HIGH-INTENSITY RADIATED FIELD FAULT-INJECTION EXPERIMENT FOR A FAULT-TOLERANT DISTRIBUTED COMMUNICATION SYSTEM

Amy M. Yates, Wilfredo Torres-Pomales, and Mahyar R. Malekpour,
National Aeronautics and Space Administration, Hampton, VA
Oscar R. González and W. Steven Gray, Old Dominion University, Norfolk, VA

Abstract

Safety-critical distributed flight control systems require robustness in the presence of faults. In general, these systems consist of a number of input/output (I/O) and computation nodes interacting through a fault-tolerant data communication system. The communication system transfers sensor data and control commands and can handle most faults under typical operating conditions. However, the performance of the closed-loop system can be adversely affected as a result of operating in harsh environments. In particular, High-Intensity Radiated Field (HIRF) environments have the potential to cause random fault manifestations in individual avionic components and to generate simultaneous system-wide communication faults that overwhelm existing fault management mechanisms. This paper presents the design of an experiment conducted at the NASA Langley Research Center's HIRF Laboratory to statistically characterize the faults that a HIRF environment can trigger on a single node of a distributed flight control system.

Introduction

Safety-critical distributed closed-loop flight control systems require a fault-tolerant communication system to reliably transfer sensor data and control commands [1]. The performance of the control system may be affected if these transmissions are altered in the presence of harsh environments, such as high energy atmospheric neutrons [2],[3] and high intensity radiated fields (HIRF) [4],[5]. Aircraft operating in HIRF require special certification (see, e.g., [6],[7],[8]). A fault-tolerant communication system is used by an aircraft, the *plant* in a flight control system, to reliably send the sensor measurements to the controller that calculates the commands to be sent back to the plant's actuators as shown in Figure 1.

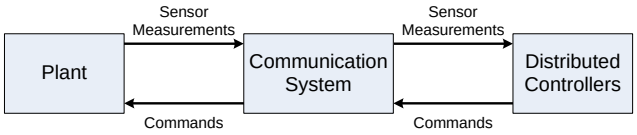

Figure 1. Simplified Schematic of the Plant – Distributed Controllers Communication

The Scalable Processor-Independent Design for Enhanced Reliability (SPIDER) architecture is the platform used to implement a distributed flight control system for this experiment [9]. This architecture has been specifically designed to recover from faults and is based on formally proven protocols. SPIDER's current fault-tolerant communication system is ROBUS-2, which consists of bus interface units (BIUs) that connect the processing elements (PEs) to the bus, redundancy management units (RMUs) to manage the communication traffic and provide robust bus-level fault-tolerance, and fiber optic data links [10],[11]. In this study two types of PEs are considered: Control Law PEs, which produce control commands by performing control law calculations based on sensor data, and I/O PEs, which pass sensor data from the plant to the communication system and relay control commands to the plant's actuators. Each PE is implemented in a separate physical device, or node, with a BIU that provides the PE access to the communication bus, creating PE-BIU nodes, whereas each RMU is implemented on its own node. The SPIDER distributed flight control system used for this study will be referred to as the SPIDER system. The SPIDER system configuration is denoted by the number of Control Law PE-BIUs × the number of RMUs. For example, the representation for a SPIDER system with one I/O PE-BIU, N Control PE-BIUs, and M RMUs is $N \times M$. For simplicity, only one I/O PE-BIU is considered in this study.

The basic operation of a SPIDER based flight control system is as follows. The control system operation begins at the beginning of a control cycle

with the I/O PE-BIU acquiring the plant's sensor measurements. The data is then passed from the BIU to the RMUs. From the RMUs, the data is transferred to all the BIUs, including the BIUs associated with the Control Law PEs. At these PEs, the control commands are computed based on the sensor data. The commands are then sent from the BIUs connected to the Control Law PEs back to the RMUs and on to the BIU connected to the I/O PE. This PE sends the commands to the plant. This sequence of closed-loop events shown in Figure 1 forms a *control cycle* that is periodically repeated. The inverse of the period is the sampling frequency, $f_{s,}$ of the control system. Each control cycle is implemented over a number of *ROBUS cycles*, which consist of a sequence of events needed to manage the fault-tolerant communications [10],[11].

The following terminology is typical in the fault-injection field. A *fault* is a defect or flaw that occurs in a hardware or software component. An *error* is a manifestation of a fault when it causes a component to deviate from correctness [12],[13]. In a fault-tolerant system, many faults will not become errors. When the errors cause a system to not function properly and not satisfactorily complete the service required, this is called a failure [13].

The goal of this paper is to present the design of an experiment that exposes one Control Law PE to HIRF while the designed SPIDER system is simulating the operation of a closed-loop flight control system. The purpose of the experiment is to gather data on the occurrence of errors on the exposed node, to develop statistical models for them, and to use these models to predict the closed-loop system performance degradation. The experiment was recently conducted at the NASA Langley Research Center's HIRF Laboratory. For this experiment, an error monitoring system had to be designed, implemented, and validated. A comprehensive treatment is presented in [14],[15]. The experimental HIRF-induced error data will be used to estimate their effect on the tracking error of a Boeing 747 closed-loop digital flight control system. This tracking error will then be compared with estimates that use the statistical models characterizing the occurrence of these errors together with a performance model used for tracking error prediction [16]. This paper reports on the experiment design and presents some initial data analysis. The physical platform of the SPIDER nodes used in the experiment is not flight certified. Each physical node contains a CPU module for the software, a Field Programmable Gate Array (FPGA) module, where Very-High-Speed Integrated Circuit (VHSIC) Hardware Description Language (VHDL) is synthesized on the FPGA, a power supply, a fan module, and two boards for the eight optical I/O communication ports [14]. The CPU module runs White Dwarf LINUX and has a keyboard input and a monitor display. All of the components of a physical node are encapsulated within an enclosure. However, the enclosure was removed to increase susceptibility of the node to HIRF-induced errors.

The experiment was conducted in the reverberation Chamber A [17]. The dimensions of this chamber are $2.90m \times 7.01m \times 14.33m$ (height $\times$ width $\times$ depth). Inside the chamber are placed a transmit antenna, receive antenna, two stirrers near opposite corners, and two cameras. In addition, the node under test is placed on a non-conductive foam block. Following RTCA/DO-160F Section 20 standards [8], the objects inside the chamber are appropriately separated from each other and from the walls. For this experiment, the transmit antenna radiates continuous waves of a given frequency and field strength. The receive antenna is connected to a measurement system that includes a spectrum analyzer. The transmit antenna and continuously rotating stirrers create a time-varying electromagnetic field environment that can induce electrical current, and thereby inject faults into the node under test.

The rest of the paper is organized as follows. The next section presents the HIRF experiment design and implementation. This section has six subsections. The first one gives the design specifications and assumptions, including a high-level overview of the design and a description of the management systems that coordinate the execution of the experiment. Next is a subsection introducing the Function Monitors (FMons), which are used to detect the occurrence of errors in the node under test. The next two subsections present the selection of the parameters for the electromagnetic environment and the experimental configuration. The techniques implemented to prevent permanent damage to the power supply are presented in the following one. The final subsection gives a summary of the procedure to select the parameters of an experimental

round. The next section gives the experimental results. It is divided into three subsections: summary of rounds completed, Function Monitor observations, and data analysis. The paper's conclusions are given in the final section.

Design of the HIRF Experiment

Design Specifications and Assumptions

The ultimate goal of this work is to predict the performance degradation of an aircraft that uses a flight control system based on the SPIDER system currently available. HIRF exposure can cause both transient and permanent faults in electronic systems [18]. However, to prevent damage to the node subjected to HIRF, the first experiment specification is to focus on transient faults. This specification will be met by limiting the maximum amplitudes of the electrical field strengths that are selected for the experiment. The second specification is for the interface between the SPIDER system and the experiment management system. The interface should not affect the ROBUS-2 fault tolerant communication system. The third specification is for the experiment to monitor the occurrence of errors in real-time. It is assumed that detected errors in the node under test during the experiment translate to fail-silent type error behavior in the post-test analysis. The fail-silent assumption means that the node under test either works properly or does not produce any output when an error is manifested [12],[19]. Fail-silent behavior can be achieved at a reasonable cost [12], but in the experiment it will not be enforced. The assumption, nevertheless, will simplify the analysis of the effect of the errors. A design for these specifications is presented next.

At a high level, the HIRF experiment consists of an $N \times M$ SPIDER system, a SPIDER management system, and a HIRF management system. The SPIDER management system is partitioned into an experimental execution subsystem and a data management subsystem. The experimental execution subsystem is implemented in a test controller, a node that is not part of the SPIDER system but physically equivalent. The test controller is interfaced to each PE-BIU and RMU node using links that are not part of ROBUS. Physically, each node has eight independent transmit and receive channels. For the purposes of this experiment, some of these links were used for ROBUS communication and some to communicate with the test controller. Since each node has only eight communication ports, the maximum possible connections from a single test controller to the PE-BIUs and RMUs is eight. To prepare for future tests containing more than eight SPIDER system nodes in the configuration, it was necessary to split the test controller into two nodes interfaced to the PE-BIU and RMU nodes named the Primary Test Controller (PTC) and Secondary Test Controller (STC), respectively.

To monitor the occurrence of errors in the Control Law PE exposed to HIRF, an $N \times M$ SPIDER system was configured to operate in a simulated closed-loop system as depicted in Figure 2. The STC is not depicted here because it does not provide observations about the errors monitored. To close the loop, simulated sensor signals are generated in the PTC and transmitted to the I/O PE, which broadcasts the data via its BIU to all the PE's in the SPIDER system. To verify that each PE correctly received the sensor data, the PEs send it back to the PTC. To simplify the implementation, the Control Law PEs do not perform any calculations, but receive simulated computed control commands from the PTC. The Control Law PEs then broadcast the commands via their BIUs to all the PEs, and the PEs complete the loop by sending the commands back to the PTC. At the PTC, comparisons are made to diagnose if there was an error in the transmission of sensor and/or command data during each control cycle.

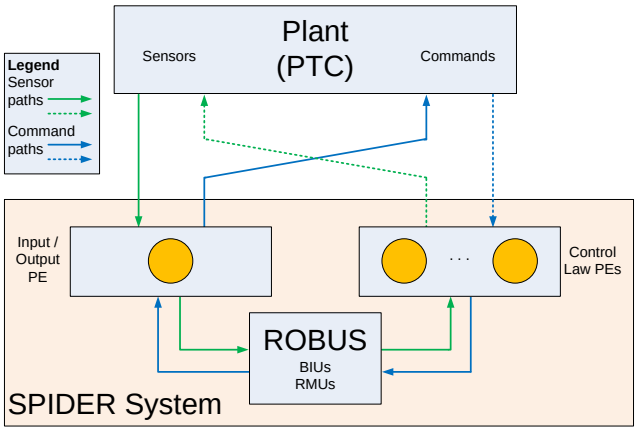

Figure 2. Schematic of the Data Paths for Error Monitoring

An experimental *round* is the sequence of actions performed by the SPIDER management

system shown in Figure 4. The events of a round include the operator enable at each test controller, the setting of the radiation parameters, the execution of a selected number of control cycles, and the transfer of the observation data and test logs to the repository [14].

The ROBUS cycle events shown in Figure E are: clock synchronization, diagnosis of SPIDER nodes, schedule update, and PE Broadcast [14]. During the PE broadcast block of a ROBUS cycle, each PE has time allocated to broadcast its messages. The ROBUS communication schedule is such that the PEs only broadcast once per control cycle. When the node under test contains a transient fault, the node may be isolated from the rest of the SPIDER system during the diagnosis phase of a ROBUS cycle. Once the fault has lifted, the reintegration of the node under test into the SPIDER system can take up to about 7.37 ROBUS cycles. Assuming the fault does not last more than one ROBUS cycle, at least 9 ROBUS cycles are required per control cycle to guarantee that the fault only affects one control cycle. For this experiment, the control cycle duration was 10 ROBUS cycles. The first ROBUS cycle was chosen to have all PEs broadcast their data. During the remaining nine cycles, no PEs are scheduled to transmit. To determine the duration of a control cycle, its inverse, i.e., the sampling period f_s is calculated as follows. For this experiment, the ROBUS clock runs at 3 MHz and each ROBUS cycle is set to last about 1715 clock cycles. Thus,

$$f_s = \frac{3\times10^6 \text{ clock cycles}}{\text{second}} \times \frac{1 \text{ ROBUS cycle}}{1715 \text{ clock cycles}} \times \frac{1 \text{ control cycle}}{10 \text{ ROBUS cycles}} \approx 175 \text{ Hz.}$$

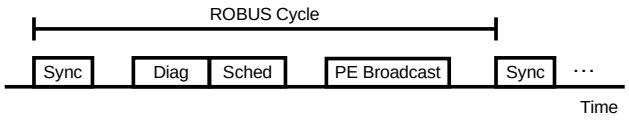

Figure 3. ROBUS Cycle (1/(10f_s) sec)

To validate the operation of the SPIDER management system, the capability to inject software-induced faults was also designed and implemented. Figure 4 depicts a top-level view of the SPIDER management system interfaced with a SPIDER system, including the modules needed for software-induced fault-injection. The Test Execution Software and the Data Management Software have a user interface including a display monitor and a keyboard input. The execution of each set of rounds starts by reading a Test Specification File in the Data Management Personal Computer (PC) which provides the test controllers with runtime parameters specific to that test. The Test Specification File is given only to the PTC, which then transfers to the STC the necessary information. The Controller Coordination Links (CCLs) are fiber optic data links between the PTC and STC. The Test Execution Software of both test controllers handles the specifications from the file and communicates with the Data Management PC to store the observations and test logs at the end of a round. The PTC and STC use Primary Test Links (PTLs) and Secondary Test Links (STLs), respectively, to communicate with the corresponding nodes. Each controller has a software interface unit to manage the communication between the software and hardware.

A brief description of a test controllers' protocol follows. At the beginning of the round, the operator enables the round at each test controller specifying the radiation settings for the round. The CPUs in the PTC and STC specify the beginning of the round by sending a signal to the software interface in the FPGA during the System Enable mode. The Round Controller starts the Round Timer (RTmr) and enables the round. Once the System Health Monitors determine that SPIDER is ready, the Round Controllers exchange Ready messages across the CCL and signal to the software that SPIDER is ready. The SPIDER management system then enters the Fault Injection Setup mode, where the PTC software sends the fault-injection specifications, if any, to the PTC Fault Injection Cotnroller (FIC). The PTC FIC forwards the received data across the CCL to the STC FIC. The fault-injection specifications are sent to the fault injectors within the PE-BIUs and RMUs via the PTLs and STLs, respectively. Once the FICs have all the fault injection data, the STC is ready to start executing the round and sends a Start message across the CCL. Then the SPIDER management system enters the Function Setup mode, where the PTC software sends application assignments (I/O or Control Law) for the PEs and the duration of a control cycle and round to the Function Monitors at the PTC.

After this setup is complete, the Round Controller at the PTC then sends a Start message across the CCL to the STC, thus beginning the Function Execution mode. The Round Controller enables the PTC FMons and Function Timer (FTmr) as well as the PTC and STC FICs and System Health Monitors. The control cycles begin executing, and the State Monitors (SMons), FMons, FTmr, and RTmr generate records to send back to the software. In this experiment, only the FMon records were read by the PTC's software. FMon results are used to diagnose if a SPIDER error occurred and provide some insight into the possible causes. The FMons receive data about where errors occurred in the system and provide additional checks to check if the correct data was transmitted through the entire communication system successfully.

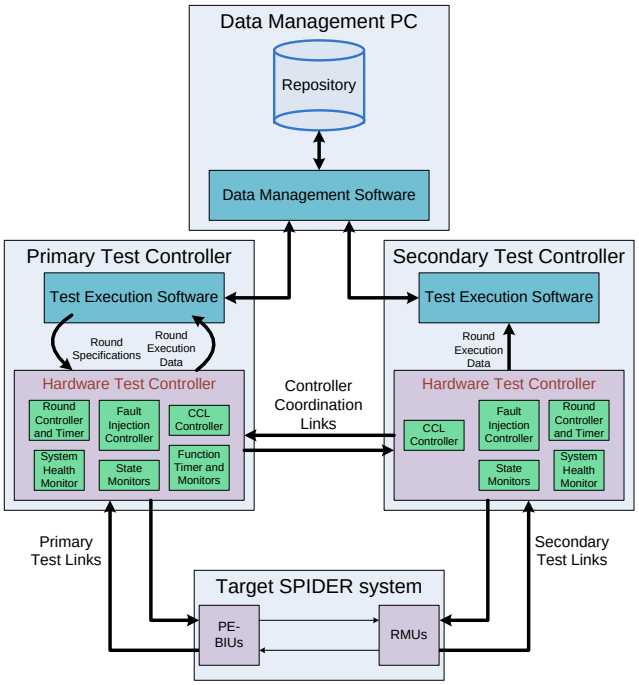

Figure 4. SPIDER Management System

The PTC software informs the Round Controller that the round is complete once FMon observations have been gathered for the preset number of control cycles. Then the Round Controller stops execution at the PTC and sends a Stop message across the CCL to the STC Round Controller that then stops the execution at the STC. Both Round Controllers report the stop trigger condition to the software. The PTC and STC Test Execution Software send this stop condition to the Data Management Software to be stored in the repository. The FMon observation records stored in the PTC Test Execution Software are then transferred to the Data Management Software, which stores the data in the repository, and a round of execution is complete. The next subsection describes how these FMon observations are formed.

Function Monitors

The main purpose of the Function Monitors in the PTC is to determine the occurrence of errors in the SPIDER system. Associated with each PE, there is a function monitor transmitter and a function monitor receiver. During the PEs' scheduled time period, the function monitor transmitters send sensor and command data to the I/O PE and the Control Law PEs, respectively. Each PE's BIU then transfers the data through the RMUs to all of the BIUs, including the transmitting BIU shown in Figure 5. From the BIUs, the data is transferred to their associated PEs and FMon receivers. Because of the broadcast nature of ROBUS-2, all of the FMon receivers make observations on each sensor or command data transmission. For example, in a 2×1 system, when the FMon associated with PE-BIU 1 transmits, the FMons associated with all the PEs make an observation about the transmission through ROBUS-2. To simplify the notation, the FMon associated with PE-BIU i is denoted FMon i. Each observation made at FMon i about the transmission originating from FMon j is a value from 0 to 7. The definitions of these observations are given in Table 1. For further details on the operation of the FMons see [14].

The observations for each control cycle can be represented with a square matrix of FMon observations with row and column dimensions equal to the number of PEs as shown in Table 2. PEs are considered transmitting (Tx) when they broadcast their sensor or command data to ROBUS-2 and receiving (Rx) when they obtain data from ROBUS-2. Each row gives the observation codes determined by a function monitor receiver with respect to a function monitor transmitter on each column. The observation code 7 in the i, j th entry of the matrix means that the data that was sent by the j th PE was received without errors by the i th PE. If there are any observations other than 7 in the matrix of FMon observations for a given control cycle, then the node

in the HIRF Chamber is said to be operating in the error mode for that control cycle.

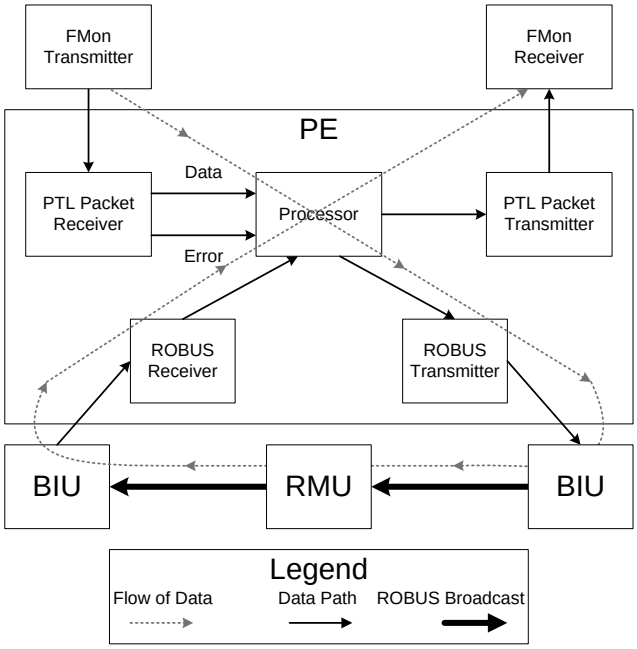

Figure 5. FMon Control Data Transmission Path

Table 1. Definitions of FMon i Observation Codes

Code	Observation	Definition
0	Omitted Sender Id	FMon i received no message from FMon j
1	Invalid Sender Id	FMon i received a message from FMon j when PE j is not active
2	Repeated Sender Id	FMon i received multiple messages from FMon j
3	Bad Payload Length	FMon i received a message from FMon j with an incorrect message length
4	Detected Reception Error at Receiver PE	FMon i received a message from FMon j where PE i detected a reception error on ROBUS
5	Detected Reception Error at Sender PE	FMon i received a message from FMon j where PE j detected a reception error on PTL
6	Bad Message Content	FMon i received a message from FMon j with incorrect content (undetected error)
7	Good Message	FMon i received a message from FMon j with the correct content

Table 2. Possible FMon Observations

	FMon Tx 0	FMon Tx 1	FMon Tx 2
FMon Rx 0	0-7	0-7	0-7
FMon Rx 1	0-7	0-7	0-7
FMon Rx 2	0-7	0-7	0-7

HIRF Specifications

The experiment exposed a single SPIDER node to HIRF inside a reverberation chamber. It has been shown that the electromagnetic environment in the aircraft cavity is similar to that of a mode-stirred chamber [20],[21]. To select the electromagnetic environment parameters, standards for radiated susceptibility tests were followed [8],[7]. For airworthiness certification of electronic equipment, the FAA requires immunity to RF environments at various field strengths between 10 KHz and 40 GHz [7]. RTCA/DO-160 procedures for HIRF testing of airborne equipment cover a frequency range of 100 MHz to 18 GHz [8]. To narrow down the choice of frequencies and to follow the experiment specification of avoiding permanent errors, only frequencies between 100 MHz to 200 MHz were considered. Supporting this choice are the fly-by test results described in [22]. They report electromagnetic coupling effects inside the cabin of NASA's Boeing 757 when flying near "a fixed transmitter driving a log periodic array (172 MHz)." Thus, the HIRF environment generated was a continuous wave at one of the chosen frequencies between 100 MHz and 200MHz with the stirrers continuously rotating (for mode-stirring). To select the frequencies a procedure similar to that in [8] was followed. Twenty-five logarithmically spaced frequencies were selected in the decade starting at 100 MHz. The first eight frequencies are under the chosen limit of 200 MHz. The selected frequencies are: 100, 110.07, 121.15, 133.35, 146.78, 161.56, 177.83, and 195.73 MHz. This frequency selection was also used in a previous HIRF experiment with a SPIDER system [17]. For each frequency, the range of field strengths selected for this experiment was up to 300 V/m.

There were two main sources of randomness in the experiment. First, the electromagnetic environment is random: the mathematical models for the average and peak electrical fields in the chamber are given in terms of probability distributions [20].

Second, at the beginning of each round the phase difference between the stirrers is random. Thus, the data for this experiment will be a random sequence of observations for each trial, i.e., each round. Invoking the fail-silent assumption makes it possible to reduce the possible values of this sequence at each control cycle to only two values: 0 and 1, denoting no error and error detected, respectively. These sequences are samples of a random process that is denoted by $z_1(k)$, where the integer k denotes the control cycle number. The round duration was selected so that it would be possible to repeat the trial a sufficient number of times for statistical analysis. The choice was 20,800 control cycles per round or about 2 minutes. A few rounds with 939,600 control cycles or about 90 minutes were also executed.

The final HIRF Chamber specification that needs to be determined is the rotation speed of the two stirrers. For mode stirring chamber operation, the stirrers are continuously rotating. Each stirrer is independently positioned by a stepper motor with 507,904 steps. By rotating them at different speeds, the effective period when both stirrers return to the same position can be selected. For the 2-minute rounds, the stirrers are configured to rotate at 7 and 18 seconds per revolution, providing an effective period of 2.1 minutes. For the 90-minute rounds, the stirrers rotate at 8 and 8.01 seconds per revolution, providing an effective period of about 106.8 minutes. NIST Technical Note 1508 [20] describes how the electromagnetic environment at any particular location in a reverberation chamber is defined by the electromagnetic boundary conditions set by all paddle positions. Thus, for mode stirring, the chamber boundary conditions repeat in a periodic fashion as the paddle rotations repeat. NIST TN 1508 also provides data for NASA's Chamber A configuration using two independent paddles, and describes how the paddles may be rotated at different rates to increase the effective period of repeating electromagnetic environment, i.e., total number of effective paddle positions. This is the reason that the effective period of the pair of stirrers was selected to be slightly more than the round duration. This difference adds another source of randomness: each round is exposed to a different electromagnetic environment for a few seconds. The statistical effect of this difference is expected to be reduced when a round is repeated at the same frequency and field strength.

Experiment Configuration

A 2×1 SPIDER system configuration was chosen for the HIRF experiment with an I/O PE, a control law PE, and an RMU outside of the HIRF chamber, and only one control law PE subject to HIRF induced errors inside the chamber as shown in Figure 6. The smallest SPIDER system configuration is 1×1 with an I/O PE-BIU, a Control Law PE-BIU, and an RMU. The additional Control Law PE located outside of the chamber was included in the system to neutralize the effects of possible fail-silent assumption violations. The node in the chamber is connected to the RMU and PTC with fiber-optic cables which are known not to be affected by HIRF [12]. Also, the two test controllers are connected to a Data Management PC via Ethernet. For this physical fault-injection experiment, a HIRF Test Controller (HTC) is used to control the HIRF chamber instrumentation and maintain coordination with the PTC and STC nodes. The HTC runs in another PC. It was implemented with Agilent Visual Engineering Environment (VEE) software by the HIRF personnel [23]. The HTC communicates with the PTC and STC using two independent RS-232 serial lines to coordinate the events of a round of execution. It is necessary for the HTC to control the radiation and activity within the HIRF chamber to meet DO-160F standards [8].

Power Supply Failure Detection

A problem of exposing these physical nodes to HIRF is permanent damage to the node's power supply [17]. In [17] brownouts in a power supply were found to be precursors to power supply failure. A brownout occurs when one or more boards in the node subjected to radiation experience a voltage drop [24],[25]. Thus, the fault-monitoring system needs to prevent brownouts as much as possible, have a good brownout detection mechanism and have a clear protocol for what to do if one occurs. During a power supply brownout, the node does not receive enough power to function, indicating a power supply failure. The node detects this internally by means of a voltage-sensing reset circuit on the CPU board, which causes the node to remain in reset until the voltage returns to the normal level for operation. When the node is in reset or initializing after a reset, it does not produce any outputs, i.e., behaves fail-silently, for a significant number of control cycles in a row. The first detection mechanism is to identify

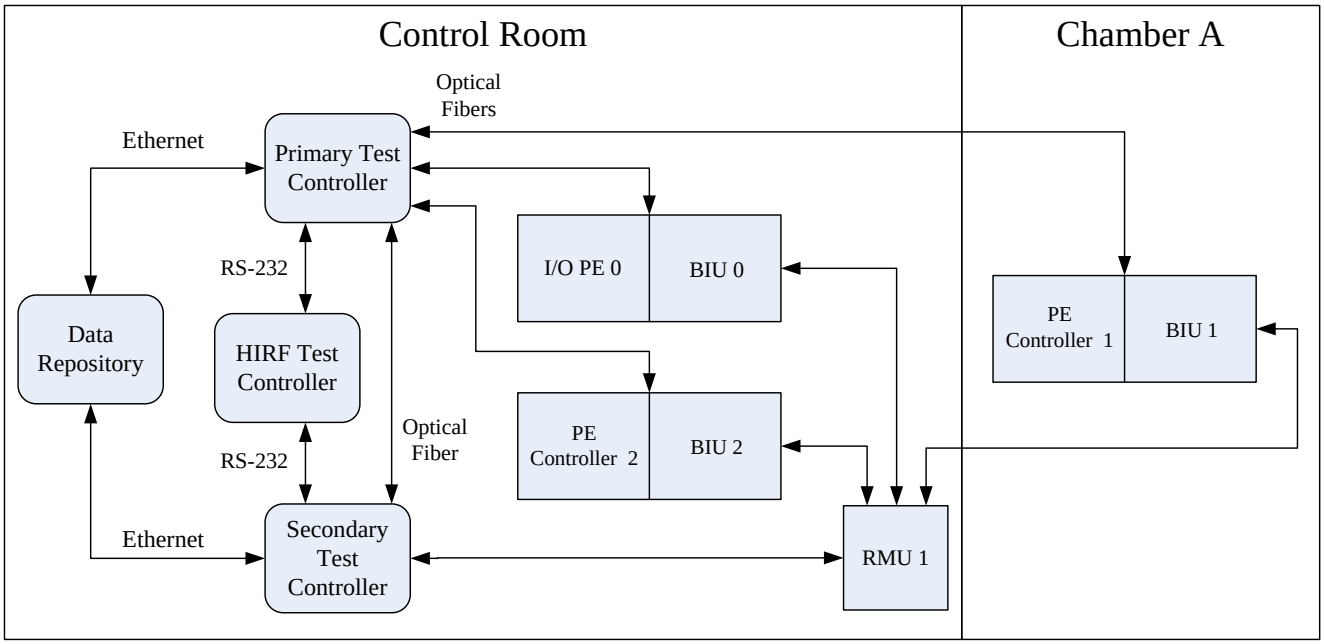

Figure 6. SPIDER System HIRF Experiment Configuration

when several control cycles of no output are seen in the FMon results. Over time, exposure to high field strengths can cause the current drawn by the power supply to start increasing. A second detection mechanism is to monitor for current trending up in the power supply. If either mechanism detects an impending power supply failure, the radiation is automatically turned off and the round would end to avoid damaging the node.

Procedure for Selecting Round Settings

The HIRF experiment consists of rounds where two main radiation parameters are varied: frequency (f) and field strength (|E|). Since the occurrence of errors ($z_1(k) = 1$ for a control cycle k) is frequency dependent, the field strength was gradually increased in 10 V/m steps until the percentage of control cycles having an error in a round was over 5%. This percentage is the sample time average of $z_1(k)$, i.e., the sample error mean given by

$$\hat{\mu}_{z_1,k} = \frac{1}{k}\sum_{i=1}^{k} z_1(i) .$$

When k equals the round's last control cycle, the sample error mean is denoted by $\hat{\mu}_{z_1}$. A near real-time running estimate of $\hat{\mu}_{z_1,k}$ was shown in a computer display monitored by the operator. This made it possible for the operator to manually stop the round if the estimate of $\hat{\mu}_{z_1,k}$ got to be too large.

There were two different types of rounds executed during the HIRF experiment: sweeps and Monte Carlo runs. A sweep consists of a set of 2-minute rounds for each of the eight frequencies and a range of field strengths in 10 V/m increments. Ten sweeps were completed at each frequency. The first round of a sweep starts with a field strength of 10 V/m and each subsequent round is 10 V/m higher than the previous one until $\hat{\mu}_{z_1} \approx 0.05$. All future sweeps start a few 10 V/m increments below the first field strength that showed at least one error. The highest field strength of the sweeps for one frequency is the same as for first sweep unless a brownout is detected. If there is a brownout, the rest of the sweeps are truncated at a lower field strength. A set of Monte Carlo runs consists of repeated trials with the same duration and a constant frequency and field strength. These sets were created for both the 2-minute and 90-minute durations.

Results of the HIRF Experiment

Rounds Completed

A total of 1813 rounds were executed during the HIRF experiment. Sweeps were conducted at each of the eight chosen frequencies. All the sweeps were 2

minutes long. They started at the lowest field strength of 10 V/m and increased in 10 V/m increments as planned until the operators deemed it unsafe to continue. The sweeps for 100 MHz, 121.15 MHz, and 133.35 MHz went up to 200 V/m, 140 V/m, and 210 V/m, respectively. After the first run of a sweep, the subsequent runs started about two levels lower than when the first errors were detected. The executed rounds for the other five frequencies resulted in $\hat{\mu}_{z_1}$ much less than 0.05 up to 250 V/m. For these frequencies, there were not a significant number of errors for the tested field strengths.

For the three frequencies where errors were observed in the sweeps, the maximum and mid-range field strengths were chosen to run at least 150 2-minute rounds as shown in Table 3. Then using the same stirrer rates that give an effective period of about 2.1 minutes, four 90-minute rounds for each frequency with the maximum field strength were completed. A single 90-minute round for each of the three frequencies at their respective maximum field strengths was run with the stirrer rates that give an effective period of 106.8 minutes. These are shown in Table 4.

After the Monte Carlo runs, extended sweeps were executed using higher field strengths as long as $\hat{\mu}_{z_1} < 0.18$ or the field strength reached 270 V/m.

Table 3. Monte Carlo Runs at Stirrer Rates of 7 and 18 sec / rev

f (MHz)	|E| (V/m)	Duration (min)	Total Rounds
100	200	2	151
121.15	140	2	150
133.35	210	2	150
100	180	2	150
121.15	120	2	163
133.35	190	2	150
100	200	90	4
121.15	140	90	4
133.35	210	90	4

Table 4. Single 90-Minute Runs at Stirrer Rates of 8 and 8.01 sec / rev

f (MHz)	|E| (V/m)
100	200
121.15	140
133.35	210

Function Monitor Observations

Since PE-BIU 0 and 2 are not subjected to radiated fields (see Figure 6), it is assumed that they never experience faults. Because of this assumption, each function monitor observation in the matrix in Table 2 that does not include PE-BIU 1 in the transmission path should always be 7. Thus, the four corners of the matrix, which correspond to the observations involving only PE-BIU 0 and 2 should be 7 for each control cycle. If, at any time during the HIRF experiment, these observations are not equal to 7 then that is a clear indication that the fault-injection and monitoring system is not functioning properly. Throughout the experiment, these observations were always 7.

During the Monte Carlo runs of the HIRF experiment, the function monitor observations only reported three different error type matrices for the control cycles. Error Type 0 is the case where everything worked properly and the data that was sent was received for that control cycle. The other two error types are given in Table 5 and Table 6 as Error Type 1 and Error Type 2, respectively. The observations for each round can be represented by the random process $\xi(k)$ taking values in $\{0,1,2\}$. By combining or *lumping* the last two error types into a single type, the previously defined random process $z_1(k)$ follows.

Table 5. Observations for Error Type 1

	FMon Tx 0	FMon Tx 1	FMon Tx 2
FMon Rx 0	7	4	7
FMon Rx 1	0	0	0
FMon Rx 2	7	4	7

Table 6. Observations for Error Type 2

	FMon Tx 0	FMon Tx 1	FMon Tx 2
FMon Rx 0	7	5	7
FMon Rx 1	7	5	7
FMon Rx 2	7	5	7

The HIRF experiment design allows for a partial characterization of the causes of Error Types 1 and 2 during a control cycle. The main assumption is that only the node located in the HIRF chamber can experience errors. The goal of this fault analysis is not to perform fault isolation, i.e., pinpoint exactly where within the node the faults occurred. This is not needed because of the fail-silent assumption that as long as the fault is detected, it will not produce outputs in the closed-loop system analysis. The node under test has two types of communication links: PTLs to the PTC and ROBUS links (RLs) to the RMU. Each of these types of links is a pair of transmit and receive links. Since the function monitors report observations (comparisons of sent and received data) through both types of links, only errors manifested in at least one of the four links of the node in the chamber can result in an error observation other than 7. From a previous experiment where the physical nodes were placed in a HIRF environment [17], most errors were found to be caused by communication link faults.

One possible cause for Error Type 1 is located in the link from the function monitor transmitter to the node receiver. At the beginning of the control cycle, the node in the chamber is supposed to receive a synchronization message (see [14]) and a function monitor set of command data through its link to the PTC. In order for a PE-BIU node to remain in normal operation, it must receive the synchronization message within a pre-specified interval during a ROBUS cycle. If there is an error with this message detected at the receiver, then the PE-BIU node declares a failure, triggers a reset, and goes to an initialization mode [14]. The node may remain in this mode for multiple control cycles depending on the duration of the fault. During these control cycles, the function monitor observations will be as in Table 5 for Error Type 1. The FMon message is a data message that is not critical to the operation of the SPIDER system unlike the synchronization message, which is why they cause different errors in the same link.

The link from the function monitor transmitter to the node receiver can also cause Error Type 2. When an error in the function monitor data is detected by the node in the chamber, it asserts an error flag in the header of the data message, and broadcasts it through the RMU according to the schedule. The RMU then transmits this message to all of the PE-BIUs, where it is then forwarded to their associated function monitors in the PTC. This error flag is then translated into a 5 in the observations and is located in the column for the PE-BIU in the chamber transmitting as shown in Table 6 for Error Type 2.

The communication link from the node in the chamber to the PTC is used to send a data message during a control cycle. If the receiver at the function monitor detects a format or CRC packet error, then it rejects the entire message thus reporting a 0 observation. In this case, all other FMon observations should be 7. Since this FMon matrix did not occur during the HIRF Experiment Monte Carlo runs, it is highly unlikely that an error on this link occurred.

The third communication link connects the transmitter of the RMU to the receiver of the PE-BIU in the HIRF chamber via a RL. If an error is detected at the PE-BIU receiver of this link, then the BIU diagnoses the RMU as being faulty. Since this experiment only has a single RMU and the BIU in the chamber does not trust this RMU, this essentially means the PE-BIU node is disconnected from the bus. Since the PE-BIU is not able to communicate on the bus, it declares a failure. Similar to when the PE-BIU detected an error with the synchronization message from the PTC, the failure triggers a reset and the node returns to initialization mode where the FMon observations are recorded as in Table 5.

The final communication link involving the node under test connects the PE-BIU RL transmitter to the RMU receiver. If an error is found on a message from the PE-BIU in the chamber, the RMU diagnoses that node as bad. If the diagnosis remains when ROBUS executes the global diagnostic protocol at the end of the diagnostic cycle [14], the PE-BIU subject to HIRF is removed from the system and goes into reset. This is another failure condition within the PE-BIU node that triggers a node reset and results in the observations of Table 5. Also, whenever the PE is scheduled to transmit and the RMU has diagnosed it as faulty, the RMU replaces

the message from the PE with a SOURCE_ERROR message and broadcasts it to all of the PE-BIUs [14]. The BIU of the node in the chamber recognizes that the message received is not the same as the message that was sent, which means that either the PE-BIU node is faulty or the single RMU node is faulty. For both of these cases, the PE-BIU goes into recovery and the PE stops sending messages to its function monitor, which results in Table 5 as well. As long as the node remains in the initialization phase, the FMon will not receive updates from the PE, so the Error Type 1 remains.

Given that only the node in the chamber is subject to faults and only the data at the interfaces of the node are affected, all faults are shown with at least one FMon observation other than 7 for the control cycle. This means that all faults in the system will show up in the FMon observations. Fault-Tree Analysis is used to depict the causes of the error types found during the experiment [26],[27] as shown in Figure 7. There is no way to distinguish which communication link caused Error Type 1 using only FMon observations. Error Type 2 observed in the HIRF Experiment only had one root cause satisfying our assumptions as shown in Figure 8.

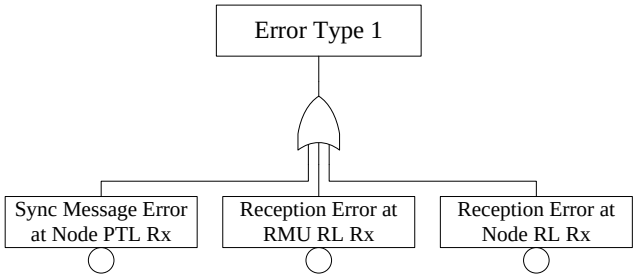

Figure 7. FMon Observations for Error Type 1

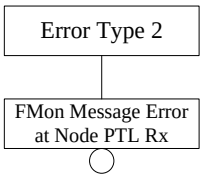

Figure 8. FMon Observations for Error Type 2

Data Analysis

Preliminary analysis of the 2-minute random sequences $\xi(k)$ and $z_1(k)$ for the three sets of sweeps and six sets of Monte Carlo runs shown in Table 3 are presented in this section. For each control cycle k, $\xi(k)$ and $z_1(k)$ are discrete random variables. Their probability distributions can be estimated via ensemble averages. To analyze the sweeps, where only ten rounds at each frequency and field strength were recorded, ensemble averages would be poorly estimated. On the other hand, sample time averages give better estimates, since each 2-minute round has 20,880 samples. When the ensemble averages of each random variable are constant and finite, the sample time averages can be used to estimate the ensemble mean, if the random processes $\xi(k)$ and $z_1(k)$ are ergodic in the mean [28]. Under these assumptions, the sample time averages of $z_1(k)$ were analyzed. Since this process at each time instant is either 0 or 1, the sample time averages are in fact estimating the probability that an error occurred during a control cycle, $P\{z_1(k)=1\}$. The error probabilities for each round of the sweeps were calculated, resulting in ten estimates for each frequency and field strength. The average and standard deviation of these estimates are shown as error bars in Figure 9. As expected, the higher field strengths produced higher error probabilities. This error probability is frequency dependent with much higher errors resulting at 121.15 MHz than for 100 and 133.35 MHz. The figure shows an artifact due to the way the experiment was conducted. In each case, the extended sweeps exposed the node to a different electromagnetic environment since the phase difference of the stirrers was changed when they were made to rotate with an effective period of 106.8 minutes during the Monte Carlo runs in Table 4, which were executed before the extended sweeps. This change resulted in lower error probabilities during the extended sweeps. If the extended sweeps are not taken into account, the average probabilities for each sweep are well approximated by a quadratic or cubic polynomial. Cubic polynomial fits of the sweeps and their predicted values for higher field strengths are also shown in Figure 9. The error probabilities of the sweep at 121.15 MHz start about 15 and 635 times higher than the error probabilities of the sweeps at 100 MHz and 133.35 MHz, respectively. The cubic fit extrapolations predict that at 121.15 MHz, the error probability will reach 1 at 267 V/m. At this field strength, the error probability of the sweep at 121.15 MHz is about 7 and 5.6 times higher than the error probabilities of the sweeps at

100 MHz and 133.35 MHz, respectively. The error probabilities of the sweeps at 100 MHz and 133.35 MHz are about the same below 180 V/m. As the field strength increases, the approximations predict that the error probabilities at 133.35 MHz will be about 1.2 times those at 100 MHz. It is surprising that the error probability predictions of the fits at 100 MHz match well the extended sweep measurements at 133.35 MHz. Figure 9 also shows the error probabilities between 80 V/m to 290 V/m estimated using another sweep at 100 MHz. It is denoted as *Sweep* 0, since the position of the node and its cables is slightly different than in the rest of the HIRF experiment. Nevertheless, the error probabilities of this sweep are similar to the other estimates up to 200 MHz, and they are also consistent with the predictions up to 290 V/m except for the error probability estimates at 270 V/m and 280 V/m.

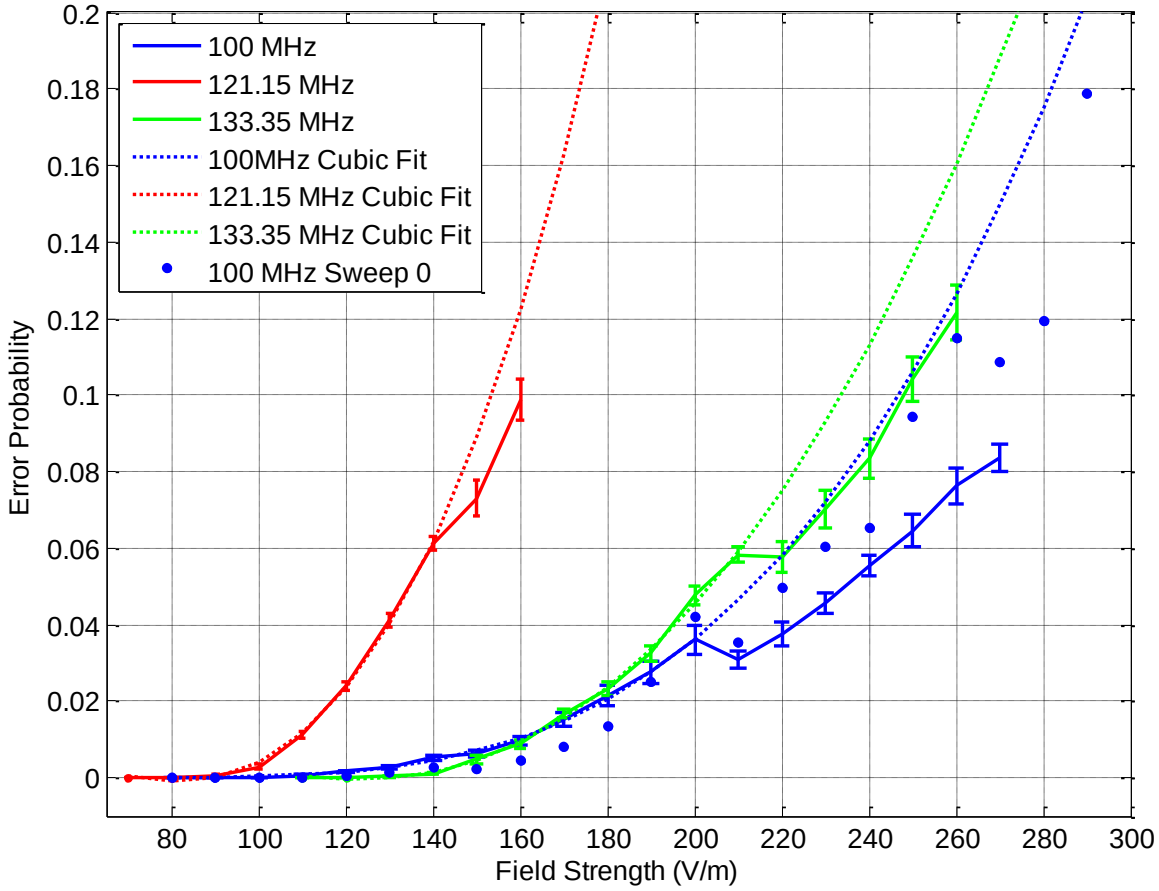

Figure 9. Estimates of Error Probabilities for each Sweep

To analyze the sets of Monte Carlo runs, it is possible to approximate the probabilities of each error type by averaging over the ensemble. The arithmetic average of the ensemble probability estimates for $\xi(k)$, denoted by $\hat{P}\{\xi(k)=i\}$, i=0,1,2, is given in Table 7. Note that

$$\hat{P}\{z_1(k)=0\}=\hat{P}\{\xi(k)=0\}$$

and

$$\hat{P}\{z_1(k)=1\}=\hat{P}\{\xi(k)=1\}+\hat{P}\{\xi(k)=2\}.$$

The averages of these probabilities are given in Table 7. A visualization of the error probabilities, $\hat{P}\{z_1(k)=1\}$, is given in Figure 10. The box plots of each Monte Carlo set denote the medians with a red

line, and the top and bottom of the boxes correspond to the 25th and 75th percentiles, respectively. The data outside of the whiskers is considered an outlier. Only the 100 MHz at 200 V/m had outliers. The boxplot also shows that the data is skewed, since the median is not near the center. These error probability estimates were then compared to the sample time averages of the Monte Carlo runs. For each error type the sample time average probability estimate is almost the same as the ensemble average. This result reassured us to make the sample time average analysis of the sweeps.

Table 7. Error Type Probabilities for 2-Minute Monte Carlo Runs

f (MHz)	|E| (V/m)	$\hat{P}\{\xi(k)=0\}$	$\hat{P}\{\xi(k)=1\}$	$\hat{P}\{\xi(k)=2\}$
100	180	0.9848	0.0119	0.0033
100	200	0.9663	0.0286	0.0051
121.15	120	0.9819	0.0140	0.0040
121.15	140	0.9355	0.0570	0.0075
133.35	190	0.9840	0.0130	0.0031
133.35	210	0.9454	0.0479	0.0067

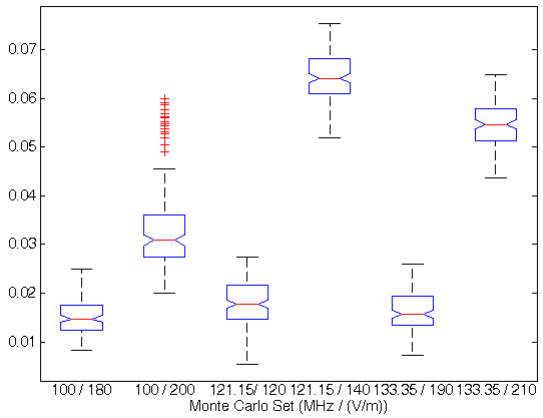

Figure 10. $\hat{P}\{z_1(k)=1\}$ **for 2-Minute Monte Carlo Runs**

Conclusion

A HIRF fault-injection experiment was designed, implemented, and conducted at the NASA Langley Research Center's HIRF Laboratory. The experiment consisted of a 2×1 SPIDER system running in simulated closed-loop with a test controller that provided sensor and control commands as well as monitored for incorrect transmissions in the SPIDER system. For this experiment one of the SPIDER nodes simulating a Control Law computational node was subjected to radiation in a HIRF chamber while the rest of the 2×1 system was not exposed to HIRF and assumed not to fail during the experiment. HIRF parameter settings were selected to inject faults in the system without causing permanent damage to the node. Out of the eight selected frequencies, three triggered errors in the node for field strengths that did not lead to permanent errors.

Results from two main types of experimental rounds, sweeps and Monte Carlo runs, were presented. The time averages were used to estimate the probability of errors during the sweeps. It was shown that these errors can be fitted with a polynomial, which could be used to predict the error probability at higher field intensities. A preliminary statistical analysis of the Monte Carlo data collected during the physical HIRF fault-injection experiment was also presented. The data showed that time averages can be used to estimate the ensemble averages. During all the Monte Carlo runs, most of the control cycles passed with no errors and only two types of errors were detected. These errors could have been caused by a number of different types of faults in the communication links of the node in the chamber. A more comprehensive analysis of the data and their effect in a flight control system will appear in future publications.

References

[1] Rushby, J., October 2001, Bus Architectures for Safety-Critical Embedded Systems, EMSOFT 2001: First Workshop on Embedded Software, Springer-Verlag Lecture Notes in Computer Science, vol. 2211, Lake Tahoe, CA, pp. 306-323.

[2] Mendell, R. B., S. A. Korff, 1963, Fast-Neutron Flux in Atmosphere, *Journal of Geophysical Research*, vol. 68, pp. 5487-5495.

[3] Zhang, H., W. S. Gray, O. R. González, A. V. Lakdawala, 2009, Tracking Performance of a Recoverable Flight Control System in Neutron Environments, *IEEE Transactions on Aerospace and Electronic Systems*, vol. 45, pp. 321-335.

[4] Clough, B. T., 1996, Effects of Electromagnetic Interference on Digital Control, Wright Laboratory,

Wright-Patterson AFB, OH, WL-TR-96-3122.

[5] Shooman, M. L., 1994, A Study of Occurrence Rates of Electromagnetic Interference (EMI) to Aircraft with a Focus on HIRF (External High Intensity Radiated Fields), NASA CR-194895.

[6] S. A. E. Aerospace, 2003, Guide to Certification of Aircraft in a High Intensity Radiated Field (HIRF) Environment, ARP5583, S.A.E.

[7] FAA, 2007, The Certification of Aircraft Electrical and Electronic Systems for Operation in the High-Intensity Radiated Fields (HIRF) Environment.

[8] RTCA Program Management Committee, 2007, Environmental Conditions and Test Procedures for Airborne Equipment, DO-160F.

[9] Miner, P. S., June 2000, Analysis of the SPIDER Fault-Tolerance Protocols, LFM 2000: Fifth NASA Langley Formal Methods Workshop, NASA Langley Research Center, Hampton, VA.

[10] Torres-Pomales, W., M. R. Malekpour, P. S. Miner, March 2005, ROBUS-2: A Fault-Tolerant Broadcast Communication System, NASA TM-2005-213540.

[11] Torres-Pomales, W., M. R. Malekpour, P. S. Miner, November 2005, Design of the Protocol Processor for the ROBUS-2 Communication System, NASA TM-2005-213934.

[12] Girault, A., E. Rutten, 2005, Modeling Fault-tolerant Distributed Systems for Discrete Controller Synthesis, *Electronic Notes in Theoretical Computer Science*, vol. 133, pp. 81-100.

[13] Tejada, A., O. R. González, W. S. Gray, 2008, Stability of Digital Control Systems Implemented in Error-Recoverable Computers, *International Journal of Control*, vol. 81, pp. 1665-1681.

[14] Torres-Pomales, W., A. M. Yates, M. R. Malekpour, August 2010, Fault-Injection and Monitoring Capability for a Fault-Tolerant Distributed Computation System, NASA TM-2010-216834.

[15] Yates, A. M., Fault-Injection Experiment for the Statistical Characterization of Faults in an Avionics Communication System, Master Thesis, Old Dominion University, Norfolk, VA, to appear.

[16] Gray, W. S., R. Wang, O. R. González, J. R. Chávez-Fuentes, June 2010, Tracking Performance Analysis of a Distributed Recoverable Boeing 747 Flight Control System Subject to Digital Upsets, Proceedings of the 2010 American Control Conference, Baltimore, Maryland, pp. 548-554.

[17] Torres-Pomales, W., M. R. Malekpour, P. S. Miner, S. V. Koppen, May 2008, Plan for the Characterization of HIRF Effects on a Fault-Tolerant Computer Communication System, NASA TM-2008-215306.

[18] Fuller, G. L., 1995, Understanding HIRF – High Intensity Radiated Fields, Aviation Communications, Inc., Leesburg, VA.

[19] Butler, R. W., February 2008, A Primer on Architectural Level Fault Tolerance, NASA TM-2008-215108.

[20] Ladbury, J., G. Koepke, D. Camell, 1999, Evaluation of the NASA Langley Research Center Mode-Stirred Chamber Facility, National Institute of Standards and Technology Technical Note 1508.

[21] Freyer, G. J., M. O. Hatfield, D. M. Johnson, M. B. Slocum, October 1996, Characterization of the Electromagnetic Environment in Aircraft Cavities Excited by Internal and External Sources, *Proceedings of the 15th Digital Avionics Systems Conference*, Atlanta, GA, pp. 327-332.

[22] Poggio, A. J., R. A. Zacharias, S. T. Pennock, C. A. Avalle, H. Carney, April 1996, NASA B-757 HIRF Test Series Low Power On-The-Ground Tests, *Aerospace and Electronic Systems Magazine*, vol. 11, no. 4, pp. 27-33.

[23] Agilent Technologies, Inc., 2005, Agilent VEE Pro: VEE Pro User's Guide, 9th ed.

[24] Maxim Integrated Products, 2004, Application Note: 3227 Power-On Reset and Related Supervisory Functions.

[25] Wadhwa, S. K., G. K. Siddhartha, A. Gaurav, 2006, Zero Steady State Current Power-on-Reset Circuit with Brown-Out Detector, *Proceedings of the 19th International Conference on VLSI Design*, Hyderabad, India, pp. 631-638.

[26] Lee, W. S., D. L. Grosh, F. A. Tillman, C. H. Lie, 1985, Fault Tree Analysis, Methods, and Applications - A Review, *IEEE Transactions on Reliability*, vol. R-34, no. 3, pp. 194-203.

[27] Zampino, E. J., 2001, Application of Fault-Tree Analysis to Troubleshooting the NASA GRC Icing Research Tunnel, *Proceedings of the Annual Reliability and Maintainability Symposium*, Philadelphia, PA, pp. 16-22.

[28] Stark, H., J. W. Woods, 2002, *Probability and Random Processes with Applications to Signal Processing*, 3rd ed., Prentice Hall, Upper Saddle River, New Jersey.

Acknowledgements

This research was supported by the NASA Langley Research Center under grant NNX07AD52A.

The authors wish to thank Sandra Koppen for her extensive aid in the design and implementation of the HIRF experiment. The authors also appreciate Laura Smith and the NASA HIRF Laboratory staff for their assistance in conducting this experiment.

Email Addresses

A. M. Yates ayate004@odu.edu

W. Torres-Pomales w.torres-pomales@nasa.gov

M. R. Malekpour mahyar.r.malekpour@nasa.gov

O. R. González ogonzale@odu.edu

W. S. Gray sgray@odu.edu