

Camera Configuration Tool 외주 관리 - 개발 요구 사항서 (D5)

[R&D] Camera Graphics Team

Exported on 07/15/2025

Table of Contents

1	요구사항서 수정	5
1.1	배경.....	5
1.2	주요 변경.....	5
2	초기 실행 화면.....	6
3	각 UI 컴포넌트 별 요구사항	9
3.1	I2C Combo Box	9
3.2	External Devices.....	9
3.2.1	(External Devices) 초기화면	9
3.2.2	(External Devices) Path 설정	10
3.2.3	(External Devices) Component 설정	10
3.2.4	(External Devices) Main UI 내 Packet 전시	11
3.3	MIPI.....	12
3.4	ISP.....	13
3.5	Camera Mux.....	14
3.6	SVDW	16
3.7	VWDMA/VIN.....	18
3.8	CIED	19
3.9	MDW	20
4	Read 수행.....	22
4.1	Read 수행 시 main core의 항목과 sub-core 항목이 충돌하는 경우	23
4.1.1	1. main core DTB 로드.....	23
4.1.2	2. sub core DTB 로드	24
4.1.3	3. 충돌 확인.....	24
4.1.4	4. 충돌 해결.....	26
5	Camera Resource Configuration 수행.....	28
5.1	Configuration 수행.....	30
6	Write 수행.....	33

7	Device Tree Parsing 하는 방법	34
7.1	Driver 별로 Parsing 하여야 하는 속성	34
7.2	CAMSENSOR driver	35
7.3	SER(Serializer)/DES(Deserializer) driver	37
7.4	MIPI-CSI/ISP driver	42
7.5	VideoInput driver	49
7.6	CIED driver	51
7.7	MDW driver	55

본 문서는 TCC807x (Dolphin5) Camera Subsystem의 Configuration Tool을 개발 요구 사항서이다.

1 요구사항서 수정

1.1 배경

- 기존 요구사항서 및 릴리즈 형상은 실제 디바이스 트리 설정 시나리오와 맞지 않음
- 또한 main/sub 구분으로 인해 중복되는 UI컴포넌트가 많아 UI가 직관적이지 않고 사용하기 불편함

1.2 주요 변경

- Device Tree 뷰 (main, sub, unified) 추가
- External Devices 항목 추가
- Port 중심의 UI 요구사항 추가
 - Path 활성화 여부 파악 목적
 - Endpoint 중심의 설정 제공 목적
- Camera subsystem 내 각 컴포넌트 별 UI 요구사항 추가

2 초기 실행 화면

TCC807x Camera subsystem과 관련된 모든 Component를 쉽게 설정할 수 있도록 도식화하여 사용자가 볼 수 있도록 하여야 한다.

아래 그림은 Camera Configuration Tool을 실행하였을 때 보여지는 첫 번째 Screen 이다.

아래 그림 중 **ISP** 는 최대 4개를 Plug-in 할 수 있으며, Camera 채널에 따라 Plug-in 될 수 있는 위치를 나타낸다.

그리고, **MIPI**는 2개가 있고 최대 8개를 연결할 수 있으며 현재 MIPI0 4 Channel 카메라를 연결하고 있어서 남은 MIPI1은 아래 연결하지 않았다.

추가 기능 버튼으로 Read, Save 버튼이 제공되어야 하며 그 기능은 아래와 같다.

버튼	기능
Read	Device Tree Blob 파일 내 Camera Resource 설정을 파싱하여 보여주는 기능
Save	설정한 Camera Subsystem Resource를 dts 파일로 저장하는 기능
Clear	UI 초기화: 로드한 DTB 내용 Unload

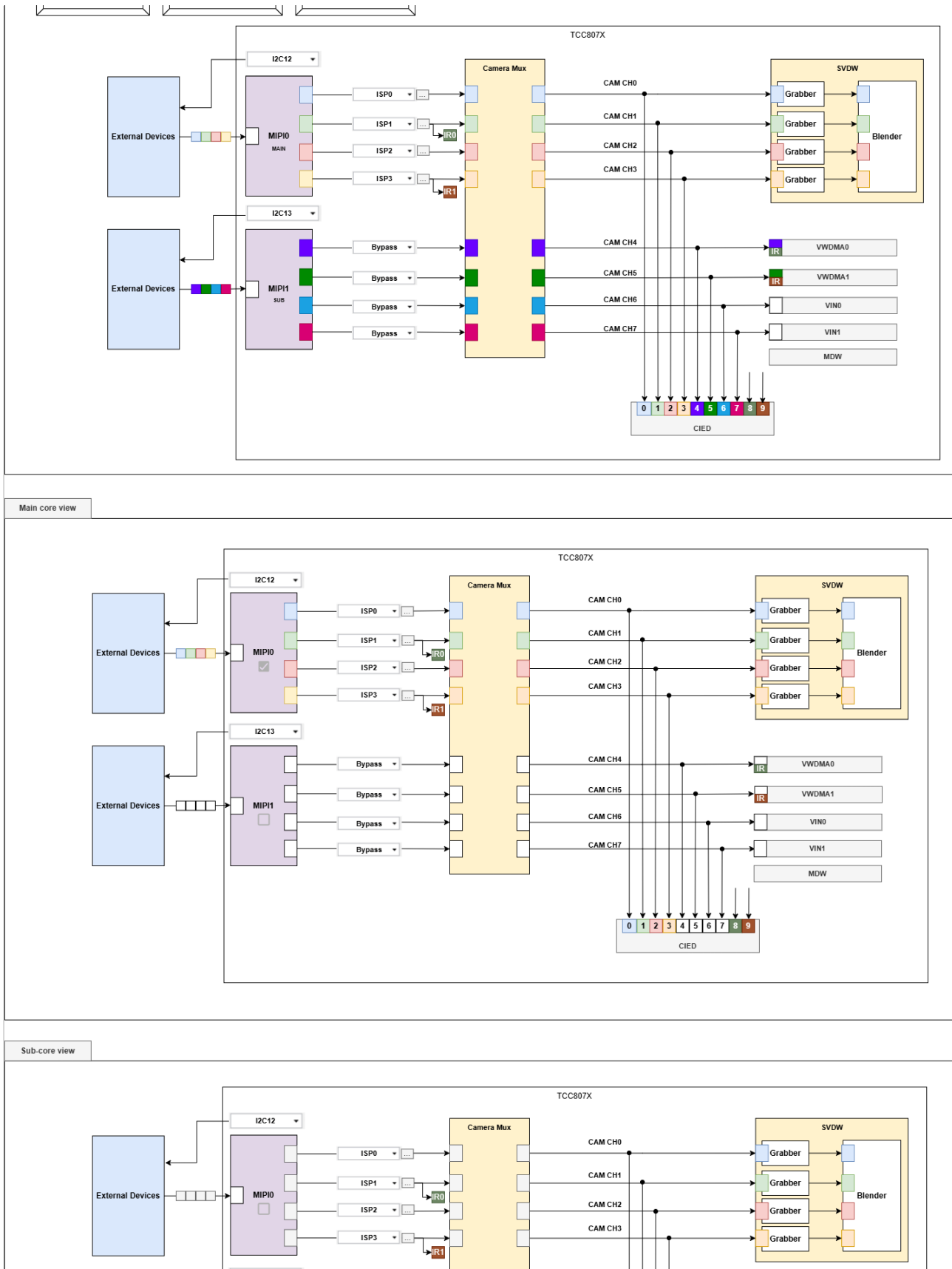
DTB View

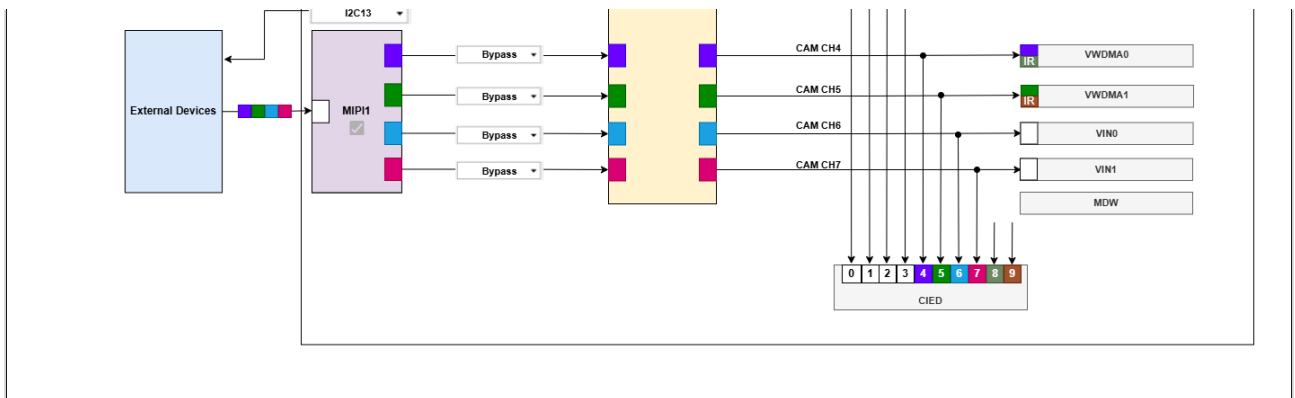
View	기능
Unified	메인 코어와 서브코어 설정을 한번에 보여준다.
Main core	메인 코어의 DTB 구성을 보여주며, 사용자의 설정은 메인 코어 DT 를 설정한다.
Sub-core	서브 코어의 DTB 구성을 보여주며, 사용자의 설정은 서브 코어 DT 를 설정한다.

아래 Screen은 이해를 돕기 위한 예시이다.

Note : 색상이나 폰트의 사이즈 등 기능에 지장이 없는 요소들은 변경될 수 있으나 기본적인 기능은 동일하게 제공되어야 한다.







- Unified 뷰에서는 MIPI에 할당된 코어를 레이블 형태로 보여주며, main, sub-core 뷰 각각에서는 체크박스를 통해 사용하고자 하는 MIPI 를 선택할 수 있다.
- Unified 뷰에서 main 혹은 sub-core를 보여줄 수 없는 상태일 경우, 빈 아이템으로 전시한다. (예: 리소스 충돌 오류)

3 각 UI 컴포넌트 별 요구사항

3.1 I2C Combo Box

- I2C 콤보박스는 Device Tree를 읽어 사용 가능한 I2C 채널을 아이템으로 갖는다.
- 사용자는 콤보박스를 클릭하여 External Devices (Sensor, External ISP, SerDes)를 위해 사용하는 I2C 채널 정보를 설정할 수 있다.

3.2 External Devices

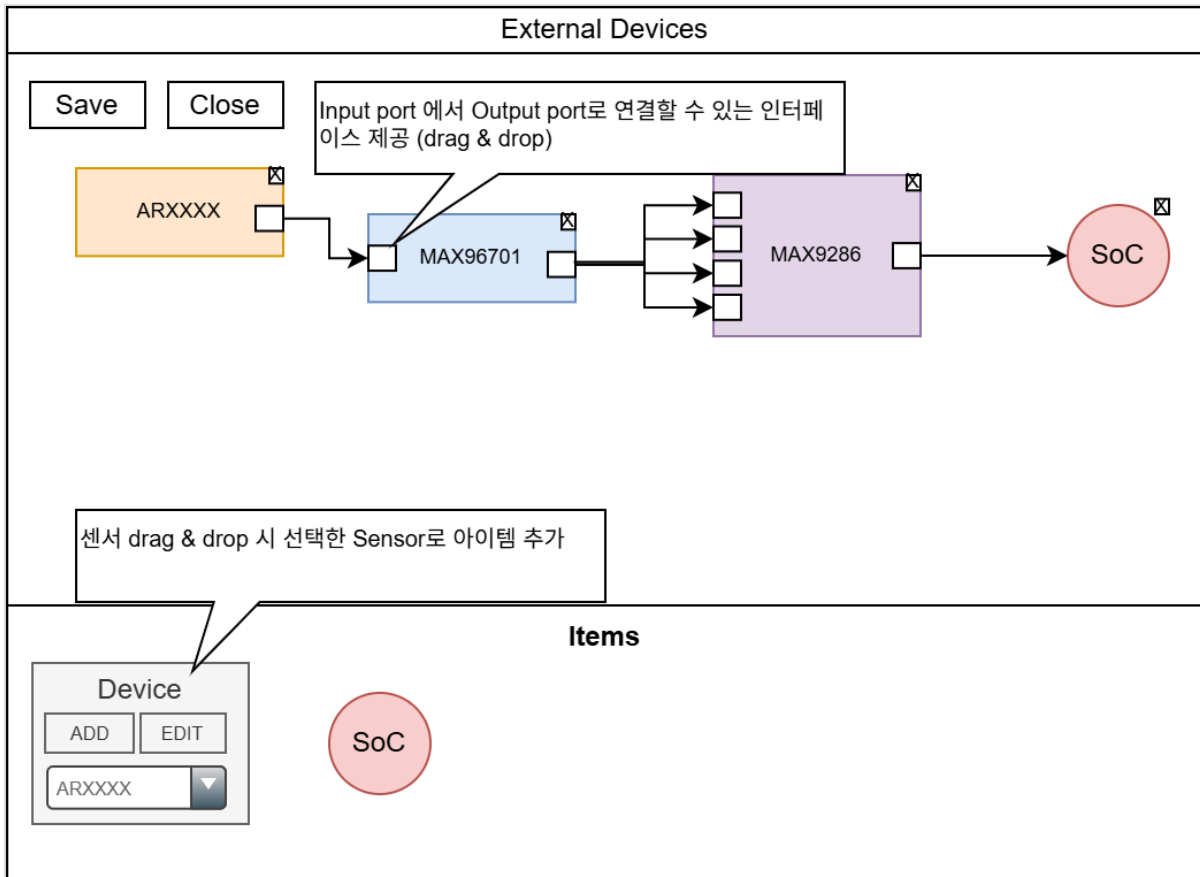
- Telechips AP 기준 외부에 해당하는 디바이스를 External Devices 로 표시하며, External Devices를 클릭하면 외부 IP들을 설정할 수 있는 인터페이스 **\[External Devices\] 초기화면**이 전시된다.

3.2.1 (External Devices) 초기화면

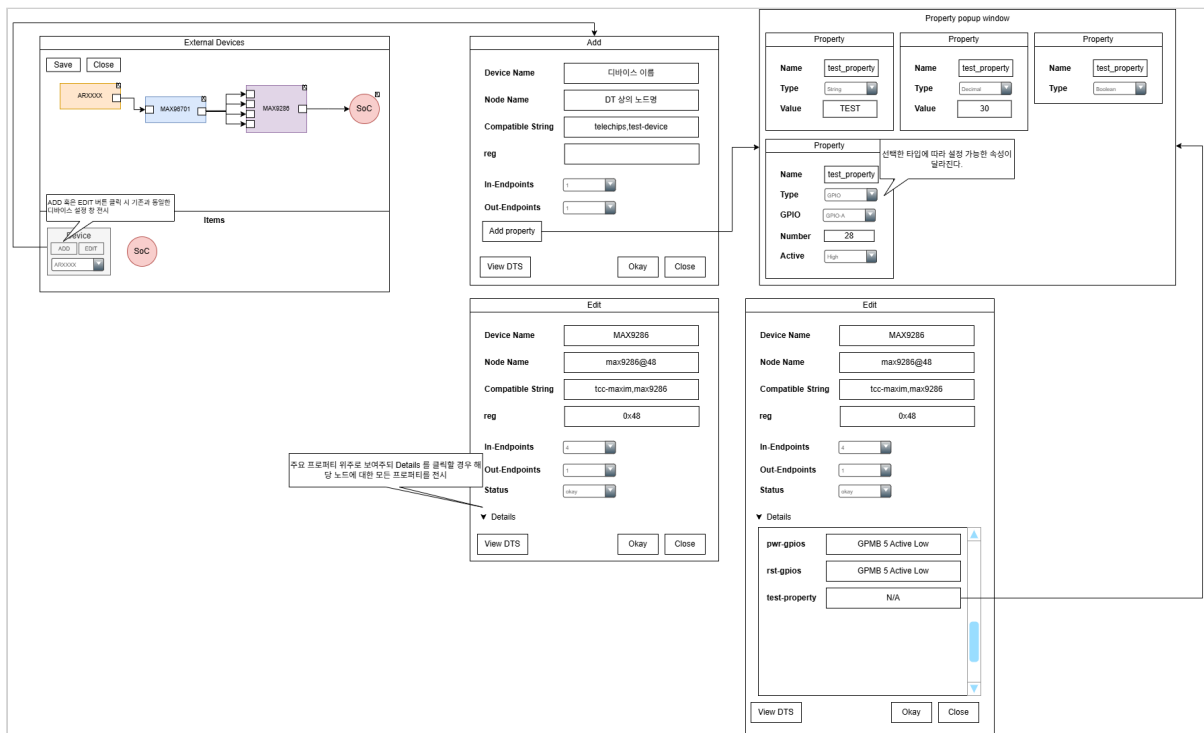
The screenshot shows a window titled "External Devices". At the top left, there are "Save" and "Close" buttons. The main content area is divided into two horizontal sections. The top section is labeled "Items" and contains a "Device" sub-section with "ADD" and "EDIT" buttons, a dropdown menu showing "ARXXXX", and a red circle labeled "SoC".

- Generic Device의 추가 여부 결정 필요

3.2.2 (External Devices) Path 설정

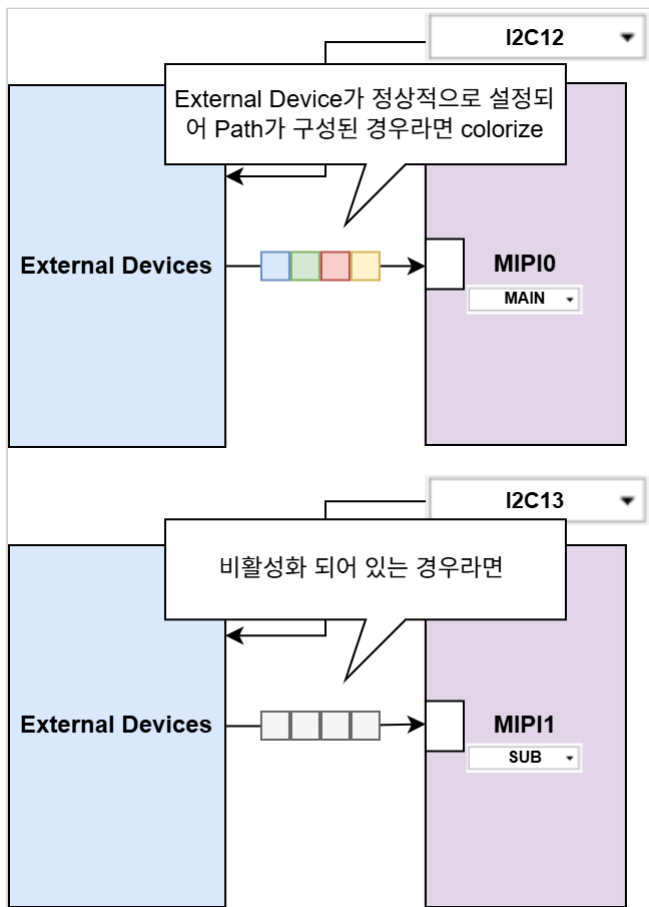


3.2.3 (External Devices) Component 설정



- 사용자가 Details를 클릭 시 주요 프로퍼티 외에 다른 속성도 함께 보여줄 수 있어야 한다. (예: pwr-gpios, rst-gpios)

3.2.4 (External Devices) Main UI 내 Packet 전시

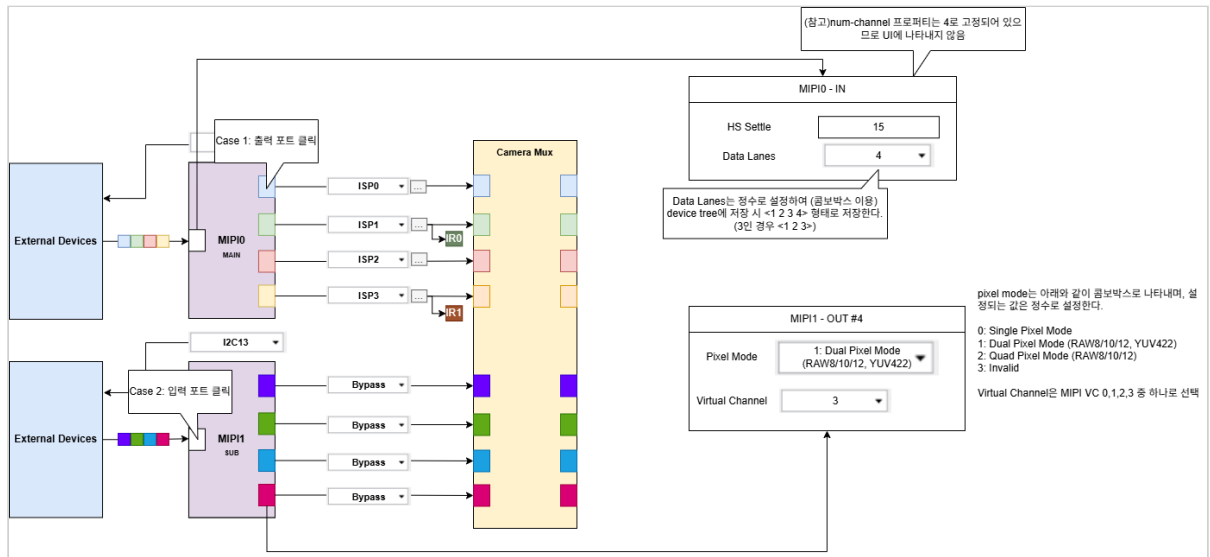


3.3 MIPI

- D5 기준 MIPI는 MIPI0, MIPI1로 구성되며 각각 main core/sub-core 를 선택하여 할당할 수 있다. (특정 코어가 모두 사용할 수도 있음)
- 사용자 설정에 따라 Virtual Channel을 조정하기 위해 MIPI output endpoint 4개 중 원하는 개수만큼 활성화하여 사용할 수 있으며 이는 endpoint 와 직접적으로 연관된다. 활성화 되지 않는 경우, 해당 data lane은 비활성화 한다.

확인 필요: @한동열 (DongYeol Han)

- MIPI 설정을 위해 각 포트를 선택 시 아래와 같이 설정 창을 전시한다.

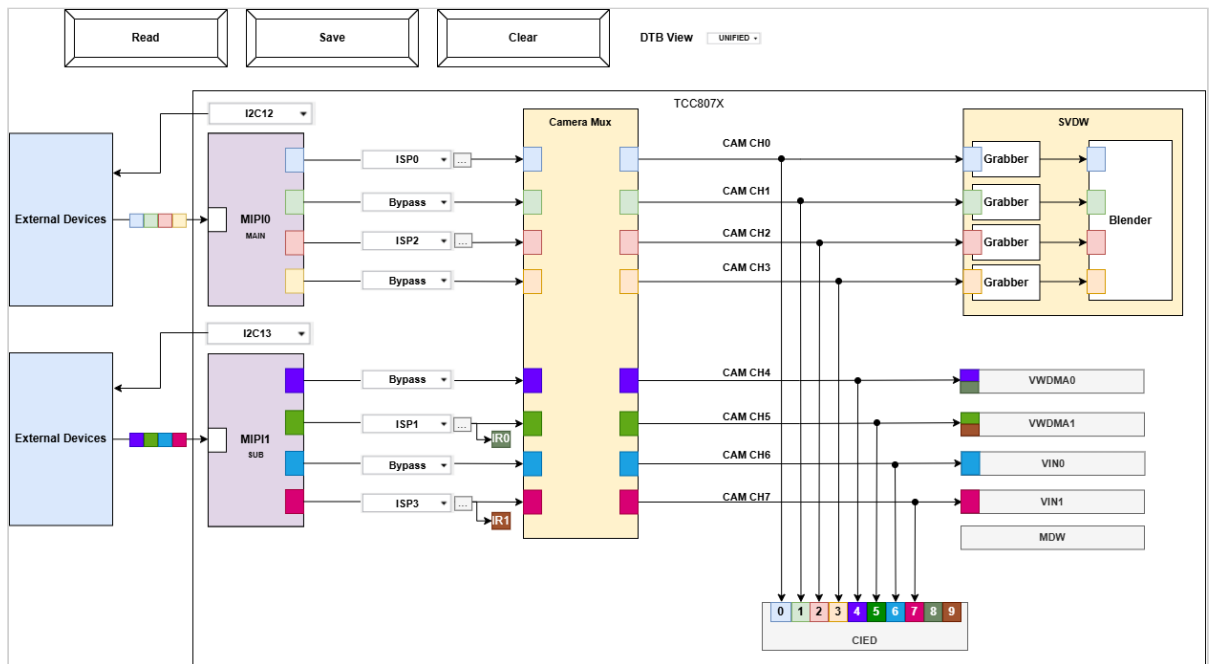


3.4 ISP

- MIPI CSI 출력 채널에서 ISP가 사용되는지 설정하기 위해 사용한다. 이 때, MIPI0와 MIPI1은 상호 배타적이어야 한다.

MIPI0	MIPI1	사용 가능 ISP H/W
MIPI0-CH0	MIPI1-CH0	ISP0
MIPI0-CH1	MIPI1-CH1	ISP1
MIPI0-CH2	MIPI1-CH2	ISP2
MIPI0-CH3	MIPI1-CH3	ISP3

- 아래는 이해를 돕기 위한 설정 예시이다.



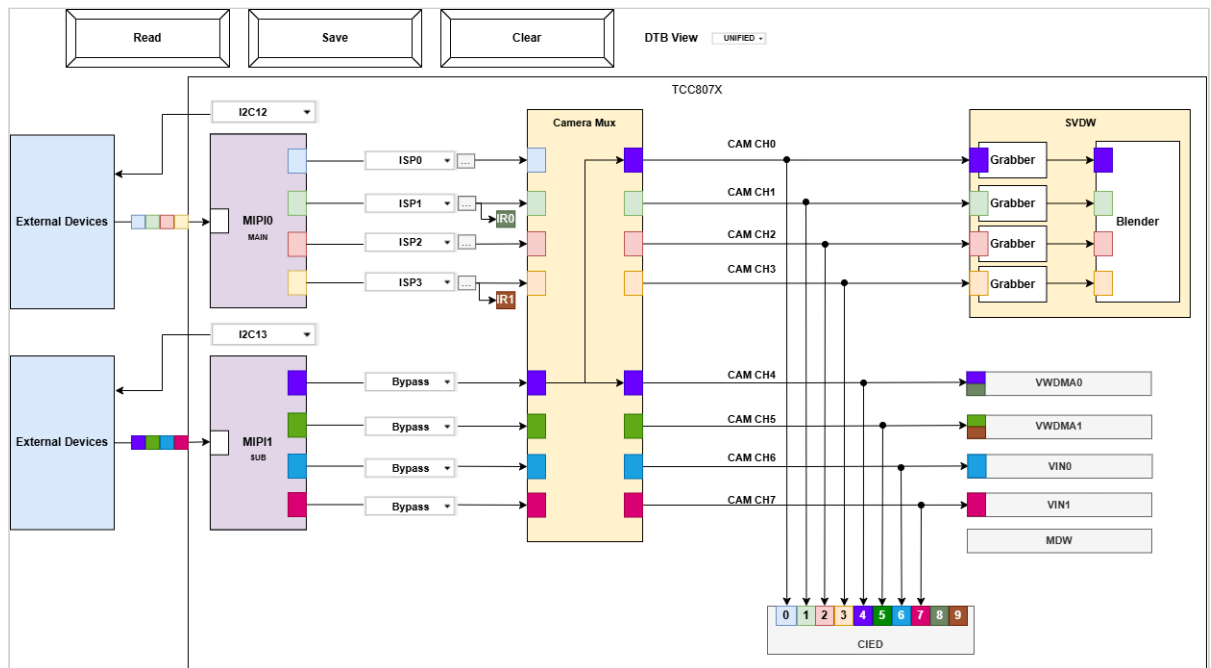
- ISP 설정 박스(...)를 클릭하면 아래와 같이 CFA, MEMORY Share 속성을 설정할 수 있는 설정창이 전시된다.

ISP #0	
CFA	0: RGGB
Shared Memory	Yes

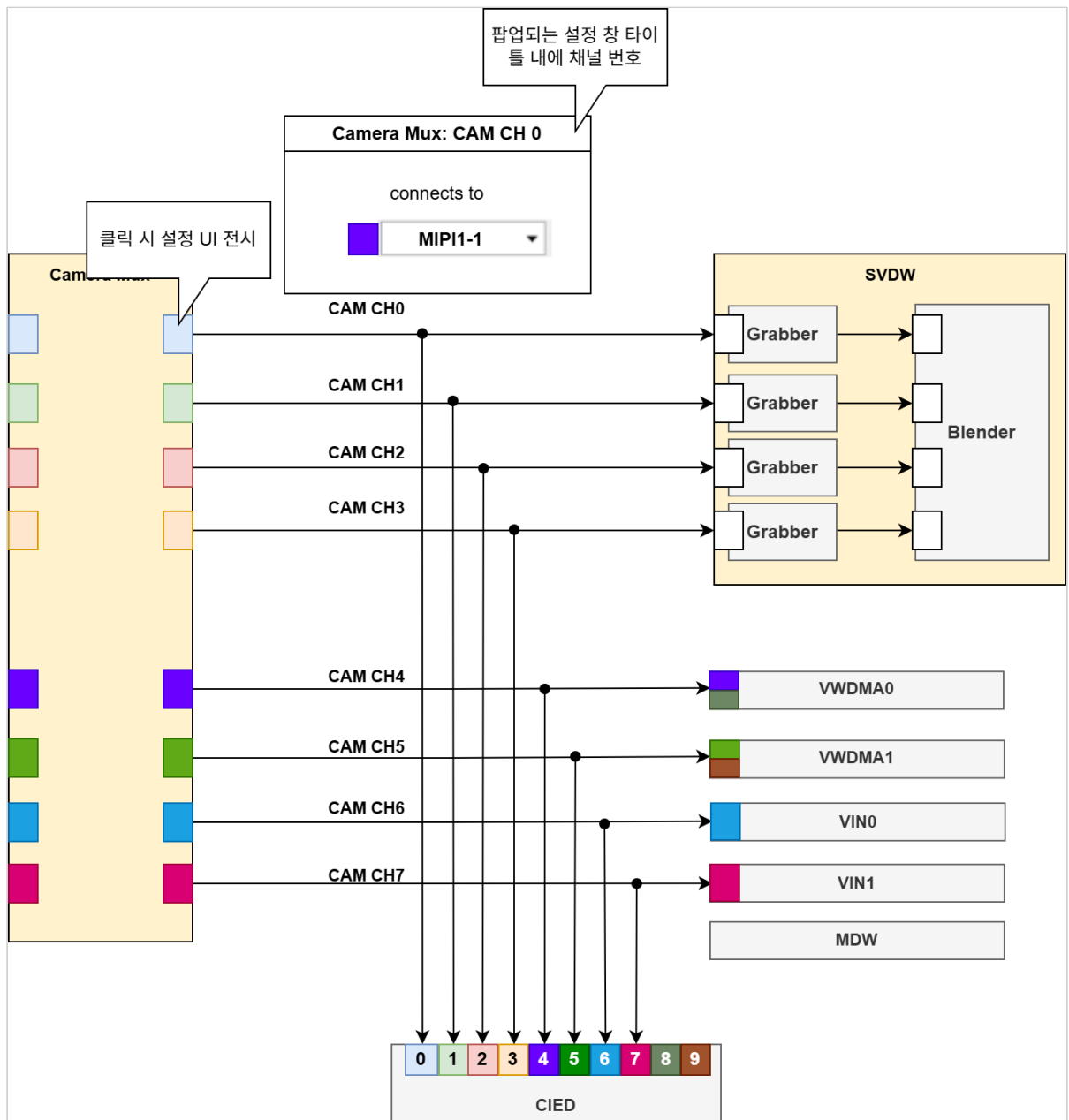
CFA 값:
 0: RGGB
 1: Not used
 2: Not used
 3: IRGBIR
 -> 3은 ISP1과 3만 가능

3.5 Camera Mux

- Camera Mux는 MUX 기준 입력 채널을 Muxing 해주는 역할을 한다.
- 그림과 같이 입출력 상태를 파악하기 쉽도록 포트별 색깔을 이용하여 입출력 설정을 쉽게 알 수 있도록 한다.
 - 다이어그램의 색상은 예시이며, 알기 쉬운 색상을 사용한다. (예: 빨강, 주황, 노랑, 초록, 청록, 파랑, 남색, 보라색)
- Camera Mux에 대한 설정은 DMA component의 cam-ch 속성을 참고한다. 예를 들어, SVDW#0의 cam-ch 이 <4>로 설정되어 있다면 아래와 같이 Mux가 설정되어 있는 것이다.
(아래 다이어그램에서 Camera Mux 내 연결선은 단순하게 연결 상태를 설명하기 위한 것으로 실제 형상에서는 그리지 않도록 한다.)



- Camera Mux 설정은 **중복 설정** 가능하다. 위의 다이어그램과 같이 특정 입력을 여러 개의 채널로 multiplexing 할 수 있다.
- Camera Mux 설정은 아래와 같이 output 포트를 설정했을 때 어떤 입력으로 mux를 설정할 것인지 UI를 제공한다.
- 설정한 cam-mux (output) 값은 대응되는 컴포넌트 내 cam-ch 속성으로 저장된다.
예를 들어, 아래와 같이 Camera Mux의 첫번째 CAM CH0을 MIP11-1에 대응하는 경우, SVDW #1에 해당하는 디바이스 노드 내에 cam-ch 속성을 4로 설정한다.

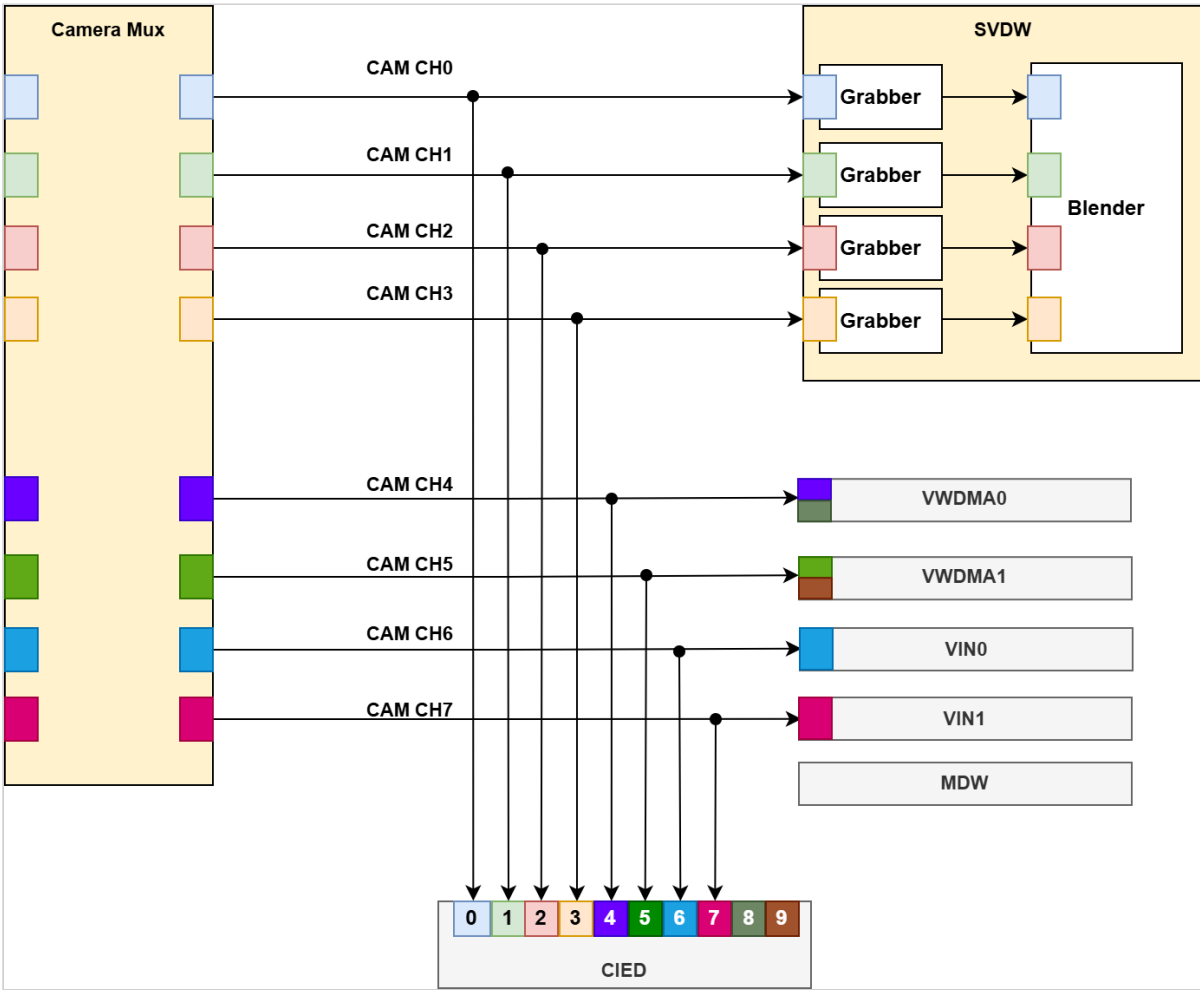


3.6 SVDW

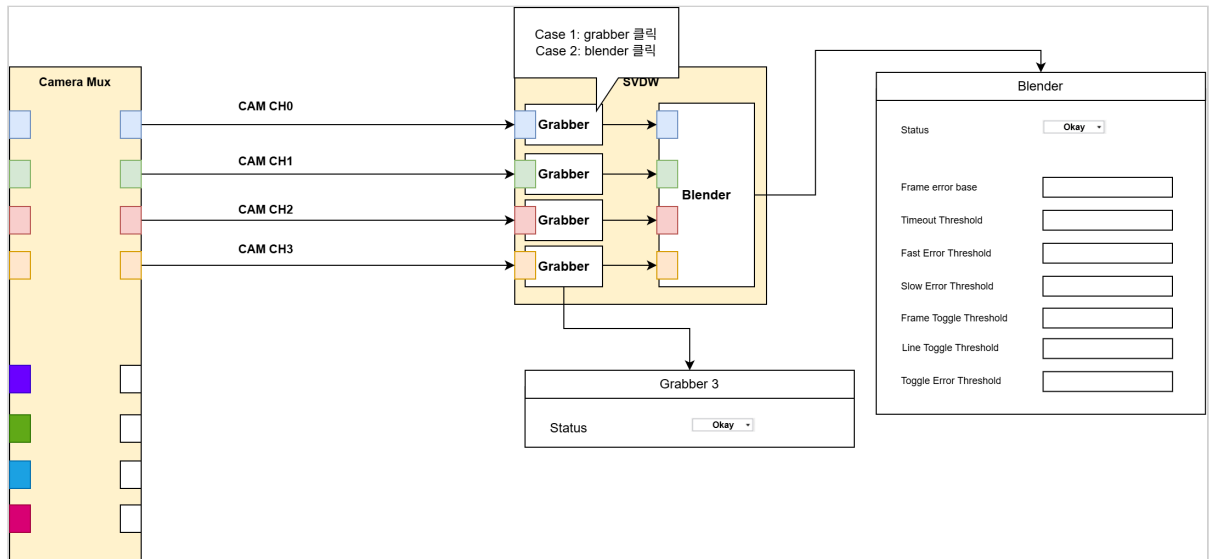
- D5 기준 SVDW는 디바이스 트리 내에 "telechips,tcc-svdw-grabber", "telechips,tcc-svdw-blender" 의 compatible string을 갖는다.
- SDK의 호환성을 위해 아래의 compatible 경우도 함께 지원할 수 있어야 한다.
(Legacy SDK의 compatible string과 Recent SDK의 compatible string이 함께 사용되는 경우는 없음)

Legacy SDK	Recent SDK
telechips,dewarp	telechips,tcc-svdw-grabber
telechips,tcc-svdw	telechips,tcc-svdw-blender

- Blender 활성화 상태에 따라 포트의 색깔을 활성화하거나 비활성화 한다. 아래는 Blender가 활성화됐을 경우 Blender의 입력포트에 동일한 색상으로 설정된 걸 알 수 있다.



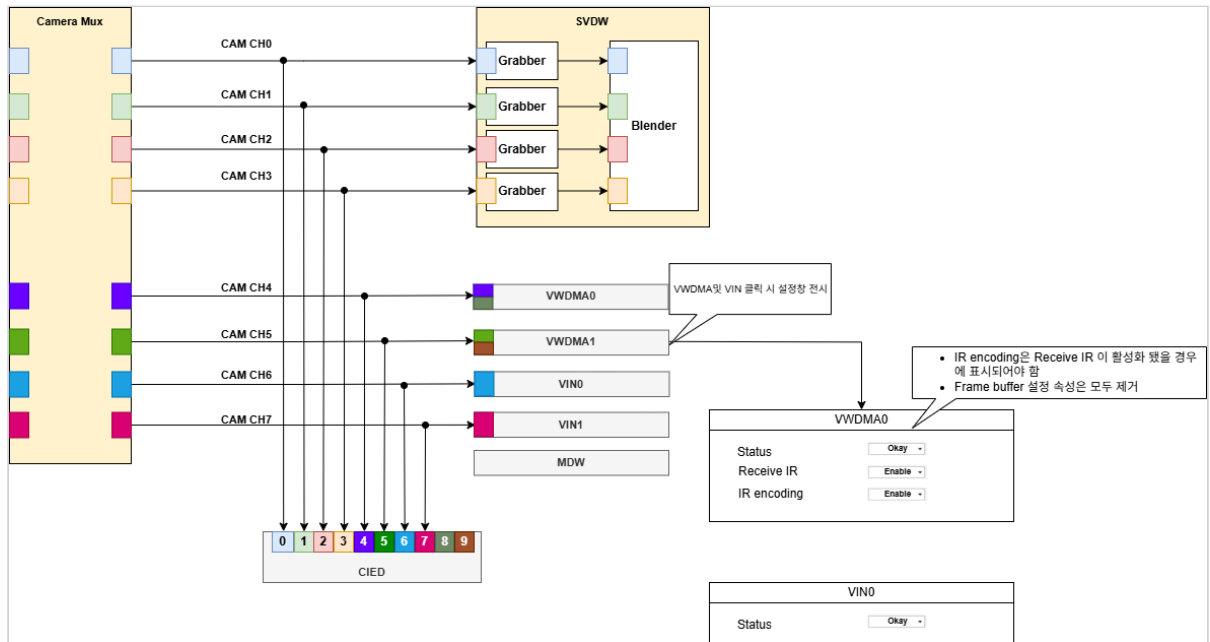
- SVDW를 클릭하면 아래와 같이 설정 창이 전시된다.



- SVDW blender에 연결되는 4개 포트가 모두 정상적으로 설정되어 있지 않다면 Status 를 설정할 수 없어야 한다. 클릭 시 포인터 위치에 Hint 나 메시지를 띄워 사용자가 설정을 시도할 경우 4개 입력이 들어오지 않고 있음을 알릴 수 있어야 한다.

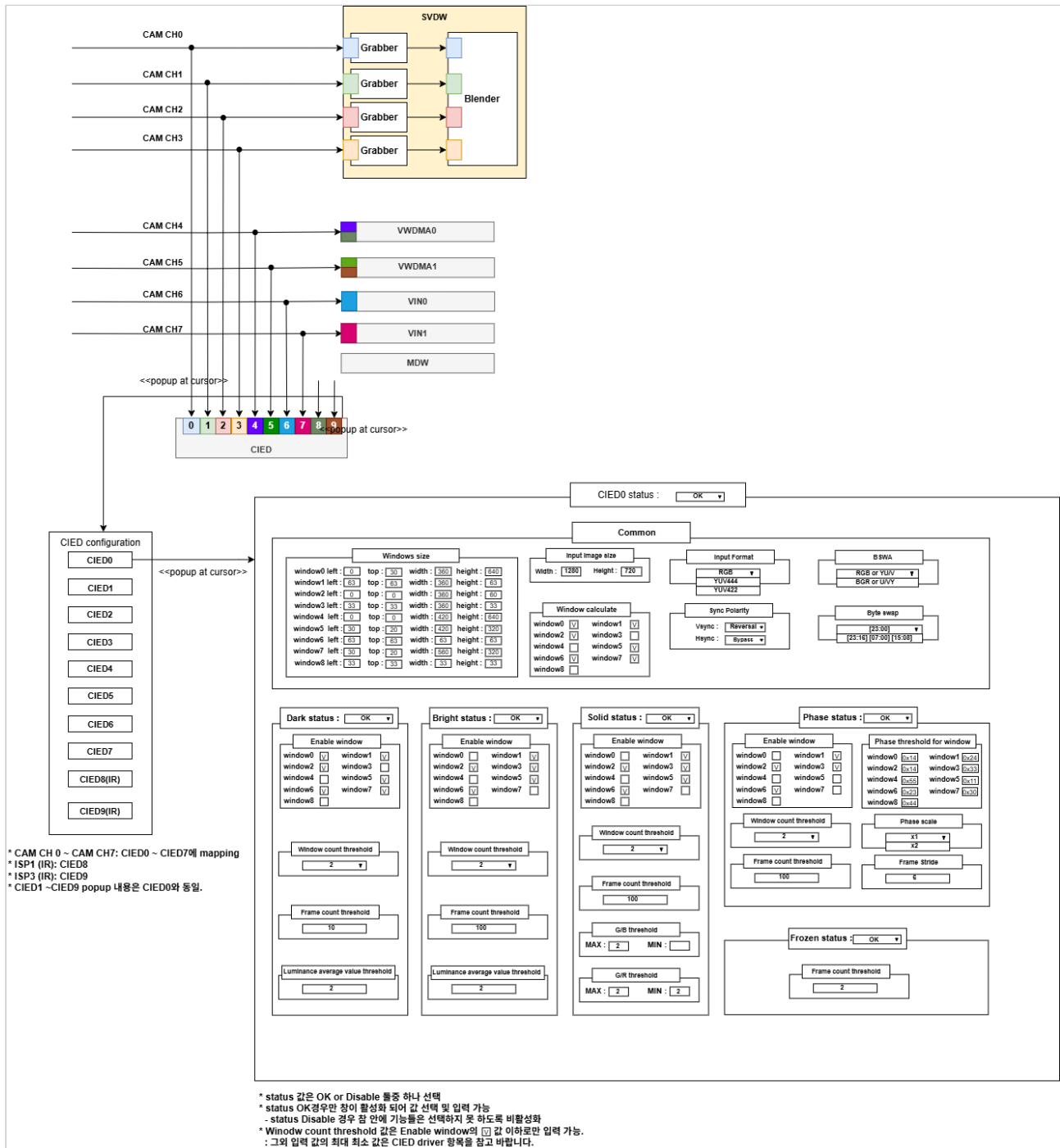
3.7 VWDMA/VIN

- VWDMA 및 VIN은 SVDW와 마찬가지로 메모리에 DMA하는 IP이다.
- VWDMA와 VIN 각각 "telechips,vwdma", "telechips,video-input"을 compatible string으로 갖는다.
- VWDMA의 경우 IR 에 대한 설정 프로퍼티가 추가되며, 각각에 대해 Enable/Disable 설정을 할 수 있다. Receive IR 속성은 "ir-enable" DT 속성에, IR encoding은 "ir-encoding" DT 속성에 매핑된다.
- VWDMA의 2가지 색상 중 하위 색상은 IR데이터를 나타내며, VWDMA0,VWDMA1은 각각 IR0, IR1에 매핑된다. IR 데이터는 ISP1, ISP3 사용 여부에 달라지고 CAM_MUX의 설정과는 무관하다.

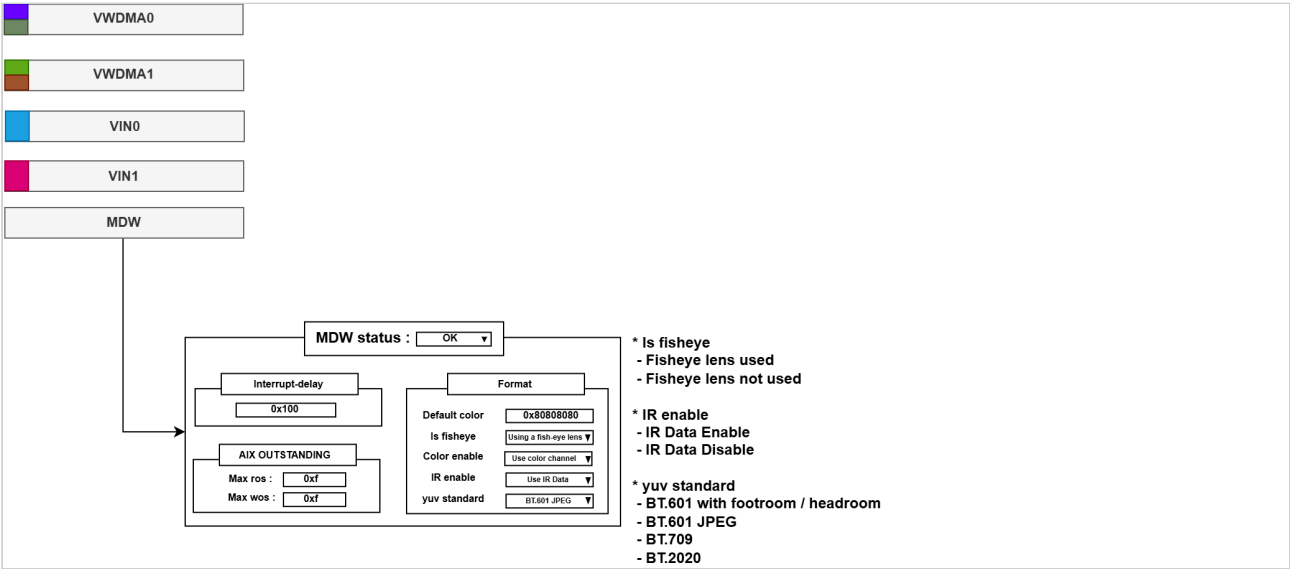


3.8 CIED

- CIED는 박스를 클릭하거나 각 포트를 선택할 경우 각각에 대한 설정창이 팝업된다.
- CIED8은 ISP1 경로에, CIED9는 ISP3 경로에 매핑되어야 하므로 사용자의 설정에 따라 포트의 색상이 변경된다.



3.9 MDW



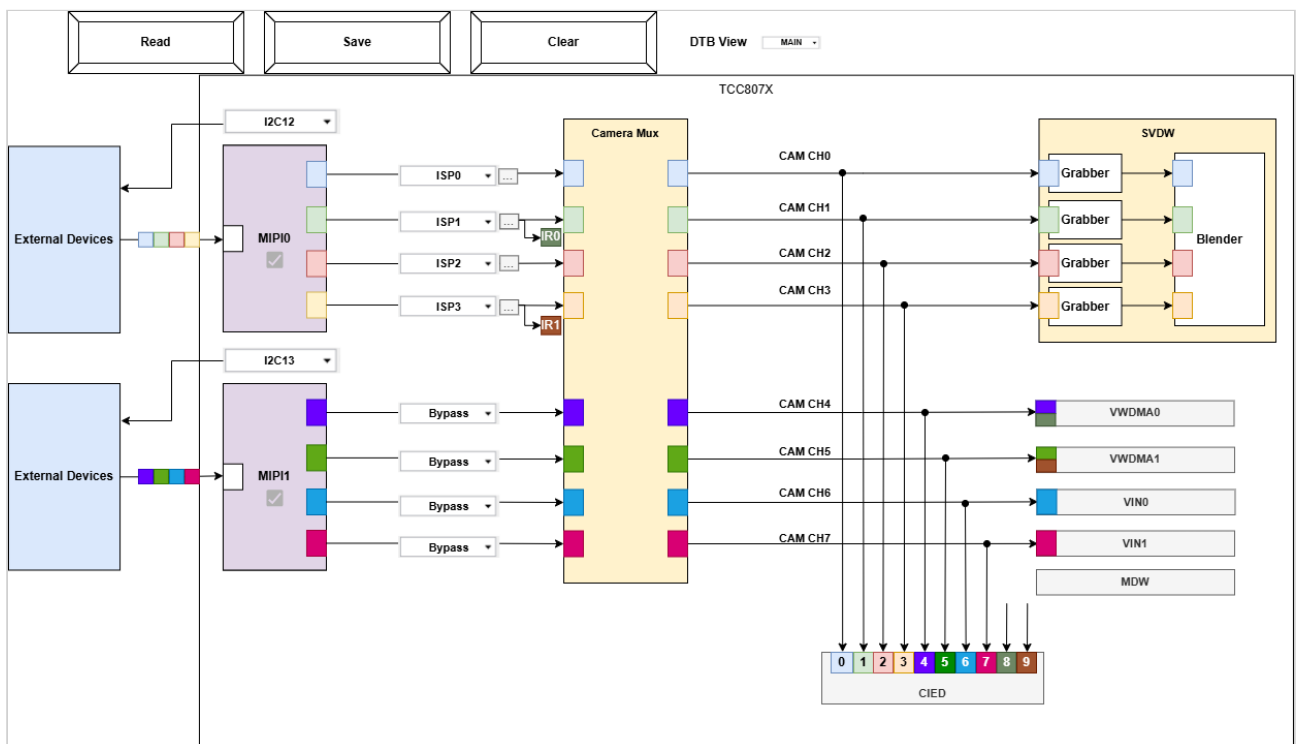
4 Read 수행

사용자가 아래의 Read 버튼을 누르면 Read 동작이 수행되어야 한다.

Tool 은 TCC807x camera subsystem의 Device Tree에 설정된 설정값을 Read 한 후, 각 설정된 Component들과 Pulg-in 된 Path를 도식화하여 사용자에게 보여주어야 한다.

아래 Screen은 이해를 돕기 위한 예시이다.

Note : 색상이나 폰트의 사이즈 등 기능에 지장이 없는 요소들은 변경될 수 있으나 기본적인 기능은 동일하게 제공되어야 한다.



Read 동작 시나리오는 아래와 같다.

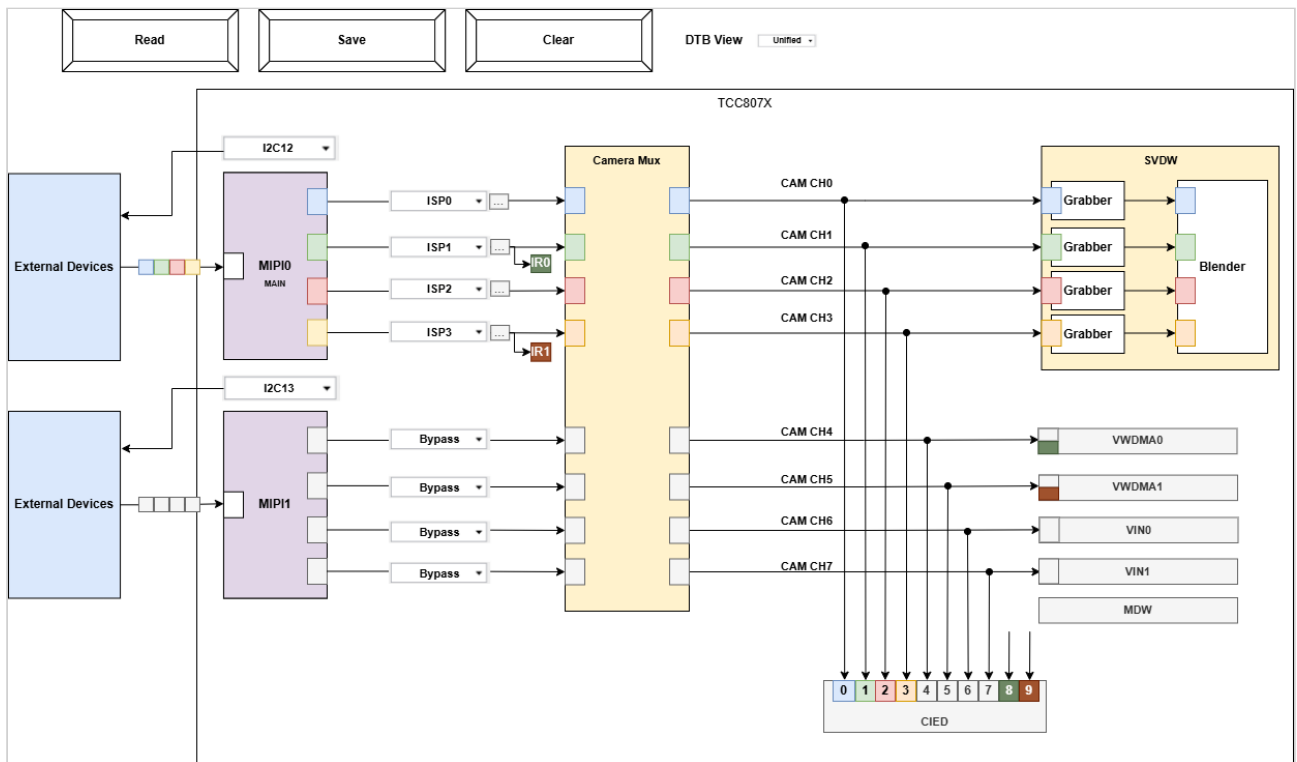
단계	동작
dtb 혹은 dts file에서 정보 읽어오기	<ul style="list-style-type: none"> • Device Tree의 dtb, dts 파일에서 Camera Resource 정보를 읽어 올 때, Main cluster의 dtb와 Sub cluster의 dtb 파일을 사용자가 지정할 수 있는 기능을 제공하여야 한다. • Main cluster dtb, dts 파일에서 읽어온 Camera Resource 정보에는 그 앞에 (M)으로 표시한다. • Sub cluster dtb, dts 파일에서 읽어온 Camera Resource 정보에는 그 앞에 (S)으로 표시한다. • Camera Resource의 경우, 최대 Resource가 정해져 있기 때문에 이를 넘어갈 시에는 에러를 표시하고 Resource가 충돌이 났다고 사용자에게 알립니다. <ul style="list-style-type: none"> • MIPI - 0, 1 • - 0, 1, 2, 3 • SVDW - 0, 1, 2, 3 • VWDMA - 0, 1 • VIN - 0, 1 • MDW - 0 • 동일한 Cluster 내에서는 동일한 Resource를 Software driver 들이 공유하는 경우 에러를 표시하고 Resource를 공유할 수 없음을 사용자에게 알린다. <p>Note : Device Tree를 Parsing하는 방법은 4. Device Tree Parsing 방법 참고.</p>
결과 보여주기	<ul style="list-style-type: none"> • Device Tree의 dtb, dts 파일에서 Parsing한 결과는 Tool 초기 Screen에 업데이트하여 사용자에게 보여주어야 한다.

4.1 Read 수행 시 main core의 항목과 sub-core 항목이 충돌하는 경우

아래와 같이 main core dtb와 sub-core dtb 항목 내 디바이스 노드의 사용이 충돌하는 경우를 기술한다. (MIPI0, SVDW0/1 충돌)

- Main core: ARXXXX - MAX96701 - MAX9286 - **MIPI0** - ISP bypass - SVDW0/1/2/3 (blender enabled)
- Sub-core: AR0147 - MAX96701 - MAX96712 - **MIPI0** - ISP0/1/2/3 - **SVDW0/1** & VWDMA0/1

4.1.1 1. main core DTB 로드



4.1.2 2. sub core DTB 로드

이 때, 이미 로드한 sub-core 와 중복되는 H/W 리소스 (MIPI0, SVDW0/1) 이 문제가 되므로 에러메시지를 팝업하여 중복되는 리소스를 알린다.

Warning
Resource conflict detected in DTB:
MIPI0
SVDW0
SVDW1

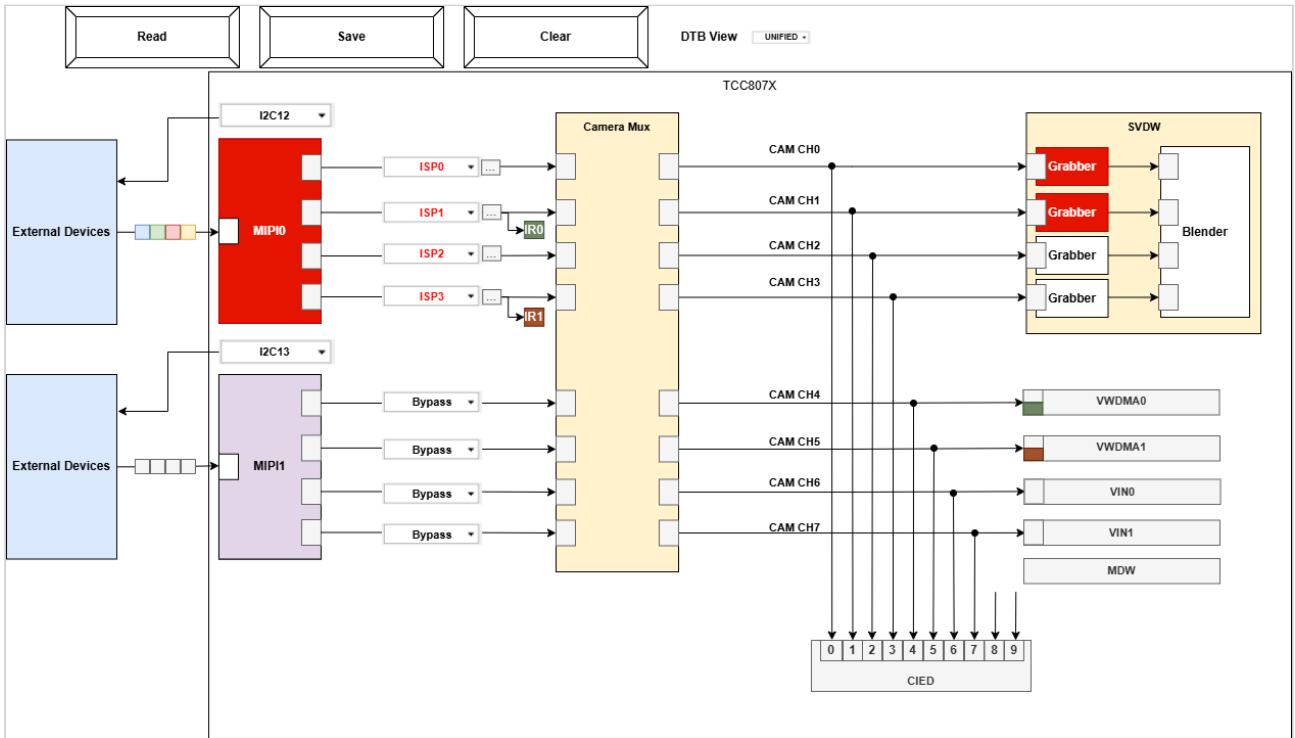
4.1.3 3. 충돌 확인

사용자는 각 path 를 설정하고 정상적으로 충돌이 해결되었는지 UI로써 알 수 있어야 한다. 이를 위해 메인코어와 서브 코어 각각 설정할 수 있는 **DTB View** 를 제공해야 하며, Main, Sub, Unified 뷰를 제공한다. 충돌이 발생한 경우 각각에 대한 뷰는 아래와 같다.

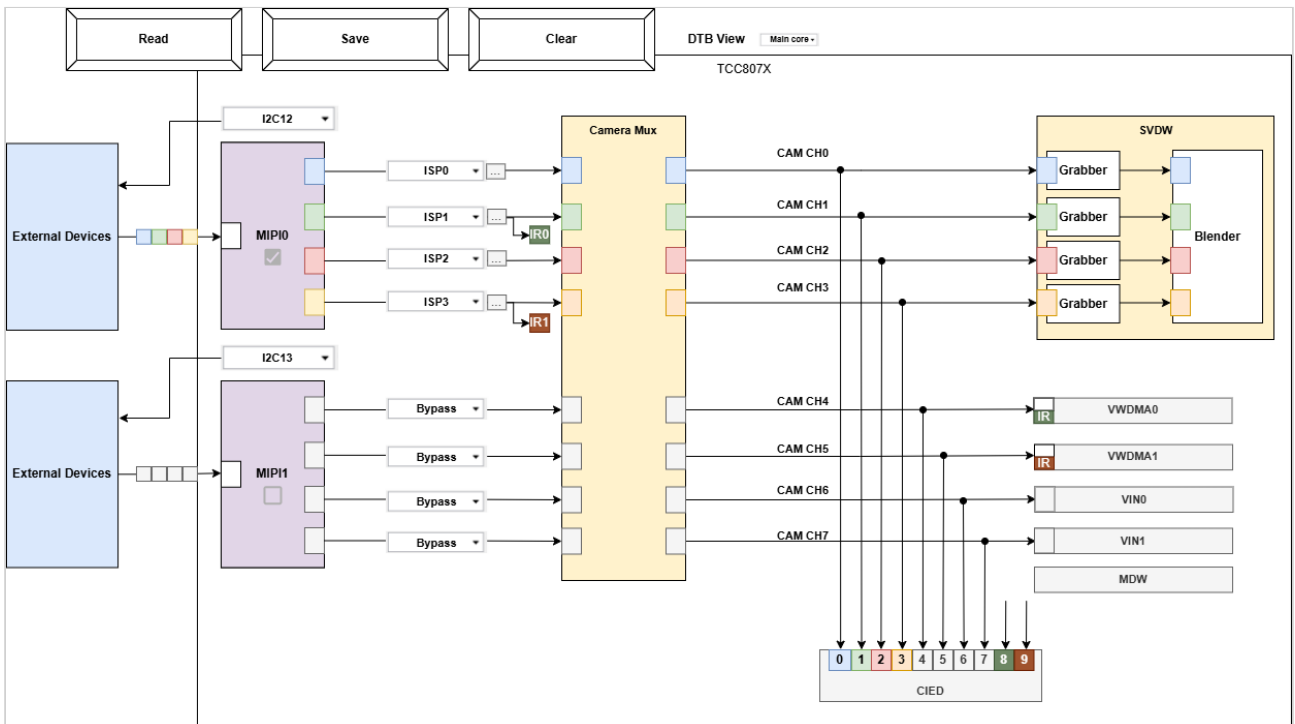
main core, sub-core 뷰 각각에서 사용자는 사용하고자 하는 MIPI를 선택할 수 있으며, Unified view에서는 콤보박스 형태로 main/sub를 선택할 수 있다. 단, 충돌이 발생할 경우 특정 아이템으로 결정될 수 없으므로 빈칸으로 표시된다.

Unified View

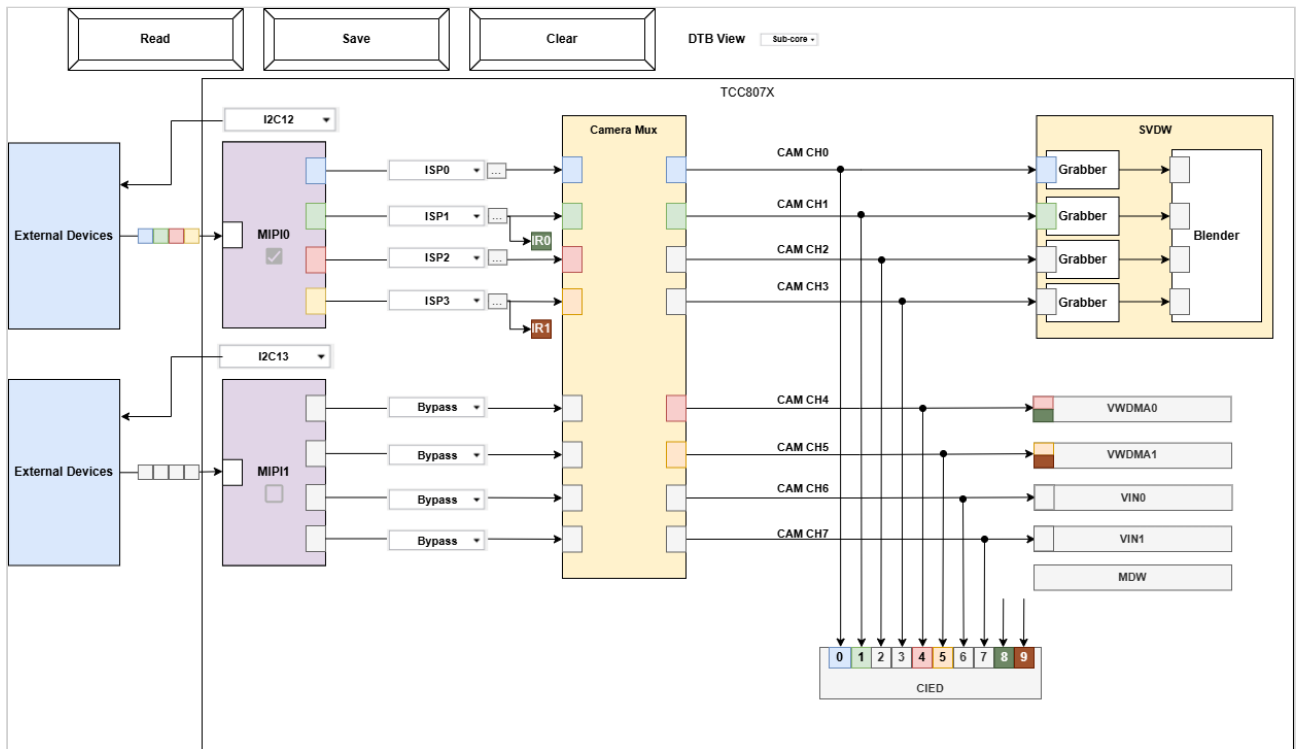
충돌이 발생한 경우, 문제가 발생하는 MIP10 이후로부터의 Path는 유효하지 않으므로 colorize 된 Port는 유효하지 않음을 알 수 있도록 포트 색깔이나 연결선들을 처리한다. MIP10는 충돌로 Main 이나 Sub 로서 레이블 텍스트를 나타낼 수 없으며, 중복 사용되는 컴포넌트는 경고색으로 색상 처리한다.



Main core View



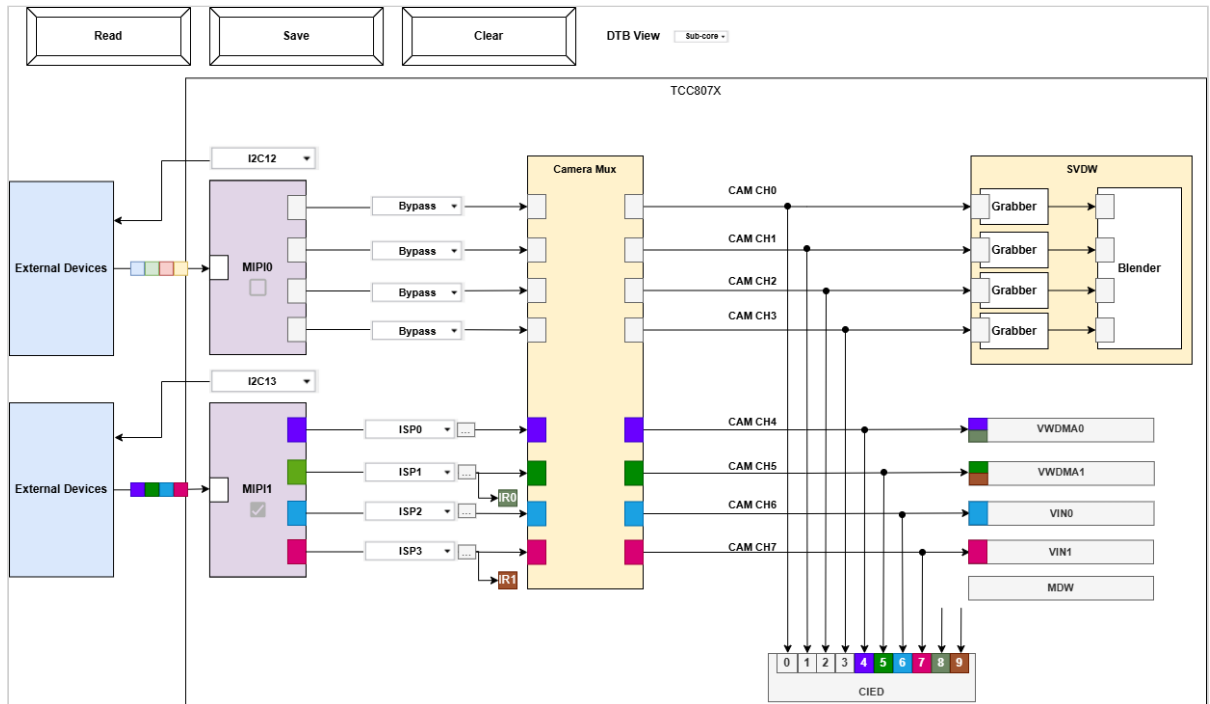
Sub-core View



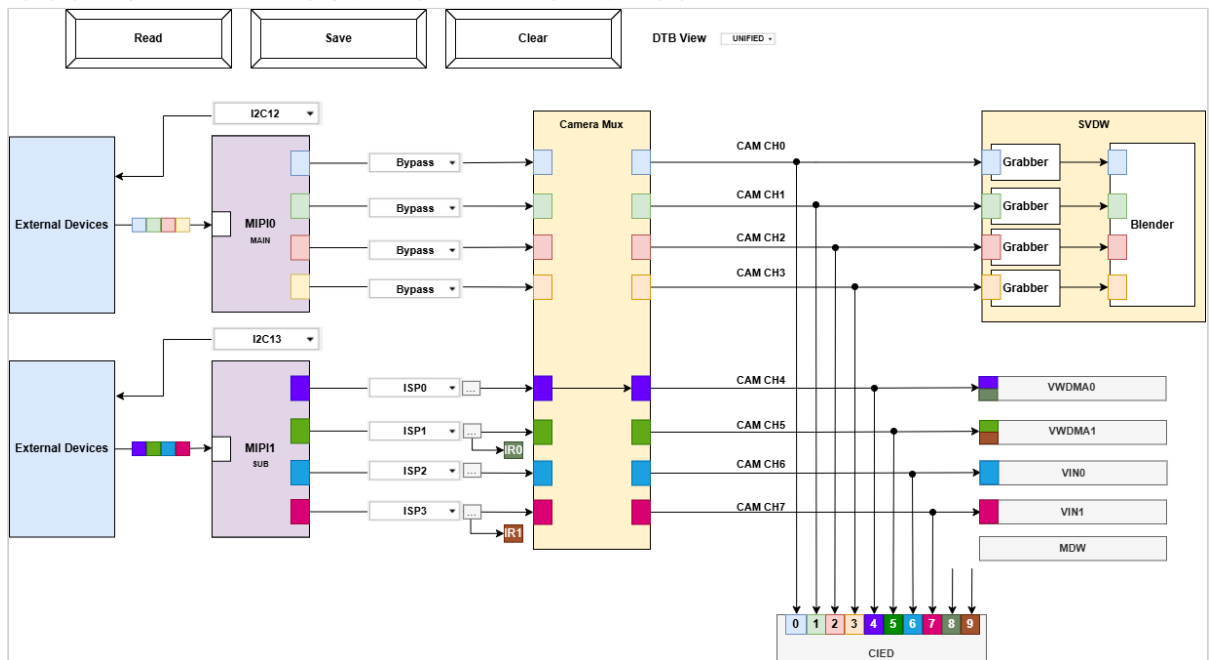
4.1.4 4. 충돌 해결

아래의 순서대로 sub-core 의 MIPI 사용을 아래와 같이 수정

- 기존 MIPI0 응용을 MIPI1으로 변경하기 위해, MIPI1에 연결되는 External Devices 구성
- MIPI1의 체크박스 이용해 sub-core 로 설정하고 MIPI0는 해제
- Camera Mux를 이용해 CAM CH4,5,6,7로 연결되는 Path로 구성



- 아래와 같이 Unified View에서 충돌 해결된 것을 확인 가능해야 함



5 Camera Resource Configuration 수행

Read 실행 후 Visualization 된 UI 인터페이스를 통해 사용자는 원하는 Driver 및 Component 들을 설정/선택하고 연결될 수 있도록 Tool에서 시각화되어 해당 기능을 지원하여야 한다.

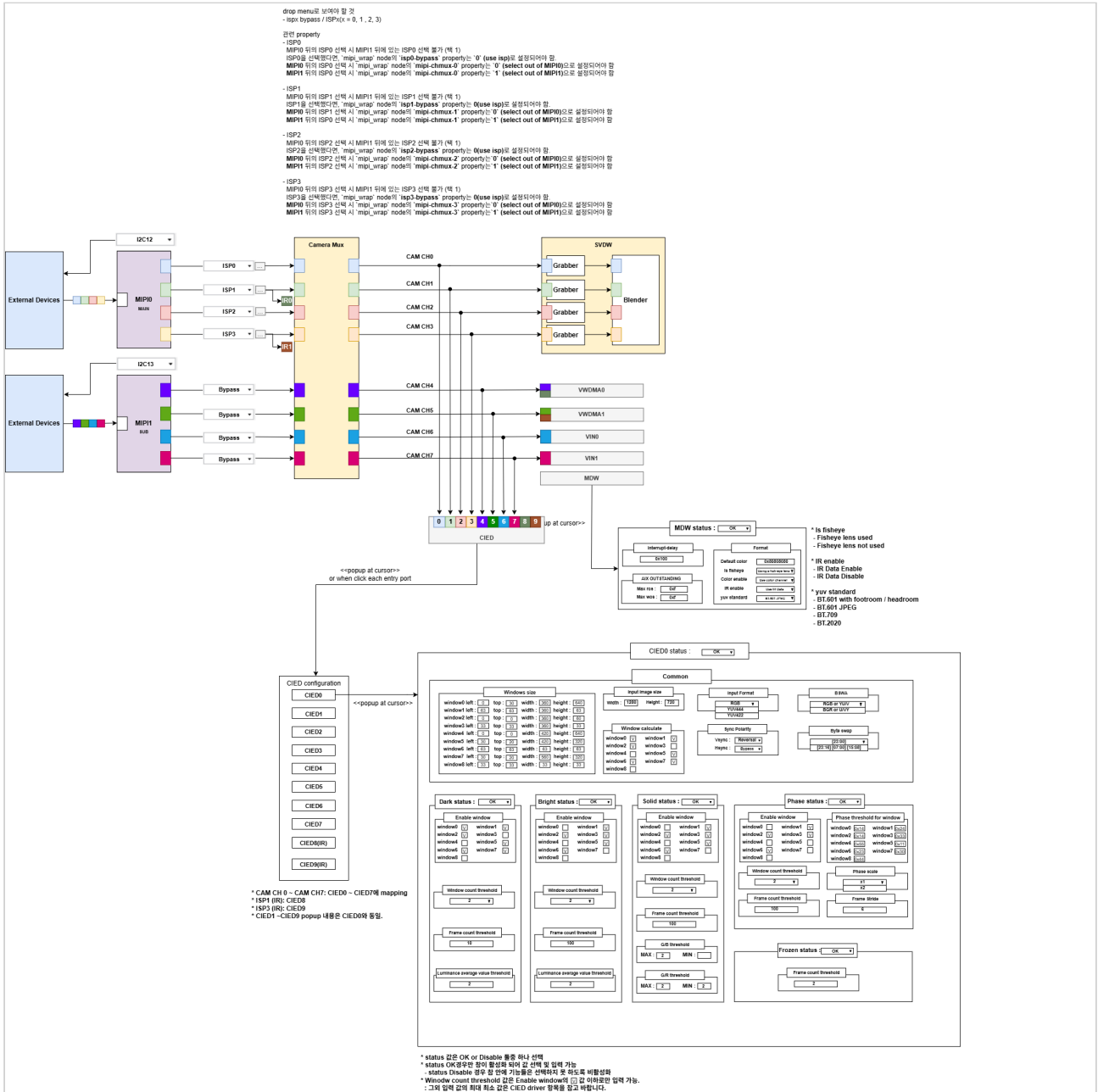
Note : Camera Resource Configuration을 실행하기 위해서는 Read 동작이 우선 실행되어야 한다.

따라서 Read 동작이 실행되지 않은 상태에서 사용자가 Camera Resource Configuration 동작을 수행하려 하는 경우 아래와 같이 동작해야 한다.

- 사용자에게 Camera Resource Configuration을 실행하기 위해서는 Read 동작이 우선 실행되어야 함을 알린다.

아래 Screen은 이해를 돕기 위한 예시이다. (최대 8개의 Camera를 연결한 예시입니다.)

Note : 색상이나 폰트의 사이즈 등 기능에 지장이 없는 요소들은 변경될 수 있으나 기본적인 기능은 동일하게 제공되어야 한다.



설정이 완료된 후 사용자가 **Save** 버튼을 누르면 설정값을 dts 파일에 저장하는 기능을 제공하여야 한다.

기능	동작
Save	<ul style="list-style-type: none">• Main cluster에서 설정한 정보는 Main cluster의 dts 파일에 Sub cluster에서 설정한 정보는 Sub cluster의 dts 파일에 저장한다.• Save 후에는 콤보 박스들이 비활성화 상태로 변경되어야 한다.

5.1 Configuration 수행

사용자가 Read 기능을 수행하여 Device Tree Blob 파일을 로드 후 Configuration 를 실행할 수 있도록 콤보 박스를 비롯한 설정 UI 컴포넌트들이 활성화 되어야 한다.

수행되어야 할 기능은 아래와 같다.

기능	설명
Chip 외부 카메라 관련 Driver를 선택 및 설정	<ul style="list-style-type: none"> • 고객사에서 외부 카메라 Driver를 추가할 수 있어야 한다. • Camera Module (Camera Sensor, Serializer)를 박스로 추가하고 해당 박스를 클릭하여 세부 설정을 할 수 있어야 한다. • Deserializer 박스를 추가하고 해당 박스를 Double Click 하면 세부 설정을 할 수 있어야 한다. • CIF 연결시 Video Decoder를 추가하고 해당 박스를 Double Click 하면 세부설정을 할 수 있어야 한다. • Chip Hardware(CIF or MIPI)에 연결할 수 있도록 설정해야 한다. • 자세한 것은 각 UI 컴포넌트 별 요구사항의 External Devices 항목 참고
MIPI, DMA Component (VWDMA, VIN, SVDW) 연결	<p>칩 외부 카메라 관련 Driver와 MIPI와 연결 한 후 DMA Component에 연결해야 한다.</p> <ul style="list-style-type: none"> • Note : MDW는 memory-to-memory 이므로 MIPI 출력을 입력으로써 사용하지 않는다.
CIED 설정	<p>CIED (#0 ~ #9) 에 해당하는 Camera Mux 출력 채널은 아래와 같다.</p> <ul style="list-style-type: none"> • CIED0: channel 0 • CIED1: channel 1 • CIED2: channel 2 • CIED3: channel 3 • CIED4: channel 4 • CIED5: channel 5 • CIED6: channel 6 • CIED7: channel 7 • CIED8: channel 4 (IR) • CIED9: channel 5 (IR)

기능	설명
ISP CFA 확인	<p>ISP1, 3만 RGBIR (값: 3) 이 가능하다.</p> <pre> &isp1 { status = "okay"; /* * 0: RGGB * 1: NOT used * 2: NOT used * 3: RGBIR (ISP1, 3 only) */ cfa = <3>; } </pre>
ISP Bypass 여부 설정	<p>ISP Bypass dropbox를 통해 MIPI0, MIPI1에서의 ISP 사용 여부에 대한 부분을 설정할 수 있다.</p> <ul style="list-style-type: none"> ISP0/1/2/3 은 MIPI0 또는 MIPI1 한 군데에서만 사용 가능하다. 8개 입력 모두 Bypass로 설정될 수 있다.
Camera Mux 설정	<p>Camera Mux 설정 UI를 통해 Camera Mux 전후 입출력에 대한 매핑을 설정할 수 있다.</p>

Chip Hardware에서 지원하는 각 Component 의 숫자

	MIPI	ISP	SVDW	VWDMA	VIN	MDW	CIED
최대 지원 개수	2	4	4 + 1 (Blender)	2	2	1	1

Chip 외부에 연결할 수 있는 Component는 아래와 같다.

	Camera Sensor	Serializer	Deserializer
최대 지원 개수	8	8	2

MIPI와 연결할 수 있는 Deserializer는 최대 2개이다. Deserializer는 Spec에 따라 최대 4개의 Serializer를 연결할 수 있다.

Camera Sensor Module은 보통 Sensor + Serializer로 연결되어 있다.

6 Write 수행

모든 설정이 완료된 후에 **Save 버튼**을 통하여 기능을 실행하면 Main cluster / Sub cluster를 구분하여 dts 형태의 파일이 생성되어야 한다.

이후, Read 동작 재 실행 시 (또는 저장된 dts를 토대로 자동으로) 변경된 dts 로 부터 업데이트 된 정보를 읽어 아래의 예시와 같이 Read 실행 Screen에 보여져야 한다.

7 Device Tree Parsing 하는 방법

다음은 각 Software driver 들에 대한 Device Tree 설정에 대한 예시로 Camera Resource 가 어떻게 시각화 되어야 하는지에 대한 예시이다.

Software driver로는 CAMSENSOR, SER/DES, MIPI-CSI, ISP, VideoInput(VIN-VIOC) driver가 있다.

Camera Resource는 칩 내부뿐만 아니라 연결될 외부 카메라와 SER/DES 설정까지 이루어져야 한다.

파일은 아래와 같이 device tree가 파싱하여 연결되어 진다.

MainCore : tcc8070-lpd4x322-sv1.0.dts → override/tcc8070-lpd4x322-sv1.0-camera.dtsi → ../tcc807x-camipc.dtsi

SubCore : tcc8070-subcore-lpd4x322-sv1.0.dts → override/tcc8070-lpd4x322-subcore-camera.dtsi → tcc807x-subcore-videoinput-reserved-memory.dtsi (pmap 설정), tcc8070-lpd4x322-videoinput.dtsi (default 설정, all status disabled), ../tcc807x-subcore-camipc.dtsi, tcc807x-subcore-videoinput-camera-module.dtsi → tcc807x-videoinput-svdw-mipi0-hd.dtsi (mipi0에 연결된 hd camera 4channel 설정)

위에서 보았을 때 실질적으로 수정 및 설정하는 부분은 tcc807x-videoinput-svdw-mipi0-hd.dtsi 혹은 tcc807x-videoinput-svdw-mipi0-hd.dtsi 입니다.

7.1 Driver 별로 Parsing 하여야 하는 속성

Driver	속성	추가 내용
CAMSENSOR	status, port, remote-endpoint, io-direction	속성 중에 status = "okay" (활성화된 상태) 인 경우에만 나머지 속성들을 parsing 하면 된다.
SER/DES	status, ports, remote-endpoint, io-direction	속성 중에 status = "okay" (활성화된 상태) 인 경우에만 나머지 속성들을 parsing 하면 된다.
MIPI-CSI	status, ports (input, output), remote-endpoint, num-channel, interleave-mode, hs-settle, data-lanes, channel, pixel-mode	속성 중에 status = "okay" (활성화된 상태) 인 경우에만 나머지 속성들을 parsing 하면 된다.
ISP	status, ports, remote-endpoint, io-direction, cfa, out_win_crop, rgbir-order(cfa가 3인 경우)	속성 중에 status = "okay" (활성화된 상태) 인 경우에만 나머지 속성들을 parsing 하면 된다.

Driver	속성	추가 내용
VIN	status, memory-region, port, remote-endpoint, io-direction, stream-enable, flush-vsync, cam-mux, interrupt-delay, cam-ch	<p>속성 중에 status = "okay" (활성화된 상태) 인 경우에만 나머지 속성들을 parsing 하면 된다.</p> <ul style="list-style-type: none"> videoinput0, 1, 2, 3 → svdw (a.k.a. dewarp) videoinput4, 5 → vwdma videoinput6, 7 → vin
VWDMA	status, memory-region, port, remote-endpoint, io-direction, interrupt-delay, cam-ch, ir-enable, ir-encoding	
SVDW (blender)	<ul style="list-style-type: none"> status, memory-region, remote-endpoint, interrupt-delay frm-time-cycles, frm-timeout-thres, frm-fast-thres, frm-slow-thres, frm-toggle-thres, line-toggle-thres, max-toggle-thres 	
CIED	<ul style="list-style-type: none"> status, rect, winflag, win_dark, win_bright, win_solid, ctype, bswa, byte_swap, hsp, vsp, win_thres, frm_thres, lum_thres r_min_thres, r_max_thres, b_min_thres, b_max_thrs, phz_scal, phz_stride, windows 	기본적으로 디바이스트리에 설정된 속성들을 parsing 하여 사용하고, status = "okay" 인 경우에만 속성들을 수정 할 수 있도록 활성화 한다.
MDW	<ul style="list-style-type: none"> status, memory-region, interrupt-delay, aix-max-ros, aix-max-wos default-color, is_fisheye, color_enable, ir_enable, yuv_standard 	기본적으로 디바이스트리에 설정된 속성들을 parsing 하여 사용하고, status = "okay" 인 경우에만 속성들을 수정 할 수 있도록 활성화 한다.

7.2 CAMSENSOR driver

CAMSENSOR driver
<p>FILE : ../override/tcc8070-lpd4x322-videoinput.dtsi</p> <pre> &i2c13 { #address-cells = <1>; #size-cells = <0>; </pre>

```

/* sensor */
arxxxx_1: arxxxx_1@50 {
    status      = "disabled";
    compatible   = "tcc-onnn,arxxxx";
    reg         = <0x50>;
};

/* external ISP */
cxd5700_1: cxd5700_1@18 {
    status      = "disabled";
    compatible   = "tcc-sony,cxd5700";
    reg         = <0x18>;
};

};

&i2c12 {
    #address-cells = <1>;
    #size-cells   = <0>;

    /* sensor */
    imx424_front: imx424_front@1c {
        status      = "disabled";
        compatible   = "tcc-sony,imx424";
        /* translated by serdes (0x1b -> 0x1c) */
        reg         = <0x1c>;
    };

    imx424_rear: imx424_rear@1d {
        status      = "disabled";
        compatible   = "tcc-sony,imx424";
        /* translated by serdes (0x1b -> 0x1d) */
        reg         = <0x1d>;
    };

    arxxxx: arxxxx@50 {
        status      = "disabled";
        compatible   = "tcc-onnn,arxxxx";
        reg         = <0x50>;
    };

    ar0147: ar0147@10 {
        status      = "disabled";
        compatible   = "tcc-onnn,ar0147";
        reg         = <0x10>;
    };

    ar0820: ar0820@10 {
        status      = "disabled";
        compatible   = "tcc-onnn,ar0820";
        reg         = <0x10>;
    };
};

```

```

ar0231: ar0231@10 {
    status      = "disabled";
    compatible   = "tcc-onnn,ar0231";
    reg         = <0x10>;
};

ar0239: ar0239@10 {
    status      = "disabled";
    compatible   = "tcc-onnn,ar0239";
    reg         = <0x10>;
};

FILE : ./override/tcc807x-videoinput-svdw-mipi0-hd.dtsi

&arxxxx {
    /* Camera Image Sensor */
    status      = "okay";
    port {
        arxxxx_out: endpoint {
            remote-endpoint = <&max96701_in>;
            io-direction    = "output";
        };
    };
};

```

외부 chip이라 i2c address 및 remote-endpoint로 Serializer와 연결해 놓는 과정이 필요하다. (ARXXXX - MAX96701)

위 예시를 보면 ARXXXX(Out) - MAX96701(In) 으로 구성된 것을 알 수 있다.

Item	Description
status	sensor driver를 enable 한다.
reg	i2c slave address를 설정한다.
remote-endpoint	serializer에 들어갈 endpoint를 설정한다.
io-direction	입출력 방향에 대하여 설정한다.(input/output)

7.3 SER(Serializer)/DES(Deserializer) driver

Ser/Des driver

FILE : ./override/tcc8070-lpd4x322-videoinput.dtsi

```
&i2c13 {

    /* serializer */
    max96701_1: max96701_1@40 {
        status      = "disabled";
        compatible   = "tcc-maxim,max96701";
        reg          = <0x40>;
    };

    max9275_1: max9275_1@40 {
        status      = "disabled";
        compatible   = "tcc-maxim,max9275";
        reg          = <0x40>;
    };

    max9295_1: max9295_1@40 {
        status      = "disabled";
        compatible   = "tcc-maxim,max9295";
        reg          = <0x40>;
    };

    /* deserializer */
    max9286_1: max9286_1@48 {
        status      = "disabled";
        compatible   = "tcc-maxim,max9286";
        pwd-gpios    = <&gpmc 0 1>;
        reg          = <0x48>;          // 0x90 >> 1
    };

    max96712_1: max96712_1@2a {
        status      = "disabled";
        compatible   = "tcc-maxim,max96712";
        pwd-gpios    = <&gpmc 0 1>;
        reg          = <0x2A>;
    };
};

&i2c12 {

    /* serializer */
    max96717: max96717@40 {
        /* imx424 rear cam */
        status      = "disabled";
        compatible   = "tcc-maxim,max96717";
        reg          = <0x40>;
    };
};
```

```

max96701: max96701@40 {
    status          = "disabled";
    compatible       = "tcc-maxim,max96701";
    reg              = <0x40>;
};

max9275: max9275@40 {
    status          = "disabled";
    compatible       = "tcc-maxim,max9275";
    reg              = <0x40>;
};

max9295: max9295@40 {
    /* ar0820 */
    status          = "disabled";
    compatible       = "tcc-maxim,max9295";
    reg              = <0x40>;
};

max96705: max96705@40 {
    status          = "disabled";
    compatible       = "tcc-maxim,max96705";
    reg              = <0x40>;
};

max9295e: max9295e@42 {
    /* imx424 front cam */
    status          = "disabled";
    compatible       = "tcc-maxim,max9295";
    reg              = <0x42>;
};

/* deserializer */
max9286: max9286@48 {
    status          = "disabled";
    compatible       = "tcc-maxim,max9286";
    pwd-gpios       = <&gpg 5 1>;
    //rst-gpios      = <&gpb 15 1>; /* mcnex module sensor */
    reg              = <0x48>;      // 0x90 >> 1
};

max96712: max96712@2a {
    status          = "disabled";
    compatible       = "tcc-maxim,max96712";
    pwd-gpios       = <&gpg 5 1>;
    rst-gpios       = <&gpma 17 1>; /* linked to GPI01 */
    reg              = <0x2A>;
};
};

```

FILE : ./override/tcc807x-videoinput-svdw-mipi0-hd.dtsi

```

&max96701 {
    /* serializer */
    status      = "okay";
    ports {
        #address-cells = <1>;
        #size-cells = <0>;
        port@0 {
            reg = <0>;
            max96701_in: endpoint {
                remote-endpoint = <&arxxx_out>;
                io-direction    = "input";
            };
        };
        port@1 {
            reg = <1>;
            max96701_out: endpoint {
                remote-endpoint = <&max9286_in0>;
                io-direction    = "output";
            };
        };
    };
};

&max9286 {
    /* deserializer */
    status      = "okay";
    /*
     * broadcasting mode access each linked devices
     * by the same I2C slave address.
     *
     * Also,
     * using the serdes I2C address mapping table,
     * each linked devices can be accessed
     * by the unique I2C slave address.
     */
    broadcasting-mode;
    ports {
        #address-cells = <1>;
        #size-cells = <0>;
        /*
         * 0 ~ 3
         * input ports. The number is matched with VC
         *
         * 4
         * output port.
         */
        port@0 {
            reg = <0>;
            max9286_in0: endpoint {
                remote-endpoint = <&max96701_out>;
                io-direction    = "input";
            };
        };
    };
};

```



```
};  
    };  
port@1 {  
    reg = <1>;  
    max9286_in1: endpoint {  
        remote-endpoint = <&max96701_out>;  
        io-direction     = "input";  
        channel           = <1>;  
    };  
};  
port@2 {  
    reg = <2>;  
    max9286_in2: endpoint {  
        remote-endpoint = <&max96701_out>;  
        io-direction     = "input";  
        channel           = <2>;  
    };  
};  
port@3 {  
    reg = <3>;  
    max9286_in3: endpoint {  
        remote-endpoint = <&max96701_out>;  
        io-direction     = "input";  
        channel           = <3>;  
    };  
};  
port@4 {  
    reg = <4>;  
    max9286_out: endpoint {  
        remote-endpoint = <&mipi_csi2_0_in>;  
        io-direction     = "output";  
        channel           = <0>;  
    };  
};  
};
```

위 내용을 토대로 보면 ARXXXX(CameraSensor) - MAX96701(SER) - MAX9286(DES) 으로 연결된다.
CameraSensor당 1개의 Serializer가 붙고 4개의 카메라가 Deserializer로 연결된다.

MAX96701은 port 0은 input port로 arxxx_out과 output port로 max9286_in으로 구성됩니다.

MAX9286은 broadcasting-mode로 4개의 input port와 1개의 output port로 구성됩니다.아래 Table에서 각 속성에 대해 설명한 내용을 기반으로 VIOC Resource를 확인해보면 RDMA14 -> SCALER6 -> WDMA7, RDMA15 -> SCALER7 -> WDMA8 로 구성된 것을 알 수 있다.

Driver	Item	Description
Serializer	status	serializer를 enable("okay")한다.

Driver	Item	Description
	reg	i2c slave address를 설정한다.
	port	입출력에 대한 port를 설정한다.
	remote-endpoint	sensor, deserializer 연결된 endpoint를 설정한다.
	io-direction	입출력방향에 대하여 설정한다.(input/output)
Deserializer	status	deserializer를 enable("okay")한다.
	reg	i2c slave address를 설정한다.
	port	입출력에 대한 port를 설정한다. Deserializer에 따라 최대 4개dml
	remote-endpoint	serializer, mipi-csi2 rx 에 연결된 endpoint를 설정한다.
	io-direction	입출력방향에 대하여 설정한다.(input/output)
	channel	VC(Virtual Channel) number를 맞춘다.

7.4 MIPI-CSI/ISP driver

MIPI-CSI/ISP driver

FILE : ./override/tcc807x-videoinput-odw-mipi0-hd.dtsi (include isp camera)

```
&mipi_csi2_0 {
    status = "okay";
    ports {
        #address-cells = <1>;
        #size-cells = <0>;
```

```

/*
 * 0
 * input port.
 *
 * 1 ~ 4
 * output ports. (1: VC0 ~ 4: VC3)
 */
port@0 {
    reg = <0>;
    mipi_csi2_0_in: endpoint {
        remote-endpoint = <&max9286_out>;
        io-direction    = "input";
        num-channel      = <4>;

        /*
         * 0: CH0 only, no data interleave
         * 1: DT only
         * 2: VC only
         * 3: VC and DT
         */
        interleave-mode = <3>;
        hs-settle        = <17>;
        data-lanes       = <1 2 3 4>;
    };
};
port@1 {
    reg = <1>;
    mipi_csi2_0_out0: endpoint {
        remote-endpoint = <&videoinput0_in>;
        io-direction    = "output";
        channel          = <0>;
        /*
         * 0: Single pixel mode
         * 1: Dual pixel mode (RAW8/10/12, YUV422)
         * 2: Quad pixel mode (RAW8/10/12)
         * 3: Invalid
         */
        pixel-mode      = <1>;
    };
};
port@2 {
    reg = <2>;
    mipi_csi2_0_out1: endpoint {
        remote-endpoint = <&videoinput1_in>;
        io-direction    = "output";
        channel          = <1>;
        pixel-mode      = <1>;
    };
};
port@3 {
    reg = <3>;
    mipi_csi2_0_out2: endpoint {

```

```

        remote-endpoint = <&videoinput2_in>;
        io-direction    = "output";
        channel         = <2>;
        pixel-mode      = <1>;
    };
};
port@4 {
    reg = <4>;
    mipi_csi2_0_out3: endpoint {
        remote-endpoint = <&videoinput3_in>;
        io-direction    = "output";
        channel         = <3>;
        pixel-mode      = <1>;
    };
};
};

FILE : ./override/tcc807x-videoinput-odw-mipi0-hd-ispless.dtsi (ispless camera)
&mipi_csi2_0 {
    status = "okay";
    ports {
        #address-cells = <1>;
        #size-cells = <0>;
        /*
         * 0
         * input port.
         *
         * 1 ~ 4
         * output ports. (1: VC0 ~ 4: VC3)
         */
        port@0 {
            reg = <0>;
            mipi_csi2_0_in: endpoint {
                remote-endpoint = <&max96712_out>;
                io-direction    = "input";
                num-channel     = <4>;
                interleave-mode = <3>;
                hs-settle = <19>; /* 900M */
                data-lanes = <1 2 3 4>;
            };
        };
        port@1 {
            reg = <1>;
            mipi_csi2_0_out0: endpoint {
                remote-endpoint = <&isp0_in>;
                io-direction    = "output";
                channel         = <0>;
                pixel-mode      = <0>;
            };
        };
        port@2 {

```

```

        reg = <2>;
        mipi_csi2_0_out1: endpoint {
            remote-endpoint = <&isp1_in>;
            io-direction    = "output";
            channel          = <1>;
            pixel-mode = <0>;
        };
    };
    port@3 {
        reg = <3>;
        mipi_csi2_0_out2: endpoint {
            remote-endpoint = <&isp2_in>;
            io-direction    = "output";
            channel          = <2>;
            pixel-mode = <0>;
        };
    };
    port@4 {
        reg = <4>;
        mipi_csi2_0_out3: endpoint {
            remote-endpoint = <&isp3_in>;
            io-direction    = "output";
            channel          = <3>;
            pixel-mode = <0>;
        };
    };
};

&isp0 {
    status = "okay";
    /*
     * 0: RRGB
     * 1: NOT used
     * 2: NOT used
     * 3: RGBIR (ISP1, 3 only)
     */
    cfa = <0>;
    mem_share = <1>;
    ports {
        #address-cells = <1>;
        #size-cells = <0>;
        port@0 {
            reg = <0>;
            isp0_in: endpoint {
                remote-endpoint = <&mipi_csi2_0_out0>;
                io-direction    = "input";
            };
        };
        port@1 {
            reg = <1>;
            isp0_out: endpoint {

```

```

                                remote-endpoint = <&videoinput0_in>;
                                io-direction      = "output";
                                };
                            };
    };

    &isp1 {
        status = "okay";
        /*
        * 0: RGGB
        * 1: NOT used
        * 2: NOT used
        * 3: RGBIR (ISP1, 3 only)
        */
        cfa = <0>;
        mem_share = <1>;
        ports {
            #address-cells = <1>;
            #size-cells = <0>;
            port@0 {
                reg = <0>;
                isp1_in: endpoint {
                    remote-endpoint = <&mipi_csi2_0_out1>;
                    io-direction      = "input";
                };
            };
            port@1 {
                reg = <1>;
                isp1_out: endpoint {
                    remote-endpoint = <&videoinput1_in>;
                    io-direction      = "output";
                };
            };
        };
    };

    &isp2 {
        status = "okay";
        /*
        * 0: RGGB
        * 1: NOT used
        * 2: NOT used
        * 3: RGBIR (ISP1, 3 only)
        */
        cfa = <0>;
        mem_share = <1>;
        ports {
            #address-cells = <1>;
            #size-cells = <0>;
            port@0 {
                reg = <0>;

```

```

        isp2_in: endpoint {
            remote-endpoint = <&mipi_csi2_0_out2>;
            io-direction    = "input";
        };
    };
    port@1 {
        reg = <1>;
        isp2_out: endpoint {
            remote-endpoint = <&videoinput2_in>;
            io-direction    = "output";
        };
    };
};

&isp3 {
    status = "okay";
    /*
    * 0: RGGB
    * 1: NOT used
    * 2: NOT used
    * 3: RGBIR (ISP1, 3 only)
    */
    cfa = <0>;
    mem_share = <1>;
    ports {
        #address-cells = <1>;
        #size-cells = <0>;
        port@0 {
            reg = <0>;
            isp3_in: endpoint {
                remote-endpoint = <&mipi_csi2_0_out3>;
                io-direction    = "input";
            };
        };
        port@1 {
            reg = <1>;
            isp3_out: endpoint {
                remote-endpoint = <&videoinput3_in>;
                io-direction    = "output";
            };
        };
    };
};
};

```

아래 Table에서 각 속성에 대해 설명한 내용을 기반으로 VIOC Resource를 확인해보면 RDMA7 -> WDMA1로 구성된 것을 알 수 있다.

Driver	Item	Description
MIPI_CSI2	status	mipi-csi를 enable한다.
	port	입출력에 대한 port를 설정한다. Deserializer에서 나오는 VC(Virtual Channel)가 4개일 경우, input 1개, output 4개로 설정한다.
	remote-endpoint	deserializer와 내부 isp 혹은 videoinput과 연결할 endpoint를 설정한다.
	io-direction	입출력방향에 대하여 설정한다.(input/output)
	num-channel	input port시 들어오는 channel 갯수를 설정한다.
	interleave-mode	default : <3>, interleave mode를 설정한다. <ul style="list-style-type: none"> 0 : CH0 only, no data interleave, 1: DT only, 2: VC only, 3: VC and DT
	hs-settle	MIPI-CSI2 Tx(Deserializer)에서 들어오는 bit rate에 따라 설정한다. 아래 표 참조 bit rate는 clock lane의 frequency의 2배를 설정한다. (MIPI-CSI2는 Double Data Rate를 사용한다.)
	data-lanes	MIPI-CSI2 TX에서 들어오는 data lanes 수를 설정합니다. <ul style="list-style-type: none"> 1 lane : <1> , 2 lanes : <1 2> , 3 lanes : <1 2 3> , 4 lanes : <1 2 3 4>
	pixel-mode	pixel mode를 설정한다.(output port) <ul style="list-style-type: none"> 0 : Single pixel mode, 1: Dual pixel mode (RAW8/10/12, YUV422), 2: Quad pixel mode (RAW8/10/12), 3: Invalid
ISP	status	isp0~3를 enable한다.
	port	입출력에 대한 port를 설정한다. input, output 1개씩 있다.

Driver	Item	Description
	remote-endpoint	mipi-csi 와 videoinput으로 연결할 endpoint를 설정한다.
	io-direction	입출력방향에 대하여 설정한다.(input/output)

Table 2.5 HS Settle Value Table (A)

Bit Rate (Mbps)	HS Settle (Decimal)	Bit Rate (Mbps)	HS Settle (Decimal)
2070 < bit rate ≤ 2100	46	1030 < bit rate ≤ 1080	23
2030 < bit rate ≤ 2070	45	990 < bit rate ≤ 1030	22
1980 < bit rate ≤ 2030	44	940 < bit rate ≤ 990	21
1940 < bit rate ≤ 1980	43	900 < bit rate ≤ 940	20
1890 < bit rate ≤ 1940	42	850 < bit rate ≤ 900	19
1850 < bit rate ≤ 1890	41	810 < bit rate ≤ 850	18
1800 < bit rate ≤ 1850	40	760 < bit rate ≤ 810	17
1760 < bit rate ≤ 1800	39	720 < bit rate ≤ 760	16
1710 < bit rate ≤ 1760	38	670 < bit rate ≤ 720	15
1670 < bit rate ≤ 1710	37	630 < bit rate ≤ 670	14
1620 < bit rate ≤ 1670	36	580 < bit rate ≤ 630	13
1580 < bit rate ≤ 1620	35	540 < bit rate ≤ 580	12
1530 < bit rate ≤ 1580	34	490 < bit rate ≤ 540	11
1490 < bit rate ≤ 1530	33	450 < bit rate ≤ 490	10
1440 < bit rate ≤ 1490	32	400 < bit rate ≤ 450	9
1400 < bit rate ≤ 1440	31	360 < bit rate ≤ 400	8
1350 < bit rate ≤ 1400	30	310 < bit rate ≤ 360	7
1310 < bit rate ≤ 1350	29	270 < bit rate ≤ 310	6
1260 < bit rate ≤ 1310	28	220 < bit rate ≤ 270	5
1220 < bit rate ≤ 1260	27	180 < bit rate ≤ 220	4
1170 < bit rate ≤ 1220	26	130 < bit rate ≤ 180	3
1120 < bit rate ≤ 1170	25	90 < bit rate ≤ 130	2
1080 < bit rate ≤ 1120	24	80 < bit rate ≤ 90	1

7.5 VideoInput driver

VideoInput driver

```
&videoinput0 {
    status          = "okay";
    memory-region   = <0 &pmap_rearcamera>;
    port {
        videoinput0_in: endpoint {
            remote-endpoint = <&mipi_csi2_0_out0>;
            io-direction    = "input";
        };
    };
};
```

```

    };
};

&videoinput1 {
    status          = "okay";
    memory-region   = <0 &pmap_rearcamera1>;
    port {
        videoinput1_in: endpoint {
            remote-endpoint = <&mipi_csi2_0_out1>;
            io-direction    = "input";
        };
    };
};

&videoinput2 {
    status          = "okay";
    memory-region   = <0 &pmap_rearcamera2>;
    port {
        videoinput2_in: endpoint {
            remote-endpoint = <&mipi_csi2_0_out2>;
            io-direction    = "input";
        };
    };
};

&videoinput3 {
    status          = "okay";
    memory-region   = <0 &pmap_rearcamera3>;
    port {
        videoinput3_in: endpoint {
            remote-endpoint = <&mipi_csi2_0_out3>;
            io-direction    = "input";
        };
    };
};
};

```

Driver	Item	Description
VideoInput	status	0xffff인 경우에는 사용되는 것이 아니기 때문에 MC 는 사용되지 않는다
	memory-region	pmap mapping 한다. 주로 preview에 대한 설정을 진행한다.
	io-direction	입출력방향에 대하여 설정한다.(input/output)

7.6 CIED driver

CIED driver

```
&cied0 {
    status = "okay";

    rect = <0 0 1280 720>;

    /*
     * winflg : enable winx calculate bit
     * win0 ~ win8 enable : 0x1FF
     * win0 ~ win8 disable : 0x000
     */
    winflag = <0x000>;

    /*
     * win_dark: enable detect dark error in winx
     * win_dark0 ~ win_dark8 enable : 0x1FF
     * win_dark0 ~ win_dark8 disable : 0x000
     */
    win_dark   = <0x000>;
    win_bright = <0x000>;
    win_phase  = <0x000>;
    win_solid  = <0x000>;

    /*
     * ctype 0 : RGB
     * ctype 1 : YUV444
     * ctype 2 : YUV422
     */
    ctype = <0x0>;

    /*
     * bswa 0 : RGB or YUV444
     * bswa 1 : BGR or VUY
     */
    bswa = <0x0>;

    /*
     * byte_swap 0x0 : [23:00]
     * byte_swap 0x1 : [23:16]
     *                [07:00]
     *                [15:08]
     */
    byte_swap = <0x0>;
```

```

/*
 * hsp 0 : Hsync Active low
 * hsp 1 : Hsync Active high
 */
hsp = <0x0>;

/*
 * vsp 0 : Vsync Active low
 * vsp 1 : Vsync Active high
 */
vsp = <0x0>;

darkness {
    win_thres = <0x05>;
    frm_thres = <0x1E>;
    lum_thres = <0x05>;
    status = "disabled";
};

brightness {
    win_thres = <0x05>;
    frm_thres = <0x1E>;
    lum_thres = <0xF5>;
    status = "disabled";
};

frozen {
    frm_thres = <0x05>;
    status = "disabled";
};

solid {
    win_thres = <0x05>;
    frm_thres = <0x1E>;
    r_min_thres = <0x10>;
    r_max_thres = <0x70>;
    b_min_thres = <0x10>;
    b_max_thres = <0x70>;
    status = "disabled";
};

phase {
    win_thres = <0x05>;
    frm_thres = <0x1E>;
    phz_scale = <0x00>;
    phz_stride = <0x03>;
    status = "disabled";
};

windows {
    /* [left] [top] [width] [height] [phase_thres] */
    window0 = <0 0 360 360 10>;

```

```

window1 = <0 0 360 360 10>;
window2 = <30 40 720 480 10>;
window3 = <50 60 720 480 10>;
window4 = <50 60 720 480 10>;
window5 = <100 100 720 480 10>;
window6 = <200 200 720 480 10>;
window7 = <300 250 720 480 10>;
window8 = <0 0 720 480 10>;

};
};

```

Driver	Item	UI Match & Description
CIED	status	status - OK : 설정 창 활성화 - Disable : 설정 창 비활성화
	rect	Input Image size - 최소 0x0 - 최대 16383x16383
	winflag	Window calculate - 0x0 ~ 0x1ff : bit0 → window0, bit1 → window1
	win_dark	Dark Error Detect winodw - 0x0 ~ 0x1ff : bit0 → window0, bit1 → window1
	win_bright	Bright Error Detect winodw - 0x0 ~ 0x1ff : bit0 → window0, bit1 → window1
	win_phase	Phase Error Detect winodw - 0x0 ~ 0x1ff : bit0 → window0, bit1 → window1
	win_solid	Solid Error Detect winodw - 0x0 ~ 0x1ff : bit0 → window0, bit1 → window1
	ctype	Input Format - 0x0 : RGB - 0x1 : YUV444 - 0x2 : YUV422

Driver	Item	UI Match & Description
	bswa	BSWA(Bit swap) - 0x0 : RGB or YUV - 0x1 : BGR or VUY
	byte_swap	Byte swarp - 0x0 : [23:00] - 0x1 : [23:16] [07:00] [15:08]
	hsp	Hsync polarity - 0x0 : Bypass - 0x1 : reversal
	vsp	Vsync polarity - 0x0 : Bypass - 0x1 : reversal
	win_thres	window count threshold - 0 ~ enable window 갯 수
	frm_thres	Frame count threshold - 0 ~ 511
	lum_thres	Luminance count threshold - 0 ~ 255
	r_min_thres	G/R Minium - 0 ~ 255
	r_max_thres	G/R Maxium - 0 ~ 255
	b_min_thres	G/B Minium - 0 ~ 255
	b_max_thres	G/B Maxium - 0 ~ 255
	phz_scale	Phase scale - 0 ~ 3

Driver	Item	UI Match & Description
	phz_stride	Phase stride - 0 ~ 255
	windows	Window size & Phase threshold for window ex) /* [left] [top] [width] [height] [phase_thres] */ window0 = <0 0 360 360 10>; - left : 0 ~ input width - 1 - top : 0 ~ input height - 1 - width : 1 ~ input width - height : 1 ~ input height - Phase threshold for winodw : 0 ~ 65535

7.7 MDW driver

MDW driver		
<pre> &tcc_mdw { status = "okay"; memory-region = <&pmap_mdw>; interrupt-delay = <0x100>; aix-max-ros = <0xF>; aix-max-wos = <0xF>; format { default-color = <0x80808080>; is_fisheye = <0x1>; color_enable = <0x1>; ir_enable = <0x0>; yuv_standard = <0x0>; }; }; </pre>		

Driver	Item	UI Match & Description
CIED	status	status - OK : 설정 창 활성화 - Disable : 설정 창 비활성화
	memory-region	Read한 디바이스트리 내용 값 사용.

Driver	Item	UI Match & Description
	interrupt-delay	Interrupt delay(Done Interrupt delay) - 0x0 ~ 0xffff
	aix-max-ros	Max ros(Maximum AXI read outstanding count) - 0x0 ~ 0x3f
	aix-max-wos	Max wos(Maximum AXI write outstanding count) - 0x0 ~ 0x3f
	default-color	Default color - 0x0 ~ 0xffffffff
	is_fisheye	Is fisheye - 0x0 : Model for non-fisheye lenses. - 0x1 : Model for fisheye lenses.
	color_enable	Color enable - 0x0 : disable - 0x1 : enable
	ir_enable	IR enable - 0x0 : IR data not used - 0x1 : Use IR Data
	yuv_standard	yuv standard(RGB-YUV color conversion standard) - 0x0 : BT.601 with footroomd/headroom - 0x1 : BT.601 JPEG - 0x2 : BT.709 - 0x3 : BT.2020

