trace.xplorer

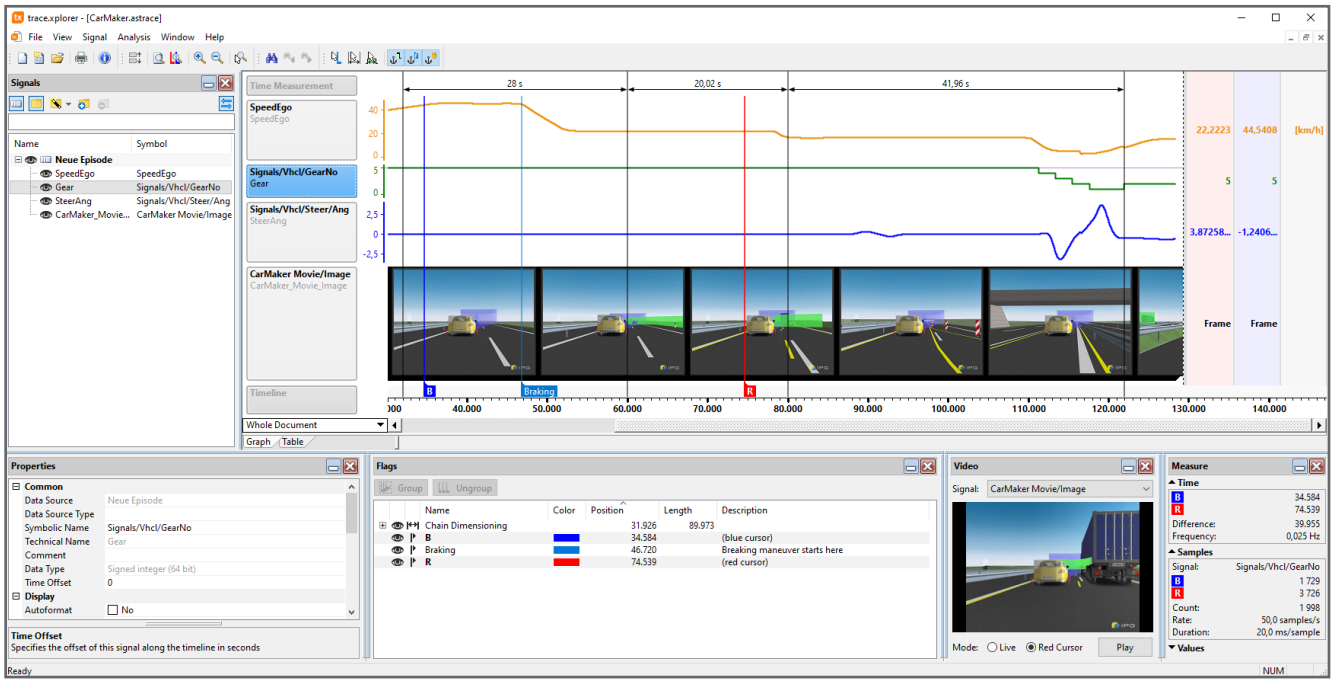
Datenblatt

Stand: Mai 2025

Mit dem **trace.xplorer** gelingt eine sehr detaillierte und zugleich interaktive Analyse aufgenommener Messdaten. Einmal in **ecu.test** aktiviert, lassen sich Testfallbewertungen sowie Signaldaten unterschiedlichen Ursprungs vielfältig visualisieren und dank eines großen Funktionsumfangs auf konkrete Sachverhalte untersuchen.

Signaldaten visualisieren

Während des wiederholten, automatisierten Tests von Steuergeräten (ECU) in jedem Realisierungslevel (MiL, SiL, HiL, Fahrzeug) entstehen umfangreiche Traces mit Signaldaten. Die so aufgebaute Datenbasis kann sowohl numerische Signale und Busbotschaften als auch Video- und Audiostreams umfassen. Nach dem Einlesen der Daten im **trace.xplorer** können die Signale ausgiebig betrachtet, skaliert, mit anderen Signalen verglichen, Ereignisse markiert oder auch Zeitspannen und Signalamplituden ausgemessen werden. Schon mit den Basisfunktionen ist ein tiefer Einblick in die aufgezeichneten Daten möglich.



Testreports explorativ interpretieren

Zur Bewertung des System-Under-Test (SUT) werden meist automatisierte Offline-Traceanalysen eingesetzt, die auf die gespeicherte Datenbasis angewendet werden.
Der **trace.xplorer** liefert das notwendige Detailwissen über die vorhandenen Signaldaten für die Entwicklung solcher Offline-Traceanalysen. So lassen sich oftmals schon im Vorfeld folgende Fragen klären:

- Welche Signale sind zur Bewertung des Testfalls relevant?
- Welche konkreten, zeitlichen Abhängigkeiten bestehen zwischen verschiedenen Signalen?
- Welche Signalwerte sind als Schwellwerte für Bewertungen oder Triggerbereiche sinnvoll?

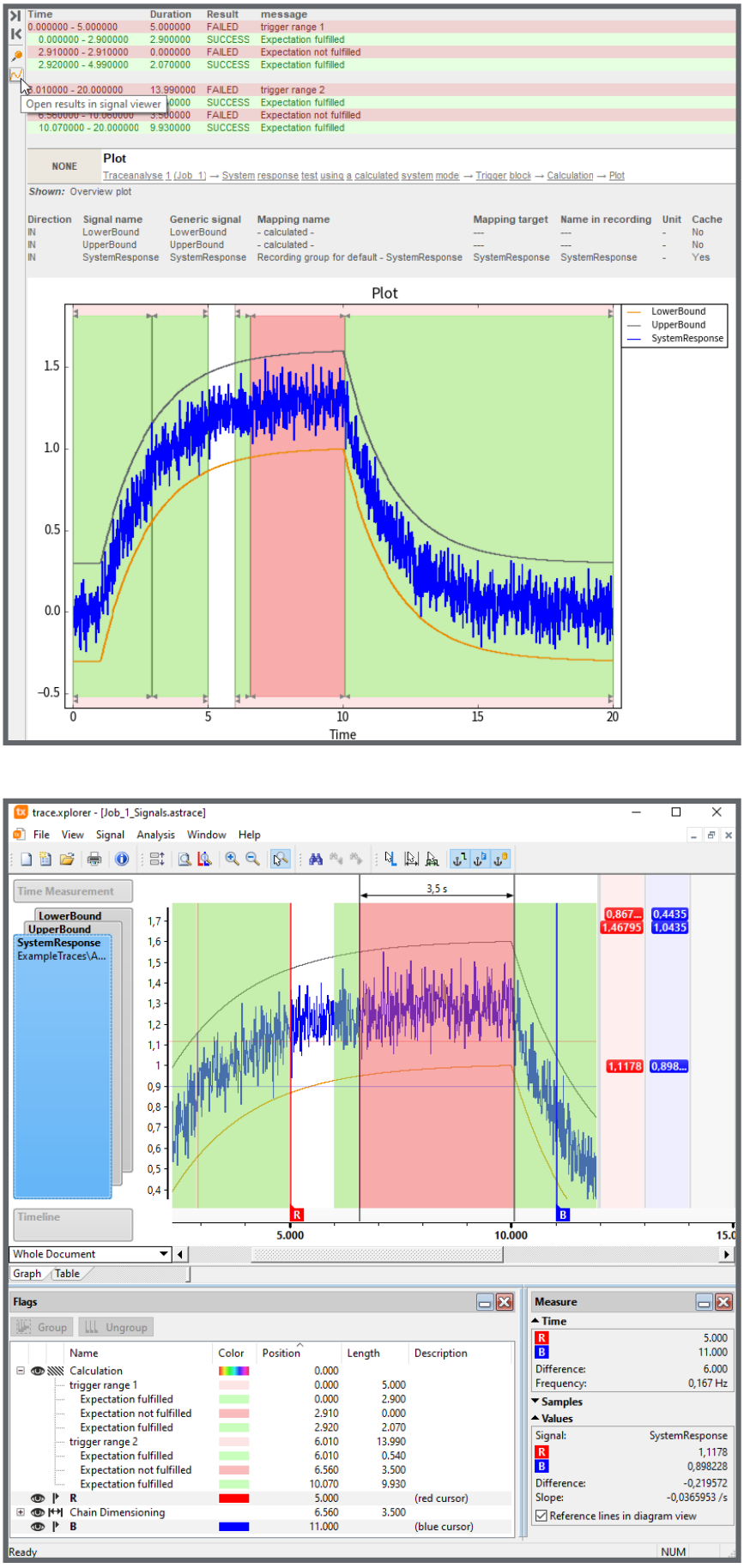
Traceanalysen erzeugen bei ihrer Ausführung Testreports mit detaillierten Analyseergebnissen. Die darin enthaltenen statischen Informationen lassen sich mit dem der **trace.xplorer** explorativ nachvollziehen und erlauben Rückschlüsse auf Fragestellungen, wie:

- Muss die Traceanalyse (oder auch der Testfall) korrigiert, robuster gemacht oder erweitertet werden?
- Wie genau ist die vorliegende Bewertung des Testobjektes (SUT) zustande gekommen?
- Wie gut ist die Qualität der erfassten Signaldaten (Messauflösung, Aussetzer, Ausreißer, ...)?
- Erfordert der automatisierte Test der ECUs die Erfassung zusätzlicher Signaldaten?

Finden ohne langes Suchen

Geeignete Schwellwerte für die Parametrierung von Testbedingungen bestimmen? Signalzustände im Trace abfragen? Schnell und einfach bestimmte Ereignisse im Trace finden?
Mithilfe der integrierten Suchfunktion und der Eingabe einfacher Signalbedingungen ist das möglich. Teilbedingungen können dabei logisch (UND/ ODER) verknüpft werden.

Die Suche kann vorwärts oder rückwärts erfolgen und setzt einen



*Testreport in **ecu.test** (oben), Gezoomte Detailansicht mit Cursors und Maßkette im **trace.xplorer** (unten)*

Messcursor zur nächsten bzw. vorhergehenden zeitlichen Position, an der die formulierte Gesamtbedingung erfüllt ist.

Signaldaten nachträglich importieren

Sind für das Verständnis oder die Bewertung einer Traceanalyse weitere Signaldaten erforderlich, können diese auch nachträglich in ein bestehendes **trace.xplorer**-Dokument importiert werden. Das hat den Vorteil, dass die Analyse nicht erweitert und erneut ausgeführt werden muss.

Die Importfunktion unterstützt MDF4- und CSV-Dateien, Busaufzeichnungen (PCAP, ASC, BLF, TTL) sowie Videodateien (MP4, MKV, AVI, ASF). Bei Bedarf können diese Dateien im Batch-Betrieb in das **trace.xplorer**-Dateiformat (ASTRACE) konvertiert werden, um ausgewählte Daten für nachgelagerte Analysen in einheitlicher Form archivieren zu können.

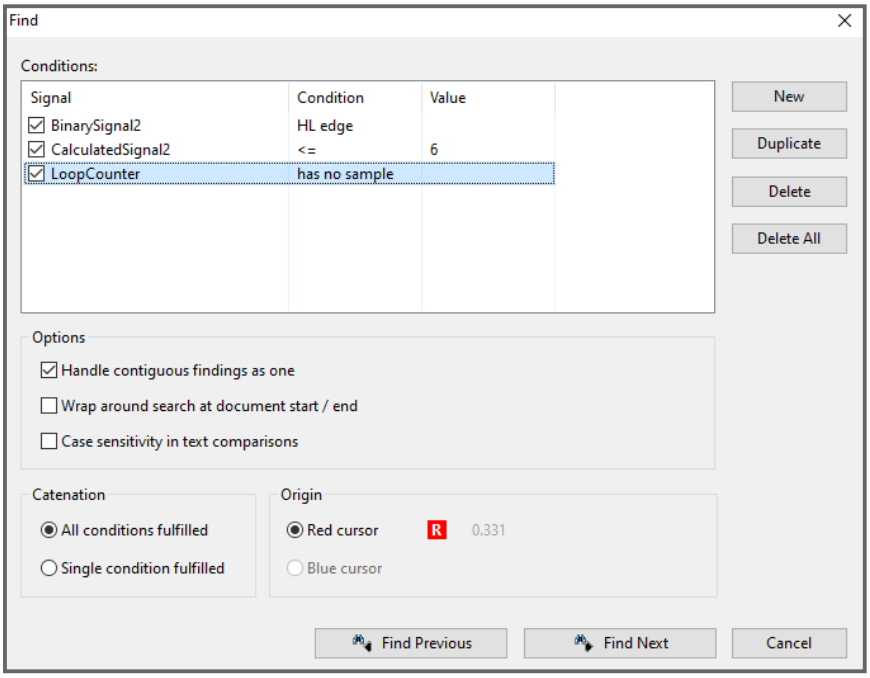
Auch das Zusammenführen von einzelnen **trace.xplorer**-Dateien in ein gemeinsames Dokument ist auf diesem Weg möglich, beispielsweise um die Ergebnisse mehrerer Testläufe direkt miteinander vergleichen zu können.

Signaldaten individuell analysieren

Mithilfe von Auswerteskripten kann die Analyse benutzerdefiniert und für komplexere Aufgaben erweitert werden, bspw.

- Erkennen von Mustern (Prellen)
- Zählen von Ereignissen (Impulse)
- Prüfen von abgeleiteten Signalgrößen (Anstiege)

Die Skripte werden in Visual Basic Script programmiert. Der Zugriff auf die gespeicherten Signaldaten erfolgt über die dokumentierte **trace.xplorer** Scripting API. Programmgesteuert lassen sich damit Kennzahlen ermitteln und ausgeben, Markierungen (Flags) im Trace setzen oder ausgewählte Signalwerte und Zusatzinformationen in eine CSV-Datei exportieren. Einmal erstellte Skripte können wiederverwendet werden. So lassen sich sporadische Ereignisse in fortlaufend gewonnenen Messdaten auf Knopfdruck finden oder auch wiederkehrende Analyseaufgaben zeitsparend automatisieren.



*Suche nach Signalbedingungen im **trace.xplorer***

Leistungsmerkmale im Überblick

- Anzeige von mehr als 1.000 Signalen in Diagramm- oder Tabellenansicht
- Hierarchische Trace-Navigation – auch für sehr lange Messdateien mit großen Datenmengen und einer zeitlichen Auflösung im Nanosekundenbereich
- Darstellung von Videosignalen als Einzelbilder oder als Video mit Wiedergabefunktionalität
- Markierung und Beschreibung von relevanten Ereignissen im Trace durch Flags
- Messen von Zeiten und Amplituden
- Bemaßen von Zeitspannen
- Suchfunktion zum einfachen Finden von Signalzuständen, Flanken oder Schwellwerten
- Drucken, Importieren und Exportieren von Signaldaten
- Programmierbare Auswerteskripte in Visual Basic Script
- Konfiguration von zusätzlichen Wireshark-Aufzeichnungsschnittstellen für Ethernet, CAN, CAN-FD und LIN über Vector XL / SIL Kit oder Messboxen diverser Hersteller
- Bedienoberfläche: Deutsch oder Englisch
- Unterstützte Datenformate:
  - ASTRACE, AS3TRACE
  - CSV, MDF4, PCAP, PCAPNG, ASC, BLF, TTL
  - MP4, MKV, AVI, ASF

Systemanforderungen

- Betriebssystem: Windows 10 oder 11, 64 Bit
- CPU: mindestens 4 Kerne
- Freie Festplattenkapazität: mindestens 3 GB
- Arbeitsspeicher: mind. 4 GB, Empfehlung 8 GB
- Bildschirmauflösung: mindestens Full HD (1920 x 1080)

Zur Verwendung von Dateipfaden mit einer Länge von mehr als 256 Zeichen muss in Windows die betriebssystemseitige Unterstützung aktiviert sein (siehe https://learn.microsoft.com/de-de/windows/win32/fileio/maximum-file-path-limitation)