

Cheap Complex Cameras

Pavel Machek
<pavel@denx.de>



About me

Kernel hacker

Decided to want up-to-date kernel on cellphone

...and that's N900

Wanted flash LED control



Hardware is cheap and complex

flash

voice coil support for focus

two sensors

- back camera - et8ek8

- front camera - smiapp

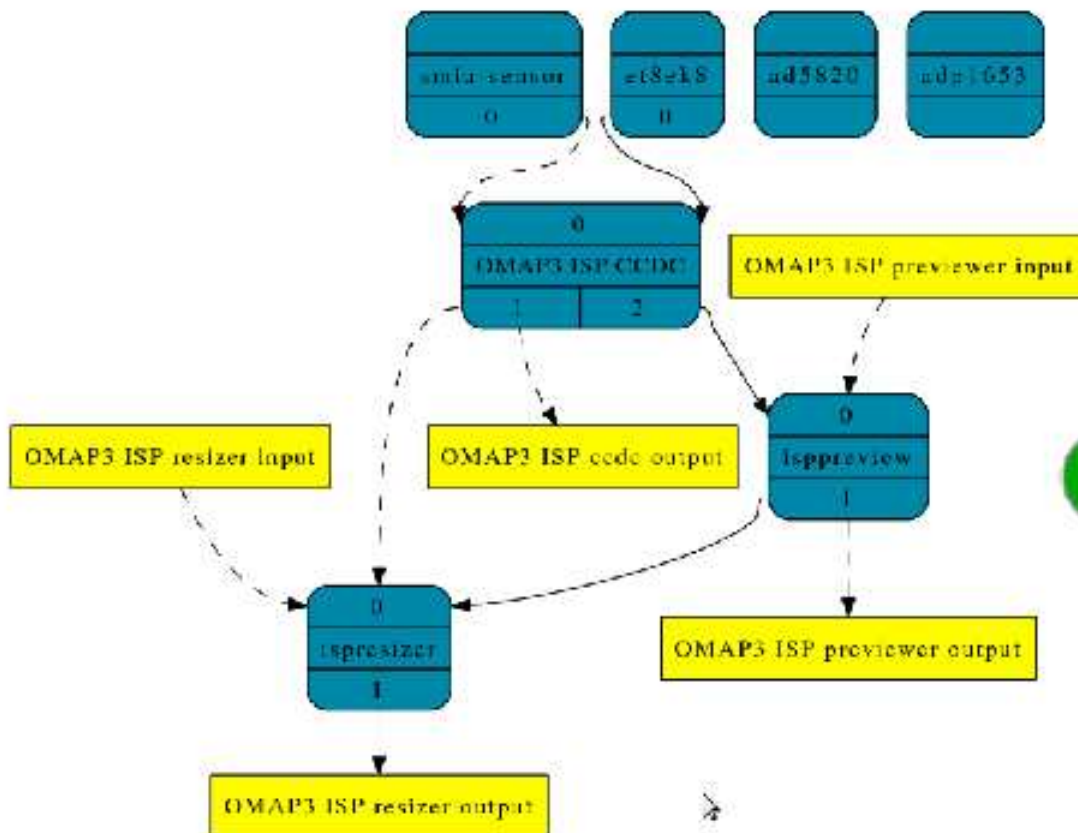
GPIO controlled switch

front end

preview module

resizer

statistics collection



Nokia N900

12.4.3.2 CSI2 Receiver Block Diagram

Figure 12-62 is the block diagram of the CSI2 receiver connected to the com

Figure 12-62. CSI2 Receiver Block Diagram

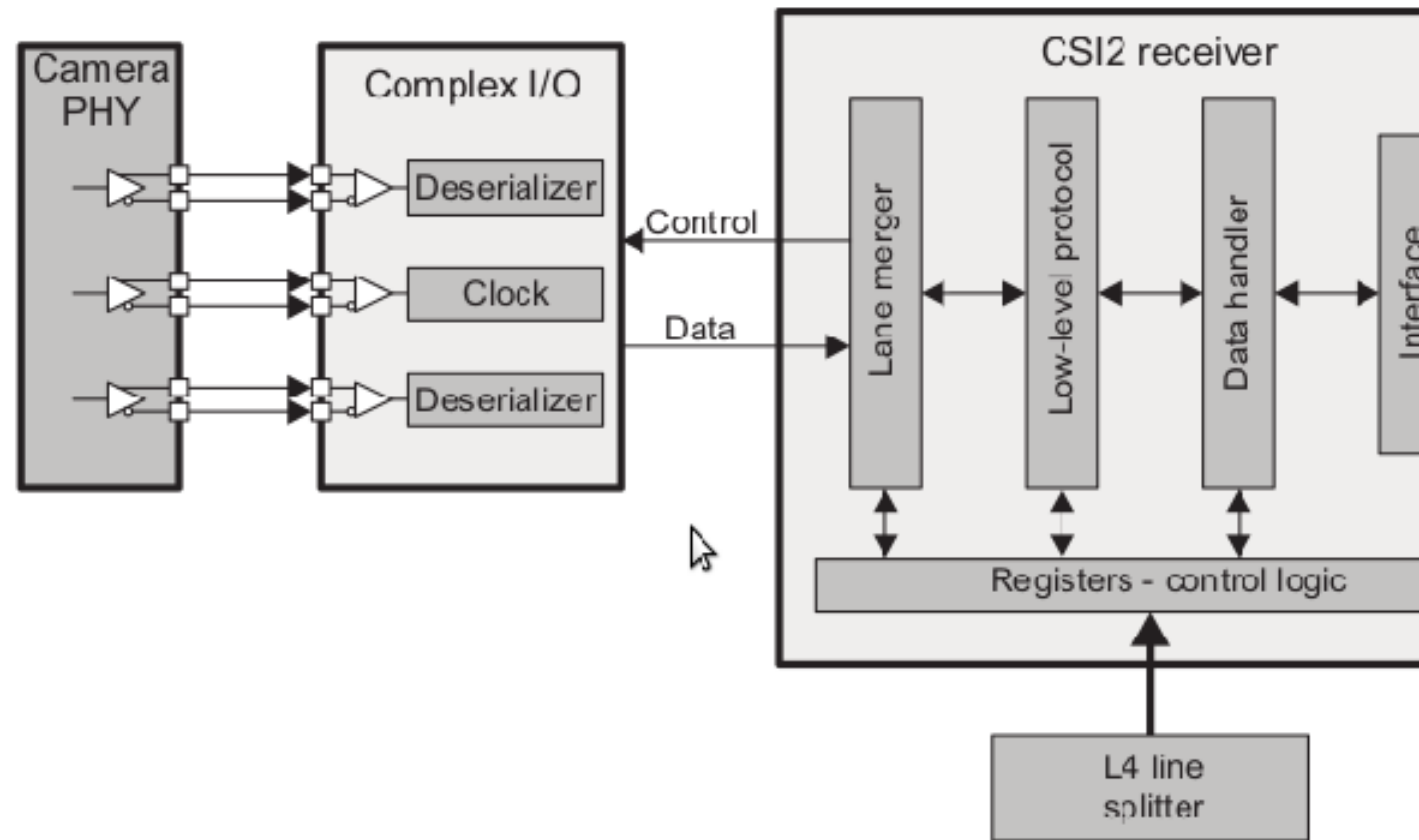


Figure 12-72. CCDC Block Diagram

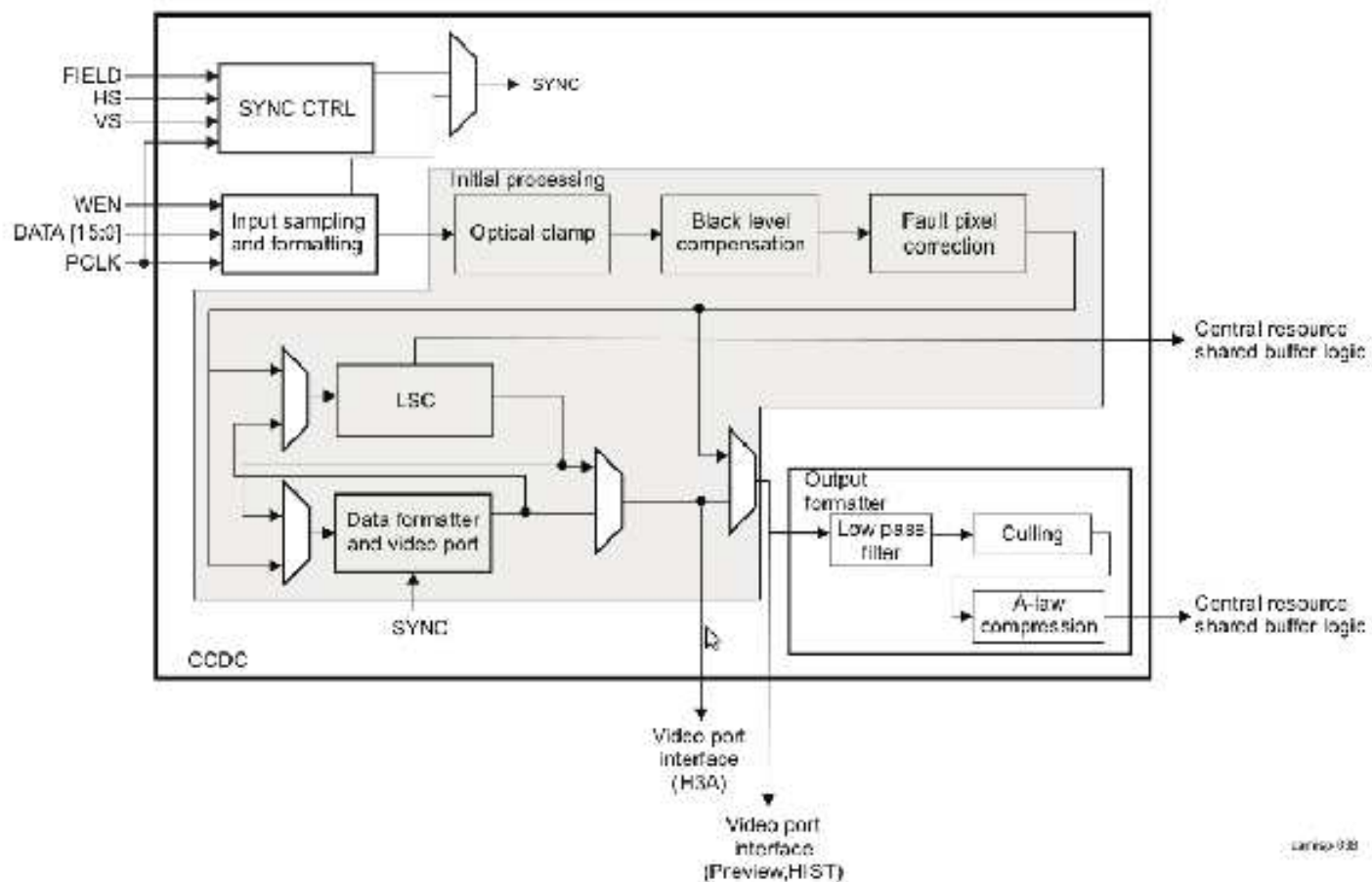
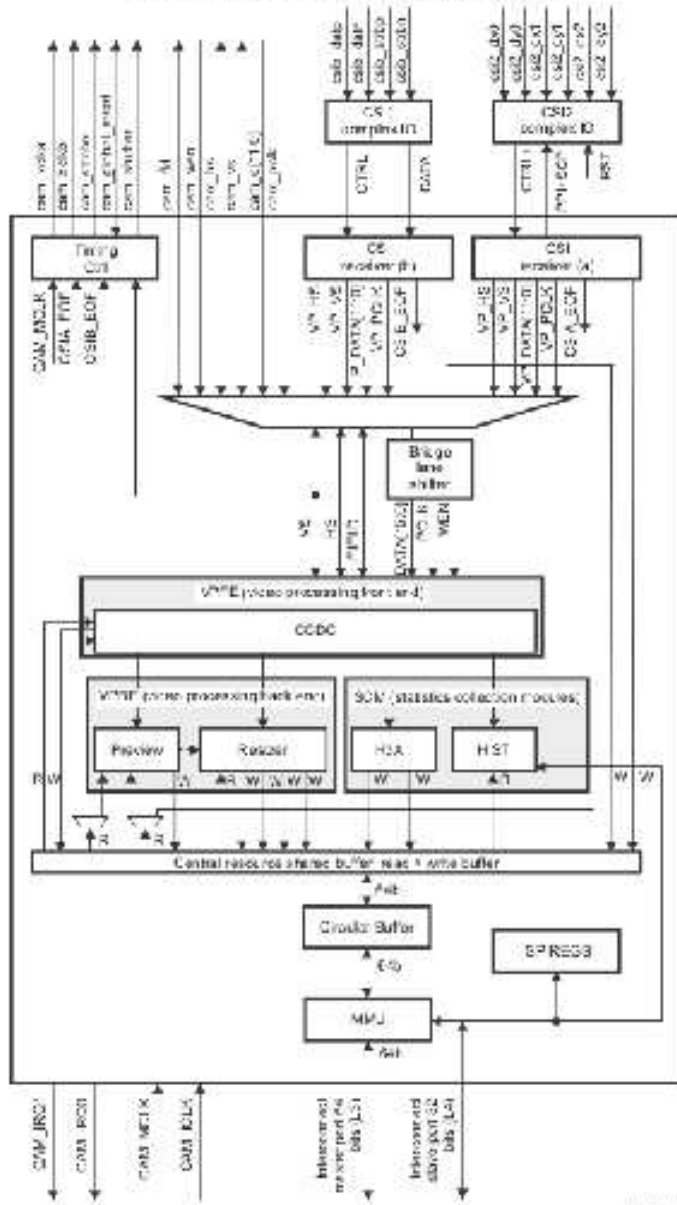


Table 12-28 summarizes allowed data flows through the CCDC.

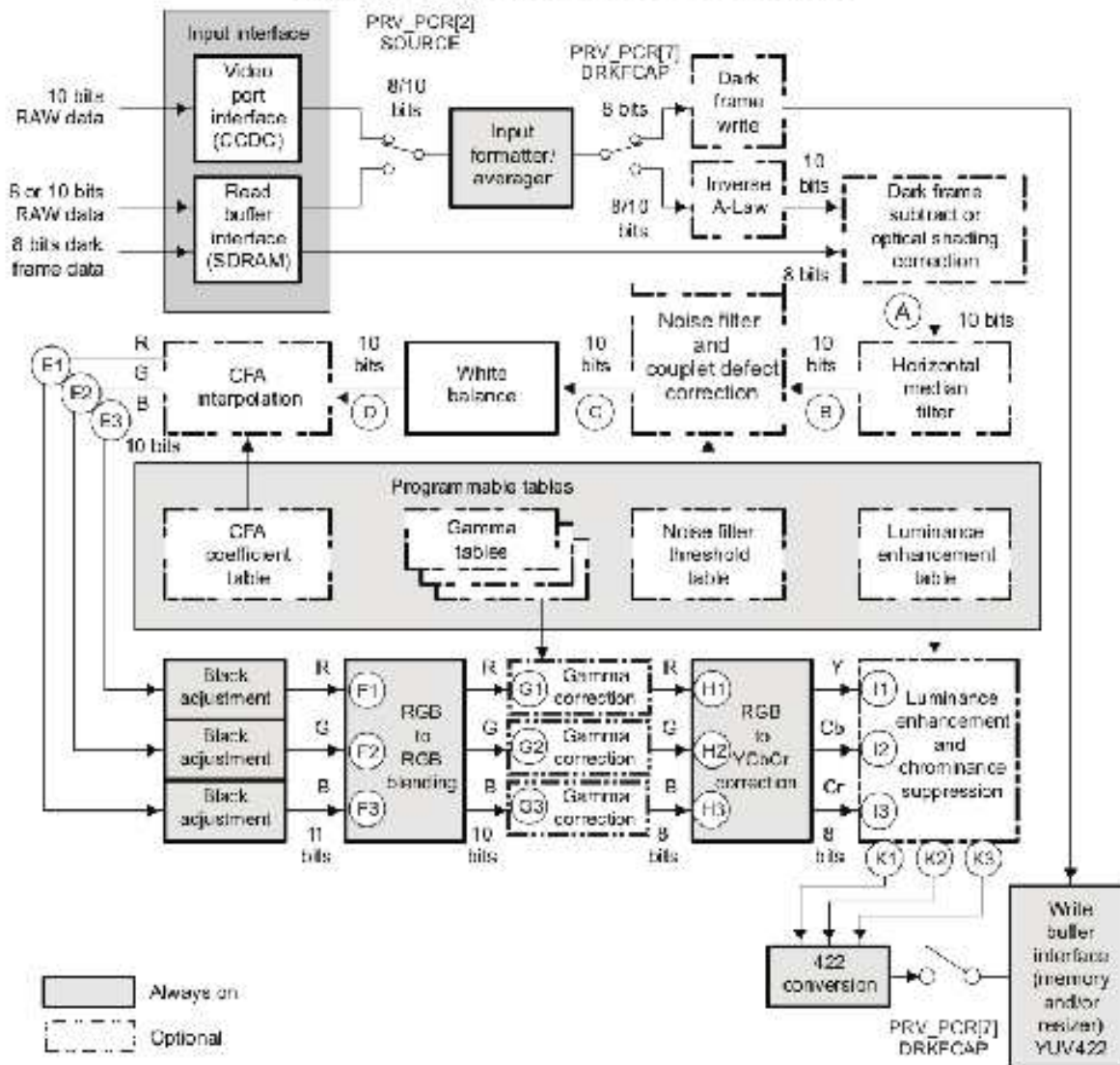
Table 12-28. Allowed Data Flows Through the CCDC

Figure 12-53. Camera ISP Block Diagram



12-53

Figure 12-78. Preview Engine Block Diagram

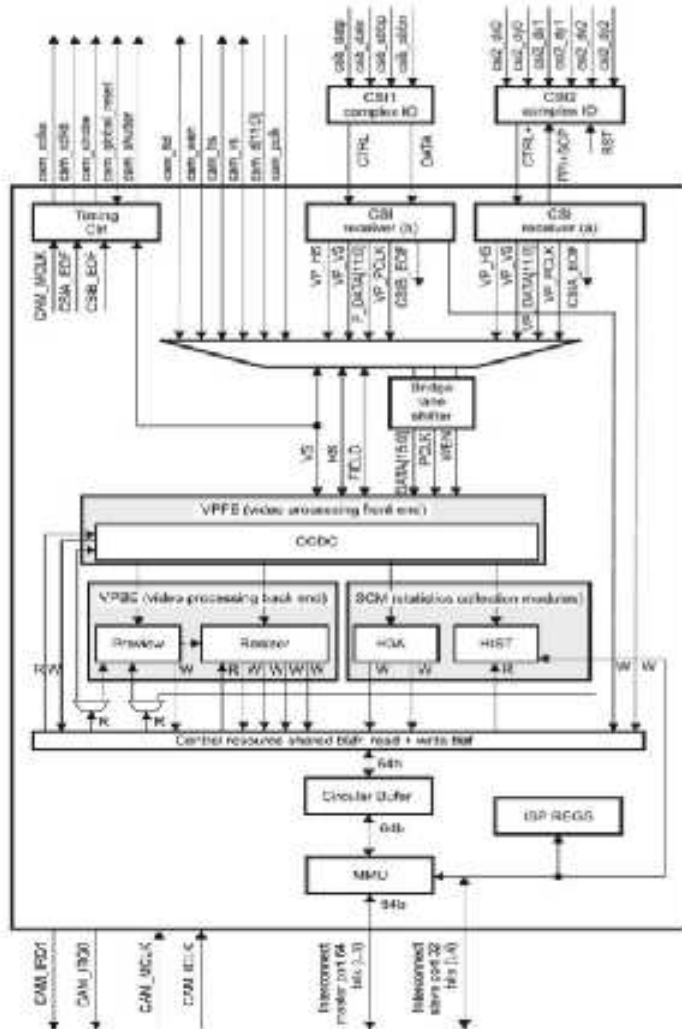


Embedded mess

2010-02-06

Media controller - FOSDEM 2010

31



OMAP3430 ISP

- Reconfigurable pipeline
- Parallel processing
- Memory-to-memory paths
- Fine-grain parameters

How do we handle the zillion configuration options through a single video device ?

Drawing is © Texas Instrument

V4L2 makes world simple

TV card: /dev/video0

enum format: 1024x768, RGB24

capture

simple, right?



Media-control API

video0 OMAP3 ISP CCP2 input
video1 OMAP3 ISP CSI2a output
video2 OMAP3 ISP CCDC output
video3 OMAP3 ISP preview input
video4 OMAP3 ISP preview output
video5 OMAP3 ISP resizer input
video6 OMAP3 ISP resizer output
v4l-subdev0 OMAP3 ISP CCP2
v4l-subdev1 OMAP3 ISP CSI2a
v4l-subdev2 OMAP3 ISP CCDC
v4l-subdev3 OMAP3 ISP preview
v4l-subdev4 OMAP3 ISP resizer
v4l-subdev5 OMAP3 ISP AEWB
v4l-subdev6 OMAP3 ISP AF
v4l-subdev7 OMAP3 ISP histogram
v4l-subdev8 ad5820 focus

Is this V4L3 ?

- No, V4L2 is still alive and well
- Best effort to provide V4L2-only compatibility for existing applications (API and ABI)
- Advanced features will require Media Controller

OMAP3430 ISP

- Default pipeline through `/dev/video0`
- Limited set of resolutions, limited set of controls

V4L2 plain API

2010: Media-control API is not V4L3

It really is V4L3

Nothing works before pipeline is setup

Mostly nothing works after that

Not even format enumeration works



Kernel progress

N9 sensor: in 4.13

N900 sensor: merged in 4.14-rc4, 1.3MPix only

AF coil support: being reviewed

flash support: being reviewed for N9



v4l-utils

□ Oct 2017 version

Alive and well

C

No media-control support

□ thus no resolution change

(Poor) auto-gain

No auto-focus

(Unsuitable) auto-whitebalance

8-bit only

Programming interface limited by kernel interface

□ no easy way to add detailed autofocus/autogain control

□ no way to convert existing data



FCam-dev

Full featured camera application

10-bit support

Including autofocus, autogain, raw+jpeg, HDR

Ability to change resolutions

Accelerated preview

Nice programming interface (university project)

C++

Threads

Custom kernel interface

Dead project

Gets us photos, but not application support



Goals

Real

- LED light
- Kernel testing

Bonus

- Basic camera application
- Auto-gain
- Preview
- Some way to get photos
- Quick shutter speed
- Run over ssh



Future goals

Hard

- Accelerated preview
- High-quality jpeg

Very hard

- Video capture
- Concurrent access to camera from multiple applications



Performance research, on 1MPix data

GRBG10 -> RGB24 conversion is too slow

Displaying select pixels is not

- small window, reduced framerate

Sampling 1 in 361 pixels for autogain is not

Sampling three thirds of line for autofocus is not



sdldcam project

Available at git@gitlab.com:tui/v4l-utils.git

Reasonable branch is my-1.13



Bad news

Hard-coded pipeline parameters

Simple ioctl propagation

Hard-coded picture parameters

Simple ui in SDL

Capture into RAW

- .dng is too complex

- .pgm is suitable

8-bit internally

Capture into JPEG is missing stuff

- white balance

- dead pixel processing

- lens shading



Good news

Auto-gain works

Auto-focus mostly works

Fast shutter speed

negative shutter delay possible

RAW capture

Good enough for testing kernels



Auto-gain

old: Average is target

new: Get enough bright pixels

□ but not too many



Auto-focus

Single-shot focus

- sweep whole range
- small steps around best focus

Continuous focus

- constantly moves lens around
- to see if it improves on either side



Wishlist for kernel

Default pipeline config

Format enumeration for media-ctl

Absolute units for controls

Provide capture settings for each frame

V4L2 is too asynchronous

Current interface

- select resolution
- start capture
- frame comes
- select gain
- frame comes (what gain was used?)
- frame comes (what gain was used?)
- frame comes (what gain was used?)
- set focus
- frame comes (what focus was used?)



Wishlist for v4l-utils

Media-ctl support

- resolution changing

libv4lconvert

- API not modelled after kernel one
- usable without /dev/videoX devices

16-bit support

Ability to get pixel color for single pixel



Questions?









Hardware is complex

CSI1 / MIPI CSI2 / parallel interface

front end

- optical clamping
- black-level compensation
- faulty pixel correction
- lens-shading compensation

Hardware is complex

preview module

- A-law compression
- dark frame subtraction
- horizontal median filter
- programmable 3x3 filter
- couplet faulty pixel correction
- digital gain
- white balance
- color filter array interpolation
- black adjustment
- color correction (RGB -> RGB)
- gamma correction

Hardware is complex

preview

- color conversion (RGB->YCbCr 4:4:4)
- color subsampling (YCbCr 4:4:4->4:2:2)
- luminance enhancement

resizer module

- x.25 to x4

statistic collection

- 3A metrics for AWB, AE, AF
- histogram