

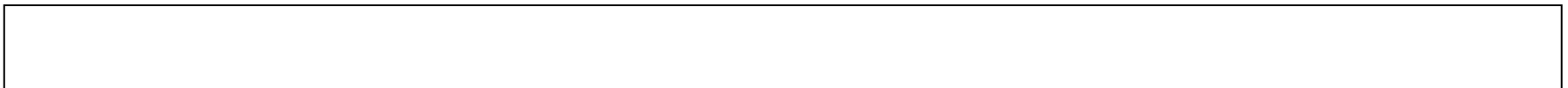
# **Parallel Context Processing techniques for high coding efficiency entropy coding in HEVC (JCTVC-B088.doc)**

**Madhukar Budagavi, Mehmet Umut Demircin**

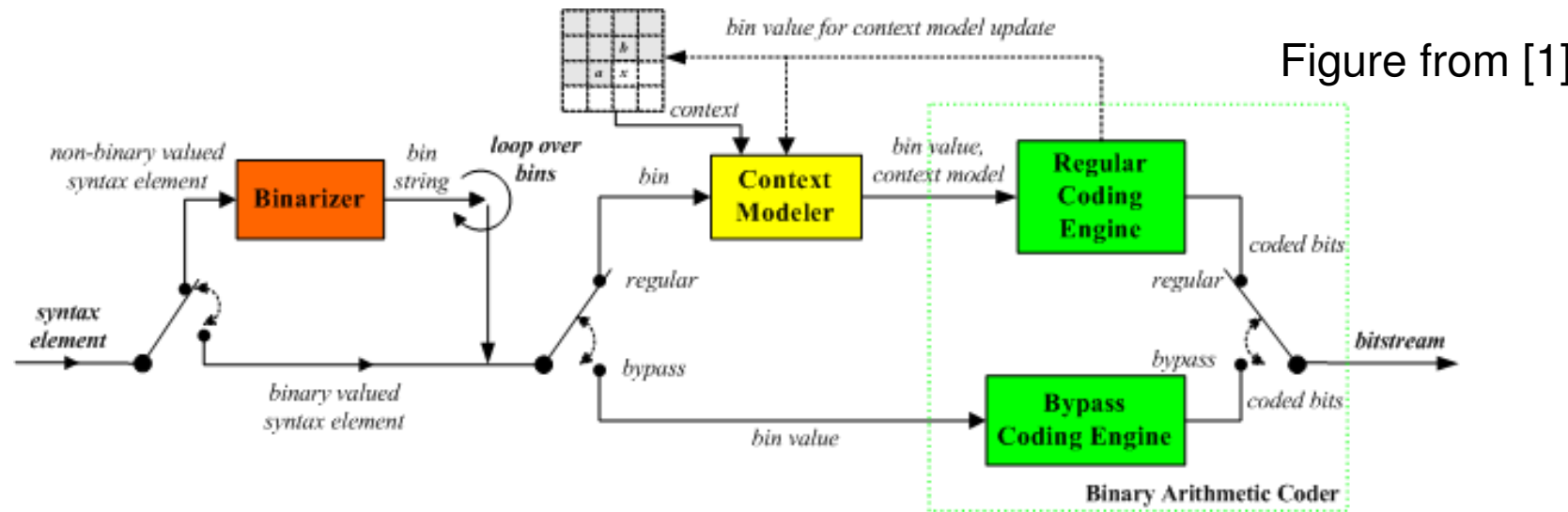
**Texas Instruments Inc.**

**Joint Collaborative Team on Video Coding (JCT-VC)  
of ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11**

**2nd Meeting: Geneva, CH, 21-28 July, 2010**



# Motivation



- CABAC is a highly serial processing block
- Serial nature comes from:
  - Binarizer (variable length coding)
  - Context modeler (probability depends on LPS/MPS, position of bin)
  - Binary arithmetic coder (interval subdivision)
- CABAC is bottleneck for real-time decoding at high bit-rate

# Parallelization techniques for CABAC

- Bin-level parallelization
  - N-bin binary arithmetic coder (NBAC)
  - PIPE
  - V2V
- Syntax-element-level parallelism
- Sub-slice-level parallelism
  - Entropy slices
  - Interleaved entropy slices
  - “Sub-streams”

# Bin-level parallelization

- NBAC
  - Codes N-bin/cycle
  - Contexts for N-bins are calculated using conditional probability
- PIPE/V2V

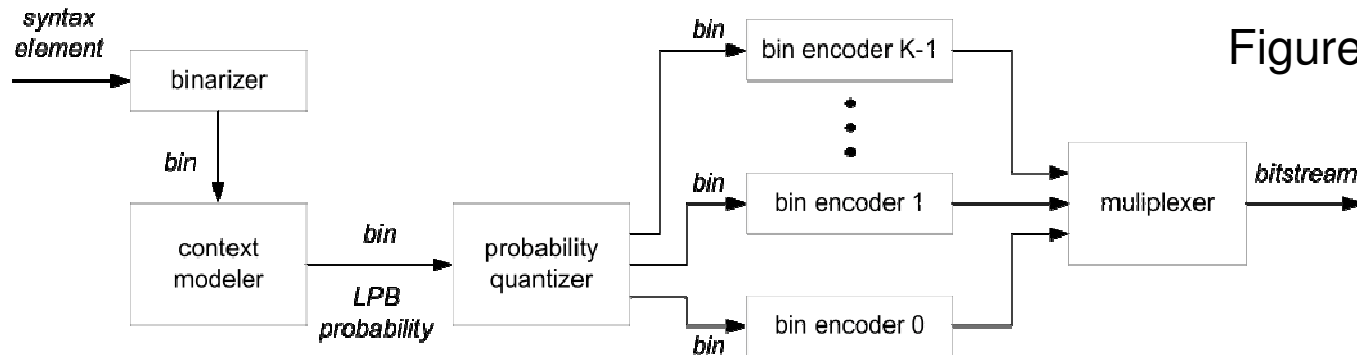
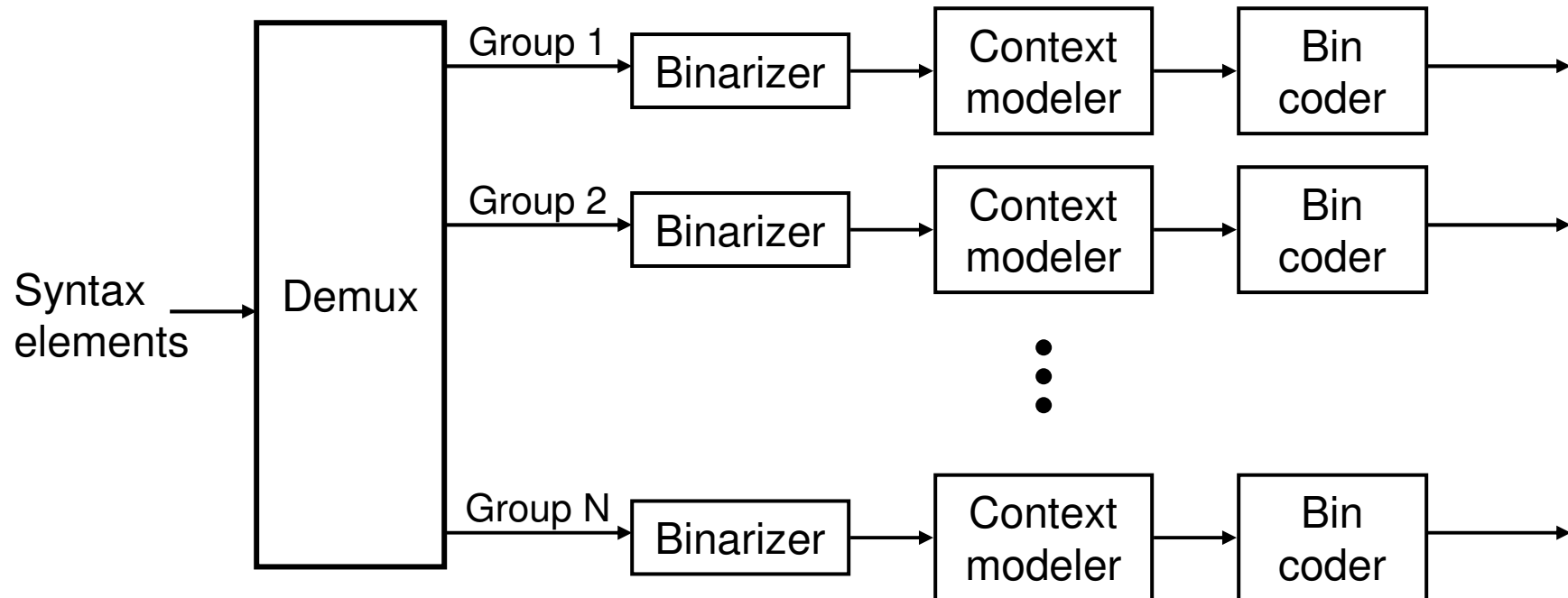


Figure from [2]

- PIPE estimated to achieve 3 bins/cycle decoding
- (AVC CABAC is being implemented in industry at 2 bins/cycles)
- Overall throughput of CABAC is still limited
  - Serial dependency in binarizer and context processing still exists
  - Need techniques for parallelizing these two blocks in CABAC to leverage throughput gains at bin-level

# Syntax-element partitioning



- Parallelizes at all levels: binarizer, context modeler, bin coder
- Bin coder can be arithmetic coder or PIPE/V2V
- More results in [4][9]

# Parallel Context Processing (PCP)

- Simple data rearrangement techniques within a block to enable parallelization of context processing
- Three types of PCP proposed
  - Coeff Level BinIdx 0 PCP
  - Coefficient Sign PCP
  - Significance Map PCP

# Coeff Level BinIdx 0 PCP (1)

- Context (ctxIdxInc) used for `coeff_abs_level_minus1` depends on the position of the bin (binIdx) in AVC

If binIdx is equal to 0, ctxIdxInc is derived by

$$\text{ctxIdxInc} = ( ( \text{numDecodAbsLevelGt1} \neq 0 ) ? 0 : \text{Min}( 4, 1 + \text{numDecodAbsLevelEq1} ) ) \quad (9-23)$$

Otherwise (binIdx is greater than 0), ctxIdxInc is derived by

$$\text{ctxIdxInc} = 5 + \text{Min}( 4 - ( ( \text{ctxBlockCat} == 3 ) ? 1 : 0 ), \text{numDecodAbsLevelGt1} ) \quad (9-24)$$

- Coeff Level BinIdx 0 PCP technique codes BinIdx 0 in a different plane
- Context processing for all the bins with binIdx 0 for all the coeffs level in a block can be carried out in parallel to bin processing of binIdx = 0 and before the decoding of bins with binIdx 0

# Coeff Level BinIdx 0 PCP (2)

- Implemented in TMuC-0.1, 121 frames were encoded using the scripts in cfg\cfp-fast
- As expected, no coding efficiency loss

Coefficient Bin0 PCP BD-Rate increase			
		Alpha	Beta
S01	Traffic	-0.06	
S02	PeopleOnStreet	0.00	
S03	Kimono	0.03	0.02
S04	ParkScene	-0.01	0.00
S05	Cactus	-0.06	-0.04
S06	BasketballDrive	-0.02	0.03
S07	BQTerrace	0.16	-0.10
S08	BasketballDrill	-0.07	0.01
S09	BQMall	-0.07	0.08
S10	PartyScene	0.01	-0.10
S11	RaceHorses	-0.17	0.10
S12	BasketballPass	-0.06	-0.05
S13	BQSquare	0.07	0.14
S14	BlowingBubbles	0.06	0.08
S15	RaceHorses	0.00	0.00
S16	Vidyo1		0.09
S17	Vidyo3		0.22
S18	Vidyo4		0.11
Avg		-0.01	0.04
Min		-0.17	-0.10
Max		0.16	0.22



# Coeff sign PCP (1)

AVC coding of coefficient levels and sign

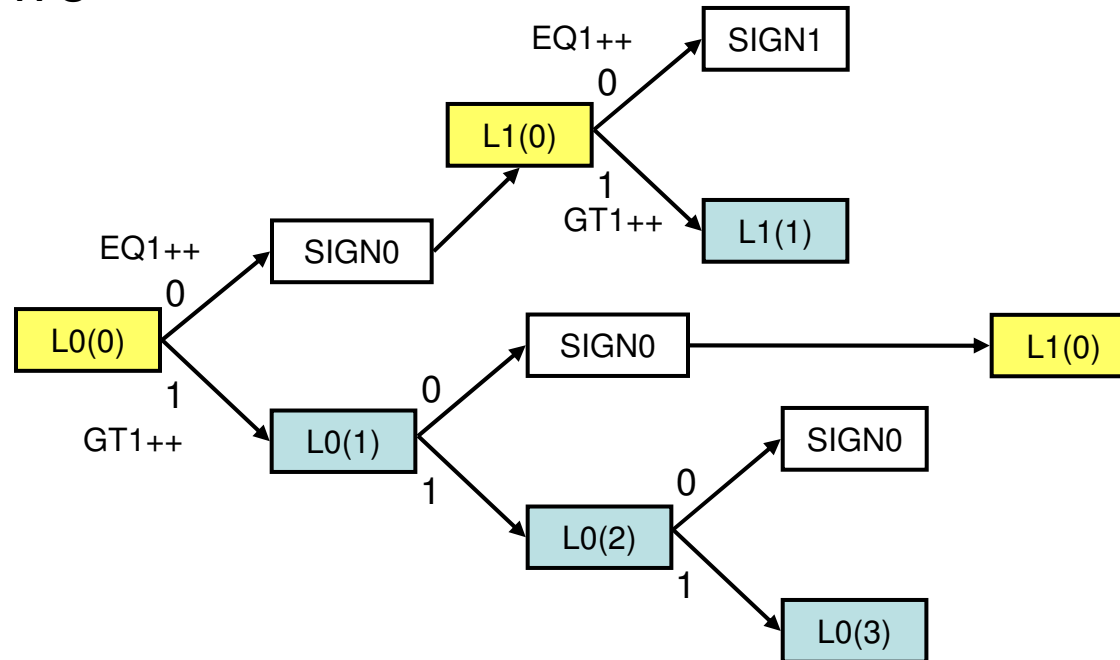
```
for (i = MaxNumCoeff(BlockType)-1; i >= 0; i--)  
{  
    {  
        Encode coeff_abs_level_minus1[i];  
        Encode coeff_sign_flag[i];  
    }  
}
```

Coefficient sign PCP: Code sign in a different plane

```
for (i = MaxNumCoeff(BlockType)-1; i >= 0; i--)  
{  
    Encode coeff_sign_flag[i];  
}  
  
for (i = MaxNumCoeff(BlockType)-1; i >= 0; i--)  
{  
    Encode coeff_abs_level_minus1[i];  
}
```

# Coeff sign PCP (2)

Context processing tree for parallel decoding of coefficient level/sign decoding in AVC

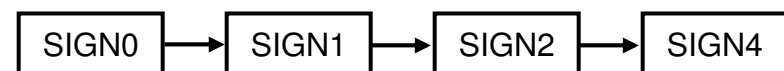
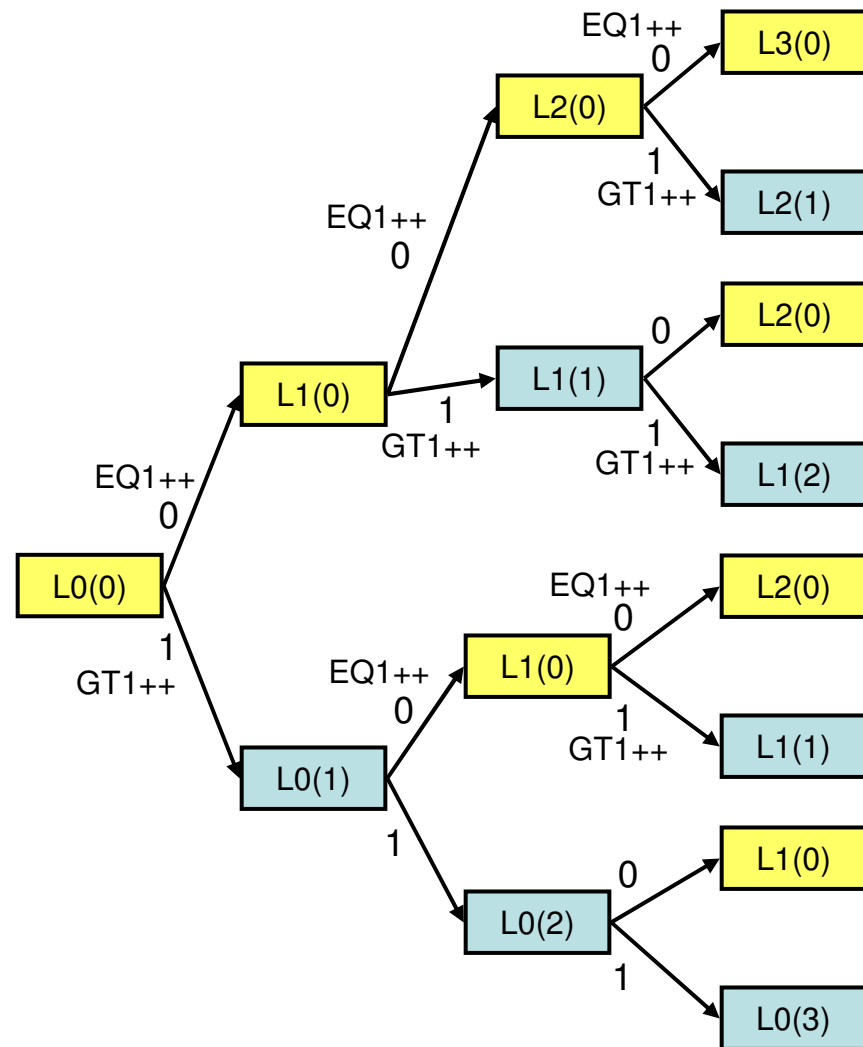


$L_k(n)$  – nth bin of kth level  
 $SIGN_k$  – Sign of kth level

- Context processing that happens at SIGN node is wasteful since SIGN is by-pass coded
- Context processing efficiency is about 60%

# Coeff sign PCP (3)

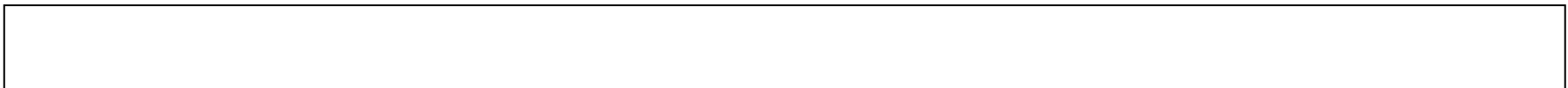
- Separating sign into a separate plane improves context processing efficiency
- Context processing efficiency is 100%
- In PIPE/V2V, coding of sign in separate plane also reduces bitstream switching overhead



# Coeff sign PCP (4)

- Implemented in TMuC-0.1, 121 frames were encoded using the scripts in cfg\cfp-fast
- As expected, no coding efficiency loss

Coefficient Sign PCP BD-Rate increase			
		Alpha	Beta
S01	Traffic	-0.08	
S02	PeopleOnStreet	-0.01	
S03	Kimono	0.02	-0.02
S04	ParkScene	0.03	0.02
S05	Cactus	-0.07	-0.01
S06	BasketballDrive	0.05	0.06
S07	BQTerrace	0.15	0.10
S08	BasketballDrill	0.01	0.00
S09	BQMall	-0.01	0.10
S10	PartyScene	-0.03	0.00
S11	RaceHorses	-0.09	-0.01
S12	BasketballPass	0.08	-0.30
S13	BQSquare	0.12	0.06
S14	BlowingBubbles	0.06	-0.11
S15	RaceHorses	-0.03	-0.07
S16	Vidyo1		0.18
S17	Vidyo3		0.07
S18	Vidyo4		0.06
Avg		0.01	0.01
Min		-0.09	-0.30
Max		0.15	0.18



# Significance Map PCP (1)

AVC significance map coding

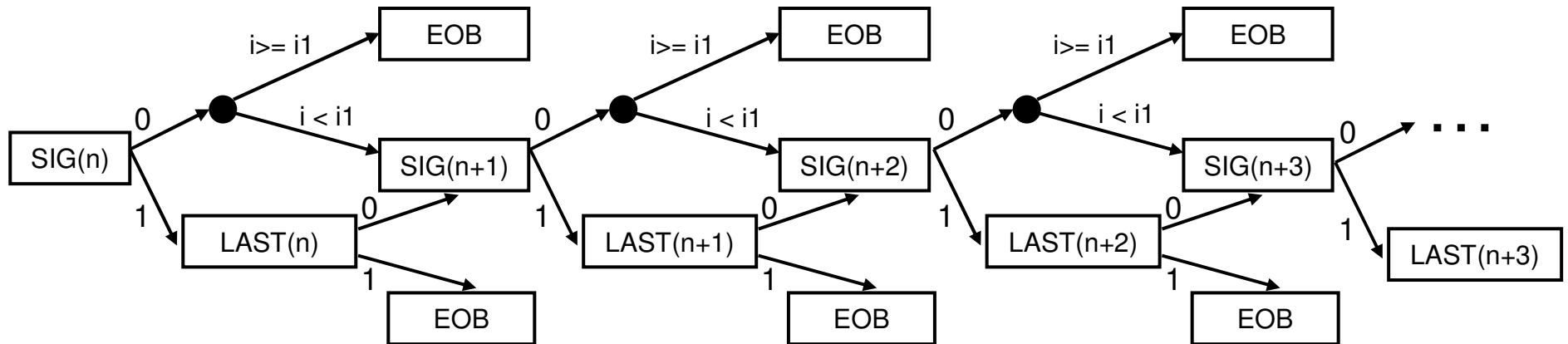
```
for (i = 0 ;i < MaxNumCoeff(BlockType)-1 ;i++)
{
    Encode significant_coeff_flag[i];
    if(significant_coeff_flag[i])
        Encode last_significant_coeff_flag[i];
    if (last_significant_coeff_flag[i])
        break ;
}
```

# Significance Map PCP (2)

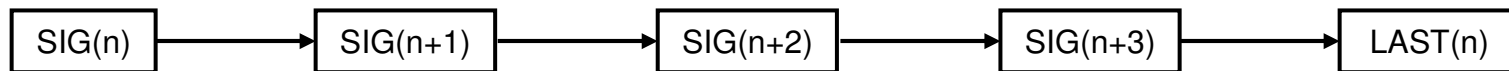
Significance map PCP – Code one last every N significant\_coeff\_flag

```
for (i = 0 ;i < MaxNumCoeff(BlockType)-1 ;i += K)
{
    if(i+K < MaxNumCoeff(BlockType)-1)
        j1 = K ;
    else
        j1 = K-1 ;
    sig = 0 ;
    for (j = 0 ;j < j1 ;j++)
    {
        Encode significant_coeff_flag[i*K+j];
        sig += significant_coeff_flag[i*K+j]
    }
    if(sig)
        Encode last_significant_coeff_flag[i];
    if (last_significant_coeff_flag[i])
        break ;
}
```

# Significance map PCP (3)



(a) 5X parallelism in AVC CABAC SigMap context processing using speculative computing (which happens every bin). Counter  $i$  indicates bin position,  $i1$  is  $\text{MaxNumCoeff}(\text{BlockType})-1$ , EOB denotes end of block. SIG - significant\_coeff\_flag. LAST - last\_significant\_coeff\_flag.



(b) 5X parallelism in "N SIG, 1 LAST" context processing using speculative computing (which happens at only 5<sup>th</sup> bin).

# Significance map PCP BD-Rate (4)

- Last every 2:
  - Alpha: 0.07%
  - Beta: 0.05%
  - 3X parallelism
- Last every 4
  - Alpha: 0.25%
  - Beta: 0.28%
  - 5X parallelism

Significance map PCP BD-Rate increase							
		Alpha					
		Last 4	Last 2			Beta	
						Last 4	Last 2
S01				S01	Traffic		
S02	Traffic	0.17	0.09	S02	PeopleOnStreet		
S03	PeopleOnStreet	0.34	0.04	S03	Kimono	0.39	0.02
S04	Kimono	0.46	0.08	S04	ParkScene	0.17	0.04
S05	ParkScene	0.28	0.12	S05	Cactus	0.63	0.13
S06	Cactus	0.34	-0.03	S06	BasketballDrive	0.53	0.27
S07	BasketballDrive	0.41	0.15	S07	BQTerrace	0.15	0.12
S08	BQTerrace	0.26	0.27	S08	BasketballDrill	0.18	0.04
S09	BasketballDrill	0.30	0.06	S09	BQMall	0.40	0.16
S10	BQMall	0.17	-0.01	S10	PartyScene	-0.13	-0.10
S11	PartyScene	0.00	0.03	S11	RaceHorses	0.46	0.01
S12	RaceHorses	0.32	-0.11	S12	BasketballPass	0.11	-0.10
S13	BasketballPass	0.25	0.28	S13	BQSquare	-0.34	0.09
S14	BQSquare	-0.08	0.08	S14	BlowingBubbles	0.15	-0.08
S15	BlowingBubbles	0.11	-0.02	S15	RaceHorses	0.53	0.06
S16	RaceHorses	0.46	0.04	S16	Vidyo1	0.72	0.02
S17	Vidyo1			S17	Vidyo3	0.25	0.14
S18	Vidyo3			S18	Vidyo4	0.62	0.25
	Vidyo4						
	Avg	0.25	0.07		Avg	0.28	0.05
	Min	-0.08	-0.11		Min	-0.34	-0.10
	Max	0.46	0.28		Max	0.72	0.27

7/23/2010



# Conclusions

- Bin-level parallelism techniques alone do not solve CABAC throughput problem
- Context processing parallelism and binarization parallelism are required too
- Syntax element partitioning parallelizes binarization, context processing and bin-level
- Three parallel context processing (PCP) techniques are proposed that can be used with any context-based entropy coding scheme
  - Techniques involve simple data rearrangement within a block to enable parallelization of context processing
  - Coefficient Sign PCP, Coeff Level BinIdx 0 PCP, significance map PCP
- Recommend starting tool experiment on parallel context processing (PCP)