MTS4EAV7
HEVC / AVC Video and Compressed Audio Analyzer

User Manual

071-3228-00

Tektronix

MTS4EAV7
HEVC / AVC Video and Compressed Audio Analyzer

User Manual

Register now!
Click the following link to protect your product.
► **www.tektronix.com/register**
This document supports software version 7.0 and above.

www.tektronix.com
071-3228-00

Tektronix

Copyright © Tektronix. All rights reserved. Licensed software products are owned by Tektronix or its subsidiaries or suppliers, and are protected by national copyright laws and international treaty provisions.

Tektronix products are covered by U.S. and foreign patents, issued and pending. Information in this publication supersedes that in all previously published material. Specifications and price change privileges reserved.

TEKTRONIX and TEK are registered trademarks of Tektronix, Inc.

"Dolby", "Pro Logic", and the double-D symbol are trademarks of Dolby Laboratories.

Contacting Tektronix

Tektronix, Inc.
14150 SW Karl Braun Drive
P.O. Box 500
Beaverton, OR 97077
USA

For product information, sales, service, and technical support:

- In North America, call 1-800-833-9200.
- Worldwide, visit www.tektronix.com to find contacts in your area.

Warranty

Tektronix warrants that the media on which this software product is furnished and the encoding of the programs on the media will be free from defects in materials and workmanship for a period of three (3) months from the date of shipment. If any such medium or encoding proves defective during the warranty period, Tektronix will provide a replacement in exchange for the defective medium. Except as to the media on which this software product is furnished, this software product is provided “as is” without warranty of any kind, either express or implied. Tektronix does not warrant that the functions contained in this software product will meet Customer's requirements or that the operation of the programs will be uninterrupted or error-free.

In order to obtain service under this warranty, Customer must notify Tektronix of the defect before the expiration of the warranty period. If Tektronix is unable to provide a replacement that is free from defects in materials and workmanship within a reasonable time thereafter, Customer may terminate the license for this software product and return this software product and any associated materials for credit or refund.

THIS WARRANTY IS GIVEN BY TEKTRONIX WITH RESPECT TO THE PRODUCT IN LIEU OF ANY OTHER WARRANTIES, EXPRESS OR IMPLIED. TEKTRONIX AND ITS VENDORS DISCLAIM ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. TEKTRONIX' RESPONSIBILITY TO REPLACE DEFECTIVE MEDIA OR REFUND CUSTOMER'S PAYMENT IS THE SOLE AND EXCLUSIVE REMEDY PROVIDED TO THE CUSTOMER FOR BREACH OF THIS WARRANTY. TEKTRONIX AND ITS VENDORS WILL NOT BE LIABLE FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES IRRESPECTIVE OF WHETHER TEKTRONIX OR THE VENDOR HAS ADVANCE NOTICE OF THE POSSIBILITY OF SUCH DAMAGES.

[W9b – 15AUG04]

IMPORTANT: READ BEFORE OPERATING EQUIPMENT

This software is provided under license from Tektronix, Inc. Retention of this program for more than thirty (30) days or use of the program in any manner constitutes acceptance of the license terms.

CAREFULLY READ THE ENCLOSED SOFTWARE LICENSE AGREEMENT. If you cannot agree to the license terms, promptly contact the nearest Tektronix Field Office for return assistance.

TEKTRONIX SOFTWARE LICENSE AGREEMENT

THE PROGRAM, OR PROGRAMS, ENCODED OR INCORPORATED WITHIN EQUIPMENT, IS FURNISHED SUBJECT TO THE TERMS AND CONDITIONS OF THIS AGREEMENT. RETENTION OF THE PROGRAM FOR MORE THAN THIRTY DAYS OR USE OF THE PROGRAM IN ANY MANNER WILL BE CONSIDERED ACCEPTANCE OF THE AGREEMENT TERMS. IF THESE TERMS ARE NOT ACCEPTABLE, THE UNUSED PROGRAM AND ANY ACCOMPANYING DOCUMENTATION SHOULD BE RETURNED PROMPTLY TO TEKTRONIX FOR A FULL REFUND OF THE LICENSE FEE PAID. (FOR INFORMATION REGARDING THE RETURN OF PROGRAMS ENCODED OR INCORPORATED WITHIN EQUIPMENT, CONTACT THE NEAREST TEKTRONIX SALES OFFICE.)

DEFINITIONS. "Tektronix" means Tektronix, Inc., an Oregon corporation, or local Tektronix' legal entity that is supplying the equipment.

"Program" means the Tektronix software product (executable program and/or data) enclosed with this Agreement or included within the equipment with which this Agreement is packed.

"Customer" means the person or organization in whose name the Program was ordered.

LICENSE. Customer may:

1. Use the Program on a single machine at any one time;
2. If the Program is provided in connection with a floating-user license, the Program may be used on multiple machines provided that the user is authorized, and the total number of users at any one time does not exceed the total number of licensed concurrent users;
3. Modify the Program or merge it with another for use on the single machine; and
4. Copy the Program for archival or backup purposes, provided that no more than one (1) such copy is permitted to exist at any one time. If the Program is provided in connection with a floating-user license, the Program may be copied onto multiple machines for use by authorized users.

Each copy of the Program made by Customer must include a reproduction of any copyright notice or restrictive rights legend appearing in or on the copy of the Program as received from Tektronix.

Customer may not:

1. Use the Program on more than one machine at any one time, unless covered by a floating-user license or separate site license;
2. Transfer the Program to any person or organization outside of Customer or the corporation of which Customer is a part without the prior written consent of Tektronix, except in connection with the transfer of the equipment within which the programs are encoded or incorporated;

3. Export or re-export, directly or indirectly, the program, any associated documentation, or the direct product thereof, to any country to which such export or re-export is restricted by law or regulation of the United States or any foreign government having jurisdiction without the prior authorization, if required, of the Office of Export Administration, Department of Commerce, Washington, D.C. and the corresponding agency of such foreign government;

4. For object-code Programs only, reverse compile or disassemble the Program for any purpose; or

5. Copy the documentation accompanying the Program.

For Programs designed to reside on a single-machine and support one or more additional machines, either locally or remotely, without permitting the Program to be transferred to an additional machine for local execution, the additional machines shall be considered within the definition of "single machine". For programs permitting the Program to be transferred to an additional machine for local execution, a separate license shall be required for each such machine with which the Program may be used, or each concurrent user authorized under a floating-user license.

Title to the Program and all copies thereof, but not the media on which the Program or copies may reside, shall be and remain with Tektronix or others for whom Tektronix has obtained a respective licensing right.

Customer shall pay when due all property taxes that may now or hereafter be imposed, levied or assessed with respect to the possession or use of the Program or this license and shall file all reports required in connection with such taxes.

Any portion of the Program modified by Customer or merged with another program shall remain subject to these terms and conditions.

If the Program is acquired by or for an agency of the U.S. Government, the Program shall be considered computer software developed at private expense and the license granted herein shall be interpreted as granting Customer restricted rights in the Program and related documentation as defined in the applicable acquisition regulation.

THE PROGRAM MAY NOT BE USED, COPIED, MODIFIED, MERGED, OR TRANSFERRED TO ANOTHER EXCEPT AS EXPRESSLY PERMITTED BY THESE TERMS AND CONDITIONS.

UPON TRANSFER OF ANY COPY, MODIFICATION, OR MERGED PORTION OF THE PROGRAM, THE LICENSE GRANTED HEREIN IS AUTOMATICALLY TERMINATED.

TERM. The license granted herein is effective upon acceptance by Customer, and shall remain in effect until terminated as provided herein. The license may be terminated by Customer at any time upon written notice to Tektronix. The license may be terminated by Tektronix or any third party from whom Tektronix may have obtained a respective licensing right if Customer fails to comply with any term or condition and such failure is not remedied within thirty (30) days after notice hereof from Tektronix or such third party. Upon termination by either party, Customer shall return to Tektronix or destroy, the Program and all associated documentation, together with all copies in any form.

LIMITED WARRANTY. Tektronix warrants that the media on which the Program is furnished and the encoding of the Program on the media will be free from defects in materials and workmanship for a period of three (3) months from the date of shipment. If any such medium or encoding proves defective during the warranty period, Tektronix will provide a replacement in exchange for the defective medium. Except as to the media on which the Program is furnished, the Program is provided "as is" without warranty of any kind, either express or implied. Tektronix does not warrant that the functions contained in the Program will meet Customer's requirements or that the operation of the Program will be uninterrupted or error-free.

In order to obtain service under this warranty, Customer must notify Tektronix of the defect before the expiration of the warranty period. If Tektronix is unable to provide a replacement that is free from defects in materials and workmanship within a reasonable time thereafter, Customer may terminate the license for the Program and return the Program and any associated materials for credit or refund.

THIS WARRANTY IS GIVEN BY TEKTRONIX WITH RESPECT TO THE PROGRAM IN LIEU OF ANY OTHER WARRANTIES, EXPRESS OR IMPLIED. TEKTRONIX AND ITS VENDORS DISCLAIM ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. TEKTRONIX' RESPONSIBILITY TO REPLACE DEFECTIVE MEDIA, OR REFUND CUSTOMER'S PAYMENT IS THE SOLE AND EXCLUSIVE REMEDY PROVIDED TO THE CUSTOMER FOR BREACH OF THIS WARRANTY.

LIMITATION OF LIABILITY, IN NO EVENT SHALL TEKTRONIX OR OTHERS FROM WHOM TEKTRONIX HAS OBTAINED A LICENSING RIGHT BE LIABLE FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES ARISING OUT OF OR CONNECTED WITH CUSTOMER'S POSSESSION OR USE OF THE PROGRAM, EVEN IF TEKTRONIX OR SUCH OTHERS HAS ADVANCE NOTICE OF THE POSSIBILITY OF SUCH DAMAGES.

THIRD-PARTY DISCLAIMER. Except as expressly agreed otherwise, third parties from whom Tektronix may have obtained a licensing right do not warrant the program, do not assume any liability with respect to its use, and do not undertake to furnish any support or information relating thereto.

GENERAL. This Agreement contains the entire agreement between the parties with respect to the use, reproduction, and transfer of the Program.

Neither this Agreement nor the license granted herein is assignable or transferable by Customer without the prior written consent of Tektronix.

This Agreement and the license granted herein shall be governed by the laws of the state of Oregon.

All questions regarding this Agreement or the license granted herein should be directed to the nearest Tektronix Sales Office.

ADDITIONAL LICENSE GRANT FOR VIDEO TEST SEQUENCES. The Software Product may include certain test patterns, video test sequences and video clips (together “Video Test Sequences”). If so, the following terms describe Your rights to the Video Test Sequences:

You may use, copy and modify the Video Test Sequences and display or distribute copies of individual Video Test Sequences in connection with Your video testing activity.

You are not licensed to do any of the following:

1. You may not distribute the collection of Video Test Sequences, except in connection with the sale of original equipment containing the Video Test Sequences, without prior written permission from Tektronix.
2. You may not distribute the collection of Video Test Sequences, except in connection with the sale of original equipment containing the Video Test Sequences, without prior written permission from Tektronix.
3. You may not sell, license or distribute copies of the Video Test Sequences on a standalone basis or as part of any collection, product, or service where the primary value of the product or service is the Video Test Sequences.

You must indemnify, hold harmless, and defend Tektronix from and against any claims or lawsuits, including attorneys’ fees, that arise from or result from the use or distribution of Video Test Sequences as modified by You.

You must include a valid copyright notice on Your products and services that include copies of the Video Test Sequences.

Table of Contents

Compliance information x
Environmental considerations x
Preface xi
Related material xi
Product naming conventions xi
Manual conventions xii
Number conventions xii
User manual and tutorials xii

Introduction

Overview 1-1
Product models 1-1
Product features 1-2
Applications 1-4

Software, hardware, and user prerequisites

Software, hardware, and user prerequisites 2-1
Windows operating system prerequisites 2-1
Audio playout prerequisites 2-1
Computer prerequisites 2-1
User prerequisites 2-2
MTS4000 MPEG Test Systems 2-2

Installation and licensing

Installation and licensing 3-1
License type 3-1
MTS4EA software installation 3-2
Troubleshooting the license server 3-7

Compression standards and file types

Compression standards and file types 4-1
Supported video compression standards 4-1
Supported audio compression standards 4-14
Permitted video types/formats 4-16
Permitted audio types/format 4-23
Standards references 4-24

How to use the MTS4EA

How to use the MTS4EA 5-1
- Window elements 5-2
- Starting to use the MTS4EA 5-3
- Main menus 5-13

File menu 5-14
- Open stream... Ctrl+O 5-14
- Close stream 5-30
- Save video stream as... 5-30
- New project 5-31
- Open project... 5-31
- Save project... 5-31
- Save project as... 5-31
- Close project 5-32
- Open other 5-32
- Preferences 5-32
- Example files 5-33
- Export A/V delay streams 5-40
- Recent File 5-41
- Exit 5-41

Play menu 5-42
- Audio sounds broken up 5-43
- Audio/video synchronization 5-44
- Play Ctrl+P 5-46
- Fast forward Ctrl+F 5-47
- Blind fast forward Ctrl+B 5-48
- Stop Ctrl+S 5-48
- Pause/step forward Ctrl+A 5-49
- Reverse play Ctrl+Shift+P 5-49
- Fast backward Ctrl+Shift+F 5-49
- Blind fast backward Ctrl+Shift+B 5-50
- Pause/step backward Ctrl+Shift+A 5-50
- Skip forward Ctrl+K 5-51
- Skip backward Ctrl+Shift+K 5-51
- Video start position 5-52
- Continuous play 5-52
- Hold last frame 5-52
- Pause on frame... 5-53
- Video decoder options... 5-54
- Audio decoder options 5-61

Mute audio 5-62
Synchronize audio 5-62
Overlay menu 5-63
Summary & MacroBlock tooltips manipulation 5-64
Video-frame summary tooltip Ctrl+U 5-66
MB tooltip Ctrl+M 5-76
CTU tooltip Ctrl+N 5-84
CU tooltip Ctrl+D 5-86
Image inspector Ctrl+I 5-89
MB types Ctrl+Y 5-91
CU types Ctrl+J 5-95
Partition types Ctrl+L 5-96
Luma TU Ctrl+Q 5-97
Chroma TU Ctrl+Shift+Q 5-98
Slice Ctrl+Shift+S 5-99
Tile Alt+T 5-100
Motion vectors Ctrl+E 5-101
MB statistics 5-105
CTU statistics 5-115
Overlay color Ctrl+W 5-126
Visual difference 5-127
Luma only 5-131
Blank video 5-131
Audio options 5-132
Audio summary tooltip 5-132
Audio frame tooltip 5-134
Show audio waveform 5-136
Show audio frame markers 5-141
Show sync markers 5-141
Show histogram 5-141
Show plot 5-145
Analysis menu 5-146
Video trace enable... Ctrl+T 5-147
Video graph enable... Ctrl+G 5-172
Video fidelity enable... 5-186
Output decoded video file... 5-191
Video analysis options 5-191
View video 5-192
View video navigator 5-195
View video trace Ctrl+Shift+T 5-198
View video graphs 5-203

View video stream hex Ctrl+H 5-204
View video stream structure Ctrl+R 5-210
View video buffer analysis 5-213
View video fidelity analysis 5-225
View bits per video frame analysis 5-228
View AVC to SVC MB percentage view 5-231
View CU size distribution 5-231
View QP variation 5-232
Audio trace enable 5-233
Audio analysis options 5-243
View audio 5-243
View audio trace 5-243
View audio stream hex 5-246
View audio stream structure 5-252
Audio/Video delay 5-254
Enforce extra restrictions 5-256
Scalability level selection 5-257
Edit menu 5-258
Edit audio/video stream 5-258
Saving edited stream 5-261
Edit commands 5-261
Alert menu 5-262
Alert levels 5-262
Explanation of Fatal/Error/Warning/Info display 5-263
Description of alert levels 5-270
Configure alerts 5-272
View Alert log 5-274
Status bar indication of alert status 5-280
Window menu 5-281
Quickly switch windows 5-281
Cascade, Tile, Arrange Icons, Close All functions 5-282
Toolbars 5-284
Status bar 5-284
Video scale 5-285
HexView settings 5-286
Graph view settings 5-287
Synchronize views 5-288
Active views 5-289
Help menu 5-290
User manual (PDF) F1 5-290
Tutorials (PDF) 5-290

License manager 5-290
About MTS4EA 5-290
Toolbars, Status bar, and shortcut keys 5-291
Icon toolbars 5-291
Context-sensitive toolbars/tooltips 5-294
Status bar 5-298
Ctrl shortcut keys 5-299
Alt menu keys 5-300
Command line/batch mode 5-301
Command line mode 5-301
Batch mode 5-301
Running the MTS4EA in command line/batch mode 5-302

Bitstream syntax debugging

Bitstream syntax debugging 6-1
General codes used in Trace files and Alerts 6-1
Bitstream syntax debugging using the MTS4EA 6-9
Procedure for bitstream syntax debugging 6-10

Appendices

Appendix A: Decoder plugins for the MTS4EA A-1
Purpose of the MTS4EA Decoder Plugins A-1
Support of Decoder Plugins A-1
Decoder Plugin file layout on the CD A-2
Format for generating Decoder Plugins A-3
Use of Decoder Plugins A-3
Decoder Plugins provided A-4
Appendix B: Tests of the MTS4EA B-1
Tests with MPEG-4 Normative and Donated bitstreams B-1
Bitstreams: Normative ISO B-2
Bitstreams: Donated \ I-VOP B-3
Bitstreams: Donated \ Overall B-3
Bitstreams: Donated \ Short Header B-4
Bitstreams: Donated \ P-VOP B-5
Bitstreams: Donated \ Error B-7
Appendix C: Third party software licenses C-1
BSD license C-1

Glossary

Index

Index

List of Figures

Figure 5-1: CU tooltip – Intra Prediction Mode 5-86
Figure 5-2: CU tooltip – Inter Prediction Mode 5-86
Figure 5-3: MPEG-4 trace parse example 5-153
Figure 5-4: H.265/HEVC example, at start of byte stream (full MTS4EA Trace view window) 5-154
Figure 5-5: H.264/AVC example, at start of byte stream (full MTS4EA Trace view window) 5-154
Figure 5-6: H.264/SVC example, at start of byte stream (full MTS4EA Trace view window)..... 5-155
Figure 5-7: H.263 example, at start of bitstream..... 5-155
Figure 5-8: MPEG-4 trace interpret example 5-157
Figure 5-9: H.264/AVC example, at start of byte stream (full MTS4EA Trace view window) 5-158
Figure 5-10: H.264/SVC example, at start of byte stream (full MTS4EA Trace view window) 5-158
Figure 5-11: H.263 example at start of bitstream 5-159
Figure 5-12: MPEG-2 example (interlaced example with top and bottom fields)..... 5-161
Figure 5-13: H.263 example 5-162
Figure 5-14: H.265 example 5-162
Figure 5-15: H.264/AVC example..... 5-163
Figure 5-16: MPEG-4 example 5-163
Figure 5-17: H.265 example 5-164
Figure 5-18: Frame fidelity – H.265/HEVC example 5-169
Figure 5-19: H.264 graph..... 5-172
Figure 5-20: H.265 graph..... 5-173
Figure 5-21: Bits per Frame Analysis graph – MPEG-4 example 5-228
Figure 5-22: Bits per Frame Analysis graph – H.265/HEVC example 5-228
Figure 5-23: Bits per Frame Analysis graph – MPEG-2 example of Frame units display 5-229
Figure 5-24: Bits per Frame Analysis graph – MPEG-2 example of Picture units display 5-230
Figure 5-25: MP4 containing AAC LC 5-252
Figure 5-26: Container files, for example, MPEG-2 PES..... 5-252
Figure 5-27: Overall alert level: Infos disabled (all other levels enabled)..... 5-280
Figure 5-28: Overall alert level: Warnings disabled (Errors and Fatals enabled)..... 5-280
Figure 5-29: Overall alert level: Errors disabled 5-280
Figure 5-30: Overall alert level: All disabled - except one/more specified Warning alerts 5-280
Figure 5-31: Overall alert level: All disabled - except one/more specific Error alerts 5-280
Figure 5-32: H.264/AVC example..... 5-297
Figure 5-33: H.264/SVC example..... 5-297
Figure 5-34: MPEG-4 example 5-297
Figure 6-1: MPEG-4 example, at start of bitstream 6-13
Figure 6-2: MPEG-4 example, at start of VOP 6-13
Figure 6-3: MPEG-4 Example, at start of bitstream..... 6-16
Figure 6-4: MPEG-4 example, at start of VOP 6-16
Figure 6-5: H.264/AVC example, at start of byte stream 6-17

Figure 6-6: H.263 example at start of bitstream …… 6-17

List of Tables

Table 4-1: H.264/AVC tools (part 1) 4-3
Table 4-2: H.264/AVC tools (part 2) 4-5
Table 4-3: H.264/SVC tools 4-7
Table 5-1: Icons general to all standards 5-211
Table 5-2: Standard specific icons 5-212
Table 5-3: Bitstream editor icons 5-260
Table B-1: Bitstreams: Normative ISO B-2
Table B-2: Bitstreams: Donated \ I-VOP B-3
Table B-3: Bitstreams: Donated \ Overall B-3
Table B-4: Bitstreams: Donated \ Short Header B-4
Table B-5: Bitstreams: Donated \ P-VOP B-5
Table B-6: Bitstreams: Donated \ Error B-7

Compliance information

Environmental considerations

This section provides information about the environmental impact of the product.

Product end-of-life handling

Observe the following guidelines when recycling an instrument or component:

Equipment recycling. Production of this equipment required the extraction and use of natural resources. The equipment may contain substances that could be harmful to the environment or human health if improperly handled at the product's end of life. To avoid release of such substances into the environment and to reduce the use of natural resources, we encourage you to recycle this product in an appropriate system that will ensure that most of the materials are reused or recycled appropriately.

This symbol indicates that this product complies with the applicable European Union requirements according to Directives 2002/96/EC and 2006/66/EC on waste electrical and electronic equipment (WEEE) and batteries. For information about recycling options, check the Support/Service section of the Tektronix Web site (www.tektronix.com).

Restriction of hazardous substances

This product is classified as an industrial monitoring and control instrument accessory, and is not required to comply with the substance restrictions of the recast RoHS Directive 2011/65/EU until July 22, 2017.

Preface

This manual describes the functions and use of the Tektronix MTS4EAV7 HEVC / AVC Video and Compressed Audio Analyzer. The manual is organized into the following sections:

- Section 1: Introduction
- Section 2: Applications
- Section 3: Software, hardware and user prerequisites
- Section 4: Installation and licensing
- Section 5: Compression standards and file types
- Section 6: How to use the MTS4EA
- Section 7: Bitstream syntax debugging
- Appendix A: Decoder Plugins for the MTS4EA
- Appendix B: Tests of the MTS4EA with MPEG-4 Normative and Donated bitstreams
- Appendix C: Third party software licenses
- Glossary
- Index

Related material

Additional documentation, such as ReadMe files, may be included on the installation CD-ROM.

The following URLs access the Web sites for the standards organizations listed (the URLs listed were valid at the time of writing):

- MPEG-2 standards (International Organization for Standards) www.iso.org/
- DVB standards (European Technical Standards Institute) www.etsi.org/
- ATSC standards (Advanced Television Systems Committee) www.atsc.org/

Product naming conventions

Although the order code for this product is MTS4EAV7, the product is generally referred to as MTS4EA throughout the software and documentation.

Manual conventions

Naming conventions for the interface elements are based on standard Microsoft Windows naming conventions. Naming conventions for MPEG-2, ATSC, and DVB structures follow the conventions derived from the referenced standards documents.

Additionally, mono-spaced text can indicate the following:

- Text you enter from a keyboard Example: Enter the network identity (`http://TSMonitor01`)
- Characters you press on your keyboard Example: Press `CTRL+C` to copy the selected text.
- Paths to components on your hard drive Example: The program files are installed at the following location: `C:\Program Files\Tektronix\`

This manual is applicable to versions 7.0 and above of the MTS4EA software.

Number conventions

Within MTS4EA all numbers are decimal unless they are preceded by 0x, in which case they are hexadecimal.

For example:

- 16 [decimal] = 16 decimal
- 0x16 [hexadecimal] = 22 decimal

User manual and tutorials

The information in this manual is also provided in the MTS4EA Help system (press F1).

The PDF version of this manual can be also accessed from MTS4EA Help menu > User manual (PDF).

The PDF version of this manual is provided:

- Because it is sometimes easier/clearer to navigate through a PDF document than the standard Windows Help system
- So that you can easily print out additional copies of the manual

Tutorials are supplied as a separate PDF file accessible through the Help menu.

The PDF version of this manual and the tutorials are also on the installation CD-ROM.

Introduction

Overview

The MTS4EA HEVC / AVC Video and Compressed Audio Analyzer is a powerful real-time analytical tool for the investigation of compressed video and audio data that has been encoded using the H.265/HEVC, H.264/AVC, H.264/SVC, MPEG-1/2, MPEG-4 (including AAC and HE-AAC), VC-1, AC-3, and H.263 video standards.

The MTS4EA operates as a standalone software application for use on a personal computer (PC) in a Windows environment. It can also be installed on Tektronix MTS4000 Series MPEG Test Systems.

Product models

The MTS4EA is ordered using the following order code:

- MTS4EAV7

Product features

The MTS4EA enables the display, analysis, debug, and optimization of video compression algorithms or equipment by providing the following features:

- Video and audio decode and analysis
- Verification of the stream's compliance with the encoding standard
- Extraction of elementary streams from containers
- Comprehensive stream navigation and tracking to follow all aspects of the decoding process
- Multiple displays and overlays of codec information
- Synchronized video, audio, and data views for instant cross reference
- Wide range of frame and macroblock statistics, syntax traces including bitstream, interpret, alerts, frame, macroblock, transform, pixel level, and fidelity traces
- Buffer analysis with graphical plots including spatial bits/MB, MV histogram, quantization, DCT frequency, MB coded frequency, and intra-coding frequency
- Video differencing and fidelity analysis
- Bitstream editor for making changes, reanalysing the stream then saving
- Exports data for detailed graphical analysis (requires Microsoft Excel®)
- Comprehensive batch mode for automated regression testing with log reports
- YUV decoded video output for baseband video analysis
- Generation and analysis of Trace files up to 2 GB
- Easy navigation using the Zoom feature for the Video View (Mouse scroll buttons and Keyboard shortcuts)
- Video buffer size (up to 1 GB) allows longer rewind and play
- Decoded frame number shown on the status bar along with the display frame number
- Audio compression analysis (option)
- AV delay measurement (option)
- Built in help and tutorials
- For H.264 streams:
  - Handling of AVC streams that have DPB errors
  - Level-specific tests as per Annex A of AVC

Optional H.265/HEVC features

With Option HEVC installed, the MTS4EA enables the display, analysis, debug, and optimization of H.265/HEVC (High Efficiency Video Coding) video compression algorithms or equipment by providing the additional features listed below:

- Analysis and verification of HEVC streams as per HM 12.0
- Analysis of all levels in Main and Main 10 profiles
- Shows Decoded, Predicted and Residual pictures for every HEVC frame
- Tooltips provide comprehensive data about CUs, CTUs, Frames and Streams
- HEVC specific overlays - Coding Tree Units, Coding Units, Prediction Units, Transform Units (Chroma and Luma separately), Slices and Tiles
- Motion Vectors overlays for HEVC streams
- Trace support for HEVC streams
- HEVC support in existing graphs and statistical views – Bits per frame, Spatial bits, MV Histogram, CTU coded frequency, Average bits, quants, etc.
- HEVC specific graphs – CU Size distribution and QP Variation
- HEVC Picture Level and sub-Picture Level Buffer Analysis
- HEVC support in the Video Navigator View
- Image Inspector shows the pixel values at the CU Level for HEVC streams (other standards shown the MB Level)
- Hovering the mouse over a CTU displays the CTU, CU, and PU boundaries
- The window position of the HEVC toolbar, HEVC tooltips, and HEVC views such as the CU Size Distribution graph and QP Variation graph are stored in the MTS4EA project files
- Support for HEVC tracks contained in Transport Streams

Applications

The key uses for a next generation compression ES analysis tool are as follows:

- Analyze the performance (qualitative and quantitative) of audio and video compression algorithms, to assist with optimization, comparison, and evaluation.
- Check conformity and correctness of a compressed audio or video bit-stream against the various standards.
- Confirm interoperability between different vendor's audio and video compression algorithms.
- Provide comprehensive statistics that will enable the informed user to optimize audio and video compression algorithms.
- Display analysis results in text or graphical formats.

Equipment manufacturers

- Semiconductor device designers and manufacturers
- Video and audio codec software and hardware developers
- STB, PVR, DVD consumer electronics developers for cable, satellite, terrestrial, and IP applications
- Video conferencing and communications equipment developers
- Mobile video and audio infrastructure and handset developers

Video content transmission and distribution

- CODEC and equipment evaluation and comparison in cable, satellite, terrestrial, and IP applications
- Network operators
- Network equipment providers
- Application and service providers
- Streaming media applications
- Broadcasters checking for introduction of AV delay

Software, hardware, and user prerequisites

Software, hardware, and user prerequisites

This section lists the various software, hardware, and user prerequisites for operating the MTS4EA software.

Windows operating system prerequisites

The MTS4EA software has been tested to run on the Microsoft Windows 7 operating system.

The MTS4EA software is not supported under any other operating system.

Audio playout prerequisites

In order to extract and play audio streams in the MTS4EA, the PC requires an audio device, correctly configured.

Computer prerequisites

The MTS4EA software will work satisfactorily on relatively modest PC systems (for example 800 MHz).

To decode and display CIF (Common Intermediate Format) video streams in real time the following minimum PC configuration is recommended:

- Windows 7, 64-bit operating system
- Processor speed >2.5 GHz
- Memory at least 4 GB RAM
- Screen resolution at least 1024x768
- 250 GB minimum hard disk space

When writing trace information to the hard disk, the limiting factor is the speed of the hard disk (regardless of the video standard), so the faster the better. Also, the video and Trace files can be large (hundreds of MB) so a large hard disk is needed:

- Hard disk speed: at least - 7200 rpm
- Hard disk size: as required for high-speed access to the video files to be analyzed plus Trace information

Trace file writing/RAM disk

The real limitation on speed is when the trace information is being written to disk. This is the slowest operation; roughly 2 MB to 4 MB of data are written per video frame for the more extensive trace functions. For H.265/HEVC video, if all of the options are selected, 800 MB to 1 GB of data are written per video frame.

For most customers, this is not a problem, but if you want to have the highest speed when playing video and writing the trace files, a RAM disk can be used. This uses some of the system memory to emulate a disk drive.

There are several software products that offer this functionality; search on the Internet for ‘RAM disk’.

If a RAM disk is used, a memory configuration of at least 512 MB is recommended, with approximately 200 MB of RAM reserved for the RAM disk. This would allow a 200 MB Trace file to be saved on the RAM disk.

User prerequisites

The MTS4EA is sophisticated analysis software, which presents detailed information relating to the video and audio standards.

It therefore assumes that you understand these audio and video standards in significant detail and can therefore interpret the information presented by the MTS4EA.

A list of references are provided. (See page 4-1, *Compression standards and file types*.)

MTS4000 MPEG Test Systems

You can install version 7.0 and above of the MTS4EA software on Tektronix MTS4000 Series MPEG test systems.

Installation and licensing

Installation and licensing

This chapter describes the MTS4EA license type and the installation of the MTS4EA software and license server.

License type

The MTS4EA software is provided with a floating license, which uses a software key located on a license server or local machine to provide the license key.

This means that in order to run the MTS4EA software, you must have the MTS4EA software key installed on the computer or server running the software.

NOTE. *The software key is very important as it contains the MTS4EA license and therefore embodies the value of the MTS4EA product. If the software key is lost, you will have to purchase another software key, which may be at full list price of the software.*

All product options are visible in the installed MTS4EA menus. Options that are not enabled by the license will be unavailable (grayed out). The status and scope of the license can be checked at any time by clicking on the License manager option in the Help menu.

License server requirements

The floating license uses a software key located on a license server or local machine to provide the license key.

Note the following system requirements:

- The software key will require a USB connection on the remote server or local machine.
- If installed on a remote server, a license server process must be run on the server machine (this process has modest requirements for CPU load).
- If installed on a remote server, clients require TCP/IP access to the server (this could be provided via VPN between remote sites).

MTS4EA software installation

The MTS4EA software installer provides two setup choices: Local machine and Remote server. If you are going to install the MTS4EA software on a remote server, the server that is to host the license software must be set up first.

The MTS4EA floating license system enables a single software key (the DES DK2 dongle) to serve up licenses to multiple users connected over a network. The software key is pre-programmed with licenses before shipment.

To install the floating license system on the network, you must:

- Install the DK2 Network Server software on a server (the license server).
- On every client that is to use the floating license, install the MTS4EA software that also installs the DK2 client software.

If you are going to install the MTS4EA software only on a local machine, proceed to *Install the MTS4EA software*. (See page 3-4.)

Install the license server

A license server hosts the software key, which contains one or more licenses. It is possible to have several license servers on the network, each with their own software key, forming a pool of licenses for the clients to use. It is also relatively easy to have a machine acting as both a server and a client.

To allow a machine to act as a license server, the following procedure should be repeated on each machine:

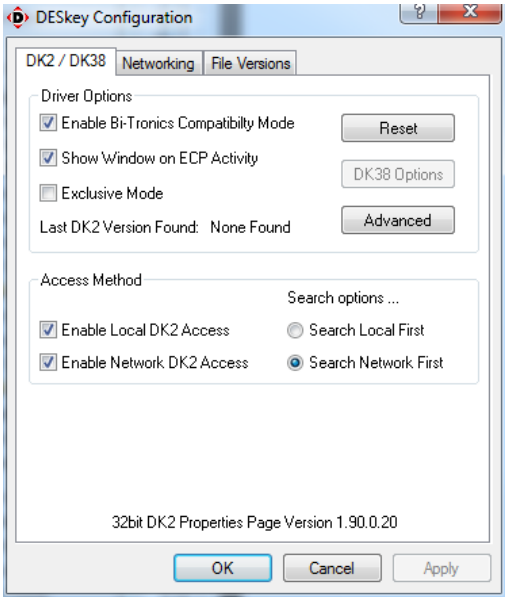
1. Select a server machine and connect the software key (either via the USB port or parallel port depending on software key type).
2. On the server machine, insert the MTS4EA installation CD.

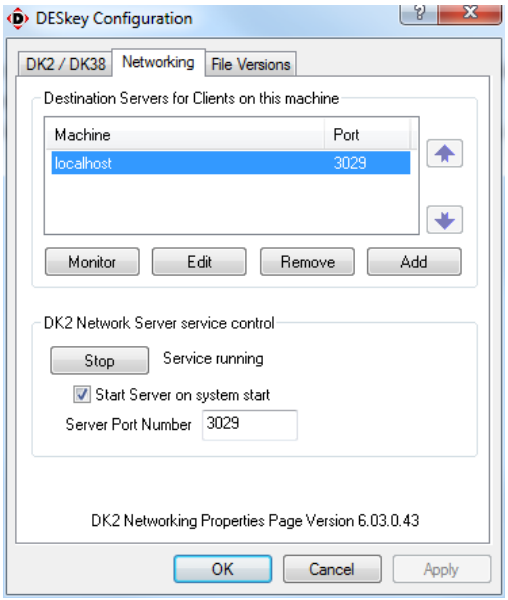
NOTE. *When the CD is inserted, the MTS4EA setup screen will be displayed. Select the Close button at the bottom of the window to terminate the MTS4EA installation program.*

3. Using Windows Explorer, locate and open the DES directory. Run the program named `dnsrv32.exe`. During the installation process, accept the default entries.
4. Reboot the server machine if requested to do so at the end of the installation process.

5. Open the Control Panel (Windows Start > Settings > Control Panel) and select the DESkey icon. This will show the DESkey Configuration dialog, the first two tabs of which are shown in the following illustrations (note the version numbers shown near the bottom may vary).

NOTE. *For the server installation, the DK2/DK38 tab and the top half of the Networking tab can be ignored; these refer to the client installation that shares the same configuration dialog.*





6. Select the Networking tab. The field at the bottom labeled Server Port Number shows the port to be used for communication with any clients; the default is 3029. Press the OK button to finish.

Depending on the policy of your IT department, port 3029 may be blocked by the network firewall, in which case you will need to consult your network administrator for another unrestricted port. Enter the new port and click the Apply button. You may then get a prompt asking about restarting the server; select Yes – this will stop and start the network server service using the new port. When the port has been applied, press the OK button to finish. This does not cause the server to reboot; it only restarts the license server.

Install the MTS4EA software

If the MTS4EA is to be installed on a remote server, ensure that the following information is available:

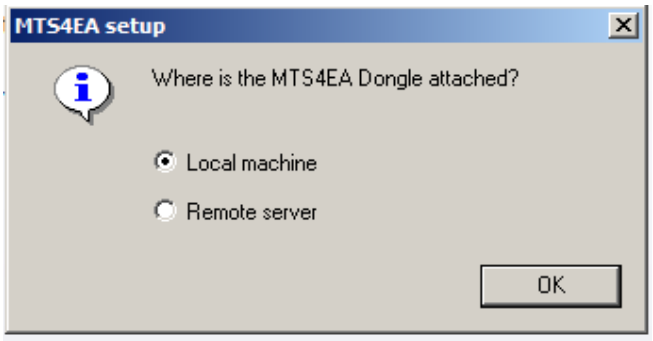
- The network identity of the floating license server
- The server port number

These parameters are set up in the *Install the license server* section. (See page 3-2.)

7. Insert the MTS4EA installation CD.

NOTE. *If the installation process does not start automatically, run setup.exe from the root directory of the installation CD.*

8. At the bottom of the window showing the readme file, click the Install button to start the installation.
9. Select the desired MTS4EA setup option.

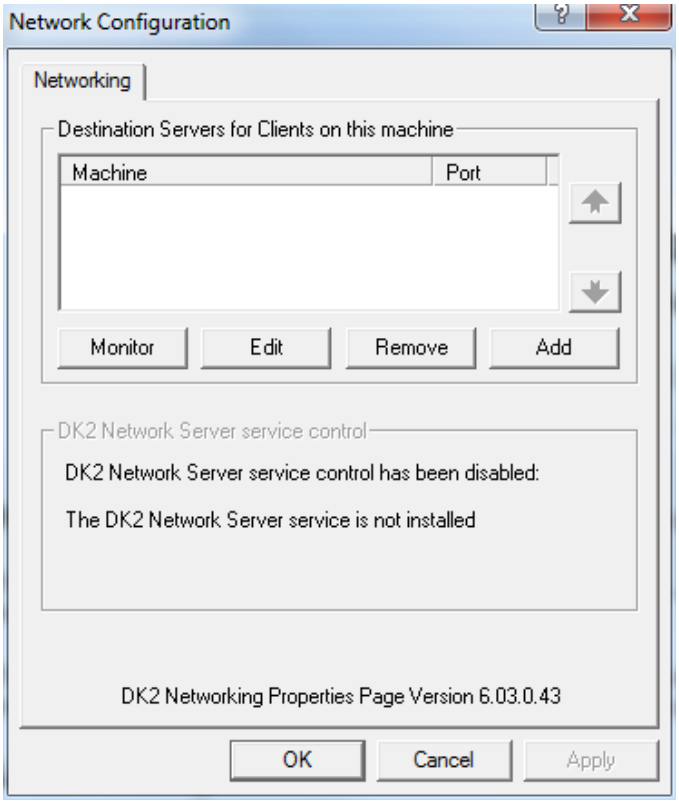


   a. Select **OK** to continue.
   b. If you selected the Local machine option, proceed to step 14.
   c. If you selected the Remote server option, proceed to the next step.

10. An initial dialog box is shown. Press **Next**. The DK2 software key driver is installed.

NOTE. *If you have installed the license server on this machine, launching the client installation may show a dialog box titled DK2 Driver/Server Object that asks if you want to start the DK2 Network Server - attach the software key and select Yes.*

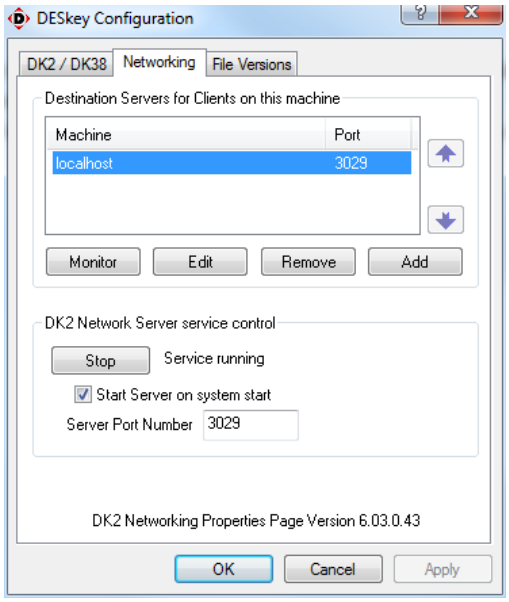
11. At the end of the driver installation setup, a Network Configuration dialog box will appear (shown below).



12. This enables the server(s) installed in the previous section to be added to the list of machines to query for licenses. If the required license server identity is not already listed, select Add and enter the required license server network identity and port number.

NOTE. *If the DK2 installer detects a previous installation of the DK2 driver on this machine, the Network Configuration dialog box will not appear. In this case, allow the installation to proceed and complete. After installation, open the DESkey Networking Configuration dialog box (Windows Start > Settings > Control Panel > DESkey > Networking tab) and proceed as described below.*

If you are installing the MTS4EA client software on the license server, the server name to add is localhost. This is shown in the Networking Tab diagram of the DESkey Configuration dialog box in the License Server section above. The actual name of the machine can be used if preferred.



NOTE. *Changes made to the Server list on the Networking tab will only take effect when the Apply button is pressed.*

13. Add all the servers that are to be accessible to this client, and then press the OK button to close the Network Configuration dialog box.

CAUTION. *At this stage, depending on the history of the machine, a prompt may ask to reboot. Select No, otherwise the MTS4EA installation will not complete correctly and need to be rerun.*

14. The MTS4EA installation will continue. When finished, reboot the machine.

Test the license server

To ensure that the previous installations were successful, perform the following steps:

15. If the MTS4EA software was installed on a remote server, remove any Tektronix DK2 software keys from the client machine and ensure the server machine has a suitably programmed software key connected.

16. Run the MTS4EA application using the desktop icon. If it starts, the license server is functioning correctly. If it does not start or displays the error message “DESkey dongle failure. Please see the manual for troubleshooting advice,” then check the following:

On the server machine, ensure the software key is inserted correctly and the DK2 Network Server service is running. The latter can be checked by opening the DESkey Configuration dialog box from the Control Panel. The Networking tab shows the status of the service - if it is not running, then start it and try the MTS4EA application again.

Troubleshooting the license server

If the service will not start

Reboot the machine and try again. If it still does not work, it may be a faulty or disabled port. For a USB software key, check the Device Manager for a DESkey entry. (Open the Windows Control Panel, select the System icon, on the Hardware tab click the Device Manager button. Any problem will show up as a yellow triangle next to the DESkey entry.) For a parallel port software key, check that the port is enabled in the machine's BIOS settings. Try a different port connection if more than one is available.

It could also be a faulty software key, try connecting the software key to the client machine and run the MT4EA software again (ensure the Enable Local DK2 Access checkbox is selected in the DK2/DK38 tab of the DESkey configuration dialog box). If this works, the software key is working, so try another server machine. If this does not work, contact Tektronix Customer Support.

If the service starts but the MTS4EA will not run

Check with the network administrator that the port being used is not being blocked by a firewall. If it is either get it unblocked or choose another port, changing it in both the server and all clients; remember to press Apply in the associated dialog boxes before closing them.

On the client machine, check the machine name entered in the client server list is present and correct. If you entered it manually, try navigating to it using the Browse feature to confirm that it is correct.

Check that the port number specified by the server is the same as the port number specified in the client’s server list.

On the client machine, ensure the Enable Network DK2 Access checkbox is ticked on the DK2/DK38 tab of the DESkey Configuration dialog.

Compression standards and file types

Compression standards and file types

This section contains the following information:

- A description of the standards (and elements within those standards), that MTS4EA supports, both video and audio
- The file types MTS4EA can work with, both video and audio
- The exact titles and numbers of the standards on which the MTS4EA is based

Supported video compression standards

The video compression standards supported in this version of the MTS4EA are:

- H.265/HEVC
- H.264/AVC
- H.264/SVC
- VC-1
- MPEG-4 Part 2
- MPEG-2 Part 2
- H.263

The MTS4EA will open container files (containing the above types of video):

- MPEG-2 Transport Stream (H.265/HEVC, H.264/AVC, H.264/SVC, VC-1 Advanced profile, MPEG-2 Program Stream, MPEG-2 Packetized Elementary Stream, MPEG-2 Elementary Stream only)
- MP4 (MPEG-4 Part 2, H.263 Baseline only and H.264)
- 3GPP (MPEG-4 Part 2, H.263 Baseline only and H.264)
- VOB/MPEG-2 Program Stream (MPEG-2 only)
- Microsoft ASF files
- RCV files

The MTS4EA will also open uncompressed video files: (See page 5-25, *Opening an uncompressed video file (any file extension)*.)

The MTS4EA will also open audio files of the standards given in *Supported Audio Compression Standards*, and container files that have these types of audio within them. (See page 4-14, *Supported audio compression standards*.)

NOTE. *Not all container files support all types of audio listed under Supported Audio Compression Standards. (See page 4-14, Supported audio compression standards.) and (See page 4-23, Permitted audio types/format.)*

H.265/HEVC

This version of the MTS4EA supports the following elements of the H.265/HEVC standard (see Reference [27] and Reference [28] under *Standards references*).

H.265/HEVC Profiles, Levels. The following Profiles and Levels are supported:

Main Profile

- All Levels from 1 to 6.2 inclusive

Main 10 Profile

- All Levels from 1 to 6.2 inclusive

H.264/AVC

This version of MTS4EA supports the following elements of the H.264/AVC standard (see Reference [12] under *Standards references*).

H.264/AVC Profiles, Levels. The following Profiles and Levels are supported:

Baseline Profile

- All Levels from 1 to 5.1 inclusive

Extended Profile

- All Levels from 1 to 5.1 inclusive

Main Profile

- All Levels from 1 to 5.1 inclusive

High Profile

- All Levels from 1 to 5.1 inclusive

High 10-bit Profile

- All Levels from 1 to 5.1 inclusive

High 4:2:2 Profile

- All Levels from 1 to 5.1 inclusive

High 4:4:4 Predictive Profile

- All Levels from 1 to 5.1 inclusive

High 10 Intra Profile

- All Levels from 1 to 5.1 inclusive

High 4:2:2 Intra Profile

- All Levels from 1 to 5.1 inclusive

High 4:4:4 Intra profile

- All Levels from 1 to 5.1 inclusive

CAVLC 4:4:4 Intra profile

- All Levels from 1 to 5.1 inclusive

H.264/AVC tools. (For a description of these tools, see Reference [12] under *Standards references*)

All tools are supported, as permitted in the H.264/AVC standard, listed below.

Table 4-1: H.264/AVC tools (part 1)

Tool [1]	Baseline	Extended	Main	High	High 10	High 4:2:2
Profile and level indications:						
profile_idc	66	88	77	100	110	122
constraint_set0_flag	1	0 or 1	0 or 1	0 or 1	0 or 1	0 or 1
constraint_set1_flag	0 or 1	0 or 1	1	0 or 1	1	1
constraint_set2_flag	0 or 1	1	0 or 1	0 or 1	0 or 1	0 or 1
constraint_set3_flag	0 or 1	0 or 1	0 or 1	0	0	0
level_idc	10 - 51	10 - 51	10 - 51	10 - 51	10 - 51	10 - 51
Slice types:						
I	Yes	Yes	Yes	Yes	Yes	Yes
P	Yes	Yes	Yes	Yes	Yes	Yes
B	No	Yes	Yes	Yes	Yes	Yes
SP	No	Yes	No	No	No	No
SI	No	Yes	No	No	No	No
NAL unit types:						
1 - coded slice of a non-IDR picture	No	Yes	No	No	No	No
2 - coded slice data partition A	No	Yes	No	No	No	No
3 - coded slice data partition B	No	Yes	No	No	No	No

Tool [1]	Baseline	Extended	Main	High	High 10	High 4:2:2
4 - coded slice data partition C	Yes	Yes	Yes	Yes	Yes	Yes
5 - coded slice of an IDR picture	Yes	Yes	Yes	Yes	Yes	Yes
6 - supplemental enhancement information	Yes	Yes	Yes	Yes	Yes	Yes
7 - sequence parameter set	Yes	Yes	Yes	Yes	Yes	Yes
8 - picture parameter set	Yes	Yes	Yes	Yes	Yes	Yes
9 - access unit delimiter	Yes	Yes	Yes	Yes	Yes	Yes
10 - end of sequence	Yes	Yes	Yes	Yes	Yes	Yes
12 - end of stream	Yes	Yes	Yes	Yes	Yes	Yes
13 - filler data	Yes	Yes	Yes	Yes	Yes	Yes
Structural:						
data partitioning (NALU type 2-4)	No	Yes	No	No	No	No
interlace	No	Yes (Except L1-2 & 4.2 - 5.1)	Yes (Except L1-2 & 4.2 - 5.1)	Yes (Except L1-2 & 4.2 - 5.1)	Yes (Except L1-2 & 4.2 - 5.1)	Yes (Except L1-2 & 4.2 - 5.1)
arbitrary slice order	Yes	Yes	No	No	No	No
slice groups (num_slice_groups_minus1)	1 - 8	1 - 8	1	1	1	1
redundant coded pictures (redundant_pic_cnt_present_flag)	Yes	Yes	No	No	No	No
Weighted prediction:						
weighted_pred_flag	0	=> 0	=> 0	=> 0	=> 0	=> 0
weighted_pred_idc	0	=> 0	=> 0	=> 0	=> 0	=> 0
Entropy coding (entropy_coding_mode_flag):						
CAVLC	Yes	Yes	Yes	Yes	Yes	Yes
CABAC	No	No	Yes	Yes	Yes	Yes
B frames:						
direct_8x8_inference_flag	n/a	1	0 (L1-2.2) 1 (L3-5.1)	1 (L3–5.1)	1 (L3–5.1)	1 (L3–5.1)
MinLumaBiPredSize	n/a	8x8 L3.1 - 5.1	8x8 L3.1 - 5.1	8x8 L3.1 - 5.1	8x8 L3.1 - 5.1	8x8 L3.1 - 5.1
Fidelity range:						
chroma format (chroma_format_idc)	0	0	0	0 - 1	0 - 1	0 - 1
bit depth (bit_depth_luma_minus8, bit_depth_chroma_minus8)	0	0	0	0	0 - 2	0 - 2
transform bypass (qpprime_y_zero_transform_bypass_flag)	n/a	n/a	n/a	0	0	0

1 No = not allowed; n/a = not applicable

Table 4-2: H.264/AVC tools (part 2)

Tool [1]	High 4:4:4 predictive	High 10 Intra	High 4:2:2 Intra	High 4:4:4 Intra	CAVLC 4:4:4 Intra
Profile and level indications:					
profile_idc	244	110	122	244	44
constraint_set0_flag	0 or 1	0 or 1	0 or 1	0 or 1	0 or 1
constraint_set1_flag	1	1	1	1	1
constraint_set2_flag	0 or 1	0 or 1	0 or 1	0 or 1	0 or 1
constraint_set3_flag	0	1	1	1	1
level_idc	10 - 51	10 - 51	10 - 51	10 - 51	10 - 51
Slice types:					
I	Yes	Yes	Yes	Yes	Yes
P	Yes	No	No	No	No
B	Yes	No	No	No	No
SP	No	No	No	No	No
SI	No	No	No	No	No
NAL unit types:					
1 - coded slice of a non-IDR picture	No	No	No	No	No
2 - coded slice data partition A	No	No	No	No	No
3 - coded slice data partition B	No	No	No	No	No
4 - coded slice data partition C	Yes	Yes	Yes	Yes	Yes
5 - coded slice of an IDR picture	Yes	Yes	Yes	Yes	Yes
6 - supplemental enhancement information	Yes	Yes	Yes	Yes	Yes
7 - sequence parameter set	Yes	Yes	Yes	Yes	Yes
8 - picture parameter set	Yes	Yes	Yes	Yes	Yes
9 - access unit delimiter	Yes	Yes	Yes	Yes	Yes
10 - end of sequence	Yes	Yes	Yes	Yes	Yes
12 - end of stream	Yes	Yes	Yes	Yes	Yes
13 - filler data	Yes	Yes	Yes	Yes	Yes
Structural:					
data partitioning (NALU type 2-4)	No	No	No	No	No
interlace	Yes (Except L1-2 & 4.2 - 5.1)	Yes (Except L1-2 & 4.2 - 5.1)	Yes (Except L1-2 & 4.2 - 5.1)	Yes (Except L1-2 & 4.2 - 5.1)	Yes (Except L1-2 & 4.2 - 5.1)
arbitrary slice order	No	No	No	No	No
slice groups (num_slice_groups_minus1)	1	1	1	1	1
redundant coded pictures (redundant_pic_cnt_present_flag)	No	No	No	No	No

Tool [1]	High 4:4:4 predictive	High 10 Intra	High 4:2:2 Intra	High 4:4:4 Intra	CAVLC 4:4:4 Intra
Weighted prediction:					
weighted_pred_flag	=> 0	n/a	n/a	n/a	n/a
weighted_pred_idc	=> 0				
Entropy coding (entropy_coding_mode_flag):					
CAVLC	Yes	Yes	Yes	Yes	Yes
CABAC	Yes	Yes	Yes	Yes	No
B frames:					
direct_8x8_inference_flag	1 (L3–5.1)	n/a	n/a	n/a	n/a
MinLumaBiPredSize	8x8 (L3.1 - 5.1)	n/a	n/a	n/a	n/a
Fidelity range:					
chroma format (chroma_format_idc)	0 - 3	0 - 1	0 - 1	0 - 3	0 - 3
bit depth (bit_depth_luma_minus8, bit_depth_chroma_minus8)	0 - 4	0 - 2	0 - 2	0 - 4	0 - 4
transform bypass (qpprime_y_zero_transform_bypass_flag)	0 or 1	0	0	0 or 1	0 or 1

1 No = not allowed; n/a = not applicable

H.264/SVC

This version of MTS4EA supports the following elements of the H.264/SVC standard (see Reference [26] under *Standards references*).

H.264/SVC Profiles, Levels. The following Profiles and Levels are supported:

Scalable Baseline Profile

- All Levels from 1 to 5.1 inclusive

Scalable High Profile

- All Levels from 1 to 5.1 inclusive

Scalable High Intra Profile

- All Levels from 1 to 5.1 inclusive

H.264/SVC tools. (For a description of these tools, see Reference [26] under *Standards references*)

Table 4-3: H.264/SVC tools

Tool [1]	Scalable Baseline	Scalable High	Scalable High Intra
Profile and level indications:			
profile_idc	83	86	86
constraint_set0_flag	1 for base layer, 0 or 1 for enhancement layers	0 or 1	0 or 1
constraint_set1_flag	1 for base layer, 0 or 1 for enhancement layers	0 or 1	0 or 1
constraint_set2_flag	1 for base layer, 0 or 1 for enhancement layers	0 or 1	0 or 1
constraint_set3_flag	0	0	1
Slice types:			
I	Yes	Yes	Yes
P	Yes	Yes	No
B	No	Yes	No
EI	Yes	Yes	Yes
EP	Yes	Yes	No
EB	Yes	Yes	No
SP	No	No	No
SI	No	No	No
NAL unit types:			
1 - coded slice of a non-IDR picture	Yes	Yes	No
2 - coded slice data partition A	No	No	No
3 - coded slice data partition B	No	No	No
4 - coded slice data partition C	No	No	No
5 - coded slice of an IDR picture	Yes	Yes	Yes

Tool [1]	Scalable Baseline	Scalable High	Scalable High Intra
6 - supplemental enhancement information	Yes	Yes	Yes
7 - sequence parameter set	Yes	Yes	Yes
8 - picture parameter set	Yes	Yes	Yes
9 - access unit delimiter	Yes	Yes	Yes
10 - end of sequence	Yes	Yes	Yes
11 - end of stream	Yes	Yes	Yes
12 - filler data	Yes	Yes	Yes
13 - sequence parameter extension	Yes	Yes	Yes
14 - prefix NAL unit	Yes	Yes	Yes
15 - subset sequence parameter set	Yes	Yes	Yes
19 - coded slice of an auxiliary coded picture without partitioning	Yes	Yes	Yes
20 - coded slice extension	Yes	Yes	Yes
Structural:			
data partitioning (NALU type 2-4)	No	No	No
interlace	No	Yes (Except L1-2 & 4.2 - 5.1)	Yes (Except L1-2 & 4.2 - 5.1)
arbitrary slice order	Base layer - No Enhancement layer - Yes	No	No
slice groups (num_slice_groups_minus1 + 1)	Base layer - 1 Enhancement layer - 1 to 8	1	1
redundant coded pictures (redundant_pic_cnt_present_flag)	Base layer - No Enhancement layer - Yes	No	No
Entropy coding (entropy_coding_mode_flag):			
CAVLC	Yes	Yes	Yes
CABAC	Base layer - No Enhancement layer - Yes	Yes	Yes
B frames:			
direct_8x8_inference_flag	Base layer - n/a Enhancement layer - 1	1 (L3 - 5.1)	1 (L3 - 5.1)
MinLumaBiPredSize	n/a	8x8 L3.1 - 5.1	8x8 L3.1 - 5.1
Fidelity range:			
chroma format (chroma_format_idc)	0	0	0
bit depth (bit_depth_luma_minus8, bit_depth_chroma_minus8) + 8	8	8	8
transform bypass (qpprime_y_zero_transform_bypass_flag)	0	0	0

[1] No = not allowed; n/a = not applicable

VC-1

This version of MTS4EA supports the following elements of the VC-1 standard. (see Reference [16] under *Standards references*).

VC-1 Profiles, Levels. The following Profiles and Levels are supported (this is all Profiles and all Levels):

- Simple Profile:
  - Low
  - Medium
- Main Profile:
  - Low
  - Medium
  - High
- Advanced Profile:
  - L0
  - L1
  - L2
  - L3
  - L4

MPEG-4

This version of MTS4EA supports the following elements of the MPEG-4 standard (see Reference [1] under *Standards references*).

NOTE. *The MTS4EA support for MPEG-4 has been updated to the 2004 version of the MPEG-4 standard, which has several differences from the previous version.*

MPEG-4 Profiles, Levels. The following Profiles and Levels are supported:

- Simple Profile:
  - Level 0 (see note below)
  - Level 1
  - Level 2
  - Level 3
  - Level 4a
  - Level 5
- Advanced Simple Profile:
  - Level 0
  - Level 1
  - Level 2
  - Level 3
  - Level 3b (see Reference [5], *Standards references*.)
  - Level 4
  - Level 5
- Main Profile:
  - Level 2

NOTE. *Level 0, which is not in the MPEG-4 Standard Reference [1], is an addition to Simple Profile. Level 0 is targeted at mobile applications: for example, with a maximum picture size of QCIF and maximum frame rate of 15 frames per second. (See page 4-24, Standards references.)*

MPEG-4 tools. (See the following standard for a description of the MPEG-4 tools: MPEG-4 Part 2 (Visual), Reference [1] Table 9-1.)

For MPEG-4 Simple Profile, Advanced Simple Profile and Main Profile, the following tools are supported:

Tool	Simple profile	Advanced simple profile	Main profile
Basic: ■ I-VOP ■ P-VOP ■ AC/DC prediction ■ 4 Motion Vectors ■ Unrestricted Motion Vectors	Y	Y	Y
Short header	Y	Y	Y
Error resilience: ■ Slice Resynchronization ■ Data Partitioning ■ Reversible VLC	Y	Y	Y
B-VOPs	N/A	Y	N
Method 1/Method 2 quantization	N/A	Y	Y
Global Motion Compensation: ■ Mode 0 ■ Mode 1 ■ Mode 2 ■ Mode 3	N/A	Y	N/A
Quarter Sample Motion Compensation (quarter pel)	N/A	Y	N/A
Interlace	N/A	Y (L4, 5 only)	N

1 N/A means that this tool is not allowed in the profile concerned.

NOTE. *MPEG-4 Simple Profile Short Header is decoded by MTS4EA as H.263 Baseline, as defined by the MPEG-4.*

The following additional tools are not supported for Main Profile:

- P-VOP based temporal scalability:
  - Rectangular
  - Arbitrary shape
- Binary shape
- Gray shape
- Sprite

MPEG-2

This version of MTS4EA supports the following elements of the MPEG-2 standard. (see Reference [13] under *Standards references*).

MPEG-2 Profiles, Levels. The following Profiles and Levels are supported, subject to the restrictions. (See page 4-23, *Permitted audio types/format*.)

- Main Profile:
  - Main Level
  - High Level
  - High Level 1440
- 4.2.2 Profile:
  - Main Level
  - High Level

MPEG-2 tools. For MPEG-2 Main Profile at Main Level, at High Level and at High Level 1440, the following tools are supported:

Tool	Main level	High level	High level 1440
I-frames	Y	Y	Y
P-frames	Y	Y	Y
B-frames	Y	Y	Y
Field-coded pictures (Interlaced)	Y	Y	Y
Frame-coded pictures: ■ With field or frame order MacroBlocks	Y	Y	Y
Layers: ■ GOP ■ Picture ■ Slice ■ MacroBlock ■ Block	Y	Y	Y

NOTE. *Higher syntactic structures are also supported within MTS4EA - PES and VOB. (See page 4-18, MPEG-2 higher syntactic structures.)*

H.263 The following H.263 standards are supported:

- H.263 baseline standard (Reference [2] under *Standards references*).

Supported audio compression standards

MTS4EA supports audio extraction and playout. MTS4EA also supports audio stream analysis functionality for all supported standards except Dolby Digital (AC-3). Supported standards are:

- MPEG-1 Part 3 audio

  (References [22] in *Standards references*.)

- MPEG-2 Part 3 audio

  (References [18] in *Standards references*.)

- MPEG-2 Part 7 AAC

  (Reference [19] in *Standards references*.)

- MPEG-4 Part 3 audio

  (Reference [20] in *Standards references*.)

- Dolby Digital (AC-3)

  (Reference [24] in *Standards references*.)

MTS4EA also supports audio files in the following container formats:

- MPEG-4 Part 3 ADTS files

  (Reference [21] in *Standards references*.)

- MPEG-4 Parts 1 and 14

  (References [10] and [23] in *Standards references*.)

- MPEG-2 Transport, Program, PES streams

  (Reference [14] in *Standards references*.)

- 3GPP

  (Reference [11] in *Standards references*.)

MPEG-1 Part 3 audio

MTS4EA supports the MPEG-1 Part 3 audio standard. as defined in Reference [22] in *Standards references*. MTS4EA supports the following layers of this standard:

- Layer I
- Layer II

MTS4EA supports monophonic and stereophonic streams in this standard.

MPEG-2 Part 3 audio

MTS4EA supports the MPEG-2 Part 3 audio standard, as defined in Reference [18], *Standards references*. MTS4EA supports the extension of MPEG-1 part 3 audio to LSF (Low Sampling Frequency) for the following layers:

- Layer I
- Layer II

MTS4EA supports monophonic and stereophonic streams in this standard.

MPEG-2 Part 7 AAC audio

MTS4EA supports the MPEG-2 Part 7 AAC audio standard, as defined in Reference [19], *Standards references*. MTS4EA supports the following profiles of this standard:

- Main
- LC (Low Complexity)
- LTP (Long Term Prediction)

MTS4EA supports monophonic and stereophonic streams in this standard.

MPEG-4 Part 3 audio

MTS4EA supports the MPEG-4 Part 3 audio standard, as defined in Reference [20], *Standards references*. MTS4EA supports the following profiles of this standard:

- Main
- LC (Low Complexity)
- LTP (Long Term Prediction)
- HE-AAC (High Efficiency, ‘AAC plus’), low complexity profile with SBR (Spectral Band Replication)

MTS4EA supports monophonic and stereophonic streams in this standard.

MPEG-4 Part 3 ADTS files

MTS4EA supports the Audio Data Transport Stream (ADTS) for AAC audio, as defined in Reference [21], *Standards references*.

Permitted video types/formats

MTS4EA analyzes and displays many file types, such as:

- Video files containing video Elementary Streams
- Container files, such as ASF files, RCV files, MP4/3GPP files, MPEG-2 Transport Streams, and MPEG-2 Program streams, which include video, audio and other data
- MTS4EA Trace files (which have a.vpt file extension)
- Uncompressed video files, such as YUV, RGB or grayscale color models, 8-16 bit sample depth, various chroma sub-sampling formats
- Any data file, using MTS4EA HexView (hexadecimal file viewer, on the Analysis menu)

The primary analysis functions of MTS4EA are performed on video Elementary Streams, and features are provided to extract these from within container files such as MP4 and 3GPP files.

Microsoft ASF files

ASF files typically contain video, audio and other data. An ASF file can contain more than one video stream. See Reference [17] in *Standards references*.

RCV files

RCV files contain video in the VC-1 format. See Reference [16] in *Standards references*.

MP4 files

MP4 files typically contain video, audio and other data. An MP4 file can contain more than one video stream; these streams can be located consecutively or split up in many areas of the MP4 file.

The MP4 files that the MTS4EA analyzes should conform to the relevant part of the MPEG-4 standard – Reference [10] in *Standards references*.

By default, the MTS4EA looks for MP4 files with an .mp4 file extension, although any file name can be used: MTS4EA will determine that it is an MP4 file by looking through the file contents.

The video streams can be extracted from the MP4 file and analyzed directly, or extracted and analyzed later. (See page 5-14, *Open stream... Ctrl+O*.)

The structure of the MP4 file can be examined, using the **View file structure** option in the Analysis menu. (See page 5-1, *How to use the MTS4EA*.)

3GPP files

3GPP files are a subset of MPEG-4 files, with an indicator in the file to denote that it is a 3GPP file.

3GPP files should conform to the standard concerned – Reference [11] in *Standards references*.

By default, the MTS4EA looks for 3GPP files with a .3gp file extension, although any file name can be used: MTS4EA will determine that it is a 3GPP file by looking through the file contents.

The video streams are extracted from 3GPP files and analyzed directly or extracted and analyzed later in the same way as MP4 files. For information on opening 3GPP files and extracting and saving the video streams. (See page 5-14, *Open stream... Ctrl+O*.)

The structure of the 3GPP file can be examined, using the **View file structure** option in the Analysis menu. (See page 5-1, *How to use the MTS4EA*.)

MPEG-2 Transport Stream files

MPEG-2 Transport Streams typically contain video, audio, and other data. An MPEG-2 Transport Stream can contain more than one video stream - in fact hundreds or more; these streams can be located largely consecutively in the Transport Stream packets or split up in many areas of the Transport Stream.

Transport Streams with 188 bytes per packet, 196 bytes per packet, 204 bytes per packet and 208 bytes per packet can be analyzed.

MPEG-2 higher syntactic structures

MPEG-2 Program Stream and PES files. An MPEG-2 Program Stream contains one or more Packetized Elementary Streams (PES), with a common time base. (The Program Stream is designed for use in a relatively error-free environment.)

The format of the Program Stream is defined in Part 1 of the MPEG-2 standard, Reference [14] in *Standards references*.

A Program Stream can contain a complete Elementary Stream, or it can contain only part of an Elementary Stream.

MTS4EA will open MPEG-2 Program Stream files (typically with a file extension of .mpg) and display the structure of the Program Stream, such as the PES packets inside.

In accordance with the MPEG-2 standard (see Reference [14] in *Standards references*), the Program Stream file will start with a 32-bit start code: 0x000001ba, 0000 0000 0000 0000 0000 0001 1011 1010.

MPEG-2 VOB files (DVD). VOB files are used on a DVD to store the video and audio; a VOB contains a Program Stream.

A single video Elementary Stream (for example, a movie) is usually distributed among several VOB files.

NOTE. *You can specify the different VOBs that contain a single Elementary Stream, so that the MTS4EA can extract the Elementary Stream from the different VOBs.*

MTS4EA will open MPEG-2 VOB files (typically with a file extension of .vob) and display the structure of the Program Stream, such as the PES packets inside.

See Reference [15] in *Standards references*.

NOTE. *When opening a VOB, there is no requirement that the VOB begins with an I-frame. In this situation, the video will not display correctly until an I-frame is reached.*

Encrypted (scrambled) VOBs cannot be opened.

Uncompressed video file format

The general uncompressed video file format is as follows:

- No headers of any kind (no file or frame headers)
- Concatenated planar image data
- Row raster order (top picture row first)
- Unsigned samples

For 8-bit sample depth:

- One byte per sample

For 9-16 bit sample depth:

- Two bytes per sample
- Both little- and big-endian byte orders supported

For YUV format:

- Concatenated Y, U and V planes
- U and V planes sub-sampled as required
- Y plane samples are unsigned
- U and V plane samples are unsigned with a DC offset of 2^{-1}, where *n* is the chroma sample bit depth

For RGB format:

- Concatenated R, G and B planes

For grayscale format:

- Luma plane only

Compressed video file format

The formats of the raw compressed video data files (for example called Elementary Streams, typically with a file extension of .m4v or .cmp in MPEG-4) that MTS4EA analyzes are defined in the relevant standards. Proprietary file formats are not supported.

This means that the compressed video file to be analyzed should not be an AVI file (which puts a wrapper around the encoded video and audio).

The compressed video elementary stream file cannot contain audio, although the original container file (such as MP4 or 3GPP) may have contained audio.

Once the video is extracted, the MTS4EA will automatically determine the format of the compressed video in the file by searching through the video and looking for the appropriate start codes. Accordingly, the file name does require a particular file extension to be decoded correctly.

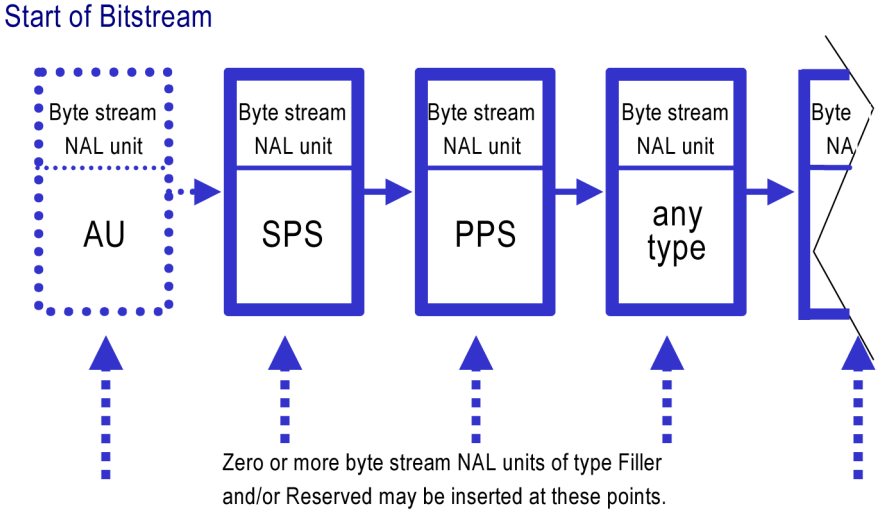
H.264/AVC and H.265/HEVC Byte Stream file format. In accordance with the H.264/AVC and H.265/HEVC standards (see Reference [12] and Reference [27] in *Standards references*), the video file must start as given in the following diagram.

The abbreviations mean:

AU: access_unit_delimiter_rbsp_nal_unit

SPS: sequence_parameter_set_nal_unit

PPS: picture_parameter_set_nal_unit



For example, if the bitstream does not contain the optional items at the start then the first six bytes in an Extended profile byte stream file will be: 00 00 00 01 67 58 (all values in hexadecimal)

VC-1 Advanced Profile and Elementary Stream file format. Within the VC-1 standard (see Reference [17] in *Standards references*), only the Advanced Profile can exist as a separate Elementary Stream. (The Simple and Main Profiles are required to be part of a container file - such as RCV, ASF or MPEG-2 TS - so that the required header information is stored.)

MTS4EA fully supports the VC-1 implementations of the Simple,

Main

and Advanced Profiles in each standard.

(See the standards in Reference [16] in *Standards references* for details of the start codes and syntax permissible for each of these.)

MPEG-4 Elementary Stream file format. In accordance with the MPEG-4 standard (see Reference [1] in *Standards references*), the file will start with a valid header start code that contains configuration information such as:

`visual_object_sequence_start_code (0xB0)`

or `visual_object_start_code (0xB5)`

or `video_object_layer_start_code (0x20 - 0x2F)`

or `video_object_start_code (0x00 - 0x1F)`

For MPEG-4 Short Header video there is no header code other than the 22-bit start code:

- `video_plane_with_short_header()` (see below)

The start codes actually used will vary from one stream to another.

These start codes (except Short Header) may be followed by further configuration data until the start code for the video data is reached, which must start with one of the following functions:

- `Group_of_VideoObjectPlane()` start code: 0xB3
- `VideoObjectPlane()` start code: 0xB6

See the MPEG-4 standard Reference [1] (in section 6.1.2) in *Standards references* for more information.

Note the entry points `MeshObject()` and `fba_object()` also given in section 6.1.2 of the MPEG-4 standard are not supported, but also these are nonallowable tools in the Profiles and Levels supported by MTS4EA.

This means that for

- `Group_of_VideoObjectPlane()`
- `VideoObjectPlane ()`

the 24 bits in the video file before the start code must be the Start Code Prefix: 0000 0000 0000 0000 0000 0001. This is then followed by the 8-bit start code value.

MPEG-2 Elementary Stream file format. In accordance with the MPEG-2 standard (see Reference [14] in *Standards references*), the Elementary Stream file will start with a 32-bit start code: Hex: 0x000001b3; Bin: 0000 0000 0000 0000 0000 0001 1011 0011.

H.263 file format. The first bits in an H.263 compressed file should be the 22-bit PSC (Picture Start Code): 0000 0000 0000 0000 1000 00.

Source video format

The above means that when the video was encoded, the uncompressed source format of the video file to be encoded should have been YUV 4:2:0 - that is, with 4 bytes of Y data (luminance) for every 1 byte of U and 1 byte of V data (color differences). (The color difference data is sub-sampled by 2 in horizontal and vertical directions.)

For H.264/AVC High Profiles, the following alternative image formats are also supported:

- YUV, RGB or grayscale color models
- Alternative YUV chroma sub-sampling: 4:2:0, 4:2:2, or 4:4:4
- Alternative image sample bit depths: from 8 to 12 bits

It is up to the encoder to take the input data and order it correctly, so that in the compressed bitstream, the pixel data must appear with the top left pixel first, followed by the pixel to the right. If the video data stream input to the encoder does not follow this order then it is up to the encoder to rearrange the data order.

Video image size

The maximum size of the video images that can be analyzed within MTS4EA is effectively up to 16384x16384 pixels.

For each standard, there is a maximum size and within this limit, MTS4EA will analyze video to the maximum size allowed in the video standard concerned, for example:

- 4096 pixels wide x 2304 pixels high in H.264/AVC
- 1920 pixels wide x 1088 pixels high in MPEG-2 Main Profile @ High Level

There is no minimum size.

Video file size

The maximum video file size that can be analyzed is effectively limited only by the disk space on your computer (the limit to the size is actually 1 million Terabytes [2^{60}] or 1 billion Gigabytes).

MTS4EA will open the first part of the video file and start playing and analyzing this immediately if possible: in some cases, MTS4EA must search through a certain amount of the file before there is sufficient data for display.

In addition, the results from analysis of the input video file are cached in MTS4EA memory in the Step-back buffer. If the results for the section of video to be displayed/analyzed are not in the cache then this is indicated on-screen and MTS4EA will then pause while loading the relevant data in the cache.

PAL/NTSC

Both PAL and NTSC format video can be analyzed with MTS4EA.

Permitted audio types/format

Audio is supported under the following container formats, as described in the previous section:

- MPEG-4 Part 3 ADTS files
- MPEG-2 Transport Stream, Program Stream, Packetized Elementary Stream (PES)
- 3GPP
- MP4

MPEG-2 Transport Stream files

MPEG-2 Transport Streams typically contain video, audio and other data. An MPEG-2 Transport Stream can contain more than one media stream - in fact hundreds or more; these streams can be located largely consecutively in the Transport Stream packets or split up in many areas of the Transport Stream.

Transport Streams with 188 bytes per packet, 196 bytes per packet, 204 bytes per packet and 208 bytes per packet can be analyzed.

MPEG-2 Program Stream and PES files

An MPEG-2 Program Stream contains one or more Packetized Elementary Streams (PES), with a common time base. (The Program Stream is designed for use in a relatively error-free environment.)

The format of the Program Stream is defined in Part 1 of the MPEG-2 standard (see Reference [14] in *Standards references*).

A Program Stream can contain a complete Elementary Stream, or it can contain only part of an Elementary Stream.

The MTS4EA will open MPEG-2 Program Stream files (typically with a file extension of .mpg) and display the structure of the Program Stream, such as the PES packets inside.

In accordance with the MPEG-2 standard (see Reference [14] in *Standards references*), the Program Stream file will start with a 32-bit start code: 0x000001BA, for example, 0000 0000 0000 0000 0000 0001 1011 1010

3GPP files

3GPP files are a subset of MPEG-4 files, with an indicator in each 3GPP file to denote that it is a 3GPP file.

3GPP files should conform to the standard concerned (see Reference [11] in *Standards references*).

By default, the MTS4EA looks for 3GPP files with a .3gp file extension, although any file name can be used; the MTS4EA will determine that it is a 3GPP file by looking through the file contents.

The audio streams are extracted from 3GPP files and analyzed directly or extracted and analyzed later in the same way as for MP4 files. For information on opening 3GPP files, extracting and saving the audio streams see the *How to use the MTS4EA* section.

The structure of the 3GPP file can be examined, using the **View file structure** option on the Analysis menu. (See page 5-1, *How to use the MTS4EA*.)

MP4 files

MP4 files typically contain video, audio and other data. An MP4 file can contain more than one video stream; these streams can be located consecutively or split up in many areas of the MP4 file.

The MP4 files that the MTS4EA analyzes should conform to the relevant part of the MPEG-4 standard (see Reference [10] in *Standards references*).

By default, the MTS4EA looks for MP4 files with an .mp4 file extension, although any file name can be used; the MTS4EA will determine that it is an MP4 file by looking through the file contents.

The audio streams can be extracted from the MP4 file and analyzed directly, or extracted and analyzed later. For information on opening MP4 files and extracting and saving the audio streams see the *How to use the MTS4EA* section.

The structure of an MP4 file can be examined, using the **View file structure** option on the Analysis menu. (See page 5-1, *How to use the MTS4EA*.)

Standards references

[1] MPEG-4 Part 2 (Visual): standard number ISO/IEC 14496-2:2004; ISO title: Information technology - Coding of audio-visual objects: Part 2: Visual, 3rd Edition 2004-06-01; plus 14496-2:2004 Technical Corrigendum 1 Published 2004-06-15

[2] H.263: Video Coding for Low Bit Rate Communication. International Telecommunication Union (ITU) 1998

[3] MPEG-4 Video Verification Model version 18.0: document reference number ISO/IEC JTC1/SC29/WG11 N3908 dated January 2001

[4] MPEG-4 Part 2 (Visual) ISO/IEC 14496-2 Amendment 2, 2002-02-01: Streaming video profile (contains Advanced Simple Profile)

[5] MPEG-4 Part 2 (Visual) ISO/IEC 14496-2:2001 Final Draft Amendment 3 FDAM 3:2003(E): New levels and tools for MPEG-4 visual (contains Advanced Simple Profile Level 3b)

[6] MPEG-4 Part 2 (Visual) ISO/IEC 14496-2 Microsoft reference software: FDAM1-2.3-001213 version 2 dated July 3rd 2000

[7] MPEG-4 Part 4 (Conformance Testing) ISO/IEC Study of CD 14496-4 N3067 1999-03-18, Visual clause w3067_4(v)

[8] MPEG-4 Part 4 (Conformance Testing) ISO/IEC 14496-4 MPEG-4 Normative ISO bitstreams dated 05/11/2001, specified in sections 4.5.3.1 and 4.5.7 of Reference [7]

[9] MPEG-4 Part 4 (Conformance Testing) ISO/IEC 14496-4 MPEG-4 Donated bitstreams dated 14/07/2000, referred to in section 4.5.8 of Reference [7]

[10] MPEG-4 Part 1 (Systems) ISO/IEC 14496-1: Information technology - Coding of audio-visual objects: Part 1: Systems, 3rd Edition dated March 2002

[11] 3rd Generation Partnership Project; Technical Specification Group Services and System Aspects; Transparent end-to-end Packet-switched Streaming Service (PSS); Protocols and codecs (Release 5); reference 3GPP TS 26.234 V5.5.0 (2003-06)

[12] H.264/AVC: Advanced video coding for generic audiovisual services. International Telecommunication Union (ITU) March 2010

[13] MPEG-2 Part 2 (Visual): ISO/IEC 13818-2 Second edition 2000-12-15 (2000 E): Information technology - Generic coding of moving pictures and associated audio information: Video with Amendment 1: Content description data (2001-12-15, corrected version 2002-08-01) and Technical Corrigendum 1 (published 2002-03-01)

[14] MPEG-2 Part 1 (Systems): ISO/IEC 13818-1 Third edition 2007-10-15 (2007 E): Information technology - Generic coding of moving pictures and associated audio information: Systems with Amendment 3 Transport of scalable video over ITU-T Rec. H222.0/ ISO/IEC 13818-1, corrected version dated 2009-12-01

[15] DVD Standard for Video: DVD-Video Book Part 3: Video Specifications v1.1

[16] SMPTE "Proposed SMPTE Standard for Television: VC-1 Compressed Video Bitstream Format and Decoding Process" committee draft 2, revision 1, reference number SMPTE CD xxxM (otherwise referred to as VC-1)

[17] Advanced Systems Format (ASF) Specification: revision 01.20.02; Microsoft Corporation, June 2004

[18] MPEG-2 audio: ISO/IEC 13818-3:1998 Information technology – Generic coding of moving pictures and associated audio information – Part 3: Audio

[19] MPEG-2 AAC: ISO/IEC 13818-7:2004 Information technology – Generic coding of moving pictures and associated audio information – Part 7: Advanced Audio Coding (AAC)

[20] MPEG-4 AAC (AAC plus): ISO/IEC 14496-3:2001 Information technology – Coding of audio-visual objects – Part 3: Audio plus Cor1:2002, Cor1:2004, Cor2:2004, Amd1:2003, Amd2:2004, Amd3

[21] HE AAC codecs: ISO/IEC 14496-3:2001 Information technology – Coding of audio-visual objects – Part 3: Audio plus Cor1:2002, Cor1:2004, Cor2:2004, Amd1:2003, Amd2:2004, Amd3

[22] MPEG-1 audio: ISO/IEC 11172-3:1993 Information technology – Coding of moving pictures and associated audio for digital storage media at up to about 1.5Mbit/s – Part3: Audio

[23] MPEG-4 Part 14: ISO/IEC 14496-14:2003 Information technology – Coding of audio-visual objects – Part 14: MP4 file format

[24] Digital Audio Compression Standard (AC-3) ATSC A/52, (1995-12)

[25] Operational Guidelines for Digital Terrestrial Television Broadcasting, ARIB Technical Report TR-B14, Version 2.8.

[26] H.264/SVC: Advanced video coding for generic audiovisual services – Annex G (Scalable Video Coding). International Telecommunication Union (ITU) March 2010

[27] H.265/HEVC: ITU-T H.265, Series H: Audiovisual and Multimedia Systems, Infrastructure of audiovisual services – Coding of moving video, High efficiency video coding April 2013

[28] HEVC in TS: ISO/IEC 13818-1:201X/PDAM 3 (E) Rec. ITU-T H. 222.0/Amd.3, Information Technology – Generic Coding of Moving Pictures and Associated Audio Information: Systems, Amendment 3, Transport of High Efficiency Video Coding (HEVC) Streams over ITU-T Rec H.222.0 | ISO/IEC 13818-1

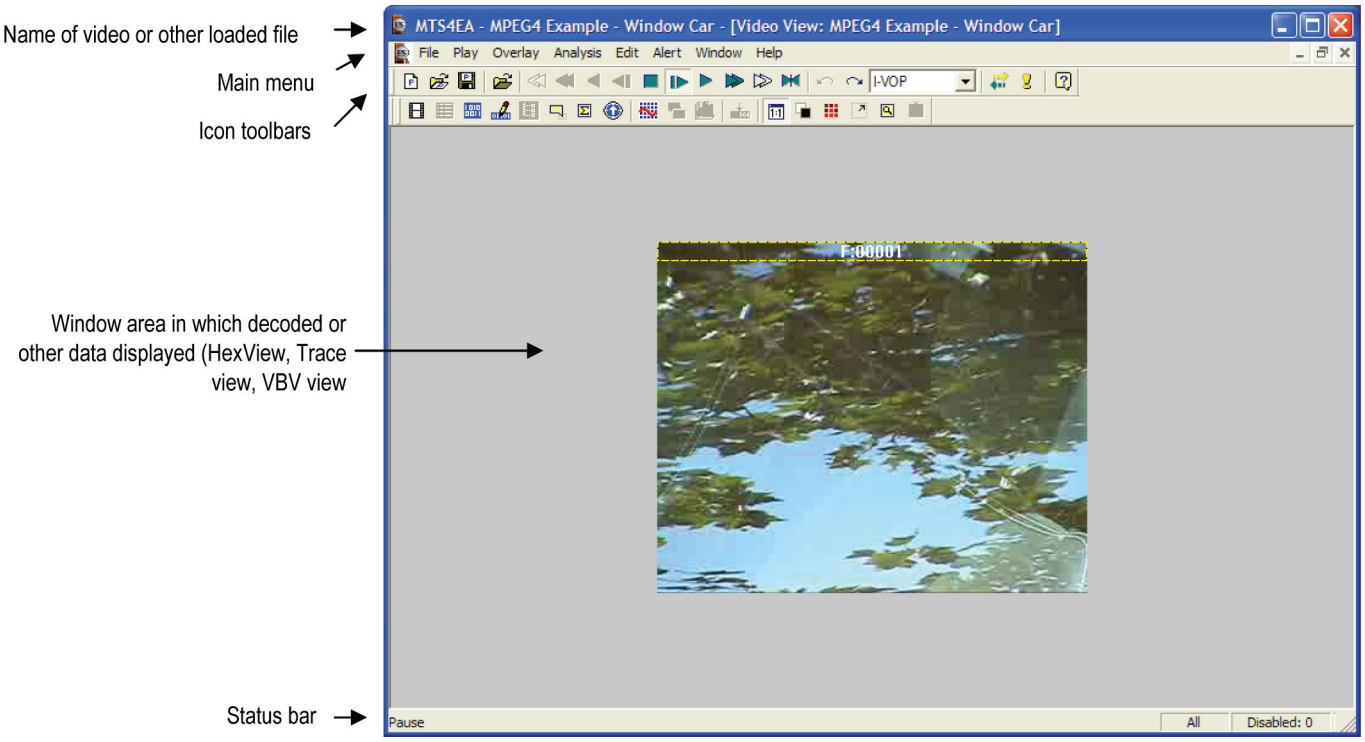
How to use the MTS4EA

How to use the MTS4EA

This section:

- Describes the elements of MTS4EA display window. (See page 5-2, *Window elements*.)
- Tells how to start using MTS4EA display. (See page 5-3, *Starting to use the MTS4EA*.)
- Describes each of the menu items in detail and how to use them. (See page 5-13, *Main menus*.)
- Explains the icon toolbars, below the main menu. (See page 5-291, *Icon toolbars*.)
- Explains the context-sensitive toolbars, which appear depending upon the video standard used and/or the actions being performed. (See page 5-294, *Context-sensitive toolbars/tooltips*.)
- Explains the information provided on the status bar. (See page 5-284, *Status bar*.)
- Lists the shortcut keys, such as Ctrl+A to Pause/Frame Advance one frame. (See page 5-299, *Ctrl shortcut keys*.)
- Explains the use of the Alt key to select menu items. (See page 5-300, *Alt menu keys*.)
- Explains the use of the command line/batch mode. (See page 5-301, *Command line/batch mode*.)

Window elements

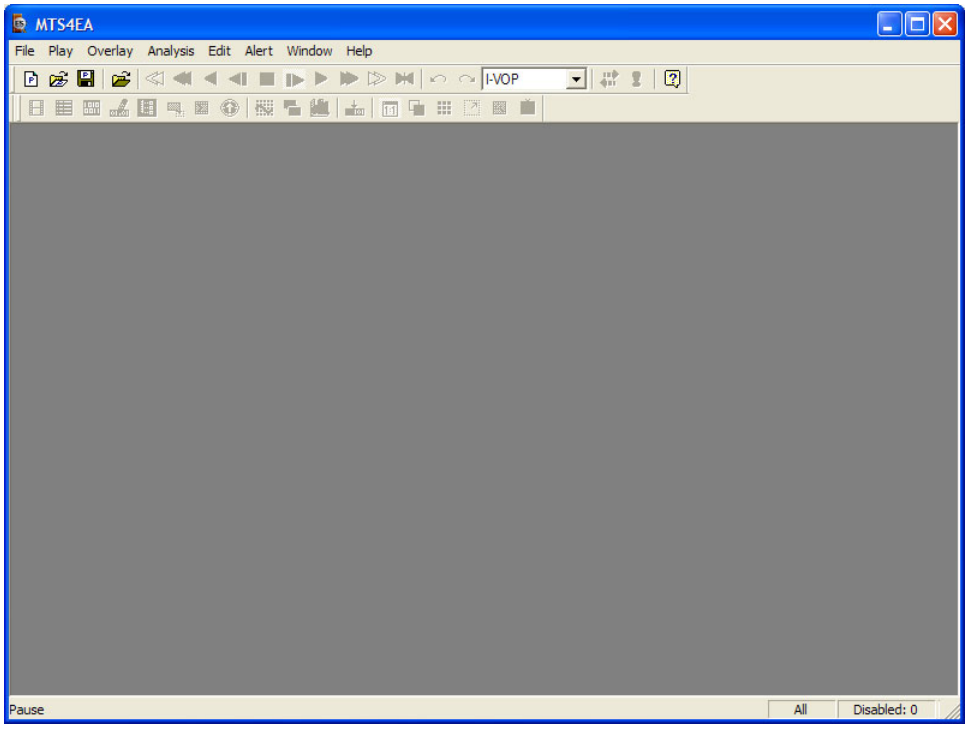


NOTE. *The above screenshot was taken with the MTS4EA running under Windows XP.*

In addition to the above toolbars/windows, there are multiple other toolbars that are visible when particular functions are enabled. Also, there are some toolbars, tooltips and menu items that are context-sensitive, depending on the video/audio standard being used and/or the explicit function being performed.

Starting to use the MTS4EA

When the MTS4EA starts, the display initially looks as below:

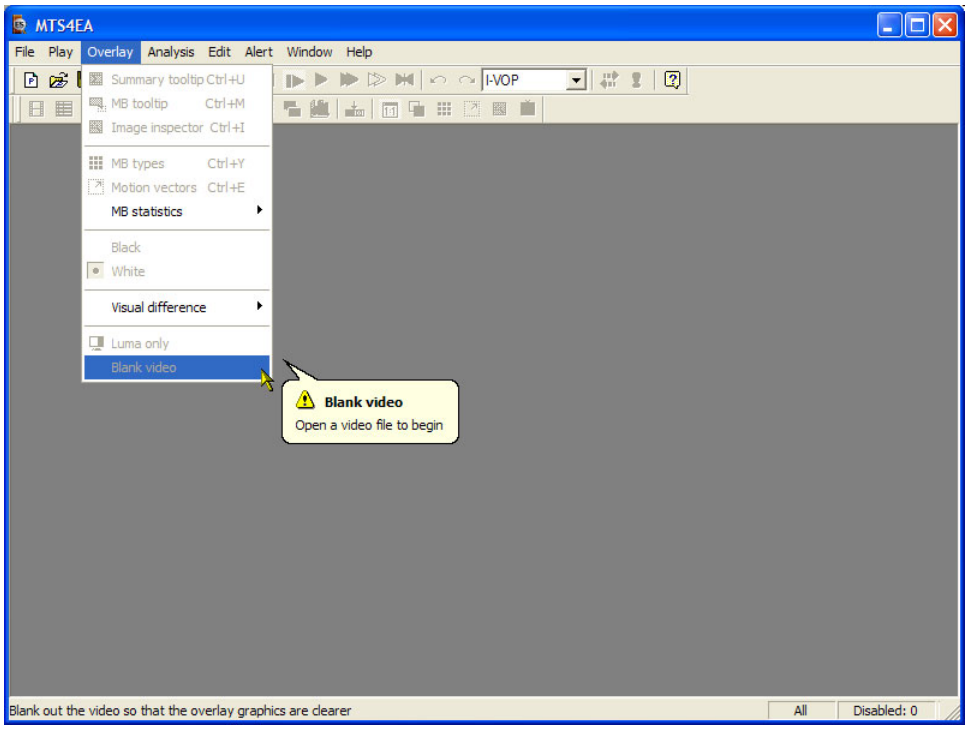


The gray background in the window indicates that no video or other files are loaded. The only icons on the toolbar that are currently enabled are the icons:

- For the Project (New Project, Open Project and Save Project)
- Open stream
- Help

NOTE. *Some features on some menus may stay permanently grayed out (unavailable). There are two possible reasons for this: the function or feature is never enabled or is not relevant to the standard you are using; or the existing license you have for MTS4EA does not allow all the features to be used - see MTS4EA License Manager in chapter 4 for more information.*

When menu items are grayed out, leaving the mouse at the location of the grayed out menu item for a few seconds will display some popup text explaining why the menu item is disabled and/or how it can be enabled. The following example is on the Overlay menu, for the Blank video menu selection:



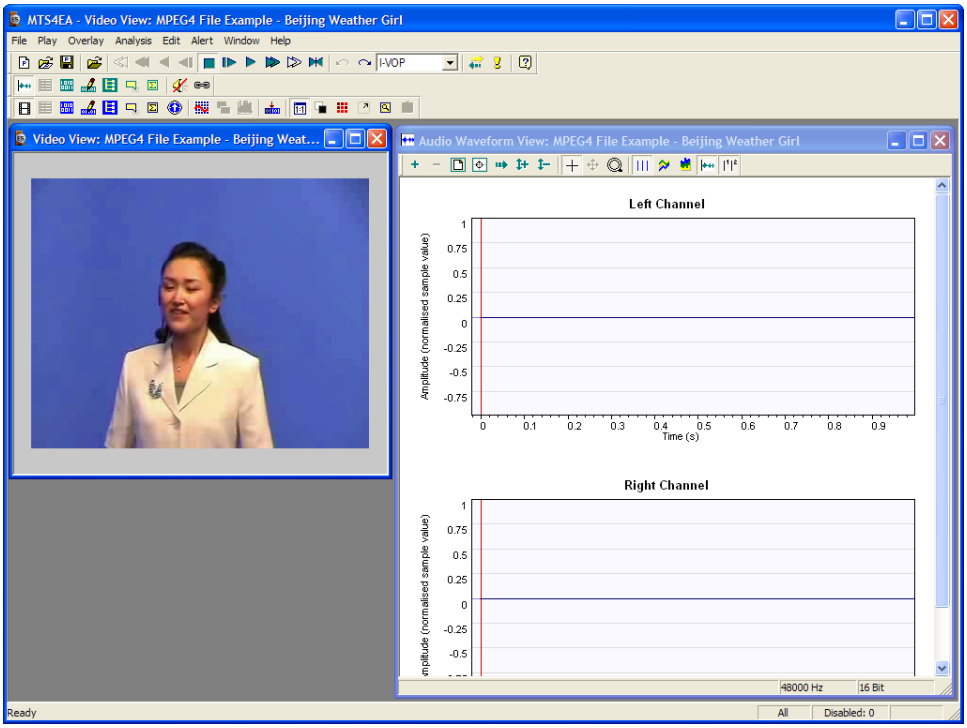
Opening a file

To open a file, do one of the following:

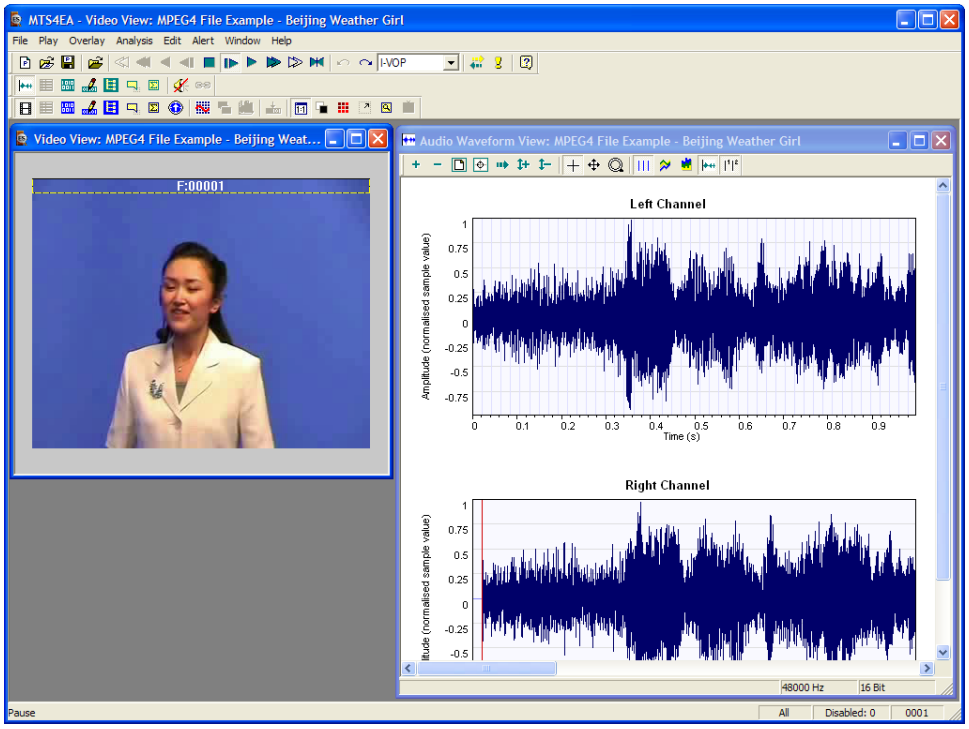
- **Click on File > Open stream**
- Click the Open file icon
- Press Ctrl+O

When a stream is loaded, the first frame is decoded and displayed by the MTS4EA. While the first frame of the stream is displayed, the stream itself is not in Play mode, therefore Analysis options can be selected and configured.

In the following screenshot, the example file Beijing Girl is loaded with its first frame displayed, but as indicated by the Stop icon and the empty frame counter in the bottom right corner of the screen, the stream is yet to be played. Where there is audio content, the audio waveform view is also displayed.



If the loaded stream is initially advanced by one frame, the first frame is again decoded/played before entering the Pause mode. In this mode, Analysis options are not available. When stopped, the first frame reappears on the display until the stream is advanced/played further.



As is normal with Windows applications, the window can be dragged around within the frame of the application, resized, maximized, and minimized.

If another file is opened, then this replaces the file last opened; another window is not opened.

NOTE. *You can open several copies of the MTS4EA to allow direct, side-by-side comparisons of multiple streams.*

Playing mode restrictions

There are two Play modes: these are indicated at the lower left corner of the status bar:

- **Stop**: stream is stopped
- **Play**: stream is paused (after Pause/Step forward) or playing at normal speed, Fast forward, Blind, or Silent fast forward.

Certain actions can be performed only when the open file is in the appropriate Play mode. A few such examples are:

- The Trace enable and Graph enable options on the Analysis menu are only available in Stop mode. (See page 5-147, *Video trace enable... Ctrl+T*.) and (See page 5-172, *Video graph enable... Ctrl+G*.)
- The real-time onscreen image overlays, such as MB types and motion vector overlay do not update the video window when in Stop mode

  (See page 5-91, *MB types Ctrl+Y*.) and (See page 5-101, *Motion vectors Ctrl+E*.)

NOTE. *See also Audio/Video Synchronization for further information and details on playing audio and audio-video synchronization. (See page 5-44.)*

MTS4EA projects

When working with the MTS4EA, the projects feature allows all current setups to be saved at any point and reloaded, for example, to allow work to recommence at exactly the same point. The project file includes information on the name of the file being analyzed, the open views and the location of these, plus many other items. (See page 5-31, *New project*.)

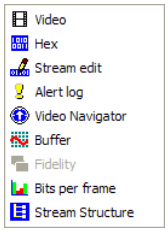
Synchronized views/navigating the views

When a video or audio stream is played, MTS4EA has many views of the video, audio and other data:

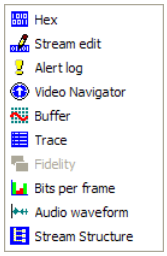
- Alert log
- Video view
- Audio/video hex, stream edit and navigator views
- Buffer analysis view
- Fidelity view (for example PSNR analysis)
- Bits per frame view
- Trace views
- Audio waveform, hex, alert, trace and stream edit views
- File structure view
- Excel graphs

Some of these views are linked. A right-click in one view will open a context menu with a Goto command that will take you to the corresponding location in another view (provided the file being played has both video and audio streams in order to bring up the Audio waveform view).

Example of Goto view menus:



From Trace file



From video view

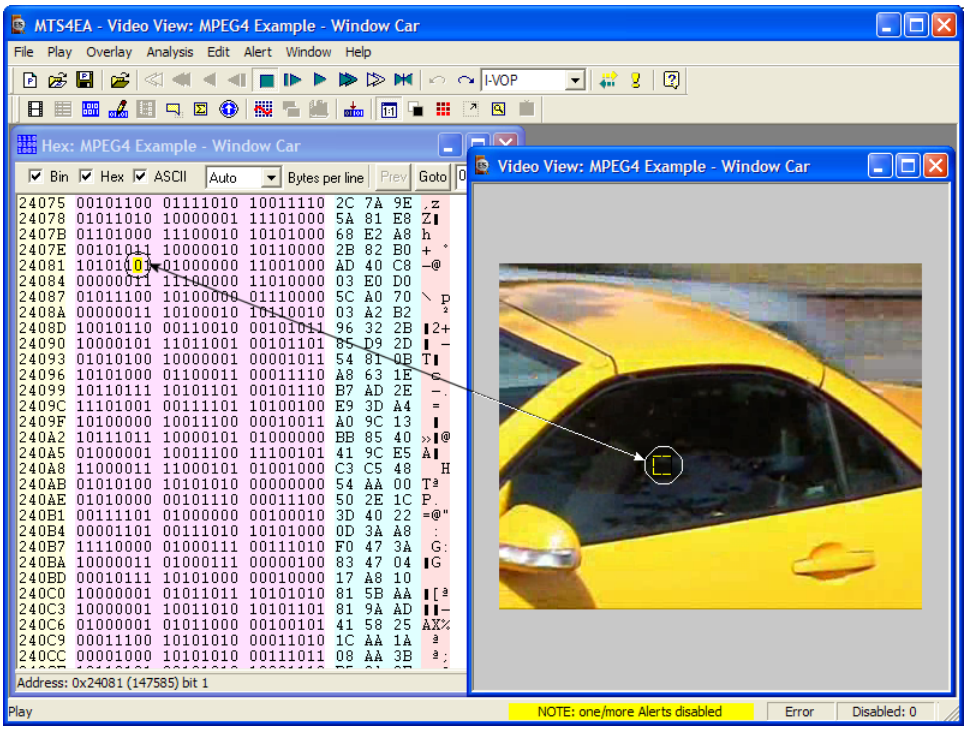
NOTE. *When in the video or audio view, to make it easier to see which area of the window is providing the link to another view, there is always a small square or rectangle in the view that highlights the MacroBlock/CTU or frame for cross-referencing in the other view.*

Example of moving between views. For example, when in the video view and using the MacroBlock/CTU tooltip, to see the hex/binary data for a specific MacroBlock/CTU, right-click at the MacroBlock/CTU location of interest and select Goto view from the context menu, and then select the HexView. The HexView window opens (if it is not already opened for this stream) and the first byte of the selected MacroBlock/CTU is highlighted.

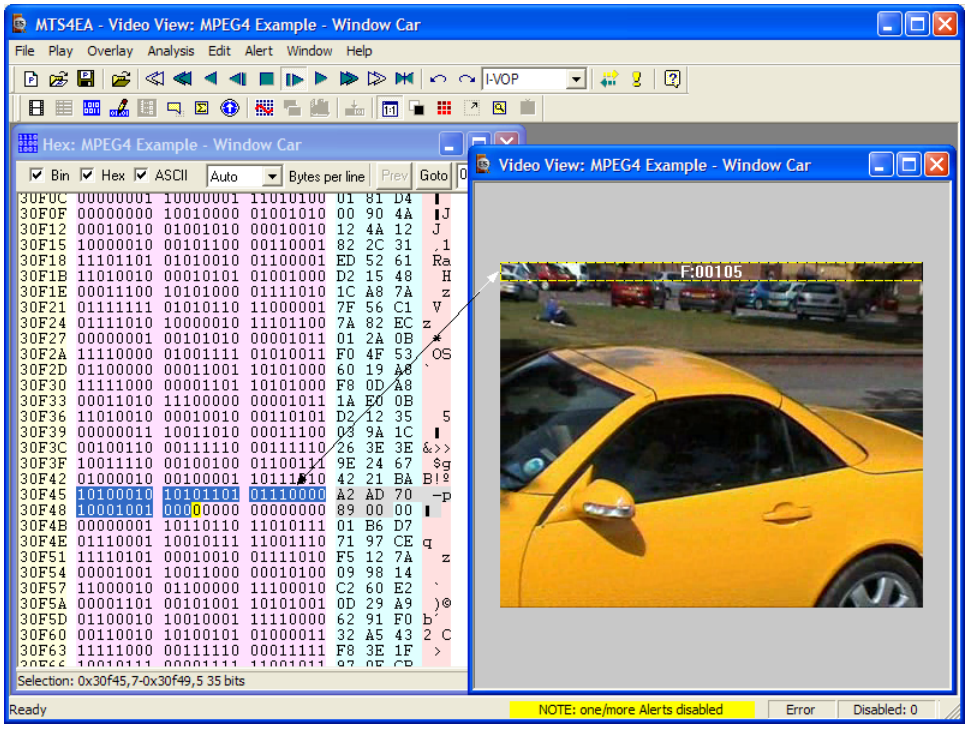
Selecting a specific byte in the HexView, and then using the right-click context menu to select the Video view will take the video window to the MacroBlock/CTU that contains the selected byte, with the MacroBlock/CTU highlighted. (See page 5-9, *Highlighting of selected areas in the video view*.)

Highlighting of selected areas in the video view. If a data area is selected in a view other than the video view, and the context menu is used to go to the video view, the corresponding area of the video view is highlighted, as follows:

- If the area can be resolved to a single MacroBlock/CTU, then a yellow and black dotted box is displayed around the MacroBlock/CTU concerned:



- If the area cannot be resolved to a single MacroBlock/CTU, then a yellow and black dotted rectangle is displayed surrounding the top row of MacroBlocks/CTUs in the video frame that contains the first selected area:



- If the selected area is in a frame or file header, not within a specific MacroBlock/CTU, then a yellow and black dotted rectangle is displayed surrounding the top row of MacroBlocks/CTUs in the video frame that immediately follows the frame or file header (see the preceding illustration: the data selected in the HexView window is in the VOP header, so the highlight rectangle is displayed across the top row of MacroBlocks/CTUs of the corresponding VOP).

Highlighting of selected areas in the audio view. Highlighting selected areas in the audio waveform view works in a similar manner to highlighting areas in the video view. (See page 5-9, *Highlighting of selected areas in the video view*.)

NOTE. *An audio frame consists of a fixed number (per channel) of audio samples representing a fixed time length of decoded audio. Typically each iteration through the decode loop will output one audio frame of decoded audio.*

Synchronization of views/Synchronize views icon. When the Synchronize views icon is pushed in, then all open windows automatically follow the selection that you made.

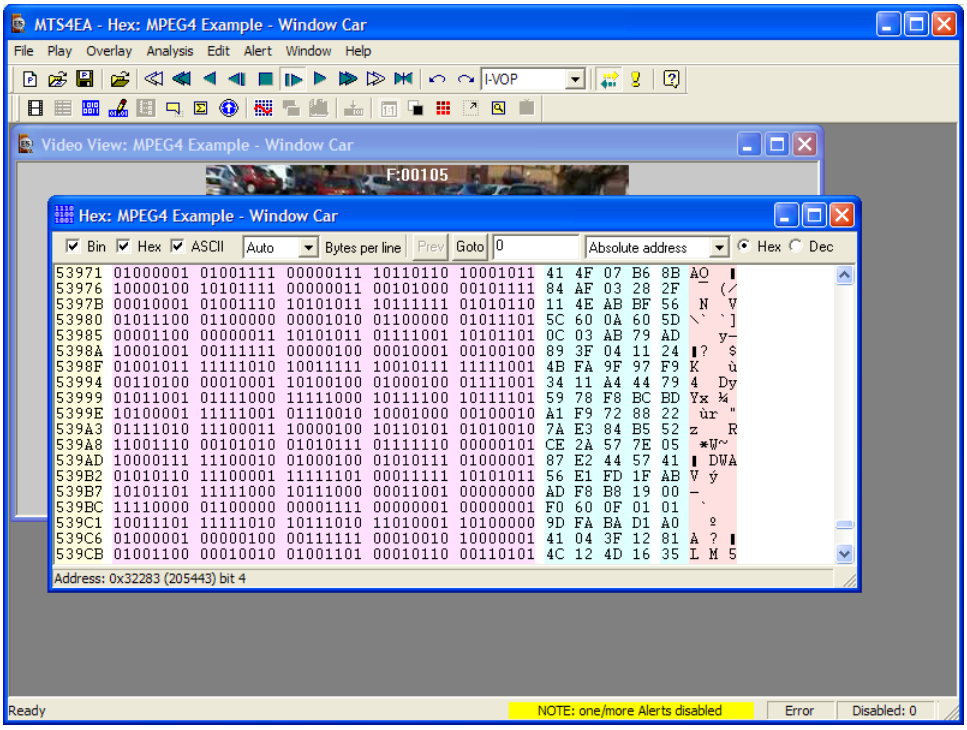
For example, if the Video Navigator and Hex views are both open at the same time as the Audio waveform view, and the Synchronize views icon is pushed in, and video is then played, all views display the same relative points in the stream at the same time.

NOTE. *This can cause a delay when playing video for some views (such as the Trace views) or when the memory buffers used by the MTS4EA do not contain all the required data and therefore time is taken to decode and interpret the video and refill the buffers.*

Opening multiple windows

Within the MTS4EA it is possible to open many windows at once. These can be:

- A video/audio window with another window, such as a HexView window with a video window (as shown in the following figure):



- Other combinations, such as the video/audio and HexView with the Trace file view and file structure view.

To open windows other than for video files (or files that contain video, such as MP4 or 3GPP files, or MPEG-2 Program/Transport Streams):

- For Trace views, do one of the following:
  - Select **File > Open other** (or click the icon) to open a Trace file previously stored or associated with another video stream
  - Select **Analysis > View trace** (or Ctrl+V) to open the current trace file, as named in the Analysis menu - Trace tab
- To view the statistical graphs (in Excel), click on **Analysis > View graphs**
- For HexView, click on **Analysis > View stream hex** (or click the icon or press Ctrl+H)
- For the stream structure view, click on **Analysis > View file structure** (or click the icon or press Ctrl+R)
- For the buffer analysis view, click on **Analysis > View buffer analysis** (or click the icon)
- For the fidelity analysis, click on **Analysis > View fidelity analysis** (or click the icon)
- For the Alert Log, click on **Alert > View alert log** (or click the icon)
- For the Audio waveform view, click on the icon . (If there is audio content, the toolbar and the view will be displayed automatically when the video view is opened.)

NOTE. *The Audio waveform view is available only when an audio stream or file/project containing an audio stream or streams is open.*

Main menus

The submenu options available under each main menu selection (for example, File, Play, Overlay) are described in detail in their respective sections. A summary of each main menu item is given here.

Several of the common functions can be accessed through:

- The icon toolbars. (See page 5-291, *Icon toolbars*.)
- Ctrl keys. (See page 5-299, *Ctrl shortcut keys*.)

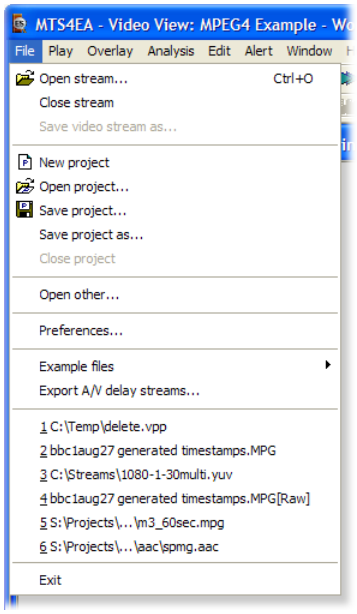
NOTE. *You can open more than one copy of MTS4EA to allow direct, side-by-side comparisons of multiple video streams.*

Functional summary for each main menu

Menu name	Principal functions/suboptions
File	Opening and saving files and projects, recent file list, example files, exit
Play	Play video and audio, stop, pause, fast forward, set decoder options
Overlay	Hide/show video/audio data and tooltips
Analysis	Set Trace and Graph options; view Trace, Graph files, Hex view, file structures, buffer analysis, fidelity analysis
Edit	Edit elementary video/audio stream, edit bitstream selection, save edited stream
Alert	Enable/disable/configure overall alert levels and individual alerts
Window	Arrange windows and toolbars, view settings, synchronize views, window list
Help	Help topics, PDF help and tutorials, license manager, version info

File menu

The following paragraphs describe the options available in the File menu.



Open stream... Ctrl+O

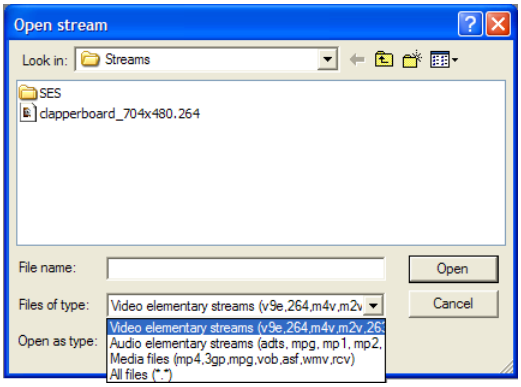
With this option, you can browse the file system to find a file to analyze. This file can be:

- A file containing only audio, for example, an MPG file
- A compressed video file that contains video data only, for example, an MPEG-4 video Elementary Stream. (See page 4-1, *Compression standards and file types*.)
- A container file that includes audio and transport information as well as video, for example MP4, 3GPP and MPEG-2. (See page 4-1, *Compression standards and file types*.)
- An uncompressed video file. (See page 5-25, *Opening an uncompressed video file (any file extension)*.)

Autodetect file type

When Autodetect is selected in the Open as type drop-down box, the file contents are automatically selected by the MTS4EA.

In this case, the Files of type drop-down is simply used to narrow the list of file extensions searched for:



The default file extensions that the MTS4EA looks for are:

- For video-only files: .v9e, .264, .m4v, .m2v, .263, .261, .cmp, .bits, .bit, .bin, .265
- For audio-only files: .adts, .mpg, .mp1, mp2, aac, ac3
- For container or media files including video: .mp4, .3gp, .mpg, .vob, .asf, .wmv, .rcv

With Autodetect selected, whichever of the file types/extensions is selected, the MTS4EA will automatically override the selection and determine the standard that the file is compressed to.

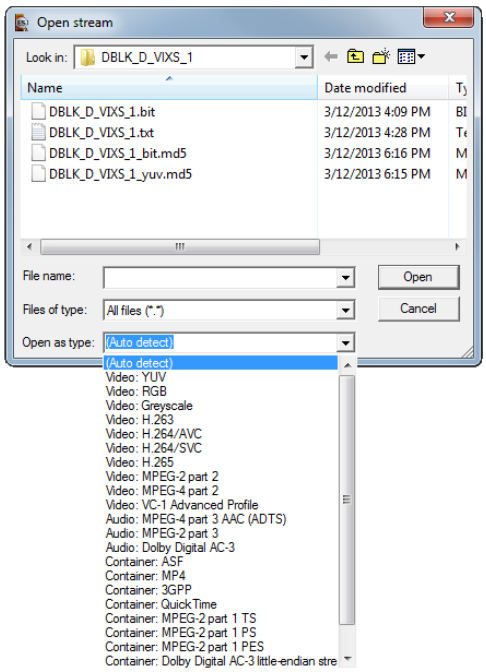
NOTE. *The exceptions to this autodetection are VOB files, where MTS4EA will take different actions due to the VOB extension.* *(See page 5-14, Open stream... Ctrl+O.)*

The selected file extension is remembered by the MTS4EA for the next time a file is opened.

Force to open as a specific file type

Sometimes, particularly if there are errors in a file, the autodetect function of the MTS4EA does not correctly identify the type of data contained in the file.

In this case, the file type can be forced by selecting a file type from the drop-down Open as type list. In this case, the file extension is ignored.



NOTE. *To open an uncompressed video file, select the appropriate item from the list and the MTS4EA will open the file in the appropriate format, regardless of the file extension or the data found in the file.*

Opening a video/audio elementary stream

When a raw video/audio elementary stream file is opened, initial checks are done on the file for validity and to determine the compression standard.

When the file has been read in correctly, the appearance of the Play icon on the toolbar changes from grayed out to blue , to indicate that the file is ready to play.

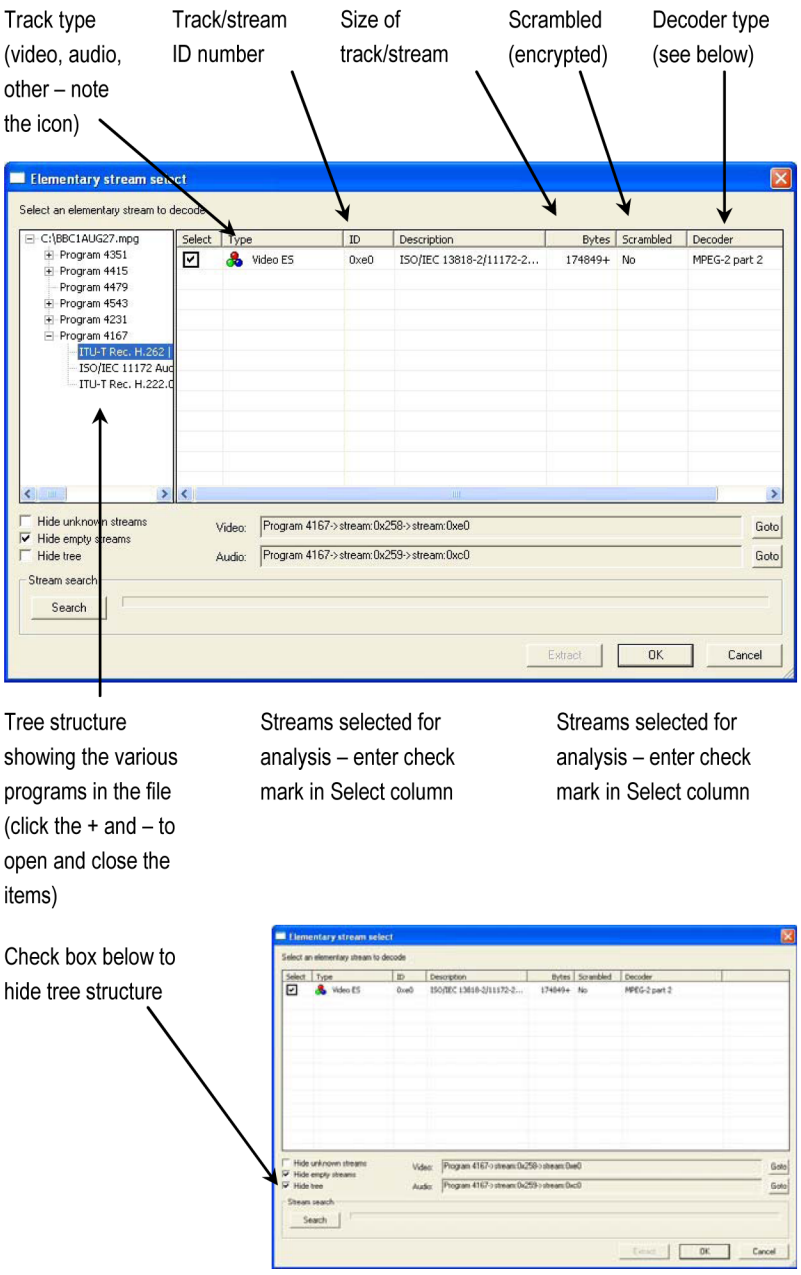
If MTS4EA recognizes that the open file includes an audio stream, the toolbar is displayed.

- Opens the Audio waveform view. (See page 5-136, *Show audio waveform*.)
- Turns the sound off/on: the stream continues to play and decode even when sound is disabled. (See page 5-62, *Mute audio*.)

If the file contains only audio data, the video toolbar will not be displayed. The Synchronize audio icon is also unavailable, as there is no video with which to synchronize. (See page 5-44, *Audio/video synchronization*.)

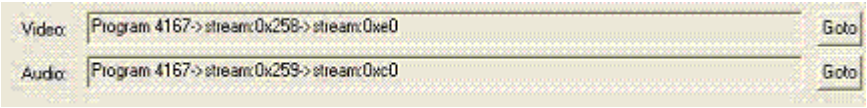
The title of the elementary stream is displayed in the title bar of the video and audio window.

Opening a container file



When a container file is loaded, the MTS4EA automatically opens the Elementary stream select window (as shown above), in which are listed all streams, audio and video. The streams contained can be browsed, individually selected for analysis, with your choice of decoder, and either immediately played and analyzed within the MTS4EA or extracted and saved to a file before going on to play/analyze them.

Select. To select the stream, audio or video, for analysis, enter a check mark in this column beside the desired stream. Its description appears in the Video or Audio fields under the browsing area (as shown in the following screenshot). Only one video and one audio stream can be selected; if a stream is selected and a check mark is subsequently entered beside another stream of the same type, the description will change to show the updated selection.



Clicking on Goto will move the view in the browser window to the location in the tree structure containing that stream, where it can be deselected or the Decoder can be chosen from the drop-down menu (see below).

Type. This is the media stream type as identified by the container signaling information. If insufficient signaling information is present, this may show a general type such as Video ES (Elementary Stream), Audio ES or, where the type is unknown, no text at all.

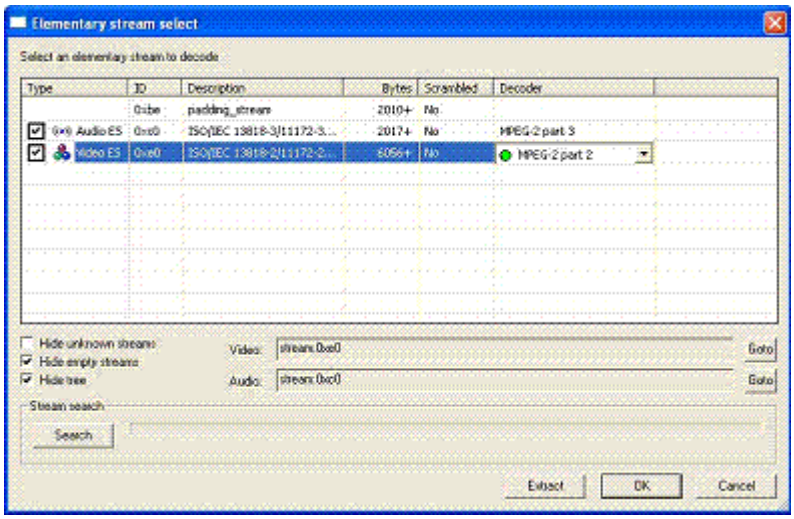
ID. This is the media-stream identifier as dictated by the container signaling information, using appropriate notation for the container standard.

Description. This is a description of the media stream as dictated by the container signaling information, incorporating text from the relevant standard.

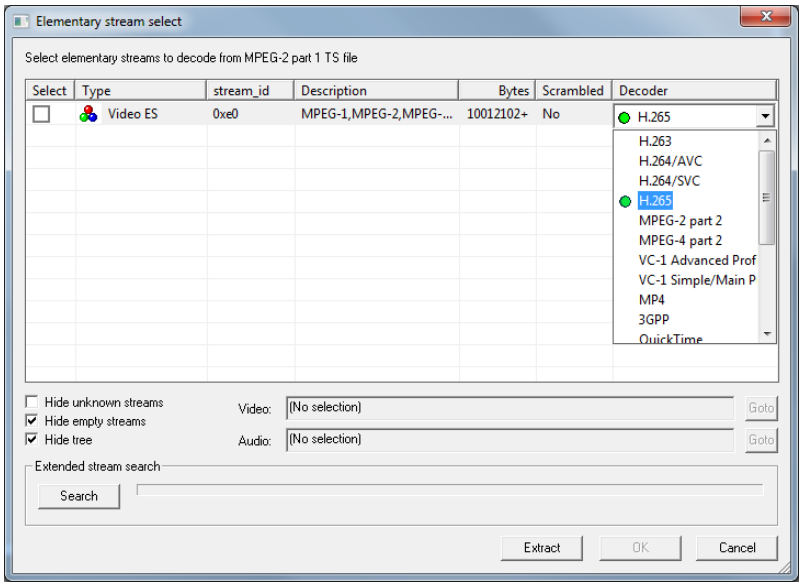
Bytes. This is the number of bytes that will be extracted for the stream. If the value is suffixed by a + sign, the total stream size is not known; without the + sign, this value is the final total.

Scrambled. This column shows Yes if the stream is scrambled or encrypted in some way. Scrambled streams cannot be decoded.

Decoder. Once a stream is selected, the decoder must be chosen. The Decoder column shows the decoder(s) found for the selected stream. There can be multiple streams containing video or audio, each with different decoders. If the Decoder column is empty, then the MTS4EA has been unable to match one of its decoders with the data it found.



Clicking on the drop-down arrow shows a list of all the available decoders, with the decoder that has been autodetected shown by a green circle:

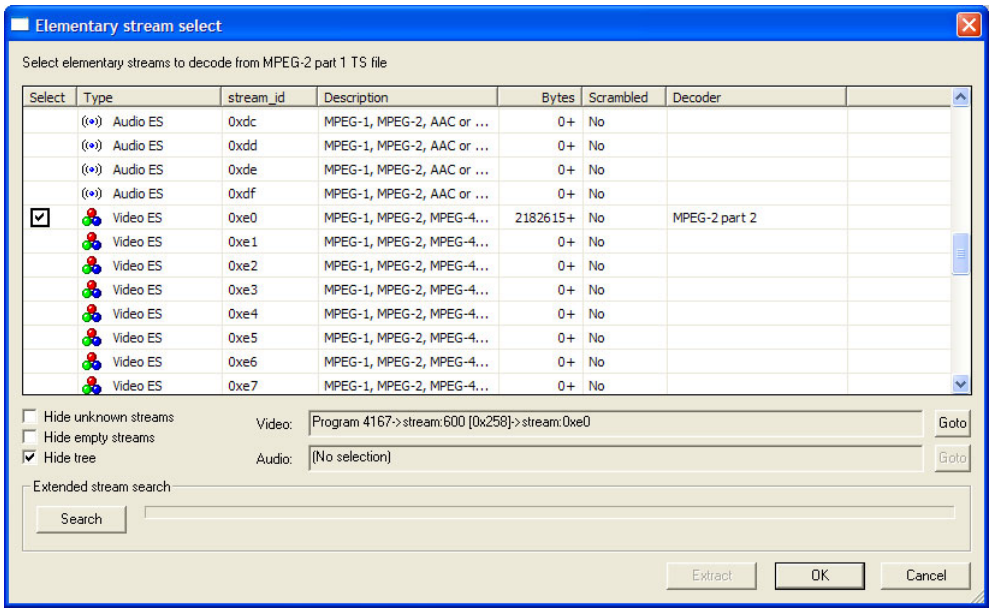


Selecting a different decoder forces the MTS4EA to use that decoder for this track/Elementary Stream.

Hide unknown streams. When selected, Hide unknown streams means that the streams are not shown for which the MTS4EA has been unable to match one of its decoders.

When Hide unknown streams is enabled, the empty streams are also hidden, because they will not contain valid data for which the MTS4EA can select a decoder.

Hide empty streams. By default, this is enabled. When enabled, this hides streams that appear to be 0 bytes in size (note that these can still be searched to see if they are in fact larger - see below, under Search).

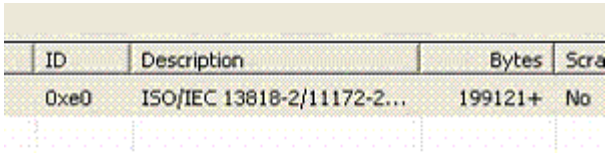


Search (extended stream search). For some standards, the total size of the stream is indicated in the container file - for example, MPEG-4. In this case, the size of the elementary stream is shown.

For some container formats, the size of the elementary streams is not indicated in the container file, and the only way to know the total size is by searching the entire file. (An example of this is MPEG-2 VOBs.)

For these types of files, the MTS4EA reads the first part of the file only, until an MTS4EA decoder can be matched with the data found.

In this case, the MTS4EA will display the size in the Bytes column with a + sign to the right of the size, as shown in the following figure:

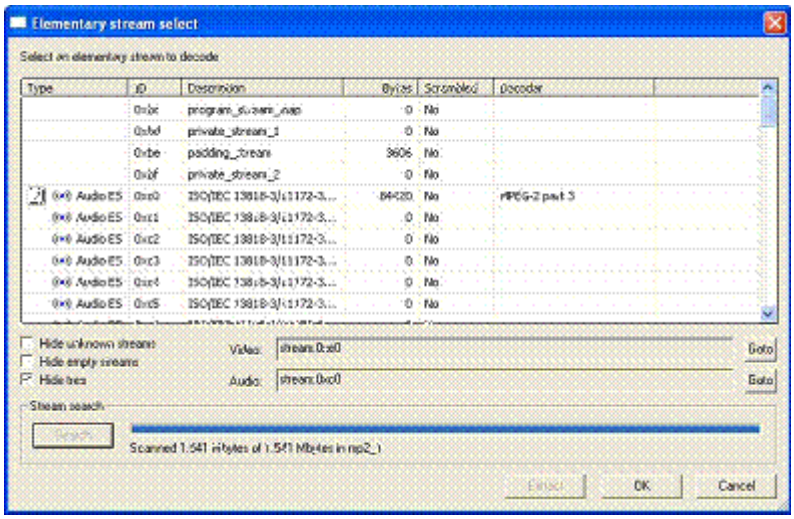


The + sign indicates that the whole file has not been searched and that there may be additional data.

NOTE. *This stream selection also means that other views, such as the HexView, may not show the data from the whole file (in which case, a warning is shown).* *(See page 5-204, View video stream hex Ctrl+H.)*

In order to find the total size of the whole file, and to search other streams, use the Search button. All streams, including the highlighted, or selected, stream, are then searched.

When the search is complete, the display will be similar to the following figure:



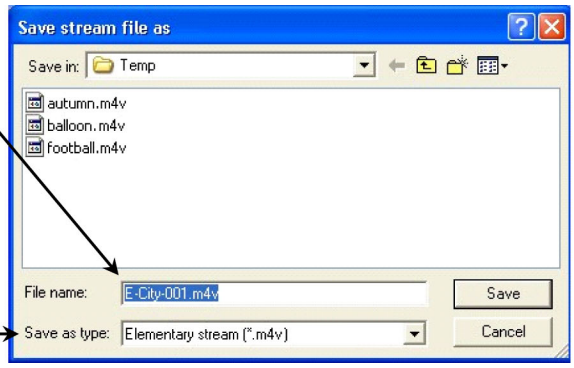
During the search, the blue progress bar fills, until the search is complete.

During the search process, the MTS4EA will search through all the linked files, and therefore may find other streams. It will be able to determine the total size of the streams found. The result is that the additional streams (and their sizes) will be shown in the stream window.

Extract. This allows the highlighted track, audio or video, to be saved as a separate elementary stream (in the case below, .m4v is offered as the file type, as the container file was MP4):

The video track ID from the MP4/3GPP file is appended to the name of the MP4/3CPP file name

A file extension is added automatically, depending on the video type (see below)



NOTE. *If you decide to play and analyze the selected stream immediately, and then decide to save it later, this can be done by clicking the Save as option on the File menu. (See page 5-30, Save video stream as....)*

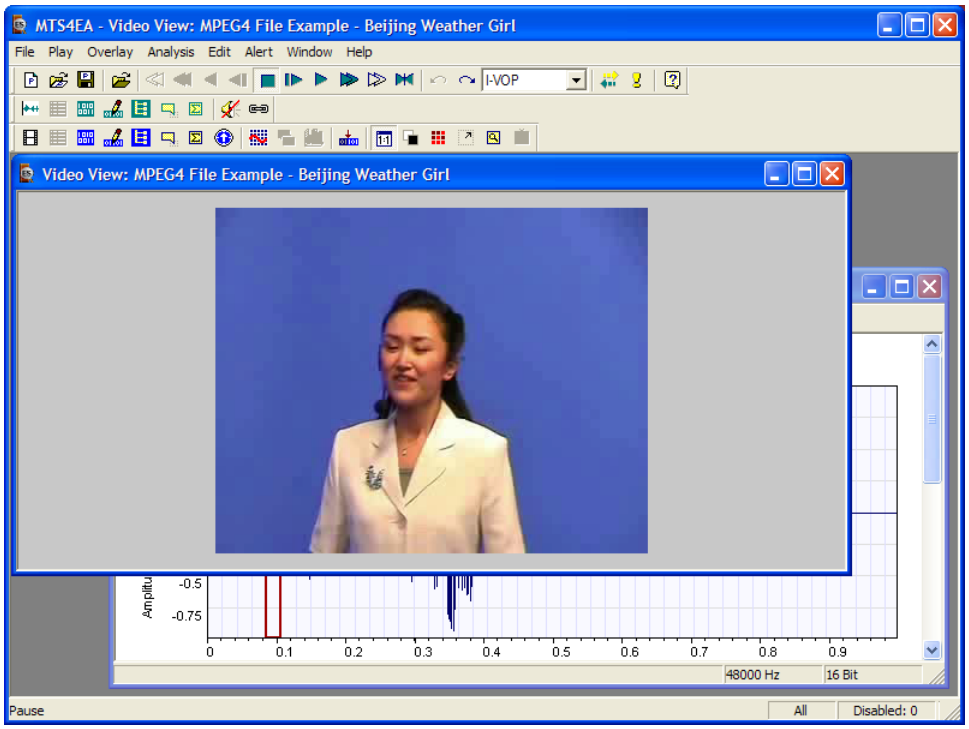
The correct type is automatically added as a file extension, depending upon the data in the file:

- H.265/HEVC video format files are given the extension: .265
- H.264/AVC video format files are given the extension: .264
- H.264/SVC video format files are given the extension: .264 or .svc
- VC-1 Advanced Profile video files are given the extension: .v9e
- MPEG-4 video format files are given the extension: .m4v (except for MPEG-4 Short Header, with extension .263)
- MPEG-2 video format files are given the extension: .m2v
- H.263, and MPEG-4 Short Header video format files are given the extension: .263

NOTE. *The suggested file name and extension do not have to be accepted - any file name and extension can be entered.*

OK. OK opens the selected stream using the decoder shown; if the OK button is grayed out, then the selected stream cannot be opened.

A window is opened with the title of the container file with the track/stream number.



Because the MTS4EA has recognized that the open file contains both audio and video streams, the video and the audio views are opened. The audio toolbar is also displayed and the synchronize is enabled. (See page 5-62, *Synchronize audio*.)

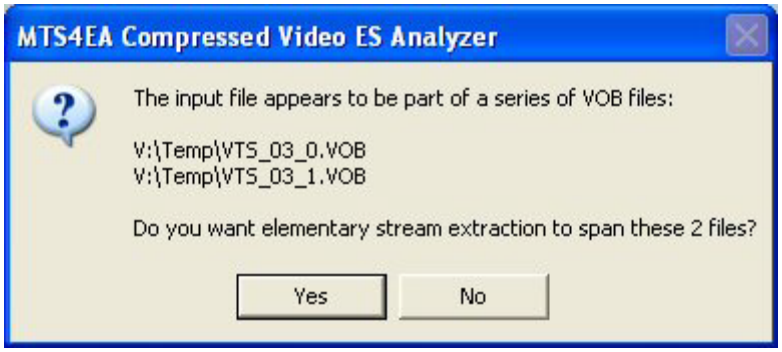
NOTE. *If you decide to play and analyze the selected stream immediately, and then decide to save it later, this can be done by clicking the Save as option on the File menu. (See page 5-30, Save video stream as....)*

To examine the structure of the file, click the View file structure option on the Analysis menu. (See page 5-210, *View video stream structure Ctrl+R*.)

Opening a VOB/series of VOBs (.vob file extension)

If there are two or more VOBs with sequential numbers from the first VOB selected, then MTS4EA will display a popup box:

NOTE. *This only applies to files with a VOB extension (the MTS4EA treats these files differently when compared to files with other file extensions).*



Clicking Yes means that the MTS4EA will extract the video Elementary Stream from all the VOBs indicated; clicking No means that the MTS4EA will only select the Elementary Stream from the VOB selected in the File open dialog box.

Once this is done, the MTS4EA then uses the standard stream selection dialog box as used for all container files. (See page 5-18, *Opening a container file*.)

Opening an uncompressed video file (any file extension)

Uncompressed video files can be opened by selecting YUV, RGB or grayscale in the Open as type list.

In each case the MTS4EA ignores the extension and does not look for a compressed video standard within the data - it assumes that the data is in one of the following formats.

The YUV data is either:

- 8 bits per sample, 4:2:0
- More than 8 bits per sample, and/or 4:2:2 or 4:4:4 (as used by H.264/AVC High 10, High 4:2:2 and High 4:4:4 profiles and MPEG-2 4:2:2 profile)

YUV format of 8 bits per sample 4:2:0. (See page 5-187, *YUV format of 8 bits per sample 4:2:0*.) The YUV file output is raw YUV with no headers of any kind. This is the same format used by the Microsoft MPEG-4 Part 2 reference encoder Reference and used commonly by other programs: (See page 4-1, *Compression standards and file types*.)

- No headers of any kind (no file or frame headers)
- One byte per sample
- Row raster order (top picture row first)

- Planar YUV 4:2:0 subsampled (4 bytes of Y data for each byte of U data and each byte of Y data)
- Y plane values are 0 - 255 unsigned
- U and V plane values are unsigned with a DC offset of 128

Other uncompressed formats. The general uncompressed video file format is as follows:

- No headers of any kind (no file or frame headers)
- Concatenated planar image data
- Row raster order (top picture row first)
- Unsigned samples

For 8-bit sample depth:

- One byte per sample

For 9-16 bit sample depth:

- Two bytes per sample
- Both little- and big-endian byte orders supported
- For YUV format:
- Concatenated Y, U and V planes
- U and V planes subsampled as required
- Y plane samples are unsigned
- U and V plane samples are unsigned with a DC offset of 2^{-1}, where *n* is the chroma sample bit depth

For RGB format:

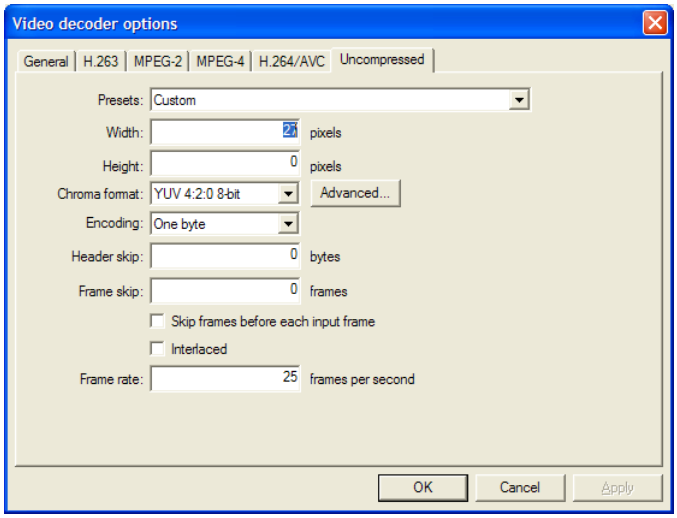
- Concatenated R, G and B planes

For grayscale format:

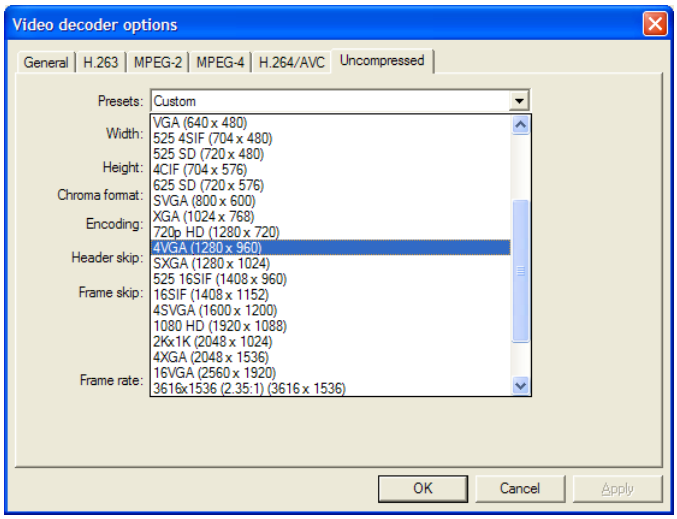
- Luma plane only

When opening an uncompressed video file, by default MTS4EA shows the uncompressed video decoder options, to enter the required parameters.

Setting the uncompressed video frame size. Within an uncompressed video file, there is no place to indicate the frame size, frame rate, and so forth. So when the uncompressed video file is opened, the MTS4EA will display the following screen (this is a tab of the Decoder options, which can also be accessed from the Play menu):



Presets (width and height). If there are numbers in the filename that could indicate the frame size, then the MTS4EA will attempt to read these and offer these in the Width and Height fields shown in the previous figure. The correct values can be entered or selected using the drop-down list:



Chroma format. This control specifies the color model and format for the uncompressed video data. The combo provides several common combinations, including 8-bit YUV 4:2:0, 8-bit YUV 4:2:2, 8-bit YUV 4:4:4, 8-bit RGB and 8-bit grayscale. Select an entry from the list to use one of these common formats. Other formats can be specified by choosing the Custom entry in the list - this will open the Image format dialog.

Advanced button. Click this button to specify a custom chroma format - this will open the Image Format dialog box. (See page 5-29, *Image format dialog box*.)

Encoding. This control specifies the encoding of image samples in the file format. The following options are available:

- One byte - this specifies that image samples are stored in one byte per sample. This format is appropriate if all image planes are 8-bits deep
- Two byte MSB first - this specifies that image samples are stored in a pair of bytes for each sample. The most significant byte occurs first in each pair (big-endian). This format is appropriate if one or more image planes are deeper than 8-bits
- Two byte LSB first - this specifies that image samples are stored in a pair of bytes for each sample. The least significant byte occurs first in each pair (little-endian). This format is appropriate if one or more image planes are deeper than 8-bits

Header skip. The number of bytes at the start of the file before the first frame; the MTS4EA will skip past these bytes (ignoring them).

Frame skip and Skip frames before each input frame. Frame skip is the number of frames (not bytes) to skip between each frame that is viewed in the video window.

By default, these frames are skipped after each viewed frame; by selecting the Skip frames box before each input frame, the number of frames is skipped before each viewed frame.

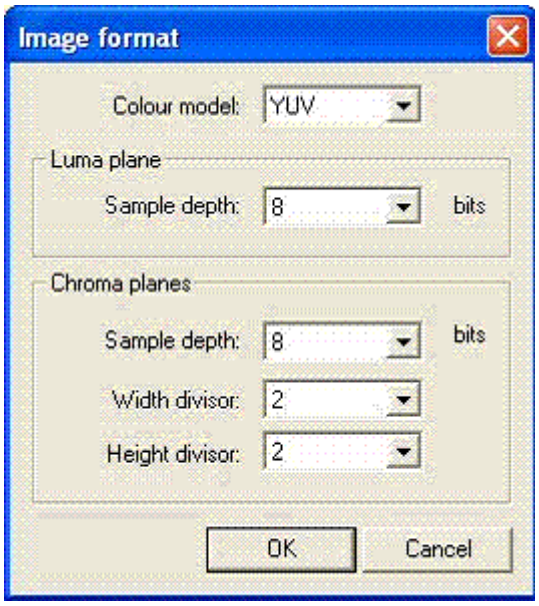
Interlaced. The uncompressed video file has interlaced data in the format of complete frames, with both fields within one frame, top-field first, each field on alternate lines.

Enabling this check box switches on the Interlace toolbar in the MTS4EA, which allows the two fields to be viewed independently, either one above the other (top field above bottom field) or either field with the field lines repeated to full frame height.

Frame rate. The rate at which to display the uncompressed frames, in frames per second.

NOTE. *The number entered in the Frame rate field can be an integer (for example, 30), or a fraction (for example, 30000/1001) or a decimal number (for example, 29.97).*

Image format dialog box. The Advanced button opens the following window:



This dialog box allows you to specify a custom image format by choosing a color model, sample depths and sub-sampling ratios.

Color model. Use this control to choose the class of color model for the image format. Options are YUV, RGB, and grayscale.

Sample depth. The sample depth controls are used to specify the number of bits per sample for each image plane. For RGB and grayscale, only one sample depth is specified. For YUV, the luma and chroma depths can be specified separately.

Width divisor. This control specifies the horizontal sub-sampling factor for YUV chroma planes. This option is only available for YUV color models. The value is the factor by which the image width is divided to find the chroma plane width.

Height divisor. This control specifies the vertical sub-sampling factor for YUV chroma planes. This option is only available for YUV color models. The value is the factor by which the image height is divided to find the chroma plane height.

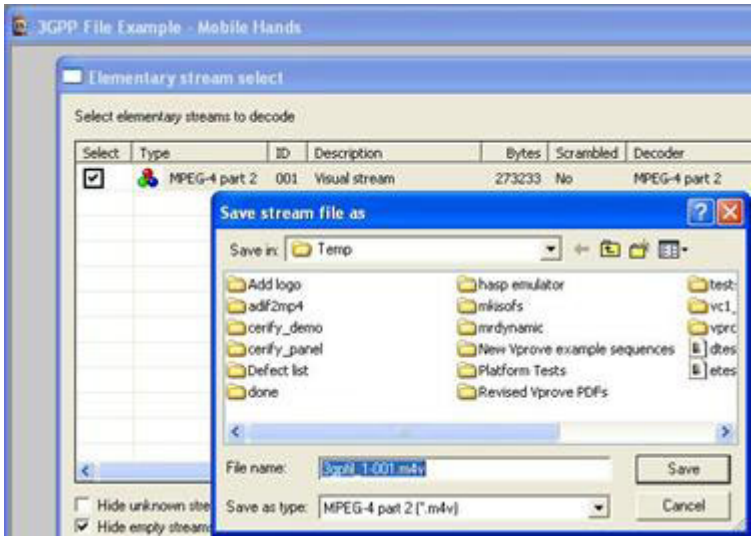
Close stream

This allows you to close the current stream but keeps the MTS4EA window open.

Save video stream as...

This allows the active file to be saved to a particular file name.

This is typically used to save a video elementary stream, when this has been extracted from an MP4, 3GPP, MPG (MPEG-2), ASF or other file type, but was not saved at the time it was extracted. (See page 5-14, *Open stream... Ctrl+O*.)



In this example, the video elementary stream from E-City.mp4 video track 1 is being saved to an .m4v file: the -001 (for track 1) and .m4v are automatically added by MTS4EA (because MTS4EA knows this is a 3GPP file), although any file name and extension can be entered.

The video can also be saved as an uncompressed video file, in a specific format. (See page 5-25, *Opening an uncompressed video file (any file extension)*.)

NOTE. *The Save stream as feature will only save uncompressed video if the input format was also uncompressed. For compressed input, use the Decoded file output feature.*

New project

Some of the information saved in a project file includes the following:

- File name being analyzed (and track number, if relevant)
- List of open views and the screen location of these views
- Location of the tooltips and toolbars
- Errors that have been disabled
- Analysis options

Project files

The project file has a file extension of .vpp. Double-clicking on a VPP file in Windows explorer will open the MTS4EA with the project file concerned.

The project file is actually a human-readable file in XML format. It is strongly recommended that it be not manually edited, because doing so may prevent the MTS4EA from opening a project file correctly.

Project default file

MTS4EA has a default project file called default.vpp. This is used as the template when New project is selected - all the settings in default.vpp are copied into a new project.

Open project...

This allows you to open a previously saved project file.

(See page 5-31, *New project*.)

Save project...

This saves the current project to the current project file name.

(See page 5-31, *New project*.)

Save project as...

This allows you to save the current project to a new file name.

(See page 5-31, *New project*.)

Close project

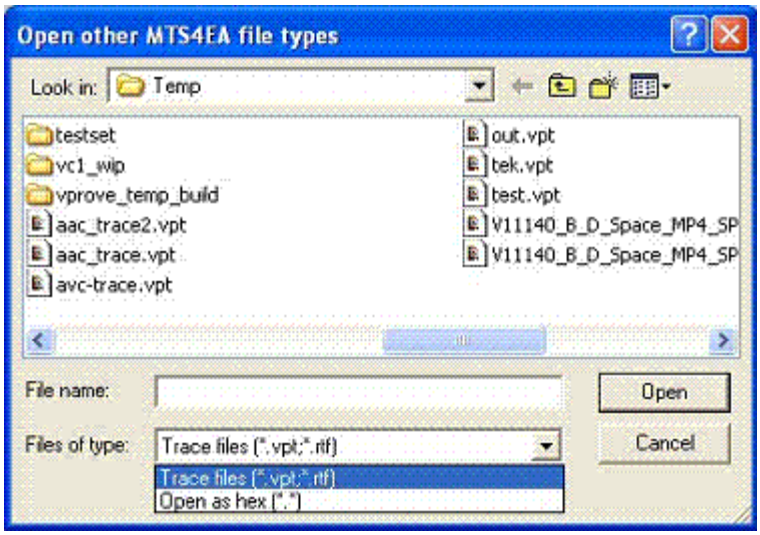
This closes the current project file.

(See page 5-31, *New project*.)

Open other...

This allows you to browse the file system to find and open other file types, such as:

- The MTS4EA trace files, with the file extension .vpt or .rtf
- Any other file, to open in the HexView



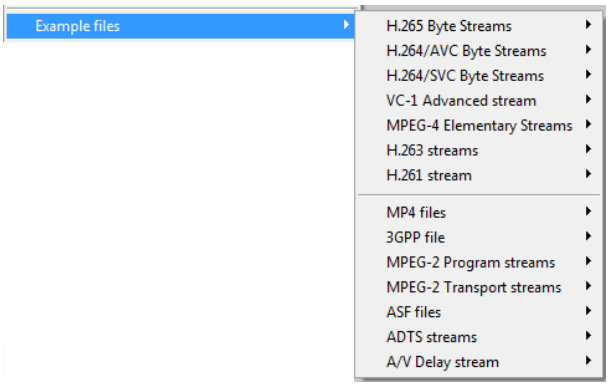
NOTE. *Opening of .rtf trace files is included for backward compatibility with older trace files generated by earlier versions of MTS4EA. From version 2.0 onwards, all trace files generated by the MTS4EA will be .vpt files. Currently, .vpt files contain only ASCII text, future versions are expected to contain additional data.*

Preferences

The Preferences dialog box allows you to select the audio playback device. Your selection will depend on the your PC setup.

Example files

Below this option are various example files that can be played/analyzed in the MTS4EA:



These are provided to give examples of various compressed files to experiment with and to compare with your own compressed files. Some of these are without error; others have known errors and generate appropriate Warning/Error messages (both as popup alerts and in the Trace files).

H.265/HEVC Byte Streams

These are H.265/HEVC compressed video Byte Streams. The following example files are provided: (See page 4-1, *Compression standards and file types*.)

Name	H.265 Profile/Level	Warnings/Errors	Parameters
Forbidden city	HEVC Main Profile / Main Tier / Level 6	None	CTU size: 64x64 Resolution: 1920 ×1088 Total number of frames: 60 Total number of slices/frame: 6 Total number of tiles/frame: 16 QP variation: Every frame has different QP value than the previous and next frames
Grenadier guards	HEVC Main Profile / Main Tier / Level 6	None	CTU size: 32x32 Resolution: 1280 ×720 Total number of frames: 60 Total number of slices/frame: 1 Total number of tiles/frame: 9 QP variation: Every frame has different QP value than the previous and next frames
Red portico	HEVC Main Profile / Main Tier / Level 2.1	Alerts: ■ Picture Size in Samples exceeds Level Limit ■ Incorrect number of Tile Rows ■ Incorrect number of Tile Columns	CTU size: 64x64 Resolution: 1280 ×720 Total number of frames: 60 Total number of slices/frame: 1 Total number of tiles/frame: 9 QP variation: Every frame has different QP value than the previous and next frames

H.264/AVC Byte Streams

These are H.264/AVC compressed video Byte Streams. Six example files are provided: (See page 4-1, *Compression standards and file types*.)

Name	H.264 Profile/Level	Warnings/Errors
Neon Night	Baseline/2	- none -
Canary Wharf	Extended/3	Use of an invalid (uninitialized) Picture Parameter Set Incorrect Direct_8x8_inference_flag
Bus Junction	Baseline/3	Fails HRD conformance error (timing violation)
Grenadier Guards	Main/3	HRD buffer overflow in many frames (starting from frame 8)
Stripey Shirts	High 10	Maximum number of motion vectors per two consecutive MacroBlocks exceeded
Times Square	High 4:4:4	Maximum number of motion vectors per two consecutive MacroBlocks exceeded

NOTE. *A YUV reference file is provided for the first ten frames of the Grenadier Guards bitstream; this YUV reference file can be used for fidelity analysis and visual difference display.*

The filename is automatically filled in for the YUV reference file, but the frame rate must be set to 25.

H.264/SVC Byte Stream

This is a H.264/SVC Scalable Baseline Profile Elementary stream. The stream supports the following:

- Spatial scalability (two layers)
- Coarse Grained SNR Scalability (two layers)
- Temporal Scalability (three layers)

One example file is provided: (See page 4-1, *Compression standards and file types*.)

Name	SVC Profile/Level	Warnings/Errors
Soccer	Scalable Baseline Profile/Level 3	- none -

VC-1 Advanced Stream

This is a VC-1 format Advanced Profile compressed video Elementary Stream. (See page 4-1, *Compression standards and file types*.) Advanced Profile of VC-1 is the only profile that can be a standalone Elementary Stream; the Simple and Main Profiles can only exist in an ASF file or in an .rcv file. One example file is provided:

Name	VC-1 Profile	Warnings/Errors
Central Park	Advanced	Invalid value (reserved) and invalid VLC

MPEG-4 Elementary Streams

These are MPEG-4 compressed video Elementary Streams. Six example files are provided: (See page 4-1, *Compression standards and file types*.)

Name	MPEG-4 Profile	Warnings/Errors
Woman Drinking	Main	- none -
Train in Station	Main	- none -
Space	Simple	modulo time base Method 1 quant used
Man Walking	Advanced Simple/Level 0 (with B-VOPs, Method 1 quant)	VBV overflow
Synthetic	Advanced Simple/Level 2 (with B-VOPs, Method 1 quant, Quarter Sample)	VCV overflow
Window Car	Advanced Simple/Level 2 (with B-VOPs, Method 1 quant, GMC with 3 warping points, affine model)	stuffing bits VCV overflow

NOTE. *A YUV reference file is provided for the whole Man Walking bitstream; this YUV reference file can be used for fidelity analysis and visual difference display.*

The filename is automatically filled in for the YUV reference file, but the frame rate must be set to 30.

H.263 Streams

These are H.263 compressed video files. Three example files are provided: (See page 4-1, *Compression standards and file types*.)

Name	Standard/Annex	Warnings/Errors
Rally (250k)	H.263 baseline	■ PSUPP is sent when file is H.263 baseline (non-critical warning but PSUPP field values only used in Annex L) ■ invalid variable length code for MCBPC in the last frame (also causes Out of sync error and other consequential errors)
Rally (2M)	H.263+ Annex T	PSUPP is sent but Annex L not used (non-critical warning but PSUPP field values only used in Annex L)
Person Track	H.263+ Annexes D, F, I, J, S, T	- none -

MP4 Files

These are MPEG-4 .mp4 container files, containing compressed video and other data. Six example files are provided: (See page 4-1, *Compression standards and file types*.)

Name	Standard/Annex	Warnings/Errors
Packet Woman	MPEG-4 Simple Profile/Level 1	Level is set at 1; max. frame size in this Level is 176x144 pixels, but the video is 352x288. VCV overflow and VBV underflow
Piccadilly Circus	MPEG-4 Simple Profile/Level 2	VCV overflow and VBV underflow
Beijing	MPEG-4 Simple Profile/Level 5 (video); AAC LC (audio)	VBV overflow
Las Vegas	H.264/AVC Baseline Profile/Level 1.2	HRD information missing: no pic_timing SEI message
Live Band	MPEG-4 Simple Profile/Level 1 (video); AAC HE (audio)	Error in section data causes the calculated number of scalefactor bands to exceed the maximum allowed value
Stage Interview	MPEG-4 Simple Profile/Level 1 (video); AAC Main (audio)	None

3GPP File

This is a 3GPP .3gp container file, containing compressed video and other data. One example file is provided: (See page 4-1, *Compression standards and file types*.)

Name	Standard	Warnings/Errors
Mobile Hands	MPEG-4 Simple Profile/Level 1	Reserved value for Profile and level indication (set to 0)

MPEG-2 Program Streams

Three MPEG-2 .mpg Program Streams are provided, containing compressed video and other data (as described in *Compression Standards and File Types* section). (See page 4-1.) These examples contain audio streams, but they are all silent.

Name	Standard	Warnings/Errors
Bus Junction	MPEG-2 Main Profile/Main Level	- none -
Person Track	MPEG-2 Main Profile/Main Level	Bad slice order Invalid VLC for dct_differential
Grenadier Guards	MPEG-2 Main Profile/Main Level	Numerator and denominator of frame_rate_extension are both set to 1 but, when equal, they must be set to 0: this also generates the errors of invalid values for frame_rate_extension_d and frame_rate_extension_n

NOTE. *A YUV reference file is provided for the first ten frames of the Grenadier Guards bitstream; this YUV reference file can be used for fidelity analysis and visual difference display.*

MPEG-2 Transport Streams

Two MPEG-2 .m2t Program Streams are provided, containing compressed video and other data. Four example files are provided: (See page 4-1, *Compression standards and file types*.)

Name	Standard	Warnings/Errors
Golden Gate	H.264/AVC Main Profile	Use of undefined disable_deblocking_filter_idc = 3
Mangroves	MPEG-2 Main Profile/Main Level	- none -
Captain Bob	MPEG-2 Main Profile/Main Level (video); MPEG-1 Audio Layer II (audio)	- none -

ASF files

Two Microsoft ASF files are provided, containing compressed video and other data. (See page 4-1, *Compression standards and file types*.)

Name	Standard	Warnings/Errors
Beach Girl	VC-1 Simple Profile	- none -
Great Wall	VC-1 Simple Profile	- none -

ADTS streams

Name	Standard	Warnings/Errors
Dance Track	AAC LC	Error in ADTS header specifies an incorrect syncword

A/V Delay stream

Name	Standard	Warnings/Errors
Clapperboard [1]	MPEG-2 Main Profile / Main Level (video) Dolby AC-3 (audio)	

1 The Clapperboard file uses parts of the file archiving utility "7-Zip". "7-Zip is licensed under the GNU LPGL (Lesser General Project License). Visit www.7-zip.org.

YUV source files for example fidelity analysis

YUV source files used for encoding have been provided for example fidelity analysis, with three of the example files provided:

Standard	Stream name	YUV filename*
H.264/AVC (Main Profile/Level 3)	Grenadier Guards	guards_yuv (frames 1-9 only: 25 frames/sec)
MPEG-2 (Main Profile/Main Level)	Grenadier Guards	guards_yuv (frames 1-9 only: 25 frames/sec)
MPEG-4 (Advanced Simple Profile/Level 0)	Man Walking	man_walking_yuv (30 frames/sec)

NOTE. *The YUV filename is automatically filled in by MTS4EA - there is no need to enter the filename.*

For the Grenadier Guards examples, the YUV of only the first nine frames is provided, because the YUV files are so large.

Export A/V delay streams…

The A/V Delay measurement feature provides a complete solution for measuring audio/video synchronization delay in an encoding system. (See page 5-254, *Audio/Video delay*.) The Export A/V Delay Streams menu option allows the supplied test streams to be exported in a variety of formats for use with encoding systems.

You can use the standards Windows controls (Ctrl and Shift) to select multiple output formats for a selected stream.

NOTE. *Some of the export formats can produce very large files. Ensure that you have enough disk space to accommodate the output.*

NOTE. *The Export A/V Delay Streams feature uses parts of the file archiving utility 7-Zip. 7-Zip is licensed under the GNU LPGL (Lesser General Project License). For more information, refer to the 7-Zip Web site (www.7-zip.org).*

Both video and audio streams can be exported. You need to be careful to match the correct audio with each video file because the timings of the audio signal vary slightly depending upon the frame rate of the video that it is to be paired with.

The supplied video formats are:

- NTSC: 704x480 interlaced at 29.97… fps
- PAL: 720x576 interlaced at 25 fps
- 720p: 1280x720 progressive at 59.94… fps
- 1080i: 1920x1080 interlaced at 29.97… fps

Each video format is supplied encoded in four ways:

- H.264/AVC: High profile
- MPEG-2: Main profile
- VC-1: Advanced profile
- Uncompressed: 4:2:0 YUV, 8 bits per sample. (See page 5-25, *YUV format of 8 bits per sample 4:2:0*.)

The supplied audio formats are all 16-bit sampled at 48kHz:

- '25fps': to be used with video at 25 fps. This should be used with the PAL video streams.
- '29.97fps': To be used with video at 29.97… fps. This should be used with the NTSC and 1080i video streams.
- '59.94fps': To be used with video at 59.94… fps. This should be used with the 720p video streams.

Each audio format is supplied encoded in four ways:

- Dolby AC-3
- AAC
- MPEG-1 Audio Layer II
- Uncompressed: PCM 16 bits per sample at 48kHz.

Recent File

The program retains a list of the eight most recently used files, numbered 1 to 8. The recent files can be selected by highlighting them in turn.

Where a particular file could be opened in more than one view - for example, a video bitstream could be opened as a video file (in the video window) or opened in the HexView - the file is stored on the Recent File list with the last view type appended to the end:

filename.m4v [hex] (this one is opened in the HexViewer)

filename.m4v (this one is opened as a video file)

filename.adts (this one is opened as an audio file)

filename.yuv [yuv] (this one is opened as YUV video)

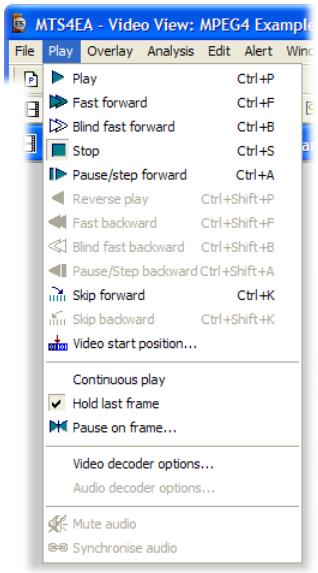
The MTS4EA then uses this information to determine in which window the file should be reopened.

Exit

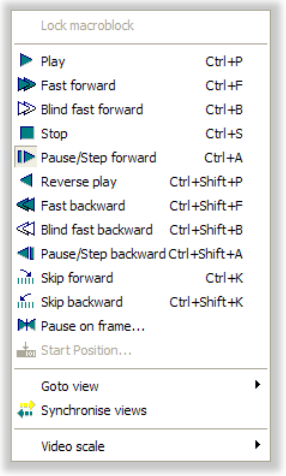
This option exits from the MTS4EA.

Play menu

This menu controls the playing of the compressed file:



In addition to the menu selection, there is a right-click popup Play menu:

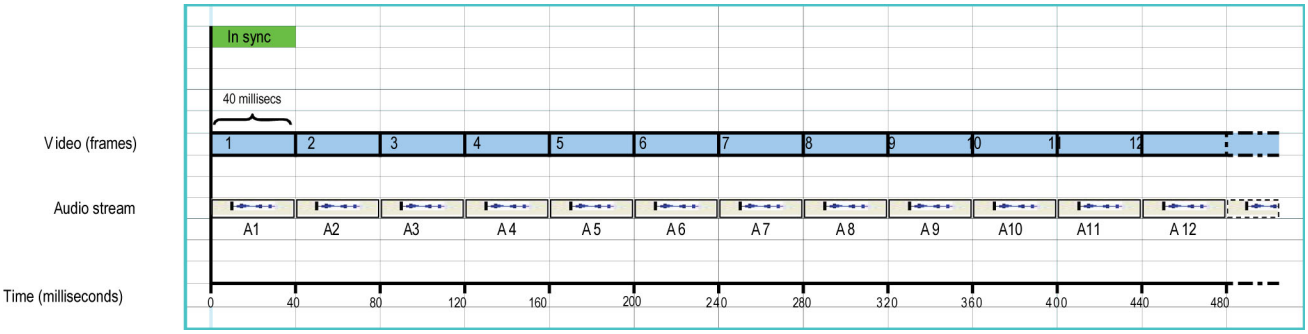


The functions of each of these Play menu items are explained on the following pages. (See page 5-8, *Synchronized views/navigating the views*.)

Audio sounds broken up

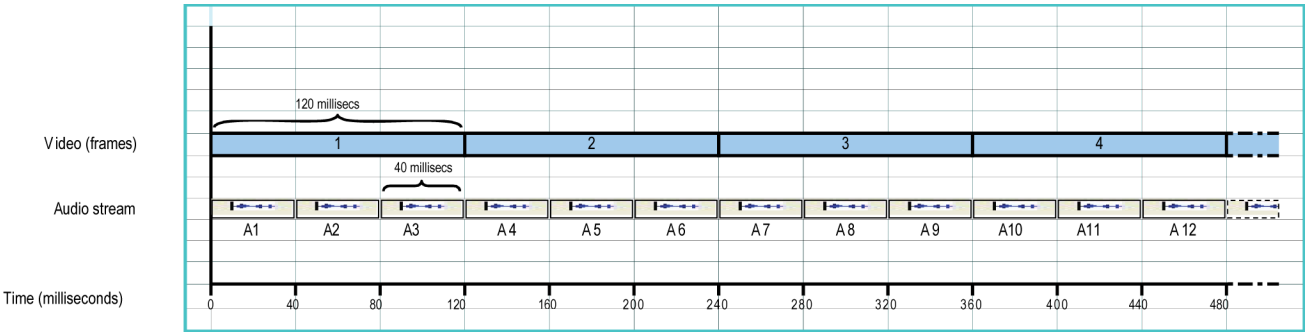
When a file containing both audio and video streams is played and the Audio synchronize function is engaged, if the PC is not sufficiently powerful, synchronization may not be possible and the resultant audio will be broken. The figures below describe, in simplified terms, how audio and video streams interrelate when decoded in the MTS4EA.

When the Synchronize audio icon is depressed, and the requested synchronization is successful, the green display appears in the Status bar and the streams run concurrently in real-time.



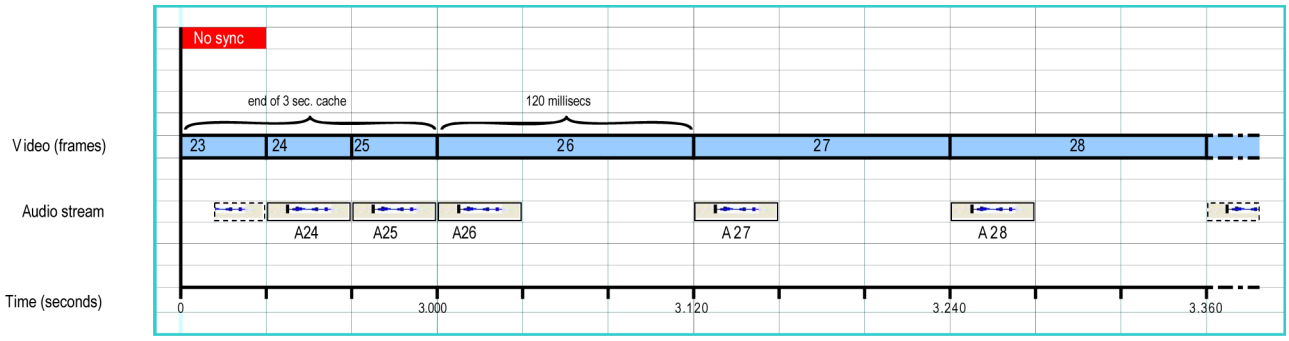
The illustration above shows video being decoded at a display rate of one frame every 40/1000 seconds. Audio plays, unbroken, at the same rate as the video.

When synchronization is not requested, the Synchronize audio icon is not depressed, and audio and video will decode independently.



The illustration above shows video playing more slowly, at a display rate of one frame every 120/1000 seconds. The audio plays at its own decode rate, outstripping the video.

When the Synchronize audio icon is depressed, and the requested synchronization is unsuccessful, the red display appears in the Status bar and the video decodes at a rate determined by available processor power, while the audio is broken.



The illustration above shows an unsuccessful attempt to synchronize audio and video: the cached data plays in sync, thereafter, limited processor power causes the audio to fragment (usually more unevenly than suggested in the above illustration) because it attempts to keep pace with the video by means of timestamps.

Audio/video synchronization

MTS4EA will decode and analyze audio and video streams, but the audio will only be played and audible when the stream is in forward play mode.

Icon	Function in video mode	Function in audio mode	Function in synchronous mode		Function in asynchronous mode	
			Video	Audio	Video	Audio
	Plays to end or specified frame or until paused.	Plays to end.	Attempts to play video and audio in synchrony. Whenever play is resumed, audio picks up at current location.		Audio and video play independently; no common timing enforced.	
	Plays quickly in forward direction.		Video plays quickly in forward direction.	Audio muted.	Video plays quickly in forward direction.	Audio muted.
	Decodes file as quickly as possible without display.		Decodes video file as quickly as possible without display.	Audio muted.	Decodes video file as quickly as possible without display.	Audio muted.
	Stops.	Stops.	Stops video and audio at once.		Video stops.	Audio muted.

Icon	Function in video mode	Function in audio mode	Function in synchronous mode		Function in asynchronous mode	
			Video	Audio	Video	Audio
	Pressed once in Play mode, pauses. Pressed again, or at top of file, steps forward 1 frame.	Pause.	Pressed once in Play mode, pauses. Pressed again, or at top of file, steps forward 1 frame.	Audio muted.	Pressed once in Play mode, pauses. Pressed again, or at top of file, steps forward 1 frame.	Audio muted.
	Plays file backwards, at normal speed.		Plays video file backwards, at normal speed.	Audio muted.	Plays video file backwards, at normal speed.	Audio muted.
	Plays file quickly backwards.		Plays video file quickly backwards.	Audio muted.	Plays video file quickly backwards.	Audio muted.
	Plays file as quickly as possible backwards without display.		Plays video file as quickly as possible backwards without display.	Audio muted.	Plays video file as quickly as possible backwards without display.	Audio muted.
	Pressed once in Play mode, pauses. Pressed again, steps back 1 frame.		Pressed once in Play mode, pauses. Pressed again, steps back 1 frame.	Audio muted.	Pressed once in Play mode, pauses. Pressed again, steps back 1 frame.	Audio muted.
	Skips on to specified frame/type/time/number of frames.		Skips on to specified frame/type/time/number of frames.		Audio and video play independently; no common timing enforced.	
	Skips back to specified frame/type/time/number of frames.		Skips back to specified frame/type/time/number of frames.		Audio and video play independently; no common timing enforced.	
	Pauses when specified video frame is reached.		Pauses when specified video frame is reached.	Audio muted.	Pauses on specified frame.	Audio muted.

Play Ctrl+P

When selected, this plays the compressed stream, video or audio, until it reaches the end of the data. If, when playing a compressed video stream, a frame number has been set in Pause on frame the video will pause at this frame number.

If a file containing both audio and video streams is being played, and the streams are not synchronized, the video and audio will begin to play and continue to play independently. So, if a compressed video file is playing slowly due to the available processing power of the PC, it is likely that the video will lag behind the accompanying audio stream. If audio synchronization is selected, the rate at which the audio data is decoded adjusts and the audio stream keeps pace with the video, by means of timestamps. (See page 5-62, *Synchronize audio*.)

NOTE. *Depending upon the speed of your PC, when playing compressed video, the standard in use and the size of the video frames being viewed, there can be a significant delay after pressing the Play or Pause/Step forward button, during which the message Buffering <nnn> (nnn is a number) is displayed near the top of the video window. This indicates that the video decoder is loading the required number of video frames into its internal buffer. This message disappears when the video is ready to display. (This is most likely to be seen with the H.264/AVC, H.264/SVC, and H.265/HEVC video standards, where up to 16 frames are loaded into the decoder buffer before displaying the first frame.)*

At the end of the data, the gray background screen is displayed unless:

- Hold last frame is selected, in which case the last frame stays visible
- Hold first frame is selected, in which case the first frame is re-displayed

Fast forward Ctrl+F

When this option is selected, the compressed file will be played in Fast forward mode, played more quickly than normal. The speed at which this is replayed depends on the speed of the computer you are using.

This command is unavailable when playing audio files.

If a file containing both audio and video streams is being played, regardless of synchronization, the video plays fast forward and the audio is muted.

NOTE. *When playing video forwards or backwards, the MTS4EA buffers the video and associated data. By default, the MTS4EA allocates a buffer size of 500 MB for video and 100 MB for audio. The amount of buffer required per frame of video and associated data varies considerably with the video size and the video standard. Typically, 100 MB is sufficient to buffer the video and data for 10-500 frames. However, the actual number of frames and frame data stored depend on the resolution of the video being analyzed. For example, 500 MB is required to buffer the video and data for around 10-500 H.265/HEVC frames.*

To change the buffer size, use the Play menu, Decoder options, General tab. (See page 5-54, *Video decoder options....*)

The video frame count is given in a box at the top of the image, to indicate how far through the video sequence the currently displayed frame is (the frame count is also displayed at the bottom right of MTS4EA window in the status bar).

The frame count is displayed in white text if the option in the Overlay menu is currently set to White, or in black text if this is set to Black (toolbar icon or Ctrl+W).

The fast forward may be stopped at any time by clicking the Pause/Step forward icon - (or pressing Ctrl+A).

If a frame number has been set in Pause on frame, the video will pause at this frame number.

Blind fast forward Ctrl+B

When this is selected, the compressed file is decoded as quickly as possible without displaying it.

If a file containing both audio and video streams is being played, regardless of synchronization, the video blind fast forwards and the audio is muted.

NOTE. *It can be useful to use Blind fast forward to quickly decode the file to see if there are any alerts generated, or to move to a selected Pause on frame… as quickly as possible.*

See the note regarding a possible buffering delay immediately after this is selected. (See page 5-47, *Fast forward Ctrl+F*.)

If a frame number has been set with the Pause on frame then the video is displayed until the designated frame number is reached.

The Blind fast forward is stopped in the same way as Fast forward.

Stop Ctrl+S

When this option is selected, it will stop the playing of the compressed file.

The gray background screen is displayed in the video window unless:

- Hold last frame is selected, in which case the last displayed frame stays visible.
- Hold first frame is selected, in which case the first frame is redisplayed.

Pause/step forward Ctrl+A

This option can be selected from the Play menu, by pressing Ctrl+A, or by clicking the tool bar icon Pause/Step forward .

This command is available when playing audio files, but audio will only be heard when the stream is in forward play mode.

NOTE. *Repeated selection of this option will advance the displayed video one frame at a time.*

This has two functions:

- If the file is playing, selecting this option will pause the decoding of the stream and, in the case of a video file, display the most recently decoded frame
- If the file has stopped or is already paused, selecting this option causes the file to advance by one displayed frame

NOTE. *See the Note regarding a possible buffering delay immediately after this is selected.* *(See page 5-47, Fast forward Ctrl+F.)*

In H.264/AVC where SI-frames and/or SP-frames are used, when using Pause/Step forward (Ctrl+A), at the point of the switch, both the frames used for the switch are displayed, one after the other. These are not fully displayed in the video when it is viewed normally, and so are not displayed when using Play or Fast forward selections.

Reverse play Ctrl+Shift+P

Play the file backwards, at normal speed.

This command is unavailable when playing audio files. Audio will only be heard when the stream is in forward play mode.

This option can be selected from the Play menu or by pressing Ctrl+Shift+P, or by clicking the tool bar icon Reverse play - .

Fast backward Ctrl+Shift+F

Play the file backwards, as fast as possible, while still displaying it.

This command is unavailable when playing audio files. Audio will only be heard when the stream is in forward play mode.

This option can be selected from the Play menu or by pressing Ctrl+Shift+F, or by clicking the tool bar icon Fast backward - .

Blind fast backward Ctrl+Shift+B

Play the file backwards, as fast as possible, without displaying it.

This command is unavailable when playing audio files. Audio will only be heard when the stream is in forward play mode.

This option can be selected from the Play menu or by pressing Ctrl+Shift+B, or by clicking the tool bar icon Blind fast backward - .

Pause/step backward Ctrl+Shift+A

Step backwards through the video file one displayed frame at a time.

This command is unavailable when playing audio files. Audio will only be heard when the stream is in forward play mode.

(See page 5-49, *Pause/step forward Ctrl+A*.)

This option can be selected from the Play menu or by pressing Ctrl+Shift+A, or by clicking the tool bar icon Pause/Step backward - .

Skip forward Ctrl+K

This will skip the video to the next frame type/time/number of frames indicated in the drop-down box on the icon toolbar.

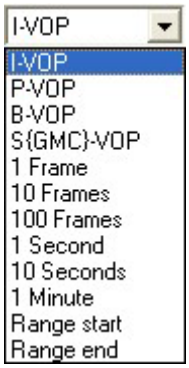
For example, if the box next to the skip forward icon has I-VOP in it (for example,) then clicking the Skip forward icon will skip to the next I-VOP.

This option can be selected from the Play menu or by pressing Ctrl+K, or by clicking the toolbar icon Pause/Step forward .

If going to the selected frame type/time/number of frames is beyond the length of the video stream, then the stream is advanced to the end.

See also Audio/Video Synchronization (See page 5-44.) and Video Decoder Options (See page 5-54.) regarding the buffer used to hold the video and associated data. When skipping forward or backward beyond the end of the buffer, the MTS4EA must reload the buffer, so some delay may be experienced.

The possible selections vary with the selected video standard; the example below is for MPEG-4 Advanced Simple Profile.



This command is unavailable when playing audio files. Audio will only be heard when the stream is in forward play mode.

Skip backward Ctrl+Shift+K

(Ctrl+Shift+K) This is the same as Skip forward but in the reverse direction. (See page 5-51.)

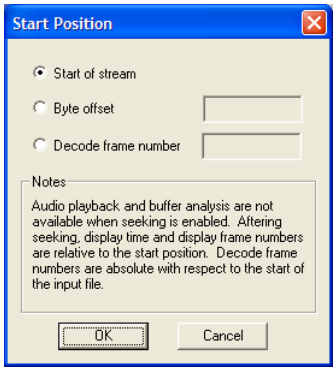
This option can be selected from the Play menu, by pressing Shift+Ctrl+K, or by clicking the tool bar icon Pause/Step backward - .

This command is unavailable when playing audio files. Audio will only be heard when the stream is in forward play mode.

Video start position...

NOTE. *This option is not available for H.265/HEVC streams.*

In the Start Position dialog box, you can set either a byte offset or decode frame number as a starting position for decoding.



The byte offset is relative to the start of the file.

Continuous play

This command is unavailable when playing audio files.

When this option is selected and Play/Fast forward/Blind fast forward/ Reverse play/Fast backward/Blind fast backward is clicked, then the file will be continually played in a loop. That is, the file is decoded until its end, at which point decoding will start again at the beginning of the file, until Stop or Pause is used.

Audio will only be heard when the stream is in forward play mode.

Hold last frame

When this option is selected and the video/audio file is played to the end, the last decoded frame will be displayed. If Hold Last Frame is not selected, the gray background video screen will be displayed when the video is not playing.

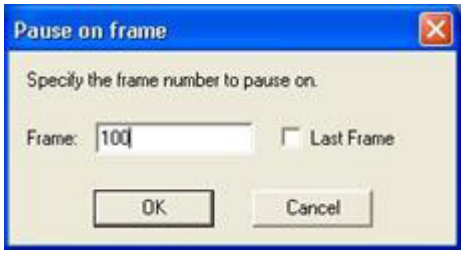
Hold Last Frame is selected as on by default when MTS4EA starts.

This command is not relevant for audio files.

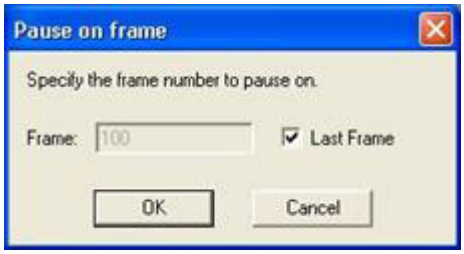
Pause on frame...

This command is only applicable to the playing of video files.

This option allows you to select a frame number to pause on when the video is played. Having selected a frame to pause on, and clicking Play or Fast forward, or Blind fast forward, the video will play until the selected frame number and then pause.



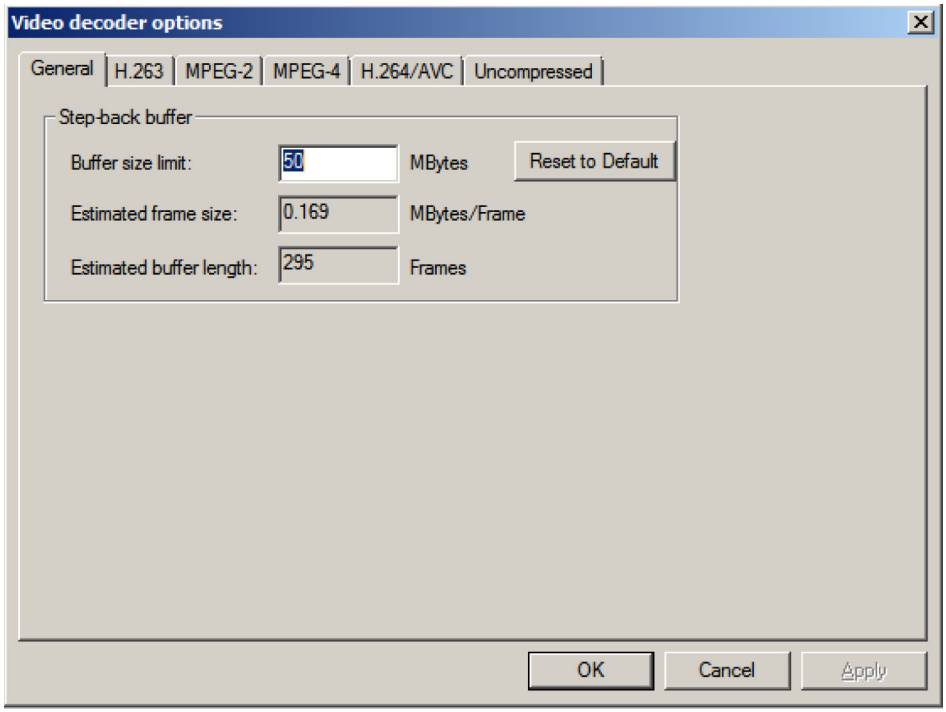
The last frame can be selected by using the Last Frame box. When this is done, the video pauses on the last frame. The previously entered frame number (if there is one) is still stored, and can be enabled again by clearing the Last Frame check box:



Video decoder options...

The Video Decoder dialog box has various tabs that set the video decoder options for each standard. When there are no configurable options for a particular video standard, there is no tab on the dialog.

General



Step-back buffer. This buffer is used to hold the video and associated data, to make video play (forwards and backwards) occur without delay.

Once the Buffer Size limit is modified and set, the value is retained until changed again by the user. Even if the application is closed and restarted, or if another stream is opened, the setting is persistent. To reset the Buffer Size limit back to the default value, click the Reset to Default button.

NOTE. *The size of the step-back buffer is shown, with an estimation of how many frames of data can be held within this buffer size. When playing video forwards or backwards, the MTS4EA buffers the video and associated data. By default, MTS4EA allocates a buffer size of 500 MB for video and 100 MB for audio. The amount of buffer required per frame of video and associated data varies considerably with the video size and the video standard. Typically, 100 MB is sufficient to buffer the video and data for 10 to 500 frames. However, the actual number of frames and frame data stored depend on the resolution of the video being analyzed. For example, 500 MB is required to buffer the video and data for around 10-500 H.265/HEVC frames.*

The popup alerts (Warnings, Errors, and so forth) that occur when video is decoded by the MTS4EA, are not displayed if the video and associated data is in the Step-back buffer. (See page 5-266, Popup alerts in the step-back buffer.)

H.263

This has no effect and is grayed out for standards other than H.263.

NTSC/PAL option. Within the H.263 standard the NTSC standard is assumed (frame rate of 29.97 frames per second) and video is played back at this frame rate by default. The NTSC option is selected as standard.

Sometimes H.263 video in PAL format (frame rate of 25 frames per second) does not set these options. In this situation PAL video would play back at 29.97 frames per second and it would play too quickly. Selecting the PAL option on this tab will play the decoded video at 25 frames per second.

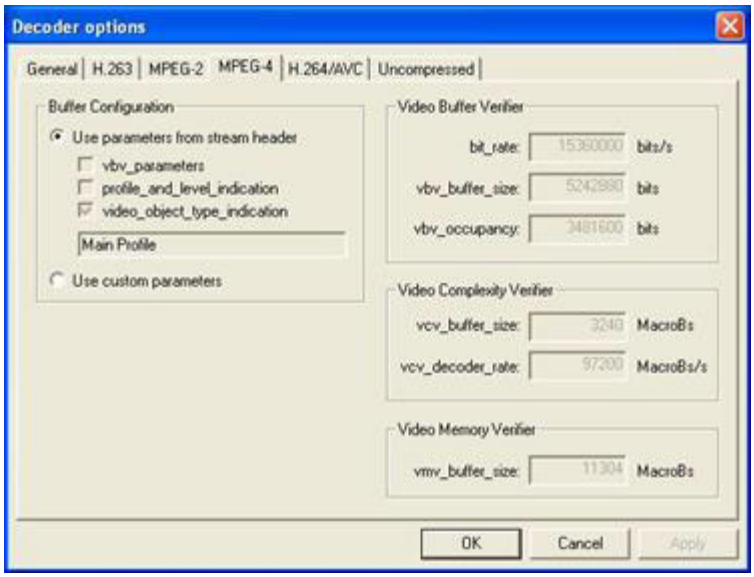
MPEG-2

This has no effect and is grayed out for standards other than MPEG-2.

This dialog box relates to the use of parameters for VBV buffer analysis in MPEG-2.

(See page 5-219, *Buffer analysis controls: MPEG-4 and MPEG-2*.) and (See page 5-220, *Use custom buffer parameters: MPEG-4 and MPEG-2*.)

MPEG-4



This has no effect and is grayed out for standards other than MPEG-4.

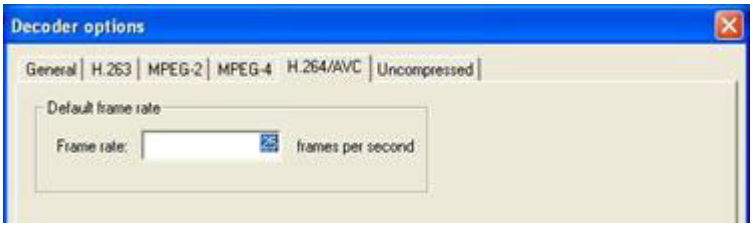
This relates to the use of parameters for buffer analysis (VBV, VCV, VMV) in MPEG-4.

(See page 5-219, *Buffer analysis controls: MPEG-4 and MPEG-2*.) and (See page 5-220, *Use custom buffer parameters: MPEG-4 and MPEG-2*.)

H.264/AVC

Many H.264/AVC byte streams do not provide the SEI (Supplemental enhancement information) data for specifying a frame rate, as this is optional in the byte stream (and where this happens, the frame play data is typically provided at the systems level for the video decoder).

Therefore the setting is provided on this option tab, so that the stream will play at the correct frame rate when the appropriate SEI data is not present.



Uncompressed video

The data is either:

- 8 bits per sample, 4:2:0, or
- More than 8 bits per sample, and/or 4:2:2 or 4:4:4 (as used by H.264/AVC High Profile/FRExt, High/10, High/4:2:2, High/4:4:4)

YUV format of 8 bits per sample 4:2:0. The YUV file output is raw YUV with no headers of any kind: this is the same format as used by the Microsoft MPEG-4 Part 2 reference encoder and as used commonly by other programs:

- No headers of any kind (no file or frame headers)
- One byte per sample
- Row raster order (top picture row first)
- Planar YUV 4:2:0 sub-sampled (4 bytes of Y data for each byte of U data and each byte of Y data)
- Y plane values are 0-255 unsigned
- U and V plane values are unsigned with a DC offset of 128

(See page 4-1, *Compression standards and file types*.)

Other uncompressed formats. The general uncompressed video file format is as follows:

- No headers of any kind (no file or frame headers)
- Concatenated planar image data
- Row raster order (top picture row first)
- Unsigned samples

For 8-bit sample depth:

- One byte per sample
- For 9-16 bit sample depth:
- Two bytes per sample
- Both little- and big-endian byte orders supported

For YUV format:

- Concatenated Y, U and V planes
- U and V planes sub-sampled as required
- Y plane samples are unsigned
- U and V plane samples are unsigned with a DC offset of 2^{-1}, where *n* is the chroma sample bit depth

For RGB format:

- Concatenated R, G and B planes.

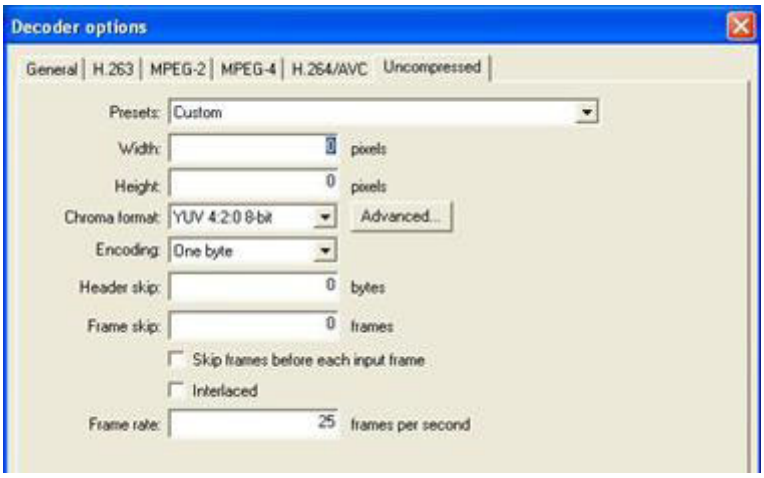
For grayscale format:

- Luma plane only

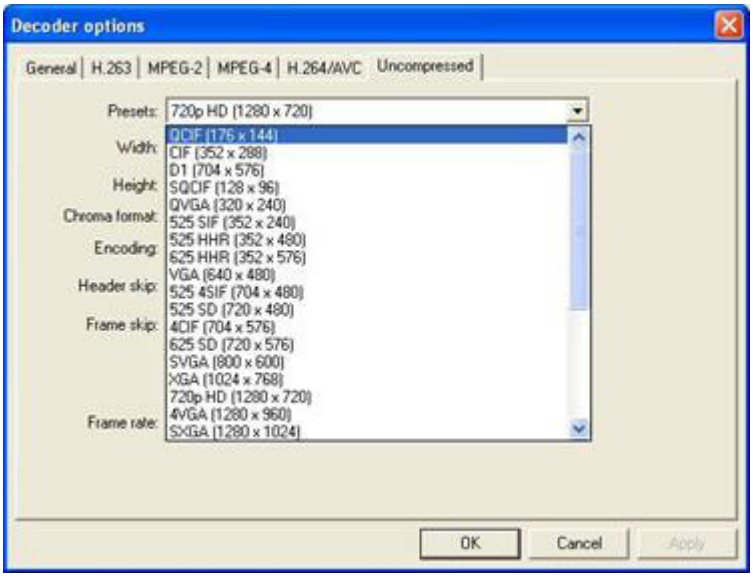
When opening an uncompressed video file, by default the MTS4EA shows the uncompressed video decoder options, to enter the required parameters.

NOTE. *If you try to open an .m4v or .264 or other compressed video file by using Open as uncompressed video, the MTS4EA will not decode the compressed video but will assume it is in uncompressed format.*

Setting the frame size. Within an uncompressed video file, there is no place to indicate the frame size, frame rate, and so forth. When the file is opened, the MTS4EA will display the following screen (this is a tab of the Decoder options, which can also be accessed from the Play menu, Decoder options):



Presets (width and height). If there are numbers in the filename that could indicate the frame size, then the MTS4EA will attempt to read these and offer these in the Width and Height fields shown in the previous screenshot. The correct values can be entered or selected using the drop-down list:



Chroma format. This control specifies the color model and format for the uncompressed video data. The combo provides several common combinations, including 8-bit YUV 4:2:0, 8-bit YUV 4:2:2, 8- bit YUV 4:4:4, 8-bit RGB and 8-bit grayscale. Select an entry from the list to use one of these common formats. Other formats can be specified by choosing the Custom entry in the list - this will open the Image format dialog.

Advanced button. Click this button to specify a custom chroma format - this will open the Image Format dialog. (See page 5-29, *Image format dialog box*.)

Encoding. This control specifies the encoding of image samples in the file format. The following options are available:

- One byte - this specifies that image samples are stored in one byte per sample. This format is appropriate if all image planes are 8-bits deep
- Two byte MSB first - this specifies that image samples are stored in a pair of bytes for each sample. The most significant byte occurs first in each pair (big-endian). This format is appropriate if one or more image planes are deeper than 8-bits
- Two byte LSB first - this specifies that image samples are stored in a pair of bytes for each sample. The least significant byte occurs first in each pair (little-endian). This format is appropriate if one or more image planes are deeper than 8-bits

Header skip. The number of bytes at the start of the file before the first frame; MTS4EA will skip past these bytes (ignoring them).

Frame skip and **Skip frames before each input frame**: Frame skip is the number of frames (not bytes) to skip between each frame that is viewed in the video window.

By default, these frames are skipped after each viewed frame; by selecting the Skip frames box before each input frame, the number of frames are skipped before each viewed frame.

Interlaced: The uncompressed video file has interlaced data in the format of complete frames, with both fields within one frame, top-field first, each field on alternate lines.

Enabling this check box switches on the Interlace toolbar in the MTS4EA which allows the two fields to be viewed independently, either one above the other (top field above bottom field) or either field with the field lines repeated to full frame height.

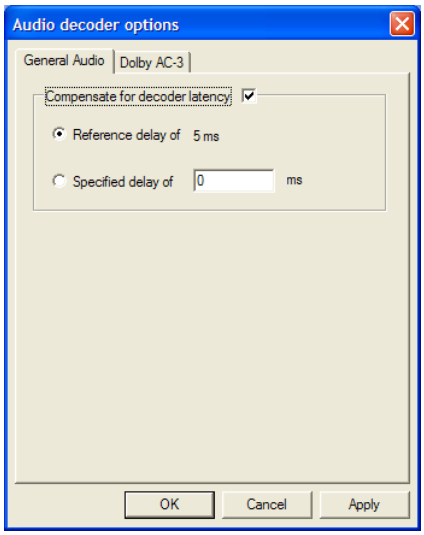
Frame rate. The rate at which to display the uncompressed frames, in frames per second.

NOTE. *The number entered in the Frame rate field can be an integer (for example, 30), or a fraction (for example, 30000/1001) or a decimal number (for example, 29.97).*

Audio decoder options…

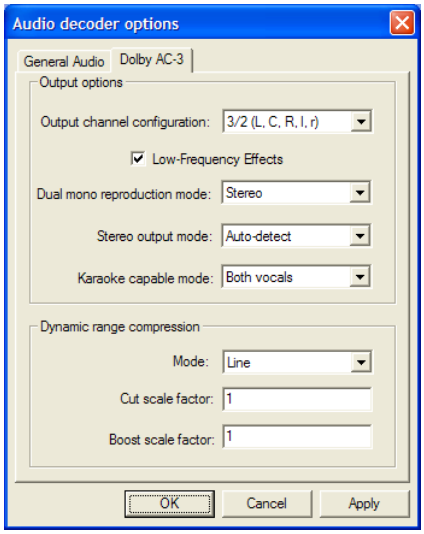
The Audio Decoder option is enabled only when audio is present in a stream.

General Audio. Where a decoder has inherent latency, the MTS4EA can compensate automatically.



With Compensate for decoder latency selected, a reference delay appropriate for the decoder in use is automatically available. If you want to specify a different delay, enter the value in the Specified delay field and select the option.

Dolby AC-3. The Audio Decoder Dolby AC-3 tab allows you to set up output options and dynamic range compression values.



The fields correspond to those used in the Dolby AC-3 development kit. Changes will take effect the next time the stream is played (that is, decoded).

Mute audio

Clicking on this button turns the sound on/off. The audio stream will continue to decode, even with sound disabled.

Synchronize audio

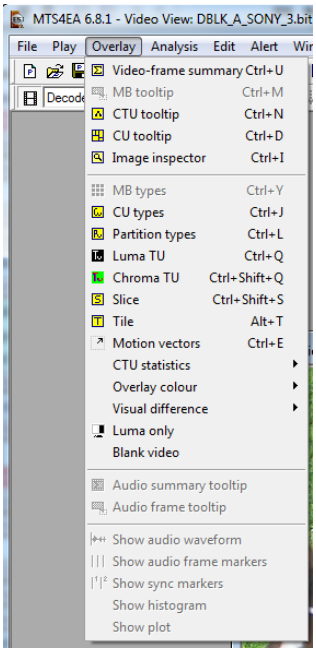
Without synchronization enabled, audio and video streams will play independently, at rates determined by the available processing power of your PC, with audio usually playing more quickly than video. When the Synchronize audio icon is depressed, audio data will be played as the video frame to which it pertains is played. (See page 5-44, *Audio/video synchronization*.)

To show whether audio and video streams are successfully synchronized or not, the following displays appear in the Status bar:

This menu item and icon are only available when a file containing both audio and video is loaded.

Overlay menu

This menu controls the display of overlaid real-time information and statistical information that has been captured over a period of time.



This menu returns information on video streams only. The audio options for this menu are described. (See page 5-132.)

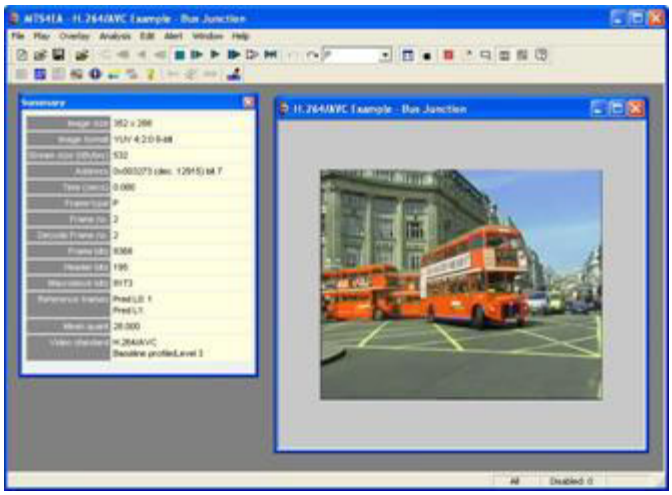
Summary & MacroBlock tooltips manipulation

The Summary tooltip and the MacroBlock tooltip windows move and dock in the same way.

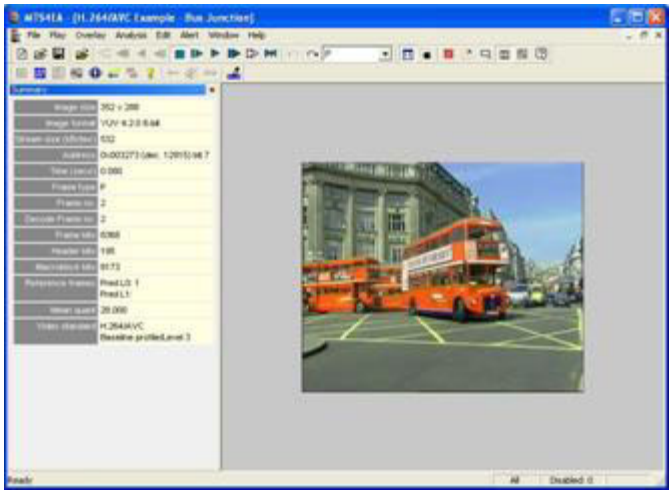
NOTE. *When the tooltips are dragged near the edge of the window, they automatically dock. To force undocking, press and hold the Ctrl key while dragging with the mouse.*

The tooltips bring up independent windows that can be:

- Moved around within the main window and put next to a video window:



- Docked to one of the edges of the main window:



When the Summary/MacroBlock tooltip is docked (as in the preceding figure):

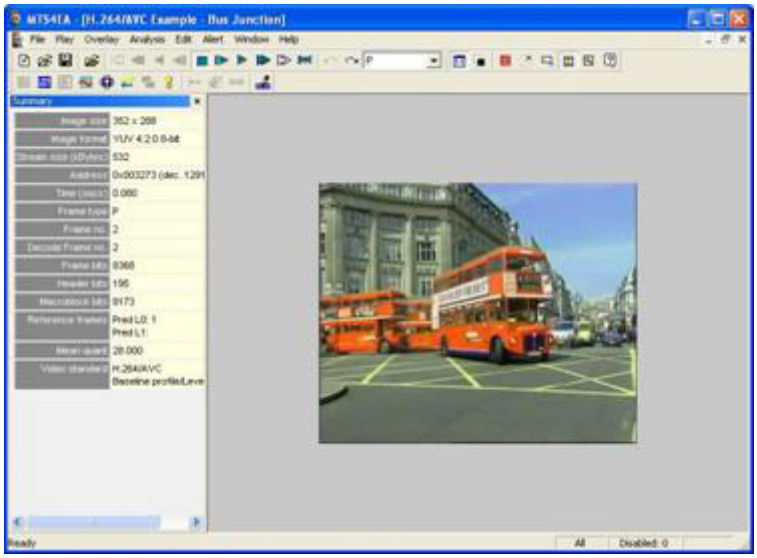
- And the window is maximized, the Summary/ MacroBlock tooltip remains visible and the other window is only maximized into the space available
- The title bar of the Summary/MacroBlock tooltip changes to a pale color to indicate it is docked (and the title text is no longer bold)

Docking/undocking

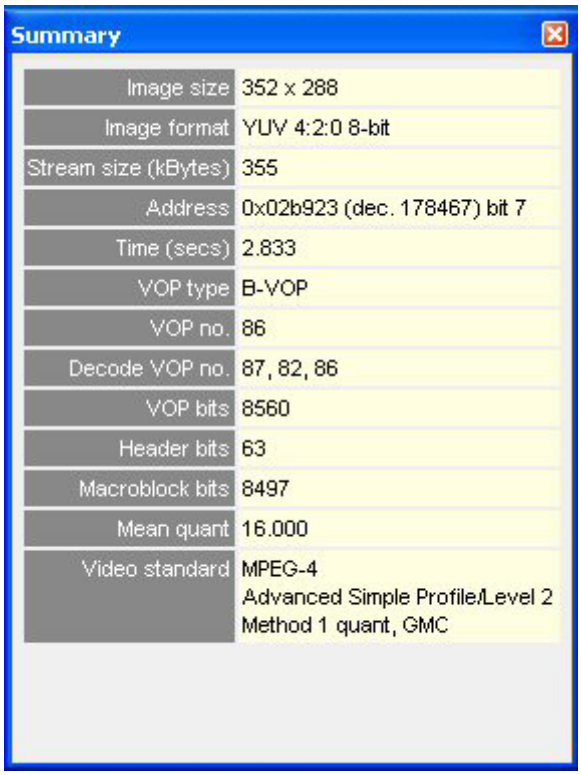
- To dock the Summary/MacroBlock tooltip: drag it to one of the edges of the main window (top, left, right, bottom).
- To undock, hold the keyboard control (Ctrl) key and click on the title bar.

Scrolling/scroll bars

When the available area is too small for the whole tooltip to be seen, scroll bars are automatically displayed:



Video-frame summary tooltip Ctrl+U



When this is switched on, a window is displayed that provides summary information about the frame being displayed or the sequence as a whole. The window can be dragged to any position within the MTS4EA main window.

NOTE. *The titles in the Summary tooltip and the displayed information vary with the selected video standard and the options that have been used in the specific video file.*

If the sequence has not been decoded to the end, then the summary up to that point is provided. The information given is described in detail below.

Tref [H.263 only] Tref is the time to the current frame from the start of the sequence, measured in increments of the base frame time for NTSC or PAL video (~33 milliseconds or 40 milliseconds respectively). Tref has a maximum value of 255; when it reaches this value it wraps back to zero.

Skip [H.263 only]

Skip is the difference in Tref for this frame, compared to Tref for the previous frame.

NOTE. *Skip and Tref are only displayed for H.263.*

For NTSC video at roughly 30 frames per second (actually at 30,000/1,001 Hz), frames are displayed every 0.033 seconds (approximately). However, many compressed video sequences do not have 30 frames per second of data (although they must of course be displayed at the correct rate, as if they did have 30 frames per second).

For example, if a compressed video sequence had 15 frames per second, the decoder would skip every other frame; Skip would equal two in this case. (And, for example, if NTSC video were displayed at 10 frames per second then Skip would equal three.)

For a compressed video sequence at 15 frames per second, it would mean that the displayed frame number would increase by 15 each second.

So, continuing the example of video at 15 frames per second, assuming that there are two Skips before the first frame, the values for the first few frames would be as follows:

Time approx. (secs)	Tref (Temporal reference)	Skip (increment)	Frame number (count of frames decoded and displayed
0.067	2	2	1
0.133	4	2	2
0.200	6	2	3
0.267	8	2	4

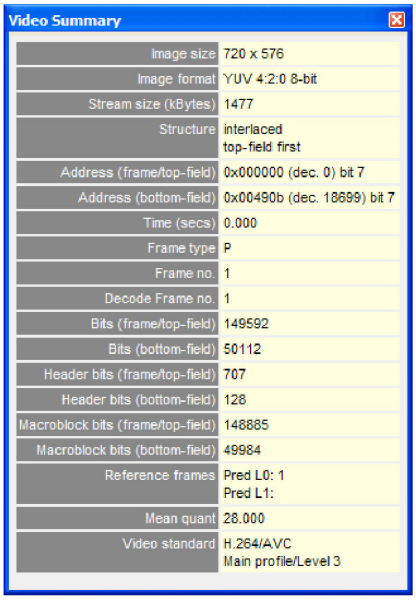
There are many variations to this; often more than one frame is skipped.

NOTE. *Also, within MTS4EA, for H.263 video, Tref is limited to 8 bits, a maximum value of 255, as this is how Tref is defined within the H.263 standards (actually TR).*

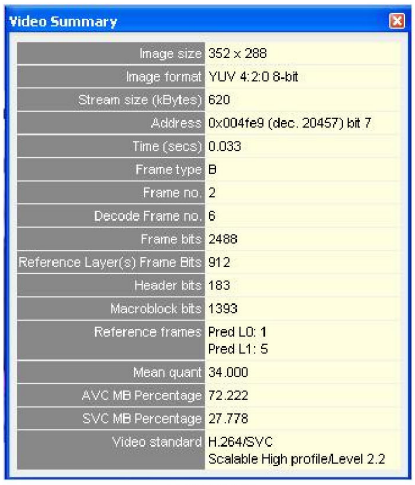
Some of the Trace outputs of the MTS4EA keep track of the total number of frames skipped - this is called Cumul. skip (short for Cumulative skip). In the last row of the table in the preceding example Cumul. skip would be eight.

Summary tooltip varies by video standard

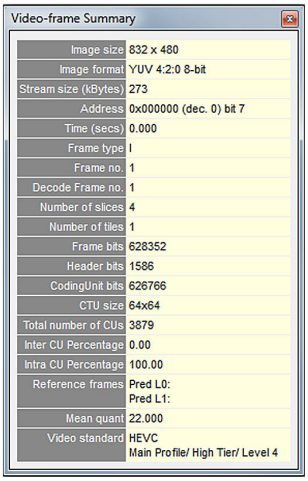
The information provided by the Summary tooltip varies depending upon the video standard that is being analyzed, and the specific options in the selected video file. Examples for different streams are shown below:



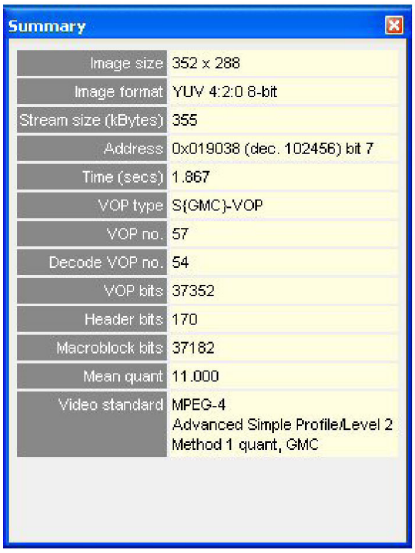
H.264/AVC



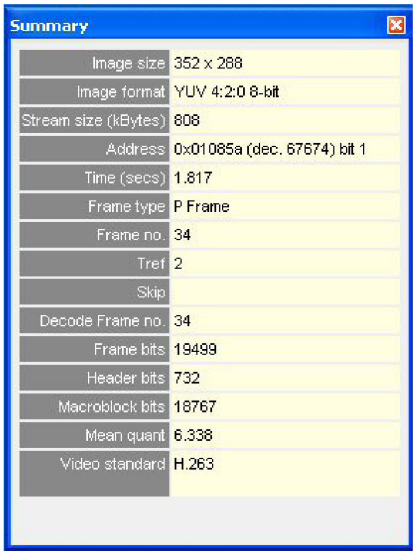
H.264/SVC



H.265/HEVC



MPEG-4 Advanced Simple Profile



H.263

NOTE. *The information displayed at the end of video sequence is different. (See page 5-75, Summary tooltip at the end of a video sequence.)*

Image size [all standards]. Picture size in pixels, Width x Height (pixel Columns x Rows).

For H.264/AVC where the cropping rectangle feature has been used, the full image size is given and displayed in the video window, but the cropped image size is also given in this field of the tooltip in the form:

Image size	352x288 (cropped: 200x100)

Image format [all standards]. Format of selected video file.

Stream size (kBytes) [all standards]. Size of the video file in KB.

NOTE. *For container files such as MP4, 3GPP and MPG (MPEG-2), the file size given is the size of the video track that has been selected, not the size of the whole MP4/3GPP/MPG file.*

Structure [H.264/AVC, VC-1 Adv. Profile and MPEG-4 ASP]. For H.264/AVC byte streams and MPEG-4 Advanced Simple Profile, where interlace could be used, the structure of the displayed frame is one of the following:

- Progressive
- Interlaced

Address or Address (frame/top-field) and Address (bottom-field) [interlace]. The start address of the current video frame/VOP within the video file.

The H.264/AVC byte streams, two address fields are given to display the address of both fields in interlaced streams. If a stream is not interlaced, then the Address (bottom field) has no data.

NOTE. *The first byte in the video file is byte 0; the first bit is bit 7 of byte 0.*

The byte address is displayed in hexadecimal first, for example, 0x002681, then in decimal in brackets, for example, (dec. 9857) and finally the bit location within the byte, where bit 7 is the most significant bit (occurs first in the stream).

Time (secs) [all standards]. The time in seconds from the start to that point in the sequence, calculated as if the sequence was playing normally (MTS4EA keeps track of the playing time, regardless of whether the sequence is paused at some point or played in fast forward mode).

This information is calculated from:

- [H.264/AVC]: the SEI parameters if they are present in the bitstream.

NOTE. *Many H.264/AVC bitstreams do not have the relevant SEI values (these are optional) and in this case, MTS4EA will play the video at the frame rate indicated in the H.264/AVC tab of the Decoder options (on the Play menu)*

- [VC-1]: the parameters either in the ASF file, if the video is within this, or within the VC-1 stream
- [MPEG-4]: the parameters such as `modulo_time_base`, `vop_time_increment`
- [MPEG-2]: the parameters `frame_rate_value`, `frame_rate_extension_n` and `frame_rate_extension_d`
- [H.263]: the time-stamps in the picture header. If the Temporal references [Tref] are not set or are not correctly set, this time may not be the actual playing time from the start of the sequence.

VOP type [MPEG-4] or Frame type [H.264/AVC, MPEG-2, H.263, VC-1].

NOTE. *For interlaced bitstreams, the frame/VOP type reported is that relating to the second field in a frame. This means that if the top field is first in frame/VOP 1 (and this will always be Intra-coded) and the bottom field is Inter [P] coded, then the frame will be reported as P.*

- [MPEG-4] the VOP type, I-VOP or P-VOP or B-VOP or S-GMC VOP (see the following):
  - I-VOP: indicates that this frame is Intra coded (completely coded within itself) without any interpolation from earlier or later frames. The first frame in an MPEG-4 sequence is always an I-VOP
  - P-VOP: P or Predicted VOP: this frame is Inter coded (partly coded based upon earlier frames)
  - B-VOP: *(MPEG-4 Advanced Simple Profile only)* B or Bidirectional Interpolated: this frame is calculated based upon both earlier and later frames. B-VOPs may only be interpolated based upon I-VOPs/P-VOPs (not on other B-VOPs)
  - S-GMC: *(MPEG-4 Advanced Simple Profile only)* the VOP is an S-GMC VOP which uses Global Motion Compensation (GMC)
- [H.264/AVC, VC-1] the Frame type, I-frame or P-frame or B-frame or SI-frame or SP-frame (see below):
  - I-frame: I indicates that this frame is Intra coded (completely coded within itself) without any interpolation from earlier or later frames. The first frame in an H.264/AVC sequence is always an I frame
  - P-frame: P indicates Predicted: this frame is Inter coded (partly coded based upon earlier frames)
  - B-frame: *(Extended Profile only)* B or Bidirectional Interpolated: this frame is calculated based upon both earlier and later frames
  - SI-frame: SI indicates that this frame is a Switching-Intra coded frame - there is a switch between two different streams at this point
  - SP-frame: SP indicates that this frame is a Switching-Inter coded frame - there is a switch between two different streams at this point

NOTE. *For H.264/AVC, the frame type is derived from the most complex slice type present. For example, if any of the slices in the frame are B-slices, the whole frame is reported as a B-frame. If any of the slices in the frame are P-slices, the whole frame is reported as a P-frame. (This is as suggested in the standard.)*

- [MPEG-2] the Frame type, I-frame or P-frame or B-frame (see the following):
  - I-frame: I indicates that this frame is Intra coded (completely coded within itself) without any interpolation from earlier or later frames. The first frame in an MPEG-2 Elementary Stream sequence is always an I frame
  - P-frame: P indicates Predicted: this frame is Inter coded (partly coded based upon earlier frames)
  - B-frame: B or Bidirectional Interpolated: this frame is calculated based upon both earlier and later frames
- [H.263] Frame type: the frame type, I-frame or P-frame or B-frame (see the following):
  - I-frame: I indicates that this frame is Intra coded (completely coded within itself) without any interpolation from earlier or later frames. The first frame in an H.263 sequence is always an I frame
  - P-frame: P indicates Predicted: this frame is Inter coded (partly coded based upon earlier frames)

VOP no. [MPEG-4] or Frame no. [H.264/AVC, MPEG-2, H.263, VC-1]. The number of the displayed VOP [MPEG-4]/frame [H.264/AVC, MPEG-2, H.263] in the sequence.

NOTE. *For sequences with B-VOPs/B-frames and in some other cases, the displayed frame number may be different from the decode for the VOPs/frames - see the following.*

Tref and Skip [H.263 only]: (See page 5-66, *Tref [H.263 only]*.)

Decode VOP no. [MPEG-4] or Decode Frame no. [H.264/AVC, MPEG-2, H.263, VC-1]. The data that appears in this field differs from the video standard.

For H.264/AVC, one number is displayed (the current display frame number) as each MacroBlock can be bi-directionally predicted from different frames; to find out which frames are used for prediction, the MacroBlock tooltip is used.

For MPEG-4 Advanced Simple Profile and MPEG-2, when B-VOPs/B-frames are used, the data in these VOPs/frames is calculated from data in preceding and following VOPs/frames stored in the video file.

That is, for sequences with B-VOPs the order in which the VOPs/frames are stored in the video file is different from the order in which the VOPs/frames are displayed.

In this circumstance, this field of the Summary tooltip shows the source VOPs/frames in the video file from which the displayed VOPs/frames are generated.

The three numbers, xxx, bbb, fff, are:

- xxx is the frame number of the decoded VOP
- bbb is the frame number of the backward reference
- fff is the frame number of the forward reference

NOTE. *When B-VOPs are used, it means that the display is different from the decode order in every VOP after the first B-VOP occurs in the bitstream (in the display order).*

VOP bits [MPEG-4] or Bits (frame/top-field) and Bits (bottom-field) [H.264/AVC, VC-1] or Frame bits [H.264/AVC, MPEG-2, H.263, VC-1]. The number of bits used in that:

- [MPEG-4] VOP
- [H.264/AVC, VC-1] frame if progressive coded or top-field if interlaced; data only appears in the Bits (bottom-field) if the frame is interlaced.
- [MPEG-2, H.263, VC-1] frame

NOTE. *For H.264/AVC, if any of the slices in the frame are interlaced, the whole frame is reported as interlaced.*

Reference Layer(s) Frame Bits [H.264/SVC only]. Represents the cumulative number of bits in the reference layer(s), used for decoding the target frame in a given access unit.

Header bits [all standards] or Header bits (frame/top-field) and Header bits (bottom-field) [interlace]. The sum of non-MacroBlock data bits within the frame minus the header bits at the start of a frame that are before the bits specific to the first MacroBlock in the frame.

MacroBlock bits [all standards] or MacroBlock bits (frame/top-field) and MacroBlock bits (bottom-field) [interlace]. The sum of MacroBlock data bits within the frame minus those bits specifically used to encode the MacroBlocks (excluding the header bits at the start of the frame).

PSNR (Y, U, V) [all standards] or PSNR (Y, U, V) (frame/top-field) and PSNR (Y, U, V) (bottom field) [interlace]. When fidelity analysis is enabled, the selected fidelity analysis values are displayed for the VOP/frame currently being displayed.

The Y, U, V values given are the average of the values for each MacroBlock.

Mean quant [all standards]. The mean average of all coded MacroBlock quant values for this frame.

Reference frames [H.264/AVC only]. This shows the List 0 and List 1 reference frames to which the current display frame refers.

NOTE. *The images of the reference frames can easily be seen using the Video navigator view.*

AVC MB Percentage [H.264/SVC only]. The percentage of "pure" AVC Macroblocks in the current frame.

NOTE. *The "Pure" AVC MB type does not employ any of the SVC specified inter-layer prediction tools.*

SVC MB Percentage [H.264/SVC only]. The percentage of SVC Macroblocks in the current frame.

Video standard [all standards]. The video standard used, with various additional information as appropriate, such as:

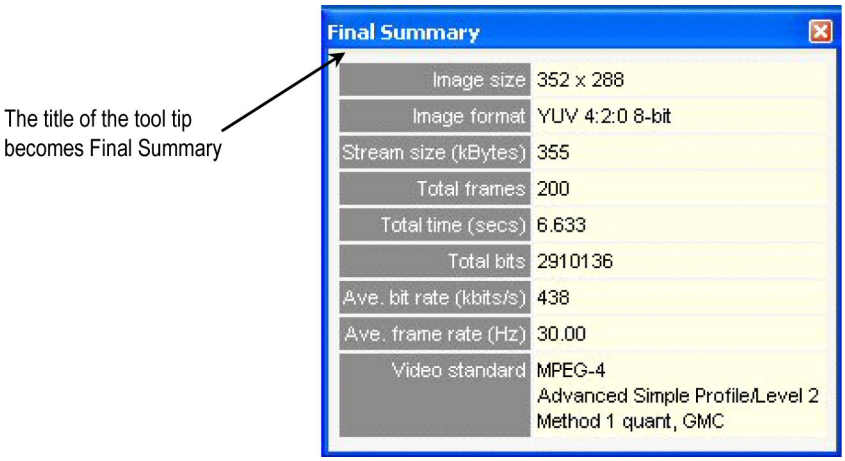
- [MPEG-4]:
  - Profile and Level, if this information is given in the file
  - Data partitioning, Resync, RVLC, as appropriate
  - [MPEG-4 Advanced Simple Profile] Interlace, QS, Method 1 quant, as appropriate
- [MPEG-2, VC-1]:
  - Profile and Level
- [H.264/AVC]:
  - Profile and Level
  - Entropy coding mode, CAVLC

  NOTE. *For H.264/AVC, many of the other elements (for example, data partitioning) can vary by frame/slice (and in some cases MacroBlock by MacroBlock), so these other elements are reported in the MacroBlock tooltip.*

- [H.264/SVC]:
  - Profile and Level
- [H.263] Annexes, as appropriate.

Summary tooltip at the end of a video sequence

When the file is completely decoded, the summary tooltip displays as follows:



The meanings of Image size, File size (KB) and Standard are the same. (See page 5-66, *Video-frame summary tooltip Ctrl+U*.)

The other items displayed are:

Total frames. The total decoded frames in the sequence.

Total time (secs). The total time in seconds from the start to the end of the sequence, calculated as if the sequence were playing normally (MTS4EA keeps track of the playing time, regardless of whether the sequence is paused at some point or played in fast forward mode).

Total bits. The size of the video file in bits.

NOTE. *For container files such as MP4 and 3GPP, the size given is the number of bits in the video track that has been selected (not the container file).*

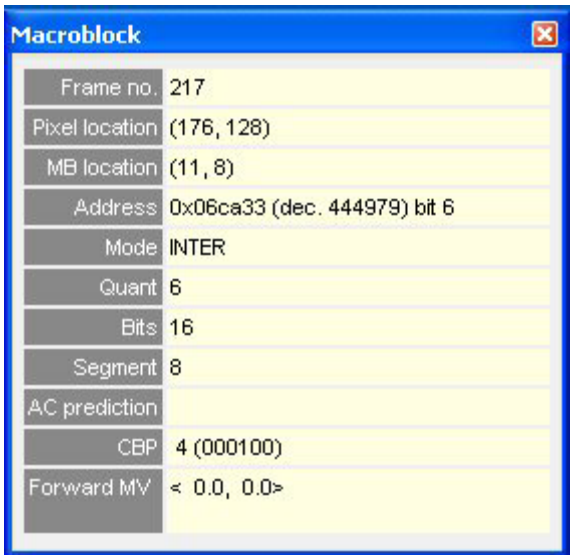
Ave. bit rate (kbits/s). This is Total bits divided by Total time, in k bits per second.

Ave. frame rate (Hz). This is Total frames divided by Total time, in frames/second (Hz).

Ave. PSNR (Y, U, V) [all standards] or Ave. PSNR (Y, U, V) (frame/top-field) and Ave. PSNR (Y, U, V) (bottom field) [interlace]. When fidelity analysis is enabled, the average of the selected fidelity analysis values across all the VOPs/frames is displayed.

MB tooltip Ctrl+M

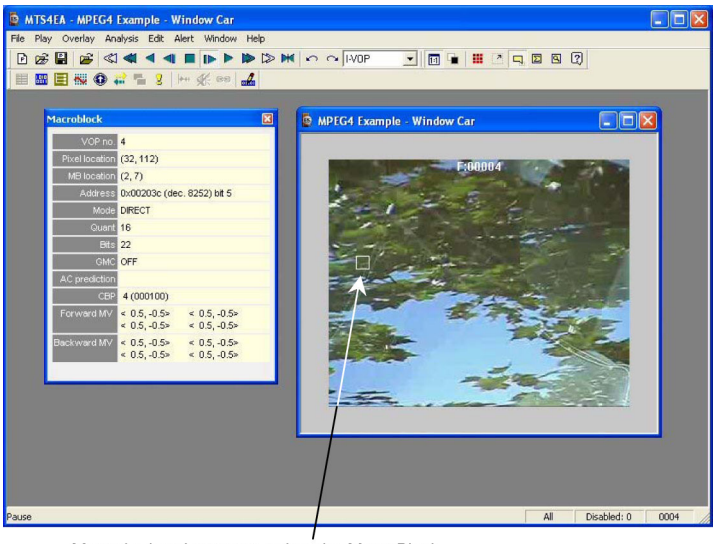
This displays information about a selected MacroBlock:



NOTE. *The information displayed by the MB tooltip varies with the video standard and the options used within the particular video sequence.* *(See page 5-77, Viewing the MacroBlock data in other views.)*

The example above is for an MPEG-4 sequence, and is examining an Inter MacroBlock.

The MacroBlock selected is signified by a white delineation box around the 16x16 MacroBlock border:



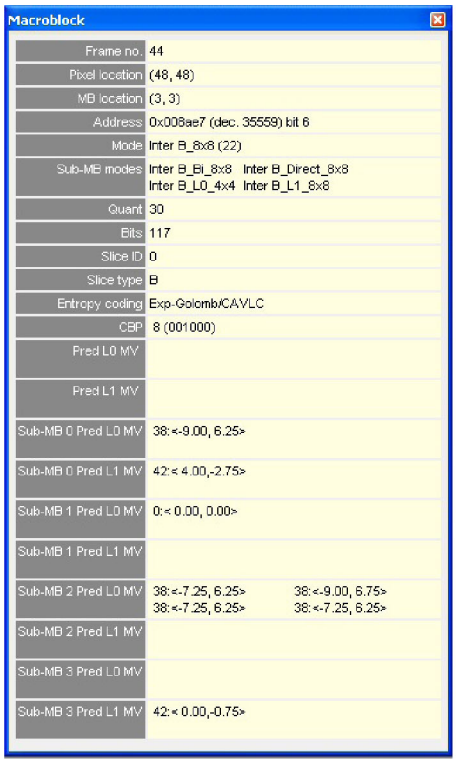
Move the hand cursor to select the MacroBlock

Viewing the MacroBlock data in other views

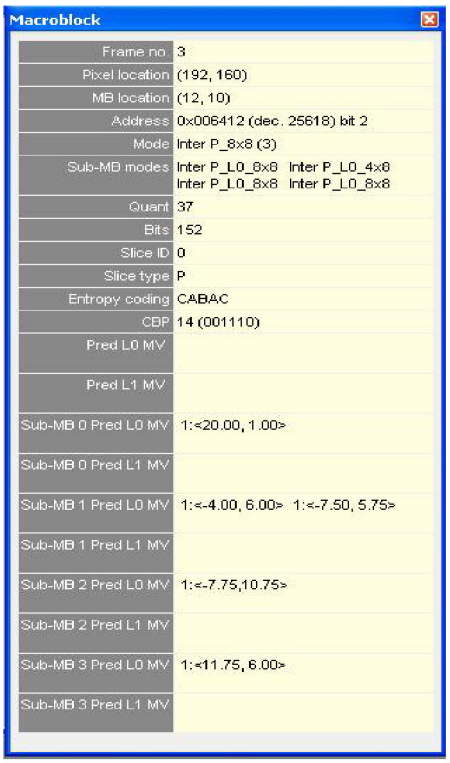
To see the MacroBlock data in other views - for example, the HexView - right click at the MacroBlock location of interest and select Goto view from the context menu, then select one of the views. For example, if Hex View is selected, the first byte of the selected MacroBlock is displayed in the HexView. (See page 5-8, *Synchronized views/navigating the views*.)

MacroBlock tooltip by video standard

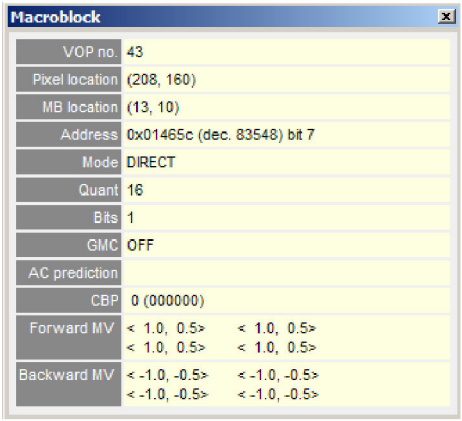
The information provided by the Macroblock tooltip varies depending upon the video standard that is being analyzed, and the options used in the video file. Examples of different streams are shown below:



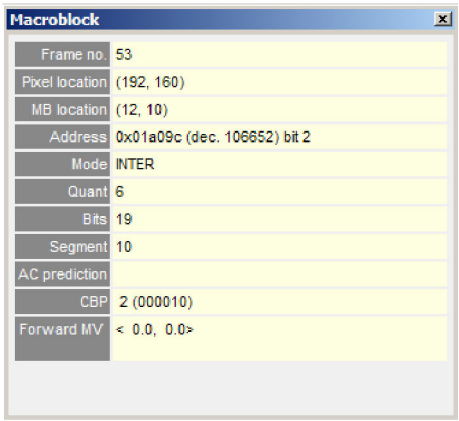
H.264/AVC



H.264/SVC



MPEG-4 Intra MB (in a Data partitioned VOP)



H.263 Inter MB (note Segment)

1641-012

Picture type. [Interlace only]. Shows whether the MacroBlock belongs to a top- or bottom-field.

VOP no. [MPEG-4] or Frame no. [H.264/AVC, VC-1, MPEG-2, H.263]. The number of the displayed VOP [MPEG-4]/frame [H.264/AVC, MPEG-2, H.263] in the sequence.

In MPEG-4 Advanced Simple Profile, H.264/AVC and MPEG-2 for sequences with B-VOPs/B-frames, the *displayed* VOP/frame number is different from the currently *decoded* VOP/frame number for every VOP/frame after the first B-VOP/B-frame occurs in the bitstream (in the *display* order) - see Video Summary Tooltip Ctrl+U for more information on decoded versus displayed VOP/frame numbers. (See page 5-66.)

Pixel location [all standards]. The location of the top-left corner pixel of the MacroBlock, where:

- 0, 0 is the top-left corner of the displayed image
- The first digit is horizontal (X) location and the second digit is the vertical (Y) location.

MB location [all standards]. The location of the MacroBlock where:

- 0, 0 is the top-left MacroBlock of the displayed image
- The first digit is horizontal (X) count and the second digit is the vertical (Y) count of MacroBlocks. For example, for a CIF-size image this will go from 0,0 (top left MacroBlock) to 21,17 (bottom right MacroBlock).

Address [all standards]. The start address of the MacroBlock within the video file.

NOTE. *The first byte in the video file is byte 0; the first bit is bit 7 of byte 0.*

For container files such as MP4, 3GPP and MPG (MPEG-2), the address given is the offset from the start of video track that has been selected, not the address within the container file (which will be different).

The byte address is displayed in hexadecimal first, for example, 0x002b8a then in decimal in brackets, for example, (dec. 11146) and finally the bit location within the byte, where bit 7 is the most significant bit (occurs first in the stream).

Frame/Field coding [H.264/AVC, VC-1, MPEG-4 Advanced Simple Profile and MPEG-2]. This displays the MacroBlock coding as either frame (progressive) or field (interlaced).

2nd DP part, 3rd DP part [optional, MPEG-4 only]. Within the MPEG-4 standard, the video data within a VOP can be divided into sections for error resilience purposes, by using the Data Partitioned flag. In this situation, the data for each MacroBlock is located at three different areas in the bitstream.

The Address field displays the address of the first part of the MacroBlock data. The 2nd DP part and 3rd DP part display the addresses of the second and third parts of the Data Partitioned MacroBlock data respectively.

Mode [all standards]. The MacroBlock type/coding mode as given in MacroBlock Types. (See page 5-91, *MB types Ctrl+Y*.)

Sub-MB modes [H.264/AVC and VC-1 Advanced Profile only]. The type/coding mode of the blocks within the MacroBlock as given in H.264/AVC. (See page 5-91, *MB types Ctrl+Y*.)

Quant [all standards]. The Quantizer used by the Luminance.

Bits [all standards]. The total number of bits used to encode the MacroBlock (including motion vector data). For Data partitioned MacroBlocks, this is the number of bits used in all three parts of the Data partitioned data for the MacroBlock.

GMC [optional, MPEG-4 Advanced Simple Profile only]. This denotes whether the MacroBlock uses Global Motion Compensation (GMC) in the calculation of motion vectors. This is either:

- ON - GMC is on
- OFF - GMC is off

Segment [H.263 only]. The segment number, representing the GOB number or slice (starting at zero from the top of the frame).

Slice [MPEG-2 only]. The number of the slice (starting at zero from the top of the frame).

Slice ID [H.264/AVC and MPEG-4 VC-1 Advanced Profile only]. The number of the slice (starting at zero from the top of the frame).

Slice type [H.264/AVC and MPEG-4 VC-1 Advanced Profile only]. The slice type, I or P or [Extended Profile only] B or SI or SP.

Entropy coding [H.264/AVC and MPEG-4 VC-1 Advanced Profile only]. The entropy coding mode of the slice, CAVLC.

Transform [MPEG-4 Advanced Simple Profile and VC-1 Advanced Profile]. Shows the DCT type: field or frame, as determined by the value of dct_type in MPEG-4 ASP interlaced_information().

AC prediction [MPEG-4, H.264/AVC, and H.263]. This is used differently for MPEG-4 and for H.263:

- MPEG-4:
  - ON - AC prediction is on
  - OFF - AC prediction is off
- H.263, Advanced Intra Coding mode as specified in Annex I, the values used are:
  - 0 - DC prediction only
  - 1 - Vertical DC & AC prediction
  - 2 - Horizontal DC & AC prediction

CBP [all standards]. Coded Block Pattern: this is a number from 0-63, signifying which of the six 8x8 blocks that make up a MacroBlock have been coded - the bit pattern is reproduced for the block in brackets, where reading left to right shows which of the six blocks were coded (1 = coded, 0 = not coded). The blocks are ordered as Y0, Y1, Y2, Y3, U, V.

Forward MV [MPEG-4, MPEG-2, and H.263 only] and Forward MV (frame/top-field), Forward MV (bottom field) [interlace]. The Motion Vectors for the MacroBlock. The numbers in <> brackets indicate the relative position of the area from where the data for this MacroBlock has come. The numbers are for X and Y respectively (positive numbers are down/to the right; negative numbers are up/to the left).

For example, <-7.5, -5.0> means that the data in this MacroBlock came from the area of the preceding VOP/frame, which is 7.5 pixels up and 5.0 pixels to the left of the current MacroBlock.

(The term Forward in this field is used to specify that these Motion Vectors are forward predicted – they have been predicted forward from a preceding VOP/frame.)

[MPEG-4 Advanced Simple Profile] For interlaced MacroBlocks, this shows which fields the forward Motion Vectors predict from, as determined by the value of forward_top_field_reference and forward_bottom_field_reference in MPEG-4 Adv. Simple Profile interlaced_information(). This is displayed as follows:

Display	Definition
Forward motion vector (frame/top-field)	In frame coded MacroBlocks, shows the forward motion vectors. In field coded MacroBlocks, shows the forward top-field motion vectors.
Forward motion vector (bottom-field)	In field coded MacroBlocks, shows the forward bottom-field motion vectors.

Backward MV [MPEG-4 Advanced Simple Profile (B-VOPs), MPEG-2 and VC-1 Advanced Profile] and Backward MV (frame/top-field), Backward MV (bottom field) [interlace]. These only appear for B-VOPs in MPEG-4 Advanced Simple Profile, MPEG-2 B-frames and VC-1 Advanced Profile; these are the Backward predicted Motion Vectors. The numbering follows the convention given above for Forward motion vectors.

(The term Backward in this field is used to specify that these Motion Vectors are backward predicted, they have been predicted backward from a following VOP/frame.)

[MPEG-4 Advanced Simple Profile] For interlaced MacroBlocks, this shows which fields the forward Backward Vectors predict from, as determined by the value of backward_top_field_reference and backward_bottom_field_reference in MPEG-4 Adv. Simple Profile interlaced_information(). This is displayed as follows:

Display	Definition
Backward motion vector (frame/top-field)	In frame coded MacroBlocks, shows the backward motion vectors. In field coded MacroBlocks, shows the backward top-field motion vector.
Backward motion vector (bottom-field)	In field coded MacroBlocks, shows the backward bottom-field motion vector.

Pred L0 MV, Pred L1 MV [H.264/AVC, MPEG-4 and VC-1 Advanced Profile]. The Motion Vectors for the MacroBlock. Data only appears in this field if there is no subdivision of the MacroBlock:

- L0 refers to List 0 reference frames (forward reference, from past frames; as an example, P type MacroBlocks can only be List 0)
- L1 refers to List 1 reference frames (is forward and backward reference, from past and future frames; as an example, B type MacroBlocks can be List 1 or List 0).

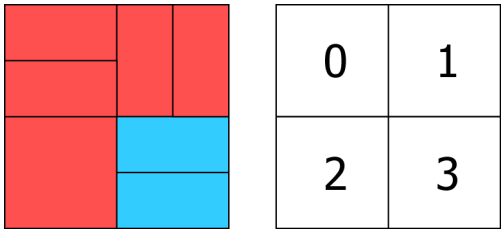
For H.264/AVC only, the number of the referenced frame is also included in front of the Motion Vector, for example, 38:<9.00,6.25> means that these motion vectors reference frame 38.

NOTE. *The images of the reference frames can easily be seen using the Video navigator view.*

Sub-MB 0/1/2/3 Pred L0/L1 MV [H.264/AVC, VC-1 Advanced Profile and MPEG-4]. The Motion Vectors for the blocks within the MacroBlock, when the MacroBlock has been subdivided (when not subdivided, there is no data in these fields).

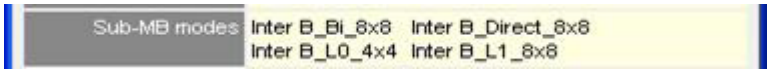
The data provided here follows the standard and is shown in the layout as given in the Sub-MB modes field of the MacroBlock tooltip and MB types overlay.

As an example, for a MacroBlock which has been divided as follows:



where the red blocks are Inter_List_0 (P) and the blue blocks are Inter_List_1 (B) and the numbers denote the sub-MacroBlock areas 0, 1, 2, 3.

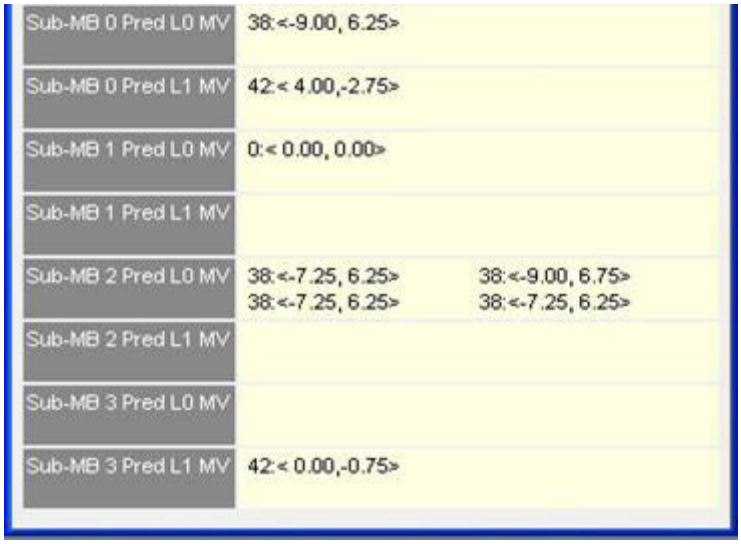
Then the Sub-MB modes field would look like this:



where:

- The top-left reported sub-MB mode is for area 0
- The top-right is for area 1, and so forth.

The MV fields would look like this:

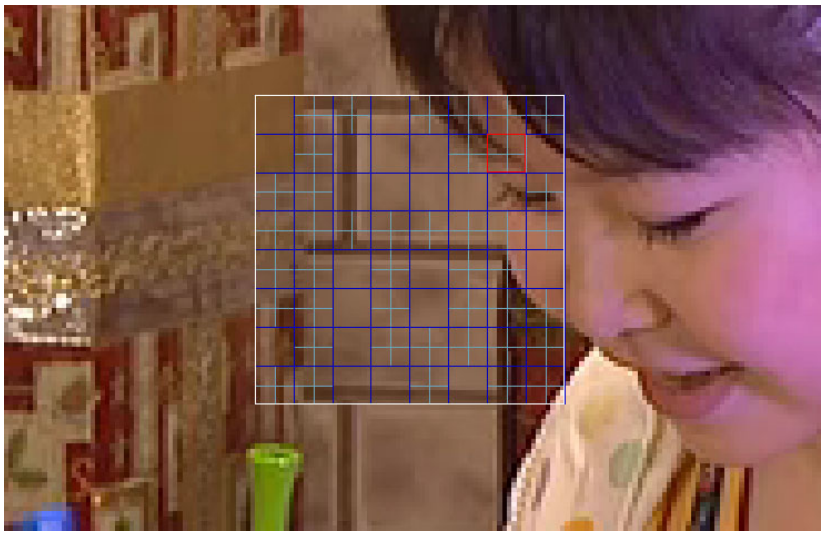


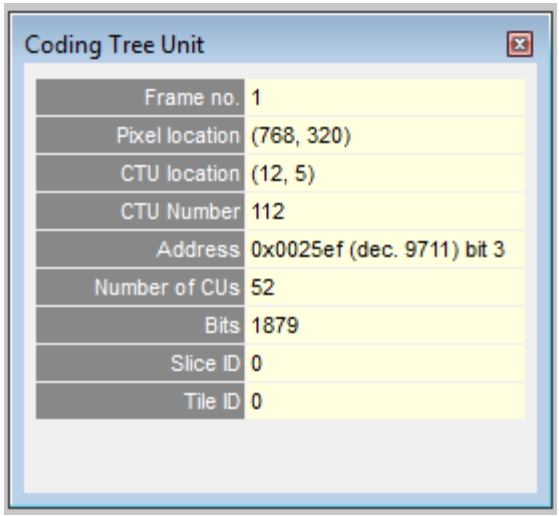
PSNR (Y, U, V) or PSNR (Y, U, V) (frame/top-field) and PSNR (Y, U, V) (bottom field) [interlace]. When fidelity analysis is enabled, the selected fidelity analysis value for the selected MacroBlock is displayed.

CTU tooltip Ctrl+N

This tooltip is available only for H.265/HEVC stream analysis. The icon selection for the CTU tooltip is available in the HEVC toolbar.

When you select a CTU, the CTU is highlighted as shown below. The white delineation box shows the selected CTU. The red box shows the locked CU, whose details will be updated in the CU tooltip. The teal color grids show the Prediction unit partitions in the selected CTU.





The CTU tooltip displays information about the selected CTU as described below:

Frame no. The number of the frame in the sequence.

Pixel location. The location of the top-left corner pixel of the CTU, where:

- 0, 0 is the top-left corner of the displayed image
- The first digit is horizontal (X) location and the second digit is the vertical (Y) location

CTU location. The location of the CTU where:

- 0, 0 is the top-left CTU of the displayed image
- The first digit is horizontal (X) count and the second digit is the vertical (Y) count of CTUs. For example, for a CIF-size image this will go from 0,0 (top left CTU) to 21,17 (bottom right CTU).

CTU number. The absolute CTU number in the video frame, in "raster scan order."

Address. The start address of the CTU within the video file.

NOTE. *The first byte in the video file is byte 0; the first bit is bit 7 of byte 0.*

For container files, the address given is the offset from the start of video track that has been selected, not the address within the container file (which will be different).

The byte address is displayed in hexadecimal first, for example, 0x002b8a then in decimal in brackets, for example, (dec. 11146) and finally the bit location within the byte, where bit 7 is the most significant bit (occurs first in the stream).

Number of CUs. The total number of CUs in the CTU.

Bits. The total number of bits in the CTU (including the header bits in the CTU).

Slice ID. The number of the slice (starting at zero from the top of the frame).

Tile ID. The number of the tile (starting at zero from the top of the frame).

CU tooltip Ctrl+D

This tooltip is available only for H.265/HEVC stream analysis. The icon selection for the CU tooltip is available in the HEVC toolbar.

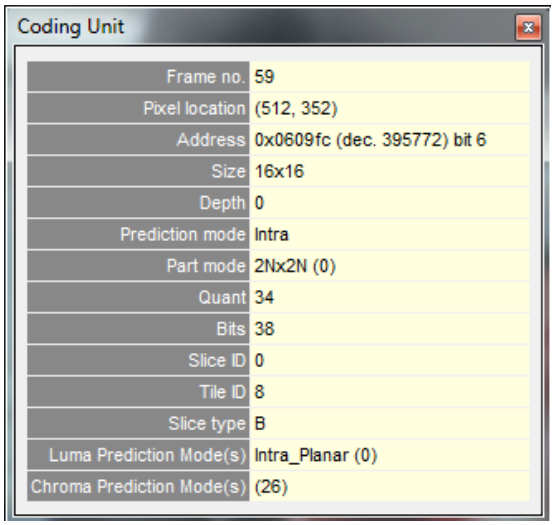


Figure 5-1: CU tooltip – Intra Prediction Mode

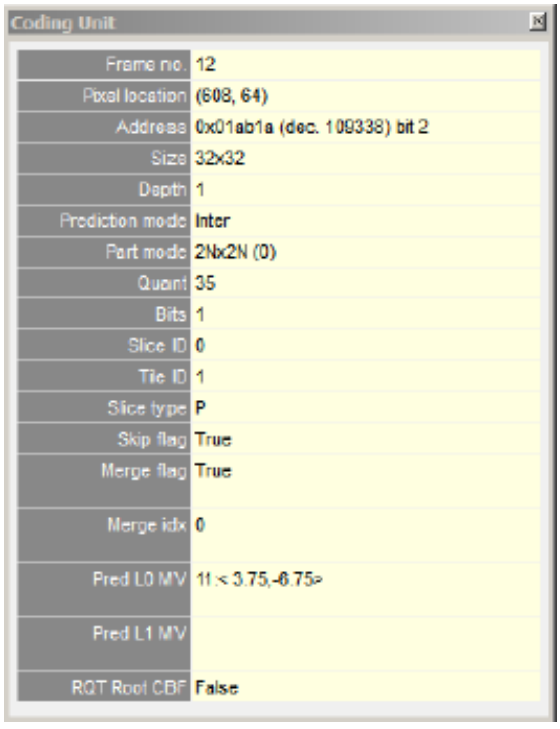


Figure 5-2: CU tooltip – Inter Prediction Mode

The CU tooltip displays information about the selected CU as described below:

Frame no. The number of the frame in the sequence.

Pixel location. The location of the top-left corner pixel of the CU, where:

- 0, 0 is the top-left corner of the displayed image
- The first digit is horizontal (X) location and the second digit is the vertical (Y) location

Address. The start address of the CU within the video file.

NOTE. *The first byte in the video file is byte 0; the first bit is bit 7 of byte 0.*

For container files, the address given is the offset from the start of video track that has been selected, not the address within the container file (which will be different).

The byte address is displayed in hexadecimal first, for example, 0x002b8a then in decimal in brackets, for example, (dec. 11146) and finally the bit location within the byte, where bit 7 is the most significant bit (occurs first in the stream).

Size. The size of the Coding Unit (CU).

Depth. Indicates the level of CU split (ranging from 0 = No Split, to 3 = Maximum possible CU division).

Prediction mode. The Prediction Mode used while encoding the CU. The possible values are Intra and Inter.

Quant. The Quantizer used by the Luminance.

Bits. The total number of bits used to encode the CU (including motion vector data). For Data partitioned CUs, this is the number of bits used in all three parts of the Data partitioned data for the CU.

Slice ID. The number of the slice (starting at zero from the top of the frame).

Tile ID. The number of the tile (starting at zero from the top of the frame).

Slice type. The slice type, I or P or (Extended Profile only) B or SI or SP.

Luma prediction mode(s). This field appears only for Intra CUs. Specifies the prediction direction to be used to predict the luma (Y) signal from the neighboring CU blocks of the same frame (Range 0 to 34, inclusive).

Chroma prediction modes(s). This field appears only for Intra CUs. Specifies the prediction direction to be used to predict the chroma (U and V) signal from the neighboring CU blocks of the same frame. The allowed values are 0, 1, 10, 26, 34, or 36.

Skip flag and Merge flag. These fields appear only for Inter CUs.

Merge idx. This field appears only for Inter CUs. This is one of the parameters used to decide how to predict the MV for the current PU (Range 0 to 6, inclusive).

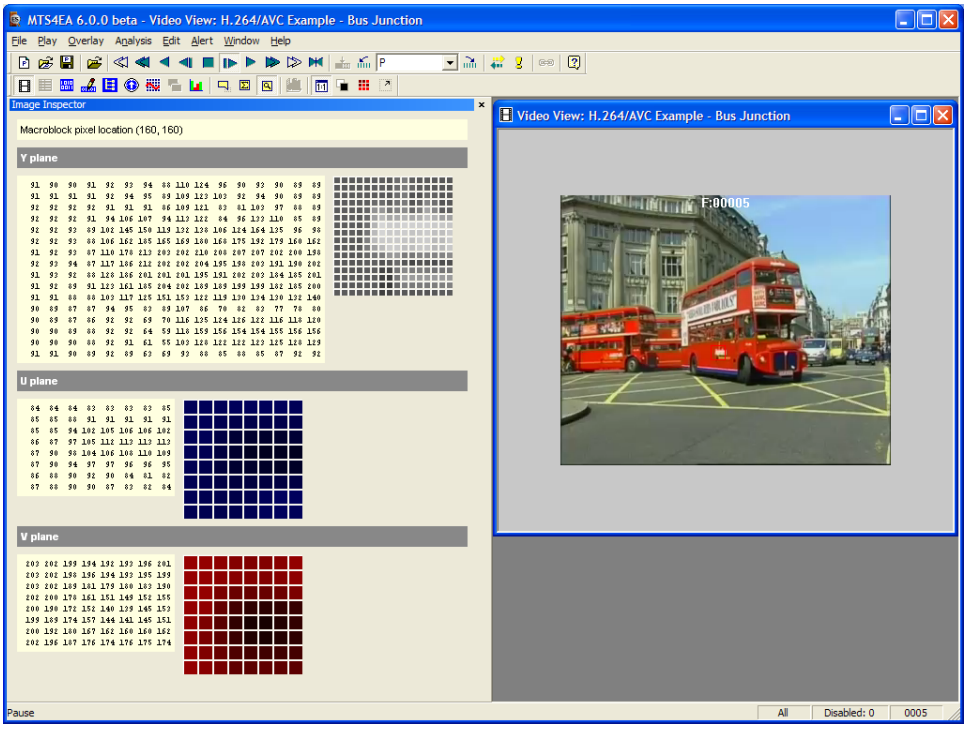
Pred L0 MV and Pred L1 MV. These fields appear only for Inter CUs. These are similar to the AVC standard. They specify the MV values in List 0 and List 1 used for inter-prediction. The possible values depend on the profile, tier, and level of the stream.

RQT Root CBF. This field appears only for Inter CUs. The Residual Quad Tree Root Coded Block Flag indicates the presence of TU for the current CU.

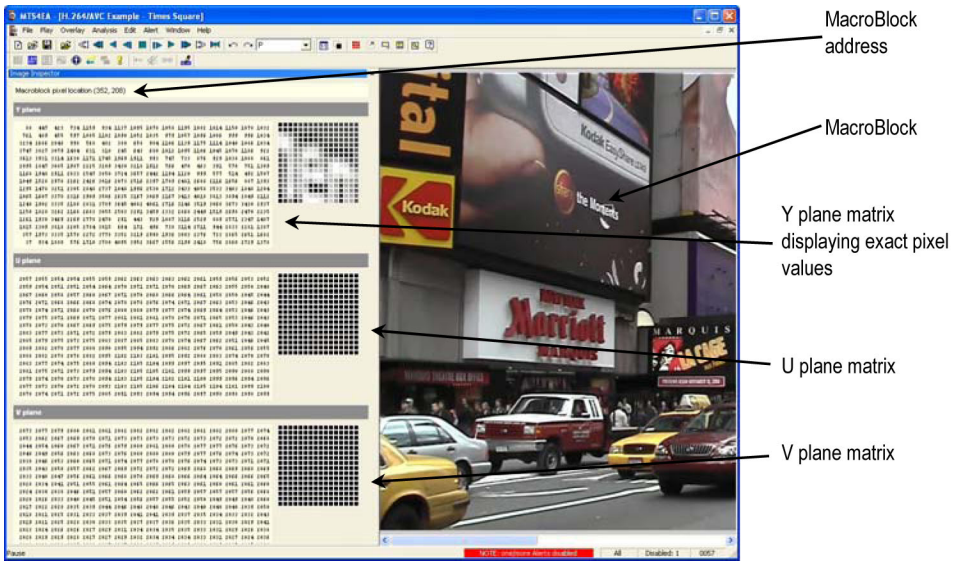
Image inspector Ctrl+I

The Image inspector provides a magnifying glass on the decoded (or predicted, or residual video frame, in the case of H.265/HEVC streams) based on the Video View type. It also provides the ability to view pixel data for the individual Macroblocks (for H.264 and earlier video standards) or Coding Units (for H.265/HEVC video).

The pixel matrix size depends on both the selected CU/Macroblock size and bit depth. In the case of H.265/HEVC streams, the CU size can be 8x8, 16x16, 32x32 and 64x64.



The subsampling ratio of these component channels can vary. As can be seen in the preceding screenshot, when this is the case, the matrices containing the pixel data are sized accordingly. The preceding screenshot shows an example of 8 bits per sample, 4:2:0 YUV video, the following screenshot shows an example of 12 bits per sample, 4:4:4 video.

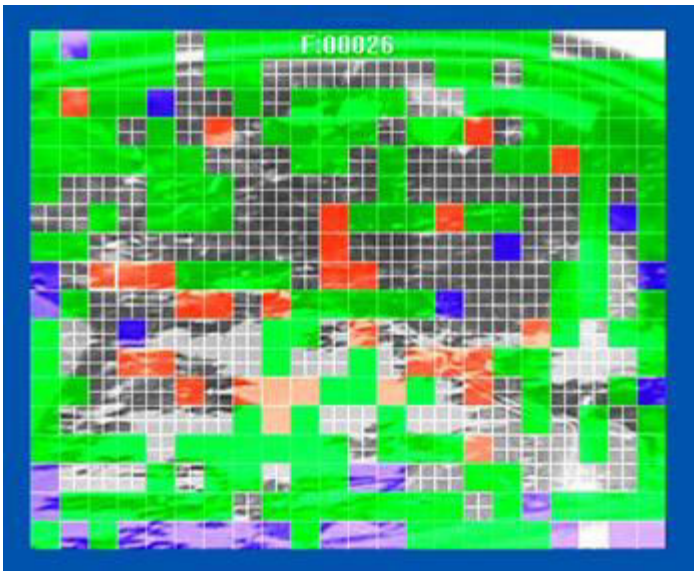


The colors of the image blocks change, depending on the data type:

	Top image block	Middle image block	Bottom image block
YUV	Gray (Y)	Blue (Cb)	Red (Cr)
RGB	Red	Green	Blue

MB types Ctrl+Y

This example is a B-VOP of an MPEG-4 Advanced Simple Profile stream.



This overlay allows you to easily view the MacroBlock types used in the encoding. The types are identified by changing the color of the individual MacroBlocks, by modifying the chrominance while leaving the luminance unchanged, so that the data is still partially visible.

The colors used are dependent on the video compression standard in use. The colors are shown in the following tables and are also displayed on-screen in the MacroBlock types color key tooltip. (See page 5-296, *MacroBlock Types color key tooltip*.)

The lines denote the edges of the MacroBlock, and if the MacroBlock has four motion vectors, the lines also divide the MacrBlock into four quadrents. The color of the lines can be changed from white to black by clicking the icon (or clicking the Black menu item on the Overlay menu).

H.264/AVC

H.264/AVC has the following possible coded MacroBlock types (see Reference [12] in *Compression Standards and File Types* section) (See page 4-1.), shown in the following colors:

MB type	Prediction mode	Table index	Sub-MB types	Color	Slice type(s)
I	Intra 16x16 (incl. Intra inferred)	1 - 25	-	Green	I, P, B, SI, SP
I	Intra 4x4	0	-	Yellow	I, P, B, SI, SP
I	Intra 8x8	0	-	Orange	I, P, B, SI, SP
P	Inter list 0	0 - 2	0 - 3	Red	P, SP
B	Inter list 0	1, 4, 5	1, 4, 5, 10	Red	B
B	Inter list 1	2, 6, 7	2, 6, 7, 11	Blue	B
B	Inter list 0 + 1	3, 20, 21	3, 8, 9, 12	Pink	B
B	Inter mixed	8 - 19	-	Light blue	B
B	Inter direct	0	-	Gray	B
SI	Intra 4X4 and 16x16	0	-	Green	SI
SP	Inter list 0	0 - 2	0 - 3	Red	SP

NOTE. *The Intra inferred MacroBlock type was shown in yellow in previous versions of MTS4EA; but, because it occurs infrequently and it is a 16x16 type, it is now grouped in the Intra 16x16 MacroBlock type, and the Intra 4x4 MacroBlock type is now shown in yellow.*

H.264/SVC

H.264/SVC has the following possible coded MacroBlock types (see Reference [26] in *Compression Standards and File Types* section) (See page 4-1.), shown in the following colors:

MB type	Prediction mode	Table index	Sub-MB types	Color	Slice type(s)
I	Intra 16x16 (incl. Intra inferred)	1 - 25	-	Green	I, P, B, EI, EP, EB
I	Intra 4x4	0	-	Yellow	I, P, B, EI, EP, EB
I	Intra 8x8	0	-	Orange	I, P, B, EI, EP, EB
P	Inter list 0	0 - 2	0 - 3	Red	P, EP
B	Inter list 0	1, 4, 5	1, 4, 5, 10	Red	B, EB
B	Inter list 1	2, 6, 7	2, 6, 7, 11	Blue	B, EB
B	Inter mixed	8 - 19	-	Light blue	B, EB
B	Inter direct	0	-	Gray	B, EB
Intra_BL	base_mode _flag = 1	-	-	Teal	I, P, B, EI, EP, EB
Inter_BL	base_mode _flag = 1	-	-	Bright green	P, B, EP, EB
Residual prediction	residual _prediction_flag = 1	-	-	Brown	P, B, EP, EB
Motion prediction	motion _prediction_flag = 1	-	-	Black	P, B, EP, EB

VC-1

VC-1 has the following possible coded MacroBlock types (see Reference [16] in *Compression Standards and File Types* section) (See page 4-1.), shown in the following colors:

Type	Motion vector	Color	Frame type(s)
Intra	0	Green	I, P, B, BI
Forward	0, 1, 2, or 4	Red	P, B
Backward	0, 1, 2, or 4	Blue	B
Bi-directional	0, 2, or 4	Pink	B

MPEG-4

MPEG-4 has the following possible coded MacroBlock types (see Reference [1] - Table B-1 in *Compression Standards and File Types* (See page 4-1.) section), shown in the following colors:

Index	Type	Dquant/ Dbquant	Motion vector	Color	VOP type(s)
0	Inter		1	Red	P, S-GMC
1	Inter+Q		1	Blue	P, S-GMC
2	Inter4V		4	Pink	P, S-GMC
3	Intra			Green	I, P, S-GMC
4	Intra+Q			Yellow	I, P, S-GMC
-	Direct		8	Gray	B
-	Interpolate		2	Green	B
-	Forward		1	Red	B
-	Backward		1	Blue	B

MPEG-2

MPEG-2 has the following possible coded MacroBlock types (see Reference [13] in *Compression Standards and File Types* (See page 4-1.) section), shown in the following colors:

Type	Dquant	Motion vector	Color	Frame type(s)
Intra		1	Green	I, P, B
Forward		2	Red	P, B
Backward		2	Blue	B
Bi-directional		4	Pink	B

H.263

H263 has six possible coded MacroBlock types (see Reference [2] - Table 9/H.263 - in *Compression Standards and File Types* section) (See page 4-1.), shown in the following colors:

Index	Type	Dquant	Motion vector	Color
0	Inter		1	Red
1	Inter+Q		1	Blue
2	Inter4V		4	Pink
3	Intra			Green
4	Intra+Q			Yellow
5	Inter4V+Q		4	Gray

CU types Ctrl+J

This overlay is available only for H.265/HEVC stream analysis. The icon selection for the CU overlay is available in the HEVC toolbar.

This overlay allows you to visually see the Coding Unit (CU) types used while encoding the frame. The data in each of the CUs is partially visible and the chroma component is modified to indicate the CU Type.

The colors used are displayed in the CU types key tooltip. The CUs that are displayed grey implies that they are skipped (Skip flag-true; this can be checked in the Coding Unit tooltip).

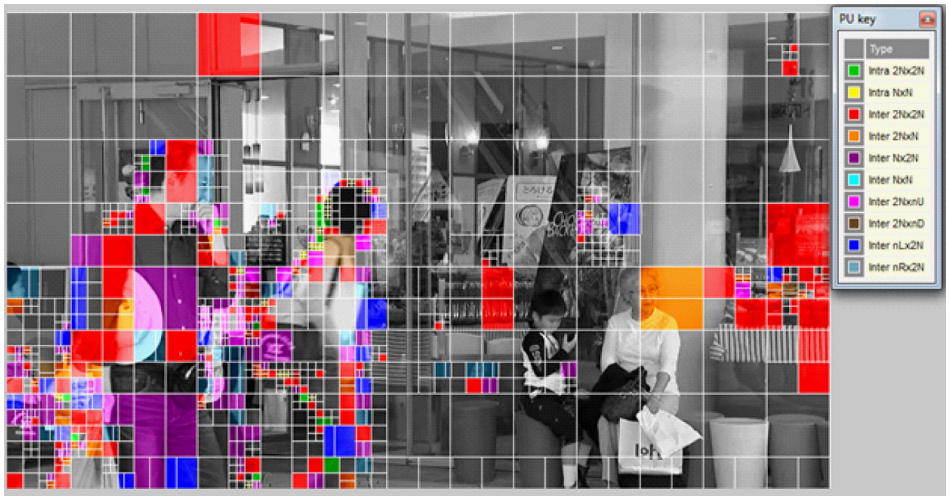
The gridlines shown above are displayed only when the video is either in pause or stop mode. In all other modes, only the CU overlay colors are displayed.

The color of the gridlines can be set to white or black by toggling the icon or by configuring the Overlay color setting in the Overlay menu.

Partition types Ctrl+L

This overlay is available only for H.265/HEVC stream analysis. The icon selection for the Prediction Unit (PU) overlay is available in the HEVC toolbar.

This overlay allows you to visually see the partition types (PU types) used while encoding the frame. The data in each of the PUs is partially visible and the chroma component is modified to indicate the PU Type.



The colors used are displayed in the PU key tooltip. The PUs that are displayed grey implies that they are skipped (Skip flag-true; this can be checked in the Coding Unit tooltip).

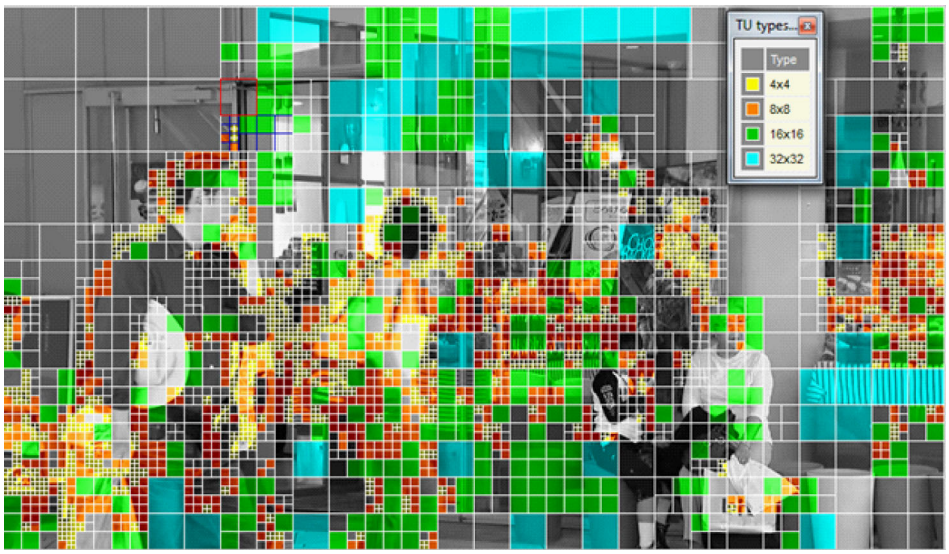
The gridlines shown above are displayed only when the video is either in pause or stop mode. In all other modes, only the PU overlay colors are displayed.

The color of the gridlines can be set to white or black by toggling the icon or by configuring the Overlay color setting in the Overlay menu.

Luma TU Ctrl+Q

This overlay is available only for H.265/HEVC stream analysis. The icon selection for the Luma Transform Unit (TU) overlay is available in the HEVC toolbar.

Unlike the other overlays, where the types are primarily based on the type of encoding (inter/intra etc), the Luma TU overlay provides a picture of TU types based on their sizes (4x4, 8x8, etc).



The colors used are displayed onscreen in the TU types color key tooltip. TUs that are displayed grey imply that they are skipped (Skip flag-true; this can be checked in the Coding Unit tooltip).

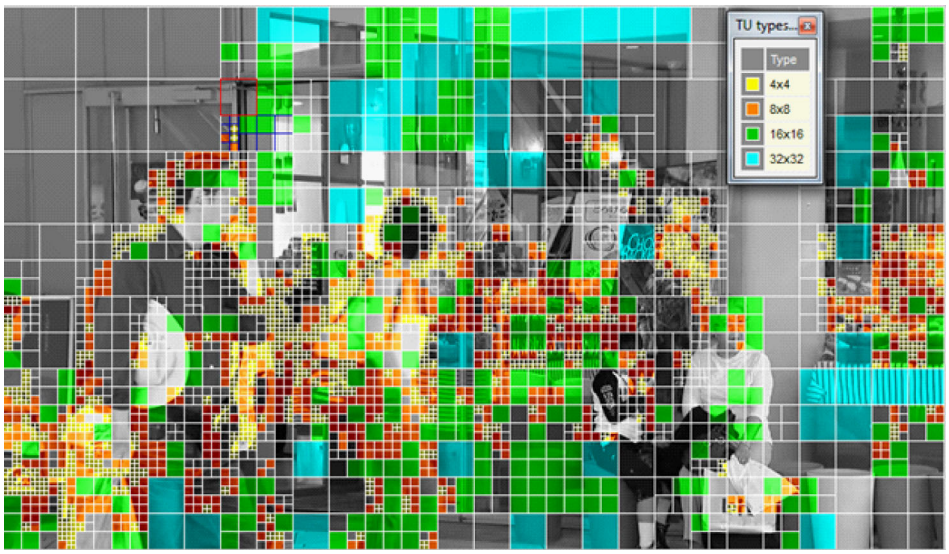
The gridlines shown above are displayed only when the video is either in pause or stop mode. In all other modes, only the TU overlay colors are displayed.

The color of the gridlines can be set to white or black by toggling the icon or by configuring the Overlay color setting in the Overlay menu.

Chroma TU Ctrl+Shift+Q

This overlay is available only for H.265/HEVC stream analysis. The icon selection for the Chroma Transform Unit (TU) overlay is available in the HEVC toolbar.

Unlike the other overlays, where the types are primarily based on the type of encoding (inter/intra etc), the Chroma TU overlay gives a picture of TU types based on their sizes (4x4, 8x8, etc).



The colors used are displayed onscreen in the TU types color key tooltip. TUs that are displayed grey imply that they are skipped (Skip flag-true; this can be checked in the Coding Unit tooltip).

The gridlines shown above are displayed only when the video is either in pause or stop mode. In all other modes, only the TU overlay colors are displayed.

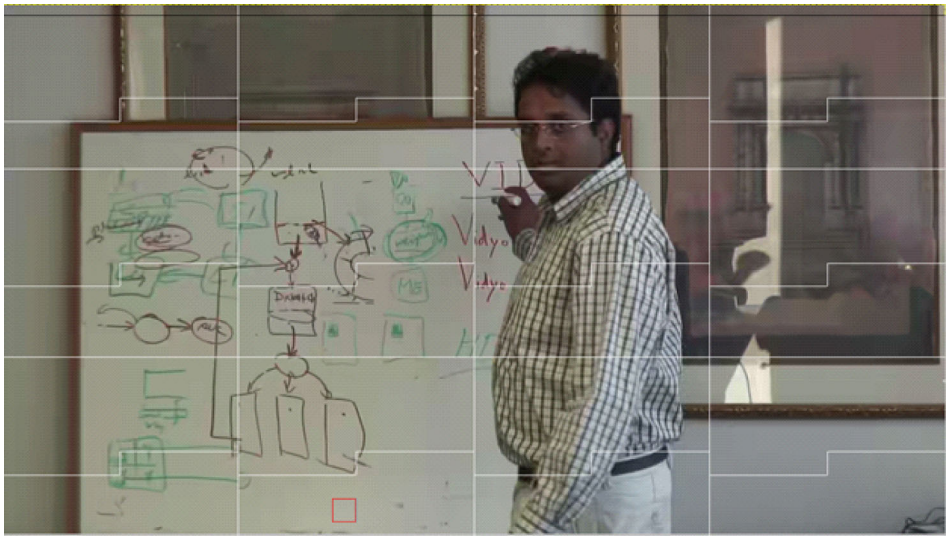
The color of the gridlines can be set to white or black by toggling the icon or by configuring the Overlay color setting in the Overlay menu.

In case of the Chroma TU, a 8x8 CU can be colored yellow completely marking 4x4 (as per color key tooltip). This implies that only a 4x4 region of the 8x8 CU has been transform coded (4:2:0 subsampling).

Slice Ctrl+Shift+S

This overlay is available only for H.265/HEVC stream analysis. The icon selection for the Slice overlay is available in the HEVC toolbar.

The Slice overlay allows you to visually see the slice partitions in the frame.



In cases where the frame has a single slice, the Slice overlay does not overlay any slice boundaries. The number of slices in the frame can be found from the "Number of slices" field in the Video-frame summary tooltip.

Slice overlay grids are available even while the video is being played (unlike the CU, PU, and TU overlays).

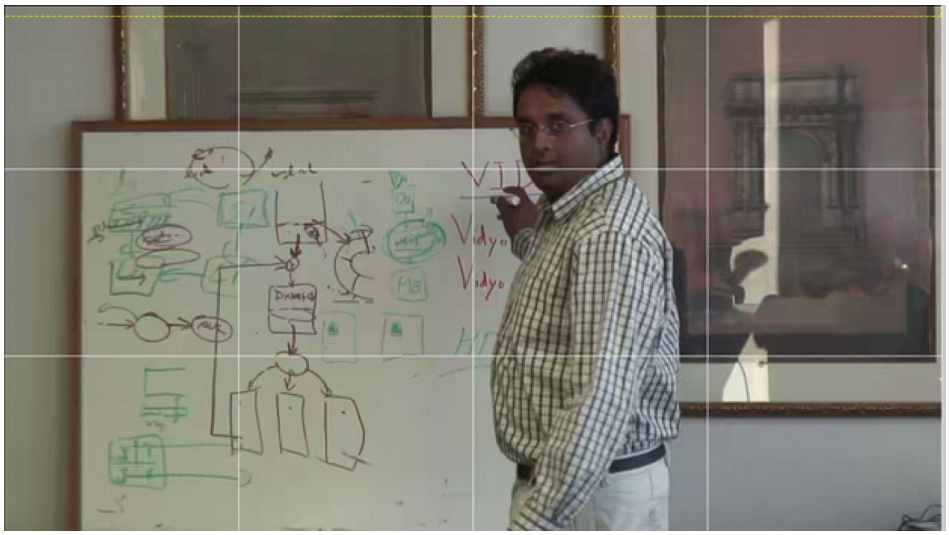
The color of the gridlines can be set to white or black by toggling the icon or by configuring the Overlay color setting in the Overlay menu.

Locking of a particular CU is available in the Slice overlay. However, when you hover the mouse over a selected CTU, the CU cursor is not updated when the Slice overlay is enabled. Given that the frame does not contain more than one slice, the selected CTU is drawn both on mouse-hover and lock of the CU.

Tile Alt+T

This overlay is available only for H.265/HEVC stream analysis. The icon selection for the Tile overlay is available in the HEVC toolbar.

The Tile overlay allows you to visually see the tile partitions in the frame.



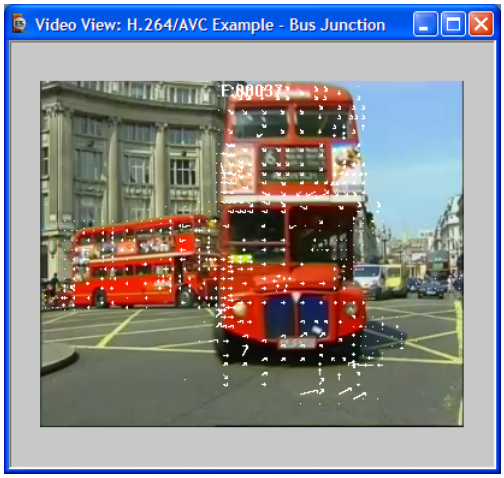
In cases where the frame has just a single tile, the selection of the Tile overlay does not put up any tile boundary. The number of tiles in the frame can be found from the “Number of tiles” field in the Video-frame summary tooltip.

Tile overlay grids are available even while the video is being played (unlike the CU, PU, and TU overlays).

The color of the gridlines can be set to white or black by toggling the icon or by configuring the Overlay color setting in the Overlay menu.

Locking of a particular CU is available in the Tile overlay. However, when you hover the mouse over a selected CTU, the CU cursor is not updated when the Tile overlay is enabled. Given that the frame does not contain more than one tile, the selected CTU is drawn both on mouse-hover and lock of the CU.

Motion vectors Ctrl+E



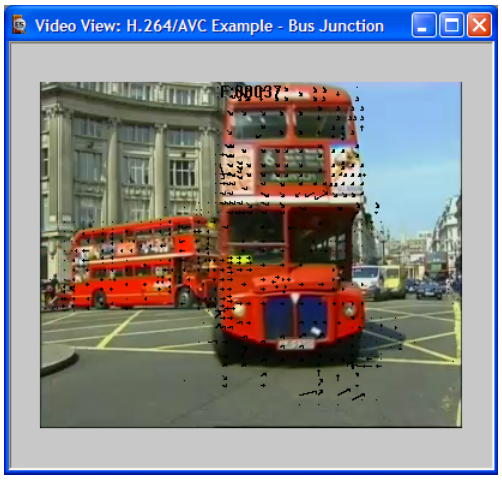
The motion vectors used in relevant video frames can be displayed on the frame currently being viewed. They are switched on and off by doing one of the following:

- From the Overlay menu, select the Motion vectors option
- Click the toolbar icon
- Press Ctrl+E

The motion vectors are drawn in the image in either white or black (the color can be selected using the Black/White Digits button on the Toolbar see Overlay Color for all frames except B-VOPs/B-frames - see *Motion Vectors in B-VOPs/B-frames* for information on this.

(See page 5-126, *Overlay color Ctrl+W*.) (See page 5-103, *Motion vectors in B-VOPs/B-frames*.)

This is an example of the above video frame with the motion vectors plotted in black:



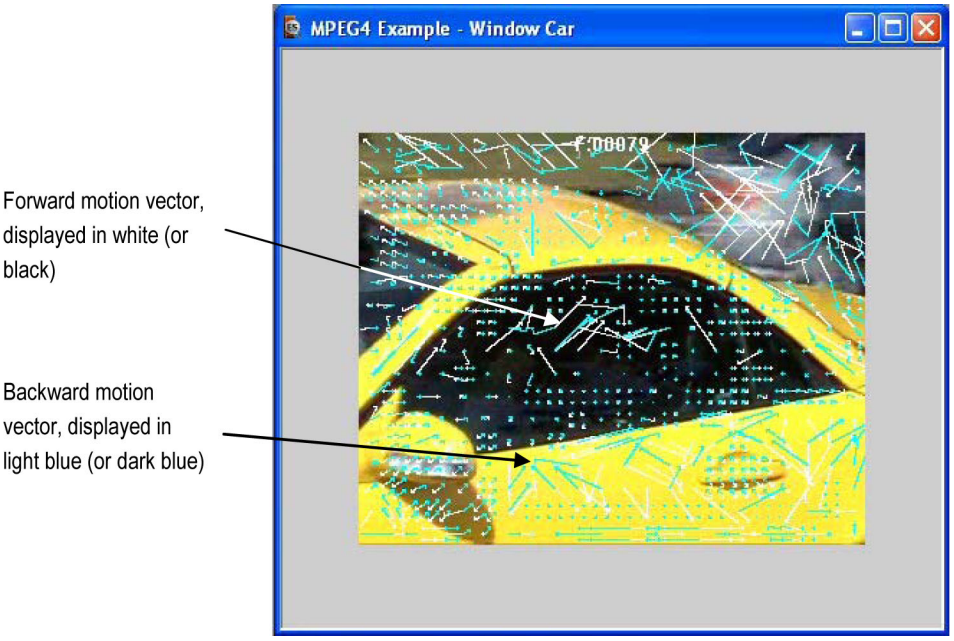
The motion vectors are plotted to the length of that actually used by the decoder.

The vector has an arrowhead at its end and is pointing to the center of the region of pixels in the previous frame that was used for the prediction of the current MacroBlock or sub-MacroBlock.

Some video standards allow motion vectors for an 8x8 region; other video standards (such as H.264/AVC) allow motion vectors for smaller regions as well. In these situations, all the motion vectors are plotted.

Motion vectors in B-VOPs/B-frames

For frames that are bidirectionally predicted (B-VOPs or B-frames) there can be two motion vectors for each MacroBlock or sub-MacroBlock region, or four MacroBlocks in the case of interlaced video sequences:



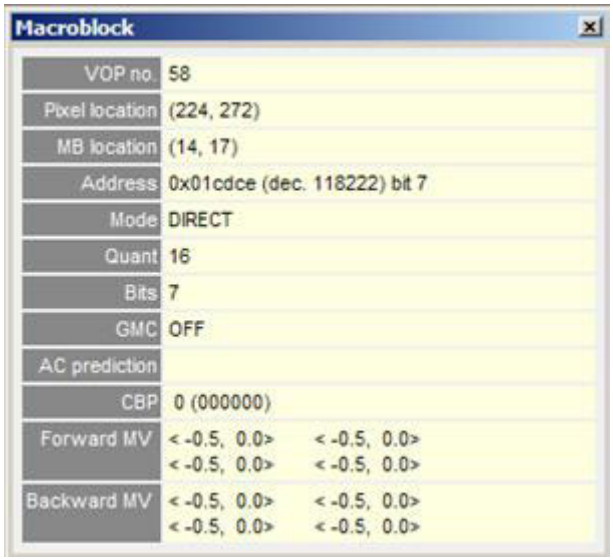
In Interlaced video sequences the motion vectors are displayed in the following colors:

MV type	Field	MV color - out	MV color - out
Forward	Top	White	Black
Backward	Top	Light blue	Dark blue
Forward	Bottom	Yellow	Green
Backward	Bottom	Magenta	Red

For H.264/AVC, the white (or black) arrows denote the List 0 motion vectors and the light blue (or dark blue) arrows denote the List 1 motion vectors.

For more information on the motion vector displays in the H.264/AVC standard. (See page 4-1, *Compression standards and file types*.)

For MacroBlocks with four (or more) motion vectors, the MacroBlock tooltip is expanded with extra information to display all the motion vectors. The example below is for a MacroBlock in a B-VOP (in MPEG-4 Advanced Simple Profile) where there are four Forward and four Backward motion vectors:



MB statistics

The MB statistics option appears only for non-HEVC streams. For H.265/HEVC streams, this option is replaced by the CTU statistics option.(See page 5-115, *CTU statistics*.)

This option displays statistics on a MacroBlock-by-MacroBlock basis, either for the current frame only or averaged over a range of frames.

The options descibed in this section are also available from the MB Statistics toolbar.

The top of this submenu allows selection of:

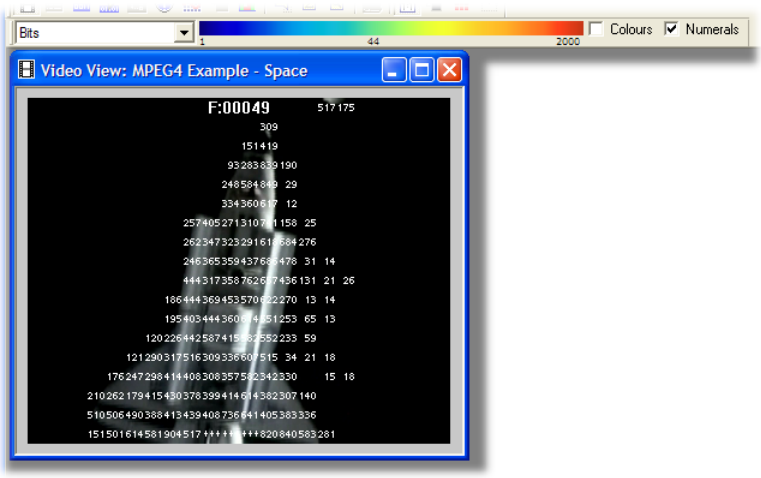
- Numerals (numbers)
- Colors

Numerals

When this is selected, numbers are displayed for the MacroBlock statistics, overlaid onto each MacroBlock.

(In the preceding example, Numerals is selected, but None is also selected, so no statistics will be displayed.)

The displayed number is selected from this submenu; the following example shows bits per MacroBlock in frame 49 of the MPEG-4 example file Space (no bits = not coded):



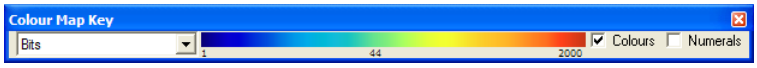
When statistics have been selected to be displayed, the MB Statistics toolbar is also displayed:

See the next section for a description of the MB Statistics toolbar.

NOTE. *In the MB Statistics toolbar, selecting the Colors displays the colors and the numerals.*

Colors When this is selected, colors are displayed for the MacroBlock statistics, overlaid onto each MacroBlock.

The colors give a quick visual indication of the values in the video; the smaller the number, the more towards blue, the larger the number, the more towards red:

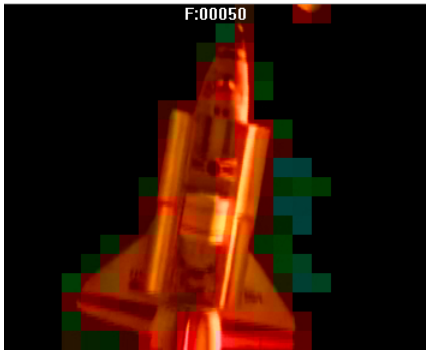


NOTE. *If it is hard to see the colors, the video can be blanked (set to mid gray) by selecting Blank video in the Overlay menu. You can also blank the video by pressing the B key when the video is selected.*

If exact numbers are required, turn on the numerals, instead of, or in addition to, the colors, or use the MacroBlock tooltip.

If it is not displayed, the MB Statistics toolbar can also be displayed from Windows > Toolbars. Or you can right-click on the menu area to display the context menu and turn MB statistics on or off.

The example below shows bits per MacroBlock in frame 50 of the MPEG-4 example file Space (no color = no bits):



Linear/logarithmic scales

Some number ranges are best represented in linear scales (such as quants and slices); others are best in logarithmic scales.

MTS4EA automatically chooses an appropriate scale and displays the values at the bottom, middle, and top of the color range.

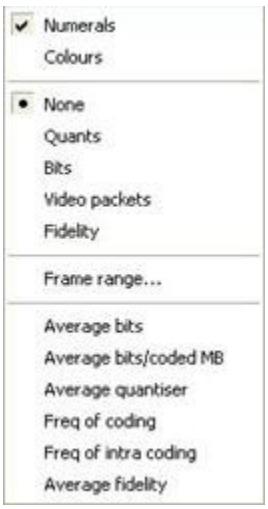
The choice of linear/logarithmic scale made by the MTS4EA is indicated by the middle value on the color key. If the value is not numerically the average of the top and bottom values, a logarithmic scale has been chosen by the MTS4EA.

Linear scale (for example, quants, slices):

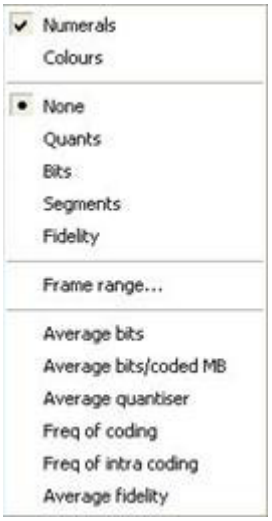
Logarithmic scale (for example, bits, average bits):

Menu variation with standard

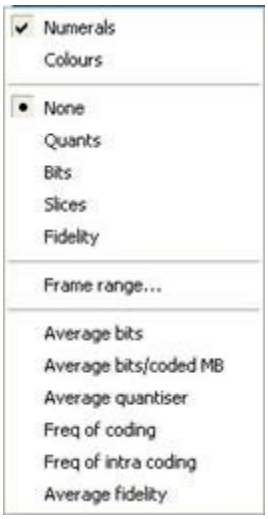
The menu below the selection of numerals/colors varies (slightly) depending on the video standard:



MPEG-4



H.263



H.264/AVC, MPEG-2

NOTE. *It can also useful to use this option with the Hold last frame option in the Play Menu, which allows the total sequence statistics to be viewed in relation to the picture content. For interlaced video streams, some of the MB statistics overlays are not visible if the video is displayed in combined view (frame view). To see the statistics in this situation, switch to the separated fields view using the interlace toolbar.* *(See page 5-295, Separated fields view.)*

The first four of the MB statistics overlays apply only to the current frame:

- Quants
- Bits
- Segments or Slices
- Fidelity

The remaining selections apply to the range of frames selected by Frame range.

- Average bits
- Average bits/coded MB
- Average quantizer
- Freq of coding
- Freq of intra coding
- Average fidelity

NOTE. *The frame range over which statistics are to be accumulated cannot be changed during pause mode; otherwise MTS4EA could not ensure that the Trace and Graph data were collected over the correct range of frames. For example, if the video was paused at frame 23 and Trace was enabled for a range of frames from 20 to 25, the Trace data would be incorrect.*

When a range of frames has been set, the Frame range item on the MB statistics menu changes, with a check mark to indicate that a range of frames has been set. (See page 5-112, *Frame range*.)

The status bar indicates whether the video frame being displayed is in or out of the selected range of frames (See page 5-114, *Frame range in/out indicator on status bar*.)

None. This option switches off any displayed overlaid MacroBlock statistics on the current frame.

NOTE. *The None option does not turn off the motion vector plot or MacroBlock type display.*

Quants. This option shows the quantizer used for the luminance for each decoded MacroBlock.

If the MacroBlock was not coded, then no number/color is displayed.

This option can be used with Motion Vectors and/or MacroBlock types, but not with any other digit overlay option.

Bits. This option overlays the number of bits used to encode each MacroBlock in the current frame.

If the MacroBlock was not coded, no bits are shown/no color is shown, even though in MPEG-4 and H.263 a single bit is sent to signify an uncoded MacroBlock.

NOTE. *Because there is only enough space to display three digits, if the MacroBlock used more than 999 bits, then +++ is displayed to signify an overflow (a number greater than 999).*

This option can be used with Motion Vectors and/or MacroBlock types, but not with any other digit overlay option.

Video packets [MPEG-4]. This option shows the grouping of the MacroBlocks into video packets. The start of a new video packet is distinguished by a resynchronization marker in the stream.

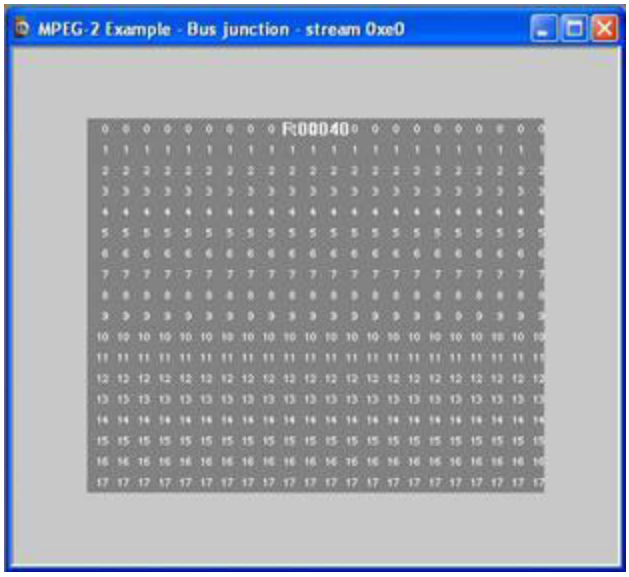
NOTE. *Video packets are only used in displays of information relating to MPEG-4.*

Segments [H.263]. This option shows the segment numbering in use for the current frame. Segments are defined as being Groups of Blocks. If a segment number is zero, then it is not displayed - hence if no GOB headers or Slices are sent, then this option will not overlay anything.

NOTE. *Segments are used only in displays of information relating to H.263.*

This option can be used with Motion Vectors and/or MacroBlock types, but not with any other digit overlay option.

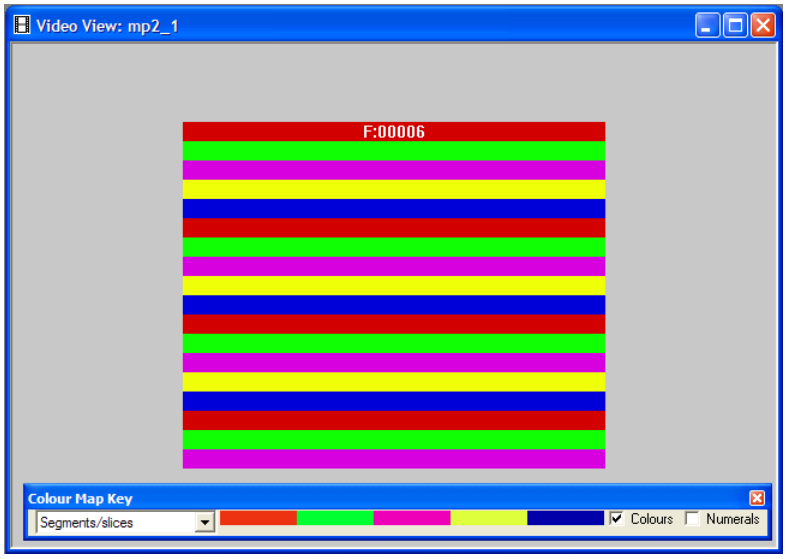
Slices [H.264/AVC, MPEG-2]. This option shows the slice numbering in use for the current frame; an example is given below:



NOTE. *Slices are used only in displays relating to H.264/AVC and MPEG-2.*

(In order to see the slice ID numbers more clearly, the video has been blanked, using the Overlay > Blank video menu selection.)

This shows the same frame, with the slices displayed using the color overlays:



The slice ID overlay option can be used with Motion Vectors and/or MacroBlock types, but not with any other digit overlay option.

Fidelity. This option shows the fidelity analysis values for each MacroBlock, for example, PSNR in dB (decibels):

NOTE. *Fidelity is grayed out until fidelity analysis is enabled. (See page 5-186, Video fidelity enable....)*

The type of fidelity analysis done is shown in the Fidelity tab of the Analysis options. (See page 5-186, *Video fidelity enable....*)

Frame range

This option allows the start and end frames to be set for the calculation of the MacroBlock statistics (the range is inclusive):

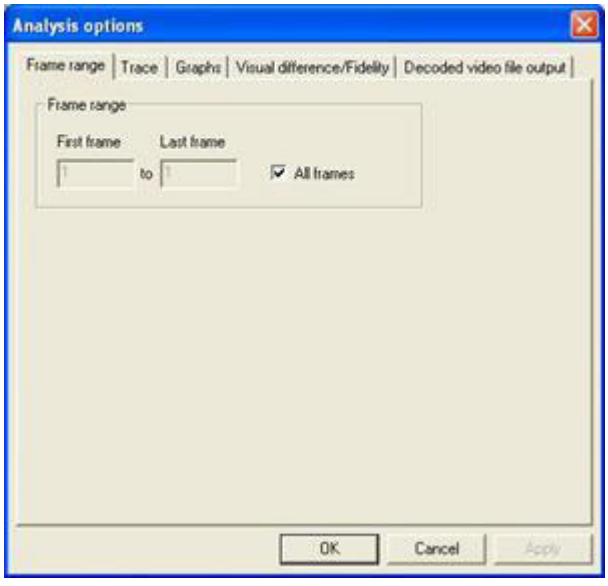
- Average bits
- Average bits/coded MB
- Average quantizer
- Freq of coding
- Freq of intra coding
- Average fidelity

NOTE. *The frame range set on this tab is the same frame range used for the Trace and Graph analysis (See page 5-234, Frame range tab.)*

When the frame range has been set, the statistics are calculated from the first to the last frame within the range (inclusive frame numbers) and the last values they reach are held on the displayed frame when outside the range.

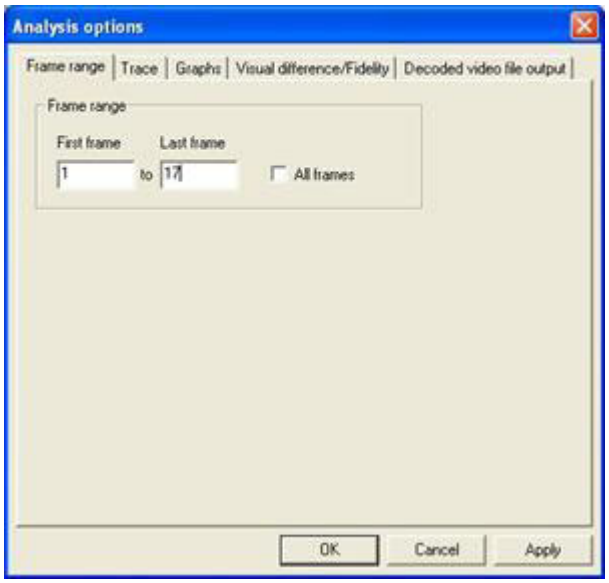
The frame range over which statistics are to be accumulated cannot be changed during pause mode; otherwise the MTS4EA could not ensure that the Trace and Graph data were collected over the correct range of frames. For example, if the video was paused at frame 23 and Trace was enabled for a range of frames from 20-25, the Trace data would be incorrect.

Clicking on Frame range opens the frame range tab:

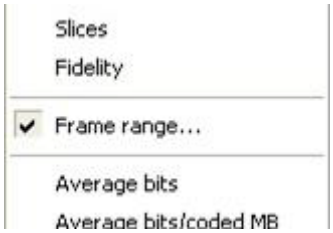


All frames. This option is the default and calculates the sequence statistics from the start of the sequence to the current frame.

To set a range of frames. If the All frames check box is cleared, then a range of frames can be entered (in this case, frames 1 to 17 inclusive):

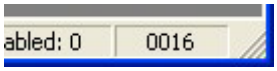


When a range of frames has been set, the Frame range item on the MB statistics menu changes, with a check mark to indicate that a range of frames has been set:



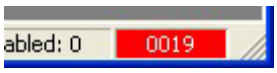
Frame range in/out indicator on status bar. The frame number indicator at the bottom right of the status bar changes when in/out of the frame range as follows:

The frame counter text is in black with a gray background:



When a frame range has been set and the displayed frame is within the specified range, the frame counter background changes to green, as shown:

When a frame range has been set but the displayed frame is outside of the range set, the frame counter changes to white text on a red background.

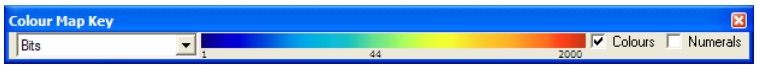


Average bits. This option displays the average number of bits used to code each MacroBlock, by dividing the total accumulated bits for each MacroBlock by the total number of frames in the frame range.

Both coded and not coded MacroBlocks are included in the average bits statistics.

NOTE. *Because there is only enough space to display three digits, if the MacroBlock used more than 999 bits, then +++ is displayed to signify an overflow (a number greater than 999).*

The color range displayed with the color overlays is a logarithmic scale:



Average bits/coded MB. This option displays the average number of bits used to code each MacroBlock, but allowing for the number of times it has been coded, the total accumulated bits used to code the MacroBlock over the frame range specified divided by the number of times that particular MacroBlock has been coded.

The color range displayed with the color overlays is a logarithmic scale: see preceding screenshot.

Average quantizer. This option displays the average quantizer used to code each MacroBlock over the specified frame range. It is displayed to one decimal place.

The color range displayed with the color overlays is a linear scale:

Freq of coding. This option displays the frequency of coding (number of times each MacroBlock has been coded over the frame range specified).

NOTE. *Because there is only enough space to display three digits, if the MacroBlock used more than 999 bits, then +++ is displayed to signify an overflow (a number greater than 999).*

Freq of intra coding. This option displays the number of times each MacroBlock has been coded in Intra Mode over the frame range specified.

Average fidelity. This option shows the average fidelity analysis values for each MacroBlock over the selected frame range.

The type of fidelity analysis done is shown in the Fidelity tab of the Analysis options. (See page 5-186, *Video fidelity enable....*)

CTU statistics

The CTU statistics option appears only for H.265/HEVC streams. For non-HEVC streams, this option is replaced by the MB statistics option.(See page 5-105, *MB statistics*.)

This option displays statistics on a CTU level for HEVC streams.

The options described in this section are also available from the CTU Statistics toolbar.

The top of this submenu allows selection of:

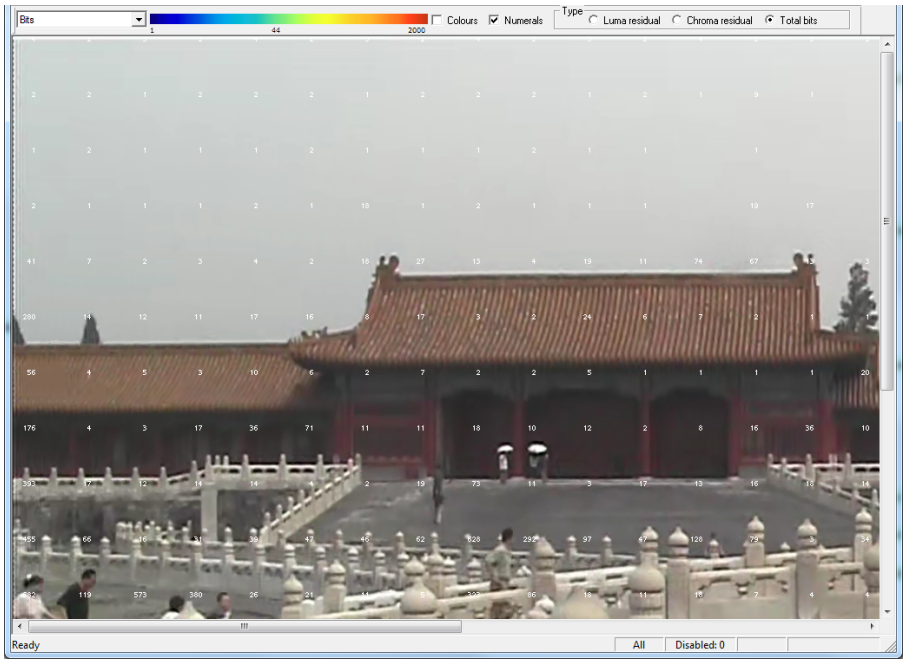
- Numerals (numbers)
- Colors
- Bits (Luma Residual, Chroma Residual, and Total)

Numerals

When this is selected, numbers are displayed for the CTU statistics, overlaid onto each CTU.

(In the preceding example, Numerals is selected, but None is also selected, so no statistics will be displayed.)

The numbers displayed are based on the statistic type selected from this submenu; the following example shows bits per CTU in the H.265 example file Forbidden city (no bits = not coded):



When statistics have been selected to be displayed, the CTU Statistics toolbar is also displayed as shown below. Use the drop-down box to select the desired CTU statistic to display.

See the next section for a description of the CTU Statistics toolbar.

NOTE. *In the CTU Statistics toolbar, selecting Colors displays the only the colors and selecting Numerals displays only the numerals. You need to select both Colors and Numerals to see both overlays.*

Colors When this is selected, colors are displayed for the CTU statistics, overlaid onto each CTU.

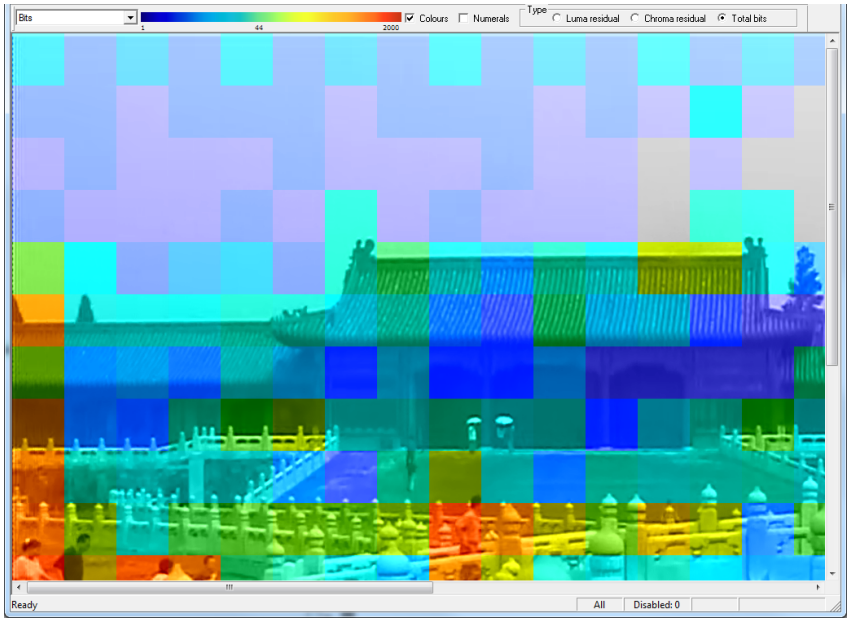
The colors give a quick visual indication of the values in the video; the smaller the number, the more towards blue, the larger the number, the more towards red:

NOTE. *If it is hard to see the colors, the video can be blanked (set to mid gray) by selecting Blank video in the Overlay menu. You can also blank the video by pressing the B key when the video is selected.*

If exact numbers are required, turn on the numerals, instead of, or in addition to, the colors, or use the CTU tooltip. Note that the CTU tooltip does not display all of the parameters present in the CTU statistics.

If it is not displayed, the CTU Statistics toolbar can also be displayed from Windows > Toolbars, or you can right-click on the menu area to display the context menu and turn CTU statistics on or off.

The example below shows bits per CTU in the H.265 example file Forbidden city (no color = no bits):



Bits

Bit related fields in the CTU statistics, for example the bits, average bits, and the average bits/coded CTU fields, are provided with the option to select which bit values are displayed: luma, chroma, or total bits. The bits type selection filter is available only for H.265/HEVC streams.

- **Luma residual:** Overlays the luma bit values used to encode the CTU on the video frame.
- **Chroma residual:** Overlays the chroma bit values used to encode the CTU on the video frame.
- **Total bits:** Overlays the sum of the luma, chroma, and header bits used to encode the CTU on the video frame.

Linear/logarithmic scales

Some number ranges are best represented in linear scales (such as quants and slices); others are best in logarithmic scales.

MTS4EA automatically chooses an appropriate scale and displays the values at the bottom, middle, and top of the color range.

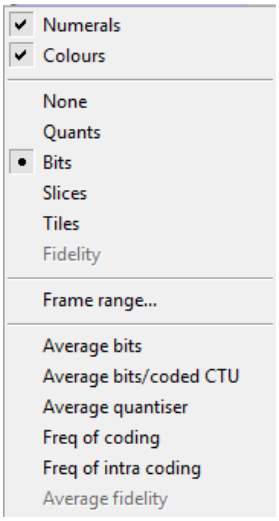
The choice of linear/logarithmic scale made by the MTS4EA is indicated by the middle value on the color key. If the value is not numerically the average of the top and bottom values, a logarithmic scale has been chosen by the MTS4EA.

Linear scale (for example, quants, slices):

Logarithmic scale (for example, bits, average bits):

CTU statistics menu

The menu below the selection of numerals/colors varies (slightly) depending on the video standard. The following illustration shows the menu for H.265/HEVC streams.



H.265/HEVC

The first five of the CTU statistics overlays apply only to the current frame:

- Quants
- Bits
- Segments or Slices
- Tiles
- Fidelity

The remaining selections apply to the range of frames selected by Frame range.

- Average bits
- Average bits/coded CTU
- Average quantizer
- Freq of coding
- Freq of intra coding
- Average fidelity

NOTE. *The frame range over which statistics are to be accumulated cannot be changed during pause mode; otherwise the MTS4EA could not ensure that the Trace and Graph data were collected over the correct range of frames. For example, if the video was paused at frame 23 and Trace was enabled for a range of frames from 20 to 25, the Trace data would be incorrect.*

When a range of frames has been set, the Frame range item on the CTU statistics menu changes, with a check mark to indicate that a range of frames has been set. (See page 5-123, *Frame range*.)

The status bar indicates whether the video frame being displayed is in or out of the selected range of frames (See page 5-125, *Frame range in/out indicator on status bar*.)

None. This option switches off any displayed overlaid CTU statistics on the current frame.

NOTE. *The None option does not turn off the motion vector plot or CTU type display.*

Quants. This option shows the quantizer used for the luminance of each decoded CTU (Quants value is available at the CU level; averaging the CU quants to get the CTU level quantizer value).

If the CTU was not coded, then no number/color is displayed. A CTU is considered not coded only if all of the CUs in the CTU are not coded.

This option can be used with Motion Vectors and/or CTU types, but not with any other digit overlay option.

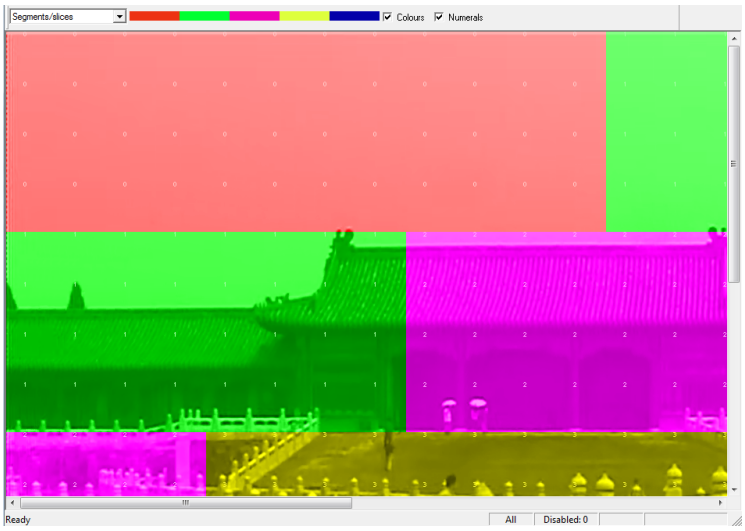
Bits. This option overlays the number of bits used to encode each Coding Tree Unit (CTU) in the current frame.

If the CTU was not coded, no bits are shown/no color is shown, even though in H.265 a single bit is sent to signify an uncoded CTU.

NOTE. *For a 16×16 CTU, because of space limitations, the value is limited to 3 digits. In the case of 32×32 and 64×64 CTUs, the number of digits displayed is increased to 5.*

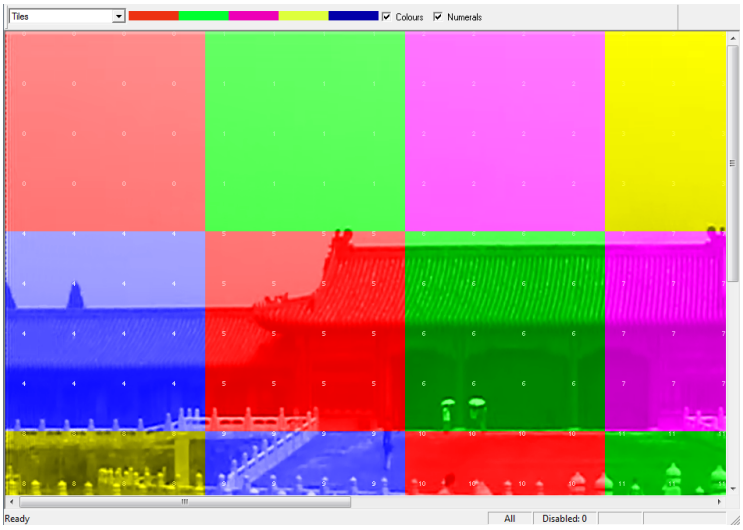
This option can be used with Motion Vectors and/or CTU types, but not with any other digit overlay option.

Slices [H.265/HEVC]. This option shows the slice numbering in use for the current frame; an example is given below with the slices displayed using the color overlays:



The slice ID overlay option can be used with Motion Vectors and/or CTU types, but not with any other digit overlay option.

Tiles [H.265/HEVC]. This option shows the tile numbering in use for the current frame.



Fidelity. This option shows the fidelity analysis values for each CTU, for example, PSNR in dB (decibels):

NOTE. *Fidelity is grayed out until fidelity analysis is enabled. (See page 5-186, Video fidelity enable....)*

The type of fidelity analysis done is shown in the Fidelity tab of the Analysis options. (See page 5-186, *Video fidelity enable....*)

Frame range

This option allows the start and end frames to be set for the calculation of the CTU statistics (the range is inclusive):

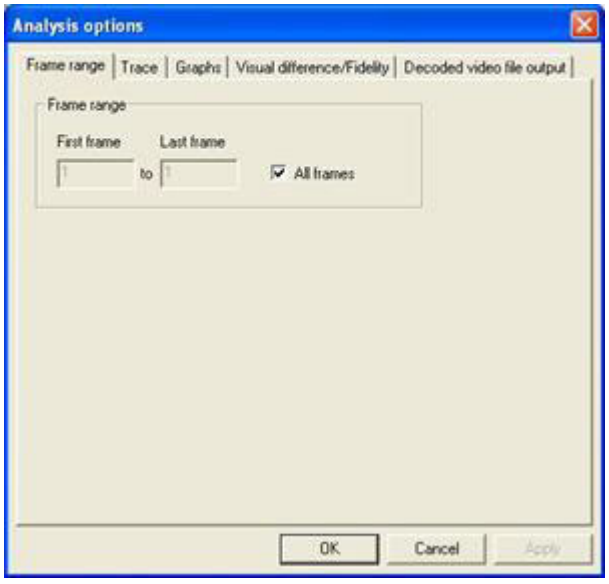
- Average bits
- Average bits/coded CTU
- Average quantizer
- Freq of coding
- Freq of intra coding
- Average fidelity

NOTE. *The frame range set on this tab is the same frame range used for the Trace and Graph analysis (See page 5-234, Frame range tab.)*

When the frame range has been set, the statistics are calculated from the first to the last frame within the range (inclusive frame numbers) and the last values they reach are held on the displayed frame when outside the range.

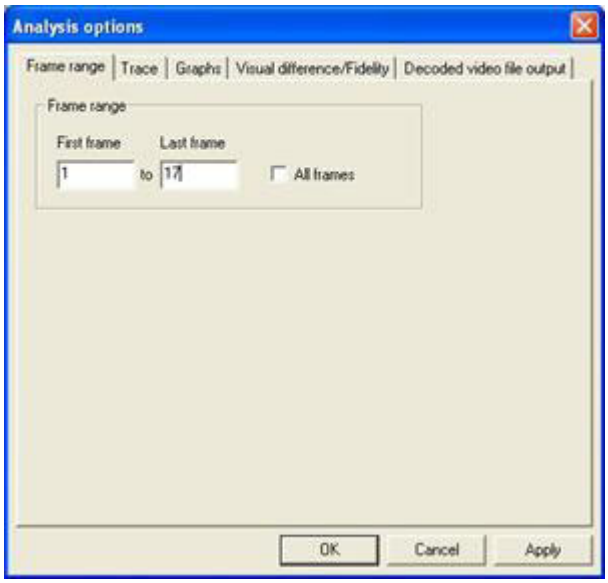
The frame range over which statistics are to be accumulated cannot be changed during pause mode; otherwise the MTS4EA could not ensure that the Trace and Graph data were collected over the correct range of frames. For example, if the video was paused at frame 23 and Trace was enabled for a range of frames from 20-25, the Trace data would be incorrect.

Clicking on Frame range opens the frame range tab:

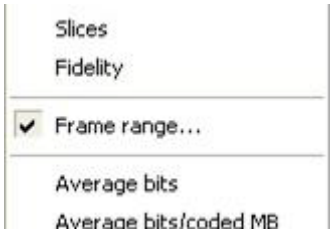


All frames. This option is the default and calculates the sequence statistics from the start of the sequence to the current frame.

To set a range of frames. If the All frames check box is cleared, then a range of frames can be entered (in this case, frames 1 to 17 inclusive):

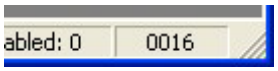


When a range of frames has been set, the Frame range item on the CTU statistics menu changes, with a check mark to indicate that a range of frames has been set:



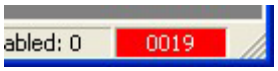
Frame range in/out indicator on status bar. The frame number indicator at the bottom right of the status bar changes when in/out of the frame range as follows:

The frame counter text is in black with a gray background:



When a frame range has been set and the displayed frame is within the specified range, the frame counter background changes to green, as shown:

When a frame range has been set but the displayed frame is outside of the range set, the frame counter changes to white text on a red background.



Average bits. This option displays the average number of bits used to code each CTU, by dividing the total accumulated bits for each CTU by the total number of frames in the frame range.

Both coded and not coded CTUs are included in the average bits statistics.

NOTE. *For a 16×16 CTU, because of space limitations, the value is limited to 3 digits. In the case of 32×32 and 64×64 CTUs, the number of digits displayed is increased to 5.*

The color range displayed with the color overlays is a logarithmic scale:

Average bits/coded CTU. This option displays the average number of bits used to code each CTU, but allowing for the number of times it has been coded, the total accumulated bits used to code the MacroBlock over the frame range specified divided by the number of times that particular MacroBlock has been coded.

The color range displayed with the color overlays is a logarithmic scale: see preceding screenshot.

Average quantizer. This option displays the average quantizer used to code each CTU over the specified frame range. It is displayed to one decimal place.

The color range displayed with the color overlays is a linear scale:

Freq of coding. This option displays the frequency of coding (number of times each CTU has been coded over the frame range specified).

Freq of intra coding. This option displays the number of times each CTU has been coded in Intra Mode over the frame range specified.

Average fidelity. This option shows the average fidelity analysis values for each CTU over the selected frame range.

The type of fidelity analysis done is shown in the Fidelity tab of the Analysis options. (See page 5-186, *Video fidelity enable...*.)

Overlay color Ctrl+W

This item allows the selection of the color of the overlays. You can see the values even if the decoded image is very dark (choose white) or very light (choose black). It affects all the number overlays and the motion vector overlay.

In the motion vector overlay, two colors may be used for some video standards - in this case:

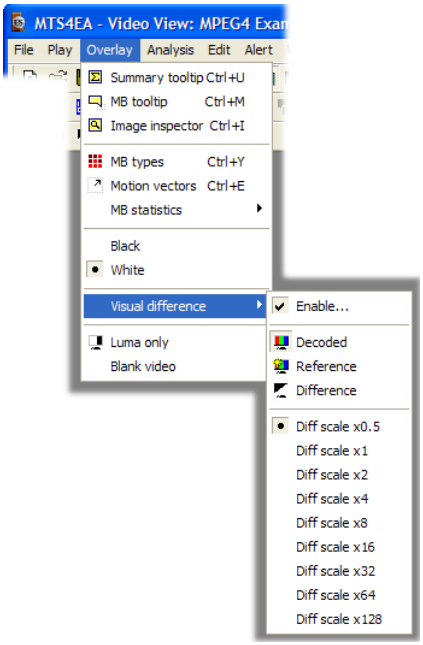
- Black selects black and dark blue colors for overlays
- White selects white and light blue colors for overlays

(See page 5-101, *Motion vectors Ctrl+E*.)

Black. This selects a black/dark blue color for the overlays.

White. This selects a white/light blue color for the overlays.

Visual difference



This menu option enables the visual difference video display; this uses an uncompressed video reference file to show a video view of any of the following:

- The encoded (compressed) bitstream
- The uncompressed video reference bitstream
- The visual difference between the encoded bitstream and the uncompressed video reference file

NOTE. *The visual difference display can be done on a range of frames only, if this range is at the start of a file. For example, only the first ten frames of the YUV reference file are provided for the H.264/AVC and MPEG-2 bitstreams Grenadier Guards.*

When displaying the compressed bitstream or the uncompressed video reference file or the difference between the two, all the standard MTS4EA video playing controls can be used to play the video forward/backward (although some of the unrelated seek functions be grayed out).

Visual difference icon toolbar

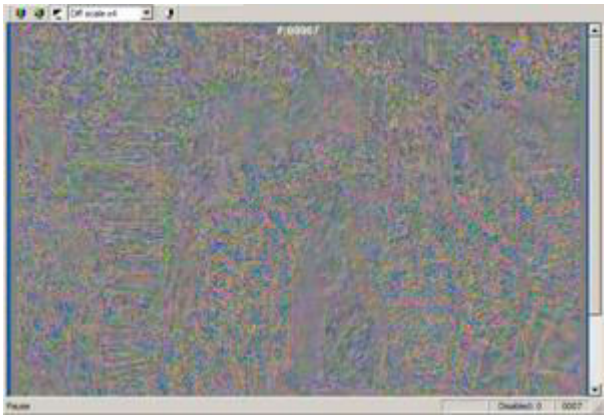
This toolbar is displayed when the visual difference view is enabled:

The functions of the icons are as follows:

Show encoded (compressed). Selecting this menu option (or clicking on the icon) displays the standard video window (shows the compressed bitstream that was decoded by the MTS4EA):

Show uncompressed video reference. Selecting this menu option (or clicking on the icon) displays the frame in the YUV reference file that corresponds most closely in time to the corresponding frame in the compressed bitstream:

Show difference. Selecting this menu option (or clicking on the icon) subtracts the uncompressed video reference video from the compressed video, frame-by-frame:



The MTS4EA uses the timing given in the:

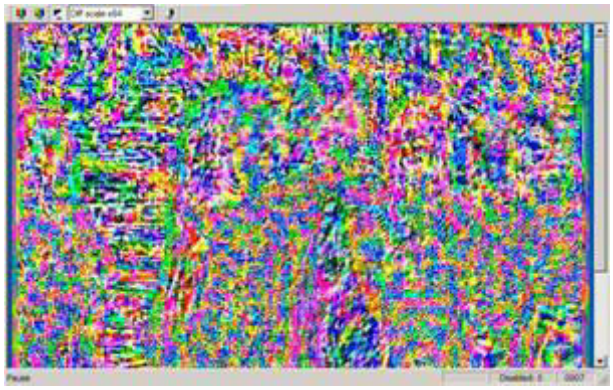
- Compressed bitstream itself for the displayed frame times of the compressed bitstream
- Uncompressed video reference set-up tab of the MTS4EA (the frame rate) for the uncompressed video reference file

When doing the subtraction, the MTS4EA uses the uncompressed video reference file and the corresponding frames from the compressed bitstream that are closest to each other in time.

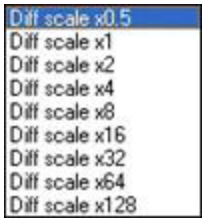
NOTE. *The visual difference view can be used to display the difference between two uncompressed video files; to do this, open the first uncompressed video file using the standard File-open (selecting Open as type) and then select the second uncompressed video as the reference file.*

If the frame rate of the uncompressed video reference file is set incorrectly, the visual differences will invariably appear to be much greater than they should be.

Magnify visual difference by X. This menu selection or drop-down menu can be used to magnify the visual differences on-screen, so that they are easier to see:



The magnification (multiplication factor) can be any of the factors shown:



Luma only

This displays the luma (luminance) only of the compressed bitstream, or uncompressed video reference, or difference.

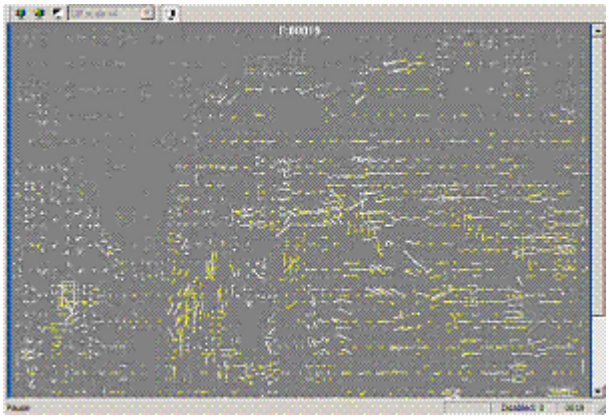
When this option is selected ( ), only the luminance of the image in the video window is shown:

Luma only not selected

Luma only selected

Blank video

This option will blank the video output to a gray color. It is included so that the motion vectors and other data can be viewed more clearly without the visual interference of the decoded image.



In this example, the motion vector overlay is on and the video is blanked. You can also blank the video by pressing the B key when the video is selected.

NOTE. *The Blank video option is available only when one of the overlays on the Overlay menu is in use; otherwise it is grayed out.*

Audio options

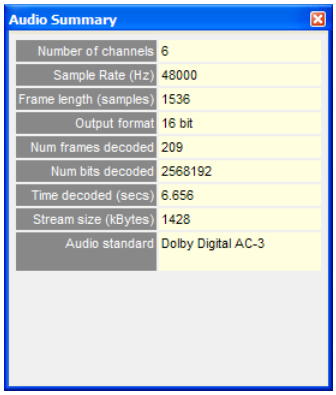
These options are only available when an audio waveform is being viewed.

NOTE. *Full audio analysis is not currently supported for the Dolby Digital (AC-3) standard. For this standard, only audio extraction and playback will be available.*

The audio summary and frame tooltips can be manipulated in the same way as the video tooltips. (See page 5-64, *Summary & MacroBlock tooltips manipulation*.)

Audio summary tooltip

When this option is selected, a window is displayed that provides summary information about the audio frame being displayed or the sequence as a whole. The information displayed differs depending on whether the stream is playing/paused or is in stop mode. The selection of information provided by the audio summary tooltip is as follows:



Number of channels — If the audio stream is mono, this number will be one (1); if the stream is stereo, this number will be two or more.

Sample rate (Hz) — Sample rate of the decoded audio in samples per second (Hz).

Frame length (samples) — Number of audio samples in each decoded audio frame.

Output format — The output audio sample format bit depth in bits per sample.

Num frames decoded — The running total of decoded frames.

Total frames The total number of decoded audio frames in the stream.

Time decoded (secs) The time in seconds from the start to the current point in the sequence, calculated as if the sequence were playing normally (the MTS4EA keeps track of the playing time, irrespective of whether the sequence is paused at some point or played in fast forward mode).

Total time (secs) The duration of the stream in seconds.

Ave. bit rate (kbits/s) This is total bits (kbits) divided by total time (secs).

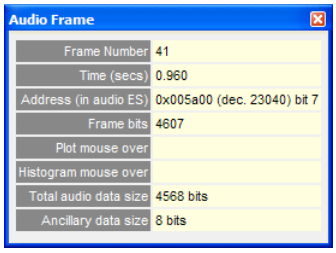
Ave. frame rate (Hz) This is total frames divided by total time (secs).

Stream size (kBytes) The size of the stream in KB.

Audio standard The audio standard used.

Audio frame tooltip

This tooltip displays information about a selected audio frame. The information provided by the frame tooltip varies according to the audio standard being analyzed and the options available for that standard.



Frame number The number of the currently selected audio frame.

Time (secs) The time in seconds from the start of the audio sequence to the start of the selected audio frame, calculated as if the sequence were playing normally (MTS4EA keeps track of the playing time, irrespective of whether the sequence is paused at some point or played in fast forward mode).

Address (in audio ES) The address of the beginning of the selected audio frame within the audio elementary stream.

Frame bits The number of bits within the selected audio frame.

Plot mouse over When the mouse pointer is over an audio plot, this field shows the name of the audio plot and its value for the selected audio frame.

Histogram mouse over When the mouse pointer is over an audio histogram, this field shows the name of the part of the audio histogram the mouse is over and its value for the selected audio frame.

Total audio data size [MPEG-1] The total number of bits used for the audio data part of the selected frame. This consists of all the data that decodes to produce the audio frame, and does not include other data such as the header, error check, or ancillary data.

Ancillary data size [MPEG-1] The total number of bits used for the ancillary data part of the selected audio frame.

Total SBR data size [AAC] The total number of bits used for the SBR data part of the selected audio frame.

Total config data size [AAC] The total number of bits used for the configuration data part of the selected audio frame. This includes all data used to configure the AAC decoder (for example, ADTS headers and AudioSpecificConfig data.)

Total data size [AAC] The total number of bits used to represent the main and subsidiary payload structures of the selected audio frame.

Max scalefactor blocks [AAC] The maximum number of scalefactor bands per group for the selected audio frame.

Global gain [AAC] Global gain, equal to the value of the first scalefactor, for the selected audio frame.

LTP data present [AAC] Flag to specify whether LTP data is present in the selected audio frame.

Predictor data present [AAC] Flag to specify whether predictor data is present in the selected audio frame.

MS mask present [AAC] Flag to specify whether MS mask is present in the selected audio frame.

Pulse data present [AAC] Flag to specify whether pulse data is present in the selected audio frame.

TNS data present [AAC] Flag to specify whether TNS data is present in the selected audio frame.

Gain control data present [AAC] Flag to specify whether gain control data is present in the selected audio frame.

Window sequence [AAC] Shows the type of window sequence used in the selected audio frame.

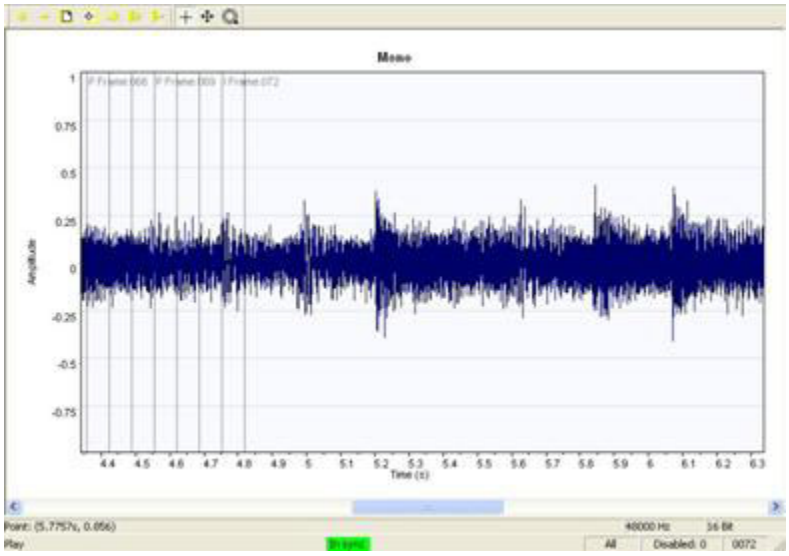
Window shape [AAC] Shows the type of window shape used in the selected audio frame.

Show audio features... A range of options showing audio related data.

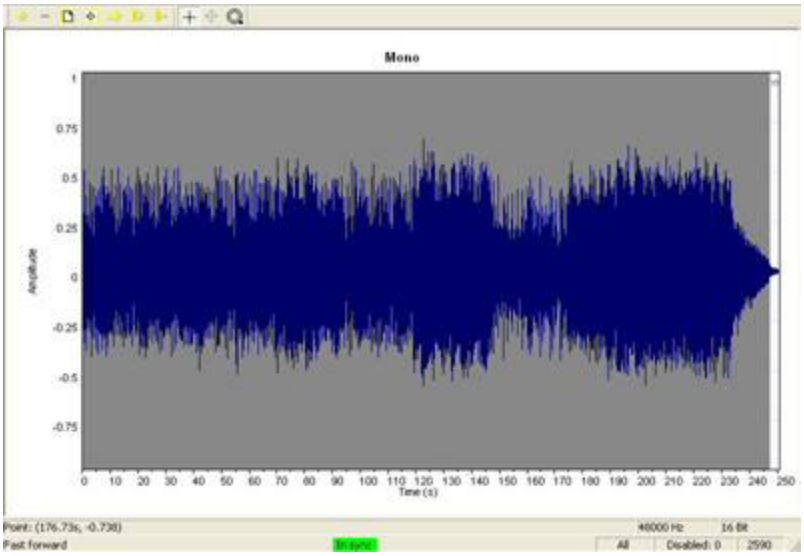
Show audio waveform

MTS4EA offers the ability to view an audio streams waveform: the y-axis is normalized amplitude, and the x-axis is time (seconds).

The audio streams waveform can be viewed while the stream is playing, showing the frames being marked off on vertical bars as the stream is decoded:

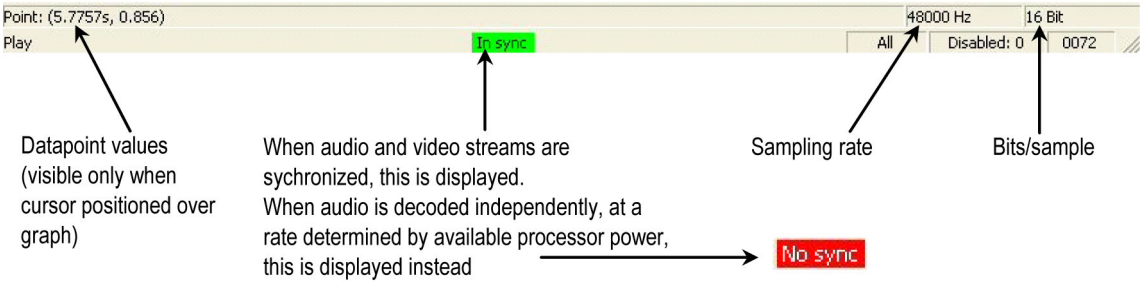


It can also be viewed when the stream has been played through to its end:

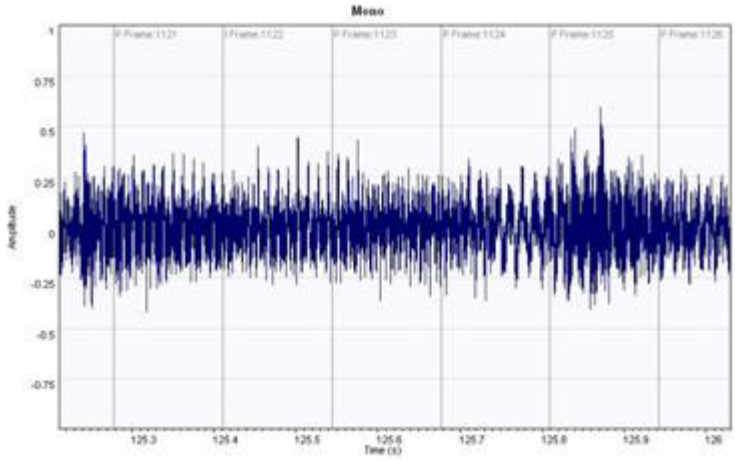


The gray background is due to the density of marked-off frames; when the zoom out icon is used, these will become more spaced and the shape of the waveform more easily discernible (see the following figure). At each frame marker, the frame type and number is shown.

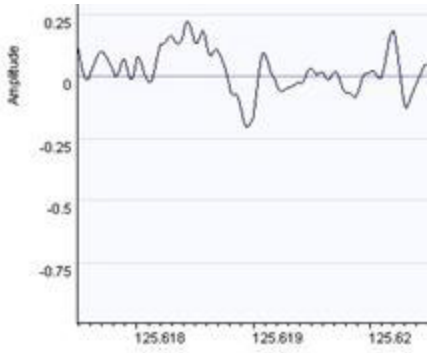
Additional information can be found in the Audio waveform views status bar:

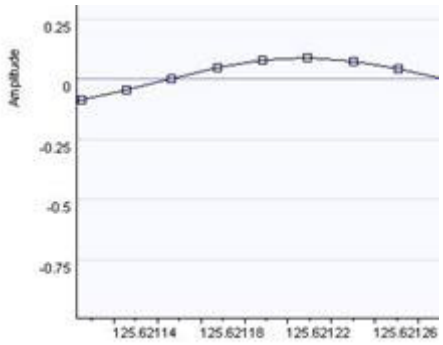


It is possible to zoom in and out on this graph view using the first two icons on the waveform view toolbar, as shown in the following screenshot.



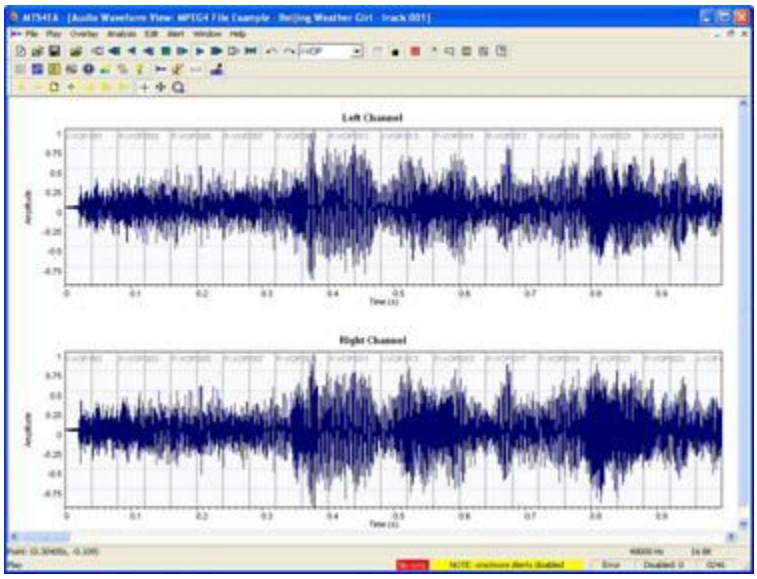
The illustrations below show that it is possible to zoom in so far as to see the individual data points plotted in the audio waveform.





The functions of the other icons are described under *Audio Waveform Toolbar icons*. (See page 5-140.)

The following screenshot shows the waveform of a stream with multiple channels:



Audio waveform toolbar icons

Icon	Function
	Zoom in (+) and zoom out (-) centered on the window (affects scale of x-axis only).
	Fit all data into window.
	Locate origin (zero), for example, start of sequence.
	Autoscroll, for example, fill the analysis data in real-time as the audio is decoded, and scroll the window to the right.
	Increase channel height (affects scale of y-axis only).
	Decrease channel height (affects scale of y-axis only).
	Measure the data at the cursor. Values are reported in status bar. Hold and drag to measure offsets and angles/slopes of lines.
	Scroll/pan (the cursor changes to show the scroll/pan direction).
	Zoom in/out centered on the cursor. Press <Shift> to zoom out.
	Show or hide audio frame markers.
	Show or hide the plots and plot key. A full account of analysis with the plot tool is given later in this section. (See page 5-145, *Show plot*.)
	Show or hide the selected histogram and associated key. A full account of histographical analysis options is given later in this section. (See page 5-141, *Show histogram*.)
	Show or hide the audio waveform.
	Show or hide the audio/video synchronisation markers.

Show audio frame markers

( ) Shows/hides the audio frame markers.

Show sync markers

( ) Shows/hides the audio synchronization markers.

Show histogram

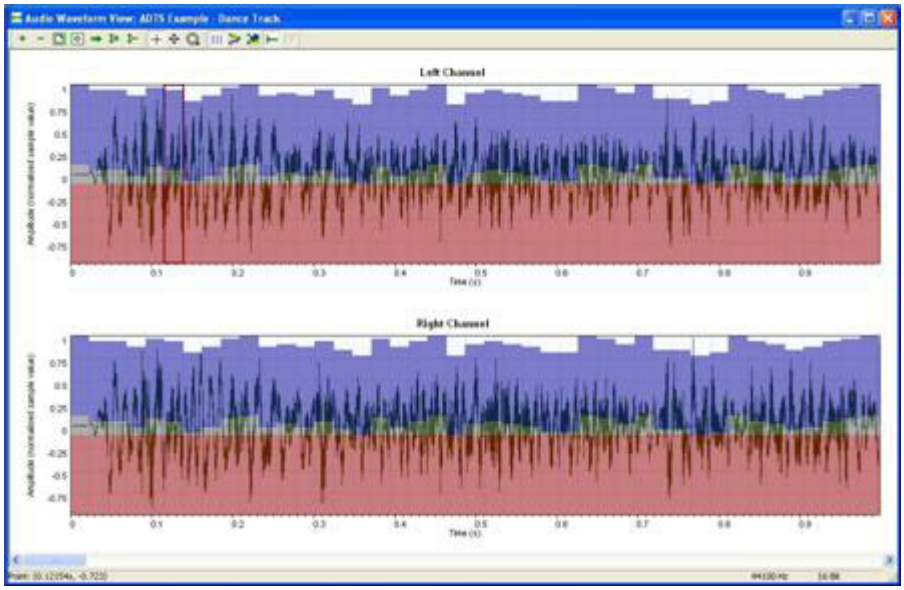
Shows/hides the distribution of different types of coded data within each audio frame. The available histograms depend on the audio standard in use. Only one type of histogram can be overlaid onto the audio waveform at any one time. The following histograms are available:

Audio data bit usage histogram [MPEG-1/2]

Shows the distribution of bits used for the following audio data syntax elements within each frame:

- allocation
- sample
- samplecode
- scalefactor
- scfsi

(See page 4-1, *Compression standards and file types*.)



Frame data bit usage histogram [MPEG-1/2]

Shows the distribution of bits used for the following syntactic structures within each frame:

- AncillaryData
- AudioData
- AudioSequence
- ErrorCheck
- Frame
- Header
- HuffmanCodeBits
- MainData

(See page 4-1, *Compression standards and file types*.)

Frame bit usage histogram [AAC]

Shows the distribution of bits used for the following top-level categories of data within each frame:

- Config
- Main payload
- SBR (Spectral Band Replication)
- Subsidiary payload

(See page 4-1, *Compression standards and file types*.)

Main payload bit usage histogram [AAC]

Shows the distribution of bits used for the following syntax elements within the main payload data of each frame:

- channel_pair_element
- coupling_channel_element
- data_stream_element
- fill_element
- gain_control_data
- ics_info
- lfe_channel_element
- program_config_element
- pulse_data
- raw_data_block
- single_channel_element

(See page 4-1, *Compression standards and file types*.)

SBR bit usage histogram [AAC]

Shows the distribution of bits used for the following syntax elements within the SBR data of each frame:

- sbr_channel_pair_base_element
- sbr_channel_pair_element
- sbr_channel_pair_enhance_element
- sbr_data
- sbr_dtdf
- sbr_envelope
- sbr_extension
- sbr_grid
- sbr_header
- sbr_invf
- sbr_noise
- sbr_single_channel_element
- sbr_sinusoidal_coding
- Unused SBR data when no header has been found

(See page 4-1, *Compression standards and file types*.)

Subsidiary payload bit usage histogram [AAC]

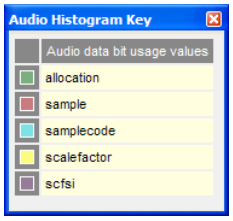
Shows the distribution of bits used for the following syntax elements within the subsidiary payload data of each frame:

- dynamic_range_info
- extension_payload
- individual_channel_stream
- ltp_data
- reordered_spectral_data
- scale_factor_data
- section_data
- spectral_data
- tns_data

(See page 4-1, *Compression standards and file types*.)

Show histogram key

Shows/hides the currently selected histogram and the associated key; the first available histogram is displayed if there is no current selection. The fields listed in the key will vary according to which histogram is displayed.



If a key is open when the histogram overlay is changed, the key will be automatically updated.

Show plot

This menu presents you with a choice of different line chart overlays. The available line plots will depend on the specific audio standard in use. Any combination of the available plots can be plotted at any one time. The plots available for each standard are as follows:

Total frame size [All standards]: The total number of bits used for all data of each audio frame.

Total audio data size [MPEG-1/2]: The total number of bits used for the audio data part of each frame. This consists of all the data that decodes to produce the audio frame, and does not include other data such as the header, error check and ancillary data.

Ancillary data size [MPEG-1/2]: The total number of bits used for the ancillary data part of each audio frame.

Total SBR data size [AAC]: The total number of bits used for the SBR data part of each audio frame.

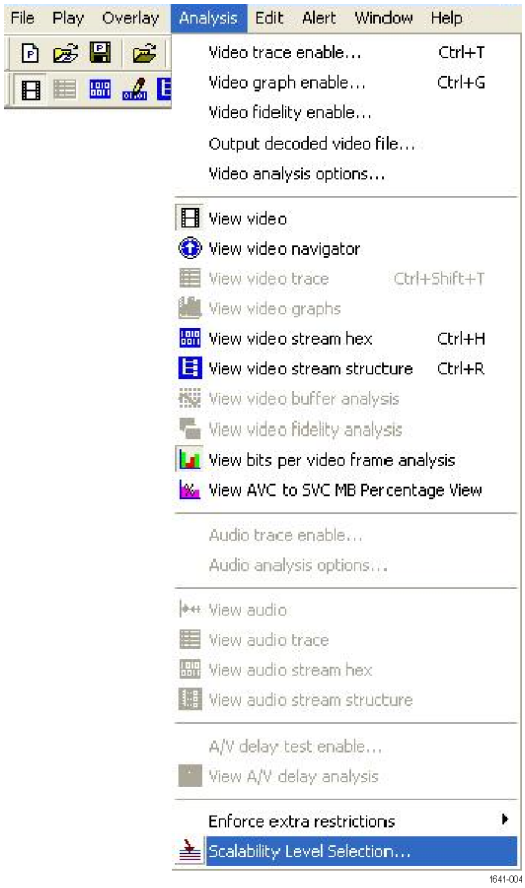
Total config data size [AAC]: The total number of bits used for the configuration data part of each audio frame. This includes all data used to configure the AAC decoder (for example, ADTS headers, AudioSpecificConfig data).

Total data size [AAC]: The total number of bits used to represent the main and subsidiary payload structures of each audio frame.

Max scalefactor blocks [AAC]: The maximum number of scalefactor bands per group for each audio frame.

Global gain [AAC]: Global gain, equal to the value of the first scalefactor, for each audio frame.

Analysis menu



This menu controls the collection and display of:

- Trace information, such as Trace/Parse bitstream and Trace/Interpret
- Graph information
- Fidelity analysis
- The selected layer for SVC streams

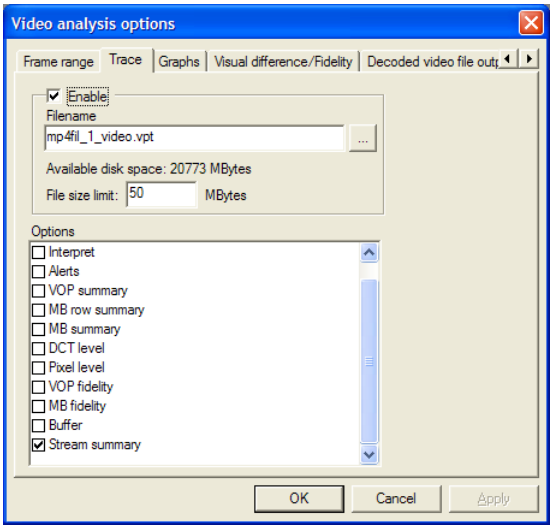
and the display of:

- Hex (hexadecimal) view of stream data
- The structure of the file being analyzed
- Analysis of the video buffer usage

The Analysis options menu item provides access to all the Trace, Graph and Fidelity options as well as setting of the range of frames over which to collect the Trace, Graph and Fidelity information.

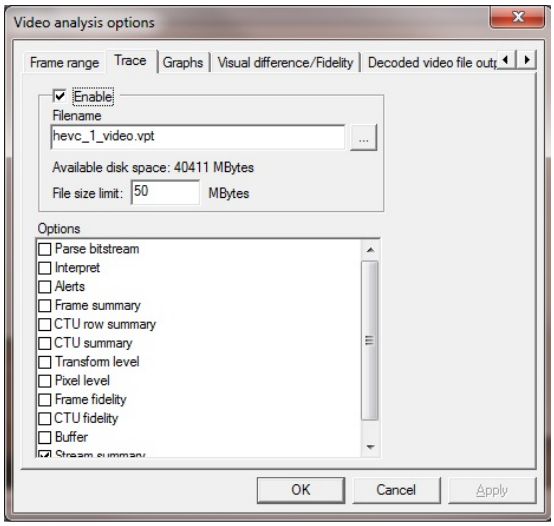
NOTE. *Trace enable, Graph enable, Fidelity enable, Decoded file output enable, Scalability Level Selection, and Analysis options cannot be set in pause mode - the video must be stopped. Otherwise, MTS4EA cannot ensure that the Trace, Graph, and Fidelity data is collected over the correct range of frames.*

Video trace enable... Ctrl+T

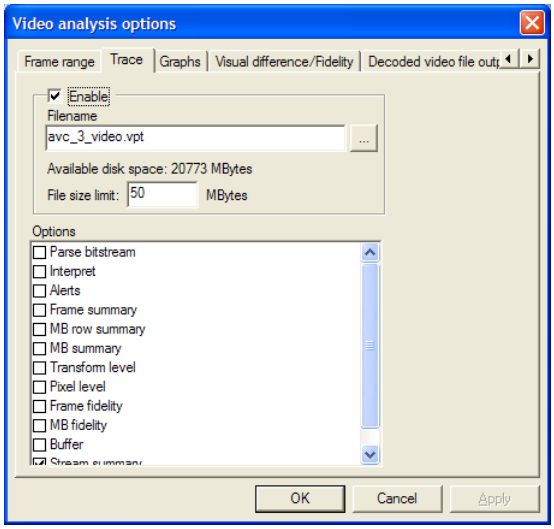


The Trace options provide text outputs that describe the contents of an encoded video stream, to various levels of detail.

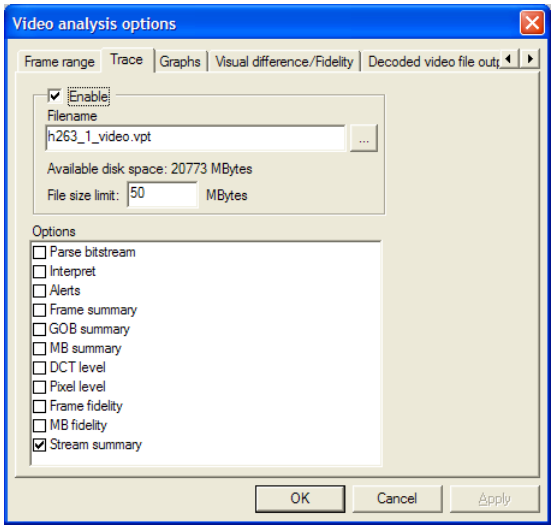
The available Trace options vary with each video standard; the screenshot above is for MPEG-4; the screenshots below for H.264/AVC and H.263 respectively.



H.265/HEVC



H.264/AVC



H.263

NOTE. *Using these options can generate a lot of data - files hundreds of MB or more in size, so it is often advisable to collect the data on a range of frames, not all (this is a selectable option).*

Alternatively, you can set a maximum file size, after which no more data is output to the trace file. A message is displayed if the specified file size is exceeded. (See page 5-153, *File size limit (available disk space)*.)

NOTE. *The video can be stopped and the Trace file opened immediately after going past the last frame number in the frame range – there is no need to wait until the end of the video sequence.*

To set the range of frames over which to collect the Trace data, click on the Frame range tab.

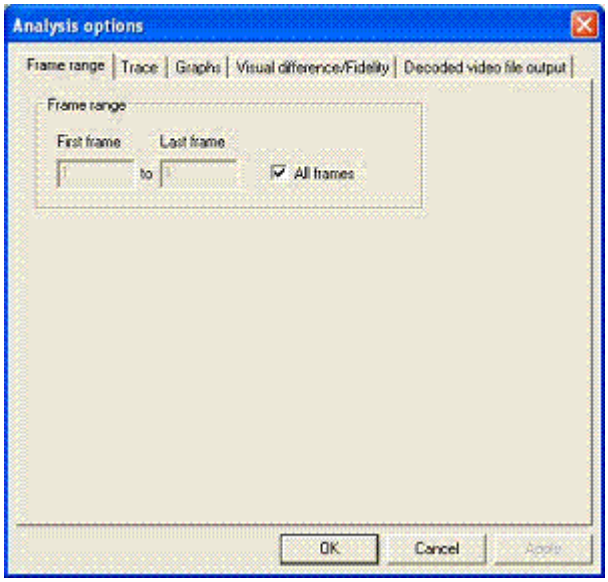
Frame range tab

This option allows you to specify the start and end frames between which to gather the Trace and Graphs information.

NOTE. *The frame range set on this tab is the same frame range used for the collection of MacroBlock statistics, where these are collected over a range of frames.* *(See page 5-112, Frame range.)*

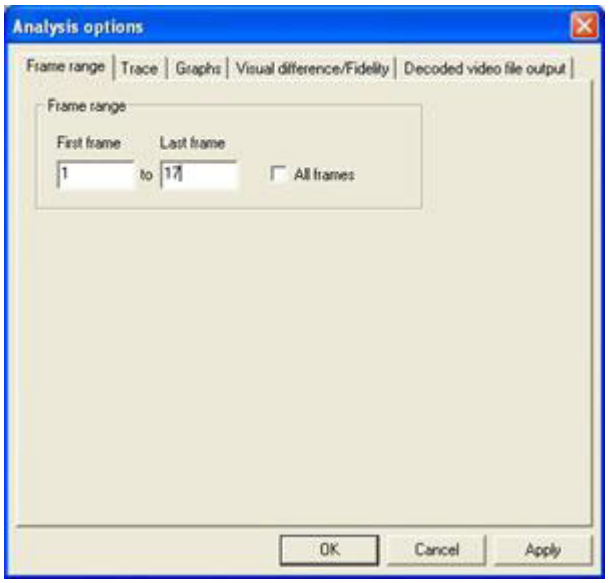
The frame range over which statistics are to be accumulated cannot be changed during pause mode; otherwise the MTS4EA could not ensure that the Trace and Graph data were collected over the correct range of frames. For example, if the video was paused at frame 23 and Trace was enabled for a range of frames from 20-25, the Trace data would be incorrect.

Clicking on Frame range takes you to the frame range tab:

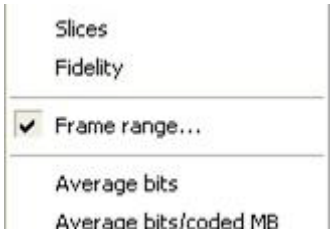


All frames. This option is the default and calculates the sequence statistics from the start of the sequence until the current frame.

To set a range of frames. If the All frames check box is cleared, then a range of frames can be entered (in this case, frames 1 to 17 inclusive):

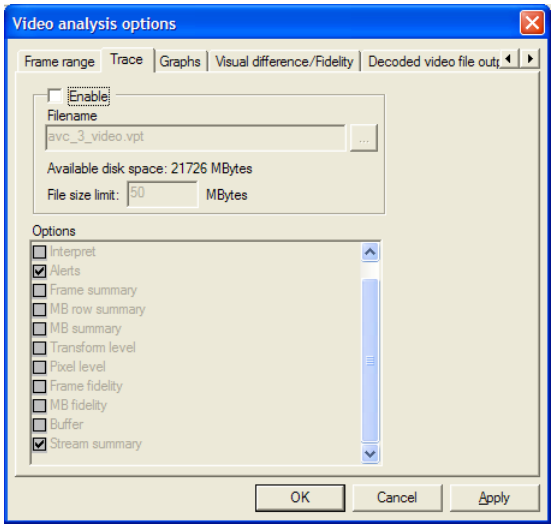


When a range of frames has been set, the Frame range item on the MB statistics menu changes, with a check mark to indicate a range of frames has been set:



Trace tab

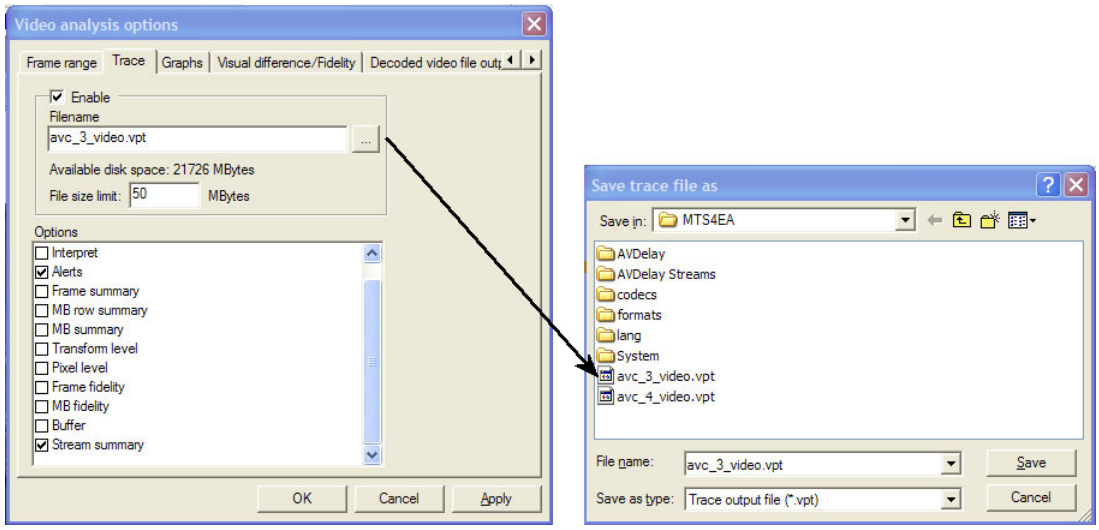
Enable. This option Enables/Disables the Trace output. If the output is disabled, the remaining selections on the Trace tab are unavailable (grayed out) although the values are stored during the current use of the MTS4EA. (They are reset to their default values the next time the MTS4EA is started.)



Filename. By default, the suggested filename is the same as the base name of the input video file (the video filename without the extension).

Any other filename can be entered; the MTS4EA will append a `.vpt` extension to this filename if it does not already end in this. (.vpt = MTS4EA trace).

To browse to a specific folder and enter the filename there, click on the [...] button:



File size limit (available disk space). Trace files can generate a lot of data: for example, 2-3 MB per frame for the Parse bitstream and Interpret outputs. This means that collecting data over many frames can easily generate a Trace file hundreds of MB in size.

This entry allows you to limit the amount of disk space taken by the Trace file.

Once the Trace file reaches this size, it stops writing more data.

Parse Bitstream. If this option is selected, then the bitstream is parsed to a file. This gives the individual bit patterns for the bitstream fields as well as the mnemonic used within the standard to identify the field.

NOTE. *The data from the output of Parse Bitstream is one of the best ways that the MTS4EA provides of doing bitstream syntax debugging.*

See General codes used in Trace files and Alerts in the Bitstream Syntax Debugging section for detailed information on the meaning of these. (See page 6-1.)

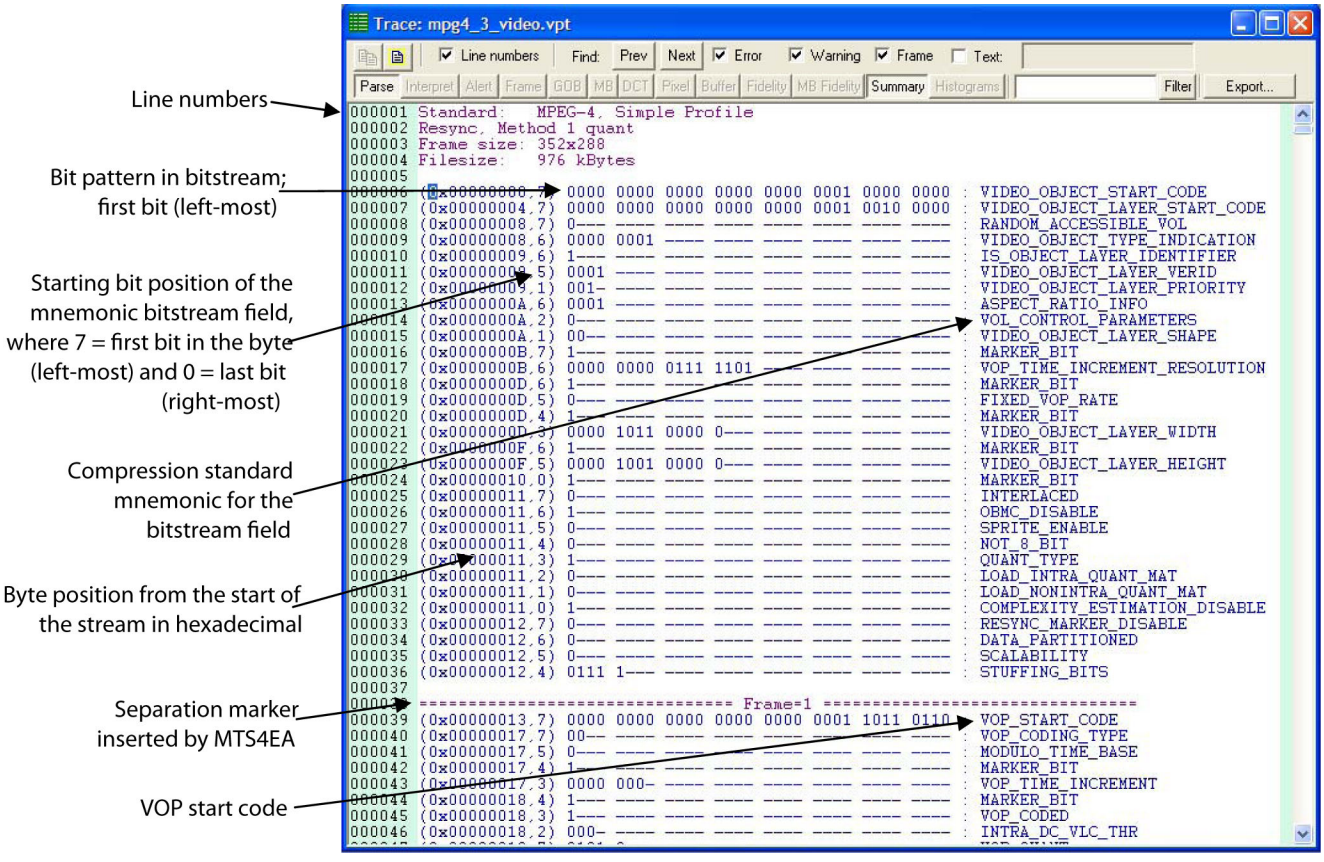


Figure 5-3: MPEG-4 trace parse example

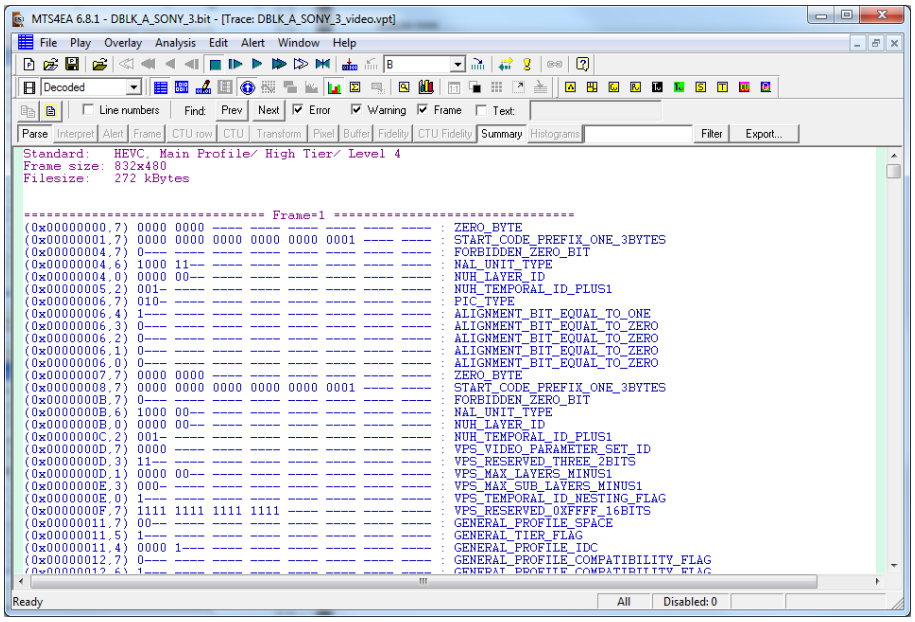


Figure 5-4: H.265/HEVC example, at start of byte stream (full MTS4EA Trace view window)

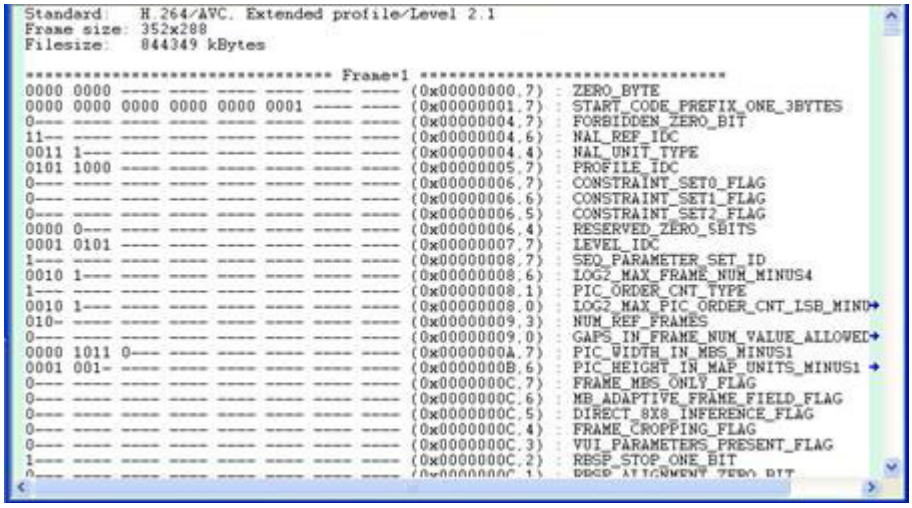


Figure 5-5: H.264/AVC example, at start of byte stream (full MTS4EA Trace view window)

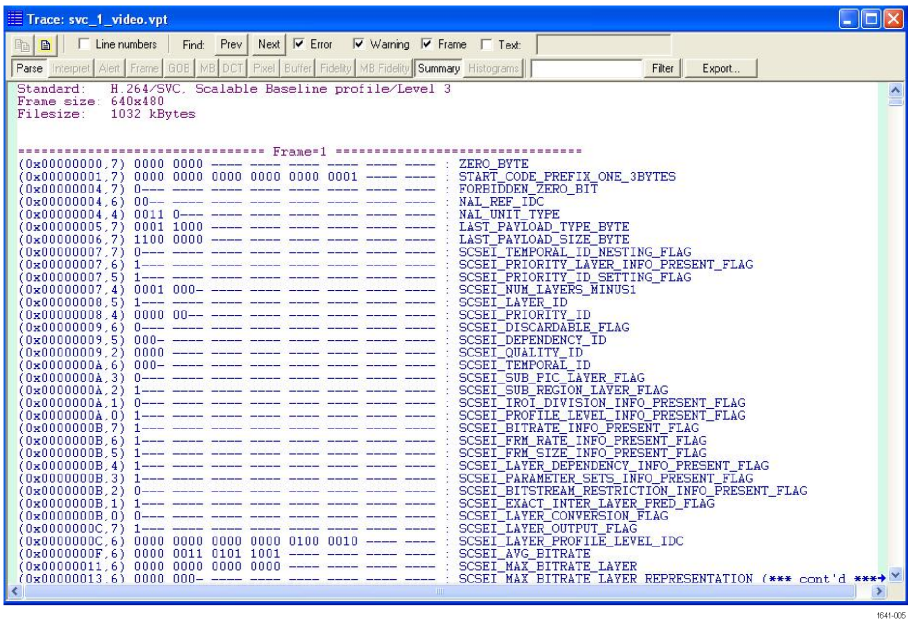


Figure 5-6: H.264/SVC example, at start of byte stream (full MTS4EA Trace view window)

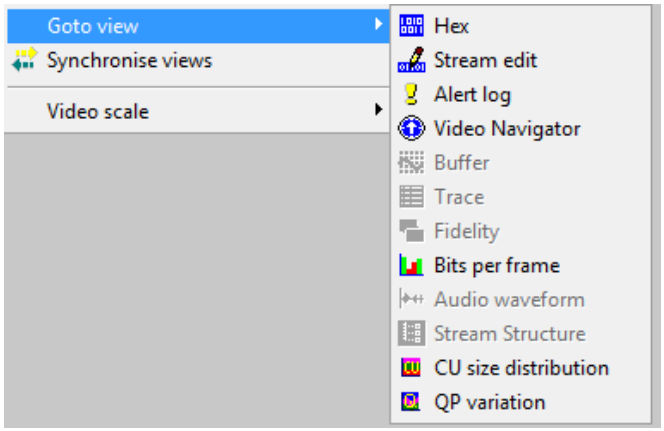
```
0000 0000 0000 0000 1000 00-- ---- ---- (0x00000000,7) : PICTURE START CODE
0000 0--- ---- ---- ---- ---- ---- ---- (0x00000002,6) : START_CODE
0000 0000 ---- ---- ---- ---- ---- ---- (0x00000002,1) : TREF
1000 0111 ---- ---- ---- ---- ---- ---- (0x00000003,1) : PTYPE: 1-0-ssi-dci-fpfr-sf
001- ---- ---- ---- ---- ---- ---- ---- (0x00000004,1) : UFEP
0110 1011 1000 1110 00-- ---- ---- ---- (0x00000005,6) : OPPTYPE
0000 0000 1--- ---- ---- ---- ---- ---- (0x00000007,4) : MPPTYPE
0--- ---- ---- ---- ---- ---- ---- ---- (0x00000008,3) : CPM
1--- ---- ---- ---- ---- ---- ---- ---- (0x00000008,2) : UUI
0110 1--- ---- ---- ---- ---- ---- ---- (0x00000008,1) : PQUANT
0--- ---- ---- ---- ---- ---- ---- ---- (0x00000009,4) : PEI
011- ---- ---- ---- ---- ---- ---- ---- (0x00000009,3) : MCBPC_I
0--- ---- ---- ---- ---- ---- ---- ---- (0x00000009,0) : AIC
0110 ---- ---- ---- ---- ---- ---- ---- (0x0000000A,7) : CBPY
0111 ---- ---- ---- ---- ---- ---- ---- (0x0000000A,3) : COEFF
1--- ---- ---- ---- ---- ---- ---- ---- (0x0000000B,7) :      SIGN
0011 00-- ---- ---- ---- ---- ---- ---- (0x0000000B,6) : COEFF
```

Figure 5-7: H.263 example, at start of bitstream

NOTE. *Interpret also provides an excellent tool to do bitstream syntax debugging. (See page 5-157, Interpret.)*

To quickly find if there are any errors in the Trace file, use the Find Prev/Next buttons on the trace view toolbar to navigate to the next/previous error or warning.

Go to other views/right click menu. A context-sensitive menu (Interpret & Parse Bitstream only) allows you to go to the area in another view which corresponds to the area that has been selected in the Trace file. The CU size distribution and QP variation options are enabled for H.265/HEVC streams only.



NOTE. *This right-click context-sensitive menu is only active in the Trace/Interpret and Trace/Parse Bitstream trace files.*

Interpret. This provides interpretation of the decoded data in the bitstream. Example outputs are shown in the following diagrams:

NOTE. *It is important to also see General codes used in Trace files and Alerts in the Bitstream Syntax Debugging section for an explanation of some of these items.* *(See page 6-1.)*

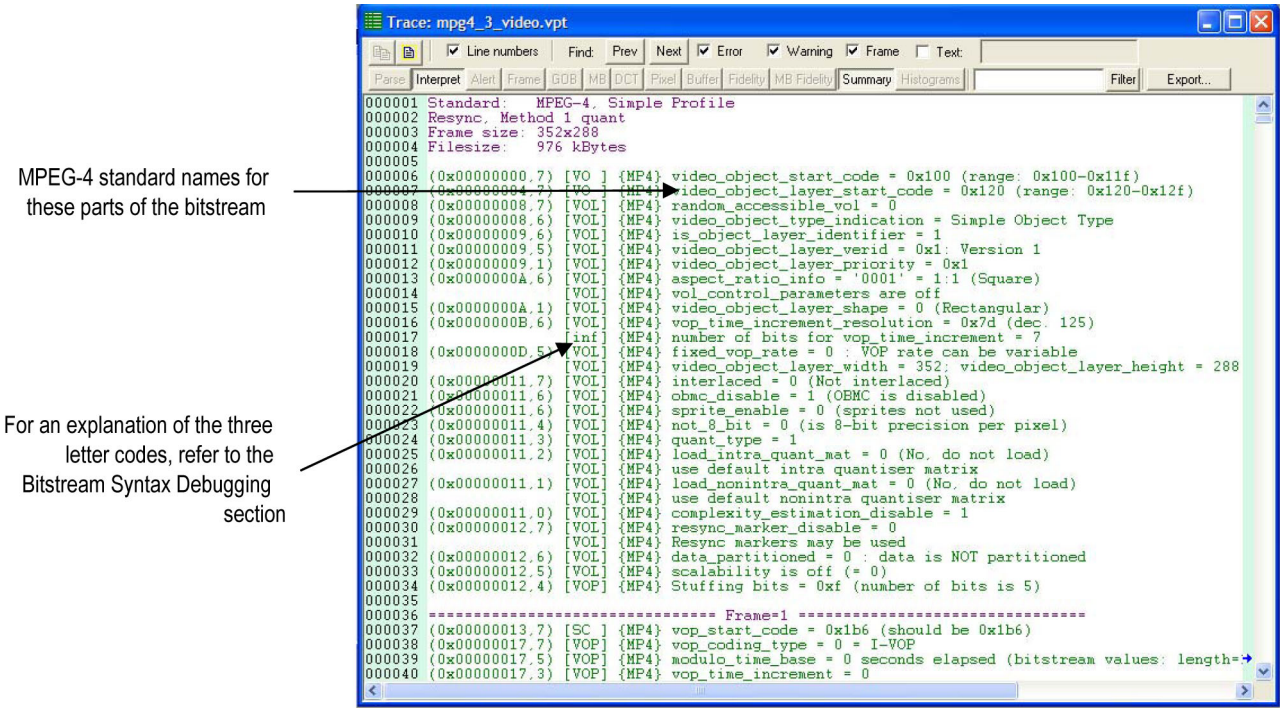


Figure 5-8: MPEG-4 trace interpret example

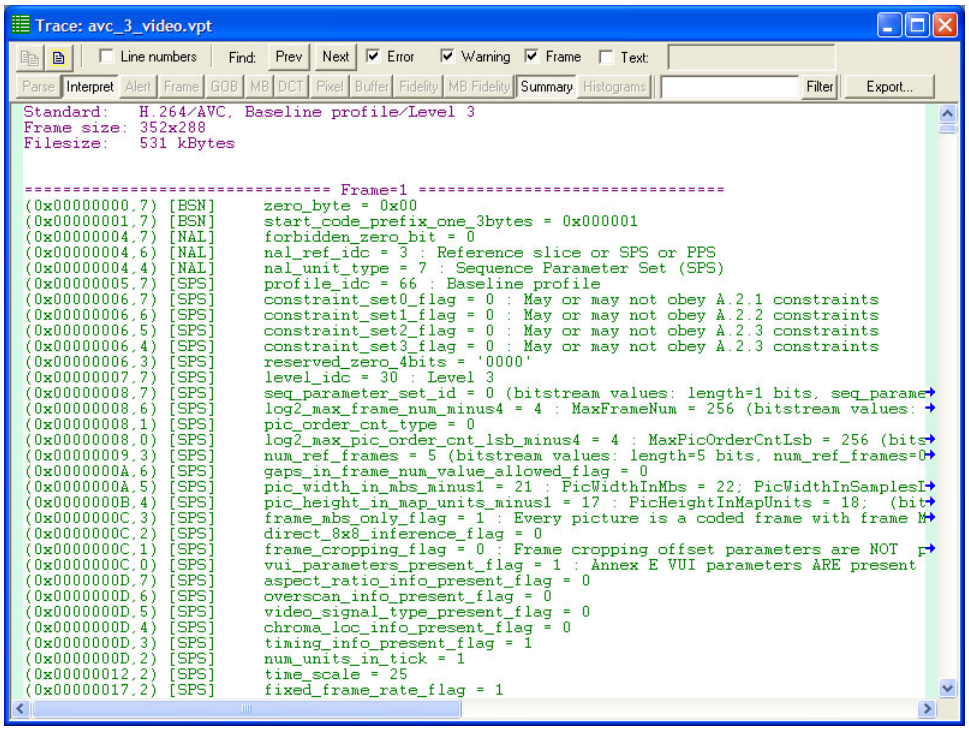


Figure 5-9: H.264/AVC example, at start of byte stream (full MTS4EA Trace view window)

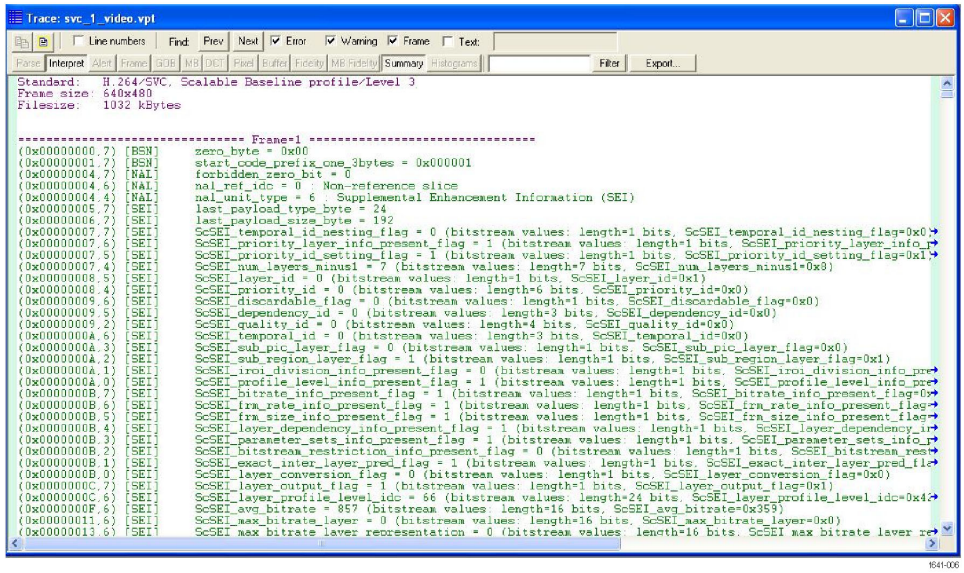


Figure 5-10: H.264/SVC example, at start of byte stream (full MTS4EA Trace view window)

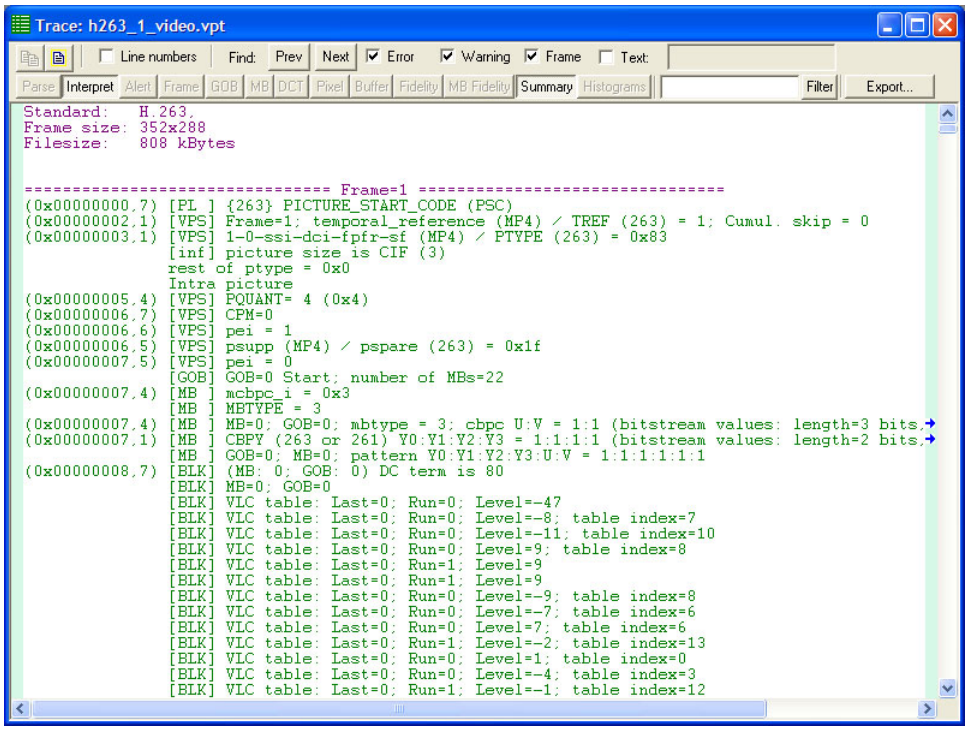
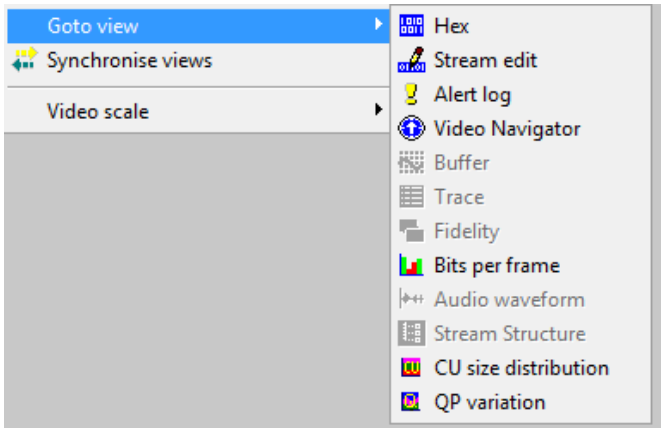


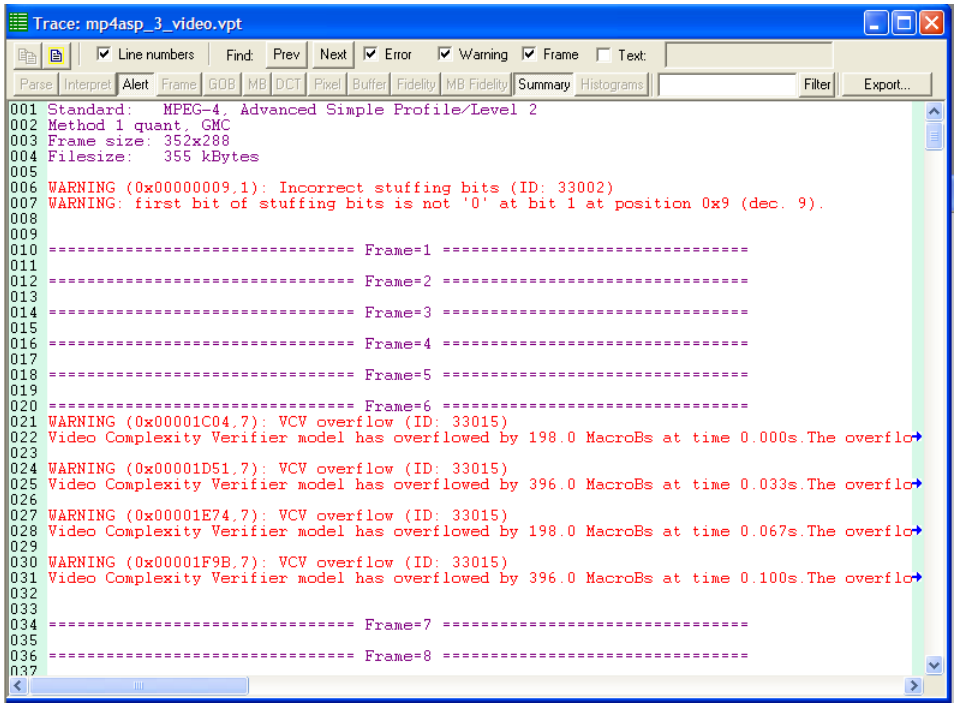
Figure 5-11: H.263 example at start of bitstream

A context-sensitive menu (Interpret & Parse Bitstream only) allows you to go to the area in another view which corresponds to the area that has been selected in the Trace file. The CU size distribution and QP variation options are enabled for H.265/HEVC streams only.



NOTE. *This right-click context-sensitive menu is only active in the Trace/ Interpret and Trace/Parse Bitstream trace files.*

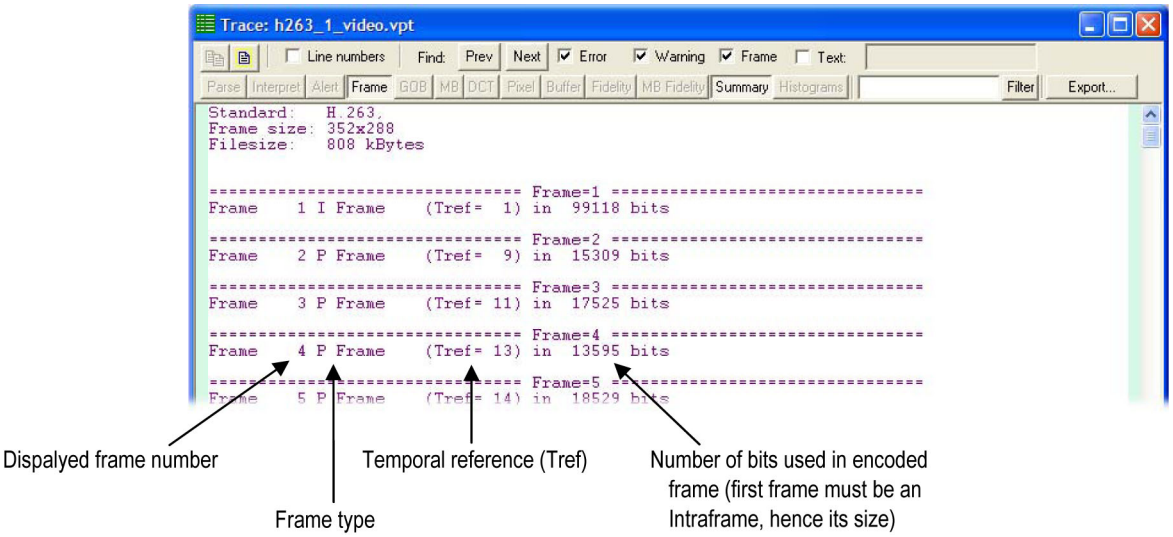
Alerts [all standards]. When selected, this option filters alerts and provides the alert level, hex address, title, identity, and description.



Frame summary [H.265/HEVC, H.264/AVC, MPEG-2, and H.263] or VOP summary [MPEG-4]

If this level of Trace is selected, then the Frame summary is output to the file, specifying the number of bits generated, the index of the decoded frame, the frame type, the temporal reference (if appropriate) and other data.

This example is for H.263, where Tref is used.



MB row summary [H.264/AVC, MPEG-4, MPEG-2] or GOB summary [H.263] or CTU row summary [H.265/HEVC]. This Trace outputs summary data for each horizontal row of MacroBlocks/CTUs, starting from the top of the frame and working downwards.

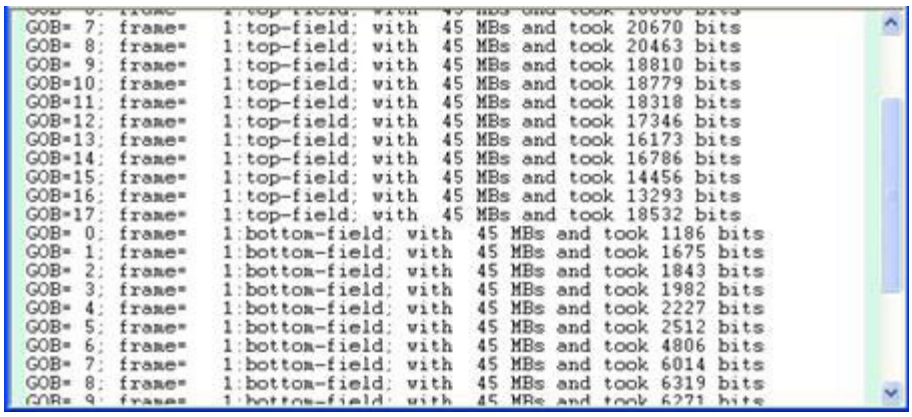


Figure 5-12: MPEG-2 example (interlaced example with top and bottom fields)

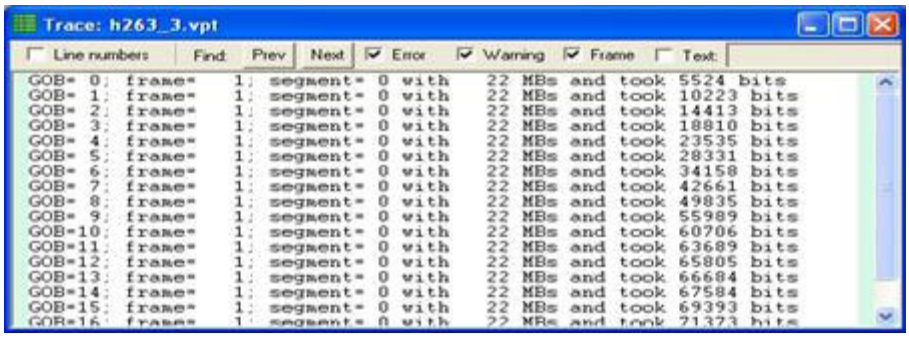


Figure 5-13: H.263 example

(See page 5-80, *Segment [H.263 only]*.)

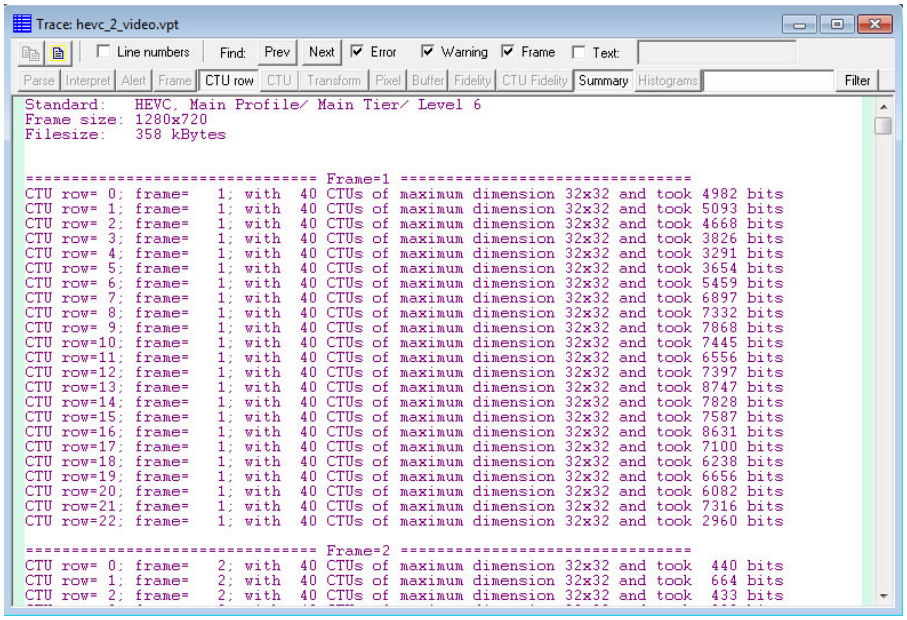


Figure 5-14: H.265 example

MB/CTU summary. For each MacroBlock/CTU, this option provides the location of the MacroBlock/CTU (column and row, frame), the type of coding used, the quantizer used, and the number of bits used. MB/CTU col=0, MB/CTU row=0 is the top left MacroBlock/CTU in the frame.

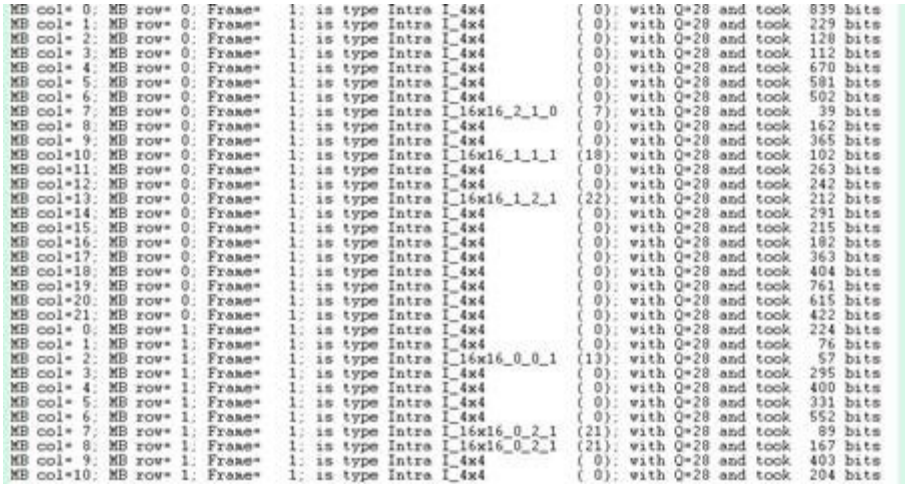


Figure 5-15: H.264/AVC example

Figure 5-16: MPEG-4 example

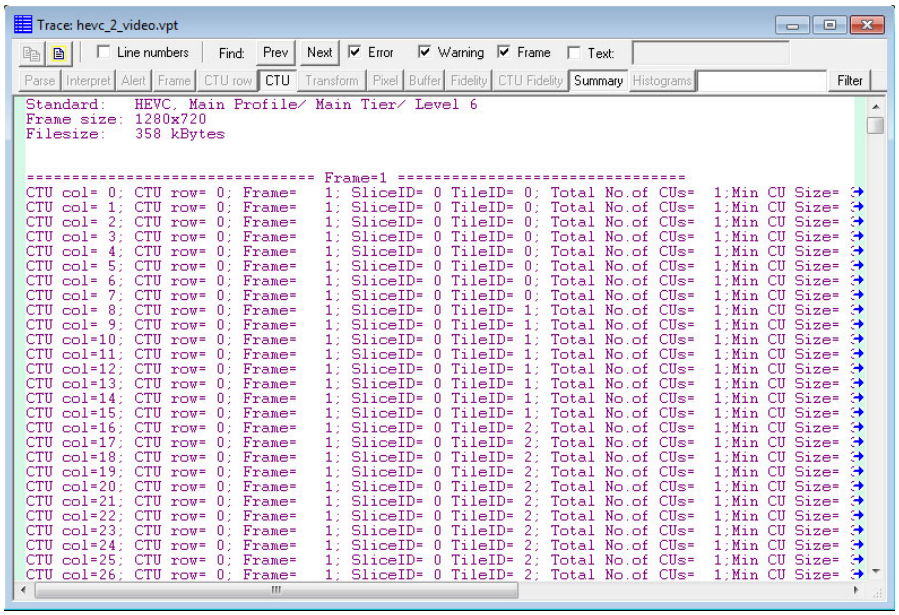


Figure 5-17: H.265 example

Transform level [H.264/AVC] or DCT level [MPEG-4, MPEG-2, and H.263].

NOTE. *In H.264/AVC the report is of transform coefficients that are not DCTs.*

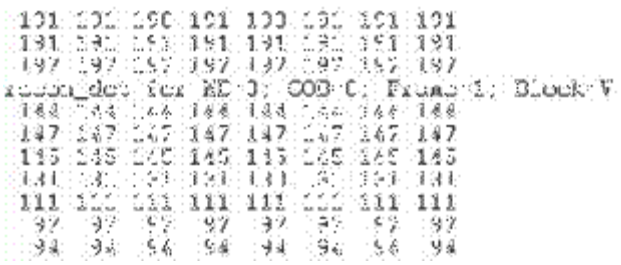
For MPEG-4, and H263 this gives three sets of information:

- Before dequantization = the values of the DCT coefficients in the bitstream
- After dequantization = the before values after they have been multiplied by the quantizer matrix

```
before dequantisation (MB=0; GOB=0; Frame=1; Block=Y0)
  18   0   0   0   0   0   0   0
  -2  -1   0   0   0   0   0   0
   1   0   0   0   0   0   0   0
  -1   0   0   0   0   0   0   0
   1   0   0   0   0   0   0   0
   0   0   0   0   0   0   0   0
   0   0   0   0   0   0   0   0
   0   0   0   0   0   0   0   0
after dequant (MB=0; GOB=0; Frame=1; Block=Y0) quant=1
 432   0   0   0   0   0   0   0
 -68 -36   0   0   0   0   0   0
  40   0   0   0   0   0   0   0
 -42   0   0   0   0   0   0   0
  44   0   0   0   0   0   0   0
   0   0   0   0   0   0   0   0
   0   0   0   0   0   0   0   0
```

Block Y0 before and after dequantization

- recon_dct = the inverse DCT residual values, before summing with any prediction. The data is written into the file in two separate places - the second part is below:



In H.264/AVC the report is of transform levels:

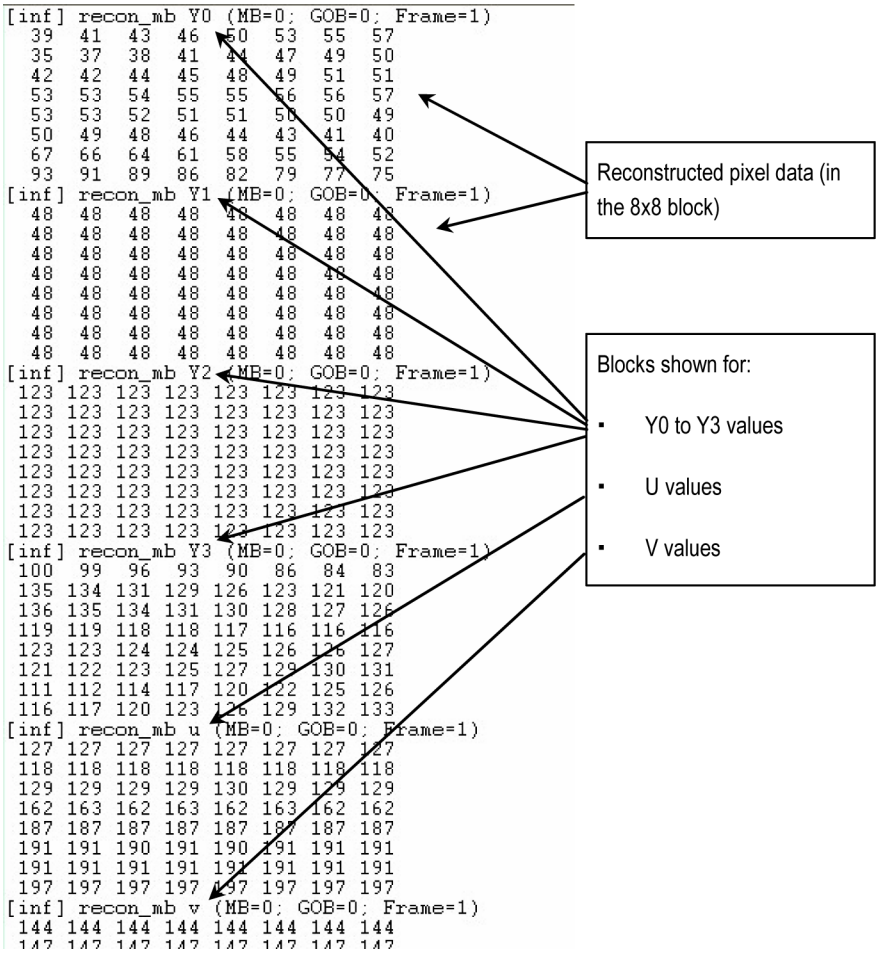
```
levels (MB: 0; Frame: 1; Block 0:Y0)
 -2816  -640  -256  -320
     0     0     0     0
     0     0     0     0
     0     0     0     0
levels (MB: 0; Frame: 1; Block 0:Y1)
  -256     0     0     0
     0     0     0     0
     0     0     0     0
     0     0     0     0
levels (MB: 0; Frame: 1; Block 0:Y2)
     0     0     0     0
     0     0     0     0
     0     0     0     0
     0     0     0     0
levels (MB: 0; Frame: 1; Block 0:Y3)
     0     0     0     0
     0     0     0     0
     0     0     0     0
     0     0     0     0
levels (MB: 0; Frame: 1; Block 0:U)
  1024     0     0     0
     0     0     0     0
     0     0     0     0
     0     0     0     0
levels (MB: 0; Frame: 1; Block 0:V)
  1024     0     0     0
     0     0     0     0
     0     0     0     0
     0     0     0     0
levels (MB: 0; Frame: 1; Block 1:Y0)
     0     0     0     0
     0     0     0     0
     0     0     0     0
     0     0     0     0
levels (MB: 0; Frame: 1; Block 1:Y1)
     0     0     0     0
     0     0     0     0
```

In MPEG-2 the data is reported in the following format:

```
Using default intra quantizer matrix:
   8  16  19  22  26  27  29  34
  16  16  22  24  27  29  34  37
  19  22  26  27  29  34  34  38
  22  22  26  27  29  34  37  40
  22  26  27  29  32  35  40  48
  26  27  29  32  35  40  48  58
  26  27  29  34  38  46  56  69
  27  29  35  38  46  56  69  83

Using default non-intra quantizer matrix:
  16  16  16  16  16  16  16  16
  16  16  16  16  16  16  16  16
  16  16  16  16  16  16  16  16
  16  16  16  16  16  16  16  16
  16  16  16  16  16  16  16  16
  16  16  16  16  16  16  16  16
  16  16  16  16  16  16  16  16
  16  16  16  16  16  16  16  16

DCT coefficients for MB 0 (x=0, y=0), block Y0:
Before dequantisation:                After dequantisation:                 After inverse transfo
 -86   0   0   0   0   0   0   0      -688   0   0   0   0   0   0   0      -86 -86 -86 -86 -86
   0   0   0   0   0   0   0   0         0   0   0   0   0   0   0   0      -86 -86 -86 -86 -86
   0   0   0   0   0   0   0   0         0   0   0   0   0   0   0   0      -86 -86 -86 -86 -86
   0   0   0   0   0   0   0   0         0   0   0   0   0   0   0   0      -86 -86 -86 -86 -86
   0   0   0   0   0   0   0   0         0   0   0   0   0   0   0   0      -86 -86 -86 -86 -86
   0   0   0   0   0   0   0   0         0   0   0   0   0   0   0   0      -86 -86 -86 -86 -86
   0   0   0   0   0   0   0   0         0   0   0   0   0   0   0   0      -86 -86 -86 -86 -86
   0   0   0   0   0   0   0   0         0   0   0   0   0   0   0   0      -86 -86 -86 -86 -86

DCT coefficients for MB 0 (x=0, y=0), block Y1:
Before dequantisation:                After dequantisation:                 After inverse transfo
  19 -43 -10   0  -1   0   0   0       152-430-118   0 -16   0   0   0      -74 -47  -9  24  51
  -4   4  -1   0   0   0   0   0       -40  40 -13   0   0   0   0   0      -79 -52 -16  17  43
   0   0   0   0   0   0   0   0         0   0   0   0   0   0   0   0      -80 -54 -18  14  41
   2  -1   0   0   0   0   0   0        27 -13   0   0   0   0   0   0      -76 -50 -14  21  48
   1   0   0   0   0   0   0   0        13   0   0   0   0   0   0   0      -74 -48 -10  26  57
```

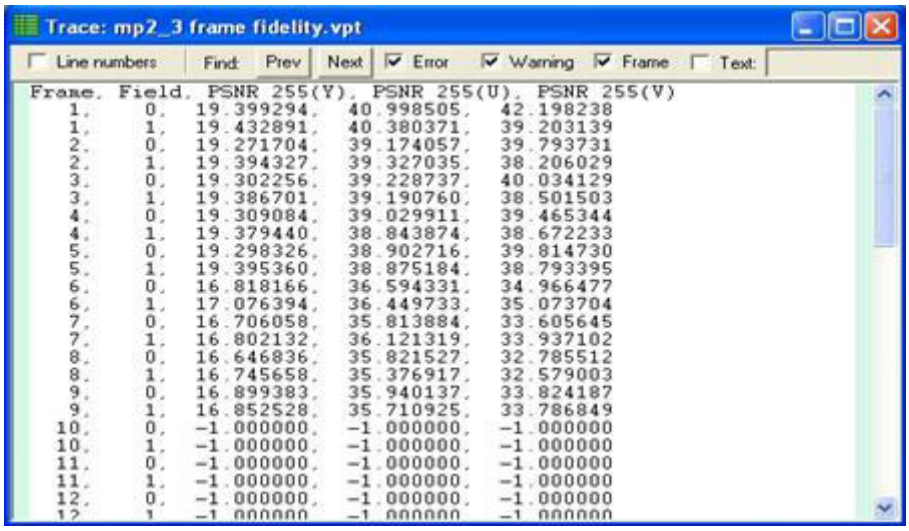
NOTE. *This can generate a lot of data.*

Pixel level. This lowest level of Trace provides the output pixel values in 8x8 blocks for the video decoding process.



NOTE. *This can generate a lot of data.*

Frame fidelity [H.265/HEVC, H.264/AVC, MPEG-2, and H.263] or VOP fidelity [MPEG-4]. This provides a Trace output of the fidelity analysis frame-by-frame. (See page 5-186, *Video fidelity enable....*)



NOTE. *The lines in the preceding screenshot for frames 10 and above are empty because the corresponding YUV file stops at this point (there are no more frames in the YUV file). The -1 in the trace file indicates that the YUV file is missing.*

Some of the example files provided have the necessary corresponding YUV files for fidelity analysis. (See page 5-39, YUV source files for example fidelity analysis.)

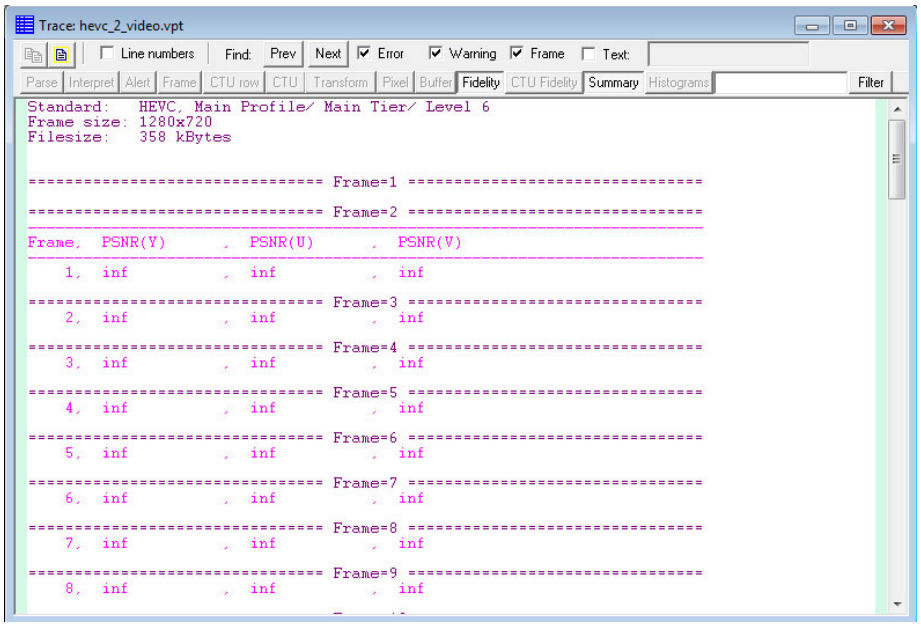
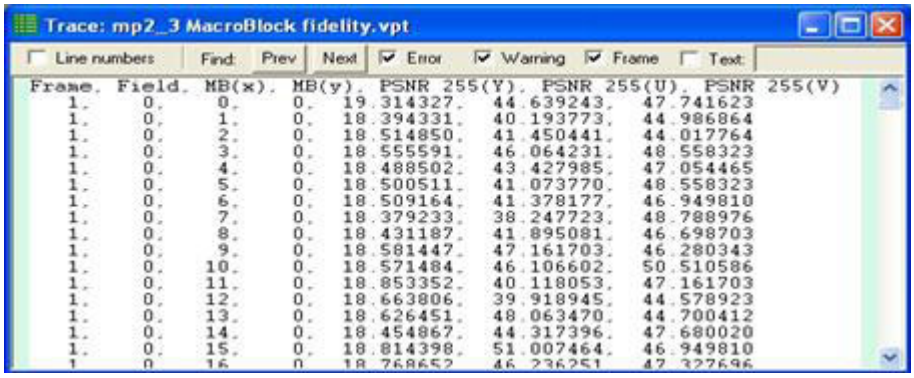


Figure 5-18: Frame fidelity – H.265/HEVC example

MacroBlock fidelity. This provides a Trace output of the fidelity analysis for each MacroBlock in the frame range. (See page 5-186, *Video fidelity enable....*)



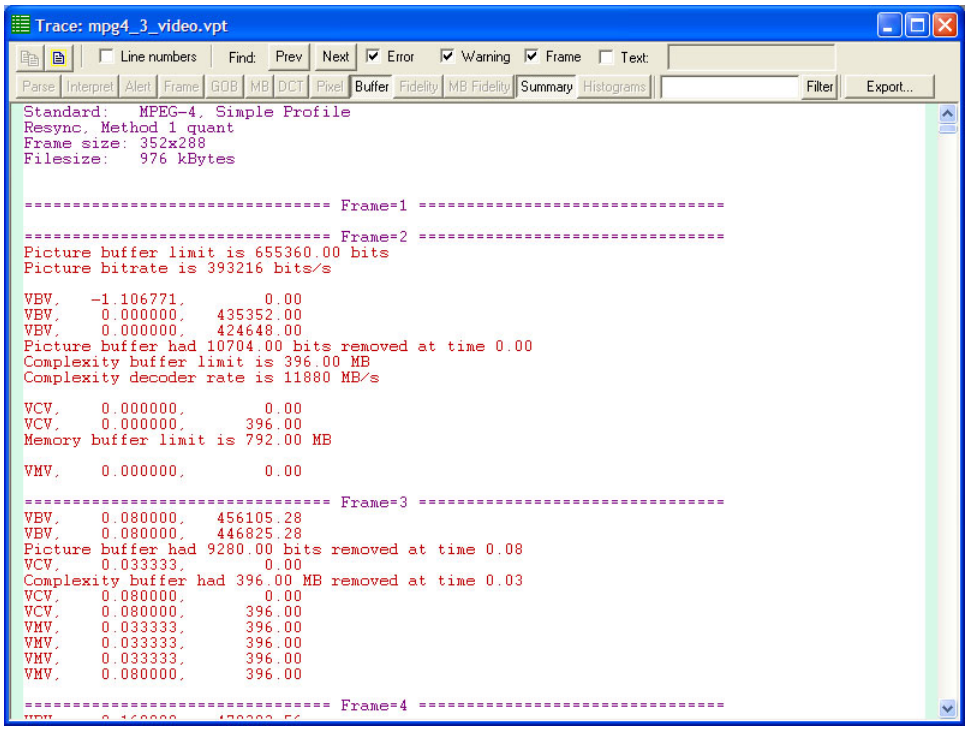
NOTE. *Some of the example files provided have the necessary corresponding YUV files for fidelity analysis.* *(See page 5-39, YUV source files for example fidelity analysis.)*

Buffer analysis [MPEG-4 Part 2, MPEG-2, H.264/AVC, and H.265/HEVC]. When selected, this option traces the buffer analysis.

The data output for the H.264/AVC standard can include multiple picture buffers. The buffer limit and picture bit rate are supplied at the start of the trace data. The remainder of the trace data indicates the time and the bits present in the buffer. Other information recorded includes when bits are removed, and buffer underflows and overflows. The H.264/AVC trace will only be available if the stream contains sufficient information to perform the buffer analysis.

The data output for the MPEG-2 standard is the same as for H.264/AVC, but only one buffer is ever traced.

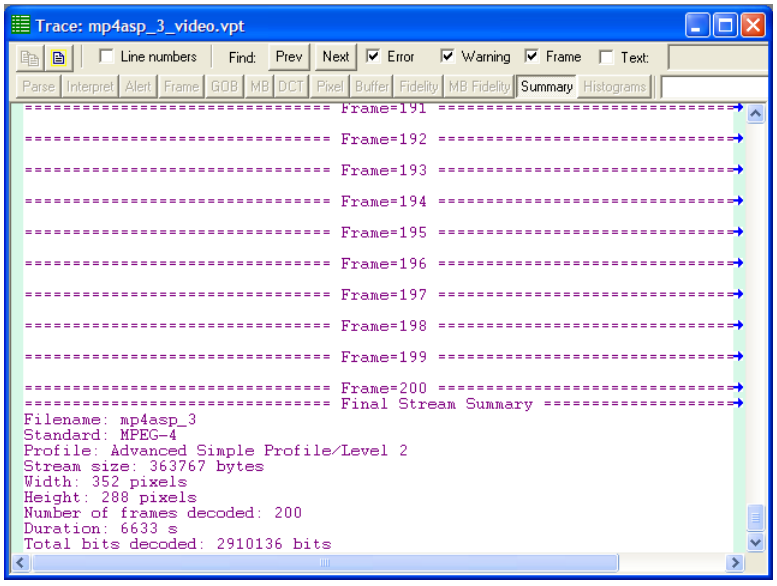
The data output for the MPEG-4 Part 2 includes a video buffer verifier (VBV), a video memory verifier (VMV) and a video complexity verifier (VCV). The video buffer outputs the same data as H.264/AVC, that is, buffer limit and picture bit rate. The memory and complexity verifiers output their data in macroblock units.



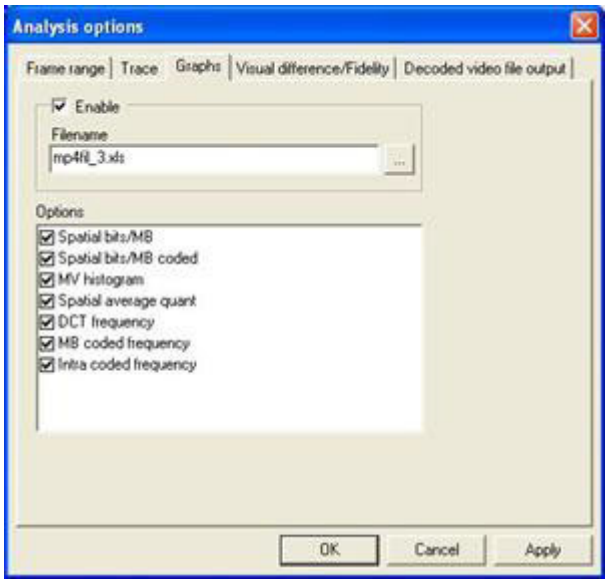
For H.265/HEVC streams, in the case that the subpicture HRD parameters are available in the stream, the buffer details at the subpicture level are also displayed as part of the buffer trace feature.

```
================================ Frame=1 ============================
Picture buffer 0 limit is 368640.00 bits
Picture bitrate 0 is 614400 bits/s

VBV0 ,     0.000000,         0.00
VBV0 ,     0.000000,         0.00
VBV0 ,     0.392513,    241160.00

~~~~~~~~~~~~~~~~~~ Sub-picture buffer details begin ~~~~~~~~~~~~~~~~~~
Picture buffer 0 limit is 368640.00 bits
Picture bitrate 0 is 614400 bits/s

VBV0 ,     0.000000,         0.00
VBV0 ,     0.000000,         0.00

DU Index:0
VBV0 ,     0.000000,         0.00
VBV0 ,     0.100755,     61904.00
```

Stream summary [all standards]. A summary of the stream properties is appended to the end of all traces, unless it is actively disabled.



Video graph enable... Ctrl+G



Graph enable turns on the collection of statistics for display using Microsoft Excel. It does this by saving the statistics of the decoding process in an .xls file. Microsoft Excel is then run and a macro converts the statistics data into graphs, which are then available in various tabs of Excel.

The collection of statistics varies depending on the standard concerned - for example, there is no DCT as such in H.264/AVC or H.265/HEVC, so this is not available:

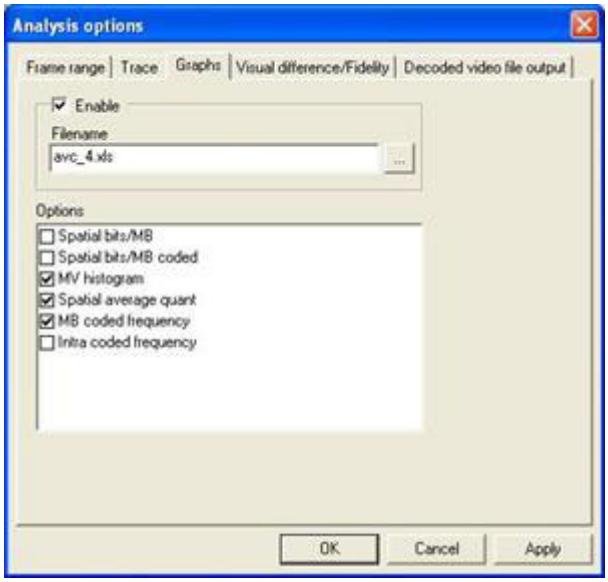


Figure 5-19: H.264 graph

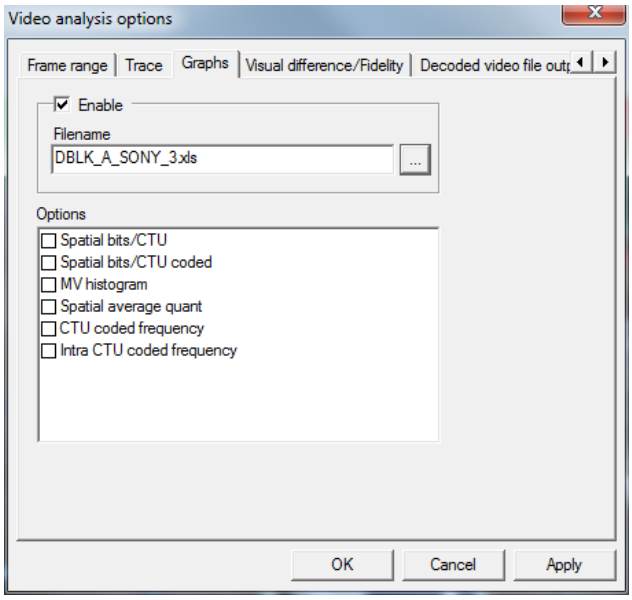


Figure 5-20: H.265 graph

Many of these graphs are equivalent to the statistical summaries that can be overlaid on the final frame, but are presented in a more visual format.

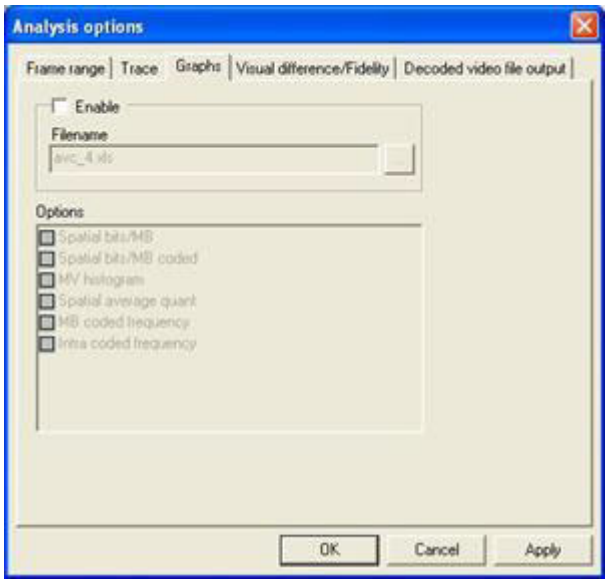
NOTE. *The MTS4EA assumes that the Microsoft Excel program is used for analysis of the provided statistics, and the macro given as part of MTS4EA is designed to operate with Excel versions 97, XP, and 2000 to 2010.*

For H.265/HEVC, when the number of CTUs in a frame is more than 10,000, the Excel macros are unable to convert the statistics data into graphs.

Frame range tab

You can set the range of frames over which the statistics are collected can be set. (See page 5-234, *Frame range tab*.)

Enable. This option enables/disables the collection of the statistics for the Graph data. When disabled, the Graph tab appears as follows:



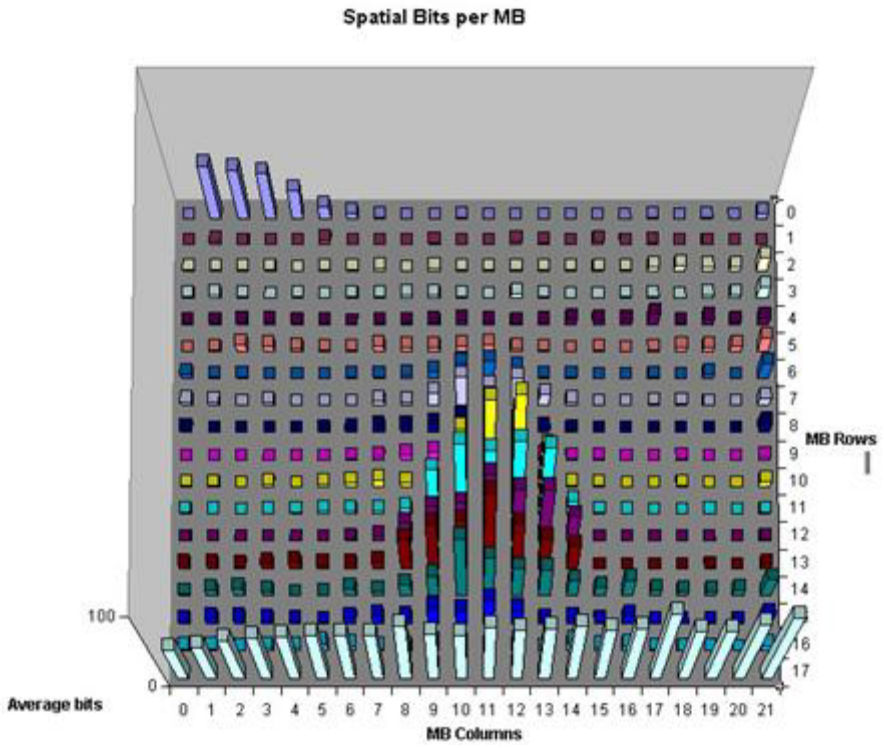
Filename. By default, the suggested filename is the same as the base name of the input video file given the extension .xls.

A different filename can be entered (but not a different file extension), and the folder where the file is to be stored can be changed by clicking the browse box.

NOTE. *An Excel spreadsheet can only be 256 columns wide - if the motion vectors are outside the range ± 127, the values are put into bins to scale the values within the 256 columns.*

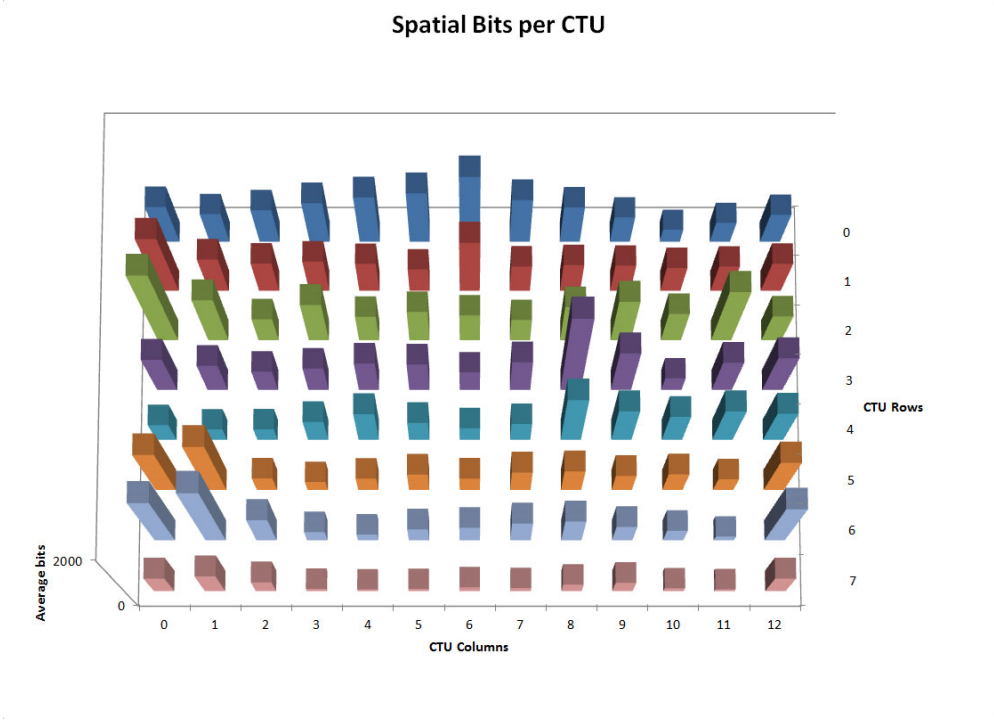
Graphs tab

Spatial bits/MB. This option creates a graph that shows the average bits per MacroBlock as a contour plot, over a two-dimensional plane that represents the picture. The contours are plotted in different colors, and the meanings of the colors are represented in a key positioned at the side of the graph.

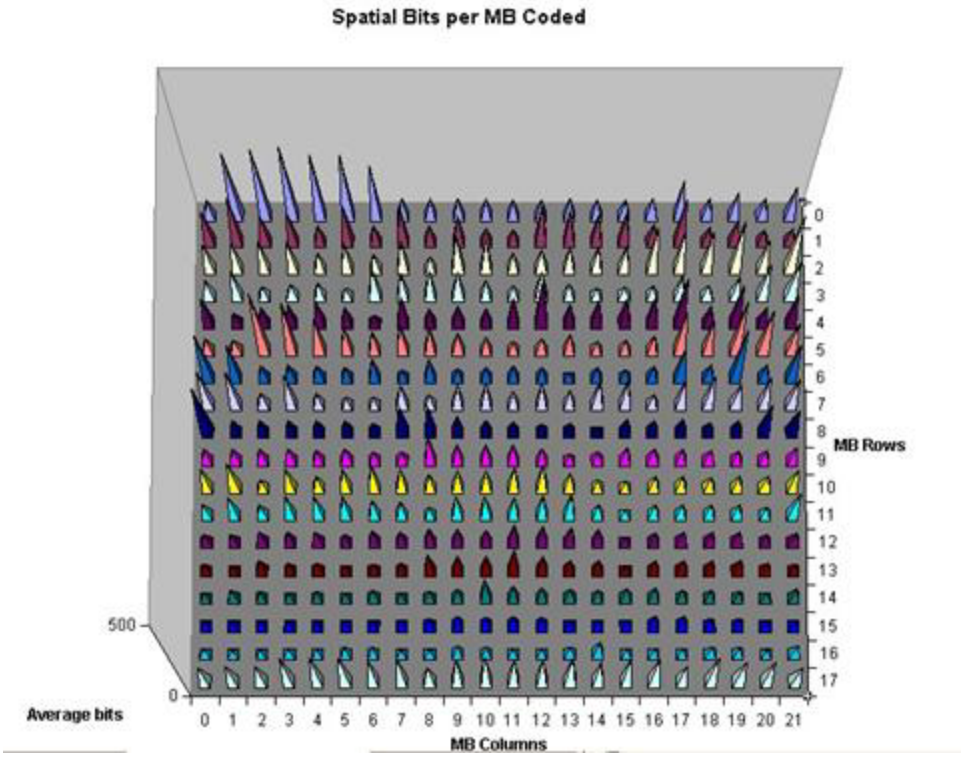


Spatial bits/CTU. This H.265/HEVC option creates a graph that shows the average bits per CTU as a contour plot, over a two-dimensional plane that represents the picture. The contours are plotted in different colors, and the meanings of the colors are represented in a key positioned at the side of the graph.

NOTE. *When the number of CTUs in a frame exceeds 10,000, the Excel macros are unable to convert the statistics data into graphs.*

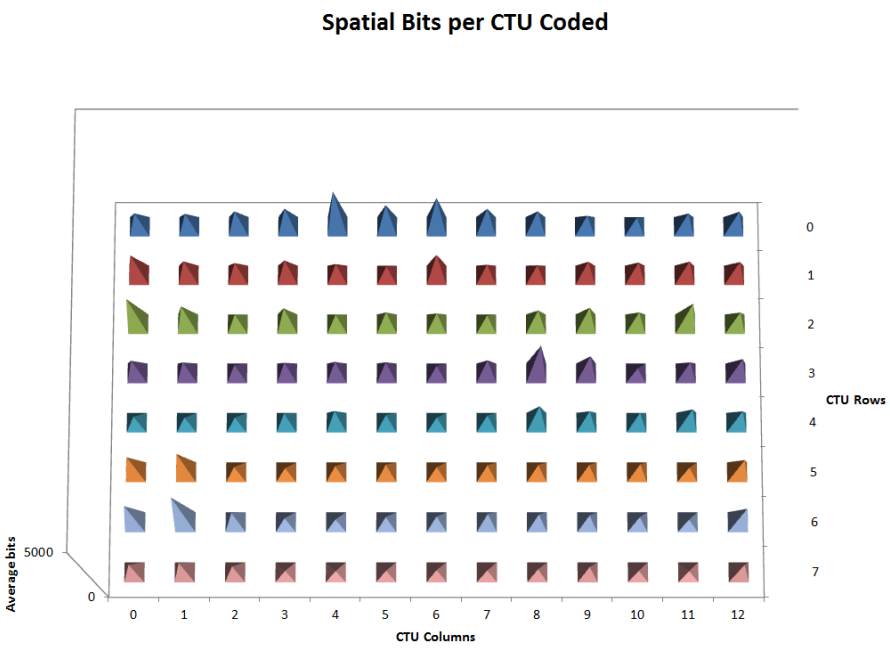


Spatial bits/MB coded. This option creates a three-dimensional pyramidal graph, where the height of the pyramid represents the average number of bits each MacroBlock used for each time it was coded.

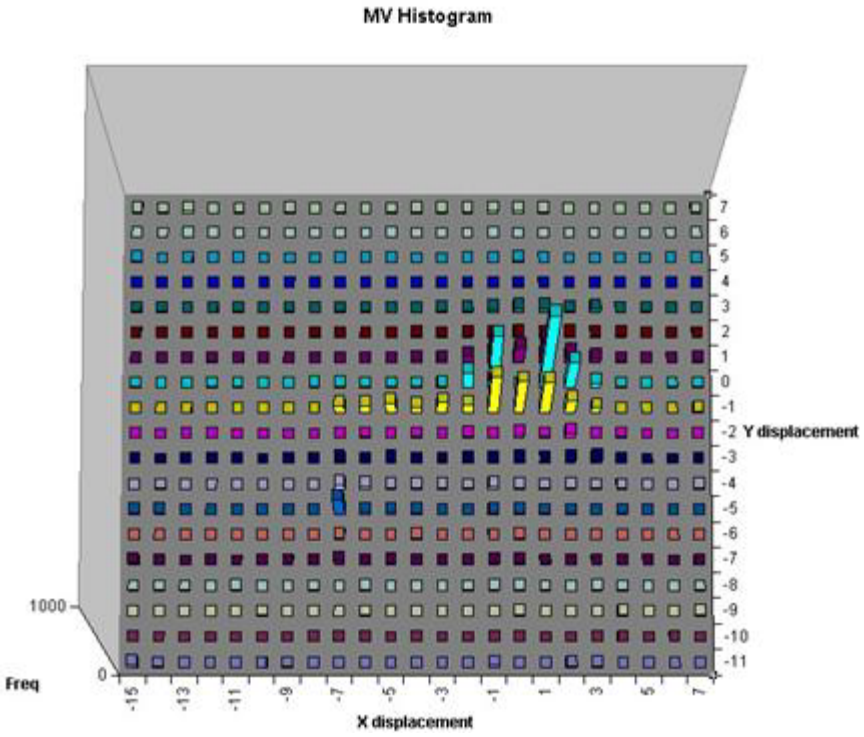


Spatial bits/CTU coded. This H.265/HEVC option creates a three-dimensional pyramidal graph, where the height of the pyramid represents the average number of bits each CTU used for each time it was coded.

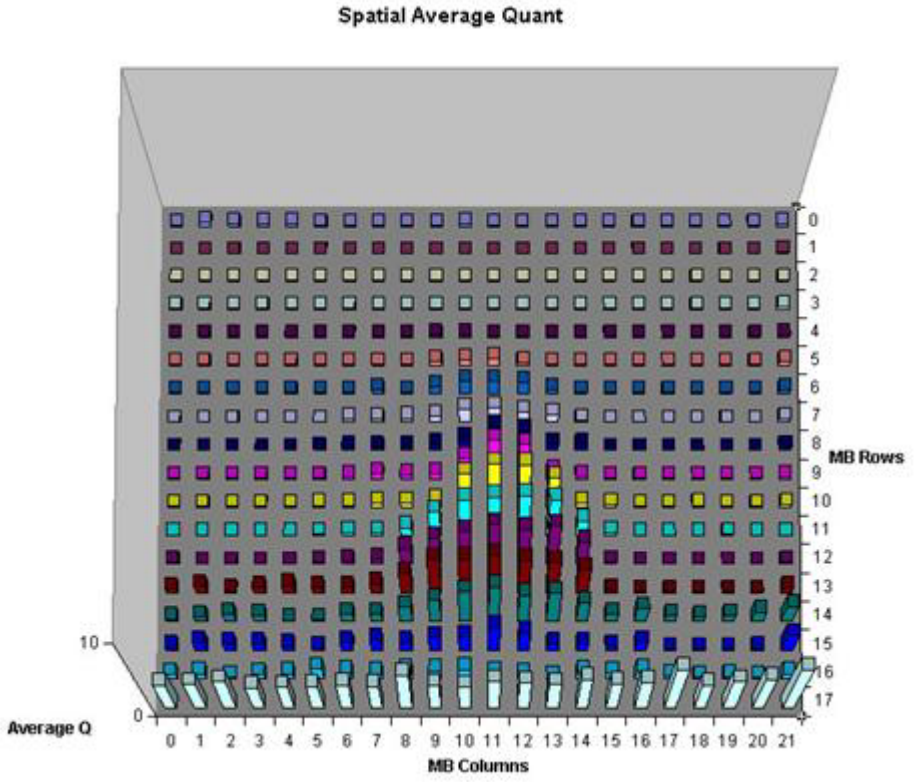
NOTE. *When the number of CTUs in a frame exceeds 10,000, the Excel macros are unable to convert the statistics data into graphs.*



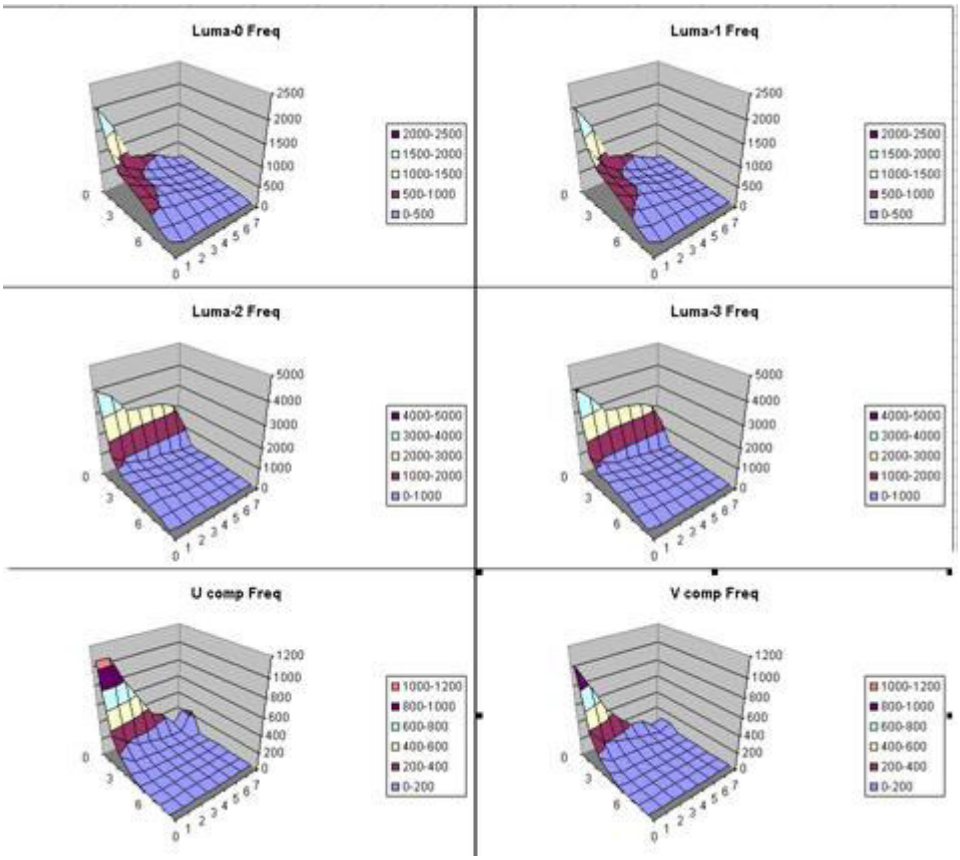
MV histogram. The motion vector histogram creates a graphical representation of the frequency of use of different motion vectors. This can be used to identify the range of search of the motion estimator in the encoder. The heights of the blocks represent the frequency with which the motion vectors were chosen.



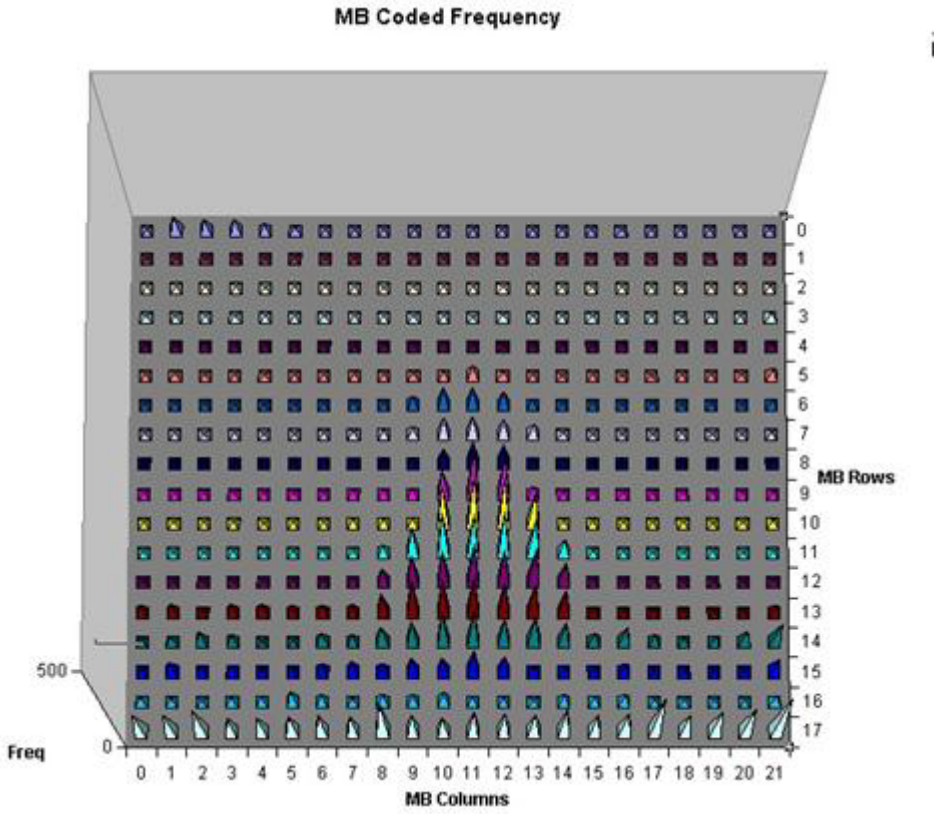
Spatial average quant. This option creates a graph of the quantizer choice at each part of the picture. This can be useful for determining poor quantizer control in certain encoders (often too low quantizers are chosen at the top of pictures, which then overshoots the bit-budget and then higher quantizers must be used progressively as the frame is encoded - meaning that the top of the picture is always coded more accurately than the bottom).



DCT frequency [MPEG-4, MPEG-2, and H.263 only]. This option creates a graph of the frequency with which the different components of the Discrete Transform Cosine are used. The positions of the transmitted coefficients are recorded and shown here for each of the six 8x8 blocks that make up the MacroBlock. This feature can be useful for identifying high frequency noise being transmitted in the picture because of poor front-end filtering.

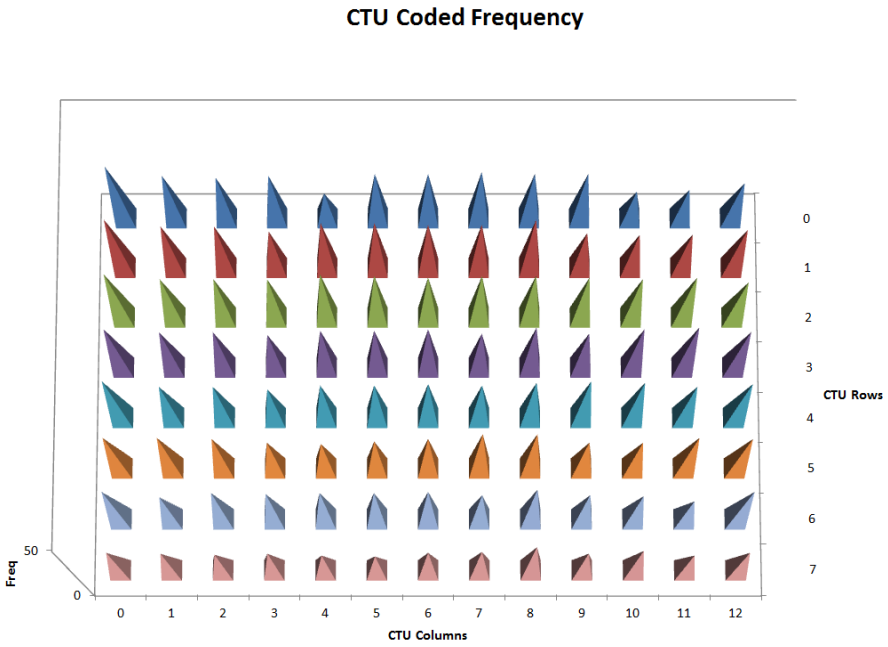


MB coded frequency. This option creates a graph of the frequency of coding of each MacroBlock. It can help to identify the perpetual encoding of particular regions of the image due to issues in the digitization process, the rate control process, or the sensor.

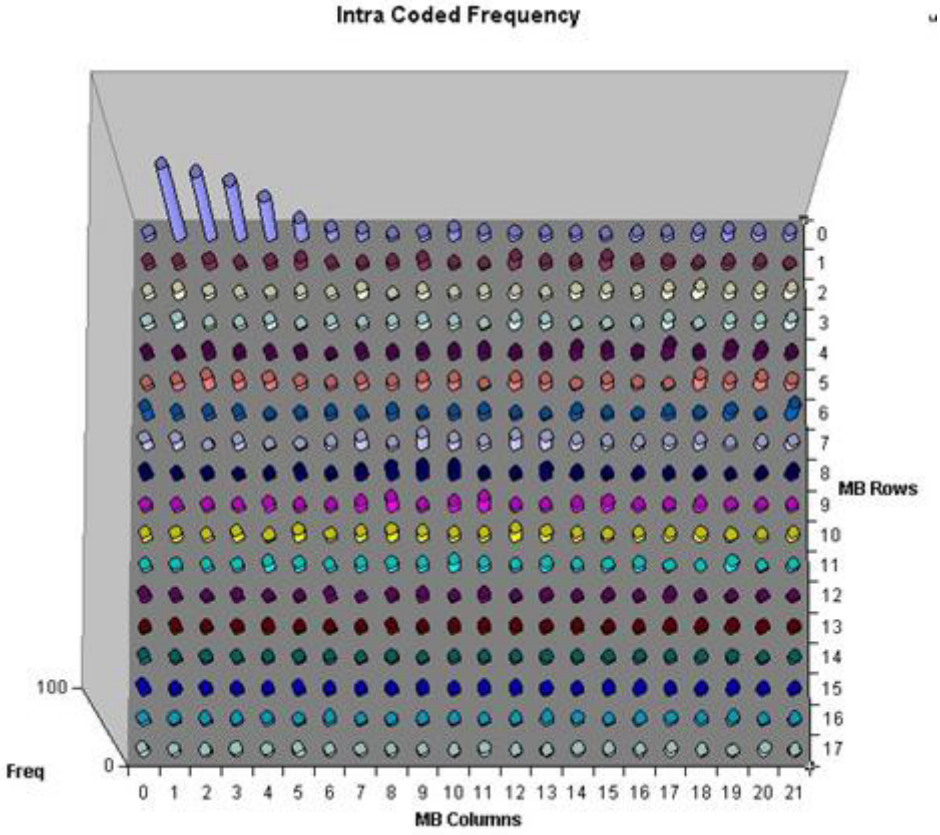


CTU coded frequency. This H.265/HEVC option creates a graph of the frequency of coding of each CTU. It can help to identify the perpetual encoding of particular regions of the image due to issues in the digitization process, the rate control process, or the sensor.

NOTE. *When the number of CTUs in a frame exceeds 10,000, the Excel macros are unable to convert the statistics data into graphs.*

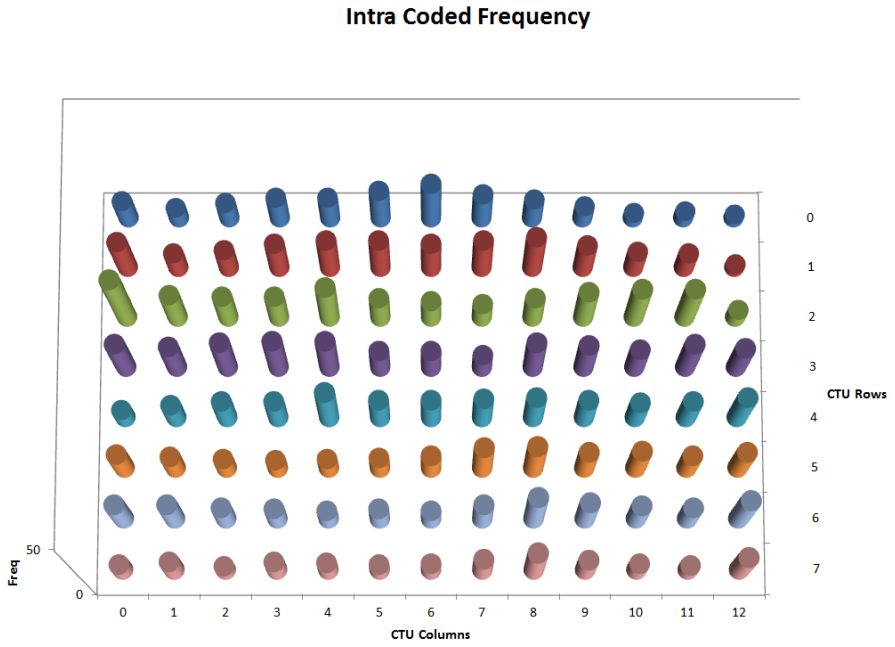


Intra coded frequency. This option creates a graph of the frequency of intra coding of each MacroBlock. Over time, each MacroBlock can be updated in Intra mode every so often (as required in the Standards) in case there is a line error or a mismatch between encoder and decoder. Encoders that do this more often are more likely to have better error robustness, although at the expense of more bits being used (as intra coding takes more bits than inter coding).



Intra CTU coded frequency. This H.265/HEVC option creates a graph of the frequency of intra coding of each CTU. Over time, each CTU can be updated in Intra mode every so often (as required in the Standards) in case there is a line error or a mismatch between encoder and decoder. Encoders that do this more often are more likely to have better error robustness, although at the expense of more bits being used (as intra coding takes more bits than inter coding).

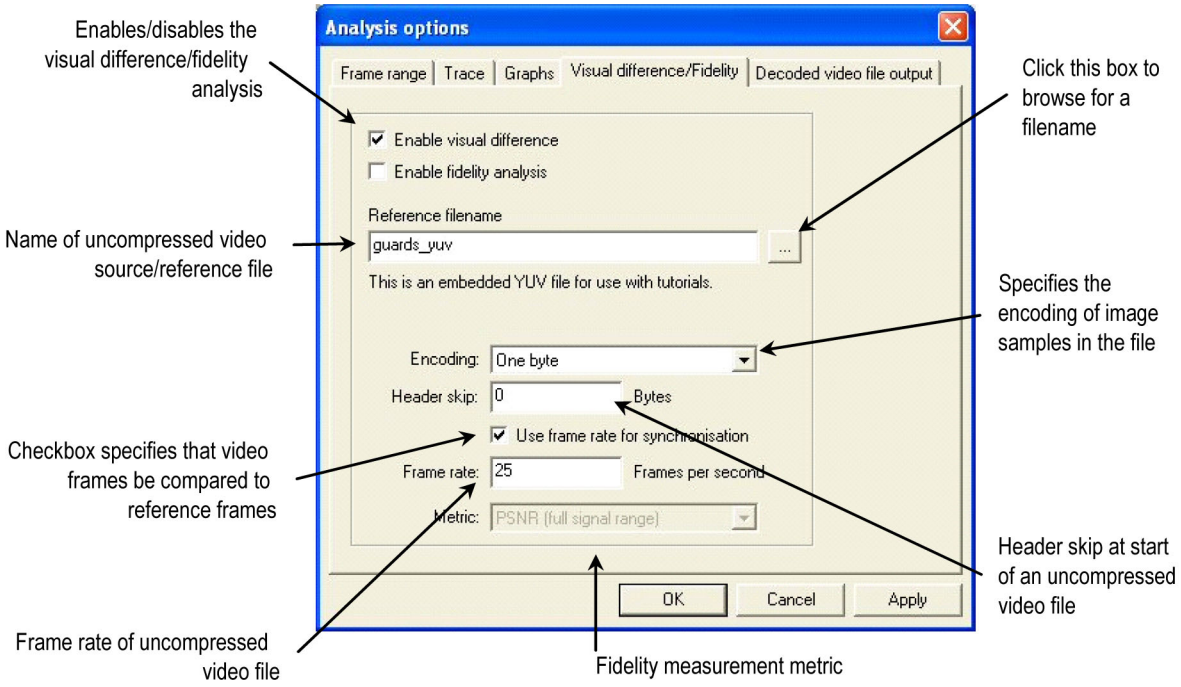
NOTE. *When the number of CTUs in a frame exceeds 10,000, the Excel macros are unable to convert the statistics data into graphs.*



Video fidelity enable...

NOTE. *Some options on this tab are different when H.264/AVC High Profile is used.*

This opens the Fidelity analysis tab of the Analysis options:



The results of the fidelity analysis are displayed:

- In the Trace files, when the appropriate Trace option is selected (See page 5-147, *Video trace enable... Ctrl+T*.) (See page 5-168, *Frame fidelity [H.265/HEVC, H.264/AVC, MPEG-2, and H.263] or VOP fidelity [MPEG-4]*.) (See page 5-169, *MacroBlock fidelity*.)
- As real-time overlays, when the overlay is selected (See page 5-105, *MB statistics*.) (See page 5-111, *Fidelity*.) (See page 5-115, *Average fidelity*.)

Reference filename

The name of the file used as the reference or source of encoding, of the encoded stream.

This file must be one of the following formats:

- 8 bits per sample, 4:2:0
- More than 8 bits per sample, and/or 4:2:2 or 4:4:4 (as used by H.264/AVC High Profile/FRExt, High/10, High/4:2:2, High/4:4:4)

YUV format of 8 bits per sample 4:2:0. The YUV file output is raw YUV with no headers of any kind: this is the same format as used by the Microsoft MPEG-4 Part 2 reference encoder Reference [6] in the *Compression Standards and File Types* section and as used commonly by other programs: (See page 4-1.)

- No headers of any kind (no file or frame headers)
- One byte per sample
- Row raster order (top picture row first)
- Planar YUV 4:2:0 sub-sampled (4 bytes of Y data for each byte of U data and each byte of Y data)
- Y plane values are 0-255 unsigned
- U and V plane values are unsigned with a DC offset of 128

Other uncompressed formats. The general uncompressed video file format is as follows:

- No headers of any kind (no file or frame headers)
- Concatenated planar image data
- Row raster order (top picture row first)
- Unsigned samples

For 8-bit sample depth:

- One byte per sample
- For 9-16 bit sample depth:
- Two bytes per sample
- Both little- and big-endian byte orders supported

For YUV format:

- Concatenated Y, U and V planes
- U and V planes sub-sampled as required
- Y plane samples are unsigned
- U and V plane samples are unsigned with a DC offset of 2^{-1}, where *n* is the chroma sample bit depth

For RGB format:

- Concatenated R, G and B planes
- For grayscale format:
- Luma plane only

NOTE. *There are built-in example YUV reference files for three of the example bitstreams: for MPEG-4 Man Walking, for H.264/AVC Grenadier Guards and for MPEG-2 Grenadier Guards. When these example streams are selected and fidelity analysis is enabled, the file names are automatically filled in. (See page 5-33, Example files.)*

Encoding

This control specifies the encoding of image samples in the file format. The following options are available:

- One byte - this specifies that image samples are stored in one byte per sample. This format is appropriate if all image planes are 8-bits deep
- Two byte MSB first - this specifies that image samples are stored in a pair of bytes for each sample. The most significant byte occurs first in each pair (big-endian). This format is appropriate if one or more image planes are deeper than 8-bits
- Two byte LSB first - this specifies that image samples are stored in a pair of bytes for each sample. The least significant byte occurs first in each pair (little-endian). This format is appropriate if one or more image planes are deeper than 8-bits

Header skip

The number of bytes at the start of the file before the first frame: MTS4EA will skip past these bytes (ignoring them).

Use frame rate for synchronization

This checkbox specifies that decoded video frames should be compared to reference frames according to corresponding time stamps. If this checkbox is not selected, then corresponding frame numbers are used.

Frame rate

The rate at which the uncompressed video frames were recorded, in frames per second.

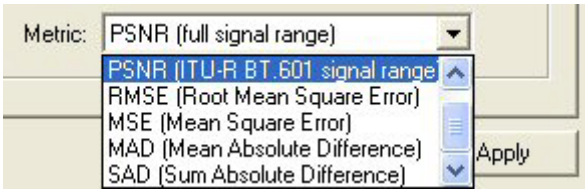
NOTE. *The number entered in the Frame rate field can be an integer (for example, 30), or a fraction (for example, 30000/1001) or a decimal number (for example, 29.97).*

Note that this frame rate is often different from the frame rate of the encoded video; for example, there can be frame skipping in the encoder, or a lower frame rate selected (such as encoding 30 frames/second video at 15 frames/second).

The Frame Rate value is used to determine which uncompressed video frame to associate with which encoded frame; MTS4EA works out the time code for each uncompressed video frame based upon the Frame Rate value and associates the uncompressed video frame concerned with the encoded frame which has the closest time code.

Metric

This selects the measurement metric to use, which is one of the following:



The objective fidelity metrics provided by MTS4EA measure the degradation of the decoded image with respect to a reference image. The metric is evaluated independently for each image plane in the color space dictated by the video decoder (typically YUV).

The fidelity metrics are calculated either per MacroBlock or overall on the frame as a whole, according to the following equations.

For the explanations of fidelity metrics, the following nomenclature is used:

Symbols	Descriptions
| ... |	Denotes taking the absolute value of an expression
∑	Denotes the summation of an expression over the range of (x, y) in the image plane
f(*x*, *y*)	Is the sample value at the location (x, y) in the reference image plane
g(*x*, *y*)	Is the sample value at the location (x, y) in the decoded image plane
N	Is the total number of samples in the image plane

PSNR (full signal range). The PSNR (Peak Signal to Noise Ratio) metric is defined as the ratio between signal power and noise power, on a decibel scale. In the context of image processing, signal power is taken to be the square of the peak image sample value and noise power is taken as the square of RMS error in the image.

$$PSNR = 10 \cdot \log (S2/ RMS2)$$

For PSNR (full signal range) the peak image value is assumed to be the maximum value for the bit-depth in all three image planes. For example, in an 8-bit image, this implies:

$$PSNR255 = 20 \cdot \log (255/ RMS)$$

PSNR (ITU-R BT.601 signal range). For PSNR (ITU-R BT.601 signal range) the PSNR is calculated differently in Y than in U and V:

$$PSNR_Y = 20 \cdot \log (220 / RMS)$$

$$PSNR_{U,V} = 20 \cdot \log (225 / RMS)$$

RMSE (Root Mean Square Error). The RMS is the defined as the square root of MSE.

$$RMS = \sqrt{MSE}$$

MSE (Mean Square Error). The MSE metric is defined as the mean of the squares of differences between samples in the reference and decoded image planes.

$$MSE = 1/ N \, S \, [f(x, y) - g(x, y)]^2$$

MAD (Mean Absolute Difference). The MAD metric is defined as the mean average of absolute differences between samples in the reference and decoded image planes.

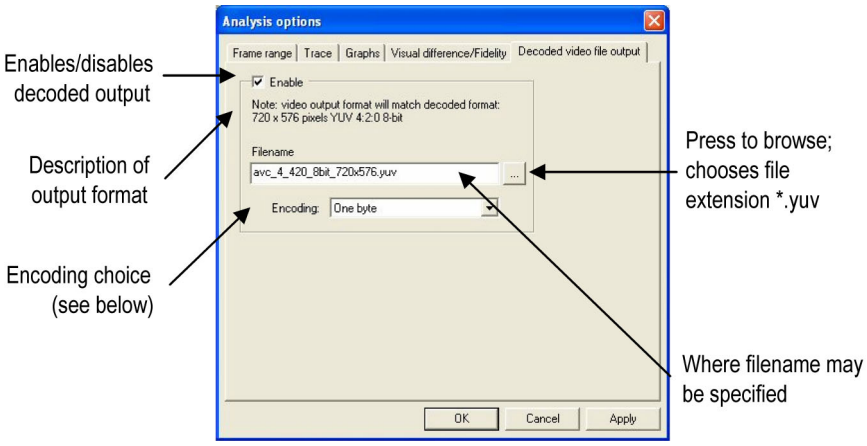
$$MAD = 1/ N \, S \, | f(x, y) - g(x, y) |$$

SAD (Sum Absolute Difference). The SAD metric is defined as the sum of absolute differences between samples in the reference and decoded image planes.

$$SAD = S \, | f(x, y) - g(x, y) |$$

Output decoded video file...

This opens the Decoded output tab of the Analysis options and allows you to write a decoded file to disk.



Encoding

This control specifies the encoding of image samples in the file format. The following options are available:

- One byte - this specifies that image samples are stored in one byte per sample. This format is appropriate if all image planes are 8-bits deep.
- Two byte MSB first - this specifies that image samples are stored in a pair of bytes for each sample. The most significant byte occurs first in each pair (big-endian). This format is appropriate if one or more image planes are deeper than 8-bits.
- Two byte LSB first - this specifies that image samples are stored in a pair of bytes for each sample. The least significant byte occurs first in each pair (little-endian). This format is appropriate if one or more image planes are deeper than 8-bits.

Video analysis options...

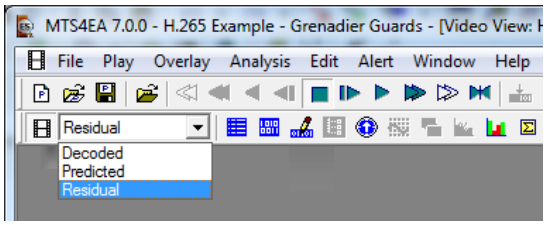
The Video Analysis menu option opens the Video Analysis dialog box; this allows you to select and set up the range of video analysis options.

View video

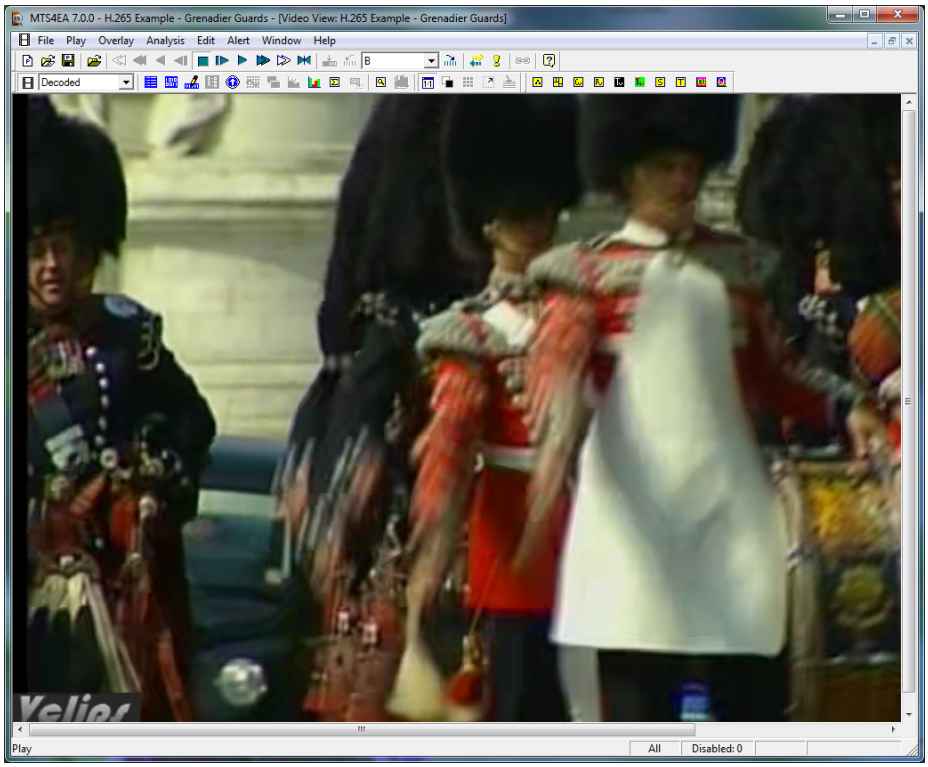
Open the video view.

H.265/HEVC video views

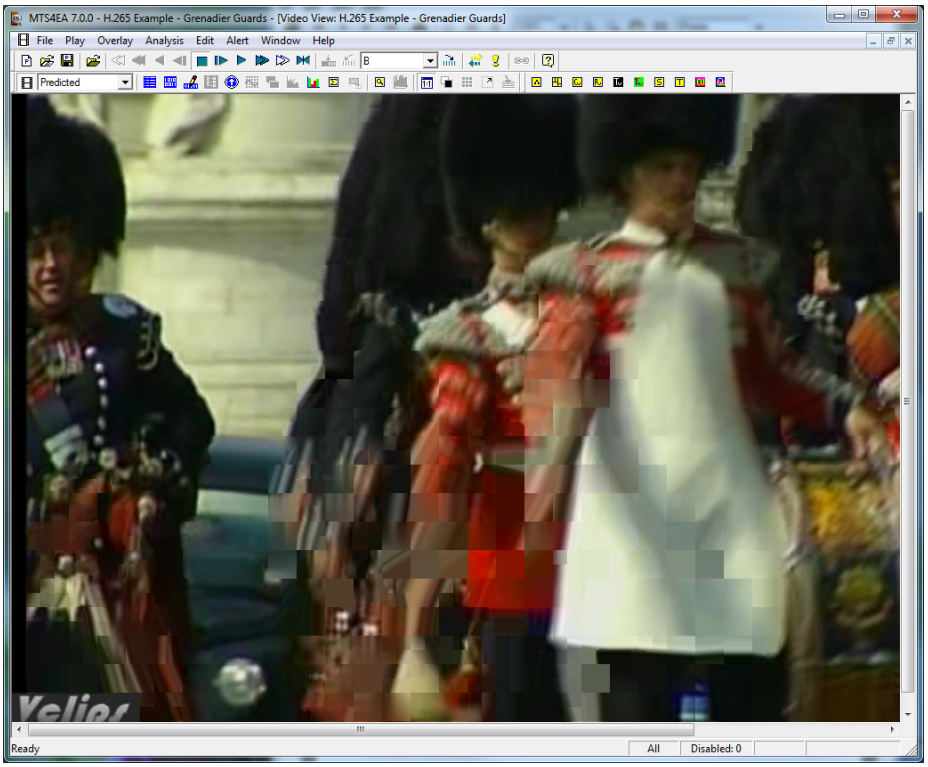
For H.265/HEVC streams only, you can select between showing the Decoded, Predicted, or Residual views using the drop-down box in the toolbar. You can use the Image Inspector to examine the selected view.



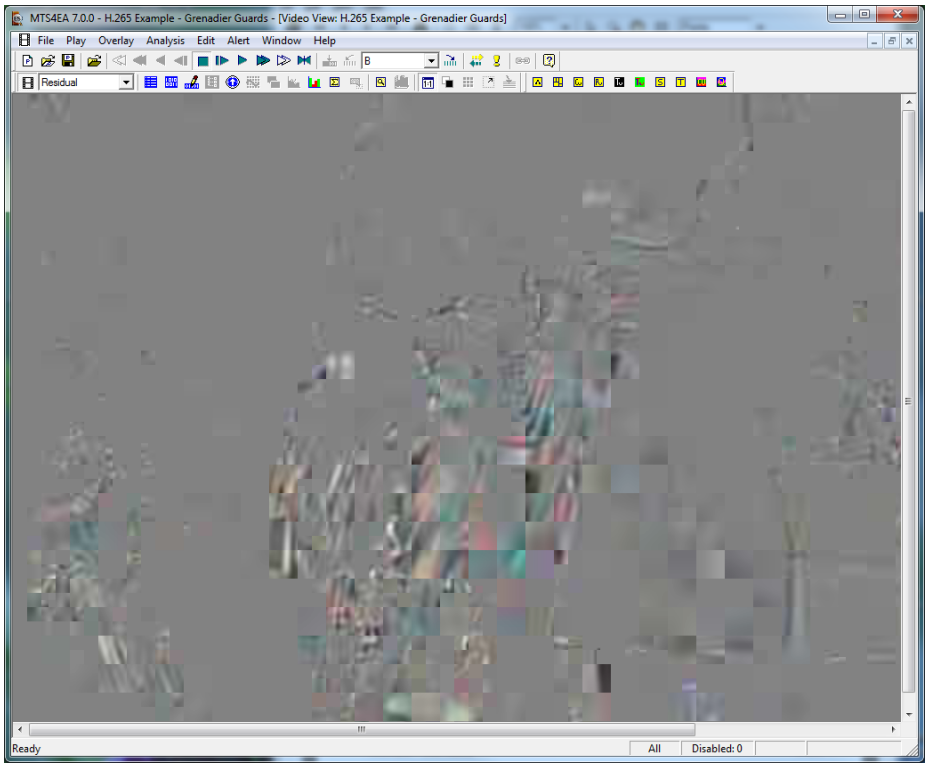
Decoded. The Decoded video view shows the video as a combination of the Predicted and Residual views.



Predicted. The Predicted video view shows the video as derived from the Prediction units in the video frame CTUs.



Residual. The Residual video view shows the video as derived from the residual bits used to encode the CTUs on the video frame.

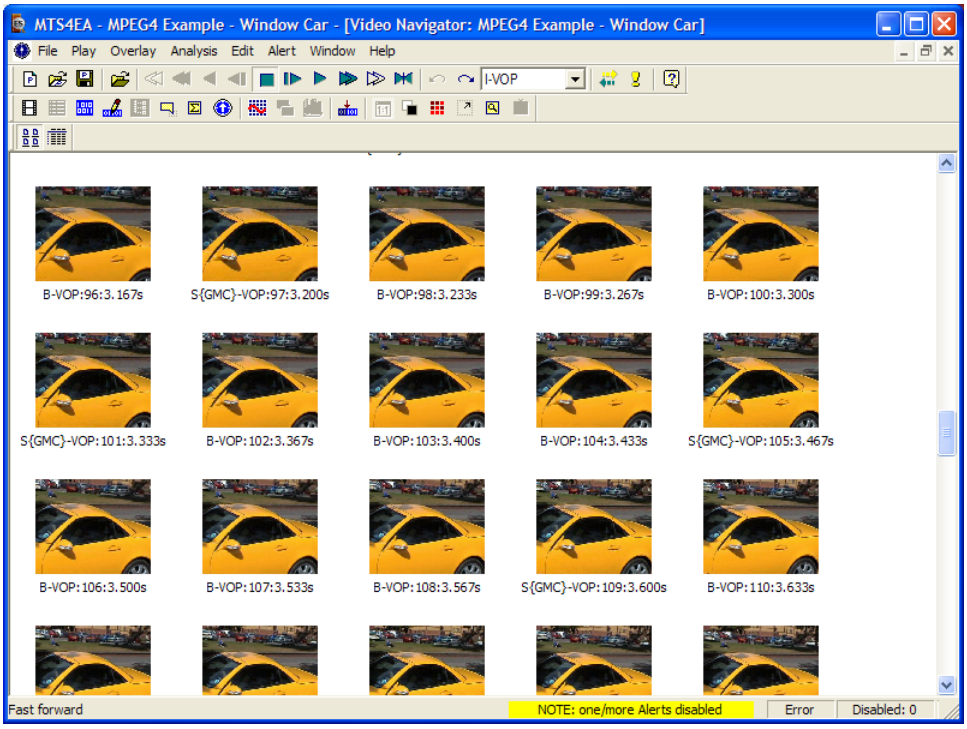


View video navigator

Thumbnail view

This shows a thumbnail view of the frames in a video file, with some basic information.

Two different views can be selected. In either case, a right-click context-sensitive menu allows selection of the other views and navigation to the corresponding frame in the other views (or the start address of the corresponding frame, as appropriate). (See page 5-197, *Right-click popup menu/Goto views*.)



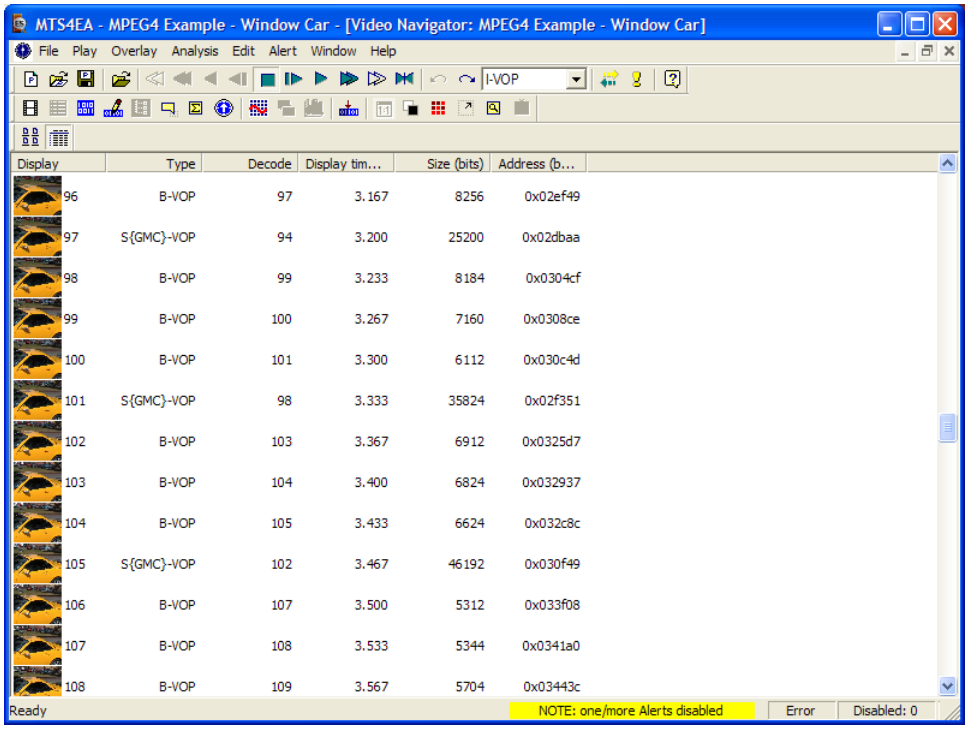
The video frames are shown in display order (decode order can be different if there are B-frames/B-VOPs).

The displayed information is:

- Frame type/VOP type
- Display frame/VOP number
- Display time in seconds

NOTE. *For interlaced bitstreams, the frame/VOP type reported is that relating to the second field in a frame. This means that if the top field is first in frame/VOP 1 (and this will always be Intra-coded) and the bottom field is Inter [P] coded, the frame will be reported as P.*

Detail view



The displayed information is:

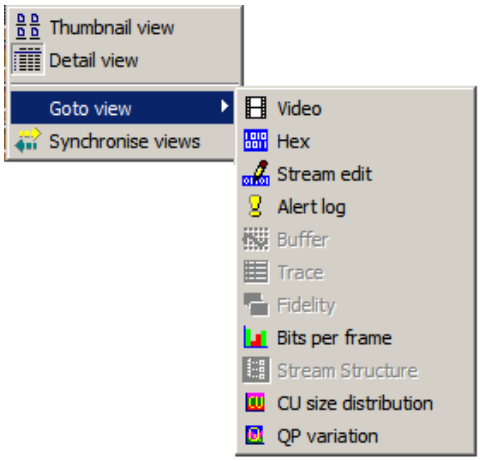
- Display frame/VOP number
- Frame type/VOP type

NOTE. *For interlaced bitstreams, the reported frame/VOP type is that relating to the second field in a frame. This means that if the top field is first in frame/VOP 1 (and this will always be Intra-coded) and the bottom field is Inter [P] coded, the frame will be reported as P.*

- Decode frame/VOP number
- Display time in seconds
- Frame size in bits
- Address of start of frame (in the video elementary stream, not in the overall container file if there is one)

The columns may be resized and dragged into a different order if desired.

Right-click popup menu/Goto views.

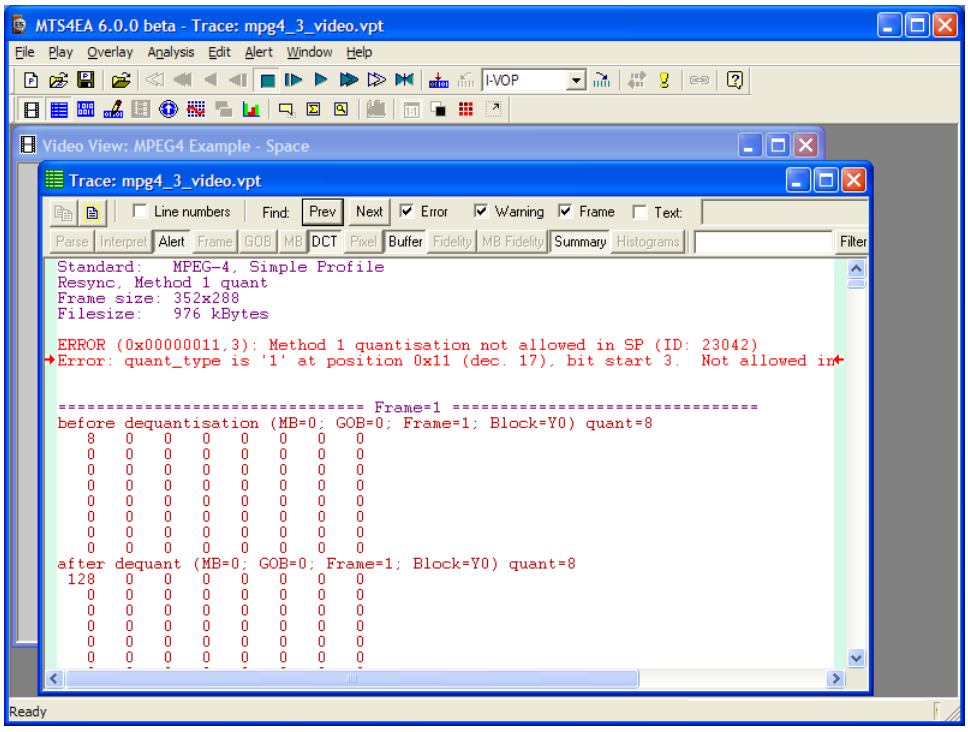


(See page 5-8, *Synchronized views/navigating the views*.)

View video trace Ctrl+Shift+T

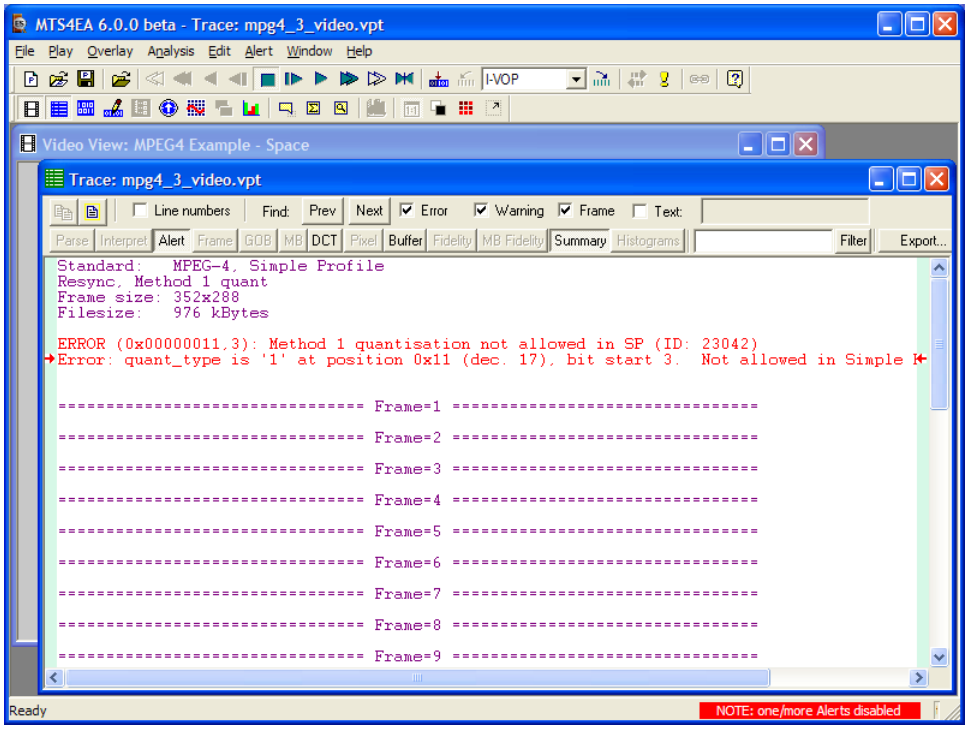
This option opens the Trace view on the current output Trace file, enabled using the Analysis > Video trace enable… option. (See page 5-147, *Video trace enable... Ctrl+T*.)

The screenshot below shows a Trace file in which the following options have been selected when creating the file: Alerts, DCT, Buffer, and Summary.



All available trace options are displayed in the example screenshot. The display of different trace options can be selectively toggled by pressing the relevant button in the Trace view toolbar. Trace options not selected when creating the trace file are greyed out.

In the following example screenshot, the DCT and Buffer trace options have been turned off and are therefore not displayed.



An additional filtering feature is the custom filter; the text box on the right side of the Trace view toolbar. By entering a string in the custom filter text box and selecting Filter, only those lines containing the entered text present in the current display will be displayed. The custom filter is applied to the currently displayed trace data. The filter text box is highlighted when in use. If one of the normal filter buttons is selected, the normal trace display is reinstated.

NOTE. *If there is not a current Trace file (and Trace enable is not enabled) then this menu option/toolbar icon is unavailable (grayed out).*

Trace files created by versions of MTS4EA prior to version 6 will be loaded as plain text, without color highlighting or smart filtering.

If the video stream has been rerun and a Trace file saved with the same name as that already open, then clicking the View trace menu option or *icon or Ctrl+V will refresh the open Trace file.*

Go to other views/right click menu - Interpret & Parse Bitstream only

When you right-click, a context-sensitive menu appears that allows going to the area in another view which corresponds to the area which has been selected in the Trace file.

Copying text

The filtered contents of the Trace window can be copied to the Windows clipboard.

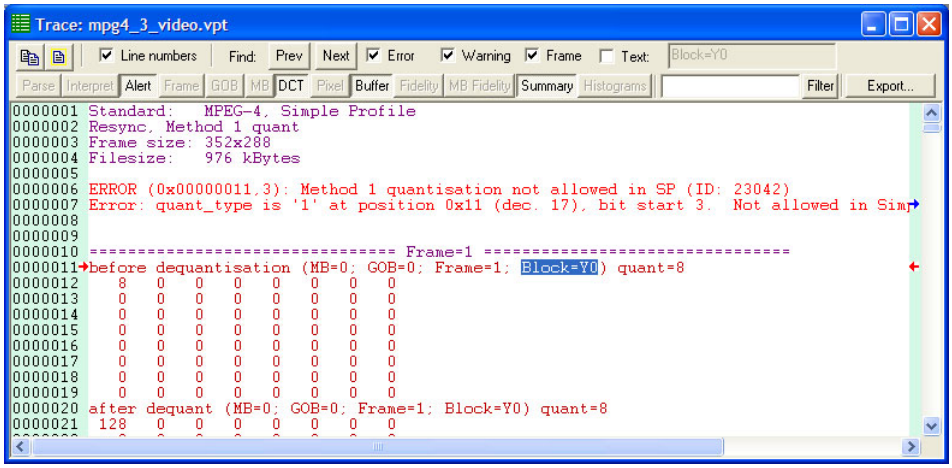
This can be done in several ways:

- By pressing `Ctrl+C`
- By right-clicking with the mouse and selecting Copy
- By clicking the copy icon -

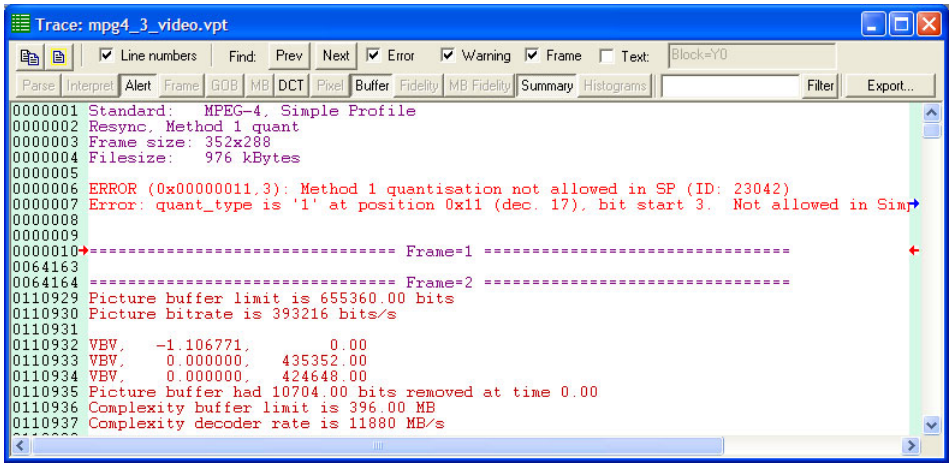
The whole of the contents can be selected by clicking the Select All icon - .

Line numbers

Line numbers can be turned on/off:



Line numbers displayed are those within the whole file (see following screenshot), not just those in the visible, filtered lines.



Find data

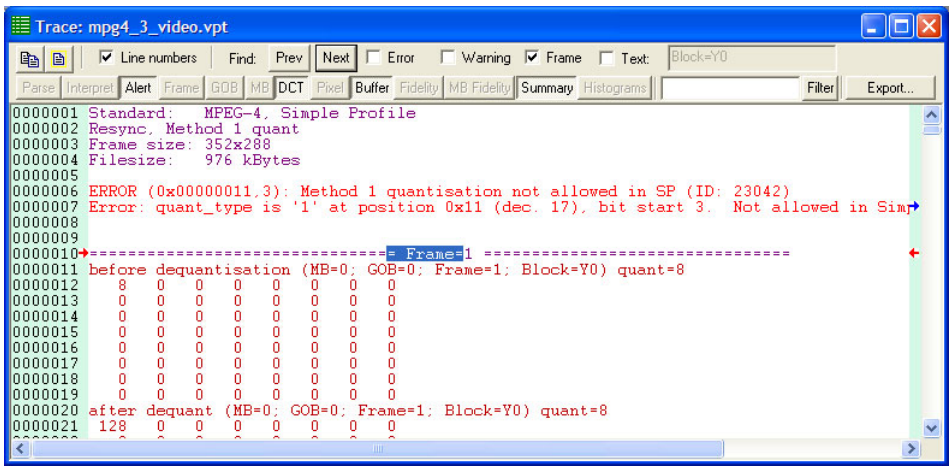
The Trace file can be searched for any data, using the Find: Prev (previous) and Next buttons. This finds the previous/next occurrence of any of the enabled strings. The line containing the result will be indicated by highlighting and red arrows in the margin.

NOTE. *The F3 key can also be pressed, to find next. The Shift+F3 key can also be pressed, to find previous.*

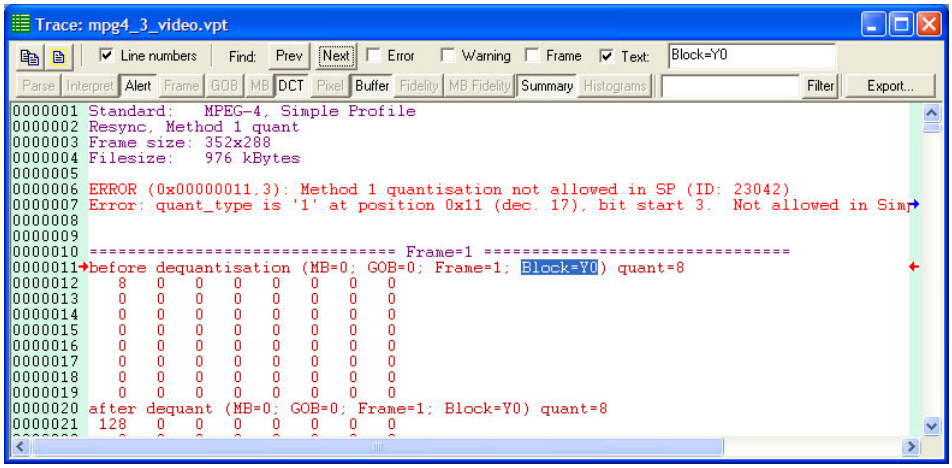
There are some standard strings that are useful to search for:

- Error
- Warning
- Frame

These are found by enabling the relevant check box (Frame example below):



To search for any text, select the box next to Text and enter the text in the box (in this example, “Block=Y0”):



When the current view is filtered, the find feature will still search the whole (unfiltered) file, but if the result is not being currently viewed, the closest visible line will be indicated with red arrows.

Lines too long for window width

If a line is too long to be displayed within the width of the View trace window, then a horizontal scroll bar is displayed and a blue arrow is appended to the right-hand side of the View trace window, to indicate that a line extends to the right.

Trace file export

The MTS4EA saves Trace files with a `.vpt` file extension. Files are exported with the currently filters active; only the displayed content is exported.

Currently, the .vpt file is a standard ASCII file, which could be viewed in any text file viewer. The `.vpt` extension is associated with the MTS4EA in Windows, and by double-clicking on a `.vpt` file in Windows Explorer, the file is automatically opened in the MTS4EA

Trace file import

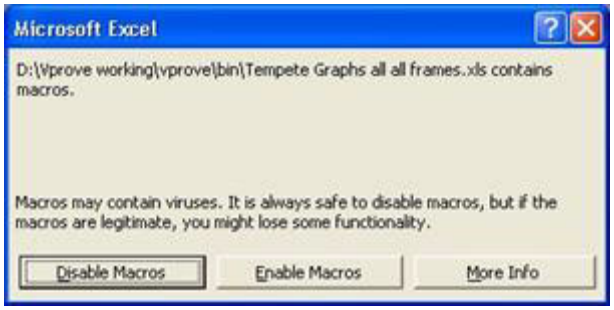
Using the File > Open other… option, previously saved or exported trace files can be opened. Older files will be opened as simple text file; only custom filtering can be applied.

View video graphs

This option calls Microsoft Excel to display the graphs generated: a macro is then run within Excel to create the graphs.

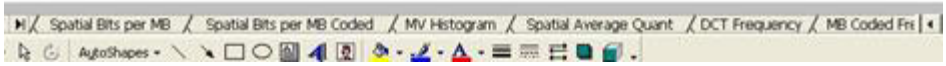
NOTE. *Macros must be enabled within Microsoft Excel to see the graphs.*

The default selection within Microsoft Excel is (usually) to disable macros (as below) - you must click on Enable Macros:

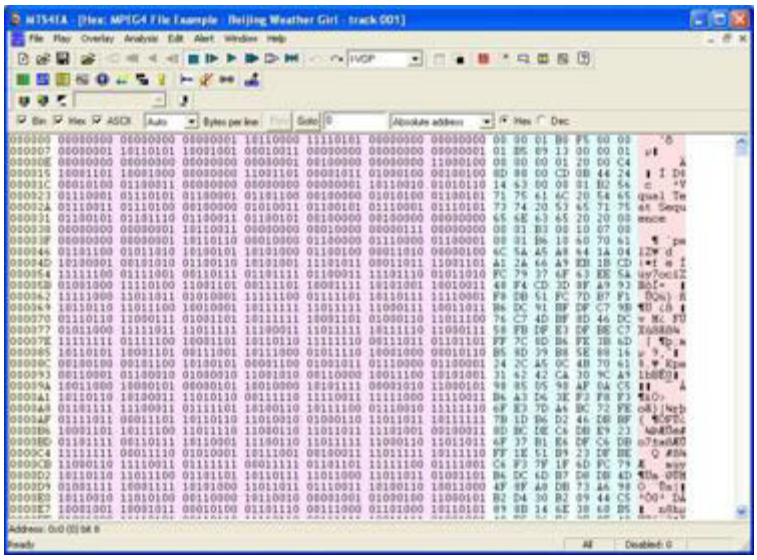


NOTE. *The data used to generate the graphs is available on the Data tab in the Excel file.*

The statistics options selected on the Graph enable tab within MTS4EA are then displayed on individual tabs within Excel:



View video stream hex Ctrl+H



This opens the current video file in a hex viewer, which shows the data in binary, hexadecimal and ASCII data (in any combination of the three).

NOTE. *Multiple View hex windows can be opened at the same time.*

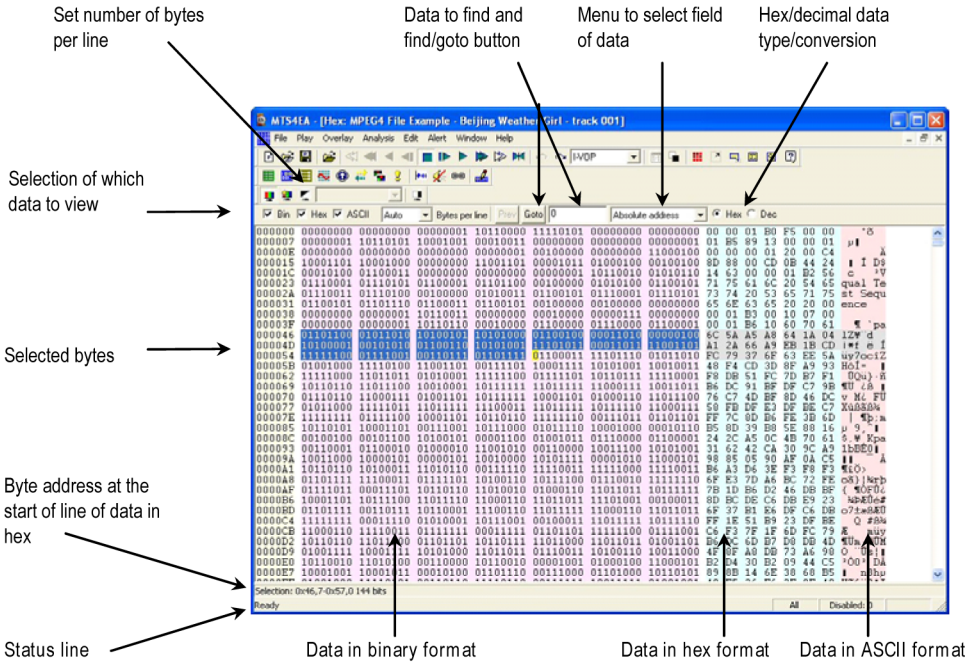
Using HexView, you can scroll through the file, and search for specific:

- Absolute address (from the start of the file)
- Relative address (from the currently selected location
- Bit patterns
- Hex data
- ASCII data

NOTE. *In the field where the data to find is entered, a wildcard character can be entered - this is . [period]*

Also, entering a hex value, then clicking Dec will convert this number to decimal (and vice versa).

Sections of HexView window



Setting information displayed; window width

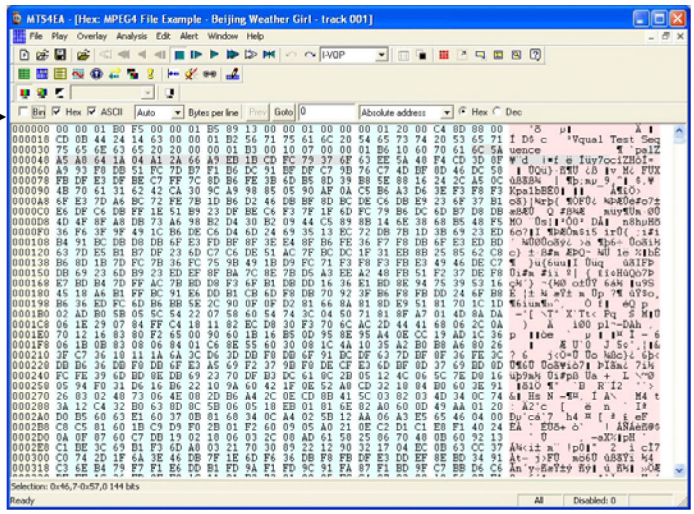
The HexView window can be set (for example, resized) as with any other window in Windows.

The same data is shown in each of the three sections, in their respective formats:

- Binary
- Hexadecimal
- ASCII

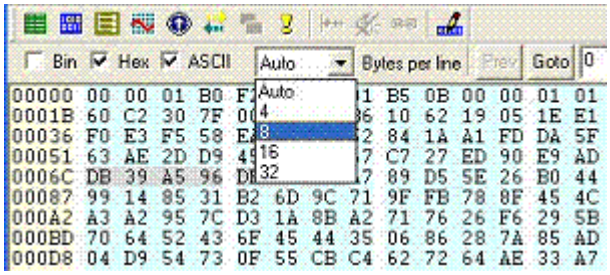
Each of these areas can be individually displayed/not displayed by clicking the appropriate check box in the top left.

In this example, the binary display has been switched off, leaving only hex and ASCII

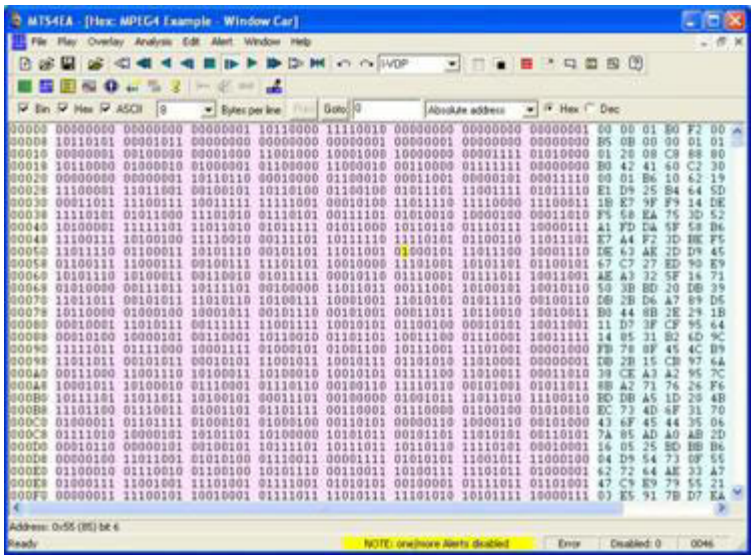


Setting bytes per line

This menu is used to set the number of bytes per line:

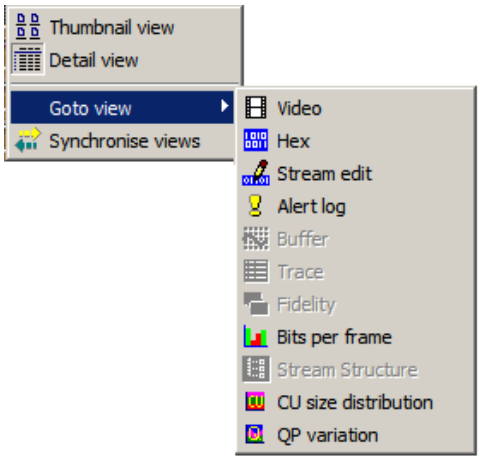


If the current width of the window is insufficient to show all the columns, then a scroll bar automatically appears at the bottom, allowing scrolling left/right to see all the columns, as in the following example:



Right-click popup menu/Goto view

Right-clicking in the HexView window generates the menu:



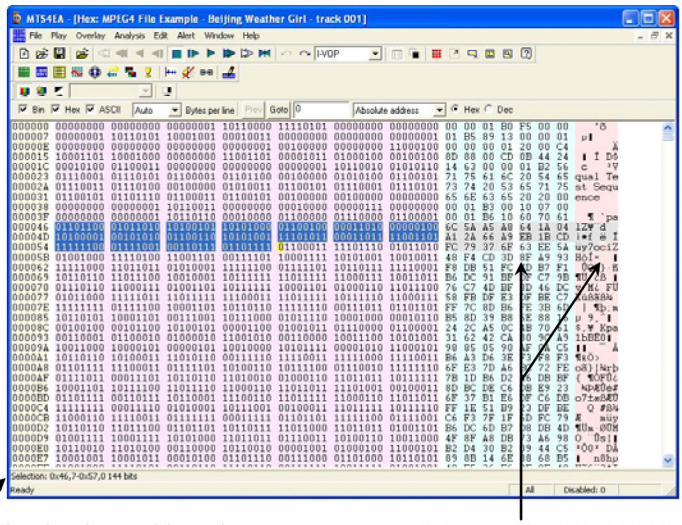
The top four buttons of the menu have the same functions as the buttons given on the top line (and as the HexView settings option on the Window menu).

Goto view. The Goto view takes the focus to the corresponding area in the selected view. (See page 5-8, *Synchronized views/navigating the views*.)

Synchronize views. When the Synchronize views icon is pushed in, then all open windows automatically follow the selection. (See page 5-8, *Synchronized views/navigating the views*.)

Highlighting a section

A section of data can be highlighted in any of the (visible) binary, hex or ASCII sections by clicking and dragging the mouse over the area:



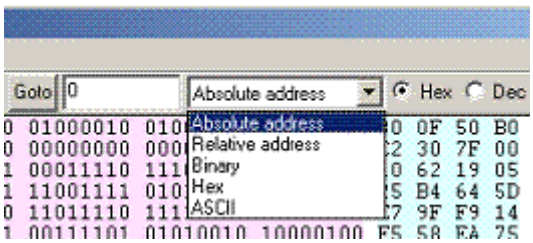
The addresses of the selected area and the number of bytes selected are displayed in the status line

Whichever section is highlighted, the corresponding areas of the other sections are similarly highlighted

A specific address can be found in the video stream, either of the following:

- An absolute address, from the start of the video file
- A relative address, from the first byte of the currently selected area

Find absolute address. To go to an absolute address, select Absolute address from the drop-down menu:



The address to find is entered in the box next to the Goto button. This address can be in hex or decimal. If an invalid character is entered for an address (for example, entering anything other than 0-9 or a-f for a hexadecimal address) then the Goto button is grayed out.

Note that the Goto button changes to Next for all data to find except for the Absolute address. (And when Absolute address is selected, the Prev button is grayed out.)

The Hex/Dec radio button selects the format of the data being searched - Hexademical or Decimal.

Find relative address. An address relative to the currently highlighted address can be found. If no address is highlighted, the address found is the offset from 0.

The address to find is entered in the box next to the Next button. This address can be in hex or decimal. If an invalid character is entered for an address (for example, entering anything other than 0-9 or a-f for a hexadecimal address), the Find button is grayed out.

NOTE. *The F3 key can be used, to find next; Shift+F3 key, to find previous.*

The Hex/Dec radio button selects the format of the data being searched.

Find binary/hex/ASCII. These options in the menu find data in the bitstream.

Up to 64 characters can be entered.

NOTE. *The Binary search searches for the bit pattern regardless of byte location; the Hex and ASCII searches are byte aligned.*

The F3 key can be used, to find next; Shift+F3 key, to find previous.

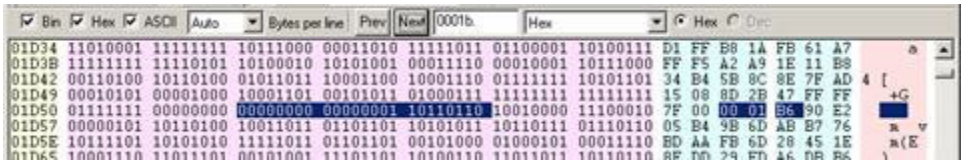
Wildcard searching using period. A wildcard can be entered in the data to be found - . [period].

The wildcard matches a single digit in the selected base; the wildcard matches:

- One bit in the binary find
- A hex digit in the hex find
- An ASCII character in the ASCII find

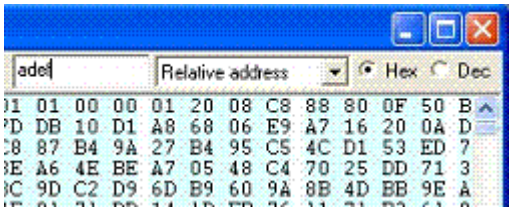
For example, entering '0001B.' as a hex string will find the following patterns in the selected example stream:

The wildcard can be inserted in any position, for example, '03..456.8' is a valid search string, which will find any 9 consecutive digits where the first two are 03, digits 5-7 are 456, and the last digit is 8.



Conversion of hex<->decimal

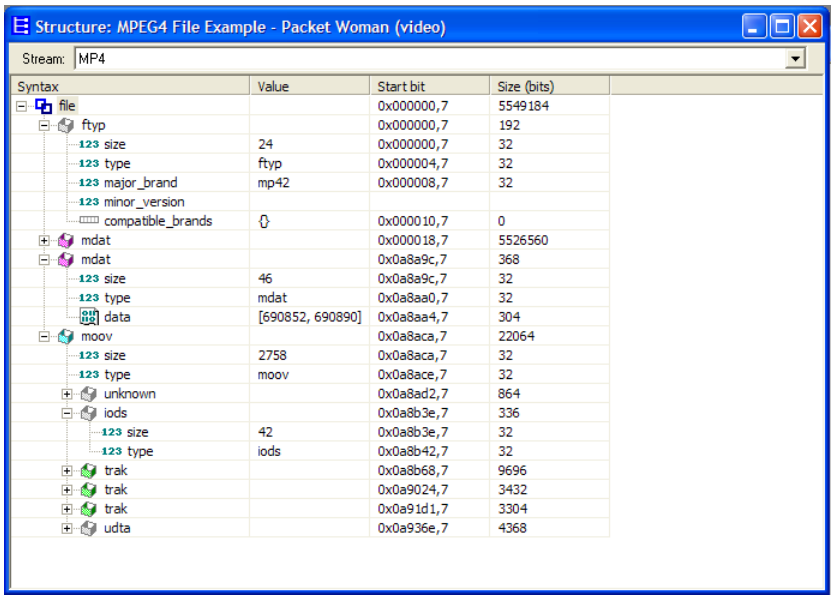
If a hex value is entered into the find box:



When the Dec button is clicked, this value is converted to hex (and vice versa):

View video stream structure Ctrl+R

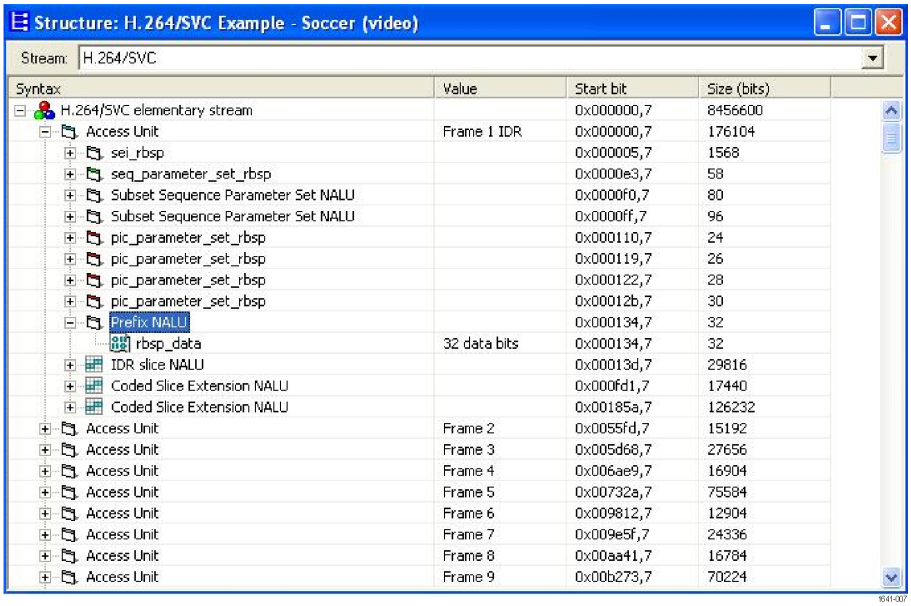
This allows the structure of all supported container formats to be viewed and expanded/contracted (by clicking on the - and + symbols at the left end of each line). In addition, for H.264/AVC video elementary stream format, the view shows the top-level structure of the stream.



The view is divided into four columns:

- **Syntax**: Shows the stream syntax parse tree, including the names of each syntax element or syntactic structure. Nested syntax structures are shown by nesting nodes of the tree. An icon shows the type of data at each tree node. (See page 5-211, *Icons in Stream structure view*.)
- **Value**: Shows the interpreted value of syntax elements in the tree. The content of this field will vary according to the type of data to be shown.
- **Start bit**: Shows the start bit address of the syntax node. The address is given as a hexadecimal byte position followed by a bit position number.
- **Size**: Shows the size in bits of the syntax node. For nodes with children, this shows the total bits of the node and all its children.

The following example shows the video stream structure for a H.264/SVC stream:



Icons in Stream structure view

In the Stream structure view, the icons that appear at the node of each branch of the tree structure denote the type of data contained in that node. There are several icons which are common to all standards as described in the following table.

Table 5-1: Icons general to all standards

Icon	Meaning
	Video elementary stream
	Audio elementary stream
	Container stream
	Packet or other syntactic grouping (various colors used – see standard specific icons)
	Binary data
123	Integer data field
101	Binary data field
01X	Exponential-Golomb coded data field
ab	String data field
{ID}	Identifier data field
	Time data field
	Custom data field

Table 5-1: Icons general to all standards (cont.)

Icon	Meaning
	Unknown data field
	User data
	Array based data structure
	List based data structure
	Video ES slice data
	Video ES macroblock data

The following table show icons that are specific to different standards.

Table 5-2: Standard specific icons

Standard	Icon	Meaning
H.264/AVC		IDR access unit
		Non-IDR access unit
		‘moov’ box type
		‘mdat’ box type
		‘trak’ box type
		Generic MP4 descriptor
MPEG-2 TS		Transport packet
		Program association section
MPEG-2 PS, PES		System header
		Pack header
		Pack syntactic structure
ASF		Header object
		File properties object
		Stream properties object
		Index object

View video buffer analysis

This allows the buffer usage to be analyzed in detail. The analysis varies by video standard:

For H.265/HEVC the analysis is of:

- Performance using the HRD at the picture level (Access Unit level - displayed in red)
- Performance using the HRD at the sub-picture level, if available (Decoding Unit level - displayed in dotted blue)

For H.264/AVC the analysis is of:

- Performance using the Hypothetical Reference Decoder (HRD) (displayed in red)

For MPEG-4 the analysis is of:

- Video Buffer Verifier (displayed in Red)
- Video Complexity Verifier (displayed in Green)
- Video Memory Verifier (displayed in Blue)

For MPEG-2 the analysis is of:

- Video Buffer Verifier (displayed in Red)

The icon controls and general look of the buffer analysis is the same for all standards:

- The buffer analysis controls that are common to all standards are explained in the sections beginning *Buffer analysis toolbar icons* to *Buffer analysis alerts/scroll bar area* (although some of the diagrams in these sections are specific to a standard regarding the titles and data content, the functions are the same across all standards). (See page 5-214, *Buffer analysis toolbar icons*.) (See page 5-218, *Buffer analysis alerts/scroll bar area*.)
- MPEG-4 and MPEG-2 VBV analysis are very similar and are explained in the sections beginning *Buffer analysis controls: MPEG-4 and MPEG-2* to *Buffer analysis popup alerts: MPEG-4 and MPEG-2*. (See page 5-219, *Buffer analysis controls: MPEG-4 and MPEG-2*.) (See page 5-221, *Buffer analysis popup alerts: MPEG-4 and MPEG-2*.)

- The H.264/AVC HRD analysis is distinct and is explained in the sections beginning *HRD buffer analysis: H.264/AVC* (See page 5-221.) to *HRD buffer overflow/underflow indication: H.264/AVC* (See page 5-224.).
- The H.265/HEVC HRD analysis is distinct and is explained in the sections beginning *HRD buffer analysis: H.265/HEVC* (See page 5-222.) to *HRD buffer overflow/underflow indication: H.264/AVC* (See page 5-224.).

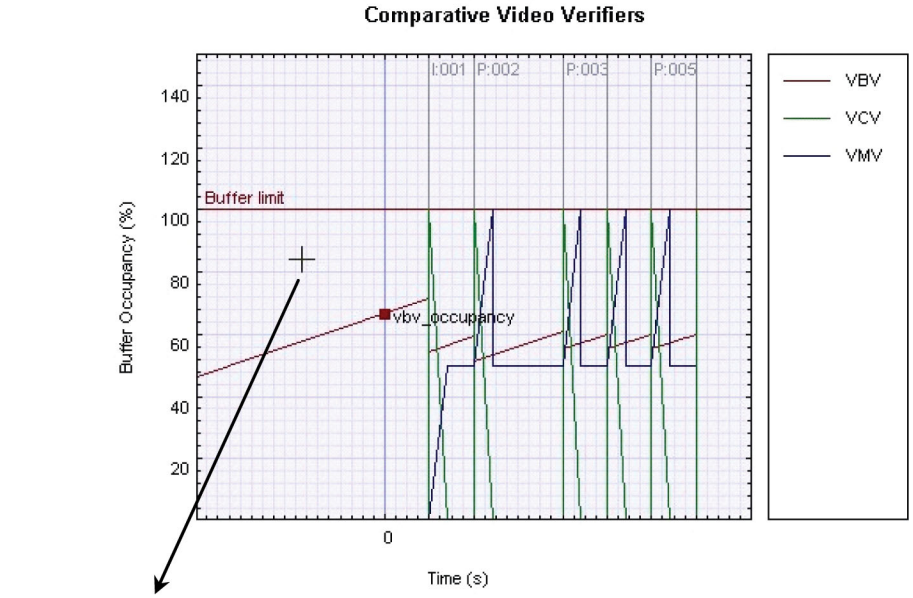
Buffer analysis toolbar icons

Icon	Function
	Zoom in (+) and zoom out (-) centered on the center of the window (affects scale of x-axis only)
	Fit all data into window
	Locate origin (zero), start of sequence
	Lock X/Y zoom in/out and scrolling/panning. For example, when the Lock Y button is pressed, zoom in and zoom out and scroll/pan only affect the X-direction. This allows (for example) you to keep a useful vertical scale, while still viewing the whole length of the video sequence
	Autoscroll (fill the analysis data in real-time) as the video is being decoded and scroll the window to the right
	Increase track height (affects scale of y-axis only)
	Decrease track height (affects scale of y-axis only)
	Measure the data at the cursor. The data values are reported on the status line at the bottom of the analysis window Offsets and angles/slopes of lines can also be measured, by holding the mouse and dragging
	Scroll/pan (the cursor changes to show the scroll/pan direction)
	Zoom in/zoom out centered on the location of this cursor. Press the <shift> key to zoom out

These functions are also available on the right-click menu.

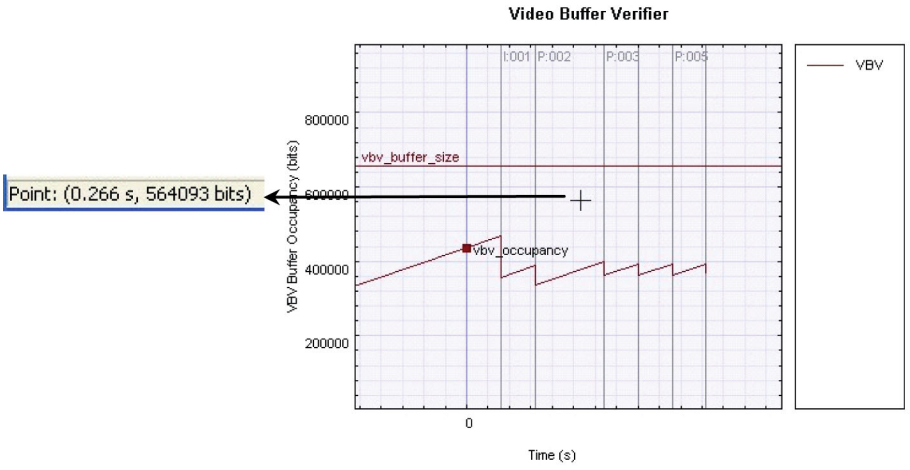
Using the measure tool

Moving the measure tool over the graph displays the values in the status line:



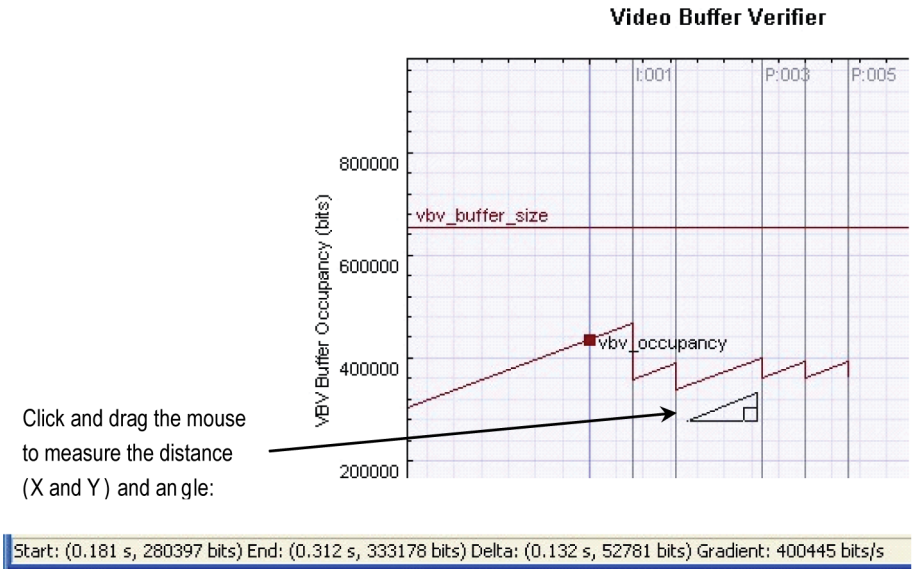
Point: (-0.149 s, 84 %)

The left-hand value is the X-axis (time) value in seconds, to the nearest millisecond; the right-hand is the Y-axis value: either in percentage terms , if there is more than one graph; or in the correct units, if there is only one graph (as below)

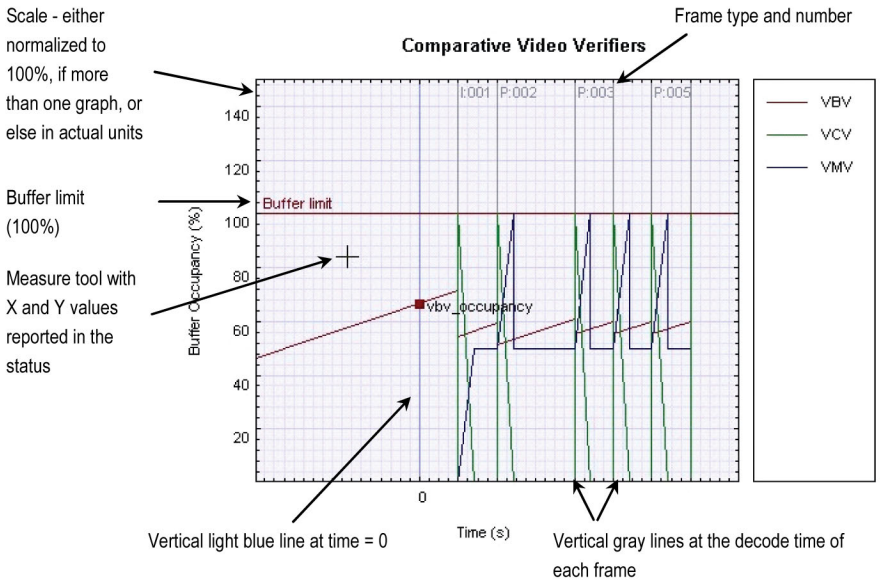


Point: (0.266 s, 564093 bits)

(Only the VBV is shown here)

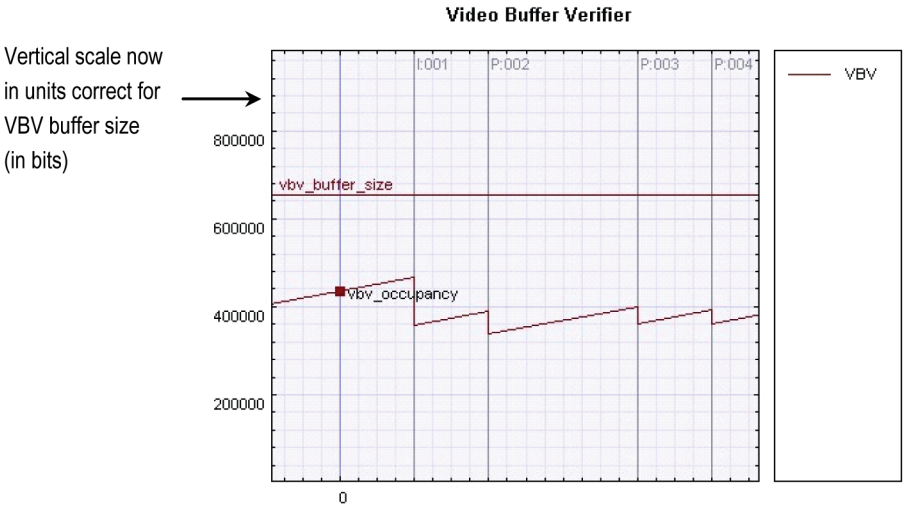

Video Buffer Verifier
I:001
P:003
P:005
800000
600000
400000
200000
VBV Buffer Occupancy (bits)
vbv_buffer_size
vbv_occupancy
Click and drag the mouse
to measure the distance
(X and Y) and an gle:
Start: (0.181 s, 280397 bits) End: (0.312 s, 333178 bits) Delta: (0.132 s, 52781 bits) Gradient: 400445 bits/s

Buffer analysis graph area



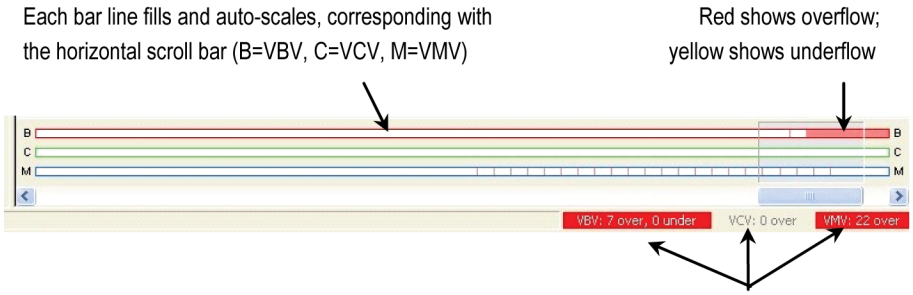
The left axis displays:

- Values normalized to 100%, if there is more than one graph
- Values appropriate for that graph (see the following figure)

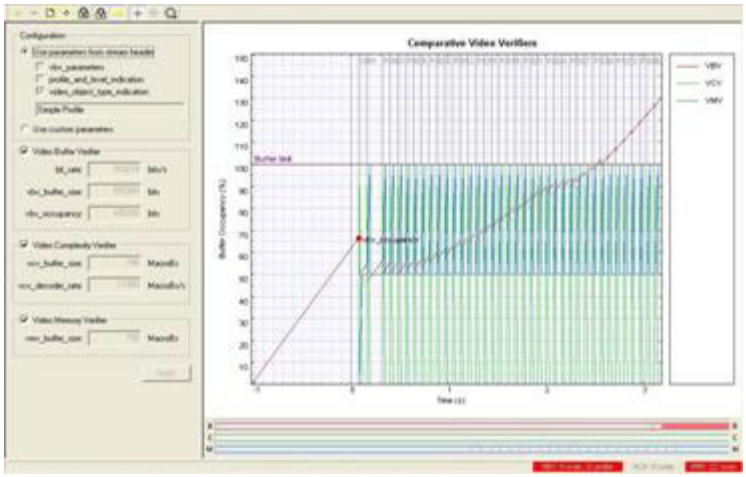


When the buffer data is too wide/too high for the current window, scroll bars appear at the bottom/right (as appropriate).

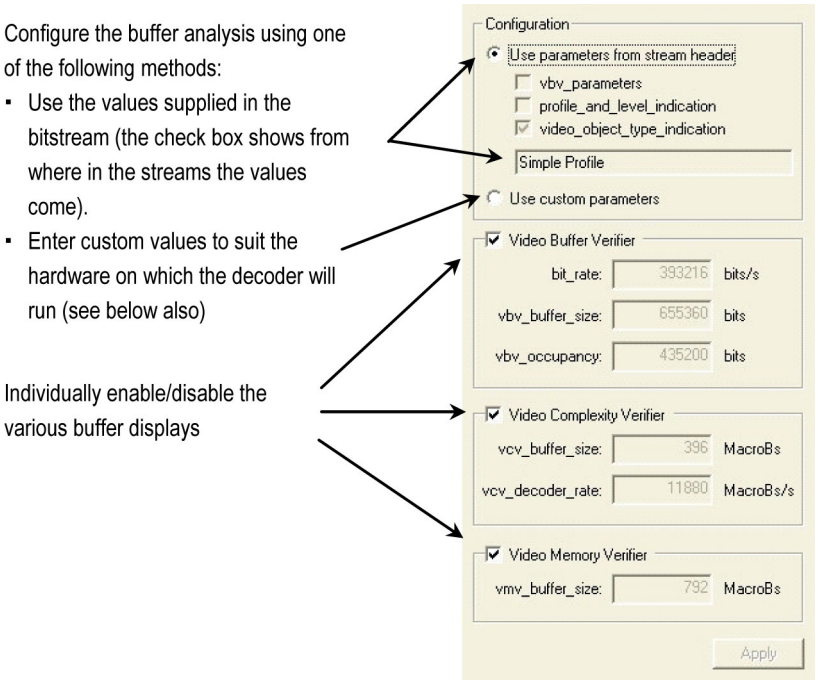
Buffer analysis alerts/scroll bar area. The buffer overflow/underflow is indicated in the Graph window as shown below:



MPEG-4 and MPEG-2 buffer analysis. All the example screenshots and explanations below are for MPEG-4; similar displays and information are used for MPEG-2.



Buffer analysis controls: MPEG-4 and MPEG-2.



The values used for VBV, VCV and VMV are displayed, but cannot be altered unless Use custom parameters is selected.

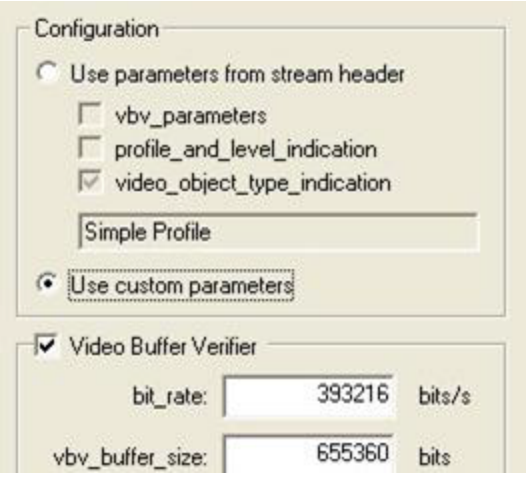
The boxes below the title Use parameters from stream header show where the values used have come from: in the example above, there were no values specified in the `vbv_parameters`, nor was there a `profile_and_level_indication`, so the maximum value implied by Simple Profile as given in the `video_object_type_indication` is used.

NOTE. *There is an order of precedence: if the `vbv_parameters` are specified in the bitstream, then these values are used; if not, then the maximum values are used as allowed in the Profile/Level given by the `profile_and_level_indication`. If there is no `profile_and_level_indication`, then the maximum values are used as implied by the `video_object_type_indication`.*

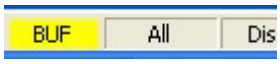
Use custom buffer parameters: MPEG-4 and MPEG-2. Custom parameters can be entered for a variety of reasons, for example:

- The VBV/VCV/VMV parameters have not been specified in the bitstream: in this case the MTS4EA will assume the maximum allowable values for the Profile/Level and these may be too large for the particular operation.
- The VBV/VCV/VMV parameters specified in the bitstream do not reflect the actual limitations of the hardware on which the decoder will have to decode the bitstream.
- To try different values to see if the conformance parameters are met with different limits.

Custom parameters are entered as below:



When custom parameters are used, this is indicated in the status bar of the main MTS4EA window by the word BUF:



NOTE. *These values can also be selected/entered from the Decoder options selection on the Play menu (the MPEG-4 tab).*

The values entered as Custom parameters affect both the alert popup warnings and the warnings that are reported in the Trace files; the Custom parameters are the values used to trigger these warnings.

To reset the values entered to the values specified by the bitstream, reselect the Use parameters button from the stream header.

Buffer analysis popup alerts: MPEG-4 and MPEG-2. Where the bitstream exceeds the values given by:

- `vbv_parameters`
- `profile_and_level_indication`
- The custom parameters

When each of these is selected as the source of the parameters, popup alerts occur during video decoding unless disabled in the Alerts menu (the video must be playing for the popup alerts to be generated).

NOTE. *When the buffer parameters are implied by* `video_object_type_indication`*, no popup alert is generated in the event of exceeding the limits, nor is there a warning in the Trace files. This is in accordance with the implied rules in the MPEG-4 standard.*

HRD buffer analysis: H.264/AVC. The Hypothetical Reference Decoder (HRD) in H.264/AVC provides a complex mechanism for buffer analysis; this is much more complex than the buffer analysis in MPEG-4/MPEG-2.

NOTE. *The HRD analysis is of the Coded Picture Buffer (CPB) only, not of the Decoded Picture Buffer (DPB).*

In order to understand the HRD analysis, you must read and understand Annex C of the H.264/AVC standard, ISO document 14496-10 (E).

HRD parameters do not have to be specified in a bitstream; many H.264/AVC streams do not have them specified (only the Bus Junction and Grenadier Guards H.264/AVC example streams provided with the MTS4EA have HRD information).

NOTE. *If HRD parameters are not provided in an H.264/AVC bitstream, the icon and menu item for View buffer analysis is grayed out.*

HRD buffer analysis: H.264/SVC. HRD buffer analysis is not supported for H.264/SVC streams. When you open a H.264/SVC stream, the icon and menu item for View buffer analysis is grayed out.

HRD buffer analysis: H.265/HEVC. The H.265/HEVC HRD analysis is similar to the H.264/AVC HRD analysis at the picture level. In addition, it can contain some parameters, at the subpicture level. The operations at the subpicture level are similar to the ones at the picture level, except that they apply to a subset of NAL units (called "Decoding Units"), comprising the entire picture.

The HRD operation is explained in detail in Annex C of the HEVC standard.

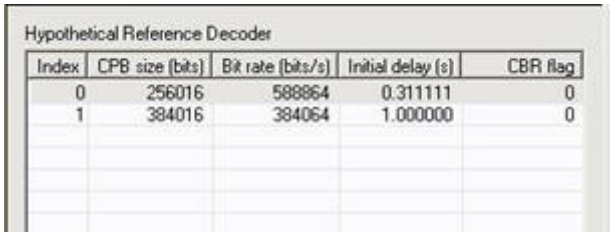
HRD parameters do not have to be compulsorily specified in a bitstream; many HEVC streams do not have them specified.

NOTE. *If HRD parameters are not provided in a HEVC bitstream, the icon and menu item for the View buffer analysis is grayed out.*

HRD list of schedule indexes (buckets): H.264/AVC and H.265/HEVC. Due to the complexity of HRD analysis, entering custom parameters is not currently possible in MTS4EA; the display shows the list of indexes or buckets that are given in the bitstream.

If the HRD parameters are given in a bitstream, there can be 1 to 31 indexes or buckets, numbered from 0.

In the following example below, there are two indexes/buckets: index 0 is selected and is used for the graph display, as shown. Clicking on the line below would display the data from index 1 in the graph display.



The meanings of the column headings are:

Index: The bucket or index number.

CPB size (bits): The size in bits of the Coded Picture Buffer (CPB).

Bit rate (bits/s): The bit rate of the hypothetical transmission channel.

Initial delay (s): The initial delay of the hypothetical transmission channel before the first frame is decoded.

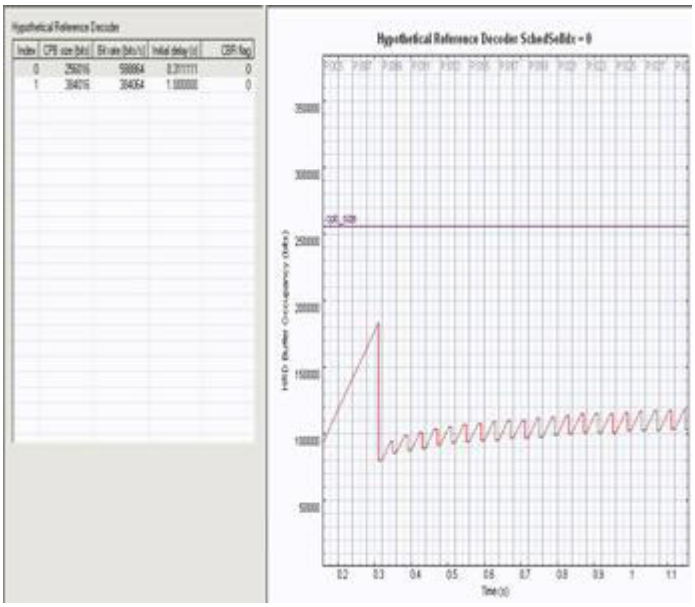
CBR flag: The status of the Constant Bit Rate (CBR) flag: 0=off; 1 = on.

Subpicture HRD details: H.265/HEVC. The Number of DUs column specifies the number of Decoding Units contained in each frame. The meanings of the other columns are similar to the picture level details.

Sub-pic HRD:

Index	CPB size (bits)	Bit rate (bits/s)	Initial delay (s)	Number of DUs
0	368640	614400	0.480200	4

HRD results display: H.264/AVC and H.265/HEVC. When a bitstream has HRD parameters specified, and the HRD buffer analysis has been done by MTS4EA, clicking on the appropriate index/bucket number displays the graph for that index/bucket (in the following example, index/bucket number 0):

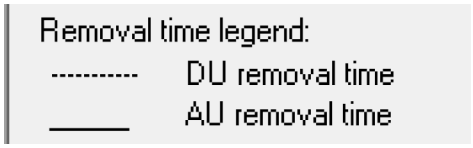


The areas of the graph display are explained in *Buffer analysis graph area* (See page 5-217.) and *Buffer analysis alerts/scroll bar area* (See page 5-218.).

As examples:

- Frame type and number is given at the top of the graph (in light gray)
- The vertical lines correspond with the decode times of the relevant frame

NOTE. *In case of H.265/HEVC, the dotted blue vertical lines correspond to the decode/removal times of the appropriate Decoding Unit (DU).*



HRD buffer overflow/underflow indication: H.264/AVC and H.265/HEVC. HRD buffer analysis overflow and underflow is indicated in the following locations:

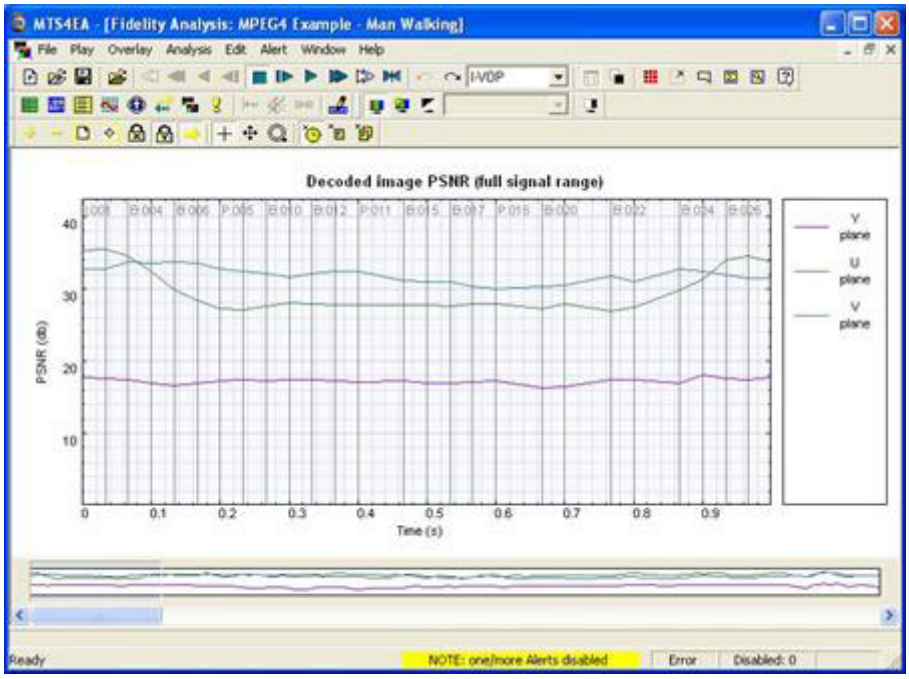
- The bar line below the graph area
- The status bar
- Popup alerts

The bar line and status bar indications are the same as overflow and underflow indications for MPEG-4/MPEG-2. (See page 5-218, *Buffer analysis alerts/scroll bar area*.)

The popup alerts for HRD buffer overflow/underflow are shown on-screen and enabled/disabled as with any other popup alert. (See page 5-263, *Explanation of Fatal/Error/Warning/Info display*.)

View video fidelity analysis

This displays the fidelity analysis graph view:



The fidelity analysis graph view is similar to the graph view used for buffer analysis in MPEG-4, MPEG-2 and H.264/AVC. Both views have the same:

- Control icons (See page 5-214, *Buffer analysis toolbar icons*.)
- Method of displaying frame types, frame numbers, frame times in the graph area. (See page 5-217, *Buffer analysis graph area*.)
- Means of measuring angles (See page 5-215, *Using the measure tool*.)
- Right-click popup menu

The fidelity analysis results are shown separately for Y, U and V planes:

- Y-plane in mauve
- U-plane in green
- V-plane in blue

NOTE. *The MPEG-4 example stream Man Walking has the YUV reference file supplied with MTS4EA: therefore this is a useful example to choose to examine the fidelity analysis functions.*

The first ten frames of the YUV reference file are supplied for the Grenadier Guards example streams that are provided with MTS4EA both as an H.264/AVC example and as an MPEG-2 example.

Where the frame rate is incorrectly set for the uncompressed video file, then the correspondence between the encoded video frames and the uncompressed video source frames is lost; this will substantially reduce the fidelity analysis values.

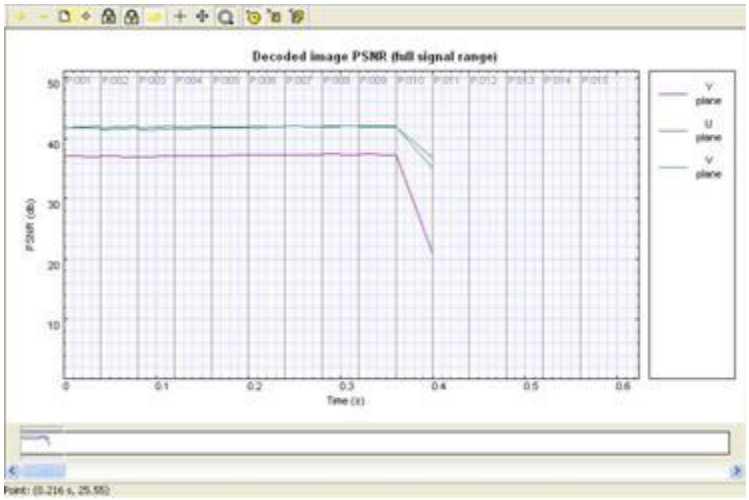
Fidelity analysis view icons toolbar

The icons above have the same functions as the icons on the buffer analysis toolbar; see *Buffer analysis toolbar icons*. (See page 5-214.) The icons below are unique to the Fidelity analysis view:

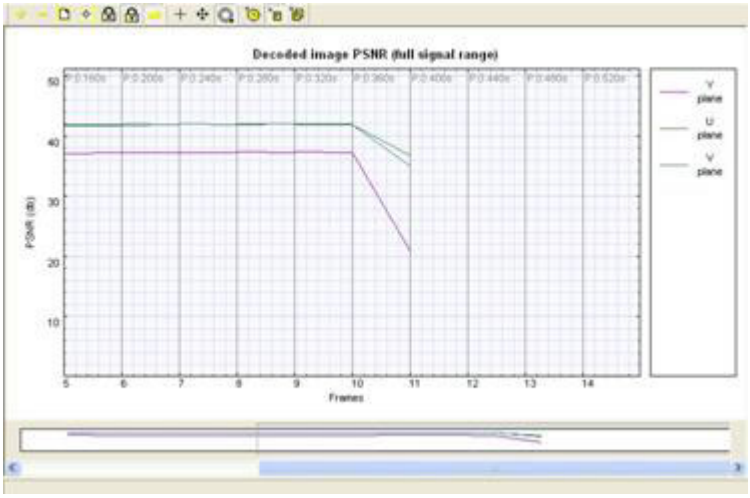
Of these three, the first two are available at all times. The third is only available when interlacing is in use and the picture can be broken down into units; it is grayed out at all other times.

These icons represent the following:

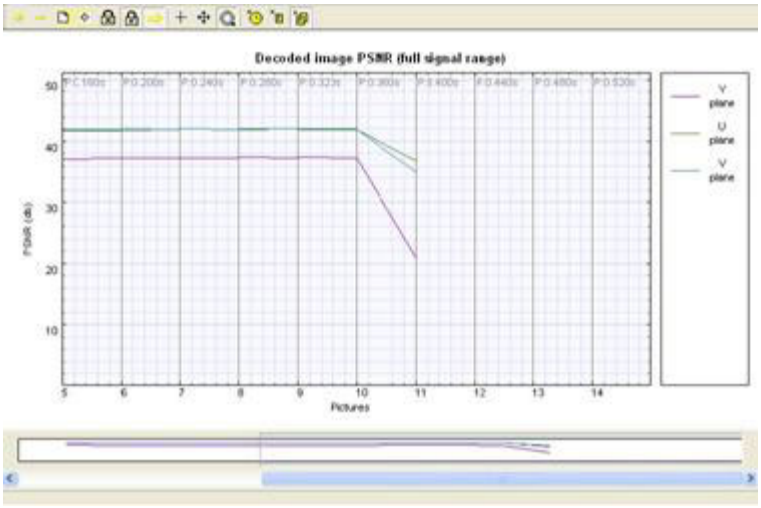
The ability to plot time in seconds on the x-axis, as shown:



The ability to plot frame number on the x-axis, as shown:



The ability to plot picture units on the x-axis, as shown:



Fidelity metrics available. Six different fidelity metrics are available:

- PSNR (255 signal range)
- PSNR (ITU-R BT.601 signal range)
- RMSE (Root Mean Square Error)
- MSE (Mean Square Error)
- MAD (Mean Absolute Difference)
- SAD (Sum Absolute Difference)

(See page 5-189, *Metric*.)

View bits per video frame analysis

This displays the Bits per Frame Analysis graph.

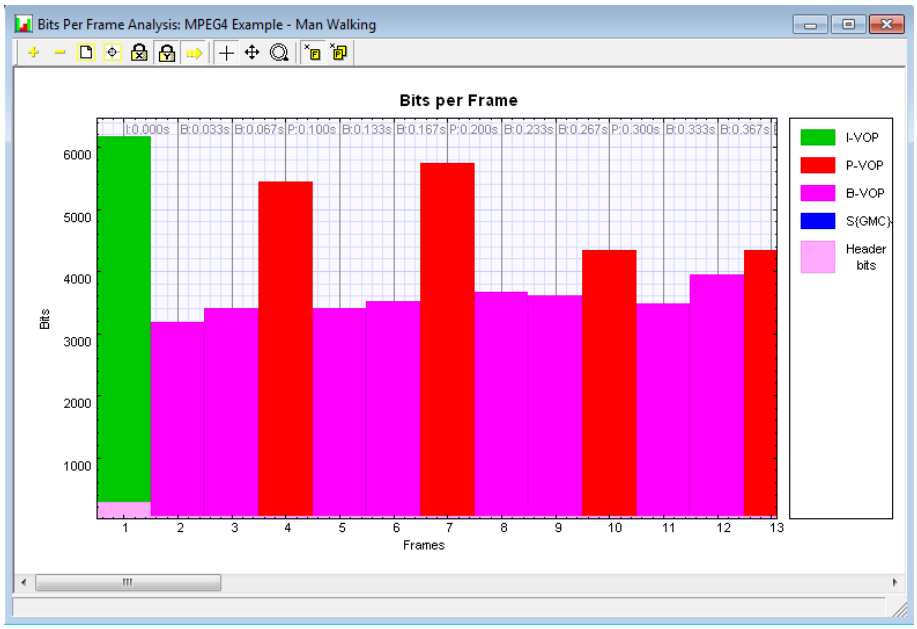


Figure 5-21: Bits per Frame Analysis graph – MPEG-4 example

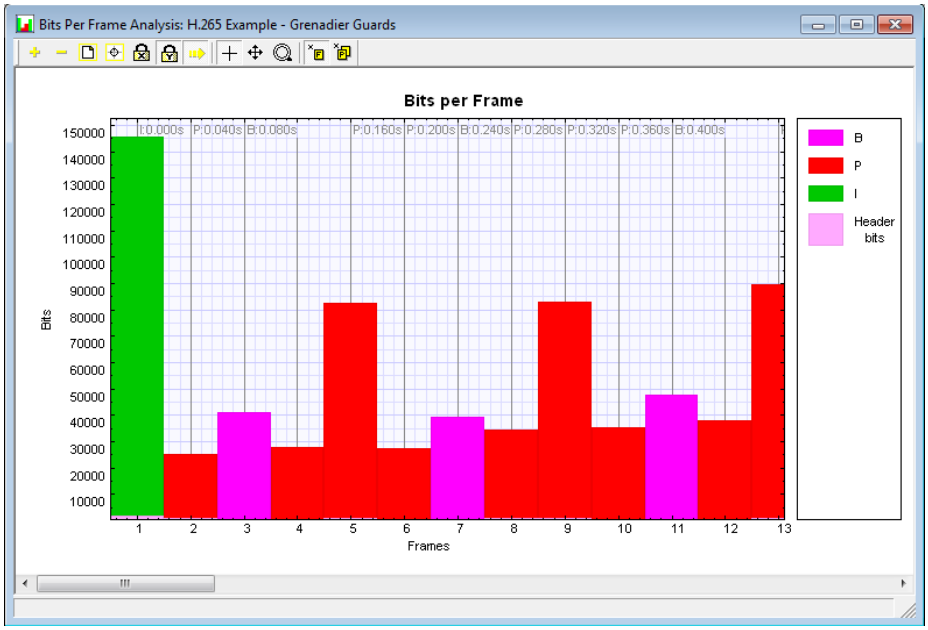
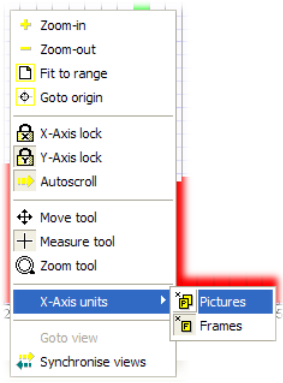


Figure 5-22: Bits per Frame Analysis graph – H.265/HEVC example

Depending on the stream format, the key (on the right side) gives a color code for the VOP/frame types; for more information see *VOP type [MPEG-4] or Frame type [H.265/HEVC, H.264/AVC, MPEG-2, H.263, VC-1].* (See page 5-71.)

In addition to the standard buttons, the toolbar contains two buttons that allow the X-axis to be either frame or picture oriented:



The Frame unit toolbar button displays the total bits per frame; for interlaced pictures this would be the sum of the picture fields:

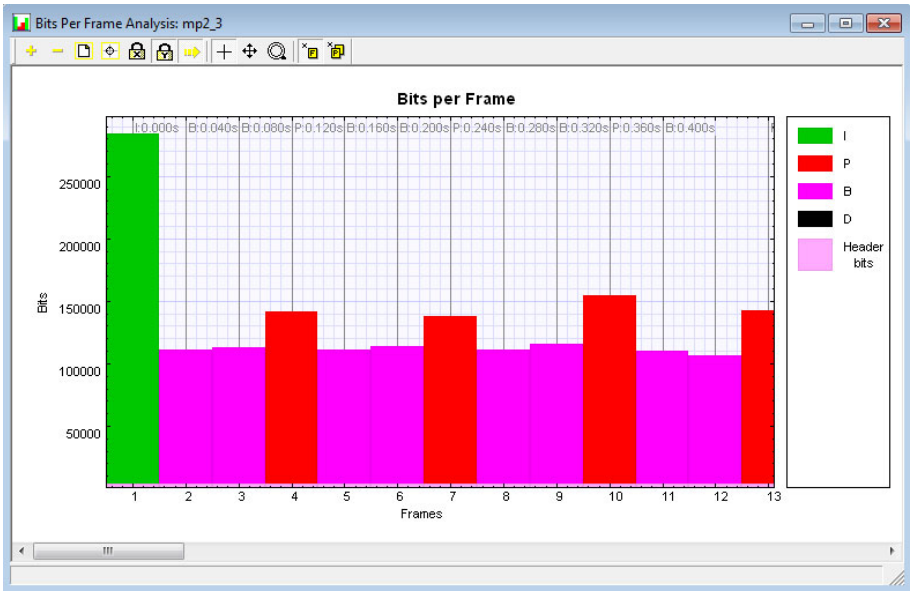


Figure 5-23: Bits per Frame Analysis graph – MPEG-2 example of Frame units display

The Picture units toolbar button also displays the total bits per frame, but, in the case of interlaced pictures, each frame contains two picture fields:

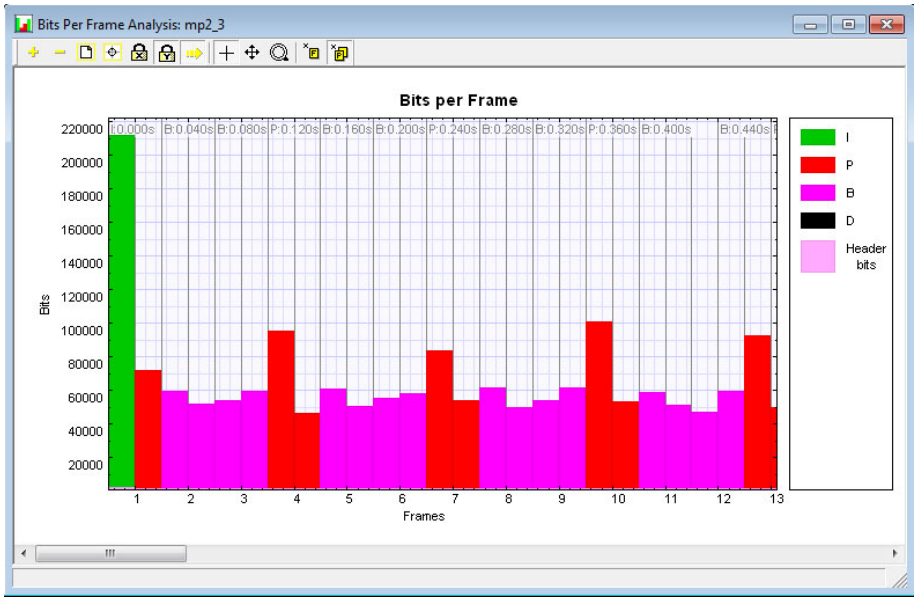
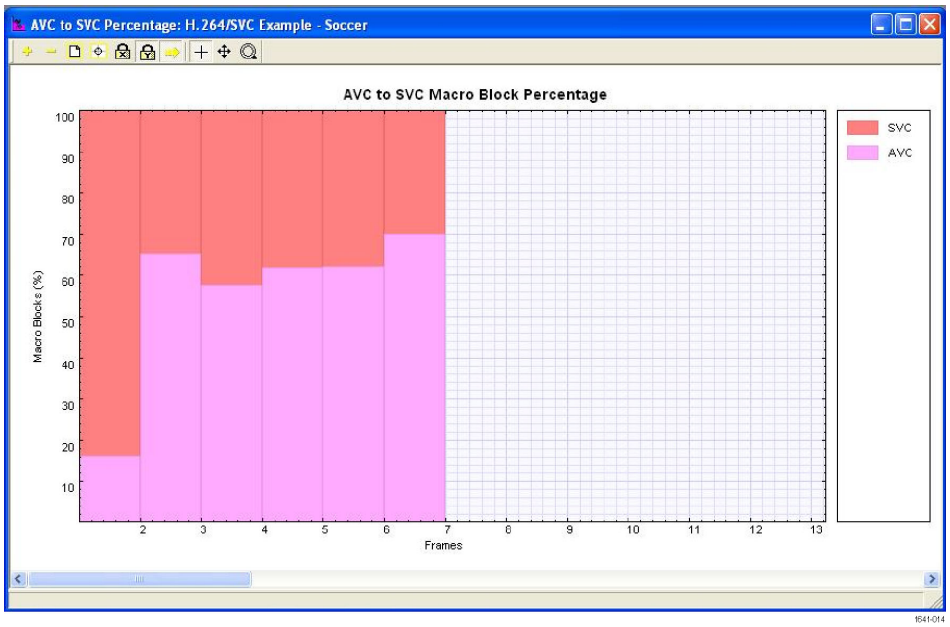


Figure 5-24: Bits per Frame Analysis graph – MPEG-2 example of Picture units display

The bits per frame can be measured by selecting the toolbar measurement tool, placing the crosshair cursor over the point to be measured, and reading the result from the left side of the status bar.

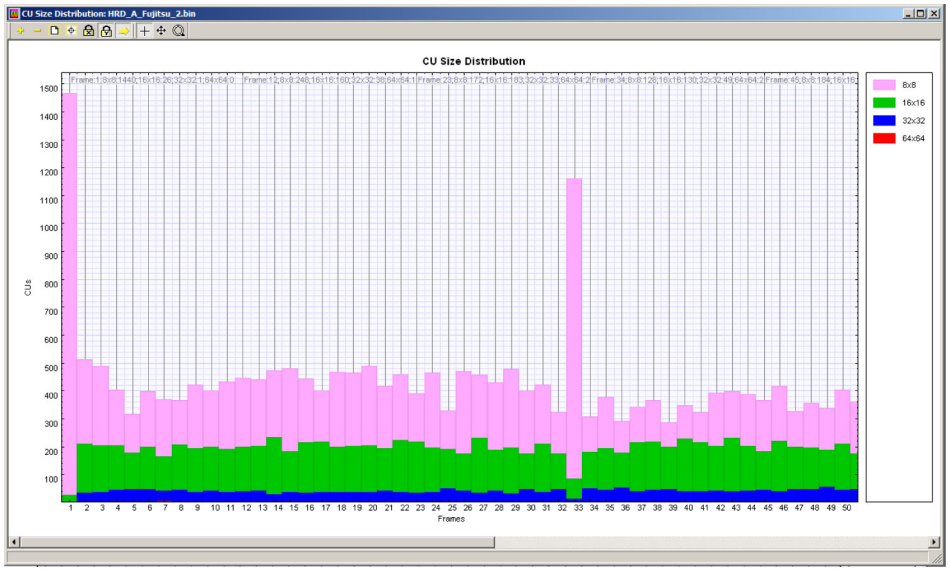
View AVC to SVC MB percentage view

This displays the AVC versus SVC MB percentage for the current picture. The key (on the right side) gives a color code for the MB types (SVC and AVC).



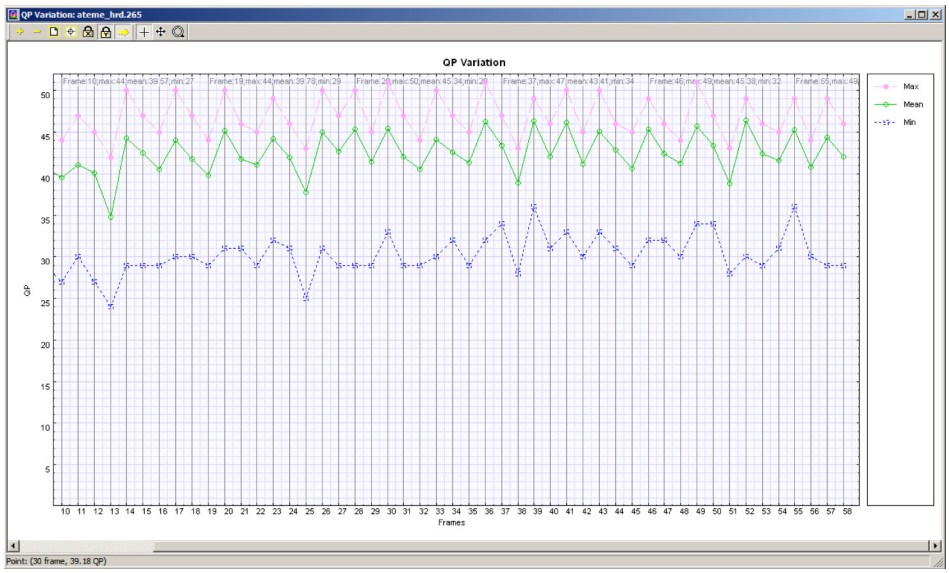
View CU size distribution

For H.265/HEVC, this displays the CU Size Distribution analysis graph. The key (on the right side) gives a color code for the different CU size types.

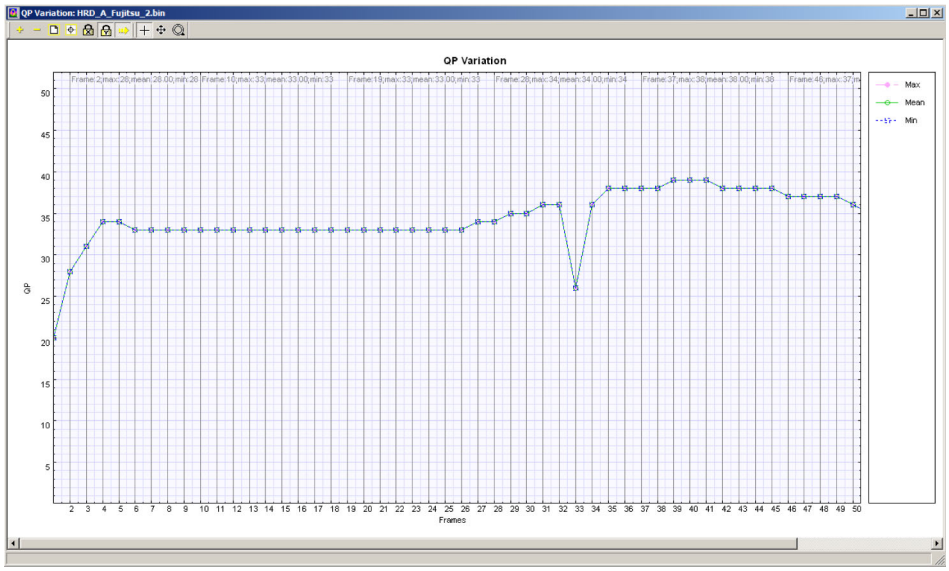


View QP variation

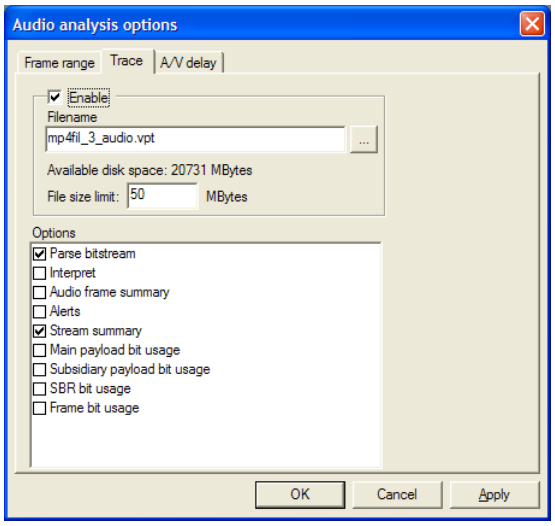
For H.265/HEVC, this displays the Quantization Parameter Variation analysis graph. The key (on the right side) gives a color code for the Maximum, Mean and Minimum QP values within a frame.



In the case when the Maximum, Mean and Minimum QP values are the same, the graph will appear as shown below.



Audio trace enable...



The Trace options provide text outputs, which describe the contents of an encoded stream, to various levels of detail. The available options vary with each audio standard, see following sections (unless otherwise stated, each option applies to every audio standard).

NOTE. *Using these options can generate a lot of data - files hundreds of megabytes or more in size. It may be advisable to collect the data on a range of frames, not all (this is a selectable option).*

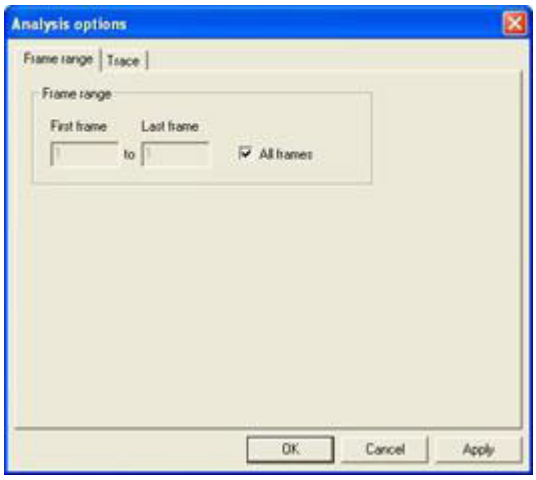
Alternatively, you can set a maximum file size, after which no more trace data is output to the selected file.

NOTE. *The audio can be stopped and the Trace file opened immediately after going past the last frame number in the frame range – there is no need to wait until the end of the sequence.*

NOTE. *To set the range of frames over which to collect the Trace data, click on the Frame range tab.*

Frame range tab

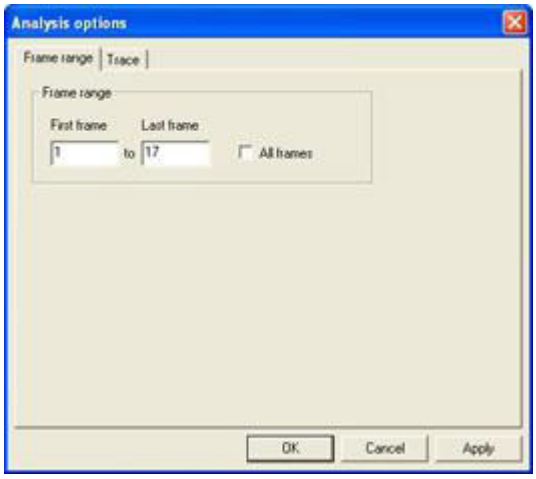
This option allows you to specify the start and end frames between which to gather the Trace information.



NOTE. *Frame range is unavailable in pause mode (indicated when the Pause icon is pushed in on the toolbar and Play menu). This is because the frame range over which statistics are to be accumulated cannot be changed during pause mode. (If it could, you would not be sure that the Trace information was collected over the correct range of frames.) For example, if the audio was paused at frame 23 and Trace was enabled for a range of frames from 20-25, the Trace data would be incorrect.*

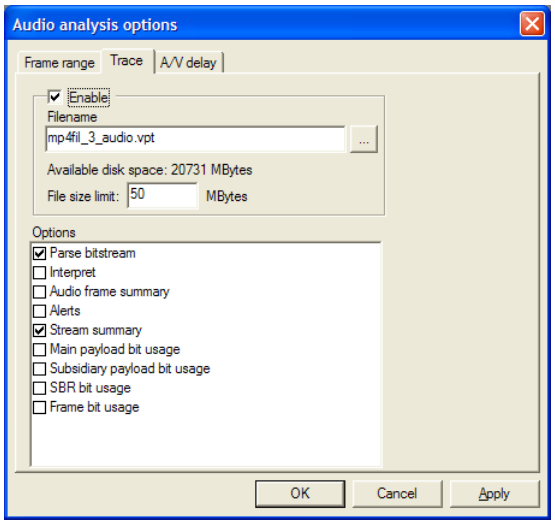
All frames. This option is the default and calculates the sequence statistics from the first to the last frame.

To set a range of frames. If the All frames check box is cleared, a range of frames can be entered (in this case, frames 1 to 17 inclusive):



Trace tab

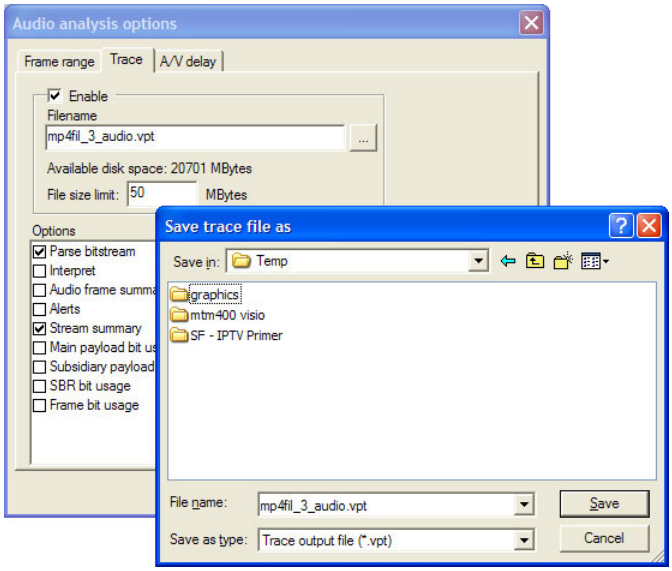
Enable. This option enables/disables the Trace output. If the output is disabled, the remaining selections on the Trace tab are unavailable, although the values are stored during the current use of MTS4EA (they are reset to their default values the next time the MTS4EA is started).



Filename. By default, the filename suggested is the same as the base name of the input audio file, that is, the audio filename without the extension.

Any other filename can be entered; MTS4EA will append an .vpt extension to this filename if it does not already end in this (.vpt = MTS4EA trace).

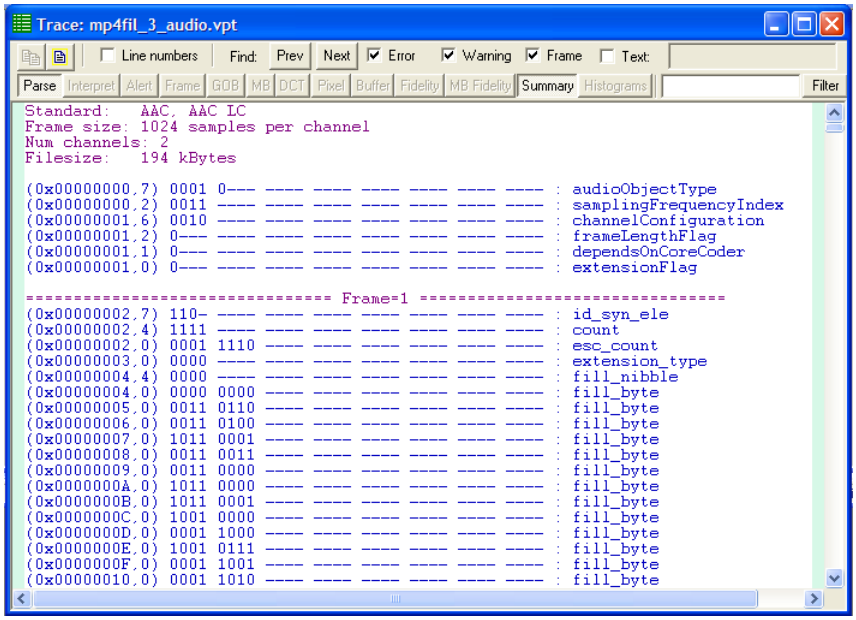
To browse to a specific folder and enter the filename there, click on the [...] button:



File size limit (available disk space). Trace files can generate a lot of data. This means that collecting data over many frames can easily generate a Trace file hundreds of megabytes in size.

This entry allows you to limit the amount of disk space taken by the Trace file. Once the Trace file reaches this size, it stops writing more data.

Parse bitstream. If this option is selected, then the bitstream is parsed to a file. This gives the individual bit patterns for the bitstream fields as well as the mnemonic used within the standard to identify the field.



The preceding example is the Parse bitstream output for an MPEG-1 Audio Layer II stream.

To quickly find if there are any errors in the Trace file, simply use the Find next button function and search on either ‘Error’ or ‘Warning’.

NOTE. *The data from the output of Parse bitstream is one of the best ways that MTS4EA provides to do bitstream syntax debugging.*

Go to other views/right click menu. A context-sensitive menu, brought up by a right-click of the mouse, is available in Interpret and Parse bitstream files. It allows you to:

- Copy selected trace data to the clipboard.
- Go to the area in another view that corresponds to the area that was selected in the Trace file.

NOTE. *The right-click 'Goto view' context-sensitive menu is only active in the Trace/Interpret and Trace/Parse Bitstream trace files.*

Interpret. This provides interpretation of the decoded data in the bitstream. Example outputs are shown below for MPEG-1:

```
(0x00000000.7) [AUS:HDR] syncword = 0xfff
(0x00000001.3) [AUS:HDR] ID = 1 : Mpeg1 Audio (ISO/IEC 11172-3)
(0x00000001.2) [AUS:HDR] layer = 2 : Layer II
(0x00000001.0) [AUS:HDR] protection_bit = 1 : CRC protection absent
(0x00000002.7) [AUS:HDR] bitrate_index = 10 : Bitrate is 192 kbits/s
(0x00000002.3) [AUS:HDR] sampling_frequency = 1 : Sampling frequency is 48000 Hz
(0x00000002.1) [AUS:HDR] padding_bit = 0 : Padding absent
(0x00000002.0) [AUS:HDR] private_bit = 0
(0x00000003.7) [AUS:HDR] mode = 0 : Stereo
(0x00000003.5) [AUS:HDR] mode_extension = 0 : For joint stereo, subbands 0-3 coded in stereo, 4-31
(0x00000003.3) [AUS:HDR] copyright = 0 : No copyright
(0x00000003.2) [AUS:HDR] original_copy = 0 : Copy, not original
(0x00000003.1) [AUS:HDR] emphasis = 0 : No emphasis present
(0x00000004.7) [AUS:AUD] allocation = 5 : 63 quantization levels, no sample grouping
(0x00000004.3) [AUS:AUD] allocation = 5 : 63 quantization levels, no sample grouping
(0x00000005.7) [AUS:AUD] allocation = 3 : 15 quantization levels, no sample grouping
(0x00000005.3) [AUS:AUD] allocation = 3 : 15 quantization levels, no sample grouping
(0x00000006.7) [AUS:AUD] allocation = 3 : 15 quantization levels, no sample grouping
(0x00000006.3) [AUS:AUD] allocation = 3 : 15 quantization levels, no sample grouping
(0x00000007.7) [AUS:AUD] allocation = 5 : 15 quantization levels, no sample grouping
(0x00000007.3) [AUS:AUD] allocation = 4 : 9 quantization levels, 3 samples grouped in one code
(0x00000008.7) [AUS:AUD] allocation = 4 : 9 quantization levels, 3 samples grouped in one code
(0x00000008.3) [AUS:AUD] allocation = 4 : 9 quantization levels, 3 samples grouped in one code
```

NOTE. *To quickly find if there are any errors in the Trace file, use the Find Prev/Next buttons on the trace view toolbar to navigate to the next/previous error or warning.*

Go to other views/right-click menu. By right-clicking with the mouse, a context-sensitive menu appears (Interpret & Parse Bitstream only), which allows you to:Copy selected trace data to the clipboard.

Go to the area in another view which corresponds to the area which has been selected in the Trace file.

Audio frame summary. This Trace output provides a series of data items dependent on the audio standard; for each individual audio frame, it will output data as listed here. All items after the first are per channel, and this portion of the list is repeated once for each channel in the audio stream. The n in square brackets is replaced by the index of the channel the data item refers to.

AAC:

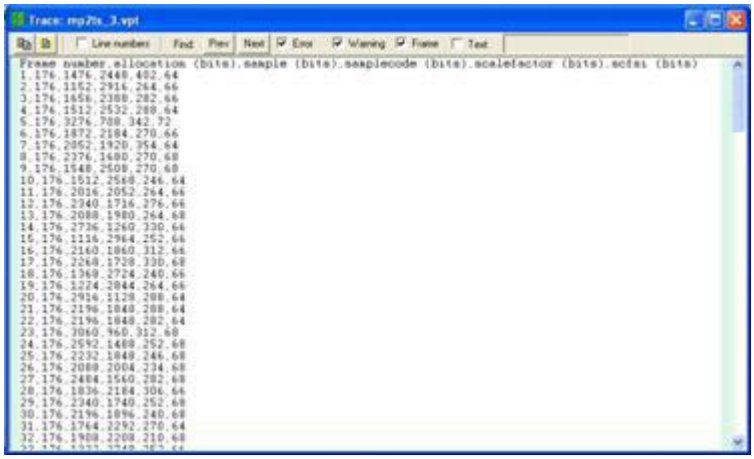
- Frame number
- Total SBR data size[*n*] (bits)
- Total config data size[*n*] (bits)
- Total data size[*n*] (bits)
- Max scalefactor blocks[*n*] (blocks)
- Global gain[*n*]
- LTP data present[*n*]
- Predictor data present[*n*]
- MS mask present[*n*]
- Pulse data present[*n*]
- TNS data present[*n*]
- Gain control data present[*n*]
- Window sequence[*n*]
- Window shape[*n*]

MPEG-2 Audio:

- Frame number
- Total audio data size[*n*] (bits)
- Ancillary data size[*n*] (bits)

The definition of each of these columns is given in the description of the Frame tooltip; (See page 5-146, *Analysis menu*.)

Audio data bit usage [MPEG-2].

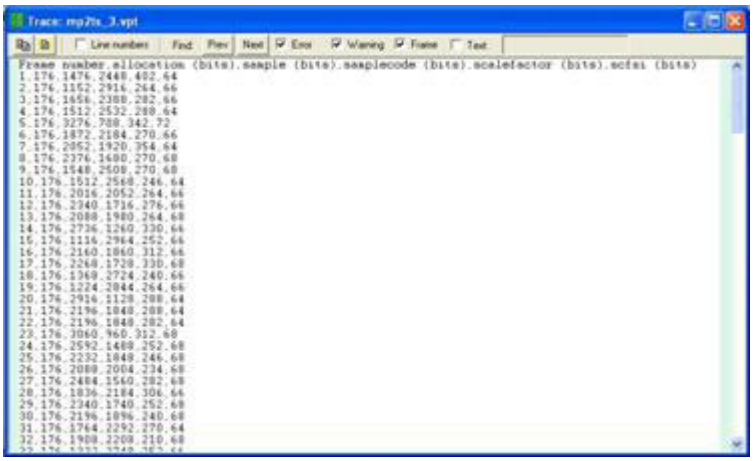


This Trace output provides a series of six data items, listed below, for each individual audio frame:

- Frame number
- allocation (bits)
- sample (bits)
- samplecode (bits)
- scalefactor (bits)
- scfsi (bits)

The definition of each of these columns is given in the description of the histograms. (See page 5-141, *Show histogram*.)

Frame data bit usage [MPEG-2].

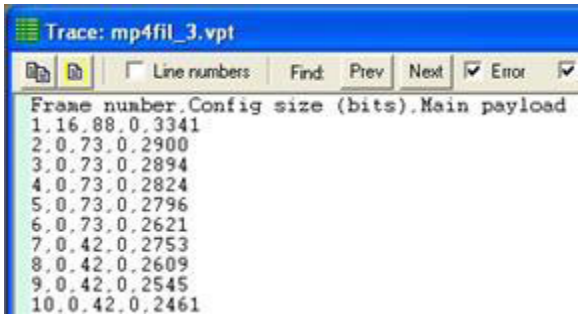


This Trace output provides a series of nine data items, listed below, for each individual audio frame:

- Frame number
- AncillaryData (bits)
- AudioData (bits)
- AudioSequence (bits)
- ErrorCheck (bits)
- Frame (bits)
- Header (bits)
- HuffmanCodeBits (bits)
- MainData (bits)

The definition of each of these columns is given in the description of the histograms. (See page 5-141, *Show histogram*.)

Main payload bit usage [MPEG-4].



This Trace output provides a series of 5 data items, listed below, for each individual audio frame:

- Frame number
- Config size (bits)
- Main payload size (bits)
- SBR size (bits)
- Subsidiary payload size (bits)

The definition of each of these columns is given in the description of the histograms. (See page 5-141, *Show histogram*.)

Subsidiary payload bit usage [MPEG-4].

```
1,56,0,0,12,0,11,0,0,0,9,0
2,56,0,0,0,0,11,0,0,0,6,0
3,56,0,0,0,0,11,0,0,0,6,0
4,56,0,0,0,0,11,0,0,0,6,0
5,56,0,0,0,0,11,0,0,0,6,0
6,56,0,0,0,0,11,0,0,0,6,0
7,21,0,0,0,0,15,0,0,0,6,0
8,21,0,0,0,0,15,0,0,0,6,0
9,21,0,0,0,0,15,0,0,0,6,0
10,21,0,0,0,0,15,0,0,0,6,0
11,56,0,0,0,0,11,0,0,0,6,0
```

The following data items form the series of twelve seen in the Trace excerpt preceding (frames 1-11 shown):

- Frame number
- channel_pair_element (bits)
- coupling_channel_element (bits)
- data_stream_element (bits)

- fill_element (bits)
- gain_control_data (bits)
- ics_info (bits)
- lfe_channel_element (bits)
- program_config_element (bits)
- pulse_data (bits)
- raw_data_block (bits)
- single_channel_element (bits) (bits)

The definition of each of these columns is given in the description of the histograms. (See page 5-141, *Show histogram*.)

SBR bit usage [MPEG-4]. Similar to the subsidiary payload bit usage Trace above, this Trace output provides a series of 16 data items, listed below, for each individual audio frame:

- Frame number
- sbr_channel_pair_base_element (bits)
- sbr_channel_pair_element (bits)
- sbr_channel_pair_enhance_element (bits)
- sbr_data (bits)
- sbr_dtdf (bits)
- sbr_envelope (bits)
- sbr_extension_data (bits)
- sbr_extension (bits)
- sbr_grid (bits)
- sbr_header (bits)
- sbr_invf (bits)
- sbr_noise (bits)
- sbr_single_channel_element (bits)
- sbr_sinusoidal_coding (bits)
- unused sbr data (no header found) (bits)

The definition of each of these columns is given in the description of the histograms. (See page 5-141, *Show histogram*.)

Frame bit usage [MPEG-4].

```
Standard:   AAC, LC
Frame size: 1024 samples per channel
Num channels: 2
Filesize:   194 kBytes

Frame number,dynamic_range_info (bits),extension_pay
1,0,352,22,0,0,432,324,2115,96
2,0,0,22,0,0,391,315,2172,0
3,0,0,22,0,0,363,351,2158,0
4,0,0,22,0,0,371,306,2125,0
5,0,0,22,0,0,375,360,2039,0
6,0,0,22,0,0,343,234,2022,0
7,0,0,22,0,0,113,133,2485,0
8,0,0,22,0,0,131,98,2358,0
9,0,0,22,0,0,125,84,2314,0
10,0,0,22,0,0,118,94,2227,0
```

This Trace output provides a series of 10 data items, listed below, for each individual audio frame:

- Frame number
- dynamic_range_info (bits)
- extension_payload (bits)
- individual_channel_stream (bits)
- ltp_data (bits)
- reordered_spectral_data (bits)
- scale_factor_data (bits)
- section_data (bits)
- spectral_data (bits)
- tns_data (bits)

The definition of each of these columns is given in the description of the histograms. (See page 5-141, *Show histogram*.)

Audio analysis options...

The Audio Analysis Options dialog box provides three tabs: Frame range, Trace and A/V delay. Frame range and Trace are described in *Audio Trace*. (See page 5-233.)

The A/V delay options are described in *Audio/Video Delay*. (See page 5-254.).

View audio

(See page 5-136, *Show audio waveform*.)

View audio trace

This option opens the Trace view on the current output Trace file, as given in the Analysis menu Trace tab.

NOTE. *If there is not a current Trace file (and Trace is not enabled) this menu option/toolbar icon is unavailable (grayed out).*

NOTE. *Multiple audio trace files can be opened and viewed simultaneously. They can be opened using the Open Other option in the File menu.*

NOTE. *If the stream has been rerun and a Trace file by the same name is already open, then clicking the View trace menu option or icon will refresh the currently open Trace file.*

Go to other views/right click menu - Interpret & Parse bitstream only

By right-clicking, a context-sensitive menu appears, which allows you to go to the corresponding area in another view:

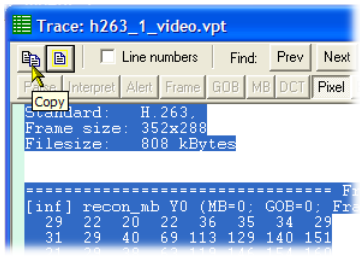
NOTE. *This right-click context-sensitive menu is only active in the Trace/Interpret and Trace/Parse Bitstream trace files.*

Copying text

The contents of the Trace window can be copied to the Windows clipboard.

This can be done in several ways:

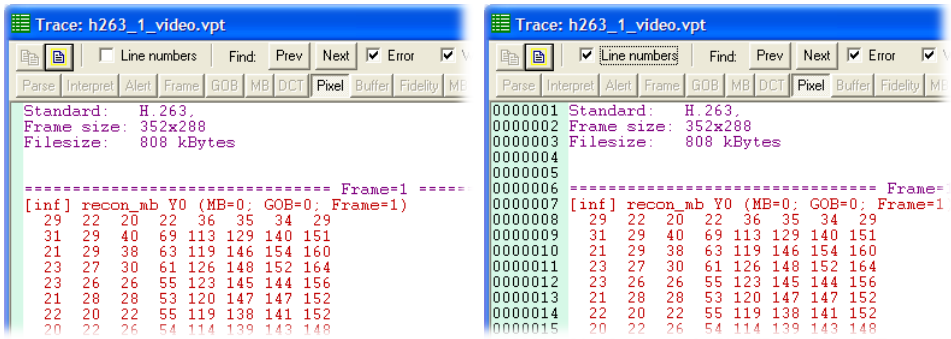
- By pressing <Ctrl+C>
- By right-clicking and selecting Copy
- By clicking the Copy icon (see below)



The icon to the immediate right of the Copy icon is the Select all icon. Rather than highlighting and dragging over what could be a lengthy Trace file, you can click on this icon to highlight all of the text in the Trace file for copying.

Line numbers

Line numbers can be turned on/off:



Find data

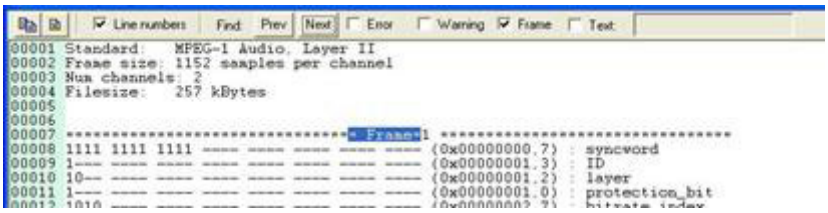
The Trace file can be searched for any data, using the Find: Prev (previous) and Next buttons. This finds the previous/next occurrence of any of the enabled strings.

NOTE. *The <Shift+F3> key can also be pressed, to find previous.*

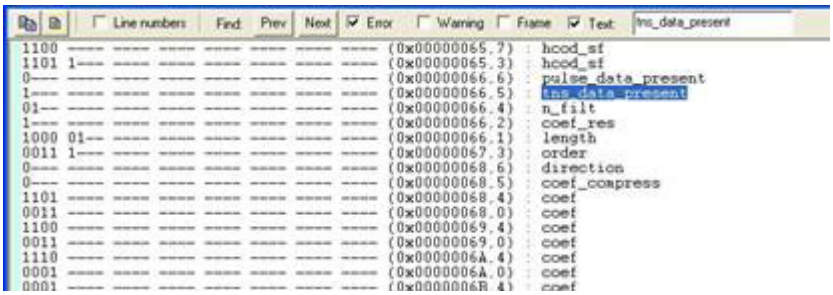
There are some standard strings that are useful to search for:

- Error
- Warning
- Frame

These are found by enabling the relevant check box (Frame example below):

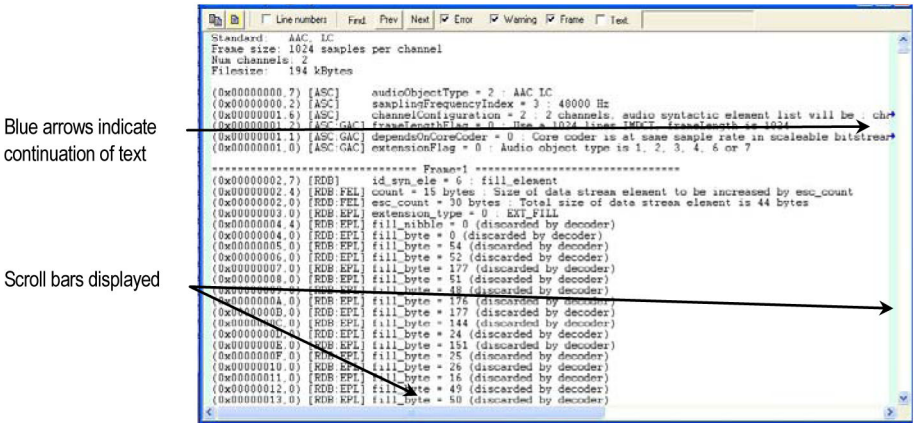


To search for any text, check the box next to Text and enter the text in the box (in this example, searching for 'tns_data_present'):



Lines too long for window width

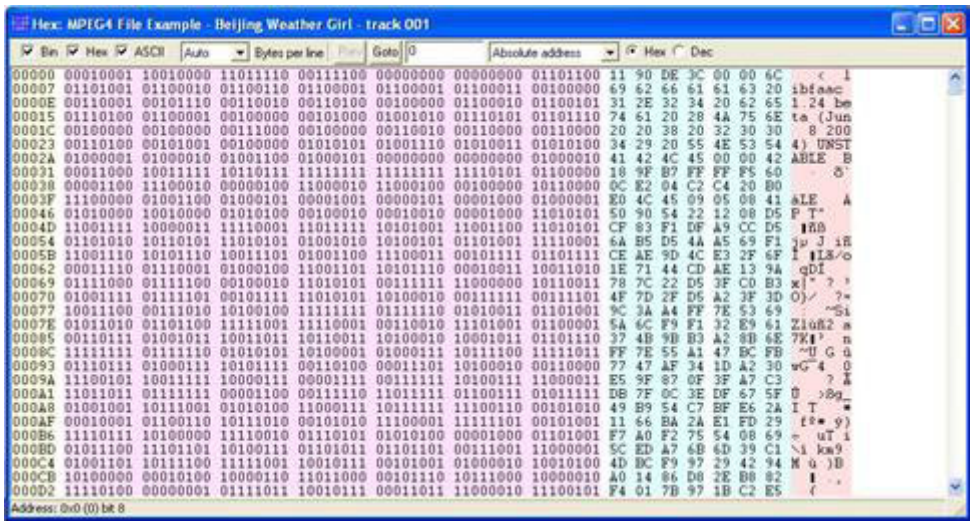
If a line is too long to be displayed within the width of the View trace window, a horizontal scroll bar is displayed and a blue arrow is appended to the right-hand side of the View trace window, to indicate the line extends to the right:



Trace file format

MTS4EA saves Trace files with a .vpt file extension. Currently, the .vpt file is a standard ASCII file which can be viewed in any text file viewer.

View audio stream hex



This opens the current audio stream in a hex viewer, which shows the data in binary, hexadecimal, and ASCII data (in any combination of the three).

NOTE. *Multiple View hex windows can be opened at the same time.*

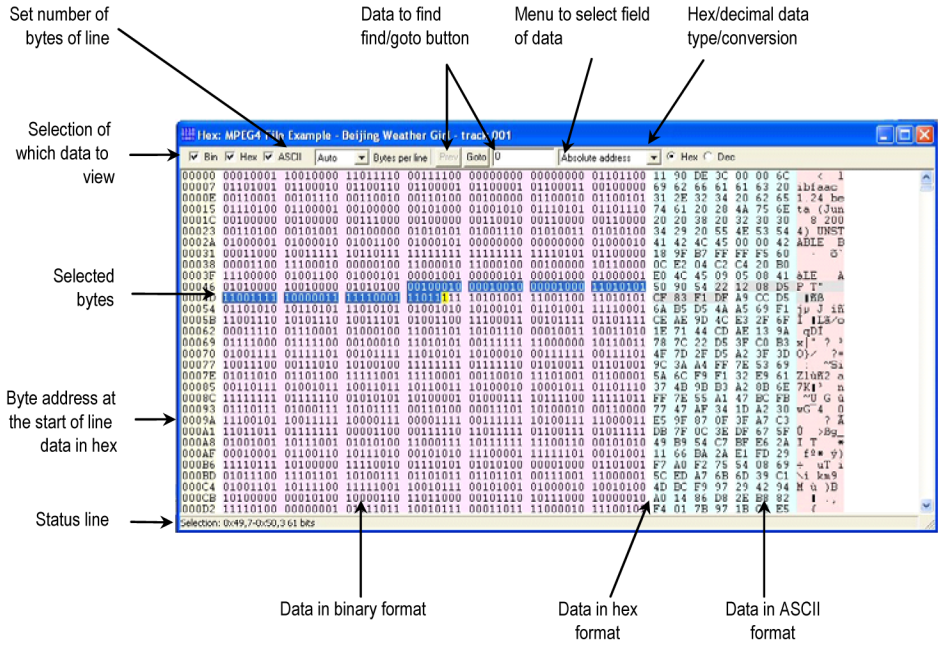
Using HexView, you can scroll through the file, and search for specific:

- Absolute address (from the start of the file)
- Relative address (from the currently-selected location)
- Binary data
- Hex data
- ASCII data

NOTE. *In the field where the data to find is entered, a wildcard character can be entered - this is '.' (a period).*

NOTE. *By entering a hex value and then clicking on Dec, you can convert the number to decimal (and vice versa).*

Sections of HexView window.



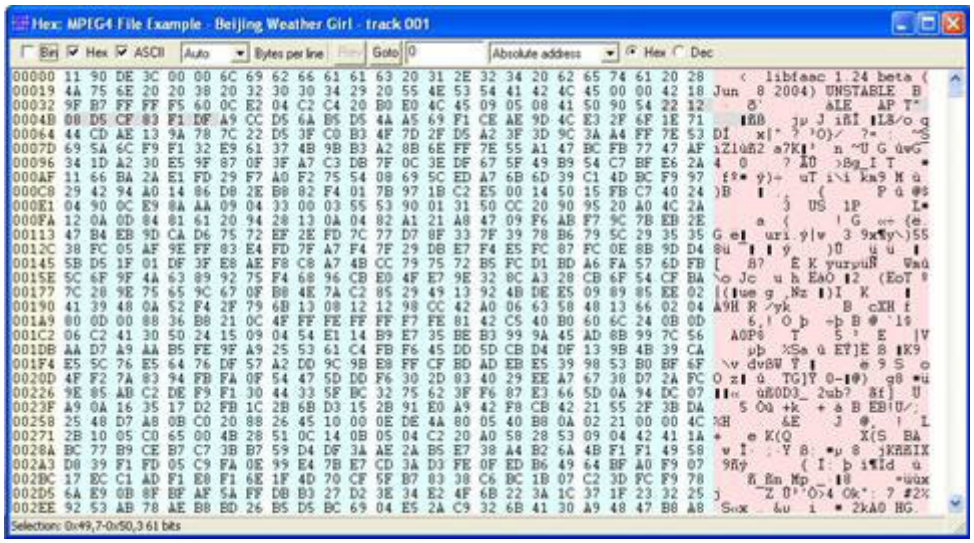
Setting information displayed; window width

The width and size of the HexView window can be set as with any other window in Windows. The same data is shown in each of the three sections:

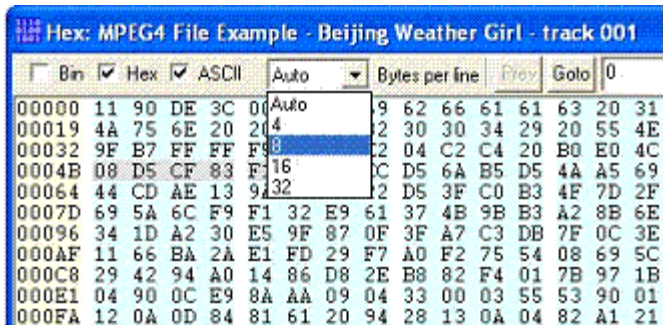
- Binary
- Hexadecimal
- ASCII

Each of these areas can be individually displayed/hidden by clicking the appropriate tick box in the top left-hand corner of the window.

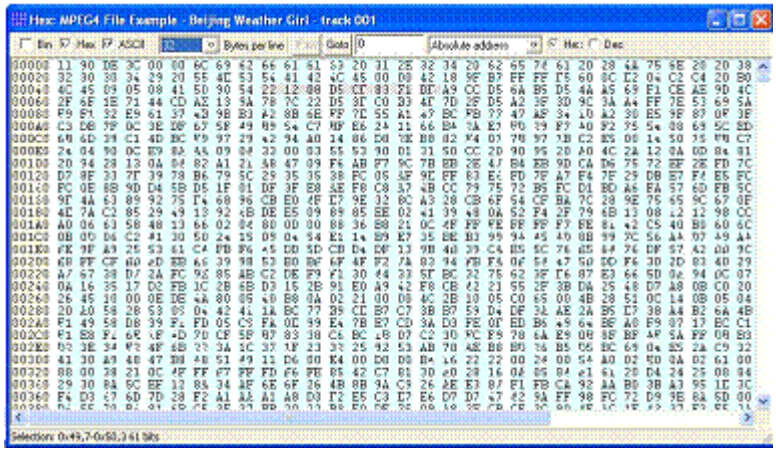
In this example the binary display has been turned off, leaving only hex and ASCII:



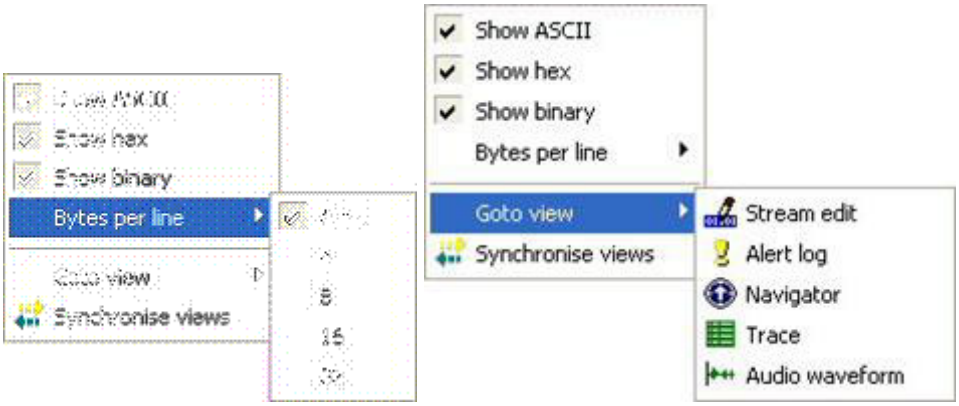
Setting bytes per line. This dropdown menu is used to set the number of bytes per line:



If the current width of the window is insufficient to show all the columns, then a scroll bar automatically appears at the bottom, allowing you to scroll left/right to see all the columns, as in the following example:

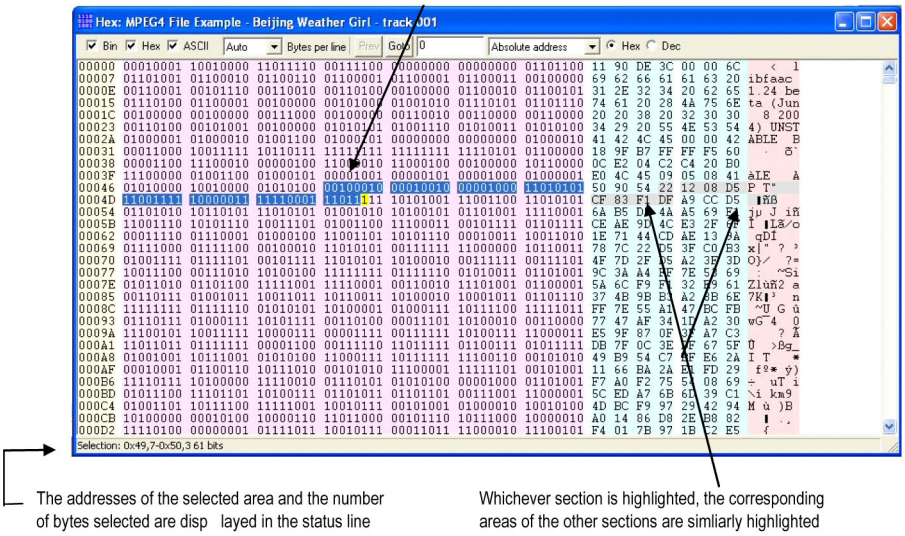


Right-click popup menu/Goto view. Right-clicking in the HexView window generates a menu, from which it is possible both to set the number of bytes displayed per line and to navigate to the same location within the stream in an alternative view:



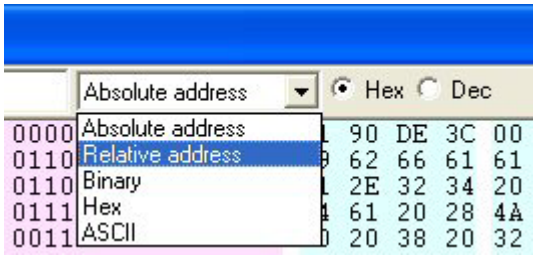
Synchronize views. When the Synchronize views icon is pushed in, all open windows automatically follow the selection that you make.

Highlighting a section. A section of data can be highlighted in any of the (visible) binary, hex or ASCII sections by clicking and dragging the mouse over the area:



Find absolute address. A specific address can be found in the audio stream, either:

- An absolute address, from the start of the audio file
- A relative address, from the first byte of the currently selected area



The address to find is entered in the box next to the Goto button. This address can be in hex or decimal.

If an invalid character is entered for an address (entering anything other than '0-9' or 'a-f' for a hexadecimal address), the Goto button is greyed out.

Note that the Goto button changes to Next for all data to find except for the Absolute address. (When Absolute address is selected, the Prev button is greyed out.)

The Hex/Dec option button selects the format of the data being searched - hexademical or decimal.

Find relative address. An address relative to the currently highlighted address can be found. If there is no address highlighted, the address found is the offset from 0.

The address to find is entered in the box next to the Next button. This address can be in hex or decimal. If an invalid character is entered for an address (for example, entering anything other than '0-9' or 'a-f' for a hexadecimal address), the Find button is greyed out.

NOTE. *The <F3> key can also be pressed to find next. The <Shift+F3> key can also be pressed to find previous.*

The Hex/Dec radio button selects the format of the data being searched.

Find Binary/Hex/ASCII. These menu options find data in the bitstream. Up to 64 characters can be entered.

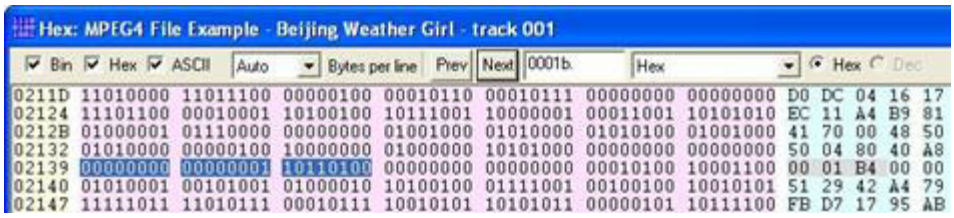
NOTE. *The Binary search searches for the bit pattern irrespective of byte location; the Hex and ASCII searches are byte aligned.*

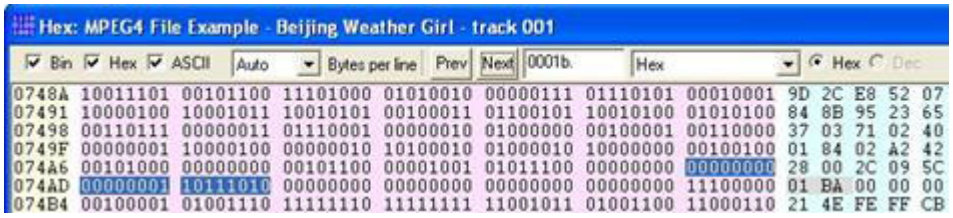
Wildcard searching using '.'. A wildcard can be entered in the data to be found, this is the period character - '.'

The wildcard matches a single digit in the base selected; the wildcard matches:

- One bit in the binary find
- A hex digit in the hex find
- An ASCII character in the ASCII find

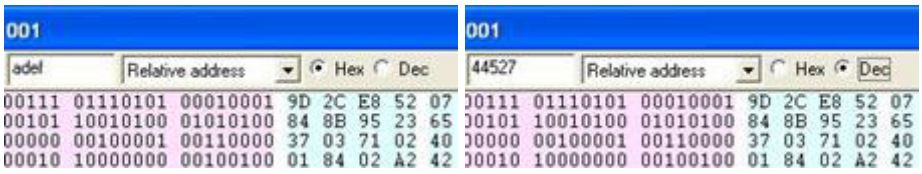
For example, entering '0001b.' as a hex string to find (any character after the '0001b' pattern will find the following patterns in the selected example stream:





The wildcard can be inserted in any position. For example, '03..456.8' is a valid search string, which will find any nine consecutive digits where the first two are '03', digits 5-7 are '456' and the last digit is '8'.

Conversion of hex<->decimal. If a hex value is entered into the find box, and the Dec option button is clicked, the value is converted to hex (and vice versa):



View audio stream structure

This allows the structure of the following file types to be viewed and expanded/contracted (by clicking on the '-' and '+' symbols at the left end of each line). Each branch of the structure can be clicked on to expand it and see the nodes below it - see the examples following:

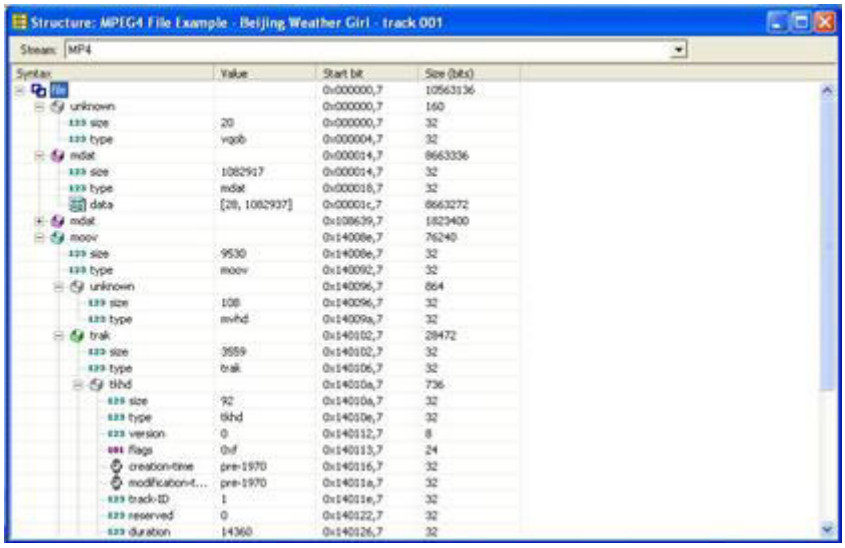


Figure 5-25: MP4 containing AAC LC

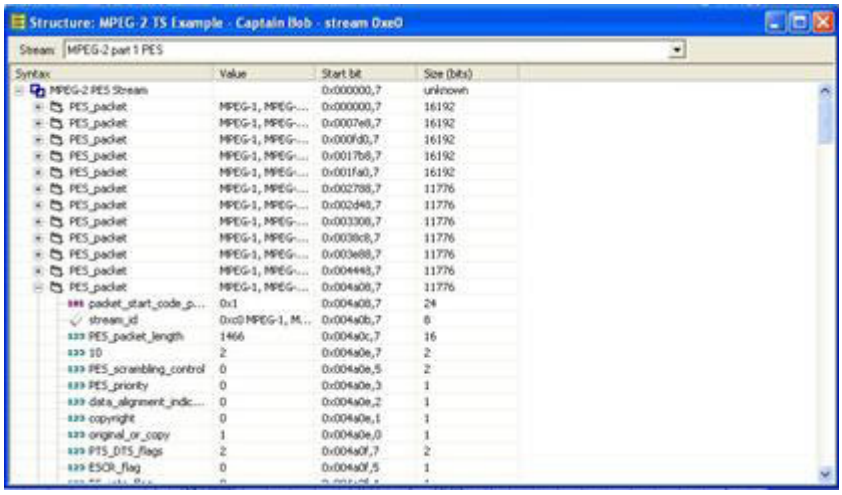


Figure 5-26: Container files, for example, MPEG-2 PES

The view is divided into four columns:

Syntax. This column shows the stream syntax parse tree, including the names of each syntax element or syntactic structure. Nested syntax structures are shown by nesting nodes of the tree. An icon shows the type of data at each tree node.

Value. This column shows the interpreted value of syntax elements in the tree. The content of this field will vary according to the type of data to be shown.

Start bit. Shows the start bit address of the syntax node. The address is given as a hexadecimal byte position followed by a bit position number.

Size. Shows the size in bits of the syntax node. For nodes with children, this shows the total bits of the node and all its children.

(See page 5-210, *View video stream structure Ctrl+R*.)

Audio/Video delay

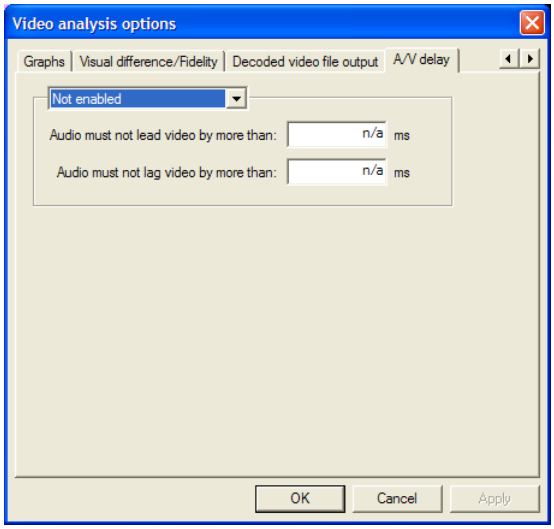
This feature provides a complete solution for measuring audio/video synchronization delay in an encoding system. MTS4EA provides a selection of predefined, clock-tick test streams in various formats. You can select an audio and a video stream and encode them using your own equipment. You can then analyze the resulting encoded stream using the MTS4EA and measure any time delay between the audio and video that may have been introduced by the encoding.

The available test streams include uncompressed streams for testing elementary stream encoders and compressed streams for testing transport multiplexers.

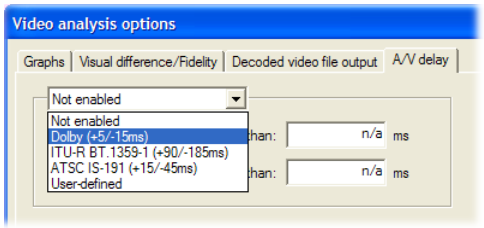
(See page 5-40, *Export A/V delay streams*….)

A/V Delay test enable

This menu option opens the Video Analysis Options dialog box. This option, when used with the supplied A/V Delay streams, allows you to measure audio/video delay, which may have been introduced when encoding elementary streams. When enabled, the feature allows you to select preset or user selected delays.

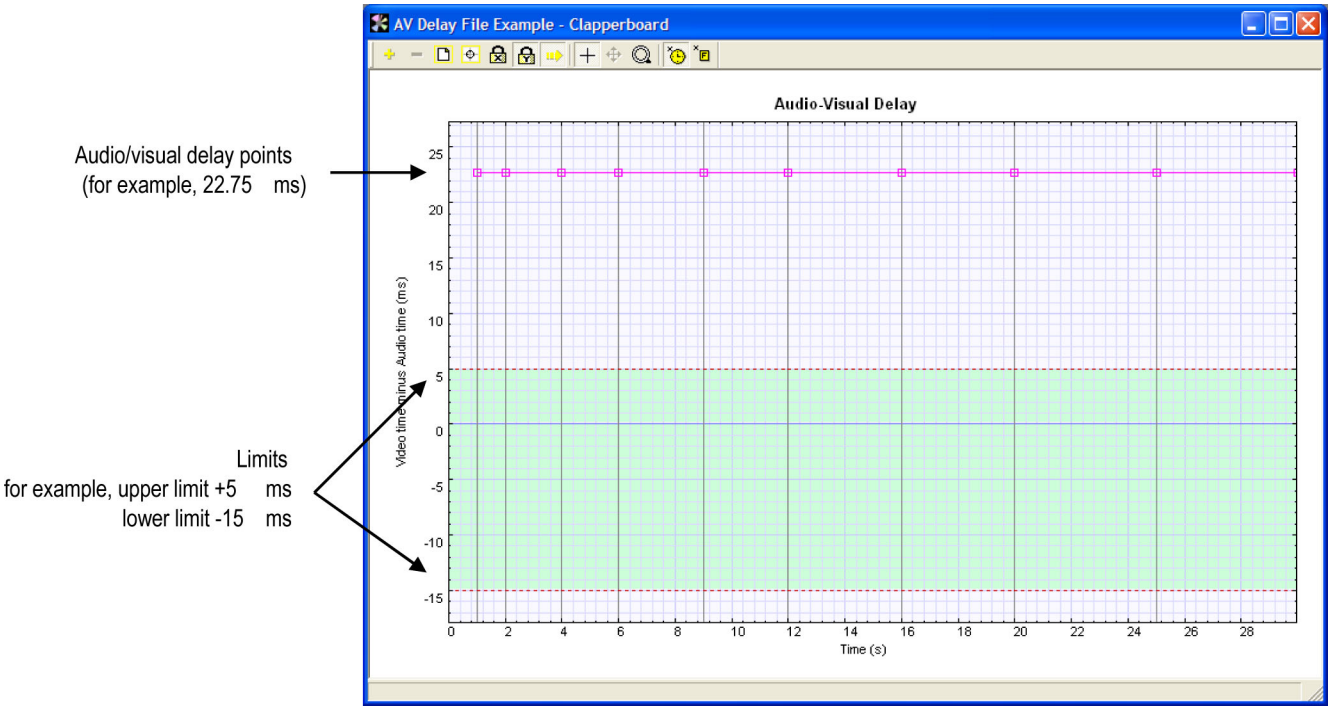


Select the required option from the drop-down list; the preset lead and lag values are shown in the list. If you want to set your own delay values, select User Defined and enter values in the lead and lag fields.



View A/V Delay analysis

This menu option displays the A/V Delay graph. You will need to decode the stream to view the results.



The audio/visual limits you set (either preset or user-defined) are displayed in green. The actual delay is shown in red. In the preceding example, the limits are set at +5 ms and -15 ms. The example stream that has been decoded shows that the audio leads the video by 22.75 ms.

Enforce extra restrictions

This Analysis menu option enables a range of extra checks specific to a selected standard to be performed on the stream as it is decoded.

When a specific option is enabled from the Enforce Extra Restrictions menu, that option will apply only to streams of the appropriate standard and will have no effect on any other stream.

When an option is selected, the status bar will display the message “Extra restrictions in force”.

To turn off the extra restrictions, select the item in the submenu again; the status bar will change so that “Extra restrictions” is no longer displayed. Closing the stream or opening a new stream does not reset the option; it will remain in force until it is turned off using the menu or until MTS4EA is restarted.

When a stream violates one of the extra checks, an alert is raised in the usual way. The alert details are available in the same way as all other alerts, both in the Alert view and in the Alert trace (if enabled).

Extra restrictions can also be enabled using the command line –q option.

ARIB TR-B14 option

This option consists of a set of additional restrictions for H.264/AVC elementary streams as specified in Tables 5-2 to 5-14 of the ARIB TR-B14 standard (see Reference [24] in *Standards References*). Other System Layer and Elementary Stream Layer checks in ARIB TR-B14 are not carried out.

When this option is enabled it will be active only for H.264/AVC streams; there will be no effect for other streams.

Extra restrictions can also be enabled from the command line. (See page 5-301, *Command line/batch mode*.)

H.264 transform option

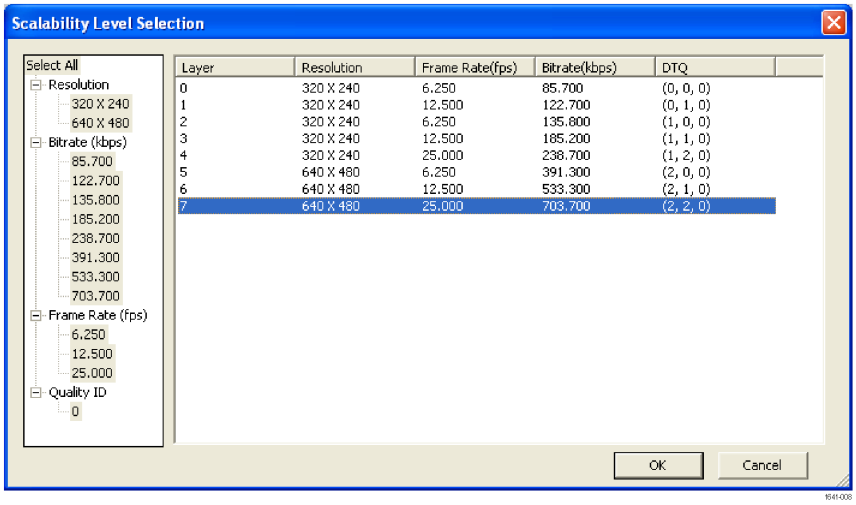
This option consists of a set of additional restrictions for H.264/AVC elementary streams as specified in Scaling and Transformation Processes in sections 8.5.9 - 8.5.12 of the H.264 standard (see Reference [12] in *Standards references*).

When this option is enabled it will be active only for H.264/AVC streams; there will be no effect for other streams.

Extra restrictions can also be enabled from the command line. (See page 5-301, *Command line/batch mode*.)

Scalability level selection

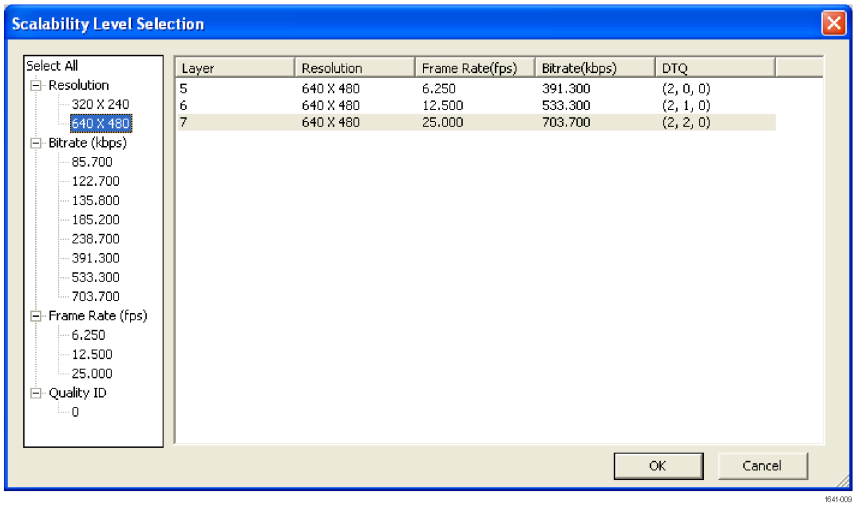
This Analysis menu option enables the display and selection of a specific layer to be analyzed. This option is enabled only for SVC streams, and only if the SVC stream has the scalability_info SEI message encoded. Otherwise, the highest enhancement layer is analyzed by default.



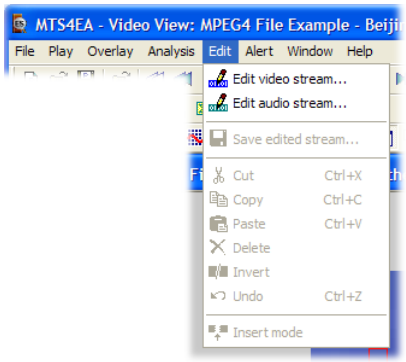
You can filter layers for selection based on one of the following parameters:

- Resolution
- Bitrate (kbps)
- Frame Rate (fps)
- Quality ID

An example of filtering based on resolution is shown below:



Edit menu



The Bitstream editor facility allows you to:

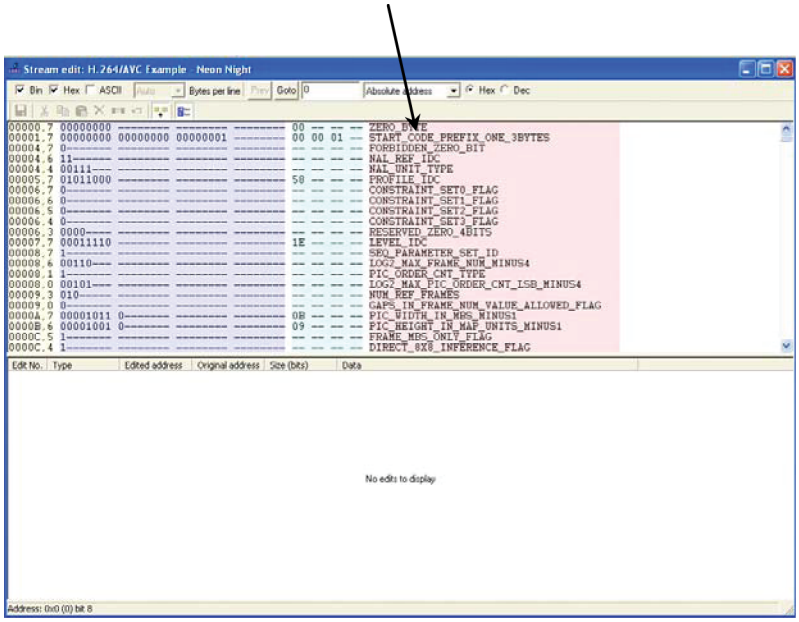
- Insert, delete and modify bits and bytes at any point in the bitstream
- Copy sections of the bitstream
- Rerun the analysis on the edited bitstream
- See a change log of all edits that were made

Edit audio/video stream

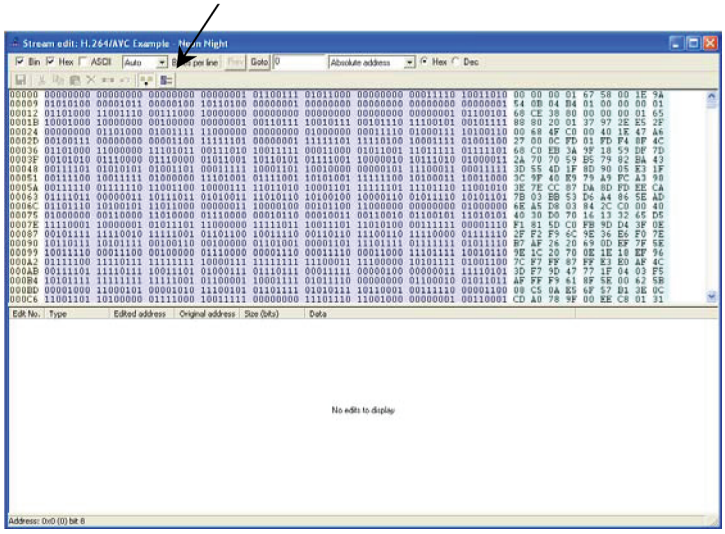
When the Bitstream editor window is initially accessed, the following message will pop up advising you that the syntax will not be visible in this view until it is played (decoded):

> MTS4EA Elementary Stream Analyzer: Warning
> Syntax element labels will be available in the stream editor after the stream is decoded (video/audio is played).

Once decoded, the syntax area will be populated.



The pink area containing the syntax is displayed by default. If so desired, these labels can be removed from the display by clicking on the “Show labels” button:



The bottom half of the window - the change log - is empty until edits have been made to the stream.

Bitstream editor icons

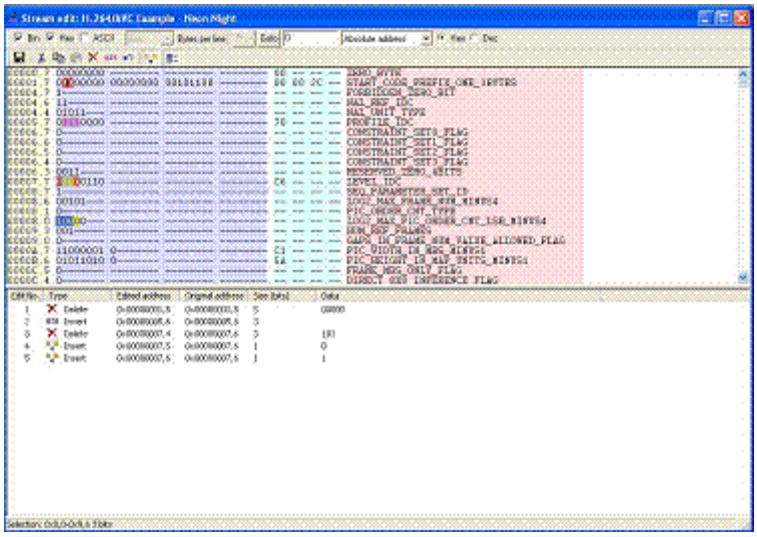
The functions of the remaining icons in the Bitstream editor view are outlined in the following table:

Table 5-3: Bitstream editor icons

Icon	Function
	The edited stream can be given a new name and saved
	Cut (Ctrl+X) becomes available only when part of the stream is highlighted
	Copy (Ctrl+C) becomes available only when part of the stream is highlighted
	Paste (Ctrl+V) becomes available only when part of the stream is highlighted and copied to the clipboard
	Delete bits; becomes available only when part of the stream is highlighted
	Inverts bits highlighted 0 becomes 1, 1 becomes 0; becomes available only when part of the stream is highlighted
	Undo last edit (Ctrl+Z)
	Insert bits mode (this is enabled by default)
	Show labels (this is enabled by default

Change log

As changes are made to the bitstream, the displayed text changes in color and the changes made appear as list items in the bottom half of the window. As in most of the views in MTS4EA, it is possible to highlight and double-click on any one of these items to be taken to the corresponding area in the relevant view. In this case, the bitstream view that occupies the top half of the window.



As can be seen in this example, each numbered change lists both the Edited and Original addresses. This is because edits will have a cumulative effect on the addresses of bits later in the stream. So, if a bit or bits are deleted or inserted, the addresses of the following bits are changed. Therefore, to avoid confusion, both the original address and the address following earlier deletions or insertions are given.

Undo last edit Ctrl+Z

The last edit made to the bitstream can be undone by clicking on this icon or by using the shortcut Ctrl+Z.

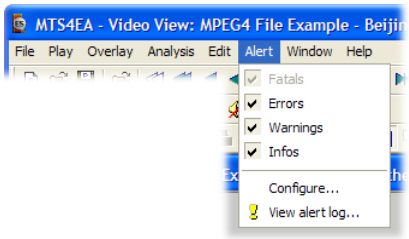
Saving edited stream…

Once all desired changes have been made, the stream can then be immediately played and reanalyzed. Alternatively, the stream can be saved under a new name and reopened later for analysis.

Edit commands

The Edit menu options (Cut, Copy, Paste, Delete, Invert, Undo, Insert Mode) have the same functions as those described in the Bitstream editor icons table. (See Table 5-3.)

Alert menu



This menu controls the setting of the overall level of the popup alerts and gives access to configure individual alerts as enabled/disabled.

These popup alerts occur when the video is being decoded. Consequently:

- The popup alert appears when the frame concerned is decoded, but for bitstreams containing B-frames/B-VOPs the display order of frames/VOPs may be different
- When seeking backwards/forwards through a stream, the popup alerts will not appear if that part of the bitstream has already been decoded and is in the step-back buffer cache. (See page 5-266, *Popup alerts in the step-back buffer*.)

NOTE. *For some Alerts, extra information is provided about the section of the stream that has generated the error, and the standard if it is standard-specific.*

(See page 6-1, General codes used in Trace files and Alerts.)

Alert levels

There are four different levels of alerts: Fatal, Error, Warning and Info (in order of decreasing severity). (See page 5-270, *Description of alert levels*.)

When a less severe alert is set, the more severe alerts are set automatically. (For example, setting Warnings on automatically sets Errors and Fatal on.)

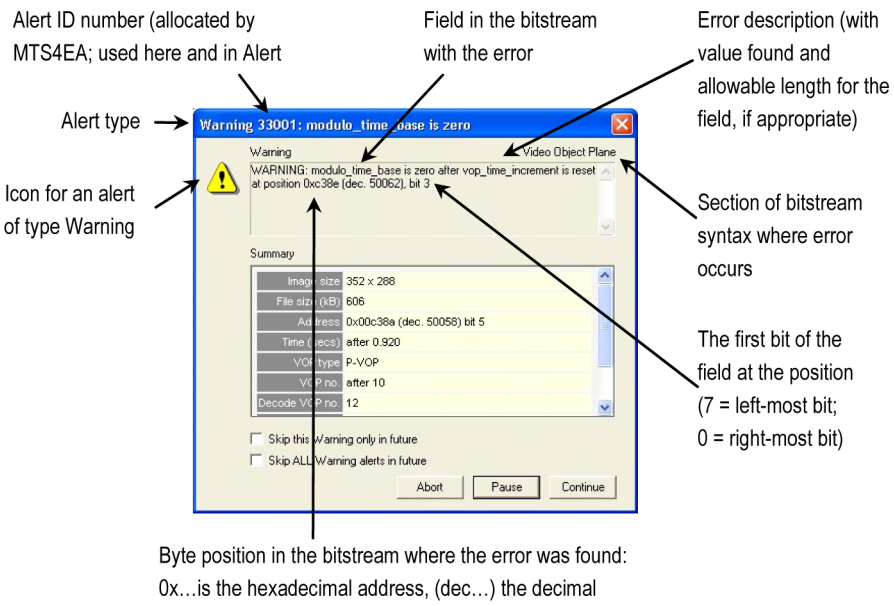
NOTE. *The Fatal alert is always set (and cannot be turned off). Each time a video stream is opened in the MTS4EA, the alert level is set to Info (the strictest) and all Alerts are reenabled.*

When an alert is triggered, a message box is displayed. At the same time, decoding is suspended.

The message box will provide you with the reason that decoding was interrupted - see the following section.

Explanation of Fatal/Error/Warning/Info display

Information at top of alert popup



For bitstreams that contain B-VOPs/B-frames, it is vital to understand that the popup alert relates to the decode frame number, not the displayed frame number. (See page 5-265, *Decode/display frame/VOP numbers in bitstreams with B-frames/B-VOPs*.)

For all Fatal, Error, Warning and Info displays:

- The bitstream position of the error is the number of bytes from the beginning of the video bitstream, where the first byte is byte zero.
- The position is displayed in hexadecimal (0x) and decimal (dec.).
- The bit start is the first bit of this bitstream field, where bit 7 is the most significant bit in a byte and bit 0 the least. This means that bit 7 is the first bit in each byte of the bitstream.
-

In each of the compression standards, there are many bitstream fields that give values that are least significant bit first (lsbf) and many others that are most significant bit first (msbf). There are also many others that are simply bit-patterns: they do not have a direct numeric value.

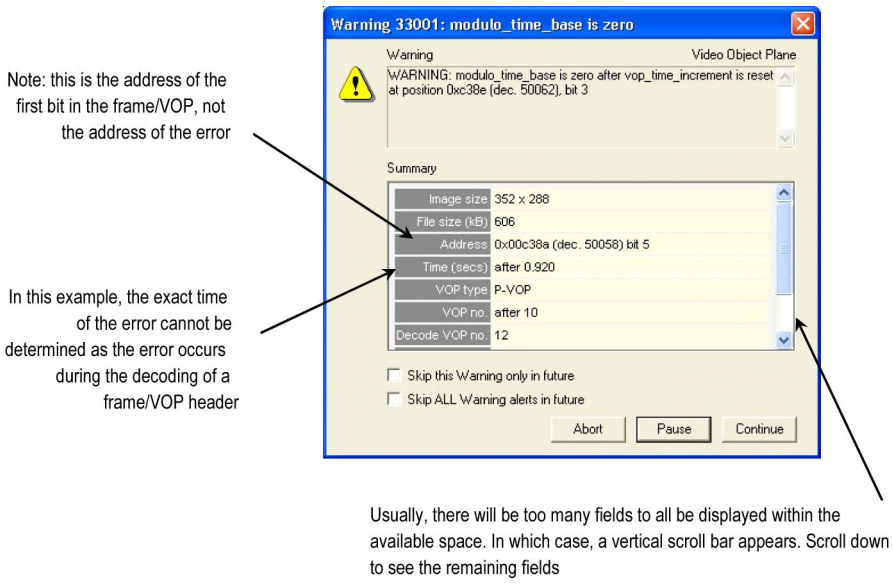
For msbf bitstream fields, numbering bit 7 as the first bit in each byte is correct. For lsbf bitstream fields, the bit order has to be reversed when calculating the given value . See *Explanation of bit/bitstart* in the *Bitstream Syntax Debugging* section for more information.

(See page 6-8, *Explanation of bit/bit start*.)

Information in Summary box

The information provided in the Summary box is the same as that provided in the Summary tooltip;

(See page 5-66, *Video-frame summary tooltip Ctrl+U*.)



Decode/display frame/VOP numbers in bitstreams with B-frames/B-VOPs. The popup alert relates to the Warnings and Errors that the MTS4EA finds during decoding.

This means that for bitstreams that contain B-frames/B-VOPs (where the decode order is different from the display order) it can appear that the popup alerts are out of sequence. In this case, it is very important to look carefully at the data in the Summary box, which shows the Decode and Display frame numbers.

For example, if this is the order of data in the bitstream (the decode order):

Decode frame number	1	2	3	4	5
Frame type	I	P	B	B	P
Error occurs in frame	a	b	c	d	e

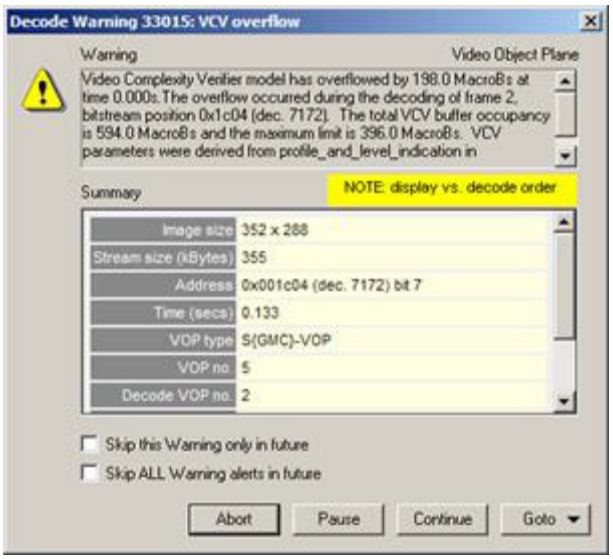
Then this would be the order of data in the display:

Display frame number	-	1	2	3	4	5
Frame type	-	I	B	B	P	P
Error displayed in frame	-	a, b	c	d	-	e

NOTE. *Frame display delayed by one frame.*

In these diagrams, it can be seen that for the first P-frame, the error associated with decoding it is displayed before it appears on-screen - at the same time as the first frame is displayed.

If this has happened, text in a yellow warning rectangle will appear in the popup alert:



The difference in decode versus display order:

- Can occur in any bitstream which has B-frames/B-VOPs
- And can happen in any frame/VOP in the bitstream concerned, where the frame/VOP is a B-frame/B-VOP

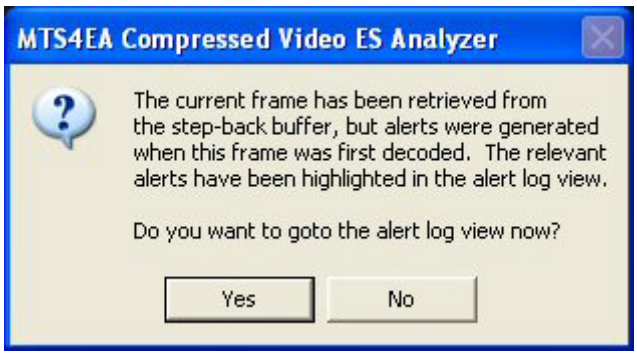
NOTE. *Note that the yellow warning can also appear in other circumstances, where the alert has occurred some time after a displayed frame.*

Popup alerts in the step-back buffer. When a stream is decoded, popup alerts may occur. The results of the decoding are stored in a cache, the Step-back buffer. (See page 5-54, *General*.)

It is possible to step forward/backward over the frames where these alerts occur. To avoid a continual repetition of the popups (and potentially many such alerts appearing), the popup alerts are not displayed when seeking forward/backward in these situations.

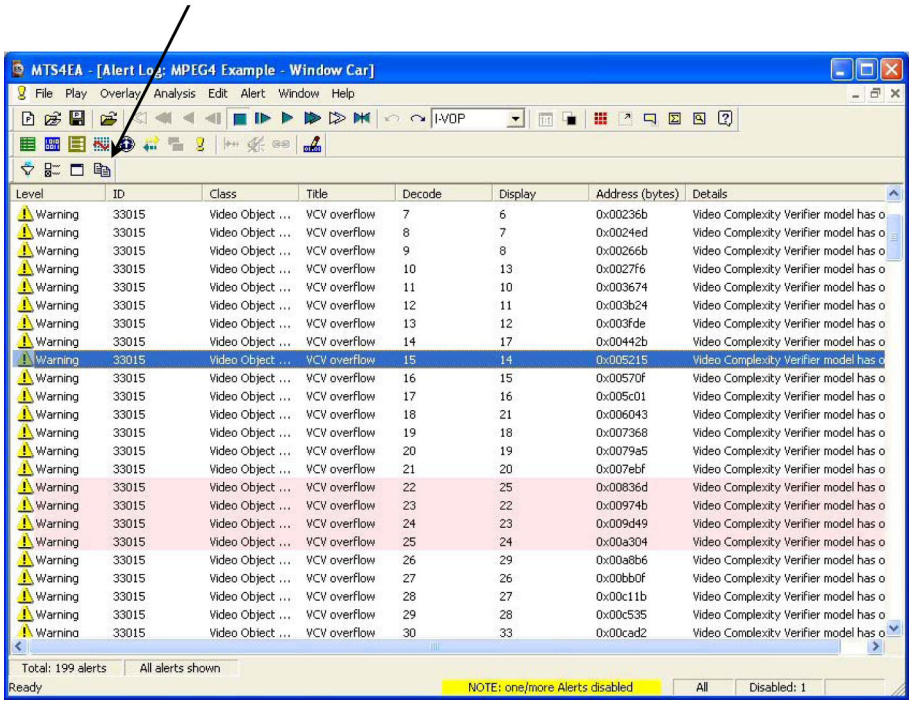
If one or more popup alerts would have occurred, then this is indicated in one of two ways:

- There may be a popup message, at most once per seek forward or backward:



- If the Alert log window is open, the alerts that would have popped-up are highlighted in the Alert log window. (See page 5-274, *View Alert log*.)

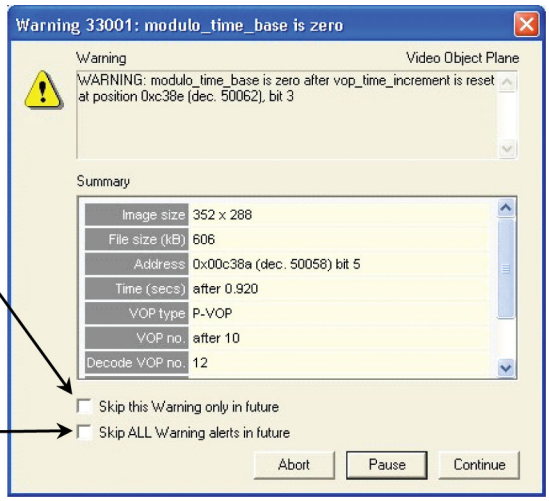
Copy alert details to the Windows clipboard. The details of an alert can be copied using Ctrl+C or the copy icon.



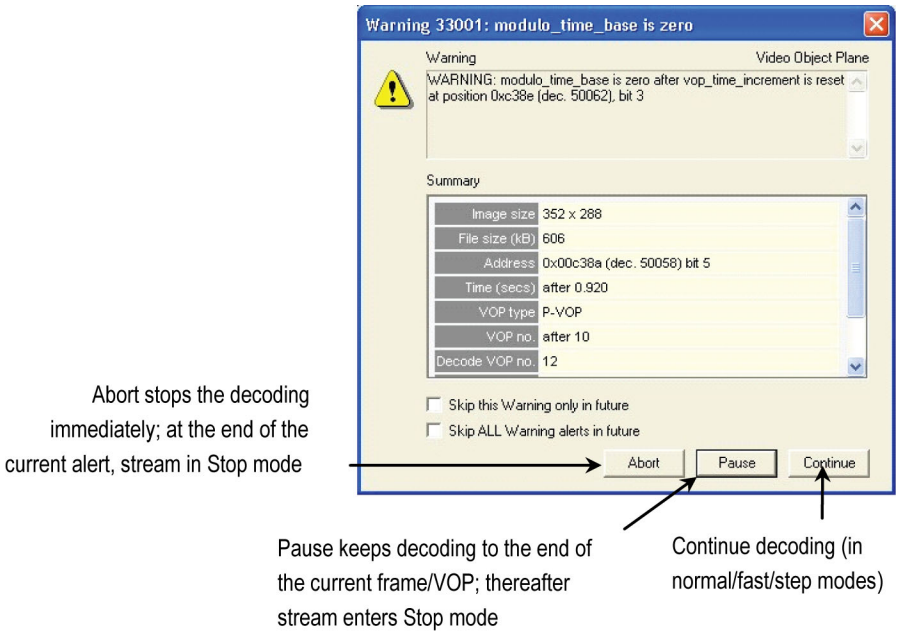
Check box options.

Check this box to prevent seeing further alert(s) for this specific warning. Other Warnings, Errors and Fatals will still trigger an alert

Check this box to prevent seeing further alert(s) for any alert level

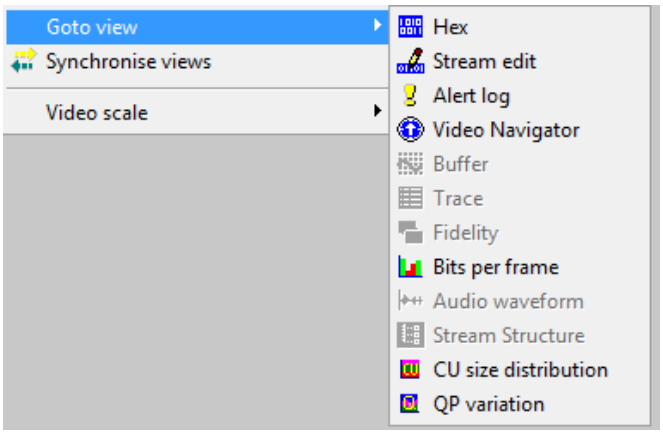


Abort, Pause, Continue buttons.



NOTE. *Abort can also be requested by pressing the Esc key.*

Goto button. This takes the focus to the position in the selected view that corresponds most closely to the location of the error. (See page 5-8, *Synchronized views/navigating the views*.)



In these views, selecting Goto takes the focus to the following locations:

View	Location taken to	
Video	If the error can be located to within a particular MacroBlock…	…to the MacroBlock containing the error (highlighted with a yellow/black dotted square)
	If not (the error is in a frame header or file header)…	…to the top of the frame that immediately follows the error, indicated by a yellow/black dotted rectangle around the first row of MacroBlocks
Hex	The byte where the error is	
Stream edit	The byte where the error is	
Alert log	The log of alert popups that have occurred (not those filtered/not shown)	
Navigator	The frame containing the error or the frame that immediately follows the error (if the error is in a header)	
Buffer	The frame containing the error or the frame that immediately follows the error (if the error is in a header)	
Trace	The syntax element in the Interpret or Parse bitstream file corresponding to the location where the error has been found	
Fidelity	Graph of fidelity analysis (such as PSNR)	
Audio waveform	Graph of audio stream: Y-axis is normalized amplitude; X-axis is time (seconds)	
Bits per frame	Bits per frame graph – the frame containing the error	
Stream Structure	Stream structure – highlight node containing error	
CU size distribution	CU size Distribution graph – the frame containing the error	
QP variation	QP variation graph – the frame containing the error	

Description of alert levels

Fatal

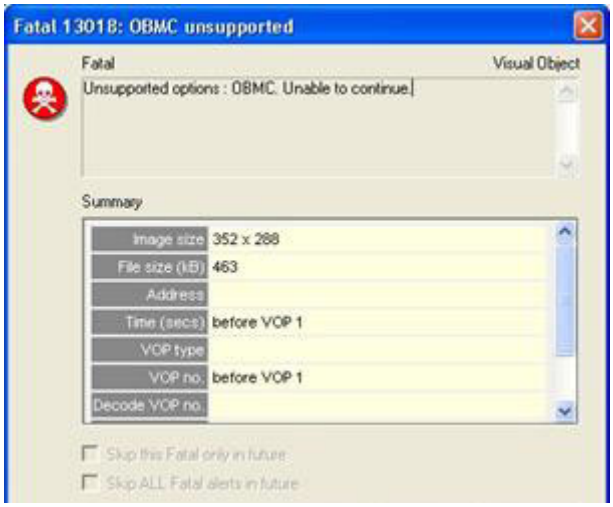
A Fatal error occurs when there is something encountered in the bitstream that cannot be decoded by the MTS4EA.

This selection cannot be turned off, and the decoder cannot continue after a Fatal error.

Fatal errors in the MTS4EA are commonly caused by:

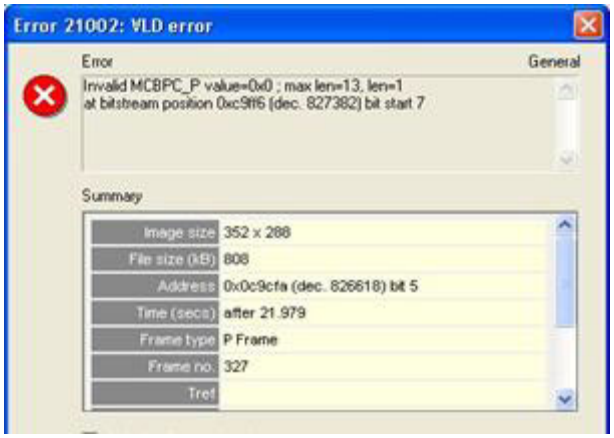
- Severe errors in the syntax of the bitstream, which make it completely unintelligible to MTS4EA
- Options being selected in the video bitstream that are not yet supported by MTS4EA

For example, the Fatal error below was caused by the OBMC flag in an MPEG-4 bitstream being set (on), that the video bitstream in question is using overlapped block motion compensation. (OBMC = Overlapped Block Motion Compensation: although it is in the current MPEG-4 standard, none of the MPEG-4 profiles or levels defined currently support OBMC.)



Error When this item is selected, the decoder will stop at any point where there is an Error in the bitstream.

These occur commonly where out-of-range values have been used in the bitstream, or invalid codes (the example below in an H.263 stream):



Warning When selected, this item will provide several warnings that indicate behavior that is not correct, but does not necessarily damage the decoding of the bitstream. Items such as too many leading zeros before picture start codes, temporal references not set, and MacroBlocks not updated in Intra mode will cause such warnings (the example below in an H.263 stream):

Info When checked, this item will provide information popups if relevant:

- It may be that the behavior is within the acceptable bounds of the standard concerned (and the relevant Profile/Level), but perhaps is used in a non-recommended manner or at the limits
- It is felt that additional information would be useful

Configure alerts

This determines which specific alerts are enabled/disabled.

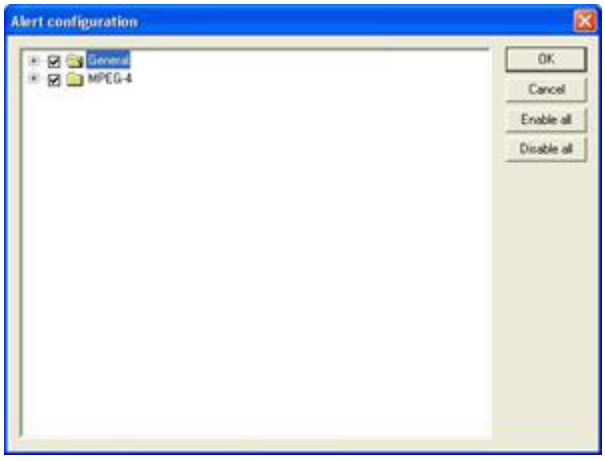
NOTE. *The overall alert level is set from the Alerts menu options Errors, Warnings and Infos, and this the enabling of specific alerts but not the disabling of specific alerts.*

This means, for example:

- That a specific Warning alert can be enabled/disabled, but trigger a Warning alert if the overall alert level is set to Errors or Fatal only. This is indicated in the status bar. (See page 5-280, *Status bar indication of alert status*.)
- But if the overall alert level is Info and an individual Error alert is disabled, then an alert will not be triggered for this individual error
- Examples of different combinations of overall and individual alerts and the consequential status bar displays are given under (See page 5-280, *Status bar indication of alert status*.)

NOTE. *Depending upon the standard, all the alerts that MTS4EA tests for are not necessarily currently available on-screen to be individually enabled/ disabled. Access to more of these will be provided in future versions of MTS4EA.*

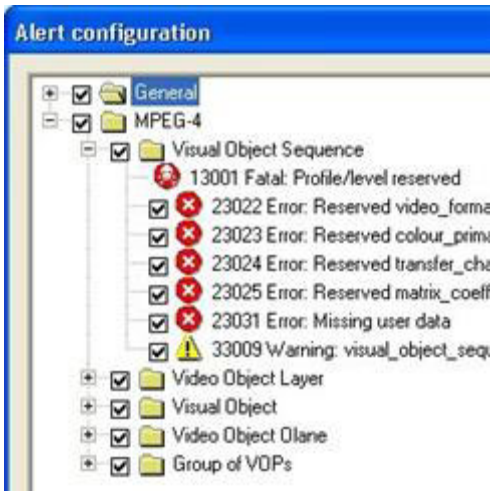
Enable/disable specific alerts



This displays two folders at the top level:

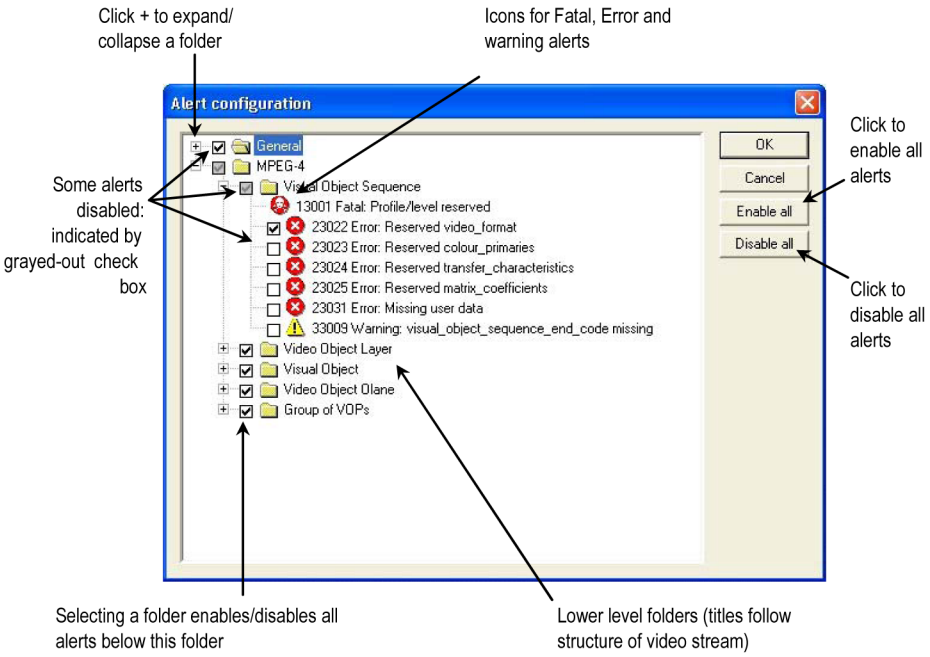
- General alerts, which can apply regardless of the standard concerned
- A standard folder (MPEG-4, H.263) that contains alerts specific to the standard concerned

The + next to each folder can be clicked to reveal lower levels:



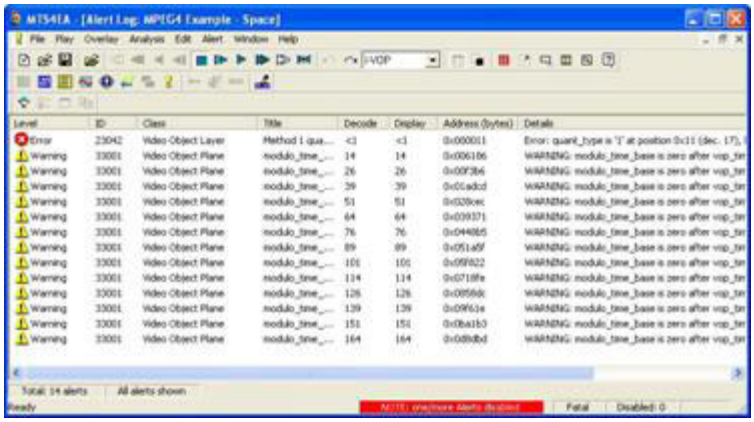
NOTE. *This is used to configure the individual alerts. Even if an alert is enabled here, if the overall alert level is set higher, the alert will not trigger a popup.*

Elements of the Alert configuration window



View Alert log

The Alert log keeps a list of the popup alerts that have occurred during the decoding of the bitstream:



The Alert log window icons and column titles are as follows:

Alert filter icon. (See page 5-276, *Showing/hiding alerts (alert filter)*.)

Show/hide video alerts.

Show/hide audio alerts.

Configure alerts icon. Clicking on this icon opens the Alert configuration dialog box. (See page 5-272, *Configure alerts*.) If this icon is clicked, then the Alert configuration dialog box is opened with the error selected in the Alert log window highlighted in the Alert configuration dialog box. If this icon is grayed out, it can be enabled by selecting any alert that is shown.

View alert details icon. If this icon is grayed out, it can be enabled by selecting any alert that is shown. (See page 5-277, *View details of the alerts*.)

Copy. This icon becomes available only once an alert is highlighted. This function allows the selected alert to be copied to the Windows clipboard for pasting into any Windows application. Copying can also be done by pressing Ctrl+C.

Level. The error level: Fatal or Error or Warning or Info.

ID. The unique ID number assigned to this error type within MTS4EA.

Class. The area within the hierarchy of the compressed standard; see the screenshot above which shows an error in the Video Object Layer and seven warnings in the Video Object Plane within the MPEG-4 bitstream. The classes vary with the video standards.

Title. The title assigned to this error type.

Decode the number of the decoded frame in which this error was found. See *Decode/display frame/VOP numbers in bitstreams with B-frames/B-VOPs* for more information on decoded versus displayed frame numbers, in bitstreams that include B-frames/B-VOPs. (See page 5-265.)

NOTE. *If <1 is displayed, the error occurred before the decode of the first frame - the error occurred in the header.*

Display. The number of the displayed frame in which this error is shown.

See *Decode/display frame/VOP numbers in bitstreams with B-frames/B-VOPs* for more information on decoded versus displayed frame numbers, in bitstreams that include B-frames/B-VOPs. (See page 5-265.)

NOTE. *If <1 is displayed, this means that the error has occurred before the decode of the first frame, the error has occurred in the header.*

Address (bytes). The byte address in the bitstream where this error was found.

NOTE. *For video bitstreams that have been extracted from container files (for example, MP4 files, 3GPP files, MPEG-2 Packet Streams) this address is the byte address in the extracted video stream, not in the overall container file.*

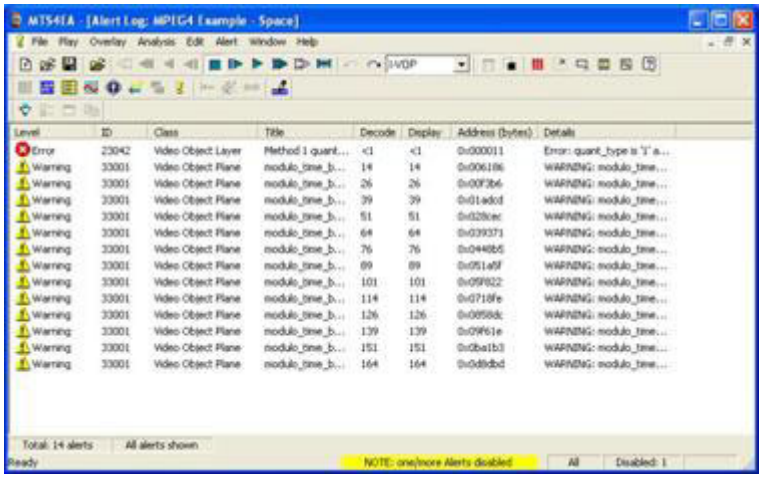
Details. This is the description text that is shown when the popup alert is displayed.

Showing/hiding alerts (alert filter)

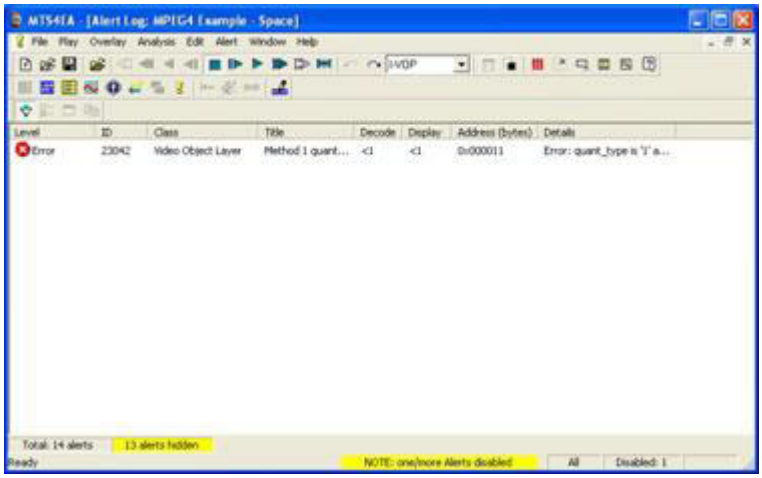
In some cases there can be hundreds or thousands of alerts. These might be known issues and you do not want to list them in the Alert log window (because they would clutter the window unnecessarily).

Therefore, the alerts that are shown in the Alert log window can be filtered. The filtering is based on the setting of the alert popup enable in the Alert configuration (See page 5-272, *Configure alerts*.)

If the filter icon is out all alerts are shown, regardless of whether they are enabled for a popup alert:



If the filter icon is pushed in , only those alerts that are enabled are shown:



(In this example, 13 alerts are hidden out of the total of 14 alerts)

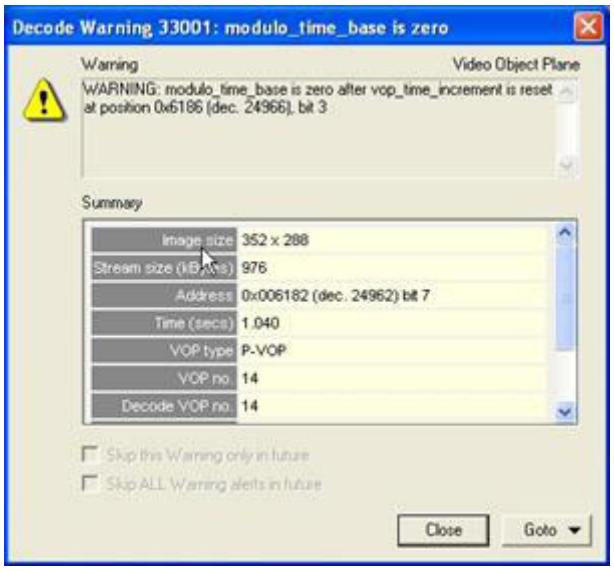
The Alert log filter can also be turned on/off by right-clicking and selecting the option from the popup menu.

NOTE. *When the filter icon is grayed out, all the alerts are enabled for display, so that there are no alerts to filter.*

View details of the alerts. This can be done in several ways:

- Double-click an alert in the Alert log window
- Single-click an alert, right-clicking to get the popup menu, and then click View alert details
- Select an alert by single-clicking an alert, and then clicking the icon

When this is done, a display very similar to the original popup alert display is shown:

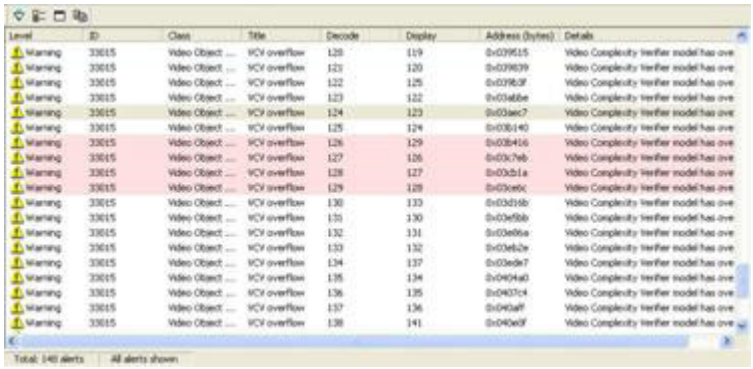


Alert highlighting when seeking forward/backward through video in the step-back buffer. The MTS4EA buffers video (and the data associated with each video frame) in a step-back buffer or cache. (See page 5-54, *General*.)

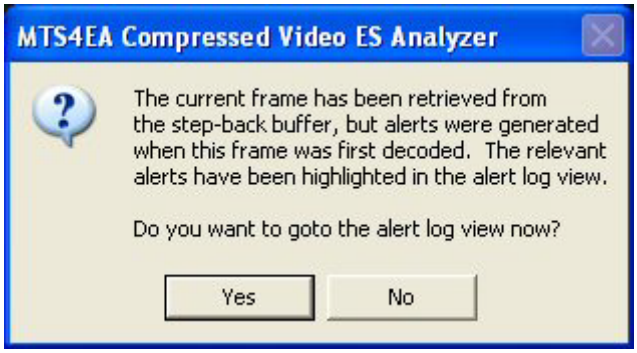
When stepping backward/forward through video that is held in the step-back buffer, to avoid continually having known alerts appear. The MTS4EA only highlights the alerts relevant to the area being skipped in the Alert log.

For example, if you skip forward 100 frames, and there would normally be four popup alerts during the decoding of these frames, the MTS4EA will highlight the four alerts in the Alert log. (If the alerts have been skipped and are hidden in the Alert log, as indicated by the status bar of the Alert log, the MTS4EA cannot highlight the alerts.)

An example of the highlighting of alerts is shown in the following screenshot:



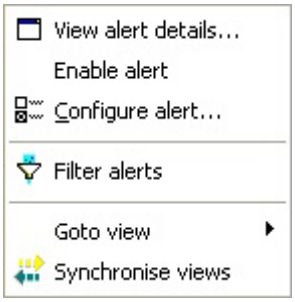
If the Alert log is closed, then the MTS4EA displays a warning, as shown in the following figure:



This is displayed only once for all the Alerts between the relevant frames.

Right-click popup menu

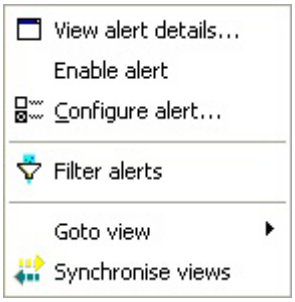
The available functions in the Alert log can be selected by right-clicking and selecting the function from the popup menu:



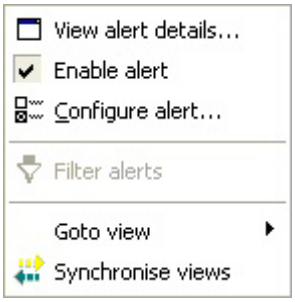
The functions of this menu are:

View alert details. (See page 5-277, *View details of the alerts*.)

Enable alert. An individual alert can be enabled or disabled by clicking on this menu selection:



Alert disabled



Alert enabled

NOTE. *This allows an individual alert to be enabled/disabled from this menu, without having to open the Alert configuration dialog box.*

Configure alert. (See page 5-272, *Configure alerts*.)

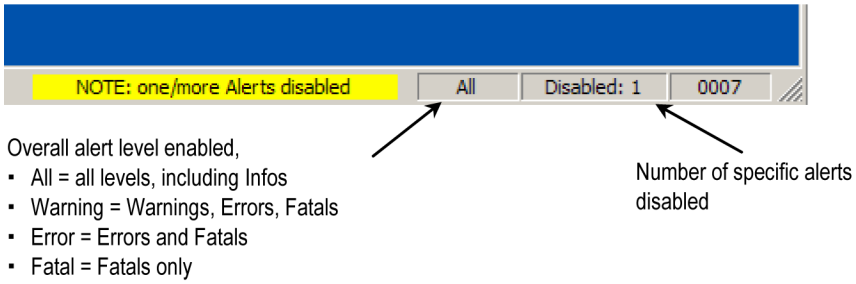
Filter alerts. (See page 5-276, *Showing/hiding alerts (alert filter)*.)

Go to view. (See page 5-8, *Synchronized views/navigating the views*.)

Synchronize views. (See page 5-8, *Synchronized views/navigating the views*.)

Status bar indication of alert status

The status bar displays the status of the alerts.



One or more alerts disabled

When one or more alerts are disabled, either because:

The overall alert level does not include Warnings, Errors and Fatals.

- One or more of the specific alerts have been disabled
- The status bar changes as shown:

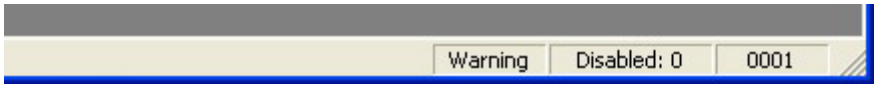


Figure 5-27: Overall alert level: Infos disabled (all other levels enabled)

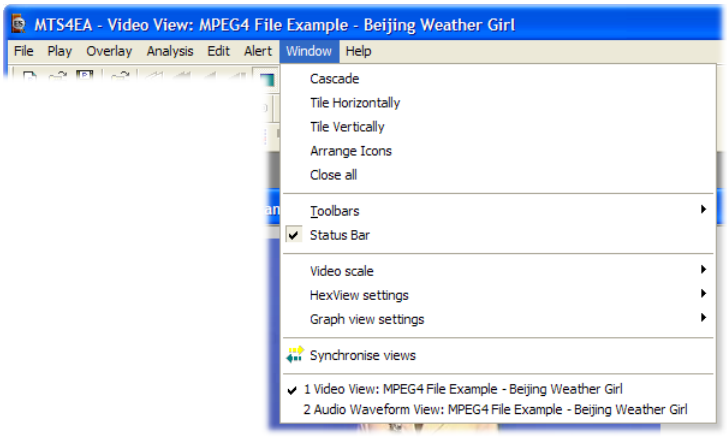
Figure 5-28: Overall alert level: Warnings disabled (Errors and Fatals enabled)

Figure 5-29: Overall alert level: Errors disabled

Figure 5-30: Overall alert level: All disabled - except one/more specified Warning alerts

Figure 5-31: Overall alert level: All disabled - except one/more specific Error alerts

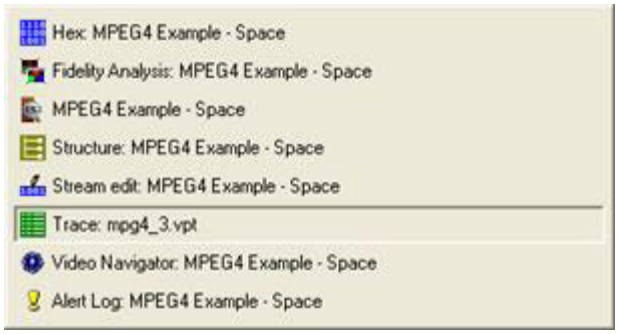
Window menu



The commands on this menu are typical of a standard Windows application, such as arranging windows; list of open windows.

Quickly switch windows

Pressing Ctrl+Tab together displays a list of open MTS4EA windows:

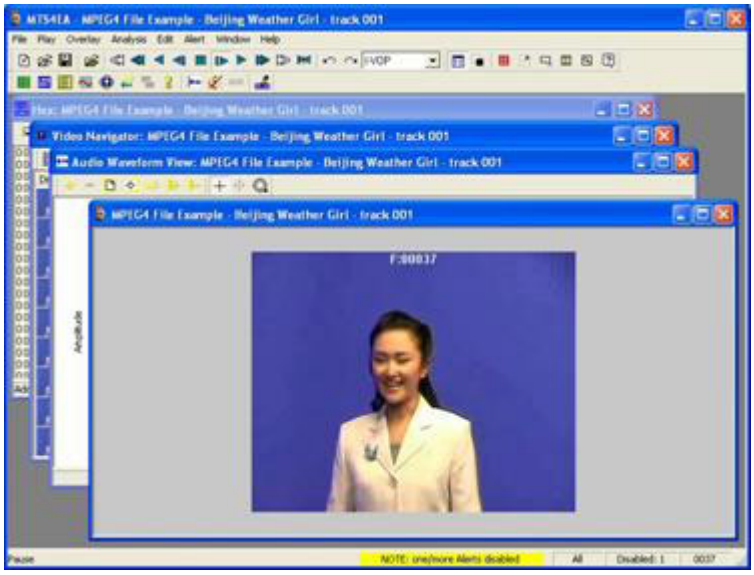


Repeatedly pressing Ctrl+Tab steps through all the open windows. Alternatively, the mouse can be used to select a window.

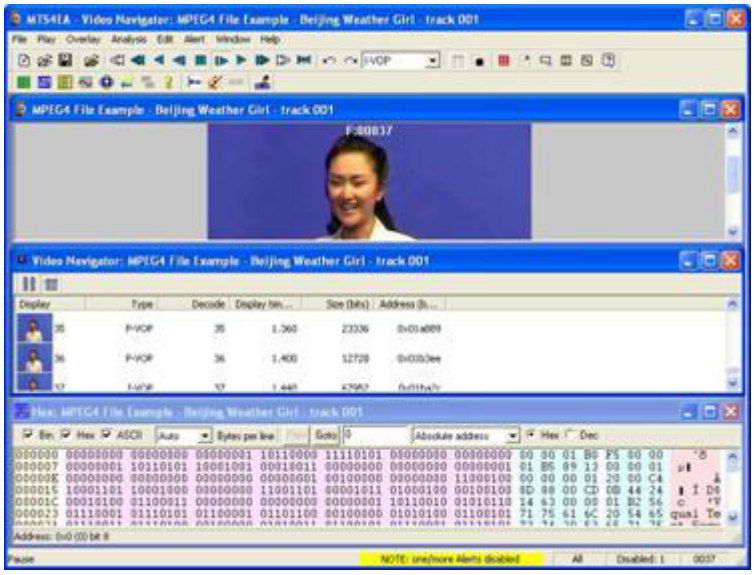
Cascade, Tile, Arrange Icons, Close All functions

These function according to standard Windows commands:

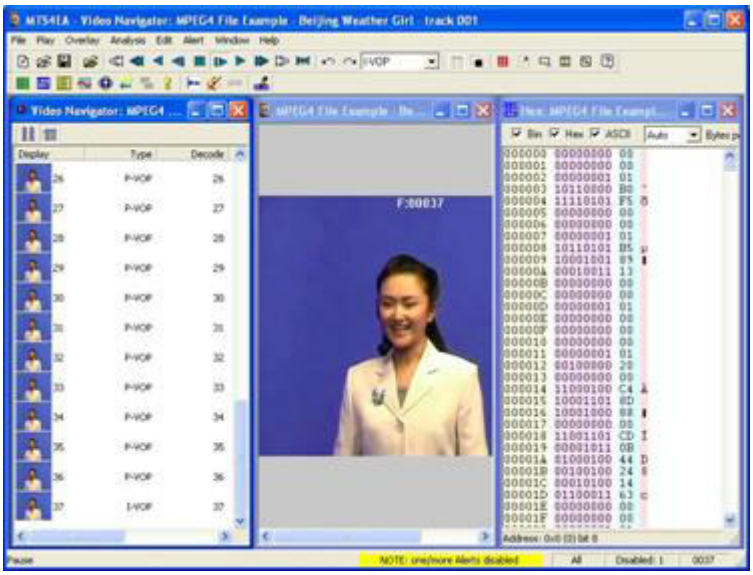
- Cascade arranges the open windows in a cascade:



- Tile Horizontally arranges the open windows as shown:



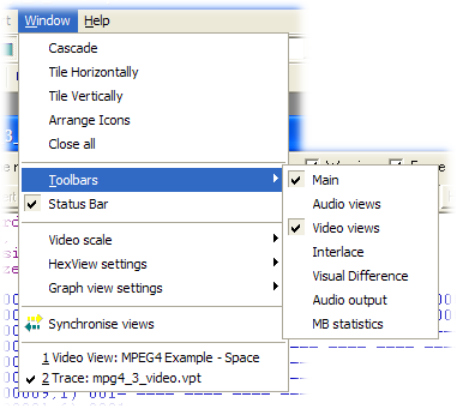
- Tile Vertically arranges the open windows as shown:



- Arrange Icons lines up the windows when they are minimized
- Close all closes all windows in the MTS4EA

Toolbars

In the following example, the Main and Views toolbars are shown, but not the Interlace and Visual Difference toolbars. When selected, a toolbar is displayed with the icons visible. This is indicated by the check mark in the Window menu. Toolbar buttons may be grayed out when a function is inactive.



Main (See page 5-292, *Functions of the Main toolbar buttons*.)

Video Views (See page 5-292, *Functions of the Main toolbar buttons*.)

Interlace The Interlace toolbar is not shown by default; it is always turned on when a stream that is interlaced (or could be interlaced - in H.264/AVC this may not be known until later in the stream) is opened. (See page 5-294, *Interlace toolbar*.)

Visual Difference The Visual Difference toolbar is not shown by default; it is not shown until Visual Difference is enabled on the Overlay menu. (See page 5-126, *Overlay color Ctrl+W*.)

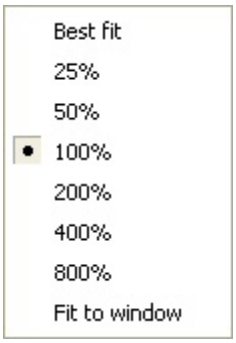
MB Statistics The MB Statistics toolbar is not shown by default. No details are shown on the toolbar until a selection is enabled in the MB Statistics options in the Overlay menu. (See page 5-105, *MB statistics*.)

Status bar

When selected (which is the default setting), the Status Bar is displayed at the bottom of the window. When not selected, it is hidden.

(See page 5-298, *Status bar*.)

Video scale



This sets the size that the decoded video appears in the video window.

Best fit This zooms the displayed video as far as is possible within the video window, while still maintaining the original aspect ratio of decoded video.

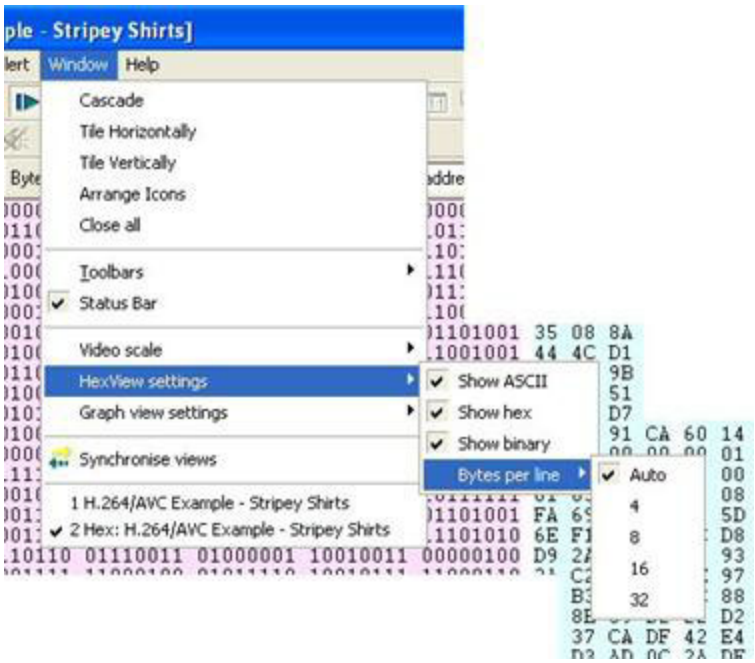
50%; 100% (1:1); 200%; 400%; 800% These set the displayed video to the selected size in the video window.

If the selected size is larger than the available window area, then a section of the video window will be displayed, with scroll bars:

When the MTS4EA starts, it is set by default to 100% or 1:1; the video displays at its actual size with one screen pixel equaling one pixel in the displayed video.

Fit to window This zooms the displayed video to completely fill the video window; it does not maintain the original aspect ratio of decoded video.

HexView settings



The HexView menu has four options:

- Show ASCII
- Show hex
- Show binary
- Bytes per line

Show ASCII, Show hex, Show binary

These options individually turn on/off the display of the specified area of the HexView.

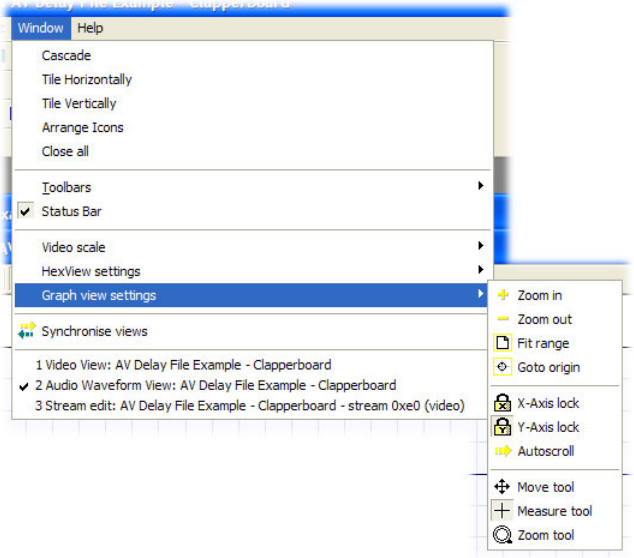
They can be turned on/off from either this menu or from the top line of the HexView window itself, when it is open.

Bytes per line

This sets the displayed number of bytes per line:

- Auto sets the maximum number of bytes that will fit within the active HexView window at the size it is, allowing for the visible ASCII/hex/binary areas.
- When 4, 8, 16 or 32 is selected, then the given number of bytes are displayed, and if the display is too wide for the active HexView window, a horizontal scroll bar is displayed along the lower edge of the HexView window.

Graph view settings



The Graph view menu has ten options, which control how the data appears on the Buffer analysis graph. These options are described in the following table:

Icons in menu	Equivalent toolbar icons	Function
Zoom-in		Zoom in (centered on current window)
Zoom-out		Zoom out (centered on current window)
Fit to range		Fit all data into the visible windo
Goto origin		Go to the origin (time = 0)
X-Axis lock		Lock the X-axis when zooming/scrolling
Y-Axis lock		Lock the Y-axis when zooming/scrolling
Autoscroll		Autoscroll to follow frames as decoded
Move tool		Move window left/right/up/down
Measure tool		Measure the values at center point of +
Zoom tool		Zoom in/out, centered on cursor
Goto view	n/a	See Synchronized Views/Navigating the Views (See page 5-8.)
Synchronise views	n/a	

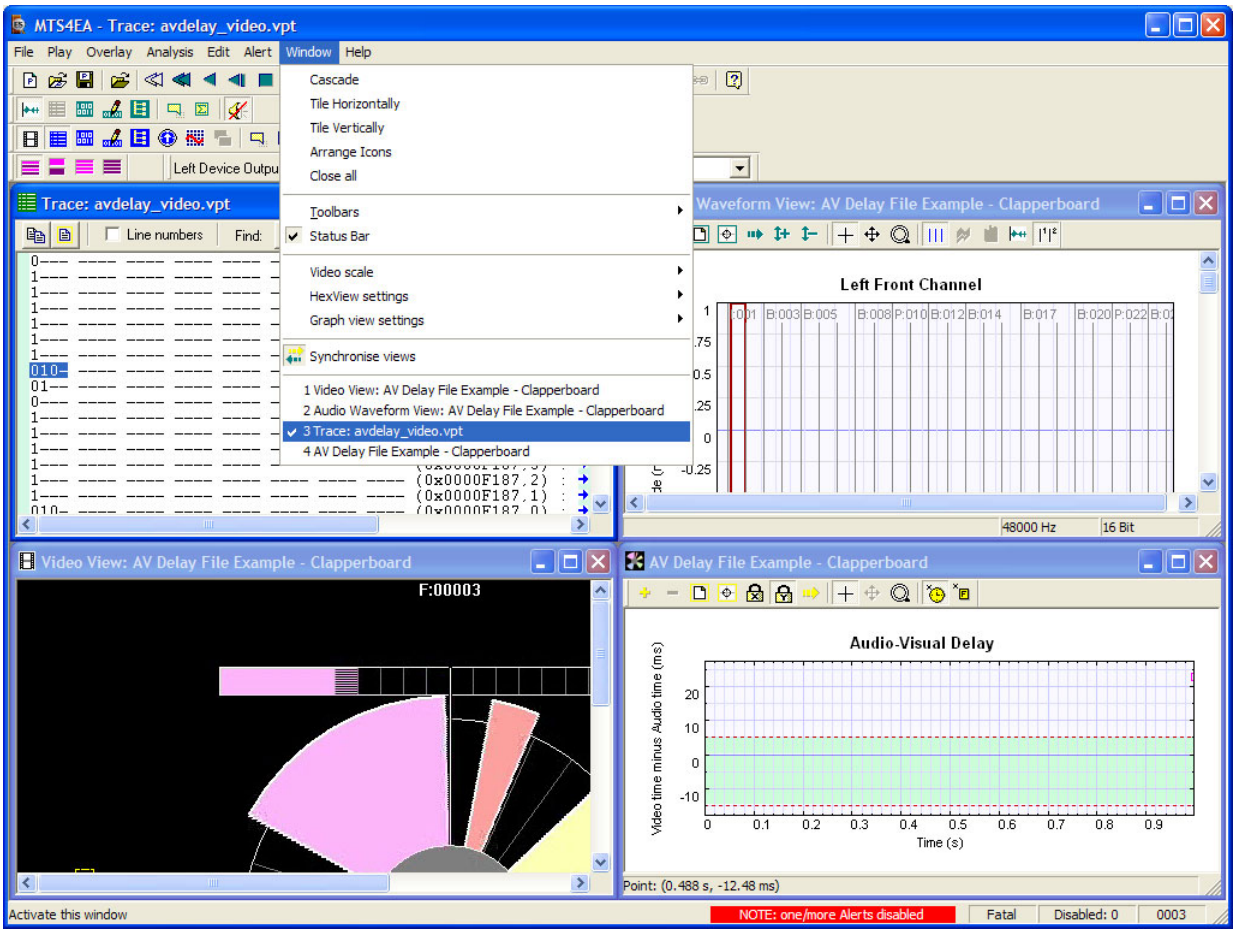
Synchronize views

When Window > Synchronise Views ( ) is selected, all open windows synchronize automatically.

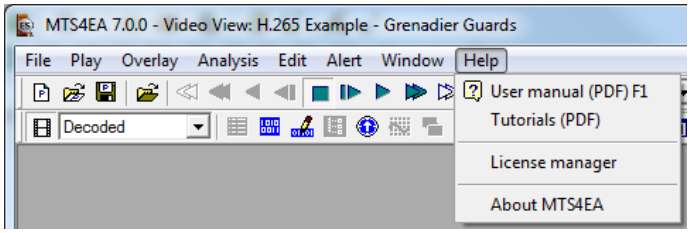
For example, if the Video navigator and Hex views are both open at the same time as the Audio waveform view, the Synchronize views icon is pushed in and the stream is played. The individual video frames appear in the navigator view and the highlighting in the HexView window advances so that all views display the same relative point in the stream at any one time.

Active views

A list of all of the open views is displayed at the bottom of the Windows menu. A check mark appears next to the currently selected view. Highlight an entry to change the selection.



Help menu



This menu provides access to the Help information (User Manual PDF), the tutorials, configuration information, and license information (including licensed options).

User manual (PDF) F1

This displays a PDF version of this user manual when the F1 key is pressed or icon on the toolbar is clicked. In the manual PDF, you can use the table of contents or the Navigation page to quickly view the desired section in the manual. You can also print out the complete manual or specific pages.

See the Adobe Acrobat documentation for details about how to navigate through PDF documents.

Tutorials (PDF)

This displays a PDF version of the tutorials on how to optimize your use of the MTS4EA. It is strongly recommended that you work through the tutorials.

License manager

This displays the MTS4EA License manager, including:

- Current license status information
- Other possible MTS4EA options that can be licensed
- Option to update the license key

For more information on the license manager, see Installation and Licensing.

About MTS4EA

This displays the exact version number of the MTS4EA and the copyright message, and allows access to the MTS4EA build options.

Toolbars, Status bar, and shortcut keys

Icon toolbars

These toolbars provide quick selection of some of the commonly used functions. There are two main icon toolbars, which can be individually moved and docked:

- Main toolbar, containing functions such as play the video and open files.
- Views toolbar, to access the views other than the video views.

NOTE. *There are other toolbars for individual windows and controls, for example for Interlace and Buffer analysis. This section refers to only the main icon toolbars.*

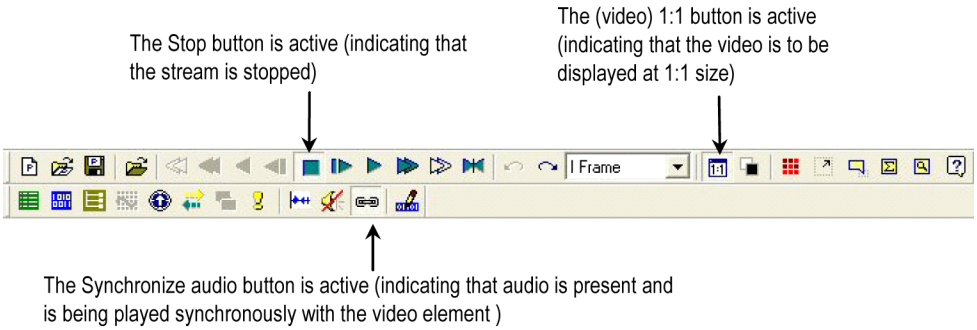
Disabled toolbar buttons

At various times, some of the toolbar buttons are grayed out, indicating that they are disabled because that function cannot be active at the time.

The only buttons available in the above example are the Open video and Help buttons.

Active/inactive toolbar buttons

When a toolbar icon looks as if it is pressed in, it is active (selected).



Moving/docking

The Main toolbar is docked at the top of the window below the main menu by default, but it can be dragged from this position to be floating or be docked on any of the other three edges of the main window, by clicking and dragging the narrow vertical line at the left edge of the toolbar:

The Views toolbar is normally docked below the Main toolbar, but can be moved/undocked as required.

Functions of the Main toolbar buttons

Each of the Main toolbar buttons provides quick access to a specific function of the MTS4EA, as follows:

Toolbar icon	Ctrl +	Name	Description	See page
	O	Open stream…	Open a video/audio file	(See page 5-5.) (See page 5-14.)
	P	Play forward	Play a video/audio file (forwards)	(See page 5-46.)
	F	Fast forward	Fast forward a video file	(See page 5-47.)
	B	Blind fast forward	Blind fast forward a video file (video blanked)	(See page 5-48.)
	A	Pause/Step forward	Pause a video file/ advance by one frame	(See page 5-49.)
	S	Stop	Stop playing a video/audio file	(See page 5-48.)
		Pause on frame…	Pause on specific frame number	(See page 5-53.)
	Shift + P	Play backward	Play a video file backwards	(See page 5-49.)
	Shift + F	Fast backward	Fast backward a video file	(See page 5-49.)
	Shift + B	Blind fast backward	Blind fast backward a video file (video blanked)	(See page 5-50.)
	Shift + A	Pause/Step backward	Pause a video file/ step back by one frame	(See page 5-50.)
	K	Skip forward	Skip forward in increment selected in the skip step control	(See page 5-51.)
	Shift + K	Skip backward	Skip backward in increment selected in the skip step control	(See page 5-51.)
		Skip step	Select skip increment	(See page 5-51.)
	1	1:1	Set video scale: Best fit or 1:1	(See page 5-285.)
	W	Black/White	Set overlay digits to black/white	(See page 5-126.)
	Y	MB types	MacroBlock type overlay	(See page 5-91.)
	E	Motion vectors	Overlay motion vectors display	(See page 5-101.)
	M	MB tooltip	Open/close the MacroBlock tooltip	(See page 5-76.)
	U	Summary tooltip	Open/close the Summary tooltip	(See page 5-66.)

Toolbar icon	Ctrl +	Name	Description	See page
	I	Image inspector	Open/close the Image inspector	(See page 5-89.)
	F1	Help *	Open the User Manual PDF	(See page 5-290.)

* The F1 key is pressed without pressing the Ctrl key

Functions of the Views toolbar buttons

Each of the Views toolbar buttons provides quick access to a specific function of MTS4EA.

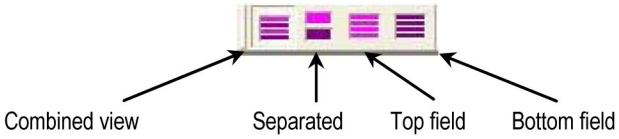
Toolbar icon	Ctrl +	Name	Description	Page
	Shift+T	View trace…	View the currently selected Trace file	(See page 5-198.)
	H	View stream hex…	Open the current steam in the HexView	(See page 5-204.)
	R	View file structure…	Open the current video/container file and view the structure in navigable tree form	(See page 5-210.)
		View buffer analysis…	Graphs of VBV/VCV/VMV. (depends upon standard)	(See page 5-213.)
		Video navigator…	Show thumbnail viewer of each video frame and basic information	(See page 5-195.)
		Synchronize views	Synchronize all open views	(See page 5-288.)
		View fidelity analysis…	Show fidelity analysis	(See page 5-225.)
		View graphical analysis	Show graphical analysis (in Excel)	
		View alert log…	Display log of alerts	(See page 5-274.)
		Audio waveform view…	Open the current audio stream on the Audio waveform view	(See page 5-136.)
		Mute audio	Turn audio off/on	(See page 5-62.)
		Synchronize audio	Synchronize audio and video streams	(See page 5-62.)
		Edit video stream…	Open the stream for editing	(See page 5-260.)

Context-sensitive toolbars/tooltips

These are toolbars and tooltips (or sometimes information windows) that only appear or are only available (not grayed out) for specific video standards or specific functions.

NOTE. *These toolbars dock and are minimized/maximized in the same way as standard Windows toolbars, and in the same manner as the MacroBlock and Summary tooltips.*

Interlace toolbar



Only one of these four buttons is pushed in (active) at any time.

NOTE. *The Interlace toolbar only appears when a bitstream could have Interlace within it. The fact that the Interlace toolbar is there does not necessarily mean that the bitstream is interlaced – only that interlace is permissible/possible in the given standard, Profile and Level.*

These are examples of video streams where the Interlace toolbar can appear (this is not a complete list):

- H.264/AVC (All supported profiles)
- VC-1 Advanced profile
- MPEG-4 Advanced simple profile, levels 4 and 5
- MPEG-2 Main profile, Main level

NOTE. *When the Interlace toolbar has not been automatically displayed by the MTS4EA (or it has been closed), it can be displayed by clicking the Windows menu, and then selecting Toolbars, Interlace.*

Combined frame view. Both fields are shown together, as a single image:

NOTE. *Some data and functions are not available when the video is displayed in this view. For example, the MacroBlock tooltip is empty and cannot be used, and motion vectors, MacroBlock Types and other overlaid data do not appear.*

Separated fields view. The top and bottom fields are shown separately, one above the other:

Top-field view. Only the top-field image data is displayed. For field coded MacroBlocks, the MB tooltip and overlays show data relevant to the top field. For frame coded MacroBlocks, the MB tooltip and overlays show data relevant to the frame.

Each row is shown twice (duplicated) to produce an image to the full image height:

Bottom-field view. As above but for bottom field.

MacroBlock Types color key tooltip

This is an information tooltip that appears when the MacroBlock Types overlay is on: its function is to explain the colors used. This appears when the MB types overlay is on, the icon is active.

NOTE. *To force undocking of the MB types color key tooltip, hold the Ctrl key while dragging with the mouse.*

The MB types color key tooltip can also be turned on again by going to the Window menu, selecting Toolbars, and then clicking the MB types key.

This tooltip can be closed by clicking on the X at the top of the tooltip; to redisplay it click the MB types overlay icon off then on.

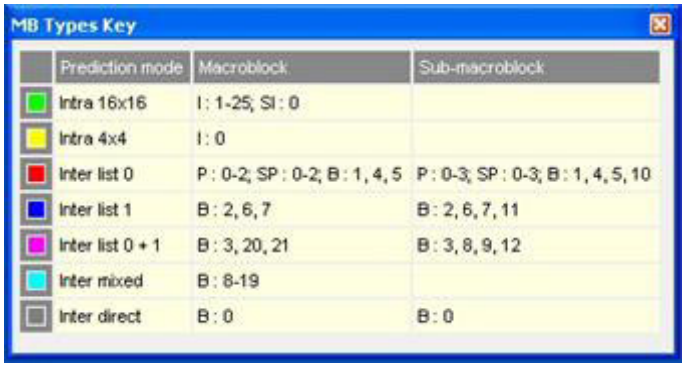


Figure 5-32: H.264/AVC example

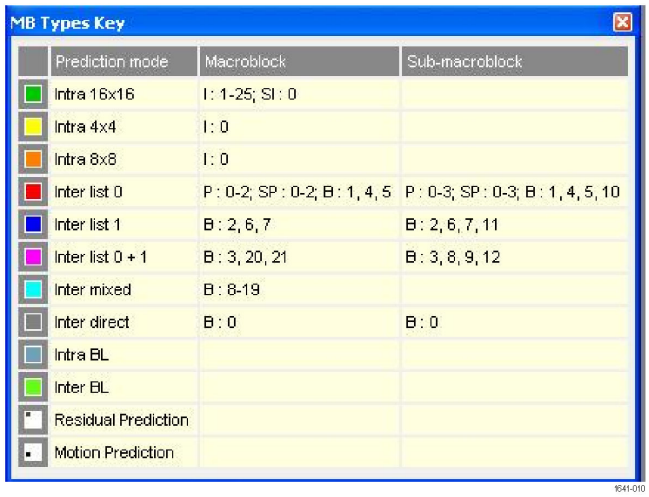


Figure 5-33: H.264/SVC example

NOTE. *The Intra inferred MacroBlock type was shown in yellow in previous versions of MTS4EA. However, because it occurs infrequently and is a 16x16 type, it is now grouped in the Intra 16x16 MacroBlock type, and the Intra 4x4 MacroBlock type is now shown in yellow.*

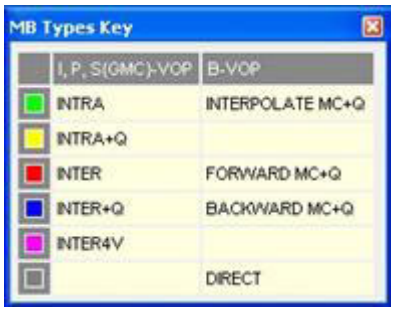


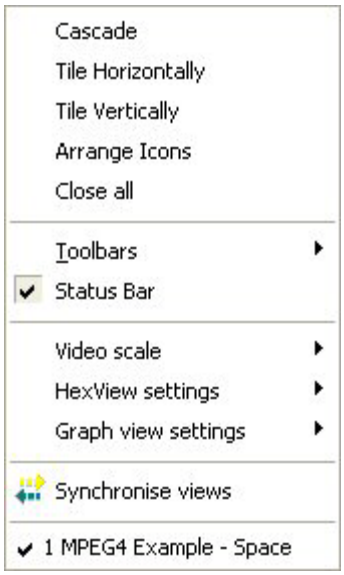
Figure 5-34: MPEG-4 example

Status bar

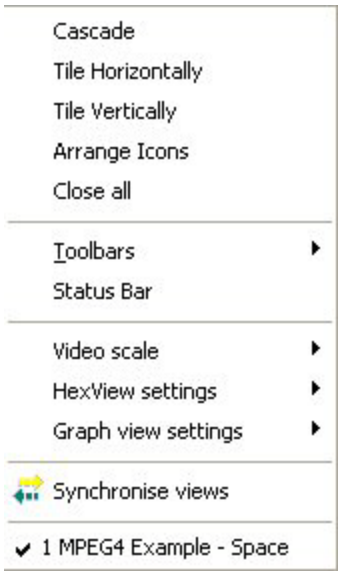
The status bar provides:

- A one-sentence description of each menu item, as the mouse is moved over the menu item
- The stream Play mode. (See page 5-7, *Playing mode restrictions*.)
- Context-sensitive information. The information displayed depends on which window type is open and the current status of the file within that window

The status bar can be displayed or hidden from within the Window menu, in the same way as the icons toolbar:



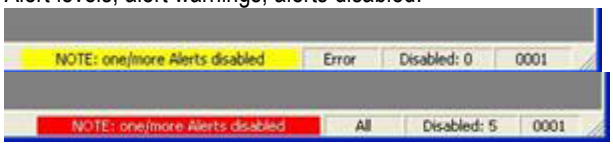
Status bar displayed



Status bar hidden

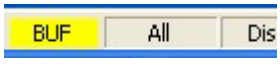
For more information on status bar indications of:

Alert levels, alert warnings, alerts disabled: (See page 5-280.)



Frame range (in/out): (See page 5-114.)

When custom parameters are used for buffer analysis, this is indicated in the status bar of the main MTS4EA window by the word BUF: (See page 5-219.) (See page 5-220.)



Ctrl shortcut keys

All shortcut keys are accessed by holding the Ctrl key and pressing the letter given in the following table, except the F1 key:

Toolbar icon	Ctrl +	Name	Description
	mouse	Force undock	Force undocking of tooltip (for example, Summary/MacroBlock)
	1	1:1	Set video scale: Best fit or 1:1
	A	Pause/Step forward	Pause a video file/advance by one frame
	Shift + A	Pause/Step backward	Pause a video file/ step back by one frame
	B	Blind fast forward	Blind fast forward a video file (video blanked)
	Shift + B	Blind fast backward	Blind fast backward a video file (video blanked)
	C	Copy	Copy highlighted selection
	E	Motion vectors	Overlay motion vectors display
	F	Fast forward	Fast forward a video file
	Shift + F	Fast backward	Fast backward a video file
	G	Graph enable	Enable the graph output
	H	View stream hex…	Open the current steam in the HexView
	I	Image inspector	Open/close the Image inspector
	K	Skip forward	Skip to next I-frame/forward n frames/forward n seconds
	Shift + K	Skip backward	Skip to previous I-frame/backward n frames/backward n seconds
	M	MB tooltip	Open/close the MacroBlock tooltip
	O	Open stream…	Open a video/audio file
	P	Play forward	Play a video/audio file (forwards)
	Shift + P	Play backward	Play a video file backwards
	R	View file structure…	Open the current video/container file and view the structure in navigable tree form
	S	Stop	Stop playing a video/audio file
	T	Trace enable	Enable the Trace output
	Shift+T	View trace…	View the currently selected Trace file

Toolbar icon	Ctrl +	Name	Description
	U	Summary tooltip	Open/close the Summary tooltip
	V	Paste	Paste cut/copied selection in Bitstream editor view
	W	Black/White	Set overlay digits to black/white
	X	Cut	Cut highlighted selection in Bitstream editor view
	Y	MB types	MacroBlock type overlay
	Z	Undo last edit	Undo last edit in Bitstream editor view
	Tab	Switch windows	Quickly switch between open windows
	F1	Help [1]	Go to Help topics
	F3	Find next [1]	In Trace and HexView only
	Shift+F3	Find previous [1]	In Trace and HexView only

[1] The F1, F2 & F3 keys are pressed without pressing the Ctrl key

* The F1, F2 & F3 keys are pressed without pressing the Ctrl key

Alt menu keys

Menus can be selected by pressing the Alt key and the letter underlined in the menu option name. For example, pressing Alt+f will open the File menu. The cursor keys (or the mouse) can be used to select the option required.

Command line/batch mode

This feature allows the MTS4EA to be run in one of two ways:

- Command line mode
- Batch mode

Command line mode

In command line mode, the MTS4EA opens with the Windows display as usual, but using the files and flags specified in the command line.

Batch mode

In batch mode, there is no Windows display (the window is actually minimized and not activated) and the input file, flags and output files are specified in the command line.

The exception to this in batch mode is when no input file is specified. In this case, the MTS4EA opens in Windows mode, allows you to specify a bitstream file name, and then closes Windows mode and goes back to batch mode.

The MTS4EA exits automatically from batch mode when the last selected frame has been decoded.

NOTE. *Remember the –b option to run MTS4EA in batch mode (rather than command-line mode).*

Running the MTS4EA in command line/batch mode

To use command line or batch mode:

- Use Windows Start/Run and type in the line
- Open a command prompt window (typically from Windows Start/Programs/Accessories) and enter the line

The format of the line to be entered is:

`MTS4EA [options...]  <filename>` where:

- `[options...]` is zero or more of the options listed in 0
- `<filename>` is the input video file to be decoded

NOTE. *If you are using the demo version of MTS4EA, then you should use mts4ea_demo instead of mts4ea in the command line.*

Specification of filename (input video file)

The input video filename must be the full filename (including any file extension after the . [period]).

Using demonstration sequences in batch mode. To use the batch mode with the example files provided in the File menu (File > Examples…) you must use the –d option with the file name specified in the following table.

Demo sequence	Standard	File name
H.265/HEVC byte streams		
Forbidden city	HEVC Main Profile / Main Tier / Level 6	hevc_1
Grenadier guards	HEVC Main Profile / Main Tier / Level 6	hevc_2
Red portico	HEVC Main Profile / Main Tier / Level 2.1	hevc_3
H.264/AVC byte streams		
Neon Night	H.264/AVC Baseline Profile/Level 2	avc_1
Canary Wharf	H.264/AVC Extended Profile/Level 3	avc_2
Bus Junction	H.264/AVC Baseline Profile/Level 3	avc_3
Grenadier Guards	H.264/AVC Main Profile/Level 3	avc_4
Stripey Shirts	H.264/AVC High Profile/10	avc_5
Times Square	H.264/AVC High Profile/4:4:4	avc_6
H.264/SVC byte stream		
Soccer	Scalable Baseline Profile/Level3	svc_1
VC–1 advanced stream		
Central Park	VC-1 Advanced	vc1_1
MPEG–4 elementary streams		
Woman Drinking	MPEG-4 Main Profile	mpg4_1
Train in Station	MPEG-4 Main Profile	mpg4_2
Space	MPEG-4 Simple Profile	mpg4_3
Man Walking	MPEG-4 Advanced Simple Profile	mp4asp_1
Synthetic	MPEG-4 Advanced Simple Profile	mp4asp_2
Window Car	MPEG-4 Advanced Simple Profile	mp4asp_3
H.263 streams		
Rally (250k)	H.263 Baseline/MPEG-4 Short Header	h263_1
Rally (2M)	H.263+ with Annex T	h263_2
Person Track	H.263+ with Annexes D, F, I, J, S, T	h263_3
MP4 files		
Packet Woman	MP4 file containing MPEG-4 Simple Profile/ Level 1	mp4fil_1
Piccadilly Circus	MP4 file containing MPEG-4 Simple Profile/ Level 2	mp4fil_2
Beijing Weather Girl	MP4 file containing MPEG-4 Simple Profile/ Level 5 (with audio)	mp4fil_3

Demo sequence	Standard	File name
Las Vegas	H.264/AVC Baseline Profile/Level 1.2	mp4fil_4
Live Band	MPEG-4 Simple Profile/Level 1 (video); AAC HE (audio)	audio_4
Stage Interview	MPEG-4 Simple Profile/Level 1 (video); AAC Main (audio)	audio_5
3GPP file		
Mobile Hands	3GPP file containing MPEG-4 Simple Profile/ Level 1	3gpfil_1
MPEG-2 program streams		
Bus Junction	MPEG-2 Main Profile/ Main Level	mp2_1
Person Track	MPEG-2 Main Profile/ Main Level	mp2_2
Grenadier Guards	MPEG-2 Main Profile/ Main Level	mp2_3
MPEG-2 transport streams		
Golden Gate	H.264/AVC Main Profile	mp2ts_1
Mangroves	MPEG-2 MP@ML	mp2ts_2
Captain Bob	MPEG-2 MP@ML (with MPEG-1 Audio Layer II)	mp2ts_3
Microsoft® ASF files		
Beach Girl	Simple Profile	asf_1
Great Wall	Main Profile	asf_2
ADTS streams		
Dance Track	AAC LC	audio_6
A/V Delay stream		
Clapperboard	MPEG-2 Main Profile / Main Level (video) Dolby AC-3 (audio)	avdelay

To use these demonstration sequences, the option -d is used. (See page 5-305, *List of options*.)

In addition, the following YUV files are provided for fidelity analysis and visual differencing with the example streams:

YUV files	Use with compressed demo sequence	Frame rate	Number of frames
guards_yuv	H.264/AVC Grenadier Guards MPEG-2 Grenadier Guards	25	10
man_walking_yuv	MPEG-4 Man Walking	30	All

Command line/batch mode options

Form of options. All options take the following form:

```
-x <value>
```

(Where x is the option and <value> is the value entered; for some options there is no value.)

If there is a value, there is always a <space> between the -x and the value.

All options must be separated by spaces.

NOTE. *Where filenames or folders have spaces in the path or name, these must be put inside double quotes.*

There is limited checking on the options/option values.

List of options.

Option	Value (if any)	Description
-a	<alertlevel>	Alert level, which must be one of the following text strings: error, warn, or fatal. If the warning alert level is on, trace outputs will also include error and fatal level alerts. Fatal level alerts are always on.
-b		When present, enable batch mode. If not present, command line mode is used.
-c	<codec>	Override the automatic codec detection. (See page 5-309, *CODEC identities*.)
-d		When this flag is present, it signifies that one of the demonstration tutorial video sequences as provided with the MTS4EA is to be used as the input video file. The particular demonstration sequence to be used is specified by the filename. (See page 5-303, *Using demonstration sequences in batch mode*.). (This filename is given at the end of the command line, like any other input filename.)
-e	<number>	Audio lead tolerance in milliseconds.
-f	<firstframe>	Integer value specifying the first frame to be used for the YUV and Trace outputs. If this option is not specified then frame number 1 (the first frame in the video sequence) is used.
-g	<number>	Audio lag tolerance in milliseconds.
-h		Displays help Window (MTS4EA opens after OK is clicked).

Option	Value (if any)	Description
-i	<trackID>	Where the input file is a container file such as MP4 or 3GPP or MPEG-2 Program Stream, there can be more than one video track in the file. If -i is not specified then the first video track found is used. If the track ID specified is not present, a file missing error is given. (See page 5-308, *Specifying track identities*.)
-j	<trackID>	Same as for –i, but audio related.
-k	[F:]<integer>	Specify a byte offset entry point to the video elementary stream. If the F: is used then the integer is treated as a frame offset.
-l	<lastframe>	Integer value specifying the last frame to be used for the YUV and Trace outputs. If this option is not specified, the last frame in the video sequence is used.
-m	<size>	Integer value specifying the maximum video trace file size, in megabytes. If -m is not given or the value is set to 0, there is no limit on the size of the trace file (other than the available disk space).
-n	<size>	Integer value specifying the maximum audio trace file size, in MB.
-o	<options>	The Video Trace file options are any combination of the following letters with no spaces between them: **Option — Video trace output generated** a — Macroblock fidelity b — Parse bitstream d — DCT level f — Frame summary g — GOB summary i — Interpret l — Alerts m — Macroblock summary p — Pixel level s — Stream summary u — Buffer v — vop fidelity
-p	<options>	The Audio Trace file options are any combination of the following letters with no spaces between them: **Option — Audio trace output generated** b — Parse bitstream h — Audio histograms i — interpret

Option	Value (if any)	Description
-q	<name>	Enforces extra restrictions per standard: **Options** **Restrictions enforced** aribb14 Transform ARIB TR-B14 transform intermediate value checking
-r	<yuvfile>	Full name of the YUV reference file used for fidelity analysis. (See also the options -s, -u and -v.)
-s	<size>	Header skip of the YUV reference file used for fidelity analysis. (See also the options -r, -u and -v.)
-t	<tracefile>	Full name of the Video Trace output file. If no video trace output file is specified, the trace options are ignored.
-u	<number>	Frame rate of the YUV reference file used for fidelity analysis. The number can be an integer, or a fraction expressed as a/b (where a and b are integers) or a decimal number. (See also the options -r, -s and -v.)
-v	<string>	Metric used for fidelity analysis: the string value must be one of the following: psnr - PSNR with 255 signal range psnritu - PSNR with ITU-R BT.601 signal range rmse - Root Mean Square Error mse - Mean Square Error mad - Mean Absolute Differences sad - Mean Absolute Differences (See also the options -r, -s and -u.)
-w	<error_file>	Output trace file with warnings/errors/fatal alerts only - no other trace information. <error_file> is the name of the file that holds the warnings/errors/fatal alerts. If there are no warnings/errors/fatal alerts, the <error_file> is zero length

Option	Value (if any)	Description
-x	<options>	The format of image samples in uncompressed video files for input or output. This option is used when working with H.264/AVC High Profile to denote: The correct format for the uncompressed input when doing PSNR analysis The format for uncompressed output The format option is one of the following codes: 1 - one byte per sample 2le - two bytes per sample, little-endian 2be - two bytes per sample, big-endian
-y	<yuvfile>	Full name of the YUV output file (See page 5-310.)
-z	<tracefile>	Full name of the audio trace output file. If no audio trace output file is specified, the trace options are ignored.
-B	<layerID>	This selection applies to H.264/SVC streams only. Layer ID to be analyzed. This must correspond to one of the Layer IDs as specified in the scalable_info SEI message (Integer value).

Specifying track identities

The format for specifying the track <trackID> to decode for the command line options –i and –j is as follows: A series of track numbers separated by a period (.) that give the path to the elementary stream from the top container. The numbers can be base 10 or in a hexadecimal format. Optionally a codec may be specified after an equals sign (=) at the end of the path. If the codec is specified, then MTS4EA will force that decoder to be attached to the stream. To specify just the codec to attach use the –c option.

Example 1: `mts4ea –i 0x1000.0xe0=AVC jaws4.ts`

This looks on track 0x1000 (4096) and then on sub track 0xe0 (224) and tries to attach the H.264/AVC decoder. This is a typical format for transport streams with the PID first, followed by the PES stream_id.

Example 2: `mts4ea –c VC1A lostInSpace.vc1`

This forces the VC-1 Advanced Profile decoder to be used for the input stream. This can be useful in rare cases when the stream type is not recognised or is mis-recognised by MTS4EA.

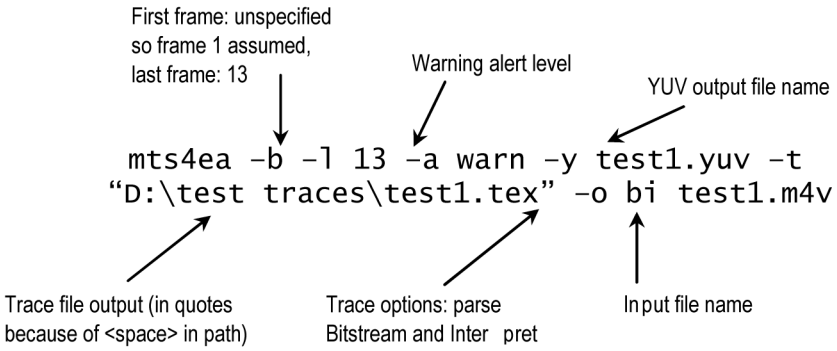
CODEC identities

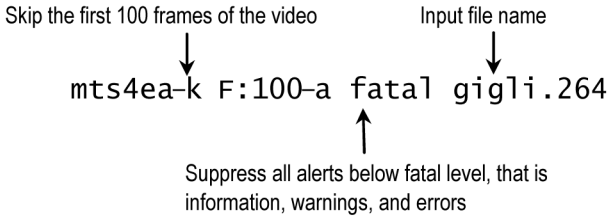
The allowed values for the codec are given in the following table. If the value contains a space then it must be given within double quotes:

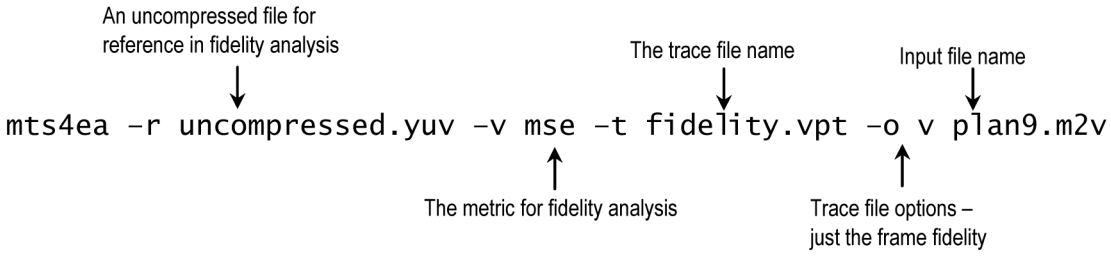
Value	Type	Description
H.263	video	H.263
AVC	video	H.264/AVC
SVC	video	H.264/SVC
HEVC	video	H/265/HEVC
MPEG-2	video	MPEG-2 video
MPEG-4	video	MPEG-4 part 2 video
VC1A	video	VC-1 Advanced Profile
VC1SM	video	VC-1 Simple/Main Profile
ASF	container	Microsoft Advanced Systems Format
MP4	container	ISO
3GPP	container	3GPP
ADTS	container	ISO
MPEG-2 TS	container	ISO
MPEG-2 PS	container	ISO
MPEG-2 PES	container	ISO
AAC	audio	ISO
MPEG-2 audio	audio	ISO
MPEG-1 audio	audio	ISO
AC-3	audio	Dolby Digital AC-3

Example command lines

Three examples of valid command lines follow:







Format of uncompressed video file output from batch mode

NOTE. *The uncompressed video output file in this section is the uncompressed video output resulting from decoding the compressed video; this is a different file than the uncompressed video reference file that is used when doing fidelity analysis.*

The YUV data is either:

- 8 bits per sample, 4:2:0
- More than 8 bits per sample, and/or 4:2:2 or 4:4:4 (as used by H.264/AVC High Profile/FREXt, High/10, High/4:2:2, High/4:4:4)

YUV format 8 bits per sample 4:2:0. The YUV file output is raw YUV with no headers of any kind. This is the same format as used by the Microsoft MPEG-4 Part 2 reference encoder Reference [6] (see *Compression Standards and File Types* section) (See page 4-1.) and as used commonly by other programs:

- No headers of any kind (no file or frame headers)
- One byte per sample
- Row raster order (top picture row first)
- Planar YUV 4:2:0 sub-sampled (4 bytes of Y data for each byte of U data and each byte of Y data)
- Y plane values are 0-255 unsigned
- U and V plane values are unsigned with a DC offset of 128

Other uncompressed formats. The general uncompressed video file format is as follows:

- No headers of any kind (no file or frame headers)
- Concatenated planar image data
- Row raster order (top picture row first)
- Unsigned samples

For 8-bit sample depth:

- One byte per sample

For 9-16 bit sample depth:

- Two bytes per sample
- Both little- and big-endian byte orders supported

For YUV format:

- Concatenated Y, U and V planes
- U and V planes sub-sampled as required
- Y plane samples are unsigned
- U and V plane samples are unsigned with a DC offset of 2^{-1}, where *n* is the chroma sample bit depth

For RGB format:

- Concatenated R, G and B planes
- For grayscale format:
- Luma plane only

Bitstream syntax debugging

Bitstream syntax debugging

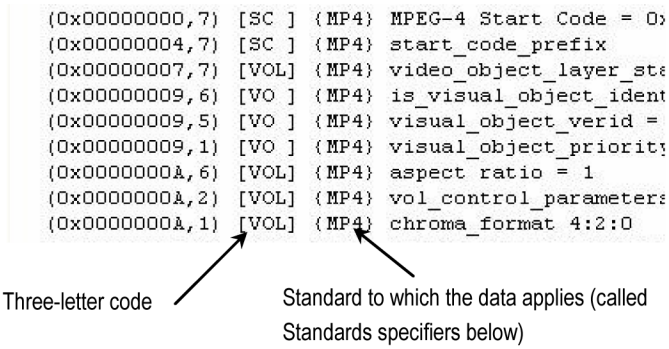
This section gives some tips and suggestions on debugging the syntax of video bitstreams syntax.

In order to understand these, you will need to have the relevant video standard(s) to hand.

NOTE. *This manual section applies to streams that only contain video, such as MPEG-4 Elementary Streams - it does not apply to debugging container files such as MP4/3GPP, which can also contain audio and other data.*

General codes used in Trace files and Alerts

Trace data and Alerts will often have two sets of information fields, such as shown below:



Explanation of three-letter codes

Three-letter codes are used within the outputs generated by Trace/Interpret, Alerts and (sometimes) Trace/Parse Bitstream.

The codes are used to indicate the part of the picture hierarchy that applies to the text data given.

Three-letter codes for H.265/HEVC

Three-letter code	Syntactic structure in H.265/HEVC
BSN	byte_stream_nal_unit
NALU	
NAL	nal_unit
STB	rbsp_slice_trailing_bits
RTB	rbsp_trailing_bits
Top-level RBSPs	
AUD	access_unit_delimiter_rbsp
ESQ	end_of_seq_rbsp
EST	end_of_stream_rbsp
FLD	filler_data_rbsp
PPS	pic_parameter_set_rbsp
SEI	sei_rbsp
SPS	seq_parameter_set_rbsp
VPS	video_parameter_set_rbsp
PTL	profile_tier_level
SCL	scaling_list
DRP	dec_ref_pic_marking
PWT	pred_weight_table
RPL	ref_pic_list_modification
SLH	slice_segment_header
Slice segment data	
SLD	slice_segment_data
SAO	soa_data
CTU	coding_quadtree
CDU	coding_unit
PRU	prediction_unit
TRT	transform_tree
TRU	transform_unit
RES	residual
SEI	
BUP	buffering_period
PIT	pic_timing
PSR	pan_scan_rect
FLP	filler_payload
UDR	user_data_registered_itu_t_t35
UDU	user_data_unregistered
REP	recovery_point

Three-letter code	Syntactic structure in H.265/HEVC
SCI	scene_info
PSS	picture_snapshot
PRS	progressive_refinement_segment_start
PRE	progressive_refinement_segment_end
FGC	film_grain_characteristics
PFH	post_filter_hint
TMI	tone_mapping_info
FPA	frame_packing_arrangement
DIO	display_orientation
SPI	structure_of_pictures_info
APS	active_parameter_sets
DUI	decoding_unit_info
TSL	temporal_sub_layer_zero_index
SCN	scalable_nesting
RRI	region_refresh_info
DPH	decoded_picture_hash
RSM	reserved_sei_message
VUI	
HRD	hrd_parameters
VUI	vui_parameters

Three-letter codes for H.264/AVC

Three-letter code	Syntactic structure in H.264/AVC
BSN	byte_stream_nal_unit
NALU	
NAL	nal_unit
STB	rbsp_slice_trailing_bits
RTB	rbsp_trailing_bits
RBSPs	
AUD	access_unit_delimiter_rbsp
ESQ	end_of_seq_rbsp
EST	end_of_stream_rbsp
FLD	filler_data_rbsp
PPS	pic_parameter_set_rbsp
SEI	sei_rbsp
SPS	seq_parameter_set_rbsp
SLA	slice_data_partition_a_layer_rbsp
SLB	slice_data_partition_b_layer_rbsp
SLC	slice_data_partition_c_layer_rbsp
SLW	slice_layer_without_partitioning_rbsp
Slice header	
DRP	dec_ref_pic_marking
PWT	pred_weight_table
RPL	ref_pic_list_reordering
SLH	slice_header
Slice data	
MBL	macroblock_layer
MBP	mb_pred
RES	residual
CAB	residual_block_cabac
CAV	residual_block_cavlc
SLD	slice_data
SMP	sub_mb_pred
SEI	
BUP	buffering_period
DRR	dec_ref_pic_marking_repetition
FLP	filler_payload
FFF	full_frame_freeze
FFR	full_frame_freeze_release
FRS	full_frame_snapshot

Three-letter code	Syntactic structure in H.264/AVC
MCS	motion_constrained_slice_group_set
PSR	pan_scan_rect
PIT	pic_timing
PRE	progressive_refinement_segment_end
PRS	progressive_refinement_segment_start
REP	recovery_point
RSM	reserved_sei_message
SCI	scene_info
SEM	sei_message
SEP	sei_payload
SPP	spare_pic
SSC	sub_seq_characteristics
SSI	sub_seq_info
SSL	sub_seq_layer_characteristics
UDR	user_data_registered_itu_t_t2
UDU	user_data_unregistered
VUI	
HRD	hrd_parameters
VUI	vui_parameters

If appropriate, these are paired together in the form: [TTT:LLL] where the following are used to generate the data given in the Trace file:

- TTT indicates the top called syntactic structure
- MMM indicates the middle called syntactic structure
- LLL indicates the lowest called syntactic structure

Three-letter codes for MPEG-4 and H.263

Three-letter code	Name in H.263	Name in MPEG-4	Standard section
[SC]	Picture Start Code PSC	Start Code	6.2, table 6.3
[VOS]	-	VisualObjectSequence	6.2.2
[VO]	-	VisualObject	6.2.2
[VOL]	-	VideoObjectLayer	6.2.3
[PL]	Picture Layer	-	
[GOV]	-	Group_of_VideoObjectPlane	6.2.4
[VOP]	-	VideoObjectPlane	6.2.5
[VPS]	Picture Layer	video_plane_with_short_header	6.2.5.2
[GOB]	GOB (Group of Blocks) layer	gob_layer	6.2.5.2
[MB]	MB (MacroBlock) layer	macroblock	6.2.6.2
[MOV]	MVD, MVD $_{2\text{-}4}$	motion_vector	6.2.6.2
[BLK]	Block layer	block	6.2.7
[inf]	Information about the internal state of the decoder or variables that may be described in the relevant		

Three-letter codes for MPEG-2

Three-letter code	Syntactic structure in H.264/AVC
Top level	
EUD	extension_and_user_data
GPH	group_of_pictures_header
PCD	picture_data
PCH	picture_header
PCX	picture_coding_extension
SQH	sequence_header
SQX	sequence_extension
VSQ	video_sequence
Components	
APS	additional_pan_scan_parameters
ARW	active_region_window
BLK	block
CBP	coded_block_pattern
CDD	content_description_data
CPL	coded_picture_length
CTC	capture_timecode

Three-letter code	Syntactic structure in H.264/AVC
EXD	extension_data
FCT	frame_or_field_capture_timestamp
MBK	macroblock
MBM	macroblock_modes
MVS	motion_vectors
NSC	next_start_code
PAD	padding_bytes
SLI	slice
USD	user_data
VEC	motion_vector
Extensions	
CRX	copyright_extension
CPX	camera_parameters_extension
ITU	itu_t_extension
PDX	picture_display_extension
PSS	picture_spatial_scalable_extension
PTS	picture_temporal_scalable_extension
QMX	quant_matrix_extension
SDX	sequence_display_extension
SSX	sequence_scalable_extension

If appropriate, these are paired together in the form: [TTT:LLL] where the following are used to generate the data given in the Trace file:

- TTT indicates the top called syntactic structure and
- MMM indicates the middle called syntactic structure(s) and
- LLL indicates the lowest called syntactic structure

Explanation of bit/bit start

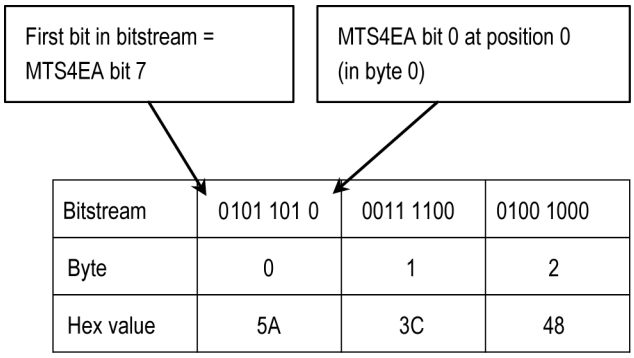
In each of the compression standards there are many bitstream fields which give values which are least significant bit first (lsbf) and many others which are most significant bit first (msbf). There are also many others that are simply bit-patterns: they do not have a direct numeric value.

MTS4EA has adopted the convention of naming the left-most bit as bit 7; as this is correct for presentation of numbers in bytes where the left-most bit is the most significant bit of the byte.

For msbf bitstream fields, numbering bit 7 as the first bit in each byte is correct.

For lsbf bitstream fields, the bit order has to be reversed when calculating the value given (and of course, all other decoders have to do this, not just MTS4EA).

To explain this further, consider a hypothetical bitstream:



Bitstream	0101 101 0	0011 1100	0100 1000
Byte	0	1	2
Hex value	5A	3C	48

Using the preceding example bitstream, the following statements are illustrated in the following table:

- The position is 1 (byte 1)
- The bit/bit start is 4 (in byte 1)
- The length is 7 bits

Bitstream	0101 1010	0011 1100	0100 1000
Byte	0	1	2
Hex value	5A	3C	48

This field (1110001) has a hexadecimal value of 0x71 if most significant bit first (1110001) or 0x47 if least significant bit first (1000111).

Bitstream syntax debugging using the MTS4EA

Using the various tools in the MTS4EA, bitstream syntax debugging can be quick and straightforward.

How the MTS4EA does syntax checking

The MTS4EA checks for hundreds of potential errors to ensure that the bitstream under test complies with the relevant standard. The MTS4EA checks for errors at multiple levels:

- At the first level, it will check for the presence of a particular allowable bitstream field (for example, if the picture size should be specified, that data consistent with this is present within the bitstream).
- At the second level, it will check within the bitstream field that the value given is permitted, that is, the bitstream field is self-consistent. (For example, that the values given for picture size are within the permitted range of values).
- At the third level, it will check that the values across multiple bitstream fields are consistent (for example, that there are the correct number of MacroBlocks for the picture size).

Principal tools within the MTS4EA for syntax debugging

There are three main tools for syntax checking:

- The Alerts provided when displaying the video sequence using the MTS4EA (see description of alert levels in the *How to use the MTS4EA* section).
- The Trace/Parse Bitstream function (See page 5-1, *How to use the MTS4EA*.)
- The Trace/Interpret function (See page 5-1, *How to use the MTS4EA*.)

Essentially, of the three principal tools provided within the MTS4EA (Alerts, Trace/Parse Bitstream and Trace/Interpret):

- Alerts give a summary of all levels of debug (first, second and third, as mentioned in *How the MTS4EA does syntax checking*)
- Trace/Parse Bitstream concentrates on the first and second levels
- Trace/Interpret concentrates on the second and third levels

All three tools are provided so that:

- A quick overview is available (using the Alerts)
- Sometimes it is useful to be able to continue debugging at a lower level, even if some higher-level errors are found (which is the reason for Trace/Parse Bitstream and Trace/Interpret)

NOTE. *You can open multiple windows in the MTS4EA, so that one window has the video, with windows for the Trace/Parse Bitstream output and a third window for Trace/Interpret output, all on the same video sequence.*

Procedure for bitstream syntax debugging

The following is a suggested procedure for getting started with debugging the video bitstreams you have generated.

After a period of time, you will generate your own methodologies, but this is a good starting point.

General tips

Start debugging with short video sequences. It is easier to debug a shorter video sequence than a longer one, so start debugging with shorter versions of your compressed video sequences.

If you have to Trace out all the data of a whole video sequence into a Trace file, this file will not be too (unmanageably) large.

Use the HexView. It is often useful to also view the video bitstream file in the HexView provided within the MTS4EA, looking at the same point as the displayed video or a printed trace.

(See page 5-1, *How to use the MTS4EA*.)

Overview

In general, it is better to proceed as follows:

1. Alerts
2. Analysis/Trace/Parse Bitstream (with or without the HexView)
3. Analysis/Trace/Interpret

For some debugging, it is more useful to use Trace/Interpret first, followed by Parse Bitstream, if required, depending on the nature of the errors.

For each of these steps, use the appropriate procedure:

Alerts debugging

1. Open the video sequence under test (using File/Open stream)
2. The Alerts level is set to All when any video stream is opened.
3. If the syntax of the header of the bitstream is (reasonably) correct, then there will not be any Error or Warning messages; in which case, proceed to step 5.
4. If there are Error or Warning messages at this point, read the message carefully, noting the position (byte address) and bit position given. (See *How to use the MTS4EA* for information on how to interpret the data provided in these pop-up windows. (See page 5-1.)) At this point, it is likely you will need to understand the bits in the bitstream go to *Trace/Parse bitstream debugging*. (See page 6-11.)
5. If there are no Errors or Warnings on File/Open, play the video sequence and see if any Errors or Warnings occur. If they do not, go to *Trace/Parse bitstream debugging*. (See page 6-11.)
6. If there are Errors and/or Warnings, stop playing the video, turn on the Summary Tooltip, (see *How to use the MTS4EA* section) and play the video again until it stops at the first Error/Warning. (See page 5-1.)
7. Note the Frame number at which the Error/Warning occurs, and the other data (or open multiple copies of MTS4EA). then go to *Trace/Parse Bitstream Debugging*. (See page 6-11.)

Trace/Parse bitstream debugging

Procedure.

1. If you know the frame number at which an Error or Warning occurs, set the Frame Range to one frame before and one frame after the location of the Error or Warning. (See page 6-11, *Trace/Parse bitstream debugging*.)
2. Set the Trace filename and the maximum Trace File size. (See page 5-1, *How to use the MTS4EA*.)
3. Set Parse Bitstream on and all other Trace functions off.
4. Turn Summary Tooltip on (icon) and play the video from the start; you can stop the video as soon as it has passed the frame numbers to be captured.
5. Use View trace to see the Trace file output.
6. You should see a Trace out file similar to that shown in the see *How to use the MTS4EA* section. (See page 5-1.)
7. Search for Errors and Warnings (using the Find next button in the View trace window).
8. If there are no Errors or Warnings, it is a higher-level error. (See page 6-14, *Trace/Interpret debugging*.)

9. If there are Errors/Warnings, examine the bitstream data and the mnemonic given on the right side of the trace file and relate this to the video standard you are using to see if you can discover the problem.

NOTE. *It is very important to understand that the error may well have occurred at some prior point in the bitstream. It could be that the error/warning reported is the ripple effect result of some earlier incorrect value. This is usually the case with an Out of Sync error.*

For example:

In MPEG-4, if using video_plane_with_short_header() (MPEG-4 standard section 6.2.5.2), and pei = 1, then following this should be eight bits of a psupp field (these can be repeated), followed by the gob_layer(), gob_resync_marker (17 bits, 0000 0000 0000 0000 1)

However, if pei = 0, then the gob_resync_marker should follow directly.

Therefore if pei was set to 0 by accident, but there was a psupp field inserted set to all zeroes, the bit pattern following pei would be: 0000 0000 0000 0000 0000 0000 1. This would appear to be an error in the gob_resync_marker (that is because the 17th bit after pei=0 would not be set to 1).

10. Check that there are no elements of a standard being used that should not be there. For example, if you are compressing data to H.263 baseline or MPEG-4 short_header, search for: {263+} and {263 Ann} to find any H.263+ annexes used.

11. Check the three-letter codes in the square brackets used by MTS4EA and reported in the trace output, such as [SC], as given in *Three-letter codes for H.264/AVC*, and that the three-letter code corresponds with the relevant standard section. (See page 6-4.)

Parse bitstream example outputs. See also Parse Bitstream in the *How to use the MTS4EA* section for more information.

(See page 5-1.)

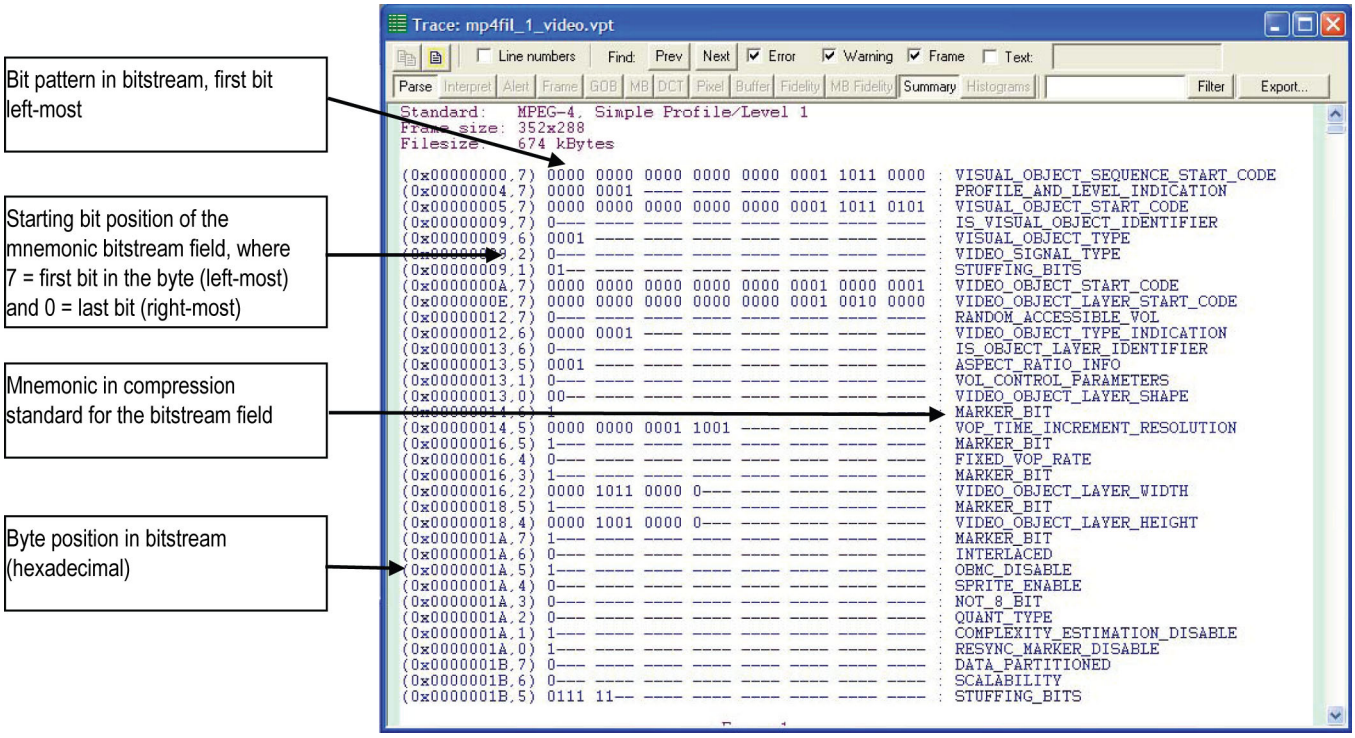


Figure 6-1: MPEG-4 example, at start of bitstream

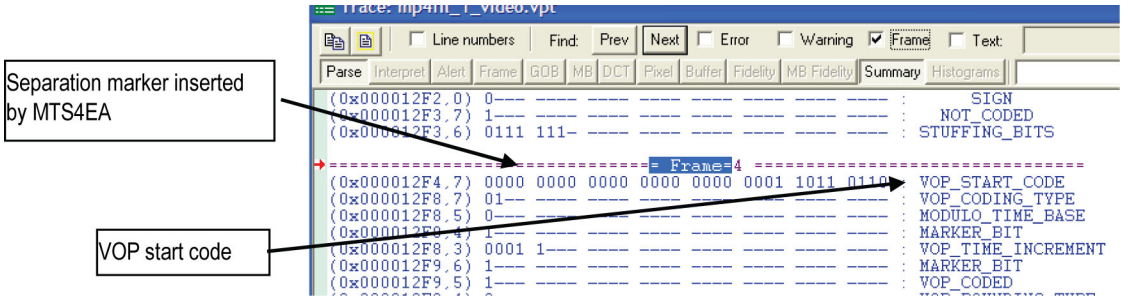


Figure 6-2: MPEG-4 example, at start of VOP

Trace/Interpret debugging

Overview. This is the highest level of debugging; MTS4EA interprets the bitstream and tries to tie it up with allowable values in the standards.

There are many occasions on which Trace/Parse Bitstream will not give an error/warning but Trace Interpret will. For example:

- Parse Bitstream will find a bitstream field and check that it is in the permissible range of values
- But Interpret will ensure that the value given is consistent with the rest of the data in the bitstream

There are times when it is useful to use both.

Procedure. The procedure for Trace/Interpret debugging is very similar to that for Trace/Parse Bitstream:

1. If you know the frame number at which an Error or Warning occurs, then set the Frame Range to one frame before and one frame after the location of the Error or Warning. (See page 6-11, *Trace/Parse bitstream debugging*.)
2. Set the Trace File name and the maximum Trace File size. (See page 5-1, *How to use the MTS4EA*.)
3. Set Interpret on and all other Trace functions off.
4. Turn the Summary Tooltip on and play the video from the start; you can stop the video as soon as it has passed the frame numbers to be captured.
5. Use View trace to see the Trace file output.
6. You should see a Trace out file similar to that shown in the see How to Use the MTS4EA section.
7. Search for Errors and Warnings (using the Find next button in the View trace window).

8. If there are Errors/Warnings/Out of Syncs, then examine the bitstream data and the mnemonic given on the right side of the trace file and relate this to the video standard you are using to see if you can discover the problem.

NOTE. *It is very important to understand that the error may have occurred at some prior point in the bitstream. It could be that the error/warning reported is the ripple effect result of some earlier incorrect value. This is usually be the case with an Out of Sync error.*

For example:

In MPEG-4, if using video_plane_with_short_header() (MPEG-4 standard section 6.2.5.2), and pei = 1 then following this should be 8 bits of a psupp field (these can be repeated), followed by the gob_layer(), gob_resync_marker (17 bits, 0000 0000 0000 0000 1).

However, if pei = 0 then the gob_resync_marker should follow directly.

Therefore if pei was set to 0 by accident, but there was a psupp field inserted set to all zeroes, the bit pattern following pei would be: 0000 0000 0000 0000 0000 0000 1. This would appear to be an error in the gob_resync_marker (that is because the 17th bit after pei=0 would not be set to 1)

9. Check that there are no elements of a standard being used that should not be there. For example, if you are compressing data to H.263 baseline or MPEG-4 short_header, search for: {263+} and {263 Ann} to find any H.263+ annexes used.

10. Check the three-letter codes in the square brackets used by the MTS4EA and reported in the trace output, such as [SC], as given in *Three-letter codes for H.264/AVC*, and that the three-letter code corresponds with the relevant standard section. (See page 6-4.)

Interpret example outputs. See also *Interpret* in the *How to Use the MTS4EA* section for more information. (See page 5-1.)

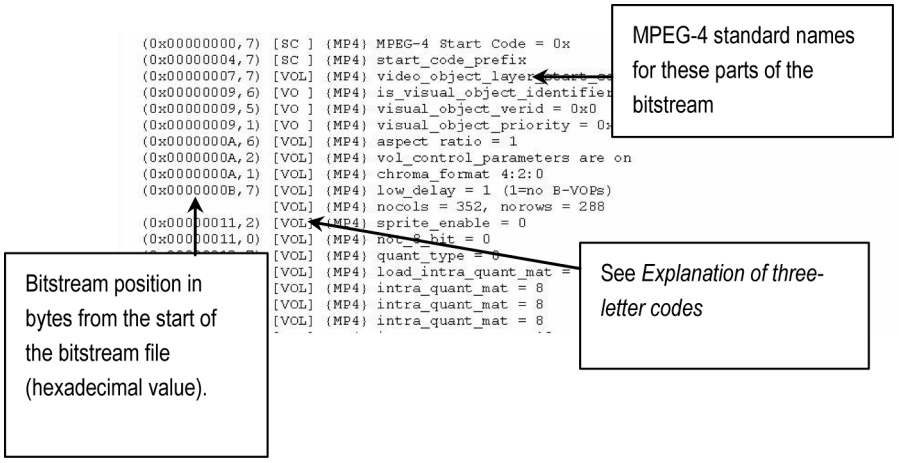


Figure 6-3: MPEG-4 Example, at start of bitstream

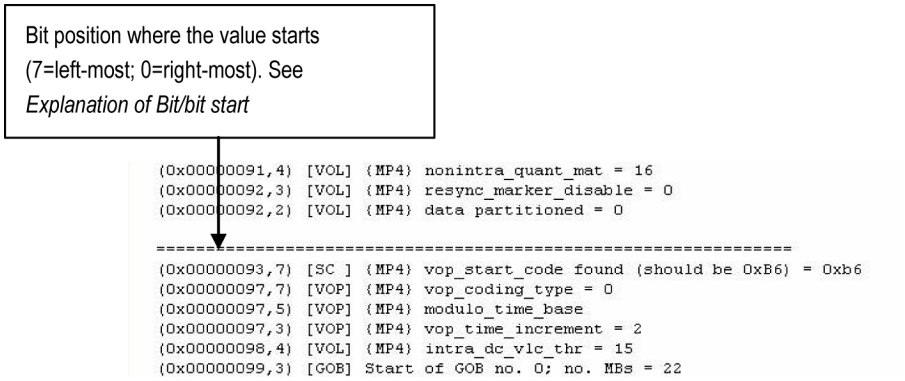


Figure 6-4: MPEG-4 example, at start of VOP

```
Standard:   H.264/AVC, Extended profile/Level 2.1
Frame size: 352x288
Filesize:   844349 kBytes

------------------------------ Frame=1 ------------------------------
(0x00000000,7) [BSN]     zero_byte = 0x00
(0x00000001,7) [BSN]     start_code_prefix_one_3bytes = 0x000001
(0x00000004,7) [NAL]     forbidden_zero_bit = 0
(0x00000004,6) [NAL]     nal_ref_idc = 3 : Reference slice or SPS or PPS
(0x00000004,4) [NAL]     nal_unit_type = 7 : Sequence Parameter Set (SPS)
(0x00000005,7) [SPS]     profile_idc = 88 : Extended profile
(0x00000006,7) [SPS]     constraint_set0_flag = 0 : May or may not obey A.2.1 constra
(0x00000006,6) [SPS]     constraint_set1_flag = 0 : May or may not obey A.2.2 constra
(0x00000006,5) [SPS]     constraint_set2_flag = 0 : May or may not obey A.2.3 constra
(0x00000006,4) [SPS]     reserved_zero_5bits = '00000'
(0x00000007,7) [SPS]     level_idc = 21 : Level 2.1
(0x00000008,7) [SPS]     seq_parameter_set_id = 0 (bitstream values: length=1 bits, se
(0x00000008,6) [SPS]     log2_max_frame_num_minus4 = 4 : MaxFrameNum = 256 (bitstream
(0x00000008,1) [SPS]     pic_order_cnt_type = 0
(0x00000008,0) [SPS]     log2_max_pic_order_cnt_lsb_minus4 = 4 : MaxPicOrderCntLsb = 2
(0x00000009,3) [SPS]     num_ref_frames = 1 (bitstream values: length=3 bits, num_ref_
(0x00000009,0) [SPS]     gaps_in_frame_num_value_allowed_flag = 0
(0x0000000A,7) [SPS]     pic_width_in_mbs_minus1 = 21 : PicWidthInMbs = 22; PicWidthIn
(0x0000000B,6) [SPS]     pic_height_in_map_units_minus1 = 8 : PicHeightInMapUnits = 9;
(0x0000000C,7) [SPS]     frame_mbs_only_flag = 0 : Pictures may be coded frames or coo
(0x0000000C,6) [SPS]     mb_adaptive_frame_field_flag = 0 : No switching between frame
(0x0000000C,5) [SPS]     direct_8x8_inference_flag = 0
(0x0000000C,4) [SPS]     frame_cropping_flag = 0 : Frame cropping offset parameters ar
(0x0000000C,3) [SPS]     vui_parameters_present_flag = 0 : Annex E VUI parameters are
(0x0000000C,2) [SPS]     rbsp_trailing_bits = '100'
(0x0000000D,7) [BSN]     zero_byte = 0x00
```

Figure 6-5: H.264/AVC example, at start of byte stream

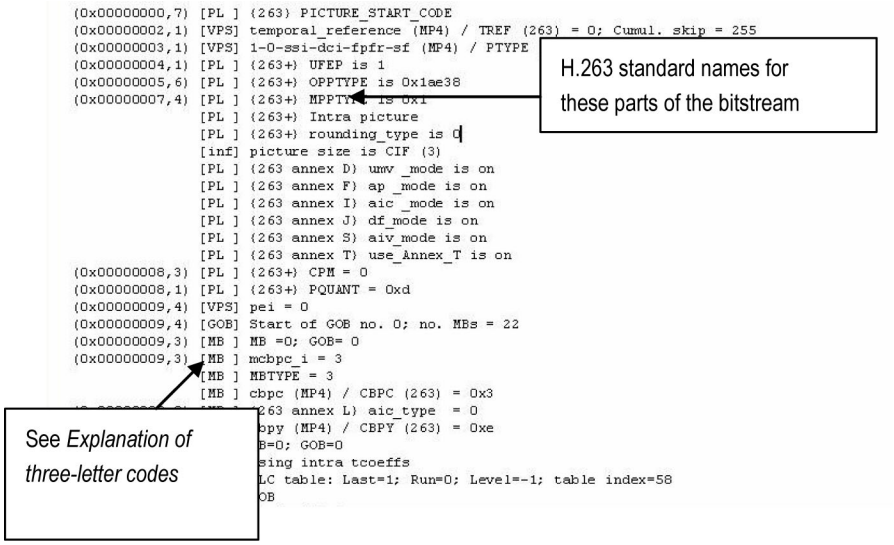


Figure 6-6: H.263 example at start of bitstream

Appendices

Appendix A: Decoder plugins for the MTS4EA

This section covers use of MTS4EA Decoder Plugins.

Purpose of the MTS4EA Decoder Plugins

The MTS4EA Decoder Plugins allow the advanced user to substitute various elements of MTS4EA decoder with custom/alternate elements.

There are several reasons why you may want to do this:

- There are various divergences between the outputs of different decoders due to rounding errors or different floating-point arithmetic used, because the video standards do not completely define these. The differences given mean that the output given by the MTS4EA - particularly using the YUV output in batch mode (see *Command line/batch mode* in the *How to use the MTS4EA* section for more information) - cannot be exactly compared with the output from another decoder. By being able to substitute a user's own part of the codec for that section of the decoder, the YUV outputs generated should exactly match those of the user. (See page A-4, *idct Plugin*.)
- Sometimes the video standard has errors and differs from common implementations. Many implementations differ from the standard in this way. (See page A-4, *qs Plugin*.)
- Sometimes it is useful to be able to use the MTS4EA function within your own decoder, to guarantee that numerically identical results are output by the MTS4EA decoder and your own.

NOTE. *The DLLs provided by Tektronix are provided and licensed solely for the purposes of test and may NOT be redistributed under ANY circumstances.*

Support of Decoder Plugins

Use of the Decoder Plugins assumes a considerable and detailed understanding of the video standard(s) concerned, and a thorough understanding of how to write Windows DLLs, and are provided for use by advanced developers only.

NOTE. *Other than the documentation provided here and in the files provided on the CD in the folder of the Decoder Plugins, no technical support will be provided to a developer in developing their own Decoder Plugin.*

Decoder Plugin file layout on the CD

All the Decoder Plugins are on the CD in the folder: `\Decoder plugins`. Below this are three folders:

`\bin`: contains the DLLs

`\docs`: contains the documentation

`\include`: contains the .h files to include in your programs

bin folder

Below this folder are the DLLs provided by the MTS4EA. The DLLs are organized in their own folders.

Where a DLL is not dependent upon a particular video standard, this folder resides directly below the bin folder. Where there are standards-dependencies, then the DLL folders reside below the folder of the standard concerned.

Examples are:

Folder	Files	Description

`\Decoder plugins\bin\idct\`	`tekIdct.dll`
	`tekIdct.exp`
	`tekIdct.lib`
		IDCT DLL file } export LIB files to link to } the IDCT DLL

`\Decoder plugins\bin\mpeg4\gmc`	`tekGmc.dll`
	`tekGmc.exp`
	`tekGmc.lib`
		GMC (Global Motion Compensation) DLL file } export LIB files to link to } the GMC DLL

`\Decoder plugins\bin\mpeg4\qs`	`tekQs.dll`
	`tekQs.exp`
	`tekQs.lib`
		Quarter Sample (QS) DLL file } export LIB files to link to } the QS DLL

docs folder

This folder contains the documentation for each supplied DLL. To view the documentation, open the file: `index.html` using a browser.

The API to the DLLs, file lists, structures, data fields and globals are all accessed by clicking on the related links.

include folder

This folder contains all the `.h` files that contain the declarations of the interface for each plugin.

For each plugin, only one include is required: that is for the related DLL For example, for the IDCT plugin, only the `tekIdct.h` file needs to be included in your own source files.

NOTE. *Any other .h files that are listed in the include folder, and which are also needed by the included `.h` file, will be included within the `.h` file.*

An example of this is the 'tekIdct.h', which includes 'MTS4EA.h'.

Format for generating Decoder Plugins

The Decoder Plugins were written using Microsoft Visual C 6.0.

The Plugins are all Microsoft Windows DLLs.

The only versions of Windows supported are given in the *Software, hardware, and user prerequisites* section.

NOTE. *No other development tools or variants to these are supported for the Decoder Plugins.*

Use of Decoder Plugins

To use a specific Plugin, copy it from the plugin folder to the System folder located where the MTS4EA executable is (typically `C:\Program files\Tektronix\MTS4CC\System`).

There will already be a DLL of the same name within the MTS4EA executable folder; the copied Plugin should overwrite this.

As delivered, all the Plugins that are in the MTS4EA executable folder are MTS4EA variants of these.

Decoder Plugins provided

idct Plugin

This allows you to substitute your own IDCT function instead of using the one provided with the MTS4EA. By using the same IDCT function in the MTS4EA and in your own decoder, the YUV outputs from the MTS4EA should be numerically identical to the YUV outputs provided by your own decoder.

qs Plugin

The MPEG-4 standard (Reference [1], see *Compression standards and file types* section) specifies the recommended method for calculating quarter sample accuracy predicted blocks (relating to motion vectors). However, the standard is non optimal in the way that these are calculated (for example introducing more rounding errors than necessary) and is not as clear as it might be. Consequently, common implementations of the quarter sample functions implement the calculations slightly differently (typically in a different order) to that given in the standard.

The Microsoft reference decoder (Reference [6], see *Compression standards and file types* section), although it is (theoretically) normative, is one such implementation that does not actually implement the quarter sample calculations in accordance with the standard; it takes a better approach.

(See page 4-1, *Compression standards and file types*.)

There has been some debate about this within the MPEG committee, and the prevailing conclusion is that the Microsoft implementation may be the preferred version and that the standard should at some point be changed to reflect this.

The MTS4EA uses the version as specified in the standard, and the MTS4EA will continue to track the standard and so will change only if the standard is changed. You can substitute your own quarter sample Plugin DLL, to use your own or emulate the one used by Microsoft.

gmc Plugin

This allows you to substitute your own GMC (Global Motion Compensation) function instead of using the one provided with the MTS4EA.

The Microsoft reference decoder software (Reference [6], see *Compression standards and file types* section) differs from the MPEG-4 standard (Reference [1], see *Compression standards and file types* section) in several respects, for example, rounding differences and differences in coordinate calculations. This means that the results of the YUV outputs from a Microsoft decoder will be different than the standard if GMC is used. (These differences are reflected in the Microsoft encoder, so that the Microsoft software is self-consistent.) There is no indication from the MPEG-4 committee of any planned change in the standard. This is simply a variance between the standard and the reference software.

(See page 4-1, *Compression standards and file types*.)

MTS4EA follows the standard. If you want to follow the Microsoft version of the implementation of GMC or use your own, you can substitute your own GMC DLL.

Appendix B: Tests of the MTS4EA

This section covers a report on the MTS4EA tests of the MPEG-4 Normative bitstreams and Donated bitstreams. (See page 4-1, *Compression standards and file types*.)

Tests with MPEG-4 Normative and Donated bitstreams

Many of the MPEG-4 Normative bitstreams and Donated bitstreams have errors: some of these errors are clear non-conformance to the standard; others are errors in the data encoded in the bitstreams.

Also, in a number of cases, the Microsoft reference decoder software will not decode these bitstreams at all or does so incorrectly.

These divergences are listed below. Many of them are recognized and documented by the MPEG committee as errors, but they have yet to be corrected in the Standard and/or bitstreams and/or Microsoft reference decoder software.

Notes on versions used:

- MTS4EA: v2.0.0.0
- Microsoft reference software: FDAM1-2.3-001213 version 2 dated July 3rd 2000 (note: this reference software has been updated since these tests, so the current reference software may behave differently)
- MPEG-4 Normative ISO bitstreams: dated 05/11/2001, see *Compression standards and file types* section
- MPEG-4 Donated bitstreams: referred to in section 4.5.8 in document N3067 dated 1999-03-18; streams dated 14/07/2000, see *Compression standards and file types* section

Bitstreams: Normative ISO

n/a = not applicable

Table B-1: Bitstreams: Normative ISO

Stream name	Decodes with ?	Notes
vcon-ge1	n/a	Interlace not allowed in Simple Profile
vcon-ge2	n/a	Interlace not allowed in Simple Profile
vcon-ge3	n/a	Interlace not allowed in Simple Profile
vcon-ge4	n/a	Interlace not allowed in Simple Profile
vcon-ge6	n/a	Interlace not allowed in Simple Profile
vcon-ge8	n/a	OBMC not allowed in Simple, Advanced Simple, or Main Profiles
vcon-ge10	n/a	Interlace not allowed in Simple Profile
vcon-ge11	n/a	Interlace not allowed in Simple Profile
vcon-ge12	n/a	Interlace not allowed in Simple Profile
vcon-ge13-L1	Yes	
vcon-ge13-L2	Yes	
vcon-ge13-L3	Yes	
vcon-ge14	n/a	OBMC not allowed in Simple, Advanced Simple, or Main Profiles
vcon-ge16-L1	Yes	MS reference software cannot decode
vcon-ge16-L2	Yes	MS reference software cannot decode
vcon-ge16-L3	Yes	MS reference software cannot decode
vcon-ge18	n/a	Interlace not allowed in Simple Profile
vcon-ge19	n/a	OBMC not allowed in Simple, Advanced Simple, or Main Profiles
vcon-ge23	n/a	Interlace not allowed in Simple Profile
vcon-ge24	n/a	OBMC not allowed in Simple, Advanced Simple, or Main Profiles
vcon-ge25	n/a	OBMC not allowed in Simple, Advanced Simple, or Main Profiles

Bitstreams: Donated \ I-VOP

Table B-2: Bitstreams: Donated \ I-VOP

Stream name	Decodes with ?	Notes
hit000.m4v	Yes	Single frame. Time listed incorrectly in MPEG-4 part 4: time is actually 33 ms (vop_time_increment = 30).
jvc000.m4v	Yes	
mit000.m4v	Yes	Uses error resilience tool (data partitioning)
mit001.m4v	Yes	Uses error resilience tool (resynchronization)
mit002.m4v	Yes	Uses error resilience tool (resynchronization)
mit003.m4v	Yes	Uses error resilience tool (resynchronization)
mit004.m4v	Yes	Uses error resilience tool (data partitioning)
mit005.m4v	Yes	Uses error resilience tool (data partitioning)
mit006.m4v	Yes	Uses error resilience tool (data partitioning)
san000.m4v	Yes	
san001.m4v	Yes	

Bitstreams: Donated \ Overall

Table B-3: Bitstreams: Donated \ Overall

Stream name	Decodes with ?	Notes
hit016.m4v	Yes	
hit017.m4v	Yes	Visible artifacts are in bitstream (plays the same with MS software)
hit018.m4v	Yes	
hit019.m4v	Yes	
hit020.m4v	Yes	
hit021.m4v	Yes	
hit022.m4v	Yes	
hit023.m4v	Yes	
hit024.m4v	Yes	
mit030.m4v	Yes	
mit031.m4v	Yes	Uses error resilience tool (data partitioning)

Bitstreams: Donated \ Short Header

Table B-4: Bitstreams: Donated \ Short Header

Stream name	Decodes with ?	Notes
hit031.m4v	Yes	
hit032.m4v	Yes	Visible errors but these are encoded in the bitstream (MS software plays the same)
hit033.m4v	Yes	
hit034.m4v	Yes	
hit035.m4v	Yes	MPEG committee reports that MS software fails to decode (not tested)
hit036.m4v	Yes	Not short_header compliant as Pspare is sent (H.263+ compliant)
hit037.m4v	Yes	
hit038.m4v	Yes	
hit039.m4v	Yes	
hit040.m4v	Yes	
jvc022.m4v	Yes	
jvc023.m4v	Yes	
jvc024.m4v	Yes	
jvc025.m4v	Yes	
mit020.m4v	Yes	
mit021.m4v	Yes	MPEG committee reports that MS software fails to decode (not tested)
mit022.m4v	Yes	MPEG committee reports that MS software fails to decode (not tested)
mit023.m4v	Yes	MPEG committee reports that MS software fails to decode (not tested)
mit024.m4v	Yes	MPEG committee reports that MS software fails to decode (not tested)
san021.m4v	Yes	
san022.m4v	Yes	Poor frames are encoded in bitstream (MS software plays the same)
san023.m4v	Yes	Non-smooth motion is encoded in bitstream (MS software plays the same)
san024.m4v	Yes	Non-smooth motion is encoded in bitstream (MS software plays the same)

Bitstreams: Donated \ P-VOP

Table B-5: Bitstreams: Donated \ P-VOP

Stream name	Decodes with ?	Notes
hit001.m4v	Yes	Some visible errors, for example, frame 1, GOB 2, MB 15 block Y3 but these are in bitstream (MS software plays the same)
hit002.m4v	Yes	Some visible errors, for example, in frame 2, GOB4, MB 1 and in frame 3 - several MBs down left side, for example, MB 0 at GOB 14, 15, 16 - these are errors in the encoded bitstream (MS software plays the same)
hit003.m4v	Yes	Errors in bitstream, for example, in MB 0 GOB 6 frame 4. It seems that MVs were not correctly calculated in encoded stream. (MS software plays the same)
hit004.m4v	Yes	
hit005.m4v	Yes	Some visible artifacts in bitstream (MS software plays the same)
hit006.m4v	Yes	
hit007.m4v	Yes	
hit008.m4v	Yes	
hit009.m4v	Yes	
hit010.m4v	Yes	
hit011.m4v	Yes	
hit012.m4v	Yes	
hit013.m4v	Yes	
hit014.m4v	Yes	
jvc001.m4v	Yes	
jvc002.m4v	Yes	
jvc003.m4v	Yes	
jvc004.m4v	Yes	
jvc005.m4v	Yes	
jvc006.m4v	Yes	
jvc007.m4v	Yes	
jvc008.m4v	Yes	
jvc009.m4v	Yes	
jvc010.m4v	Yes	
jvc011.m4v	Yes	
jvc012.m4v	Yes	
jvc013.m4v	Yes	
jvc014.m4v	Yes	Non-smooth motion is encoded in bitstream (MS software plays the same)
jvc015.m4v	Yes	
jvc016.m4v	Yes	
jvc017.m4v	Yes	
jvc018.m4v	Yes	

Table B-5: Bitstreams: Donated \ P-VOP (cont.)

Stream name	Decodes with ?	Notes
jvc019.m4v	Yes	
jvc020.m4v	Yes	
jvc021.m4v	Yes	
mit007.m4v	Yes	Syntax error in bitstream: video_object_type_indicator is 0 at byte: 8. bit start=0. This is defined as a RESERVED value (MPEG-4 std. p.110 Table 6-10). Plays this stream, assuming it is Simple Profile (an Error message is given to this effect)
mit008.m4v	Yes	
mit009.m4v	Yes	Visible artifacts are encoded in bitstream (MS software plays the same)
mit010.m4v	Yes	Uses error resilience tool (data partitioning)
mit011.m4v	Yes	Visible artifacts are in bitstream, for example, frame 4 halo above head in GOB 0 MBs 3,4,5 and poor MBs with hand movement, for example, frame 4, GOB,MB: 10,8 11,8 (MS software plays the same)
mit012.m4v	Yes	Uses error resilience tool (data partitioning)
mit013.m4v	Yes	Uses error resilience tool (data partitioning). Stream not decoded correctly by MS software - occasional green lines in some MBs
mit014.m4v	Yes	Uses error resilience tool (data partitioning)
mit015.m4v	Yes	Uses error resilience tool (data partitioning)
mit016.m4v	Yes	Uses error resilience tool (data partitioning)
mit017.m4v	Yes	Uses error resilience tool (data partitioning)
mit018.m4v	Yes	Uses error resilience tool (data partitioning)
mit019.m4v	Yes	
san002.m4v	Yes	
san003.m4v	Yes	
san004.m4v	Yes	
san005.m4v	Yes	Non-smooth motion is encoded in bitstream (MS software plays the same)
san006.m4v	Yes	Non-smooth motion is encoded in bitstream (MS software plays the same)
san007.m4v	Yes	Non-smooth motion is encoded in bitstream (MS software plays the same)
san008.m4v	Yes	Non-smooth motion is encoded in bitstream (MS software plays the same)
san009.m4v	Yes	Non-smooth motion is encoded in bitstream (MS software plays the same)
san010.m4v	Yes	Non-smooth motion is encoded in bitstream (MS software plays the same)
san011.m4v	Yes	Non-smooth motion is encoded in bitstream (MS software plays the same)
san012.m4v	Yes	Non-smooth motion is encoded in bitstream (MS software plays the same)
san013.m4v	Yes	
san014.m4v	Yes	
san015.m4v	Yes	Graininess, visible artifacts, and intermediate lower resolution frames are encoded in the bitstream (MS software plays the same)
san016.m4v	Yes	Visible artifacts and intermediate lower resolution frames are encoded in the bitstream (MS software plays the same)

Table B-5: Bitstreams: Donated \ P-VOP (cont.)

Stream name	Decodes with ?	Notes
san017.m4v	Yes	
san018.m4v	Yes	
san019.m4v	Yes	
san020.m4v	Yes	

Bitstreams: Donated \ Error

Table B-6: Bitstreams: Donated \ Error

Stream name	Decodes with ?	Notes
hit025.m4v	Yes	Use of resync markers
hit026.m4v	Yes	Use of resync markers
hit027.m4v	Yes	Uses error resilience tool (data partitioning)
hit028.m4v	Yes	Uses error resilience tool (data partitioning)
hit029.m4v	Yes	Uses error resilience tool (data partitioning)
hit030.m4v	Yes	Uses error resilience tool (data partitioning)
mit025.m4v	Yes	Use of resync markers
mit026.m4v	Yes	Use of resync markers
mit027.m4v	Yes	Uses error resilience tool (data partitioning) Poor quality picture (MS software plays the same)
mit028.m4v	Yes	Uses error resilience tool (data partitioning)
mit029.m4v	Yes	Uses error resilience tool (data partitioning). Artifacts in frames 5 and 6 are errors in the encoded bitstream: seems to be motion vectors incorrectly calculated when the stream was encoded (MS software plays the same)

Appendix C: Third party software licenses

BSD license

The copyright in this software is being made available under the BSD License, included below. This software may be subject to other third party and contributor rights, including patent rights, and no such rights are granted under this license.

Copyright (c) 2010-2013, ITU/ISO/IEC
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- Neither the name of the ITU/ISO/IEC nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Glossary

Glossary

AC coefficient

Any DCT coefficient for which the frequency in one or both dimensions is non-zero

Annex

The optional methods of coding allowed in H263+

B-VOP

A VOP that is coded using motion compensated prediction from past and/or future reference VOPs

Backward motion vector

A motion vector that is used for motion compensation from a reference VOP at a later time in display order

Backward prediction

Prediction from the future reference VOP

Bitstream

A compressed data entity where each binary digit has a specific meaning that is defined by the compression standard

Block

An 8-row by 8-column matrix of samples, or 64 DCT coefficients (source, quantized or dequantized)

Buffer analysis

Analysis of use of processor, memory and/or other resources by a particular video decoder standard

Coded Block Pattern (cbp)

A word used in compression to identify which blocks of a MacroBlock are coded

Coefficients

The Discrete Cosine Transform transforms data from a spatial domain (pixels or pixel differences) into the frequency domain, because it makes the data easier to compress. In the standards used here, transforms turn a block of 8x8 pixel data (or pixel differences) into a block of 8x8 transform coefficients

CSV file

A file format which contains data separated by commas (Comma Separated Variable)

DC coefficient

The DCT coefficient for which the frequency is zero in both directions

DCT

Discrete Cosine Transform - the mathematical transform that all these compression standards use as their basis

DCT coefficient

Amplitude of the specific DCT basis function

Dequantization

Process of rescaling the quantized DCT coefficients after their representation in the bitstream has been decoded and before they are presented to the inverse DCT

Filter

A mathematical transform designed to remove certain frequencies from a signal. Here mainly used either within the coding loop to try and avoid some of the coding artifacts and reduce bit-rate, or used as a post-process to improve the subjective quality

Frame

An individual picture from a video sequence

Forward motion vector

A motion vector that is used for motion compensation from a reference frame VOP at an earlier time in the display order

Forward prediction

Prediction from a past reference VOP

Global Motion Compensation

Use of global spatial transformation to improve the efficiency of the prediction of sample values by providing offsets into the past reference VOPs containing previously decoded sample values that are used to form the prediction error

GMC

Global Motion Compensation

GOB

Group Of Blocks - an entity defined within some of the standards in order to subdivide a frame into more manageable units

Histogram

A graph of the frequency of occurrence of a variable

I-VOP, intra-coded VOP

A VOP coded using information only from itself

Intra coding

Coding of a MacroBlock or VOP using information only from that MacroBlock or VOP

MacroBlock, MB

Basic coding unit of the standards used in this program. It consists of four blocks of 8x8 luminance data (arranged in a 16x16 manner) together with the two chrominance components U and V, which are also 8x8 blocks, but which cover the same area of the picture as the 16x16 luminance pixels - part of the compression is that chrominance can be sampled at a lower frequency than luminance

MacroBlock Type

The mode, according to the Standard, in which the MacroBlock is encoded

Mode

Classification of the coding type of the MacroBlock

Motion Vector

Two-dimensional vector that points from the current MacroBlock to an area in the previous frame that is used to predict the current data

Motion compensation

Use of motion vectors to improve the efficiency of the prediction of sample values, where the motion vectors provide offsets into the past and/or future reference VOPs containing previously decoded sample values that are used to form the prediction error

Motion estimation

Process of estimating motion vectors during the encoding process

Parse Bitstream

The process of parsing a bitstream into the constituent words that are allowed within the standard

Quantizer

The discrete value that is used to reduce the amount of information present in the DCT of a block. It can vary from 1 to 31 in most standards, where 1 is the finest level (most accurate coding) and 31 is the coarsest level (least accurate coding)

Quantization matrix

Set of sixty-four8-bit values used by the dequantizer

Quantized DCT coefficients

DCT coefficients before Dequantization, represented in variable-length coded form in the bitstream

Quantizer scale

Scale factor coded form in the bitstream and used by the decoding process to scale the dequantization

Slice

A subdivision of a picture that is used as a unit of encoding, as used in H.263 and MPEG-2

SEI

Supplemental enhancement information

VOP

Effectively a frame of video (MPEG-4)

Index

Index

Symbols and Numbers

3-letter codes
  H.263, 6-6
  H.264/AVC, 6-4
  H.265/HEVC, 6-2
  MPEG-2, 6-6
  MPEG-4, 6-6
  used in Trace debugging, 6-1
3GPP files, 4-24
  close video after opening, 5-30
  example files, 5-38
  extract video, 5-23
  opening video files, 5-18
  save video after opening, 5-30
  tracks within, 5-18

A

A/V Delay
  enable, 5-254
  view, 5-254
A/V Delay stream
  example files, 5-39
ADTS streams
  example files, 5-39
Alert log
  address, 5-275
  alert filter, 5-276
  details, 5-275
  display frame number, 5-275
  filter icon, 5-274
  highlighting in step-back buffer, 5-278
  ID number, 5-19, 5-274
  level, 5-274
  right-click pop-up menu, 5-279
  seeing details of alerts, 5-277
  showing/hiding, 5-276
  title, 5-275
  view, 5-274
Alerts
  3-letter codes, 6-1
  abort, pause, continue, 5-268
  alert log, 5-274
  configuration alert window, 5-273
  debugging, 6-11
  description of levels, 5-270
  during seek forwards/backwards, 5-267
  enable/disable specific, 5-272
  error, 5-271
  fatal, 5-270
  goto, 5-269
  info, 5-271
  levels, 5-262
  one/more disabled (status bar), 5-280
  skip this/skip all, 5-267
  summary section, 5-264
  synchronized views, 5-269
  top section, 5-263
  warning, 5-271
Alt menu keys
  activating, 5-300
Analysis
  PSNR, 5-189
Analysis options
  frame range, Glossary-1
ASF files
  close video after opening, 5-30
  example files, 5-39
  extract video, 5-23
  format, 4-16
  opening video files, 5-18
  save video after opening, 5-30
  tracks within, 5-18
Audio
  Audio waveform view toolbar icons, 5-140
  audio-video synchronization, 5-44
  broken audio, 5-43
  file formats, 4-23
  standards references, 4-24
Audio files
  opening elementary streams, 5-17
Audio playout prerequisites, 2-1

B

Batch mode
  introduction, 5-301
  options, 5-305
  use of example sequences, 5-303
  use of YUV example sequences, 5-303
Buffer analysis
  alerts, 5-218
  controls, 5-220
  H.264/AVC, 5-221
  H.264/SVC, 5-221
  limits, 5-217
  measuring values, 5-215
  MPEG-2, 5-218
  MPEG-4, 5-218
  MPEG-4 & MPEG-2, 5-219
  scroll bar area, 5-218
  toolbar icons, 5-214
  X (horizontal) scale, 5-217
  Y (vertical) scale, 5-217

C

Cache/tsee Step-back buffer, 5-266
Colors
  CTU overlays, 5-117
  MacroBlock overlays, 5-106

Command line mode
example, 5-310
introduction, 5-301
options, 5-305
Computer prerequisites, 2-1
Ctrl keys
Ctrl+Tab to switch windows, 5-281
list, 5-299
CTU
average bits per coded, 5-125
average bits per over frame range, 5-125
average fidelity, 5-126
average quantiser, 5-125
bits used for, 5-120
fidelity, 5-122
frequency of coding, 5-126
frequency of Intra coding, 5-126
graph, bits per CTU, 5-176, 5-178
graph, coded frequency, 5-183
graph, Intra CTU coded frequency, 5-185
quantizer used, 5-120
statistics, 5-115
colors, 5-117
numerals, 5-115

D

Debugging
3-letter codes, 6-1
alert pop-up, 5-263
alerts, 6-11
bit/bit start, explanation of, 6-8
how THE MTS4EA checks syntax, 6-9
main tools, 6-9
Parse bitstream, 5-236
tips, 6-10
Trace, interpret, 6-14
Trace, parse bitstream, 6-11
Decoder cache, 5-55
Decoder options
audio, 5-61
General, 5-54
H.263, 4-13, 5-55, 5-94
H.264/AVC, 4-2, 5-57, 5-92
H.264/SVC, 4-7, 5-93
H.265/HEVC, 4-2
MPEG-2, 5-56, 5-94
MPEG-4, 5-56, 5-94
step-back buffer, 5-54
uncompressed video, 5-57
video, 5-54
Decoder plug-ins
gmc, A-4
idct, A-4
purpose, A-1
qs, A-4
use of, A-3

E

Example file
H.264/SVC, 5-35
Example files, 5-33
3GPP, 5-38
A/V Delay stream, 5-39
ADTS streams, 5-39
ASF files, 5-39
fidelity analysis using, 5-39
H.263, 5-37
H.264/AVC, 5-35
H.265/HEVC, 5-34
MP4, 4-16, 5-37
MPEG-2 Program Streams, 5-38
MPEG-2 Transport Streams, 5-38
MPEG-4, 5-36
use in batch mode, 5-303

F

Fidelity
analysis using example files, 5-39
average per CTU, 5-126
average per MacroBlock, 5-115
frame Trace, 5-168
MacroBlock Trace, 5-169
MAD, 5-190
MAD (Mean Absolute Difference), 5-189
metric, 5-189
MSE, 5-190
MSE (Mean Square Error), 5-189
overlay, 5-111, 5-122
per CTU, 5-122
per MacroBlock, 5-111
PSNR (255 signal range), 5-190
PSNR (255), 5-189
PSNR (ITU-R BT.601), 5-189
reference file, 5-186
RMSE, 5-190
RMSE (Root Mean Square Error), 5-189
SAD, 5-190
SAD (Sum Absolute Difference), 5-189
Fidelity analysis
icons toolbar, 5-226
metrics available, 5-227
PSNR, 5-225
video enable, 5-168, 5-186
view, 5-225

File formats
3GPP container files, 4-17
3GPP files, 4-24
ASF container files, 4-16
audio, 4-23
MP4 container files, 4-16, 4-24, 5-37
MPEG-2
PES files, 4-18, 4-23
Program Stream files, 4-18, 4-23
Transport Stream files, 4-23
MPEG-2 PES, 4-18
MPEG-2 Program Stream, 4-18
MPEG-2 Transport Streams, 4-17
RCV files, 4-16
video, 4-16
Frame range
Trace, Glossary-1

G

GMC
decoder plug-in, A-4
Graph
average quantizer, 5-180
bits per coded MacroBlock, 5-177
bits per CTU, 5-176, 5-178
bits per MacroBlock, 5-175
CTU coding frequency, 5-183
Intra coded frequency, 5-184
Intra CTU coded frequency, 5-185
MacroBlock coding frequency, 5-182
motion vectors per MacroBlock, 5-179
video enable, 5-172

H

H.263
3-letter codes used in Trace, 6-6
decoder options, 4-13, 5-55, 5-94
example files, 5-37
MacroBlock types, 4-13, 5-55, 5-94
segments, 5-109
standards supported, 4-13, 5-55, 5-94
H.264/AVC
3-letter codes used in Trace, 6-4
Buffer analysis, 5-221
decoder options, 4-2, 5-57, 5-92
example files, 5-35
MacroBlock types, 4-2, 5-57, 5-92
slices, 5-110
video Byte Stream format, 4-20
H.264/SVC
Buffer analysis, 5-221
decoder options, 4-7, 5-93
example file, 5-35
MacroBlock types, 4-7, 5-93
H.265/HEVC
3-letter codes used in Trace, 6-2
decoder options, 4-2
example files, 5-34
MacroBlock types, 4-2
slices, 5-121
video Byte Stream format, 4-20
Help
license manager, 5-290
PDF tutorials, 5-290
PDF user manual, 5-290
HexView
bytes per line, 5-206
convert hex to decimal, 5-209, 5-251
find absolute address, 5-208
find binary, hex, ASCII, 5-208, 5-250
find next relative address, 5-208, 5-250
find next/previous, 5-208, 5-250
find previous relative address, 5-208, 5-250
find relative address, 5-208, 5-250
highlighting a section, 5-207
parts of window, 5-205, 5-247
right-click pop-up menu, 5-207, 5-249
settings in Window menu, 5-287
wildcard searching, 5-209, 5-251
Histograms
audio data bit usage, 5-141
frame bit usage, 5-142
frame data bit usage, 5-142
main payload bit usage, 5-143
subsidiary payload bit usage, 5-144
HRD buffer analysis
buckets, 5-222
display, 5-223
H.264/AVC, 5-221
H.264/SVC, 5-221
overflow/underflow, 5-224
schedule indexes, 5-222
Hypothetical Reference Decoder, 5-221

I

Icons
Audio waveform view, 5-140
buffer analysis toolbar, 5-214
function of toolbar icons, 5-292

Interlace
bottom field view, 5-296
combined (frame) view, 5-295
separated (fields) view, 5-295
toolbar, 5-294
top field view, 5-296
Interpret, 5-157, 5-237
right-click pop-up menu, 5-237

L

License
remote server requirements, 3-1

M

MacroBlock
average bits per coded, 5-114
average bits per over frame range, 5-114
average fidelity, 5-115
average quantiser, 5-114
bits used for, 5-109
fidelity, 5-111
fidelity Trace, 5-169
frequency of coding, 5-115
frequency of Intra coding, 5-115
graph, bits per coded MB, 5-177
graph, bits per MB, 5-175
graph, coded frequency, 5-182
graph, Intra coded frequency, 5-184
graph, motion vectors, 5-179
quantizer used, 5-108
statistics, 5-105, 5-284
colors, 5-106
frame range, Glossary-1
numerals, 5-105
types
color key, 5-296
H.263, 4-13, 5-55, 5-94
H.264/AVC, 4-2, 5-57, 5-92
H.264/SVC, 4-7, 5-93
H.265/HEVC, 4-2
MPEG-2, 5-56, 5-94
MPEG-4, 5-56, 5-94
VC-1, 5-93
MacroBlock tooltip
AC prediction, 5-81
examples for different video standards, 5-77
pixel location, 5-79
segment (H.263), 5-80
slice (MPEG-2), 5-80
Motion vectors
graph, 5-179
MP4 file format
close video after opening, 5-30
example files, 4-16, 5-37
extract video, 5-23
opening video files, 5-18
save video after opening, 5-30
tracks within, 5-18
MP4 files, 4-24
MPEG-2
3-letter codes used in Trace, 6-6
buffer analysis (VBV), 5-218
buffer analysis controls, 5-219
decoder options, 5-56, 5-94
MacroBlock types, 5-56, 5-94
opening VOBs, 5-25
PES, 4-18
slices, 5-110
Transport Stream file format, 4-17
MPEG-2 PES files, 4-18, 4-23
MPEG-2 Program Stream files, 4-18, 4-23
MPEG-2 Program Streams
close video after opening, 5-30
example files, 5-38
opening video files, 5-18
save video after opening, 5-30
tracks within, 5-18
MPEG-2 Transport Stream files, 4-23
MPEG-2 Transport Streams
example files, 5-38
extract video, 5-23
opening video files, 5-18
tracks within, 5-18

MPEG-4
3-letter codes used in Trace, 6-6
buffer analysis (VBV, VCV, VMV), 5-218
buffer analysis controls, 5-219
decoder options, 5-56, 5-94
example files, 5-36
MacroBlock types, 5-56, 5-94
MP4 container file format, 4-16, 5-37
video Elementary Stream format, 4-21
MTS4000 MPEG Test Systems, 2-2
MTS4EA
product features, 1-2
product models, 1-1
starting to use, 5-3
window elements, 5-2

O

Overlay
fidelity, 5-111, 5-122
making clearer with Blank video, 5-131

P

PAL/NTSC
analysis with THE MTS4EA, 4-22
Parse bitstream, 5-236
right-click pop-up menu, 5-156, 5-237
PES
MPEG-2, 4-18
Pixel data
Trace pixel level, 5-167
uncompressed video output, batch mode, 5-310
Play
audio decoder options, 5-61
continuous, 5-52
video decoder options, 5-54
Playing video
continuously in a lop, 5-52
keeping first frame on-screen, 5-52
keeping last frame on-screen, 5-52
mode restrictons, 5-7
preferences, 5-32
Prerequisites
audio playout, 2-1
computer, 2-1
user, 2-2
Windows operating system, 2-1
Product features, 1-2
Product models, 1-1
Program Stream
MPEG-2, 4-18
Projects
close, 5-32
default project files, 5-31
project files, 5-31
PSNR
fidelity analysis, 5-225
fidelity metric, 5-189

Q

QuarterSample
decoder plug-in, A-4

R

Reference file for fidelity analysis, 5-186
Right-click pop-up menu
alert log, 5-279
HexView, 5-207, 5-249
interpret, 5-237
parse bitstream, 5-156, 5-237

S

Scalability level selection, 5-257
Shortcut keys
Alt menu, 5-300
Ctrl, 5-299
Ctrl+Tab to switch windows, 5-281
icon toolbar, 5-292
Standards
MPEG-1 Part 3 audio, 4-14
MPEG-2 Part 3 audio, 4-15
MPEG-4 Part 3 audio, 4-15
references, 4-24
3GPP, 4-24
MP4, 4-24
MPEG-1 audio, 4-24
references, H.264/AVC, 4-24
references, H.264/SVC, 4-24
references, H.265/HEVC, 4-24
references, HE AAC codecs, 4-24
references, MPEG-2, 4-24
references, MPEG-2 AAC, 4-24
references, MPEG-2 audio, 4-24
references, MPEG-4, 4-24
references, MPEG-4 AAC, 4-24
references, VC-1, 4-24
supported
H.263, 4-13, 5-55, 5-94
H.264/AVC, 4-2, 5-57, 5-92
H.264/SVC, 4-7, 5-93
H.265/HEVC, 4-2
MPEG-2, 4-12
MPEG-4, 4-10
VC-1, 4-9
Status bar
overview, 5-284, 5-298
Step-back buffer
affect on pop-up alerts, 5-266
alert highlighting when seeking, 5-278
decoder options, 5-54

Synchronized views
alerts Goto button, 5-269
highlighting in audio view, 5-10
highlighting in video view, 5-9
introduction, 5-8
yelllow dotted box, 5-9

T

Toolbar
Audio waveform view, 5-140
buffer analysis icons, 5-214
disabled buttons, 5-291
Interlace, 5-294
list of functions, 5-292
moving & docking, 5-291
Tooltips
force un-docking, 5-65
MacroBlock types color key, 5-296
Trace
3-letter codes, 6-1
alerts, 5-160
audio data bit usage, 5-238
audio frame summary, 5-237
debugging, 6-11
export, 5-202
file format, 5-202
file size limit, 5-153, 5-236
find next, 5-201, 5-244
find previous, 5-201, 5-244
finding errors, warnings, data, 5-201, 5-244
finding Errors, Warnings, data, 5-201, 5-244
frame data bit usage, 5-239
frame fidelity, 5-168
frame range, Glossary-1
frame summary, 5-161
import, 5-202
interpret, 5-157, 5-237
line numbers on/off, 5-200, 5-244
lines too long (blue arrow displayed), 5-202, 5-245
MacroBlock fidelity, 5-169
main payload bit usage, 5-240
MB summary, 5-163
parse bitstream, 5-153, 5-236, Glossary-4
pixel level, 5-167
stream summary, 5-171
video enable, 5-147
Tutorials
PDF (Help menu), 5-290

U

Uncompressed video
decoder options, 5-57
opening video files, 5-25
show visual difference to encoded bitstream, 5-129
visual difference to encoded bitstream, 5-127, 5-284
User prerequisites, 2-2

V

VC-1
close video after opening, 5-30
MacroBlock types, 5-93
opening video files, 5-18
profiles & levels, 4-9
save video after opening, 5-30
tracks within, 5-18
Video
standards references, 4-24
trace enable, 5-147
video navigator, 5-195
Video files
example files, 5-33
opening
ASF files, 5-18
elementary streams, 5-17
greyscale format, 5-25
MP4 container files, 5-18
MPEG-2 Program Streams, 5-18
MPEG-2 Transport Streams, 5-18
MPEG-2 VOBs, 5-25
RGB format, 5-25
uncompressed video files, 5-25
VC-1 files, 5-18
YUV format, 5-25
Video navigator
detail view, 5-196
Video window
best fit, 5-285
fit to window, 5-285
zooming in, 5-285
View
alert log, 5-274
graphs, 5-203
video navigator, 5-195
Visual difference
difference magnifier, 5-130
icon toolbar, 5-128
show encoded bitstream, 5-128
VOB files
opening, 5-25

W

Window
quickly switch, 5-281
Window elements
status bar, 5-284, 5-298
Window menu
CTU statistics toolbar, 5-115
differences toolbar, 5-127, 5-284
interlace toolbar, 5-284
MB statistics toolbar, 5-105, 5-284
video scale, 5-285
video views toolbar, 5-284
Windows operating system
prerequisites, 2-1
WMV9 files
extract video, 5-23
Profiles & Levels, 4-9

Y

Yellow dotted box (video view)
synchronized view highlighting, 5-9
YUV
example files for fidelity analysis, 5-39
format, 5-25

Z

Zoom
video window, 5-285