Configure and Start TRACE32 from Python

2023-03-22 - Comment (1) - Debug

Since 2020, Python programs can control TRACE32 via the lauterbach-trace32-rcl module (pyrcl). Up to now, TRACE32 must be started using a config file, which requires familiarization with the TRACE32 configuration file syntax or the use of the configuration tool t32start.exe. Now Lauterbach offers a new lauterbach-trace32-pystart module (pystart) which allows the configuration and start of TRACE32 directly from Python. See example below.

Beta testers can install pystart by using `pip install --upgrade lauterbach-trace32-pystart`. The documentation is available for download below under “Attachments”.

For feedback and questions, please contact support@lauterbach.com (include "pystart" in the subject).

Supported host OSes: Windows, Linux, MacOSX.

Example for starting TRACE32 PowerView for TriCore using a Python script:

```
import lauterbach.trace32.pystart as pystart

import sys

debugger_node_name = sys.argv[1]

pystart.defaults.system_path = r"C:\T32"

powerview =
pystart.PowerView(pystart.UDPConnection(debugger_node_name),
"t32mtc")
powerview.title = f"TRACE32 PowerView for TriCore 0 at PowerDebug Pro
{debugger_node_name}"
powerview.id = "T32_tc0"

powerview.start()
```

```
powerview.wait()
```

Example for starting TRACE32 PowerView for TriCore using a config file and the command line:

```
; classic TRACE32 configuration file

OS=
ID=T32_tc0
SYS=C:\T32

PBI=
NET
NODE=${1}

SCREEN=
HEADER=TRACE32 PowerView for TriCore 0 PowerDebug Pro ${1}
```

```
; host OS command line
t32mtc.exe debugger_node_name
```

Attachments

- lauterbach_trace32_pystart_v0_1_6_documentation.zip [2.82 MB]
- lauterbach_trace32_pystart_v0_1_6_examples.zip [9.26 KB]
- lauterbach_trace32_pystart_v0_1_7_documentation.zip [2.82 MB]
- lauterbach_trace32_pystart_v0_1_7_examples.zip [9.26 KB]

Comment (1)

Comment (1)

W(**Wilson Mark (ETAS-VOS/XEO-ARC9)**

10 months ago

This is excellent! I really appreciate the work Lauterbach has done for adding Python support.