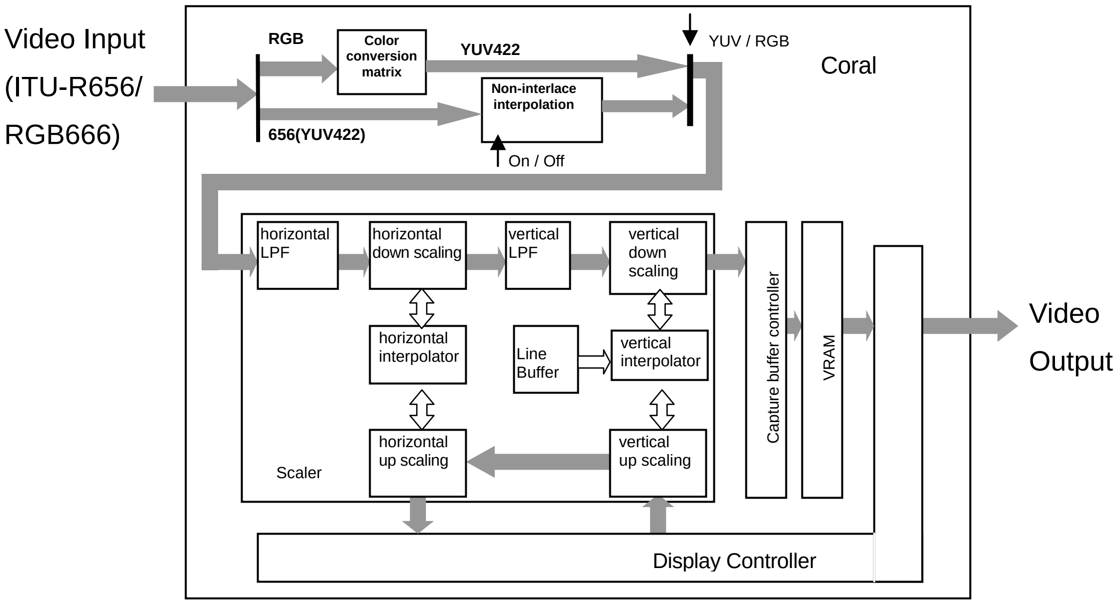
1. Video Capture Function

Some devices of Coral family have the video capture function.

The digital video stream of ITU RBT-656 or RGB666 format conformity is inputted. The each images are stored into the graphics memory, then read them to display on the L1-layer.

1.1 Image processing flow

The below is shown the image processing flow.



Note:

1.The Coral-LB is able to input <u>ITU RBT-656 stream only</u> and <u>doesn't support up-scaling</u> function.

2.The image data don't go through the up-scaling module when the scaling ratio is down.

- Non-interlace interpolation processing

When VI of a video capture mode register (VCM) is 0, an interlace screen is interpolated vertically using the data in the same field. A screen is doubled vertically. When VI is 1, it is not interpolated vertically.

- Horizontal low-pass filter processing

As a preprocessing when scaling down a picture horizontally, a low-pass filter can be covered horizontally. Regardless of scaling up and scaling down of a picture, ON/OFF is possible for a level low path filter (LPF).

The horizontal low-pass filter consists of FIR filters of five taps. A coefficient is specified in the following register.

CHLPF_Y	Horizontal LPF Luminance element and RGB element coefficient code
CHLPF_C	Horizontal LPF chrominance element coefficient code

The coefficient is specified by the coefficient code in two bits independently by luminance (Y) signal and chrominance (Cb and Cr) signals. The coefficient is a symmetric coefficient.

CHLPF_x	K0	K1	K2	K3	K4
00	0	0	1	0	0
01	0	1/4	2/4	1/4	0
10	0	3/16	10/16	3/16	0
11	3/32	8/32	10/32	10/32	3/32

Horizontal LPF becomes turning off (through) because of the setting of the coefficient code "00".

Note:

- In the case of Native RGB mode (NRGB=1), only a setup of CHLPF_Y code becomes effective.

- Down and Up scaling processing of horizontal direction

Please set bit15-00 of capture scale register (CSC) to do the down and up scaling processing of horizontal direction.

Horizontal direction is scaled down before writing in VRAM. Horizontal direction is scaled up after reading from VRAM.

The interpolation filter processing of luminance (Y) signal is done by cubic interpolation (Cubic Interpolate) method. The interpolation filter processing of chrominance (Cb and Cr) signal is done by BiLinear interpolation (BiLinear Interpolate) method. The interpolation filter processing of Native-RGB signal is done by cubic interpolation (Cubic Interpolate) method.

- Vertical low-pass filter processing

The low-pass filter can be put on the vertical direction as a preprocessing when the image is scaled down to the vertical direction. Vertical low-pass filter (LPF) can be set to turning on regardless of the scaling up or down of the vertical direction.

A vertical low-pass filter is composed of the FIR filter of three taps. The coefficient is specified by the following register.

CVLPF_Y	Vertical LPF Luminance element and RGB element coefficient code
CVLPF_C	Vertical LPF chrominance element coefficient code

The coefficient is specified by the coefficient code in two bits independently by luminance (Y) signal and chrominance (Cb and Cr) signals. The coefficient is a symmetric coefficient.

CVLPF_x	K0	K1	K2
00	0	1	0
01	1/4	2/4	1/4
10	3/16	10/16	3/16
11	Prohibition of setting		

Vertical LPF becomes turning off (through) because of the setting of the coefficient code "00".

Note:

- In the case of Native RGB mode (NRGB=1), only a setup of CVLPF_Y code becomes effective.

- Down and up scaling processing of Vertical direction

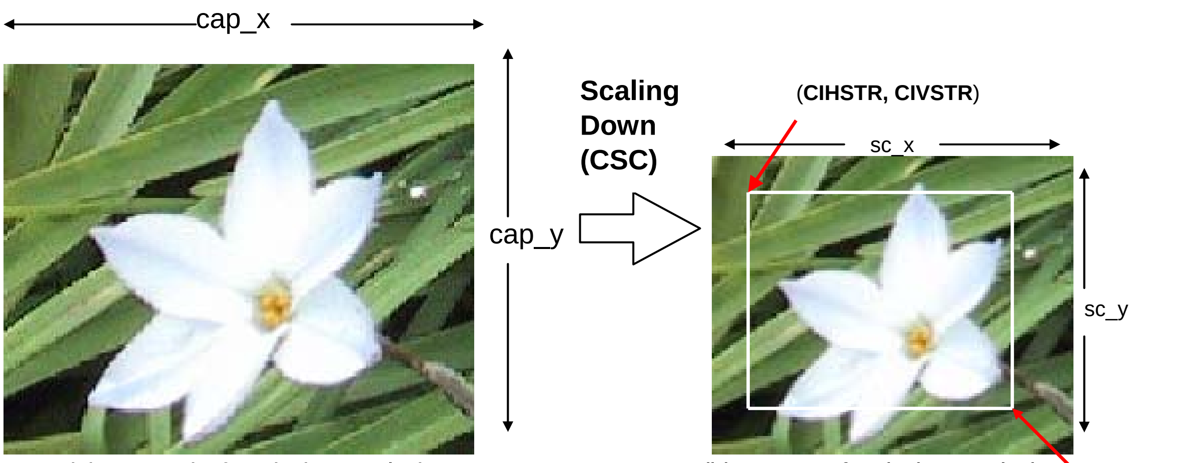
Please set bit31-16 of capture scale register (CSC) to do the down and up scaling processing in the vertical direction.

The vertical direction is scaled down before writing in VRAM. The vertical direction is scaled up after reading from VRAM.

The interpolation filter processing of luminance (Y) signal is done by cubic interpolation (Cubic Interpolate) method. The interpolation filter processing of chrominance (Cb and Cr) signal is done by BiLinear interpolation (BiLinear Interpolate) method. The interpolation filter processing of Native-RGB signal is done by cubic interpolation (Cubic Interpolate) method.

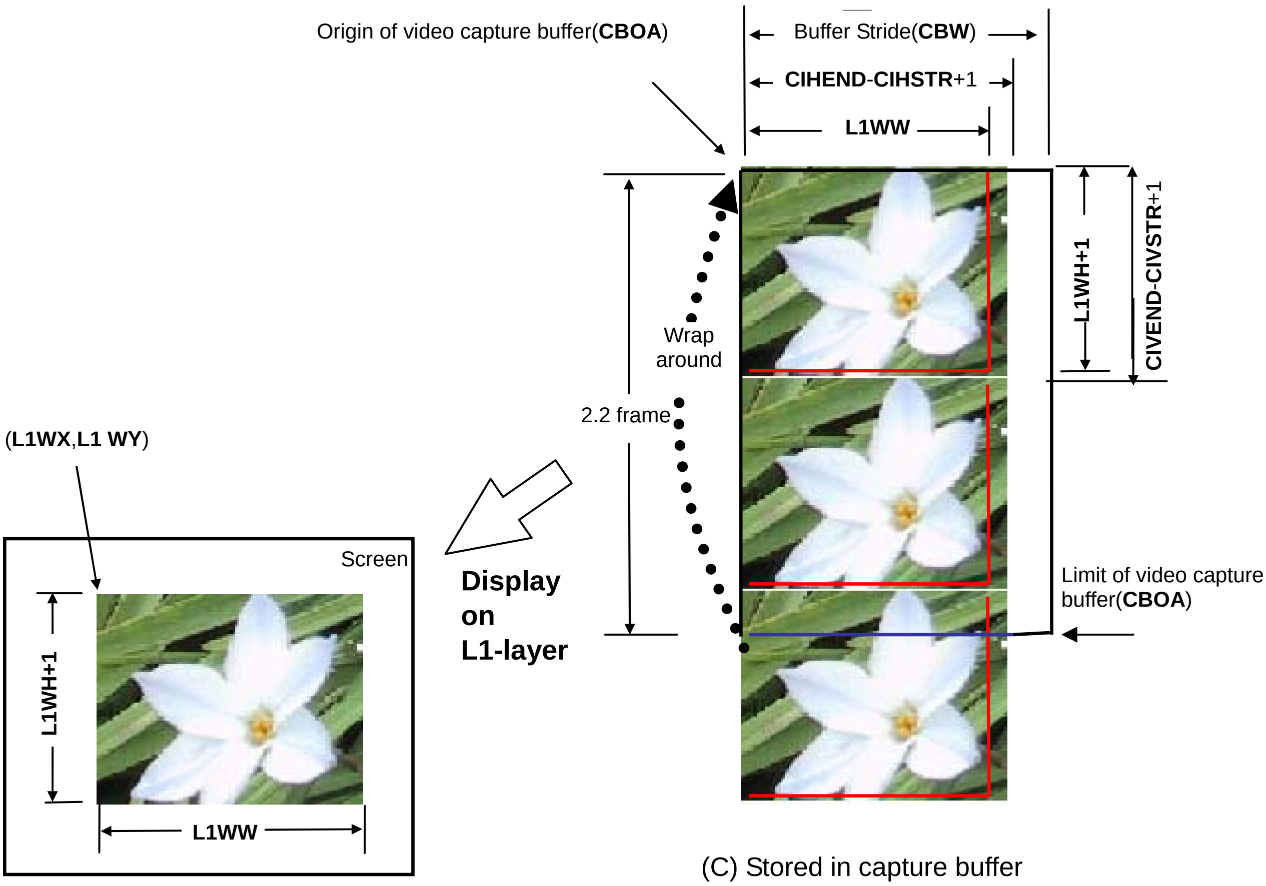
1.2 Various parameters

The following shows the various parameters from the original image to displayed image.



(a) Frame before being scaled
(Original Image)

(b) Frame after being scaled



(C) Stored in capture buffer

(D) Display image to screen

1) Processing with the Capture Scaler

   Compresses image to sc_x*sc_y size compared to the original image (cap_x*cap_y). Vertical and horizontal compression ratios are set with the CSC register's VSCI, VSCF, HSCI, HSCF. Refer to ***Scaling section*** in the ***Hardware manual*** for details regarding methods of calculating setting values..

   Also, the starting and ending points set by CIHSTR, CIVSTR, CIHEND and CIVEND are clipped at the frame after scaling and written to the buffer.

2) Processing with the Buffer Controller

   Stores images passed through the buffer region scaler set by the CBOA/CBLA/CBW registers. The buffer is controlled using the ring buffer method.

3) Processing with the Display Controller

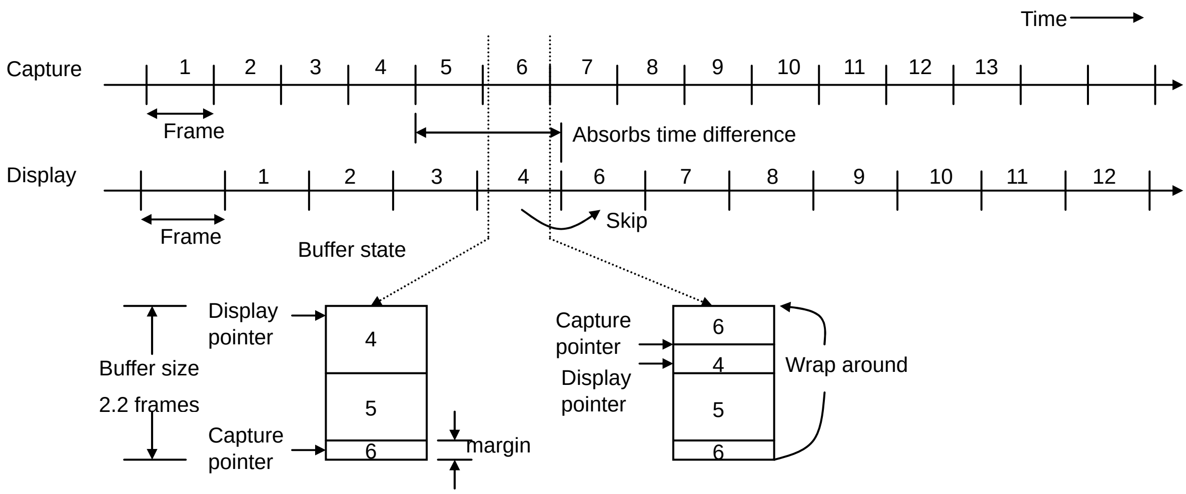
   The display controller displays the data stored in the capture buffer in the L1 layer. The display controller always displays the latest captured frames.

   In the WEAVE mode, the display controller generates frames from the odd and even fields for display.

1.3 Synchronization of frames

The capture buffer controller writes the sequentially captured image to the buffer. On the other hand, the display controller reads the captured-latest frames for display. The result is that frames can be skipped or the same frame can be displayed two or more times.

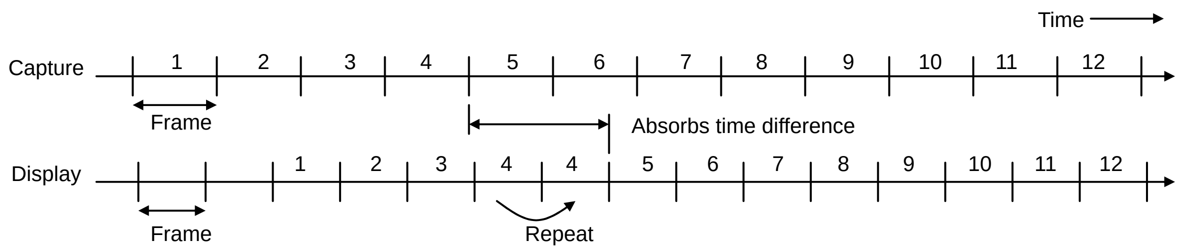
1) When capturing faster than display:



In the operation as shown in the above figure, when the “Display pointer” reaches the last of frame 4, if the “Capture pointer” reaches the last of frame 6, in the worst case, a frame can be skipped. This prevents the “Capture pointer” from passing the “Display pointer.”

To avoid passing the “Capture pointer,” it is necessary to allocate enough margin(2.2 frames) to the buffer.

2) When capturing more slowly than display:



1.4 Parameter setting values

The following determines each parameter.
The following are variable definitions.

(xs, ys): Image read starting point

(ws, hs): Original image size

(wd, hd): Size to display in window

1) Selecting non-interlacing

The non-interlacing mode is determined by the input image format and the display output size.

n: Raster count around the input image field.

$n \geq hd$: BOB and WEAVE conversion unnecessary

$n < hd \leq 2n$: BOB or WEAVE

When not using BOB and WEAVE, displays the odd field and even field alternately in the same position. However, in such cases, this is appropriate for displaying in comparatively small windows.

2) Setting compression rate

Refer ***Scaling section*** in the ***Hardware Manual*** to set compression rate.

rh = ws / wd

rv = hs/ hd

Compression rate integers are set to HSCI and VSCI. Fractions are set to HSCF and VSCF.

HSCI * 2048 + HSCF = floor (rh*2048+0.5)

VSCI * 2048 + VSCF = floor (rv*2048+0.5)

*floor calculates the floor function (discard) of the value .

3) Setting image range to store in the capture buffer

CIHSTR = floor(xs/rh+0.5)

CIVSTR = floor(ys/rv+0.5) for not WEAVE mode

= floor(ys/rv/2+0.5) for WEAVE mode

CIHEND = CIHSTR+wd-1

CIVEND = CIVSTR+hd-1 for not WEAVE mode

= CIVSTR+hd/2-1 for WEAVE mode

4) Setting capture buffer and W layer stride

CBW = WW = ceil(wd/32

*ceil calculates the ceiling function (round up) of the value .

5) Setting buffer size

CBLA = CBOA + ceil(K*hd)*CBW*64

K, here, shows the buffer size according to the frame size. This value is set to 2.5 from 2.2. K = 2.2 is a general value for use while K = 2.5 is a safer value.

6) Setting L1 layer parameter

L1WW = xd

L1WH = yd –1