FOR UNRESTRICTED DISTRIBUTION
DATE _____ WEC

Westinghouse Energy Systems

9203040111 920226
PDR ADOCK 05000390
A PDR

rec'd with letter dtd 2/26/92

WESTINGHOUSE CLASS 3

WCAP-12375
REV. 1

TOPICAL REPORT
EAGLE-21
MICROPROCESSOR-BASED
PROCESS PROTECTION SYSTEM

L. E. ERIN

DECEMBER, 1991

Approved: P. J. Morris

Manager,
Instrumentation and
Control Systems Licensing
Nuclear and Advanced
Technology Division

Westinghouse Electric Corporation
Nuclear and Advanced Technology Division
P. O. Box 355
Pittsburgh, Pennsylvania 15230

(C) 1991 Westinghouse Electric Corporation
All Rights Reserved

ACKNOWLEDGEMENTS

The author wishes to express his appreciation to Carl A. Vitalbo of the Westinghouse Process Control Division whose help was instrumental in the preparation and completeness of this topical report.

TABLE OF CONTENTS

ABSTRACT

1.0 INTRODUCTION

2.0 DESIGN PHILOSOPHY AND FEATURES

2.1 Form-Fit-Function Concept

2.1.1 Typical Analog Process Channel
2.1.2 Typical Eagle-21 Process Channel

2.2 Installation

2.3 Design Features

2.3.1 Single Failure Criterion
2.3.2 Instrument Power Source
2.3.3 Channel Integrity
2.3.4 Channel Independence
2.3.5 Control and Protection System Interaction
2.3.6 Automatic Surveillance Testing
2.3.7 Self Calibration
2.3.8 Channel Bypass
2.3.9 Access to Setpoint and Tuning Constant Adjustments
2.3.10 Diagnostics
2.3.11 Failure Modes

3.0 TECHNICAL DESCRIPTION

3.1 Eagle-21 Architecture

3.1.1 Input/Output (I/O) Subsystem
3.1.2 Loop Processor Subsystem
3.1.3 Tester Subsystem
3.1.3.1 Man-Machine-Interface (MMI)

TABLE OF CONTENTS (cont)

3.2 Eagle-21 Hardware Description

3.2.1 Analog Input Module
3.2.2 Contact Input Module
3.2.3 Analog Output Module
3.2.4 Contact Output Module
3.2.5 Trip Output Module
3.2.6 Microprocessor Card Chassis Modules

3.2.6.1 [ ] a,c
3.2.6.2
3.2.6.3
3.2.6.4
3.2.6.5
3.2.6.6

3.2.7 Miscellaneous Hardware

3.2.7.1 Microprocessor Card Chassis
3.2.7.2 Test Panel
3.2.7.3 DC Power Supply Chassis
3.2.7.4 Power Distribution and Filtering
3.2.7.5 Termination Frame
3.2.7.6 Cabinet Cooling Assembly

3.3 Software

3.3.1 Software Development
3.3.2 Software Implementation

TABLE OF CONTENTS (cont)

4.0 EQUIPMENT QUALIFICATION

4.1 Equipment Qualification Background
4.2 Equipment Qualification Program Description

4.2.1 Environmental Testing
4.2.2 Seismic Testing

4.3 Equipment Qualification Documentation

5.0 NOISE, FAULT, SURGE WITHSTAND CAPABILITY, AND RADIO FREQUENCY INTERFERENCE (RFI) TESTS

5.1 Test Description

5.1.1 Noise Tests
5.1.2 Fault Tests
5.1.3 Surge Withstand Capability (SWC) Tests
5.1.4 Radio Frequency Interference (RFI) Tests

5.2 Test Documentation

6.0 DESIGN, VERIFICATION AND VALIDATION PLAN

6.1 Background
6.2 Applicable Standards

7.0 COMPLIANCE WITH CRITERIA

7.1 IEEE Std. 279-1971

APPENDIX A "Eagle 21 Replacement Hardware Design, Verification and Validation Plan"

LIST OF FIGURES

Figure	Title
1-1	Eagle-21 Implementation
2-1	Eagle-21 Design Philosophy
2-2	Typical Analog Process Protection Channel
2-3	Eagle-21 Existing Cabinet Installation
3-1	Eagle-21 Subsystems
3-2	Eagle-21 Input/Output Subsystem
3-3	Eagle-21 Loop Processor Subsystem
3-4	Eagle-21 Tester Subsystem
3-5	Eagle-21 Architecture
3-6	Analog Input Functional Configuration
3-7	Contact Input Functional Configuration
3-8	Analog Output Functional Configuration
3-9	Contact Output Functional Configuration
3-10	Partial Trip Output Functional Configuration
3-11	Eagle-21 Software Development
3-12	Eagle-21 Layered Software Approach

ACRONYMS and ABBREVIATIONS

A/D	Analog to Digital
AC	Alternating Current
DC	Direct Current
cfm	cubic feet per minute
D/A	Digital to Analog
DFP	Digital Filter Processor
EAI	Eagle-21 Analog Input
EAO	Eagle-21 Analog Output
EMI	Electro-Magnetic Interference
EPT	Eagle-21 Partial Trip
ESF	Engineered Safety Features
GHz	Giga-Hertz
Hz	Hertz
I/O	Input/Output
IEEE	Institute of Electrical and Electronics Engineers
IPS	Integrated Protection System
LCP	Loop Calculation Processor
LED	Light Emitting Diode
mH	milli-Henries
MHz	Mega-Hertz
MMI	Man-Machine-Interface
NRC	Nuclear Regulatory Commission
OBE	Operating Basis Earthquake
PROM	Programmable Read Only Memory
QDPS	Qualified Display Processing System
RFI	Radio Frequency Interference
RT	Reactor Trip
RTD	Resistance Temperature Detector
SIR	Signal Injection and Response
SSE	Safe Shutdown Earthquake
SWC	Surge Withstand Capability
TSP	Test Sequence Processor
V&V	Verification & Validation

ABSTRACT

Process Instrumentation is comprised of those devices (and their interconnection into systems) which measure and process signals for temperature, pressure, fluid flow, and fluid levels. Process instrumentation specifically excludes nuclear and radiation measurements.

Process Instrumentation includes equipment which performs functions such as: process measurement, signal conditioning, dynamic compensation, calculations, setpoint comparison, alarm actuation, indication and recording, which are all necessary for day-to-day operation of the Nuclear Steam Supply System as well as for monitoring the plant and providing initiation of protective functions upon approach to unsafe plant conditions. The Westinghouse Eagle-21 microprocessor based process protection upgrade system is applicable for those instrument systems which are "safety-related" as defined by IEEE Std. 279-1971, "Criteria for Protection Systems for Nuclear Power Generating Stations". The Eagle-21 portion of process instrumentation includes all necessary devices with the exception of transmitters, indicators, and recorders.

The Westinghouse Eagle-21 microprocessor-based process protection system is a functional replacement for existing analog process protection equipment used to monitor process parameters at nuclear generating stations and initiate actuation of the reactor trip and engineering safeguards systems.

1.0 INTRODUCTION

The majority of nuclear power generation stations presently employ analog process protection equipment. This equipment was designed in the 1960's and early 1970's. As illustrated in Figure 1-1, the analog protection system receives inputs from sensors, provides information to the operator, performs calculations on these values, and compares the results to allowable limits. If the limits are exceeded, a partial reactor trip is generated. External logic performs a voting algorithm on the partial trips from the four redundant protection sets, and conditionally generates a reactor trip. A similar path exists for the generation of engineered safeguard system actuations. These actuations mitigate the effects of an undesired event. The process protection system also provides isolated signals for use by non-safety systems such as the control system, the plant computer, and portions of the control board.

Westinghouse Process Protection Systems include three generations of analog electronics: Foxboro H-Line, Westinghouse 7100 Series, and Westinghouse 7300 Series Equipment.

The first generation of analog process protection equipment was Foxboro H-Line which is described in WCAP 7671 "Topical Report - Process Instrumentation for Westinghouse Nuclear Steam Supply Systems." This equipment was manufactured for use during the 1965 - 1972 time frame. Twenty-five nuclear generating stations utilize this equipment.

The second generation of analog process protection equipment was the Westinghouse 7100 Series, also described in WCAP 7671. This equipment was manufactured for use during the 1970 - 1973 time frame. Thirteen nuclear generating stations utilize this equipment.

The third generation of analog process protection equipment was the Westinghouse 7300 Series, described in WCAP 7913 "Process Instrumentation for Westinghouse Nuclear Steam Supply Systems (4 Loop Plants Using WCID 7300 Series Process Instrumentation). This equipment was manufactured for use during the 1973 - 1983 time frame. Forty-four nuclear generating stations utilize this equipment.

As a result of technological advances, the earlier analog process protection systems are rapidly approaching the point of obsolescence. Additionally, utility personnel have identified the following difficulties with the analog systems:

A. Time consuming calibration and surveillance test procedures.
B. Extensive maintenance time for troubleshooting and repair.
C. Difficulty in maintaining equipment qualification.
D. Difficulty in maintaining adequate spare parts inventory.
E. Lack of expansion space to install hardware for functional upgrades and plant improvements.

The Westinghouse Eagle-21 Process Protection System is a modular microprocessor based upgrade system for replacing the existing analog process protection equipment. Features of the Eagle-21 equipment include the following:

A. Automatic surveillance testing to significantly reduce the time required to perform surveillance tests.

B. Self calibration to eliminate rack drift and time consuming calibration procedures.

C. Self diagnostics to reduce the time required for troubleshooting.

D. Significant expansion capability to allow for rack consolidation and easily accommodate functional upgrades and plant improvements.

E. Modular design to allow for a phased installation into existing process racks and use of existing field terminations.

2.0 DESIGN PHILOSOPHY AND FEATURES

2.1 Form-Fit-Function Concept

The Eagle-21 Process Protection System, as shown in Figure 2-1, is a digital form, fit, and functional replacement for the existing analog equipment. All system inputs (from plant sensors) and system outputs (reactor trip logic, engineered safety features logic, indication and control) are preserved. Thus, the installation of Eagle-21 process equipment has no affect on the existing external interfaces.

2.1.1 Typical Analog Process Channel

A typical analog process protection instrument channel is shown in Figure 2-2. A field sensor is connected to cabinet mounted terminal blocks. The process electronics power the field sensor and perform signal conditioning, calculation, trip logic, and isolation operations on the input signal. Each element of the process is an individual electronic module or printed circuit board assembly. Typical functions performed by these modules are as follows: loop power supply, summation, lead/lag, multiplication, comparator, square root, amplification, signal conversion, and isolation.

2.1.2 Typical Eagle-21 Process Channel

In a typical Eagle-21 Process Protection Instrument Channel, field sensors are connected to cabinet mounted terminal blocks. The process electronics power the sensors and perform signal conditioning, calculation, and isolation operations on the input signals. However, each element of the process is not an individual electronic module or printed circuit board assembly. A multiple channel Analog Input module is used to power the field sensor(s) and perform signal conditioning. All calculations for the process channel functions are performed by a centralized Loop Calculation Processor (LCP). Typical functions performed by the Loop Calculation Processor are as follows: summation, lead/lag, multiplication, comparator, averaging, and square root conversion. Trip logic is provided through multiple channel Partial Trip Output modules. Multiple channel isolated analog outputs are provided by Analog Output modules. In addition, all Eagle-21 process protection channels are configured to perform automatic surveillance testing via a centralized Test Sequence Processor (TSP).

Typical protection channels which may be processed with the Eagle-21 Process Protection System are as follows:

A. Average Temperature and Delta Temperature
B. Pressurizer Pressure
C. Pressurizer Water Level
D. Steam Flow and Feedwater Flow
E. Reactor Coolant Flow
F. Turbine Impulse Chamber Pressure
G. Steam Pressure
H. Containment Pressure
I. Reactor Coolant Wide Range Temperatures
J. Reactor Coolant Wide Range Pressure
K. Boric Acid Tank Level
L. Pressurizer Liquid and Vapor Temperatures
M. Steam Generator Narrow Range and Wide Range Water Level
N. RHR Pump Discharge Temperature
O. Refueling Water Storage Tank Level
P. Containment Sump Level
Q. Containment Spray Pump Header Flow
R. Condensate Storage Tank Level

2.2 Installation

The Eagle-21 Process Protection System is a modular electronics upgrade package for the existing analog plant process protection equipment. The Eagle-21 equipment has been designed to fit into existing process racks and to interface with other plant systems in a manner identical to the existing analog equipment. The standard design maintains the existing field terminals to avoid new cable pulls or splices within the rack. The Eagle-21 equipment design also allows for rack consolidation of process protection channels resulting in a reduction of the total number of racks required for a protection system upgrade. The components for each rack are built into subassemblies which can be easily installed into the existing racks. All internal rack cabling is pre-fabricated. The subassemblies are tested in a factory mock-up to verify proper fit and operation. Detailed installation procedures and drawings are provided with each system.

An example of Eagle-21 hardware being installed into an existing process rack is depicted in Figure 2-3.

2.3 System Design Features

2.3.1 Single Failure Criterion

The Eagle-21 Process Protection System is designed to provide three or four instrumentation channels and outputs to two trip logic trains for each protective function. These redundant channels and trains are electrically isolated and physically separated. Thus, any single failure within a channel or train will not prevent a required protective system action.

2.3.2 Instrument Power Source

Electrical power for the Eagle-21 Process Protection System instrumentation is obtained from four separate instrument busses that are equal to each other in reliability and quality of the power available. The arrangement of the four busses with respect to the ultimate power source is covered in detail in the FSAR for each plant. The use and availability of the four busses is important to the plant instrumentation in the following ways:

A. Each of the four protection sets is assigned to one of the instrument busses and no other.

B. Instrument channels are arranged so that loss of any one bus will not force a trip of the reactor. However, all reactor trip comparators and most of the safeguards comparators will trip in that protection set. (e.g. all 2 out of 3 reactor trip logic will revert immediately to a condition of 1 out of 2 logic.)

C. Loss of any one bus will not put the plant in an unprotected condition.

D. Coincident loss of any two busses will trip the reactor immediately as a result of the preferred failure mode of the comparators and initiate

most safeguards action associated with those protection sets (e.g. two of the logic inputs for each associated 2 out of 3 or 2 out of 4 logic will immediately exist as trip signals).

2.3.3 Channel Integrity

The Eagle-21 Process Protection System has been designed to operate and maintain necessary functional capability under extremes of conditions relating to environment, energy supply, malfunctions and accidents. The environmental and energy supply extremes throughout which the system will perform are detailed in Sections 4.0 and 5.0.

2.3.4 Channel Independence

Within the Eagle-21 Process Protection System, there are four separate and independent rack sets. Channels which provide signals for the same protective functions are each located in different rack sets ensuring that they will be independent and physically separated. Since all equipment within any rack is associated with a single Protection Channel Set, there is no requirement for separation of wiring and components within the rack.

2.3.5 Control and Protection System Interaction

The Eagle-21 Process Protection System functions completely independent from the control systems. Its' operation in protecting the plant from unsafe conditions is not affected by any fault or malfunction in the control systems.

The transmission of signals from the Eagle-21 Process Protection System to the control systems is through isolation devices that are classified as part of the protection system. No credible fault at the output of an isolation device can prevent the associated Eagle-21 Protection System channel from meeting the minimum performance requirements specified in the design bases. Fault testing of the isolation devices is described in more detail in Section 5.0 of this document.

The same type of electrical isolation is also used to separate from the

Eagle-21 Protection System, those signals (such as RCS average temperature), which are required and used to control actual plant variables. For this use, however, consideration must be given to possible protection channel failures that can both prevent a particular trip signal from that channel and cause the control system to drive the plant toward the unsafe condition for which the particular trip signal is needed. In each case where this is possible, either four protection channels have been provided and 2 out of 4 logic is used to ensure the plant remains fully protected even when degraded by a second random failure, a control system median signal selector is used to prevent control system reaction to the initiating protection system failure or a diverse means for providing a reactor trip is available.

2.3.6 Surveillance Testing

The Eagle-21 Process Protection System accomodates automatic or manual surveillance testing of the digital process protection racks via a portable or rack installed Man Machine Interface (MMI) test unit. The MMI test unit is connected to the process rack by inserting a cable/connector assembly into the process rack test panel. Using the MMI, the "Surveillance Test" option is then selected. Following instructions entered through the MMI, the rack installed test processor and the user interactively perform the following operations:

1. Selection of the individual process channel to be tested.

2. Selection of Automatic or Manual test mode.

3. Placement of the individual channel trip outputs in either "Channel Trip" or "Bypass" (password protected) mode.

   A. Bypass Mode: disables the individual channel comparator trip circuitry which forces the associated logic input relays to remain in the non-tripped state until the "bypass" is removed.

   B. Channel Trip Mode: interrupts the individual channel comparator outputs to the logic circuitry to de-energize the associated logic input relay(s).

4. Calibration of the test reference signals and verification of the tester time base.

5. Determination of subtests to be performed.

6. Activation of the test injection signal.

7. Performance of Analog to Digital (A/D) converter accuracy test.

8. Calculate minimum and maximum deviation from comparator setpoint(s).

9. Perform partial trip actuation test.

10. Perform channel time response test.

11. Perform loop cycle time deviation tests.

12. Completion of test cycle and automatically remove "Channel Trips" or "Bypass".

13. Verify calibration of the test injection signals.

14. Display test results on the MMI screen.

Interruption of the comparator output to the logic circuitry for any reason (test, maintenance purposes, or removed from service) causes that portion of the logic to be actuated and accompanied by a channel trip alarm and channel status light in the control room. Status lights on the process rack test panel indicate when the associated comparators have tripped. Each channel is fully testable via the MMI test unit.

2.3.7 Calibration

The Eagle-21 Process Protection System provides for continuous on-line self-calibration of analog input signals. The Digital Filter Processor (DFP) addresses high and low reference signals via a multiplexer circuit on each

analog input channel. The Loop Calculation Processor (LCP) then compares the output of the DFP Analog to Digital (A/D) Converters to stored high and low reference values to determine if any errors have been introduced by analog signal processing and A/D conversion. If necessary, the LCP automatically compensates gain and offset coefficients to eliminate any errors that have been introduced.

2.3.8 Channel Bypass

The Eagle-21 Process Protection equipment is designed to permit any one channel to be maintained in a bypassed condition, and when required, tested during power operation without initiating a protective action at the systems level. This accomplished without lifting electrical leads or installing temporary jumpers.

If an Eagle-21 protection channel has been bypassed for any purpose, a signal is provided to allow this condition to be continuously indicated in the control room.

The Eagle-21 design has provided for administrative controls and multiple levels of security for bypassing a protection channel. To place a protection channel in bypass, an individual must have access to the following:

a,c

The material in this section has been deleted from this non-proprietary document due to its proprietary nature. The material in this section discussed channel bypass capability.

Additionally, it is not possible to disconnect the MMI test cart from an Eagle-21 protection rack and leave a channel in "bypass" (see Section 3.2.5).

The channel bypass feature of the the Eagle-21 system may be used for the following purposes:

1. To allow for an Inoperable Reactor Trip (RT) or Engineered Safety Feature (ESF) Channel to be maintained in an untripped condition for the purpose of troubleshooting

2. To allow for a failed RT or ESF Channel to be Bypassed for the purpose of testing a redundant channel of the same function.

3. To routinely Bypass a RT or ESF Channel for the purpose of testing.

2.3.9 Access to Setpoint Adjustments

The Eagle-21 design has provided for administrative controls and multiple levels of security for access to setpoint and tuning constant adjustments. In order to adjust a setpoint or tuning constant in the Eagle-21 system, an individual must have access to the following:

a,c

The material in this section has been deleted from this non-proprietary document due to its proprietary nature. The material in this section discussed access to setpoint adjustments.

2.3.10 Diagnostics

The Eagle-21 Process Protection equipment provides specific diagnostic information to the user via control board alarms and annunciators, numerous test panel and printed circuit card status LEDs, as well as information available through the Man-Machine-Interface (MMI) test unit. These design features allow for easy recognition, location, replacement, and repair or

adjustment of malfunctioning components or modules.

2.3.11 Failure Modes

The Eagle-21 system has been designed to provide for predetermined failure modes for loss-of-power and diagnostic situations.

Upon loss-of-power, Reactor Trip (RT) and Engineered Safety Features (ESF) actuation comparators will revert to a preferred failure mode as specified by the plant specific protection system functional requirements. All analog outputs to indicators, recorders, control systems, plant computer, etc. will fail off-scale low.

Diagnostic detections are classified into two major groups, Fatal and Non-Fatal.

Fatal diagnostic detections result in the setting of all RT/ESF comparators to the preferred failure mode specified by the plant specific protection system functional requirements. All analog outputs to indicators, recorders, control systems, plant computer, etc. are designed to fail-as-is upon detection of a fatal diagnostic in order to prevent perturbation of plant control systems. Each Eagle-21 protection system rack provides a contact for a "Channel Set Failure" alarm and annunciator (1 per protection set) to alert the operator that a diagnostic detection has occurred.

Non-Fatal diagnostic detections do not result in deliberate action to the RT/ESF comparators or the analog outputs. However, a contact is actuated for a "Trouble" status light and/or "Channel Set Failure" alarm and annunciator to provide for operator detectability.

3.0 TECHNICAL DESCRIPTION

3.1 Eagle-21 Architecture

The Eagle-21 Process Protection System replaces existing analog process protection equipment with multiple microprocessor based subsystems. Typically for each Eagle-21 protection system, three subsystems are used: a Loop Processor Subsystem, a Tester Subsystem, and an Input/Output Subsystem (see Figure 3-1). An overall view of the Eagle-21 architecture is shown on Figure 3-5.

3.1.1 Input/Output (I/O) Subsystem

The input portion of the I/O subsystem (see Figure 3-2) consists of customized Analog Input and Contact Input signal conditioning modules specially designed for use in Process Protection Systems of nuclear generating stations. These modules satisfy all of the unique signal conditioning, signal conversion, isolation, buffering, termination and testability requirements.

The signal conditioning modules are configurable to accept various process inputs including: 10-50 mA current loop (active or passive), 4-20 mA current loop (active or passive), 0-10 vdc, RTD's and field contacts. Both the Analog Input and Contact Input Modules provide signals to the Loop Processor Subsystem. These modules also interface with the Tester Subsystem for test and diagnostic purposes.

The output portion of the I/O subsystem consists of Analog Output, Contact Output and Partial Trip Output modules. These modules receive data from the Loop Processor Subsystem and construct analog, contact, and trip logic output signals. Class 1E isolation is provided for all analog and contact output signals.

To minimize the total installation effort for the Eagle-21 equipment, the existing input/output interfaces are fully emulated. In plants with more advanced control or display equipment, Class 1E isolated data links may be extended directly to those systems, thereby eliminating the analog hardware at both ends.

3.1.2 Loop Processor Subsystem

The Loop Processor Subsystem is that portion of the Eagle-21 system which computes all of the algorithms and comparisons for the protective functions. A Loop Processor Subsystem (see Figure 3-3) consists of a Digital Filter Processor (DFP), Loop Calculation Processor (LCP), Communication Controller, Digital I/O Module, and a Digital to Analog (D/A) converter.

The Digital Filter Processor receives analog signals from Analog Input Modules and performs both Analog to Digital (A/D) conversions and anti-aliasing filtering operations on the input signals. The outputs of the Digital Filter Processor are then passed on the Loop Calculation Processor.

The Loop Calculation Processor performs calculations for protection channel functions, data comparison to setpoint values, and initiation of trip signals based on the data received from the Digital Filter Processor.

The Communication Controller collects information from the Loop Calculation Processor and transmits it to the Tester Subsystem.

The Digital I/O module is utilized to process contact inputs, contact outputs, and trip logic output signals.

The D/A converter module is utilized to convert digital values from the Loop Calculation Processor into analog values which are sent to analog output modules for further processing.

3.1.3 Tester Subsystem

The Tester Subsystem is the focal point of human interaction with the protection system. Together with the Man-Machine-Interface (MMI) test unit it provides the interface which allows test personnel to adjust setpoints and tuning constants, and to perform automatic or manual surveillance tests on the protection system. A Tester Subsystem (see Figure 3-4) consists of a Test Sequence Processor (TSP), Communication Controller, Digital to Analog (D/A) Converter Module, and a Digital I/O Module.

The Test Sequencer Processor (TSP) reads information from the Communication Controller, Digital I/O Module, and the MMI test unit. This information allows the TSP to monitor the overall status of the Eagle-21 protection rack, perform self diagnostics, and initiate surveillance testing. The TSP provides information to the Communication Controller, Digital I/O Module, D/A Converter, and MMI test unit. This information provides for status indication and creation of the Signal Injection and Response (SIR) bus. This bus is distributed through the signal conditioning modules and allows the Tester Subsystem to control and test each module.

The Communication Controller receives information from the Loop Processor Subsystem Communication Controller. This information is then read by the TSP which allows it to monitor the status of the LCP. The Tester Subsystem Communication Controller also provides a serial link to the Test Panel, which allows for information display and printing when connected to the MMI test unit.

The D/A Converter Module receives digital information from the TSP and converts it into high resolution analog signals that are used for test injection via the Signal Injection and Response (SIR) bus.

The Digital I/O module receives information from the TSP and provides signals to a Contact Output Module that provides contacts for field devices.

3.1.3.1 Man-Machine-Interface (MMI)

The Man-Machine-Interface (MMI) test unit is connected to the protection system by inserting a cable/connector assembly into the process rack test panel. The MMI permits the user to perform the following functions:

[ ]a,c

a,c

The material in this section has been deleted from this non-proprietary document due to its proprietary nature. The material in this section discussed Man-Machine-Interface design.

3.2 Eagle-21 Hardware Description

The Eagle-21 Process Protection System is comprised of a number of hardware modules and sub-assemblies which are described in this section.

3.2.1 Analog Input Module

The analog input module provides the interface between process transmitters, RTD's and the Eagle-21 computer hardware. Each analog input module provides the capability to interface with a maximum of four inputs. Analog input modules are capable of interfacing with both 4-20 mA and 10-50 mA current loops, 0 to 10 VDC signals, and four-wire RTD inputs.

The 4-20 mA and 10-50 mA current loops are arranged as two-wire current loops with the transmitter power supplied from the analog input module. Separate current loop power supplies and separate signal conditioning circuitry are provided for each transmitter.

The 0-10 vdc inputs are arranged as two-wire double-ended input signals. Separate signal conditioning circuitry is provided for each input signal.

The RTD inputs are arranged to accept a four-wire input configuration. The RTD excitation current source is supplied from the analog input module. Separate current sources and separate signal conditioning circuits are provided for each RTD input.

Included on the analog input module are provisions for automatic testing and automatic calibration. The automatic testing is accomplished via the Tester Subsystem. The analog input module communicates serially with the Test Sequence Processor (TSP) over the Signal Injection and Response (SIR) Bus. Test commands are transmitted to the input module which allows a selected analog input channel to switch from the field sensor to one of the multiple analog reference signals controlled by the TSP and carried by the SIR Bus. During surveillance test, the test injection signal is varied with respect to amplitude and frequency to verify proper channel operation.

On-line calibration is controlled continuously by the Digital Filter Processor (DFP) to eliminate potential gain and offset drift in the analog hardware of the input module and the analog-to-digital (A/D) converter located on the DFP. During a calibration cycle, the DFP sends a command to the analog input module to switch from the field sensor to either the high or low on-board precision reference. The values that the DFP receives for the calibration references are used by the Loop Calculation Processor (LCP) to calculate a correction factor that is applied to the input signal.

Referring to Figure 3-6, the analog input module provides the following features for each input signal:

a,c

The material in this section has been deleted from this non-proprietary document due to its proprietary nature. The material in this section discussed Analog Input Module design.

a,c

The material in this section has been deleted from this non-proprietary document due to its proprietary nature. The material in this section discussed Analog Input Module design.

3.2.2 Contact Input Module

The contact input module provides the interface between field contact devices and the Eagle-21 computer hardware. Each contact input module is capable of processing either four complementary contact pairs, or eight independent contacts. The output signals from the contact input module are read directly by the Loop Calculation Processor through digital I/O ports.

Referring to Figure 3-7, the contact input module provides the following features for each contact input:

a,c

The material in this section has been deleted from this non-proprietary document due to its proprietary nature. The material in this section discussed Contact Input Module design.

3.2.3 Analog Output Module

The analog output module (Figure 3-8) provides an interface between the Eagle-21 computer hardware and field devices.

a,c

The material in this section has been deleted from this non-proprietary document due to its proprietary nature. The material in this section discussed Analog Output Module design.

3.2.4 Contact Output Module

The contact output module (Figure 3-9) provides the interface between field devices operated by contact logic and the Eagle-21 computer hardware. Each contact output module is capable of providing up to eight complementary contact pairs for output purposes. The Loop Calculation and Test Sequence Processors control the relay status/contact logic through Digital I/O cards connected to the IEEE Std. 796 bus.

The contact output module provides the following features for each output:

a.c

The material in this section has been deleted from this non-proprietary document due to its proprietary nature. The material in this section discussed Contact Output Module design.

3.2.5 Partial Trip Output Module

The partial trip output module (Figure 3-10) provides the interface between the Eagle-21 computer hardware and the trip logic system. Each partial trip module is capable of providing up to four channels of logic outputs. The trip output module converts a signal from the Loop Processor Subsystem Digital Input/Output module into an On/Off voltage used to drive relays in the trip logic system. Additional features of the partial trip output module are:

a,c

The material in this section has been deleted from this non-proprietary document due to its proprietary nature. The material in this section discussed Partial Trip Output Module design.

3.2.6 Microprocessor Card Chassis Modules

3.2.6.1 [ ]a,c

a,c

The material in this section has been deleted from this non-proprietary document due to its proprietary nature. The material in sections 3.2.6.1 through 3.2.6.6 discuss the application of Original Equipment Manufacturer multibus modules within the microprocessor card chassis.

[ ]a,c

3.2.6.2 [ ]a,c

[ ]a,c

[ ]a,c

3.2.6.3 [ ]a,c

[ ]a,c

3.2.6.4 [ ]a,c

[ ]a,c

3.2.6.5 [ ]a,c

[ ]a,c

[a,c]

3.2.6.6 [ ]a,c

[a,c]

3.2.7 Miscellaneous Hardware

3.2.7.1 Microprocessor Card Chassis

[a,c

The material in this section has been deleted from this non-proprietary document due to its proprietary nature. The material in this section discussed the Microprocessor Card Chassis design.]

3.2.7.2 Test Panel

[a,c

The material in this section has been deleted from this non-proprietary document due to its proprietary nature. The material in this section discussed the Test Panel design.]

a,c

3.2.7.3 DC Power Supply Chassis

a,c

The material in this section has been deleted from this non-proprietary document due to its proprietary nature. The material in this section discussed the DC Power Supply Chassis design.

a,c

3.2.7.4 Power Distribution and Filtering

a,c

The material in this section has been deleted from this non-proprietary document due to its proprietary nature. The material in this section discussed the Power Distribution and Filtering design.

3.2.7.5 Termination Frame

The Eagle-21 Termination Frames are modular assemblies which accommodate a single Input/Output printed circuit board. The Termination Frame serves to stiffen the Input/Output board against seismic input and provides terminals for power and signal connections. Each Eagle-21 process protection rack will contain sixteen termination frames, installed one above the other in a structure known as the termination framework.

3.2.7.6 Cabinet Cooling Assembly

a,c

The material in this section has been deleted from this non-proprietary document due to its proprietary nature. The material in this section discussed the Cabinet Cooling Assembly design.

3.3 Software

3.3.1 Software Development

a,c

The material in this section has been deleted from this non-proprietary document due to its proprietary nature. The material in this section discussed Software Development.

a,c

3.3.2 Software Implementation

a,c

The material in this section has been deleted from this non-proprietary document due to its proprietary nature. The material in this section discussed Software Implementation.

4.0 EQUIPMENT QUALIFICATION

4.1 Equipment Qualification Background

In November of 1974, the NRC issued Regulatory Guide 1.89, "Qualification of Class 1E Equipment for Nuclear Power Plants" which endorsed IEEE Std. 323-1974, "IEEE Standard for Qualifying Class 1E Equipment for Nuclear Power Generating Stations" and in March of 1976 the NRC issued Regulatory Guide 1.100, "Seismic Qualification of Electrical Equipment for Nuclear Power Plants" which endorsed IEEE Std. 344-1975, "IEEE Recommended Practices for Seismic Qualification of Class 1E Equipment for Nuclear Power Generating Stations."

Westinghouse recognized that NRC approval of testing methodology and parameters prior to performance of the test is desirable to avoid retesting. Therefore, the initial strategy with the IEEE Std. 323-1974 Qualification Program was to obtain NRC approval prior to implementation of the Qualification program. To accomplish this, in October 1975, Westinghouse issued WCAP-8587, Revision 0, "Methodology for Qualifying Westinghouse WRD-Supplied NSSS Safety-Related Electrical Equipment" and in May 1980, Westinghouse issued WCAP-9714, "Methodology for the Seismic Qualification of Westinghouse WRD Supplied Equipment." Meetings were held with the NRC staff to discuss qualification methods. Based on this interaction and state-of-art methodology, revisions were made to WCAP-8587. As a result, WCAP-8587, Revision 6 and WCAP 9714 were accepted by the NRC staff in a letter from Cecil O. Thomas, Chief, Standardization and Special Projects Branch, Division of Licensing, to E. P. Rahe, Jr., Manager, Nuclear Safety Department, Westinghouse Electric Corporation, dated November 10, 1983.

4.2 Equipment Qualification Program Description

The Equipment Qualification Program demonstrated that the Eagle-21 Process Protection Equipment is capable of performing its designated safety related functions under all specified environmental and seismic conditions. This was accomplished by testing as follows:

4.2.1 Environmental Testing (IEEE Std. 323-1974)

The Eagle-21 equipment was tested under both "normal" and "abnormal" environmental conditions.

a,c

The material in this section has been deleted from this non-proprietary document due to its proprietary nature. The material in this section discussed Environmental Test Parameters.

4.2.2 Seismic Testing (IEEE Std. 344-1987)

The Eagle-21 equipment was subjected to multi-axis, multi-frequency inputs in accordance with Regulatory Guide 1.100. The equipment was subjected to both Operation Basis Earthquake (OBE) and Safe Shutdown Earthquake (SSE) events.

4.3 Equipment Qualification Documentation

The Westinghouse Equipment Qualification test program demonstrated the Eagle-21 Process Protection Equipment is capable of performing its designated safety related functions under all specified environmental and seismic conditions. The equipment qualification test plan and results consists of the following documents:

1. WCAP-8587 "Methodology for Qualifying Westinghouse WRD Supplied NSSS Safety Related Electrical Equipment" which is a Westinghouse Class 3 (Non-Proprietary) report and represents the generic program parent document and describes the basis methodology for the Westinghouse equipment qualification program.

2. WCAP-8587, Supplement 1, EQDP-ESE-69 Rev. 1 "Equipment Qualification Data Package" is a Westinghouse Class 3 (Non-Proprietary) report which presents a summary of the Eagle-21 test parameters, performance specifications, acceptance criteria, and test results.

3. WCAP-8687, Supplements 2-E69A, 2-E69B, and 2-E69C "Equipment Qualification Test Reports," which are Westinghouse Class 2 (Proprietary) reports and present a detailed description of the Eagle-21 test parameters, performance specifications, acceptance criteria, and test results.

5.0 NOISE, FAULT, SURGE WITHSTAND CAPABILITY, AND RADIO FREQUENCY INTERFERENCE TESTS

5.1 Test Description

The Noise, Fault, Surge Withstand Capability, and Radio Frequency Interference (RFI) tests demonstrated that the Eagle-21 process protection equipment is capable of performing its designated safety-related functions when subjected to these specified conditions. This testing was accomplished as follows:

5.1.1 Noise Tests

The Eagle-21 equipment was subjected to five types of noise testing:

a,c

The material in this section has been deleted from this non-proprietary document due to its proprietary nature. The material in this section discussed Noise Test Parameters.

a,c

5.1.2 Fault Tests

The Eagle-21 process Equipment was subjected to following fault voltages:

a,c

The material in this section has been deleted from this non-proprietary document due to its proprietary nature. The material in this section discussed Fault Test Parameters.

5.1.3 Surge Withstand Capability (SWC) Tests (IEEE Std. 472-1974)

The Eagle-21 process equipment was subjected to the following surge signals:

a,c

The material in this section has been deleted from this non-proprietary document due to its proprietary nature. The material in this section discussed Surge Withstand Capability Test Parameters.

5.1.4 Radio Frequency Interference (RFI) Tests

[

]a,c

5.2 Test Documentation

The Eagle-21 Noise, Fault, Surge and Radio Frequency Interference (RFI) tests and results are documented in WCAP-11733 (Proprietary) and WCAP-11896 (Non-Proprietary).

6.0 DESIGN, VERIFICATION ANO VALIDATION PLAN

6.1 Background

Westinghouse introduced the concept of microprocessor based Protection Systems in the early 1970's on the Integrated Protection System (IPS) which was part of the RESAR 414 standard plant design. The software verification program conducted on this prototype is documented in WCAP-9153 "414 Integrated Protection System Prototype Verification Program", and WCAP-9739 "Summary of Westinghouse Integrated Protection System Verification and Validation Program".

Building upon the experience gained in performing software Verification and Validation on the IPS prototype and implementing the "lessons learned" from the Nuclear Regulatory Commission (NRC) audit process, a much improved V&V program was defined for the South Texas Qualified Display Processing System (QDPS). In July 1987 the NRC issued a favorable Safety Evaluation Report (NUREG 0781, Supplement No.4) "Safety Evaluation Report related to the operation of South Texas Project Units 1 and 2."

In September of 1986, Tennessee Valley Authority purchased Eagle-21 Protection System Replacement Hardware for Watts Bar Units 1 and 2. The Eagle-21 V&V process was the same as the one conducted on the South Texas QDPS, modified only to the extent of refining the process based on previous experience and resolution of NRC audit comments.

The NRC final audit of the Eagle-21 equipment for Watts Bar was conducted at the Westinghouse Instrumentation Technology and Training Center in April, 1989. The NRC Safety Evaluation Report on the Eagle 21 system utilization for Watts Bar was transmitted June 13, 1989 from Suzanne

Black, Assistant Director for Projects, TVA Projects Division, Office of Nuclear Reactor Regulation, to Mr. Oliver D. Kingsley Jr., Senior Vice President, Nuclear Power, Tennessee Valley Authority. This report was submitted on Docket Numbers 50-390 and 50-391 for Watts Bar Units 1 and 2 respectively.

In May of 1989, Tennessee Valley Authority purchased Eagle-21 Protection System Replacement Hardware for Sequoyah Units 1 and 2. The Eagle-21 V&V process is the same as the one conducted on the original Watts Bar Eagle-21 application.

The NRC final audit of the Eagle-21 process protection system for Sequoyah was conducted at the Westinghouse Process Control Division in April, 1990. The NRC Safety Evaluation Report on the Eagle-21 system utilization for Sequoyah Unit 1 was transmitted May 16, 1990 from Suzanne Black, Assistant Director for Projects, TVA Projects Division, Office of Nuclear Reactor Regulation, to Mr. Oliver D. Kingsley Jr., Senior Vice President, Nuclear Power, Tennessee Valley Authority. This report was submitted on Docket Number 50-327. On October 31, 1990, the NRC staff issued a similar Safety Evaluation Report on Docket Number 50-328 approving Eagle-21 system utilization for Sequoyah Unit 2.

The Eagle-21 Design, Verification and Validation Plan is attached as "Appendix A" to this report.

6.2 Applicable Standards

The standards which are applicable to the Eagle-21 Design, Verification and Validation Plan are listed below:

A. IEEE Std. 603-1980

"IEEE STANDARD CRITERIA FOR SAFETY SYSTEMS FOR NUCLEAR POWER GENERATING STATIONS"

B. REGULATORY GUIDE 1.153, December, 1985

"CRITERIA FOR POWER, INSTRUMENTATION, AND CONTROL PORTIONS OF SAFETY SYSTEMS"

- Regulatory Guide 1.153 endorses the guidance IEEE Std. 603-1980.

C. ANSI/IEEE-ANS-7-4.3.2 1982

"APPLICATION CRITERIA FOR PROGRAMMABLE DIGITAL COMPUTER SYSTEMS IN SAFETY SYSTEMS OF NUCLEAR POWER GENERATING STATIONS"

- ANSI/IEEE-ANS-7-4.3.2 1982 expands and amplifies the requirements IEEE Std. 603-1980.

D. REGULATORY GUIDE 1.152, November 1985

"CRITERIA FOR PROGRAMMABLE DIGITAL COMPUTER SYSTEM SOFTWARE IN SAFETY-RELATED SYSTEMS IN NUCLEAR PLANTS"

- Regulatory Guide 1.152 endorses the guidance of ANSI/IEEE-ANSI-7-4.3.2.

7.0 COMPLIANCE WITH CRITERIA

7.1 IEEE Std. 279-1971

"Criteria for Protection Systems for Nuclear Power Generating Stations. Some of the information in this report demonstrates the means with which the Eagle-21 Process Protection Equipment satisfies the applicable requirements detailed in Section 4 of the above criteria. References are provided as follows:

Requirement 4.1 "General Functional Requirement"

- This report in general describes the Eagle-21 process equipment and its performance requirements.

Requirement 4.2 "Single Failure Criterion"

- See Section 2.3.1

Requirement 4.4 "Equipment Qualification"

- See Section 4.0

Requirement 4.5 "Channel Integrity"

- See Sections 2.3.3, 4.2 and 5.0

Requirement 4.6 "Channel Independence"

- See Section 2.3.4

Requirement 4.7 "Control and Protection System Interaction"

- See Section 2.3.5

Requirement 4.9 "Capability for Sensor Checks"

- See Section 2.3.6

Requirement 4.10 "Capability for Test and Calibration"

See Sections 2.3.6 and 2.3.7

Requirement 4.11 Channel Bypass or Removal from Operation"

- See Section 2.3.8

Requirement 4.13 "Indication of Bypasses"

- See Section 2.3.8

Requirement 4.14 "Access to Means for Bypasses"

- See Section 2.3.8

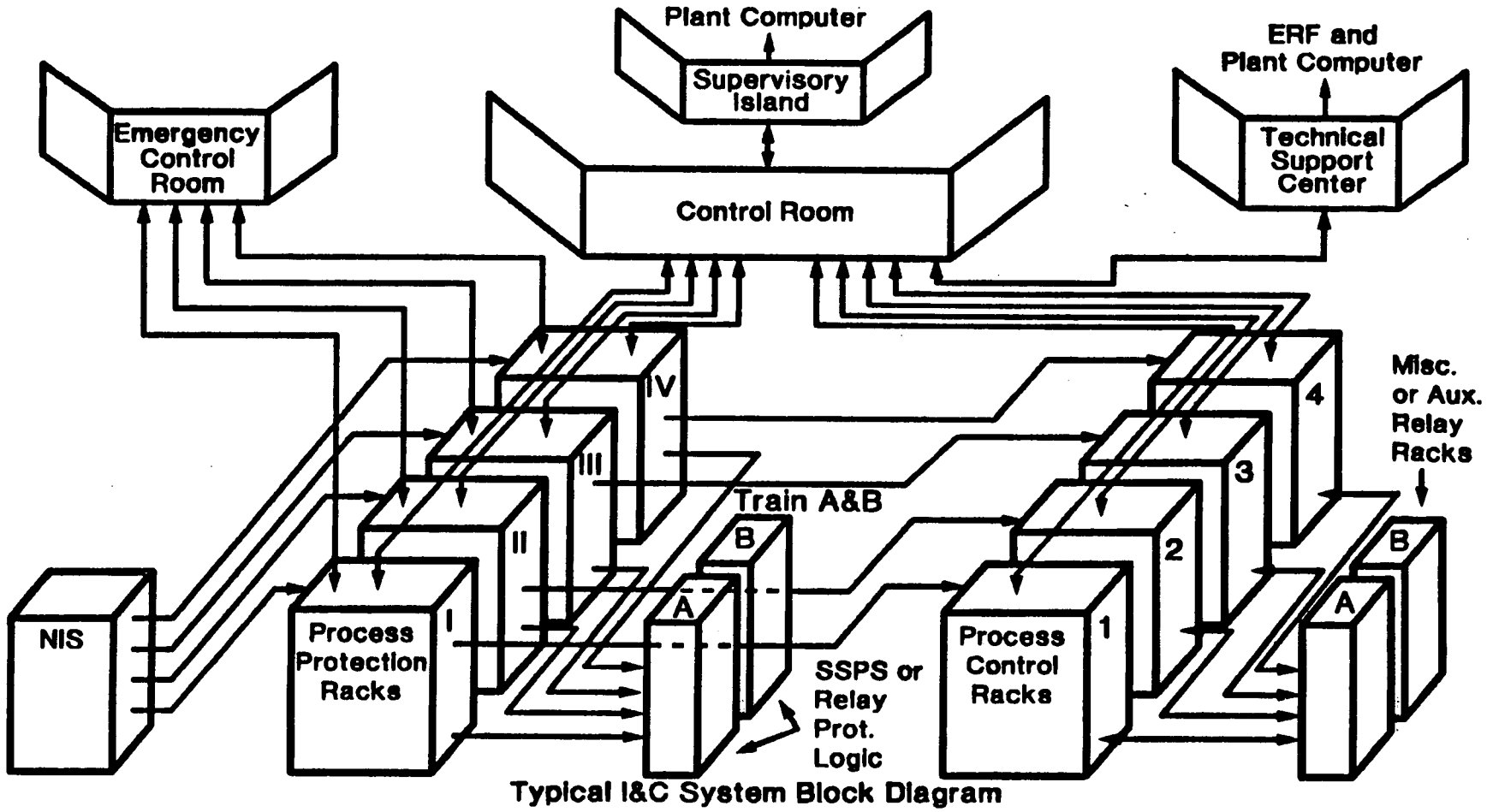
Requirement 4.18 "Access to Set Point Adjustments, Calibration, and Test Points"

- See Sections 2.3.7, 2.3.8, 2.3.9 and 3.2.7.2

Requirement 4.21 "System Repair"

- See Section 2.3.10

EAGLE-21 IMPLEMENTATION



Typical I&C System Block Diagram

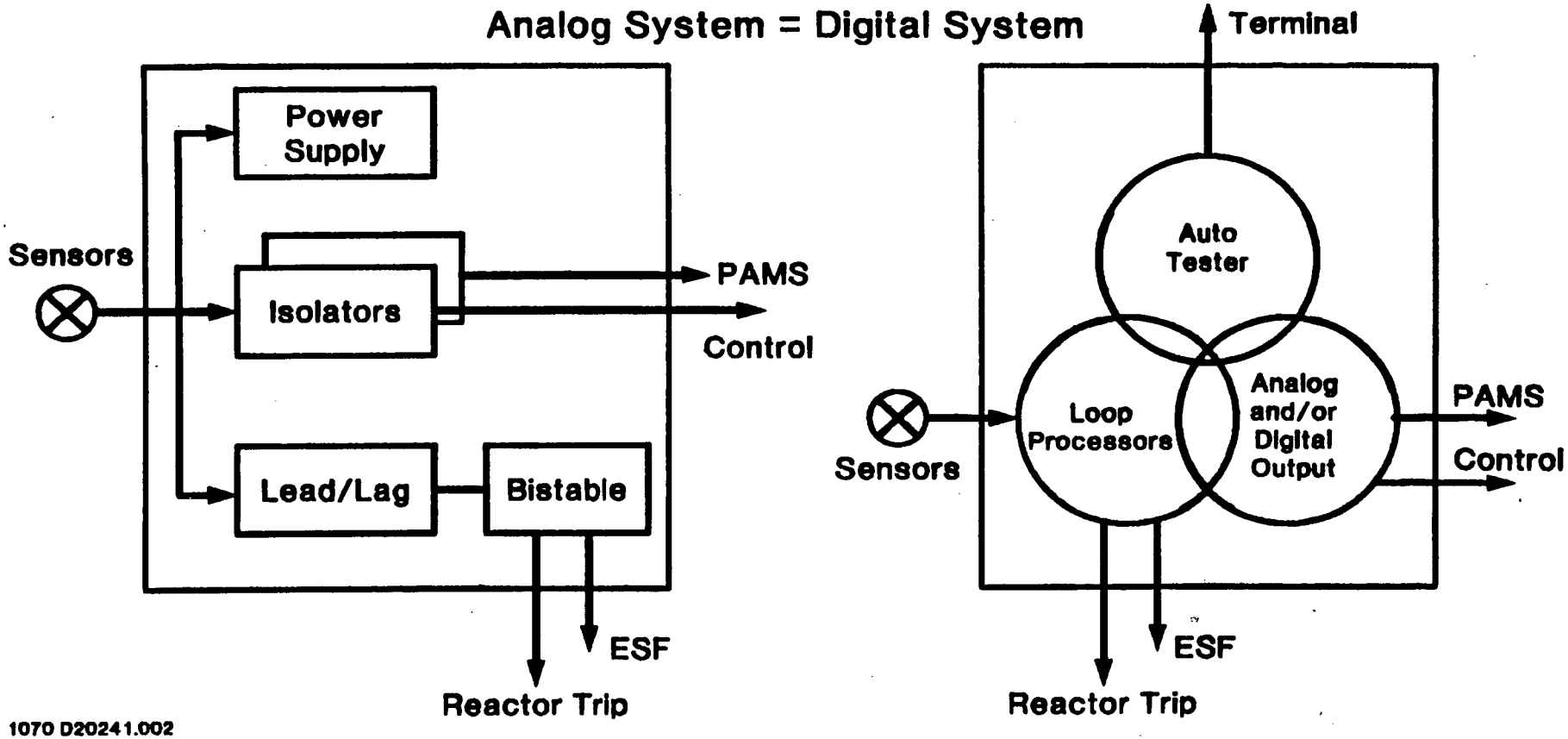
1070 D20241MX.001

FIGURE 1-1

EAGLE-21
DESIGN PHILOSOPHY

- Form, fit and function replacement

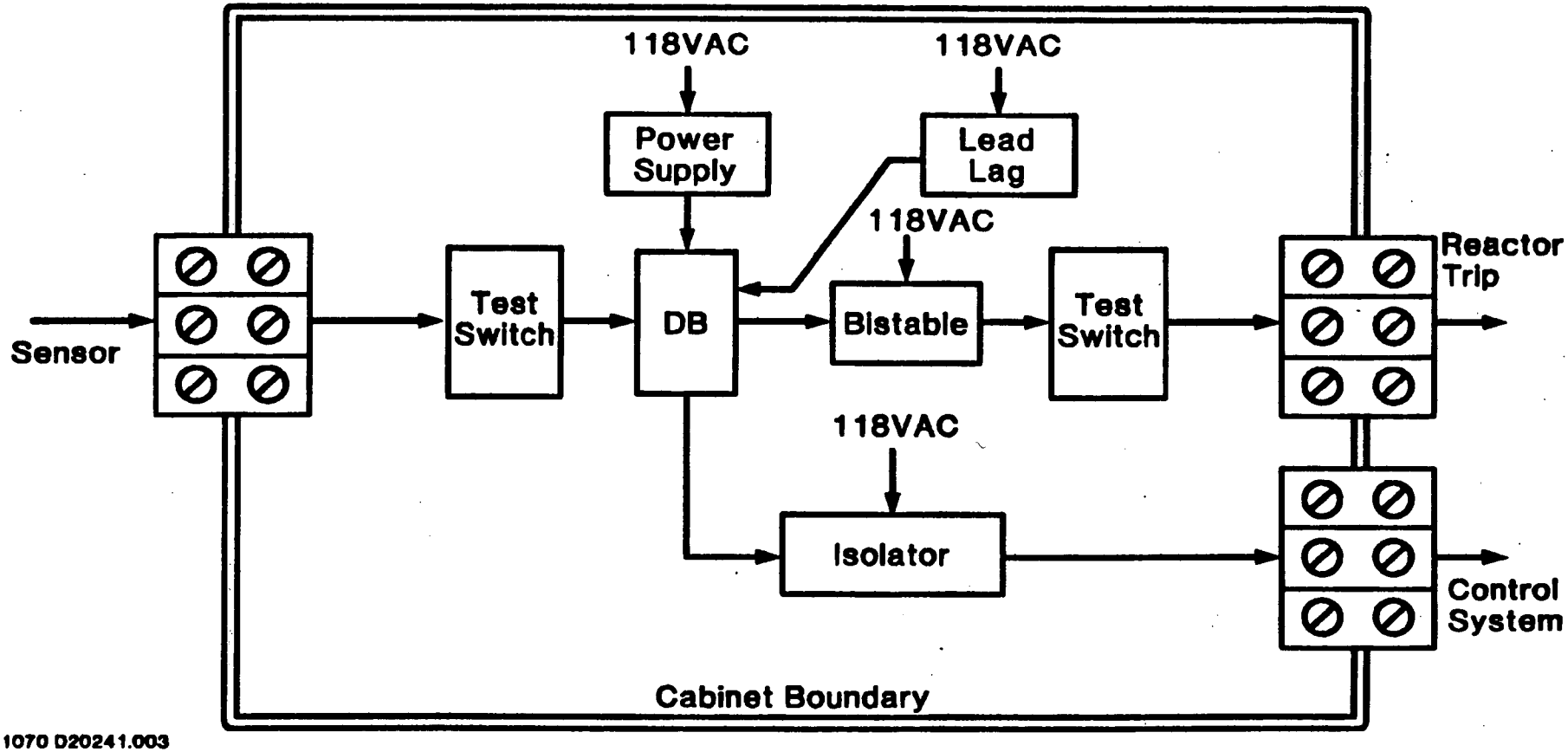
Analog System = Digital System



1070 D20241.002

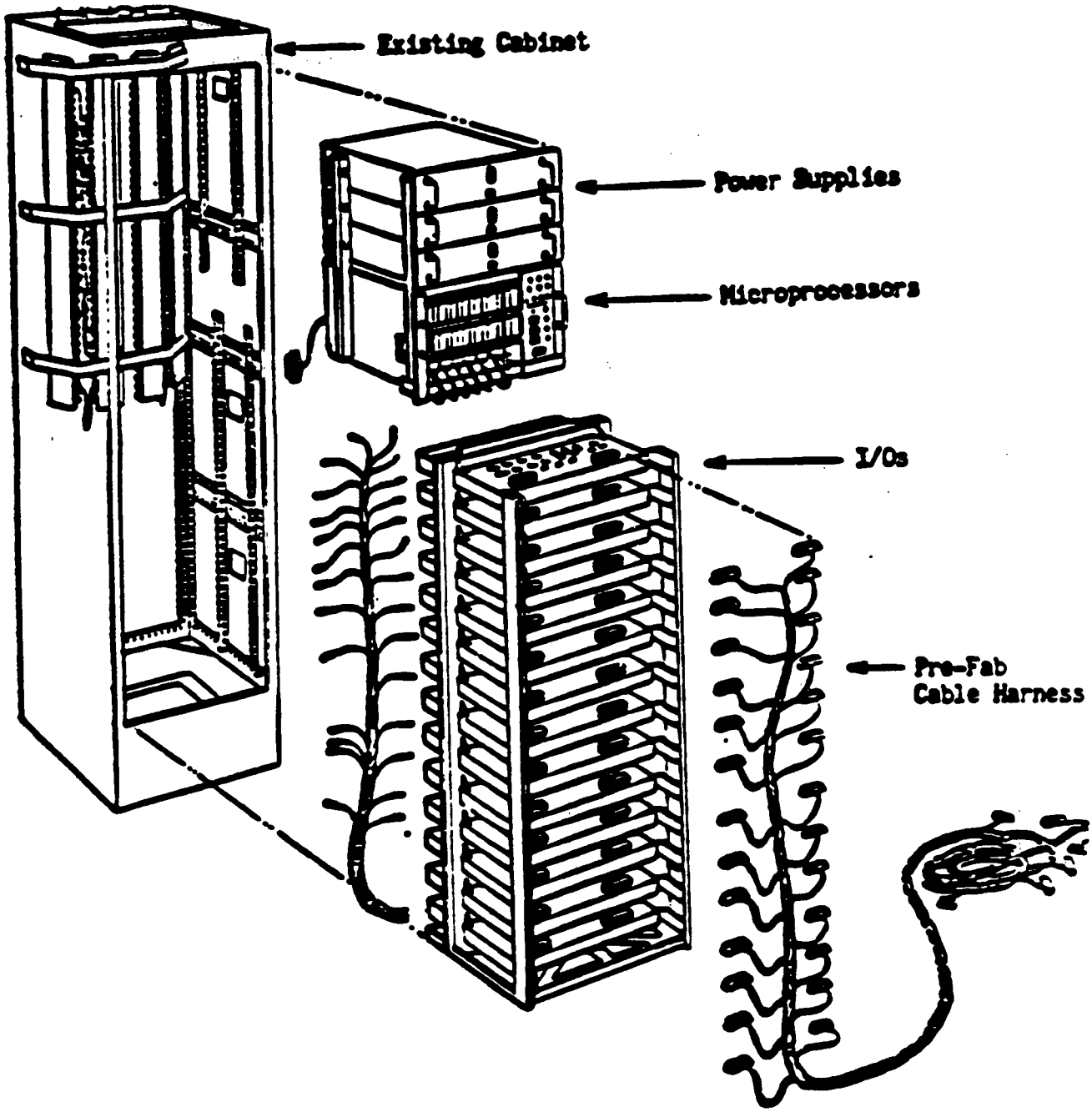
FIGURE 2-1

Typical Analog Process Protection Channel



1070 D20241.003

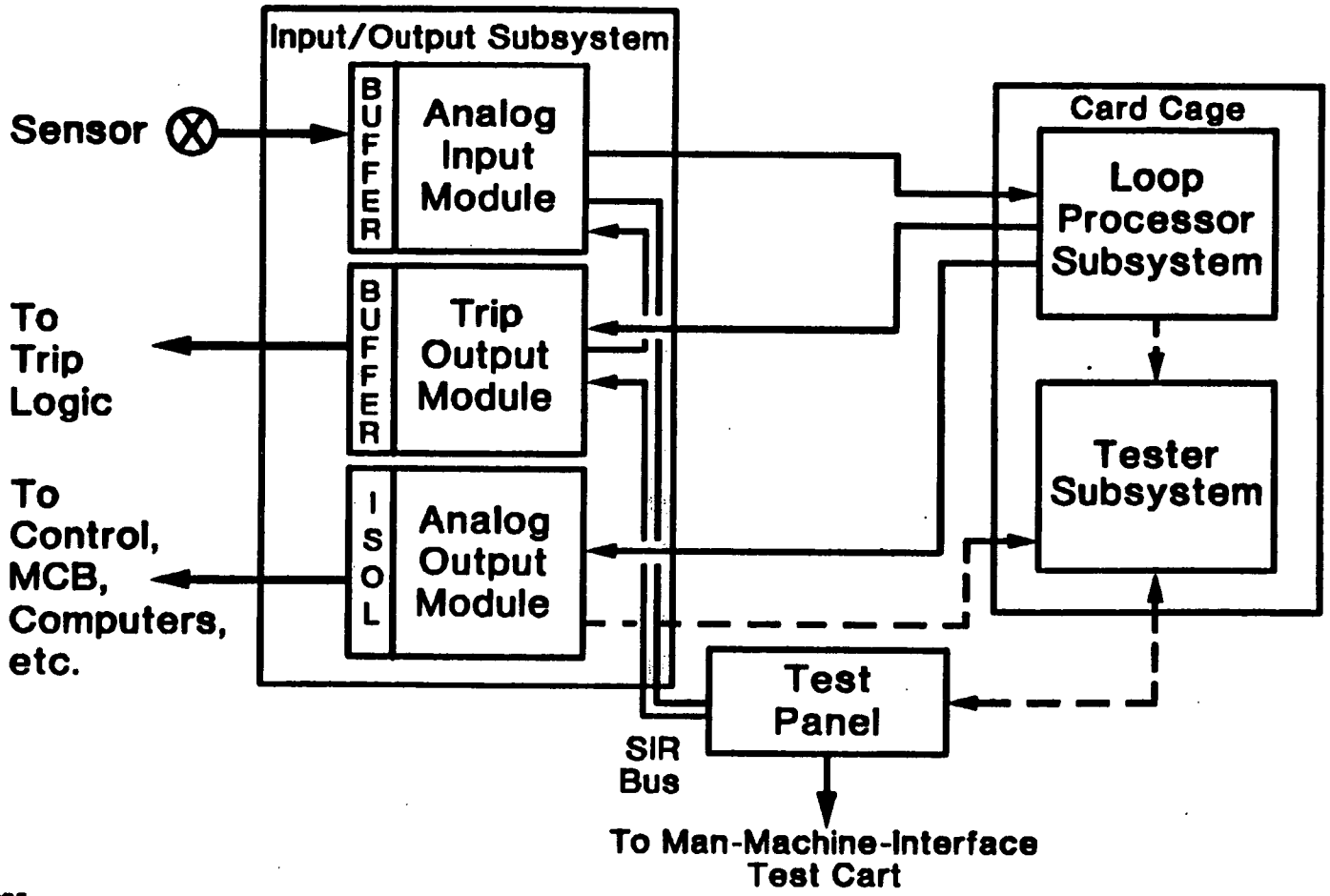
FIGURE 2-2



Existing Cabinet Installation of Eagle 21 Equipment

FIGURE 2-3

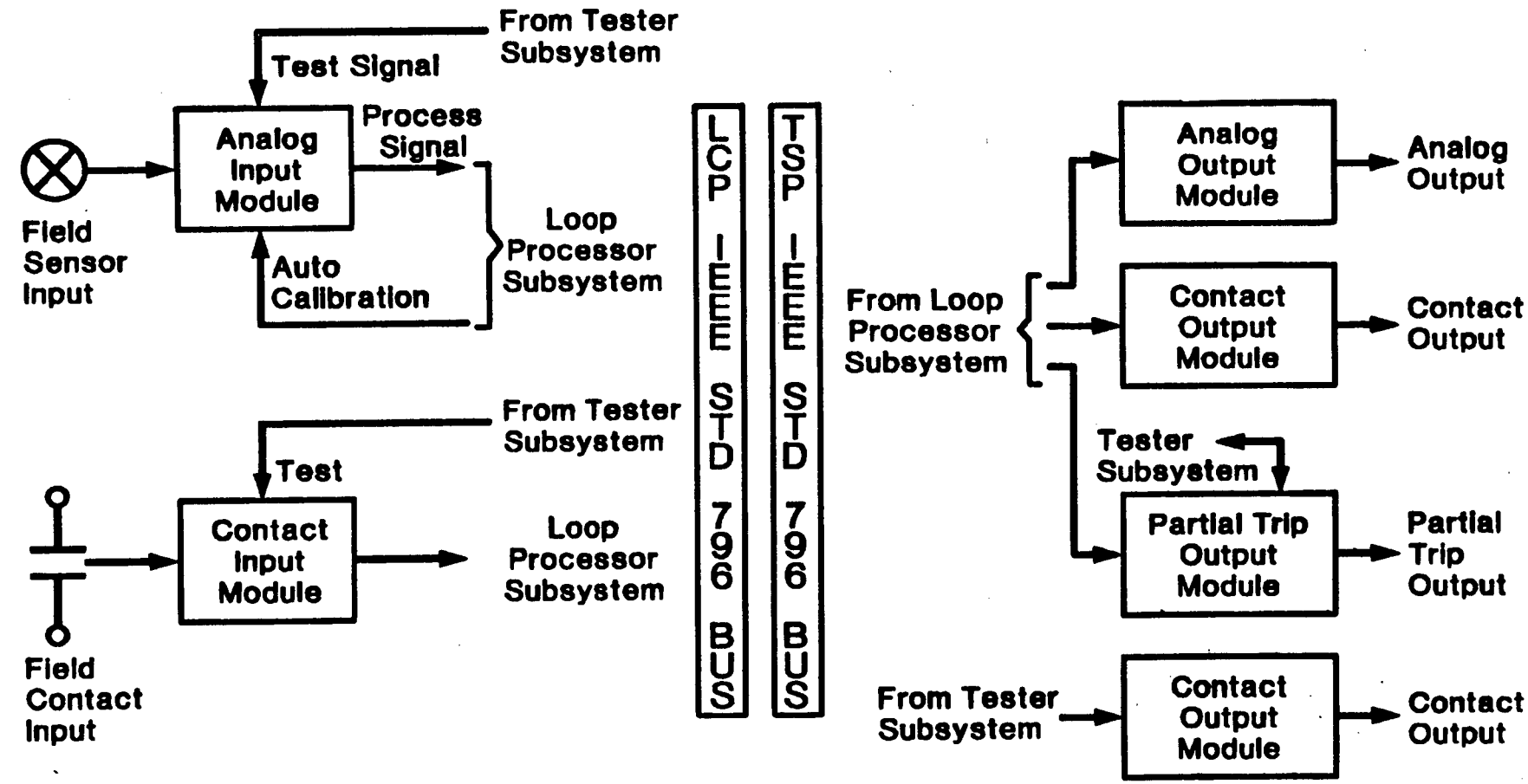
EAGLE-21 SUBSYSTEMS



1070 D20241.005

FIGURE 3-1

EAGLE-21
INPUT/OUTPUT SUBSYSTEM



1070 D20241.007

FIGURE 3-2

EAGLE-21
LOOP PROCESSOR SUBSYSTEM

a,c

FIGURE 3-3

EAGLE-21
TESTER SUBSYSTEM

a,c

FIGURE 3-4

EAGLE-21 ARCHITECTURE

a,c

FIGURE 3-5

ANALOG INPUT FUNCTIONAL CONFIGURATION

a,c

FIGURE 3-6

CONTACT INPUT FUNCTIONAL CONFIGURATION

a,c

FIGURE 3-7

ANALOG OUTPUT FUNCTIONAL CONFIGURATION

a,c

FIGURE 3-8

CONTACT OUTPUT FUNCTIONAL CONFIGURATION

a,c

FIGURE 3-9

PARTIAL TRIP OUTPUT FUNCTIONAL CONFIGURATION

a,c

FIGURE 3-10

SOFTWARE DEVELOPMENT

a,c

FIGURE 3-11

EAGLE 21
LAYERED SOFTWARE APPROACH

a,c

FIGURE 3-12

APPENDIX A

EAGLE 21 REPLACEMENT HARDWARE

DESIGN, VERIFICATION AND VALIDATION PLAN

DESIGN SPECIFICATION SHEET
WESTINGHOUSE FORM 54064G

WESTINGHOUSE ELECTRIC CORPORATION
Nuclear Energy Systems
P.O. Box 355
Pittsburgh, Pennsylvania 15230

DESIGN SPECIFICATION	DATED	REVISION NO.	DATED	ORIGINAL ISSUE	SUPERSEDES PREVIOUS REVISIONS
408A47	11-7-86	3	5-12-89	☐	☒

PROJECT: Generic

EQUIPMENT: EAGLE 21 Replacement Hardware Design, Verification and Validation Plan

SHOP ORDER: 322 / 393

SYSTEM: Process Protection System

ATTACHMENTS

Reviewed by: B E Lang / Z E Erin 11-18-86
Nuclear Safety

REV 1 B E Lang / Z E Erin 01-09-87

REV 2 B E Lang / Z E Erin 02-26-87

REV 3 Z E Erin 5-31-89

☒ NON PROPRIETARY
☐ WESTINGHOUSE PROPRIETARY:

APPROVALS

	ORIGINAL ISSUE	REV. 1	REV. 2	REV. 3	REV. 4	REV. 5	REV. 6
AUTHOR	J.B.Waclo I.J.Tennenbaum			5-12-89			
SHOP ORDER HOLDER	C.E.Corl J.B.Waclo			5-12-89 5/12/89			
MANAGER	D.P.Adomaitis	1/13/87	2/27/87	5/12/89			
PRODUCT ASSURANCE	B.F.Barnett	1/13/87	2/27/87	5/12/89			
PROJECT MANAGER	W.C.Gangloff			5/12/89			

11/18/86

TABLE OF CONTENTS

1.0 Introduction

1.1 Purpose

1.2 System Functions

1.3 System Architecture

2.0 References

3.0 Definitions

4.0 System Development

5.0 System Verification

5.1 Introduction

5.2 Verification Philosophy

5.3 Verification Techniques

5.3.1 Reviews

5.3.1.1 Design Documentation Review

5.3.1.2 Source Code Review

5.3.1.3 Functional Test Review

5.3.2 Software Testing

5.3.2.1 Structural Testing

5.3.2.2 Functional Testing

5.4 Verification Level

5.4.1 Safety Classification

5.4.2 Hierarchical Level of Software Components

5.4.3 Justification of Verification Level

5.4.3.1 Safety Related Software (Level 1)

5.4.3.2 Non-Safety Related Software (Level 2)

5.4.4 Application of the Verification Level and Criteria Utilized for Software Testing for the Eagle-21 Replacement Hardware

5.4.4.1 Application of the Verification Level

5.4.4.2 Criteria Utilized for Software Testing

6.0 System Validation

6.1 Validation Philosophy

6.2 Validation Testing Overview

6.2.1 General Description

6.2.2 Top-Level Functional Requirements

6.2.3 Functional Requirements Testing

6.2.4 Abnormal-Mode Testing

6.2.5 System Prudency Review Testing

7.0 Development, Verification and Validation Team Organization

7.1 Development Team

7.1.1 Chief Programmer

7.1.2 Programmers

7.2 Verification Team

7.2.1 Chief Verifier

7.2.2 Verifiers

7.2.3 Librarian

7.3 Validation Team

7.3.1 Chief Verifier

7.3.2 Functional Requirements Decomposer

7.3.3 Lead Validator

7.3.4 Test Engineer

7.3.5 Librarian

7.3.6 Test Technician

1.0 INTRODUCTION

1.1 Purpose

The purpose of this plan is to provide a description of the design, verification, and validation process and the general organization of activities that are being used in these areas on the Eagle-21 Process Protection System replacement hardware. The material contained herein is modeled after the guidance provided in (a) the 414 Integrated Protection System Prototype Verification Program, which was presented to the NRC in 1977 as part of the Westinghouse RESAR 414 system, (b) ANSI/IEEE-ANS-7-4.3.2-1982 and (c) Regulatory Guide 1.152, and (d) the Design, Verification, and Validation Plan implemented for the South Texas Qualified Display Processing System (QDPS).

1.2 System Functions

The Eagle-21 Process Protection System replacement hardware performs the following major functions:

1. Reactor Trip Protection (Channel Trip to Voting Logic)

2. Engineered Safeguard Features (ESF) Actuations.

3. Isolated Outputs to Control Systems, Control Panels, and Plant Computers.

4. Isolated Outputs to information displays for Post Accident Monitoring (PAM) indication.

5. Automatic Surveillance Testing to verify channel performance.

1.3 System Architecture

The Eagle-21 System Architecture is shown in Figure 1. The basic subsystems are:

1. Loop Processor Subsystem

   The Loop Processor Subsystem receives a subset of the process signals, performs one or more of the protection algorithms, and drives the appropriate channel trip (or partial engineered safeguards actuation) signals. It also drives the required isolated outputs.

2. Tester Subsystem

   The Tester Subsystem serves as the focal point of the human interaction with the channel set. It provides a user-friendly interface that permits test personnel to configure (adjust setpoints and tuning constants), test, and maintain the system.

3. Input/Output (I/O)

The microprocessor based system interfaces with the field signals through various input/output (I/O) modules. These modules accommodate the plant signals and test inputs from the Tester Subsystem, which periodically monitors the integrity of the Loop Processor Subsystem.

2.0 REFERENCES

The following is a list of relevant industrial standards which were considered in the development of this plan:

1. ANSI/IEEE-ANS-7-4.3.2.-1982, "Application Criteria for Programmable Digital Computer Systems in Safety Systems of Nuclear Power Generating Stations"

2. IEEE Std. 279-1971, "Criteria for Protection Systems for Nuclear Power Generating Stations"

3. IEEE Std. 603-1980, "Criteria for Safety Systems for Nuclear Power Generating Stations"

4. WCAP 9153, "414 Integrated Protection System Prototype Verification Program," Westinghouse Electric Corp., August 1977.

5. WCAP 9740, "Summary of the Westinghouse Integrated Protection System Verification and Validation Program," Westinghouse Electric Corp., September 1984.

6. Regulatory Guide 1.97, Rev. 2, "Instrumentation for Light-Water-Cooled Nuclear Power Plants to Assess Plant and Environs Conditions During and Following an Accident," December 1980

7. ANSI/ASME NQA-1-1983, "Quality Assurance Program Requirements for Nuclear Power Plants"

8. IEEE Std 729-1983, "Standard Glossary of Software Engineering Terminology"

9. IEEE Std 730-1981, "Standard for Software Quality Assurance Plans"

10. IEEE Std 828-1983, "Standard for Software Configuration Management Plans"

11. IEEE Std 829-1983, "Standard for Software Test Documentation"

12. IEEE Std 830-1984, "Guide to Software Requirements Specifications"

13. NBS Special Publication 500-75 (February 1981), "Validation, Verification and Testing of Computer Software"

14. NBS Special Publication 500-93 (September 1982), "Software Validation, Verification, Testing Technique and Tool Reference Guide"

15. NBS Special Publication 500-98 (November 1982), "Planning for Software Validation, Verification and Testing"

16. IEC SC 45A/WG-A3 (January 1984), "Draft: Software for computer in the Safety System of Nuclear Power Stations"

17. Regulatory Guide 1.152, "Criteria for Programmable Digital Computer System Software in Safety-Related Systems of Nuclear Power Plants"

18. Regulatory Guide 1.153, "Criteria for Power, Instrumentation, and Control Portions of Safety Systems"

19. Design, Verification and Validation Plan for the South Texas Project - Qualified Display Processing System. Design Specification Number 955842, Revision 3, July 1985.

3.0 DEFINITIONS

The definitions in this section establish the meaning of words in the context of their use in this plan.

COMPUTER SOFTWARE BASELINE - The computer program, computer data and computer program documentation which comprises the complete representation of the computer software system at a specific stage of its development.

DESIGN REVIEW - A meeting or similar communication process in which the requirements, design, code, or other products of a development project are presented to a selected individual or group of personnel for critique.

FUNCTIONAL TESTING (FT) - Exercise of the functional properties of the program to the design requirements.

FUNCTIONAL TEST REVIEW (FTR) - A review which is performed on the documented functional tests that were run by the programmer on his code.

INSPECTION - An evaluation technique in which software requirements, design, code, or other products are examined by a person or group other than the designer to detect faults, differences between development standards, and other problems.

INTEGRATION TESTS - Tests performed during the hardware-software integration process prior to microprocessor system validation to verify compatibility of the software and the microprocessor system hardware.

MODULE (M) - Refers to a significant partial functional capability of a subprogram and consists of more than one unit. Modules are usually stand-alone procedures or routines which may call other lower level modules or units.

PEER REVIEW - An evaluation technique in which software requirements, design, code, or other products are examined by persons whose rank, responsibility, experience, and skill are comparable to that of the designer.

PROGRAM - Totality of software in a system or one independent part of software of a distributed system implemented by a particular CPU.

SOFTWARE DESIGN SPECIFICATION (SDS) - A document which represents the designer's definition of the way the software is designed and implemented to accomplish the functional requirements, specifying the expected performance. An SDS can be for a system, subsystem, module, or unit.

SOFTWARE DEVELOPMENT PERSONNEL - A team of individuals or an individual assigned to design, develop and document software.

SOFTWARE TEST SPECIFICATION (STS) - A document detailing the tests to be performed, test environment, acceptance criteria and the test methodology. An Approved SDS document forms the basis for the STS.

SOURCE CODE REVIEW (SCR) - A review which is performed on the source code.

SUBPROGRAM (SP) - Refers to a major functional subset of a program and is made up of one or more modules. A subprogram is typically represented by the software executed by a single processor.

STRUCTURAL TESTING (ST) - Comprehensive exercise of the software program code and its component logic structures.

UNIT (U) - The smallest component in the system software architecture, consisting of a sequence of program statements that in aggregate perform an identifiable service.

VALIDATION - The test and evaluation of the integrated computer system to ensure compliance with the functional, performance and interface requirements

VERIFICATION - The process of determining whether or not the product of each phase of the digital computer system development process fulfills all the requirements imposed by the previous phase.

VERIFIER(S) - An individual or group of individuals assigned to review source code, generate test plans, perform tests, and document the test results for a microprocessor system. If the activity is extensive, a chief verifier will be appointed to guide and lead the Verification and Validation personnel.

VERIFICATION TEST REPORT (VTR) - A document containing the test results. In conjunction with the Software Test Specification it contains enough information to enable an independent party to repeat the test and understand it.

4.0 SYSTEM DEVELOPMENT

The development of the Eagle 21 System, as shown in Figure 2, involves three stages:

1. Definition
2. Design
3. Implementation and Test

A brief description of each stage is given below:

1) The definition stage is characterized by the statement of the objective to be achieved, the construction of an initial project plan, and a high-level definition of the system. During this stage, the overall functional requirements of the system are identified. Within Westinghouse, these requirements are brought together in a System Design Requirements document.

2) The design stage is characterized by the decomposition of these System Design Requirements into System Design Specifications and Hardware and Software Design Specifications of sufficient detail to enable the implementation of the system. The Software Design Specifications for the system are then further decomposed into subsystem, module and unit specifications.

3) The implementation and test stage is characterized by the actual construction of the hardware, coding of the various software entities, and testing. The software development team is responsible for the writing, assembling, testing, and documenting the computer code. As the software entities are completed, beginning at the unit level, they are formally turned over to the verifiers for final independent review and/or testing as specified in Section 5.0.

   Software development can be viewed as a sequence of well-defined steps similar to system development. The System Design Specification is used to generate Software Design Specifications which in turn are used to develop high level language programs. These programs are converted by a compiler into assembly language, then by the assembler into machine code. The linker combines groups of assembled code with the library to produce relocatable object code for input to the loader. The loader generates the absolute code which is then burned into read only memory (ROM).

   The use of a high level language allows the designer to express his ideas in a form that is more natural to him. The computer adjusts to his language and not he to the language of the computer. Software written in a high level language is more readily reviewed by an independent party who may not be familiar with the computer assembly language instruction set. Some features of the high level language aid the development of reliable software. For example, block structuring helps identify and reduce the number of possible execution paths.

   As part of testing, the various hardware components and software entities are assembled in a stepwise manner. Additional testing at each step to ensures that each component performs its required function when integrated with its associated components.

The final activity associated with the system implementation and testing stage is the testing of the system. A system test plan is derived from

the system functional requirements and system design specifications to confirm that the system exhibits a level of functionality and performance which meets or exceeds the stated requirements. This final system test is referred to as the Factory Acceptance Test.

Several design assurance techniques are utilized throughout all stages of the development process to ensure that the hardware and software components meet the required specifications.

Formal design reviews are held within Westinghouse to ensure that the System Design Specifications meet the System Functional Requirements. The design review team consists of a group of knowledgeable multidisciplineary engineers to ensure that all aspects of the design are reviewed.

During the implementation and test stage, acceptance testing and review are conducted by the designers on the hardware components, circuit boards, and subsystems to ensure they exhibit a level of functionality consistent with the Hardware Design Specifications and Software Design Specifications.

The final design assurance technique utilized is the execution of the system Factory Acceptance Test to ensure the system performance meets the system functional requirements and system design specifications.

5.0 SYSTEM VERIFICATION

5.1 Introduction

With the application of programmable digital computer systems in safety systems of nuclear power generating stations, designers are obligated to conduct independent reviews of the software associated with the computer system to ensure the functionality of software to a level consistent with that described in the system requirements.

Section 5.2 provides an overview of the verification philosophy. Section 5.3 describes the verification techniques utilized in performing the verification process. Section 5.4 describes the criteria that the verification personnel use for determining the level of verification that should be applied to each software entity.

5.2 Verification Philosophy

Figure 2 illustrates the integration of the system verification and validation process with the system design process. The verification process may be divided into two distinct phases: verification of design documentation, and verification of software.

As shown on figure 2, independent verification of design documentation is performed during the design stage. For example, independent verification will occur to ensure that the translation from the Functional Requirements to the Software Design Requirements has been performed properly and thoroughly.

Figure 2 illustrates where an independent review and signoff will be conducted during the design process. Verification of the design documentation will be completed prior to the implementation and test phase.

During the implementation and test stage, when the writing, testing, assembling, and documenting associated with each software entity (beginning at the unit level) is completed by the design team, the software entity is formally turned over to the verifier. At this point, an independent review and/or testing of the software entities is performed to verify that the functionality of the software entities meet the applicable Software Design Specifications. After the verifier is satisfied that all requirements are met, the software is configured for use in the final system and subsequent system validation process.

The software verification process begins at the unit software level, i.e.,the simplest building block in the software. After all software units that are utilized in a software module are verified, the verifier proceeds to verify that module. Not only is the software module verified to meet the module Software Design Specification, but the verifier ensures that the appropriate units are utilized in generating the software module.

After all software modules necessary to accomplish a software subprogram are verified to meet the applicable Software Design Specifications, the verifier proceeds to verify that subprogram. As in the case of the software module, the verifier not only verifies that the subprogram meets the applicable Software Design Specifications, but also verifies that the appropriate software modules were utilized in generating the subprogram entity. This verification philosophy ensures that the verifier tests and/or reviews the interface between the software unit, module and subprogram entities.

Depending upon the hardware implementation, the verification process may utilize system hardware in the verification of the software modules and subsystems.

5.3 Verification Techniques

Verification techniques used in software development fall into two basic categories: review and testing.

5.3.1 Reviews

There are three types of reviews used in the verification of software: Design documentation reviews, code reviews and functional test reviews.

5.3.1.1 Design Documentation Review

This activity involves the comparison of a design document for a subsystem, module, or unit to the design document of the component above it to ensure that all of the performance requirements stated in the higher level document are met.

5.3.1.2 Source Code Review

Source code review, as opposed to code testing, is a verification method in which the software program is examined visually. The operation of the software is deduced and compared with the expected operation. In effect, the operation of the software is simulated mentally to confirm that it agrees with the specification.

Source code reviews will be used to verify the transformation from a Design Specification into high level code. High level code is easy to read and understand, and therefore full inspection at that level is feasible.

5.3.1.3 Functional Test Review

A functional test review is a review by the verifier of the documentation associated with the functional tests which were performed by the designer. This review will provide a high degree of assurance that the software performs the functions specified in the design requirements.

5.3.2 Software Testing

Software tests can be divided into two categories: structural and functional.

5.3.2.1 Structural Testing

Structural testing, which attempts to comprehensively exercise (via computer emulation) the software program code and its component logic structures, is usually applied at the unit level. The functionality of the program is verified along with the internal structure utilized within the program to implement the required function.
Structural testing requires that the verifier inspect the code and understand how it functions before selecting the test inputs. The test inputs should be chosen to exercise all the possible control paths within the software component. If this is not possible, the test inputs should be chosen to exercise every statement within the component. For example, if

a trigonometric function is calculated in several different ways, depending on the range of the input argument, then the test inputs include tests for the argument in each of these ranges, as well as on the boundaries between ranges. In particular, they exercise the upper limit, the lower limit, and at least one intermediate value within each range.

5.3.2.2 Functional Testing

In the functional approach to program testing, the internal structure of the program is ignored during the test data selection. Tests are constructed from the functional properties of the program which are specified in the Design Specification. Functional testing is the method most frequently used at the module or subsystem level. Examples of functional testing include random testing and special cases by function.

Random testing is the method of applying a test input sequence chosen at random. The method can be used in the following circumstances: to simulate real time events that are indeed random; to increase the confidence level in the correctness of a very complex module; to test a subsystem or a system where it is not necessary to test all the possible paths; to get a quantitative measure on the accuracy of a numeric calculation; or to get a measure of the average time required by some calculation.

Special cases by function can be deduced from the Design Specification of the module and will determine some test cases. For example, a subroutine for matrix inversion should be tested using almost-singular and ill-conditioned matrices. Subroutines which accept arguments from a specified range should be tested with these arguments at the extreme points of the range. An arithmetic package should be tested with variables which have the largest and smallest mantissa, largest and smallest exponent, all zeroes, and all ones and negative variables.

5.4 Verification Level

The choice of particular verification techniques to be utilized on a system component is a function of the following parameters:

A. The safety classification of the system

B. The hierarchical level of the software component (unit, module or subprogram)

5.4.1 Safety Classification

The safety classification of an item is defined according to IEEE-279-1971 and IEEE Std 603-1980. In general, the safety classification of the system establishes the verification requirements for the system. However, since all the components contained in the system do not necessarily perform equal safety functions, a higher or lower level of verification may be assigned to specific system components depending on the exact functions performed. If a different level of verification is assigned to a component, the interactions between that component and the other components in the system must be carefully considered and reviewed.

5.4.2 Hierarchical Level of Software Components

For software that is organized in a hierarchical structure, the intricacies of the actual code can not be easily grasped at the upper levels. For all but simple systems it is prudent to approach verification in a progressive manner, beginning at the unit level. It is at the unit level that the code can be most easily inspected or comprehensively tested as necessary.

As the software is built up into higher level components during the integration stage, it becomes possible to demonstrate complete processing functions. This process allows the validation of functional performance requirements. Thus, validation testing assumes a functional theme, with the main emphasis on the interaction between subsystems and their interfaces.

5.4.3 Justification of Verification Level

Considering the parameters detailed above, different verification methods are required for different subsystems and software components. Table 1 illustrates the levels of verification. Each level of the table specifies the type of testing or review that will be performed on the software component within that classification. The justification of the verification levels follows.

5.4.3.1 Safety Related Software (Level 1)

The software associated with actuation and/or implementation of reactor trip, engineered safety features, and information displays for manually controlled actions (as defined by IEEE Std. 279-1971 and IEEE Std. 603-1980) must receive the highest level (level 1) of verification identified. As such, all software must be

structurally tested to ensure that all lines indeed meet the intended design specification. Since the plant operators rely upon the automatic actuation of the reactor trips and/or engineered safeguards actuations, as well as information displays for manually controlled actions, the highest level of confidence must be afforded.

5.4.3.2 Non-Safety Related Software (Level 2)

The following criteria will by applied to all software units. If <u>all</u> of the following conditions are met, the software is level 2; level 1 will be used otherwise.

1. <u>FUNCTIONS</u>

   a. Does not generate information used during any operational mode (eg. normal, testing, maintenance) by level 1 software functions. 3

   b. Does not perform tests, the results of which are used by level 1 software functions.

2. <u>CONNECTIONS</u>

   a. There is no direct path to level 1 software functions via a common bus structure.

   b. There is no direct path do hardware I/O used by level 1 software functions.

   c. Data Link transmission to level 1 software functions is prevented by hardware design.

3. <u>ORGANIZATION</u>

   a. The software design does not permit writing to areas of RAM memory used by level 1 software functions.

   b. The software design does not permit inhibiting access to memory locations utilized by level 1 software functions.

   c. Software is not part of, nor can alter, the execution path for level 1 software functions.

NOTE: The above criteria will be <u>re-applied</u> when evaluating the impact of future software modifications.

5.4.4. Application of the Verification Matrix and Criteria Utilized for Software Testing for the Eagle-21 Replacement Hardware

5.4.4.1 Application of the Verification Level

The Eagle-21 Replacement system can be divided into two groups: 1) that which performs Safety Related functions, has impact on Safety Related functions, and which tests Safety Related functions and 2) that which monitors the system and provides Non-Safety Related information to the user.

The first group consists of the following (Reference Figure 1):

1. All of the Loop Processor Subsystem
2. The portion of the Tester Subsystem that runs surveillance tests and therefore, has an impact on the I/O modules
3. That portion of the Tester Subsystem which controls communication to the Loop Processor for parameter update.
4. That portion of the MMI cart which allows the operator to input new parameters and which does the limit checking on those inputs.

This group, which meets the criteria for Section 5.4.3.1, will be verified at level 1 to give the highest degree of confidence to this code.

The second group consists of the following (Reference Figure 1):

1. That portion of the Tester Subsystem which has no direct link to the Loop Processor other than a read-only datalink. This includes the software which updates the test panel lights and outputs analog trend points.

2. All of the MMI software except that listed in 4) above.

This group will be verified at level 2 since it meets the criteria of section 5.4.3.2.

5.4.4.2 Criteria Utilized for Software Testing

This criteria will be applied to level 1 software units. Refer to Table 1.

Based on previous verification experience, the following criteria will be used to identify the testing requirements for non-complex procedures. If <u>all</u> of the following conditions are met, manual structural testing will be performed; computer emulation will be used otherwise.

1. Procedure Uniqueness - The verifier must determine that the particular procedure is not unique in such a way that computer emulation is necessary.

2. Math Operations (+, -, *, /) - The procedure performs math only with RCM based variables or data constants.

3. Logical Operations (True/False) - The procedure uses only standard definitions for True and False; True=1, False=0

4. Logical Operations (Masking) - The procedure uses only logical operations which do not set or clear (mask) status or control bits.

5. Mulitple Paths - The procedure has only one direct software path.

6. Procedure Size - The size of the procedure is less than 20 executable lines. Executable line count does not include procedure declare, procedure end, and comments.

7. Internal Procedures - The procedure does not include internal procedure(s).

6.0 SYSTEM VALIDATION

6.1 Validation Philosophy

Whereas the system verification process verifies the decomposition of the system requirement documents in the definition and design stage and also verifies the functionality of the software entities (unit, module, and subprogram) beginning from the smallest software entity and progressing to the program level, the system validation process is performed to demonstrate the system functionality. By conducting the system validation test, the results demonstrate that the system design meets the system functional requirments. Hence, any inconsistencies that occurred during the system development, in this area, that were not discovered during the various design verification activities discussed in Section 5.0, would indeed be reviewed, identified, and tracked by the verifiers through resolution by the design team.
Following completion of the system validation test, the user can indeed have a high degree of confidence that the system functional requirements are met.

6.2 Validation Testing Overview

During verification, a bottom-up microscopic approach is utilized to thoroughly and individually review and/or test each piece of software within the total system. This requires a significant effort and verifies that each software element operates properly as a stand-alone entity.

Validation complements the verification process and not only insures that the final implemented system satisfies the top-level functional requirements but also that good engineering practice was utilized during the design and implementation of the system. Following are the major phases of validation:

* Top-down functional requirements testing
* Prudency review of the design and its implementation
* Specific Man-Machine Interface (MMI) testing

The macroscopic top-down functional requirements phase of validation testing treats the system as a black box while the prudency review phase requires that the internal structure of the integrated software/hardware system be analyzed in great detail. Due to this dual approach, validation testing provides a level of thoroughness and testing accuracy which is at least equivalent to that which occurs during verification and insures detection of any deficiencies that occurred during the design process but not discovered during verification. Validation testing is performed on the verified software residing within the final target hardware.

6.2.1 General Description

The Validation plan defines a methodology that must be followed to perform a series of top-down functional requirement based reviews and tests which compliment the bottom-up approach utilized during the Verification testing phase.

Four independent types of reviews and/or tests are to be conducted to insure over-all system integrity:

1. Functional Requirements Testing - this insures that the design meets the functional requirements.

2. Abnormal-mode Testing - this insures that the design operates properly under abnormal-mode conditions.

3. System Prudency Review/Testing - this ensures that good design practice was utilized in the design and implementation of critical areas of the system. The items covered within this section require the internals of the system design and implementation to be analyzed in detail.

4. Specific Man-Machine Interface testing - this insures that the operator interface utilized to modify the system's data-base performs properly under normal-mode and abnormal-mode data-entry sequences. This is a critical area requiring special attention due to the impact on the software of the system-level information which can be modified via this interface.

The functional requirements and abnormal-mode testing phases of Validation utilize a black-box systems approach while the System Prudency Review/Testing phase emphasizes the need to understand the internal operations and interactions within the system.

6.2.2 Top Level Functional Requirements

The functional requirements serve as the basis for identifying the tests that must be conducted during the Validation testing phase.

6.2.3 Functional Requirements Testing

The Validation functional requirements testing phase consists of the following steps:

1. Functional requirements decomposition

   The top-level functional requirements must be decomposed into detailed sub-requirements. For each sub-requirement, a test or a series of tests must be identified and performed to insure that the specific sub-requirement is satisified.

Some sub-requirements are fairly general so it is important that the same individual that performs the decomposition also provides the interpretation as to the type of test which must be executed to insure that the sub-requirement is met.

2. Validation test procedure generation

   Once the decomposition has occurred, the specifics of the test(s) must be defined in test procedural form such that it (they) can be conducted during validation testing.

3. Validation test execution (Refer to Section 7.3)

   The detailed tests per the Validation test procedures must be conducted by a Validation Test Technician and the results must be reviewed by the Validation Test Engineer.

Each functional sub-requirment must be uniquely identified. The test procedure generated to test each sub-requirement must be cooresponding ly identified for ease of cross-referencing.

6.2.4 Abnormal-Mode Testing

During this phase of Validation the functional requrements are reviewed to define a series of abnormal conditions underwhich the system must operate properly without results in or causing any inadvertent or detrimental actions.

The Validation abnormal-mode testing phase consists of the following steps:

1. Functional requirements decomposition

   The top-level functional requirements must be reviewed to identify detailed abnormal-mode conditions. The type of test that must be conducted to exercise the system under each abnormal-mode condition must also be defined.

2. Validation test procedure generation

   Once the decomposition has occurred, the specifics of the test(s) must be defined in test procedural form such that it (they) can be conducted during Validation testing.

3. Validation test execution (Refer to Section 7.3)

The detailed tests per the test procedures must be conducted by a Validation Test Technician and the results must be reviewed by the Validation Test Engineer.

Each abnormal-mode condition must be uniquely identified. The test procedure generated to test each sub-requirement must be correspondingly identified for ease of cross-referencing.

6.2.5 System Prudency Review/Testing

During this phase of Validation, the system design and implementation is analyzed and reviewed against the "System Prudency Checklist". The system must be evaluated against this checklist to insure that good engineering practice has been followed.

The System Prudency Checklist addresses the following critical design areas:

* Firmware program storage
* Data-base information storage
* Multiple-processor shared memory architectures
* Data-link oriented system architectures
* Diagnostics
* System time synchronization

Most of these items do not relate directly to a functional requirement or to a series of functional requirements but address the issue of integrated system integrity.

7.0 DEVELOPMENT, VERIFICATION AND VALIDATION ORGANIZATION

During the system design process, two independent functions will be utilized: one for development, and one for verification. The software development personnel receive the System Design Specification, generate the Software Design Specifications, and then designs, develops, tests, and documents the code. The verification personnel receive the released code and its documentation, performs the required reviews and tests as dictated by the Software Verification Level within the Verification Matrix and produces a Verification Test Report (VTR).

This type of organization has several advantages. The use of two independent entities introduces diversity to the process of software generation and reduces the probability of undetected errors. Another benefit is that such a scheme forces the designer to produce sufficient and unambiguous documentation before verification can take place.

Functional independence is essential to achieve these goals. In particular, the two functions will have separate lead engineers. Note that the development personnel submits the code for verification only after the development team has confirmed the code to its satisfaction. Errors discovered (debugging) during the development phase testing are not required to be documented by the verification engineers.

The use of the above procedures does not preclude the possibility that the developer of one module may be the verifier of a different module, as long as that person did not participate in the design or coding of the module being verified.

7.1 Development Activity

The composition of the development team is dependent upon the functions that are required to be performed by the team. Typical team functions include the following:

7.1.1 Chief Programmer

This is the team software leader who is responsible for the software technical matters. The duties of the Chief Programmer include:

a. Software Design Specification

The chief programmer has the responsibility for the development of the Software Design Specifications, which are based on the System Design Specification.

b. Architecture

Global decisions on the structure of the software, decomposition and data base are made by the chief programmer.

c. Coding

Some critical sections of the programs (both in terms of importance and complexity) can be coded by the chief programmer.

d. General

The chief programmer supervises the rest of the team in software technical matters.

7.1.2 Programmers

It is anticipated that there will be more than one programmer, and that at least one programmer will function as a back-up to the chief programmer. The programmers' tasks are to develop the code for modules and/or sub-systems as directed by the Software Design Specifications.

7.2 Verification Activity

The functions of the verification team are as follows:

7.2.1 Chief Verifier

Team leader who is responsible for all technical matters. The duties of the Chief Verifier include:

a. Review System Design Requirements and Specifications received from the development engineer for completeness and unambiguity. (This review may be performed by another qualified individual who is independent of the design area being reviewed.)

b. Review the Software Design Specifications received from the development engineer for completeness and unambiguity.

c. Review verifier's Software Test Specificatins for completeness.

d. Oversee verification of critical sections in the software.

e. Supervise and consult with the verification team.

f. Review Test Reports

7.2.2 Verifiers

a. Perform source code inspections and review Software Design Specifications.

b. Write Software Test Specifications.

c. Run tests on subprograms, modules and units.

d. Write test reports.

7.2.3 Librarian Function

The Librarian performs the following duties in the maintenance of the Verification Software Library:

a. Responsible for the storage and configuration control of the computer software being verified as follows:

(1) Establishes identification of each software element (i.e. unit, module, subprogram) within the Computer Software Baseline (CSB)

(2) Enforces procedures for software and documentation changes during reverification effort

(3) Maintains configuration control of the current CSB

b. Controls the transmittal of computer software to authorized personnel only

c. Ensures no unauthorized changes occur to the CSB

7.3 Validation Function

The functions of the Validators are as follows:

7.3.1 Chief Verifier

a. Coordinate total Validation program

b. Review Validation testing results and write final report

c. Supervise and consult with the validators

7.3.2 Functional Requirements Decomposer (optional/Chief Verifier)

a. Coordinate Validation of a specific area

b. Review functional decomposition for completeness and accuracy (this review may be performed by another qualified individual who is independent of the design area being reviewed)

7.3.3 Lead Validator (optional/Chief Verifier)

a. Coordinate Valdation of a specific area

b. Review functional decomposition for completeness and accuracy (this review may be performed by another qualified individual who is independent of the design area being reviewed)

c. Review and approve test procedure vs functional requirement test specification to insure test procedure is adequate

d. Along with the Librarian, insure that proper verified code is being validated

7.3.4 Validation Test Engineer

a. Write Validation test procedures

b. Oversee Validation testing and review test results

c. Generate Validation Trouble Reports

7.3.5 Librarian

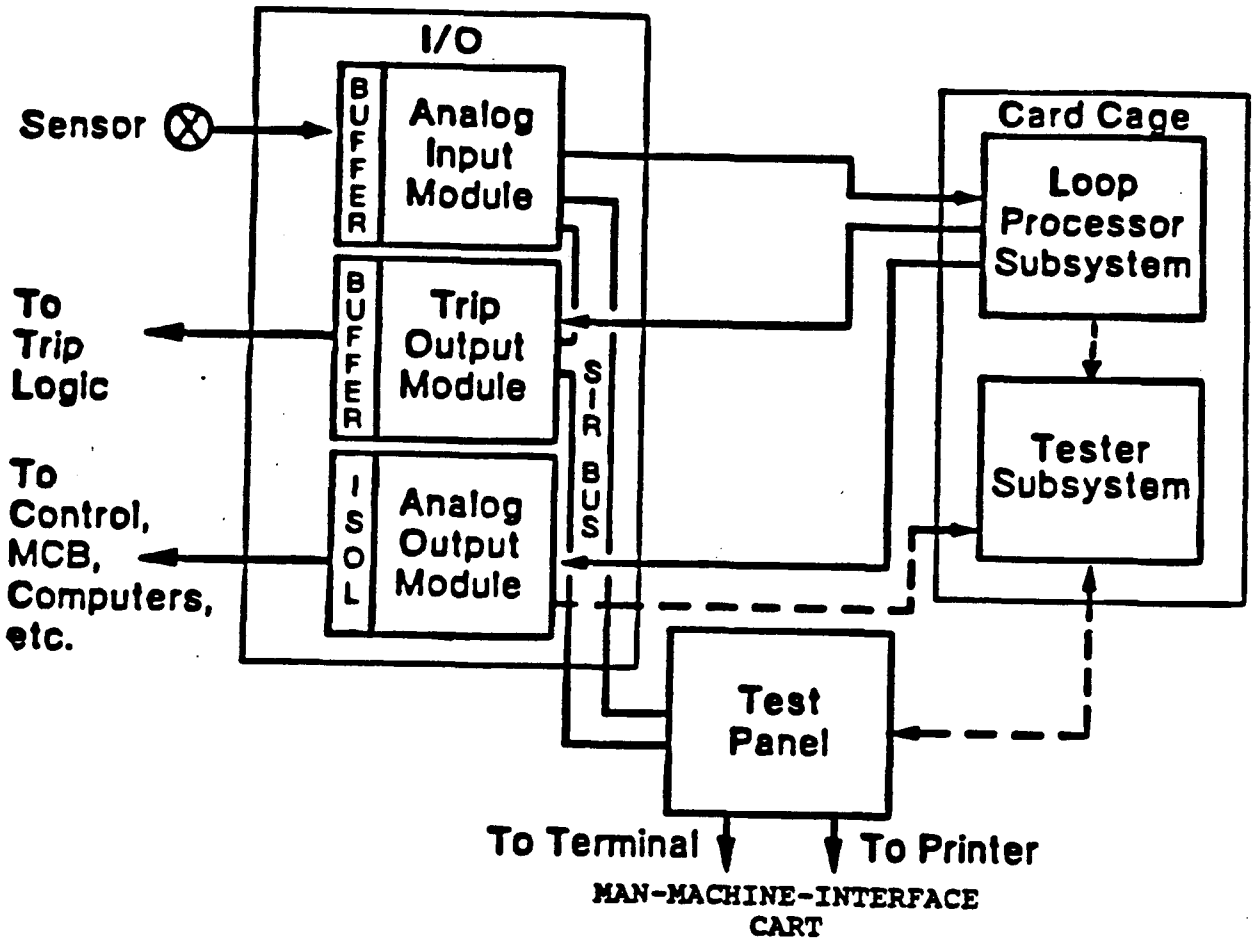
a. Coordinate with the Chief Verifier/Lead Validator(s) and/or Validation test Engineers to insure that proper verified code is being validated.

b. Coordinate dissemination of Validation trouble reports to the appropriate design engineer.

7.3.6 Validation Test Technician

a. Perform Validation tests under direction of the Validation Test Engineer

b. Document test results



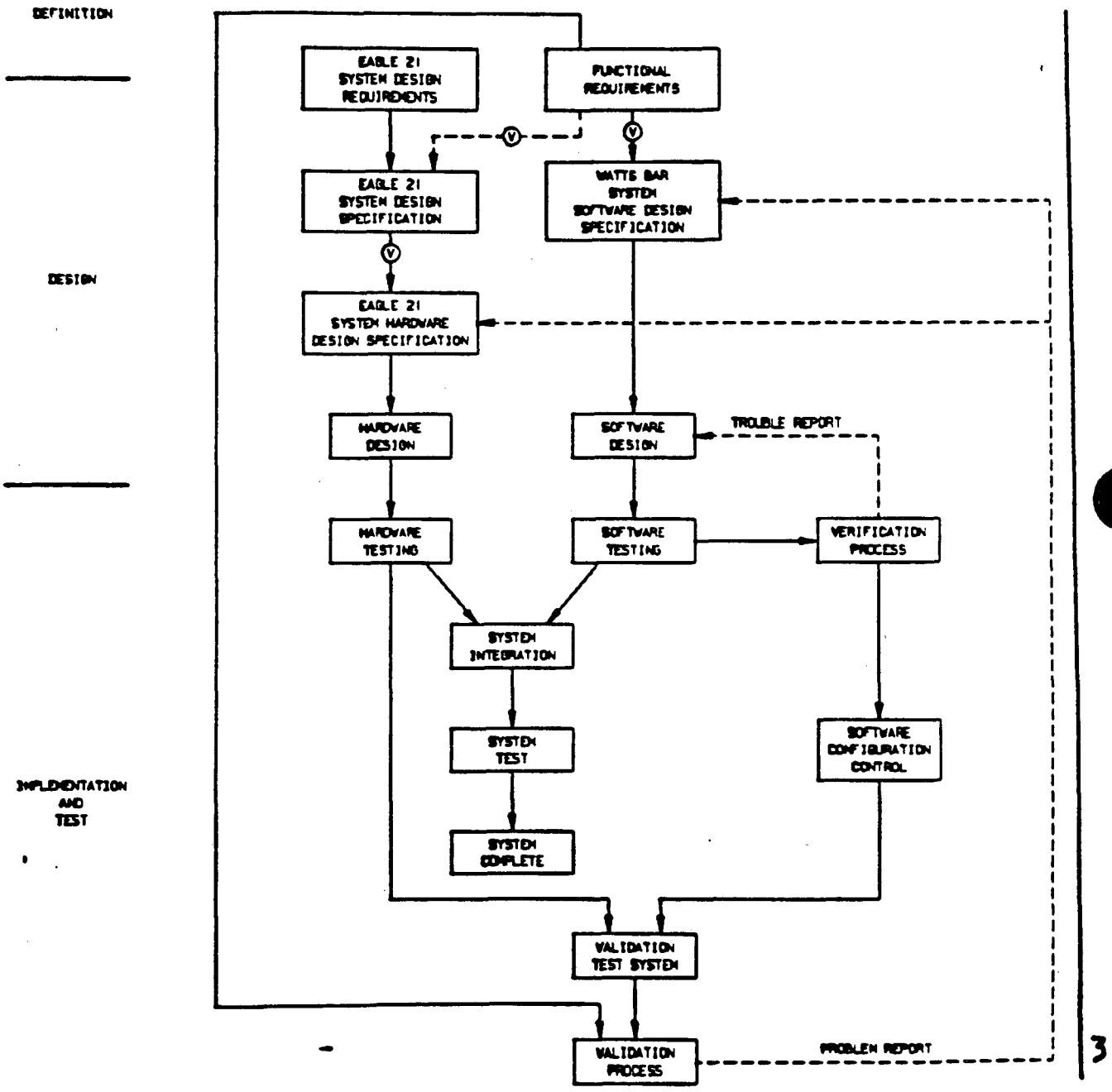
EAGLE - 21

PROCESS PROTECTION SYSTEM

ARCHITECTURE

FIGURE 1

FIGURE 2
DESIGN, VERIFICATION, AND VALIDATION PROCESS



(V) DENOTES INDEPENDENT VERIFICATION REVIEW

SOFTWARE VERIFICATION PROCESS
TABLE 1

	Verification Level	
> FORMAL LIBRARY	Level 1	Level 2
- Code Maintenance.	x	x
- Documentation Maintenance	x	x
- Report (TR & CL) Maintenance	x	x
- Verification Results	x	x
- PROM Files (Hex & Checksum)	x	x
- Impact Analysis Results	x	x
- V&V Tools Documentation	x	x
- V&V Procedures Manual	x	x
> VERIFICATION TESTING		
- Documentation Review	x	x
- Source Code Review	x	x
- Unit Testing		
Structural (5.4.4.2 Criteria)	*	
Functional	x	+
- Trouble Reports	x	x
- Clarification Reports	x	x
- Impact Analysis	x	x

---

x Indicates item will be performed on all software procedures.

* Manual Structural Testing will be performed if <u>all</u> conditions of the 5.4.4.2 Criteria are satisfied; computer emulation will be used otherwise.

+ Review of functional test results performed by designer. Refer to section 5.3.1.3.