HF Controls Corp. • 1624 West Crosby Road Suite 124 • Carrollton, TX 75006 USA • Phone 469.568.6500 • Fax 469.568.6599 • www:hfcontrols.com

July 20th, 2009

U.S. Nuclear Regulatory Commission
Office of Nuclear Reactor Regulation
Division of Policy and Rulemaking
11555 Rockville Pike,
Rockville, Maryland 20852

Subject: DOCUMENTS AS RESPONSES TO CONFERENCE CALL WITH NRC ON 7-9-09 1:00PM – 2:30PM EST WITH REGARD TO TOPICAL REPORT, PP901-000-01, REVISION C, "HFC-6000 SAFETY SYSTEM"

Reference: HFC-6000 Safety System

Docket number: 000-00731

Enclosed are documents which HF Controls (HFC) made available to the Commission based on the conference call on 7-9-09 from 1:00 – 2:30pm EST.

Document	Name	Revision
PP004-000-01	Software Safety Plan	C
QPP 7.3	Commercial Grade Software Evaluation	C
RR901-000-31	Traceability Matrix for HFC-6000 product line	B
RS901-000-01	HFC-6000 Product Line Requirements Specifications	F
TS001-000-01	Component Test Procedure	A
TS901-000-02	SBC06_DPM06 Prototype Test	B
TS901-000-09	DC33 Prototype Test	D
TS901-000-12	DO8J Prototype Test	B
TS901-000-63	HFC-6000 23" Controller Rack Production Test Procedure	B
400484-03	HFC-6000 Digital and Analog Board Functional Test	F
700901-04	HFC-DPM06 Requirements Spec	C
700901-05	HFC-SBC06 Requirements Spec	B
700901-06	HFC-6000, IO Requirements Spec	F

The 700901-04, 700901-05, 700901-06 documents are the requirement specifications which are the attachments to RS901-000-01, "HFC-6000 Product Line Requirements Specification".

During the conference call:

- Question 10: In response to RAI #9 and #10 from the submission of version A of the TR, HFC provided a listing of part numbers and version numbers for the hardware and software modules included in the HFC-6000 platform at the time. Please update the listing and confirm the version/part numbers for the platform as it is now defined.

  Answer: We confirm that the lists in the responses to RAI #9 and #10 are accurate.

- There was a question about HFC-6000 system deterministic operation. Below is an explanation:

  "As a generic nuclear safety system, HFC-6000 is capable of handling multiple executions on application logic with the current tasks arrangement due to all the rest of utility tasks are guaranty to be completed within 10 ms. Since the I/O counts and the size of application logics are fixed, the configurable cycle time to execute all tasks by the Operating System (i.e. 50ms, 100 ms…) and application cycle times (1,…N) should be defined for particular application. In this approach, the execution of TASK 7 (Application Task) shall never be interrupted by the timer routine of the Operating System."

  To gain an in-depth understanding of the deterministic operation of the HFC-6000 system, HFC recommends scheduling a white board discussion during the on-site visit of the NRC's technical reviewer(s).

- 5 examples from the HFC Traceability Matrix for the HFC-6000 product line were asked to be included in this letter. They are listed in the following 2 pages:

System Critical Characteristics (From Customer's Contract)	Requirement Definition	Design Phase Reference	Source Code Reference	Source Code Review	Prototype Test	Production Test
SKN Technical Specification Section 4	**RS901-000-37 SC, SAP, SEP Firmware Requirements Specification**					
	4.1.1 SC Firmware					
4.05.B.4.c4 – Functional Algorithm The analog control circuits shall be able to support the following functional algorithms as a minimum and those required to perform the functions as shown on the Control & Instrumentation Diagrams.	The main controller software includes an interpreter routine that accesses modules for both analog and Boolean functions.	DS001-000-01, ¶2.3, 3.4, 3.4.1 DS001-000-02 DS001-000-03	912038-11 PROM code ($E717_H$ CRC)	SR001-000-013 thru SR001-000-018 SR001-000-020 thru SR001-000-055	TS901-000-02, ¶5.9	TS901-000-63, ¶5.3.1
	4.1.2 SEP Firmware Requirements					
4.05.B.1.b – Redundancy for the process controllers Each redundant controller shall be capable of communicating with the system data communication bus independently. Each of either the primary or backup controller to be connected to the data highway shall have appropriate communication port. The data highway shall have a redundant communication function, in order to be designed to comply with the single-failure-criterion. 4.05.B.4.a3 – Data Communication Redundant high speed data communication buses between control circuit modules shall be provided for data exchange between the control circuits.	If the HFC-SBC06 is configured as part of a non redundant controller, it will begin dynamic operation of its Ethernet ports as sons as the controller becomes SANE. If the HFC-SBC06 is configured as part of a redundant controller, only the CPC processor on the primary card will perform DDB broadcasts to the link.	DS002-000-01, ¶3.1.1, 4, 5.4.1.1, 5.4.1.2	912966-13 PROM code ($5E88_H$ CRC)	SR002-000-007 thru SR002-000-009	TS901-000-02, ¶5.6	TS901-000-63, ¶5.2
	4.1.3 SAP Firmware Requirements					
4.05.B.4.a3 – Data Communication Redundant high speed data communication buses between control circuit modules shall be provided for data exchange between the control circuits.	The firmware will send poll messages to each configured QIO station in the sequence provided by the configuration table. Each poll message will be constructed in accordance with defines supplied by the SAP code (message code, address field, data field, CRC-16)	DS002-000-02, ¶2.2	912949-12 PROM code ($2F55_H$ CRC)	SR002-000-001 thru SR002-000-006	TS901-000-02, ¶5.6, 5.8	TS901-000-63, ¶5.3.1
	700901-06 HFC-6000 IO Requirements Specification					
	2.3.10.2 HFC-DO8J Software					
4.05.B.1.e – Data Acquisition The system shall provide high and low level signal isolation, as well as suitable shielding, filtering and noise reduction circuitry. 4.05.B.1.e.e2 – Contact Output Contact output from the system shall be conditioned by noise suppression networks to avoid spurious operation or damage caused by induced or conducted electrical noises from external transients. Output contacts shall be selected to operate the load circuits reliably based on the specified current, voltage and estimated number of operations. The Supplier shall obtain required contact rating information prior to selection of the output contacts.	During each scan cycle, the software will transfer the validated DO image to all eight output channels.	DS901-000-03, ¶4.3.1	9120677-14 PROM code ($AD95_H$ CRC)	SR001-000-099	TS901-000-12, ¶5.10	400484-03, ¶7.3.2
	2.3.11.2 HFC-DC33 Software					
4.05.B.1.e – Data Acquisition The system shall provide high and low level signal isolation, as well as suitable shielding, filtering and noise reduction circuitry.	The software will start regular input scan cycles after receiving the first valid message from the controller. The software will read	DS901-000-05, ¶4.1, 4.3.1	9120943-10 PROM code ($1C5D_H$ CRC)	SR001-000-097	TS901-000-09, ¶5.6, 5.8	400484-03, ¶7.3.2

System Critical Characteristics (From Customer's Contract)	Requirement Definition	Design Phase Reference	Source Code Reference	Source Code Review	Prototype Test	Production Test
4.05.B.1.e.e1 – Contact Input Contact inputs to the system shall be conditioned by noise suppression networks to avoid spurious operation or damage from transient electrical noises induced externally. All logic inputs shall be biased so as to insure the input goes to logic "0" upon loss of signal. An input used with more than one functional circuit shall be designed such that the circuit failures or maintenance involving one functional circuit do not prevent other functional circuits from operating properly. 4.05.B.4.b3.2 – Logic Circuit Power When logic circuit power is lost, output switches that are closed may open. Plant equipment operated by a continuously closed output contact may stop at that time. The logic circuits shall return to operation without manual actions when logic circuit power is restored. Plant equipment operated by continuously closed output switches may then require restarting. The logic circuit design shall avoid false closing of output contact. It shall also avoid false switching of control mode, etc. and false change of alarm state upon restoration of logic power. Loss of logic circuit power including power to control switch modules and input circuits and logic card removal shall be alarmed. 4.05.B.4.b3.3 Coil Continuity The output circuits controlling circuit breakers, motor starters and solenoid valves shall include circuit continuity monitors for use with the logic circuits and alarms.	status of ac power, and coil continuity along with the status of the 12 DI channels. This data will be reported as the data field for each poll response.					
4.05.B.4.b3.1 Equipment Control Power The system shall monitor the plant equipment control power and provide a logic input to a logic circuit for an alarm upon detection of a power loss. When plant equipment control power is lost, continuously energized equipment will be de-energized. The equipment shall return to operation when control power is restored, unless otherwise indicated on the control logic diagrams. The plant equipment status inputs shall provide status information to logic circuits and control switch modules when control power is lost.	The software read the status of the excitation power branches. The logic will activate an alarm status flag on loss of any branch of excitation voltage.	DS901-000-05, ¶4.1	9120943-10 PROM code ($1C5D_H$ CRC)	SR001-000-097	TS901-000-09, ¶5.4	400484-03, ¶7.3.2

I declare under penalty of perjury that the information in this letter is true and correct to the best of my knowledge.

Sincerely,

Ivan Chow

Ivan Chow
V&V Team, Manager

AFFIDAVIT, STATE OF TEXAS, COUNTY OF DALLAS
Before me, the undersigned authority, personally appeared Ivan Chow, who, being by me duly sworn according to law, deposes and says that he is authorized to execute this Affidavit on behalf of Doosan HF Controls Corporation (HFC) and the averments of fact set forth in this Affidavit are true and correct to the best of his knowledge, information and belief:

Ivan Chow
Ivan Chow

Sworn to and subscribed
Before me this 20th day
of July, 2009

Notary Public

Joyce Druga
7/20/09

NOTARY PUBLIC STATE OF TEXAS
JOYCE DRUGA
My Commission Expires
May 19, 2012

For any discrepancies in the submittals, should they arise, please contact:

Contact Person: Ivan Chow
Email address: ivan.chow@doosan.com
Phone number: 469-568-6566
Mail address: 1624 W. Crosby Rd. Suite 124, Carrollton, TX 75006

Information of the Electronic Document Components:

File Name	File Size	Sensitivity Level
001 Rsp Letter to Conference Call 7-9-09.pdf – Contents: 1. This transmittal letter 2. PP004-000-01 Software Safety Plan Rev. C 3. QPP 7.3 Commercial Grade Software Evaluation, Rev. C	1 MB	Publicly Available
002 Rsp Letter to Conference Call 7-9-09.pdf – Contents: 1. RR901-000-31 Traceability Matrix for HFC-6000 product line Rev. B 2. RS901-000-01 HFC-6000 Product Line Requirements Specifications Rev. F 3. TS001-000-01 Component Test Procedure Rev. A 4. TS901-000-02 SBC06_DPM06 Prototype Test Rev. B 5. TS901-000-09 DC33 Prototype Test Rev. B	8 MB	Non-Publicly Available
003 Rsp Docs to Conference Call 7-9-09.pdf – Contents: 1. TS901-000-12 DO8J Prototype Test Rev. B 2. TS901-000-63 HFC-6000 23 " Controller Rack Production Test Procedure Rev. B 3. 400484-03 HFC-6000 Digital and Analog Board Functional Test Rev. F 4. 700901-04 HFC-DPM06 Requirements Spec. Rev. C 5. 700901-05 HFC-SBC06 Requirements Spec. Rev. B 6. 700901-06 HFC-6000 IO Requirements Spec. Rev. F	14 MB	Non-Publicly Available

The submitting method for all document components is **Electronic**.

HF Controls

SOFTWARE SAFETY PLAN

Document No: PP004-000-01 Rev. C

Effective Date: 7/14/2009

Prepared By: Ivan Chow

Reviewed By: Jonathan Taylor

Approved By: Gregory Rochford

Revision History

Date	Revision	Author	Changes
12/2/08	A	I. Chow	Initial document based on Jon Taylor's software safety plan for SKN 1
3/11/09	B	I. Chow	Corrected programming language used in DSP Removed CMS library reference
7/9/09	C	I. Chow	Revised for the organizational changes

Table of Contents

Number Title Page

1.0 PURPOSE 4

2.0 DEFINITIONS, ACRONYMS, ABBREVIATIONS, AND REFERENCES .. 4

2.1 Definitions 4
2.2 Acronyms and abbvreviations 5
2.3 References 6

3.0 SOFTWARE SAFETY MANAGEMENT 7

3.1 Organization and Responsibilities 7
3.1.1 Project Manager 7
3.1.2 Application Engineering Group 7
3.1.3 Director of Engineering 8
3.1.4 V&V Team 8
3.1.5 QA Manager 8
3.2 Resources 9
3.3 Staff Qualifications 9
3.4 Software Life Cycle 9
3.5 Documentation Requirements 9
3.5.1 Software Project Management 9
3.5.2 Software Configuration Management 10
3.5.3 Software Quality Assurance 10
3.5.4 Software Safety Requirements 11
3.5.5 Software Safety Design 12
3.5.6 Software Development Methodologies 12
3.5.7 Software Test Documentation 13
3.5.8 Software Verification and Validation 14
3.5.9 Software Safety V&V Report (SVVR) 14
3.5.10 Software User Documentation 14
3.5.11 Software Safety Requirements Analysis Report 14
3.5.12 Software Safety Design Analysis Report 15
3.5.13 Software Safety Code Analysis Report 15
3.5.14 Software Safety Test Analysis Report 15
3.5.15 Software Safety Change Analysis Report 15

3.6 Software Safety Program Records ... 16
3.7 Software Configuration Management Activities ... 16
3.8 Software Quality Assurance Activities ... 16
3.8.1 Plan Life Cycle ... 16
3.8.2 Resulting Technical Recommendations ... 16
3.8.3 Reviews and Audits ... 16
3.8.4 Software Safety Program Monitoring ... 16
3.9 Software Verification and Validation Activities ... 17
3.10 Tool Support and Approval ... 17
3.11 Previously Developed or Purchased Software ... 17
3.11.1 Definition of Software Interfaces ... 17
3.11.2 Design Documentation ... 17
3.11.3 Evaluation of Existing Design Capabilities ... 18
3.11.4 Software Component Tests ... 18
3.11.5 Safety-System Tests ... 18
3.11.6 Hazard Analyses ... 18
3.12 Subcontract Management ... 18
3.13 Process Certification ... 18

4.0 SOFTWARE SAFETY ANALYSES ... 19

4.1 Software Safety Analyses Preparation ... 19
4.2 Software Safety Requirements Analysis ... 19
4.3 Software Safety Design Analysis ... 20
4.3.1 System-Level Software ... 20
4.3.2 Application Logic ... 21
4.4 Software Safety Code Analysis ... 22
4.4.1 System-Level Software ... 22
4.4.2 Application Logic ... 22
4.5 Software Safety Test Analysis ... 22
4.6 Software Safety Change Analysis ... 22

5.0 POST DEVELOPMENT ... 23

5.1 Training ... 23
5.2 Deployment ... 23
5.2.1 Installation ... 23
5.2.2 Startup and Transition ... 23
5.2.3 Operations Support ... 24
5.3 Monitoring ... 24
5.4 Maintenance ... 24
5.5 Retirement and Notification ... 25

6.0 PLAN APPROVAL ... 25

1.0 PURPOSE

This document defines the Software Safety Plan (SSP) (or Software Safety Analysis Plan) to be executed during the design and development of a HFC-6000 plant control system. The purpose of this plan is to define additional scrutiny, analyses, and evaluations to be included with the standard V&V tasks for software to be used in safety-related applications. Software that is identified as non-safety related is excluded from the requirements of this plan.

2.0 DEFINITIONS, ACRONYMS, ABBREVIATIONS, AND REFERENCES

2.1 DEFINITIONS

applied project The project which delivers HFC-6000 plant control system.

Project Development Plan The development plan for the applied project.

The following definitions are consistent with IEEE Std 610.12-1990.

accident An unplanned event or series of events that results in death, injury, illness, environmental damage, or damage to or loss of equipment or property.

application software Software created to fulfill specific needs of a user.

operating system A collection of software, firmware, and hardware elements that control the execution of computer programs and provide such services as computer resource allocation, job control, input/output control, and file management in a computer system.

previously developed software Software that has been produced prior to or independent of the project for which the Plan is prepared, including software that is obtained or purchased from outside sources.

risk A measure that combines both the likelihood that a system hazard will cause an accident and the severity of that accident.

safety-critical software Software that falls into one or more of the following categories:

- Software whose inadvertent response to stimuli, failure to respond when required, response out-of-sequence, or response in combination with other responses can result in an accident.
- Software that is intended to mitigate the result of an accident.
- Software that is intended to recover from the result of an accident.

Safety-related software See ***safety-critical software***.

safety system Those systems that are relied upon to remain functional during and following design basis events to ensure (*i*) the integrity of the reactor coolant pressure boundary, (*ii*) the capability to shut down the reactor and maintain it in a safe shutdown condition, or (*iii*) the capability to prevent or mitigate the

consequences of accidents that could result in potential offsite exposures comparable to 10 CFR Part 100 guidelines.

software hazard A software condition that is a prerequisite to an accident.

software module A program unit that is discrete and identifiable with respect to compiling, combining with other units, and loading. (For HFC control systems, the program for a microprocessor typically consists of a single module.)

software safety Freedom from software hazards.

software safety program A systematic approach to reducing software risks.

system hazard A system condition that is a prerequisite to an accident.

system safety Freedom from system hazards.

system software Software created to facilitate the operation and maintenance of a computer system and its associated programs.

2.2 ACRONYMS AND ABBVREVIATIONS

A&E	Application and Engineering
ACE	Abnormal Conditions and Events
CID	Control and Instrumentation
CLD	Control Logic Diagram
CPLD	Complex Programmable Logic Device
CSM	Control Switch Module
DSP	Digital Signal Processor
FMEA	Failure Modes and Effects Analysis
HFC	HF Controls
M/A	Manual/Automatic
MSS	Maintenance Subsystem
OIS	Operator Interface System
OS	Operating System
PC	Personal Computer
PCB	Printed Circuit Board
PDP	Project Development Plan
PDS	Previously Developed Software
PHA	Preliminary Hazard Analysis
PROM	Programmable Read-Only Memory
QA	Quality Assurance
QAPM	QA Program Manual
QPP	Quality Process Procedures (HFC Internal Procedures)
QVDL	Quality Verification Data List
SHA	Software Hazard Analysis
SPMP	System Project Management Plan
SRS	System Requirements Specification
SSP	Software Safety Plan

SVVP	System Verification and Validation Plan
SVVR	System V&V Report
V&V	Verification and Validation
VP	Vice President
WI-ENG	Work Instructions – Engineering (HFC Internal Procedures)

2.3 REFERENCES

The following industry standards, HFC procedures, work instructions are specifically referenced in this plan.

IEEE Std 610.12-1990	IEEE Standard Glossary of Software Engineering Terminology
IEEE Std 1012-2004	IEEE Standard for Software Verification and Validation
IEEE Std 1228-1994	IEEE Standard for Software Safety Plans
IEEE Std 7-4.3.2-2003	IEEE Standard Criteria for Digital Computers in Safety Systems of Nuclear Power Generating Stations
QPP 6.1	Control and Distribution of Documents
QPP 16.3	10CFR Part 21 Reporting
QPP 17.1	Quality Records
WI-ENG-022	Software Verification and Validation
WI-ENG-204	Software/Firmware Coding Procedure
WI-ENG-205	Development of Software/Firmware Test Procedures
WI-ENG-206	CMS Library Software Source Code Control
WI-ENG-207	CPLD Source Code Control and Rebuilding
WI-ENG-830	Source Code Review

3.0 SOFTWARE SAFETY MANAGEMENT

3.1 ORGANIZATION AND RESPONSIBILITIES

Refer to QPP 1.2 "Organizational Responsibilities" for the overall organization and the responsibilities all HFC personnel.

Software is generated both within the Engineering Department and within the Nuclear Project Department, but the nature of the software differs between the two groups:

- The Nuclear Project Department develops the overall system design, produces the hardware packaging design, and implements the specific functional application program code based on the customer's logic design. As such, the application typically includes both safety-related and non-safety-related components.

- The Engineering Department creates firmware code for CPLD devices, DSPs, system-level code for printed circuit board (PCB) assemblies, and application programs for workstation PCs. None of this software/firmware is designed to control the operation of any particular safety system, but reliable operation of system-level code is essential for that of the application code.

The following paragraphs describe the major responsibilities of the top-level managers and their organizations as they relate to software safety. Refer to the project development plan (PDP) for a specific project for a more extensive description of the project team.

3.1.1 Project Manager

The Project Manager for a particular project is responsible for overall coordination and execution of the project. The Project Manager coordinates with the Director of Engineering and the Director of Nuclear Projects to ensure that software safety concerns are adequately addressed for program code included in the system.

3.1.2 Nuclear Project Department

Because HFC will function as a subcontractor to the Primary A&E Contractor for the project, responsibilities of the Nuclear Project Department will be limited to the following tasks:

- Implement the architectural design created by the A&E Contractor using components of the HFC product line.

- Take all necessary steps to establish the basis for qualification of both hardware and software components for all safety-related and safety-impact portions of the control system design.

- Implement the application logic design created by the A&E Contractor using the tools and logic source language designed and developed by HFC. Those portions of the

application logic designated as safety-related by the A&E Contractor will be subject to the code-level analyses defined by this plan. However, final review and approval authority for the HFC application logic rests with the Primary A&E Contractor.

3.1.3 Director of Engineering

The Director of Engineering is a member of the Executive Steering Committee and is responsible for oversight of all internal hardware and software development efforts for the HFC product lines. Of the three major HFC product lines, the HFC-6000 has been specifically developed for use in safety-related applications. Since this product line will be used for the safety-related portions of the project, its components will be subjected to the analyses and reviews mandated by this plan. The Software Safety Plan is created, issued, and implemented under the authority of the Director of Engineering, and he will be responsible for ensuring that any outstanding finings are resolved.

3.1.4 V&V Team

The V&V team reports directly to the President of HFC and is responsible for executing all V&V tasks without having any personal involvement with project-related tasks. Major responsibilities of the V&V team are as follows:

- Create the SVVP and traceability matrix for specified projects.
- Verify that all required inputs for each task are complete.
- Verify that indicated task outputs are generated.
- Validate the task outputs based on the task inputs.
- Identify and flag discrepancies in procedures, execution, or content.
- Generate formal reports covering project execution.

The content of this plan will define task inputs and required outputs for safety-related software that exceed those required for development of non-safety-related software. The V&V team will ensure that these additional inputs and outputs are incorporated into the SVVP at the appropriate points.

3.1.5 QA Manager

The QA Manager is responsible for developing and maintaining the HFC QA Program Manual. The HFC QA program for nuclear projects was designed to meet requirements of NQA-1. The QA manager conducts training for both existing HFC personnel and newly hired staff to provide indoctrination in HFC practices and procedures. In addition, the QA program mandates periodic audits and spot checks for evaluating and verifying compliance with HFC procedures and related industry standards.

Documentation generated in compliance with QA program procedures provides objective evidence for conformity with required standards. During execution of a safety-related project, these documents are retained as part of the project Quality Verification Data List (QVDL). The QVDL for every safety-related project is retained as a permanent record of the development effort for that project.

3.2 RESOURCES

Refer to the project specific PDP for detailed information about resources, resource allocation, and scheduling.

3.3 STAFF QUALIFICATIONS

Refer to the project specific PDP for detailed information about staff qualifications. No formal training is anticipated for the existing key HFC staff. Additional staff (both full time and temporary) may be hired from time to time. All newly hired personnel are required to become familiar with the HFC QA program and those work instructions that directly affect their area of responsibility. In addition, key subject matter experts will train new personnel in the specific areas of their responsibilities (including QA requirements for safety-related projects, V&V task requirements, application logic development, etc.). Detailed records of training and of specific certifications are retained in the personnel file of each HFC employee. The level of responsibility assigned to individual employees is commensurate with both the level of experience and specific training of that employee.

3.4 SOFTWARE LIFE CYCLE

The life cycle model used by HFC for its software development and associated life cycle processes are defined in QPP 3.2. Specific implementations of these work instructions for a project are defined in the project specific PDP and SVVP.

3.5 DOCUMENTATION REQUIREMENTS

The scope of documentation specified by IEEE 1228 for the software safety program is identical to that of the program as a whole. Rather than generate a second set of documentation just for software safety, this topic will be integrated into the appropriate documents to address the additional tasks and scrutiny required for safety-related and safety-impact software.

3.5.1 Software Project Management

Software project management is a subset of the overall project management task and is addressed in the Software Project Management Plan. Three separate types of software development task will take place during execution of the project, and each will have different management requirements:

- Application program code. The Primary A&E Contractor generates the detailed logic design for the safety and non-safety systems associated with the project. HFC is responsible for translating this design logic into functional logic and for converting the functional logic into program code that will operate on the HFC control system platform.

- Modifying existing system software. When functional requirements exceed the capability of existing HFC modules, either the existing designs will require modification or completely new modules will require development. System modification constitutes a design change to PDS and will be treated as such.

- New system software. Creation of a new hardware module typically requires development of new system software; however, addition of new system-level requirements may require generation of new program code for existing hardware modules. Both types of development will be treated as new software.

3.5.2 Software Configuration Management

Methods for software configuration management are addressed in the System Configuration Management Plan. Similar methods of configuration management are used for both the application logic and for the system-level software. HFC does not make any distinction between safety-related and non-safety-related software: configuration management requirements for all program code is the same.

3.5.3 Software Quality Assurance

The Project Quality Plan describes the QA program that will be used for the project. HFC does not have separate QA programs for safety and non-safety software/firmware. This SSP mandates additional tasks, analyses, and evaluations be conducted for the safety-related software. These additions are to be added to the SVVP for the project, and their execution is to be monitored and verified through the V&V program. Reports on the execution of the safety program will then be then included at appropriate points in the V&V task and phase reports.

3.5.4 Software Safety Requirements

HFC control systems typically include five kinds of program code:

- Application code controls functional operation of one or more processes by regulating field devices and monitoring process status. Definition of requirements for this code will be completed by the Primary A&E Contractor. Consequently, this task is beyond the scope of the present development project assigned to HFC.

- System program code installed in PROM or flash memory on individual PCB assemblies. System-level code does have the potential for impacting the execution of safety system operations. Safety considerations will be addressed at the requirements level for all new system-code to be included in a safety system. PDS will be evaluated as an existing design (paragraph 3.11). If PDS requires modification to satisfy functional requirements for the project, then a consideration of safety requirements will constitute part of the change requirements definition.

- HFC uses CPLDs as a compact replacement for large numbers of simple hardware logic components. Although the design is implemented by means of program logic statements, the programmed CPLD is physically mounted on individual PCB assemblies. CPLD code requirements are defined as part of the hardware design effort, but development of CPLD code typically will be covered by the V&V program.

- HFC develops firmware for the DSPs used on some of the I/O cards. Depending on the device, the firmware codes may be written in assembly or C programming language. The development of these codes is covered by the V&V program.

- HFC has developed several software packages that operate on workstation PCs. These application programs operate outside of all safety systems and have no online control function within a nuclear control system. These programs are usually written in C++ programming language. Primary functions of these programs are the following:

  - System status display
  - Alarm annunciation
  - Historical archiving and report generation
  - Troubleshooting, fault isolation, and maintenance
  - Offline application code maintenance and development.

  These programs are designated as non-safety and are not covered by this SSP.

3.5.5 Software Safety Design

The three major categories of software/firmware associated with the project have separate requirements phase characteristics.

- The Primary A&E Contractor is responsible for developing the application program design and will supply that design to HFC is the form of CID and CLD drawings. The scope of HFC's task is to convert those drawings into specific application logic that will run on the HFC control system platform. Since HFC has no design control over the CID/CLD source logic, the design effort is limited to verification that the application logic produced by HFC design fully implements all functions depicted in the CID and CLD.

- System program code essentially serves as the operating system (OS) for the hardware components that make up the control system platform. HFC has developed a set of safety designs that are implemented within every system program as applicable for a particular hardware module:

  - Comprehensive initialization test routines
  - Hardware watchdog triggered by the operating system
  - Software watchdog timers implemented within the main controllers
  - Checksum and CRC validation routines for communication
  - System status monitors and error counters
  - Support for structural redundancy
  - Support for proprietary communication protocols

  The above safety designs are also applied to the DSP firmware codes.

- Because the programmed CPLD functions as a replacement for hardware logic, development of the CPLD program design takes place as part of the hardware design effort. Safety design considerations for CPLD code are addressed as part of the hardware FMEA.

3.5.6 Software Development Methodologies

The overall approach used for software development at HFC is defined by the project SVVP. Standard practices for software development are defined by the following HFC work instructions:

- WI-ENG-204, *Software/Firmware Coding Procedure*
- WI-ENG-205, *Development of Software/Firmware Test Procedures*
- WI-ENG-207, *CPLD Source Code Control and Rebuilding*

These internal procedures cover standard HFC practices for source code generation and configuration control for a particular project. Because the project will develop program code for nuclear safety systems, several different safety analyses of the program software

will be required during project execution. These analyses exceed the standard HFC practices for software development. The following additional activities and methodologies will be addressed by this plan:

- **Preliminary Hazard Analysis (PHA)**. The PHA normally is developed for an entire system or portions of a system to identify hazardous states and sequences that cause that system to enter an identified hazardous state. The PHA for the project was accomplished by the Primary A&E Contractor and constituted the basis for the system logic that will be supplied to HFC. The PHA for the entire system covered by the project and for its component safety system typically is beyond the scope of HFC's contract with the Primary A&E Contractor.

- **Software Hazard Analysis (SHA)**. HFC will conduct a hazard analysis for the software to be used to implement the safety system design, including both the system-level software and the application code. (Possible failure modes for the CPLD will be considered within the hardware FMEA.) Because no generally accepted method for such an analysis currently exists, the methods of hardware FMEA will be adapted for evaluation of software. Three different sources of failure will be considered: 1) The possible existence of latent software design errors, 2) The impact of marginal hardware design errors on software performance, and 3) The impact of random hardware failures on software performance. The result of this analysis will produce a list of software-related Abnormal Conditional and Events (ACEs), which will provide the basis for subsequent analyses through the life cycle phases.

- **Life Cycle Phase Safety Analyses**. This SSP mandates a separate analysis to be conducted at the completion of each life cycle phase for each safety-related software component. The purpose of these analyses is to assess 1) The completeness and quality of the task outputs produced by the current lifecycle phase, 2) How effectively the activities of the current lifecycle phase addressed ACEs identified by the previous phase, and 3) Identify any ACEs that must be addressed by the next lifecycle phase.

3.5.7 Software Test Documentation

HFC develops a comprehensive test program for each application project. This test program encompasses hardware qualification, safety-critical software performance, system-program code performance, and application program operation. In the case of software, specific test procedures and test cases will be developed for each separately identifiable program with specific emphasis on those programs to be used in safety systems. All test procedures will be developed in accordance with HFC internal procedures consistent with requirements of industry standards and evaluated by the HFC V&V program. Test reports resulting from the test activities will become part of the QVDL and will be summarized in the V&V phase reports.

3.5.8 Software Verification and Validation

The standard HFC V&V program is defined by QPP 3.2 and WI-ENG-022. The detailed V&V program to be followed for the project is defined in the project SVVP.

3.5.9 Software Safety V&V Report (SVVR)

Requirements for the SVVR are defined in the project SVVP. Essentially, the SVVR consists of a final compilation and summary of all the V&V phase reports generated during execution of the project.

3.5.10 Software User Documentation

The primary content of the equipment that HFC will supply under the present contract consists of control system hardware, system software to provide the OS for the control system platform, application firmware based on the system design received from the Primary A&E Contractor, workstation PCs, and application programs for those PCs. The user library for this equipment will consist of the following:

- Site Installation and Planning Guide
- System Reference Manual with hardware and software design reference documentation
- Maintenance Manual
- Software Installation Guide
- OIS User's Guide
- MSS User's Guide
- One Step Software User's Guide (offline tool for application development)
- System Parts List

These documents typically are delivered to site following shipment of the equipment from HFC.

3.5.11 Software Safety Requirements Analysis Report

This report will be generated based on the findings of the SHA to be conducted in lieu of a Preliminary Hazard Analysis. (Execution of a PHA exceeds the scope HFC's contract.) The scope of the SHA will cover all software and firmware components initially identified as being safety related. The primary focus of the analysis will be to identify any potential hazards or ACEs that will need to be addressed at the requirements level. If additional software receives the safety-related designation after completion of the SHA, either the analysis will be revised to cover the newly identified software or a separate analysis will be conducted just for these components. The content of this report will be summarized in the requirements phase V&V report and in the final SVVR.

3.5.12 Software Safety Design Analysis Report

This analysis will be conducted as one of the required V&V outputs during the design phase of each software module identified as safety related. The contents of these analyses will be summarized in the design phase of the SVVR.

3.5.13 Software Safety Code Analysis Report

This analysis will be conducted as one of the required V&V outputs during the implementation (coding) phase of each software module identified as safety related. The contents of these analyses will be summarized in the implementation phase of the SVVR.

3.5.14 Software Safety Test Analysis Report

The scope of testing for each safety-related program module will be defined by a written test plan that will identify specific test cases. Typically, the test will include at least one test case for every identified requirement and should include every ACE identified for that module. The test analysis will consist of a quantitative and qualitative evaluation of documented test results. The contents of these analyses will be summarized in the test phase of the SVVR.

3.5.15 Software Safety Change Analysis Report

A safety change analysis refers to the evaluation of anticipated changes to existing software that is designated as safety related. Any of the following conditions will necessitate such an analysis:

- PDS selected for use in a safety application does not meet all functional requirements for its intended use. Rather than generate completely new software, the existing program module will be modified to satisfy the additional requirements.

- Project requirements were modified at some point after formal acceptance of the requirements specification for a particular software module. Such a change will require reevaluation of the original requirements, addition or elimination of individual requirements, and reevaluation of potential ACEs.

- Evaluation of the software design, software code, or software test results indicates that the software requirements as defined are not adequate to fulfill functional requirements of the final system. Consequently, the requirements for the affected software module need to be adjusted, and a corresponding analysis must be conducted to evaluate the impact of that modification.

Regardless of the initial cause of the change, a formal analysis will be conducted prior to the implementation of that change to assess the potential impact on other portions of the system and the scope of regression testing that will be necessary. The resulting analysis report will be incorporated into the project SVVR.

3.6 SOFTWARE SAFETY PROGRAM RECORDS

All plans, procedures, analyses, test records, reports generated in conjunction with the project will be designated as quality records in accordance with the PDP for a specific project. During project execution, this documentation will be tracked and controlled in accordance with requirements of the HFC QAPM to ensure that only the latest version of each document is used for control, development, analysis, and testing. Following completion of the project, all of the project documentation will be placed in long-term storage to ensure that it will be preserved and remain accessible for the design lifetime of the control system. Refer to the project specific PDP for more detailed coverage on project records and record retention.

3.7 SOFTWARE CONFIGURATION MANAGEMENT ACTIVITIES

The software configuration management program is a subset of the configuration requirements of the project as a whole. Refer to the project specific PDP and the System Configuration Management Plan for detailed information on this topic.

3.8 SOFTWARE QUALITY ASSURANCE ACTIVITIES

3.8.1 Plan Life Cycle

Refer to section 6.0 of this plan for the approval process.

3.8.2 Resulting Technical Recommendations

Any technical recommendations regarding to software safety tasks will be recorded as part of the software safety analysis report. If the recommendations include design changes to the software, change requests should be issued to the Engineering Department for appropriate actions.

3.8.3 Reviews and Audits

Reviews and audits will be performed on the software safety analysis records in accordance with the project specific PDP. Based on the review and audit information, design changes can be requested to engineering to ensure software safety concerns, requirements, guidelines, and process certification are addressed.

3.8.4 Software Safety Program Monitoring

As mentioned in section 3.6 of this document, all plans, procedures, analyses, test records, reports generated in conjunction with the project will be designated as quality records in accordance with the project specific PDP. By monitoring the project in which this plan is being executed, the conduct of the software safety program is also being monitored.

3.9 SOFTWARE VERIFICATION AND VALIDATION ACTIVITIES

The specific HFC V&V program with additions mandated by the SSP is defined by the System Verification and Validation Plan.

3.10 TOOL SUPPORT AND APPROVAL

Refer to the project specific PDP for detailed information on project tools and approval requirements.

3.11 PREVIOUSLY DEVELOPED OR PURCHASED SOFTWARE

The safety-related software will include the following broad categories of software components:

- HF-6000 System Software (combination of new development and PDS)
- Safety System Application Software (all new based on CLD and CID received from the Primary A&E Contractor)
- Control Switch Module (CSM) CPLD code
- M/A Station firmware for analog control functions

The following paragraphs describe the process that will be used for accepting previously designed and developed software for safety-critical applications.

3.11.1 Definition of Software Interfaces

Essentially all system-level software for HFC control systems is designed to operate on a single hardware module as the control program for a single microprocessor. Because each hardware assembly has fixed interfaces, identifying the hardware module that will use a particular software module uniquely defines its interfaces with the rest of the system.

3.11.2 Design Documentation

All of the PDS for the HF-6000 product line has existing design and test documentation that has been developed. That existing documentation will be the starting point for evaluation of the control system architecture for the present project. If additional capabilities will be required for the system-level software of any particular module, that will be treated as a design change with the accompanying evaluations and component level testing.

3.11.3 Evaluation of Existing Design Capabilities

A review of existing PDS capabilities will be conducted and documented. This review will be used to identify any areas where the existing design fails to meet specific requirements of the present project. Any such areas will define the scope of additional software development, as previously indicated.

3.11.4 Software Component Tests

Software component tests have been completed for PDS, and reports of test results are on record. Where additional software development is necessary, new component-level tests and regression tests will be developed, approved, and executed to demonstrate satisfactory performance of the integrated hardware/software modules. The results of such tests will be retained to provide objective evidence of component performance.

3.11.5 Safety-System Tests

Tests above the code and module level will be based on the individual functional tests developed for the project. Since system-level code merely provides the operating environment for the application code, the final scope of these tests will cover the fully integrated safety system. Particular emphasis for such tests will rest on the safety-critical aspects of each safety system. The range of test cases will cover normal operating sequences, abnormal states, and failure states.

3.11.6 Hazard Analyses

All PDS will be included in the SHA as well as all of the other analyses and evaluations to be conducted under this plan. Any potential problem or ACE to be identified will be evaluated further to determine if the potential for failure poses an unacceptable risk. Conclusions from all such evaluations will be documented and retained with the project records. Existence of any such risk will provide the basis for a possible modification to the existing software with the requisite functional and regression testing.

3.12 SUBCONTRACT MANAGEMENT

HFC does not plan to use subcontractors to perform additional software development to meet requirements of the project. HFC may, however, employ contract software engineers, who will work in house under direct supervision of HFC managers. All such programming work will be completed under the HFC QA program and scrutinized by the HFC V&V program.

3.13 PROCESS CERTIFICATION

Process certification will be accomplished by means of V&V evaluation of tasks defined by this SSP and by the SVVP. Results of those evaluations will be recorded in V&V reports, a final summary of which will be included as one item in the SVVR. Final certification will be accomplished under the joint authority of the Director of Engineering Development and the V&V Team Manager.

4.0 SOFTWARE SAFETY ANALYSES

4.1 SOFTWARE SAFETY ANALYSES PREPARATION

This area of the SSP normally contains plans for conducting a Preliminary Hazard Analysis (PHA) for one or more safety systems. The Primary A&E Contractor, not HFC, is responsible for the safety system designs and for determining which functions will be performed by software (i.e., control system functions) and which will be performed by hardware (i.e., field equipment external to the control system, some of which may also include software). HFC has been contracted to design and manufacture the control system based on the high-level system design received from the Primary A&E Contractor based on the PHA that they have conducted.

HFC will perform a hazard analysis based on the preliminary control system architectural design developed for the proposal and refined during the post contract award meetings. This analysis will exclude the non-safety portions of the control system from consideration, and it will exclude the specific functions of the application program, which will be designed by the Primary A&E Contractor. (The actual design of the application program is beyond the scope of the present contract with HFC, and that design was not available to HFC at the time that this plan was written.) The hazard analysis conducted by HFC will cover the following:

- The preliminary control system architecture will be assumed. This architecture includes the specific arrangement of hardware modules and communication links. Because each hardware module has an associated software/firmware module and fixed communication interfaces, this architecture also defines the structure of the system-level software.

- The analysis will focus on potential failure states that might be introduced by the system software and evaluate possible mitigating features.

- The analysis will exclude specifically hardware failure modes, but it will consider interaction between hardware and software.

The final output of this analysis will be a set of ACEs to be assessed at the requirements level for the system. Formal review and approval of the report covering this analysis will be done as part of the planning phase of the system life cycle in accordance with the project SVVP.

4.2 SOFTWARE SAFETY REQUIREMENTS ANALYSIS

The first stage of the requirements analysis will be the identification of any new hardware/software development that is evident from the beginning. All new design will require the complete life-cycle development scheme defined by HFC procedures. For those HF-6000 modules that already exist, the following analyses will be performed:

- A detailed evaluation of system requirements versus module capabilities will be executed and documented. The result of these analyses for each existing module will be a definition of new or additional requirements for individual modules.

- The applicability of ACEs identified during the hazard analysis will be evaluated. Mitigation for ACEs may be addressed at the requirements phase for an individual module, at a later stage in the life cycle for an individual module, or by the architecture of the safety system as a whole. The preliminary control system architecture will be assumed at this time.

Because each software module of the HF-6000 is a self-contained entity, each one will require a separate evaluation during the requirements phase. All identified requirements for system-level software will be subject to the following evaluations:

- Assessment of completeness, correctness, accuracy, testability, integrity, reliability, and security

- Timing analysis to evaluate reliability and deterministic behavior

The anticipated scope of new development will be defined and documented for each module in accordance with the SVVP. Since the Primary A&E Contractor will supply the safety system design to HFC, the application software will be excluded from this phase of the development life cycle.

4.3 SOFTWARE SAFETY DESIGN ANALYSIS

4.3.1 System-Level Software

System-level software controls the following functions, or 'design elements' within an HFC control system:

- Input scan to receive data from field hardware
- Level conversion and scaling (for analog or pulse input data)
- Output scan to control digital status or analog output signal levels to field hardware
- Communication between control system modules
- Memory management and timing functions
- Individual processing algorithms that can be called from the application logic
- Execution of algorithms identified in the application logic
- Error detection and status monitoring
- Failure detection and failsafe functions

The specific combination of 'design elements' included in any particular software module is determined by the hardware assembly on which it operates. No system-level software module performs any process control function in itself, but it creates the operating environment for the application program. Accordingly, the system-level code is not safety related unless it operates within a safety system. However, all system software within a safety system will be treated as safety related, because a failure of the

system-level software could prevent the application program from performing its safety function.

The scope of new design needed to meet the performance requirements of the present contract will be determined by the list of requirements generated during the previous development phase. The following types of design level analyses will be conducted as applicable to the particular 'design element:'

- Logic analysis evaluates algorithm requirements to determine if one or more new processing algorithms need to be developed.

- Data analysis evaluates data requirements of the new system to determine if the data categories that currently exist are adequate. If the existing database structure is not adequate for some aspect of control system performance, creation of a new point type or reorganization of data partitions may be necessary.

- Interface analysis evaluates the existing communication interfaces for the safety-critical systems. This analysis will determine if the message protocols will require modification. Areas of concern will include message structure, timing, deterministic operation, error detection, and response time.

- Constraint analysis evaluates the physical constraints on system performance imposed by the design architecture of the control system. This evaluation will assess possible design trade-offs that may be necessary to achieve desired performance characteristics.

Results of these analyses will be documented and recorded as part of the design phase V&V report. The resulting design specification for each software module to be modified or newly created will be reviewed and documented in accordance with the SVVP.

4.3.2 Application Logic

HFC will receive the application design from the Primary A&E Contractor in the form of CLD and CID logic. These drawings depict the control functions to be implemented and include both the software-related and the hardware-related functions of the final installation. (The Primary A&E Contractor will determine the division of design elements and the structure of redundant elements before HFC receives the CLD/CID.) HFC application engineers are responsible for extracting the software-related functions from the CLD and CID and for producing control logic that can operate on an HFC platform. Because the Primary A&E Contractor retains final approval authority over the application logic produced by HFC, internal scrutiny will be limited to reviews of the logic produced versus the source design documents. However, HFC will conduct traceability analyses to ensure that all software portions of CLD/CID logic are converted into HFC logic diagrams. The results of the reviews and traceability analyses will be documented in accordance with the SVVP.

4.4 SOFTWARE SAFETY CODE ANALYSIS

4.4.1 System-Level Software

Most of the system-level programs for control system modules are coded in assembly language. In general, each software module operates in isolation from all other modules except for structured messages received from hardware communication ports. The following analyses and evaluations will be conducted on the final program code as part of the source code review based on an internal HFC procedure:

- Logic analysis. This evaluation will check the source code module for infinite loops, logic errors, inconsistencies, and processing errors

- Interface analysis. Since every program in the system is isolated from every other program by physical installation on a hardware module, all data transfer must pass through a hardware communication interface. This evaluation will verify software support for the communication protocol required to support data transfer.

- Design evaluation. The content of the program code will be evaluated to verify traceability to the requirements and design specifications.

- Programming style. The source code will evaluated for compliance with internal HFC coding standards for usage and adequacy of comments.

Results of these analyses will be documented in Software Safety Analysis Report.

4.4.2 Application Logic

The source code for the application logic is generated directly from the logic diagrams by an HFC software tool called One Step. The source code files for critical application code will be reviewed line-by-line in comparison with the HFC logic diagrams. The purpose of these reviews will be to verify accurate conversion of the graphic representations into program code. The results of these reviews will be documented in accordance with the SVVP.

4.5 SOFTWARE SAFETY TEST ANALYSIS

HFC develops a test program to encompass component, module, integration, and system tests for each project. The primary safety-related tests mandate specific testing of control algorithms essential for safety operation or for safety shutdown under emergency conditions.

4.6 SOFTWARE SAFETY CHANGE ANALYSIS

This aspect of software development begins at the point where the existing design or planned design proves to be inadequate to meet identified system requirements. (Refer to the SVVP on implementation of design changes into software during development.)

When this occurs, the following evaluations will be conducted to determine the scope of change necessary:

- Evaluation of what system requirements cannot be met by the existing design or design architecture.
- Evaluation of what modifications to the existing hardware/software designs will be required.
- Evaluation of how the anticipated changes to the existing hardware/software designs could impact other modules.
- Return to the requirements phase tasks for the modules to be modified, and follow the modifications through the life cycle phases.

5.0 POST DEVELOPMENT

5.1 TRAINING

Training of operators, maintenance personnel, and system engineers in the operation and maintenance of the HFC control system hardware and software is a standard part of the basic contract requirements. Refer to the project specific PDP for detailed information on the training program to be implemented.

5.2 DEPLOYMENT

5.2.1 Installation

The control system components (software and hardware) to be developed under this plan may be part of a new or of a retrofit installation. Initial software installation will be completed at the HFC facility prior to shipment of the equipment cabinets and PCs to site. One or more HFC engineers will be present on site to support and assist with hardware installation of the control system cabinets and initial testing of I/O connections, control algorithms, and communication functions. All on-site equipment installation will be accomplished in accordance with assembly, wiring and logic diagrams developed during project execution. Any design modifications required for control algorithms or for the system software will be fully supported by personnel at the HFC manufacturing facility, and installation media will be supplied to site with supporting documentation. After completion of initial installation of the hardware and software, satisfactory performance of the control system will be demonstrated by execution of a comprehensive field performance test (FPT).

5.2.2 Startup and Transition

Offline software tools for automatic tuning of PID logic algorithms and monitoring of logic performance are part of the scope of supply for the project. After satisfactory completion of the field performance test, these tools will be used to support systematic

tuning of the analog control loops for dynamic operation. As before, one or more experience HFC engineers will be on site to support the overall startup process. This phase of the installation is expected to continue until site is authorized to begin preliminary fueling prior to the start of operating trials.

5.2.3 Operations Support

Except for warrantee issues, the scope of the present contract ends for each unit with the start of normal operations. HFC will offer on-site maintenance services as part of a separate contract to be established at that time.

5.3 MONITORING

Following on site acceptance of the plant control system hardware and software, the power plant management takes over primary responsibility for monitoring performance of the safety-critical software and for maintaining software configuration management. The following safeguards will be built in to the system to facilitate these tasks:

- The plant control system design includes the capability for internal performance monitoring. If critical components fail to operate or are out of established limits, the system will be designed to generate alarms to notify plant operators.

- Safety overrides will be designed in to both the safety-critical control logic and the system-level software to prevent or mitigate hazardous conditions.

- All control logic and system software will include identification data to permit verification of revision levels. The method of software installation will preclude any inadvertent replacement of one program (either a different version of the same program or a completely different program) by another.

In addition, HFC has established a 10CFR Part 21 program (QPP 16.3, "*10CFR Part 21 Reporting*"). If HFC detects a component defect or compliance that creates a significant safety hazard, this program requires HFC to make appropriate notifications.

5.4 MAINTENANCE

The control system software is composed of the control logic that makes up the application program and the system software that executes the application program. The system program code is primarily composed of compiled assembly language modules; the application program was generated as a set of logic diagrams and then converted into equation logic statements. The user has no access to the source code programs from which the system software was generated, and HFC will continue to maintain configuration control on this code. However, the logic diagrams, database, and equation files that make up the application code will all be supplied to the customer.

Following project completion, the end user takes over control of configuration management for the application code and the project database. The user can generate new logic diagrams or modify those that were supplied as part of the original system design. The new or modified logic diagrams can be converted to application logic, and these modifications can be installed in the system. The user is responsible to ensure that the record copies of the control system logic remain consistent with the application code that is actually installed in the system. Tools for performing this function are built into the system to support configuration management of the application, but the user must perform this function.

5.5 RETIREMENT AND NOTIFICATION

The plant control system will be designed for an expected operating life of 40 years. Three factors could force partial retirement prior to that time:

- Hardware obsolescence of a field device under control. Such a condition could force a modification or complete change of one or more control algorithms.

- Hardware obsolescence of a critical component within the plant control system. If a critical hardware component (e.g., a controller microprocessor) becomes obsolete, HFC might be required to replace that component with a device or assembly that was form, fit, and function equivalent.

- Detection of a hardware or software design error in a critical module would require replacement of that module as a minimum.

Replacement or modification of one or more application algorithms would not require early retirement of the control system hardware or system software. Accordingly, HFC would not become involved in such an effort unless it was in conjunction with a maintenance contract or similar agreement. However, replacement of a critical component within the plant control system might require replacement of some portion of the system software, even if the application code remains unchanged.

6.0 PLAN APPROVAL

This SSP will be reviewed by the project manager, lead application engineer, V&V team leader, and QA manger for internal approval. Following this internal approval, the plan will be submitted to the primary A&E contractor for approval. Final approval will be indicated by the sign-offs on the cover page.

HF Controls

COMMERCIAL GRADE SOFTWARE EVALUATION

QUALITY PROCESS PROCEDURE

Procedure No: **QPP 7.3** Revision **C**

Effective Date: 10/22/07

Prepared By: J.F. Ingram

Reviewed By: Y.B. Kim

Approved By: J. Stevens

Table of Contents

1.0	**PURPOSE AND SCOPE**	**3**
2.0	**ORGANIZATIONAL RESPONSIBILITIES**	**3**
3.0	**REFERENCE**	**4**
4.0	**GENERAL INFORMATION**	**4**
5.0	**PROCEDURE**	**5**
6.0	**QA RECORDS**	**10**
7.0	**ATTACHMENTS**	**10**

1.0 PURPOSE AND SCOPE

1.1 This procedure provides the requirements and responsibilities for the dedication of Commercial Grade Software (CGS) produced by HFC or a third party vendor for use in Nuclear Safety-Related/Class 1E applications.

1.2 This procedure is applicable to CGS for any system, sub-system, or item designed to be used in a Nuclear Safety-Related/Class 1E application, as defined by customer contract.

1.3 The critical characteristics of the software to be dedicated will be categorized into the three areas of physical, performance and dependability.

1.4 The CGE form created by this is to be maintained as a test document and as such must be configuration controlled per WI-ENG-812. The completed test document is to be controlled in accordance with QPP 17.1.

2.0 ORGANIZATIONAL RESPONSIBILITIES

2.1 Software Engineer: Responsible for the following:

- Classifying the item per Nuclear Safety-Related Class 1E criteria.
- Verifying origins per paragraph 4.1.3
- Determining the critical characteristics necessary to ensure proper function in final application.
- Determining the verification methods for the critical characteristics.
- Submitting the completed CGS Evaluation Form, Attachment 7.1 for verification and validation.

2.2 QA Director / Designee – Responsible for ensuring that critical characteristics are incorporated into audits, surveillances, tests, and inspections as necessary to provide objective evidence that systems, sub-systems, or components meet engineering criteria.

2.3 VP of Engineering / Designee– Approving the CGS Evaluation ensuring that the evaluation is technically correct.

2.4 Director of Projects / Designee – Assigning a qualified individual to perform the CGS evaluation, ensuring that the dedication form is completed, reviewed and approved by the evaluator and the V&V Team.

2.5 Verification and Validation Team – Responsible for ensuring that documentation and assessment of the CGS meets HFC written requirements and specifications.

3.0 REFERENCE

3.1 QAPM, "Quality Assurance Program Manual"

3.2 EPRI TR-106439, "Guideline on Evaluation and Acceptance of Commercial Grade Digital Equipment for Nuclear Safety Applications"

3.3 IEEE 7-4.3.2, Appendix C

3.4 QPP 3.1, "Design Control"

4.0 GENERAL INFORMATION

4.1 Definitions:

4.1.1 Critical Characteristics for Design – Those properties or attributes (requirements, process, and format), which are essential for the dedication of the performance and dependability of the software design. These characteristics are the identifiable and/or measurable attributes that provide assurance that the item will perform its design function.

4.1.2 Critical Characteristics for Acceptance – Critical characteristics that are identifiable and measurable attributes/variables of a Nuclear Safety-Related/Class 1E item, which, once verified, provide reasonable assurance or objective evidence that the software will perform as specified.

4.1.3 CGS Evaluation – An evaluation of CGS with the intent of determining if it meets the requirements and specifications of Nuclear Safety-Related Class 1E criteria set forth in EPRI and HFC's published documents.

4.1.4 CGS Assessment – An assessment of CGS for its suitability for nuclear safety-grade applications based on the assessment of operational history of the HFC product lines.

4.1.5 Method 1 - Special Tests and Inspections will be used to verify characteristics.

4.1.6 Method 2 - Design Documentation and/or Quality Commercial Grade survey of vendor will be used to verify characteristics.

4.1.7 Method 3 - Verification of Processes by HFC Quality will be used to verify characteristics.

4.1.8 Method 4 - Historical Performance Records will be used to verify characteristics.

5.0 PROCEDURE

5.1 The Software Engineer, knowledgeable of the functions of the parent component and the system in which it is to be installed, identifies the critical characteristics of the software design and acceptance for the specific CGS Evaluation. A qualified Software Engineer shall perform an engineering evaluation using the CGS Evaluation forms (Attachment 7.1).

5.2 Physical, performance and dependability characteristics shall be included in the critical characteristics for acceptance, as applicable. The following are to be used as examples of the criteria for acceptance and are not definitive of a specific software design: (See Attachment 7.3)

5.2.1 Physical Critical Characteristics for Acceptance:

A. Product/part Identification:

- Firmware part number and revision level
- Software part number revision level (embedded in code)

B. Physical characteristics of device interfaces, e.g.:

- Signal
- Data communications
- Human-machine Interfaces

5.2.1.1 Methods of Verification for Physical Attributes:

A. Firmware part identification and revision. *(Method 1).*

B. Verification that the software has met the Software/Firmware Requirements and Design Specifications (Methods 1 and/or 2).

5.2.2 Performance Criteria Characteristics for Acceptance:

A. Required functionality, e.g.:

- Input processing
- Specific functions or algorithms required
- Output signal requirements
- HMI functionality
- Test and diagnostic functions — on-line and off-line

B. Performance requirements related to the required functionality, e.g.:

- Response time
- Accuracy
- Range
- Stability
- Data throughput rate
- Interfaces protocol, including power, signal and data communication
- Effectiveness of HMI

C. Behavior under specific abnormal or faulted conditions:

- Response to specific abnormal conditions and events (ACEs)
- Fail-safe characteristics

5.2.2.1 Methods of Verification for Performance Characteristics (Method 1):

A. The evaluator may review the results of test cases that were performed by the software developer and may run supplementary tests as part of the dedication. Some characteristics may be verified through special stress or "challenge" testing performed by the developer or dedicator (e.g., tests of performance under conditions of high data rates or calculation burden).

B. Depending on the complexity of the item and the specific characteristics that need to be verified the Engineer may choose to run tests on only one or a sample of items, while others may need to be performed for each individual item evaluated. For characteristics that are verified only on a sample basis, there should be a sufficient understanding of the software design and the factors that influence those characteristics to ensure that the sampling is adequate (e.g., is the characteristic determined solely by the basic design and thus can be checked on only one or a few units).

5.2.3 Dependability Critical Characteristics for Acceptance:

A. The acceptance characteristics of reliability and maintainability as it is related to the required functionality and quality of the software design are to be evaluated for:

- Quality control of design
- Quality control of manufacture
- Failure management
- Compatibility with human operators, maintainers
- Fault tolerance
- Robustness
- Deterministic operation

B. Configuration control and traceability requirements of:

- Hardware
- Software
- Firmware (aspects of both hardware and software configuration control)
- Problem reporting

5.2.3.1 Expanded Dependability Acceptance Criteria

A. Criteria for reliability, availability and maintainability should be derived from the requirements of the intended application(s), if appropriate, specific criteria may be established such as numerical criteria for reliability or availability of required functions, or maintainability criteria of the software. If numerical criteria are used, the method of demonstration should be specified (e.g., statistical analysis of failure rate data from field experience).

B. The basic criterion for built-in quality is equivalent to the quality of software developed and applied under a 10 CFR

50 Appendix B program. Judgment of equivalent quality is based on a combination of:

- Design and design review processes, including software life cycle, V&V, etc.
- Software Design documentation
- Software Configuration management
- QA program and practices
- Software requirements definition and requirements traceability

C. Consideration of failure modes and ACEs in design and verification.

- Qualifications and experience of personnel involved in design and verification activities
- Product operating history
- Software testing by the vendor or dedicator

D. Minimum criterion for configuration control and traceability is that these are sufficient to support use of operating history data and to ensure the item delivered can be traced back to the documents reviewed as part of acceptance.

E. As a minimum, problem reporting must be sufficient to support use of product operating history and to allow the dedicator to carry out QPP16.3 (10 CFR 21) responsibilities. Specific criteria should be established (e.g., on coverage, timeliness, reporting to the right organization or department).

5.2.3.2 Acceptable Methods of Verification for Dependability Attributes. Comprehensive functional testing will be required to gain reasonable assurance that the CGS will perform reliably in a safety-related application:

A. Review historic operating data. Review and assess software design requirements and specifications. *(Method 4*)

B. Perform reliability analysis. *(Method* 2)

C. Review the development processes and documentation *(Method 2 or 3*):

- Design, development and verification processes
- Quality assurance program and practices
- V&V program and practices

D. Design reviews:
- Architecture review, code reviews, walkthroughs, use of analytical techniques, etc. (Method 2)
- Review of product operating history (from vendor, users, user groups, industry reports, etc.) (Method 4):
- Documented (records, traceable) (Method 4):
- Sufficient units, years in service. (Method 4):
- Successful error tracking shows good performance and device including software is stable. (Method 4):
- Relevant same or similar hardware/software configuration, functions used, operated similarly, etc. (Method 4):

E. Configuration control; review vendor configuration management program and practices. Examine actual practices, records. *(Method 2 or* 3)

F. Problem reporting: review developer procedures and practices. Assess performance record with previous customers. *(Method* 2)

5.3 The Software Engineer develops specific acceptance criteria for each critical characteristic for acceptance identified. Where applicable, the Software Engineer develops tolerances for these criteria. Typical acceptance criteria to consider may include:

- Software Requirements,
- Software Design Specifications,
- Software/Firmware Test Procedure,
- Qualification Test Procedures,
- Qualification Test Reports,
- Design review in accordance with WI-ENG-001 for CDS,
- Review of any operating history.

6.0 QA RECORDS

6.1 Attachment 7.1 is a quality record and shall be controlled in accordance with QPP 17.1.

7.0 ATTACHMENTS

7.1 "Commercial Grade Software Evaluation Form"

7.2 Example of a completed Commercial Grade Software Evaluation Form

ATTACHMENT 7.1

HF Controls

COMMERCIAL GRADE SOFTWARE EVALUATION

EVALUATION NUMBER: CGSE _____REV:_____

Revision History

Date	Revision	Author	Changes
07/23/03	0	B Han	Draft
08/01/03	A	B Han	Comments incorporated

Safety Related: [ ] YES [ ] NO

MANUFACTURER: _____

DESCRIPTION:_____

SOFTWARE NUMBER: _____**Rev:**_____

Assigned Engineer:_____Date: _____

Technical Reviewer:_____Date: _____

Approved By:_____Date: _____

V & V Review:_____Date: _____

ATTACHMENT 7.1

CGS CLASSIFICATION SHEET

CGSE_____ **Rev.**_____

Software P/N:_____**Rev:**_____**Description:** _____

1. What is the safety function of the parent component in which this item is to be installed?

_____
_____
_____
_____
_____

2. What is the function of the item?

_____
_____
_____
_____
_____

3. What are the ways in which the item should fail in actual service? Provide reference to ACE analysis _____.

_____
_____
_____
_____
_____

4. Would any of these postulated failures of the item prevent the parent component or associated components from performing their intended safety function?

_____
_____
_____
_____

Yes [ ] (Item has safety function)

No [ ] (Item has non-safety function)

COMMERCIAL GRADE SOFTWARE EVALUATION

ATTACHMENT 7.1

CGS CLASSIFICATION SHEET

CGSE:_____ Rev:_____
Software P/N:_____Rev:_____Description: _____

Method (Mark all that apply}

	YES	NO
1. Is the item not subject to design or specification requirements that are unique to facilities or activities licensed or regulated by the USNRC?	[ ]	[ ]
2. Is the item used in applications other than facilities or activities licensed or regulated by the USNRC?	[ ]	[ ]
3. Is the item ordered from manufacturer/supplier on the basis of specifications set forth in the manufacturer's published description?	[ ]	[ ]

A "YES" in all three criteria is required for the item to be classified as commercial grade.

Commercial Grade Software Item [ ] [ ]

COMMERCIAL GRADE SOFTWARE EVALUATION

ATTACHMENT 7.1

CGS CLASSIFICATION SHEET

CGSE:_____ **Rev:**_____

Software P/N:_____**Rev:**_____**Description:** _____

CRITICAL CHARACTERISTICS FOR SOFTWARE DESIGN:

_____

_____

_____

_____

_____

_____

_____

CRITICAL CHARACTERISTICS FOR ACCEPTANCE:

_____

_____

_____

_____

_____

_____

_____

METHOD (MARK ALL THAT APPLY):

[ ] Method 1 - Special Tests and Inspections.

[ ] Method 2 - Design Documentation and/or Quality Commercial Grade survey of vendor.

[ ] Method 3 - Verification of Processes by HFC Quality.

[ ] Method 4 - Historical Performance Records.

COMMERCIAL GRADE SOFTWARE EVALUATION

ATTACHMENT 7.1

CGS CRITICAL CHARACTERISTICS SHEET

Software P/N:_____**Description:**_____**CGSE** _____ **Rev**____ **Rev**_____

Physical Critical Characteristic	Acceptance Criteria	Method of Verification
CC#		
CC#		
CC#		
CC#		
CC#		

COMMERCIAL GRADE SOFTWARE EVALUATION

ATTACHMENT 7.1

CGS CRITICAL CHARACTERISTICS SHEET

Software P/N:_____**Description:**_____**CGSE** _____ **Rev**____ **Rev**_____

Performance Critical Characteristic	**Acceptance Criteria**	**Method of Verification**
CC#		
CC#		
CC#		
CC#		

Attachment 7.1

CGS CRITICAL CHARACTERISTICS SHEET

Software P/N:_____**Description:**_____**CGSE** _____ **Rev**____ **Rev**_____

Dependability Critical Characteristic	**Acceptance Criteria**	**Method of Verification**
CC#		
CC#		
CC#		
CC#		
CC#		

COMMERCIAL GRADE SOFTWARE EVALUATION

Attachment 7.1

CGS M&TE SHEET

Software P/N:_____**Description:**_____**CGSE** _____ **Rev**____ **Rev**_____

Record all M&TE used for this test:

M&TE CIN#	M&TE Desc.	Cal. Due Date	M&TE CIN#	M&TE Desc.	Cal. Due Date

Condition Report (CR): [ ] Yes [ ] No [ ] N/A If yes, identify below:
CR# [ ]Acceptable [ ] Not Acceptable Initials/Date: _____
CR# [ ]Acceptable [ ] Not Acceptable Initials/Date: _____
CR# [ ]Acceptable [ ] Not Acceptable Initials/Date: _____
Other:

ATTACHMENT 7.1

CGS CRITICAL CHARACTERISTICS RECORD SHEET

Software P/N_____ **Description:**_____**CGSE** _____ **Rev**____
Rev_____ **Sheet 1 of** ____

CC#	P/F/N	Preformed By	Date	Verified By	Date

P = Pass, F = Fail, N = Not Applicable

Characteristics Record Sheet Competition:

Verified By _____ **Approved By** _____
Date:_____ **Date:**_____

ATTACHMENT 7.2
COMMERCIAL GRADE SOFTWARE EVALUATION

EVALUATION NUMBER: CGSE 1234
REVISION: A

Safety Related**:** [X] YES [ ] NO

MANUFACTURER: HF Controls

DESCRIPTION: Controller Operating System

SOFTWARE NUMBER: 80040001 Rev: A

EXAMPLE

Assigned Engineer: Steve Yang Date 4/26/05

Technical Reviewer: Jon Taylor Date 4/27/05

Approved By: Allen Hsu Date: 4/28/05

V & V Review: Bob Cain Date: 4/27/05

COMMERCIAL GRADE SOFTWARE EVALUATION

ATTACHMENT 7.2
CGS CLASSIFICATION SHEET

CGSE 1234 Rev A
Software P/N 80004001 Rev A Description: Controller Operating System

1. What is the safety function of the parent component in which this item is to be installed?

   Power Plant Control _____

2. What is the function of the item?

   Control Functional Operation of Safety System Component_____

EXAMPLE

3. What are the ways in which the item should fail in actual service? Reference ACE number 2307.

   Controller Stall_____

4. Would any of these postulated failures of the item prevent the parent component or associated components from performing their intended safety function?
   __**YES**_____

   Yes [**X**] (Item has safety function)

   No [ ] (Item has non-safety function)

COMMERCIAL GRADE SOFTWARE EVALUATION

ATTACHMENT 7.2
CGS CLASSIFICATION SHEET

CGSE 1234 Rev A

Software P/N 80004001 Rev A Description: Controller Operating System

Method (Mark all that apply}	**YES**	**NO**
1. Is the item not subject to design or specification requirements that are unique to facilities or activities licensed or regulated by the USNRC?	[**X**]	[ ]
2. Is the item used in applications other than facilities or activities licensed or regulated by the USNRC?	[**X**]	[ ]
3. Is the item ordered from manufacturer/supplier on the basis of specifications set forth in the manufacturer's published description?	[**X**]	[ ]

EXAMPLE

A "YES" in all three criteria is required for the item to be classified as commercial grade.

Commercial Grade Software Item [**X**] [ ]

COMMERCIAL GRADE SOFTWARE EVALUATION

ATTACHMENT 7.2
CGS CLASSIFICATION SHEET

CGSE 1234 Rev A
Software P/N 80004001 Rev A Description: Controller Operating System

CRITICAL CHARACTERISTICS FOR SOFTWARE DESIGN:
a. Model number
b. Software revision Number
c. Electrical power
d. Grounding and shield termination provisions
e. Signal conditioning
f. Response time
g. Human-Machine Interface
h. EMI
i. Behavior under faulted conditions
j. Quality of design.

CRITICAL CHARACTERISTICS FOR ACCEPTANCE:
a. Vendor model# for main unit, plus model #s for each module to be dedicated.
b. Vendor Software version number for each module or software.
c. Power specifications for each configuration to be dedicated.
d. Acceptance per customer specification.
e. Acceptance per customer specification for each configuration.
f. Acceptance per customer specification for each configuration.
g. Acceptance per customer specification for operational requirements.
h. Acceptance per customer specification using a suitable method.
i. Acceptance per specific requirements regarding fail-safe conditions.
j. Verification of an acceptable QA program.

EXAMPLE

METHOD (MARK ALL THAT APPLY):

[X] Method 1 - Special Tests and Inspections.

[X] Method 2 - Design Documentation and/or Quality Commercial Grade survey of vendor.

[ ] Method 3 - Verification of Processes by HFC Quality.

[X] Method 4 - Historical Performance Records.

COMMERCIAL GRADE SOFTWARE EVALUATION

Attachment 7.2

CGS CRITICAL CHARACTERISTICS SHEET

Software P/N 80004001 Description: Controller Operating System CGSE 1234 Rev A
Rev A

Physical Critical Characteristic	**Acceptance Criteria**	**Method of Verification**
CC#1 Model Number	Vendor model# for main unit plus model #s for each module to be dedicated.	Method 1 – Receipt inspection of each unit received.
CC#2 Software revision Number	Vendor Software version number for each module of software.	Method 1 – Receipt inspection of each unit received.
CC#3 Electrical Power	Power specifications for each configuration to be dedicated.	Method 1 – Receipt inspection of each unit received. Testing to verify correct interfaces/connection types.
CC#4 Grounding and shield termination provisions	Acceptance per customer specification	

EXAMPLE

COMMERCIAL GRADE SOFTWARE EVALUATION

ATTACHMENT 7.2

CGS CRITICAL CHARACTERISTICS SHEET

Software P/N 80004001 Description: Controller Operating System CGSE 1234 Rev A
Rev A

Performance Critical Characteristic	**Acceptance Criteria**	**Method of Verification**
CC#5 Signal conditioning	Acceptance per customer specification for each configuration.	Methods 1, 2, and 3 Verified through a combination of: • Review of documented vendor tests. • Site acceptance testing. • Review of design.
CC#6 Response time	Acceptance per customer specification for each configuration.	Methods 1, 2, and 3 Review of system design including input sample rate, processing time, and total cycle time. Final testing of integrated system
CC#7 Human-Machine Interface	Acceptance per customer specification for operational requirements.	Methods 1, 2, and 3 Special testing by customer or integrator and human factors evaluation by customer engineering and operations.
CC#8 EMI	Acceptance per customer specification using a suitable method.	Methods 1, 2, and 3 Laboratory testing of controller susceptibility and emissions in configurations that mimic as close as possible the installed configurations. Post installation testing.
CC#9 Behavior under faulted conditions	Acceptance per specific requirements regarding fail-safe conditions.	Methods 1, 2, and 3 Detailed review of design and hardware/software architecture during CG evaluation, behavior under abnormal and faulted conditions

EXAMPLE

ATTACHMENT 7.2

CGS CRITICAL CHARACTERISTICS SHEET

Software P/N 80004001 Description: Controller Operating System CGSE 1234 Rev A Rev A

Dependability Critical Characteristic	Acceptance Criteria	Method of Verification
CC#10 Quality of design.	Verification of an acceptable QA program.	Method 4 Review of vendors QA program against relevant standards. Review of vendor procedures and practices for digital system/software development. Thread audit to check actual practices

EXAMPLE

COMMERCIAL GRADE SOFTWARE EVALUATION

CGS CRITICAL CHARACTERISTICS SHEET

Attachment 7.1

Software P/N 80004001 **Description: Controller Operating System** **CGSE 1234** **Rev A**
Rev A

Record all M&TE used for this test:

M&TE CIN#	M&TE Desc.	Cal. Due Date	M&TE CIN#	M&TE Desc.	Cal. Due Date

Condition Report (CR): [**X**] Yes [ ] No [ ] N/A If yes, identify below:
CR# **2005-0001** [**X**]Acceptable [ ] Not Acceptable Initials/Date: **EF 4/20/05**
CR# **2005-0030** [**X**]Acceptable [ ] Not Acceptable Initials/Date: **JT 4/25/05**
Other:

EXAMPLE

CGS CRITICAL CHARACTERISTICS RECORD SHEET

Attachment 7.1

Software P/N 80004001 Description: Controller Operating System CGSE 1234 Rev A
Rev A Sheet 1 of 1

CC#	P/F/N	Performed By	Date	Verified By	Date
1	**P**	Steve Yang	3/10/05	B Cain	3/11/05
2	**P**	Steve Yang	3/16/05	B Cain	3/20/05
3	**P**	Steve Yang	3/18/05	B Cain	3/23/05
4	**P**	J. Taylor	4/02/05	S Yang	4/10/05
5	**P**	Steve Yang	4/10/05	J Taylor	4/13/05
6	**P**	J Taylor	4/12/05	B Cain	4/13/05
7	**P**	Steve Yang	4/18/05	E. Fisher	4/18/05
8	**P**	Steve Yang	4/12/05	B Cain	4/13/05
9	**P**	J Taylor	4/12/05	E Fisher	4/13/05
10	**P**	E Fisher	3/16/05	B Cain	4/13/05

EXAMPLE

P = Pass, F = Fail, N = Not Applicable

Verified By _____ **Approved By** _____
Date:_____ **Date:**_____