



# JCTVC-D263 Parallel deblocking filter

Masaru Ikeda, Junichi Tanaka, Teruhiko Suzuki

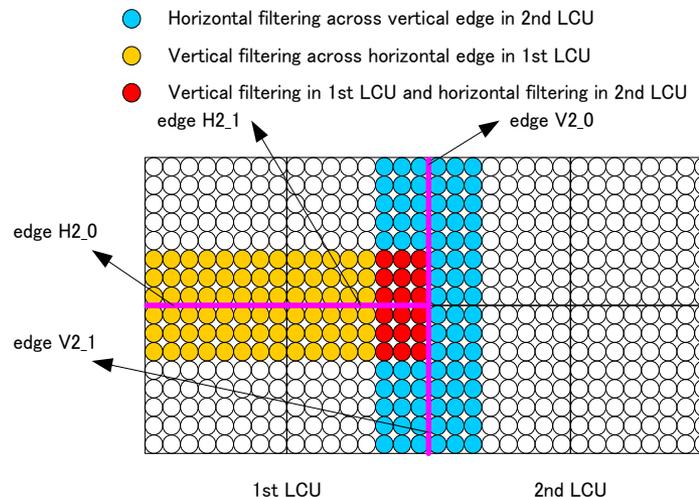
Signal Processing Technology Department No.1  
Technology Development Group  
Sony Corporation

# Content

- Motivation
- Proposal
- Experimental results
- Conclusions

# Motivation

- Deblocking filter in TMuC-0.9
  - Vertical and horizontal filtering has dependency each other between LCUs
    - Horizontal filtering (red pixels) in 2<sup>nd</sup> LCU needs to be performed after vertical filtering in 1<sup>st</sup> LCU is completed
  - ⇒ In other words, horizontal/vertical filtering can be performed in parallel within only LCU, respectively

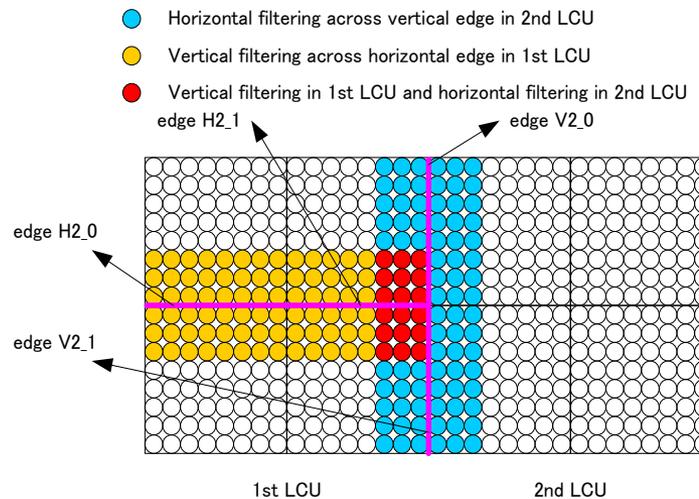


LCU : 16x16

# Motivation

- Deblocking filter in TMuC-0.9
    - Vertical and horizontal filtering has dependency each other between LCUs
      - Horizontal filtering (red pixels) in 2<sup>nd</sup> LCU needs to be performed after vertical filtering in 1<sup>st</sup> LCU is completed
- ⇒ In other words, horizontal/vertical filtering can be performed in parallel within only LCU, respectively

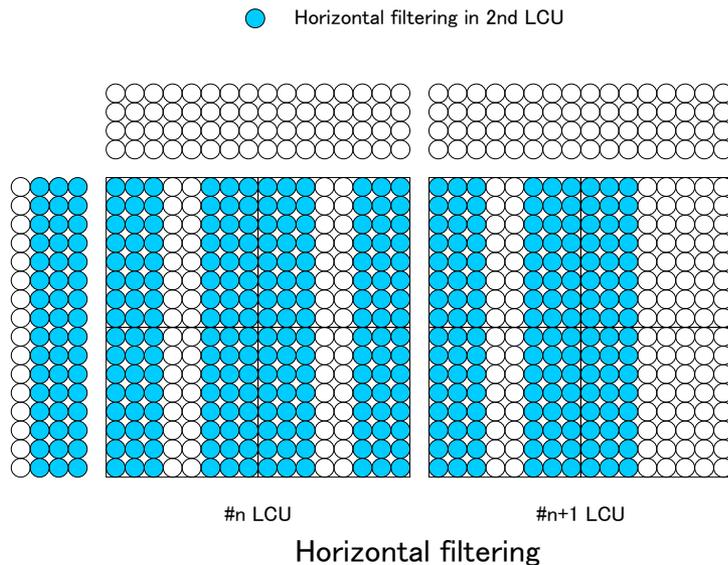
So, we propose the **parallelization** of the filtering **within a frame**



LCU : 16x16

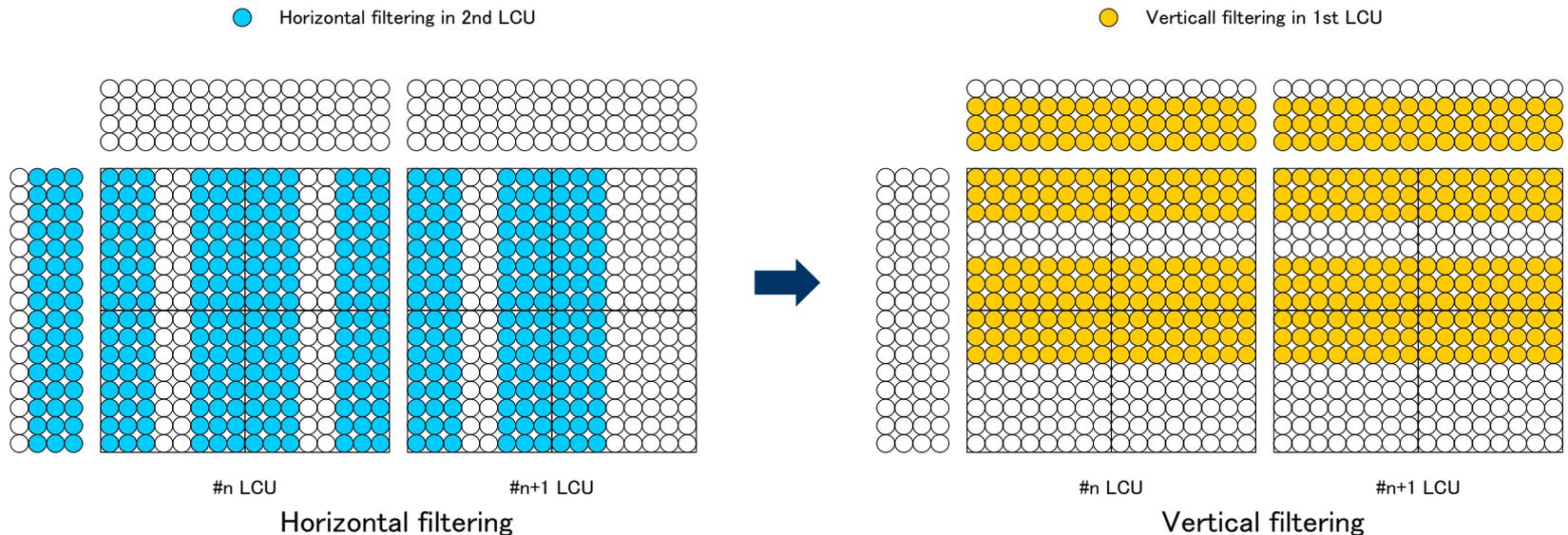
# Proposal

- Parallelization of filtering
  - Change the order of horizontal and vertical filtering
    1. Horizontal filter is performed across all block edges in processing frame



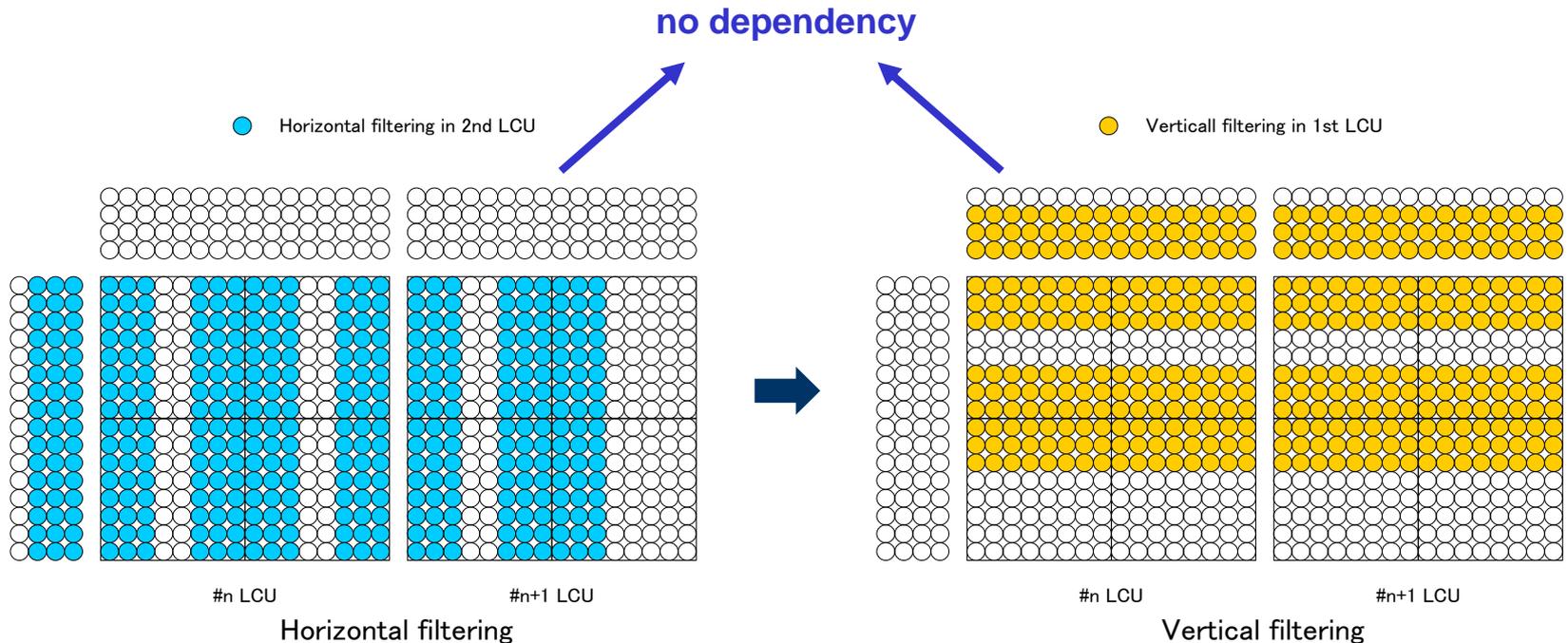
# Proposal

- Parallelization of filtering
  - Change the order for horizontal and vertical filtering
    1. Horizontal filter is performed across all block edges in processing frame
    2. Vertical filter is performed across all block edges in processing frame



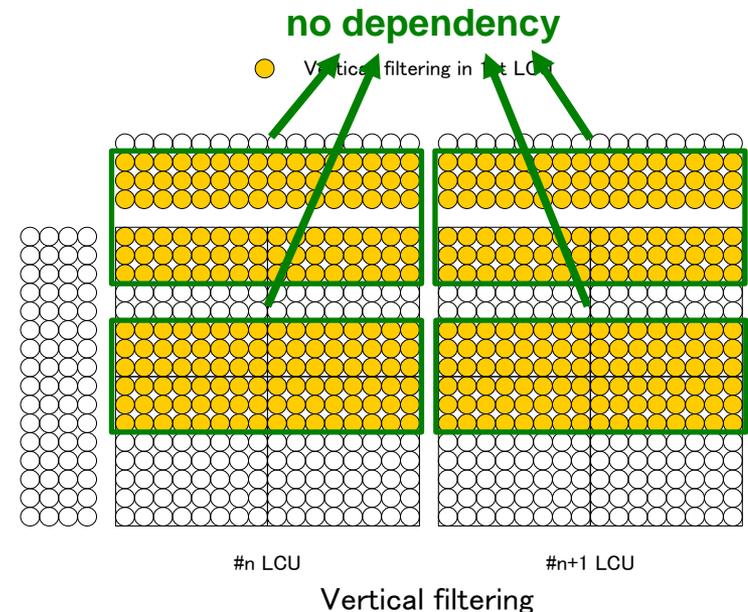
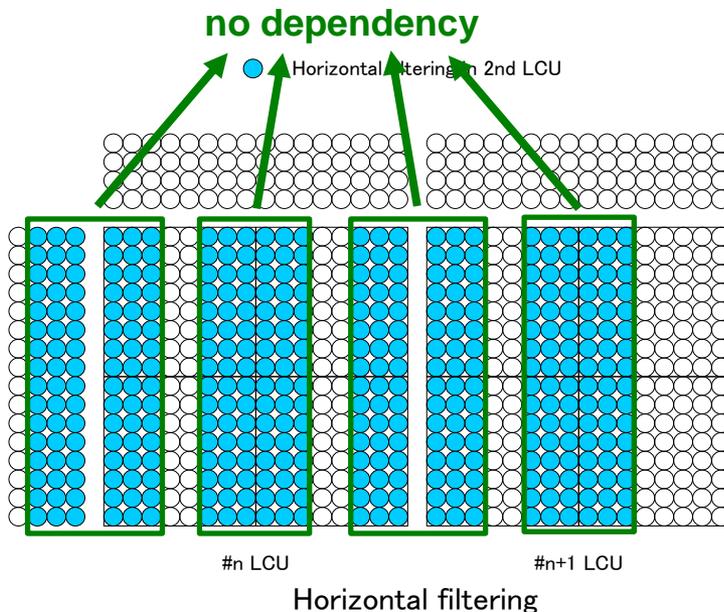
# Proposal

- Parallelization of filtering
  - Change the order for horizontal and vertical filtering
    1. Horizontal filter is performed across all block edges in processing frame
    2. Vertical filter is performed across all block edges in processing frame
 ⇒ ★Horizontal and vertical filtering has no dependency



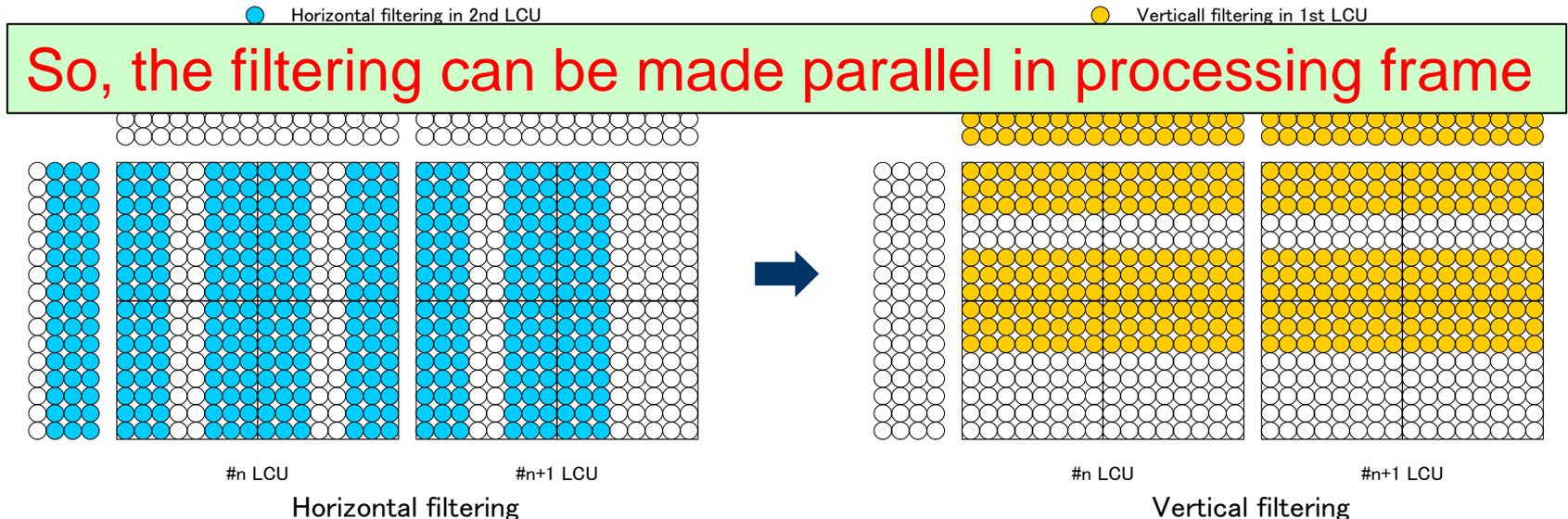
# Proposal

- Parallelization of filtering
  - Change the order for horizontal and vertical filtering
    1. Horizontal filter is performed across all block edges in processing frame
    2. Vertical filter is performed across all block edges in processing frame
 ⇒ **★Horizontal and vertical filtering has no dependency**  
**★The same directional filtering has no dependency in 8x8 adjacent edge**



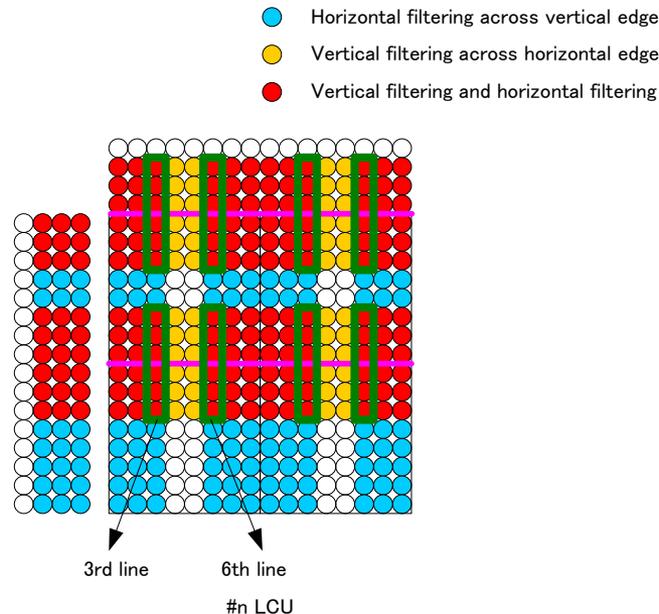
# Proposal

- Parallelization of filtering
  - Change the order for horizontal and vertical filtering
    1. Horizontal filter is performed across all block edges in processing frame
    2. Vertical filter is performed across all block edges in processing frame
 ⇒ **★Horizontal and vertical filtering has no dependency**  
**★The same directional filtering has no dependency in adjacent edge**



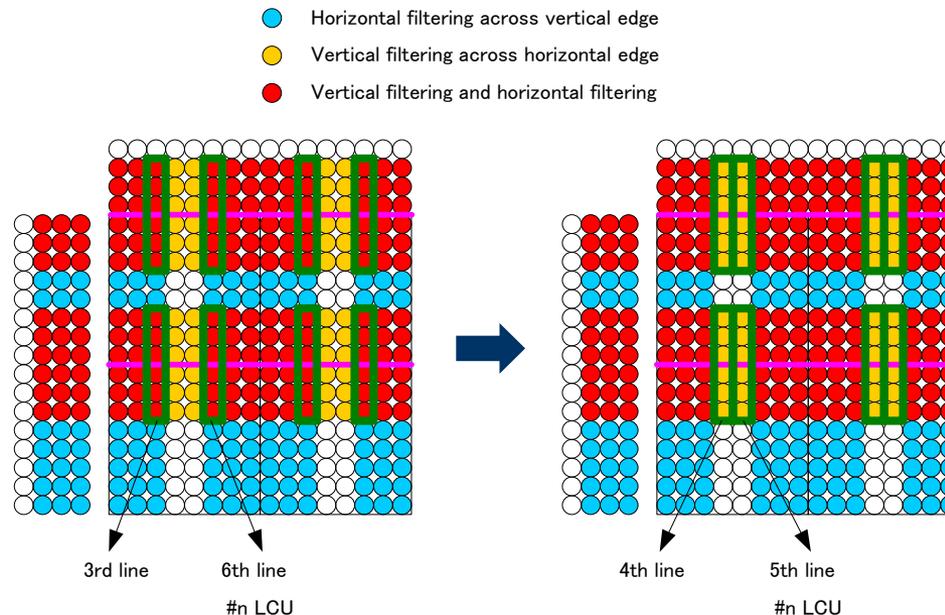
# Proposal

- Parallelization of edge judgment of block boundary
  - Both 3<sup>rd</sup> and 6<sup>th</sup> lines are used for the judgment in TMuC-0.9
    - ⇒ These lines is filtered by horizontal and vertical filter, so the judgment has **dependency**



# Proposal

- Parallelization of edge judgment of block boundary
  - Both 3<sup>rd</sup> and 6<sup>th</sup> lines are used for the judgment in TMuC-0.9
    - ⇒ These lines is filtered by horizontal and vertical filter, so the judgment has dependency
    - ⇒ We propose to use both **4<sup>th</sup> and 5<sup>th</sup> lines** for the judgment to **remove above dependency**

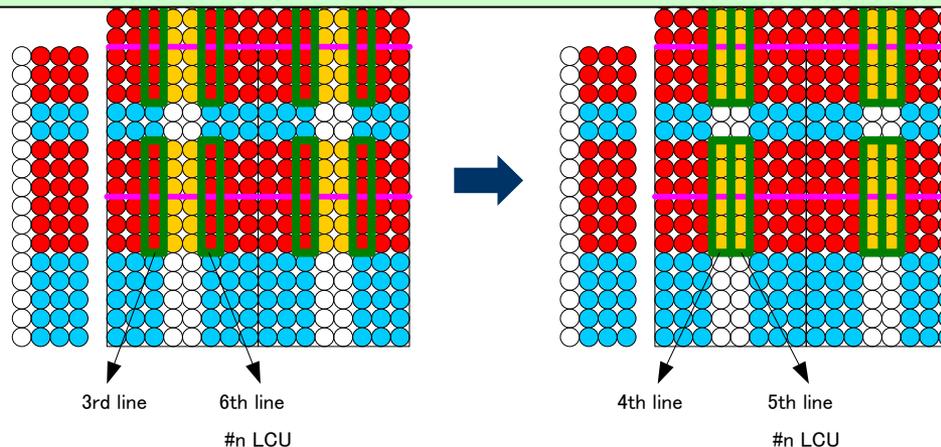


# Proposal

- Parallelization of edge judgment of block boundary
  - Both 3<sup>rd</sup> and 6<sup>th</sup> lines are used for the judgment in TMuC-0.9
    - ⇒ These lines are filtered by horizontal and vertical filter, so the judgment has dependency
    - ⇒ We propose to use both **4<sup>th</sup> and 5<sup>th</sup> lines** for the judgment to **remove above dependency**

- Horizontal filtering across vertical edge
- Vertical filtering across horizontal edge

So, the judgment can be made parallel in processing frame



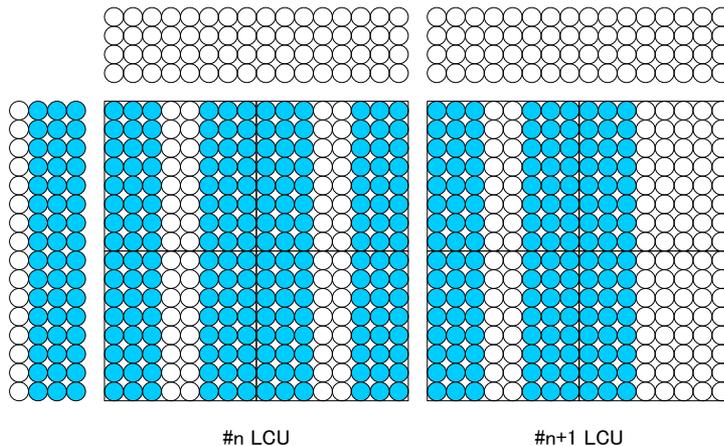
# Expected effect

- More flexible deblocking filter configuration
  - We can freely design the number of the parallelization according to the design specification
  - In addition to the parallel processing, the filtering can be performed in LCU-unit like the filtering in TMuC-0.9

# Expected effect

- More flexible deblocking filter configuration
  - We can freely design the number of the parallelization according to the design specification
  - In addition to the parallel processing, the filtering can be performed in LCU-unit like the filtering in TMuC-0.9
    - First the horizontal filter is performed across horizontal edges in 1<sup>st</sup> and 2<sup>nd</sup> LCU

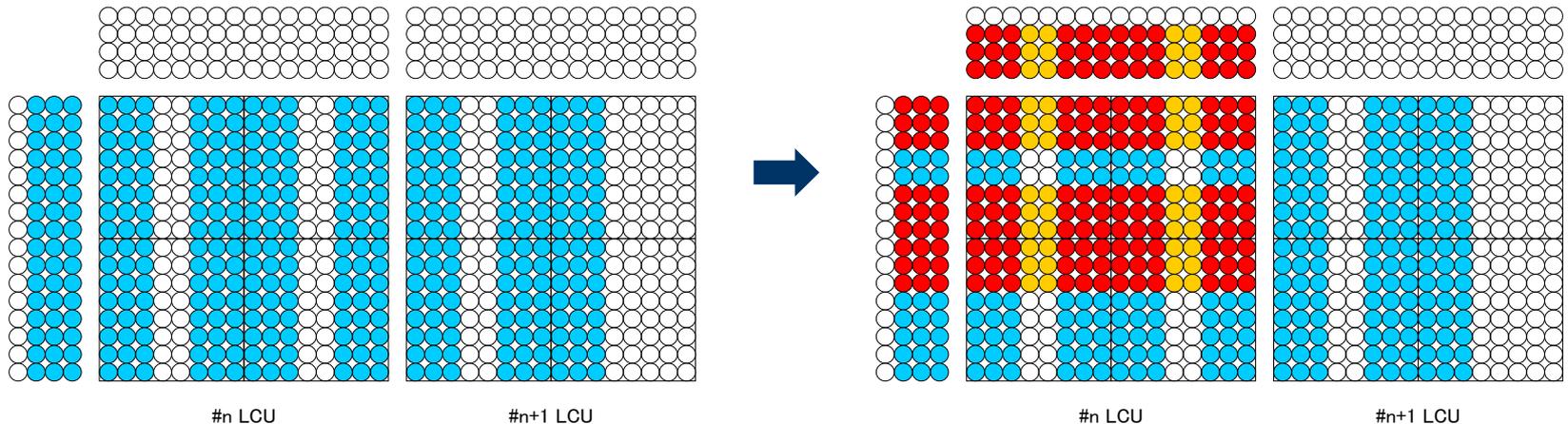
- Horizontal filtering across vertical edges
- Vertical filtering across horizontal edges
- Vertical filtering and horizontal filtering



# Expected effect

- More flexible deblocking filter configuration
  - We can freely design the number of the parallelization according to the design specification
  - In addition to the parallel processing, the filtering can be performed in LCU-unit like the filtering in TMuC-0.9
    - First the horizontal filter is performed across horizontal edges in 1<sup>st</sup> and 2<sup>nd</sup> LCU
    - Second the vertical filter is performed across vertical edges in 1<sup>st</sup> LCU

-  Horizontal filtering across vertical edges
-  Vertical filtering across horizontal edges
-  Vertical filtering and horizontal filtering



# Experimental results

- Parallelization of filtering and judgment
  - It has slight larger loss in all test conditions, but the loss is still small enough

		High Efficiency			Low Complexity		
		Y BD-rate	U BD-rate	V BD-rate	Y BD-rate	U BD-rate	V BD-rate
Intra	Class-A	0.1	0.0	0.0	0.1	0.0	0.0
	Class-B	0.1	0.0	0.0	0.1	0.0	0.0
	Class-C	0.1	0.0	0.0	0.1	-0.1	-0.1
	Class-D	0.1	0.0	0.0	0.1	0.0	-0.1
	Class-E	0.1	0.0	0.0	0.1	0.0	0.0
	<b>All</b>	<b>0.1</b>	<b>0.0</b>	<b>0.0</b>	<b>0.1</b>	<b>0.0</b>	<b>0.0</b>
Random access	Class-A	0.1	0.2	0.3	0.1	0.0	0.1
	Class-B	0.1	0.1	-0.1	0.1	0.1	0.0
	Class-C	0.1	0.1	-0.1	0.1	0.1	0.0
	Class-D	0.1	-0.1	0.1	0.1	0.0	0.1
	<b>All</b>	<b>0.1</b>	<b>0.0</b>	<b>0.0</b>	<b>0.1</b>	<b>0.1</b>	<b>0.0</b>
Low delay	Class-B	0.2	-0.1	-0.2	0.2	0.0	-0.1
	Class-C	0.1	0.0	0.1	0.2	-0.1	0.1
	Class-D	0.1	-0.4	0.1	0.1	0.3	0.2
	Class-E	0.4	0.4	0.0	0.4	0.3	0.1
	<b>All</b>	<b>0.2</b>	<b>-0.1</b>	<b>0.0</b>	<b>0.2</b>	<b>0.1</b>	<b>0.1</b>

# Experimental results

- Parallelization of only filtering
  - The difference of TMuC-0.9 is negligible

		High Efficiency			Low Complexity		
		Y BD-rate	U BD-rate	V BD-rate	Y BD-rate	U BD-rate	V BD-rate
Intra	Class-A	0.0	0.0	0.0	0.0	0.0	0.0
	Class-B	0.0	0.0	0.0	0.0	0.0	0.0
	Class-C	0.0	0.0	0.0	0.0	-0.1	-0.1
	Class-D	0.0	0.0	0.0	0.0	0.0	-0.1
	Class-E	0.0	0.0	0.0	0.0	0.0	0.0
	<b>All</b>	<b>0.0</b>	<b>0.0</b>	<b>0.0</b>	<b>0.0</b>	<b>0.0</b>	<b>0.0</b>
Random access	Class-A	0.0	0.2	0.3	0.0	0.1	0.0
	Class-B	0.0	0.0	0.0	0.0	0.1	0.0
	Class-C	0.0	0.0	0.0	0.0	0.0	0.0
	Class-D	0.0	0.1	0.1	0.0	0.0	0.0
	<b>All</b>	<b>0.0</b>	<b>0.1</b>	<b>0.0</b>	<b>0.0</b>	<b>0.0</b>	<b>0.0</b>
Low delay	Class-B	0.0	0.0	-0.3	0.0	-0.1	-0.1
	Class-C	0.0	0.0	0.0	0.0	0.0	0.0
	Class-D	0.1	-0.5	0.3	-0.1	0.0	-0.1
	Class-E	-0.1	0.2	-0.1	0.0	0.1	0.0
	<b>All</b>	<b>0.0</b>	<b>-0.1</b>	<b>0.0</b>	<b>0.0</b>	<b>0.0</b>	<b>0.0</b>

# Conclusions

- We propose parallelization of both filtering and judgment
  - It has slight losses but they are still small enough
  - It has more flexible configuration than parallelization of only judgment