

System Testing of Protection Devices and Schemes – What Is It and Why Do We Need It?

A. P. APOSTOLOV
OMICRON electronics
USA
alex.apostolov@omicronusa.com

KEYWORDS

System testing, protection IEDs, protection schemes

1 INTRODUCTION

The IEC 61850 standard for communication networks and systems in substations allows the development of high-speed peer-to-peer communications based distributed protection applications that result in significant changes in the ways protection functions are implemented. This replacement of functions implemented in a single device with equivalents using exchange of analog and status information over the substation local area network requires new technology also for their testing

The paper describes in detail the principles of different IEC 61850 distributed applications and systems and analyses the factors that will affect their performance. The definitions of the individual components of distributed systems are presented in detail, including the different possible allocations of sub-functions and functional elements in physical devices. Each complex function actually represents a system with a different level of complexity depending on its design.

Definitions of function boundaries and methods for testing dependent on the purpose of the test are then described. Type, acceptance, commissioning and maintenance testing are considered.

It later discusses the requirements and principles of their testing. A comparison between the functional testing of conventional devices and testing of communications based Intelligent Substation Devices (ISD) is described later in the paper.

The paper then discusses some examples of distributed applications based on GOOSE messages and sampled analog values from the point of view of the requirements for their testing.

Methods and tools for functional testing of distributed IEC 61850 based systems are presented at the end of the paper. Full, partial and hybrid implementations of IEC 61850 are analyzed. The different steps in the testing process are described. They include the configuration, simulation, operation detection and results analysis.

The impact of non-protection related events on the performance of distributed functions and how it can be covered in the test process is presented at the end of the paper.

2 WHAT IS SYSTEM TESTING?

In order to properly define the methods for testing of complex IEC 61850 substation automation systems it is important to properly define what a system definition is and to consider what existing methods for system testing are known. Complex systems are not specific to the electric power systems domain only. They exist in industry, communications, computing and many other fields. Software development can be considered the development of complex systems that exchange information between different functional modules. Modern substation automation systems in reality are complex distributed software applications based on exchange of information over the substation local area

network. That is why there are significant similarities between the testing of complex software tools and substation automation systems.

IEC 61850 defines a system as “The logical system is a union of all communicating application-functions performing some overall task like “management of a substation”, via logical nodes. The physical system is composed of all devices hosting these functions and the interconnecting physical communication network. The boundary of a system is given by its logical or physical interfaces. Within the scope of the IEC 61850 series, ‘system’ always refers to the Substation Automation System (SAS), unless otherwise noted”.

This is not very far from an abstract definition of a system as a group of interacting, interrelated or independent elements forming a complex whole. Each component of a system is interacting or related to at least one other component/element. Any object which has no relationship with any other element of the system is obviously not a component of that system.

Depending on the complexity of the system, its components can be simple functional elements, subsystems or combination of the two. A subsystem is then defined as a set of elements, which is a system itself, and also a part of the whole system.

In the substation protection and automation domain we can consider different functions performed by the system as subsystems. The hierarchy of a complex system is shown in Figure 2 as a UML diagram.

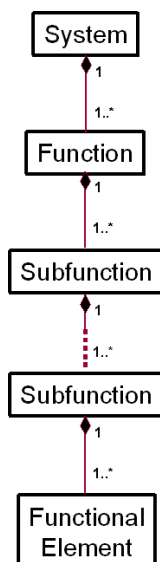


Fig. 1: System hierarchy UML diagram

From Figure 1 it can be seen that the system can contain 1 to many functions, that can have several layers of 1 to many subfunctions and at the bottom – a subfunction can contain 1 to many functional elements. The functional elements correspond to the IEC 61850 logical nodes.

System testing is testing conducted on a complete, integrated substation automation system, subsystem or distributed function. Its goal is to evaluate the system's compliance with its specified requirements.

System testing falls within the scope of Black Box Testing. This means that the test system does not have to have any knowledge of the internal logic and the behavior of the different subsystems or functional elements included in it.

System testing can be performed in a top-down or bottom-up approach. This is to a great extent dependent on the purpose of the test. If the test is a factory acceptance test it might be a good idea to use the bottom-up approach. In this case the testing starts first with the individual parts of the system – the functional elements. They are then grouped together to form sub-functions or functions, which are in turn linked into more complex functions until the complete system is tested.

When we do commissioning or maintenance testing we assume that the individual functional elements are operating properly, especially if there are no alarms in any of the IEDs that are included in the system test. In this case a top-down approach is suitable, since we are interested in the overall

performance of the tested system function and not in the behavior of the components of the system. Most of the time it fits the Black Box approach, which means that we take an external perspective of the test object to derive the test cases and analyze the results.

Functional testing of any function or subfunction requires from the test designer to select a set of valid or invalid inputs and determine the correct expected output for each test condition defined in the test plan. This will serve to define the evaluation criteria to determine if the test result is PASS or FAIL.

The purpose of functional element testing is to determine if the tested element has the expected behavior under different realistic test conditions. The functional elements in system testing are considered units, i.e. the smallest components of the system that have visible interface and defined behavior. From the testing point of view we can say that a unit is the smallest testable part of any system.

Integration testing is used to detect any potential interoperability problems between the functional elements and/or subfunctions that are integrated together in a function or a system. It not only tests the performance of the system, but also observes the exchanges between the different components being integrated into a system.

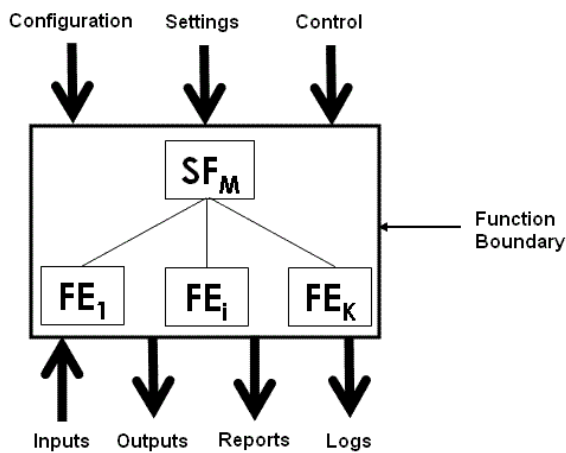


Fig. 2: Function boundary definition

System testing looks at the overall performance of the system from an external observer point of view.

In the top-down testing model the system is defined as a whole with its boundaries and behavior, without considering the details for any part of it. Each sub-part of the system then can be tested using the same approach until we get to the bottom of the functional hierarchy where we perform the functional elements testing.

In bottom-up testing we start with the functional elements testing and then go up the functional hierarchy testing sub-functions until we finish with the overall system testing.

In all cases it is important to clearly identify the system or function boundary that will define the requirements for simulation by the test system and monitoring the behavior of the tested function.

In Figure 2 above SF indicates a subfunction that contains K functional elements. The functional elements are the smallest component in the system that can be defined with a function boundary, interface and behavior, i.e. that can be tested.

All the above described principles can be used in the testing of IEC 61850 based substation protection and automation systems.

3 FUNCTIONAL TESTING METHODS

Functional testing methods can be divided into several categories. They are related to the complexity of the functionality of the individual devices being used in the different levels of the hierarchical system, as well as the types of distributed functions implemented in it.

The following are the more commonly used testing methods:

- Functional element testing
- Integration testing
- Function testing
- System testing

A function in this case can be considered as a sub-system with different level of complexity, for example a system monitoring function, while the system is the complete redundant scheme.

Regardless of what is being tested, the test object needs to meet the requirement for testability. This is a design characteristic which allows the status (operable, inoperable, or degrade) of a system or any of its sub-systems to be confidently determined in a timely fashion. Testability attempts to qualify those attributes of system design which facilitate detection and isolation of faults that affect system performance. From the point of view of testability a functional element in a scheme is the unit that can be tested, because it is the smallest element that can exist by itself and exchange information with its peers in the scheme.

Another consideration is the purpose of the test and needs to clarify if the tests are performed in relation to acceptance of a new product or function to be used as a system monitor or process controller (or both), the engineering and commissioning of a substation component or the complete scheme or its maintenance. From that perspective different testing methods can be implemented even in the testing of the same functional element or function.

For example the testing of a function during the user acceptance phase may focus on the testing of the measuring element characteristic using search test methods, while during the commissioning the operating times for different system conditions be the important ones achieved through transient simulation methods.

The knowledge of the internal behavior of the test object or more specifically the logic or algorithms implemented determines how the tests are being executed. The most commonly used test methods from this point of view are:

- Black box testing
- White box testing

An important aspect that needs to be considered during the testing is the availability of redundant devices performing the different scheme functions.

The following sections discuss in more detail the different testing methods listed above.

3.1 Black Box Testing

Black Box Testing is a very commonly used test method where the tester views the test object as a black box. This means that we are not interested in the internal behavior and structure of the tested function.

In the case of black box testing the test system is only interested in finding conditions under which the test object does not behave according to its specifications. Test data are derived solely from the specifications without taking advantage of knowledge of the internal structure of the function.

Black box testing is typically used for:

- functional elements testing
- scheme factory testing
- scheme site acceptance testing

Since functional elements are defined as units that are the smallest that can exist independently and are testable, it is clear that black box testing is the only method that can be used for their testing.

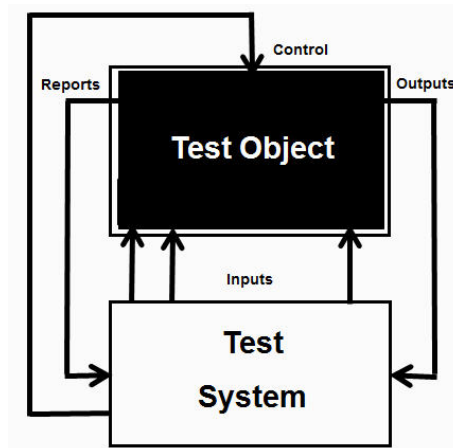


Fig. 3: Black Box Testing

The response of the test object to the stimuli applied to the test object inputs can be monitored by the test system using the operation of physical outputs, communications messages or reports.

3.2 White Box Testing

White box testing is a method where the test system is not only concerned with the operation of the test object under the test conditions, but also views its internal behavior and structure.

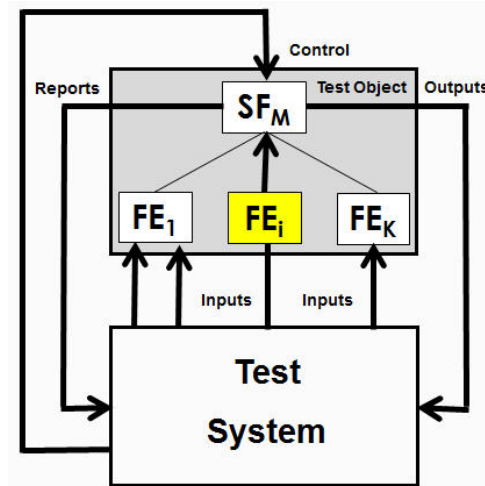


Fig. 4: White Box Testing

In the case of SCHEME it means that it will not only monitor the operation of the system at its function boundary, but also monitor the exchange of signals between different components of the system.

The testing strategy allows us to examine the internal structure of the test object and is very useful in the case of analysis of the behavior of the test object, especially when the test failed.

In using this strategy, the test system derives test data from examination of the test object's logic without neglecting the requirements in the specification. The goal of this test method is to achieve high test coverage through examination of the operation of different components of a complex function and the exchange of signals or messages between them under the test conditions.

This method is especially useful when we are testing distributed functions based on different logical interfaces. The observation of the behavior of the sub-functions or functional elements is achieved by the test system through monitoring of the exchange of messages between the components of the test object.

The test scenarios however do not have to be different from the ones used under black box testing.

3.3 Top-down Testing

Top-down testing is a method that can be widely used for a scheme, especially during site acceptance testing, when we can assume that all the components of the system have already been configured and tested.

Top-down testing can be performed using both black box and white box testing methods.

The testing starts with the complete system, followed by function or sub-function testing and if necessary functional element testing.

In the case of factory acceptance testing, when not all components of a system or sub-system are available, it is necessary for the test system to be able to simulate their operation as expected under the test scenario conditions. In this case the test system creates the so called Stubs for functions or functional elements that are not yet available.

Top-down testing results in re-testing of higher level elements when new lower level elements of the system are added. The adding of new elements one by one should not be taken too literally. Sometimes a collection of elements will be included simultaneously, and the whole set of elements will serve as test harness for each functional element test.

Each functional element is tested according to a functional element test plan, with a top-down strategy.

A testing stub is a module which simulates the operations of a module which is invoked within a test. The testing stub can replace the real module (for example a line monitor) for testing purposes.

The testing of the individual components of a system function might be required in the case of failure of a specific test, which is shown in Figure 5. The function boundary for each of these tests will be different and will require a different set of stimuli from the test system, as well as the monitoring of the behavior of the functional elements using different signals or communications messages.

For example if Test 1 (see Figure 5) of the complete scheme fails, the user needs to start testing subfunctions down the scheme functional hierarchy. If any of these Test 2 level tests fails, then Test 3 level tests need to be performed, until eventually a failure of a function element at the bottom of the hierarchy is detected.

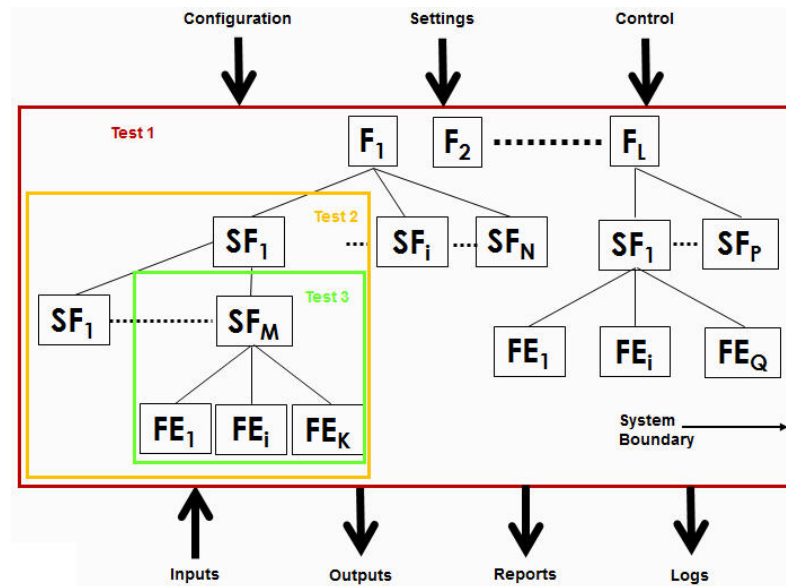


Fig. 5: Top-down testing of a system monitoring function

3.4 Bottom-up Testing

Bottom-up testing is a method that starts with lower level functions – typically with the functional elements used in the system – for example FE1, FEi, etc..

This method is more suitable for type testing by a manufacturer or acceptance testing by the user.

When testing complex multilevel functions or systems, driver functional elements must be created for the ones not available. The test system must be able to simulate any missing component of the system when performing for example factory acceptance testing.

There are many similarities in the test scenarios used in the bottom-up, compared to the top-down method. The main difference between the two methods is the order that the tests are performed and the number of tests required.

TESTING OF IEC 61850 SYSTEMS

The method for testing of both types of systems is proposed based on the following order of system components tests:

- Testing of IEC 61850 protocol compliance of the individual components of the system
- Testing of Merging Units
- Testing of IEC 61850 compliant IEDs
- Testing of bay level distributed applications
- Testing of substation level distributed applications

The goal of conformance testing is to ensure that IEC 61850 with all its models and services is properly implemented. The test procedures and tools used are based on the definitions in Part 10 of the standard. This improves the chances for interoperability between the individual devices integrated in the system.

IEC 61850 SYSTEM TESTING

A test system designed for IEDs or distributed systems based on IEC 61850 have multiple components that are needed for the testing of the individual functions, as well as a complete application. A simplified block diagram of such a system is shown in Figure 6.

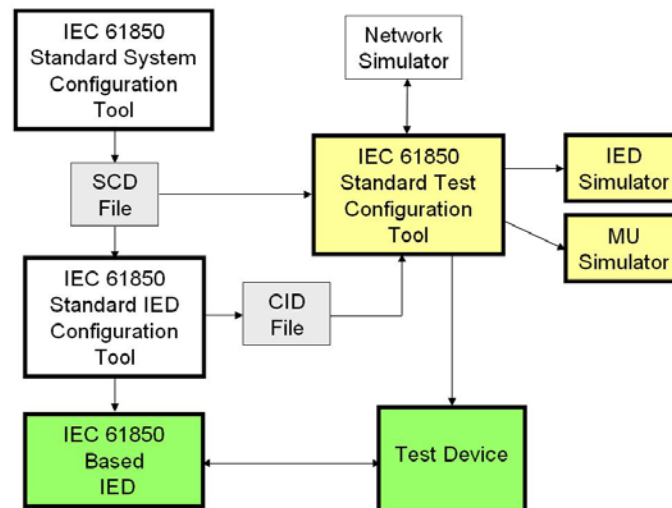


Fig. 6: Test system/configuration tool, simplified block diagram

The first component of the test system is the test Configuration Tool. It takes advantage of one of the key components of the IEC 61850 standard – the Substation Configuration Language. The Configuration Tool is used to create the files required for configuration of different components of the test system. It imports or exports different configuration files defined by Part 6 of IEC 61850.

The test system Configuration Tool reads the information regarding all IEDs, communication configuration and substation description sections. This information is in a file with .SCD extension (for Substation Configuration Description) and is used to configure the set of tests to be performed.

The overall functionality of any IEC 61850 compliant device is available in a file that describes its capabilities. This file has an extension .ICD for IED Capability Description.

The IED configuration tool sends to the IED information on its instantiation within a substation automation system (SAS) project. The communication section of the file contains the current address of the IED. The substation section related to this IED may be present and then shall have name values assigned according to the project specific names. This file has an extension .CID (for Configured IED Description).

The second component of such a system is a Simulation Tool that generates the current and voltage waveforms. The specifics of each simulated test condition are determined by the complete, as well as the configured functionality of the tested device or application.

The simulation tool requirements will also be different depending on the type of function being tested. For example, if the tested function is based on RMS values or phasor measurements, the simulation tool may include a sequence of steps with the analog values in each of the steps defined as Phasors with their magnitude and phase angle. Based on these configuration parameters the simulation tool will generate the sine waveforms to be applied as analog signals or in a digital format to the tested components or systems.

4 CONCLUSIONS

System testing requires a very good understanding of the functionality and the hierarchy of the protection, automation and control system.

The definition of a function or a system boundary and the behavior of the test object is essential for the proper definition of the different test cases, the testing system and the evaluation of the performance during the testing.

Depending on the purpose of the test, different methods can be used:

- Bottom-up
- Top-down
- Black box
- White box

Testing of an IEC 61850 based system adds another dimension and requires the availability of a test system that can support both GOOSE and sampled values publishing and subscriptions based on SCD files for its configuration.