Doc. TSG-800-4r0
26 September 2006

Working Draft Partitioning of ATSC Digital Television Standard (A/53) Revision E Part 4 – MPEG-2 Video System Characteristics

Advanced Television Systems Committee
1750 K Street, N.W.
Suite 1200
Washington, D.C. 20006
www.atsc.org

The Advanced Television Systems Committee, Inc., is an international, non-profit organization developing voluntary standards for digital television. The ATSC member organizations represent the broadcast, broadcast equipment, motion picture, consumer electronics, computer, cable, satellite, and semiconductor industries.

Specifically, ATSC is working to coordinate television standards among different communications media focusing on digital television, interactive systems, and broadband multimedia communications. ATSC is also developing digital television implementation strategies and presenting educational seminars on the ATSC standards.

ATSC was formed in 1982 by the member organizations of the Joint Committee on InterSociety Coordination (JCIC): the Electronic Industries Association (EIA), the Institute of Electrical and Electronic Engineers (IEEE), the National Association of Broadcasters (NAB), the National Cable Television Association (NCTA), and the Society of Motion Picture and Television Engineers (SMPTE). Currently, there are approximately 140 members representing the broadcast, broadcast equipment, motion picture, consumer electronics, computer, cable, satellite, and semiconductor industries.

ATSC Digital TV Standards include digital high definition television (HDTV), standard definition television (SDTV), data broadcasting, multichannel surround-sound audio, and satellite direct-to-home broadcasting.

Table of Contents

Index of Tables and Figures

ATSC Digital Television Standard – Part 4: MPEG-2 Video System Characteristics

1. SCOPE

This Part describes the characteristics of the video subsystem of the Digital Television Standard. The input formats and bit stream characteristics are described in separate sections.[1]

2. REFERENCES

2.1 Normative References

The following documents contain provisions which, through reference in this text, constitute provisions of this standard. At the time of publication, the editions indicated were valid. All standards are subject to revision, and parties to agreement based on this standard are encouraged to investigate the possibility of applying the most recent editions of the documents listed below.

[1] CEA-708-B: "Digital Television (DTV) Closed Captioning," Consumer Electronics Association.

[2] ISO/IEC IS 13818-1:2000 (E), International Standard, Information technology – Generic coding of moving pictures and associated audio information: systems.

[3] ISO/IEC IS 13818-2:2000 (E), International Standard, Information technology – Generic coding of moving pictures and associated audio information: video.

[4] SMPTE 125M (1995): Standard for Television—Component Video Signal 4:2:2, Bit-Parallel Digital Interface, Society of Motion Picture and Television Engineers, White Plains, N.Y.

[5] SMPTE 170M (2004): Standard for Television—Composite Analog Video Signal, NTSC for Studio Applications, Society of Motion Picture and Television Engineers, White Plains, N.Y.

[6] SMPTE 267M (1995): Standard for Television—Bit-Parallel Digital Interface, Component Video Signal 4:2:2 16 x 9 Aspect Ratio, Society of Motion Picture and Television Engineers, White Plains, N.Y.

[7] SMPTE 274M (2005): Standard for Television—1920 x 1080 Scanning and Analog and Parallel Digital Interfaces for Multiple Picture Rates, Society of Motion Picture and Television Engineers, White Plains, N.Y.

[8] SMPTE 293M (2003): Standard for Television—720 x 483 Active Line at 59.94-Hz Progressive Scan Production, Digital Representation, Society of Motion Picture and Television Engineers, White Plains, N.Y.

[1] Note that there is a coordinated effort underway among ATSC, CEA, and SMPTE to revise and clarify standards related to delivering closed captions so that each describes the aspects of the system for which they are primarily responsible without overlap. This effort is expected to result in revisions of those sections in the ATSC Standards.

[9] SMPTE 296M (2001): Standard for Television—1280 x 720 Progressive Image Sample Structure, Analog and Digital Representation and Analog Interface, Society of Motion Picture and Television Engineers, White Plains, N.Y.

[10] ETSI TS 101 154 V1.7.1, Digital Video Broadcasting (DVB): Implementation Guidelines for the use of MPEG-2 Systems, Video and Audio in Satellite, Cable and Terrestrial Broadcasting Applications, Annex B, June 2005.

2.2 Informative References

[11] Digital Receiver Implementation Guidelines and Recommended Receiver Reaction to Aspect Ratio Signaling in Digital Video Broadcasting, Issue 1.2.1, February 2001, Digital TV Group.

[12] ITU-R BT.601-5 (1994): Encoding Parameters of Digital Television for Studios.

[13] ITU-R BT. 709-5 (2002): Parameter values for the HDTV Standards for Production and International Programme Exchange.

[14] ANSI/SCTE 21 2001 (formerly DVS 053): Standard for Carriage of NTSC VBI Data in Cable Digital Transport Streams, Society of Cable Telecommunications Engineers.

[15] CEA-CEB-10-A: "EIA-708-B Implementation Guidance," December 2002, Consumer Electronics Association.

[16] CEA-CEB16: "Active Format Description (AFD) & Bar Data Recommended Practice," Consumer Electronics Association, Arlington, VA, 31 July 2006.

[17] SMPTE: "Proposed SMPTE 2016-1 [in development]: Standard for Television—Format for Active Format Description and Bar Data," Society of Motion Picture and Television Engineers, White Plains, NY.

3. COMPLIANCE NOTATION

As used in this document, "shall" denotes a mandatory provision of the standard. "Should" denotes a provision that is recommended but not mandatory. "May" denotes a feature whose presence does not preclude compliance, that may or may not be present at the option of the implementor.

4. SYSTEM OVERVIEW (INFORMATIVE)

A basic block diagram representation of the system is shown in Figure 4.1. According to this model, the digital television system can be seen to consist of three subsystems.

- Source coding and compression
- Service multiplex and transport
- RF/transmission

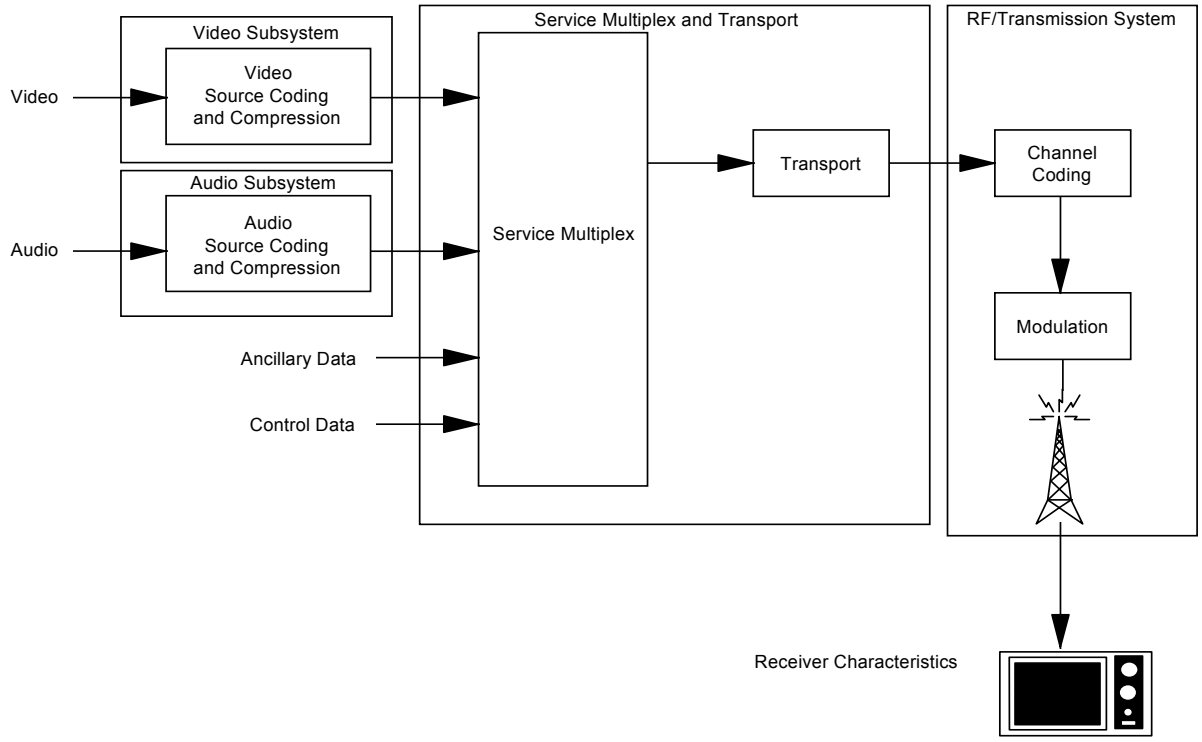

Figure 4.1 ITU-R digital terrestrial television broadcasting model.

Figure 4.2 illustrates a high level view of the encoding equipment. This view is not intended to be complete, but is used to illustrate the relationship of various clock frequencies within the encoder.

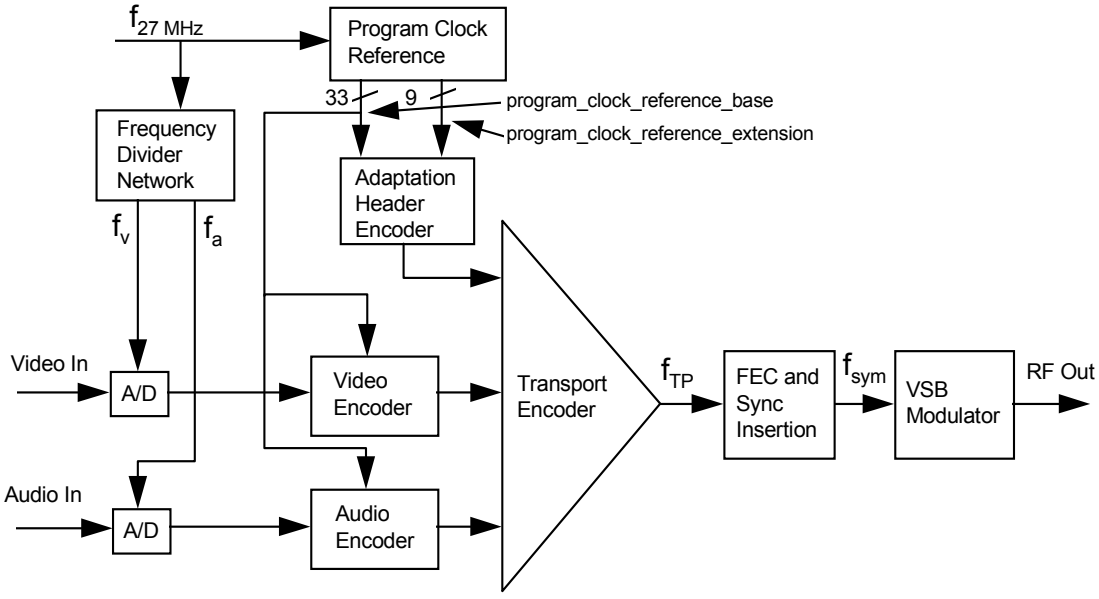

Figure 4.2 High level view of encoding equipment.

The source coding domain, represented schematically by the video, audio, and transport encoders, uses a family of frequencies which are based on a 27 MHz clock (f_{27MHz}). This clock is used to generate a 42-bit sample of the frequency which is partitioned into two parts defined by the MPEG-2 specification. These are the 33-bit program_clock_reference_base and the 9-bit program_clock_reference_extension. The former is equivalent to a sample of a 90 kHz clock which is locked in frequency to the 27 MHz clock, and is used by the audio and video source encoders when encoding the presentation time stamp (PTS) and the decode time stamp (DTS). The audio and video sampling clocks, f_a and f_v respectively, are frequency-locked to the 27 MHz clock. This can be expressed as the requirement that there exist two pairs of integers, (n_a, m_a) and (n_v, m_v), such that

$$f_a = \frac{n_a}{m_a} \times 27 \text{ MHz}$$

and

$$f_v = \frac{n_v}{m_v} \times 27 \text{ MHz}$$

5. POSSIBLE VIDEO INPUTS

While not required by this standard, there are certain television production standards, shown in Table 5.1, that define video formats that relate to compression formats specified by this standard.

Table 5.1 Standardized Video Input Formats

Video Standard	Active Lines	Active Samples/ Line
SMPTE 274M [7]	1080	1920
SMPTE 296M [9]	720	1280
ITU-R BT.601-5 [12]	483	720

The compression formats may be derived from one or more appropriate video input formats. It may be anticipated that additional video production standards will be developed in the future that extend the number of possible input formats.

6. SOURCE CODING SPECIFICATION

The DTV video compression algorithm shall conform to the Main Profile syntax of ISO/IEC 13818-2 [3]. The allowable parameters shall be bounded by the upper limits specified for the Main Profile at High Level.[2] Additionally, all bit streams shall meet the constraints and specifications described in Sections 6.1 and 6.2.

6.1 Constraints with Respect to ISO/IEC 13818-2 Main Profile

The following tables list the allowed values for each of the ISO/IEC 13818-2 [3] syntactic elements which are restricted beyond the limits imposed by MP@HL.

[2] See ISO/IEC 13818-2 [A3], Section 8 for more information regarding profiles and levels.

In these tables conventional numbers denote decimal values, numbers preceded by 0x are to be interpreted as hexadecimal values and numbers within single quotes (e.g., '10010100') are to be interpreted as a string of binary digits.

6.1.1 Sequence Header Constraints

Table 6.1 identifies parameters in the sequence header of a bit stream that shall be constrained by the video subsystem and lists the allowed values for each.

Table 6.1 Sequence Header Constraints

Sequence Header Syntactic Element	Allowed Value
horizontal_size_value	see Table 6.2
vertical_size_value	see Table 6.2
aspect_ratio_information	see Table 6.2
frame_rate_code	see Table 6.2
bit_rate_value (≤ 19.4 Mbps)	≤ 48500
bit_rate_value (≤ 38.8 Mbps)	≤ 97000
vbv_buffer_size_value	≤ 488

The allowable values for the field bit_rate_value are application-dependent. In the primary application of terrestrial broadcast, this field shall correspond to a bit rate which is less than or equal to 19.4 Mbps. In the high data rate mode, the corresponding bit rate is less than or equal to 38.8 Mbps.

6.1.2 Compression Format Constraints

Table 6.2 lists the allowed compression formats.

Table 6.2 Compression Format Constraints

vertical_size_value	horizontal_size_value	aspect_ratio_information	frame_rate_code	progressive_sequence
1080[3]	1920	1,3	1,2,4,5	'1'
			4,5	'0'
720	1280	1,3	1,2,4,5,7,8	'1'
480	704	2,3	1,2,4,5,7,8	'1'
			4,5	'0'
	640	1,2	1,2,4,5,7,8	'1'
			4,5	'0'
Legend for MPEG-2 coded values:				
aspect_ratio_information: 1 = square samples, 2 = 4:3 display aspect ratio, 3 = 16:9 display aspect ratio				
frame_rate_code: 1 = 23.976 Hz, 2 = 24 Hz, 4 = 29.97 Hz, 5 = 30 Hz, 7 = 59.94 Hz, 8 = 60 Hz				
progressive_sequence: '0' = interlaced scan, '1' = progressive scan				

[3] Note that 1088 lines are actually coded in order to satisfy the MPEG-2 requirement that the coded vertical size be a multiple of 16 (progressive scan) or 32 (interlaced scan). The bottom 8 lines are black, per MPEG rules.

6.1.3 Sequence Extension Constraints

Table 6.3 identifies parameters in the sequence extension part of a bit stream that shall be constrained by the video subsystem and lists the allowed values for each. A sequence_extension structure is required to be present after every sequence_header structure.

Table 6.3 Sequence Extension Constraints

Sequence Extension Syntactic Element	Allowed Values
progressive_sequence	see Table 3
profile_and_level_indication	see Note
chroma_format	'01'
horizontal_size_extension	'00'
vertical_size_extension	'00'
bit_rate_extension	'0000 0000 0000'
vbv_buffer_size_extension	'0000 0000'
frame_rate_extension_n	'00'
frame_rate_extension_d	'0000 0'

Note: The profile_and_level_indication field shall indicate the lowest profile and level defined in ISO/IEC 13818-2 [3], Section 8, that is consistent with the parameters of the video elementary stream.

6.1.4 Sequence Display Extension Constraints

Table 6.4 identifies parameters in the sequence display extension part of a bit stream that shall be constrained by the video subsystem and lists the allowed values for each.

Table 6.4 Sequence Display Extension Constraints

Sequence Display Extension Syntactic Element	Allowed Values
video_format	'000'

The values for color_primaries, transfer_characteristics, and matrix_coefficients shall be explicitly indicated in the sequence_display_extension. While all values for color_primaries, transfer_characteristics, and matrix_coefficients defined in Tables 6-7, 6-8, and 6-9 of ISO/IEC 13818-2 [3] are allowed in the transmitted bit stream, it is noted that those of ITU-R BT.709 [13] and SMPTE 170M [5] are the most likely to be in common use.

Note: Some previously-encoded legacy material may not have the colorimetry (i.e., color_primaries, transfer_characteristics, and matrix_coefficients) explicitly indicated in the sequence_display_extension, in which case the colorimetry is most likely ITU-R BT.709 [13] for all formats except those formats with vertical_size_value = 480, which are most likely to have colorimetry according to SMPTE 170M [5].

6.1.5 Picture Header Constraints

In all cases other than when vbv_delay has the value 0xFFFF, the value of vbv_delay shall be constrained as follows:

$$vbv\_delay \leq 45000$$

6.1.6 Picture Coding Constraints

The value frame_pred_frame_dct shall be '1' if progressive_frame is '1'.

6.2 Bit Stream Specifications Beyond MPEG-2

This section covers the extension and user data part of the video syntax. These data are inserted at the sequence, GOP, and picture level. The syntax used for the insertion of closed captioning[4] in picture user data is described.

6.2.1 Picture Extension and User Data Syntax

The picture user data shall be constructed per [3]. Table 6.5 is provided to show the syntax that is required for picture extension and user data.

Table 6.5 Picture Extension and User Data Syntax

Value	No. of Bits	Format
extension_and_user_data(2) {		
while ((nextbits() == extension_start_code) || (nextbits() == user_data_start_code)) {		
if (nextbits() == extension_start_code)		
extension_data(2)		
if (nextbits() == user_data_start_code)		
user_data()		
}		
}		

6.2.2 Picture User Data Syntax

Table 6.6 describes the picture user data syntax that shall be used.

[4] Implementers should note that CEA-708-B [A1] describes the semantics for closed captions. Additional information on implementation of captions may be found in CEA-CEB-10-A [15].

Table 6.6 Picture User Data Syntax[5]

Syntax	No. of Bits	Format
user_data() {		
user_data_start_code	32	bslbf
ATSC_identifier	32	bslbf
user_data_type_code	8	uimsbf
if (user_data_type_code == '0x03')		
cc_data()		
else if (user_data_type_code == '0x06')		
bar_data()		
else {		
while (nextbits() != '0000 0000 0000 0000 0000 0001')		
ATSC_reserved_user_data	8	
}		
next_start_code()		
}		

In accordance with the bit stream syntax in Table 6.5, more than one picture user data construct may follow any given picture header. However, no more than one picture user data construct using the same user_data_type_code shall follow any given picture header.

Note that picture user data with a 32-bit field following user_data_start_code having a value other than ATSC_identifier may be present in an ATSC-compliant video bit stream. As an example, the afd_identifier (value 0x44544731) is defined for use in ATSC video Elementary Streams (see Section 6.2.4). Receiving devices are expected to process this field and use it to determine the syntax and semantics of the user data construct to follow.

Receiving devices are expected to silently discard any unrecognized video user data encountered in the video bit stream. For example, if an unrecognized 32-bit identifier is seen following the user_data_start_code, or an unrecognized 8-bit user_data_type_code is seen following the ATSC_identifier, data should be discarded until another start code is seen.

6.2.3 ATSC Picture User Data Semantics

user_data_start_code – This is set to 0x0000 01B2.

ATSC_identifier – This is a 32 bit code that indicates that the video user data conforms to this specification. The value ATSC_identifier shall be 0x4741 3934.

user_data_type_code – An 8-bit value that identifies the type of ATSC user data to follow. Value 0x03 indicates cc_data(), value 0x06 indicates bar_data(), and other values are either in use in other standards or are reserved for future use.

cc_data() – A data structure defined in Table 6.7.

[5] Shaded cells in this table indicate syntactic and semantic additions to the ISO/IEC 13818-2 [A3] Standard. Note: user_data_type_code values 0x04 and 0x05 are assigned in ANSI/SCTE 21 2001 [14].

bar_data() – A data structure defined in Table 6.8 indicating the sizes of letterbox or pillarbox areas within the coded video frame .

ATSC_reserved_user_data – Reserved for use by ATSC or used by other standards.

6.2.3.1 Captioning Data

Table 6.7 describes the syntax of captioning data.

Table 6.7 Captioning Data Syntax

Syntax	No. of Bits	Format
cc_data() {		
reserved	1	'1'
process_cc_data_flag	1	bslbf
additional_data_flag	1	bslbf
cc_count	5	uimsbf
reserved	8	'1111 1111'
for (i=0 ; i < cc_count ; i++) {		
marker_bits	5	'1111 1'
cc_valid	1	bslbf
cc_type	2	bslbf
cc_data_1	8	bslbf
cc_data_2	8	bslbf
}		
marker_bits	8	'1111 1111'
if (additional_data_flag) {		
while (nextbits() != '0000 0000 0000 0000 0000 0001') {		
additional_cc_data		
}		
}		
}		

process_cc_data_flag – This flag is set to indicate whether it is necessary to process the cc_data. If it is set to '1', the cc_data has to be parsed and its meaning has to be processed. When it is set to '0', the cc_data can be discarded.

additional_data_flag – This flag is set to '1' to indicate the presence of additional user data.

cc_count: This 5-bit integer indicates the number of closed caption constructs following this field. It can have values 0 through 31. The value of cc_count shall be set according to the frame rate and coded picture structure (field or frame) such that a fixed bandwidth of 9600 bits per second is maintained for the closed caption payload data. Sixteen (16) bits of closed caption payload data are carried in each pair of the fields cc_data_1 and cc_data_2.

cc_valid – This flag is set to '1' to indicate that the two closed caption data bytes that follow are valid. If set to '0' the two data bytes are invalid, as defined in CEA-708-B [1].

cc_type – Denotes the type of the two closed caption data bytes that follow, as defined in CEA-708-B [1].

cc_data_1 – The first byte of a closed caption data pair as defined in CEA-708-B [1].

cc_data_2 – The second byte of a closed caption data pair as defined in CEA-708-B [1].

additional_cc_data – Reserved for future ATSC definition.

6.2.3.2 Bar Data

Table 6.8 describes the syntax of bar data. Bar data should be included in video user data whenever the rectangular picture area containing useful information does not extend to the full height or width of the coded frame6 and AFD alone is insufficient to describe the extent of the image. See Section 6.2.4.

When present, bar data shall be carried in the data structure bar_data(), within the picture user data syntax as shown in Table 6.6. After any sequence_header() such bar data shall appear before the next picture_data() within extension_and_user_data(2). After introduction, such bar data shall remain in effect until:

1) the next sequence_header(), or
2) extension_and_user_data(2) containing a bar_data() structure which contains new bar data, or
3) extension_and_user_data(2) containing AFD per Section 6.2.4.

After any sequence_header(), unless AFD data is present specifying otherwise, the absence of bar data shall indicate that the rectangular picture area containing useful information extends to the full height and width of the coded frame.

Bar data is constrained (below) to be signalled in pairs, either top and bottom bars or left and right bars, but not both pairs at once. Bars may be unequal in size. One bar of a pair may be zero width or height.

[6] In other words, the video is letterboxed (bars above and/or below video) or pillarboxed (bars left and/or right of video).

Table 6.8 Bar Data Syntax

Syntax	No. of Bits	Format
bar_data() {		
top_bar_flag	1	bslbf
bottom_bar_flag	1	bslbf
left_bar_flag	1	bslbf
right_bar_flag	1	bslbf
reserved	4	'1111'
if (top_bar_flag == '1') {		
marker_bits	2	'11'
line_number_end_of_top_bar	14	uimsbf
}		
if (bottom_bar_flag == '1') {		
marker_bits	2	'11'
line_number_start_of_bottom_bar	14	uimsbf
}		
if (left_bar_flag == '1') {		
marker_bits	2	'11'
pixel_number_end_of_left_bar	14	uimsbf
}		
if (right_bar_flag == '1') {		
marker_bits	2	'11'
pixel_number_start_of_right_bar	14	uimsbf
}		
marker_bits	8	'1111 1111'
while (nextbits() != '0000 0000 0000 0000 0000 0001') {		
additional_bar_data		
}		
}		

Designation of line numbers for line_number_end_of_top_bar and line_number_start_of_bottom_bar is video format-dependent and shall conform to the applicable standard indicated in Table 6.9.

Note: The range of line numbers and pixels within the coded frame for each image format is specified in Table 2 of SMPTE 2016-1 [17].

Table 6.9 Line Number Designation

Video Format	Applicable Standard
480 Interlaced 4:3	SMPTE 125M [4]
480 Interlaced 16:9	SMPTE 267M [6]
480 Progressive	SMPTE 293M [8]
720 Progressive	SMPTE 296M [9]
1080 Interlaced	SMPTE 274M [7]
1080 Progressive	SMPTE 274M [7]

top_bar_flag – This flag shall indicate, when set to '1', that the top bar data is present. If left_bar_flag is '1', this flag shall be set to '0'.

bottom_bar_flag – This flag shall indicate, when set to '1', that the bottom bar data is present. This flag shall have the same value as top_bar_flag.

left_bar_flag – This flag shall indicate, when set to '1', that the left bar data is present. If top_bar_flag is '1', this flag shall be set to '0'.

right_bar_flag – This flag shall indicate, when set to '1', that the right bar data is present. This flag shall have the same value as left_bar_flag.

line_number_end_of_top_bar – A 14-bit unsigned integer value representing the last line of a horizontal letterbox bar area at the top of the reconstructed frame. Designation of line numbers shall be as defined per each applicable standard in Table 6.9.

line_number_start_of_bottom_bar – A 14-bit unsigned integer value representing the first line of a horizontal letterbox bar area at the bottom of the reconstructed frame. Designation of line numbers shall be as defined per each applicable standard in Table 6.9.

pixel_number_end_of_left_bar – A 14-bit unsigned integer value representing the last horizontal luminance sample of a vertical pillarbox bar area at the left side of the reconstructed frame. Pixels shall be numbered from zero, starting with the leftmost pixel.

pixel_number_start_of_right_bar – A 14-bit unsigned integer value representing the first horizontal luminance sample of a vertical pillarbox bar area at the right side of the reconstructed frame. Pixels shall be numbered from zero, starting with the leftmost pixel.

additional_bar_data – Reserved for future ATSC definition.

6.2.3.2.1 Recommended Receiver Response to Bar Data

Receiving device designers are strongly encouraged to study Consumer Electronics Association (CEA) bulletin CEB16 [16], which contains recommendations regarding the processing of bar data.

6.2.4 Active Format Description Data

Active Format Description (AFD) should be included in video user data whenever the rectangular picture area containing useful information does not extend to the full height or width of the coded frame. AFD data may also be included in user data when the rectangular picture area containing useful information extends to the full height and width of the coded frame.

When present, the AFD shall be carried using the syntax defined in [10], in extension_and_user_data(2) in the MPEG-2 video Elementary Stream. After any sequence_header() the default aspect ratio of the area of interest shall be that signaled by the parameters in the sequence_header() and sequence_display_extension() structures. After any sequence_header() the AFD, when present, shall appear before the next picture_data(). After introduction, such an AFD shall remain in effect until the next sequence_header() or until a new AFD is introduced.

> Note: The AFD syntax as shown in Section 6.4.2.1 is identical to that specified in ETSI TS 101 154 V1.7.1 [10], and is reprinted here with permission. Semantics are documented in Section 6.2.4.2; some are intentionally different from those in ETSI 101 154.

6.2.4.1 AFD Syntax

Table 6.10 shows the syntax defined in [10] which is provided for the convenience of the reader.

Table 6.10 Active Format Description Syntax

Syntax	No. of Bits	Format
user_data_start_code	32	bslbf
afd_identifier	32	bslbf
zero	1	'0'
active_format_flag	1	bslbf
reserved	6	'00 0001'
if (active_format_flag == '1') {		
reserved	4	'1111'
active_format	4	bslbf
}		

6.2.4.2 AFD Semantics

afd_identifier – A 32-bit field that identifies that the syntax of the user data is Active Format Description. Its value is 0x44544731.

active_format_flag – A 1 bit flag. A value of '1' indicates that an active format is described in this data structure.

active_format – A 4 bit field describing the "area of interest" in terms of its aspect ratio within the coded frame as defined in ISO/IEC 13818-2 [3].

The active_format is used by the decoder in conjunction with the "source aspect ratio." The source aspect ratio is derived from the "display aspect ratio" (DAR) signaled in the aspect_ratio_information, the horizontal_size, vertical_size, and display_horizontal_size and display_vertical_size if present (see ISO/IEC 13818-2 [3]):

- If sequence_display_extension() is not present, source aspect ratio = DAR
- If sequence_display_extension() is present, source aspect ratio = $\mathrm{DAR} \times \frac{\text{display\_horizontal\_size}}{\text{display\_vertical\_size}} \times \frac{\text{vertical\_size}}{\text{horizontal\_size}}$

The combination of source aspect ratio and active_format allows the decoder to identify whether the "area of interest" is the whole of the frame (e.g. source aspect ratio 16:9, active_format 16:9 center), a letterbox within the frame (e.g. source aspect ratio 4:3, active_format 16:9 center), or a "pillarbox" within the frame (e.g. source aspect ratio 16:9, active_format 4:3 center).

Table 6.11 defines the coding of the active_format field that shall be used.

Table 6.11 Active Format

active_format	Description	
	4:3 coded frames	**16:9 coded frames**
'0000'	undefined (see below)	undefined (see below)
'0001'	Reserved	Reserved
'0010' – '0011"	Not recommended	Not recommended
0100	Aspect ratio greater than 16:9 (see below)	Aspect ratio greater than 16:9 (see below)
'0101' – '0111'	Reserved	Reserved
'1000'	4:3 full frame image	16:9 full frame image
'1001'	4:3 full frame image	4:3 pillarbox image
'1010'	16:9 letterbox image	16:9 full frame image
'1011'	14:9 letterbox image	14:9 pillarbox image
'1100'	Reserved	Reserved
'1101'	4:3 full frame image, alternative 14:9 center	4:3 pillarbox image, alternative 14:9 center
'1110'	16:9 letterbox image, alternative 14:9 center	16:9 full frame image, alternative 14:9 center
'1111'	16:9 letterbox image, alternative 4:3 center	16:9 full frame image, alternative 4:3 center

AFD '0000' indicates that information is not available and is undefined. Unless bar data is available, DTV receivers and video equipment should interpret the active image area as being the same as that of the coded frame.

AFD '0000', when accompanied by bar data, signals that the image's aspect ratio is narrower than 16:9, but is not either 4:3 or 14:9. The bar data should be used to determine the extent of the image.

AFD '0100', which should be accompanied by bar data, signals that the image's aspect ratio is wider than 16:9, as is typically the case with widescreen features. The bar data should be used to determine the height of the image.

Use of either '0010' or '0011' is not recommended in the ATSC television system. Values '0001', '0101' through '0111', and '1100' are reserved.

6.2.4.3 Recommended Receiver Response to AFD

Receiving device designers are strongly encouraged to study Consumer Electronics Association (CEA) bulletin CEB16 [16], which contains recommendations regarding the processing of AFD.

6.2.5 Relationship Between Bar Data and AFD (Informative)

Certain combinations of Active Format Description and bar data may be present in video user data (either, neither, or both). Note that AFD data may not always exactly match bar data because AFD only deals with 4:3, 14:9, and 16:9 aspect ratios while bar data can represent nearly any aspect ratio. When AFD and bar data are present together, AFD should be used in preference to bar data, except in the cases of AFD '0000' and '0100', where bar data should be used in concert with AFD as described above.

End of document