ANL/AOD/CP-104609

AUTOMATED REAL-TIME TESTING (ARTT) FOR EMBEDDED CONTROL SYSTEMS (ECS)

Jon Hawkins, Group Leader, Interlock System & Instrumentation Group

Reginald B. Howard, Test Engineer Consultant

Haung V. Nguyen, Engineering Assistant, Interlock System & Instrumentation Group

Key Words: Testing Embedded Systems, Real-Time Testing, Automated Testing, Personnel Safety Systems

SUMMARY & CONCLUSION

In this paper, we presented an example of how Vali\Test Pro testing methodology was used to structure tests for a Personnel Safety System. We constructed validation matrixes to map PSS system requirements to their respective unit level of tests in order to measure the system's test coverage. We integrated a suite of four software-testing tools onto a single computer platform with a fifth tool working separately on a separate notebook computer. The four software modules that were integrated onto a Windows NT computer platform were: 1) Programmable Industrial Control Simulation (PICS) software which provided real-time Human Machine Interface (HMI) and Input/Output (I/O) simulation for testing programmable logic control (PLC) based systems; 2) WinRunner a functional testing tool provided a graphical user interface to Windows base applications; 3) Test Director provided test planning and works with WinRunner to provide graphical test planning, batch testing and defect tracking; and 4) RSLogic provided the means to down load software to Allen Bradley's PLC's and to verify faults and task states in Chain B of the PSS system. In addition, a fifth tool called State Logic from GE was setup to run on a separate notebook computer to verify faults and task states of Chain A of the PSS system. Working together, the five software tools created a testing platform that is capable of providing complete black box testing for the Personnel Safety System in real-time.

1.0 INTRODUCTION

Many of today's automated real-time testing systems for embedded systems were developed using expensive custom hardware and software. In this article we describe how to use commercially available off-the-shelf hardware and software to design and develop an automated real-time test systems for Embedded Programmable Logic Controller (PLC) Based Control Systems. Our system development began with the implementation of the VALI/TEST Pro testing methodology as a means for structuring the testing. Using this methodology, we were able to decompose system requirement documents for a Personnel Safety System (PSS) into its high, intermediate and detail level requirements. Next, the validation procedures for the PSS system were decomposed into testing units called builds, test runs and test cases. To measure the PSS system's test coverage three levels of system requirements were mapped to their respective unit level of test using a specially constructed validation matrix that was designed to handle over 150 test cases and requirements. All of the above work led to the development of an Automated Real-Time Test System (ARTTS) that is capable of performing complete black box testing in real-time for Embedded PLC Based Control Systems. Also note, that the PSS system under test and mentioned in this paper is located at the Advance Photon Source (APS) at Argonne National Laboratory Basic Energy Science Facility in Argonne, Illinois (see figure 1).

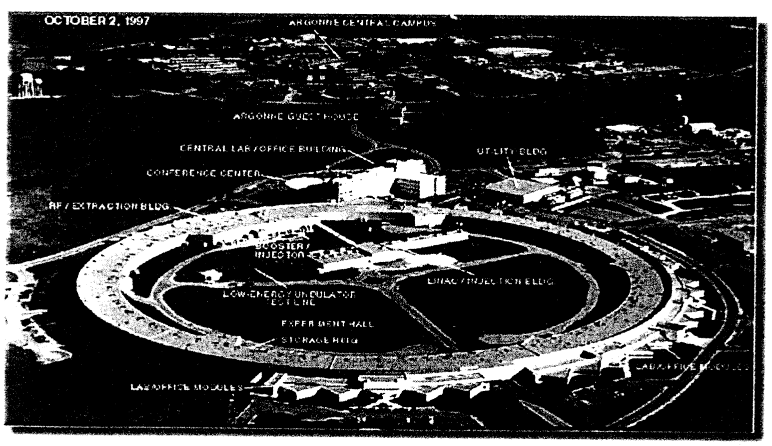

Fig. 1 APS facility

2.0 PSS SYSTEM OPERATION

In this section, we explain the theory of operation of a Personnel Safety System (PSS) for which the prototype ARTTS System was design to test.

2.1 PSS System Description

Consider figure 2, it depicts a typical configuration of "Station A", which is a major component of the overall PSS system. Note that the overall function of a PSS system is by definition, a highly reliable, fail-safe, redundant, stand-alone system that closely monitors and control personnel access into potentially hazardous Experimental Stations. It is also responsible for reducing hazards to mitigate harm to personnel against direct X-ray radiation from the Advance Photon Source.

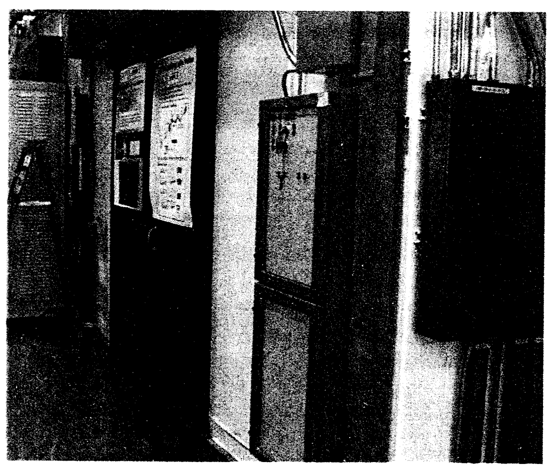

Fig. 2 Personnel Safety System (PSS)

2.2 PSS "Station A" User Panel Operation

Depicted in figure 3 is a typical layout of "Station A" user panels. Note that "Station A" consist of three panels 1) Station A user panel, 2) Station A door panel, and 3) System Controller panel.

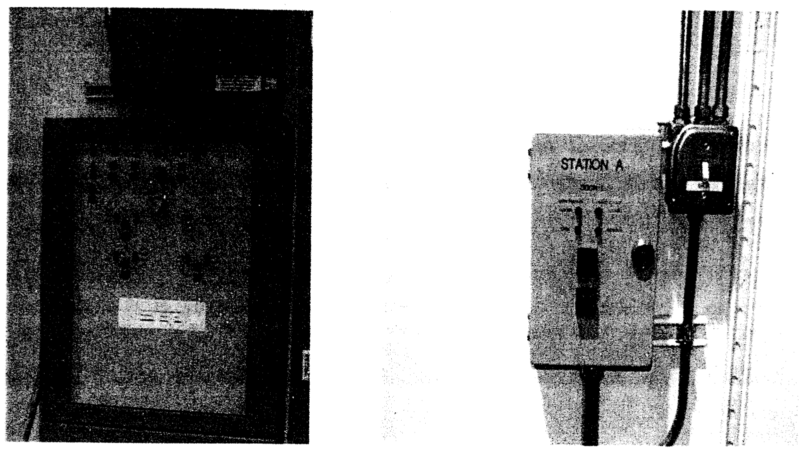

Fig. 3 PSS User Panels

3.0 TESTING METHODOLOGY

In this section, we describe how the PSS system requirements and validation procedures were structured using the Vali/Test Pro methodology. Note that the Vali/Test Pro is a methodology developed by Interim Technology Company and is widely used in the testing industry as a means to provide a visual approach to validating requirement coverage of a system or software application. This methodology provided us with the means to decomposed PSS system requirement documents (SAD, DOE, ES&H) into their high, intermediate and detail levels. In addition, we decomposed the validation procedures for one PSS beamline into testing units called builds, test runs and test cases (see fig. 4). Afterwards, we mapped each requirement level to its respective testing unit in a validation matrix in order to measure the test coverage. Figure 5 below shows an example of a high-level validation matrix with high-level requirements mapped to their respective testing levels. Below we defined the different requirement levels and testing units used to defined each type of validation matrix.

- High level requirements: They represent the broadest categories of a system functions, such as business processes, including data entry and data capture, normal and exception processing, and database updates.

- Intermediate level requirements: These are components of the high-level requirements and often at the transaction or batch processing level. They may include sequences of actions to be taken, information to be displayed, error routines and associated messages.
- Detail Level Requirements: They are the specific steps in application processing, such as action steps field definitions; edit criteria, calculations, and error messages.
- Builds: A logically complete subset of an application that can be tested independently, and then integrated and tested with other builds.
- Test Runs: They consist of a set of related test cases that are used to validate the requirements at the intermediate level.
- Test Case: They represent the smallest unit of work in the testing process.
- Coverage: A measure of the effectiveness of test cases.
- Requirement Validation Matrix: A table that provides a cross-reference between a system's requirements and test case specifications.

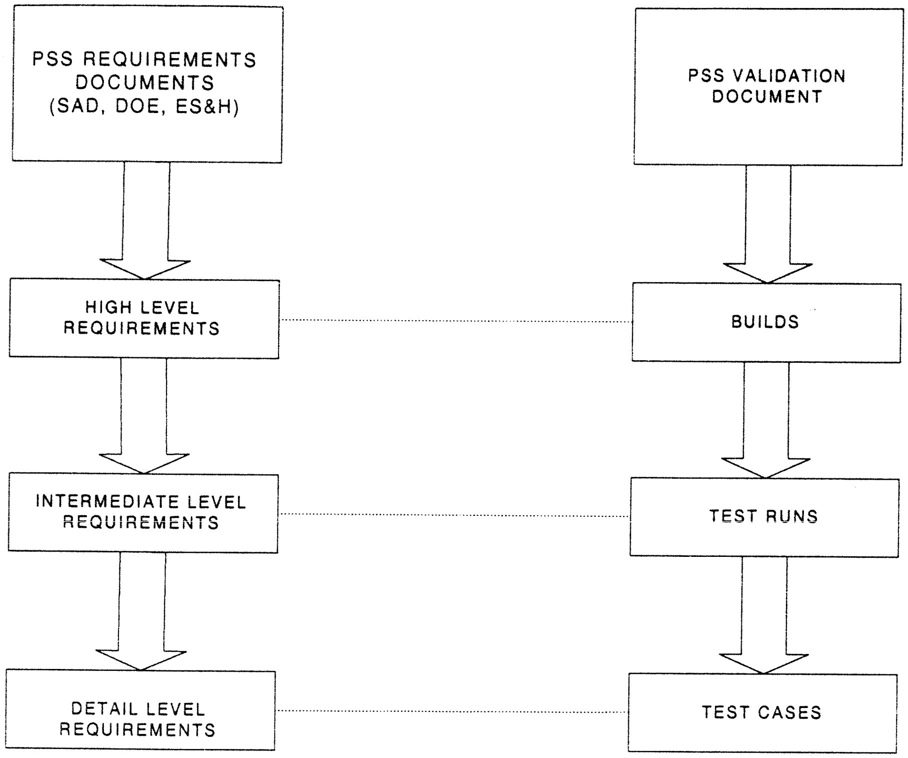

Fig. 4 Vali/Test Pro Testing Methodology

High Level Requirements	Builds													
	1	2	3	4	5	6	7	8	9	10	11	12	13	14
1. Maintain PSS System Independence		X	X	X	X	X	X	X	X	X	X	X	X	X
2. Interface PSS to ACIS			X											X
3. Provide permit for front-end shutter operation		X	X	X		X	X	X		X	X	X		
4. Separate PSS from machine components protection system		X	X	X	X	X	X	X	X	X	X	X		X
5. Provide data collection of the computer interlock and radiation meters		X	X	X		X	X	X		X	X	X	X	
6. Provide safe access to enclosure.			X	X			X	X		X	X	X		
7. Provide emergency access and shutdown			X					X	X		X			
8. Provide search and secure			X	X	X		X	X	X		X			
9. Provide beam interlocking		X	X	X			X	X	X		X			
10. Provide system integrity.		X	X	X	X	X	X	X	X	X	X	X		
11. Provide an orderly means of turning off X-ray source via a control system without operation of interlock.		X	X	X		X	X	X	X		X			X
12.0 Interface PSS to EPICS.	X													
13.0 PSS shall monitor remote I/O communications.		X												
14.0 Provide verification and validation of procedures for software.	X													

Fig. 5 High-level Validation Matrix

4.0 AUTOMATED REAL-TIME TEST SYSTEM (ARTTS) DEVELOPMENT

In this section, we described how the ARTTS system was developed as a tool used to automate the testing of the PSS system described above. The development of the ARTTS system described here consists of three major areas of system integration: 1) Test Requirements, 2) Software Tools, and 3) System Hardware. Below we explain how the use of commercially available off-the-shelf software and hardware was integrated to perform all required testing for the PSS system.

4.1 Test Requirements

The department of energy regulations requires the personnel safety system for each beamline to be validated every 6 months. In addition to the high level matrix described in section 3.0 there is an intermediate level and a detail level matrix that were required to completely define the PSS system requirements.

4.2 Reasons to Automate PSS Testing

The main reasons to automate the PSS system testing is driven by an increase in the need for additional validations over the past five years. Another reason to automate PSS testing is result of a steady increase in software change requests trend as shown in figure 7. Additional reasons to automate the PSS system are listed below

- Manual testing of the PSS takes up to 3 days to validate one beamline depending on its configuration.
- An automated testing system will catch more software errors before final testing is performed on the lab floor.
- Will reduce the overall time required to validate a beamline.
- Since 98% of all software change requests (SCR's) are related to HMI and not safety, most of the testing performed on the lab floor can be move to the ARTTS simulator.
- Will increased software reliability.

- Will reduce the cost of testing.

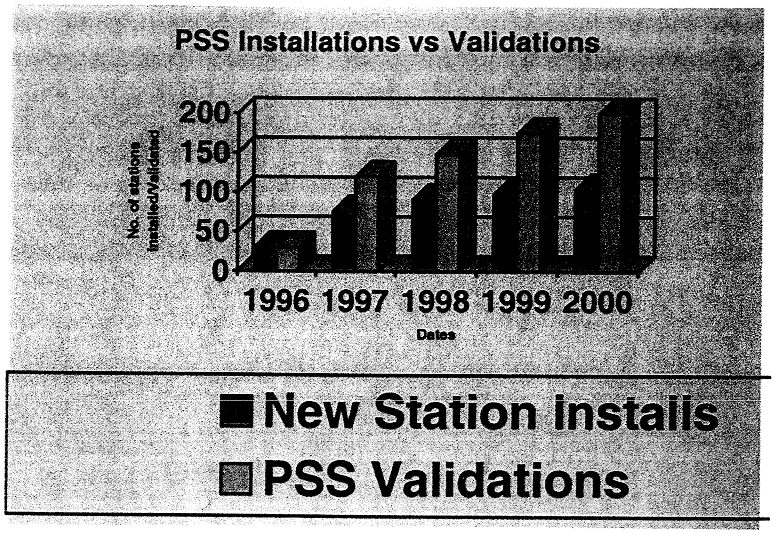

Figure 6 New stations vs PSS Validations

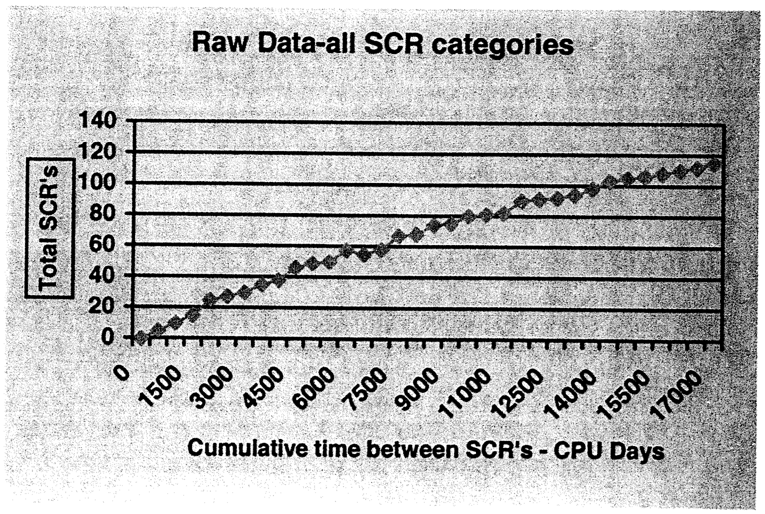

Figure 7 Software Change Request vs CPU Days

4.3 Software Tool Integration

To fully automate the testing of a PSS system required the integration of four software tools onto a single computer platform along with a fifth tool that was setup to work on a separate notebook computer (see figure 9). Together the software tools provided the following functions: 1) Real time I/O simulation for programmable logic controllers, 2) Human Machine Interface (HMI) to users control panels, 3) Graphical User Interface (GUI) testing for Windows based applications, 4) Graphical test planning, batch testing and defect tracking, 5) PLC fault code verification for Allen Bradely PLC's, and 6) PLC fault code verification for GE PLC's.

The software tools selected to performed the above system functions were as follows:

- Programmable Industrial Control Simulation (PICS) software provided a real-time HMI and I/O simulator for testing PLC-based control system.

- WinRunner software provided a functional testing tool designed primarily to test graphical user interfaces of Windows base applications. It was used here in the ARTTS system to design and verify the state of the PSS system's I/O.

- Test Director software module provided graphical test planning, batch testing, defect tracking and an interface to the WinRunner module.

- RSLogic is an A/B tool designed primarily to down load software to A/B's PLC's and to verify faults and task states in Chain B of the PSS system.

- State Logic is a GE software tool that provided the means to verify faults and task states in Chain A of the PSS system.

4.4 ARTTS Hardware

The ARTTS system's hardware is best described by the systems hardware layout in figure 8. The following hardware makes up the ARTTS prototype system:

- Gateway GP7-500 – 500 MHZ CPU, 1GB RAM, 20 GB HD IDE, Vision Tek video card, A/B 1784KT card, AB-5236 SD, 10/100 MB/s Network Interface Cards
- Nokia 17 inch Monitor
- Allen Bradley PLC-5/30 Rack: A/B 120 VAC power supply, DH+A, DH+B, A/B PLC-5/30 CPU
- GE Fanuc Series 90-70 Rack: series 90-70 power supply, State Logic CPU, Ethernet interface
- Unicom Dyna-net/4 Null Hub

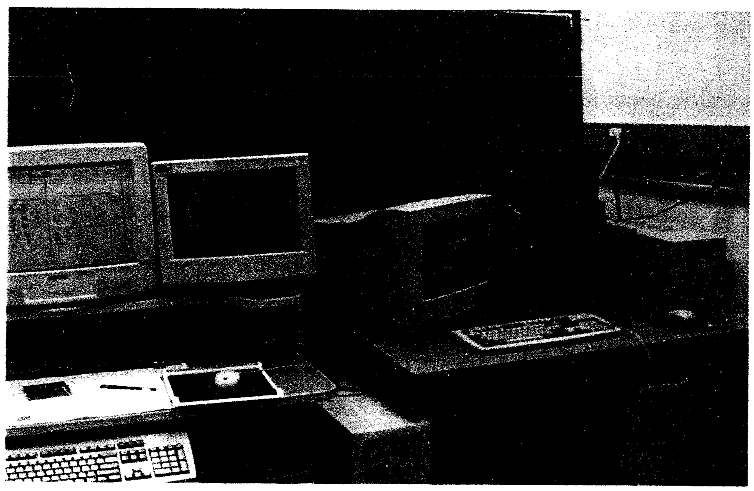

Fig. 8 ARTTS Hardware

4.5 ARTTS System Schematic

A schematic with all the ARTTS system components integrated together is shown in figure 9 below.

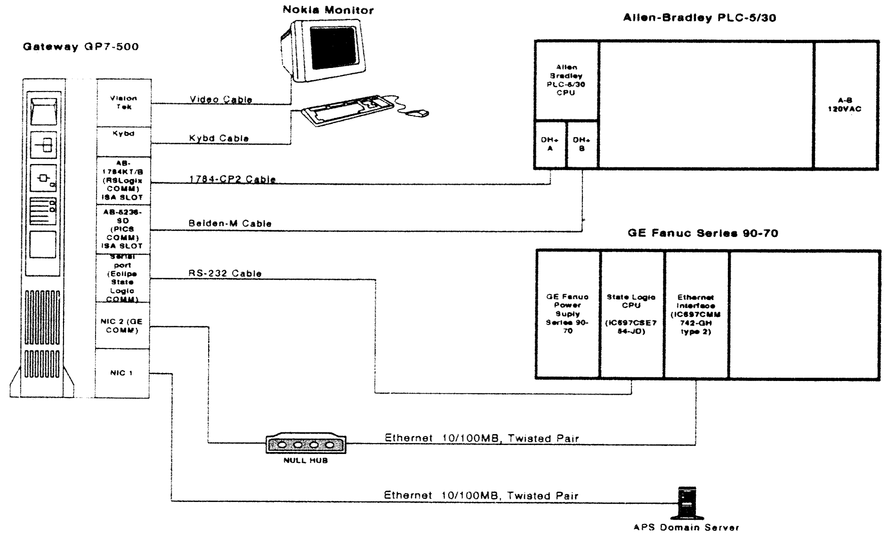

Fig. 9 ARTTS System Schematic

5.0 ARTTS OPERATION

5.1 Human Machine Interface (HMI)

A users interface to the ARTTS system is via a computer keyboard, mouse and color screen computer monitor. Using PICS HMI graphical user interface, a user can control and monitor all functions of the PSS system by activating the proper switches on the user control panel and then monitoring the system function via LED outputs (see figures 10).

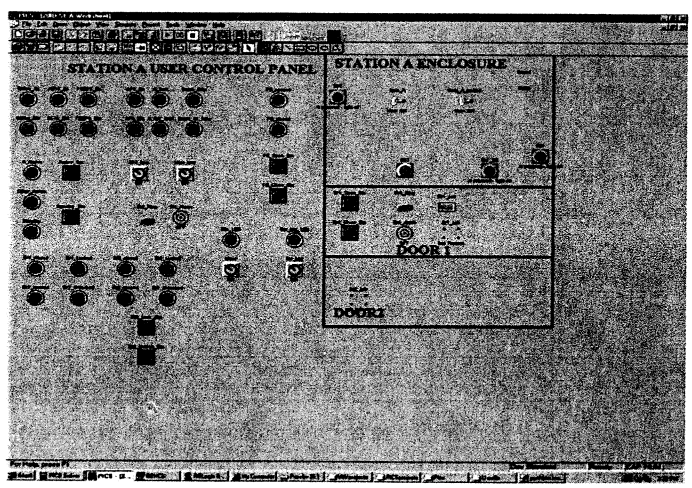

Fig. 10. ARTTS HMI Panel

5.2 Test Planning

A user can setup or activate batches of test to run by selecting test plan from TestDirector's menu. Once selected a batch of test can be ran by simply selecting Run Test from TestDirector's menu. This step automatically activates WinRunner to execute the selected test scripts (see figure 11).

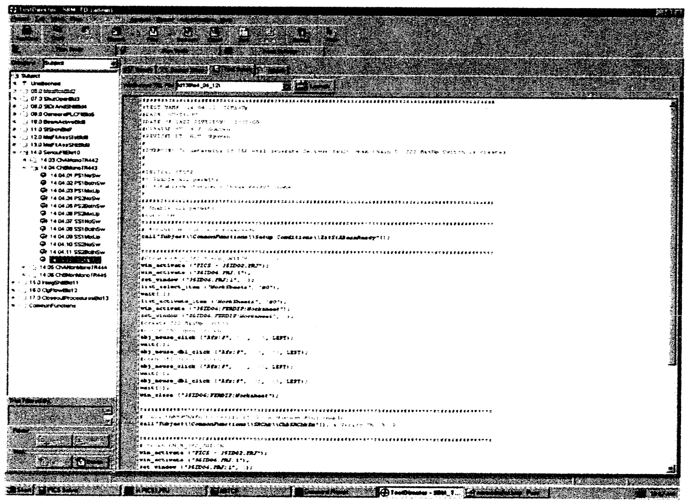

Fig. 11 ARTTS Test Planner

5.3 Test Results

Executed tests results are verified within Test Director by selecting the icon labeled "Details" from the menu. The results of each test will be listed as pass or failed along with the date and time of execution of each test (see figure 12).

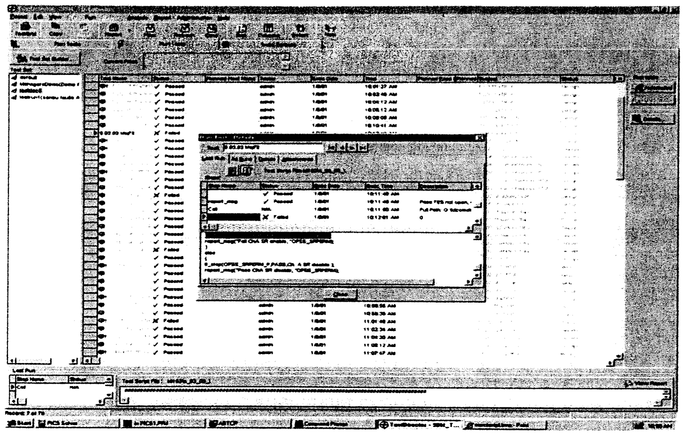

Fig. 12. ARTTS Test Results

6.0 ARTTS SYSTEM PERFORMANCE

The ARTTS system was very successful reducing the amount of time required to perform PSS testing. For example, the ARTTS system reduced the overall amount of time required to process a batch of 72 test cases from about 4 hours of manual testing to 1 hour and 36 minutes of automated testing. In general, the ARTTS system achieved an execution time of 1 minute and 20 seconds per test case. Furthermore, as a result of additional test coverage the ARTTS system greatly increased the reliability of the PSS software.

7.0 ARTTS SYSTEM ADVANTAGES VS DISADVANTAGES

Advantages	vs	Disadvantages
• Scalable.		• Uses 5 different software packages.
• Major reduction in time required to validate a batch of test cases.		• Configuration management is more a challenge with multiple software packages.
• Commercially available software.		
• Commercially available hardware.		
• Reduced development time.		
• Easy to build.		
• Better test coverage.		

Fig. 13 Advantages vs Disadvantages

BIOGRAPHIES:

Jon Hawkins, Group Leader

Argonne National Laboratory

Interlock Systems & Instrumentation Group

Bldg 401, Rm C1249

Argonne, IL 60439, USA

hawkins@aps.anl.gov

Jon K. Hawkins is a Group Leader at the Advanced Photon Source, Argonne National Laboratory. He is responsible for the design, testing, installation and validation of real-time embedded computer based mission critical systems. Mr. Hawkins is a senior engineer, who's experience ranges from high speed analog and digital system design with full custom ASIC's to computer based system requirements/risk analysis. Mr. Hawkins hold a US patent, has been presented the Argonne National Laboratory Directors Award and has published numerous articles in technical journals.

Reginald B. Howard, P.E.

Argonne National Laboratory

Interlock Systems & Instrumentation Group

Bldg 401, Rm L1111

Argonne, IL 60439 USA

rhoward@aps.anl.gov

Reginald B. Howard is a test engineer consultant at Argonne National Laboratory while working for an IT consultant firm named Web Business Controls, Inc. He received his BS degree (1987) in electrical engineering from Old Dominion University. He has been working at Argonne National Laboratory for the last 2 years assisting with developing the automated real-time testing system for embedded control systems. In addition, he is responsible for developing and implementing a web based electronic documentation system for the Laboratory. Previously he worked as an electrical engineer with the Federal Aviation Administration (FAA) where he was responsible for developing the FAA's first touch screen control system for approach light control systems at O'Hare International Airport Air Traffic Control Tower (ATCT). In addition, he has designed electrical distribution systems and lightning protection systems for ATCT facilities. Furthermore, he has worked in the manufacturing industry where he designed and tested electrical power conditioning products. Mr. Howard is a licensed professional engineer in the state of Wisconsin and is a member of the IEEE Computer Society and "The Instrumentation, Systems and Automation Society" (ISA).

Haung V. Nguyen
Argonne National Laboratory
Interlock Systems & Instrumentation Group
Bldg 401, Rm B1196
Argonne, IL 60439 USA
nguyen@aps.anl.gov

Van Nguyen is an Engineering Assistant at Argonne National Laboratory (ANL). He is also an undergrad computer science student whom is planning to pursue a graduate degree in computer science at Illinois Institute of Technology (IIT). While working at ANL he has been responsible for testing Personnel Safety Systems (PSS) and responsible for maintaining Programmable Logic Controller (PLC) software for PSS systems. Furthermore he has been involved with developing and writing software tests scripts for Automated Real-Time Test System (ARTTS).

REFERENCES

1. Interim Technology, "Vali/Test Pro-A Validation and Testing Process" Ver.1.1. pp. 6.1 –7.16