

# Certification and Development with the Automated Realtime Tester

Benedikt Etzold, M.Sc.

OTH Amberg-Weiden – University of Applied Sciences  
aia – Automations Institute Amberg (PICC)



- 1 Automated RT-Tester Basics
- 2 What to expect from a Testcase?
- 3 Demo

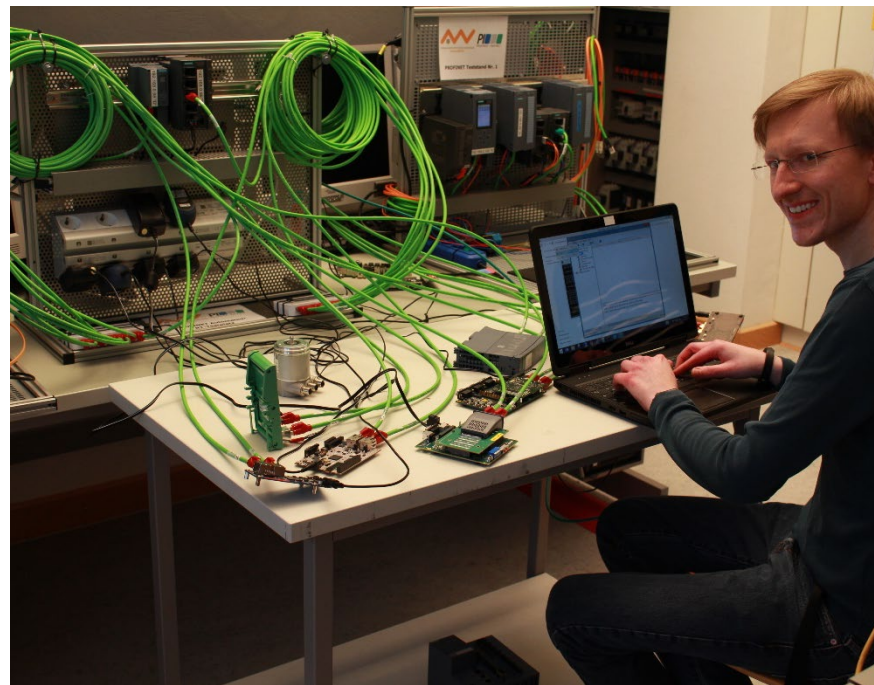
# Agenda Part 1: Automated RT-Tester Basics

---

- 1 The Automated RT-Tester
- 2 PROFINET Device Certification
- 3 PROFINET Controller Certification
- 4 PC System & Testsetups
- 5 Command Line Interface

# The Automated RT-Tester

- Conformance testing and certification for PROFINET Devices and Controllers
- Fully automated testing in terms of
  - Test execution/processing
  - Result validation
- Designed for
  - Certification in PITLs
  - In-house (pre-certification) testing
  - Development and regression testing



Testing with the Automated RT-Tester

- Characteristics:
  - Easy handling
  - Max. test coverage and high coverage of PROFINET spec
  - Reliable and detailed results
  - Reproducibility
  - Full API access
- Developed at OTH Amberg-Weiden, WG AUT with support of aia – automations institute (PICC)
  - <https://aut-oth.de/>
  - <https://aia-oth.de/>



OTH Campus in Amberg

# The Automated RT-Tester

## Starting the selected TCs

## Testsetup widget

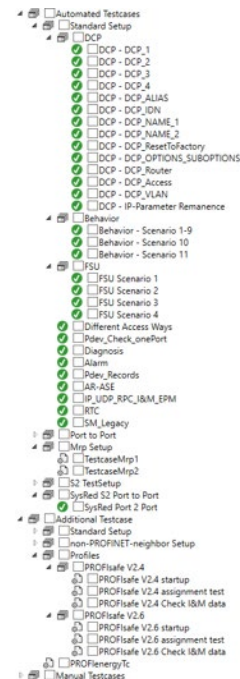
## Testcase selection

## Report window

## Output window

[illegible]

- PROFINET functionality is tested in different testcases, each with its own focus, e.g.
  - DCP
  - AR handling
  - Alarm statemachines
  - System Redundancy
  - PROFIsafe, PROFIenergy
  - ...
- Specified sequence of operations
- Published and thoroughly reviewed
- Coverage: >80% of PROFINET base spec



Testcases for Device certification

- Testcase reports
  - Each testcase produces a transparent report with a clear result
    - Pass
    - Fail
    - PassWithHint
    - Inconclusive
    - Skip
  - Detailed information and log regarding
    - DUT
    - Version of Automated RT-Tester and testcase
    - Testcase execution log
    - Hints and errors

Test Case Report: Workshop TC Automated RT Tester 2.42.3.1

## Test Case Report

Test case: Workshop TC

Result: **Pass**

### Test case description:

This Testcase will be used to show the important steps in TC development.  
It will be part of the FMO Certification Workshop.

### General Information:

#### Date

Date 12.01.2022

#### Time

Time 09:35

#### Automated RT Tester

Version 2.42.3.1

Project Name EB200P

#### Test case

Version 0.0.1

#### Device

Vendor ID 0x002A

Device ID 0x0008

Mac Address 08:00:06:02:01:10

1 / 5

## Example report



Test Case Report: Workshop TC Automated RT Tester 2.42.3.1

## Test Case Report

Test case: Workshop TC

Result: Pass

### Test case description:

This Testcase will be used to show the important steps in TC development.  
It will be part of the PNO Certification Workshop.

### General Information:

#### Date

Date 12.01.2022

Time 09:35

#### Automated RT Tester

Version 2.42.3.1

Project Name EB200P

#### Test Case

Version 0.0.1

#### Device

Vendor ID 0x002A

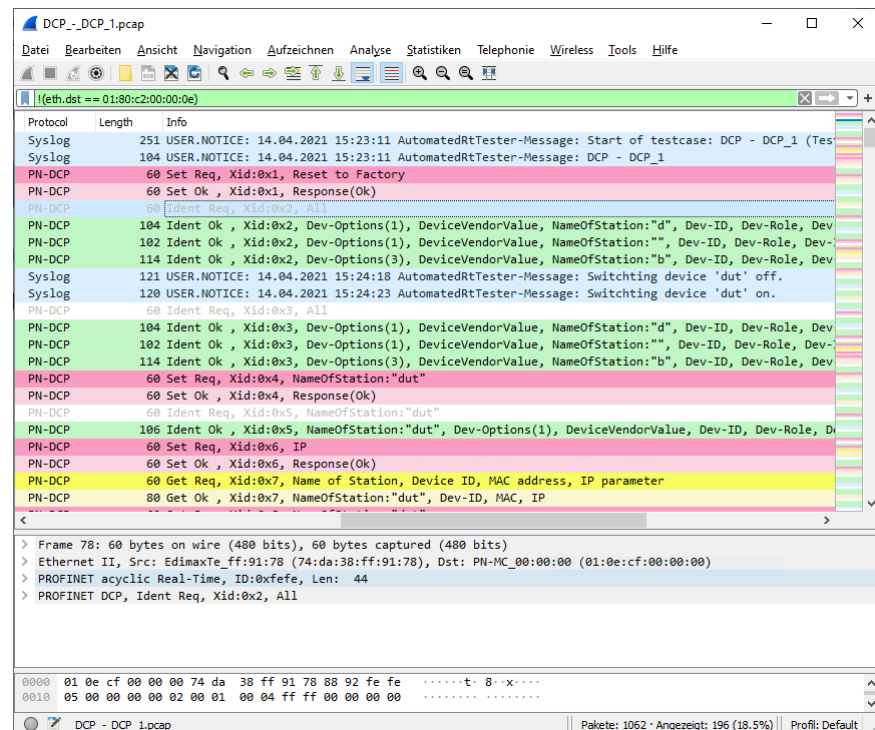
Device ID 0x0008

Mac Address 08:00:06:02:01:10

Technologies Austria AG

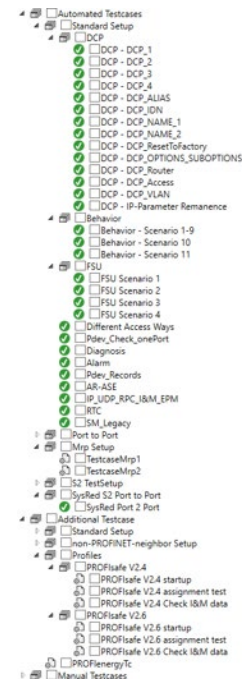
## ■ PCAPs

- Each testcase produces a PCAP file
- Dumps of all network traffic of the NICs connected with the test setup
- Injection of Syslogs for easy navigation and highlighting
- Detailed analysis
- Archiving
- Transparency



PCAP recorded with the ART

- DUT is Device
- Automated RT-Tester simulates a PROFINET Controller
- GSD as base for PROFINET configuration
- Current status:
  - 40 main device testcases
  - Split into 64 single executable testcases



## Testcases for Device Certification

- ☒ Automated Testcases
  - ☒ Controller Testsetup
    - ☒ DCP
      - ☒ DCP\_Ctrl\_Options\_Suboptions
      - ☒ DCP\_Ctrl\_Reset\_Set
      - ☒ DCP\_Ctrl\_Vlan
      - ☒ DCP\_Ctrl\_Alias
    - ☒ RTC\_Ctrl
    - ☒ Behavior\_Ctrl
    - ☒ RPC\_IP\_UDP\_EPM\_Ctrl
    - ☒ Read\_Write\_Ctrl
    - ☒ Alarm\_Handling\_Ctrl
    - ☒ DHT\_Ctrl
    - ☐ ErrorChecks\_Ctrl
    - ☐ FSU\_Ctrl
    - ☐ TopologyRead\_Ctrl
    - ☐ Legacy\_Ctrl
  - ☐ Additional Testcase
    - ☐ Standard Setup
      - ☐ Topology discovery check (debug)
    - ☐ non-PROFINET-neighbor Setup
      - ☐ Topology discovery check (Aruba) (debug)
      - ☐ Topology discovery check (Cisco) (debug)
      - ☐ Topology discovery check (Cisco + Aruba) (debug)
  - ☐ Manual Testcases
    - ☐ Controller Testsetup
      - ☒ DCP\_Ctrl\_Signal
      - ☒ Manual\_DHT\_Ctrl

- DUT is Controller
- Automated RT-Tester simulates a PROFINET Device
- TFPC (Test Flow Protocol for Controllers) for Device → Controller test commands
- Controller applications necessary (examples provided with every bundle)
- CDML (Controller Description ML)
- Current status:
  - 17 main controller testcases
  - Split into 21 single executable testcases

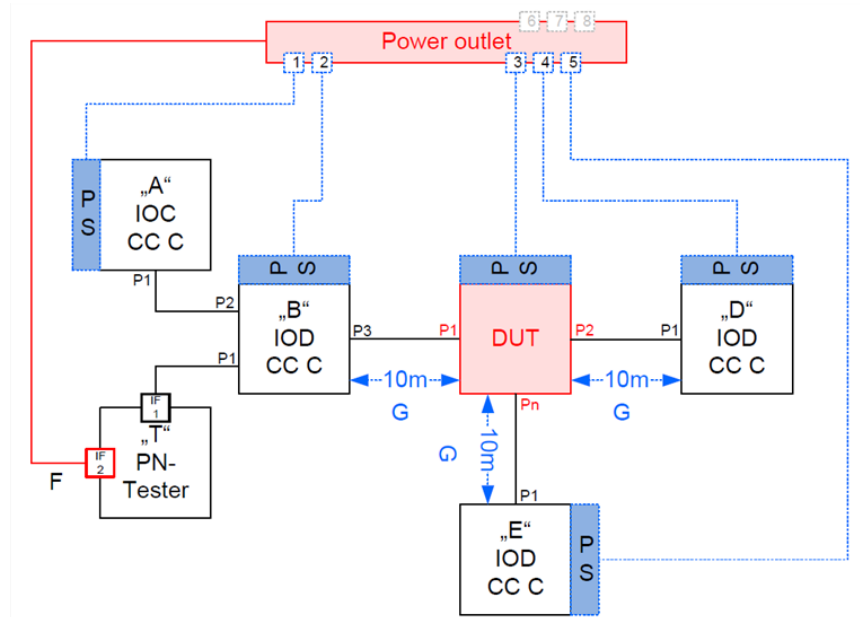
## Testcases for Controller Certification

## Hardware for testing – PC system

- Specified PC system with following requirements:
  - Windows 10® 64 Bit Professional, Version 20H2 (21H2 from next release on)
  - I350-T2/T4: 100BaseTX network card(s) with possibility to disable striping of VLAN-Tag for connection with the testsetup
  - Additional network card for connection with the power outlet
- Motivation: Reliable results and guaranteed reproducibility
- See test specification for full details



Test development at OTH Amberg



Standard testsetup

- Testsetups are specified in detail
  - PSU (Power Supply Unit)
  - Device B
  - Neighbor Devices
  - Cabling and ports
  - ...
- Testsetups vary based on the testcases, e.g.
  - Standard Testsetup
  - S2 System Redundancy Testsetup
  - Port-to-Port Testsetup
  - ...

- Specified in test specification
  - Different testsetups
  - Dedicated testsetup for each testcase
  - Additional devices, e.g.
    - Converters for FO devices
    - Non-PROFINET-Nighbor devices for TedCheck



One of the PROFINET test stands at OTH

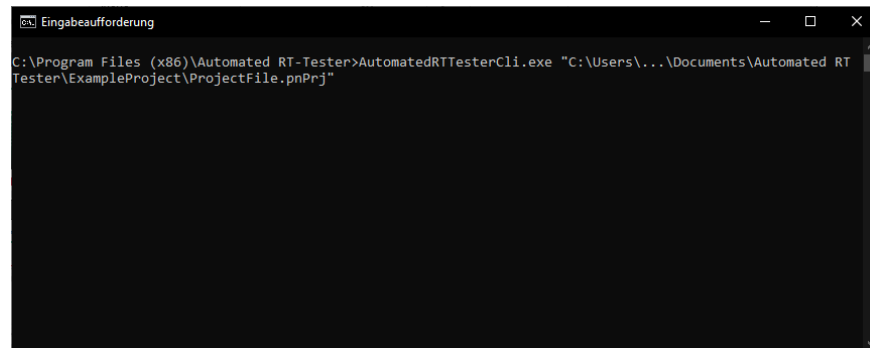


# Standard Testsetup

Demo



- Motivation:
  - CLI for further test automation
  - Regression testing
  - Automated comparison of test results
- AutomatedRTTesterCli.exe is provided with each installer of the ART
  - Default location: installation folder (C:\Program Files (x86)\Automated RT-Tester)
- Use:
  - Setup ART project
  - Pass path to project(s) as parameter



```
Eingabeaufforderung
C:\Program Files (x86)\Automated RT-Tester>AutomatedRTTesterCli.exe "C:\Users\...\Documents\Automated RT-Tester\ExampleProject\ProjectFile.pnPrj"
```

Using the CLI

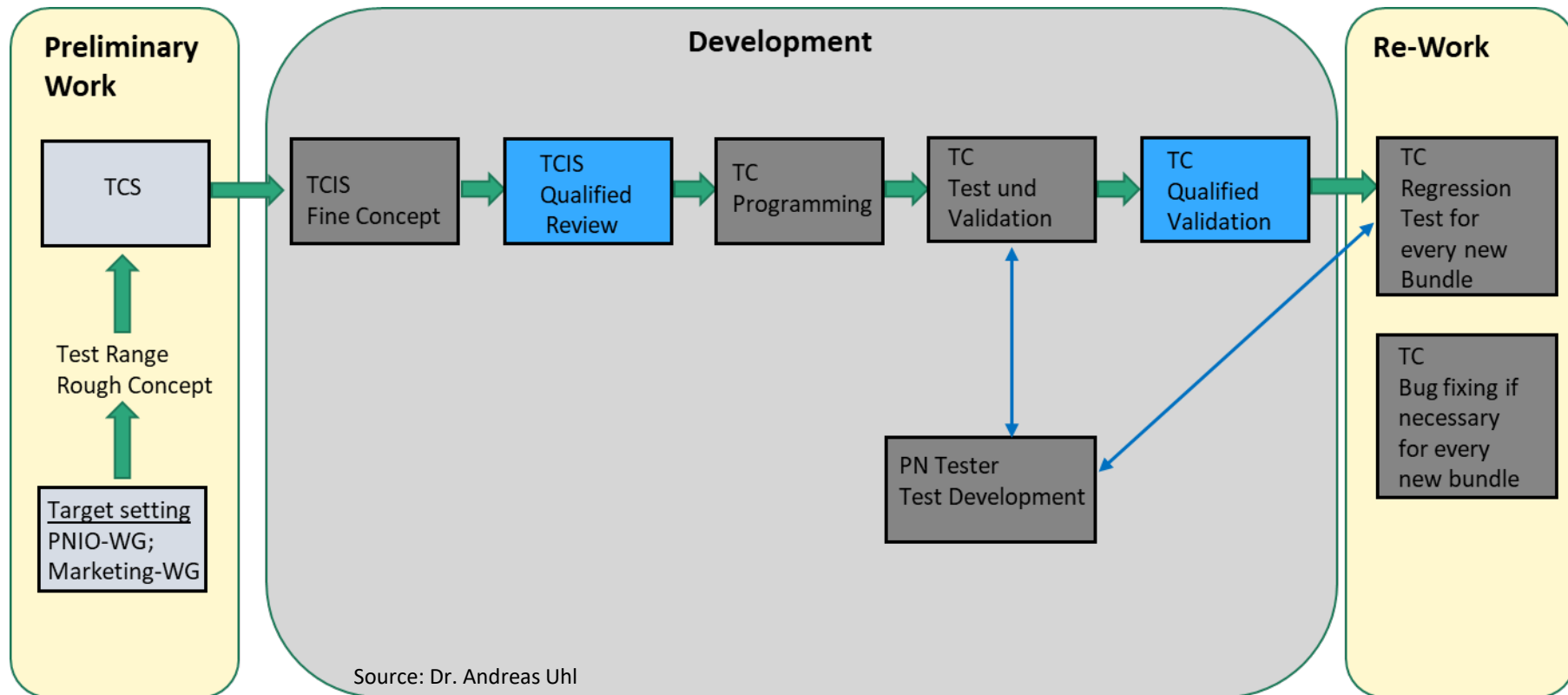
## End of part 1

Upcoming: What to expect from a testcase?

## Agenda part 2: What to expect from a testcase?

- 1 Process
- 2 Resources (Testspec, TCS, TCIS, Code)
- 3 Testcase IP\_UDP\_RPC\_EPM\_I&M in detail
- 4 Writing your own testcases

# Process



- Testspec
  - General test concept
  - Basics and coverage of scope of certification process
  - Information to provide to your PITL
  - Overview on all test systems (Automated RT-Tester, Spirta, Security Level 1, ...)
  - All testsetups
    - Devices with order numbers and settings (NameOfStation, IpParameters, ...)
    - Media converters
    - PSU
    - Cabling

- TCS – Test Case Specification
  - Specific to a certain testcase
  - What to test? What to consider when testing?
  - General part:
    - Referenced versions of base specifications
    - GSD versions valid for certification testing (loading)
    - GSD attributes used in this testcase
    - Testsetup
    - Possible results with explanation
  - Testcase details:
    - Description of the testcase
    - Focus points for test development

- TCIS – Test Case Implementation Specification
  - Specific to a certain testcase, based on the corresponding TCS
  - Exception: TCIS General containing details valid for all TCIS
    - e.g. AR-Establishment, AlarmHandling, DCP check / set NameOfStation / IpParameters, ...
  - How to test? Detailed explanation of the implementation
  - General part:
    - Additions to TCS, e.g. deviations
  - Implementation part:
    - Flow charts for all scenarios
    - Detailed description of every test step
    - Protocol tables for all tested PDUs

- Testcase code
  - Plain source code for all testcases is included in every installer of the Automated RT-Tester
  - Default location: C:\Program Files (x86)\Automated RT-Tester\Documentation\Testcases\_NO\_SUPPORT.zip
  - Better understanding of testcase
  - Hints and examples for development of user testcases



## Scenario I: Checking of RPC, IP, UDP, Part 1

1. ICMP Ping request and validation of response
2. IOC-AR (little endian RPC)
  1. Check ModuleDiffBlock and IOPS (good)
  2. Read I&M0 Data and validate response
3. IOC-AR (big endian RPC)
  1. Check ModuleDiffBlock and IOPS (good)
  2. Read I&M0 Data
    1. Validate response
    2. Compare response to block received above
4. IOC-AR, ConnectRequest fragmented on RPC layer

## Scenario I: Checking of RPC, IP, UDP, Part 2

### 5. IOC-AR (with errors)

- Record Data Read with:
  - Wrong checksum in IP part
  - Wrong checksum in UDP part
  - Wrong checksums in IP and UDP part
  - Data length in IP part is set too big
  - Data length in UDP part is set to the same value as in IP part
  - Data length in IP part is set too small
  - Data length in UDP part is set too small
- None of those frames shall be forwarded to the DCE RPC layer
- DCE RPC pings are used to ensure functionality of DCE RPC service

## Scenario II: Consistency check of EPM, I&M and DCP

1. NDR Endpoint Mapper Lookup Requests (rpc\_c\_ep\_all\_elts)
2. Read I&M0Data of device representative
3. Send DCP IdentifyAllReq
4. Compare for consistency:
  - VendorId
  - DeviceVendorValue
  - OrderId
  - HardwareRevision
  - SoftwareRevisionPrefix
  - SoftwareRevision
  - ...

## Scenario III: I&M Write

### 1. IOC-AR

- For all Submodules in I&M0FilterData.FilterDataSubmodules:
  - Read I&M0 Data
  - Write I&M0 Data and validate error code
  - Read I&M0 Data
  - For I&M1 – 4: If supported (I&M.IM\_Supported, GSD):
    - Write IM, validate success/error code
  - Read I&M5, validate success (if supported) or error code
  - Write I&M5 and validate error code

### 2. Release IOC-AR

### 3. Power cycle DUT

## Scenario III: I&M Write

### 4. IOC-AR

- Read I&O Data and compare to previously written data
- If supported: ResetToFactory Mode 1, 4, 8 or 9
  - Read writable I&M Data, check if successfully reset to factory settings.

### 5. Release IOC-AR

## Scenario IV: Check of ActivityUUID

1. IOC-AR, save RPCActivityUUID from ApplicationReadyReq
2. Release IOC-AR
3. IOC-AR, save RPCActivityUUID from ApplicationReadyReq
4. Release IOC-AR
5. Compare ActivityUUIDs
  - Subsequent ActivityUUIDs shall not be equal
  - Last 6 bytes of ActivityUUIDs shall match the MAC address of the device

- Users of the Automated RT-Tester can develop and run their own testcases
- Motivation:
  - Debugging of stack
  - Regression tests
  - Development of new features
  - ...
- API of Automated RT-Tester framework is available
- CLI support
- Detailed description of setup is included in every installer.

```

19 namespace MyFirstTestCase
20 {
21     0 references
22     public class ArtDemo : ITestcase
23     {
24         private byte[] _currentData;
25
26         private byte[] _currentOutput;
27
28         private Boolean runTest;
29
30     0 references
31     public void Start(RtTesterModuleBase rtTesterModuleBase, BackgroundWorker backgroundWorker)
32     {
33         _report = new TestcaseReport();
34         _currentData = new byte[1];
35         _currentOutput = new byte[1];
36
37         _report.Result = TestcaseResult.Pass;
38
39         ArConfiguration arConfig = new ArConfiguration()
40         {
41             Ar = ArType.IoController,
42             Startup = StartupType.Advanced
43         };
44
45         runTest = true;
46
47         ArCommParameters arCommParas;
48         ...
49         ArProcessingLevel arProclvl = new ArProcessingLevel()
50         {
51             Alarm = ProcessingLevel.Auto,
52             Connect = ProcessingLevel.Auto,

```

## Development of user testcases

# Implementing the ITestcase interface

- Testcases have to implement the ITestcase interface
- Members:
  - Name: Name of the Testcase
  - Id: Id of the Testcase as shown in the report.
  - Category: Location in TC tree
  - TestSetup (Standard, P2P, SystemRedundancy, ...)
  - NeededDevices: Will be checked while TC preparation.

```

0 Verweise
public class CertificationDemo : ITestcase
{
    Mehr als 99 Verweise
    public String Name { get; }
    90 Verweise
    public Int32 Id { get; }
    Mehr als 99 Verweise
    public String Category { get; }
    92 Verweise
    public String Version { get; }
    91 Verweise
    public String[] Description { get; }
    Mehr als 99 Verweise
    public TestSetup TestSetup { get; }
    Mehr als 99 Verweise
    public TestcaseReport Report { get; }
    93 Verweise
    public UInt32 EstimatedExecutionTime { get; }
    93 Verweise
    public List<TestSetupComponents> NeededDevices { get; }
    73 Verweise
    public void Start(RtTesterModuleBase rtTesterModuleBase, BackgroundWorker backgroundWorker)
    {
        throw new NotImplementedException();
    }
}
  
```

Development of user testcases



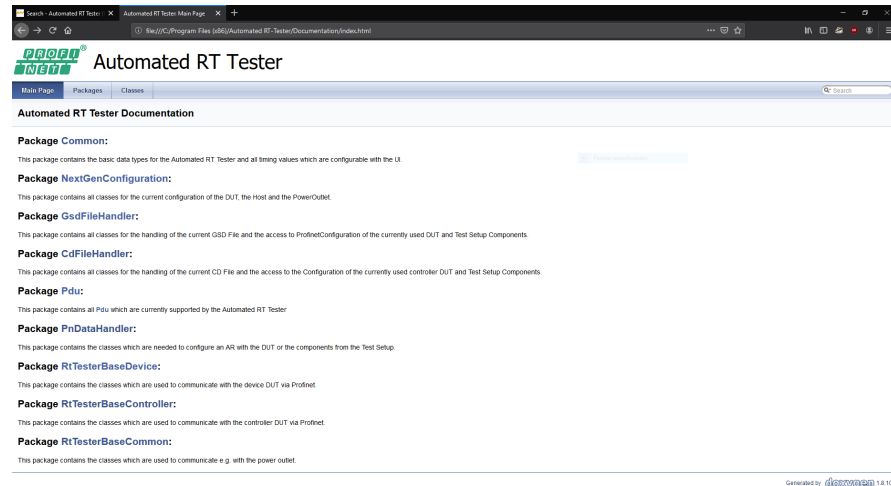
- Start()-Method
  - Will be executed on TC start
  - Holds or manages the TC logic
- rtTesterModuleBase
  - Provides access to all data regarding configuration of DUT, network communication and PowerOutlet of the test setup
- backgroundWorker
  - Cancellation of running TC by the user

```

0 Verweise
public class CertificationDemo : ITestcase
{
    Mehr als 99 Verweise
    public String Name { get; }
    90 Verweise
    public Int32 Id { get; }
    Mehr als 99 Verweise
    public String Category { get; }
    92 Verweise
    public String Version { get; }
    91 Verweise
    public String[] Description { get; }
    Mehr als 99 Verweise
    public TestSetup TestSetup { get; }
    Mehr als 99 Verweise
    public TestcaseReport Report { get; }
    93 Verweise
    public UInt32 EstimatedExecutionTime { get; }
    93 Verweise
    public List<TestSetupComponents> NeededDevices { get; }
    73 Verweise
    public void Start(RtTesterModuleBase rtTesterModuleBase, BackgroundWorker backgroundWorker)
    {
        throw new NotImplementedException();
    }
}
    
```

Development of user testcases

- Docs for developers:
  - The latest version of the documentation is always part of a bundle.



## Documentation for developers

- MSF-Vathauer: SME with a broad product spectrum for frequency converters, drives etc. for automation.
  - Latest project: Frequency converters and soft-starter with PROFINET connection.
- Complete test tool for development, pre-certification and testing within assembly based on the Automated RT-Tester Framework.
- Complete independency from other software or IO-Controllers.



## End of part 2

Upcoming: Live demo (structure and contents of the bundle, project setup, test execution)

## Part 3: Live demo

Structure and contents of the Test Bundle

Project Setup for Device & Controller Certification

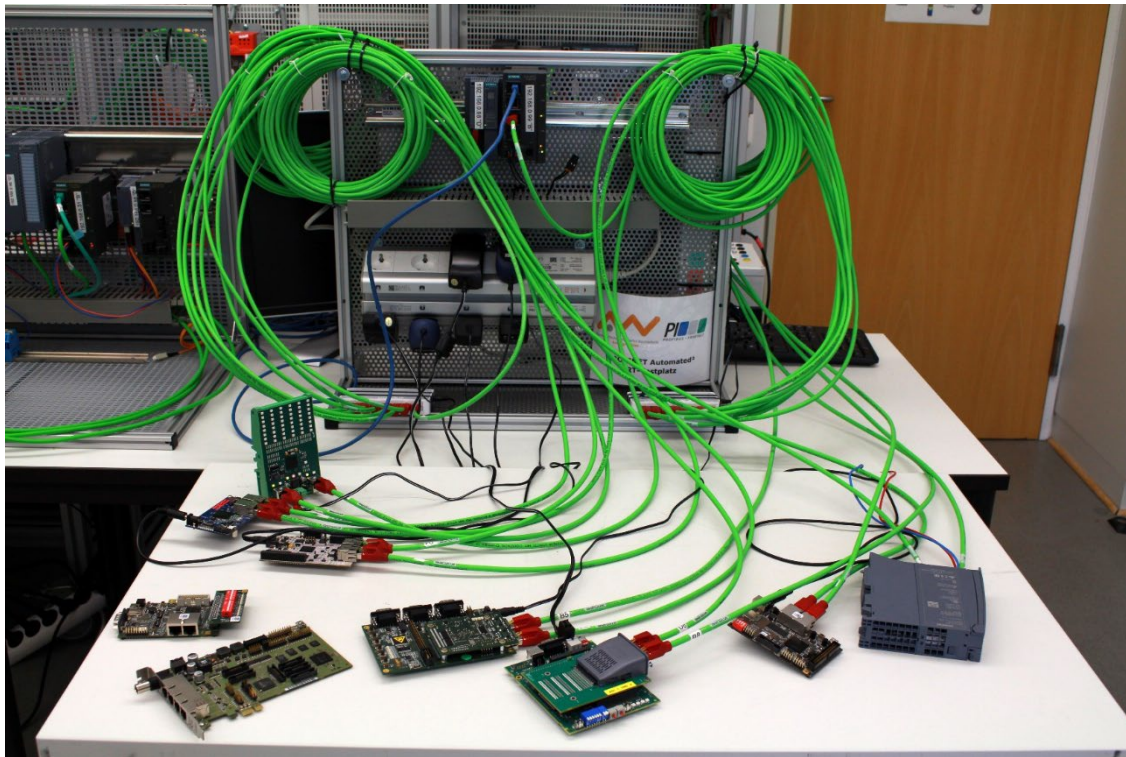
Test execution

- 10 Standard Testsetups
- 5 Unified Testsetups
- More than 40 PROFINET Devices of a wide range of manufacturers
- PROFINET Controllers of major manufacturers



OTH Testbed





**Leuze electronic**  
the **sensor** people

**WAGO**®

**port**  
PROFESSIONAL TOOLS

**BECKHOFF**



**EH**

Endress+Hauser

**MURR**  
ELEKTRONIK

**FESTO**





- aia – automations institute GmbH is an affiliated institute at OTH Amberg-Weiden.
- Headquarters on the Technology Campus at the OTH Amberg-Weiden.
- Software solutions for small and medium-sized enterprises and industry in the field of automation.



aia offices at OTH campus in Amberg

**Thank you!**

Do you have any questions?

b.etzold@oth-aw.de | pn-ts-support@oth-aw.de



Ostbayerische Technische Hochschule  
**Amberg-Weiden**

