

BWXT

Report No.: Y/LB-16,072 (Paper)

●Y-12

A BWXT/Bechtel Enterprise

COMPUTER GENERATED INPUTS FOR NMIS PROCESSOR VERIFICATION

J. A. Mullens
J. E. Breeding
J. A. McEvers
R. W. Wysor
L. G. Chiang
J. R. Lenarduzzi
J. T. Mihalcz
J. K. Mattingly

**Y-12
National
Security
Complex**

**Nuclear Materials Management and
Storage Program Office**

June 29, 2001

**Prepared by the
Y-12 National Security Complex
Oak Ridge, Tennessee 37831
managed by
BWXT Y-12, L.L.C.
for the
U.S. DEPARTMENT OF ENERGY
under contract DE-AC05-00OR22800**

**MANAGED BY
BWXT Y-12, L.L.C.
FOR THE UNITED STATES
DEPARTMENT OF ENERGY**

DISCLAIMER

This report was prepared as an account of work sponsored by an agency of the United States government. Neither the United States Government nor any agency thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

COMPUTER GENERATED INPUTS FOR NMIS PROCESSOR VERIFICATION

J. A. Mullens, J. E. Breeding, J. A. McEvers, R. W. Wysor,
L. G. Chiang, J. Roberto Lenarduzzi, J. T. Mihalcz, J. K. Mattingly

Abstract

Proper operation of the Nuclear Identification Materials System (NMIS) processor can be verified using computer-generated inputs [BIST (Built-In-Self-Test)] at the digital inputs. Preselected sequences of input pulses to all channels with known correlation functions are compared to the output of the processor. These types of verifications have been utilized in NMIS type correlation processors at the Oak Ridge National Laboratory since 1984. The use of this test confirmed a malfunction in a NMIS processor at the All-Russian Scientific Research Institute of Experimental Physics (VNIIEF) in 1998. The NMIS processor boards were returned to the U.S. for repair and subsequently used in NMIS passive and active measurements with Pu at VNIIEF in 1999.

Introduction

Two key concepts in international weapons inspections are the classified information barrier and inspector authentication of the measurement system (IB&A) [1]. The information barrier protects the host country's weapons information, keeping it out of the hands of the inspectors. Since the inspectors cannot see the data collected by the measurement system, their confidence in its conclusions depends entirely on their confidence that the measurement system is working correctly.

Authentication has two aspects. The most fundamental aspect is authentication of the measurement system's design and implementation, giving the inspectors confidence that the system can correctly and reliably make the measurement. This requires detailed analysis and testing of the design. Authentication of the design is certainly possible, although possibly expensive if the system is complex. The second aspect is authentication of the copy of the measurement system actually used in inspections. This requires that the inspectors be satisfied that the system has not been subverted. Authentication of the field copy is difficult given some constraints commonly assumed, primarily that the field copy will never leave the host country's control once it has been accepted for inspection use. This constraint means that the system is stored at the host site between inspections, and the inspecting agency can never privately inspect any system once it is used for an inspection.

This paper describes the use of self-test functions on the NMIS data acquisition boards in the light of these authentication needs.

Description of the Data Acquisition Boards

There have been several versions of the NMIS system from 1975 to the present. NMIS systems in use since 1984 have had built-in self-test (BIST) functions. About 1985 the NMIS system used a VAX computer and rack-mounted signal processing system, housed in a semi-truck trailer for mobility. By 1996 the system was housed in a desktop personal computer using two PCI bus cards, timing pulses from five detectors with nanosecond (ns) resolution [2].

The PCI boards read detector pulses through a NIM bin constant-fraction discriminator (CFD) and sends the pulse times to the PC in digital format. The board locates the leading edge of the analog fast NIM pulse, and places it in its stream of digital bits, 1 bit per ns, signifying whether or not a pulse edge was detected during that ns. The board uses field programmable gate arrays (FPGAs) to compress this series of bits by removing the 0 bits, then merge the five detectors' data into a single stream formatted for the PC's use. The formatted data is transferred over the PCI interface via the board's PCI interface chip. FIFOs buffer bursts of data at each detector channel's output section and at the PCI interface.

In the board version currently used in the field, an ORNL-designed application specific integrated chip (ASIC) collects the counts from five detectors at 1 GHz sampling rates, and FPGAs perform compression and formatting. The latest ORNL board uses only commercial off the shelf (COTS) chips. The custom ASIC has been replaced by Synchronous Optical NETwork (SONET) deserializer chips to transform the detector pulse serial pulse stream input into 16 bit (16 ns) parallel input, and FPGAs to perform its other functions. An external power supply required for the ASIC chip was also eliminated. The efficiencies of the new board have raised throughput to the point that the system can now do the signal processing for most five-detector measurements in real-time.

An important aspect of authentication is the trust the inspectors have in the system's design and implementation. The new NMIS board, based on FPGAs, has several characteristics that help reliability:

- The SONET chips are designed to serialize at 2.5 Gigabit per second (Gbps), but run at only 1 Gbps on this board.
- The FPGA code can run at 89 MHz according to the FPGA compiler, but is running at only 66 MHz on this board.
- The hottest component on the board has a temperature margin of 24°C.

There are also several characteristics that help information barrier goals:

- The FPGA code could be loaded into one-time programmable read-only memory chip (OTPROM) instead of the standard erasable programmable read-only memory chip (EPROM) FPGA chips. (The current version does not have socketed OTPROMs, but the FPGAs are compatible with the chips required.)
- LEDs which indicate the board's processing states are surface-mount parts that can be removed, should that be necessary to protect classified information.

- All writeable board memory is volatile.

The board has an industry-standard JTAG interface. This allows field verification of the FPGA contents and other trouble shooting. The JTAG port can be disabled if that is required to protect classified information or make authentication easier.

BIST

Both the ASIC and FPGA boards have similar built-in self-test (BIST) features. The purpose of these BIST functions is to test the board's data processing circuitry and FPGA code. These functions place known signals (pulses) at the board's digital inputs, process the data, and output the results to the PC for comparison with the expected result.

The FPGA board processes input data in 16 ns (16-bit) chunks. It can generate two types of BIST input signals:

- a repeating, fixed 16 ns pulse pattern (10 patterns are available), and
- a "walking bit" pattern in which a single pulse is shifted by one nanosecond for each insertion of the 16 ns pattern into the input stream.

The fixed pattern can be inserted at every 16 ns interval, or every 2 to 2^{16} intervals (selectable in powers of 2). The board has the capacity to expand this set of inputs and we are discussing what additions would be useful.

The built-in tests exercise the board hardware and the FPGA logic. Specific uses of BIST have been to test:

- detection of the leading edge of the pulse on any channel,
- determination of the time of the leading edge,
- registering close pulses (as little as 8 ns apart),
- counting of number of pulses within a block of time,
- time synchronization of the five detector channels,
- counting of sequential empty blocks of time (for data compression), and
- handling of acquisition pauses when board FIFOs are nearly full.

The board checks the time synchronization of the five SONET chips on the five data acquisition channels, and will recheck upon command. This test injects signals at the analog front end of the board so it is also a test of the those components.

A test program, DaBrdTst, provides testing through register-level control of the board, control of the PCI bus DMA options, and automation of long tests. DaBrdTst is also designed for tests using analog pulses. DaBrdTst employs both periodic pulses and random pulses to perform specialized tests to determine the rate of missed pulses, uneven time sampling windows, PCI bus transfer problems, and other indicators of errors.

Authentication of the Field Copy

The authentication problem actually originates with the weapon component in the closed container. The task is to *authenticate the weapon component*. To do this, a measurement system is introduced which must in turn be authenticated. Similarly the information barrier concerns also propagate to the measurement system. The weapon component has classified information, and the measurement system, once used, cannot be released for inspection because the host country must assume that it contains classified information in some form. Thus the fundamental problem has not been changed by the introduction of the measurement system. If this problem simply moves to each new level of authentication introduced, there is no closure, only a “heightened barrier” to tampering and incremental improvements in inspector confidence.

Through the JTAG interface, using standard commercial software and hardware, the FPGA code can be verified bit-by-bit against a copy in a PC file. Alternatively, FPGA code on a socketed one-time programmable read only memory (OTPROM) could be verified through a PROM reader. Since the board’s operation is controlled by the FPGA code, the FIFOs, and the PCI interface chip, verifying the FPGA code eliminates many tampering opportunities. However, a difficulty arises over who controls the COTS system used to verify the FPGA code. If the host controls it, the inspectors might have doubts about its veracity. If the inspectors control it, the host might have concerns that it subverts the information barrier. Introducing yet another system *to verify the system that verifies the FPGA code* moves the same problem to the new system. To achieve some closure to this problem, a COTS FPGA comparison system that neither party ever had private access to might be introduced. Or a PROM used during an inspection could be verified afterwards on the inspector’s equipment then replaced with a new PROM out of secure storage. This problem is very similar to the problem of verifying the computer’s BIOS and other code in PROM, and we expect a similar solution could be adopted for both problems.

If the FPGA controls the behavior of the board, and the FPGA code can be verified, then all that remains is to verify the hardware that the FPGA code runs on. A circuit-level examination of the chips at the inspection site would not be practical. Instead, the operation of the verified FPGA code on the board circuitry would be checked by observing the board’s processing of known inputs. This is most conveniently done by BIST, but could also be done using detector pulses.

A problem with this approach is that the number of possible input and FPGA part states is almost certainly very large. For example, suppose the FPGA assigns a detector pulse to one of 256 channels. Each count found therefore creates 256 possible “states” of the board. If the board’s operation might be subverted to depend on how closely in time two counts arrive, the number of board states after only two counts could be as large as $256 * 256 = 65,536$ states. If the part uses accumulators, odd behavior may not show up until the FPGA has accumulated millions of counts. Our experience has been that a large amount of randomly-generated data should be presented to a newly-designed part to find

those rare states that trigger subtle problems (such as FPGA timing errors). Since a subverted board is a “new design”, this is probably true for authentication tests also.

However, it may be that all that is absolutely necessary is that the inspector’s tests of the board have enough chance of finding tampering that the threat of discovery is credible. This might be achieved by giving the inspectors a very flexible, well thought out, automated board test routine embedded in the computer’s BIOS, and a way to verify the computer’s BIOS code along the lines of the method used to verify the board’s FPGA code.

Experiences

BIST has been our board designers’ primary means of testing during development, and is always used to check newly-built boards.

Failure of the board’s data processing circuitry in the field has been rare and there has been no strong motivation to use BIST routinely in the field. However, in 1998, an ASIC board at the All-Russian Scientific Research Institute of Experimental Physics (VNIIEF) developed a problem. The visible symptom was that very high count rates were being recorded for a detector channel. Investigation by the VNIIEF staff narrowed the problem down to the ASIC board and suggested that thermal stress was causing an open circuit. At that point VNIIEF supplied us with the output of a BIST run. From this data we deduced which circuit was open, and the board was repaired.

In addition to BIST, the PC examines the stream of data received from the board for any evidence of errors. In the case of the ASIC board, the data stream format was inefficient but its redundancy made many checks possible. The FPGA board uses an efficient format, but there are still 8 checks the PC can perform to detect a corrupted data stream when it violates the format rules. Recently, we began buying 1 GHz Pentium III computers for field use. The NMIS software detected problems immediately on the two computers purchased. The PCI interface chip, when operated in the newer PCs, was occasionally causing a 32-bit word of the data stream to be misplaced. New boards now use a newer but compatible generation of that PCI chip which has worked flawlessly.

Summary

The goal of a weapons component inspection is to establish the *authenticity* of the weapons component without revealing classified information. This is accomplished through a measurement system which analyzes the classified data then issues its judgment without giving the inspecting party any supporting information. The inspecting party must rely on the measurement system, which requires that the inspecting party be satisfied with the *authenticity* of the measurement system. The inspecting party must first be satisfied that the system is a robust design and implementation, and must then be satisfied that an exact copy is used for inspections. The latter is difficult to achieve given the

host's overriding need to protect its classified information. Practical measures increase the inspecting party's confidence that an exact copy is being used, but do not give absolute certainty.

Boards based on FPGAs have both a software and hardware component. Verification of the software is possible, while field verification of the hardware component might be impractical. However, any tampering that leaves the FPGA code intact must involve the board hardware, so tests of the board's operation with verified FPGA code can focus on detecting hardware anomalies. Such tests might rely on built-in test data patterns, generated by the FPGA code, or use detector data.

References

1. Joint DoD/DOE Information Barrier Working Group, *Functional Requirements for Information Barriers*, PNNL-13285, Pacific Northwest National Laboratory, Richland, WA, May 1999.
2. J. T. Mihalczo, J. A. Mullens, J. K. Mattingly, and T. E. Valentine, "Physical Description of Nuclear Materials Identification System (NMIS) Signatures," *Nuclear Instruments & Methods in Physics Research A*, 450 (2000) 531-555.