벤트.
- 스캔 종료 이벤트

이러한 이벤트는 I/O 도구 상자의 "[이벤트 입력 도구](#)" 페이지150를 통해 사용할 수 있습니다.

동기화된 라인-스캔 수집이 필요한 어플리케이션의 경우 DF- 슬레이브의 I/O 도구 상자 EIN 도구를 로컬 시작 스캔 및 스캔 종료 트리거 소스로 사용해야 합니다.

설정 스크립트 예제

다음 스크립트는 동기화된 회선 검색 수집을 위해 데이터 전달 프레임 그래버를 구성합니다.

```
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);
  }
}
```



참고 이 예에서 스캔 시작 트리거는 마스터에서 스캔 시작 이벤트 수신이지만 스캔 종료 트리거는 로컬에서 생성됩니다. 한 가지 대안은 EIN2 as EndOfScanTriggerSource로 사용하는 것입니다.

11.4. 플랫폼 필드 보정

Applies to:

QuadG3

QuadG3LH

QuadG3DF

Octo



경고

이 기능은 제한된 제품 및 펌웨어 변형 조합에만 사용할 수 있습니다!

참조: 고급 처리 열에서 "Firmware Variants per Product" 페이지20 및 FFC
를 검색합니다.

플랫폼 필드 보정란 무엇입니까?	176
Coaxlink FFC	179
FFC 마법사 샘플 프로그램	182

플랫 필드 보정란 무엇입니까?

플랫 필드 보정 (Wikipedia: FFC)은 다음을 수정하는 데 사용되는 방법입니다:

- 카메라의 픽셀 센서 사이의 빛의 감도의 차이
- 광학 시스템과 관련된 일부 아티팩트(예: 비 균일 조명 및 비네팅)

목표는 균일한 배경이 시스템 (카메라 및 렌즈)에 의해 캡처될 때 결과 출력 이미지가 균일하도록 캡처된 (원시) 이미지의 픽셀을 수정하는 것입니다.

공식

이 보정은 다음 작업을 원본 이미지의 각 픽셀에 적용하여 수행됩니다:

$$\text{보정된 픽셀} = (\text{원시 픽셀} - \text{오프셋}) * \text{게인}$$

여기서 오프셋 및 게인 계수는 각 픽셀에 대한 특정 값입니다.

계수 오프셋 및 게인의 평가에는 다음 단락에서 설명하는 교정 절차가 필요합니다.

교정

계수 계산을 위한 교정 절차는 두 단계로 수행됩니다.

어두운 이미지 수집

시스템에서 어두운 이미지를 얻습니다. 이것은 일반적으로 캡으로 렌즈를 덮어서 달성합니다. 캡처된 이미지는 센서의 다크 전류를 나타내며 정상 상태에서 이미지를 수집할 때 제거하려는 고정 바이어스로 간주됩니다. 이 보정은 다크 프레임 공제라고 합니다.

$$\text{보정된 픽셀} = \text{원시 픽셀} - \text{어두운 픽셀}$$

각 픽셀에 대해 어두운 픽셀 값은 위 FFC의 오프셋에 해당합니다.

플랫 이미지 수집

시스템에서 플랫 이미지를 얻습니다. 예를 들어, 플랫(균일한) 배경을 캡처하여 하얗게 되지 않도록 너무 밝지 않게 하고 너무 어둡게 하지 않습니다.

어둡고 플랫 수집에서 FFC의 *게인* 값을 계산하기에 충분한 데이터가 있습니다.

이를 위해 플랫 이미지에 대해 맞다고 생각하는 픽셀 값으로 *보정된 픽셀*을 정의합니다. 이 값을 평면 이미지(*평균(플랫)*)의 평균 픽셀 값으로 설정하고 어두운 이미지의 평균(*평균(플랫)*)으로 보정해 보겠습니다. FFC 조건에서 다음을 제공합니다:

$$\text{평균(플랫)} - \text{평균(어두운)} = (\text{플랫픽셀} - \text{어두운 픽셀}) * \text{게인}$$

이것은 게인 값으로 이어집니다

$$\text{게인} = \text{평균(플랫)} - \text{평균(어두운)} / (\text{플랫픽셀} - \text{어두운 픽셀}) * \text{게인}$$

동일한 계산이 플랫하고 어두운 이미지의 모든 픽셀을 포괄하기 위해 (*가로*높이*)번 반복됩니다. 그 결과 이미지의 각 픽셀에 대한 특정 보정이 이루어집니다.



참고참고: 카메라 장치, 조명 또는 광학 장비를 포함하여 시스템의 일부가 변경된 경우 이 보정 절차를 다시 수행해야 합니다.

컬러 픽셀 형식의 보정

컬러 픽셀 형식의 경우 평균(플랫) 값을 계산하는 여러 가지 방법이 있습니다. 모든 경우에 모든 픽셀 구성 요소를 포함하기 위해 *개인* 계산이 반복됩니다 (*너비 * 높이 * componentsPerPixel*). 이미지의 각 픽셀 구성 요소에 대해 특정 수정이 이루어집니다.

픽셀 구성 요소 개별 처리

즉_{RGB}에서:

- 빨강 구성요소의 *개인* 값을 계산하기 위해 *average(Flat[Red])*을 사용하는 단계;
- 녹색 구성요소의 *개인* 값을 계산하기 위해 *average(Flat[Green])*을 사용하는 단계;
- 파랑 구성요소의 *개인* 값을 계산하기 위해 *average(Flat[Blue])*을 사용하는 단계;

픽셀 구성 요소를 함께 처리

즉_{RGB}에서는 빨강, 녹색 및 파랑 구성 요소의 *개인* 값을 계산하기 위해 *average(average(Flat[Red]), average(Flat[Green]), average(Flat[Blue]))*를 사용합니다.

평균을 계산하는 이러한 방식 (즉, 픽셀 구성 요소에 대해)은 구성 요소 간의 균형을 보정하는 FFC 계수를 생성합니다. 따라서, 플랫 이미지를 얻기 위해 사용되는 균일한 배경의 품질에 따라, FFC는 효과적으로 화이트 밸런스 보정을 수행할 수 있습니다.

Coaxlink FFC

Coaxlink 제품의 플랫 필드 보정 구현

일부 카메라에는 내장 FFC 모듈이 있지만 다른 카메라는 이 기능을 구현하지 않습니다. 그러나 이 기능이 없는 장치는 Coaxlink 카드의 FFC 코어로 수정할 수 있습니다.

Coaxlink 펌웨어의 FFC 코어는 이미지의 위치에 해당하는 계수 (오프셋 and 게인)를 사용하여 FFC를 적용하여 카메라에서 직접 오는 픽셀을 보정합니다. 보정은 Coaxlink 픽셀 프로세싱 체인의 초기 단계에서 이루어지기 때문에 *RedBlueSwap*, *LUT*, 및 *Bayer Decoding*과 같은 다른 픽셀 처리 기능이 수정된 픽셀에서 수행됩니다.

게인 및 오프셋 계수 형식

보정 절차에서 계산된 계수는 다음과 같이 인코딩된 경우 Coaxlink 카드에 로드할 수 있습니다.

- 하나의 픽셀 성분에 대한 오프셋 및 게인 값들은 함께 16비트 리틀 엔디안 값으로 패키징됩니다:
 - *게인*은 비트 9..0에서 [Wikipedia:UQ2.8](#)로 인코딩됩니다.
 - *오프셋*은 비트 15..10의 6비트 부호없는 정수입니다.
- 픽셀 구성 요소 값과 관련된 계수는 이미지의 픽셀 구성 요소와 동일한 순서로 개별적으로 처리됩니다. 예를 들어, *RGB8* 형식에서 한 픽셀은 연속적인 8비트 값 (*Red*, *Green*, *Blue*)으로 인코딩되므로 하나의 *RGB8* 픽셀을 수정하기 위해 3개의 연속적인 16비트 팩형 계수가 필요합니다.

16비트 팩형 계수가 (순서대로) 바이너리 파일에 저장된다면 (예를 들어 *'path/to/coefficients.ffc'*) Euresys 스크립트에서 쉽게 로드할 수 있습니다.

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

그래버가 설정할 그래버를 참조하는 스크립트 변수를 호출하여 쉽게 로드할 수 있습니다.



참고 그러한 바이너리 파일은 Euresys [ffc-wizard](#) 샘플 응용 프로그램에 의해 생성될 수 있습니다

명세서

카메라 유형

FFC 기능은 픽셀 구성 요소 당 8, 10, 12, 14 또는 16비트 데이터를 제공하는 단색, Bayer CFA 및 RGB 컬러 영역 스캔 카메라에 적용할 수 있습니다.

최대 이미지 크기

다음 표는 FFC가 활성화된 경우 지원되는 모든 픽셀 형식의 최대 이미지 크기를 보여줍니다:

비트 심도	최대 이미지 크기[픽셀]		
	흑백 및 바이엘 CFA 형식	RGB 형식	RGBa 형식
8-비트	134,217,728 픽셀	44,739,242 픽셀	33,554,432 픽셀
10-비트	107,374,182 픽셀	35,791,394 픽셀	26,843,545 픽셀
12-비트	89,478,485 픽셀	29,826,161 픽셀	22,369,621 픽셀
14-비트	76,695,844 픽셀	25,565,281 픽셀	19,173,961 픽셀
16-비트	67,108,864 픽셀	22,369,621 픽셀	16,777,216 픽셀

성능

활성화된 경우 FFC 기능은 각 처리된 픽셀에 대해 추가 16비트 계수 데이터를 가져오기 때문에 DRAM 메모리에 상당한 로드를 추가합니다. FFC가 활성화되면 Coaxlink 카드는 CoaXPress Link가 달성할 수 있는 최대 데이터 속도의 일부만 유지할 수 있습니다. 이 치수가 없는 값의 이름은 "지속 가능한 상대적 데이터 전송률"입니다

다음 표는 FFC를 지원하는 모든 비트 심도 및 제품/펌웨어 변형 조합에 대한 지속 가능한 상대적 데이터 전송률을 보여줍니다.

지속 가능한 상대적 데이터 전송률

비트 심도	지속 가능한 상대적 데이터 전송률 [4차선 CXP-6 최대 데이터 전송률]	
	1633 Coaxlink Quad G3 - 1-카메라 1633-LH Coaxlink Quad G3 LH - 1-카메라	3602 Coaxlink Octo - 2-카메라
8-비트	70.0 %	123.2 %
10-비트	77.8 %	136.9 %
12-비트	84.0 %	147.8 %
14-비트	89.1 %	156.8 %
16-비트	93.3 %	164.3 %



참고"지속 가능한 상대적 데이터 전송률"은 보드에 연결된 모든 카메라에 대해 글로벌입니다. 예를 들어, **3602 Coaxlink Octo** 에서 2-카메라 8비트 유스 케이스의 지속 가능한 데이터 속도는 123.2%로 카메라의 경우 100%로, 다른 카메라의 경우 23.2%로 균등하게 나눌 수 있습니다.



참고Coaxlink 카드는 블랭킹 간격 동안 데이터를 수집하지 않습니다. 라인 및 프레임 블랭킹 간격은 메모리 대역폭을 소비하지 않으므로 카메라 데이터 속도 계산시 제외해야 합니다.

대기 시간, FIFO 버퍼 오버 플로우 및 프레임 손실을 피하기 위해 Euresys는 (글로벌) 카메라 데이터 전송률을 적절하게 제한할 것을 권장합니다.

FFC 사용

데이터 스트림 모듈에서 **FfcControl** 기능 값을 **Enable**로 설정하십시오.

FFC 비활성화

데이터 스트림 모듈에서 **FfcControl** 기능 값을 **Disable**로 설정하십시오.

FFC 마법사 샘플 프로그램

Eureys는 계수를 계산하고 Coaxlink 카드를 대상으로하는 이진 파일로 압축하는 `ffc-wizard` 라는 샘플 응용 프로그램의 소스 코드를 제공합니다.

이 샘플 코드의 목적은 세 가지입니다:

- 보정 절차를 통해 사용자를 안내합니다;
- 이 문서에서 설명하는 것을 기술적으로 번역합니다;
- 커스텀 애플리케이션 개발을 위한 블록 빌딩을 제공합니다.

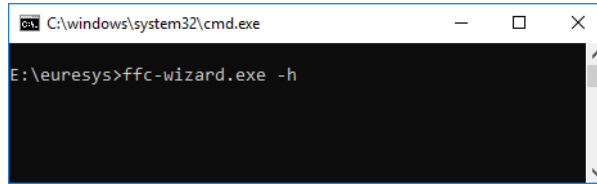
빌딩

소스 코드는 단일 소스 파일에 있습니다: `src/ffc-wizard.cpp`. 응용 프로그램을 작성하는 것은 간단해야 합니다.

- Windows의 경우 Microsoft Visual Studio 프로젝트 파일 `ffc-wizard.vcproj`가 있습니다.
- Linux 및 macOS의 경우 `Makefile`가 제공됩니다.

사용법

마법사는 콘솔 응용 프로그램입니다. 플래그 `-h`(또는 `--help`)가 제공되면 도움말 메시지가 표시됩니다:



도움말 메시지 받기

```
> ffc-wizard.exe --help
플랫 필드 보정 마법사
Fcc-마법사 [OPTIONS]

선택:
--if=INT          사용할 GenTL 인터페이스의 색인
--dev=INT         사용할 GenTL 장치의 색인
--ds=INT         사용할 GenTL 데이터 스트림의 색인
--average=INT     평균 이미지 수 (기본값: 10)
--roi_x=INT       ROI 왼쪽 상단 모서리의 픽셀 단위 가로 오프셋 (기본값: 0)
--roi_y=INT       ROI 왼쪽 상단 모서리의 픽셀 단위 세로 오프셋 (기본값: 0, 라인 스캔
에는 무시됨)
--roi_width=INT   ROI 너비 (기본값: 전체 이미지)
--roi_height=INT  ROI 높이 (기본값: 전체 이미지, 라인 스캔에서는 무시)
--balance        각 구성 요소가 아닌 모든 구성 요소에서 플랫 이미지 평균 계산
--Linescan       즉, 강제 라인 스캔 모드. 평균 이미지 라인 (라인 스캔 카드에 대해 자
동으로 활성화됨)
--dark-setup=SCRIPT 어두운 수집을 위한 설정 스크립트 경로
--flat-setup=SCRIPT 플랫 수집을 위한 설정 스크립트 경로
--timeout=MS      이미지를 기다리는 최대 시간 (밀리초) (기본값: 1000)
--dark-histogram=FILE 출력하고 열 평균 어두운 이미지의 히스토그램 html 페이지 경로
--flat-histogram=FILE 출력하고 열 평균 평균 이미지의 히스토그램 html 페이지 경로
--output-ffc=FILE 계수 출력 파일 경로 (Coaxlink ffc 이진 형식)
--load-ffc=FILE   Coaxlink 계수 분할로 계수로 로드 (기본값: 계산된 계수)
--no-interact     사용자 상호 작용 건너뛰기
-h --help        도움말 메시지 표시
```

참고: ROI 옵션을 사용하면 평균을 계산하는 동안 직사각형 영역을 정의할 수 있으며, 비네팅이 적용되는 이미지에서 테두리 근처의 픽셀을 제거하는 데 유용합니다.

옵션 세부 정보

플래그	세부 정보
<code>--if, --def, --ds</code>	사용(및/또는 구성)할 데이터 스트림을 식별하는 GenTL 모듈의 인덱스
<code>--average</code>	다크 및 플랫 수집을 위해 수집할 이미지의 수; 이러한 수집의 평균만 보정 절차에 사용됩니다
<code>--roi_x, --roi_y, --roi_width, --roi_height</code>	평면 이미지(<i>average(Flat)</i>)의 평균 픽셀 값을 계산할 때 고려할 선택적 직사각형 영역. 이것은 각 픽셀에 대한 이득 값의 평가에 영향을 미칩니다

플래그	세부 정보
<code>--balance</code>	화이트 밸런스를 활성화합니다. 즉, 컬러 픽셀 성분의 계수가 성분 값을 밸런싱하도록 계산되는지의 여부; 분명히 이것은 평평한 이미지를 수집하기 위해 사용하는 플랫한 배경이 가능한 한 실제 회색에 가까울 것을 요구합니다 (모든 RGB 구성 요소는 동일한 값을 가짐)
<code>--dark-setup, --flat-setup</code>	다크 및 플랫 수집 전에 실행하기 위한 선택적 구성 스크립트
<code>--dark-histogram, --flat-histogram</code>	선택. 다크 및 플랫 이미지 픽셀 구성 요소의 히스토그램을 보여주는 출력 파일의 경로. 이것은 플랫 이미지의 변화뿐만 아니라 암 전류 변화의 시각적 개요를 제공합니다
<code>--no-interact</code>	일반적으로 마법사는 다크 및 평면 수집을 시작하기 전에 사용자가 시스템을 설정할 때까지 기다립니다. 이 플래그가 사용되면 마법사는 사용자를 기다리지 않고 (생성된 히스토그램 html 페이지도 열지 않습니다)

예제

다음은 화이트 밸런스 기능을 사용하여 FFC 계수 (`coefficients.ffc` 파일에 기록)를 생성하는 예제입니다. 콘솔 창에서 실행하는 명령은 다음과 같습니다:

```
C:\windows\system32\cmd.exe
E:\euresys>ffc-wizard.exe --balance --output-ffc=coefficients.ffc
```

프로그램이 시작되고 수행할 작업을 알려줍니다:

```
C:\windows\system32\cmd.exe - ffc-wizard.exe --balance --output-ffc=coefficients.ffc
E:\euresys>ffc-wizard.exe --balance --output-ffc=coefficients.ffc
Flat Field Correction Wizard
Please setup the camera for dark acquisitions (e.g. using the lens cap) and press
Enter to continue.
Note: the dark component values will be considered as dark current and will be use
d for dark frame subtraction
```


어두운 수집을 위한 설정을 준비하고 준비가 되면 Enter를 누를 수 있습니다. 그런 다음 일련의 어두운 이미지를 수집하고 다음 단계에 대한 지침을 표시합니다

```

C:\windows\system32\cmd.exe - ffc-wizard.exe --balance --output-ffc=coefficients.ffc
E:\euresys>ffc-wizard.exe --balance --output-ffc=coefficients.ffc
Flat Field Correction Wizard
Please setup the camera for dark acquisitions (e.g. using the lens cap) and press
Enter to continue.
Note: the dark component values will be considered as dark current and will be use
d for dark frame subtraction

Mono8 1280x1024, 1 component per pixel, 1 byte per component
Components processed individually: Luminance
Acquiring images ...
Please setup the camera for flat acquisitions and press Enter to continue.

```

평면 수집을 위한 설정을 준비하고 준비가 되면 Enter를 누를 수 있습니다. 그런 다음 일련의 평면 이미지를 획득하고 해당 계수를 계산하여 coefficients.ffc 파일에 씁니다.

```

C:\windows\system32\cmd.exe
E:\euresys>ffc-wizard.exe --balance --output-ffc=coefficients.ffc
Flat Field Correction Wizard
Please setup the camera for dark acquisitions (e.g. using the lens cap) and press
Enter to continue.
Note: the dark component values will be considered as dark current and will be use
d for dark frame subtraction

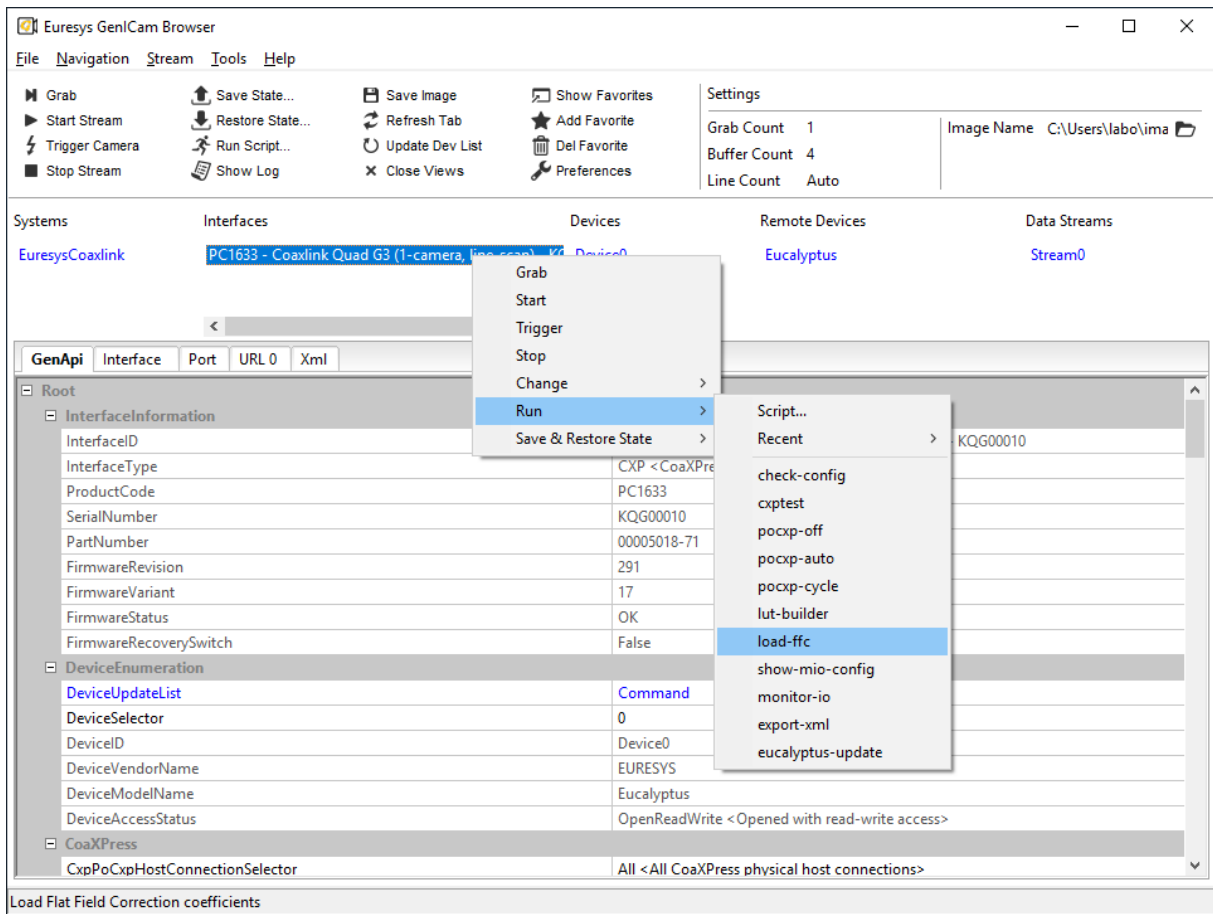
Mono8 1280x1024, 1 component per pixel, 1 byte per component
Components processed individually: Luminance
Acquiring images ...
Please setup the camera for flat acquisitions and press Enter to continue.

Mono8 1280x1024, 1 component per pixel, 1 byte per component
Components processed individually: Luminance
Acquiring images ...
Computing coefficients ...
Packing coefficients ...
Writing coefficients.ffc ...
Done.

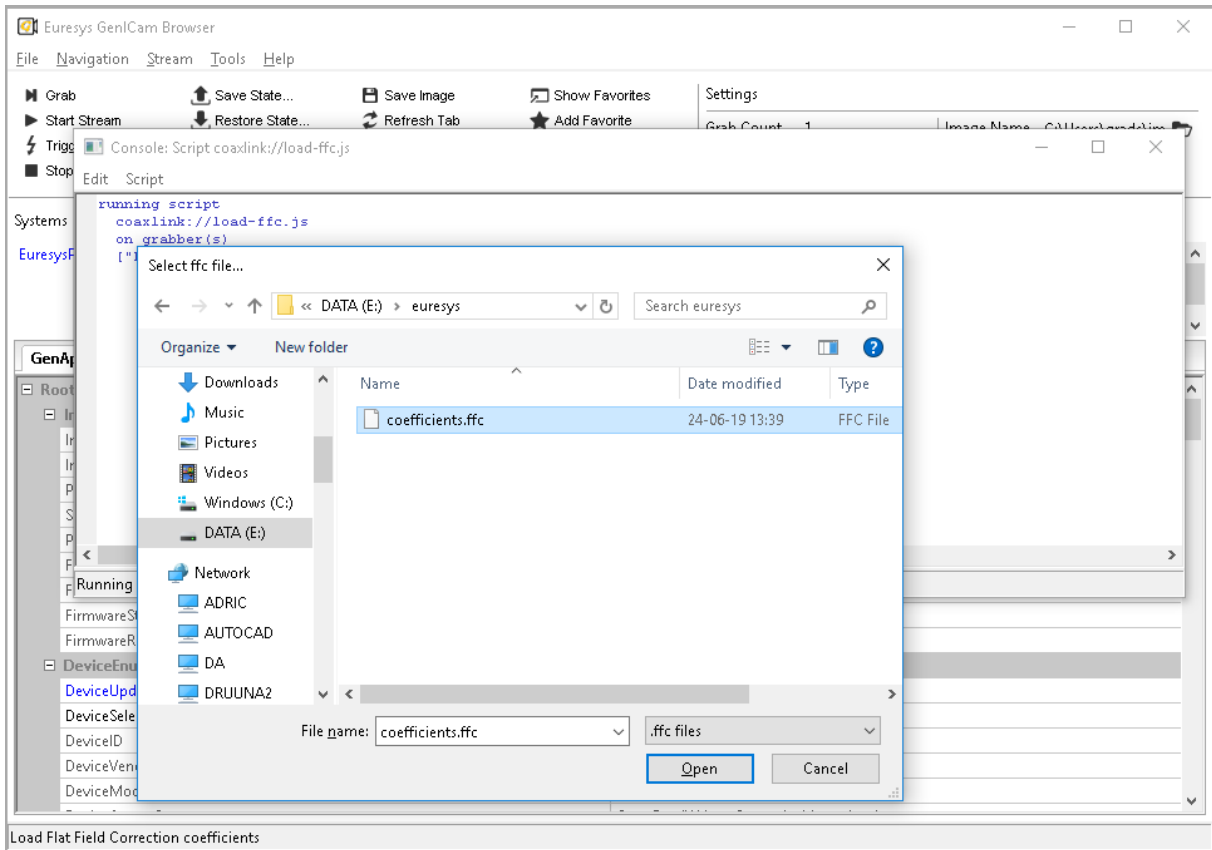
E:\euresys>

```

나중에 다음과 같이 load-ffc 스크립트를 실행하여 genicam-browser에서 계수를 로드할 수 있습니다:



그런 다음 이전에 만든 파일 계수를 선택할 수 있습니다.



그 순간부터 계수가 Coaxlink 메모리에 로드되고 FFC 처리가 활성화됩니다.

디자인

계수의 패킹뿐만 아니라 캘리브레이션 절차는 주 기능 `ffcWizard`에 의해 제어됩니다.

`ffcWizard` 태스크

태스크	기능별 완료
일련의 다크 이미지를 획득하여 평균 다크 이미지를 생성합니다	<code>acquireImages</code>
일련의 평면 이미지를 수집하여 평평한 이미지를 만듭니다	<code>acquireImages</code>
각 픽셀 구성 요소에 대한 <code>gain</code> 값 계산	<code>computeGain</code>
다크 픽셀 구성 요소 값을 <code>offset</code> 값으로 사용	<code>computeOffset</code>
<code>offset</code> and <code>gain</code> 값을 16비트 리틀 엔디안 값으로 패킹	<code>packCoefficients</code>
팩된 계수를 2진 파일에 기록	<code>savePackedCoefficients</code>

맞춤 설정

샘플 애플리케이션은 이미 다음과 같은 몇 가지 일반적인 픽셀 형식을 지원합니다: Mono, RGB, RGBa and Bayer.

제한: 샘플 애플리케이션의 복잡성을 제한하기 위해 모든 구성 요소의 크기가 동일해야 합니다 (픽셀 당 여러 구성 요소가 있는 픽셀 형식의 경우). `addImage` and `addComponents`를 업데이트하여 다양한 구성 요소 크기를 지원하는 픽셀 형식을 지원할 수 있습니다.

이전의 제한 조건 하에서 새로운 픽셀 형식을 지원하려면 두 가지 기능을 수정해야 합니다:

1. `Image::getComponentsPerPixel` - PFNC 이름으로 식별되는 새로운 형식의 픽셀 당 구성 요소 수를 반환합니다.
2. 새 형식 (PFNC 이름으로 식별됨)의 픽셀 구성 요소가 이미지에 어떻게 배치되는지 설명하는 `Image::getComponentFilters`, to return a `std::vector of ComponentFilter` 객체

다크하고 플랫한 이미지의 `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 구성 요소입니다.

픽셀 구성 요소 레이아웃 구성에 대한 자세한 내용은 `Image::getComponentFilters` 소스 코드를 참조하십시오.

11.5. 조회 테이블 처리



경고

이 기능은 제한된 제품 및 펌웨어 변형 조합에만 사용할 수 있습니다!

참조: "Firmware Variants per Product" 페이지20 및 고급 처리 열에서 LUT
을 검색하십시오.



경고

Coaxlink 드라이버의 이 배포는 독점적으로 흑백 픽셀 형식에 대한 조회
테이블 처리를 제공합니다.

LUT 처리 소개	190
단색 조회 테이블 처리	191
LUT 콘텐츠 정의	192
LUT 설정 절차	199

LUT 처리 소개

Coaxlink 드라이버의 이 배포는 독점적으로 흑백 픽셀 형식에 대한 조회 테이블 처리를 제공합니다.

참조: 자세한 설명은 "단색 조회 테이블 처리" 다음 페이지를 참조하십시오.

Coaxlink 드라이버는 조회 테이블의 내용을 정의하는 네 가지 방법을 제공합니다.

참조: "LUT 콘텐츠 정의" 페이지192

참조: "LUT 설정 절차" 페이지199는 조회 테이블을 설정합니다.

단색 조회 테이블 처리

구성

다음 표에는 단색 픽셀에 사용할 수 있는 모든 조회 테이블 구성이 나와 있습니다:

구성	입력 픽셀 형식[PFNC]	입력 비트	출력 비트
M_8x8	Mono8, 원시	8	8
M_10x8	Mono10	10	8
M_10x10			10
M_10x16			16
M_12x8	Mono12	12	8
M_12x12			12
M_12x16			16



참고 단색 8비트 픽셀은 단색 8비트 픽셀로 변환할 수 있으며 단색 10비트 픽셀은 단색 8비트, 10비트 또는 16비트 픽셀로 변환할 수 있으며 단색 12비트 픽셀은 단색 8비트, 12비트 또는 16비트 픽셀로 변환할 수 있습니다.

조회 테이블 데이터 세트

조회 테이블 데이터 세트는 픽셀의 각 구성 요소에 대해 하나의 검색 테이블을 구성하는 데 필요한 데이터 집합으로 정의됩니다. 단색 픽셀의 경우, 조회 테이블 데이터 세트는 하나의 단일 조회 테이블 콘텐츠만을 포함합니다.

업로드할 수 있는 조회 테이블 데이터 세트의 개수는 조회 테이블 구성에 따라 다릅니다.

구성	데이터 세트
M_8x8	16
M_10x8, M_10x10, M_10x16	4
M_12x8, M_12x12, M_12x16	1

LUT 콘텐츠 정의

Coaxlink 드라이버는 조회 테이블의 내용을 정의하는 네 가지 방법을 제공합니다.

반응 조절 방법

이 방법은 룩업 테이블의 전달 함수를 4 개의 파라미터로 정의합니다: **Contrast**, **Brightness**, **Visibility** 및 **Negative**.

Contrast 및 **Brightness** 매개 변수는 텔레비전 모니터의 밝기 및 대비 컨트롤과 유사한 컨트롤을 제공합니다.

Visibility 매개 변수는 전체 입력 범위를 포괄하는 전달 함수를 부드럽게 바꿀 수 있는 컨트롤을 제공합니다.

Negative 매개 변수를 사용하면 이미지를 음수 이미지로 변환할 수 있습니다.

강조 방법

이 방법은 두 개의 매개 변수를 사용하여 조회 테이블의 전달 함수를 정의합니다: **Emphasis** 및 **Negative**.

그것은 γ (감마) 함수라고 알려진 곱셈 법칙을 사용하여 이미지를 변형할 수 있습니다.

Negative 매개 변수를 사용하면 이미지를 음수 이미지로 변환할 수 있습니다.

임계값 방법

이 방법은 다섯 개의 매개 변수를 사용하여 이중 임계 값 변환 법칙을 정의합니다: **SlicingLevel**, **SlicingBand**, **LightResponse**, **BandResponse**, **DarkResponse**.

테이블 방법

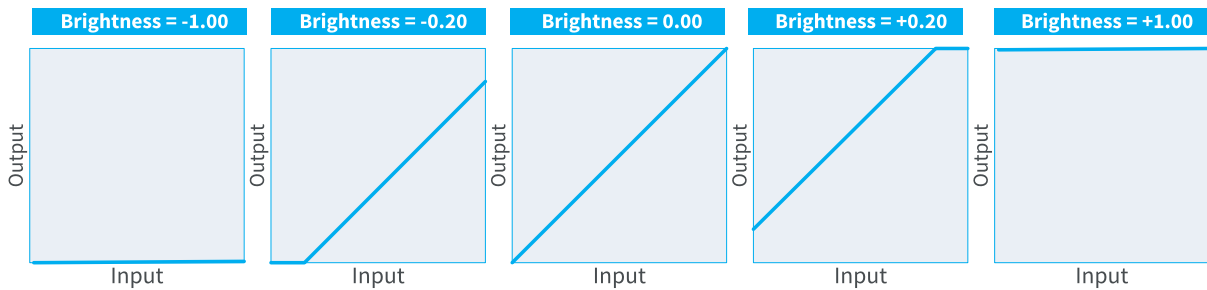
이 방법은 조회 테이블의 전달 함수를 표 형식으로 정의합니다.

밝기 매개 변수

Brightness 매개 변수는 응답 제어 조회 테이블 정의 메소드에만 적용됩니다.

그것은 텔레비전 모니터의 밝기 제어와 유사한 제어를 구현합니다.

밝기	참고
-1.00	최소값. 가장 어두운 출력. 전체 입력 범위 데이터가 완전 검정색으로 변환됩니다. 이 규칙은 선택한 Contrast 값에 적용됩니다.
0.00	기본값 중간 레벨 입력 레벨 0.5는 동일한 출력 레벨 0.5로 변환됩니다. 이것은 다른 매개 변수의 모든 값에 적용됩니다. 밝기가 +1.00으로 증가하면 더 밝은 출력이 됩니다. -1.00에 대한 밝기가 감소하면 어두운 출력이 됩니다.
+1.00	최대 값. 가장 밝은 출력. 전체 입력 범위 데이터가 완전 흰색으로 변환됩니다. 이 규칙은 선택한 Contrast 값에 적용됩니다.



다른 모든 조절이 기본값으로 설정된 경우 Brightness의 효과:
= ; = ; = .Contrast1.00Visibility0.00NegativeFALSE

명암 매개 변수

Contrast 매개 변수는 응답 제어 조희 테이블 정의 메소드에만 적용됩니다.

그것은 텔레비전 모니터의 대비 제어와 유사한 제어를 구현합니다.

변환 법칙의 기울기는 **Contrast** 매개 변수에서 비선형적으로 제어되는 게인입니다.

수학적으로 이 관계는 다음과 같습니다:

$$\text{게인} = 10^{2 \times (\text{명암} - 1)}$$

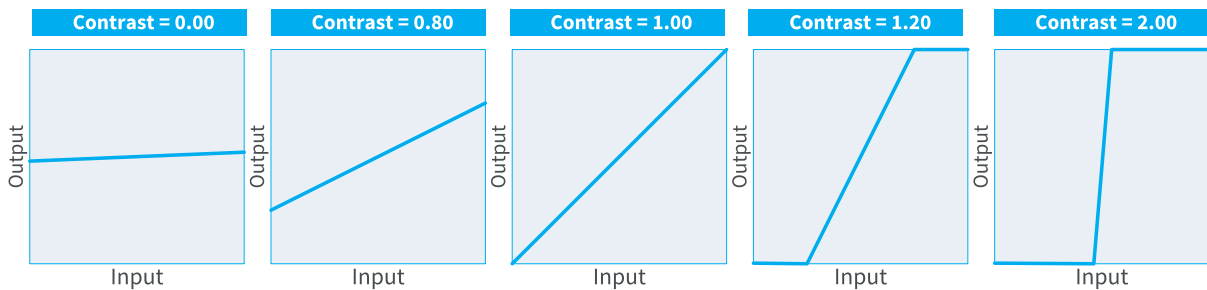
명암	게인	참고
0.00	0.01	최소 Contrast 값; 가장 작은 게인
1.00	1	기본 Contrast 값; 단위 게인
2.00	100	최대 Contrast 값; 가장 큰 게인

필요한 게인을 얻으려면 명암 조절을 다음과 같이 설정해야 합니다:

$$\text{명암} = 1 + (\log_{10} \text{게인}) / 2$$

필요한 게인이 데시벨(dB)로 표시되는 경우:

$$\text{명암} = 1 + \text{게인(dB)} / 40$$



다른 모든 컨트롤이 기본값으로 설정된 경우 **Contrast**의 효과:

`= ; = ; = Brightness0.00Visibility0.00NegativeFALSE`

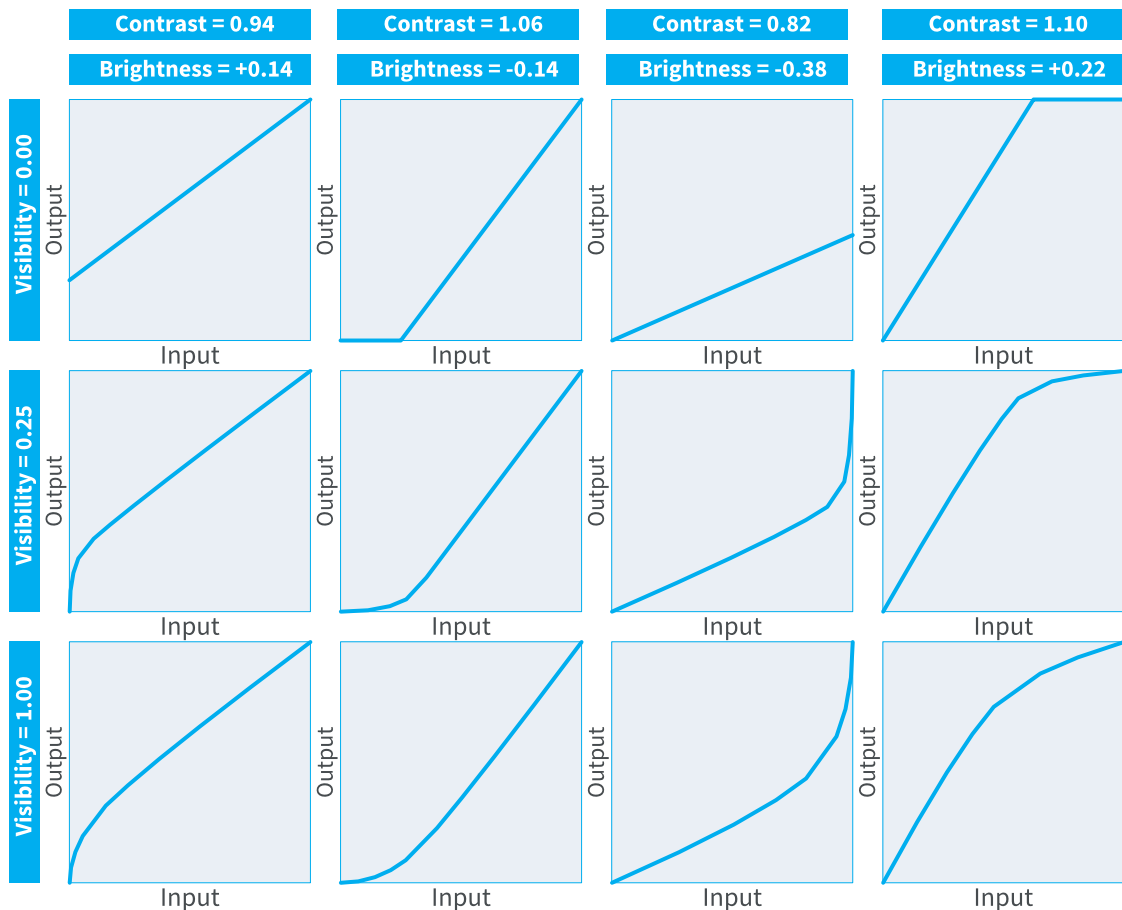
가시성 매개 변수

Visibility 매개 변수는 응답 제어 조회 테이블 정의 메소드에만 적용됩니다.

Contrast 및 매개 변수를 사용하면 입력 역학의 Brightness 일부가 제거되는 경우가 있습니다. 이미지의 매우 어두운 영역은 전체 검정색으로 변형되어 보이지 않게 됩니다. 이것은 매우 밝은 영역에서 완전 흰색으로 클리핑되는 경우에도 적용됩니다.

Visibility 매개 변수는 이미지의 숨겨진 부분을 부드럽게 나타내기 위해 만들어졌습니다.

시계 (Visibility)	계 인	참고
0.00	0.01	최소값과 기본값. 이것은 조각별 선형 변환 곡선을 생성합니다. +1에 가까운 값을 선택하면 더 부드러운 곡선이 생성됩니다.
1.00	100	최대 값.



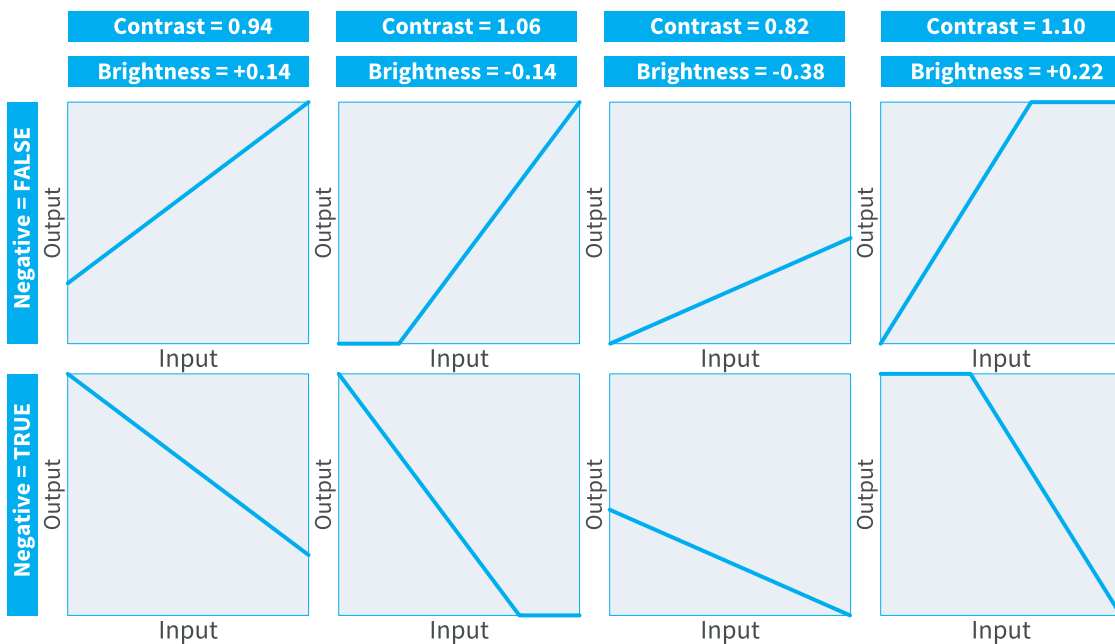
Negative = FALSE라고 가정한 Contrast 및 Brightness 매개 변수의 일반적인 값에 대한 Visibility의 영향

음수 매개 변수

이 **Negative** 매개 변수는 반응 조절 및 강조 조희 테이블 정의 방법에 모두 적용됩니다.

이 컨트롤을 사용하면 이미지를 가장 어두운 영역이 가장 어둡게 보이고 가장 어두운 영역이 가장 밝게 보이는 이미지를 네가티브 이미지로 변환할 수 있습니다.

음수	참고
거짓	기본값
참	변환 테이블은 그래프의 세로 축을 기준으로 미러링됩니다. 이렇게하면 흑백 값이 바뀌고 사진에 네가티브 효과가 발생합니다.



다른 컨트롤의 일반적인 값에 대한 **Negative** 효과

강조 매개 변수

이 **Emphasis** 매개 변수는 반응 조절 조회 테이블 정의 방법에만 적용됩니다. 그것은 출력 법칙을 사용하여 이미지를 변환할 수 있습니다:

$$\text{출력} = \text{입력}^\gamma$$

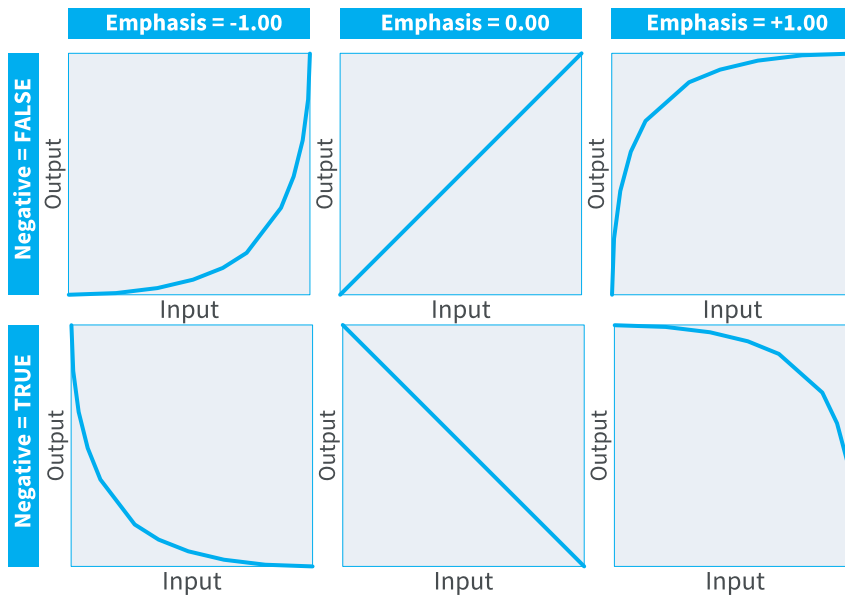
γ - 감마 - 지수는 다음과 같이 수학적으로 강조 표시됩니다:

$$\gamma = 10^{-\text{강조}}$$

강조	감마	참고
1.00	0.1	최대. 강조 값; 가장 작은 γ 값
0.00	1	기본 강조 값; 선형 법칙
-1.00	10	최소강조 값; 가장 큰 γ 값

필요한 γ 를 얻으려면 다음과 같이 **Emphasis** 설정해야 합니다:

$$\text{강조} = -\log_{10}\gamma$$



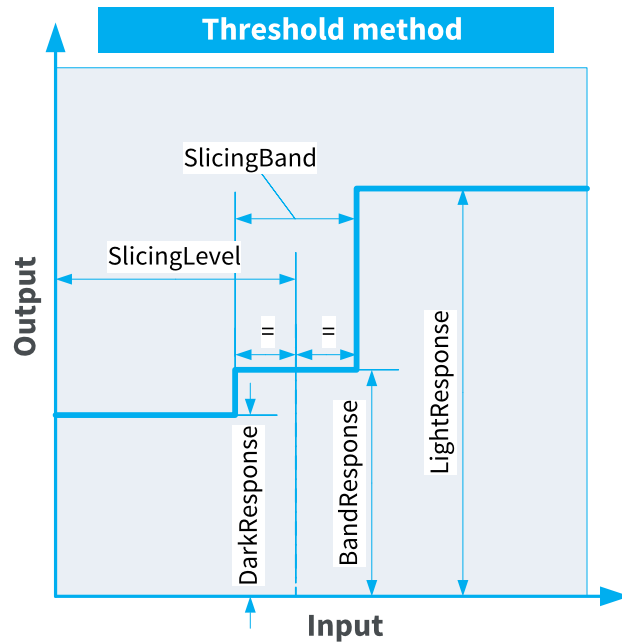
Emphasis의 일반적인 값과 **Negative**의 두 값에 대한 강조 효과

임계값 방법 매개 변수

SlicingLevel, SlicingBand, LightResponse, BandResponse 및 DarkResponse 매개 변수는 임계값 조희 테이블 정의 메소드에만 적용됩니다.

다음 그림에서와 같이 매개 변수 세트는 이중 임계 변환법을 정의합니다.

매개 변수	최소값	기본값	최대값
SlicingLevel	0.00	0.50	1.00
SlicingBand	0.00	0.50	1.00
LightResponse	0.00	0.75	1.00
BandResponse	0.00	0.50	1.00
DarkResponse	0.00	0.25	1.00



이중 임계값 전달 함수



참고 SlicingLevel는 입력 범위에서 두 임계값의 평균값을 지정합니다.

LUT 설정 절차

조회 테이블 처리를 설정하려면 다음을 수행하십시오.

1. 조회 테이블 사용 안 함
2. 조회 테이블 구성 정의
3. 조회 테이블의 내용 정의
4. 조회 테이블 내용을 지정된 조회 테이블 데이터 세트에 업로드하십시오
5. 지정된 데이터 세트로 조회 테이블 사용

조회 테이블 사용 안 함

조회 테이블 사용 안 함:

- LUTEnable 기능을 Off로 설정하십시오.

조회 테이블 구성 정의

찾아보기 테이블 구성을 정의하려면 다음에 따라 LUTConfiguration 기능을 설정하십시오:

- 카메라 픽셀 유형 및 비트 깊이
- 필요한 출력 비트 심도.

참조: 단색 픽셀에 적용 할 수 있는 구성은 "단색 조회 테이블 처리" 페이지 191입니다.



참고 조회 테이블 구성은 다른 작업 전에 설정해야 합니다.

조회 테이블 내용 정의

참조: 조회 테이블 내용을 정의하는 데 사용되는 매개 변수 및 표 메서드에 대한 설명은 "LUT 콘텐츠 정의" 페이지 192를 참조하십시오.



참고 최소한 하나의 찾아보기 테이블 세트가 정의되어야 합니다.

조회 테이블 콘텐츠 업로드

한 번의 작업으로 조회 테이블 콘텐츠를 업로드하려면 다음을 수행하십시오:

- 적절한 값을 LUTSet 기능에 할당하여 액세스할 조회 테이블 데이터 세트를 선택하십시오. 예를 들어 Set1.
- LUTIndex 기능을 0로 설정합니다.
- LUTValue 기능에 LUTLength 값 문자열을 작성하십시오.



참고또한 애플리케이션은 임의의 개별 조회 테이블 엔트리 또는 임의의 연속 조회 테이블 엔트리 블록을 선택적으로 업로드할 수 있습니다.

조회 테이블 데이터 세트 다시 읽기

한 번의 작업으로 조회 테이블 데이터 세트를 다시 읽으려면 다음과 같이 하십시오:

- `LUTSet` feature. For instance `Set1`에 적절한 값을 지정하여 액세스할 조회 테이블 데이터 세트를 선택하십시오.
- `LUTIndex` 기능을 0로 설정합니다.
- `LUTReadBlockLength` 기능을 `LUTLength`가 리턴한 값으로 설정하십시오.
- `LUTValue` 기능으로부터 `LUTReadBlockLength`값의 문자열을 가져옵니다.



참고애플리케이션은 개별적으로 또는 임의의 연속적인 엔트리 블록을 선택적으로 판독할 수 있습니다.

조회 테이블 사용

조회 테이블 사용함:

- `LUTEnable` 기능을 사용할 조회 테이블 데이터를 지정하는 값으로 설정하십시오.

설정 스크립트 예제

다음 스크립트는 단색 8비트에서 8비트 연산에 대한 조회 테이블을 구성하고 다른 조회 테이블 정의 방법을 사용하여 4개의 조회 테이블 데이터 세트를 정의 및 업로드하는 방법을 보여주는 예입니다.

```
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. 바이엘 CFA 디코딩



경고

이 기능은 제한된 제품 및 펌웨어 변형 조합에만 사용할 수 있습니다!

참조: "Firmware Variants per Product" 페이지20 및 고급 처리 열에서 CFA
을 검색하십시오.

바이엘 CFA에서 RGB 전환	203
Bayer CFA 디코더 사용	207

바이엘 CFA에서 RGB 전환

Bayer CFA 디코더는 카메라에서 발행 한 원시 Bayer CFA 데이터 스트림을 RGB 색상 데이터 스트림으로 변환합니다.

누락된 픽셀 구성 요소는 다음 보간 방법 중 하나를 사용하여 가장 가까운 구성 요소로부터 재구성됩니다.

- 레거시 보간 방법은 `Mean()` 함수를 배타적으로 적용하여 누락된 색상 구성 요소를 계산합니다.
- 고급 보간 방법은 `Mean()` 및 `Median()` 함수를 사용하여 누락된 색상 구성 요소를 계산합니다. 고급 보간은 이미지의 고 대비 선명한 전환에 대한 앨리어싱 효과를 제거합니다.

함수 정의

이 `min()` 함수는 2개의 정수값 집합에서 가장 낮은 정수값을 반환합니다.

이 `max()` 함수는 2개의 정수값 집합에서 가장 높은 정수값을 반환합니다.

이 `mean()` 함수는 2 개의 정수의 평균값을 나타내는 정수값 하나를 반환합니다. 다음과 같이 계산됩니다:

함수	Mean(a, b)
공식	$(a + b + 1) \gg 1$

이 `median()` 함수는 네 개의 정수로 구성된 두 개의 중간값인 두 개의 정수값 집합을 반환합니다. 다음과 같이 계산됩니다:

함수	Median(a, b, c, d)
값1 공식	<code>Min [Max(a, b); Max(c, d)]</code>
값2 공식	<code>Max [Min(a, b); Min(c, d)]</code>

디코더 동작

소스 이미지의 각 픽셀에 대해 CFA 디코더는 주변 픽셀에서 누락된 두 색상 구성 요소를 계산합니다.

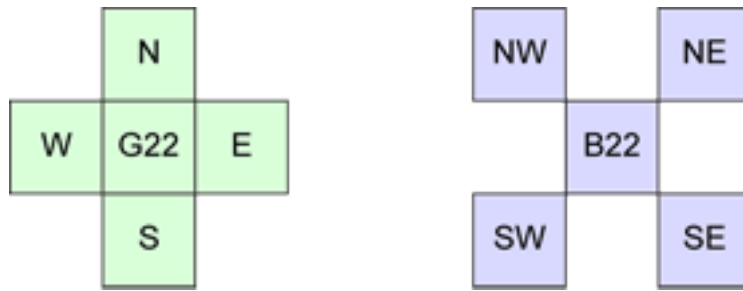
이하 4 x 4 베이어 CFA 배열의 22, 32, 23 및 33 위치에 있는 4개의 중심 픽셀을 계산하는 방법을 설명합니다:

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

주변 픽셀의 상대적 위치는 방위 표시로 식별됩니다:

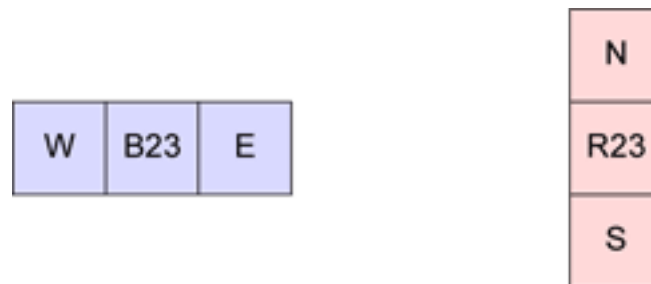
NW	N	NE
W		E
SW	S	SE

위치 22에 대한 수식



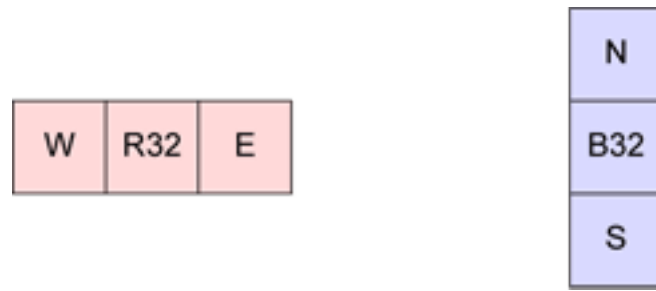
구성 요소	방법	공식
R22	레거시, 고급	R22
G22	레거시	Mean[Mean(N,S), Mean(W,E)]
	고급	Mean[Median(N, S, E, W)]
B22	레거시	Mean[Mean(NW, SW), Mean(NE, SE)]
	고급	Mean[Median(NW, SW, NE, SE)]

위치 23에 대한 수식



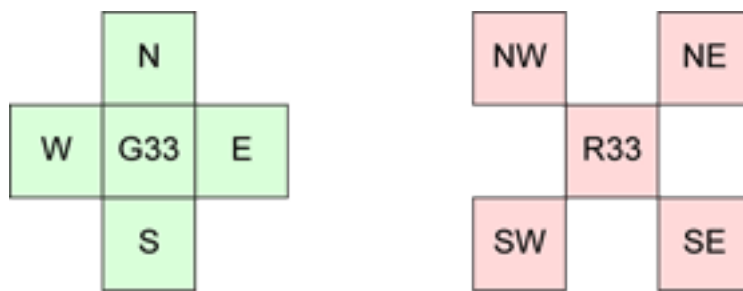
구성 요소	방법	공식
R23	레거시, 고급	Mean(N,S)
G23	레거시, 고급	G23
B23	레거시, 고급	Mean(W,E)

위치 32에 대한 수식



구성 요소	방법	공식
R32	레거시, 고급	$\text{Mean}(W,E)$
G32	레거시, 고급	G32
B32	레거시, 고급	$\text{Mean}(N,S)$

위치 33에 대한 수식



구성 요소	방법	공식
R33	레거시 고급	$\text{Mean}[\text{Mean}(NW, SW), \text{Mean}(NE, SE)]$ $\text{Mean}[\text{Median}(NW, SW, NE, SE)]$
G33	레거시 고급	$\text{Mean}[\text{Mean}(N,S), \text{Mean}(W,E)]$ $\text{Mean}[\text{Median}(N, S, E, W)]$
R33	레거시, 고급	B33

Bayer CFA 디코더 사용

선결 요건

1. 프레임 그래버

Coaxlink 카드 및 펌웨어 변형(CFA 디코더 포함). "Firmware Variants per Product" 페이지20를 열고 고급 처리 열에서 CFA를 검색하십시오.

2. 카메라

a. 바이엘 CFA 지역 스캔

b. 한 줄당 최대 16384 픽셀

c. 첫 번째로 전송된 첫 번째 전송된 픽셀의 처음 두 개의 전송 픽셀과 픽셀 비트 깊이(8 비트, 10 비트, 12 비트, 14 비트 또는 16 비트)의 색상 등록(GR, RG, GB 또는 BG) 비트)을 CoaXPress 이미지 헤더의 `PixelF` 필드에 올바르게 지정합니다.



경고 필드 및/또는 가 0보다 큰 `Xoffsets` 메라는 전송된 `Xoffsets` 해당하는 값을 보고해야 합니다! `Xoffsets`

바이엘에서 RGB 픽셀 처리 구성

Bayer CFA 디코더가 활성화되면:

- "픽셀 구성 요소 언패킹" 페이지68 제어가 작동하지 않으면 Coaxlink 카드가 10비트, 12비트 또는 14비트 픽셀을 lsb로 언팩합니다
- "픽셀 구성 요소 순서 재정렬" 페이지70 기능을 사용하면 빨간색 및 파란색 구성 요소를 교환하고 BGR 또는 RGB 픽셀을 전달할 수 있습니다.

Input Pixel Format	FFC	RedBlueSwap	Output Pixel Format
Bayer**8	off on	off	RGB8
		on	BGR8
Bayer**10pmsb	off on	off	RGB10
		on	BGR10
Bayer**12pmsb	off on	off	RGB12
		on	BGR12
Bayer**14pmsb	off on	off	RGB14
		on	BGR14
Bayer16	off on	off	RGB16
		on	BGR16

바이엘 CFA 디코더 활성화

데이터 스트림 모듈에서 원하는 보간 방법에 따라 기능 값을
또 BayerMethodBayerMethodBayerMethod

바이엘 CFA 디코더 비활성화

데이터 스트림 모듈에서 기능 값을 로 설정하십시오. BayerMethodBayerMethod

바이엘 CFA 디코딩 성능

제품	펌웨어 변형	픽셀 처리 속도
1633 Coaxlink Quad G3 1633-LH Coaxlink Quad G3 LH	1-카메라	1,000 메가픽셀/초
1635 Coaxlink Quad G3 DF	1-df-카메라	1,000 메가픽셀/초
3602 Coaxlink Octo	1-카메라	2,000 메가픽셀/초
	2-카메라	스트림당 1,000 메가픽셀/초
3603 Coaxlink Quad CXP-12	1-카메라	2,000 메가픽셀/초
3620 Coaxlink Quad CXP-12 JPEG	4-카메라	stream0에서 500 메가픽셀/초



참고 Bayer CFA 디코더의 픽셀 처리 속도는 RGB8 또는 BGR8 픽셀을 전송할 때의 DMA 전송 성능과 거의 일치합니다. 이는 보드가 달성할 수 있는 최대 집계 입력 데이터 속도보다 현저히 낮습니다.



경고 유효 픽셀 처리 속도는 PCI Express 가용 대역폭이 충분하지 않거나 매우 짧은 라인을 처리할 때 픽셀 처리 속도와 크게 다를 수 있습니다.

PCI Express 성능

PCI Express 인터페이스	지속 가능한 PCIe 데이터 전송률	RGB8	RGB10, RGB12, RGB14, RGB16
4-레인 Rev 3.0 PCIe 엔드 포인트	3,350 MB/s 일반	~1,117 Mpixels/초	~ 558 Mpixels/초
8-레인 Rev 3.0 PCIe 엔드 포인트	6,700 MB/s 일반	~2,238 Mpixels/초	~1,117 Mpixels/초



참고 PCI 인터페이스는 8 비트 비트 깊이의 병목 현상이 아닙니다!



경고 10 비트, 12 비트, 14 비트 및 16 비트 비트 심도의 경우 지속 가능한 데이터 출력 속도는 호스트 PC의 [PCI Express Interface](#) 성능에 의해 더 제한됩니다.

지연 속도

하드웨어 CFA 디코더는 데이터 처리량이 사용 가능한 PCI Express 대역폭에 의해 제한되지 않을 때 무시할 수 있는 대기 시간을 사용하여 직접 변환을 수행합니다!

11.7. JPEG 인코딩

Applies to:

QuadCXP12J



경고

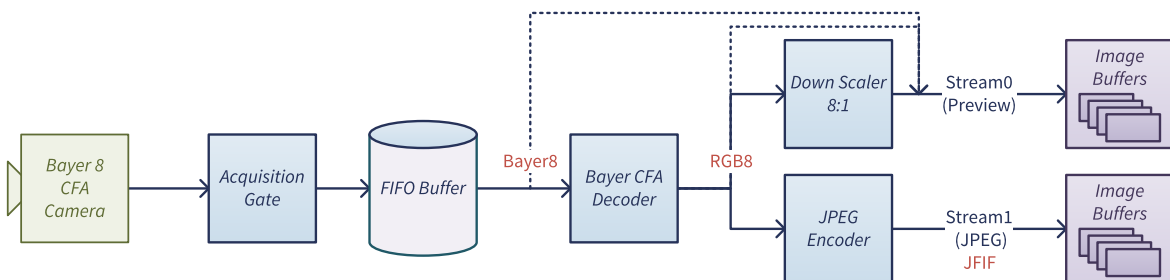
이 기능은 제한된 제품 및 펌웨어 변형 조합에만 사용할 수 있습니다!

참조: "Firmware Variants per Product" 페이지20 및 고급 처리 열에서 JPEG을 검색하십시오.

3620 Coaxlink Quad CXP-12 JPEG 는 임베디드 CFA 디코더 및 JPEG 인코더가 장착된 4 연결 CoAXPress CXP-12 프레임 그래버입니다.

4 카메라 펌웨어 버전은 다음과 같이 구성된 4개의 독립적인 채널을 구현합니다:

- PoCXP 및 카메라 트리거링 기능이 있는 1-연결 CoAXPress CXP-12 호스트 인터페이스,
- 캡처할 프레임을 선택하는 획득 게이트,
- 대형 탄성 버퍼를 제공하는 512MB FIFO 버퍼,
- 고급 알고리즘을 사용하여 Bayer8 카메라 데이터 스트림을 RGB8 스트림으로 변환하는 Bayer CFA 디코더,
- RGB8-YUV422 변환기 프런트 엔드 및 기본 ISO/IEC 10918-1 호환 JPEG 인코더가 포함된 JPEG 인코더,
- 양방향으로 RGB8 이미지 해상도를 8배까지 감소시키는 다운 스케일러.



카메라 요구 사항

- 8 비트 Bayer CFA 영역 스캔 컬러 카메라,
- 이미지 해상도 (H x V) : 128 x 16에서 5120 x 3840까지, 폭 및 높이는 8의 배수,
- CXP-1부터 CXP-12 CoaXPress까지의 인터페이스.

채널 사양

- 픽셀 처리 속도: 최대 250메가픽셀/초
- 출력 스트림: 2 ('미리보기' 및 'JPEG')
- 인코더 컨트롤: JPEG 품질

미리보기 스트림 사양

- 픽셀 형식: RGB8 또는 Bayer8
- 해상도: '전체' 또는 '8:1'(RGB8만 해당)

JPEG 스트림 사양

- JPEG 컨트롤: 1에서 100까지의 품질 (기본값: 75)
- 픽셀 형식: CustomJFIF
- 해상도: '원본'
- 대기 시간: 일반적으로 20라인

11.8. 레이저 선 추출

Applies to:

Quad3DLLE



경고

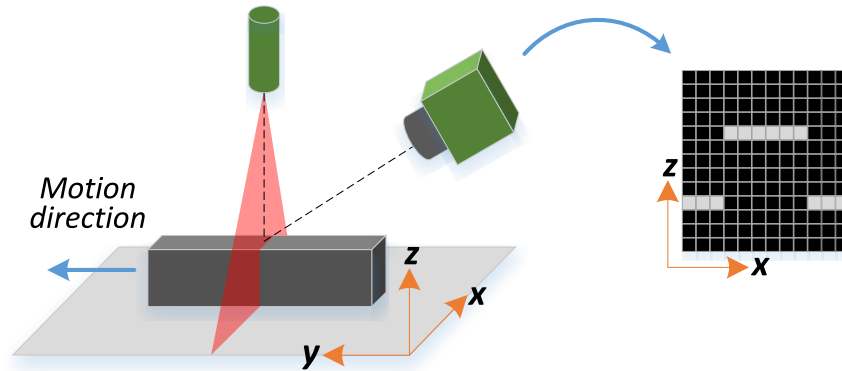
이 기능은 제한된 제품 및 펌웨어 변형 조합에만 사용할 수 있습니다!

참조: "Firmware Variants per Product" 페이지20 및 고급 처리 열에서 LLE
를 검색하십시오.

소개	213
코어 구현을 처리하는 LLE	215
사용 사례의 예	224
부록	231

소개

레이저 라인 삼각 측량 시스템에서, 레이저 라인(또는 임의의 다른 "좁은 가벼운 스트라이프" 생성 방법)이 3D 오브젝트 상에 투영됩니다. 레이저가 아닌 다른 시점에 배치된 카메라를 사용하여 객체의 모양으로 변형된 해당 선의 이미지를 캡처합니다. 선의 변형은 레이저 선의 평면에서 3D 객체의 모양을 직접 표현한 것입니다. 물체를 스캔하여 레이저 라인 아래로 움직이고 여러 이미지를 찍으면 3D 모양을 재구성할 수 있습니다.



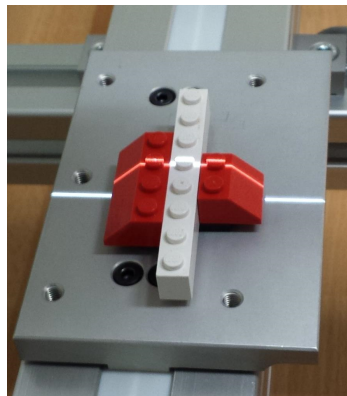
간소화된 레이저 삼각 측량 시스템

레이저 라인 추출 알고리즘

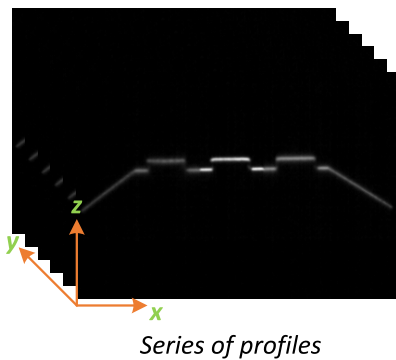
카메라 센서로 캡처 된 객체의 프로파일 시퀀스를 기반으로 *깊이 맵*을 생성하려면 레이저 라인 추출(LLE) 알고리즘이 필요합니다.

LLE 알고리즘의 목적은 레이저 선이 관심 영역 (ROI)을 수평으로 가로 지르는 위치를 추정하는 것입니다. 감지는 프레임의 각 열을 개별적으로 분석하여 수행할 수 있습니다.

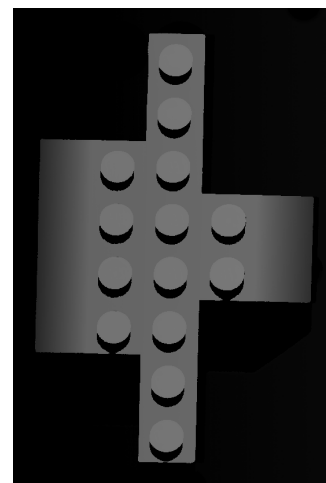
LLE 알고리즘은 전형적으로 ROI를 따라 검출된 레이저 라인의 수직 위치를 포함하는 데이터 어레이를 출력합니다. 즉, 각각의 계산된 ROI는 단일 데이터 어레이를 생성합니다.



Measured object



LLE



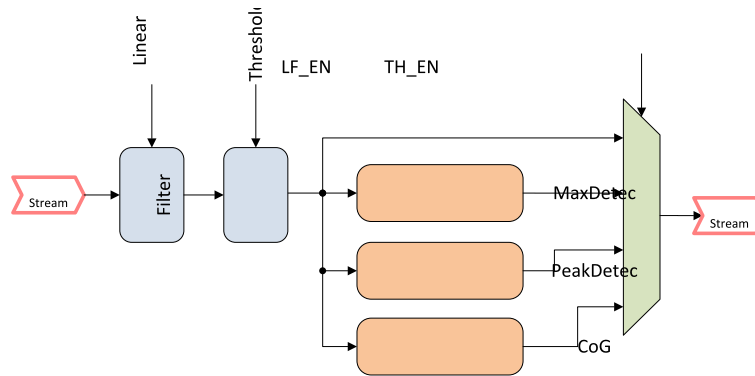
Depth Map

깊이 맵 생성

코어 구현을 처리하는 LLE

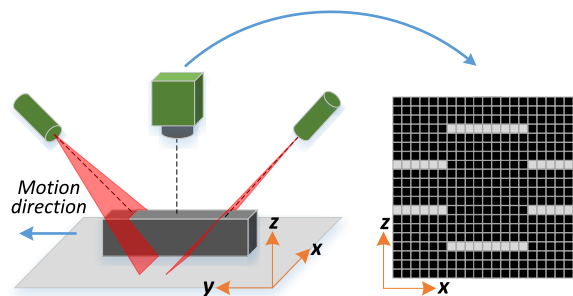
LLE 처리 코어는 **1637 Coaxlink Quad 3D-LLE** 프레임 그래버에 내장되어 있습니다. 호스트 CPU 사용률을 제로로 하고 지연 시간을 제로로 하여 측정 대상의 깊이 맵을 계산할 수 있습니다.

레이저 라인 추출 (최대 감지, 피크 감지, 중력의 중심)과 필터링 및 임계 값 처리를 위한 전처리 단계를 위한 3가지 알고리즘을 제공합니다.



LLE 프로세싱 코어의 단순화된 블록 다이어그램

LLE 프로세싱 코어는 또한 단일 입력 ROI로부터 최대 2개의 깊이 맵을 동시에 계산할 수 있습니다. 이 기능을 *Dual Laser Line Extraction (Dual-LLE)* 이라고 합니다. 이 기능은 동일한 물체에 투사된 2개의 레이저 라인을 사용하는 응용 프로그램에 유용합니다.



간소화된 이중 레이저 삼각 측량 시스템

Dual-LLE 모드가 활성화되면 LLE 프로세싱 코어 입력은 입력 ROI를 LLE-ROI라고 하는 두 개의 하위 ROI로 분할합니다. 각 LLE-ROI는 LLE 프로세싱 코어에 의해 독립적으로 처리되어 해당 깊이 맵을 생성합니다.

코어 특징을 처리하는 LLE

절대 최대 LLE-ROI XSize

- 모든 알고리즘: 8192 픽셀

최대. 유효 LLE-ROI YSize [픽셀]

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

출력 형식 [GenICam PFNC v2.1]

MaxDetec 8	MaxDetec 16	PeakDetec 11_5	PeakDetec 8_8	CoG 11_5	CoG 8_8
8비트 (정수)	16비트 (정수)	UQ11.5	UQ8.8	UQ11.5	UQ8.8

출력 픽셀 형식

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

출력 하위 픽셀 해상도 [픽셀]

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 값

- MaxDetec 8 알고리즘: 0x00
- 다른 알고리즘: 0x0000



참고이 값은 유효하지 않은 결과를 나타냅니다.

유효 출력 범위

- MaxDetec 8 알고리즘: 0x01 ~ 0xFF
- 다른 알고리즘: 0x0001 ~ 0xFFFF

지원되는 입력 픽셀 형식

- 모든 알고리즘: Mono8

레이저 라인 수

- 모든 알고리즘: LE-ROI 당 1개(입력 ROI 당 최대 2개)

최대 성능

- 모든 알고리즘: 2.500 메가 픽셀/초
- 모든 알고리즘: 2048 x 256 또는 4096 x 128 이미지에서 9,500 프로필/초
- 모든 알고리즘: 2048 x 128 이미지에서 19,000 프로필/초
- 모든 알고리즘: 1024 x 128 이미지에서 38,000 프로필/초
- 모든 알고리즘: 1024 x 64 이미지에서 76,000 프로필/초



참고위의 수치는 4- 연결 CXP-6 CoaXPress 링크의 최대 대역폭을 기준으로 하며 이중 LLE 모드가 활성화된 것으로 간주합니다.

사용 가능한 출력 유형

- 모든 알고리즘:
 - 깊이 맵
 - 원시 카메라 이미지

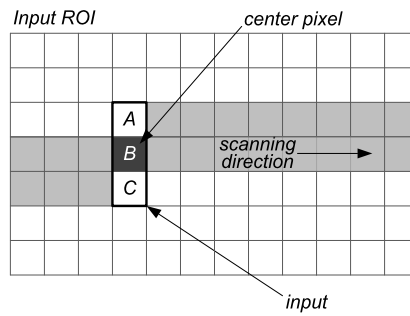


경고두 출력이 동시에 출력되는 것은 아닙니다!

선형 필터

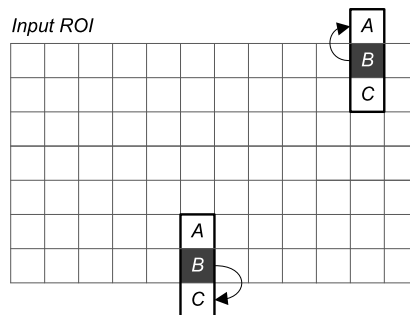
선형 필터 모듈은 카메라에서 오는 대로 데이터 흐름에 1x3 슬라이딩 윈도우에 컨볼루션 연산자를 직접 적용합니다. 컨볼루션 커널 (A, B 및 C)의 3개의 요소는 3개의 요소의 합이 1과 512 사이의 값인 모든 양의 정수를 허용하도록 구성할 수 있습니다.

이 그림은 주어진 LLE-ROI 내에서 컨볼루션 커널 요소의 위치를 보여줍니다.



선형 필터 커널 배치

커널의 요소가 LLE-ROI 경계 (일반적인 슬라이딩 윈도우 문제) 밖에 있을 때 입력 픽셀값이 중앙 창 픽셀로 대체됩니다.



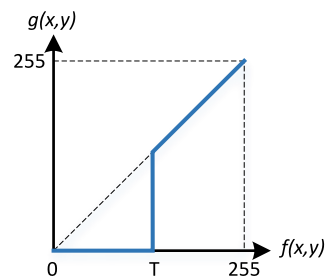
LLE-ROI 경계에서의 선형 필터 동작.

코어링 임계 값

코어링은 다음 규칙에 따라 픽셀을 두 가지 범주로 분류하는 간단한 분할 기법을 수행하는 비선형 연산자입니다.

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

T는 코어링 임계 값입니다.



코어링 반응

최대 감지

최대 감지 모듈은 8비트 또는 16비트 데이터 폭으로 깊이 맵을 출력하도록 구성할 수 있습니다. 이 두 모드의 차이점은 최대 탐지 모듈이 지원하는 최대 유효 LLE-ROI YSize입니다. 8비트 모드는 최대 255 픽셀의 높이를 표시할 수 있으며 16비트 모드는 최대 65535 픽셀의 높이를 나타낼 수 있습니다.

주어진 LLE-ROI 열에서 둘 이상의 픽셀에서 최대 강도가 감지되면 최대 감지 알고리즘은 가장 높은 위치의 것을 나타냅니다.

InvalidDataFlag

다음과 같은 경우 *InvalidDataFlag*가 생성됩니다.

- 감지된 라인 위치가 최대 LLE-ROI 높이보다 높습니다.
- LLE-ROI 열에서 MAX 강도가 감지되지 않습니다.

피크 감지

피크 감지 모듈은 16비트 고정 소수점 단어로 표현되는 깊이 맵을 생성합니다. 다음 두 가지 정밀도로 구성할 수 있습니다:

- 5 비트(LSB)가 소수 부분을 나타내고 11비트(MSB)가 정수 부분을 나타내는 *UQ11.5*. 이 모드에서 피크 감지 모듈이 지원하는 최대 유효 LLE-ROI YSize는 거의 2048 픽셀입니다.
- 8비트(LSB)가 소수 부분을 나타내고 8비트(MSB)가 정수 부분을 나타내는 *UQ8.8*. 이 모드에서 피크 감지 모듈이 지원하는 최대 유효 LLE-ROI YSize는 거의 256 픽셀입니다.

주어진 LLE-ROI 열에서 둘 이상의 피크가 검출되면 피크 검출 모듈은 해당 $f(x)$ 픽셀이 가장 높은 강도를 갖는 픽셀의 위치를 나타냅니다. 하나 이상의 해당 $f(x)$ 픽셀이 동일한 조건(가장 높은 강도)을 갖는 경우, 그 중 가장 높은 위치의 것이 표시됩니다.

InvalidDataFlag

다음과 같은 경우 *InvalidDataFlag*가 생성됩니다.

- 감지된 라인 위치는 지원되는 최대 LLE-ROI 높이보다 높습니다.
- LLE-ROI 열에서 선이 감지되지 않습니다.

중력의 중심

중력의 중심 모듈은 16비트 고정 소수점 단어로 표현되는 깊이 맵을 생성합니다. 다음 두 가지 정밀도로 구성할 수 있습니다:

- 5 비트(LSB)가 소수 부분을 나타내고 11비트(MSB)가 정수 부분을 나타내는 *UQ11.5*. 이 모드에서는 Center of Gravity 모듈에서 지원하는 최대 유효 LLE-ROI YSize는 거의 2048 픽셀입니다.
- 8비트(LSB)가 소수 부분을 나타내고 8비트(MSB)가 정수 부분을 나타내는 *UQ8.8*. 이 모드에서는 Center of Gravity 모듈에서 지원하는 최대 유효 LLE-ROI YSize는 거의 256 픽셀입니다.

주어진 LLE-ROI 열에서 둘 이상의 피크를 감지하면, 중력 중심 모듈은 해당 $f(x)$ 픽셀이 가장 높은 강도를 갖는 위치를 나타냅니다. 하나 이상의 해당 $f(x)$ 픽셀이 동일한 조건(가장 높은 강도)을 갖는 경우, 그 중 가장 높은 위치의 것이 표시됩니다.

InvalidDataFlag

다음과 같은 경우 *InvalidDataFlag*가 생성됩니다.

- 감지된 라인 위치는 지원되는 최대 LLE-ROI 높이보다 높습니다.
- 오버플로는 내부 합계에서 발생합니다. 이 조건은 동일한 LLE-ROI 열에서 많은 수의 연속 픽셀이 임계값 레벨을 초과하는 강도를 나타낼 때 발생할 수 있습니다.
- LLE-ROI 열에서 선이 감지되지 않습니다.

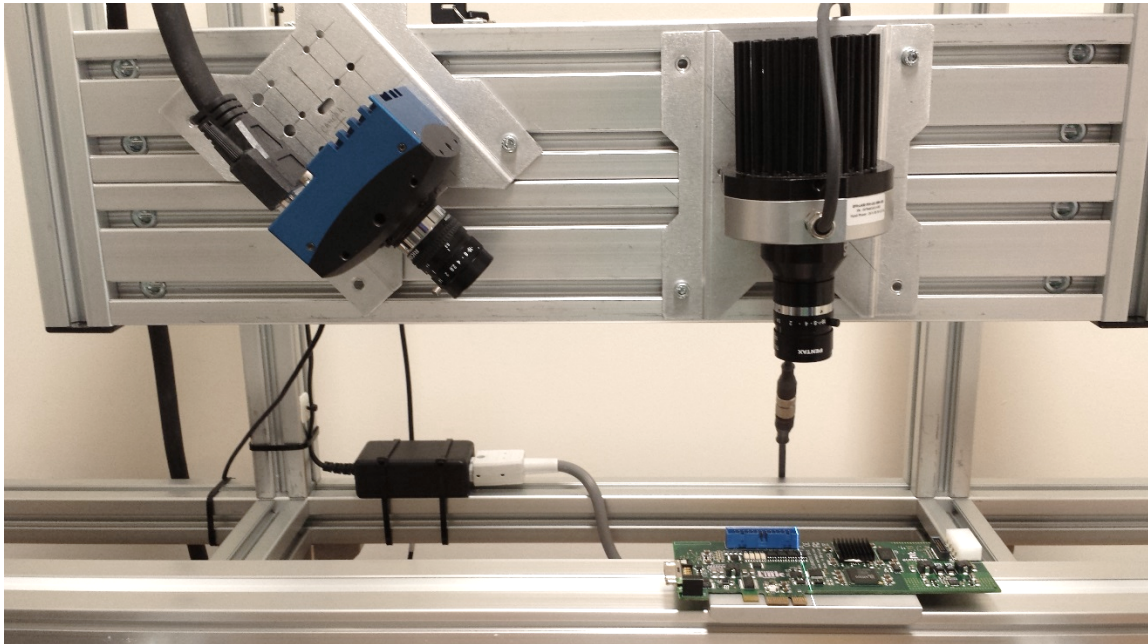
사용 사례의 예

오브젝티브	225
입력 ROI 정의	226
필터링 및 임계값 설정	227
LLE 알고리즘 정의	228
스캔 길이 및 버퍼 크기 정의	230

오브젝티브

이 예에서 오브젝티브는 PCB 표면의 신뢰할 수 있는 척도를 얻는 것입니다. 간략화를 위해 획득 매개 변수는 GenICam 브라우저를 사용하여 설정되며 카메라는 제어되지 않습니다(프리 런). 삼각 측량 기하학, 렌즈 및 레이저 색상 및 전력과 같은 다른 매개 변수는 이 예제에서 다루지 않습니다.

이 그림은 이 예제에서 사용 된 레이저 라인 삼각 측량 설정을 보여줍니다.

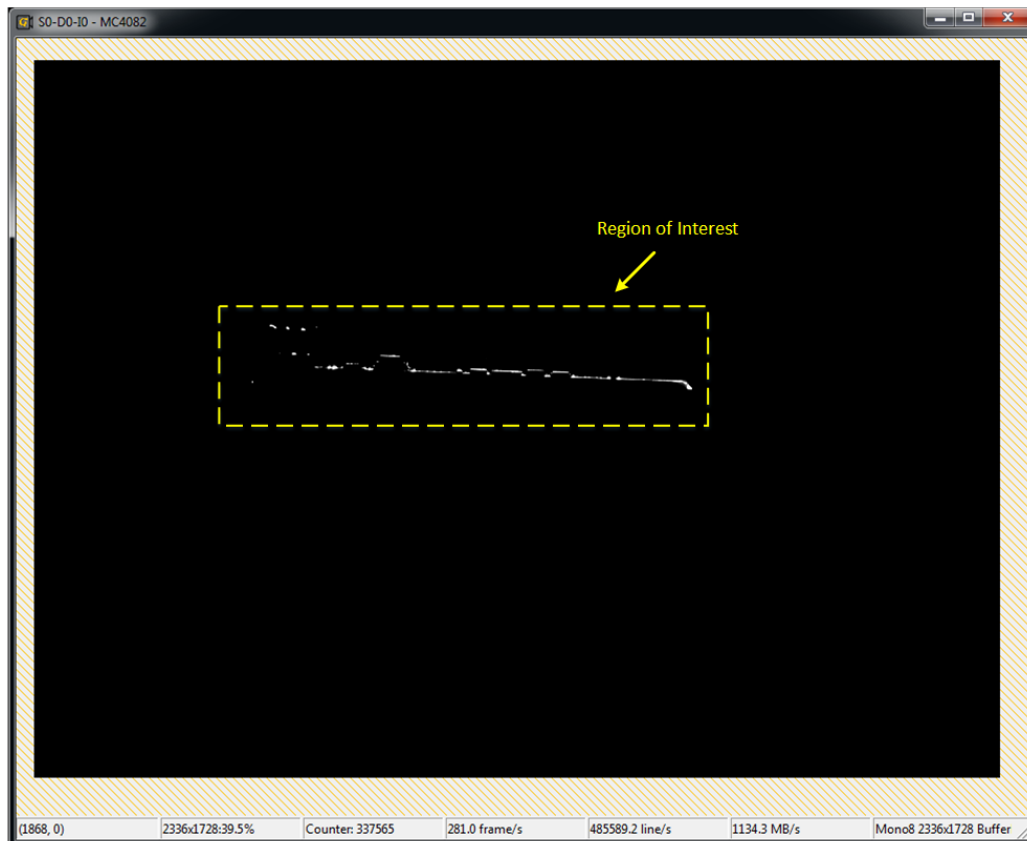


레이저 라인 삼각 측량 설정

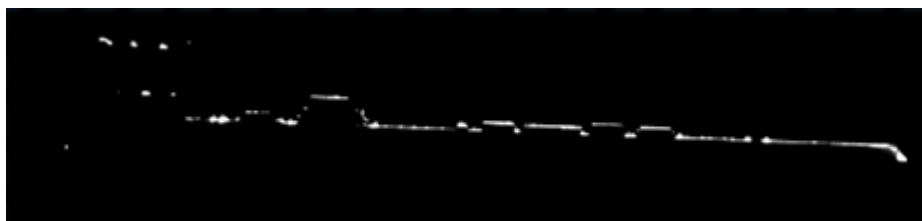
입력 ROI 정의

정의된 입력 ROI는 레이저 라인 변형이 발생하는 전체 센서 영역을 포함해야 합니다. 초기 단계는 전체 해상도 ROI를 설정하여 전체 이미지를 확인하는 것입니다.

결과 프레임은 다음 그림에서 볼 수 있는데, 여기서 레이저 라인 변형이 발생하는 유효 영역을 확인할 수 있습니다.



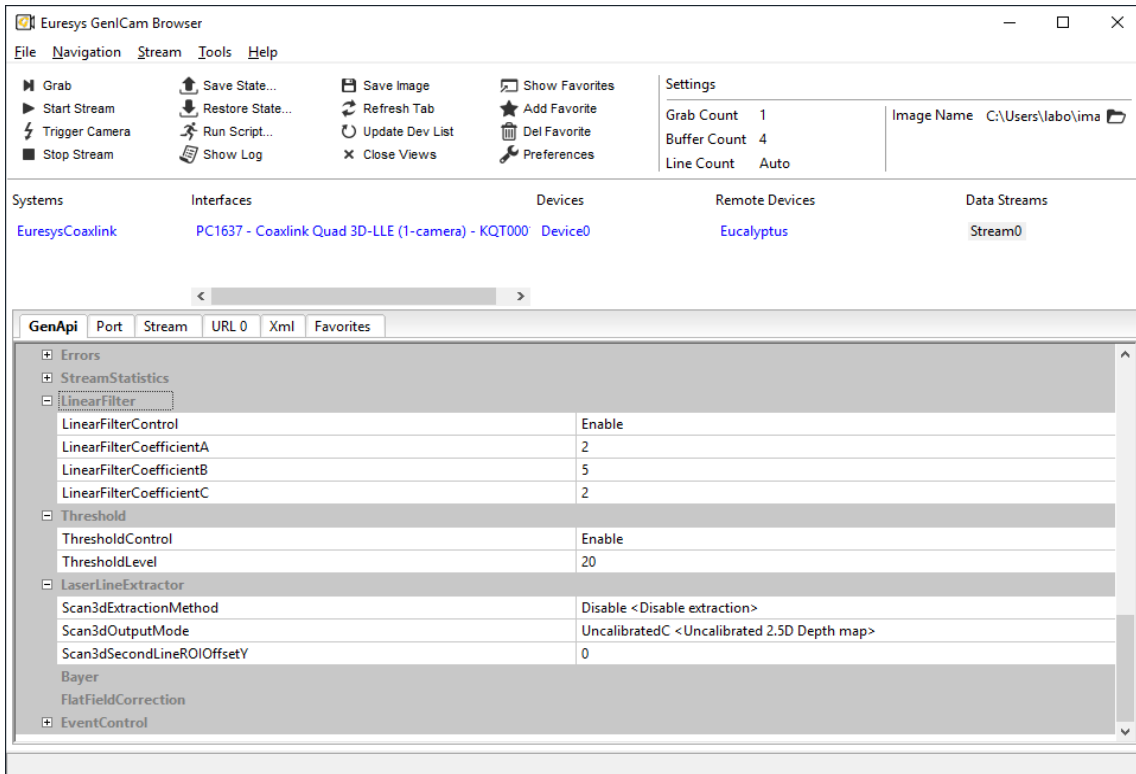
결과 이미지 (전체 해상도 ROI)



입력 ROI 확장된 보기

필터링 및 임계값 설정

LLE 프로세스에서 사전 처리 단계를 활성화할 수 있습니다. 아래의 예에서 임계 레벨은 20으로 설정되고 회선 필터 커널은 A=2, B=5 및 C=2로 설정됩니다. 이 매개 변수를 사용하면 임계 레벨 아래의 배경 노이즈가 0으로 대체된 부드러운 이미지를 얻을 수 있습니다.

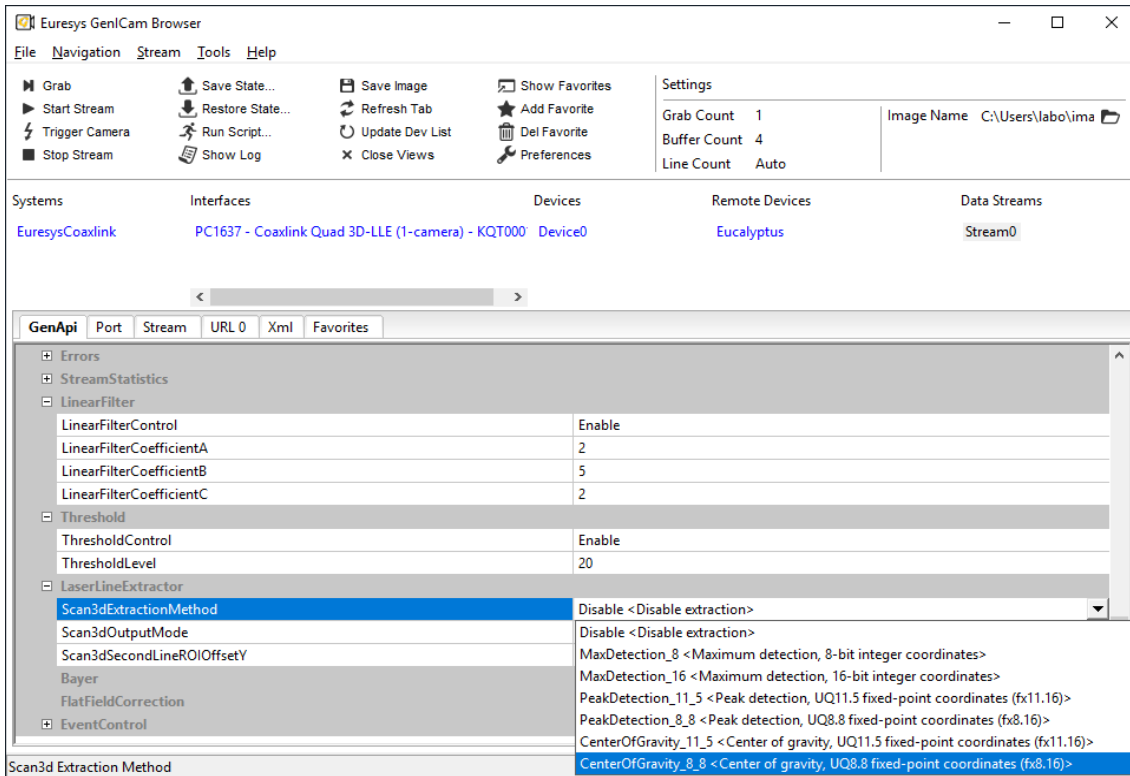


필터링 및 임계 매개 변수

LLE 알고리즘 정의

GenICam 기능 **Scan3dExtractionMethod**는 사용 가능한 모든 LLE 알고리즘에 대한 액세스를 제공합니다.

다음 그림은 이 기능이 GenICam 브라우저에서 어떻게 나타나는지 보여줍니다.



레이저 라인 추출 옵션

사용 가능한 알고리즘 사이의 선택은 대상 응용 프로그램과 관련된 여러 요소에 따라 다릅니다. 사용 가능한 알고리즘 간의 주요 차이점을 여기에서 강조합니다.

LLE-ROI YSize

LLE-ROI (YSize)의 유효 세로 픽셀의 수는 아래에 설명된 대로 선택한 알고리즘의 출력 형식으로 표현할 수 있는 최대 정수 값으로 제한됩니다.

- **MaxDetection_8**(255 픽셀).
 - 0x00 값은 `InvalidDataFlag`를 나타내기 위해 예약되어 있습니다.
- **MaxDetection_16F** (65535 픽셀).
 - 0x0000 값은 `InvalidDataFlag`를 나타내기 위해 예약되어 있습니다.
- **PeakDetection_11_5**(거의 2048 픽셀).
 - 0x0000 값은 `InvalidDataFlag`를 나타내기 위해 예약되어 있습니다.
- **PeakDetection_8_8**(거의 256 픽셀).

- 0x0000 값은 InvalidDataFlag를 나타내기 위해 예약되어 있습니다.
- CenterOfGravity_11_5의 2048 픽셀).
 - 0x0000 값은 InvalidDataFlag를 나타내기 위해 예약되어 있습니다.
- CenterOfGravity_8_8의 256 픽셀).
 - 0x0000 값은 InvalidDataFlag를 나타내기 위해 예약되어 있습니다.

알고리즘 해상도

사용 가능한 알고리즘의 위치 해상도는 아래에 설명 된대로 1픽셀에서 1/256픽셀까지 다양합니다.

- MaxDetection_8(8 비트) : 1 픽셀.
- MaxDetection_16(16 비트) : 1 픽셀.
- PeakDetection_11_5.
- PeakDetection_8_8셀.
- CenterOfGravity_11_5 (UQ11.5): 1/32 픽셀.
- CenterOfGravity_8_8 (UQ8.8): 1/256 픽셀.

알고리즘의 특이성

앞서 언급했듯이 LLE 알고리즘의 선택은 대상 응용 프로그램에 따라 크게 달라집니다. 다음은 애플리케이션 관점에서 각 LLE 알고리즘의 주요 기능에 대한 설명입니다.

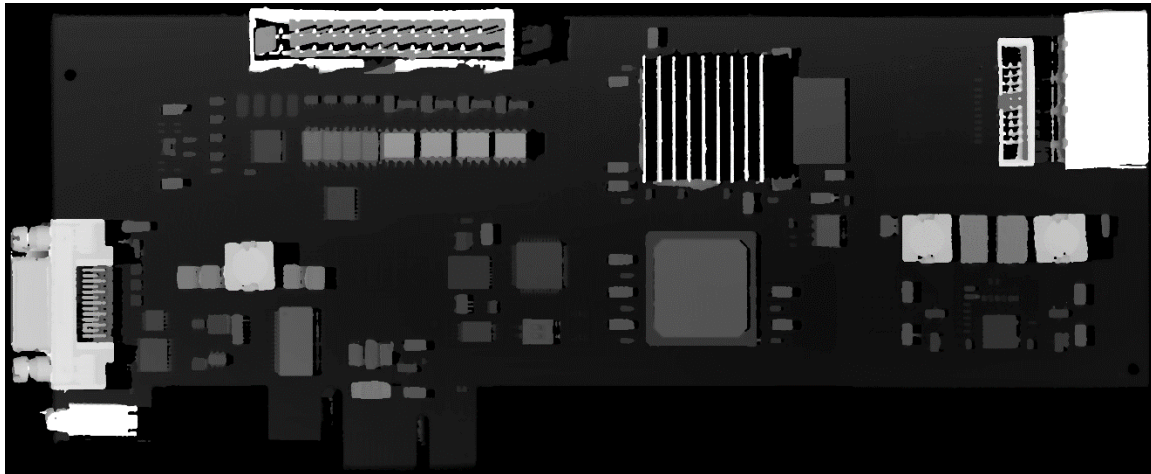
- MaxDetection_8이 맵 출력이 있는 이 버전의 MaxDetection 모듈은 3D 후 처리 단계에서 계산 성능이 덜 필요한 매우 적은 양의 데이터를 생성합니다.
- MaxDetection_16수 깊이 맵 출력으로 MaxDetection 모듈 버전은 매우 큰 LLE-ROI를 지원할 수 있습니다.
- PeakDetection 모듈의 두 가지 버전은 하위 픽셀 해상도와 최대 유효 LLE-ROI 크기 사이의 균형을 제공합니다. PeakDetection은 강도 의존성 바이어스에 둔감하도록 설계되었습니다.
- CenterOfGravityGravity 모듈은 하위 픽셀 해상도와 최대 유효 LLE-ROI 크기 간의 균형을 제공합니다. CenterOfGravity는 노이즈 많은 입력 및 넓은 라인에 대해 견고하도록 설계되었습니다.

스캔 길이 및 버퍼 크기 정의

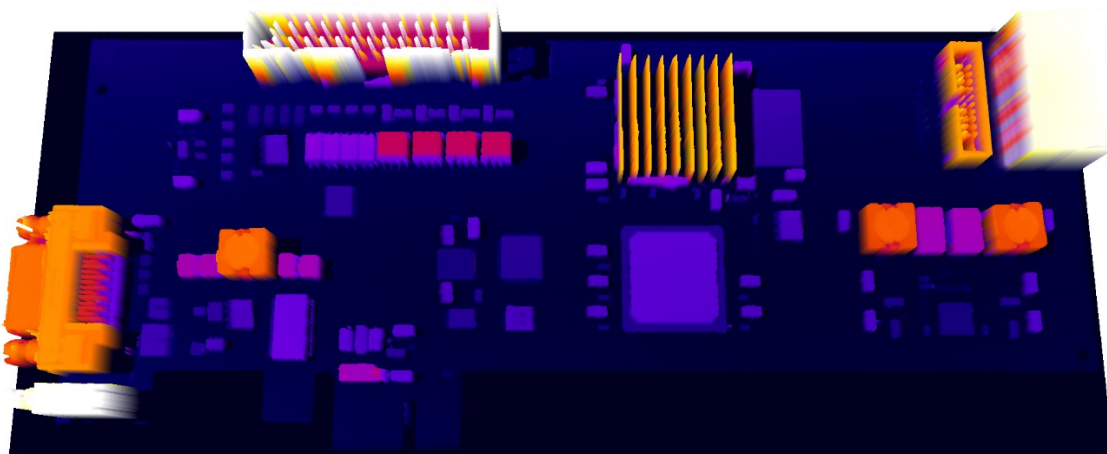
ScanLength와 BufferHeight는 주어진 객체로부터 획득할 프로파일의 수에 따라 달라집니다. 스캔되는 물체의 크기, 카메라 프레임 속도 및 물체 이동 속도는 ScanLength와 함께 정의되는 필수 매개 변수입니다.

스캔 결과

측정된 PCB의 깊이 맵은 다음 그림에서 볼 수 있습니다.



깊이 맵 예제



깊이 맵 예제: 3D 표면 플롯

부록

고정 소수점 "Q-형식" 표기법

UQm.n 시스템은 Q 형식의 고정 소수점 숫자 표현입니다. 여기서 Q는 Q 형식 표기법의 숫자를 지정하고, U 선행 Q는 부호없는 값을 나타내며, m은 정수 형식의 정수 부분 비트 수를 나타냅니다. 번호, n은 분수 부분의 번호를 나타냅니다. 예를 들어, UQ11.5는 부호없는 고정 소수점 값으로, 11 비트 정수와 5 소수 비트가 있는 16비트 단어로 표현됩니다. UQ11.5에 대한 다른 유효한 표기법은 fx11.16과 0:11:5입니다.

"UQ11.5" 고정 소수점 값을 부동 소수점으로 변환하는 방법

레이저 라인 추출기로 생성된 UQ11.5 고정 소수점 값은 11정수 비트 및 5소수 비트가 있는 값을 나타내는 16비트 단어입니다.

이 형식을 float로 변환하려면 먼저 값을 float로 변환 한 다음 $2^5 = 32$ 로 나눕니다.

```
float toFloat(unsigned short x) {
float f = x;
return f / 32;
}
```

"UQ8.8" 고정 소수점 값을 부동 소수점으로 변환하는 방법

레이저 라인 추출기로 생성된 UQ8.8 고정 소수점 값은 8정수 비트 및 8소수 비트가 있는 값을 나타내는 16비트 단어입니다.

이 형식을 float로 변환하려면 먼저 값을 float로 변환 한 다음 $2^8 = 256$ 으로 나눕니다.

```
float toFloat(unsigned short x) {
float f = x;
return f / 256;
}
```

11.9. 라인 스캔 수집



경고

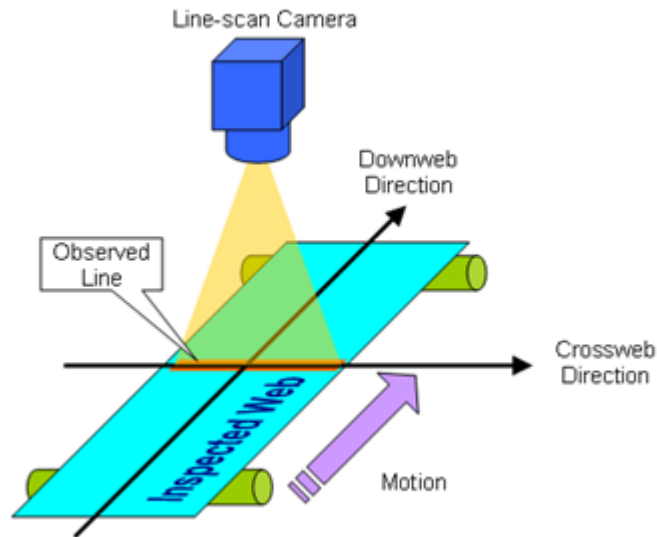
이 기능은 제한된 제품 및 펌웨어 변형 조합에만 사용할 수 있습니다!

참조: "Firmware Variants per Product" 페이지20를 입력하고 펌웨어 변형 열에서 ", line-scan" 접미사를 검색하십시오.

라인 스캔 수집 원칙	233
라인 스캔 수집 사례	237

라인 스캔 수집 원칙

라인 스캔 이미징



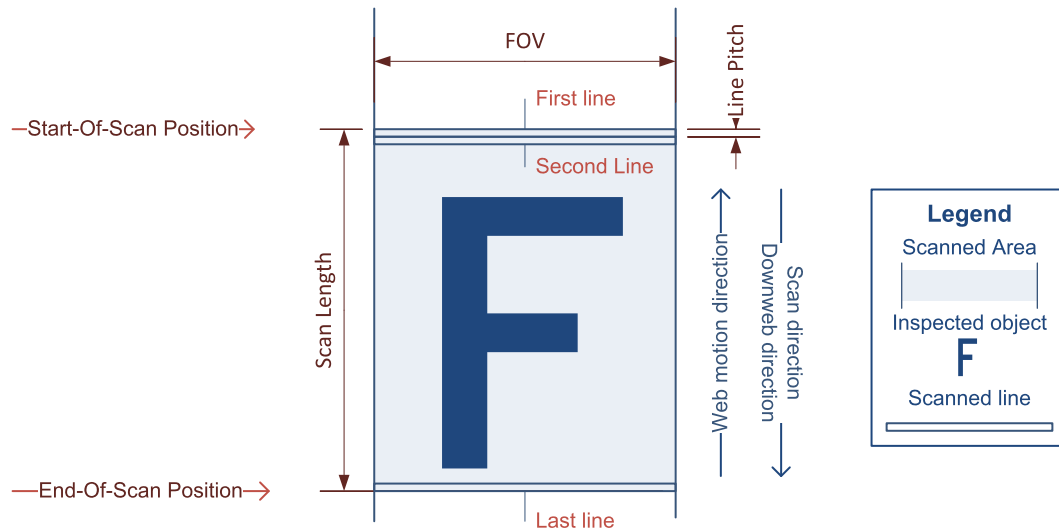
일반적인 라인 스캔 이미징 시스템

"라인 스캔 이미징"이라는 표현은 대상과 상대적으로 이동하는 1차원 이미징 장치에서 캡처한 연속 이미지 라인의 조합으로 2차원 이미지를 얻는 머신 비전 응용 프로그램을 나타냅니다.

라인 스캔 이미징에서:

- 이미징 장치는 종종 라인 스캔 카메라일 수는 있지만 반드시 그런 것은 아닙니다.
- 검사 대상은 종종 연속 웹이며 고정 된 크기 또는 가변 크기를 갖는 개별 객체일 수도 있습니다.
- 검사된 웹은 카메라에 상대적으로 움직입니다. 수집 중 모션 속도는 고정되거나 가변적일 수 있습니다.
 - 교차 웹 방향 또는 가로 방향은 카메라가 관측하는 웹 평면상의 축입니다.
 - 하향 웹 방향 또는 축 방향은 카메라에 대해 상대적으로 검사된 웹의 운동 방향입니다.

스캐닝 영역



스캐닝 영역 정의

스캐닝 영역은 너비가 FOV이고 길이가 스캔 길이와 동일한 웹상의 2차원 영역입니다.

교차 웹 방향(위 그림의 수평 방향)에서 스캐닝 영역은 카메라의 시야(FOV)로 구분됩니다.

하향 웹 방향 (위 그림에서 수직 방향)에서, 스캐닝 영역은 스캔 시작 및 스캔 종료 위치에 의해 구분됩니다. 라인 피치는 웹 속도와 카메라 라인 속도 사이의 비율에 의해 결정됩니다.

라인 스캔 카메라의 *시야 - FOV*- 는 광학 설정과 센서 기하학적 특성에 의해서만 결정됩니다.

*스캔 시작 위치*는 첫 번째 수집 라인 앞에 있는 스캔 라인 경계에 해당하는 웹 상의 위치입니다.

*스캔 중단 위치*는 마지막 수집 라인 다음의 스캔 라인 경계에 해당하는 웹 상의 위치입니다.

대부분의 라인 스캔 카메라는 매 카메라 사이클마다 한 줄의 픽셀을 제공합니다. 결과적으로, *여러 카메라 사이클*은 객체 이미지를 구축하는 데 필요합니다.

픽셀 중횡비 제어

영역 스캔 이미징과는 달리, 라인 스캔 이미징을 사용하면 애플리케이션이 이미지 픽셀 중횡비를 제어할 수 있습니다.

대다수의 경우, 이미징 어플리케이션은 일정하고, 1:1 이미지 픽셀 중횡비를 선호하여 요구합니다.

크로스 웹 피치는 센서 피치 및 광학 배율에 의해 고정되며, 이미지 픽셀 중횡비는 라인 피치 제어를 통해서만 제어할 수 있습니다.

다음 표는 Coaxlink에 적용할 수 있는 일정한 라인 피치를 제공하는 방법을 요약한 것입니다.

메서드 이름	설명
VCR	<p>가변 카메라 주기.</p> <p>웹 속도는 가변적이며 카메라 순환 속도는 웹 속도에 비례하여 유지됩니다. 프레임 그래버는 카메라가 전달한 모든 연속 라인을 캡처하여 객체 이미지를 만듭니다.</p>
CCC	<p>일정한 카메라 주기.</p> <p>카메라는 일정한 사이클 속도로 작동하고 프레임 그래버는 카메라가 전달한 모든 연속 라인을 캡처합니다.</p>

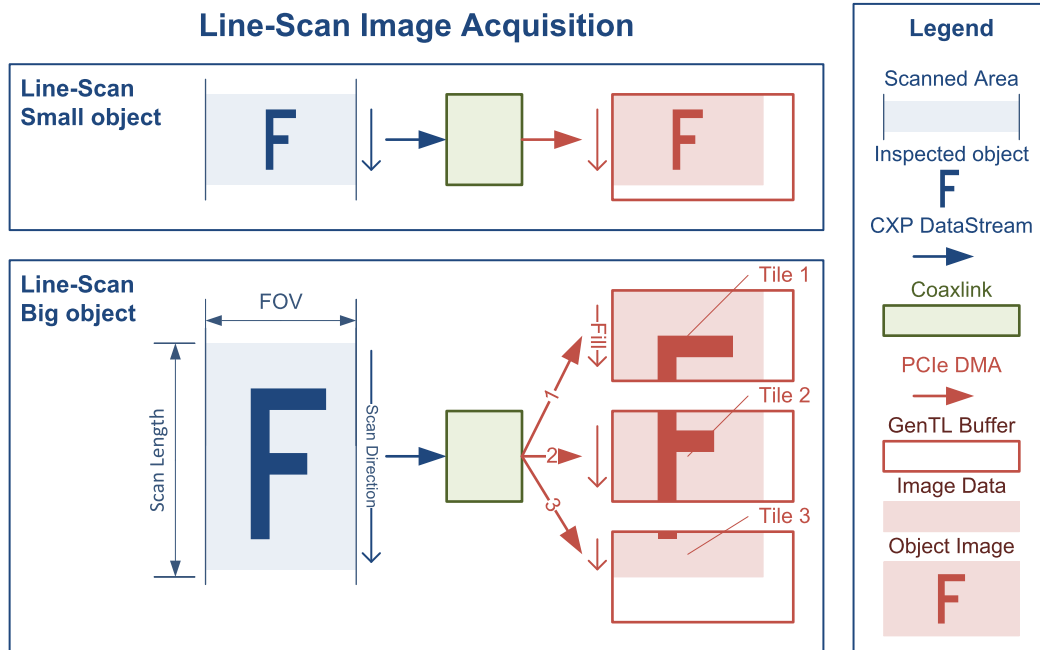
VCR 방법에는 다음이 필요합니다:

- 웹 속도를 측정하기 위한 *모션 인코더*.
- 모션 인코더 이벤트의 *실시간 처리*로 모션 인코더 이벤트 속도에 *비례*하는 속도로 카메라 트리거를 구축합니다.

비례율은 *Divider* 도구 또는 *Multiplier/Divider* 도구를 통해 얻을 수 있습니다:

- Divider 도구는 입력값을 정수값으로 데시 메이트하여 N개의 들어오는 이벤트 중 1개를 전달합니다.
- Multiplier/Divider 도구를 사용하면 RCR 값의 0.1%보다 우수한 정확도로 0.001에서 1000 사이의 비율 변환 비율 값(RCR)을 허용하여 이미지 픽셀 중횡비를 정밀하게 제어할 수 있습니다.

라인 스캔 이미징 장치를 사용한 이미지 수집



크고 작은 물체의 이미지 캡처

CoaxPress 링크를 통한 전송의 경우 (대부분의) 라인 스캔 카메라는 하나의 CoaxPress 이미지 데이터 스트림을 사용합니다.

이미지 데이터의 전달 방법과 관련하여 두 가지 경우를 고려해야 합니다.

- 작은 객체의 경우 객체 이미지 데이터가 단일 GenTL 버퍼로 전달됩니다.
- 대형 객체의 경우, 객체 이미지 데이터가 여러 GenTL 버퍼로 전달됩니다.

두 경우 모두 이미지 데이터는 단일 PCIe DMA 채널을 통해 전달되며 Coaxlink를 통한 전송 지연은 낮습니다: "하나의 이미지 라인".

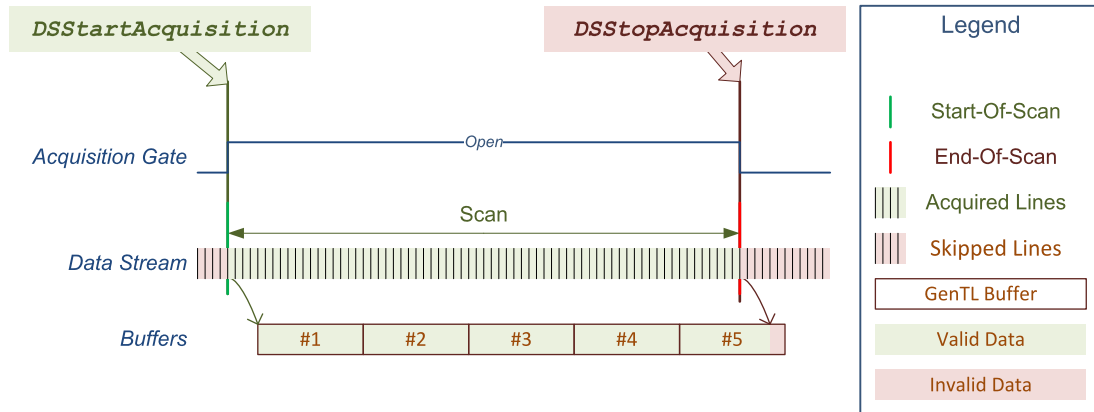
GenTL 버퍼 채우기 규칙 - 라인 스캔 카메라

라인 스캔 이미징에서 GenTL 버퍼는 다음 규칙에 따라 채워집니다.

- 스캔의 첫 번째 수집된 라인 데이터는 기본적으로 새 버퍼의 시작 부분에 저장됩니다. **StripeArrangement**를 **Geometry_1X_1YE**로 설정하여 세로 이미지 반전을 사용하면 스캔의 첫 번째 수집된 라인 데이터가 새 버퍼의 마지막 전체 라인 위치에 저장됩니다.
- 버퍼는 정수 개의 이미지 라인 데이터를 포함합니다.
- 버퍼의 나머지 공간이 이미지 라인 데이터를 저장하기에 충분하지 않을 때, 획득은 새로운 버퍼로 계속되고, 채워진 버퍼는 처리를 위해 애플리케이션에 이용 가능해집니다.
- 스캔의 마지막 라인 데이터가 수집되면, 부분적으로 채워질 수 있는 마지막 버퍼가 처리를 위해 애플리케이션에서 사용 가능하게 됩니다.

라인 스캔 수집 사례

연속 물체 스캐닝



연속 물체 스캐닝

이 경우는 **연속 물체**의 이미지 스캐닝에 적용됩니다.

Coaxlink 수집 컨트롤러는 다음과 같이 구성됩니다:

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

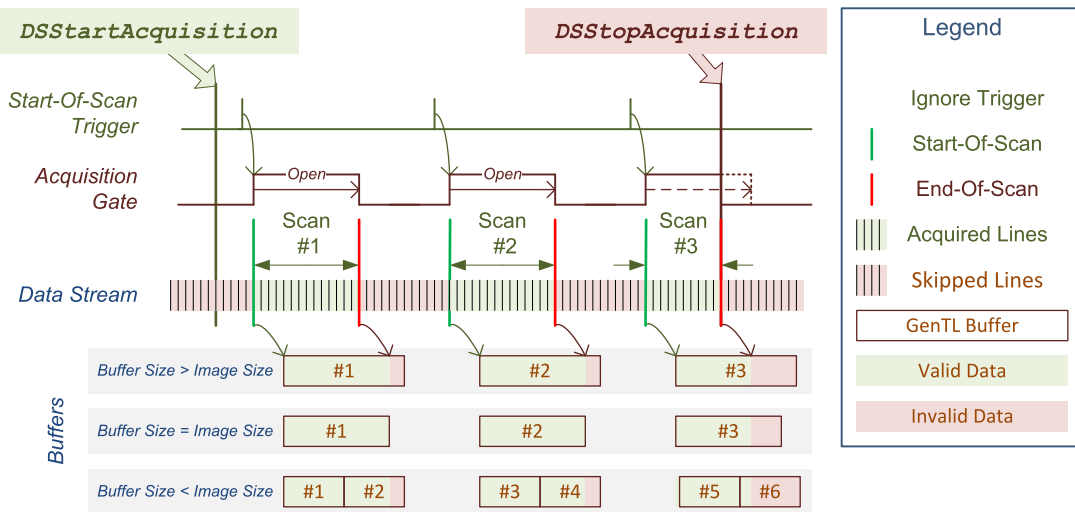
`DSStartAcquisition` 함수가 호출되면 다음 라인 경계에서 스캔이 시작됩니다.

응용 프로그램이 `DSStopAcquisition` 함수를 호출하면 수집 게이트가 닫힙니다.

할당된 버퍼 크기 및 스캔 지속 시간에 따라 객체 이미지는 단일 버퍼에 적합하거나 여러 버퍼가 필요합니다.

각 버퍼는 채워지자마자 애플리케이션에 전달됩니다. 부분적으로 채워질 가능성이 있는 마지막 버퍼는 마지막 이미지 데이터가 쓰여지자마자 전달됩니다.

불연속 물체의 고정 길이 스캐닝



공통 스캔 길이를 가진 개별 객체 스캔

Coaxlink 수집 컨트롤러는 다음과 같이 구성됩니다:

- StartOfScanTriggerSource = StartScan 또는 적용 가능한 I/O 도구 상자 벤트 출력. 예: LIN1.
- EndOfScanTriggerSource = ScanLength.
- ScanLength는 오브젝트 이미지를 완전히 캡처하는 데 필요한 라인 를 나타내는 양수입니다.

DSStartAcquisition 함수가 호출되면 수집 컨트롤러의 스캔 시작 트리거가 활성화됩니다. 그런 다음 수집 컨트롤러는 유효한 스캔 시작 트리거 이벤트가 처음 발생할 때까지 대기합니다.

유효한 스캔 시작 이벤트가 생성될 수 있습니다.

- StartScan 령을 사용하는 응용 프로그램.
- 선택한 하드웨어 이벤트 소스별로 StartOfScanTriggerSource 지정되는 경우.

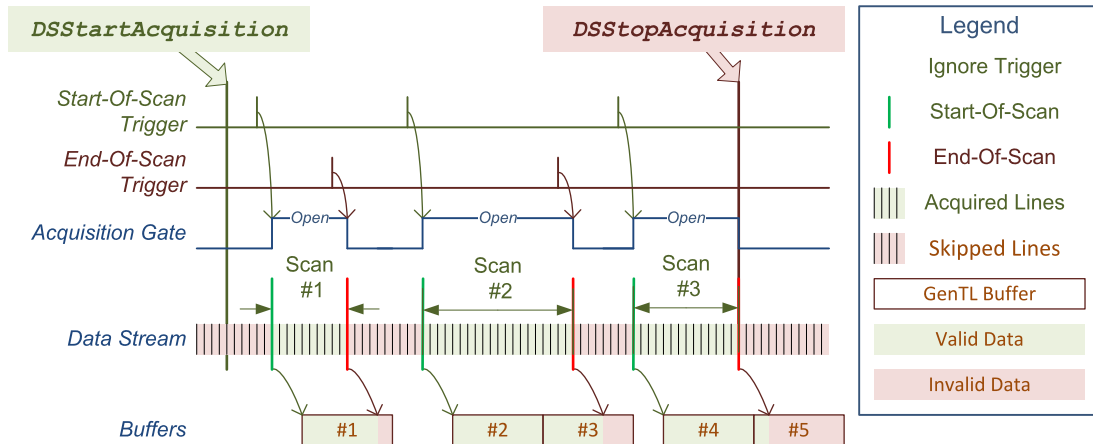
수집 컨트롤러는 스캔이 진행되는 동안 스캔 시작 트리거 이벤트를 무시합니다.

수집 게이트는 스캔 시작 이벤트 다음의 첫 번째 라인 경계에서 열립니다.

수집 게이트는 지정된 라인 수를 획득한 후 또는 응용 프로그램이 DSStopAcquisition 함수를 호출할 때 자동으로 닫힙니다.

할당된 버퍼 크기에 따라 객체 이미지는 단일 버퍼에 적합하거나 여러 버퍼가 필요합니다. 각 버퍼는 채워지자마자 애플리케이션에 전달됩니다. 엔드 오브 스캔에서 부분적으로 채워진 버퍼가 즉시 전달됩니다. 다음 이미지 획득은 항상 새 버퍼로 시작합니다.

불연속 물체의 가변 길이 스캐닝



다양한 스캔 길이가 필요한 개별 객체 스캔

Coaxlink 수집 컨트롤러는 다음과 같이 구성됩니다:

- **StartOfScanTriggerSource = StartScan** 또는 적용 가능한 I/O 도구 상자 벤트 출력. 예: **LIN1**.
- **EndOfScanTriggerSource = StopScan** 또는 적용 가능한 I/O 도구 상자 벤트 출력. 예: **LIN2**.

`DSStartAcquisition` 함수가 호출되면 수집 컨트롤러의 스캔 시작 트리거가 활성화됩니다. 그런 다음 수집 컨트롤러는 유효한 스캔 시작 트리거 이벤트가 처음 발생할 때까지 대기합니다.

유효한 스캔 시작 이벤트가 생성될 수 있습니다.

- **StartScan** 령을 사용하는 응용 프로그램.
- 선택한 하드웨어 이벤트 소스별로 **StartOfScanTriggerSource** 지정되는 경우.

수집 컨트롤러는 스캔이 진행되는 동안 스캔 시작 트리거 이벤트를 무시합니다.

수집 게이트는 스캔 시작 이벤트 다음의 첫 번째 라인 경계에서 열립니다.

수집 게이트는 유효한 스캔 종료 이벤트를 따르거나 응용 프로그램이 `DStopAcquisition` 함수를 호출할 때 바로 첫 번째 경계에서 닫힙니다.

유효한 엔드 오브 스캔 이벤트가 생성될 수 있습니다:

- **StopScan** 령을 사용하는 응용 프로그램.
- 선택한 하드웨어 이벤트 소스별로 **EndOfScanTriggerSource** 지정되는 경우.

수집 컨트롤러는 스캔이 진행되는 동안 스캔 중단 트리거 이벤트를 무시합니다.

할당된 버퍼 크기에 따라 객체 이미지는 단일 버퍼에 적합하거나 여러 버퍼가 필요합니다. 각 버퍼는 채워지자마자 애플리케이션에 전달됩니다. 엔드 오브 스캔에서 부분적으로 채워진 버퍼가 즉시 전달됩니다. 다음 이미지 수집은 항상 새 버퍼로 시작합니다.

11.10. C2C- 링크

C2C 링크 상호 연결	241
C2C-링크 전기 사양	243
트리거 전파 지연	245
주기 트리거 동기화	247
C2C-링크 설정 순서	250

C2C 링크 상호 연결

C2C 링크는 단일 C2C 링크 마스터 장치가 여러 C2C 링크 슬레이브 장치와 트리거 이벤트를 안정적으로 공유할 수 있도록 하는 하드웨어 통신 매체입니다.

영역 스캔 응용 프로그램에서 C2C 링크 인터커넥트:

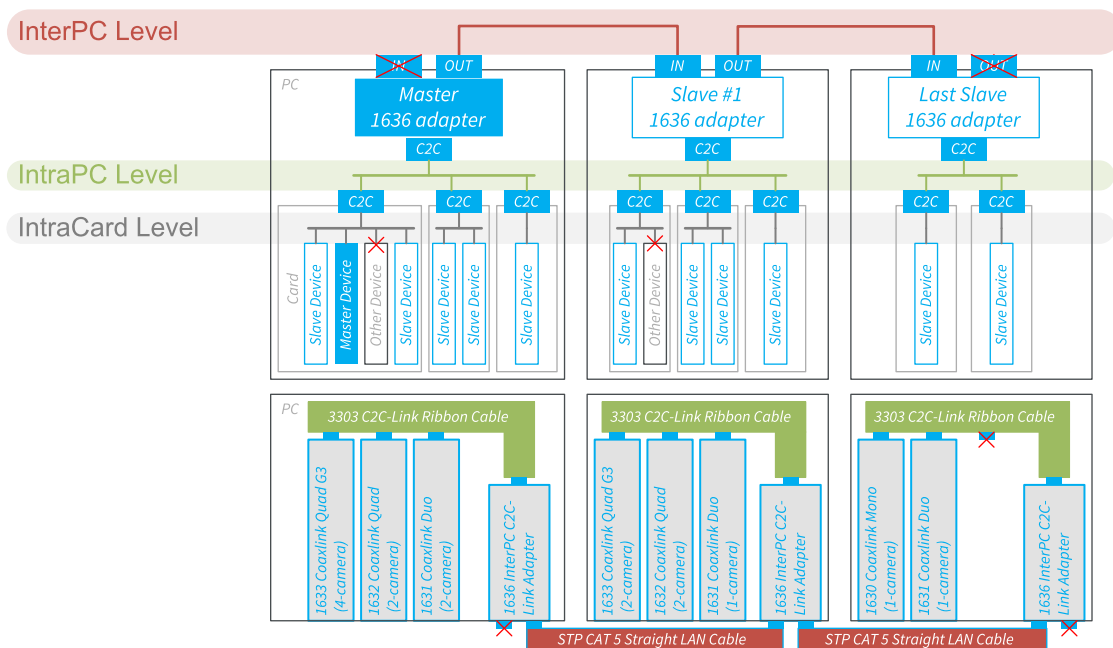
- 최대 2개의 트리거를 공유할 수 있습니다: CIC주기 트리거(필수) 및 하나의 I/O 도구 상자 이벤트(옵션).
- C2C 링크 장치가 C2C 링크 마스터 장치에 트리거를 승인할 수 없음을 표시할 수 있습니다.

라인 스캔 응용 프로그램에서 C2C 링크 인터커넥트는 최대 3개의 트리거를 공유할 수 있습니다: CIC 사이클 트리거(필수) 및 2개의 I/O 도구 상자 이벤트(옵션).

C2C-링크는 동일한 Coaxlink 카드에 속한 장치나 같은 Pc에 있는 다른 카드 또는 다른 Pc에 있는 다른 카드에 연결된 장치를 상호 연결할 수 있습니다.

C2C-링크 상호 연결은 최대 세 가지 상호 연결 수준을 결합할 수 있습니다.

- *IntraCard Level*은 FPGA 내부 리소스를 사용하여 동일한 카드에 속한 둘 이상의 C2C-링크 장치를 상호 연결합니다.
- *IntraPC Level*은 동일한 PC의 두 개 이상의 카드에서 C2C-링크 장치를 상호 연결합니다. **3303 C2C-Link Ribbon Cable** 또는 맞춤형 C2C-Link 케이블과 같은 액세스리 케이블이 각 PC에 필요합니다.
- *IntraPC Level*은 둘 이상의 PC에서 C2C-링크 장치를 상호 연결합니다. 각 어댑터에는 **1636 InterPC C2C-Link Adapter** 가 필요하고 마지막 어댑터는 하나의 RJ 45 CAT 5 STP 스트레이트 LAN 케이블이 필요합니다.



InterPC, IntraPC 및 IntraCard 레벨을 사용한 C2C 링크 설정 예

C2C-링크 전기 사양

정의

트리거 지연

마스터 장치에서 슬레이브 장치로의 트리거 신호 전파 지연. 이 지연은 전자 장치 (FPGA, 어댑터 ...) 내부의 전파 지연과 상호 연결 케이블로 구성됩니다.

케이블의 경우 지연은 케이블 길이에 비례합니다: 일반적으로 5 ns/m.

트리거 지연 왜곡

C2C-링크에 속한 모든 장치에 트리거 지연 값이 분산됩니다.

트리거 지연 지터

온도, 신호 잡음 등과 같은 외부 요인에 따른 트리거 지연의 변화...

트리거 비율

반복적인 트리거 이벤트의 발생률. 역수(1/트리거 비율)는 연속 트리거 사이에 필요한 최소 시간 간격입니다.

인트라 C2C-링크 상호 연결 수준

매개 변수	최소	유형	최대.	단위
트리거 지연 지터			5	ns
트리거 비율	0		2.5	MHz

인트라PC C2C-링크 상호 연결 수준

매개 변수	최소	유형	최대.	단위
케이블 커넥터 수 (1636 어댑터 포함)	2		4	-
케이블 길이	-		0.6	m
트리거 지연 지터	5		10	ns
트리거 비율	0		2.5	MHz

인터PC C2C-링크 상호 연결 수준

다음 사양은 가장 높은 트리거 속도가 필요한 응용 프로그램을 대상으로 합니다:

매개 변수	최소	유형	최대.	단위
1636 어댑터 수	2		10	-
어댑터-어댑터 LAN 케이블 길이	0.3		100	m
누적된 어댑터-어댑터 LAN 케이블 길이	0.3		100	m
트리거 지연 지터	10	20	40	ns
트리거 비율	0		2.5	MHz

다음 사양은 도달 가능한 가장 긴 거리가 필요한 애플리케이션을 대상으로 합니다:

매개 변수	최소	유형	최대.	단위
1636 어댑터 수	2		32	-
어댑터-어댑터 케이블 길이	0.3		1200	m
누적된 어댑터-어댑터 케이블 길이	0.3		1200	m
트리거 지연 지터			1	μ s
트리거 비율	0		200	kHz



참고 최대 트리거 속도 사양은 길이 \times 주파수 곱이 일정하다고 가정할 때 중간 거리 100m에서 1200m까지 외삽될 수 있습니다: 이 경우 250 [m. MHz]입니다.

트리거 전파 지연

마스터 장치에서 슬레이브 장치로의 트리거 신호 전파 지연은 신호 경로의 각 세그먼트에서 발생하는 일반적인 지연을 추가하여 대략적으로 추정할 수 있습니다.

C2C-링크 세그먼트 당 일반 지연값

지연 요소	최소	유형	최대.	단위
(1) 인트라PC 상호 연결 (FPGA I/O 및 인트라PC 케이블을 포함한 전체 인트라PC C2C-링크 상호 연결)	0	5	10	ns
(2) 1636 InterPC C2C-Link Adapter - 인트라PC-인터PC 지연		15		ns
(3) 1636 InterPC C2C-Link Adapter - 인터PC-인트라PC 지연		20		ns
(4) 1636 InterPC C2C-Link Adapter - 인터PC-인터PC 지연		0		ns
(5) 인터PC LAN 케이블		5		ns/m

예 1 - 인트라PC 구성

인트라PC 전용 구성의 경우 고려할 지연 요소가 하나만 있습니다: (1)

매개 변수	최소	유형	최대.	단위
(1) 인트라PC 상호 연결	0	5	10	ns
총 트리거 지연	0	5	10	ns

예 2 - 3- 어댑터 인터PC 구성; 20m+20m LAN 케이블

이 구성은 3개의 인트라-PC 세그먼트로 구성됩니다. 마스터 디바이스와 동일한 인트라PC 세그먼트에 속한 디바이스의 경우, 고려해야 할 요소는 하나뿐입니다.

매개 변수	최소	유형	최대.	단위
(1) 인트라PC 상호 연결	0	5	10	ns
마스터 인터PC 세그먼트의 장치에 대한 총 트리거 지연	0	5	10	ns

슬레이브1 어댑터와 동일한 인트라PC 세그먼트에 속하는 장치에는 다음과 같은 5가지 지연 요소가 고려됩니다:

매개 변수	최소	유형	최대.	단위
(1) 마스터 인트라PC 상호 연결	0	5	10	ns
(2) 마스터 1636 InterPC C2C-Link Adapter - 인트라PC-인터PC 지연		15		ns
(5) 마스터 - 슬레이브1 인터PC LAN 케이블 - 20m		100		ns

매개 변수	최소	유형	최대.	단위
(3) 슬레이브1 1636 InterPC C2C-Link Adapter - 인터PC-인트라PC 지연		20		ns
(1) 슬레이브1 인트라PC 상호 연결	0	5	10	ns
슬레이브1 인터PC 세그먼트의 장치에 대한 총 트리거 지연		145		ns

슬레이브2 어댑터와 동일한 인트라PC 세그먼트에 속하는 장치에는 다음과 같은 7가지 지연 요소가 고려됩니다:

매개 변수	최소	유형	최대.	단위
(1) 마스터 인트라PC 상호 연결	0	5	10	ns
(2) 마스터 1636 InterPC C2C-Link Adapter - 인트라PC-인터PC 지연		15		ns
(5) 마스터 - 슬레이브1 인터PC LAN 케이블 - 20m		100		ns
(4) 슬레이브1 1636 InterPC C2C-Link Adapter - 인터PC-인터PC 지연		0		ns
(5) 슬레이브1-슬레이브2 인터PC LAN 케이블 - 20m		100		ns
(3) 슬레이브2 1636 InterPC C2C-Link Adapter - 인터PC-인트라PC 지연		20		ns
(1) 슬레이브2 인트라PC 상호 연결	0	5	10	ns
슬레이브2 인터PC 세그먼트의 장치에 대한 총 트리거 지연		245		ns

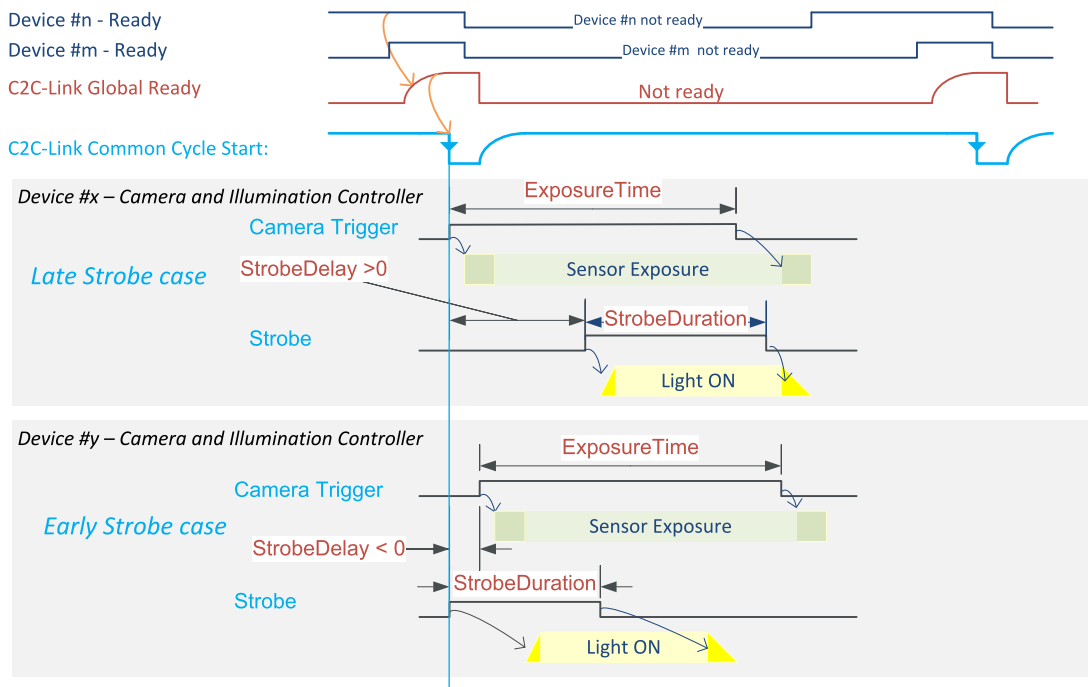
글로벌 준비 신호

글로벌 준비는 영역 스캔 펌웨어 변형에만 적용됩니다!

글로벌 준비 신호는 유선 및 스킴을 사용하여 모든 C2C-링크 슬레이브에 의해 정교해집니다. C2C-링크 슬레이브는 시작 조건이 모두 충족 될 때까지 글로벌 신호를 'false'로 만듭니다.

C2C-링크 마스터 장치는 전역 준비 신호가 false인 동안 사이클 트리거 이벤트 어서션을 더 지연시킵니다.

타이밍 다이어그램



C2C-링크 타이밍 다이어그램을 통한 CIC 동기화(글로벌 준비)

위의 다이어그램은 글로벌 준비와 함께 두 개의 연속적인 공통 CycleTrigger 이벤트의 타이밍 다이어그램을 보여줍니다.

C2C-링크 글로벌 준비 신호는 모든 참여 장치의 준비가 참이 될 때까지 낮게 유지되어 C2C-링크 마스터가 시작 이벤트를 발행하지 못하게 합니다. 모든 참여 디바이스에서 출시될 때 최대 100ns의 상승 시간으로 빠르게 상승합니다.

C2C-링크 글로벌 레디 신호가 높게 확인되자마자 마스터 디바이스는 공통 주기 시작 신호에서 갑작스러운 낮은 전환을 표명합니다. 이 에지는 모든 참여 디바이스의 타이머의 모든 시작 입력으로 전파됩니다.

주기가 시작 되자마자 모든 cic는 다음 주기를 시작하기 위한 모든 로컬 조건이 충족되지 않는 한 준비 상태를 강제로 낮게 설정합니다.

각 장치의 타이머는 각각의 지연 및 지속 시간 설정과 함께 Camera Trigger 및 스트로브를 발행합니다. 일반적으로 설정은 모든 참여 장치에서 동일합니다. 필요한 경우 응용 프로그램에서 다른 응용 프로그램을 적용할 수 있습니다.

C2C-링크가 허용하는 Cycle Start 400ns입니다. 2.5MHz의 이론적인 주파수 제한을 허용합니다.

C2C-링크 설정 순서

하드웨어 설정

이 단계는 각 C2C-링크 구성에 따라 다릅니다.

IntraCard C2C 링크 구성

이 설정은 FPGA 내부 리소스를 독점적으로 사용하여 C2C-링크 상호 연결을 구축합니다. 추가 하드웨어가 필요하지 않습니다!

IntraPC C2C-링크 구성

이 구성에는 각 Pc에 **3303 C2C-Link Ribbon Cable** 또는 맞춤형 C2C 링크 케이블과 같은 액세스 서리 케이블이 하나씩 필요합니다.

C2C-링크 케이블의 C2C-링크 암 커넥터를 각 참여 Coaxlink 카드의 C2C-링크 핀 헤더 커넥터에 삽입하십시오.

InterPC C2C-링크 구성

이 구성에는 각 어댑터마다 하나의 **1636 InterPC C2C-Link Adapter** 가 필요하고 마지막 어댑터는 하나의 RJ 45 CAT 5 STP 스트레이트 LAN 케이블이 필요합니다.

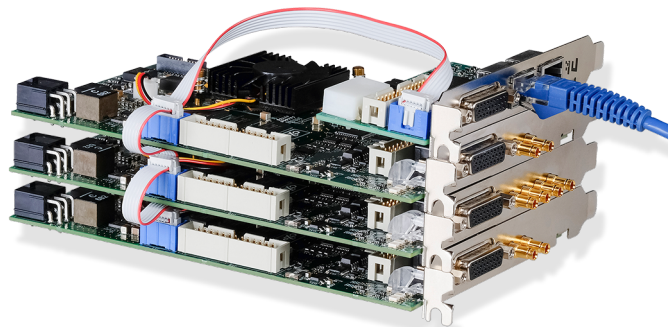
참여하는 각 PC에서:

1. **1636 InterPC C2C-Link Adapter** 를 빈 슬롯에 설치하고 브래킷을 고정하십시오.
2. 어댑터를 전원에 연결하십시오.

참조: 하드웨어 설명서의 1636섹션에 있는 "어댑터 전원 공급" 항목.

3. 하나의 **3303 C2C-Link Ribbon Cable** 를 사용하여, 모든 참여 카드의 C2C-링크 커넥터를 어댑터 카드의 C2C-링크와 함께 바인딩하십시오.
4. LAN 케이블을 사용하여 어댑터를 상호 연결하십시오.

참조: 하드웨어 설명서의 1636섹션에 있는 "InterPC 인터커넥트" 주제.



인트라PC 설정의 인트라PC 세그먼트

GenAPI 설정

마스터 C2C 링크 장치

- GenTL 장치 모듈의 `C2CLinkConfiguration`에 값 `Master` 를 지정하십시오.
- 사이클 트리거를 공유하도록 GenTL 인터페이스 모듈의 I/O 도구 상자 C2C1 도구를 구성하십시오:
 - `C2CLinkSynchronizationToolSource`에 값 `CycleTrigger` 를 지정하십시오.
 - `C2CLinkSynchronizationToolClock`에 값 `Immediate` 를 지정하십시오.
- 선택적으로 응용 프로그램 요구 사항에 따라 GenTL 인터페이스 모듈의 I/O 도구 상자 C2C2 및 C2C3 도구를 구성하십시오.

슬레이브 C2C 링크 장치

GenTL 장치 모듈의 `C2CLinkConfiguration`에 값 `Slave` 를 지정하십시오.

11.11. OEM 안전 키

OEM 안전 키 소개

OEM 안전 키 기능을 사용하면 응용 프로그램에서 다음을 수행할 수 있습니다.

- Coaxlink 카드의 비휘발성 메모리에 "OEM 안전 키"를 프로그램합니다.
- 방금 프로그래밍된 OEM 안전 키의 암호화된 버전을 검색합니다.
- 프로그래밍된 OEM 안전 키 또는 암호화된 버전에 대해 키를 확인합니다.

OEM 안전 키

OEM 안전 키는 응용 프로그램에서 정의한 문자열입니다. 공문자를 제외한 모든 문자가 허용됩니다. 문자열 길이는 무제한입니다.

키 프로그래밍

`ProgramOemSafetyKey` GenICam 기능에 OEM 안전 키 값이 있는 경우 Coaxlink 드라이버는 암호화된 OEM 안전 키 버전을 계산하여 Coaxlink 카드의 비휘발성 메모리에 저장합니다.

암호화된 값은 `ProgramOemSafetyKey` 설정한 직후의 `EncryptedOemSafetyKey` 값을 가져와서 검색할 수 있습니다.



경고 `ProgramOemSafetyKey`를 설정한 동일한 응용 프로그램 프로세스만 암호화된 값을 검색할 수 있습니다. 이것은 다른 GenICam 기능이 설정될 때까지 허용됩니다.

키 확인

응용 프로그램에서 `OemSafetyKeyVerification`의 값 하나를 선택해야 합니다.

Coaxlink 카드의 OEM 안전 키를 확인하기 위해 응용 프로그램에서 `CheckOemSafetyKey` [`selector`] 기능에 "도전"값을 설정합니다.

[선택기] 인수가 `EncryptedKey`로 설정되면 요청 문자열이 암호화된 OEM 안전 키 문자열과 동일한 경우에만 설정 작업이 정상적으로 종료됩니다.

[선택기] 인수를 `ProgrammingKey`로 설정하면 챌린지 문자열이 프로그래밍 OEM 안전 키 문자열과 동일한 경우에만 설정 작업이 정상적으로 종료됩니다.

[선택기] 인수를 `ProgrammingKeyOrEncryptedKey`로 설정하거나 생략하면 챌린지 문자열이 원래의 OEM 안전 키 문자열이나 암호화된 OEM 안전 키 문자열과 동일할 때만 설정 작업이 정상적으로 종료됩니다.

Euresys는 **EncryptedKey** 선택기 사용을 권장합니다. 프로그래밍 키는 최종 사용자 응용 프로그램의 어느 곳에서도 나타나지 않아도 되므로 보안 수준이 향상됩니다. 암호화된 키만 있으면 최종 사용자는 원래 프로그래밍 키를 검색할 수 없습니다.

OEMSafetyKey 사용

프로그래밍 단계 - 옵션 A

GenICam 브라우저 사용하기:

- 인터페이스 모듈의 GenApi 탭으로 이동하십시오.
- 에 비밀 키를 작성합니다. **ProgramOemSafetyKey**
- 의 값을 복사하고 적절한 곳에 붙 **EncryptedOemSafetyKey**



참고 프로그래밍 키와 암호화된 키 사이에 직접적인 관계가 있습니다. 주어진 프로그래밍 키는 다른 컴퓨터 나 다른 Coaxlink 카드를 사용해도 항상 동일한 암호화 된 키로 연결됩니다. 이렇게 하면 암호화된 키를 한 번 읽고 OEM 안전 키로 보호해야 하는 애플리케이션에서 이 값을 하드 코드할 수 있습니다.

프로그래밍 단계 - 옵션 B

사용자 정의 응용 프로그램 사용:

1. 에 비밀 키를 작성하여 Coaxlink 카드의 OEM 안전 키를 프로그래밍하십시오. **ProgramOemSafetyKey**
2. 를 읽어 암호화 된 키를 다시 읽습니다. **EncryptedOemSafetyKey** 이 값을 적절한 곳에 쓰십시오.

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

확인 단계

OEM 키로 보호해야 하는 응용 프로그램:

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



참고 암호화된 키가 발견되어 공격자가 이를 사용하여 카드를 재프로그래밍하더라도 위의 확인은 실패합니다.