Status	Final Text
Date of Last Update	2015/01/06
Person Assigned	David Clunie mailto:dclunie@dclunie.com
Submitter Name	Mathieu Malaterre mailto:mathieu.malaterre@gmail.com
Submission Date	2014/06/21

Correction Number CP-1422
Log Summary: Correct High Bit value description in various IODs
Name of Standard PS3.3, PS3.5 2014c
Rationale for Correction: High Bit for all DICOM Image IODs (since DICOM was first introduced in 1993) has always been one less than Bits Stored, and Bits Allocated has always been a multiple of 8, or 1 for single bit (binary) images, such as are used in the Multi-frame Single Bit SC Image and Segmentation IODs and Icon Image Sequences. The only exception is the original Secondary Capture Image IOD, in which there were no such constraints. The earlier ACR-NEMA standards permitted "packing" pixels across word boundaries, or indeed arbitrary alignment, as well as arbitrary word length, and description of this persisted into the DICOM standard in PS3.5, even though no modality-specific IODs permitted it. Apart from test cases to exercise this mechanism, no real-world implementations that use it are known, so this theoretical usage can be retired. In addition, a few IODs incorrectly describe High Bit when more than one enumerated value is permitted but do not use the consistent "one less than Bits Stored" language, leading to potentially ambiguities if the "obvious" selection appropriate to 8 or 16 bits is not made. Correct this, by amending PS3.5 and removing examples of packing and other than the "normal" alignment, and specify the constraint in the general macro used by image IODs.
Correction Wording:

Amend PS3.3 Annex C as follows:

C.7.6.3 Image Pixel Module

...

Table C.7-11b specifies the common attributes that describe the pixel data of the image.

Table C.7-11b. Image Pixel Macro Attributes

Attribute Name	Tag	Type	Attribute Description
Bits Allocated	(0028,0100)	1	Number of bits allocated for each pixel sample. Each sample shall have the same number of bits allocated. **Bits Allocated (0028,0100) shall be either 1, or a multiple of 8.** See PS3.5 for further explanation.
Bits Stored	(0028,0101)	1	Number of bits stored for each pixel sample. Each sample shall have the same number of bits stored. See PS3.5 for further explanation.
High Bit	(0028,0102)	1	Most significant bit for pixel sample data. Each sample shall have the same high bit. **High Bit (0028,0102) shall be one less than Bits Stored (0028,0101).** See PS3.5 for further explanation.
...	...	...	....

C.8.3.1 MR Image Module

Table C.8-4 contains the Attributes that describe MR images.

Table C.8-4. MR Image Module Attributes

Attribute Name	Tag	Type	Attribute Description
Bits Allocated	(0028,0100)	1	Number of bits allocated for each pixel sample. Each sample shall have the same number of bits allocated. See Section C.8.3.1.1.4 for specialization.
Bits Stored	**(0028,0101)**	**1**	**Number of bits stored for each pixel sample. Each sample shall have the same number of bits stored.**
High Bit	**(0028,0102)**	**1**	**Most significant bit for pixel sample data. Each sample shall have the same high bit. High Bit (0028,0102) shall be one less than Bits Stored (0028,0101).**

C.8.3.1.1 MR Image Attribute Descriptions

...

C.8.3.1.1.4 Bits Allocated

Enumerated Values:

16

C.8.5.6 US Image Module

Table C.8-18 specifies the Attributes that describe ultrasound images.

Table C.8-18. US Image Module Attributes

Attribute Name	Tag	Type	Attribute Description
Bits Allocated	(0028,0100)	1	Number of bits allocated for each pixel sample. See Section C.8.5.6.1.13 for specialization.
Bits Stored	(0028,0101)	1	Number of bits stored for each pixel sample. See Section C.8.5.6.1.14 for specialization.
High Bit	(0028,0102)	1	Most significant bit for pixel sample data. See Section C.8.5.6.1.15 for specialization.

C.8.5.6.1 US Image Attribute Descriptions

...

C.8.5.6.1.13 Bits Allocated

For US Images, Bits Allocated (0028,0100) is specified to use the following values for specific Photometric Interpretations:

Table C.8-20. US Bits Allocated

Photometric Interpretation	Bits Allocated Value
MONOCHROME2	8
RGB	8
YBR_FULL	8
YBR_FULL_422	8
YBR_PARTIAL_422	8
YBR_RCT	8
YBR_ICT	8
YBR_PARTIAL_420	8
PALETTE COLOR	8 - 8 bit palette, or 16 - 16 bit palette

C.8.5.6.1.14 Bits Stored

For US Images, Bits Stored (0028,0101) **shall be equal to Bits Allocated (0028,0100)**~~**is specified to use the following values for specific Photometric Interpretations:**~~

Table C.8-21. ~~US Bits Stored~~

~~Photometric Interpretation~~	~~Bits Stored Value~~
~~MONOCHROME2~~	~~8~~
~~RGB~~	~~8~~
~~YBR_FULL~~	~~8~~
~~YBR_FULL_422~~	~~8~~
~~YBR_PARTIAL_422~~	~~8~~
~~YBR_RCT~~	~~8~~
~~YBR_ICT~~	~~8~~
~~YBR_PARTIAL_420~~	~~8~~

~~Photometric Interpretation~~	~~Bits Stored Value~~
~~PALETTE COLOR~~	~~8 - 8 bit palette, or~~ ~~16 - 16 bit palette~~

C.8.5.6.1.15 High Bit

For US Images, High Bit (0028,0102) **shall be one less than Bits Stored (0028,0101)**~~is specified to use the following values for specific Photometric Interpretations:~~

Table C.8-22. ~~US High Bit~~

~~Photometric Interpretation~~	~~High Bit Value~~
~~MONOCHROME2~~	~~7~~
~~RGB~~	~~7~~
~~YBR_FULL~~	~~7~~
~~YBR_FULL_422~~	~~7~~
~~YBR_PARTIAL_422~~	~~7~~
~~YBR_RCT~~	~~7~~
~~YBR_ICT~~	~~7~~
~~YBR_PARTIAL_420~~	~~7~~
~~PALETTE COLOR~~	~~7 - 8 bit palette, or~~ ~~15 - 16 bit palette~~

C.8.12.4 VL Whole Slide Microscopy Image Module

Table C.8.12.4-1 specifies the Attributes that describe the VL Whole Slide Microscopy Image Module.

Table C.8.12.4-1. VL Whole Slide Microscopy Image Module Attributes

Attribute Name	Tag	Type	Attribute Description
...	...	...	...
Bits Allocated	(0028,0100)	1	Number of bits allocated for each pixel sample. **Enumerated Values:** **8** **16**
Bits Stored	(0028,0101)	1	Number of bits stored for each pixel sample. **Shall be equal to Bits Allocated (0028,0100).** ~~**Enumerated Values:**~~ ~~**8**~~ ~~**16**~~

Attribute Name	Tag	Type	Attribute Description
High Bit	(0028,0102)	1	Most significant bit for pixel sample data. **High Bit (0028,0102) shall be one less than Bits Stored (0028,0101).** ~~**Enumerated Values:**~~ ~~**7**~~ ~~**15**~~
...	...	...	...

C.8.24.3 Enhanced US Image Module

...

Table C.8.24.3-1. Enhanced US Image Module Attributes

Attribute Name	Tag	Type	Attribute Description
...	...	...	...
Bits Allocated	(0028,0100)	1	Number of bits allocated for each pixel sample. **Enumerated Values:** **8** **16**
Bits Stored	(0028,0101)	1	Number of bits stored for each pixel sample. **Shall be equal to Bits Allocated (0028,0100).** ~~**Enumerated Values:**~~ ~~**8**~~ ~~**16**~~
High Bit	(0028,0102)	1	Most significant bit for pixel sample data. **High Bit (0028,0102) shall be one less than Bits Stored (0028,0101).** ~~**Enumerated Values:**~~ ~~**7**~~ ~~**15**~~
...	...	...	...

C.8.28.2 Ophthalmic Thickness Map Module

Table C.8.28.2-1 specifies the Attributes that describe an Image produced by ophthalmic thickness mapping devices.

Table C.8.28.2-1. Ophthalmic Thickness Map Module Attributes

Attribute Name	Tag	Type	Attribute Description
...	...	...	...

Attribute Name	Tag	Type	Attribute Description
Bits Allocated	(0028,0100)	1	Number of bits allocated for each pixel sample. Each sample shall have the same number of bits allocated. See ???? for further explanation. **Enumerated Values:** **8** **16**
Bits Stored	(0028,0101)	1	Number of bits stored for each pixel sample. **Shall be equal to Bits Allocated (0028,0100).** ~~**Enumerated Values:**~~ ~~**8**~~ ~~**16**~~
High Bit	(0028,0102)	1	Most significant bit for pixel sample data. ~~**Each sample shall have the same high bit.**~~ **High Bit (0028,0102) shall be one less than Bits Stored (0028,0101).** ~~**See ???? for further explanation.**~~ ~~**Enumerated Values:**~~ ~~**7 if Bits Stored (0028,0101) = 8**~~ ~~**15 if Bits Stored (0028,0101) =**~~ ~~**16**~~
...	...	...	...

C.8.30.2 Corneal Topography Map Image Module

Table C.8.30.2-1 specifies the Attributes that describe an Image produced by corneal topography mapping devices.

Table C.8.30.2-1. Corneal Topography Map Image Module Attributes

Attribute Name	Tag	Type	Attribute Description
...	...	...	...
Bits Allocated	(0028,0100)	1	Number of bits allocated for each pixel sample. Each sample shall have the same number of bits allocated. See ???? for further explanation. **Enumerated Values:** **8** **16**

Attribute Name	Tag	Type	Attribute Description
Bits Stored	(0028,0101)	1	Number of bits stored for each pixel sample. **Shall be equal to Bits Allocated (0028,0100).** ~~**Enumerated Values:**~~ ~~**8**~~ ~~**16**~~
High Bit	(0028,0102)	1	Most significant bit for pixel sample data. ~~**Each sample shall have the same high bit.**~~ **High Bit (0028,0102) shall be one less than Bits Stored (0028,0101).** ~~**See ???? for further explanation.**~~ ~~**Enumerated Values:**~~ ~~**7 if Bits Stored (0028,0101) = 8**~~ ~~**15 if Bits Stored (0028,0101) =**~~ ~~**16**~~
...	...	...	...

Amend PS3.5 Section 8 as follows:

8.1.1 Pixel Data Encoding of Related Data Elements

Encoded Pixel Data of various bit depths shall be accommodated. The following three Data Elements shall define the Pixel structure:

- Bits Allocated (0028,0100)
- Bits Stored (0028,0101)
- High Bit (0028,0102)

Each Pixel Cell shall contain a single Pixel Sample Value. The size of the Pixel Cell shall be specified by Bits Allocated (0028,0100). Bits Stored (0028,0101) defines the total number of these allocated bits that will be used to represent a Pixel Sample Value. Bits Stored (0028,0101) shall never be larger than Bits Allocated (0028,0100). High Bit (0028,0102) specifies where the high order bit of the Bits Stored (0028,0101) is to be placed with respect to the Bits Allocated (0028,0100) specification. **Bits Allocated (0028,0100) shall either be 1, or a multiple of 8. High Bit (0028,0102) shall be one less than Bits Stored (0028,0101).**

Note

1. For example, in Pixel Data with 16 bits (2 bytes) allocated, 12 bits stored, and bit ~~**15**~~**11** specified as the high bit, one pixel sample is encoded in each 16-bit word, with the 4 ~~**least**~~**most** significant bits of each word not containing Pixel Data. See Annex D for other examples of the basic encoding schemes.
2. Formerly, bits not used for Pixel Sample Values were described as being usable for overlay planes, but this usage has been retired. See PS3.5-2004.
3. **Formerly, High Bit (0028,0102) was not restricted to being be one less than Bits Stored (0028,0101) in this Part, or in the general case, though almost all Information Object Definitions in PS3.3 imposed such a restriction. See PS3.5 2014c.**
4. **Receiving applications may not assume anything about the contents of unused bits, and in particular may not assume that they are zero, or that they contain sign extension bits.**

Additional r Restrictions **that** are placed on acceptable Values for Bits Allocated (0028,0100), Bits Stored (0028,0101), and High Bit (0028,0102) ~~**and**~~ are specified in the Information Object Definitions in PS3.3. Also, the Value Field containing Pixel Data, like all other Value Fields in DICOM, shall be an even number of bytes in length. This means that the Value Field may need to be padded with

data that is not part of the image and shall not be considered significant. If needed, the padding bits shall be appended to the end of the Value Field, and shall be used only to extend the data to the next even byte increment of length.

In a multi-frame object that is transmitted in Native Format, the individual frames are not padded. The individual frames shall be concatenated and padding bits (if necessary) apply to the complete Value Field.

Note

Receiving applications should be aware that some older applications may send Pixel Data with excess padding, which was not explicitly prohibited in earlier versions of the Standard. Applications should be prepared to accept such Pixel Data elements, but may delete the excess padding. In no case should a sending application place private data in the padding data.

The field of bits representing the value of a Pixel Sample shall be a binary 2's complement integer or an unsigned integer, as specified by the Data Element Pixel Representation (0028,0103). The sign bit shall be the High Bit in a Pixel Sample Value that is a 2's complement integer. The minimum actual Pixel Sample Value encountered in the Pixel Data is specified by Smallest Image Pixel Value (0028,0106) while the maximum value is specified by Largest Image Pixel Value (0028,0107).

Amend PS3.5 Annex D as follows:

D.1 Detailed Example of Pixel Data Encoding

...

An individual pixel may consist of one or more Pixel Sample Values (e.g., color or multi-planar images). Each Pixel Sample Value can be expressed as either a binary 2's complement integer or a binary unsigned integer, as specified by the Pixel Representation Data Element (0028, 0103). The number of bits in each Pixel Sample Value is specified by Bits Stored (0028,0101). For 2's complement integer Pixel Samples the sign bit is the most significant bit of the Pixel Sample Value.

A Pixel Cell is the container for a Pixel Sample Value and optionally additional bits. These additional bits ~~**can be**~~**are** used ~~**for overlay planes, or**~~ to place Pixels on certain boundaries (byte, word, etc.). A Pixel Cell exists for every individual Pixel Sample Value in the Pixel Data. The size of the Pixel Cells is specified by Bits Allocated (0028,0100) and is greater than or equal to the Bits Stored (0028,0101). The placement of the Pixel Sample Values within the Pixel Cells is specified by High Bit (0028,0102).

Any restrictions on the characteristics of a Pixel Cell and the Pixel Sample Value contained therein are specific to the Information Object Definition (e.g., Image Object) containing the Pixel Data Element (see PS3.3).

The Pixel Data Element, as specified by the DICOM Default Transfer Syntax in Section 10.1, has a Value Representation of OW (Other Word String). The Pixel Data in DICOM 3.0, as it was in ACR-NEMA 2.0, is packed**, except that Bits Allocated is always either 1, or a multiple of 8** (see Figure D-2). One way to visualize this packed encoding is to imagine encoding the Pixel Cells as a concatenated stream of bits from the least significant bit of the first Pixel Cell up through the most significant bit of the last Pixel Cell. Within this stream, the most significant bit of any Pixel Cell is followed by the least significant bit of the next Pixel Cell. The Pixel Data can then be broken up into a stream of physical 16-bit words, each of which is subject to the byte ordering constraints of the Transfer Syntax.

All other (non-default) DICOM Transfer Syntaxes make use of explicit VR encoding. For these Transfer Syntaxes, all Pixel Data where Bits Allocated is less than or equal to 8 may be encoded with an explicit VR of OB (see Annex A). As in the OW case, Pixel Cells are packed together, but in this case the Pixel Data is broken up into a stream of physical 8-bit words.

Note

For Pixel Data encoded with an explicit VR of OB, the encoding of the Pixel Data is unaffected by Little Endian or Big Endian byte ordering.

Retain Figure D-2 Encoding (Packing) of Arbitrary Pixel Data with a VR of OW, since it serves to illustrate 1 or 8 Bits Stored within 16 bit OW case

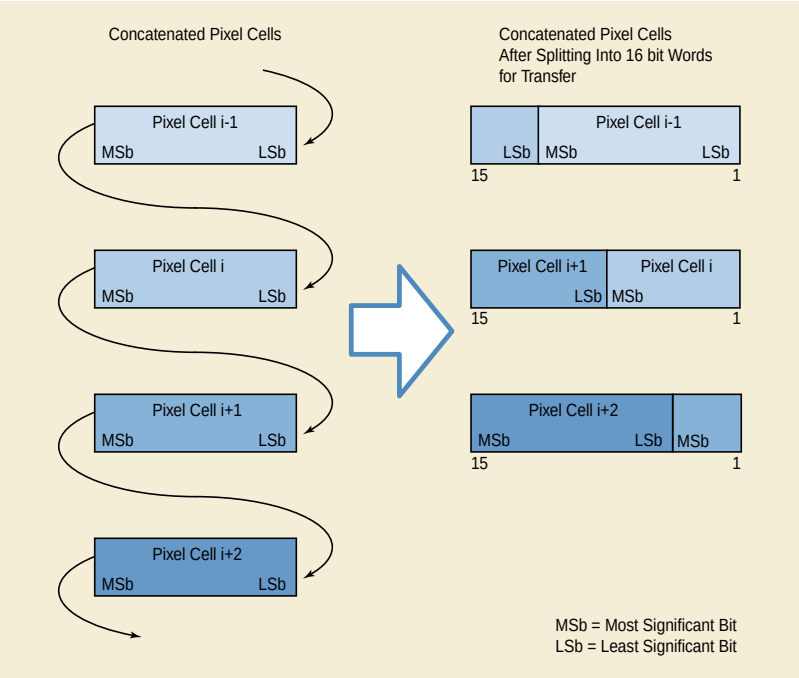


Figure D-2. Encoding (Packing) of Arbitrary Pixel Data with a VR of OW

~~**IODs tend to specify**~~ **With the exception of single bit images,** Pixel Cells ~~**so that they**~~ begin and end on byte or word boundaries and such that the Pixel Sample Value contained within also fits 'neatly' within a cell. ~~**However, this does not have to be the case.**~~

Figure D-3 is an example~~**We now carry forward two examples**~~ of Pixel Data encoding using the Value representation of OW for the purposes of clarification. **The** Example **is**~~**1 will be**~~ a valid example for a CT Image Information Object~~**, while Example 2 will be for a hypothetical information object (see Figure D-3)**~~.

Replace Figure D-3

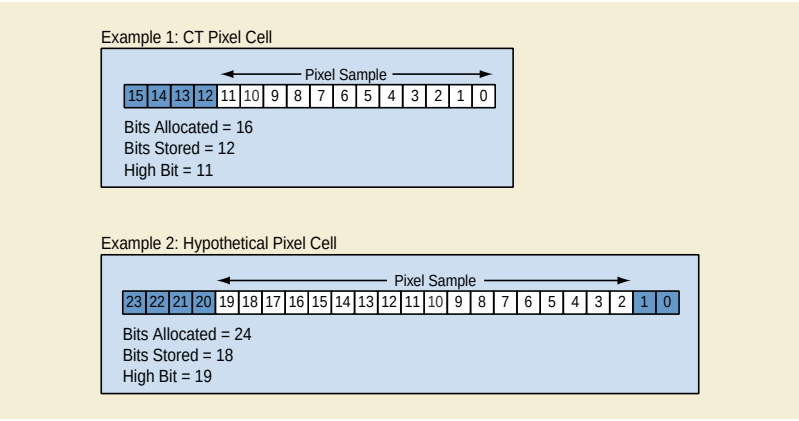


Figure D-3. Example Pixel Cells

with the single CT example and remove the hypothetical example

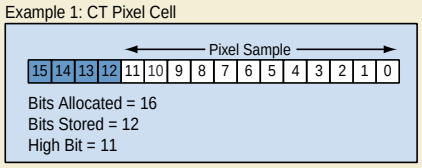


Figure D-3. Example Pixel Cells

Replace Figure D-4

Figure D-4 shows Pixel Data constructed of these example Pixel Cells as they are packed into a stream of 16-bit words.

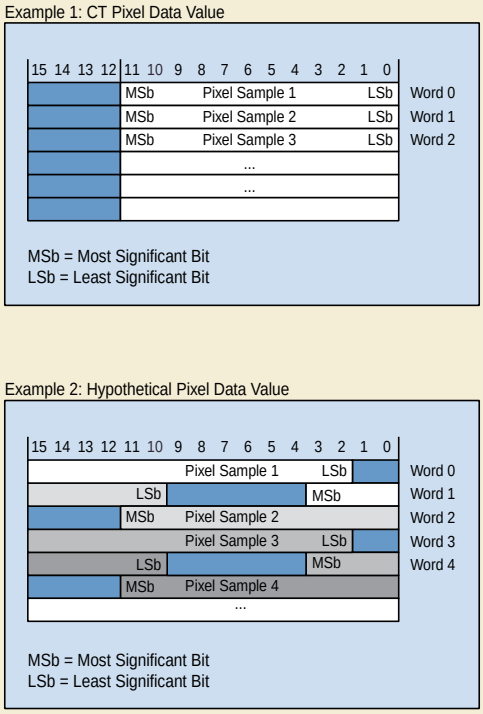


Figure D-4. Example Pixel Cells Packed into 16-bit Words (VR = OW)

with the single CT example and remove the hypothetical example

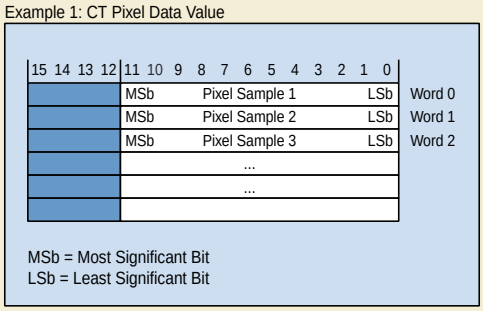


Figure D-4. Example Pixel Cells Packed into 16-bit Words (VR = OW)

Byte ordering becomes a consideration when we represent the Pixel Data physically, in memory, a file, or on a network.

In the memory of a byte-addressable Big Endian machine, the highest order byte (bits 8 - 15) in each 16-bit word has a binary address of x...x0. While in a byte-addressable Little Endian machine, the lowest order byte (bits 0 - 7) in each 16-bit word has a binary address of x...x0. Figure D-5 pictures our example Pixel Data streams as they would be addressed in the memory of both a Big Endian and a Little Endian machine.

Replace Figure D-5

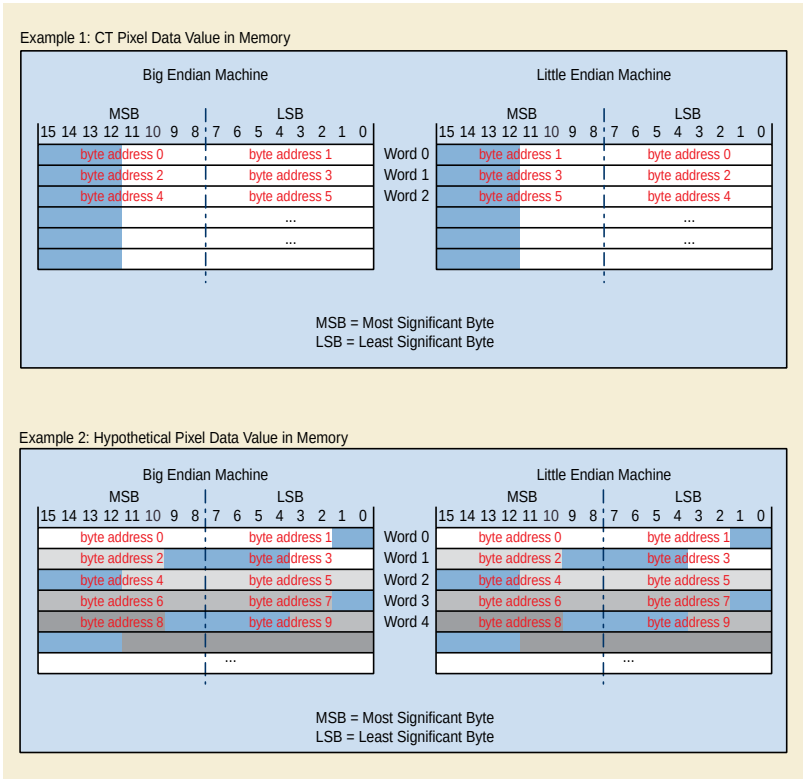


Figure D-5. Example Pixel Cells Byte Ordered in Memory (VR = OW)

with the single CT example and remove the hypothetical example

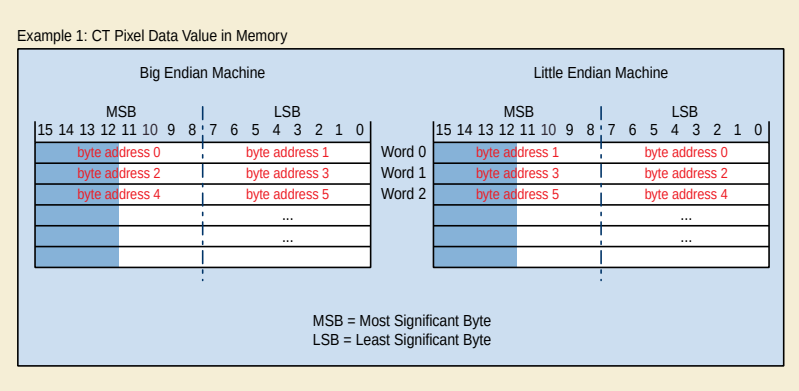


Figure D-5. Example Pixel Cells Byte Ordered in Memory (VR = OW)

Byte ordering is also specified as part of the negotiated Transfer Syntax used in the exchange of a DICOM message. Sixteen bit words are transmitted across the network (a byte at a time) least significant byte first in the case of a Little Endian Transfer Syntax and most significant byte first when using a Big Endian Transfer Syntax (see Figure D-6).

Replace Figure D-6

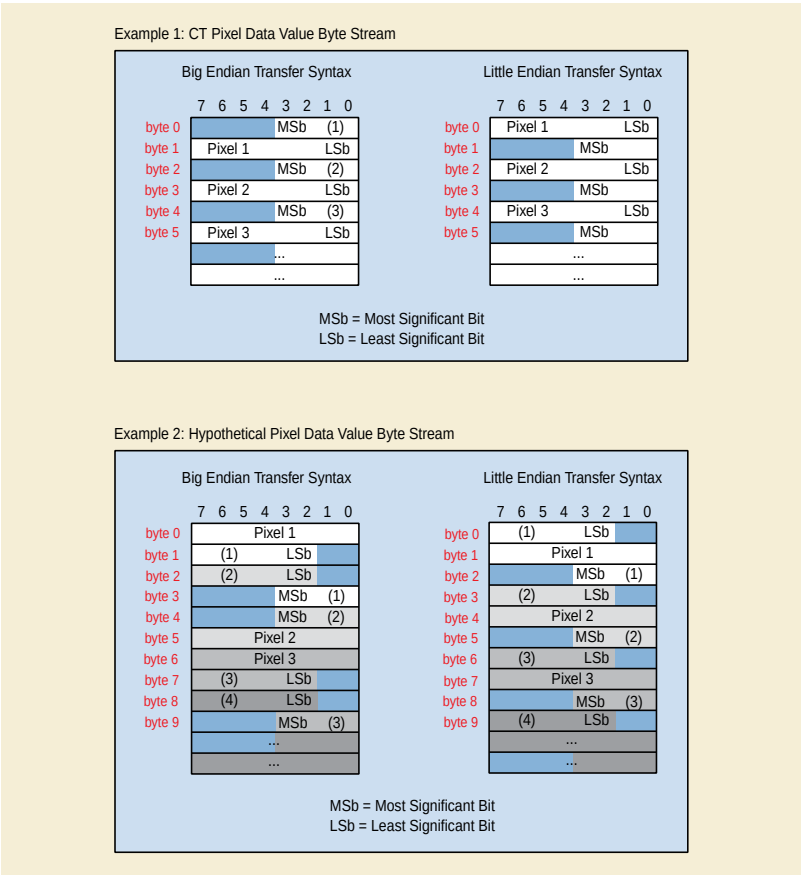


Figure D-6. Sample Pixel Data Byte Streams (VR = OW)

with the single CT example and remove the hypothetical example

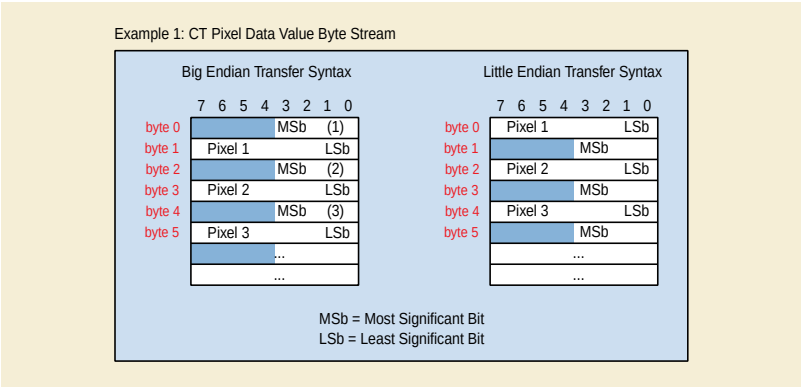


Figure D-6. Sample Pixel Data Byte Streams (VR = OW)

...

D.2 Various Additional Examples of Pixel and Overlay Data Cells

The following examples further illustrate the use of the data elements for Bits Allocated (0028,0100), Bits Stored (0028,0101) and High Bit (0028,0102) in the encoding of Pixel and Overlay Data. All examples show sample Pixel Cells before being encoded in byte streams (and before being affected by a particular Transfer Syntax).

Replace Figure D.2-1

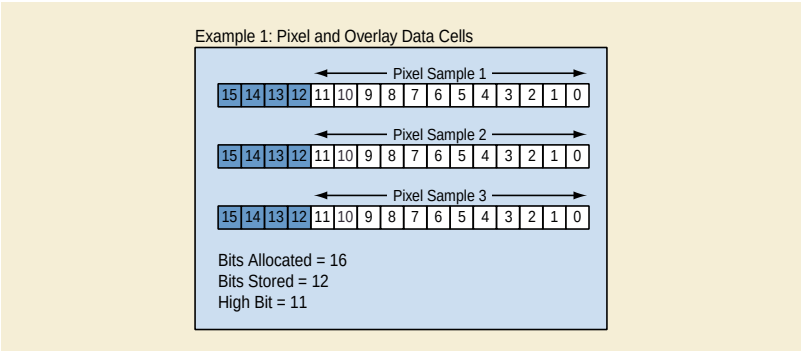
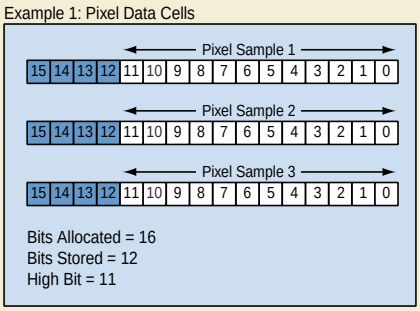


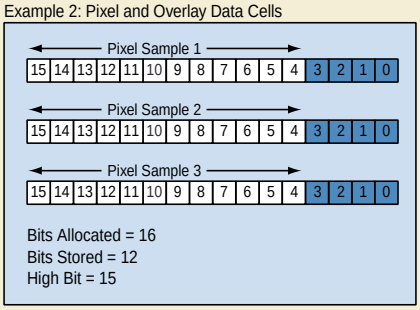
Figure D.2-1. Example 1 of Pixel and Overlay Data Cells

1 *with new version that does not mention overlays*



2 **Figure D.2-1. Example 1 of Pixel Data Cells**

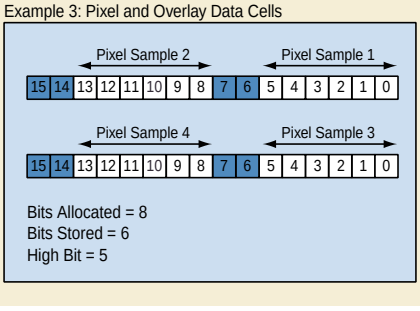
3 *Delete Figure D.2-2 Example 2 of Pixel and Overlay Data Cells*



4 **Figure D.2-2. Example 2 of Pixel and Overlay Data Cells**

5 **Figure D.2-2 Example 2 of Pixel and Overlay Data Cells has been retired. See PS3.3 2014c.**

6 *Replace Figure D.2-3*



7 **Figure D.2-3. Example 3 of Pixel and Overlay Data Cells**

8 *with new version that does not mention overlays*

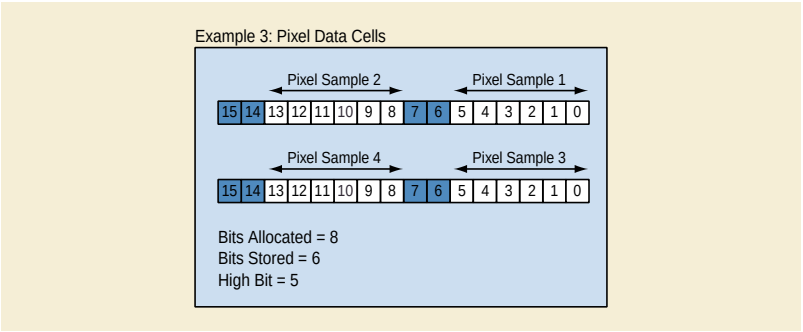


Figure D.2-3. Example 3 of Pixel Data Cells

...

Leave Figure D.2-4 alone

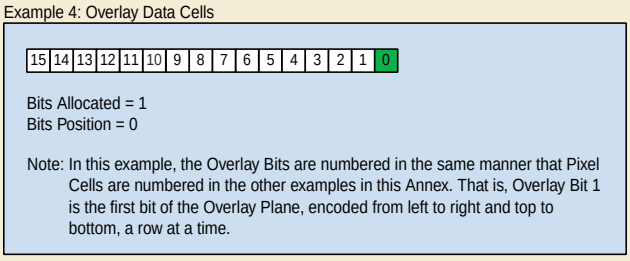


Figure D.2-4. Example 4 of Overlay Data Cells

Add new Figure D.2-5 to illustrate single bit (binary) pixel data encoding

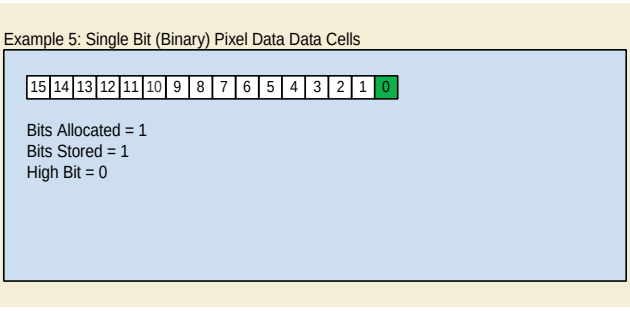


Figure D.2-5. Example 5 of Single Bit Pixel Data Cells (VR=OW)