Bridge of Life Education

Final Project Presentation
HDR Image Signal Processor (ISP)

NTHU/Team 7
Leader:張傑閔
Member:蔡宗穎

©BOLEDU

Outline

- Application of HDR ISP
- HDR ISP Pipeline
- Effort & Content of Work
  - Implementation Status
  - HLS Implementation of HDR ISP Modules
- Analysis - Insight & Finding
- Reference

©BOLEDU

Application of HDR ISP

©BOLEDU

Application

- Application :
  - Preserve the true essence of your surroundings with HDR imaging techniques.

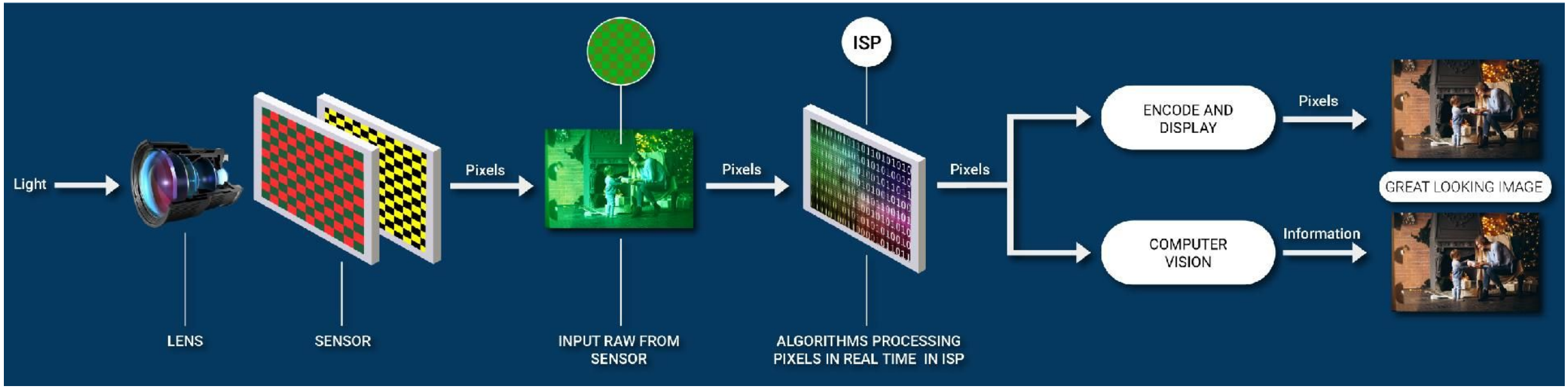
This image is copied from: https://vmi.tv/blog/learn-help/hdr_reality_and_monitoring-a_dops_perspective/

©BOLEDU

HDR ISP Pipeline

©BOLEDU

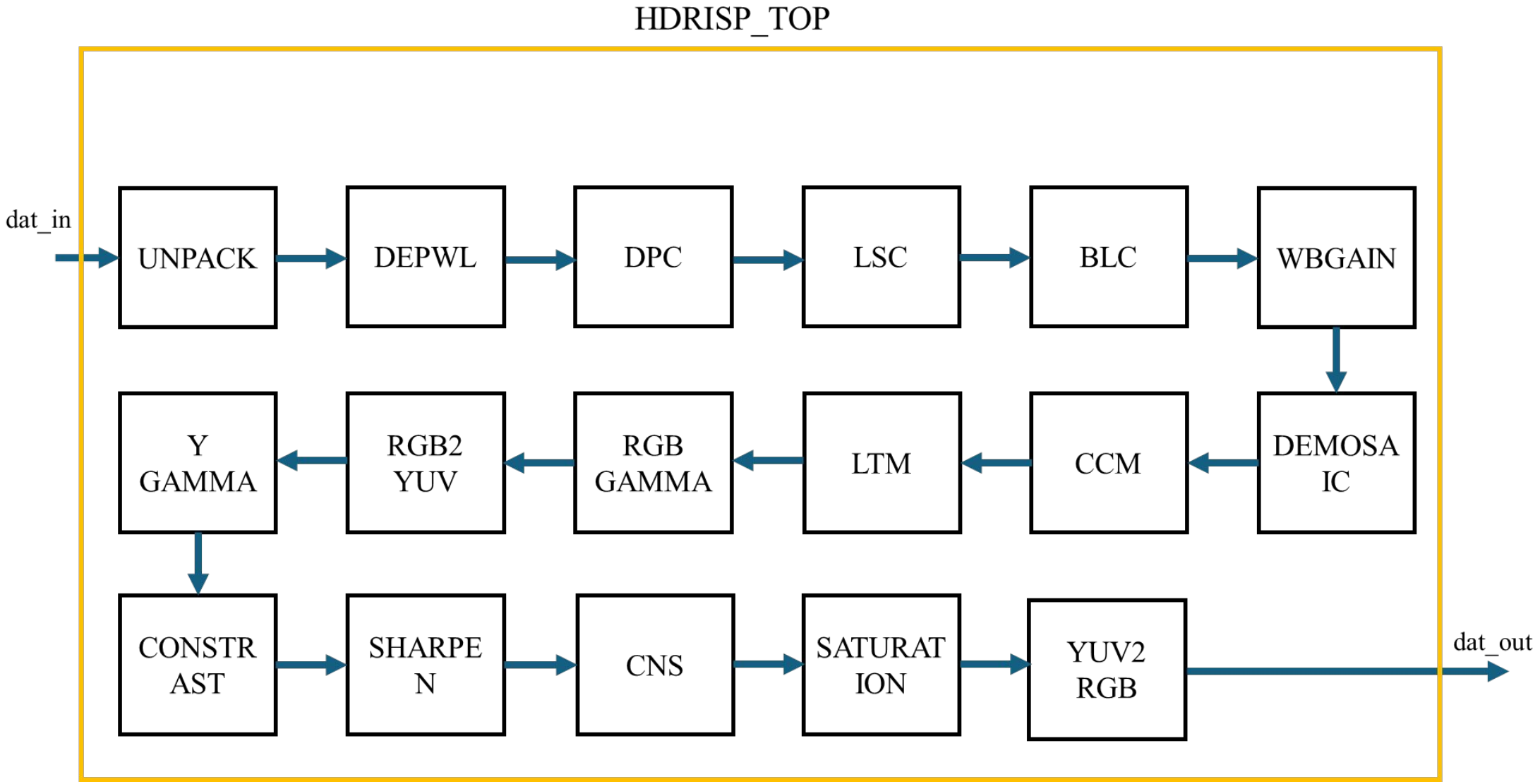
ISP is used to processing pixels in real time



This image is copied from: https://www.faststreamtech.com/blog/image-signal-processor/

©BOLEDU

HDR ISP Pipeline



©BOLEDU

Effort & Content of Work
- Implementation Status

©BOLEDU

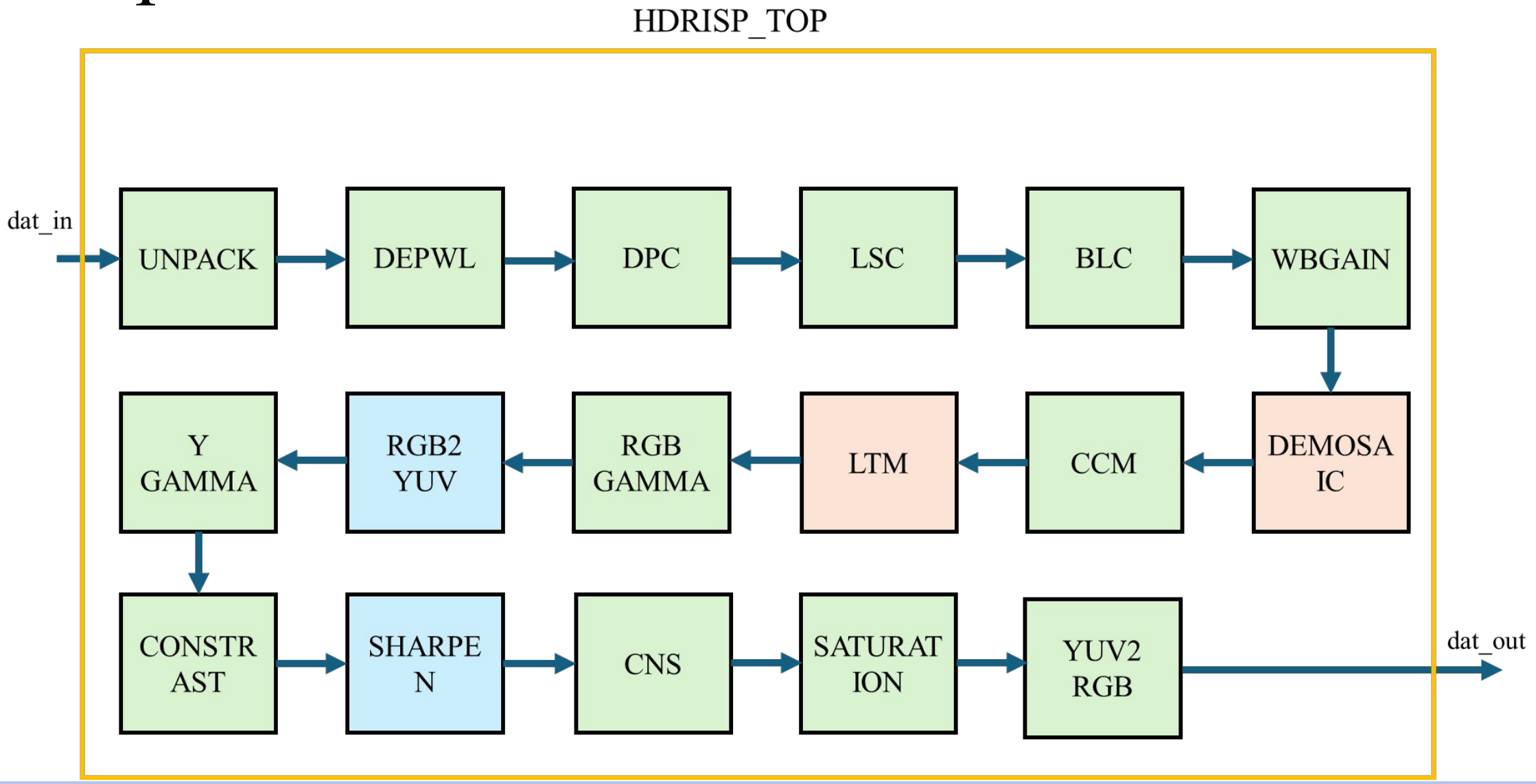
HDR ISP Pipeline Profiling

```
INFO [default] ============= user pipeline print start ==============
INFO [default] mod[0] -> unpack
INFO [default] mod[1] -> depwl
INFO [default] mod[2] -> lsc
INFO [default] mod[3] -> dpc
INFO [default] mod[4] -> blc
INFO [default] mod[5] -> wbgain
INFO [default] mod[6] -> demoasic
INFO [default] mod[7] -> ccm
INFO [default] mod[8] -> ltm
INFO [default] mod[9] -> rgbgamma
INFO [default] mod[10] -> rgb2yuv
INFO [default] mod[11] -> ygamma
INFO [default] mod[12] -> contrast
INFO [default] mod[13] -> sharpen
INFO [default] mod[14] -> cns
INFO [default] mod[15] -> saturation
INFO [default] mod[16] -> yuv2rgb
INFO [default] ============= user pipeline print end ==============
```

```
INFO [default] ============= user pipeline running ==============
INFO [default] mod unpack        time: 0ms
INFO [default] mod depwl         time: 6ms
INFO [default] mod lsc   time: 14ms
INFO [default] mod dpc   time: 10ms
INFO [default] mod blc   time: 3ms
INFO [default] mod wbgain        time: 4ms
INFO [default] mod demoasic      time: 16ms
INFO [default] mod ccm   time: 13ms
INFO [default] mod ltm   time: 1694ms
INFO [default] mod rgbgamma      time: 13ms
INFO [default] mod rgb2yuv       time: 7ms
INFO [default] mod ygamma        time: 5ms
INFO [default] mod contrast      time: 3ms
INFO [default] mod sharpen       time: 40ms
INFO [default] mod cns   time: 242ms
INFO [default] mod saturation    time: 6ms
INFO [default] mod yuv2rgb       time: 10ms
INFO [default] ============= user pipeline running end ==============
```

©BOLEDU

Implementation Status



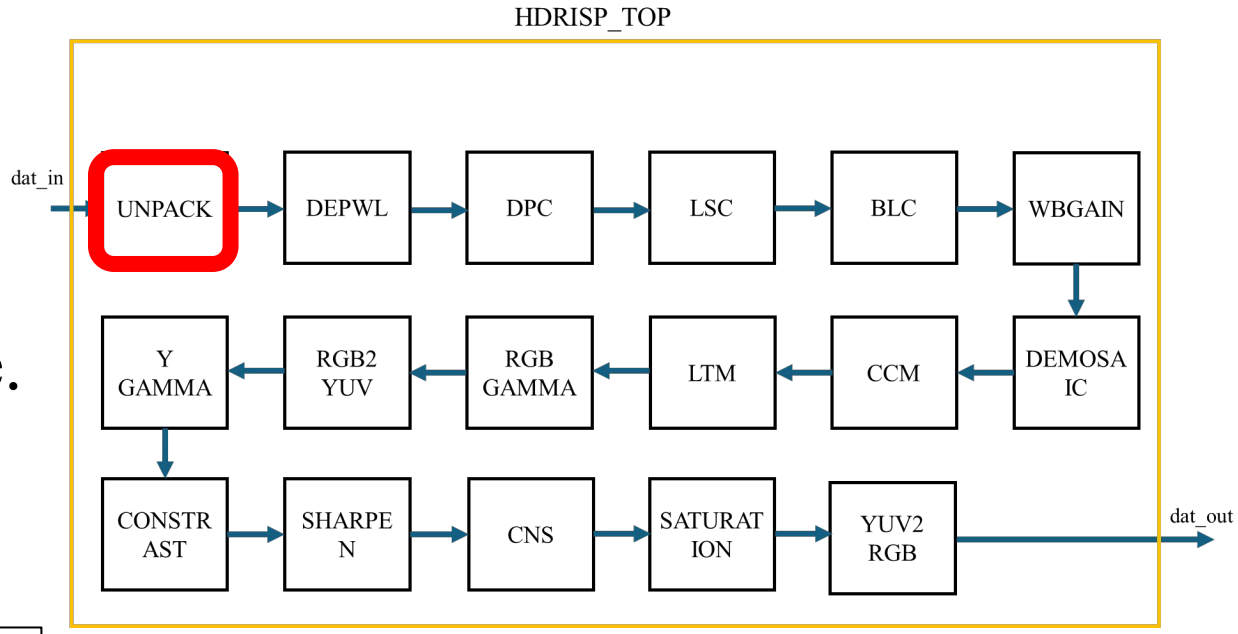
©BOLEDU

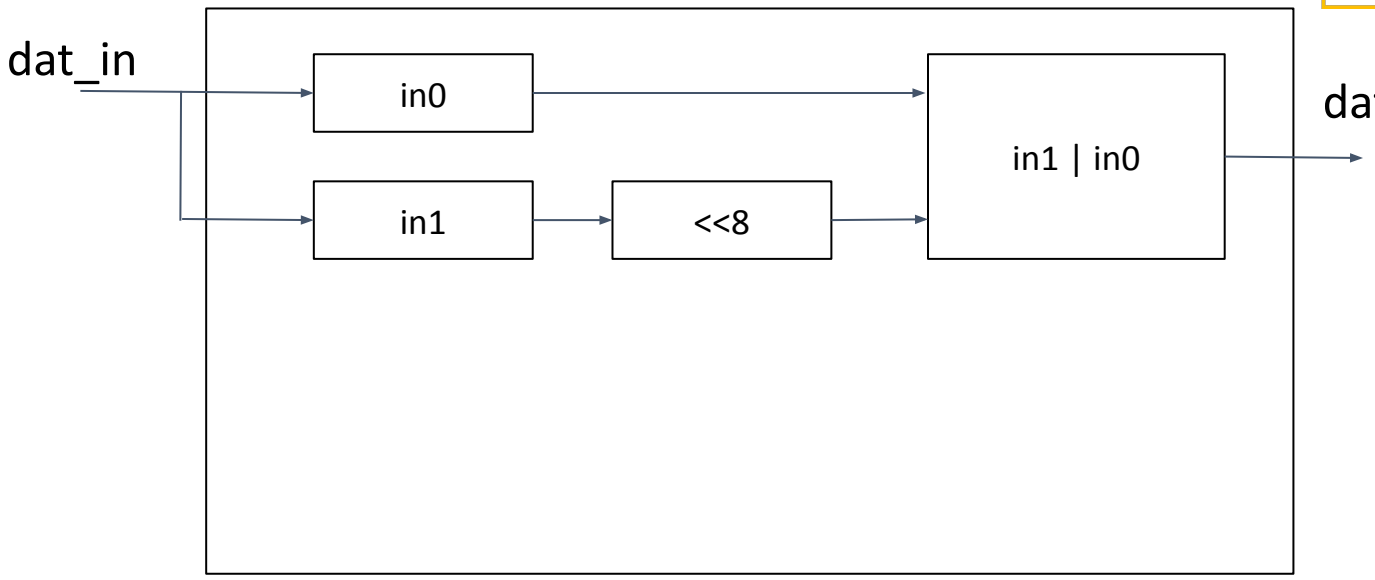
Effort & Content of Work
- HLS Implementation of HDR ISP Modules

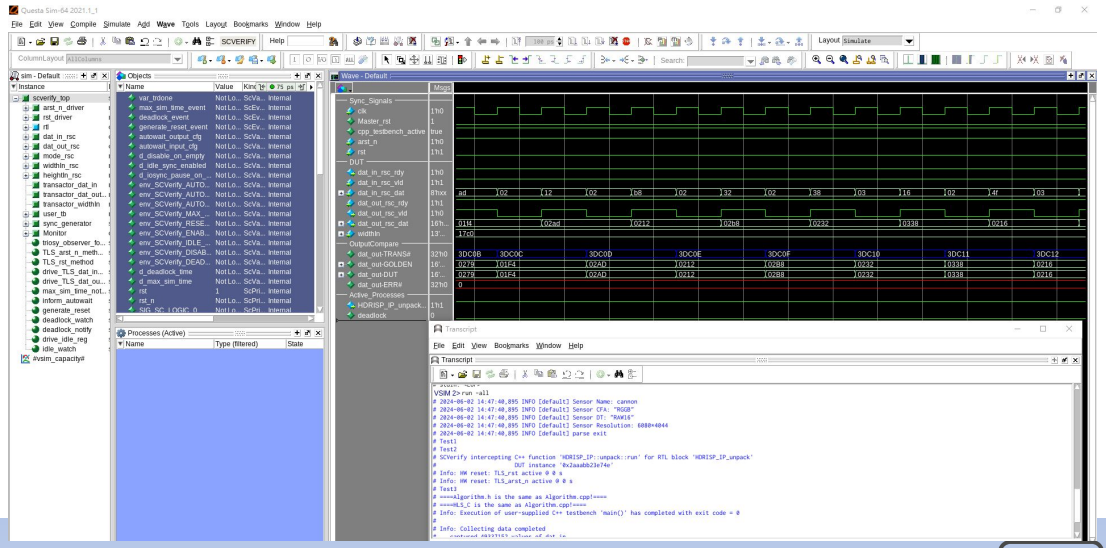
©BOLEDU

Unpack

- Function: Transform data in MIPI (raw16) form into uint16 data type.
- Architecture:



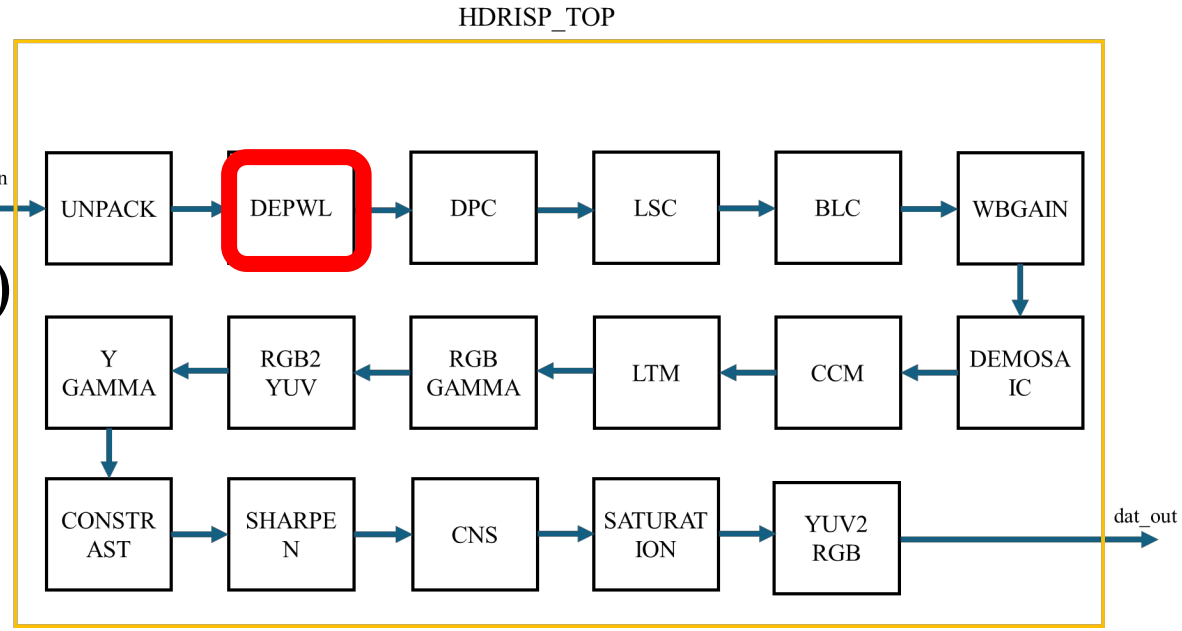


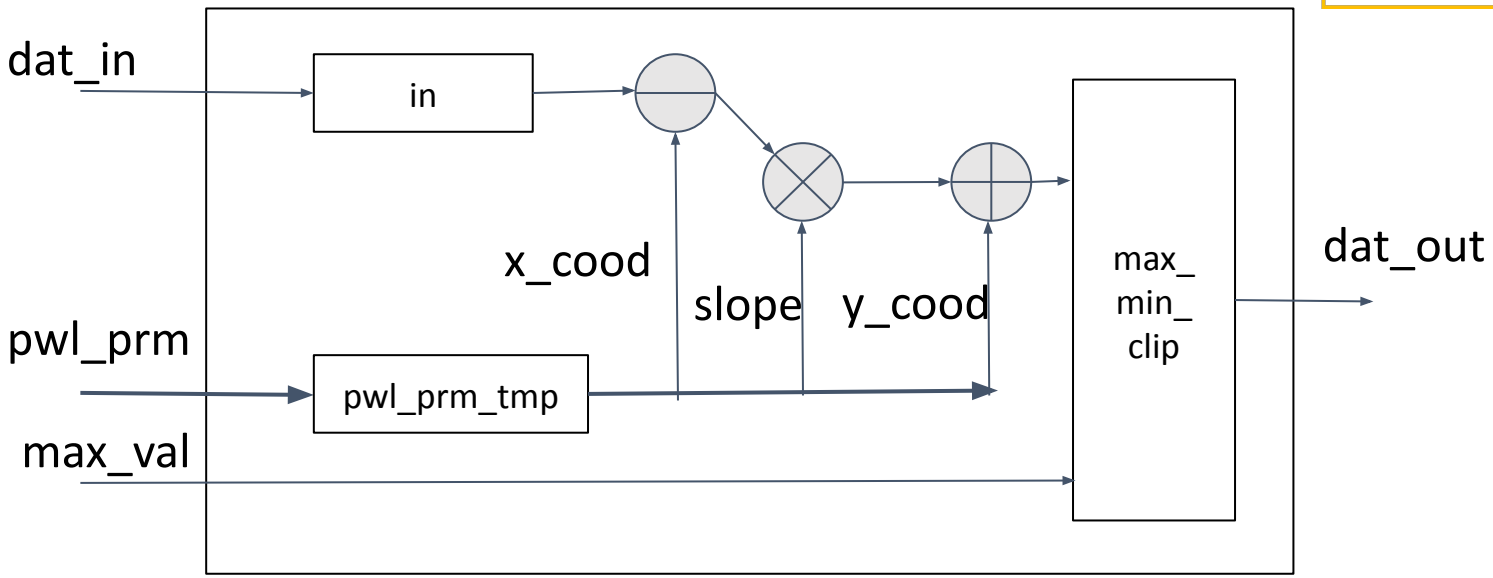


©BOLEDU

Decode PWL

- Function: Decode the 16-bit (uint16) data to 32-bit (int32) data according to a PWL curve.
- Architecture:



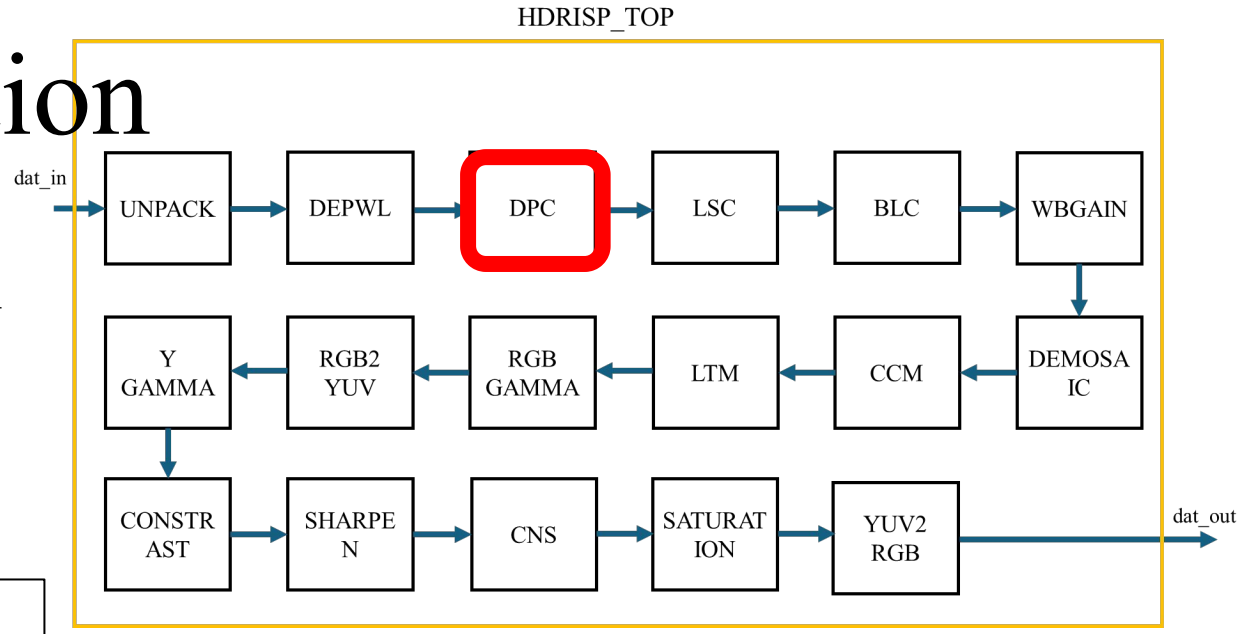


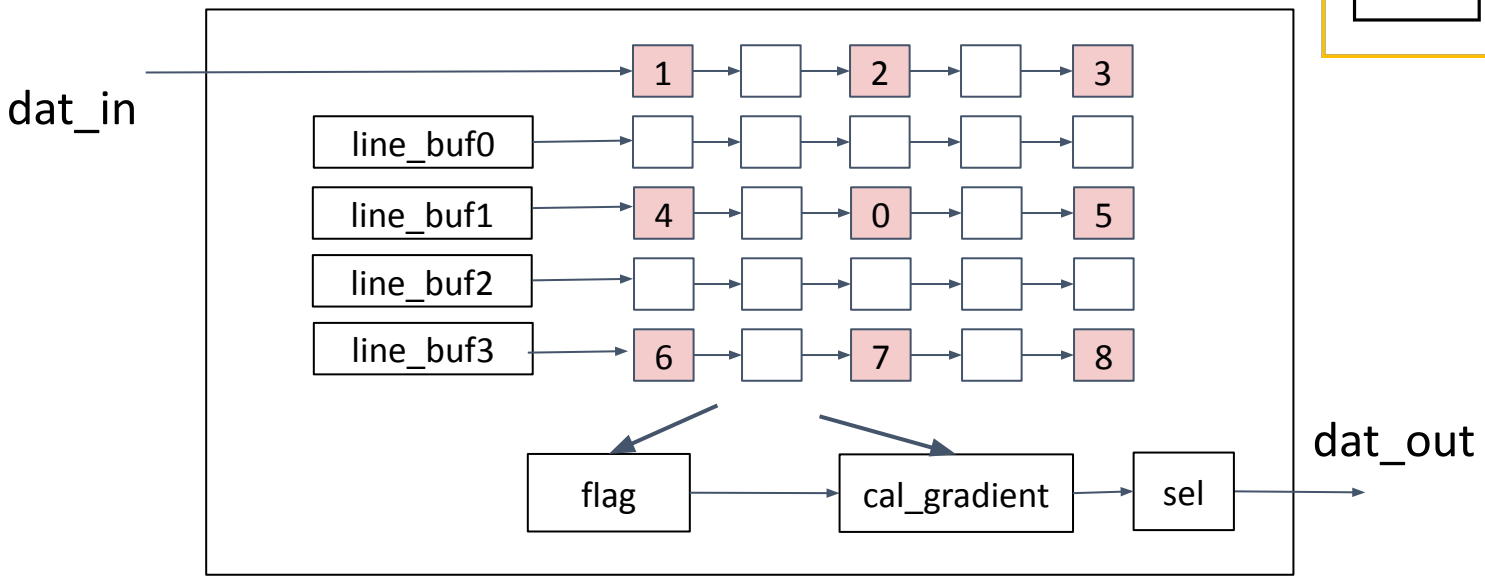

```
# i = 99, data_hls = 2838
# i = 100, data_hls = 1914
# Test2
# isp_prms.depwl_prm.pwl_nums = 2
# depwl_prmTmp.pwl_nums = 2
# Test2_1
# Test2_2
# SCVerify intercepting C++ function 'HDRISP_IP::depwl::run' for RTL block 'HDRISP_IP_depwl'
#                      DUT instance '0x2aaabb2f871d'
# Info: HW reset: TLS_rst active @ 0 s
# Info: HW reset: TLS_arst_n active @ 0 s
# heightIn = 512, widthIn = 768
# Test3
# ====Algorithm.h is the same as Algorithm.cpp!====
# ====HLS_C is the same as Algorithm.cpp!====
# Info: Execution of user-supplied C++ testbench 'main()' has completed with exit code = 0
#
# Info: Collecting data completed
#    captured 393216 values of dat_in
#    captured 1 values of widthIn
#    captured 1 values of heightIn
#    captured 1 values of max_val
#    captured 1 values of pwl_prms_pwl_nums
#    captured 1 values of pwl_prms_pedestal_before_pwl
#    captured 1 values of pwl_prms_pedestal
#    captured 1 values of pwl_prms_x_cood
#    captured 1 values of pwl_prms_y_cood
#    captured 1 values of pwl_prms_slope
#    captured 393216 values of dat_out
# Info: scverify_top/user_tb: Simulation completed
#
# Checking results
# 'dat_out'
#    capture count        = 393216
#    comparison count     = 393216
#    ignore count         = 0
#    error count          = 0
#    stuck in dut fifo    = 0
#    stuck in golden fifo = 0
#
# Info: scverify_top/user_tb: Simulation PASSED @ 3942426 ns
# ** Note: (vsim-6574) SystemC simulation stopped by user.
# 1
```

©BOLEDU

Dynamic Pixel Correction

- Function: Using neighboring pixel values to correct some bad pixels.
- Architecture:



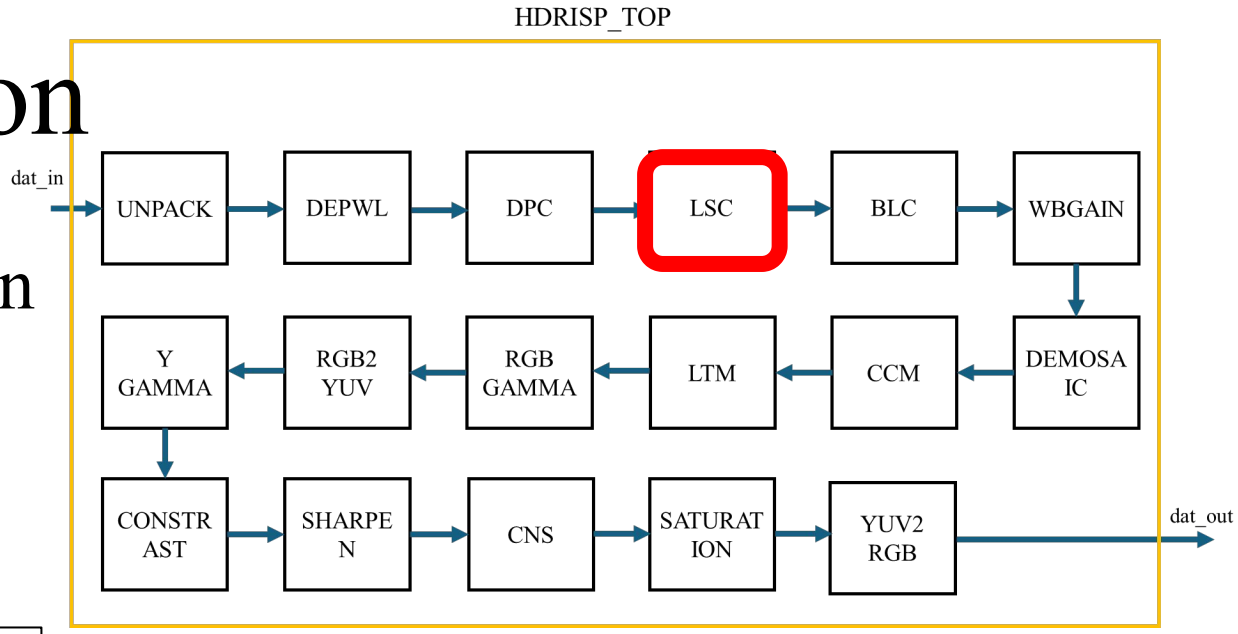


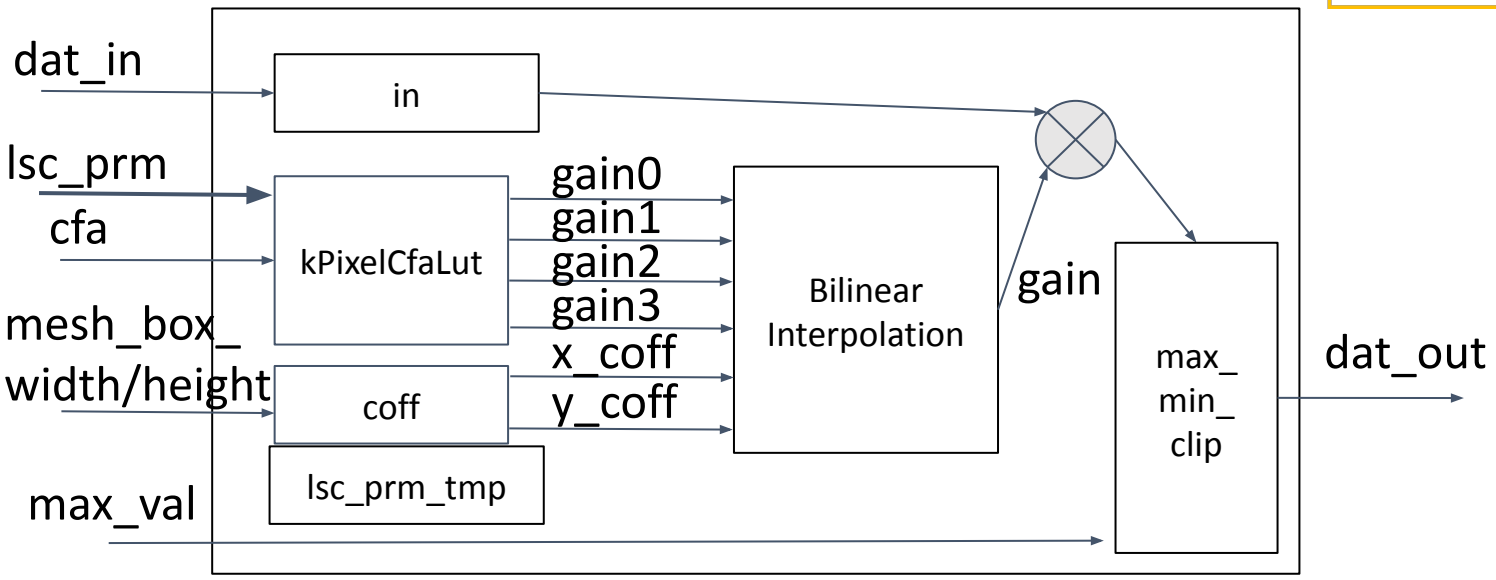

```
# i = 94, data_hls = 1225
# i = 95, data_hls = 2026
# i = 96, data_hls = 1450
# i = 97, data_hls = 2323
# i = 98, data_hls = 1631
# i = 99, data_hls = 2838
# i = 100, data_hls = 1914
# Test2
# Test2_1
# Test2_2
# SCVerify intercepting C++ function 'HDRISP_IP::dpc::run' for RTL block 'HDRISP_IP_dpc'
#                       DUT instance '0x2aaabb36871c'
# Info: HW reset: TLS_rst active @ 0 s
# Info: HW reset: TLS_arst_n active @ 0 s
# heightIn = 512, widthIn = 768
# Test3
# ====Algorithm.h is the same as Algorithm.cpp!====
# ====HLS_C is the same as Algorithm.cpp!====
# Info: Execution of user-supplied C++ testbench 'main()' has completed with exit code = 0
#
# Info: Collecting data completed
#    captured 393216 values of dat_in
#    captured 1 values of widthIn
#    captured 1 values of heightIn
#    captured 1 values of dpc_prms_thres
#    captured 1 values of dpc_prms_mode
#    captured 394752 values of dat_out
# Info: scverify_top/user_tb: Simulation completed
#
# Checking results
# 'dat_out'
#    capture count       = 394752
#    comparison count    = 394752
#    ignore count        = 0
#    error count         = 0
#    stuck in dut fifo   = 0
#    stuck in golden fifo = 0
#
# Info: scverify_top/user_tb: Simulation PASSED @ 3993766 ns
# ** Note: (vsim-6574) SystemC simulation stopped by user.
# 1
#
```

©BOLEDU

Lens Shading Correction

- Function:Correcting local deviation pixels caused by lens shading.
- Architecture:





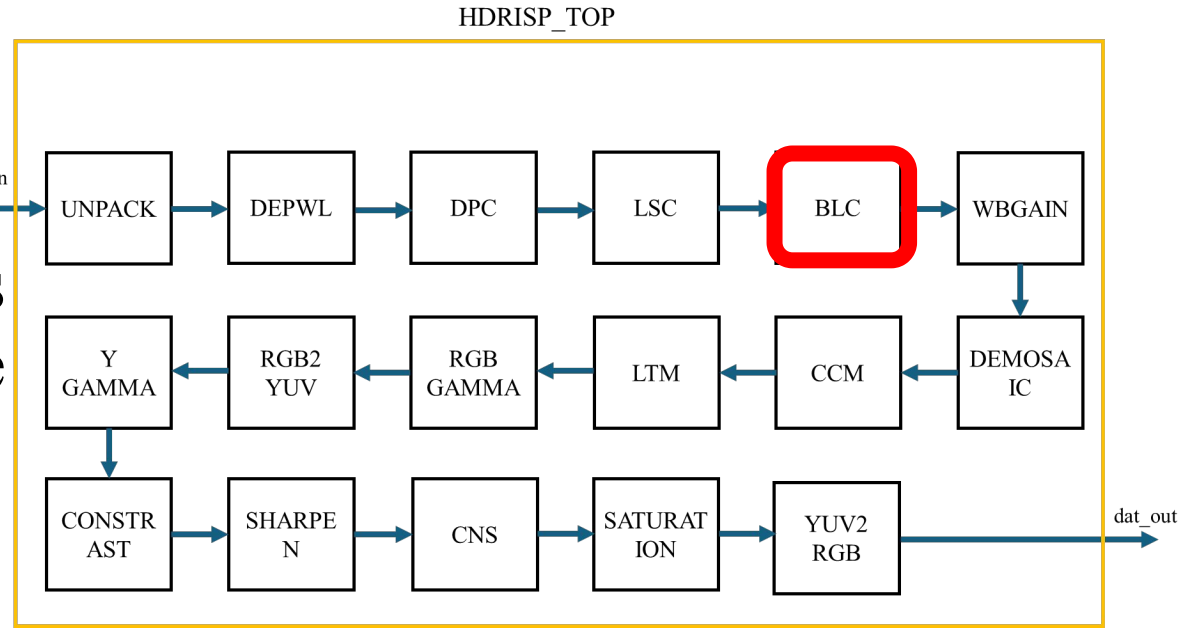
Transcript

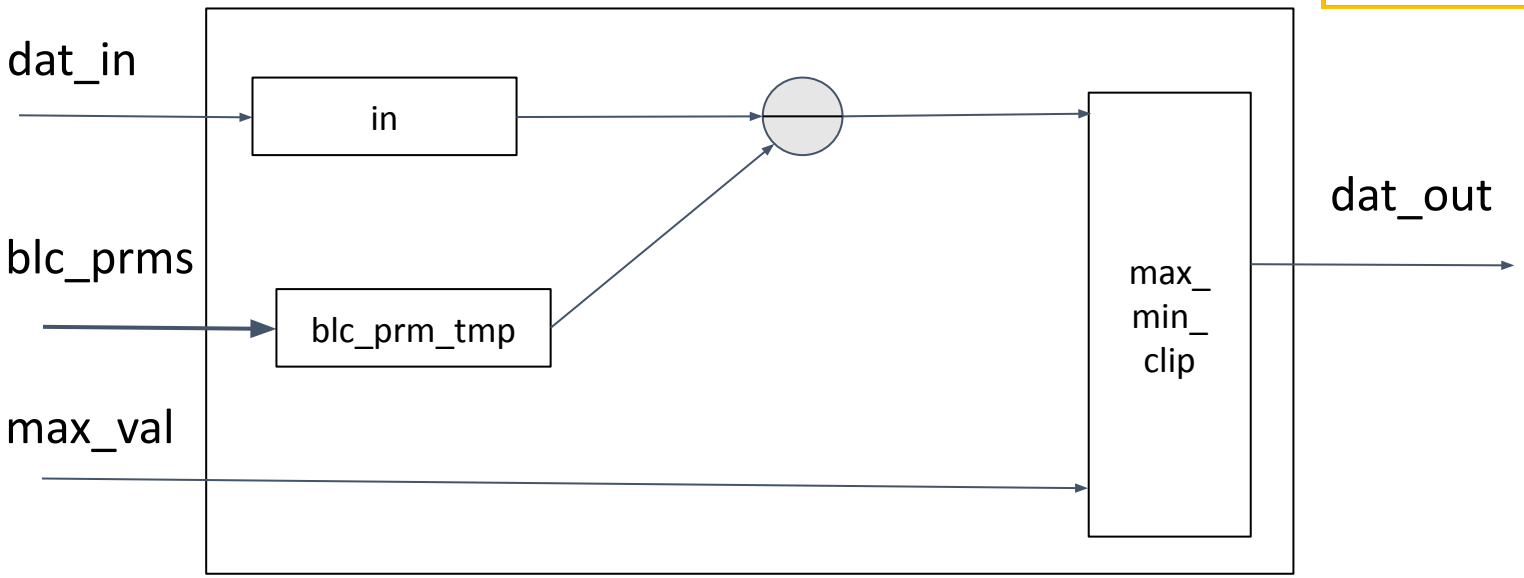
```
# i = 100, data_hls = 0
# Test2
# Test2_1
# Test2_2
# SCVerify intercepting C++ function 'HDRISP_IP::lsc::run' for RTL block 'HDRISP_IP_lsc'
#                       DUT instance '0x2aaabb2b564b'
# Info: HW reset: TLS_rst active @ 0 s
# Info: HW reset: TLS_arst_n active @ 0 s
# heightIn = 512, widthIn = 768
# Test3
# ====Algorithm.h is the same as Algorithm.cpp!====
# ====HLS_C is the same as Algorithm.cpp!====
# Info: Execution of user-supplied C++ testbench 'main()' has completed with exit code = 0
#
# Info: Collecting data completed
#    captured 393216 values of dat_in
#    captured 1 values of widthIn
#    captured 1 values of heightIn
#    captured 1 values of cfa
#    captured 1 values of max_val
#    captured 1 values of mesh_box_width
#    captured 1 values of mesh_box_height
#    captured 1 values of lsc_prms_mesh_r
#    captured 1 values of lsc_prms_mesh_gr
#    captured 1 values of lsc_prms_mesh_gb
#    captured 1 values of lsc_prms_mesh_b
#    captured 393216 values of dat_out
# Info: scverify_top/user_tb: Simulation completed
#
# Checking results
# 'dat_out'
#    capture count       = 393216
#    comparison count    = 393216
#    ignore count        = 0
#    error count         = 0
#    stuck in dut fifo   = 0
#    stuck in golden fifo = 0
#
# Info: scverify_top/user_tb: Simulation PASSED @ 3962906 ns
# ** Note: (vsim-6574) SystemC simulation stopped by user.
# 1
#
```

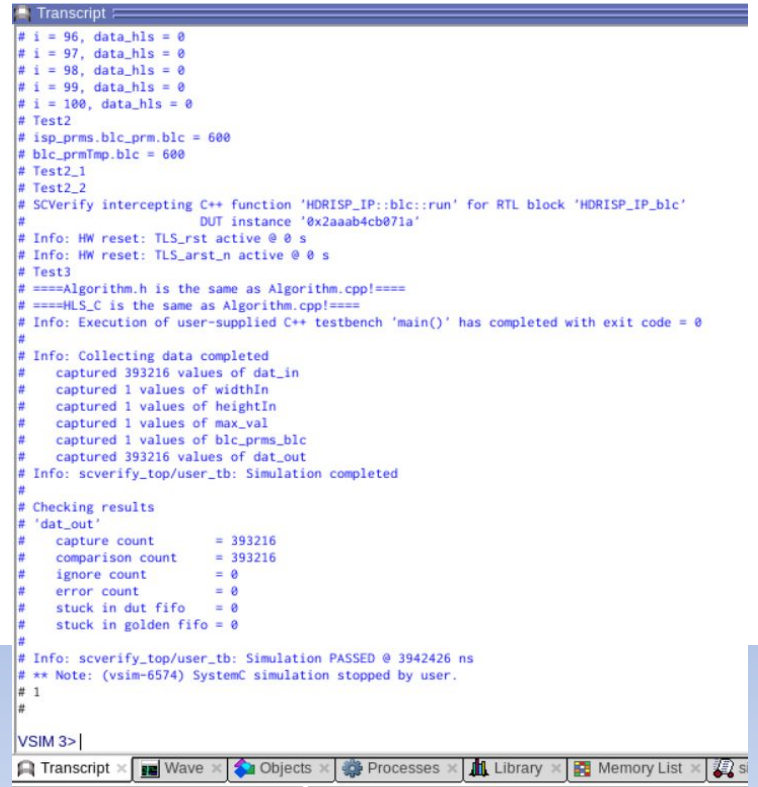
©BOLEDU

BLC

- Function:Correcting the pixel values by subtracting the constant blc value caused by dark current.
- Architecture:



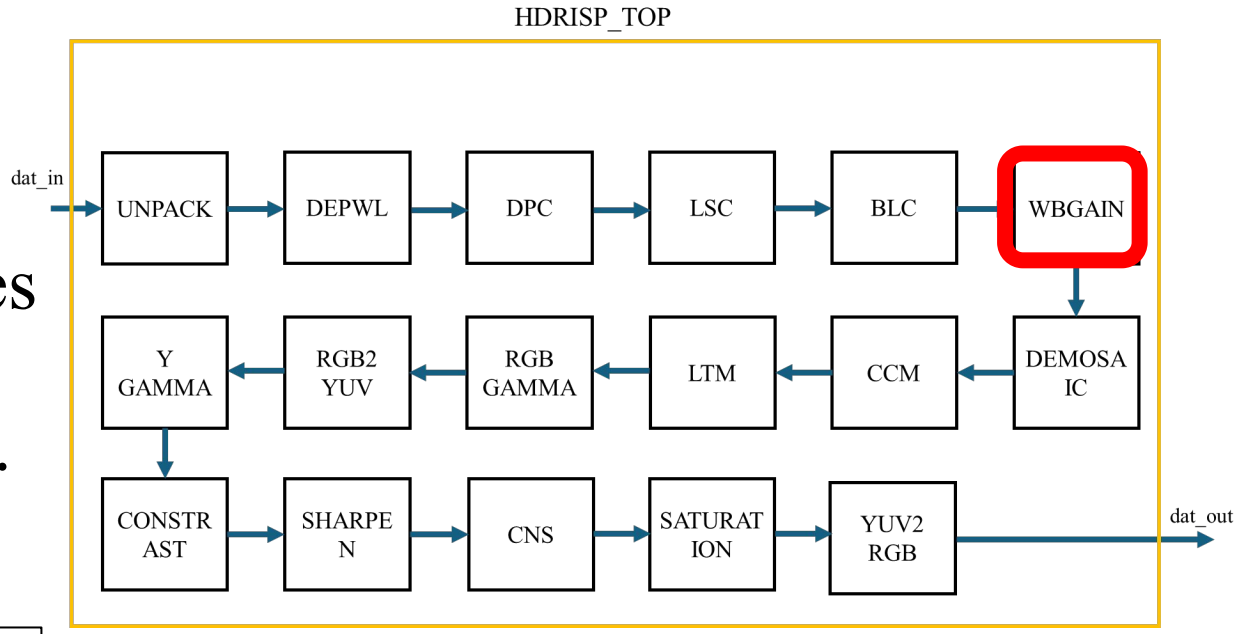


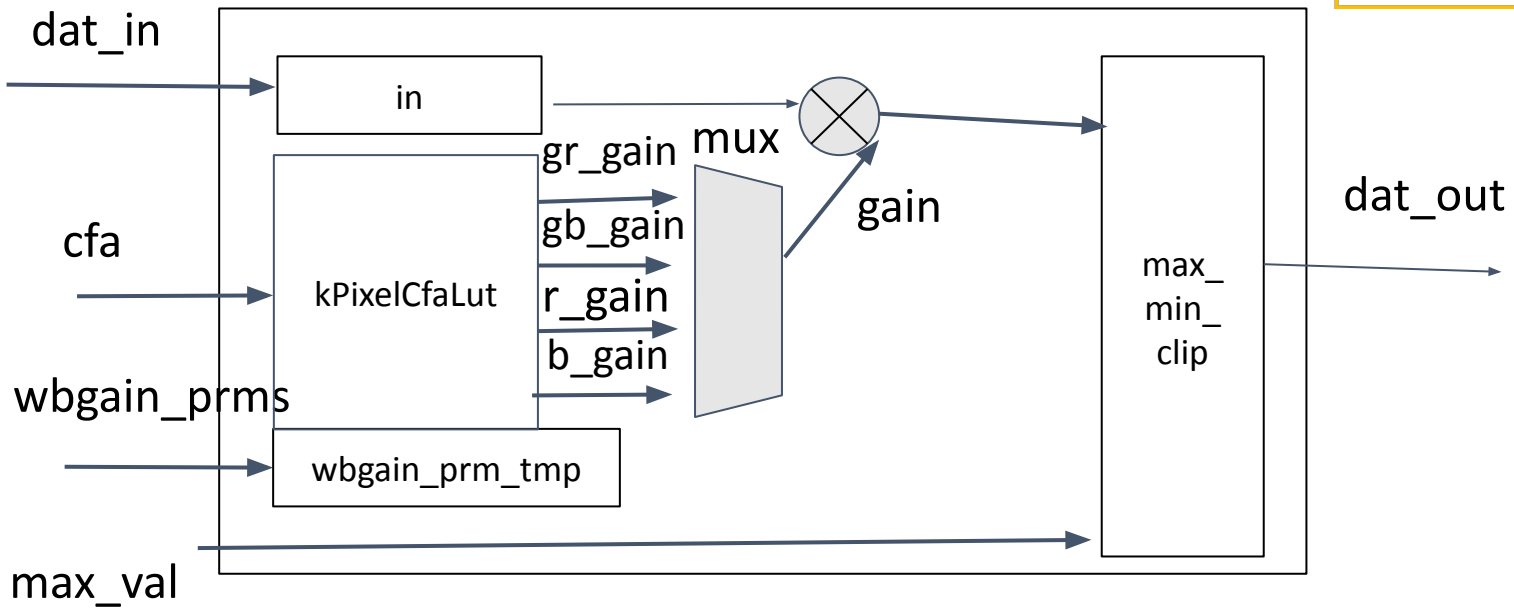


©BOLEDU

WBGAIN

- Function:Adjusting the pixel values by multiplying color gains calculated from the original pixels.
- Architecture:



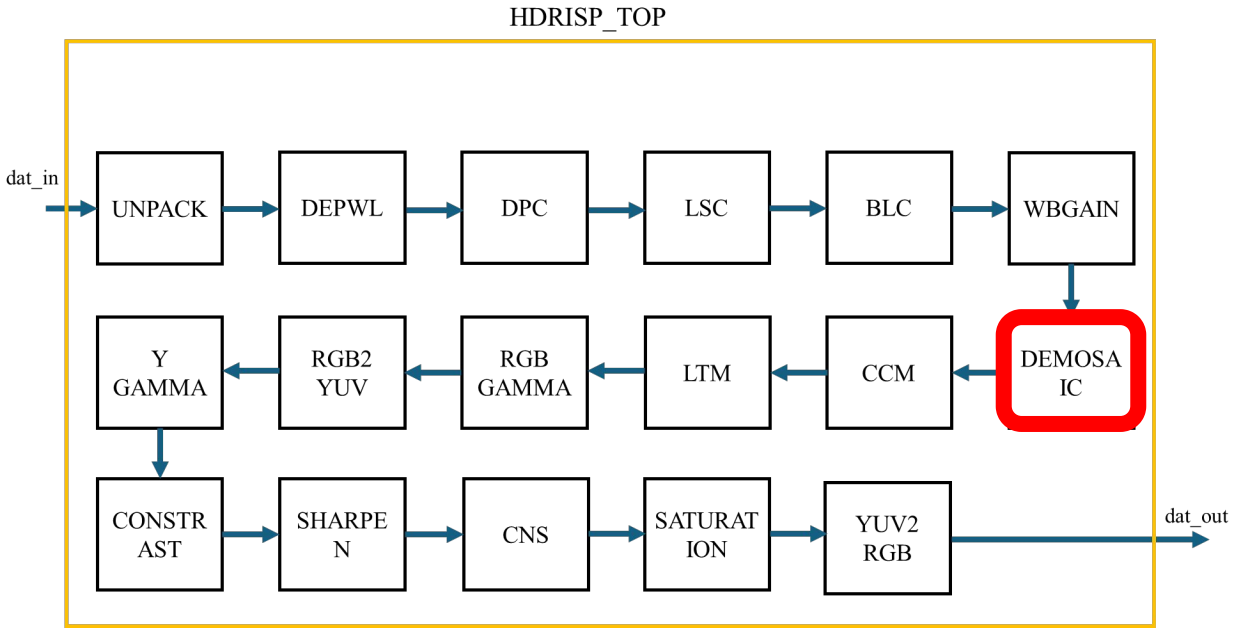



```
# i = 97, data_hls = 0
# i = 98, data_hls = 0
# i = 99, data_hls = 0
# i = 100, data_hls = 0
# Test2
# Test2_1
# Test2_2
# SCVerify intercepting C++ function 'HDRISP_IP::wbgain::run' for RTL block 'HDRISP_IP_wbgain'
#                       DUT instance '0x2aaab4d086b9'
# Info: HW reset: TLS_rst active @ 0 s
# Info: HW reset: TLS_arst_n active @ 0 s
# Test3
# ====HLS_C is the same as Algorithm.cpp!====
# Info: Execution of user-supplied C++ testbench 'main()' has completed with exit code = 0
#
# Info: Collecting data completed
#    captured 393216 values of dat_in
#    captured 1 values of widthIn
#    captured 1 values of heightIn
#    captured 1 values of max_val
#    captured 1 values of cfa
#    captured 1 values of wbgain_prms_d65_gain
#    captured 1 values of wbgain_prms_d50_gain
#    captured 1 values of wbgain_prms_f11_gain
#    captured 1 values of wbgain_prms_f12_gain
#    captured 393216 values of dat_out
# Info: scverify_top/user_tb: Simulation completed
#
# Checking results
# 'dat_out'
#    capture count       = 393216
#    comparison count    = 393216
#    ignore count        = 0
#    error count         = 0
#    stuck in dut fifo   = 0
#    stuck in golden fifo = 0
#
# Info: scverify_top/user_tb: Simulation PASSED @ 3942426 ns
# ** Note: (vsim-6574) SystemC simulation stopped by user.
# 1
#
```

©BOLEDU

DEMOSAIC

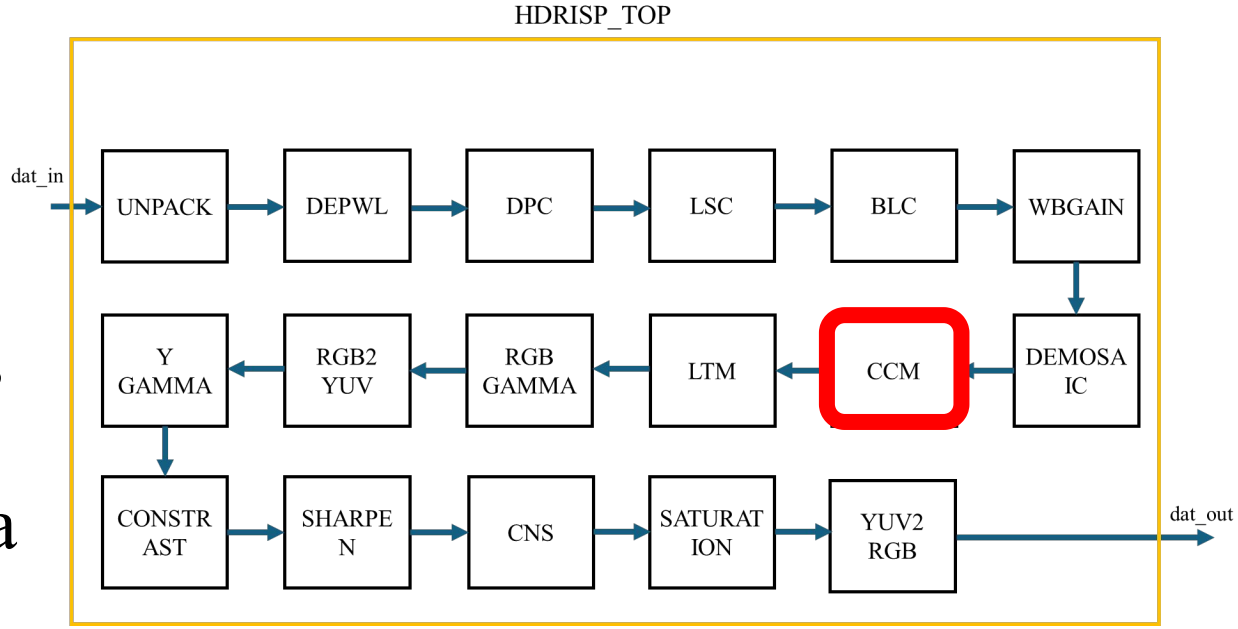
- Function:Converting a single channel Bayer RAW image into a three-color image.
- Architecture:

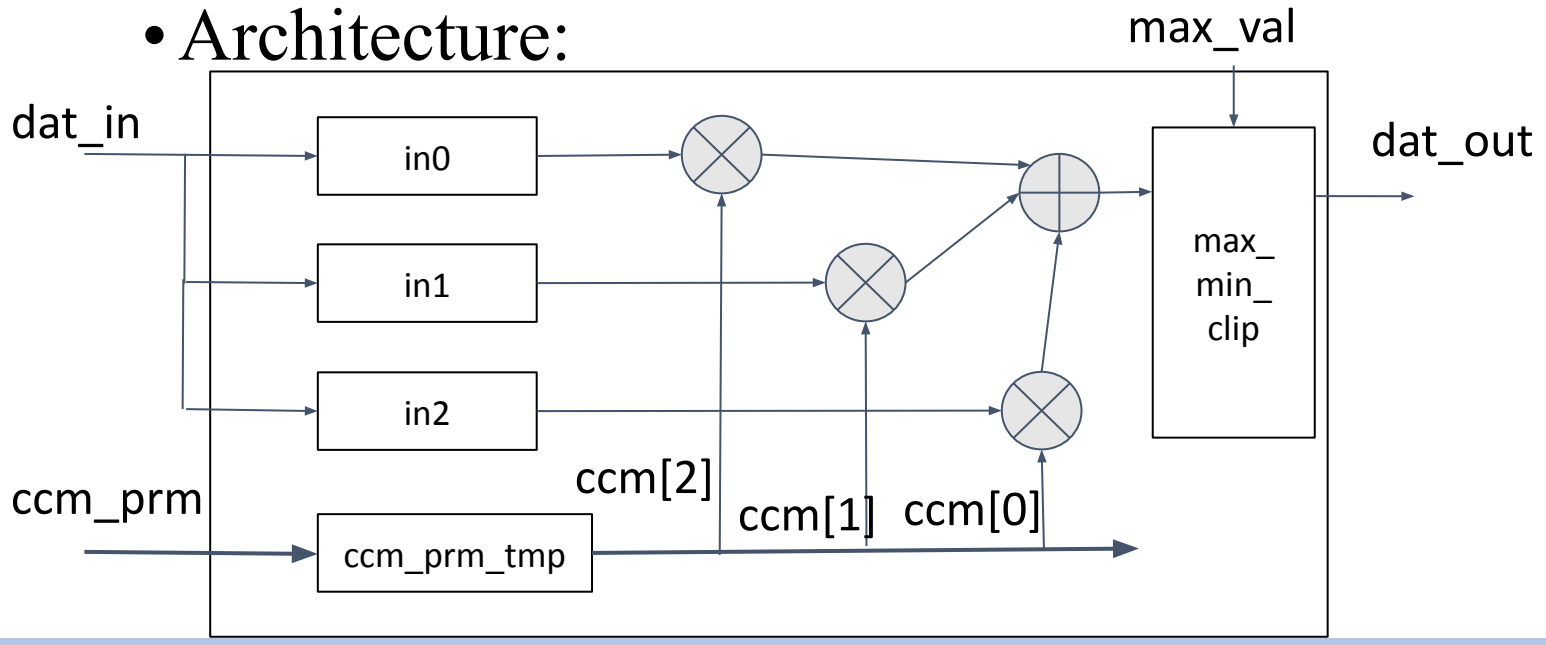


©BOLEDU

Color Correct Matrix

- Function:Using this matrix to correct the pixel values and results in a color picture that approaches the color seen by humans. This is a method to enhance white balance.
- Architecture:





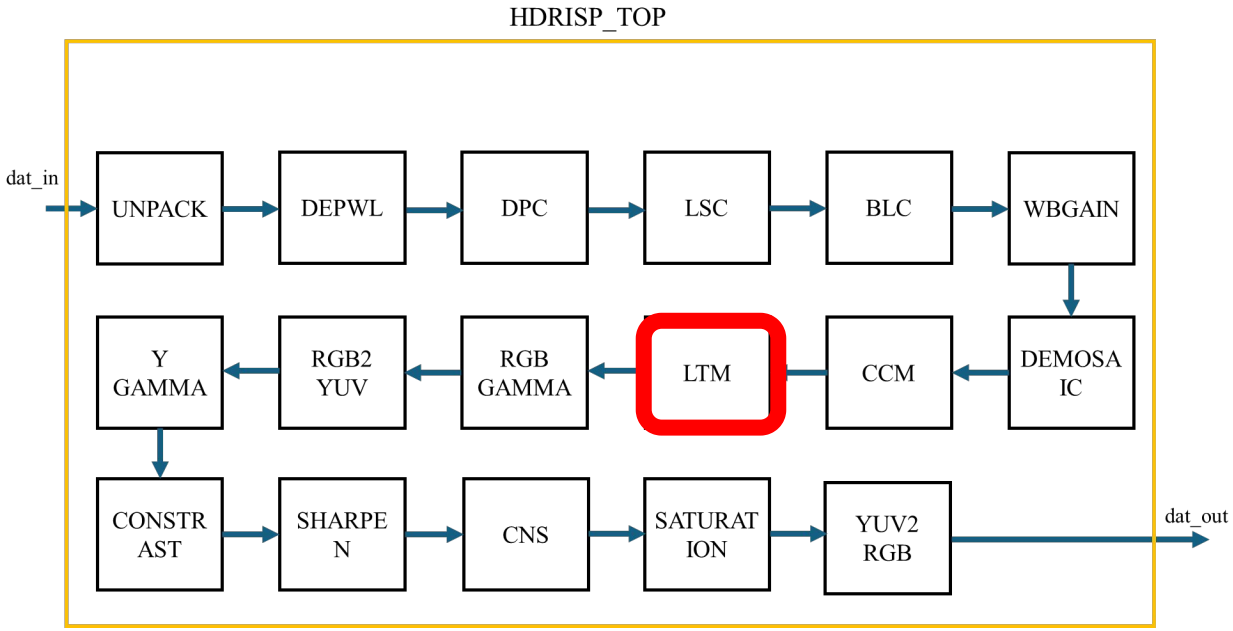
Transcript

```
VSIM 2> run -all
# 2024-06-12 16:15:02,011 INFO [default] Sensor Name: dsc
# 2024-06-12 16:15:02,011 INFO [default] Sensor CFA: "RGGB"
# 2024-06-12 16:15:02,012 INFO [default] Sensor DT: "RAW16"
# 2024-06-12 16:15:02,012 INFO [default] Sensor Resolution: 768*512
# 2024-06-12 16:15:02,012 INFO [default] parse exit
# Test1
# Test2
# Test2_1
# Test2_2
# SCVerify intercepting C++ function 'HDRISP_IP::ccm::run' for RTL block 'HDRISP_IP_ccm'
#                      DUT instance '0x2aaabb2996e7'
# Info: HW reset: TLS_rst active @ 0 s
# Info: HW reset: TLS_arst_n active @ 0 s
# heightIn = 512, widthIn = 768
# Test3
# ====Algorithm.h is the same as Algorithm.cpp!====
# ====HLS_C is the same as Algorithm.cpp!====
# Info: Execution of user-supplied C++ testbench 'main()' has completed with exit code = 0
#
# Info: Collecting data completed
#    captured 1179648 values of dat_in
#    captured 1 values of widthIn
#    captured 1 values of heightIn
#    captured 1 values of max_val
#    captured 1 values of ccm_prms_ccm
#    captured 1181949 values of dat_out
# Info: scverify_top/user_tb: Simulation completed
#
# Checking results
# 'dat_out'
#    capture count       = 1181949
#    comparison count    = 1181949
#    ignore count        = 0
#    error count         = 0
#    stuck in dut fifo   = 0
#    stuck in golden fifo = 0
#
# Info: scverify_top/user_tb: Simulation PASSED @ 19704346 ns
# ** Note: (vsim-6574) SystemC simulation stopped by user.
# 1
#
```

©BOLEDU

LTM

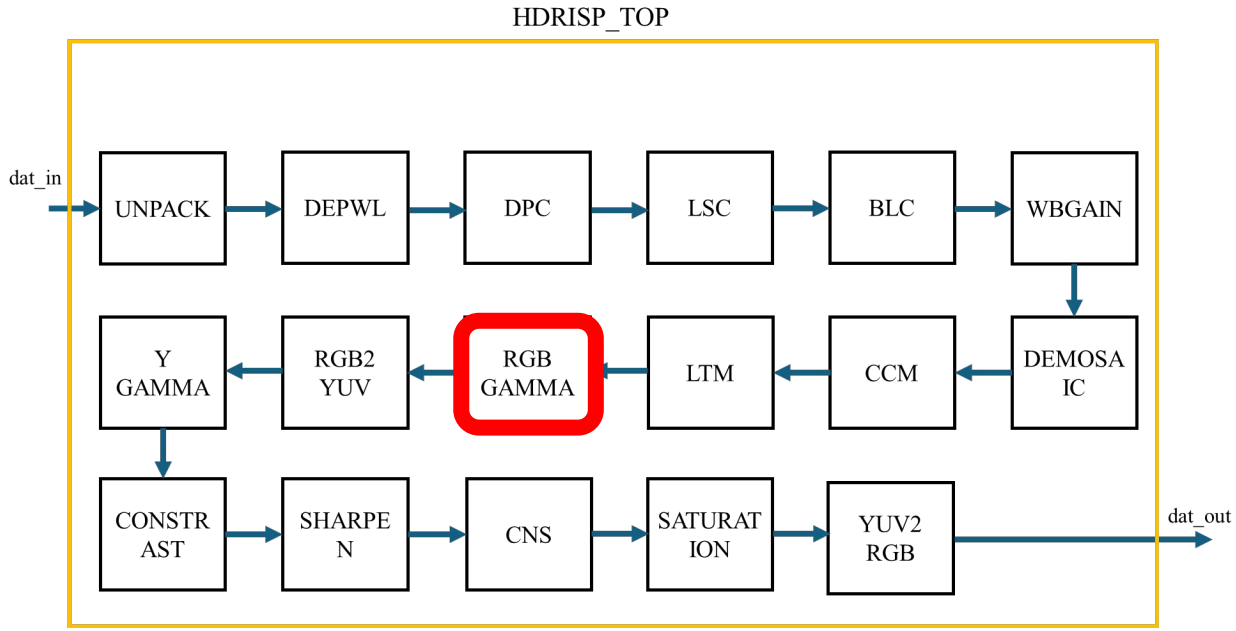
- Function:Local contrast enhancement based on brightness domain.
- Architecture:

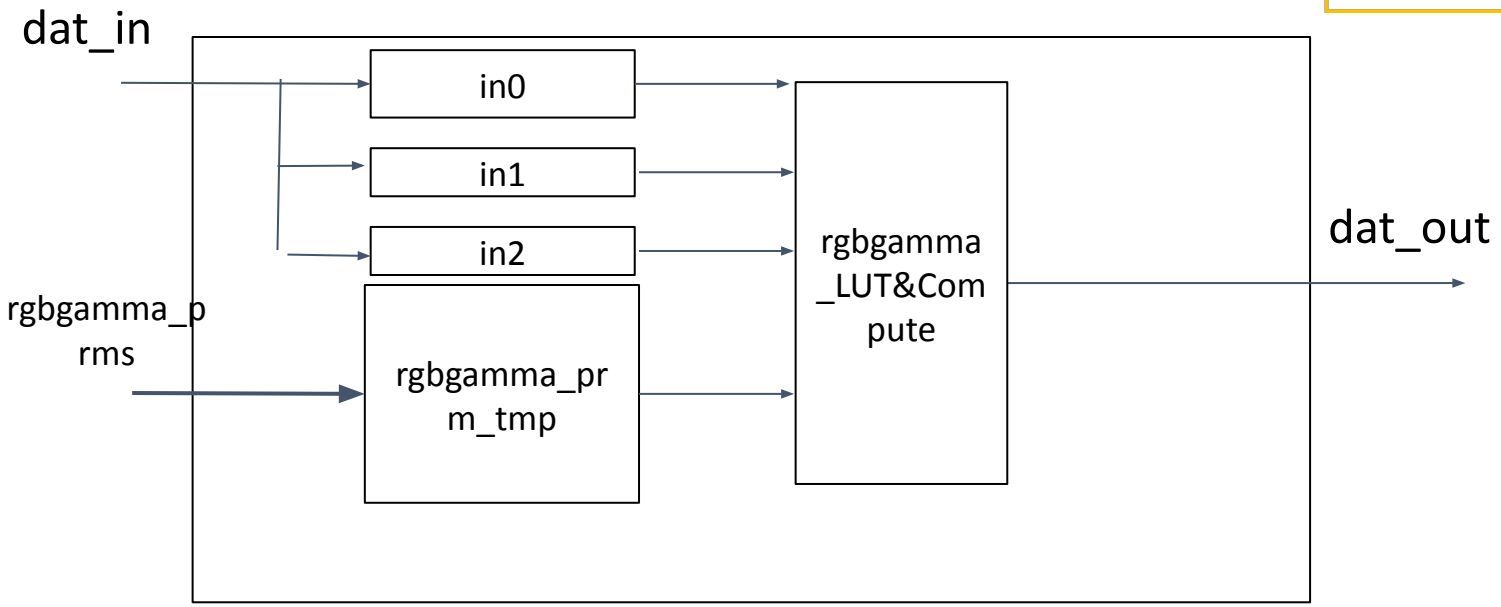


©BOLEDU

RGBGAMMA

- Function:Gamma correction is to perform nonlinear processing on the R/G/B channel image.
- Architecture:



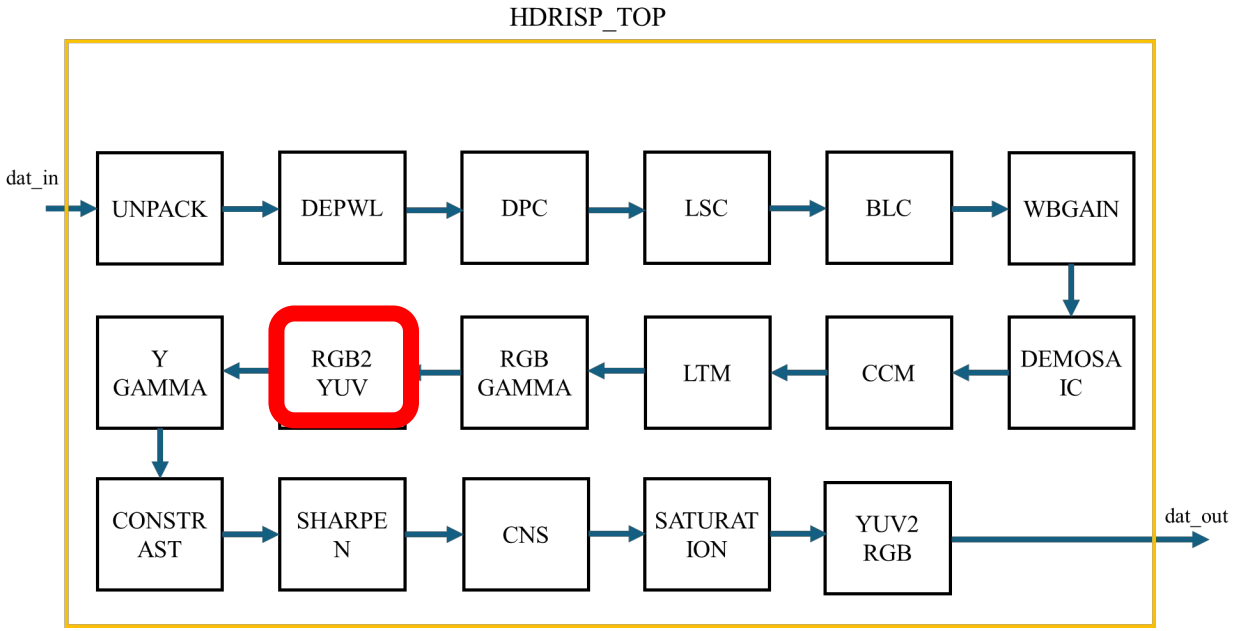


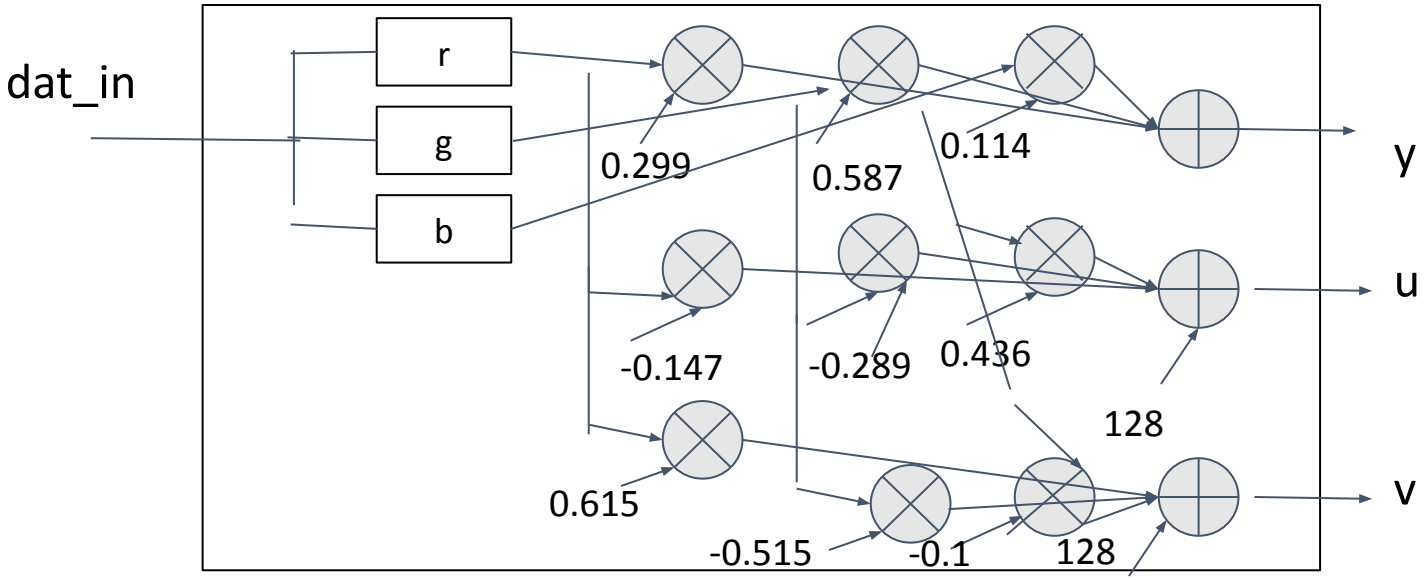

```
# rgbgamma_prmTmp.curve[i] = 0
# isp_prms.rgb_gamma.curve[i] = 0
# Test2_1
# Test2_2
# SCVerify intercepting C++ function 'HDRISP_IP::rgbgamma::run' for RTL
#                       DUT instance '0x2aaab4d1acae'
# Info: HW reset: TLS_rst active @ 0 s
# Info: HW reset: TLS_arst_n active @ 0 s
# temp_num = 10
# temp_in_bit = 1024
# step_coff = .009765625
# out_max = 255
# Test3
# ====Algorithm.h is the same as Algorithm.cpp!====
# ====HLS_C is the same as Algorithm.cpp!====
# Info: Execution of user-supplied C++ testbench 'main()' has completed
#
# Info: Collecting data completed
#    captured 3538944 values of dat_in
#    captured 1 values of widthIn
#    captured 1 values of heightIn
#    captured 1 values of rgbgamma_prms_nums
#    captured 1 values of rgbgamma_prms_in_bits
#    captured 1 values of rgbgamma_prms_out_bits
#    captured 1 values of rgbgamma_prms_curve
#    captured 1181949 values of dat_out
# Info: scverify_top/user_tb: Simulation completed
#
# Checking results
# 'dat_out'
#    capture count        = 1181949
#    comparison count     = 1181949
#    ignore count         = 0
#    error count          = 0
#    stuck in dut fifo    = 0
#    stuck in golden fifo = 0
#
# Info: scverify_top/user_tb: Simulation PASSED @ 19704346 ns
# ** Note: (vsim-6574) SystemC simulation stopped by user.
# 1
```

©BOLEDU

RGB2YUV

- Function:Convert RGB domain to YUV domain.
- Architecture:

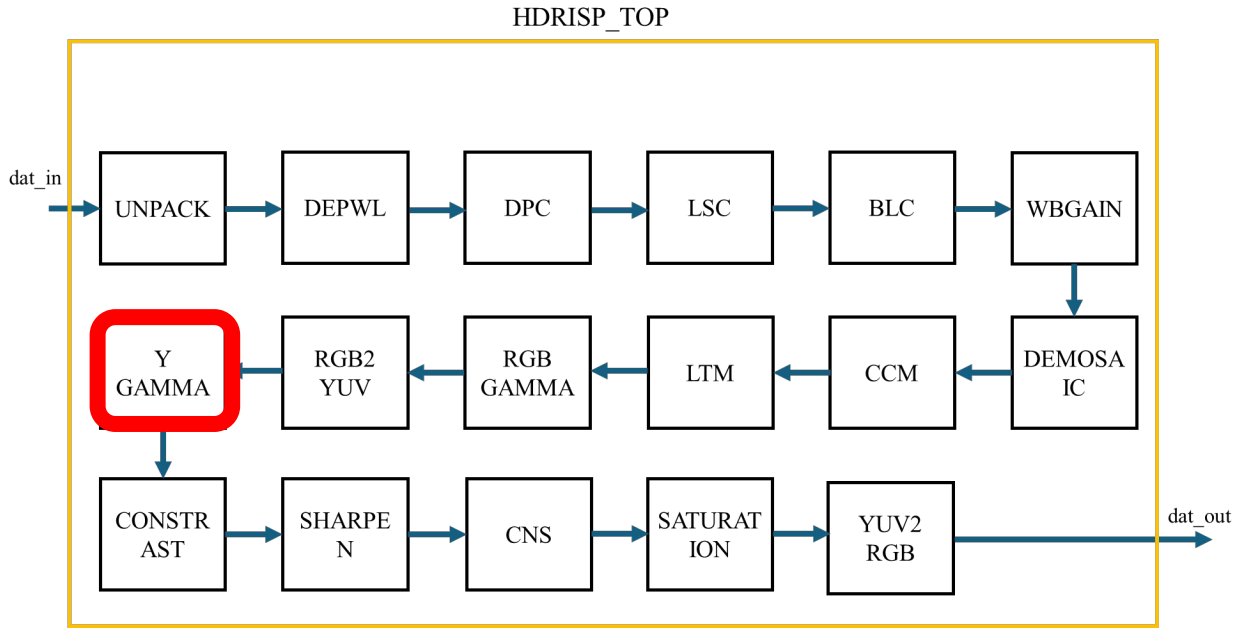


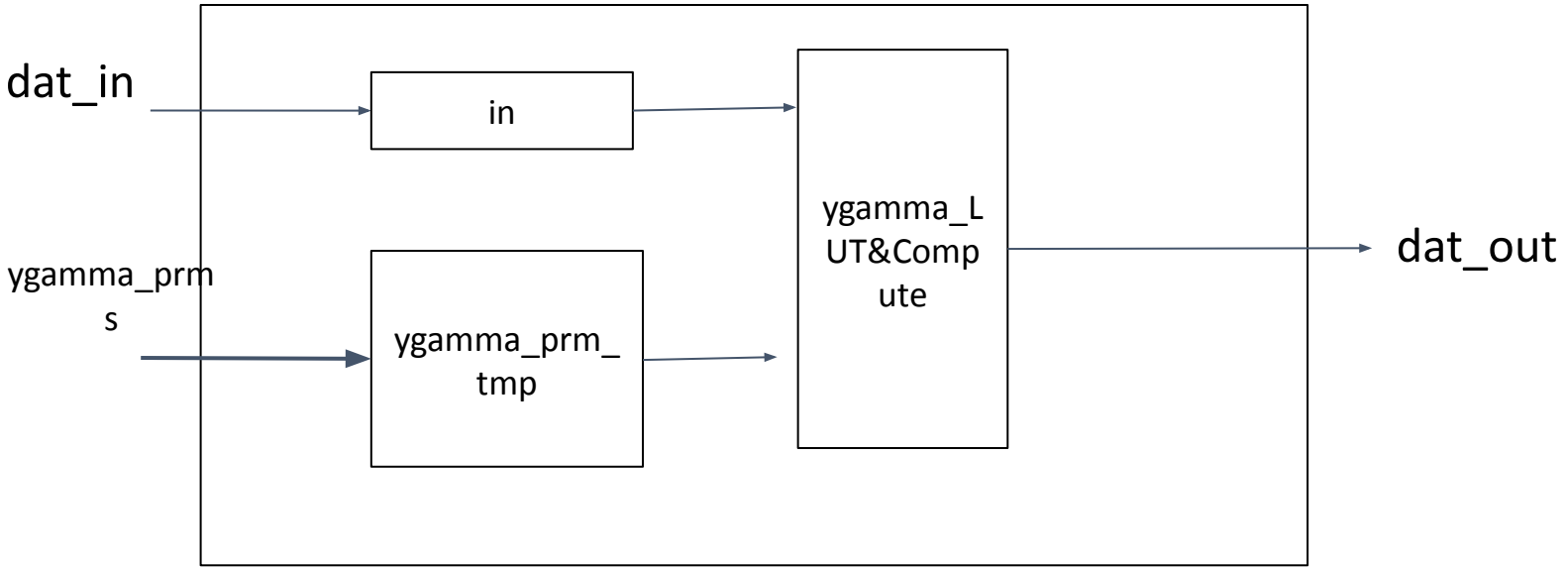


©BOLEDU

YGAMMA

- Function:Gamma correction is to perform nonlinear processing on the Y channel image.
- Architecture:

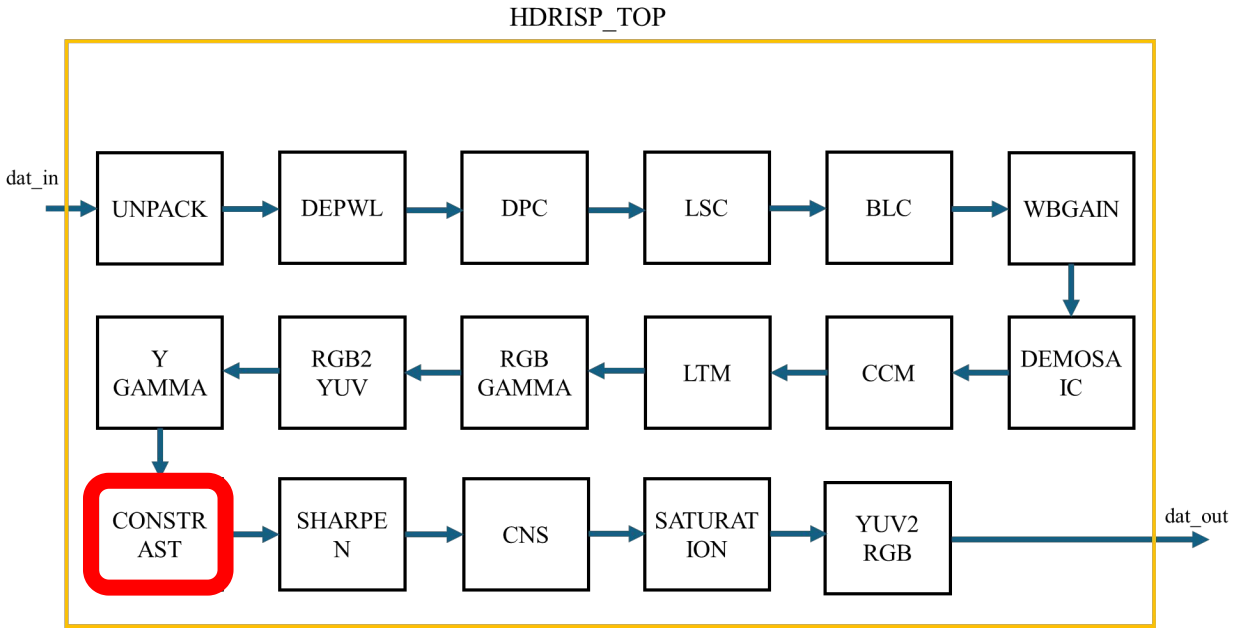


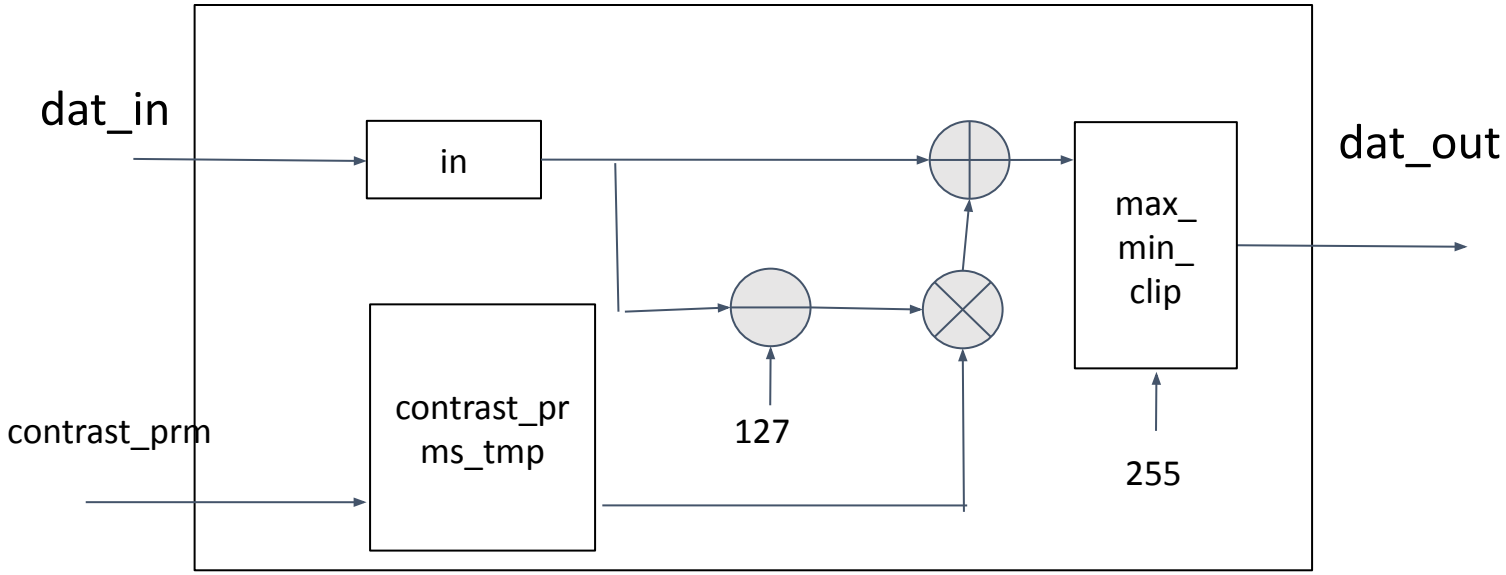


©BOLEDU

CONTRAST

- Function: Control the brightness to change the image.
- Architecture:





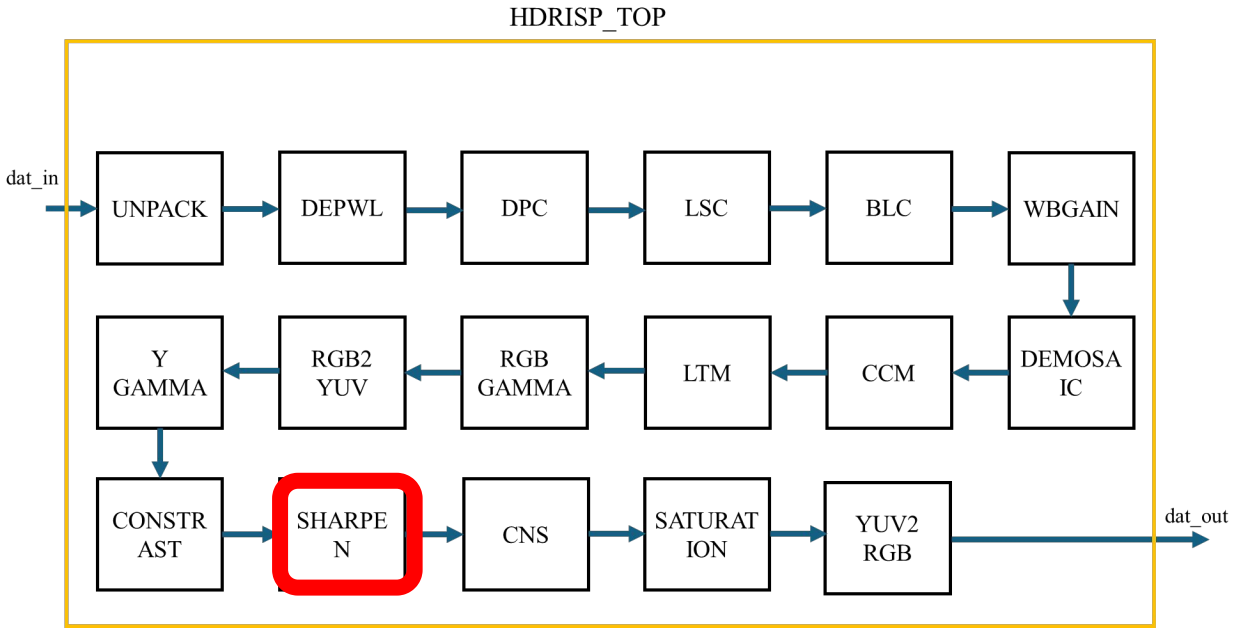

```
Transcript
# i = 95, data_hls = 32896
# i = 96, data_hls = 32896
# i = 97, data_hls = 32896
# i = 98, data_hls = 32896
# i = 99, data_hls = 32896
# i = 100, data_hls = 32896
# Test2
# isp_prms.contrast_prms.ratio = 0.1
# contrast_prmTmp.ratio = .09999942779541015625
# Test2_1
# Test2_2
# SCVerify intercepting C++ function 'HDRISP_IP::contrast::run' for RTL block 'HDRISP_IP_contrast'
#                      DUT instance '0x2aaab4cdbc4b'
# Info: HW reset: TLS_rst active @ 0 s
# Info: HW reset: TLS_arst_n active @ 0 s
# Test3
# ====Algorithm.h is the same as Algorithm.cpp!====
# ====HLS_C is the same as Algorithm.cpp!====
# Info: Execution of user-supplied C++ testbench 'main()' has completed with exit code = 0
#
# Info: Collecting data completed
#    captured 393216 values of dat_in
#    captured 1 values of widthIn
#    captured 1 values of heightIn
#    captured 1 values of contrast_prms_ratio
#    captured 393216 values of dat_out
# Info: scverify_top/user_tb: Simulation completed
#
# Checking results
# 'dat_out'
#    capture count       = 393216
#    comparison count    = 393216
#    ignore count        = 0
#    error count         = 0
#    stuck in dut fifo   = 0
#    stuck in golden fifo = 0
#
# Info: scverify_top/user_tb: Simulation PASSED @ 3942426 ns
# ** Note: (vsim-6574) SystemC simulation stopped by user.
# 1
#

VSIM 3>
```

©BOLEDU

SHARPEN

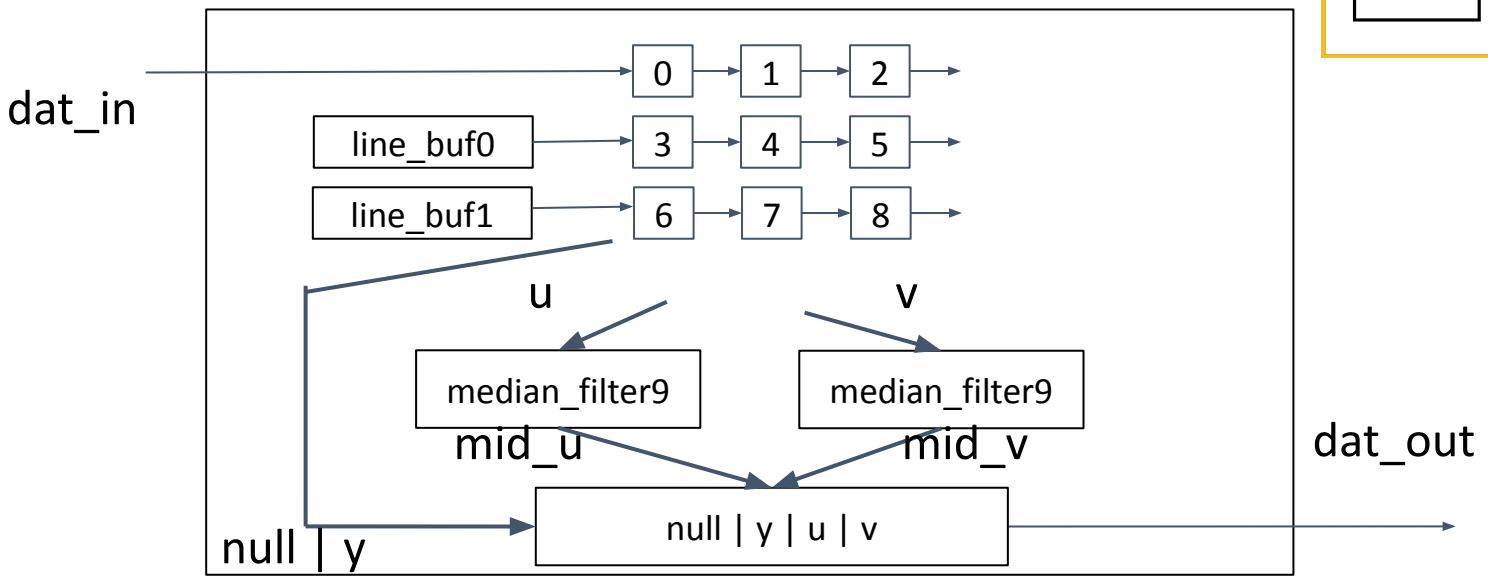
- Function:Enhance the edges of the image.
- Architecture:

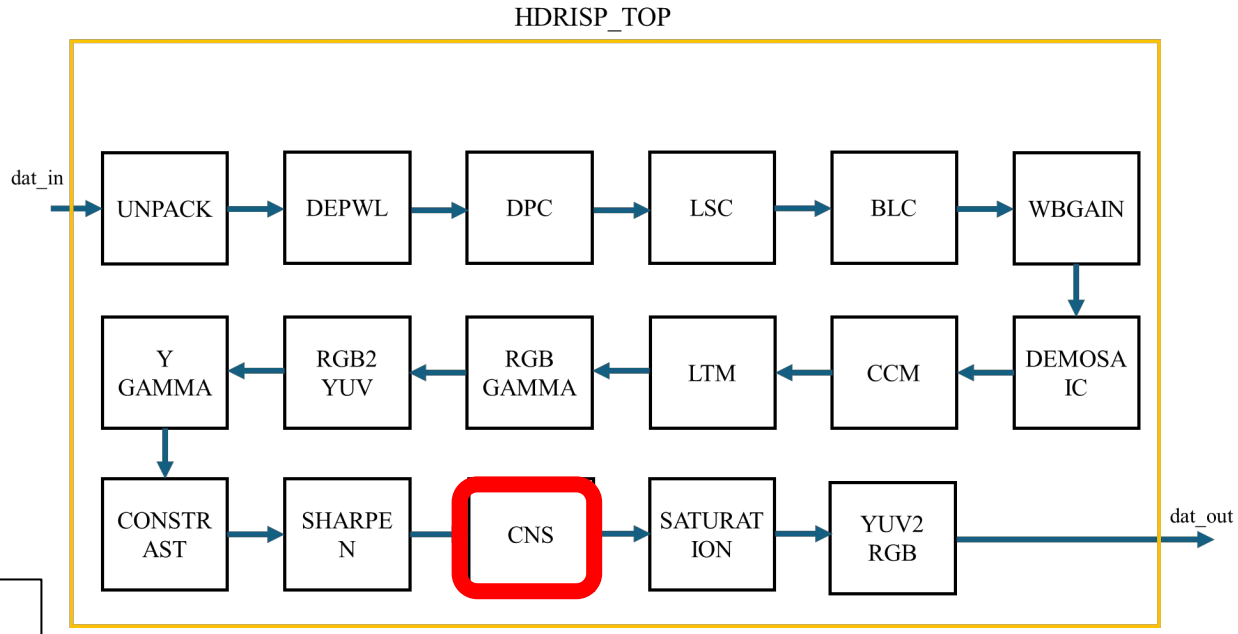


©BOLEDU

Chroma Noise Filter

- Function: Reduce noise related to chroma (U, V).
- Architecture:



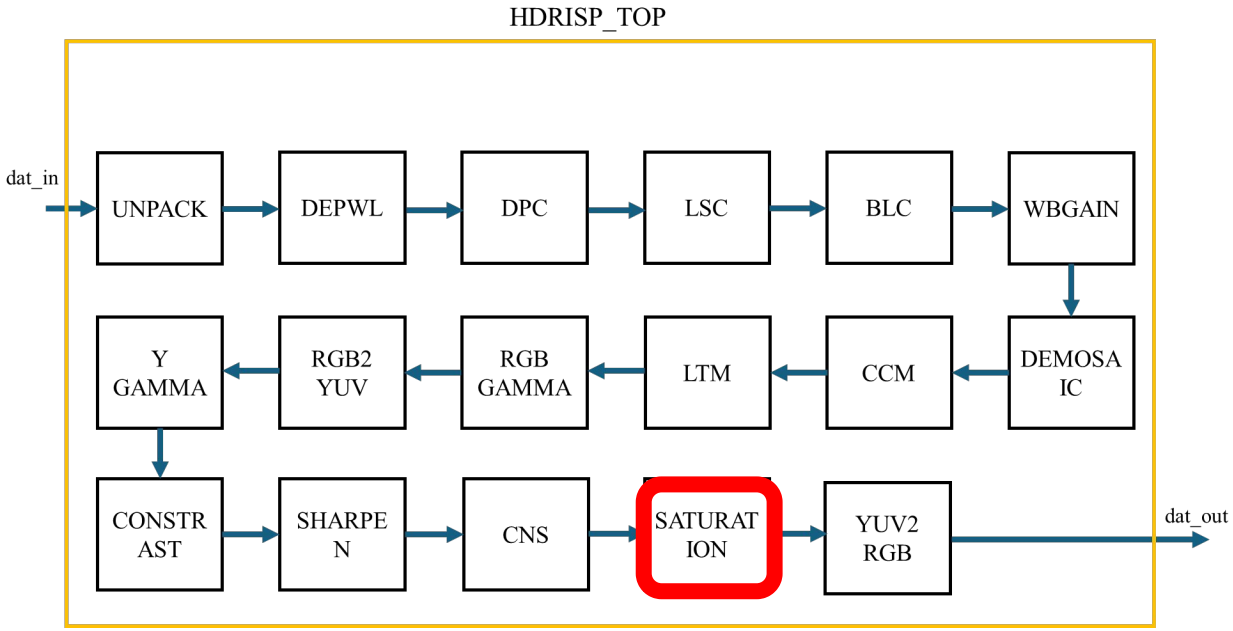



```
Transcript
# i = 92, data_hls = 32896
# i = 93, data_hls = 32896
# i = 94, data_hls = 32896
# i = 95, data_hls = 32896
# i = 96, data_hls = 32896
# i = 97, data_hls = 32896
# i = 98, data_hls = 32896
# i = 99, data_hls = 32896
# i = 100, data_hls = 32896
# Test2
# Test2_1
# Test2_2
# SCVerify intercepting C++ function 'HDRISP_IP::cns::run' for RTL block 'HDRISP_IP_cns'
#                      DUT instance '0x2aaabb2a3bbe'
# Info: HW reset: TLS_rst active @ 0 s
# Info: HW reset: TLS_arst_n active @ 0 s
# heightIn = 512, widthIn = 768
# Test3
# ====Algorithm.h is the same as Algorithm.cpp!====
# ====HLS_C is the same as Algorithm.cpp!====
# Info: Execution of user-supplied C++ testbench 'main()' has completed with exit code = 0
#
# Info: Collecting data completed
#    captured 393216 values of dat_in
#    captured 1 values of widthIn
#    captured 1 values of heightIn
#    captured 393216 values of dat_out
# Info: scverify_top/user_tb: Simulation completed
#
# Checking results
# 'dat_out'
#    capture count       = 393216
#    comparison count    = 393216
#    ignore count        = 0
#    error count         = 0
#    stuck in dut fifo   = 0
#    stuck in golden fifo = 0
#
# Info: scverify_top/user_tb: Simulation PASSED @ 3965506 ns
# ** Note: (vsim-6574) SystemC simulation stopped by user.
# 1
#
```

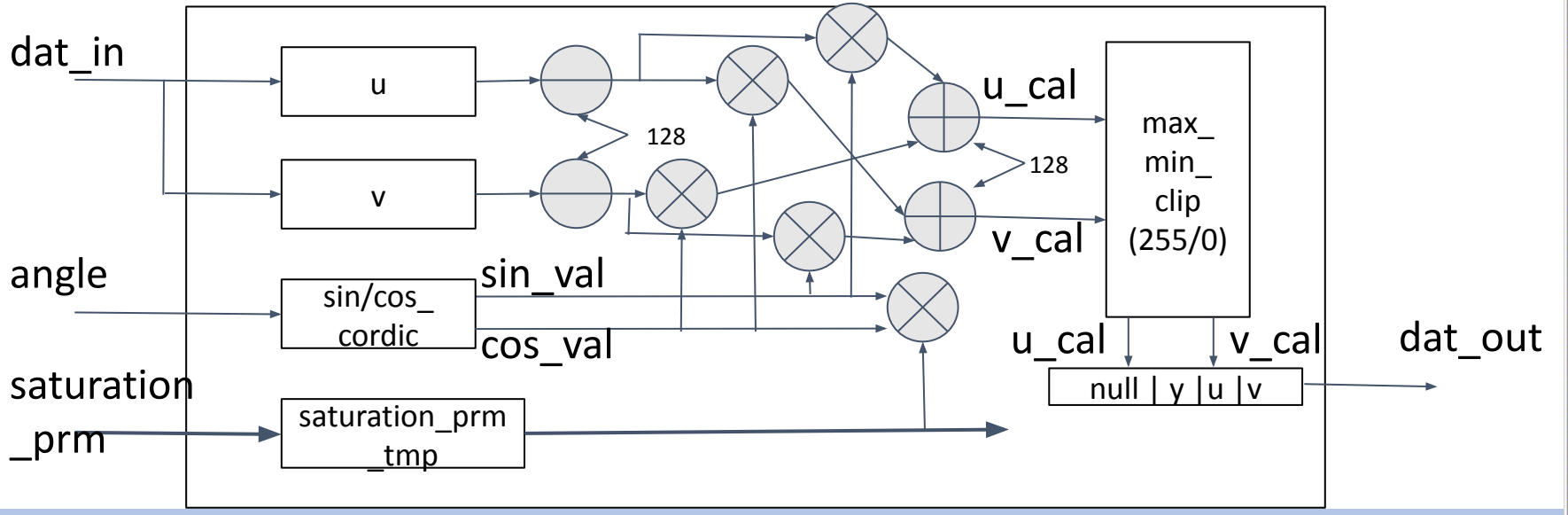
©BOLEDU

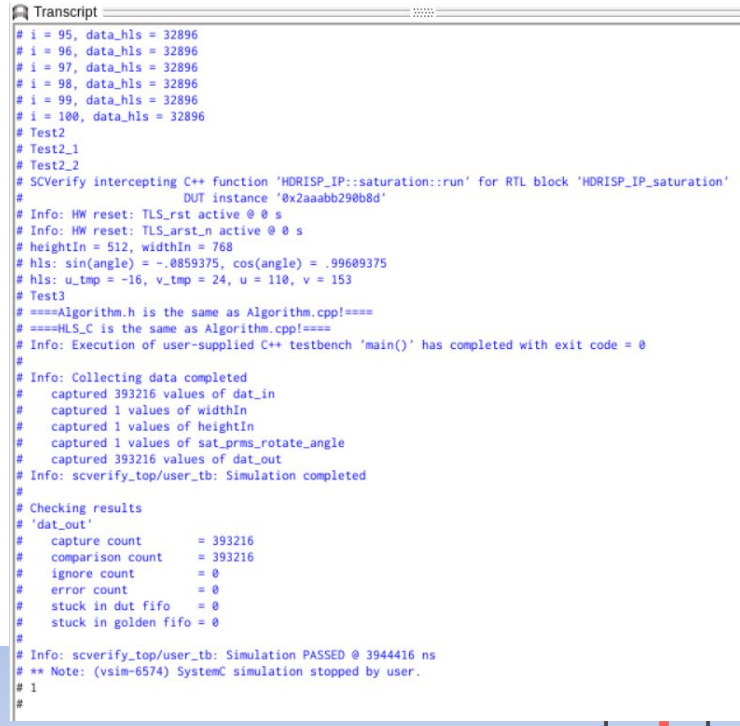
SATURATION

- Function: Adjust color saturation.



- Architecture:

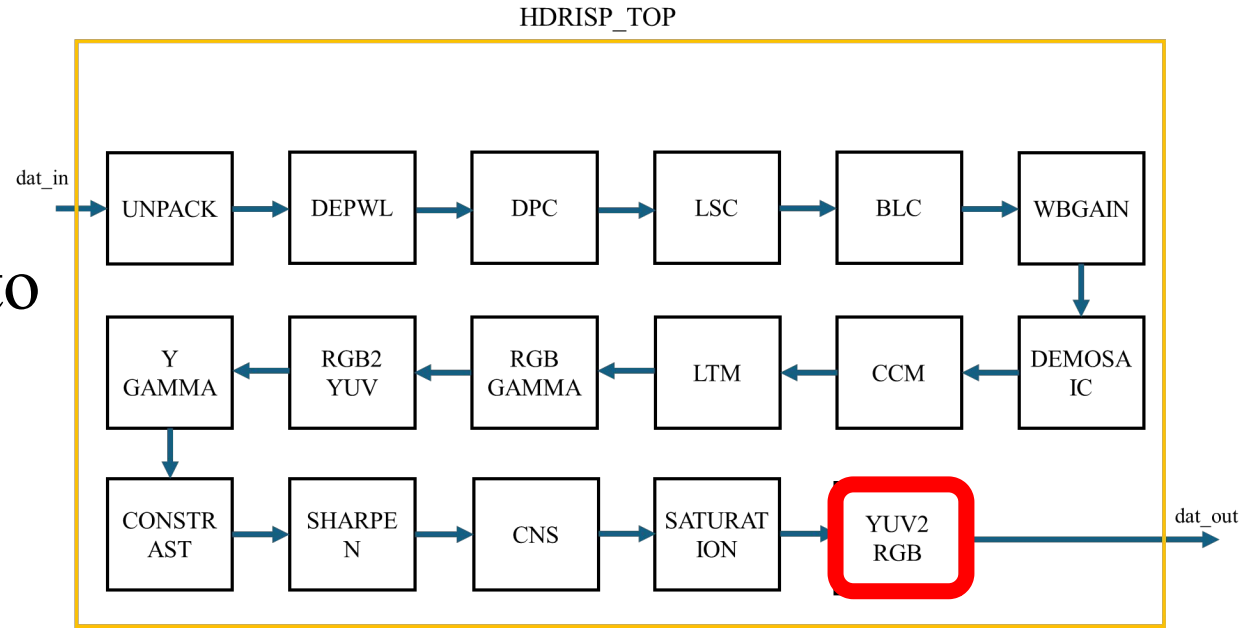


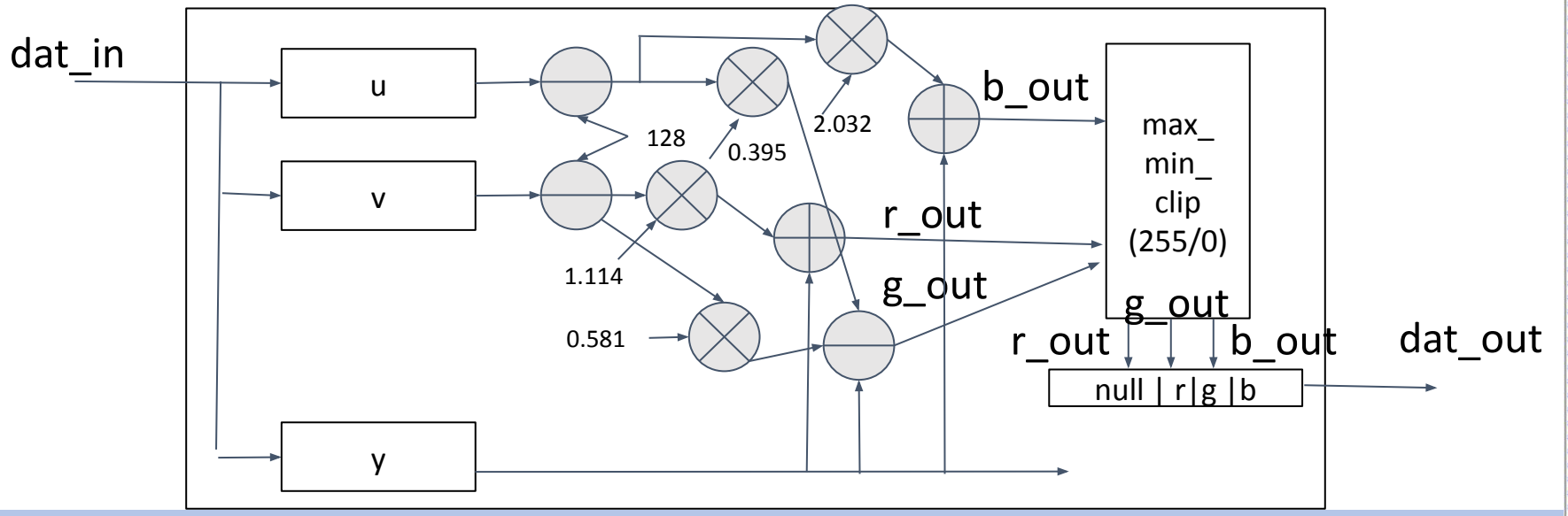


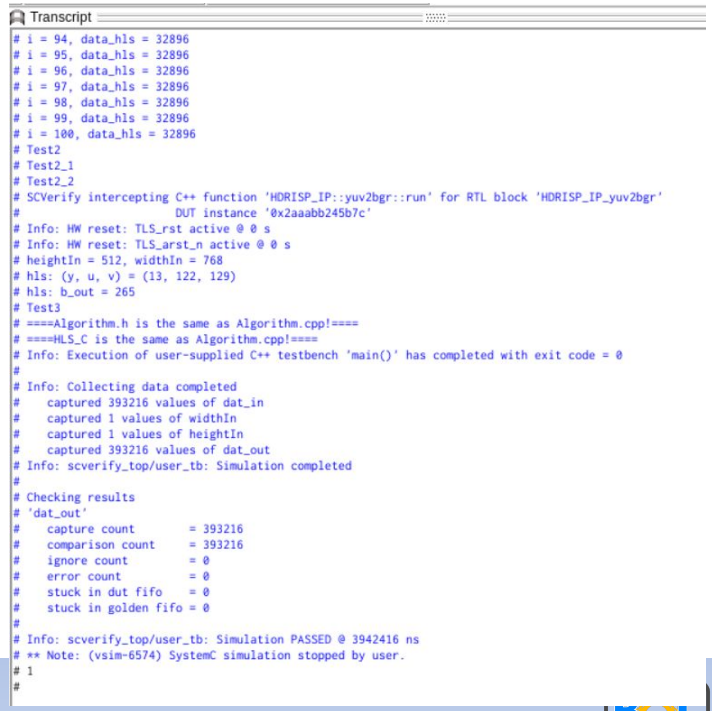
©BOLEDU

YUV2RGB

- Function: Transform YUV pixels to RGB pixels.
- Architecture:







©BOLEDU

Analysis - Insight & Finding

©BOLEDU

Improvement and solution effectiveness

- Streaming Interface
  - Stream the pixels one by one to the next module for real-time processing.
- Architecture Optimization
  - Line buffer + Window architecture for local filter calculation (i.e. 3*3, 5*5).
    - DPC, CNS
  - CNS: Median filter optimization - Finding median value in 3 steps.
    - Step 1: Median filter 3 * 3 - 3 rows of max/mid/min,
    - Step 2: Median filter 3 * 3 - max/mid/min in max/mid/min group,
    - Step 3: Median filter 3 * 1 - max_min/mid_mid/min_max.
  - Simplify HW resources in the module
    - Floating point operation => Fixed point operation
-

©BOLEDU

What you learn from the final project

- ISP Pipeline:
  - The composition of HDR ISP pipeline,
  - the reasons for the existence of some modules, and
  - the algorithms for various functions.
- Design Techniques:
  - Mapping from algorithms to architectures.
  - Faster way for sorting the median value.

©BOLEDU

Reference

©BOLEDU

List of Papers for Reference

[1]B. C. Huang, C. S. Fuh, “Image pipeline algorithms for standard mobile imaging architecture sensors”, 2005 18th IPPR Conference on Computer Vision, Graphics and Image Processing (CVGIP), pp. 1118-1125, 2005.
[2]S. H. Choi, et al. “A parallel camera image signal processor for SIMD architecture”, EURASIP Journal on Image and Video Processing, vol 2016, pp.1-14, 2016.
[3]Park H.S. (2016) Architectural Analysis of a Baseline ISP Pipeline. In: Kyung CM. (eds) Theory and Applications of Smart Cameras. KAIST Research Series. Springer, Dordrecht.

©BOLEDU

Open-source to use

- We use the following open-source projects:
  - Baseline: Image Signal Process For HDR CMOS Image Sensor
    - We follow the HDR ISP pipeline in this repository.
  - Catapult HLS Implementation reference: xkISP
    - Both the Vitis and Catapult HLS C modules are provided for reference, but the algorithm C codes are encrypted and provided in the form of executable binary file.

©BOLEDU

Thank you!

©BOLEDU

Roles & Task assignment

- 張傑閔:
- Identify algorithm C-source code and run Catapult C-sim
  - Run Catapult C-sim
- Kernel HLS implementation, Host implementation
  - Modules: LTM to YUV2RGB
- Individual Kernel FPGA validation/integration test
  - FSIC simulation

- 蔡宗穎:
- Identify algorithm C-source code and run Catapult C-sim
  - Identify test dataset
- Kernel HLS implementation, Host implementation
  - Modules: UNPACK to CCM
- Individual Kernel FPGA validation/integration test
  - FSIC validation

©BOLEDU

Check-point, Time Duration, and Deliverables

Date	Check-point	Time duration	Deliverables
5/5	Identify algorithm C-source code and run Catapult C-sim	1w	1. Test dataset 2. C-sim result
5/12	Kernel HLS implementation, Host implementation (I)	1w	1. # of modules implementation & unit test done
5/26	Kernel HLS implementation, Host implementation (II)	2w	1. # of modules implementation & unit test done
6/9	Individual Kernel FPGA validation/integration test	2w	1. FSIC simulation pass or not 2. FSIC validation pass or not
6/16	Kernel and Host Optimization	1w	1. Performance speedup or QoR

©BOLEDU

Final Project Proposal Guidelines

©BOLEDU

Final Project (Team)

- Refer to “Referenced Final Projects” for suggested projects
  - https://docs.google.com/spreadsheets/d/1FqAnTJP_vOm9G4UFEAlNB9KhNTy6tggH/edit?usp=drive_link&ouid=106716318998274820333&rtpof=true&sd=true
- Your research topics
- Requirement: End-to-end application acceleration
  - Scope at application level
  - Profiling to identify function to accelerate
  - SW application parallelization
  - Kernel optimization
  - Run on FPGA

Start thinking the Final Project Now

©BOLEDU

Purpose

The proposal report serves

1. Provide a framework to construct a project, from idea to execution
2. Define the project scope and evaluate its complexity
3. Basis for one-on-one discussion

©BOLEDU

Content of Final Project Proposal

- Project Title
- Team: Leader + Members
- Problem statement
- Project scope
- Project plan
- Reference

©BOLEDU

Problem Statement

- Context: what is the application the project is applied to?
- Issue: what is the problem to solve?
- Objective: what is the target to achieve? e.g. performance/area improvement, adding new functions ...

©BOLEDU

Project Scope

- Background Introduction
- System block diagram, and its operation flow
- Identify the area of work
- What FPGA platform to implement on, e.g. U50, PYNQ-Z2, KV260
- Target Specification – throughput, accuracy, speedup, area, or a design methodology proposed

©BOLEDU

Project Plan

- Workflow definition: Breakdown the project into a set of tasks, and describe the dependency among the tasks
- Assign the task to members / define the role of each member
- Identify the check-point, estimate the time duration, deliverables, and quality requirement

©BOLEDU

Workflow

- Identify algorithm C-source code - 1w
  - self-contained, no library function call
  - Identify test dataset
  - Partition host + kernel
- Run C-sim in Vitis environment Partition - 2w
  - run through dataset -> check correctness
- Kernel HLS implementation, Host implementation - 2w
  - define host/kernel communication, including debugging
  - If multiple kernels, allow validate separately.
  - Host program implement two modes for each kernels (C-code, or FPGA kernel)
- Individual Kernel FPGA validation/integration test - 1w
- Kernel and Host Optimization

©BOLEDU