Agenda

Part 1: Introduction to Siemens PTD EA
- What are the products of Siemens PTD EA
- What are test cases for these products

Part 2: Distributed testing with TTCN-3

Part 3: The Test Setup and the Results
- Test purpose
- Using TTCN-3 distributed testing
- Results
- Extension of TTCN-3 for Testing of communication in process automation
Agenda

Part 1: Introduction to Siemens PTD EA
  ▪ What are the products of Siemens PTD EA
  ▪ What are test cases for these products

Part 2: Distributed testing with TTCN-3

Part 3: The Test Setup and the Results
  ▪ Test purpose
  ▪ Using TTCN-3 distributed testing
  ▪ Results
  ▪ Extension of TTCN-3 for Testing of communication in process automation

Protection systems are essential for a safe and reliable energy system

Generation
  • Generator protection
  • Transformer protection

Transmission
  • Distance protection
  • Line protection
  • Feeder protection
  • Busbar protection

Distribution
  • Transformer protection
  • Line protection
  • Overcurrent protection
As in other industries, digitalization totally changed the way protection devices work.

Moving Coil devices

Analogue devices

Digital devices


Today, energy management relies on communication technology.
A typical test case: the protection setup for a busbar

When a fault occurs, the protection device A turns off the power switch

**Procedure:**
- A fault occurs, e.g. due to strike of lightning
- A fault current flows
- Device A detects the current
- Device A sends a command to the power switch after 50ms
- The power switch is opened
- The current flow is disrupted
An external fault should not lead to switching off the power switch assigned to device A

**Procedure:**
- A fault occurs e.g. due to strike of lightning
- A fault current flows
- Device A detects the current
- Device B detects the current
- Device B sends a blocking command to device A
- Device A does not switch off the power switch
- Device B switches off the power switch immediately

The timing is very important when a blocking scheme is used for protection

**Timing:**
- Detection of a fault current: *approx. 20ms*
- Switch-off of external faults: *0 ms after detection*
- Switch-off of busbar faults: *50 ms after occurrence*

The blocking signal has to be sent, transmitted, and received within a window of 30 ms

In our test we want to show that this happens even under heavy load.
A lot of communication sources should be used simultaneously to perform a load test

Test case description:

- Generate a fault current
- Within 30ms after fault occurrence:
  - Send a T103 telegram
  - Send an http request
  - Send an operational request

In order to perform these tests:
- A distributed test system with 3 test machines is needed
- The test machines have to be synchronized

Summary

- Load testing of a blocking scheme with two protection devices

  Challenges:
  - Sending messages to a system under test from 3 test computers
  - The test computers have to be synchronized with each other
  - The analogue signal generator has to be synchronized with the test machines
  - The test machines have to support different protocols

  Our “synchronicity” requirement for this scenario is: “Within an interval of 10 ms”!
Agenda

Part 1: Introduction to Siemens PTD EA
- What are the products of Siemens PTD EA
- What are test cases for these products

Part 2: Distributed testing with TTCN-3

Part 3: The Test Setup and the Results
- Test purpose
- Using TTCN-3 distributed testing
- Results
- Extension of TTCN-3 for Testing of communication in process automation

Dynamic Test Configurations in TTCN-3

![Diagram showing dynamic test configurations in TTCN-3]
Agenda

Part 1: Introduction to Siemens PTD EA
- What are the products of Siemens PTD EA
- What are test cases for these products

Part 2: Distributed testing with TTCN-3

Part 3: The Test Setup and the Results
- Test purpose
- Using TTCN-3 distributed testing
- Results
- Extension of TTCN-3 for Testing of communication in process automation

Test purpose

The blocking signal has to be transmitted within 30 ms even if it is stressed by max. 5 clients which connect "simultaneously" to the protection device B (server)

Simultaneousness means that all messages "connect" from all clients have to be transmitted within a time window of 10 ms.

Using TTCN-3 distributed testing where 3 clients are simulated and deployed on 3 PCs
Challenges and Solution

**Challenges:**

- Test adapters for communication protocol IEC 61850 not available
- All Components of a distributed test system have to be timely synchronized
- Synchronization of send operations at 3 Clients in TTCN-3 is not successful. Absolute time is needed
- Real-Time behavior not available (Windows/JAVA)

**Solution:**

- Programming the Test adapters
- Time Synchronisation with GPS
- Extention of send operation in system adapter with synchronization methods. Absolute time of operation system
- It’s not good, but can be acceptable for the given test purpose

---

Synchronization of the send operation at all 3 Clients

The thread execution is stopped until the send-out time (GPS time) is reached.
The code snippet demonstrates how the synchronization is programmed in TTCN-3

```plaintext
def function DistributedConnect() runs on MTCTYPE{
    ConnectIECDevice(Connect: {IP_ADDRESS, 102, 55000});
    if (getverdict() == fail)
        DisconnectIECDevice();
}

type component MTCTYPE

testcase distributed() runs on MTCTYPE TestComponent{
    var MTCTYPE Component1, Component2, Component3;
    Component1 := MTCTYPE.create;
    Component2 := MTCTYPE.create;
    Component3 := MTCTYPE.create;
    map(Component1: iec61850, system: iec61850);
    map(Component2: iec61850, system: iec61850);
    map(Component3: iec61850, system: iec61850);
    Component1.start(DistributedConnect());
    Component2.start(DistributedConnect());
    Component3.start(DistributedConnect());
    all component.done;
}
```

In another setup the delay between three synchronized test computers was measured

**Test description:**

- At a determined time all three test computers send a message to the protection device
- The protocol analyzer records all three messages

In this test we measured the delay between:
- The first and
- The last arriving message
The measurements show that the synchronicity of the distributed tests meets our requirements.

**Result:**

- The average difference between the arriving messages was 2.6 ms.
- No test showed a difference of more than 8 ms.

The synchronicity requirement "Within an interval of 10 ms" is fulfilled.

---

**Requirements of distributed testing in energy automation (process automation):**

- All test component can process absolute time which is needed for processing received informations (indication or measured valued).
- All test component and the SUT are time synchronized.
- All operations of a test component are processed in real-time. That means they are processed in a deterministic time window.
**Our Modification of TTCN-3 test system architecture**

- Extension of the system adapter (SA) so that it can process absolute time which is retrieved from the GPS Adapter by using an API
- Extension of the system adapter (SA) with a synchronization operation

---

**What we expect from TTCN-3 distributed testing**

- The Platform Adapter (PA) should offer operations for handling absolute time
- The Component Handler (CH) should support time synchronization or test system architecture should be extended by a time synchronization interface
- For high performance real-time distributed testing all operations of the TTCN-3 test system should have deterministic time behavior. This can be realized by using real-time operation system (e.g. Solaris SPARC) and real-time programming (e.g. JRTS) for developing TTCN-3 test system
Summary

Synchronization of three TTCN-3 test computers with GPS – Results

- TTCN-3 distributed testing was successfully applied
- TTCN-3 test system architecture was modified for time synchronization and processing absolute time
- The test application fulfills the requirements
- TTCN-3 should support absolute time and time synchronization
- High performance real-time distributed testing doesn’t have to be standardized but it can be implemented if necessary

Questions?

Thank you for your attention!