AUTOMATED IEC 61850 SERVER REGRESSION TESTING

This document describes how to automate the IEC 61850 Ed2 server regression test. It describes:

- Why automated regression testing is important
- The scope of the regression test
- Regression Test Environment
- Build server integration
- Test approach
- Automation details for UniGrid SA

Why automated regression testing is important

Imagine a developer that accidently makes an mistake in the IEC 61850 firmware. Without IEC 61850 regression testing it may take several months before a test lab will perform a conformance test and detect the issue. Then the issue need to be fixed, but the development team is working on another project and it may take a while before the issue is resolved. Then the firmware needs to be re-tested again by the internal or external tet lab. This is very time consuming and costly process.

Imagine you have automated regression testing. The developer makes an accidental mistake in the IEC 61850 firmware. But this time the automated regression test is executed the same day. The next day the developer will receive a notification about a failed test script, he remembers what he was working on and can quickly identify the cause and fix it.

It's common knowledge that fixing a software bug during the early stage in the development is 10 to 100 times faster and cheaper than fixing it at the end of the development stage or even worse when the IED firmware is already released to end-users.

We estimate that preventing a few mistakes already returns the investment. Secondly it will improve and guarantee the quality of each major/minor IEC 61850 firmware version.

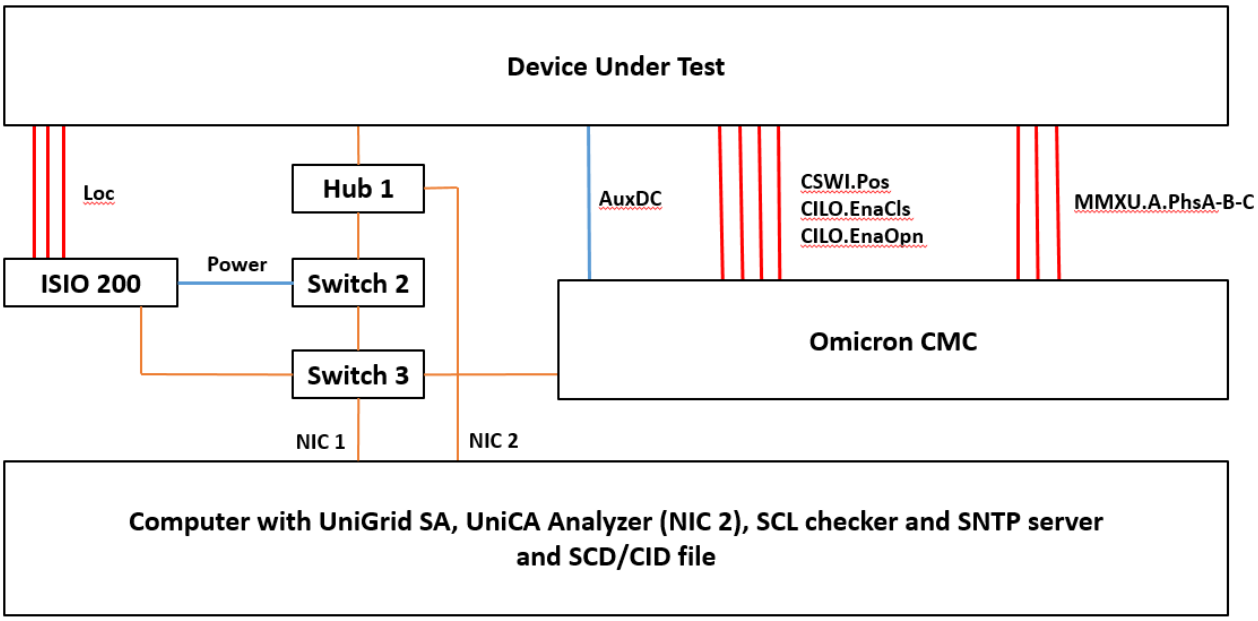
The scope of the regression test

The start of the regression testing is to execute the automated part IEC 61850-10 conformance test. Some test cases may require manual input which is out-of-scope for automated regression testing. With the proper test environment manual input can be prevented and over 90% of the conformance test can be automated.

Apart form the conformance test you can develop and add your own test scripts. The UniGrid test tool offers a script editing and on special request even script debugging features to develop and verify your own scripts that can be added to the regression test.

Regression Test Environment

To automate the regression test, set-up/wire the test environment and configure the device under test (DUT) as specified in below tables. Note that the Omicron CMC is limited to 4 binary outputs, the Omicron ISIO 200 has 8 outputs. UniGrid SA can control one CMC and one ISIO simultaneously.



Build server integration

The UniGrid SA regression test toold can be integrated with common build server automation systems like for example Mr. Jenkins.

The interface from the build server to UniGrid SA is executed via the command line interface, for example the following command creates a new project, imports applicable settings and executes multiple test scripts:

```
UniGridCLI.exe --create-project Server61850Ed2Test --import-settings
"C:\JENKINS\regressionsettings6.xml" --testcase sAss1, sAss2, sAss3, sAssN2, sAssN3,
sAssN4, sAssN5
```

The result is passed or failed when one of the test cases failed. When failed the developer can analyse the network trace and script log that are created and archived automatically by UniGrid SA.

Test Approach

Test Step	Activity
0.	**Build the firmware**
1.	Load the firmware into the device
2.	**Load the test configuration into the device**
3.	Execute the automated UniGrid SA conformance test scripts Execute your own automated UniGrid SA test scripts
4.	**The build server distributes the test results**

Annex A: Automation details supported by UniGrid SA

Automation for Association tests

Disconnect physical link (sAssN4, sRpN5, sBr20)	X	Turn power on/off of an Ethernet switch using ISIO 200 output X
Power supply interrupt (sAssN5, sSg11, sSg12, sRp7, sGop3, sGop7)	X X	Turn power on/off using ISIO 200 ouput X Set AuxDC Voltage to 0 or X volt using Omicron CMC

Automation for Server tests

Change Mod to Test (sSrv12, sSrv13)	X	Operate Mod control object XXXX
Change Mod to Off (sSrv12, sSrv13)	X	Operate Mod control object XXXX

Automation for Unbuffered & Buffered Reporting tests

ST Data change A / B (sRp3, sRp8)	X X	ISIO 200 output X / Y Omicron CMC binary output X / Y
MX Data change A / B (sRp8)	X X	ISIO 200 output X / Y Omicron CMC analogue 50/60 Hz AMP/VOL phase A/B/C
Quality change (sRp3, sRp10)	X X X	ISIO 200 output X Omicron CMC output X Operate Mod to on/off/test
Data update (sRp3)	X X X	ISIO 200 output X Omicron CMC output X Trigger service tracking of LTRK.xxxx

Automation for Control tests

DOns position open / close	X X	ISIO 200 output X / Y Omicron CMC binary output X /
SBOns position open / close	X X	ISIO 200 output X / Y Omicron CMC binary output X / Y
DOes position open / close	X X	ISIO 200 output X / Y Omicron CMC binary output X / Y
SBOes position open / close	X X	ISIO 200 output X / Y Omicron CMC binary output X / Y
CILO.EnaCls / EnaOpn (sCtl7)	X X	ISIO 200 output X / Y Omicron CMC binary output X
Local/Remote (sCtl16)	X X	ISIO 200 output X Omicron CMC binary output X
LocSta (sCtl17)	X X X	ISIO 200 output X Omicron CMC binary output X Operate the LocSta control object
MltLev (sCtl17)	X X	Change the MltLev setting T/F Change the Active setting group X
Change Mod to Test (sCtl5)	X	Operate Mod control object XXXX
Change Mod to Off (sCtl15)	X	Operate Mod control object XXXX

CmdBlk (sCtl18)	X	Operate the CmdBlk control object
BlkEna (sCtl19)	X	SetDataValues (BlkEna [BL])

Automation for GOOSE Publish

GoCBs	-	**GoCB1** with dataset LLN0$KEMAsetDA - **GoCB2** with dataset LLN0$KEMAsetDO - **GoCBWithMaximumLength_PaddingPad** with 32 char gocbname, 129 char goID and 32 char dataset: MaxLengthDatasetGOCB32characters - **GoCB7** with empty dataset (when supported)
Publish GOOSE.Sim=T	X	Setting, SetDataValues
GOOSE Data change	X X X	ISIO 200 output X Omicron CMC binary output X Subscribe to GOOSE message from KEMASIM

Automation for GOOSE Subscribe

GoCBs	-	**GoCB4** for GOOSE ping pong mechanism subscribing to FCDA, copy the first Boolean value in the first Boolean element - **GoCB5** for GOOSE ping pong mechanism subscribing to FCD, copy the first Boolean value in the first Boolean element - **GoCB6** for GOOSE ping pong mechanism subscribing to MAX, copy the first Boolean value in the first Boolean element
LPHD.Sim = T (sGos6b)	X X	Setting SetDataValues
LGOS	X	Settting

Automation for File Transfer tests

Generate COMTRADE file (sFt1, sTm3)	X X X	ISIO 200 output X Omicron CMC binary output X Periodically each X seconds

Contact
DNV GL – Energy
P.O. Box 9035
6800 ET Arnhem
The Netherlands
T +31 26 356 2025
E pc.helpdesk@dnvgl.com
www.dnvgl.com/pctc