Сжатие видео
H.264

Дмитрий Ватолин

Московский Государственный Университет
CS MSU Graphics&Media Lab

Version 1.1

Структура материала

- Введение
- Компенсация движения
- Преобразование
- Квантование
- Фильтрация
- Перестановка
- Entropy coding
- NAL

Эволюция стандартов

GRAPHICS & MEDIA LAB
VIDEO GROUP

Стан-т	Год	Пр-тель	Что нового
H.261	1993	ITU-T	16/16 целочисленная компенсация движения 8×8 DPCM/DCT, преобразование, VLE кодирование
MPEG-1	1993	ISO/IEC	Bi-предсказание, half- компенсация движения
MPEG-2	1997	ISO/IEC ITU-T	Выбор качества, Profiles и Levels, Interlaced video
H.263	1998	ITU-T	Поддерживает передачу данных ниже 30 кбит/сек, цепи и сети с пакетной коммуникаций P-slice могут включать I-блоки
H.263+, H.263++ H.26L	2001 2001 2002	ITU-T	1) Поддерживает все скорости передачи информации в битах, большее количество опций 2) Акцент на погрешностях 3) Long-term и дальнейшее развитие до H.264
MPEG-4	1998	ISO/IEC	
H.264	2003	ISO/IEC ITU-T	10-ая часть MPEG-4, см далее

Список сокращений

GRAPHICS & MEDIA LAB
VIDEO GROUP

MPEG- Moving Picture Experts Group

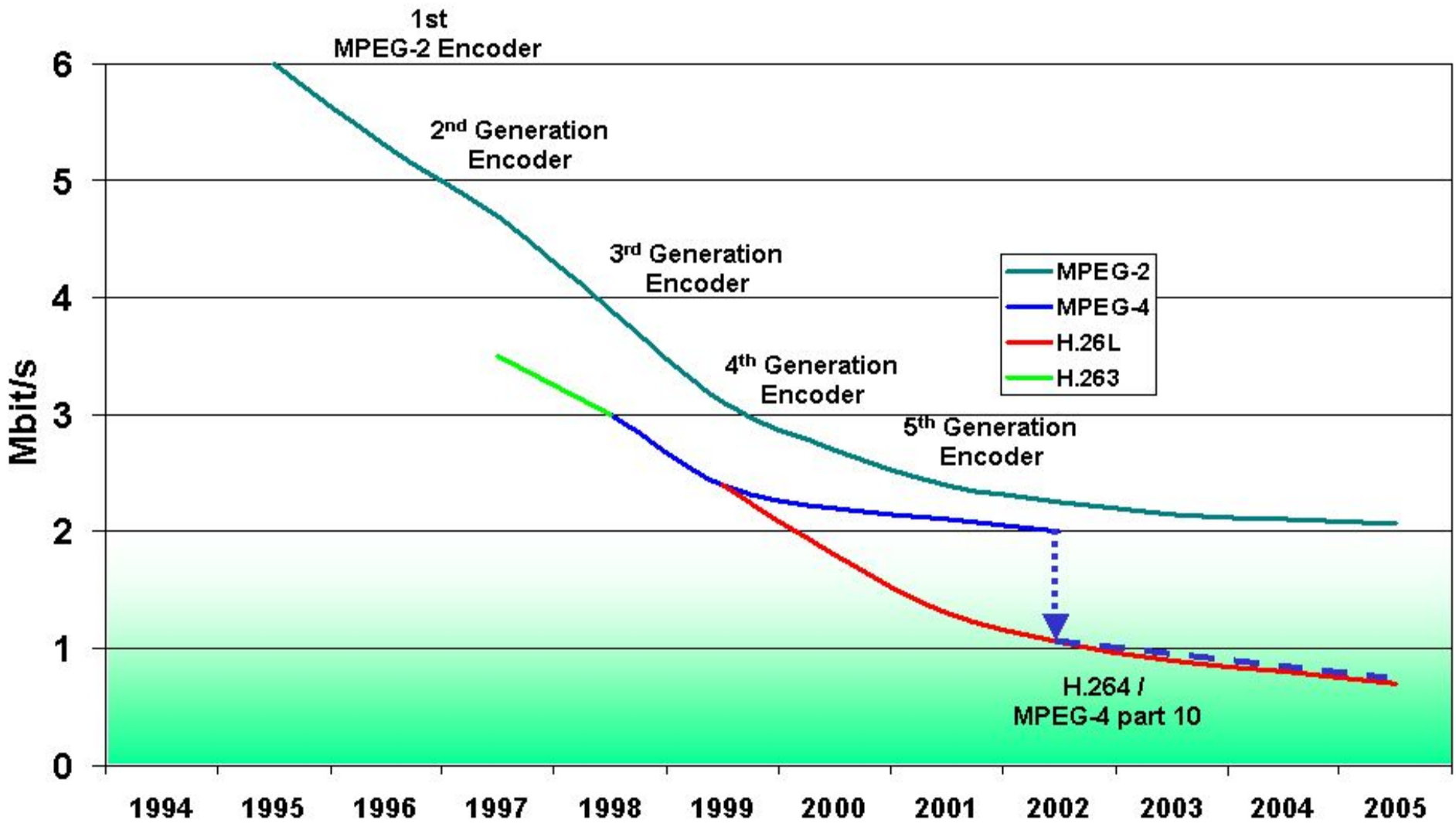
ISO - International Standards Organisation

VCEG - Video Coding Experts Group

IEC- International Electrotechnical Commission

ITU - International Telecommunications Union

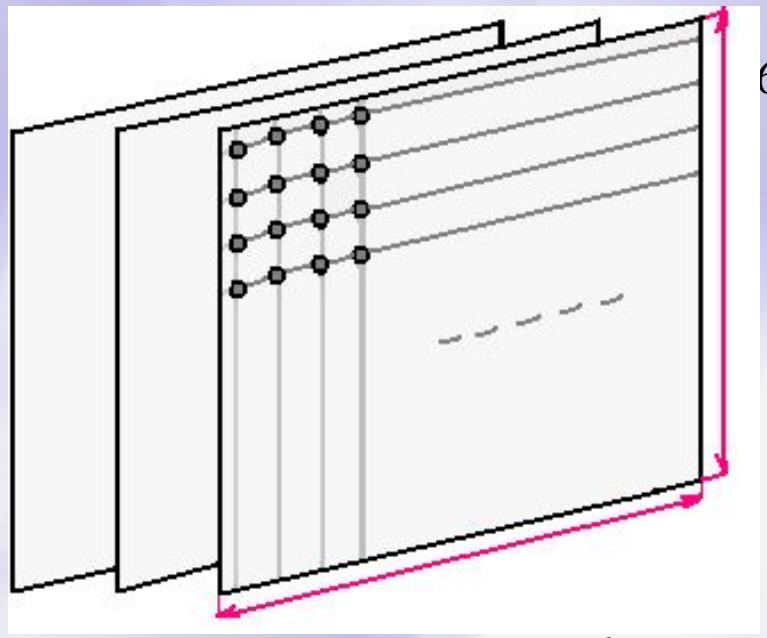
Эволюция стандартов



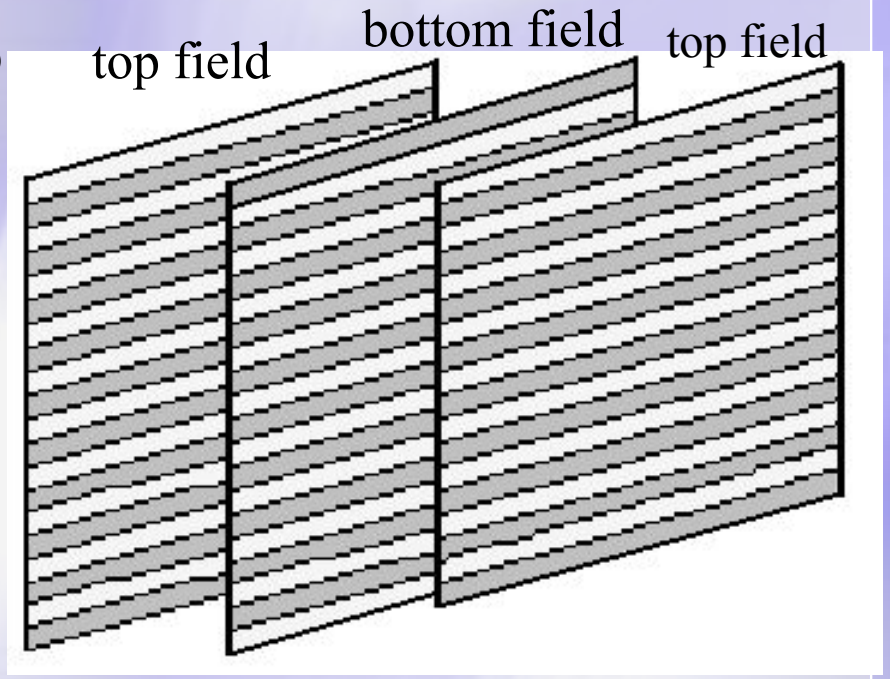
Новое в H.264

- Многорежимная компенсация движения
- 1/4-, 1/8-компенсация движения
- B-предсказание
- Целочисленное преобразование над блоками 4×4
- Многорежимное intra-предсказание
- Deblocking-фильтр
- Entropy-кодирование
- NAL (Network Abstraction Layer)
- SP, SI -slice

H.264 / используемые форматы видео



6b

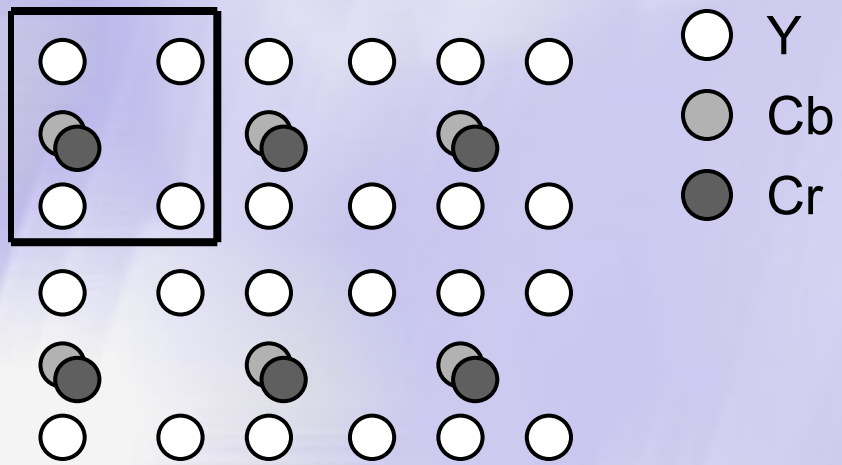


progressive video

- Кодируются отдельные изображения (фреймы)
- H.264 поддерживает progressive и interlaced форматы видео
- Размер изображений должен быть кратен 16 по обоим измерениям

H.264 / формат изображения

Изображение представлено в формате YCbCr 4:2:0



Y=0.299R+0.587G+0.115B

Cb=0.564(B-Y)

Cr=0.713(R-Y)

Изображение разбивается на так называемые макроблоки размера 16 x 16 по яркостной компоненте и, соответственно, размера 8x8 для цветовых компонент

H.264/ Термины

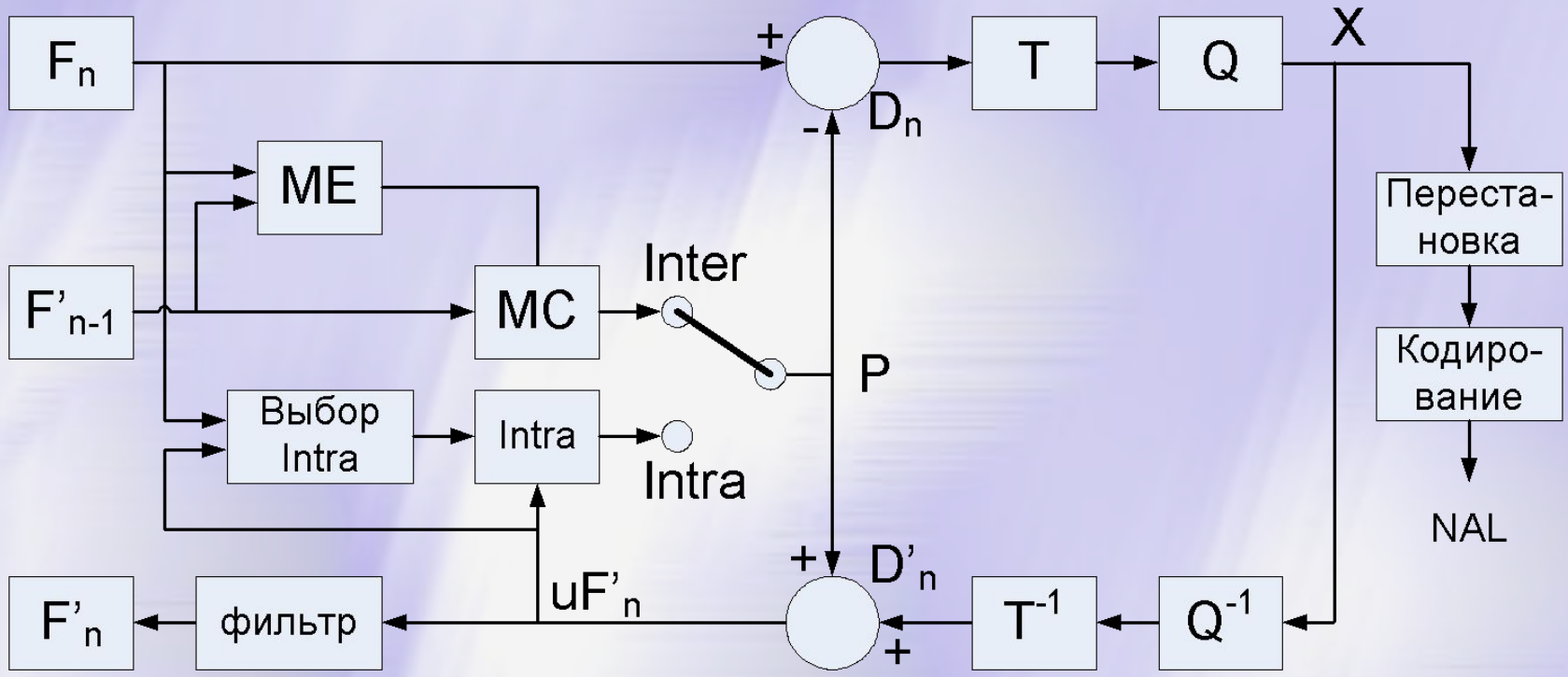
Макроблок A *предсказывается* макроблоком B, если из всевозможных макроблоков из предшествующих фреймов, разница между блоками B и A минимальна.

Эта разница называется компенсацией движения (motion compensation)

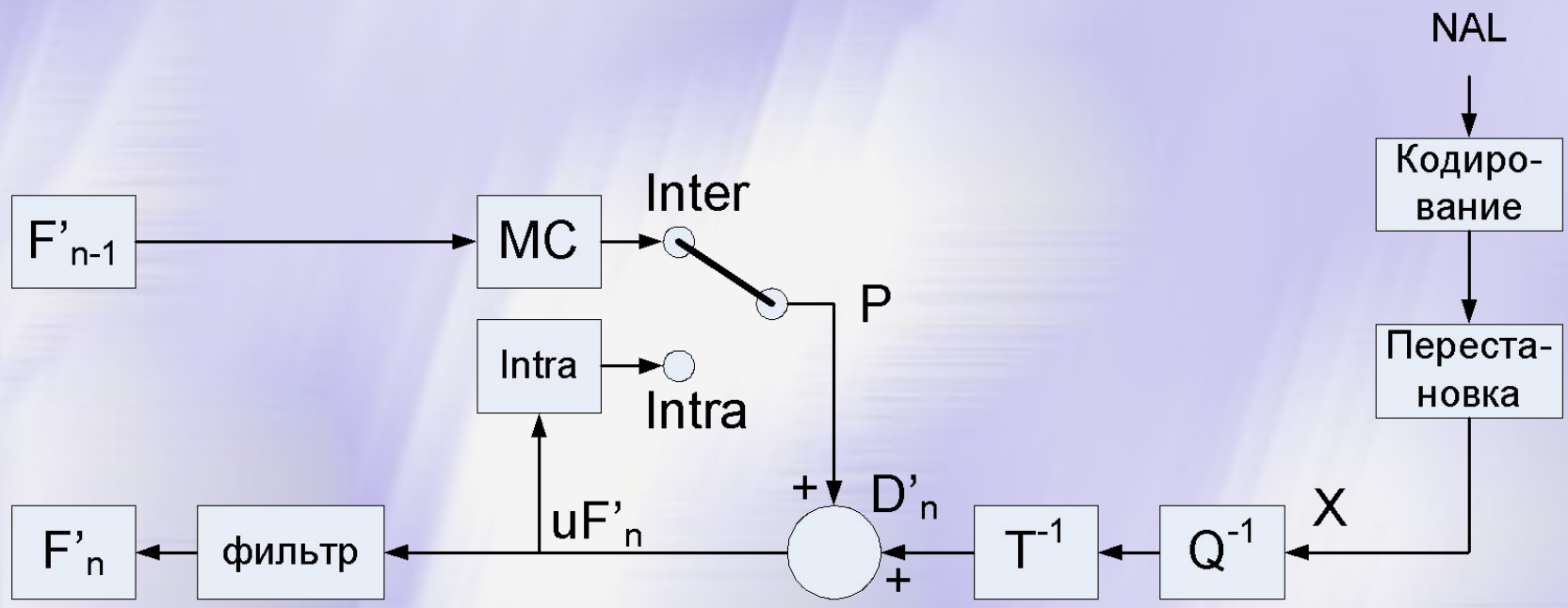
***Intra – кодирование* – кодирование без использования компенсации движения**

***Inter – кодирование* – кодируется с использованием компенсации движения**

H.264 Кодирование



H.264 Декодирование



H.264 / Кодирование-декодирование

F_n - текущий фрейм, множество макроблоков, каждый из которых кодируется в intra или inter режимах

P - предсказание

uF'_n - восстановленный, но без фильтра, текущий фрейм

F'_{n-1} - один или два ранее закодированных фрейма

D_n - разница между блоком и его предсказанием

H.264 / Кодирование-декодирование

GRAPHICS & MEDIA LAB
VIDEO GROUP

ME - motion estimate, поиск наиболее подходящего предсказания

MC - motion compensation, компенсация движения (вычисление векторов движения)

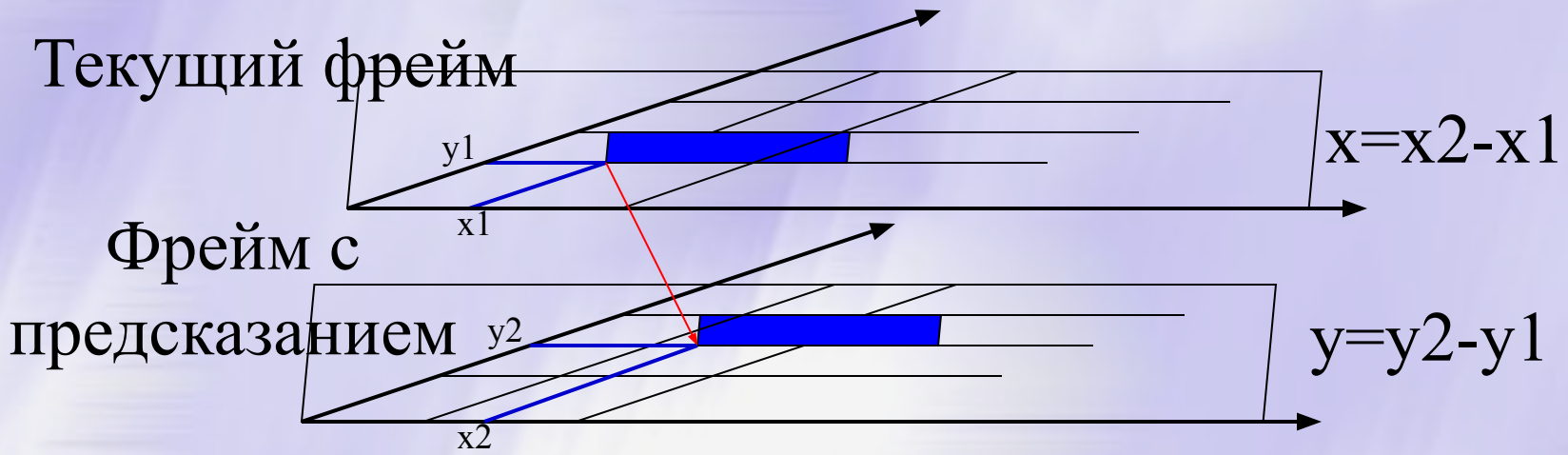
T - transformation, преобразование

Q - квантование

Entropy Coding – непосредственно кодирование

NAL - Network Abstraction Layer , формат для передачи по различным каналам

Компенсация движения / Inter-блоки



При кодировании *inter*-(макро)блока возникают вектора движений

Если вектор движения по яркостной компоненте Y равен (x,y), то соответствующий вектор по Cr и Cb будет равен (x/2,y/2)

Компенсация движения / Slices

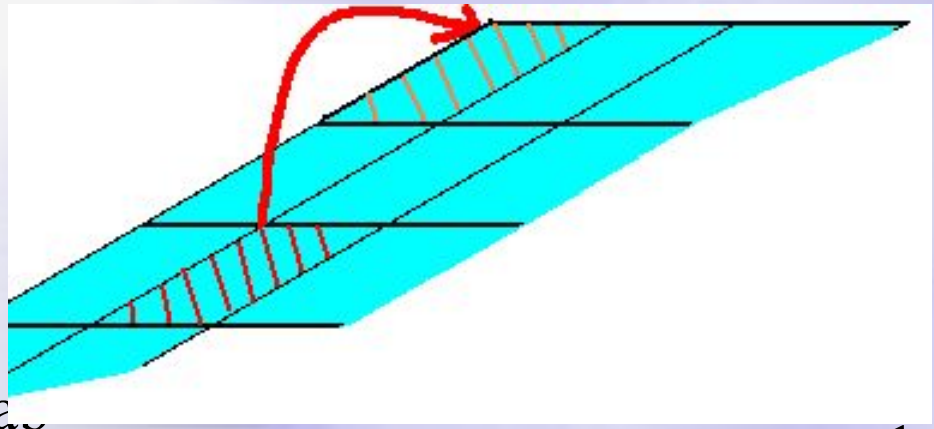
Каждый фрейм представляется как одна плоскость (slice) или несколько видеоплоскостей.

Типы slice: I, P, B, SP, SI

Компенсация движения / I-slice

<u>I-slice</u> – плоскость, состоящая исключительно из intra-макроблоков (первый фрейм всегда является I-slice)

Макроблоки могут ссылать только на уже закодированные макроблоки из той же slice

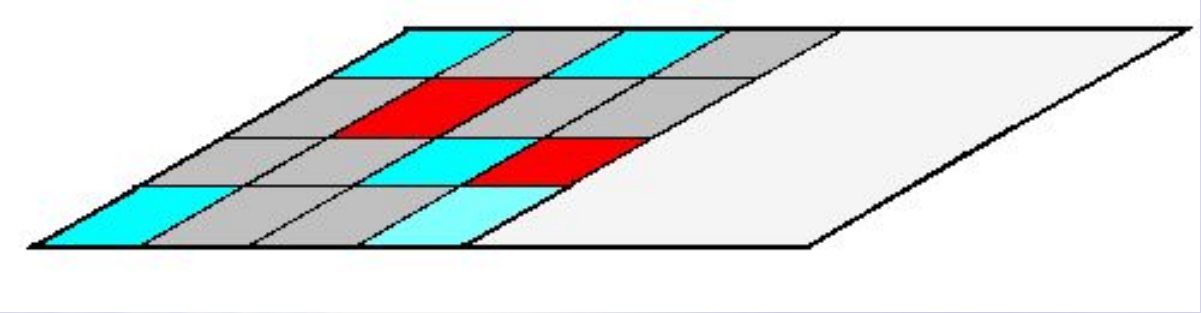


Компенсация движения / P-slice

intra

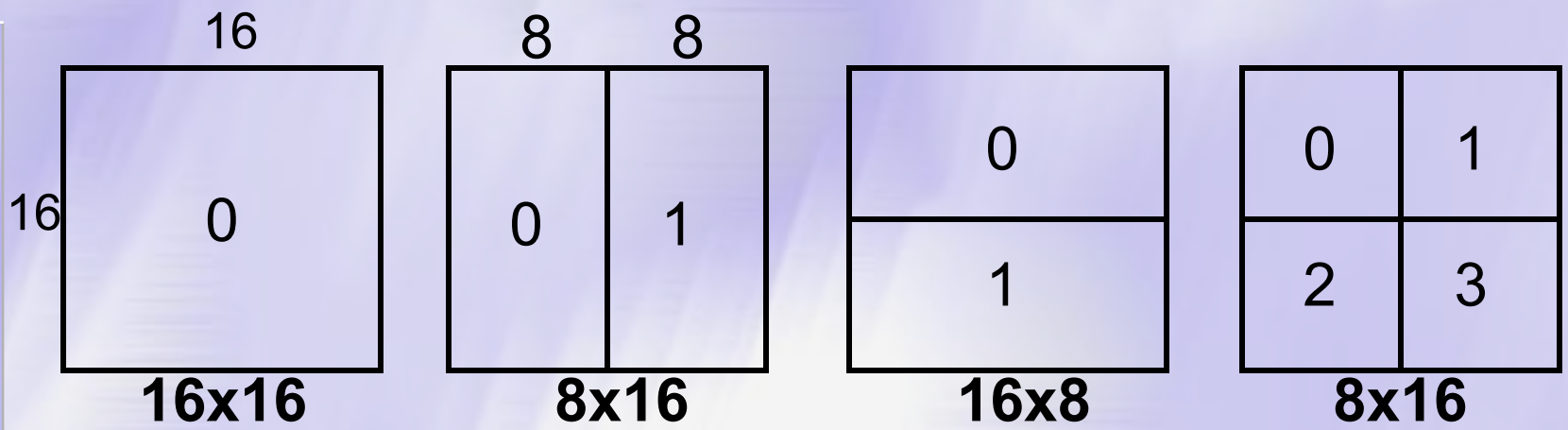
inter

Skipped

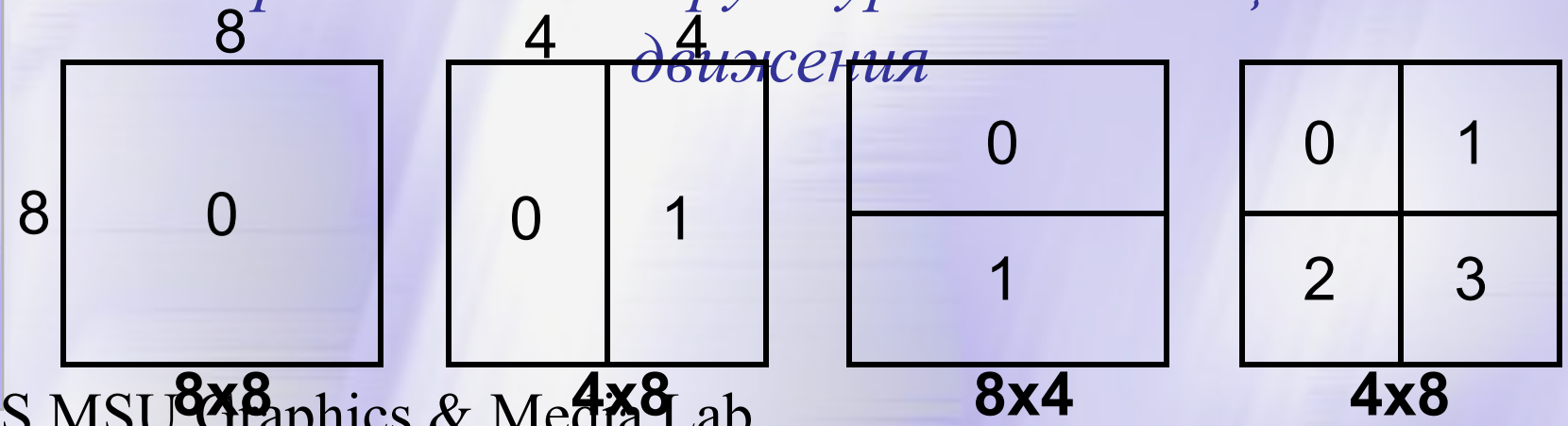


Для пропускаемых макроблоков посылается только сигнал о том, что он пропускаемый и больше никаких данных

Компенсация движения / Разбиение макроблока



древовидная структура компенсации движения



Компенсация движения / SP and SI slices

Основная идея: SP (switching P) и SI (switching I) slices – это кодирование ДОПОЛНИТЕЛЬНЫХ кадров, дающих В ТОЧНОСТИ такой же результат, как и основные кадры в потоке.

Компенсация движения / SP and SI slices

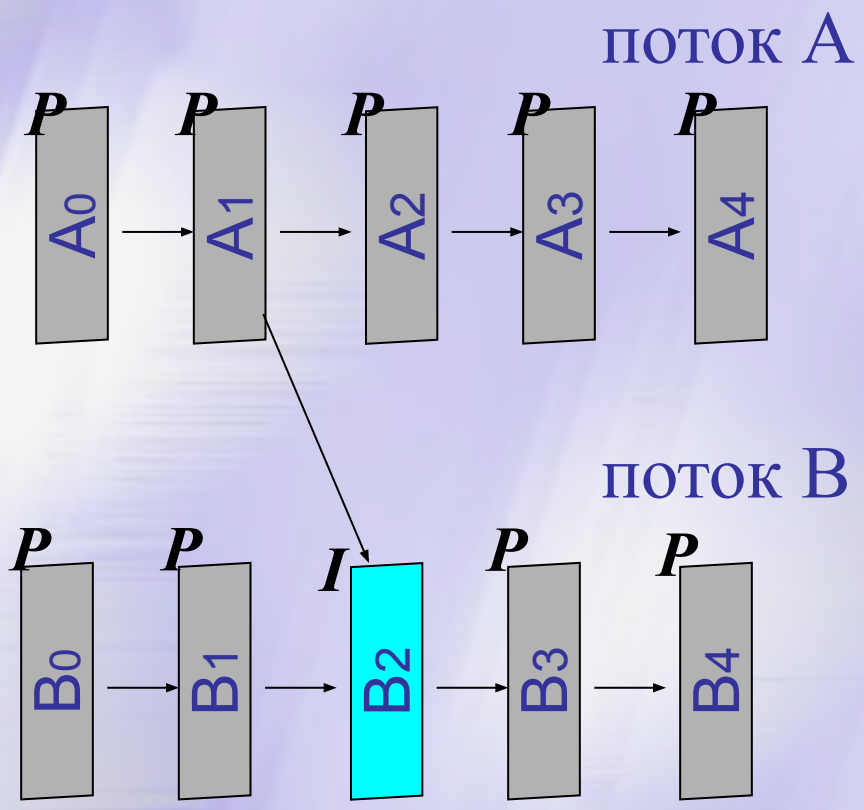
SP (switching P) и SI (switching I) slices могут использоваться для:

- обеспечения устойчивости к ошибкам
- для переключения между различными потоками (когда передается, например, стереопоток)
- для fast forward.

Компенсация движения / SP-slice

Использование SP/SI-slices для переключения между потоками:

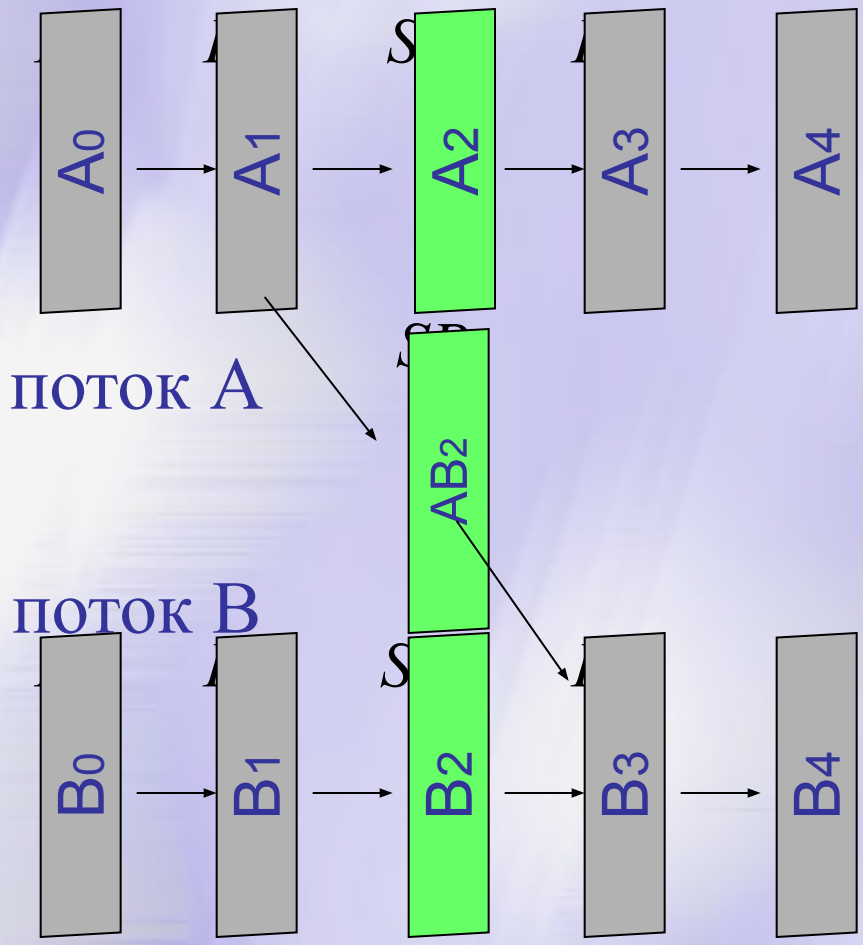
В стандартах до H.264 возникала “switch point”. При этом образуется I-slice что приводит к увеличению числа данных в каждой такой точке



Компенсация движения / SP- and SI-slices

SP-slices (AB_2 и B_2) используются для кодирования одной и той же плоскости (B_3), ссылаясь на два фрейма (A_1 и B_1) из разных потоков.
(аналогично строится BA_2)

Из двух SP-slice получаются одни и те же данные.



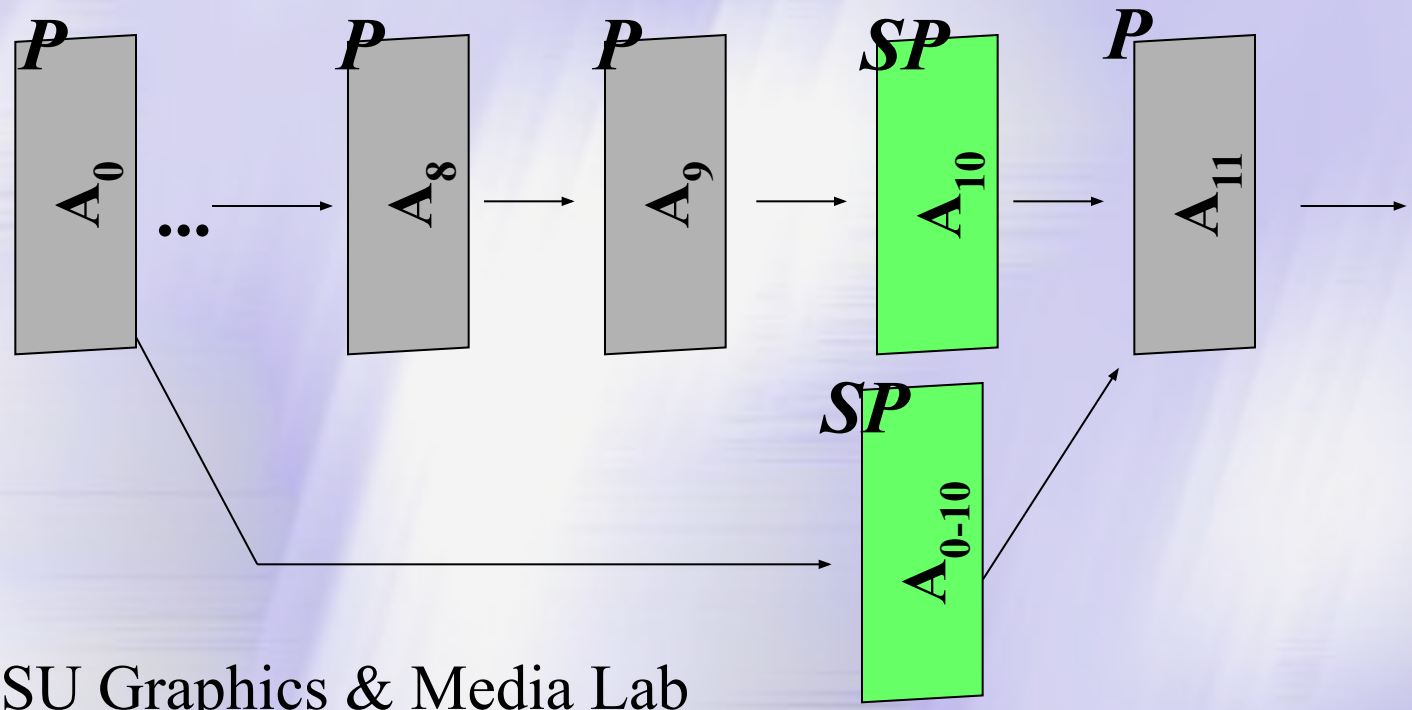
Компенсация движения / SP- and SI-slices

Устойчивость к ошибкам за счет использования SP/SI slices:

- Кодировщик кодирует SP-slices регулярными интервалами. Для каждой SP-slice имеется еще по крайней мере одна SP-slice, которая использует другой ссылочный фрейм или SI-slice.
- Если во время передачи фрейм поврежден, получатель может информировать передатчик через обратную связь, и передатчик пошлет SP-slice, использующую другой ссылочный фрейм или SI-slice.

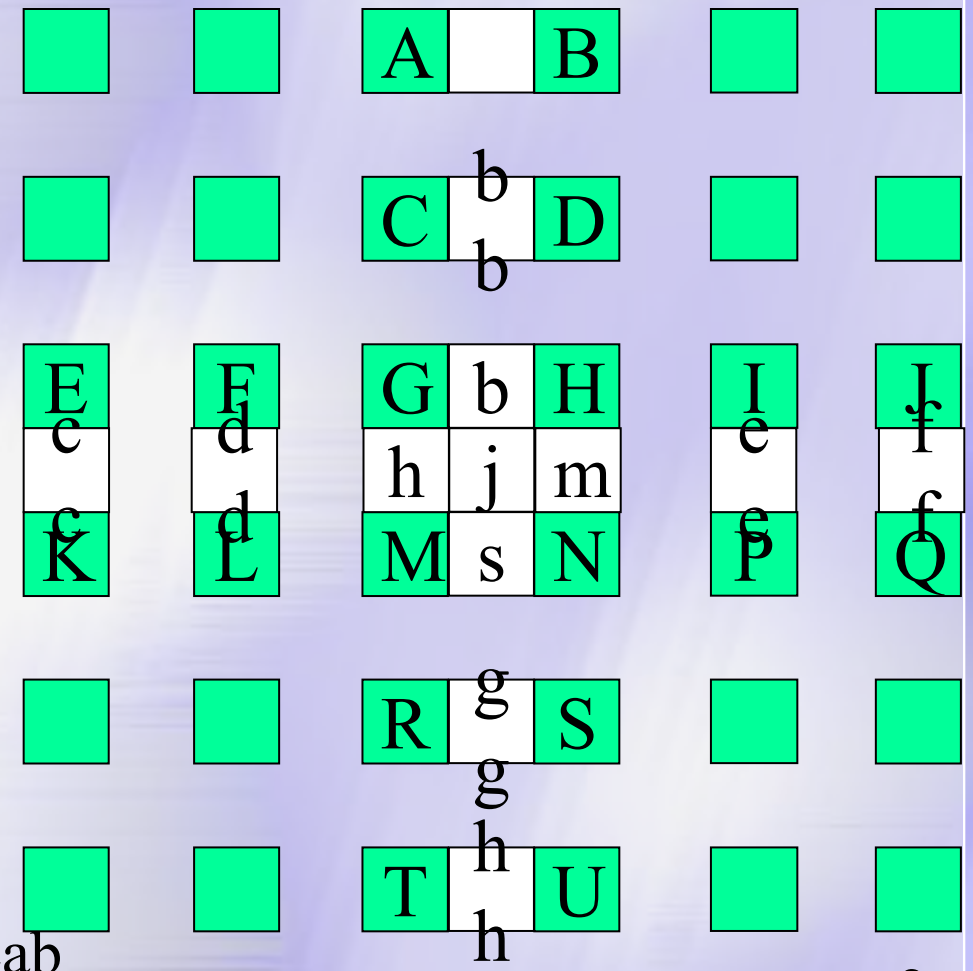
Компенсация движения / SP and SI slices

fast forward (основная идея: A_{11} полностью одинаков после $A_{0\text{-}10}$ и после A_{10})



Компенсация движения / Интерполирование

- Увеличение макроблока возможно в 2 и 4 раза.
- Используется FIR (Finite Impulse Response)



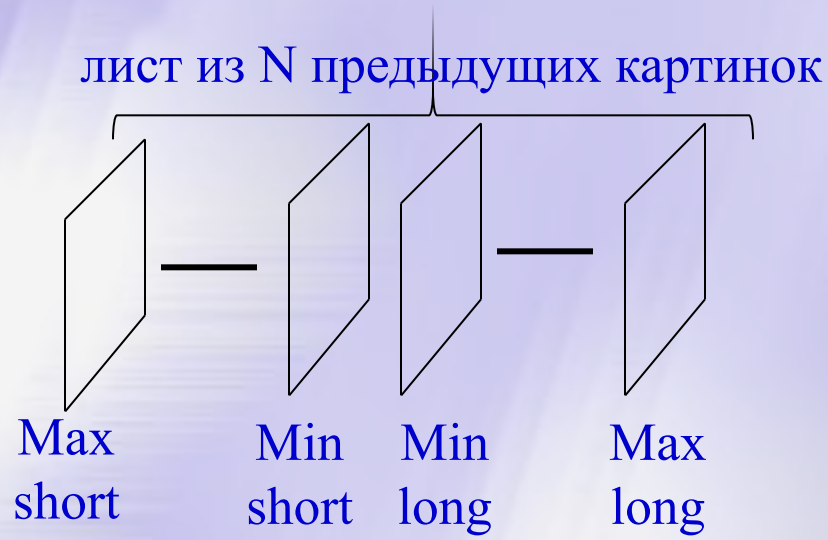
$$\mathbf{b} = \text{round}((E - 5F + 20G + 20H - 5I + J)/32)$$

$$\mathbf{j} = \text{round}((h + m)/2)$$

Компенсация движения / short и long terms

После того, как фрейм закодирован (декодирован), шифратор (дешифратор) может присвоить ему один из следующих статусов:

a) недоступен для использования
b) short term
c) long term
d) готов к выводу на экран



short term → ***long term***

Adaptive memory control command

long term → ***недоступен для использования***

Компенсация движения / список0 и список1

long и short terms выстраиваются в два списка:

список 0 - фреймы с номерами меньше текущего по убыванию,затем - больше текущего, по возрастанию

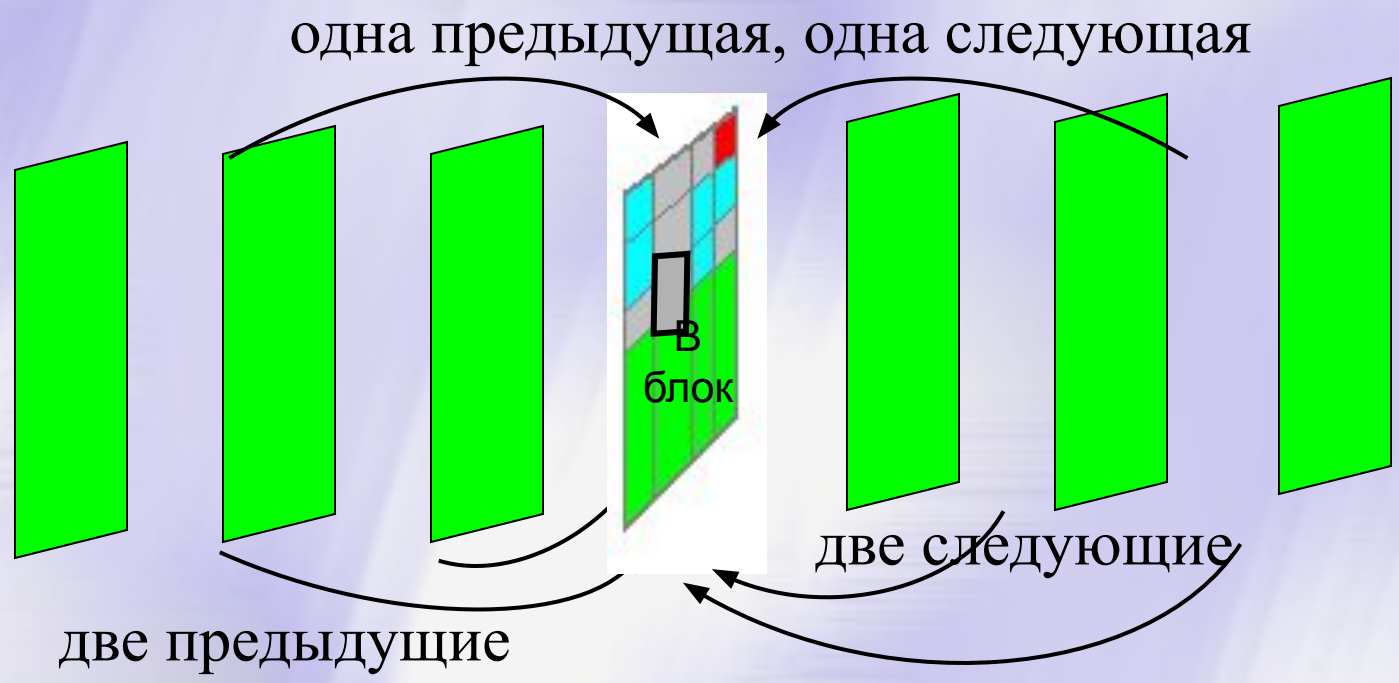
список 1 -фреймы с номерами больше текущего, по возрастанию, затем- меньше текущего по убыванию

индекс	список0	список1
0	126	128
1	125	129
2	123	130
3	128	126
4	129	125
5	130	123

Пример: имеются terms 123,125,126,128,129,130

текущий фрейм – номер 127

Компенсация движения / B-slice



Каждое разбиение inter-кодируемого макроблока в B-slice может быть предсказано одной или двумя ссылочными кадрами, до или после текущего кадра

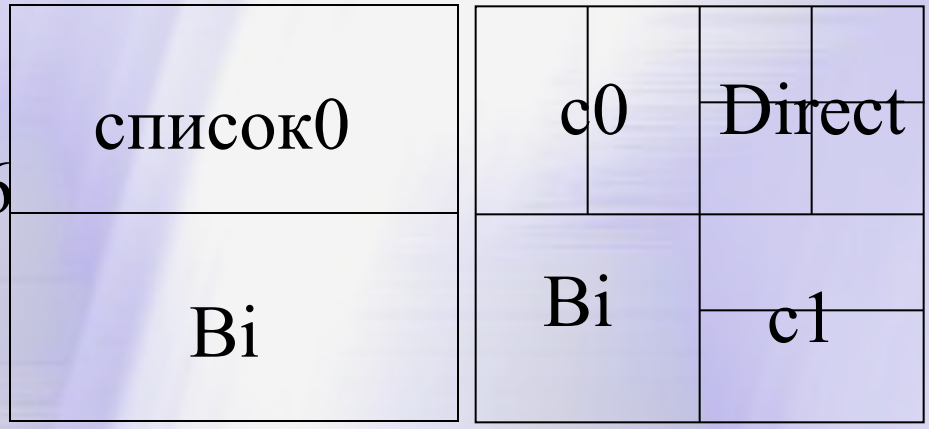
Компенсация движения / B-slice

каждый (макро)блок в B-slice предсказан:

- в режиме Direct
- компенсация движения из списка0
- компенсация движения из списка1
- Bi-предсказание из списка0 и списка1

16x8,8x16

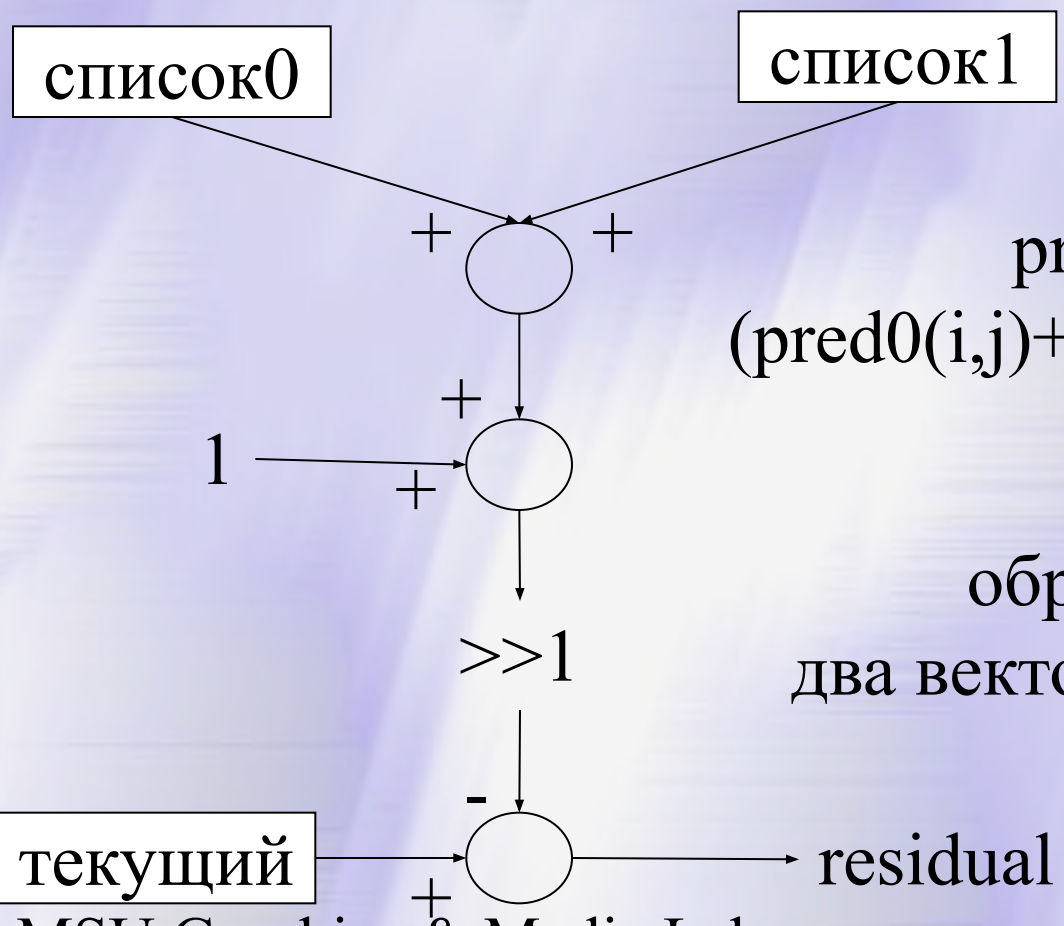
~~Direct~~



16x16

8x8 : все части кодируются в одном режиме

Компенсация движения / Bi-предсказание

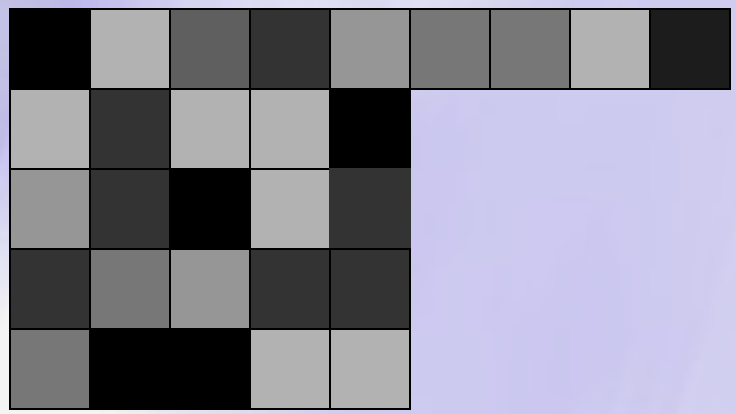


pred(i,j)=
(pred0(i,j)+pred1(i,j)+1)>>1

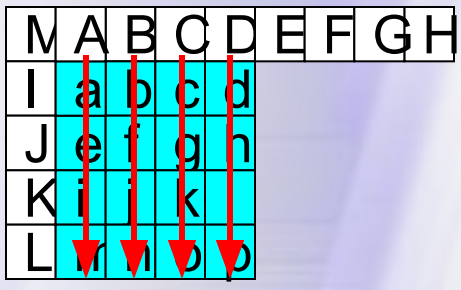
образуется
два вектора движения

Компенсация движения / intra-предсказание

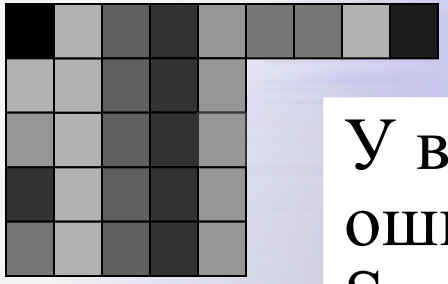
M	A	B	C	D	E	F	G	H
I	a	b	c	d				
J	e	f	g	h				
K	i	j	k	l				
L	m	n	o	p				



9 режимов предсказания для Y 4x4,
4 для Y 16x16, 4 для Cr(Cb) 8x8

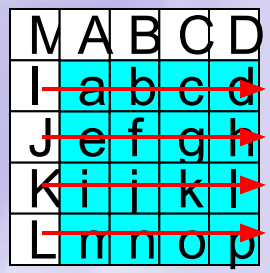


режим 0 (вертикальный)

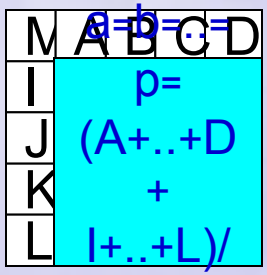


У всех режимов разная ошибка предсказания Sum of Absolute Errors(SAE)

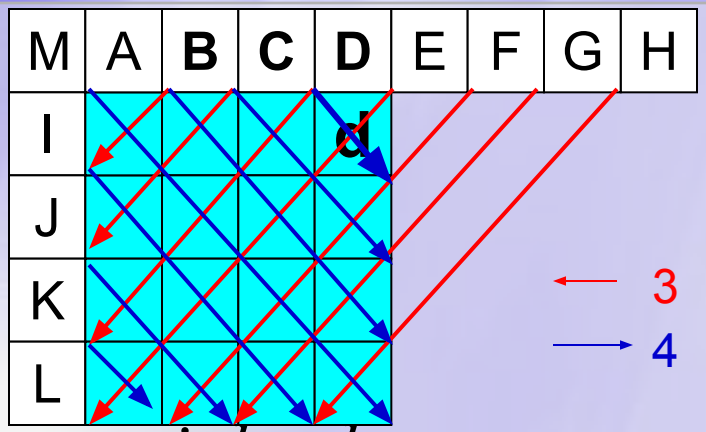
Компенсация движения / intra-предсказание



режим 1

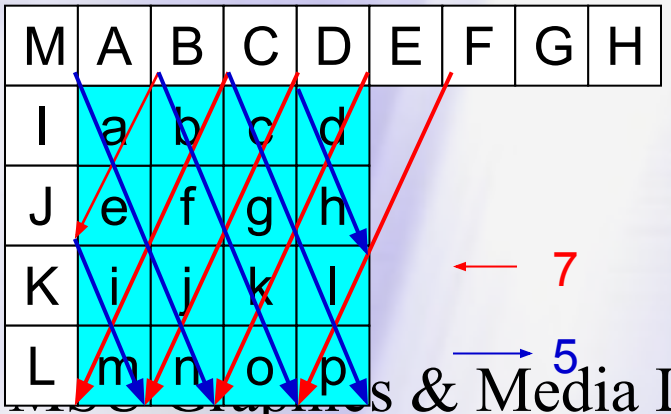


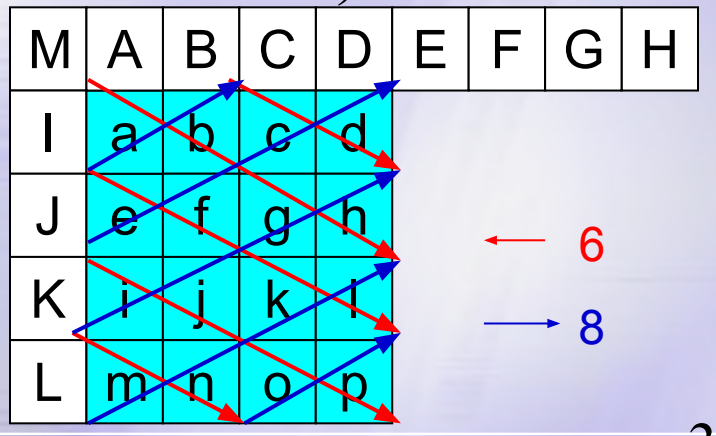
режим 2(DC)



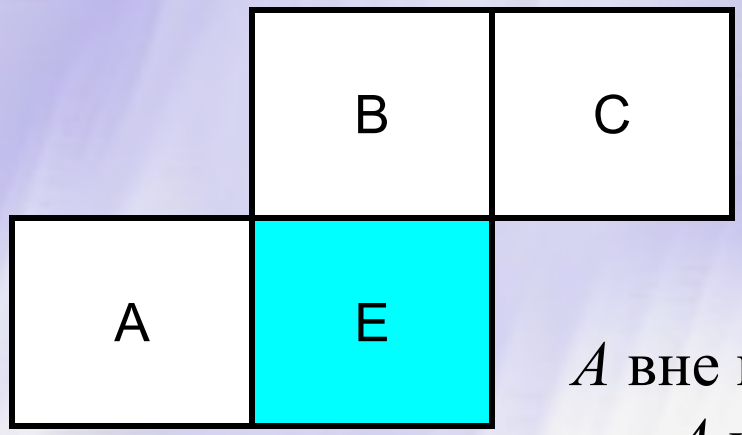
в режимах 3-8 используется *weighted average* :

в режиме 4: d=(B/8+C/2+D/8)





Компенсация движения / intra-предсказание



m*E*=min(mode*A*,mode*B*)-наиболее вероятный режим блока E

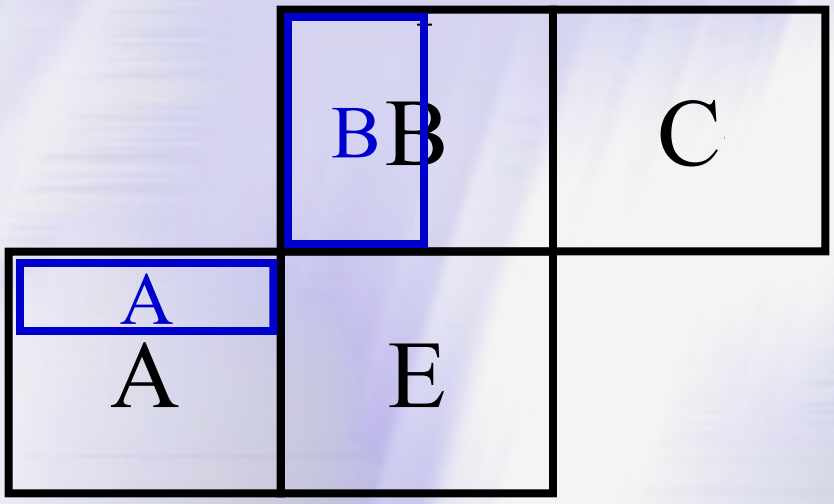
A вне плоскости
A не *intra*
→ m*E*=min(2,mode*B*)

кодировщик(декодер) посылает для каждого блока 4x4 2 флага:
prev_intra_4x4_pred_mode и **rem_intra_4x4_pred_mode**

$$\text{mode}E = \begin{cases} mE, & \text{если prev=0} \\ rem, & \text{если prev=1, rem<}mE \\ rem+1, & \text{если pred=1, rem>=}mE \end{cases}$$

Компенсация движения / Предсказание ВД

Кодируется не сам вектор движения, а разница между ним и другим вектором движения



Для ВД блоков макроблока E предсказаниями являются ВД из блоков:

1) E=2 блока 16x8 *B/A*
2) E=2 блока 8x16 *A|C*
3) в остальных случаях – среднее из *A*,*B* и *C*

— A,B,E – одинакового размера

— A,B,E – разного размера

Преобразование

$$a=\frac{1}{2} \quad b=\sqrt{\frac{1}{2}}\cos\left(\frac{\pi}{8}\right)$$

$$\mathbf{Y}=\underbrace{\begin{pmatrix} 1 & 1 & 1 & 1 \\ 2 & 1 & -1 & -2 \\ 1 & -1 & -1 & 1 \\ 1 & -2 & 2 & -1 \end{pmatrix}}_{\mathbf{C_f}} \begin{pmatrix} \mathbf{X} \end{pmatrix} \underbrace{\begin{pmatrix} 1 & 2 & & 1 \\ 1 & 1 & -1 & -2 \\ 1 & -1 & -1 & 2 \\ 1 & -2 & 1 & -1 \end{pmatrix}}_{\mathbf{C_f^T}} \otimes \underbrace{\begin{pmatrix} a^2 & \frac{ab}{2} & a^2 & \frac{ab}{2} \\ \frac{ab}{2} & \frac{b^2}{4} & \frac{ab}{2} & \frac{b^2}{4} \\ a^2 & \frac{ab}{2} & a^2 & \frac{ab}{2} \\ \frac{ab}{2} & \frac{b^2}{4} & \frac{ab}{2} & \frac{b^2}{4} \end{pmatrix}}_{E_f}$$

Матрица $\mathbf{CXC^T}$ – основа 2D трансформации

E- матрица динамических векторов

Квантование

коэффициенты квантования

$$Z_{ij} = round\left(\frac{Y_{ij}}{Qstep}\right)$$

Qstep - определяется параметром квантования QP

QP	0	1	2	3	4	5	…	18	…	24	…	30	…	51
Qstep	0.625	0.6875	0.825	0.8125	1	1.125		5		10		20	…	224

Фильтрация (деблокинг)

Фильтрация применяется на границах блоков

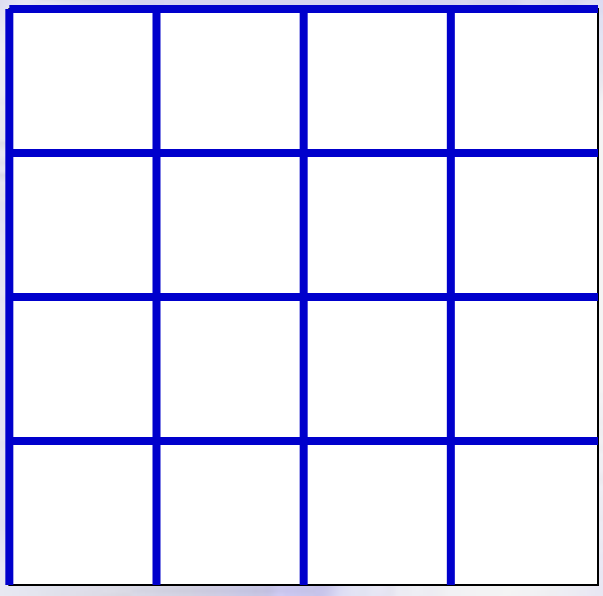
(за исключением границ плоскости)

При условии, что

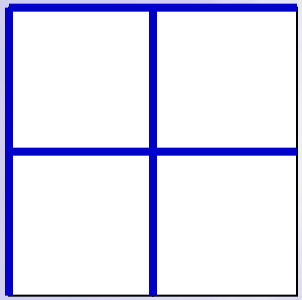
a) bs>0

b) $|p0-q0|<\alpha$ $|q1-q0|<\beta$

$|p1-p0|<\beta$

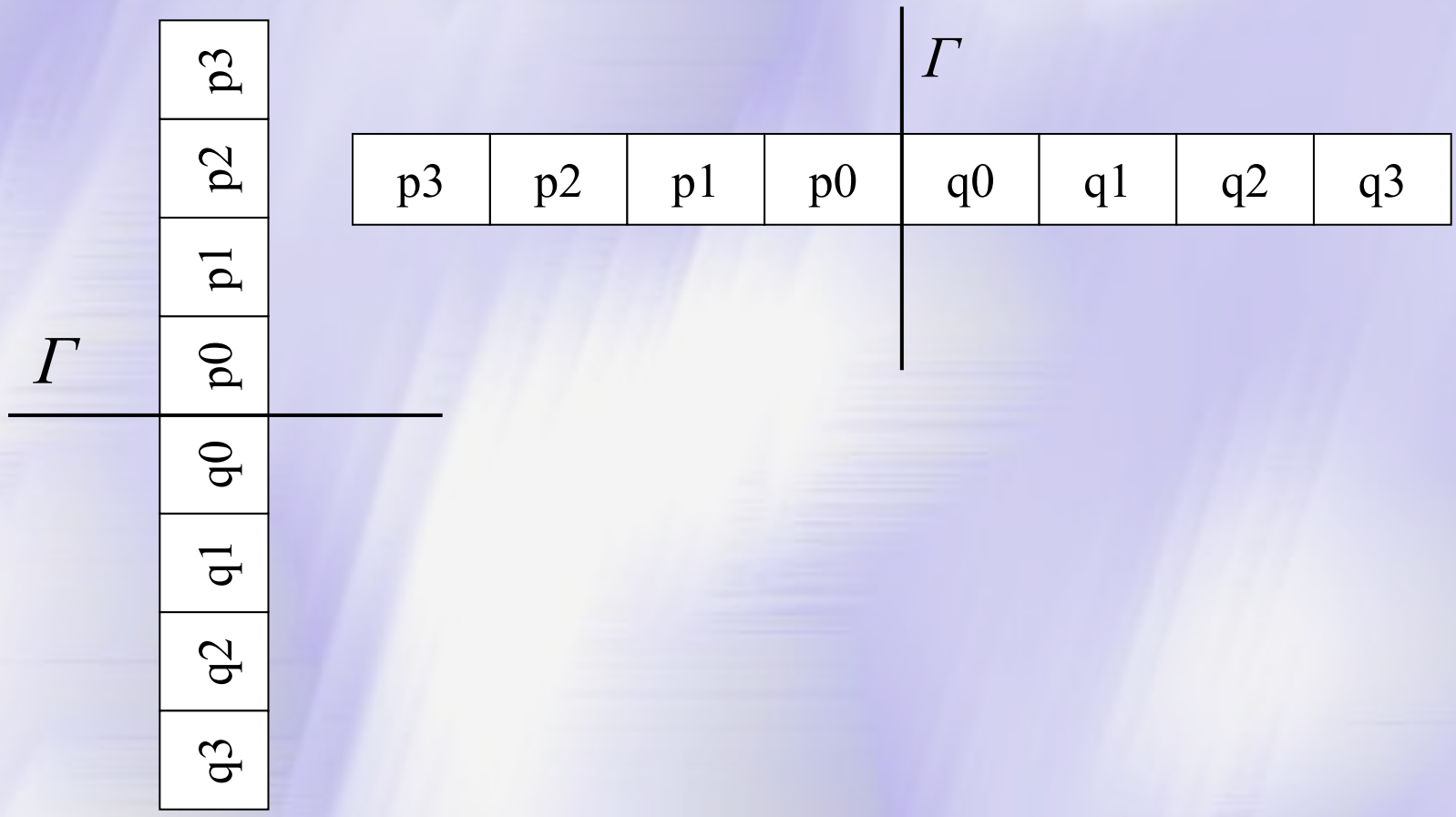


Y 16x16

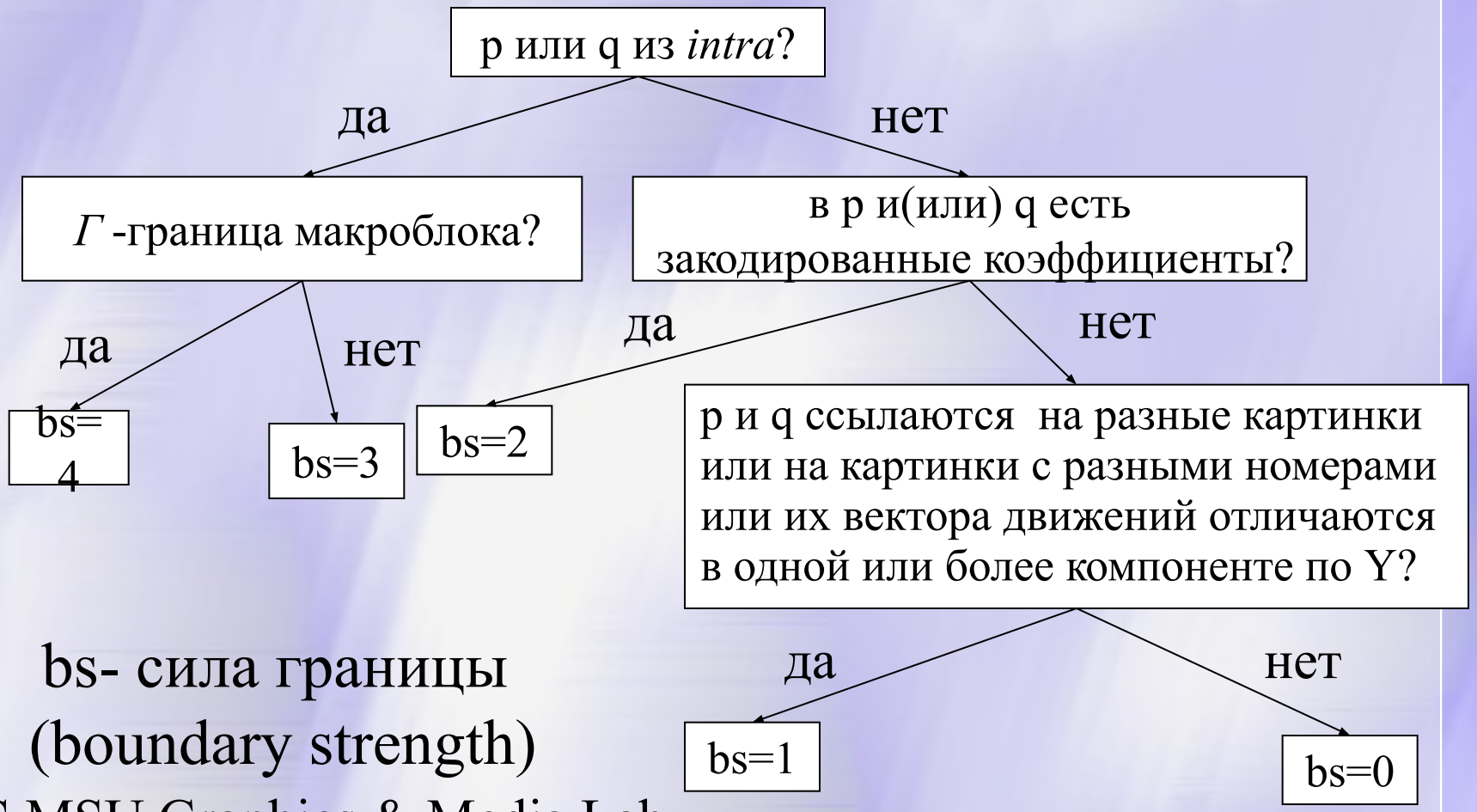


Cr 8x8

Фильтрация (деблокинг)

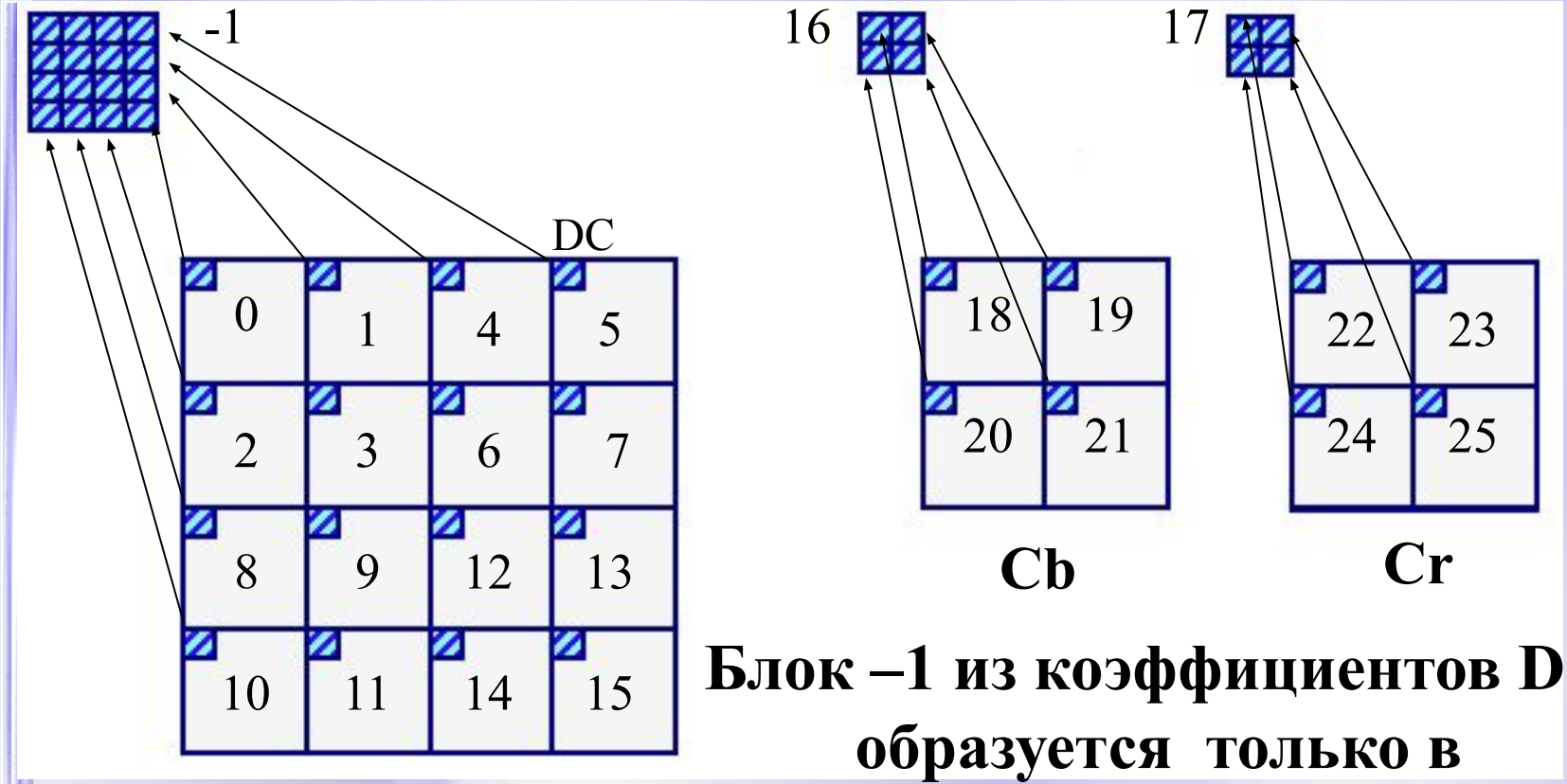


Фильтрация (деблокинг)



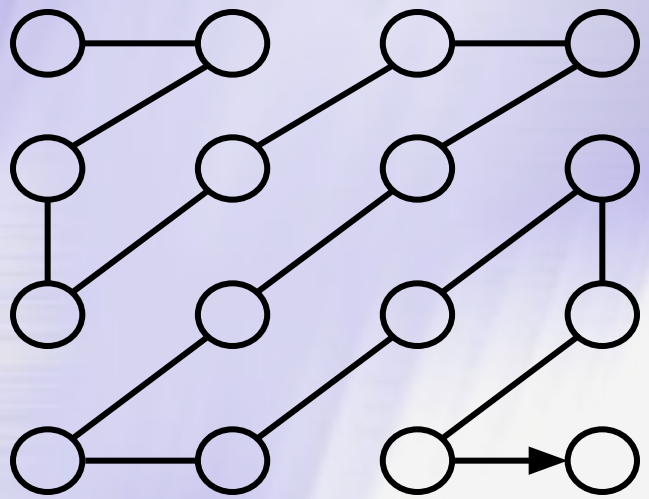
bs- сила границы (boundary strength)

Перестановка/Порядок считывания макроблока

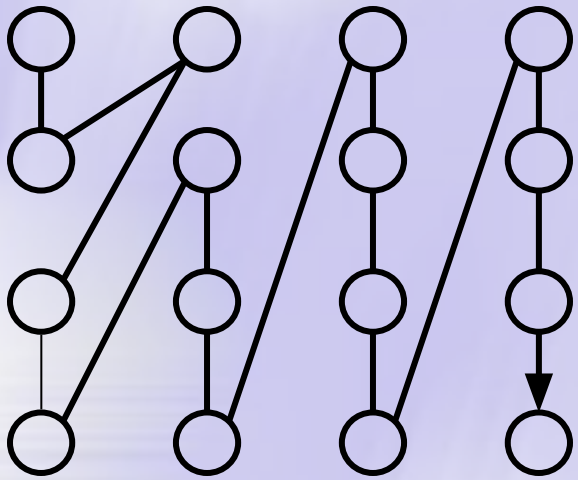


Блок –1 из коэффициентов DC образуется только в 16x16 Intra-кодируемых блоках

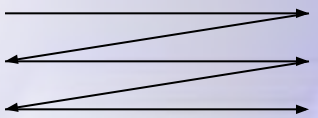
Перестановка



intra 4x4 для *progressive video*

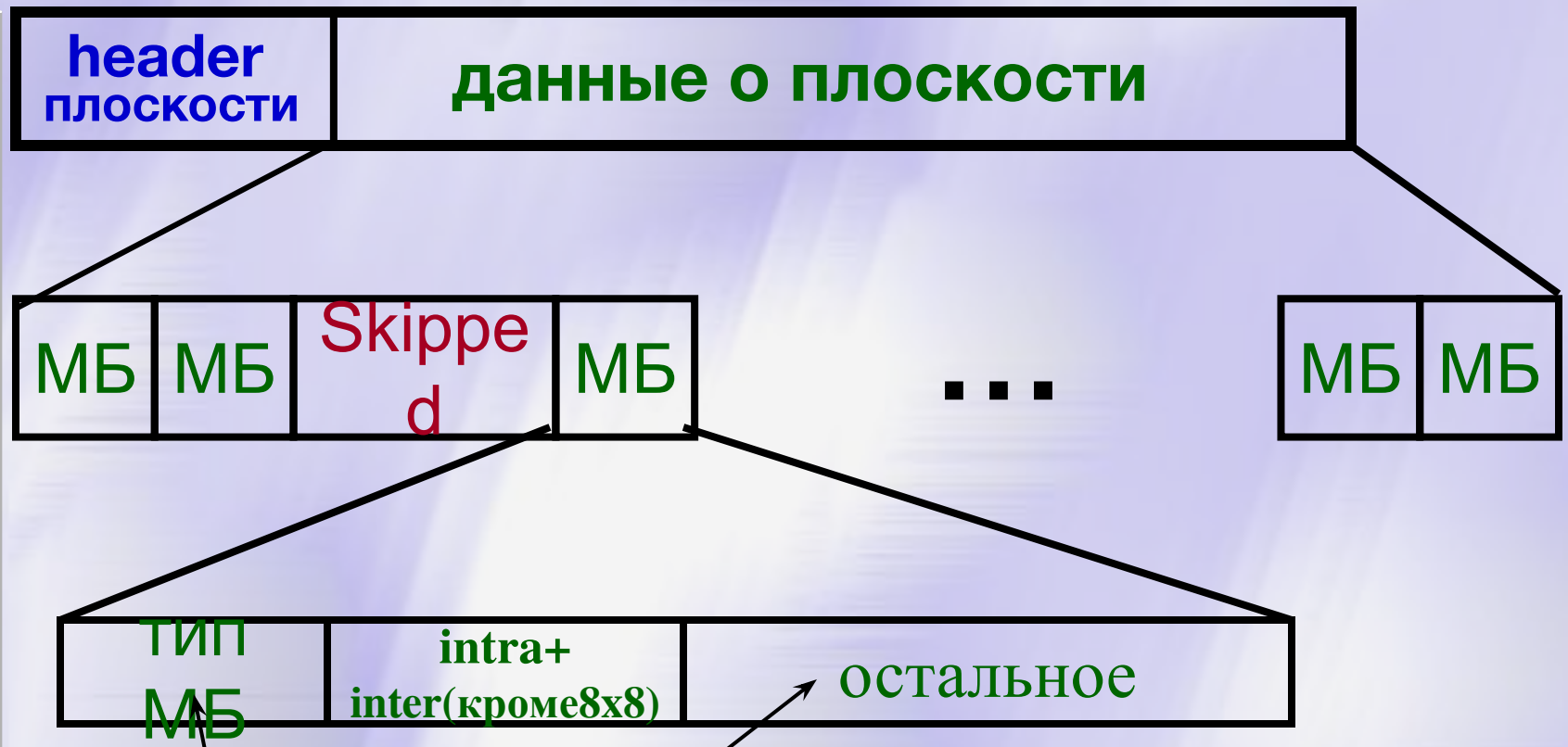


intra 4x4 для *interlaced video*



порядок обхода макроблоков в плоскости

Entropy Coding / Кодирование плоскости



inter 8x8, QP ,закодированные коэффициенты трансформации

inter* или *intra

Entropy Coding

GRAPHICS & MEDIA LAB
VIDEO GROUP

Кодируются следующие данные:

наименование	**назначение**
mb_type	тип макроблока
Coded_block _pattern	показывает какие блоки внутри макроблока содержат закодированные коэффициенты
QP	параметр квантования, кодируются как разность с предыдущим
Refer_frame_index	номера фреймов для предсказания
Residual	разница между блоком и его предсказанием
MV	вектор движения

Entropy Coding

GRAPHICS & MEDIA LAB
VIDEO GROUP

entropy_coding_mode:

0: все данные – CABAC
(Context-based Adaptive Binary Arithmetic Coding)

1: Residual – CALVC
(Context Adaptive Variable Length Coding)
остальное - Exp-Colomb

Entropy Coding / CALVC

После компенсации движения, преобразования и квантования, блок обычно состоит из большого числа нулей

0	3	-1	0
0	-1	1	0
1	0	0	0
0	0	0	0

Блок считывается в зигзаг порядке (см. Перестановка)

0,3,0,1,-1,-1,0,1,0...0

Entropy Coding / CALVC

Кодируются следующие величины:

название	**Что означает**	**Доп. знач.**	**при-мер**
Coeff_token	Число ненулевых коэффициентов	0..16	5
Trailing_ones	Число 1 и -1 (если их больше, то остальные кодируются обычным способом)	0..3	3
Trailing_ones_sign_flag	Знаки при коэф. 1 и -1, кодир-ся как trailing_ones, начиная с конца последовательности	0 (+) 1 (-)	0,1,1

Entropy Coding / CALVC

Далее кодируются:

оставшиеся коэффициенты

1 - level(1) (тоже в обратном порядке)

3 - level(0)

total_zeros –число нулей до последнего ненулевого коэффициента (3)

run_before

0, 3, 0, 1, -1, -1, 0, 1, 0...0

не кодируется 1 0 0 1

Entropy Coding / Exp-Colomb

параметр → code_num → codeword

зависит от частоты встречаемости параметра

codeword=[M нулей][1][INFO]

где

$$M = \lfloor \log_2(code_num+1) \rfloor$$

$$INFO = code_num+1-2^M$$

code_num	codeword
0	1
1	010
2	011
3	00100
4	00101
5	00110
6	00111
7	0001000
8	0001001
…	….

Entropy Coding / CABAC

GRAPHICS & MEDIA LAB
VIDEO GROUP

CABAC (Context Adaptive Binary Arithmetic Coding):

- Выбор вероятностной модели для каждого синтаксического элемента в соответствии с его контекстом
- Адаптация оценок вероятности на основании локальной статистики
- Использование арифметического кодирования предпочтительнее VLC

Entropy Coding / CABAC

Кодирование данных *value* проходит стадии:

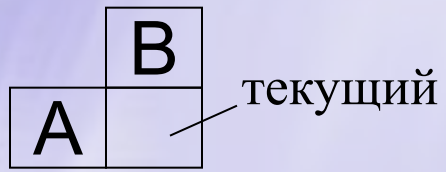
I. Бинаризация

|value|	Бинаризация
0	0
1	10s
2	110s
3	1110s
…..	…..
8	111111110s
>8	Exp-Colomb

Entropy Coding / CABAC

GRAPHICS & MEDIA LAB
VIDEO GROUP

II. Выбор контекстной модели



$e_k = |value_A| + |value_B|$

bin	Контекстная модель
1	$0 \le e_k < 3$ - 0; $3 \le e_k < 33$ -1 ; $33 \le e_k$ -2
2	3
3	4
4	5
5 и более	6

где bin1 – первый бит *value*, bin2- второй и т.д.

Entropy Coding / CABAC

III.Арифметическое кодирование каждого бина
(используются две вероятности p[bin=‘0’] p[bin=‘1’])

IV.Обновление модели

Пример. Выбрана model2 и bin=‘0’

```
model2_0_count++;
if  (model2_0_count>threshold) {
     model2_0_count=0
     model2_1_count=0;
}
```

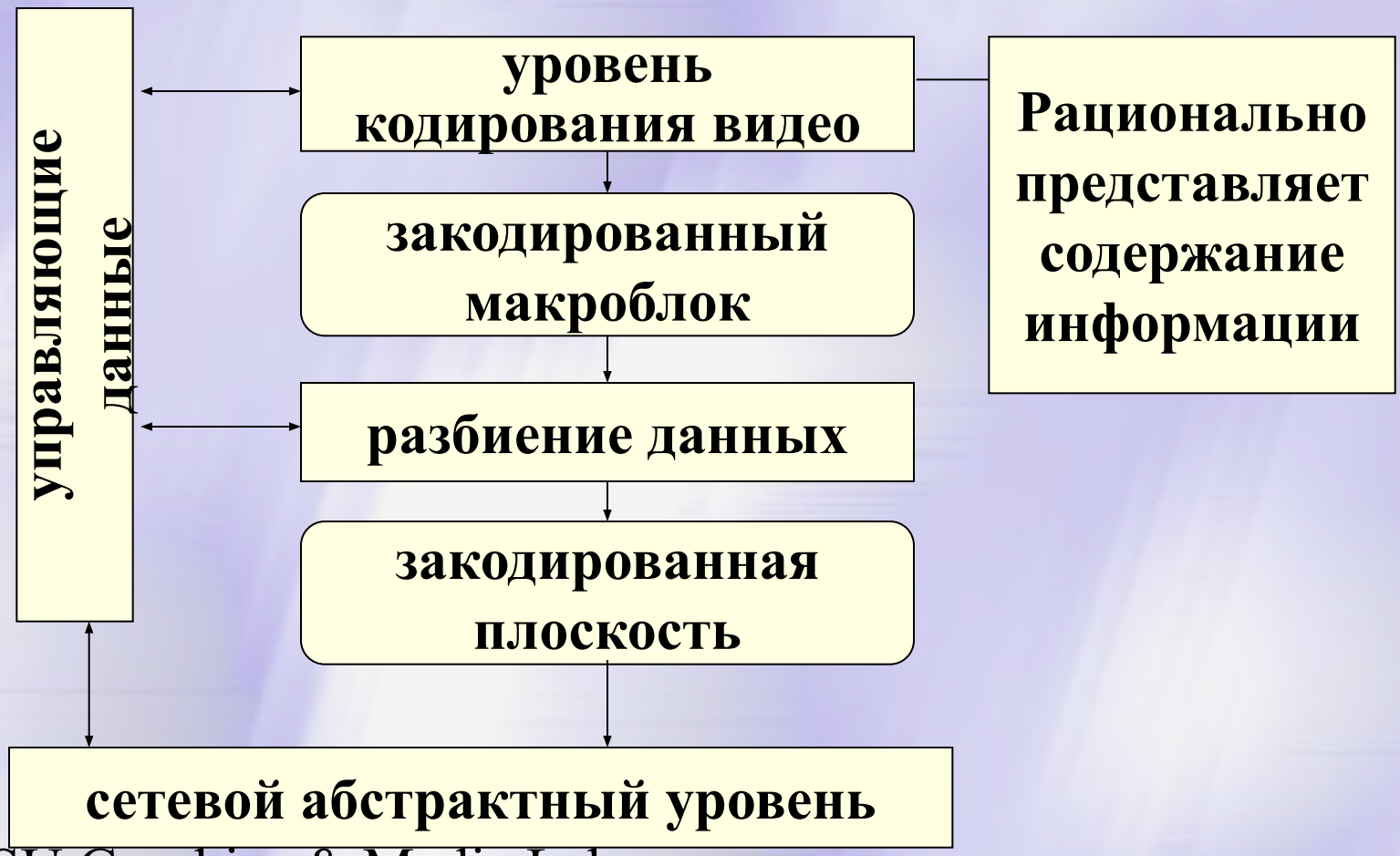
NAL

Network Abstraction Layer - Сетевой Абстрактный Уровень

Форматирует видео после Entropy Encoding и добавляет к данным заголовок нужного типа, чтобы в дальнейшем передать информацию по любому из всего разнообразия транспортных уровней или носителей данных

…	NAL header	посл-ть байтов	NAL header	посл-ть байтов	…

NAL


управляющие данные
уровень кодирования видео
закодированный макроблок
разбиение данных
закодированная плоскость
Рационально представляет содержание информации
сетевой абстрактный уровень