# Configure and Start TRACE32 from Python

2023-03-22 - Comments (7) - 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

# Comments (7)

## Comments (7)

W( **Wilson Mark (ETAS-VOS/XEO-ARC9)**
2 years ago
This is excellent! I really appreciate the work Lauterbach has done for adding Python support.

壮 **壮徐**
1 year ago
Now I can start trace32 through python. Next, I want to run a cmmm script after starting and automatically click a function button in the script. Please tell me how to do it, thank you very much!

**Wafi Jmal**
1 year ago
Hello, To execute a CMM script through a Python script, you can use the "cbg.cmm()"function. Please refer to this for more information: (~~\demo\api\python\rcl\doc\html) Additionally, you can modify the CMM script to accept parameters, which may help resolve the issue related to the button execution. Regards ,Wafi

A **Ayush**
1 year ago
How to execute a cmm script using python. I am not able to find any function after launching trace32 from Powerview.

**Wafi Jmal**
1 year ago

To execute a CMM script you should use dbg.cmm () function

This example will help in this case:

import lauterbach.trace32.rcl as t32api

# Connect to the Trace32 debugger

dbg = t32api.connect()

# Execute the CMM script using the dbg.cmm

dbg.cmm(r"D:\debug\T32\program.cmm")

Best Regards

Wafi

YG **Yash Goyani**
1 year ago
Hello, Thank you for adding pyrcl and pystart for Trace32 python integration. As I understand pystart only launch Trace32 on machine where python script is executing, is there any way where I can launch Trace32 on different machine on same LAN? Consider a scenario where I need to automate a script execution on remote machine on same LAN. I want to first start a Trace32 instance and a simulator inside that instance on remote machine. After that I want to connect to it via remote api pyrcl from my machine and execute script. Is it possible to archive this without any physical access to remote machine?

**Wiem Wala Benayed**
1 year ago

You can execute the script that launch TRACE32 in the remote machine using ssh:

ssh user@remote_ip "python3 -u -" < script.py #if the script in the local machine

or

ssh user@remote_ip "python3 /path/to/script.py" #if the script in the remote machine

Please note that currently, it is possible to access TRACE32 via API from a remote machine. However, we plan to change this approach in the future, and an option will need to be specified to enable remote access.

For this a trace32-pystart will be updated to be able to allow remote access.