

## Flexible and automated analysis of measured data

TRACE-CHECK is the highly efficient tool for the automatic evaluation and validation of recorded measured data. With TRACE-CHECK, even complex requirements can be formalized efficiently and reusable. The created test specifications enable the automatic validation of measured data and thus a fast assessment of large amounts of data in different recording formats.

### Overview of the key features

- Easy specification via
  - Graphical signal-modeling
  - Timing diagrams
  - Real-time logic
  - Python interface
- Support for all common recording formats
- High reusability of analyses
- Intuitive graphical user interface
- Clear presentation of results
- Seamless integration into the test automation software ECU-TEST

### Benefits of TRACE-CHECK

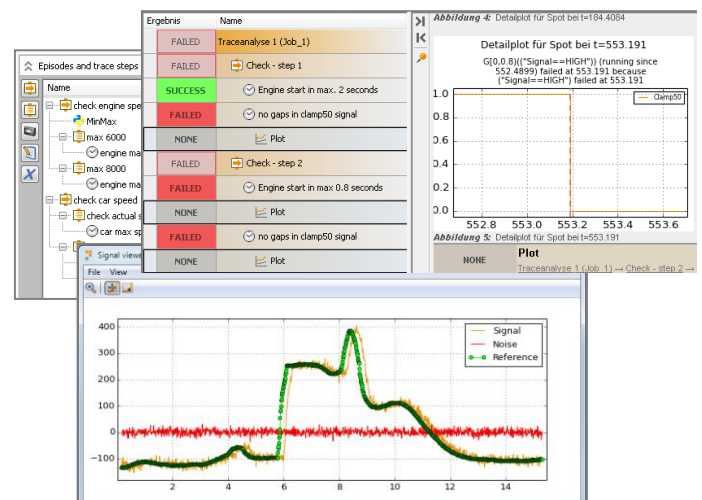
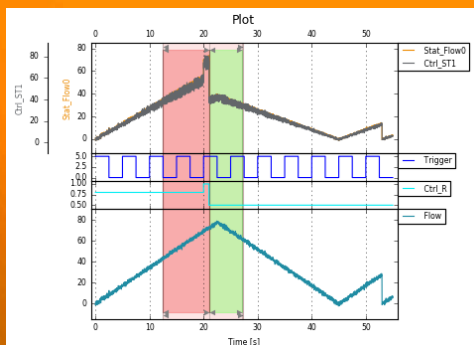
With TRACE-CHECK, trace analysis becomes an integral part of manual and automated test execution. TRACE-CHECK facilitates design, implementation, execution and evaluation of trace analyses, including the generation of intuitive test reports.

### User-friendliness

The graphical user interface assists developers and testers in the whole process from creating and managing analysis components, over the execution of individual analyses and complete analysis projects, to the creation of structured reports.

## New

- Configurable presentation of signals and results



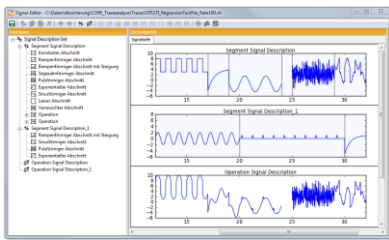
### Modularity and reusability

In TRACE-CHECK, analyses are composed from individual, parameterizable building blocks. Due to a generic analysis description, the created analyses can be used independently from the specific recording hardware and software and are therefore reusable to a great extent.

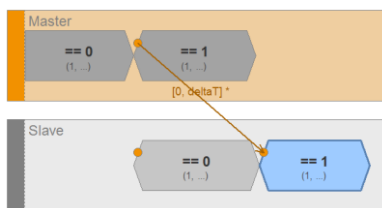
## Means of specification

To the user various means of specification are available to formalise requirements in the form of analysis components:

**Graphical signal modelling** enables the user to specify reference signals intuitively.



With **Timing diagrams** both simple and complex signal relationships can be described clearly and



at the same time formally.

The **Real-time logic** is based on Metric Temporal Logic (MTL) and especially enables the user to formalise requirements on the temporal behaviour of signals in a compact way.

```

G - F - X - U - R - H0 O0 LET IN YIELD and -
if "EngineStartSig == HIGH" then
    Finally0,5,1,5 ("EngineRunning == HIGH")
else
    Globally0,0,5 ("EngineRunning == LOW")
    
```

Auswertung der Formel: An jeder Messstelle von Signalen des Traceschrittes

With the **Python interface** the users can implement the analysis specifications themselves. In doing so, they can focus on the actual requirements, as many tasks – from processing

```

Code:
1 def Next(event, hold, dataContainer, parameters):
2     # get curVal for comparison and storing
3     curVal = event.GetValue("StateSig")
4     # init state
5     if event.GetTime() > 7.0:
6         # real value found => wrong state
7         if event.GetValue("Sig1") != curVal:
8             dataContainer.SetResultFailed()
9         else:
10            dataContainer.state += 1
11            # correct value in even state
12            if dataContainer.state % 2 == 0:
13                # store coreVal for later reuse
14                dataContainer.lvalue = curVal
15
16
    
```

traces to generating report entries – are carried out fully automatic by TRACE-CHECK.

## Supported formats

- Raw-data trace formats:
  - ASC, BLF, TTL: CAN and FlexRay recordings
  - PCAP: Ethernet recordings
- Signal catalogs:
  - DBC
  - FIBEX 4.1.1
- Signal-based trace formats:
  - CSV
  - MAT: MATLAB/Simulink, ControlDesk
  - MDF 3.0, 3.1, 3.2, 3.3, 4.0, 4.1
  - STI, STZ 2.0 ASAM HiL-API
  - TDMS: National Instruments

Support for further formats can be added upon request.

## Interfaces

Via **automation interfaces** (XML-RPC, COM) all TRACE-CHECK work steps can be controlled, for instance for a seamless and fully automatic operation in an existing tool chain.

## TRACE-CHECK versions

**TRACE-CHECK:** Complete package, consisting of editor, runtime engine and user manual.

**Trace-analyze-module for ECU-TEST:** Integration of the complete function set of TRACE-CHECK into the test automation solution ECU-TEST.

**TRACE-CHECK module for EXAM:** Integration of the complete function set of TRACE-CHECK into the EXAM test environment.

**TRACE-CHECK as a web service:** Provision of TRACE-CHECK in corporate networks via web service interface.

## Service and support

In addition to phone and email support, TraceTronic offers training courses for developers and testers. We are happy to assist you with the integration of TRACE-CHECK in established process chains, with qualification of TRACE-CHECK in your projects according to ISO26262 as well as in the setup of solutions for test and validation and for tests of control unit functions.

On a project basis, TraceTronic carries out special adjustments and integration of further software (e.g. test management tools) and formats in TRACE-CHECK.