<7th JCT-VC meeting @Geneva, November 2011>

<JCTVC-G416> CU-based Merge Candidate List Construction

Hui Yong Kim

hykim5@etri.re.kr

Realistic Media Research Team, ETRI
Media Lab., KyungHee Univ.

21 November 2011

Summary

❑ CU-based approach for merge candidate list construction

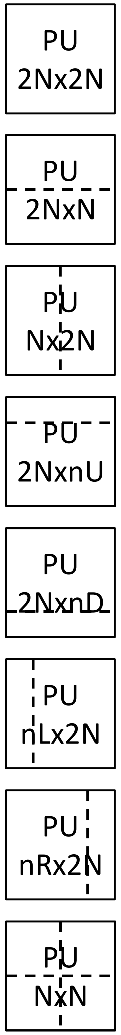
- ❖ A CU can have at most one merge candidate list that can be constructed prior to encoding and decoding of the internal PUs.
- ❖ Only one common merge candidate list is used for all PUs in a CU regardless of the PU partition type.

- ❖ Provides simpler design, reduced complexity, and improved parallelism compared to the PU-based one used in HM4.
- ❖ Roughly 3~6 % encoding time reduction with the penalty of roughly 0.2~0.5% coding loss depending on the test configurations.

Introduction

❑ Observation 1

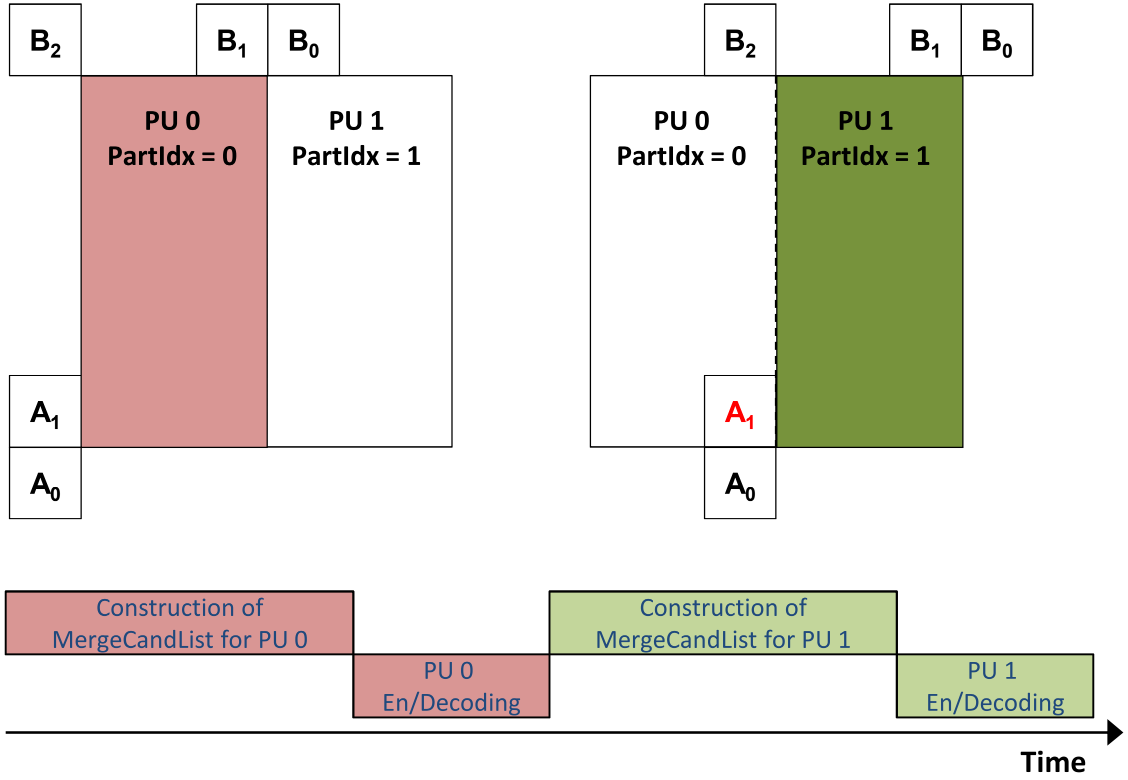
- ❖ The current design has 8 PU partition types with 17 geometry (i.e., size and position) variants
  - ➢ MCL (merge candidate list) has to be constructed max 17 times for encoding and max 4 times for decoding a CU
  - ➢ Max 17 variants of MCL construction logics should be prepared for encoding or decoding a CU



Introduction (cont'd)

❑ Observation 2

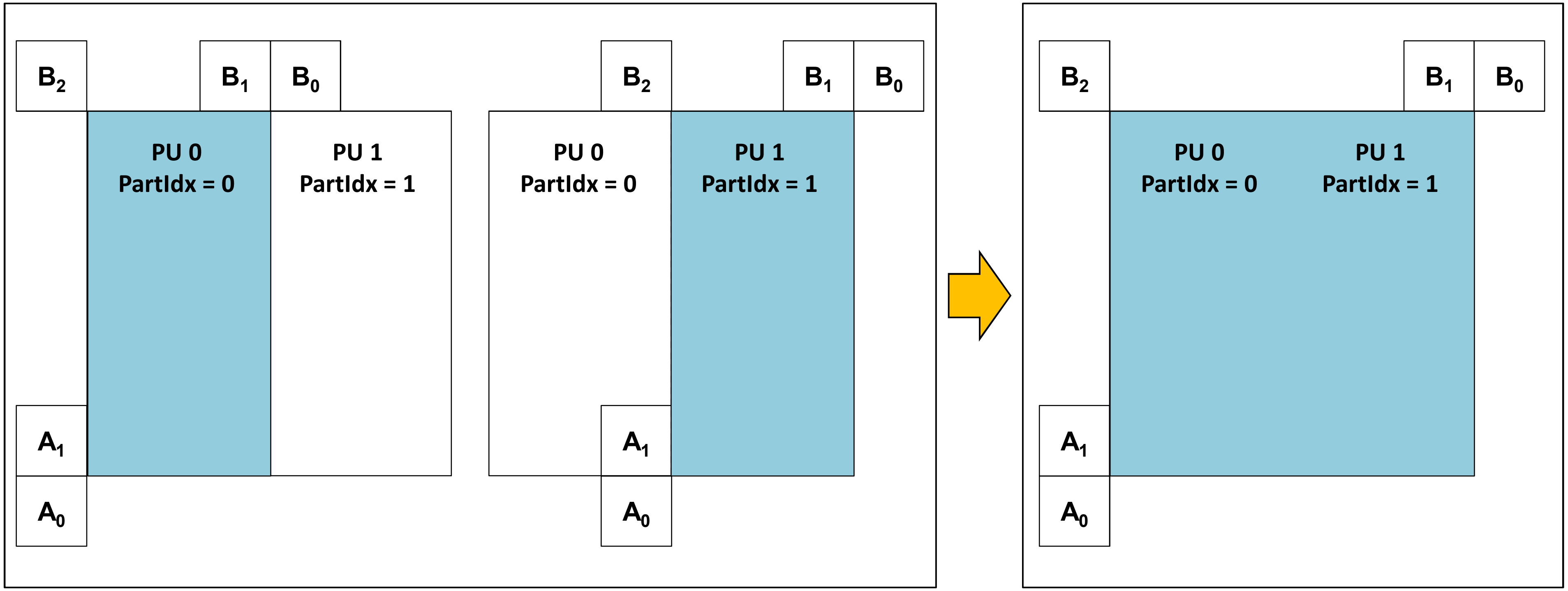
- ❖ In the current design, the PUs in a CU should be sequentially encoded and decoded.
- ❖ Example)



Proposal

❑ CU-based Merge Candidate List Construction

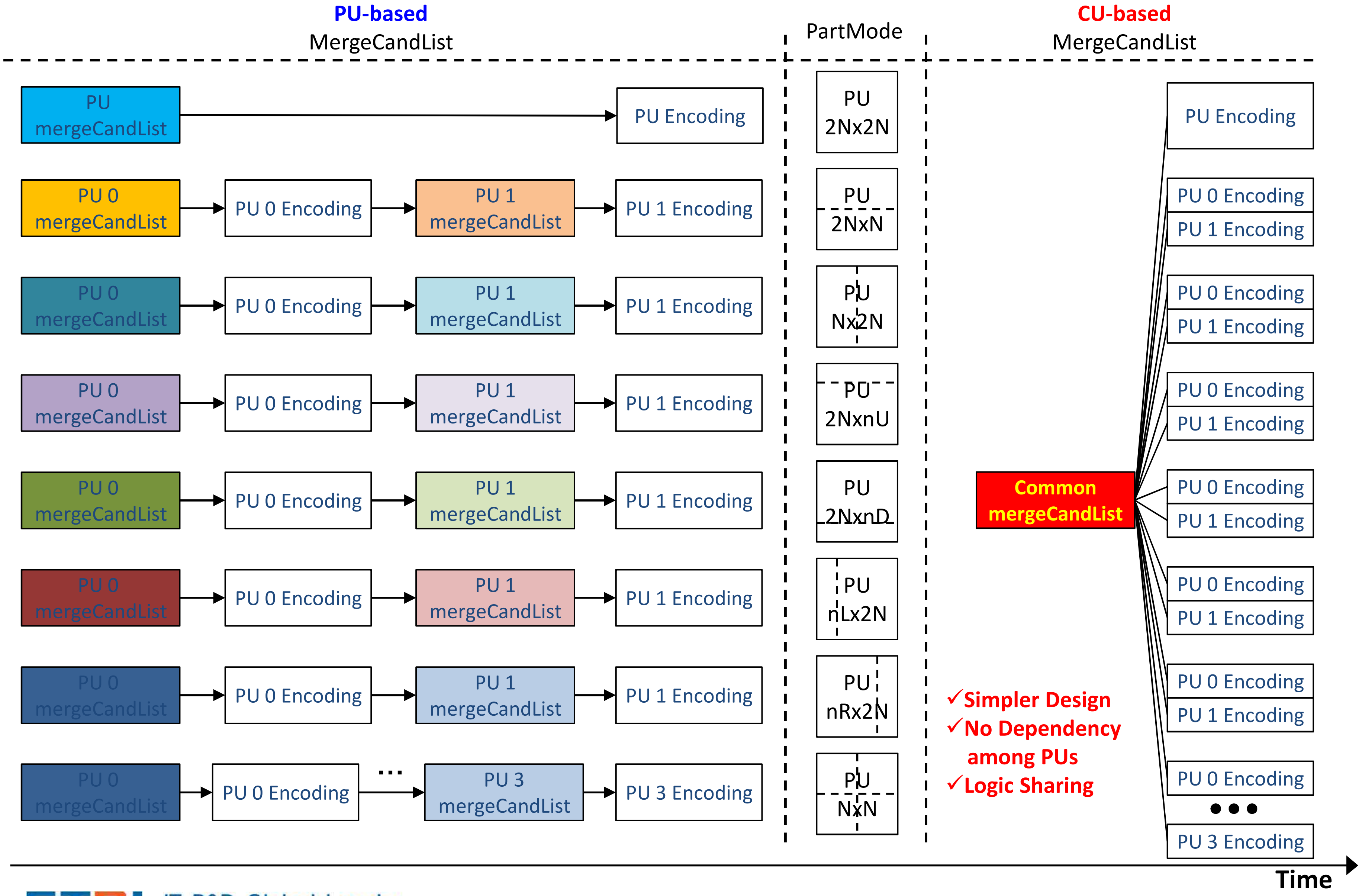
- ❖ Only one common merge candidate list for all PUs are allowed in a CU, regardless of the PU partition type.
- ❖ The common merge candidate list is the same one as 2Nx2N PU.



HM4.0

proposed

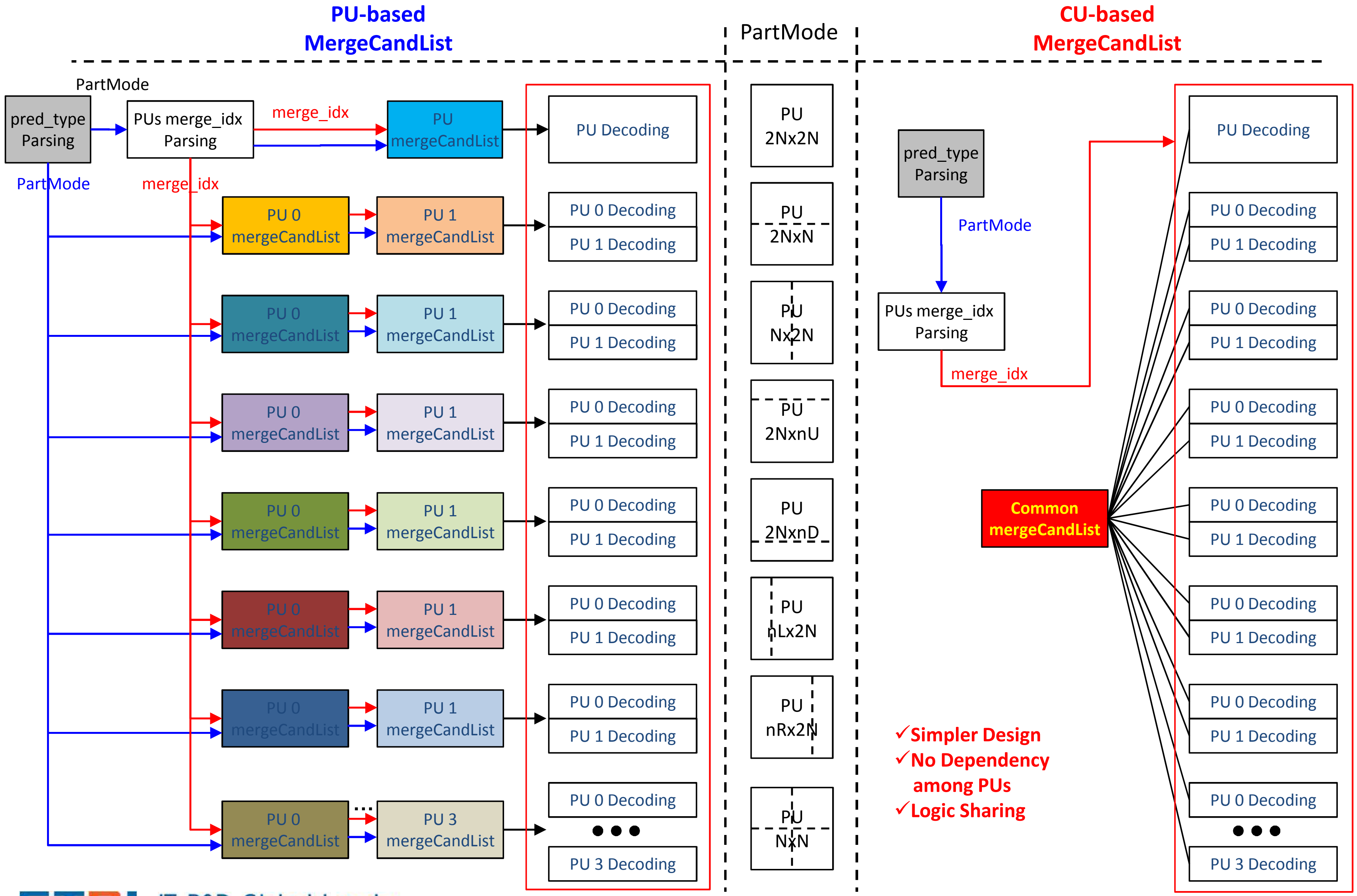
Comparison - Encoder



✓Simpler Design
✓No Dependency among PUs
✓Logic Sharing

ETRI IT R&D Global Leader

Comparison - Decoder



✓Simpler Design

✓No Dependency among PUs

✓Logic Sharing

Comparison (Summary)

❑ **Complexity benefits of the proposed method (compared with HM4)**

Aspects	Proposed (CU-based)	HM4 (PU-based)
○ the # of executions of MCL construction process	at most once for a CU	max 17 times for a CU (encoding) max 4 times for a CU (decoding)
○ the required # of MCL construction modules (H/W or S/W)	single module can be shared for all PU types of a CU size	17 variants are required for each CU size
○ parallel encoding/decoding of the merge-mode PUs in a CU	Once the common merge candidate list has constructed for a CU, the remaining encoding/decoding process for the internal PUs can be performed in parallel.	Parallel construction of merge candidate lists is impossible. After the sequential construction of merge candidate lists for all the PUs, the remaining encoding/decoding process for the internal PUs can be done in parallel.

Experimental Results

❑ Proposed vs HM4.0

- ❖ Cross-checked by HHI (JCTVC-G899)

	Random Access HE			Random Access LC		
	Y	U	V	Y	U	V
Class A	0.2%	0.0%	0.0%	0.2%	0.1%	0.2%
Class B	0.2%	0.1%	0.2%	0.1%	0.2%	0.2%
Class C	0.3%	0.2%	0.3%	0.3%	0.2%	0.3%
Class D	0.4%	0.3%	0.3%	0.4%	0.1%	0.4%
Class E						
Overall	0.3%	0.2%	0.2%	0.2%	0.2%	0.3%
	0.3%	0.2%	0.2%	0.2%	0.2%	0.3%
Enc Time[%]		97%			96%	
Dec Time[%]		100%			100%	

	Low delay B HE			Low delay B LC		
	Y	U	V	Y	U	V
Class A						
Class B	0.3%	0.3%	0.5%	0.3%	0.3%	0.4%
Class C	0.4%	0.5%	0.6%	0.4%	0.7%	0.3%
Class D	0.5%	0.7%	0.3%	0.4%	0.6%	0.3%
Class E	0.9%	1.2%	1.5%	0.7%	0.7%	0.5%
Overall	0.5%	0.6%	0.7%	0.4%	0.5%	0.4%
	0.5%	0.6%	0.6%	0.4%	0.5%	0.4%
Enc Time[%]		94%			96%	
Dec Time[%]		100%			100%	

Conclusion

- ❑ **CU-based approach for merge candidate list construction**
  - ❖ It uses only one common merge candidate list for all PUs in CU regardless of the PU partition type.
- ❑ **The proposed method provides**
  - ❖ Simpler design - common merge candidate list
  - ❖ Reduced complexity - at most once for a CU(encoding/decoding)
  - ❖ Logic sharing - single module
  - ❖ Improved parallelism
- ❑ **Experimental results**
  - ❖ 3~6 % encoding time reduction
  - ❖ 0.2~0.5% coding loss
- ❑ **It is recommended to adopt this change into the HEVC test model.**

Thank You Very Much !

www.etri.re.kr