DSC v1.2a Errata E1
Published 12/31/18

This errata contains all SCRs published through the end of 2018.

The following SCRs are included in DSC_v1.2a_E1:

- DSC 1.2a buffer fullness term typographic error
- DSC summary and objectives clarifications
- DSC 1.2a Correct rcXformBpgOffset and bpSad to match C Model
- DSC 1.2a Correct bitSaveMode to match C Model
- DSC 1.2a Table 4-1 nsl_bpg_offset clarification SCR
- DSC 1.2a rate control clarification SCR

VESA STANDARDS CHANGE REQUEST FORM

To be Filled in by Submitter (Refer to VESA Document VP235H, Section 5)

TITLE:	DSC 1.2a buffer fullness term typographic error
AFFECTED DOCUMENT:	DSC 1.2a
REVISION CATEGORY:	Refer to VP235H Appendix A; will be subject to Task Group review
SUBMITTED TO:	DSC TG
SPONSOR:	Dale Stolitzka, Samsung, Fred Walls, Broadcom and Craig Wiley, Parade Technologies

SCR REVISION HISTORY	
(DATE)	**(CHANGE)**
02/06/2017	**Initial Submission of SCR**

(add more rows as needed)

To be Filled in by VESA Office:

VESA SCR NUMBER:	(To be assigned by VESA office)
SCR ENTRY DATE:	02/06/2017

To be Filled in by Task Group or VESA Office

SCR ADOPTED, REJECTED, or otherwise DISPOSITIONED for other action	SCR is (adopted) or (rejected) or (Dispositioned for other action) If rejected, explain reason for acceptation or rejection If dispositioned, explain action or plan for action (such as including in future draft specification revision, or re-visiting at future date, or other)
DATE SCR ADOPTED or REJECTED or DISPOSITIONED	04/20/2017 Adopted

Summary of the Proposed Change(s)

The change suggested herein eliminates an operand from an informative term, which describes the largest, minimum number of bits remaining in the encoder rate buffer at the end of a slice. Normative language does not apply to these terms. The description is related to rcXformOffset and rcXformScale that are normative values.

IPR (Intellectual Property Rights) declaration, if any

None known

Benefits as a Result of the Changes

Corrects an informative equation

Assessment of the Impact

None

Analysis of the Device Hardware Implication

No impact

Analysis of the Device Software Implications

None

Analysis of the Compliance Test & Interop Implications

None

New Referenced Documents Resulting from Change

None other than this SCR

Attachments

None

Proposed Document Change(s) or Addition(s)

In DSC 1.2a, page 105, remove "*pixelsPerGroup" in the following terms:

initial_xmit_delay * ***bits_per_pixel*** - **`numExtraMuxBits`**

- End of Document -

VESA STANDARDS CHANGE REQUEST FORM

To be Filled in by Submitter (Refer to VESA Document VP235H, Section 5)

TITLE:	DSC summary and objectives clarifications
AFFECTED DOCUMENT:	DSC 1.2a
REVISION CATEGORY:	Refer to VP235H Appendix A; will be subject to Task Group review
SUBMITTED TO:	DSC TG
SPONSOR:	Madhukar Budagavi and Dale Stolitzka (DSC TG Chair), Samsung Electronics

SCR REVISION HISTORY	
(DATE)	**(CHANGE)**
03/21/2017	**Initial Submission of SCR**
06/05.2017	Final for an adoption vote

(add more rows as needed)

To be Filled in by VESA Office:

VESA SCR NUMBER:	(To be assigned by VESA office)
SCR ENTRY DATE:	03/21/2017

To be Filled in by Task Group or VESA Office

SCR **ADOPTED, REJECTED, or otherwise DISPOSITIONED for other action**	SCR is (adopted) or (rejected) or (Dispositioned for other action) If rejected, explain reason for acceptation or rejection If dispositioned, explain action or plan for action (such as including in future draft specification revision, or re-visiting at future date, or other)
DATE SCR **ADOPTED or REJECTED or DISPOSITIONED**	06/29/2017 Adopted

Summary of the Proposed Change(s)

DSC has become widely adopted by many transport specifications in the industry. Examples of transport links known to this task group as supporting DSC include MIPI Alliance DSI and DSI-2, and VESA eDP and DP.

DSC adoption has outgrown its original plan described in the DSC Standard Summary section and the Theory of Operations – Objectives section. VESA has made strides to enable DSC without bias, in other words "transport agnostic" as captured by the Summary clause, second paragraph,

> "DSC does not specify a Transport Layer. Practical systems that use DSC must follow a suitable transport specification in which the Transport Layer conveys DSC streams, from source to destination."

The DSC standard can and should be clarified by the wording in this SCR to enhance adoption where not anticipated. Here are examples of potential conflicts that can be improved and clarified:

Present wording	Section	Potentially Hindered	Proposed Fix
Display interfaces defined in the Requirements clause calls out MIPI Alliance DSI, VESA DP and eDP, HDMI	Summary, 3rd para	Packetized interfaces, that are not necessarily display interfaces (I/F) * wireless * ethernet * automotive I/Fs * SMPTE studio I/Fs	Change "display interface" to "interface"
Real-time streaming	Summary, 1st para and Objectives clause 3.1, 2nd para	* faster than real-time * partial frame update, which is slower than real time * MST modes	Change to able to work in real-time
Rate buffer, frame buffer storage behavior loosely defined	Objectives clause 3.1, 2nd para.	Temporary storage is vague when considering frame buffer data that does not change often or data stored in context-saving displays	Describe DSC scope to avoid overlaps with the transport layer or application layer

IPR (Intellectual Property Rights) declaration, if any

None known

Benefits as a Result of the Changes

Clarifications relieve confusion over potential, unintended limitations of DSC usage over commonly used industry interfaces

Assessment of the Impact

Positive impact for more wide-spread DSC 1.2a adoption

Analysis of the Device Hardware Implication

No implications to the standard

Analysis of the Device Software Implications

No implications to the standard

Analysis of the Compliance Test & Interop Implications

No implications to the standard

New Referenced Documents Resulting from Change

None foreseen at this time

Attachments

No attachments

Proposed Document Change(s) or Addition(s)

1. Modify the Summary clause, first paragraph:

The DSC Standard is a specification of the algorithms used for compressing and decompressing image display streams, including the specification of the syntax and semantics of the compressed video bitstream. DSC is designed to be able to work in real time for systems that contain compression, transmission, decompression, and display.

2. Modify the Summary clause, third paragraph:

DSC is a compression and decompression standard for display streams between two distinct devices, either from one box to another separate box, or from one chip to another separate chip within a box, by way of an interface. Interfaces that could apply this standard include those between a mobile application host processor and display panel module, between a computer graphics output and display monitor, or between a consumer electronics source device to a display device, such as a television. Interfaces can be either wired or wireless.

3. Modify clause 3.1 Overview, second paragraph:

This Standard specifies the encoding process, bitstream syntax and semantics, and decoding process used for compressing display streams. The entire system is capable of operating in real time. Uncompressed video enters the encoder in raster scan order. The encoder compresses incoming pixels to form a bitstream and then temporarily stores portions of the bitstream in its rate buffer. The rate buffer's output is the Picture Layer of a DSC bitstream, i.e., everything except the picture parameter set (PPS). The DSC bitstream is conveyed from the encoder to the decoder, by way of a Transport Layer, which is outside the scope of this Standard. The decoder receives the bitstream into a rate buffer or a frame buffer or a combined rate and frame buffer, which stores portions of the bitstream. The decoder decodes bits from the rate buffer and then forms uncompressed pixels, which are output in raster scan order, and then sent to a display. The image output from the decoding process has the same format as the image input to the encoding process. Figure 3-1 illustrates how DSC works in an end-to-end system.

Background Information

No additional background at this time.

- End of Document -

VESA STANDARDS CHANGE REQUEST FORM

To be Filled in by Submitter (Refer to VESA Document VP235H, Section 5)

TITLE:	DSC 1.2a Correct rcXformBpgOffset and bpSad to match C Model
AFFECTED DOCUMENT:	VESA Display Stream Compression (DSC) Standard, Version 1.2a
REVISION CATEGORY:	Revision Category 1 – Specification Typos, Informative Changes, and Minor Normative Changes
SUBMITTED TO:	Display Stream Compression Task Group
SPONSOR:	David Stears, NVIDIA

SCR REVISION HISTORY	
(DATE)	**(CHANGE)**
05/26/2017	**Initial Submission of SCR**

(add more rows as needed)

To be Filled in by VESA Office:

VESA SCR NUMBER:	(To be assigned by VESA office)
SCR ENTRY DATE:	05/26/2017

To be Filled in by Task Group or VESA Office

SCR ADOPTED, REJECTED, or otherwise DISPOSITIONED for other action	SCR is (adopted) or (rejected) or (Dispositioned for other action) If rejected, explain reason for acceptation or rejection If dispositioned, explain action or plan for action (such as including in future draft specification revision, or re-visiting at future date, or other)
DATE SCR ADOPTED or REJECTED or DISPOSITIONED	08/03/2017 Adopted

Summary of the Proposed Change(s)

This SCR proposes two changes to the DSC 1.2a spec text to bring it in line with behavior of the C Model. Neither change impacts the C Model itself.

There is a minor error in the definition for rcXformBpgOffset in section 6.8.4 of the DSC 1.2a spec.

There is also a minor error in the formula for bpSad in section 6.4.4.1. The text describing this calculation is correct, but the formula is incorrect.

IPR (Intellectual Property Rights) declaration, if any

Nothing new

Benefits as a Result of the Changes

This helps avoid confusion to adopters who may rely more on the spec text rather than the C code to understand the algorithm.

Assessment of the Impact

This is a minor change.

Analysis of the Device Hardware Implication

None. The C code is normative and is correct.

Analysis of the Device Software Implications

None

Analysis of the Compliance Test & Interop Implications

None

New Referenced Documents Resulting from Change

None

Attachments

None

Proposed Document Change(s) or Addition(s)

Section 6.8.4, page 110

Change the "=" to "+=" for the two terms related to the second line offset. The effect of these parameters should be cumulative to the first line offset, not separate.

```
if (first line of slice)
      rcXformBpgOffset = first_line_bpg_offset;
else
```

```
        rcXformBpgOffset = -floor(nfl_bpg_offset);
if (second line of slice)
        rcXformBpgOffset += second_line_bpg_offset;
else
        rcXformBpgOffset += -floor(nsl_bpg_offset);
rcXformBpgOffset -= floor(slice_bpg_offset);
```

Section 6.4.4.1, page 81

To construct the 9x1 SAD, three 3x1 SADs are added together. The resulting sum is right shifted by three to drop the lsbs (as described in the text). The equation in the spec text shows a clamp to 9 bits, which is incorrect.

(Surrounding text included for context)

> Regardless of mode, the resulting sum of **modifiedAbsDiff** values is a 10-bit value that represents
> the 3x1 partial SAD for a **candidateVector**; this 10-bit value is clamped to nine bits (i.e., values
> greater than 511 are clamped to 511). Three 9-bit, 3-pixel partial SADs are summed to produce
> the final 9-pixel SAD, which is an 11-bit number. The three lsbs of each 9x1 SAD are truncated
> before comparison:
>
> **bpSad**[**candidateVector**] = ~~MIN(511,~~ sad3x1_0[**candidateVector**] + sad3x1_1[**candidateVector**] + sad3x1_2[**candidateVector**]) >> 3;

Background Information

Included below is the relevant section of dsc_codec.c for rcXformBpgOffset. In this part of the code, the bpg offset is applied directly to current_bpg_target (highlighted).

```
// Account for first line boost
if (vPos == 0)
{
        current_bpg_target = dsc_cfg->first_line_bpg_ofs;
        increment = -(dsc_cfg->first_line_bpg_ofs << OFFSET_FRACTIONAL_BITS);
} else {
        current_bpg_target = -(dsc_cfg->nfl_bpg_offset >> OFFSET_FRACTIONAL_BITS);
        increment = dsc_cfg->nfl_bpg_offset;
}

// Account for 2nd line boost
```

```
        if (vPos == 1)
        {
                current_bpg_target += dsc_cfg->second_line_bpg_ofs;
                increment += -(dsc_cfg->second_line_bpg_ofs << OFFSET_FRACTIONAL_BITS);
                if(!dsc_state->secondOffsetApplied)
                {
                        dsc_state->secondOffsetApplied = 1;
                        dsc_state->rcXformOffset -= dsc_cfg->second_line_ofs_adj;
                }
        } else {
                current_bpg_target += -(dsc_cfg->nsl_bpg_offset >> OFFSET_FRACTIONAL_BITS);
                increment += dsc_cfg->nsl_bpg_offset;
        }
```

Included below is the relevant section of dsc_codec.c for bpSad (bp_sads below). BP_SIZE is equal to 3, so the second loop is used to sum the three 3x1 partial SADs.

```
for (candidate_vector=0; candidate_vector<BP_RANGE; ++candidate_vector)
{
        bp_sads[candidate_vector] = 0;

        for (i=0; i<BP_SIZE; ++i)
        {
                int sad3x1 = 0;

                // Use all 3 components
                for(j=0; j<dsc_state->numComponents; ++j)
                        sad3x1 += dsc_state->lastErr[j][i][candidate_vector];
                // sad3x1 is 9 bits
                sad3x1 = MIN(511, sad3x1);

                bp_sads[candidate_vector] += sad3x1;  // 11-bit SAD
        }
        // Each bp_sad can have a max value of 63*9 pixels * 3 components = 1701 or 11 bits
        bp_sads[candidate_vector] >>= 3;  // SAD is truncated to 8-bit for comparison
}
```

- End of Document -

VESA STANDARDS CHANGE REQUEST FORM

To be Filled in by Submitter (Refer to VESA Document VP235H, Section 5)

TITLE:	DSC 1.2a Correct bitSaveMode to match C Model
AFFECTED DOCUMENT:	VESA Display Stream Compression (DSC) Standard, Version 1.2a
REVISION CATEGORY:	Revision Category 1 – Specification Typos, Informative Changes, and Minor Normative Changes
SUBMITTED TO:	Display Stream Compression Task Group
SPONSOR:	David Stears, NVIDIA

SCR REVISION HISTORY	
(DATE)	**(CHANGE)**
07/10/2017	**Initial Submission of SCR**
07/17/2017	Rename the SCR to include "DSC 1.2a"

(add more rows as needed)

To be Filled in by VESA Office:

VESA SCR NUMBER:	(To be assigned by VESA office)
SCR ENTRY DATE:	07/10/2017

To be Filled in by Task Group or VESA Office

SCR ADOPTED, REJECTED, or otherwise DISPOSITIONED for other action	SCR is (adopted) or (rejected) or (Dispositioned for other action) If rejected, explain reason for acceptation or rejection If dispositioned, explain action or plan for action (such as including in future draft specification revision, or re-visiting at future date, or other)
DATE SCR ADOPTED or REJECTED or DISPOSITIONED	09/15/2017 Adopted

Summary of the Proposed Change(s)

This SCR proposes a change to the DSC 1.2a spec text to bring it in line with behavior of the C Model. This change does not impact the C Model itself.

The pseudo-code describing the calculation of bitSaveMode in section 6.8.4 has a few minor errors.

IPR (Intellectual Property Rights) declaration, if any

Nothing new

Benefits as a Result of the Changes

This helps avoid confusion to adopters who may rely more on the spec text rather than the C code to understand the algorithm.

Assessment of the Impact

This is a minor change.

Analysis of the Device Hardware Implication

None. The C code is normative and is correct.

Analysis of the Device Software Implications

None

Analysis of the Compliance Test & Interop Implications

None

New Referenced Documents Resulting from Change

None

Attachments

None

Proposed Document Change(s) or Addition(s)

Section 6.8.4, page 107

"!" needs to be added prior to ichSelected in two places in this pseudocode:

```
if (dsc_version_minor == 2 && not first line of slice &&
no flatness signaled for supergroup)
      {
            If (!ichSelected && (mpSel >= 3))
            {
                  mppState = MIN(mppState + 1, 2);
                  if(mppState >= 2)
                        bitSaveMode = 2;
            }
```

```
            else if (!ichSelected && predActivity >= bitSaveThresh)
                  bitSaveMode = bitSaveMode; // Don't reset
            else if ichSelected
                  bitSaveMode = MAX(1, bitSaveMode);
            else
                  bitSaveMode = mppState = 0;
      }
      else
            bitSaveMode = mppState = 0;
```

Background Information

Included below is the equivalent section of dsc_codec.c for bitSaveMode. The relevant terms have been highlighted.

```
      if((dsc_cfg->dsc_version_minor == 2) && dsc_state->vPos > 0 && dsc_state->firstFlat
== -1)
      {
            if(!dsc_state->ichSelected && (mpsel >= 3))
            {
                  dsc_state->mppState++;
                  if(dsc_state->mppState >= 2)
                        dsc_state->bitSaveMode = 2;
            }
            else if (!dsc_state->ichSelected && predActivity >= bitSaveThresh)
                  dsc_state->bitSaveMode = dsc_state->bitSaveMode;      // Don't reset
            else if (dsc_state->ichSelected)
                  dsc_state->bitSaveMode = MAX(1, dsc_state->bitSaveMode);
            else
            {
                  dsc_state->mppState = 0;
                  dsc_state->bitSaveMode = 0;
            }
      }
      else
      {
            dsc_state->bitSaveMode = 0;
            dsc_state->mppState = 0;
      }
```

- End of Document -

VESA STANDARDS CHANGE REQUEST FORM

To be Filled in by Submitter (Refer to VESA Document VP235H, Section 5)

TITLE:	DSC 1.2a Table 4-1 nsl_bpg_offset clarification SCR
AFFECTED DOCUMENT:	DSC 1.2a specification
REVISION CATEGORY:	Category 1 (Refer to VP235H Appendix A; will be subject to Task Group review)
SUBMITTED TO:	Task Group
SPONSOR:	Fred Walls, Broadcom Ltd., Tim Papenfuss, Socionext, Inc.

SCR REVISION HISTORY	
(DATE)	**(CHANGE)**
08/14/2017	**Initial Submission of SCR**

(add more rows as needed)

To be Filled in by VESA Office:

VESA SCR NUMBER:	(To be assigned by VESA office)
SCR ENTRY DATE:	08/16/2017

To be Filled in by Task Group or VESA Office

SCR ADOPTED, REJECTED, or otherwise DISPOSITIONED for other action	SCR is (adopted) or (rejected) or (Dispositioned for other action) If rejected, explain reason for acceptation or rejection If dispositioned, explain action or plan for action (such as including in future draft specification revision, or re-visiting at future date, or other)
DATE SCR ADOPTED or REJECTED or DISPOSITIONED	10/26/2017 Adopted

Summary of the Proposed Change(s)

This change involves adding clarifying text to the nsl_bpg_offset entry in Table 4-1 that the value has 11 fractional bits.

IPR (Intellectual Property Rights) declaration, if any

N/A

Assessment of the Impact

This is editorial and has no impact on implementations.

Analysis of the Device Hardware Implication

None.

Analysis of the Device Software Implications

None.

Analysis of the Compliance Test & Interop Implications

None.

New Referenced Documents Resulting from Change

N/A

Proposed Document Change(s) or Addition(s)

Section 4.1.1, Table 4-1:

Table 4-1: Picture Parameter Set Syntax Elements

Syntax Element	Size (Bits)	Format	Description
…	…	…	…
nsl_bpg_offset	16	Unsigned (11 fractional bits)	Specifies the number of bits (including fractional bits) that are deallocated for each group that is ***not*** in the second line of a slice. If the second line has an additional bit budget, the additional bits that are allocated must come out of the budget for coding the remainder of the slice. Therefore, the value must be programmed to ***second_line_bpg_offset*** / (***slice_height*** - 1), and then rounded up to 16 fractional bits. Value shall be 0 if either of the following conditions exist: • ***dsc_version_minor*** = 1 • ***native_420*** = 0.
…	…	…	…

Background Information

Date: Mon, Aug 14, 2017 at 4:52 AM
Subject: DSC1.2a nsl_bpg_offset
To: "support@vesa.org" <support@vesa.org>

Dear Sir,

Regarding the DSC1.2a standard, I have a question concerning the PPS parameter “nsl_bpg_offset”.

In the standard document, its format is given as 16bit unsigned, but in the description it says “including fractional bits”.

From the reference model it looks like its format is 16bits of which 11 fractional bits, like “nfl_bpg_offset” and “slice_bpg_offset”.

Could you please confirm the format?

VESA STANDARDS CHANGE REQUEST FORM

To be Filled in by Submitter (Refer to VESA Document VP235H, Section 5)

TITLE:	DSC 1.2a rate control clarification SCR
AFFECTED DOCUMENT:	DSC 1.2a specification
REVISION CATEGORY:	Category 1 (Refer to VP235H Appendix A; will be subject to Task Group review)
SUBMITTED TO:	Task Group
SPONSOR:	Fred Walls, Broadcom Inc. Tim Papenfuss, Socionext Inc.

SCR REVISION HISTORY	
(DATE)	**(CHANGE)**
09/06/2018	**Initial Submission of SCR**
10/09/2018	Updated sponsor list

(add more rows as needed)

To be Filled in by VESA Office:

VESA SCR NUMBER:	(To be assigned by VESA office)
SCR ENTRY DATE:	09/06/2018

To be Filled in by Task Group or VESA Office

SCR ADOPTED, REJECTED, or otherwise DISPOSITIONED for other action	SCR is (adopted) or (rejected) or (Dispositioned for other action) If rejected, explain reason for acceptation or rejection If dispositioned, explain action or plan for action (such as including in future draft specification revision, or re-visiting at future date, or other)
DATE SCR ADOPTED or REJECTED or DISPOSITIONED	11/23/2018 - Adopted

Summary of the Proposed Change(s)

This change involves adding text to the rate control sections 6.6.4 and 7.2 to cover the additional entropy encoder/decoder outputs required by the DSC 1.2a rate control improvements.

IPR (Intellectual Property Rights) declaration, if any

N/A

Assessment of the Impact

This is editorial and has no impact on implementations.

Analysis of the Device Hardware Implication

None.

Analysis of the Device Software Implications

None.

Analysis of the Compliance Test & Interop Implications

None.

New Referenced Documents Resulting from Change

N/A

Proposed Document Change(s) or Addition(s)

Section 6.6.4:

For the purposes of RC, the entropy encoder outputs ~~two~~several values:

- **codedBits**
- **rcSizeGroup**
- **midpointSelected**
- **ichSelected**
- **predictedSize[0..3]**

The **codedBits** value represents the actual number of bits that are used to code a group. The **rcSizeGroup** value is set to the number of bits that DSU-VLC would have spent coding that group, if the size prediction had exactly matched the actual sizes of the residuals within the group. That is, for each unit within the group, find the largest size of the residuals within the unit, times the number of samples in the unit, plus 1 for a prefix coding the value 0, then add the resulting sizes of the units in the group. If MPP is selected, the value of the largest size of the residual within the unit for this purpose is assumed to be **cpntBitDepth - qLevel**. If ICH-mode is selected, **rcSizeGroup** is set to 1 + **ichIndicesPerGroup** * 5. **midpointSelected** is the number of units in a group for which MPP was selected. **ichSelected** is a flag indicating whether ICH-mode is selected for a group. **predictedSize[0..3]** are the predicted sizes for each unit.

Section 7.2:

The entropy decoder parses the bits from the incoming bitstream after demultiplexing. The Picture Layer is demultiplexed to extract the Slice Layer bits for each slice. The substream demultiplexer demultiplexes the Slice Layer data into three or four substreams. The entropy decoder parses the Substream Layer, which is described in Section 4.5.

Each group in the Substream Layer is sequentially processed. Some groups have conditional bits at the beginning of the luma unit, associated with flatness determination. After each group is processed, the entropy decoder sends the residual and ICH index data to the pixel reconstruction and ICH blocks. The entropy decoder outputs the total number of bits parsed for the entire group (**codedGroupSize**), ~~and~~the number of bits that would have been used had the sizes been optimally predicted (**rcSizeGroup**), the number of units in the group that used midpoint prediction (**midpointSelected**), a flag to indicate whether ICH-mode is selected or not (**ichSelected**), and the predicted sizes for each unit (**predictedSize[0..3]**) to the rate control.

…

Background Information

Socionext email from 9/5/18:

```
From: Papenfuss, Tim
Date: Wed, Sep 5, 2018 at 2:33 AM
Subject: DSC1.2a feedback
```

To: support@vesa.org<mailto:support@vesa.org> <support@vesa.org<mailto:support@vesa.org>>

Dear Sir,

I have some feedback regarding the DSC1.2a standard (dated 18 January 2017, reference model 1.57).

I believe that some sections in the written standard were not updated to reflect the changes in Rate Control when moving from version 1.1 to 1.2a.

In section 6.6.4 Outputs to Rate Control, it says:
"For the purposes of RC, the entropy encoder outputs two values:
* codedBits
* rcSizeGroup"

Similarly in section 7.2 Entropy Decoding, it says:
"The entropy decoder outputs the total number of bits parsed for the entire group (codedGroupSize) and number of bits that would have been used had the sizes been optimally predicted (rcSizeGroup) to the rate control."

However, the reference model RateControl() function uses further values from entropy encoding:
* dsc_state->midpoint_selected (dsc_codec.c @ 1199)
* dsc_state->ich_selected (dsc_codec.c@1278)
* dsc_state->predictedSize (dsc_codec.c @ 1270)

This can lead to surprises when implementing hardware interfaces between Rate Control and Entropy Encoder/Decoder.

Best regards,

TIM PAPENFUSS
design engineer
bu iot & graphics solutions

- End of Document -