

CENTRALIZED TESTING AND VALIDATION FOR POWER ELECTRONICS EQUIPMENT

Monica POPA¹, Florin IONESCU²

Această lucrare prezintă o metoda de a centraliza controlul testarii și validării unor echipamente industriale complexe de electronică de putere. Metoda propune efectuarea și raportarea testelor într-o manieră automată, permițând intervenția operatorului uman pentru parametrizare și setare a intrărilor de test. Sunt prezentate sistemul de achiziții de date și software-ul construite pentru această aplicație, precum și rezultatele experimentale. Este evidențiat și modul în care programul software se integrează ca modul într-o aplicație de planificarea resurselor într-o întreprindere, respectând cerințele de proiectare și performanță ale unui sistem informatic critic pentru afacere. Concluziile finale menționează alte aplicații și posibilități de dezvoltare ulterioară.

This paper presents a method of centralizing control and reporting of testing and validations on industrial power electronics equipment of high complexity. The method provides automated test performing and reporting, while allowing the human intervention in parameterization and establishment of test entries. The hardware data acquisition infrastructure and software tools built for this application are described, as well as experimental results. The paper shows how this software integrates as a module within a broader enterprise resource planning application, following the design and performance requirements imposed by business critical software. The conclusions include other possible applications and further development.

Keywords: testing, validation, software, data acquisition, power electronics, enterprise resource planning

1. Introduction

Manufacturers of electronic equipment have been facing the conundrum of performing reliable testing in a time and cost-efficient manner for a long time [1]. The challenges in this area are diverse. They lie, in some cases, in the complexity of electronic products; in other areas, the testing process needs to be flexible and easily adaptable to newly designed products. Some manufacturers are constrained to reduce manpower allocated to testing.

¹ PhD student, Dept. of Electrical Engineering, University POLITEHNICA of Bucharest, Romania, e-mail: adinamonicapopa@yahoo.com

² Prof., Dept. of Electrical Engineering, University POLITEHNICA of Bucharest, Romania

Automatic test equipment (ATE) is generally used to solve this problem [3][4]. ATE is a generic term for a system that is used to test electronic devices (known as devices under test), and that uses automation to perform the tests or evaluate the test results, or both. A wide range of ATEs have been developed for different purposes. They have evolved from specialized systems for particular tests to a wide range of applications in various fields of the electronics industry. Lately, most ATEs are computer controlled. Some have the ability to perform complex tests and are accompanied by specialized software which adds capabilities of processing, visualizing, storing and exporting the test data.

For high complexity equipment in power electronics, automatic testing systems are built using a combination of ATE tools and custom written test scripts. Programming of tests requires knowledge of the specific proprietary languages or interfaces provided with the test instruments [5].

Integrated hardware and software solutions that fit certain applications are available from some vendors [15]. Automatic testing systems that are built for specific equipment usually have a high acquisition price, and more than one type may be required to cover all the needs of a power electronics manufacturer. They require configuration and are therefore more cost-effective if used for high-volume testing, on equipment produced in large series. For complex equipment that is produced in small series, many manufacturers prefer in-house test setups.

The purpose of the present study was to create a cost-effective, easy to use and extensible solution of automating the verifications and processing the test results for power electronics complex equipment. The solution was developed with a standard data acquisition system (hardware and device driver software) and a novel software module built for this application. A novelty factor is that the software is separated and independent from the equipment under test and the acquisition system. Thus, the same software design can be applied to a variety of purposes. The test reports and performance charts, which are the final deliverables of this system can be easily integrated in a company's management software, are available at any time for visualization for a number of users and can be exported and printed in different formats.

2. Method Overview

This paper presents a method of automatically processing functional and performance test data for power electronics equipment of high complexity, such as static excitations, electrical supply and control for drilling, electric drives for various applications.

The equipment under test is connected to a data acquisition system that samples and converts analogical and digital signals and provides the digital data to the acquisition software installed on a personal computer. This setup is described

in more detail in chapter 3. The inputs from the power electronics equipment are voltages, currents and status indicators (normally-open and normally-closed contact).

The measurement data from the acquisition is fed to a custom built software system. For each type of equipment under test, this software stores a number of test entries that need to be completed in order for the equipment to be declared valid according to the technical specifications. The names and descriptions of the test entries are stored in a readable format in a relational database. The conditions that need to be fulfilled for each test are defined in the code. Measured data from the acquisition system is stored in the database and compared against the test rule. The software decides if each of the tests has been passed after a round of measurements. Measurements over time are also used to create performance charts.

Some of the test results can be filled in by an operator directly in the application, in the case of visual verifications, for instance.

The final deliverable is a test report with performance charts, that can be viewed in the software's interface or exported and printed in other formats. Separate test reports can be built for type or lot testing, for tests that take place at the producer's facility or at the end user.

This software module is integrated, from the point of view of the presentation and the underlying design, with an enterprise resource management software (ERP). ERPs are software packages which are designed to integrate a complete range of business processes and function in an organization, in order to uniformly present the business from the perspective of a single information and IT architecture [11]. Among the advantages of this integration, we mention a shorter development cycle due to preexisting base libraries for data access and presentation, a single authentication and authorization framework, performance and scalability criteria defined for the ERP are extended to the testing and verifications module.

3. Data Acquisition

The output from the equipment under test reaches the processing software by means of a data acquisition system.

This comprises a custom card for obtaining unified signal (± 10 V), a data acquisition (DAQ) card produced by National Instruments, model 6025E [7], a personal computer with a PCMCIA slot and Windows XP operating system, LabVIEW software from National Instruments [8].

Analogical signals can be sampled and converted into digital values with the aid of data acquisition cards, in order to be manipulated by a computer program. By adding inputs to a personal computer, this can be transformed into a

high-performance and flexible testing system. The DAQ card used permits data acquisition on 16 channels of analog input and 32 lines of digital I/O. It uses the National Instruments DAQ-STC system timing controller for time-related functions [9]. The DAQ-STC consists of three timing groups that control analog input, analog output, and general-purpose counter and timer functions. The maximum aggregate timing resolution is 50 ns. The sampling rate can be seamlessly changed.

The acquisition card allows for analogical signals in the range of ± 10 V. The intermediary custom card is needed for signal scaling, which can be performed separately on each channel.

As driver software for the DAQ card, National Instruments provides NI-DAQ for PC compatibles, with extensive libraries to use in a programming environment. However, other options were available, that provide the same functionality, integrated in: LabVIEW, Measurement Studio and VirtualBench. LabVIEW was chosen for reasons of availability, the powerful graphical programming environment and the easy integration with external systems. The LabVIEW Data Acquisition VI Library is functionally equivalent to NI-DAQ software.

Although LabVIEW has many features and uses, we only employ the data acquisition capabilities. In a subsequent version, the software could make use the more lightweight NI-DAQ libraries for the data acquisition. Measurement data from the equipment under test is thus read through a LabVIEW graphical program, and exported in a tab separated format to be processed and stored by the software built for this application. This is presented in the next chapter.

4. Software Architecture

The architecture of the software module was designed to satisfy a number of goals:

- the testing reports and verification results must be generated automatically from two categories of data: obtained directly from the equipment under test, and manually inserted by a testing operator;
- the reports must be visible as they are being completed through a user-friendly interface;
- the application must be accessible from local and remote personal computers, within an intranet network;
- the system must ensure protection of data against unauthorized access; all access is subject to user identification and password control;
- the module must integrate seamlessly within an ERP software, and must inherit the security, reliability and performance constraints imposed on the larger architecture.

This software was built as a web-based application with a three-tier architecture:

- an application layer, written in object-oriented PHP, responsible for the interface logic, user interaction and data display processing,
- a business layer, also written in PHP, containing the data acquisition and storage logic, the testing and verification procedures,
- a relational database layer, for storing the testing raw data, test report templates and results.

From a use-case perspective, users that are logged can visualize test reports, fill in the information that is required from the test operator, create a new test report and initialize the automated verification process. The possible action flows are presented in figure 1. Authorization is necessary for each view and edit action.

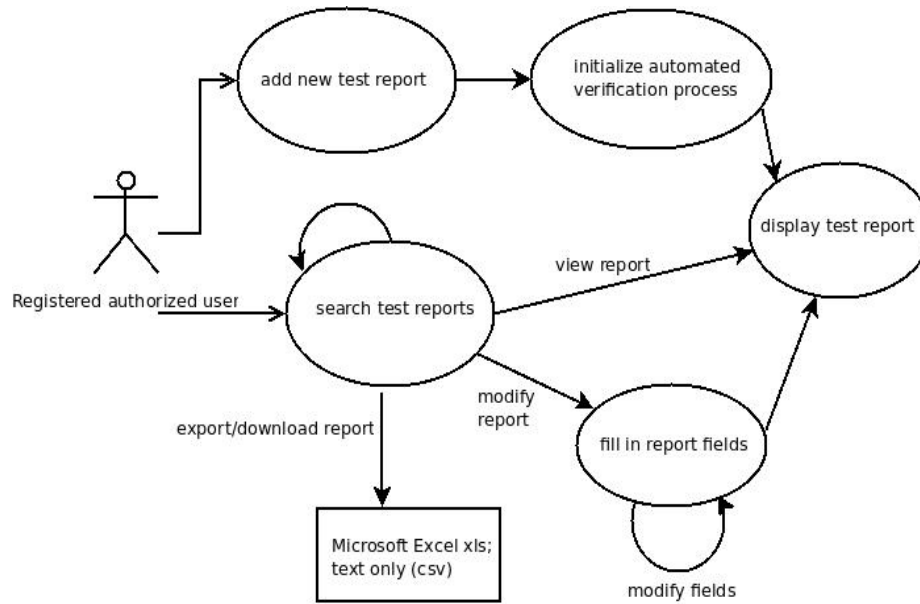


Fig. 1. Use-case view on the testing module for a user with full rights

Access to this module's interface is regulated via a permissions system used in the whole ERP software. User accounts are defined for the ERP application as a whole, and user roles are defined to cover different functionalities spanning one or more modules. Application users are assigned any number of predefined roles, with certain rights on actions and interfaces.

For the verifications software, access is allowed via three predefined roles:

- viewer, that can only search and display test reports,

- operator, with viewer privileges, plus the permissions to create a new test report, initialize an automated verifications process, fill in data that is required to be manually inserted by the operator, export/download test reports in csv or xls formats,

- administrator, with operator rights, plus the privilege to modify report fields, receive alerts related to the failed tests, and summaries about the testing activities.

The implementation of this software application makes use of design patterns and principles used in the containing ERP application, actually extending base objects and using data access libraries that are applied by all modules in an ERP. In order for this module to function there is no need for the entire ERP to be installed or functional, only a core of base objects and utility libraries has to be present. The authentication and authorization database needs to be installed as well.

The application is built on the MVC (model-view-controller) architectural pattern [12][14]. This design isolates application logic from input and presentation logic (user interface), permitting independent development, testing and maintenance. The base MVC objects are supplied by the ERP core.

Data is stored in an open-source MySQL relational database, which contains the predefined test report forms, as well as the measurement results and the resulting test reports for all tested products. The test report forms are defined for each product category, but can have custom fields for each tested equipment, if necessary. The measurements are correlated with the test report forms, compared against the test conditions and the information of whether the equipment has passed a certain test is also stored in the database.

The integration with the company's ERP application brings a good visibility over the characteristics and technical documentation of all the product categories that can be tested. The status of the testing operations is shown in other areas of the ERP, such as progress of contracts, human resource allocation, reporting for management purposes.

5. Experimental Results

As an example, we show the test results from a static excitation equipment, a product used in power generation. This type of equipment ensures the supply, control, command and protection of the excitation winding of the generators in thermal and hydro electrical power plants. These generators provide the mains supply in the national network. By means of this equipment, the parameters of the controlled generator are maintained at specific values and the machine running is optimized.

The tests conducted on this product take place in two steps, the first one at the producer's facility and the second at the end user, where the equipment is connected to the generator, which is coupled at the mains. The test setup at the end user's facility is shown in figure 2.

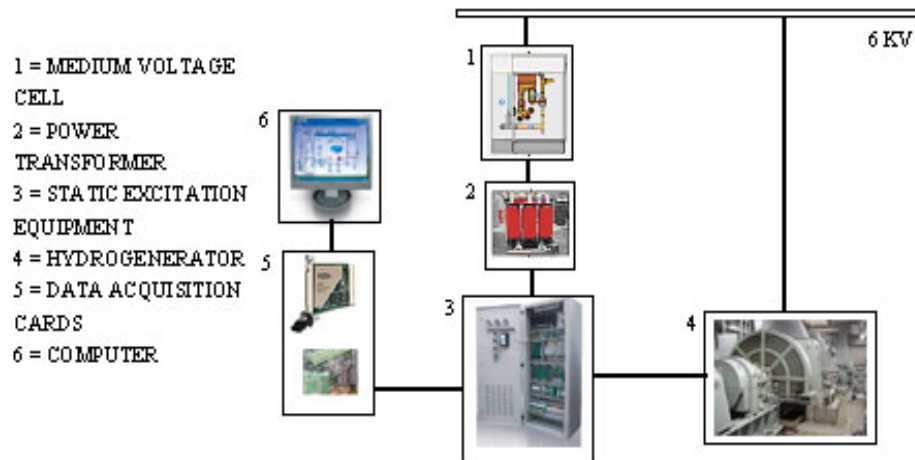


Fig. 2. The functional scheme at the end user's site

The tests confirm the good operating of the equipment by:

- maintaining the parameters at the preset values,
- enforcing the parameter limitations in the preset ranges,
- correct response to several command sequences,
- correct behavior at failure and breakdown simulations,
- the dynamic behavior of the specific parameters, with the equipment connected to the generator.

Among the performed tests, we mention a few operating tests for the digital automatic regime:

- measurement of the excitation current (preset value $\pm 1\%$),
- generator current control, in a range of 0.8 to 1.15 of the nominal generator current ($\pm 1\%$),
- the limitation of the excitation current at two times the nominal excitation current, at forcing regime,
- the limitation of the minimum excitation current in the range of 0.3 to 0.8 of the idle regime excitation current ($\pm 1\%$),
- the limitation of the output generator voltage at 1 to 1.2 of the nominal generator voltage.

These tests were performed in a hydroelectric power plant. A small excerpt of a resulting test report for a static excitation equipment is shown in figure 3.

TEST REPORT No./ Date								
PRODUCT: STATIC EXCITATION EQUIPMENT								
TECHNICAL SPECIFICATION SF No.4/2001								
TEST	UM	TEST RULE	TEST RESULT	TEST PASSED	TYPE TEST	LOT TEST	TESTED AT MANUFACTURER	TESTED AT END USER
Operation regime: AUTOMATIC - digital								
generator current (I_{gen}) control: $0,8+1,15I_{nom\ gen} \pm 1\%$	A	SF condition 2.4.2.2, method 4.4.2.2	241 - 347	yes	yes	yes	yes	yes
limitation of I_{exc} at $2I_{nom\ exc} \pm 1\%$, forcing regime	A	SF condition 2.4.2.2, method 4.4.2.3	70,5	yes	yes	yes	yes	yes
limitation of I_{exc} at $2I_{nom\ exc} \pm 1\%$, after forcing	A	SF condition 2.4.2.2, method 4.4.2.4	70,6	yes	yes	yes	yes	yes
limitation of $I_{min\ exc}$ in the range $0,3+0,8I_{exc\ idle} \pm 1\%$	A	SF condition 2.4.2.2, method 4.4.2.5	15,1 - 40,2	yes	yes	yes	yes	yes
limitation of $U_{out\ gen}$ at $1+1,2U_{nom\ gen} \pm 1\%$	kV	SF condition 2.4.2.2, method 4.4.2.6	10,09-12,1	yes	yes	yes	yes	yes

Fig. 3. Fragment of test report for a static excitation equipment

As an example for the dynamic performances of the equipment, we present in figure 4 the response of the generator voltage when the input is a step signal of $\pm 10\%$, the equipment operating in idle automatic regime. From 95% to 85% from nominal generator voltage, shown in the interval 15s -17s on the chart below, and from 85% to 95% from nominal generator voltage, the interval 24s – 26s. The voltage generators reach the new value in 3s, within the accepted time interval. The measured parameters are scaled according to the design setups.

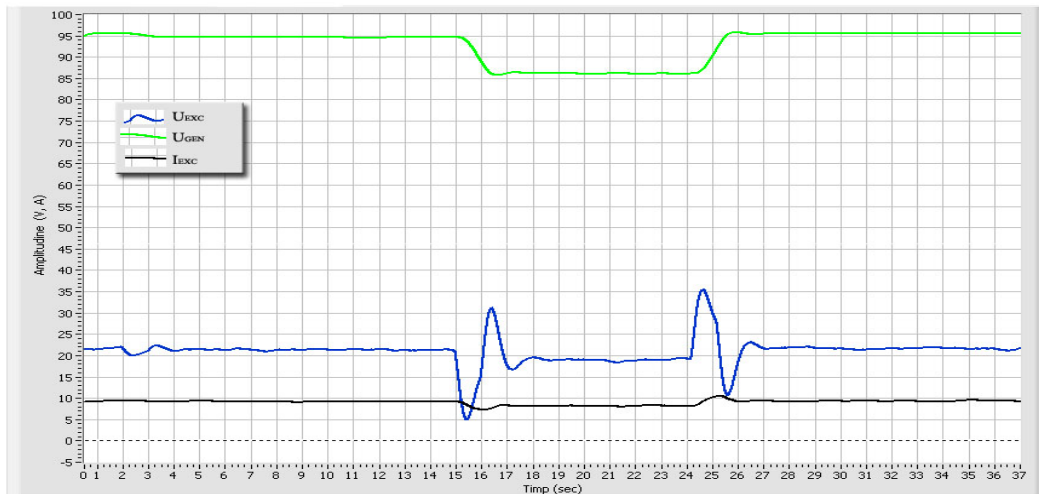


Fig. 4. Generator voltage response to a step signal, for a static excitation equipment

6. Conclusions

This paper has shown a new method of automating testing and validations for complex power electronics equipment. The novelty is the custom built software which is logically independent from the equipment under test or the data acquisition system.

The aim was to create an affordable solution from the point of view of the hardware and software setup, while on the same time being extensible to other product categories. Because of the separation from the equipment constraints, the software can be applied to a wide range of products, using the same or different data acquisition setups.

The integration with an ERP package brings easier development and maintenance, scalability, security and integrity of data, extensibility of features and of the presentation layer. Due to the associations that the ERP software brings, between the tested equipment and the product categories, contracting, human resources and logistics associated with the testing process, it provides a larger view of the quality assurance activities in an enterprise. For purposes of reporting or managerial survey, the testing software and data can be manipulated without difficulty.

Future directions of development may include replacing the software for data acquisition with custom built libraries, using the minimal acquisition driver libraries, which may also increase the performance of the solution. This would bring down the software costs and could permit better automated control of the testing process. From the point of view of the final results presentation, the software could be extended to generate different types alerts related to test exceptions, and the test report documentation could automatically be included in the contract documentation and the company's virtual library.

REFERENCES

- [1] *A.H. Parker*, Automatic testing of electronic systems, in Non-Destructive Testing, **vol. 1**, issue 1, August 1967, pages 25-29
- [2] *K.A. Bakshi, A.V. Bakshi*, Electronic Measurements, Technical Publications, 2008
- [3] *A.C. Stover*, ATE: automatic test equipment, McGraw-Hill Book Co., 1984
- [4] *K.F. Ibrahim*, Instruments and automatic test equipment: an introductory text book, Longman Scientific & Technical, 1988
- [5] *W. Fleischhammer*, Quality by design for electronics, Springer, 1996
- [6] *L.T. Wang, Y.W. Chang, K.T. Cheng*, Electronic Design Automation: Synthesis, Verification, and Test, Morgan Kaufmann, 2009
- [7] DAQ 6023E/6024E/6025E User Manual, National Instruments, Dec. 2000
- [8] LabVIEW User Manual, National Instruments, Apr. 2003
- [9] DAQ-STC Technical Reference Manual, National Instruments, May 1995
- [10] *D. Stanomir*, Semnale si sisteme analogice (Analogous Systems and Signals), Editura Politehnica, Bucuresti, 2002

- [11] *H. Klaus, M. Rosemann, G.C. Gable*, What is ERP?, in Information Systems Frontiers, **vol. 2**, no. 2, 2000
- [12] *T.M.H. Reenskaug*, MVC, XEROX PARC, 1978-1979
- [13] *E. Gamma, R. Helm, R. Johnson, J. Vlissides*, Design Patterns, Addison-Wesley, 1995
- [14] *J. Kargl*, PHP-MVC-frameworks- Evaluierung von Web Application Frameworks für die serverseitige Scriptsprache PHP, GRIN Verlag, 2009
- [15] National Instruments Web Site, <http://www.ni.com/products/>