

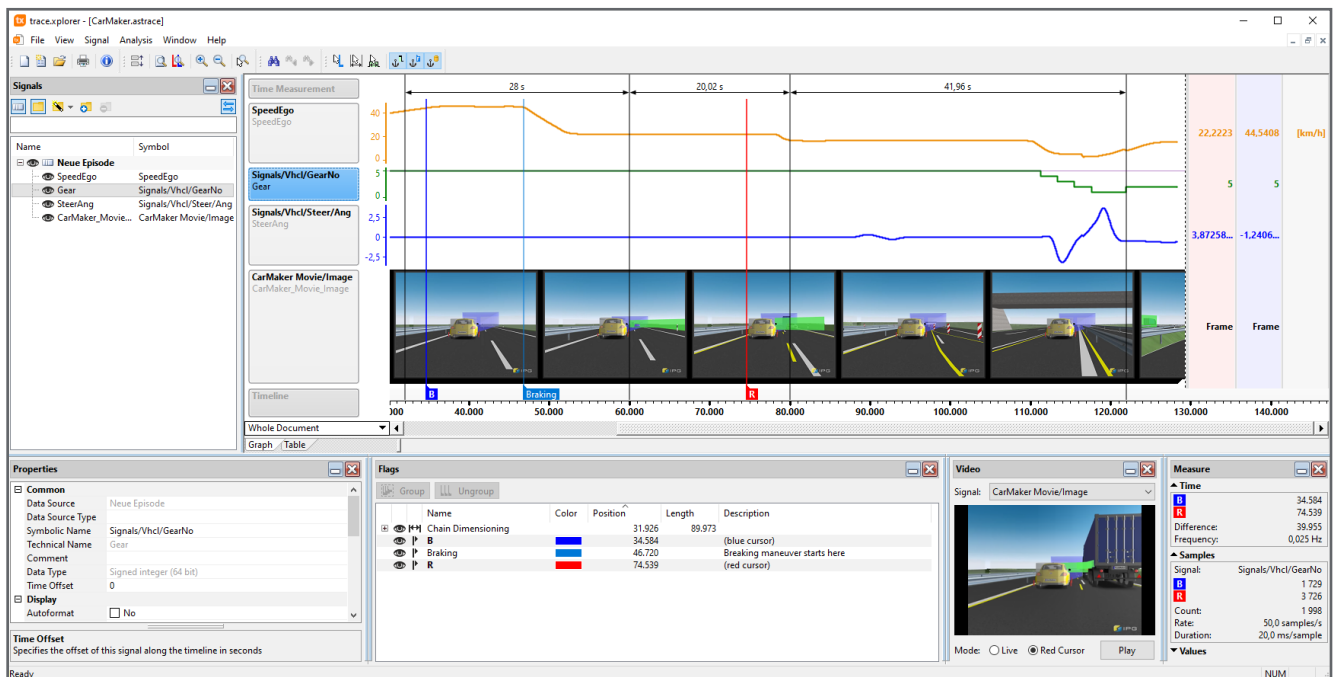
The **trace.xplorer** enables a very detailed and also interactive analysis of recorded measurement data. Once activated in **ecu.test**, test case evaluations as well as signal data from different sources can be visualized in a variety of ways and, thanks to a wide range of features, examined for concrete issues.

Visualizing signal data

During the repeated, automated testing of electronic control units (ECUs), extensive traces with signal data are generated at every realiza-

tion level (MiL, SiL, HiL, vehicle). The database built in this way can include numeric signals and bus messages as well as video and audio streams. After feeding the data into the **trace.xplorer**, the signals can be viewed, scaled

or compared extensively with other signals, events can be marked, and time periods and signal amplitudes can be measured. Even with the basic functions, an in-depth insight into the recorded data is possible.



Exploratory interpretation of test reports

To evaluate the system under test (SUT), automated offline trace analyses are usually used, which are applied to the stored database.

The **trace.xplorer** provides the necessary detailed knowledge on the existing signal data to develop such offline trace analyses. In this way, the following questions can often be clarified in advance:

- Which signals are relevant for evaluating the test case?
- What are the concrete, temporal dependencies between different signals?
- Which signal values are useful as thresholds for evaluations or trigger ranges?

Trace analyses generate test reports with detailed analysis results when they are executed. The static information contained therein can be subject to exploratory tracking using the **trace.xplorer** and it allows conclusions to be drawn about questions such as:

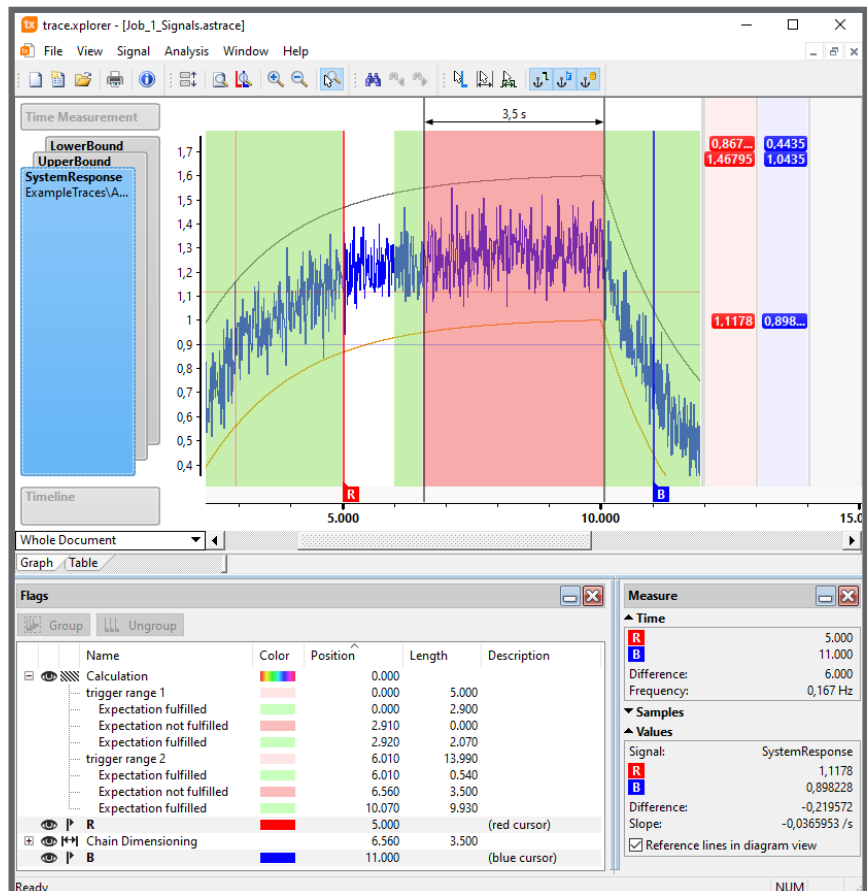
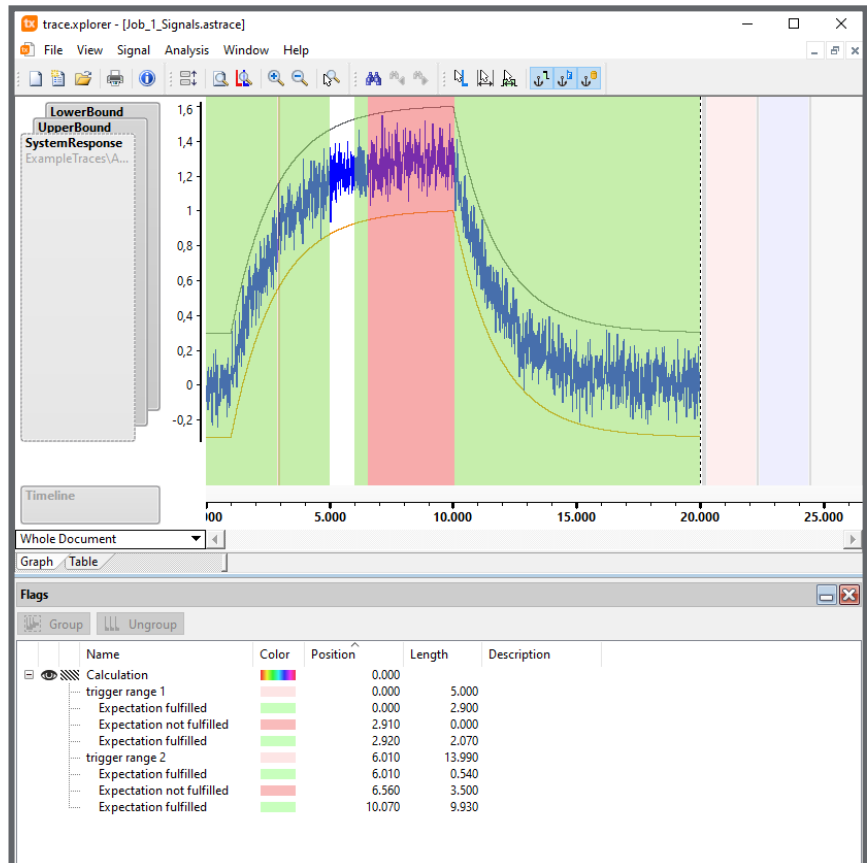
- Should the trace analysis (or even the test case) be corrected, made more robust, or extended?
- How exactly did the present evaluation of the system under test (SUT) come about?
- How good is the quality of the recorded signal data (measurement resolution, dropouts, outliers, ...)?
- Does automated testing of ECUs require additional signal data to be recorded?

Finding without a long search

Need to determine suitable threshold values for the parameterization of test conditions? Or query signal states in the trace? Or quickly and easily find specific events in the trace?

This is possible with the help of the integrated search function and the input of simple signal conditions. Partial conditions can be logically combined (AND/OR).

The search can be carried out forwards or backwards and it sets a measurement cursor at the next or previous point in time at which the formulated overall condition is fulfilled.



Test report in **ecu.test** (top), zoomed detail view with cursors and dimension chain in **trace.xplorer** (below)

Importing signal data subsequently

If additional signal data is required for understanding or evaluating a trace analysis, it can also be subsequently imported into an existing **trace.xplorer** document. The advantage of this is that the analysis does not have to be extended and executed again.

The import function supports MDF4 and CSV files as well as bus logging files (ASC, BLF, TTL). If required, these files can be batch converted to the **trace.xplorer** file format (ASTRACE) to allow selected data to be archived in a consistent form for subsequent analysis.

Individual **trace.xplorer** files can also be merged into a single document in this way, for example, to enable direct comparison of the results of several test runs.

Analyzing signal data individually

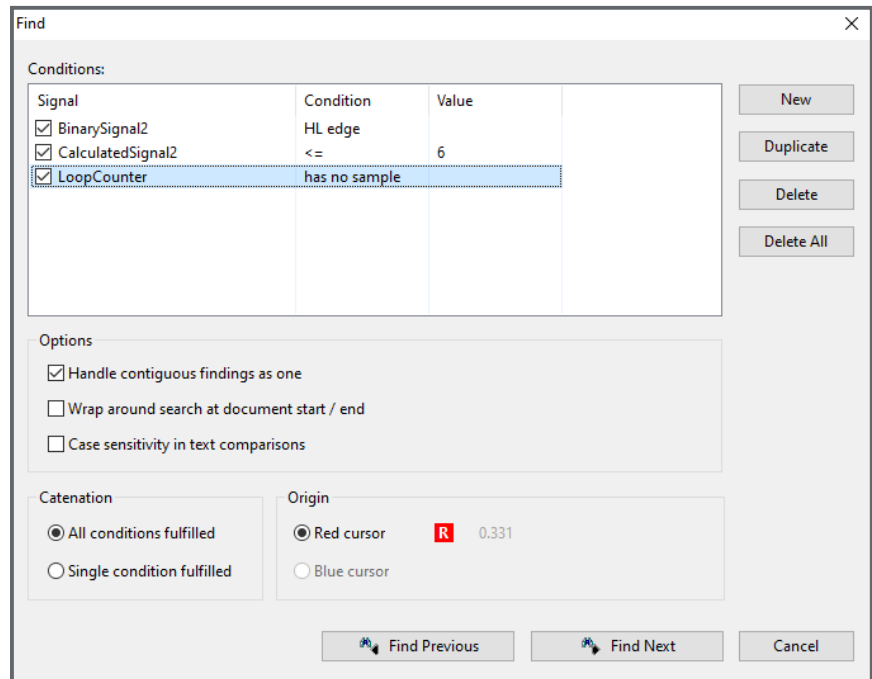
With the help of analysis scripts, the analysis can be user-defined and extended for more complex tasks, e.g.

- To recognize patterns (bouncing)
- To count events (pulses)
- To test derived signal quantities (slopes)

The scripts are programmed in Visual Basic Script. The stored signal data is accessed via the documented **trace.xplorer** Scripting API.

This allows program-controlled determination and output of key figures, setting flags in the trace or exporting selected signal values and additional information to a CSV file.

Once created, scripts can be reused. In this way, sporadic events in continuously acquired measurement data can be found at the push of a button, or recurring analysis tasks can be automated in a time-saving manner.



Search for signal conditions in the **trace.xplorer**

Key features at a glance

- Display of more than 1,000 signals in graph or table view
- Hierarchical trace navigation – even for very long measurement files with large amounts of data and a temporal resolution in nanosecond range
- Display of video signals as single images or as video with playback function
- Marking and description of relevant events in the trace using flags
- Measuring times and amplitudes
- Dimensioning time spans
- Search function to find signal states, edges or thresholds easily
- Printing, importing and exporting of signal data
- Programmable analysis scripts in Visual Basic Script
- User interface: German or English
- Supported data formats:
 - ASTRACE, AS3TRACE
 - MDF4, ASC, BLF, TTL
 - CSV

System requirements

- Operating system: Windows 10 or 11, 64-bit
- CPU: at least 4 cores
- Free hard disk capacity: at least 3 GB
- RAM: at least 4 GB, 8 GB recommended
- Screen resolution: at least Full HD (1920 x 1080)

To use file paths longer than 256 characters, enable operating system support in Windows (see <https://learn.microsoft.com/en-us/windows/win32/fileio/maximum-file-path-limitation>)