

JCTVC-K0160

Field indication SEI

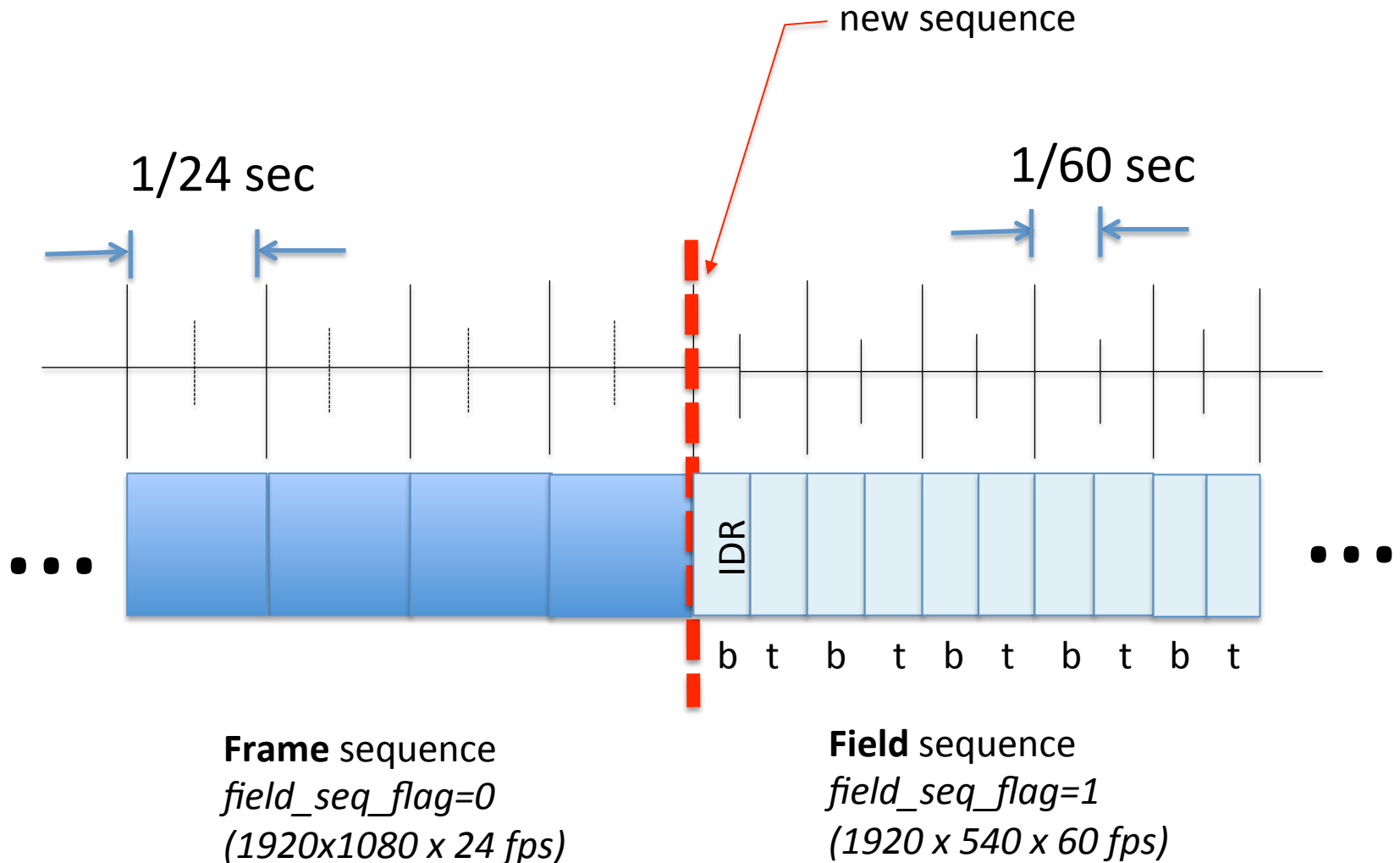
Problem (K165, K146)

- U.S. broadcast desires to signal explicit output behavior for each picture in fixed-frame rate sequences
- Need top/bottom cadence signalled in frame sequences (currently only indicated in field sequences)

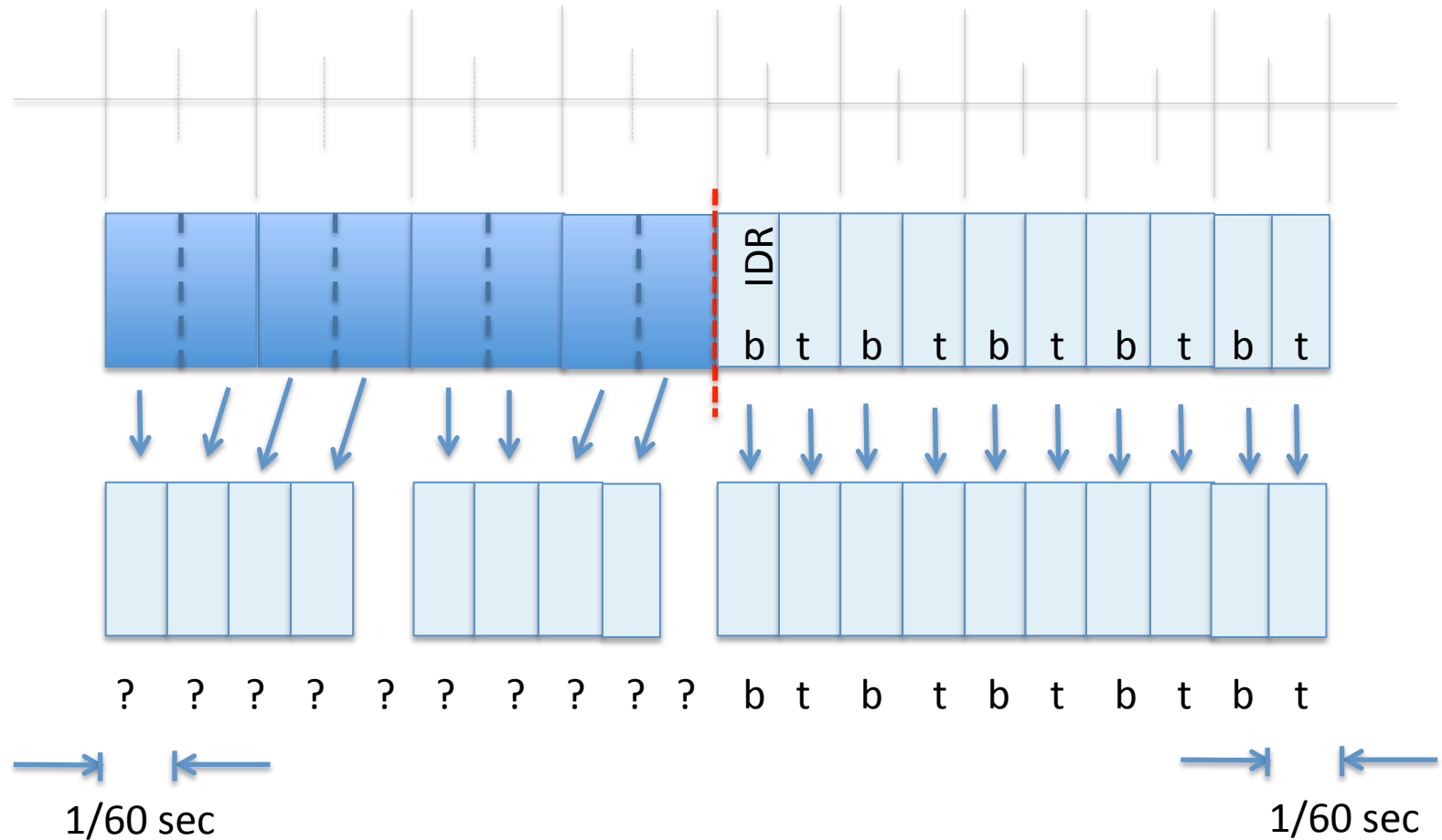
History of field sequences

- July 2011 Torino: request for interlace support
- Jan 2012 San Jose: BoG H720 field indication adopted into CD
- May 2012 Geneva: JCT amends to tree syntax (I393) to address confusion
- October 2012 Shanghai: several requests for change

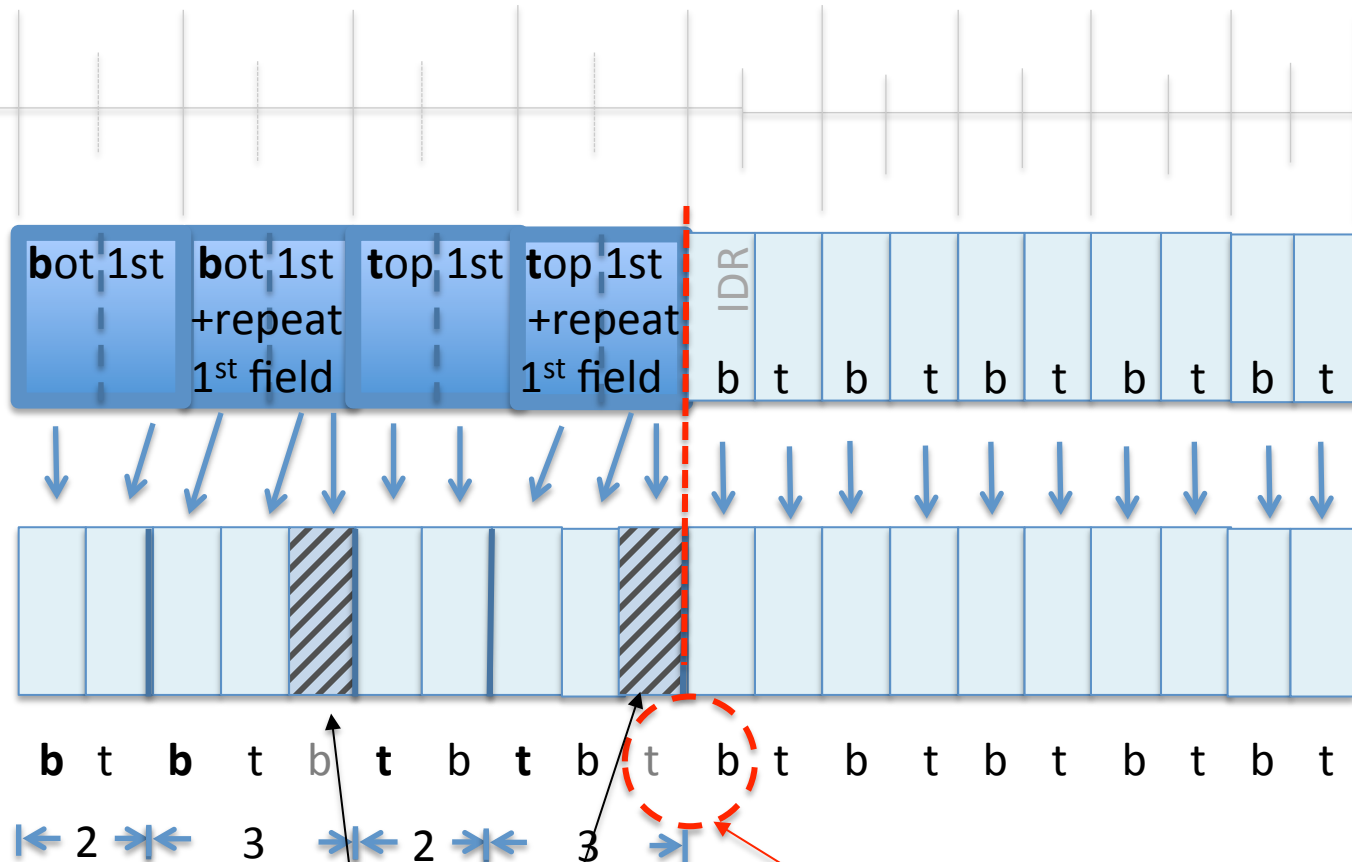
SAFF current practice



Output for fixed-frame rate



Generic solution: explicit instructions



Repeated
field

Cadence maintained across
SAFF transition

Current field indication SEI

field_indication(payloadSize) {	Descriptor
field_pic_flag	u(1)
progressive_source_flag	u(1)
duplicate_flag	u(1)
if(field_pic_flag)	
bottom_field_flag	u(1)
else if(!progressive_source_flag)	
top_field_first_flag	u(1)
else	
reserved_zero_1bit /* equal to 0 */	u(1)
reserved_zero_4bits /* equal to 0 */	u(4)
}	

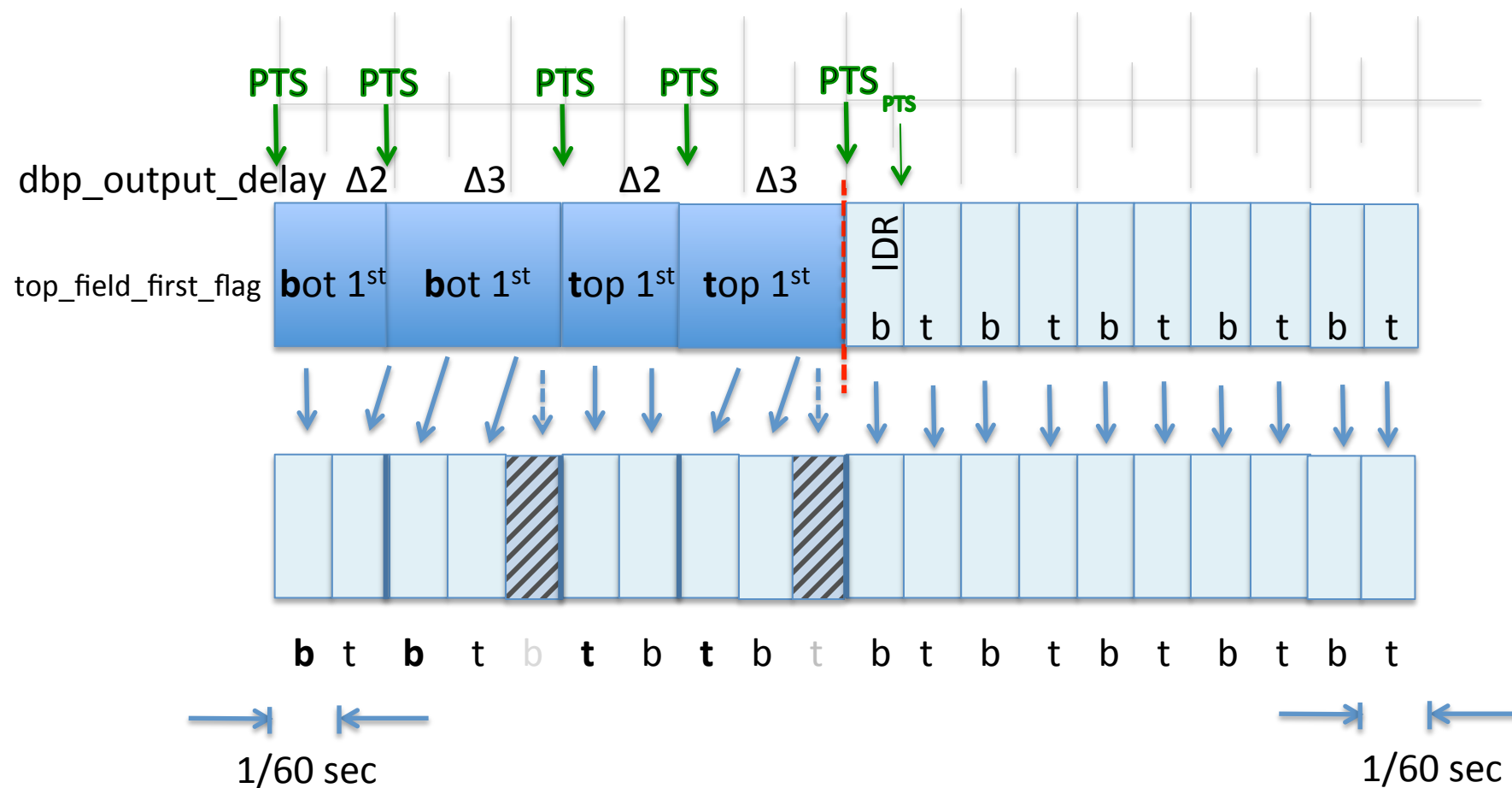
possible solutions

1. Return to Geneva April 2012 definition (JCTVC I0030) of independent flags
 - Change semantics to enable `top_field_first_flag` to signal cadence in frame sequences.
 - Rely on `dpb_output_delay` and PTS to determine output duration of each coded frame
2. Replace `field_indication()` SEI with `pic_struct()` SEI as per Broadcom (K146), or add Harmonic (K165)
 - `pic_struct` (+ `ct_type` + `n_frames`)
3. Amend `field_indication()` with `frame_display_pattern` and `field_display_pattern` (K160)
4. Indicate only the parity of the first picture following VUI

1. Return to Geneva style (I-0030)

field_indication(payloadSize) {	Descriptor
field_pic_flag	u(1)
progressive_source_flag	u(1)
bottom_field_flag	u(1)
top_field_first_flag	u(1)
duplicate_flag	u(1)
reserved_zero_3bits /* equal to 0 */	u(3)
}	

1. Geneva April 2012 (I-0030)



dpb_output_delay

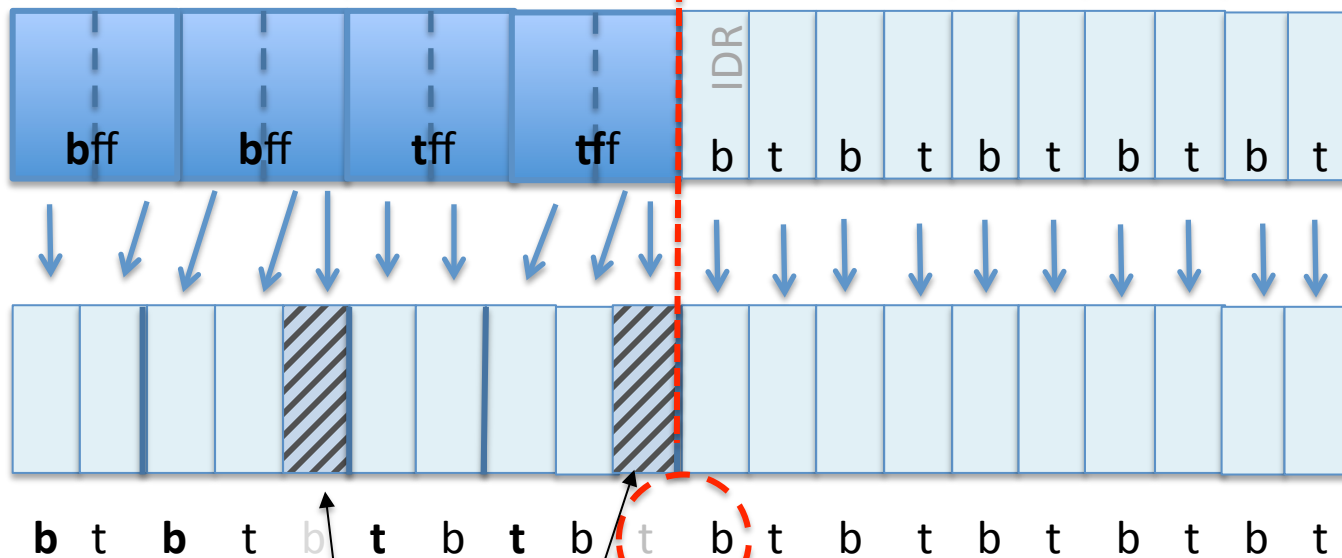
Δ dpb_delay_output ticks 1 tick = $\sim 1/60$ sec time_scale = 6000 num_units_in_tick=1001	Last field output parity	Current frame output pattern	Pic_struct equivalent
2	Top	Bottom, Top	4
	Bottom	Top, Bottom	3
3	Top	Bottom, Top, Bottom	6
	Bottom	Top, Bottom, Top	5
4	N/A	Frame doubling	7
6	N/A	Frame tripling	8

2. Add pic_struct (K146, K165)

	Descriptor
pic_timing(payloadSize) {	
if(CpbDpbDelaysPresentFlag) {	
cpb_removal_delay	u(v)
dpb_output_delay	u(v)
[..]	
}	
if (pic_struct_present_flag) {	
pic_struct	u(4)
}	
}	

pic_struct

pic_struct	4	6	3	5	2	1	2	1	2	1	2	1	2	1
ct_type	1	1	1	1	0	0	0	0	0	0	0	0	0	0
n_frame	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+10	+11	+12	+13
POC	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+10	+11	+12	+13



NumClockTS

← 2 → 3 ← 2 → 3 →

Repeated field

Cadence maintained across SAFF transition

3. Current field_indication SEI (K030)

field_indication(payloadSize) {	Descriptor
field_pic_flag	u(1)
progressive_source_flag	u(1)
duplicate_flag	u(1)
if(field_pic_flag)	
field_display_pattern	u(3)
else	
frame_display_pattern	u(3)
reserved_zero_2bits /* equal to 0 */	u(2)
}	

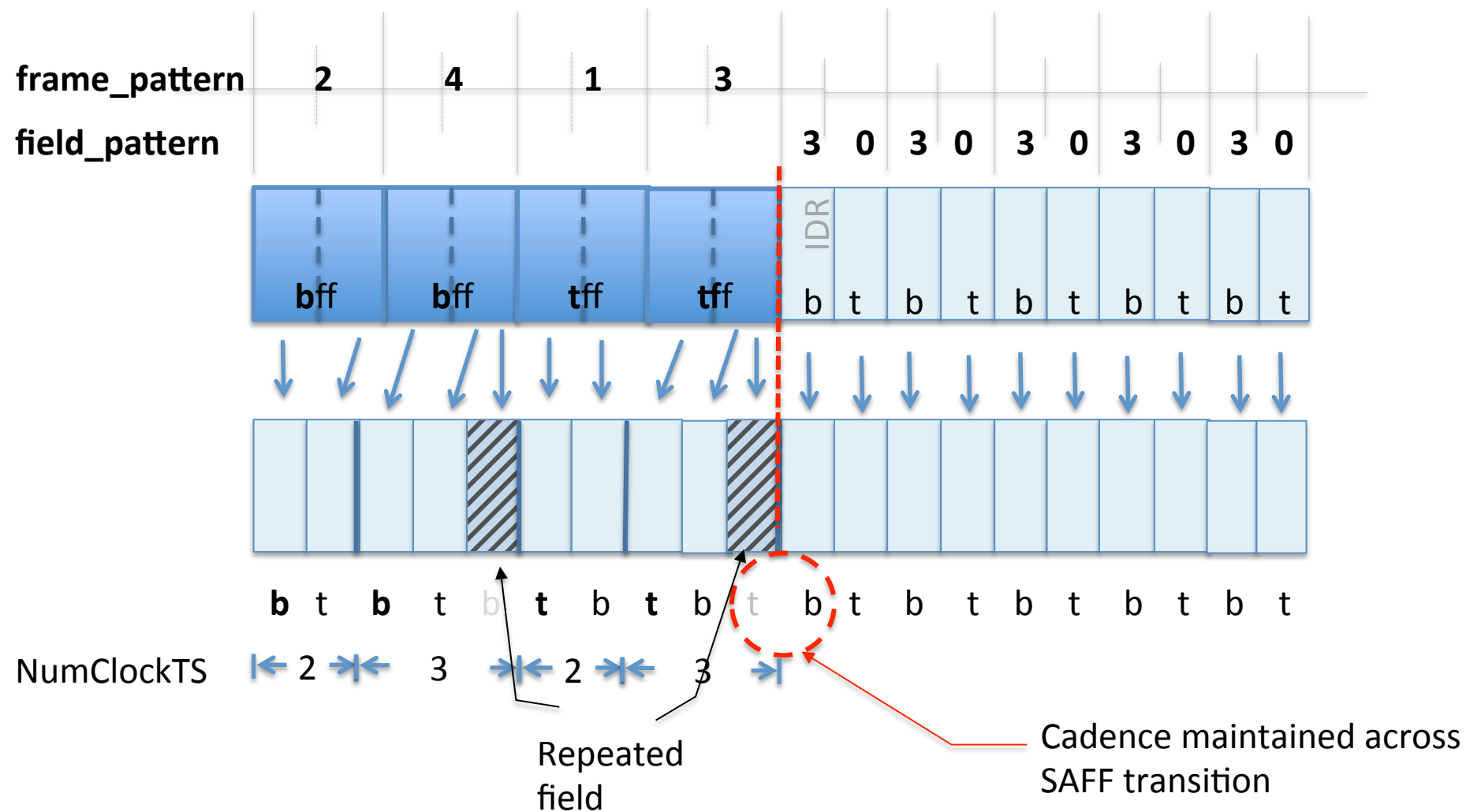
field_display_pattern

Value	Field display pattern
0	Top field associated with previous bottom field in output order
1	Bottom field associated with previous top field in output order
2	Top field associated with next bottom field in output order
3	Bottom field associated with next top field in output order
4	Top non-associated field
5	Bottom non-associated field
6	Reserved
7	Reserved

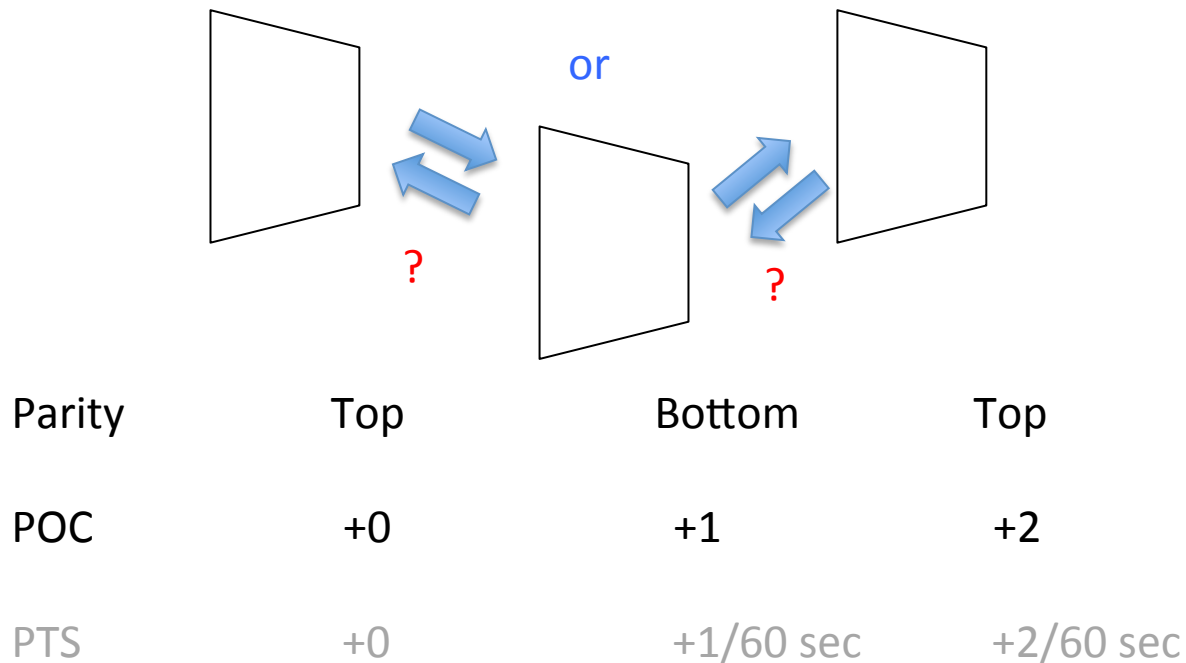
frame_display_pattern

Value	Frame display pattern	Notes
0	Progressive frame (no frame repetition)	progressive_source_flag shall be 1
1	Top field, then bottom field	
2	Bottom field, then top field	
3	Top field, then bottom field, then repeat top field	progressive_source_flag shall be 1
4	Bottom field, then top field, then repeat bottom field	progressive_source_flag shall be 1
5	Frame doubling	progressive_source_flag shall be 1
6	Frame tripling	progressive_source_flag shall be 1
7	Reserved	

_display_pattern



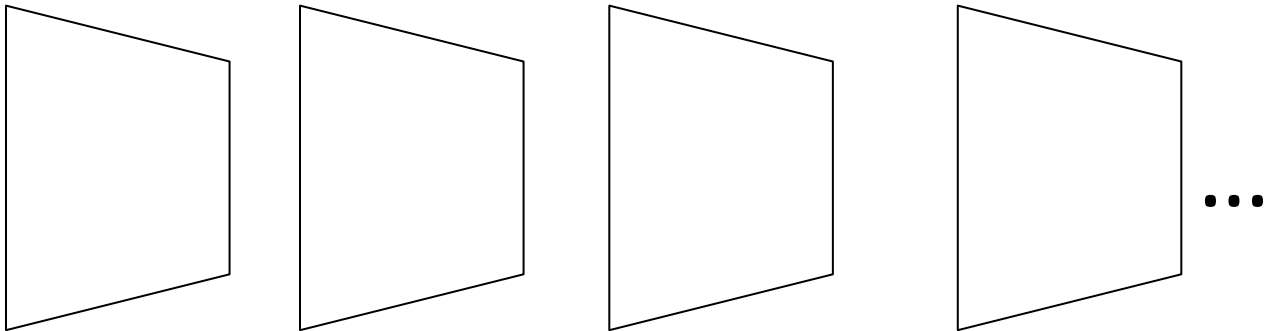
Progressive field pair association



3:2 pulldown 4 film frame → 10 output field epoch in a frame seq.

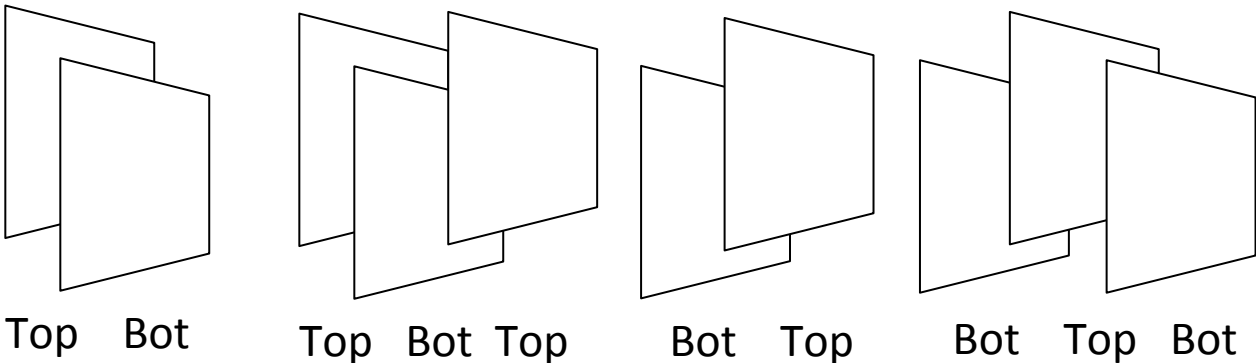
Encoded frame sequence

field_seq_flag=0

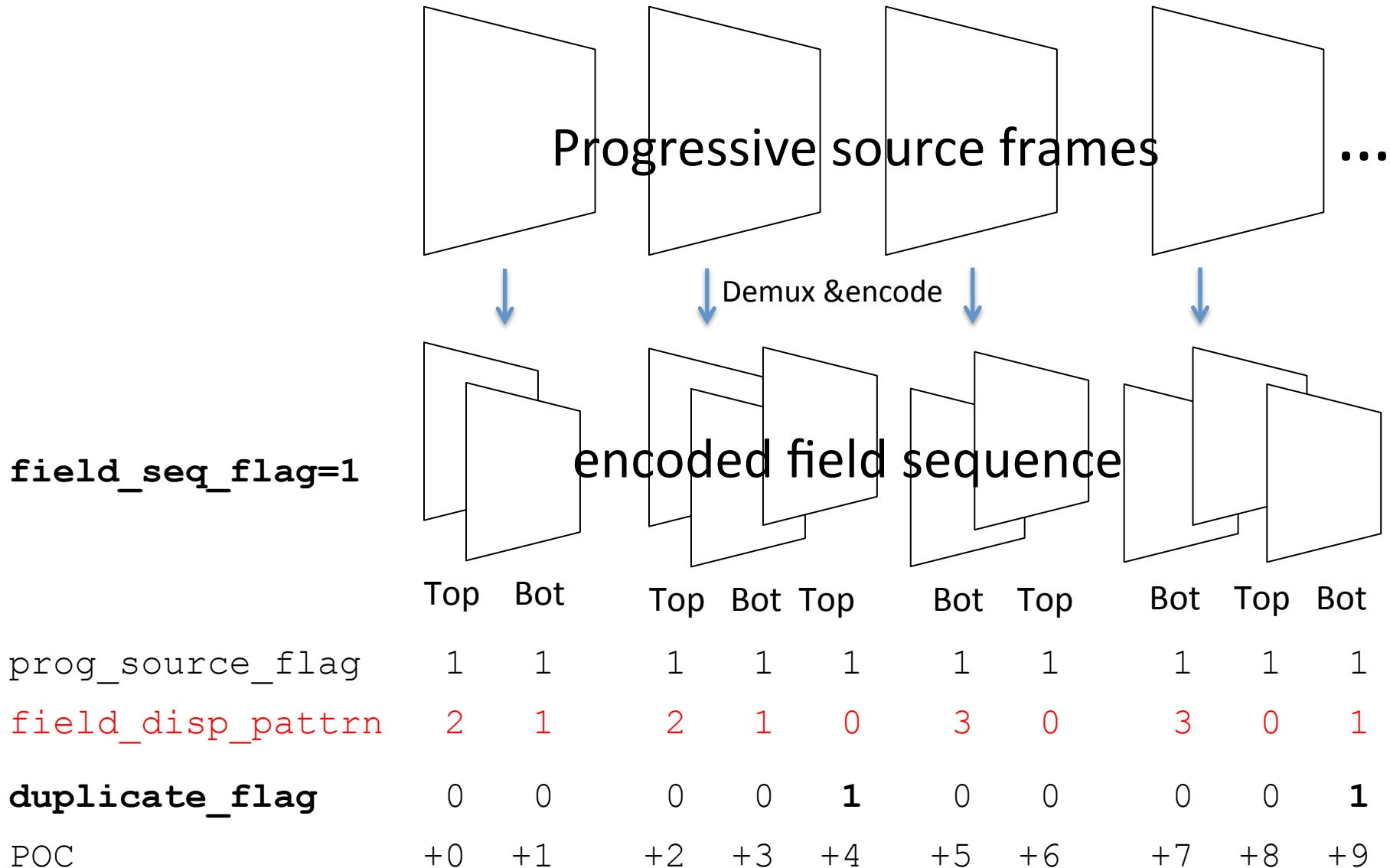


prog_source_flag	1	1	1	1
duplicate_flag	0	0	0	0
Frame_displ_patrn	1	2	3	4
POC (example)	+0	+2	+5	+7

Padded field display output (app. layer)



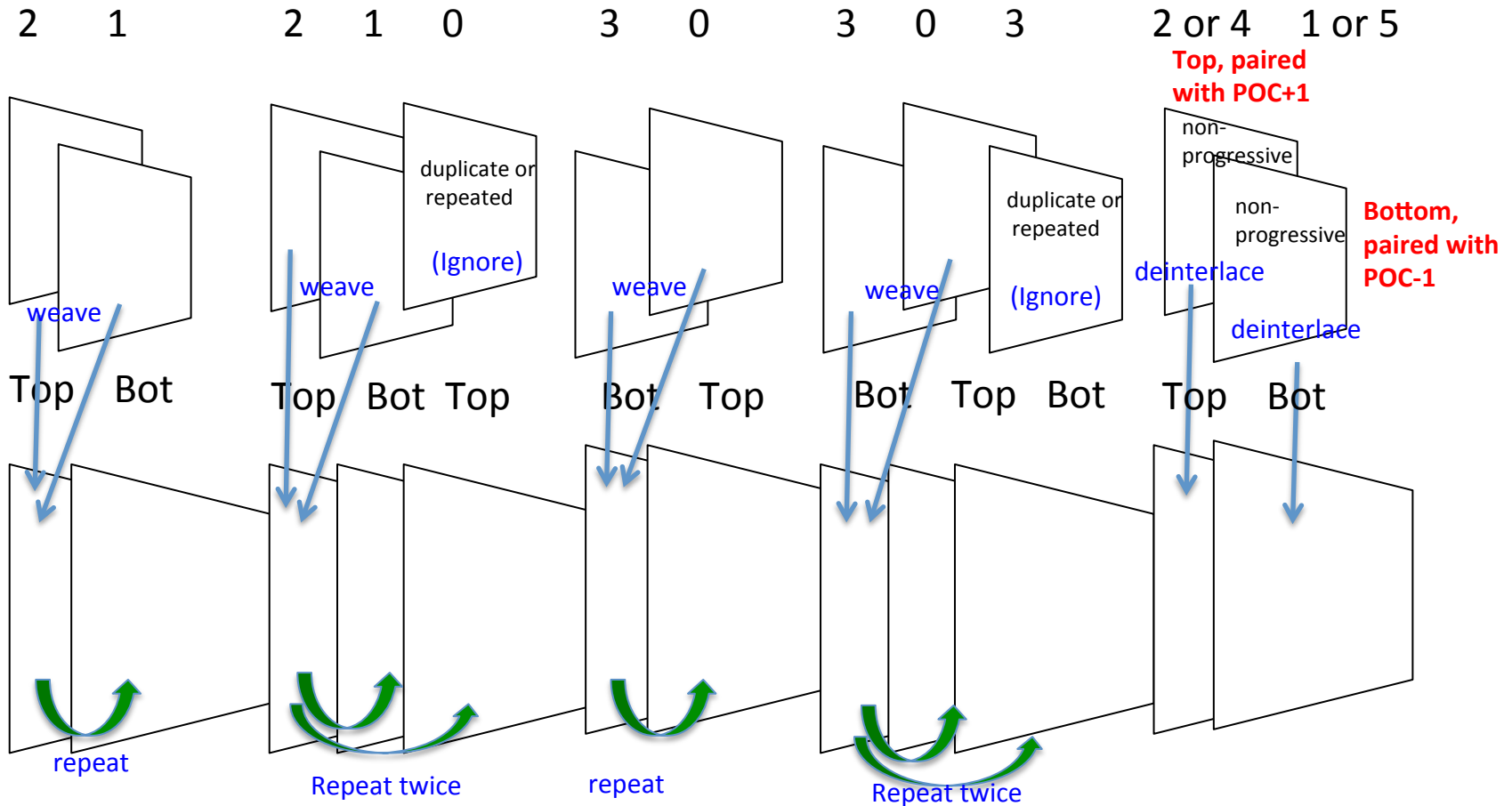
3:2 pulldown in a field sequence



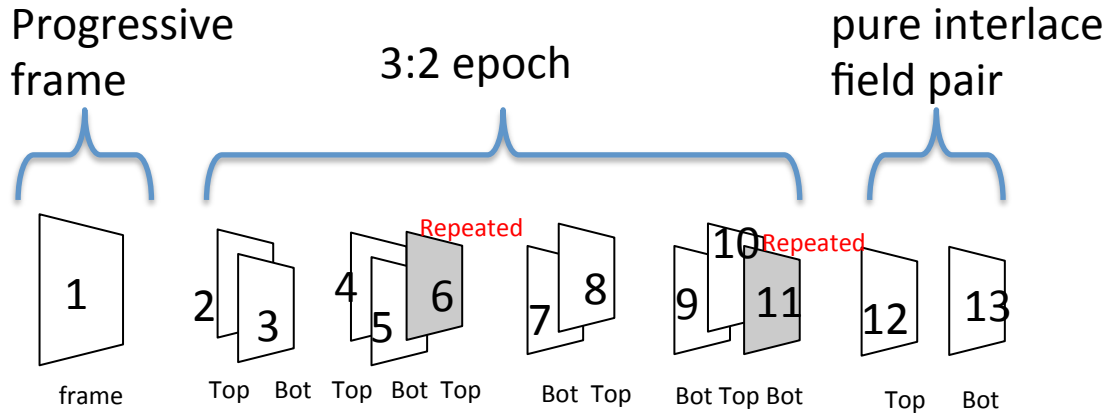
60 field/s output to 60 frame/s prog. display

1:1 correspondence between <= level 4.1 output fields and 1080v60frame/sec display output frames

field_display_pattern



Frame seq -> field seq → prog display

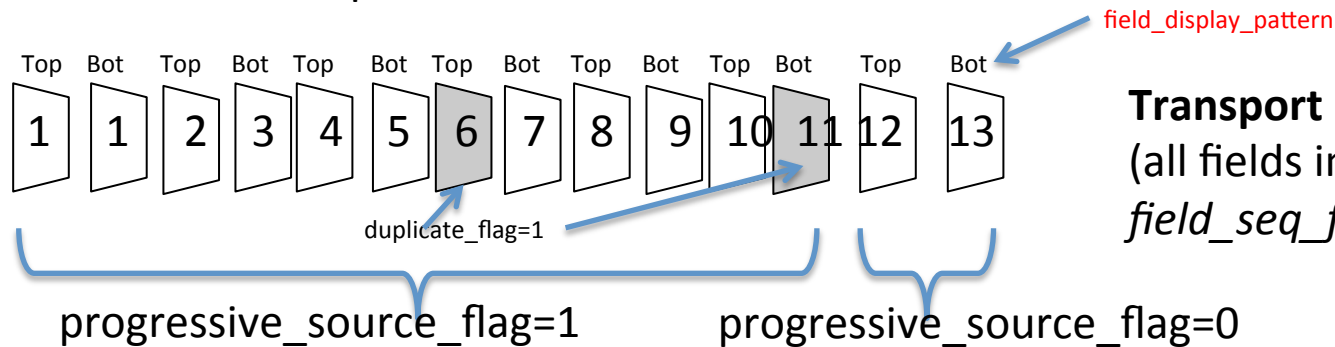


Source

(example: mixed content)

Picture numbers are to show mapping from source to field input to encoder.. It is not POC, frame_num, etc.

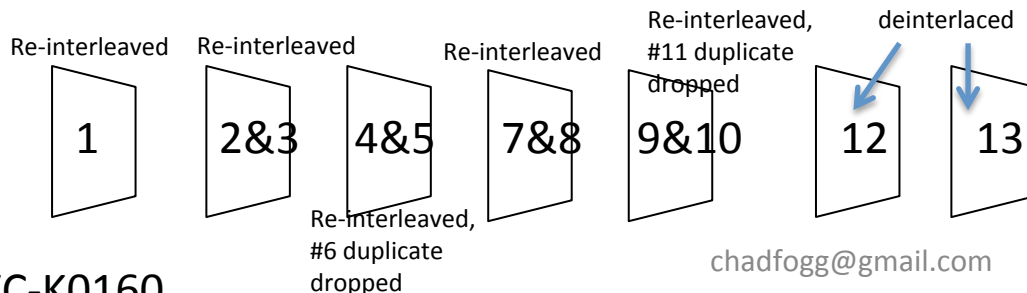
Pre-mux prior to HEVC encode



Transport & encoding

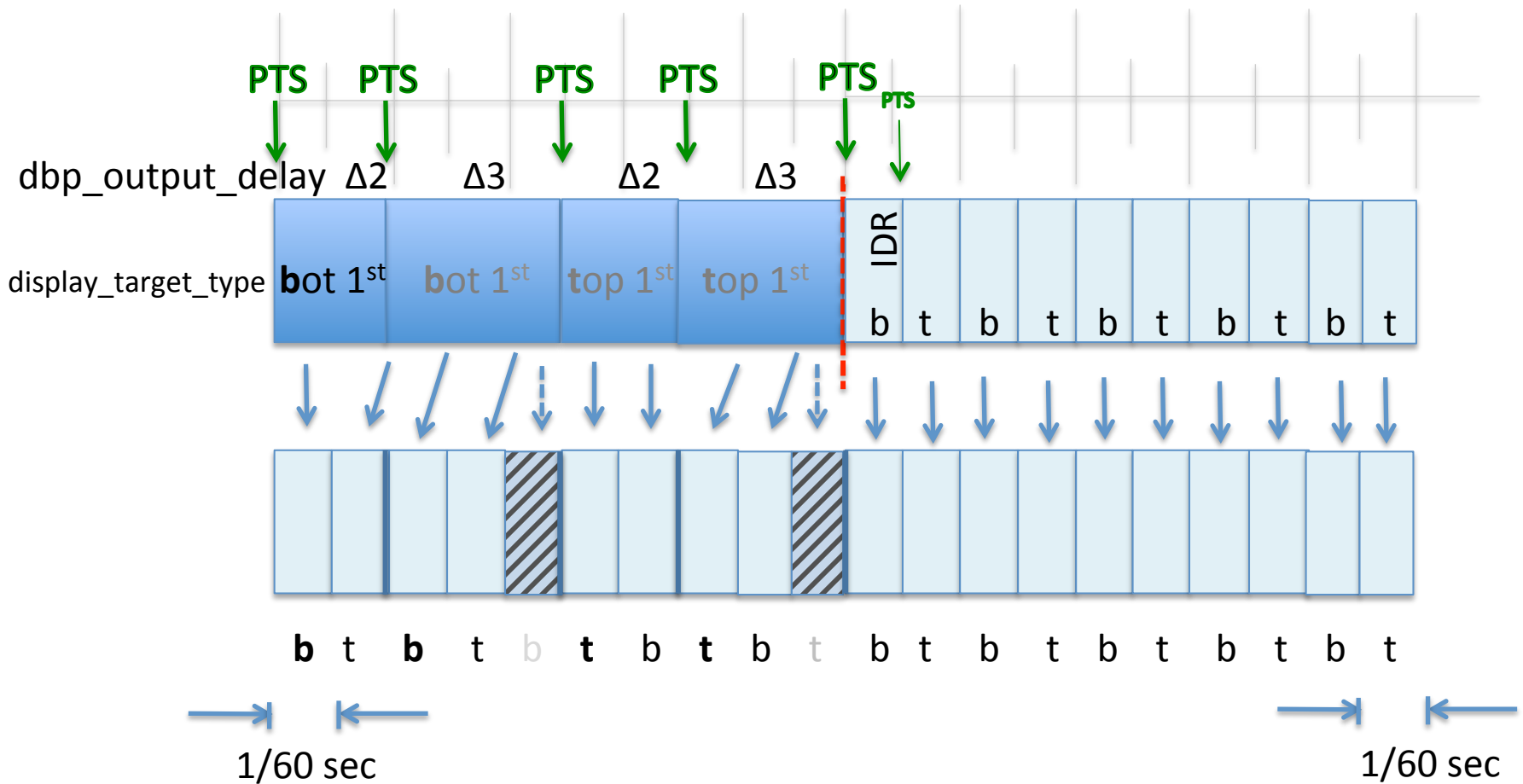
(all fields in a field sequence)

field_seq_flag=1



Progressive display

4. First field in sequence or follow VUI



4. dpb_output_delay

Δ dpb_delay_output ticks 1 tick = $\sim 1/60$ sec time_scale = 6000 num_units_in_tick=1001	Last field output parity	Current frame output pattern	Pic_struct equivalent
2	Top	Bottom, Top	4
	Bottom	Top, Bottom	3
3	Top	Bottom, Top, Bottom	6
	Bottom	Top, Bottom, Top	5
4	N/A	Frame doubling	7
6	N/A	Frame tripling	8

Comparison of proposed syntax

Proposal	Current method (K0030)	Pic_struct + modified field_indication SEI (K0165)	Field_display_pattern & frame_display_pattern (K0160)
Cadence	Only in field sequences (however Geneva I0030 can signal parity in both frames and fields)	pic_struct	field_display_pattern & frame_display_pattern
Frame output interval	PTS & dpb_output_delay	pic_struct	field_display_pattern & frame_display_pattern
Progressive field pair association	PTS	PTS	field_display_pattern

Comparison

Proposal	Current method (K0030)	Pic_struct + modified field_indication SEI (K0165)	Field_display_pattern & frame_display_pattern (K0160)
Advantage	Exists in HEVC spec Separates source and coding aspects from each other	Familiar to AVC developers Does not rely on PTS to indicate output duration	Fits into existing style Explicit progressive field pairing Puts all “interlace” info in one SEI
Disadvantage	Less explicit convention; relies on PTS (systems layer)	Requires both field_indication and pic_timing SEI's	It's new

Standards elements correspondence

	MPEG-2 H.262	AVC H.264	HEVC H.265
Frame rate signalling	frame_rate	fixed_frame_rate_flag num_units_in_tick	fixed_pic_rate_flag num_units_in_tick
Field from frame output order	top_field_first	pic_struct	display_pattern
Repetition control	repeat_first_field	pic_struct	display_pattern (frame seq.) duplicate_flag
Frame or field structured picture	pict_struct	field_pic_flag	field_seq_flag, field_pic_flag
Prog / interlace coding tool switch at seq. layer	progressive_sequence	frame_mbs_only_flag	N/A
Progressive display	progressive_sequence	??	display_target_type
Content type for current picture	progressive_frame	ct_type	progressive_source_flag

pic_struct vs. display_output_indication SEI

AVC-style

display_output_indication

Pic_struct	Output order	Num ClocksTS	ct_type	Field pic flag	Progressive source flag	Field displ pattern	Frame dis. pattern
0	Prog. frame	1	(0)	0	1	-	0
1	Top		0	1	1	0,2	-
			1		0	4	
2	Bot		0		1	1,3	
			1		0	5	
3	Top,bot	2	0		0	1	
4	Bot,top		1	0		2	
			0	1			
			1	0			
5	Top,bot,top	3	(0)	1		-	3
6	Bop,top,bot						4
7	Frame doubling	2					5
8	Frame tripling	3					6

AVC, MPEG-2, HEVC translation

Output pattern	AVC			MPEG-2					HEVC (proposed)			
Output order	Pic_struct	NumClockTS	ct_type (0=prog, 1=interlace)	Top_field_first	Repeatt first fld	Pict_struct	Prog_frame	Prog_sequence	Field_pic_flag	Prog_src_flag	field_dis_patrn	frame_dis_ptrn
Prog. frame	0	1	(0)	-	-	11	1	0,1	0	1	-	0
Top field (progressive)	1	1	0	-	-	01	1	0	1	1	0,2	-
Top field (interlaced)	1	1	1	-	-	01	0	0	1	0	4	-
Bottom field (progressive)	2	1	0	-	-	10	1	0	1	1	1,3	-
Bottom field (interlaced)	2	1	1	-	-	10	0	0	1	0	5	-
Top, bottom (progressive)	3	2	0	1	0	11	1	0	0	1	-	1
Top, bottom (interlaced)	3	2	1	1	0	11	0	0	0	0	-	1
Bottom, Top (progressive)	4	2	0	0	0	11	1	0	0	1	-	2
Bottom, top (interlaced)	4	2	1	0	0	11	0	0	0	0	-	2
Top, bottom, top	5	3	(0)	1	1	11	1	0	0	1	-	3
Bottom, top, bottom	6	3	(0)	0	1	11	1	0	0	1	-	4
double	7	2	(0)	0	1	11	1	1	0	1	-	5
triple	8	3	(0)	1	1	11	1	1	0	1	-	6